

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

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

Diseño de estimulador con inteligencia artificial para el congelamiento de
la marcha en la enfermedad de Parkinson

PROYECTO INTEGRADOR

Previo la obtención del Título de:

Ingenieros en Mecatrónica

Presentado por:

Ronald Adrián Cedeño Barahona

José Gabriel Párraga Mendoza

GUAYAQUIL - ECUADOR

Año: 2022

DEDICATORIA

El presente trabajo lo dedico a Dios, por brindarme fuerzas y bendiciones que me han ayudado a seguir adelante a lo largo de mi vida.

A mi madre Janeth Barahona, por ser mi pilar fundamental, mi mayor felicidad y mi motivación para siempre esforzarme en todo trabajo.

A mi padre Carlos Cedeño, por ser mi guía, mi cuidador y por entregarme consejos que me permiten crecer como buena persona.

A mi hermano Andrés Cedeño, por ser mi acompañante en los momentos más alegres y siempre poder contar con él.

A mi abuelita, por darme su bendición y amor, quien con un abrazo me ha llegado a brindar mucha tranquilidad.

Ronald Adrián Cedeño Barahona

DEDICATORIA

Le dedico el presente proyecto a mi familia, sin ella no podría haber llegado a ser lo que soy y tener las aspiraciones que poseo.

A mi madre Laura Mendoza, aunque haya dejado de estar presente en mi vida desde una edad temprana, sigue estando dentro en mi corazón con mucho amor y felicidad.

A mi padre José Wilfrido Párraga, gracias a su cuidado, enseñanzas y sabiduría que me ha impartido en la formación personal y académicamente puedo ser una excelente persona que busca el bien de los demás.

A mi hermano mayor José Andrés Párraga, por siempre buscar mi bienestar y apoyarme en los momentos difíciles de la vida demostrando su gran desempeño como hermano.

A mi hermano menor José Luis Párraga, por convertirse en mi compañero de confianza y entretenimiento en los momentos de felicidad.

José Gabriel Párraga Mendoza

AGRADECIMIENTOS

Agradezco a Dios por sus bendiciones y permitirme alcanzar mis objetivos personales y profesionales.

A mis padres y mi hermano, por confiar en mí, estar a mi lado en los momentos más difíciles y darme su apoyo a lo largo de mi crecimiento académico.

A cada uno de los docentes de la ESPOL quienes han sido parte esencial durante mi formación.

Al Ph.D. Carlos Saldarriaga y al Ph.D. Francis Loayza, por sus consejos, compromiso y colaboración en la elaboración del presente trabajo.

Al M.Sc. Efraín Terán, quien como coordinador nos ha dado su guía durante toda la carrera.

A mis amigos, quienes han sido parte de mi vida universitaria, por su acompañamiento y gran amistad, logrando que cada momento sea memorable junto a ellos.

Ronald Adrián Cedeño Barahona

AGRADECIMIENTOS

Agradezco a mi padres y hermanos por siempre estar a mi lado y brindarme el apoyo necesario para seguir adelante en el correcto camino de la vida.

A todos los docentes de la ESPOLE que me han compartido de sus conocimientos para encaminar mi formación académica.

Al Ph.D. Carlos Saldarriaga y al Ph.D. Francis Loayza, por la compañía académica y ayuda aportada durante la elaboración del presente proyecto.

Al M.Sc. Efraín Terán, aportando con sus consejos y conocimientos como coordinador de la carrera.

A mis amigos del colegio César y Diego, que desde pequeños siempre me han apoyado y compartido momentos de felicidad llegando a considerarlos hermanos.

A mis amigos de la universidad, los cuales considero como mi segunda familia, en especial Juan, Miguel, Ronald y Sofía que me han brindado de su compañía indispensable para crear buenos recuerdos.

José Gabriel Párraga Mendoza

DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; *Ronald Adrián Cedeño Barahona* y *José Gabriel Párraga Mendoza* damos nuestro consentimiento para que la ESPOI realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”



Ronald Cedeño Barahona



José Párraga Mendoza

EVALUADORES

Carlos Saldarriaga, PhD.

PROFESOR DE LA MATERIA

Francis Loayza, PhD.

PROFESOR TUTOR

RESUMEN

La enfermedad de Parkinson (EP) es un trastorno neurodegenerativo global que incluye el congelamiento de la marcha (CDM) caracterizado por una interrupción en el movimiento durante la caminata. En el Ecuador, esta condición es más común e impacta significativamente en la calidad de vida. Este proyecto se basa en el diseño de un sistema de adquisición de datos (DAQ) y un sistema de estimulación eléctrica aplicando algoritmos de inteligencia artificial (IA). Para ello, se diseñó un DAQ que permitió registrar la actividad de los pacientes durante pruebas de caminata. Estos datos se recolectaron por medio de sensores inerciales ubicados en las extremidades inferiores y se etiquetaron mediante una aplicación móvil. Luego, se utilizó un modelo de IA basado en un perceptrón multicapa (MLP) para el análisis e identificación de los patrones del CDM. Este modelo se implementó en un microcontrolador Arduino Nano 33 IoT en unión con un sistema de estimulación eléctrica que generó pulsos a los pacientes en las extremidades inferiores cuando se detecta el CDM. Se reclutó a 20 sujetos sanos que realizaron el mismo recorrido imitando el CDM, midiendo las detenciones voluntarias e involuntarias con una frecuencia de muestreo de 23 Hz, el algoritmo de IA obtuvo un 89,72% de precisión en el entrenamiento, y un voltaje de estimulación eléctrica de hasta 50V con regulación de baja frecuencia. Con este proyecto, se espera mejorar la calidad de vida de las personas con EP que sufren de CDM en Ecuador y en otros países del mundo.

Palabras Clave: Enfermedad de Parkinson, congelamiento de la marcha, sensores inerciales, perceptrón multicapa, pulsos eléctricos.

ABSTRACT

Parkinson's disease (PD) is a global neurodegenerative disorder that includes freezing of gait (FOG) characterized by an interruption in movement during walking. In Ecuador, this condition is more common and has a significant impact on quality of life. This project is based on the design of a data acquisition system (DAQ) and an electrical stimulation system applying artificial intelligence (AI) algorithms. For this, a DAQ was designed that allowed recording of the activity of the patients during walk tests. These data were collected using inertial sensors located in the lower extremities and labeled using a mobile application. Then, an AI model based on a multilayer perceptron (MLP) was used for the analysis and identification of the CDM patterns. This model was implemented in an Arduino Nano 33 IoT microcontroller in conjunction with an electrical stimulation system that delivered pulses to patients in the lower extremities when CDM is detected. 20 healthy subjects were recruited who performed the same walk imitating the CDM, measuring voluntary and involuntary stops with a sampling frequency of 23 Hz. The AI algorithm achieved 89.72% training accuracy and a voltage of electrical stimulation up to 50V with low frequency regulation. This project is expected to improve the quality of life of people with PD who suffer from CDM in Ecuador and other countries of the world.

Keywords: *Parkinson's disease, freezing of gait, inertial sensors, multilayer perceptron, electrical pulses.*

ÍNDICE GENERAL

RESUMEN.....	I
<i>ABSTRACT</i>	II
ÍNDICE GENERAL.....	III
ABREVIATURAS	VI
SIMBOLOGÍA	VII
ÍNDICE DE FIGURAS.....	VIII
ÍNDICE DE TABLAS	X
CAPÍTULO 1	
1. Introducción	1
1.1 Descripción del problema	2
1.2 Justificación del problema.....	3
1.3 Objetivos.....	4
1.3.1 Objetivo general	4
1.3.2 Objetivos específicos	4
1.4 Marco teórico	5
1.4.1 Tratamiento de pacientes con la enfermedad de Parkinson	7
1.4.2 Hardware implementado en proyectos de diseño de estimuladores	8
1.4.3 Dispositivos comerciales para el tratamiento del CDM	9
1.4.4 Estado del arte	10
CAPÍTULO 2	
2. Metodología	13
2.1 Requerimientos de diseño	13
2.2 Selección de alternativa de solución.....	14
2.3 Diseño detallado	17
2.4 Proceso del diseño	18

2.5	Diseño de sistema de adquisición de datos.....	18
2.5.1	Selección de componentes	19
2.5.2	Conversión de lecturas del sensor	21
2.5.3	Etiquetado de datos	21
2.5.4	Diagrama de flujo para experimentación.....	23
2.6	Diseño electrónico del sistema de estimulación eléctrica	23
2.6.1	Bloques del circuito de estimulación	24
2.7	Diseño de espacio de pruebas	27
2.7.1	Dimensionamiento de zona de pruebas.....	27
2.7.2	Pruebas experimentales en sujetos sanos.....	28
2.8	Diseño de algoritmo de inteligencia artificial.....	28
2.8.1	Preprocesamiento de datos	29
2.8.2	Creación y despliegue del modelo de inteligencia artificial	30
CAPÍTULO 3		
3.	Resultados y análisis	32
3.1	Hardware resultante	32
3.1.1	Sistema de adquisición de datos.....	32
3.1.2	Sistema de estimulación eléctrica	34
3.1.3	Elaboración de placa PCB	35
3.1.4	Contenedor del dispositivo	36
3.2	Resultados de los experimentos del DAQ	36
3.3	Resultados del algoritmo de inteligencia artificial	37
3.3.1	Creación del modelo y predicción de muestras.....	37
3.3.2	Comparación de predicciones y valores reales	39
3.3.3	Despliegue de algoritmo en placa de control	40
3.4	Resultados de las estimulaciones en sujetos de prueba	41

3.5	Características técnicas del dispositivo	42
3.6	Visualización de uso del dispositivo.....	42
3.7	Análisis de costos	43
CAPÍTULO 4		
4.	Conclusiones y recomendaciones.....	46
4.1	Conclusiones	46
4.2	Recomendaciones	47
BIBLIOGRAFÍA		
APÉNDICES		

ABREVIATURAS

ESPOL	Escuela Superior Politécnica del Litoral
IA	Inteligencia Artificial
ML	Machine Learning
EP	Enfermedad de Parkinson
CDM	Congelamiento de la marcha
DBS	Deep Brain Stimulation
USB	Universal Serial Bus
UART	Universal Asynchronous Receiver-Transmitter
I2C	Inter Integrated Circuits
SPI	Serial Peripheral Interface
APK	Android Package Kit
VDC	Voltaje de Corriente Directa
MLP	MultiLayer Perceptron
CI	Circuito Integrado
DOF	Degrees Of Freedom
AMPS	Automated Mechanical Peripheral Stimulation
IOT	Internet of Things
DAQ	Data Acquisition
PCB	Printed Circuit Board
AIFES	Artificial Intelligence for Embedded Systems
LSTM	Long Short-Term Memory
VPN	Valor Presente Neto
TIR	Tasa Interna de Retorno
TIO	Tasa de Interés de Oportunidad
DIN	Deutsches Institut für Normung
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization

SIMBOLOGÍA

V	Voltios
mA	Miliamperios
kB	Kilobytes
μ A	Microamperios
bps	Bytes por segundo
Mbps	Megabits por segundo
GHz	Gigahertz
m	Metro
s	Segundo
g	Gravedad
$\frac{m}{s^2}$	Metro sobre segundo al cuadrado
$\frac{deg}{s}$	Grado sobre segundo
μ s	Microsegundo
μ F	Microfaradio
k Ω	Kiloohmio
mH	Milihenrio
mm	Milímetro
ms	Milisegundo
Hz	Hertz

ÍNDICE DE FIGURAS

Figura 1.1 Cantidad de pacientes con EP por intervalos de edad	5
Figura 1.2 Probabilidad de supervivencia en pacientes sanos y con CDM.....	6
Figura 1.3 Placa de microcontrolador Arduino Uno R3.....	8
Figura 1.4 Módulo LSM9DS1 9-DOF	9
Figura 1.5 Dispositivo de movilidad NexStride.....	9
Figura 1.6 Dispositivo de tratamiento médico GONDOLA	10
Figura 1.7 Estimulación vibratoria para reducir episodios de CDM.....	11
Figura 1.8 Dispositivo de estimulación óptica para combatir el CDM	11
Figura 1.9 Modelo de estimulación auditiva.....	12
Figura 2.1 Diagrama de diseño de la alternativa 1.....	14
Figura 2.2 Diagrama de diseño de la alternativa 2.....	15
Figura 2.3 Diagrama de diseño de la alternativa 3.....	15
Figura 2.4 Diagrama de diseño conceptual	17
Figura 2.5 Diagrama de proceso del diseño	18
Figura 2.6 Conexiones en el sistema DAQ	20
Figura 2.7 Ventanas de diseño de la aplicación móvil	22
Figura 2.8 Diagrama de flujo del funcionamiento.....	23
Figura 2.9 Pulsos de corriente continua monofásica	23
Figura 2.10 Circuito multivibrador monoestable.....	24
Figura 2.11 Circuito generador de pulsos cuadrados	25
Figura 2.12 Circuito elevador de voltaje.....	26
Figura 2.13 Circuito de estimulación eléctrica	26
Figura 2.14 Vista superior de la zona de pruebas.....	27
Figura 2.15 Graficas de las variables para el pie izquierdo	29
Figura 2.16 Graficas de las variables para el pie derecho	29
Figura 2.17 Modelo de red neural MLP.....	30
Figura 2.18 Ventana de datos para el entrenamiento.....	31
Figura 3.1 Sistema completo de adquisición de datos.....	32
Figura 3.2 Posición de los ejes utilizados para la toma de datos.....	33
Figura 3.3 Etiquetado de datos en tiempo real	33
Figura 3.4 Señalética de estados con diodos leds.....	34

Figura 3.5 Implementación de sistema de estimulación eléctrica	34
Figura 3.6 Simulación de pulsos cuadrados en osciloscopio y elevación	35
Figura 3.7 Circuito de estimulación eléctrica montada en placa PCB.....	35
Figura 3.8 Contenedor del dispositivo diseñado en 3D.....	36
Figura 3.9 Variaciones en los datos obtenidos por los sujetos de pruebas	36
Figura 3.10 Señalización de los estados en las pruebas experimentales.....	37
Figura 3.11 Visualización de la estructura general del modelo.....	38
Figura 3.12 Comparación de etiquetas reales y predicción	39
Figura 3.13 Observación de variables “Etiqueta” y “Predicción”	39
Figura 3.14 Predicción de nuevos datos recolectados.....	40
Figura 3.15 Mejor localización con sensibilidad eléctrica - Arco del pie.....	41
Figura 3.16 Visualización de uso para el dispositivo y sus componentes.....	43

ÍNDICE DE TABLAS

Tabla 2.1 Requerimientos de diseño del sistema	13
Tabla 2.2 Descripción de los criterios de selección	16
Tabla 2.3 Pesos y ponderaciones de los criterios de selección.....	16
Tabla 2.4 Matriz de decisión	17
Tabla 2.5 Especificaciones técnicas de placa de control	19
Tabla 2.6 Especificaciones técnicas del sensor inercial	19
Tabla 2.7 Especificaciones técnicas del módulo bluetooth	19
Tabla 2.8 Especificaciones técnicas de módulo de almacenamiento	20
Tabla 2.9 Descripción de botoneras en aplicación móvil	22
Tabla 3.1 Costos de componentes para la elaboración del sistema.....	43
Tabla 3.2 Costos totales finales	44

CAPÍTULO 1

1. INTRODUCCIÓN

La Enfermedad de Parkinson (EP) fue establecida como una condición médica reconocida en 1817 por el Doctor James Parkinson denominada “Parálisis Agitante” [1]. En los últimos 25 años la enfermedad ha tenido un crecimiento demasiado acelerado en comparación a otros tipos de trastornos neurológicos, según la Organización Mundial de la Salud [2], involucrando múltiples discapacidades motoras que generan: temblores, coordinación de movimientos, rigidez en diferentes extremidades, entre otros síntomas.

Esta enfermedad es de carácter crónico y degenerativo provocado por las células de la sustancia negra pars compacta, ubicada en la parte profunda del cerebro y encargadas de producir dopamina, dando como consecuencia una extensa variedad de dificultades motoras que perjudica al sistema nervioso central, provocando movimientos y contracciones involuntarias. Cada paciente vive con diferentes síntomas de la enfermedad a lo largo de su vida, y no existen estudios oficiales que permitan predecir cómo evolucionará su enfermedad a futuro.

Mientras la enfermedad sigue evolucionando en un paciente por algunos años podría traer consigo un fenómeno de trastorno episódico conocido como congelamiento de la marcha (CDM) o “*Freezing of Gait*” (FOG por sus siglas en inglés). Indica la incapacidad de moverse dentro de su estado natural de caminata, creando una sensación de pies “pegados” al suelo, provocando frustraciones e incluso accidentes como caídas al realizarse una pausa repentina. La presencia de este bloqueo motor en el paciente puede llegar a depender de diferentes factores en su entorno, como la ejecución de giros o la presencia de obstáculos, por lo que la comprensión de estas causas aún sigue en estudios [3]. Basado en lo anterior, el presente proyecto busca diseñar un sistema de estimulación en extremidades inferiores de forma automática, usando sensores basados en módulos electromecánicos, sistema de adquisición de datos, algoritmos de inteligencia artificial (IA) y estimuladores con pulsaciones eléctricas hacia el sujeto cuándo el sistema detecte algún parámetro de pausa involuntaria.

1.1 Descripción del problema

La evolución de la enfermedad de Parkinson produce síntomas cada vez más incómodos al paciente como es el caso del congelamiento de la marcha, lo que produce que el paciente se detenga de forma involuntaria ante cualquier cambio en su entorno, siendo más común en áreas con la presencia de obstáculos o cambios de dirección de la marcha. Los factores perjudiciales al ocurrir estas pausas pueden ser físicas como fuertes impulsos repentinos hacia adelante hasta caídas con daños severos que pueden llegar a producir el fallecimiento del paciente, de igual forma problemas psicológicos como frustraciones, incomodidades y vergüenza en su vida diaria [4].

El objetivo del proyecto consiste en brindar al individuo un dispositivo de ayuda para la estimulación automática, reduciendo los episodios de congelamiento de la marcha, a través de un método no invasivo, es decir que no se utilizan instrumentos que penetren alguna parte del cuerpo. Para ello se requiere el monitoreo constante de datos del individuo mientras el dispositivo se encuentre encendido; midiendo variables como desplazamiento, velocidad, aceleración y orientación, lo que pretende identificar los episodios de CDM (detenciones involuntarias) de forma automática usando técnicas de Inteligencia Artificial (IA). Además, se requiere que el sistema sea compacto debido a su ubicación en las extremidades inferiores, que no genere incomodidad en el usuario. No obstante, la limitación en la disponibilidad horaria que tendrían los pacientes es escasa, por lo tanto, se requerirá utilizar sujetos sanos que simulen los movimientos que tendría un paciente de Parkinson con congelamiento de marcha para obtener resultados similares a los esperados en el sistema.

1.2 Justificación del problema

Hasta el momento se ha realizado una extensa investigación por parte de diferentes organizaciones con el fin de determinar las consecuencias y predicción de diagnósticos sobre la EP, de igual magnitud, múltiples propuestas aparecen con el pasar de los años para reducir diversos síntomas que puede generar la enfermedad. Para los episodios de CDM no se cuenta con un producto a nivel nacional, por otro lado, a nivel internacional una de muchas empresas se encarga de proporcionar un dispositivo con señales visuales y auditivas que permita liberar al paciente del estado de congelamiento, restableciendo la comunicación entre el cerebro y el cuerpo [5].

Sin embargo, la variedad de dispositivos que brindan señales auditivas o visuales las realizan constantemente, siendo no posible determinar los momentos en el que el paciente sufra la pausa motora, involucrando un constante uso del aparato y gasto energético. Por estas razones, otros estudios efectúan la recolección de datos previos y aprendizaje del comportamiento basados en el congelamiento involuntario por diferentes pacientes, y proporcionar una vía alterna de comunicación a través de señales de estimulación eléctrica sensorial teniendo resultados eficientes [6].

Por consiguiente, el presente proyecto propone reducir los episodios de congelamiento, brindando al sujeto un dispositivo estimulador ubicado en sus extremidades inferiores que es activado automáticamente al detectar valores anormales que sean considerados pausas involuntarias, usando un algoritmo de IA entrenado por los diversos datos recolectados en sujetos sanos, logrando así la activación de estímulos sensoriales y reinicio de la marcha.

1.3 Objetivos

1.3.1 Objetivo general

Diseñar un sistema de estimulación automático mediante la recolección de datos e implementación de modelos de inteligencia artificial como terapia del congelamiento de la marcha para pacientes con enfermedad de Parkinson.

1.3.2 Objetivos específicos

- Diseñar un sistema de adquisición de datos para el registro de aceleración en los movimientos de los pies en los individuos.
- Implementar en la tarjeta de adquisición de datos un sistema de estimulación eléctrica o vibratoria para los pies del usuario.
- Realizar experimentos en sujetos sanos para el registro de datos y ubicación de la estimulación.
- Implementar un algoritmo de inteligencia artificial para detectar el congelamiento de la marcha automáticamente.
- Seleccionar el modelo de aprendizaje óptimo para embeberlo en la tarjeta de adquisición de datos permitiendo generar los estímulos basado en las señales de entrada tomadas en tiempo real.

