

DESARROLLO DE UN SISTEMA DE COMUNICACIÓN  
MEDIANTE ELECTROMIOGRAFÍA PARA PACIENTES  
CON DISCAPACIDAD MOTORA SEVERA

Yuber Alejandro Galeano Traslaviña  
Luis Felipe Bejarano Gil

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERIAS FISICO-MECANICAS  
ESCUELA DE INGENIERÍA ELECTRÓNICA  
BUCARAMANGA 2014

DESARROLLO DE UN SISTEMA DE COMUNICACIÓN  
MEDIANTE ELECTROMIOGRAFÍA PARA PACIENTES  
CON DISCAPACIDAD MOTORA SEVERA

Yuber Alejandro Galeano Traslaviña  
Luis Felipe Bejarano Gil

Trabajo de grado para optar por el título de:  
Ingeniero Electrónico

Director:  
Daniel Alfonso Sierra Bueno

Codirectora:  
Lola Xiomara Bautista Rozo

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERIAS FISICO-MECANICAS  
ESCUELA DE INGENIERÍA ELECTRÓNICA  
BUCARAMANGA 2014

## **AGRADECIMIENTOS**

Agradecemos ante todo a Dios por hacerlo todo posible. Gracias a nuestros directores de proyecto Daniel Sierra y Lola Xiomara Bautista por la paciencia y los consejos, a Diego Bejarano por toda su ayuda y soporte en el desarrollo de la interfaz y a nuestros familiares por todo su amor y fortaleza.

## **DEDICATORIA**

A Leonardo quien con su espíritu de fortaleza y firme convicción de vivir, nos inspira a dejar a un lado las limitaciones y buscar la forma de ayudar a los demás.

## TABLA DE CONTENIDO

<b>INTRODUCCIÓN .....</b>	<b>13</b>
<b>1.MARCO TEÓRICO .....</b>	<b>13</b>
1.1. <i>PARPADEO (ocular) .....</i>	<i>13</i>
1.2. <i>Electromiografía .....</i>	<i>14</i>
1.3. <i>Conceptos generales.....</i>	<i>14</i>
1.4. <i>Características de la señal EMG .....</i>	<i>14</i>
1.5. <i>Características del ruido eléctrico .....</i>	<i>14</i>
1.6. <i>Mejorar la fidelidad de la señal EMG .....</i>	<i>15</i>
1.7. <i>Electrodo de referencia .....</i>	<i>15</i>
1.8. <i>Conceptos de seguridad eléctrica .....</i>	<i>15</i>
1.9. <i>Interfaz gráfica .....</i>	<i>15</i>
1.10. <i>Comunicación USB.....</i>	<i>16</i>
1.11. <i>Interfaz de Comunicación .....</i>	<i>17</i>
1.12. <i>Operación del Dispositivo .....</i>	<i>17</i>
1.13. <i>Puerto Virtual COM mediante comunicación CDC [11].....</i>	<i>17</i>
<b>2.DISEÑO E IMPLEMENTACIÓN DEL SISTEMA.....</b>	<b>18</b>
2.1. <i>Captura y adecuación de la señal .....</i>	<i>18</i>
2.2. <i>Preamplificación diferencial.....</i>	<i>18</i>
2.3. <i>Circuito del tercer electrodo (referencia) .....</i>	<i>18</i>
2.4. <i>Filtrado.....</i>	<i>18</i>
2.5. <i>Amplificación.....</i>	<i>19</i>
2.6. <i>Etapa de aislamiento del paciente .....</i>	<i>19</i>
2.8. <i>Primera propuesta de interfaz gráfica .....</i>	<i>20</i>
2.9. <i>Segunda propuesta de interfaz gráfica .....</i>	<i>21</i>
.....	<i>21</i>
2.10. <i>Implementación de la comunicación por puerto USB [15].....</i>	<i>22</i>
<b>3.RESULTADOS .....</b>	<b>22</b>
3.1. <i>Circuito para la captura de biopotenciales .....</i>	<i>22</i>
<b>4.Conclusiones y Recomendaciones.....</b>	<b>25</b>
<b>Referencias .....</b>	<b>25</b>

## TABLA DE FIGURAS

Figura 1 Músculo orbicular de los párpados .....	14
Figura 2 Músculo elevador del párpado superior .....	14
Figura 3 Puerto COM virtual mediante USB .....	17
Figura 4 Captura y adecuación de la señal .....	18
Figura 5 Esquema del circuito del tercer electrodo .....	18
Figura 6 Respuesta filtros pasa altas con frecuencia de corte 70 [Hz] ...	18
Figura 7 Respuesta filtros pasa bajas con frecuencia de corte 400 [Hz]	19
Figura 8 Diagrama del circuito conversor DC/DC.....	19
Figura 9 Arquitectura del proyecto.....	19
Figura 10 Entorno gráfico ventana de consonantes .....	19
Figura 11 Archivos generados de la interfaz .....	20
Figura 12 Presentación inicial.....	20
Figura 13 Ventana Principal.....	20
Figura 14 Ventana de frases.....	20
Figura 15 Ventana principal .....	21
Figura 16 Ventana guardar .....	21
Figura 17 Ventana de frases.....	21
Figura 18 Ventana principal de escritura .....	21
Figura 19 Ventana de Vocales.....	21
Figura 20 Ventana de consonantes .....	22
Figura 21 Ventana de herramientas .....	22
Figura 22 Señal EMG asociada a la acción del parpadeo .....	23
Figura 23 Señal EMG rectificada. Fuente Autores. ....	23
Figura 24 Señal EMG rectificada e integrada. ....	23
Figura 25 Intensidad de señal para diferentes tipos de parpadeo .....	23
Figura 26 Posición de los electrodos al utilizar el dispositivo .....	24
Figura 27 Enrutado del circuito .....	24
Figura 28 Placa del dispositivo .....	24
Figura 29 Carcasa del dispositivo.....	24

## RESUMEN

**TÍTULO:**

Desarrollo de un sistema de comunicación mediante electromiografía para pacientes con discapacidad motora severa<sup>1</sup>.

**AUTORES:**

Yuber Alejandro Galeano Traslaviña  
Luis Felipe Bejarano Gil<sup>2</sup>

**PALABRAS CLAVE:**

Comunicación escrita, discapacidad motora, Electromiografía, interfaz gráfica, ocular, bioingeniería.

**DESCRIPCIÓN:**

Este informe presenta la descripción del diseño y la implementación de un sistema portable y de bajo costo, que en conjunto con cualquier pc convencional facilite la comunicación básica a pacientes con discapacidad motora severa (En las extremidades y en el habla). El sistema funciona valiéndose de la señal eléctrica del parpadeo (señal digital de dos estados) para controlar un software de escritura.

La técnica utilizada para capturar la señal eléctrica es la electromiografía ocular, realizando el proceso de captura de forma convencional para electromiografía en los músculos de la cara y aislando eléctricamente al paciente de la alimentación del circuito para asegurar de este modo la seguridad de este. Esta señal es llevada a un microcontrolador que realiza dos tareas fundamentales: convertir la señal analógica en una señal digital, y enviar esta al software instalado en el PC. Por otra parte se realizó el diseño y desarrollo de un software para el proceso de escritura, que funciona valiéndose de una única señal de dos estados.

El sistema presenta tiempos aceptables de escritura tomando en cuenta que únicamente utiliza una señal para este fin, además el uso del software resulta bastante intuitivo y sencillo para los pacientes.

<sup>1</sup> Proyecto de Grado

<sup>2</sup> Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica Electrónica y Telecomunicaciones. Director: Daniel Alfonso Sierra Bueno. Codirectora: Lola Xiomara Bautista Rozo.

## ABSTRACT

**TITLE:**

Development of a communication system by electromyography for patients with severe motor disabilities<sup>3</sup>.

**AUTHORS:**

Yuber Alejandro Galeano Traslaviña  
Luis Felipe Bejarano Gil<sup>4</sup>

**KEYWORD:**

Written communication, motor disability, Electromyography, GUI, ocular, bioengineering.

**DESCRIPTION:**

This paper presents the description of the design and implementation of a portable low-cost device that with any conventional PC aids people with severe motor disability (quadriplegia and speech) with their communication. The device works taking the electrical signal of a blink (one-bit digital signal) to control a writing software that has been specifically designed for this kind of signal.

The technique used to capture the electrical signal is ocular electromyography, making the capture process for conventional manner for electromyography in facial muscles, the patient circuit is electrically isolating of the power thus to ensure the safety of this. This signal is taken to a microcontroller that performs two main tasks: to convert the analog signal into a digital signal, and send this to the software installed on the PC. Moreover, the design and development of software for the writing process, which operates using a single two-state signal was performed.

The write times for the system are acceptable considering that only one signal is used for this purpose, besides the use of the software is quite intuitive and easy for patients.

<sup>3</sup> Draft Grade

<sup>4</sup> Faculty of Mechanical and Physical Engineering. School of Electrical Electronics and telecommunications engineering. Director: Daniel Alfonso Sierra Good. Co-Director: Lola Bautista Xiomara Rozo.

