

**Escuela Superior Politécnica del Litoral**

**Facultad de Ingeniería en Mecánica y Ciencias de la Producción**

Diseño de prótesis de brazo basado en la clasificación de señales  
mioeléctricas mediante inteligencia artificial

INGE-2786

**Proyecto Integrador**

Previo la obtención del Título de:

**Ingeniero en Mecatrónica**

Presentado por:

Milton Josué Sánchez Véliz

Miguel Sebastián Jara Cadena

Guayaquil - Ecuador

Año: 2024

## Dedicatoria

---

Dedico este proyecto a mi familia, cuyo apoyo incondicional me motivó a seguir adelante en todo momento, dejando siempre las cosas en manos de Dios.

A Lunita por ser mi fiel compañera mientras me desvelaba haciendo mis tareas.

A Miguel, Mario, Djalmar, David, Doménica y José quienes se convirtieron en los verdaderos amigos que hice en el camino.

*Josué Sánchez V.*

## Dedicatoria

---

Dedico este proyecto a mi familia ya que ellos siempre han estado ahí para mí apoyándome en todo momento en este proyecto, a mis padres que siempre me han querido ver convertido en un profesional, a mis hermanas que siempre me han ayudado en todo momento en mis estudios.

A mis amigos Milton, Mario, Domenica, David, Djalmar y Jose, quienes hicieron esta experiencia universitaria muy bonita y se que me llevo a unos excelentes amigos para toda la vida.

*Miguel Jara C.*

## Agradecimientos

---

Le agradezco primeramente a Dios por darme la sabiduría y fortaleza para completar exitosamente esta etapa de mi vida.

A mis profesores, por compartir su conocimiento y motivarme a crecer profesionalmente.

A todos mis compañeros y a todas las personas que, con su apoyo y esfuerzo, contribuyeron a lo largo de mi carrera universitaria y en la culminación de este proyecto.

*Josué Sánchez V.*

## Agradecimientos

---

Le agradezco a mi familia por siempre estar ahí para mí, apoyándome y dándome ánimos para poder convertirme en una excelente persona y un muy buen profesional.

A mis compañeros de la universidad que hicieron de esta experiencia más llevadera.

A mis profesores que me impartieron todos los conocimientos que ellos podían brindarme para desarrollar mis conocimientos y que me han motivado a seguir adelante con mis estudios y vida profesional.

*Miguel Jara C.*

## Declaración Expresa

---

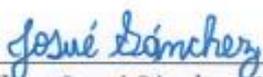
Nosotros, Milton Josué Sánchez Véliz y Miguel Sebastián Jara Cadena, acordamos y reconocemos que:

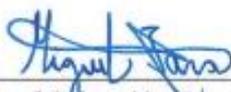
La titularidad de los derechos patrimoniales de autor (derechos de autor) del proyecto de graduación corresponderá a los autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores.

La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de nuestra innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique a los autores que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, 7 de octubre del 2024.

  
Milton Josué Sánchez Véliz

  
Miguel Sebastián Jara Cadena

## **Evaluadores**

---

**Jonathan Roberto Leon Torres, M. Sc**

Profesor de Materia

---

**Bryan Joao Puruncajas Maza, Ph. D.**

Tutor de proyecto

## Resumen

El presente trabajo desarrolla una prótesis accesible basada en inteligencia artificial para personas en Ecuador. Se busca diseñar un sistema que interprete señales musculares y controle una prótesis. La hipótesis sostiene que la inteligencia artificial mejora la precisión en la interpretación de movimientos. La justificación radica en la necesidad de soluciones económicas y personalizadas.

Se emplearon sensores para capturar señales musculares, un microcontrolador para procesarlas y un modelo de inteligencia artificial para predecir movimientos. Además, se creó una prótesis virtual para simular los movimientos antes de fabricar un prototipo. Se utilizaron técnicas de deep learning y simulación 3D en Blender.

Los resultados muestran que el diseño es más económico que las opciones importadas. El sistema alcanzó un 95% de precisión en la interpretación de señales musculares y logró simular movimientos básicos como pinza, puño, descanso y movimientos individuales de los dedos.

En conclusión, el modelo permitió clasificar movimientos con alta precisión, validando la viabilidad de la simulación en tiempo real. La integración de inteligencia artificial, biomecánica y animación 3D representa un avance en prótesis accesibles, con aplicaciones potenciales en rehabilitación y entrenamiento controlado.

Palabras Clave: Prótesis Mioeléctrica, Inteligencia Artificial, Simulación 3D, Accesibilidad, Innovación Tecnológica.

## ***Abstract***

*This work develops an accessible prosthesis based on artificial intelligence for people in Ecuador. The objective is to design a system that interprets muscle signals and controls a prosthesis. The hypothesis states that artificial intelligence improves the accuracy of movement interpretation. The justification lies in the need for affordable and personalized solutions.*

*Sensors were used to capture muscle signals, a microcontroller to process them, and an artificial intelligence model to predict movements. Additionally, a virtual prosthesis was created to simulate movements before manufacturing a prototype. Deep learning techniques and 3D simulation in Blender were employed.*

*The results show that the design is more affordable than imported options. The system achieved 95% accuracy in interpreting muscle signals and successfully simulated basic movements such as pinch, fist, rest, and individual finger movements.*

*In conclusion, the model classified movements with high accuracy, validating the feasibility of real-time simulation. The integration of artificial intelligence, biomechanics, and 3D animation represents an advancement in accessible prostheses, with potential applications in rehabilitation and controlled training.*

*Keywords: Myoelectric Prosthesis, Artificial Intelligence, 3D Simulation, Accessibility, Technological Innovation.*

## Índice general

Resumen .....	I
Abstract.....	II
Índice general.....	III
Abreviaturas .....	VII
Simbología.....	VIII
Índice de figuras.....	IX
Índice de tablas.....	XI
Índice de planos.....	XII
Capítulo 1.....	1
1. Introducción.....	2
1.1 Descripción del problema .....	3
1.2 Justificación del problema.....	4
1.3 Objetivos .....	5
1.3.1 Objetivo general .....	5
1.3.2 Objetivos específicos.....	5
1.4 Marco teórico.....	5
1.4.1 Historia de las prótesis de brazo .....	5
1.4.2 Tipos de prótesis de brazo .....	6
1.4.3 Anatomía del brazo .....	8
1.4.4 Tipos de movimientos de la mano .....	13
1.4.5 Sensores EMG .....	14

1.4.6	Inteligencia artificial.....	15
1.4.7	Otros proyectos realizados .....	18
Capítulo 2.....		22
2.	Metodología.....	23
2.1	Selección de alternativa de solución .....	23
2.2	Requerimientos de diseño .....	26
2.3	Proceso de diseño .....	27
2.4	Diseño electrónico.....	27
2.4.1	Selección del sensor EMG .....	28
2.4.2	Selección de microcontrolador para toma de datos .....	29
2.4.3	Selección de microcontrolador para prueba de movimientos .....	30
2.4.4	Integraciones.....	32
2.4.5	Procedimiento para adquisición de datos .....	33
2.4.6	Diseño PCB.....	36
2.4.7	Cálculo de baterías.....	37
2.5	Diseño de programación.....	39
2.5.1	Preprocesamiento .....	39
2.5.2	Modelo CNN y LSTM.....	41
2.6	Diseño mecánico.....	47
2.6.1	Antecedente.....	47
2.6.2	Especificaciones del diseño mecánico .....	48
2.6.3	Compartimentos integrados para componentes electrónicos .....	49
2.7	Diseño conceptual .....	49

Capítulo 3.....	52
3.    Análisis de resultados .....	53
3.1  Diseño mecánico.....	53
3.1.1  Modelo 3D de la prótesis y conexión con el sistema de predicciones .....	53
3.1.2  Propuesta de Mejora de Prótesis Física.....	54
3.2  Diseño electrónico .....	55
3.2.1  Diseño de la PCB .....	55
3.2.2  Toma de datos .....	56
3.3  Análisis de costos.....	57
3.3.1  Costos de componentes electrónicos .....	57
3.3.2  Costos de desarrollo.....	58
3.3.3  Costos de fabricación .....	58
3.3.4  Costo total del proyecto .....	59
3.3.5  Comparación con prótesis comerciales .....	59
3.4  Análisis de resultados del modelo .....	60
3.4.1  Desempeño del modelo de deep learning.....	60
3.4.2  Matriz de confusión .....	63
3.4.3  Curvas ROC .....	64
3.5  Análisis de resultados de la simulación .....	64
3.5.1  Pruebas en tiempo real .....	65
3.5.2  Integración con blender.....	65
Capítulo 4.....	67
4.    Conclusiones y recomendaciones .....	68

4.1 Conclusiones.....	68
4.2 Recomendaciones.....	69
Referencias.....	70
Apéndices .....	76

## Abreviaturas

ABS	Acrilonitrilo butadieno estireno
ADC	Analógico-digital
AUC	Area Under the Curve
CAD	Computer-Aided Design
CNN	Convolutional Neural Networks
EMG	Electromiografía
GND	Ground
IA	Inteligencia Artificial
IDE	Integrated Development Environment
INEC	Instituto Ecuatoriano de Estadísticas y Censos
LSTM	Long Short-Term Memory
MAE	Máquinas de Aprendizaje Extremo
PCB	Printed Circuit Board
PLA	Ácido poliláctico
RNA	Redes Neuronales Artificiales
ROC	Receiver Operating
SVM	Support Vector Machine
USB	Universal Serial Bus
USD	Dólar estadounidense

**Simbología**

cm	centímetros
g	gramos
GB	Gigabytes
GHz	Gigahertz
h	horas
Hz	Hertz
mA	miliamperios
mAh	miliamperios-hora
mm	milímetros
ms	milisegundos
mV	milivoltios
V	voltios

## Índice de figuras

Figura 1.1. <i>Ejemplo de prótesis pasiva [10]</i> .....	7
Figura 1.2. <i>Ejemplo de prótesis Corporal [11]</i> .....	7
Figura 1.3. <i>Ejemplo de prótesis Mioeléctrica [14]</i> .....	8
Figura 1.4. <i>Músculos presentes en el brazo [15]</i> .....	9
Figura 1.5. <i>Flexor radial del carpo [16]</i> .....	9
Figura 1.6. <i>Flexor ulnar del carpo [17]</i> .....	10
Figura 1.7. <i>Flexor superficial de los dedos [18]</i> .....	10
Figura 1.8. <i>Flexor profundo de los dedos [19]</i> .....	11
Figura 1.9. <i>Extensor radial largo del carpo [20]</i> .....	11
Figura 1.10. <i>Extensor radial corto del carpo [21]</i> .....	12
Figura 1.11. <i>Extensor de los dedos [22]</i> .....	13
Figura 1.12. <i>Movimiento de extensión (a). flexión (b), pinza (c) y prehensión (d) de las manos [41] [24]</i> .....	14
Figura 1.13. <i>Movimiento de supinación (a) y pronación de las manos (b) [42]</i> .....	14
Figura 1.14. <i>Clasificación de secuencias con convoluciones en 1D [29]</i> .....	16
Figura 1.15. <i>Clasificación de series temporales con análisis de wavelets y Deep Learning [29]</i> .....	17
Figura 1.16. <i>Comparación del comportamiento de una CNN con el cerebro humano [30]</i> ..	18
Figura 1.17. <i>Prótesis mioeléctrica y electrónica comercial [11]</i> .....	20
Figura 2.1. <i>Diagrama de flujo de proceso de diseño</i> .....	27
Figura 2.2. <i>Diagrama de conexión para uso del sensor SEN0240</i> .....	33
Figura 2.3. <i>Posición descanso</i> .....	34
Figura 2.4. <i>Posición puño</i> .....	35
Figura 2.5. <i>Posición anular-pinza</i> .....	35
Figura 2.6. <i>Posición índice</i> .....	35
Figura 2.7. <i>Posición anular</i> .....	35

Figura 2.8. <i>Toma de datos</i> .....	36
Figura 2.9. <i>Diagrama de la PCB</i> .....	37
Figura 2.10. <i>Arquitectura del modelo CNN+LSTM</i> .....	42
Figura 2.11. <i>Modelo CAD de la prótesis</i> .....	49
Figura 2.12. <i>Ilustración de la ejecución del proyecto</i> .....	51
Figura 3.1. <i>Protesis en Software Blender</i> .....	54
Figura 3.2. <i>Modelo 3D de la PCB</i> .....	55
Figura 3.3. <i>Señal capturada de puño</i> .....	56
Figura 3.4. <i>Señal capturada de descanso</i> .....	56
Figura 3.5. <i>Señal capturada de anular</i> .....	56
Figura 3.6. <i>Señal capturada de pinza anular</i> .....	56
Figura 3.7. <i>Señal capturada de índice</i> .....	57
Figura 3.8. <i>Gráfica de precisión en entrenamiento y validación</i> .....	61
Figura 3.9. <i>Gráfica de pérdidas en entrenamiento y validación</i> .....	61
Figura 3.10. <i>Matriz de confusión</i> .....	63
Figura 3.11. <i>Curvas ROC para cada clase</i> .....	64
Figura 3.12. <i>Pose pinza-anular</i> .....	65
Figura 3.13. <i>Pose Descanso</i> .....	65
Figura 3.15. <i>Pose puño</i> .....	66
Figura 3.14. <i>Pose anular</i> .....	66
Figura 3.16. <i>Pose índice</i> .....	66

## Índice de tablas

Tabla 2.1. <i>Ponderaciones para la matriz de decisión</i> .....	24
Tabla 2.2. <i>Matriz de decisión alternativa de solución</i> .....	25
Tabla 2.3. <i>Tabla de requerimientos del sistema</i> .....	26
Tabla 2.4. <i>Matriz de decisión de sensor EMG</i> .....	28
Tabla 2.5. <i>Matriz de decisión para selección de microcontrolador para toma de datos</i> .....	29
Tabla 2.6. <i>Matriz de decisión para selección de microcontrolador para prueba de movimientos</i> .....	30
Tabla 2.7. <i>Especificaciones Raspberry PI 4B</i> .....	31
Tabla 2.8. <i>Distribución de tareas para toma de datos</i> .....	34
Tabla 2.9. <i>Parámetros globales</i> .....	39
Tabla 2.10 <i>Resumen de las dimensiones del modelo</i> .....	45
Tabla 3.1. <i>Tabla de componentes electrónicos</i> .....	57
Tabla 3.2. <i>Tabla de costos de desarrollo</i> .....	58
Tabla 3.3 <i>Costos de fabricación</i> .....	58
Tabla 3.4 <i>Costo total del proyecto</i> .....	59
Tabla 3.5. <i>Reporte de clasificación</i> .....	62

**ÍNDICE DE PLANOS**

- PLANO 1 Vista explosionada con 3 servos
- PLANO 2 Ensamble completo
- PLANO 3 Medidas de orificios del antebrazo

# Capítulo 1

## 1. Introducción

La necesidad de prótesis mioeléctricas se ha incrementado significativamente en los últimos años, especialmente en personas que han sufrido amputaciones o nacen con malformaciones congénitas. A nivel mundial, el avance tecnológico en el campo de la bioingeniería ha permitido el desarrollo de dispositivos que pueden imitar movimientos naturales mediante el uso de señales eléctricas generadas por los músculos. Estos dispositivos, conocidos como prótesis mioeléctricas, ofrecen a los usuarios un mayor control y libertad de movimiento, mejorando significativamente su calidad de vida.

En Ecuador, la realidad es diferente. A pesar de los avances tecnológicos en otros países, el acceso a prótesis mioeléctricas es limitado. La mayoría de las prótesis utilizadas a nivel nacional se importan, lo que aumenta su costo, haciéndolas inaccesibles para muchas personas. Además, si se llegan a producir localmente por medio de una impresión 3D a través de programas CAD-CAM, no hay una personalización adecuada para los pacientes ecuatorianos [1]. Esta falta de oferta nacional de prótesis avanzadas genera una necesidad insatisfecha que puede abordarse mediante el uso de tecnologías emergentes como *machine learning*.

El uso de *machine learning*, aplicado a la clasificación de señales mioeléctricas, puede mejorar la funcionalidad de las prótesis, haciendo que sean más precisas en la interpretación de las señales eléctricas del cuerpo [2]. Este proyecto busca desarrollar una prótesis mioeléctrica que utilice *deep learning* para la correcta identificación de movimientos musculares, ofreciendo una alternativa más económica y accesible para los pacientes en Ecuador. Con esto, no solo se aborda la necesidad de prótesis accesibles, sino que también se promueve la innovación tecnológica en el país, alineándose con los Objetivos de Desarrollo Sostenible, como la promoción de la salud y el bienestar para todos [3].

El uso de *machine learning*, aplicado a la clasificación de señales mioeléctricas, puede mejorar la funcionalidad de las prótesis, haciendo que sean más precisas en la interpretación de las señales eléctricas del cuerpo [2]. Este proyecto busca desarrollar una prótesis

mioeléctrica que utilice *deep learning* para la correcta identificación de movimientos musculares, ofreciendo una alternativa más económica y accesible para los pacientes en Ecuador. Con esto, no solo se aborda la necesidad de prótesis accesibles, sino que también se promueve la innovación tecnológica en el país, alineándose con los Objetivos de Desarrollo Sostenible, como la promoción de la salud y el bienestar para todos [3].

### **1.1 Descripción del problema**

Según el Consejo Nacional para la Igualdad de Discapacidades, en un informe elaborado en el 2022, existe un total de 259.874 personas que requieren de una prótesis. Además, alrededor del 50% de las personas con discapacidad en Ecuador viven en condiciones de bajos recursos, donde sus ingresos mensuales oscilan entre 0 y 30 dólares. Esta situación económica limita su desarrollo y acceso a recursos apropiados que permitan mejorar sus condiciones de vida [4].

A nivel nacional, la discapacidad física es la más común, afectando al 48,89% de las personas con alguna discapacidad. En cuanto a género, los hombres representan el 56% de la población con discapacidad, aunque esta diferencia con respecto a las mujeres (44%) no es considerable. La mayor parte de este grupo, un 57,55%, está en el rango de edad de 20 a 64 años, mientras que el 28,03% corresponde a personas mayores de 65 años. En el caso de los niños menores de 9 años, poseen una prevalencia dentro de la población con discapacidad es del 2,93% [4].

En Ecuador, solo el 24% de las personas con discapacidad participan en el mercado laboral, dejando a un 76% sumido en el desempleo, esto se traduce en aproximadamente 650.000 personas fuera de la fuerza de trabajo, según datos del INEC que datan del 2022 [5]. De esta población, muchas personas requieren prótesis de brazo, ya sea por malformaciones congénitas, accidentes o enfermedades que resultan en amputaciones. Sin embargo, el acceso a prótesis mioeléctricas es sumamente limitado debido a que estas no se fabrican localmente y deben ser importadas. En adición, su alto costo está fuera del poder adquisitivo para la mayoría de los ecuatorianos. Además, la limitada capacidad de personalización de prótesis en

Ecuador restringe su funcionalidad, ya que los componentes estandarizados no suelen adaptarse bien a las necesidades específicas de cada paciente, especialmente para quienes requieren realizar actividades de alta intensidad. En adición, los largos tiempos de espera en el sistema de salud pública y la falta de tecnología avanzada dificultan aún más el acceso a prótesis adecuadas, retrasando el proceso de rehabilitación y el retorno a una vida activa [1].

## **1.2 Justificación del problema**

Desarrollar una prótesis mioeléctrica accesible y funcional en Ecuador podría transformar profundamente la vida de muchas personas con amputaciones de miembros superiores. No solo reduciría la barrera económica, dado que las prótesis importadas alcanzan precios de \$8000 [1], sino que ofrecería soluciones personalizadas y funcionales que se adapten mejor a las necesidades prácticas de cada individuo. Con una prótesis asequible y diseñada localmente, los usuarios tendrían mayores posibilidades de reinsertarse en el ámbito laboral, contribuyendo así a su estabilidad e independencia económica.

El impacto va más allá de la capacidad funcional: estas prótesis podrían mejorar notablemente la autonomía y calidad de vida de los usuarios. Con la posibilidad de realizar tareas cotidianas y laborales, los usuarios recuperan su independencia y dignidad, lo cual influye positivamente en su autoestima y bienestar emocional. Además, contar con una prótesis que responda a sus necesidades específicas ayudaría a las personas a integrarse mejor socialmente, ya que facilita una participación activa y elimina barreras de movilidad y funcionalidad.

La personalización de estas prótesis permitiría ajustar su diseño a las expectativas y contextos particulares de cada paciente, asegurando que puedan realizar actividades específicas sin sentir limitaciones en su desempeño. En conjunto, estas mejoras prácticas, emocionales y sociales crean una oportunidad de cambio significativo, brindando a las personas una herramienta poderosa para recuperar su rol en la comunidad y vivir de manera plena.

### **1.3 Objetivos**

#### **1.3.1 Objetivo general**

Validar el funcionamiento de una prótesis de brazo mioeléctrica mediante simulaciones que utilicen *deep learning* para clasificar señales musculares, con el objetivo de optimizar la movilidad de la prótesis

#### **1.3.2 Objetivos específicos**

1. Desarrollar un sistema de control basado en sensores mioeléctricos para identificar y ejecutar los movimientos esenciales de la mano.
2. Desarrollar y validar un modelo CAD de prótesis de brazo que responda a movimientos clasificados por un sistema de control basado en señales mioeléctricas, utilizando simulaciones para asegurar la correcta traducción de dichas señales en movimientos físicos y que pueda ser posteriormente implementada en futuros proyectos.
3. Crear un modelo de *deep learning* a partir de un conjunto de datos para el entrenamiento del clasificador de señales mioeléctricas, optimizando la precisión del sistema en la identificación de movimientos.