1.4 Marco teórico

La enfermedad de Parkinson presente en la población del Ecuador fue contabilizada por medio de un estudio realizado en la provincia de Manabí, en el año 2013, por la Universidad Técnica de Manabí y el Centro Médico de Neurociencias en Guayaquil, dando como resultado que cada 285 personas de 116,983 habitantes a partir de los 40 años padezcan de EP, llegando a considerarse como el 0,24% de la población presente en Manabí [7]. Además, se tiene presente la cantidad de personas por intervalo de diez años con respecto a esta enfermedad, lo cual se observa en la Figura 1.1 [7].

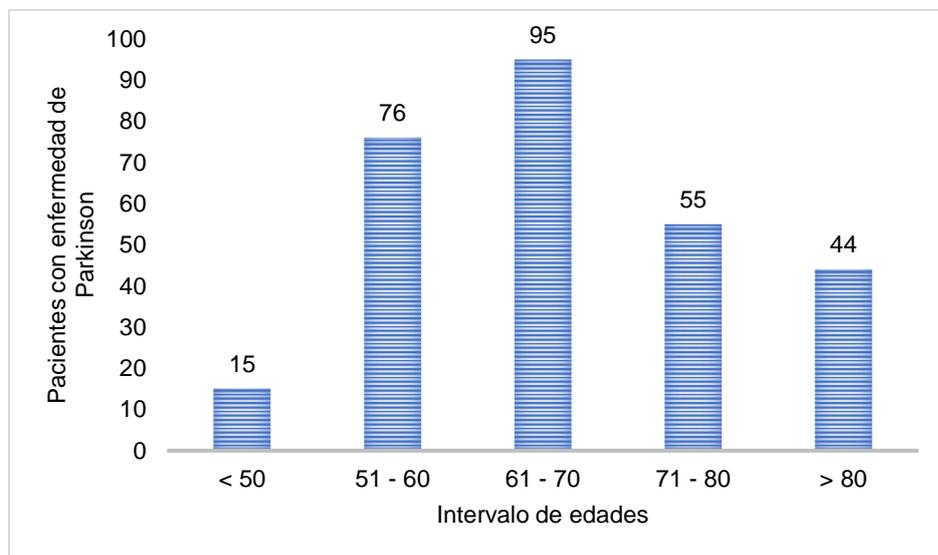


Figura 1.1 Cantidad de pacientes con EP por intervalos de edad [7]

Estudios globales recientes demostraron que un 26% de pacientes con EP en etapas tempranas y que no fueron expuestos a tratamientos de levodopa (L-dopa), sufren de CDM [8]. Sin embargo, el uso de L-dopa no asegura en su totalidad que los pacientes eviten por completo sufrir de CDM en algún transcurso de su vida. Las personas con EP de etapa avanzada y gran exposición a tratamientos de L-dopa, tiene un rango de porcentaje de prevalencia de CDM es de 20 a 60% [9]. Varias investigaciones con respecto a la enfermedad de Parkinson sugieren que la probabilidad de que un paciente sufra de CDM depende de aspectos patológicos que pueden no estar asociados a todos los pacientes, de forma que el 80% nunca estuvo expuesto al CDM en ninguna de sus etapas de la enfermedad [10].

Se considera a las fracturas de caderas por caídas como la consecuencia más probable de ocurrir cuando se tiene CDM, además, según un estudio realizado en el Condado de Olmsted en Minnesota, determinaron que el 25% de pacientes con CDM sufren de una fractura en las caderas dentro de los primeros 10 años desde que se diagnostica esta enfermedad [11]. Los pacientes con problemas de marcha y balance son más propensos a tener un alto riesgo de mortalidad, el promedio de supervivencia que tienen desde que las caídas comienzan a ser recurrentes es equivalente a 7 años, llegando a fallecer por varias causas directas, entre estas se pueda encontrar el hematoma epidural masivo [12].

Según un estudio realizado por David A. Bennett, et al. presentado en la revista de medicina en Nueva Inglaterra, se pudo graficar la probabilidad de supervivencia en contraste con los años de vida futuros para los pacientes que poseen CDM y los que no padecen de este tipo de desorden de movimiento en la EP [12], tal como se observa en la Figura 1.2.

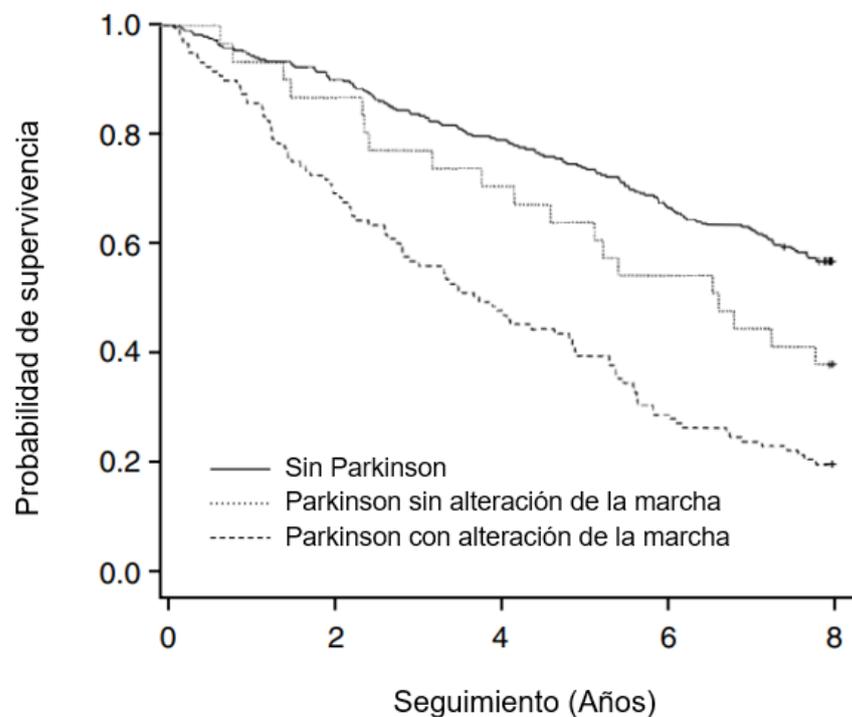


Figura 1.2 Probabilidad de supervivencia en pacientes sanos y con CDM [12]

1.4.1 Tratamiento de pacientes con la enfermedad de Parkinson

Los tipos de tratamientos encargados de minimizar y controlar el congelamiento de la marcha pueden ser categorizados en dos grupos: tratamientos farmacológicos y operaciones internas/invasivas que ayudarán luego en la rehabilitación motora.

1.4.1.1 Tratamientos farmacológicos

Según un estudio realizado en la organización Translational Neurodegeneration en el año 2020, existen varios tratamientos farmacológicos que ayudan a combatir el CDM, en donde se pueden destacar los siguientes tipos de fármacos: Levodopa, Gel intestinal de levodopa-carbidopa (LCIG), Dopa mine agonist, Selegiline, Rasagiline, Methylphenidate (MPH), Istradefylline, Antidepresivos, L-DOPS - droxidopa, Amantadine, Atomoxetine, Inhibidor de la acetilcolinesterasa, Toxina botulínica [13]. Los resultados obtenidos en los pacientes demostraron que estos fármacos si presentan una mejora en los episodios de CDM. Sin embargo, cuando la dosis es regular o menor a la recetada, los resultados disminuyen llegando a ser casi nulos; además, se necesita utilizarlos por un gran periodo de tiempo [13], lo cual genera un gasto económico considerable para los pacientes que necesitan de estos fármacos y una dependencia de ellos para combatir al CDM. Cabe recalcar que para algunos tratamientos farmacológicos no se tienen buenos resultados en todos los pacientes, de forma que algunos no presentarían mejoras.

1.4.1.2 Cirugías para el tratamiento de la EP

Análisis recientes demostraron que las intervenciones invasivas dentro del cerebro, a partir de estimulaciones en los núcleos pálido bilateral y subtalámico bilateral, conocido como estimulación cerebral profunda (DBS por sus siglas en inglés), dieron como resultado el decrecimiento en los efectos que tienen el CDM presente en los pacientes con EP durante su etapa "off" (transcurso de tiempo donde las dosis de tratamientos dejan de funcionar) [14]. Sin embargo, estos tipos

de intervenciones cerebrales no surten efecto cuando el paciente está en su etapa “on” (momento en el que la dosis de tratamiento está funcionando) [15].

Se debe considerar los riesgos que generaría realizar algunas de estas operaciones internas en el cerebro, las cuales son diferentes para cada tipo de intervención llegando a convertirse en problemas mayores, entre los siguientes se tiene: intervención subtalámica bilateral, ocasionando daños considerables en la postura del paciente; intervención pálida bilateral, incrementando el riesgo de CDM y empeorar el balance [16].

1.4.2 Hardware implementado en proyectos de diseño de estimuladores

Al pasar de los años se han diseñado diferentes tipos de estimuladores eléctricos, utilizados para casos de rehabilitación, terapia física, programas de entrenamiento, entre otros. Los modelos se han realizado en mayor parte con la implementación de Arduino como principal unidad de control, dado que conlleva una estructura más sencilla. Siendo la placa más usada el Arduino UNO basada en el microcontrolador ATmega328P, que cuenta con pines digitales y analógicos, diferentes protocolos de comunicación como: UART, I2C, SPI [17].



Figura 1.3 Placa de microcontrolador Arduino Uno R3 [17]

No obstante, es posible implementar diversos componentes externos, para las estimulaciones se necesitan medir parámetros de aceleración y/o sentido de giro, por lo que se puede añadir el módulo LSM9DS1 (módulo embebido en el Arduino Nano 33 BLE) que cuenta con 9 grados de libertad (DOF), teniendo como sensores: acelerómetro, magnetómetro y giroscopio, cada uno con sus respectivos 3 ejes, para un entorno de espacio 3D. [18].

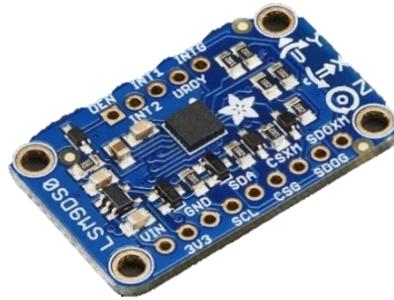


Figura 1.4 Módulo LSM9DS1 9-DOF [18]

1.4.3 Dispositivos comerciales para el tratamiento del CDM

En el país no se cuenta con dispositivos especializados para la estimulación en pacientes con CDM, sin embargo, existen varios estudios y productos localizados fuera del país, uno de ellos es NexStride, siendo un pequeño dispositivo que emite señales visuales y auditivas que permiten liberar al paciente del estado de congelamiento, reestableciendo la comunicación entre el cerebro y el cuerpo. El dispositivo se ubica en bastones o andadores propios del paciente; cuenta con carga vía USB y duración de 6.5 horas de uso continuo [5].



Figura 1.5 Dispositivo de movilidad NexStride [5]

También existen tratamientos médicos utilizando estimuladores no invasivos, siendo el caso de GONDOLA, un dispositivo de terapia médica, donde el paciente requiere una cita para evaluar y definir los mejores ajustes del dispositivo, ser llevado a casa y aplicarse el tratamiento según lo indicado. El método aplicado es la estimulación periférica mecánica automatizada (AMPS) que consiste en aplicar presión en cuatro áreas de la planta del pie, que mejora una autonomía de movimiento y menor riesgos de caídas en pacientes con CDM [19].



Figura 1.6 Dispositivo de tratamiento médico GONDOLA [19]

1.4.4 Estado del arte

A lo largo de los años y en distintos países se han planteado soluciones para poder controlar el CDM que sufren algunos de los pacientes con la enfermedad de Parkinson, los cuales pueden ser tratados por medio de diferentes tipos de estimulaciones al momento de combatir al CDM.

1.4.4.1 Estimuladores vibratorios

En Ecuador, se creó un dispositivo basado en estimuladores vibratorios colocados en la planta de los pies, los cuales serán accionados de forma manual por un operador externo al paciente, ya sea el doctor o una persona acompañante, de esta forma se logró minimizar los momentos donde la persona se veía afectada por el CDM mientras atravesaba por obstáculos y cambios de terreno [20]. Posteriormente, otros dispositivos basados en el mismo principio fueron

desarrollados en la Universidad Salesiana de Cuenca por Mónica Huerta, et al. [21], basándose en la adquisición y procesamiento de datos para generar el estímulo final.



Figura 1.7 Estimulación vibratoria para reducir episodios de CDM [20]

1.4.4.2 Estimuladores visuales

El departamento de bioingeniería politécnico de Milano en Italia, planteo una solución con respecto al combate del CDM en los pacientes con EP que se basa en estímulos visuales generados por medio de unas gafas diseñadas para mejorar los flujos continuos de movimientos hacia delante y atrás a partir del campo periférico del usuario y otros estímulos que son generados debido a eventos que ocurren en el exterior [22].



Figura 1.8 Dispositivo de estimulación óptica para combatir el CDM [22]

1.4.4.3 Estimuladores auditivos

En la universidad de Coruña en España, se diseñó un dispositivo que envía estímulos auditivos de forma rítmica a los pacientes con Parkinson cuando se encuentran en los episodios de congelamiento de la marcha, dando como resultado una mejora en el movimiento de estos individuos (velocidad, cadencia, entre otras variables) cuando se establece una frecuencia específica para generar las estimulaciones rítmicas que disminuyen los efectos del CDM [23].

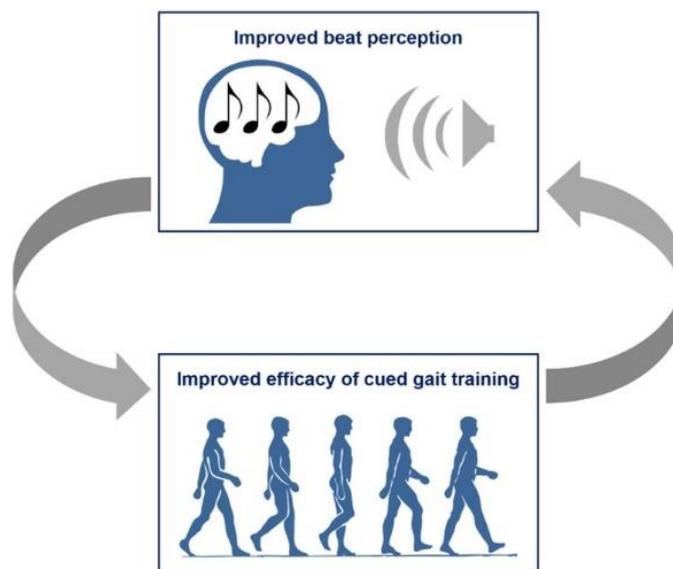


Figura 1.9 Modelo de estimulación auditiva [24]

CAPÍTULO 2

2. METODOLOGÍA

Este capítulo presenta y describe los procedimientos utilizados para solucionar la problemática del proyecto. La etapa de metodología está constituida como: primero, se plantearon los requerimientos para el desarrollo de la solución; segundo, se definieron las diferentes alternativas de solución y la elección de una de ellas; tercero, se elaboró el diseño detallado de forma esquemática; cuarto, se organizó cada una de las etapas del proceso de diseño; por último, se encontraron los parámetros para los diferentes tipos de diseño de todo el sistema.

2.1 Requerimientos de diseño

Los requerimientos para la resolución del proyecto basado en las principales características enfocada a nuestro cliente se definen en la Tabla 2.1, considerando algunos criterios utilizados para la selección de la alternativa de solución.

Tabla 2.1 Requerimientos de diseño del sistema

Criterio	Requerimiento
Comodidad	El sistema debe ser compacto y ligero, de forma que no genere incomodidad
Precio	Dispositivo asequible al cliente comparado con otros productos de estimulación en el mercado
Facilidad de uso	Fácil de manejar y controlar, evitando confusiones sin la ayuda de otra persona externa
Uso de sensores	Sensores de fácil implementación en todo el sistema
Guardado de archivos	Guardar archivos de texto con los datos en formato de separación por comas
Etiqueta de datos	Requerir de la escritura de datos remoto para el etiquetado de datos
Creación de nuevos datos	Creación de diferentes archivos de forma automática por cada sujeto de prueba

2.2 Selección de alternativa de solución

Tomando en consideración los tipos de estimulaciones y hardware descritos en el capítulo 1, además de la búsqueda de otros elementos disponibles que sean posibles de encontrar en el mercado local, se logró establecer 3 tipos de soluciones presentados de la siguiente forma:

- **Alternativa 1**

Diseño de estimulador auditivo con Arduino Nano 33 BLE Sense, ya equipado con sensores de movimiento, aceleración, sonido, entre otros, además de integrar funcionalidades básicas ya embebidas para machine learning, permitiendo una activación automática.

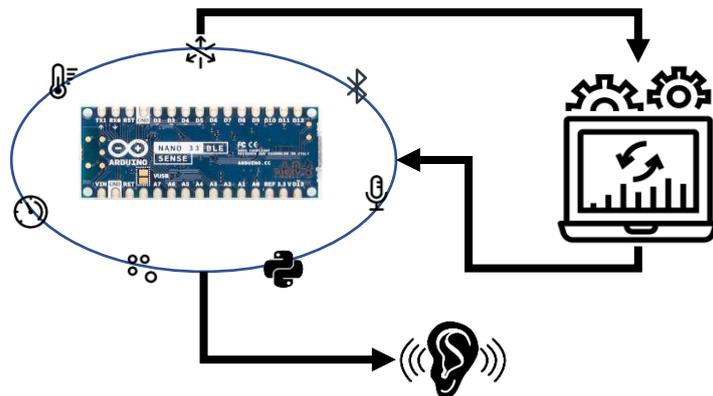


Figura 2.1 Diagrama de diseño de la alternativa 1

- **Alternativa 2**

Diseño de estimulador eléctrico con Arduino Nano 33 IoT y dos acelerómetros MPU-6050, obteniendo datos de aceleración y giroscopio que servirán para entrenar un modelo de inteligencia artificial, siendo la responsable de activar automáticamente los estimuladores eléctricos colocados en las extremidades inferiores cuando el sujeto presente episodios de CDM.

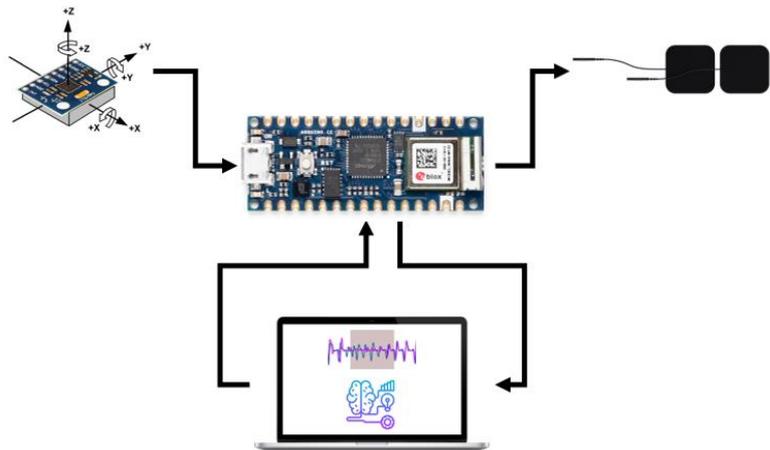


Figura 2.2 Diagrama de diseño de la alternativa 2

- **Alternativa 3**

Diseño de estimulador vibratorio utilizando Raspberry Pi 3 combinada con el módulo LSM9DS1, permitiendo recolectar datos de magnetismo, aceleración y giroscopio, los cuales serán almacenados y procesados en un algoritmo de IA encargado de la activación automática.

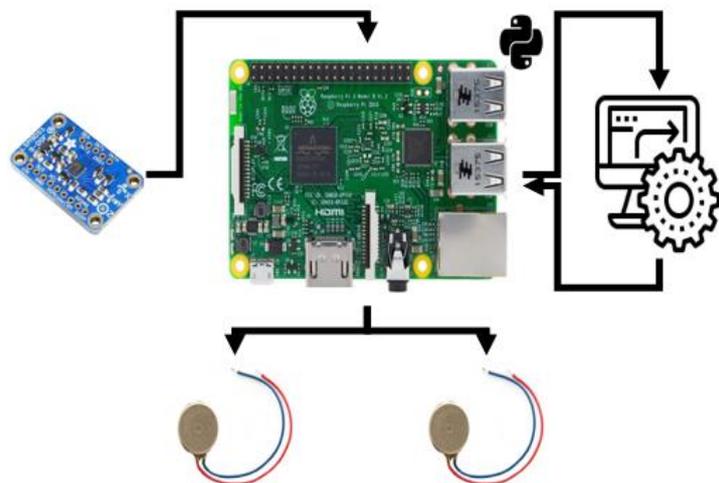


Figura 2.3 Diagrama de diseño de la alternativa 3

Para determinar la mejor alternativa se establecieron 6 criterios de selección relacionados al enfoque del diseño del sistema, una breve descripción de cada uno de ellos se muestra en la Tabla 2.2.

Tabla 2.2 Descripción de los criterios de selección

Criterio	Descripción
Comodidad	Características físicas que pueden influir en el prototipo final
Precio	Valor total que posee la implementación de la alternativa
Funcionalidad	Los resultados obtenidos se acercan a los esperados
Viabilidad	Posibilidad de elaborar el sistema en el país con los recursos planteados
Innovación	Novedades al momento de utilizar diversos recursos e implementaciones
Facilidad de uso	Facilidad de creación, implementación y compatibilidad de los elementos

Se definió la evaluación del peso específico de cada criterio considerando la importancia que tiene cada uno en relación con los otros, en base a ello se obtuvo los pesos y ponderaciones que permitieron determinar la mejor alternativa de solución del proyecto (Apéndice A), los valores se encuentran en la Tabla 2.3.