# Desarrollo de un sistema de comunicación mediante electromiografía para pacientes con discapacidad motora severa<sup>5</sup>

*Luis Felipe Bejarano Gil Yuber Alejandro Galeano Traslaviña*

**Resumen**—Este informe presenta la descripción del diseño y la implementación de un dispositivo portable y de bajo costo, que en conjunto con cualquier pc convencional facilite la comunicación básica a pacientes con discapacidad motora severa (En las extremidades y en el habla). El dispositivo funciona valiéndose de la señal eléctrica del parpadeo (señal digital de dos estados) para controlar un software de escritura diseñado especialmente para este tipo de señal.

**Abstract**—This paper presents the description of the design and implementation of a portable low-cost device that with any conventional PC aids people with severe motor disability (quadriplegia and speech) with their communication. The device works taking the electrical signal of a blink (one-bit digital signal) to control a writing software that has been specifically designed for this kind of signal.

**Palabras clave**—Comunicación escrita, discapacidad motora, Electromiografía, interfaz gráfica, ocular, bioingeniería.

## INTRODUCCIÓN

La discapacidad motora severa en las extremidades, acompañada de dificultades en el habla es una condición muy atípica, por consiguiente es casi nula la disponibilidad de sistemas en el mercado con las características que requieren pacientes de este tipo para su comunicación. Además dado que el desarrollo y la comercialización de equipos de este tipo en Colombia es casi nula, cualquier sistema con características similares o cercanas a las requeridas resulta excesivamente costoso. Por esta razón se decidió desarrollar un prototipo que cumpla con los requerimientos básicos para las necesidades de esta discapacidad específica, cuya construcción implique un bajo costo para hacerlo asequible a personas que lo requieran.

El diseño del dispositivo se basó en la forma en que se comunica un paciente actualmente, utilizando una persona que interpreta sus parpadeos utilizando una matriz de letras. El sistema consiste en una interfaz gráfica de escritura en el PC y un dispositivo para la captura de biopotenciales eléctricos que se conecta al puerto USB y se comunica mediante este. De esta forma el parpadeo del paciente, con su señal eléctrica asociada, controla la interfaz gráfica de escritura.

Este reporte presenta una descripción detallada del proceso de diseño e implementación del dispositivo de comunicación. La técnica a utilizar para capturar la señal eléctrica es la electromiografía ocular, realizando el proceso de captura de forma convencional para electromiografía y aislando eléctricamente al paciente de la alimentación del circuito. Esta señal es llevada a un microcontrolador que realiza dos tareas fundamentales: convertir la señal análoga en una señal digital, y enviar esta al software instalado en el PC. Por otra parte se realizó el diseño y desarrollo de un software para el proceso de escritura, que funciona valiéndose de una única señal de dos estados.

## 1. MARCO TEÓRICO

### 1.1. PARPADEO (ocular)

Un parpadeo sucede cuando una persona cierra y abre sus párpados una o más veces. Esta acción particular puede ser refleja, por ejemplo cuando el cuerpo siente que los ojos necesitan hidratación, o voluntaria, cuando una persona agita sus pestañas parpadeando rápidamente o hace un guiño.

Hay múltiples músculos que intervienen en el proceso de parpadeo, los principales en el párpado superior que controlan la apertura y cierre son el músculo orbicular de los ojos y el músculo elevador del párpado superior.

El músculo orbicular de los ojos o músculo orbicular de los párpados mostrado en la Figura 1 es un músculo situado debajo de la piel, delante de la órbita ocular, en forma de anillo, ancho, aplanado y delgado, constituido por tres porciones: orbitaria, parpebral y lagrimal, se encarga de cerrar los párpados.

Se inserta, por dentro en el tendón orbicular, que a su vez se inserta en los labios anterior y posterior del canal lagrimal, en la apófisis ascendente del maxilar superior y la apófisis orbitaria interna del frontal y por fuera, en la cara profunda de la piel [1].

<sup>5</sup> Proyecto de grado en Ingeniería Electrónica, Universidad Industrial De Santander (UIS) (Colombia). Director: Daniel Alfonso Sierra Bueno, Profesor Asociado de la Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones de la UIS. Codirectora: Lola Xiomara Bautista Roza, Profesora Auxiliar de la Escuela de Ingeniería de Sistemas de la UIS.



Figura 1 Músculo orbicular de los párpados. [2]

El músculo elevador del párpado superior mostrado en la Figura 2 es un músculo voluntario que está situado en el interior de la órbita y cumple la función de elevar el párpado superior, como indica su nombre. Se inserta en la parte superior o techo de la órbita, desde donde se dirige hacia delante, para terminar en la piel del párpado superior. Cuando se contrae produce la elevación de dicho párpado. Esta acción es contrarrestada por el músculo orbicular de los párpados.

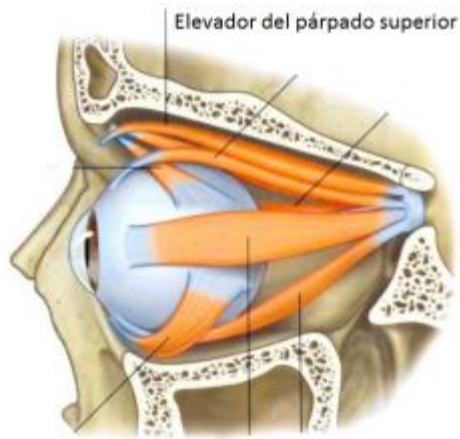


Figura 2 Músculo elevador del párpado superior. [2]

Otro músculo que interviene en la acción del parpadeo, cuando esta se realiza de forma voluntaria y forzada es el músculo corrugador de la ceja, mostrado en la Figura 1, ya que al realizar parpadeos con una fuerza superior a la habitual, se frunce el ceño y este músculo entra en acción, pero dado su tamaño, la contribución final a la señal EMG es menor.

### 1.2. Electromiografía

La actividad muscular genera potenciales eléctricos asociados, y su registro constituye el electromiograma (EMG). Estos potenciales se pueden medir en la superficie del cuerpo cerca del músculo de interés o de manera directa atravesando la piel con electrodos de aguja. Puesto que la mayoría de medidas EMG se proponen obtener una indicación de la cantidad de actividad de un músculo determinado, o grupo de músculos, más que de una fibra muscular individual, la señal es usualmente la suma de potenciales de acción individuales de todas las fibras que constituyen el músculo o los músculos

donde se mide. Al igual que el ECG, los electrodos reciben potenciales de todos los músculos dentro de su alcance, esto quiere decir que los potenciales de los músculos grandes cercanos pueden interferir al querer medir el EMG aun si los electrodos se colocan directamente sobre los músculos pequeños.

El potencial de acción de un músculo determinado (o fibra nerviosa) posee una magnitud fija, independientemente de la intensidad del estímulo que genera la respuesta. Así, en un músculo, la intensidad de la acción no incrementa la amplitud neta del impulso de potencial de acción sino el ritmo con que se dispara cada fibra muscular y el número de fibras que se activan en un instante determinado. La amplitud de la forma de onda EMG medida es la suma de todos los potenciales generados en un instante determinado. Dado que estos potenciales se producen tanto con polaridades tanto positivas como negativas, en un par de electrodos determinados, a veces se adicionan y a veces se cancelan, de este modo la señal EMG tiene aspecto de ruido aleatorio, siendo la energía de la señal función de la cantidad de actividad muscular [3].

### 1.3. Conceptos generales

Cuando se detecta y se captura la señal EMG, hay dos asuntos clave en relación a la fidelidad de la señal. El primero es la relación señal a ruido. Esta es la proporción entre la energía de la señal EMG y la energía de la señal de ruido. En general el ruido es definido como las señales eléctricas que no son parte de la señal EMG deseada. El otro es la distorsión de la señal, que consiste en la contribución de algunos componentes frecuenciales que no pueden ser eliminados.

### 1.4. Características de la señal EMG

Ya está establecido que la amplitud de la señal EMG es de naturaleza estocástica (aleatoria) y puede ser representada por una función de distribución gaussiana. La amplitud de la señal está en el rango de 0 a 10mV (pico a pico) o 0 a 1.5mV (rms). La energía útil de la señal está limitada al rango de frecuencias 0-500Hz, con la energía dominante en el rango de 50-500Hz. Las señales a usar son aquellas con la energía por encima del nivel de ruido [3].

### 1.5. Características del ruido eléctrico

El ruido puede provenir de diferentes fuentes como:

Ruido de los componentes electrónicos - Todos los equipos electrónicos generan ruido eléctrico. Este ruido tiene componentes de frecuencia que van desde 0 a algunos miles de Hz. Este ruido no puede ser eliminado, el solo puede ser reducido utilizando componentes electrónicos de alta calidad, técnicas inteligentes de diseño y construcción de circuitos.

Ruido ambiente - Este ruido es originado por fuentes de radiación electromagnética, como transmisión de radio y televisión, cables eléctricos de potencia, bombillas de luz, lámparas fluorescentes, etc. En general todos los dispositivos

electromagnéticos generan y pueden contribuir con ruido. La superficie de nuestro cuerpo está constantemente inundada con radiación electromagnética y es virtualmente imposible evitar la exposición a esta en la superficie del planeta. La mayor concentración del ruido ambiente está en 60Hz o (50Hz).

Movimiento - Hay dos principales fuentes de ruido por movimiento: una la superficie de detección del electrodo y la piel, la otra por el movimiento del cable que conecta el electrodo a los amplificadores. Ambas pueden ser significativamente reducidas con un diseño apropiado del circuito. La señal eléctrica de ambas fuentes de ruido concentra la mayor parte de su energía en el rango de 0 a 20Hz.

