REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

**Universidad Católica Santa María**

**Facultad de Ciencias e Ingenierías Físicas y Formales**

**Escuela Profesional de Ingeniería Electrónica**

# DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ROBÓTICO BIOMÉDICO PARA CONTROLAR LOS MOVIMIENTOS DE UNA PRÓTESIS MIOELÉCTRICA TRANSHUMERAL

Tesis presentada por los bachilleres:

**Talavera Ormeño, Kevin Wilver**

**Taco Medina, Oscar**

Para optar al Título Profesional de:

**Ingeniero Electrónico con especialidad en Automatización y Control**

Asesora:

**Mg. Valdivieso Herrera, Diana Isabel**

**Arequipa - Perú**

**2024**

Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

UCSM-ERP

# UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## INGENIERIA ELECTRONICA

## CON ESPECIALIDAD EN AUTOMATIZACIÓN Y CONTROL

## TITULACIÓN CON TESIS

## DICTAMEN APROBACIÓN DE BORRADOR

Arequipa, 29 de Noviembre del 2023

**Dictamen: 005351-C-EPIE-2023**

Visto el borrador del expediente 005351, presentado por:

**2011204011** - **TALAVERA ORMEÑO KEVIN WILVER**
**2010246101** - **TACO MEDINA OSCAR**

Titulado:

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ROBÓTICO BIOMÉDICO PARA CONTROLAR LOS MOVIMIENTOS DE UNA PRÓTESIS MIOELÉCTRICA TRANSHUMERAL.**

Nuestro dictamen es:

**APROBADO**

**29278874** - **DELGADO BARRA LUCY ANGELA**
**DICTAMINADOR**

**29267682** - **RODRIGUEZ GONZALES PEDRO ALEX**
**DICTAMINADOR**

**29715414** - **COAGUILA GOMEZ RONALD PERCING**
**DICTAMINADOR**

# DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ROBÓTICO BIOMÉDICO PARA CONTROLAR LOS MOVIMIENTOS DE UNA PRÓTESIS MIOELÉCTRICA TRANSHUMERAL

INFORME DE ORIGINALIDAD

| 6% | 7% | 1% | 2% |
|---|---|---|---|
| INDICE DE SIMILITUD | FUENTES DE INTERNET | PUBLICACIONES | TRABAJOS DEL ESTUDIANTE |

FUENTES PRIMARIAS

| | | |
|---|---|---|
| 1 | tesis.ucsm.edu.pe<br>Fuente de Internet | 1% |
| 2 | zaguan.unizar.es<br>Fuente de Internet | 1% |
| 3 | repositorio.uap.edu.pe<br>Fuente de Internet | 1% |
| 4 | uelectronics.com<br>Fuente de Internet | 1% |
| 5 | polaridad.es<br>Fuente de Internet | 1% |
| 6 | datospdf.com<br>Fuente de Internet | 1% |
| 7 | hdl.handle.net<br>Fuente de Internet | 1% |
| 8 | doaj.org<br>Fuente de Internet | 1% |

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## DEDICATORIA

*A mis padres porque son mis guías de vida,*

*gracias por todo su esfuerzo y dedicación.*

*A mis hermanas por todo el apoyo que*

*siempre me brindan.*

*A toda mi familia y amigos por siempre*

*estar animándome en todo este tiempo.*

**Kevin Talavera**

*A mis padres, que con su paciencia y apoyo*

*incondicional me permiten cumplir una meta más.*

*A mis hermanos por su apoyo y confianza.*

*A mi familia, amigos y compañeros que me apoyaron*

*durante todo este proceso, gracias por su ayuda.*

**Oscar Taco**



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## AGRADECIMIENTOS

*Agradezco a Dios por iluminarme, siempre darme sabiduría e inteligencia en todo este proceso.*

*A mis padres Wilver y Rossana quienes con su amor, paciencia y esfuerzo me han permitido llegar a cumplir hoy un sueño más, gracias por inculcar en mí el ejemplo de esfuerzo y valentía, de no temer las adversidades porque Dios está conmigo siempre.*

*A mis hermanas por su cariño y apoyo incondicional, durante todo este proceso, por estar conmigo en todo momento gracias. A toda mi familia porque con sus consejos y palabras de aliento hicieron de mí una mejor persona y de una u otra forma me acompañan en todos mis sueños y metas.*

*Finalmente quiero expresar nuestro más grande y sincero agradecimiento a la Ingeniera Diana Isabel Valdivieso Herrera, principal colaboradora durante todo este proceso, quien con su dirección, paciencia, conocimiento y enseñanza permitió el desarrollo de esta investigación.*

*Talavera Ormeño, Kevin*

*A Dios, por darme la oportunidad de vivir y permitirme el haber llegado hasta este momento tan especial en mi vida.*

*A mis padres, por acompañarme en cada paso que doy como persona y profesional.*

*A mis hermanos, que siempre han estado junto a mí y brindándome todo su apoyo y confianza.*

*A mi familia en general, por motivarme y animarse a ser mejor*

*Taco Medina, Oscar*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA


## RESUMEN

En tiempos remotos, la tecnología no era algo innovador para las personas que de alguna u otra manera habían sufrido accidentes graves, y por diagnósticos tenían que ser amputadas de los miembros superiores (brazos), debido a ello, no se tenía mayores avances de las prótesis para mejorar la calidad de vida de estas personas, debido a la ausencia de alguno de sus extremidades; actualmente, con los avances de la tecnología se ha ampliado los requerimientos de los usuarios, permitiendo desarrollar prótesis mio-eléctricas y biónicas mejorando las versiones mecánicas y funcionales, a pesar de estos avances los usuarios han sumado la preocupación del manejo de las mismas, por lo que, han terminado presentando dificultades para poder controlarlas, debido a que las prótesis Mio-eléctricas son mayormente manejadas por los sensores de fuerzas utilizados, que no generan una retroalimentación háptica para mejorar la acción de fuerza y así poder adaptarse. De acuerdo con este problema, se presenta esta investigación que tiene como objetivo el diseño e implementación de un sistema robótico biomédico para una prótesis mio-eléctrica transhumeral logrando un factible manejo de sus movimientos. En esta investigación se explica todo lo relacionado con el diseño electrónico y mecatrónica de la prótesis, con la finalidad de describir las como las redes neuronales transmiten sensaciones de fuerza, para el movimiento de la misma, cabe destacar que, el diseño robótico biomédico es para una prótesis de un miembro superior (brazo), con tres grados de libertad. Esta investigación cuenta con una metodología de tipo exploratoria, con un diseño descriptivo y experimental, que se podrá aplicar a una población representada por las personas que tienen amputación de la extremidad superior. Pudiendo concluir, que una vez elaborado el diseño de la prótesis mio-eléctrica se espera mejorar la calidad de vida de las personas que puedan tener acceso al uso de las mismas.


**Palabras claves:** Mecatrónica, prótesis, prótesis mio-eléctrica, amputación, robótica biomédica, miembros superiores, electrónica.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

**ABSTRACT**

In the past, technology was not something innovative for people who in one way or another had suffered serious accidents, and due to diagnoses they had to be amputated of the upper limbs (arms), due to this, there were no major advances in prostheses to improve the quality of life of these people, due to the absence of any of its members; Currently, with the advances in technology, the requirements of the users have been expanded, allowing the development of myoelectric and bionic prostheses, improving the mechanical and functional versions, despite these advances, users have added the concern of handling them, so that , have ended up presenting difficulties to control them, because myoelectric prostheses are mostly managed by the force sensors used, which do not generate haptic feedback to improve the force action and thus be able to adapt. According to this problem, this research is presented that aims to design a transhumeral myoelectric prosthesis to implement a biomedical robotic system that controls its movements. In this research, everything related to the mechatronic design of the prosthesis is explained, in order to describe the haptic units that transmit sensations of force, for its movement, it should be noted that the designed prosthesis is of an upper limb (arm), with three degrees of freedom. This research has an exploratory type methodology, with a descriptive and experimental design, which can be applied to a population represented by people who have an upper limb amputation. Being able to conclude that once the design of the myoelectric prosthesis is elaborated, it is expected to improve the quality of life of the people who may have access to the use of them.

**Keywords:** Mechatronics, prosthesis, myoelectric prosthesis, amputation, biomedical robotic, upper limbs, electronic.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

# ÍNDICE GENERAL

**DEDICATORIA** ..... **iii**

**AGRADECIMIENTOS** ..... **iv**

**RESUMEN** ..... **v**

**ABSTRACT** ..... **vi**

**ÍNDICE GENERAL** ..... **vii**

**ÍNDICE DE FIGURAS** ..... **xii**

**ÍNDICE DE TABLAS** ..... **xvii**

**INTRODUCCIÓN** ..... **xviii**

**CAPÍTULO I** ..... **1**

**1. PLANTEAMIENTO DE LA INVESTIGACIÓN** ..... **1**

1.1. Enunciado del problema ..... 1

1.2. Identificación del problema ..... 1

1.3. Descripción del problema ..... 2

1.4. Formulación del problema ..... 3

1.4.1. Problema general ..... 3

1.4.2. Problemas específicos ..... 3

1.5. Área de conocimiento ..... 4

1.6. Justificación de la investigación ..... 4

1.7. Objetivos de la investigación ..... 5

1.7.1. Objetivo general ..... 5

1.7.2. Objetivos específicos ..... 5

1.8. Alcances de la investigación ..... 6

1.9. Variables de la investigación ..... 6

1.9.1 Variable independiente ..... 6

1.9.2. Variable dependiente ..... 6



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

1.9.3. Operacionalización de las variables .....6

**CAPÍTULO II .....8**

**2. MARCO TEÓRICO.....8**

2.1. Antecedentes de la investigación .....8

2.1.1. Antecedentes Internacionales .....8

2.1.2. Antecedentes nacionales .....9

2.2. Bases teóricas .....12

2.2.1. Fisiología del brazo .....12

2.2.2. Fisiología en amputados de brazo .....17

2.2.3. Prótesis .....19

2.2.4. Tipos de fabricación de prótesis: .....26

2.2.5. Lógica difusa .....30

2.2.6. Redes neuronales .....32

2.2.7. Háptico .....44

2.2.8. Tecnología Háptica .....44

2.2.9. La organización psicomotriz .....45

2.2.10. Materiales de la prótesis .....46

2.2.11. ESP32 .....51

2.2.12. Raspberry pi. .....56

2.3. Conceptualización de términos básicos .....57

**CAPÍTULO III .....60**

**3. DESARROLLO DEL PROYECTO .....60**

3.1. Metodología .....60

3.1.1 Diferencia entre Raspberry pi pico frente a ESP32 .....64

3.2. Programas necesarios del Proyecto .....65

3.3. Diferencias de las redes neurales perceptrón simple, multicapa y red base radial ...66

3.3.1. Diferencia entre redes .....66



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

3.3.2. Justificación de la elección de la red neuronal ..... 69
3.4. Controlador PID ..... 70
3.5. Aplicación de los circuitos electrónicos ..... 72
3.5.1 Adquisición de las señales mio-eléctricas ..... 72
3.5.2. Circuito de adquisición de señales mio-eléctricas ..... 75
3.6. Cálculos ..... 86
3.6.1. Amplificación ..... 86
3.6.2. Filtro pasa bajas ..... 86
3.6.3. Filtro pasa altas ..... 87
3.6.4 Filtro notch ..... 88
3.6.5. Amplificador variable ..... 89
3.6.6. Rectificación ..... 89
3.7. Software ..... 91
3.7.1. Algoritmo principal para el funcionamiento de la prótesis ..... 91
3.7.2. Funciones y clases ..... 95
3.8. Medición de señales mio-eléctricas ..... 99
3.8.1. Amplificación ..... 100
3.8.2. Filtrado ..... 101
3.8.3. Amplificación variable ..... 101
3.8.4. Rectificación ..... 102
3.9. Implementación del circuito en PCB ..... 103
3.10. Aprendizaje de la red neuronal ..... 104
3.10.1. Adquisición de datos ..... 104
3.11. Procesamiento de datos ..... 108
3.12. Aprendizaje ..... 110
3.12.1. Parámetros de la red neuronal ..... 111
3.12.2. Etapa de aprendizaje ..... 112



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

3.13. Circuito de control ..... 113
3.13.1. Circuito general ..... 113
3.13.2. Control de motor CD ..... 115
3.13.3. Nivel de la batería ..... 116
3.14. Aplicación Android ..... 117
**CAPÍTULO IV ..... 124**
**10. DISEÑO DE LA PRÓTESIS ..... 124**
4.1. Diseño mecánico de la prótesis ..... 124
4.1.1. Mano ..... 124
4.1.2. Muñeca ..... 126
4.1.3. Brazo ..... 127
4.1.4. Ensamble del brazo ..... 128
4.1.5. Codo ..... 129
4.2. Construcción de la prótesis ..... 132
4.2.1. Fabricación de chasis de prótesis ..... 132
4.2.2. Ensamble del brazo ..... 137
4.2.3. Incorporación de motores A y B ..... 143
4.2.4. Ensamble del codo ..... 145
4.2.5. Brazalete ..... 149
**CAPÍTULO V ..... 151**
**11. PRUEBAS Y RESULTADOS ..... 151**
5.1. Pruebas del funcionamiento ..... 151
5.1.1. Conexión Prótesis y computadora ..... 151
5.1.2. Adquisición de señales mio-eléctricas ..... 152
5.1.3. Procesamiento de datos para las señales ..... 154
5.1.4. Pruebas de la etapa de aprendizaje ..... 155
5.2. Resultados del funcionamiento ..... 157



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

5.3. Indicaciones de uso .....160
5.4. Especificaciones del prototipo .....160
5.5. Presupuesto de la prótesis .....162
**CONCLUSIONES .....164**
**RECOMENDACIONES .....166**
**REFERENCIAS .....167**
**ANEXOS .....174**
Anexo 1 Programación Principal .....174
Anexo 2 Programación Codo .....179
Anexo 3 Programación para implmentación de red neuronal perceptrón multicapa .....182
Anexo 4 Programación para control de servomotor .....184
Anexo 5 Programación Funciones auxiliares .....185
Anexo 6 Adquisición de datos de señales mio-eléctricas .....187
Anexo 7 Adquisición de señales .....188
Anexo 8 Procesar datos .....189
Anexo 9 Aprendizaje Red Neuronal .....190
Anexo 10 Red Neuronal Perceptrón Multicapa .....192
Anexo 11 Funciones de uso común .....196
Anexo 12 Diagramas de conexión eléctrica Circuito general .....198
Anexo 13 Circuito de Adquisición .....199
Anexo 14 Circuito general .....200



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

# ÍNDICE DE FIGURAS

**Figura 1.** Fisiología articular del antebrazo ..... 13
**Figura 2.** Músculo del antebrazo superficial y profunda ..... 13
**Figura 3.** Músculos anteriores y posteriores del antebrazo ..... 14
**Figura 4.** Complejo articular de la muñeca con sus huesos y articulaciones ..... 14
**Figura 5.** Relación del músculo con el movimiento ..... 16
**Figura 6.** Movimiento de los dedos ..... 16
**Figura 7.** Amputaciones extremidades superiores ..... 18
**Figura 8.** Elementos constitutivos de una prótesis ..... 21
**Figura 9.** Prótesis neumáticas ..... 22
**Figura 10.** Partes de la prótesis mio-eléctrica ..... 23
**Figura 11.** Prototipo de pinza y rotador para prótesis de mano mio-eléctrica ..... 23
**Figura 12.** Materiales utilizados para el desarrollo de una prótesis ..... 26
**Figura 13.** Etapas para la creación de la prótesis ..... 27
**Figura 14.** Procesamiento de la señal ..... 30
**Figura 15.** Ejemplo de Perceptrón simple con n = 5 entradas. ..... 35
**Figura 16.** Arquitectura de la red perceptrón multicapa ..... 38
**Figura 17.** Arquitectura de la red de neuronas de base radial ..... 43
**Figura 18.** Amplificador de instrumentación AD620 ..... 46
**Figura 19.** Paquetes de plomo ..... 47
**Figura 20.** Amplificador TL084 ..... 48
**Figura 21.** Batería Lipo ..... 49
**Figura 22.** Servomotor MG996R ..... 50
**Figura 23.** Servomotor TD8120MG ..... 51
**Figura 24.** Módulo ESP32 ..... 53
**Figura 25.** ESP32 38 Pines ..... 55
**Figura 26.** Diagrama de bloques ..... 62
**Figura 27.** Algoritmo para implementación del controlador PID ..... 72
**Figura 28.** Circuito para la adquisición de señales mio-eléctricas. ..... 73
**Figura 29.** Diagrama de bloques de la etapa de instrumentación. ..... 73
**Figura 30.** Amplificador de instrumentación AD620. ..... 75
**Figura 31.** Integrado TC7660. ..... 76



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

**Figura 32.** Circuito de amplificación .....77
**Figura 33.** Circuito de filtro pasa bajas .....79
**Figura 34.** Circuito del filtro pasa altas.....80
**Figura 35.** Circuito del filtro notch. .....82
**Figura 36.** Circuito del amplificador variable. .....83
**Figura 37.** Circuito del rectificador.....85
**Figura 38.** Algoritmo principal para el funcionamiento de la prótesis .....91
**Figura 39.** Código 1 Función principal para el funcionamiento de la prótesis. .....92
**Figura 40.** Diagrama de flujos del programa.....93
**Figura 41.** Diagrama de bloques general del programa .....93
**Figura 42.** Código 2.1 Clase servo.....95
**Figura 43.** Código 2.2 Clase PID.....96
**Figura 44.** Código 2.3 Clase codo .....97
**Figura 45.** Código 2.4 Clase para la red neuronal perceptrón multicapa.....98
**Figura 46.** Ubicación de electrodos. .....99
**Figura 47.** Implementación del circuito de adquisición de señales mio-eléctricas. .....100
**Figura 48.** Señal amplificada por el AD620 (etapa A). .....100
**Figura 49.** Señal obtenida por la operación de filtrado .....101
**Figura 50.** Señal obtenida por la etapa de amplificación variable. .....102
**Figura 51.** Señal de salida. .....102
**Figura 52.** Señal de salida. .....103
**Figura 53.** Diagrama eléctrico de la conexión de la señal mio-eléctrica en el ESP32.....105
**Figura 54.** Señales mio-eléctricas adquiridas por el ADC del ESP32. .....106
**Figura 55.** Señales mio-eléctricas adquiridas por ADC del ESP32.....106
**Figura 56.** Señales mio-eléctricas generadas para mover las articulaciones de la prótesis .....108
**Figura 57.** Señales mio-eléctricas obtenidas de la etapa de procesamiento.....109
**Figura 58.** Comparación de señales mio-eléctricas .....110
**Figura 59.** Señales normalizadas a valores -1 y 1.....111
**Figura 60.** Topología de red neuronal empleada .....112
**Figura 61.** Error obtenido durante la etapa de aprendizaje. .....112
**Figura 62.** Circuito de control de la prótesis. .....114
**Figura 63.** Grafica de respuesta de motor obtenida en base a observación (el tiempo es aproximado).....115



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

**Figura 64.** Entorno de desarrollo App Inventor.....117
**Figura 65.** Programa para aplicación Android. .....118
**Figura 66.** Primera parte del programa para la aplicación Android .....119
**Figura 67.** Segunda parte del programa para la aplicación Android .....120
**Figura 68.** Tercera parte del programa para la aplicación Android. .....120
**Figura 69.** Proceso de instalación de la aplicación en el móvil. .....121
**Figura 70.** Diagrama de flujos de la aplicación Android. .....121
**Figura 71.** Diagrama de bloques de la aplicación Android. .....122
**Figura 72.** Diseño general de la prótesis. .....124
**Figura 73.** Diseño de los dedos que componen la mano. .....125
**Figura 74.** Diseño de la palma de la mano. .....125
**Figura 75.** Diseño de los dedos que componen la mano. .....126
**Figura 76.** Pieza para emular el movimiento de la muñeca. .....126
**Figura 77.** Pieza de acoplamiento para la muñeca. .....127
**Figura 78.** Pieza para emular el brazo. .....127
**Figura 79.** Pieza de acoplamiento para corona de servomotor de la muñeca. .....128
**Figura 80.** Pieza de acoplamiento del codo. .....128
**Figura 81.** Representación del ensamble del brazo. .....129
**Figura 82.** Vista explosionada del ensamble del brazo. .....129
**Figura 83.** Chasis para el motor del codo. .....130
**Figura 84.** Representación del ensamble del brazo. .....130
**Figura 85.** Representación del ensamble del brazo. .....131
**Figura 86.** Representación del ensamble del brazo. .....131
**Figura 87.** Diseño de mano realizado; (b) segundo diseño. .....132
**Figura 88.** Segundo diseño de mano realizado; (a) falanges de dedos de la mano; (b) ensamble de piezas que componen la mano. .....133
**Figura 89.** Materiales utilizados para el ensamble de la mano. .....133
**Figura 90.** Segundo diseño de muñeca realizado. .....134
**Figura 91.** Cople para muñeca y brazo. .....135
**Figura 92.** Segundo diseño de muñeca realizado. .....136
**Figura 93.** Piezas que componen el mecanismo del codo en la prótesis. .....137
**Figura 94.** Descripción del ensamble. .....138
**Figura 95.** Ensamble entre pieza muñeca y cople. .....138
**Figura 96.** Resultado de ensamble entre cople y muñeca. .....139



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

**Figura 97.** Ensamble entre piezas muñeca y brazo. ..... 139
**Figura 98.** Ensamble entre piezas muñeca y brazo. ..... 140
**Figura 99.** Cople para muñeca y brazo. ..... 141
**Figura 100.** Vista interna de la pieza brazo que permite visualizar las tuercas empleadas para la sujeción de la pieza cople codo. ..... 141
**Figura 101.** Cople codo acoplado a pieza brazo. ..... 142
**Figura 102.** Cople para eje del motor-reductor. (a) Pieza cople eje motor-reductor; (b) motor-reductor y cople codo. ..... 142
**Figura 103.** Resultado del ensamble de las piezas mano, muñeca, brazo y cople codo. ..143
**Figura 104.** Servomotor para otorgar el movimiento de la muñeca (pronación y supinación). ..... 144
**Figura 105.** Circuito para incorporar dos sensores de efecto Hall. ..... 145
**Figura 106.** Ensamble de motor-reductor para emular movimientos del codo (flexión y contracción) y circuito de potencia. ..... 145
**Figura 107.** Circuito auxiliar. ..... 146
**Figura 108.** Circuitos de control y potencia en el chasis del codo de la prótesis. ..... 146
**Figura 109.** Circuitos de control y potencia en el chasis del codo de la prótesis. ..... 147
**Figura 110.** Ensamble de las piezas que componen el codo de la prótesis. ..... 147
**Figura 111.** Dimensión de las piezas que componen el brazo y mano de la prótesis. ..... 148
**Figura 112.** Dimensión de motor-reductor y codo ..... 148
**Figura 113.** Prótesis ensamblada. ..... 149
**Figura 114.** Piezas que componen el brazalete. ..... 149
**Figura 115.** Conectores de electrodos colocados sobre el brazalete. ..... 150
**Figura 116.** Brazalete construido. ..... 150
**Figura 117.** Programas y Driver para reconocimiento de ESP32. ..... 151
**Figura 118.** Ajuste de interprete y conexión del puerto en el programa Thonny. ..... 152
**Figura 119.** Programas Thonny y Spyder para la adquisición de datos. ..... 153
**Figura 120.** Sincronización de Bluetooth y Adquisición de señales en Programa Spyder. ..... 153
**Figura 121.** Adquisición de señales y datos en archivo Excel. ..... 154
**Figura 122.** Adquisición de la señal sin procesar. ..... 155
**Figura 123.** Adquisición de la señal procesada. ..... 155
**Figura 124.** Gráficas de aprendizaje (a)No normalizados (b)Normalizados. ..... 156
**Figura 125.** Gráfica del error del aprendizaje. ..... 156



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

**Figura 126.** Toma de los pesos sinápticos.....157
**Figura 127.** Resultado de movimientos de la mano.....158
**Figura 128.** Resultado de movimientos de la muñeca. .....158
**Figura 129.** Resultado de movimientos del codo. .....159
**Figura 130.** Interfaz de la aplicación Android conectándose y ejecutándose. .....159



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

## ÍNDICE DE TABLAS

**Tabla 1** Operacionalización de las variables. .....6
**Tabla 2** Características de amplificador de instrumentación AD620. .....46
**Tabla 3** Características amplificador TL084. .....48
**Tabla 4** Especificaciones servomotor MG996R. .....50
**Tabla 5** Especificación servomotor TD8120MG. .....51
**Tabla 6** Variedad de interfaces externas de módulo ESP32. .....52
**Tabla 7** Especificaciones ESP32 38 Pines. .....55
**Tabla 8** Cuadro comparativo de redes neuronales .....69
**Tabla 9** Frecuencias de trabajo para filtros. .....82
**Tabla 10** Ángulo de movimiento para motores .....96
**Tabla 11** Rangos de adquisición de datos .....106
**Tabla 12** Tabla de distribución de frecuencias obtenidas de las señales para mover la mano. .....108
**Tabla 13** Tabla de distribución de frecuencias obtenida de las señales para mover la mano. .....109
**Tabla 14** Especificaciones tecnicas del equipo .....161
**Tabla 15** Presupuesto de la prótesis. .....162



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

# INTRODUCCIÓN

La presente investigación muestra una alternativa para todas aquellas personas con la ausencia de una extremidad superior (brazo), la implementación de un sistema robótico biomédico para que el control de los movimientos de una prótesis mio-eléctrica transhumeral sean factibles y así facilitar las acciones que se suelen realizar en la vida cotidiana y permitir la integración en su entorno laboral, social, etc.

Para el desarrollo de este proceso, en el capítulo I, se realiza un estudio metodológico, planteando el problema principal, justificando el estudio, delimitando los principales objetivos a los que nos enmarcamos y los principales alcances además de los medios que contamos para la realización de la presente.

En el capítulo II, se realiza un estudio de antecedentes a la presente investigación; tanto a nivel internacional como nacional, en el cual se evidencia la situación de las prótesis de extremidades superiores en otras partes del mundo y del país, sobre todo el complicado manejo que tienen las personas que hacen uso de estas y las limitadas acciones que pueden realizar. Por otro lado, observarnos la identificación de las principales variables delimitadas. Partiendo de la explicación de las principales bases teóricas correspondientes a las dimensiones e indicadores y la conceptualización de los términos básicos utilizados en la investigación.

En el capítulo III, se analiza la metodología y software de una prótesis, con la explicación de sus principales componentes; prosiguiendo con la metodología del diseño: programaciones, redes neuronales, diseño de los circuitos, filtros, cálculos, entre otros, seguido por el capítulo IV diseño del hardware de la prótesis.

Finalmente se presenta en el capítulo V los resultados de las pruebas de funcionamiento por medio de la tecnología, mostrando los principales resultados alcanzados mediante dispositivos y dando indicaciones de uso.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

# CAPÍTULO I

## 1. PLANTEAMIENTO DE LA INVESTIGACIÓN

### 1.1.Enunciado del problema

"Diseño e implementación de un sistema robótico biomédico para controlar los movimientos de una prótesis mio-eléctrica transhumeral."

### 1.2. Identificación del problema

"La discapacidad no debería ser un obstáculo para el éxito"

Stephen W. Hawking.

Gracias a la locomoción corporal, es que podemos realizar actividades laborales, cotidianas y de dispersión, pero existe un grupo poblacional que, por causas traumáticas o no traumáticas han perdido un miembro superior, viendo limitada su capacidad de ejecución de distintas labores cotidianas, lo que repercute en su vida personal, laboral y social. Por otro lado, las tecnologías electrónicas avanzan a pasos agigantados, pudiendo ofrecer alternativas y complementos que permitan a esta población incorporarse con mayor facilidad al mundo profesional.

Todos los seres humanos, sin y con discapacidad, tenemos los mismos derechos respecto a oportunidades e inclusión, en los últimos años, muchos estados, instituciones y organizaciones a nivel mundial han incidido en este punto (población con discapacidad), dando normativas de inclusión laboral, o diseño de infraestructura urbana, entre otros programas que buscan facilitar su atención, desenvolvimiento y desplazamiento.

El Instituto Nacional de Rehabilitación (INR) entre el 2002 y 2007, hizo un estudio demográfico sobre el paciente amputado en Perú, dando como cifra un total de 570 pacientes registrados con amputaciones, de los cuales, 445 corresponden a miembros inferiores y 125 corresponden a miembros superiores, y si bien es cierto el mayor índice de amputación es



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

de miembros inferiores (Farro et al, 2012), también se ve en el mercado más alternativas de prótesis, inclusive las sillas de ruedas, pero nos llamó la atención los datos del INEI, sobre el porcentaje de PEA (población económicamente activa) de la población con discapacidad, siendo más de la mitad (55.2%) de personas que no trabajan ni buscan empleo, y solo el 43% está empleada y el 1.8% buscando empleo (Carhuavilca e INEI, 2020).

Es así, como deducimos que si a las personas con amputación de miembros superiores, se les facilita realizar acciones cotidianas, como sostener objetos, utilizar de forma continua y fluida la computadora, celular o aparatos digitales, electrónicos, ellos podrían tener mejores desenvolvimientos y por ende conseguir o generarse un empleo.

### 1.3. Descripción del problema

En la actualidad, al tocar temas relacionados con las prótesis se evidencia cierta apatía de los mismos usuarios e inclusive al utilizar una, a pesar de ser algo normal en personas que tienen amputación de alguno de los dos miembros superiores; aunque años atrás los accidentes terminaran en amputación eran poco vistos, hoy en día, gracias a la tecnología, la innovación, el desarrollo y fabricación de las prótesis ha mejorado considerablemente, primeramente porque, hay una mayor aceptabilidad por parte de los usuarios, ya que, son más accesibles dependiendo el diseño, la fabricación de la misma y sobretodo porque mejora la calidad de vida del individuo, esto se ha constatado en estudios que se han realizado anteriormente que han mejorado el funcionamiento de las sensaciones de fuerza, deslizamiento y manipulación de las prótesis de miembros superiores sin causar ningún tipo de incomodidades en los usuarios al momento de utilizarlas.

En este estudio queremos diseñar e implementar un sistema robótico biomédico para una prótesis mio-eléctrica transhumeral logrando un factible manejo de sus movimientos para mejorar la calidad de vida de usuarios que han sufrido amputaciones del miembro superior, pudiendo con ella hacer pruebas que sean satisfactorias en cuanto a movimientos



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

se refiere, para mejorar la funcionabilidad y destreza de los usuarios, reconociendo todas las sensaciones asociadas al movimiento consiguiendo con ello una mejor aceptación por parte de la sociedad.

## 1.4. Formulación del problema

### *1.4.1. Problema general*

¿De qué manera se puede diseñar e implementar un sistema robótico biomédico para una prótesis mio-eléctrica transhumeral logrando un factible manejo de sus movimientos?

### *1.4.2. Problemas específicos*

- ¿Qué componentes debe contemplar el diseño de una prótesis mio-eléctrica transhumeral tomando en cuenta el tipo de paciente de prueba, las dimensiones, el tamaño y el peso total de carga?
- ¿Cómo se puede construir una prótesis mio-eléctrica transhumeral que integre un circuito de adquisición y acondicionamiento de señales, tarjeta controladora, motores, y batería?
- ¿Qué grado de movilidad posee la prótesis mio-eléctrica transhumeral en cada una de sus secciones, los grados de movilidad necesarios para el uso de TIC (Tecnologías de Información y Comunicación) y algunas actividades cotidianas?
- ¿Cuáles son los sistemas y dispositivos robóticos biomédicos, que pudiesen ser adaptados a la prótesis mio-eléctrica transhumeral?
- ¿De qué manera se puede integrar el sistema de retroalimentación háptico, con la prótesis mio-eléctrica transhumeral, para controlar sus movimientos?
- ¿Qué aplicativo móvil se podría diseñar para el control permanente de la prótesis?



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### 1.5.Área de conocimiento

El área del conocimiento a ejecutar se centra en el diseño e implementación de un sistema robótico biomédico para controlar los movimientos de una prótesis mio-eléctrica transhumeral.

- **Área:** Ingeniería electrónica, ingeniería mecánica, ingeniería mecatrónica, ingeniería de sistemas.
- **Línea:** Robótica.

### 1.6. Justificación de la investigación

La presente tesis utiliza software libre sin cargos de costos por uso de programa, reduciendo el costo final. Presenta un microcontrolador Esp32 con una red neuronal capaz de procesar las señales adquiridas y, son de bajo costo. Utiliza piezas mecánicas hechas con PLA lo que la hace más ligera y menos costosa.

De la misma manera, este tema beneficiará a todas aquellas personas que han tenido algún accidente comprometiendo alguno de sus miembros superiores (Brazos), permitiendo que, puedan optar a prótesis mio-eléctricas para mejorar su calidad de vida.

Por otro lado, se espera que la presente investigación sirva de guía y antecedente para futuras investigaciones relacionadas al tema, además, sea una fuente de información para implementar programas y proyectos que se ajusten en realzar la importancia de las prótesis mio-eléctricas transhumerales en nuestra región.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## 1.7. Objetivos de la investigación

### *1.7.1. Objetivo general*

Diseñar e implementar una prótesis mio-eléctrica transhumeral basada en inteligencia artificial, utilizando redes neuronales para identificar el tipo de señal mio-eléctrica recibida de una parte del cuerpo del usuario y así mover la articulación correspondiente.

### *1.7.2. Objetivos específicos*

- Realizar el diseño y fabricación de las piezas que componen la prótesis, considerando la dimensión (tamaño) del brazo derecho de una persona adulta.
- Implementar el algoritmo de la red neuronal perceptrón multicapa en el microcontrolador ESP32 utilizando el lenguaje de programación microPython, e implementar el circuito de adquisición de señales.
- Capturar las señales mio-eléctricas del usuario, para el entrenamiento y validación de la red neuronal, validando el funcionamiento del modelo de la prótesis propuesto.
- Determinar el grado de libertad de la prótesis mio-eléctrica transhumeral en cada una de sus secciones, los grados de libertad necesarios para el uso de algunas actividades cotidianas.
- Analizar los distintos sistemas y dispositivos robóticos biomédicos, que pudiesen ser adaptados a la prótesis mio-eléctrica transhumeral.
- Definir los componentes para el diseño de una prótesis mio-eléctrica transhumeral tomando en cuenta el tipo de paciente de prueba, las dimensiones, el tamaño y el peso total de carga.
- Diseñar un aplicativo móvil para el control permanente de la prótesis.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## 1.8. Alcances de la investigación

En esta investigación se obtendrá, el diseño y la implementación de sistema robótico biomédico para una prótesis mio-eléctrica transhumeral con tres grados de libertad (codo, muñeca y mano) logrando un factible manejo de sus movimientos, de esta manera le otorgará movilidad a la mano la cual abrirá y cerrará siendo tipo pinza, la muñeca que tendrá el movimiento de pronación y supinación de 180° y el codo que tendrá el movimiento de contracción y flexión; asimismo contará con un aplicativo que recibirá la señal vía Bluetooth desde la prótesis mediante la tarjeta ESP32, permitiendo al usuario llevar el control del estado de la batería, humedad, temperatura y estado de los servomotores. El tipo de investigación es aplicada.

## 1.9. Variables de la investigación

### *1.9.1 Variable independiente*

- Sistema robótico biomédico.

### *1.9.2. Variable dependiente*

- Prótesis mio-eléctrica transhumeral.

### *1.9.3. Operacionalización de las variables*

***Tabla 1*** *Operacionalización de las variables.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

| | VARIABLES | DEFINICIÓN CONCEPTUAL | DIMENSIONES | INDICADORES |
|---|---|---|---|---|
| INDEPENDIENTE | Sistema robótico biomédico | Sistema de comunicación mediante la percepción y el movimiento del usuario. | Sistema de comunicación | • Sistema inalámbrico<br>• Interacción humana/ objeto.<br>• Retroalimentación<br>• Patrón de movimientos. |
| | | | Tecnología háptica | • Tacto y la piel.<br>• Contacto y sensaciones<br>• Receptores sensoriales nerviosos.<br>• Movimientos. |
| | | | Dispositivos hápticos | • Retroalimentación de fuerza<br>• Retroalimentación táctil<br>• Vibradores electromagnéticos<br>• Dispositivo móvil. |
| DEPENDIENTE | Prótesis mio-eléctrica transhumeral. | Prótesis de miembros superiores que separa el hueso del humero diseñadas para ser controladas electrónicamente por contracciones musculares por parte del usuario. | Amputación transhumeral | • Nivel de amputación corta.<br>• Nivel de amputación media.<br>• Nivel de amputación larga. |
| | | | Grados de libertad | • Sensaciones de fuerza<br>• Deslizamiento<br>• Manipulación<br>• Orientación |
| | | | Estructura de la prótesis | • Diseño<br>• Fabricación<br>• Composición (brazo, codo, mano).<br>• Princípios biomecánicos.<br>• Bateria.<br>• Alimentación externa. |
| | | | Ajustes de la prótesis | • Articulación del brazo, codo y mano.<br>• Adaptadores<br>• Elementos de ajustes. |



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

# CAPÍTULO II

## 2. MARCO TEÓRICO

### 2.1. Antecedentes de la investigación

#### *2.1.1. Antecedentes Internacionales*

- Según, Valenzuela (2016) en su tesis titulada: "Sistema de teleoperación basado en interfaz háptica para brazo robótico" concluye lo siguiente: En este estudio se incorporó un sistema de retroalimentación háptico en un brazo robótico calculando la fuerza aplicada a partir de una point cloud que se obtiene con el manejo de una cámara; está investigación tuvo como objetivo mejorar la calidad de vida entre los seres humanos y máquinas que trabajan con interfaces de usuarios mejorando la toma de decisiones, cabe destacar que en este estudio se trabajaron de la mano la retroalimentación háptica con la retroalimentación visual para alcanzar la precisión deseada para manejar un brazo robótico a través de los sensores de fuerzas, precisión y proximidad. Con esta investigación se logró proponer un sistema de teleoperación bilateral con retardo temporal basado en la retroalimentación háptica a través de la simulación. De acuerdo a la metodología de teleoperación háptica que se utilizó, los usuarios alcanzaron una mayor precisión completando la tarea de seguimiento de trayectoria en un entorno de operación desconocido. Asimismo, cuando se utiliza la retroalimentación háptica se completa la tarea más rápido, con una menor cantidad de ajustes de posición del efecto final del robot y con una menor cantidad de colisiones del robot con el entorno (Valenzuela, 2016).
- Aragón (2016), en su tesis titulada: "Diseño y Construcción de un guante Háptico con retroalimentación de fuerza" concluye lo siguiente: en esta investigación se elaboró un prototipo donde se pudo controlar el movimiento de los dedos de la mano mediante un sistema físico o virtual, en este caso un robot, siendo la mano



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

una parte esencial de movilidad en el organismo, buscando simplificar los modelos cinemáticos, para poder controlar sus movimientos dividiendo el sistema háptico del guante enfocándose en sistemas para resolver los problemas de movimientos involucrados (Aragón, 2016).

- Mancipe (2013), en su tesis titulada: "Sistema de control de fuerza e interfaz háptica para un prototipo de prótesis electromiografía de mano" concluye lo siguiente: En este estudio se tomó en cuenta un prototipo de una mano con 1 grado de libertad donde a partir de una señal electromiografía se habilita el movimiento de un servomotor lo cual permite que pueda abrir y cerrar la mano con un control que permite mantener la fuerza ejercida, para ello, se tomó en cuenta el valor de referencia, de igual manera, se logró implementar un control de fuerza en base a FPGA, también una estructura de prótesis tipo pinza, un circuito analógico de acondicionamiento junto con un lenguaje de programación, un vibrador utilizado como interfaz háptica, se integró el servomotor con la pinza, el sensor de fuerza, los circuitos, pudiendo así elaborar la prótesis mio-eléctrica de mano con un costo 40 veces inferior a una prótesis comercial, aunque es necesario para trabajos futuros mejorar las características (Mancipe, 2013).

### *2.1.2. Antecedentes nacionales*

- Según, Romero (2018) en su tesis titulada: "Diseño de un sistema de retroalimentación háptico para una prótesis mio-eléctrica transradial del miembro superior" concluye lo siguiente: el diseño de prótesis en la actualidad ha venido evolucionando con el pasar de los años, esto con la finalidad de ayudar a mejorar la calidad de vida de los seres que por diversos accidentes en los que se han visto inmersos que les haya producido algún tipo de discapacidad o simplemente han tenido alguna amputación de algún miembro superior, sin embargo, en muchos



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

estudios se ha visto la necesidad de actualizar a las personas que recurren a esta prótesis, ya que, muchos han presentado problemas en el manejo de la misma, esto es debido a los sistemas de control que se usan actualmente que se encuentran ajenos a la participación del usuario, asimismo, esto ha causado dificultades debido a que el usuario que ha recibido la prótesis no puede manipularla de la forma que quisiera por la fuerza aplicada que no está acorde a lo que necesita la prótesis; en este estudio se propuso hacer un diseño de un sistema háptico no invasivo que desarrollará una prótesis mio-eléctrica permitiendo así, al usuario de la misma obtener la sensación de fuerza y deslizamiento que le permitiese controlar la prótesis de mano, fueron muchos diseños a los cuales se tuvo que recurrir para la elaboración de los requerimientos que queríamos abarcar, una vez, teniendo el diseño se hizo una simulación para evaluar el manejo de la misma, también se evaluaron presupuestos de diseño y fabricación para estimar la viabilidad del estudio. De la misma manera, el diseño se encontró acondicionado al antebrazo de la persona que se tomó como ejemplo para el estudio, también se puede hacer énfasis en que la carcasa de la prótesis es tan sólida que permite un adecuado funcionamiento superando los 12 MPa, En ambos se obtiene un valor mínimo de factor de seguridad de 1.7; este valor es superior a 1.5 por lo que es considerado válido. En cuanto a la calibración de la fuerza a utilizar la curva obtenida tuvo un comportamiento lineal que coincide con lo requerido por el fabricante en los datos del sensor, ahora bien, finalmente, se corroboró que el funcionamiento de la prótesis fue satisfactorio, ya que, se realizaron las pruebas pertinentes sobre personas que mostraron que el actuador funciona y no causa incomodidades en los usuarios y tiene intensidad de vibración alta (Romero, 2018).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- Llantoy (2020), en su tesis titulada: Diseño e implementación del sistema eléctrico para una prótesis transradial mio-eléctrica, concluye que: En años anteriores las prótesis de brazos y manos no eran tan estudiadas debido a la poca información que se obtenía de ellas pero con los años de acuerdo a los avances de la tecnología, se han producido avances innovadores con respecto a este tipo de prótesis, ahora el problema que ha aparecido es que los usuarios no saben utilizar ni manejar los movimientos de este tipo de prótesis, ya que no cuentan con un sistema eléctrico que permita un mejor manejo del mismo, es por ello que este estudio se basó en el diseño e implementación de un sistema electrónico de bajo costo para una prótesis transradial mio-eléctrica que permita al usuario mejorar su efectividad en la ejecución de los gestos deseados en su prótesis de mano obteniendo resultados positivos, ya que, se pudo incluir al mercado un sistema eléctrico para una prótesis transradial mio-eléctrica con un costo de S/ 2475,22 (Llantoy, 2020).
- Sullcahuamán (2013) en su tesis titulada: "Diseño mecánico de un prototipo de prótesis mio-eléctrica transradial" concluye lo siguiente: El objetivo general de este estudio fue diseñar y fabricar un prototipo de prótesis para personas con amputación por debajo del codo y que pudiese soportar 500grs., el prototipo tuvo un peso de 350grs. Rango promedio con la mano humana por lo que se elaboraron para la simulación 2 de los 5 dedos de la mano, para ejercer movimiento se utilizó un grado de libertad y un sistema de engranajes, la parte mecánica se adaptó con el control del prototipo siendo ayudado por un cilindro mecánico asumiendo señales eléctricas emitidas por el control (Sullcahuamán, 2013).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## 2.2. Bases teóricas

### *2.2.1. Fisiología del brazo*

Hace énfasis al comportamiento que tiene el miembro, su anatomía y los movimientos que este realiza mediante sus articulaciones (Romero, 2018).

#### 2.2.1.1. Antebrazo

En cuanto a la fisiología de la mano es importante señalar al antebrazo, ya que, en él se encuentran los músculos que permiten los movimientos esenciales de la muñeca y de la mano, el antebrazo, no es más que, la articulación del extremo del húmero y el extremo proximal del radio y el cúbito; existen movimientos característicos en el codo, como la flexión ejecutada por los músculos braquial y bíceps braquial y la extensión que se realiza a través de los tríceps. Movimientos que se explican en el plano sagital (Romero, 2018).

#### 2.2.1.2. Biomecánica del antebrazo

La prono supinación es el movimiento de rotación del antebrazo en torno a su eje longitudinal, para lo cual es necesaria la intervención de la asociación mecánica de dos articulaciones: la radiocubital superior y la radiocubital inferior (Huaroto, 2000).

La prono supinación es un movimiento completo, tan específico, que engloba articulaciones distantes y no es del codo ni del carpo, sino del antebrazo y se lleva a cabo a través de la integridad de la articulación húmero cubital, de la membrana interósea, de la articulación radiocubital distal, del fibrocartílago triangular y gracias a la especial forma del cúbito (recta) y el radio (curvada). En la supinación intervienen el supinador corto y el bíceps; en la pronación actúan el pronador cuadrado y el pronador redondo (Huaroto, 2000).

#### 2.2.1.3. Anatomía del antebrazo

Los músculos del antebrazo están divididos en múltiples grupos, los cuales originan el movimiento de los dedos y de la muñeca. Estos grupos son los flexores anteriores, donde la mayoría son inervados por el nervio mediano; y los extensores posteriores, donde la mayoría son inervados por el nervio radial.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

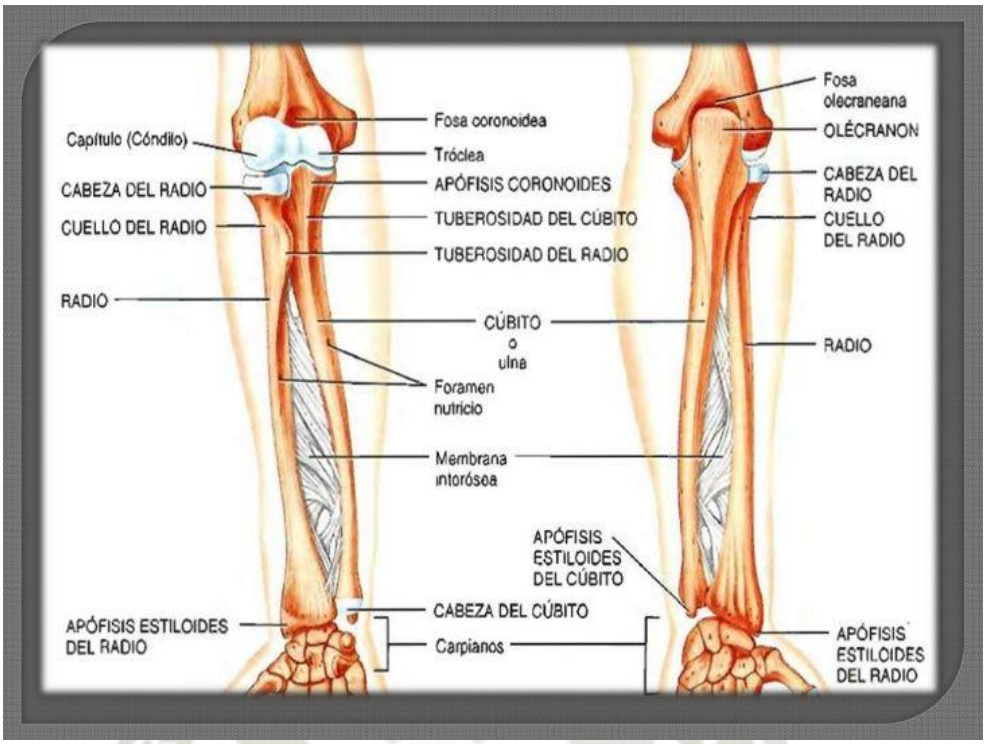



***Figura 1.*** *Fisiología articular del antebrazo*
***Nota,*** *Rostro, (2021)*

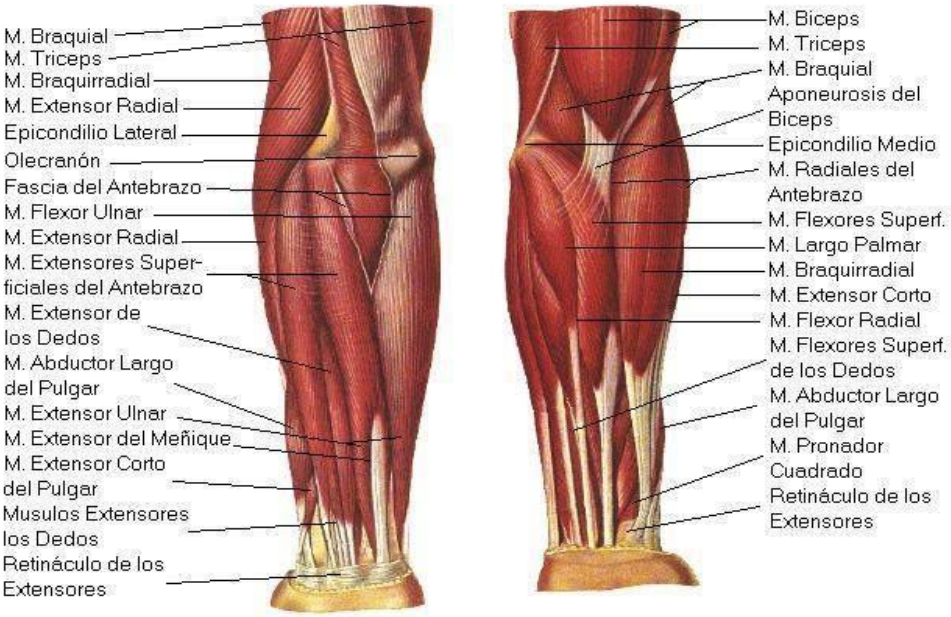



***Figura 2.*** *Músculo del antebrazo superficial y profunda*
***Nota,*** Verastegui et al, (2019)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

| | CAPA | MÚSCULO | IMAGEN | ACCIÓN |
|---|---|---|---|---|
| FLEXORES MÚSCULOS | SUPERFICIAL | Flexor Radial del carpo |  | Flexiona y separa la mano mediante la muñeca |
| MÚSCULOS EXTENSORES POSTERIORES | SUPERFICIAL | Extensor radial largo del carpio |  | Extiende y separa la mano mediante la muñeca |
| | | Extensor radial corto del carpo |  | |

***Figura 3.*** *Músculos anteriores y posteriores del antebrazo*
**Nota,** Verastegui et al, (2019)

### 2.2.1.4. Muñeca

Observando la ubicación de la muñeca, la misma no se encuentra en el extremo discal del antebrazo, su ubicación precisa en la palma de la mano, justo en los músculos del pulgar y el meñique, esta parte del brazo realiza movimientos de flexión y extensión (Romero Muñiz, 2018).

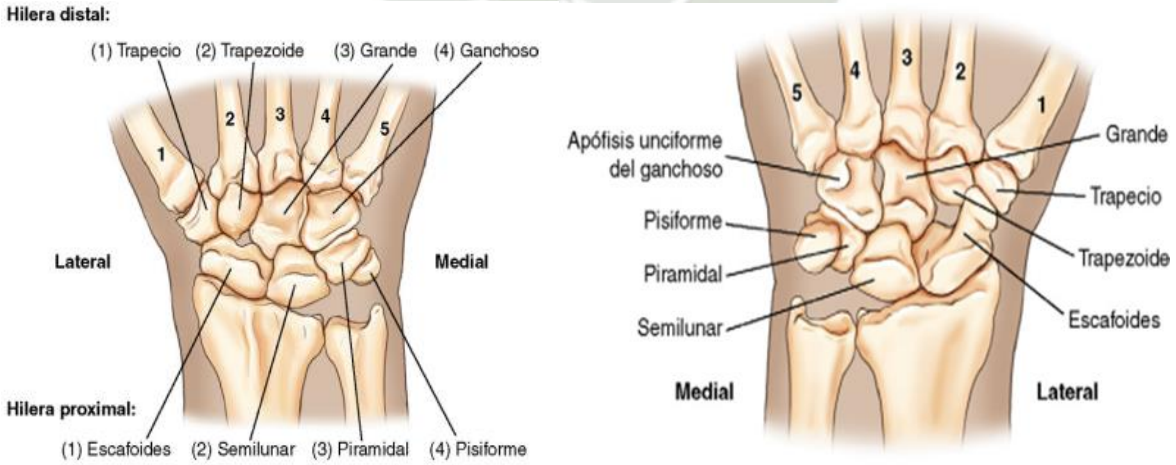



***Figura 4.*** *Complejo articular de la muñeca con sus huesos y articulaciones*
***Nota,*** GoConqr, (2022)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

**2.2.1.4.1. Biomecánica de la muñeca, rango de movilidad articular**

En relación con los elementos biomecánicos en el complejo articular de la muñeca, considerando el conjunto de huesos, articulaciones y músculos que lo forman, se observa que este sistema permite el desarrollo de distintos tipos de movimientos complejos en diferentes planos del espacio. Estos movimientos se han ido organizando como consecuencia del desarrollo filogenético, por lo que, con el tiempo y las necesidades funcionales de la mano, se han logrado movimientos en flexo-extensión, aducción-abducción y rotación axial (prono-supinación). Estos movimientos se producen en el marco de una conveniente estabilidad de la articulación, lográndose con esto una adecuada funcionalidad de la mano como órgano vital en el desarrollo evolutivo del *homo sapiens*.

Los movimientos de la muñeca, se logran con una gran estabilidad de las articulaciones, combinándose control de la fuerza y precisión lo que permite alcanzar objetivos concretos. Estos movimientos son posibles por el trabajo conjunto de las articulaciones antes mencionadas, o sea, radiocubital distal, radiocarpiana, mediocarpianas, intercarpianas y carpometacarpianas.

La estabilidad viene dada por la capacidad para mantener una relación normal entre los diferentes huesos que conforman el complejo articular y los tejidos blandos que los limitan, bajo una carga fisiológica en todo el arco de movilidad. De esa manera, para que estas u otras articulaciones sean funcionales, deben de tener un arco de movilidad funcional estable, lo que implica transferir cargas fisiológicas sin generar estrés anormal en el cartílago articular y un movimiento en todo su rango sin alteraciones súbitas de la alineación de sus huesos. (Medina et al, 2016)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

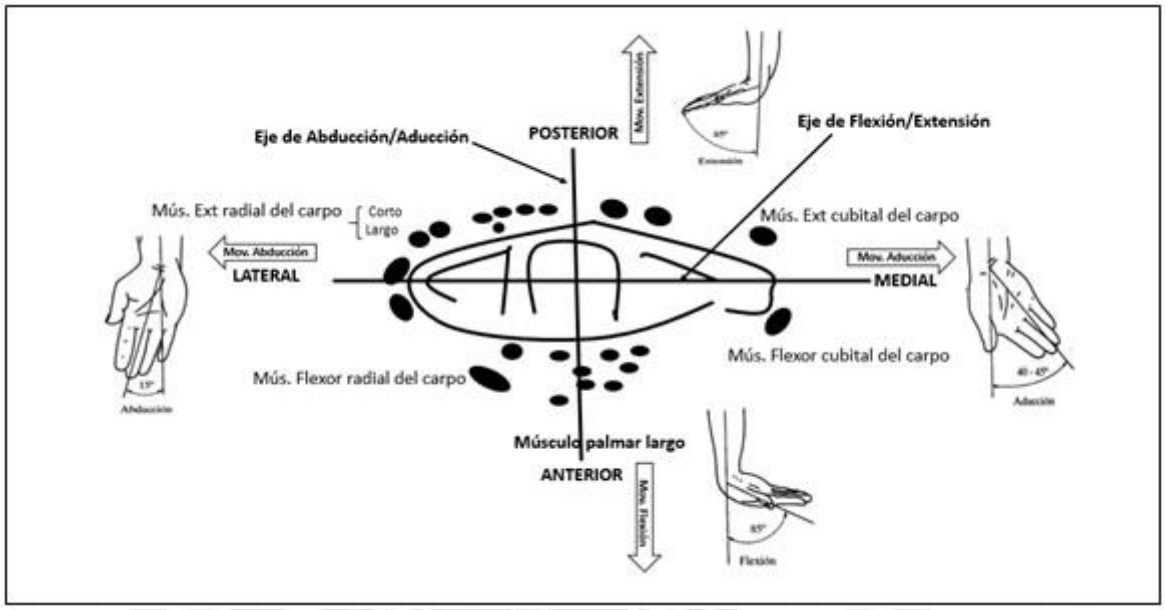



***Figura 5.*** *Relación del músculo con el movimiento*
***Nota,*** *Medina et al, (2016)*

### *2.2.1.5. Mano*

Comprende los dedos pulgares, índice, medio, anular y meñique con tres articulaciones que generan los movimientos de los mismos. Comprenden articulaciones metacarpofalángicas que permiten la realización de dos movimientos flexión y extensión, las articulaciones interfalángicas ejercen una función más limitada (Romero, 2018).