Tabla 2.3 Pesos y ponderaciones de los criterios de selección

Evaluación del peso específico		
Viabilidad = Funcionalidad > Comodidad = Precio > Facilidad de uso > Innovación		
Criterio	Peso	Ponderación (%)
Comodidad	3.5	16.67
Precio	3.5	16.67
Funcionalidad	5.5	26.19
Viabilidad	5.5	26.19
Innovación	1	4.76
Facilidad de uso	2	9.52
Total	21	100

Se realizó una matriz de decisión para determinar la mejor alternativa para el diseño del sistema. Basándose en los resultados de la Tabla 2.4, se observó que la alternativa 2 es la más favorable en cumplir en mayor medida los criterios de funcionalidad y viabilidad que tienen una alta ponderación de importancia, además con un precio en los componentes más económicos.

Tabla 2.4 Matriz de decisión

Criterios	Peso	Alternativa 1		Alternativa 2		Alternativa 3	
		Puntaje sin peso	Puntaje con peso	Puntaje sin peso	Puntaje con peso	Puntaje sin peso	Puntaje con peso
Comodidad	3.5	3	10.5	2	7	1	3.5
Precio	3.5	2	7	3	10.5	1	3.5
Funcionalidad	5.5	2	11	1	5.5	3	16.5
Viabilidad	5.5	1	5.5	3	16.5	2	11
Innovación	1	2	2	3	3	1	1
Facilidad de uso	2	2	4	3	6	1	2
		Total	40	Total	48.5	Total	37.5

2.3 Diseño detallado

Se realizó un diagrama de funcionamiento basado en la alternativa seleccionada con todos los subsistemas sobre la recolección de datos y envío de estímulos eléctricos al usuario, esto se visualiza en la Figura 2.4.

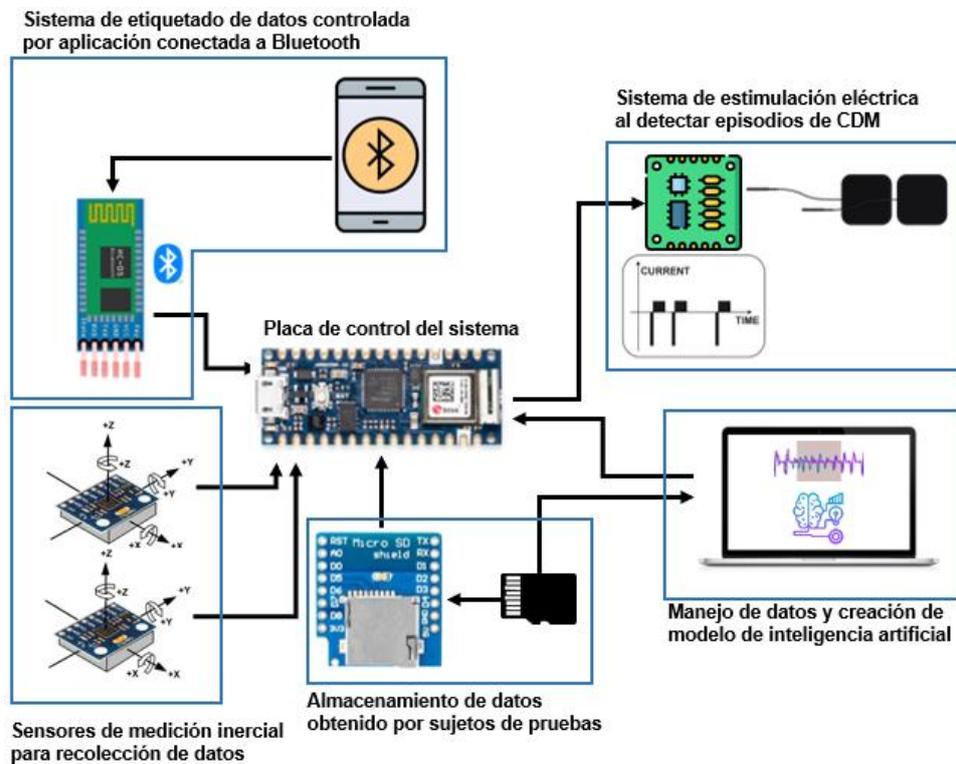


Figura 2.4 Diagrama de diseño conceptual

2.4 Proceso del diseño

En la elaboración del proceso de diseño se abordaron 5 etapas para llegar a la solución del problema, logrando cumplir los requerimientos y llevando una progresión en el desarrollo del sistema. Se pueden observar las etapas en el diagrama de flujo de la Figura 2.5:

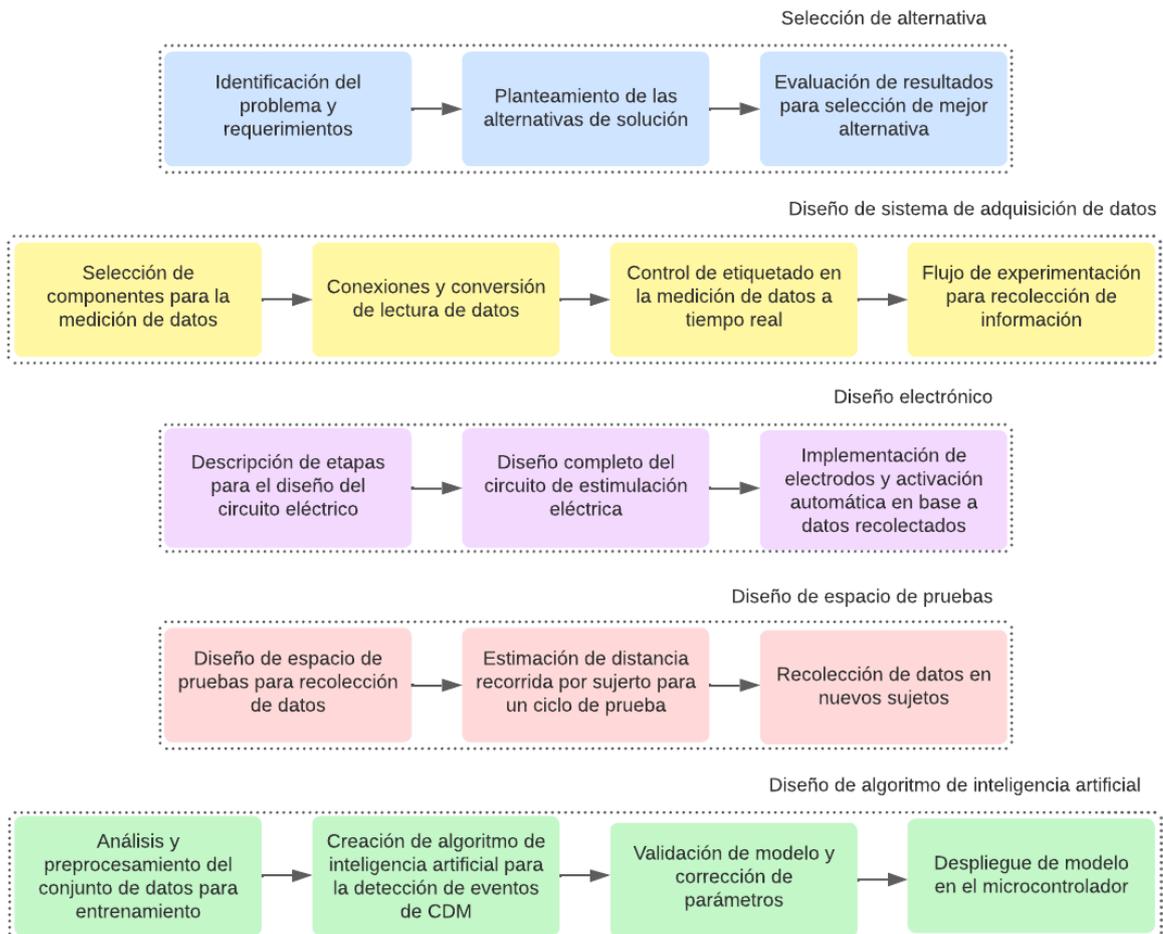


Figura 2.5 Diagrama de proceso del diseño

2.5 Diseño de sistema de adquisición de datos

El sistema de adquisición de datos se compuso por un microcontrolador, sensores inerciales, comunicación y almacenamiento, donde los nombres de los componentes y su información técnica se encuentra en la siguiente subsección.

2.5.1 Selección de componentes

Para la elaboración de este diseño se detallaron los aspectos técnicos de cada uno de los elementos utilizados para la recolección de datos, las siguientes tablas muestran las características fundamentales para la comprensión y conexión correcta del circuito.

Tabla 2.5 Especificaciones técnicas de placa de control [25]

Placa de control Arduino	
Modelo	Nano 33 IoT
Microcontrolador	ATSAMD21
Conector USB	Mini-B USB
Comunicación	UART / I2C / SPI
Voltaje de alimentación	4.5 – 21 VDC
Voltaje de entrada USB	5 VDC
Voltaje de operación	3.3 VDC
Corriente de entradas/salidas digitales	7 mA
Memoria SRAM	32 kB
Memoria Flash	256 kB

Tabla 2.6 Especificaciones técnicas del sensor inercial [26]

Sensor de medición inercial	
Modelo	MPU-6050
Voltaje de alimentación	2.375 – 3.46 VDC
Protocolo de comunicación	I2C
Salida digital	Ejes X-Y-Z
Acelerómetro	
Rango de escala	±2g, ±4g, ±8g y ±16g
Corriente de operación	3.6 mA
Giroscopio	
Rango de escala	±250, ±500, ±1000 y ±2000
Corriente de operación	500 µA

Tabla 2.7 Especificaciones técnicas del módulo bluetooth [27]

Módulo Bluetooth	
Modelo	HC-05
Voltaje de alimentación	3.3 – 6 VDC
Consumo de corriente	50 mA

Protocolo de comunicación	UART
Velocidad de transmisión	1200 bps hasta 1.3 Mbps
Tasa de baudios	38400
Frecuencia	2.4 GHz
Alcance	10 m

Tabla 2.8 Especificaciones técnicas de módulo de almacenamiento [28]

Módulo de almacenamiento	
Tarjeta	MicroSD Shield
Voltaje de alimentación	3.3 VDC
Protocolo de comunicación	SPI

2.5.1.1 Conexiones del sistema de adquisición

Las conexiones realizadas en el DAQ se pueden apreciar en la Figura 2.6, en donde se tiene la lectura de datos con los sensores MPU-6050, el etiquetado por medio de la aplicación móvil que se conecta al módulo HC-05, el almacenamiento de archivos a través del MicroSD y el microprocesador Arduino Nano 33 IoT que ejecuta los comandos (Apéndice B).

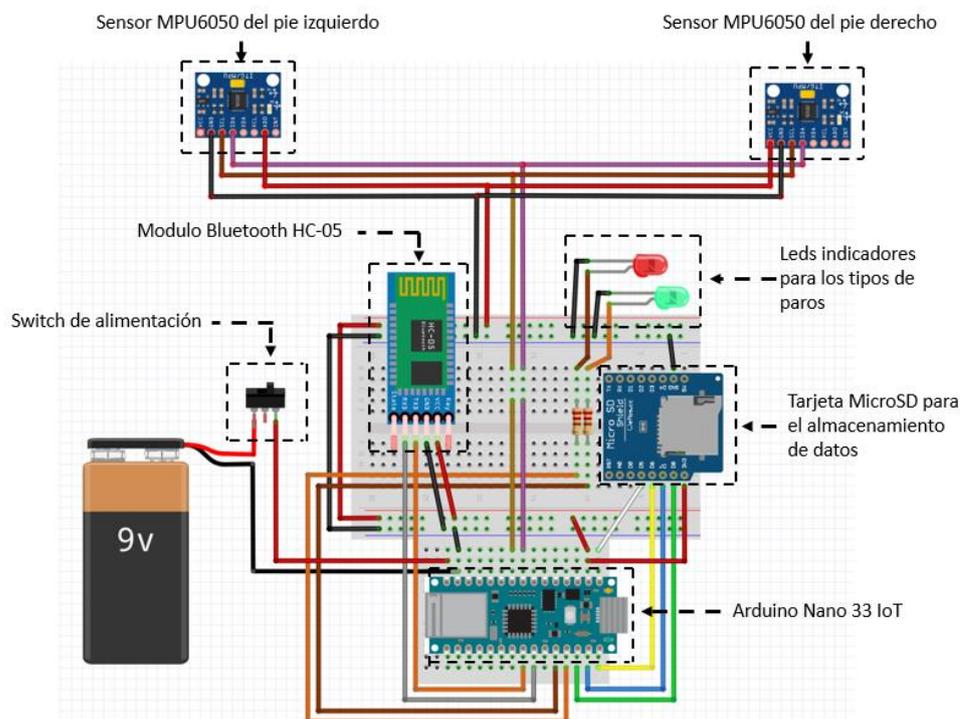


Figura 2.6 Conexiones en el sistema DAQ

2.5.2 Conversión de lecturas del sensor

El sensor inercial MPU-6050 posee un rango de lectura de ± 32768 bits en cada una de sus variables y sus respectivos ejes XYZ [29]. Sin embargo, para el análisis de datos fue necesario escalarlos y convertirlos a resultados unidimensionales y valores estándares para cada variable.

Se escogieron los intervalos por defecto para las variables trabajadas, obteniendo $\pm 2g$ y $\pm 250^\circ/s$ para la aceleración y velocidad angular respectivamente [29]. El proceso de conversión de datos se basó en una sola formula que luego fue modificada para cada variable utilizada que consiste en la ecuación 2.1.

$$\text{Variable escalada} = \text{Lectura de datos} \times \frac{\text{Rango de escalamiento}}{\text{Rango de bits}} \quad (2.1)$$

Cada variable tiene un rango de escalamiento diferente; por lo tanto, al remplazar estos valores, la ecuación 2.1 cambio en cada una de ellas y así se obtuvieron los siguientes resultados con sus unidades respectivas:

- Aceleraciones en XYZ

$$a_{xyz} \text{ escalada} = \text{lectura de } a_{xyz} \times \frac{9.81}{16384} [m/s^2]$$

- Velocidades angulares en XYZ

$$\omega_{xyz} \text{ escalada} = \text{lectura de } \omega_{xyz} \times \frac{250}{32768} [deg/s]$$

2.5.3 Etiquetado de datos

Es fundamental etiquetar los datos recopilados por los sujetos sanos, con el fin de determinar las detenciones voluntarias e involuntarias. Las detenciones involuntarias pueden ser simuladas por los sujetos imitando el estado de congelamiento de la marcha. La identificación precisa de estos estados es crucial para comprender plenamente el comportamiento de la enfermedad de Parkinson.

2.5.4 Diagrama de flujo para experimentación

El sistema para la adquisición y etiquetado de datos se describió a partir de un diagrama de flujo presente en la Figura 2.8, donde se explicó todos los pasos del funcionamiento respectivo para cada sujeto de pruebas.

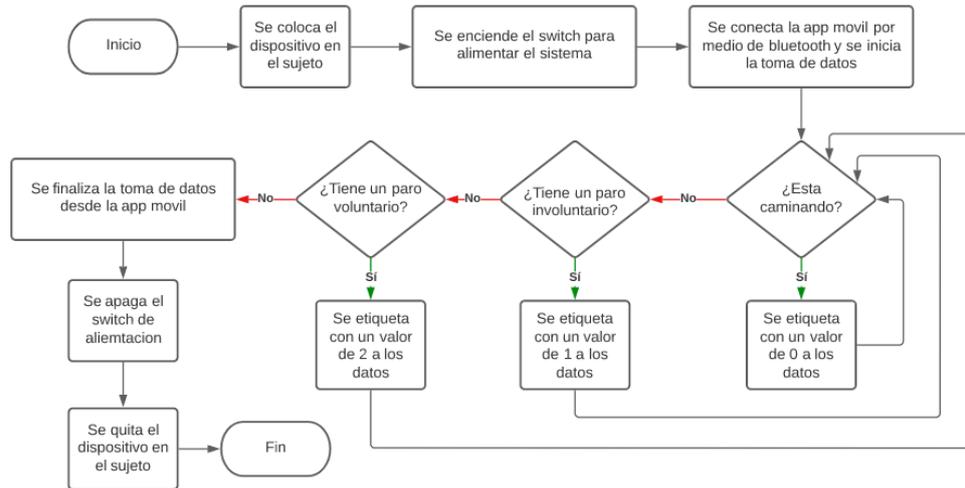


Figura 2.8 Diagrama de flujo del funcionamiento

2.6 Diseño electrónico del sistema de estimulación eléctrica

Para el diseño del sistema de estimulación eléctrica y la elaboración del circuito se empleó la normativa eléctrica DIN, además se consideró diferentes parámetros y factores de seguridad en pacientes. Los estímulos enviados son ondas de pulsos monofásicos con determinada frecuencia, amplitud, y ancho de pulso. Basado en estudios anteriores [31] se toman en cuenta las siguientes especificaciones que a su vez cumplen con la normativa IEC 60601-1: frecuencia de estimulación entre 20 – 100 Hz, amplitud del pulso < 80 V y ancho de pulso entre 60 – 800 μ s.

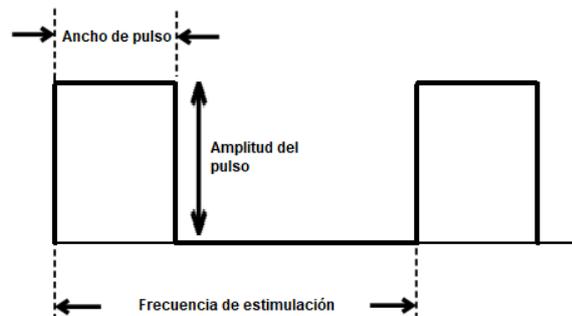


Figura 2.9 Pulsos de corriente continua monofásica

2.6.1 Bloques del circuito de estimulación

Considerando que se desea una salida de ondas de pulso, se implementó el circuito integrado (CI) LM555. Para ello se recibe en la entrada una señal de disparo como accionamiento, realizada por el Arduino Nano 33 IoT, siendo la base para lograr extender el ancho de pulso a la salida, el circuito se puede observar en la Figura 2.10.

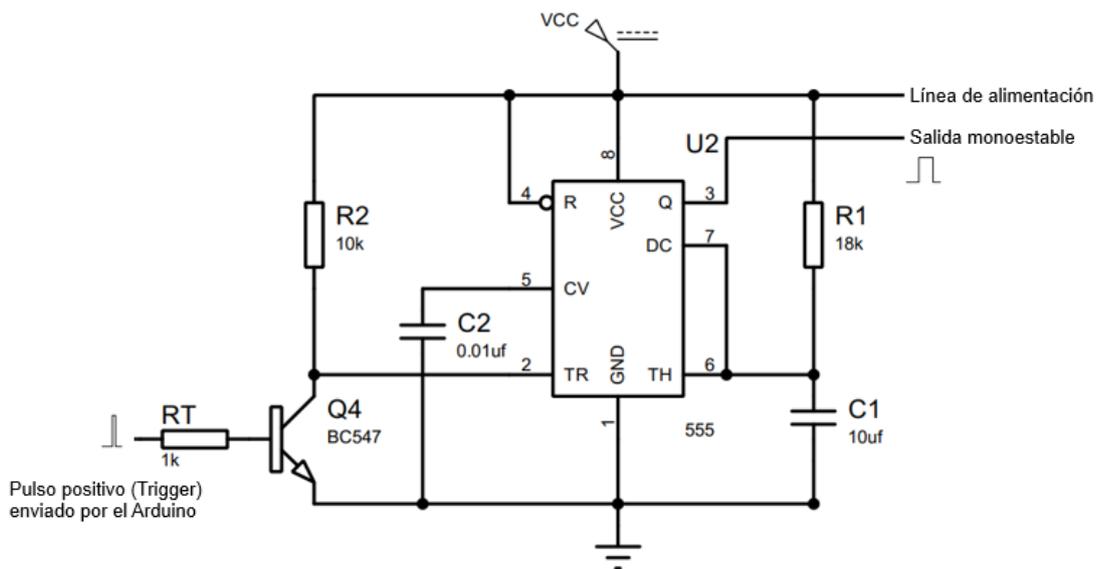


Figura 2.10 Circuito multivibrador monoestable

La fórmula para delimitar el tiempo de estimulación (t) controlado para un pulso de entrada pequeña sería la siguiente:

$$t = 1.1 \times R_1 \times C_1 \quad (2.2)$$

Obteniendo como resultados en base al circuito realizado:

$$t = 1.1 \times (18K\Omega) \times (10\mu F)$$
$$t = 0.2 \text{ s}$$

Si el disparo de entrada es inferior al tiempo t , entonces su límite de extensión de pulso llegará hasta t , sin embargo, si el disparo es mayor, entonces la estimulación se detendrá cuando se deje de emitir el disparo.

Una vez se consiguió controlar esta variable, se implementó otro 555 para crear los pulsos deseados, el diseño se puede observar en la Figura 2.11. Es importante considerar que los componentes que acompañan al CI están predefinidos y solo se modifican sus valores para la salida deseada.