Inestabilidad inherente de la señal - La amplitud de la señal EMG es de naturaleza aleatoria. Los componentes frecuenciales entre 0-20Hz son particularmente inestables, debido a que se ven afectados por la naturaleza cuasi-aleatoria de la tasa de disparo de las unidades motoras que en la mayoría de condiciones están en esta región de frecuencia. Debido a la naturaleza inestable de estos componentes de la señal, es aconsejable considerar esto como ruido no deseado y eliminarlos de la señal [3].

#### 1.6. Mejorar la fidelidad de la señal EMG

Es deseable obtener una señal que contenga la máxima cantidad de información de la señal EMG y la mínima cantidad de contaminación por ruido eléctrico. Por lo tanto es importante que cualquier dispositivo de detección y captura procese la señal linealmente. En particular la señal no puede ser recortada, esto es, los picos no deben ser alterados y no se debe realizar un filtrado innecesario.

#### 1.7. Electrodo de referencia

Cuando un amplificador diferencial captura biopotenciales, la tensión del paciente respecto a la tierra del amplificador es llamada tensión en modo común, y ya que esta tensión puede ser transformada por el amplificador en una señal de interferencia diferencial, es recomendable conectar un tercer electrodo al paciente. Este electrodo provee una ruta de baja impedancia entre el paciente y la tierra del amplificador y por lo tanto la tensión en modo común es pequeña. No es aconsejable conectar el electrodo directamente a la tierra del circuito por dos razones: 1) si el circuito no está aislado, corrientes altas podrían pasar a través del paciente. 2) un mal contacto del electrodo de referencia puede significar una resistencia de hasta 100k $\Omega$  entre el paciente y tierra [4].

El uso más común y efectivo del tercer electrodo es conectarlo al conocido circuito de pierna derecha (*driven-right-leg circuit*), este circuito resuelve los problemas mencionados anteriormente. Reduce la resistencia efectiva del electrodo en varios órdenes de magnitud, y permite que circule únicamente una cantidad segura de corriente a través del tercer electrodo.

El electrodo de referencia es localizado en un tejido eléctricamente neutro (es decir, una prominencia ósea). Y es importante que el electrodo de referencia haga un buen contacto

eléctrico con la piel. Por esta razón el electrodo puede ser grande (2cm X 2cm). Si es pequeño el material debe ser altamente conductor y debe tener un adhesivo fuerte para asegurarlo a la piel con considerable estabilidad. Los geles eléctricamente conductores son particularmente adecuados para este propósito. Frecuentemente el ruido de la línea de potencia es atenuado o eliminado con un buen posicionamiento del electrodo de referencia.

#### 1.8. Conceptos de seguridad eléctrica

La falla de cualquier sistema electrónico en contacto galvánico directo o indirecto con la piel, puede causar que pase corriente a través de la piel del sujeto. Este concepto es menos relevante si se trata de dispositivos alimentados exclusivamente por baterías de baja tensión (3-15V). Para garantizar la seguridad, el sujeto puede ser eléctricamente aislado de una conexión eléctrica (línea de alimentación o tierra) asociada con la fuente de tensión. Este aislamiento generalmente es alcanzado de dos formas: ya sea a través del uso de aisladores ópticos o a través de transformadores de aislamiento. Los dos enfoques son recomendados, pero ambos requieren consideraciones cuidadosas para no distorsionar la señal EMG. Esto especialmente cierto cuando se usa transformadores. Este aislamiento además da otro beneficio; reduce la cantidad de ruido asociado a la red eléctrica.

#### 1.9. Interfaz gráfica

Una interfaz gráfica se usa básicamente para hacer sencilla la comunicación entre un usuario y el sistema operativo de un computador. Pero además de esto, una interfaz gráfica permite la interacción con un dispositivo electrónico. Este tipo de interfaz es conocido como GUI (*graphical user interface*) la cual permite esta interacción a través de iconos, gráficos e indicadores visuales [5].

Se deben tener las siguientes consideraciones para plantear una interfaz, todo esto bajo la premisa de que se está desarrollando una interfaz para pacientes sin movilidad alguna, sólo el parpadeo.

Enfocarse en el usuario y sus tareas, no en la tecnología - Es necesario enfocarse en el usuario final, hacerse preguntas como, ¿Para quién está diseñado este software?, ¿Quiénes son los usuarios? , ¿Para qué es el software?, ¿En qué actividad se enfoca?, ¿Cuáles de los problemas que tienen los usuarios serán resueltos?, ¿Que problemas tiene el usuario ahora?, ¿Que le gusta y que no le gusta de cómo están haciendo la actividad actualmente?, ¿Cuáles son las habilidades y conocimientos de los usuarios? [6]

Todo esto para saber cómo construir la interfaz, y tener conciencia de hacia quiénes va enfocada.

Considerar las funciones primero, luego la presentación – En el caso de una interfaz de usuario es muy importante la funcionalidad, la forma en que el usuario final interactúa y

navega a través de esta, por lo tanto resulta secundaria la parte gráfica (colores fuentes y demás).

Ajustarse a cómo ve el usuario la tarea que debe realizar - Se diseña respecto al punto de vista del usuario final, y para esto es necesario conocerlo.

Diseño centrado en el caso más común - Lo mejor es hacer que la tarea realizada más común sea fácil. Por ejemplo, para el caso de pacientes con cuadriplejia, el tener predeterminadas frases comunes como, “debo ir al baño”, “tengo hambre”, “me pica un ojo”, etc. O tener una opción que le permita al paciente registrar sus propias palabras, y de esta manera tener una memoria de frases donde el paciente llame la frase sin necesidad de escribirla de nuevo.

Facilitar el aprendizaje - Una de las quejas que se tiene como usuario de aplicaciones de software es que son difíciles de entender o toma mucho tiempo hacerlo. Por esto es necesario no asumir que el usuario ya sabe cómo usar el aplicativo, sino que se diseñe de tal forma que pueda ser más perceptivo, usando íconos y señales que permitan entender cómo usarlo.

Entre las empresas dedicadas a desarrollar herramientas que permiten diseñar interfaces gráficas, Microsoft propone tecnologías que permite diseñar a diferentes niveles en el propio lenguaje fuente del programador y que puede ser usado por la mayoría de aplicaciones creadas para la plataforma Windows. Una de estas tecnologías es WPF, que ofrece diversidad para hacer aplicaciones gráficas, y al ser propia de Microsoft, puede ser usado *Microsoft Visual Studio* como entorno de desarrollo, y como motor de desarrollo *.NET*, también propio de esta empresa.

*Windows Presentation Foundation (WPF)* - Es una tecnología de Microsoft que está incluida en Windows 7, Windows Vista, Windows Server 2008 y también está disponible para Windows XP *Service Pack 2* o posterior y Windows Server 2003. Esta permite el desarrollo de interfaces con la habilidad de interactuar con Windows. También toma características de aplicaciones de Windows y aun de aplicaciones web [7].

WPF ofrece una amplia infraestructura y potencia a nivel gráfico con la que se puede desarrollar aplicaciones atractivas visualmente, y que cuentan con amplias facilidades de interacción en las que se incluyen animación, vídeo, audio, documentos, navegación y gráficos 3D. Utiliza el lenguaje declarativo XAML y los lenguajes de programación de *.NET*, proveyendo al desarrollador un kit de herramientas para fácilmente personalizar la apariencia de controles existentes de Windows o para crearlos. [7]

#### *Microsoft .NET*

*.NET* es un software *framework*, es decir, es una abstracción en la que el software proporciona una funcionalidad genérica que se puede cambiar de forma selectiva adicionando código

específico del usuario, lo que proporciona la capacidad de generar software de aplicaciones específicas.

Este es desarrollado por Microsoft y corre principalmente en Windows. Incluye una gran cantidad de librerías y provee lenguaje inter-operacional (cada lenguaje puede usar código escrito en otros lenguajes) a través de varios lenguajes de programación.

La librería de clase básica de *.NET* provee interfaces de usuario, acceso a los datos, base de datos para conectividad, criptografía, desarrollo de aplicaciones web, algoritmos numéricos y comunicación entre redes. Los programadores producen software combinando su propio lenguaje fuente con *.NET* y otras librerías. *.NET* está destinado para ser usado por la mayoría de aplicaciones creadas para la plataforma de Windows.

Microsoft también desarrolló un entorno de desarrollo integrado adaptado para *.NET* llamado Visual Studio [8].

XAML (*eXtensible Application Markup Language*) como su nombre lo indica, es el lenguaje de formato para la interfaz de usuario para *Windows Presentation Foundation (WPF)*, uno de los pilares de la interfaz de programación de aplicaciones *.NET*

XAML es un lenguaje declarativo basado en XML, que ha sido optimizado para descripción gráfica de interfaces de usuario robustas desde el punto de vista gráfico, tales como las creadas por medio de Adobe Flash.

En su uso típico, los archivos tipo XAML son producidos por una herramienta de diseño visual, como *Microsoft Visual Studio* (usada para este proyecto) o *Microsoft Expression*. El XML resultante es interpretado en forma instantánea por un sub-sistema de despliegue de Windows Vista que reemplaza al GDI de las versiones anteriores de Windows. Los elementos de XAML se interconectan con objetos del Entorno Común de Ejecución para Lenguajes, que para este caso en específico es *.NET*. Los atributos se conectan con propiedades o eventos de esos objetos.