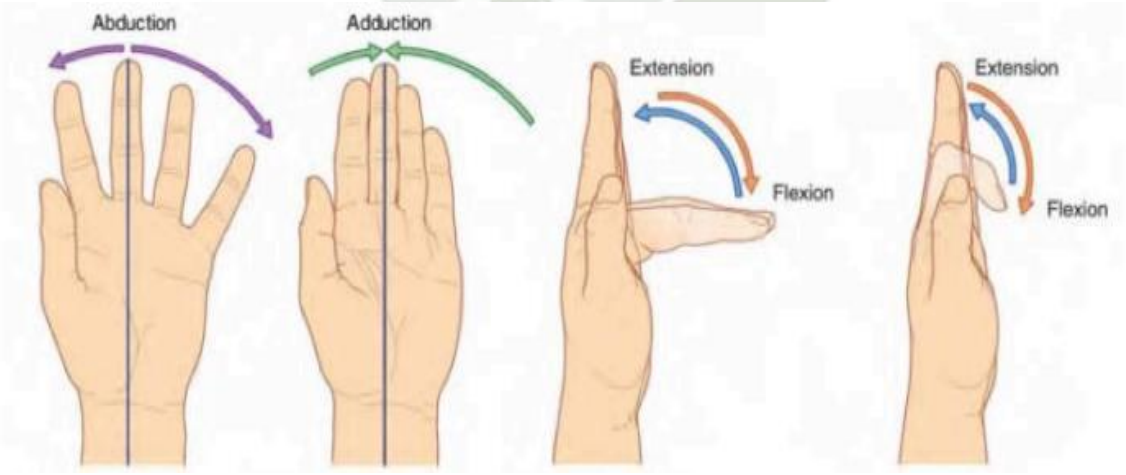



***Figura 6.*** *Movimiento de los dedos*
***Nota,*** *Drake et al, (2015)*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### *2.2.2. Fisiología en amputados de brazo*

#### 2.2.2.1. Amputación

Es la pérdida total o parcial de una extremidad; consiste en la separación completa de una parte del organismo del cuerpo (Pérez y Aguilar, 2014), cuando la amputación se da a nivel de una articulación se llama desarticulación, las amputaciones pueden ser causadas por:

- Enfermedades vasculares.
- Traumatismos.
- Infección.
- Tumores.
- Defectos congénitos (Pérez, 2004).

En los casos de amputaciones parciales se producen de manera excesiva hemorragias, ya que los vasos se seccionan solo en una parte provocando que salga sangre por el dicho orificio, a diferencia de cuando la amputación se da de manera total, ya que los vasos se constriñen lo que hace que la hemorragia se reduzca o detenga (Pérez y Aguilar, 2014).

**Niveles de amputación**

- **Amputaciones de las manos y los dedos**: Generalmente son causadas por traumatismos en accidentes laborales, siendo el pulgar el dedo más importante, por lo que la prioridad es conservar su longitud y función posterior a la lesión.
- **Desarticulación de la muñeca**: se considera que dicho nivel de amputación es de mala protetización.
- **Amputación a nivel del antebrazo**: son comúnmente conocidas como amputaciones debajo del codo, esta amputación facilita la implementación de



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

prótesis mio-eléctrica ya que conserva grandes grupos musculares y los tendones son suturados de tal forma que mantienen una ligera tensión.

- **Desarticulación del codo**: no es considerado un buen nivel de amputación.
- **Amputación a nivel del brazo**: conocida comúnmente como amputación por encima del codo, puede ir desde el cuello quirúrgico del húmero hasta el tercio distal de este. (Pérez, 2004).

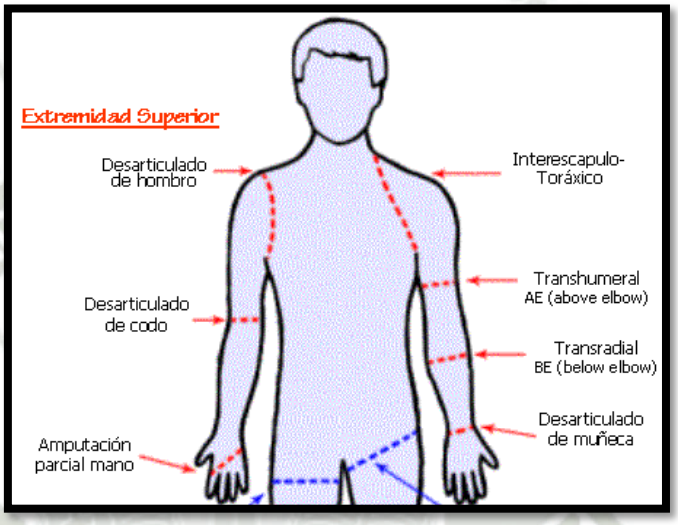



***Figura 7.*** *Amputaciones extremidades superiores*
***Nota,*** *Verastegui et al, (2019)*

### 2.2.2.2. Dolor de mano fantasma en amputados

Es un dolor que sienten las personas que han sufrido amputaciones como si proviniera de la parte del cuerpo amputado; inicialmente los médicos veían dicha sensación como un problema netamente psicológico, actualmente reconocen que dichos dolores se experimentan de manera real y son originados en la médula espinal y el cerebro; dichos dolores pueden mejorar con el tiempo si la persona es sometida a tratamientos, aunque para algunos ciudadanos controlar el dolor representa todo un desafío; cuando las personas han sufrido alguna amputación y llegan a sentir dolor o manifiestan sentir que dicha parte del cuerpo todavía existe, a esto se llama sensación de miembro fantasma (MayoClinic, 2020).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### *2.2.3. Prótesis*

Según el diccionario médico de la Clínica Universidad de Navarra, la prótesis es la sustitución ya sea de una parte del cuerpo o de un órgano, por un implante especial, la cual es reproducida de una manera casi idéntica a la parte que va a sustituir; para la ortopedia corresponde al aparato que sustituirá una extremidad faltante ya sea de manera parcial o completa (párr. 1). De igual manera para la Organización Mundial de la Salud las prótesis son dispositivos cuya aplicación es externa, con el propósito de remplazar de manera parcial o total un miembro ausente o su funcionalidad es deficiente (OMS, 2017).

Es una extensión artificial cuyo objetivo es reemplazar una parte de cuerpo faltante ya sea por una amputación o razones congénitas, la cual cumple con las funciones de la parte que está sustituyendo, en algunos casos se usan prótesis con intenciones netamente estéticas; entre sus funciones podemos destacar:

- Restauración de la función pérdida a consecuencia de la amputación o malformación congénita.
- Restablecer la imagen.
- Reposición parcial del centro de gravedad.

#### 2.2.3.1. Clasificación de las prótesis

- **Endoprótesis**: Precisa para su colocación procedimientos quirúrgicos y su diseño es endoesqueletico.
- **Exoprótesis:** se puede retirar según la necesidad del usuario, es considerada un aparato ortopédico y es dividida en:
    - **Motoras**: Estas son ejecutadas en una acción simple.
    - **Sensoriales**: ambas extremidades informan la interacción con el entorno, de una forma automática es calculada la fuerza que se debe aplicar por el aparato



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

locomotor del individuo y es clasificada según su material de la siguiente manera:

- Plástico.
- Láminas de titanio.
- Resina.
- Metal.

Según la energía presentada, que permite los movimientos de las articulaciones se clasifican en:

- **Interna**: El individuo mueve la articulación protésica por sí mismo utilizando el aparato locomotor.
- **Externa**: sistemas mecánicos, hidráulicos o eléctricos sobre el usuario que actúan de manera indirecta y son movidos por pequeños motores cuya energía proviene de baterías.

### 2.2.3.2. Tipos de prótesis de miembros superiores

Actualmente existe una gran variedad de prótesis, pero todas están inmersas en dos categorías, la primera son las prótesis estéticas o también conocida como inactivas y las prótesis funcionales o comúnmente llamadas activas.

- **Prótesis mecánicas:** este tipo de prótesis está diseñada para realizar actividades o funciones básicas como: abrir o cerrar los dedos de la mano, encontrándose limitadas a levantar objetos livianos, se consideran imprecisas y no requieren de ningún componente eléctrico ni electrónico, su funcionamiento se basa en los movimientos de otras partes de cuerpo y se acciona mediante cuerdas o arnés, basándose en la extensión de los mismos para su apertura o cierre, y el cierre o apertura se efectúa solo con la relajación del músculo respectivamente, gracias a un resorte y la fuerza ejercida de presión o pellizco, los elementos son cubiertos con guantes con la



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

finalidad de dar una apariencia estética. El tamaño y número de ligas depende de la fuerza del material, variando de acuerdo a las necesidades propias de la persona que la utiliza, ya que sus acciones son generadas por el mismo cuerpo, es preciso que el usuario domine el movimiento general de su pecho, depresión y elevación del hombro, abducción y aducción escapular y flexión glenohumeral (Dorador, 2004).

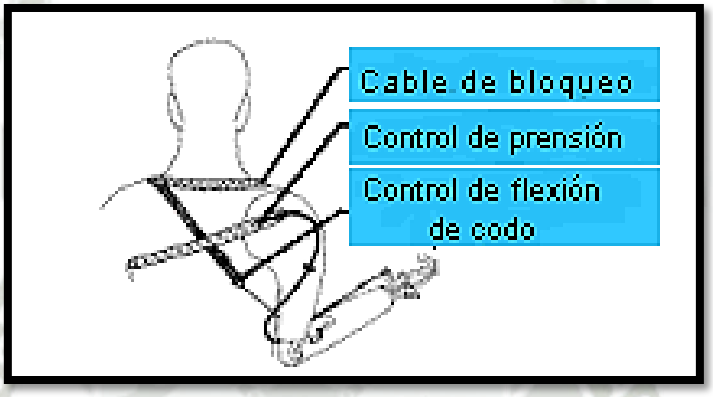



***Figura 8.*** *Elementos constitutivos de una prótesis*
***Nota,*** *Dorador, (2004).*

- **Prótesis neumáticas**: eran accionadas por ácido carbónico comprimido, con la capacidad de generar una gran cantidad de energía, que se transmitía con pérdidas mínimas, por medio de un sistema de tubos, cada una de las válvulas eran ordenadas de una manera específica, acorde a la amputación o tipo de muñón que presentaba la persona o paciente, pero al mismo tiempo presentaba dificultades debido a lo complicado de sus aparatos, accesorios y sobre todo por el riesgo de usar ácido carbónico.

  Las prótesis neumáticas hacen uso de aire a presión obtenido por medio de un compresor, su ventaja principal es proporcionar una gran fuerza y rapidez de movimientos; sus desventajas principales son los dispositivos que se implementan para su control y funcionamiento, ya que son relativamente grandes y su mantenimiento es costoso y dificultoso. (Brito et al, 2013)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA



***Figura 9.*** *Prótesis neumáticas*
***Nota,*** *Brito et al, (2013)*

- **Prótesis mio-eléctricas:** actualmente los sistemas protésicos mio-eléctricos son los que proporcionan el grado de rehabilitación más alto. Son en realidad prótesis eléctricas controladas por medio de una interfaz mio-eléctrica. Sintetizan el mejor aspecto estético con una gran fuerza y velocidad de prensión, así como varias posibilidades de combinación y ampliación. Se basan en el empleo de la señal eléctrica (EMG), que se produce al contraerse un músculo, como señal de control. Este tipo de prótesis tienen la ventaja de que solo requieren que el usuario flexione sus músculos para operarla, a diferencia de las prótesis accionadas por el cuerpo que requieren el movimiento general del cuerpo. Elimina el arnés de suspensión, usando una de las siguientes técnicas para mantener la prótesis en el lugar correspondiente: bloqueo de tejidos blandos esqueleto o succión. (Loaiza y Arzola, 2011)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

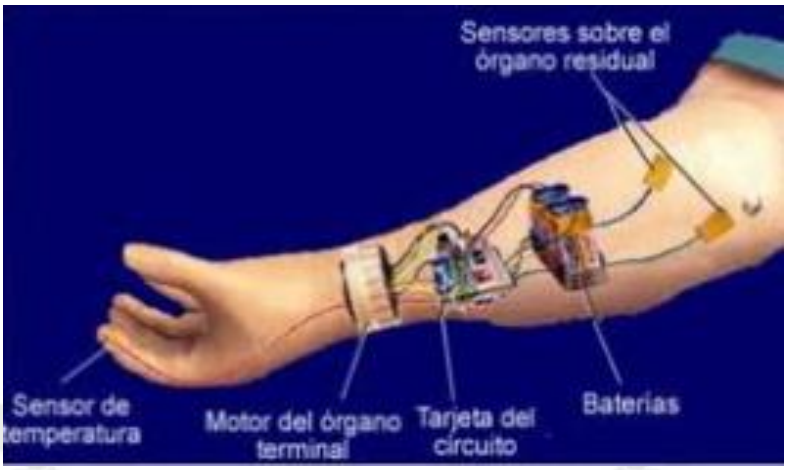



***Figura 10.*** *Partes de la prótesis mio-eléctrica*
***Nota,*** *Brito el al, (2013)*

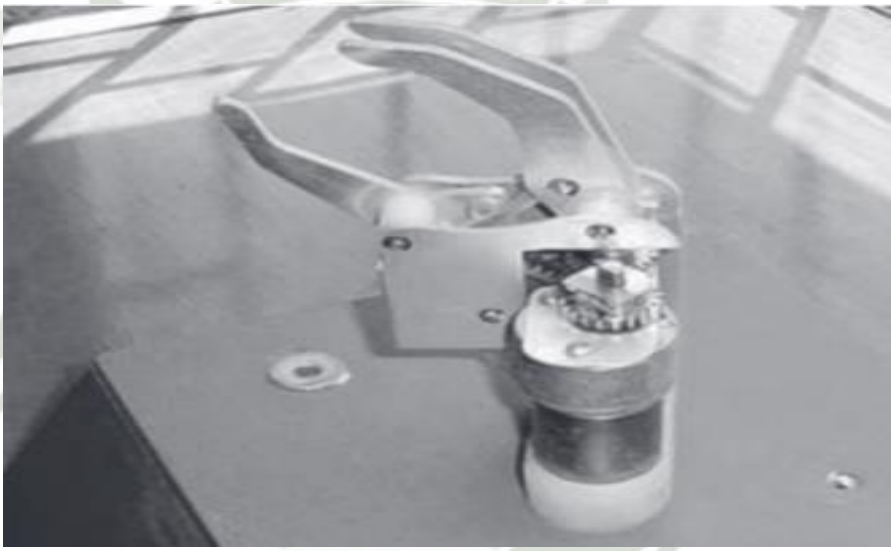


***Figura 11.*** *Prototipo de pinza y rotador para prótesis de mano mio-eléctrica*
***Nota,*** *Loaiza y Arzola, (2011)*

La prótesis mio-eléctricas tiene su origen en Rusia por los años 1960, eran controladas por potenciales eléctricos extraídos de los músculos del muñón, los impulsos eléctricos eran conducidos y amplificados; de esa forma poder energizar dichas señales y finalmente obtener el movimiento de la prótesis. Las prótesis elaboradas de una manera convencional se caracterizan por usar materiales como:

- **Silicona**: por su fácil maleabilidad y flexibilidad, es un buen aislante de electricidad, con una buena resistencia a las altas temperaturas, por lo que su tiempo de vida es extenso.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- **Acero inoxidable**: tiene una gran resistencia a los altos esfuerzos de carga.
- **Aluminio:** es un material bastante ligero, su resistencia mecánica es media, su nivel de oxidación es mínimo, con un alto coeficiente intelectual.
- **Fibra de carbón**: si bien es un material ligero tiene propiedades mecánicas elevadas.
- **Poli tetra flúor etileno (teflón) endurecido con carbón**: dicho plástico tiene niveles de resistencia bastante elevados a temperaturas extremas, su desgaste y deformación es mínimo debido a sus propiedades de carbón.
- **Espuma de poliuretano y polietileno**: tiene una gran capacidad aislante, es económico y facilidad al deformar.

En la actualidad, se han llevado a cabo nuevos trabajos con el propósito de realizar prótesis con tecnología 3D, de bajo costo, dado que es una tecnología de fácil utilización, pueda llegar a realizar estructuras con geometrías complejas bastante económicas; cabe destacar que a medida que han aumentado las investigaciones de prótesis con la tecnología 3D, también han venido desarrollando la innovación de nuevas tecnologías para poder incorporar componentes como por ejemplo electrodos los cuales se encargan de enviar en pequeñas proporciones pulsaciones eléctricas al usuario cuando toca algún objeto, lo que le permite al paciente tener la sensación de percibir texturas; también cuentan con sensores de fuerza que permite la retroalimentación al momento de manipular cualquier objeto sin llegar a romperlo por muy frágil que sea, lo que le ayudará al usuario tener un mayor control para llevar a cabo movimientos más precisos, logrando que la prótesis emule la funcionalidad de la mano.

Las opciones en relación a la variedad de prótesis están orientadas hacía un mayor control, una de esas opciones en la inteligencia artificial, razón por la cual algunas prótesis están siendo conectadas al cuerpo humano y las señales que recibe proviene de las señales



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

musculares, en el que se amerita estar procesando las señales de manera constante y finalmente, lograr originar los movimientos de las prótesis. Entre los tipos de inteligencia artificial más utilizadas tenemos: la lógica difusa y las redes neuronales, ambas tienen como propósito lograr la emulación de la toma de decisiones tal como se haría en la vida cotidiana una persona sin prótesis, siendo la más utilizada la lógica difusa, ya que la utilización computacional es mucho menor comparado con la red neuronal que requieren un gran número de parámetros para poder realizar la programación correspondiente y un efectivo entrenamiento. (Loaiza & Arzola, 2011)

#### 2.2.3.3. Prótesis de mano para personas amputadas de mano y muñeca

Los progresos recientes en el desarrollo de prótesis automáticas se deben principalmente a la incorporación de la física y la matemática, tanto a nivel de la descripción del movimiento de la mano como en el análisis de las señales mio-eléctricas. La cinemática y la cinética permiten, por medio de la teoría Newtoniana, encontrar una descripción suficientemente precisa del comportamiento de la mano humana, tanto desde el punto de vista de la geometría del movimiento como de la generación y aplicación de la fuerza a través de los músculos, tendones y estructura ósea que conforma la mano. Por otra parte, la transformada de Fourier se ha convertido en la principal herramienta para descifrar, interpretar y reproducir de forma aproximada las señales que por medio de impulsos electroquímicos generan los movimientos de la mano, a través de la red nerviosa y los músculos. (Loaiza y Arzola, 2011)

#### 2.2.3.4. Empleo de materiales inteligentes en la prótesis

En la actualidad el término inteligente se utiliza como una definición para calificar y describir una serie de materiales que presentan la capacidad de cambiar sus propiedades físicas (rigidez, viscosidad, forma, color, etc.) en presencia de un estímulo concreto. Para controlar dicha respuesta de una forma predeterminada, se diseñan mecanismos de control y



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

selección. El tiempo de respuesta es corto y el sistema comienza a regresar a su estado inicial tan pronto como el estímulo cesa. En la figura 12 se presentan un grupo de materiales inteligentes empleados en el desarrollo de prótesis de mano. (Loaiza & Arzola, 2011)

| | |
|---|---|
| **Materiales con memoria de forma** | Aleaciones con memoria de forma: NiTi, CuAlTi, NiTiCu [2, 51] |
| | Polímeros con memoria de forma: SMP, Veriflex [2, 52] |
| | Cerámicos con memoria de forma [2, 53] |
| | Aleaciones con memoria de forma ferromagnética [2, 51, 53] |
| **Materiales electro y magneto activos** | Materiales piezoeléctricos [2, 53, 54] |
| | Materiales electro y magneto restrictivos [2, 53] |
| **Materiales foto y cromo activos** | Electroluminiscentes [2, 54] |
| | Fluorescentes [2, 54] |
| | Foto cromáticos [2, 54] |
| | Termo cromáticos [2, 54] |
| | Electro cromáticos [2, 54] |

***Figura 12.*** *Materiales utilizados para el desarrollo de una prótesis*
***Nota,*** *Loaiza y Arzola, (2011)*

### *2.2.4. Tipos de fabricación de prótesis:*

Algunas formas de realizar movimientos en las prótesis son de manera mecánica y electrónica, las primeras son comandadas utilizando un arnés que está sujeto alrededor de los hombros parte del pecho y brazo normalmente son utilizados para el agarre de objetos grandes y redondos debido a la poca precisión del mecanismo. Por otro lado, los electrónicos hacen uso de motores eléctricos en los dedos, muñeca y codo, además de incluir una batería recargable. (Gonzales et al, 2018)

Existen 2 clasificaciones de prótesis de manera general, las cuales son:

- **Estéticas**: Con un valor únicamente estético realizando movimientos simples y posicionales.
- **Funcionales**: Con valor funcional y estético realizan funciones vitales del miembro como abrir y cerrar la mano, rotación de la muñeca y codo, etc. (Gonzales et al, 2018)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### 2.2.4.1. Metodología para la creación de la prótesis:

El trabajo de Gonzales et al. realizaron una investigación que involucra 6 etapas, que van desde el diseño e implementación del equipo para electromiografía hasta la impresión de la prótesis. (Gonzales Badillo, Germánico; Torres Urestí, David; Espinoza Lopez, Victor; Guerreo Mora, Guillermina, 2018)

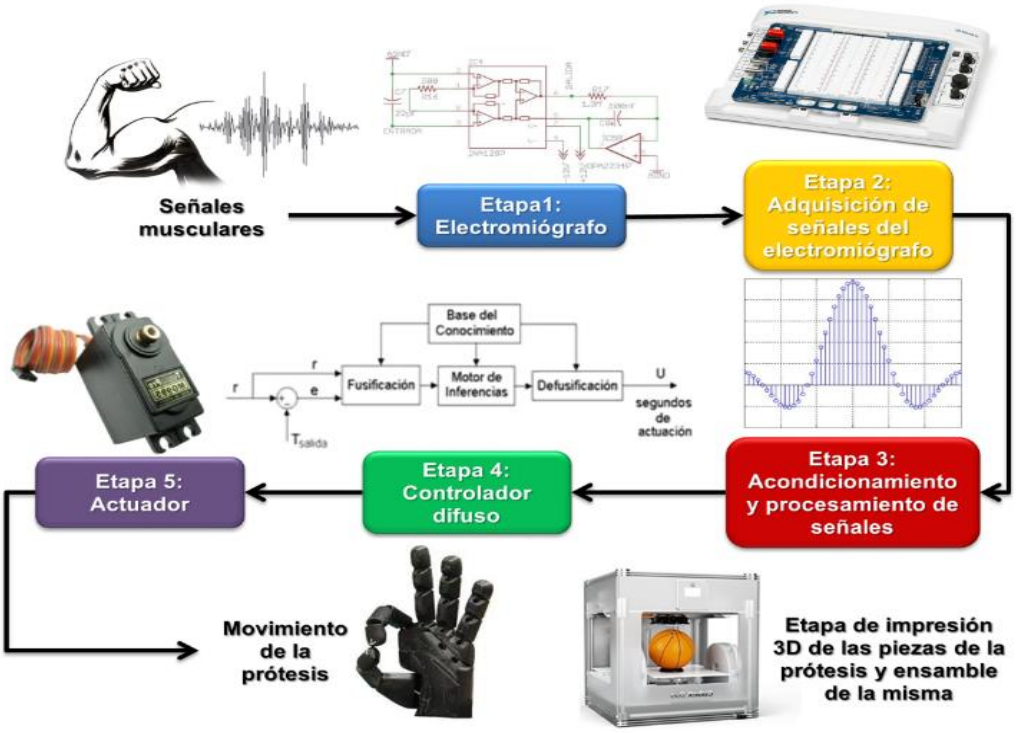



***Figura 13.*** *Etapas para la creación de la prótesis*
***Nota,*** *Gonzales et al, (2018)*

### 2.2.4.2. Desarrollo para la obtención de señales electromiográficas:

Con el fin de leer los impulsos eléctricos provenientes del músculo es necesario un equipo de electromiografía que permita monitorear la señal proveniente del músculo y transformarla en una señal eléctrica de amplitud suficiente, con la menor cantidad de ruidos y perturbaciones, para así controlar de manera efectiva los movimientos que deberá tener la prótesis. Para los fines de este trabajo se proponen dos versiones del electromiógrafo, la primera con 5 etapas de procesamiento y la segunda de 7 etapas. Esto con el fin de observar



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

cuál de las dos versiones ofrece mejores resultados para la implementación en la prótesis de mano.

Electromiógrafo con cinco etapas de procesamiento:

Se organizan por etapas para el diseño y construcción del equipo, entre los más críticos se encuentran los siguientes:

- Brindar una buena referencia a las señales que van a ser adquiridas.
- Adquisición de las señales por medio de un amplificador de instrumentación.
- Filtrado de la señal para eliminar la información innecesaria.
- Amplificación de la señal, con una ganancia con un valor grande para que pueda ser visible de manera más fácil por los instrumentos de medición.

Electromiógrafo con siete etapas de procesamiento:

En base a las observaciones hechas en el primer electromiógrafo se diseñó una segunda versión con 7 etapas de acondicionamiento de la señal, las cuales se describen a continuación:

- La primera etapa busca que se asegure el punto que va a funcionar como referencia de las señales que van a adquirir. Esta fase permite lograr el acoplamiento de aislamiento e impedancia del electrodo de referencia.
- La segunda etapa se encarga de adquirir las señales EMG (electromiografía) mediante un amplificador de instrumentación AD620, la ganancia en esta fase se obtiene a través de una resistencia de ganancia (Rg) de 470Ω debido a que las resistencias internas R1 y R2 de este amplificador tienen un valor por defecto de fábrica de 24.7kΩ; proporcionando una ganancia de 106.1.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- En la tercera etapa se emplea bajo una configuración inversora un amplificador operacional, cuya ganancia es de 15, con la intencionalidad de que la señal se aprecie en los dispositivos de instrumentación.

- La cuarta etapa se encarga de filtrar la señal con el objetivo de la información innecesaria sea eliminada, mediante un filtro pasa-bajos activos que se encarga de eliminar la información mencionada y de esa manera quede los más limpia posible la señal, la frecuencia de corte que se configura es de $fc$= 106.1Hz.

- La quinta etapa rectifica la señal con la intencionalidad de mantenerla dentro del rango de los valores positivos, mediante el uso de un rectificador de onda completa activo el cual se encarga de que sean positivos las señales de los ejes este producto de que mayormente solo se pueden realizar lectura de voltajes positivos los microcontroladores.

- La sexta etapa se encarga de que la señal se pueda apreciar mejorar mediante una suavización. Esta fase hace uso de filtros pasa altos activos con frecuencia de corte $fc$ = 1.983 Hz, con el propósito de que las señales de bajas frecuencias sean obtenidas con la finalidad de que las señales EMG se aprecien mejor y lograr con un microcontrolador procesarlas.

- La séptima etapa se encarga de amplificar con ganancia negativa y de esa manera logrará que la señal quede en el eje de valores positivos. La amplificación realizada en la segunda fase usa un amplificador operacional que se configura con ganancia de 10, con la finalidad de que la señal se pueda procesar mejor. (Gonzales Badillo, Germánico; Torres Urestí, David; Espinoza Lopez, Victor; Guerreo Mora, Guillermina, 2018)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

**2.2.4.3. Acondicionamiento y procesamiento de señales por medio de herramientas de National Instruments**

- Amplificación digital con una ganancia de -50
- Rectificación
- Filtrado
- Detector de picos
- Visualización

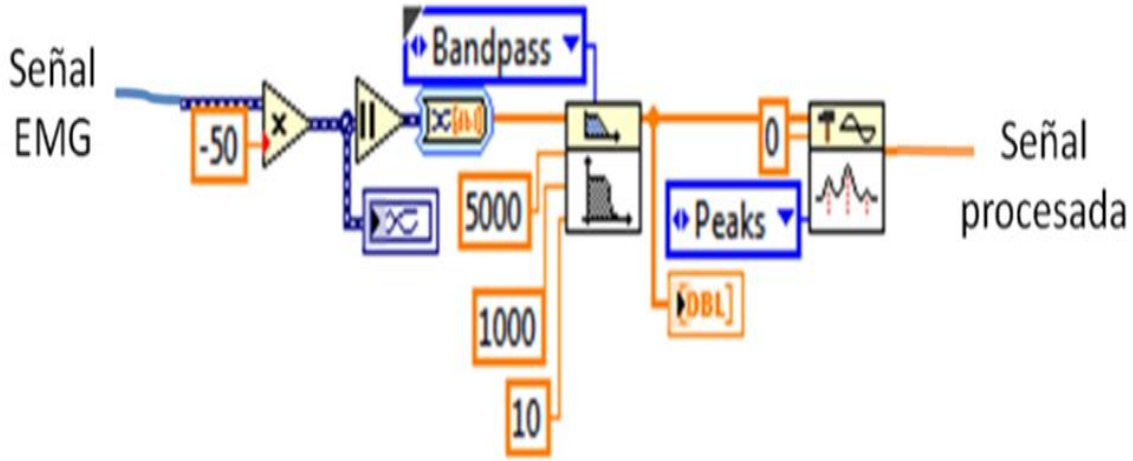



***Figura 14.*** *Procesamiento de la señal*
***Nota,*** *Gonzales et al, (2018)*

### *2.2.5. Lógica difusa*

Recientemente, la cantidad y variedad de aplicaciones de lógica difusa han crecido considerablemente. La lógica difusa es una lógica alternativa a la lógica convencional que pretende introducir un grado de vaguedad en las cosas que evalúa. En el mundo en que vivimos existe mucho conocimiento ambiguo e impreciso por naturaleza. El razonamiento humano con frecuencia actúa con este tipo de información. La lógica difusa fue diseñada precisamente para imitar el comportamiento del ser humano. (Ramos, 2008)

La lógica difusa en comparación con la lógica convencional permite trabajar con información que no es exacta para poder definir evaluaciones convencionales, contrario con la lógica tradicional que permite trabajar con información definida y precisa. (Ramos, 2008)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Es el conjunto de principios matemáticos basados en grados de pertenencia, cuya función es modelar información basándose en reglas lingüísticas, su función se centra en aproximar la entrada con la salida del sistema; mostrando rangos de membresías entre un intervalo de 0 y 1, a diferencia de la lógica convencional, se tomó el valor de 0 a 1. El control de la lógica difusa tiene mejor respuesta en comparación con los controladores lineales cuando se presentan variaciones en los parámetros, perturbaciones, por lo que se considera que es más robusto, implicando por medio de este que el controlador tendrá la capacidad de asegurar el funcionamiento sin importar las incertidumbres que existan. (Ramos, 2008)

#### 2.2.5.1. Algunas aplicaciones de la lógica difusa

Actualmente la lógica, difusa tiene un sin número de aplicaciones que afectan nuestra vida cotidiana de alguna u otra manera, pero en ocasiones no nos percatamos. La lógica difusa se ha desarrollado en diferentes áreas y a continuación se mencionan algunas:

- Control de sistemas, control de tráfico, control de vehículo, control de compuertas en plantas hidroeléctricas, etc.
- Predicción de terremotos, optimización de horarios.
- Reconocimientos de patrones y visión por ordenador: Seguimiento de objetos con cámara, reconocimiento de escritura manuscrita, reconocimiento de objetos, compensación de vibraciones en la cámara, sistemas de enfoque automático.
- Sistema de información o conocimiento: base de datos, sistemas expertos. (Ramos, 2008)

#### 2.2.5.2. Aplicaciones de la lógica difusa para un sistema robótico biomédico

En las prótesis de mano es necesario que tenga un control robusto para que así pueda discernir eficazmente cuando se va a realizar algún movimiento de la mano, evitando así los movimientos involuntarios, entre otros de los beneficios de este control es que no se precisa



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

conocer el modelo matemático del sistema real, ya que este puede ser tomado como una caja negra, en la que se introducen entradas y por medio del sistema de planta es generada la salida deseada. (Ramos, 2008)

### *2.2.6. Redes neuronales*

La perspectiva sub-simbólica trata de estudiar los mecanismos físicos que nos capacitan como seres inteligentes, frente a cómputos clásicos que simplemente son autómatas a cumplir órdenes concretas cuya función es muy contraria a la del sistema nervioso central. A través del diseño bottom – up, se recoge información de las características físicas que se desea imitar por medio de mecanismos prefijados de aprendizaje, logrando máquinas con elementos neuronales de procesamiento paralelo, donde la red emule, sea la forma más fiel del sistema neuronal animal.

La red de neuronas artificiales son nodos constituidos por unidades de procesamiento idénticas capaces de propagar información a través de los arcos, distinguiéndose por una entrada una salida y los intermedios. Si la función de activación es lineal, el introducir más capas en la red resulta irrelevante; es decir, existirá siempre una red con una sola capa equivalente frente a otra con un número de capas arbitrarias, siendo su cálculo trivial. Al incluir funciones de activación lineales, hará que la red pierda gran parte de sus capacidades.

El aprendizaje es la parte más importante de una red de neuronas basada en ejemplos; la capacidad de una red para resolver un problema estará ligada fundamentalmente a los ejemplos que dispone el proceso de aprendizaje. Se caracteriza por ser significativo; es decir, tener un número suficiente de ejemplos. A la vez representativo, se refiere a ser diverso. Consiste en la determinación de valores precisos de pesos para las conexiones que capacite.

La opinión de convergencia depende del tipo de red utilizado o problema a resolver mediante un número fijo de ciclos, cuando el error descienda por debajo de una cantidad preestablecida definiendo la función de error, la modificación de los pesos será irrelevante.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Existen tres tipos de aprendizaje:

- Supervisados, los datos propiamente dichos y cierta información relativa a la solución del problema.
- No supervisados, solo tienen información de los ejemplos y nada permite guiar en el proceso de aprendizaje.
- Por refuerzo, se refiere a información concreta del error cometido.

Las redes neuronales artificiales no son programadas, aprenden a partir de ejemplos, normalmente presentamos una serie de patrones, ejemplo de los cuales ella debe aprender, estos vienen representados por vectores y pueden ser obtenidos por fuentes de imágenes, sensores, movimientos del brazo de un robot.

Según la historia, Frank Rosenblatt generalizó el modelo de células llamando a este modelo el Perceptrón. Dos años más tarde, Bernad Widrow diseñó una red neuronal similar llamada Adaline. En la década de los 70 y 80 fueron reconocidas las contribuciones del algoritmo de Retro-propagación de Terence Sejnowski.

Los primeros modelos computacionales de neurona fueron propuestos en el año 1943 por Warren McCulloch y Walter Pitts; sin embargo, los primeros estudios sobre redes de Neuronas Artificiales datan de los años 50 con el modelo Perceptrón capaz de realizar tareas de clasificación de forma automática aun así el problema puede complicarse si en vez de dos dimensiones hay mucha más, en este caso no habrá que determinar rectas sino hiperplanos.

Adaline, en la capa de salida las células codifican una función de salida de escalón, función 2.2; al ser binarias solo pueden codificar un conjunto discreto de estados. Los datos de entrenamiento en este caso son conjuntos de valores enteros compuestos por un vector de entrada y su salida asociada:



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

$$P= \{(\overrightarrow{x_1}, y_1), \dots, \overrightarrow{x_m}, y_m)\} \qquad (1)$$

En 1960 diseñaron Adaptive Linear Neuron o en su acrónimo Adaline prácticamente idéntico a Perceptrón, el cual es un mecanismo físico combinador adaptativo capaz de realizar aprendizaje:

$$\bar{y} = \sum_{i=1}^{n} w_i \cdot x_i + \theta \qquad (2)$$

Van presentando los patrones uno a uno modificando parámetros de la red mediante regla de descenso del gradiente. La idea es realizar un cambio en cada peso proporcional a la derivada del error.

$$\Delta_p \omega_j = -y \frac{\partial E^P}{\partial w_j} \qquad (3)$$

Aplicando la regla de la cadena a la expresión anterior queda como sigue:

$$\frac{\partial E^P}{\partial w_j} = \frac{\partial E^P}{\partial y^p} \frac{\partial y}{\partial w_j} \qquad (4)$$

Al ser unidades lineales, sin función de activación en la capa de salida, se cumple:

$$\frac{\partial y^p}{\partial w_j} = x_j \qquad \frac{\partial E^p}{\partial y^p} = -(d^p - y^p) \qquad (5)$$

Que sustituyendo queda como sigue:

$$\Delta_P w_j = \gamma (d^P - y^p) x_j \qquad (6)$$

Se aprecia que la diferencia es precisamente la introducción de la diferencia entre la salida deseada y la obtenida en la regla de aprendizaje.

Así pues, la regla Delta es una extensión de la regla de Perceptrón. El procedimiento de aprendizaje por la regla de Delta será una tasa de aprendizaje γ. (Munt, 2018)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### 2.2.6.1. Perceptrón simple o monocapa

El Perceptrón simple fue introducido en 1958 por Frank Rosenblantt. Se trata del modelo más sencillo de redes neuronales artificiales, ya que consta de una sola capa de neuronas con una única salida *y*. (Munt, 2018)

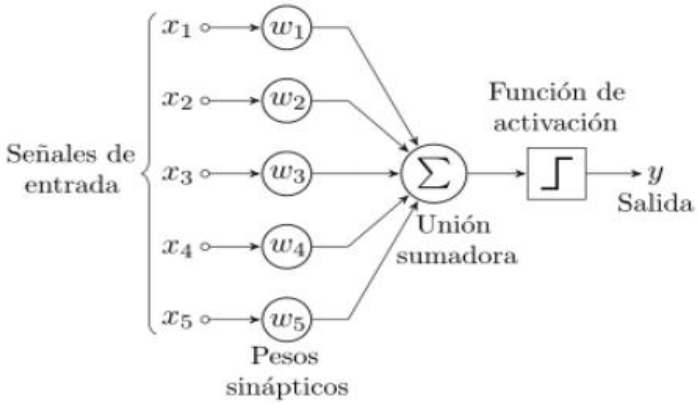



***Figura 15.*** *Ejemplo de Perceptrón simple con n = 5 entradas.*
**Nota,** *Munt, (2018)*

#### 2.2.6.1.1. Regla de aprendizaje

Para la determinación de los pesos sinápticos ***y*** del umbral vamos a seguir un proceso adaptativo que consiste en comenzar con unos valores iniciales aleatorios e ir modificándolos iterativamente cuando la salida de la unidad ***y*** no coincida con la salida deseada ***z***. Vamos a tener en cuenta que el umbral se puede considerar como el peso correspondiente a un nuevo sensor de entrada que tiene siempre una entrada igual a $x_{n+1} = -1$. (Munt, 2018)

La regla que vamos a seguir para modificar los pesos sinápticos se conoce con el nombre de regla de aprendizaje del Perceptrón simple y viene dada por la expresión:

- **Ecuación 1** $\begin{cases} (w_j\ (k+1) = w_j\ (k) + \Delta w_j(k), k = 1,2,\dots \\ siendo\ \Delta w_j\ (k) = \eta(k)[\ z(k) - y(k)]\ x_j\ (k) \end{cases}$ (7)

Es decir,

- **Ecuación 2** $w_j(k+1) = w_j(k) + \eta(k)[z(k) - y(k)]x_j(k)$ (8)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Nos indica que la variación del peso $w_j$es proporcional al producto del error *z(k) − y(k)* por la componente j-ésima del patrón de entrada que hemos introducido en la iteración *k*, es decir, $x_j(k)$. La constante de proporcionalidad η(k) es un parámetro positivo que recibe el nombre de tasa de aprendizaje, ya que cuanto mayor es, más se modifica el peso sináptico y viceversa. Es decir, es el parámetro que controla el proceso de aprendizaje. Cuando es muy pequeño la red aprende poco a poco. Si se toma constante en todas las iteraciones, η(k) = η > 0 tendremos la regla de aprendizaje con incremento fijo. Cuando usamos la función signo como función de transferencia, la regla de aprendizaje se puede escribir de la siguiente forma:

- **Ecuación 3** $$w_j(k+1) = \begin{cases} w_j(k) + 2\eta(k)x_j(k) & si\ y(k) = -1 \quad y \quad z(k) = 1 \\ w_j(k) & si\ y(k) = z(k) \\ w_j(k) - 2\eta(k)x_j(k) & si\ y(k) = 1 \quad y \quad z(k) = -1 \end{cases} \quad (9)$$

Esta regla de aprendizaje es un método de detección del error y corrección. Solo aprende, es decir, modifica los pesos, cuando se equivoca. Cuando tenemos un patrón que pertenece a la primera clase *(z(k) = 1)* y no es asignado a la misma *(y(k) = −1)*, entonces corrige el valor sináptico añadiéndole una cantidad proporcional al valor de la entrada, es decir, lo refuerza, mientras que si el patrón de entrada no pertenece a esta clase *(z(k) = −1)* y el perceptrón lo asigna a ella *(y(k) = 1)*, lo que hace es debilitar el peso restándole una cantidad proporcional al patrón de entrada. No modificaremos los pesos cuando el valor deseado coincida con la salida de la red *(z(k) = y(k))*. (Munt, 2018)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

#### 2.2.6.1.2. Algoritmo de aprendizaje

**Paso 0: Inicialización.**

Inicializar los pesos sinápticos con números aleatorios del intervalo [-1,1].

Ir al paso 1 con k = 1.

**Paso 1: (k-ésima iteración).**

Calcular $y(k) = sgn\left(\sum_{j=1}^{n+1} w_j x_j(k)\right)$ (10)

**Paso 2: Corrección de los pesos sinápticos.**

Si *z(k) 6= y(k)* modificar los pesos sinápticos según la expresión:

$w_j(k+1) = w_j(k) + \eta(k)[z(k) - y(k)]x_j(k), \quad j = 1, \dots, n, n+1$ (11)

**Paso 3: Parada.**

Si no se han modificado los pesos en las últimas *p* iteraciones, es decir,

$w_j(r) = w_j(k), \quad j = 1, \dots, n+1, \quad r = k+1, \dots, k+p$ (12)

parar. La red se ha estabilizado. En caso contrario, ir al paso 1 con k = k +1.

### 2.2.6.2. Perceptrón Multicapa

La red perceptrón multicapa o *Multi- Layer Perceptrón MLP*, está compuesta por un arreglo de neuronas de tipo perceptrón ordenados por capas (Rosenblatt, 1958). Una MLP presenta al menos tres capas de neuronas: capa de entrada, oculta y de salida. Así mismo, esta red tiene al menos dos neuronas en cada capa oculta y una neurona en las capas de entrada y salida. El perceptrón multicapa, tiene un amplio rango de aplicación en multitud de problemas de diferentes áreas del conocimiento, y es considerado una de las arquitecturas de redes neuronales más versátiles (Nunes da Silva, 2017). Algunas de sus principales aplicaciones son: aproximador de funciones universales, reconocimiento de patrones, control y procesos de identificación. En la figura 16 se muestra la representación de una red MLP, la cual cuenta con entradas (*x1, x2..., $x_n$*) en la capa de entrada ($C_{en}$); una capa oculta ($C_{oc}$), con $n_1$ neuronas, así como una capa de salida ($C_{sal}$) con $y_m$ neuronas.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

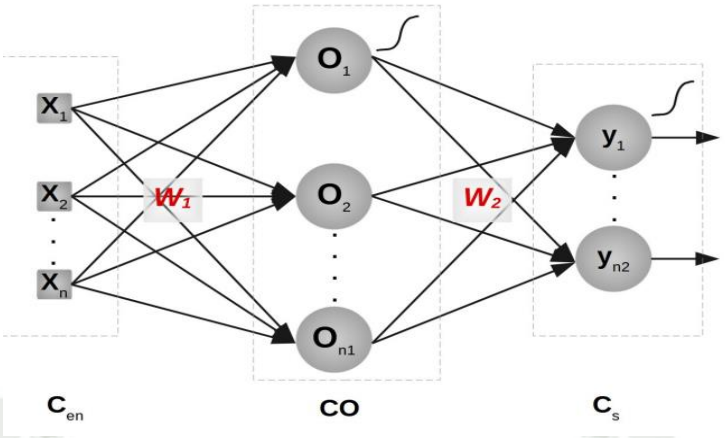



***Figura 16.*** *Arquitectura de la red perceptrón multicapa*

Para enlazar las neuronas de $C_{en}$ con las de $C_{oc}$, se emplea la matriz $w_1$ que representa la sinapsis; por otro lado, la matriz $w_2$ representa las sinapsis que transporta la información de cada neurona en $C_{oc}$ hacia las neuronas de $C_{sal}$. A cada componente de una matriz *w*, se le denomina peso sináptico, mismo que se multiplica por la señal de entrada de la sinapsis, la cual puede provenir de una neurona en $C_{en}$ o de $C_{oc}$, según sea el caso. (Rosenblatt, 1958)

**2.2.6.2.1. Proceso de aprendizaje**

El proceso de aprendizaje para la red MLP se denomina *backpropagation*, el cual se puede dividir en tres partes: la primera parte es la propagación hacia adelante (*forward propagation*), que consiste en generar la salida de la red a través de la activación de sus neuronas; la segunda es la retro-propagación (*backpropagation*), en donde se lleva a cabo el ajuste de los pesos sinápticos de las matrices *w*, en base al error cometido por la red; y por último, se lleva a cabo el proceso de evaluación, que trata de generar el error global de la red, y si dicho error sobrepasa un nivel de umbral de precisión definido por ε, se detiene el proceso de aprendizaje. A continuación, se describen cada una de estas etapas que conforman el algoritmo *backpropagation*. (Rumelhart, 1989)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

#### 2.2.6.2.2. Programación hacia adelante

La propagación hacia adelante consiste en producir las salidas de cada neurona mediante la ecuación 4, y posteriormente acotar dicha salida mediante una función de activación, como las que se presentan en la ecuación 5. Este proceso comienza con las neuronas de la primera capa oculta, hasta llegar a las neuronas de la capa de salida.

- **Ecuación 4** $u = \sum_{i=1}^{n} (w_n * x_n) - \theta$ (13)

- **Ecuación 5** $G(u) = \tanh(u)$ $o\ bien\ G(u) = sigmoide\ (u)$ (14)

Donde *u* representa el potencial de activación de una neurona; *n* indica el número de entradas (sinapsis de la neurona), que para el caso de las neuronas de $C_{oc}$, representa la cantidad de características ($x_n$) dado por un vector de entrada *x*, mientras que para las neuronas de $C_{sal}$, corresponderá al número de neuronas en la capa $C_{oc}$; y G($u$) representa la salida o activación de dicha neurona, que está dada por una función de activación, la cual puede ser sigmoidal o tangente inversa, esto se hace para acotar el valor de salida entre 0 y 1 o −1 y 1 respectivamente. Una vez generada la salida de la red se concluye el proceso de propagación hacia adelante, y se da paso a la segunda etapa, que consiste en la retro-propagación del error, para corregir el valor de los pesos sinápticos *w*. (Rumelhart, 1989)

#### 2.2.6.2.3. Retro-propagación

Esta etapa consiste en modificar el valor de los pesos sinápticos, comenzando con los que se encargan de enlazar las neuronas de la capa $C_{oc}$ con las neuronas de la capa $C_{sal}$ definido por la matriz $w_2$, y posteriormente se continúa con los de la matriz $w_1$. Inicialmente se obtiene el error de la red dada una muestra *x*, mediante la diferencia entre la salida deseada *d* y la producida por la red *y* (ecuación 6).

- **Ecuación 6** $e = d - y$ (15)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Obteniendo la medida del error (*e*), se procede a calcular $\delta_{Csal}$, que corresponde al error producido por la red dado un cambio generado en el potencial *u* de una neurona en la capa $C_{sal}$, con un conjunto de pesos sinápticos $w_2$ (ecuación 7).

- **Ecuación 7** $\delta C_{sal} = \frac{\partial y}{\partial u} * e_i$ (16)

En seguida, se lleva a cabo el ajuste de los pesos sinápticos $w_2$, tal como se indica en la ecuación 8.

- **Ecuación 8** $w_2 = w_2 + (\lambda * \delta C_{sal} * y_i)$ (17)

Donde λ, es el factor de aprendizaje, el cual puede tener un valor entre 0 y 1. Por último, realiza el ajuste del umbral $\theta_{Csal}$ correspondiente a las neuronas de salida y, el cual se calcula mediante la ecuación 9.

- **Ecuación 9** $\theta C_{sal} = \theta C_{sal} + (\lambda * \delta C_{sal})$ (18)

Ahora se continúa con el ajuste de los pesos sinápticos de la matriz $w_1$. Para esto se debe obtener $\delta_{Coc}$, que corresponde al cambio producido en el potencial *u* de las neuronas de la capa $C_{oc}$ ante una entrada *x* con pesos sinápticos $w_1$, y para transportar la medida del error cometido por la red (*e*), se hace uso del valor $\delta C_{sal}$, ecuación 10.

- **Ecuación 10** $\delta C_{oc} = \frac{\partial G(u)}{\partial u} * w_2 * \delta C_{sal}$ (19)

El ajuste de $w_1$ está definido por la ecuación 11.

- **Ecuación 11** $w^{(1)} = w^{(1)} + (\lambda * \delta C_{oc} * x)$ (20)

Por último, se ajusta el umbral ($\theta_{Coc}$), utilizando la ecuación 12.

- **Ecuación 12** $\theta C_{oc} = \theta C_{oc} + (\lambda * \delta C_{oc})$ (21)

Una vez terminado, el ajuste de los pesos *w* y $\theta$ respectivos a las capas $C_{sal}$ y $C_{oc}$, se vuelve a llevar a cabo el proceso de propagación hacia adelante, pero ahora considerando los nuevos valores ajustados, con lo cual se espera obtener una salida *y*, que se aproxima de



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

mejor manera que en un tiempo anterior. En consecuencia, se producirá un error diferente al obtenido previamente, el cual se utilizará para generar el error global de la red *E*, a continuación, se presenta este proceso. (Rumelhart, 1989)

#### 2.2.6.2.4. Evaluación

Comienza cuantificando el error cuadrático medio (*e* (*k*)) cometido por la red dado una entrada *x*, correspondiente a una muestra *i* del conjunto de muestras X ecuación 13.

- **Ecuación 13** $\boldsymbol{e(k)_i = \frac{1}{2}\,(d_i - y_i)^2}$ (22)

El resultado de *e (k)* se almacena para cada muestra *i* dentro del conjunto *X*, y al finalizar se hace el promedio de todos los errores cometidos para generar la medida del error global de la red *E* (ecuación 14).

- **Ecuación 14** $E = \frac{1}{p}\sum_{k=1}^{p} e(k)$ (23)

Donde *p* corresponde al total muestras *x* dentro del conjunto *X*. Por último, se compara el error global *E* con el valor de error permitido (ε), y si *E* es menor que ε, entonces se detiene el proceso de aprendizaje, concluyendo que la red ha sido entrenada. Además, se suele utilizar el parámetro *e pocas*, para detener el proceso de aprendizaje, dado un número de iteraciones máximo, en caso de que la red no logre alcanzar la precisión ε. (Rumelhart, 1989)

### 2.2.6.3. Red neuronal Base Radial

Las funciones de base radial fueron primeramente introducidas en la solución de problemas de interpolación multivariante. Los primeros trabajos se atribuyen a Powell (1987) y ahora es uno de los principales campos en la investigación del análisis numérico.

Bromhead y Lowe (1988) fueron los primeros que explotaron el uso de las funciones de base radial en el diseño de redes neuronales. Otras contribuciones importantes a la teoría, diseño y aplicación de las redes de función de base radial son los artículos de Moody y Darken



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

(1989), Renals (1989) y Poggio y Girosi (1990). El artículo de Poggio y Girosi enfatizaba el uso de la teoría de la regularización aplicada a esta clase de red neuronal como un método para mejorar la generalización de nuevos datos. (Juan Carlos Pérez Mesa, 2007)

Las redes neuronales de base radial están formadas por tres capas: La capa de entrada, una única capa oculta y la capa de salida, como se muestra en la figura. La capa de entrada la componen un conjunto de neuronas que reciben las señales del exterior, transmitiéndolas a la siguiente capa sin realizar ningún procesado sobre dichas señales. Las neuronas de la capa delta reciben de las señales de la capa de entrada y realizan una transformación local y no lineal sobre dichas señales. Este carácter local es lo que las diferencian del perceptrón multicapa, no solo en cuanto a arquitectura, sino también en cuanto a comportamiento. Esta capa es la única que incluye componentes no lineales en redes de base radial. Y, finalmente, la capa de salida que realiza una combinación lineal de las activaciones de las neuronas ocultas, que actúa además como salida de la red.

Las redes de neuronas de base radial son redes con conexiones hacia adelante, como se observa en la figura, estas conexiones se dirigen siempre de una capa a la siguiente capa. La red se caracteriza porque las conexiones de la capa de entrada a la capa oculta no llevan asociado ningún peso, mientras que, y como es habitual en el contexto de redes de neuronas, las conexiones de la capa oculta en la capa de salida sí llevan asociado un número real o peso de la conexión. En lo referente a los umbrales de las neuronas, en las redes de base radial únicamente las neuronas de salida poseen un umbral, que también se suele tratar al igual que en el perceptrón multicapa, como una conexión más de la neurona cuya entrada es constante e igual a 1.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

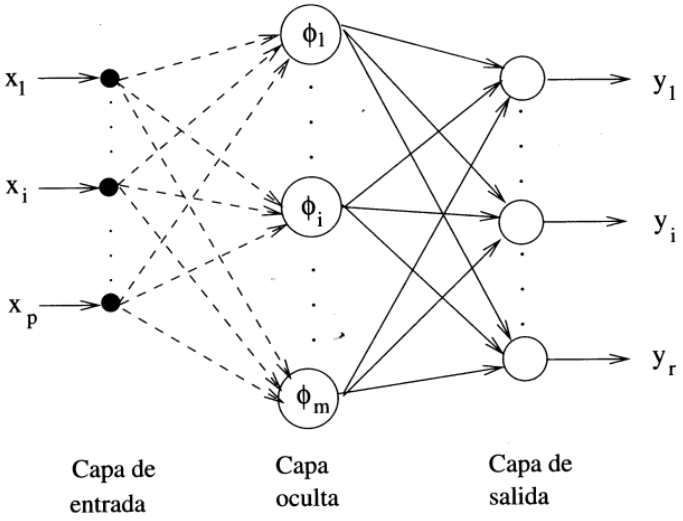



***Figura 17.*** *Arquitectura de la red de neuronas de base radial*
**Nota,** *Pedro Isasi Viñuela, (2004)*

Las redes de neuronas de base radial definen una relación no lineal entre las variables de entrada y las variables de salida de la red, propagando hacia la salida la señales o muestras recibidas en la entrada. (Pedro Isasi Viñuela, 2004)

De modo general, el valor generado por una red RBF con una única salida, viene definido por la ecuación:

- **Ecuación 15** $y_i = \sum_{j=1}^{N} w_j f_j \left( \frac{\|x_j - u_j\|}{\sigma_j} \right) y_i = \sum_{j=1}^{N} w_j f_j \left( \frac{\|x_j - u_j\|}{\sigma_j} \right)$ (24)

### 2.2.6.3.1 Aprendizaje de las redes neuronales de base radial

El aprendizaje de las redes neuronales artificiales de Base Radial generalmente es de tipo híbrido, realizándose en dos etapas. Primeramente, se lleva a cabo un entrenamiento no supervisado en la capa oculta y posteriormente un entrenamiento supervisado en la capa de salida. Con esta división se consigue acelerar notablemente el proceso de aprendizaje con respecto a otros algoritmos. Este proceso de aprendizaje se realiza de forma iterativa, pues el ajuste final de los parámetros se logra a partir de aproximaciones sucesivas de los mismos. (Egly Barrero Viciedo, 2014)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### *2.2.7. Háptico*

Etimológicamente, la palabra proviene del griego *haptesthai*, que significa tocar, es un adjetivo utilizado para referirse al sentido del tacto. Este sentido recibe y genera información háptica, que contiene la información que viene del sentido del tacto para controlar los impulsos en los miembros que por alguna u otra razón ya no se encuentran (Raya, 2015)

### *2.2.8. Tecnología Háptica*

La percepción háptica hace referencia a la información que recibe el cerebro por contactos que tiene la piel con cualquier objeto, se conoce como uno de los sentidos más importante con los que cuenta el ser humano, el tacto. El sistema háptico pone en contacto directo al individuo con el mundo que lo rodea, pudiendo a través de este sistema palpar superficies, tamaños, formas, texturas, cualidades de consistencia como, duro, suave, áspero, frío, caliente, tibio, todo dependiendo de la información recibida por el cuerpo, algunas veces diferenciando la estática con la dinámica, determinando y dependiendo del tipo de movimiento que esté ejerciendo el individuo lineal o aleatorio (Mainieri y Méndez, 2005).

#### 2.2.8.1. Información propioceptiva

Es un sentido de intercepción del cuerpo mediante el cual se informa al organismo sobre la posición exacta de todas las partes del cuerpo en cada momento, este sentido se da por la integración de impulsos somato sensoriales, conscientes e inconscientes, que se transmiten a través de receptores ubicados en estructuras de todo el cuerpo, controlando neuromuscularmente las acciones motrices del sujeto (Zarza, 2014).

#### 2.2.8.2. Información táctil

Permite la interacción con las terminaciones nerviosas de la piel, que indican textura, calor y presión. A través del sentido del tacto, que se encuentra principalmente en la piel, la cual posee distintos tipos de receptores nerviosos, que transforman la información cruda,



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

recibida del exterior por medio de estímulos, en información susceptible para que el cerebro la interprete (Raya, 2015).

#### 2.2.8.3. Información vestibular

Generada por el sistema vestibular, que está formado por el oído interno y el cerebro, los cuales procesan la información sensorial sobre el control de equilibrio y el movimiento ocular (Raya, 2015).