### **1.4 Marco teórico**

#### **1.4.1 Historia de las prótesis de brazo**

Las prótesis han experimentado una notable evolución a lo largo de la historia humana. Los primeros registros de prótesis se remontan al antiguo Egipto, donde se encontraron prótesis de dedos de pie en ciertas momias, como el dedo Greville Chester y el dedo del Cairo. Aunque estos dispositivos servían principalmente propósitos cosméticos vinculados a las creencias culturales de la época, se considera que algunas de estas prótesis, como el dedo del Cairo, pudieron haber tenido una funcionalidad parcial, ajustándose según las necesidades de los usuarios [6]. Durante la Edad Media, el uso de prótesis se centraba en ocultar heridas de guerra o deformidades, con dispositivos como brazos con punta de gancho o patas de palo.

Estos dispositivos básicos permitían realizar algunas tareas cotidianas y eran accesibles, especialmente para personas de la alta sociedad [7].

El desarrollo significativo de las prótesis comenzó a partir de la Segunda Guerra Mundial en Estados Unidos, un período en el cual los veteranos de guerra que habían perdido extremidades requerían soluciones más avanzadas y funcionales. En respuesta, se desarrollaron prótesis con nuevas tecnologías que ofrecían movilidad y comodidad, mejorando la calidad de vida de los pacientes mediante el uso de materiales ligeros y altamente adaptables como el titanio, el aluminio y el plástico [8]. Estos avances han permitido una adaptación más precisa de las prótesis a las necesidades específicas de cada persona, transformando la funcionalidad de estas herramientas en una extensión práctica del cuerpo humano y promoviendo un mundo más accesible para todos.

#### **1.4.2 Tipos de prótesis de brazo**

Actualmente, existen diversos tipos de prótesis de brazo, cada una diseñada para satisfacer diferentes necesidades. Estos tipos incluyen:

- **Prótesis pasivas**

Las prótesis pasivas tienen un propósito principalmente estético, en la Figura 1.1 se muestra una imagen de ejemplo. Están diseñadas para restaurar la apariencia natural del cuerpo, priorizando la estética sobre la funcionalidad. Generalmente están hechas de materiales ligeros y de bajo mantenimiento, y su costo oscila entre los 5,000 y 7,000 dólares, dependiendo del grado de realismo que se desee alcanzar en la réplica de un brazo [9].

**Figura 1.1.**

*Ejemplo de prótesis pasiva [10]*

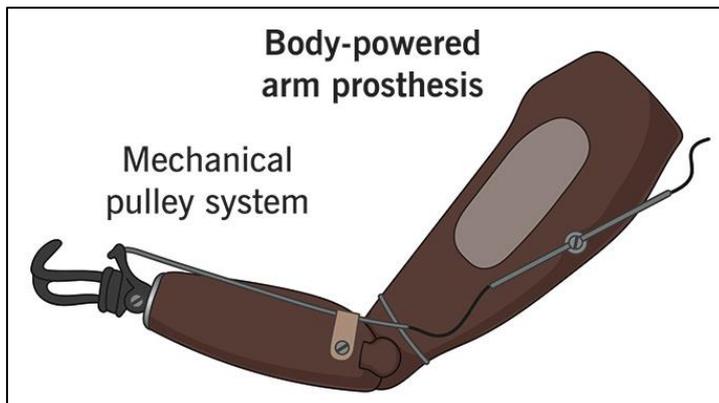


- **Prótesis corporales accionadas por el cuerpo**

Este tipo de prótesis utiliza un sistema de cables y poleas, como se observa en la Figura 1.2, que transmiten el movimiento del cuerpo, por ejemplo, del hombro, hacia la prótesis, permitiendo realizar movimientos específicos. Esto les otorga funcionalidad y las convierte en una opción asequible en comparación con otros tipos de prótesis [10].

**Figura 1.2.**

*Ejemplo de prótesis Corporal [11]*



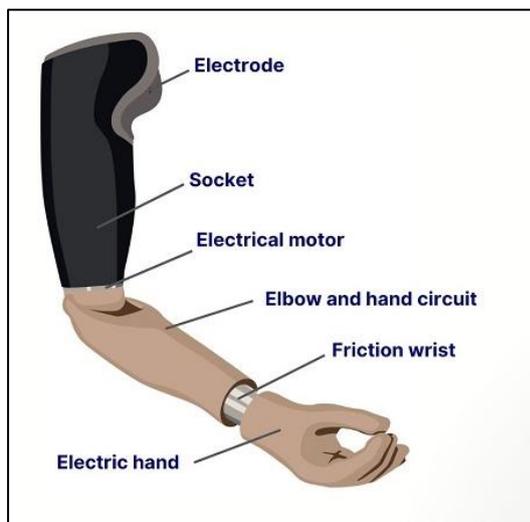
- **Prótesis mioeléctricas o con motor externo**

Las prótesis mioeléctricas son las más avanzadas tecnológicamente, se puede observar un ejemplo de esta en la Figura 1.3. Funcionan captando señales musculares de los músculos remanentes en el paciente mediante electrodos. Estas señales se envían a microcontroladores que las filtran y las procesan para controlar los movimientos de la prótesis. Aunque son más

costosas debido a la complejidad de la tecnología, ofrecen una mayor funcionalidad al permitir movimientos más precisos y naturales [12].

### Figura 1.3.

*Ejemplo de prótesis Mioeléctrica [14]*



#### 1.4.3 Anatomía del brazo

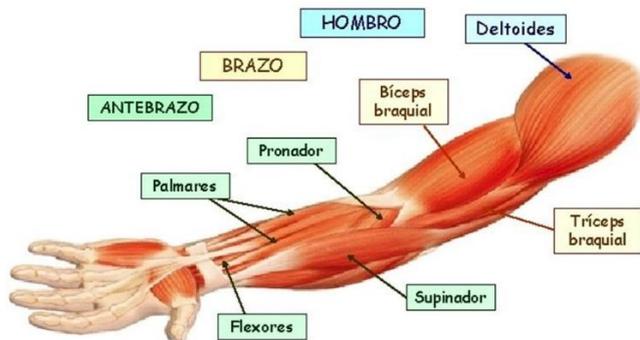
Para comprender el funcionamiento y diseño de las prótesis de brazo, es fundamental entender la anatomía del brazo humano. El brazo está compuesto por varios componentes clave:

- **Huesos:** Incluyen el húmero, el radio y el cúbito. Estos forman la estructura ósea principal y soportan el movimiento del brazo.
- **Articulaciones:** La articulación del codo y la muñeca permiten la flexión, extensión y rotación del brazo.
- **Músculos:**

Los músculos del brazo se dividen en flexores y extensores, los cuales permiten el movimiento de la articulación del codo y la muñeca, en la Figura 1.4 se puede observar una imagen con los músculos del brazo para facilitar su identificación y localización.

**Figura 1.4.**

*Músculos presentes en el brazo [15]*



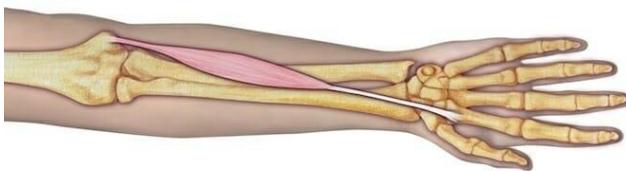
Entre los principales músculos que intervienen en la apertura y cierre de la mano, así como en los movimientos de pinza, se encuentran:

### 1. **Flexor radial del carpo**

El músculo se encuentra ubicado en la parte anterior del antebrazo, específicamente en el lado externo. Su función principal es flexionar la muñeca y contribuir a la abducción de la mano, lo que facilita el movimiento de cierre (ver Figura 1.5).

**Figura 1.5.**

*Flexor radial del carpo [16]*

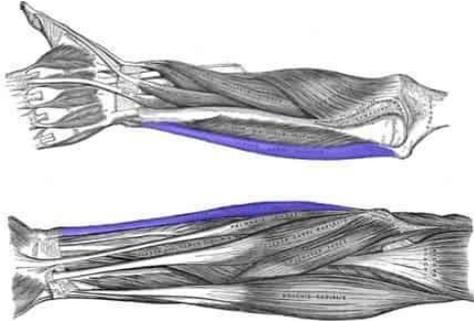


### 2. **Flexor ulnar del carpo**

El músculo está localizado en la parte anterior del antebrazo, específicamente en el lado interno. Su función principal es flexionar la muñeca y colaborar en la aducción de la mano, desempeñando un papel importante en los movimientos de cierre que requieren fuerza (ver Figura 1.6).

**Figura 1.6.**

*Flexor ulnar del carpo [17]*

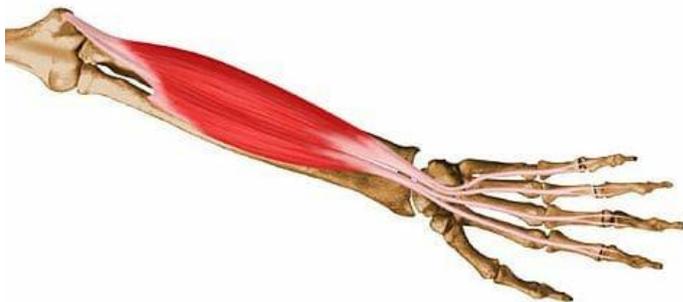


### **3. Flexor superficial de los dedos**

El músculo se encuentra en la parte anterior del antebrazo, ubicado por debajo de los flexores superficiales. Su función principal es permitir la flexión de las articulaciones interfalángicas proximales de los dedos, siendo esencial para cerrar la mano y realizar agarres (ver Figura 1.7).

**Figura 1.7.**

*Flexor superficial de los dedos [18]*



#### 4. Flexor profundo de los dedos

El músculo se localiza en la parte anterior del antebrazo, por debajo del flexor superficial. Su función es facilitar la flexión de las articulaciones interfalángicas distales de los dedos, contribuyendo al cierre de la mano con precisión y fuerza (ver Figura 1.8).

**Figura 1.8.**

*Flexor profundo de los dedos [19]*

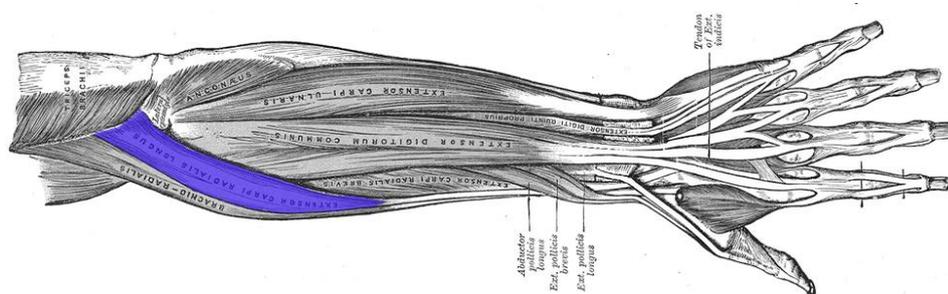


#### 5. Extensor radial largo del carpo

El músculo está ubicado en la parte posterior del antebrazo, específicamente en el lado lateral. Su función principal es extender la muñeca y colaborar en la abducción de la mano, facilitando la apertura de esta (ver Figura 1.9).

**Figura 1.9.**

*Extensor radial largo del carpo [20]*



## 6. Extensor radial corto del carpo

El músculo se encuentra en la parte posterior del antebrazo, localizado debajo del extensor radial largo. Su función es extender la muñeca y colaborar en la abducción de la mano, contribuyendo a su apertura (ver Figura 1.10).

### Figura 1.10.

*Extensor radial corto del carpo [21]*



## 7. Extensor de los dedos

El músculo está situado en la parte posterior del antebrazo. Su función principal es extender los dedos en las articulaciones interfalángicas, siendo crucial para abrir la mano y realizar movimientos de pinza (ver Figura 1.11).

**Figura 1.11.**

*Extensor de los dedos [22]*



- **Tendones y Ligamentos:** Estructuras que conectan los músculos con los huesos y proporcionan estabilidad a las articulaciones.

Comprender la anatomía del brazo es crucial para el diseño de prótesis, ya que se busca replicar los movimientos naturales de la extremidad y proporcionar al usuario una sensación de normalidad y funcionalidad en sus actividades diarias. Los músculos flexores y extensores desempeñan un papel vital en este proceso, ya que su correcto funcionamiento permite realizar acciones cotidianas como agarrar, soltar y manipular objetos.

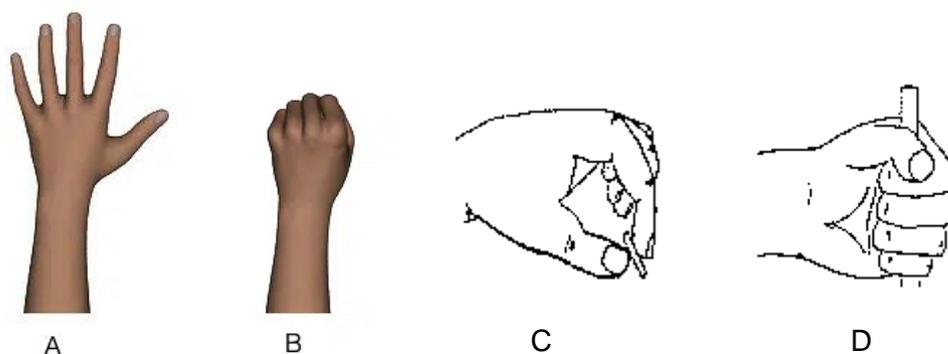
#### **1.4.4 Tipos de movimientos de la mano**

La mano puede realizar varios movimientos que se clasifican en función de la dirección, los músculos implicados y la complejidad de las acciones que permiten realizar. Los tipos principales de movimientos de la mano incluyen flexión, extensión, prehensión, supinación y pronación [22].

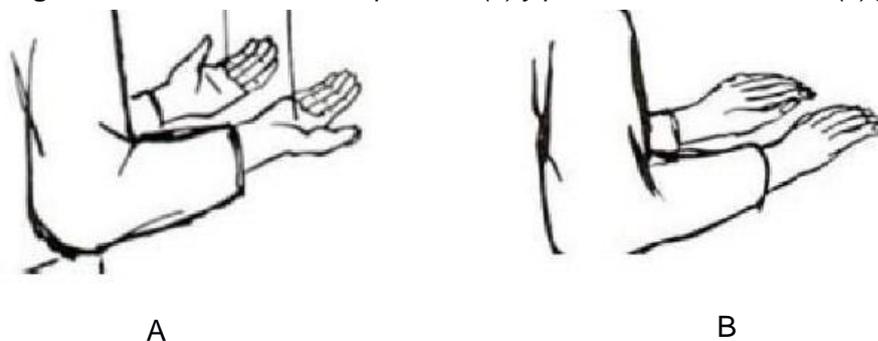
La flexión es el movimiento que reduce el ángulo entre las falanges, acercando las puntas de los dedos hacia la palma, mientras que la extensión aumenta este ángulo, alejando los dedos de la palma (Figura 1.12 a y b). La pronación y la supinación son movimientos de rotación del antebrazo: la primera orienta la palma hacia abajo, y la segunda, hacia arriba (Figura 1.13 a y b). En cuanto a movimientos de agarre, la prehensión consiste en rodear un objeto con los dedos al cerrarse sobre él, mientras que la pinza permite sujetar un objeto

utilizando la punta del pulgar con la punta de otro dedo, ya sea el índice, el anular, el dedo medio o el meñique (Figura 1.12 c y d) [23].

**Figura 1.12.** *Movimiento de extensión (a), flexión (b), pinza (c) y prehensión (d) de las manos [41]*



**Figura 1.13.** *Movimiento de supinación (a) y pronación de las manos (b) [42]*



Los movimientos de las falanges proximales en las articulaciones metacarpofalángicas son principalmente dirigidos por diversos grupos musculares: los extensores del antebrazo (como el extensor largo de los dedos, el extensor corto del pulgar, el extensor del quinto dedo y el extensor del índice), los músculos del metacarpo (lumbricales e interóseos palmares y dorsales), así como los músculos tenares (flexor corto y aductor del pulgar) y los hipotenares (abductor y flexor corto del meñique) [24].

#### **1.4.5 Sensores EMG**

El sensor de electromiografía (EMG) capta señales musculares débiles, las amplifica y las filtra, permitiendo que un procesador como Arduino u otro dispositivo similar pueda interpretar la señal resultante [25]. Su amplitud oscila típicamente entre 0 y 6 mV, mientras que la frecuencia útil se encuentra en un rango de 0 a 500 Hz, concentrándose la mayor parte de la energía entre 50 y 150 Hz [26].

Existen distintos modelos de sensores EMG para la adquisición de señales musculares, tales como el sensor *Myoware* y el sensor analógico SEN0240.

#### **1.4.6 Inteligencia artificial**

La inteligencia artificial (IA) es un campo científico que busca desarrollar máquinas capaces de razonar y aprender, emulando la inteligencia humana y abordando datos a gran escala. Esta disciplina abarca áreas como la informática, la estadística, la neurociencia y hasta la filosofía. En el ámbito empresarial, la IA se basa en tecnologías como el aprendizaje automático y profundo, usadas para analizar datos, hacer predicciones, procesar lenguaje natural y recuperar información de manera inteligente [32].

- **Aprendizaje supervisado.**

Este modelo asigna entradas a resultados específicos utilizando datos de entrenamiento etiquetados. Por ejemplo, para que en un algoritmo identifique imágenes de gatos, se le entrena con imágenes previamente etiquetadas como tales [32].

- **Aprendizaje no supervisado.**

Es un modelo de aprendizaje automático que identifica patrones en datos no etiquetados, clasificándolos en grupos según sus características. A diferencia del aprendizaje supervisado, no tiene un resultado definido de antemano, sino que analiza los datos para descubrir relaciones y realizar modelado descriptivo [28].

- **Redes neuronales convolucionales (CNN)**

Las redes neuronales convolucionales (CNN por sus siglas en inglés) son ampliamente utilizadas en inteligencia artificial debido a su capacidad para procesar datos complejos como imágenes, señales y series temporales. Estas redes emplean múltiples capas especializadas, como la capa convolucional, que aplica filtros para extraer características específicas, y la capa de agrupación (*pooling*), que reduce la dimensionalidad y simplifica el modelo al disminuir el número de parámetros. Finalmente, una capa completamente conectada (*fully connected*) integra toda la información y realiza la clasificación o predicción final [28].

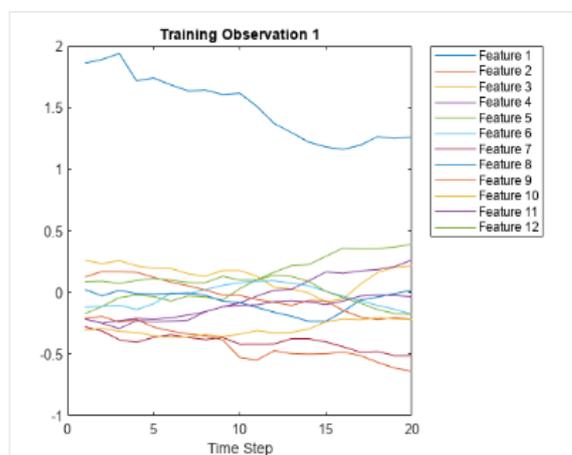
En el caso de reconocimiento de imágenes, las primeras capas de la red identifican características básicas como colores y bordes, mientras que las capas posteriores detectan patrones más complejos, como formas u objetos específicos. Este enfoque jerárquico permite que las CNN aprendan representaciones útiles de los datos, sin importar la posición, escala o rotación de los elementos [29].

Además de imágenes, las CNN también pueden aplicarse a datos de señales y series temporales mediante un preprocesamiento adecuado. Por ejemplo, al convertir señales en representaciones gráficas como escalogramas o al utilizar convoluciones en una dimensión (1D), es posible extraer características significativas para el análisis [29].

En la Figura 1.14, se muestra un ejemplo de clasificación de series temporales utilizando convoluciones 1D. Aquí, cada característica extraída a lo largo del tiempo se representa como una serie, facilitando la identificación de patrones subyacentes en los datos secuenciales. Por otro lado, la Figura 1.15 ilustra el uso de escalogramas, una técnica que combina el análisis de *wavelets* con *Deep learning*, para representar señales en el dominio tiempo-frecuencia. Esto permite capturar información espectral que resulta esencial para tareas de clasificación de series temporales.

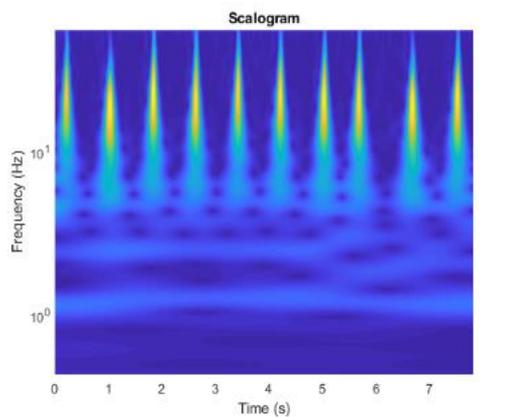
**Figura 1.14.**

*Clasificación de secuencias con convoluciones en 1D [29].*



**Figura 1.15.**

*Clasificación de series temporales con análisis de wavelets y Deep Learning [29]*



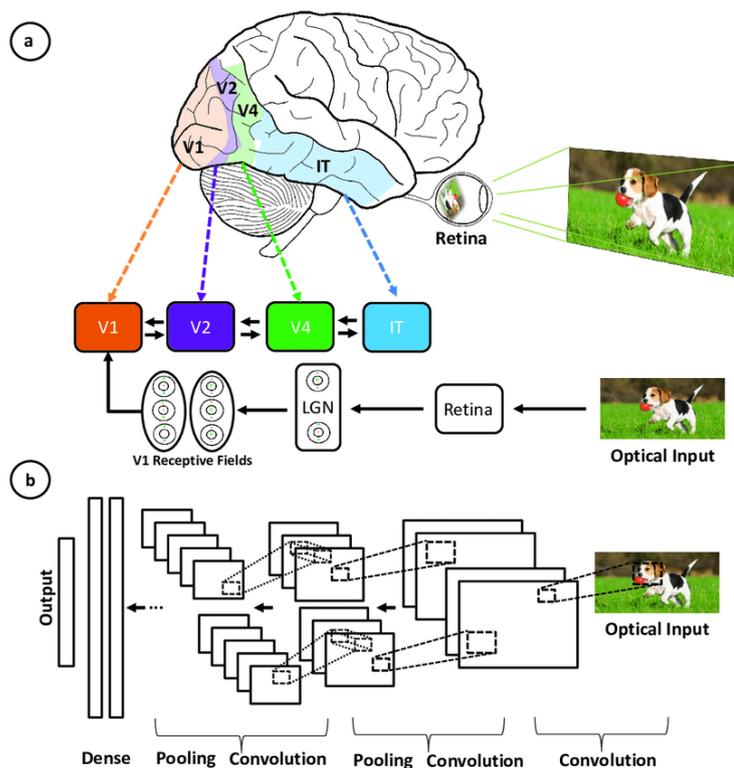
El proceso de entrenamiento en las CNN incluye varias etapas clave, como la activación ReLu, que elimina valores negativos para mejorar la eficiencia del entrenamiento, y la agrupación, que simplifica los datos manteniendo las características más importantes. Gracias a su flexibilidad, estas redes se han vuelto cruciales en múltiples áreas, diferenciándose de algoritmos clásicos como las máquinas de soporte vectorial (SVM por sus siglas en inglés) y los árboles de decisión, que requieren una ingeniería manual de características. Además, existen modelos avanzados como VGG-16, ResNet50 e Inceptionv3, que pueden adaptarse fácilmente a nuevas tareas mediante técnicas como el ajuste fino; esto ha ampliado el alcance de las CNN más allá del reconocimiento de imágenes, abarcando también el procesamiento del lenguaje natural, el análisis de series temporales y el reconocimiento de voz [30].

