



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Dpto. de Ingeniería de Sistemas y Automática

Dispositivo empotrado para medir características de palas
de padel

Trabajo Fin de Máster

Máster Universitario en Automática e Informática Industrial

AUTOR/A: Bujedo Saiz, Santiago

Tutor/a: Blanes Campos, Carlos

Cotutor/a: Correcher Salvador, Antonio

CURSO ACADÉMICO: 2021/2022



“DISPOSITIVO EMPOTRADO PARA MEDIR CARACTERÍSTICAS DE PALAS DE PÁDEL”

**MÁSTER EN AUTOMÁTICA E INFORMÁTICA INDUSTRIAL
INSTITUTO DE AUTOMÁTICA E INFORMÁTICA INDUSTRIAL**

AUTOR:

Santiago Bujedo Saiz

TUTORES:

Carlos Blanes Campos

Antonio Correcher Salvador

CURSO 2021/2022

Septiembre 2022

RESUMEN DEL TRABAJO FIN DE MÁSTER

Título: Dispositivo empotrado para medir características de palas de Pádel.

Autor: Santiago Bujedo Saiz

Tutor (es): Carlos Blanes Campos, Antonio Correcher Salvador

RESUMEN

La gran diversidad de modelos de palas de pádel existentes en el mercado y la enorme variedad de materiales unidos a los métodos de fabricación poco industrializados hacen que sea una tarea difícil conocer las características de las palas tanto para el jugador como para el fabricante.

Este Proyecto Fin de Máster tiene como objetivo la creación de un dispositivo empotrado que permitirá medir diferentes parámetros de jugabilidad derivados de cada golpeo, como pueden ser velocidades, aceleraciones y energía.

Con el prototipo se realizarán ensayos en laboratorio y en pista, con el objetivo de correlacionar estos datos para así obtener patrones y datos de caracterización de los diferentes golpes.

Para llevarlo a cabo se utilizará un dispositivo de control unido a unos sensores, los cuales se encargarán de la adquisición de datos para luego ser enviados a una aplicación de visualización, donde posteriormente serán tratados.

ABSTRACT

The great variety of padel racket models on the market and the enormous variety of materials in addition to the manufacturing methods that are not very industrialised make it a difficult task to know the characteristics of the racket for both the player and the manufacturer.

The aim of this master's degree project is to create an embedded device that will measure different playability parameters derived from each stroke, such as velocities, accelerations and energy.

The prototype will be tested in the laboratory and on the paddle court, with the aim of correlating this data in order to obtain patterns and characterisation data for the different strokes.

To carry it out, a control device will be used together with sensors, which will be in charge of acquiring the data and then, sending it to a visualisation application, where it will later be processed.

RESUM

La gran diversitat de models de pales de pàdel existents en el mercat i l'enorme varietat de materials units als mètodes de fabricació poc industrialitzats fan que siga una tasca difícil conèixer les característiques de les pales tant per al jugador com per al fabricant.

Aquest Projecte Fi de Màster té com a objectiu la creació d'un dispositiu encastrat que permetrà mesurar diferents paràmetres de jugabilitat derivats de cada colpege, com poden ser velocitats, acceleracions i energia.

Amb el prototip es realitzaran assajos en laboratori i en pista, amb l'objectiu de correlacionar aquestes dades per a així obtindre patrons i dades de caracterització dels diferents colps.

Per a dur-ho a terme s'utilitzarà un dispositiu de control unit a uns sensors, els quals s'encarregaran de l'adquisició de dades per a després ser enviats a una aplicació de visualització, on posteriorment seran tractats.

MEMORIA

1. INTRODUCCIÓN Y OBJETIVOS	12
1.1 Antecedentes del proyecto	12
1.1.1 Wearables en el pádel.....	14
1.2 Objetivo del proyecto.....	15
2. ESTADO DEL ARTE.....	16
2.1 Dispositivos integrados a la pala.....	16
2.1.1 Sony Smart Tennis sensor.....	16
2.1.2 Zepp Tennis 2 Swing and Math Analyzer	17
2.1.3 Head Tennis Sensor	18
2.1.4 BABOLAT Play.....	19
2.1.5 Artengo tennis coach	20
2.2 Dispositivos Wearables	20
2.2.1 BABOLAT POP	20
2.2.2 Otros dispositivos wearables.....	21
3. ESTUDIO DE NECESIDADES DE MONITORIZACIÓN.....	22
3.1 Tamaño y ubicación en la pala.....	22
3.1.1 Estudio de ubicación	22
3.1.2 Estudio de dimensiones	24
3.1.2 Definición de las magnitudes a medir	25
4. DESCRIPCIÓN Y JUSTIFICACIÓN DE LA SOLUCIÓN ADOPTADA	26
4.1 Elección del Hardware.....	26
4.1.1 Dispositivos de control.....	26
4.1.2 Micro controlador elegido.....	28
4.1.3 Dispositivos de adquisición de datos	29
4.1.3.1 Ensayos para la selección del acelerómetro	29
4.1.3.2 Elección de los acelerómetros	29
4.1.4 Dispositivos de alimentación.....	31
4.1.4.1 Batería elegida.....	32
4.2 Elección del software	32
4.2.1 Software de programación del micro controlador.....	32
4.2.1.1 ESP-IDF	33
4.2.1.2 IDE de Arduino	33
4.2.1.3 Elección del entorno de programación del micro controlador	33
4.2.2 Software de recepción, tratamiento y visualización de los datos.....	34

4.2.2.1	MATLAB	34
4.2.2.2	LabVIEW	35
4.2.2.3	Elección del programa de tratamiento de datos.....	35
4.3	Protocolos de comunicaciones	35
4.3.1	Protocolo UDP	36
4.3.2	Protocolo TCP	36
4.3.3	Protocolo ESP-NOW	36
4.3.4	Protocolo elegido	37
4.4	Funcionamiento del dispositivo	37
4.4.1	Funcionamiento del sistema de adquisición.....	37
4.4.1.1	Planteamiento inicial	37
4.4.1.2	Funcionamiento elegido	38
4.4.2	Funcionamiento de la aplicación de tratamiento de datos	38
5.	DISEÑO DEL DISPOSITIVO.....	40
5.1	Decisiones de diseño.....	40
5.2	Necesidades de diseño.....	41
5.2.1	Conexión del Acelerómetro Analógico.....	41
5.2.1.1	Voltaje de entrada del ADC del ESP32S2	41
5.2.1.2	Voltaje de salida del acelerómetro analógico.....	42
5.3	Esquema eléctrico	45
5.4	Primer Prototipo	45
5.5	Segundo Prototipo	45
5.6	Dispositivo Final	47
6.	PROGRAMACIÓN DE LOS MICROCONTROLADORES	49
6.1	Programación del dispositivo de adquisición.....	49
6.1.1	Diagrama de flujo	49
6.1.2	Explicación del programa	50
6.2	Programación del dispositivo receptor	53
6.2.1	Explicación del programa	53
7.	APLICACIÓN PARA EL TRATAMIENTO Y VISUALIZACIÓN DE LOS DATOS.....	55
7.1	Programa de recepción y almacenamiento de datos en ficheros.....	55
7.2	Programa para la visualización y tratamiento de los datos	57
7.2.1	Parámetros a visualizar	57
7.2.1.1	Escalado de la señal analógica	58

7.2.2	Tratamiento de datos	59
8.	ENSAYOS Y MEDIDAS REALIZADAS	63
8.1	Tipos de ensayos realizados	63
8.1.1	Ensayo en estación de salida de bola	63
8.1.2	Ensayo en pared	64
8.2	Palas utilizadas en los ensayos	65
8.3	Metodología de realización de cada ensayo	67
8.3.1	Estación de salida de bola	67
8.3.2	Golpeo en pared	70
9.	ANÁLISIS DE RESULTADOS	72
9.1	Estudio y correlación de diferentes palas	73
9.1.1	Resultados de los ensayos de la IMU y del acelerómetro analógico en la estación de salida de bola	74
9.1.2	Resultados de los ensayos solo con el acelerómetro analógico en la estación de salida de bola	77
9.1.3	Resultados de los ensayos realizados en pared	79
9.2	Estudio del tipo de golpe	79
9.2.1	Gráficas golpes	80
9.2.2	Conclusiones tipo de golpe	89
10.	MEJORAS Y TRABAJOS FUTUROS	92
10.1	Hardware	92
10.2	Comunicaciones	92
10.3	Software	92
11.	CONCLUSIONES	94
12.	REFERENCIAS	96
	ANEXO Nº 1 PRESUPUESTOS	98
1.1	Justificación de los precios	98
1.1.1	Costes	98
1.1.1.1	Costes Directos	98
1.1.1.2	Costes Indirectos	102
1.2	MEDICIONES	102
1.3	CUADROS DE PRECIOS	103
1.3.1	Cuadro de precios Nº1	103
1.3.2	Cuadro de precios Nº2	104
1.3.2.1	Unidad de obra Nº1 Diseño y montaje del dispositivo	104

1.3.2.2	Unidad de obra N°2 Programación del dispositivo	105
1.3.2.3	Unidad de obra N°3 Programación de la aplicación.....	106
1.3.2.4	Unidad de obra N°4 Ensayos y recogida de datos	106
1.3.2.5	Unidad de obra N°5 Redacción del proyecto y estudio de los datos	107
1.3.2.6	Unidad de obra N°6 Control de calidad, con prueba de funcionamiento ..	108
1.4	PRESUPUESTOS	108
1.4.1	Presupuesto de Ejecución Material.....	109
1.4.2	Presupuesto de Ejecución por Contrata.....	109
ANEXO N°2 CÓDIGO DE LOS MICRO CONTROLADORES.....		110
2.1	Código del dispositivo de adquisición, Acelerómetro analógico e IMU.....	110
2.2	Código del dispositivo de adquisición solo acelerómetro analógico	115
2.3	Código del dispositivo receptor Acelerómetro analógico e IMU	119
2.4	Código del dispositivo receptor solo acelerómetro analógico.....	121

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Evolución de las pistas de Pádel en Europa.	13
Ilustración 2: Crecimiento de Clubes de Pádel en Europa.	13
Ilustración 3: Ranking europeo en función del número de pistas de Pádel por país.	13
Ilustración 4: Comparativa micro controladores[9], [10].....	27
Ilustración 5: Módulo esp32s2 lipo olimex	28
Ilustración 6: Batería Lipo.....	32
Ilustración 7: Logotipo Arduino.....	33
Ilustración 8: Logotipo MATLAB	34
Ilustración 9: Logotipo LabVIEW.....	35
Ilustración 10: Módulo esp32 con el Led y botón señalizados	40
Ilustración 11: Tabla Voltaje esp32s2	41
Ilustración 12: Voltaje de salida a 0g	42
Ilustración 13: Sensibilidad frente a voltaje de alimentación.....	42
Ilustración 14: Divisor de Tensión	44
Ilustración 15: Esquema eléctrico.....	45
Ilustración 16: Conexiones y soldaduras segundo prototipo	46
Ilustración 17: Vista desde arriba segundo prototipo	46
Ilustración 18: Placa de sensores segundo prototipo	46
Ilustración 19: Soldaduras y conexiones segundo prototipo	46
Ilustración 20: Esquema PCB.....	47
Ilustración 21: PCB final	47
Ilustración 22: Prototipo final vista inferior	48
Ilustración 23: Prototipo final vista superior	48
Ilustración 24: Prototipo final en la pala	48
Ilustración 25: Flujoograma Modo de funcionamiento 1	50
Ilustración 26: Flujoograma Modo de funcionamiento 2	50
Ilustración 27: Bloque de datos modo de funcionamiento 2	54
Ilustración 28: Bloque de datos modo de funcionamiento 1	54
Ilustración 29: Parte de configuración del puerto serie	55
Ilustración 30: Parte de separar los datos y almacenarlos en un fichero	56
Ilustración 31: Pantalla de visualización	57
Ilustración 32: Conjunto total de datos.....	59
Ilustración 33: Conjunto de datos recortados	60
Ilustración 34: Gráficas de muestra de los valores	60
Ilustración 35: Espectros de las ondas	61
Ilustración 36: Valores calculados en el tratamiento de datos.....	61
Ilustración 37: Estación de salida de bola	64
Ilustración 38: Ensayo en pared con el prototipo final.....	64
Ilustración 39: Adidas Adipower Atk1.9	66

Ilustración 40: Pala Artengo PR190	66
Ilustración 41: Pala Kelme olímpica	67
Ilustración 42: Colocación de los sensores en la paña.....	67
Ilustración 43: Pala colocada en la estación de golpeo	68
Ilustración 44: Estación preparada para medir	68
Ilustración 45: Preparación de la pala para el ensayo.....	69
Ilustración 46: Preparación del golpeo	70
Ilustración 47: Golpeo ensayo en pared	71
Ilustración 48: Tabla de datos IMU Y acelerómetro analógico.....	73
Ilustración 49: Tabla de datos solo acelerómetro analógico.....	73
Ilustración 50: Datos finales pala Artengo	74
Ilustración 51: Desviaciones típicas pala Artengo	74
Ilustración 52: Datos finales pala Kelme Olímpica.....	75
Ilustración 53: Desviación típica datos pala Kelme	75
Ilustración 54: Datos finales pala Adidas	75
Ilustración 55: Desviación típica datos pala Adidas.....	75
Ilustración 56: Datos finales pala Artengo solo acelerómetro analógico	77
Ilustración 57: Desviación típica datos pala Artengo.....	77
Ilustración 58: Datos finales pala Kelme solo acelerómetro analógico.....	77
Ilustración 59: Desviación típica datos pala Kelme	78
Ilustración 60: Datos finales pala Adidas solo acelerómetro analógico	78
Ilustración 61: Desviación típica datos pala Adidas.....	78
Ilustración 62: Comparativa golpeo de derecha eje X.....	90
Ilustración 63: Comparativa golpeo de revés eje X.....	90
Ilustración 64: Comparativa golpeo de remate eje X	90

ÍNDICE DE TABLAS

Tabla 1: Tablas salariales según convenio año 2020 [17]	99
Tabla 2: Cuotas a ingresar a la seguridad social según tipo. [18]	100
Tabla 3: Porcentaje de cotización por accidentes CNAE-2009 26 [4]	100
Tabla 4: Costes horarios de la maquinaria.....	101
Tabla 1: Mediciones unidades de obra	102
Tabla 2: Cuadro de precios N° 1	103
Tabla 3: Coste de materiales Unidad de obra N°1	104
Tabla 4: Coste de mano de obra Unidad de obra N°1	104
Tabla 5: Coste de maquinaria Unidad de obra N° 1	104
Tabla 9: Coste de mano de obra Unidad de obra N°3	106
Tabla 10: Coste de maquinaria Unidad de obra N°3	106
Tabla 11: Coste de mano de obra Unidad de obra N°4	106

Tabla 12: Coste de maquinaria Unidad de obra N°4	107
Tabla 11: Coste de mano de obra Unidad de obra N°4	107
Tabla 12: Coste de maquinaria Unidad de obra N°4	107
Tabla 16: Coste de mano de obra Unidad de obra N°2	108
Tabla 17: Coste de maquinaria Unidad de obra N°2	108
Tabla 18: Presupuesto de ejecución Material	109



1. INTRODUCCIÓN Y OBJETIVOS

El presente proyecto con título “Desarrollo de un sistema empotrado de monitorización en tiempo real de impactos en palas de pádel” ha sido realizado por Santiago Bujedo Saiz, alumno del Máster en Automática e Informática Industrial de la Universidad Politécnica de Valencia.

