

UACM

Universidad Autónoma
de la Ciudad de México

Nada humano me es ajeno

COLEGIO DE CIENCIA Y TECNOLOGÍA

LICENCIATURA EN INGENIERÍA DE SISTEMAS ELECTRÓNICOS Y DE
TELECOMUNICACIONES

**BCI Mobile - Telefonía celular para personas con discapacidad
motora y del habla en México**

TRABAJO RECEPCIONAL
PARA OBTENER EL TÍTULO DE LICENCIADAS EN
INGENIERÍA EN SISTEMAS ELECTRÓNICOS Y DE TELECOMUNICACIONES

PRESENTA
**RUTH VALENCIA REAL
GRACIELA HERNÁNDEZ RAMÍREZ**

Director del trabajo recepcional
Mtro. Miguel Ángel Vivanco Gallardo

Ciudad de México, Septiembre de 2016

SISTEMA BIBLIOTECARIO DE INFORMACIÓN Y DOCUMENTACIÓN



UNIVERSIDAD AUTÓNOMA DE LA CIUDAD DE MÉXICO COORDINACIÓN ACADÉMICA

RESTRICCIONES DE USO PARA LAS TESIS DIGITALES

DERECHOS RESERVADOS[©]

La presente obra y cada uno de sus elementos está protegido por la Ley Federal del Derecho de Autor; por la Ley de la Universidad Autónoma de la Ciudad de México, así como lo dispuesto por el Estatuto General Orgánico de la Universidad Autónoma de la Ciudad de México; del mismo modo por lo establecido en el Acuerdo por el cual se aprueba la Norma mediante la que se Modifican, Adicionan y Derogan Diversas Disposiciones del Estatuto Orgánico de la Universidad de la Ciudad de México, aprobado por el Consejo de Gobierno el 29 de enero de 2002, con el objeto de definir las atribuciones de las diferentes unidades que forman la estructura de la Universidad Autónoma de la Ciudad de México como organismo público autónomo y lo establecido en el Reglamento de Titulación de la Universidad Autónoma de la Ciudad de México.

Por lo que el uso de su contenido, así como cada una de las partes que lo integran y que están bajo la tutela de la Ley Federal de Derecho de Autor, obliga a quien haga uso de la presente obra a considerar que solo lo realizará si es para fines educativos, académicos, de investigación o informativos y se compromete a citar esta fuente, así como a su autor ó autores. Por lo tanto, queda prohibida su reproducción total o parcial y cualquier uso diferente a los ya mencionados, los cuales serán reclamados por el titular de los derechos y sancionados conforme a la legislación aplicable.

I. Dedicatoria.

La vida es difícil, pero la mía es plena y feliz porque tengo dos brillantes faros que me han guiado por la senda. Qué fácil es todo cuando hay apoyo, sin ustedes no sería lo que soy, ¡No estaría donde estoy!

Con cariño para nuestros padres:

Pedro Hernández Corona

Julia Ramírez Francisco

Adan Valencia Campos

Rosa María Real Preciado.

II. Agradecimientos.

Este trabajo de tesis realizado en la Universidad Autónoma de la Ciudad de México plantel Cuauhtepac, es un esfuerzo en el cual participaron distintas personas opinando, corrigiendo, teniéndonos paciencia, dándo ánimos, acompañándonos en los momentos de crisis y en los momentos de felicidad. Este trabajo nos ha permitido aprovechar la experiencia de muchas personas a las que les deseamos agradecer en este breve apartado.

En primer lugar, a nuestro director de tesis, Miguel Ángel Vivanco Gallardo, nuestro más amplio agradecimiento por su valiosa orientación, supervisión, apoyo y tiempo en la elaboración de esta tesis, así como de su experiencia y educación que ha sido nuestra fuente de motivación y de curiosidad durante estos años.

Especial reconocimiento merece el interés mostrado por mi trabajo y las sugerencias recibidas de los profesores: Rubén Flores Leal, Fernando Cruz Martínez, Federico Juárez Almaraz, Jorge Enrique Wals Selvas y Rocío Idalia Romero Ángeles, con la que nos encontramos en deuda por el apoyo, asesorías y por su dedicación y tiempo en nosotras depositadas.

También nos gustaría agradecer al profesor Marco Antonio Prado Zayago, por el préstamo del Laboratorio de matemáticas para la realización de pruebas.

Un agradecimiento muy especial merece la comprensión, paciencia y el ánimo recibidos de nuestra familia y amigos.

A todos ellos, muchas gracias.

III. Resumen.

La habilidad para comunicarse con otras personas es una de las características principales del ser humano; a través de la comunicación es posible expresar ideas, deseos, sentimientos y desarrollarse en las actividades de la vida diaria, es por ello que mediante este trabajo se pretende facilitar el acceso a la telefonía móvil para aquellas personas que se les dificulta el habla y que dependen de alguien para comunicarse o dependen de algún aparato para llevar a cabo la comunicación; particularmente para aquellas personas que cuentan con discapacidad motora y del habla, brindándoles así una mejor independencia para comunicarse por sí mismos.

La tecnología permite diseñar y crear; bienes y servicios que faciliten la adaptación al entorno del usuario con el fin de satisfacer necesidades esenciales de la vida diaria, en este caso, la necesidad de comunicación de las personas que cuentan con discapacidad motora y del habla.

La solución propuesta incluye un sistema de interacción hombre-máquina capaz de traducir intenciones y emociones. El funcionamiento básico de éste sistema hombre-máquina es proporcionar un medio físico para que señales producidas por la actividad cerebral del usuario lleguen al exterior; una vez recibidas las señales neuronales, es posible que las personas con discapacidad motora y del habla puedan utilizarlas para recuperar su independencia y control. Para poder llevar a cabo la interacción hombre-máquina se hará uso de una interfaz cerebro-computador (BCI) del cual, se obtendrán señales de tipo electroencefalográficas (EEG), así como también se hará uso de redes inalámbricas de área personal (Wireless Personal Access Network-WPAN); se pretende crear un sistema de comunicación que ayude a los usuarios con discapacidad motora y del habla a fomentar su independencia para poderse comunicar por sí mismos con otras personas y también con sistemas.

Cabe enfatizar que la BCI no lee la mente, es decir, que no extrae los pensamientos de las personas con discapacidad; sino que “es una habilidad que el usuario y el sistema, cualquiera que éste sea, adquiere y mantiene”.

Palabras clave – Interfaz cerebro-computador, Sistema de interacción hombre-máquina, Señales electroencefalográficas, redes inalámbricas de área personal.

IV. Figuras

Figura 1. INEGI Censo de población y vivienda 2010. [7].....	5
Figura 2. INEGI Censo de población y vivienda 2010 [8].....	5
Figura 3. INEGI Censo de población y vivienda 2010 [7].....	6
Figura 4. INEGI Censo de población y vivienda 2010 [7].....	7
Figura 5. INEGI Censo de población y vivienda 2010 [7].....	8
Figura 6. Lóbulos cerebrales. [11].....	12
Figura 7. Ondas cerebrales típicas. [11]	14
Figura 8. Sistema Internacional 10-20 para la colocación de electrodos extracraneales. [14]	16
Figura 9. Pila de protocolos de Bluetooth. [33]	23
Figura 10. Pila de protocolo de ZigBee. [49]	28
Figura 11. Arquitectura del estándar GSM [53]	32
Figura 12. Estructura pantalla TFT. [78]	40
Figura 13. Composición de un pixel. [79].....	41
Figura 14. Organigrama para la implementación del sistema.	44
Figura 15. Sistema de comunicación.	45
Figura 16. Componentes del sistema BCI Mobile.	46
Figura 17. Lóbulos Cerebrales.	48
Figura 18. Dispositivos que componen al Sistema BCI Mobile.	53
Figura 19. Dispositivos en físico que comprenden al sistema BCI Mobile.	54
Figura 20. Ejecución del código para la configuración del Bluesmirf.	56
Figura 21. Bluesmirf Modo comando.....	57
Figura 22. Trama Larga de la MindWave Mobile. [86].....	61
Figura 23. Trama corta de la MindWave Mobile. [86]	61
Figura 24. Gráficas de detección del parpadeo voluntario e involuntario.	64
Figura 25. Comunicación inalámbrica de la MindWave Mobile y el Bluesmirf.....	71
Figura 26. Comunicación serial entre el Bluesmirf y el Arduino ADK.....	71
Figura 27. Trama corta y Trama larga en CoolTerm.....	73
Figura 28. Comunicación entre el Arduino ADK y la pantalla Robot LCD.	76
Figura 29. Comunicación del Arduino ADK y el Shield Arduino GSM.....	79
Figura 30. Primera prueba del Sistema BCI Mobile con Jessica Guadalupe.....	81
Figura 31. Segunda prueba del sistema BCI Mobile con Jessica Guadalupe.	82
Figura 32. Jessica Guadalupe haciendo uso del Sistema BCI Mobile.....	82

V. Tablas

Tabla 1. Clasificación de electrodos. [11]	15
Tabla 2. Interfaces Cerebro Computador en el mercado.	19
Tabla 3. Características principales del estándar Bluetooth. [34] y [35]	23
Tabla 4. Interfaces Bluetooth en el mercado.....	26
Tabla 5. Características principales del estándar Zigbee. [49] y [50]	28
Tabla 6. Interfaces ZigBee en el mercado.	29
Tabla 7. GSM en el mercado.	34
Tabla 8. Herramientas de desarrollo basados en microcontroladores existentes en el mercado.....	39
Tabla 9. Pantallas TFT en el mercado.	42
Tabla 10. Tabla comparativa de interfaces cerebro computador.	47
Tabla 11. Tabla comparativa de interfaces Bluetooth.....	49
Tabla 12. Tabla comparativa de herramientas de desarrollo basados en microcontroladores.	50
Tabla 13. Tabla comparativa de pantallas TFT.	51
Tabla 14. Tabla comparativa de módulos GSM.....	52
Tabla 15. Cotización del sistema BCI Mobile.....	53
Tabla 16. Conexión con el Bluesmirf y el Arduino ADK.....	56
Tabla 17. Componentes y códigos de la trama larga. [86].....	60
Tabla 18. Componentes y códigos de la trama corta. [86].....	61

VI. Contenido

I. Dedicatoria.	I
II. Agradecimientos.	I
III. Resumen.	II
IV. Figuras	III
V. Tablas	IV
VI. Contenido	V
Capítulo I	
1. Introducción.	1
1.1. Estado del Arte.	2
1.2. Justificación.	4
1.2.1. Estadísticas de discapacidad en México.	4
2. Planteamiento del problema.	9
2.1. Descripción del Problema.	9
2.2. Objetivo General.	9
2.3. Objetivos Específicos.	10
Capítulo II	
3. Conceptos Teóricos.	11
3.1. Cerebro.	11
3.1.1. Función.	11
3.1.2. Características principales.	12
3.2. Señales cerebrales.	13
3.2.1. Señales electroencefalográficas.	13
3.3. Interfaz cerebro-computador.	17
3.3.1. Función.	17
3.3.2. Historia.	17
3.3.3. Clasificación.	17
3.3.4. BCI en el mercado.	18
3.4. Redes Inalámbricas de Área Personal (WPAN).	20
3.5. Estándares de comunicación inalámbrica (IEEE 802.15).	21
3.6. Estándar de comunicación inalámbrica IEEE 802.15.1 – Bluetooth.	22
3.6.1. Historia.	22
3.6.2. Definición.	22
3.6.3. Composición.	22

3.6.4.	Especificaciones básicas de Bluetooth.	23
3.6.5.	Pila de protocolos	23
3.6.6.	Perfiles	25
3.6.7.	Bluetooth en el mercado.	25
3.7.	Estándar de comunicación inalámbrica IEEE 802.15.4 – ZigBee.	27
3.7.1.	Historia.	27
3.7.2.	Definición.	27
3.7.3.	Composición.	27
3.7.4.	Especificaciones básicas de ZigBee.	27
3.7.5.	Pila de protocolos.	28
3.7.6.	Perfiles.	29
3.7.7.	ZigBee en el mercado.	29
3.8.	GSM- Sistema Global para Comunicaciones Móviles.	30
3.8.1.	Historia.	30
3.8.2.	Definición.	30
3.8.3.	Características	30
3.8.4.	Servicios.	31
3.8.5.	Bandas de frecuencias	31
3.8.6.	Tipos de Acceso múltiple	31
3.8.7.	Modulación	32
3.8.8.	Arquitectura.	32
3.8.9.	Ventajas.	33
3.8.10.	Desventajas.	34
3.8.11.	GSM en el mercado.	34
3.9.	Microcontroladores.	35
3.9.1.	Historia.	35
3.9.2.	Definición.	35
3.9.3.	Composición.	35
3.9.4.	Arquitectura de los microcontroladores.	37
3.9.5.	Herramientas de desarrollo basados en microcontroladores en el mercado.	39
3.10.	Pantalla TFT (Thin-film transistor).	40
3.10.1.	Historia.	40
3.10.2.	Definición.	40
3.10.3.	Composición.	40
3.10.4.	Ventajas.	41
3.10.5.	Pantallas TFT en el mercado.	41

Capítulo III

4.	Marco Metodológico.	43
4.1.	Diseño del Sistema.	45
4.1.1.	Funcionamiento y composición del sistema BCI Mobile.	46
4.1.2.	Especificaciones técnicas.	47
4.1.3.	Cotización del Sistema BCI Mobile.	53
4.2.	Programación.	54
4.2.1.	Programación para la configuración del Bluesmirf.	54

4.2.2.	Programación del Sistema BCI Mobile.	58
--------	--------------------------------------	----

Capítulo IV

5.	Pruebas y Resultados.	71
5.1.	Pruebas.	71
5.1.1.	Prueba 1: Adquisición y transferencia de datos.	71
5.1.2.	Prueba 2: Manipulación de los datos adquiridos.	74
5.1.3.	Prueba 3: Creación del menú.	76
5.1.4.	Prueba 4: Envío de mensajes.	79
5.2.	Resultados.	81
6.	Conclusiones.	83
6.1.	Conclusiones de Graciela Hernández Ramírez.	83
6.2.	Conclusiones de Ruth Valencia Real.	86
	Referencias.	89

Capítulo I

1. Introducción.

El siguiente trabajo de tesis presenta un sistema híbrido de comunicación llamado “Sistema BCI Mobile”, este sistema hace uso de una Interfaz cerebro computador (BCI) que permite la lectura de señales electroencefalográficas (EEG), mediante la cual se obtuvieron valores para la detección del parpadeo voluntario del usuario; en nuestro caso está dirigido a personas con discapacidad motora y del habla en México. Por consiguiente, el sistema será de fácil uso para el usuario por lo que se implementa una interfaz cerebro computador en forma de diadema, la cual cuenta con un electrodo para la lectura de las señales EEG y detección del parpadeo en el usuario con discapacidad motora y del habla, después de ello a partir de los datos obtenidos del parpadeo podrá ser controlado un sistema de comunicación para realizar el envío de mensajes cortos SMS por medio de telefonía celular haciendo uso de la tecnología GSM.

El sistema BCI Mobile se compone de programación, por lo que se utilizó el software Arduino para la manipulación de los datos obtenidos por la diadema BCI MindWave Mobile y hacer uso de otros componentes tales como: un microcontrolador, una interfaz GSM (interfaz física del Sistema Global de comunicaciones Móviles), una antena, una interfaz Bluetooth y una pantalla TFT. En la pantalla TFT se muestra un menú para que el usuario tenga opción escoger el tipo de mensaje a enviar y el destinatario; cabe mencionar que todas las opciones de dicho menú son predefinidas.

Se realizaron dos pruebas del sistema BCI Mobile en la universidad Autónoma de la Ciudad de México plantel Cuauhtémoc, dando como resultado que el sistema BCI Mobile funciona de manera correcta y es de fácil uso para las personas que cuentan con alguna discapacidad motora y del habla, ya que solo se hace uso del parpadeo para la manipulación del sistema.

Para mayor información sobre esta tesis, se diseñó una página web con la siguiente dirección: <http://uacmtesis6.wix.com/bcimobile>

1.1. Estado del Arte.

Los trabajos de investigación que actualmente han estudiado la BCI y han dado soluciones para la comunicación en personas con discapacidad motora son los siguientes:

- I. La tesis titulada **“Interfaz Cerebro – Computadora para el Control de un Cursor Basada en Ondas Cerebrales”**, se plantea un proyecto de investigación que permita el desarrollo de una interfaz para la comunicación entre el usuario y una computadora haciendo uso de sus ondas cerebrales electroencefalográficas (EEG), la investigación se enfoca en la extracción de las características de las señales EEG para el desarrollar una interfaz BCI y tener control de un cursor en pantalla mediante comandos obtenidos de las lecturas EEG. Debido a que no obtuvieron buenos resultados con señales EEG, se pasó a explorar señales de tipo Electro-Oculográficas, que son señales provenientes de los movimientos oculares. [1]
- II. El artículo de **Computational Intelligence and Neuroscience** titulado **“EGG-based Brain Computer Interfaces”**, se encuentra una descripción detallada del desarrollo de la BCI haciendo uso del **“10-20 international electrode system”**; realizan un sistema ABI (Adaptative Brain Interface), el cual describen, se muestra detalladamente la manera en la que adquirieron los datos, extrajeron características principales de las señales para llevar a cabo el entrenamiento de su sistema, ya que su principal objetivo es que personas con discapacidad puedan controlar una BCI en 30 minutos, lo cual es de gran utilidad para esta investigación ya que ayudará en el entendimiento y comprensión de los términos, nomenclatura y características que compone a una BCI. [2]
- III. La tesis llamada **“Interfaz BCI de altas prestaciones basada en la detección y procesamiento de la actividad cerebral (BCI-DEPRACAP)”**, presenta el diseño y desarrollo de un sistema BCI (Brain-Computer Interface) basado en la detección y procesamiento de la actividad cerebral (BCI-DEPRACAP). Para conseguir este objetivo se parte de una profunda revisión histórica y tecnológica del sector, a partir de la cual se realizan dos estudios previos sobre el uso del P300 (la P es la posición de amplitud pico positivo que ocurre unos 300ms después del estímulo visual) y los potenciales evocados visuales de estado estable en sistemas BCI. [3]
- IV. El proyecto titulada **“Sistema de comunicación móvil, accesible, abierto y de bajo coste, para personas con discapacidad motora- ACCEMOVIL”**; Este trabajo se desarrolló en el marco del proyecto AcceMovil, cuyo objetivo es mejorar

la accesibilidad del teléfono móvil para las personas con discapacidad motora, con afectación en los miembros superiores; el sistema está basado en una plataforma abierta como es Android que, mediante la interconexión de conmutadores comerciales externos, permitiendo al usuario acceder a funciones básicas de la telefonía móvil, como son: la recepción y realización de llamadas, el envío y recepción de mensajes (SMS), el uso de despertador y el uso de una agenda. [4]

- V. En la tesis llamada ***“Aplicación ICC: silla de ruedas para cuadripléjicos controlada con señales electroencefalográficas del movimiento voluntario de los parpados”***, presenta el diseño y desarrollo de un sistema ICC (Interfaz Cerebro Computador) basado en la detección y procesamiento de las señales electroencefalográficas. Para que la ICC controle una silla de ruedas y se realicen movimientos de rotación hacia la derecha, izquierda, avance adelante y atrás; se crea también un sistema auxiliar con una pantalla de visualización. Esta tesis hace uso de algoritmos utilizando lenguaje de programación de bajo nivel y el uso de Matlab. [5]

1.2. Justificación.

Nuestra sociedad alberga un amplio número de personas que de alguna u otra manera presentan alteraciones lingüísticas; ya sea para producir mensajes, para recibirlos o para procesarlos, y algunas otras condiciones como deficiencia física por enfermedad congénita o producida por accidente; donde sus principales problemas son: dificultades de coordinación, alcance limitado y fuerza reducida; en general aquellas personas que tienen limitaciones al caminar o moverse, y que cuentan con alguna alteración lingüística, sin problemas mentales o de aprendizaje. [6]

En el mundo con el que interactuamos tiene un alto contenido sonoro, visual y motor, al que una parte de la población no tiene acceso de ninguna manera, y es por ello que se busca crear una herramienta cuyo objetivo sea el facilitar la comunicación entre las personas con discapacidad motora y del habla, mediante la ayuda de la tecnología aplicada a BCI y telefonía celular.

1.2.1. Estadísticas de discapacidad en México.

A continuación se muestran estadísticas del censo 2010 de acuerdo al Instituto Nacional de Estadística y Geografía (INEGI), en el cual podemos demostrar el impacto y la importancia del sistema de comunicación BCI Mobile hacia las personas que cuentan con algún tipo de discapacidad motora y del habla:

- ***Causa de discapacidad.***

Las causas que originan la discapacidad son diversas, en el Censo 2010 se identificaron cinco causas básicas: nacimiento, enfermedad, accidente, edad avanzada y otras causas. Las enfermedades y la edad avanzada explican cerca de 60% de los casos de discapacidad, mientras que las causas relacionadas con el nacimiento y los accidentes ocupan el tercero y cuarto lugar respectivamente. [7]

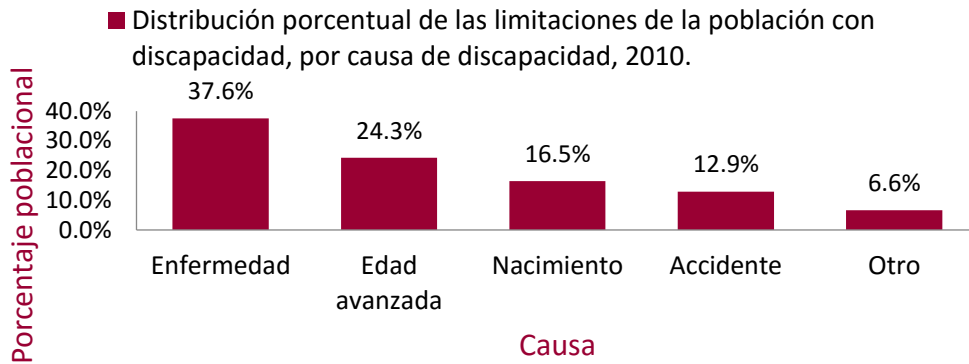


Figura 1. INEGI Censo de población y vivienda 2010. [7]

- *Situación demográfica.*

El Censo de Población y Vivienda 2010, en su cuestionario ampliado¹, en México habitan alrededor de 112 millones de personas. De ellas, aproximadamente 5 millones 739 mil personas (5.1% de la población total) reportan tener dificultad o limitación para realizar una o más actividades (caminar o moverse, ver, escuchar, hablar, atender el cuidado personal, déficit de atención, mentales). (INEGI 2010) [7]

En 2010, el número de personas que tienen alguna limitación física o mental corresponde a 5 millones 739 mil personas, de los cuales 51.1% son mujeres y 48.9% son hombres. [8]

Porcentaje de la población con discapacidad por sexo 2010.

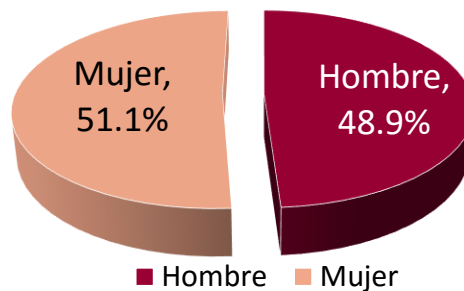


Figura 2. INEGI Censo de población y vivienda 2010 [8]

¹ Durante el Censo de Población y Vivienda 2010 se utilizaron dos tipos de cuestionarios: uno básico y otro ampliado. El último incluyó las preguntas del primero e incorporó otras para profundizar en diversos temas. Se aplicó solo un tipo de cuestionario por vivienda. Con el cuestionario ampliado se censaron alrededor de 2.9 millones de viviendas seleccionadas mediante un muestreo probabilístico, lo cual garantiza que esta muestra permite hacer inferencias sobre las características de toda la población.

- *Distribución territorial.*

A nivel nacional, las entidades federativas con mayor cantidad de habitantes, concentran también el más alto número de gente con discapacidad. Sin embargo, la proporción de personas con discapacidad al interior de cada entidad no responde a la cantidad de población sino a factores como: la estructura por edad de la población, las condiciones socioeconómicas y sanitarias de la zona, etc. (INEGI 2010) [7]

Las entidades federativas con mayor cantidad de población con discapacidad son: el Estado de México y la Ciudad de México. [7]

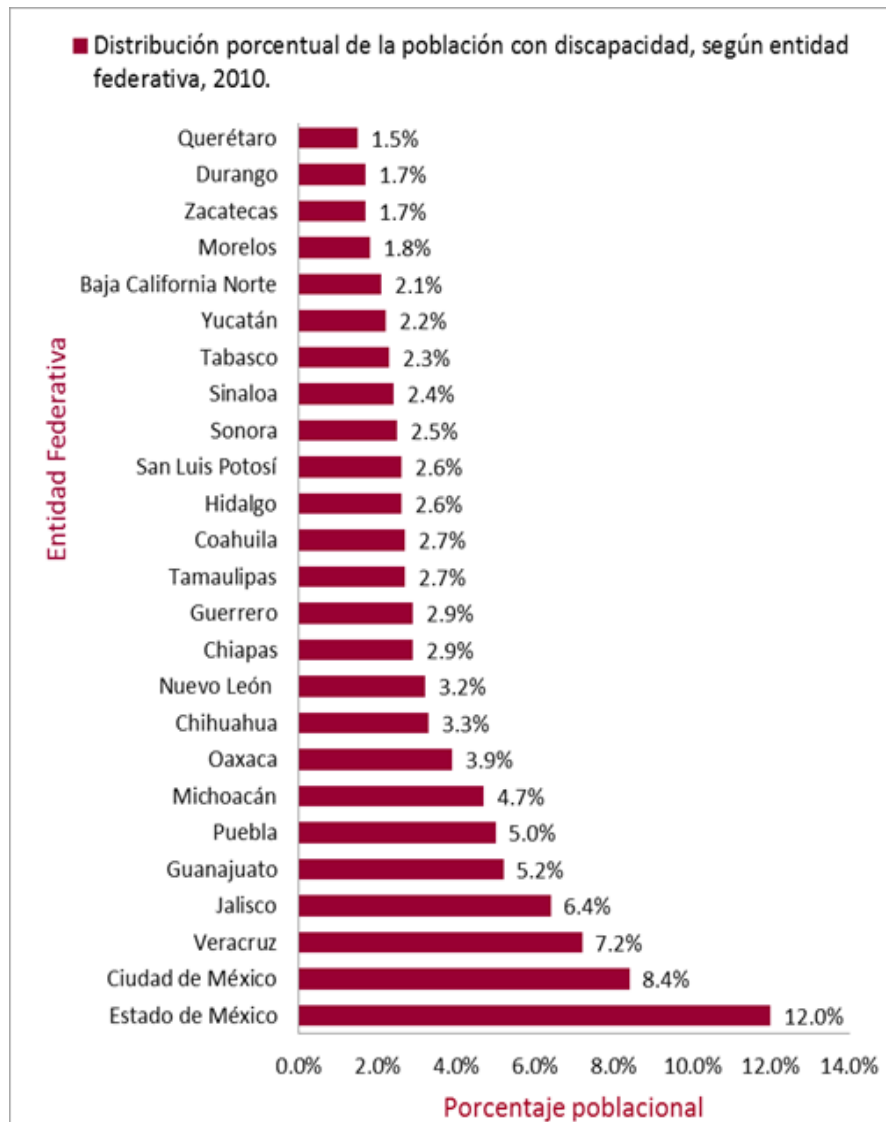


Figura 3. INEGI Censo de población y vivienda 2010 [7]

- *Tipos de discapacidad.*

El tipo de discapacidad más frecuente es según el Censo 2010, la relacionada con la movilidad (caminar o moverse), 58% señala presentarla². Le siguen las limitaciones para ver (27.2%), escuchar (12.1%), mental (8.5%), hablar o comunicarse (8.3%), atender el cuidado personal (vestirse, bañarse o comer) (5.5%) y, finalmente, para poner atención o aprender (4.4 %). (INEGI 2010) [7]

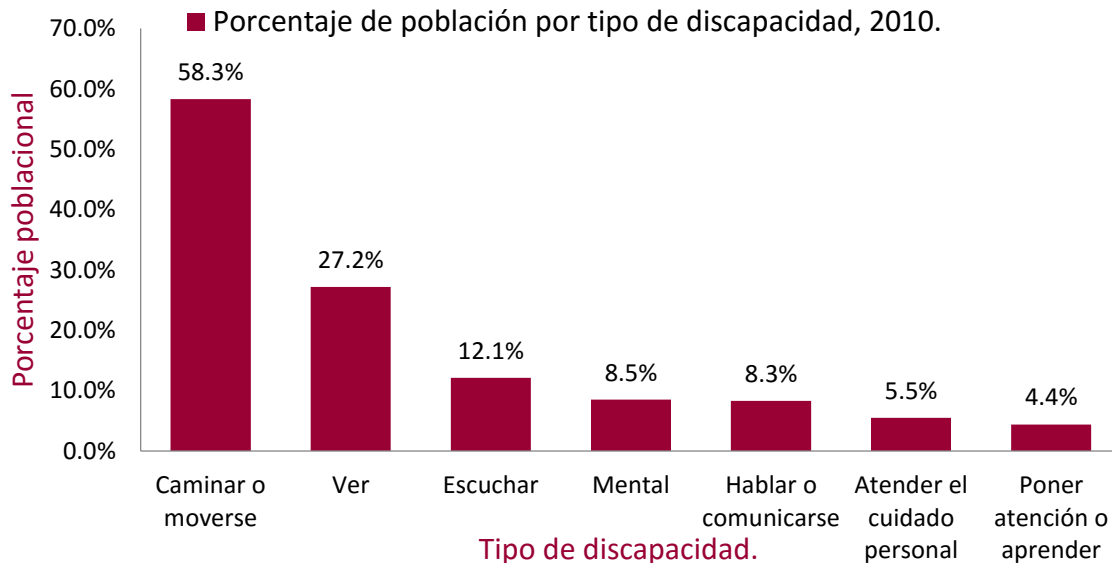


Figura 4. INEGI Censo de población y vivienda 2010 [7]

De acuerdo con los resultados de la Encuesta Nacional sobre Discriminación en México (ENADIS) 2010, 70 de cada 100 personas en el país consideran que no se respetan o sólo se respetan en parte los derechos de la población con discapacidad, mientras que 93.8% de las personas con discapacidad reportan igual opinión. [8]

Por otra parte, las principales áreas donde las personas con discapacidad identifican mayores problemas son: el desempleo y la discriminación, entre los ámbitos menos problemáticos se encuentran el respeto a sus derechos, carencia de lugares públicos e inseguridad. [8]

² Es importante señalar que debido a que la población podía reportar más de una dificultad, la cantidad de éstas es superior al número de personas con discapacidad.

- Distribución porcentual de población con discapacidad según respuesta a la pregunta ¿cuál cree que es el principal problema de las personas en su condición en México, hoy en día?, 2010

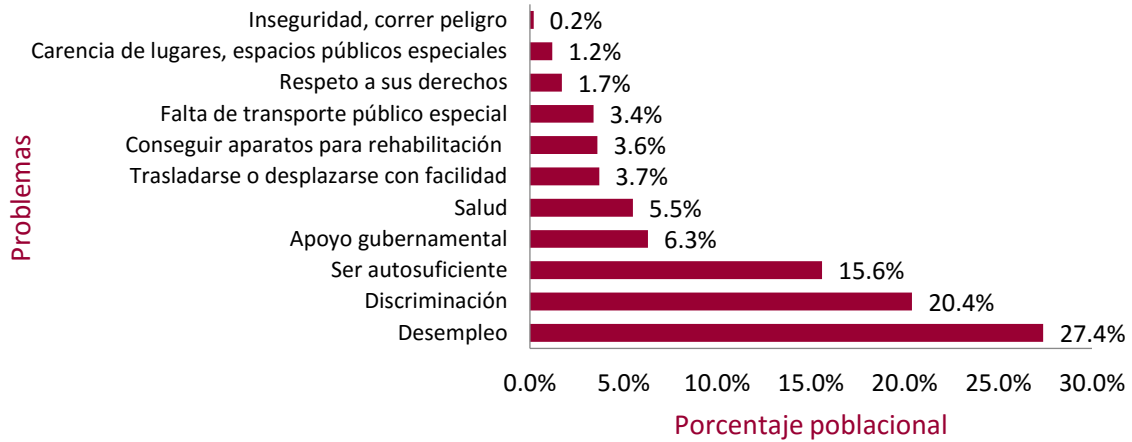


Figura 5. INEGI Censo de población y vivienda 2010 [7]

Como se observa en las estadísticas anteriores, podemos dejar en claro que uno de los principales objetivos es buscar soluciones a las necesidades de las personas con discapacidad, ya sea que las personas cuenten con una discapacidad permanente o temporal enfocándonos a aquellas personas con discapacidad motora y del habla, ya que es uno de los principales problemas a resolver hoy en día en la población es la necesidad de comunicación de las personas con discapacidad en la Ciudad de México.