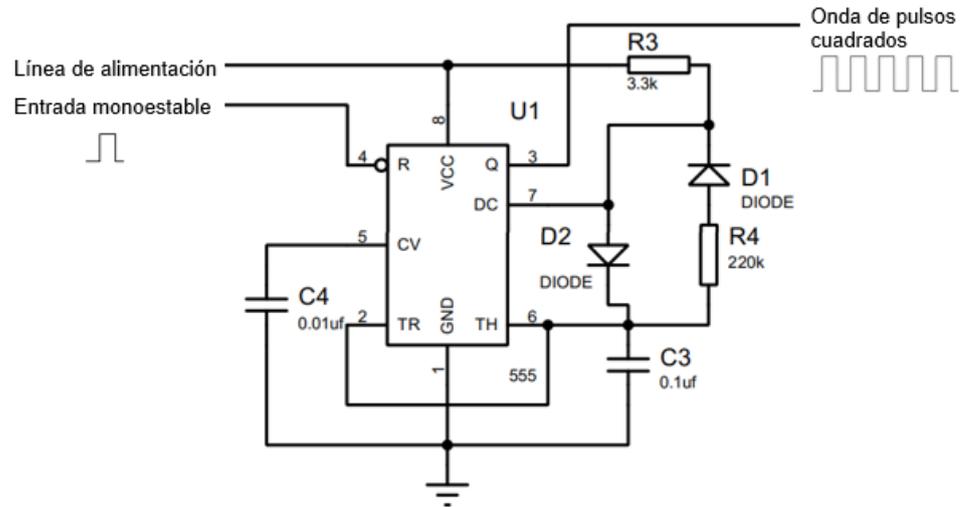


Figura 2.11 Circuito generador de pulsos cuadrados

Basado en las especificaciones del estimulador eléctrico, se consideró una frecuencia aproximada de 30 Hz y para el ancho de pulso un valor aproximado de 200 μ s o 0.2 ms, con sus siguientes ecuaciones:

$$f = \frac{1.44}{(R_3 + (2 \times R_4)) \times C_3} \quad (2.3)$$

$$T_{ON} = 0.693 \times R_3 \times C_3 \quad (2.4)$$

Obteniendo como resultado la frecuencia (f):

$$f = \frac{1.44}{(3.30 \text{ K}\Omega + (2 \times 220 \text{ K}\Omega)) \times 0.10 \mu\text{F}} = 32.48 \text{ Hz}$$

Y para el tiempo de encendido o ancho de pulso (T_{ON}):

$$T_{ON} = 0.693 \times (3.30 \text{ K}\Omega) \times (0.10 \mu\text{F})$$

$$T_{ON} = 228.69 \mu\text{s}$$

Una vez obtenida las ondas de pulso cuadradas, se debe de amplificar el voltaje de salida, para lo cual se realizó un convertidor elevador (Boost converter), el diseño del circuito fue el siguiente de la Figura 2.12.

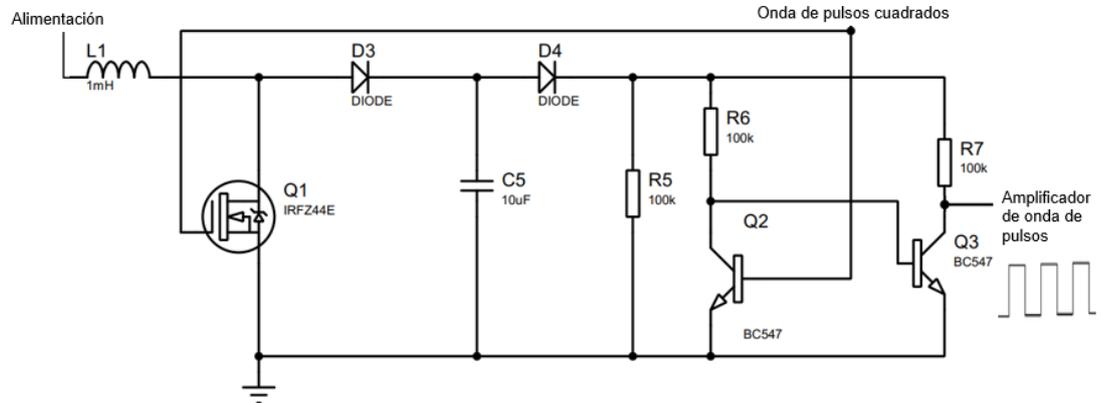


Figura 2.12 Circuito elevador de voltaje

Para la elevación de voltaje, se tiene en cuenta las especificaciones definidas en un inicio y se maneja un voltaje de salida aproximado de 50 V. El diseño del circuito juntando cada uno de los bloques se puede observar en la Figura 2.13 a continuación:

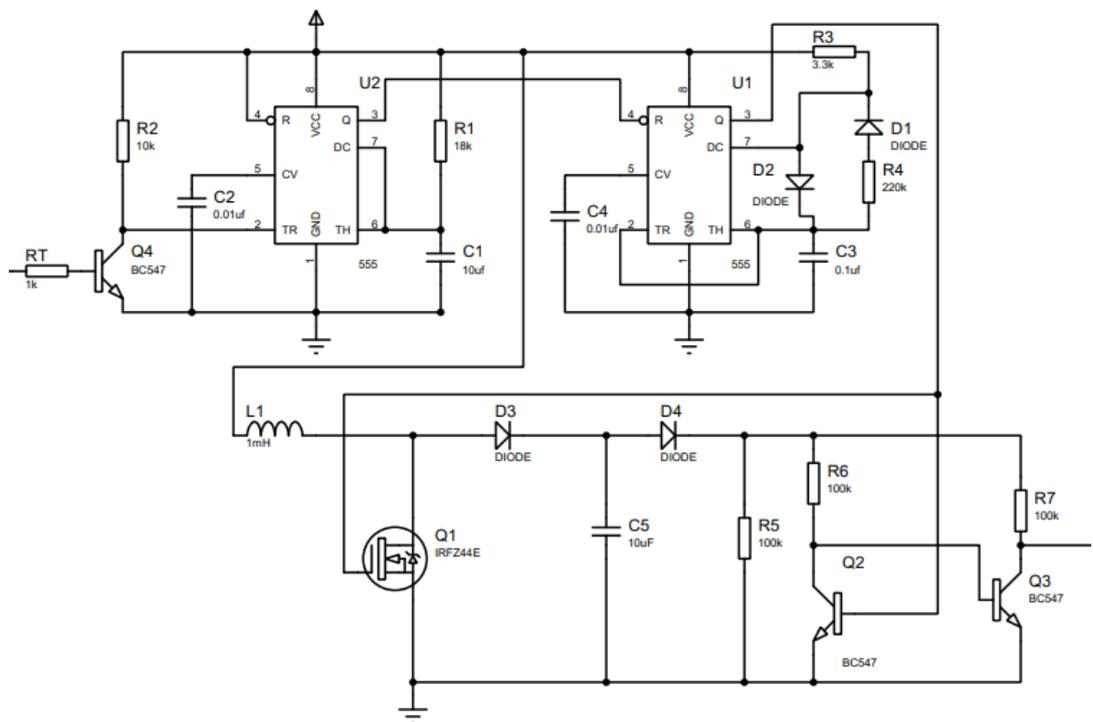


Figura 2.13 Circuito de estimulación eléctrica

2.7 Diseño de espacio de pruebas

Los datos obtenidos para el análisis de datos provinieron de varias pruebas experimentales con sujetos sanos, a los cuales se les hizo firmar un acuerdo de consentimiento en donde se les explicaba los procedimientos a los cuales se les iba a someter. El formato del consentimiento informado se encuentra en el Apéndice C.

2.7.1 Dimensionamiento de zona de pruebas

Se definió un lugar en particular donde puedan movilizarse los sujetos de prueba, donde se tienen medidas específicas y los diferentes tipos de obstáculos en donde se esperó que se presente un episodio de CDM, para los cuales se tuvieron: puertas, giros, obstáculos, entre otros. En la Figura 2.14 el tramo que deben recorrer.

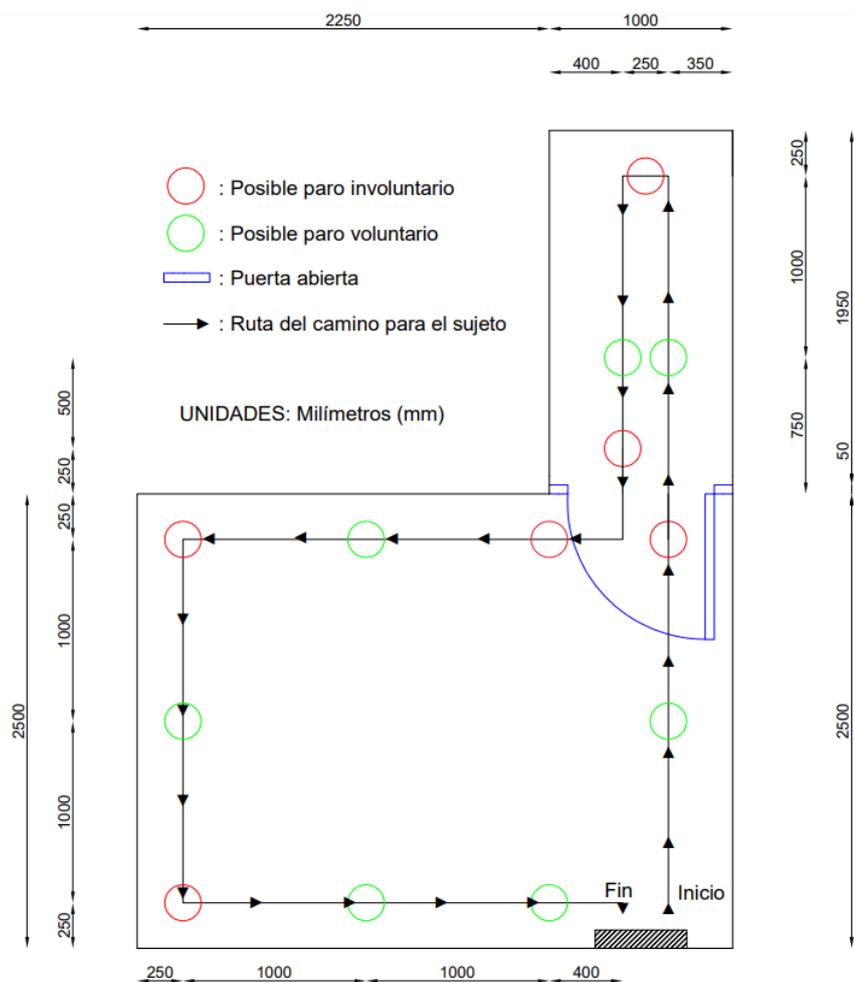


Figura 2.14 Vista superior de la zona de pruebas

2.7.2 Pruebas experimentales en sujetos sanos

Los pacientes de Parkinson con CDM poseen poco tiempo disponible para participar en las pruebas debido a la baja frecuencia con la que visitan las citas médicas. Por lo tanto, para conseguir los datos fue necesario obtener estos datos a partir de sujetos sanos que imiten el comportamiento de CDM, para lo cual se les tuvo que mostrar varios videos en donde se aprecia cómo afecta el congelamiento de marcha en el movimiento.

En la toma de datos de los 20 sujetos sanos, cada uno realizó un patrón diferente de paros, es decir, la cantidad y continuidad de paros voluntarios e involuntarios varió entre cada uno de ellos a partir de las posibles localidades previamente establecidas en la zona de pruebas. Para ello se iba visualizando el comportamiento del sujeto y manualmente con la aplicación móvil se iban etiquetando los datos, dependiendo del estado en el que se encuentre en aquel momento.

2.8 Diseño de algoritmo de inteligencia artificial

El intervalo de tiempo con el que se obtuvieron los datos está regulado por las velocidades internas que poseen los sensores, resultado en un aproximado de 45 milisegundos [ms]. La frecuencia de muestreo para la adquisición de datos fue de 22 a 24 Hz, la cual pudo ser calculada a partir de la ecuación experimental 2.5.

$$Frecuencia\ de\ muestreo\ (Hz) = \frac{Cantidad\ de\ datos\ muestreados}{Tiempo\ de\ muestreo\ (s)} \quad (2.5)$$

Con la ecuación 2.2 se obtuvo el resultado de frecuencia previamente ya nombrado, se remplazaron las variables con los valores obtenidos de una de las pruebas experimentales con los sujetos sanos.

$$f = \frac{1463}{62.45 [s]}$$
$$f = 23.43 [Hz]$$

Para comprobar que el sistema esté funcionando correctamente se graficaron los datos de un sujeto de pruebas, lo cual demostró que las variables presentan patrones similares entre ellas.

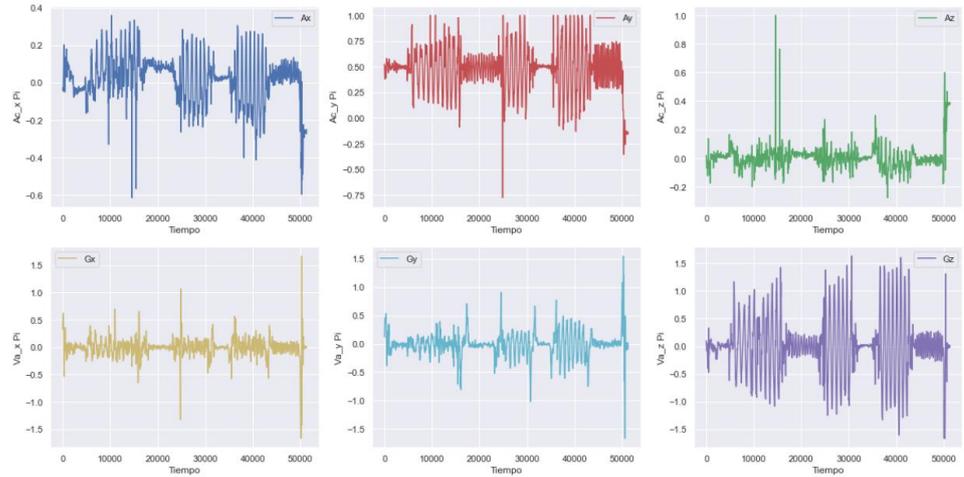


Figura 2.15 Graficas de las variables para el pie izquierdo

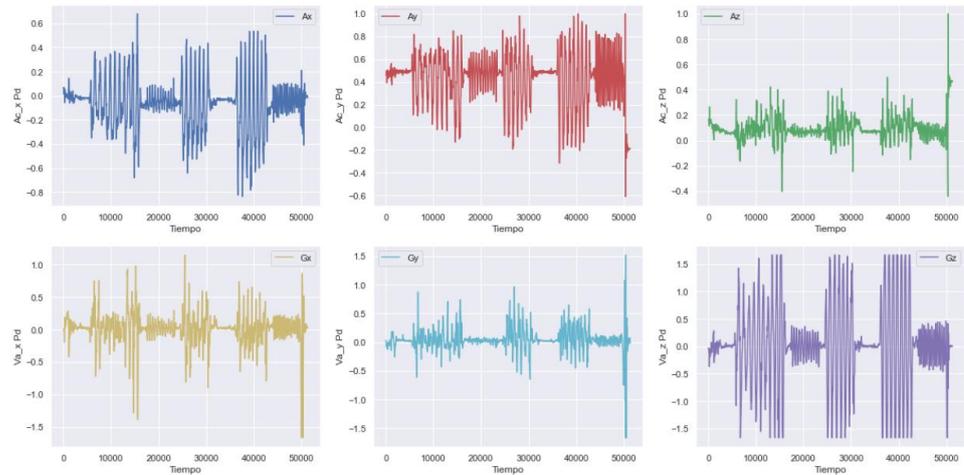


Figura 2.16 Graficas de las variables para el pie derecho

2.8.1 Preprocesamiento de datos

Se realizó un preprocesamiento para la correcta manipulación de los datos, en donde se agruparon todos los archivos para obtener un conjunto de datos con toda la información, también se reemplazó la etiqueta “2” en los datos por la de “0”, de forma que solamente se tuvieran las categorías de CDM y No CDM; además, se redefinieron los índices de tiempo a valores continuos únicos de enumeraciones.

Por último, se realizó el escalamiento de los datos con “MinMaxScaler”, el cual consistió en distribuir los datos entre valores mínimos y máximos para cada una de las variables. Además, se rediseñaron las categorías de entrenamiento y prueba, siguiendo la estructura de una serie de tiempo, obteniendo así “ventanas de datos” creadas a partir de varias filas, en donde cada ventana continúa a partir del final de la anterior. En el Apéndice D se presenta mayor detalle del preprocesamiento.

2.8.2 Creación y despliegue del modelo de inteligencia artificial

Para el algoritmo se escogió la implementación de la red neuronal perceptrón multicapa (MLP por sus siglas en inglés), la elección de esta red es consecuente a la capacidad computacional que tiene el Arduino Nano 33 IoT, considerando también que la MLP genera buenos resultados con menor cantidad de cálculos internos. El código del proceso y creación del modelo se detalla en el Apéndice E.

El concepto general de una MLP con enfoque a este proyecto se basa en la entrada de parámetros, búsqueda y selección de mejores pesos multiplicativos, que reducen el error en la predicción de datos, y en su salida una función de activación que permite realizar una clasificación cuándo los pacientes llegan a presentar o no episodios de CDM. Una descripción gráfica se puede observar en la Figura 2.17.

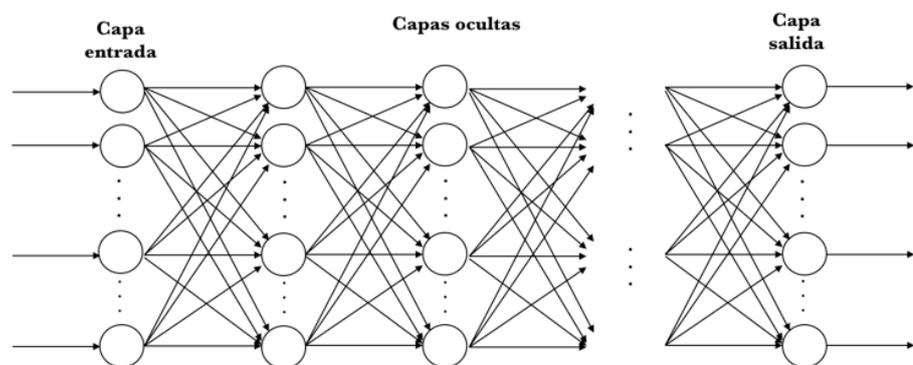


Figura 2.17 Modelo de red neuronal MLP

Se determinó la cantidad de capas ocultas y neuronas mediante un proceso de prueba y validación durante el entrenamiento. La entrada de datos consiste en una ventana de 12 columnas y 10 filas, que se ha transformó a 1 dimensión evitando una gran cantidad de datos en el arreglo. Se definieron las capas ocultas y de salida de la red en función de los resultados de las predicciones hasta lograr una buena precisión.

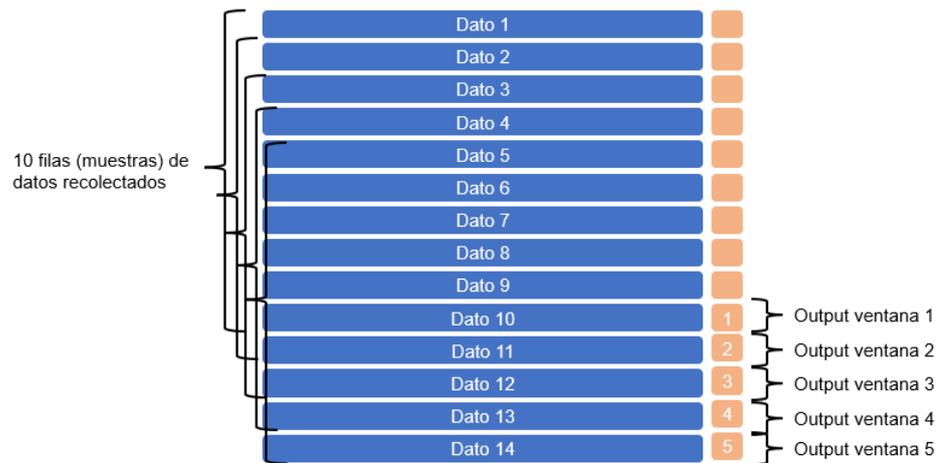


Figura 2.18 Ventana de datos para el entrenamiento

Durante el proceso de compilación, se usó una función de pérdida llamada entropía cruzada binaria, como métrica se usó "accuracy", y se utilizó el optimizador Adam con una tasa de aprendizaje inicial de 0.1. Si la pérdida no disminuía, el valor de la tasa de aprendizaje se redujo usando funciones callbacks.

Después del entrenamiento, los pesos pueden ser cargados en un microcontrolador, como el Arduino Nano 33 IoT, pero es importante tener en cuenta sus limitaciones de memoria ya que el procesamiento de la inteligencia artificial puede superarlas.

CAPÍTULO 3

3. RESULTADOS Y ANÁLISIS

En el presente capítulo se detallan los resultados obtenidos para los sistemas funcionales de la DAQ y la estimulación eléctrica, además de su unificación e incorporación con el código encargado de la detección del CDM, como también la toma de datos en sujetos de prueba, el algoritmo de IA, y las reacciones por el estimulador. Por último, se elaboró un análisis de costos a nivel local con todos los componentes implementados en el sistema.

3.1 Hardware resultante

Esta sección involucra los resultados del proceso del sistema de DAQ, entre ellos la recolección de información dado por los sensores inerciales, el etiquetado de datos y su almacenamiento. Además, se indican los valores obtenidos por parte del estimulador eléctrico y una alternativa para acoplar los componentes en una placa de circuito impreso (PCB). Finalmente, se muestra el diseño de un contenedor para compactar todo el sistema y facilitar de uso.