Ha sido tutorizado por los profesores Antonio Correcher Salvador y Carlos Blanes Campos del instituto Automática e Informática Industrial, y llevándose a cabo en los laboratorios de dicho instituto, concretamente en el laboratorio de Testea Pádel.

Dicho proyecto consiste en la creación de un dispositivo que obtiene datos de los golpes de una pala de pádel para estudiar sus características, para ello se estudian diferentes parámetros de los golpes.

Para ello se ha decidido instalar dos sensores en la pala de pádel, siendo uno una unidad inercial (IMU) y el otro un acelerómetro, unidos los dos a un dispositivo de control, los cuales adquieren los datos y luego son enviados mediante un protocolo de comunicación inalámbrico a un ordenador, donde posteriormente se analizan y estudian mediante una aplicación.

1.1 Antecedentes del proyecto

En los últimos años, el pádel está teniendo un aumento en su popularidad, teniendo cada día más personas que lo practican. Un deporte que llegó a España en los años 80 y que desde ese momento no ha dejado de crecer, especialmente en los últimos años.

Según un estudio llamado “Global Padel Report” realizado por la consultora Deloitte junto con la empresa de pádel Playtomic, el pádel se ha convertido en uno de los principales deportes actualmente.

Este estudio aporta los siguientes datos:

- Desde 2016 se ha triplicado la creación de nuevas pistas en Europa, estando en torno a las veintisiete mil, más de un 20 por ciento anual de crecimiento.

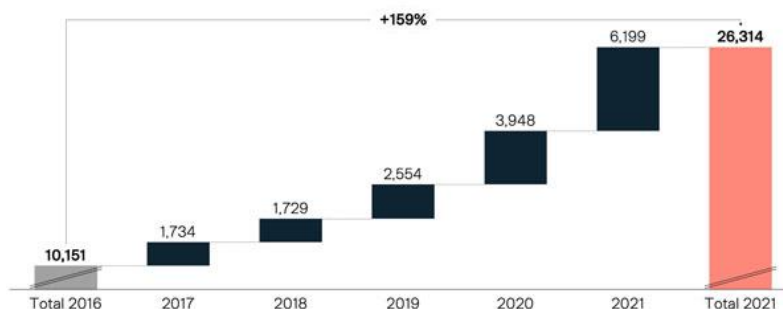


Ilustración 1: Evolución de las pistas de Pádel en Europa.

- El número de clubes en Europa ha aumentado un 181 % en los cinco últimos años, siendo España, Italia, Suecia y Francia los países que lideran el crecimiento.

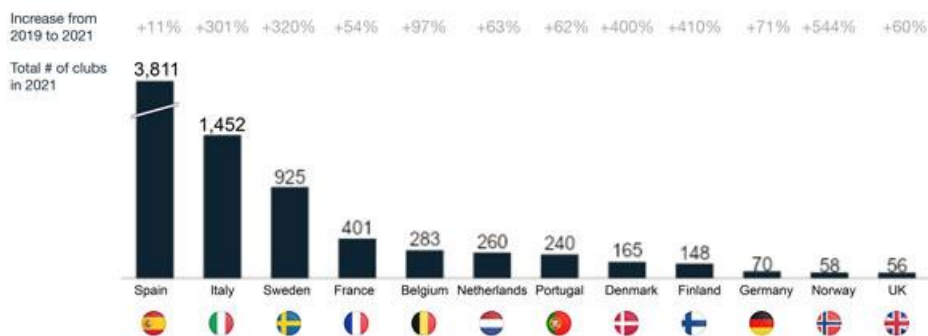


Ilustración 2: Crecimiento de Clubes de Pádel en Europa.

- España Italia Suecia Francia y Bélgica lideran el Ranking en número de pistas construidas, teniendo España catorce mil.

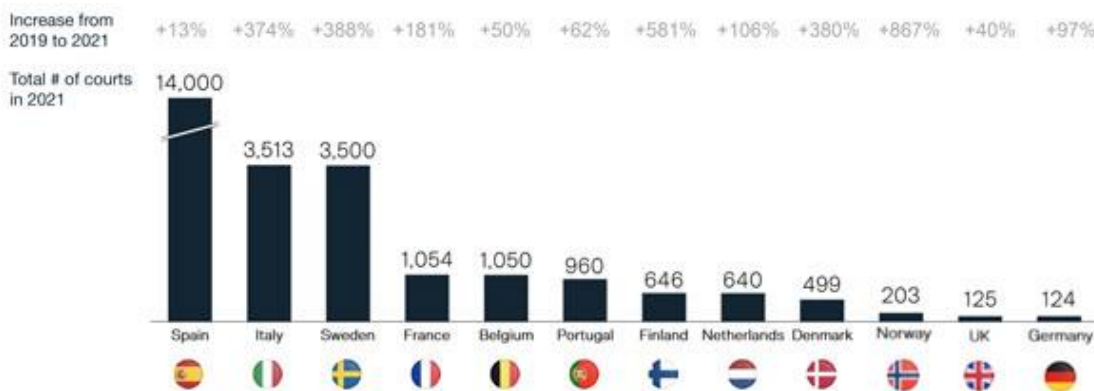


Ilustración 3: Ranking europeo en función del número de pistas de Pádel por país.



De hecho, durante los años de la pandemia (2020 y 2021) se han inaugurado unas 100 pistas y creado 29 clubes cada semana.[1]

Se pronostica que puede convertirse en uno de los deportes más practicados del mundo, llegando a superar al tenis.

1.1.1 Wearables en el pádel

Los dispositivos wearables son dispositivos electrónicos que se incorporan a nuestro cuerpo con la finalidad de realizar alguna función concreta, como por ejemplo la monitorización de una actividad.

Algunos ejemplos de dispositivos wearables son:

- Relojes o pulseras inteligentes, que monitorizan los pasos, el sueño, la temperatura...
- Gafas inteligentes
- Zapatillas inteligentes, que monitorizan pasos, fuerza y potencia al correr
- Ropa inteligente como por ejemplo chalecos que miden la actividad física.

Haciendo una búsqueda de estos dispositivos por el mercado, se ha observado que existe una gran cantidad para la mayoría de deportes, especialmente running, ciclismo, caminar...

Pero a la hora de encontrar wearables para monitorizar los deportes de raqueta, con algún dispositivo que se adhiera a la pala, se ha visto que principalmente existen dispositivos para monitorizar el tenis, puesto que es un deporte más asentado.

En el pádel existe una carencia de estos dispositivos encontrándose una pala de pádel inteligente que si que permite monitorizar el juego.

Por ello se ha planteado el presente proyecto, con el objetivo de poder crear un dispositivo intercambiable entre diferentes palas para monitorizar el juego en tiempo real.



1.2 Objetivo del proyecto

El proyecto cuenta con los siguientes objetivos:

- Crear un dispositivo del tamaño más reducido posible que se pueda adherir al mango de la pala para poder obtener datos de aceleraciones e inclinaciones a la hora del golpeo.
- Hacer una aplicación visual para poder observar estos datos y poder realizar un tratamiento de los mismos.
- Realizar ensayos con el dispositivo, en la estación de salida de bola del laboratorio de pádel ai2 y estudiar diferentes datos como frecuencia de la señal, energía en el golpe, aceleración máxima...
- Realizar ensayos con las mismas palas que en la estación, pero en pista o contra una pared, estudiar los mismos datos y buscar correlaciones entre los dos tipos de ensayos.
- Categorizar el tipo de golpe (revés, derecha, volea) en función de los datos obtenidos por la IMU.



2. ESTADO DEL ARTE

Para iniciar este proyecto fin de máster se va a realizar un estudio sobre las distintas posibilidades que existen en el mercado para monitorizar en tiempo real el juego en los deportes de raqueta.

Se han encontrado diferentes dispositivos, que se dividen en dispositivos que se acoplan a la pala y dispositivos wearables.

2.1 Dispositivos integrados a la pala

2.1.1 Sony Smart Tennis sensor

Este dispositivo se trata de un sensor que se coloca directamente en el mango de la raqueta y recolecta los siguientes datos:

- Velocidad de la bola tras el golpeo
- Zona de impacto de la pelota
- Velocidad de la raqueta
- Número de golpes dados durante la sesión

Sus características son las siguientes:

Cuenta con un peso de 8 gramos, tiene protección antigolpes, agua y suciedad contando con la certificación IP65.

La conexión del sensor con la aplicación móvil se realiza mediante bluetooth, con una duración de batería de 90 minutos utilizando el bluetooth, pudiéndose utilizar el sensor sin activar el bluetooth que almacenarán los datos de la sesión hasta que se transfieran, llegando la duración de la batería a las 3 horas.

Este dispositivo tiene diferentes modos de uso:

- Modo en vivo que permite monitorizar los datos de cada golpe.
- Modo entrenador que permite ver las estadísticas de todos los golpes de la sesión.
- Modo vídeo en el cual otra persona graba con el móvil mediante la aplicación para que a la vez que se produzcan los golpes se vea por pantalla los datos recogidos por el sensor, contando con una opción slow motion para poder corregir mejor los golpes.[2]

2.1.2 Zepp Tennis 2 Swing and Math Analyzer

Se trata de un sensor diseñado por la empresa Zepp que se coloca en la raqueta con el objetivo de analizar todos los golpes y permite crear y compartir videos con los que obtener un control del rendimiento.

Este sensor realiza un seguimiento de las métricas de rendimiento clave para cada golpeo a la pelota. Sus principales tareas son:

- Analiza el tipo de golpe (smash, volea, derecha, revés, servicio)
- Analiza el tipo de efecto
- Registra con que punto de la raqueta se golpea la pelota
- Mide la velocidad y giro de la pelota

Además, mediante la aplicación se permite grabar videos para poder capturar los diferentes momentos durante el juego para poder analizarlo.

Tiene diferentes formas de colocación en la raqueta:

- **Montaje pro:** Se pega en la parte final de la raqueta
- **Montaje Flexible:** Se puede quitar fácilmente y reutilizar
- **Insertar montaje:** Utilizado en modelos selectos de raquetas que vienen preparados

Tanto el montaje pro como el montaje flexible funcionan en cualquier tipo de raquetas.



Ilustración 4: Tipos de montaje

Sus características técnicas son:

- **Peso:** 6,25 g
- **Medidas:** 25,4mm x 25,4mm x 12,3 mm
- **Sensores:** Acelerómetros duales y giroscopios duales de 3 ejes



- **Batería:** Iones de litio recargable, ciclo de carga 1,5 h y duración 8 h.[3]
- **Conectividad:** Bluetooth LE.

2.1.3 Head Tennis Sensor

Se trata de un dispositivo similar al sensor Zepp, puesto que utiliza la misma tecnología, con la diferencia que este sensor únicamente es válido para ciertas raquetas de la marca Head, en las que queda integrado perfectamente.

Tiene la gran ventaja que no sobre sale nada de la raqueta, por lo que no modifica las características de la raqueta.

Mediante su aplicación se puede seleccionar que raqueta de las incluidas se está utilizando.

Cuenta con diferentes modos: Juego, entrenamiento, competición y servicio en los que analiza los diferentes parámetros como fuerza, velocidad, efecto...[4]



Ilustración 5:Head tennis sensor

2.1.4 BABOLAT Play

Se trata de la solución por la que la marca Babolat ha optado para monitorizar el juego, esta marca se ha decantado por hacer raquetas “inteligentes” en las que ha colocado diferentes sensores para poder analizar los golpes.

A diferencia de otros sensores que son compatibles con diferentes raquetas, la marca babolat tiene únicamente tres raquetas con las que se puede monitorizar el juego con su sistema, las cuales son:

- Babolat Play Pure Drive
- Babolat Play AeroPro Drive
- Babolat Play Pure Drive Lite

El sensor se encuentra colocado dentro de la empuñadura y cuenta con acelerómetros, giroscopios y sensores piezoeléctricos.

Los acelerómetros se encargan de determinar la dirección de la raqueta, mientras que los giróscopos determinan la rotación para analizar que golpe se está realizando.

Los sensores piezoeléctricos son los encargados de calcular donde se ha golpeado la pelota mediante el análisis de la vibración, para así poder informar del punto óptimo de golpeo.

La raqueta se conecta a la aplicación de Babolat mediante Bluetooth o utilizando un cable USB.

Por lo general Babolat ofrece la misma información que los sensores Zepp, es decir tipo de golpe, potencia, efecto, número de golpes, velocidad de servicios y localizador de impacto. Incluyendo un análisis 3D del servicio.[5], [6]



Ilustración 6: Babolat play

2.1.5 Artengo tennis coach

Se trata de un dispositivo creado por la empresa Decathlon mediante su marca artengo destinado a monitorizar el tenis. Este sensor a diferencia de los explicados anteriormente se adhiere a la raqueta en la parte que se encuentra encima del mango, justo por debajo de la zona de golpeo, se puede ver en la siguiente imagen:



Ilustración 7: Artengo tenis sensor

Este sensor calcula datos similares al resto de sensores explicados en los otros apartados, cuenta con tres modos de medición: Entrenamiento, partido y desafío.

La gran desventaja de este sensor es que cuenta con un peso total de 24 gramos que es un peso considerable para una raqueta de tenis profesional y llega a modificar el balance de pesos de la raqueta.[7]

2.2 Dispositivos Wearables

Estos son conocidos como dispositivos inteligentes que se incorporan en la vestimenta o usados en el cuerpo como accesorios para monitorizar diferentes datos.

2.2.1 BABOLAT POP

Se trata de una pulsera que se coloca en la muñeca mientras el usuario juega al tenis, tiene la ventaja que funciona con cualquier raqueta puesto que no depende de ella.

Mide las siguientes características:

- Velocidad de la raqueta en los golpes
- Número de golpes que se dan
- Efecto
- Potencia
- Tiempo de juego
- Análisis del tipo de golpe

Al ir colocado en la muñeca mide la velocidad y movimiento de la muñeca y no de la raqueta.[5], [8]



Ilustración 8: Babolat POP

2.2.2 Otros dispositivos wearables

En el mercado existen infinidad de smartwatches y pulseras de actividad que son capaces de monitorizar los deportes de raqueta, dentro de los cuales entra el pádel.

Estos dispositivos al ser genéricos miden principalmente tiempo en movimiento, pulsaciones y calorías, pero no datos específicos.