2. Planteamiento del problema.

2.1. Descripción del Problema.

Un problema fundamental en nuestra sociedad, es la exclusión de las personas con discapacidad. Por medio del censo de población llevado a cabo por el INEGI en el 2010 (mencionado anteriormente), se muestra que en México habitan alrededor de 112 millones de personas y aproximadamente el 5.1% de esta población cuenta con alguna discapacidad, a la que se le limita o dificulta poder realizar actividades como: caminar, ver, escuchar, hablar, etc; las entidades federativas con mayor cantidad de población con discapacidad son: El estado de México con el 12% y el Distrito Federal, ahora llamada Ciudad de México con el 8.4%.

Los tipos de discapacidad más frecuente es según el Censo 2010 son: la movilidad (caminar o moverse), tener dificultad para poder hablar o comunicarse y tener limitaciones en la parte de atención de cuidado personal (vestirse, bañarse o comer).

En general el 15.6% de la población mexicana se considera no ser autosuficiente ya que enfrenta múltiples barreras que limitan su desarrollo personal; por lo que la población con discapacidad en México ya es considera una cifra significativa y de suma importancia, de esta forma, este trabajo se llevó a cabo para ayudar y cubrir las necesidades básicas de comunicación de un sector de la población con discapacidad, de acuerdo a las cifras mostradas anteriormente, se busca fomentar la autonomía de las personas que cuentan con alguna discapacidad motora o del habla, que no cuenten con problemas mentales o de aprendizaje.

Para ayudar a cubrir la necesidad de comunicación haremos uso de diferentes tecnologías, y así lograr un mejoramiento en la calidad de vida de las personas con discapacidad motora o del habla en México.

2.2. Objetivo General.

Crear un sistema híbrido haciendo uso de software y hardware, que permita satisfacer la necesidad de comunicación en la población con discapacidad motora y del habla hacia otros usuarios en México, para fomentar su independencia al comunicarse y mejorar su calidad de vida; por medio de la telefonía celular y otras tecnologías.

2.3. Objetivos Específicos.

- I. Investigar, analizar y conocer el funcionamiento básico de las interfaces cerebro-computador (BCI).
- II. Realizar una evaluación y selección las interfaces cerebro-computador (BCI) existentes en el mercado.
- III. Investigar, analizar y comparar las tecnologías de redes inalámbricas de área personal (WPAN) existentes en el mercado.
- IV. Investigar, analizar y comparar microcontroladores existentes en el mercado.
- V. Investigar y conocer el sistema de telefonía más usado en México; así como analizar y comparar las plataformas existentes en el mercado.
- VI. Investigar y conocer la pantalla Thin Film Transistor (TFT); así como analizar y comparar las existentes en el mercado.
- VII. Diseñar y desarrollar algoritmos y códigos de programación, para procesar y almacenar información en un sistema de cómputo.
- VIII. Analizar y elaborar un sistema de comunicación (BCI Mobile), haciendo uso de software e Implementar interfaces de diferentes tecnologías (hardware).
- IX. Elaborar un manual de usuario del sistema BCI Mobile, donde se explique cada componente y como se hace uso del mismo.
- X. Realizar pruebas del sistema BCI Mobile.

Capítulo II

3. Conceptos Teóricos.

3.1. Cerebro.

El cerebro es una masa de tejido gris-rosáceo compuesto por aproximadamente 100,000 millones de células nerviosas interconectadas, que son las responsables del control de todas las funciones vitales de supervivencia, funciones mentales, emociones y sentimientos a través de la recepción e interpretación de innumerables señales. El cerebro puede dividirse en dos partes: el diencefalo y el telencefalo, que forma los hemisferios cerebrales. [9] y [10]

3.1.1. Función.

El funcionamiento del cerebro se basa en el concepto de la neurona³ capaz de recibir información procedente de otras células nerviosas. La transmisión de señales se realiza mediante unas sustancias químicas conocidas como neurotransmisores.

El cerebro consta de dos hemisferios cerebrales que están conectados por una masa de sustancia blanca denominada cuerpo calloso: El hemisferio derecho e izquierdo. El hemisferio derecho esta especializado en la percepción de los sonidos no relacionados con el lenguaje (música, llanto, etc.), en la percepción táctil y en la localización espacial de los objetos. Por otra parte el hemisferio izquierdo está especializado en producir y comprender los sonidos del lenguaje, el control de los movimientos y los gestos con la mano derecha. [9] y [10]

Cada hemisferio se divide en cuatro grandes lóbulos: frontal, parietal, temporal y occipital. Como se muestran en la Figura 6.

- El lóbulo occipital: se recibe y analiza las informaciones visuales. [9]
- El lóbulo frontal: se encuentran las principales neuronas motoras. [11]
- El lóbulo parietal: se asocian con los sentidos del tacto. [9]
- Los lóbulos temporales: están situados justo por encima de los oídos; se encargan de la parte auditiva.[11]

³. Es una unidad anatómica, integrada por un cuerpo celular del que salen numerosas ramificaciones llamadas dendritas.

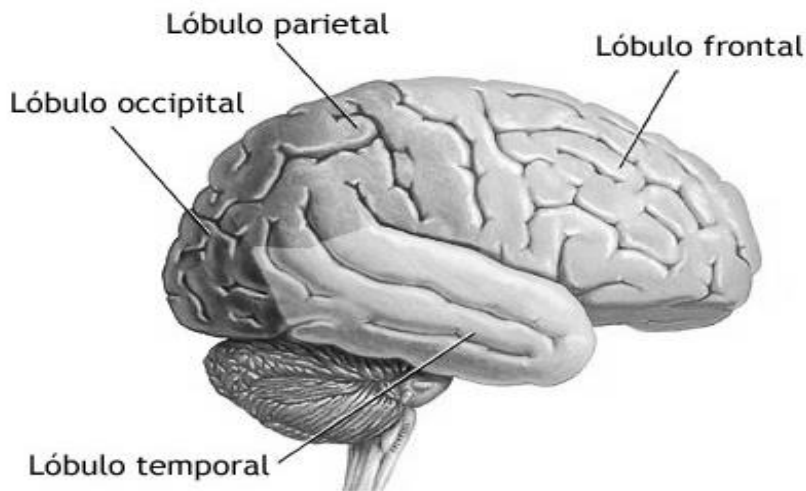


Figura 6. Lóbulos cerebrales. [11]

3.1.2. Características principales.

El cerebro es el órgano del cuerpo que más trabaja, ya que todo lo que hacemos, sentimos o pensamos se debe a él. El cerebro es multidireccional y sus funciones son mucho más complejas, ya que procesa la información analizándola, sintetizándola e integrándola a través de procesos paralelos simultáneos. Toda la información que recibe el cerebro la obtiene captando estímulos mediante los sentidos externos e internos, recibiendo y enviando constantemente señales que nos informan de todo lo que sucede a nuestro alrededor [9].

El cerebro contiene células nerviosas; las cuales son componentes fundamentales que se conectan entre si formando redes. Estas redes tienen una actividad eléctrica y química constante conocido como Sistema Nervioso. El sistema nervioso está constituido por el tejido nervioso del organismo y los elementos de soporte asociados. Desde un punto de vista estructural o anatómico, el sistema nervioso se divide en dos: el Sistema Nervioso Central (SNC) y el Sistema Nervioso Periférico (SNP). El SNC está formado por el cerebro y la médula espinal, y el SNP comprende los nervios, ganglios y receptores especializados. [12] y [9]

Las señales (o impulsos) que llegan al sistema nervioso periférico, se transmiten a partir de receptores al sistema nervioso central, donde la información es registrada y procesada convenientemente. Una vez registradas y procesadas, las señales son enviadas desde el sistema nervioso central a los distintos órganos a fin de proporcionar las respuestas adecuadas. [9]

3.2. Señales cerebrales.

Nuestro cerebro es un generador de procesos electroquímicos, en el cuál se emiten impulsos eléctricos a diferentes frecuencias y amplitudes. La frecuencia es una medida de rapidez de la onda, y se mide en Hertz (Hz). Cada una de estas frecuencias tiene distintos efectos el estado mental, la conducta y las emociones. La amplitud es medida en Voltios. Para poder describir las señales cerebrales se hace uso de un aparato externo llamado electroencefalograma (EEG), éste realiza el registro y la evaluación de los potenciales eléctricos generados por el cerebro y obtenidos por medio de electrodos situados sobre la superficie del cuero cabelludo. [13], [14] y [11]

3.2.1. Señales electroencefalográficas.

Las ondas cerebrales recogidas por el electroencefalograma (EEG) son llamadas señales electroencefalográficas y son procedentes del córtex cerebral o corteza cerebral por lo que son el reflejo de la actividad neuroquímica en estructuras más profundas. [13]

3.2.1.1. Clasificación.

Las ondas registradas en el electroencefalograma han sido clasificadas con base en su frecuencia, las principales ondas de frecuencia son las siguientes: [15]

- Ondas Beta: se registran en frecuencias de 14 y 30 Hz, con una amplitud de 25 μ V, se presentan en plena actividad mental y con la apertura de ojos de vigilia. Se encuentra principalmente en las regiones centrales y frontales. [11], [14] y [15]
- Ondas Alfa: están asociadas con el estado de relajación y pensamientos tranquilos. Son ondas de mayor amplitud que la beta y se presentan en una frecuencia de 8 a 13 Hz, con una amplitud de 50 μ V. Se presenta en la vigilia principalmente en áreas posteriores, aunque puede verse también en áreas anteriores y centrales. Es el tipo de banda que predomina en el sujeto adulto en reposo y con los ojos cerrados, generalmente “desaparece” al abrir los ojos. [16], [14] y [15]
- Ondas Theta: son ondas de mayor amplitud y menor frecuencia, por lo que se presentan en una frecuencia de 4 a 7 Hz y usualmente tienen una amplitud de 30 a 60 μ V junto con delta son llamadas ondas lentas. Se Presentan durante la infancia, así como en estados de adormecimiento y sueño, también se ha asociado con la ejecución de procesos cognitivos. Estas ondas aparecen en las regiones temporales. [11], [14], [15] y [16]

- Ondas Delta: estas ondas están caracterizadas por frecuencia menos de 4Hz y alta amplitud de 75 μ V; son ondas presentes durante los primeros años de vida en el ser humano, y va desapareciendo conforme madura el sistema nervioso. En el adulto se encuentra principalmente durante las fases 3 y 4 del sueño de ondas lentas. Sin embargo, se presenta también cuando se pone atención a procesos internos como son la realización de cálculos mentales o la memorización. Se considera que la actividad delta se genera prácticamente en su totalidad en la corteza, especialmente entre capas profundas. [17], [15] y [14]

Nuestra habilidad para atender, planificar, contactar o hacer cualquier cosa está íntimamente relacionada con la combinación de las ondas cerebrales. Es importante recordar que el cerebro siempre está produciendo múltiples ondas cerebrales al mismo tiempo en el transcurso del día, como por ejemplo en la realización de un parpadeo, el cual está relacionado con las ondas Beta, Alfa, Theta y Delta, teniendo mayor presencia con las dos primeras ondas, debido a que se encuentran principalmente en regiones frontales, mencionado anteriormente.

A continuación en la figura 7 se muestran las ondas típicas de las señales EEG respectivamente:

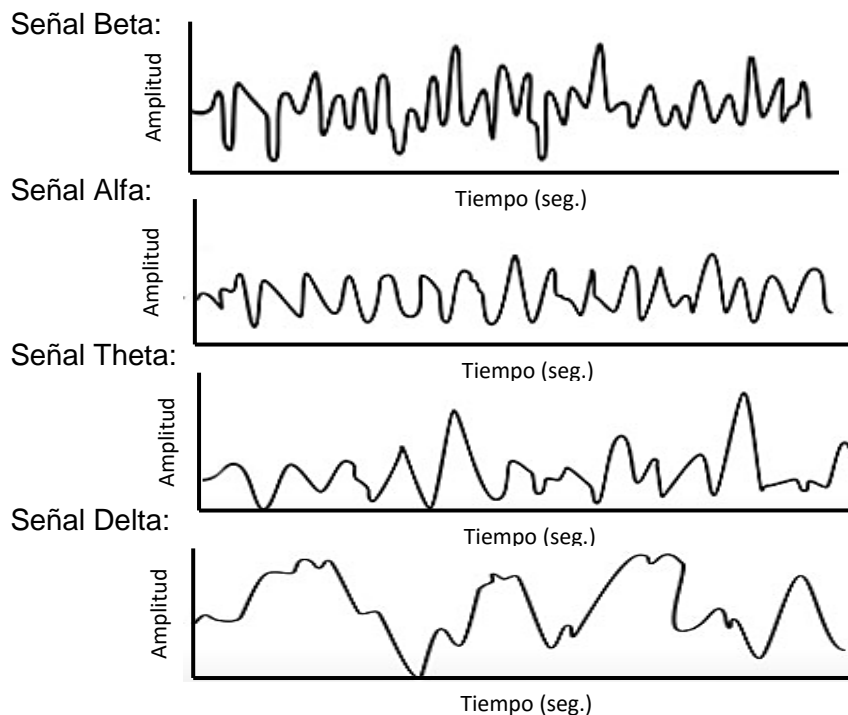


Figura 7. Ondas cerebrales típicas. [11]

3.2.1.2. Electroodos.

Para realizar una correcta lectura de las señales electroencefalográficas se hace uso de diversos dispositivos llamados electroodos, estos se clasifican en superficiales y quirúrgicos, a continuación se muestra la clasificación de electroodos para la lectura de señales electroencefalográficas en la Tabla 1.

Clasificación de electroodos		
Tipo de electroodo		Características principales
superficiales	Adheridos	Pequeños discos metálicos de 5 mm de diámetro. Se adhieren con pasta conductora y se fijan con colodión que es aislante.
	De contacto	Pequeños tubos de plata clorurada roscados a soportes de plástico. En su extremo de contacto se colocan una almohadilla que se humedece con solución conductora. Se sujetan al cráneo con bandas elásticas y se conectan con pinzas de «cocodrilo».
	En casco de malla	Los electroodos están incluidos en una especie de casco elástico. Se sujetan con cintas a una banda torácica. Cuentan con gran inmunidad a los artefactos ⁴ tales como: señales electromiograma (EMG), señales electrooculograma (EOG) y señales electrocardiograma (ECG).
	De aguja	Se emplean exclusivamente en recién nacidos. Pueden ser desechables (de un solo uso) o de uso múltiple. En este caso, su esterilización y manipulación deben ser muy cuidadosos
Quirúrgicos	Durales	Se utilizan durante el acto quirúrgico y son manipulados exclusivamente por el neurocirujano.
	Corticales	
	Intercerebrales	

Tabla 1. Clasificación de electroodos. [11]

3.2.1.3. Sistemas de colocación de electroodos.

La colocación de electroodos sobre el cuero cabelludo está sujeta a un sistema internacional 10-20 (American Electroencephographic Society, 1986) (Figura 8), denominado así porque los electroodos están espaciados entre el 10% y el 20% de la distancia total entre puntos reconocibles del cráneo. En total se colocan 21 electroodos, 19 en el cráneo y 2 en las orejas, que se conoce como puntos de referencia A1 y A2. [14] y [18].

El cráneo se divide en 4 puntos clave a partir de los cuales se realiza la colocación de electroodos para la lectura de las señales EEG:

- Nasión: indentación entre frente y nariz.
- Inión: occipital.
- Punto preauricular: delante de cada pabellón de la oreja.

⁴ Los artefactos son señales no deseadas que contaminan la actividad cerebral y su origen no es cerebral, pueden clasificarse en dos categorías: artefactos fisiológicos y artefactos no fisiológicos o técnicos. Los artefactos fisiológicos son por lo general causados por el movimiento de músculos, ojos y pulsaciones del corazón, Los técnicos se atribuyen principalmente al ruido de la línea eléctrica de los electroodos.

- Vertex: parte central del cráneo. [18]

Los electrodos se denominan con diferentes letras según su ubicación. [18]

- F: Frontal.
- FP: Frontal-polar.
- T: Temporal.
- P: Parental.
- C: Central.
- O: Occipital.
- A: Auricular. [14]

Además, se asignan números pares para el hemisferio izquierdo, números noes para el hemisferio derecho y se asigna “z” para los electrodos de la línea media [14].

Cabe enfatizar que los electrodos para detección de los parpadeos se ubican en las posiciones FP1 y FP2 del sistema 10-20.

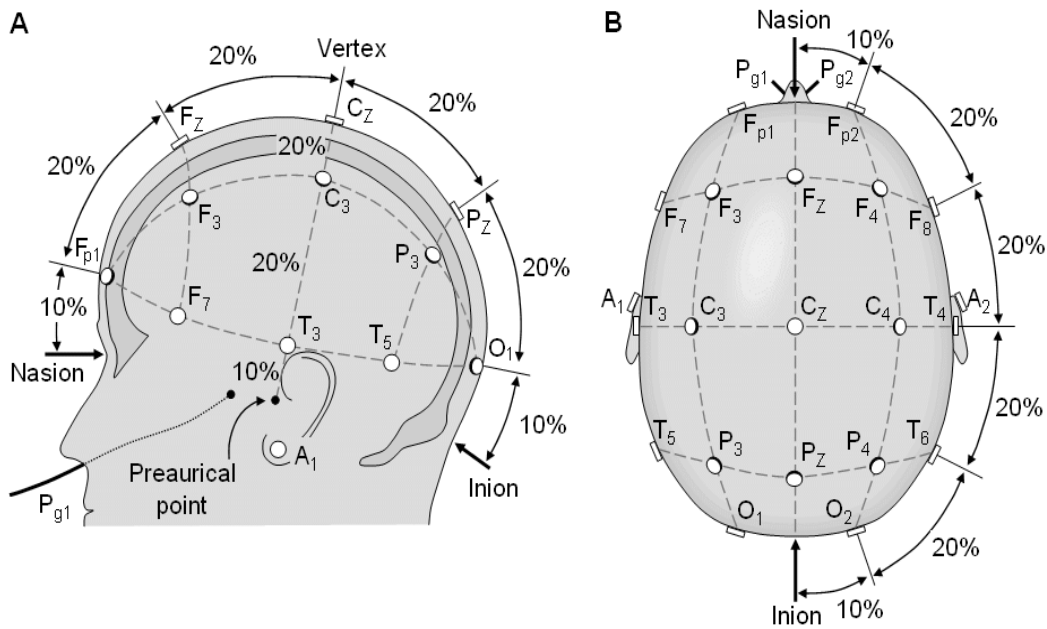


Figura 8. Sistema Internacional 10-20 para la colocación de electrodos extracraneales. [14]

3.3. Interfaz cerebro-computador.

3.3.1. Función.

Una interfaz cerebro computador (BCI-Brain Computer Interface) es un dispositivo, el cual crea un enlace de comunicación directa entre el cerebro y algún dispositivo externo como por ejemplo un ordenador. Estos dispositivos, detectan y analizan las ondas emitidas por el cerebro humano, estas ondas son procesadas y convertidas en órdenes o acciones determinadas sobre algún dispositivo. Todo esto sin la necesidad de ningún tipo de movimiento muscular por parte del usuario, únicamente haciendo uso de la habilidad mental del mismo. [19]

3.3.2. Historia.

La historia de la interfaz cerebro computador, surgió cuando el pionero Hans Berger un neurocientífico alemán, descubre la actividad eléctrica de las ondas alfa de un cerebro humano con EEG (electroencefalograma) en 1924. En el año de 1970, comenzaron a surgir diferentes programas de investigación de las BCI's, motivados por la observación científica de las señales de tipo EEG, en ese mismo año la Defensa de Proyectos de Investigación Avanzados de EE.UU inició el programa para explorar las comunicaciones del cerebro usando el electroencefalograma (EEG). [20] y [21]

Jacques Vidal, investigador de ciencias informáticas en la UCLA (Brain Research Institute University of California), trabajó en el desarrollo de la primera interfaz directa cerebro-ordenador (BCI), utilizando una computadora IBM (International Business Machines) y uso el termino BCI en 1976. [20] y [21]

En 2003 se expone al público la primera BCI llamada BrainGate, éste es un sistema de implante en el cerebro, fue desarrollado por la empresa de biotecnología Cyberkinetics en conjunto con el Departamento de Neurociencias de la Universidad de Brown. El 26 de junio del 2011 en la Universidad de Tecnología de Sydney (UTS) se llevó a cabo el control mental de una silla de ruedas (BCI no invasivo). [20]

Hoy en día las BCI's se están convirtiendo en el campo de investigación más importante en el mundo para dar solución en personas con discapacidad motora. [20]

3.3.3. Clasificación.

Una interfaz cerebro computador (BCI-Brain Computer Interface), es un canal de comunicaciones directo entre el cerebro y un dispositivo externo que transforma directamente los procesos cerebrales en acciones. [19]

Las BCI pueden clasificarse en tres tipos distintos:

3.3.3.1. BCI invasiva.

Para hacer uso de una BCI invasiva se requiere una intervención quirúrgica para implantar electrodos en la corteza cerebral (Electrocortigrama, ECoG) o directamente en el cerebro mediante neurocirugía. [19] y [22]

3.3.3.2. BCI parcialmente invasiva.

En la BCI parcialmente invasiva los electrodos son implantadas en el interior del cráneo pero fuera de la materia gris. [20]

3.3.3.3. BCI no invasiva.

En la BCI no invasiva no se realiza ningún tipo de intervención quirúrgica en el usuario, en este caso es posible recoger la actividad cerebral utilizando diversos métodos: la encefalografía (EEG), la magneto encefalografía (MEG), la tomografía de emisión de positrones (PET), la imagen de resonancia magnética (fMRI) y la imagen óptica cercana al infrarrojo (fNIR). La principal desventaja de la BCI no invasiva es que la calidad y la fiabilidad de la señal no es tan alta como en los BCI invasivos y BCI parcialmente invasivos. Aun así es posible detectar las ondas, lo que resulta más complejo es establecer la zona del cerebro de la que proviene. [20] y [22]

3.3.4. BCI en el mercado.

Anteriormente se menciona, que para hacer uso de la BCI invasiva se requieren conocimientos de neurocirugía así como hacer uso de recursos especiales (quirófano, instrumentación y personal), para llevar a cabo la intervención quirúrgica e implantar los electrodos en la corteza cerebral o directamente en el cerebro; por lo tanto, como no se cuenta con los conocimientos, ni con los recursos necesarios para llevar a cabo la colocación de los electrodos de manera invasiva o en general para hacer uso de la BCI invasiva, esta será descartada y solo se hará referencia al uso de BCI no invasiva.

A continuación en la tabla 2 se muestran las interfaces cerebro computador comerciales no invasivas en el mercado.

Interfaz Cerebro Computador (No invasiva)								
Fabricante	Modelo	Número de sensores	Tipo de sensores	Dispositivos compatible	Herramienta de desarrollo	Tipo de Batería	Tipo de Conexión	Referencia
Emotiv	EPOC/EPOC+	14	Sensores salinos.	Microsoft Windows, Android, Linux y Apple iOS	SDK, c, c#, C++, java, Python.	Batería recargable, 12Hr de uso continuo	Bluetooth	[23]
Emotiv	Insight	5	Biosensor polímero hidrófilo	Microsoft Windows, Android, Linux, Apple iOS	SDK, c, c#, C++, java, Python	Batería recargable, 4Hr de uso continuo	Bluetooth	[24]
Neurosky	MindWave mobile	1	Sensor giroscópico.	Apple iOS y Android	Arduino, SDK, iPhone, Raspberry Pi y .NET.	Batería AAA con 8 Hr de duración	Bluetooth	[25]
InteraXon	Muse	7	Señor muse	Apple iOS y Android	SDK básico para desarrollo	5 Hr de duración	Bluetooth	[26]
OpenBCI	Kit OpenBCI	8	Biosensores	Arduino	Arduino	Batería AA de 6V.	Bluetooth	[27]
iFocus band	Focus Band	1	----	Android	SDK	-----	Bluetooth	[28]

Tabla 2. Interfaces Cerebro Computador en el mercado.

3.4. Redes Inalámbricas de Área Personal (WPAN).

Las redes inalámbricas de datos, son redes de comunicaciones no guiadas que interconectan varios dispositivos y proporciona un medio no guiado (a través de ondas electromagnéticas) para el intercambio de información entre ellos. La aparición de las redes inalámbricas de área personal (WPAN) trae consigo el concepto POS-Personal Operating Space (Espacio de operación personal), este concepto constituye un área de reducida cobertura alrededor de un individuo y que le otorga el potencial de conectar su dispositivo personal a otros dispositivos, es decir, si el espacio se vincula a un individuo entonces se mueve cuando este se desplaza. [29] y [30]

Las redes inalámbricas de área personal (WPAN) permiten la conexión a través de enlaces inalámbricos de rango reducido (desde rangos inferiores al metro hasta alrededor de 100 metros), las redes inalámbricas proporcionan diversas ventajas: [30]

- Rápida instalación; no se necesitan medios de comunicación guiados.
- Movilidad; permite una movilidad dentro del radio de recepción de la señal.
- Accesibilidad; la mayoría de los dispositivos portátiles soportan o incluyen varias tecnologías inalámbricas.
- Se proporciona como solución para zonas a las que no llega la estructura física. [29]

Las redes inalámbricas proporcionan ciertas desventajas:

- Interferencias externas; de otros emisores.
- Errores; debido a las interferencias.
- Mayores costos; los dispositivos son más caros.
- Velocidad limitada; WPAN ofrece solo velocidades de 1 Mbps, 2Mbps o hasta 3Mbps a comparación a otras redes inalámbricas. [29]

3.5. Estándares de comunicación inalámbrica (IEEE 802.15).

Un estándar es un conjunto de normas que se establecen para garantizar la compatibilidad entre elementos independientes, por lo que el establecimiento de los estándares de comunicación garantiza la comunicación entre computadores de diferentes fabricantes, dando como resultado la creación del instituto de Ingenieros Eléctricos y Electrónicos (Institute of Electrical and Electronics Engineers- IEEE) fundado en 1884; se encarga de realizar diversas tareas, entre las cuales se encuentra la regularización de diferentes tecnologías mediante la creación de normas o estándares. Para ello el comité 802 de IEEE se ha organizado en diferentes grupos de trabajo, cada uno de los grupos recibe el nombre IEEE 802.xx. IEEE está dividido en diferentes comités de estandarización que abordan diversas áreas de conocimiento. En febrero de 1980 se constituyó el comité 802, encargado de definir estándares para redes de área local y redes de área metropolitana (LAN, Local Area Network y MAN, Metropolitan Area Network). [31]

El grupo de trabajo IEEE 802.15 es parte del Comité de Normas de la red local y Área Metropolitana 802 del IEEE Computer Society; se encarga de la responsabilidad de establecer las normas relativas a la red de área personal inalámbrica; IEEE 802.15 se centra en el desarrollo de normas de consenso para redes de área personal o redes inalámbricas de corta distancia. Estos WPAN abordan las redes inalámbricas de los dispositivos informáticos portátiles y móviles, como ordenadores, asistentes digitales personales (PDA), periféricos y teléfonos celulares; permitiendo que estos dispositivos se comuniquen e interoperen con otros. El objetivo es publicar normas, prácticas recomendadas, o guías que tienen amplia aplicabilidad de mercado y tratan eficazmente con los problemas de convivencia y la interoperabilidad con otras soluciones de redes cableadas e inalámbricas. [32]

3.6. Estándar de comunicación inalámbrica IEEE 802.15.1 – Bluetooth.

3.6.1. Historia.

En 1994 un grupo de ingenieros de Ericsson, una empresa sueca, inventó una tecnología de comunicación inalámbrica, más tarde llamado Bluetooth. En 1998, el grupo original de las empresas promotoras-Ericsson, Intel, Nokia, Toshiba e IBM se unieron para formar el Bluetooth Special Interest Group (SIG). Los miembros SIG trabajan juntos para conservar, educar y fomentar la tecnología Bluetooth. [33]

3.6.2. Definición.

Bluetooth es una tecnología estandarizada por la IEEE 802.15, define un estándar inalámbrico global de comunicaciones inalámbricas de corto alcance que permite conectividad segura para una creciente gama de dispositivos y servicios. Las características principales de la tecnología Bluetooth son ubicuidad⁵, de baja potencia y bajo costo. La especificación Bluetooth define una estructura uniforme para una amplia gama de dispositivos para conectar y comunicarse entre sí, por lo que le permite compartir voz, datos, música, fotos, videos entre otra información de forma inalámbrica entre dispositivos vinculados. [33]

3.6.3. Composición.

El núcleo del sistema Bluetooth consta de un transceptor de radio frecuencia (RF), de banda base, y la pila de protocolos. El sistema ofrece servicios que permiten la conexión inalámbrica de dispositivos y el intercambio de una gran variedad de clases de datos. Muchas de las funciones de la especificación principal son opcionales, lo que permite implementaciones de productos flexibles. [33]

⁵ Ubicuidad- Presencia en todas partes a la vez

3.6.4. Especificaciones básicas de Bluetooth.

Sus características principales del estándar Bluetooth son descritas en la tabla 3.

Características principales de Bluetooth	
Velocidad de transmisión RF.	<ul style="list-style-type: none"> • 1Mbps. • Si se utiliza EDR (Enhanced Data Rate)- de 2 o 3 Mbps.
Banda de frecuencia.	<ul style="list-style-type: none"> • 2.4 a 2.485 GHz en la banda ISM (Unlicensed industrial, scientific and medical).
Técnica de modulación.	<ul style="list-style-type: none"> • Espectro ensanchado por salto de frecuencia (FHSS); utilizan 79 canales de radio con un ancho de banda de 1 MHz.
Potencia de transmisión.	<ul style="list-style-type: none"> • Clase1: 100mW. • Clase2: 2.5mW. • Clase3: 1mW.
Rango de operación.	<ul style="list-style-type: none"> • 1 a 10 metros ó hasta 100 metros para la clase1.
Topología.	<ul style="list-style-type: none"> • Piconet y Scatternet.

Tabla 3. Características principales del estándar Bluetooth. [34] y [35]

3.6.5. Pila de protocolos

En la Figura 9 se muestra la pila de protocolos que define al estándar 802.15.1 – Bluetooth:

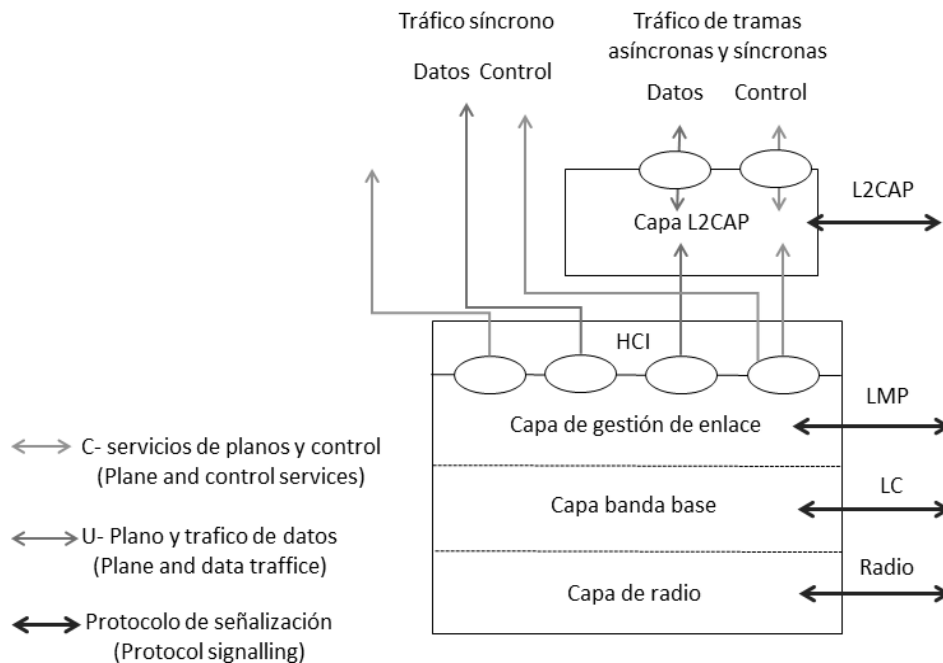


Figura 9. Pila de protocolos de Bluetooth. [33]

La pila de protocolos de Bluetooth involucra el **controlador Bluetooth** y el **anfitrión de Bluetooth**. El **controlador de Bluetooth** es un subsistema que consiste en la agrupación de las 3 capas inferiores como son: [33]

- Capa de radio (Radio Layer): Está formado por el transceptor físico y sus componentes asociados; como utiliza la banda ISM de 2.4 GHz, esto facilita la calidad en la señal y la compatibilidad entre transceptores. La capa de radio es la responsable de transmitir y recibir paquetes de información sobre el canal físico. [33]
- Capa Banda Base (Baseband Layer): Se encarga de controlar las operaciones sobre bits y paquetes, realiza detección y corrección de errores, multidifusión y cifrado, también emite confirmaciones y peticiones de retransmisión. [33]
- Capa de gestión de enlace (Link Manager Layer): Se encarga del establecimiento y finalización de conexiones, es responsable de la creación, modificación y publicación de enlaces lógicos, así como la actualización de los parámetros relacionados con los enlaces físicos entre los dispositivos. El gestor de enlaces logra esto mediante la comunicación con el gestor de enlaces en los dispositivos Bluetooth remotos utilizando: [33]

El **anfitrión de Bluetooth** involucra la capa de adaptación de enlace lógico (L2CAP). [33]

- La capa L2CAP se encarga de controlar la comunicación proveniente de niveles superiores y la asocia a los sistemas de transporte de datos, multiplexando los canales L2CAP en enlaces lógicos y segmentando las tramas adecuadamente. L2CAP puede añadir opcionalmente detección de errores y retransmisión de paquetes a Banda base, así como control de flujo basado en protocolos de ventana deslizante, asignación de buffers y QoS (Quality of Service). [33]

Como se mencionó anteriormente el controlador de Bluetooth opera las tres capas más bajas y la capa L2CAP está contenido con el resto de la aplicación Bluetooth en un sistema huésped. La interfaz estándar se llama el host de interfaz de controlador (HCI). La implementación de esta interfaz de servicio estándar es opcional. [33]

La especificación Bluetooth permite la interoperabilidad entre los sistemas Bluetooth independientes mediante el intercambio de los mensajes de protocolo entre capas equivalentes, así como la interoperabilidad entre subsistemas Bluetooth independientes

mediante la definición de una interfaz común entre los controladores Bluetooth y anfitriones Bluetooth. [33]

El núcleo del sistema Bluetooth ofrece servicios a través de una serie de puntos de acceso de servicio que se muestra en la Figura 9 como elipses. Los servicios se pueden dividir en tres tipos: Servicios de dispositivos de control: modifican el comportamiento y los modos de un dispositivo Bluetooth, Servicios de control de transporte: crea, modifica y libera portadores de tráfico (canales y enlaces) y Servicios de datos: se utilizan para presentar los datos para su transmisión por portadores de tráfico [33].