3.1.1 Sistema de adquisición de datos

Se utilizaron varios componentes para la DAQ, como se muestra el circuito físico en la Figura 3.1, los cuales se basaron en la obtención, etiquetado y almacenamiento de datos para posteriormente ser examinados.

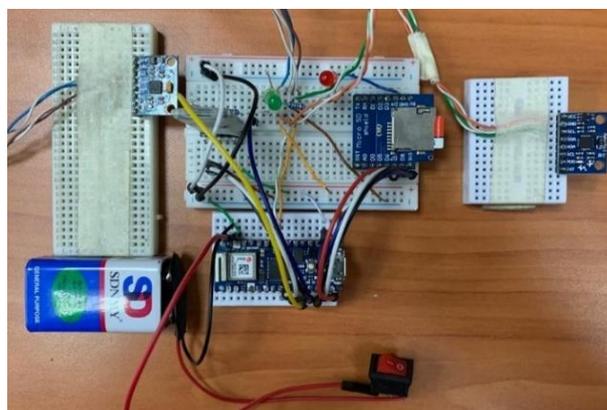


Figura 3.1 Sistema completo de adquisición de datos

Los sensores de movimiento MPU-6050 fueron colocados uno en cada pierna (Figura 3.2), de forma que el eje X estuviera apuntando justo en la misma dirección en la que camina el usuario, el eje Y se encuentra paralelo a la pierna, mientras que el eje Z es perpendicular a esta.



Figura 3.2 Posición de los ejes utilizados para la toma de datos

Después de obtener los datos con los sensores, se los etiquetó simultáneamente por medio de la aplicación móvil, en la Figura 3.3 se enviaron señales por medio del accionamiento de botones para luego almacenarse en un archivo de texto con su etiqueta respectiva al final de cada fila de datos.

```

SUJ_13: Bloc de notas
Archivo Edición Formato Ver Ayuda
22168,-11.10,2.24,-2.33,11.42,-16.73,-4.66,-6.08,7.08,1.40,2.60,-3.34,5.58,0
22220,-10.00,3.39,-2.14,3.34,-3.65,-2.63,-6.76,6.78,0.91,-12.80,19.20,-1.54,0
22262,-6.44,6.32,-5.34,-18.76,26.83,9.05,-7.00,6.74,1.71,-16.80,26.56,9.89,0
22311,-3.34,7.50,-6.58,-0.32,6.74,1.15,-7.95,5.07,2.81,0.96,11.17,1.96,0
22354,-5.14,3.29,-6.35,-4.11,11.94,-11.05,-7.66,3.89,2.90,9.81,-13.54,10.24,0
22396,-8.47,2.41,-4.28,-10.24,7.94,-8.79,-6.73,7.37,-0.64,14.42,-6.46,-6.22,0
22439,-8.73,5.55,-3.89,-11.70,-6.60,0.75,-5.21,8.88,0.34,1.02,1.76,9.92,1
22482,-4.10,7.80,-6.51,-1.45,0.87,20.94,-7.96,5.22,3.61,-20.62,21.05,-2.61,1
22524,-4.46,5.76,-6.63,-0.34,10.02,5.53,-8.56,4.50,1.89,11.69,-6.22,0.35,1

```

Figura 3.3 Etiquetado de datos en tiempo real

En promedio se obtuvieron 1500 filas de datos por sujeto de prueba, representadas por el tiempo, velocidades, aceleraciones y etiquetado. En la señalización de estados, se usaron 2 diodos leds (Figura 3.4), permitiendo al usuario conocer los escenarios de paros voluntarios e involuntarios.

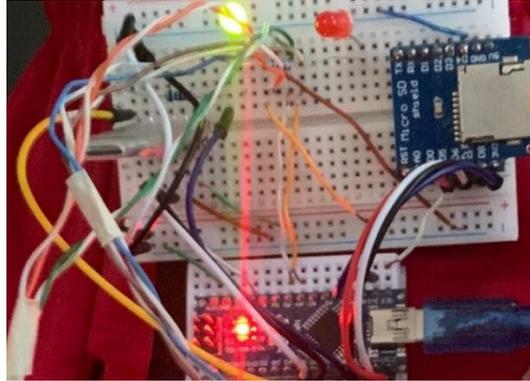


Figura 3.4 Señalética de estados con diodos leds

3.1.2 Sistema de estimulación eléctrica

El sistema de estimulación eléctrica consistió en 3 bloques principales:

- La extensión del ancho de pulso en un tiempo mínimo $t = 0.2 s$.
- El ajuste controlado de ondas cuadradas, teniendo una frecuencia $f = 32.48 Hz$ y, duty cycle $T_{ON} = 228.69 s$.
- La amplificación de voltaje alcanzando un máximo de $V_{Estimulación} \approx 50V$.

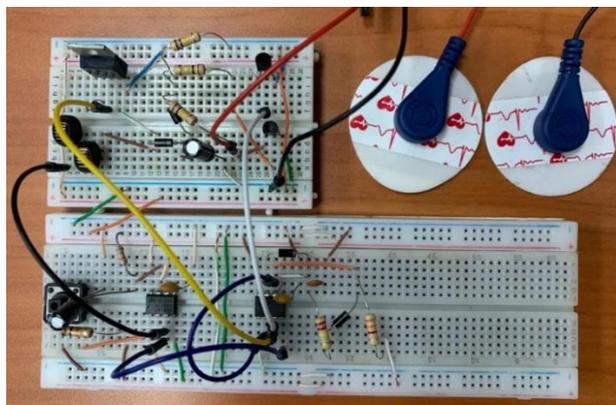


Figura 3.5 Implementación de sistema de estimulación eléctrica

En la Figura 3.5 se mostró el circuito implementado físicamente, y las gráficas obtenidas durante la experimentación se visualizan en el osciloscopio de la Figura 3.6, donde se observó la generación de pulsos cuadrados del segundo circuito integrado 555 y la elevación de voltaje.



Figura 3.6 Simulación de pulsos cuadrados en osciloscopio y elevación

3.1.3 Elaboración de placa PCB

El circuito encargado de la estimulación eléctrica presentó varias dificultades debido a la gran cantidad de cables implementados, además de otros factores adicionales. Incorporando una placa PCB se obtendrían mejores resultados que no fueran alterados por los problemas previamente ya descrito, el diseño de la placa se observa en la Figura 3.7:

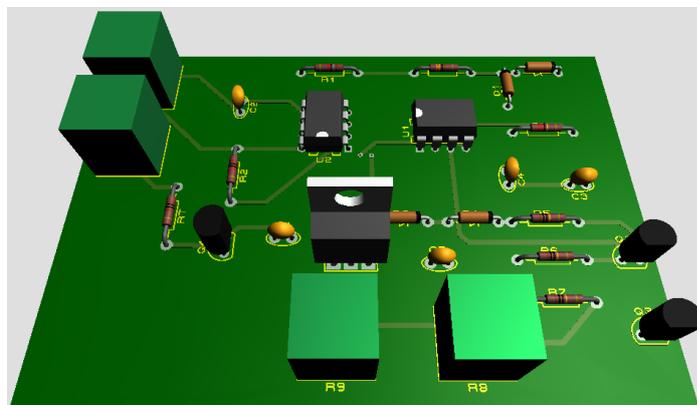


Figura 3.7 Circuito de estimulación eléctrica montada en placa PCB

3.1.4 Contenedor del dispositivo

Las pruebas experimentales demostraron que fue necesario diseñar un contenedor (creado a partir de la normativa ISO) que permita almacenar los componentes electrónicos y la placa PCB del circuito de estimulación, generando una mayor comodidad en el sujeto de prueba y evitar los riesgos de daño en el sistema (Apéndice F).

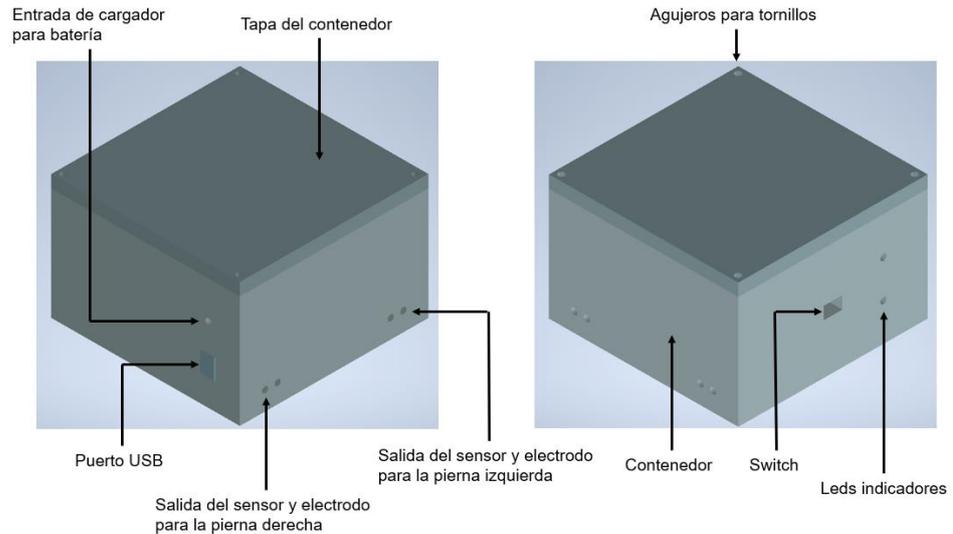


Figura 3.8 Contenedor del dispositivo diseñado en 3D

3.2 Resultados de los experimentos del DAQ

El CDM no posee un patrón común entre los pacientes, por lo cual cada uno de ellos presenta episodios con diferentes particularidades como: el tiempo, la frecuencia, el entorno donde se detienen, entre otros. En la Figura 3.9 se observan 3 ejemplos de sujetos de prueba en la toma de datos, donde las respuestas de cada uno de ellos son independientes.

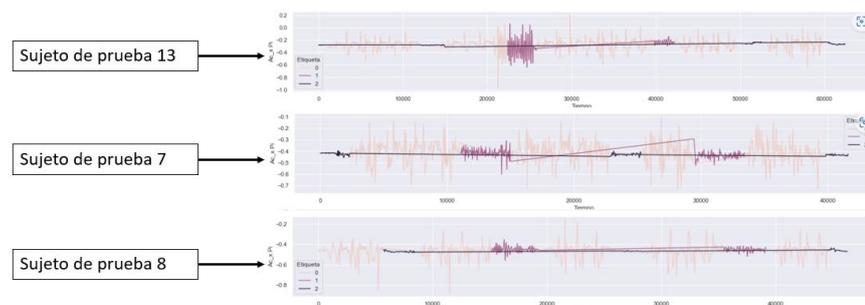


Figura 3.9 Variaciones en los datos obtenidos por los sujetos de pruebas

Los resultados obtenidos a partir de la DAQ llegaron a mostrar varios patrones de congelamiento, la Figura 3.10 muestra la simulación de los sujetos sanos con variaciones en los paros involuntarios y sus cantidades, el tiempo en el cual pueden volver a caminar y la frecuencia con la que tiemblan sus extremidades inferiores.

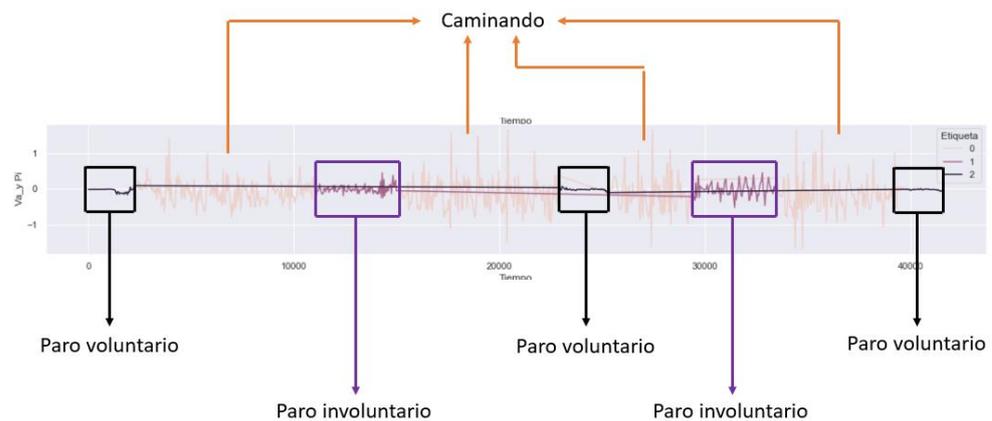


Figura 3.10 Señalización de los estados en las pruebas experimentales

3.3 Resultados del algoritmo de inteligencia artificial

En esta sección se detalla la creación del modelo de perceptrón multicapa. Luego, se comparó las predicciones y los valores reales del set de prueba, por último, se muestra el despliegue en el Arduino Nano 33 IoT para realizar las predicciones en la obtención de nuevos datos. Los resultados del preprocesamiento se encuentran a detalle en el Apéndice D.

3.3.1 Creación del modelo y predicción de muestras

El modelo se entrenó con una MLP, siendo una colección de red fully-connected, para ello se utilizó Tensorflow-Keras. Se inicializó un modelo “*Sequencial*”, que constó de 5 capas en total:

- 1 capa de entrada con 120 valores (Ventana de 10 filas).
- 3 capas densas ocultas con 8 neuronas y activación sigmoideal.
- 1 capa densa de salida binaria dado que con activación sigmoideal.

La Figura 3.11 muestra la estructura del modelo indicando el tamaño de entrada y salida del conjunto de datos, al igual que las capas usadas con sus respectivas activaciones.

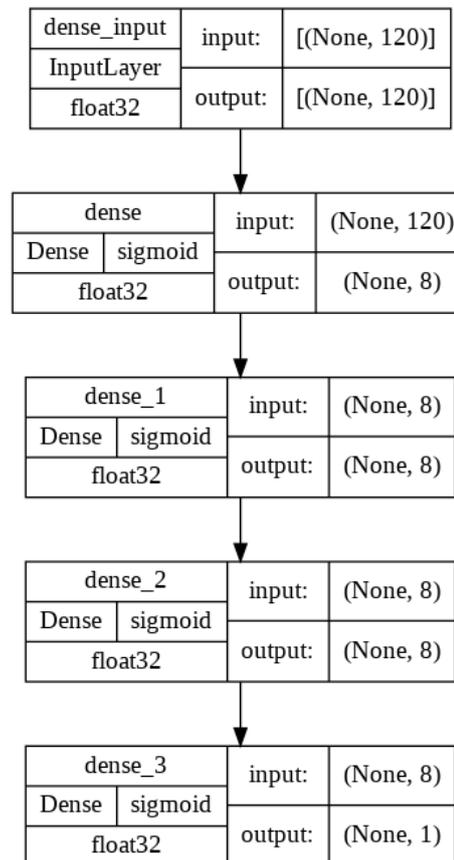


Figura 3.11 Visualización de la estructura general del modelo

Considerando que existen 120 datos por cada ventana, el entrenamiento se realizó con un “*Batch*” de 10, donde se definió en un inicio 100 épocas, sin embargo, se crearon funciones “*Callbacks*” para reducir la tasa de aprendizaje del optimizador, detener el entrenamiento cuando no existan más cambios en la exactitud del set de prueba y guardar el mejor modelo.

Esto obtiene como resultado final un 89.72% en el set de entrenamiento, y 86.12% en el set de prueba, siendo excelentes resultados en base a lo esperado, resultando con pequeñas fallas de predicción en el set de pruebas.

3.3.2 Comparación de predicciones y valores reales

Los resultados de la predicción del set de prueba se comparan con las etiquetas reales para obtener una comparación visual de cómo ha sido el comportamiento de los datos. La Figura 3.12 muestra una vista general donde la etiqueta 1, es bastante similar a la predicción.

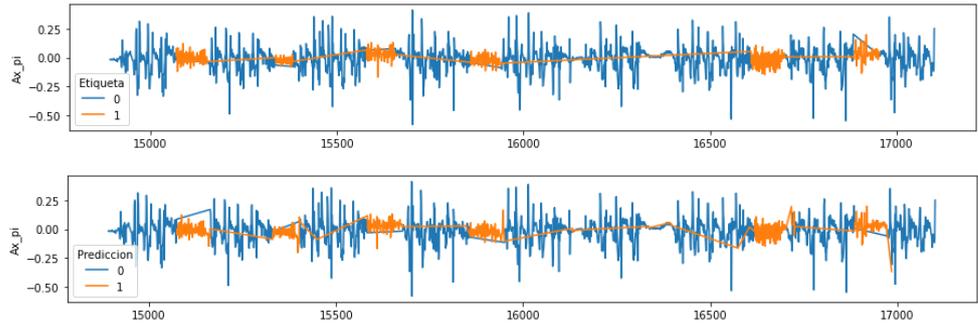


Figura 3.12 Comparación de etiquetas reales y predicción

En los estados de CDM, el algoritmo logró determinar las mismas localidades de dichos estados. En la Figura 3.13 solo se graficó las etiquetas a lo largo del tiempo, siendo los resultados bastante aceptables y los casos donde falla el algoritmo son en periodos cortos, que son controlables en el código de Arduino, donde se esperan 5 predicciones continuas con el valor de 1 para activar el estimulador eléctrico.

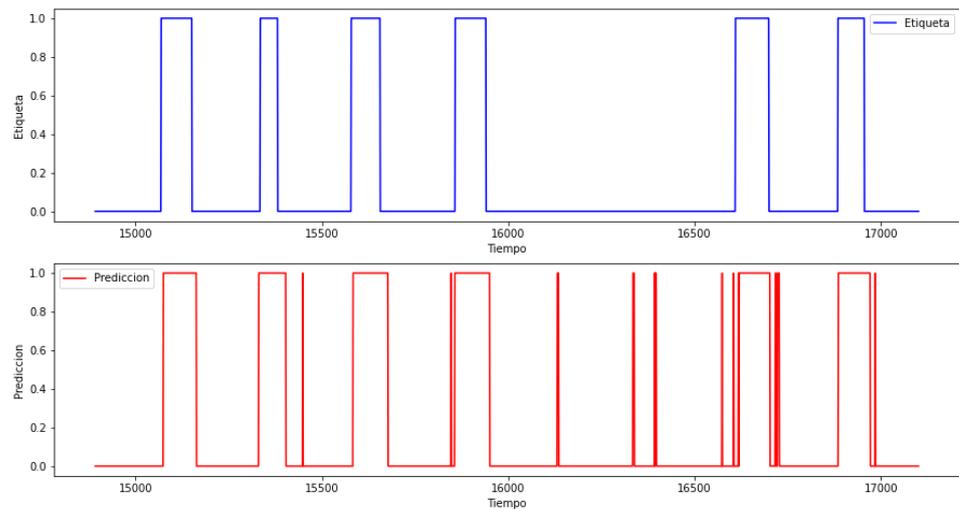


Figura 3.13 Observación de variables “Etiqueta” y “Predicción”

3.3.3 Despliegue de algoritmo en placa de control

Para desplegar el modelo en el Arduino Nano 33 IoT, se implementó la librería AlfES [32], que fue alimentada con los pesos del modelo entrenado, indicándole la cantidad de capas y neuronas del modelo. Para facilitar la recolección de todos los pesos, AlfES tiene una librería encargada de convertir el modelo de Keras en código C, este paso solo fue utilizado para adquirir los pesos en un arreglo de 1 dimensión. La elaboración del código se encuentra en el Apéndice G.

Los resultados se muestran en la Figura 3.14, observando que las señales más pequeñas predicen correctamente el CDM, mientras que en los demás escenarios el algoritmo lo considera como un estado de No CDM.

```
COM6
1.66,1.00,1.69,0.02,0.14,0.25,1.36,0.92,1.80,0.01,0.06,0.19
Prediction: 1
1.47,1.00,1.85,0.28,1.62,0.05,1.24,1.00,1.96,0.08,1.79,0.24
Prediction: 1
1.85,1.00,1.66,0.17,1.70,0.43,1.38,0.89,1.86,1.89,2.00,0.41
Prediction: 1
1.07,1.00,0.05,0.27,1.16,1.66,0.76,0.78,0.09,0.35,1.00,0.55
Prediction: 1
0.54,0.09,0.26,1.74,0.24,1.56,2.55,0.07,0.22,1.92,0.40,1.45
Prediction: 1
0.54,0.43,0.22,0.02,1.95,2.03,0.28,0.40,0.19,0.03,1.96,0.01
Prediction: 1
0.53,0.42,0.22,0.00,1.99,0.01,0.29,0.39,0.19,0.02,0.01,2.00
Prediction: 0
0.55,0.42,0.21,0.01,1.99,0.02,0.31,0.39,0.18,0.02,2.00,0.01
Prediction: 1
0.53,0.43,0.22,0.00,1.99,0.02,0.30,0.40,0.18,0.01,0.01,0.01
Prediction: 1
0.56,0.42,0.21,2.00,1.99,0.03,0.32,0.39,0.18,0.01,0.01,0.01
Prediction: 0
0.53,0.42,0.22,2.00,1.99,0.02,0.31,0.39,0.18,0.02,0.01,0.00
Prediction: 0
0.56,0.42,0.20,0.00,0.00,0.02,0.32,0.39,0.19,0.01,0.02,0.00
Prediction: 0
0.56,0.42,0.21,0.01,2.00,2.03,0.31,0.39,0.20,0.01,0.02,0.00
Prediction: 0
0.56,0.43,0.21,2.00,0.00,0.02,0.32,0.39,0.19,0.02,0.02,0.00
Prediction: 0
```

Figura 3.14 Predicción de nuevos datos recolectados

3.4 Resultados de las estimulaciones en sujetos de prueba

Para obtener la mejor localización sensorial para la reactivación motora en los pacientes se realizaron varias pruebas de estimulación en diferentes zonas, concluyendo que una de ellas corresponde al arco del pie (Figura 3.15), que presentó alta sensibilidad eléctrica debido al poco contacto que posee con los exteriores y la delgada línea de recubrimiento hacia los nervios.