La Figura 1.16 compara el procesamiento visual en el cerebro humano con el de una red neuronal convolucional (CNN).

- **Parte (a):** Muestra cómo el cerebro procesa información visual, desde la retina a través de áreas especializadas (V1, V2, V4, IT), interpretando características como forma y color.
- **Parte (b):** Ilustra el flujo en una CNN, donde la imagen pasa por capas de convolución y agrupación, imitando el procesamiento cerebral hasta llegar a la clasificación final del objeto.

Figura 1.16.

Comparación del comportamiento de una CNN con el cerebro humano [30]



#### 1.4.7 Otros proyectos realizados

El proyecto titulado “Diseño e implementación de una prótesis robótica con señales EMG usando técnicas de inteligencia artificial” realizado en una universidad de Guayaquil, describe el desarrollo de una mano biónica impresa en 3D, controlada por señales EMG a través de una red neuronal de cinco capas, que logra una precisión del 78.67% en la clasificación de tareas musculares específicas. Utiliza el sensor *Myoware*, un sensor de superficie accesible y de bajo costo, ideal para captar señales musculares sin invasión. La prótesis está diseñada para realizar tres tipos de agarre, incluyendo extensión, flexión de la muñeca y puño. El uso de inteligencia artificial en tiempo real mejora la capacidad de la prótesis para interpretar y responder a las señales musculares, optimizando su funcionalidad y control [35].

Por otra parte, en la ciudad de Chimborazo se realizó otro proyecto en el que se desarrolló un sistema de adquisición y procesamiento de señales EMG con Matlab para identificar los movimientos de extensión, flexión, pronación y supinación del brazo. Utiliza un sensor EMG y un acelerómetro, cuyos datos son enviados a un Arduino y transmitidos de forma inalámbrica a un computador, donde las señales son procesadas en Matlab. Los resultados se muestran en tiempo real mediante gráficos y mensajes. Validado con una precisión superior al 96%, el sistema ha demostrado ser efectivo para las cuatro posiciones. Aunque la distancia es limitada por la antena del transmisor, la respuesta en tiempo real se mantiene. La correcta colocación del sensor y una fuente de alimentación cargada son esenciales para evitar errores en los datos [36].

En adición, un artículo de investigación presentado en Pereira, Colombia, realizó un análisis comparativo entre las Máquinas de Aprendizaje Extremo (MAE) y Redes Neuronales Artificiales (RNA) obtenidas del brazo humano para el control de una prótesis de mano robótica en un entorno virtual por medio del sensor EMG *Myoware*, entrenando ambos modelos y obteniendo una precisión del 94,00% - 94,74% con MAE y el 87,81% - 94,85% con RNA, al extraer un total de 10 características de las señales EMG mediante electrodos en el antebrazo. La clasificación obtenida se implementó en una mano robótica virtual usando V-Rep, lo que demuestra el potencial de estos modelos para mejorar el control de prótesis robóticas en aplicaciones futuras [37].

En la provincia de Jaén, España, existe una empresa especializada en ortopedia que comercializa prótesis mioeléctricas y electrónicas (véase Figura 1.17); las cuales son utilizadas en casos donde no se pueden emplear prótesis convencionales, como en amputaciones con muñones cortos o desarticulaciones de hombro. Las mioeléctricas detectan impulsos musculares mediante electrodos que se encuentran en contacto con los músculos afectados, mientras que las electrónicas aprovechan la movilidad residual y sus microinterruptores. Incluyen una mano artificial, un codo electromecánico con hasta 150° de flexión, y un sistema

de pronosupinación que permite rotación de 360°. La energía proviene de acumuladores recargables [10].

**Figura 1.17.**

*Prótesis mioeléctrica y electrónica comercial [11]*



En resumen, la inteligencia artificial aplicada a la clasificación de señales mioeléctricas (EMG) ha permitido el desarrollo de prótesis de brazo controladas por señales musculares, logrando alta precisión y respuesta en tiempo real. Estas tecnologías han permitido crear prótesis funcionales y accesibles, diseñadas para interpretar movimientos específicos como flexión, extensión, pronación y supinación, que son esenciales para actividades cotidianas. Los sensores EMG, como el *Myoware*, destacan por su bajo costo y facilidad de uso, capturando señales musculares sin necesidad de procedimientos invasivos.

El proyecto propuesto busca superar los desafíos encontrados en investigaciones previas, como la generalización de los modelos de inteligencia artificial a escenarios reales. Esto se logra mediante el diseño de un modelo optimizado que mejora la interpretación de movimientos complejos y asegura una transición fluida entre el entrenamiento del modelo y su implementación en una prótesis física.

Además, se prioriza la eficiencia en el uso de recursos, seleccionando sensores accesibles y desarrollando un sistema de validación práctico para garantizar que la prótesis reproduzca los movimientos previstos con precisión.

En términos de costos, este enfoque propone una solución económica en comparación con prótesis comerciales avanzadas, lo que permite un acceso más amplio a estas tecnologías.

La implementación incluye una plataforma de *hardware* que utiliza sensores EMG, procesadores eficientes y algoritmos avanzados para lograr un control intuitivo y en tiempo real. Este avance no solo mejora la funcionalidad de las prótesis, sino que también sienta las bases para futuras optimizaciones y ampliaciones del uso de inteligencia artificial en dispositivos médicos.

## **Capítulo 2**

## 2. Metodología.

Este capítulo describe las alternativas de solución formuladas para abordar el problema planteado, los requerimientos de diseño que guiaron el desarrollo del sistema, y el diseño conceptual y detallado que permitió implementarlo. Asimismo, se presentan las especificaciones técnicas del producto y la estrategia de identificación utilizada para relacionar las señales mioeléctricas con los movimientos del brazo.

### 2.1 Selección de alternativa de solución

Se definieron tres alternativas principales para el diseño del sistema:

- **Alternativa 1:** Consiste en el uso de una Raspberry Pi como unidad principal de procesamiento, complementada por un microcontrolador para manejar la adquisición de señales musculares. El microcontrolador procesa las señales analógicas de los sensores musculares y las envía a la Raspberry Pi, que ejecuta un modelo de *deep learning* entrenado para interpretar estas señales y controlar la prótesis en tiempo real.
- **Alternativa 2:** Implica entrenar el modelo de *deep learning* en una computadora con alta capacidad de procesamiento y luego convertir el modelo para ejecutarse en un microcontrolador compacto, optimizando el sistema en términos de espacio y consumo energético.
- **Alternativa 3:** Esta alternativa propone el uso de un sistema híbrido donde un microcontrolador realiza el preprocesamiento inicial de las señales musculares, mientras que los datos procesados son enviados a la nube. En la nube, un modelo de *deep learning* entrenado interpreta las señales y envía las instrucciones correspondientes al dispositivo protésico. Este enfoque permite aprovechar la capacidad de procesamiento y almacenamiento escalable de la nube, reduciendo la carga de trabajo en los dispositivos locales y mejorando la eficiencia del sistema

Para seleccionar la mejor alternativa, se definieron los siguientes criterios, asignándoles un peso máximo de 3, con incrementos posibles de 0.5 desde 0. Cada criterio se detalla a continuación:

- **Costo:** Representa la inversión económica necesaria para implementar la alternativa, incluyendo componentes, desarrollo y mantenimiento.
- **Precisión del modelo:** Indica el nivel de exactitud en la interpretación de las señales musculares para controlar la prótesis.
- **Consumo energético:** Mide la cantidad de energía requerida para operar el sistema en condiciones normales.
- **Escalabilidad y flexibilidad:** Evalúa la capacidad del sistema para ser ampliado o adaptado a otras configuraciones o necesidades.
- **Tiempo de procesamiento:** Considera la velocidad y la capacidad computacional del sistema para ejecutar las tareas de procesamiento necesarias, incluyendo la eficiencia en el manejo de algoritmos complejos o grandes volúmenes de datos.

Los criterios fueron ponderados en función de su importancia relativa (véase Tabla 2.1).

**Tabla 2.1.**

*Ponderaciones para la matriz de decisión*

Criterio	Peso	Rango de importancia	Porcentaje de decisión
Costo	3	1	30%
Precisión del modelo	2.5	2	25%
Tiempo de procesamiento	2	3	20%
Escalabilidad y flexibilidad	1.5	4	15%
Consumo Energético	1	5	10%

Esta ponderación establece una jerarquía clara entre los criterios, permitiendo que el análisis de las alternativas sea objetivo y consistente.

La evaluación de las alternativas propuestas se llevó a cabo utilizando la matriz de decisión mostrada en la Tabla 2.2. Se asignaron puntajes de 1 a 3 (donde 3 representa el mejor desempeño) para cada alternativa según su rendimiento en cada criterio. Posteriormente, los puntajes se multiplicaron por los pesos correspondientes para obtener el puntaje ponderado.

**Tabla 2.2.**

*Matriz de decisión alternativa de solución*

	Criterios					Resultados		
Pesos	3	2.5	2	1.5	1			
Opciones								
	Costo	Precisión del modelo	Tiempo de procesamiento	Escalabilidad y flexibilidad	Consumo Energético	Puntaje sin Peso	Puntaje con Peso	Prioridad
Alternativa 1	2	3	3	3	3	14	27	1
Alternativa 2	3	3	3	2	1	12	26.5	2
Alternativa 3	1	2	3	2	3	11	20	3

Tras evaluar cada una de las alternativas según los criterios establecidos, se determinó que la alternativa 1 es la más adecuada para abordar la problemática planteada. Esta alternativa destacó por obtener el mayor puntaje ponderado en la matriz de decisión, lo que refleja su capacidad para cumplir con los objetivos planteados en términos de precisión, escalabilidad y tiempo de procesamiento.

El diseño de la alternativa 1 combina la potencia de procesamiento de la Raspberry Pi con las capacidades específicas de un microcontrolador, optimizando la adquisición y procesamiento de las señales musculares. La Raspberry Pi se encarga de ejecutar algoritmos avanzados de procesamiento de señales y aprendizaje automático, lo que permite una

interpretación precisa y rápida de los datos. Por su parte, el microcontrolador se enfoca en tareas de bajo nivel, como la captura y filtrado de las señales en tiempo real, garantizando una respuesta ágil y eficiente.

Esta solución no solo supera las limitaciones técnicas observadas en otras alternativas, como la falta de potencia para ejecutar el modelo de forma correcta o la poca flexibilidad, sino que también ofrece una implementación eficiente y práctica para el desarrollo del sistema. Además, la arquitectura propuesta permite una fácil escalabilidad, ya que tanto la Raspberry Pi como el microcontrolador pueden ser actualizados o reemplazados por componentes más avanzados en el futuro, sin necesidad de rediseñar todo el sistema.

Otro aspecto destacable es lo modular del diseño, que facilita la integración de nuevos sensores o funcionalidades, adaptándose a las necesidades específicas del usuario.

## 2.2 Requerimientos de diseño

Desde el inicio del proyecto, se identificaron necesidades específicas para garantizar el funcionamiento y la efectividad del sistema. Los requerimientos ilustrados en la Tabla 2.3 se categorizaron en funcionales y técnicos, formando la base para el diseño y desarrollo.

**Tabla 2.3.**

*Tabla de requerimientos del sistema*

	Aspecto	Descripción
FUNCIONALES	Movimientos básicos	Identificar 5 movimientos esenciales del brazo: flexión, extensión, agarre y movimiento de pinza, garantizando una funcionalidad básica significativa.
	Duración de uso	Las baterías que posea la prótesis deben de tener una duración mínima de 8 horas.
	Sincronización con el software CAD	Lograr replicar los movimientos en un software CAD que posea el diseño mecánico de la prótesis
TÉCNICOS	Técnicas utilizadas	Implementar técnicas avanzadas de aprendizaje automático para mejorar la precisión y adaptabilidad del sistema.
	Precisión del modelo	Alcanzar una precisión de clasificación del 90% en movimientos identificados.

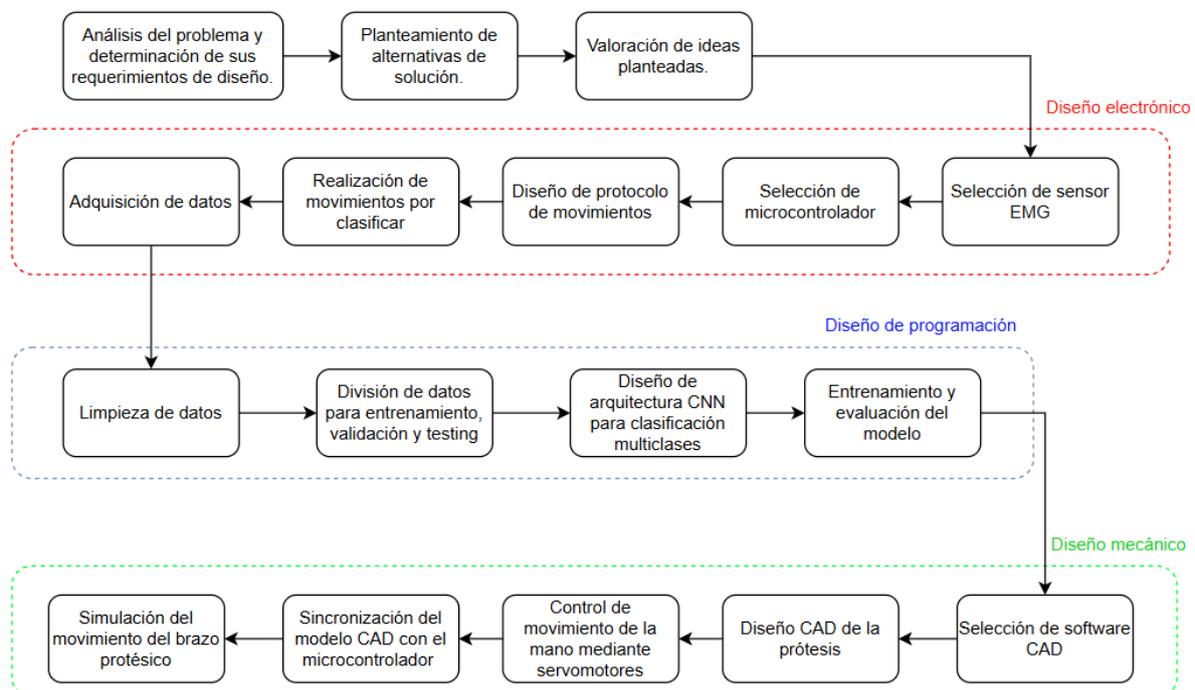
Robustez del sistema	Asegurar precisión en la interpretación de movimientos mediante técnicas de filtrado, garantizando resistencia al ruido de los sensores EMG y un desempeño confiable en diversas condiciones.
Tiempo de muestreo	Garantizar un tiempo de respuesta aproximado de 100 ms para clasificar un movimiento después del preprocesamiento de las señales.

### 2.3 Proceso de diseño

La realización del diseño incluyó varias fases en las que se desarrollaban las áreas correspondientes a la electrónica, programación y mecánica, siguiendo en todo momento los requerimientos del cliente. La Figura 2.1 muestra el proceso que se siguió durante toda la etapa de diseño.

**Figura 2.1.**

*Diagrama de flujo de proceso de diseño*



### 2.4 Diseño electrónico

El diseño electrónico del brazo protésico se basa en la selección e integración de componentes que permitan la correcta adquisición, procesamiento y clasificación de las señales mioeléctricas de manera eficiente y precisa. Para ello, se evaluaron diversas

alternativas de sensores EMG y microcontroladores, garantizando un diseño con un balance entre el rendimiento, costo y facilidad de implementación.

#### **2.4.1 Selección del sensor EMG**

Los sensores EMG SEN0240, Myoware y AD8232 permiten medir la actividad eléctrica de los músculos para aplicaciones en control robótico, interfaces hombre-máquina e investigación biomédica. El SEN0240 de DFRobot, con un costo aproximado de \$50, incluye un electrodo seco, transmisor de señal y cables de conexión, facilitando su integración con dispositivos electrónicos. El Myoware, con un precio de \$75, amplifica y procesa señales musculares, proporcionando salidas de envolvente y EMG sin procesar, además de contar con indicadores LED y ajuste de ganancia. Finalmente, el AD8232, disponible por \$33.99, ofrece salida analógica, detección de Leads-Off y compatibilidad con Arduino, siendo compacto y adecuado para diversas aplicaciones. Especificaciones técnicas más detalladas de los sensores EMG se encuentran en el Apéndice A.

Para capturar las señales mioeléctricas necesarias para el control de la prótesis, se consideraron tres sensores: AD8232, *Myoware* y SEN0240. Los criterios de selección incluyeron precisión, costo, compatibilidad y facilidad de uso según se muestra en la Tabla 2.4

**Tabla 2.4.**

*Matriz de decisión de sensor EMG*

Criterio	Peso (%)	AD8232	Myoware	Sen0240
Costo	40%	9	8	7
Precisión	30%	7	8	9
Compatibilidad	10%	9	9	9
Facilidad de uso	20%	7	8	9
Puntaje total	100%	8	8.1	8.2

El sensor SEN0240 fue seleccionado debido a su buena precisión y facilidad de uso al no necesitar de muchos componentes para las conexiones en comparación con los otros sensores, además de su buena compatibilidad que lo convirtió en una opción ideal para el proyecto.

#### **2.4.2 Selección de microcontrolador para toma de datos**

La toma de datos implica capturar las señales mioeléctricas a través de sensores, para luego analizarlas y reproducir los movimientos que se pretenden realizar. La selección del microcontrolador adecuado es crucial para garantizar que los datos sean adquiridos de manera precisa y eficiente, lo que permitirá el correcto funcionamiento de la prótesis.

**Tabla 2.5.**

*Matriz de decisión para selección de microcontrolador para toma de datos*

Criterio	Peso	Raspberry Pi	ESP32	Arduino UNO
Desempeño en tiempo real	30%	7	9	6
Tamaño físico	10%	6	8	8
Consumo energético	10%	4	8	9
Costo	20%	6	9	8
Conectividad	10%	9	10	6
Facilidad de programación	10%	7	9	8
Compatibilidad con sensores	10%	8	9	7
Puntaje total	100%	6.7	8.9	7.2

La ESP32 fue seleccionada con un puntaje de 8.9 según la Tabla 2.5 debido a su excelente desempeño en tiempo real, baja latencia y bajo consumo energético. Por otro lado, aunque la Raspberry Pi obtuvo un puntaje de 6.7, destaca por su capacidad de procesamiento

y conectividad general. Sin embargo, presenta una limitación significativa al no contar con un conversor analógico-digital (ADC) integrado, lo que dificulta su conexión directa con sensores analógicos como el SEN0240. Esta limitación significa que se necesitaría de un componente extra para realizar esta conversión y poder obtener los datos del sensor. Finalmente, el Arduino UNO sobresale en costo y eficiencia energética, pero presenta limitaciones en cuanto a procesamiento y conectividad, logrando un puntaje de 7.2.

### **2.4.3 Selección de microcontrolador para prueba de movimientos**

La Tabla 2.6 presenta la matriz de decisión para seleccionar el microcontrolador adecuado para probar los movimientos de la prótesis. Se evaluaron tres opciones (Raspberry Pi 4, ESP32 y Arduino UNO) basándose en criterios clave como la capacidad de procesamiento, memoria disponible y desempeño en tiempo real.

**Tabla 2.6.**

*Matriz de decisión para selección de microcontrolador para prueba de movimientos*

Criterio	Peso	Raspberry Pi 4	ESP32	Arduino UNO
Capacidad de procesamiento	30%	9	6	5
Desempeño en tiempo real	25%	8	5	4
Memoria disponible	25%	9	4	3
Facilidad de programación	10%	8	9	8
Conectividad	10%	9	10	9
Costo	10%	7	9	8
Puntaje total	100%	8.1	6.2	5.7

Como se observa en la Tabla 2.6, la Raspberry Pi 4 obtuvo el puntaje más alto debido a su mayor capacidad de procesamiento y memoria suficiente para ejecutar modelos de *Deep Learning*, lo que permitió superar las limitaciones encontradas con el ESP32, que no tenía suficiente memoria para probar los movimientos de la prótesis.

El Raspberry Pi 4 Model B es una computadora de placa única de bajo costo y alto rendimiento, desarrollada por la Fundación Raspberry Pi. Con su capacidad para ejecutar una amplia gama de aplicaciones, desde proyectos educativos hasta tareas de computación avanzada, el Raspberry Pi 4 ha revolucionado el acceso a la informática y la programación. Su diseño compacto y versátil lo convierte en una opción ideal para diversas aplicaciones. En nuestro caso, se utilizará el Raspberry Pi 4 para ejecutar un modelo de deep learning con el fin de clasificar señales, aprovechando su capacidad de procesamiento para realizar inferencias en tiempo real, con un bajo costo energético y espacio físico.