3.6.6. Perfiles

Para hacer uso de la tecnología inalámbrica Bluetooth, los dispositivos deben ser capaces de interpretar ciertos perfiles Bluetooth. Los perfiles Bluetooth son las definiciones de las posibles aplicaciones, especifica comportamientos generales que los dispositivos Bluetooth utilizan para comunicarse con otros dispositivos Bluetooth [33].

Se hará mención de los perfiles que se utilizan generalmente en la BCI's con tecnología Bluetooth:

- SSP- Serial Port Profile: define cómo configurar los puertos seriales virtuales y conectar dos dispositivos Bluetooth habilitados. [36].
- A2DP - Advanced Audio Distribution Profile: describe cómo el audio de calidad estéreo se puede transmitir de un medio de comunicación a unos auriculares inalámbricos o altavoces estéreo inalámbricos [37].
- AVRCP - Audio / Video Remote Control Profile: proporciona una interfaz estándar para permitir que un solo control remoto (u otro dispositivo) controle de todo el equipo de audio y video a la que tiene acceso un usuario [38].
- HSP - Headset Profile: describe cómo un auricular Bluetooth activado debe comunicarse con un ordenador u otro dispositivo Bluetooth compatible, como un teléfono móvil. [39].
- HFP - Hands-Free Profile: describe cómo un dispositivo de pasarela se puede utilizar para realizar y recibir llamadas hacia un dispositivo de manos libres [40].

3.6.7. Bluetooth en el mercado.

Hoy en día se encuentran diversas marcas de interfaces Bluetooth, sin embargo solo se hará mención de tres marcas reconocidas en el mercado, como se muestra en la Tabla 4.

Interfaces Bluetooth								
Fabricante	Modelo	Velocidad	Alcance	Clase	Voltaje de funcionamiento	Antena	Configuración	Referencia
Botscience Worldwide Imports	HC 06	1200 a 115200 bps.	10 m.	2	3.3V-6V	Incluida	Comandos AT	[41]
Botscience Worldwide Imports	HC 05	1200bps a 1.3Mbps	10 m.	2.0 +EDR	3.3 V.	Incluida	Comandos AT	[42]
SparkFun	Bluesmirf Mate Silver	2400 a 115200 bps.	10 m.	2	3.3V-6V	Incluida	Comandos AT	[43]
SparkFun	Bluesmirf Gold	2400 a 115200 bps	100 m.	1	3.3V-6V	Incluida	Comandos AT	[44]
Adafruit	Bluefruit EZ Link	2400 a 230400 bps	100 m	1	3V	incluida	Comandos AT	[45]

Tabla 4. Interfaces Bluetooth en el mercado.

3.7. Estándar de comunicación inalámbrica IEEE 802.15.4 – ZigBee.

3.7.1. Historia.

La especificación Zigbee comienza a evolucionar en el año 1998 y definen a Zigbee Alliance. En mayo del 2003 se decide basar su implementación en el estándar IEEE 802.15.4, es así que en Diciembre del 2004 se publica Zigbee 1.0. El tiempo sigue transcurriendo y en el año 2007 se publica la especificación que introducen dos perfiles de operación: Zigbee-PRO y zigbee. [46]

3.7.2. Definición.

ZigBee es la tecnología inalámbrica basada en estándar 802.15.4, diseñado para hacer frente a las necesidades únicas de comunicación de dispositivo a dispositivo en casi cualquier mercado, por consiguiente es una tecnología inalámbrica de corto alcance y bajo consumo de energía para aplicaciones en el hogar como seguridad, automatización, entre otros, centrada en la creación de redes de alta velocidad. Las características básicas con las que cuenta ZigBee son las siguientes: [47]

- Fiable; utiliza redes de malla multi-hop para eliminar puntos únicos de fallo y ampliar el alcance de las redes.
 - Es de bajo consumo; permite a los dispositivos durar por años con una sola batería.
- [48]

3.7.3. Composición.

Una red ZigBee puede estar formada hasta por 255 nodos los cuales contienen un “tranceiver ZigBee” con el objeto de consumir menos que otras tecnologías inalámbricas, ya que puede ser alimentado con dos pilas AA de 6 meses a 2 años, dependiendo del consumo. [47]

3.7.4. Especificaciones básicas de ZigBee.

Sus características principales del estándar Zigbee son descritas en la tabla 5.

Características principales de Zigbee	
Velocidad de transmisión RF.	20 kbps y 250 kbps.
Banda de frecuencia.	2.4 GHz en la banda ISM (Unlicensed industrial, scientific and medical).
Técnica de modulación.	Espectro expandido por secuencia directa (DSSS).
Rango de operación.	10m a 75m.
Topología.	Estrella, árbol y de malla.

Tabla 5. Características principales del estándar Zigbee. [49] y [50]

3.7.5. Pila de protocolos.

La pila de protocolos ZigBee consta de varios componentes en capas como el estándar IEEE 802.15.4 (la capa de Control de Acceso al Medio -MAC y la capa física - PHY) y la capa de red Zigbee –NWK. Como se muestra en la Figura 10. [49]

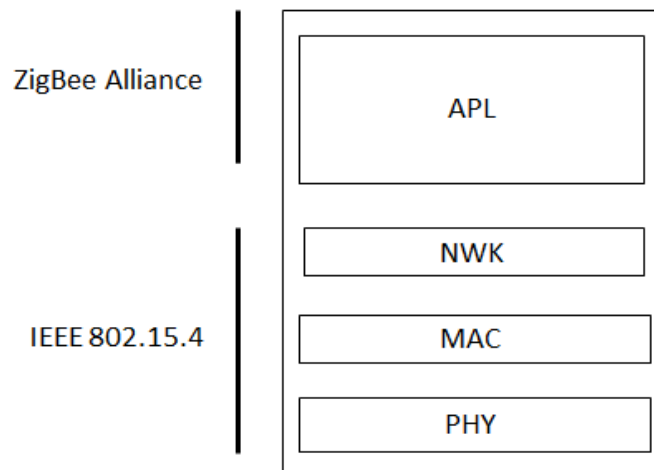


Figura 10. Pila de protocolo de ZigBee. [49]

- La capa física se encarga de transmitir y recibir tramas de bits sobre el medio físico. [49]
- La capa de control de acceso al medio (MAC) definida en el estándar IEEE 802.15.4, su finalidad es controlar y gestionar el acceso al medio, para lo que utiliza un protocolo CSMA/CA (Carrier Sense Multiple Access / Collision Avoidance); así como sincronizar tramas y controlar el flujo. También es el encargado de reducir al máximo el gasto energético del dispositivo. [49]
- La capa de nivel de red (NWK) se encarga de la topología de la red, añadiendo o eliminando dispositivos de la misma; asignando direcciones de red a los dispositivos;

y re-direccionando las tramas de información hacia el destinatario por el camino más adecuado, también se encarga de garantizar la fiabilidad y calidad de los datos recibidos en el nodo, mediante el control y la corrección de los errores. [49]

- La capa del nivel de aplicación (APL) es la responsable de ejecutar las aplicaciones del sistema [49].

3.7.6. Perfiles.

En la capa aplicación hay perfiles que se diseñaron para unificar el intercambio de datos en esta capa. Un perfil caracteriza tipos de dispositivos, funciones y formato de los mensajes y acciones que se usarán en ciertas aplicaciones. Los perfiles pueden ser:

- Perfiles públicos: los especifican la Alianza ZigBee para proveer algún tipo de interoperabilidad entre dispositivos de distintos fabricantes. [50]
- Perfiles privados: los especifica un fabricante o un usuario para sus aplicaciones específicas que no pueden realizarse con un perfil público. [50]

3.7.7. ZigBee en el mercado.

- Hoy en día se encuentran una sola marca de interfaces ZigBee reconocida en el mercado mexicano, como se muestra a continuación (Ver Tabla 6).

Interfaces ZigBee							
Fabricante	Modelo	Velocidad	Alcance	Voltaje de funcionamiento	Antena	Configuración	Referencia
Digi International	XBee	250kbps	120 m.	3.3V	Incluida	Comandos	[51]
	ZigBee Mesh					AT	
Digi International	XBee Pro	250kbps	1500 m.	3.3 V.	Incluida	Comandos AT	[52]

Tabla 6. Interfaces ZigBee en el mercado.

3.8. GSM- Sistema Global para Comunicaciones Móviles.

3.8.1. Historia.

En el inicio de los 80's se desarrollaron los primeros sistemas de comunicaciones celulares móviles en el mundo. Cada país desarrollo su propio sistema celular, debido a la incompatibilidad de estos sistemas hacia otros países, surgió en 1982 el estándar de telefonía celular GSM (Special Mobile Group) creado por la Conferencia de Correos y Telégrafos Europeos (CEPT), el cual formo un grupo de estudio para desarrollar un sistema telefónico móvil terrestre y público. Posteriormente rebautiza las siglas de Global System For Mobile Telecommunications, con el propósito de desarrollar especificación técnica para una red Europea de Telecomunicaciones móvil capaz de soportar los millones de futuros clientes de este servicio. Los servicios móviles basados en la tecnología GSM se pusieron en marcha por primera vez en 1991, se convirtió en un estándar internacional llamado "Global System for Mobile Communication". [53]

3.8.2. Definición.

El estándar GSM es el estándar utilizado en las tecnologías móviles 2G y consiste en una red digital de circuitos conmutados, esto es, una transmisión de datos desde su origen hasta su destino a través de una red de nodos, que proporciona principalmente servicios de llamadas de voz y servicios de mensajes cortos (SMS). [54]

3.8.3. Características

Una de sus principales características de la red GSM es la movilidad que proporciona al usuario, gracias al uso de una tarjeta inteligente denominada módulo de identidad del abonado, SIM (Subscriber Identity Module). Esta tarjeta permite acceder a los servicios que se tengan contratados con la red GSM de cualquier terminal, por lo que permite transmitir datos incluso si los usuarios se encuentran en distintas coberturas o en el dominio de diferentes operadores. [55]

Otra característica con la que cuenta GSM, es la capacidad de transmitir datos y voz a una velocidad de transmisión de 9.6Kbits/s. Esto le ha permitido ofrecer el servicio de mensajes cortos (SMS) de forma bidireccional, es decir, enviar y recibir mensajes alfanuméricos de hasta 140 caracteres. [54] y [56]

3.8.4. Servicios.

La red GSM presta servicios de conexión a la red telefónica pública mediante una red inalámbrica, en la cual los usuarios tienen la posibilidad de originar y recibir llamadas telefónicas, así mismo, permite el intercambio de mensajes de texto; Los servicios de la red GSM se pueden clasificar en servicios de portador, teleservicios y servicios suplementarios como llamada en espera, llamadas de emergencia y llamadas en conferencia; baja transferencia de datos móviles para aplicaciones de mensajería entre otros. [53]

La telefonía móvil tiene 2 modalidades para su servicio: Prepago, que funciona por medio de recargas periódicas y Pos-pago, en la que se fijan montos y productos incluidos en el servicio. [56]

3.8.5. Bandas de frecuencias

GSM puede operar en diversas frecuencias, en México generalmente operan en las bandas de 850MHz y 1900MHz. Para aprovechar las bandas de frecuencias, divide las frecuencias de su espectro radioeléctrico en diferentes portadoras de 200kHz., cada una de estas portadoras se divide en 8 ranuras de tiempo. [54] y [56]

3.8.6. Tipos de Acceso múltiple

GSM utiliza la combinación de técnicas Acceso Múltiple por División de Tiempo (TDMA del inglés time division multiple access) y Acceso Múltiple por División de Frecuencia (FDMA del inglés frequency division multiple access), para proporcionar acceso simultaneo de múltiples unidades móviles a estación base. [53]

El acceso múltiple por división de frecuencia, se caracteriza porque el ancho de banda del sistema de comunicación es dividido en bandas de frecuencia menores (canales de frecuencia) con una separación suficiente entre ellas (banda de guarda), de tal forma que se evita la interferencia y los usuarios no requieren sincronizarse para poder disponer de un canal. [57]

El acceso múltiple por división de tiempo, es una técnica en la que los usuarios realizan una transmisión utilizando la misma portadora durante intervalos específicos de tiempo (ranuras de tiempo), por lo que se hace necesaria la sincronización de los usuarios para evitar interferencias. [57]

3.8.7. Modulación

El sistema de modulación usado en GSM debe tener una alta eficiencia espectral y presentar una fuerte resistencia frente a efectos de ruido que introduce el canal de radio, el grupo de esquema de modulación con fase continua y envolvente constante se usa mucho en sistemas con atenuación variable debido a las interferencias y la atenuación; un miembro de esta familia es la modulación por desviación mínima Gaussiana (GMSK del inglés Gaussian Minimum Shift Keying), es una modulación de fase continua de portadora única, la cual utiliza un filtro Gaussiano, este filtro se encarga de aplanar la trayectoria de fase de la señal y por lo tanto, estabiliza las variaciones de la frecuencia a través del tiempo, por lo que la amplitud se mantiene constante. Esta modulación se caracteriza por tener envolvente constante y permite reducción del ancho de banda requerido. [58]

3.8.8. Arquitectura.

La arquitectura del sistema GSM, mostrada en la Fig.11 consiste en tres subsistemas principales interconectados, que interaccionan entre sí y con los suscriptores a través de interfaces especificadas de red. [53]

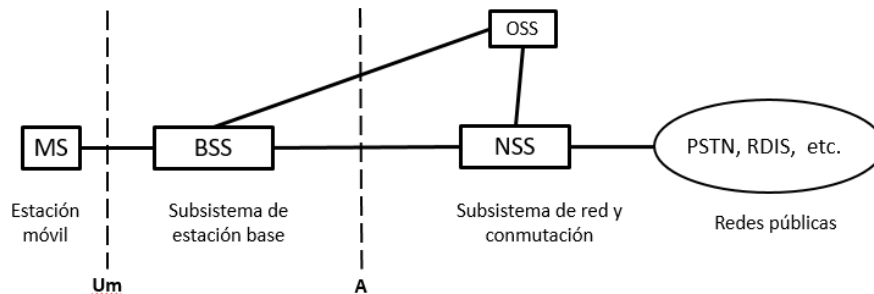


Figura 11. Arquitectura del estándar GSM [53]

Los tres subsistema principales de la red GSM son: el subsistema de estación base (BSS - Base Station Subsystem), el subsistema de red y conmutación (NSS - Network and Switching Subsystem) y subsistema de soporte operacional (OSS, Operational Support Subsystem). Aunque la estación móvil es, técnicamente, otro subsistema, en general se le considera como parte del subsistema de estación base. [53]

Podemos distinguir las principales funciones de los subsistemas que constituye la arquitectura básica de un sistema GSM:

- MS - Estación móvil: Está comprendida por las terminales de usuario tales como teléfonos móviles, su comunicación con la red tiene lugar vía la interfaz radio Um, conocida como interfaz aérea. [30]
- BSS - Subsistema de estación base: se encarga de la gestión de los recursos para la transmisión vía radio, se utilizan para formar las células de cobertura GSM, permite elegir la celda correcta y seleccionar el canal de radio más apto para efectuar la conexión.[30]
- NSS - Subsistema de red y conmutación: realiza el establecimiento, enrutamiento, control y terminación de llamadas; enlaza a los subsistemas de estaciones base de una zona a través de enlaces punto a punto, constituyendo así la interfaz A. Contiene un registro de localización local, que es una base de datos donde se almacenan parámetros de los abonados móviles; y también hace uso de un registro de identificación del equipo, la cual almacena la identidad internacional del equipo móvil (IMEI). [30]
- OSS - Subsistema de soporte operacional: Soporta la operación y el mantenimiento del sistema y permite vigilar, diagnosticar y localizar las fallas en cada aspecto de la red GSM. [30]
- Redes externas: Permiten la interconexión de redes, como: PSPDN (Packet-Switched Public Data Networks), CSPDN (Circuit- Switched Public Data Networks), PSTN (Public- Switching Telephone Networks) y ISDN (Integrated Service Digital Network). [30]

3.8.9. Ventajas.

- La red GSM utiliza tecnología digital para comunicaciones de voz y mensajes de texto corto. [54]
- GSM proporciona movilidad a los usuarios haciendo uso de la tarjeta SIM, ya que los usuarios se pueden transferir fácilmente entre las redes y mantener el mismo número. [55]
- Al hacer uso de una conmutación de circuitos el retardo es menor. [59]
- De acuerdo con 3G Americas, el 91% del mercado en América Latina utilizan la tecnología GSM. [57]

3.8.10. Desventajas.

- Ineficiencia por dedicar ancho de banda cuando no se requiere, ya que cuando se ha establecido la comunicación entre las dos terminales se ocupa el canal de comunicación el cual permanece ocupado hasta que se libera el canal de comunicación. [59]
- Tarda en el establecimiento y cierre de la conexión, lo cual provoca un retardo adicional. [59]

3.8.11. GSM en el mercado.

Como se mencionó anteriormente GSM tiene la capacidad de transmitir datos y voz a una velocidad de transmisión de 9.6Kbits/s, el cual, le ha permitido ofrecer el servicio de mensajes cortos (SMS) de forma bidireccional. Existen módulos GSM en el mercado para realizar la transmisión y recepción de los mensajes cortos (SMS). Como se muestra en la tabla 7.

Módulos GSM/GPRS							
Fabricante	Modelo	Frecuencia de trabajo	Dispositivos Compatible	Tipo de tarjeta	Voltaje de funcionamiento	Antena	Referencia
SparkFun	SparkFun Cellular Shield - SM5100B	850MHz 900MHz	Arduino uno, Mega, ADK o Leonardo.	SIM	5V.	No incluida	[60]
Arduino	Arduino GSM Shield (integrated antenna) A000043	850MHz 900MHz	Arduino Uno, Mega, ADK o Leonardo.	SIM	5V.	Incluida	[61]
	Arduino GSM Shield (antenna connector) A000044	850MHz 900MHz	Arduino Uno, Mega, ADK o Leonardo.	SIM	5V.	No incluida	[62]
Gemalto	Gemalto M2M	850MHz 2100MHz	Arduino	SIM ó MIM	5V	Incluida	[63]

Tabla 7. GSM en el mercado.

3.9. Microcontroladores.

3.9.1. Historia.

En 1971 Intel fabrica el primer microprocesador llamado 4004, contaba con una memoria de 4 bits, reconocía 45 instrucciones y podía ejecutar una instrucción en 20 μ s. Después en 1972 el microprocesador 8008 contaba con una versión expandida de memoria de 8 bits. [64]

Además de producir el primer microprocesador, Intel también ha desarrollado muchos microcontroladores. En 1976, el 8048 fue uno de los primeros microcontroladores de Intel y fue utilizado como el procesador en el teclado de la computadora personal de IBM (International Business Machines). El microcontrolador 8051 siguió en 1980, convirtiéndose en una de las familias de microcontroladores más populares. [64] y [65]

En la década de los 80's comienza la ruptura entre la evolución tecnológica de los microprocesadores y la de los microcontroladores, ya que los primeros se han incorporado cada vez más y con mejores capacidades para las aplicaciones en donde se requiere el manejo de grandes volúmenes de información, por otro lado, los segundos han incorporado más capacidades que les permiten la interacción con el mundo físico en tiempo real. Muchos microcontroladores actuales como: el Microchip y Atmel, incorporan la tecnología de memoria flash; que permite transportar fácilmente la cantidad de información, lo que nos conduce a la disminución del tamaño, por esta razón los microcontroladores se han vuelto más pequeños y más potentes para los desarrolladores. [64] y [65]

3.9.2. Definición.

Un microcontrolador es un circuito integrado, que es capaz de ejecutar órdenes grabadas en su memoria. Su composición está dada por varios bloques funcionales, los cuales se emplean para controlar el funcionamiento de una tarea determinada para las que ha sido programado de forma autónoma. [66]

3.9.3. Composición.

El Microcontrolador está formado por un Microprocesador y tres principales unidades funcionales de una computadora, las cuales son: unidad central de procesamiento (CPU), memoria (RAM Y ROM), puertos de entrada y salida, y periféricos. [66]

3.9.3.1. Unidad central de procesamiento (CPU).

Es el núcleo del microcontrolador se encarga de interpretar, procesar y ejecutar información mediante instrucciones lógicas que definen instrucciones almacenadas en la memoria. Es lo que habitualmente llamamos microprocesador ya que forma parte de un microcontrolador. [67]

3.9.3.2. Memoria.

La memoria se utiliza para almacenar información durante un periodo determinado de tiempo. Los microcontroladores tienen una memoria en la que se almacenan datos y otra en la que almacena instrucciones. Ambas se caracterizan por el tiempo que se tarda en escribir o leer cualquier posición de la memoria. A continuación se indican las principales características de las diferentes memorias: [67]

I. Memoria de datos.

Se puede leer y escribir durante la ejecución del programa y pueden ser de dos tipos:

- Memoria volátil: Pierden información contenida en ellas cuando se les deja de aplicar la tensión de alimentación. Son memorias de lectura y escritura no simultaneas, es decir, este tipo de memorias solamente se pueden seleccionar una posición de memoria simultánea y realizar en ella la operación de escritura o lectura. En general se les suele conocer por el nombre de RAM (Random Access Memory). Todo microcontrolador contiene como mínimo una memoria de este tipo. [67]
- Memoria no volátil: Son memorias en las que se pueden leer y escribir mediante pulso eléctricos y reciben el nombre de EEPROM (Electrically Erasable programmable Read Only Memories). Se usan para almacenar un número pequeño de datos, que se debe conservar al aplicar la tensión de alimentación. [67]

II. Memoria de Instrucciones.

Esta memoria solamente se lee durante la ejecución del programa y no puede ser volátil, es decir, debe conservar la información cuando se le deja de aplicar la tensión de alimentación. En los microcontroladores se pueden utilizar todos los tipos de memorias pasivas no volátiles. Puede ser de tres tipos de memorias en los microcontroladores: [67]

- Memorias ROM (Read Only Memory): Son aquellas memorias que al colocarlas en un sistema digital solo pueden ser leídas, ya que la escritura o grabación es realizada por el fabricante durante su proceso de fabricación. [67]

- Memorias EPROM (Electrically Programmable Read Only Memories): Se utilizan en las aplicaciones en las que se prevé realizar cambios, no solo durante la fase de diseño sino también a lo largo de la vida útil del microcontrolador.[67]
- Memorias Flash: Son las más utilizadas para almacenar el programas permanentemente. [67]

3.9.3.3. Periféricos.

Para que el microprocesador se comunique con el microcontrolador necesita de unidades de entrada y de salida que codifiquen los mensajes para interpretarlos, a las unidades o dispositivos que funcionan como interfaz entre el mundo externo y el procesador se les llama periféricos. [66]

Existe dos formas de transmitir información entre un periférico y el procesador: en paralelo y en serie. La transmisión en paralelo utiliza todas las líneas de comunicación, y no requiere realizar ningún sincronismo entre el periférico y el procesador. Por su contrario, la transmisión en serie hace la transformación de paralelo a serie y transmite el byte, bit por bit; por lo que se requiere un sincronismo entre el procesador y el periférico. [66]

Los periféricos pueden clasificarse de forma general en periféricos de entrada y periféricos de salida. Se le llama periférico de entrada aquel que tiene como objetivo codificar los mensajes o señales del exterior para que el procesador pueda interpretarlos, por ejemplo un teclado o un sensor. Por otra parte, se tienen los periféricos de salida que simplemente permiten observar los resultados arrojados por el procesador de una manera más cómoda que si se presentaran como unos y ceros, por ejemplo una pantalla TFT. [66]

3.9.3.4. Puertos de entrada y salida.

Los puertos E/S (entrada/salida) se usan para intercambiar información y controlar todo el entorno externo al microcontrolador, generalmente los puertos de entrada o lectura se encarga de aceptar datos, mientras que los puertos de salida o de escritura realizan alguna función de control. El número de terminales de entrada/salida varía según el modelo del microcontrolador. [65] y [66]

3.9.4. Arquitectura de los microcontroladores.

Anteriormente se mencionó que los microcontroladores están formados por varias unidades funcionales. Dentro de ellos existe como elemento principal la CPU, que se interconecta con sus periféricos para formar lo que se conoce como arquitectura interna. Esta puede ser de dos tipos: Von Neuman y Harvard. [66]

Inicialmente todos los microcontroladores adoptaron la arquitectura de Von Neumann, que se caracteriza por disponer de una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. Actualmente la arquitectura Harvard es la más utilizada; dispone de dos memorias independientes una, que contiene sólo instrucciones y otra, sólo datos. Ambas memorias disponen de sus respectivos sistemas de acceso por lo que es posible realizar operaciones de acceso (lectura o escritura) simultáneamente en ambas memorias. [66]

Hoy en día se fabrican muchos tipos de microcontroladores comerciales tales como: Texas Instruments, Motorola, Atmel, Intel, Microchip, Toshiba, Nacional, etc. Todos ellos ofrecen microcontroladores con características más o menos similares, sin embargo, en términos generales se puede decir que todos los microcontroladores sirven para leer y ejecutar los programas del usuario. La selección de un microcontrolador para una determinada aplicación vendrá por las necesidades requeridas del sistema en cuestión; por ejemplo la placa Arduino ADK está basada en un microcontrolador ATmega 2560 de Atmel. [68] y [69]

El microcontrolador ATmega 2560 tiene tipos y cantidades de memoria que se alojan en su interior, entre las cuales se encuentran: la memoria SRAM, memoria de datos de tipo volátil donde se alojan los datos que necesita el programa para su correcto funcionamiento. Estos datos suelen tener un contenido variable a lo largo del tiempo de ejecución del programa, es decir, un dato puede contener un valor numérico entero, un número decimal, valor de tipo carácter, cadenas de texto u otros tipos de datos. En el caso del ATmega 2560 la memoria SRAM tiene una capacidad de 8KB; el ATmega 2560 también tiene una memoria EEPROM, memoria de datos de tipo no volátil con una capacidad de 4KB, así como también contiene una memoria Flash, memoria de instrucciones de tipo no volátil con una capacidad de 256KB. [68] y [69]

Si desea consultar mayor información y las características especificadas del ATmega 2560, consulte la referencia [69]

3.9.5. Herramientas de desarrollo basados en microcontroladores en el mercado.

Como se mencionó anteriormente el microcontrolador es un chip que ejecuta instrucciones almacenadas en su memoria. Existen herramientas de desarrollo en el mercado basadas en un microcontrolador como se muestran en la Tabla 8.

Herramientas de desarrollo basados en Microcontroladores								
Fabricante	Modelo	E/S Pines digitales	Entradas analógicas	Puertos de E/S	PWM	Memoria	Interfaz de programación	Referencia
Arduino	Arduino uno	14	6	2	6	32KB	USB	[70]
	Arduino mega 2560	54	16	8	14	128 KB	USB	[71]
	Arduino ADK	54	16	8	15	256KB	USB	[72]
	Arduino DUE	54	12	8	12	512KB	USB	[73]
	Arduino Leonardo	20	12	2	7	32KB	Micro USB	[74]
Raspberry pi	Raspberry Pi 1 Modelo B	17	-----	2	-----	512 MB	USB	[75]
	Raspberry Pi 1 Modelo A+	28	-----	2	-----	256 MB	USB	[76]
Intel	Intel Galileo Gen 2.	20	6	2	6	8MB	USB	[77]

Tabla 8. Herramientas de desarrollo basados en microcontroladores existentes en el mercado.

3.10. Pantalla TFT (Thin-film transistor).

3.10.1. Historia.

Las pantallas de cristal líquido (LCD) han estado disponibles desde hace más de medio siglo. Las primeras versiones populares de las TFT's fueron hechas de semiconductores compuestos, mostradas por primera vez en 1973. Sin embargo, no se realizó una producción en masa de este tipo de LCD, por complicaciones en el control de los semiconductores compuestos por las propiedades del material de película fina. [78]

El avance en el campo de las TFT dio origen a la primera TFT funcional hecha de silicio amorfo hidrogenado (a-Si:H) con una capa de dieléctrico (nitruro de silicio) en 1979. [78]

3.10.2. Definición.

Una pantalla con tecnología TFT (Thin-Film Transistor) es una pantalla de cristal líquido (LCD) basado en transistores, los cuales controlan la iluminación de la pantalla.

3.10.3. Composición.

La estructura del LCD se compone de dos placas de sustrato de vidrio, un filtro de color RGB, filtros de polarización, transistores, luz de fondo, entre otros. [78]

La figura 12 muestra las diferentes capas que conforman una pantalla TFT, enunciadas anteriormente. [78]

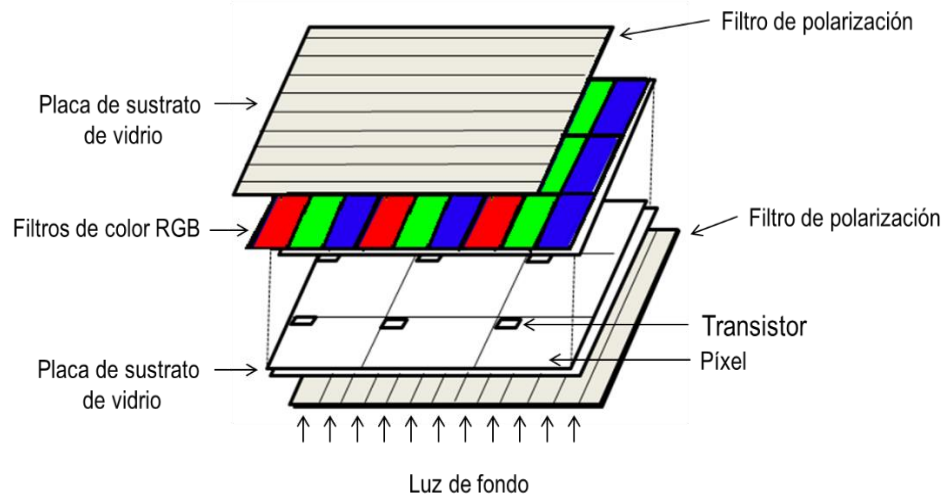


Figura 12. Estructura pantalla TFT. [78]

La TFT se le conoce como tecnología de “matriz activa” debida a que la TFT utiliza una rejilla de los transistores con la capacidad de mantener la carga durante un periodo limitado de tiempo, haciendo una función parecida al condensador, por lo que cada pixel se encuentra conectado a un transistor; el pixel deseado recibe y mantiene la carga, hasta la siguiente actualización, como a cada transistor le llega información de intensidad de corriente se puede encender y apagar a un ritmo más rápido, esto causa que la pantalla sea más brillante, muestra un movimiento más suave, mejorando la calidad de imagen. [79]

Cada pixel utiliza un transistor y se compone por 3 subpíxeles (celda): rojo, verde y azul, como se muestra en la Figura 13. Cada color admite 256 niveles de intensidad; para codificar un color es suficiente indicar las intensidades de cada uno de los tres colores. [79]

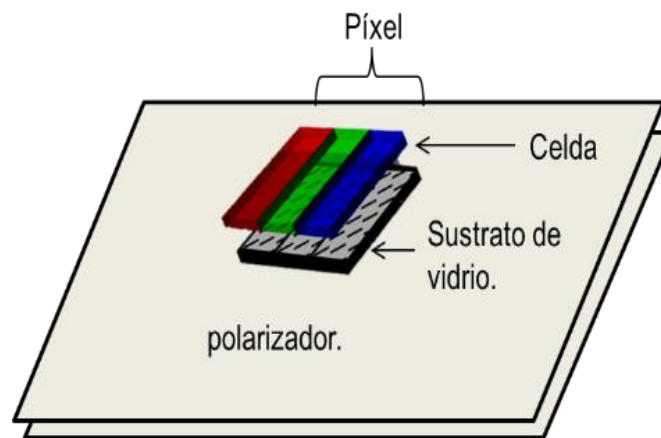


Figura 13. Composición de un pixel. [79]

3.10.4. Ventajas.

La ventaja de la TFT, es que cada píxel en una pantalla TFT es respaldada por un pequeño transistor, que es mucho más eficiente que los monitores antiguos (de tubos de rayos catódicos). Las pantallas TFT de LCD pueden mostrar texto nítido, colores vívidos, animaciones rápidas y gráficos complejos. Esta tecnología de transistores múltiples también permite un rápido re-trazado de la pantalla de manera que la imagen no parpadee. [79]

3.10.5. Pantallas TFT en el mercado.

Hoy en día se encuentran diversas marcas de pantallas TFT por lo que solo se hará mención de algunas marcas reconocidas en el mercado en México, como se muestra a continuación (Ver Tabla 9).

