



**UNIVERSIDAD
DE ANTIOQUIA**

**MEJORA Y REDISEÑO DE DOS PRÓTESIS DE
MIEMBRO SUPERIOR DE LA COMPAÑÍA E-
NABLE MEDELLÍN.**

Autor
Mauricio Zapata Contreras.

Universidad de Antioquia
Facultad De Ingeniería, Bioingeniería.
Medellín, Colombia
2020



MEJORA Y REDISEÑO DE DOS PRÓTESIS DE MIEMBRO SUPERIOR DE LA COMPAÑÍA E-NABLE
MEDELLÍN.

Mauricio Zapata Contreras.

Informe final presentado con la modalidad de práctica social presentado como requisito parcial
para optar al título de:

Bioingeniero

Asesores:

Jonathan Gallego Londoño. Mtro. Ingeniería Biomédica.

Hugo Esteban Rojas Muñoz. Bioingeniero.

Línea de Investigación:

Bioinstrumentación

Universidad de Antioquia

Facultad De Ingeniería, Bioingeniería.

Medellín, Colombia

2020.

AGRADECIMIENTOS

A mi familia y compañeros que me han ayudado a realizar este proyecto, de que si no fuera por ellos no estaría en culminando mis estudios de carrera profesional.

A mi asesor interno, Jonathan Gallego que me instruyó de la mejor forma posible y que si no es por su guía brindada hubiese sido muy difícil elaborar este proyecto en modalidad de práctica social.

A mi asesor externo, Esteban Rojas que me ayudó en los temas donde no poseo buenos fuertes y a darme opiniones y consejos acerca de las prótesis. Además de enseñarme muchos requisitos para lo que es la impresión 3D.

A la fundación E-Nable Medellín y sus empleados y voluntarios por darme una oportunidad de realizar mis prácticas allí y brindar mi ayuda a esta fantástica compañía sin ánimo de lucro.

A todos los llevaré muy adentro de mi corazón.

Contenido

1	Introducción	1
2	Objetivos	2
2.1	Objetivo General	2
2.2	Objetivos Específicos:.....	2
3	Marco Teórico.	3
3.1	PRÓTESIS.	3
3.2	ELECTROMIOGRAFÍA (EMG).....	3
3.3	SENSORES DE DISTANCIA	4
3.3.1	SENSORES DE DISTANCIA POR ULTRASONIDO.....	4
3.3.2	SENSORES DE DISTANCIA INDUCTIVOS.....	4
3.3.3	SENSORES DE DISTANCIA LASER	5
3.4	BRAZO ROBÓTICO “EL MEDALLO VERSION1”	5
3.5	HARDWARE Y SOFTWARE LIBRE.....	6
3.6	ARQUITECTURA MODELO-VISTA-CONTROLADOR (MVC)	6
3.6.1	MODELO	6
3.6.2	VISTA	6
3.6.3	CONTROLADOR	7
4	Metodología.....	7
4.1	RAMA I.....	7
4.2	RAMA II.....	8
5	Cronograma de Actividades.	9
6	Resultados y Análisis.	10
6.1	PROTOCOLO DE IDENTIFICACIÓN Y CALIBRACIÓN DE LA SEÑAL EMG	10
6.2	DISEÑO PCB	11
6.3	REDISEÑO DE PRÓTESIS CON DETECCIÓN DE DISTANCIA DE OBJETOS	14
6.4	DOCUMENTACIÓN DETALLADA DE LOS CAMBIOS REALIZADOS BAJO LA PLATAFORMA DE LA COMPAÑÍA E-NALBE MEDELLÍN	19
7	Conclusiones.....	21
8	Referencias bibliográficas	22
9	ANEXOS	24

TABLA DE FIGURAS

Figura 1. Prótesis de mano[5].	3
Figura 2. Trazos electromiográficos de los músculos [6].	3
Figura 3. Funcionamiento sensor ultrasónico [8].	4
Figura 4. Sensor de distancia inductivo [7].	4
Figura 5. Funcionamiento sensor de distancia laser [9].	5
Figura 6. Brazo robótico "El Medallo Version1" [10].	6
Figura 7. Diagrama de desarrollo de la metodología.	9
Figura 8. Vista entregada por el protocolo de la calibración.	10
Figura 9. PCB elaborado para la versión 1, capa superior-capa inferior.	11
Figura 10. PCB elaborada para la versión 2, capa superior-capa inferior.	12
Figura 11. Circuito realizado para el brazo robótico "El Medallo Version2".	13
Figura 12. Sensor de distancia GP2Y0D815Z0F.	14
Figura 13. Sensor de distancia VL6180X.	15
Figura 14. Acople del sensor para insertar en la parte superficial de la prótesis	16
Figura 15. Motor vibratorio tipo moneda	16
Figura 16. Circuito utilizado para sensor de distancia.	17
Figura 17. Esquema de funcionamiento del código de control de vibración.	17
Figura 18. Circuito implementado para brazo con detección de distancia.	18
Figura 19. Prótesis final de brazo detector de distancia.	18
Figura 20. Ilustración página de contenidos.	20
Figura 21. Ilustración página de inicio.	20

LISTADO DE TABLAS

Tabla 1. Cronograma de actividades.	10
Tabla 2. Parámetros para pedido en PCBWay.	13
Tabla 3. Tabla comparativa de ambos sensores de distancia.	15

TABLA DE ANEXOS

Anexo 1. Código utilizado para la calibración.	24
Anexo 2. Link para hacer pedidos de diseños PCB con la plataforma PCBWay.	27
Anexo 3. Archivos con diseños PCB "El Medallo V2".	27
Anexo 4. Archivos gerber para "El Medallo V2".	27
Anexo 5. Archivos con diseños PCB de brazo detector de distancia.	27
Anexo 6. Código utilizado para control de vibración.	28
Anexo 7. Links de código abierto elaborados durante este proyecto.	29

1 Introducción

En Colombia uno de los problemas más importantes de salud pública es el caso de personas amputadas y con discapacidad física en sus extremidades. Según la Asociación Colombiana de Medicina Física y Rehabilitación se estima que son entre 200 a 300 personas entre 100.000 que sufren de amputaciones [1]. Estos pacientes normalmente se encuentran en edades activas de trabajo o también son menores de edad, lo cual altera su calidad de vida y su ámbito social.

Además, el conflicto armado creado en la década de 1960 dio lugar a la creación de las minas antipersonas (MAP) y otro tipo de artefactos explosivos para neutralizar a las fuerzas militares para evitar su avance a territorios donde se encontraban las fuerzas revolucionarias. Debido a esto, en Colombia se tiene el mayor número de personas afectadas por este tipo de armas. Entre 1990 y 2008 se registraron en Colombia 3.674 accidentes y 8.785 incidentes debido a las MAP [2]. Es por esto que en Colombia se presenta una gran problemática acerca de la pérdida de las extremidades superiores e inferiores y donde hay personas que necesitan de una prótesis funcional para poder acoplarse al modo de vida del siglo XXI.

Con el fin de apoyar a las personas que hayan sido afectadas por la guerra civil o que hayan perdido una o ambas extremidades superiores por diferentes motivos nació en Colombia la empresa E-NABLE Medellín la cual es una compañía sin ánimo de lucro fundada en mayo de 2017 la cual se ha encargado de diseñar y entregar prótesis gratis y personalizadas a dichas personas. En esta compañía se desarrollan varios tipos de prótesis que el paciente puede escoger a su gusto y luego se hace una entrega gratuita que se acomode a las necesidades del paciente [3].

E-NABLE Medellín también realiza diseños de innovación y de aplicación de nuevas tecnologías para la elaboración de nuevas prótesis para así poder entregar un producto que se adapte mejor a las condiciones de cada paciente, uno de los nuevos diseños es un brazo robótico llamado “El Medallo Version1”, el cual es un brazo que funciona a partir de servomotores que hacen la apertura y el cierre de la mano controlados por la acción electromiográfica de la persona. Este brazo como tal aún está calificado como prototipo, pero se tiene la certeza de su correcto funcionamiento. La idea es generar un nuevo modelo de este prototipo para que cualquier paciente pueda escogerlo y hacerle su debida entrega como a cualquier otro de los modelos que presenta la compañía y así poder ayudar de una mejor forma a los pacientes con pérdida de una o ambas extremidades superiores. Para la elaboración de este brazo se hizo un trabajo durante aproximadamente dos años donde se analizaron varios tipos de electrodos para electromiografía (EMG), se han realizado varias

pruebas y procesamientos de la señal para poder obtener el correcto funcionamiento del brazo. Se ha realizado una ardua documentación para el montaje, impresión 3D y ensamblaje del mismo donde en esta se muestra el proceso paso a paso para la correcta realización de calibración de EMG. Estos pasos suelen ser complejos para una persona no entrenada en el tema porque requiere que el usuario debe tener idea de cómo realizar una comunicación serial con un microcontrolador y esto suele generar dificultades en el proceso de calibración del EMG. El proceso de momento funciona, pero no de forma práctica, ni eficiente y por lo tanto se desea mejorarlo.