### *2.2.9. La organización psicomotriz*

"Se encuentra vinculada con el funcionamiento del sistema háptico, que está vinculado al sentido del tacto y las informaciones provenientes de los músculos y articulaciones" (Pierre y Méndez, 2010, p. 106). El sistema háptico es muy importante porque de su existencia percibimos información del medio ambiente tales como el tacto de:

- Información geométrica sobre áreas o superficies.
- Textura de las superficies.
- Consistencia de las cosas que tocamos.
- Las temperaturas.
- Presión.
- Los patrones dinámicos de movimiento del tronco, brazos, piernas, mandíbulas y lenguas.
- Si el cuerpo está estático o en movimiento.
- Sensibilidades a movimientos.
- Localización de objetos en relación con el cuerpo (Pierre y Méndez, 2010, p. 106).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### *2.2.10. Materiales de la prótesis*

- **Amplificador de instrumentación AD620**: Es un amplificador de instrumentación de alta precisión y bajo costo que requiere solo una resistencia externa para establecer ganancias de 1 a 1000. Además, el AD620 cuenta con encapsulados SOIC y DIP de 8 pines encapsulados que son más pequeños que los diseños discretos y que ofrecen menos potencia (solo corriente de suministro máxima de 1.3 [$mA$]), lo que lo hace muy útil para aplicaciones con baterías, portátiles (o remotas).

***Tabla 2*** *Características de amplificador de instrumentación AD620.*

| **Características principales** | |
|---|---|
| Ganancia de Voltaje | De 1 a1000 |
| Se requiere una sola resistencia para configurar la ganancia | |
| Amplio rango de voltaje de alimentación | ± 2,3[V] a ± 18[v] |
| Bajo consumo | 1,3[mA] máx. |
| Excelente desempeño en DC | 50[$\mu V$] offset máx. |
| Desvío máximo | de 0,6[$\mu V/°C$ |
| Ancho de banda | 120[kHz] con G=100 |
| Bajo ruido | |
| Encapsulado | DIP 8 |

**Nota,** *Santin, (2020)*

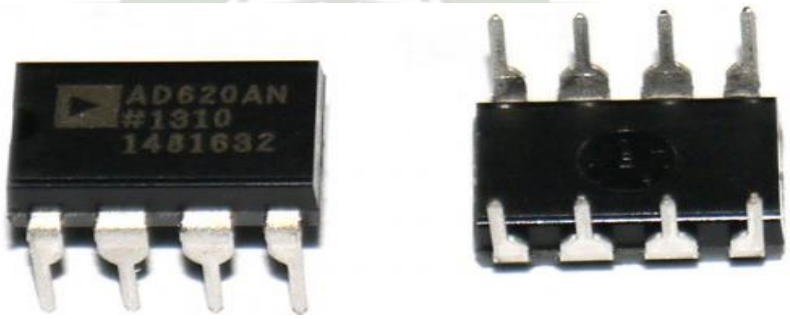



***Figura 18.*** *Amplificador de instrumentación AD620*
***Nota,*** *Electronilab, (s.f.)*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

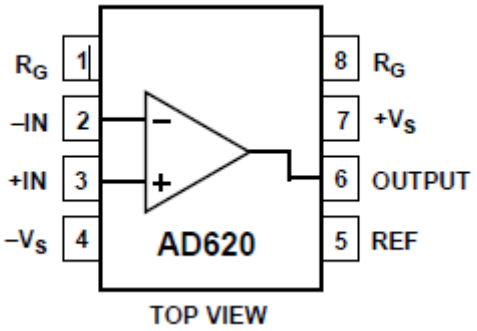



***Figura 19.*** *Paquetes de plomo*
**Nota,** *Santin, (2020)*

- **Amplificador TL084:** posee cuatro amplificadores internos con un ancho de banda de 3 MHz, alta velocidad, entradas de alta impedancia JFET y bajo consumo de potencia. El modelo TL084, el cual dispone de cuatro amplificadores operacionales internos con un ancho de banda de 3MHz (U6A, U6B, U6C y U6D), alta velocidad, entradas de alta impedancia JFET y bajo consumo de potencia. (Palacio Velasquez, 2013)**.** Según Orts (2019) para la etapa de amplificación utilizamos uno de los amplificadores operacionales que nos proporciona el encapsulado TL084. El siguiente proceso consiste en amplificar la señal obtenida después de atenuar las excitaciones de baja frecuencia, con lo que conseguimos reducir las posibilidades de superponer interferencias no deseadas, mayormente ruidos, a nuestra señal de trabajo. Un amplificador que cumple con los requisitos es el TL084 el cual presenta las siguientes características eléctricas:



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

***Tabla 3*** *Características amplificador TL084.*

| | |
|---|---|
| Tipo de entradas | JFET |
| Impedancia de entrada/salida | Alta |
| Admisión de ruido | Bajo |
| Niveles de voltaje de alimentación | $\pm 5Vdc\ a \pm 15Vdc$ |

**Nota,** *Bastidas, (2008)*




***Figura 20.*** *Amplificador TL084*
***Nota,*** *Electronilab, (s.f.)*

Los TL084, TL084A y TL084B son amplificadores operacionales cuádruples de entrada J-FET de alta velocidad que incorporan transistores bipolares y J-FET de alto voltaje bien combinados en un circuito integrado monolítico y coeficiente de temperatura de voltaje.

- Amplio Modo Común (Hasta Vcc+) y Rango de Voltaje Diferencial.
- Polarización de Entrada Baja y Corriente de Compensación.
    - Protección Contra Cortocircuito de Salida.
    - Impedancia de Entrada Alta Etapa De Entrada J–FET
    - Compensación de Frecuencia Interna 16V/αs (typ)

    (STMicroelectronics, 1999)

    - **Batería LiPo:** Marcada como “3,7 V/ (4,2 V)” indica que inicialmente es capaz de aportar un voltaje máximo de 4,2 V, pero rápidamente desciende a



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

3,7 V, el cual será su voltaje medio durante la mayor parte de su vida útil, hasta que finalmente baje rápidamente hasta los 3 V y automáticamente deje de funcionar. En este sentido, es útil consultar la documentación oficial ofrecida por el fabricante para cada batería particular (el llamado "datasheet" de la batería) para saber la variación del voltaje aportado en función del tiempo de funcionamiento. (Torrente, 2013)




***Figura 21.*** *Batería Lipo*
***Nota,*** *Electronilab, (s.f.)*

- **Este servomotor digital MG996R:** cuenta con engranajes de metal que dan como resultado 10 kg extra estancado en un paquete diminuto. El MG996R es esencialmente una versión mejorada del famoso servo MG995, y cuenta con una protección contra golpes mejorada y una PCB e IC rediseñados, sistema de control que lo hacen mucho más preciso que su predecesor. El engranaje y el motor, también se han actualizado para mejorar el ancho de banda muerto y el centrado. La unidad viene completa con cable de 30 cm y conector de cabecera hembra tipo 'S' de 3 pines que se adapta a la mayoría de los receptores, incluidos Futaba, JR, GWS, Cirrus, Blue Bird, Blue Arrow, Corona, Berg, Spektrum y Hitec.

  Este servo estándar de alto esfuerzo de torsión puede girar aproximadamente 120 grados (60 en cada dirección). Puede usar cualquier código de servo, hardware o biblioteca para controlar estos servos, por lo que es ideal para



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

principiantes que quieren hacer que las cosas se muevan sin construir un controlador de motor con retroalimentación y caja de cambios, especialmente porque cabe en lugares pequeños. El MG996R Metal Gear Servo también viene con una selección de brazos y herrajes para que pueda configurarlo de manera agradable y rápida.

***Tabla 4*** *Especificaciones servomotor MG996R.*

| Características servomotor digital MG996R | |
|---|---|
| Peso | 55 g |
| Dimensión | 40,7 x 19,7 x 42,9 mm aprox. |
| Par de parada | 9,4 kgf·cm (4,8 V), 11 kgf·cm (6 V) |
| Velocidad de funcionamiento | 0,17 s/60º (4,8 V), 0,14 s/60º (6 V) |
| Tensión de funcionamiento | 4,8 V a 7,2 V |
| Corriente de funcionamiento | 500 mA – |
| Corriente de parada | 2,5 A (6 V) |
| Ancho de banda muerta | 5 µs |
| Diseño de rodamiento | bolas dobles estable y a prueba de golpes |
| Rango de temperatura | 0 ºC – |
| 900mA (6V) | |
| 55ºC | |

**Nota,** Datasheets, (2014)




***Figura 22.*** *Servomotor MG996R*
***Nota,*** *Electronilab, (s.f.)*

- **TD-8120MG Digital Servo:** Un servo muy potente que es fácil de controlar. Viene con algunas piezas de plástico que puedes montar en él, a las que luego puedes montar otras cosas. Este servo puede generar corrientes relativamente altas, especialmente con una carga (hasta aproximadamente 2500 mA a 6 V en reposo). Así que elige una fuente de alimentación adecuada para el servo. No se



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

recomienda utilizar el Arduino como fuente de alimentación. Si el servo tiene un rango de menos de 180 grados, el ancho de pulso mínimo (500us) y máximo (2500us) debe ajustarse en el controlador.

***Tabla 5*** *Especificación servomotor TD8120MG.*

| | |
|---|---|
| **Talla** | 40.0*20.5*40.5mm |
| **Peso** | 56g±5% |
| **Tipo de engranaje** | 5 metal |
| **Límite de Angulo** | 360° |
| **Soporte** | 2BB |
| **Estría del engranaje de la bocina** | 25T 5.9mm |
| **Tipo de bocina** | Plastic, POM |
| **Caso** | Plásticos de Ingeniería+ aleación de aluminio |
| **Cable Conector** | 320mm±5mm |
| **Motor** | Motor de escobillas carbón |
| **Resistencia al agua** | No |

**Nota,** Tinytronics, (s.f.)




***Figura 23.*** *Servomotor TD8120MG*
**Nota,** Tinytronics, (s.f.)

### *2.2.11. ESP32*

- **El módulo ESP32**: ESP32 es la denominación de una familia de chips SoC (System on a chip / Sistema en un Chip) de bajo costo y consumo de energía, con tecnología Wi-Fi y Bluetooth. El ESP32 fue creado y desarrollado por Espressif Systems y es el sucesor de la familia ESP8266.

  Es una solución de Wi-Fi/Bluetooth todo en uno, integrada y certificada que



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

proporciona no solo la radio inalámbrica, sino también un procesador integrado con interfaces para conectarse con varios periféricos. El procesador en realidad tiene dos núcleos de procesamiento cuyas frecuencias operativas pueden controlarse independientemente entre 80 megahercios (MHz) y 240 MHz. Los periféricos del procesador facilitan la conexión a una variedad de interfaces externas como:

***Tabla 6*** *Variedad de interfaces externas de módulo ESP32.*

| **Interfaces externas** |
|---|
| Interfaz periférica serial (SPI) |
| I2C |
| Transmisor receptor asíncrono universal (UART) |
| I2S |
| Ethernet |
| Tarjetas SD |
| Interfaces táctiles y capacitivas |

**Nota,** Beningo, (2020)

A continuación, exploramos la versatilidad del ESP32 y por qué es tan popular en electrónica y programación:

**1. Potencia de procesamiento**: El ESP32 cuenta con dos núcleos Xtensa LX6 de 32 bits, lo que lo hace un microcontrolador muy potente. También tiene una velocidad de reloj de hasta 240 MHz, lo que lo convierte en una excelente opción para proyectos que requieren un alto rendimiento.

**2. Conectividad Wi-Fi y Bluetooth:** La conectividad Wi-Fi y Bluetooth integrada en el ESP32 lo hace ideal para proyectos de IoT. Puede conectarse a Internet o a otros dispositivos Bluetooth y enviar y recibir datos de forma inalámbrica.

**3. Bajo consumo de energía:** El ESP32 cuenta con varias características de ahorro de energía, como un modo de suspensión profunda y un procesador de bajo consumo de energía. Esto lo hace una excelente opción para proyectos que



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

necesitan funcionar con baterías o energía solar.

**4. Amplia gama de periféricos:** El ESP32 cuenta con una amplia variedad de periféricos integrados, como sensores, convertidores analógico-digitales, controladores de pantalla, etc. Esto hace que sea fácil conectar y controlar otros componentes en un proyecto.

**5. Fácil programación:** El ESP32 se puede programar con varios lenguajes de programación, como Python, C++, y Arduino IDE. Además, existe una gran cantidad de documentación y ejemplos en línea para ayudar a los programadores a comenzar con el ESP32.

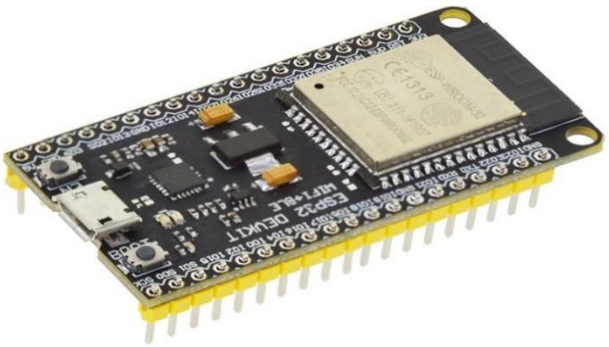


***Figura 24.*** *Módulo ESP32*
**Nota,** Beningo, (2020)

- **ESP32 38 Pines**: ESP32 de 38 Pines es una placa de desarrollo que integra el microcontrolador ESP32-WROOM-32 SMD de Espressif. Esta placa permite controlar todo tipo de sensores, módulos y actuadores mediante Wi-Fi y BLUETOOTH, para proyectos de Internet de las cosas “LoT” de forma eficiente y económica. Cuenta con micro USB de tipo B para ser alimentado y para programar el ESP32 e igual integra el controlador USB a UART CP2102.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- **Utilidad de ESP32 con 38**: El módulo posee un regulador de voltaje que permite ingresar 5V por el puerto USB, de igual manera puedes alimentarlo con 3.3v en los pines de 3.3v y gnd, “Debes de tener cuidado, ya que si ingresas el voltaje de 5V en el pin de 3.3V del ESP32 lo podrás dañar”. De igual manera cuenta con el chip CP2102 el cual se encarga de la comunicación USB-Serial.

  Puede ser programado con gran variedad de softwares, lenguajes de programación, frameworks, librerías, código/ejemplos y otros recursos. Los más comunes a elegir son: Esp-idf (Espressif IoT Development Framework) desarrollado por el fabricante del chip, Arduino IDE (en lenguaje C++), Simba Embedded Programming Platform (en lenguaje Python), RTOS’s (como Zephyr Project, Mongoose OS, NuttX RTOS), MicroPython, LUA, JavaScript (Espruino, Duktape, Mongoose JS), Basic.

  El ESP32 38 Pines está diseñado especialmente para trabajar con un protoboard y con ayuda de cables Dupont conectar de manera rápida todo tipo de sensores. **(**Beningo, 2020)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

***Tabla 7*** *Especificaciones ESP32 38 Pines.*

| Especificaciones y Características | |
|---|---|
| Tipo | Módulo Wifi + Bluetooth |
| Modelo | ESP32 38 Pines |
| Voltaje de Alimentación (USB) | 5V DC |
| Voltaje de Entradas/Salidas | 3.3V DC |
| Consumo de energía de 5µA | modo de suspensión |
| CPU principal | Tensilica Xtensa 32-bit LX6 |
| Desempeño | Hasta 600 DMIPS |
| Frecuencia de Reloj | Hasta 240Mhz |
| Procesador secundario | Permite hacer operaciones básicas en modo de ultra bajo consumo |
| Wifi | 802.11 b/g/n/e/i (802.11n @ 2.4 GHz hasta 150 Mbit/s) |
| Bluetooth | 4.2 BR/EDR BLE Modo de control dual |
| Memoria | 448 KByte ROM, 520 KByte SRAM, 6 KByte SRAM en RTC y QSPI admite múltiples chips flash /SRAM |
| Chip USB-Serial | *CP2102* |
| Antena | PCB |
| Pines Digitales GPIO | 24 (Algunos pines solo como entrada) |
| Convesor Analógico Digital | Dos ADC de 12bits tipo SAR, soporta mediciones en hasta 18 canales, algunos pines soportan un amplificador con ganancia programable |
| Seguridad | IEEE 802.11, incluyendo WFA, WPA/WPA2 y WAPI |
| Criptografía acelerada por hardware | AES, SHA-2, RSA, criptografía de curva elíptica (ECC), generador de números aleatorios (RNG) |

**Nota,** Unit Electronics, (2016)




***Figura 25.*** *ESP32 38 Pines*
**Nota,** Leantec (s.f.)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### *2.2.12. Raspberry pi.*

Básicamente, es un ordenador adaptado a necesidades de programación muy básicas, de placa reducida, bajo coste, sistema operativo abierto Raspbian. Inicialmente, fue diseñado para la enseñanza de informática en escuelas, sin embargo, resalta por sus capacidades emergentes, las cuales gracias a los usuarios son cada vez más variadas. Pudiendo usarse como un ordenador, hasta como controlador para robots, entre miles de otras aplicaciones.

Esta versión es más potente y moderna, ya que tiene un tamaño práctico, similar a una tarjeta de crédito (8.56cm x 5.65 cm) y un peso de 45 gramos. Es un ordenador de placa única (SBC), tiene las funciones básicas de una computadora, navegación por internet, vídeo, documentos, juegos, programaciones. Su funcionamiento es similar al Arduino, lo que permite usar pines GPIO y comunicación tipo I2C y SPI. Sin embargo, su practicidad viene siendo aprovechada y explorada en proyectos sobre electrónica y robótica, ya que pueden ensamblarse sensores y actuadores, que lo hacen sumamente útil en procesar imágenes, vídeos, cámaras y cálculos matemáticos complejos. Su componente principal es System on Chip (SoC) Broadcom BCM2711, que cuenta con un chip regulador de voltaje, que aumenta la eficiencia de energía, además cuenta con un procesador ARMv8 de 4 núcleos de 64 bits, con una velocidad de 1.5 GHz, GPU Broadcom Video Core VI y Memoria RAM de 4GB LPDDR4, que le permite funcionar con ARM, GNU/Linux, aunque Raspberry diseño su propio sistema Raspberry Pi OS, y funciona también con Sanppy Ubuntu Core y Microsoft Windows 10 loT. Tiene 2 puertos USB comunes y 2 puertos USB 3.0 de mayor velocidad, conexión micro-HDMI (type-D) para 2 monitores, Wifi doble banda, Bluetooth 5.0 BLE, ethernet gigabit mejorado, entre otras características que lo hacen una de las potentes del mercado actual. (Santana, 2019)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## 2.3. Conceptualización de términos básicos

a) **Amputación:** Se define como el corte quirúrgico de una extremidad o de parte de la misma parcial, en los casos en que otros procedimientos no han dado resultado o se encuentra en riesgo el paciente por el deterioro de su salud (Topdoctors, 2018).

b) **Anatomía:** Es la ciencia o rama de la biología que se encarga de estudiar todo lo referente a la organización y estructura de todos los seres vivos, sea cual sea su estado evolutivo (Munive, 2009).

c) **Articulaciones:** Es el órgano de unión entre dos o más huesos que permite el desplazamiento, su estabilidad depende de los ligamentos y se movilizan por la musculatura, la anatomía cambia entre articulaciones según la necesidad de función, movimiento y estabilidad. (García y Vázquez , 2010)

d) **Discapacidad: S**e define como la "restricción o falta (debido a una deficiencia) de la capacidad para realizar una actividad en la forma o dentro del margen que se consideran normales para un ser humano. Engloba las limitaciones funcionales o las restricciones para realizar una actividad que resultan de una deficiencia". Las discapacidades son trastornos definidos en función de cómo afectan la vida de una persona. Algunos ejemplos de discapacidades son las dificultades para ver, oír o hablar normalmente; para moverse o subir las escaleras; para bañarse, comer o ir al servicio (Padilla, 2010).

e) **Flexión:** Es el movimiento que realizan las articulaciones del cuerpo, en la cual se acercan las zonas de unas estructuras a otras (Fisioonline, 2019)

f) **Háptica:** Hace referencia a la ciencia del tacto, cuyo objetivo es el de trasladar sensaciones realistas a través de tacto, cuyos avances son un poco difíciles de alcanzar y su tecnología aún es un poco rudimentaria (Israel, 2020)

g) **Miembros superiores:** Son llamados normalmente brazos, y se encuentra unidos al tronco a nivel de la articulación del hombro, se encuentra constituidos por:



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

clavículas, omóplatos, húmeros, cúbitos, los huesos del carpo, los huesos metacarpianos y las falanges; de igual manera cuenta con tres articulaciones las cuales son el hombro, el codo y la muñeca, cabe destacar que el hueso húmero es bastante sólido lo que es más resistente a las fracturas que los del antebrazo (CCM Salud, 2016).

**h) Presión:** Es la fuerza que algo ejerce sobre otra cosa con lograr apretarlo o comprimirlo (Significados, 2021).

**i) Prótesis mio-eléctrica:** Son prótesis controladas electrónicamente a través de contracciones musculares intencionales por parte del usuario; dicha prótesis dispone de dos electrodos dentro del encaje que están en contacto con la superficie de la piel en todo momento; recogen la actividad eléctrica generada por el músculo durante la contracción y la convierten en tensión para el control de la mano protésica (Ortosur, 2018).

**j) Prótesis:** Es el sustituto artificial de una parte del cuerpo que por diversas razones le haga falta a una persona, la cual le permite a la misma que se sienta mejor y funcione de manera casi natural en sus actividades diarias (ACS, 2020).

**k) Prototipo:** Se entiende como el primer modelo que utiliza como representación o simulación del producto final, permitiendo comprobar el diseño y ratificar que cuenta con las particularidades específicas planteadas (Prototip0, 2020).

**l) Retroalimentación:** Entiéndase como el método de control de sistemas en el cual los resultados obtenidos de una tarea o actividad son reintroducidos de nuevo en el sistema con el fin de controlar y perfeccionar su comportamiento. La retroalimentación se emplea prácticamente a cualquier proceso que implique mecánicas semejantes de ajuste y autorregulación de un sistema.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

m) **Robótica:** Es la ciencia que congrega otros campos tecnológicos, con el vital propósito de diseñar máquinas robotizadas preparadas para realizar diversas tareas automatizadas en función de la capacidad de su software (EDS, 2021).

n) **Sensibilidad:** Es la dimensión de reacción de una variable definida por los cambios de otra; entendiéndose también como el cambio de una variable frente a la variación en un elemento que pueda impactar en su valor, ya sea positiva o negativamente (Westreicher, 2020)

o) **Textura:** Es la organización de los compendios que forman la materia de un cuerpo cualquiera, concretamente referido a los que están en su superficie y son apreciables por vista o tacto. La textura es el aspecto externo y superficial de la estructura de los materiales, objetos y cosas que nos rodean. Dicho término hace alusión a las sensaciones que genera el roce o tacto con cualquier materia u objeto, se encuentra enlazado con la superficie exterior de los cuerpos (Pérez, 2021).

p) **Transhumeral:** Hace referencia a la amputación transhumeral, la cual consiste en una amputación en el área del brazo, cuya separación se da en el hueso del húmero (Totora, 2021).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

# CAPÍTULO III

## 3. DESARROLLO DEL PROYECTO

### 3.1. Metodología

El presente trabajo de investigación se basó en el desarrollo de un sistema robótico biomédico, que le permitiera al usuario de una prótesis poder realizar movimientos de pronación y supinación en la parte de la muñeca, así como la contracción y flexión del codo. Para ello, se desarrolló el sistema robótico biomédico tomando en consideración que: el usuario de la prótesis tenga la capacidad de realizar dichos movimientos, mediante las señales mio-eléctricas los cuales ayudaran para poder mover la prótesis.

Para la realización de la prótesis mio-eléctrica, se llevaron a cabo actividades de la siguiente manera:

- Se estudió la cinemática de la articulación del codo y muñeca.
- Se identificaron las zonas correctas para obtener las señales mio-eléctricas.
- Se diseñó un circuito para la adquisición y acondicionamiento de las señales mio-eléctricas.
- Se identificaron los motores para efectuar el movimiento de las articulaciones, considerando: torque y tamaño.
- Se diseñó la parte mecánica de la prótesis y se procedió a su fabricación.
- Construcción, integración de los componentes: circuito de adquisición y acondicionamiento de señales, tarjeta controladora, motores, batería y piezas de la prótesis.

El estudio se compuso de seis pasos; primeramente, se comenzó con el diseño mecánico de los distintos componentes que conformaron el chasis de la prótesis, esto mediante software de impresión en 3D. Una vez hecho lo anterior, se da lugar al modelado matemático de la prótesis, partiendo del modelo del péndulo simple,



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

considerando los diferentes eslabones que representa cada articulación; posteriormente se llevará a cabo la fabricación de las piezas que componen el chasis de la prótesis, mediante impresión 3D; en seguida, se diseñó el circuito de instrumentación, para la adquisición de señales mio-eléctricas y su acondicionamiento; como siguiente paso, se comenzó con el diseño del algoritmo de control, el cual incluye la adquisición de las señales mio-eléctricas previamente acondicionadas y su interpretación mediante redes neuronales artificiales, también se incluyó el algoritmo de control PID para el control de los motores; aprovechando la tecnología de comunicación Bluetooth que incluye el módulo ESP32, se desarrolló una aplicación móvil, para recibir a través de esta, el estado de la batería y el tiempo de uso de la prótesis; por último, se llevó a cabo la construcción del prototipo, integrando todos los componentes (sensores, motores, tarjeta de control, batería), y se realizaron las pruebas de funcionalidad.

A partir de lo anterior descrito, se presenta la metodológica para un sistema robótico biomédico:

- Se investigaron los transductores: piezorresistivos y capacitivos. Para identificar las respuestas de éstos, ante la fuerza ejercida por la mano al momento de tocar o tomar un objeto.
- Se diseñó un circuito eléctrico que permitió obtener la señal de los dispositivos transductores, esto conllevó diseñar e implementar circuitos, para el acondicionamiento de las señales producidas por los transductores.
- Se identificaron y eligió el microcontrolador, cuya principal característica es el bajo consumo de energía eléctrica (propuestas: ESP32F103 de 32 bits, PIC18F45K50 de 8 bits). El microcontrolador, contiene la lógica del sistema, recibiendo las señales acondicionadas de los transductores e interpretarlas.
- Se identificaron los motores y su ubicación.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- Se integraron los sistemas, mediante la conexión de los diferentes componentes, transductores, circuito de captación y acondicionamiento de señales, microcontrolador y motores.
- Se ejecutó una etapa de pruebas y ajustes.

A continuación, se presenta un diagrama con la representación de las tareas que se realizaron para el correcto funcionamiento del dispositivo.

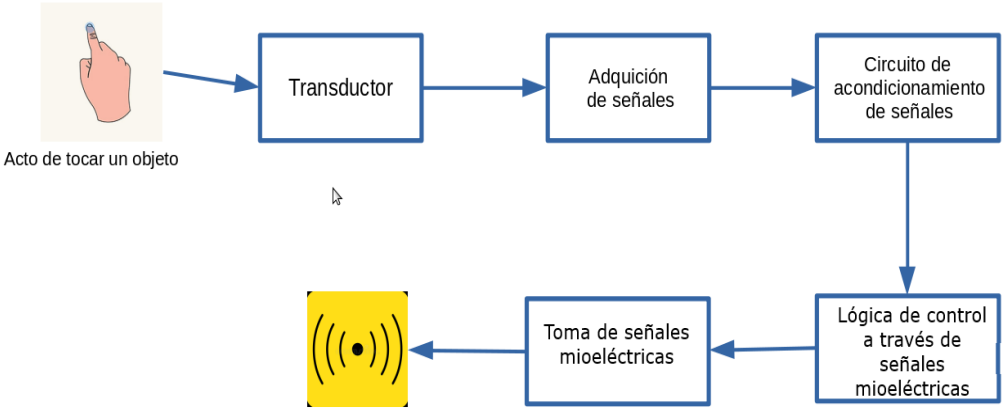



***Figura 26.*** *Diagrama de bloques*

El prototipo de prótesis de mano, muñeca y codo, está planeado para emular algunos de los movimientos de las partes que la componen. Estos movimientos son los siguientes:

- **Mano**: Abrir y cerrar dedos tipo pinza.
- **Muñeca**: Pronación y supinación.
- **Codo**: Flexión y contracción leves.

Con el objetivo principal de este proyecto, se comenzó con el diseño de las partes que componen la estructura (chasis) de la prótesis, cada parte fue diseñada en el software y fueron fabricadas mediante impresión 3D con material PLA. Posterior a su impresión se llevaron a cabo pruebas de funcionalidad, en base al peso ejercido por los componentes, así



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

como el correcto funcionamiento de los mismos, a continuación, se describe este procedimiento.

- **Descripción del diagrama de bloques**

  - ✓ **Transductor:** Esta etapa está compuesta por el sensado la cual está compuesta de amplificación, filtrado y rectificación para esta etapa se usaron los integrados AD620, el TL084, UI7608, electrodos para la toma de señales. Esta información se detalla en el punto *3.5 diseño de circuitos electrónicos.*
  - ✓ **Adquisición de señales:** Esta etapa consiste en amplificar una señal de entrada proveniente de un transductor y es ampliamente usada para incrementar la magnitud de señales de voltaje muy bajo. Esta información se detalla en el punto *3.5.2.1 Amplificación.*
  - ✓ **Circuito de acondicionamiento de señales:** En esta etapa se ve el filtrado de las señales mio-eléctricas que tomamos de la etapa anterior, aquí vemos también la etapa donde se incluyó un amplificador variable de ganancia 2, y rectificación que esta última etapa consiste en rectificar la señal a un nivel de corriente continua. Aquí se usa el TL082, potenciómetro de precisión, toda esta información explicada la encontramos en los puntos *3.5.2.2 filtrado, 3.5.2.3 Amplificador Variable y en la 3.5.2.4 Rectificación.*
  - ✓ **Lógica de control a través de señales mio-eléctricas:** En esta etapa se realiza el aprendizaje de la red neuronal, donde inicialmente se llevó a cabo la adquisición y almacenamiento de los datos de las señales mio-eléctricas, posteriormente, se realizó la etapa de procesamiento de los datos, y finalmente el aprendizaje de la red neuronal. El modelo de red neuronal utilizado es el perceptrón multicapa. Esta información se detalla en el punto *3.10 Aprendizaje de la red neuronal*.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- ✓ **Toma de señales mio-eléctricas:** Consistió en capturar las señales mio-eléctricas a través de los electrodos, esta señal una vez hecho el proceso de aprendizaje de la red neuronal, permitió el movimiento de las articulaciones que componen la prótesis.

### *3.1.1 Diferencia entre Raspberry pi pico frente a ESP32*

- **Ventajas de usar Raspberry Pi Pico:**
  - Es posible usar sus dos núcleos desde microPython.
  - Contiene mayor cantidad de memoria flash en comparación con la ESP32, lo que puede permitir guardar y correr más programas.
  - No contiene interfaz de comunicación Wi-Fi y Bluetooth, por lo que es más factible justificar el uso de un módulo Bluetooth externo.
- **Desventajas de usar Raspberry Pi Pico:**
  - Únicamente tiene 3 módulos de conversión analógico a digital (ADC) disponibles, aunque se puede usar un cuarto ADC (puerto usado como referencia del ADC). Siendo en total 4 ADC's.
  - El prototipo requiere como mínimo 4 ADC's, sin embargo, es recomendable contar con disponibilidad de ADC's contemplando un posible caso en el que se requiera integrar algún otro dispositivo ya sea un sensor u otra señal mio-eléctrica.
  - No contiene interfaz de comunicación Wi-Fi y Bluetooth.
- **Ventajas de usar ESP32:**
  - Contiene mayor cantidad de ADC's en comparación con la Raspberry Pi Pico, siendo 15 en total.
  - Dispone de Bluetooth 4.2.
  - Está pensado para dispositivos IoT.
  - Al disponer de dos núcleos permite dedicar a la comunicación Wi-Fi y el otro al resto de procesos.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- **Desventajas de usar el ESP32:**
  - MicroPython no puede usar sus dos núcleos.
  - Aún no hay una herramienta (biblioteca) que permita usar el Bluetooth incorporado de manera general (para todo tipo de dispositivos), ya que puede funcionar para algunos modelos de celulares, pero para otros modelos no es capaz de aparearse o emparejarse.

Se escogió la ESP 32 por su mayor cantidad de ADC's que nos ofrecía en comparación con el Raspberry pi pico, ya que en nuestro proyecto se necesitan mayor cantidad de ADC's. También, la ESP32 se puede programar fácil y su secuencia de operaciones puede llegar a ser de 240MHz y también cuenta con un gran número de periféricos. Otro motivo por el cual se escogió la ESP32 fue porque la Raspberry pi Pico tiene un mayor consumo de energía eléctrica, y ocupa mayor espacio en comparación con la Esp32. Lo anterior hace inviable el uso de una Raspberry pi Pico en dispositivos donde la prioridad es reducir el espacio y disminuir el consumo de energía eléctrica, como en el prototipo desarrollado.

### 3.2. Programas necesarios del Proyecto

El desarrollo efectivo de la presente investigación hizo necesario el uso de diversos Software que permitieron el funcionamiento de los componentes antes mencionados, los cuales son:

- **Software de impresión 3D:** Utilizamos este software para el diseño mecánico y ***UltiMaker Cura*** para la impresión de la prótesis.
- **Thonny:** Este es un software libre para usar microPython que es un Firmware que permitirá realizar la programación de la placa módulo ESP32.
- **App Inventor**: Se empleó para desarrollar una aplicación Android para mostrar información de la prótesis (nivel de batería, temperatura, humedad y estado de los motores).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## 3.3. Diferencias de las redes neurales perceptrón simple, multicapa y red base radial

### *3.3.1. Diferencia entre redes*

#### 3.3.1.1. Perceptrón simple o monocapa frente a perceptrón multicapa

Las redes monocapa se caracterizan por estar compuestas por una capa de entrada y una capa de salida. Las neuronas ubicadas en la capa de entrada se encuentran conectadas a través de sus pesos con las neuronas que se encuentran en la capa de salida, compartiendo información mediante esta conexión. Las redes monocapa son rápidas en términos de procesamiento debido a que no demandan demasiados cálculos y actualizaciones, pero no deben ser usadas en casos en los que se presenten problemas de no linealidades significativas en el tiempo.

Las redes multicapa se pueden definir como varias redes monocapa conectadas en cascada. A diferencia de las redes monocapa, las multicapas poseen un nivel intermedio entre la salida y la entrada denominado capa oculta, recibe este nombre debido a que este nivel no tiene una relación directa con el entorno. Las neuronas ubicadas en la capa de entrada se encargan de recibir la información que proviene del exterior y propagarla hacia la capa oculta, en este nivel (capa oculta) se realiza el procesamiento de la información que luego es llevado hacia la capa de salida para obtener la respuesta de la red ante el estímulo que recibió de los patrones o vectores de entrada.

El perceptrón simple se caracteriza por poseer una estructura de dos niveles o capas, en el cual, todo el procesamiento se desarrolla en la capa de salida, siendo esta una de sus principales desventajas junto con poder hacer clasificaciones de patrones que solamente sean separables linealmente, como es el caso de una compuerta lógica OR o AND. El perceptrón multicapa fue desarrollado para resolver el problema que presenta el perceptrón simple, este problema es solucionado aumentando el número de capas, agregando las denominadas capas



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

ocultas, que reciben este nombre porque no interactúan directamente con el entorno, y se encuentran entre la capa de entrada y la capa de salida. (Ovando, 2019)

#### 3.3.1.2. Redes de base radial frente a perceptrón multicapa

Las redes de base radial y perceptrón multicapa son dos tipos de redes de neuronas no lineales que poseen sus neuronas agrupadas en capas y las conexiones están dirigidas hacia adelante red-feedforward. Ambas redes son además aproximadores universales, pues ha demostrado que pueden aproximar cualquier función continua.

Estos dos tipos de redes presentan también algunas diferencias como el número de capas ocultas una única capa en el caso de las redes de base radial y tantas capas como se desee en el caso del perceptrón multicapa, la distribución en los pesos de las redes de base radial las conexiones de la capa de entrada a la oculta no llevan pesos asociados y, por ejemplo, también la linealidad en la capa de salida, la cual no es imprescindible para el caso del perceptrón multicapa.

Sin embargo, la principal diferencia entre las redes de base radial y perceptrón multicapa radica en la función de activación de las neuronas ocultas de la red. Las primeras utilizan funciones de base radial, las cuales hacen que las neuronas ocultas de la red posean un carácter local, activándose cada neurona en una determinada región del espacio de entrada. Por otro lado, el perceptrón multicapa emplea funciones de activación sigmoidales, que permiten que las neuronas ocultas de la red posean activación global, es decir, que se activen en todo el espacio de entrada.

El uso de diferentes funciones de activación con las propiedades anteriormente mencionadas, hace que cada uno de estas arquitecturas, perceptrón multicapa y redes de base radial, tenga sus propias características, las cuales se analizan a continuación.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- ✓ **El perceptrón multicapa construye aproximaciones globales**

  Debido al uso de funciones de activación sigmoidal, el perceptrón multicapa construye relaciones globales entre los datos de entrada y salida disponibles. Esto hace que el aprendizaje de la red sea lento, pues el cambio en un solo peso de la red provoca cambios en la salida para todos los patrones de entrada presentados anteriormente, reduciéndose así el efecto de previos ciclos de aprendizaje y retrasando la convergencia del algoritmo de aprendizaje.

- ✓ **Las redes de base radial construyen aproximaciones locales**

  Cada neurona oculta de la red de base radial se especializa en una determinada región del espacio de entrada y construyen una aproximación local en dicha región. Por tanto, la relación que definen las redes de base radial entre los datos de entrada y salida es una suma de funciones no lineales y locales para diferentes regiones del espacio de entrada. A diferencia de cuando se construyen aproximaciones globales, la construcción de aproximaciones globales permite que el aprendizaje sea más rápido, ya que el cambio en un solo peso de red afecta únicamente a la neurona oculta asociada a dicho peso y, por tanto, a un determinado grupo de patrones de entrada, los pertenecientes a la clase que representa la neurona oculta en cuestión.

  Debido al carácter local, el aprendizaje de estas redes, generalmente menos sensible al orden de presentación de los patrones que en el caso del perceptrón multicapa.

  En muchos casos, sin embargo, ocurre que para poder construir una aproximación mediante la suma de aproximaciones locales se requiere un alto número de neuronas ocultas, lo cual podría influir negativamente en la capacidad de generalización de las redes de base radial.

  Finalmente, debe señalarse que el número de neuronas ocultas de la red puede aumentar exponencialmente con la dimensión del espacio de entrada. Por tanto, para



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

aplicaciones que requieren un alto número de variables de entrada, las redes de base radial podrían no ser las más adecuadas. (Pedro Isasi Viñuela, 2004)

### *3.3.2. Justificación de la elección de la red neuronal*

Se utilizó la red neuronal perceptrón multicapa: Porque es la red neuronal más versátil, se puede utilizar para diferentes aplicaciones y resolución de problemas no lineales. A diferencia de las redes neuronales, perceptrón simple, que solo se pueden aplicar a problemas linealmente separables. Por otro lado, la red neuronal de base radial, ha demostrado tener un desempeño cercano a la red perceptrón multicapa, no obstante, su implementando suele ser más complejo, además que demanda un mayor costo computacional para computar la salida de las neuronas de la capa oculta.

***Tabla 8*** *Cuadro comparativo de redes neuronales*

| PERCEPTRÓN SIMPLE | PERCEPTRÓN MULTICAPA | RED DE BASE RADIAL |
|---|---|---|
| **1.** Consta de una capa de entrada y una de salida. No tiene capas ocultas.<br>**2.** Su conexión es unidireccional.<br>**3.** Es una red simple.<br>**4.** Resuelve problemas lineales<br>**5.** Aprendizaje supervisado y conexiones en un solo sentido.<br>**6.** Su aprendizaje es un método de detección del error y corrección.<br>**7.** Es fácil de programar.<br>**8.** Reconocido por su capacidad de aprender y reconocer patrones | **1.** Este modelo se comprende de: capa de entrada que solo recibe señales de entrada y la propaga a la siguiente capa, capa de salida esta proporciona al exterior. Tiene capas ocultas y en cada neurona mínimo 2.<br>**2.** Su conexión es hacia adelante.<br>**3.** Es una red versátil.<br>**4.** Resuelve problemas no lineales y lineales.<br>**5.** Aprendizaje supervisado algoritmo *Backpropagation.*<br>**6.** Su aprendizaje se divide en tres partes: la primera parte es la propagación hacia adelante (*forward propagation*), la segunda es la retro-propagación (*backpropagation*); y, por último, se lleva a cabo el proceso de evaluación.<br>**7.** Su programación es fácil y más entendible.<br>**8.** El rango de tareas que puede realizar el perceptrón multicapa es mucho mayor que simples decisiones y reconocimiento de patrones.<br>**9.** Construye relaciones globales entre los datos de entrada y salida disponibles. | **1.** Está compuesta de tres capas, una capa de entrada, una única capa oculta y una capa de salida. Solo tiene una capa oculta.<br>**2.** Su conexión es hacia adelante.<br>**3.** Es una red compleja.<br>**4.** Resuelve problemas no lineales y lineales.<br>**5.** Aprendizaje hibrido totalmente supervisado.<br>**6.** Su aprendizaje se lleva a cabo con un entrenamiento no supervisado en la capa oculta y posteriormente un entrenamiento supervisado en la capa de salida.<br>**7.** Su programación es fácil, pero de mayor costo computacional.<br>**8.** El rango de tareas que puede realizar es de decisiones y reconocimiento de patrones.<br>**9.** Es de carácter local es lo que las diferencian del perceptrón multicapa, no solo en cuanto a arquitectura, sino también en cuanto a comportamiento. |



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### 3.4. Controlador PID

Uno de los controladores más comunes para motores DC, es el controlador PID (Proporcional-Integral-Derivativo) (García E. , 2008). El controlador PID está definido por la siguiente ecuación:

- **Ecuación 16** $u(t) = K_{p^e}(t) + \frac{K_p}{T_i}\int_t^0 e(t)dt + K_p\, T_d \frac{d_{ve}(t)}{dt}$ (25)

Donde *e(t)* es el error de la señal y *u(t)* es la entrada de control del proceso; $K_p$ es la ganancia proporcional; $T_i$ es la constante de tiempo integral y $T_d$ es la constante derivativa.

En el dominio de frecuencia (*s*), el controlador PID se puede definir como:

- **Ecuación 17** $U(s) = K_p \left[1 + \frac{1}{T_i s} + T_d s\right] E(s)$ (26)

Donde $K_p$, $T_i$ y $T_d$, son parámetros que deben ser ajustados para conseguir el óptimo funcionamiento del controlador.

Para obtener el valor de los parámetros se decidió emplear el método de Ziegler-Nichols. Este consiste en un análisis en laso abierto del sistema a controlar. En lazo abierto se pueden definir procesos de acuerdo con la función de transferencia siguiente:

- **Ecuación 18** $G(s) = \frac{K_0 * e^{-st_o}}{(1+s\gamma 0)}$ (27)

Los coeficientes $\kappa_0$, $\tau_0$ y $\gamma_0$ se obtienen de la observación de la respuesta al sistema en lazo abierto dado una entrada escalón. El análisis parte del sistema estable (o en reposo) donde *y(t) = y0* con *u(t) = u₀*; se aplica una entrada escalón de $u_0$ a $u_1$, y se registra la respuesta de la salida hasta que se estabilice en el nuevo punto de operación. A través de la observación de esta respuesta se pueden obtener los siguientes parámetros:

$$t_0 = t_1 - t_0$$

$$\gamma_0 = t_2 - t_1$$



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

$$k_0 = \frac{y_1 - y_0}{u_1 - u_0} \qquad (28)$$

Según Ziegler-Nichols, la relación de los coeficientes con los parámetros del controlador es:

- **Ecuación 19** $K_p = \frac{1{,}2 * \gamma 0}{K_0 * t_0}, \quad T_i = 2 * t_0, \quad T_d = 0{,}5 * t_0$ (29)

Dado que, el controlador a implementar será implementado en un sistema digital, se emplea el controlador PID discreto dado por la transformada Z, que se describe a continuación:

- **Ecuación 20** $U(z) = E(z)K_p \left[1 + \frac{T}{T_{i\,(1-z^{-1})}} + T_d \frac{(1-z^{-1})}{T}\right]$ (30)

También:

- **Ecuación 21** $\frac{U(z)}{E(z)} = a + \frac{b}{1-z^{-1}} + c\,(1 - z^{-1})$ (31)

Donde:

- **Ecuación 22** $a = K_p, \quad b = \frac{K_p * T}{T_i}, \quad c = \frac{K_p * T_d}{T}$ (32)

A continuación, se presenta el algoritmo para la implementación del controlador PID utilizado. Donde se tienen como entradas: $S_{adc}$, que corresponde a la señal del ADC correspondiente a la captura de la señal mio-eléctrica; *set point* es el valor que se desea alcanzar que, corresponde al ángulo de inclinación del codo; $V_{min}$ es el valor PWM mínimo a la salida del PID, que normalmente es de valor 0; $V_{max}$ es el valor de salida PWM máximo a generar que, para el caso del ESP32 con PWM configurado a 10 bits, la salida máxima corresponde a 1023.

La salida que se obtiene del controlador PID ($s_{PID}$), es de tipo PWM acotado entre los valores $V_{min}$ *y* $V_{max}$.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

**Algoritmo 2.2.1:** Algoritmo para implementación del controlador PID.

**Input:** $S_{adc}$, $setpoint$, $V_{min}$, $V_{max}$

1 Computar error $\rightarrow e$;
2 Computar termino integral $(PI) \rightarrow b*e*PI[t-1]$;
3 Computar termino derivativo $(PD) \rightarrow c*e-e[t-1]$;
4 Obtener salida PID $(s_{PID}) \rightarrow PI+PD+a*e$;
5 **if** $s_{PID} > V_{max}$ **then**
6 $\quad s_{PID} = V_{max}$;
7 **else if** $s_{PID} < V_{min}$ **then**
8 $\quad s_{PID} = V_{min}$;
9 Almacenar tiempo actual $\rightarrow PI[t-1]$;
10 Almacenar error actual $\rightarrow e[t-1]$;
11 **return** $s_{PID}$;

***Figura 27.*** *Algoritmo para implementación del controlador PID*

## 3.5. Aplicación de los circuitos electrónicos

### *3.5.1 Adquisición de las señales mio-eléctricas*

En esta sección se presenta el desarrollo de la etapa de instrumentación para la adquisición de las señales mio-eléctricas. Esta etapa está compuesta por el sensado (transductor), amplificación, filtrado y rectificación. En la figura 28, se muestra el circuito utilizado para la adquisición de las señales mio-eléctricas, donde se muestran las etapas que lo componen por las letras A, B, C, D, E y F. (*Ver en el anexo13 del circuito en formato A4*)



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

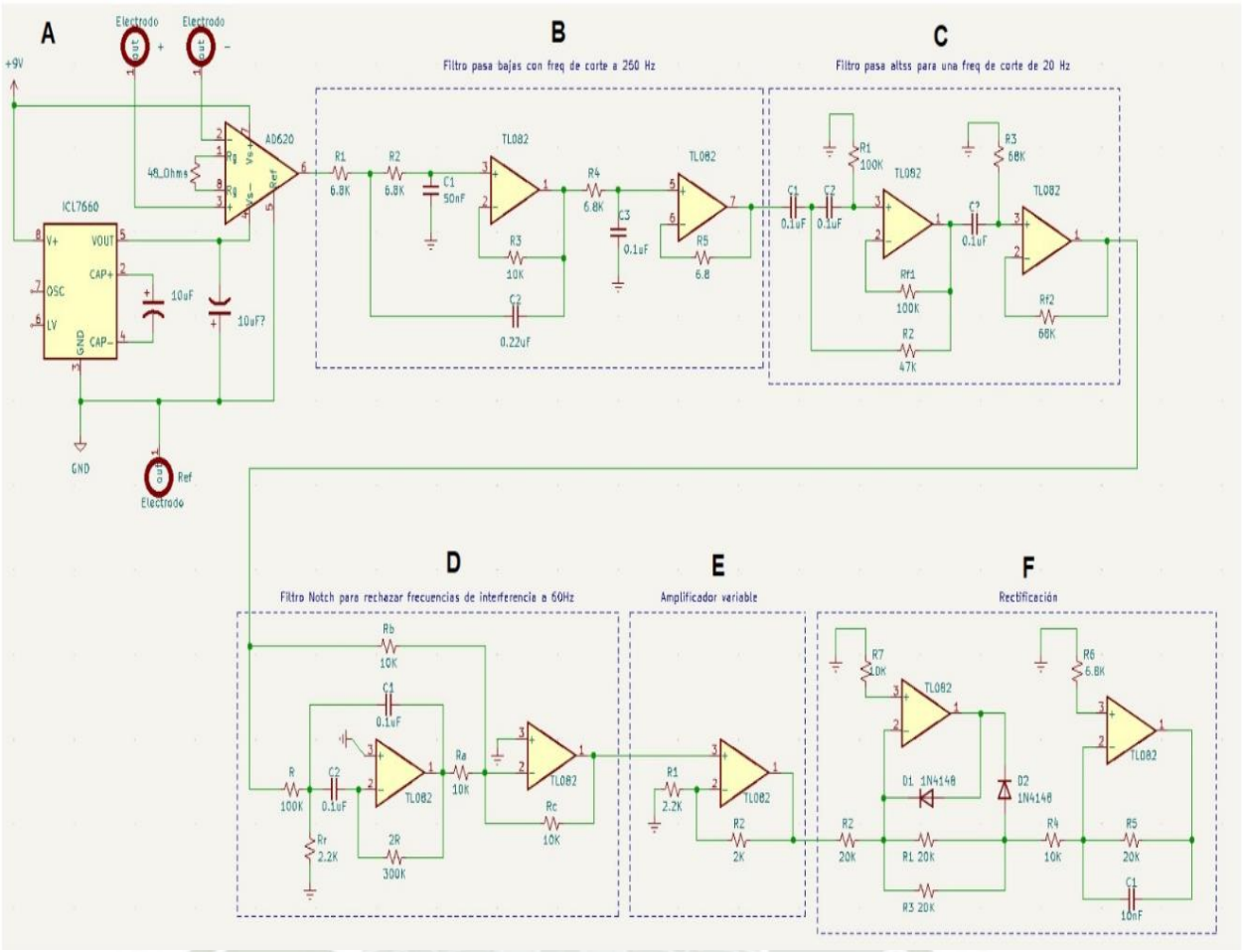



***Figura 28.*** *Circuito para la adquisición de señales mio-eléctricas.*

La etapa de sensado consistió en la selección e implementación del tipo de sensor, que para este trabajo se emplearon electrodos superficiales, los cuales permiten captar las señales mio-eléctricas producidas por la diferencia de potencial generada por los músculos del cuerpo, por lo que la primera fase de la etapa de instrumentación está cubierta.

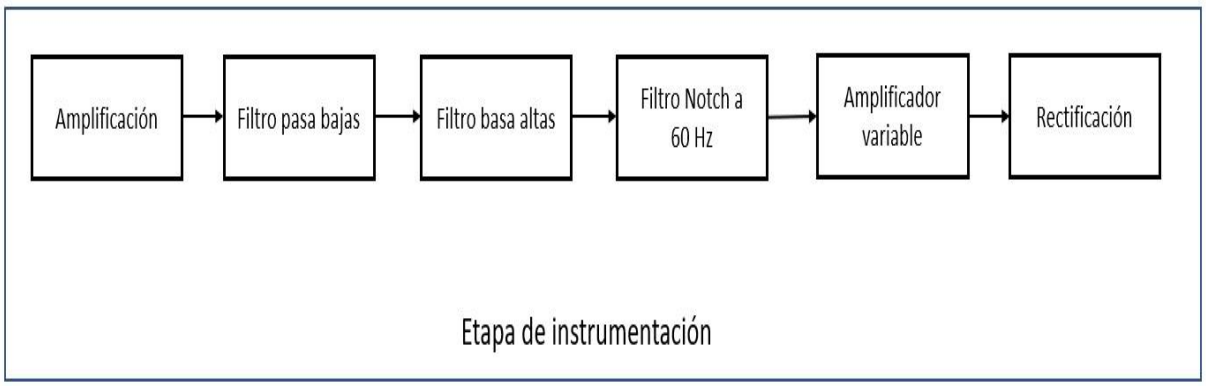



***Figura 29.*** *Diagrama de bloques de la etapa de instrumentación.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- **Descripción del diagrama de bloques**

  - ✓ **Amplificación:** Consiste en amplificar una señal de entrada proveniente de un transductor y es ampliamente usada para incrementar la magnitud de señales de voltaje muy bajo. Esto se detalla en el punto 3.5.2.1.
  - ✓ **Filtro pasa bajas:** Empleado para limpiar la señal mio-eléctrica de frecuencias de interferencia, y consiste de dos filtros conectados en cascada, el primero de segundo orden y el segundo de primer orden con una frecuencia de corte de 250 Hz, con lo que se obtiene un filtro con pendiente de 60 dB/década. Esto se detalla en el punto 3.5.2.2.
  - ✓ **Filtro pasa altas:** Se implementa para limitar frecuencias altas con una frecuencia de corte de 20 Hz. Este se encuentra conectado en cascada con el filtro pasa bajas presentado anteriormente con lo cual se forma un filtro pasa banda, cuyo rango de frecuencia permisible es de 20 Hz (frecuencia de corte del filtro pasa altas) a 250 Hz (frecuencia de corte del filtro pasa bajas). Esto se detalla en el punto 3.5.2.2.
  - ✓ **Filtro Notch:** Este filtro está compuesto por un filtro pasa banda angosta y un sumador, su funcionamiento se basa en dejar pasar las frecuencias seleccionadas, pero con signo contrario a la señal de entrada, posteriormente en el sumador entra la señal de entrada, y la salida del filtro pasa banda, con lo cual el voltaje de salida es una resta entre la señal de entrada y la salida del filtro pasa banda. Esto se detalla en el punto 3.5.2.2.
  - ✓ **Amplificador variable:** En esta etapa se incluye un amplificador variable de ganancia 2, y para su construcción se utiliza el integrado TL082, donde se configura la ganancia en modo variable para ajustarla la amplitud de la señal mio-eléctrica de acuerdo a cada usuario. Esto se detalla en el punto 3.5.2.3.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- ✓ **Rectificación:** Esta última etapa consiste en rectificar la señal a un nivel de corriente continua esto debido a que la naturaleza de la señal mio-eléctrica es alterna (analógica). Esto se detalla en el punto 3.5.2.4.

### *3.5.2. Circuito de adquisición de señales mio-eléctricas*

A continuación, se presentan las fases de amplificación, filtrado y adquisición de las señales mio-eléctricas.

#### 3.5.2.1. Amplificación

Esta etapa consiste en amplificar una señal de entrada proveniente de un transductor y es ampliamente usada para incrementar la magnitud de señales de voltaje muy bajo. Debido a que las señales mio-eléctricas son generadas en el orden de mili-volts, la amplificación de esta señal es un paso necesario. Para llevar a cabo la amplificación de la señal mio-eléctrica se empleó el amplificador de instrumentación *AD620*, que se presenta en la figura 30.

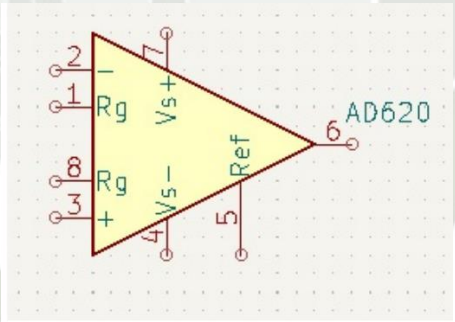



***Figura 30.*** *Amplificador de instrumentación AD620.*

Debido a que el *AD620* requiere de una fuente simétrica para su funcionamiento, se implementó al circuito de amplificación el integrado *TC7660* presentado en la figura 31, que permite generar una señal de voltaje negativo al recibido.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

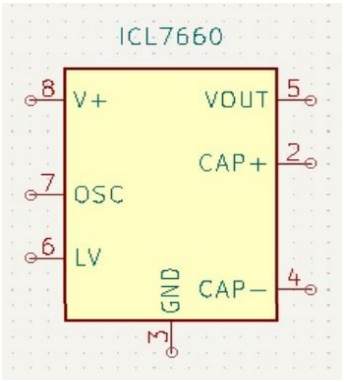



***Figura 31.*** *Integrado TC7660.*

En la figura 32, se presenta el circuito de amplificación, que está compuesto inicialmente por el integrado *TC7660*, el cual recibe por el pin *V+* un voltaje positivo de 9 V y por el pin *VOUT* se entrega el voltaje negativo de -9 V (el circuito de funcionamiento de este integrado viene dado por el fabricante).

Con lo anterior es generada una fuente simétrica que permite alimentar al amplificador de instrumentación *AD620*, donde la ganancia viene data por la ecuación 23

- **Ecuación 23** $\boldsymbol{G = \frac{49{,}4\ K\Omega}{R_G} + 1}$ (33)

Por lo tanto, la resistencia de ganancia $\boldsymbol{R_G}$ se puede obtener por la ecuación 24.