#### *1.10. Comunicación USB*

Para implementar la comunicación USB se usa el microcontrolador PIC18f2550 de microchip de 32 KB de memoria programable y 10 canales ADC. Cuenta con un puerto serial FS-USB (12 Mbit/s). También posee grandes cantidades de memoria RAM (RAM de 1K byte de puerto doble) para *buffering* y memorias programables FLASH mejoradas que lo hacen ideal para aplicaciones que requieren conexión periódica con un ordenador personal a través de USB para cargar y descargar datos [9].

Características de la Interface Full Speed USB 2.0 (12Mbit/s)

- 1) 1K byte Dual Port RAM + 1K byte GP RAM
- 2) Full Speed Transceiver
- 3) 16 Endpoints (IN/OUT)
- 4) Resistencias de Pull Up internas (D+/D-)
- 5) Rendimiento de 48 MHz (12 MIPS)

### 1.11. Interfaz de Comunicación

Existen tres clases que componen la definición para la comunicación de dispositivos: la clase de dispositivo de comunicación CDC, la clase de interfaz de comunicación y la clase de interfaz de datos. CDC es un nivel de definición del dispositivo y es usado por el *host* para identificar un dispositivo de comunicación que puede presentar diferentes tipos de interfaces. La clase de interfaz de comunicación define un mecanismo de propósito general que puede ser usado para habilitar todos los servicios de comunicación USB. La clase de interfaz de datos define un mecanismo de propósito general que habilita la transferencia tipo *bulk* o isócrona por USB cuando los datos no cumplen los requerimientos de cualquier otra clase [10].

### 1.12. Operación del Dispositivo

Los dispositivos de comunicación presentan los datos al *host* en una forma definida por clases, por ejemplo audio, datos o una interfaz HID. Para permitir que el controlador de clase apropiado maneje estos datos, el *host* es presentado con una o más interfaces, que son específicas para cada clase. Para permitirle al *host* reconocer apropiadamente el tipo de clase donde múltiples interfaces son usadas para crear una simple función, el dispositivo puede identificarse opcionalmente con el código CDC (*Communication Device Class code*). Esto le permite al *host* si es necesario, cargar cualquier controlador especial para configurar apropiadamente las múltiples interfaces en una sola función en el *host*.

Las características estáticas del dispositivo, como las conexiones físicas, son descritas en términos del dispositivo USB, la interface y los *endpoints* descriptores. Mientras que los datos que se mueven por las interfaces físicas, que por naturaleza son dinámicos, causan que las características de las interfaces cambien a medida que los requerimientos de los datos cambian. Estos cambios dinámicos son definidos en términos de mensajes transmitidos entre el dispositivo y el *host* mediante la interfaz de clase de comunicación.

### 1.13. Puerto Virtual COM mediante comunicación CDC [11]

Durante la enumeración, el *firmware* (código base en el microcontrolador) responde a la petición de descriptores que identifican la función del dispositivo CDC. Todo dispositivo USB contiene una serie de descriptores que contienen información de las prestaciones de los dispositivos.

Las especificaciones para USB 2.0 definen el contenido y el formato de los descriptores USB estándar. Las especificaciones para CDC definen descriptores adicionales de clase específicos.

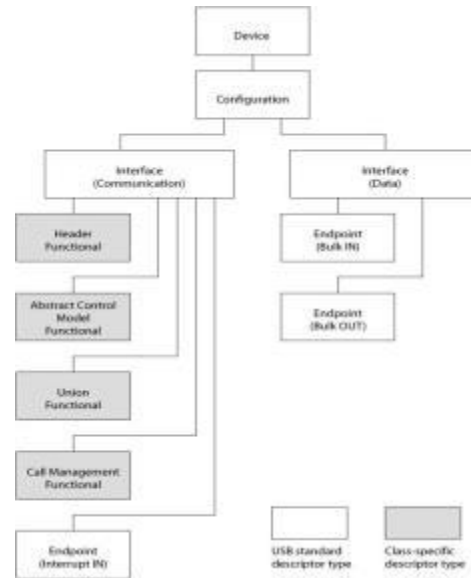


Figura 3 Puerto COM virtual mediante USB [12]

La Figura 3 muestra los descriptores en un dispositivo típico CDC que funciona como un puerto COM genérico virtual. Todo dispositivo USB debe tener un *Device Descriptor*, un *communication interface descriptor*, y un *Data interface descriptor*, que deben estar definidos en el *firmware* del microcontrolador.

El *device descriptor* nombra la clase de dispositivo de comunicación (CDC), el código de la clase específica CDC como la clase del dispositivo, el ID de proveedor y los valores de ID de Producto que identifican al dispositivo específico. El ID del proveedor y el ID del producto (VID/PID) especificado en el *firmware* deben ser los mismos definidos en el controlador del *host*.

Este descriptor también contiene un número serial que retiene el número del puerto COM. En el caso que se conecte en otro puerto USB del computador. Un dispositivo que no contenga un número serial, obtiene un nuevo número de puerto cada vez que se conecte a un puerto diferente del PC. Además de esto, un dispositivo CDC de puerto COM virtual tiene un descriptor de configuración y dos descriptores de interface. El descriptor de configuración especifica los requerimientos de potencia y el número de interfaces en la configuración. Los descriptores de interface le dicen al *host* como el dispositivo implementa sus funciones de comunicación.

El *communication interface descriptor* nombra una subclase CDC y el protocolo. También provee un *endpoint* de interrupción para enviar notificaciones al *host* USB.

La interface de comunicación tiene cuatro descriptores específicos de clase y un *endpoint* descriptor. El *header functional* descriptor nombra la versión de la especificación CDC que cumple con la interfaz. El *abstract control model descriptor* especifica que notificaciones y peticiones de clase el dispositivo soporta. El *union functional descriptor* identifica las interfaces que pertenecen a la función CDC, que generalmente son la interfaz de comunicación más una interfaz de datos. Y el

management functional descriptor dice cómo el dispositivo administra las llamadas.

Y finalmente el *data interface descriptor* es el responsable del envío y la recepción de los datos del puerto COM.

## 2. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

### 2.1. Captura y adecuación de la señal

Tomando en cuenta las características de la señal de electromiografía ocular, y las posibles distorsiones a que esta se puede ver expuesta, se plantea un diseño para obtener la señal en la forma necesaria para el propósito planteado. En este caso se seleccionó un diseño analógico utilizando circuitos integrados, que se vale de diferentes etapas. Como se puede observar en la Figura 4, cada una de estas, cuenta con una tarea específica dentro del objetivo final. La preamplificación diferencial, además de eliminar en su gran mayoría el ruido capturado por los electrodos, tiene una ganancia baja para no amplificar demasiado los ruidos que están presentes en el medio y que pueden saturar el sistema. La etapa de filtrado elimina los componentes frecuenciales por fuera del ancho de banda de la señal EMG, o aquellos que contienen grandes cantidades de ruido (como 60Hz). Y la amplificación final entrega la señal con una amplitud adecuada para su procesamiento posterior en el micro-controlador.



Figura 4 Captura y adecuación de la señal. Fuente: autores.

### 2.2. Preamplificación diferencial

Para realizar la etapa de preamplificación se utilizó un amplificador de instrumentación, se tuvieron en cuenta distintas características como: Impedancia de entrada en modo común, impedancia de entrada en modo diferencial, relación de rechazo en modo común (CMRR). El circuito integrado seleccionado fue el INA128 que presenta las características mostradas en la Tabla 1.

	$Z_m$	$Z_d$	CMRR
Valor	$10e10/2$ $\Omega/pF$	$10e11/9$ $\Omega/pF$	120 dB (min)

Tabla 1 Características INA128. [13]

### 2.3. Circuito del tercer electrodo (referencia)

El esquema para la conexión del circuito del tercer electrodo en realimentación activa que se utilizó se muestra en la Figura 5.

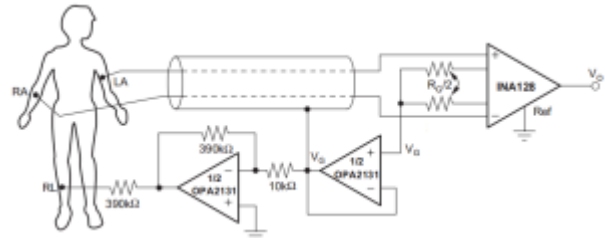


Figura 5 Esquema del circuito del tercer electrodo [13]

### 2.4. Filtrado

Después de realizar la adquisición y la primera preamplificación se debe realizar el filtrado de la misma para eliminar componentes frecuenciales no deseados. Para la elección de la respuesta deseada del filtro se deben tener en cuenta las características frecuenciales de la señal y las respuestas que ofrecen los distintos tipos de filtros. Como se mostró anteriormente, la señal producida por los músculos orbiculares es una señal de tipo EMG, con un rango de frecuencias que van desde 20Hz hasta 500Hz, y energía uniformemente distribuida en todo el espectro. Ya que las principales fuentes de ruido en este tipo de diseño se encuentran en 60Hz (red eléctrica) y en el intervalo 0-20Hz (tensión entre la superficie del electrodo y la piel y ruido por movimientos de cables), el rango de frecuencias adecuado para esta señal se encuentra de 70 a 500Hz. Por esto, se optó por un filtro pasa-altas con frecuencia de corte de 70Hz de tipo Chebyshev con un corte agudo y una banda de paso con ondulaciones como se muestra en la Figura 6 pues se necesitaba atenuar lo más posible el ruido de 60Hz y porque interesaba más la magnitud de la señal resultante que su forma. Se conecta en cascada un filtro pasa-bajas con frecuencia de corte 400Hz como se muestra en la Figura 7 de tipo Chebyshev para atenuar todo tipo de ruido en frecuencias altas, y para limitar el espectro de frecuencia. Usando el programa filterPro<sup>6</sup> se diseñaron los filtros de acuerdo a las características anteriormente expuestas.