Además, también se desea hacer un ajuste a uno de los brazos que se ofertan en la compañía para pacientes que no solo hayan perdido una extremidad superior, sino que también sean invidentes, usando un sistema de sensado de distancia para que ayude a identificar la presencia cercana de un objeto a su prótesis para así poder agarrar este objeto. Este tipo de prótesis, como se encuentra actualmente, no es eficaz porque el sensado de la distancia no es el adecuado y no cuenta con buena precisión, generando fallas a la hora de usarla, y por lo tanto es importante y necesario realizar el ajuste de medición de distancia para que pueda funcionar en dichos pacientes.

2 Objetivos

2.1 Objetivo General

Mejorar y rediseñar el proceso de identificación, calibración, ensamble circuital y detección de distancia de objetos de dos prótesis de miembro superior de la compañía E-NABLE Medellín.

2.2 Objetivos Específicos:

- Diseñar un nuevo protocolo de identificación y calibración de las señales EMG para el brazo robótico “El Medallo Version1” que facilite el proceso actual de calibración con el que cuenta la empresa E-NABLE.
- Mejorar el diseño circuital del brazo robótico “El Medallo Version1” mediante el diseño de un PCB usando software libre, que permita el acceso y modificación por usuarios con acceso al repositorio del diseño.
- Validar y documentar en el formato de acceso libre de la empresa el protocolo de calibración de señales EMG y la funcionalidad de la tarjeta PCB.
- Evaluar sensores de distancia para la implementación en el prototipo de una segunda prótesis diseñada por E-NABLE para pacientes invidentes con ausencia de alguna extremidad superior.
- Rediseñar los archivos de impresión 3D de la segunda prótesis para pacientes invidentes de tal manera que permita el futuro acople del sensor de distancia.

3 Marco Teórico.

3.1 PRÓTESIS.

Una prótesis es un dispositivo que se implanta en el cuerpo para reemplazar una parte faltante del mismo, pueden ser implantados en cualquier parte del cuerpo y de acuerdo a esto es que se caracterizan. Puede haber prótesis oculares, de manos, pies, dentales, entre otros [4]. Un ejemplo puede ser una prótesis de mano que se puede ver en la Figura 1:



Figura 1. Prótesis de mano[5].

3.2 ELECTROMIOGRAFÍA (EMG).

La electromiografía, también conocida como EMG, es un registro de la actividad eléctrica de las unidades motoras del músculo. En este registro se mide la diferencia de potencial entre un par de electrodos y se pueden apreciar la activación de varios músculos cuando se contraen o presentan actividad motora [6], tal como se muestra en la Figura 2. ¡Error! No se encuentra el origen de la referencia.

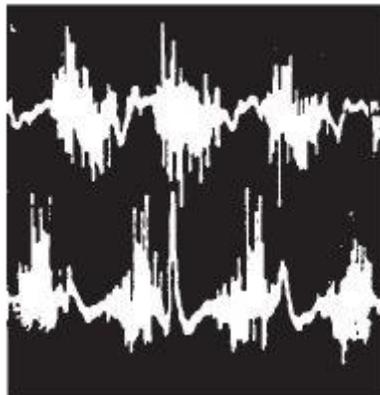


Figura 2. Trazos electromiográficos de los músculos [6].

3.3 SENSORES DE DISTANCIA

Los sensores de distancia están pensados para realizar la medida de distancia o desplazamiento lineal, de una forma automatizada, ya que proporcionan una señal eléctrica según la variación de la distancia [7].

3.3.1 SENSORES DE DISTANCIA POR ULTRASONIDO

Los sensores de ultrasonido son dispositivos que usan las ondas ultrasónicas para determinar la distancia de un objeto. Su funcionamiento se puede ver en la Figura 3 **Error! No se encuentra el origen de la referencia.** donde se tiene un transmisor que emite una señal o pulso ultrasónico hacia un objeto, donde esta señal o pulso rebota y llega a un receptor este calcula el tiempo que transcurre entre emitirse la señal o pulso y en percibirse, luego de esto usando la fórmula mostrada en (1) se determina la distancia del objeto [8].

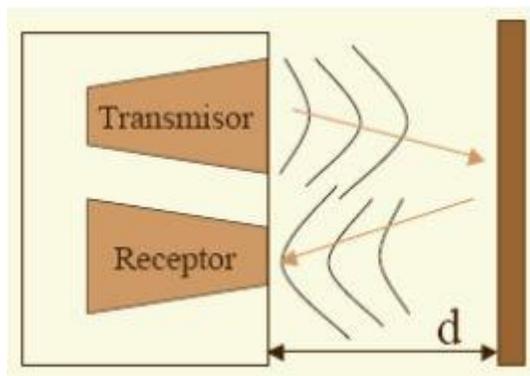


Figura 3. Funcionamiento sensor ultrasónico [8].

$$d = \frac{1}{2} V * t \quad (1)$$

3.3.2 SENSORES DE DISTANCIA INDUCTIVOS

Los sensores de distancia inductivos permiten realizar medidas de distancia sin contacto gracias a su tecnología inductiva. La medida la realizan de forma proporcional, detectando la distancia a la que se encuentra el metal, según esa distancia proporciona una señal analógica en corriente o voltaje, según el rango con el que cuente el sensor de distancia inductivo[7]. La mayor restricción de este tipo de sensores es que se aplica únicamente para detectar superficies metálicas como latón, acero, aluminio, etc. Un ejemplo de este se puede ver en la Figura 4 **Error! No se encuentra el origen de la referencia.**



Figura 4. Sensor de distancia inductivo [7].

3.3.3 SENSORES DE DISTANCIA LASER

Estos sensores tienen como principio básico la detección del cambio de la luz, que es percibido por un receptor y así determinar la distancia de un objeto. El funcionamiento de estos sensores es un láser que emite un rayo hacia el objeto como se muestra en la Figura 5. La luz reflejada por el objeto es concentrada por el lente del receptor, formando una imagen sobre el elemento receptor de luz. Cuando la distancia cambia, la luz concentrada se refleja en un ángulo diferente, y la posición de la imagen cambia correspondientemente[9].

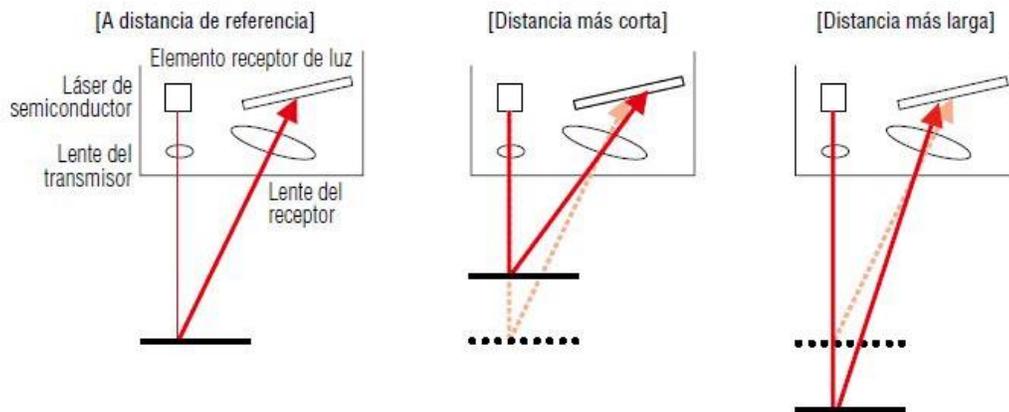


Figura 5. Funcionamiento sensor de distancia laser [9].

3.4 BRAZO ROBÓTICO “EL MEDALLO VERSION1”

Este brazo robótico es el nuevo prototipo de prótesis realizada por la compañía E-NABLE Medellín, el cual está diseñado para pacientes que hayan perdido una extremidad superior de forma que su extremidad tenga una parte distal del codo hacia la mano. Este brazo funciona adquiriendo la señal EMG del antebrazo y del bíceps del paciente, la cual de acuerdo a la actividad del músculo se compara con unos umbrales preestablecidos para cada paciente y dependiendo de esta actividad del músculo se controlan la apertura y cierre de la mano. Este diseño se puede apreciar en la Figura 6.



Figura 6. Brazo robótico "El Medallo Version1" [10].