- **Ecuación 24** $R_G = \frac{49{,}4\ K\Omega}{G-1}$ (34)

Con lo anterior, se configuró el amplificador de instrumentación para generar una ganancia de 1000 implementando una resistencia de 48 Ohms (ecuación 25)

- **Ecuación 25** $\boldsymbol{R_G = \frac{49{,}4\ K\Omega}{1000-1} = 49{,}44\Omega}$ (35)

Por las entradas (-) y (+) se ingresaron las señales mio-eléctricas obtenidas a través de los electrodos de monitoreo 3M superficiales, además, se empleó un tercer electrodo utilizado como referencia. La señal que se obtendrá de los electrodos es muy pequeña, difícil



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

de poder tomar las señales en el osciloscopio, es por ello que se emplea el circuito de amplificación.

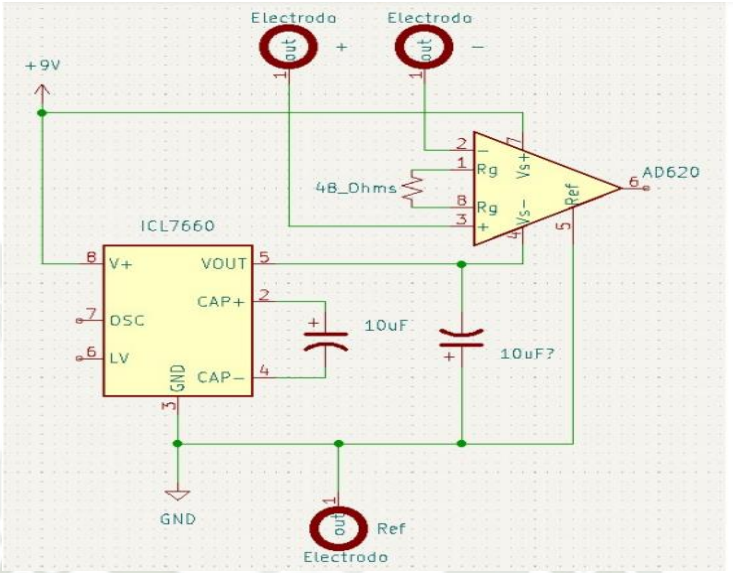



***Figura 32.*** *Circuito de amplificación*

### 3.5.2.2. Filtrado

La etapa anterior permite recibir la señal mio-eléctrica y amplificarla, al hacer esto no solo se amplifica la señal de interés (mio-eléctrica) sino que también señales no deseadas (ruido) generado por diferentes factores, como el sudor, tipo de material conductor, etc. Debido a esto es necesario implementar filtros que permitan eliminar los diferentes efectos de ruido que "contaminan" a la señal mio-eléctrica, por lo que en esta etapa con el objetivo de reducir el ruido en las señales se implementaron filtros pasa bajas, pasa altas y filtro notch o muesca.

Para la implementación de los filtros, se utilizó el integrado TL082, que está compuesto por unos dos amplificadores de propósito general, por lo que tienen dos canales de entrada tipo JFET (*Junction Field-Effect Transistor*) con corriente de consumo de 2.5 mA por canal.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

**Filtro pasa bajas**

Este filtro fue empleado para limpiar la señal mio-eléctrica de frecuencias de interferencia, y consiste de dos filtros conectados en cascada, el primero de segundo orden y el segundo de primer orden con una frecuencia de corte de 250 Hz, con lo que se obtiene un filtro con pendiente de 60 dB/década. En la figura 33, se muestra el circuito del filtro utilizado, donde la entrada corresponde a la salida de la etapa anterior (salida del amplificador de instrumentación *AD620*).

Los parámetros de este filtro se obtuvieron basándonos en la frecuencia de corte (250 Hz). Estos filtros son de tipo Butterworth; primero se definió: la atenuación de la frecuencia de corte (Ac) = 3 dB, atenuación en decibeles de la frecuencia de paro (Ap) = 60 dB, frecuencia de corte (fc) = 250 Hz y frecuencia de paro (fp) = 2500 Hz.

El orden del filtro se obtuvo mediante la ecuación 26.

- **Ecuación 26** $$n = \frac{\log\left(\frac{\varepsilon_2}{\varepsilon_1}\right)}{\log\left(\frac{fp}{f_c}\right)} \quad (36)$$

Donde: $$\varepsilon_1 = \sqrt{10^{0.1*A_c} - 1}$$

$$\varepsilon_2 = \sqrt{10^{0.1*A_p} - 1}$$

Por lo tanto: $$\varepsilon_1 = 0{,}997$$

$$\varepsilon_2 = 999{,}99$$

$$n = \frac{\log\left(\frac{999{,}99}{0{,}997}\right)}{\log\left(\frac{2500}{250}\right)} = 3{,}001$$

Partiendo de la ecuación 39 se propuso el valor de la capacitancia de $C_3$ = 0.1µF.

- **Ecuación 27** $$C_n = \frac{C_i}{2\pi f_{cR}} \quad (37)$$



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

El factor de escalonamiento se obtuvo por:

- **Ecuación 28** $R = \frac{1}{2\pi f_c C} = \frac{1}{2\pi(250)(0{,}1)} = 6{,}36\ K\Omega$ (38)

De acuerdo con la tabla de coeficientes para el diseño de filtros Butterworth se tienen los valores: $C_1$ = C (1.414) y $C_2$ = C (0.7071). Por lo tanto, los valores de capacitancia son: $C_1$ = 0.1415 µF; $C_2$ = 0.07 µF. Por conveniencia, se usaron valores comerciales de capacitancia aproximados, donde $C_1$ = 0.05 µF o 50 nF y $C_2$ = 0.22µF.

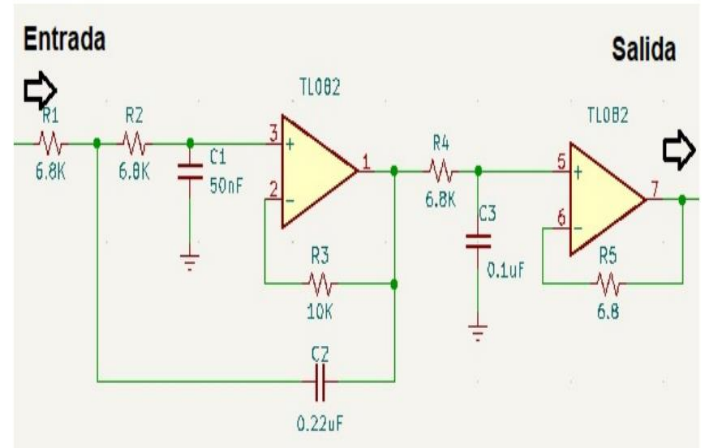



***Figura 33.*** *Circuito de filtro pasa bajas*

**Filtro pasa altas**

Este filtro se implementó para limitar frecuencias altas con una frecuencia de corte de 20 Hz. Este se encuentra conectado en cascada con el filtro pasa bajas presentado anteriormente con lo cual se forma un filtro pasa banda, cuyo rango de frecuencia permisible es de 20 Hz (frecuencia de corte del filtro pasa altas) a 250 Hz (frecuencia de corte del filtro pasa bajas).

En la figura 34, se presenta el circuito implementado para el filtro pasa altas. Para el diseño de este filtro se partió de la frecuencia de corte definida, y los siguientes valores propuestos de: atenuación en decibeles de la frecuencia de corte ($A_c$) = 3 dB; atenuación en decibeles ($A_p$) = 60 dB; frecuencia de corte ($f_c$) = 20 HZ; frecuencia de paro ($f_p$) = 2 Hz



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

El orden del filtro se obtuvo obedeciendo la ecuación 38, donde:

$$\varepsilon_1 = 0{,}997$$

$$\varepsilon_2 = 999{,}999$$

Por lo tanto, la orden de filtro pasa altas se da en la ecuación

- **Ecuación 29** $$n = \frac{\log\left(\frac{999{,}99}{0{,}997}\right)}{\log\left(\frac{2}{20}\right)} = 3{,}001 \qquad (39)$$

Se propuso el valor de capacitancia del filtro C = 0.1 µG y el factor de escalonamiento del primer orden se obtuvo usando la ecuación 41.

- **Ecuación 30** $$R = \frac{1}{2\pi f_c C} = \frac{1}{2\pi(20)(0{,}1)} = 79{,}57\ K\Omega \qquad (40)$$

Por lo tanto: $$R_1 \frac{R}{0{,}7071} = 112{,}54\ K\Omega$$

$$R_2 \frac{R}{1{,}414} = 56{,}27\ K\Omega$$

Los valores de las resistencias empleadas para la realización del circuito se basaron en los valores comerciales, por lo que R1 = 100 KΩ y R2 = 47 KΩ.

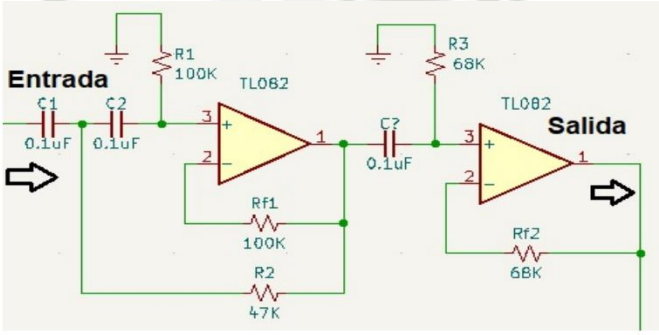



***Figura 34.*** *Circuito del filtro pasa altas*

**Filtro notch**

Este filtro está compuesto por un filtro pasa banda angosta y un sumador, su funcionamiento se basa en dejar pasar las frecuencias seleccionadas, pero con signo contrario a la señal de entrada, posteriormente en el sumador entra la señal de entrada, y la salida del



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

filtro pasa banda, con lo cual el voltaje de salida es una resta entre la señal de entrada y la salida del filtro pasa banda. Por lo tanto, para las frecuencias a las que se diseña el filtro, el voltaje de salida es 0 V, con lo cual se suprime el rango de frecuencias.

El filtro fue implementado para rechazar las frecuencias consideradas como interferencias de la línea eléctrica (60 Hz), el cual se adaptó con un ancho de banda de *10 Hz, obteniendo como resultado una banda rechazada de 55 Hz a 65 Hz*.

En la figura 35, se presenta el circuito para la implementación del filtro notch. El ancho de banda BW = 10 Hz, la frecuencia de corte es $f_c$ = 60 Hz, ya que esta se desea eliminar.

Se tiene que:

- **Ecuación 31** $Q = \frac{f_c}{BW} \therefore Q = \frac{60}{10} = 6$ (41)

Considerando que:

- **Ecuación 32** $BW = \frac{1}{2\pi\, RC}$ (42)

Se propone **C = 0.1 µF** y se despeja R de la ecuación 28:

- **Ecuación 33** $R = \frac{1}{2\pi BWC} = \frac{1}{2\pi(10)(0{,}1\mu F)} = 159{,}15\ K\boldsymbol{\Omega}$ (43)

La ganancia del circuito tiene un valor máximo de 1 cuando la frecuencia sea $f_c$ y 2R es el doble de la resistencia de entrada, por lo tanto:

$$2R = 318{,}3\ K\boldsymbol{\Omega}$$

La resistencia $R_r$ depende del valor de R y Q:

$$R_r = \frac{R}{2Q^2 - 1} \quad (44)$$

Finalmente se tiene: $R_r = \frac{159{,}15\ K\boldsymbol{\Omega}}{26^2 - 1} = 2{,}241\ K\boldsymbol{\Omega} \approx \mathbf{2{,}2}\ K\boldsymbol{\Omega}$



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Para el caso de R y 2R se usaron valores aproximados a los calculados: 100 KΩ y 300 KΩ respectivamente.

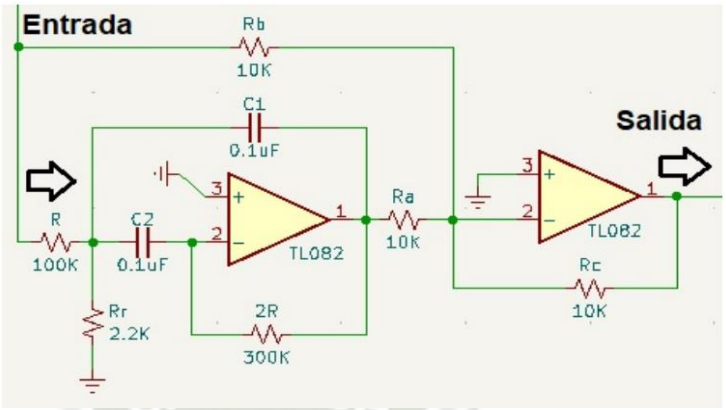



***Figura 35.*** *Circuito del filtro notch.*

Con esto se concluye la etapa de filtrado con lo cual se obtiene la parte más significativa de la señal mio-eléctrica, por lo que las señales débiles (de baja energía) y frecuencias consideradas como ruido son despreciadas.

***Tabla 9*** *Frecuencias de trabajo para filtros*

| Etapa | Rango | Resistencias calculada/valor comercial | | OPAMP |
|---|---|---|---|---|
| ***Filtro pasa bajos*** | *Fc*=250Hz<br>*Fp*=2500Hz | R=6.36KΩ | R=6.8KΩ | TL082 |
| ***Filtro pasa altos*** | *Fc*=20Hz<br>*Fp*=2Hz | R=79.57KΩ<br>R1=112.54KΩ<br>R2=56.27 | R=68KΩ<br>R1=100KΩ<br>R2=47KΩ | TL082 |
| ***Filtro Notch*** | 55Hz a 65Hz | R=159.15KΩ<br>2R=318.3KΩ<br>Rr=2.24KΩ | R=100KΩ<br>2R=300KΩ<br>Rr=2.24KΩ | TL082 |
| ***Amplificador variable*** | Ganancia 2 | - | - | - |

### 3.5.2.3. Amplificador variable

En esta etapa se incluyó un amplificador variable de ganancia 2, y para su construcción se utilizó el integrado TL082, donde se configuró la ganancia en modo variable para ajustarla la amplitud de la señal mio-eléctrica de acuerdo a cada usuario. En la figura 36, se presenta el circuito utilizado para el amplificador variable, donde la entrada



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

corresponde a la salida de la etapa de filtrado; la resistencia R2 es reemplazada por un potenciómetro de precisión para hacer el ajuste de la ganancia. Idealmente la salida de un amplificador viene dada por la ecuación 45.

- **Ecuación 34** $V_{out} = V_{R_f} + 1 * R_{in}$ (45)

Sustituyendo por ley de Ohm se tiene:

$$V_{out} = \frac{V_{in}}{R_{in}} V_{R_f} + \frac{V_{in}}{R_{in}} V_{R_{in}}$$

El voltaje de entrada en un amplificador es:

$$V_{in} = R_{in} \rightarrow I = \frac{V_{in}}{R_{in}}$$

Por lo tanto, la salida del amplificador se obtiene como:

$$V_{out} = V_{in}\left(\frac{R_f}{R_{in}} + 1\right)$$

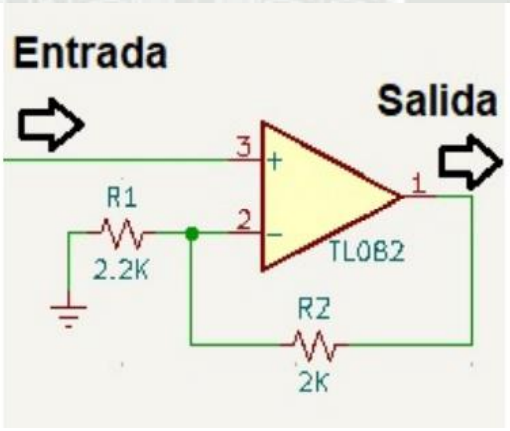



***Figura 36.*** *Circuito del amplificador variable.*

### 3.5.2.4. Rectificación

Esta última etapa consiste en rectificar la señal a un nivel de corriente continua esto debido a que la naturaleza de la señal mio-eléctrica es alterna (analógica). Para esto se implementó un circuito MAV (*Median Absolute Value*) o valor medio absoluto, se utilizó el



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

integrado TL082 para su implementación. El circuito se analizó en dos partes, como rectificador de onda completa y posteriormente como sumador.

En la figura 37, se presenta el circuito de rectificación, donde la entrada corresponde a la salida del amplificador variable.

Considerando inicialmente un semiciclo positivo:

El diodo D2 conduce y D1 no conduce. En este caso, el amplificador se comporta como inversor y por lo tanto a la entrada del sumador es:

$$V_{s1} = -V_{in} \qquad (46)$$

$V_s$ y $V_{in}$ entran al sumador, por lo que la salida es:

$$V_{out} = -\left(\frac{R_f}{R_i} * V_{s1} + \frac{R_f}{R_i} * V_{in}\right) \qquad (47)$$

$$V_{out} = \left(\frac{R}{\frac{R}{2}} * V_{s1} + \frac{R}{R} * V_{in}\right) = -(-2V_{in} + V_{in})$$

Por lo tanto: $V_{out} = V_{in}$

El segundo paso es analizar el semiciclo negativo: En esta etapa en diodo D2 no conduce y D1 conduce, por lo tanto, se obtiene la parte negativa de la señal de entrada, y la salida es:

$$V_{out} = V_{in}$$

El capacitor en el sumador tiene la función de integrar la señal rectificada para obtener así el valor promedio de la señal. Debido a que la entrada del sumador es una señal con rectificación de onda completa, se obtiene el voltaje DC de la señal como:

$$V_{dc} = \frac{2}{T}\int_0^{\frac{T}{2}} V_p sen\,(wt)\,dt \qquad (48)$$



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Considerando que $T = 2\pi$:

$$V_{dc} = \frac{2}{2\pi} \int_0^{\pi} V_p sen\,(wt)\,dt \qquad (49)$$

$$V_{dc} = \frac{1}{\pi}\left(-V_p \cos(\pi) + V_{pcos}(0)\right)$$

Por último, se tiene:

$$V_{dc} = \frac{2V_p}{\pi} \qquad (50)$$

La salida obtenida posteriormente será adquirida por el microcontrolador para su reconocimiento.

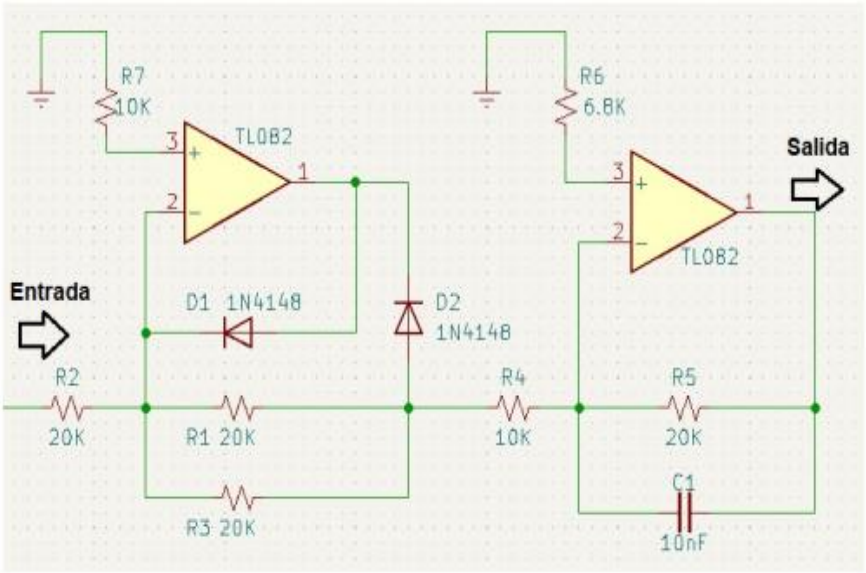



***Figura 37.*** *Circuito del rectificador*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## 3.6. Cálculos

### *3.6.1. Amplificación*

La ganancia viene data por la ecuación 35

- **Ecuación 35** $G = \frac{49{,}4k\Omega}{R_G} + 1$ (51)

Por lo tanto, la resistencia de ganancia $\boldsymbol{R_G}$ se puede obtener por la ecuación 36.

- **Ecuación 36** $R_G = \frac{49{,}4k\Omega}{G-1}$ (52)

Con lo anterior, se configuró el amplificador de instrumentación para generar una ganancia de 1000 implementando una resistencia de 48 Ohms (ecuación 37).

- **Ecuación 37** $R_G = \frac{49k\Omega}{1000-1} = 49{,}44k\Omega$ (53)

### *3.6.2. Filtro pasa bajas*

Los parámetros de este filtro se obtuvieron en base a la frecuencia de corte (250 Hz). Primero se definió: la atenuación de la frecuencia de corte (Ac) = 3 dB, atenuación en decibeles de la frecuencia de paro (Ap) = 60 dB, frecuencia de corte (fc) = 250 Hz y frecuencia de paro (fp) = 2500 Hz. El orden del filtro se obtuvo mediante la ecuación 38.

- **Ecuación 38** $\eta = \frac{\log\left(\frac{\varepsilon_2}{\varepsilon_1}\right)}{\log\left(\frac{fp}{fc}\right)}$ (54)

Donde: $\varepsilon_1 = \sqrt{10^{0{,}1*A_c-1}}$

$\varepsilon_2 = \sqrt{10^{0{,}1*A_p-1}}$

Por lo tanto: $\varepsilon_1 = 0{,}997$

$\varepsilon_2 = 999{,}99$

$\eta = \frac{\log\left(\frac{999{,}99}{0{,}997}\right)}{\log\left(\frac{2500}{250}\right)} = 3{,}001$



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Partiendo de la ecuación 39 se propuso el valor de la capacitancia $C_3 = 0.1\mu F$.

- **Ecuación 39** $$C_n = \frac{C_i}{2\pi f_c R} \quad (55)$$

El factor de escalonamiento se obtuvo por:

$$R = \frac{1}{2\pi f_c C} = \frac{1}{2\pi(250)(0{,}1)} = 6{,}36K\Omega$$

De acuerdo con la tabla de coeficientes para el diseño de filtros Butterworth se tienen los valores:$C_1 = C(1.414)$ y $C_2 = C(0.7071)$. Por lo tanto, los valores de capacitancia son:$C_1 = 0.1415\mu F$; $C_2 = 0.07\mu F$.

Por conveniencia, se usaron valores comerciales de capacitancia aproximados, donde $C_1 = 0.05\mu F$ o 50nF y $C_2 = 0.22\mu F$.

### *3.6.3. Filtro pasa altas*

El orden del filtro se obtuvo obedeciendo la ecuación 38, donde:

$$\varepsilon_1 = 0{,}997$$

$$\varepsilon_2 = 999{,}99$$

Por lo tanto, la orden de filtro pasa altas se da en la ecuación

- **Ecuación 40** $$\eta = \frac{\log\left(\frac{999{,}99}{0{,}997}\right)}{\log\left(\frac{2500}{250}\right)} = 3{,}001 \quad (56)$$

Se propuso el valor de capacitancia del filtro C = 0.1 µG y el factor de escalonamiento del primer orden se obtuvo usando la ecuación 41.

- **Ecuación 41** $$R = \frac{1}{2\pi f_c C} = \frac{1}{2\pi(20)(0{,}1)} = 79{,}57K\Omega \quad (57)$$



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Por lo tanto: $$R_1 = \frac{R}{0{,}7071} = 112{,}54K\Omega$$

$$R_2 = \frac{R}{1{,}414} = 56{,}27K\Omega$$

Los valores de las resistencias empleadas para la realización del circuito se basaron en los valores comerciales, por lo que $R_1 = 100K\Omega$ y $R_2 = 47K\Omega$.

### *3.6.4 Filtro notch*

El ancho de banda BW = 10 Hz, la frecuencia de corte es $f_c = 60Hz$, ya que ésta se desea eliminar.

Se tiene que:

- **Ecuación 42** $$Q = \frac{f_c}{BW} \therefore Q = \frac{60}{10} = 6 \quad (58)$$

Considerando que

- **Ecuación 43** $$BW = \frac{1}{2\pi RC} \quad (59)$$

Se propone $C = 0.1\mu F$ y se despeja R de la ecuación 43:

- **Ecuación 44** $$R = \frac{1}{2\pi BWC} = \frac{1}{2\pi(10)(0{,}1\mu F)} = 159{,}15K\Omega \quad (60)$$

La ganancia del circuito tiene un valor máximo de 1 cuando la frecuencia sea $f_c$ y 2R es el doble de la resistencia de entrada, por lo tanto:

$$2R = 318{,}3K\Omega$$

La resistencia $R_r$ depende del valor de R y Q:

$$R_r = \frac{R}{2Q^2 - 1} \quad (61)$$

Finalmente se tiene: $R_r = \frac{159{,}15K\Omega}{26^2-1} = 2{,}241K\Omega \approx 2{,}2\text{K}\Omega$



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Para el caso de R y 2R se usaron valores aproximados a los calculados: 100 KΩ y 300 KΩ respectivamente.

### *3.6.5. Amplificador variable*

Idealmente la salida de un amplificador viene dada por la ecuación 45.

- **Ecuación 45** $V_{out} = V_{Rf} + I * R_{in}$ (62)

Sustituyendo por ley de Ohm se tiene:

$$V_{out} = \frac{V_{in}}{R_{in}} V_{Rf} + \frac{V_{in}}{R_{in}} V_{R_{in}}$$

El voltaje de entrada en un amplificador es:

$$V_{in} = R_{in} \rightarrow I = \frac{V_{in}}{R_{in}}$$

Por lo tanto, la salida del amplificador se obtiene como:

$$V_{out} = V_{in}\left(\frac{R_f}{R_{in}} + 1\right)$$

### *3.6.6. Rectificación*

Considerando inicialmente un semiciclo positivo: El diodo D2 conduce y D1 no conduce. En este caso, el amplificador se comporta como inversor y por lo tanto a la entrada del sumador es:

$$V_{s1} = -V_{in}$$

$V_s$ y $V_{in}$ entran al sumador, por lo que la salida es:

$$V_o ut = -\left(\frac{R_f}{R_I} * V_{s1} + \frac{R_f}{R_I} * V_{in}\right) \quad (63)$$

$$R_f = R$$



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

$$V_{out} = \left( \frac{R}{\frac{R}{2}} * V_{s1} + \frac{R}{R} * V_{in} \right) = -(2V + V_{in})$$

Por los tanto:

$$V_{out} = V_{in}$$

El segundo paso es analizar el semiciclo negativo: En esta etapa en diodo D2 no conduce y D1 conduce, por lo tanto, se obtiene la parte negativa de la señal de entrada, y la salida es:

$$V_{out} = V_{in}$$

El capacitor en el sumador tiene la función de integrar la señal rectificada para obtener así el valor promedio de la señal. Debido a que la entrada del sumador es una señal con rectificación de onda completa, se obtiene el voltaje DC de la señal como:

$$V_{dc} = \frac{2}{T} \int_0^{\frac{T}{2}} V_p \operatorname{sen}(wt) dt \qquad (64)$$

Considerando que T = 2π:

$$V_{dc} = \frac{2}{2\pi} \int_0^{\pi} V_p \operatorname{sen}(wt) dt$$

$$V_{dc} = \frac{1}{\pi} \left( -V_p \cos(\pi) + V_p \cos(0) \right)$$

Por último, se tiene:

$$V_{dc} = \frac{2V_p}{\pi} \qquad (65)$$



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

## 3.7. Software

### *3.7.1. Algoritmo principal para el funcionamiento de la prótesis*

A continuación, se presenta el algoritmo desarrollado para el funcionamiento de la prótesis. El programa principal está compuesto de diferentes funciones y clases, los cuales están adjuntados en el apéndice de este documento. El algoritmo 1, muestra los pasos propuestos para que la prótesis pueda ejecutar los movimientos de la mano, muñeca y codo.

- **Algoritmo 1**

**Algoritmo 1:** Algoritmo principal para el funcionamiento de la prótesis.

```
1  Configurar puertos de entradas y salidas para motores y sensores;
2  Configurar segundo puerto serial;
3  Inicializar red neuronal;
4  Posicionar motores en modo inicial;
5  Capturar señal mio-eléctrica → S;
6  Normalizar S a valores entre −1 y 1;
7  Clasificar S con red neuronal → y;
8  if y = [1, −1] then
9  |  Accionar mano;
10 |  if Motor no se ha movido then
11 if x = [1, 1] then
12 |  Accionar muñeca;
13 |  if Motor no se ha movido then
14 if x = [−1, 1] then
15 |  Accionar codo;
16 Obtener lectura de temperatura y nivel de batería;
17 Enviar información de temperatura y nivel de batería al puerto serial 2;
```

***Figura 38.*** *Algoritmo principal para el funcionamiento de la prótesis*

El algoritmo propuesto, se implementó en lenguaje MicroPython, y el código 1, muestra la función principal codificada en base al algoritmo 1.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

```
def main():
  count = 0                                          # contador de iteraciones
  actualPosA = 100                                   # posición inicial mano
  actualPosB = 170                                   # posición inicial muñeca (supinación
    )
  nivelBateria = (bateria.read()/1024)*100           # nivel de bateria
  for i in range(cont):
    senales = []                                     # lista para almacenar muestras
    y = None
    for j in range(q):
      senales.append(signal.read())                  # generar lista de señales
    data = utils.filtroPromedio(senales, q)          # aplicar filtro promedio
    if (abs(data - dataP) > tolerancia):
      senal.append(data)                             # almacenar dato actual
      count += 1                                     # incrementar cuenta
      if count > 8:
        x = utils.minmax(max(senal))                 # normalizar señal
        xi = [bias, x]                               # vector entrada a la red neuronal
        y = rn.propagation(xi)                       # computar salida de la red neuronal
        count = 0                                    # re-establecer contador
        senal = []                                   # limpiar lista

    # accionar motores (mover articulaciones de la prótesis)
    if y is not None:
      if (utils.rint(y[0][0]) == 1) and (utils.rint(y[0][1]) == -1):
      val1 = hallSensor1.read()                      # sensor Hall mano
      if actualPosA == 100:
        servoA.move(40)
        actualPosA = 40
      else:
        servoA.move(100)
        actualPosA = 100
      # comparar lectura anterior de sensor Hall 1 con lectura actual del mismo
      if (abs(hallSensor1.read() - val1)) <= 15:
        fallo("A")                                   # enviar error al correo

      elif (utils.rint(y[0][0]) == 1) and (utils.rint(y[0][1]) == 1):
        val2 = hallSensor2.read()                    # sensor Hall muneca
        if actualPosB == 170:
          servoB.move(90)
          actualPosB = 90
        else:
          servoB.move(170)
          actualPosB = 170
        if (abs(hallSensor2.read() - val2)) <= 15:
          fallo("B")                                 # enviar información al correo

       elif (utils.rint(y[0][0]) == -1) and (utils.rint(y[0][1]) == 1):
         codo.move()                                 # mover codo

    if tomaTemp >= 35:
      nivelBateria = int((bateria.read()/1024)*100)
      uart.write(str(int(temp))+'-'+str(int(hum))+'-'+str(nivelBateria))
      readTemp() # actualizar valores de temperatura y humedad
      tomaTemp = 0 # reiniciar contador

    time.sleep_ms(20)
```

***Figura 39.*** *Código 1 Función principal para el funcionamiento de la prótesis.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- **Diagrama de flujo y bloques del programa de la prótesis**

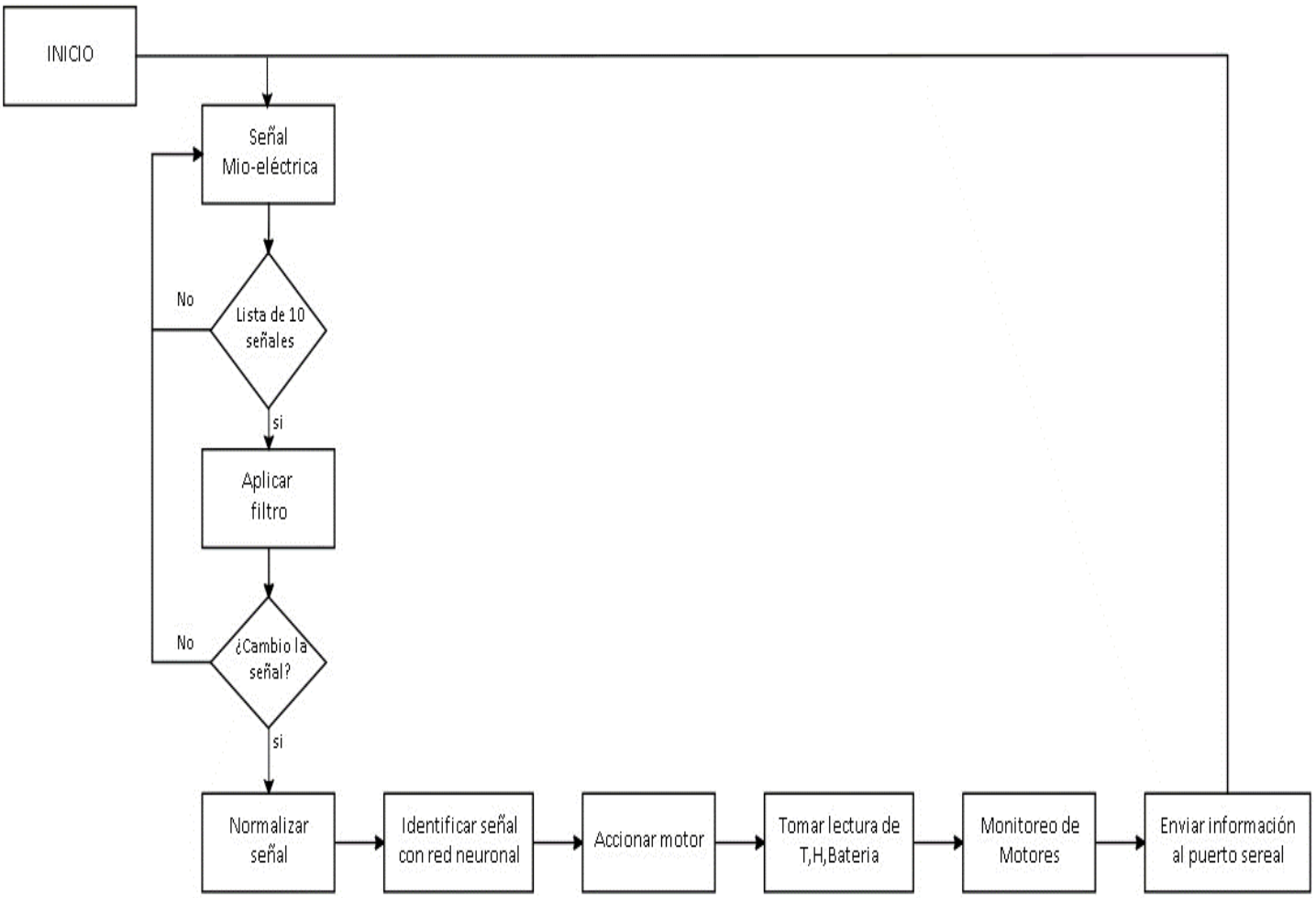



***Figura 40.*** *Diagrama de flujos del programa*

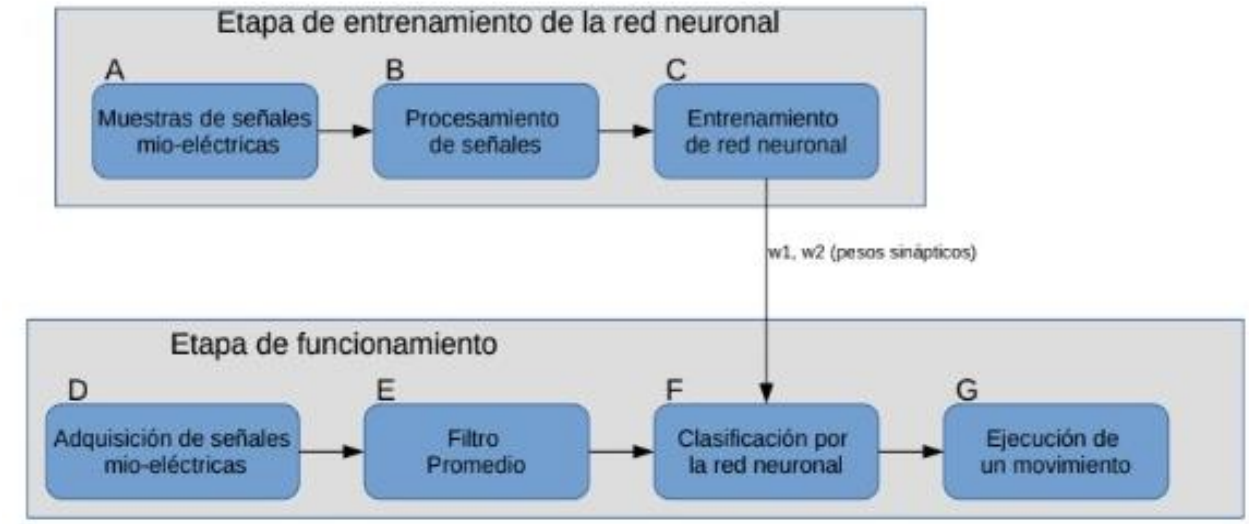



***Figura 41.*** *Diagrama de bloques general del programa*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- **Descripción del diagrama de flujo y bloques**
  - ✓ Encender módulo de control (ESP32, sensores, etc.)
  - ✓ Capturar señal mio-eléctrica, y guardarlo en una lista.
  - ✓ Si ya se han captura 10 señales, pasar al siguiente bloque, si no, regresar al punto 2.
  - ✓ Aplicar filtro promedio a la lista de 10 señales.
  - ✓ Al resultado anterior, se aplica una comparación con la señal captura capturada en el proceso anterior. Si las señales son iguales, entonces se regresa al paso 2, en caso contrario se pasa al siguiente paso.
  - ✓ Normalizar la señal, que consiste en escalar los valores entre -1 y 1, (ver el punto 3.11 del documento). Como valor máximo se utiliza el máximo valor posible a capturar con el ADC, que es 1024, el mínimo es 0.
  - ✓ Consiste en identificar la señal a través de la clasificación de la red neuronal previamente entrenada con datos de prueba.
  - ✓ Con la señal identificada, se acciona el motor correspondiente a la articulación a mover, que puede ser mano, muñeca o codo.
  - ✓ Se toma la temperatura, humedad, y nivel de batería.
  - ✓ Se identifican el movimiento de los motores a través de los sensores de efecto Hall.
  - ✓ La información de los puntos anteriores se envía a través del puerto serial del ESP32, que a su vez se conecta con el módulo Bluetooth acoplado a este.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### *3.7.2. Funciones y clases*

#### 3.7.2.1. Servo

Clase que permite controlar un servomotor (código 2.1). Como argumento, se debe enviar el puerto por el cual se desea enviar la señal de control para el servomotor.

En el constructor, se establece la frecuencia de trabajo, que para un servomotor convencional es de 20 ms. Una vez establecida la frecuencia de trabajo, se inicializa el PWM en el puerto de salida indicado en el argumento de la clase, y se establece la frecuencia calculada.

El método *move*, recibe un valor de tipo entero, entre 0 y 180, que son los grados a los que se puede mover el servomotor. En caso de introducir un valor menor que 0, o mayor que 180, se envía un mensaje en la consola de *MicroPython*.

El método *mapeo* permite mapear el valor de posición indicado, a su respectivo valor de PWM.

```
class Servo():
  def __init__(self, servoPort):
    period = 20*10**-3                              # 20 ms: Periodo de trabajo de
    servomotor estandar
    frequency = round((1) / (period))               # frecuencia de trabajo
    servoPort = Pin(servoPort)                      # puerto de salida
    self.servomotor = PWM(servoPort, freq = frequency)  # objeto pwm

  def move(self, position):
    # si el valor "posición" está en un rango de 0 a 180
    if position >= 0 and position <= 180:
    pwmVal = self.mapeo(position, 0, 180, 0, 99)    # mapear valor de posición a señal PWM
    pwmVal = int(pwmVal + 26)                       # compensar valor PWM
    self.servomotor.duty(pwmVal)                    # escribir valor PWM
    time.sleep(0.01)                                # esperar 10 ms
    # sino, imprimir mensaje
    else:
       print('ingresar un valor entre 0 a 180')

  def mapeo(self, x, inMin, inMax, outMin, outMax):
    val = ((x - inMin) * (outMax - outMin)) / ((inMax - inMin) + outMin)
    return val
```

***Figura 42.*** *Código 2.1 Clase servo*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### 3.7.2.2. Clase PID

Esta clase implementa el controlador PID con sintonización de parámetros Ziegler-Nichols, donde como argumento se colocan los tiempos de respuesta. El constructor, se computan los valores de las constantes del controlador PID, y se establecen los valores de salida mínimos y máximos.

El método *computePID*, computa la salida del controlador, la cual estará acotada entre los valores mínimos y máximos establecidos anteriormente.

```
class PID():
  def __init__(self, t0,t1,t2,y0,y1,u0,u1):
    tao = t1 - t0                    # parámetro tao
    T = (tao/4)-0.1                  # tiempo
    gam = t2 - t1                    # parámetro gamma
    kap = (y1-y0)/(u1-u0)            # parámetro kappa
    kp = (1.2*gam)/(kap*tao)         # constante proporcional
    ki = 2*tao                       # constante integral
    kd = 0.5*tao                     # constante derivativa
    self.a = kp                      # constante auxiliar a
    self.b = (kp*T)/(ki)             # constante auxiliar b
    self.c = (kp*kd)/(T)             # constante auxiliar c
    self.iT0 = 0                     # tiempo anterior
    self.eT0 = 0                     # error anterior
    self.vMax = 1023                 # valor de salida máximo
    self.vMin = 300                  # valor de salida mínimo

  def computePID(self, signalIn, setPoint):
    eT = (setPoint - signalIn)       # error en tiempo T
    iT = self.b * eT * self.iT0      # termino integral
    dT = self.c * (eT-self.eT0)      # termino derivativo
    uT = iT+self.a*eT+dT             # salida PID
    if uT > self.vMax:
      uT = self.vMax
    elif uT < self.vMin:
      uT = self.vMin
    self.iT0 = iT                    # tiempo anterior
    self.eT0 = eT                    # error anterior
    return uT
```

***Figura 43.*** *Código 2.2 Clase PID*

***Tabla 10*** *Ángulo de movimiento para motores*

| Motores | Periodo de trabajo | Grados movimiento | Retardo de trabajo | PWM |
|---|---|---|---|---|
| ***Servomotor 1*** | 20ms | 0-180º | 10ms | Si |
| ***Servomotor2*** | 20ms | 0-90° | - | Si |
| ***Motoreductor*** | 490Hz | 0-180° | - | PID |



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### 3.7.2.3. Clase codo

Esta clase permite controlar el motor DC que se encuentra empotrado en el codo de la prótesis, este mismo es el encargado de efectuar los movimientos de flexión y contracción del codo. Como argumentos se indican los puertos del integrado L298: ENA, IN1, IN2; además, se indica el puerto donde está conectado el potenciómetro. En el constructor, se configuran los puertos indicados.

El método *move*, lleva a cabo el movimiento del motor, y para esto emplea el controlador PID, donde el potenciómetro es empleado para la retroalimentación.

```
class Codo():
  def __init__(self, ENA, IN1, IN2, POT):
    enable = Pin(ENA)                          # puerto de senal de salida pwm
    motorFreq = 490                            # frecuencia de trabajo del motor dc
    self.motor = PWM(enable, motorFreq)        # objeto PWM
    self.in1 = Pin(IN1, Pin.OUT)               # puerto de salida 1
    self.in2 = Pin(IN2, Pin.OUT)               # puerto de salida 2
    potPort = Pin(POT)                         # puerto del potenciometro
    self.pot = ADC(potPort)                    # objeto ADC
    self.pot.width(ADC.WIDTH_10BIT)            # resolucion de 10 bits para lecturas de
    voltaje
    self.pot.atten(ADC.ATTN_11DB)              # 11 dB de atenuacion (0 - 3.3v)
    self.actualPos = 1                         # posición actual del codo (1: arriba, 0:
    abajo)
    self.setPoint = 50                         # posición a alcanzar
    self.pid = None                            # incializar controlador PID vacío
    self.error = 5                             # error para alcanzar setpoint

  def initPid(self):
    t0 = 0.1                                    # tiempo de arranque inicial
    t1 = 0.2                                    # tiempo de inicio de pendiente
    t2 = 1                                      # tiempo de maximo impulso
    y0 = 10                                     # ángulo mínimo a medir
    y1 = 100                                    # ángulo máximo a medir
    u0 = 350                                    # señal pwm mínima
    u1 = 1023                                   # señal pwm máxima
    self.pid = PID(t0, t1, t2, y0, y1, u0, u1) # inicializar controlador PID

  def activarMotor(self, outPid):
    self.motor.duty(round(outPid))              # activar ENABLE
    if self.actualPos == 1:
      self.in1.value(0)
      self.in2.value(1)
    elif self.actualPos == 0:
      self.in1.value(1)
      self.in2.value(0)

  def move(self):
    potVal = self.pot.read()
    pos = round(mapping(potVal, 0, 1023, 0, 180))
    # identificar a que posición a moverser
    if pos <= 35:
      self.setPoint = 40                        # setpoint para bajar brazo
      self.actualPos = 1                        # el brazo se encuentra contraido (arriba)
    elif pos > 35:
      self.setPoint = 25                        # setpoint para subir brazo
      self.actualPos = 0                        # el brazo se encuentra flexionado (abajo)
    # iniciar controlador PID
    self.initPid()                              # incializa controlador PID
    for i in range(300):
      potVal = self.pot.read()                  # obtener voltaje del potenciometro
      pos = round(mapping(potVal, 0, 1023, 0, 180))     # convertir a posición angular
      outPid = self.pid.computePID(pos, self.setPoint)  # computar PID
      self.activarMotor(outPid)                 # activar motor
      if (abs(self.setPoint - pos) <= self.error):
        self.apagarMotor()                      # desactivar motor
        break
    time.sleep(0.05)
    self.apagarMotor()
```

***Figura 44.*** *Código 2.3 Clase codo*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

### 3.7.2.4. Clase perceptrón multicapa (mlp)

Esta clase implementa la red neuronal perceptrón multicapa, en su fase de operación. Como argumentos se indica la cantidad de neuronas en la capa de entrada, la capa oculta y de salida. En el constructor, se deben colocar los pesos sinápticos W1 y W2, y se inicializan los vectores para almacenar las salidas de las neuronas de las capas oculta y de salida.

El método *propagación*, computa la salida de la red neuronal mediante la propagación de la red neuronal.

```
class mlp():
  def __init__ (self, inputNeurons, hiddenNeurons, outNeurons):
    # matriz de pesos sinápticos w1
    self.w1 = [[  0.86634403,  11.07931215],
               [ -3.9053337 ,   5.79367578],
               [  1.11753939,   7.34316316],
               [-17.14645112,  19.08804432],
               [  2.61250261,   2.90262371]]
    # matriz de pesos sinápticos w2
    self.w2 = [[-0.9065782,-4.36537641,-6.30985764, -4.08025025,-14.9470907,  -2.00108773],
               [-1.4958991,1.54169226,0.80931667, 0.49301797,2.63300427, 0.24900168]]
    """ Generar vectores de salidas de la capa oculta y salida de la red """
    # salidas en capas de la red (oculta y salida)
    self.outHiddenLayer = [[0.0] * (hiddenNeurons + 1)]
    self.outNN = [[0.0] * outNeurons]
    # topología de red
    self.inputNeurons = inputNeurons
    self.hiddenNeurons = hiddenNeurons
    self.outNeurons = outNeurons

  def propagation(self, xi):
    # obtener salida de las neuronas de la capa oculta
    for i in range(self.hiddenNeurons):
      self.outHiddenLayer[0][i+1] = (dot(xi,self.w1[i]))
      # asignar valor de umbral
      self.outHiddenLayer[0][0] = -1
    # obtener salida de la red
    for i in range(self.outNeurons):
     self.outNN[0][i] = utils.rint(dot(self.outHiddenLayer[0], self.w2[i]))
    return self.outNN
```

***Figura 45.*** *Código 2.4 Clase para la red neuronal perceptrón multicapa*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

### 3.8. Medición de señales mio-eléctricas

Una vez implementadas las diferentes etapas del circuito de adquisición de señales mio-eléctricas se visualizaron por medio de un osciloscopio las señales obtenidas de cada etapa del circuito. Para esto se colocan los electrodos de monitoreo 3M en personas con amputación o nivel del brazo que será 1/3 arriba del codo aproximadamente como lo muestra la figura 46, donde el electrodo que se encuentra en la muñeca corresponde a la señal de referencia mientras que los dos electrodos que se encuentran en la parte de humero son los que se utilizan para captar la señal mio-eléctrica.

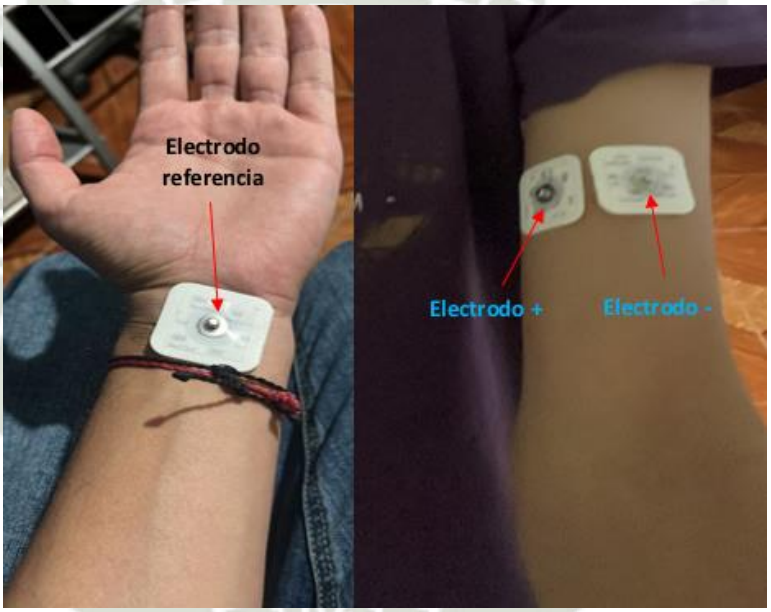



***Figura 46.*** *Ubicación de electrodos.*

La figura 47, muestra el circuito de adquisición de señales mio-eléctricas implementado en la tablilla de experimentos o protoboard, donde se pueden apreciar las 4 etapas principales: Amplificación (1), filtrado (2), amplificación variable (3) y rectificación (4).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

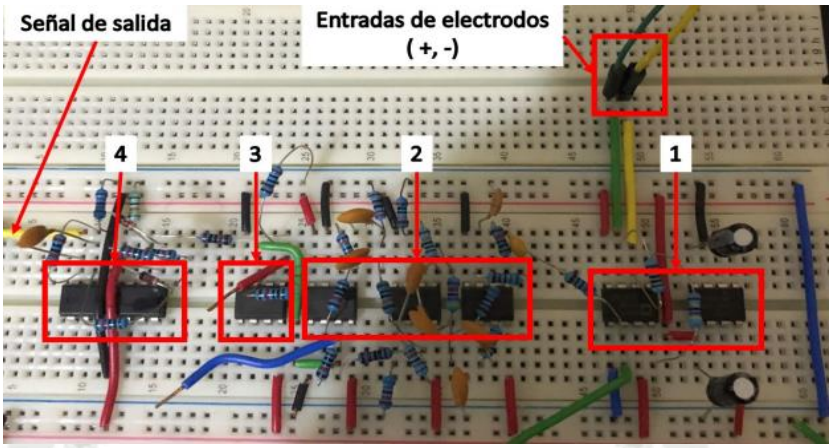



***Figura 47.*** *Implementación del circuito de adquisición de señales mio-eléctricas.*

A continuación, se presentan las capturas de pantalla de las señales.

### *3.8.1. Amplificación*

En esta etapa se capturó la señal proveniente de la salida del amplificador de instrumentación (etapa A del circuito 30). En la figura 48, se presentan las señales capturadas, donde la imagen del inciso (a) se muestra la señal obtenida cuando el músculo está en reposo (sin esfuerzo); la imagen del inciso (b) muestra la señal obtenida al momento de generar una contracción en el músculo; la imagen del inciso (c) muestra la señal obtenida al momento de generar tres contracciones seguidas.

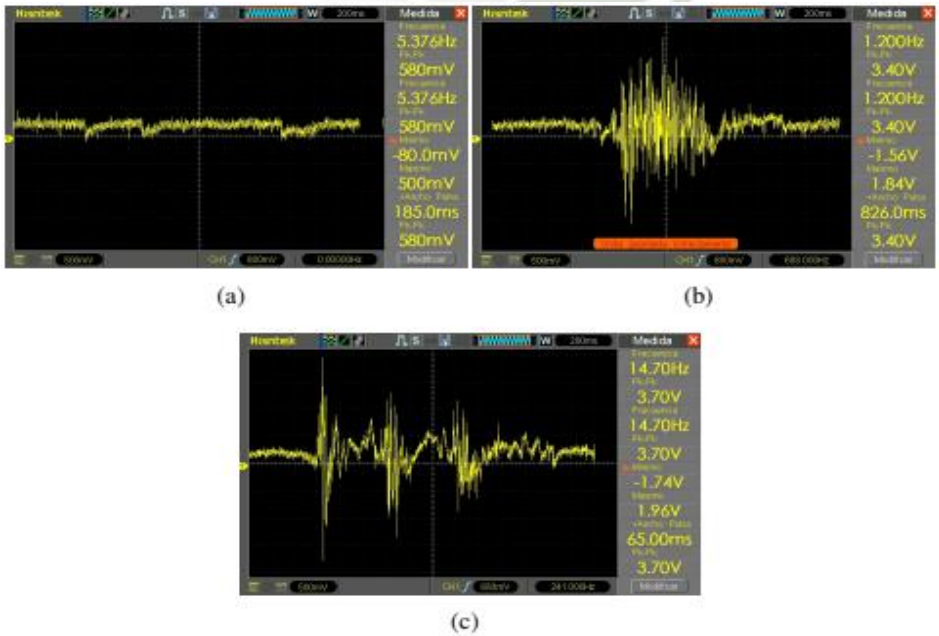



***Figura 48.*** *Señal amplificada por el AD620 (etapa A).*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### *3.8.2. Filtrado*

Posterior a obtener la señal amplificada por el amplificador de instrumentación, esta señal es filtrada para eliminar el ruido obteniendo así las señales más significativas. En la figura 49, se muestra la señal obtenida por la etapa de filtrado, donde la imagen del inciso (a) muestra la señal capturada cuando el músculo se encuentra en reposo, por otro lado, la imagen del inciso (b) presenta la señal obtenida cuando se ejerce una contracción en el músculo, donde se puede apreciar que la señal disminuye el ruido generado, además de esto la amplitud de la señal se ve disminuida con respecto a la señal de entrada (etapa anterior), esto es debido a los filtros implementados; en la imagen del inciso (c) se muestra la señal obtenida al momento de ejecutar tres contracciones seguidas.

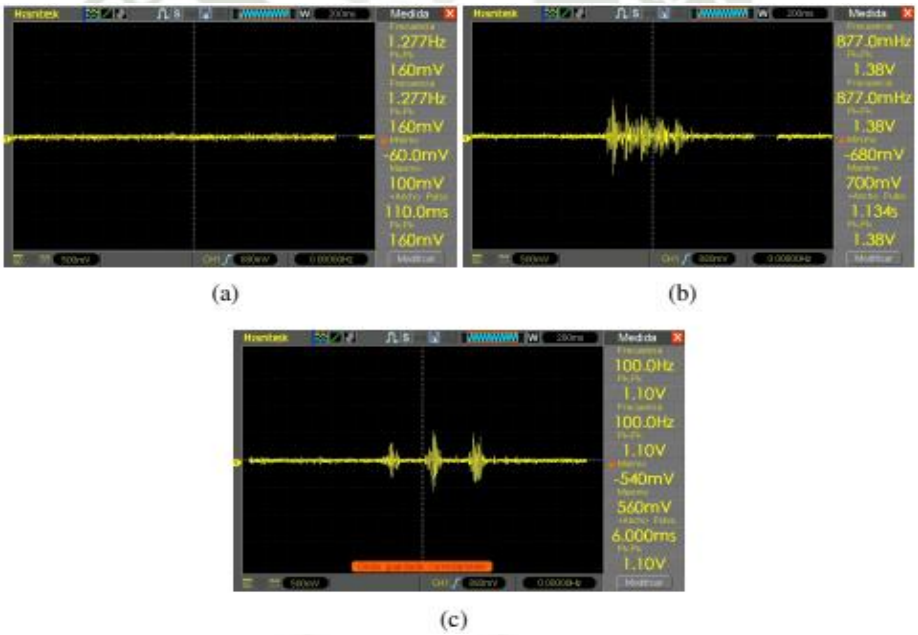



***Figura 49.*** *Señal obtenida por la operación de filtrado*

### *3.8.3. Amplificación variable*

Debido a que la etapa de filtrado disminuye la amplitud de la señal mio-eléctrica se implementó un amplificador, el cual se diseñó en modo variable para poder ajustar la amplitud de acuerdo con la señal obtenida de cada usuario. En la figura 50, se muestran las señales obtenidas en esta etapa, cabe señalar que, en estos resultados presentados, la resistencia de ganancia se cambió a 10 KΩ para obtener una ganancia de seis. La imagen del



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

inciso (a) muestra la señal obtenida al momento de ejercer una contracción en el músculo y la imagen del inciso (b) presenta la señal obtenida al momento de ejecutar tres contracciones.