3. ESTUDIO DE NECESIDADES DE MONITORIZACIÓN

En este capítulo se va a proceder a realizar un estudio de necesidades para definir el dispositivo, para ello se deben estudiar las siguientes características:

- Tamaño: estudio de ubicación en la pala, así como sus dimensiones.
- Medidas: Definición de magnitudes a medir.
- Dispositivos de adquisición y tratamiento de datos: Estudio de alternativas y selección.

3.1 Tamaño y ubicación en la pala

Con este proyecto se busca diseñar un dispositivo que pueda ser comercial y por ello es importante que:

- Tenga un tamaño reducido.
- Tenga un peso lo más pequeño posible, para evitar interferir en la distribución de pesos de la pala.
- Tenga una ubicación interfiera lo mínimo en el juego.
- Proporcione comodidad al jugador para poder utilizarlo en el tiempo y no solo para pruebas.

3.1.1 Estudio de ubicación

El lugar donde se coloque el dispositivo va a ser un factor muy importante puesto que depende del sitio puede afectar al juego o a la distribución de pesos en la pala.

Como se vio en el estado del arte, los dispositivos que se utilizan para realizar medidas de parámetros en raquetas de tenis van situados en el mango, concretamente en la parte inferior.

Se han planteado dos posibilidades diferentes de ubicación, las cuales son las dos en el mango, una es colocar el dispositivo en el interior del mango, es decir levantando la tapa que las palas traen al final del mango y la otra acoplándolo al final del mango mediante un pegamento o similar, como los dispositivos de tenis Zepp.

En la siguiente imagen se muestra esquemáticamente donde iría colocado el dispositivo.



Ilustración 9: Primera posibilidad de colocación



Ilustración 10: Segunda posibilidad de colocación

Para ver cuál de las dos ubicaciones sería más idónea se realiza un estudio por diferentes palas existentes en el laboratorio para poder ver el interior del mango de diferentes palas, ya que la primera posibilidad comprende insertarlo dentro del mango, y las diferentes medidas.



Ilustración 11: Interior del mango de diferentes palas

Tras hacer un sondeo por diferentes palas, al abrirlas y ver el mango por dentro se aprecia que ningún mango está hueco, todos tienen una chapa o algún elemento para poder sujetar la correa que sale de la pala. Por ello se descarta la ubicación del interior del mango, puesto que dependería del tipo de pala a utilizar, y obligaría al usuario a desmontar el mango para quitar los materiales que obstruyan la colocación del dispositivo.

Por lo tanto, la ubicación elegida es en la parte inferior del mango, en la que el dispositivo se adherirá mediante silicona.

3.1.2 Estudio de dimensiones

Para diseñar un dispositivo competitivo en el mercado, hay que intentar hacerlo con las mínimas dimensiones posibles. Los mangos tienen todos aproximadamente las mismas dimensiones estructurales, luego variará un poco la anchura de la parte donde se realiza el agarre dependiendo del grip que el jugador coloque, pero la parte que nos interesa será similar en todas.



Ilustración 12: Medida del ancho del mango de una pala



Ilustración 13: Medida del largo del mango de una pala

Siendo las dimensiones interiores 27 por 22,1 milímetros. Por lo que el dispositivo debería tener esas dimensiones máximas de la base para poder pegarlo correctamente.

La dimensión de profundidad va a depender de los elementos electrónicos para las medidas, el tamaño del microcontrolador y la IMU.



3.1.2 Definición de las magnitudes a medir

El objetivo final de este proyecto es medir diferentes parámetros que afecten directamente al golpeo de la bola para poder tener datos sobre el juego. Se buscarán medir los siguientes datos:

- Velocidades y aceleraciones en el mango.
- Energía con la que se realiza cada golpeo.
- Frecuencias en las ondas de aceleraciones obtenidas.
- Salida de bola en el golpeo.
- Potencia de golpeo a la bola.
- Se medirá también la temperatura a la que se juega.

4. DESCRIPCIÓN Y JUSTIFICACIÓN DE LA SOLUCIÓN ADOPTADA

En este capítulo se va a explicar las diferentes elecciones al realizar el proyecto:

- **A nivel de hardware:** que sensores y microcontrolador se ha elegido.
- **A nivel de software:** que aplicación se ha utilizado para tratar los datos.
- **A nivel de comunicaciones:** que protocolo se ha empleado para transmitir los datos a esa aplicación.

Se desarrollará que diferentes opciones se tenían y porque se han escogido.

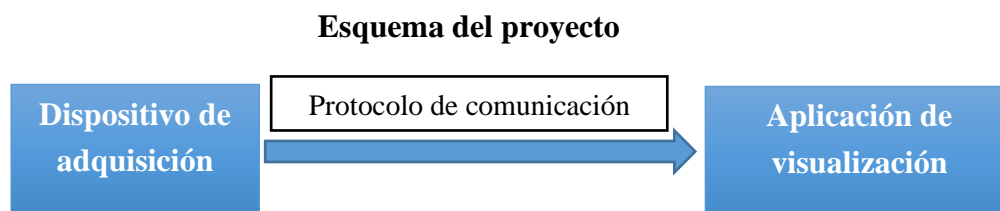


Ilustración 1: Esquema de bloques de funcionamiento del proyecto

4.1 Elección del Hardware

Una vez definidas las magnitudes que se van a medir se realiza un estudio para seleccionar los elementos que conformarán el dispositivo final, dividiéndolos estos en tres partes:

- Dispositivos de control
- Dispositivos de adquisición de datos o sensores
- Dispositivos de alimentación

4.1.1 Dispositivos de control

Para realizar la lectura y el tratamiento de los datos obtenidos por los sensores se va a utilizar un microcontrolador que posteriormente envíe los datos a un ordenador para que puedan ser tratados.

Como el dispositivo va a ir acoplado al mango de la pala de pádel, un requisito indispensable es que esta comunicación sea inalámbrica ya que jugar con un cable unido a un ordenador sería imposible.

Se realiza un estudio entre diferentes placas comerciales existentes en el mercado. A continuación, se muestra una breve tabla comparativa entre diferentes módulos.

Especificaciones	ESP32 modulo wroom-32	ESP32-S2- DevKit- LiPo	ESP8266 modulo	ARDUINO UNO	ARDUINO MICRO
Frecuencia CPU	160 MHz	240 MHz	80 MHz	16 MHz	16 MHz
WiFi	SI	SI	SI	NO	NO
Bluetooth	SI	SI	NO	NO	NO
Memoria RAM	512 KB	320 KB	160 KB	2 KB	2,5 KB
Memoria flash	16 MB	4 MB	16 MB	32 KB	32 KB
GPIO pins	36	43	17	14	20
ADC pins	18	18	1	6	12
Dimensiones	5.6 x 2.8 cm	4.8 x 2.8 cm	5.8 x 3.1 cm	8 x 5.51 cm	4.8 x 1.8 cm
Conector batería	NO	SI	NO	NO	NO

Ilustración 4: Comparativa micro controladores[9], [10]

Como se puede observar en la Tabla, se ha hecho un estudio entre diferentes microcontroladores entre la familia de Arduino y de espressif.

Por un lado, el Arduino nano, destaca por su reducido tamaño y el Arduino UNO por su facilidad de programación y gran cantidad de información existente con respecto a él. Sin embargo, han sido descartados, porque como se ha comentado anteriormente, un requisito indispensable es que cuenten con comunicación inalámbrica, y no tienen. Se podría instalar un módulo bluetooth o Wifi, pero incrementaría el tamaño que es la principal ventaja.

Entonces, la solución óptima pasó a ser elegir un módulo de espressif, dentro de los cuales están el módulo esp32, esp8266 y esp32-s2-lipo.

Los tres cuentan con comunicación wifi y los esp32 con bluetooth, en frecuencia de procesado destaca el esp32-s2-lipo, sin embargo, los otros dos cuentan con más memoria flash, y finalmente el esp32-s2-lipo también cuenta con conector para baterías, una aportación importante para este proyecto.

El objetivo final es que los datos obtenidos por el micro, lleguen a un ordenador, por lo que una forma sencilla de realizar la comunicación es utilizando otro microcontrolador que sea capaz de recibir los datos enviados por el que va acoplado a la pala. Y posteriormente este mediante el puerto serie los introduciría en el ordenador. Sería una

comunicación unidireccional en la que un dispositivo enviaría los datos y el otro los recibiría.

4.1.2 Micro controlador elegido

Por lo tanto, la elección final ha sido el chip ESP 32, concretamente dos módulos llamados **ESP32-S2-DEVKIT-LIPO OLIMEX**, ya que cuentan con alimentación mediante batería li-po.

Además, es el que mayor frecuencia de CPU tiene, importante para poder adquirir los datos lo más rápido posible.

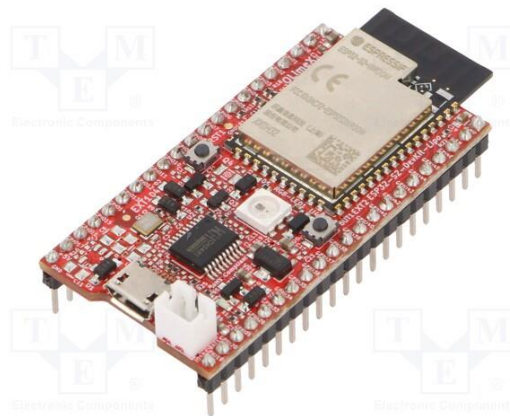


Ilustración 5: Módulo esp32s2 lipo olimex

Las principales características de este chip son:

- Voltaje: 3,3 V CC o 5 V CC
- Chip ESP32 (procesador de doble núcleo 240 Mhz)
- SRAM: 520 KB
- PSRAM: 8 MB
- Memoria flash: 4 MB
- Wifi
- Bluetooth
- Conector baterías lipo



4.1.3 Dispositivos de adquisición de datos

El principal elemento del sistema que va a adquirir los datos necesarios para poder monitorizar los golpes, será un acelerómetro que como comentamos anteriormente irá situado en el mango de la pala.

Existen infinidad de acelerómetros en función de los rangos de aceleración que se quiera medir, por ello antes de decidir ningún acelerómetro en concreto, se realizaron unos ensayos en el laboratorio de pádel del instituto ai2 de la Universidad Politécnica de Valencia.

4.1.3.1 Ensayos para la selección del acelerómetro

Para determinar, aproximadamente el rango que el acelerómetro va a necesitar, se realizó un ensayo que consistió en probar diferentes acelerómetros del instituto Ai2 al mango de la pala, y mediante una tarjeta de adquisición de datos de National Instruments, y un breve programa en Labview ver la respuesta y así ver si se saturaban al realizar la medida o no.

El ensayo consistió en dos partes, la primera en una estación estática del Ai2 en la que mediante un péndulo con una pelota de pádel al final se golpea en la pala que está estática, y la segunda prueba con un golpeo real, saliendo fuera del laboratorio, realizando el golpe una persona.

Los acelerómetros que se utilizaron tenían una salida analógica entre 0 y 5 Voltios.

- Primero se probó con un acelerómetro de 200 g pero rápidamente se vio que al realizar la medida en la estación estática llegaba a saturarse por lo que se pasó a otro con más rango.
- El siguiente fue uno de 500 g mono axial, viendo que en la estación estática no saturaba y que al realizar los golpes en exterior llegaba como máximo a 4,1- 4,3 V en los más fuertes por lo que se vio que no saturaba, aunque cabe destacar que en un juego real podría darse la posibilidad de imprimir más fuerza en los golpes.

4.1.3.2 Elección de los acelerómetros

Como el acelerómetro que se tiene que escoger es de un rango muy grande, para la mejor detección de los golpes se decide instalar dos acelerómetros

Incluyendo uno de rango pequeño con giroscopio para poder detectar la posición de la pala.



Acelerómetro rango menor

En la elección de este acelerómetro se buscan las siguientes características:

- Media en los tres ejes, X, Y, Z para detectar la posición de la pala en el instante de golpeo
- Rango de 8 o 16 g
- Alimentación de 3.3 V
- Puede ser analógico o digital, bus i2c o spi

Tras una búsqueda se barajaron dos modelos, principalmente basándose en estos dos por su cantidad de documentación y facilidad a la hora de programarlos, son los siguientes:

- **ADXL 345:**
 - Alimentación de 3.3 a 5 V
 - Comunicación i2c o SPI
 - Sensibilidad configurable de 2g a 16g
- **IMU MIKROE 1577 (MPU 6050):**
 - Alimentación 3.3 V
 - Comunicación i2c o SPI
 - Rango configurable de 2g a 16g
 - Incluye giróscopo con rango configurable
 - Incluye sensor de temperatura
 - Cuenta con 6 grados de libertad.

Acelerómetro rango mayor

Para la elección de este acelerómetro se buscaban diferentes consideraciones:

- Alimentación de 3.3 V, salida máxima de 3.3 V que es el máximo rango del ADC del micro controlador.
- Monoaxial

Se estudiaron dos modelos diferentes:

- **ADXL1004BCPZ-RL7:**
 - Acelerómetro analógico con alimentación de 3.3 a 5 V
 - Sensibilidad de 2.7 mV/g
 - 500 g de rango de medida.
- **ADXL373BCCZ-RL7:**



- Acelerómetro digital (i2c),
- Triaxial
- Alimentación de 1.6 a 3.5 V
- 400 g de rango de medida.

Dispositivos elegidos

Los dispositivos elegidos son:

- **IMU MIKROE 1577:** Para el acelerómetro de menor rango se ha optado por la unidad de medida inercial (IMU), que cuenta con el acelerómetro MPU 6050, ya que al contar con acelerómetro y giróscopo se puede determinar con mayor exactitud la posición de la pala en el instante del golpeo. Otro factor que ha ayudado a tomar esa decisión es que el sensor MPU 6050 es muy común, lo que hace que se pueda acceder a bastantes librerías y ejemplos en el caso de aparecer dudas, además de que al contar con un sensor de temperatura no será necesario añadirlo.
- **Acelerómetro ADXL1004BCPZ-RL7:** Se ha optado por este acelerómetro ya que tiene un rango de 500g, mayor que la otra opción que era de 400, suficiente para que no se sature, además cuenta con salida analógica lo que hace que se pueda muestrear más rápido que utilizando el bus i2c que va a estar ocupado por la IMU.

4.1.4 Dispositivos de alimentación

Una vez elegidos el resto de elementos, como el módulo final es inalámbrico se buscó como alimentarlo, teniendo como principal requisito que ocupe el mínimo tamaño con la máxima autonomía para interferir lo mínimo en el juego.

Entre las diferentes opciones que se barajaron estaban:

- **Una batería externa** (tipo powerbank), la cual se descartó porque tiene muchas pérdidas ya que internamente hace conversiones de voltaje internamente. Además de un tamaño elevado.
- **Pilas estándar (NI-MH):** Las pilas comunes suministran un voltaje de 1,2 V, por lo tanto, se necesitarían 3, opción descartada por el elevado tamaño.
- **Baterías recargables lipo** (polímero de Litio): Batería que suministra un voltaje de 3,7 V, adecuado para el módulo ya que viene con un regulador incorporado, tamaño reducido y alta autonomía, por lo que se eligió esta.