3.5 HARDWARE Y SOFTWARE LIBRE

Se le llama Hardware libre y Software libre a los dispositivos y programas informáticos que se pueden obtener y manipular de forma libre, normalmente para el caso del software se presenta también de forma gratis. Tomando en cuenta el hardware son sus especificaciones las que se encuentran de forma gratis [11], [12]. La compañía E-NABLE Medellín apoya este tema haciendo que los dispositivos y programas utilizados para crear el brazo robótico "El Medallo Version1" sean de estas características, debido a que se desea que sea de fácil replicación y que se abra a todo el mundo.

3.6 ARQUITECTURA MODELO-VISTA-CONTROLADOR (MVC)

Cuando se habla de programación, existe una arquitectura de software que separa los datos de una aplicación, la interfaz gráfica de usuario y la lógica de control en tres campos distintos; llamada MVC. Este tipo de arquitectura ha tomado un gran uso debido a que es de mucha practicidad y es de fácil uso para el usuario además de su gran aplicación en todo tipo de lenguajes de programación. Es por esto que se ha tenido una gran aplicación de este tipo de arquitectura [13].

3.6.1 MODELO

Es el que contiene la representación de los datos, aparte también es el encargado de definir la funcionalidad del sistema.

3.6.2 VISTA

Es la que compone la información que se envía al cliente, tiene como función recibir los datos del modelo y puede mostrar la información cambiada por el modelo o el controlador.

3.6.3 CONTROLADOR

Es el que actúa como intermediario entre el modelo y la vista gestionando el flujo de información y la adaptación de la misma para cada uno de los campos.

4 Metodología.

Para elaborar los objetivos del trabajo, este se segmentó en dos ramas. Una rama destinada para el rediseño circuital del brazo robótico “El Medallo Version1” y la segunda para el brazo con sensor de distancia. Ambas se trabajaron de forma paralela dado que cada rama es independiente de la otra. Para la primera rama se siguieron los pasos mostrados a continuación:

4.1 RAMA I

- Mejorar la calibración de EMG acoplado hardware y software del brazo robótico utilizando una arquitectura MVC y programación orientada a objetos por medio del lenguaje de programación QT Creator (C++). Debido a que este poseía un núcleo de mayor velocidad, flexibilidad a la hora de manejar la interfaz gráfica y un conjunto de clases junto con su IDE que permite una mejor implementación de la arquitectura MVC; esto junto con el beneficio de que es un software de código libre y que es de fácil acceso, apoyando así los temas que siguen en la compañía E-Nable Medellín de hardware y software libre. Esta calibración consta de una adquisición de la señal EMG provenientes de los músculos del antebrazo y bíceps con graficación en tiempo real debido a que de esta manera se logra mayor facilidad realizar la calibración.
- Diseñar el PCB de la tarjeta electrónica utilizada en el brazo robótico “El Medallo Version1” y nuevo prototipo de “El Medallo Version2” por medio del software libre EasyEDA para una posible impresión en escala industrial y también para ser compartida al mundo en caso si algún otro usuario desea realizarle algún cambio o acoplamiento para una distinta aplicación. Se escoge este software debido a que posee gran número de elementos y huellas de dispositivos electrónicos de forma confiable y que es de código abierto. Los diseños se elaboraron de acuerdo al adecuado diseño encontrado en la teoría [14], se crearon las respectivas huellas con sus correspondientes dimensiones de los reguladores de voltaje suministrados por la compañía y los demás elementos que no poseían huella bajo el software de EasyEDA. También se calculó y realizó el ancho y espaciamiento de pistas de acuerdo a la corriente que demandaría el circuito, la potencia que demandaría cada elemento y por último cómo se alimentaría el circuito de forma en que se seleccionara bien las huellas, sus conectores y posicionamiento para que este funcione de una forma adecuada y a la vez que sea fácil de conectar.
- Validar el protocolo de calibración de EMG y la funcionalidad de la tarjeta diseñada por medio de software libre para el brazo robótico “El Medallo Version1” mediante la impresión de una tarjeta y prueba de funcionamiento con el brazo.

- Agregar a la documentación previamente establecida en la nube de GitHub los cambios realizados y el manual de usuario para su respectivo uso.

Para la segunda rama destinada al brazo con sensor de distancia se seguirán los siguientes pasos:

4.2 RAMA II

- Realizar búsqueda e identificación de sensores distancia para la aplicación de esta prótesis tomando en cuenta sus dimensiones, su gasto en energía, costo y precisión del sensor. Para esto se procedió a adquirir cada sensor y elaborar ensayos de medición de distancias usando dichos sensores a objetos normales de la vida cotidiana (vasos, objetos esféricos, envases de todo tipo de productos, etc.) y observar la salida y la precisión de dichos sensores por medio del microcontrolador de arduino.
- Evaluar cada uno de los sensores identificados y seleccionar uno de acuerdo con los parámetros mencionados anteriormente de manera de escoger el óptimo.
- Ajustar los diseños de las prótesis para el ensamble del futuro sensor en la misma una vez se hayan evaluado los sensores identificados en el paso anterior.
- Documentar en el formato utilizado por la compañía todo el protocolo elaborado para la prótesis con sensor de distancia.

De forma general, los pasos a seguir en la metodología se pueden ver en la Figura 7.

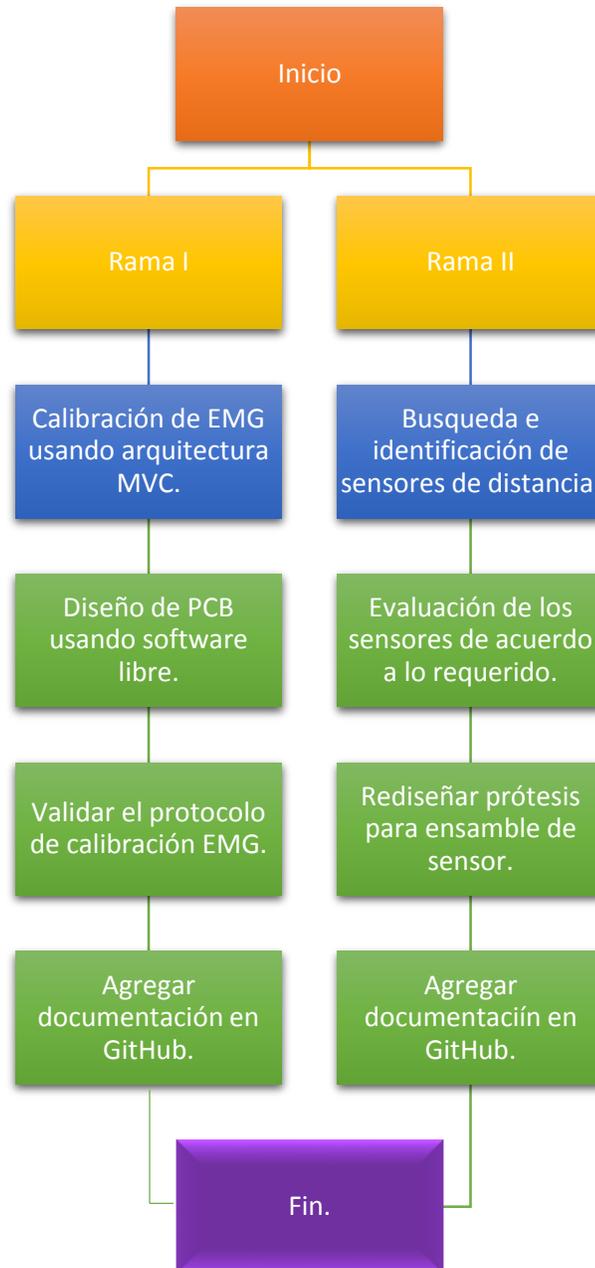


Figura 7. Diagrama de desarrollo de la metodología.

5 Cronograma de Actividades.

El cronograma de actividades utilizado para desarrollar el proyecto se muestra con la elaboración de cada paso por medio de semanas en la Tabla 1

Tabla 1. Cronograma de actividades

CRONOGRAMA PRÁCTICA ACADÉMICA																									
		SEMANA																							
FASE	ACTIVIDAD	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	1. Calibración EMG.	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█
	2. Diseño de formato PCB.													█	█	█	█	█	█	█	█				
	3. Validar protocolo de EMG.													█	█	█	█								
	4. Agregar documentación en GitHub.																					█	█	█	█
2	5. Búsqueda e identificación de sensores.	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█
	6. Evaluación de sensores.													█	█										
	7. Rediseños de las prótesis.															█	█								
	8. Documentar protocolo elaborado.																	█	█	█	█	█			
	9. H: Entrega propuesta.							█																	
	10. H: Entrega segundo informe																					█	█		
	11. H: Presentación pública.																								█

6 Resultados y Análisis.

6.1 PROTOCOLO DE IDENTIFICACIÓN Y CALIBRACIÓN DE LA SEÑAL EMG

Para esta sección, el protocolo de calibración realiza una comunicación serial directa y en tiempo real de la información entregada por el microcontrolador de arduino trabajando en paralelo con el código previamente elaborado por la compañía. La interfaz con la que interacciona el usuario posee tres ventanas de visualización, una donde se muestran las señales adjuntas durante el proceso de adquisición y otras dos donde muestran la señal previamente capturada. Así, de esta manera se lee la señal de los músculos, un ejemplo de este se puede ver en la Figura 8.