A continuación, en la Tabla 2.7 podemos ver algunas de sus especificaciones:

**Tabla 2.7**

*Especificaciones Raspberry PI 4B*

Especificación	Detalles
Procesador (CPU)	Broadcom BCM2711, ARM Cortex-A72, 4 núcleos a 1.5 GHz
Memoria (RAM)	2 GB, 4 GB, 8 GB LPDDR4-3200 SDRAM
Almacenamiento	Tarjeta microSD para almacenamiento, soporte para discos duros/flash por USB
Conectividad	Ethernet Gigabit, Wi-Fi 802.11b/g/n/ac, Bluetooth 5.0 (BLE)
Puertos USB	2 puertos USB 3.0, 2 puertos USB 2.0
HDMI	2 puertos micro-HDMI (hasta 4K a 60Hz o 2K a 60Hz)
Audio	Jack de 3.5 mm, salida digital por HDMI
GPU	Broadcom VideoCore VI, soporte para hasta dos pantallas 4K

Especificación	Detalles
Fuente de alimentación	5V DC a través de USB-C (al menos 3A), soporte para fuente de alimentación oficial
Sistema operativo	Raspberry Pi OS, Ubuntu, LibreELEC, entre otros
GPIO	40 pines GPIO (entrada/salida digital, PWM, I2C, SPI)
Dimensiones	88 mm x 58 mm x 19.5 mm
Peso	Aproximadamente 46 gramos

#### **2.4.4 Integraciones**

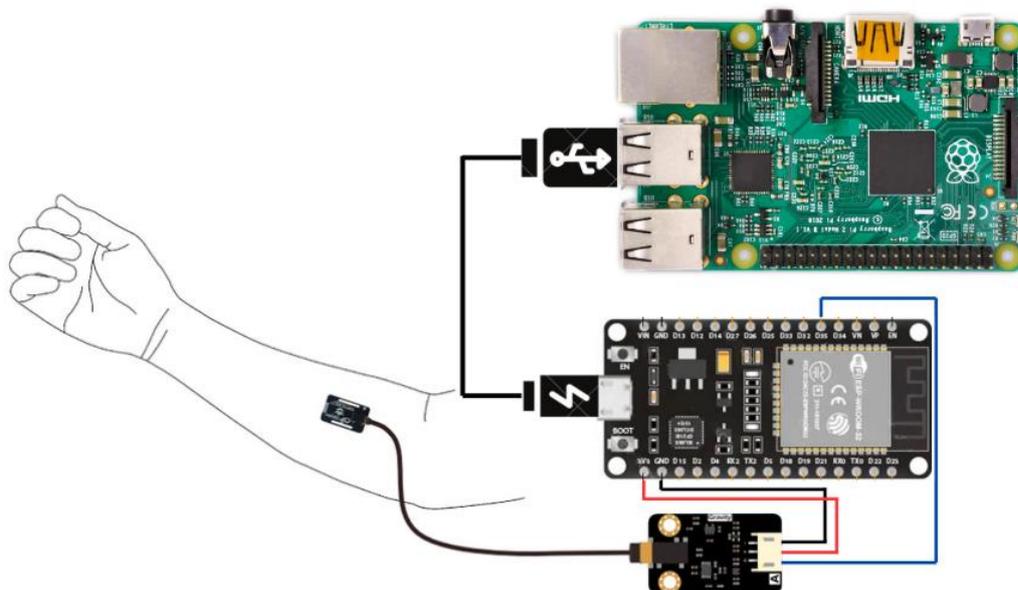
Para realizar las conexiones y tomar los datos de los sensores que serán procesados por los microcontroladores se obtuvieron los siguientes materiales:

- a) ESP32
- b) Sensor EMG SEN 0240
- c) Cable USB
- d) 1 electrodo seco

Una vez se contó con todos los materiales ya se procedió con las conexiones, las cuales se siguieron de acuerdo con lo ilustrado por la Figura 2.2, en donde se observa que el pin de la señal EMG del sensor se conectó a un pin analógico del microcontrolador, el Pin de 5V se conecta al sensor para darle energía, de igual manera, se conectó el pin de GND de la placa a la ESP32.

**Figura 2.2.**

*Diagrama de conexión para uso del sensor SEN0240*



Por último, se realizó un pequeño script en el Arduino IDE para captar las señales mioeléctricas a través del monitor serial, ajustando un delay de 10ms para tener una frecuencia de muestreo de 500HZ, normalmente usado en este tipo de proyectos.

#### **2.4.5 Procedimiento para adquisición de datos**

Para adquirir los datos producidos por las señales EMG se definió una cantidad de pacientes que deben de tomarse las pruebas, así como una lista de movimientos que debe hacer cada paciente, junto con sus repeticiones y el tiempo de cada repetición. Toda esta información se condensa en la Tabla 2.8.

**Tabla 2.8.***Distribución de tareas para toma de datos*

Tarea	Descripción	Posición de los electrodos	Sujetos	Duración
1	Mantener el brazo en reposo.			
2	Movimiento del dedo índice	Músculo flexor superficial de los dedos	3	10 segundos
3	Pinza (pulgarcillo con anular)			
4	Movimiento de puño			
5	Movimiento del dedo anular			

La tarea 1 consiste en tener el brazo en posición de reposo como una medida de referencia para el dataset, esto se puede apreciar en la Figura 2.3; en la tarea 2 se mueve el dedo índice acercándolo a la palma de la mano y se mantiene esa posición por 10 segundos (Figura 2.6). En la tarea 3 se realiza el movimiento de pinza (Figura 2.7), juntando las yemas de los dedos anular y pulgar; para la tarea 4 se intenta cerrar el puño tal y como se observa en la Figura 2.4, manteniéndolo de esa manera por 10 segundos; por último, la tarea 5 se trata de mover el dedo anular hacia la palma de la mano similar a la tarea 2 (véase Figura 2.7).

**Figura 2.3.***Posición descanso*

**Figura 2.4.**

*Posición puño*



**Figura 2.5.**

*Posición anular-pinza*



**Figura 2.6.**

*Posición índice*



**Figura 2.7.**

*Posición anular*



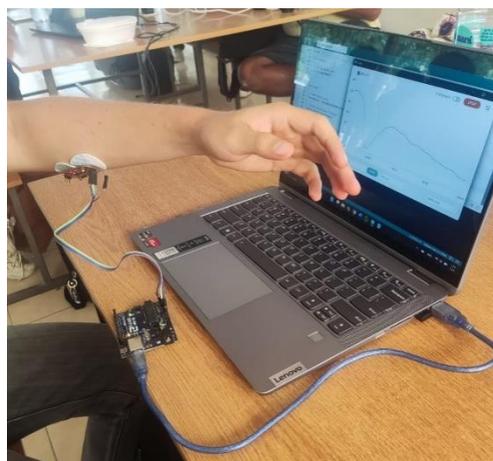
Para la captura de datos, se utilizó el software Coolterm, que se encarga de captar la información del monitor serial del Arduino IDE y guardarla como un archivo .txt. Todos estos datos se transformaron posteriormente a formato .csv y se almacenaron en una carpeta específica para cada sujeto. Dentro de cada carpeta de sujeto, se creó una subcarpeta para cada tarea, y dentro de estas subcarpetas se guardaron los archivos .csv correspondientes.

Para la adquisición de los datos del sensor, se utilizó una frecuencia de muestreo de 500 MHz, lo que permitió capturar con precisión las variaciones en las señales musculares durante la ejecución de cada tarea.

En la Figura 2.8 se muestra el momento de cuando se estaba realizando la adquisición de datos.

**Figura 2.8.**

*Toma de datos*



#### **2.4.6 Diseño PCB**

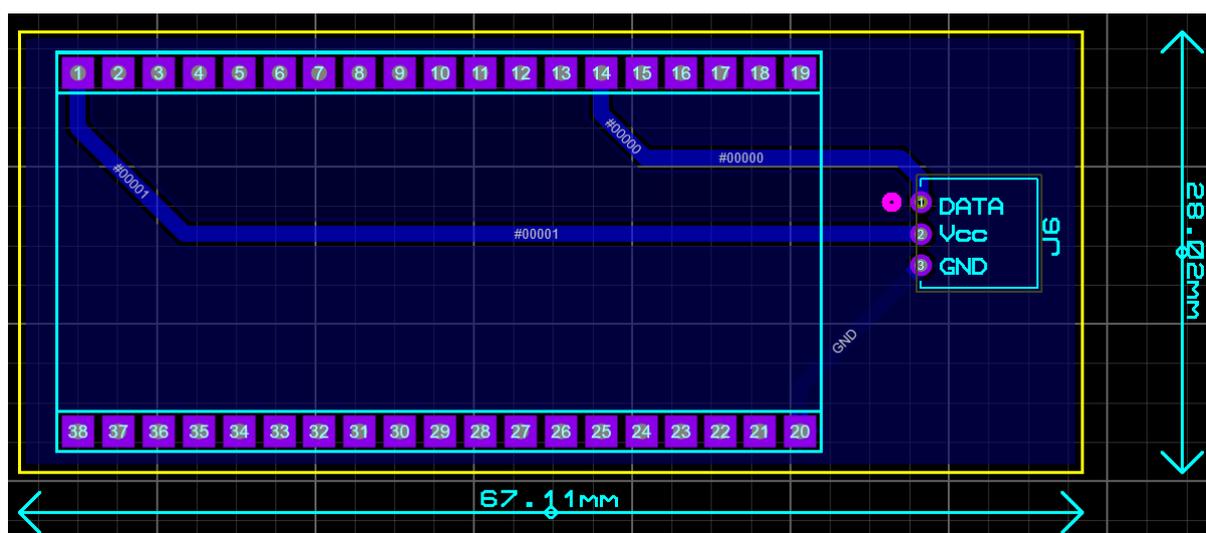
El diseño de la PCB es fundamental para la captura, procesamiento y transmisión de las señales EMG provenientes del sensor SEN0240. El diseño de la Figura 2.9 permite la integración eficiente de todos los componentes electrónicos y proporciona flexibilidad para reemplazar o actualizar el microcontrolador ESP32 en caso de fallos.

A continuación, se detalla las partes de la PCB:

- **Socket para ESP32:** La PCB incluye un socket para insertar y retirar el ESP32 fácilmente. Esto facilita su reemplazo en caso de daño o actualización.
- **Conector para el sensor SEN0240:** El sensor se conecta al PCB mediante un conector hembra de 3 pines que se conecta a los siguientes pines del ESP32:
  - **Pin 5V:** Alimentación del sensor.
  - **GND:** Tierra.
  - **Pin 35:** Entrada analógica para capturar las señales EMG.
- Desde el conector, los cables salen hacia los electrodos del sensor.

Figura 2.9.

Diagrama de la PCB



#### 2.4.7 Cálculo de baterías

Para calcular el consumo de batería del sistema, se deben estimar los requerimientos energéticos del ESP32 y del sensor SEN0240, ya que son los principales componentes que demandan energía. Según las especificaciones del sensor, su rango de voltaje de operación es de 3.3 a 5.5 V, con un consumo de corriente de al menos 20 mA en condiciones normales. Por su parte, el ESP32, de acuerdo con su ficha técnica, tiene un consumo de corriente típico en operación activa de aproximadamente 240 mA a 5 V, dependiendo de las tareas y periféricos activados.

**Consumo total estimado:**

- Sensor SEN0240: Consumo de corriente del sensor = 20 mA
- ESP32: Consumo de corriente del ESP32 = 240 mA

Por lo tanto, el consumo total del sistema es:

Consumo de corriente total = Corriente sensor + Corriente ESP32 = 20 mA + 240 mA = 260 mA

**Requisito de duración de la batería:**

Se desea que el sistema funcione al menos 8 horas antes de requerir recarga. La capacidad mínima de la batería necesaria se calcula utilizando la fórmula:

$$\textit{Capacidad batería} = \textit{Corriente total} \times \textit{tiempo de operación}$$

Donde:

- Corriente total = 260 mA
- Tiempo de operación = 8 h

$$\textit{Capacidad batería} = 260 \textit{ mA} \times 8 \textit{ h} = 2080 \textit{ mAh}$$

**Consideraciones adicionales:**

Para garantizar una operación estable y considerando posibles pérdidas en eficiencia del sistema, es recomendable utilizar una batería con una capacidad nominal de al menos 2500 mAh a 5 V, lo cual proporciona un margen de seguridad adecuado para cumplir con el requisito de funcionamiento continuo.

El sistema está diseñado para incluir una bornera en el PCB, lo que permite conectar de manera sencilla una batería externa al puerto VIN del ESP32. Optar por una batería recargable de 5 V elimina la necesidad de reguladores de voltaje adicionales, simplificando el diseño y mejorando la eficiencia energética. Esta configuración asegura que el sistema sea portátil, funcional y capaz de operar durante periodos prolongados, cumpliendo con los requisitos de diseño y autonomía.

## 2.5 Diseño de programación

La programación implementada para el modelo aglutina la preparación de datos, entrenamiento de un modelo CNN para clasificar señales EMG y la respectiva evaluación del error y su precisión. Para ello se decidió utilizar Google Colab y se programó en el lenguaje Python. Se creó un repositorio en GitHub cuyo link se encuentra en la sección A de apéndices.

### 2.5.1 Preprocesamiento

El preprocesamiento de datos es esencial para preparar las señales mioeléctricas capturadas por el sensor EMG y garantizar un entrenamiento efectivo del modelo. Esta etapa incluyó la configuración de parámetros, carga y la división en conjuntos de entrenamiento, validación y prueba, asegurando datos estructurados y listos para la clasificación de movimientos mioeléctricos.

- **Importación de librerías**

Se importaron bibliotecas necesarias para el manejo de datos (pandas, numpy), entrenamiento del modelo (tensorflow, keras), preprocesamiento (sklearn), visualización (matplotlib), y otras utilidades.

- **Configuración de parámetros**

Se definieron parámetros clave del sistema como el tamaño de cada ventana temporal, que es muy útil para análisis de series temporales. Así mismo se definió el número de canales correspondiente a la cantidad de salidas que tiene el sensor EMG que se está usando, y también se estableció el número de clases que representa la cantidad de movimientos que se pretende clasificar. Estos valores se encuentran detallados en la

Tabla 2.9.

**Tabla 2.9.**

*Parámetros globales*

Tamaño ventana temporal	300 (10 seg)
Número de canales	1
Número de clases	5

- **Carga de datos**

Se creó una función *cargar\_datos*: que se encarga de leer archivos .csv desde una carpeta, asignar etiquetas y combinar todos los datos en un solo *dataframe*. La salida de esta función es el compilado de todos los datos con una columna adicional que representa la categoría a la que corresponde el movimiento por clasificar.

- **Generación de ventanas**

La función *generar\_ventanas()* crea ventanas deslizantes a partir de un conjunto de datos, con el objetivo de estructurar las señales o muestras para el entrenamiento de un modelo. Estas ventanas se generan a partir de los datos originales con un tamaño fijo y solapamiento configurable.

➤ **Entradas**

- **Datos:** DataFrame que contiene los datos originales, incluyendo las señales y las etiquetas asociadas. El dataset presenta un total de 96902 filas (cantidad de muestras) y 2 columnas( valor numérico de la señal y etiqueta del movimiento).
- **Tamaño de ventana:** Longitud de cada ventana deslizante (número de muestras). En este caso se escogió el tamaño de 300 muestras por ventana.
- **Columna sensor:** Una sola columna que recibe los datos emitidos por el sensor que se incluirán en las ventanas.
- **Solapamiento:** Porcentaje de superposición entre ventanas consecutivas, se estableció el 90%

- **Cálculo del paso**

Calcula el número de muestras que avanzará cada ventana deslizante. Se realiza una acción similar al *stride* con la diferencia que para el solapamiento se define el porcentaje de

datos repetidos que se quiere tener entre ventanas consecutivas. A continuación, la ecuación 2.1 determina el paso a partir del solapamiento

$$paso = tamaño\_ventana \times (1 - solapamiento) \quad (2.1)$$

$$paso = 300 \times (1 - 0.90) = 30$$

- **Creación de ventanas**

- Itera a través de los datos, creando ventanas deslizantes de tamaño fijo, seleccionando las columnas especificadas.
- Obtiene la etiqueta más común dentro de cada ventana (`mode()[0]`) para asignarla como la etiqueta de esa ventana

- **Salidas**

- **Ventanas:** Un arreglo de NumPy con las ventanas de datos cuya dimensión está en función de la cantidad total de ventanas creadas que son 3332, el tamaño de cada ventana (300) y la única columna que genera el sensor utilizado (véase ecuación 2.2)

$$dimensión = N \times tamaño\_ventana \times n\_sensores \quad (2.2)$$

$$dimensión = 3332 \times 300 \times 1$$

- **Etiquetas:** Un arreglo de NumPy con las etiquetas asociadas a cada ventana.

- **División de datos**

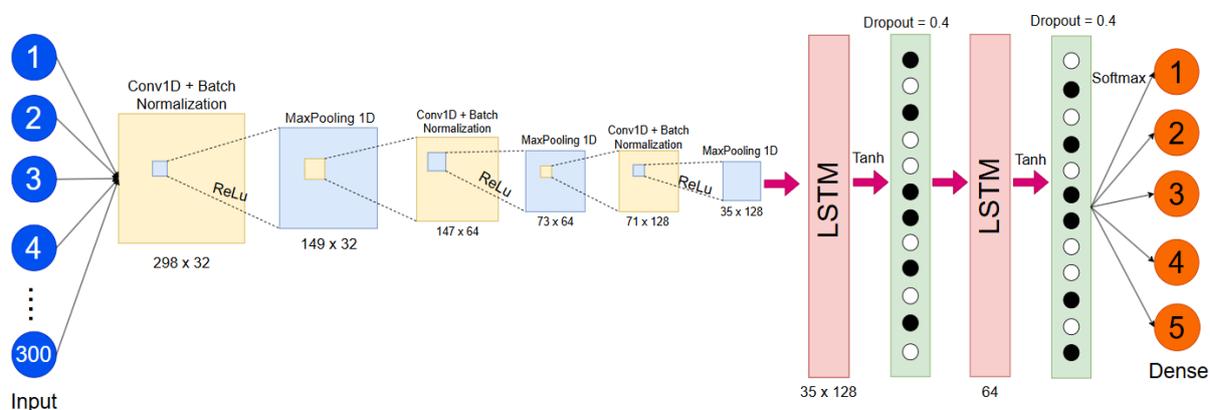
Se dividió el conjunto de datos en entrenamiento (70%), validación (15%) y prueba (15%) usando la función `train_test_split`.

### 2.5.2 Modelo CNN y LSTM

Se definió un modelo diseñado para manejar datos de señales temporales. Se usó redes neuronales convolucionales (CNN) para extraer patrones locales y *Long Short-Term Memory* (LSTM) para capturar relaciones temporales entre las características, optimizando el rendimiento en las tareas de clasificación multiclase, según se ilustra en la Figura 2.10:

Figura 2.10.

Arquitectura del modelo CNN+LSTM



- **Primera capa Conv1D**

Detectan patrones espaciales y temporales

$$salida = ReLu(W * entrada + b) \quad (2.2)$$

ReLU hace referencia a la función de activación que le dará no linealidad a la función de entrada, la cual está multiplicándose por el peso  $W$  y sumándose a un bias  $b$  de acuerdo con lo que se muestra en la ecuación 2.2.

La salida de una capa Conv1D al procesar un tensor de entrada ( $L_{entrada}, n_{canales}$ ) se convierte en un tensor de salida con forma ( $L_{salida}, n_{filtros}$ ), de las cuales, la longitud de salida  $L_{salida}$  se la obtiene según se muestra en la ecuación 2.3, considerando un tamaño del kernel de 3, el stride se lo dejó en su valor estándar que es 1 y no se consideró el padding, por lo que su valor es 0.

$$L_{salida} = \frac{L_{entrada} + 2Padding - kernel\_size}{stride} + 1 \quad (2.3)$$

$$L_{salida} = \frac{300 + 2(0) - 3}{1} + 1 = 298$$

Número de filtros: 32

Salida: (298,32)

- **Batch normalization**

Normaliza las activaciones para cada mini-lote de tamaño 64, lo que mejora la estabilidad del aprendizaje y acelera la convergencia sin afectar a las dimensiones.

- **Primera capa MaxPooling1D**

Reduce la dimensionalidad por medio de la selección del valor máximo en regiones de tamaño 2, tal y como se observa en la ecuación 2.4 donde  $P_{pool}$  representa el tamaño del pooling y L es el tamaño de la entrada. Se obtiene un tensor de salida con valores  $(L', n_{filtros})$

$$L' = \frac{L}{P_{pool}} \quad (2.4)$$

$$L' = \frac{298}{2} = 149$$

Salida: (149,32)

- **Segunda capa Conv1D**

Extrae características más complejas aplicando 64 filtros con un kernel de tamaño 3.

$$L'_2 = \frac{149 + 2(0) - 3}{1} + 1 = 147$$

Número de filtros: 64

Salida: (147,64)

- **Batch Normalization y segunda capa MaxPooling1D**

El *Batch Normalization* se mantiene igual que antes sin alterar las dimensiones de entrada y salida, mientras que MaxPooling1D reduce la dimensionalidad según el tamaño del *pooling*.

$$L'_2 = \frac{147}{2} = 73.5 \approx 73$$

Salida: (73,64)

- **Tercera capa Conv1D**

Se detectan características más avanzadas ahora con 128 filtros

$$L'_3 = \frac{73 + 2(0) - 3}{1} + 1 = 71$$

Número de filtros: 71

Salida: (71,128)

Batch Normalization y MaxPooling1D

Se normalizan las activaciones sin afectar a las dimensiones, mientras que *MaxPooling1D* reduce las dimensiones a la mitad debido al tamaño del *pooling*.

$$L_{salida} = \frac{71}{2} = 35.5 \approx 35$$

- **Primera capa LSTM**

Se procesa la secuencia temporal para capturar relaciones en el tiempo, devolviendo una secuencia completa de salida (una por cada paso)

Salida: (35,128)

Dropout

Se estableció un *dropout=0.4* con el fin de que cada neurona del modelo tenga un 40% de probabilidades de desactivarse para prevenir el sobreajuste. Esto no afecta a las dimensiones.