Pantalla TFT						
Fabricante	Modelo	Transmision	Memoria extena	Tamaño de pantalla	Dispositivos compatible	Referencia
Sparkfun	Beaglebone Negro Cabo LCD.	Seria	N/A	240 x 320 pixeles.	Beaglebone.	[80]
Ilitek	a-Si TFT LCD Single Chip Driver (ILI9325)	Serie	Tarjeta micro SD.	240 x 320 pixeles.	Arduino (excepto Leonardo).	[81]
Raspberry pi	PiTFT	Paralelo Interfaz SPI: SCK, MISO, MOSI.	N/A	480 x 320 pixeles.	-Pi modelo A o B. -Raspberry Pi 2 Model B.	[82]
Arduino	Robot LCD / LCD Module SD Card reader.	Paralelo Interfaz SPI: SCK, MISO, MOSI.	Tarjeta micro SD.	160 x 128 pixeles.	Arduino Esplora. Arduino AVR (Uno, Leonardo, Mega, etc.). Arduino Due.	[83]

Tabla 9. Pantallas TFT en el mercado.

Capítulo III

4. Marco Metodológico.

En este capítulo se describe el diseño, el desarrollo y las pruebas del sistema de comunicación siguiendo una metodología experimental que incluye una hipótesis. El objetivo general es que la persona con discapacidad motora o del habla pueda comunicarse hacia otros usuarios por medio de telefonía celular; es por esto, que en esta tesis se busca crear una herramienta tecnológica que facilite la necesidad de acceso a la comunicación en personas con discapacidad motora y del habla en México; basándonos en esta necesidad se tratará de dar una solución por medio de la implementación de un sistema de comunicación

El sistema de comunicación se desarrolló a partir de la hipótesis de realizar el envío de un mensaje por medio de la mente, basándonos en esta hipótesis se elaboró una investigación de las neurociencias para entender el funcionamiento del cerebro humano, una vez obtenida esta información, nos preguntamos ¿Es posible leer la mente humana?.

Comenzamos a indagar sobre las tecnologías que existen para “leer la mente”, sin embargo nos encontramos que efectivamente existen dispositivos para adquirir información sobre el comportamiento del cerebro, pero estas tecnologías no leen la mente, es decir, que no extraen pensamientos de las personas. Sino que son tecnología capaz de identificar ondas cerebrales a diferentes frecuencias y amplitudes. Usualmente esta tecnología se describe por su nombre en inglés Brain-Computer Interface (BCI). El funcionamiento básico de una BCI es medir la actividad cerebral, procesarla para obtener las características de interés de la señal. Con base a esta información la siguiente pregunta que nos hicimos fue ¿Qué parte de las características de las señales nos interesa? y ¿Qué acción utilizaríamos para manipular la BCI y a su vez realizar el envío de un mensaje? , entonces se realizó lo siguiente, primero investigamos el tipo de señales que adquiere la BCI, como respuesta obtuvimos que la interfaz cerebro computador (BCI) se basa en el análisis de las señales electroencefalográficas (EEG), captadas durante la actividad cerebral con la finalidad de controlar un componente externo. El sistema de comunicación está dirigido a personas con discapacidad motora y del habla pero que no tengan problemas mentales. Para esta población de discapacidad presentan alteraciones en los músculos impactando la movilidad

de la persona. Los músculos que no pierden movimiento de alguna manera son los parpados, es decir el movimiento ocular, es por eso que, se decidió la acción del parpadeo para manipular la BCI y su vez realizar el envío de un mensaje.

A continuación se muestra en la Figura 14 un ordinograma del proceso de diseño e implementación de un sistema electrónico para comunicarse la persona con discapacidad motora y del habla hacia otros usuarios.

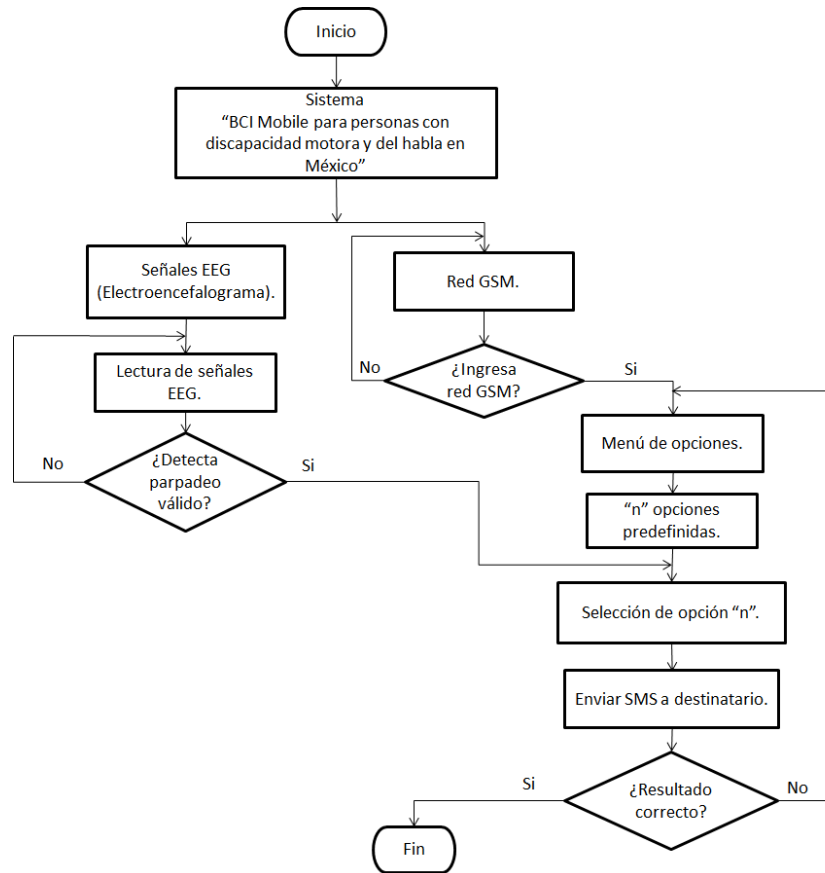


Figura 14. Organigrama para la implementación del sistema.

El sistema de comunicación se divide en tres etapas: en la primera etapa, se extraerán y adquirirán las señales electroencefalográficas (EEG) ubicadas en el cerebro del usuario con discapacidad motora o del habla por medio de una diadema BCI, estas señales EEG de tipo analógico serán referentes al parpadeo del usuario; una vez que se adquieran las señales EEG analógicas, serán procesadas por medio de un filtro, muestreo, cuantificación y codificación para adquirir las señales de tipo digital, este proceso lo realiza la diadema BCI; en la segunda etapa, serán enviadas las señales digitales al sistema, el cual se encargará de procesar y manipular las señales EEG enviadas por la BCI, para finalmente

en una tercera etapa llevar a cabo el envío de mensajes SMS hacia otros usuarios o sistemas en México. Como se muestra en la Figura 15.



Figura 15. Sistema de comunicación.

4.1. Diseño del Sistema.

Uno de los problemas fundamentales en la sociedad es la exclusión de las personas con discapacidad en el uso de los servicios de la telefonía celular: se busca crear una herramienta tecnológica, la cual facilite su acceso a la comunicación. Dando como solución la implementación de un sistema de comunicación que lleva por nombre sistema BCI Mobile; con este sistema, el usuario se comunicará haciendo uso de una herramienta llamada BCI, del cual se obtendrán señales analógicas de tipo EEG, las cuales se manipularán para dar órdenes a un dispositivo externo e inmediatamente realizar el envío de mensajes hacia otros usuarios o sistemas, creando el sistema BCI Mobile.

La interfaz cerebro-computador obtiene, decodifica y traduce la actividad de las señales eléctricas cerebrales para producir una acción, por lo que mediante el uso de una BCI no invasiva se obtendrán señales de tipo electroencefalográficas (EEG), las cuales serán detectadas a través de electrodos aplicados sobre la piel del usuario. Una vez obtenidas las señales EEG, se acondicionarán para su procesamiento posterior mediante etapas de amplificación, filtrado y digitalización. Ya acondicionadas las señales se pretende crear un enlace, en el cual serán enviadas hacia un receptor mediante la tecnología WPAN.

Las señales eléctricas cerebrales ya digitalizadas, se transformarán en código de programación creados en el software Arduino, que ejecutará los datos para manipular y operar algún dispositivo por medio de un microcontrolador en tiempo real; este algoritmo de programación se basa en un diagrama de flujo que se muestra en el Anexo A; para que

finalmente un microcontrolador envíe los algoritmos obtenidos hacia un transmisor, el cual se encargará de realizar el envío de mensajes SMS para transmitirlos por un enlace inalámbrico hacia un receptor (teléfono móvil); el enlace se podrá llevar a cabo haciendo uso de la tecnología GSM.

4.1.1. Funcionamiento y composición del sistema BCI Mobile.

Por medio de la implementación de una interfaz cerebro computador (BCI) no invasivo en el usuario 1 (Bloque 1), se adquirirán señales analógicas de tipo electroencefalográficas (EEG); dichas señales se filtran, procesan, y digitalizan para ser transferidas hacia una interfaz Bluetooth, con el que cuenta la BCI; estas señales ya digitalizadas las recibirá un microcontrolador por medio de una interfaz Bluetooth (Bloque 2). El microcontrolador se pretende programar para manipular y procesar la información contenida en las señales EEG (Bloque 3); ya obtenida la información, se implementará un menú de opciones para facilitar al usuario con discapacidad motora y del habla a elegir entre diferentes tipos de mensajes predefinidos.

El menú de opciones será visualizado a través de una pantalla (Bloque 4); una vez seleccionado el mensaje predefinido por el usuario, la información será enviada hacia un transmisor GSM (Bloque 5), el cual se encargará de realizar el envío del mensaje para transmitirlo por un enlace inalámbrico hacia un receptor (teléfono móvil), como se muestra en la Figura 16.

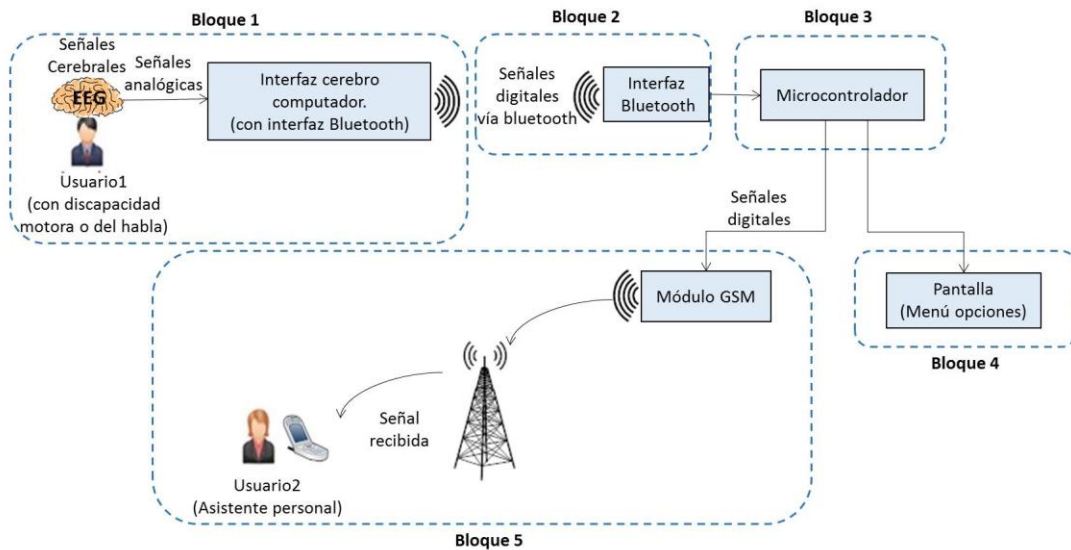


Figura 16. Componentes del sistema BCI Mobile.

4.1.2. Especificaciones técnicas.

Las especificaciones técnicas se desglosarán siguiendo los bloques que conforma el sistema BCI Mobile como se muestra en la Figura 16.

El bloque 1 abarca la adquisición y transmisión de las señales EEG haciendo uso de una diadema BCI; el bloque 2 adquiere la señal enviada de la BCI al sistema BCI Mobile; el bloque 3 se encarga del procesamiento y manipulación de los datos recibidos por el bloque 2, por medio de algoritmos de programación; el bloque 4 muestra un menú de opciones hacia el usuario con discapacidad motora o del habla por medio de una pantalla TFT; el bloque 5 realiza el envío de los mensajes predefinidos hacia otros usuarios o sistemas.

A continuación se muestra tablas comparativas por cada bloque de los componentes del sistema BCI Mobile.

4.1.2.1. Bloque 1: Interfaz Cerebro Computador.

El proyecto del Sistema BCI Mobile se basó en una Interfaz cerebro-computador (BCI) de fácil manipulación y que ofreciera recursos suficientes para poder hacer uso de la misma, por lo que se llevó a cabo una tabla comparativa para la elección de la BCI; como se muestra a continuación en la Tabla 10.

Brain Computer Interface (No invasiva)								
Fabricante	Modelo	Número de sensores	Tipo de sensores	Herramienta de desarrollo	Tipo de Batería/ duración	Tipo de Conexión	Precio MXN.	Referencia
Emotiv	EPOC/E POC+	14	Sensores salinos.	SDK, c, c#, C++, java, Python.	Batería recargable, 12Hr de uso continuo	Bluetooth	\$5586.00	[23]
Emotiv	Insight	5	Biosensor polímero hidrófilo	SDK, c, c#, C++, java, Python	Batería recargable, 4Hr de uso continuo	Bluetooth	\$4186.00	[24]
Neurosky.	MindWave mobile	1	Sensor giroscópico.	Arduino, SDK, iPhone, Raspberry Pi y .NET.	Batería AAA con 8 Hr de duración	Bluetooth	\$1,119.86	[25]
InteraXon	Muse	7	Señor muse	SDK básico para desarrollo	5 Hr de duración	Bluetooth	\$4186.00	[26]
OpenBCI	Kit Open BCI	8	Biosensores	Arduino	Batería AA de 6V.	Bluetooth	\$11199.86	[27]
iFocus band	Focus Band	1	----	SDK	----	Bluetooth	\$6930.00	[28]

Tabla 10. Tabla comparativa de interfaces cerebro computador.

La MindWave Mobile tiene una forma de diadema y hace uso de un solo electrodo ubicado en la parte Frontal, según el sistema internacional 10-20 hace uso del electrodo Fp1, por lo que con el sensor se permite la lectura de las señales EEG para la detección del parpadeo. A continuación se muestran en la Figura 17, gráficas de las ondas cerebrales EEG típicas y las pruebas que se realizaron con la herramienta software Matlab con respecto a un parpadeo.

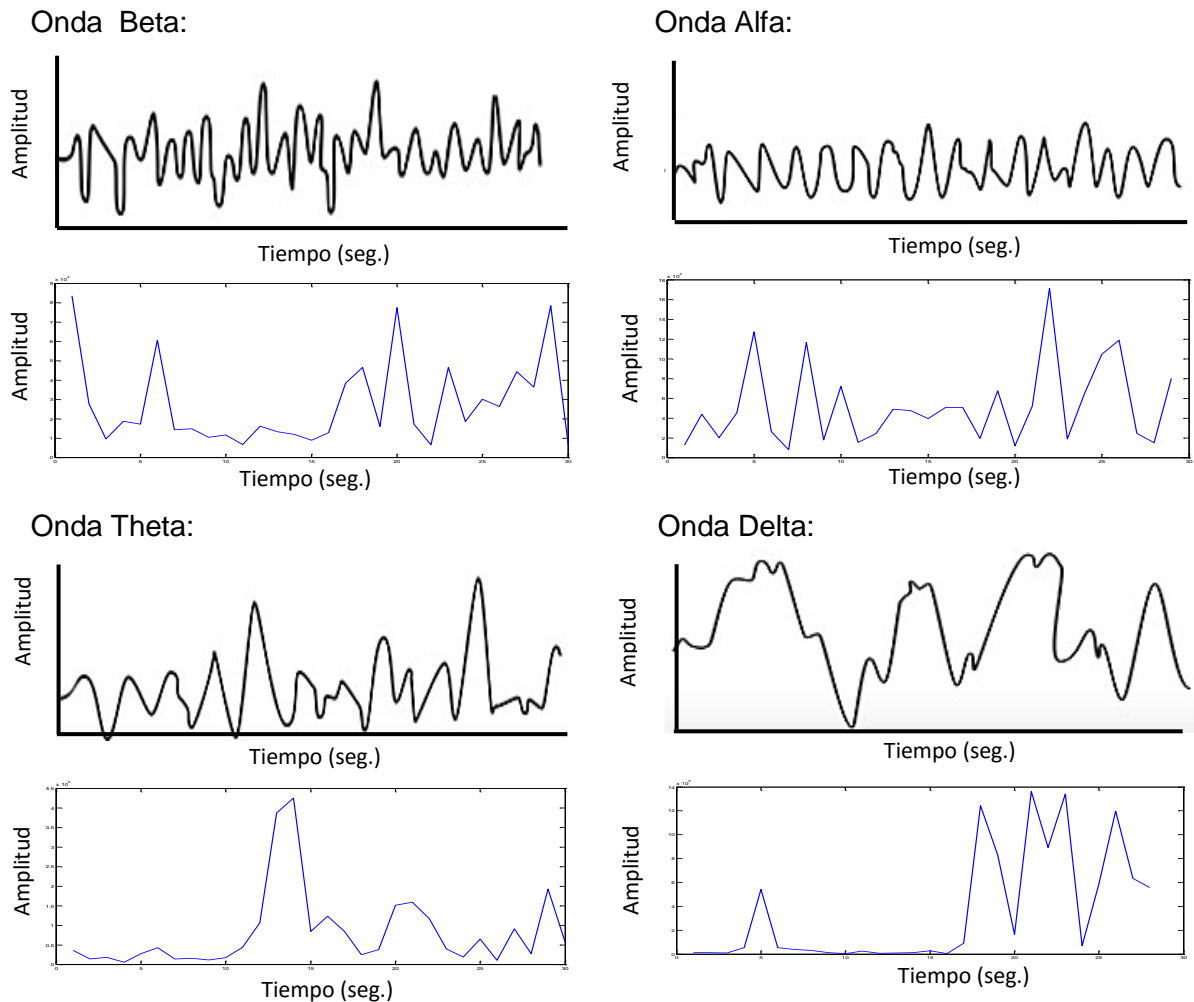


Figura 17. Lóbulos Cerebrales.

Es importante mencionar que los valores de las gráficas de cada una de las señales no tienen unidades, solo son significativos cuando se comparan entre cada una de las señales EEG y así mismas. Hay que tomar en cuenta que en cada una de las gráficas mostradas anteriormente sobre el parpadeo de las señales electroencefalográficas, tienen una duración aproximadamente de 30 segundos donde en los primeros 15 segundos no se

realizó ningún movimiento de parpadeos y de los 16 a 30 segundos se realizaron constantes parpadeos voluntarios.

Se puede concluir que las gráficas de cada una de las señales EEG que se realizaron en el software Matlab con respecto al parpadeo voluntario tienen una forma de onda muy similar a las ondas típicas de las señales EEG.

También se escogió la BCI MindWave Mobile de Neurosky ya que es un recurso de bajo costo y cuenta con las suficientes herramientas de desarrollo para manipular sus funciones.

MindWave Mobile cuenta con una interfaz Bluetooth integrada en la diadema para su comunicación hacia otros dispositivos que cuenten con la tecnología Bluetooth como son los teléfonos móviles y ordenadores, entre otros. Para realizar una comunicación entre la diadema MindWave Mobile y otros dispositivos haciendo uso de la tecnología Bluetooth; la MindWave Mobile hace uso de ciertos perfiles Bluetooth; como son: SSP- Serial Port Profile, A2DP - Advanced Audio Distribution Profile, AVRCP - Audio / Video Remote Control Profile, HSP - Headset Profile y HFP - Hands-Free Profile.

La interfaz Bluetooth que utiliza para comunicarse la MindWave Mobile es de clase 2, por lo que tiene un alcance hasta de 10 metros sin obstáculos.

4.1.2.2. Bloque 2: Interfaz Bluetooth.

La MindWave es una diadema que se comunica a través Bluetooth por lo que se eligió esta interfaz Bluetooth, para poder recibir datos de la diadema hacia algún microcontrolador, se llevó a cabo una tabla comparativa para la elección de la interfaz Bluetooth; como se muestra en la Tabla 11.

Interfaces Bluetooth								
Fabricante	Modelo	Velocidad	Clase	Voltaje	Antena	Configuración	Precio MXN.	Referencia
Botscience Worldwide Imports	HC 06	1200 a 115200 bps.	2	3.3V-6V	Incluida	Comandos AT	\$523.16	[41]
Botscience Worldwide Imports	HC 05	1200bps a 1.3Mbps	2.0 +EDR	3.6 V.- 6 V.	Incluida	Comandos AT	\$398.76	[42]
SparkFun	Bluesmirf Mate Silver	2400 a 115200 bps.	2	3.3V-6V	Incluida	Comandos AT	\$397.96	[43]
SparkFun	Bluesmirf Gold	2400 a 115200 bps	1	3.3V-6V	Incluida	Comandos AT	\$557.46	[44]
Adafruit	Bluefruit EZ Link	2400 a 230400b bps	1	3V	incluida	Comandos AT	\$399.60	[45]

Tabla 11. Tabla comparativa de interfaces Bluetooth.

La interfaz a usar será el Bluesmirf Mate Silver; es de fácil manipulación para llevar a cabo la comunicación entre la diadema MindWave Mobile y el microcontrolador, ya que cuenta con gran variedad de comandos AT. Por otra parte el Bluesmirf Mate Silver es más económico a comparación de otras marcas mostradas anteriormente; se comunica y enlaza con otros dispositivos Bluetooth haciendo uso del perfil SSP- Serial Port Profile. El perfil SSP de Bluetooth también lo utiliza la BCI MindWave Mobile; MindWave Mobile al hacer uso de una interfaz Bluetooth de clase 2 y que utiliza el perfil SSP para su comunicación, se puede decir que la MindWave Mobile es compatible con el Bluesmirf Mate Silver.

4.1.2.3. Bloque 3: Microcontrolador.

Para poder utilizar, procesar y manipular los datos provenientes de las señales EEG que nos proporciona la diadema MindWave Mobile, se hará uso de un microcontrolador, por lo que se llevó a cabo una tabla comparativa para la elección de las herramientas de desarrollo basados en un microcontrolador, como se muestra en la Tabla 12.

Herramientas de desarrollo basados en microcontroladores.									
Fabricante	Modelo	E/S Pines digitales	Entradas analógicas	Puertos de E/S	PWM	Memoria	Interfaz de programación	Precio MXN	Referencia
Arduino	Arduino uno	14	6	2	6	32KB	USB	\$360.00	[70]
	Arduino mega 2560	54	16	8	14	128 KB	USB	\$880.00	[71]
	Arduino ADK	54	16	8	15	256KB	USB	\$1050.00	[72]
	Arduino DUE	54	12	8	12	512KB	USB	\$840.00	[73]
	Arduino Leonardo	20	12	2	7	32KB	Micro USB	\$380.00	[74]
Raspberry pi	Raspberry Pi 1 Modelo B	17	----	2	----	512 MB	USB	\$548.54	[75]
	Raspberry Pi 1 Modelo A+	28	----	2	----	256 MB	USB	\$429.00	[76]
Intel	Intel Galileo Gen 2.	20	6	2	6	8MB	USB	\$989.23	[77]

Tabla 12. Tabla comparativa de herramientas de desarrollo basada en microcontroladores.

Se escogió el Arduino ADK, ya que hace uso de una memoria flash por que dispone de mayor espacio para su programación. El Arduino ADK nos brinda los pines MISO (Master In Slave Out), MOSI (Master Out Slave In), SCK (Serial Clock) y SPI (Interfaz de puerto serial); para hacer uso de una pantalla TFT; por otra parte, nos proporciona pines digitales, pines analógicos, pines PWM (modulación por ancho de pulso) y pines de comunicación (TX y Rx), así como ciertas características que nos permiten recibir las señales EEG para su manipulación.

La MindWave Mobile usa la comunicación serial inalámbrica para envío de la información de las señales EEG, por lo que se requiere que el Arduino ADK que se pueda acoplar externamente al módulo Bluetooth por medio del uso de pines de comunicación (Tx y Rx).

4.1.2.4. Bloque 4: Pantalla TFT.

Se hará uso de una pantalla TFT para mostrar un menú de opciones al que pueda acceder el usuario para manipulación del sistema BCI Mobile y poder realizar el envío de mensajes, por consiguiente se realizó una tabla comparativa para la elección de la pantalla TFT, como se muestra a continuación en la Tabla 13.

Pantalla TFT						
Fabricante	Modelo	Características	Tamaño de pantalla	Compatibilidad	Precio MXN	Referencia
Sparkfun	Beaglebone Negro Cabo LCD.	-Pantalla táctil. -Voltaje de 5v y 3.3v. -Utiliza el Sistema RGB.	240 x 320 pixeles.	Beaglebone.	\$1,087.51	[80]
Ilitek	a-Si TFT LCD Single Chip Driver (ILI9325)	-Tarjeta micro SD. -Voltaje de 5v y 3.3v.	240 x 320 pixeles.	Arduino (excepto Leonardo).	\$972.91	[81]
Raspberry pi	PiTFT	-Interfaz SPI: SCK, MISO, MOSI. -Pantalla táctil TFT. -Presenta imágenes o video. -Pantalla táctil.	480 x 320 pixeles.	-Pi modelo A o B. -Raspberry Pi 2 Model B.	\$741.22	[82]
Arduino	Robot LCD / LCD Module SD Card reader.	-Tarjeta micro SD. -dibuja texto e imágenes. -Hardware y software SPI (Interfaz de Puerto Serial): SCK, MISO, MOSI. -Pantalla táctil TFT.	160 x 128 pixeles.	Arduino Explora. Arduino AVR (Uno, Leonardo, Mega, etc.). Arduino Due.	\$355.02+IVA	[83]

Tabla 13. Tabla comparativa de pantallas TFT.

Se eligió la pantalla Robot LCD de Arduino, ya que es compatible con la plataforma Arduino ADK. Robot LCD hace uso de los pines de comunicación: MISO (Master In Slave Out), MOSI (Master Out Slave In), SCK (Serial Clock) y SPI (Interfaz de puerto serial); por lo que es de fácil conexión hacia la plataforma sin necesidad de agregar otro dispositivo externo; En general, esta pantalla tiene un tamaño compacto que permite realizar animaciones rápidas para facilitar el manejo de las imágenes y dibujar texto nítido; para que sean visualizados por el usuario de manera dinámica.

4.1.2.5. Bloque 5: Módulo GSM.

Se propuso para el proyecto hacer uso de la tecnología GSM ya que el sistema BCI Mobile solo realizará envío de mensajes SMS; Para escoger la interfaz GSM a utilizar, se realizó una cotización para la elección de la interfaz (shield) GSM, como se muestra a continuación en la Tabla 14.

Módulos GSM/GPRS								
Fabricante	Modelo	Frecuencia de trabajo	Dispositivos Compatible	Tipo de tarjeta	Voltaje de funcionamiento	Antena	Precio MXN	Referencia
SparkFun	SparkFun Cellular Shield - SM5100B	850MHz 900MHz	Arduino uno, Mega, ADK o Leonardo.	SIM	5V.	No incluida	\$1235.00	[60]
Arduino	Arduino GSM Shield (integrated antenna) A000043	850MHz 900MHz	Arduino Uno, Mega, ADK o Leonardo.	SIM	5V.	Incluida	\$1830.17	[61]
	Arduino GSM Shield (antenna connector) A000044	850MHz 900MHz	Arduino Uno, Mega, ADK o Leonardo.	SIM	5V.	No incluida	\$1764.66	[62]
Gemalto	Gemalto M2M	850MHz 2100MHz	Arduino	SIM ó MIM	5V	Incluida	\$2423.52	[63]

Tabla 14. Tabla comparativa de módulos GSM.

Se eligió el shield Arduino GSM A000044, ya que es compatible con el Arduino ADK; como provienen del mismo fabricante no hay problemas de compatibilidad. El shield Arduino GSM no cuenta con una antena integrada por lo que se adquirió una antena con las siguientes características [84]:

- Rango de frecuencia: 850 / 900 / 1800 / 1900MHz
- VSWR: 1.5: 1

- Ganancia: 3.0dBi
- Conector: SMA macho ángulo recto
- Tamaño: 110 mm de alto

Si desea consultar mayor información y especificaciones acerca de la antena GSM, consulte la referencia [84].

Una vez seleccionados los dispositivos (componentes) a utilizar, el Sistema BCI Mobile estará comprendido de la siguiente manera (Ver Figura 18):

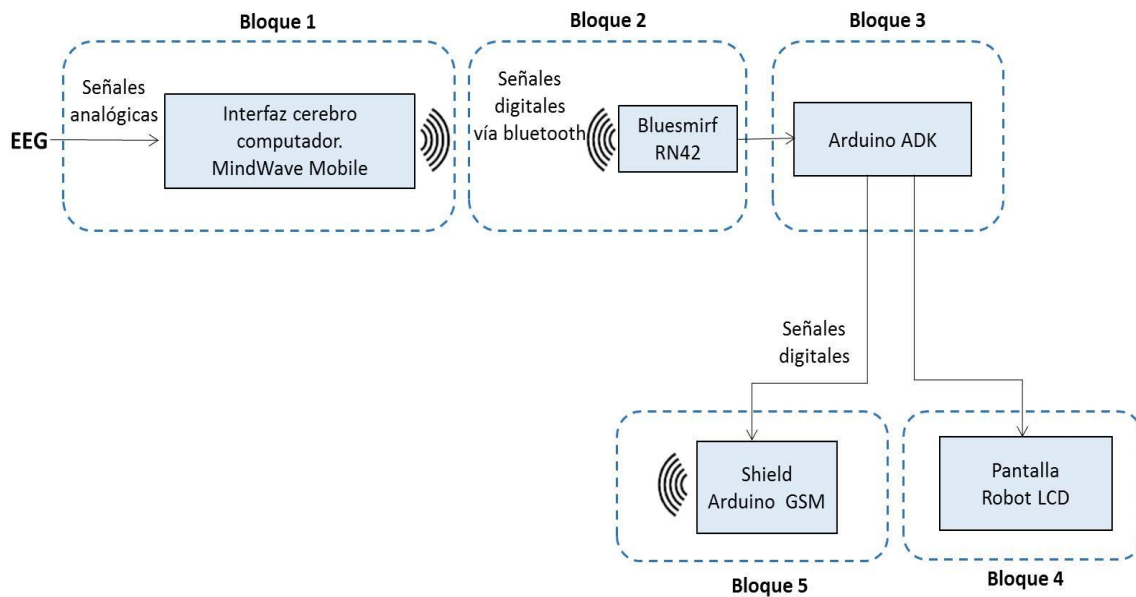


Figura 18. Dispositivos que componen al Sistema BCI Mobile.

4.1.3. Cotización del Sistema BCI Mobile.

A continuación se muestra la cotización final de los dispositivos a usar en el sistema BCI Mobile, haciendo uso de los componentes antes mencionados.

Cotización General de recursos materiales				
Dispositivo	Fabricante	Número de piezas	Precio MXN.	Referencia
Interfaz Cerebro computador: MindWave	Neurosky	1	\$1,119.86	[31]
Módulo Bluetooth: Bluesmirf RN42	Sparkfun	1	\$397.96	[47]
Plataforma: Arduino ADK	Arduino	1	\$1050.00	[88]
Arduino GSM shield	Arduino	1	\$1848.30	[75]
Robot LCD	Arduino	1	\$ 355.02	[102]
ANTENA GSM / GPRS QUDBAND		1	\$ 55.17	[103]
Precio Total MXN.			\$ 4771.14	----

Tabla 15. Cotización del sistema BCI Mobile.