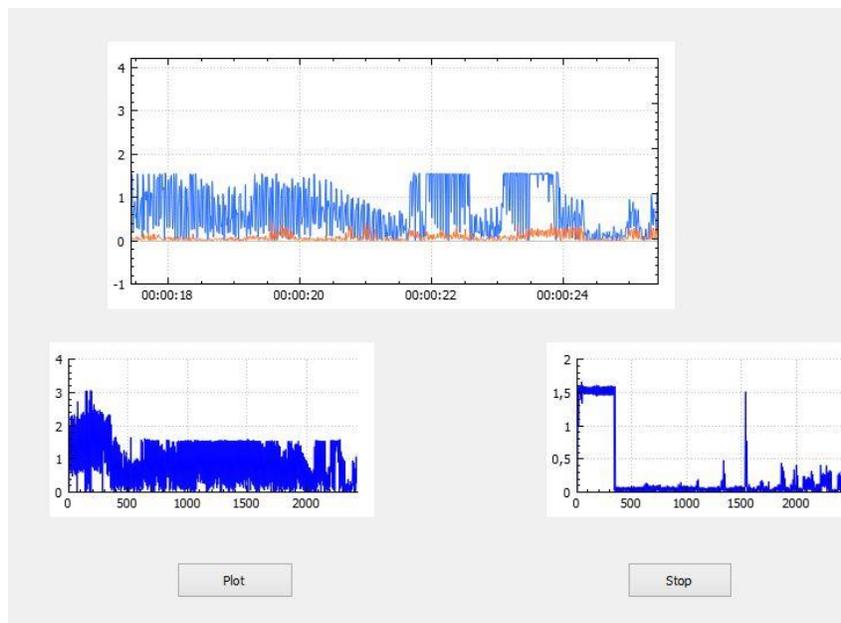


Figura 8. Vista entregada por el protocolo de la calibración.

Donde la ventana central ubicada en la parte superior de la interfaz, es la que corresponde a la ilustración de ambas señales en tiempo real. Las dos ventanas ubicadas en la parte inferior corresponden a la señal proveniente de los músculos del antebrazo (ubicada en la parte izquierda), y la proveniente del bíceps (ubicada en la parte derecha). Con este protocolo se generó una mejora en la parte de la identificación y calibración de la señal EMG, debido a que esta permite una visualización de las señales en tiempo real y con ello establecer los umbrales para cada paciente de una forma mucho más fácil y práctica.

Por parte de la señal obtenida por medio de este protocolo, cabe decir que esta señal está pensada simplemente para la visualización de la misma, debido a que los agarres generados por el brazo se generan por medio de los umbrales detectados por los músculos y que son estos los que le dan funcionamiento al brazo. No se promete que con estas señales se puedan hacer otros tipos de análisis como lo son el análisis en frecuencia y/o espectros obtenidos por ella. Si desea tener más información de cómo se elaboró el protocolo, ver el Anexo 1.

6.2 DISEÑO PCB

Para esta sección se elaboraron diseños para las dos versiones del brazo robótico “El Medallo”, los diseños pueden verse en la Figura 9 y Figura 10 respectivamente.

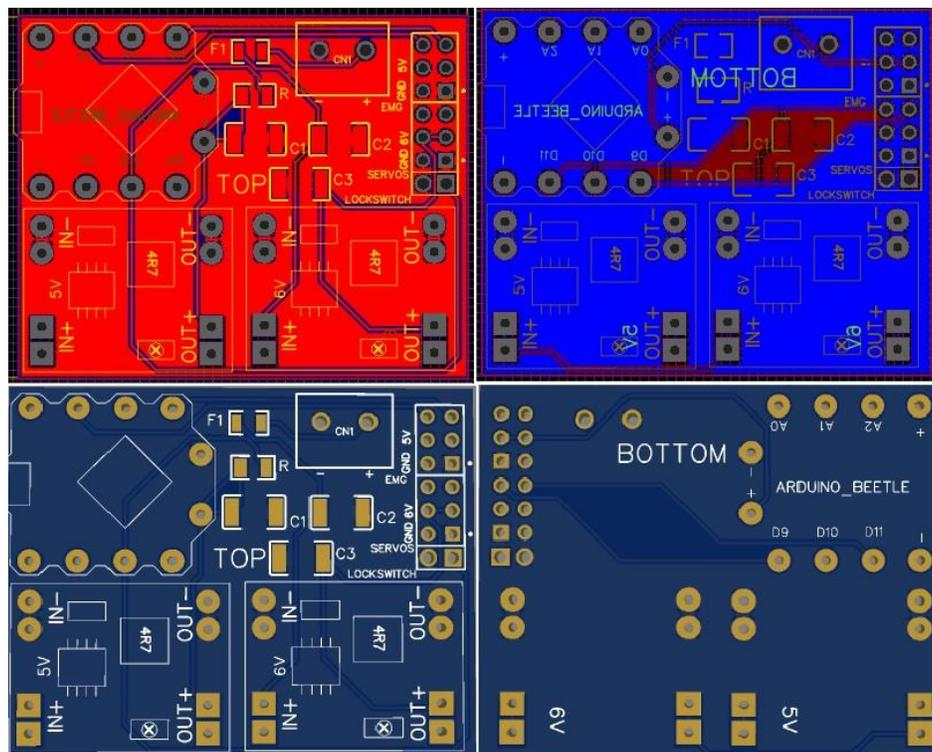


Figura 9. PCB elaborado para la versión 1, capa superior-capa inferior.

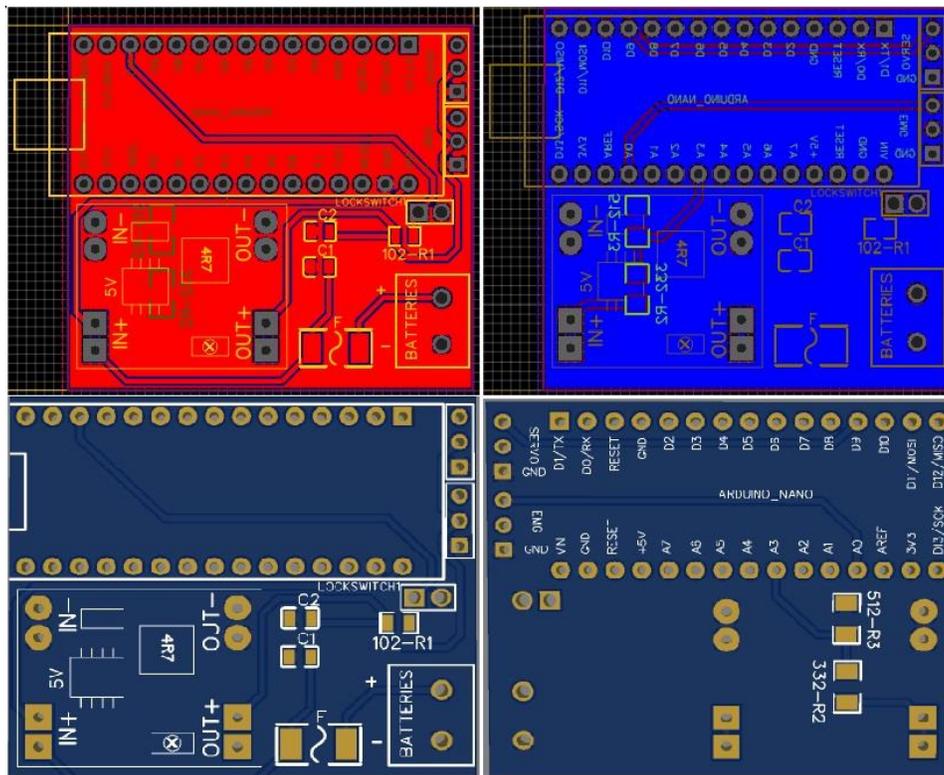


Figura 10. PCB elaborada para la versión 2, capa superior-capa inferior.

Las tarjetas electrónicas contaron con un total de 2 capas de cobre de grosor de 5oz de Cu (equivalente a 0.17399mm o a 6.85mils), se usaron componentes pasivos SMD debido a que se requería disminuir la dimensión del circuito y a su vez mantener buenos valores de tolerancia en estos elementos. Cuentan con un solo regulador de voltaje configurado a 5V (MP1585EN) otorgado por la compañía el cual genera la alimentación para los sensores EMG y los servomotores. Para realizar las conexiones de alimentación del circuito se utilizaron headers macho de 2.54mm para los servomotores, sensores EMG y botón de bloqueo debido a que estos mejoraban el acople entre estos elementos con su proveniencia de fábrica. Las tarjetas obtenidas poseen una dimensión de 40mmX44mm, el cual, es un resultado que puede aplicarse a la versión 2 del brazo robótico “El Medallo” y que mejora las dimensiones del circuito reduciendo notablemente el espacio requerido para este.