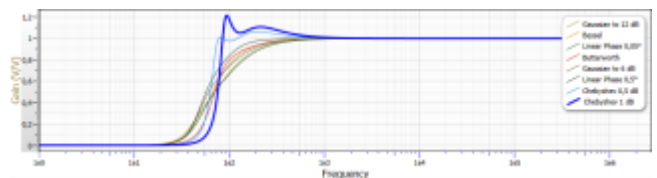


Figura 6 Respuesta filtros pasa altas con frecuencia de corte 70 [Hz]. Fuente: Autores.

<sup>6</sup> Programa con licencia gratuita obtenido en <http://www.ti.com/tool/filterpro>

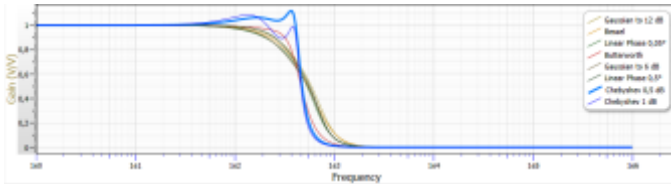


Figura 7 Respuesta filtros pasa bajas con frecuencia de corte 400 [Hz]. Fuente: Autores.

2.5. Amplificación

La amplificación final se realiza como forma de complementar la pre-amplificación y de esta forma llevar la señal a amplitudes deseables para su digitalización. En este caso, dadas las características de amplitud de la señal muscular, expuestas anteriormente, se realizó una preamplificación diferencial de 51V/V y una amplificación final variable de 0 a 200V/V. Se seleccionó una configuración no inversora utilizando el amplificador operacional OPA4227.

2.6. Etapa de aislamiento del paciente

El objetivo principal de esta etapa es brindar seguridad eléctrica a los usuarios del dispositivo, y blindarlo contra posibles fallas asociadas a la alimentación (puerto USB de un computador convencional) que lo puedan dejar expuesto a corrientes altas a través de la piel.

La etapa de aislamiento se diseñó para realizar dos tareas, la primera consiste en alimentar el circuito de captura de biopotenciales con un voltaje dual a partir de los 5V del puerto USB, la segunda es recibir la señal EMG del circuito y comunicarla al microcontrolador.

Se implementó usando convertidores DC/DC aislados no regulados (DCP02 series, Figura 8) de la empresa Texas Instruments y optoacopladores. Los convertidores DC/DC alcanzan tensiones de aislamiento de 1kV, requieren una mínima cantidad de componentes externos, vienen en un empaquetado pequeño y trabajan niveles de potencia baja lo cual los hace adecuados para esta aplicación, el DCP010505 se encarga de generar una tensión dual aislada de  $\pm 5V$  a partir de 5V a la entrada y el optoacoplador se encarga de transmitir la señal de dos estados en forma aislada al microcontrolador.

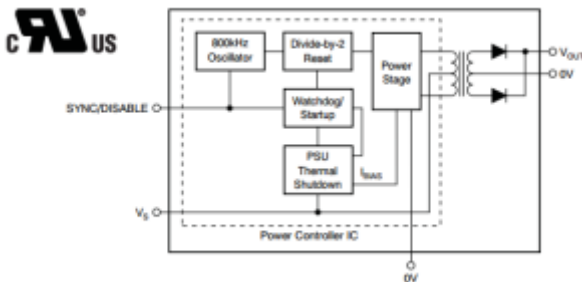


Figura 8 Diagrama del circuito convertidor DC/DC [14]

2.7. Implementación de la interfaz gráfica

El proyecto se realizó usando WPF (*Windows Presentation Foundation*), ya que esta tecnología ofrece una gran versatilidad al momento de hacer aplicaciones gráficas. Al ser propia de Microsoft, se utilizó como entorno de desarrollo Microsoft Visual Studio y como motor de desarrollo .NET. El diagrama de la figura 9 muestra la arquitectura del proyecto.

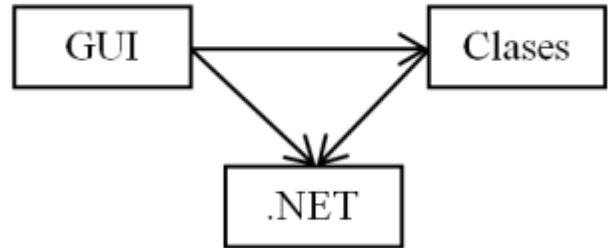


Figura 9 Arquitectura del proyecto. Fuente: Autores.

El componente GUI agrupa el entorno grafico de la aplicación, mientras que en el componente llamado clases, se encuentra la lógica del proyecto. Estos dos elementos utilizan .NET para compilarse y crear el programa resultante.

GUI - Como se mencionó anteriormente el entorno grafico esta soportado por WPF. Esta tecnología se apoya en el lenguaje declarativo XAML, este lenguaje permite organizar el código de una manera legible y fácil de mantener.

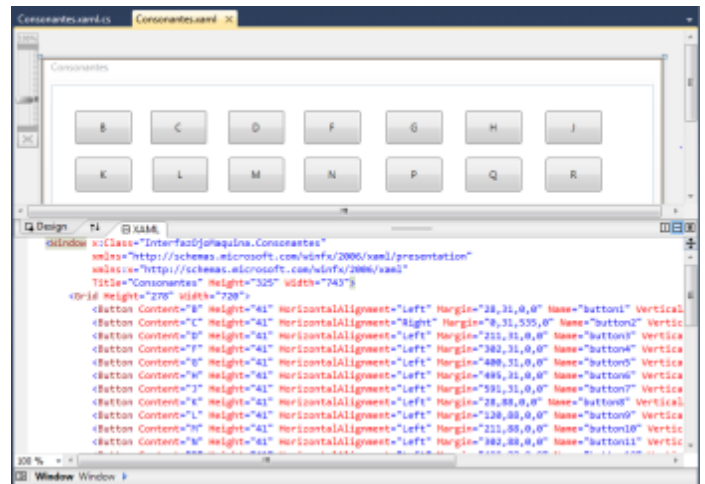


Figura 10 Entorno gráfico ventana de consonantes. Fuente: Autores.

El entorno gráfico mostrado en la Figura 10 es el correspondiente a la ventana de las consonantes, esta ventana está compuesta por etiquetas y propiedades. Con el uso de las etiquetas, se definen los elementos que van a hacer parte de la ventana y con las propiedades se especifica las características que van a tener.

Las ventanas del programa están compuestas principalmente por botones. Estos botones se detallan por la etiqueta *Button* y las características de los botones se especifican por propiedades

como *Content* (Contenido), *Height* (Altura del boton) entre otros.

Clases - En este componente se encuentran las aplicaciones que permiten la interacción hombre-máquina. También se encuentra toda la lógica del programa, esta comprende la comunicación (dispositivo electrónico con el computador y su posterior interpretación por el programa) y la traducción del entorno gráfico para que el sistema comprenda las acciones del usuario. El desarrollo de las aplicaciones de este componente se encuentran soportadas por el lenguaje de programación C Sharp (C#). Con este lenguaje se aplica el paradigma orientado a objetos y permite una integración con .NET.

Al trabajarse con elementos gráficos se deben usar eventos. Estos son los que permiten capturar las acciones del usuario y darle órdenes al sistema. Estas órdenes son las que se programan usando el lenguaje de programación C #.

Para el sintetizador de voz que permite reproducir un audio correspondiente al texto escrito en inglés, se utilizan las funciones que vienen integradas con Windows. El programa utiliza la clase del sistema *SpeechSynthesizer* y la función asociada que es llamada *SpeakAsync*. Esta función recibe un parámetro *string* que usa la función para luego convertirlo a audio.

.NET - Con .Net se ensamblan todos los componentes anteriores para crear un programa que será interpretado por el sistema operativo Windows. El resultado es el mostrado en la Figura 11:

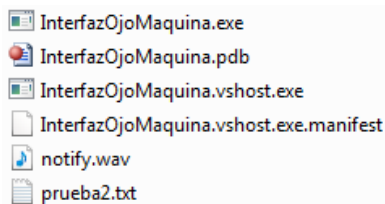


Figura 11 Archivos generados de la interfaz. Fuente: Autores.

Se plantearon dos tipos de interfaz. La primera con el mismo orden del arreglo de un teclado convencional, y la segunda con un orden propuesto para optimizar el tiempo de escritura. Todo esto para proveer un entorno de navegación sencillo, sin complicaciones y que le permita al usuario escribir en el menor tiempo posible.