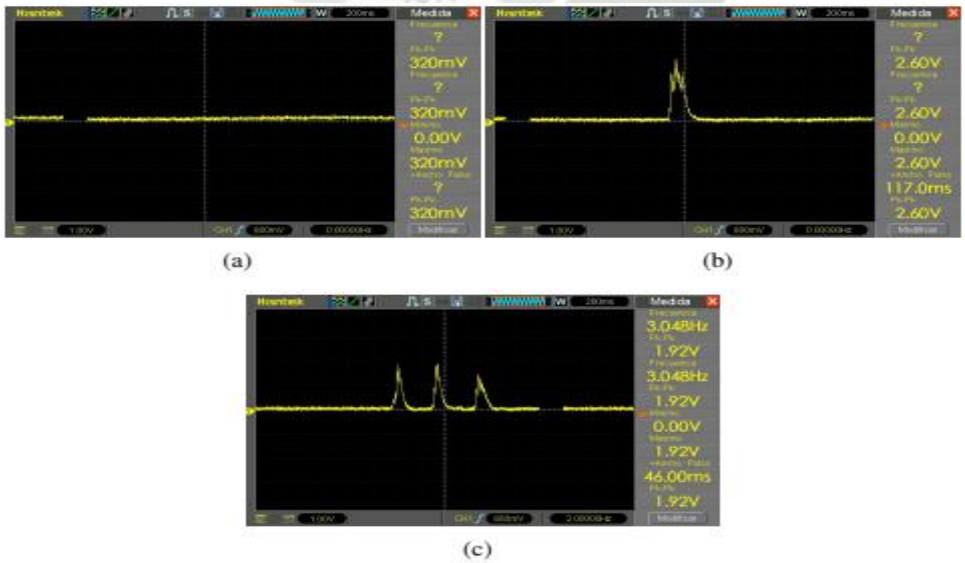



(a) (b)

***Figura 50.*** *Señal obtenida por la etapa de amplificación variable.*

### *3.8.4. Rectificación*

La última etapa del circuito de adquisición de señales mio-eléctricas consiste en la rectificación de la señal proveniente de las etapas anteriores. En la figura 51 se muestra la señal rectificada, donde la imagen del inciso (a) muestra la señal obtenida cuando el músculo está en reposo; la imagen del inciso (b) presenta la señal capturada al momento de ejercer tensión sobre el músculo; la imagen del inciso (c) muestra la señal obtenida al momento de ejecutar tres contracciones en el músculo.




(a) (b)

(c)

***Figura 51.*** *Señal de salida.*

Esta señal es enviada hacia el ADC del microcontrolador para su adquisición y posterior reconocimiento.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### 3.9. Implementación del circuito en PCB

Una vez concretado el circuito de adquisición de señales mio-eléctricas en la tarjeta de experimentos (protoboard), y con las respectivas pruebas efectuadas, se continuó con su implementación en una placa perforada.

En la figura 52, se presenta el circuito implementado en una placa perforada, donde los componentes principales de este son etiquetados. Los cuales son:

- Conector para entradas de los electrodos.
- Conector para la entrada de alimentación.
- Fuente simétrica (7660).
- Etapa de amplificación de la señal de entrada (AD620).
- Filtro pasa bajas y pasa altas (TL084).
- Filtro Notch (TL084).
- Rectificador (TL082).
- Señal de salida.

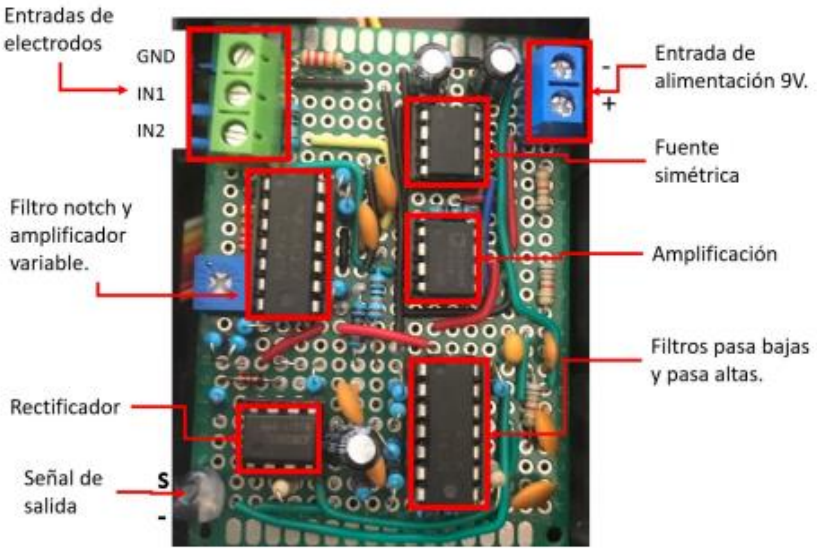



***Figura 52.*** *Señal de salida.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## 3.10. Aprendizaje de la red neuronal

Se presenta la etapa de aprendizaje de la red neuronal, donde se inicialmente se llevó a cabo la adquisición y almacenamiento de los datos de las señales mio-eléctricas, posteriormente se realizó la etapa de acondicionamiento de los datos, y finalmente el aprendizaje de la red neuronal. El modelo de red neuronal utilizado es el perceptrón multicapa. A continuación, se presentan las etapas para el aprendizaje de la red neuronal.

### *3.10.1. Adquisición de datos*

La primera etapa consistió en recibir las señales mio-eléctricas previamente filtradas y amplificadas, concretamente la señal de salida del filtro rectificador (etapa **F** del circuito de adquisición de señales mio-eléctricas). Esta señal fue enviada al puerto *ADC* de la tarjeta ESP32, donde se habilitó el puerto **GPIO 34** que pertenece al primer bloque *ADC* de la tarjeta.

El ESP32 contiene de manera interna 2 bloques de *ADCs*, donde el primer bloque corresponde a los puertos 33 al 39, mientras que el segundo bloque usa los puertos 12 al 15 y 25 a 27. El firmware de MicroPython solo puede usar el primer bloque de *ADC* de la tarjeta ESP32, por lo que para esta etapa se utilizó un puerto que se encuentra disponible en el primer *ADC*. Para esto, se configuró el *ADC* del GPIO 35 a 10 BITS para acotar la señal analógica en valores entre 0 y 1023.

En la figura 53, se muestra el puerto GPIO 34 por el cual se recibe la señal proveniente de la etapa de acondicionamiento de la señal mio-eléctrica y que está representado por la etiqueta *SM* del acrónimo *señal mio-eléctrica*. La tierra del circuito de adquisición de señales mio-eléctricas se puenteó a la tierra de la tarjeta de adquisición de datos ESP32. Por los puertos *GPIO 16* y *GPIO 17* se envían los valores de las señales mio-eléctricas adquiridas, que corresponden al segundo puerto UART (RX2, TX2).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

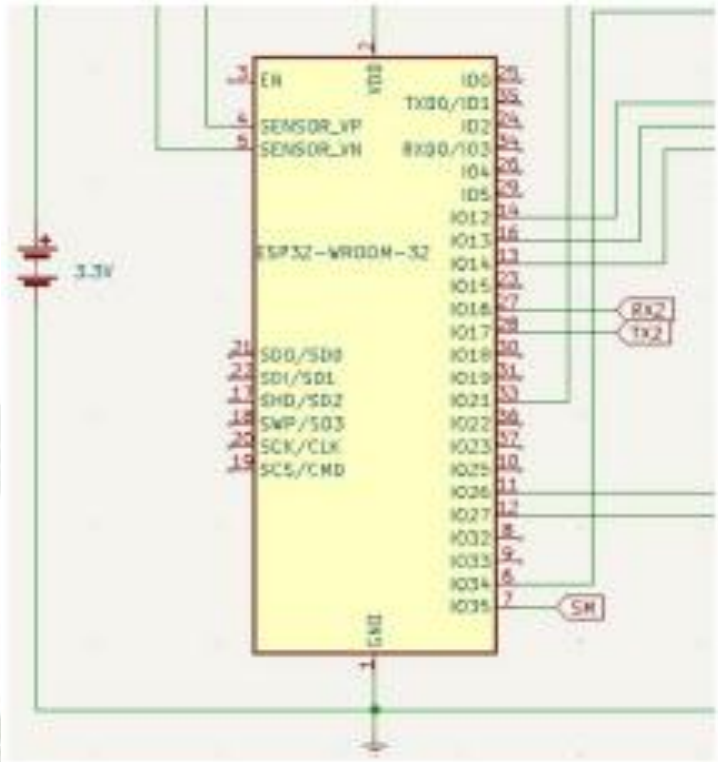



***Figura 53.*** *Diagrama eléctrico de la conexión de la señal mio-eléctrica en el ESP32.*

Para la etapa de adquisición de datos se creó un programa en MicroPython para leer las señales mio-eléctricas con el ESP32, aplicar el filtro de la media y enviarlas vía comunicación serial a un PC. En la ecuación 46, se muestra la expresión usada para aplicar el filtro promedio a una señal, donde $x_f$ es la señal filtrada; $j$ es la cantidad de muestras a tomar en cuenta, se recomienda que sea impar; $x$ es la señal de entrada; mínima (*min*) y máxima (*máx.*) indica que los datos en la señal $x$ son ordenados de menor a mayor.

- **Ecuación 46** $$x_f = x\left\{x_1^{min}, x_2, x_3, \dots, x_j^{max}\right\} \rightarrow x\,(j + 1/2) \qquad (66)$$

La figura 54, muestra la señal mio-eléctrica obtenida $x$ (color azul) y la señal resultante de la aplicación del filtro de la media $x_f$ (color naranja), al usar un valor de $j$ igual a 10, esta señal es obtenida cuando el músculo está en reposo



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

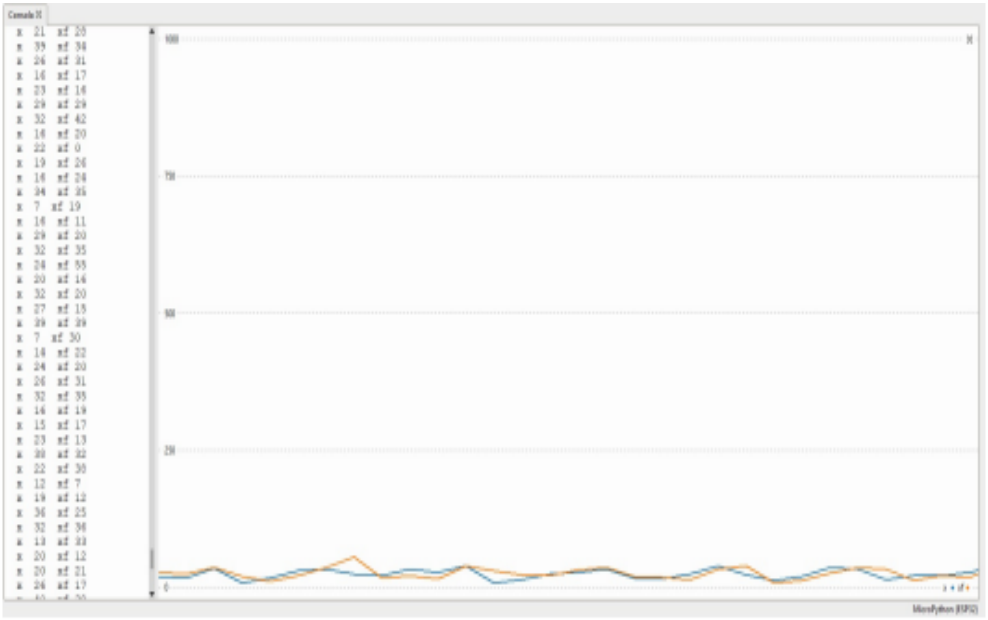


***Figura 54.*** *Señales mio-eléctricas adquiridas por el ADC del ESP32.*

***Tabla 11*** *Rangos de adquisición de datos*

| | Potencial ADC 0-1024 | Movimiento | Datos almacenamiento | Señal | Movimiento red neuronal |
|---|---|---|---|---|---|
| ***Mano*** | 50 a 65 | Abrir y cerrar | 2000 | Figura 56 (a) | [1,1] |
| ***Muñeca*** | 70 a 110 | Pronación y Supinación | 2000 | Figura 56 (b) | [1,-1] |
| ***Codo*** | 150 a 300 | Flexión y contracción | 2000 | Figura 56 (c) | [-1,1] |

En la figura 55, muestra la señal mio-eléctrica obtenida al momento de efectuar una contracción en el músculo. Las señales recibidas son mapeadas por el ADC a valores entre 0 y 1024.

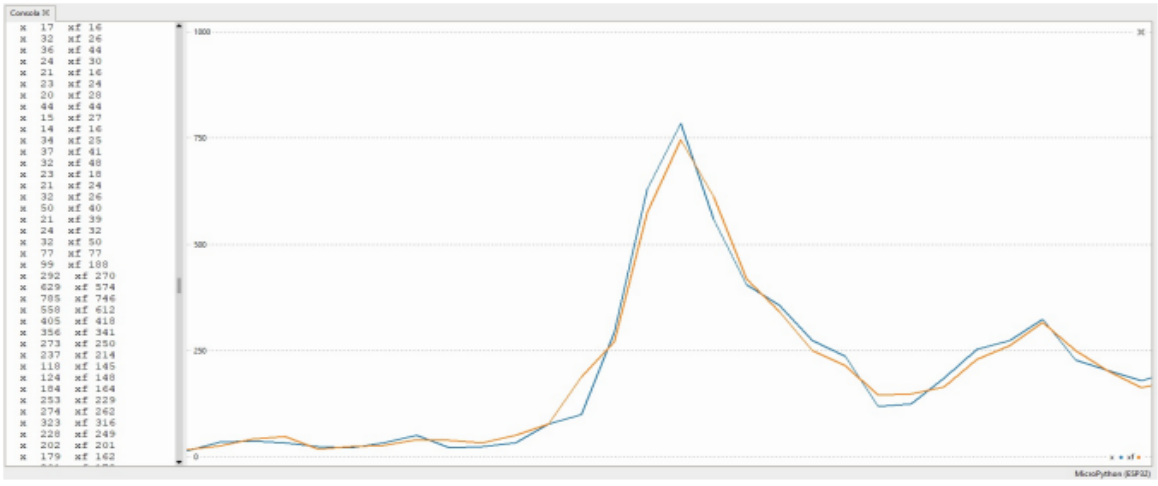


***Figura 55.*** *Señales mio-eléctricas adquiridas por ADC del ESP32.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Antes de pasar a la etapa de grabación de las señales obtenidas del cuerpo, es importante identificar patrones de potencial que pueden ser obtenidos por la ejecución de ciertos movimientos. Para esto, se identificaron los potenciales:

- 50 a 65: generados al efectuar el movimiento de abrir y cerrar la mano.
- 70 a 110: generados al mover el dedo anular.
- 150 a 300: generados al flexionar y contraer la mano.

Para recibir y almacenar los datos provenientes del puerto serial, se creó un programa en Python 3, empleando las bibliotecas: Serial, para leer los datos del puerto serial; Pandas, para almacenar los datos en un archivo de Excel, donde estableció un total de 2000 datos para almacenar.

Con lo anterior, se generaron 4 archivos Excel, con los nombres de las señales capturadas: codo, muñeca, mano y reposo, que contienen los valores de las señales mio-eléctricas que permitirán mover el codo, muñeca, mano y también la señal generada cuando los músculos están en reposo con una ligera tensión.

En la figura 56, se muestran las señales adquiridas en este proceso, donde la gráfica del inciso (a) corresponde a la señal del codo; la gráfica del inciso (b) corresponde a la muñeca; la gráfica del inciso (c) corresponde a la señal de la mano, y la gráfica del inciso (d) corresponde a la señal en reposo. Se puede apreciar que cada señal tiene una amplitud diferente, donde la señal del codo presenta una mayor amplitud (cerca de 250) mientras que la señal en reposo presenta la menor amplitud (entre 20 y 10).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

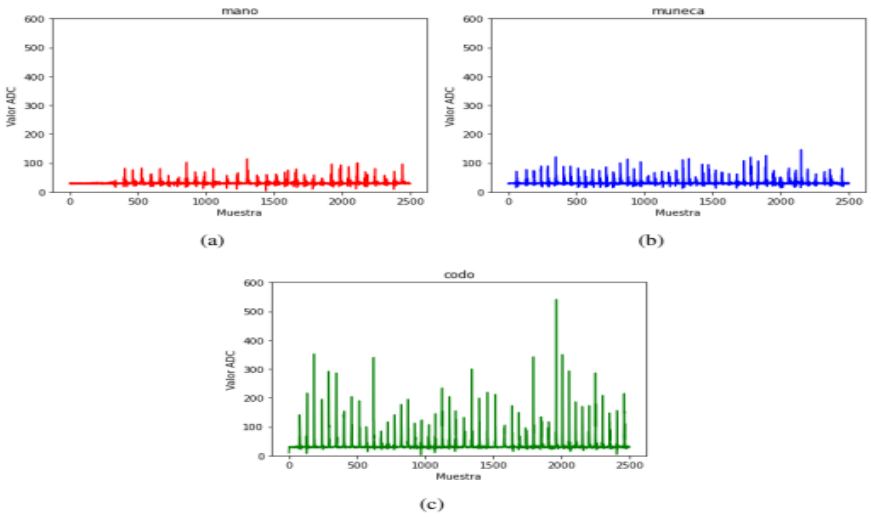



***Figura 56.*** *Señales mio-eléctricas generadas para mover las articulaciones de la prótesis*

### 3.11. Procesamiento de datos

Como segundo paso, se llevó a cabo el acondicionamiento de los datos recabados con el objetivo de obtener los valores más significativos. Para esto, se realizó un análisis estadístico de los datos, a través de una tabla de distribución de frecuencias con cinco intervalos.

En la tabla 11, se muestra la tabla de distribución de frecuencias obtenidas correspondientes a los movimientos de abrir y cerrar la mano. De acuerdo con estos valores, se despreciaron los intervalos con las frecuencias más altas, ya que estas representan el ruido capturado, por lo que, se tomaron en cuenta los datos con frecuencia mayor al de los intervalos no considerados. Estas señales serán empleadas para mover los dedos de la prótesis.

***Tabla 12*** *Tabla de distribución de frecuencias obtenidas de las señales para mover la mano.*

| Intervalo | *fi* |
|---|---|
| [3,25] | 137 |
| [25,45] | 2251 |
| [47,69] | 81 |
| [69,91] | 24 |
| [91,113] | 6 |



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

La tabla 12, muestra la distribución de frecuencias obtenida a partir de las señales obtenidas al flexionar y contraer la mano, donde se consideraron los valores que se encuentran en los intervalos de dos y tres, ya que estos corresponden al patrón observado previamente. Estas señales serán empleadas para accionar el motor del codo de la prótesis.

***Tabla 13*** *Tabla de distribución de frecuencias obtenida de las señales para mover la mano.*

| Intervalo | *fi* |
|---|---|
| [2,110] | 2395 |
| [110,218] | 92 |
| [218,326] | 8 |
| [326,434] | 4 |
| [434,542] | 1 |

Con lo anterior, se obtuvieron los valores que mejor representan las señales producidas con los movimientos mencionados. En la figura 57, se muestran las señales obtenidas al aplicar este paso.

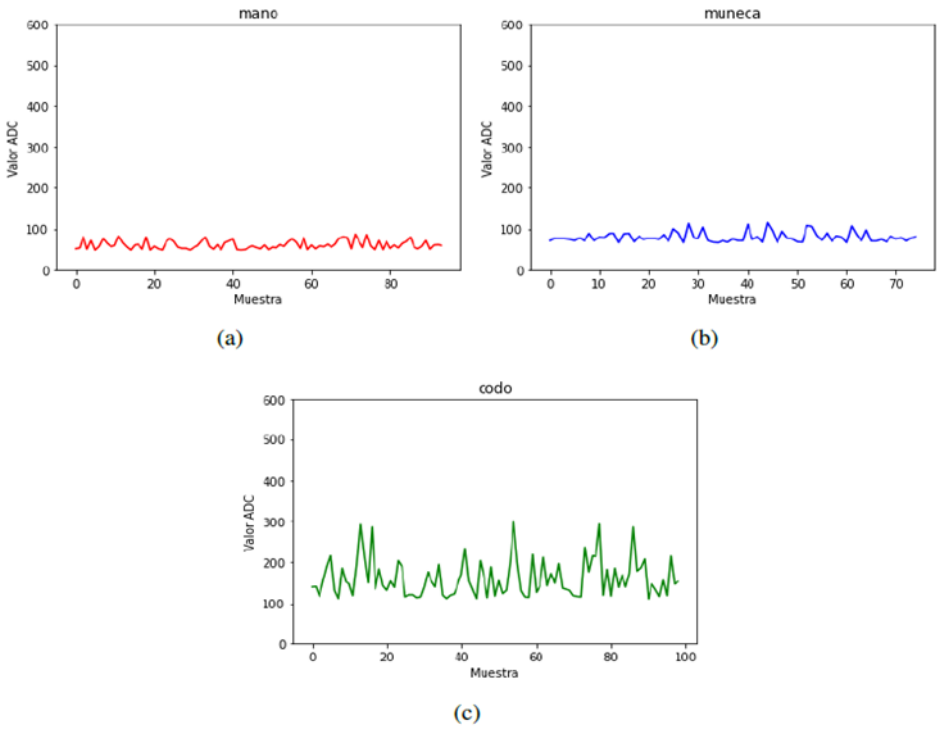



***Figura 57.*** *Señales mio-eléctricas obtenidas de la etapa de procesamiento*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

En la figura 58, se presenta la comparación de las señales capturadas para efectuar los movimientos de la mano, muñeca y codo de la prótesis.

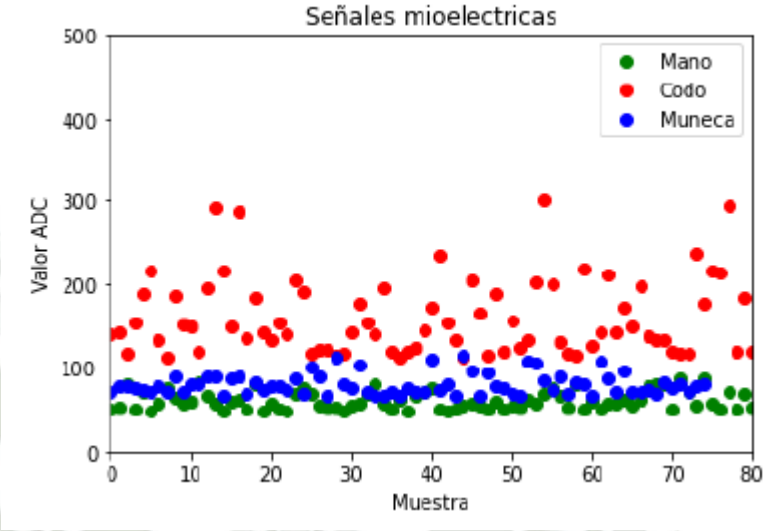



***Figura 58.*** *Comparación de señales mio-eléctricas*

### 3.12. Aprendizaje

En esta etapa se llevó a cabo el entrenamiento de la red neuronal con los datos previamente procesados. Inicialmente, se normalizaron los datos de las señales a valores entre -1 y 1 usando la ecuación 47.

- **Ecuación 47** $$x = 2\left(\frac{x - x_{max}}{x_{max} - x_{min}}\right) - 1 \qquad (67)$$

Donde, $x$ es la señal normalizada; $x_{max}$ es el valor máximo obtenido entre las señales; $x_{min}$ es el valor mínimo entre las señales; debido a que los valores que nos puede entregar el ADC están acotados entre 0 y 1024, estos se asignaron a $x_{min}$ y $x_{max}$ respectivamente. Las señales resultantes se muestran en la gráfica de la figura 59.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

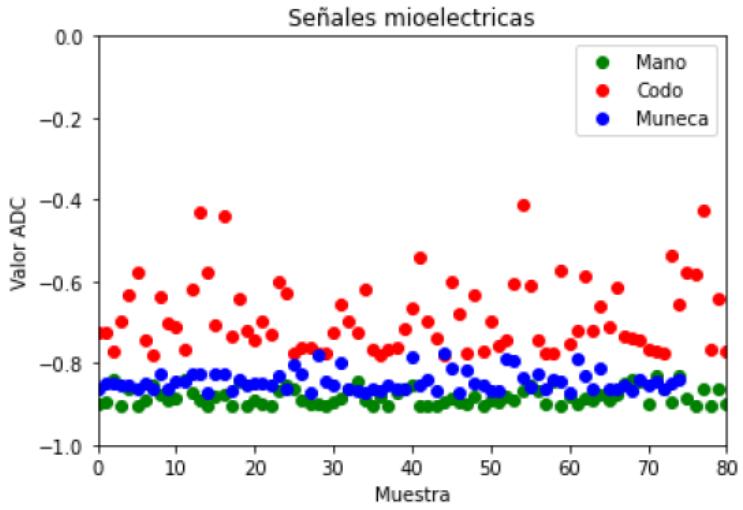



***Figura 59.*** *Señales normalizadas a valores -1 y 1.*

### *3.12.1. Parámetros de la red neuronal*

Antes de llevar a cabo el aprendizaje con la red neuronal perceptrón multicapa se definieron sus parámetros, los cuales se presentan a continuación:

- **Factor de aprendizaje**: 0.001. Que determina el incremento o decremento que se da al valor de los pesos sinápticos durante el aprendizaje.
- **Épocas: 12000**. Indica la cantidad de interacciones en las que se llevará a cabo el aprendizaje.
- **Entradas: 1**. Es la cantidad de señales de entrada a la red neuronal, para el caso de este trabajo solo se tiene una entrada.
- **Salidas: 2**. Es la cantidad de salidas que pueden obtenerse de la red neuronal, depende del tipo de problema a resolver, para este caso se usaron 2 salidas porque se desean clasificar 4 tipos de señales las cuales están etiquetadas como:
  - $[-1,1]$: Señal de movimiento del codo. Activar motor del codo.
  - $[1,-1]$: Señal de movimiento de muñeca. Activar motor de la mano.
  - [1,1]: Señal de movimiento de la mano. Activar motor de la muñeca.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- **Neuronas en capa oculta: 5**. Es la cantidad de neuronas dentro de la capa oculta, es

arbitrario y depende del problema a resolver por la red neuronal, para el caso de este trabajo se llevó a cabo una serie de pruebas buscando un número de neuronas mínimo, ya que a mayor cantidad de neuronas mayor tiempo de procesamiento.

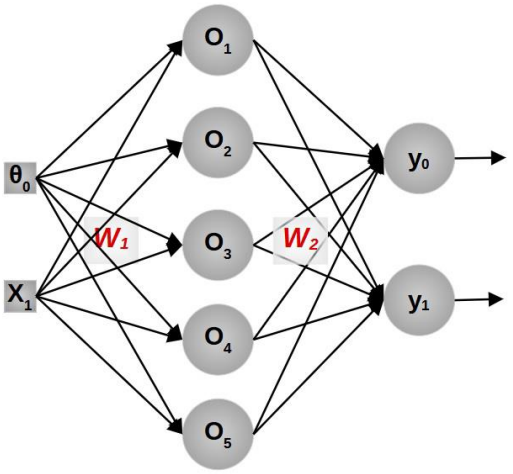



***Figura 60.*** *Topología de red neuronal empleada*

### 3.12.2. *Etapa de aprendizaje*

Con lo anterior establecido se ejecutó el algoritmo de aprendizaje de la red perceptrón multicapa, donde se obtuvo el decremento del error que se presenta en la figura 61.

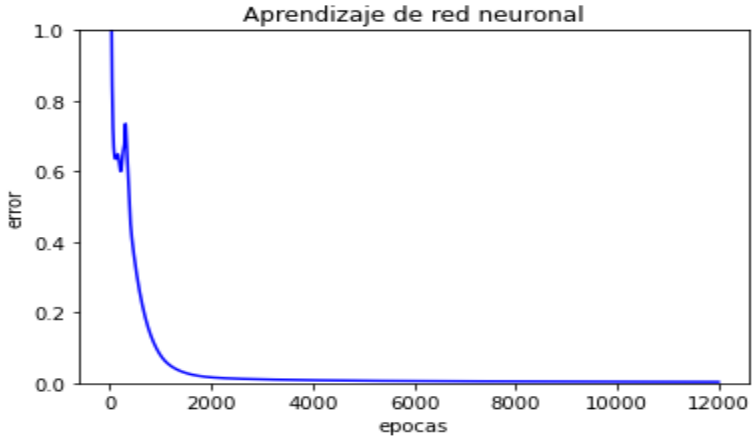



***Figura 61.*** *Error obtenido durante la etapa de aprendizaje.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

El tiempo de entrenamiento duro aproximadamente 45 minutos, donde se emplearon 268 muestras, y la efectividad obtenida fue de 0.7; que se calculó usando la ecuación 48, donde *vp* representa la cantidad de aciertos obtenidos por la red neuronal (verdaderos positivos); *m* es el total de muestras. Por lo que, la efectividad está acotada entre 0 y 1, donde 1 corresponde a un 100% de efectividad y 0 a una nula efectividad. Durante el aprendizaje de la red neuronal, se obtuvieron 189 verdaderos positivos (*vp*).

- **Ecuación 48** $$efectividad = \frac{vp}{m} \rightarrow \frac{189}{79} = 0.705 \quad (68)$$

Una vez llevado a cabo el aprendizaje de la red neuronal, se obtuvieron los valores de los pesos sinápticos $W_1$ y $W_2$, y se registraron en el programa de *MicroPython* para ejecutar el funcionamiento de la red neuronal en el ESP32.

## 3.13. Circuito de control

En este capítulo se presenta el circuito de control de la prótesis, el cual está compuesto por una tarjeta ESP32 como principal unidad de procesamiento, 2 servomotores para dar movimiento a la mano y muñeca, un motor DC para mover el mecanismo del codo, así como un integrado L298 para el control de este motor. Como sensores, se tiene un DHT22 para monitorear la temperatura interna de la unidad de control, un potenciómetro para medir la posición angular del codo y dos sensores de efecto Hall.

Además de los componentes mencionados, se incorporó un módulo Bluetooth HC06, para establecer comunicación serial con una aplicación en un dispositivo Android, en el cual se podrá monitorear la temperatura de la prótesis, así como el nivel de batería. A continuación, se describe el circuito de control desarrollado.

### *3.13.1. Circuito general*

En la figura 62, se presenta el circuito de control de la prótesis, donde se muestran los diferentes elementos que lo componen, las conexiones están representadas por etiquetas, que



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

se describen a continuación:

- **SM**: Señal mio-eléctrica, proveniente del circuito de adquisición de señales.
- **RX2, TX2**: Puertos para comunicación serial 2, donde se conecta el módulo Bluetooth.
- **Servo A**: Señal PWM para el servomotor A.
- **Servo B**: Señal PWM para el servomotor B.
- **ENA**: Señal enable A, para el integrado L298.
- **IN1**: Señal IN1, para el integrado L298.
- **IN2**: Señal IN2, para el integrado L298.
- **BAT**: Señal para el nivel de batería.
- **POT**: Señal del potenciómetro, para medir posición angular del codo.
- **DHT**: Señal del sensor DHT22.
- **SH1**: Señal del sensor de efecto Hall 1.
- **SH2**: Señal del sensor de efecto Hall 2.

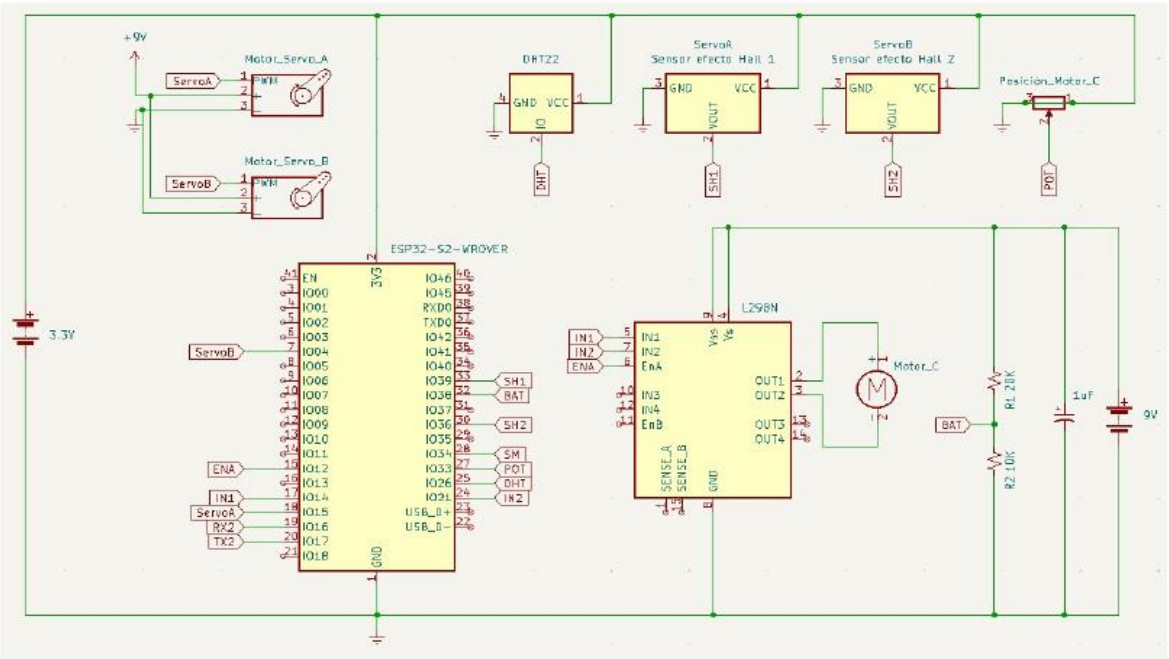



***Figura 62.*** *Circuito de control de la prótesis.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### *3.13.2. Control de motor CD*

Para poder controlar el motor de corriente continua que se presenta en la figura 106, y que está representado en el circuito de control (figura 62) como "motor C", se empleó el controlador PID.

De acuerdo a las especificaciones del motor, y la batería empleada para su alimentación, se calcularon los valores del algoritmo 2.2.1, necesarios para determinar el valor de $K_P$, $K_I$ y $K_D$ mediante el método de sintonización Ziegler-Nichols.

Los valores de tiempo de respuesta del motor al aplicarle una señal PWM entre 350 y 1023 ($u$) a la entrada, se obtuvieron mediante observación, que corresponden a $t_0 = 0{,}1$, $t_1 = 0{,}2$ y $t_2 = 1$, los cuales son, el tiempo de arranque inicial, tiempo de inicio de pendiente (cuando comienza el motor a incrementar la posición angular), y tiempo de máximo impulso (cuando el motor alcanza la posición máxima establecida) respectivamente. La figura 63, muestra los tiempos de respuesta del motor obtenidos a través de observación al aplicarles una entrada PWM de entre 350 y 1023.

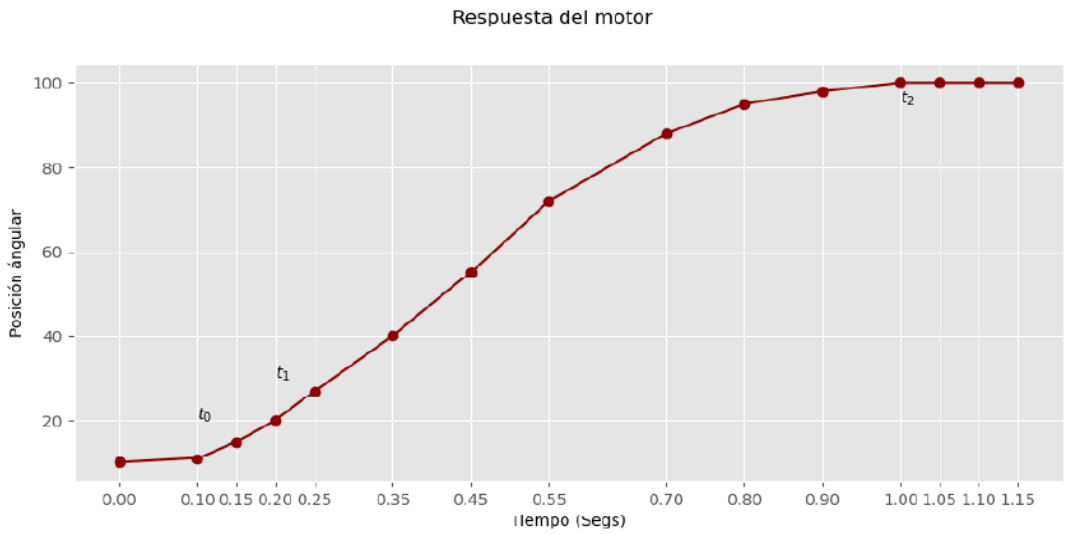



***Figura 63.*** *Grafica de respuesta de motor obtenida en base a observación (el tiempo es aproximado).*

Así mismo, los valores ángulo de trabajo $y_0$ y $y_1$, son obtenidos por medio del potenciómetro empotrado sobre el eje del motor, donde se establecieron a 10 y 100 grados



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

respectivamente. Las unidades de impulso son: u0 = 350 (valor PWM con el cual el motor comienza a arrancar) y u1 = 1023 (valor PWM con el cual el motor trabaja a velocidad máxima, dependiendo de su fuente de alimentación).

Con los valores de tiempo obtenidos, se pueden calcular $\tau_0$, $\gamma_0$ y $\kappa_0$, de la siguiente forma:

$$t_o = 0{,}2 - 0{,}1 = 0{,}1$$

$$\gamma_0 = 1 - 0{,}2 = 0{,}8$$

$$k_0 = \frac{100 - 10}{1023 - 350} = 0{,}133$$

Con lo anterior es posible obtener las constantes proporcionales $K_p$, integra $T_i$ y derivativo $T_d$:

- **Ecuación 49** $K_p = \frac{1{,}2 * \gamma_0}{k_0 * t_0} = \frac{1{,}2 * 0{,}8}{0{,}133 * 0{,}1} = 72{,}1$ (69)

$$T_i = 2 * t_0 = 2 * 0{,}1 = 0{,}2$$

$$T_d = 0{,}5 * t_0 = 0{,}5 * 0{,}1 = 0{,}05$$

Por último, se obtiene el valor de los parámetros del controlador PID:

- **Ecuación 50**

$$a = K_p = 72{,}1 \quad (70)$$

$$b = \frac{K_p * T}{T_i} = \frac{72{,}1 * 0{,}025}{0{,}2}$$

$$c = \frac{K_p * T_d}{T} = \frac{72{,}1 - 0{,}2}{0{,}025}$$

Donde $T = (\tau 0/4)$, considerado como el tiempo de muestreo en segundos.

### *3.13.3. Nivel de la batería*

Para medir el nivel de batería se diseñó un divisor de voltaje para disminuir la diferencia de potencial, esto debido a que la batería a la cual se desea medir su nivel de voltaje es de un valor superior al soportado por los ADC´s del módulo ESP32.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

La ecuación 36, presenta la relación utilizada para crear el divisor de voltaje.

➢ **Ecuación 51** $$V_{out} = \frac{V_{in} * R_2}{R_1 + R_2} \quad (71)$$

Sustituyendo el valor de $V_{out}$ = 3,3V y $V_{in}$ = 9V, y proponiendo el valor de $R_2$ = 10KΩ en la ecuación 51, se tiene lo siguiente:

➢ **Ecuación 52** $$3{,}3V = \frac{9V * 10\ K\Omega}{R_1 + 10\ K\Omega} \quad (72)$$

Por lo tanto, para encontrar el valor de $R_2$ se realiza lo siguiente:

➢ **Ecuación 53** $$3{,}3 = \frac{90\ K\Omega}{R_1 + 10\ K\Omega} \quad (73)$$

$$3{,}3\ (R_1 + 10\ K\Omega) = 90\ K\Omega$$

$$3{,}3 * R_1 + 33\ K\Omega = 33K\Omega$$

$$3{,}3 * R_1 = 90\ K\Omega - 33K\Omega$$

$$R_1 = \frac{90K\Omega - 33K\Omega}{3{,}3} = 17{,}27K\Omega \sim 20\ K\Omega$$

El valor de $R_1$ calculado es de 17.27, buscando un valor de resistencia comercial, se empleó una resistencia de $20K\Omega$.

### 3.14. Aplicación Android

Para poder acceder a la información de la prótesis desde un móvil, se llevó a cabo el diseño una aplicación Android en el entorno de desarrollo App Inventor. A continuación, se presenta la aplicación Android desarrollada. En la figura 64, se presenta una captura del entorno de desarrollo App Inventor, donde se muestra la interfaz de usuario diseñada.

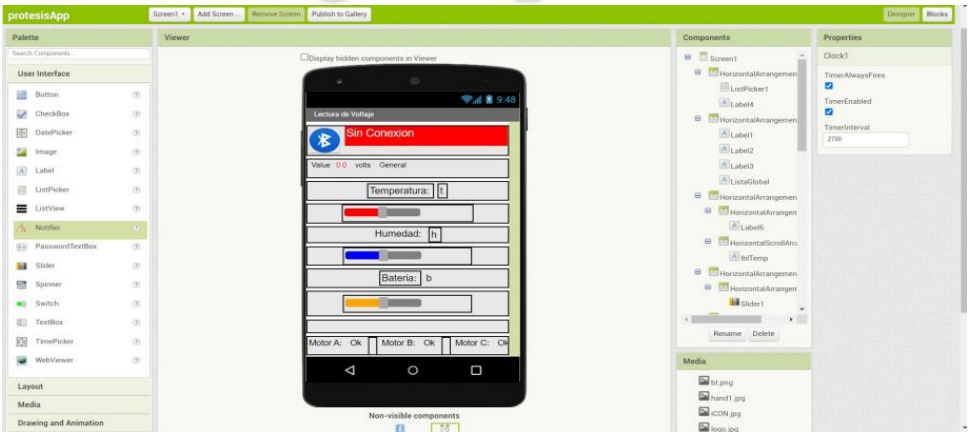



***Figura 64.*** *Entorno de desarrollo App Inventor.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

La interfaz de usuario desarrollada consiste de tres sliders (barras) para indicar la temperatura y humedad dentro del chasis de la prótesis, el tercer slides permitirá mostrar el nivel de la batería. Además, se incorporaron tres indicadores más para poder indicar el estado de los motores.

La aplicación ya instalada en el móvil se va a enlazar con la tarjeta de control por medio del Bluetooth, donde esta incorpora un módulo Bluetooth de primera generación para poder enviar la información de los sensores a la aplicación en el móvil. De esta manera el usuario de la prótesis podrá monitorear el nivel de la batería, estado de los motores, así como la temperatura interna de la misma. Para las funciones de la aplicación móvil se desarrolló el programa que se presenta en la figura 65.

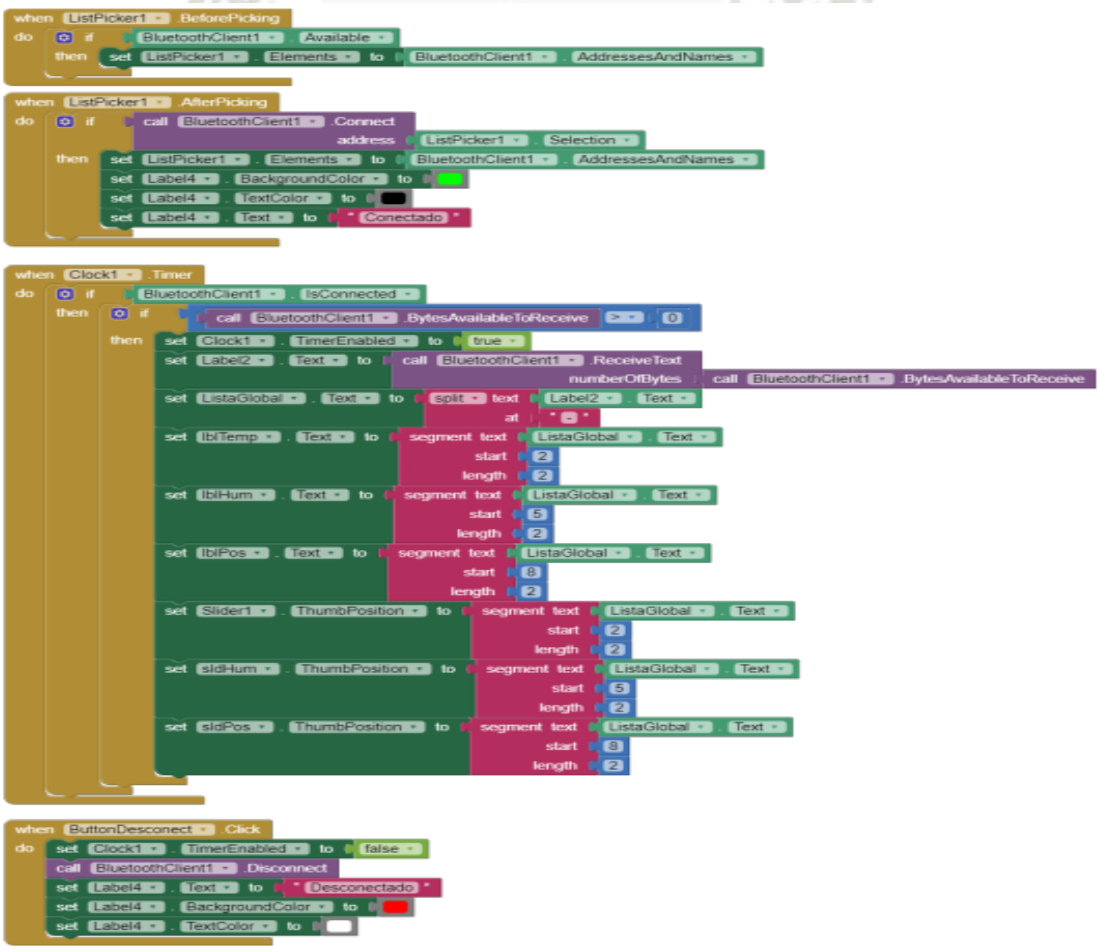



***Figura 65.*** *Programa para aplicación Android.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

La primera parte del programa desarrollado permite llevar a cabo el uso del Bluetooth del móvil para conectar con el módulo Bluetooth colocado en la placa de control. Esta parte se presenta en la figura 66, donde inicialmente se accede a los dispositivos Bluetooth apareados o vinculados y posteriormente se monitorea una conexión exitosa con algún dispositivo, y si es así se desplegará por medio de un texto en la interfaz "conectado" en un recuadro de color verde, y en caso contrario aparecerá el mensaje "desconectado".

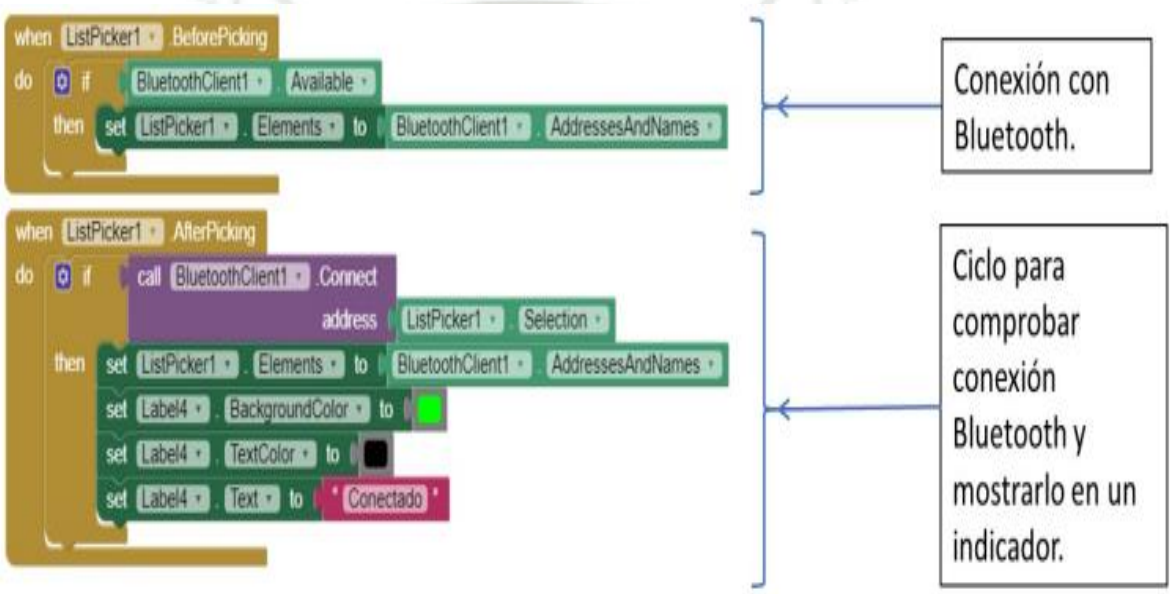



***Figura 66.*** *Primera parte del programa para la aplicación Android*

La segunda parte del programa se presenta en la figura 67, y esta se encarga de recibir los datos enviados en tipo cadena desde la tarjeta de control y almacenarlos en una variable local a la vez en que son separados detectando el símbolo −, posteriormente, se imprimen los valores en los indicadores de la interfaz, esto se hace seleccionado la posición de carácter a mostrar y la cantidad de caracteres que son requeridos.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

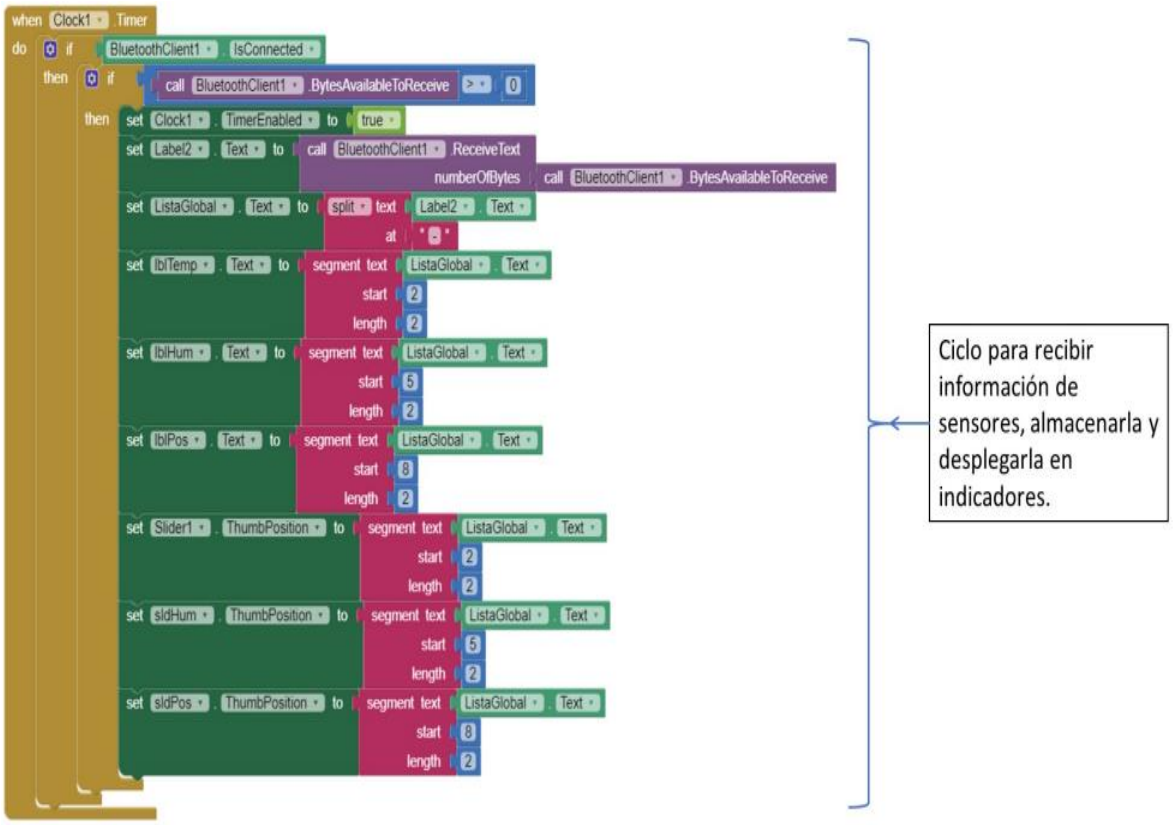



***Figura 67.*** *Segunda parte del programa para la aplicación Android*

La última parte del programa se encarga de terminar la comunicación Bluetooth con la tarjeta de control, de esta manera se dejarán de recibir los datos de la prótesis.

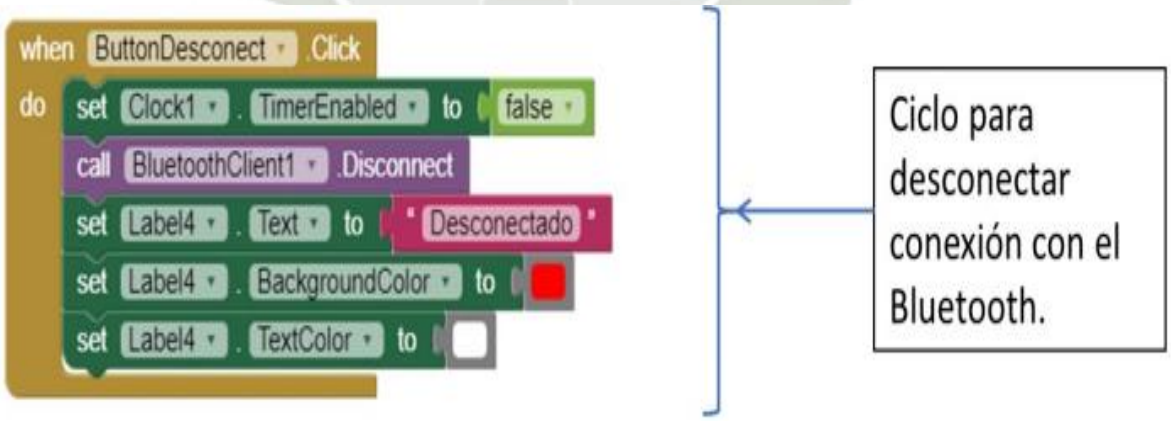



***Figura 68.*** *Tercera parte del programa para la aplicación Android.*

Una vez concluido el desarrollo de la interfaz, se descargó al teléfono móvil y se instaló, en la figura 69, se presenta este proceso. En el inciso (a) se muestra el archivo de instalación para la aplicación en el móvil; en el inciso (b) se muestra la acción de instalar la

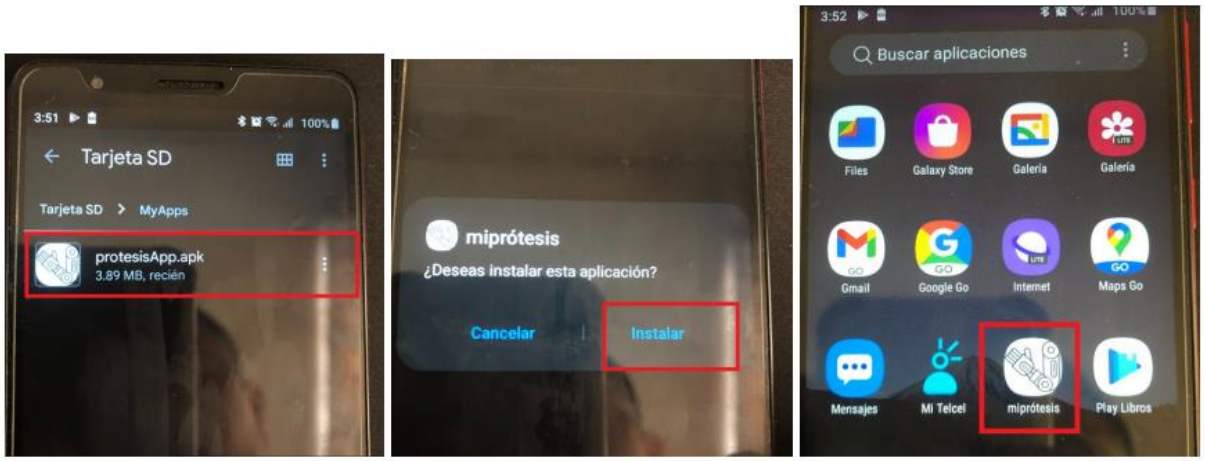


Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

aplicación; y en el inciso (c) se presenta el ícono de la aplicación ya instalada en el menú de aplicaciones del móvil.