- **Segunda capa LSTM**

*LSTM(64, activation = 'tanh', return\_sequences = False)*

Se comprime la secuencia a un único vector de tamaño 64, representando la información de toda la señal, utilizando la tangente hiperbólica como función de activación para transformar los valores de salida a un rango entre -1 y 1, activando las células internas LSTM y permitiendo que la red capture información a largo plazo de la secuencia. Se utilizó *return\_sequences=False* para devolver solo el último estado de la secuencia, lo que es adecuado para tareas de clasificación.

Salida: (64)

- **Dropout**

Se volvió a aplicar un *dropout=0.4*, la cual no afecta a las dimensiones de salida

- **Capa densa final**

Clasifica la entrada en una de las 5 clases usando la función de activación *softmax*.

Salida: (5)

En la Tabla 2.10 se observa un resumen de la arquitectura del modelo, junto con las capas y dimensiones de entrada y salida

**Tabla 2.10**

*Resumen de las dimensiones del modelo*

Capa	Dimensión de entrada	Dimensión de salida
Conv1D (32 filtros)	(300,1)	(298,32)
MaxPooling1D	(298,32)	(149,32)
Conv1D (64 filtros)	(149,32)	(147,64)
MaxPooling1D	(147,64)	(73,64)
Conv1D (128 filtros)	(73,64)	(71,128)
MaxPooling1D	(71,128)	(35,128)
LSTM (128 unidades)	(35,128)	(35,128)
Dropout (p=0.4)	(35,128)	(35,128)
LSTM (64 unidades)	(35,128)	(64)
Dropout (p=0.4)	(64)	(64)
Dense (5 clases)	(64)	(5)

- **Optimizador**

Se configuró el optimizador Adam con un *learning rate* de 0.0005. Este optimizador ajustará los pesos del modelo durante el entrenamiento usando un método adaptativo que combina momentos de primer y segundo orden.

- **Compilación del modelo**

Para compilar el modelo se utilizó el optimizador Adam descrito en la sección anterior; se usó también una función de pérdida denominada *categorical\_crossentropy*, la cual es adecuada para problemas de clasificación multiclase con salidas codificadas en *one-hot encoding* (valores discretos representados como vectores binarios). Por último, se definió

como métrica el *accuracy* para evaluar el porcentaje de predicciones correctas durante el entrenamiento y validación.

- **Callbacks**

Se ejecutaron ciertas funciones en momentos específicos del entrenamiento del modelo de manera automática, esto con el fin de personalizar y controlar aspectos del proceso de entrenamiento, como los que se mencionan a continuación:

- ***ReduceLROnPlateau***: Reduce el *learning rate* a la mitad si no mejora la pérdida en el conjunto de validación (*val\_loss*) después de 5 épocas consecutivas, con un límite mínimo de  $10^{-6}$ , donde a partir de este valor se dejará de actualizar el *learning rate*, pero seguirá aplicándose para cada época. Esto ayuda a evitar estancamiento en el entrenamiento.

```
reduce_lr = ReduceLROnPlateau(monitor = 'val_loss', factor = 0.5, patience
                              = 5, min_lr = 1e - 6, verbose = 1)
```

- ***ModelCheckpoint***: Guarda el modelo con el mejor desempeño en términos de *accuracy* en validación (*val\_accuracy*). Útil para restaurar el modelo óptimo después del entrenamiento.

```
checkpoint = ModelCheckpoint("mejor_modelo_emg_optimizado_v1_1.keras", monitor
                             = "val_accuracy", save_best_only = True, mode = "max", verbose = 1)
```

- **Cálculo de los pesos de clases**

```
class_weights = compute_class_weight(class_weight = 'balanced',
                                     classes = np.unique(np.argmax(y_train, axis = 1)),
                                     y = np.argmax(y_train, axis = 1))
```

El cálculo de los pesos de clases ayuda a mitigar el impacto del desbalanceo en los datos durante el entrenamiento. La función *compute\_class\_weight* asigna pesos inversamente proporcionales a la frecuencia de cada clase, dando mayor peso a las menos representadas y menor a las más frecuentes. Esto evita que el modelo se sesgue hacia las clases mayoritarias.

Para calcular los pesos correctamente,  $np.argmax(y_{train}, axis = 1)$  convierte las etiquetas one-hot en valores enteros, reflejando la distribución real de las clases.

- **Entrenamiento del modelo**

Se entrenó el modelo durante 50 épocas utilizando un tamaño de lote de 64 ejemplos procesados simultáneamente. Para esta fase se utilizó el 70% del dataset que corresponde al conjunto de entrenamiento, y el 15% correspondiente al conjunto de validación, tal y como se lo mencionó en secciones anteriores. Además, se utilizaron los *callbacks* y el balanceo de clases que se mencionó previamente para garantizar resultados óptimos tanto en el entrenamiento como en la validación.

- **Evaluación del modelo**

Para esta sección se consideró las pérdidas y la precisión, utilizando el conjunto de *testing* que representa el 15% del total del *dataset*.

- **Loss:** Mide qué tan bien se ajusta el modelo a los datos de prueba.
- **Accuracy:** Proporción de predicciones correctas en el conjunto de prueba.

## 2.6 Diseño mecánico

El diseño mecánico garantiza que la prótesis de brazo realice movimientos funcionales y precisos. Se partió de un modelo 3D previo, optimizando para este proyecto, con especificaciones detalladas en las articulaciones de los dedos y el pulgar. Herramientas como *Blender* permitieron validar las posiciones generadas por el modelo de *Deep learning*, asegurando un diseño eficiente y libre de interferencias.

### 2.6.1 Antecedente

El diseño mecánico del sistema se basa en el trabajo desarrollado previamente en un proyecto de materia integradora, en el que se creó el modelo 3D inicial de la prótesis de mano utilizando Autodesk Inventor y que se puede observar en la Figura 2.11. Este modelo fue optimizado y exportado a *Blender*, donde se realizaron simulaciones y animaciones que representaban las diferentes posiciones necesarias para el modelo de deep learning. El

presente proyecto comienza a partir de este diseño previo, aprovechando su base para desarrollar las mejoras y ajustes requeridos.

### **2.6.2 Especificaciones del diseño mecánico**

El modelo incluye todas las articulaciones necesarias para replicar los movimientos requeridos, con un enfoque especial en la funcionalidad y realismo de los movimientos de los dedos y la mano.

Cada uno de los cuatro dedos principales está dividido en tres secciones:

- Unión con la mano: Permite un rango de movimiento de 90 grados hacia abajo.
- Primera articulación (a mitad del dedo): Tiene un rango de movimiento de 30 grados hacia abajo.
- Segunda articulación (punta del dedo): También cuenta con un rango de movimiento de 30 grados hacia abajo.

El pulgar, por su diseño particular, tiene únicamente dos articulaciones:

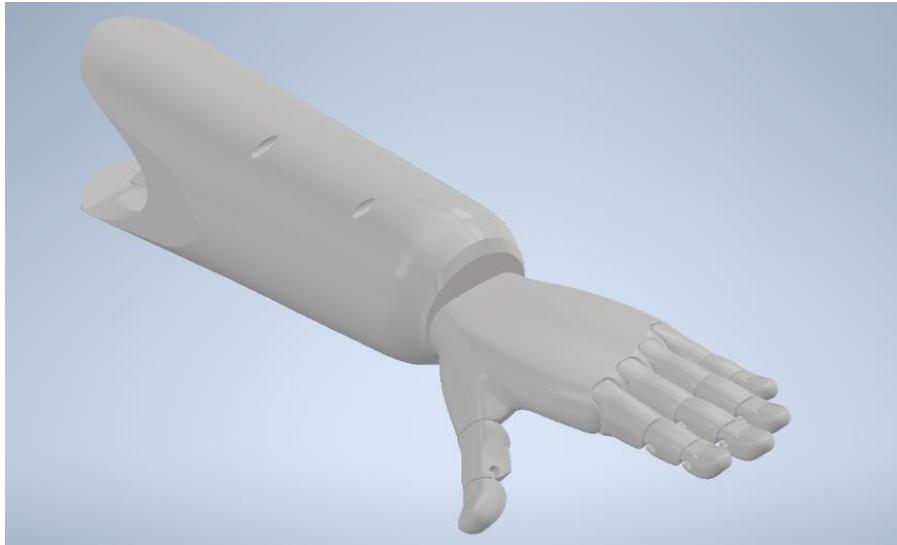
- Unión con la mano: Rango de movimiento de 30 grados hacia abajo.
- Articulación en la punta del dedo: Rango de movimiento de 30 grados hacia abajo.

Estas especificaciones aseguran que cada articulación pueda realizar movimientos funcionales y naturales, alineados con los requerimientos del modelo de *deep learning*.

La simulación realizada en *Blender* permite validar de manera precisa las posiciones generadas por el modelo de *deep learning*, asegurando que las piezas de la prótesis interactúen correctamente sin choques ni interferencias no deseadas durante su operación (véase Figura 2.11).

**Figura 2.11.**

*Modelo CAD de la prótesis*



### **2.6.3 Compartimentos integrados para componentes electrónicos**

La prótesis cuenta con dos compartimentos integrados en su diseño, los cuales están destinados a alojar todos los componentes electrónicos necesarios para su funcionamiento. Estos compartimentos están estratégicamente ubicados para no interferir con los movimientos de la prótesis y garantizar un equilibrio adecuado en su distribución de peso.

Cada compartimento está protegido por una tapa atornillada que se ajusta firmemente a la estructura externa de la prótesis. Este diseño no solo asegura que los componentes electrónicos estén protegidos contra golpes, polvo y otros factores externos, sino que también facilita el acceso para mantenimiento o reemplazo de los elementos internos cuando sea necesario.

La integración de estos compartimentos en el diseño mecánico de la prótesis elimina la necesidad de una caja externa adicional, optimizando así el espacio y manteniendo un perfil ergonómico y estético en el dispositivo final.

## **2.7 Diseño conceptual**

En la Figura 2.12 se muestra el diseño conceptual del sistema desarrollado, en el cual se integran los componentes principales necesarios para la solución seleccionada. Este diseño

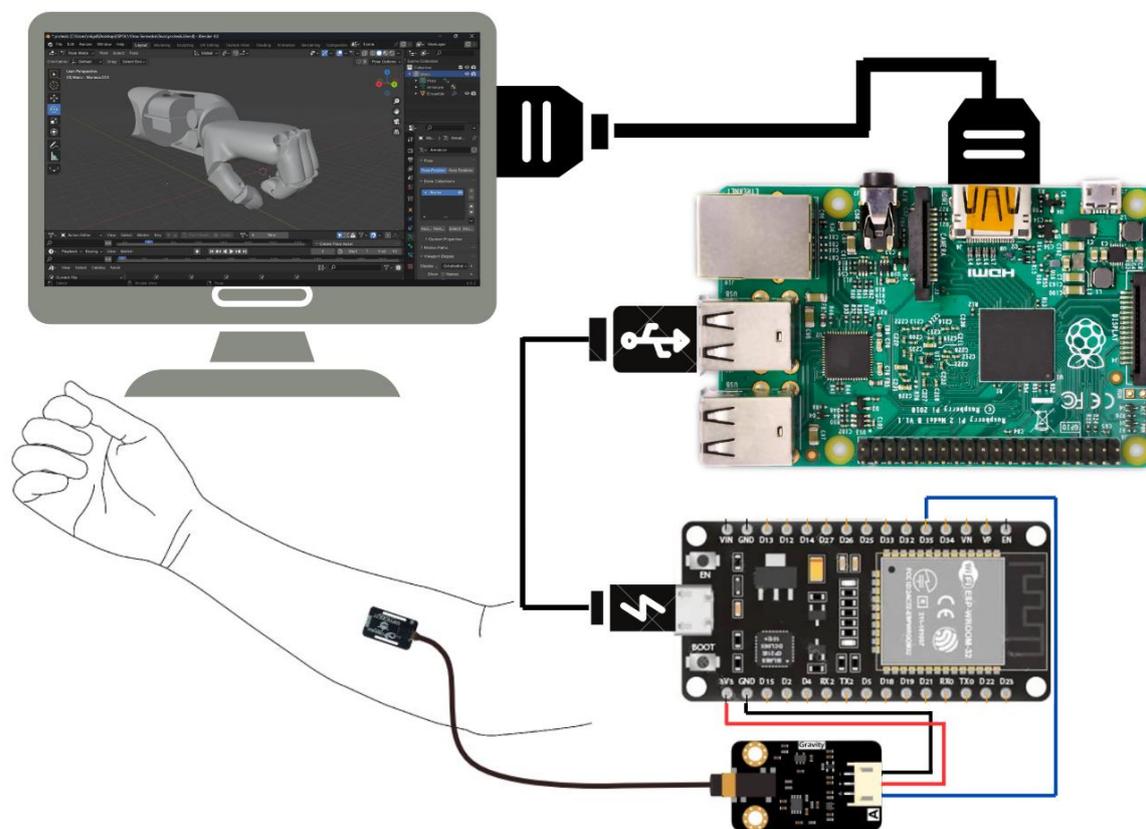
incluye un sensor EMG SEN0240 conectado al brazo del paciente, encargado de captar las señales musculares a través de electrodos. Dichas señales son transmitidas al microcontrolador ESP32, el cual se encarga únicamente de recolectar las señales y transmitir las a la Raspberry Pi 4.

La Raspberry Pi 4, que alberga el modelo de deep learning entrenado previamente en una computadora, interpreta las señales musculares recibidas del ESP32 y las traduce en movimientos específicos. Estos movimientos son luego simulados en tiempo real mediante un modelo digital de un brazo en el software *Blender*. Este enfoque permite observar y analizar de manera adecuada las acciones que se esperan implementar en un sistema físico, asegurando así el funcionamiento correcto del diseño antes de pasar a la fase de prototipado físico.

Este diseño conceptual refleja el resultado del análisis y evaluación de alternativas, considerando tanto los requerimientos técnicos como las limitaciones del proyecto. La implementación del modelo en la Raspberry Pi y la simulación en *Blender* garantizan la viabilidad y funcionalidad del sistema, estableciendo una base sólida para su posterior desarrollo.

**Figura 2.12.**

*Ilustración de la ejecución del proyecto*



## **Capítulo 3**

### 3. Análisis de resultados

Se presenta un resumen detallado del desempeño obtenido con el modelo de *deep learning* para la clasificación de señales mioeléctricas, así como las métricas evaluadas y las observaciones relevantes sobre su comportamiento en las distintas clases de movimiento. Este análisis busca identificar fortalezas y limitaciones para futuras mejoras.

#### 3.1 Diseño mecánico

El diseño mecánico de la prótesis y sus componentes incluye tanto el modelo 3D como la estructura que aloja los elementos electrónicos. Se explican las decisiones técnicas relacionadas con la funcionalidad, modularidad y durabilidad, garantizando la integración óptima entre los sistemas de predicción y los mecanismos físicos.

##### 3.1.1 Modelo 3D de la prótesis y conexión con el sistema de predicciones

El diseño mecánico de la prótesis se basa en un modelo 3D creado en *Blender*, que está completamente integrado con el sistema de predicción en tiempo real. Este modelo simula los movimientos de la mano y es controlado por las señales EMG clasificadas mediante el modelo de *deep learning*. A través de un script de *Python*, las predicciones obtenidas del ESP32 se traducen en movimientos realistas dentro de *Blender*, reflejando las posiciones de la mano de acuerdo con las señales musculares.

- **Posiciones de descanso y puño:** Las predicciones de las posiciones de descanso y puño se reflejaron con gran precisión, lo que permite una animación fluida y realista de estos movimientos en el modelo 3D.
- **Modelado detallado:** El modelo 3D de la prótesis es detallado y se adapta bien a los movimientos capturados por el sistema de predicción, proporcionando un resultado visualmente coherente con las señales EMG clasificadas.
- **Limitaciones detectadas:**
  - **Posición anular:** Durante las pruebas, se observó que la posición anular en el modelo a veces se confundía con la posición de puño. Este error podría estar

relacionado con la variabilidad de las señales EMG al ejecutar ambos movimientos, lo que genera imprecisiones en el modelo. Se recomienda un análisis más profundo de las señales y la integración de técnicas de filtrado para mejorar la precisión en esta área.

**Figura 3.1.**

*Prótesis en Software Blender*



### **3.1.2 Propuesta de Mejora de Prótesis Física**

En el desarrollo previo de la prótesis física presentado en el trabajo de Materia Integradora, se implementó un diseño basado en el uso de dos servomotores. Sin embargo, esta configuración limitaba significativamente la movilidad de la prótesis, impidiendo la activación individual de todos los dedos y restringiendo la capacidad de realizar movimientos más complejos. Con el objetivo de mejorar la funcionalidad y ampliar el rango de movimientos de la prótesis, se propone un nuevo diseño que incorpora tres servomotores en lugar de dos. Esta modificación permitirá distribuir mejor la fuerza y el control sobre los diferentes dedos, facilitando la ejecución de los movimientos necesarios según los requerimientos establecidos en la presente investigación. La Figura xx ilustra el diseño propuesto, donde se pueden apreciar los tres servomotores ubicados para optimizar la movilidad de la prótesis. Este diseño

busca mejorar la precisión y versatilidad en la manipulación de objetos, acercándose más a los movimientos naturales de una mano humana.

En el apéndice C, se incluye el plano detallando este diseño, donde se especifican las dimensiones y los componentes utilizados en la mejora de la prótesis. Este plano servirá como referencia para futuras implementaciones y ajustes en el desarrollo del dispositivo.

### 3.2 Diseño electrónico

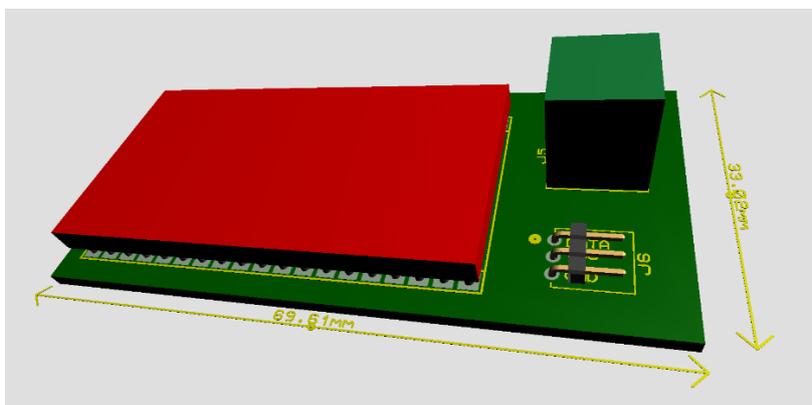
El diseño electrónico del sistema juega un papel crucial en la captura, procesamiento y transmisión de las señales EMG, que son la base para controlar los movimientos de la prótesis. Se abarca desde la creación de la PCB, que integra los componentes esenciales como el ESP32 y el sensor SEN0240, hasta las especificaciones técnicas que garantizan un funcionamiento eficiente y adaptable del sistema.

#### 3.2.1 Diseño de la PCB

El diseño final de la PCB fue realizado considerando las características y requisitos mencionados, logrando una solución compacta y funcional. Este diseño fue modelado en 3D para visualizar la disposición de los componentes y garantizar su correcta integración. En la Figura 3.2 se presenta el modelo 3D de la PCB, el cual proporciona una representación precisa del diseño físico y permite identificar posibles mejoras antes de proceder a su fabricación.

**Figura 3.2.**

*Modelo 3D de la PCB*



### 3.2.2 Toma de datos

La toma de datos se realizó conectando el sensor SEN0240 al antebrazo para capturar señales EMG asociadas a cinco posiciones musculares: descanso, anular, anular-pinza, puño e índice.

#### Pasos realizados:

1. **Conexión del sensor:** Los electrodos se colocaron en el músculo correspondiente, conectados al sensor mediante los cables adecuados.
2. **Transmisión de datos:** Las señales fueron capturadas por el sensor y enviadas al ESP32 a través del pin 35.
3. **Preprocesamiento:**
  - Filtrado de ruido realizado por la placa del sensor para garantizar datos consistentes.
  - Preparación de los datos para el entrenamiento del modelo de *deep learning*.