### 2.8. Primera propuesta de interfaz gráfica

Como primera propuesta se planteó un teclado convencional, usando la variante alemana QWERT la cual tiene una disposición de teclas según el uso más frecuente de las letras. En este caso se añade la letra ñ, para el castellano.



Figura 12 Presentación inicial. Fuente: Autores.

Presentación inicial (Figura 12)- Tiene dos opciones, la de escribir una palabra y/o frase. Y la de desplegar frases previamente guardadas por el usuario.

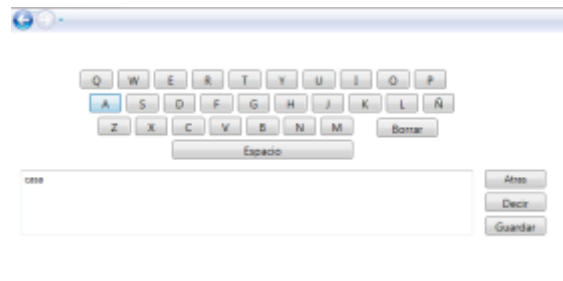


Figura 13 Ventana Principal. Fuente: Autores.

Ventana principal de escritura (Figura 13) - Si el usuario por medio del parpadeo elige la opción de escribir, se despliega el teclado alfanumérico predeterminado. Además de letras este contiene la función atrás, decir y guardar. En la opción atrás, vuelve a la presentación inicial, es decir si el usuario terminó la escritura, se reproduce un sonido que le indica a alguna persona cercana que el usuario tiene algo por decir. Para el caso que el usuario hable inglés se tendrá otro formato de interfaz, donde esta función enlaza el sintetizador de voz de Windows 7 con la plataforma para reproducir la frase que se haya escrito. Por último está la función guardar que le permite al usuario guardar la frase o palabra escrita.

Si se selecciona esta última opción, el software automáticamente guarda la frase en orden ascendente. Y en ese momento se devuelve a la presentación inicial.



Figura 14 Ventana de frases. Fuente: Autores.

Ventana de frases (Figura 14) - Por último, la función frases en la presentación inicial despliega la ventana que le muestra al usuario el directorio de palabras y frases que ha almacenado. Para seleccionar la frase, simplemente al costado izquierdo el usuario selecciona el número que representa una posición en la lista donde están las frases, y finalmente selecciona “decir”. Al hacerlo, se genera el mismo resultado para la opción decir en la ventana del teclado.

### 2.9. Segunda propuesta de interfaz gráfica

Esta interfaz se propone pensando en mejorar el tiempo de escritura del usuario. En cada presentación, el puntero se mueve a través de los iconos de forma autónoma y al recibir la señal accede al ícono en que se encuentre en ese momento.

Ventana principal (Figura 15) - En esta se selecciona si se quiere escribir una frase letra por letra, desplegar una frase guardada o simplemente guardar la frase que se encuentra en ese momento en el editor (barra de escritura).

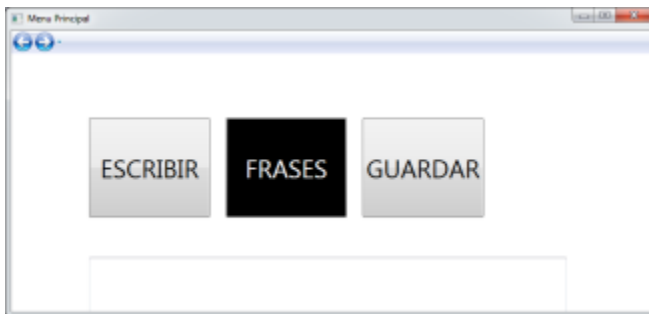


Figura 15 Ventana principal. Fuente: Autores.

Ventana de guardar (Figura 16) - Al seleccionar “guardar” la plataforma pide la confirmación y guarda la frase en la memoria en orden ascendente, luego regresa automáticamente a la ventana principal.

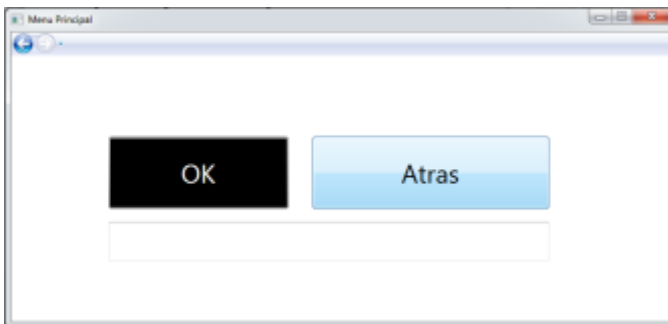


Figura 16 Ventana guardar. Fuente: Autores.

Ventana de frases (Figura 17) - Al activar la ventana de frases pide el número asociado a la frase que se quiere desplegar en el editor. Al desplegarla, regresa automáticamente a la ventana principal.

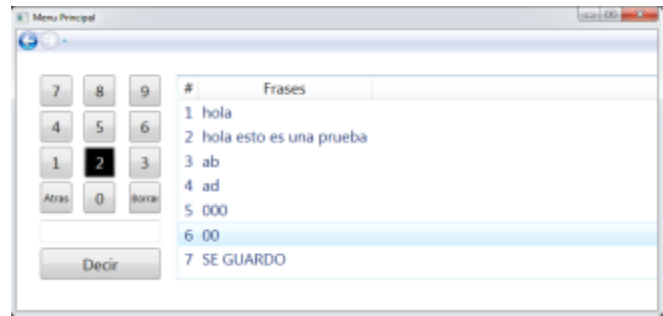


Figura 17 Ventana de frases. Fuente: Autores.

Ventana principal de escritura (Figura 18) - Esta página redirecciona a un teclado de vocales, a un teclado de consonantes y a un teclado con las herramientas de escritura más usadas (borrar letra, borrar palabra y borrar todo o salir a la ventana de escritura o a la ventana principal) se debe tener en cuenta que en el proceso de escribir se intercalan vocales y consonantes y dado que el número de vocales es menor, su acceso debe ser más rápido.



Figura 18 Ventana principal de escritura. Fuente: Autores.

Ventana de vocales (Figura 19) - Al escribir una vocal regresa a la ventana principal de escritura.

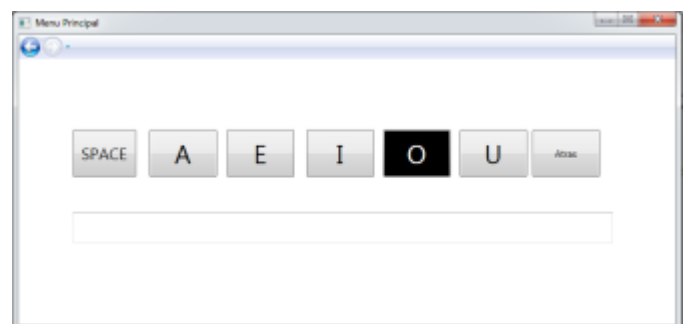


Figura 19 Ventana de Vocales. Fuente: Autores.

Ventana de consonantes (Figura 20)- Al escribir una consonante regresa a la ventana principal de escritura.



Figura 20 Ventana de consonantes. Fuente: Autores.

Ventana de herramientas (Figura 21) - Al realizar una acción, excepto la opción "Ventana Principal" regresa a la ventana principal de escritura.



Figura 21 Ventana de herramientas. Fuente: Autores.

## 2.10. Implementación de la comunicación por puerto USB [15]

Para la programación del microcontrolador PIC18f2550, se utilizó el compilador C18, y el *firmware* fue basado en el ejemplo para comunicación CDC-USB de los aplicativos de *Microchip Solutions* v2010-10-19 [15].

Descripción de los Archivos en el *firmware* - El *stack* USB requiere que existan en la carpeta del proyecto los 3 siguientes archivos:

**HardwareProfile.h:** El *stack* está pensado para utilizarlo con los demostradores de Microchip, pero como se va a usar un prototipo diferente se crea este archivo dentro de la carpeta con un contenido mínimo.

**usb\_desciptors.c:** Este archivo contiene los PID y VID con los que el sistema identifica el dispositivo electrónico conectado al puerto USB. Estos deben ser los mismos PID y VID del controlador en el *host*.

**usb\_config.h:** Este archivo establece algunas constantes que condicionan el funcionamiento del *stack*. Se usa en modo polling que permite llamar a `USBDeviceTasks()` en una prioridad más alta.

Además de estos 3 archivos, se necesitan los archivos `usb_device.c` y `usb_function_cdc.c`, que contienen las especificaciones y funciones para la comunicación USB.

Explicación del código - Inicialmente se declara uno de los canales del microcontrolador como puerto conversor Análogo/Digital, y seguido de esto se declara una variable que almacenará el número de 10 bits del registro del conversor. Esta variable servirá como condicional para el evento de parpadeo en la interfaz gráfica.

Seguidamente, para la comunicación USB se empieza con un `USBDeviceInit` que inicializa el dispositivo USB apoyándose en las tres funciones *Callback* que hay al final del código.