4.1.4.1 Bateria elegida

Finalmente, la alimentación elegida es una batería recargable lipo de 3,7 V y 1,8 Ah, debido a su reducido tamaño, facilidad de colocación y principalmente porque el módulo de control tiene conector para este tipo de baterías.



Ilustración 6: Bateria Lipo

4.2 Elección del software

En este apartado se explica el software elegido para las dos etapas del proceso:

- Software elegido para la programación del microcontrolador que adquiere y envía los datos de los sensores.
- Software para la recepción, tratamiento y visualización de esos datos.

4.2.1 Software de programación del micro controlador

Primero se detallará el programa que se ha utilizado para realizar la lectura de los dos sensores elegidos previamente, así como el envío de estos datos a la aplicación de visualización. El micro elegido es el esp32 s2, y principalmente se han barajado dos opciones:

- ESP-IDF.
- IDE de Arduino.

4.2.1.1 ESP-IDF

Es el entorno de desarrollo de software proporcionado por Espressif, que es el fabricante de las tarjetas esp32, tiene la ventaja que cuenta con librerías para aprovechar al máximo las funcionalidades del esp32. Se utiliza el lenguaje de programación c.

La principal ventaja es que esta construido sobre freeRTOS, lo que hace que se puedan utilizar simultáneamente los núcleos del esp32. Pero cuenta con la desventaja que tiene una gran complejidad de instalación y utilización si no se está familiarizado con él.

También se pueden utilizar otros IDEs como PlatformIO que se integra en diferentes softwares de programación y cuenta con todas las funcionalidades de esp-idf.

4.2.1.2 IDE de Arduino

Este IDE es el entorno de programación de las tan conocidas placas de Arduino, que está basado en c++. No solo se limita a la programación de las placas de Arduino sino que tiene capacidad de programar una gran cantidad de placas existentes en el mercado.

Cuenta con una infinidad de librerías para los diferentes sensores y protocolos que se pueden utilizar, con la desventaja que no está basado en freeRTOS aunque si cuenta con una librería.

Sus mayores ventajas son la sencillez de instalación y adaptación a diferentes placas, programación relativamente intuitiva y la gran cantidad de recursos existentes como librerías y ejemplos para diferentes sensores.



Ilustración 7: Logotipo Arduino

4.2.1.3 Elección del entorno de programación del micro controlador

Se ha tomado la decisión de elegir el IDE de Arduino, por su facilidad de programación y teniendo en cuenta la gran cantidad de librerías existentes en el mercado.

También se ha tenido en cuenta que ya cuento con experiencia en la programación de las placas esp32 mediante el uso de este entorno.

4.2.2 Software de recepción, tratamiento y visualización de los datos

Ahora se explicará el software elegido para llevar a cabo la recepción de datos enviados por el dispositivo que realiza la adquisición de datos, que se encargará no solo de recibir los datos, sino que también de realizar un tratamiento para poder visualizarlos correctamente y extraer información de ellos.

Para el tratamiento de los datos se han barajado principalmente dos opciones:

- MATLAB
- LabVIEW

4.2.2.1 MATLAB

MATLAB (MATrix LABoratory) es una aplicación de computación numérica que ofrece su propio entorno de desarrollo (IDE) que utiliza su lenguaje de programación propio, el lenguaje M.

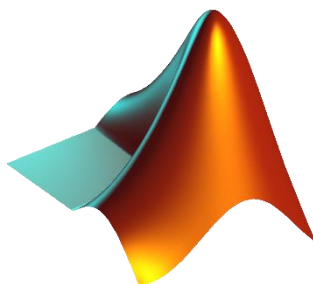


Ilustración 8: Logotipo MATLAB

Tiene una gran cantidad de posibilidades: manipulación de matrices, creación de algoritmos, representación de funciones, creación de aplicaciones con interfaz de usuario, comunicación con otros lenguajes y dispositivos, y además cuenta con una gran cantidad de aplicaciones ya creadas en su Toolbox para poder realizar estas funciones.

Es muy útil para el proceso del análisis de datos ya que es capaz de realizar todo el proceso, desde la adquisición de un dispositivo externo, el procesado y representación y su análisis numérico.[11]

4.2.2.2 LabVIEW

LabVIEW es un software que aporta un entorno de programación gráfico que permite diseñar aplicaciones principalmente de adquisición de datos, análisis de esos datos y representación. Tiene la ventaja que al ser un entorno gráfico no tiene la complejidad de otros lenguajes escritos.

Cuenta con muchas funciones de Entrada y Salida, de análisis de datos y de representación de los mismos. Tiene también muchos programas de ejemplo a los cuales se puede acceder con facilidad.[12]

En resumen, permite interactuar con señales en tiempo real, analizar datos para extraer la información de esas señales y representar esos datos a través de pantallas o informes.



Ilustración 9: Logotipo LabVIEW

4.2.2.3 Elección del programa de tratamiento de datos

Finalmente, el programa utilizado para la recepción y el tratamiento de datos ha sido LabVIEW por diferentes razones:

- Aunque MATLAB ha sido más utilizado a lo largo del Máster al cual pertenece este proyecto, no ha sido utilizado para la adquisición de datos de sensores por lo que se elige LabVIEW ya que, al ser un entorno gráfico, a priori va a ser más intuitivo y sencillo.
- Las aplicaciones de adquisición de las diferentes estaciones del laboratorio de pádel del ai2 junto con testea pádel están creadas en este software, por lo que eligiéndolo se facilitará la integración con el resto de ensayos.

4.3 Protocolos de comunicaciones

Una vez elegidos el hardware a utilizar y los softwares donde se va a realizar la programación del dispositivo de adquisición y la aplicación de tratamiento y visualización es hora de elegir de qué manera se van a transmitir los datos desde el dispositivo a la



aplicación, es decir el protocolo de comunicación y para ello se tienen las siguientes opciones:

- UDP
- TCP
- ESP-NOW

Desde el punto de vista de las comunicaciones se necesita que se tenga una rápida velocidad de envío de datos y que no se pierda ningún paquete.

4.3.1 Protocolo UDP

Se trata de un protocolo en el que no existe conexión que se basa en el envío de datagramas, un nodo envía datos a otra de forma unidireccional, sin establecer una previa conexión entre ellas.

Los datos se envían sin saber si se van a recibir. Se reduce al máximo la cantidad de información que se envía, es un protocolo rápido, pero no tiene máxima fiabilidad.

Tiene la desventaja de que para realizarse una comunicación hace falta que el nodo esté conectado a la misma red que el dispositivo que lo va a recibir, siendo esto un problema en este proyecto porque no en todas las pistas de pádel se tendrá esa conexión[13]

4.3.2 Protocolo TCP

Se trata de un protocolo de transporte orientado a conexión, que permite enviar datos de un nodo a otro y es capaz de detectar errores y corregirlos. Los nodos de envío y recepción están conectados entre sí, lo que garantiza una máxima fiabilidad.

Necesita también, al igual que UDP conexión a la misma red que el dispositivo receptor.[13]

4.3.3 Protocolo ESP-NOW

Este es un protocolo de comunicación vía radio propio de los dispositivos de la marca Esspressif. Utiliza la banda de los 2.4 GHz



Se trata de un protocolo del tipo maestro/esclavo donde hay un maestro y puede haber varios esclavos. Es utilizado para la comunicación de dispositivos de baja energía, consiguiendo un alcance de hasta 200 m.

El funcionamiento requiere que los dos dispositivos se emparejen entre sí, una vez se ha conseguido, la transmisión es automática. Es un protocolo que cuenta con una gran velocidad de envío de datos, tardándose unos 4 ms por envío.

Teniendo la desventaja que hacen falta dos nodos, el maestro y el esclavo, pero al ser un protocolo propio entre los dos dispositivos no necesitan conectarse a una red por lo que lo hace idóneo para este proyecto que se harán pruebas en diferentes pistas y lugares.[14], [15]

4.3.4 Protocolo elegido

Finalmente se ha elegido el protocolo ESP-NOW, porque a pesar de la limitación de necesitar dos dispositivos, el maestro y el esclavo, tiene la gran ventaja de no necesitar estar conectados los dispositivos a una red WIFI. Teniendo en cuenta que este proyecto se tiene que poder utilizar en diferentes ubicaciones, por ejemplo, la estación de salida de bola del ai2 y en pistas.

Además, cuenta con librerías sencillas de utilizar a la hora de implementarlo.

4.4 Funcionamiento del dispositivo

Una vez se ha decidido que hardware, software y protocolo de comunicación se va a utilizar se va a definir el comportamiento del sistema de adquisición y envío, así como del tratamiento y visualización de los datos.

4.4.1 Funcionamiento del sistema de adquisición

El sistema de adquisición tiene como objetivo recopilar los datos que miden los sensores y posteriormente enviarlos desde un micro controlador al otro para posteriormente enviarlos por monitor serie al ordenador.

4.4.1.1 Planteamiento inicial

Inicialmente se había planteado la idea de realizarse las lecturas de los sensores secuencialmente, primero se lee la IMU y luego el acelerómetro analógico, cuyas lecturas se guardan en una estructura de datos tipo struct para luego ser enviados por el protocolo esp-now al micro.



El planteamiento inicial consistía en realizar una lectura y su consiguiente envío, pero no se conseguía una velocidad de lectura y envíos adecuada, haciéndose imposible el objetivo final que es poder visualizar correctamente la onda de aceleración por lo que se descartó.

4.4.1.2 Funcionamiento elegido

Finalmente se optó por la decisión de crear un buffer donde se almacenan todos los datos leídos, y luego enviarlos, es decir cuando se inicia la adquisición, el micro controlador está un tiempo leyendo los sensores y almacenando estos datos en memoria hasta que llega a un número final que se ha definido.

Una vez se ha llenado el buffer, envía los datos uno a uno creando estructuras tipo struct. Estos datos se envían al otro micro controlador que está conectado al ordenador por el puerto USB, y mediante el puerto serie es capaz de leerlos.

Estudiando detenidamente este programa se puede observar que se tiene una frecuencia de muestreo de 2.8 ms, frecuencia que en ocasiones puede no ser adecuada por lo que para ello se crean dos modos de funcionamiento:

- IMU + acelerómetro analógico = Frecuencia de muestreo 2.8 ms
- Solo acelerómetro analógico = Frecuencia de muestreo 0.83 ms

Cabe destacar que el rango del acelerómetro de la IMU se configura a 16 g y del giróscopo a 1000 °/sec

4.4.2 Funcionamiento de la aplicación de tratamiento de datos

Como se ha comentado, los datos llegan del segundo micro controlador al ordenador, concretamente a LabVIEW por el puerto serie.

Se crearán dos aplicaciones diferentes en LabVIEW:

- Recepción y almacenamiento de datos.
- Tratamiento y muestra de esos datos.

Se creará una aplicación por cada modo de funcionamiento, es decir dos de recepción y dos de tratamiento, para los modos de IMU más acelerómetro analógico y solo acelerómetro analógico.

Aplicación de recepción y almacenamiento de datos



Se encargará de recibir los datos que llegan por el puerto serie, acondicionarlos en el formato adecuado y almacenarlos en un fichero para cada bloque de datos recibidos.

Tratamiento y muestra de esos datos

Esta aplicación se encargará de leer los ficheros de datos donde están almacenados los golpes para su posterior tratamiento y muestra en forma de gráficas.

Mediante el tratamiento se extraerán diferentes datos que se explicarán en el apartado del diseño de la aplicación.

5. DISEÑO DEL DISPOSITIVO

En este apartado se va a explicar cómo se ha realizado el montaje del hardware, explicando por los diferentes prototipos por los que se ha pasado.

Se explicarán las necesidades que se tenían que solventar antes de comenzar a construir el prototipo.

5.1 Decisiones de diseño

Para facilitar el uso del dispositivo cuando se vayan a realizar las pruebas en el dispositivo final se tomaron las siguientes decisiones:

- **Utilizar el botón del módulo** para empezar a realizar la toma de datos.
- **Utilizar el LED interno** que tiene el módulo esp32 elegido, que es un LED RGB, el cual tiene la función de indicar los tres estados del dispositivo:
 - **Parpadeo:** Tiene la función de avisar al usuario que cuando transcurran 5 segundos se va a empezar la toma de datos.
 - **Color Verde fijo:** Se está realizando la lectura de los dos acelerómetros y almacenándose los datos en el micro, la duración depende del modo de funcionamiento.
 - **Color Rojo:** Los datos recogidos se envían al otro micro, la duración depende de la cantidad de datos y por lo tanto del modo de funcionamiento.

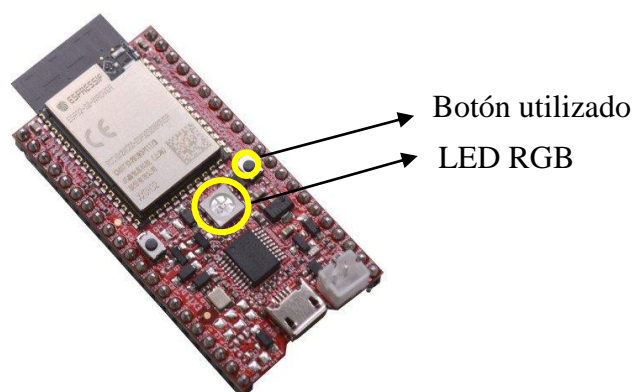


Ilustración 10: Módulo esp32 con el Led y botón señalizados

5.2 Necesidades de diseño

Para realizar el montaje, únicamente hay que conectar los dos sensores, la IMU y el acelerómetro analógico de 500g de los que se habló en el capítulo anterior.

La conexión de la IMU es sencilla, es una conexión I2C en la que se alimenta el sensor a 3.3 V y se conectan los pines SDA y SCL a los correspondientes pines del microcontrolador.

5.2.1 Conexión del Acelerómetro Analógico

En la conexión del acelerómetro analógico existe algo más de complejidad ya que el ADC del microcontrolador no tiene el mismo voltaje de entrada que el voltaje de salida del acelerómetro por lo que habrá que adaptarlo.

5.2.1.1 Voltaje de entrada del ADC del ESP32S2

Para encontrar el voltaje de entrada del ADC del ESP32s2 se recurre al datasheet y se puede observar que esta familia de micro controladores cuenta con dos ADCs de 13 bits del tipo SAR (Successive Approximation Register), los cuales están multiplexados en 20 canales. 10 pines a un ADC y 10 al otro.

En el Datasheet se encuentra la siguiente tabla.