Una vez realizados los diseños se realizó un pedido bajo la plataforma de PCBWay (ver Anexo 2) con los parámetros mostrados en la Tabla 2, se escogieron estas características debido a que los servomotores utilizados (JX PDI-4409mg) demandan mucha corriente (1.2 Amperios), de allí salen los valores de estos parámetros. Una vez se realizó el pedido se pueden ver los resultados obtenidos en la Figura 11.

Solamente se realizó el pedido para la segunda versión del brazo robótico debido a que no se contaban con los suficientes recursos monetarios en la compañía para hacer el pedido

de los demás diseños. Fuera de esto, se incentivó la creación del prototipo para la segunda versión de este brazo por lo que los diseños de la primera versión no se tuvieron en cuenta para estos resultados.

Tabla 2. Parámetros para pedido en PCBWay.

ESpecificación	DescRipCIÓN
Board type	Single Pieces
Different Design in Panel	Sin Seleccionar
Size (single)	40 mmX44 mm o 1.57 inchX1.73 inch
Quantity (single)	10-Cuántas tarjetas se desee.
Layers	2 capas
Material	FR-4
FR4-TG	TG 130-140
Thickness	1.6mm
Min Track/Spacing	8/8mil=0.2032mm
Min Hole Size	0.3mm
Solder Mask	Green
Silkscreen	White
Gold fingers	No
Surface Finish	HASL con plomo
Via Process	Tending vías-usando gerber es irrelevante
Finished Copper	5 oz Cu=0.1735mm
Extra pcb product number	Sin checkear



Figura 11. Circuito realizado para el brazo robótico "El Medallo Version2".

Para comprobar el funcionamiento del circuito, se soldaron los elementos seleccionados previamente. Inicialmente se carga el código establecido con las configuraciones de

calibración EMG en el microcontrolador de arduino NANO, se conecta el sensor EMG y se comprueba que se puede adquirir la señal del músculo del antebrazo correctamente. Luego de esto, se elaboran los ajustes en el código para el funcionamiento del brazo robótico, se conectan los servos junto con las baterías (LIR 18650) para notar el funcionamiento en conjunto de todos los elementos.

El circuito funcionó satisfactoriamente y es de total aplicación para este modelo de brazo robótico. Con este, una vez previamente realizada la calibración para cada paciente, se puede utilizar en cualquier brazo para controlar la apertura y cierre de la mano para el modelo de “El Medallo” de una forma estandarizada circuitalmente. Cabe recalcar que con este dispositivo no se puede realizar una calibración EMG para el músculo del bíceps, dado a que se le brinda prioridad a los músculos del antebrazo y con estos se brinda el control de la mano robótica. Si desea ver los diseños y los archivos Gerber generados, ver el Anexo 3 y Anexo 4 respectivamente.

6.3 REDISEÑO DE PRÓTESIS CON DETECCIÓN DE DISTANCIA DE OBJETOS

Una vez realizada la búsqueda de sensores de distancia se llegaron a dos con respuesta muy favorable para la aplicación, estos se muestran en la Figura 12 y Figura 13. Una vez caracterizados se notó su respuesta mediante ensayos de medida de distancia y se llegaron a los resultados mostrados en la Tabla 3.

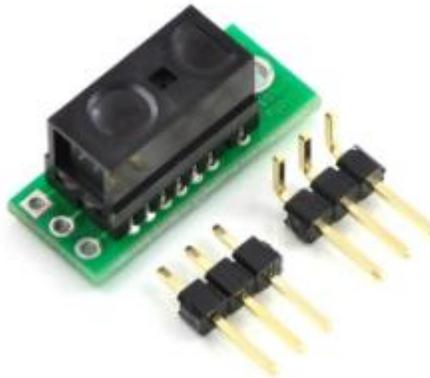


Figura 12. Sensor de distancia GP2Y0D815Z0F.

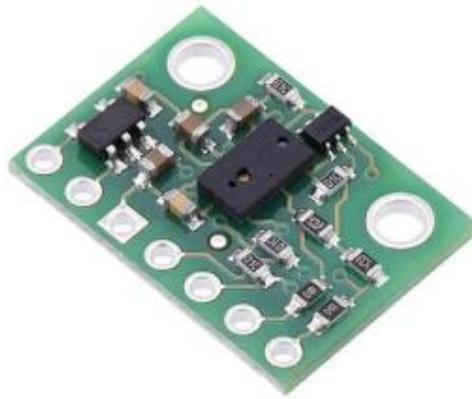


Figura 13. Sensor de distancia VL6180X.

Tabla 3. Tabla comparativa de ambos sensores de distancia.

Sensor	Tipo	Tipo de salida	Resolución		Precio [\$COP]	Nota
			según fabricante [cm]	Resolución real [cm]		
GP2Y0D815Z0F	Láser	Analógica (0V-5V)	0.5	5	34000	No escogido
VL6180X	Infrarrojo	Digital (Comunicación I2C)	0.1	0.7	28800	Escogido (Posee librería en arduino para su manejo)

Una vez escogido el sensor se diseñaron los acoples de dicho sensor por medio de software libre el cual se puede ver en la Figura 14, se utilizaron el software de Fusion360 y de 3D Builder (este último presente en computadores windows10 en adelante). Para elaborar este diseño, se tuvo en cuenta el grosor de la palma de la prótesis, las dimensiones del sensor escogido previamente, la forma en cómo se acoplaría las entradas y salidas del sensor junto

con el resto del circuito y por último el acople mecánico que tendría el sensor. Es por esto que se elaboró la geometría mostrada en la Figura 14 de forma que las líneas de comunicación I2C pasaran a través de este acople, se lograra un ajuste mecánico por medio de tornillos del sensor junto con la palma y por último se utilizaron medidas con un margen mayor a la resolución entregada por las impresoras 3D implementadas por la compañía para que el sensor tuviese un espacio adecuado en la palma de la mano. Todo esto teniendo en cuenta que el circuito debe encontrarse dentro de la mano para mayor comodidad para el paciente. Para este nuevo diseño se cambió el dispositivo de retroalimentación de alarma (Buzzer) por un motor vibratorio tipo moneda, el cual se puede ver en la

Figura 15. Debido a que, según información de antiguos pacientes, la retroalimentación por medio de la alarma resulta ser algo molesto para las demás personas. Se optó por mover las baterías al brazalete que amarra la zona del tríceps del paciente y también se realizó un diseño PCB del mismo circuito utilizado para esta prótesis. El esquemático y diseño PCB implementado se puede ver en la Figura 16 (ver Anexo 5 para más detalle).

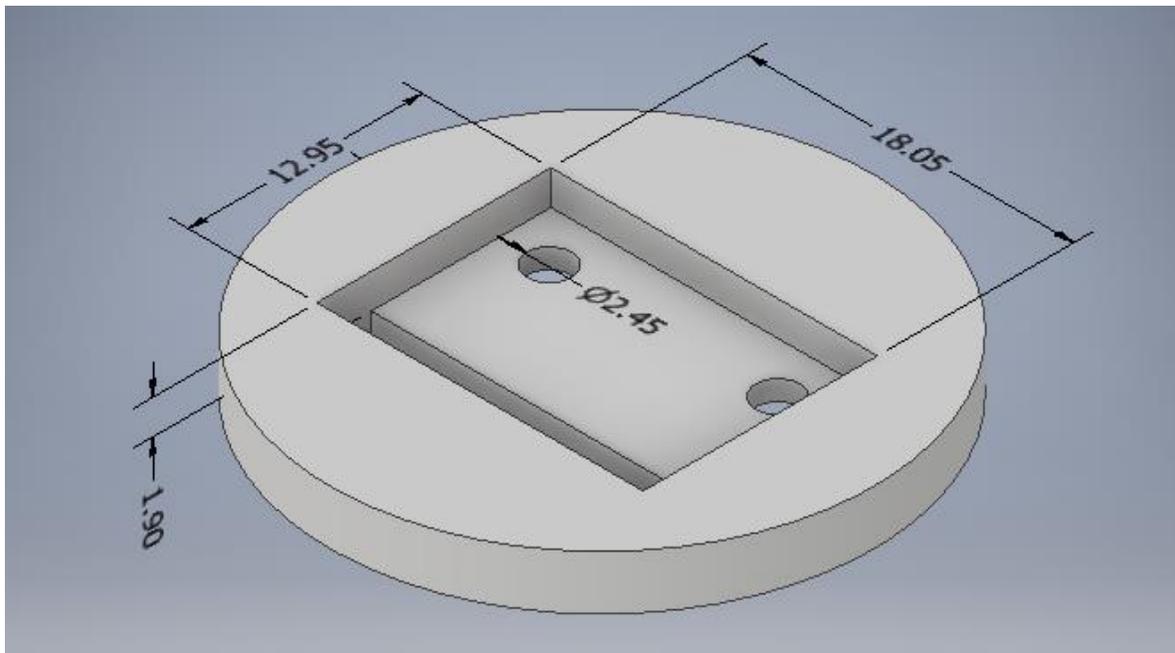


Figura 14. Acople del sensor para insertar en la parte superficial de la prótesis