A continuación simplemente se entra en un bucle infinito que llama en cada iteración a las funciones `USBDeviceTasks()` y `ProcessIO`. La primera debe llamarse periódicamente para poner a punto el estado de la comunicación USB, y en `ProcessIO` en primer lugar se comprueba si el dispositivo está aún conectado y entonces comprueba si está listo para transmitir. En caso afirmativo se compara simultáneamente si el registro donde se almacenó la entrada del puerto ADC es mayor al valor de comparación (valor que corresponde al parpadeo) y si esto se cumple, se devuelve al *buffer* de salida la palabra "mayor" con `putsUSBUSART`. Esta palabra que por medio del puerto virtual COM recibe el programa de la interfaz gráfica, representa la señal de control. Por último, se llama a `CDCTxService()` que es la responsable de enviar el *buffer* por el puerto USB de vuelta al ordenador.

Funciones *callback* - El *stack* del *firmware* USB llama las funciones *callback* `USBCBxxx()` en respuesta a ciertos eventos relacionados con la comunicación USB. Por ejemplo, si el *host* PC se está apagando, el parará enviando un paquete *SOF* (*start of Frame*) al dispositivo. En respuesta a esto, todos los dispositivos USB deberían reducir su consumo de potencia a menos de 2.5 mA. El módulo USB detecta esta condición y entonces llama la función `USBCBSuspend()`.

Las funciones *callback* que establecen la comunicación son:

`USBCBCheckOtherReq(void)`

Esta función es llamada cuando desde el *host* son enviados paquetes de *setup*. Algunos *firmware* deben procesar la petición y responder apropiadamente para cumplirla.

`USBCBInitEP(void)`

Se llama a esta función cuando los dispositivos van a ser inicializados.

`BOOL`

`USER_USB_CALLBACK_EVENT_HANDLER(USB_EVENT event, void*pdata, WORD size)`

Esta última función se llama desde el *stack* USB para notificar a una aplicación de usuario que un evento USB ocurrió [15].

## 3. RESULTADOS

### 3.1. Circuito para la captura de biopotenciales

A continuación se muestran las señales adquiridas por el dispositivo de EMG: En la Figura 22 se puede observar la señal EMG asociada al parpadeo del paciente, sus características son

las propias de este tipo de señal, con valores aleatorios que van de tensiones positivas a negativas, y amplitud pico en el orden de 4V, la señal fue amplificada por dos etapas, una amplificación inicial de 51V/V y una amplificación final de 100V/V con lo cual se tiene una amplificación total de 5100V/V. Ya que se necesitó sensar esta señal con el microcontrolador y este únicamente acepta valores positivos de tensión en el rango de 0-5V, se rectificó como se muestra en la Figura 23, eliminando los valores negativos de tensión, y posteriormente se hizo pasar por un integrador con el objetivo de obtener el contorno de la señal como se muestra en la Figura 24, haciendo la medición más fácil al microcontrolador.

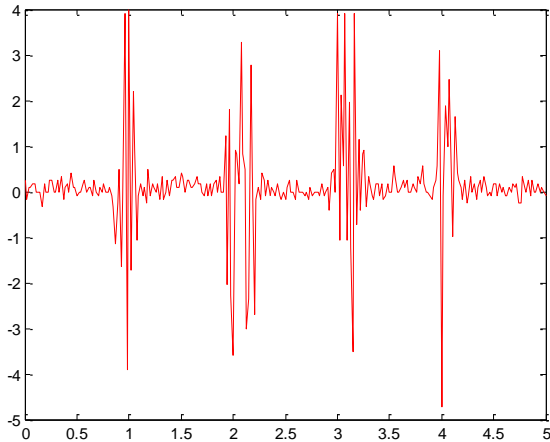


Figura 22 Señal EMG asociada a la acción del parpadeo. Fuente Autores.

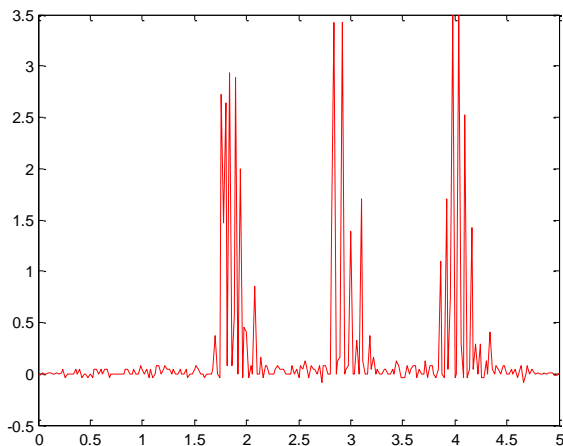


Figura 23 Señal EMG rectificadora. Fuente Autores.

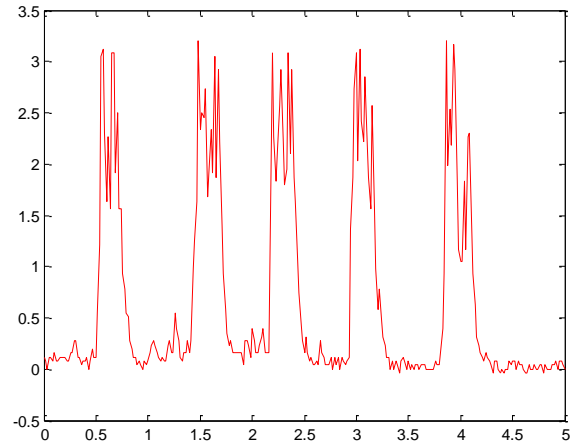


Figura 24 Señal EMG rectificadora e integrada.

Para el caso de nuestro prototipo era importante brindar precisión al momento de capturar la señal, diferenciando un parpadeo voluntario, es decir, forzado de uno involuntario o reflejo. Esta distinción se facilitó gracias a que el parpadeo reflejo maneja contracciones más débiles que implican una señal EMG más pequeña. En la Figura 25 se puede ver un comparativo entre la señal forzada (izquierda, hasta 3.5s) y refleja (derecha, de 3.5s en adelante).

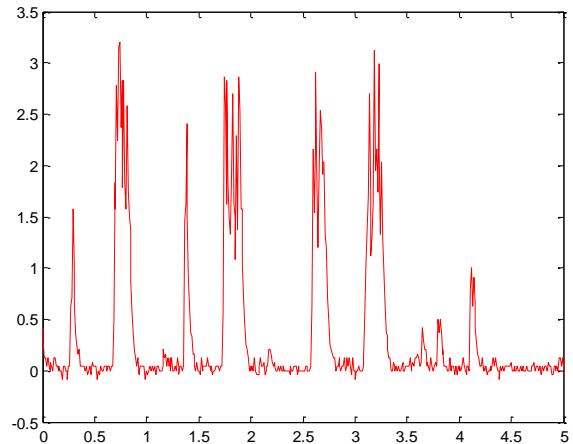


Figura 25 Intensidad de señal para diferentes tipos de parpadeo (voluntario e involuntario). Fuente: Autores.



Figura 26 Posición de los electrodos al utilizar el dispositivo. Fuente: Autores.

En la Figura 26 se puede observar la posición adecuada de los electrodos al momento de adquirir la señal. Se observa el electrodo de referencia ubicado en la frente y los electrodos diferenciales ubicados arriba y abajo del ojo, esta posición facilita la captura de la señal asociada al músculo orbicular del ojo.

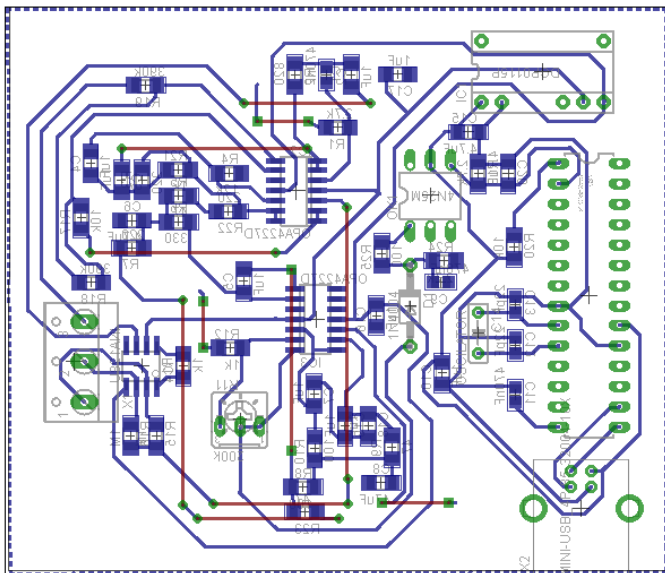


Figura 27 Enrutado del circuito. Fuente: Autores.

Se realizó un diseño que integra las diferentes etapas (captura, filtrado, amplificación, aislamiento y comunicación serial) en una misma placa, implementando todo en lo posible con elementos superficiales, para reducir de este modo el tamaño del dispositivo como se muestra en la Figura 27 y en la Figura 28.

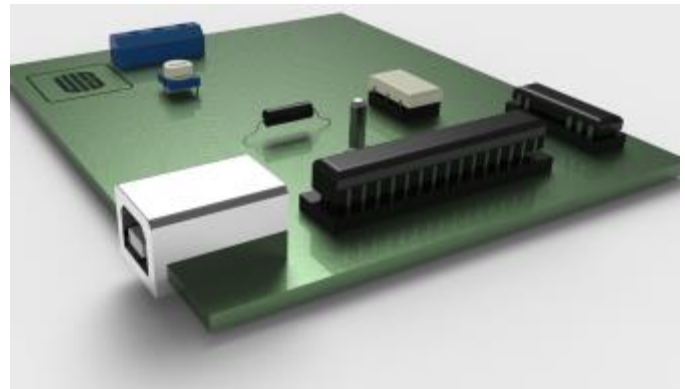


Figura 28 Placa del dispositivo. Fuente Autores.