(a) (b) (c)

***Figura 69.*** *Proceso de instalación de la aplicación en el móvil.*

- **Diagrama de flujo y diagrama de bloques de la aplicación Android**

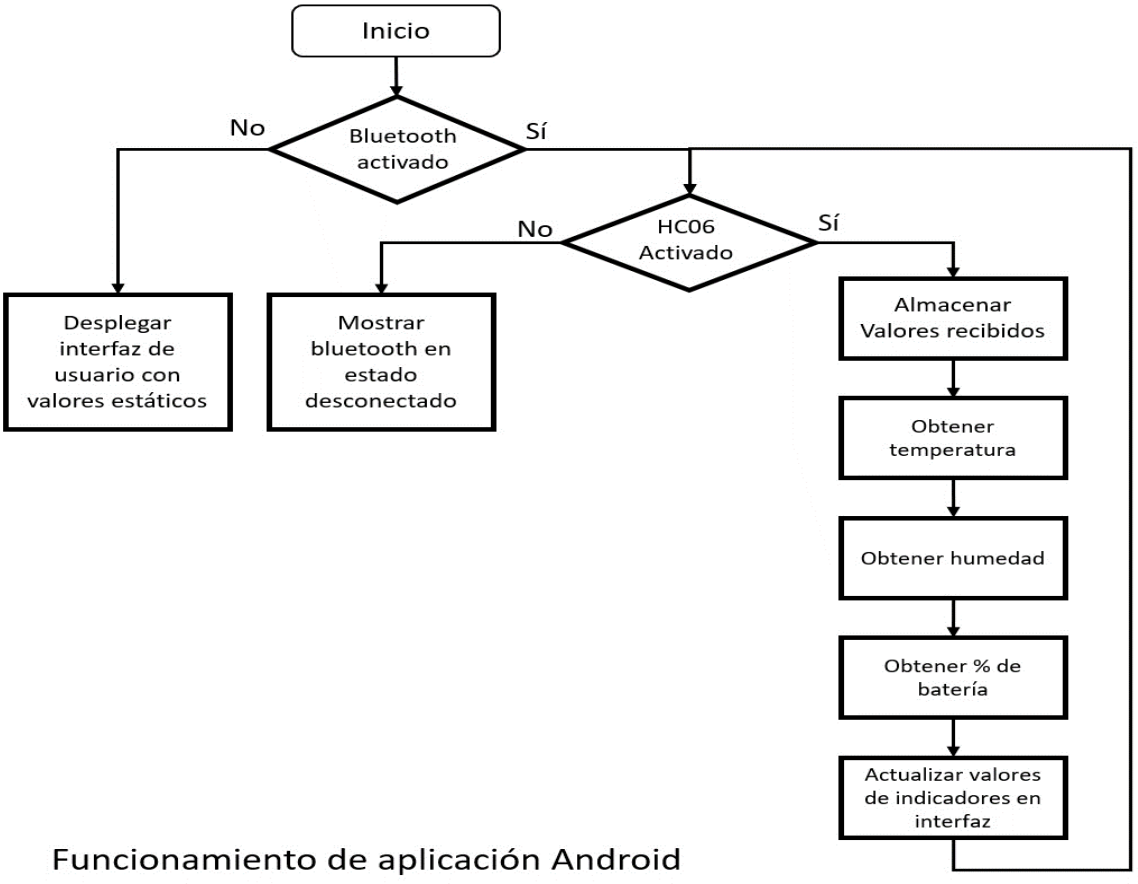



***Figura 70.*** *Diagrama de flujos de la aplicación Android.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- **Descripción del diagrama de flujos**

  - ✓ Primero iniciamos con la conexión Bluetooth si este se activa se apareará con el HC06 si es así aparecerá Conectado, si no, aparecerá Desconectado, la información detallada se encuentra en el punto *3.14 Aplicación Android.*
  - ✓ Si no le logra conectar con el módulo Bluetooth se despliega la interfaz de usuario con valores estáticos esto se detalla en el punto *3.14 Aplicación Android.*
  - ✓ Una vez conectados con el módulo HC06 podremos almacenar los valores obtenidos que son Temperatura, Humedad, estados los motores y porcentaje de la batería esto también se explica en el punto *3.14 Aplicación Android.*
  - ✓ Finalmente, para terminar de usar el aplicativo se presiona en el botón Desconectar y esta terminará de tomar los datos ya mencionados.

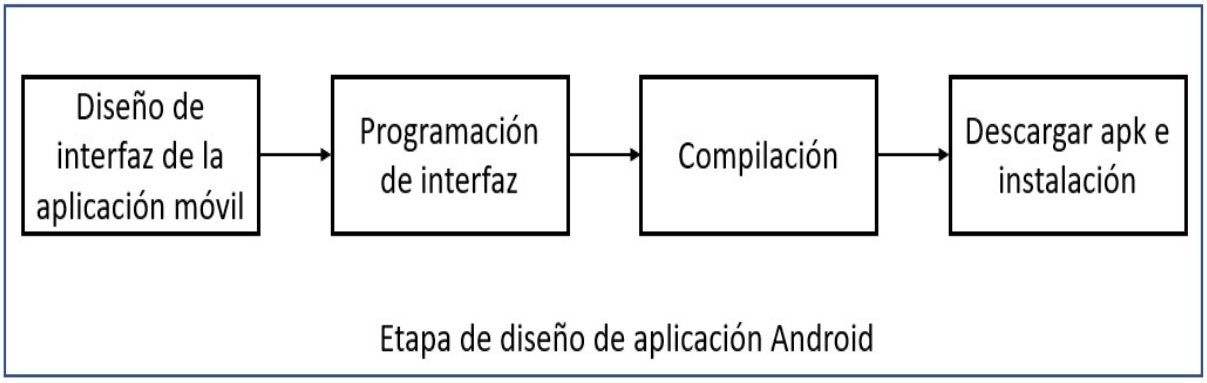



***Figura 71.*** *Diagrama de bloques de la aplicación Android.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

- **Descripción del diagrama de bloques**

La aplicación Android se desarrolló utilizando AppInventor2, ya que es una herramienta gratuita e intuitiva, que permite desarrollar de forma amigable aplicaciones Android.

- ✓ **Diseño de interfaz de la aplicación móvil:** Consistió en colocar los diferentes elementos que componen la interfaz de la app, como son, los botones, barras, texto e imágenes.
- ✓ **Programación de la interfaz:** Consistió en programar utilizando los diferentes bloques que ofrece AppInventor2, para correlacionar la función de los diferentes controles e indicadores de la interfaz. Que consiste conectarse al dispositivo Bluetooth conectado al ESP32, para recibir a través de comunicación serial los datos de temperatura, humedad, nivel de batería, y estado de los motores. Para posteriormente desplegarlos en los diferentes indicadores.
- ✓ **Compilación:** Esta etapa compila el programa generado para generar así un archivo instalable en cualquier dispositivo Android.
- ✓ **Descargar apk e instalación:** Finalmente, se descargó el archivo de instalación generado en el paso anterior (apk), y se guardó en la memoria del dispositivo Android. Luego, se instala desde el dispositivo Android, habilitando la instalación de aplicaciones de origen desconocido. Hecho esto, ya es posible utilizar la aplicación en el dispositivo Android.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

# CAPÍTULO IV

## 10. DISEÑO DE LA PRÓTESIS

### 4.1. Diseño mecánico de la prótesis

En este capítulo se presenta la etapa del diseño de la prótesis y las piezas que la componen. En la figura 72, se presenta la prótesis diseñada para este trabajo, la cual está compuesta de los sub-ensambles: mano, brazo y codo. Para otorgar movimiento a la mano y muñeca se propone utilizar servomotores, mientras que, para generar el movimiento del codo, se propone usar un motor-reductor. A continuación, se presentan los diseños realizados para cada pieza que componen la prótesis.



***Figura 72.*** *Diseño general de la prótesis.*

#### *4.1.1. Mano*

Como primer paso, se comenzó con el diseño de la mano, donde se diseñaron primeramente las falanges de los dedos. Las medidas que se le dieron a las falanges corresponden a valores promedio de los dedos. La figura 73, muestra los dedos diseñados, donde la imagen del inciso (a) corresponde al dedo pulgar, la imagen (b) al índice, la imagen



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

(c) al dedo medio, la imagen del inciso (d) al dedo anular y la imagen del inciso (e) corresponde al dedo meñique.

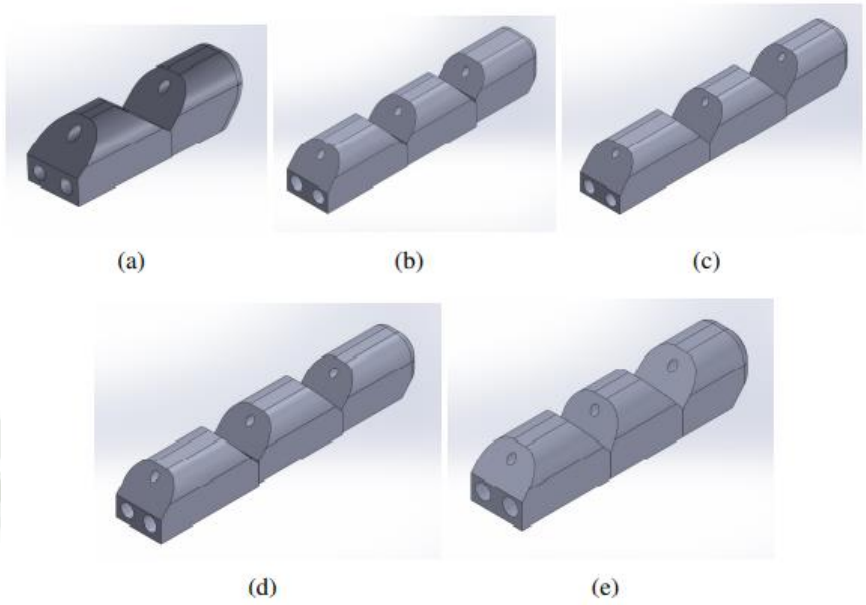



***Figura 73.*** *Diseño de los dedos que componen la mano.*

Por otro lado, se diseñó la palma de la mano configurada para que sea derecho, y se presenta en la figura 74

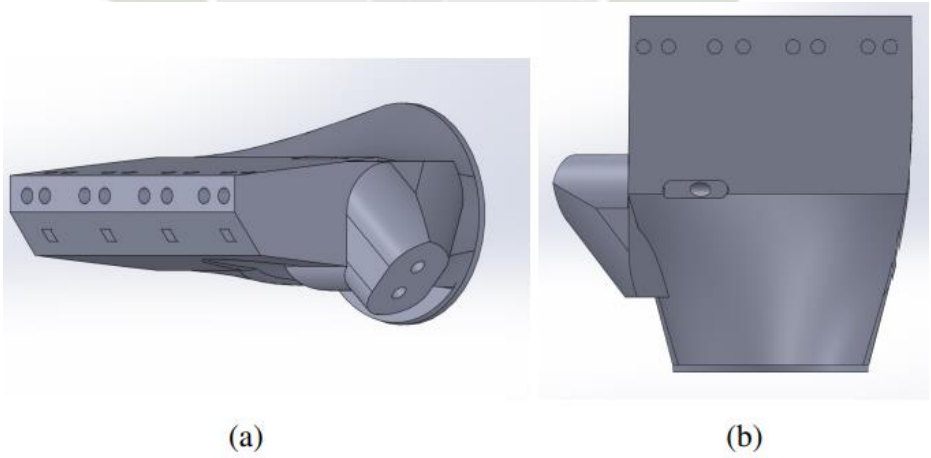



***Figura 74.*** *Diseño de la palma de la mano.*

Una vez llevado a cabo el diseño de los dedos y palma de la mano, se realizó el ensamble de la mano derecha, en la figura 75 se presenta este ensamble.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

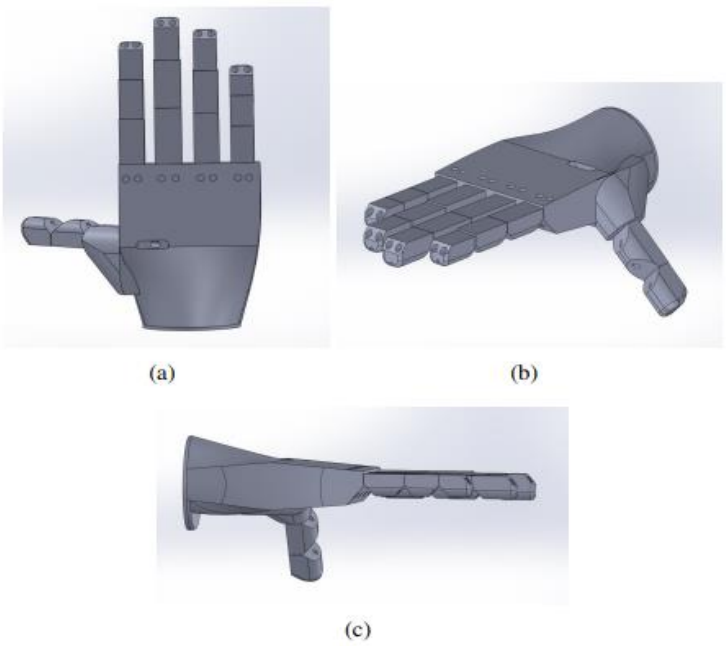



***Figura 75.*** *Diseño de los dedos que componen la mano.*

### *4.1.2. Muñeca*

Se continuó con el diseño de la muñeca, la cual corresponde a la primera parte del diseño del brazo. En la figura 76 se muestra el diseño correspondiente a la muñeca, donde se ensamblará, por un lado, la mano y por el otro la segunda parte del brazo.

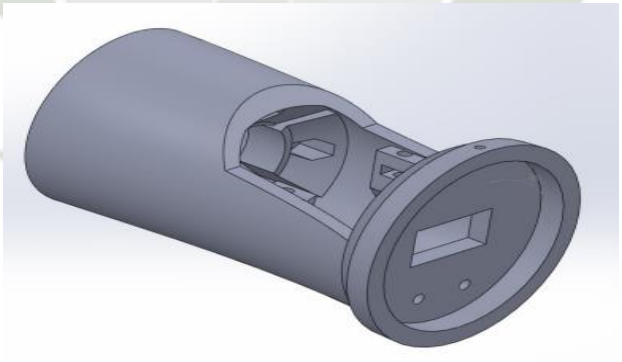


***Figura 76.*** *Pieza para emular el movimiento de la muñeca.*

Para el ensamble con la segunda parte del brazo se diseñó un cople, el cual se va a encargar de transportar el tendón encargado de generar la tensión para abrir y cerrar la mano y a su vez conecta con un rodamiento para producir así los movimientos de supinación y pronación de la muñeca. En la figura 91, se presenta el cople diseñado.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

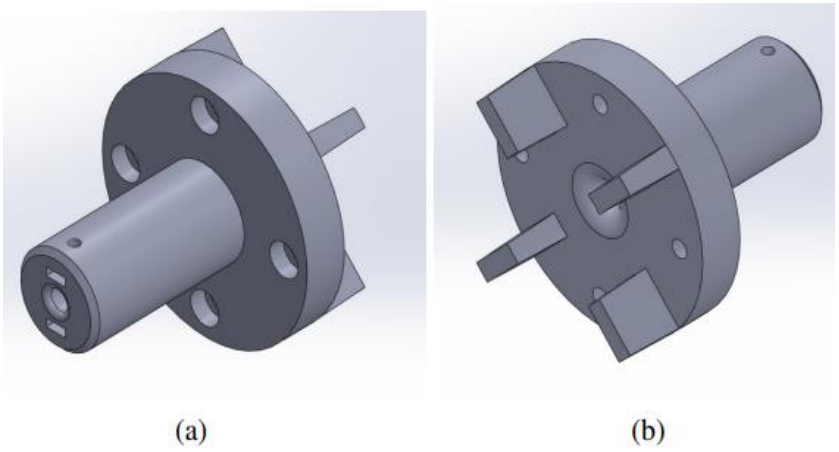



***Figura 77.*** *Pieza de acoplamiento para la muñeca.*

### *4.1.3. Brazo*

Como siguiente paso, se realizó el diseño del brazo, el cual se encarga de albergar a los dos servomotores que otorgan movimiento a la mano y muñeca, también este se acopla con la pieza de la muñeca a través del cople presentado en la figura 91. En la figura 78, se presenta el brazo diseñado, donde la imagen del inciso (a) presenta la vista frontal isométrica y la imagen del inciso (b) la vista trasera isométrica de la pieza brazo.

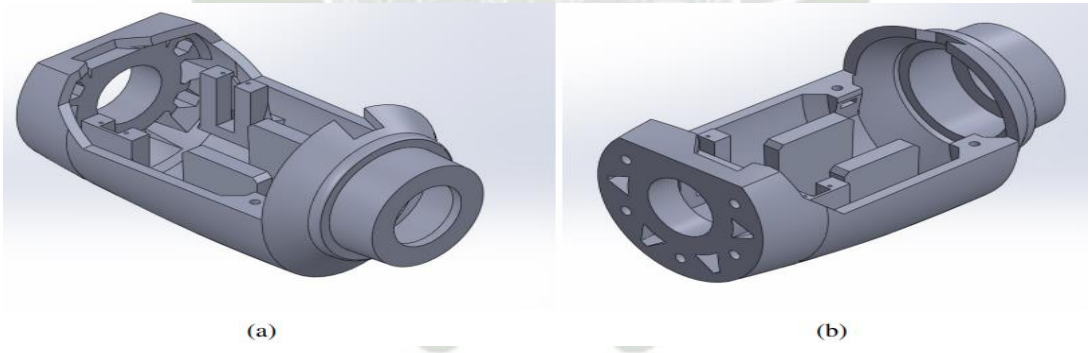



***Figura 78.*** *Pieza para emular el brazo.*

Para poder asegurar el acoplamiento con la pieza correspondiente a la muñeca y a su vez transmitir el movimiento rotacional hacia esta, se diseñó una pieza cople donde, por un lado, se va a empotrar la corona del servomotor encargado de mover la muñeca y por el otro



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

lado se asegura la pieza de la muñeca. La figura 79, se presenta el cople de aseguramiento diseñado.

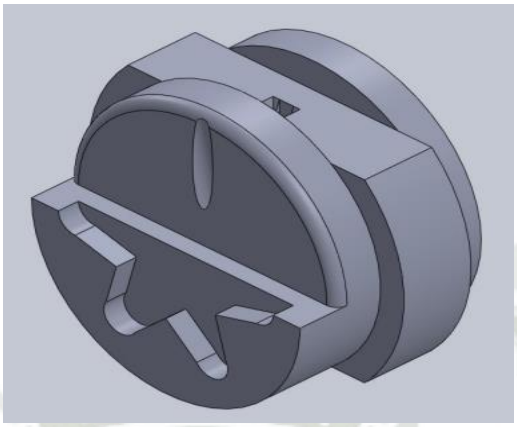


***Figura 79.*** *Pieza de acoplamiento para corona de servomotor de la muñeca.*

Para poder transmitir el movimiento del codo hacia la pieza del brazo, se diseñó un tercer cople para empotrar el eje del motor-reductor que se encargará de dar movimiento al codo. En la figura 80, se presenta el cople diseñado para el codo, así mismo, esta pieza se ensambla a la pieza del brazo por medio de cinco tornillos M4X20 y su respectiva tuerca de sujeción.

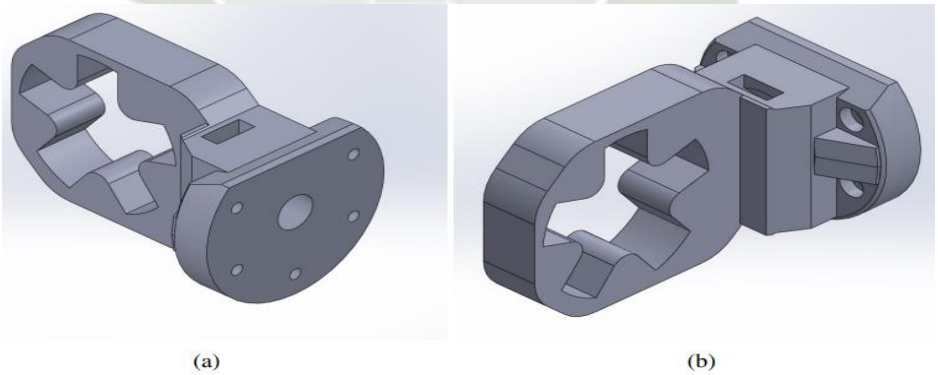



***Figura 80.*** *Pieza de acoplamiento del codo.*

### *4.1.4. Ensamble del brazo*

Con las piezas diseñadas se realizó un primer ensamble de la parte del brazo. En la figura 81, se presenta el ensamble de las piezas que componen el brazo, las imágenes de los incisos (a) y (b) presenta una vista isométrica, mientras que la imagen del inciso (b) presenta una vista lateral del ensamble del brazo.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

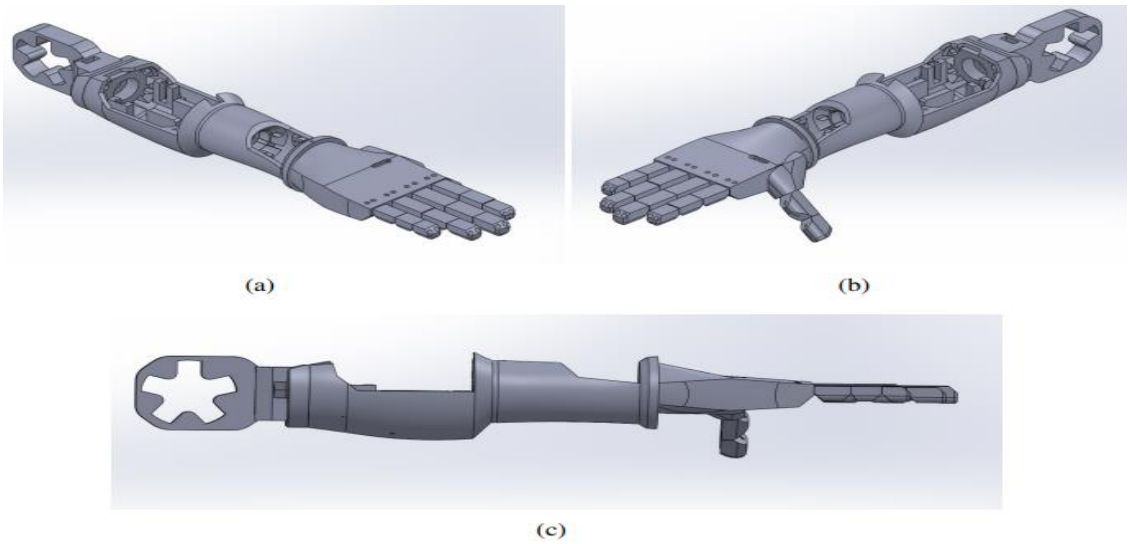



***Figura 81.*** *Representación del ensamble del brazo.*

La figura 82, presenta una vista explosionada del ensamble del brazo, donde se puede apreciar de mejor manera las diferentes piezas que lo componen.

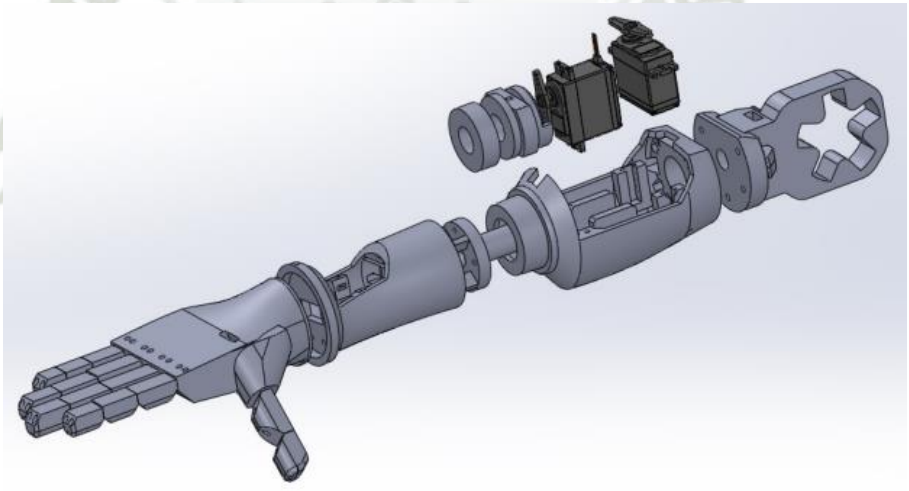


***Figura 82.*** *Vista explosionada del ensamble del brazo.*

### *4.1.5. Codo*

Por último, se diseñaron las piezas que emulan el codo de la prótesis. La figura 83, muestra el diseño del chasis para empotrar el motor-reductor (en el lado derecho).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

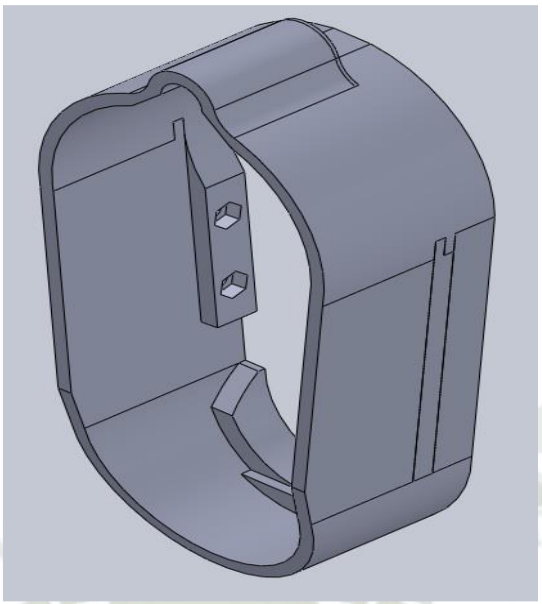


***Figura 83.*** *Chasis para el motor del codo.*

La figura 84, muestra el diseño del chasis donde se colocarán los circuitos de adquisición de señales y control (lado izquierdo).

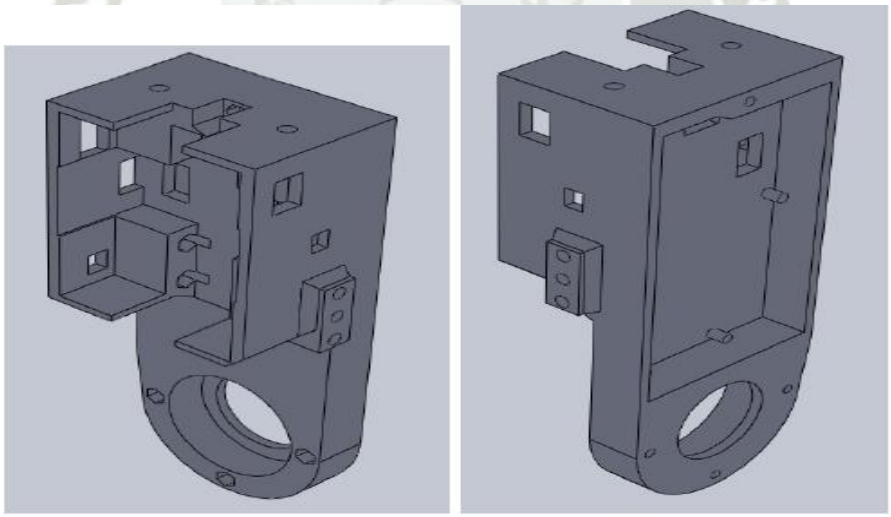



***Figura 84.*** *Representación del ensamble del brazo.*

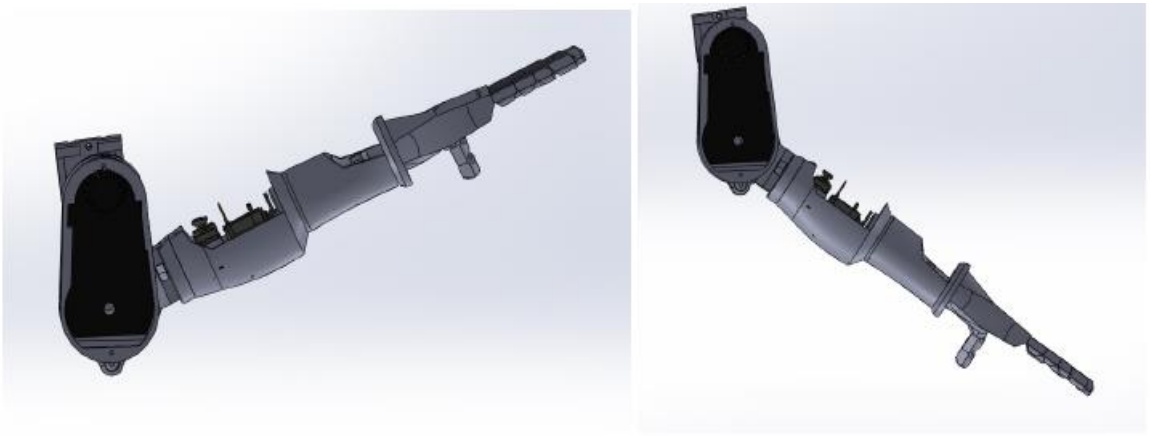
La figura 85, presenta el ensamble del codo con el brazo, donde la imagen del inciso (a) presenta una vista lateral con una posición de contracción del codo y la imagen del inciso (b) con una posición en flexión del codo.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA



(a) (b)

***Figura 85.*** *Representación del ensamble del brazo.*

Por otro lado, la figura 86 presenta el ensamble general de la prótesis, donde la imagen del inciso (a) muestra una vista frontal y la imagen del inciso (c) presenta una vista isométrica.



(a) (b)

***Figura 86.*** *Representación del ensamble del brazo.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## 4.2. Construcción de la prótesis

### *4.2.1. Fabricación de chasis de prótesis*

A continuación, se presentan las partes fabricadas en impresión 3D que componen el chasis de la prótesis, comenzando con la mano, posteriormente se presenta la muñeca y por último el brazo. Cada parte mencionada cuenta con al menos 20 horas de tiempo de impresión.

#### 4.2.1.1. Mano

La fabricación de la prótesis comenzó por la impresión de las piezas que componen la estructura de la mano. Se diseñaron dos modelos de la misma, donde el diseño fue desechado debido a que, en pruebas iniciales, presentó un bajo desempeño en el agarre de objetos, mientras que el diseño mostrado en la imagen del inciso (b), presentó un mejor agarre, porque lo seleccionó. Cabe mencionar que cada mano fabricada requirió un tiempo de fabricación de 20 horas.



(b)

***Figura 87.*** *Diseño de mano realizado; (b) segundo diseño.*

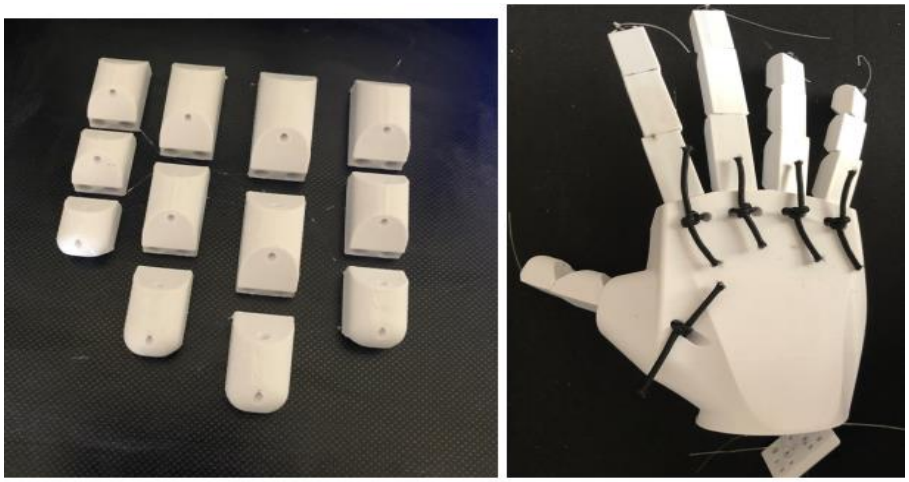
En la figura 88, se muestra el segundo diseño de mano realizado, donde en el inciso (a) se presentan las falanges impresas, y en el inciso (b) se muestra el ensamble terminado de la mano.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA



(a) (b)

***Figura 88.*** *Segundo diseño de mano realizado; (a) falanges de dedos de la mano; (b) ensamble de piezas que componen la mano.*

Para llevar a cabo el ensamble de la mano se emplearon los materiales que se presentan en la figura 89, que consisten en: hilo de plástico, el cual permitirá jalar las falanges de cada dedo de la mano, es decir, realizarán la tarea de los tendones en una mano biológica; resorte para unir las falanges que componen cada dedo de la mano; y unas tijeras para cortar los materiales mencionados.

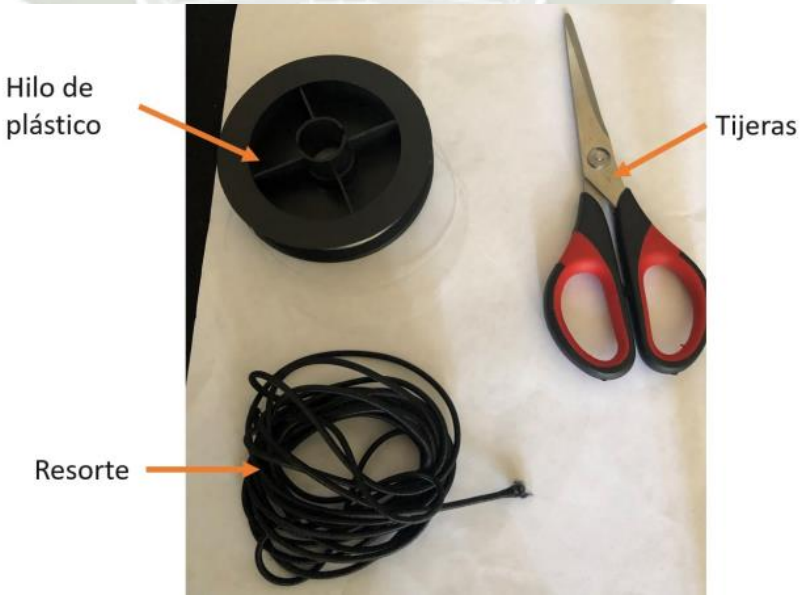



***Figura 89.*** *Materiales utilizados para el ensamble de la mano.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

#### 4.2.1.2. Muñeca

Una vez concluido la fabricación de la mano se continuó con la muñeca, para la cual se realizaron de igual manera dos diseños. Este primer diseño de muñeca realizado, el cual soporta dos servomotores, los cuales ejecutarán los movimientos para contraer los dedos de la mano y así poder tomar un objeto. Esta pieza requirió de un tiempo de impresión de 26 horas. Al momento de evaluar el funcionamiento de esta pieza se observó que se ejercía un peso considerable por los dos servomotores empotrados, lo que trae como consecuencia una ejecución de movimientos defectuosa por parte del servomotor de la muñeca.

Debido a lo anterior mencionado, se continuó con un segundo diseño, con el cual se reduce la cantidad de servomotores para poder efectuar la tarea de tomar objetos, a solo uno y, por otro lado, con el objetivo de mejorar la distribución del peso que ejercen los servomotores, se decidió pasar el servomotor hacia la parte del brazo, dicha parte se presenta en la siguiente subsección. En la figura 90, se muestra la pieza correspondiente al segundo diseño de la muñeca, la cual es menos voluminosa que el primer diseño. Esta pieza requirió de 20 horas en tiempo de impresión.



(a) (b)

***Figura 90.*** *Segundo diseño de muñeca realizado.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

Además de la pieza presentada en la figura 90, se requiere y se diseñó una segunda pieza de unión, la cual permitirá unir la muñeca con el brazo. En la figura 91, se presenta esta pieza, misma que se ensambla con la muñeca a través de tornillos M3. Esta pieza requirió de un tiempo de impresión de 8 horas.

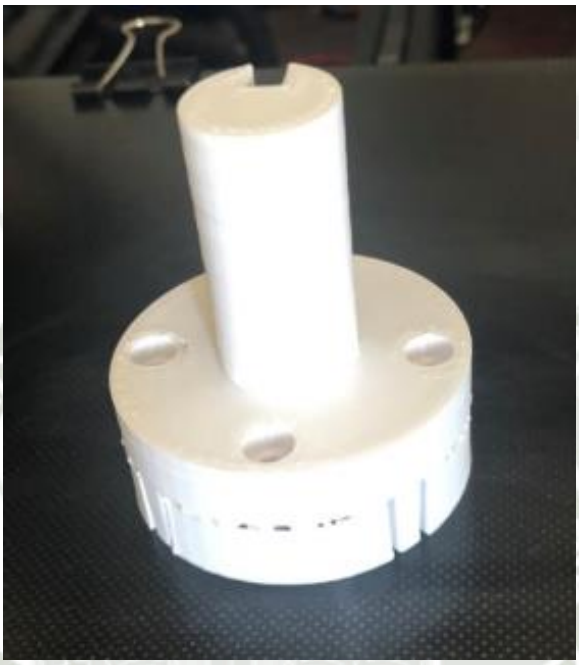


***Figura 91.*** *Cople para muñeca y brazo.*

#### 4.2.1.3. Brazo

Como pieza siguiente, se fabricó la correspondiente al brazo, misma que se encargará de empotrar los servomotores que otorgarán el movimiento de la muñeca y los tendones de la mano para contraerla y así tomar un objeto. Es decir, dos servomotores.

En la figura 92 se muestra el brazo fabricado, en la imagen del inciso (a) se presenta el proceso de impresión 3D del mismo, y la imagen del inciso (c) muestra la pieza terminada. Esta pieza requirió un tiempo de impresión de 28 horas.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

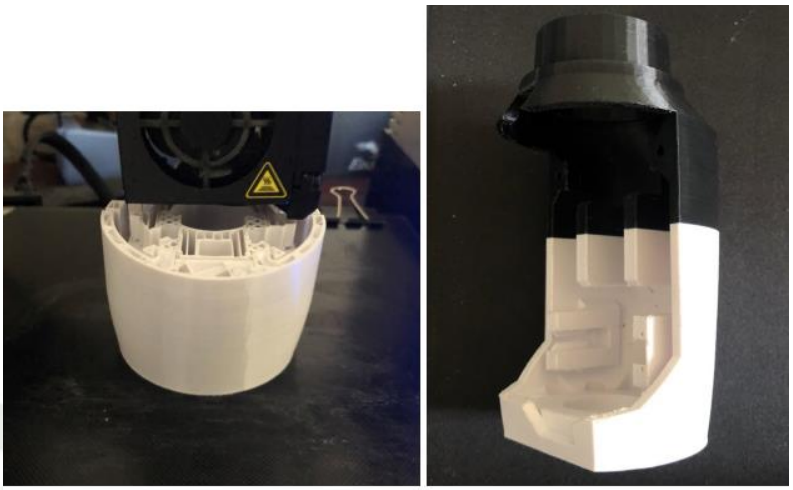



***Figura 92.*** *Segundo diseño de muñeca realizado.*

#### 4.2.1.4. Codo

Para el mecanismo que emule los movimientos del codo se diseñaron cuatro piezas principales, que consisten en: cople codo, cople eje motor, parte lateral izquierda y parte lateral derecha; estas se presentan en la figura 93.

La pieza cople codo cumple con tres funciones, la primera es, acoplarse a la pieza brazo (figura 92), la segunda es, transmitir el movimiento angular generado por el motor presentado en la figura 102 hacia el brazo de la prótesis, y la tercera es, acoplar dos resortes que ayude a levantar el brazo de la prótesis.

La pieza cople motor permite acoplar el eje del motor reductor presentado en la tabla de componentes con la pieza cople codo, para así transmitir el movimiento del motor hacia el brazo, emulando así el movimiento angular del codo.

La pieza codo lateral derecho permite empotrar el motor presentado en la figura 102 y se coloca en la parte lateral derecha de la prótesis. Así mismo, esta pieza permite acoplar el circuito de potencia.

La pieza codo lateral izquierdo permite colocar los circuitos de adquisición de señales mio-eléctricas y de control.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA



(a) Pieza cople codo. (b) Pieza cople eje motor.

(c) Pieza codo lateral derecho. (d) Pieza codo lateral izquierdo.

***Figura 93.*** *Piezas que componen el mecanismo del codo en la prótesis.*

### *4.2.2. Ensamble del brazo*

A continuación, se presenta el proceso de ensamble de las piezas que componen el brazo de la prótesis. En la figura 94, se muestra el resultado obtenido del ensamble, el cual se compone de tres piezas principales: mano, muñeca y brazo. Además, se indican las partes A, B y C, que representan lo siguiente:

- **Parte A**: Zona perteneciente a la pieza que emulara una mano, la cual está compuesta por los cinco dedos unidos por medio de cordones elásticos que emulan la función de los tendones.
- **Parte B**: Zona que corresponde a la pieza que emulara los movimientos de una muñeca.
- **Parte C**: Pertenece a la pieza del brazo, que está unida a la pieza que emula la muñeca, asimismo, esta pieza contendrá dos servomotores, uno para realizar los



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

movimientos de la muñeca, y otro para contraer los tendones de los dedos de la mano para poder tomar objetos.

- **Parte D**: Corresponde a la pieza que permitirá acoplarse a la pieza que emulará el codo.

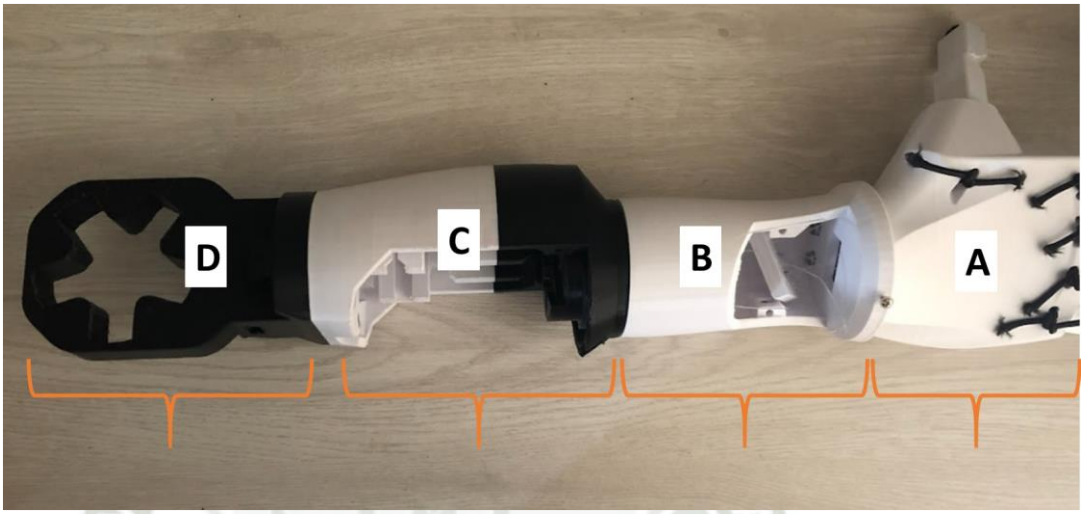



***Figura 94.*** *Descripción del ensamble.*

Para comenzar con el ensamble de las piezas impresas, se comenzó a colocar el cople (ver figura 91) que permitirá ensamblar las piezas muñeca y brazo. En la figura 95 se muestra el proceso para colocar el cople en la pieza de la muñeca, para lo cual se requiere de tornillos y tuercas M3.

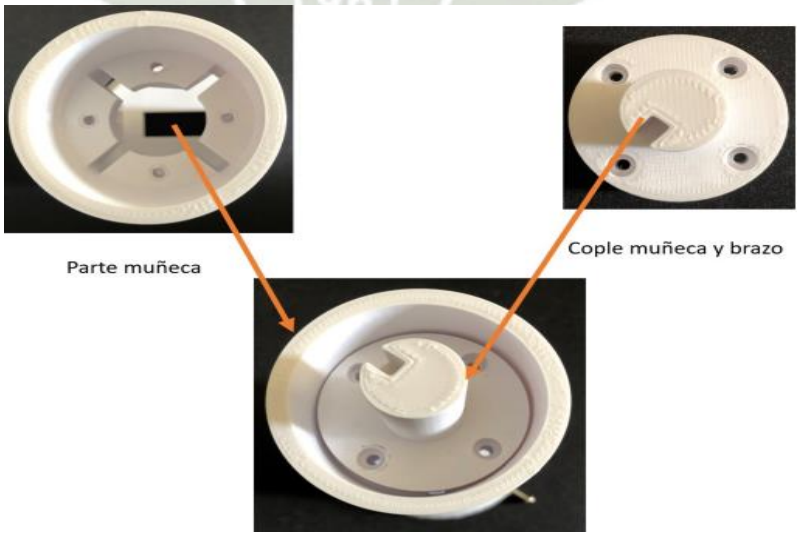



***Figura 95.*** *Ensamble entre pieza muñeca y cople.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

En la figura 96, se muestra la pieza resultante del ensamble anterior, de la cual sobresale un eje, mismo que se colocará sobre un rodamiento empotrado en la pieza del brazo, y que a su vez se empleará para ejercer los movimientos de pronación y supinación de la muñeca a través de un servomotor.



***Figura 96.*** *Resultado de ensamble entre cople y muñeca.*

Una vez hecho lo anterior, se continuó con el ensamble de la pieza brazo, esto se hace mediante la colocación del eje del cople empotrado sobre la muñeca sobre un rodamiento colocado sobre la pieza brazo. Este procedimiento se presenta en la figura 97.

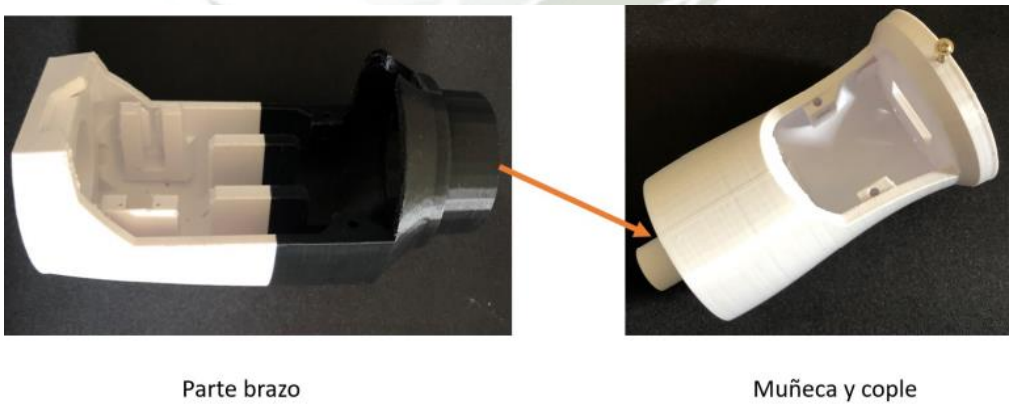



***Figura 97.*** *Ensamble entre piezas muñeca y brazo.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

En la figura 98, se muestra a detalle el proceso anterior y las piezas que lo componen.

- La imagen (a) corresponde a la muñeca, donde sobresale el eje del cople (recuadro color rojo), mismo que permite la transmisión del movimiento para la mano a través de un cordón.
- La imagen (b), presenta el rodamiento colocado sobre la pieza brazo, por la cual se empotra el eje de la muñeca.
- La imagen (c), muestra el cople para eje del servomotor B para transmisión del movimiento de la muñeca.
- La imagen (d), muestra el ensamble completo, donde se encuentran las piezas colocadas en su sitio, además se puede observar el cordón color rojo que sobresale, y que será colocado sobre el motor A, para transmitir el movimiento a la mano.

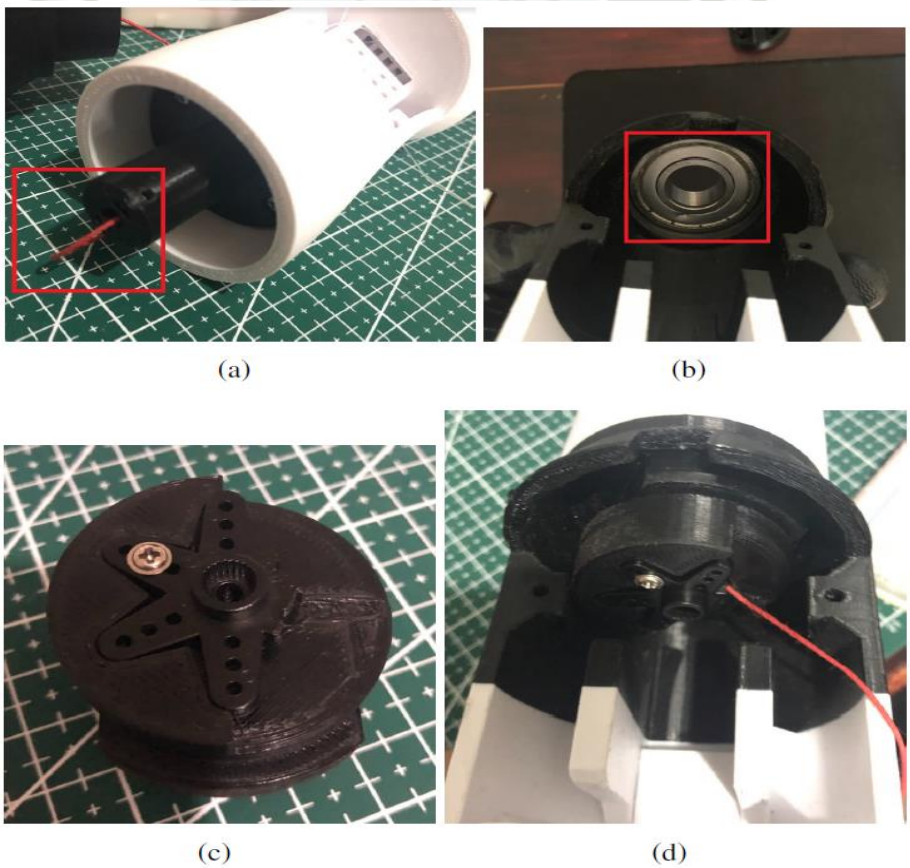



***Figura 98.*** *Ensamble entre piezas muñeca y brazo.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

El resultado final del ensamble se presenta en la figura 99.

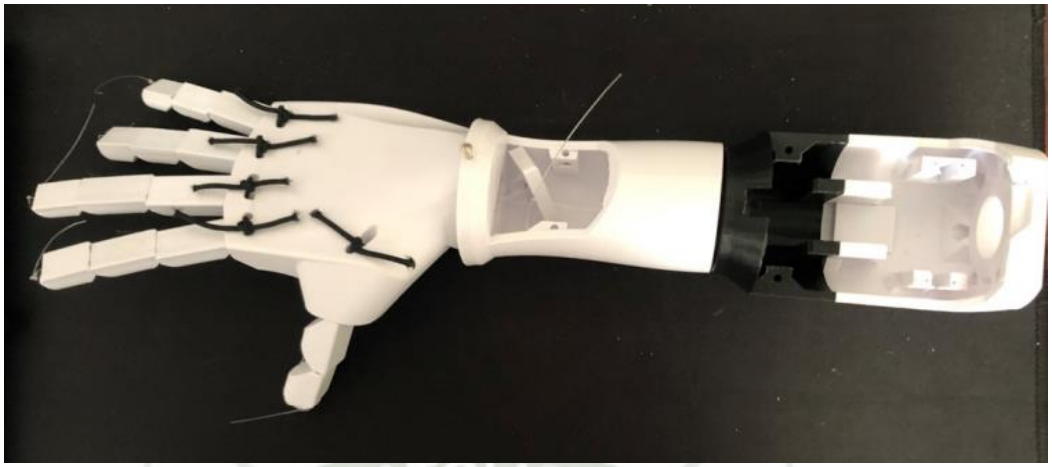


***Figura 99.*** *Cople para muñeca y brazo.*

Una vez concluido con el ensamble de las partes A, B y C (mano, muñeca y brazo, respectivamente), se continuó con la incorporación de la pieza que permitirá la unión con la pieza que emulará el codo, para ello se usará la pieza cople de codo que presenta en la figura 93 imagen (a).

La pieza, que se presenta en la imagen (a) de la figura 93, se acopla a la pieza brazo por medio de cuatro tornillos M4, como se muestra en la figura 100.

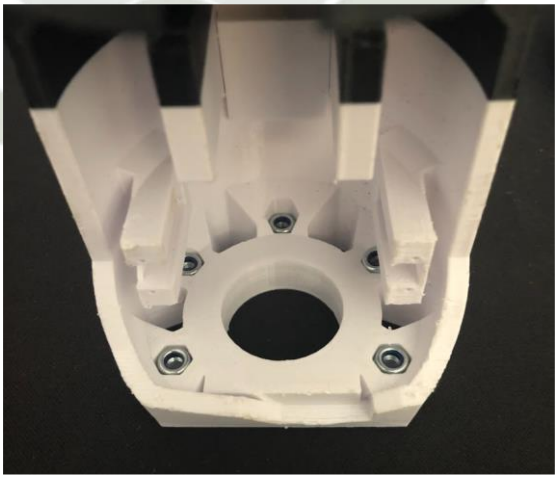


***Figura 100.*** *Vista interna de la pieza brazo que permite visualizar las tuercas empleadas para la sujeción de la pieza cople codo.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

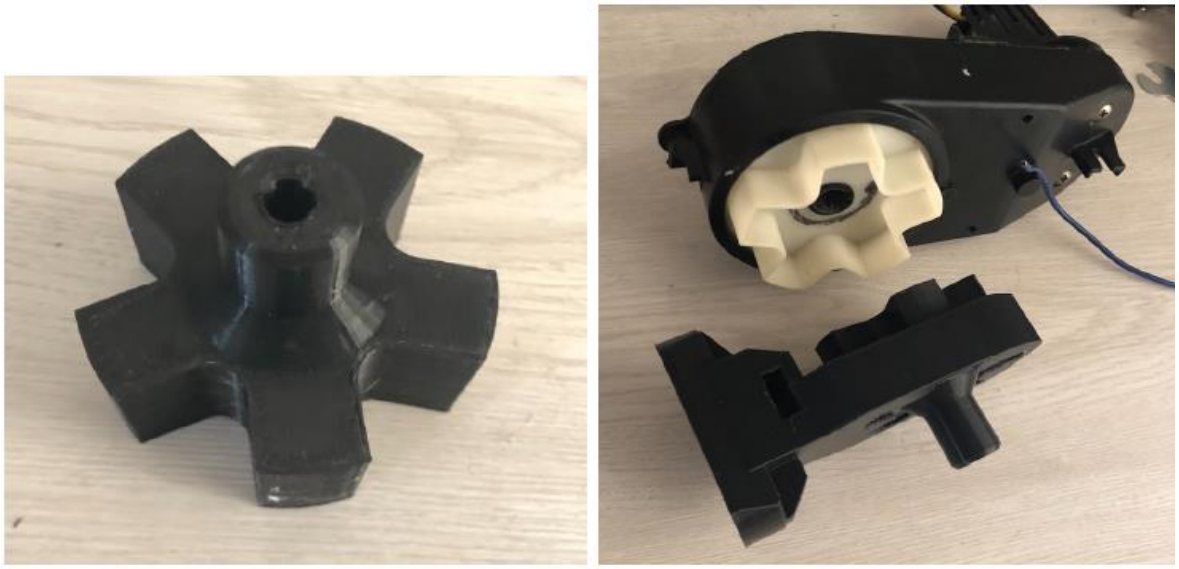
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

El resultado de lo anterior, se presenta en la figura 101, donde se puede apreciar la pieza que corresponde al brazo (color blanco) y la pieza de cople codo (color negro), esta última permitirá introducir un segundo cople para incorporar el motor que generará los movimientos de contracción y flexión del codo.



***Figura 101.*** *Cople codo acoplado a pieza brazo.*

En el inciso (a) la figura 102, se muestra el cople que permitirá colocar el eje de un motor-reductor, el cual se muestra en el inciso (b) de la misma figura.



(a) (b)

***Figura 102.*** *Cople para eje del motor-reductor. (a) Pieza cople eje motor-reductor; (b) motor-reductor y cople codo.*

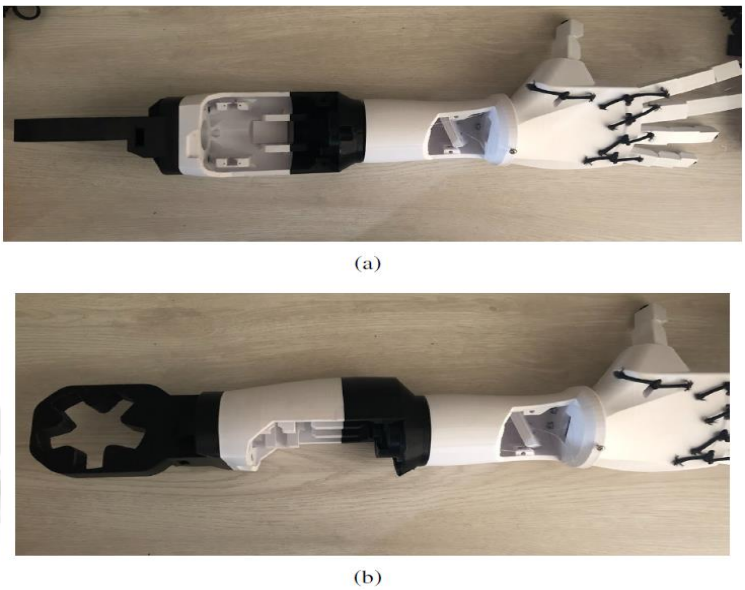
Como resultado de lo anterior, se tiene la primera parte del ensamble de la prótesis, el cual se presenta en la figura 103.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA



(a)

(b)

***Figura 103.*** *Resultado del ensamble de las piezas mano, muñeca, brazo y cople codo.*

### *4.2.3. Incorporación de motores A y B*

Antes de continuar con la segunda parte del ensamble que corresponde al mecanismo del codo, se realizó la incorporación de los motores A y B, a continuación, se presenta este procedimiento. Los motores son pieza clave para que la prótesis pueda emular los movimientos de la mano, muñeca y codo, a continuación, se presenta la manera en que fueron colocados los motores A y B, encargados de proporcionar movimiento a la mano y muñeca.