Figura 15. Motor vibratorio tipo moneda

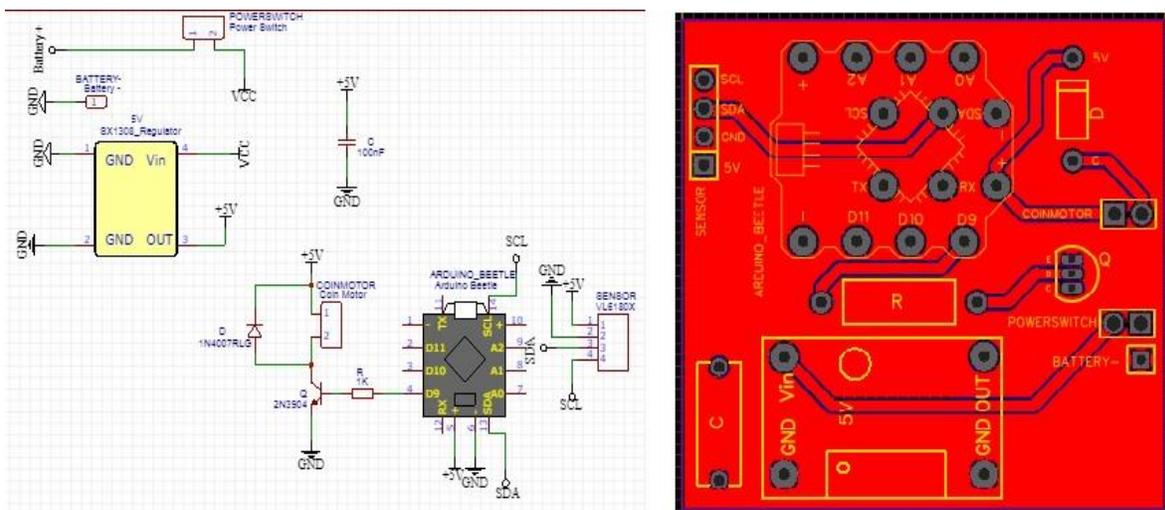


Figura 16. Circuito utilizado para sensor de distancia.

Luego de esto, se encontró una librería elaborada por Pololu para arduino donde se realiza la comunicación I2C para este sensor de una manera fácil [15]. Con esto se procedió a realizar el código para el control de vibración del motor el cual se puede ver en el Anexo 6. Un esquema del funcionamiento de este puede verse en la Figura 17.

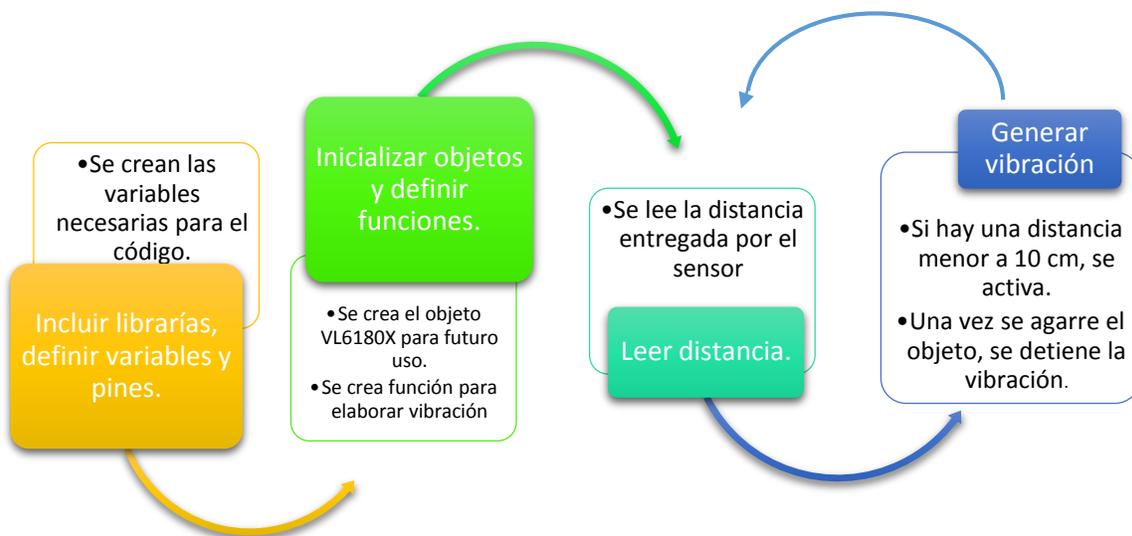


Figura 17. Esquema de funcionamiento del código de control de vibración.

Dado a que no se contó con el presupuesto para realizar una tarjeta PCB para este brazo, se procedió a realizar el esquemático mostrado en la Figura 16 de forma manual utilizando

baquetas universales y los elementos otorgados por la compañía. El circuito implementado se puede ver en la Figura 18.

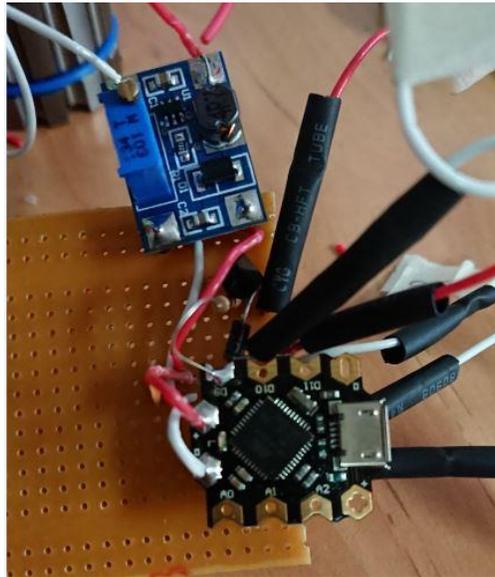


Figura 18. Circuito implementado para brazo con detección de distancia.

El brazo finalmente funcionó de manera adecuada a lo esperado, una vez se ingresó el circuito dentro de la palma y se hizo el debido cableado para hacer la alimentación del circuito, el motor vibratorio junto con el pulsador de tipo balancín del encendido y las líneas de comunicación del sensor, el brazo detecta y hace notar por medio de vibración la distancia a la cual se encuentra el objetivo, es decir, posee un control de vibración el cual indica a las personas con pérdida parcial de la visión que tan cerca está el objeto. Un ejemplo de esta prótesis puede verse en la Figura 19. **¡Error! No se encuentra el origen de la referencia.**, una vez hecho esto se procedió a realizar la documentación.



Figura 19. Prótesis final de brazo detector de distancia.

6.4 DOCUMENTACIÓN DETALLADA DE LOS CAMBIOS REALIZADOS BAJO LA PLATAFORMA DE LA COMPAÑÍA E-NALBE MEDELLÍN

La compañía E-Nable Medellín trabaja bajo la plataforma de Github, es allí donde se controlan todos los documentos y nuevos lanzamientos de versiones que se hayan compartido o creado. Todos los resultados anteriormente mencionados han sido publicados bajo esta plataforma, algunos son de carácter público como lo es el brazo detector de distancias y la primera versión del brazo robótico “El Medallo”, otros son de carácter privado como la segunda versión del brazo robótico “El Medallo” debido a que se encuentra por el momento como prototipo de la compañía. Toda la documentación establece qué elementos se necesitarán para realizar una prótesis, cómo se conseguirán dichos elementos y como utilizarlos, introduce brevemente lo en lo que se enfoca cada página, posee guías de instalación de software de código libre y también posee links de asociación a videotutoriales y páginas a tener en consideración de dichos software y diseños.

Bajo consideración del fundador de la compañía, se ha logrado hacer dicha documentación estableciendo un orden de pasos para fabricar el brazo detector de distancias y también mencionando los cambios elaborados en la calibración EMG junto con los diseños PCB. Cada página está pensada para mostrar y/o sugerir un protocolo a seguir para que cualquier persona que lea la documentación, se instruya de la mejor forma en cuanto a lo que se hizo, con el fin de que se pueda hacer una copia de estos diseños y desarrollos mencionados.