#### Señales capturadas

Figura 3.4. Señal capturada de descanso

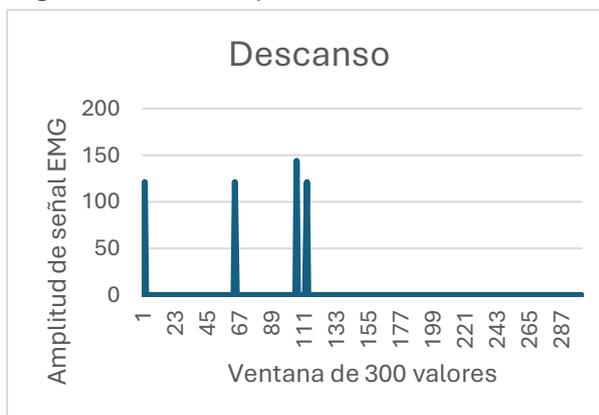


Figura 3.3. Señal capturada de puño

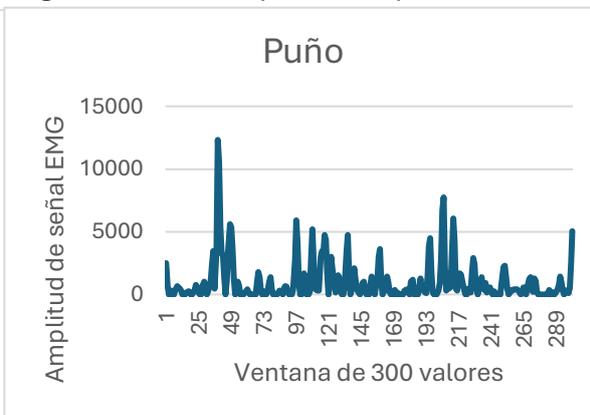


Figura 3.6. Señal capturada de pinza anular

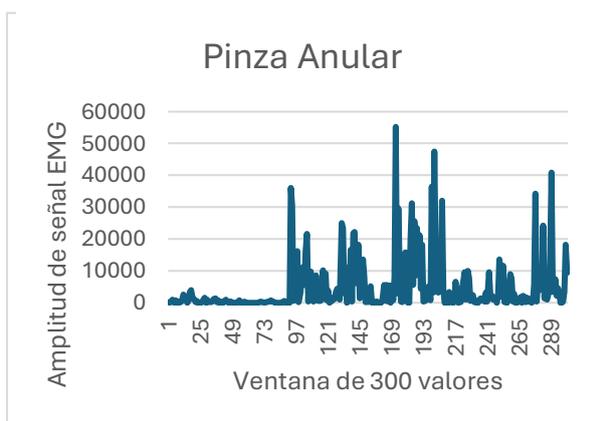
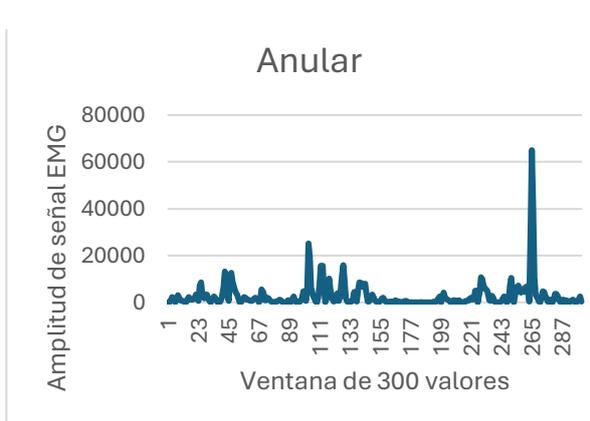
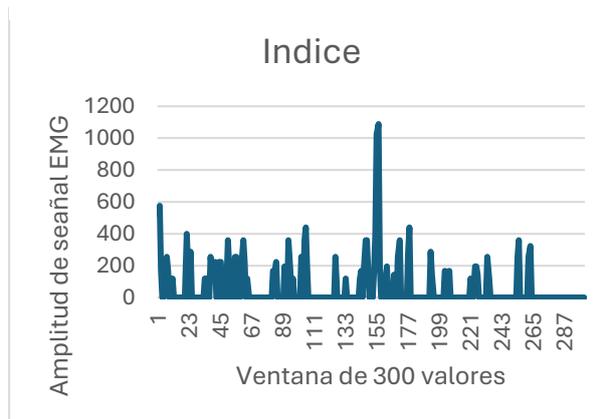


Figura 3.5. Señal capturada de anular



**Figura 3.7.** Señal capturada de índice

### 3.3 Análisis de costos

El costo total del proyecto se desglosa en dos categorías principales: costos de componentes electrónicos y costos de desarrollo. Este análisis permite identificar los recursos invertidos en el diseño y la construcción del sistema y establecer una comparación con las prótesis comerciales disponibles en el mercado.

#### 3.3.1 Costos de componentes electrónicos

**Tabla 3.1.***Tabla de componentes electrónicos*

Componente	Cantidad	Costo Unitario (USD)	Costo Total (USD)
ESP32	1	\$10.00	\$10.00
Sensor SEN0240	1	\$49.50	\$49.50
Fabricación de la PCB	1	\$20.00	\$20.00
Borneras	1	\$2.00	\$2.00
Sockets	1	\$2.00	\$2.00
Subtotal (Electrónicos)			\$83.50

### 3.3.2 Costos de desarrollo

**Tabla 3.2.**

*Tabla de costos de desarrollo*

Elemento	Cantidad	Costo unitario (USD)	Costo total (USD)
Material de impresión PLA	1 kg	\$20.00	\$20.00
Mano de obra técnica	1 equipo	\$500.00	\$500.00

**Subtotal (desarrollo):** \$520.00 USD

Nota: La mano de obra técnica se valora en \$500.00 USD, considerando el trabajo colaborativo realizado por los integrantes del equipo en actividades como diseño, programación, impresión y pruebas. Este costo refleja el valor del tiempo y la experiencia técnica necesaria para completar el proyecto.

### 3.3.3 Costos de fabricación

Los costos de fabricación corresponden a los materiales y accesorios utilizados en el diseño mecánico y electrónico de la prótesis, basados en un proyecto previo de materia integradora. Estos costos ya incluyen los componentes necesarios para la funcionalidad básica de la prótesis, tal y como se muestra en la Tabla 3.3.

**Tabla 3.3**

*Costos de fabricación*

Segmento	Costo (USD)
Mecánica	\$58.60
Electrónica	\$126.40
Accesorios	\$183.04
Subtotal Fabricación	\$368.04

### 3.3.4 Costo total del proyecto

El costo total del sistema desarrollado asciende a \$971.54 USD, incluyendo los costos de componentes electrónicos, materiales de desarrollo, mano de obra técnica y fabricación, según se observa en la Tabla 3.4.

**Tabla 3.4**

*Costo total del proyecto*

Categoría	Costo (USD)
Componentes electrónicos	\$83.50
Desarrollo	\$520.00
Fabricación	\$368.04
Costo Total	\$971.54

Un desglose más detallado de los costos del proyecto se los puede observar en el Apéndice B.

### 3.3.5 Comparación con prótesis comerciales

Las prótesis comerciales disponibles en el mercado tienen costos significativamente más altos. Por ejemplo, se encontró una prótesis biónica considerada como una de las más accesibles a nivel mundial, con un precio de \$7,995 USD. Esta prótesis utiliza sensores embebidos en el socket para leer datos musculares del miembro residual y controlar una variedad de agarres y movimientos. Además, destaca por contar con un número mayor de sensores de los estrictamente necesarios, asegurando un contacto constante con los músculos para mejorar la activación de los movimientos.

En comparación, el sistema desarrollado en este proyecto tiene un costo total de \$971.54 USD, equivalente a aproximadamente el 12.15% del costo de la mencionada prótesis. Aunque el sistema aquí presentado es más sencillo en términos de funcionalidad, es

capaz de ejecutar movimientos básicos de una prótesis utilizando señales EMG procesadas, demostrando ser una solución modular, económica y escalable.

El diseño propuesto se presenta como una alternativa accesible para personas con limitaciones económicas, sin comprometer la funcionalidad esencial requerida para movimientos básicos. La considerable diferencia en costos refuerza su viabilidad como opción para el desarrollo y uso en comunidades con recursos limitados.

### **3.4 Análisis de resultados del modelo**

En esta sección se presentan los resultados obtenidos del modelo de *Deep learning* utilizado para clasificar las señales EMG y su integración con el modelo 3D de la prótesis. Se evalúan métricas como precisión, recall, F1-score y AUC, y se discuten las limitaciones observadas durante las pruebas en tiempo real, destacando áreas de mejora en la interpretación de señales EMG y su impacto en el control de la prótesis.

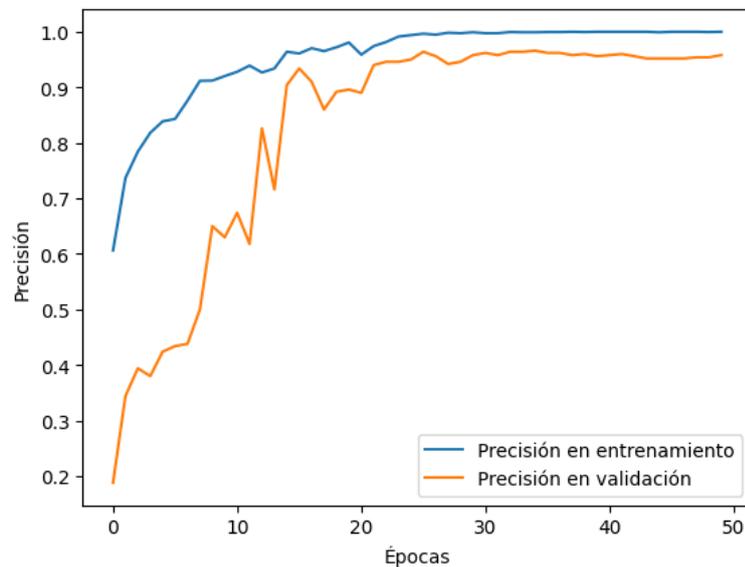
#### **3.4.1 Desempeño del modelo de deep learning**

El modelo de *deep learning* fue entrenado utilizando *TensorFlow* en Google Colab con las señales EMG preprocesadas. Se extrajeron características locales utilizando redes convolucionales en 1D (Conv1D) y se modelizaron dependencias temporales a través de capas LSTM.

Primeramente, se analizó el comportamiento del modelo en la fase de entrenamiento y validación tal y como muestra la Figura 3.8 que la precisión del modelo mejora rápidamente en las primeras épocas, pero hay una brecha notable entre la precisión en entrenamiento y validación, lo que indica un posible sobreajuste. La curva de validación se estabiliza después de fluctuaciones iniciales, alcanzando un valor constante por debajo de la precisión en entrenamiento, lo que sugiere que el modelo ha alcanzado su límite de generalización con la configuración actual y podría ser necesario ajustar los hiperparámetros del modelo o recopilar más datos para mejorar su generalización.

**Figura 3.8.**

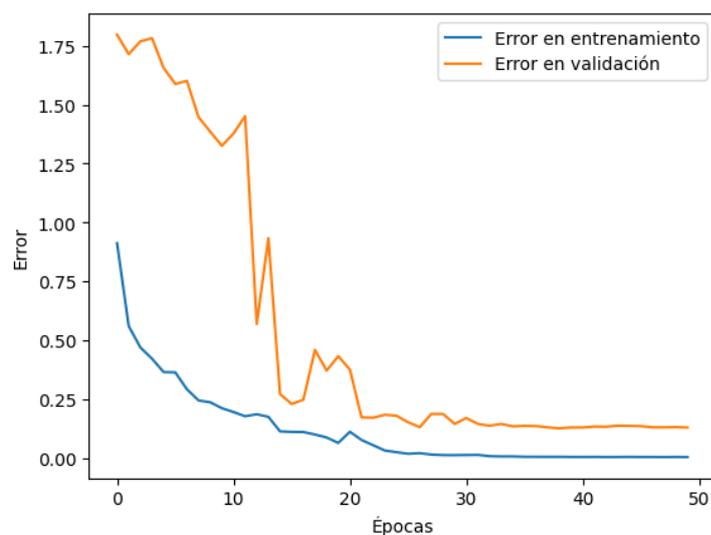
*Gráfica de precisión en entrenamiento y validación*



Luego, la Figura 3.9 muestra la evolución del error de entrenamiento disminuye constantemente, pero el error de validación se estabiliza después de 20 épocas, lo que indica sobreajuste. Esto sugiere que el modelo está memorizando los datos de entrenamiento y pierde capacidad de generalización; no obstante, el valor de error de validación es muy bajo por lo que también representa un resultado positivo para el comportamiento del modelo.

**Figura 3.9.**

*Gráfica de pérdidas en entrenamiento y validación*



Las métricas obtenidas son las que se muestran en la Tabla 3.5 y se basan en los siguientes movimientos:

- **Clase 0:** Descanso
- **Clase 1:** Índice
- **Clase 2:** Anular-Pinza
- **Clase 3:** Puño
- **Clase 4:** Anular

**Tabla 3.5.**

Reporte de clasificación

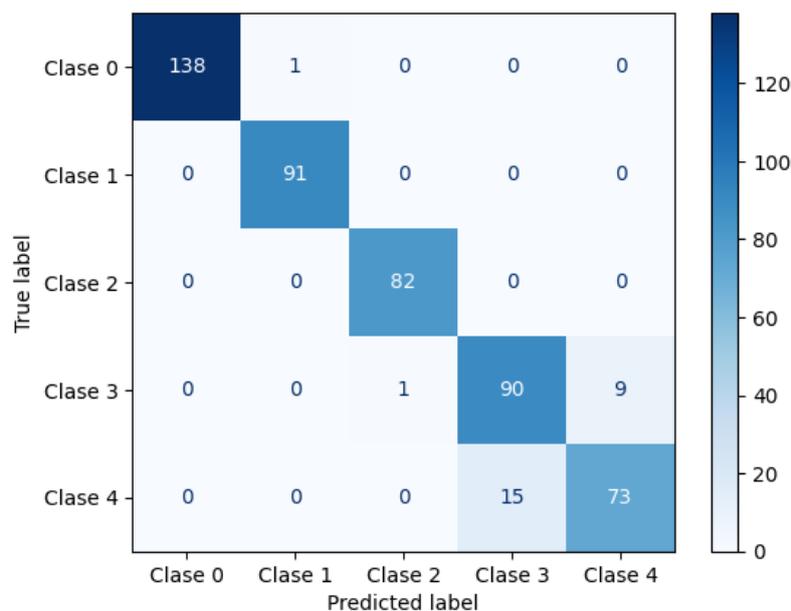
Clase	Precisión	Recall	F1-score	Support
0	1	0.99	1	139
1	0.99	1	0.99	91
2	0.99	1	0.99	82
3	0.86	0.9	0.88	100
4	0.89	0.83	0.86	88
Accuracy			0.95	500
Macro avg	0.94	0.94	0.94	500
Weighted avg	0.95	0.95	0.95	500

La Tabla 3.5 refleja un desempeño excelente, con 95% de precisión global. Las clases 0, 1 y 2 tienen métricas casi perfectas, mientras que las clases 3 y 4 muestran áreas de mejora con F1-score de 0.88 y 0.86, respectivamente. Esto indica dificultades moderadas en estas clases, probablemente por similitudes en los datos o desbalance. Ajustes específicos podrían mejorar su desempeño y el balance general del modelo.

### 3.4.2 Matriz de confusión

Figura 3.10.

Matriz de confusión



La matriz de confusión (Figura 3.10) muestra un desempeño excelente en las clases 0, 1 y 2, con predicciones 100% correctas. Sin embargo, hay cierta confusión entre las clases 3 y 4, ya que 9 instancias de clase 3 fueron clasificadas como clase 4 y 15 de clase 4 como clase 3. Esto sugiere que estas clases tienen características similares o insuficiencia de datos para diferenciarlas. La precisión general del modelo se basa en la ecuación 3.1, en la que se obtiene un 94.8% que demuestra un buen rendimiento del modelo, pero mejorar la representación de las clases 3 y 4 en el conjunto de datos podría aumentar su desempeño.

$$Accuracy = \frac{Total\ de\ predicciones\ correctas}{Total\ de\ predicciones\ realizaadas}$$

(3.1)

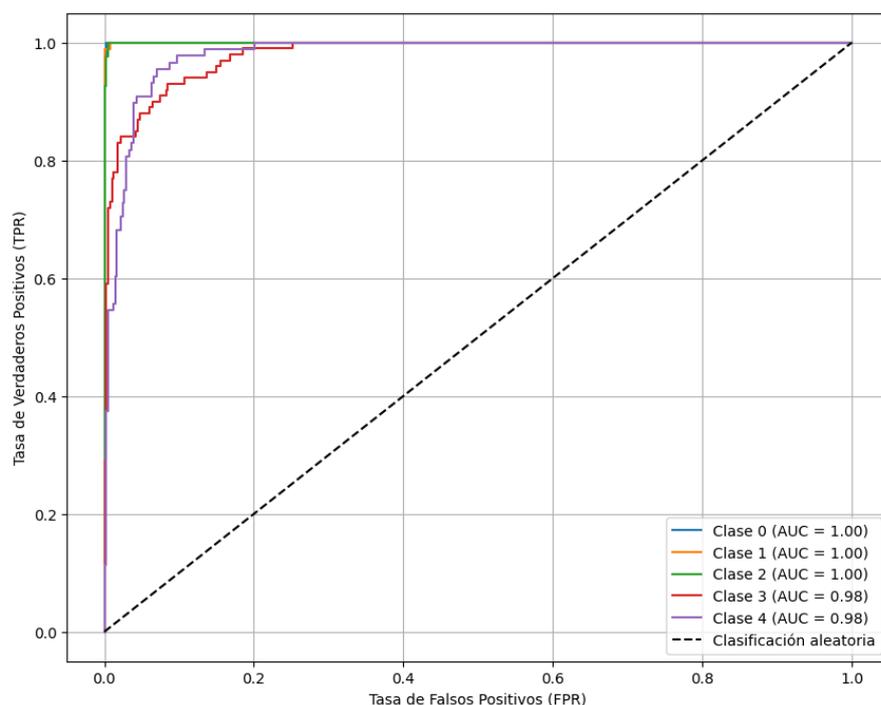
$$Accuracy = \frac{138+91+82+90+73}{138+91+82+90+73+1+1+9+15} = \frac{474}{500} = 0.948$$

### 3.4.3 Curvas ROC

Las curvas ROC (Receiver Operating Characteristic) son herramientas que evalúan la capacidad de un modelo de clasificación en diferentes umbrales de decisión.

**Figura 3.11.**

*Curvas ROC para cada clase*



La Figura 3.11 muestra que el modelo tiene un rendimiento excelente, con un Área Bajo la Curva (AUC) de 1.00 para las clases 0,1 y 2, lo que indica una clasificación perfecta y un AUC de 0.98 para las clases 3 y 4, lo que sigue siendo muy bueno, aunque muestra pequeñas dificultades para diferenciarlas de otras clases. En general, el modelo supera ampliamente la clasificación aleatoria (AUC = 0.5), pero las clases 3 y 4 presentan una ligera caída en el AUC.

### 3.5 Análisis de resultados de la simulación

En esta sección se presentan y analizan los resultados obtenidos de la simulación del sistema propuesto, que integra un modelo de Deep learning para la clasificación de señales EMG y un modelo 3D en *Blender*. El análisis se centra en la precisión del sistema en tiempo

real, el desempeño del modelo en distintos movimientos y las limitaciones en la integración de los componentes electrónicos y el software. Estos resultados permiten evaluar la viabilidad del sistema y guiar futuras mejoras.

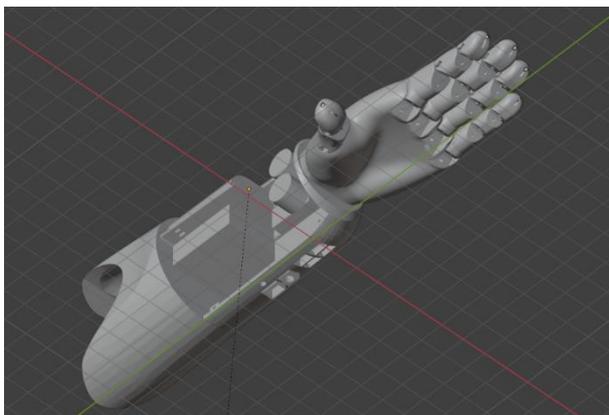
### **3.5.1 Pruebas en tiempo real**

Durante las pruebas en tiempo real, el sistema demostró un desempeño notable en ciertas áreas, aunque también se identificaron limitaciones específicas. Entre las fortalezas destacadas, las posiciones de descanso y puño se predijeron con alta precisión, reflejando la capacidad del modelo para interpretar de manera efectiva las señales EMG en estos casos. Sin embargo, se observaron algunas limitaciones durante los movimientos rápidos, como confusiones frecuentes entre las posiciones de anular y puño, lo que sugiere un posible reto en la discriminación de señales similares en tiempo reducido. Asimismo, la posición de índice presentó inconsistencias, ya que no siempre fue registrada correctamente, lo que podría atribuirse a variaciones en la intensidad de las señales capturadas por el sensor. Estos hallazgos subrayan tanto el potencial del sistema como las áreas de mejora para optimizar su desempeño en situaciones más complejas.

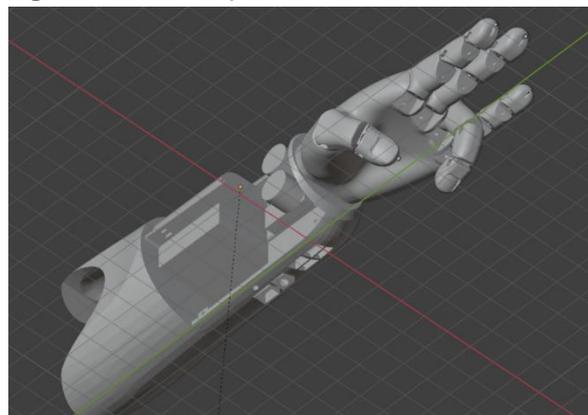
### **3.5.2 Integración con blender**

El script de Python permitió traducir las predicciones en tiempo real al modelo 3D de prótesis en *Blender*. El sistema reflejó los movimientos musculares en poses precisas.