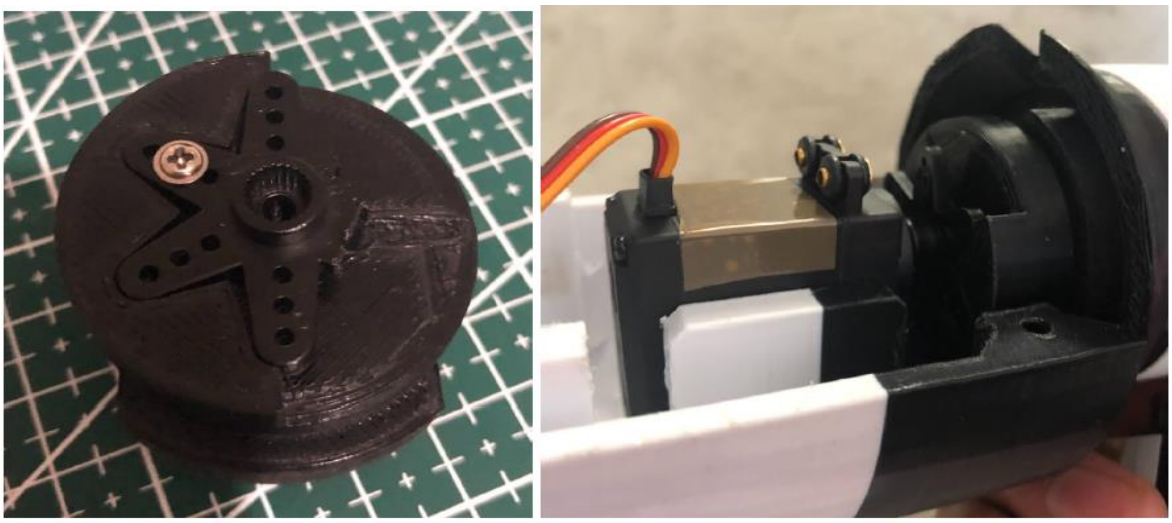
Inicialmente, colocó el servomotor B en la pieza brazo para poder otorgar los movimientos correspondientes a la muñeca (pronación y supinación), en el inciso (a) de la figura 88, se muestra el cople que permitirá colocar la corona del servo y el eje de la muñeca (cople muñeca), en el inciso (b) de la misma figura se muestra el servomotor acoplado.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA



(a) (b)

***Figura 104.*** *Servomotor para otorgar el movimiento de la muñeca (pronación y supinación).*

Con el objetivo de monitorear la posición de los servomotores para así identificar alguna falla en sus movimientos, se incorporaron dos sensores de efecto hall, para lo cual se colocó un imán de neodimio sobre el eje de cada servomotor.

La imagen (a) de la figura 105, muestra el circuito que incorpora los sensores de efecto Hall, denominados *S1* y *S2*, para monitorear el servomotor A y B respectivamente. Este circuito se alimenta con 3.3 volts provenientes de la fuente que proporciona el módulo ESP32, y como salidas tiene a las señales de los sensores *S1* y *S2*, que son conectados hacia el ESP32.

La imagen del inciso (b) de la figura 105, presenta los imanes de neodimio colocados sobre las coronas se los servomotores (recuadros remarcados en color azul), y los respectivos sensores de efecto Hall que los monitorearan (recuadros remarcados en color rojo).



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

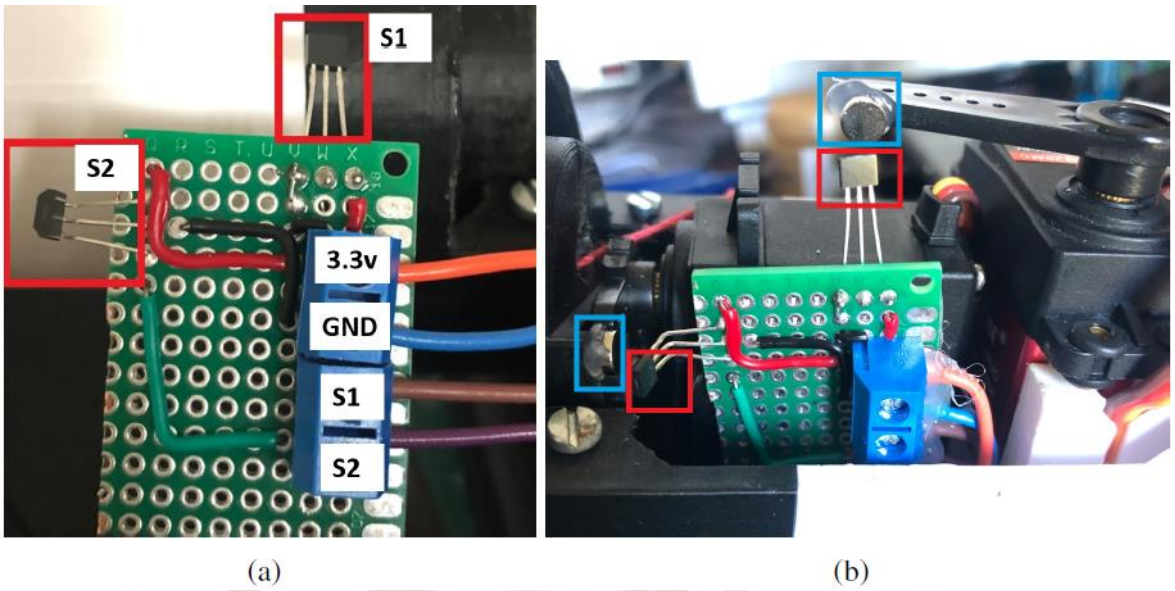



***Figura 105.*** *Circuito para incorporar dos sensores de efecto Hall.*

### *4.2.4. Ensamble del codo*

Una vez colocados los dos servomotores (A y B), se comenzó con el ensamble de las piezas que componen el mecanismo del codo.

Inicialmente, se colocó el motor-reductor que permitirá dar movimiento al codo (flexión y contracción). En el inciso (a) de la figura 106 se muestra la parte trasera del motor, y en el inciso (b) de la misma figura se muestra el circuito de potencia empotrado.

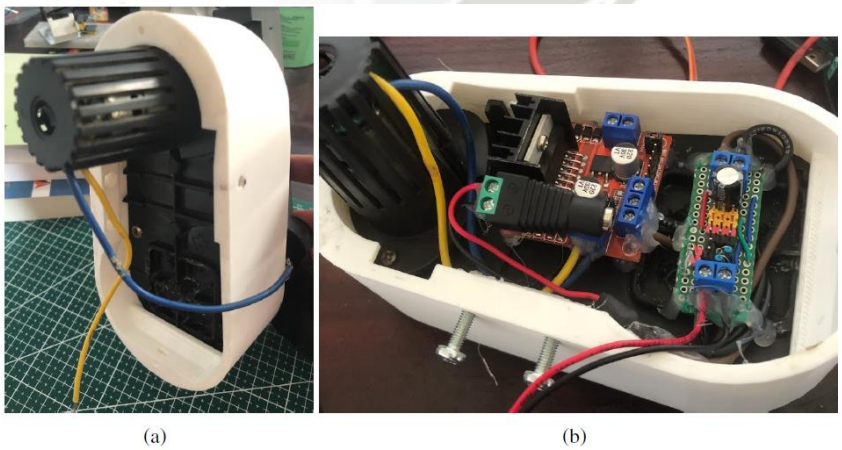



***Figura 106.*** *Ensamble de motor-reductor para emular movimientos del codo (flexión y contracción) y circuito de potencia.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

Se generó un circuito auxiliar para conectar las dos fuentes de alimentación, circuito divisor de voltaje (para medir el nivel de batería) y servomotores. El circuito generado se presenta en la figura 107.

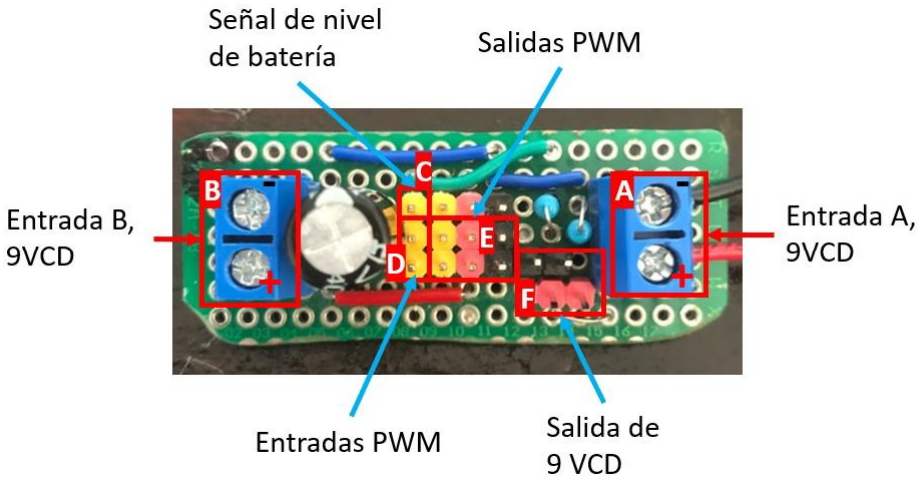
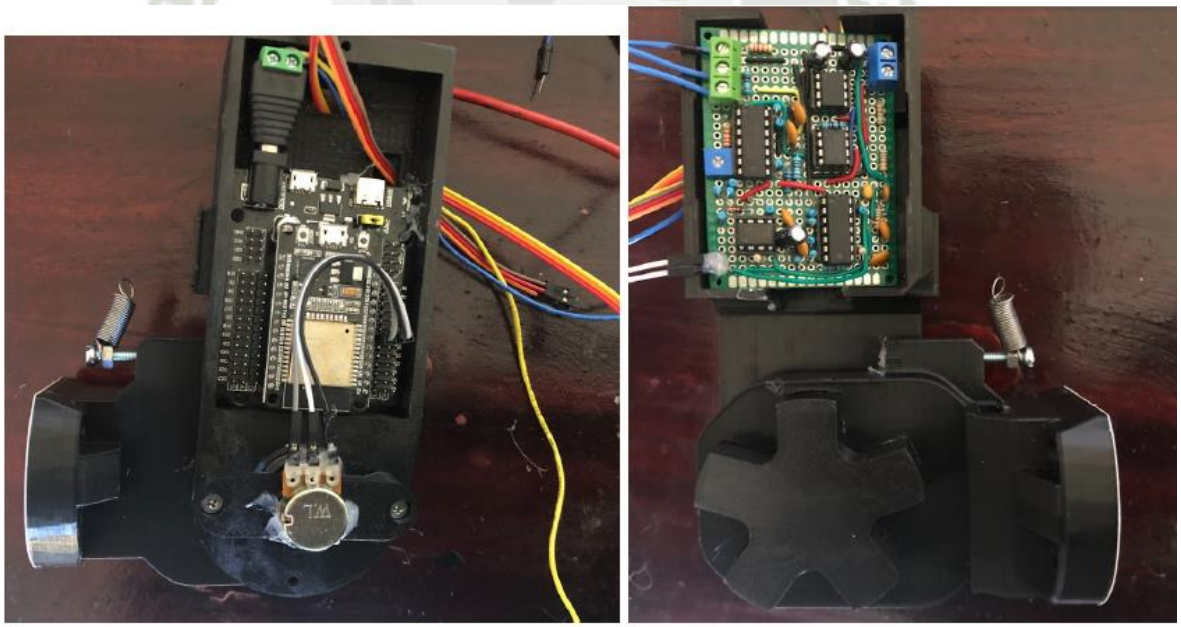



***Figura 107.*** *Circuito auxiliar.*



(a) (b)

***Figura 108.*** *Circuitos de control y potencia en el chasis del codo de la prótesis.*

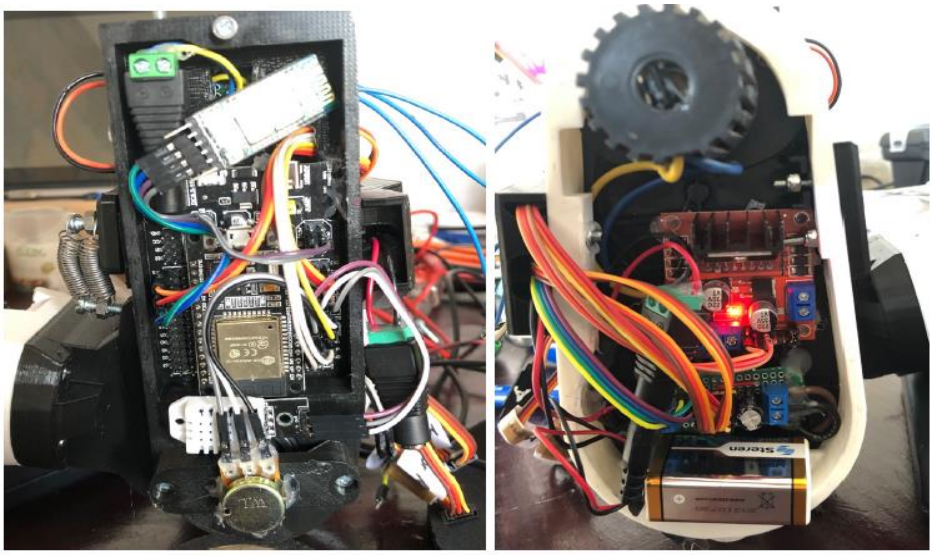


Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Posteriormente, dentro de las piezas que componen el codo de la prótesis, se colocaron los circuitos de control (inciso (a) de la figura 109), y de potencia (inciso (b) de la figura 109).



(a) (b)

***Figura 109.*** *Circuitos de control y potencia en el chasis del codo de la prótesis.*

En la figura 110, se muestra una vista lateral del codo (inciso a) y vista frontal (inciso b).



(a) (b)

***Figura 110.*** *Ensamble de las piezas que componen el codo de la prótesis.*

A continuación, en la figura 111 se presenta una relación de dimensión de las piezas que componen el brazo y mano de la prótesis, donde se puede observar que el tamaño de



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

esta es similar a la de un brazo y mano promedio, midiendo en largo 37 cm (desde dedo índice hasta codo).

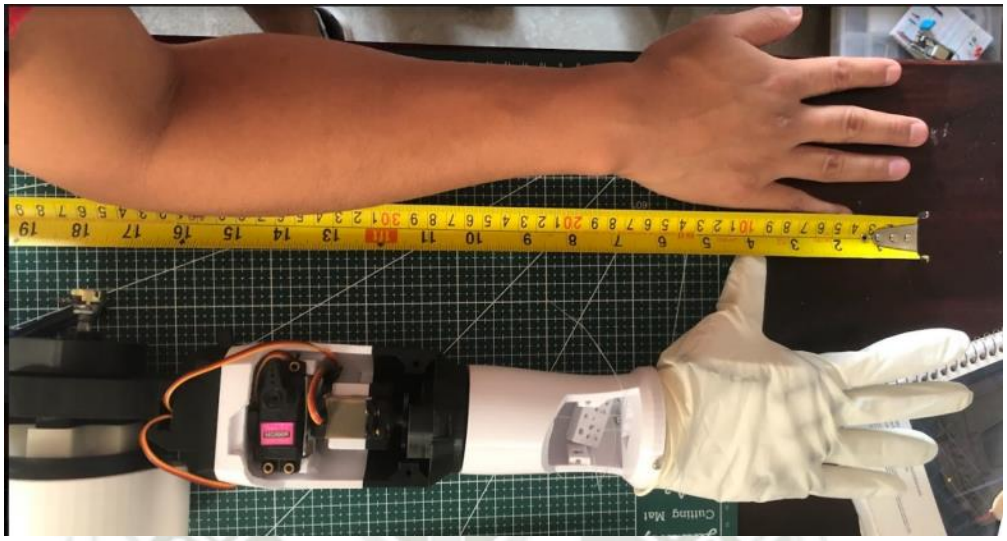


***Figura 111.*** *Dimensión de las piezas que componen el brazo y mano de la prótesis.*

Por otro lado, en la figura 112 se presenta la dimensión del motor-reductor y el codo, el cual es de 16 cm. En ensamble final de la prótesis, se muestra en la figura 113, donde la imagen del inciso (a) muestra una vista lateral de la prótesis, y la imagen del inciso (b) una vista frontal.



***Figura 112.*** *Dimensión de motor-reductor y codo.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA




(a) (b)

***Figura 113.*** *Prótesis ensamblada.*

### *4.2.5. Brazalete*

Una vez construido el prototipo de la prótesis, se continuó con la fabricación de un brazalete para colocar los electrodos. La figura 114, muestra las piezas fabricadas para colocar los conectores de electrodos, estas piezas son unidas por medio de un tornillo M3x40.



(a) (b)

***Figura 114.*** *Piezas que componen el brazalete.*

La figura 115, muestra los conectores de electrodos colocados sobre el brazalete. En estos se colocarán los electrodos encargados de captar las señales mio-eléctricas.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

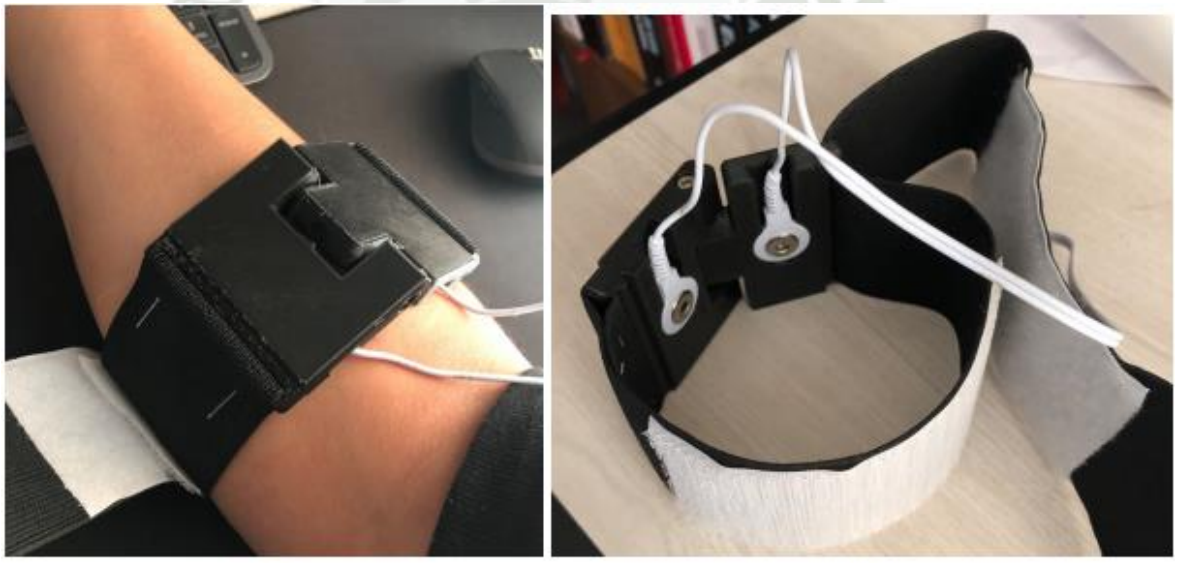
REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA



***Figura 115.*** *Conectores de electrodos colocados sobre el brazalete.*

La figura 116, muestra el brazalete construido, donde la imagen del inciso (a) muestra el brazalete colocado sobre el brazo, mientras que, la imagen del inciso (b) muestra el brazalete.



(a) (b)

***Figura 116.*** *Brazalete construido.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

# CAPÍTULO V

## 11. PRUEBAS Y RESULTADOS

### 5.1. Pruebas del funcionamiento

A continuación, se presentan las pruebas de funcionamiento que se realizaron a la prótesis.

#### *5.1.1. Conexión Prótesis y computadora*

En las pruebas que se hizo fue como primer paso, instalar los programas Thonny y Anaconda Spyder que nos servirán para la toma de señales mio-eléctricas y nos servirán como un entorno de desarrollo, una vez instalados los programas procedemos a conectar la ESP32 de la prótesis con la computadora por medio de un cable con entrada USB a tipo micro USB, conectamos la ESP32 a la computadora y para que nos reconozca la tarjeta se debe instalar el driver del CP210x que es un convertidor de USB a TTL de la ESP32. Así se podrá reconocer la tarjeta en la computadora, en la figura 116 se muestra los programas y el driver para la conexión de la ESP32 con la computadora.




***Figura 117.*** *Programas y Driver para reconocimiento de ESP32.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Una vez hecho esto se procede abrir los programas ya mencionados y descargamos la programación realizada. Revisamos y ajustamos el intérprete en el programa Thonny; luego escogemos *Micropython ESP32* y revisamos en que puerto serial se comunicará la tarjeta, como se visualiza en la figura 118. Una vez hecho esto se procederá a tomar las señales mio-eléctricas.

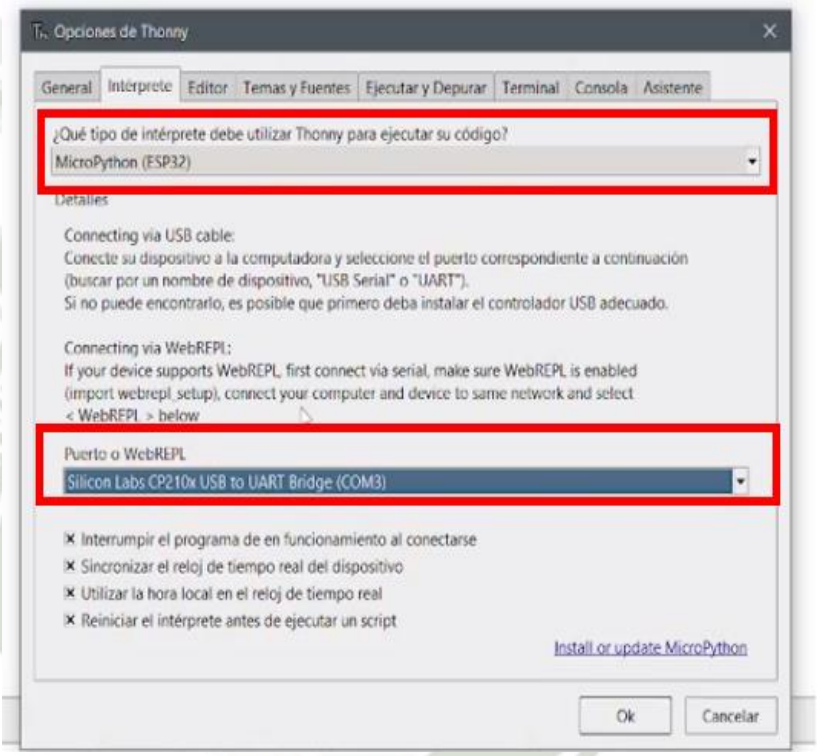



***Figura 118.*** *Ajuste de interprete y conexión del puerto en el programa Thonny.*

### *5.1.2. Adquisición de señales mio-eléctricas*

En la Adquisición de las señales mio-eléctricas se procede a ejecutar el programa principal y se procederá a comentarlo para así poder tomar las señales mio-eléctricas, seguidamente se abrirá el archivo de entrenamiento donde ubicaremos los archivos (Adquisición de señales y Adquisición) el primero se abrirá en el programa Thonny, mientras que el segundo lo abriremos en el programa Spyder, como se muestra en la figura 119.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

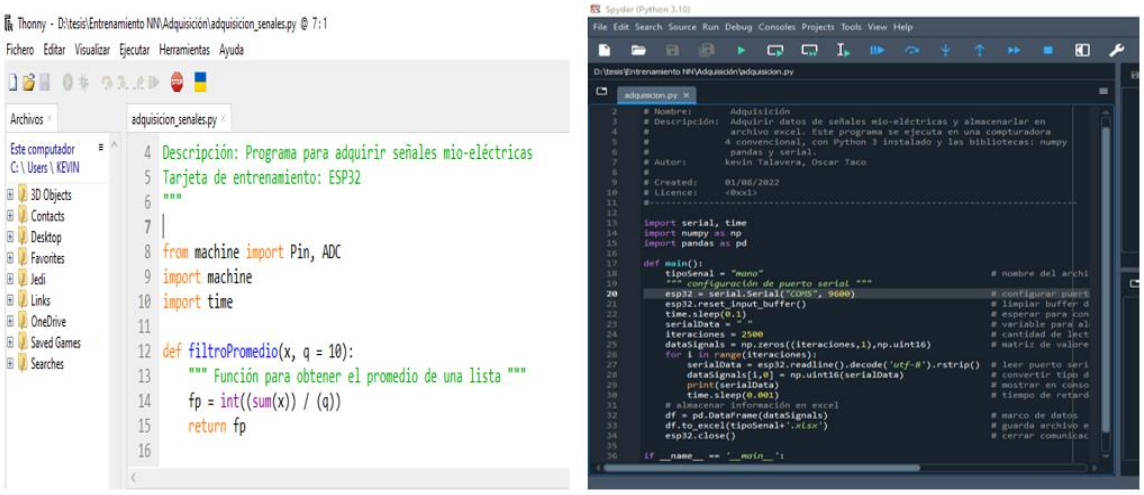



***Figura 119.*** *Programas Thonny y Spyder para la adquisición de datos.*

Después de haber abierto los programas ya mencionados, procedemos a sincronizar el Bluetooth de la prótesis con el de la computadora, para obtener las señales mio-eléctricas. Una vez sincronizado verificamos en que puerto se encuentra conectado el módulo para poder así cambiar el puerto en el programa, después de ello se procede a ejecutar los programas y a tomar las señales mio-eléctricas, los datos se guardan en archivos Excel dependiendo el dato que se haya tomado ya sea (mano, muñeca, codo).

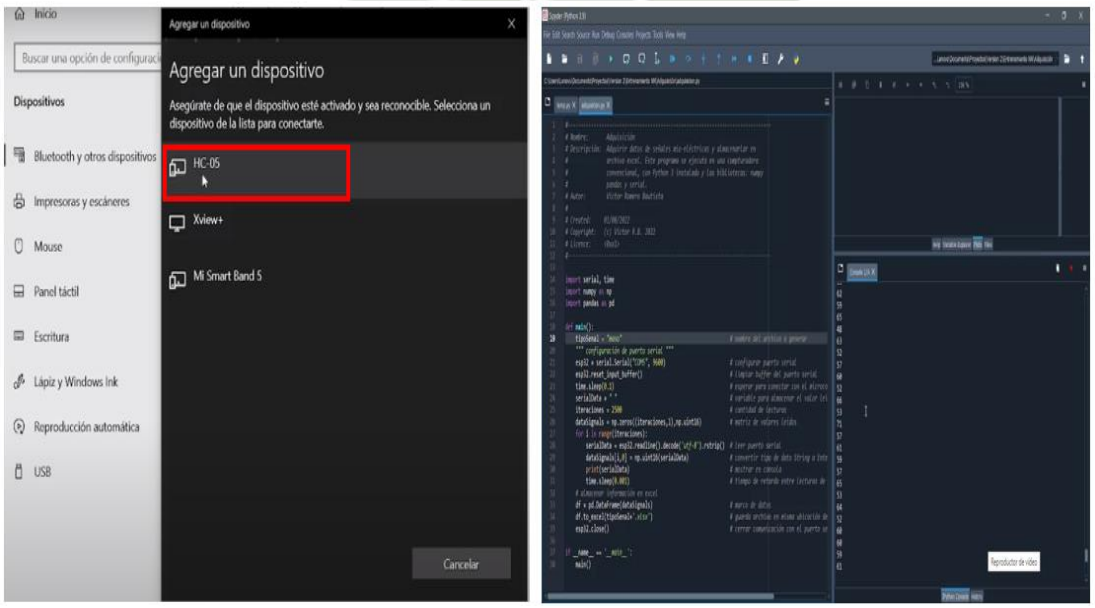



***Figura 120.*** *Sincronización de Bluetooth y Adquisición de señales en Programa Spyder.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

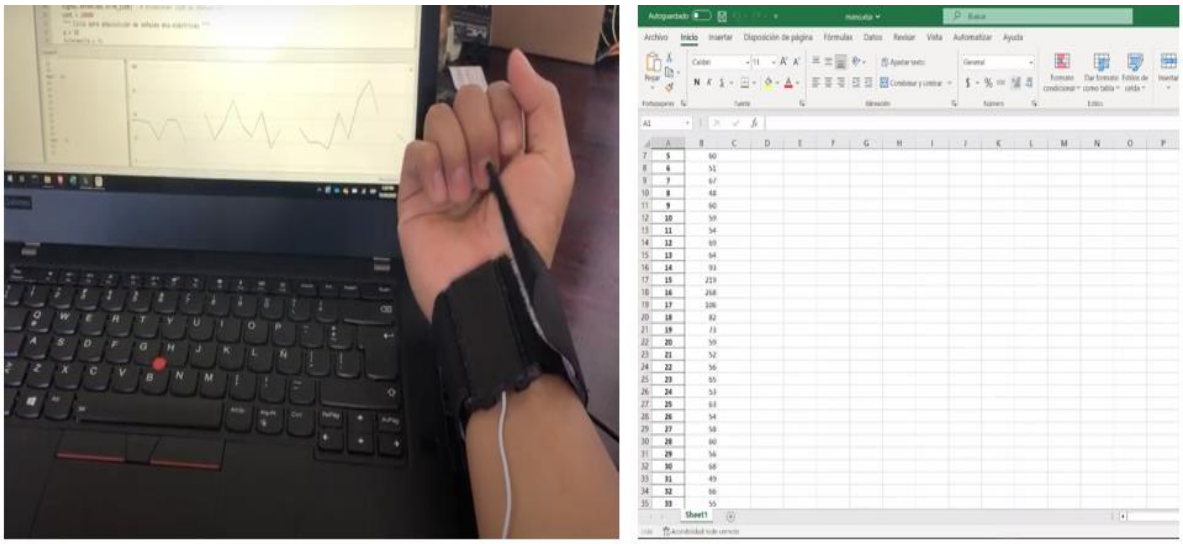


***Figura 121.*** *Adquisición de señales y datos en archivo Excel.*

### *5.1.3. Procesamiento de datos para las señales*

Después de haber obtenido los datos tomados de las señales mio-eléctricas lo que se hace es procesar las señales en Spyder, tomamos los datos de las señales obtenidas (mano, muñeca, codo). Luego colocamos en el programa que datos queremos procesar una vez hecho esto se procede a ejecutar el programa. En el cual obtendremos las gráficas procesadas y sin procesar, así como también las tablas de distribución de frecuencias. Las cuales nos servirán para sacar un promedio de los umbrales mínimos y máximos que se obtuvieron en la adquisición de señales tomadas anteriormente. Se compara la distribución de frecuencia con los datos tomados y se saca un promedio entre ellas para poder tener una señal procesada como se muestra en la figura 123.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

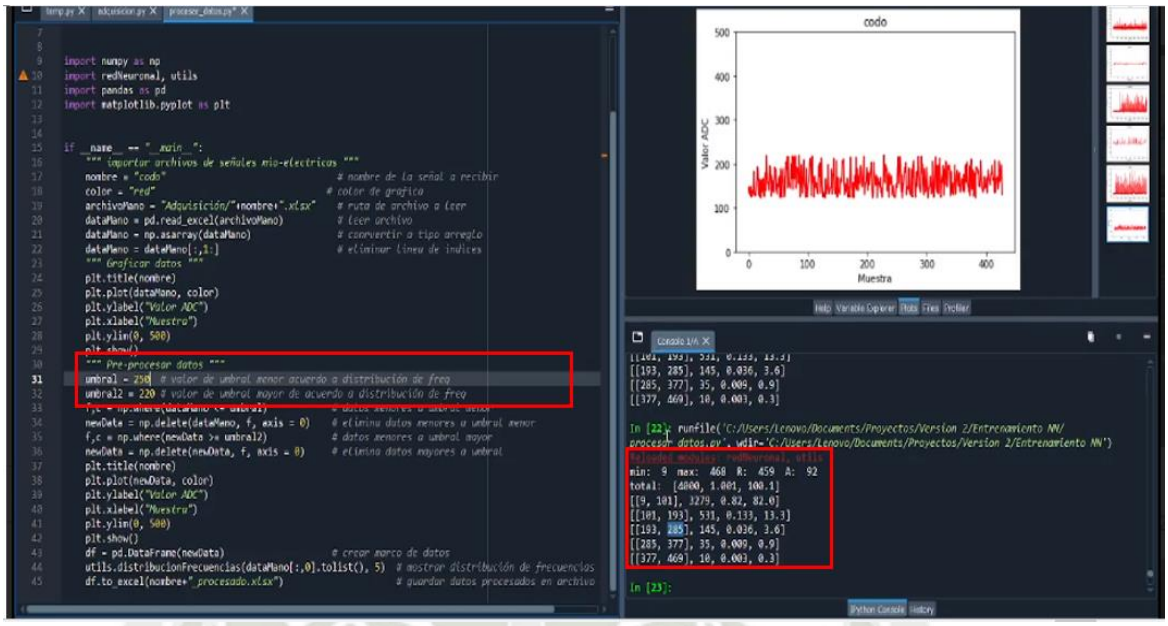



***Figura 122.*** *Adquisición de la señal sin procesar.*

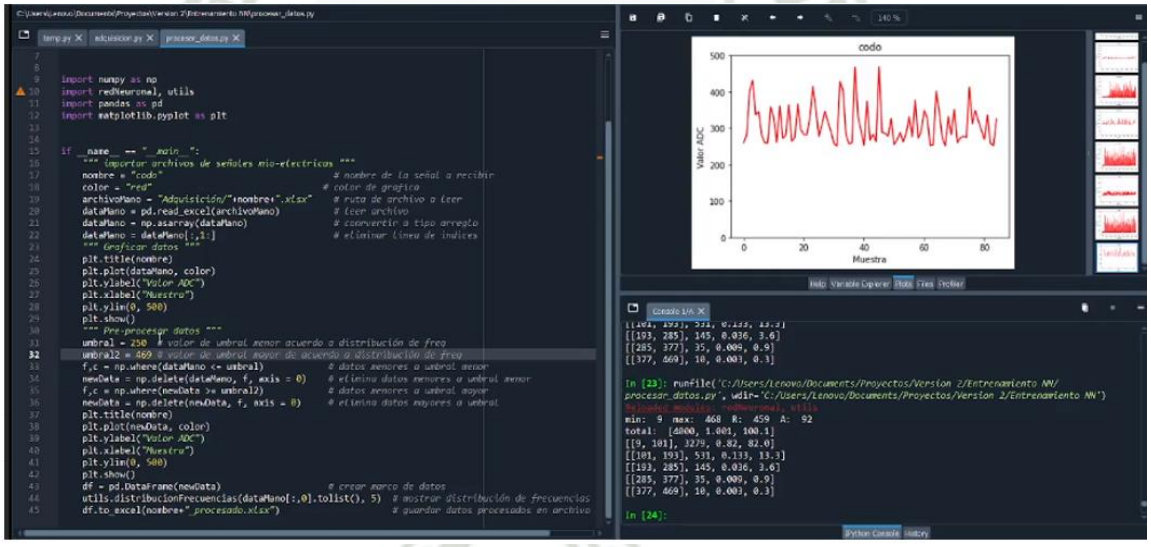



***Figura 123.*** *Adquisición de la señal procesada.*

### *5.1.4. Pruebas de la etapa de aprendizaje*

Una vez obtenidos los archivos ya procesados los cuales contienen las señales mio-eléctricas de cada parte de la prótesis (mano, muñeca, codo). Se procederá a realizar la etapa de aprendizaje, en la cual se mandará la cantidad de neuronas, el factor de aprendizaje y el error mínimo. En el aprendizaje se ejecutará en el programa de Spyder y se modificará los datos que obtenemos en el programa de Thonny.

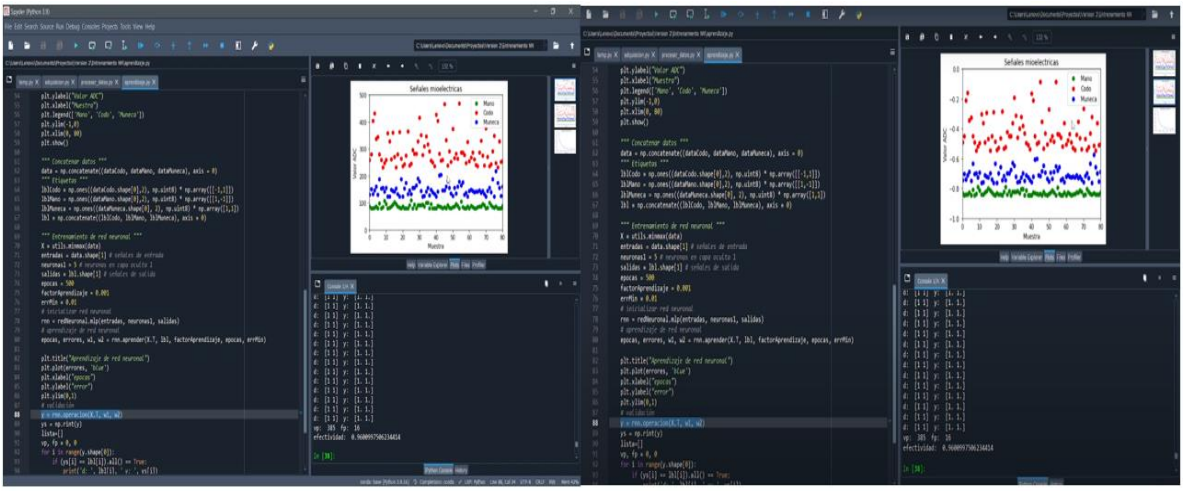


Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA

Después de haber ejecutado el programa como resultado se obtendrá los gráficos con datos no normalizados, datos normalizados y la gráfica del error de la red neuronal. Como se muestra en la siguiente imagen.



(a) No normalizados (b) Normalizados

***Figura 124.*** *Gráficas de aprendizaje (a)No normalizados (b)Normalizados.*

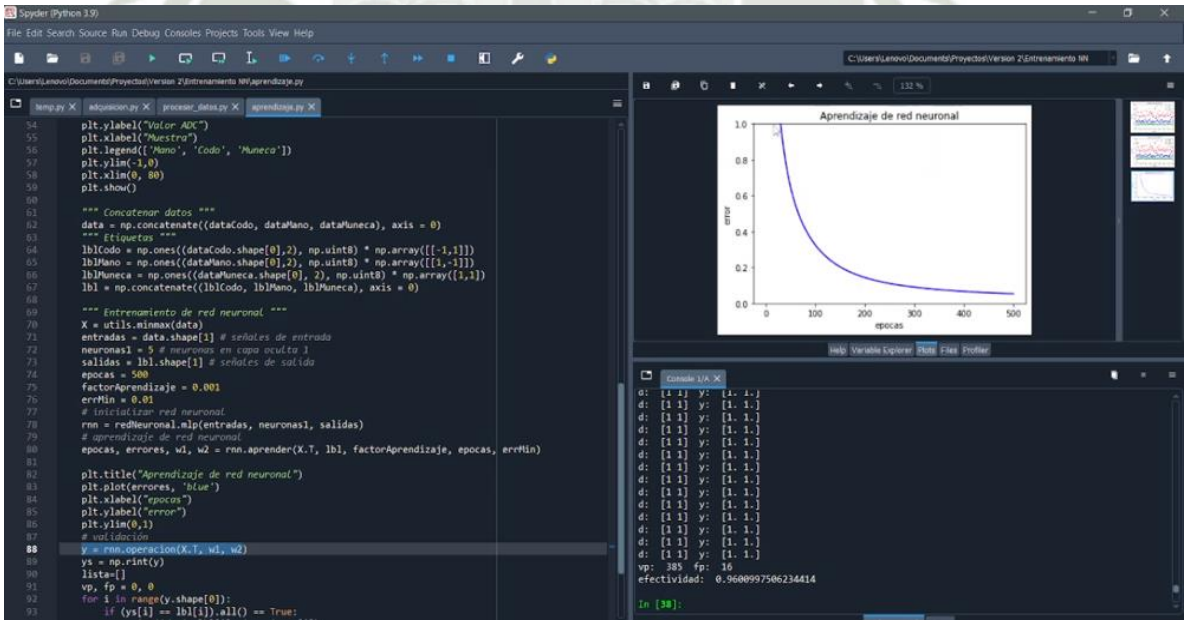



***Figura 125.*** *Gráfica del error del aprendizaje.*

Por último, vemos los resultados del aprendizaje que nos dará los valores verdaderos positivos y los falsos positivos, medimos esto para ver la efectividad de la red neuronal de cuantos movimientos correctos hará y cuantos errores tendrá, realizado esto procedemos a



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

tomar los pesos sinápticos que se generó en el aprendizaje de la red neuronal; los cuales serán modificados en el programa Thonny en la carpeta red neuronal y se guardarán para así poder correr el programa principal con los datos que se tomó.

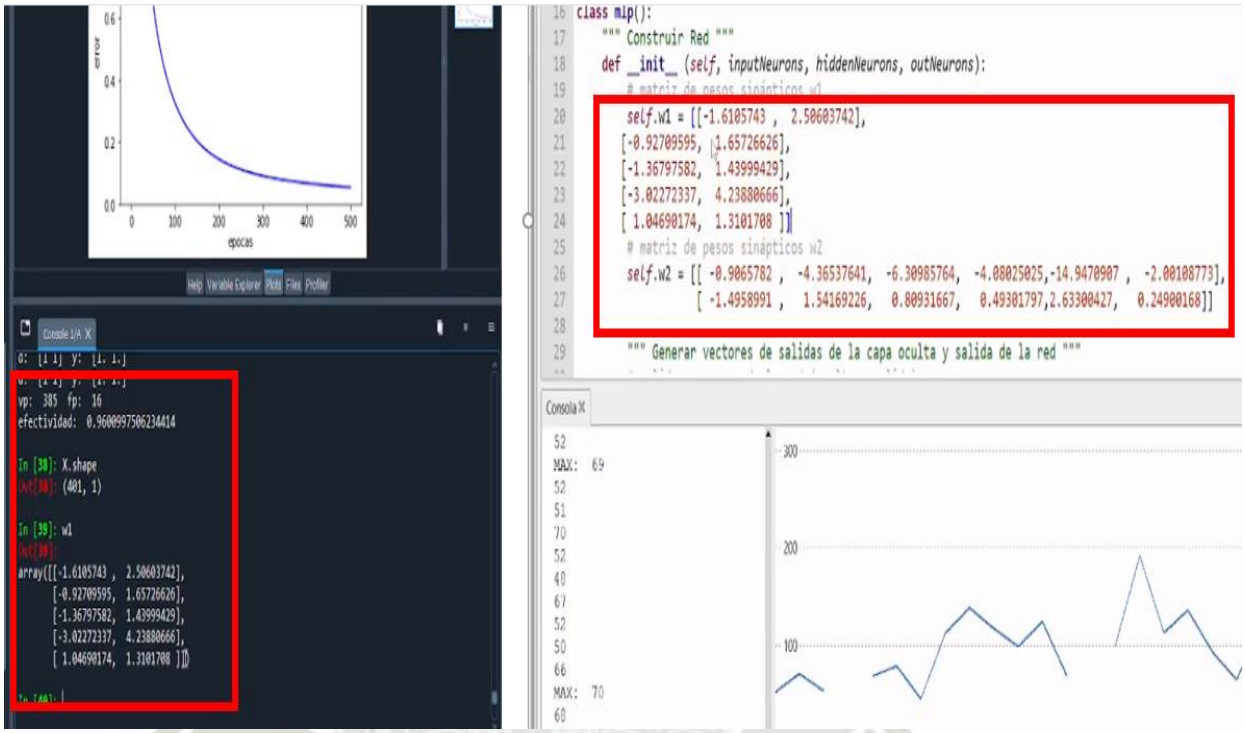



***Figura 126.*** *Toma de los pesos sinápticos.*

## 5.2. Resultados del funcionamiento

A continuación, se presentan los resultados de las pruebas de funcionamiento realizadas.

En la figura 127, se muestran los resultados de las pruebas de funcionamiento de la mano. En la imagen del inciso (a), se muestra la mano abierta, mientras que, en el inciso (b), se presenta la mano cerrada. Estos movimientos son generados cuando se realiza el movimiento de abrir y cerrar la mano.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE
TESIS UCSM

UNIVERSIDAD
CATÓLICA
DE SANTA MARÍA



(a)

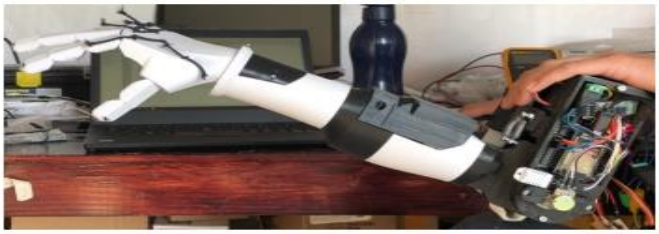


***Figura 127.*** *Resultado de movimientos de la mano.*

La figura 128, presenta los resultados obtenidos al mover el dedo anular, la señal producida al efectuar este movimiento hace que se accione el motor que mueve la muñeca de la prótesis. La imagen del inciso (a), presenta la posición de pronación de la muñeca, mientras que, la imagen del inciso (b), muestra la posición de supinación de la muñeca.



(a)



(b)

***Figura 128.*** *Resultado de movimientos de la muñeca.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

La figura 129, presenta los resultados obtenidos al efectuar los movimientos de contracción y flexión de la mano, lo cual genera que la prótesis efectúe los movimientos correspondientes al codo, que se puede mover en un ángulo de 40 grados para flexión y contracción.

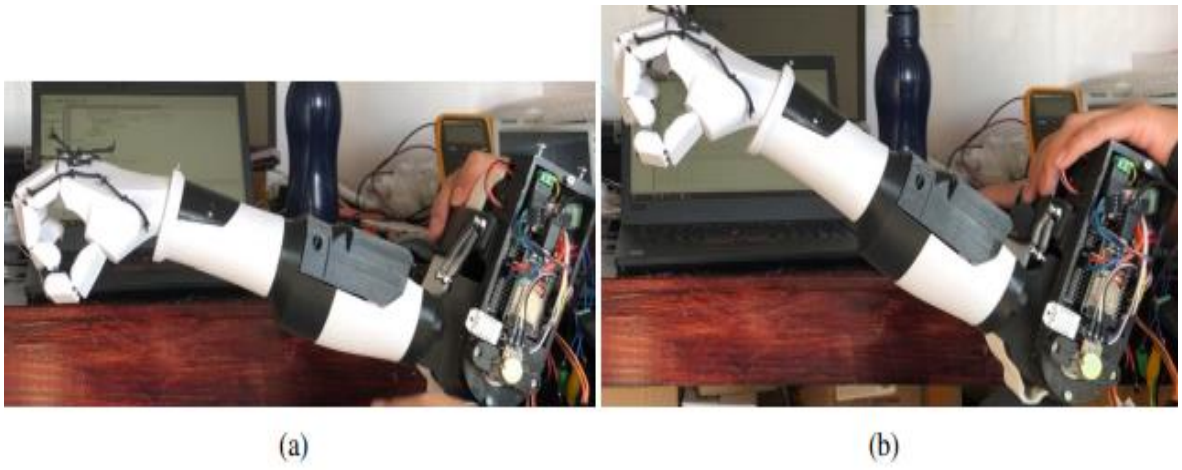



***Figura 129.*** *Resultado de movimientos del codo.*

Finalmente se muestran los resultados del aplicativo que se conecta vía Bluetooth, una vez conectado se ejecutará en el móvil. La imagen del inciso (a), presenta el nombre del dispositivo Bluetooth, mientras que, la imagen del inciso (b), muestra cómo se encuentra el nombre del Bluetooth cuando se ingresa desde el aplicativo y en el inciso (c) vemos ya ejecutándose el aplicativo en el móvil.




***Figura 130.*** *Interfaz de la aplicación Android conectándose y ejecutándose.*



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### 5.3. Indicaciones de uso

- Conexión de la tarjeta ESP32 con la computadora.
- Conexión vía Bluetooth del módulo con la computadora.
- Conexión del aplicativo con el Bluetooth.
- Se coloca las pulseras correctamente en el brazo.
- Se abren los programas Thonny y Spyder para el entrenamiento de la prótesis.
- Se abre la carpeta con el programa adquisición de datos y adquisición.
- Se procede a tomar las señales mio-eléctricas de cada sección (mano, muñeca, codo).
- Se abre la carpeta procesamiento de datos, con el programa Spyder y se comienza a procesar los datos obtenidos.
- Se abre la carpeta de aprendizaje con los programas Thonny y Spyder, donde se ajustarán los pesos sinápticos para tener la eficacia de los movimientos de la prótesis.
- Se realizan los movimientos de la prótesis que se realizaron en la etapa de entrenamiento como, por ejemplo: agarrar objetos, mover la muñeca en 90° y flexión y contracción del codo.

### 5.4. Especificaciones del prototipo

- Peso de la prótesis, aproximadamente 3 kg.
- Voltaje de alimentación de las dos baterías 9 voltios en 500mA.
- Consumo de corriente máximo para las baterías de los motores es de 500mA.
- Consumo de la batería que alimenta el ESP32 y el circuito de adquisición de señales es de 200mA.
- Tiempo de autonomía, aproximadamente de 2 a 5 horas, dependiendo del uso de los motores.
- Peso máximo a cargar 300 gramos.
- Área de agarre, la circunferencia de una lata de gaseosa.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

***Tabla 14*** *Especificaciones tecnicas del equipo*

| Especificaciones del equipos | |
|---|---|
| Voltaje de alimentación de las baterías | 5Vdc a 9Vdc |
| Voltaje de alimentación de motores | 9Vdc |
| Voltaje de alimentación de la ESP32 | 3.3V a 3.6V |
| Consumo de corriente de las baterías para motores | 500mA |
| Consumo de la batería que alimenta la ESP32 | 80mA a 200mA |
| Periodo de trabajo de servomotores | 20ms |
| Periodo de trabajo del Motoreductor | 490Hz |
| Frecuencias de trabajos de los filtros | 20hz a 250Hz |
| Frecuencias de paro (*fp*) de los filtros | 2Hz a 2500Hz |
| Frecuencia de corte (*Ac*) | 3dB |
| Frecuencia de paro (*Ap*) | 60dB |
| Amplificación de señal | 0.1 a 6mV |
| Ancho de banda BW | 10Hz |
| Rango de ganancia | Entre 1 y 10000 |
| Impedancia de entrada | 10 GΩ |
| Acondicionador de señales | 3 electrodos, 2 de señales y uno de referencia |
| Módulo de control | ESP32 |
| Módulo de transmisión | Bluetooth HC06 |
| Peso de la prótesis | 3kg |
| Peso máximo de levante | 300 gramos |
| Temperatura de trabajo | 25°C a 55°C |
| Material del equipo | PLA |
| Switch de conexión | Si |
| Señal en reposo | 0.01 a 5 mV |
| Ángulos de movimiento mano | 0°a 90° |
| Ángulos de movimiento muñeca | 0° a 180° |
| Ángulos de movimiento codo | 0° a 180° |
| Adquisición de datos mano | 50Hz a 65 Hz |
| Adquisición de datos muñeca | 70Hz a 110 Hz |
| Adquisición de datos codo | 150Hz a 300 Hz |
| Grados de libertad | 3 grados de libertad |
| Tiempo de autonomía | 2 a 5 horas |



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

### 5.5. Presupuesto de la prótesis

***Tabla 15*** *Presupuesto de la prótesis.*

| Nombre | Cantidad | Precio soles | Descripción |
|---|---|---|---|
| Tarjeta ESP 32 | 1 | 40 | NodeMCU 30 pines |
| Shield ESP32 | 1 | 20 | Tarjeta de expansión para ESP32 |
| Servomotor | 1 | 30 | MG996R DE 11 KG |
| Servomotor | 1 | 70 | TD8120 20 KG |
| Motoreductor | 1 | 100 | - |
| Amplificador | 1 | 1 | TL084 |
| Amplificador | 1 | 8 | AD620 amplificador de instrumentación |
| Puente H | 1 | 19 | Módulo L298D |
| Bluetooth | 1 | 25 | Módulo HC06 |
| Sensor | 1 | 9 | Sensor de temperatura DTH11 |
| Cable USB | 1 | 10 | Cable USB a micro USB |
| Baterías | 2 | 210 | Baterías de 9 voltios |
| Rollo PLA | 3 | 155 | Rollo de PLA para impresora 3D |
| Placa perforadora | 2 | 10 | Placas para los componentes |
| Componentes | 1 | 20 | Resistencias, condensadores, borneras. |
| Potenciómetro | 1 | 1 | - |
| Pulsador | 1 | 3 | - |
| Conectores de electrodos | 3 | 9 | 3M |
| Rodamientos | 1 | 8 | 6203 |
| Cables de conexión | 1 | 20 | - |
| Electrodos | 30 | 90 | 3M |
| Cinta elástica | 1 | 15 | 50 milímetros |
| Pega pega | 1 | 2 | 50 milímetros |



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

| | | | |
|---|---|---|---|
| Impresión 3D | 100 | 1300 | Gasto energético, tiempo, diseño |
| Ferretería | 1 | 20 | Tornillos, cintillos, cordón, nylon |
| Mano de obra ensamblaje | | 160 | Ensamblaje |
| Mano de obra programación | | 1110 | Programación |
| Mano de obra Diseño mecánico | | 700 | Diseño |
| **Total** | | **S/4165** | |



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

# CONCLUSIONES

Se propuso un diseño e implementación de un sistema robótico biomédico para controlar los movimientos de una prótesis mio-eléctrica transhumeral, con el objetivo de captar las señales mio-eléctricas y podamos realizar movimientos de la prótesis.

Con la realización de la prótesis se concluye lo siguiente:

1. Teniendo en cuenta el primer objetivo, realizó el diseño y fabricación del prototipo compuesto por tres partes principales, las cuales son: mano, muñeca y codo.
2. También en relación al primer objetivo se logró diseñar e implementar una prótesis mio-eléctrica para personas de estatura promedio mayores a 18 años.
3. Se logró implementar el circuito de adquisición de señales permite capturar la diferencia de potencial generada al contraer los músculos a través de una correcta colocación de los electrodos, cumpliendo con este el segundo objetivo.
4. Se realizó 10 pruebas de entrenamiento donde se obtuvo una efectividad de 79% de las pruebas realizadas, este hace referencia al tercer objetivo
5. En relación al cuarto objetivo logró realizar los tres grados de libertar de la prótesis, donde la mano tiene la capacidad de abrir y cerrar los dedos actuando como una pinza para de esta manera poder tomar objetos; la muñeca, tiene la capacidad de ejecutar los movimientos de pronación y supinación, con ángulo de trabajo de 90 grados. El codo tiene la capacidad de ejecutar los movimientos de flexión y contracción, con un ángulo de trabajo de 180 grados.
6. El prototipo puede desempeñar acciones como: tomar y levantar objetos de tamaño pequeño con un peso no mayor a 300 gramos, cumpliendo así el quinto y sexto objetivo.
7. Con relación al sexto objetivo, la prótesis trabaja a un 75 % de la carga real de la batería, cuando es inferior al 75% los movimientos de la prótesis no se realizarán adecuadamente.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

8. Se observó que la prótesis realizaba movimientos cuando se filtraba algún ruido por equipos prendidos, se debe evitar movimientos involuntarios. Cuando no se realiza un buen entrenamiento de los movimientos de la prótesis, esta se mueva sola sin mandar algún movimiento.
9. Se pudo observar que algunas señales se parecían mucho, por ello al tomar las señales se debía buscar los movimientos que se puedan diferenciar. Es posible también que las señales contengan ruido que no pueda ser eliminado por el circuito de acondicionamiento de señales.
10. Se pudo notar que al momento de usar el cargador de laptop y se quería tomar las señales mio-eléctricas este hacia que emitiera ruido constante.
11. En relación al séptimo objetivo, se logró diseñar una aplicación para el monitoreo de la prótesis, la cual muestra la información general de la prótesis como son: la temperatura y humedad, nivel de carga de la batería y estado de los motores.
12. Finalmente, se pudo ver que el aplicativo no puede ser usado con otro sistema operativo que no sea Android.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## RECOMENDACIONES

1. Se recomendaría buscar un circuito que pueda eliminar por completo el ruido para que así la prótesis pueda tener un mejor comportamiento al momento de la manipulación.
2. Se podría mejorar la fuerza de la prótesis usando motores de alto torque para que esta pueda cargar objetos más pesados sin alterar el diseño de la prótesis.
3. Se podría añadir el movimiento de los dedos buscando las mejores señales mio-eléctricas y que este no altere el diseño de la prótesis, es decir que se vuelva más robusto.
4. Se recomendaría buscar otras redes neuronales que puedan ayudar a mejorar la efectividad de respuesta de la que ya se obtiene con la red perceptrón multicapa.
5. Para el aplicativo Android se podría mejorar las interfaces y este tenga una versión mejorada como un aplicativo moderno.
6. Se podría intentar monitorear la prótesis por medio de relojes inteligentes.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM
UNIVERSIDAD CATÓLICA DE SANTA MARÍA

## REFERENCIAS

Carhuavilca, D., & INEI. (Julio de 2020). *Perú: Estadísticas de las personas con alguna discapacidad. Presentación ante la Comisión de Inclusión Social y Personas con Discapacidad.* Obtenido de Congreso de la República: https://www.congreso.gob.pe/Docs/comisiones2020/InclusionSocialDiscapacidad/files/presentaciones_ppt/poblaci%C3%B3n_con_alguna_discapacidad_20_julio_de_2020.pdf

Valenzuela Urrutia, D. R. (2016). *Sistema de teleoperación basado en interfaz háptica para brazo robótico.* Chile: Universidad de Chile.

Aragón Martínez, A. (2016). *Diseño y Construcción de un guante Háptico con retroalimentación de fuerza.* México: Universidad Tecnológica de la Mixteca.

Mancipe Toloza, E. (2013). *Sistema de control de fuerza e interfaz háptica para un prototipo de prótesis electromiográfica de mano.* Bucaramanga, Ecuador: Universidad Pontificia Bolivariana.