Además se diseñó una carcasa teniendo en cuenta que la manipulación de la placa causa alteraciones a su funcionamiento (dados los potenciales estáticos presentes en las manos), brindando al circuito protección, y mejorando su presentación a los usuarios, como se muestra en la Figura 29.



Figura 29 Carcasa del dispositivo. Fuente Autores

La interfaz gráfica de usuario fue realizada bajo el principio de usabilidad, es decir, tareas claras, íconos perceptibles de fácil reconocimiento y con solo lo necesario para su correcto funcionamiento. Además de esto, se implementaron dos versiones de interfaz, una con un arreglo de letras idéntico a un teclado genérico y otra optimizando el acceso a las letras y funciones más usadas por el usuario. Al realizar la tarea de escribir una frase, fue notable la diferencia de eficiencia de la interfaz propuesta por los autores en comparación con la del teclado convencional. Pues esta primera permite escribir de manera más rápida al dar un acceso selectivo.

También se logra implementar una memoria de frases que puede ser actualizada a gusto del usuario cuando así lo requiera. Con esta memoria es posible seleccionar frases que el usuario almacenó previamente y ubicarlas en el editor por si este desea agregarle algo más. Esta función le permitirá al usuario comunicarse eficientemente sin invertir tiempo volviendo a escribir frases previamente almacenadas.

Por último se logra sincronizar el sintetizador de voz ofrecido por Microsoft con el programa. De esta manera los

usuarios que escriban en inglés podrán tener la facilidad reproducir lo que previamente escribieron. Y así sin la necesidad de diseñar un sintetizador propio, se agrega un componente que no estaba contemplado dentro de los objetivos de este proyecto pero aporta mayor usabilidad e innovación.

#### 4. CONCLUSIONES Y RECOMENDACIONES

El sistema propuesto está orientado hacia el desarrollo de una interfaz cerebro-computador útil y de bajo costo para discapacitados y personas mayores. Se puede utilizar como un medio de control que permite a los minusválidos, especialmente los que tienen sólo la coordinación motora visual, interactuar de forma independiente. El parpadeo requiere un mínimo esfuerzo y permite técnicas de selección directa, esto aumenta el tiempo de respuesta y la velocidad del flujo de información.

Se logró la captura y adecuación de la señal EMG asociada al parpadeo, reduciendo significativamente el ruido debido a interferencias del ambiente (todo esto de forma analógica). Con el objetivo de que la señal fuera acorde a los requerimientos del conversor análogo digital en el microcontrolador. Seguidamente se diseñó e implementó un sistema de aislamiento eléctrico entre el paciente y la alimentación del circuito (en este caso el puerto USB de la computadora), haciendo confiable la utilización del dispositivo para el usuario. Finalmente se desarrolló una interfaz gráfica de escritura inspirada en el proceso de comunicación con un paciente real, que funciona a partir de una señal digital de dos estados.

Como recomendaciones para trabajos futuros se plantea: Realizar el proceso de filtrado, amplificación final y adecuación de la señal (rectificador de media onda e integrador) de forma digital, ya sea a través de un microcontrolador o de un software instalado en la computadora como LABVIEW o MATLAB, reduciendo ostensiblemente el tamaño del dispositivo. Diseñar un dispositivo para la captura de biopotenciales tipo auricular, que resulte más cómodo y amigable con el paciente y cuya conexión se realice de forma inalámbrica con la computadora, para facilitarle los desplazamientos a este. Implementar un procesamiento digital que mejore en lo posible la distinción entre parpadeo forzado y parpadeo reflejo mejorando la precisión del sistema.

La interfaz gráfica puede ser adaptada a otras señales corporales controladas de dos estados, como por ejemplo el movimiento de un dedo o de la boca, o en su defecto realizar procesos de adquisición de la señal sin el uso de electrodos, como procesamiento digital de imágenes.

#### REFERENCIAS

- [1] Chris Jarmey, *Atlas conciso de los músculos*. Barcelona, España: Paidotribo, 2008.
- [2] Richard L. Drake, Adam M. W. Mitchell, and A. Wayne Vogl, *Gray. Anatomía para estudiantes*. España: Elsevier, 2010.
- [3] Carlo J. De Luca. (2002) delsys.com. [Online]. [http://www.delsys.com/Attachments\\_pdf/WP\\_SEMGintro.pdf](http://www.delsys.com/Attachments_pdf/WP_SEMGintro.pdf) [citado el 28 de Enero de 2014]
- [4] Bruce B. Winter and J.G. Webster, "Driven-right-leg circuit design," *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, vol. BME-30, no.1, pp.1-3, Junio 1983.
- [5] Wendy L. Martinez, "Graphical user interfaces," in *Wiley Interdisciplinary Reviews: Computational Statistics*.: Wiley, 2011, vol. 3, pp. 119–133.
- [6] Jeff Johnson, *GUI Bloopers 2.0: Common User Interface Design Don'ts and Dos*.: Elsevier/Morgan Kaufmann Publishers, 2008.
- [7] Andrew Moore, *Create Amazing Custom User Interfaces with Wpf, C#, and Xaml in .Net 3.0*.: wiley publishing, 2007.
- [8] Jordi Conesa Caralt, Àngels Rius Gavidia, Jordi Ceballos Villach, and David Gañán Jiménez, *Introducción a.net*, Primera edición en lengua castellana ed. Barcelona: UOC, Marzo 2010.
- [9] microchip.com. [Online]. <http://ww1.microchip.com/downloads/en/DeviceDoc/39632e.pdf> [citado el 28 de Enero de 2014]
- [10] <http://www.usb.org/>. [Online]. [http://www.usb.org/developers/devclass\\_docs/usbcdc11.pdf](http://www.usb.org/developers/devclass_docs/usbcdc11.pdf) [citado el 28 de Enero de 2014]
- [11] Jan Axelson, *USB MASS STORAGE*.: Madison.
- [12] lvr.com. [Online]. [http://www.lvr.com/usb\\_virtual\\_com\\_port.htm](http://www.lvr.com/usb_virtual_com_port.htm) [citado el 28 de Enero de 2014]
- [13] (2013, Oct.) Texas Instruments. [Online]. <http://www.ti.com/lit/ds/sbos051b/sbos051b.pdf> [citado el 28 de Enero de 2014]
- [14] (2000, Diciembre) Texas Instruments. [Online]. <http://www.ti.com/lit/ds/symlink/dcp010505b.pdf> [citado el 28 de Enero de 2014]
- [15] microchip. [Online]. [http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=2680&dDocName=en547784](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2680&dDocName=en547784) [citado el 28 de Enero de 2014]
- [16] Hari Singh Dhillon, Rajesh Singla, Navleen Singh Rekhi, and Rameshwar Jha, "EOG and EMG Based Virtual Keyboard: A Brain-Computer Interface," Jalandhar (Pb.), India, 2009.
- [17] Verónica Mondragón Merino and Josefina Lobera Gracida, *Discapacidad Motriz, Guía didáctica para la inclusión inicial y básica*. México D.F.: Dirección de medios y publicaciones, 2011.
- [18] Phil Kurz. broadcastengineering. [Online]. <http://broadcastengineering.com/ott/nangutv-feature-new-hd-gui-recommendation-engine> [citado el 28 de Enero de 2014]

- [19] Paul Leon Kaufman and Albert Alm, *Alder Fisiología del ojo*. España: Elsevier, 2003.
- [20] Venkatesh Acharya , "Improving Common-Mode Rejection Using the Right Leg," Julio 2011. [Online]. <http://www.ti.com/lit/an/sbaa188/sbaa188.pdf> [citado el 28 de Enero de 2014]
- [21] maestrosdelweb. [Online]. <http://www.maestrosdelweb.com/principiantes/la-historia-del-teclado-qwerty/> [citado el 28 de Enero de 2014]
- [22] Frank H. Netter, *Atlas de anatomía humana*, 4th ed. Barcelona, España: Elsevier MASSON, 2007.
- [23] Carlo J. De Luca. (2002) *delsysemg*. [Online]. [http://delsysemg.com/Attachments\\_pdf/WP\\_Biomechanics.pdf](http://delsysemg.com/Attachments_pdf/WP_Biomechanics.pdf) [citado el 28 de Enero de 2014]
- [24] Rafael Ricardo Rivera Méndez and Amín Juan Camilo Atencia, "Diseño e implementación de un electrooculógrafo (EOG) para el procesamiento y la captura de una señal ocular con el fin de controlar el puntero de un ordenador.," Bucaramanga, Trabajo de grado en Ing Electrónica, Universidad Industrial de Santander 2012.
- [25] Carolina Arboleda, Eliana García, Alejandro Posada, and Róbinson Torres, "Diseño y construcción de un prototipo de interfaz cerebro-computador para facilitar la comunicación de personas con discapacidad motora," *EIA*, vol. 11, pp. 105-115, Julio 2009.