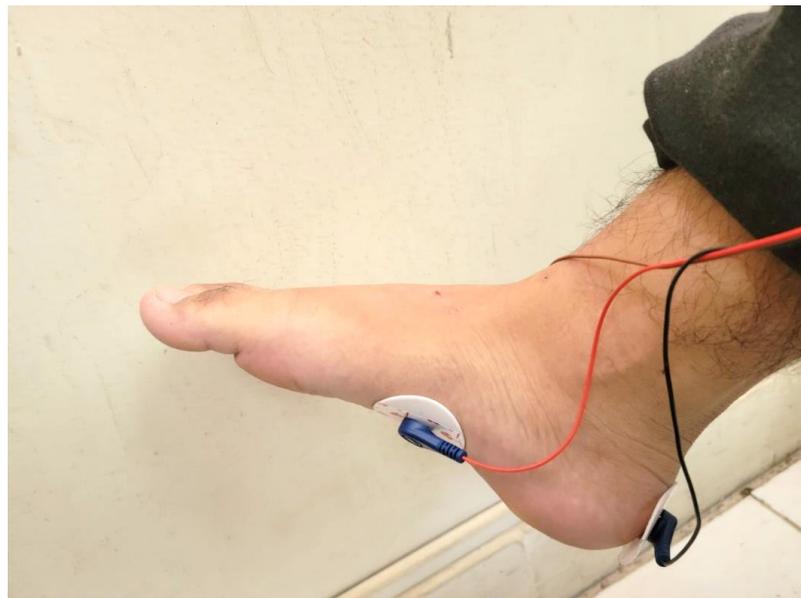


Figura 3.15 Mejor localización con sensibilidad eléctrica - Arco del pie

Cuando se colocó el electrodo adhesivo en el arco del pie, algunos sujetos de las pruebas experimentales tenían un cierto grado de pie plano, por lo tanto, su zona del arco es más propensa a ser menos sensible de lo esperado, lo cual resultó en un decrecimiento a la sensación de cosquilleo otorgada por medio de los estímulos eléctricos.

Los resultados finales que se obtuvieron a partir de los sujetos de pruebas pueden ser resumidos en el envío de la estimulación a los electrodos cuando estos comienzan a tener un episodio de CDM, el cual debe ser detectado por la IA con más de 5 casos repetidos para que omita los fallos en las predicciones del modelo, finalizando así en la situación donde los sujetos sienten el estímulo y reinician la marcha de las extremidades inferiores.

3.5 Características técnicas del dispositivo

El dispositivo de estimulación eléctrica con inteligencia artificial para personas con congelamiento de la marcha (CDM) es un sistema avanzado diseñado para proporcionar una solución efectiva para el CDM. A continuación, se describen algunas de las características técnicas clave del dispositivo:

- **Frecuencia de muestreo:** El dispositivo mide las detenciones con una frecuencia de muestreo de 23 Hz, lo que garantiza una precisión alta y un seguimiento preciso de las detenciones.
- **Voltaje de estimulación eléctrica:** El dispositivo está diseñado para proporcionar estimulación eléctrica con un voltaje máximo de 50V y regulación de baja frecuencia.
- **Algoritmo de inteligencia artificial:** El dispositivo utiliza un algoritmo de inteligencia artificial avanzado que permite identificar con precisión las detenciones voluntarias e involuntarias de los sujetos con CDM.
- **Precisión:** El dispositivo logró una precisión del 89,72% en el entrenamiento y un 86,12% en los datos de testeo.

3.6 Visualización de uso del dispositivo

En esta sección se observa una visión completa del uso del dispositivo para la estimulación eléctrica con IA. Este dispositivo está diseñado para ayudar a las personas que padecen CDM. Para lograr esto, el dispositivo está compuesto de diferentes componentes que trabajan juntos. Primero, tenemos el sensor MPU-6050 ubicado en cada uno de los laterales de los pies, que es responsable de registrar los movimientos de los sujetos. Además, se encuentran los electrodos situados en la planta del pie, que proporcionan los estímulos eléctricos necesarios para mejorar el movimiento.

Por otro lado, el contenedor del sistema eléctrico y el sistema de adquisición de datos se encuentra amarrado en la cintura de los sujetos, con un cable que sale de él y conecta con los sensores y electrodos en cada pie. Todo esto está diseñado para brindar una solución efectiva para el CDM, al mismo tiempo que se mantiene un diseño cómodo y portable para los sujetos, en la Figura 3.16 es posible observar un prototipado de una persona usando el dispositivo.

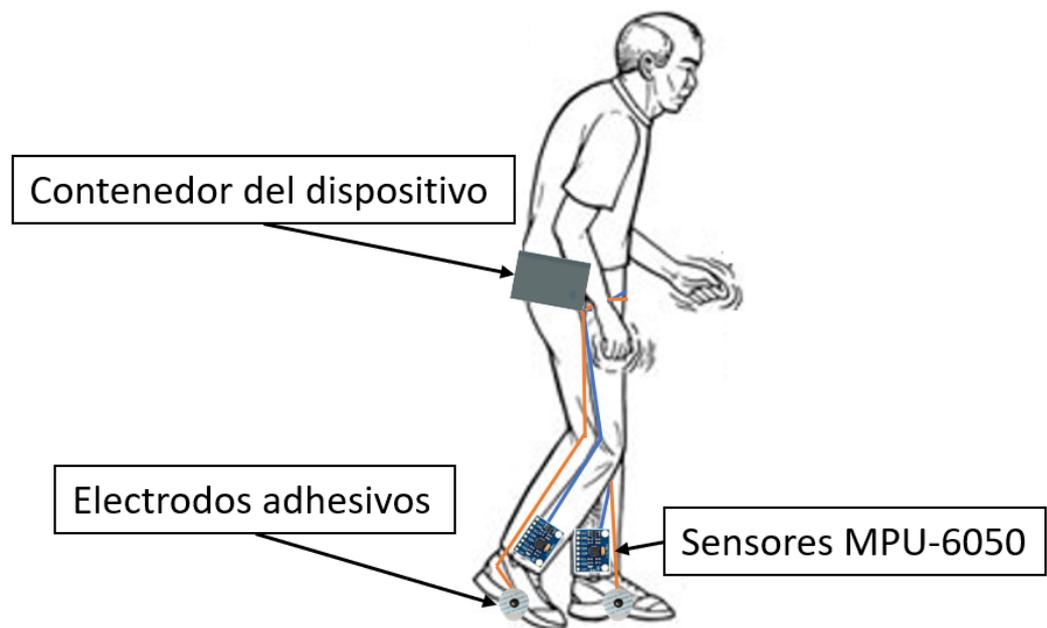


Figura 3.16 Visualización de uso para el dispositivo y sus componentes

3.7 Análisis de costos

En la Tabla 3.1 se detallan los costos totales considerando los sistemas de adquisición de datos y estimulación eléctrica, en su totalidad. Es importante considerar que el proyecto ha sido implementado con productos de fácil acceso en el mercado local, logrando así replicar el proyecto para proyectos a futuro.

Tabla 3.1 Costos de componentes para la elaboración del sistema

Sistema	Descripción	Cantidad	Costo unitario (USD)	Costo total (USD)
DAQ / Estimulador	Arduino Nano 33 IoT	1	\$35,00	\$35,00
	Sensor inercial MPU-6050	2	\$4,50	\$9,00
	Batería 9V	1	\$5,00	\$5,00
	Diodo Led	2	\$0,15	\$0,30

	Caja protectora	1	\$15,00	\$15,00
	Correas de amarre	2	\$2,00	\$4,00
DAQ	Módulo Bluetooth HC-05	1	\$7,50	\$7,50
	Shield Micro SD	1	\$2,50	\$2,50
	Tarjeta Micro SD 4Gb	1	\$5,00	\$5,00
	Switch	1	\$1,00	\$1,00
	Protoboard	2	\$2,50	\$5,00
Estimulador	CI 555	2	\$0,50	\$1,00
	Resistencias	7	\$0,05	\$0,35
	Capacitores	5	\$0,25	\$1,25
	Diodos	4	\$0,10	\$0,40
	Transistor BC547	2	\$0,25	\$0,50
	Transistor IRFZ44E	1	\$1,00	\$1,00
	Inductor toroidal 0.5 mH	2	\$1,00	\$2,00
	Electrodos adhesivos	4	\$0,50	\$2,00
Total				\$97,80

Los costos para la elaboración del dispositivo fueron divididos en secciones que involucraron a los sistemas individuales y en conjunto, lo cual nos permitió conocer las características respectivas para cada sección como el precio, cantidad de elementos, facilidad de obtención, entre otros.

Para el cálculo de los costos totales finales se tomaron en cuenta varios aspectos que participaron en la elaboración del dispositivo, los cuales son descritos como el costo de la elaboración del sistema previamente calculado, costo del transporte y el costo perteneciente a la mano de obra empleada, esto se observa en la Tabla 3.2.

Tabla 3.2 Costos totales finales

Descripción de costos	Costo (USD)
Costo de elaboración del sistema	\$97,80
Costo de transporte	\$25,00
Costo de mano de obra	\$1.500,00
Total	\$1.622,80

El costo total final fue calculado a partir de la suma de los \$97,80 empleados en la elaboración del sistema y los \$25 que fueron necesarios para el transporte de los elementos y componentes desde otra provincia dentro del país. La mano de obra empleada tomo un tiempo de 100 horas de trabajo, con un aproximado de \$15 por hora, dando un total de \$1500. Como resultado se tuvo que la suma total de todos estos costos dé un valor final de \$1.622,80.

Para estimar la rentabilidad del proyecto se definió una cantidad de productos vendidos por mes de 5 unidades con un precio de \$150 cada uno, dando como resultado que el valor del VPN (Valor Presente Neto) sea mayor a cero con \$8749.08 y el TIR (Tasa Interna de Retorno) sea más grande que el TIO (Tasa de Interés de Oportunidad) como se puede apreciar en la comparación de $22.74\% > 10\%$ (Apéndice H), de forma que se considera al proyecto como una inversión rentable y que se espera que genere un rendimiento superior al de la inversión alternativa con un nivel de riesgo similar.

En la actualidad, el mercado local no presenta dispositivos de estimulación eléctrica para combatir el CDM en la enfermedad de Parkinson como esta descrito en el presente proyecto, por lo cual este trabajo posee más ventajas económicas y de fácil adquisición que las opciones existentes internacionales, las cuales tendrían un costo más elevado y necesitarían de un tiempo de envío con sus respectivos impuestos.

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- El DAQ permitió la recolección y almacenamiento de valores para las variables de aceleración y velocidad angular obtenidos durante los experimentos en 20 sujetos de prueba, además del uso de aplicación móvil diseñada para el etiquetado de datos a través de la comunicación con el módulo Bluetooth, por tanto, el dispositivo es apto y altamente sensible para registrar datos de movimientos de los pies de pacientes para identificar episodios de CDM.
- La comparación de alternativas con respecto a sus costos y eficiencia determinó al tipo de estimulación eléctrica como bastante funcional con un bajo coste energético, el diseño del circuito eléctrico en la placa PCB tiene una salida de ondas de pulso con un valor de hasta 50V para generar la sensación de cosquilleo que reactiva la función motora, a partir de una señal de activación otorgada por parte del código dentro del microcontrolador Arduino Nano 33 IoT.
- El modelo MLP para la detección de congelamiento de la marcha fue ajustándose a diferentes valores como: la ventana de datos, número de capas, cantidad de neuronas, y parámetros de compilación, hasta obtener los mejores resultados tanto para el entrenamiento y el testeo, llegando hasta una exactitud del 89.72% y 86.12% respectivamente. El pequeño margen de error es controlado al momento de realizar las predicciones, generando así un sistema controlable que permita ser más preciso para la generación de estímulos eléctricos en los pacientes.
- Las predicciones en tiempo real con la implementación del algoritmo de inteligencia artificial en el microcontrolador mostraron las probabilidades resultantes de cada clase, correspondiente a CDM y no CDM, en donde se establecieron los valores adecuados para la activación del estimulador

dependiendo de la predicción obtenida. De esta forma, se redujo el error que impida al paciente recibir los estímulos cuando sea requerido.

- Para el costo de componentes en la elaboración del sistema de adquisición de datos y estimulación eléctrica se tuvo un valor total de \$97.80, sin embargo, para el precio de venta al público está definido como \$150, obteniendo aproximadamente un 50% de ganancia a comparación de la venta, además de considerar al dispositivo como una inversión rentable debido al buen desempeño proveniente de los requisitos respectivos al TIR y el VPN.
- La viabilidad de este proyecto es grande a futuro, puesto que el diseño involucra diferentes elementos e ideas relacionadas al campo de productos autónomos, así mismo involucra el sistema de salud en donde no solo se enfocaría en personas con enfermedad de Parkinson, sino también a otros tipos de desórdenes de movimiento, en donde se puedan recolectar datos, realizar análisis, y entrenarlos para generar estimulaciones que sean un apoyo en el caminar de los pacientes.

4.2 Recomendaciones

- Se recomienda añadir un sistema de retroalimentación en el circuito por consecuencia a la variación en la resistencia del cuerpo que posee cada persona, debido a que este valor influye en la amplitud de la onda resultante del voltaje que es enviado al cuerpo y con el cual se generaría la estimulación eléctrica, en caso de que este valor baje el paciente no logrará sentir en su totalidad al estímulo que recorre por sus extremidades inferiores.
- Es recomendable cambiar el tipo de almacenamiento con el cual se está trabajando, debido a que con un componente físico como la tarjeta MicroSD posee un rango limitado de memoria además del espacio físico empleado en el circuito dentro del contenedor, lo cual puede ser cambiado por un almacenamiento en la nube donde mayor cantidad de datos puedan ser guardados y se facilite su manipulación para el análisis y entramiento del algoritmo.

- Con el fin de obtener una mayor precisión en el modelo de IA, se sugiere utilizar datos adquiridos a partir de pacientes con CDM en la enfermedad de Parkinson, además de incrementar la cantidad de muestras empleadas para el entrenamiento del algoritmo con la finalidad de obtener más datos con los cuales predecir correctamente los resultados.
- En el estudio de los pacientes con Parkinson, es importante recopilar datos en diferentes entornos y en diferentes formas de caminar para obtener una visión más completa de su condición y progreso. Se recomienda incluir tareas como caminar en un entorno con obstáculos, detenerse, sentarse y levantarse de una silla, subir y bajar escalones, y caminar en diferentes superficies. Estos diferentes entornos y tareas ayudarán a evaluar la capacidad de los pacientes de realizar actividades cotidianas y a monitorear su progreso para la adquisición de datos.
- La implementación del sistema en una PCB puede ser una mejora importante en términos de tamaño y eficiencia, por ende, se puede optimizar el tamaño de la DAQ y hacer que sea mucho más compacto que una protoboard, además de reducir errores e interferencias eléctricas. Sin embargo, es importante considerar que el proceso de diseño y fabricación de una PCB puede ser más complejo, requerir más tiempo y recursos que construir el sistema en una protoboard.
- Se utiliza un MLP para el entrenamiento debido a limitaciones de memoria del hardware actual. Sin embargo, algoritmos como LSTM pueden ofrecer mejores resultados en la identificación de patrones en los datos. Si en el futuro se dispone de un hardware más potente, se puede considerar la implementación de una LSTM para mejorar la precisión del algoritmo de IA, siempre que el hardware lo permita, evaluando cuidadosamente las limitaciones de memoria y procesamiento previo a su uso.

BIBLIOGRAFÍA

- [1] J. Parkinson, «An essay on the shaking palsy,» London, 1817.
- [2] Organización Mundial de la Salud, «Enfermedad de Parkinson,» 13 Junio 2022. [En línea]. Available: <https://www.who.int/es/news-room/fact-sheets/detail/parkinson-disease>
- [3] Giladi N, McDermott MP, Fahn S, Przedborski S, Jankovic J, Stern M, Tanner C; Parkinson Study Group. Freezing of gait in PD: prospective assessment in the DATATOP cohort. *Neurology*. 2001 Jun 26;56(12):1712-21. doi: 10.1212/wnl.56.12.1712. PMID: 11425939.
- [4] American Parkinson Disease Association, «Freezing of Gait in Parkinson's disease,» [En línea]. Available: <https://www.apdaparkinson.org/article/freezing-gait-and-parkinsons-disease/>
- [5] Nexstride, «How Nexstride Works,» [En línea]. Available: <https://www.getnexstride.com/what-is-nexstride>
- [6] Rosenthal L, Sweeney D, Cunnington AL, Quinlan LR, ÓLaighin G. Sensory Electrical Stimulation Cueing May Reduce Freezing of Gait Episodes in Parkinson's Disease. *J Healthc Eng*. 2018 Aug 1;2018:4684925. doi: 10.1155/2018/4684925. PMID: 30154990; PMCID: PMC6092973.
- [7] J. P. M. Herdoíza, P. S. M. Perero, L. E. A. Toala, E. R. I. Mercado y D. V. Moreira-Vera, «Prevalencia de la Enfermedad de Parkinson: Estudio Puerta-Puerta en la Provincia de Manabí-Ecuador.,» Manabí, 2017.
- [8] Giladi N, Treves TA, Simon ES, Shabtai H, Orlov Y, Kandinov B, Paleacu D, Korczyn AD. Freezing of gait in patients with advanced Parkinson's disease. *J Neural Transm (Vienna)*. 2001;108(1):53-61. doi: 10.1007/s007020170096. PMID: 11261746.
- [9] Lamberti P, Armenise S, Castaldo V, de Mari M, Iliceto G, Tronci P, Serlenga L. Freezing gait in Parkinson's disease. *Eur Neurol*. 1997;38(4):297-301. doi: 10.1159/000113398. PMID: 9434089.
- [10] Bloem BR, Hausdorff JM, Visser JE, Giladi N. Falls and freezing of gait in Parkinson's disease: a review of two interconnected, episodic phenomena. *Mov Disord*. 2004 Aug;19(8):871-84. doi: 10.1002/mds.20115. PMID: 15300651.
- [11] JOHNELL, O., MELTON, L. J., ATKINSON, E. J., O'FALLON, W. M., & KURLAND, L. T. (1992). Fracture Risk in Patients with Parkinsonism: A Population-based

- Study in Olmsted County, Minnesota. *Age and Ageing*, 21(1), 32–38. doi:10.1093/ageing/21.1.32
- [12] Bennett, D. A., Beckett, L. A., Murray, A. M., Shannon, K. M., Goetz, C. G., Pilgrim, D. M., & Evans, D. A. (1996). Prevalence of Parkinsonian Signs and Associated Mortality in a Community Population of Older People. *New England Journal of Medicine*, 334(2), 71–76. doi:10.1056/nejm199601113340202
- [13] Gao, C., Liu, J., Tan, Y. et al. Freezing of gait in Parkinson's disease: pathophysiology, risk factors and treatments. *Transl Neurodegener* 9, 12 (2020). <https://doi.org/10.1186/s40035-020-00191-5>
- [14] Bakker M, Esselink RA, Renooij J, Limousin-Dowsey P, Speelman JD, Bloem BR. Effects of stereotactic neurosurgery on postural instability and gait in Parkinson's disease. *Mov Disord* 2004 (in press).
- [15] Stolze H, Klebe S, Poepping M, Lorenz D, Herzog J, Hamel W, Schrader B, Raethjen J, Wenzelburger R, Mehdorn HM, Deuschl G, Krack P. Effects of bilateral subthalamic nucleus stimulation on parkinsonian gait. *Neurology*. 2001 Jul 10;57(1):144-6. doi: 10.1212/wnl.57.1.144. PMID: 11445647.
- [16] Speelman JD. Parkinson's disease and stereotaxic neurosurgery. Amsterdam: Thesis; 1991.
- [17] Arduino, «Arduino Store,» [En línea]. Available: <https://store-usa.arduino.cc/products/arduino-uno-rev3>
- [18] Adafruit, «Adafruit Shop,» [En línea]. Available: <https://learn.adafruit.com/adafruit-lsm9ds1-accelerometer-plus-gyro-plus-magnetometer-9-dof-breakout>
- [19] GONDOLA, «GONDOLA®: Much more than a medical device, it is an effective treatment aid.,» [En línea]. Available: <https://www.gondola-medical.com/gondola-much-more-than-a-medical-device-it-is-an-effective-treatment-aid/>.
- [20] G. A. D. Vaca, R. Santibáñez y F. Loayza, «Diseño, construcción y evaluación de dispositivos de estimulación periférica en extremidades inferiores, para ayuda a pacientes de Parkinson con problemas de congelamiento de la marcha,» Guayaquil, 2014.
- [21] M. Huerta, B. Barzallo, C. Punin, A. Garcia-Cedeño y R. Clotet, «Review of Active Extracorporeal Medical Devices to Counteract Freezing of Gait in Patients with Parkinson Disease,» Cuenca, 2022.
- [22] Ferrarin, M., Brambilla, M., Garavello, L. et al. Microprocessor-controlled optical stimulating device to improve the gait of patients with Parkinson's disease. *Med. Biol. Eng. Comput.* 42, 328–332 (2004). <https://doi.org/10.1007/BF02344707>