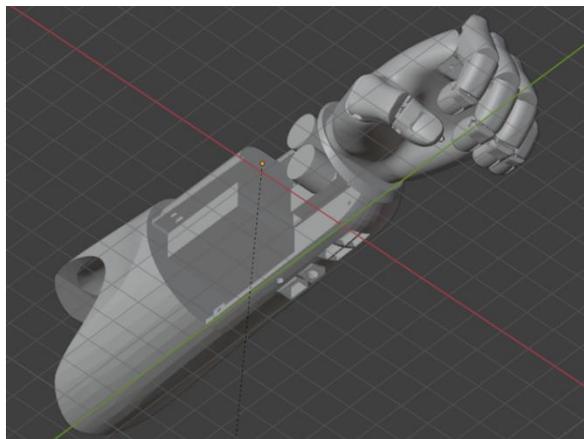
**Figura 3.13.** *Pose Descanso*



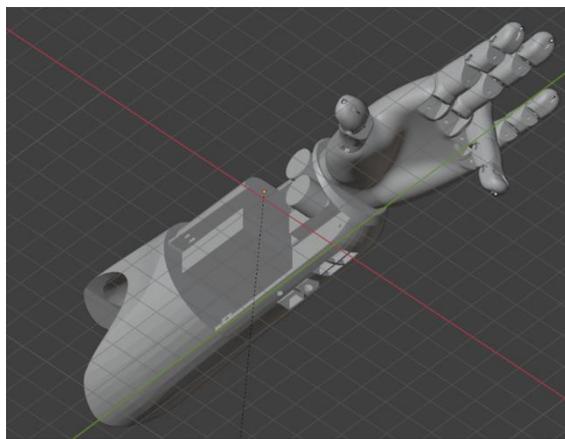
**Figura 3.12.** *Pose pinza-anular*



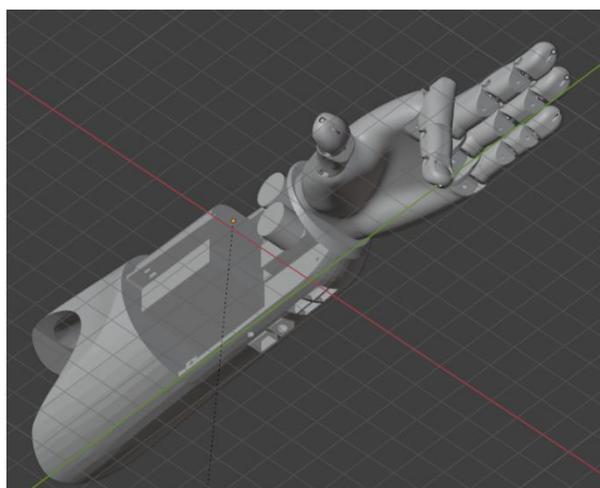
**Figura 3.14.** *Pose puño*



**Figura 3.15.** *Pose anular*



**Figura 3.16.** *Pose índice*



## Capítulo 4

## 4. Conclusiones y recomendaciones

El presente trabajo estableció una metodología innovadora que combina el uso de deep learning para interpretar señales mioeléctricas y su integración con una simulación 3D en *Blender*. A continuación, se exponen las principales conclusiones obtenidas y las recomendaciones para futuros trabajos, basadas en los resultados alcanzados y en las fortalezas y limitaciones identificadas.

### 4.1 Conclusiones

- El modelo fue capaz de clasificar cinco movimientos básicos con un nivel aceptable de precisión, cumpliendo el objetivo de simular movimientos de una prótesis en tiempo real.
- Se logró diseñar un sistema funcional que permite que las señales mioeléctricas captadas por sensores sean procesadas por un modelo de *deep learning* y reflejadas en una simulación 3D en *Blender*, demostrando la viabilidad técnica de este enfoque.
- Este trabajo destaca por su carácter multidisciplinario, integrando conocimientos de inteligencia artificial, biomecánica y animación 3D, sentando las bases para futuros desarrollos en simulación de prótesis y entornos virtuales.
- Comparado con otras investigaciones, esta propuesta contribuye al avance del uso de simulaciones en lugar de prototipos físicos, reduciendo costos y permitiendo una evaluación inicial de los sistemas de control de prótesis en entornos virtuales.
- El sistema desarrollado representa un avance significativo que puede ser mejorado y adaptado para trabajar con modelos de prótesis físicas o ampliarse a otras aplicaciones, como la rehabilitación virtual o el entrenamiento en entornos controlados.

## 4.2 Recomendaciones

- Explorar arquitecturas más ligeras y optimizadas para reducir el peso del modelo, los tiempos de predicción y aumentar la precisión en la clasificación de las señales mioeléctricas.
- Recopilar un conjunto de datos más amplio y diverso para mejorar la capacidad de generalización del modelo, considerando variaciones en las señales EMG según diferentes usuarios, posiciones y contextos. Un ajuste en los hiperparámetros o en la arquitectura también se puede traducir en un mejor desempeño del modelo.
- Expandir la clasificación a más movimientos para simular gestos más complejos, aumentando la funcionalidad y aplicabilidad del sistema.
- Utilizar sensores mioeléctricos de mayor precisión para mejorar la calidad de las señales capturadas, así como microcontroladores más potentes para reducir la latencia en la adquisición de datos.
- Enriquecer la simulación en *Blender* con detalles visuales más realistas, lo que permitiría evaluar no solo el funcionamiento técnico del sistema, sino también aspectos ergonómicos y estéticos.
- Considerar la transición de la simulación a un prototipo físico que permita probar el control de prótesis reales, evaluando el sistema en un entorno más cercano a su uso final

## Referencias

- [1] S. Roa Chejín, “GK,” 22 Agosto 2019. [En línea]. Available: <https://gk.city/2019/07/28/protesis-ecuador-discapacidad/>. [Último acceso: 22 Octubre 2024].
- [2] D. Araque y L. Barona, “Repositorio Digital - EPN,” 9 Octubre 2024. [En línea]. Available: <http://bibdigital.epn.edu.ec/handle/15000/25882>. [Último acceso: 26 Octubre 2024].
- [3] Organización de las Naciones Unidas, “Objetivos de Desarrollo Sostenible,” 2024. [En línea]. Available: <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>. [Último acceso: 4 Noviembre 2024].
- [4] Ministerio de Salud Pública, “Evaluación de resultados del proyecto “Atención Integral e Integradora a Personas con Discapacidad, Rehabilitación y Cuidados Especiales en Salud a Nivel Nacional”,” Secretaría Nacional de Planificación , Quito, 2022.
- [5] Youtopia+Rett, “Solo un 24% de las personas con discapacidad tiene empleo,” 3 Octubre 2024. [En línea]. Available: <https://youtopiaecuador.com/ecuador-discapacidad-empleo-cifras-mercado-laboral/>. [Último acceso: 7 Noviembre 2024].
- [6] NIH, “Las prótesis a través del tiempo,” MedlinePlus, 11 May 2023. [En línea]. Available: <https://magazine.medlineplus.gov/es/art%C3%ADculo/las-protesis-a-traves-del-tiempo>. [Último acceso: 30 October 2024].
- [7] “Reseña histórica de las prótesis,” UNAM, 9 Mar 2008. [En línea]. Available: <https://www.revista.unam.mx/vol.6/num1/art01/art01-1c.htm>. [Último acceso: 31 Oct 2024].

- [8] K. Norton, "History of Prosthetic Limbs," Amputee Coalition, 25 November 2007. [En línea]. Available: <https://amputee-coalition.org/resources/spanish-history-prosthetics/>. [Último acceso: 30 October 2024].
- [9] "Modelos de prótesis de brazo y mano," Proviem, 11 March 2022. [En línea]. Available: [proviem.com.mx/modelos-de-protesis-de-brazo-y-mano/](http://proviem.com.mx/modelos-de-protesis-de-brazo-y-mano/). [Último acceso: 31 Oct 2024].
- [10] García Ferríz, "Prótesis Miembro Superior (brazo, mano)," 2024. [En línea]. Available: <https://www.ortopediagarciaferriz.com/es/product/6726212-protesis-mioelectricas-y-electronicas>. [Último acceso: 5 Noviembre 2024].
- [11] Cleveland Clinic, "Prosthetic Arm," 2024. [En línea]. Available: <https://my.clevelandclinic.org/health/treatments/prosthetic-arm>. [Último acceso: 5 Nov 2024].
- [12] J. J. Stokosa, "Opciones para las prótesis de los miembros," American Prosthetics Institute, Mar 2024. [En línea]. Available: <https://www.msmanuals.com/es/professional/temas-especiales/miembro-prot%C3%A9sico/opciones-para-las-pr%C3%B3tesis-de-los-miembros>. [Último acceso: 31 Oct 2024].
- [13] E. Zepeda, "¿Cómo funcionan las prótesis de brazo?," PRIMECARE, May 2022. [En línea]. Available: <https://primecareprosthetics.com/es/blog/what-you-should-know-before-getting-a-prosthetic-arm>. [Último acceso: 31 Oct 2024].
- [14] ABCFICHAS, "Anatomía y Músculos del brazo," 27 Jul 2021. [En línea]. Available: <https://www.abcfichas.com/musculos-del-brazo/>. [Último acceso: 5 Nov 2024].
- [15] Anatomia Topografica, "Flexor radial del carpo," 2018. [En línea]. Available: <https://anatomiatopografica.com/musculos/flexor-radial-del-carpo/>. [Último acceso: 5 Nov 2024].

- [16] Anatomia Topografica, "Flexor cubital del carpo," 2018. [En línea]. Available: <https://anatomiatopografica.com/musculos/flexor-cubital-del-carpo/>. [Último acceso: 5 Nov 2024].
- [17] Anatomia Topografica, "Músculo flexor común superficial de los dedos," 2019. [En línea]. Available: <https://anatomiatopografica.com/musculos/musculo-flexor-comun-superficial-de-los-dedos/>. [Último acceso: 5 Nov 2024].
- [18] T. A. Rodriguez, "Flexor Largo Profundo Común de los Dedos," 2017. [En línea]. Available: <https://musculoamusculo.wordpress.com/anatomia-muscular/miembro-superior/muneca/flexor-largo-profundo-comun-de-los-dedos-flpcd/>. [Último acceso: 5 Nov 2024].
- [19] M. Schünke, "Músculo extensor radial largo del carpo," de *Prometheus: Anatomía general y aparato locomotor*, Panamericana, 2015, p. 316.
- [20] FisioOnline, "Músculo radial corto o segundo radial," 2015. [En línea]. Available: <https://www.fisioterapia-online.com/glosario/musculo-radial-corto-o-segundo-radial>. [Último acceso: 5 Nov 2024].
- [21] FisioOnline, "Músculo extensor común de los dedos," 2015. [En línea]. Available: <https://www.fisioterapia-online.com/glosario/musculo-extensor-comun-de-los-dedos>. [Último acceso: 5 Nov 2024].
- [22] R. D. Pinzón Morales, D. A. Morales B. y V. H. Grisales P., "Caracterización de señales electromiográficas para la discriminación de seis movimientos de la mano," *Scientia et Technica*, vol. 2, p. 42, 2009.
- [23] J. L. Melo, "Ergonomía aplicada a las herramientas - 2da Parte," 16 Agosto 2002. [En línea]. Available: <https://estrucplan.com.ar/ergonomia-aplicada-a-las-herramientas-020-parte/>. [Último acceso: 28 Octubre 2024].

- [24] M. Guzmán, “Kenhub,” 30 Octubre 2023. [En línea]. Available: <https://www.kenhub.com/es/library/anatomia-es/falanges-de-la-mano>. [Último acceso: 28 Octubre 2024].
- [25] TecBolivia.com, “Sensor de Electromiografía EMG,” 2024. [En línea]. Available: <https://www.tecbolivia.com/index.php/venta-de-componentes-electronicos-11/sensores/sensor-de-electromiograf%C3%ADa-emg-detail#:~:text=El%20Sensor%20de%20Electromiograf%C3%ADa%20EMG,s%C3%B3lo%20para%20aprendizaje%20y%20experimentaci%C3%B3n..> [Último acceso: 28 Octubre 2024].
- [26] Á. Orozco Gutiérrez, G. Betancourt Orozco y E. Giraldo Suárez, “Determinación de movimientos a partir de señales electromiográficas utilizando máquinas de soporte vectorial,” *DialNet*, vol. 1, p. 15, 2005.
- [27] Mouser Electronics, “SENO240,” Mouser Electronics Inc., 2024. [En línea]. Available: <https://www.mouser.ec/ProductDetail/DFRobot/SENO240?qs=WoyvOOoixfEotK8Y%2FVCKrg%3D%3D>. [Último acceso: 31 Octubre 2024].
- [28] Unit Electronics, “Sensor EMG Analógico SEN0240,” 2024. [En línea]. Available: <https://uelectronics.com/producto/sensor-emg-analogico-sen0240/>. [Último acceso: 28 Octubre 2024].
- [29] ElectroStore, “Sensor Muscular *Myoware* EMG,” 2024. [En línea]. Available: <https://grupoelectrostorec.com/shop/sensores/senales-bioelectricas/sensor-muscular-myoware-emg/>. [Último acceso: 2024 Octubre 2024].
- [30] PlexyLab, “Módulo sensor muscular EMG AD8832,” 2024. [En línea]. Available: <https://www.plexylab.com/product-page/m%C3%B3dulo-sensor-muscular-emg-ad8832-muscle-sensor>. [Último acceso: 1 Noviembre 2024].

- [31] ElectroStore, “Sensor Muscular ECG EMG 3,3V AD8832,” 2024. [En línea]. Available: <https://grupoelectrostore.com/shop/sensores/senales-bioelectricas/sensor-muscular-ecg-emg-33v-ad8832-arduino-raspberry/>. [Último acceso: 1 Noviembre 2024].
- [32] Google Cloud, “¿Qué es la inteligencia artificial o IA?,” 2023. [En línea]. Available: <https://cloud.google.com/learn/what-is-artificial-intelligence?hl=es-419>. [Último acceso: 28 Octubre 2024].
- [33] MathWorks, “Redes Neuronales Convolucionales,” 2023. [En línea]. Available: <https://es.mathworks.com/discovery/convolutional-neural-network.html>. [Último acceso: 4 Noviembre 2024].
- [34] Datacamp, “Introducción a las redes neuronales convolucionales (CNN),” 15 Abril 2024. [En línea]. Available: <https://www.datacamp.com/es/tutorial/introduction-to-convolutional-neural-networks-cnns>. [Último acceso: 4 Noviembre 2024].
- [35] K. P. Avilés Mendoza y N. G. Gabior León, “Diseño e implementación de una prótesis robótica con señales EMG usando técnicas de inteligencia artificial,” 2021. [En línea]. Available: <http://www.dspace.espol.edu.ec/handle/123456789/54698>. [Último acceso: 31 Octubre 2024].
- [36] L. A. Zabala Aguiar, “Desarrollo de un sistema de adquisición y procesamiento de señales electromiográficas con Matlab para la identificación de los movimientos de extensión, flexión, pronación y supinación de un brazo humano,” Octubre 2017. [En línea]. Available: <http://dspace.espol.edu.ec/bitstream/123456789/7546/1/20T00917.PDF>. [Último acceso: 31 Octubre 2024].
- [37] R. Rodríguez Serrezuela, M. Á. Tovar Cardozo, J. J. Gómez Montiel, R. Sagaro Zamora y E. Marañón Reyes, “Análisis Comparativo entre MAE y RNA en señales de EMG obtenidas para control de una prótesis mano robótica,” de *III Congreso Internacional*

*en Inteligencia Ambiental, Ingeniería de Software y Salud Electrónica y Móvil*, Pereira, 2019.

- [38] C. Serrano, “Kenhub,” 30 Octubre 2023. [En línea]. Available: <https://www.kenhub.com/es/library/anatomia-es/articulacion-de-la-muneca-radiocarpiana>. [Último acceso: 28 Octubre 2024].
- [39] REA Salud, “Anatomía de la mano I: Segmentos, componentes óseos y articulaciones,” Youtube, Valdivia, 2017.
- [40] A. Gallego, “Ejercicios de mano y muñeca,” 2024. [En línea]. Available: <https://drgallegogoyanes.es/area-pacientes/ejercicios-de-mano-y-muneca/>. [Último acceso: 28 Octubre 2024].
- [41] EcuRed, “Pronación,” 19 Abril 2019. [En línea]. Available: <https://www.ecured.cu/Pronaci%C3%B3n>. [Último acceso: 28 Octubre 2024].
- [42] RPS, “La Historia de las Prótesis,” RPS Medical, 1 Feb 2024. [En línea]. Available: <https://www.rpsmedical.com/la-historia-de-las-protesis/>. [Último acceso: 30 Oct 2024].
- [43] “Prótesis De Miembro Superior,” Interbionic, 2017. [En línea]. Available: [interbionic.mx/protesis-miembro-superior/](http://interbionic.mx/protesis-miembro-superior/). [Último acceso: 31 Oct 2024].
- [44] Ortopedia Gordillo, “Prótesis amputación antebrazo estética pasiva,” SANICOR, 2020. [En línea]. Available: <https://www.ortopediagordillo.com/producto/protesis-amputacion-antebrazo-estetica-pasiva-encaje-laminado-supracondilar-o-similar-muneca-redonda-esferica-u-oval-mano-estetica-pasiva-y-guante-cosmetico/>. [Último acceso: 5 Nov 2024].

## **Apéndices**

## APÉNDICE A

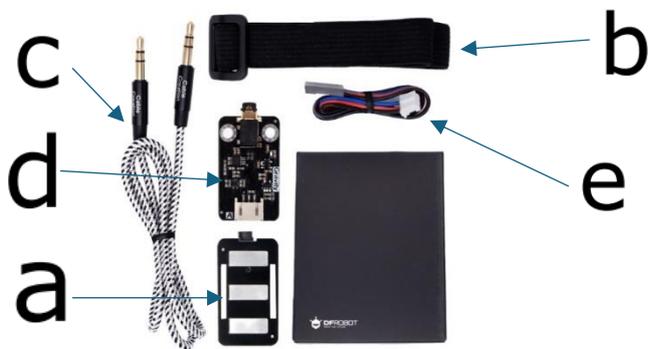
**Tabla A. 1.**

Especificaciones y componentes sensor SEN0240

<b>Especificación</b>	<b>Detalles</b>
Marca	DFRobot
Modelo	SEN0240
Incluye	a) Placa de electrodo seco EMG b) Cinturón de muñeca c) Cable de electrodo d) Placa transmisora de señal EMG e) Cable analógico de gravedad de 3 pines
Peso	35 g

**Figura A. 1.**

Componentes físicos del sensor SEN0240



**Figura A. 2.**

Transmisor de señal Sensor SEN0240

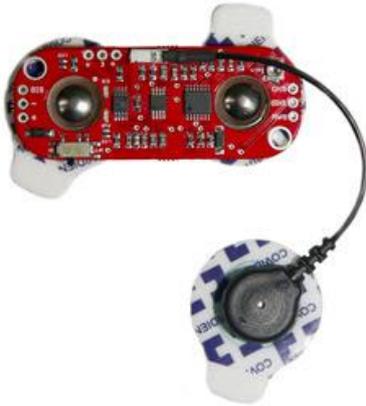


**Tabla A. 2.**

Transmisor de señal sensor SEN0240

<b>Transmisor de señal</b>	
Tensión de alimentación	+3.3V ~ 5.5V
Voltaje de funcionamiento	+3V
Rango de detección	+/- 1.5 mV
Conector de electrodo	PJ – 342
Conector del módulo	PH 2.0 – 3P
Voltaje de salida	0 ~ 3.0V
Temperatura de funcionamiento	0 ~ 50 °C
Longitud de cable PH 2.0 – 3P a Dupont Hembra	30 cm
Dimensiones	22 mm x 39.1 mm x 6.73 mm

**Figura A. 3.**  
Sensor EMG Myoware



**Tabla A. 3.**  
Especificaciones técnicas del sensor Myoware

<b>Característica</b>	<b>Especificación</b>
Dimensiones	2.1" × 0.8" (excluyendo el cable del electrodo de referencia de aproximadamente 3" de largo)
Voltaje de funcionamiento	2.9V a 5.7V (sin necesidad de suministro negativo)
Salida de tensión analógica	0V a +Vs (tensión de alimentación)
Ganancia	Ajustable
Salidas EMG	Envoltorio y EMG sin procesar disponibles
Conectores de electrodos	Electrodos encajan directamente en <i>Myoware</i> (también se pueden conectar cables de electrodos externos)
Indicadores LED	LED de encendido y LED que se ilumina al flexionar el músculo
Interruptor de alimentación	Sí
Protección de voltaje inverso	Sí
Orificios de montaje	Dos, adecuados para tornillos M3 o #4

**Figura A. 4.**  
Sensor EMG AD8832



**Figura A. 5.**  
Características y especificaciones técnicas del sensor AD 8832

Característica	Especificaciones
Voltaje de funcionamiento	3.3 V
Rango de temperatura nominal	0 ~ 70°C
Rango de temperatura de trabajo	-40 ~ 85°C
Salida	Analógica
Función adicional	Detección de Leads-Off
Conector	Jack de 3.5mm para almohadilla biomédica
Tamaño	10.00 x 6.00 x 3.00 cm
Componentes	<ul style="list-style-type: none"><li>• Placa de sensor muscular</li><li>• Cable EMG profesional</li><li>• 3 electrodos de superficie desechables</li></ul>

**Link de Github:** [https://github.com/miljosan/EMG\\_Prosthesis](https://github.com/miljosan/EMG_Prosthesis)

## APÉNDICE B

**Tabla B. 1 .** Lista de materiales y precios en proyecto (Parte Mecánica)

<b>Material</b>	<b>Cantidad</b>	<b>Precio c/u</b> <b>(USD)</b>	<b>Precio Total</b> <b>(USD)</b>
Rollo de filamento PLA - Color piel	1 Kg	\$22.00	\$22.00
Rollo de filamento TPU - Color blanco	1 Kg	\$28.00	\$28.00
Tornillo M3	8	\$0.40	\$3.20
Tornillo M2	8	\$0.30	\$2.40
Fomix 0.5 cm grosor	1	\$3.00	\$3.00
<b>Total</b>			<b>\$58.60</b>