SoC	attenuation (dB)	suggested range (mV)
ESP32	0	100 ~ 950
	2.5	100 ~ 1250
	6	150 ~ 1750
	11	150 ~ 2450
ESP32-S2	0	0 ~ 750
	2.5	0 ~ 1050
	6	0 ~ 1300
	11	0 ~ 2500

Ilustración 11: Tabla Voltaje esp32s2

Se puede observar como para una atenuación de 11 dB que es la que viene por defecto, el voltaje de entrada del ADC es entre 0 y 2.5V.

5.2.1.2 Voltaje de salida del acelerómetro analógico

Para calcular el voltaje de salida que va a tener el acelerómetro se recurre al datasheet.

Este, nos indica que se puede alimentar entre 3.3 y 5.25 V, para este proyecto se alimentará a 3.3 V que es el voltaje que proporciona el esp32.

Para calcular el voltaje de salida habrá que construir la recta de calibración, en a hoja de datos se indica que, a 0 g, es decir el acelerómetro en reposo, el voltaje de salida será $V_{DD}/2$, es decir la mitad de la alimentación.

ZERO g OFFSET (RATIOMETRIC TO V_{DD})
0 g Output Voltage | | $V_{DD}/2$ | V

Ilustración 12: Voltaje de salida a 0g

La pendiente de la recta será la sensibilidad de medida, que variará en función del voltaje de alimentación:

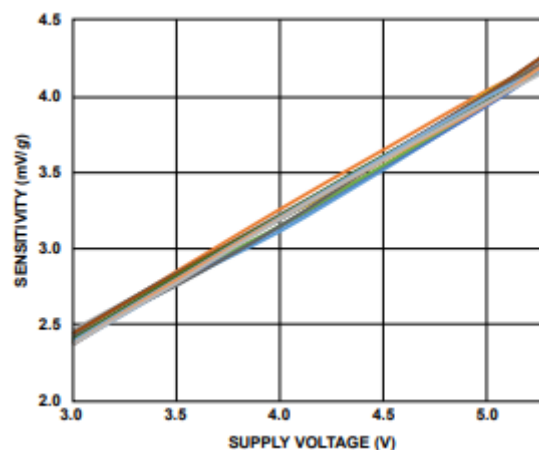


Ilustración 13: Sensibilidad frente a voltaje de alimentación

Como se aprecia a 5 Voltios de alimentación se tiene una sensibilidad de 4 mV/g, a 3.3 como es el caso de este proyecto se ve una sensibilidad de 2.7 mV/g.

Por lo que la recta se construirá así:

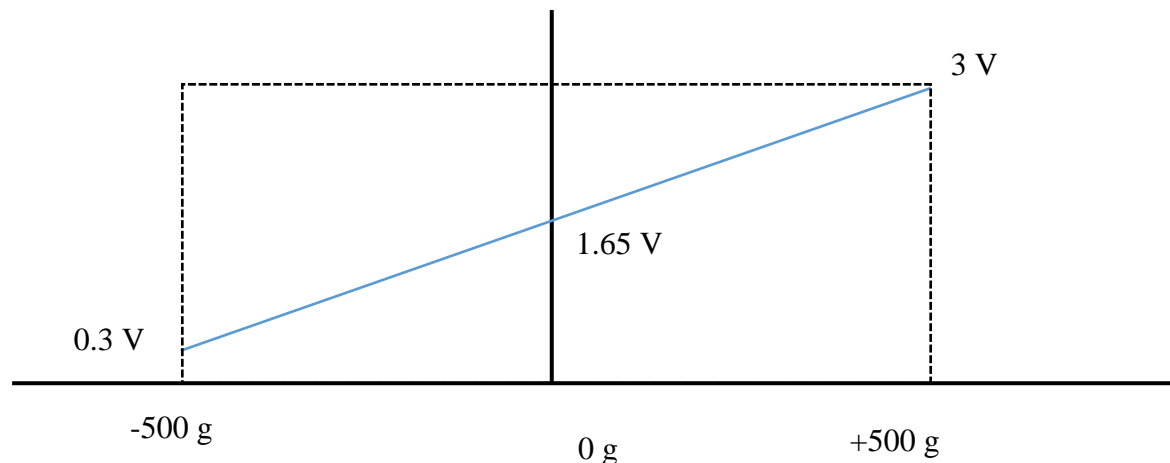
Se calculan los dos extremos de la recta.

$$A 0 g = \frac{V_{dd}}{2} = \frac{3.3}{2} = 1.65 V$$

$$a (500 g) = 2,7 \frac{mV}{g} * 500 g = 1.35 V + 1.65 V = 3 V$$

$$a (-500 g) = 2,7 \frac{mV}{g} * -500 g = -1.35 V + 1.65 V = 0.3 V$$

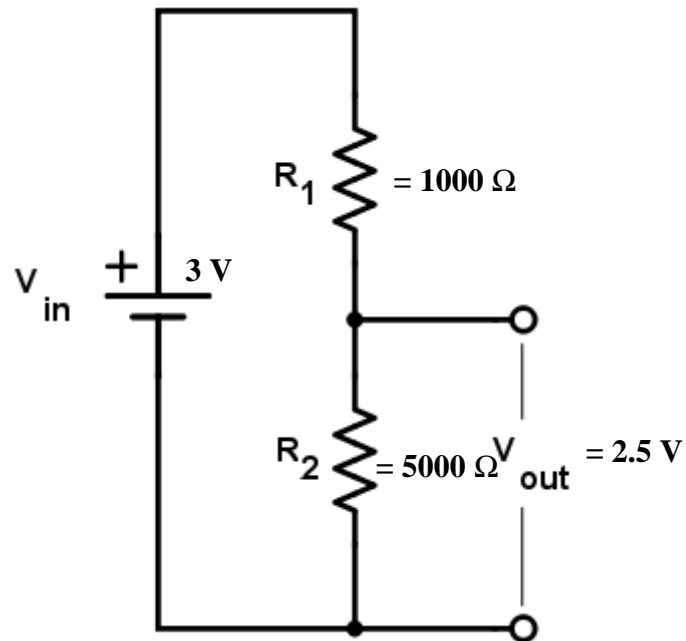
Teniendo los extremos y la pendiente se construye la recta.



Se observa que la salida del acelerómetro analógico varía entre 0.3 y 3 Voltios.

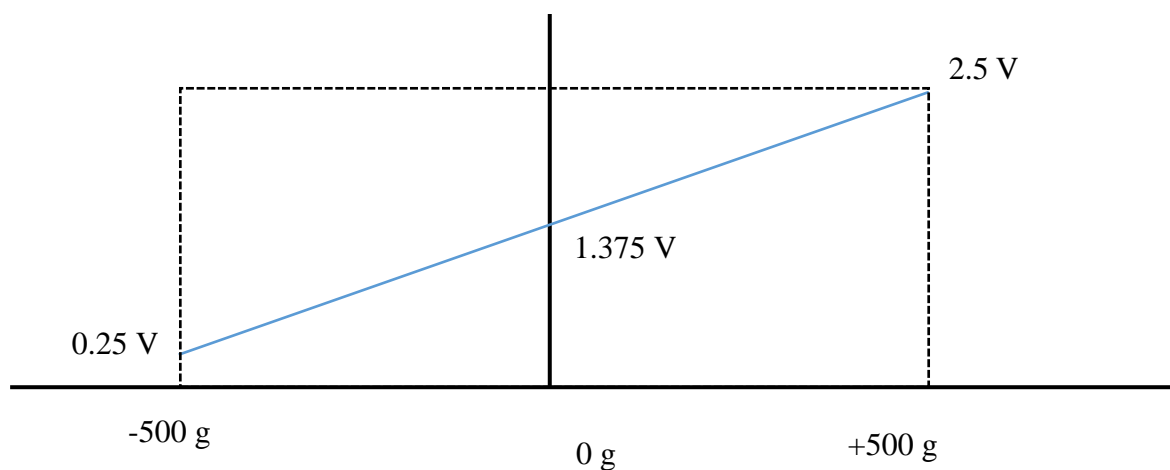
Mientras que la entrada del ADC del micro controlador va de 0 a 2.5 Voltios, por lo que habrá que acondicionar esa señal, para ello se recurre a un divisor de tensión.

Haciendo los cálculos pertinentes, se llega al resultado que hay que instalar una resistencia de 1000Ω y otra de 5000Ω



Il·lustració 14: Divisor de Tensió

Quedando así la recta de calibración a la entrada del ADC, después de pasar por el divisor de tensión, siendo la sensibilidad 0.00225 V/g:



Una vez realizados estos cálculos ya se puede realizar el esquema eléctrico para hacer el montaje.

5.3 Esquema eléctrico

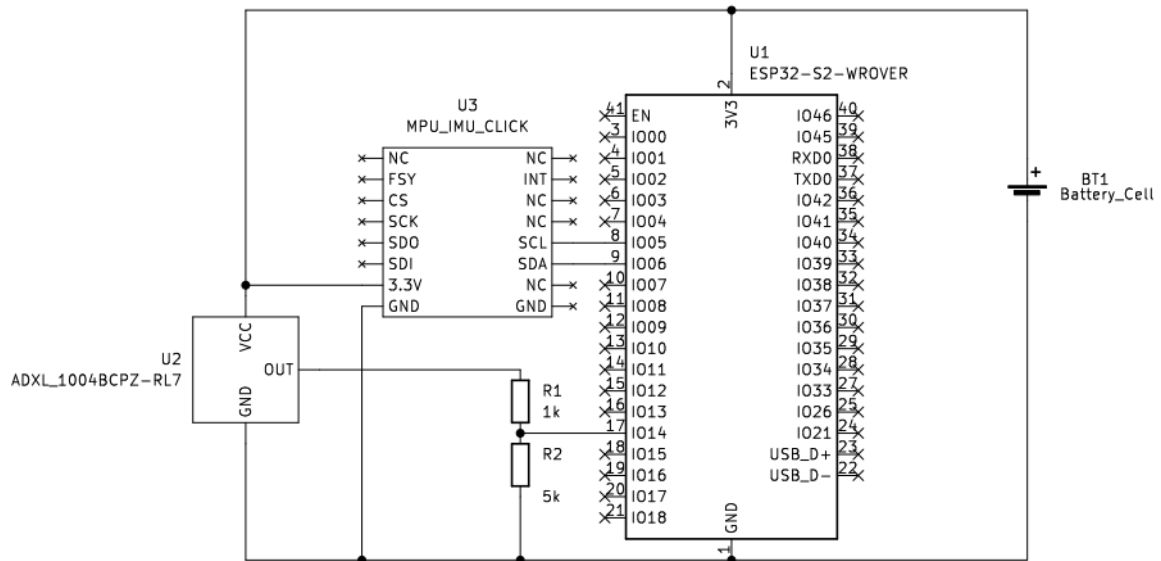


Ilustración 15: Esquema eléctrico

5.4 Primer Prototipo

Una vez se tiene claro cuál va a ser la disposición de todos los sensores, se realiza una primera prueba de montaje, para ello se hace un sencillo prototipo ayudándose de una placa protoboard, en la que los dispositivos de control van conectados en la placa protoboard y los dos sensores van unidos y conectados a la protoboard mediante un cable.

5.5 Segundo Prototipo

Viendo que las medidas no son exactas debido a que al realizar movimientos bruscos se sueltan cables, se pasa a realizar un segundo prototipo utilizando una placa perforada, a la cual se sueldan los cables y el divisor de tensión.

DÍPOSITIVO EMPOTRADO PARA MEDIR CARACTERÍSTICAS DE PALAS DE PÁDEL

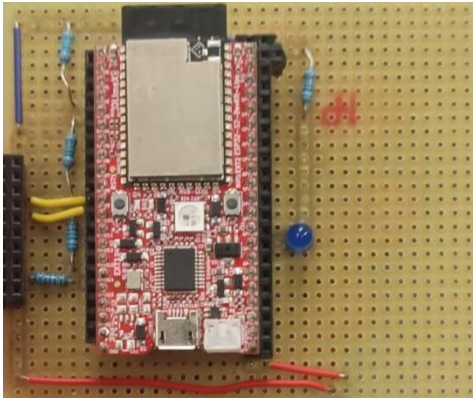


Ilustración 17: Vista desde arriba segundo prototipo

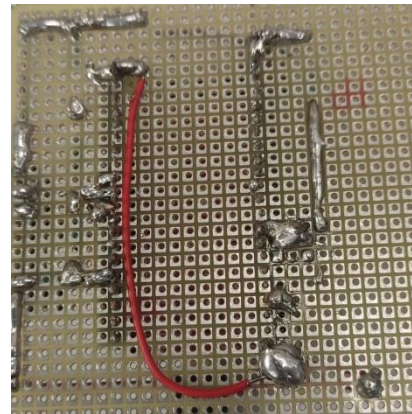


Ilustración 16: Conexiones y soldaduras segundo prototipo

Los dos sensores van conectados entre sí también mediante soldadura a una placa perforada y estos a su vez se conectan a la placa mediante un cable largo.



Ilustración 18: Placa de sensores segundo prototipo

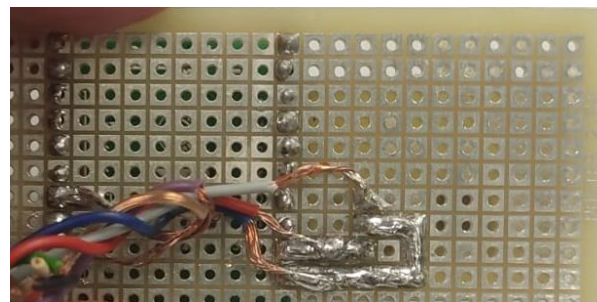


Ilustración 19: Soldaduras y conexiones segundo prototipo

Con este montaje ya se pueden empezar a realizar las primeras pruebas, pero no es máximamente funcional debido al gran tamaño de la placa y a la necesidad de tener conectados los sensores por un cable tan largo a la placa de control.

5.6 Dispositivo Final

Por último, se realiza el montaje del dispositivo final.