- [23] Arias, P., & Cudeiro, J. (2010). Effect of Rhythmic Auditory Stimulation on Gait in Parkinsonian Patients with and without Freezing of Gait. PLoS ONE, 5(3), e9675. doi: 10.1371/journal.pone.0009675
- [24] Ashoori A, Eagleman DM, Jankovic J. Effects of Auditory Rhythm and Music on Gait Disturbances in Parkinson's Disease. Frontiers in Neurology. 2015;6:234. DOI: 10.3389/fneur.2015.00234. PMID: 26617566; PMCID: PMC4641247.
- [25] ARDUINO, «Arduino Nano 33 IoT,» [En línea]. Available: <https://store-usa.arduino.cc/products/arduino-nano-33-iot>
- [26] IvenSense, «MPU-6000 and MPU-6050 Product Specification Revision 3.4,» 2013.
- [27] Naylamp Mechatronics, «MÓDULO BLUETOOTH HC05,» [En línea]. Available: <https://naylampmechatronics.com/inalambrico/43-modulo-bluetooth-hc05.html>
- [28] IARDUINO, «Micro SD Card Shield,» [En línea]. Available: <https://iarduino.ru/shop/Expansion-payments/wemos-micro-sd-card-shield.html>
- [29] Naylamp Mechatronics, «TUTORIAL MPU6050, ACELERÓMETRO Y GIROSCOPIO,» 2016. [En línea]. Available: https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html
- [30] MIT APP INVENTOR, «Hour of Code with MIT App Inventor,» [En línea]. Available: <http://appinventor.mit.edu/explore/hour-of-code>
- [31] C. A. Phillips, «Functional Electrical Stimulation and Lower Extremity Bracing for Ambulation Exercise of the Spinal Cord Injured Individual: A Medically Prescribed System,» 1989.
- [32] Fraunhofer IMS, «AlfES for Arduino,» 12 Enero 2022. [En línea]. Available: https://github.com/Fraunhofer-IMS/AlfES_for_Arduino

APÉNDICES

APÉNDICE A

Evaluación del peso específico de cada criterio por medio del método ordinal corregido de criterios ponderados

Viabilidad = Funcionalidad > Comodidad = Precio > Facilidad de uso > Innovación

Tabla A.1 Cálculos de los pesos específicos por criterio

Criterio	Comodidad	Precio	Funcionalidad	Viabilidad	Innovación	Facilidad de uso	$\Sigma+$	Ponderación
Comodidad		0.5	0	0	1	1	3.5	0.1667
Precio	0.5		0	0	1	1	3.5	0.1667
Funcionalidad	1	1		0.5	1	1	5.5	0.2619
Viabilidad	1	1	0.5		1	1	5.5	0.2619
Innovación	0	0	0	0		0	1	0.0476
Facilidad de uso	0	0	0	0	1		2	0.0952
						Suma	21	1

Evaluación del peso específico del criterio Comodidad

Alternativa 1 > Alternativa 2 > Alternativa 3

Tabla A.2 Evaluando el criterio de comodidad en las alternativas

Comodidad	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1		1	1	3	0.5
Alternativa 2	0		1	2	0.333
Alternativa 3	0	0		1	0.167
			Suma	6	1

Evaluación del peso específico del criterio Precio

Alternativa 2 > Alternativa 1 > Alternativa 3

Tabla A.3 Evaluando el criterio de precio en las alternativas

Precio	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1		0	1	2	0.333

Alternativa 2	1		1	3	0.5
Alternativa 3	0	0		1	0.167
			Suma	6	1

Evaluación del peso específico del criterio Funcionalidad

Alternativa 3 > Alternativa 1 > Alternativa 2

Tabla A.4 Evaluando el criterio de funcionalidad en las alternativas

Funcionalidad	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1		1	0	2	0.333
Alternativa 2	0		0	1	0.167
Alternativa 3	1	1		3	0.5
			Suma	6	1

Evaluación del peso específico del criterio Viabilidad

Alternativa 2 > Alternativa 3 > Alternativa 1

Tabla A.5 Evaluando el criterio de viabilidad en las alternativas

Viabilidad	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1		0	0	1	0.167
Alternativa 2	1		1	3	0.5
Alternativa 3	0	1		2	0.333
			Suma	6	1

Evaluación del peso específico del criterio Innovación

Alternativa 2 > Alternativa 1 > Alternativa 3

Tabla A.6 Evaluando el criterio de innovación en las alternativas

Innovación	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1		0	1	2	0.333
Alternativa 2	1		1	3	0.5
Alternativa 3	0	0		1	0.167
			Suma	6	1

Evaluación del peso específico del criterio Facilidad de uso

Alternativa 2 > Alternativa 1 > Alternativa 3

Tabla A.7 Evaluando el criterio de facilidad de uso en las alternativas

Facilidad de uso	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1		0	1	2	0.333
Alternativa 2	1		1	3	0.5
Alternativa 3	0	0		1	0.167
			Suma	6	1

APÉNDICE B

Algoritmo para el sistema de adquisición de datos

```
#include "I2Cdev.h" // Libreria que permite establecer la conexion I2C
#include "Wire.h" // Libreria que ayuda a establecer el tipo de conexion que poseen los pines en el arduino
#include <SPI.h> // Libreria que permite establecer la conexion SPI
#include <SD.h> // Libreria para utilizar el MicroSD
#include <SoftwareSerial.h> // Libreria que permite establecer pines digitales para la comunicacion serial
```

Figura B.1 Librerías utilizadas en el DAQ

```
SoftwareSerial miBT(5, 6); // pin 5 como RX, pin 6 como TX
char DATO='5';
int LEDROJO = 9;
int LEDVERDE = 8;
char CASO='2';
bool entrar=false;
unsigned long tiempoInicio;
const int chipSelect = 10;
String nombre;
//Variables utilizadas para la lectura de los sensores MPU-6050
const int MPU2 = 0x69, MPU1=0x68;
long accelX, accelY, accelZ;
float gyroX, gyroY, gyroZ;
float gForceX1, gForceY1, gForceZ1, rotX1, rotY1, rotZ1;
float gForceX2, gForceY2, gForceZ2, rotX2, rotY2, rotZ2;
```

Figura B.2 Variables locales para el manejo de valores

```
String obtenerNombre() {
    int incremento=0;
    String nombre="subj_"+String(incremento)+".txt";
    while(SD.exists(nombre)) {
        incremento++;
        nombre="subj_"+String(incremento)+".txt";
    }
    return nombre;
}
```

Figura B.3 Función encargada de otorgar el siguiente nombre al archivo almacenado

```

void GetMpuValue(const int MPU) {
    Wire.beginTransaction(MPU);
    Wire.write(0x3B);
    Wire.endTransmission();
    Wire.requestFrom(MPU, 6);
    while(Wire.available() < 6);
    accelX = Wire.read() << 8 | Wire.read();
    accelY = Wire.read() << 8 | Wire.read();
    accelZ = Wire.read() << 8 | Wire.read();

    Wire.beginTransaction(MPU);
    Wire.write(0x43);
    Wire.endTransmission();
    Wire.requestFrom(MPU, 6);
    while(Wire.available() < 6);
    gyroX = Wire.read() << 8 | Wire.read();
    gyroY = Wire.read() << 8 | Wire.read();
    gyroZ = Wire.read() << 8 | Wire.read();

    if(MPU==0x68) {
        gForceX1 = accelX * (9.81/16384.0);
        gForceY1 = accelY * (9.81/16384.0);
        gForceZ1 = accelZ * (9.81/16384.0);
        rotX1 = gyroX / 131.0;
        rotY1 = gyroY / 131.0;
        rotZ1 = gyroZ / 131.0;
    }
    if(MPU==0x69) {
        gForceX2 = accelX * (9.81/16384.0);
        gForceY2 = accelY * (9.81/16384.0);
        gForceZ2 = accelZ * (9.81/16384.0);
        rotX2 = gyroX / 131.0;
        rotY2 = gyroY / 131.0;
        rotZ2 = gyroZ / 131.0;
    }
}

```

Figura B.4 Función que obtiene los valores de los dos MPU-6050

```

void setup() {

    miBT.begin(38400);
    pinMode(LEDROJO, OUTPUT);
    pinMode(LEDVERDE, OUTPUT);

    SD.begin(chipSelect);
    nombre=obtenerNombre();

    Wire.begin();
    Wire.beginTransmission(MPU1);
    Wire.write(0x6B);
    Wire.write(0b00000000);
    Wire.endTransmission();
    Wire.beginTransmission(MPU1);
    Wire.write(0x1B);
    Wire.write(0x00000000);
    Wire.endTransmission();
    Wire.beginTransmission(MPU1);
    Wire.write(0x1C);
    Wire.write(0b00000000);
    Wire.endTransmission();

    Wire.begin();
    Wire.beginTransmission(MPU2);
    Wire.write(0x6B);
    Wire.write(0b00000000);
    Wire.endTransmission();
    Wire.beginTransmission(MPU2);
    Wire.write(0x1B);
    Wire.write(0x00000000);
    Wire.endTransmission();
    Wire.beginTransmission(MPU2);
    Wire.write(0x1C);
    Wire.write(0b00000000);
    Wire.endTransmission();
    Serial.begin(38400);

}

```

Figura B.5 Función setup que establece las conexiones y condiciones iniciales en los componentes del sistema

```

void loop() {
  if (entrar){
    GetMpuValue (MPU1);
    GetMpuValue (MPU2);
    unsigned long tiempoActual=millis()-tiempoInicio;
    File myFile=SD.open(nombre, FILE_WRITE);
    if (miBT.available()){
      DATO = miBT.read();

      if( DATO == '1' ){
        digitalWrite(LEDROJO, HIGH);
        digitalWrite(LEDVERDE, LOW);
        CASO='1';
      }
      if( DATO == '2' ){
        digitalWrite(LEDVERDE, HIGH);
        digitalWrite(LEDROJO, LOW);
        CASO='2';
      }
      if( DATO == '3' ){
        digitalWrite(LEDVERDE, LOW);
        digitalWrite(LEDROJO, LOW);
        CASO='0';
      }
      if (DATO=='5'){
        entrar=false;
        digitalWrite(LEDVERDE, LOW);
        digitalWrite(LEDROJO, LOW);
        miBT.print(nombre);
      }
    }
    myFile.print(String(tiempoActual)+","+String(gForceX1)+","+String(gForceY1)+","+String(gForceZ1)+","+String(rotX1)+","+String(rotY1)+","+String(rotZ1)+",");
    myFile.println(String(gForceX2)+","+String(gForceY2)+","+String(gForceZ2)+","+String(rotX2)+","+String(rotY2)+","+String(rotZ2)+","+CASO);
    myFile.close();
  }
}

```

Figura B.6 Primera parte de la función loop que adquiere, etiqueta y almacena datos

```

else{
  if (miBT.available()) {
    DATO = miBT.read();
    if(DATO=='4') {
      entrar=true;
      tiempoInicio=millis()+10;
    }
  }
}
delay(10);
}

```

Figura B.6 Segunda parte de la función loop que inicia el DAQ

APÉNDICE C

Consentimiento Informado

Yo _____ declaro que he sido informado e invitado a participar en una prueba experimental referente al congelamiento de marcha en la enfermedad de Parkinson, éste es un proyecto de titulación de estudiantes pertenecientes a la Escuela Superior Politécnica del Litoral. Entiendo que este estudio busca recolectar datos con respecto a la simulación del movimiento de una persona con congelamiento de marcha. Me han explicado que la información registrada será confidencial, y que los nombres de los participantes serán asociados a un número de serie, esto significa que las respuestas no podrán ser conocidas por otras personas. Estoy en conocimiento que los datos no me serán entregados y que no habrá retribución por la participación en este estudio, sin embargo, esta información podrá beneficiar de manera indirecta y por lo tanto tiene un beneficio para la sociedad dada la investigación que se está llevando a cabo. Asimismo, sé que puedo negar la participación o retirarme en cualquier etapa de la investigación, sin expresión de causa ni consecuencias negativas para mí.

Sí. Acepto voluntariamente participar en este estudio y he recibido una copia del presente documento.

Fecha:

Firma participante:

Figura C.1 Formato del consentimiento informado

APÉNDICE D

Preprocesamiento del set de entrenamiento y prueba

La cantidad total de muestras se dividió en un 80% entrenamiento, y un 20% prueba, obteniendo como resultado (14882,14) en entrenamiento, y (3720,14) en prueba; siguiendo el formato (*#filas, #columnas*). Luego que se dividió los datos, se escalaron los datos en rango de -1 a 1, para que el algoritmo mantenga sus valores en una misma escala y facilite su aprendizaje.

Una vez realizada la separación, se definió la ventana de datos siendo el “time step”, este valor es $T\ step = 10$, el desplazamiento de los datos se observa en la Figura D.1. Debido a la forma en cómo se crea la ventana, las últimas 10 filas del set de entrenamiento no se podrán seguir desplazando por lo que no existirá una salida correspondiente, por lo tanto, esos 10 últimos valores se repetirán al inicio del set de pruebas para usar toda la información y no descartar ninguna muestra.

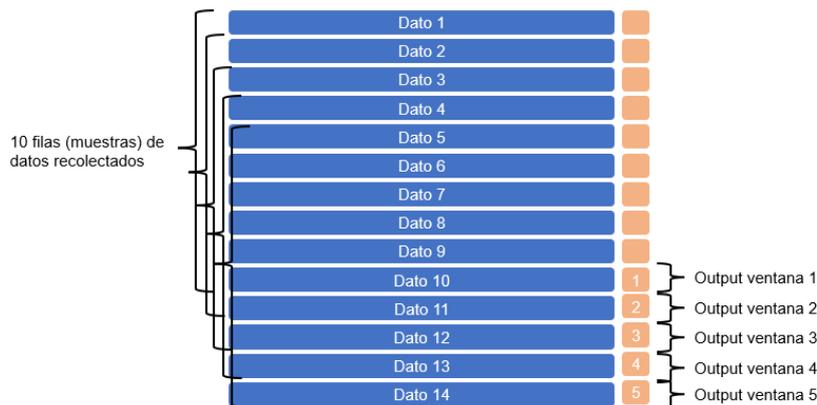


Figura D.1 Estructura de entrada y salida de datos con una ventana de 10 filas

Por último, los datos fueron aplanados (Flatten) a un arreglo de 1 dimensión, esto debido a que en el despliegue del algoritmo al Arduino Nano 33 IoT se procuró no exceder la memoria, además de facilitar el procesamiento de forma más efectiva sin llegar a las limitaciones de RAM. La cantidad de datos considerando las filas y columnas al tomar 10 muestras x 12 atributos quedaría:

- $x_{train} = (14872, 120)$
- $y_{train} = (14872,)$
- $x_{test} = (3710, 120)$
- $y_{test} = (3710,)$

APÉNDICE E

Modelo de detección para el congelamiento de la marcha

```
import pandas as pd
from pathlib import Path

# Lectura de todos los archivos de muestreo
df_fog = [pd.read_csv(file, header=None)
          for file in Path('/content/data').iterdir()]

df_fog = pd.concat(df_fog)

# Definir nombre de columnas en el DataFrame
nombres=['Tiempo',
        'Ax_pi', 'Ay_pi', 'Az_pi',
        'Vx_pi', 'Vy_pi', 'Vz_pi',
        'Ax_pd', 'Ay_pd', 'Az_pd',
        'Vx_pd', 'Vy_pd', 'Vz_pd',
        'Etiqueta']

df_fog.columns=nombres

# Conversión de valores para definir sus respectivas unidades
df_fog.loc[:, 'Ax_pi': 'Az_pi'] = df_fog.loc[:, 'Ax_pi': 'Az_pi'] / (2*9.81)
df_fog.loc[:, 'Ax_pd': 'Az_pd'] = df_fog.loc[:, 'Ax_pd': 'Az_pd'] / (2*9.81)
df_fog.loc[:, 'Vx_pi': 'Vz_pi'] = df_fog.loc[:, 'Vx_pi': 'Vz_pi'] / (150)
df_fog.loc[:, 'Vx_pd': 'Vz_pd'] = df_fog.loc[:, 'Vx_pd': 'Vz_pd'] / (150)

# Transformación a 2 etiquetas: Cdm (1) y No Cdm (0)
df_fog['Etiqueta'] = df_fog['Etiqueta'].replace(2,0)
df_fog.reset_index(drop=True, inplace=True)

# Transformación del tiempo a índices
# una vez obtenida la información de frecuencia
df_notime = df_fog.iloc[:,1:]
df_notime = df_notime.rename_axis('Tiempo').reset_index()
```

Figura E.1 Lectura y conversión de datos recolectados

```
import seaborn as sns
import matplotlib.pyplot as plt

def graph_features(dataframe_name, foot, label='Etiqueta'):
    fig,axs = plt.subplots(6,1,figsize=(15,15))
    _ = sns.lineplot(data=dataframe_name, x="Tiempo", y = "Ax_"+foot, ax = axs[0], color = "b", hue = label)
    _ = sns.lineplot(data=dataframe_name, x="Tiempo", y = "Ay_"+foot, ax = axs[1], color = "r", hue = label)
    _ = sns.lineplot(data=dataframe_name, x="Tiempo", y = "Az_"+foot, ax = axs[2], color = "g", hue = label)
    _ = sns.lineplot(data=dataframe_name, x="Tiempo", y = "Vx_"+foot, ax = axs[3], color = "y", hue = label)
    _ = sns.lineplot(data=dataframe_name, x="Tiempo", y = "Vy_"+foot, ax = axs[4], color = "c", hue = label)
    _ = sns.lineplot(data=dataframe_name, x="Tiempo", y = "Vz_"+foot, ax = axs[5], color = "m", hue = label)
```

Figura E.2 Función de visualización de señales de aceleración y velocidad para cada pie

```

# Porcentaje de separación para entrenamiento y testeo
train_total = round(len(df_notime) * 0.80)
test_total = round(len(df_notime) * 0.20)

# Separación de datos
train = df_notime.iloc[:train_total]
test = df_notime.iloc[train_total:]

# Ventana (Cantidad de filas) de datos
time_step = 10

# Modificación de testeo por la reducción de la ventana
last_train = train.iloc[-(time_step-1):]
test = pd.concat([last_train, test], axis=0)

# Separación de input y output de datos
train_x = train.iloc[:,1:-1]
train_y = train.iloc[:, -1]
test_x = test.iloc[:,1:-1]
test_y = test.iloc[time_step-1:,-1]

# Escalamiento de datos
from sklearn.preprocessing import MinMaxScaler

sc = MinMaxScaler(feature_range=(-1,1))
train_x_sc = sc.fit_transform(train_x)
test_x_sc = sc.transform(test_x)

```

Figura E.3 Separación y escalado de conjunto de datos para entrenamiento y testeo

```

import numpy as np

# Training dataset
x_train = []
y_train = []

for i in range(time_step, train_y.shape[0]):
    x_train.append(train_x_sc[i-time_step:i]) #Ej.Para el primer dato [0,10)
    y_train.append(train_y[i-1]) #Ej.Salida es la etiqueta 0 o 1

x_train, y_train = np.array(x_train), np.array(y_train)

# Testing dataset
x_test = []
y_test = []

for i in range(time_step, test_y.shape[0]):
    x_test.append(test_x_sc[i-time_step:i]) #Ej.Para el primer dato [0,10)
    y_test.append(test_y.values[i-1]) #Ej.Salida es la etiqueta 0 o 1

x_test, y_test = np.array(x_test), np.array(y_test)

```

Figura E.4 Ajuste de entrenamiento y testeo considerando la ventana de datos

```

from sklearn.utils import class_weight

class_weights = class_weight.compute_class_weight(class_weight='balanced',
                                                  classes=np.unique(y_train),
                                                  y=y_train)

class_weights = {i:class_weights[i] for i in range(2)}

```

Figura E.5 Cálculo de pesos para cada clase de CDM y No CDM

```

from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import ReduceLRonPlateau, EarlyStopping, ModelCheckpoint

from time import time

INPUT = x_train_flatten.shape[1]
LAYERS = [8, 8, 8, 1]
LR = 1e-1

keras.backend.clear_session()

model = Sequential()

model.add(Dense(input_dim=INPUT, units=LAYERS[0], activation='sigmoid'))
model.add(Dense(units=LAYERS[1], activation='sigmoid'))
model.add(Dense(units=LAYERS[2], activation='sigmoid'))
model.add(Dense(units=LAYERS[3], activation='sigmoid'))

model.compile(loss='binary_crossentropy',
              metrics=['accuracy'],
              optimizer=Adam(learning_rate=LR))

```

Figura E.6 Creación de modelo MLP

```

# Definir un método de disminución de la tasa de aprendizaje:
lr_decay = ReduceLRonPlateau(monitor='loss',
                             patience=1, verbose=1,
                             factor=0.2, min_lr=1e-4)

# Definir parada anticipada:
early_stop = EarlyStopping(monitor='val_accuracy', min_delta=0,
                           patience=12, verbose=1, mode='auto',
                           baseline=0, restore_best_weights=True)

# Guardar mejor modelo
saving_model = ModelCheckpoint("model_fog.h5", save_best_only=True)