Es importante tomar en cuenta que la cotización final solo abarca el costo total de los componentes del sistema BCI Mobile, por tanto en la figura 19 se muestran los dispositivos en físico que comprenden al Sistema.

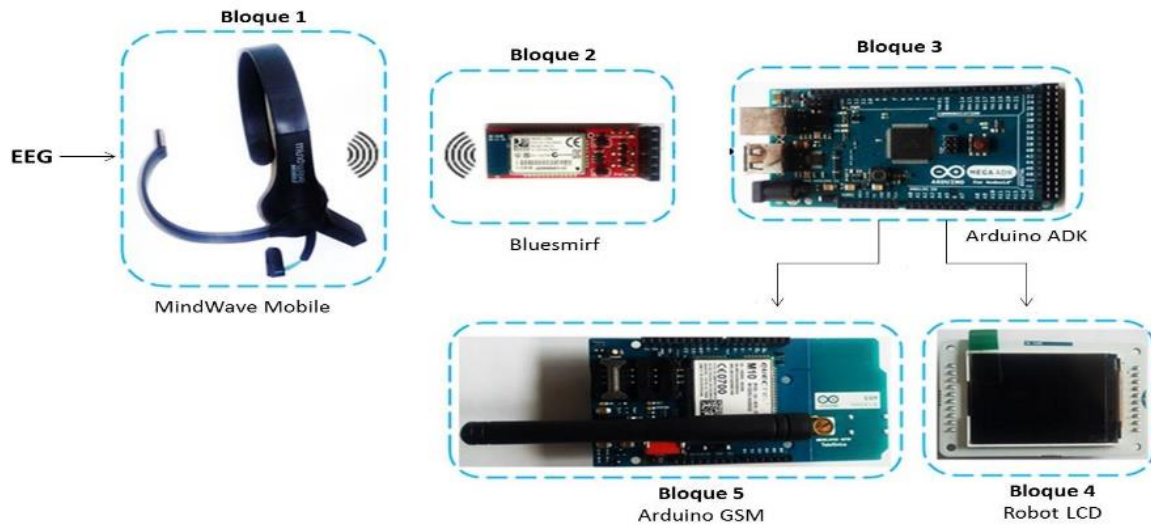


Figura 19. Dispositivos en físico que comprenden al sistema BCI Mobile.

4.2. Programación.

4.2.1. Programación para la configuración del Bluesmirf.

Se realizó un programa en código Arduino con la versión 1.0.5., además para la configuración de la interfaz Bluetooth llamada Bluesmirf se hace uso de la biblioteca SoftwareSerial.h, esta biblioteca nos permite una comunicación serie a través de pines digitales del Arduino ADK.

El Bluesmirf utiliza pines para transmisión y recepción de los datos a través de una interfaz serie del Arduino ADK, en nuestro caso se utilizó el pin digital PWM 10 para la recepción y el pin digital PWM 11 para la transmisión de datos del Arduino ADK; si desea consultar las especificaciones acerca de los pines digitales PWM 10 y 11, consulte la referencia [85].

```
1. #include <SoftwareSerial.h>
2. SoftwareSerial bluetooth(10,11);
```

El Bluesmirf hace uso de comandos AT, a continuación se enlistan los comandos a utilizar:

- Modo comando(\$\$\$)
- Cambio de velocidad del módulo Bluetooth sin paridad (U,9600,N)

- Verificación del estado del módulo.(D)
- Modo de operación.
 - I. Modo Esclavo. (SM,0)
 - II. Modo Maestro. (SM,1)
 - III. Modo Auto-conectado. (SM,3)
- Detección de dispositivos Bluetooth. (I)
- Conexión del Bluesmirf con otro dispositivo. (C, <dirección MAC>)
- Terminar conexión o modo dato. (---)

Para la configuración del Bluesmirf es importante que la velocidad de transferencia de datos se encuentre en baudios, por consiguiente el dispositivo a conectar con el Bluesmirf debe tener la misma velocidad para poder comunicarse. La velocidad del Bluesmirf se tiene que cambiar a 9600 baudios, para su configuración; debido a que el Bluesmirf por default tiene una velocidad de 115,200 baudios. Para el cambio de velocidad del Bluesmirf se hace uso del comando “U, velocidad, N”; donde “U” significa el cambio temporal de la UART, “velocidad” significa la velocidad de transmisión de datos y “N” significa que son bits sin paridad. Como se muestra en el siguiente código.

```
3. void setup(){
4.     Serial.begin(9600);
5.     bluetooth.begin(115200);
6.     bluetooth.print("$$$");
7.     delay(100);
8.     bluetooth.println("U,9600,N");
9. }
```

Para recibir los datos que provienen del Bluesmirf se hace uso de las siguientes funciones: la variable “bluetooth.available”; este comprueba si hay datos en el Bluesmirf y “Serial.print((char)bluetooth.read())”; lee byte a byte entrantes del Bluesmirf e imprime cada carácter recibido en el monitor serial. De la misma forma es para (Serial.available()),el cual, comprueba si hay datos en el buffer del Arduino y bluetooth.print((char)Serial.read()); lee byte a byte entrantes del puerto serie e imprime cada carácter recibido del Bluesmirf. A continuación se muestra el código de estas funciones en forma ordenada.

```
10. void loop(){
11.     if(bluetooth.available()){
12.         Serial.print((char)bluetooth.read());
13.     }
14.     if(Serial.available()){
15.         bluetooth.print((char)Serial.read());
16.     }
17. }
```

Se realiza la conexión física del Bluesmirf hacia el Arduino ADK de la siguiente manera.
 (Ver Tabla 16)

Bluesmirf	Arduino ADK
V _{CC}	3.3 V.
GND	GND
T _X	Pin digital PWM 10.
R _X	Pin digital PWM 11.

Tabla 16. Conexión con el Bluesmirf y el Arduino ADK.

Una vez realizada la conexión física del Bluesmirf con el Arduino ADK, se ejecuta el código mencionado anteriormente en el Software Arduino 1.0.5 (Ver Figura 20).

```

_040714p1 | Arduino 1.0.5-r2
Archivo Editar Sketch Herramientas Ayuda
_040714p1
#include <SoftwareSerial.h>
int bTx=11;
int bRx=10;
SoftwareSerial bluetooth(bRx,bTx);
void setup()
{
  Serial.begin(9600);
  bluetooth.begin(115200);
  bluetooth.print("$$$");
  delay(100);
  bluetooth.println("U,9600,N");
  bluetooth.begin(9600);
}
void loop()
{
  if(bluetooth.available());
  {
    // Send any characters the bluetooth prints to the serial moni
  }
}
Compilando el Sketch...
1 Arduino Mega 2560 or Mega ADK on COM9
  
```

Figura 20. Ejecución del código para la configuración del Bluesmirf.

Con el código ejecutado y los dispositivos conectados de manera correcta; se abre el Monitor Serial que se encuentra en el menú de Herramientas, para configurar el Bluesmirf y lograr un enlace inalámbrico hacia la diadema MindWave Mobile.

En el Monitor Serial se introducen los siguientes comandos:

- I. Para entrar en modo comando se escribe: **\$\$\$**, y se hace clic en "Enviar". (ver Figura 21)
- II. Se recibe como respuesta **CMD** (por parte de Bluesmirf).

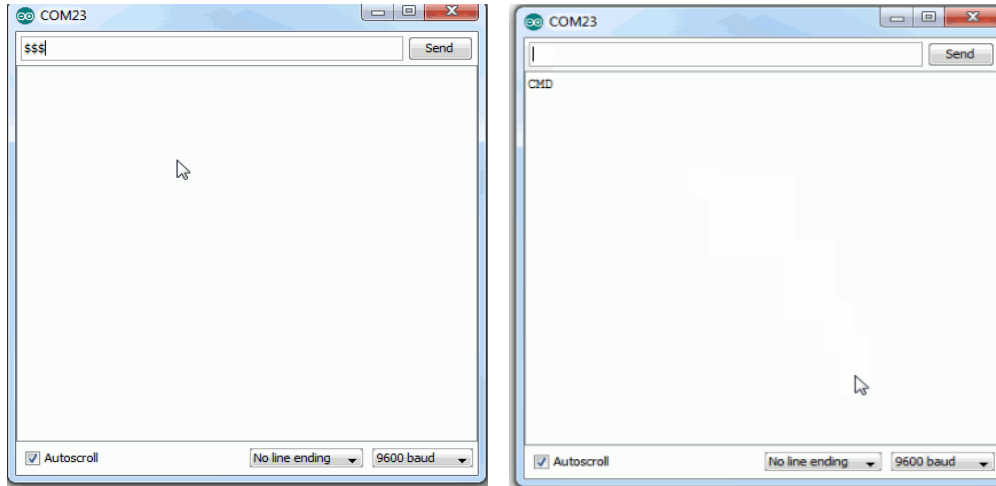


Figura 21. Bluesmirk Modo comando.

- III. Una vez que está en modo de comando, tendrá que cambiar en la opción "Newline" que se encuentra en la parte inferior derecha del Monitor Serial.
- IV. Se escribe el comando: **D**, este nos muestra el estado del Bluesmirk.
- V. Enseguida se escribe el comando: **SA, 0** para que la autenticación sea de forma abierta.
- VI. Se introduce el comando: **SL, N** para que sea sin paridad.
- VII. A continuación se escribe el comando: **SP, 0000**; este comando cambia la contraseña de fábrica por 0000.
- VIII. Se escribe el comando: **SM, 3**; este comando pone al Bluesmirk en modo de Auto-conexión. El modo de Auto-conexión se encarga de buscar un dispositivo Bluetooth sabiendo la dirección del dispositivo.
- IX. Se introduce el comando: **SR, Dirección**; este comando se encarga de almacenar la dirección de la diadema, la dirección de la MindWave Mobile a utilizar es: 74E54389590A.
- X. A continuación se cambia la velocidad del Bluesmirk con el comando: **SU, 57.6**; ya que la MindWave Mobile hace uso de una velocidad de 57600 baudios.
- XI. Una vez ingresados los comandos anteriores se vuelve a verificar el estado de configuración del Bluesmirk con el comando: **D**.
- XII. Para finalizar la configuración del Bluesmirk se hace uso del modo comando: - - -

4.2.2. Programación del Sistema BCI Mobile.

4.2.2.1. Definición de Biblioteca del sistema BCI Mobile.

Se realizó un programa en código Arduino con la versión 1.6.5, para el funcionamiento del sistema BCI Mobile; en la programación se hace uso de 4 bibliotecas:

```
1. #include <SPI.h>
2. #include <SD.h>
3. #include <TFT.h>
4. #include <GSM.h>
```

La biblioteca SPI.h, utiliza un protocolo de datos en serie síncrono para la comunicación entre el microcontrolador hacia la Robot LCD (pantalla TFT); la biblioteca SD.h, permite la lectura y escritura en tarjetas SD (contenida en la Robot LCD); la biblioteca TFT.h, permite al Arduino ADK comunicarse con Robot LCD para simplificar el proceso de elaboración de formas, líneas, imágenes y texto de pantalla.

4.2.2.2. Definición de variables del sistema BCI Mobile.

El sistema BCI Mobile funciona con una velocidad de 57600 baudios, el cual se define con la variable BAUDRATE:

```
5. #define BAUDRATE 57600
```

Para el funcionamiento de la pantalla Robot LCD se definen los pines a utilizar de acuerdo a las especificaciones del Arduino ADK (Consultar referencia [85], estos permiten la comunicación hacia el Arduino

```
6. #define sd_cs 12
7. #define cs 11
8. #define dc 9
9. #define rst 8
10. TFT TFTscreen = TFT(cs, dc, rst);
```

El pin 12 se destina para SD_CS, envía o recibe datos de la tarjeta micro SD contenida en la pantalla Robot LCD; el pin 11 se destina para CS (selección de chip), es usado para enviar y recibir datos únicos; el pin 9 se destina para DC (dato/selección de comandos), para el envío de datos o comandos; el pin 8 se destina para resetear la pantalla Robot LCD.

Para el funcionamiento del shield GSM Arduino se manda a llamar la biblioteca gsm.h, ya antes mencionada; se define una variable para el número PIN de la tarjeta SIM, sin embargo, la tarjeta SIM no es fija y puede ser intercambiada por lo que se deja en blanco

la variable; la clase GSM permite el acceso a la red GSM; la clase GSM_SMS es un clase para las funciones de la red GSM relacionados con el envío y recepción de mensajes SMS.

```
11.     #define PINNUMBER ""
12.     GSM gsmAccess;
13.     GSM_SMS sms;
```

Se definen otras variables para el procesamiento de los datos provenientes de las señales EEG que nos proporciona la diadema MindWave Mobile (Ver Anexo B). Se definieron variables de tipo carácter para el menú de opciones implementadas en la pantalla Robot LCD:

```
44.     char*bienvenido[]={ "B", "i", "e", "n", "v", "e", "n", " ", "i", "d", "o"};
45.     char* menuopr[]={ "Menu\n  principal", "SMS\n Basicos", "SMS de\n
control"};
46.     char*subm1[]={ "S.O.S", "Quiero\ncomer", "Quieroir\n
al\nsanitario", "  Tengo\n  sed", "Me duele\nla cabeza", "Me quiero\n
asear", "Salir" };
47.     char*subm2[]={ "Enciende\n la T.V.", "Apaga la\n T.V.", "Sube el\n
volumen", " Baja el\n volumen", "  Cambia\n de canal"};
48.     char*subtitulo[]={ "SMS Basicos", "SMS\n de control"};
49.     char* opcion[]={ "ENVIAR", "CANCELAR"};
```

La función PImage dibuja una imagen desde la tarjeta SD a la pantalla Robot LCD, en este caso iconos:

```
54.     PImage par;
55.     PImage mala;
56.     PImage buena;
57.     PImage nd;
```

4.2.2.3. Definición de funciones del sistema BCI Mobile.

La función void setup se ejecuta una sola vez, después de cada inicio o reinicio del sistema BCI Mobile; el void setup del sistema BCI Mobile inicializa los puertos seriales con la velocidad de 57600 baudios, inicializa la pantalla Robot LCD junto con la tarjeta de memoria microSD. En esta clase se cargan las imágenes a utilizar, como: los iconos del ojo, buena calidad y mala calidad del sistema BCI Mobile. La pantalla inicial está definida como logoa. Dentro de la función void setup, se inicializa el shield GSM para ingresar a la red y que el dispositivo se encuentre disponible para el envío de mensajes; como se muestra a continuación:

```
58.     void setup() {
59.         TFTscreen.begin();
60.         Serial1.begin(BAUDRATE);
61.         Serial.begin(BAUDRATE);
```

```

62.         TFTscreen.background(255, 255, 255);
63.         if (!SD.begin(sd_cs)) {
64.             return;
65.         }
66.     TFTscreen.begin();
67.     attachInterrupt(4,byteabyte,CHANGE);
68.     boolean notConnected = true;
69.     while (notConnected){
70.         if (gsmAccess.begin(PINNUMBER) == GSM_READY){
71.             notConnected = false;
72.         }else{
73.             delay(1000);
74.         }
75.     }
76.     bienvenida();
77. }
    
```

La función void loop es un bucle que se ejecuta un número indefinido de veces hasta que se apague o reinicie el sistema BCI Mobile. El void loop de sistema BCI Mobile contiene al menú completo del sistema.

```

88.     void loop() {
89.         menup();
90.     }
    
```

La función void bytebyte, contiene la lectura de las señales EEG cuya información es recibida byte a byte hasta formar una trama, La MindWave Mobile contiene dos tipos de tramas: trama larga y trama corta. [86]

La trama larga se compone de los siguientes campos con longitud en bytes:

Campo	Código Hexadecimal
Sincronía (Sync)	AA
Longitud de la trama	20
Calidad de la señal (PoorQuality)	02
Magnitud de las señales EEG de forma individual (ASIC_EEG_Power)	83
Longitud de los datos de ASIC_EEG_Power	18
Señal Delta	Variable (depende de la lectura del electrodo)
Señal Theta	Variable (depende de la lectura del electrodo)
Señal Alpha baja	Variable (depende de la lectura del electrodo)
Señal Alpha alta	Variable (depende de la lectura del electrodo)
Señal Beta baja	Variable (depende de la lectura del electrodo)
Señal Beta alta	Variable (depende de la lectura del electrodo)
Señal gamma baja	Variable (depende de la lectura del electrodo)
Señal gamma alta	Variable (depende de la lectura del electrodo)
Atención	04
Meditación	05
Checksum (CHK)	Variable (depende de los bytes que contiene la trama)

Tabla 17. Componentes y códigos de la trama larga. [86]

La trama larga se estructura de la siguiente manera:

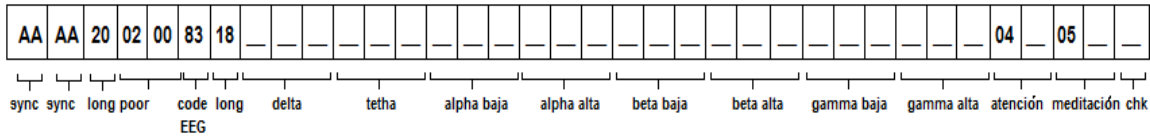


Figura 22. Trama Larga de la MindWave Mobile. [86]

Cabe mencionar que la trama larga contiene la magnitud de 8 tipos de ondas cerebrales, cada una de las señales EEG contiene 3 bits de manera ordenada.

La trama corta se compone de los siguientes campos con longitud en bytes:

Campo	Código Hexadecimal
Sincronía (Sync)	AA
Longitud de la trama	04
Magnitud de las señales EEG (Raw Wave Value)	80
Longitud de Raw Wave Value	02
Checksum (CHK)	Variable (depende de los bytes que contiene la trama)

Tabla 18. Componentes y códigos de la trama corta. [86]

La trama larga se estructura de la siguiente manera:

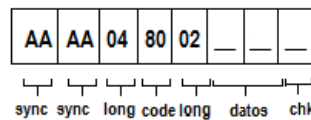


Figura 23. Trama corta de la MindWave Mobile. [86]

La función void byteabyte recibe los datos provenientes de la MindWave Mobile siempre y cuando este habilitado y disponible el puerto serial 1 (Ver Anexo B del renglón 92 al 172). En el puerto serial 1 se encuentra el receptor (R_x) del Bluesmirf.

La variable "variable" recibe todos los datos del puerto serial 1 (Ver Anexo B renglón 93), una vez que se empiezan a recibir datos byte a byte, se realiza un filtro de los datos dependiendo del tipo de trama y de los campos que la componen. El filtro se realizó para obtener dos tipos de datos: la calidad de las señales EEG (PoorQuality, Ver Anexo B renglón 161) y la magnitud de la señales EEG (Raw Wave Value, Ver Anexo B renglón 131); la magnitud de las señales EEG está dada por la lectura del electrodo de la MindWave Mobile, el cual nos proporciona información de 5 tipos de señales EEG como son: delta , theta, alpha, gamma y beta; la trama corta nos proporciona la magnitud total de las señales EEG con el código 80_{HEX}; por otro lado la calidad de la señal EEG (PoorQuality) se encuentra dada por la trama larga con el código 02_{HEX}.

Las tramas validas son aquellas que cumplen con dos condiciones: buena o excelente lectura de la señal EEG y que la trama este completa. Una trama completa se cumple siempre y cuando inicie con dos bytes de sincronía (AA_{HEX}) y termine con un byte de checksum; checksum se calcula con una suma con complemento a 1 de los datos de la trama que se recibe.

En el programa se puede observar que el checksum calculado por medio de la suma se denotó como la variable “generatedChecksum”, este debe ser igual a la variable “checksum” que se recibe de la trama.

```
126.         checksum=variable;
127.         generatedChecksum = dato[0]+dato[1]+dato[2]+dato[3];
128.         generatedChecksum= 255 - generatedChecksum;
```

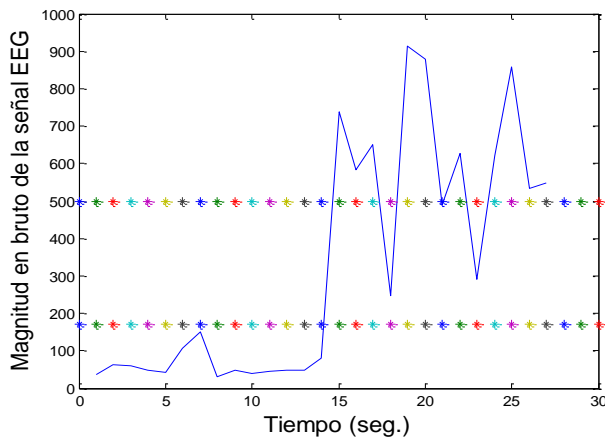
La función void parpadeo realiza la lectura de señales EEG, a partir de la magnitud de las señales EEG se denota el rango de magnitud de la señal indicando que existe un parpadeo; hay dos tipos de parpadeo: el parpadeo voluntario, que es una acción consciente y el involuntario o espontáneo que ocurre sin que exista ningún estímulo externo y está controlado por el cerebro. El sistema BCI Mobile utiliza los parpadeos voluntarios para la selección de menús, submenús y opciones definidas para el sistema. Se consideró un parpadeo voluntario válido siempre que la magnitud de las señales EEG se encuentre en el rango de entre 170 a 500; este rango se obtuvo con base a un sistema de prueba y error para clasificar la magnitud del parpadeo voluntario y diferenciarlo de los parpadeos involuntarios.

Para llevar a cabo estas pruebas como se mencionó anteriormente se utilizó software Matlab, esto es para graficar cada uno de los valores que se obtuvieron de la BCI referentes al parpadeo voluntario e involuntario. Las primeras pruebas que se realizaron, fue capturar los valores de la existencia de un parpadeo involuntario, estos valores se visualizaron en el software CoolTerm, los cuales son de baja magnitud. De la misma manera realizamos el mismo procedimiento para capturar valores del parpadeo voluntario, el cual observó que las señales son de mayor amplitud, es importante mencionar que se realizaron 14 pruebas; 7 para parpadeo voluntario y 7 para el parpadeo involuntario, de las cuales 2 no coinciden y las restantes pruebas se presentaron en el mismo nivel, es decir con la misma amplitud esto se debe a diversos factores: el estado de ánimo del usuario, distracción, entre otros. Estos factores de alguna manera afecta a las señales EEG porque aumentan o disminuyen considerablemente su amplitud. Sin embargo se realizaron 10 pruebas más para la

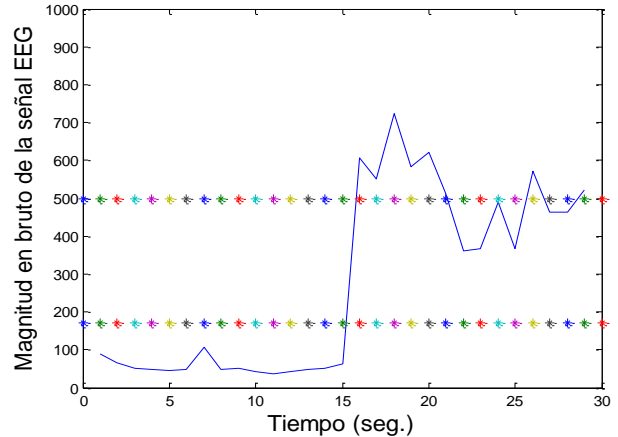
detección del parpadeo. Cada prueba se realizó en un lapso de 30 segundos donde los primeros 15 segundos aproximadamente de cada prueba, se realizaron parpadeos involuntarios y después de los 15 segundos se comenzaron a realizar parpadeos voluntarios; se graficaron estos valores en Matlab, esto es para comparar la magnitud en bruto de la señal y realizar un umbral para ajustar al nivel deseado.

El primer umbral se realizó para ajustar el punto máximo de la señal que representa el parpadeo involuntario, se estuvo ajustando poco a poco para poder llegar al nivel adecuado, esto significa, que 10 de las señales del parpadeo involuntario están por debajo de este umbral; como resultado obtuvimos un umbral de 170 en la mayoría de las señales. El segundo umbral se realizó para ajustar el nivel de las señales del parpadeo voluntario, es decir que en la mayoría de las pruebas sus puntos mínimos se encuentren por arriba del umbral; se ajusto poco a poco hasta llegar al nivel de 500, el cual fue el nivel adecuado. A continuación en la Figura 24 se muestran las 10 pruebas realizadas.

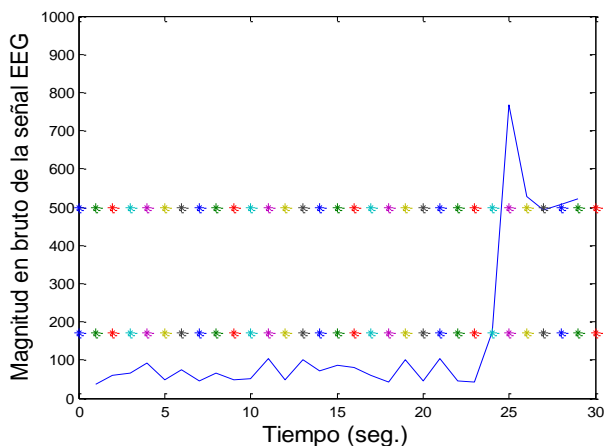
Prueba 1:



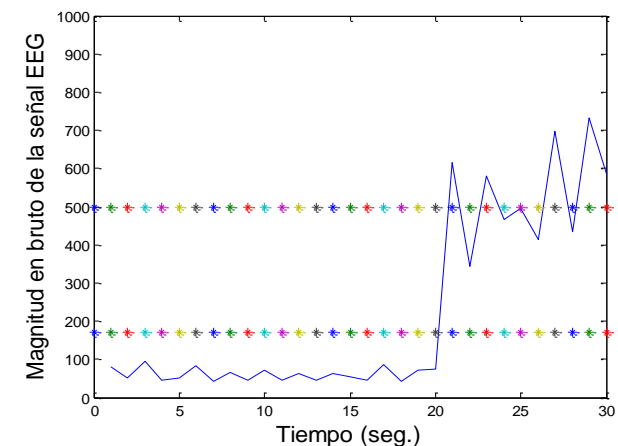
Prueba 2:



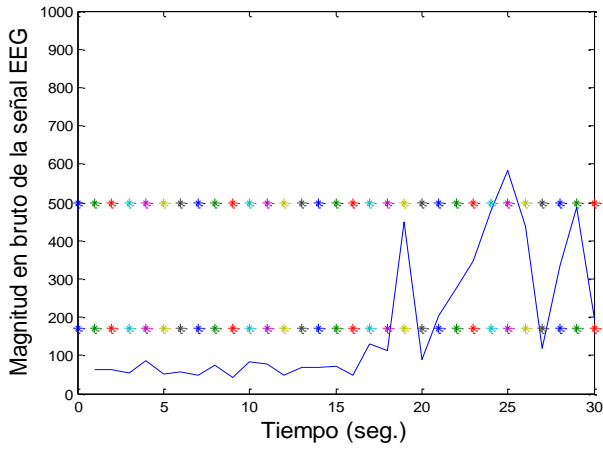
Prueba 3:



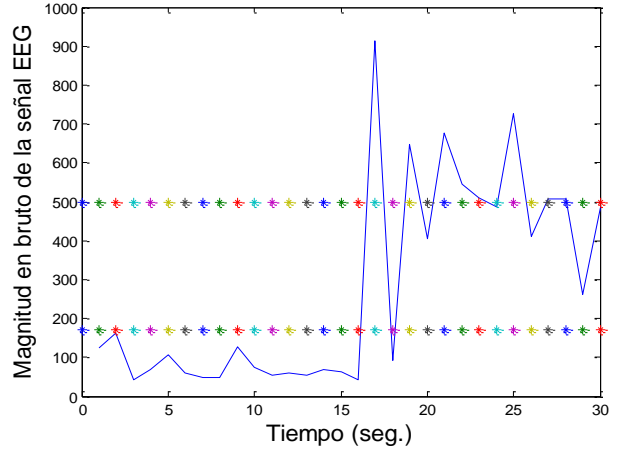
Prueba 4:



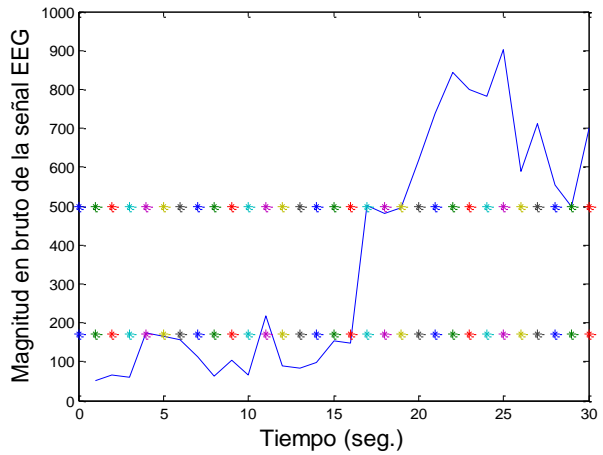
Prueba 5:



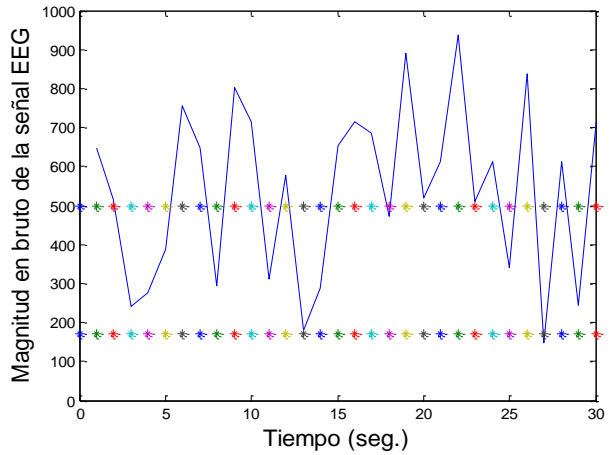
Prueba 6:



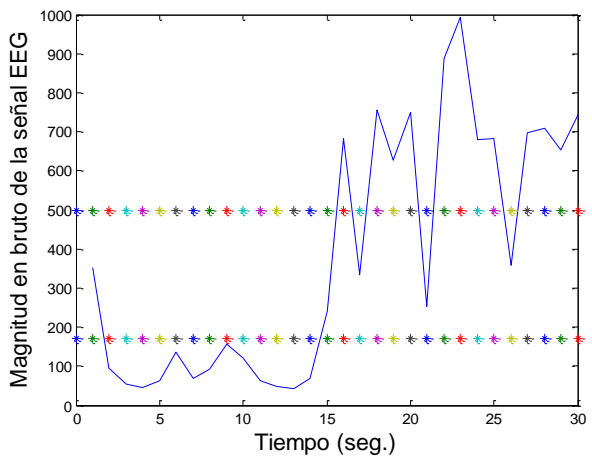
Prueba 7:



Prueba 8:



Prueba 9:



Prueba 10:

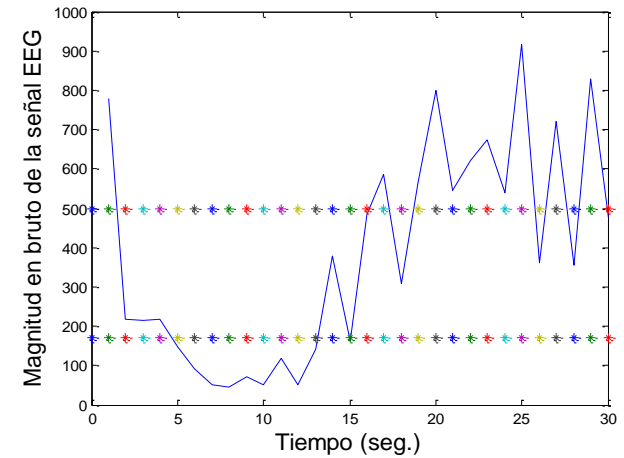





Figura 24. Gráficas de detección del parpadeo voluntario e involuntario.

Como se puede observar en cada muestra se estableció el nivel de acuerdo a la magnitud de la señal EEG, el nivel final del umbral se estableció entre 170 a 500; si la señal se encuentra dentro del nivel de 170 a 500, este es considerado un parpadeo voluntario, por el contrario si está por debajo o supera del nivel de 170 a 500, es considerado un parpadeo involuntario; como se puede observar en la mayoría de las pruebas los parpadeos involuntarios se encuentran por debajo del nivel establecido.

Cada vez que el sistema BCI Mobile detecta un parpadeo válido, la variable “detectaparpadeo” será verdadera y aparecerá un icono () en la parte superior derecha de la pantalla, si se detecta un parpadeo no válido la variable detectaparpadeo será falso y no aparecerá el icono en pantalla (Ver Anexo B del renglón 173 al 182). Para manipular el sistema BCI Mobile; el usuario tiene que realizar un parpadeo voluntario y valido para seleccionar alguna opción, menú o submenú, y poder realizar en él envió de mensajes con éxito.