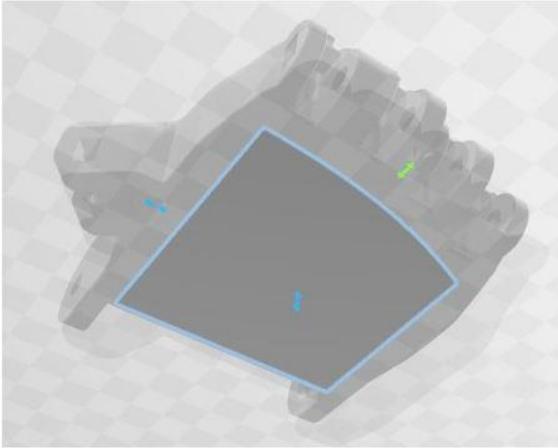
Las páginas se enlazan por medio de un orden que se muestra de forma general en las páginas de inicio y también por medio de los contenidos mostrados allí, mostrando una pequeña introducción del por qué se han realizado y un protocolo a seguir. Un ejemplo de esta se puede ver en la Figura 20 y Figura 21.

Background

To get the sizes and the files for the patient you can do the same protocol as in the [First version](#). Once you get to Integrating Electronics in Hand section, then you can come back to this particular page and follow the next step by step.

1. Getting The Palm Fixed

When you have the .stl file of the hand, you can open it in 3D Builder. After that, you have to import the .stl file that you can download [here](#) named as Body1. Probably this new file is going to have a huge dimensions, so what you have to do is just simply move the size of this stl file and scale it into an average of the palm as you can see in the next figure:



Then just click on both bodies and go to "Edit tools" and click "Merge"

Figura 20. Ilustración página de contenidos.

Contents

1. Preparation

- [Parts and Tools List-V1.5](#)
- [Patient Information Collection](#)
- [File Preparation-V1.5](#)

2. Manufacture

- [Printing](#)
- [Circuitry and Coding](#)

3. Assembly

- [Pre-Assembly](#)
- [Assembly](#)

4. User Guide

Acknowledgment

- Benjamin Complin-Background and Version 1.0
- Oliver Vogel Reed-CAD designs and illustrations for battery case.

Made by: Mauricio Zapata Contreras(MauricioZapata00)

Figura 21. Ilustración página de inicio.

▼ Pages **16**

- Home
- Assembly
- Circuitry
- Circuitry And Coding V1.5
- Distance Sensing Arm_V1.5
- File preparation
- File Preparation Version 1.5
- MK1 Prototype
- Parts and Tools List
- Parts and Tools List V1.5
- Patient Information Collection
- Possible Future Edits, if required.
- Pre Assembly
- Printing
- Ultrasonic Range Sensor Live Report
- Show 1 more pages...

+ Add a custom sidebar

Clone this wiki locally

+ Add a custom sidebar

Clone this wiki locally

Con esto, se logra elaborar una documentación clara y destinada a los requerimientos de la compañía para los objetivos elaborados en este proyecto. Alguna de esta documentación es de código libre y se puede ver en el Anexo 7.

7 Conclusiones

- Con respecto al diseño de las tarjetas de PCB, estas poseen buena aplicación para el brazo robótico “El Medallo” dado que funcionan para elaborar una calibración EMG y son de uso para controlar la apertura y cierre de la mano. Pero si se desea reducir un poco el costo de dichas tarjetas debido al elevado contenido en cobre (oz de Cu) que poseen, se pueden enviar cables de alimentación externos a los elementos que demanden más corriente diferentes a los que se crean durante el diseño, elaborando las huellas respectivas para estos cables y soldando estos manualmente. De esta forma se aprovecha el grosor y ancho de pista de los cables alternos y se puede reducir la cantidad de cobre de las tarjetas.
- También se puede establecer como búsqueda a futuro, un servomotor que se pueda acoplar a la prótesis y que a su vez no genere tanto gasto a las baterías y que cumpla con las especificaciones de torque para hacer un cierre de la mano como lo hace el servomotor implementado actualmente.
- Para las personas que hayan perdido totalmente la visión y que deseen utilizar la prótesis detector de distancias, se les restringe su uso debido a que lo mencionado anteriormente funciona para dar idea y dar percepción de la distancia a la que está el objeto. Más no funciona como percepción espacial de todos los objetos a los que la persona desee alcanzar o agarrar. El uso más apropiado es para las personas que hayan perdido parcialmente su vista pero logren distinguir algo de los objetos presentes a su alrededor. Para brindar este tipo de percepción espacial, es necesario adaptar más de un sensor al brazo y con ello entregar una ubicación aproximada a los pacientes de dónde se encuentran los objetos, una vez hecho esto, se podría lograr un buen agarre para estas pacientes con pérdida total de la visión.
- De igual forma, se deben realizar pruebas a pacientes con pérdida total de la visión. para observar distintos inconvenientes que surgen a la hora de acoplar la prótesis a estas personas, siguiendo un protocolo semejante y documentando cada problema surgido.
- Por parte de la calibración EMG, se logra mejorar dicha identificación y calibración de los pacientes usando el software de QT Creator. Con ella se tiene pensado que cualquier persona pueda realizar dicha calibración de forma fácil y práctica. Se puede lograr como trabajo a futuro que esta calibración la puedan realizar los mismos pacientes bajo su misma comodidad, para esto se podría usar las tecnologías actuales como lo son los celulares y tablets. Y así no depender de la compañía por si necesitan realizar un ajuste a este tipo de prótesis.

8 Referencias bibliográficas

- [1] F. Médica, “En Colombia gran número de los pacientes amputados no han sido evaluados por un cirujano vascular,” 2018. [Online]. Available: <https://www.formulamedica.com.co/pacientes-amputados-no-han-sido-evaluados-por-un-cirujano-vascular/>. [Accessed: 07-Jul-2019].
- [2] J. E. Restrepo, E. A. Yara, J. Cano Betancur, and L. N. Tavera, “Perfil emocional de un grupo de militares colombianos víctimas de minas antipersona o artefactos explosivos improvisados. = Emotional profile of a group of Colombian military victims of landmines or Improvised Explosive Devices.,” *Rev. Colomb. Psiquiatr.*, vol. 43, no. 2, pp. 87–95, 2014.
- [3] “e-NABLE Medellín | e-NABLE Medellín.” [Online]. Available: <https://enablemedellin.com/es/inicio/>. [Accessed: 25-Jul-2019].
- [4] “Prótesis: MedlinePlus enciclopedia médica.” [Online]. Available: <https://medlineplus.gov/spanish/ency/article/002286.htm>. [Accessed: 11-Jul-2019].
- [5] “Prótesis de Mano - Trucos Optimistas.” [Online]. Available: <https://trucosoptimistas.org/descargas/protesis-de-mano/>. [Accessed: 18-Jul-2019].
- [6] J. L. G. H. Javier de León Fraga, Dra . Alma Rosa Higuera Murillo , Dra . Rita Gabriela León Jiménez, Norma Leticia García Carbajal, *GANONG Review of a Medical Physiology*. México , D . F . , 2010.
- [7] “Sensores de distancia - Sensing, Sensores de Medida.” [Online]. Available: <https://sensores-de-medida.es/medicion/sensores-y-transductores/sensores-de-distancia/>. [Accessed: 25-Jul-2019].
- [8] D. P. De Diego, “Sensores de distancia por ultrasonidos 1.,” *ALCABOT Robótica en la Universidad de Alcalá*. p. 13, 2006.
- [9] “¿Qué es un sensor láser de tipo de reconocimiento de ‘posición’? | Fundamentos del sensor: Guía de sensores para fábricas clasificados por principios | KEYENCE.” [Online]. Available: https://www.keyence.com.mx/ss/products/sensor/sensorbasics/laser_location/info/index.jsp. [Accessed: 25-Jul-2019].
- [10] “El Medallo v1 (May 2019) · enable-medellin/robotic-arm Wiki.” [Online]. Available: [https://github.com/enable-medellin/robotic-arm/wiki/El-Medallo-v1-\(May-2019\)](https://github.com/enable-medellin/robotic-arm/wiki/El-Medallo-v1-(May-2019)). [Accessed: 25-Jul-2019].
- [11] “Hardware libre - EcuRed.” [Online]. Available: https://www.ecured.cu/Hardware_libre. [Accessed: 26-Jul-2019].
- [12] “Software libre - EcuRed.” [Online]. Available: https://www.ecured.cu/Software_libre. [Accessed: 26-Jul-2019].
- [13] “Modelo vista controlador (MVC).” [Online]. Available: <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador->

mvc.html. [Accessed: 25-Jul-2019].