Para ello se construye una PCB a medida, con el siguiente esquema:

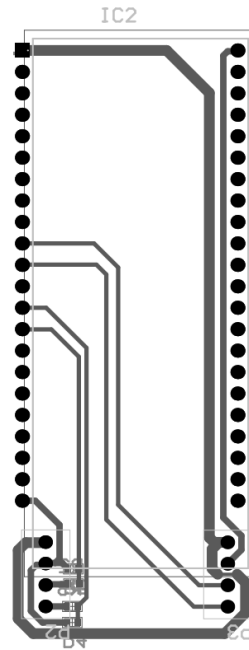


Ilustración 20: Esquema PCB

El resultado del prototipo final es el siguiente:

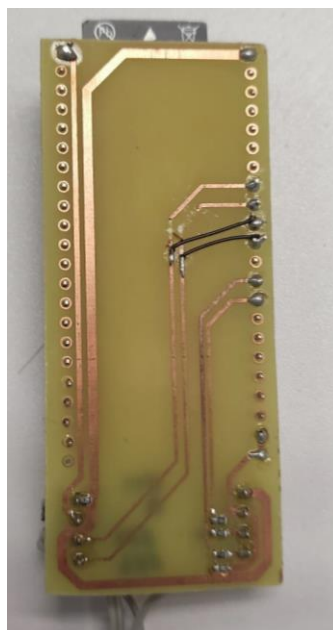


Ilustración 21: PCB final

DÍPOSITIVO EMPOTRADO PARA MEDIR CARACTERÍSTICAS DE PALAS DE PÁDEL

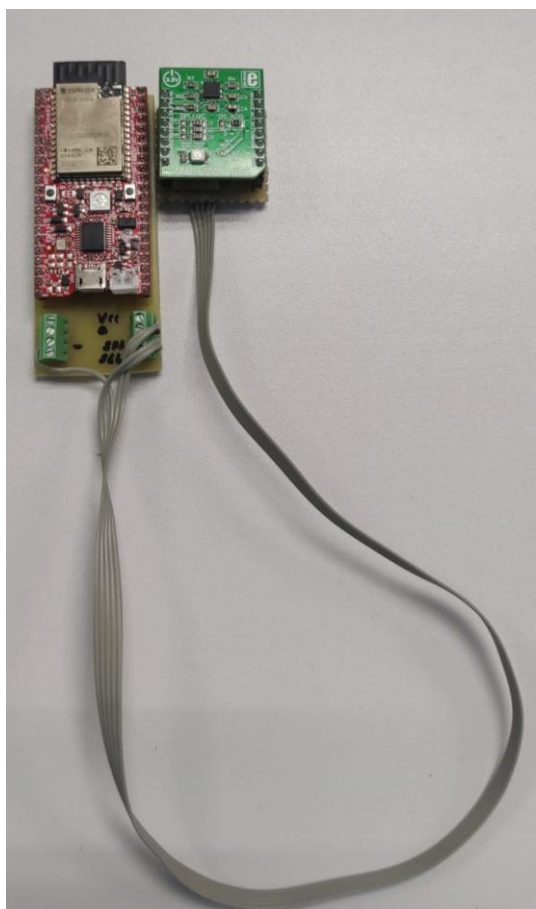


Ilustración 23: Prototipo final vista superior

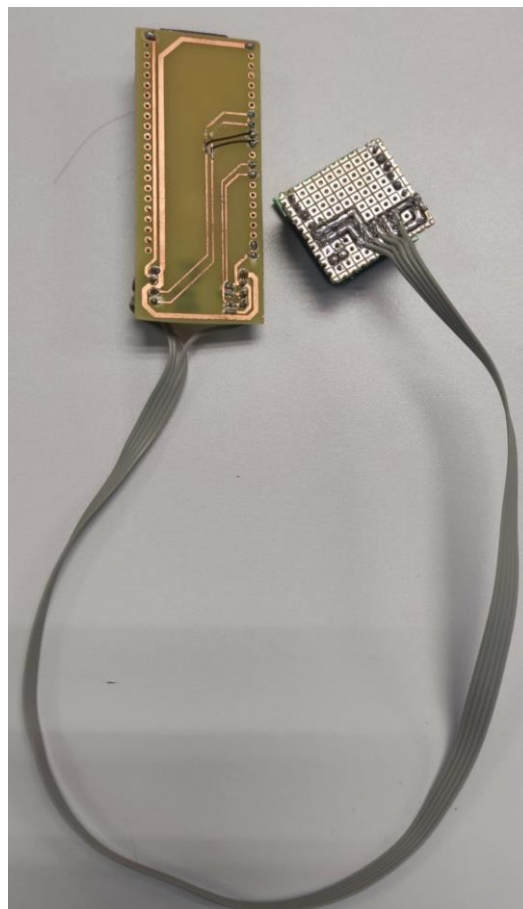


Ilustración 22: Prototipo final vista inferior



Ilustración 24: Prototipo final en la pala

Aquí se puede ver el resultado del prototipo final, en las dos primeras imágenes se ve el dispositivo de control conectado a los sensores y en la tercera, ya unido a la batería y listo para ser utilizado en diferentes ensayos.



6. PROGRAMACIÓN DE LOS MICROCONTROLADORES

Como se explicó al principio, en este proyecto se cuentan con dos micro controladores, el dispositivo de adquisición de datos, el cual recoge los datos y posteriormente los envía al dispositivo receptor, en este capítulo se va a explicar la programación de estos dos dispositivos.

6.1 Programación del dispositivo de adquisición

Una vez se ha detallado como ha sido el diseño y montaje del dispositivo de adquisición de datos, se va a hablar de su programación.

Como se explicó anteriormente, existen dos modos de funcionamiento:

- Adquisición de datos de la IMU y del acelerómetro analógico.
- Adquisición de datos solo del acelerómetro analógico.

El hecho de realizar dos programas diferentes es para poder aumentar la velocidad de muestreo ya que en el primer caso son 2,8 ms y en el segundo 0,83 ms.

Los dos programas están creados mediante el Arduino IDE en lenguaje c, cuyos nombres son:

- *Dispositivo_adquisición_acelerómetro_analogico.ino*
- *Dispositivo_adquisición_acelerómetro_analogico_e_IMU.ino*

En esta parte de la memoria se va a realizar una breve explicación de los programas, detallando cada una de las partes, y mostrándose gráficamente mediante los flujogramas.

6.1.1 Diagrama de flujo

A continuación, se muestran los dos diagramas de flujo de los programas de adquisición y envío de datos.

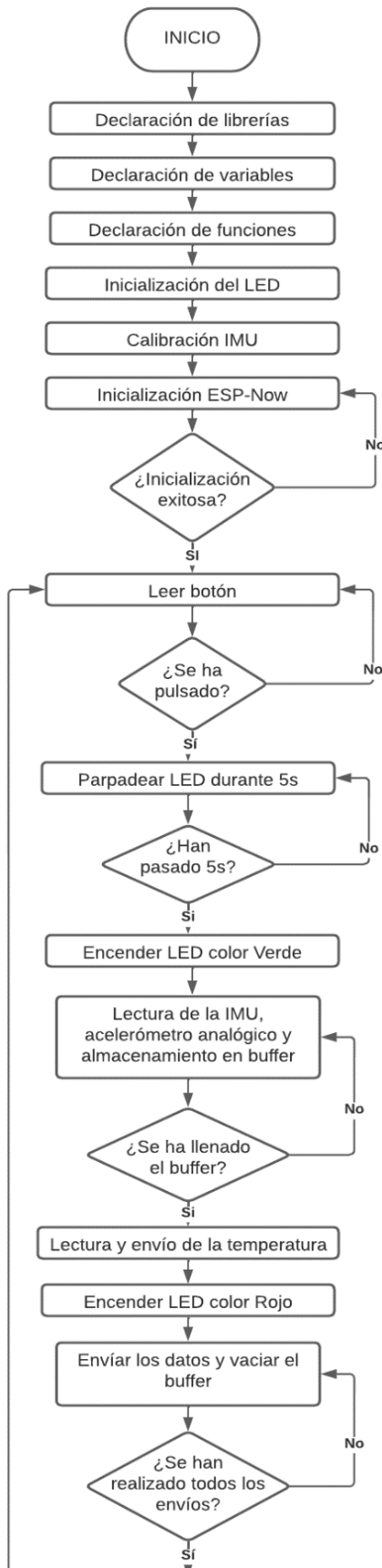


Ilustración 25: Flujoograma Modo de funcionamiento 1

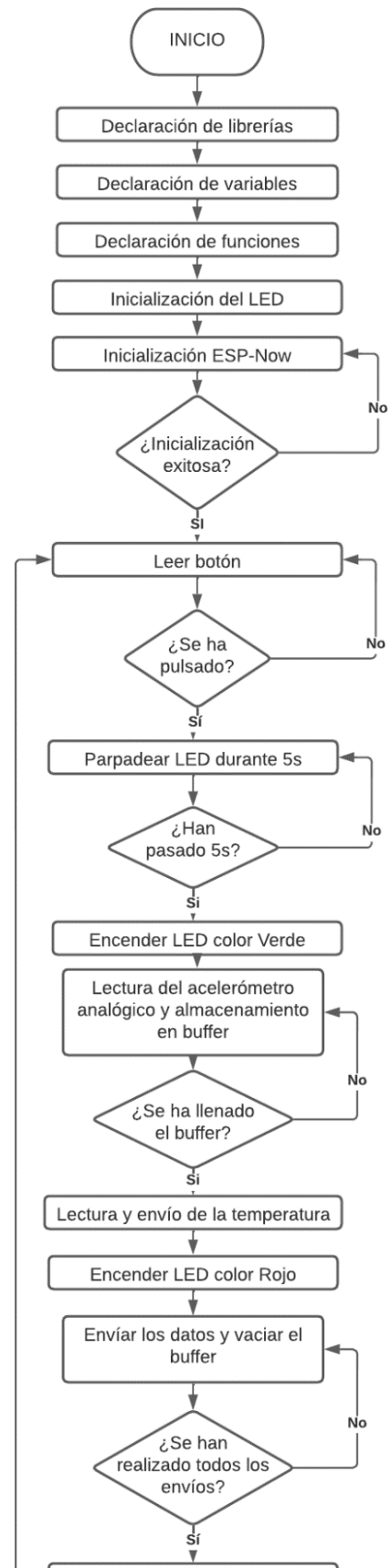


Ilustración 26: Flujoograma Modo de funcionamiento 2



6.1.2 Explicación del programa

Se va a explicar el código que se ha creado en base al flujograma, primero se explican las declaraciones y el “setup” o configuración del programa, y luego el “loop” o bucle principal de funcionamiento.

Declaraciones y SETUP

- **Declaración de librerías:** Primero se detallan las librerías que se necesitan para el uso de los sensores y protocolos:
 - **“I2Cdev.h”:** Programación de la IMU
 - **“MPU6050.h”:** Programación de la IMU
 - **“esp_now.h”:** Protocolo de comunicación
 - **“WiFi.h”:** Protocolo de comunicación
 - **“Adafruit_NeoPixel.h”:** Programación del LED
- El siguiente paso es la declaración de las variables que se utilizan en el programa, así como de las funciones que se han creado, que son las siguientes:
 - **Función lecturaVALORES():** Lee los valores de la IMU y del acelerómetro analógico, en el caso del segundo modo de funcionamiento solo lee el acelerómetro analógico
 - **Función lecturaTEMP():** Lee el valor de la temperatura.
 - **Función IRAM_ATTR empezar_enviar():** Es parte de una interrupción provocada por el botón y su misión es poner una variable a 1 una vez que se ejecuta la interrupción.
- Posteriormente se inicializan los sensores y el LED, mediante su clase, para poder ser utilizados.
- **Calibración IMU:** El siguiente paso es la calibración de la IMU, que consiste en indicarle al microcontrolador los offsets a cada parámetro de la IMU para que mida correctamente, cabe destacar que para calcular estos offsets se ha recurrido a un script de calibración presente en la librería “MPU6050.h” llamado “IMU_Zero”.
- **Inicializar esp-now:** Para terminar la configuración del programa se inicializa el protocolo de comunicación esp_now y se comprueba esta inicialización, si es correcta se termina la configuración y en caso contrario se sigue intentando realizar esta inicialización. Cabe destacar que para que la inicialización correcta de este protocolo hay que aportar al programa la dirección MAC del programa esclavo que está esperando para recibir los datos, esta dirección es sencilla de



obtener mediante un programa llamado “*getmacadress.ino*” disponible en la librería del propio protocolo.

Bucle principal LOOP

- **Leer botón:** Para empezar a la toma de datos, el programa está esperando a que se produzca una interrupción provocada por la pulsación del botón interno del micro controlador, cuando esto se produce se pone una variable a 1 y empieza la ejecución del loop.
- **Estado de aviso:** Una vez se ha pulsado el botón se pasa a un estado cuya función es avisar de que se van a empezar a tomar datos, este estado dura 5 segundos y el LED interno parpadea en color azul durante estos 5 segundos.
- **Recoger datos:** Después de transcurrir los 5 segundos, comienza el estado de recogida de datos, en el cual se ejecutan la función lecturaVALORES(), la cual obtiene los datos de la IMU y del acelerómetro analógico.
Estos valores obtenidos se van almacenando en un buffer de uno en uno, es decir se realiza una lectura y se almacena, así en cada iteración del bucle. Este estado se terminará cuando se llegue al valor final del buffer, es decir cuando se llena. El tamaño está definido en la declaración de variables y es diferente para los dos modos de funcionamiento.
Durante todo este estado está encendido el LED verde, para indicar que se están recogiendo los datos.
- **Lectura y envío de la temperatura:** Se lee la temperatura mediante la función lecturaTEMP() y posteriormente se realiza el envío.
- **Envío de los datos de aceleración:** En este estado se lee el buffer donde se ha almacenado cada dato y se asigna a una variable dentro de una estructura tipo struct, se hace con los datos de cada lectura y posteriormente se envían.
Después de cada envío se vacía la posición del buffer que ya ha sido leída.
El estado se termina cuando se leen todos los datos del buffer y se vacía entero.
Durante todo el envío se mantendrá encendido el LED rojo para indicar que se están enviando los datos.
Cuando se envían todos los datos se termina este bucle y se apaga el LED, manteniéndose a la espera que se vuelva a pulsar el botón para volver a empezar el loop.



6.2 Programación del dispositivo receptor

Este es el micro controlador que se encarga de recibir los datos enviados por el dispositivo de adquisición, el cual está conectado mediante USB al ordenador y mediante el puerto serie se envían los datos a la aplicación creada en LabVIEW.

6.2.1 Explicación del programa

Este es un programa sencillo por lo que no se recurrirá a un diagrama de flujo para su explicación.

Librerías utilizadas: “*esp_now.h*” y “*WiFi.h*”.

El funcionamiento consiste en leer los datos recibidos del otro micro controlador y mostrar por el monitor serie, aquellos datos que nos interesan para ser leídos por LabVIEW.

En el setup se realiza el encendido del wifi y la inicialización del protocolo esp-now, y la ejecución de la función que está a la espera de recibir los datos.

Dentro de esta función se ha hecho una división:

- Cuando se realiza el envío de la temperatura, solo se muestra por el monitor serie el valor de la temperatura, y el resto de datos es el valor 1000, siendo un número arbitrario.
- Cuando no se realiza el envío de la temperatura se leen los datos de manera normal y se muestran por el monitor serie.

Para indicar al programa si se está enviando o no la temperatura, existe una variable que se envía desde el dispositivo de adquisición en la que cuando se envía y tiene un 1 indica que se está enviando la temperatura y cuando tiene un 0 se están enviando los datos de aceleración.