```

Figura E.7 Funciones callbacks para controlar el comportamiento del modelo durante el entrenamiento

```
BATCH = 10
EPOCH = 100

start = time()

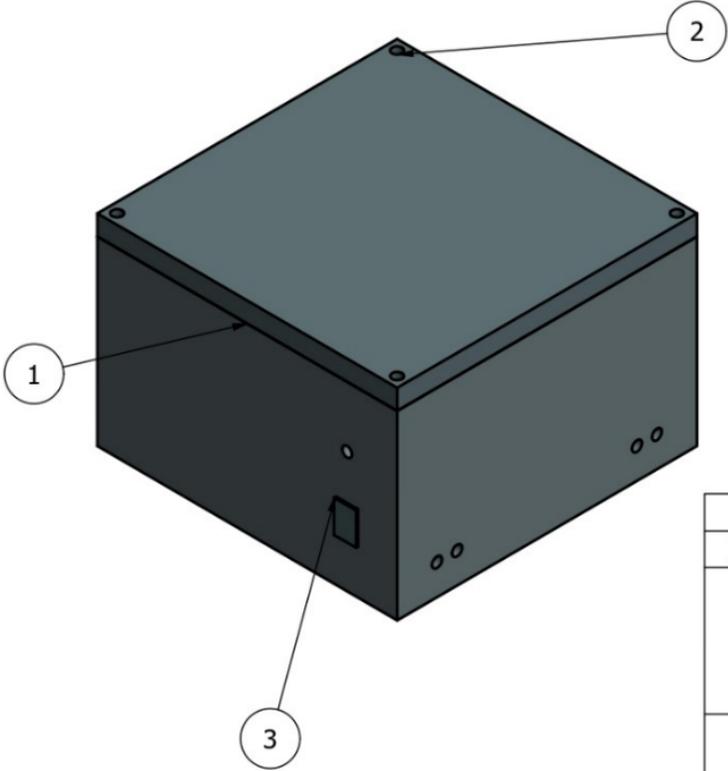
history = model.fit(
    x=x_train_flatten,
    y=y_train,
    epochs=EPOCH,
    batch_size=BATCH,
    class_weight=class_weights,
    validation_data=(x_test_flatten, y_test),
    validation_batch_size=BATCH,
    callbacks=[lr_decay, early_stop, saving_model]
)

print('-'*65)
print(f'Training was completed in {time() - start:.2f} secs')
print('-'*65)
```

Figura E.8 Entrenamiento del modelo MLP

APÉNDICE F

Planos mecánicos del contenedor



1

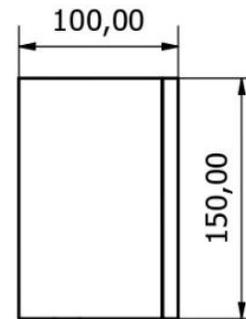
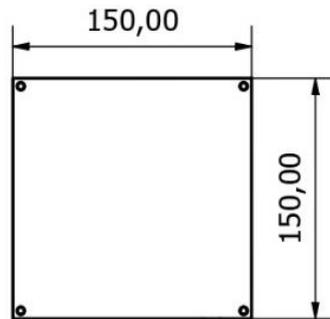
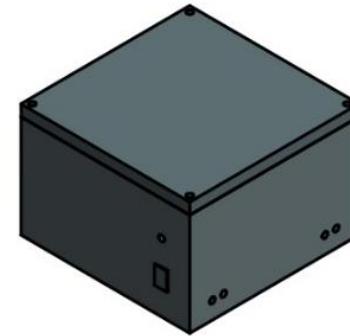
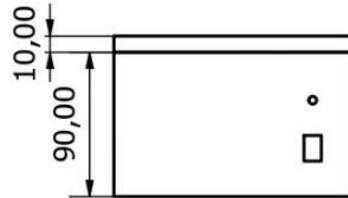
2

3

Nota: Todas las medidas están en milímetros

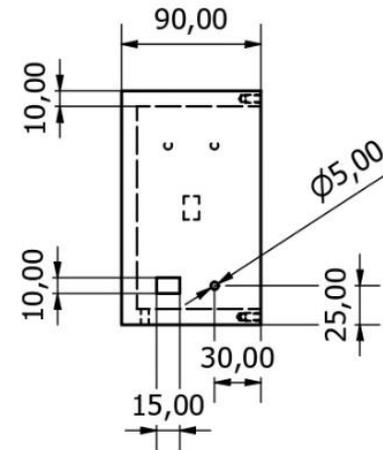
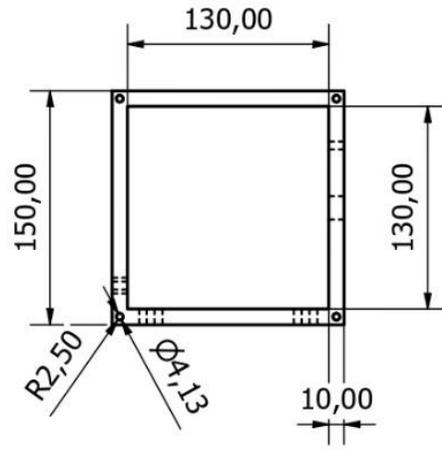
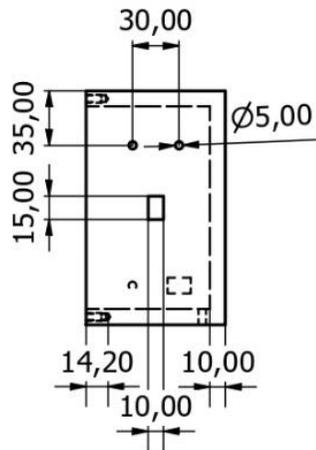
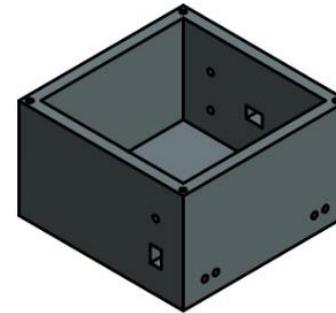
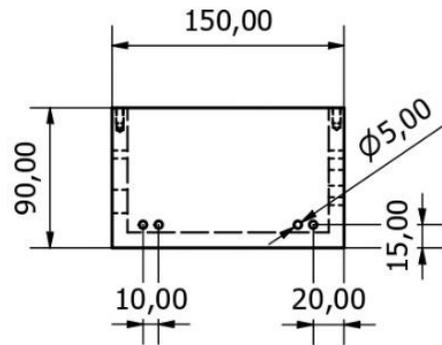
LISTA DE PIEZAS			
ELEMENTO	CTDAD	Nº DE PIEZA	DESCRIPCIÓN
1	1	Caja	Lugar donde se almacenaran los sistemas y componentes
3	1	Tapon	Tapon para el puerto USB
2	1	Tapa	Tapa que protege y guarda al contenedor

Proyecto:	Contenedor del estimulador eléctrico		
Cliente:	Juan Carlos Sola		
Diseño	20-01-23	R. Cedeño	Descripción: Piezas del contenedor
Diseño	20-01-23	J. Parraga	
Revisión No.	Fecha	Nombre	
FACULTAD DE INGENIERÍA MECÁNICA Y CIENCIAS DE LA PRODUCCIÓN		ISO E	Revisor por
Plano No.: HX-001			Hoja No.
Fecha:	20-01-23	Escala: 1:2	REV: 00
			1/5



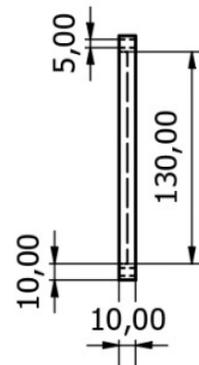
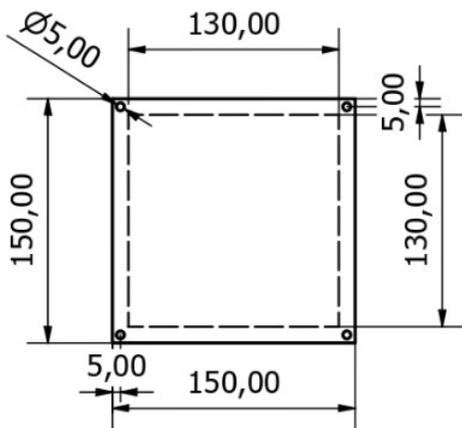
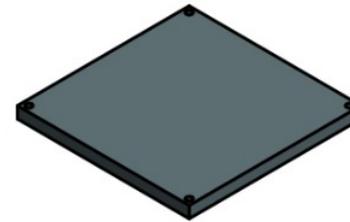
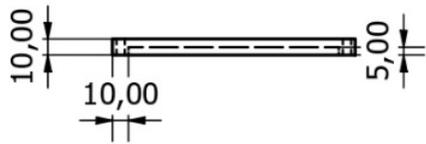
Nota: Todas las medidas están en milímetros

			Proyecto: Contenedor del estimulador eléctrico		
Diseño	20-01-23	R. Cedeño	Cliente: Juan Carlos Sola		
Diseño	20-01-23	J. Parraga	Descripción: Contenedor		
Revisión No.	Fecha	Nombre			
FACULTAD DE INGENIERÍA MECÁNICA Y CIENCIAS DE LA PRODUCCIÓN			ISO E	Revisor por	Hoja No. 2/5
			Plano No.: HX-001		
Fecha:	20-01-23	Escala: 1:4	REV: 00		



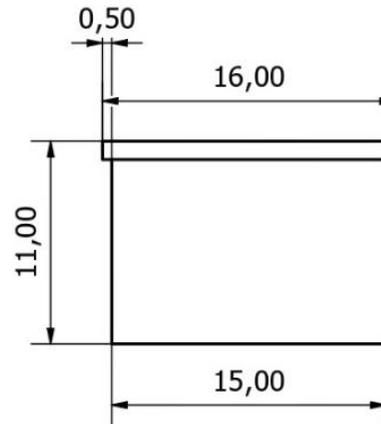
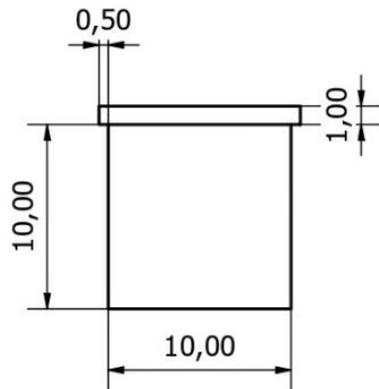
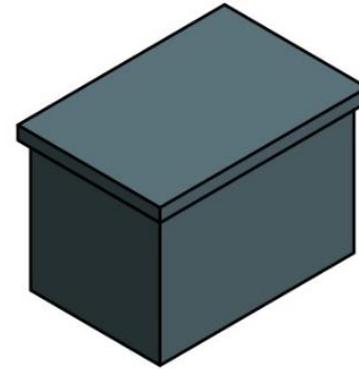
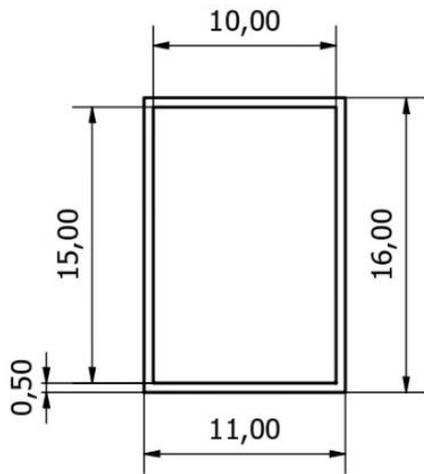
Nota: Todas las medidas están en milímetros

			Proyecto: Contenedor del estimulador eléctrico	
Diseño	20-01-23	R. Cedeño	Cliente: Juan Carlos Sola	
Diseño	20-01-23	J. Parraga	Descripción: Caja	
Revisión No.	Fecha	Nombre	ISO E	Revisor por
FACULTAD DE INGENIERÍA MECÁNICA Y CIENCIAS DE LA PRODUCCIÓN			Plano No.: HX-001	Hoja No.
			Fecha: 20-01-23	3/5
			Escala: 1:4	REV: 00



Nota: Todas las medidas están en milímetros

Diseño		20-01-23	R. Cedeño	Proyecto: Contenedor del estimulador eléctrico		
Diseño		20-01-23	J. Parraga	Cliente: Juan Carlos Sola		
Revisión No.	Fecha	Nombre		Descripción: Tapa		
FACULTAD DE INGENIERÍA MECÁNICA Y CIENCIAS DE LA PRODUCCIÓN				ISO E	Revisor por	Hoja No.
				Plano No.: HX-001		4/5
Fecha:	20-01-23	Escala:	1:4	REV:	00	



Nota: Todas las medidas están en milímetros

			Proyecto: Contenedor del estimulador eléctrico	
Diseño			20-01-23	R. Cedeño
Diseño			20-01-23	J. Parraga
Revisión No.	Fecha	Nombre	Descripción: Tapon	
FACULTAD DE INGENIERÍA MECÁNICA Y CIENCIAS DE LA PRODUCCIÓN			ISO E	Revisor por
			Plano No.: HX-001	Hoja No.
			Fecha: 20-01-23	5/5
			Escala: 3:1	REV: 00

APÉNDICE G

Código de estimulación implementado en el Arduino Nano 33 IoT

```
// *****  
// Librerías  
// *****  
#include "I2Cdev.h"  
#include "Wire.h"  
#include <aifes.h>  
  
#define DATASETS 1  
#define FNN_5_LAYERS 5  
  
int LEDVERDE = 8;  
int PREDICCIONES = 0;  
bool entrar = true;  
  
// *****  
// Variables para los 2 MPU  
// *****  
const int MPU2 = 0x69, MPU1=0x68;  
long accelX, accelY, accelZ;  
float gyroX, gyroY, gyroZ;  
float gForceX1, gForceY1, gForceZ1, rotX1, rotY1, rotZ1;  
float gForceX2, gForceY2, gForceZ2, rotX2, rotY2, rotZ2;
```

Figura G.1 Definición de librerías y variables

```
void setup() {  
  // CONFIGURACIÓN MPU 1  
  Wire.begin();  
  Wire.beginTransmission(MPU1);  
  Wire.write(0x6B);  
  Wire.write(0b00000000);  
  Wire.endTransmission();  
  
  Wire.beginTransmission(MPU1);  
  Wire.write(0x1B);  
  Wire.write(0x00000000);  
  Wire.endTransmission();  
  
  Wire.beginTransmission(MPU1);  
  Wire.write(0x1C);  
  Wire.write(0b00000000);  
  Wire.endTransmission();  
  
  // CONFIGURACIÓN MPU 2  
  Wire.begin();  
  Wire.beginTransmission(MPU2);  
  Wire.write(0x6B);  
  Wire.write(0b00000000);  
  Wire.endTransmission();  
  
  Wire.beginTransmission(MPU2);  
  Wire.write(0x1B);  
  Wire.write(0x00000000);  
  Wire.endTransmission();  
  
  Wire.beginTransmission(MPU2);  
  Wire.write(0x1C);  
  Wire.write(0b00000000);  
  Wire.endTransmission();  
  
  // SERIAL  
  Serial.begin(115200);  
  Serial.println();  
  Serial.println("Type any character");  
  Serial.println();  
  delay(100);  
  
  // SALIDA SIMULANDO ESTIMULADOR ELECTRICO  
  pinMode(LEDVERDE, OUTPUT);  
}
```

Figura G.2 Configuración de los sensores, serial para pruebas y señalización

```

// ***** CODIGO DE MODELO IA *****

// ***** Describir la red neuronal feed-forward *****
uint32_t FNN_structure[FNN_5_LAYERS] = {120,8,8,8,1};

// Seleccione las funciones de activación para la capa densa
AIFES_E_activations FNN_activations[FNN_5_LAYERS - 1];
FNN_activations[0] = AIFES_E_sigmoid; // Sigmoid for hidden (dense) layer
FNN_activations[1] = AIFES_E_sigmoid; // Sigmoid for hidden (dense) layer
FNN_activations[2] = AIFES_E_sigmoid; // Sigmoid for hidden (dense) layer
FNN_activations[3] = AIFES_E_sigmoid; // Sigmoid for output (dense) layer

// Función AIFES Express: calcula el número de pesos necesarios
uint32_t weight_number = AIFES_E_flat_weights_number_fnn_f32(FNN_structure, FNN_5_LAYERS);
int8_t error = 0;

// Matriz FlatWeights
const float FlatWeights[] = {4.070353984832764f, -1.6548107862472534f, 3.9981675148010254f, 6.05906581878662
-0.45166832208633423f, 4.16862678527832f, 0.29081398248672485f, -1.6776549816131592f, -5.637535572052002f,
0.15700256824493408f, -2.1969780921936035f, 1.078542947769165f, 1.5954407453536987f, 0.9736132621765137f,
3.5874743461608887f, 6.80825138092041f, -1.5007905960083008f, -1.8941442966461182f, 6.9867472648620605f,
-10.415395736694336f, 3.6140737533569336f, -4.72143030166626f, -14.904873847961426f, -7.3622965812683105f,
-3.886683464050293f, -18.966094970703125f, 3.3162808418273926f, -15.878082275390625f, -7.568625450134277f,
-16.636003494262695f, -5.5246453285217285f, -3.9801371097564697f, -8.137457847595215f, 2.3958771228790283f,
10.114715576171875f, -7.705780506134033f, -0.4020645022392273f, -0.1336019486188885f, 2.360858917236328f,

if(weight_number != sizeof(FlatWeights)/sizeof(float)){
    Serial.println(F("Error: number of weights wrong!"));
    return;
}

// Llenar la estructura AIFES Express
AIFES_E_model_parameter_fnn_f32 FNN;
FNN.layer_count = FNN_5_LAYERS;
FNN.fnn_structure = FNN_structure;
FNN.fnn_activations = FNN_activations;
FNN.flat_weights = (void*)FlatWeights;

// ----- Crear los tensores -----
uint16_t input_shape[] = {DATASETS, (uint16_t)FNN_structure[0]};
aitensor_t input_tensor = AITENSOR_2D_F32(input_shape, input_data);

float output_data[DATASETS];
uint16_t output_shape[] = {DATASETS, (uint16_t)FNN_structure[4]};
aitensor_t output_tensor = AITENSOR_2D_F32(output_shape, output_data);

// ----- Hacer la inferencia -----
// Función AIFES Express: Hace la inferencia
error = AIFES_E_inference_fnn_f32(&input_tensor, &FNN, &output_tensor);
error_handling_inference(error);

```

Figura G.4 Secciones importantes en la definición del modelo implementando la librería AIFES

```

// ----- Imprimir los resultados (Solo en pruebas) -----
// Imprimiendo la predicción limitada por la probabilidad obtenida
Serial.print("Prediction: ");
if (output_data[0] < 0.3){
    Serial.println("0");
}else{
    Serial.println("1");
}

// Conteo de multiples predicciones para controlar los falsos positivos
if (output_data[0] < 0.3){
    PREDICCIONES = 0;
    digitalWrite(LEDVERDE, LOW);
}
else{
    PREDICCIONES++;
}

// Definiendo límite de predicciones previo a la estimulación
if (PREDICCIONES > 3){
    digitalWrite(LEDVERDE, HIGH);
    delay(500);
}

```

Figura G.5 Impresión de resultados en puerto serial para observación de predicciones

```

void GetMpuValue(const int MPU) {
    Wire.beginTransmission(MPU);
    Wire.write(0x3B);
    Wire.endTransmission();
    Wire.requestFrom(MPU, 6);
    while(Wire.available() < 6);
    accelX = Wire.read() << 8 | Wire.read();
    accelY = Wire.read() << 8 | Wire.read();
    accelZ = Wire.read() << 8 | Wire.read();

    Wire.beginTransmission(MPU);
    Wire.write(0x43);
    Wire.endTransmission();
    Wire.requestFrom(MPU, 6);
    while(Wire.available() < 6);
    gyroX = Wire.read() << 8 | Wire.read();
    gyroY = Wire.read() << 8 | Wire.read();
    gyroZ = Wire.read() << 8 | Wire.read();

    if(MPU==0x68) {
        gForceX1 = accelX * (9.81/16384.0);
        gForceY1 = accelY * (9.81/16384.0);
        gForceZ1 = accelZ * (9.81/16384.0);
        rotX1 = gyroX / 131.0;
        rotY1 = gyroY / 131.0;
        rotZ1 = gyroZ / 131.0;
    }

    if(MPU==0x69) {
        gForceX2 = accelX * (9.81/16384.0);
        gForceY2 = accelY * (9.81/16384.0);
        gForceZ2 = accelZ * (9.81/16384.0);
        rotX2 = gyroX / 131.0;
        rotY2 = gyroY / 131.0;
        rotZ2 = gyroZ / 131.0;
    }
}

```

Figura G.6 Función de obtención de valores para cada sensor inercial

APÉNDICE H

Estimación de la rentabilidad del proyecto con las variables TIR y VPN

Investment value	\$25.368				
Rescue value	\$6.000				
Useful life	5	years			
Taxes	15%				
Annual profit	\$9.000				
TIO	10%	investor - board member (oportunnity interest rate)			
Depreciation	\$3.874				
Sale of the asset	0				

Figura H.1 Datos iniciales para el cálculo de la rentabilidad

years	0	1	2	3	4	5
cash flows	\$-25.368	\$9.000	\$9.000	\$9.000	\$9.000	\$9.000
VP	\$-25.368	\$8.182	\$7.438	\$6.762	\$6.147	\$5.588
VPN	\$8.749,08	Profitable (Valor presente neto)				
TIR	22,74%	Competitive because TIR (tasa interna de retorno) is greater than TIO (Internal rate of return)				
VPN excel	\$8.749,08					
Analisis de hipotesis						

Figura H.2 Resultados obtenidos para el TIR y el VPN