La función poorq determina si hay buena o mala calidad en la señal EEG; la buena calidad de la señal EEG está dada por PoorQuality cuando su valor o magnitud es cero. Si hay buena calidad aparecerá en la parte superior izquierda de la pantalla  y si no exista una buena calidad de la señal aparecerá otro icono  . Una mala calidad de la señal se puede dar por diversos factores: si el electrodo no hace un buen contacto con la piel del usuario, si el electrodo se encuentra en malas condiciones o sucio y que la diadema MindWave Mobile tenga baja batería o se encuentre apagada.

```
183. void poorq(){
184.   if (datog[1]==0){
185.     TFTscreen.image(buena, 75, 0);
186.     poor=true;
187.   }else{
188.     poor=false;
189.     TFTscreen.image(mala, 75, 0);
190.   }
191. }
```

Pantalla de bienvenida.

La función void bienvenida muestra en pantalla una imagen que representa el inicio del sistema BCI Mobile. Una vez que el GSM accedió a la red se muestra un texto de “Bienvenido” en la pantalla Robot LCD, con una duración de 3 seg.

```
192.void bienvenida(){
```

```
193.   TFTscreen.background(255,255,255);
194.   for (int i = 0; i < 10; i++){
195.       TFTscreen.stroke(0,0,0);
196.       TFTscreen.setTextSize(3);
197.       TFTscreen.text(bienvenido[i], x, y);
198.       x=x+16;
199.       y=y+12;
200.   }
210. }
```

Puntos de temporización.

La función void points; permite dibujar un pixel asignando el valor de las coordenadas “x” y “y”. Estos pixeles se visualizan consecutivamente en la pantalla Robot LCD en un lapso de 2.25 segundos, los cuales representan la temporización para el cambio de los menús, submenús y opciones.

```
211. void points() {
212.     for (xPos=0; xPos<=160;xPos+=10){
213.         TFTscreen.stroke(0,0,255);
214.         TFTscreen.point(xPos, yPos);
215.         TFTscreen.point(xPos, yPos+1);
216.         TFTscreen.point(xPos, yPos-1);
217.         TFTscreen.point(xPos+1, yPos);
218.         TFTscreen.point(xPos-1, yPos);
219.         TFTscreen.point(xPos+1, yPos+1);
220.         TFTscreen.point(xPos+1, yPos-1);
221.         TFTscreen.point(xPos-1, yPos+1);
222.         TFTscreen.point(xPos-1, yPos-1);
223.         delay(200);
224.     }
238. }
```

Menú principal.

En la función void menup, engloba dos Submenús: Submenu1 nombrado como “mensajes básicos” (SMS Básicos) y Submenú2 nombrado como “mensajes de control” (SMS control). Para ingresar a los submenús, el usuario tiene que realizar un parpadeo en un lapso de 3 segundos (de acuerdo a la temporización), siempre y cuando exista una buena calidad de la señal; de lo contrario permanecerá en un tiempo indefinido en el menú principal.

```
239. void menup() {
240.     contop++;
241.     if(contop==1){
242.         TFTscreen.background(255,255,255);
243.         TFTscreen.stroke(0,0,0);
244.         TFTscreen.setTextSize(2);
245.         TFTscreen.text(menupr[0],0, 20);
246.         TFTscreen.setTextSize(3);
247.         TFTscreen.text(menupr[1],0, 65);
248.         points();

```

```
249.         if( detectaparpadeo==true&&poor==true){
250.             detectaparpadeo=false;
251.             poor=false;
252.             contsub1=0;
253.             return submenu1();
254.         }else{
255.             return menu();
256.         }
278.     }
279. }
```

Submenu1.

La función void Submenu1, engloba al submenú de mensajes básicos (SMS Básicos). Este submenú contiene los siguientes mensajes: S.O.S, Quiero comer, Quiero ir al sanitario, Tengo sed, Me duele la cabeza, Me quiero asear. Estos mensajes fueron seleccionados de acuerdo a las necesidades básicas del usuario con discapacidad motora y del habla, el cual se predefinen en el sistema BCI Mobile. Para ingresar al submenú de mensajes básicos, el usuario tiene que realizar un parpadeo en un lapso de 3 segundos; Dentro del submenú1 se muestra en pantalla cada una de las opciones hasta que se detecte un parpadeo en el mensaje deseado por el usuario. Para salir del submenú de mensajes básicos y regresar al menú principal se encuentra la opción salir.

```
280. void submenu1(){
281.     contsub1++;
282.     if (contsub1==1){
283.         TFTscreen.background(255,255,255);
284.         TFTscreen.stroke(0,0,0);
285.         TFTscreen.setTextSize(2);
286.         TFTscreen.text(subtitulo[0],10, 25);
287.         TFTscreen.setTextSize(4);
288.         TFTscreen.text(subml[0],20, 60);
289.         points();
290.         if(detectaparpadeo==true&&poor==true){
291.             detectaparpadeo=false;
292.             poor=false;
293.             opcont=0;
294.             num=1;
295.             opciones();
296.         }else{
297.             submenu1();
298.         }
416.     }
417. }
```

Submenu2.

La función void submenu2, contiene el submenú de mensajes de control (SMS de control); el submenu2 contiene los siguientes mensajes: Enciende la T.V., Apaga la T.V., Sube el

volumen, Baja el volumen y Cambia de canal. Estos mensajes fueron escogidos de acuerdo a la necesidad de entretenimiento de las personas con discapacidad motora y del habla; por lo que son mensajes ya predefinidos para el sistema BCI Mobile. Para ingresar al submenú de mensajes de control el usuario tiene que realizar un parpadeo en un lapso de 3 segundos; Dentro del submenú 2 cada opción aparecerá en pantalla hasta que un parpadeo sea detectado en el mensaje deseado por el usuario. Para salir del submenú de mensajes de control y regresar al menú principal se encuentra la opción salir.

```
418. void submenu2() {
419.     contsub2++;
420.     if (contsub2==1) {
421.         TFTscreen.background(255,255,255);
422.         TFTscreen.stroke(0,0,0);
423.         TFTscreen.setTextSize(2);
424.         TFTscreen.text(subtitulo[1],10, 20);
425.         TFTscreen.setTextSize(3);
426.         TFTscreen.text(subm2[0],0, 65);
427.         points();
428.         if(detectaparpadeo==true&&poor==true) {
429.             poor=false;
430.             detectaparpadeo=false;
431.             opcont=0;
432.             num=7;
433.             opciones();
434.         }else{
435.             submenu2();
436.         }
437.     }
438. }
```

La función void opciones muestra al usuario dos opciones: enviar o cancelar; para poder ingresar a las opciones el usuario previamente tuvo que elegir algún mensaje predefinido del menú de mensajes básicos o de mensajes de control. La opción enviar le muestra al usuario una agenda para escoger el destinatario, al cual se le enviará el mensaje, mientras que la opción cancelar manda al usuario al menú principal del sistema BCI Mobile.

```
537. void opciones() {
538.     opcont++;
539.     if (opcont==1) {
540.         TFTscreen.background(255,255,255);
541.         TFTscreen.stroke(0,0,0);
542.         TFTscreen.setTextSize(3);
543.         TFTscreen.text(opcion[0], 25, 50);
544.         points();
545.         if(detectaparpadeo==true&&poor==true) {
546.             poor=false;
547.             detectaparpadeo=false;
548.             agenda();
549.         }else{
```

```
550.         opciones();
551.     }
552. }else{
553.     if(opcont==2){
554.         TFTscreen.background(255,255,255);
555.         TFTscreen.stroke(0,0,0);
556.         TFTscreen.setTextSize(3);
557.         TFTscreen.text(opcion[1], 5, 50);
558.         points();
559.         if(detectaparpadeo==true&&poor==true){
560.             poor=false;
561.             detectaparpadeo=false;
562.             contop=0;
563.             menup();
564.         }else{
565.             opciones();
566.         }
567.     }else{
568.         opcont=0;
569.         opciones();
570.     }
571. }
572. }
```

La función agenda contiene una lista de contactos predefinidos a los cuales se les podrán enviar los mensajes previamente escogidos. Este código solo tiene predefinidos dos números como posibles receptores de los mensajes. Dentro de la función agenda se encuentra la inicialización del gsm, para el envío del mensaje mediante la sentencia sms.beginSMS(remoteNum) como se muestra en la línea 586, en la variable remoteNum se encuentra almacenado un número telefónico a 10 dígitos, como se muestra en la línea 584. Para envío del mensaje sms se hace uso de sms.print(__) y de sms.endSMS(), como se muestra en la línea 587 y 588, el cual completa el envío del mensaje.

```
573. void agenda(){
574.     contag++;
575.     if (contag==1){
576.         TFTscreen.background(255,255,255);
577.         TFTscreen.stroke(0,0,0);
578.         TFTscreen.setTextSize(3);
579.         TFTscreen.text("contacto\n    1", 5, 25);
580.         TFTscreen.setTextSize(2);
581.         TFTscreen.text("5520908217", 20, 80);
582.         points();
583.         if(detectaparpadeo==true&&poor==true){
584.             poor=false;
585.             detectaparpadeo=false;
586.             char remoteNum[20]="5520908217";
587.             if(num==1){
588.                 sms.beginSMS(remoteNum);
589.                 sms.print(subm1[0]);
590.                 sms.endSMS();
591.                 enviado();
```

```
592.         }else if(num==2){
593.             sms.beginSMS(remoteNum);
594.             sms.print(subml[1]);
595.             sms.endSMS();
596.             enviado();
642.         }else{
643.             num=0;
644.         }
645.     }else{
646.         agenda();
647.     }
738. }else{
739.     contag=0;
740.     agenda();
741. }
742. }
```

Una vez realizado el envío del mensaje la función void enviado muestra un mensaje en la pantalla Robot LCD, notificando al usuario que se realizó el envío del mensaje.

```
745.void enviado(){
746.    TFTscreen.background(255,255,255);
747.    TFTscreen.stroke(0,0,0);
748.    TFTscreen.setTextSize(3);
749.    TFTscreen.text(" sms\n enviado", 0, 35);
750.    delay(1000);
751. }
```

En el Anexo B se muestra el código completo del sistema BCI Mobile

Capítulo IV

5. Pruebas y Resultados.

5.1. Pruebas.

Se llevaron a cabo diversas pruebas para poder acoplar los componentes del Sistema BCI Mobile, las pruebas se llevaron a cabo conforme la división por bloques de la Figura 18.

5.1.1. Prueba 1: Adquisición y transferencia de datos.

Se realizó un código en el software Arduino haciendo uso de una serie de comandos AT para la configuración del módulo Bluesmirf y a su vez lograr una comunicación inalámbrica entre la diadema MindWave y el módulo Bluesmirf. Como se muestra en la Figura 25.

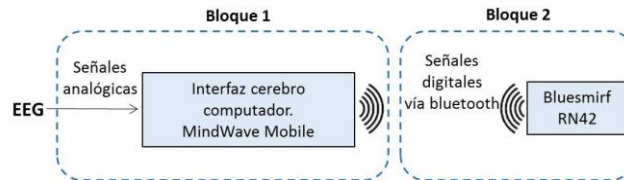


Figura 25. Comunicación inalámbrica de la MindWave Mobile y el Bluesmirf.

Para el emparejamiento entre ambos dispositivos se hace uso del código de programación para la configuración del Bluesmirf mencionado anteriormente en el subtema 4.2.1.; para leer los datos provenientes de la diadema MindWave Mobile, se hace uso de una aplicación de terminal de puerto serie llamado "CoolTerm"; esta aplicación permite visualizar los datos transmitidos del hardware conectado a los puertos serie de un ordenador como es el microcontrolador Arduino.

El Bluesmirf se conectó en la plataforma Arduino ADK para adquirir los datos provenientes de la MindWave Mobile que se transmiten de forma inalámbrica. Como se muestra en la Figura 26.

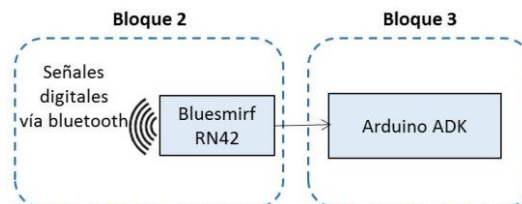


Figura 26. Comunicación serial entre el Bluesmirf y el Arduino ADK.

Para controlar el Módulo Bluesmirf se utiliza una comunicación serial, la cual nos permite transmitir y recibir datos bit a bit. Los datos que llegan a través de la vía Bluetooth los recibe

el pin RX-I del Bluesmif y su vez son transmitidos TX-O hacia el Arduino ADK RX1. Las tramas recibidas en el pin RX-I, nos muestran una serie de valores sin unidades que representan la meditación, la atención y 5 tipos de ondas cerebrales que puede adquirir la diadema MindWave Mobile.

Estas tramas tienen dos estructuras: tramas largas (Ver Figura 22) y tramas cortas (Ver Figura 23), para la lectura de tramas largas y tramas cortas se creó un código Arduino a partir de un código proporcionado por el desarrollador de la diadema MindWave Mobile Neurosky:

5.1.1.1. Código para capturar tramas cortas y tramas largas:

```
752. #define BAUDRATE 57600
753. void setup() {
754.   Serial.begin(BAUDRATE);
755.   Serial1.begin(BAUDRATE);
756.   pinMode(13, OUTPUT);
757.   pinMode(12, OUTPUT);
758. }
759. byte ReadOneByte() {
760.   int ByteRead;
761.   while(!Serial3.available());
762.   ByteRead = Serial3.read();
763.   return ByteRead;
764. }
765. void loop() {
766.   primerSincronia=ReadOneByte();
767.   if(primerSincronia == 170) {
768.     segundaSincronia=ReadOneByte();
769.     if(segundaSincronia == 170) {
770.       payloadLength = ReadOneByte();
771.       longitud=payloadLength;
772.       if(payloadLength > 169)
773.         return;
774.       if(longitud==4){
775.         longitudP=longitud;
776.         generatedChecksumP = 0;
777.         for (int i = 0; i < longitudP; i++) {
778.           payloadDataP[i] = ReadOneByte();
779.           generatedChecksumP += payloadDataP[i];
780.         }
781.         checksumP = ReadOneByte();
782.         generatedChecksumP = 255 - generatedChecksumP;
783.         if (checksumP == generatedChecksumP) {
784.           Serial.print(primerSincronia);
785.           Serial.print(segundaSincronia);
786.           Serial.print(longitud);
787.           for (int i = 0; i < longitudP; i++) {
788.             Serial.print(payloadDataP[i]);
789.           }
790.           Serial.print(checksumP);
791.         }
792.       }
```

```

793.     if (longitud==32){
794.         longitudG=longitud;
795.         generatedChecksumG = 0;
796.         for (int i = 0; i < longitudG; i++) {
797.             payloadDataG[i] = ReadOneByte();
798.             generatedChecksumG += payloadDataG[i];
799.         }
800.         checksumG = ReadOneByte();
801.         generatedChecksumG = 255 - generatedChecksumG;
802.         if (checksumG == generatedChecksumG) {
803.             Serial.print(primerSincronia);
804.             Serial.print(segundaSincronia);
805.             Serial.print(longitud);
806.             for (int i = 0; i < longitudG; i++) {
807.                 Serial.print(payloadDataG[i]);
808.             }
809.             Serial.print(checksumG);
810.         }
811.     }
812. }
813. }
814. }
  
```

En base al código mencionado anteriormente y haciendo uso del programa CoolTerm, fuimos capaces de capturar los datos provenientes de las tramas que proporciona la MindWave Mobile, como se muestra a continuación en la Figura 27; donde la trama corta está indicada en color rojo y la trama larga en color azul.

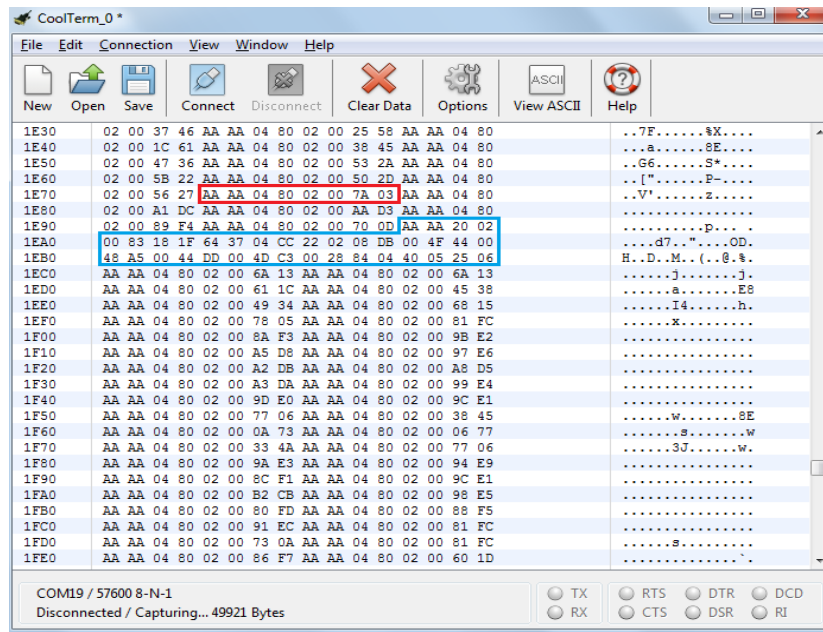


Figura 27. Trama corta y Trama larga en CoolTerm.

Los diferentes datos que se adquieren dependiendo de la trama de la cual provengan, se utilizaron las tramas largas para la detección del parpadeo en el sistema y las tramas cortas nos ayudaron para el uso de la buena calidad de la señal extraída por el electrodo colocado

en la parte frontal polar 1 (Fp1) del usuario de acuerdo Sistema Internacional 10-20 para la colocación de electrodos.

5.1.2. Prueba 2: Manipulación de los datos adquiridos.

Una vez obtenida la comunicación del Bluesmirk y la MindWave Mobile, se volvió a rediseñar el código para la detección del parpadeo a través de la amplitud de las señales EEG. Esto se realizó con el fin de seleccionar opciones predefinidas en un menú; fácil y práctico para la interacción entre el sistema y el usuario.

Para desarrollar el programa del parpadeo se realizaron las siguientes pruebas:

Se utilizó un LED para la visualización de la existencia de un parpadeo voluntario, es decir, si el usuario realiza un parpadeo voluntario inmediatamente se enciende el LED y de forma contraria el LED se mantiene apagado mientras el usuario no realice ningún parpadeo voluntario. El código de programación en Arduino se muestra a continuación:

5.1.1.2. Código para detección del parpadeo voluntario:

```
1.  #define BAUDRATE 57600
2.  volatile int state = LOW;
3.  void setup() {
4.      Serial.begin(BAUDRATE);
5.      Serial1.begin(BAUDRATE);
6.      pinMode(pin, OUTPUT);
7.      attachInterrupt(4, byteabyte, CHANGE);
8.  }
9.  void loop() {
10.     digitalWrite(pin, state);
11. }
12. void byteabyte() {
13.     if(Serial1.available()) {
14.         variable=Serial1.read();
15.         if(variable== 170 && estado==0 && contbyte==0) {
16.             estado=1;
17.             contbyte++;
18.         }else
19.         if(variable== 170 && estado==1&& contbyte==1) {
20.             estado=2;
21.             contbyte++;
22.         }else
23.         if(variable==4 && estado==2 && contbyte==2){
24.             estado=3;
25.             contbyte++;
26.         }else
27.         if(variable==128 && estado==3 && contbyte==3){
28.             dato[0]=variable;
29.             estado=4;
30.             contbyte++;
31.         }else
32.         if(variable==2 && estado==4 && contbyte==4){
33.             dato[1]=variable;
34.             cont++;
35.         }else
36.         if(cont==1 && estado==4 && contbyte==4){
```

```
37.         dato[2]=variable;
38.         varcod1=dato[2];
39.         cont++;
40.     }else
41.     if(cont==2 && estado==4 && contbyte==4){
42.         dato[3]=variable;
43.         varcod2=dato[3];
44.         cont++;
45.     }else
46.     if(cont==3 && estado==4 && contbyte==4){
47.         checksum=variable;
48.         generatedChecksum = dato[0]+dato[1]+dato[2]+dato[3];
49.         generatedChecksum= 255 - generatedChecksum;
50.         if (checksum==generatedChecksum) {
51.             if (muestra<512){
52.                 totalcod=(varcod1<<8)|(varcod2);
53.                 if(totalcod&0xF000){
54.                     totalcod= (((~totalcod)&0xFFF)+1);
55.                 }else{
56.                     totalcod = (totalcod&0xFFF);
57.                 }
58.                 totalcod1 += totalcod;
59.                 muestra++;
60.             }else{
61.                 totalcod2=totalcod1/512;
62.                 parpadeo();
63.                 muestra=0;
64.                 totalcod1=0;
65.             }
66.         }
67.         estado=0;
68.         contbyte=0;
69.         cont=0;
70.     }else{
71.         estado=0;
72.         contbyte=0;
73.         cont=0;
74.     }
75. }
76. }
77. void parpadeo(){
78.     if(170<totalcod2&&totalcod2<500){
79.         digitalWrite(pin,HIGH);
80.         state= !state;
81.     }else
82.     if (totalcod2>500){
83.         digitalWrite(pin,LOW);
84.         state=LOW;
85.     }
86. }
```

Después de realizar este código, se hizo uso del programa Matlab para verificar la amplitud de la señal del parpadeo voluntario e involuntario proveniente de la MindWave Mobile. Esto nos ayudó analizar el rango de la amplitud de un parpadeo voluntario e involuntario (ver página 63 y 64).

5.1.3. Prueba 3: Creación del menú.

Una vez obtenida la comunicación del Bluesmirf y la MindWave Mobile y la detección del parpadeo voluntario, se rediseño el código de programación para agregar la pantalla TFT Robot LCD, donde podemos visualizar un menú de mensajes predefinidos de acuerdo a las necesidades de los usuarios con discapacidad motora y del habla; la programación del menú de opciones se llevó a cabo en el Software Arduino.

Para el diseño del menú se llevó a cabo haciendo uso de un diagrama de flujo (Ver anexo A) y un diagrama de estados (Ver Anexo C), tomando en cuenta las siguientes opciones: Pantalla principal (representada por una imagen), Bienvenida, Mensajes básicos, Mensajes de control, Agenda de contactos, Cancelar mensaje, Enviar mensaje y Salir (para regresar a la pantalla principal del menú). Tomando en cuenta el cambio de opciones en un intervalo de tiempo de 2.25 seg.

La pantalla Robot LCD se conectó al Arduino ADK para visualizar el menú de opciones predefinidas para el usuario con discapacidad motora y del habla. Como se muestra en la Figura 28.

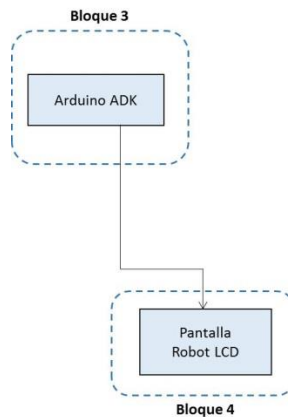


Figura 28. Comunicación entre el Arduino ADK y la pantalla Robot LCD.

Una vez obtenido el diagrama de estado para el menú de opciones, se reprogramo el código para el buen funcionamiento del menú, por lo que en esta prueba se colocó la lectura de las tramas dentro de una interrupción, para que trabaje la plataforma Arduino ADK de forma paralela en dos tareas: la lectura de tramas y mostrar el menú en la pantalla Robot LCD; con la interrupción evita que se trabe el sistema en lapsos cortos de tiempo.

5.1.1.3. Código de programación del menú de opciones y la interrupción:

```
87. #define BAUDRATE 57600
88. TFT TFTscreen = TFT(cs, dc, rst);
89. volatile boolean detectaparpadeo = false;
90. void setup() {
```

```
91.   TFTscreen.begin();
92.   TFTscreen.background(255,255,255);
93.   Serial.begin(BAUDRATE);
94.   Serial1.begin(BAUDRATE);
95.   attachInterrupt(4,byteabyte,CHANGE);
96.   }
97.   void loop(){
98.       menup();
99.   }
100. void byteabyte(){
101.   if(Serial1.available()){
102.     variable=Serial1.read();
103.     if(variable== 170 && estado==0 && contbyte==0) {
104.       estado=1;
105.       contbyte++;
106.     }else
107.     if(variable== 170 && estado==1&& contbyte==1) {
108.       estado=2;
109.       contbyte++;
110.     }else
111.       if(variable==4 && estado==2 && contbyte==2){
112.         estado=3;
113.         contbyte++;
114.       }else
115.       if(variable==128 && estado==3 && contbyte==3){
116.         dato[0]=variable;
117.         estado=4;
118.         contbyte++;
119.       }else
120.       if(variable==2 && estado==4 && contbyte==4){
121.         dato[1]=variable;
122.         cont++;
123.       }else
124.       if(cont==1 && estado==4 && contbyte==4){
125.         dato[2]=variable;
126.         varcod1=dato[2];
127.         cont++;
128.       }else
129.       if(cont==2 && estado==4 && contbyte==4){
130.         dato[3]=variable;
131.         varcod2=dato[3];
132.         cont++;
133.       }else
134.       if(cont==3 && estado==4 && contbyte==4){
135.         checksum=variable;
136.         generatedChecksum = dato[0]+dato[1]+dato[2]+dato[3];
137.         generatedChecksum= 255 - generatedChecksum;
138.         if (checksum==generatedChecksum) {
139.           if (muestra<512){
140.             totalcod=(varcod1<<8) | (varcod2);
141.             if(totalcod&0xF000){
142.               totalcod= (((~totalcod)&0xFFF)+1);
143.             }else{
144.               totalcod = (totalcod&0xFFF);
145.             }
146.             totalcod1 += totalcod;
```

```
147.     muestra++;
148.     }else{
149.     totalcod2=totalcod1/512;
150.     muestra=0;
151.     totalcod1=0;
152.     parpadeo();
153.     }
154.     }
155.     estado=0;
156.     contbyte=0;
157.     cont=0;
158.     }else{
159.     estado=0;
160.     contbyte=0;
161.     cont=0;
162.     }
163.     }
164. }
165. void parpadeo(){
166.     if(170<totalcod2&&totalcod2<500){
167.         detectaparpadeo=true;
168.     }else
169.     if (totalcod2>500){
170.         detectaparpadeo=false;
171.     }
172. }
173. void menu(){
174.     contop++;
175.     if(contop==1){
176.         TFTscreen.text(menupr[1],0, 45);
177.         points();
178.         if( detectaparpadeo==true){
179.             detectaparpadeo=false;
180.             submenu1();
181.             contop=0;
182.         }else{
183.             return menu();
184.         }
185.     }
186.     if(contop==2){
187.         TFTscreen.text(menupr[2],0, 45);
188.         points();
189.         contop=0;
190.     if(detectaparpadeo==true){
191.         detectaparpadeo=false;
192.         submenu2();
193.         contop=0;
194.     }else{
195.         return menu();
196.     }
197. }
198. }
199. void submenu1(){
200.     contsub1++;
201.     if (contsub1==1){
202.         TFTscreen.text(subm1[0],20, 55);
```

```

203.     points();
204.     }
205.     if(contsub1==2){
206.         TFTscreen.text(subm1[1], 20, 45);
207.         points();
208.         contsub1=0;
209.         if (detectaparpadeo==true){
210.             detectaparpadeo=false;
211.             return menup();
212.         }
213.     }
214. }
    
```

5.1.4. Prueba 4: Envío de mensajes.

Finalmente se desarrolla el programa para enviar un mensaje, tomando en cuenta una agenda de destinatarios predefinidos. Se conectó el shield Arduino GSM y la plataforma Arduino ADK (Ver Figura 29) para enviar un mensaje haciendo uso de biblioteca gsm.h mencionadas anteriormente.

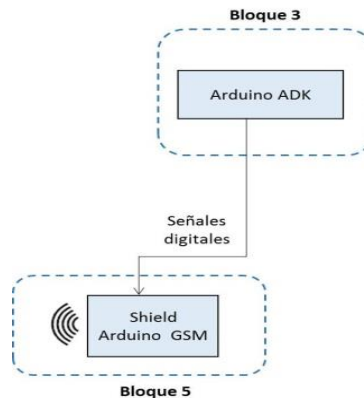


Figura 29. Comunicación del Arduino ADK y el Shield Arduino GSM.

Se ejecutó el programa SendSMS que ofrece Arduino (ver en el software Arduino –Archivo->Ejemplos->GSM-> SendSMS).

```

1. #include <GSM.h>
2. #define PINNUMBER ""
3. GSM gsmAccess;
4. GSM_SMS sms;
5. void setup()
6. {
7.     Serial.begin(9600);
8.     while (!Serial) {
9.     }
10.     Serial.println("SMS Messages Sender");
11.     boolean notConnected = true;
12.     while (notConnected)
13.     {
14.         if (gsmAccess.begin(PINNUMBER) == GSM_READY)
    
```

```
15.         notConnected = false;
16.         else
17.         {
18.             delay(1000);
19.         }
20.     }
21. }
22. void loop()
23. {
24.     Serial.print("Enter a mobile number: ");
25.     char remoteNum[20];
26.     readSerial(remoteNum);
27.     Serial.println(remoteNum);
28.     Serial.print("Now, enter SMS content: ");
29.     char txtMsg[200];
30.     readSerial(txtMsg);
31.     Serial.println("SENDING");
32.     Serial.println();
33.     Serial.println("Message:");
34.     Serial.println(txtMsg);
35.     sms.beginSMS(remoteNum);
36.     sms.print(txtMsg);
37.     sms.endSMS();
38.     Serial.println("\nCOMPLETE!\n");
39. }
40. int readSerial(char result[])
41. {
42.     int i = 0;
43.     while (1)
44.     {
45.         while (Serial.available() > 0)
46.         {
47.             char inChar = Serial.read();
48.             if (inChar == '\n')
49.             {
50.                 result[i] = '\0';
51.                 Serial.flush();
52.                 return 0;
53.             }
54.             if (inChar != '\r'){
55.                 result[i] =inChar;
56.                 i++;
57.             }
58.         }
59.     }
60. }
```

Una vez que se verifico que el código SendSMS realiza correctamente el envío del mensaje, se anexa al código del sistema BCI Mobile, obteniendo el código completo para el buen funcionamiento del Sistema BCI Mobile (Ver Anexo B).

5.2. Resultados.

Con el fin de lograr los objetivos planteados al inicio de esta tesis, se llevaron a cabo diversas pruebas en personas con discapacidad motora y del habla en México. Se realizó una primera prueba con la compañera Saavedra Valdespino Jessica Guadalupe de la Universidad Autónoma de la Ciudad de México, plantel Cuauhtépec con carrera Promoción de la Salud a la edad de 26 años; Jessica cuenta con una discapacidad llamada “Parálisis cerebral con trastornos del movimiento”. Cabe mencionar que la discapacidad de parálisis cerebral con trastornos del movimiento, se refiere a la disminución, deficiencia o pérdida parcial o total de la capacidad del movimiento, acompañada generalmente de pérdida de la sensibilidad, debido a trastornos motores o musculares. [96]

Jessica hizo uso del sistema BCI Mobile, de manera satisfactoria, se observó que el sistema es funcional y viable para personas que cuentan con la discapacidad de parálisis cerebral con trastornos del movimiento, ella realizó una crítica constructiva para la mejora del sistema BCI Mobile, debido a que el sistema llegó a tener un error general del sistema, por lo que se trabó en ciertos lapsos de tiempo seguido de un reinicio inesperado, sin embargo ella señaló que el sistema BCI Mobile es efectivo y cumple con su funcionamiento de comunicación, y que sería de gran utilidad para aquellas personas que cuentan con una discapacidad de mayor nivel a la que ella padece.



Figura 30. Primera prueba del Sistema BCI Mobile con Jessica Guadalupe.

De la primera prueba se observó que la magnitud de su señal del parpadeo de Jessica varía del 199 al 354 y estos datos están dentro del rango de 170 a 500 de parpadeos voluntarios, por lo que se verificó que este rango es efectivo para la detección del parpadeo voluntario.

Para llevar a cabo una segunda prueba se tomaron en cuenta las observaciones y críticas constructivas de la primera prueba, llevándonos a una mejora del sistema BCI Mobile disminuyendo el reinicio del sistema de forma inesperada.

Se revisó el sistema BCI Mobile para verificar el error general del sistema, durante el transcurso se encontró el error y fue modificado el código para eliminar el reinicio del sistema de forma inesperada como se puede observar en el Anexo B. La segunda prueba se llevó a cabo de nuevo con la compañera Saavedra Valdespino Jessica Guadalupe utilizando el sistema BCI Mobile, Jessica realizó él envió de dos mensajes; uno del menú de mensajes básicos y un mensaje de entretenimiento. Para comprobar que el sistema funciona, los mensajes se enviaron a su celular.



Figura 31. Segunda prueba del sistema BCI Mobile con Jessica Guadalupe.

Jessica señala que el sistema BCI Mobile mejoró ya que nunca se reinició durante la segunda prueba y ya no se trababa, por lo que el sistema cumplió con su función y objetivo de manera satisfactoria, por lo tanto se puede decir que la segunda prueba fue todo un éxito.



Figura 32. Jessica Guadalupe haciendo uso del Sistema BCI Mobile.