Romero Muñiz, E. F. (2018). *Diseño de un sistema de retroalimentación háptico para una prótesis mioeléctrica transradial de miembro superior.* Lima, Perú: Universidad Pontificia del Perú.

Llantoy Sánchez, A. (2020). *Diseño e implementación del sistema eléctrico para una prótesis transradial mioeléctrica.* Lima, Perú : Pontificia Universidad Católica del Perú.

Sullcahuamán Jáuregui, B. (2013). *Diseño mecánico de un prototipo de prótesis mioeléctrica transradial.* Lima, Perú: Pontificia Universidad Católica del Perú.

Huaroto, L. (2000). *Cirugía Ortopédica y Traumatológica.* Lima.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Rostro, L. (2021). *Udocz.* Obtenido de Udocz: https://www.udocz.com/apuntes/89759/fisiologia-articular-antebrazo

Verastegui Rodriguez , C. F., & Flores Cuenca, M. F. (2019). *Diseño y Fabricación de Prótesis Miolectrica de mano de 5 grados de libertad de bajo costo para personas con amputación transradial* . Callao: Universidad Nacional del Callao.

Medina Gonzales, C., Benet Rodríguez, M., & Marco Martínez, F. (2016). El complejo articular de la muñeca: aspectos anatofiológicos y biomecánicos, caracteristicas, clasificación y tratamiento de la fractura distal del radio. *Scielo, 14*(4). doi:http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1727-897X2016000400011

Drake L., R., Wayne Vogl, A., & M. Michell, A. (2015). *Gray. Anatonomía para estudiantes.* Elsevi.er.

Pérez, D., & Aguilar , T. (2014). *UF0677: Soporte vital básico.* Elearning S.L. doi:https://books.google.com.pe/books?id=nXxXDwAAQBAJ&pg=PA87&dq=definici%C3%B3n+de+amputaci%C3%B3n&hl=es-419&sa=X&ved=2ahUKEwjckc3dh871AhXfSjABHSesALkQ6AF6BAgIEAI#v=onepage&q=definici%C3%B3n%20de%20amputaci%C3%B3n&f=false

Pérez, J. (2004). *Manual de Fisioterapia Módulo III. Traumatología. Afecciones cardiovasculares. Otros campos de actuación.* Mad, S.L. doi:https://books.google.com.pe/books?id=95nPrEWfCY8C&pg=SL20-PA332&dq=definici%C3%B3n+de+amputaci%C3%B3n&hl=es-419&sa=X&ved=2ahUKEwjckc3dh871AhXfSjABHSesALkQ6AF6BAgDEAI#v=onepage&q=definici%C3%B3n%20de%20amputaci%C3%B3n&f=false



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

MayoClinic. (29 de octubre de 2020). *Dolor fantasma*. Obtenido de https://www.mayoclinic.org/es-es/diseases-conditions/phantom-pain/symptoms-causes/syc-20376272

OMS. (2017). *Normas de Ortoprotésica Parte 1: Normas*. Obtenido de https://apps.who.int/iris/bitstream/handle/10665/259508/9789243512488-part1-spa.pdf;sequence=1

Dorador, J. (2004). Robótica y prótesis inteligentes. *Revista Digital Universitaria, 6*(1), 2-15. doi:http://www.revista.unam.mx/vol.6/num1/art01/art01_enero.pdf

Brito, J., Quinde, M., Cusco, D., & Calle, J. (2013). Estudio del estado del arte de las prótesis de mano. *Revista de Ciencia y tecnología Ingenius* (9), 57-64. doi:file:///C:/Users/USUARIO/Downloads/Dialnet-EstudioDelEstadoDelArteDeLasProtesisDeMano-5972783.pdf

Loaiza, J., & Arzola, N. (2011). Evolución y tendencias en el desarrollo de prótesis de mano. *Portal de Revistas UN - DYNA, 78*(169), 191-200. doi:https://revistas.unal.edu.co/index.php/dyna/article/view/19354/26525

Gonzales Badillo, Germánico; Torres Urestí, David; Espinoza Lopez, Victor; Guerreo Mora, Guillermina. (2018). control de una protesis de mano fabricada por impresión 3D utilizando señales electromiográficas y lógica difusa. *Memorias del XXIV Congreso Internacional Anual de la Somin* (págs. 27-36). Mexico: Universidad Autónoma de San Luis Potosi. Obtenido de http://somim.org.mx/memorias/memorias2018/articulos/A1_124.pdf

Raya, L. (2015). *Visión global sobre tecnología háptica.* Obtenido de Asociación de Autores Científicos Tecnicos y Académicos ACTA: https://www.acta.es/medios/articulos/ciencias_y_tecnologia/061115.pdf



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Mainieri Hidalgo, A., & Méndez Barrantes, Z. (2005). *Antología. Detección de problemas de aprendizaje.* San José, Costa Rica: Editorial EUNED.

Zarza, C. (2014). *Propiocepción y control neuromuscular en el fútbol infantil.* Obtenido de Universidad Fasta: https://core.ac.uk/download/pdf/49225148.pdf

Pierre, T., & Méndez de Thomas, Z. (2010). *Psicología del niño y aprendizaje* . San José, Costa Rica: Editorial EUNED.

Palacio Velasquez, P. (2013). *Diseño e Implementación de un fonocardiógrafo digital para el laboratorio de Biolectrónica.* Institución Universitaria Pascual Bravo.

Orts Blazquez, J. (2019). *Diseño de un sistema de adquisición de señales por electromiografia superficial para la interpretación del movimiento de la mano.* Universidad Politecnica de Valencia.

STMicroelectronics. (1999). *Electronic Components Datasheet Search*. Obtenido de https://pdf1.alldatasheet.com/datasheet-pdf/view/25385/STMICROELECTRONICS/TL084.html

Torrente Artero, O. (2013). *Electrónica Básica.* RC libros. doi:eek.com/upload/libros/KIT-0011/cap1_Arduino.pdf

Electronilab. (s.f.). *electronilab.co*. Obtenido de https://electronilab.co/tienda/servomotor-mg996r-11kg-4-8-v-tower-pro-pinoneria-metalica/

Topdoctors. (2018). *¿Qué son las amputaciones?* Obtenido de https://www.topdoctors.es/diccionario-medico/amputacion

Munive, M. (2009). *Introducción al curso de anatomía y la fisiología.* México: Prentice Hall.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

García, D., & Vázquez , G. (2010). *Anatomofisiología de las articulaciones. Traumatismos articulares. Complicaciones de los traumatismos. Rigideces, anquilosis.* Cursocot.

Padilla, A. (2010). Discapacidad: contexto, concepto y modelos. *Rev. Colombiana Derecho Internacional*.

Fisioonline. (2019). *Flexión articular*. Obtenido de https://www.fisioterapia-online.com/glosario/flexion-articular

Israel, F. (21 de junio de 2020). *¿Qué es la tecnología háptica que incorporará el mando de PS5?* Obtenido de https://www.nobbot.com/tecnologia/que-es-tecnologia-haptica/

CCM Salud. (11 de enero de 2016). *Miembros superiores*. Obtenido de https://salud.ccm.net/faq/18979-miembros-superiores-brazos-definicion

Significados. (2021). *Qué es presión*. Obtenido de https://www.significados.com/presion/

Ortosur. (2018). *Cómo funciona y se ajusta una prótesis mioeléctrica*. Obtenido de https://www.ortosur.es/como-funciona-protesis-mioelectrica/

ACS. (1 de febrero de 2020). *Prótesis*. Obtenido de https://www.cancer.org/es/tratamiento/tratamientos-y-efectos-secundarios/efectos-secundarios-fisicos/protesis.html

Prototip0. (23 de septiembre de 2020). *¿Qué es un prototipo?* Obtenido de https://prototip0.com/diseno-de-prototipos/

EDS. (23 de abril de 2021). *¿Qué es la robótica?* Obtenido de https://www.edsrobotics.com/blog/que-es-la-robotica/

Westreicher, G. (23 de septiembre de 2020). *Sensibilidad*.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Pérez, M. (22 de diciembre de 2021). *Definición de texturas*. Obtenido de https://conceptodefinicion.de/textura/

Totora, J. (2021). *Amputación transhumeral*. Obtenido de https://es.scribd.com/document/449262135/AMPUTACION-TRANSHUMERAL

Rosenblatt, F. (1958). *The eperceptron: Ea eprobabilistic emodel for einformation estorage and reorganization in the brain*.

Nunes da Silva, I. (2017). *Artificial neural networks.* Springer: Cham.

Rumelhart, D. (1989). *Parallel distributed processing.* MIT Pr.

García, E. (2008). *Compilador C CCS y Simulador Proteus para Microcontroladores PIC.* Alfaomega.

GoConqr. (2022). *Articulación de la muñeca*.

OPS. (2020). *Discapacidad*.

Conadis. (2 de marzo de 2022). *Informe estadístico del Registro Nacional de la Persona con Discapacidad – Reporte enero 2022*.

INEI. (2017). *En el Perú existen 1 millón 575 mil personas con discapacidad*.

Farro, L., Tapia, R., Bautista, L., Montalvo, R., & Iriarte, H. (2012). Características clínicas y demográficas del paciente amputado. *Revista Médica Hered*, 240-243.

Ramos, O. R. (2008). *Simulación en simmechanics de un sistema de control difuso para el robot udlap.* México.

Munt, A. M. (2018). *Introducción a los modelos de redes neuronales artificiales el Perceptrón simple y multicapa.* España.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

REPOSITORIO DE TESIS UCSM

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

Juan Carlos Pérez Mesa, J. d. (2007). *Empleo de redes neuronales de base radial a un modelo econométrico de exportación de tomate.* México.

Pedro Isasi Viñuela, I. M. (2004). *Redes de neuronas artificiales un enfoque práctico.* España : Pearson Prentice Hall.

Egly Barrero Viciedo, B. F. (2014). Propuesta de procedimiento para configurar una red neuronal atificial de base radial con aplicaciones en el diagnostico de fallos. *RIELAC*, 60-75.

Ovando, F. E. (2019). *Desarrollo de un perceptrón multicapa en raspberry pi.* Valparaíso.



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

# ANEXOS

## Anexo 1 Programación Principal

```
# -*- coding: utf-8 -*-
"""
-------------------------------------------------------------------------
------
Nombre:       Principal
Descripción:  Programa para controlar prótesis
Autor:        Kevin Talavera Ormeño y Oscar Taco Medina

Created:     25/12/2022
Copyright: (c) Kevin Talavera y Oscar Taco 2022
Licence:     <0xx1>
-------------------------------------------------------------------------
------
"""

from machine import Pin, ADC
import machine, time, redNeuronal, utils, dht
from servo import Servo
from codo import Codo

def readTemp():
    """ Función para leer sensor DHT """
    global temp, hum
    temp = hum = 0
    try:
        sensor.measure()
        temp = sensor.temperature() # lectura de temperatura
        hum = sensor.humidity() # valor de humedad
        hum = round(hum, 2)
    except OSError as e:
        hum = None
        #print('Fallo al leer sensor')

"""def fallo(motor):
    # enviar info a IFTTT API
    motorA, motorB, motorC = 1,1,1
    if motor == "A":
        motorA = 0
    if motor == "B":
        motorB = 0
    values = {'value1':motorA, 'value2':motorB, 'value3':motorC}
    request_headers ={'Content-Type': 'application/json'}
    request =
urequests.post('https://maker.ifttt.com/trigger/Protesis_fallo/with/key/'
+api_key,json=values, headers=request_headers)
    #print(request.text)"""

def main():
    global temp, hum
    cont = 20000                                          # cantidad
de iteraciones
    q = 10                                                # muestras
para filtro promedio
```

```
    # Variables de flujo para leer señal mioelectrica
    tolerancia = 15                                          # tolerancia
para leer señal mioelectrica
    dataP = 10                                               # dato
anterior de señal mio-electrica (por defecto)
    senal = []                                               # lista para
almacenar señal mio-electrica
    count = 0                                                # contador
de iteraciones
    # Variable para control de lectura de temperatura
    tomaTemp = 0                                             # variable
para contar lecturas de temperatura
    readTemp()                                               # obtener
valor inicial de temperatura
    actualPosA = 100                                         # posición
inicial mano (abrir)
    actualPosB = 170                                         # posición
inicial muñeca (supinación)
    nivelBateria = (bateria.read()/1024)*100                 # nivel de
bateria
    motorMano, motorMuneca, motorCodo = 1, 1, 1              # estado
inicial de motores
    servoA.move(100)
    servoB.move(170)
    #print("Hola")
    time.sleep_ms(2000)
    # inicia funcionamiento
    for i in range(cont):
        senales = []                                         # lista para
almacenar muestras
        y = None
        for j in range(q):
            senales.append(signal.read())                    # generar
lista de señales
        data = utils.filtroPromedio(senales,q)              # aplicar
filtro promedio
        # identificar cambio en señal mio-electrica
        if (abs(data - dataP) > tolerancia):
            senal.append(data)                               # almacenar
dato actual en lista senal
            count += 1                                       #
incrementar cuenta
            if count > 4:
                x = utils.minmax(max(senal))                 # normalizar
señal
                xi = [bias, x]                               # vector
entrada a la red neuronal
                print("xi ", xi, "  s:", max(senal))
                y = rn.propagation(xi)                       # computar
salida de la red neuronal
                #print("x:", x, " y:",y, " y0:")
                count = 0                                    # re-
establecer contador
```

```
                senal = []                                          # limpiar
lista

        # accionar motores (mover articulaciones de la prótesis)
        if y is not None:
            if (utils.rint(y[0][0]) == 1) and (utils.rint(y[0][1]) == -
1):
                #print("Mano")
                val1 = hallSensor1.read()
# sensor Hall mano
                #actualPosA = moverServo(servoA, actualPosA, 35, 100)
# mover motor A (mano)
                if actualPosA == 100:
                    servoA.move(27)
                    time.sleep_ms(600)
                    actualPosA = 40
                else:
                    servoA.move(100)
                    time.sleep_ms(600)
                    actualPosA = 100
                # comparar lectura anterior de sensor Hall 1 con lectura
actual del mismo
                if (abs(hallSensor1.read() - val1)) > 15:
                    a =0
                    print("Motor A se ha movido")
                    motorMano = 1
                else:
                    print("Motor A falló")
                    motorMano = 0
                    #fallo("A")
# enviar información al correo

            elif (utils.rint(y[0][0]) == 1) and (utils.rint(y[0][1]) ==
1):
                print("Muñeca")
                val2 = hallSensor2.read()
# sensor Hall muneca
                #actualPosB = moverServo(servoB, actualPosB, 90, 170)
# mover motor B (muñeca)
                if actualPosB == 170:
                    servoB.move(90)
                    time.sleep_ms(600)
                    actualPosB = 90
                else:
                    servoB.move(170)
                    time.sleep_ms(600)
                    actualPosB = 170
                # comparar lectura anterior de sensor Hall 2 con lectura
actual del mismo
                if (abs(hallSensor2.read() - val2)) > 15:
                    print("Motor B se ha movido")
                    motorMuneca = 1
                else:
```

```
                    print("Motor B Falló")
                    motorMuneca = 0
                    #fallo("B")
# enviar información al correo

            elif (utils.rint(y[0][0]) == -1) and (utils.rint(y[0][1]) ==
1):
                print("Codo")
                codo.move()
# mover codo

        if tomaTemp >= 35:
            nivelBateria = int((bateria.read()/1024)*100)
            #uart.write(str(int(temp))+'-'+str(int(hum))+'-
'+str(nivelBateria))
            uart.write(str(int(motorMano))+'-'+str(int(motorMuneca))+'-
'+str(int(motorCodo))+'-'+str(int(temp))+'-'+str(int(hum))+'-
'+str(nivelBateria))
            readTemp() # actualizar valores de temperatura y humedad
            tomaTemp = 0 # reiniciar contador
            #print('temp: ', temp, ' hum: ', hum, ' bat: ', nivelBateria)
            print('ma: ',motorMano, ' mu: ', motorMuneca, ' co: ',
motorCodo, 'temp: ', temp, ' hum: ', hum, ' bat: ', nivelBateria)

        # enviar valor de posicion al serial
        dataP = data
        tomaTemp += 1
        time.sleep_ms(20)

if __name__ == "__main__":
    sensor = dht.DHT22(Pin(26))                          # crear objeto
DHT, conectado al puerto GPIO26
    uart = machine.UART(2, 9600)                         #
Configuración de puerto serial
    # Configuración ADCs
    signalPort = 34                                  # inicializa ADC1
en GPIO34, puerto para conectar señal de amplificador
    signal = ADC(Pin(signalPort))                    # crear objeto ADC
    signal.width(ADC.WIDTH_10BIT)                    # configurar
resolución a 10 bits
    signal.atten(ADC.ATTN_11DB)                      # establecer 11dB
de atenuación (0v a 3v)
    bateria = ADC(Pin(32))                           # inicializa ADC1
en el GPIO32
    bateria.width(ADC.WIDTH_10BIT)                   # configura ADC a
resolucion de 10 bits
    bateria.atten(ADC.ATTN_11DB)                     # establece
atenuacion de entrada ente 0.0v a 3.6v
    hallSensor1 = ADC(Pin(39))                       # inicializa ADC1
en el GPIO39
    hallSensor1.width(ADC.WIDTH_10BIT)               # configura ADC a
10 Bits
```

```
    hallSensor1.atten(ADC.ATTN_11DB)                    # configura ADC
para una lectura de 3.3v
    hallSensor2 = ADC(Pin(36))                          # inicializa ADC1
en el GPIO36
    hallSensor2.width(ADC.WIDTH_10BIT)                  # configura ADC a
10 Bits
    hallSensor2.atten(ADC.ATTN_11DB)                    # configura ADC
para una lectura de 3.3v
    # Configuración de puertos de entradas/salidas
    ENA = 12                                            # Puerto GPIO para
conectar Enable del L298
    IN1 = 14                                            # Puerto GPIO para
conectar IN1 del L298
    IN2 = 27                                            # Puerto GPIO para
conectar IN2 del L298
    POT = 33                                            # puerto GPIO para
conectar el potenciometro (ADC1)
    codo = Codo(ENA, IN1, IN2, POT)                     # incializar codo
    # Configuración de servomotores
    pinServoA = 15
    pinServoB = 4
    servoA = Servo(pinServoA)                           # inicializar
objeto servo
    servoB = Servo(pinServoB)                           # inicializar
objeto servo
    # Parámetros de red neuronal
    entradas = 2                                              # señales de
entrada
    neuronas = 5                                              # neuronas
en capa oculta
    salidas = 2                                               # salidas de
la red
    bias = -1                                                 # valor de
sesgo o umbral
    rn = redNeuronal.mlp(entradas, neuronas, salidas)         # crear
objeto de red neuronal

    main()                                              # ir a ciclo de
control principal
```

## Anexo 2 Programación Codo

```
# -*- coding: utf-8 -*-
"""
------------------------------------------------------------------------
------
Nombre:       Codo
Descripción:  Biblioteca para el control del motor empotrado en el codo.
Autor:        Kevin Talavera Ormeño y Oscar Taco Medina
Created:     20/08/2022
Copyright:   (c) Kevin Talavera y Oscar Taco 2022 |
Licence:     <0xx1>
------------------------------------------------------------------------
------
"""

# Bibliotecas
from machine import Pin, PWM, ADC
import time
from servo import Servo

# Objeto PID
class PID():
    def __init__(self, t0,t1,t2,y0,y1,u0,u1):
        tao = t1 - t0                   # parámetro tao
        T = (tao/4)-0.1                 # tiempo
        gam = t2 - t1                   # parámetro gamma
        kap = (y1-y0)/(u1-u0)           # parámetro kappa
        kp = (1.2*gam)/(kap*tao)        # constante proporcional
        ki = 2*tao                      # constante integral
        kd = 0.5*tao                    # constante derivativa
        self.a = kp                     # constante auxiliar a
        self.b = (kp*T)/(ki)            # constante auxiliar b
        self.c = (kp*kd)/(T)            # constante auxiliar c
        self.iT0 = 0                    # tiempo anterior
        self.eT0 = 0                    # error anterior
        self.vMax = 1023                # valor de salida máximo
        self.vMin = 300                 # valor de salida mínimo

    def computePID(self, signalIn, setPoint):
        eT = (setPoint - signalIn)      # error en tiempo T
        iT = self.b * eT * self.iT0     # termino integral
        dT = self.c * (eT-self.eT0)     # termino derivativo
        uT = iT+self.a*eT+dT            # salida PID
        if uT > self.vMax:
            uT = self.vMax
        elif uT < self.vMin:
            uT = self.vMin
        self.iT0 = iT                   # tiempo anterior
        self.eT0 = eT                   # error anterior
        return uT

# Función para mapear
def mapping(inVal, inMin, inMax, outMin, outMax):
```

```
    mapValue = ((inVal - inMin) * (outMax - outMin)) / ((inMax-inMin) +
outMin)
    return mapValue

# Objeto Codo
class Codo():
    def __init__(self, ENA, IN1, IN2, POT):
        enable = Pin(ENA)                           # puerto de senal de
salida pwm
        motorFreq = 490                             # frecuencia de trabajo
del motor dc
        self.motor = PWM(enable, motorFreq)         # objeto PWM
        self.in1 = Pin(IN1, Pin.OUT)                # puerto de salida 1
        self.in2 = Pin(IN2, Pin.OUT)                # puerto de salida 2
        potPort = Pin(POT)                          # puerto del
potenciometro
        self.pot = ADC(potPort)                     # objeto ADC
        self.pot.width(ADC.WIDTH_10BIT)             # resolucion de 10 bits
para lecturas de voltaje
        self.pot.atten(ADC.ATTN_11DB)               # 11 dB de atenuacion
(0 - 3.3v)
        self.actualPos = 1                          # posición actual del
codo (1: arriba, 0: abajo)
        self.setPoint = 50                          # posición a alcanzar
        self.pid = None                             # incializar
controlador PID vacío
        self.error = 5                              # error para alcanzar
setpoint

    def initPid(self):
        t0 = 0.1                                     # tiempo de arranque
inicial
        t1 = 0.2                                     # tiempo de inicio de
pendiente
        t2 = 1                                       # tiempo de maximo
impulso
        y0 = 10                                      # ángulo mínimo a
medir
        y1 = 100                                     # ángulo máximo a
medir
        u0 = 350                                     # señal pwm mínima
        u1 = 1023                                    # señal pwm máxima
        self.pid = PID(t0, t1, t2, y0, y1, u0, u1) # inicializar
controlador PID

    def activarMotor(self, outPid):
        self.motor.duty(round(outPid))              # activar ENABLE
        # si brazo está contraido, configurar salida para flexionar
        if self.actualPos == 1:
            self.in1.value(0)
            self.in2.value(1)
            #print("mover hacia abajo")
        # si brazo está flexionado, configura salida para contraer
```

```
        elif self.actualPos == 0:
            self.in1.value(1)
            self.in2.value(0)
            #print("mover hacia arriba")

    def apagarMotor(self):
        self.motor.duty(0)

    def move(self):
        potVal = self.pot.read()
        pos = round(mapping(potVal, 0, 1023, 0, 180))
        # identificar a que posición a moverser
        if pos <= 30:
            self.setPoint = 40                                  # setpoint
para bajar brazo (flexión del codo)
            self.actualPos = 1                                  # el brazo
se encuentra contraido (arriba)
            #print("está arriba", " p:", pos)
        elif pos > 30:
            self.setPoint = 20                                  # setpoint
para subir brazo (contracción del codo)
            self.actualPos = 0                                  # el brazo
se encuentra flexionado (abajo)
            #print("está abajo", " p:",pos)
        # iniciar controlador PID
        self.initPid()                                          # incializa
controlador PID
        # mover motor
        for i in range(300):
            potVal = self.pot.read()                            # obtener
voltaje del potenciometro
            pos = round(mapping(potVal, 0, 1023, 0, 180))       # convertir
voltaje a posición angular
            outPid = self.pid.computePID(pos, self.setPoint)    # computar
PID
            if self.actualPos == 0:
                outPid += 240
            self.activarMotor(outPid)                           # activar
motor
            #print("i:",i, " pos:",pos, " pid:", outPid,
"setpoint:",self.setPoint)
            if (abs(self.setPoint - pos) <= self.error):
                self.apagarMotor()                              #
desactivar motor
                #print("Posición alcanzada")
                break
            time.sleep(0.05)
        self.apagarMotor()

def main():
    # configuración de puertos
    ENA = 12
    IN1 = 14



     IN2 = 27
     POT = 33
     codo = Codo(ENA, IN1, IN2, POT)    # incializar codo
     codo.move()                        # mover codo

 if __name__ == "__main__":
     main()
```

## Anexo 3 Programación para implmentación de red neuronal perceptrón multicapa

```
"""
Nombre:      Red neuronal Perceptrón Multicapa.
Descripción: Algoritmo de operación de la red
             neuronal perceptrón multicapa (MLP).
Autor:       Kevin Talavera Ormeño y Oscar Taco Medina.
"""

import math # biblioteca para usar funciones matemáticas
import utils # biblioteca de funciones multi-uso

def tanh(x):
    """ Función para obtener tangente hiperbolica """
    return math.tanh(x)

""" Objeto Perceptrón Multicapa con una capa oculta"""
class mlp():
    """ Construir Red """
    def __init__ (self, inputNeurons, hiddenNeurons, outNeurons):
        # matriz de pesos sinápticos w1
        self.w1 = [[-1.6105743 ,  2.50603742],
                   [-0.92709595,  1.65726626],
                   [-1.36797582,  1.43999429],
                   [-3.02272337,  4.23880666],
                   [ 1.04690174,  1.3101708 ]]
        # matriz de pesos sinápticos w2
        self.w2 = [[-0.5988126 , -1.99812525, -1.31734588,  0.1488093 , -
3.7197204 ,-1.62263009],
                   [-1.82807918,  2.49754594,  1.61004148,  2.26201155,
4.35600482,-0.29853636]]

        """ Generar vectores de salidas de la capa oculta y salida de la
red """
        # salidas en capas de la red (oculta y salida)
        self.outHiddenLayer = [[0.0] * (hiddenNeurons + 1)]
        self.outNN = [[0.0] * outNeurons]
        # topología de red
        self.inputNeurons = inputNeurons
        self.hiddenNeurons = hiddenNeurons
        self.outNeurons = outNeurons

    def propagation(self, xi):
        """ Función para generar salida de la red
            xi = entrada a la red, debe ser una lista
            donde cada componente de la misma contenga la entrada
            correspondiente """
        # obtener salida de las neuronas de la capa oculta
        for i in range(self.hiddenNeurons):
            self.outHiddenLayer[0][i+1] = (dot(xi,self.w1[i]))
        # asignar valor de umbral
        self.outHiddenLayer[0][0] = -1
        # obtener salida de la red
        for i in range(self.outNeurons):
```

```
            self.outNN[0][i] = utils.rint(dot(self.outHiddenLayer[0],
self.w2[i]))

        return self.outNN

def dot(x,w):
    """ Función para computar el producto matricial """
    suma = 0
    for i in range(len(x)):
        multiply = x[i] * w[i]
        suma = suma + multiply
    return tanh(suma)


#if "__main__" == __name__:
 #    clf = mlp(2,5,2)
  #   bias = -1
   #  xi = [bias, 1, 1]
    #y = clf.propagation(xi)
    #print("salida: ", y)
```

## Anexo 4 Programación para control de servomotor

```
"""
Autor: Kevin Talavera Ormeño y Oscar Taco Medina |
Nombre: Biblioteca para control de servomotor
Tarjeta: ESP32 Generic NodeMCU
"""

from machine import Pin, PWM, ADC
import time

# Objeto Servo
class Servo():
    def __init__(self, servoPort):
        period = 20*10**-3                                  # 20 ms:
Periodo de trabajo de servomotor estandar
        frequency = round((1) / (period))                   # frecuencia
de trabajo
        servoPort = Pin(servoPort)                          # puerto al
que se encuentra conectado el servo
        self.servomotor = PWM(servoPort, freq = frequency)  # objeto pwm

    def move(self, position):
        # si el valor "posición" está en un rango de 0 a 180
        if position >= 0 and position <= 180:
            pwmVal = self.mapeo(position, 0, 180, 0, 99)    # mapear
valor de posición a señal PWM
            pwmVal = int(pwmVal + 26)                       # compensar
valor PWM
            self.servomotor.duty(pwmVal)                    # escribir
valor PWM
            time.sleep(0.01)                                # esperar 10
ms
        # sino, imprimir mensaje
        #else:
         #   print('ingresar un valor entre 0 a 180')

    def mapeo(self, x, inMin, inMax, outMin, outMax):
        val = ((x - inMin) * (outMax - outMin)) / ((inMax - inMin) +
outMin)
        return val

"""
if __name__ == "__main__":
    pinServoA = 15
    pinServoB = 4
    servoA = Servo(pinServoA) # inicializar obtjeto servo
    servoB = Servo(pinServoB) # inicializar obtjeto servo
    while True:
        servoA.move(100)
        #servoB.move(170)
        time.sleep_ms(2000)
        servoA.move(35)
        #servoB.move(70)
        time.sleep_ms(2000)"""
```

**Anexo 5 Programación Funciones auxiliares**

```
"""
Nombre: Funciones de uso común
Autor:  Kevin Talavera Ormeño y Oscar Taco Medina
"""

import math

def minmax(x, valMax = 1024, valMin = 0):
    xnorm = 2 * ((x - valMin) / (valMax - valMin)) - 1
    return xnorm

def filtroPromedio(x, q = 10):
    """ Función para obtener el promedio de una lista """
    fp = int((sum(x)) / (q))
    return fp

def filtroPromedio2(x, q = 10):
    """ Función para obtener el promedio de una lista """
    index = int(q / (2))
    fp = x[index] # (sum(x)) / (q)
    return fp

def filtroMax(x):
    """ Función para obtener el promedio de una lista """
    fp = max(x)
    return fp

def maping(inVal, inMin, inMax, outMin, outMax):
    mapValue = ((inVal - inMin) * (outMax - outMin)) / ((inMax-inMin) +
outMin)
    return mapValue

def isPositive(x):
    """ función para identificar número positivo """
    if (x > 0):
        resp = True
    elif (x < 0):
        resp = False
    elif (x == 0):
        resp = False
    return resp

def rint(x):
    """ función para redondear valores decimales a enteros """
    signo = isPositive(x)
    # si es mayor que un entero rescatar el valor despues del punto
decimal
    if ((1 - abs(x)) < 0):
        decimal = x - int(x)
        # detectar signo de x
        if (signo == True):
            data = 1 - decimal
            data = decimal + data + int(x)
```

```
    if (signo == False):
        data = decimal + 1
        data = decimal - data - int(x)
else:
    # si no es mayor que un entero
    if (signo == True):
        data = 1 - x
        data = x + data
    if (signo == False):
        data = x + 1
        data = x - data
return data
```

## Anexo 6 Adquisición de datos de señales mio-eléctricas

```
#-------------------------------------------------------------------------
-------
# Nombre:       Adquisición
# Descripción:  Adquirir datos de señales mio-eléctricas y almacenarla en
#               archivo excel. Este programa se ejecuta en una
computadora
#               convencional, con Python 3 instalado y las bibliotecas:
numpy
#               pandas y serial.
# Autor:        Kevin Talavera Ormeño y Oscar Taco Medina
#
# Created:     01/08/2022
# Copyright:   (c) Kevin Talavera Ormeño y Oscar Taco Medina 2022
# Licence:     <0xx1>
#-------------------------------------------------------------------------
-------

import serial, time
import numpy as np
import pandas as pd

def main():
    tipoSenal = "codo"                                          # nombre
del archivo a generar
    """ configuración de puerto serial """
    esp32 = serial.Serial("COM3", 9600)                         #
configurar puerto serial
    esp32.reset_input_buffer()                                  # limpiar
buffer del puerto serial
    time.sleep(0.1)                                             # esperar
para conectar con el microcontrolador
    serialData = " "                                            #
variable para almacenar el valor leido del puerto serial
    iteraciones = 2500                                          #
cantidad de lecturas
    dataSignals = np.zeros((iteraciones,1),np.uint16)           # matriz
de valores leidos
    for i in range(iteraciones):
        serialData = esp32.readline().decode('utf-8').rstrip()  # leer
puerto serial
        dataSignals[i,0] = np.uint16(serialData)                #
convertir tipo de dato String a Entero sin signo de 16 bits
        print(serialData)                                       # mostrar
en consola
        time.sleep(0.001)                                       # tiempo
de retardo entre lecturas de 10 ms
    # almacenar información en excel
    df = pd.DataFrame(dataSignals)                              # marco
de datos
    df.to_excel(tipoSenal+'.xlsx')                              # guarda
archivo en misma ubicación de este programa
    esp32.close()                                               # cerrar
comunicación con el puerto serial



 if __name__ == '__main__':
     main()
```

## Anexo 7 Adquisición de señales

```
"""
Nombre:      Adquisición de señales mio-eléctricas
Autor:       Kevin Talavera Ormeño y Oscar Taco Medina
Descripción: Programa para adquirir señales mio-eléctricas
Tarjeta de entrenamiento: ESP32
"""

from machine import Pin, ADC
import machine
import time

def filtroPromedio(x, q = 10):
    """ Función para obtener el promedio de una lista """
    fp = int((sum(x)) / (q))
    return fp

def maping(inVal, inMin, inMax, outMin, outMax):
    mapValue = ((inVal - inMin) * (outMax - outMin)) / ((inMax-inMin) +
outMin)
    return mapValue

def main():
    """ Configuración de puerto Serial UART2"""
    uart = machine.UART(2, 9600)
    uart.init(9600, bits=8, parity=None, stop=1)
    """ Configuración de puertos de entradas/salidas """
    signalPort = 34               # puerto para conectar señal de
amplificador
    signal = ADC(Pin(signalPort)) # crear objeto ADC
    signal.width(ADC.WIDTH_10BIT) # configurar resolución a 10 bits
    signal.atten(ADC.ATTN_11DB)   # establecer 11dB de atenuación (0v a
3v)
    cont = 20000
    """ Ciclo para adquisición de señales mio-eléctricas """
    q = 10
    tolerancia = 15
    dataP = 0
    senal = []
    count = 100
    for i in range(cont):
        senales = []
        # generar lista de señales
        for j in range(q):
            senales.append(signal.read())
        data = filtroPromedio(senales, q) # aplicar filtro promedio
        # enviar valor de posicion al serial
        if (abs(data - dataP) > tolerancia):
            print(data)
            senal.append(data)
            count += 1
            if count > 8:
                print("MAX: ", max(senal))
                count = 0
                senal = []




          uart.write(str(data)+'\n')
          dataP = data
          time.sleep_ms(20)

  if __name__ == "__main__":
      main()
```

## Anexo 8 Procesar datos

```
# -*- coding: utf-8 -*-
"""
Created on Fri Oct 28 10:41:54 2022

@author: Kevin Talavera Ormeño y Oscar Taco Medina """


import numpy as np
import redNeuronal, utils
import pandas as pd
import matplotlib.pyplot as plt


if __name__ == "__main__":
    """ importar archivos de señales mio-electricas """
    nombre = "mano"                                  # nombre de la señal a
recibir
    color = "red"                                   # color de grafica
    archivoMano = "Adquisición/"+nombre+".xlsx"    # ruta de archivo a
leer
    dataMano = pd.read_excel(archivoMano)           # leer archivo
    dataMano = np.asarray(dataMano)                 # conrvertir a tipo
arreglo
    dataMano = dataMano[:,1:]                       # eliminar linea de
indices
    """ Graficar datos """
    plt.title(nombre)
    plt.plot(dataMano, color)
    plt.ylabel("Valor ADC")
    plt.xlabel("Muestra")
    plt.ylim(0, 600)
    plt.show()
    """ Pre-procesar datos """
    umbral = 47  # valor de umbral menor acuerdo a distribución de freq
    umbral2 = 91 # valor de umbral mayor de acuerdo a distribución de
freq
    f,c = np.where(dataMano <= umbral)             # datos menores a
umbral menor
    newData = np.delete(dataMano, f, axis = 0)     # elimina datos menores
a umbral menor
    f,c = np.where(newData >= umbral2)             # datos menores a
umbral mayor
    newData = np.delete(newData, f, axis = 0)      # elimina datos mayores
a umbral
    plt.title(nombre)
    plt.plot(newData, color)
    plt.ylabel("Valor ADC")
    plt.xlabel("Muestra")
    plt.ylim(0, 600)
    plt.show()
    df = pd.DataFrame(newData)                     # crear marco de datos
    utils.distribucionFrecuencias(dataMano[:,0].tolist(), 5)  # mostrar
distribución de frecuencias



     df.to_excel(nombre+"_procesado.xlsx")                    # guardar
 datos procesados en archivo de excel
```

## Anexo 9 Aprendizaje Red Neuronal

```
# -*- coding: utf-8 -*-
"""
-------------------------------------------------------------------------
------
Name:     Aprendizaje de red neuronal
Purpose:  Adquirir datos de señales mio-eléctricas y almacenarla en
          archivo Excel
Author:      Kevin Talavera Ormeño y Oscar Taco Medina
Created:     01/08/2022
Copyright:   (c) Kevin Talavera Ormeño y Oscar Taco Medina 2022
Licence:     <0xx1>
-------------------------------------------------------------------------
------
"""

import numpy as np
import redNeuronal, utils
import pandas as pd
import matplotlib.pyplot as plt


if __name__ == "__main__":
    """ importar archivos de señales mio-electricas """
    archivoCodo = "codo_procesado.xlsx"
    archivoMuneca = "muneca_procesado.xlsx"
    archivoMano = "mano_procesado.xlsx"

    dataCodo = pd.read_excel(archivoCodo)
    dataCodo = np.asarray(dataCodo)
    dataCodo = dataCodo[:,1:]

    dataMano = pd.read_excel(archivoMano)
    dataMano = np.asarray(dataMano)
    dataMano = dataMano[:,1:]

    dataMuneca = pd.read_excel(archivoMuneca)
    dataMuneca = np.asarray(dataMuneca)
    dataMuneca = dataMuneca[:,1:]

    plt.title("Señales mioelectricas")
    plt.plot(dataMano, 'go')
    plt.plot(dataCodo, 'ro')
    plt.plot(dataMuneca, 'bo')
    plt.ylabel("Valor ADC")
    plt.xlabel("Muestra")
    plt.legend(['Mano', 'Codo', 'Muneca'])
    plt.ylim(0,500)
    plt.xlim(0, 80)
    plt.show()

    plt.title("Señales mioelectricas")
    plt.plot(utils.minmax(dataMano), 'go')
    plt.plot(utils.minmax(dataCodo), 'ro')
```

```
    plt.plot(utils.minmax(dataMuneca), 'bo')
    plt.ylabel("Valor ADC")
    plt.xlabel("Muestra")
    plt.legend(['Mano', 'Codo', 'Muneca'])
    plt.ylim(-1,0)
    plt.xlim(0, 80)
    plt.show()

    """ Concatenar datos """
    data = np.concatenate((dataCodo, dataMano, dataMuneca), axis = 0)
    """ Etiquetas """
    lblCodo = np.ones((dataCodo.shape[0],2), np.uint8) * np.array([[-
1,1]])
    lblMano = np.ones((dataMano.shape[0],2), np.uint8) * np.array([[1,-
1]])
    lblMuneca = np.ones((dataMuneca.shape[0], 2), np.uint8) *
np.array([1,1])
    lbl = np.concatenate((lblCodo, lblMano, lblMuneca), axis = 0)

    """ Entrenamiento de red neuronal """
    X = utils.minmax(data)
    entradas = data.shape[1] # señales de entrada
    neuronas1 = 5 # neuronas en capa oculta 1
    salidas = lbl.shape[1] # señales de salida
    epocas = 80000
    factorAprendizaje = 0.001
    errMin = 0.01
    # inicializar red neuronal
    rnn = redNeuronal.mlp(entradas, neuronas1, salidas)
    # aprendizaje de red neuronal
    epocas, errores, w1, w2 = rnn.aprender(X.T, lbl, factorAprendizaje,
epocas, errMin)

    plt.title("Aprendizaje de red neuronal")
    plt.plot(errores, 'blue')
    plt.xlabel("epocas")
    plt.ylabel("error")
    plt.ylim(0,1)
    # validación
    y = rnn.operacion(X.T, w1, w2)
    ys = np.rint(y)
    lista=[]
    vp, fp = 0, 0
    for i in range(y.shape[0]):
        if (ys[i] == lbl[i]).all() == True:
            print('d: ', lbl[i], ' y: ', ys[i])
            vp += 1
            lista.append(i)
        elif (ys[i] == lbl[i]).all() == False:
            fp += 1
    # efectividad
    print('vp: ', vp, ' fp: ', fp)
    efectividad = (vp) / (vp + fp)




   print('efectividad: ', efectividad)
```

## Anexo 10 Red Neuronal Perceptrón Multicapa

```
# -*- coding: utf-8 -*-
"""
Created on Sun Jun  5 02:08:56 2022

@author: Kevin Talavera Ormeño y Oscar Taco Medina |
Nombre: Perceptrón Multicapa - 1 capa oculta
"""

import numpy as np
import matplotlib.pyplot as plt

class mlp():
    # constructor
    def __init__(self, inputs, neurons, outputs):
        # inicialización de arquitectura de red neuronal
        self.inputs = inputs # neuronas en capa de entrada
        self.neurons = neurons # neuronas en capa oculta
        self.outputs = outputs # neuronas en capa de salida
        # generar matrices de pesos sinápticos con valores aleatoreos
        np.random.seed(0)
        self.w1 = np.random.rand(neurons, inputs + 1)
        self.w2 = np.random.rand(outputs, neurons + 1)

    def aprender(self, data, labels, learnRate=0.01, epochs=100,
press=0.01):
        """
        Parameters
        ----------
        data : Matriz |filas: entradas o características, columnas:
muestras): numpy float
            Matriz de muestras (datos) de aprendizaje (entradas a la
red).
        labels : Matriz: Numpy Float
            Matriz de salidas deseadas (etiquetas) | filas: muestras,
col: salida.
        learnRate : Float32
            Factor o taza de aprendizaje. The default is 0.01.
        epochs : Int16 : Entero
            Épocas de aprendizaje. The default is 100.
        press : Float32
            Error mínimo o precisión. The default is 0.01.
        Returns
        -------
        None.

        """
        # generar vector de bias para la capa de entrada
        u = np.ones((1, data.shape[1]))
        data = np.concatenate((u*-1, data), axis = 0)
        meanError = 0 # iniciar valor de error cuadratico medio
        globalE = 1 # inicializar el valor del error global
        errorRed = 1 # inicializar valor del error de la red neuronal
        epocas = 0 # inicializar contador de épocas
```

```
        errores = [] # lista para almacenar errores por la red
        xi = np.empty((data.shape[0], 1)) # vector para colocar muestra
de ap
        di = np.empty((self.outputs, 1)) # vector para colocar salida
deseada
        # iniciar secuencia de entrenamiento
        while (errorRed > press):
            errorPrev = globalE
            for i in range(labels.shape[0]):
                # obtener i-esima entrada
                xi[:,0] = data[:,i]
                # obtener i-esima etiqueta
                di[:,0] = labels[i,:]
                # propagar salida de red neuronal
                Y0, Ys = self.propagar(xi)
                self.retropropagar(xi, Y0, Ys, di, learnRate)
                # volver a computar salida de la red neuronal
                Y0, Ys = self.propagar(xi)
                # computar error medio
                meanError = meanError + (np.sum(((di - Ys)**2)))

            # computar error global de la red neuronal
            globalE = ((meanError) / (labels.shape[0]))
            # diferencia de error
            errorRed = np.abs(globalE - errorPrev)
            errores.append(errorRed)
            # incrementar el número de épocas
            epocas +=1
            # terminar entrenamiento si se rebaza el número de épocas
            if epocas > epochs:
                break

        return epocas, errores, self.w1, self.w2

    def operacion(self, data, w1, w2):
        self.w1 = w1
        self.w2 = w2
        y = []
        # vector de bias
        u = np.ones((1, data.shape[1]))
        # generar matriz de datos de validación
        X = np.concatenate((u*-1, data), axis = 0)
        # vector de muestras de validación
        xi = np.empty((X.shape[0], 1))
        # comenzar validación
        for i in range(X.shape[1]):
            xi[:, 0] = X[:, i]
            Y0, Ys = self.propagar(xi)
            y.append(Ys)
        ys = np.asarray(y)
        return ys[:,:,0]
```

```
    def activacion(self, w, x):
        return np.dot(w, x)

    def propagar(self, xi):
        # generar vector de potencial de activación de neuronas en capa
oculta I
        I0 = np.empty((self.neurons, 1))
        Y0 = np.ones((self.neurons + 1, 1)) * -1
        # computar potencial de activación de neuronas en capa oculta
        I0 = self.activacion(self.w1, xi)
        # computar salida de neuronas en capa oculta
        Y0[1:, 0] = tanh(I0[:,0])
        # computar salida de red neuronal
        Is = self.activacion(self.w2, Y0)
        Ys = tanh(Is)
        return Y0, Ys

    def retropropagar(self, xi, Y0, Ys, di, learnRate):
        # vector para salidas de capa oculta
        d1 = np.empty((self.neurons, 1))
        # almacenar salidas de capa oculta
        d1[:, 0] = Y0[1:, 0]
        # computar error
        e = (di - Ys)
        # computar delta de capa de salida
        ds = np.empty((self.outputs, 1))
        ds[:, 0] = e[:,0] * (dtanh(Ys[:,0]))
        # ajustar w2
        self.w2 = self.w2 + (Y0.T * learnRate * ds)
        # computar delta de capa oculta
        do1 = (np.dot(self.w2.T, ds) * (dtanh(Y0)))
        d1[:, 0] = do1[1:, 0]
        # ajustar w1
        self.w1 = self.w1 + (xi.T * learnRate * d1)

# funcion de activacion tangente hiperbólica
def tanh(x):
    return np.tanh(x)

# Derivada de funcion tangente hiperbólica
def dtanh(x):
    return 1.0-(np.tanh(x)**2)

# funcion sigmoidal
def sigmoide(x):
    return np.tanh(x)

def dsigmoide(x):
    return 1-x**2

if __name__ == "__main__":
    # conjunto de datos para aprendizaje
    X = np.array([[-1, -1],[-1,1],[1,-1],[1,1]])
```

```
# salidas deseadas
d = np.array([[-1],[1],[1],[-1]])
# inicializar red
clf = mlp(X.shape[1], 2, 1)
epoc, error, w1, w2 = clf.aprender(X.T, d, 0.2, 5000, 0.001)
error = np.asarray(error)
plt.plot(error, 'blue')
plt.show()
X2 = np.array([[0,1],[0,0],[1,0]])
ysal = clf.operacion(X.T, w1, w2)
print(ysal)
```

**Anexo 11 Funciones de uso común**

```
# -*- coding: utf-8 -*-
"""
Created on Thu Aug  4 11:39:54 2022

@author: Kevin Talavera Ormeño y Oscar Taco Medina """

import numpy as np


def distribucionFrecuencias(poblacion, k):
    archivo = open("distribucion_frecuencias2.txt", "w")
    xmin = min(poblacion)
    xmax = max(poblacion)
    # rango
    R = (xmax - xmin)
    # amplitud
    A = round(R / k)
    print("min: ", xmin, " max: ", xmax, " R: ", R ," A: ", A)
    # generar intervalos li
    intervaloSup = xmin
    l = []
    for i in range(k):
        intervaloInf = intervaloSup
        intervaloSup = xmin + ((i+1)*A)
        l.append([intervaloInf, intervaloSup])
    # conteo (frecuencia absoluta fi)
    f = []
    for i in range(len(l)):
        count = 0
        for j in range(len(poblacion)):
            datoEst = poblacion[j]
            if i ==6:
                if (datoEst >= l[i][0]) and (datoEst <= l[i][1]):
                    count +=1
            else:
                if (datoEst >= l[i][0]) and (datoEst < l[i][1]):
                    count += 1
        f.append(count)
    total = sum(f)
    # frecuencias relativa (hi)
    h = []
    for i in range(len(f)):
        h.append(round((f[i]/total), 3))
    # porcentajes (pi)
    p = []
    for i in range(len(h)):
        p.append(round((h[i] * 100), 3))
    # generar totales
    t = []
    tabla = [l, f, h, p]
    for i in range(1, len(tabla)):
        t.append(round(sum(tabla[i][:]),3))
    print("total: ", t)
```

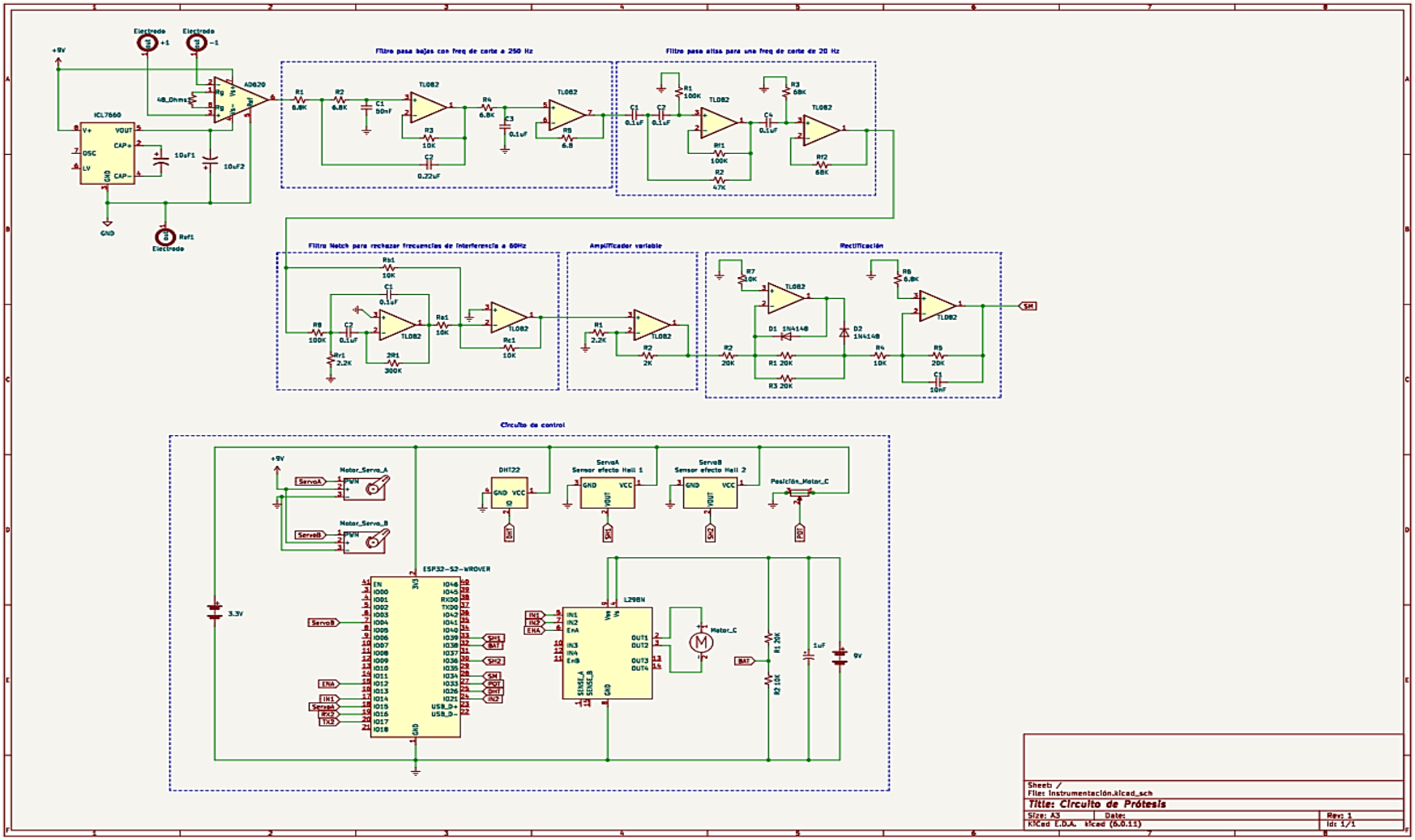
```
    # generar tabla
    titulos = ["  li    ", "fi", " hi ", "pi"]
    for i in range(len(titulos)):
        archivo.write(titulos[i])
        archivo.write(" | ")
    archivo.write("\n")
    tabla = [l, f, h, p]
    for i in range(len(tabla[1])):
        fila =[]
        for j in range(len(tabla)):
            fila.append(tabla[j][i])
            archivo.write(str(tabla[j][i]))
            archivo.write(" | ")
        archivo.write("\n")
        print(fila)
    archivo.write("Total:   |")
    for i in range(len(t)):
        archivo.write(str(t[i]))
        archivo.write(" | ")
    archivo.write("\n")
    archivo.close()

def desvEstandar(poblacion):
    """
    Parameters
    ----------
    poblacion : Vector numpy (vector fila)
        Valores de la poblaciÃ³n.
    """
    X = poblacion
    # obtener la mediana
    Xi = np.mean(X)
    # obtener la varianza
    varianza = (np.sum((X - Xi)**2))/(poblacion.shape[1]-1)
    # computar la desviaciÃ³n estandar
    return np.sqrt(varianza)

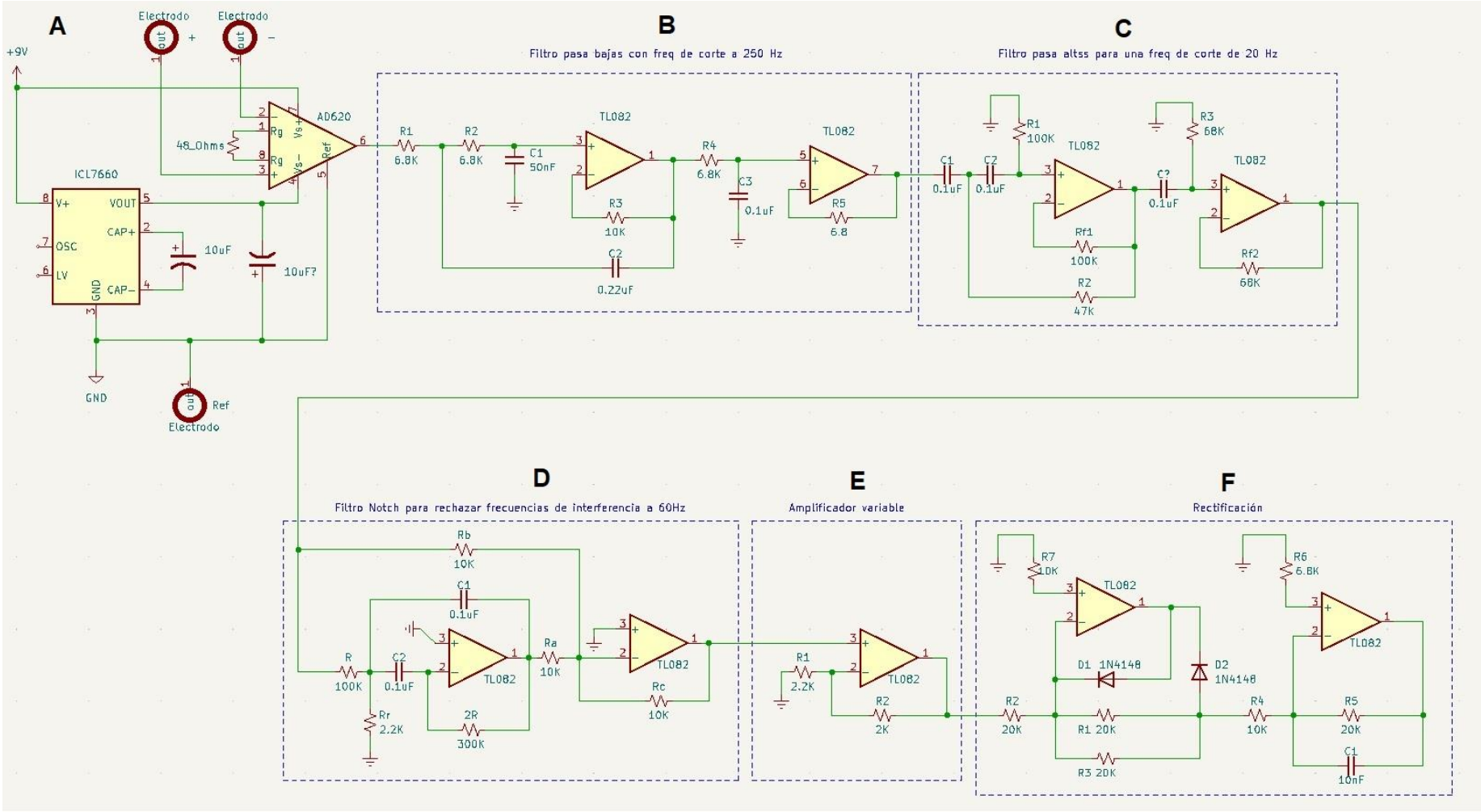
def minmax(x, valMax=1024, valMin = 0):
    # normalizar entre  -1 y 1
    xnorm = 2 * ((x - valMin)/ (valMax - valMin) ) - 1
    return xnorm

def filtroPromedio(x, q = 50):
    fp = np.zeros(x.shape)
    f, c = x.shape
    for i in range(q, f-q):
        sumValues = 0
        for j in range(q):
            sumValues = sumValues + x[i+q - j]
        fp[i,0] = (sumValues) / (q)
    return fp
```

## Anexo 12 Diagramas de conexión eléctrica Circuito general



## Anexo 13 Circuito de Adquisición



**Anexo 14 Circuito general**