También existirán dos programas diferentes para cada modo de funcionamiento, pero en este caso la única diferencia será el bloque de datos que va a recibir, siendo los siguientes;



```
typedef struct valores {  
    float acx;  
    float acy;  
    float acz;  
    float gyrox;  
    float gyroy;  
    float gyroz;  
    float temp;  
    float ac500g;  
    bool var_temperatura;  
};  
valores MPUdata;
```

Ilustración 28: Bloque de datos modo de funcionamiento 1

```
typedef struct valores {  
    float temp;  
    float ac500g;  
    bool var_temperatura;  
};  
valores MPUdata;
```

Ilustración 27: Bloque de datos modo de funcionamiento 2

Después esos datos son los que se envían al monitor serie a excepción del dato var_temperatura que es el que indica si se envían la temperatura o los datos.

Los códigos de arduino de los 4 programas se pueden ver en el Anexo nº1 Códigos de los micro controladores.

7. APLICACIÓN PARA EL TRATAMIENTO Y VISUALIZACIÓN DE LOS DATOS

Como se explicó en el apartado de “*Descripción y Justificación de la solución adoptada*”, se escogió el software LabVIEW para realizar la aplicación para el tratamiento de los datos, a continuación, se mostrarán los diferentes programas que se han creado con una breve explicación.

Se han creado dos programas diferentes:

- Uno para la recepción y almacenamiento de los datos en ficheros.
- Otro para la visualización y tratamiento de los datos almacenados en los ficheros.

Existen dos programas diferentes para cada modo de funcionamiento, en los que lo único que cambia es el número de datos con los que se trabaja, ya que la IMU no está en el segundo modo de funcionamiento.

7.1 Programa de recepción y almacenamiento de datos en ficheros

Este programa tiene como misión recibir los datos que llegan por el puerto serie desde el micro controlador receptor y posteriormente almacenar todas las tramas de datos de un golpe en un fichero para posteriormente ser analizados.

Primero es necesario configurar el puerto serie de la misma forma que esté configurado el puerto serie del programa del micro controlador: 9600 baudios.

Los datos de aceleraciones llegan en forma de string, los cuales están separados entre sí por un punto y coma.

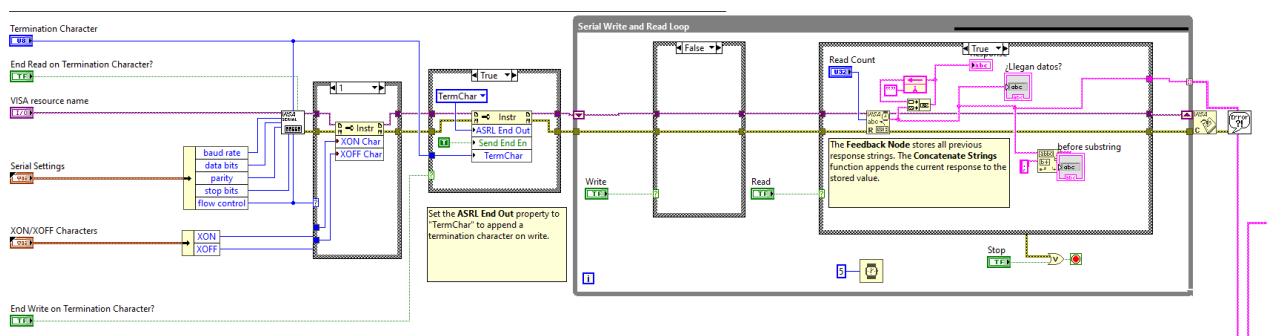


Ilustración 29: Parte de configuración del puerto serie

Como se explicó en el apartado anterior, el primer string que llega es el que indica la temperatura y para ello llega el dato de temperatura seguida del número 1000 varias veces separado por punto y coma.

Una vez se han recibido los datos, se separan todos y se convierten de formato string a formato numérico, para posteriormente volverlos a juntar en forma vectorial, de forma que cada trama de datos es un vector.

Estos vectores se almacenarán en un fichero de tipo .lvm que es el formato de LabVIEW para el almacenamiento de datos.

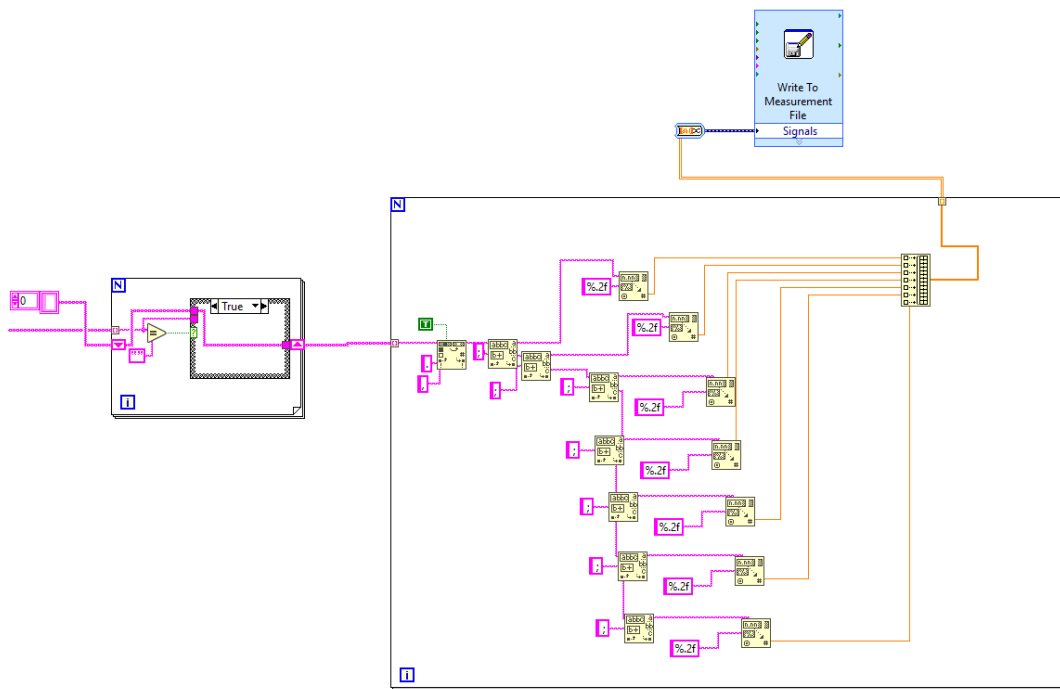


Ilustración 30: Parte de separar los datos y almacenarlos en un fichero

Esta imagen es para el modo de funcionamiento con la IMU incluida, para el acelerómetro analógico solo, como llegan menos datos hay menos módulos de separación de esos datos, con uno para el dato de la aceleración analógica es suficiente.

La visualización de la pantalla de aplicación es la siguiente:

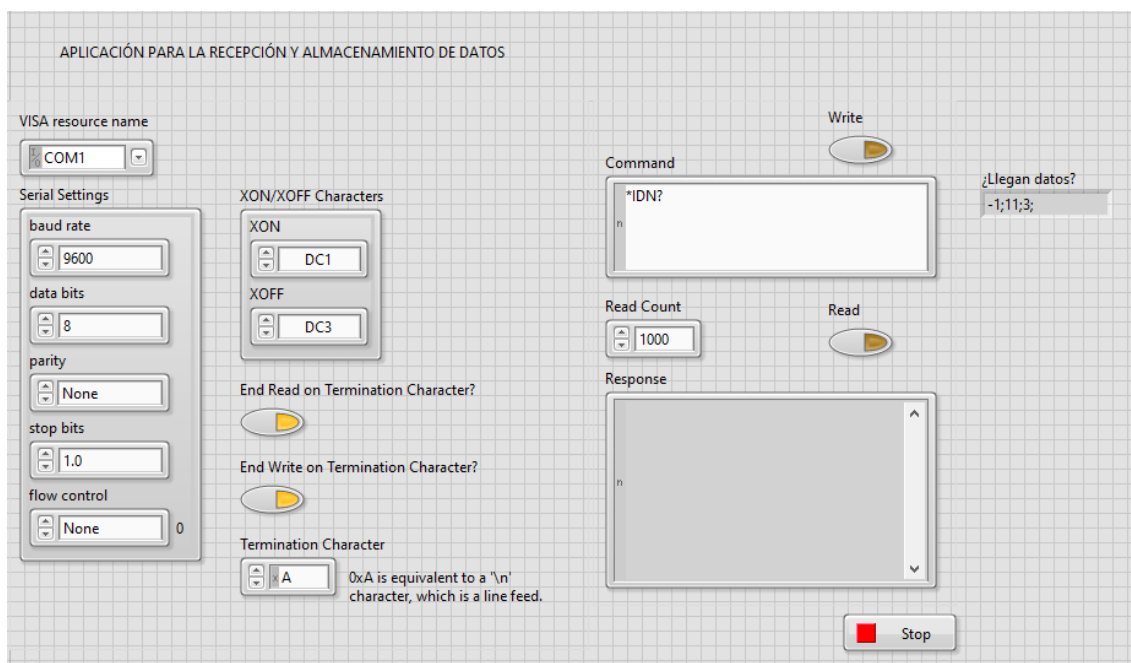


Ilustración 31: Pantalla de visualización

En la que se puede configurar el puerto serie, y se pueden ver los datos que están llegando.

7.2 Programa para la visualización y tratamiento de los datos

Esta aplicación tiene como objetivo poder visualizar los datos de los golpes que previamente se han almacenado en los diferentes ficheros, también se realizará el tratamiento de esos datos para obtener diferentes parámetros.

7.2.1 Parámetros a visualizar

Las gráficas que se pueden observar son las siguientes:

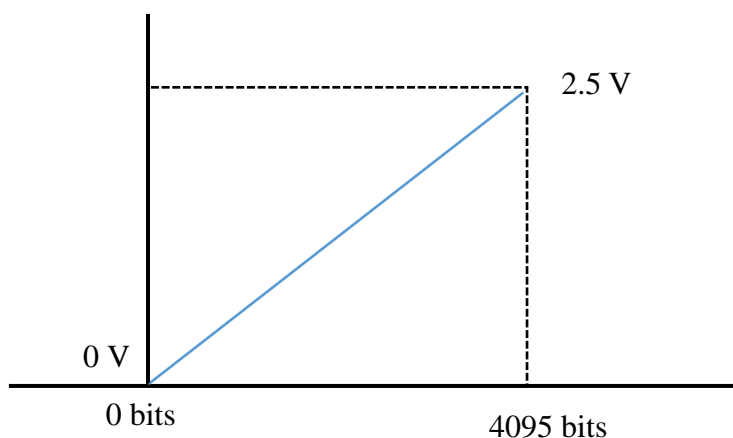
- Aceleración IMU eje X
- Aceleración IMU eje Y
- Aceleración IMU eje Z
- Aceleración 500g
- Velocidad angular eje X
- Velocidad angular eje Y
- Velocidad angular eje Z
- 4 espectros de frecuencias de todas las aceleraciones del ensayo
- Temperatura del ensayo

7.2.1.1 Escalado de la señal analógica

Cabe destacar que para graficar la aceleración del sensor analógico de 500g, hay que escalarla, puesto que llega en su valor en crudo y hay que pasarlo a su valor en g.

Para ello se pasan el valor de entrada del ADC a Voltaje, siendo la recta la siguiente:

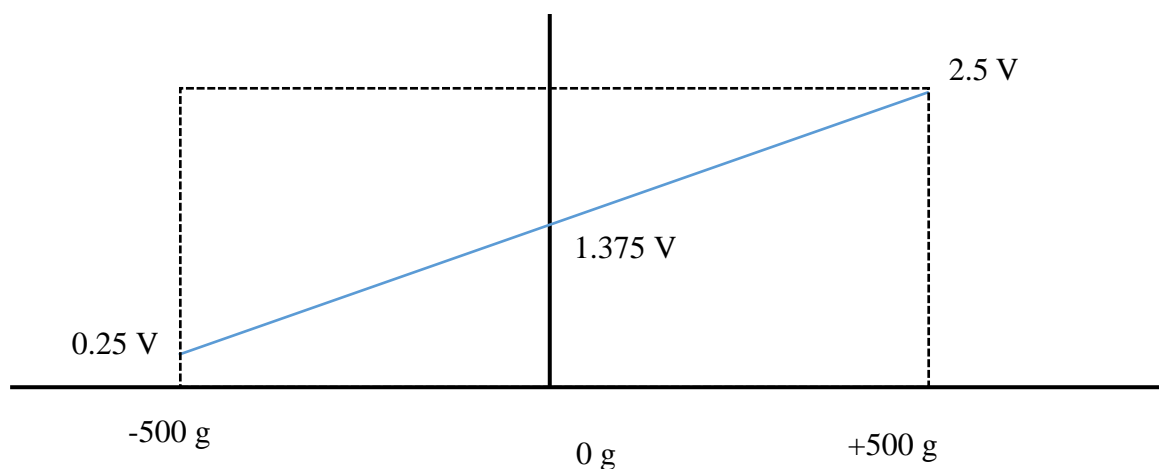
Teniendo los extremos y la pendiente se la recta.



Por lo tanto, para pasar a Voltios el valor es la siguiente ecuación:

$$ac500g (V) = \left(\frac{2.5 V}{4095} * aceleración\ en\ crudo \right)$$

Combinándolo con la recta de calibración del apartado 5.2.1 *Conexión del acelerómetro analógico*:



Se obtiene la siguiente ecuación para escalar la aceleración:

$$ac500g = \frac{\left(\frac{2.5 V}{4095} * \text{aceleración en crudo}\right) - 1.375 V}{0.002 V/g}$$

7.2.2 Tratamiento de datos

Los parámetros que se calcularán con los datos son los siguientes:

- Energía generada durante el golpeo

$$\text{Energía} = \sum a^2 * dt$$

- Módulo de aceleraciones en el momento del golpe

$$\text{módulo aceleración golpe} = \sqrt{ax^2 + ay^2 + az^2}$$

- Módulo de velocidades angulares en el momento del golpe

$$\text{módulo velocidad golpe} = \sqrt{vx^2 + vy^2 + vz^2}$$

- Armónicos de frecuencia de las ondas de aceleración
- Aceleraciones máximas
- Aceleraciones mínimas

Cada ensayo adquiere una gran cantidad de datos, siendo muchos innecesarios por ello mediante la aplicación creada se recortan los datos para centrarse solo en el golpe.