6. Conclusiones.

6.1. Conclusiones de Graciela Hernández Ramírez.

El objetivo general de este trabajo recepcional fue diseñar y desarrollar un sistema híbrido, que permita satisfacer la necesidad de comunicación en la población con discapacidad motora y de habla hacia otros usuarios en México logrando fomentar su independencia al comunicarse y mejorar su calidad de vida; por medio de la telefonía celular y otras tecnologías.

Esta idea surgió pensando en las necesidades de las personas con discapacidad motora y del habla, mostrada en el capítulo1, en este caso al comunicarse por medio de telefonía celular hacia otros usuarios o sistemas con el fin de fomentar su independencia. La discapacidad motora presenta alteraciones musculares, impactando la movilidad de la persona y la discapacidad del habla presenta problemas para comunicarse con otros usuarios.

Para lograr el objetivo, se inició con una hipótesis de realizar el envío de un mensaje por medio de la mente, basándonos a esta hipótesis se parte de una investigación teórica de las neurociencias para entender el funcionamiento del cerebro humano. Una vez obtenida esta información, nos preguntamos ¿Es posible leer la mente humana?, Comenzamos una profunda revisión histórica y tecnológica sobre “leer la mente” sin embargo, encontramos que existen tecnologías capaz de traducir nuestras intenciones en acciones con el mundo real. La tecnología se describe como Brain- Computer Interface (BCI); cuyo funcionamiento básico es detectar y analizar las ondas emitidas por el cerebro humano, estas ondas son procesadas y convertidas en órdenes o acciones determinadas sobre algún dispositivo externo, es importante mencionar que las BCI’s no leen la mente, es decir, no extraen los pensamientos de las personas.

La meta principal fue realizar una comunicación inalámbrica entre la BCI y un microcontrolador para realizar el envío de un mensaje por medio de un parpadeo; se realizó una investigación de las siguientes tecnologías: BCI (Interfaz Cerebro - Computador), Redes Inalámbricas de Área Personal (WPAN), el Sistema Global para Comunicaciones Móviles (GSM), Microcontroladores y Pantallas TFT (Thin-Film Transistor). Cada una de las tecnologías se describen el en Capitulo 2, el cual explica sus características principales tales como: Historia, Definición, Composición, Funcionamiento, existencia en el mercado y servicios que brindan las tecnologías (principalmente GSM).

Una vez seleccionadas las tecnologías, en el Capítulo 3 se presentaron los dispositivos que conformaron el sistema de comunicación (Sistema BCI Mobile), los criterios para elegir cada uno de los dispositivos y la cotización general del sistema. Así mismo se explicó el análisis del Software y Hardware, en el cual se diseñó y se construyó para lograr el funcionamiento del sistema. Se realizaron pruebas para integrar los elementos que componen el sistema de comunicación: en primer lugar se realizó una comunicación inalámbrica entre la diadema Mind Wave Mobile y el Arduino ADK, por medio de un módulo bluetooth (Bluesmirf), para obtener la comunicación inalámbrica se configuró el Bluesmirf. Una vez logrado esta configuración en seguida logramos una comunicación inalámbrica entre ambos dispositivos. Después obtuvimos datos de las señales EEG. Estas señales vienen del electrodo de la diadema que se coloca en la parte frontal del usuario. Los datos provenientes de la diadema se visualizaron en el software CoolTerm, al principio no sabíamos que interpretaban estos datos, sin embargo, se realizó una investigación profunda de la diadema Mind Wave Mobile y concluimos que los datos que envía la diadema vienen en forma de dos tipos de trama: trama pequeña y trama larga.

Se hicieron varias pruebas para determinar el significado de cada uno de los datos que contienen las tramas largas y las tramas cortas. Estos datos obtenidos se le dan el nombre de Raw Wave Value. Después de numerosas pruebas, se concluye que la trama corta nos proporciona la magnitud total de las señales EEG con el código 80_{HEX}; por otro lado la calidad de la señal EEG (poorQuality) se encuentra dada por la trama larga con el código 02_{HEX}, es decir, si presenta una trama larga nos indica que hay buena calidad de la señal EEG y por el contrario la trama corta nos indica el parpadeo. La segunda prueba se llevó a cabo un código de programación para ejecutarlo en el Arduino ADK, el cual se encargó de manipular y procesar los datos provenientes de la diadema, con el fin de identificar la presencia de un parpadeo voluntario e involuntario; se hicieron varias pruebas: la primera prueba fue realizar un código de programación para depurar los datos de la señal EEG referentes al parpadeo, el cual se concluyó que las señales beta, alfa, theta y delta están relacionadas de alguna manera con el parpadeo. Estas señales se presentaron en el capítulo 1. Para llevar a cabo el código de programación se implementó un led; esto es para visualizar la presencia de un parpadeo, es decir, si existe un parpadeo voluntario encendía el Led, de lo contrario el Led se mantenía apagado. Como resultado obtuvimos que cualquier movimiento ocular encendía el LED; La segunda prueba fue modificar el código de programación para obtener los valores del parpadeo voluntario e involuntario, se realizaron varias pruebas y se visualizaron en el software CoolTerm, el cual se observó que

la existencia de un parpadeo voluntario su magnitud en bruto de la señal EEG era mayor a comparación de la magnitud del parpadeo involuntario. A sí mismo se graficaron las muestras con respecto al tiempo en el software Matlab. Se realizaron 10 pruebas para determinar el umbral del parpadeo voluntario y del parpadeo involuntario. Cada prueba se realizó en un lapso de 30 segundos donde los primeros 15 segundos aproximadamente de cada prueba, se realizaron parpadeos involuntarios y después de los 15 segundos se comenzaron a realizar parpadeos voluntarios; esto es para comparar la magnitud en bruto de la señal y realizar un umbral para ajustar al nivel deseado. El primer umbral fue de 170 y el segundo umbral fue de 500, por lo tanto si la señal se encontraba dentro del nivel entre 170 a 500, se consideró un parpadeo voluntario, por el contrario si estaba por debajo del nivel de 170 a 500, se consideró un parpadeo involuntario. Cabe mencionar que 1 de cada 10 pruebas se presentaron casi en la misma amplitud en los dos tipos de parpadeo, esto se debe a factores como: el estado de ánimo del usuario y la distracción. Estos factores de alguna manera afecta a las señales EEG porque aumentan o disminuyen considerablemente su amplitud.

Una vez ajustado el umbral adecuado para la mayoría de las señales EEG, se diseñó un diagrama de flujo para el funcionamiento de un menú de opciones y se llevó a cabo el código de programación, por lo que se agregó en la programación de la segunda prueba. El diagrama se mostró en el anexo A, el cual describe el diseño y la implementación del menú. El diseño se realizó de manera didáctica, eficiente, enfocada a las necesidades fisiológicas (mensajes predefinidos) y de fácil comprensión, para seleccionar menús, submenús y opciones que desee enviar el usuario con discapacidad motora y del habla.

En segundo lugar se conectó la pantalla TFT en el Arduino ADK, el cual logramos seleccionar un mensaje predefinido a través de un parpadeo voluntario, esto fue posible realizando suficientes pruebas.

Finalmente se desarrolló el programa para enviar un mensaje, tomando en cuenta una agenda de destinatarios predefinidos. Se conectó el shield GSM y la plataforma Arduino ADK. Por lo tanto se integraron todos los componentes físicos del sistema BCI Mobile.

En el capítulo 4 se mostraron las pruebas que se llevaron a cabo con la compañera Jessica Guadalupe Saavedra Valdespino que cuenta con la discapacidad de parálisis cerebral con trastornos del movimiento. Ella utilizó el sistema BCI Mobile durante las pruebas, el cual se demostró que en efecto el parpadeo voluntario e involuntario se encuentra dentro del umbral que se obtuvo durante las muestras obtenidas y descritas en el Capítulo 3. De la misma forma se observó que el sistema es funcional y viable para personas que cuentan con la

discapacidad de parálisis cerebral con trastornos del movimiento. Es importante mencionar que los posibles usuarios pueden hacer uso del sistema BCI Mobile si la discapacidad es permanente o temporal (algún accidente).

En el contexto de este trabajo recepcional fue la propuesta de realizar el envío de un mensaje predefinido por medio del parpadeo, sin embargo, en el transcurso de la investigación y desarrollo de esta tesis surgieron propuestas a futuro como realizar llamadas telefónicas y el envío de un correo electrónico.

6.2. Conclusiones de Ruth Valencia Real.

Hoy en día existen millones de usuarios de telefonía celular móvil a lo largo de México, sin duda la razón ha sido la evolución tecnología de los sistemas de telefonía al paso de los años, convirtiéndose en una tecnología de fácil acceso para aquellas personas que requieran comunicarse con otras personas o sistemas.

Una red de telefonía celular que hoy en día se ha convertido en una de la redes más utilizadas ha sido el Sistema Global de Comunicaciones móviles (GSM), gran parte del territorio mexicano accede de manera cotidiana a la red GSM para llevar a cabo la realización de llamadas o envío de mensajes cortos (SMS), por lo que se ha convertido en un tecnología primordial para la comunicación entre usuarios o sistemas.

El uso de la red GSM trae consigo diversas ventajas, por utilizar una banda de frecuencia libre, nos es posible acceder a ella contando con algún aparato tecnológico como es el teléfono móvil, mejor conocido como teléfono celular, el cual cuenta con una tarjeta SIM que identifica al usuario y le brinda el acceso a la red a través de algún proveedor de la red GSM.

Los teléfonos móviles o teléfono celulares nos brindan diversas ventajas, permite que el dispositivo sea portátil al hacer uso de una tarjeta SIM, cuenta con una antena para que el envío de información sea a través de señales y al hacer uso de una fuente de energía de baja potencia, se convierte en un medio de comunicación inalámbrico personal.

Podemos dejar en claro que la tecnología hoy en día puede llegar a tomar un papel muy relevante en la vida cotidiana de las personas; el uso de la red telefónica GSM junto con otro conjunto de tecnologías crean un sistema, el cual aporta calidad de vida, independencia y se convierte en un apoyo cotidiano en aquellas personas que necesiten hacer uso de telefonía celular para comunicarse.

Actualmente el acceso al uso de algún teléfono celular se encuentra restringido para aquellas personas que cuentan con alguna discapacidad, ya que no les es fácil manipularlo

si la persona por ejemplo, no cuenta con movilidad motora; por lo que se puede considerar que los dispositivos que hacen uso de la red de telefonía celular GSM, no están adaptados para estos usuario. Con el objetivo de apoyar a nuestra sociedad y en específico a aquellas personas que cuenten con alguna discapacidad motora y del habla en México, y que a su vez cuenten con la necesidad de comunicarse con otros usuarios o sistemas, en este trabajo de tesis se implementa un sistema que hace uso principalmente de la red GSM para llevar a cabo la comunicación, el cual está integrado por diversas tecnologías que en su conjunto proporcionan un medio de comunicación de fácil manipulación, a través de señales electroencefalografías enfocadas al parpadeo.

En los primeros capítulos de esta tesis se dedicó a la investigación de temas relevantes para llevar al cabo la realización del sistema de comunicación nombrado "BCI Mobile". Se analizó información de conceptos y dispositivos fundamentales en los cuales se basa este sistema para realizar el envío de mensajes SMS, manipulándolo únicamente con el parpadeo del usuario. Se utilizó una metodología experimental ya que este proyecto y trabajo partió de la idea de implementar un sistema que se pudiera manipular con la mente para llevar a cabo una comunicación, sin embargo al ser la mente algo no manipulable, nos dimos cuenta que podemos extraer señales provenientes del cerebro, llamadas señales electroencefalograficas (EEG). Estas señales pueden obtenidas a través de electrodos para ser procesadas y finalmente manipular con ellas dispositivos externos.

Al paso de la investigación se buscó de alguna manera relacionar las señales electroencefalograficas para facilitar la manipulación de un teléfono celular, por lo se escogió el parpadeo del ser humano para poder manipular el dispositivo, ya que esta acción puede ser voluntaria y que es controlada por cada persona, es importante mencionar que la mayoría de las personas que contamos con movilidad en los parpadeos, incluyendo a aquellas personas que por algún motivo cuentan con discapacidad motora y del habla. Por lo que nos enfocamos al estudio del parpadeo a través de la señales electroencefalograficas para la manipulación del sistema BCI Mobile.

Es importante mencionar que el parpadeo se localizó en los diferentes tipos de señales EEG a distintas magnitudes, pero siempre presentes.

Para realizar la lectura de las señales EEG se implementó el uso de una interfaz cerebro computador (BCI), con ayuda de esta interfaz se obtuvieron valores que nos indican la intensidad de las señales EEG, a través de un método de prueba y error, y haciendo uso de software se estableció un intervalo de 170 a 500 para indicar un parpadeo validó voluntario. A través de un parpadeo voluntario fue posible manipular el sistema BCI Mobile

para llevar a cabo él envió de mensajes predefinidos, mostrados a través de una pantalla TFT.

El trabajo de tesis cumplió con el objetivo principal, el cual es, satisfacer la necesidad de comunicación de personas que cuentan con alguna discapacidad motora y del habla en México; por lo que se logró la construcción de un sistema de comunicación basado en el parpadeo del usuario, haciendo uso de una BCI y de otros dispositivos como una interfaz Bluetooth, un microcontrolador, un shield GSM y una pantalla TFT, los cuales se fusionan creando un sistema en el que el usuario con discapacidad motora y del habla pueda decidir el mensaje a comunicar y escoger al destinatario haciendo solo uso del parpadeo, recobrando así su independencia para comunicarse con otros usuarios o sistemas. Como cada usuario es diferente, por tanto sus actividades cerebrales son también distintas, por lo que requiere realizar un entrenamiento previo para hacer uso del sistema y poderse así acoplarse al mismo.

Para que el sistema cuente con un buen funcionamiento se necesita tener una buena calidad de las señales cerebrales al momento de su adquisición, por lo que se debe contar con una buena fijación del electrodo, así como tener una piel limpia libre de impurezas, para que no haya interferencia en la señal y la lectura se lleve a cabo de manera correcta.

Para poder controlar el sistema por medio del parpadeo fue necesario crear un código de programación, este utiliza la información extraída y adquirida por la BCI. La programación se llevó a cabo en un software llamado Arduino; realizada por etapas, hasta llegar al código de programación final. Como se mencionó anteriormente el sistema BCI Mobile se dividió en bloques para finalmente poder realizar el sistema de comunicación.

Finalmente como el sistema BCI Mobile se pensó y creo para personas que cuentan con alguna discapacidad motora y del habla, por consiguiente es didáctico y de fácil uso para el usuario. Al ya contar con el sistema BCI Mobile se realizaron pruebas con la compañera Jessica Guadalupe quedando satisfecha con el proyecto y trabajo realizado. Podemos decir que este trabajo se terminó de manera satisfactoria tanto en términos teóricos como prácticos.

Podemos dejar en claro que la tecnología hoy en día puede llegar a tomar un papel muy relevante en la vida cotidiana de las personas, en nuestro caso el uso de la red telefónica GSM junto con otro conjunto de tecnologías crean un sistema, el cual aporta calidad de vida, independencia y se convierte en un apoyo cotidiano en aquellas personas que necesiten hacer uso de telefonía celular y cuenten con alguna discapacidad motora y del habla.

Referencias.

- [1]. De la O Chávez J. Ramón, "Interfaz Cerebro – Computadora para el Control de un Cursor Basada en Ondas Cerebrales," M.S. thesis, Dept. Sci. Comput., UAM, Azcapotzalco, D.F., Azcapotzalco, [Online]. Available: http://newton.azc.uam.mx/mcc/01_esp/11_tesis/tesis/terminada/071112_delao_chavez_job_ramon.pdf
- [2]. L. Janne, et al, "EGG-based Brain Computer Interfaces for tetraplegics". Computational Intelligence and Neuroscience, volume 2007, article ID 23864, 11 pages, 2007. Online, Available: http://becs.aalto.fi/mbl/pdf/kauhanen_et_al_2007.pdf
- [3]. G. L. Miguel Ángel, "Interfaz BCI de altas prestaciones basadas en la detección y procesamiento de la actividad cerebral", Ph.D. thesis, Departamento de Arquitectura y Tecnología de Computadores, Universidad de Granada, Granada, 2009, [Online]. Available: <http://hera.ugr.es/tesisugr/17894001.pdf>
- [4]. ACCEMOVIL, Diatel y Elea Soluciones. (2012, Marzo). [Online] Available: http://www.imsero.es/InterPresent2/groups/imsero/documents/binario/010_28idi.pdf
- [5]. Vargas Escalona Diana y López Román Jesús, "Aplicación ICC: silla de ruedas para cuadripléjicos controlada con señales electroencefalográficas del movimiento voluntario de los parpados", thesis, UPIITA, Instituto Politécnico Nacional, México, 2006.
- [6]. INEGI, Discapacidad en México, 2010 [Online]. Available: <http://cuentame.inegi.org.mx/poblacion/discapacidad.aspx>
- [7]. INEGI, Personas con discapacidad en México: una visión censal [Online]. Available: http://www.inegi.org.mx/prod_serv/contenidos/espanol/bvinegi/productos/censos/poblacion/2010/discapacidad/702825051785.pdf
- [8]. INEGI, Perfil sociodemográfico Estado Unidos Mexicanos: Censo de población y vivienda 2010, [Online]. Available: http://www.inegi.org.mx/prod_serv/contenidos/espanol/bvinegi/productos/censos/poblacion/2010/perfil_socio/uem/702825047610_1.pdf
- [9]. G.C. José, Neurociencia cognitiva y educación, 1^{ra}. Ed., Fache:Perú, 2004, ch.2, pp. 31-32-33-36-37.
- [10]. S.S.Richard, Neuroanatomía clínica, 5^{ta} Ed., panamericana: España, 2003, ch.1-7, pp. 9-248.
- [11]. N.B. Rafael, "Electroencefalografía", Universidad de Alcalá, Instrumentación Biomédica, Ing. Electron, [Online]. Available: <http://www.bioingenieria.edu.ar/academica/catedras/bioingenieria2/archivos/apuntes/tema%205%20-%20electroencefalografia.pdf>

- [12]. British Neuroscience Association (BNA), "NEUROCIENCIAS: LA CIENCIA DEL CEREBRO", BNA, The Sherrington Buildings, Liverpool, UK, 2003, pp. 1
- [13]. G.S. Eveli, et al., Terapia Asistida por Neurofeedback: Aplicaciones Reales, Dept. de Psicobiología, Facultad de Psic, Universidad de Valencia, [Online]. Available: <http://www.logopediapsicologia.com/wp-content/uploads/terapia-asistida-por-neurofeedback.pdf>
- [14]. G. P. Miguel Ángel, et al., Aproximaciones al estudio de la neurociencia del comportamiento, Universidad Autónoma de Guerrero, 1^{ra}Ed., México, 2010, ISBM: 9789707649118, pp. 373-375-376-378-381.
- [15]. N.J. Luis, Breve introducción a la electroencefalografía cuantitativa, División de Ciencias de la Salud, Universidad de Quintana Roo, UQR, [Online]. Available: <http://dcs.uqroo.mx/electroencefalografia>
- [16]. B.G. Guardado, Lateralidad cerebral y zurdería: desarrollo y neuro-rehabilitación, E.U.A., 2013, Ch.5, pp. 174-175
- [17]. D.J. Terry, Cardiopulmonary Anatomy & Physiology: Essentials of Respiratory Care, 6^{ta} Ed., Cengage Learning: E.U.A, 2012, Ch.17, pp. 525.
- [18]. T. G. Jaime, et al., Neurología, 2da. Ed., Manual moderno: Bogotá, 2010
- [19]. F. Imanol, "Interfaces de control a través de la actividad cerebral", Cátedra Telefónica de la Universidad de Deusto, Diciembre, 2011, Facultad de Ing., DEUSTO, [Online]. Available: <http://www.rcysostenibilidad.telefonica.com/blogs/documentoscatedras/files/2012/02/CuadernoRed2011-ActividadCerebral.pdf>
- [20]. Md. I. Arafat, Brain-Computer Interface: Past, Present & Future, Dept. of Comput. Sci. & Eng., International Islamic University Chittagong (IIUC), Chittagong, Bangladesh [Online]. Available: http://www.academia.edu/1365518/BrainComputer_Interface_Past_Present_and_Future.
- [21]. NeuroSky, (2015, Junio 19), Evolución BCI, [Online]. Available: <http://neurosky.com/2015/06/what-is-bci-and-how-did-it-evolve/>
- [22]. G. R. Esteban José, "Implementación de un interfaz cerebro computador utilizando hardware y software libres", Universidad de Costa Rica, Facultad de Ingeniería, Escuela de Ingeniería Eléctrica, Diciembre del 2011. [Online]. Available: http://eie.ucr.ac.cr/uploads/file/proybach/pb2011/pb2011_031.pdf
- [23]. Headset, EPOC/EPOC+, Emotiv [Online]. Available: <https://emotiv.com/store/>
- [24]. Headset, Insight, Emotiv [Online]. Available: <https://emotiv.com/insight.php>
- [25]. MindWave, Neurosky [Online]. Available: <http://store.neurosky.com/products/mindwave-1>
- [26]. BCI Muse, InteraXon [Online]. Available: <http://store.choosemuse.com/products/muse>
- [27]. OpenBCI, OpenBCI project [Online]. Available: <https://openbci.myshopify.com/collections/frontpage/products/openbci-16-channel-r-d-kit>
- [28]. Focus band, iFocusBand [Online]. Available: <http://www.ifocusband.com/#!shop--cart/chiq>

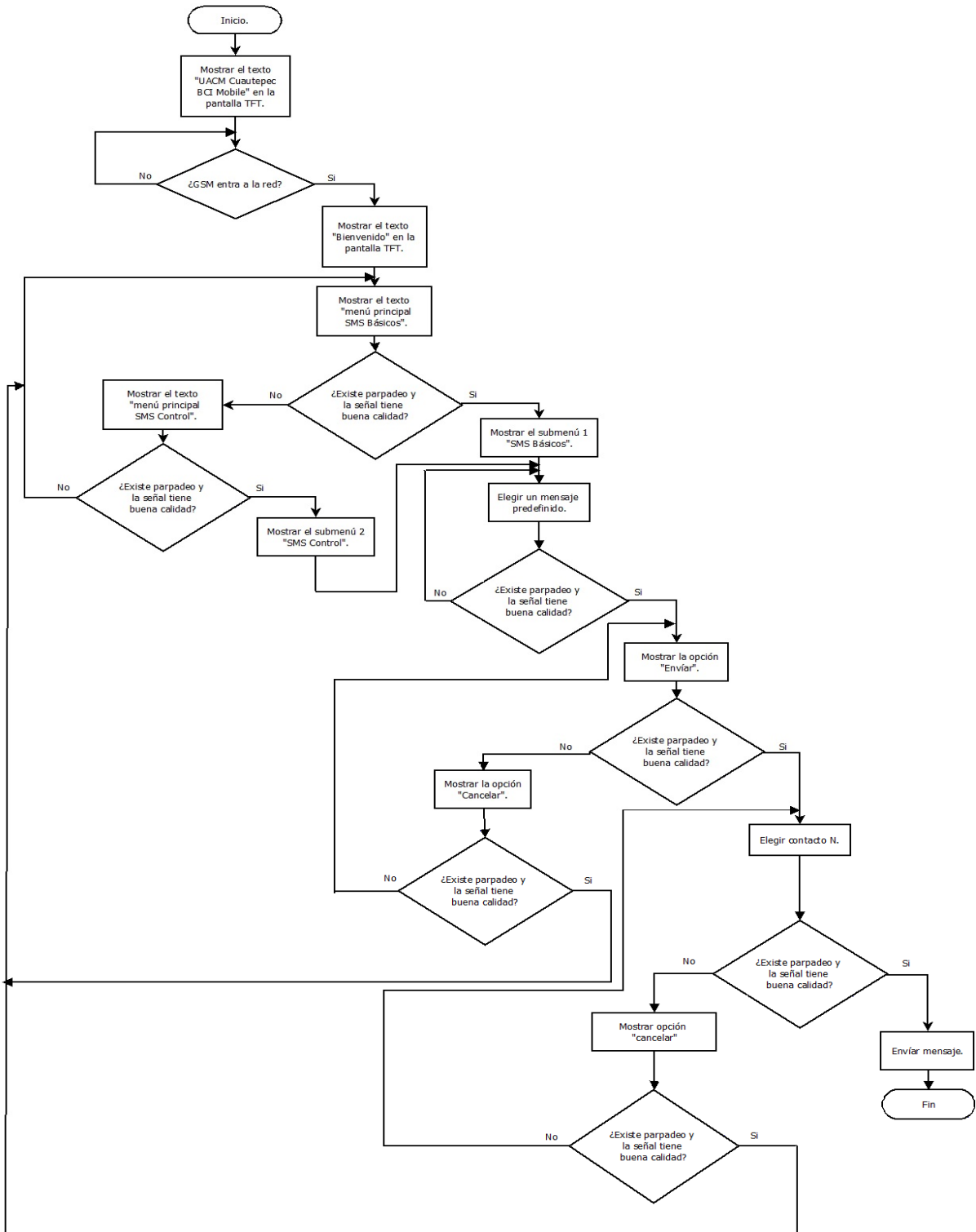
- [29]. Andreu Joaquin, "Servicios de red", Informática y Comunicaciones, Madrid, España, Editorial Editex, S. A., 2010, Unidad 8, pp.212-214.
- [30]. E. B. María Carmen, "Servicios avanzados de telecomunicaciones", 1 ra. ed. , Madrid, España, Ediciones Díaz de Santos, S. A., 2003, pp. 144-146-698-700.
- [31]. R. T. María del Carmen, et al., "Redes Locales", 2 da. ed. Madrid, España, Ediciones Paraninfo, SA. , 2014, ch. 3, sec. 3.6, pp. 54–59
- [32]. The Institute of Electrical and Electronics Engineers, Inc., "IEEE 802.15 Working Group for Wireless Personal Area Networks", IEEE 802.15, 2001 Online. Available: <http://www.ieee802.org/15/>.
- [33]. Bluetooth SIG, Inc, SSP, Bluetooth, 2015. Online. Available: <https://www.bluetooth.com/what-is-bluetooth-technology/bluetooth>
- [34]. H.M.José Manuel, et al., Sistemas de telefonía, 5^{ta} Ed., Paraninfo:España, 2006, pp. 179.
- [35]. C.G.Carlos, et al., Computadores para bases de datos, Ediciones Paraninfo, S.A., pp. 28.
- [36]. Bluetooth SIG, Inc," Serial Port Profile", Bluetooth, 2015. Online. Available: <https://developer.bluetooth.org/TechnologyOverview/Pages/SPP.aspx>
- [37]. Bluetooth SIG, Inc," Advanced Audio Distribution Profile", Bluetooth, 2015. Online. Available: <https://developer.bluetooth.org/TechnologyOverview/Pages/A2DP.aspx>
- [38]. Bluetooth SIG, Inc," Audio/Video Remote Control Profile", Bluetooth, 2015. Online. Available: <https://developer.bluetooth.org/TechnologyOverview/Pages/AVRCP.aspx>
- [39]. Bluetooth SIG, Inc," Headset Profile", Bluetooth, 2015. Online. Available: <https://developer.bluetooth.org/TechnologyOverview/Pages/HSP.aspx>
- [40]. Bluetooth SIG, Inc," Hands-Free Profile", Bluetooth, 2015. Online. Available: <https://developer.bluetooth.org/TechnologyOverview/Pages/HFP.aspx>
- [41]. ElectroniLAB, Módulo Bluetooth HC-06 Serial Rs232 (TTL), Ingeniería y diseño electrónico. Online. Available: <http://electronilab.co/tienda/modulo-bluetooth-hc-06-serial-rs232ttl/>
- [42]. Vistronica, Módulo Bluetooth Maestros/Esclavo HC-05, Online. Available: <https://www.vistronica.com/comunicaciones/modulo-bluetooth-maestros-esclavo-hc-05-detail.html>
- [43]. SparkFun, SparkFun Bluetooth Modem - BlueSMiRF Silver, Bluetooth, Online. Available: <https://www.sparkfun.com/products/12577>
- [44]. SparkFun, SparkFun Bluetooth Modem - BlueSMiRF Gold, BluetoothOnline. Available: <https://www.sparkfun.com/products/12582>
- [45]. Adafruit ®, Introducing Bluefruit EZ-Link , [Online]. Available: <https://learn.adafruit.com/introducing-bluefruit-ez-link>
- [46]. R.C.Sergio, Desarrollo de aplicaciones con comunicación remota basadas en módulos ZigBee y 802.15.4, 1^{ra}Ed., Buenos Aires: GAE, 2009,pp.138.
- [47]. J. Stefan, P Xavier, et al., Domótica y Hogar digital, ZigBee, Madrid, España, Ediciones Paraninfo, SA., unidad 6, sec. 6.8, pp.174-179.

- [48]. ZigBee Alliance, "ZigBee: The Open, Global Wireless Standard for Connecting Everyday Devices", ZigBee, 2015. Online. Available: <http://www.zigbee.org/what-is-zigbee/>
- [49]. R. M. Carlos, et al., "Gestión de redes inteligentes domésticas mediante ZigBee Smart Energy", Anales de mecánica y electricidad, Revista de la asociación de ingenieros del ICAI, septiembre – octubre 2012, No.16, Secc.3 Online. Available: http://www.revista-anales.es/web/n_16/seccion_3.html
- [50]. D. Pablo Jorge, Análisis del protocolo ZigBee, Facultad de informática, Universidad Nacional de la Plata, 2011, [Online]. Available: http://postgrado.info.unlp.edu.ar/Carreras/Especializaciones/Redes_y_Seguridad/Trabajos_Finales/Dignanni_Jorge_Pablo.pdf
- [51]. SparkFun, XBee 2mW Wire Antenna - Series 2 (ZigBee Mesh). Online. Available: <https://www.sparkfun.com/products/10414>
- [52]. SparkFun, XBee Pro 60mW U.FL Connection - Series 1 (802.15.4), Online . Available: <https://www.sparkfun.com/products/8710>
- [53]. Tomasi, Sistemas de Comunicaciones electrónicas, 4ta. Ed., Prentice Hall, Pearson Educación, México, 2003, Ch.20, pp.898-900.
- [54]. O. Joan, et. al, Instituto Federal de Telecomunicaciones: Estudio de Métricas de Eficiencia Espectral, Informe para el Instituto Federal de Telecomunicaciones (IFT) Revisión del uso del espectro en México, 03 de noviembre de 2015, Referencia: 2005111-453. Online, Available: http://www.ift.org.mx/sites/default/files/estudio_de_metricas_de_eficiencia_espectral.pdf
- [55]. H. M. José Manuel et. al, Sistema de telefonía, 5ta Ed., Thomson Paraninfo: España, 2006, pp.178.
- [56]. Instituto Federal de Telecomunicaciones, Sabías qué la Telefonía Móvil, IFT, México, 2015, Online. Available: <http://www.ift.org.mx/usuarios-telefonía-movil/sabias-que-la-telefonía-movil>
- [57]. Instituto Federal de Telecomunicaciones, El espectro radioeléctrico en México: Estudio y acciones, México, Online, Available: <http://www.ift.org.mx/sites/default/files/contenidogeneral/espectro-radioelectrico/espectro-radioelectrico-en-mexico-vp.pdf>
- [58]. M. M. José et. al, Ingeniería de Microondas: Técnicas experimentales, Prentice Hall: Madrid, 2002, ch.1, pp. 10.
- [59]. Carlos Soliz et al, Sistema domótico con funciones para el manejo remoto de electrodomésticos, ing. Elect., Univ. Cuenca, [Online]. Available: <http://dspace.ucuenca.edu.ec/bitstream/123456789/696/1/te327.pdf>
- [60]. Robodacta MX, SM5100B Celular Shield, [Online]. Available: <http://store.robodacta.mx/controladores/arduino/shields-para-arduino/sm5100b-celular-shield/>
- [61]. AGElectronica, ARDUINO GSM SHIELD (INTEGRATED ANTENNA) A000043, [Online]. Available:http://www.agelectronica.com/Virtual_Shop/index.asp#

- [62]. AGElectronica, ARDUINO GSM SHIELD (ANTENNA CONNECTOR) A000044, [Online]. Available: http://www.agelectronica.com/Virtual_Shop/index.asp#
- [63]. Gemalto, The cinterion Concept Board, 2016, [Online]. Available: <http://www.gemalto.com/m2m/development/cinterion-concept-board>
- [64]. H. L. John, P. A. David, Arquitectura de Computadores: un enfoque cuantitativo, Mc. Graw Hill, 1ra. Ed., España, Ch.4, pp. 201.
- [65]. Stallings William, Organización y Arquitectura de computadores, Pearson, Prentice Hall, 7ma. Ed., Ch. 2 y 7, pp. 38, 208-210.
- [66]. B. Daniel, Microcontroladores: Funcionamiento, programación y aplicaciones prácticas, 1ra Ed., Users, Buenos Aires, 2011, ch.4, pp. 102-105.
- [67]. M. P. Enrique et. al, Sistemas Electrónicos Digitales, 10ma. Ed., Marcombo, España, 2007, ch. 1 y 12, pp.19, 20, 647-651.
- [68]. T.A.Oscar, Arduino: curso práctico de información, RCLibros: España, 2013, ch.2, pp.76-77
- [69]. Atmel, (2015), ATmega2560, [Online]. Available: http://www.atmel.com/Images/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf
- [70]. ROBODACTA.MX, (2014), Arduino UNO, [Online]. Available: http://www.robodacta.mx/index.php?dispatch=products.view&product_id=413
- [71]. ROBODACTA.MX, (2014), Arduino MEGA 2560, [Online]. Available: http://www.robodacta.mx/index.php?dispatch=products.view&product_id=410
- [72]. ROBODACTA.MX, (2014), Arduino Mega ADK, [Online]. Available: http://www.robodacta.mx/index.php?dispatch=products.view&product_id=411
- [73]. ROBODACTA.MX, (2014), Arduino Due, [Online]. Available: http://www.robodacta.mx/index.php?dispatch=products.view&product_id=408
- [74]. ROBODACTA.MX, (2014), Arduino Leonardo, [Online]. Available: http://www.robodacta.mx/index.php?dispatch=products.view&product_id=406
- [75]. Raspberry Pi, Model B, [Online]. Available: <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>
- [76]. Raspberry Pi, Model A+, [Online]. Available: <https://www.raspberrypi.org/products/model-a-plus/>
- [77]. Intel®, Galileo Board, Intel® Galileo Gen 2 Board, and Intel® Edison Board, [Online]. Available: http://www.intel.com/buy/us/en/product/emergingtechnologies/intel-galileo-gen-2-board-462661?_ga=1.211981267.635598952.1444514601
- [78]. Kuo Yue, Thin Film Transistor Technology: Past, Present, and Future, The Electrochemical Society Interface, Spring, 2013.
- [79]. Fujitsu, Fundamentals of Liquid Crystal Displays – How They Work and What They Do, Fujitsu Microelectronics America, Inc, U.S.A., 2006.