- [14] “Top 5 de recomendaciones de diseño de PCB que todo diseñador debe conocer | Altium.” [Online]. Available: <https://es.resources.altium.com/blog-de-diseño-de-pcb/top-5-de-recomendaciones-de-diseño-de-pcb-que-todo-diseñador-debe-conocer>. [Accessed: 17-Dec-2019].
- [15] “pololu/vl6180x-arduino: Pololu Arduino library for VL6180X distance and ambient light sensor.” [Online]. Available: <https://github.com/pololu/vl6180x-arduino>. [Accessed: 17-Dec-2019].

9 ANEXOS

Anexo 1. Código utilizado para la calibración.

```
#include "tryrealplot.h"
#include "ui_tryrealplot.h"
#include <qcustomplot.h>
#include <QSerialPort>
#include <QDebug>
static QTimer dataTimer;
static QSerialPort arduino;
static QByteArray MyByte;
static QString MyStr;
static QStringList List;
static QVector<double> VecForearm,VecBicep,XAxis;
static int cont=0;
static double Forearm, Bicep;
TryRealPlot::TryRealPlot(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::TryRealPlot)
{
    ui->setupUi(this);
    ui->customPlot->addGraph(); // blue line
    ui->customPlot->graph(0)->setPen(QPen(QColor(40, 110, 255)));
    ui->customPlot->addGraph(); // orange line
    ui->customPlot->graph(1)->setPen(QPen(QColor(255, 110, 40)));
    QSharedPointer<QCPAxisTickerTime> timeTicker(new QCPAxisTickerTime);
    timeTicker->setTimeFormat("%h:%m:%s");
    ui->customPlot->xAxis->setTicker(timeTicker);
    ui->customPlot->axisRect()->setupFullAxesBox();
    ui->customPlot->yAxis->setRange(-1.0, 4.2);
    connect(ui->customPlot->xAxis, SIGNAL(rangeChanged(QCPRange)), ui->customPlot->xAxis2, SLOT(setRange(QCPRange)));
    connect(ui->customPlot->yAxis, SIGNAL(rangeChanged(QCPRange)), ui->customPlot->yAxis2, SLOT(setRange(QCPRange)));
    connect(&dataTimer, SIGNAL(timeout()), this, SLOT(makePlot()));
    //dataTimer.start(0);
    arduino.setPortName("COM6");
    arduino.setBaudRate(QSerialPort::Baud38400);
    arduino.setDataBits(QSerialPort::Data8);
    arduino.setParity(QSerialPort::UnknownParity);
    arduino.open(QIODevice::ReadOnly);
}
```

```

TryRealPlot::~TryRealPlot()
{
    delete ui;
}
void TryRealPlot::makePlot()
{
    static QTime time(QTime::currentTime());
    // calculate two new data points:
    double key = time.elapsed()/1000.0; // time elapsed since start of demo, in seconds
    static double lastPointKey = 0;
    if (key-lastPointKey > 0.009) // at most add point every 9 ms
    {
        MyByte=arduino.readLine();
        MyStr=QString::fromUtf8(MyByte);
        List=MyStr.split("\t");
        qDebug() << MyStr;
        qDebug() << List;
        qDebug() << List.size();
        for (QString i : List) {
            if(cont==3){
                qDebug() << i;
                Forearm=i.toDouble();
                VecForearm.append(Forearm);
            }else if(cont==7){
                qDebug() << i;
                Bicep=i.toDouble();
                VecBicep.append(Bicep);
            }
            cont++;
        }
        cont=0;
        //qDebug() << List.at(3);
        //qDebug() << List.at(7);
        //qDebug() << typeid (List.at(0)).name();
        qDebug() << Forearm;
        qDebug() << typeid (MyStr).name();
        qDebug() << "Espacio";
    }
}

```

```

// add data to lines:
//ui->customPlot->graph(0)->addData(key, qSin(key)+qrand()/(double)RAND_MAX*1*qSin(key/0.3843));
//ui->customPlot->graph(1)->addData(key, qCos(key)+qrand()/(double)RAND_MAX*0.5*qSin(key/0.4364));
ui->customPlot->graph(0)->addData(key, Forearm);
ui->customPlot->graph(1)->addData(key, Bicep);
// rescale value (vertical) axis to fit the current data:
//ui->customPlot->graph(0)->rescaleValueAxis();
//ui->customPlot->graph(1)->rescaleValueAxis(true);
lastPointKey = key;
}
// make key axis range scroll with the data (at a constant range size of 8):
ui->customPlot->xAxis->setRange(key, 8, Qt::AlignRight);
ui->customPlot->replot();

// calculate frames per second:
static double lastFpsKey;
static int frameCount;
++frameCount;
if (key-lastFpsKey > 2) // average fps over 2 seconds
{
    ui->statusBar->showMessage(
        QString("%1 FPS, Total Data points: %2")
            .arg(frameCount/(key-lastFpsKey), 0, 'f', 0)
            .arg(ui->customPlot->graph(0)->data()->size()+ui->customPlot->graph(1)->data()->size())
            , 0);
    lastFpsKey = key;
    frameCount = 0;
}
}

void TryRealPlot::on_Plot_clicked()
{
    dataTimer.start(0);
}

void TryRealPlot::on_Stop_clicked()

```

```

void TryRealPlot::on_Stop_clicked()
{
    dataTimer.stop();
    arduino.close();
    for (int i=0;i<VecForearm.size();i++) {
        XAxis.append(i);
    }
    MakeForearmPlot();
    XAxis.clear();
    for (int i=0;i<VecBicep.size();i++) {
        XAxis.append(i);
    }
    MakeBicepPlot();
}
void TryRealPlot::MakeForearmPlot(){
    ui->PlotForearm->addGraph();
    ui->PlotForearm->graph(0)->setData(XAxis, VecForearm);
    ui->PlotForearm->xAxis->setRange(0,XAxis.size());
    ui->PlotForearm->yAxis->setRange(0,4);
    ui->PlotForearm->replot();
}
void TryRealPlot::MakeBicepPlot(){
    ui->PlotBicep->addGraph();
    ui->PlotBicep->graph(0)->setData(XAxis,VecBicep);
    ui->PlotBicep->xAxis->setRange(0,XAxis.size());
    ui->PlotBicep->yAxis->setRange(0,2);
    ui->PlotBicep->replot();
}
}

```

Anexo 2. Link para hacer pedidos de diseños PCB con la plataforma PCBWay.

<https://www.pcbway.com/orderonline.aspx?x=100&y=100&num=10&Layersquote=2&Thicknessquote=1.6>

Anexo 3. Archivos con diseños PCB “El Medallo V2”.

https://drive.google.com/drive/folders/1Kf4RJ1qZnDIHbA7CD_h-E-59HT2qz_Jt

Anexo 4. Archivos gerber para “El Medallo V2”.

<https://drive.google.com/drive/folders/1XU9SLzBKXcdMcScbpbq1CJbTB5dvOs3X>

Anexo 5. Archivos con diseños PCB de brazo detector de distancia.

<https://drive.google.com/drive/folders/1PgtEq-BAbrVj8TfWRjtpEEXWUNnCWJdq>

Anexo 6. Código utilizado para control de vibración.

```
|/////Include libraries
#include <Wire.h>
#include <VL6180X.h>
/////Define Pins
#define Motor 9
/////Define variables
VL6180X sensor; //VL6180X Object
uint16_t Value;
uint16_t cont=0; //Times the object is too close
uint16_t Times_Activate=20; //Times the coin motor is activate
void PWM(int Delay_Time); //Enable and disable the coin motor
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  Wire.begin();
  sensor.init(); //Initialize the sensor
  sensor.configureDefault();
  sensor.setTimeout(500); //Configure how much time the sensor has to sent the information
  pinMode(Motor,OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  Value=sensor.readRangeSingleMillimeters(); //Read the distance in millimeters
  Serial.println(Value); //Print the distance
  if((Value<100)and(Value>50)){ //The object is far
    PWM(250);
    cont=0;

    }else if((Value<50)and(Value>20)){ //The object is getting close
    PWM(100);
    cont=0;
  }else if((Value<20)and(cont<Times_Activate)){ //The object is too close
    PWM(50);
    cont++;
  }else{
    digitalWrite(Motor,LOW);
  }
  delay(50);
}

void PWM(int Delay_Time){ //Activate the motor
  digitalWrite(Motor,HIGH);
  delay(100);
  digitalWrite(Motor,LOW);
  delay(Delay_Time);
}
```

Anexo 7. Links de código abierto elaborados durante este proyecto.

<https://github.com/enable-medellin/Distance-Sensing-Arm-Prosthetic-Prototype>
[https://github.com/enable-medellin/robotic-arm/wiki/El-Medallo-v1-\(May-2019\)](https://github.com/enable-medellin/robotic-arm/wiki/El-Medallo-v1-(May-2019))