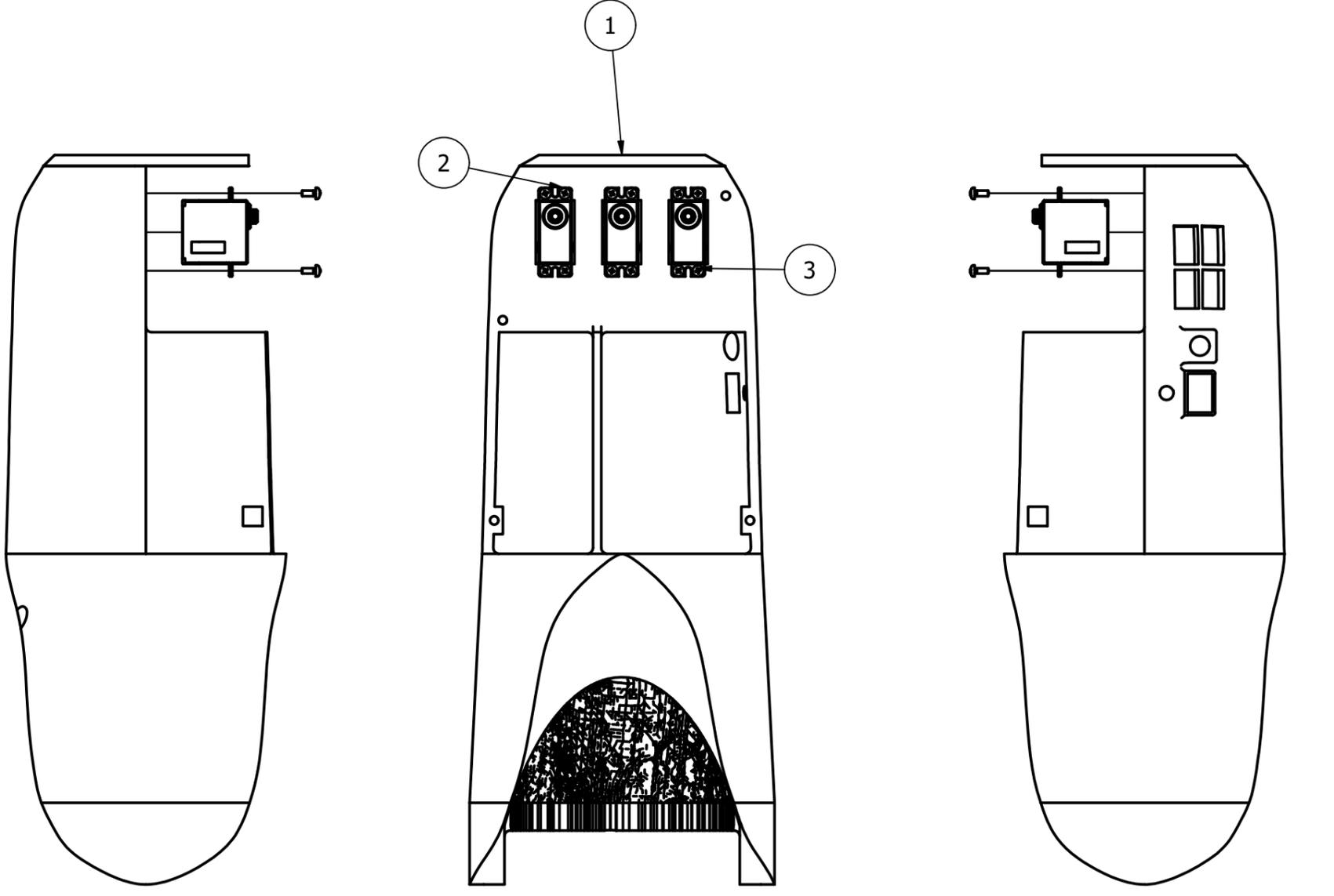
**Tabla B. 2 .** Lista de materiales y precios en proyecto (Accesorios)

<b>Material</b>	<b>Cantidad</b>	<b>Precio c/u</b> <b>(USD)</b>	<b>Precio Total</b> <b>(USD)</b>
Sellador Adimastic Pu-25 Blanco 300Ml	1	\$9.29	\$9.29
Pistola Metálica P/Silicon Liviana Azul Minter	1	\$2.89	\$2.89
Arnés médico para prótesis	1	\$150.00	\$150.00
Adaptador cargador 110AC/12DC 2A	1	\$6.49	\$6.49
Mecanizado de placa de baquelita en CNC	1	\$10.00	\$10.00
Elementos de soldadura	1	\$1.00	\$1.00
Cuerda de pesca de Nylon	1	\$1.87	\$1.87
Super pegamento brujita	1	\$1.50	\$1.50
<b>Total</b>			<b>\$183.04</b>

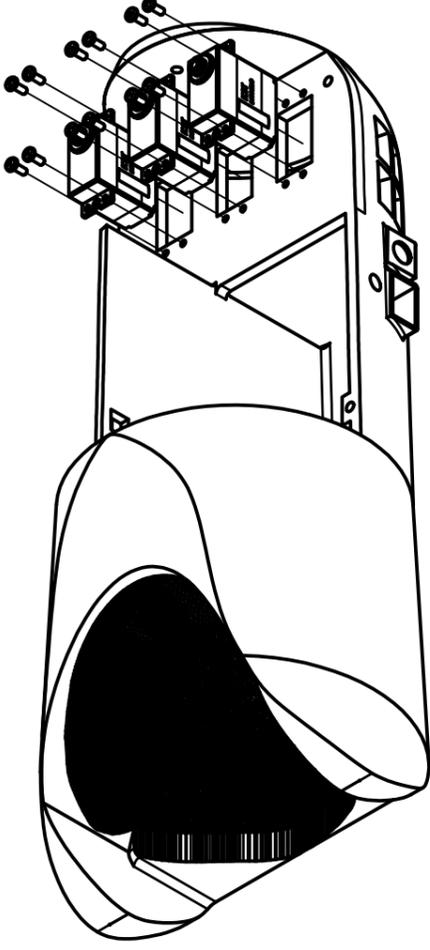
**Tabla B. 3 .** Lista de materiales y precios en proyecto (Parte Electrónica)

<b>Material</b>	<b>Cantidad</b>	<b>Precio c/u (USD)</b>	<b>Precio Total (USD)</b>
Sensor de fuerza MF01	1	\$15.00	\$15.00
Sensor EMG analógico Gravity by OYMOTION + envío	1	\$60.00	\$60.00
Cable de bobina esmaltada	1	\$0.20	\$0.20
Cable de cobre	3	\$0.40	\$1.20
Batería Ion-Litio 18650 3.7V 12000mAh	3	\$1.50	\$4.50
Portapila Ion-Litio 18650	3	\$0.50	\$1.50
Módulo de carga y protección para batería Ion-Litio 3C	1	\$3.75	\$3.75
Step-down LM2596	1	\$4.00	\$4.00
Switches NO	5	\$0.40	\$2.00
Diodo LED verde	1	\$0.10	\$0.10
Servomotor MG995 DIGI HI-SPEED	2	\$9.00	\$18.00
ESP32	1	\$9.50	\$9.50
Conector DC 9mm Macho	1	\$0.75	\$0.75
Conector DC 9mm Hembra	1	\$0.75	\$0.75
Resistencias 1kOhm 0.5 watts	5	\$0.10	\$0.50
Resistencias 10kOhm 0.5 watts	1	\$0.10	\$0.10
Bornera 2 puertos	5	\$0.25	\$1.25
Bornera 3 puertos	6	\$0.35	\$2.10
Placa de baquelita	1	\$1.00	\$1.00
Termoencogible	1	\$0.20	\$0.20
<b>Total</b>			<b>\$126.40</b>

## **APÉNDICE C**

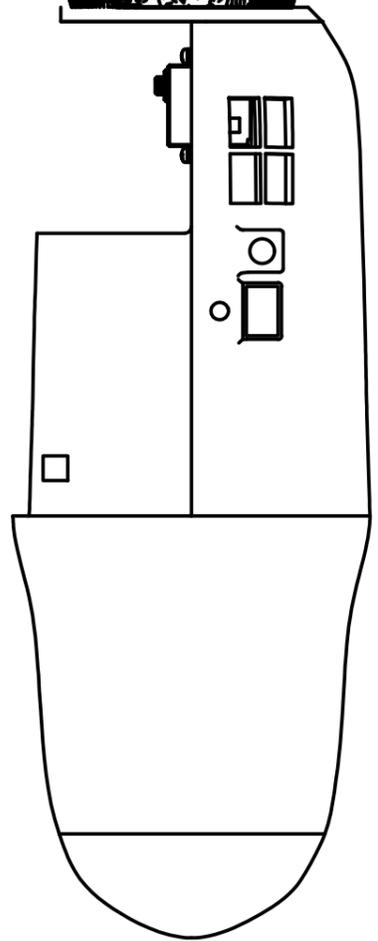
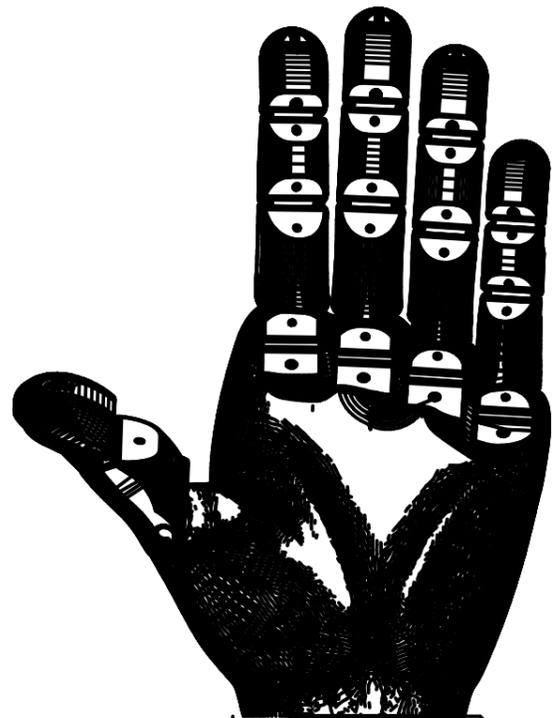
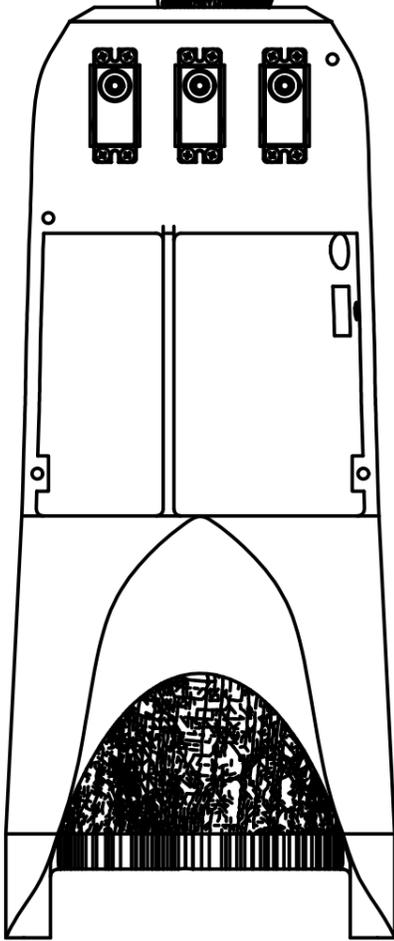
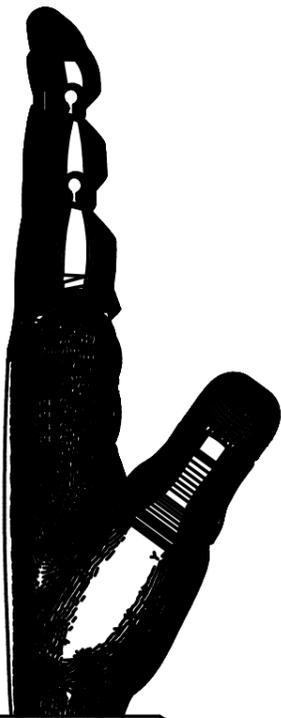
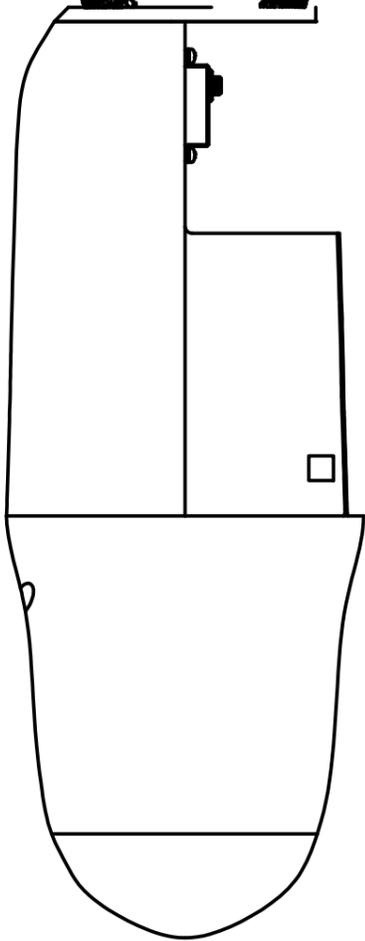


PARTS LIST		
ITEM	QTY	PART NUMBER
1	1	Antebrazo
2	3	Servo DS315MG
3	12	EN ISO 7045 - M2 x 5 - 4.8 - H

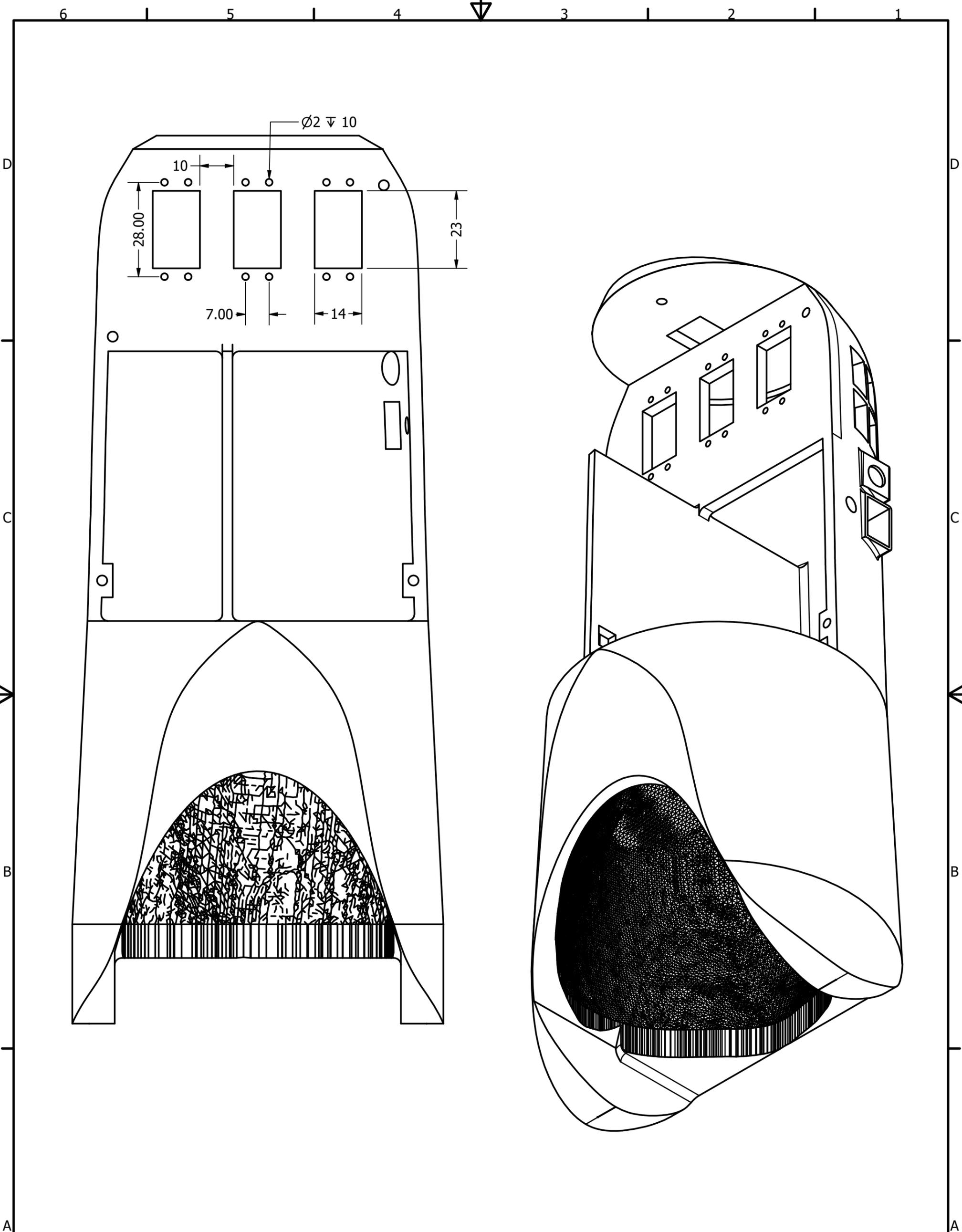


Nombre		Fecha	
Milton Sanchez		31/01/2025	
Miguel Jara			
Profesor		Bryan Puruncajas, Ph.D.	
Escala		Dibujo	
1 / 2		Propuesta de Servomotores	
Tamaño		Facultad de Ingenieria Mecanica y Ciencias de la Produccion	
A3		1	
		Unidad	
		mm	

**espol** Escuela Superior Politécnica del Litoral

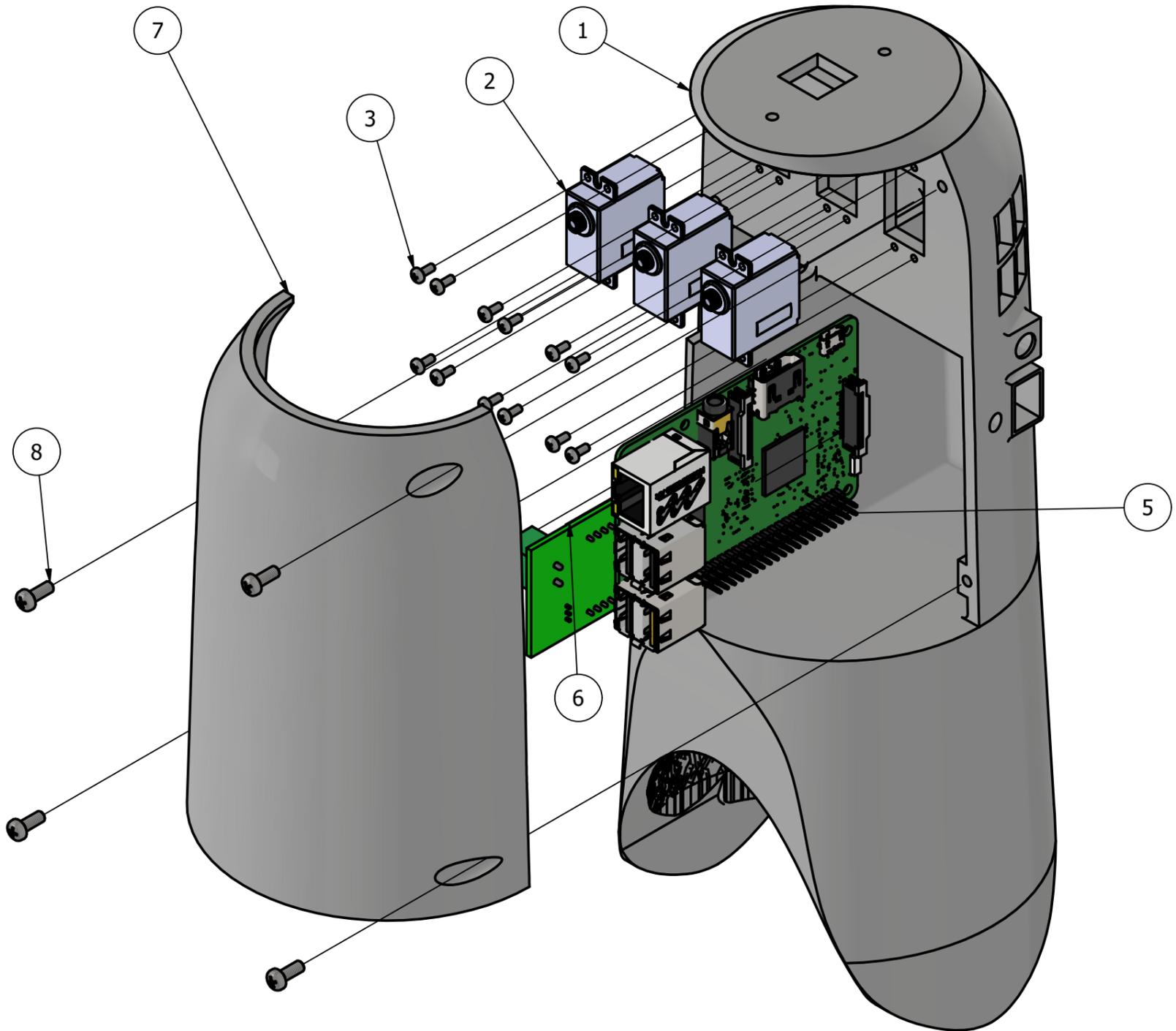


	Nombre	Fecha		
Dibujado	Milton Sanchez Miguel Jara	31/01/2025		
Profesor	Bryan Puruncajas, Ph.D.			
Escala	Dibujo	Propuesta de Servomotores	Tamaño	Facultad de Ingeniería Mecánica y Ciencias de la Producción
1 / 2			A3	Unidad
				mm



Dibujado	Nombre Milton Sanchez Miguel Jara	Fecha 31/01/2025		
Profesor	Bryan Puruncajas, Ph.D.			
Escala 1 : 1	Dibujo Propuesta de Servomotores	Tamaño A3	Facultad de Ingeniería Mecánica y Ciencias de la Producción	Unidad mm

PARTS LIST		
ITEM	QTY	PART NUMBER
1	1	Antebrazo
2	3	Servo DS315MG
3	12	EN ISO 7045 - M2 x 5 - 4.8 - H
4	1	Mano
5	1	Raspberry Pi 4B
6	1	PCB
7	1	Tapa Antebrazo
8	4	EN ISO 7045 - M3 x 8 - 4.8 - H



	Nombre	Fecha	 <b>Escola Superior Politécnica del Litoral</b>	
Dibujado	Milton Sanchez Miguel Jara	31/01/2025		
Profesor	Bryan Puruncajas, Ph.D.			
Escala	Dibujo	Propuesta de Servomotores	Tamaño	Facultad de Ingeniería Mecánica y Ciencias de la Producción
1 / 1.30			A3	Unidad
				mm