- [80]. SparkFun Electronics ®, BEAGLEBONE Negro Cabo - LCD (4,3 "), [Online]. Available: <https://www.sparkfun.com/products/12085>
- [81]. adafruit, TFT Touch Shield for Arduino - 2.8, [Online]. Available: <http://www.adafruit.com/products/376>
- [82]. adafruit , PiTFT - Assembled 480x320 3.5" TFT+Touchscreen for Raspberry Pi, [Online]. Available: <http://www.adafruit.com/products/2097>
- [83]. Arduino, Getting Started with the Arduino TFT Screen [Online]. Available: <https://www.arduino.cc/en/Guide/TFT>
- [84]. AGelectronica, ANTENA GSM/GPRS QUBBAND, [Online]. Available: http://www.walkelectronics.com/sdp/2042842/4/pd-6785379/11619789-2585027/GSM_GPRS_AMPS_Quad_Band_Antenna_WK-GSM011-SMA_MRA.html
- [85]. Arduino, Arduino ADK: Esquemático, 2016, [Online]. Available: https://www.arduino.cc/en/uploads/Main/ADK_MEGA_2560-Rev2-sch.pdf
- [86]. NeuroSky, ThinkGear Serial Stream Guide, 2016, [Online]. Available: http://developer.neurosky.com/docs/doku.php?id=thinkgear_communications_protocol

Anexo A: Diagrama de flujo del sistema BCI



Anexo B: Código del sistema BCI Mobile

```

1. #include <SPI.h>
2. #include <SD.h>
3. #include <TFT.h>
4. #include <GSM.h>
5. #define BAUDRATE 57600
6. #define sd_cs 12
7. #define cs 11
8. #define dc 9
9. #define rst 8
10. TFT TFTscreen = TFT(cs, dc, rst);
11. #define PINNUMBER ""
12. GSM gsmAccess;
13. GSM_SMS sms;
14. int variable=0;
15. int estado=0;
16. int contbyte=0;
17. int cont=0;
18. byte dato[5]={0};
19. byte datog[32]={0};
20. unsigned long int varcod1=0;
21. unsigned long int varcod2=0;
22. unsigned long int totalcod=0;
23. unsigned long int totalcod1=0;
24. unsigned long int totalcod2=0;
25. int checksum=0;
26. byte generatedChecksum= 0;
27. int muestra=0;
28. int n=0;
29. int on=0;
30. int off=1;
31. int pin=11;
32. int x=0;
33. int y=0;
34. int yPos=125;
35. int xPos=0;
36. int contop=0;
37. int contsub1=0;
38. int contsub2=0;
39. int opcont=0;
40. int contag=0;
41. int num=0;
42. int num1=0;
43. int contpoor=0;
44. char* bienvenido[]={"B", "i", "e",
"n", "v", "e", "n", "i", "d", "o"};
45. char* menupr[]={ " Menu\n principal", " SMS\n
Basicos", " SMS de\n control"};
46. char* subml[]={ "S.O.S", " Quiero\n
comer", "Quiero ir\n al\nsanitario", " Tengo\n
sed", "Me duele\nla cabeza", "Me quiero\n
asear", "Salir" };
47. char* subm2[]={ "Enciende\n la T.V.", "Apaga la\n
T.V.", " Sube el\n volumen", " Baja el\n volumen", "
Cambia\n de canal"};
48. char* subtitulo[]={ "SMS Basicos", " SMS\n de
control"};
49. char* opcion[]={ "ENVIAR", "CANCELAR"};
50. unsigned int signals;
51. volatile boolean detectaparpadeo = false;
52. volatile boolean poor = false;
53. boolean notConnected = true;
54. PImage par;
55. PImage mala;
56. PImage buena;
57. PImage nd;
58. void setup() {
59. TFTscreen.begin();
60. Serial1.begin(BAUDRATE);
61. Serial.begin(BAUDRATE);
62. TFTscreen.background(255, 255, 255);
63. if (!SD.begin(sd_cs)) {
64. return;
65. }
66. TFTscreen.begin();
67. par = TFTscreen.loadImage("ojo.bmp");
68. mala=TFTscreen.loadImage("malac.bmp");
69. nd=TFTscreen.loadImage("nada.bmp");
70. buena=TFTscreen.loadImage("buenac.bmp");
71. TFTscreen.stroke(0,0,0);
72. TFTscreen.setTextSize(2);
73. TFTscreen.text("UACM Cautepec", 5, 30);
74. TFTscreen.setTextSize(3);
75. TFTscreen.text("BCI", 55, 60);
76. TFTscreen.text("Mobile", 30, 90);
77. attachInterrupt(4,byteabyte,CHANGE);
78. boolean notConnected = true;
79. while (notConnected){
80. if (gsmAccess.begin(PINNUMBER) ==
GSM_READY){
81. notConnected = false;
82. }else{
83. delay(1000);
84. }
85. }
86. bienvenida();
87. }
88. void loop() {
89. menu();
90. }
91. void byteabyte(){
92. if(Serial1.available()){
93. variable=Serial1.read();
94. if(variable== 170 && estado==0 && contbyte==0)
{
95. estado=1;
96. contbyte++;
97. }else
98. if(variable== 170 && estado==1&&
contbyte==1) {
99. estado=2;
100. contbyte++;
101. }else
102. if(variable==4 && estado==2 &&
contbyte==2){
103. estado=3;
104. contbyte++;
105. }else
106. if(variable==128 && estado==3 &&
contbyte==3){
107. dato[0]=variable;
108. estado=4;
109. contbyte++;
110. }else
111. if(variable==2 && estado==4 &&
contbyte==4){
112. dato[1]=variable;
113. cont++;
114. }else
115. if(cont==1 && estado==4 && contbyte==4){
116. dato[2]=variable;
117. varcod1=dato[2];
118. cont++;
119. }else
120. if(cont==2 && estado==4 && contbyte==4){
121. dato[3]=variable;
122. varcod2=dato[3];
123. cont++;
124. }else

```

```

125.     if(cont==3 && estado==4 && contbyte==4){
126.         checksum=variable;
127.         generatedChecksum =
            dato[0]+dato[1]+dato[2]+dato[3];
128.         generatedChecksum=      255      -
            generatedChecksum;
129.         if (checksum==generatedChecksum) {
130.             if (muestra<512){
131.                 totalcod=(varcod1<<8)|(varcod2);
132.                 if(totalcod&0xF000){
133.                     totalcod= ((~totalcod)&0xFFF)+1);
134.                 }else{
135.                     totalcod = (totalcod&0xFFF);
136.                 }
137.                 totalcod1 += totalcod;
138.                 muestra++;
139.             }else{
140.                 totalcod2=totalcod1/512;
141.                 parpadeo();
142.                 muestra=0;
143.                 totalcod1=0;
144.                 totalcod2=0;
145.             }
146.         }
147.         cont=0;
148.     }else
149.     if(variable==32 && estado==2 &&
        contbyte==2){
150.         estado=3;
151.         contbyte++;
152.     }else
153.     if(variable==2 && estado==3 &&
        contbyte==3){
154.         datog[0]=variable;
155.         estado=4;
156.         contbyte++;
157.         contpoor++;
158.     }else
159.     if(contpoor==1 && estado==4 &&
        contbyte==4){
160.         datog[1]=variable;
161.         poorq();
162.         estado=0;
163.         contbyte=0;
164.         variable=0;
165.         contpoor=0;
166.     }else{
167.         estado=0;
168.         contbyte=0;
169.         cont=0;
170.     }
171. }
172. }
173. void parpadeo(){
174.     if(170<totalcod2&&totalcod2<500){
175.         detectaparpadeo=true;
176.         TFTscreen.image(par, 0, 2);
177.         Serial.print(totalcod2);
178.     }else if (totalcod2>500){
179.         detectaparpadeo=false;
180.         TFTscreen.image(nd, 0, 2);
181.     }
182. }
183. void poorq(){
184.     if (datog[1]==0){
185.         TFTscreen.image(buena, 75, 0);
186.         poor=true;
187.     }else{
188.         poor=false;
189.         TFTscreen.image(mala, 75, 0);
190.     }
191. }

192. void bienvenida(){
193.     TFTscreen.background(255,255,255);
194.     for (int i = 0; i < 10; i++){
195.         TFTscreen.stroke(0,0,0);
196.         TFTscreen.setTextSize(3);
197.         TFTscreen.text(bienvenido[i], x, y);
198.         x=x+16;
199.         y=y+12;
200.     }
201.     delay(3000);
202.     TFTscreen.background(255,255,255);
203.     for (int i = 0; i < 10; i++){
204.         TFTscreen.stroke(255,0,0);
205.         TFTscreen.setTextSize(3);
206.         TFTscreen.text(bienvenido[i], x, y);
207.         x=x+16;
208.         y=y+12;
209.     }
210. }
211. void points(){
212.     for(xPos=0; xPos<=160;xPos+=10){
213.         TFTscreen.stroke(0,0,255);
214.         TFTscreen.point(xPos, yPos);
215.         TFTscreen.point(xPos, yPos+1);
216.         TFTscreen.point(xPos, yPos-1);
217.         TFTscreen.point(xPos+1, yPos);
218.         TFTscreen.point(xPos-1, yPos);
219.         TFTscreen.point(xPos+1, yPos+1);
220.         TFTscreen.point(xPos+1, yPos-1);
221.         TFTscreen.point(xPos-1, yPos+1);
222.         TFTscreen.point(xPos-1, yPos-1);
223.         delay(200);
224.     }
225.     for(xPos=0; xPos<=160;xPos+=10){
226.         TFTscreen.stroke(255,255,255);
227.         TFTscreen.point(xPos, yPos);
228.         TFTscreen.point(xPos, yPos+1);
229.         TFTscreen.point(xPos, yPos-1);
230.         TFTscreen.point(xPos+1, yPos);
231.         TFTscreen.point(xPos-1, yPos);
232.         TFTscreen.point(xPos+1, yPos+1);
233.         TFTscreen.point(xPos+1, yPos-1);
234.         TFTscreen.point(xPos-1, yPos+1);
235.         TFTscreen.point(xPos-1, yPos-1);
236.         delay(50);
237.     }
238. }
239. void menu(){
240.     contop++;
241.     if(contop==1){
242.         TFTscreen.background(255,255,255);
243.         TFTscreen.stroke(0,0,0);
244.         TFTscreen.setTextSize(2);
245.         TFTscreen.text(menupr[0],0, 20);
246.         TFTscreen.setTextSize(3);
247.         TFTscreen.text(menupr[1],0, 65);
248.         points();
249.     }
250.     if( detectaparpadeo==true&&poor==true){
251.         detectaparpadeo=false;
252.         poor=false;
253.         contsubl=0;
254.         return submenul();
255.     }else{
256.         return menu();
257.     }
258.     if(contop==2){
259.         TFTscreen.background(255,255,255);
260.         TFTscreen.stroke(0,0,0);
261.         TFTscreen.setTextSize(2);
262.         TFTscreen.text(menupr[0],0, 20);
263.         TFTscreen.setTextSize(3);

```

```

264.     TFTscreen.text(menupr[2],0, 65);
265.     points();
266.     if (detectaparpadeo==true&&poor==true){
267.         detectaparpadeo=false;
268.         poor=false;
269.         contsub2=0;
270.         return submenu2();
271.     }else{
272.         return menuup();
273.     }
274.     }else{
275.         contop=0;
276.         return menuup();
277.     }
278.     }
279.     }
280. void submenu1(){
281.     contsubl++;
282.     if (contsubl==1){
283.         TFTscreen.background(255,255,255);
284.         TFTscreen.stroke(0,0,0);
285.         TFTscreen.setTextSize(2);
286.         TFTscreen.text(subtitulo[0],10, 25);
287.         TFTscreen.setTextSize(4);
288.         TFTscreen.text(subml[0],20, 60);
289.         points();
290.         if (detectaparpadeo==true&&poor==true){
291.             detectaparpadeo=false;
292.             poor=false;
293.             opcont=0;
294.             num=1;
295.             opciones();
296.         }else{
297.             submenu1();
298.         }
299.     }else{
300.         if (contsubl==2){
301.             TFTscreen.background(255,255,255);
302.             TFTscreen.stroke(0,0,0);
303.             TFTscreen.setTextSize(2);
304.             TFTscreen.text(subtitulo[0],10, 25);
305.             TFTscreen.setTextSize(3);
306.             TFTscreen.text(subml[1],0, 60);
307.             points();
308.             if (detectaparpadeo==true&&poor==true){
309.                 detectaparpadeo=false;
310.                 poor=false;
311.                 opcont=0;
312.                 num=2;
313.                 opciones();
314.             }else{
315.                 submenu1();
316.             }
317.         }else{
318.             if (contsubl==3){
319.                 TFTscreen.background(255,255,255);
320.                 TFTscreen.stroke(0,0,0);
321.                 TFTscreen.setTextSize(2);
322.                 TFTscreen.text(subtitulo[0],10, 25);
323.                 TFTscreen.setTextSize(3);
324.                 TFTscreen.text(subml[2],0, 50);
325.                 points();
326.                 if (detectaparpadeo==true&&poor==true){
327.                     detectaparpadeo=false;
328.                     poor=false;
329.                     opcont=0;
330.                     num=3;
331.                     opciones();
332.                 }else{
333.                     submenu1();
334.                 }
335.             }else{
336.                 if (contsubl==4){
337.                     TFTscreen.background(255,255,255);
338.                     TFTscreen.stroke(0,0,0);
339.                     TFTscreen.setTextSize(2);
340.                     TFTscreen.text(subtitulo[0],10, 25);
341.                     TFTscreen.setTextSize(3);
342.                     TFTscreen.text(subml[3],0, 60);
343.                     points();
344.                     if (detectaparpadeo==true&&poor==true){
345.                         poor=false;
346.                         detectaparpadeo=false;
347.                         opcont=0;
348.                         num=4;
349.                         opciones();
350.                     }else{
351.                         submenu1();
352.                     }
353.                 }else{
354.                     if (contsubl==5){
355.                         TFTscreen.background(255,255,255);
356.                         TFTscreen.stroke(0,0,0);
357.                         TFTscreen.setTextSize(2);
358.                         TFTscreen.text(subtitulo[0],10, 25);
359.                         TFTscreen.setTextSize(3);
360.                         TFTscreen.text(subml[4],0, 60);
361.                         points();
362.                         if (detectaparpadeo==true&&poor==true){
363.                             poor=false;
364.                             detectaparpadeo=false;
365.                             opcont=0;
366.                             num=5;
367.                             opciones();
368.                         }else{
369.                             submenu1();
370.                         }
371.                     }else{
372.                         if (contsubl==6){
373.                             TFTscreen.background(255,255,255);
374.                             TFTscreen.stroke(0,0,0);
375.                             TFTscreen.setTextSize(2);
376.                             TFTscreen.text(subtitulo[0],10, 25);
377.                             TFTscreen.setTextSize(3);
378.                             TFTscreen.text(subml[5],0, 60);
379.                             points();
380.                             if (detectaparpadeo==true&&poor==true){
381.                                 poor=false;
382.                                 detectaparpadeo=false;
383.                                 opcont=0;
384.                                 num=6;
385.                                 opciones();
386.                             }else{
387.                                 submenu1();
388.                             }
389.                         }else{
390.                             if (contsubl==7){
391.                                 TFTscreen.background(255,255,255);
392.                                 TFTscreen.stroke(0,0,0);
393.                                 TFTscreen.setTextSize(2);
394.                                 TFTscreen.text(subtitulo[0],10, 25);
395.                                 TFTscreen.setTextSize(4);
396.                                 TFTscreen.text(subml[6], 20, 60);
397.                                 points();
398.                                 if (detectaparpadeo==true&&poor==true){
399.                                     poor=false;
400.                                     detectaparpadeo=false;
401.                                     contop=0;
402.                                     num=0;
403.                                     menuup();
404.                                 }else{
405.                                     submenu1();
406.                                 }

```

```

407.         }else{
408.             contsub1=0;
409.             submenu1();
410.         }
411.     }
412. }
413. }
414. }
415. }
416. }
417. }
418. void submenu2(){
419.     contsub2++;
420.     if (contsub2==1){
421.         TFTscreen.background(255,255,255);
422.         TFTscreen.stroke(0,0,0);
423.         TFTscreen.setTextSize(2);
424.         TFTscreen.text(subtitulo[1],10, 20);
425.         TFTscreen.setTextSize(3);
426.         TFTscreen.text(subm2[0],0, 65);
427.         points();
428.         if(detectaparpadeo==true&&poor==true){
429.             poor=false;
430.             detectaparpadeo=false;
431.             opcont=0;
432.             num=7;
433.             opciones();
434.         }else{
435.             submenu2();
436.         }
437.     }else{
438.         if (contsub2==2){
439.             TFTscreen.background(255,255,255);
440.             TFTscreen.stroke(0,0,0);
441.             TFTscreen.setTextSize(2);
442.             TFTscreen.text(subtitulo[1],10, 20);
443.             TFTscreen.setTextSize(3);
444.             TFTscreen.text(subm2[1],0, 65);
445.             points();
446.             if(detectaparpadeo==true&&poor==true){
447.                 poor=false;
448.                 detectaparpadeo=false;
449.                 opcont=0;
450.                 num=8;
451.                 opciones();
452.             }else{
453.                 submenu2();
454.             }
455.         }else{
456.             if (contsub2==3){
457.                 TFTscreen.background(255,255,255);
458.                 TFTscreen.stroke(0,0,0);
459.                 TFTscreen.setTextSize(2);
460.                 TFTscreen.text(subtitulo[1],10, 20);
461.                 TFTscreen.setTextSize(3);
462.                 TFTscreen.text(subm2[2],0, 65);
463.                 points();
464.                 if(detectaparpadeo==true&&poor==true){
465.                     poor=false;
466.                     detectaparpadeo=false;
467.                     opcont=0;
468.                     num=9;
469.                     opciones();
470.                 }else{
471.                     submenu2();
472.                 }
473.             }else{
474.                 if (contsub2==4){
475.                     TFTscreen.background(255,255,255);
476.                     TFTscreen.stroke(0,0,0);
477.                     TFTscreen.setTextSize(2);
478.                     TFTscreen.text(subtitulo[1],10, 20);
479.                     TFTscreen.setTextSize(3);
480.                     TFTscreen.text(subm2[3],0, 65);
481.                     points();
482.                     if(detectaparpadeo==true&&poor==true){
483.                         poor=false;
484.                         detectaparpadeo=false;
485.                         opcont=0;
486.                         num=10;
487.                         opciones();
488.                     }else{
489.                         submenu2();
490.                     }
491.                 }else{
492.                     if (contsub2==5){
493.                         TFTscreen.background(255,255,255);
494.                         TFTscreen.stroke(0,0,0);
495.                         TFTscreen.setTextSize(2);
496.                         TFTscreen.text(subtitulo[1],10, 20);
497.                         TFTscreen.setTextSize(3);
498.                         TFTscreen.text(subm2[4],0, 65);
499.                         points();
500.                         if(detectaparpadeo==true&&poor==true){
501.                             poor=false;
502.                             detectaparpadeo=false;
503.                             opcont=0;
504.                             num=11;
505.                             opciones();
506.                         }else{
507.                             submenu2();
508.                         }
509.                     }else{
510.                         if (contsub2==6){
511.                             TFTscreen.background(255,255,255);
512.                             TFTscreen.stroke(0,0,0);
513.                             TFTscreen.setTextSize(2);
514.                             TFTscreen.text(subtitulo[1],10, 20);
515.                             TFTscreen.setTextSize(4);
516.                             TFTscreen.text(subm1[6], 20, 65);
517.                             points();
518.                             if(detectaparpadeo==true&&poor==true){
519.                                 poor=false;
520.                                 detectaparpadeo=false;
521.                                 contop=0;
522.                                 num=0;
523.                                 menu();
524.                             }else{
525.                                 submenu2();
526.                             }
527.                         }else{
528.                             contsub2=0;
529.                             submenu2();
530.                         }
531.                     }
532.                 }
533.             }
534.         }
535.     }
536. }
537. void opciones(){
538.     opcont++;
539.     if (opcont==1){
540.         TFTscreen.background(255,255,255);
541.         TFTscreen.stroke(0,0,0);
542.         TFTscreen.setTextSize(3);
543.         TFTscreen.text(opcion[0], 25, 50);
544.         points();
545.         if(detectaparpadeo==true&&poor==true){
546.             poor=false;
547.             detectaparpadeo=false;
548.             agenda();
549.         }else{

```

```

550.     opciones ();
551.     }
552. }else{
553.     if (opcont==2){
554.         TFTscreen.background(255,255,255);
555.         TFTscreen.stroke(0,0,0);
556.         TFTscreen.setTextSize(3);
557.         TFTscreen.text(opcion[1], 5, 50);
558.         points ();
559.         if (detectaparpadeo==true&&poor==true){
560.             poor=false;
561.             detectaparpadeo=false;
562.             contop=0;
563.             menup ();
564.         }else{
565.             opciones ();
566.         }
567.     }else{
568.         opcont=0;
569.         opciones ();
570.     }
571. }
572. }
573. void agenda () {
574.     contag++;
575.     if (contag==1){
576.         TFTscreen.background(255,255,255);
577.         TFTscreen.stroke(0,0,0);
578.         TFTscreen.setTextSize(3);
579.         TFTscreen.text("contacto\n 1", 5, 25);
580.         TFTscreen.setTextSize(2);
581.         TFTscreen.text("5528185535", 20, 80);
582.         points ();
583.         if (detectaparpadeo==true&&poor==true){
584.             poor=false;
585.             detectaparpadeo=false;
586.             char remoteNum[20]="5528185535";
587.             if (num==1){
588.                 sms.beginSMS (remoteNum);
589.                 sms.print (subml [0]);
590.                 sms.endSMS ();
591.                 enviado ();
592.             }else if (num==2){
593.                 sms.beginSMS (remoteNum);
594.                 sms.print (subml [1]);
595.                 sms.endSMS ();
596.                 enviado ();
597.             }else if (num==3){
598.                 sms.beginSMS (remoteNum);
599.                 sms.print (subml [2]);
600.                 sms.endSMS ();
601.                 enviado ();
602.             }else if (num==4){
603.                 sms.beginSMS (remoteNum);
604.                 sms.print (subml [3]);
605.                 sms.endSMS ();
606.                 enviado ();
607.             }else if (num==5){
608.                 sms.beginSMS (remoteNum);
609.                 sms.print (subml [4]);
610.                 sms.endSMS ();
611.                 enviado ();
612.             }else if (num==6){
613.                 sms.beginSMS (remoteNum);
614.                 sms.print (subml [5]);
615.                 sms.endSMS ();
616.                 enviado ();
617.             }else if (num==7){
618.                 sms.beginSMS (remoteNum);
619.                 sms.print (subm2 [0]);
620.                 sms.endSMS ();
621.                 enviado ();
622.             }else if (num==8){
623.                 sms.beginSMS (remoteNum);
624.                 sms.print (subm2 [1]);
625.                 sms.endSMS ();
626.                 enviado ();
627.             }else if (num==9){
628.                 sms.beginSMS (remoteNum);
629.                 sms.print (subm2 [2]);
630.                 sms.endSMS ();
631.                 enviado ();
632.             }else if (num==10){
633.                 sms.beginSMS (remoteNum);
634.                 sms.print (subm2 [3]);
635.                 sms.endSMS ();
636.                 enviado ();
637.             }else if (num==11){
638.                 sms.beginSMS (remoteNum);
639.                 sms.print (subm2 [4]);
640.                 sms.endSMS ();
641.                 enviado ();
642.             }else{
643.                 num=0;
644.             }
645.         }else{
646.             agenda ();
647.         }
648.     }else {
649.         if (contag==2){
650.             TFTscreen.background(255,255,255);
651.             TFTscreen.stroke(0,0,0);
652.             TFTscreen.setTextSize(3);
653.             TFTscreen.text("contacto\n 2", 5, 25);
654.             TFTscreen.setTextSize(2);
655.             TFTscreen.text("5520908217", 20, 80);
656.             points ();
657.             if (detectaparpadeo==true&&poor==true){
658.                 poor=false;
659.                 detectaparpadeo=false;
660.                 char remoteNum1[20]="5520908217";
661.                 if (num1==1){
662.                     sms.beginSMS (remoteNum1);
663.                     sms.print (subml [0]);
664.                     sms.endSMS ();
665.                     enviado ();
666.                 }else if (num1==2){
667.                     sms.beginSMS (remoteNum1);
668.                     sms.print (subml [1]);
669.                     sms.endSMS ();
670.                     enviado ();
671.                 }else if (num1==3){
672.                     sms.beginSMS (remoteNum1);
673.                     sms.print (subml [2]);
674.                     sms.endSMS ();
675.                     enviado ();
676.                 }else if (num1==4){
677.                     sms.beginSMS (remoteNum1);
678.                     sms.print (subml [3]);
679.                     sms.endSMS ();
680.                     enviado ();
681.                 }else if (num1==5){
682.                     sms.beginSMS (remoteNum1);
683.                     sms.print (subml [4]);
684.                     sms.endSMS ();
685.                     enviado ();
686.                 }else if (num1==6){
687.                     sms.beginSMS (remoteNum1);
688.                     sms.print (subml [5]);
689.                     sms.endSMS ();
690.                     enviado ();
691.                 }else if (num1==7){
692.                     sms.beginSMS (remoteNum1);
693.                     sms.print (subm2 [0]);

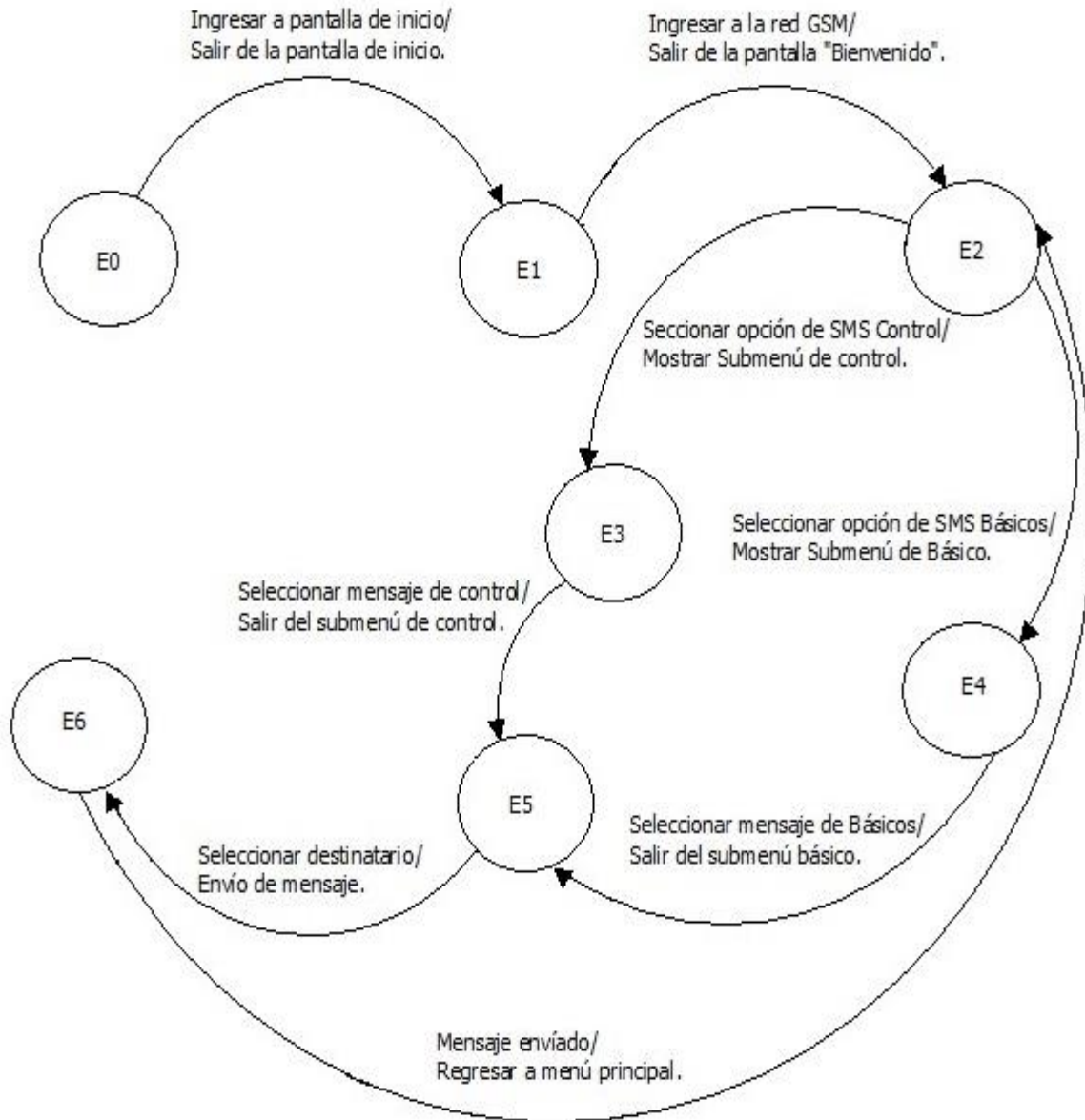
```

```

694. sms.endSMS();
695. enviado();
696.     }else if(num1==8){
697.         sms.beginSMS(remoteNum1);
698.         sms.print(subm2[1]);
699.         sms.endSMS();
700.         enviado();
701.     }else if(num1==9){
702.         sms.beginSMS(remoteNum1);
703.         sms.print(subm2[2]);
704.         sms.endSMS();
705.         enviado();
706.     }else if(num1==10){
707.         sms.beginSMS(remoteNum1);
708.         sms.print(subm2[3]);
709.         sms.endSMS();
710.         enviado();
711.     }else if(num1==11){
712.         sms.beginSMS(remoteNum1);
713.         sms.print(subm2[4]);
714.         sms.endSMS();
715.         enviado();
716.     }else{
717.         num1=0;
718.     }
719.     }else{
720.         agenda();
721.     }
722. }else{
723.     if(contag==3){
724.         TFTscreen.background(255,255,255);
725.         TFTscreen.stroke(0,0,0);
726.         TFTscreen.setTextSize(3);
727.         TFTscreen.text(opcion[1], 5, 50);
728.         points();
729.         if(detectaparpadeo==true&&poor==true){
730.             poor=false;
731.             detectaparpadeo=false;
732.             contag=0;
733.             contop=0;
734.             menu();
735.         }else{
736.             agenda();
737.         }
738.     }else{
739.         contag=0;
740.         agenda();
741.     }
742. }
743. }
744. }
745. void enviado(){
746.     TFTscreen.background(255,255,255);
747.     TFTscreen.stroke(0,0,0);
748.     TFTscreen.setTextSize(3);
749.     TFTscreen.text(" sms\n enviado", 0, 35);
750.     delay(1000);

```

Anexo C: Diagrama de estados del sistema BCI Mobile para la creación del menú.



- E0-Encender pantalla TFT.
- E1-Mostrar el texto "Bienvenido" en la pantalla TF.
- E2-Mostrar el menú principal.
- E3-Mensajes predefinidos de "SMS Control".
- E4-Mensajes predefinidos de "SMS Básicos".
- E5-Mostrar agenda "contacto N".
- E6-Enviar mensaje.