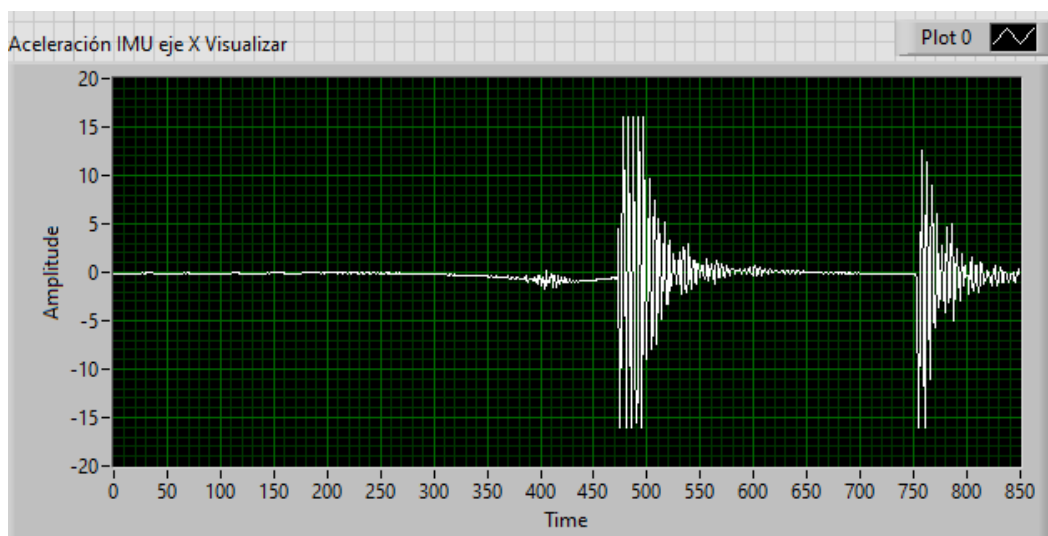


Ilustración 32: Conjunto total de datos

Introduciendo desde que punto a qué punto se quieren realizar los cálculos para el golpeo:

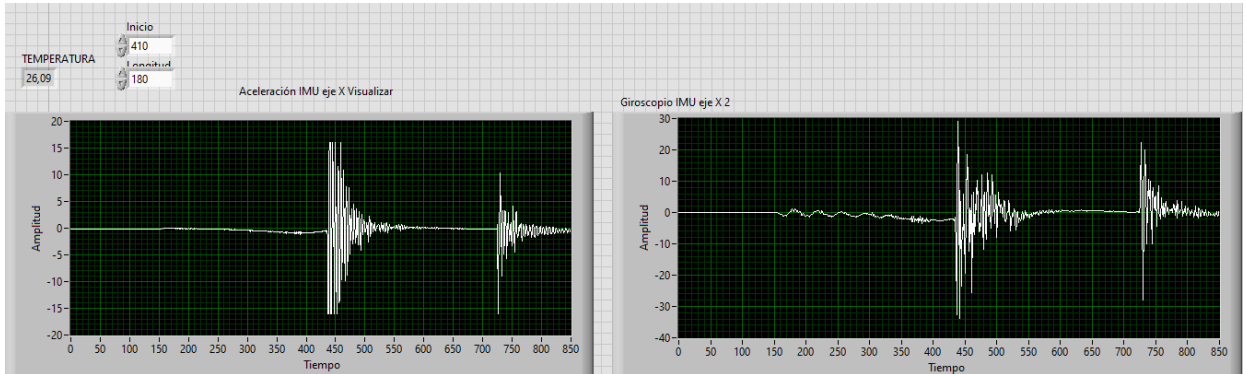


Ilustración 33: Conjunto de datos recortados

La aplicación tiene el siguiente aspecto:

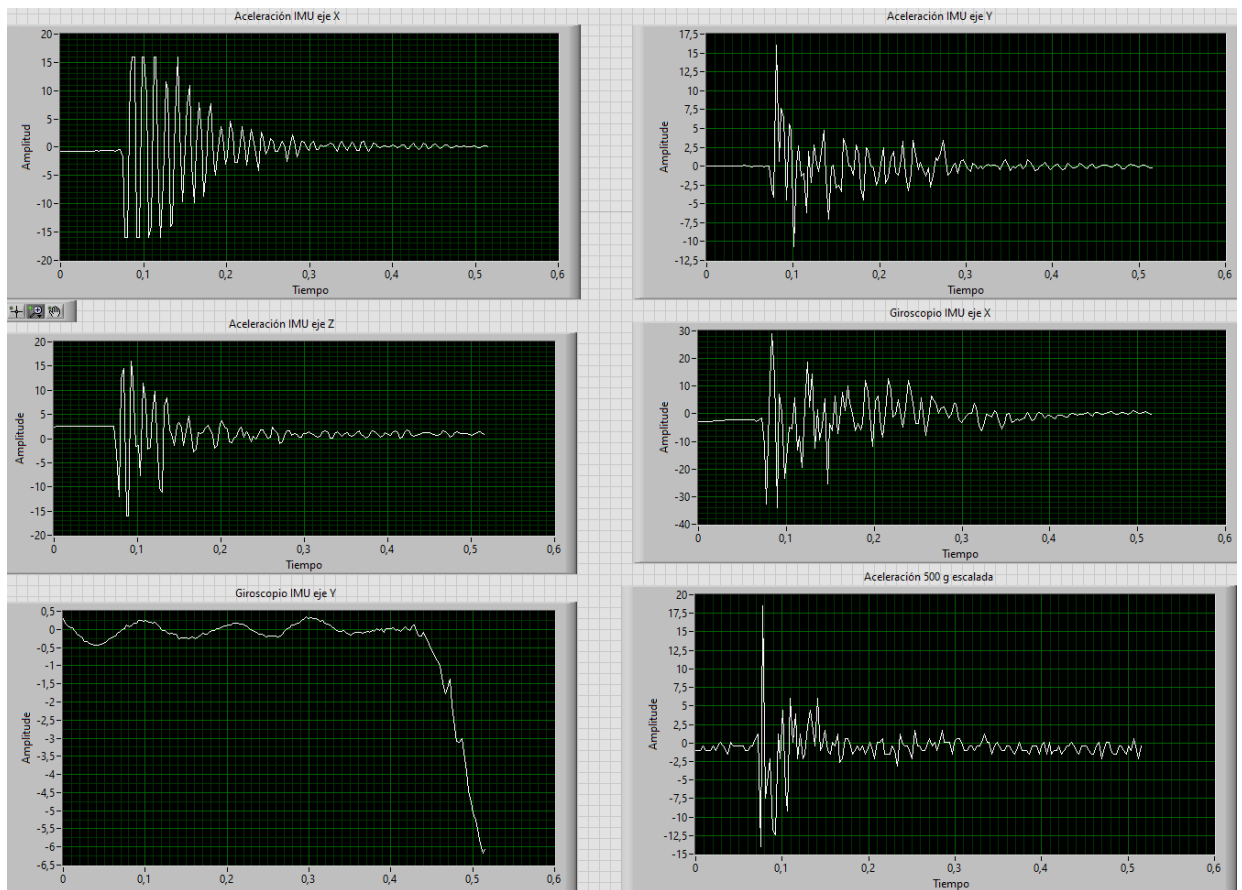


Ilustración 34: Gráficas de muestra de los valores

Aquí se pueden observar algunas de las gráficas interesantes a la hora de estudiar cada golpeo.

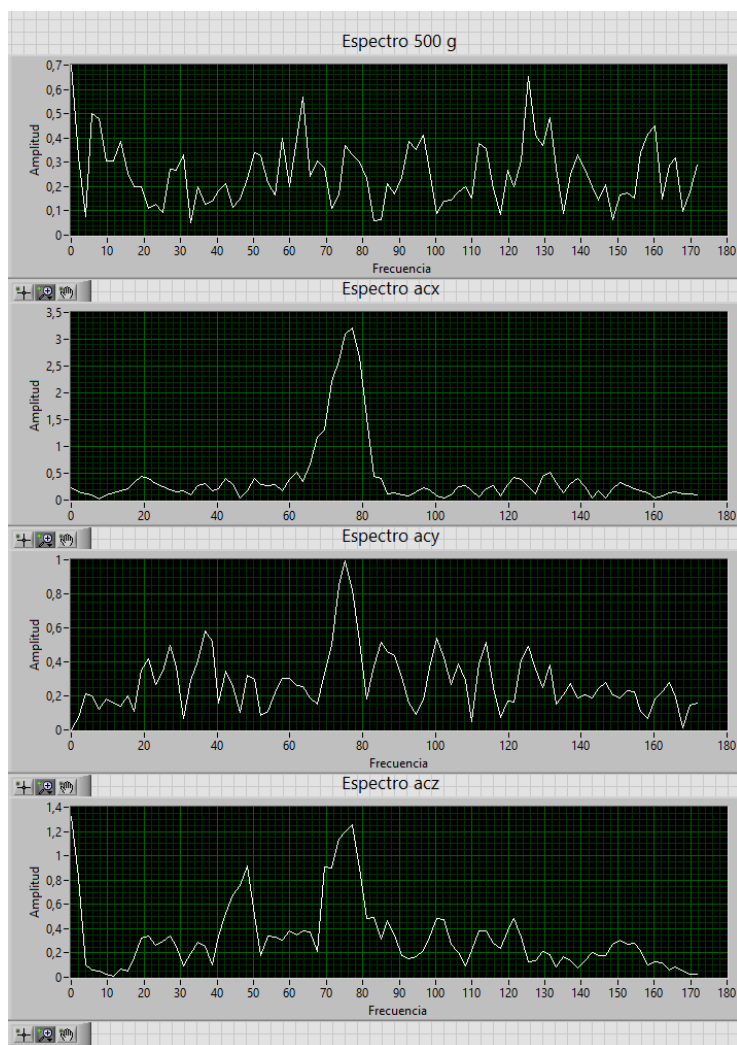


Ilustración 35: Espectros de las ondas

En esta imagen se observan las gráficas de los espectros.

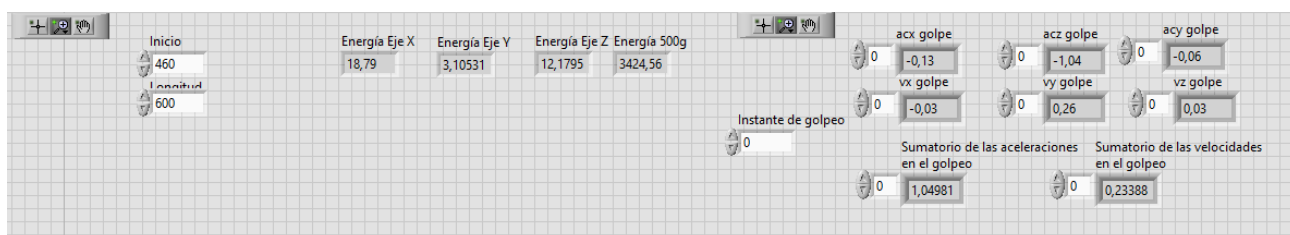


Ilustración 36: Valores calculados en el tratamiento de datos

En esta imagen se observan algunos de los datos calculados.

Una vez terminada la aplicación de tratamiento de datos, el siguiente paso es realizar los ensayos pertinentes para realizar las comparaciones entre diferentes palas y diferentes golpes.



Los datos adquiridos mediante esta aplicación se irán apuntado en un Documento Excel para su posterior estudio.

Existen dos aplicaciones de visualización y tratamiento de datos, se ha explicado la principal, para la recogida de datos con la IMU más el sensor analógico, la otra será para el modo de funcionamiento solo con el sensor analógico, siendo la diferencia que hay menos datos que tratar y visualizar.



8. ENSAYOS Y MEDIDAS REALIZADAS

Una vez se tiene el dispositivo montado y programado y la aplicación de tratamiento de los datos funcionando correctamente, el siguiente paso es la realización de los ensayos y medidas para recoger los datos de cada golpeo.

Cabe destacar que la mayoría de medidas se realizaron con el segundo prototipo ya que se obtenían medidas correctas, aunque e fuese más tedioso, y las últimas ya se realizaron con el dispositivo final.

8.1 Tipos de ensayos realizados

Con cada pala se han realizado diferentes tipos de ensayos que se explican en este apartado:

8.1.1 Ensayo en estación de salida de bola

Este es el tipo de ensayo que se realiza dentro del laboratorio de pádel del ai2 en colaboración con la empresa testea pádel.

Consiste en un péndulo en el que se sujeta el mango de la pala, sujetando así toda la pala.

Este péndulo se levanta hasta ponerse en posición casi vertical, es decir con la superficie de golpeo de la pala hacia arriba. Dejándose caer por su propio peso.

Al caer golpea con otro péndulo que está justo en el centro del ensayo, el cuál en su extremo tiene una pelota de pádel, al caer el péndulo con la pala y golpear la pelota mueve el otro péndulo en dirección contraria, como si de un golpe de pádel se tratase.

Para la realización de este ensayo los dos acelerómetros se pegan con silicona caliente a la base del mango de la pala, estos están conectados mediante un cable al micro controlador, el cual envía los datos al micro controlador conectado al ordenador.

Las ventajas de esta estación es que todos los golpes serán prácticamente iguales ya que una vez que la pala queda bien anclada en el péndulo el punto de golpeo siempre será el mismo para diferentes ensayos.

Se puede ver gráficamente en las siguientes imágenes.



Ilustración 37: Estación de salida de bola

8.1.2 Ensayo en pared

Este ensayo consiste en realizar un golpeo de la pala con la bola, lo más parecido posible al golpeo de pádel, se realiza en los exteriores del laboratorio de pádel del ai2, y al igual que en el ensayo de la estación de salida de bola los sensores van pegados al mango de la pala, conectados mediante un cable al micro controlador.

Algún ejemplo gráfico de este ensayo sería el siguiente:



Ilustración 38: Ensayo en pared con el prototipo final



Consideraciones a tener en cuenta a la hora de realizar los ensayos

Para realizar cada ensayo habrá que tener en cuenta una serie de consideraciones para intentar que las medidas sean lo más fiables posibles:

- Al principio de cada medida siempre se envía la temperatura.
- Es necesario pegar con silicona caliente los sensores a la base del mango de la pala quedando lo más centrado posible, y siempre tienen que estar los ejes en la misma dirección, para ello se puede ver en los dibujos de la IMU.
- Dentro de cada tipo de ensayo (estación y pared) existen otros dos tipos de ensayo, uno para cada modo de funcionamiento, que se explicó anteriormente, con IMU y sin la IMU, por lo tanto, para cada pala se realizarán los siguientes ensayos:
 - Ensayo en estación con IMU y acelerómetro analógico
 - Ensayo en estación solo con acelerómetro analógico
 - Ensayo en pared con IMU y acelerómetro analógico
 - Ensayo en pared solo con acelerómetro analógico

Siendo lo único que varía entre los ensayos con IMU y sin IMU el tiempo que tardan en enviarse los mensajes del micro principal al receptor, ya que los ensayos en el que solo está el acelerómetro analógico se envía una cantidad de datos mucho más elevada puesto que el tiempo de muestreo es mucho menor.

- Cuando se realiza el montaje de los acelerómetros en una pala, lo ideal es realizar todos los ensayos sin modificar ese montaje para intentar que los datos salgan lo mejor posible.

8.2 Palas utilizadas en los ensayos

Los ensayos se han realizado usando palas diferentes para así poder estudiar correlaciones y diferencias en los datos obtenidos, las palas con las que se han realizado son:

ADIDAS Adipower Attk 1.9



Ilustración 39: Adidas Adipower Attk1.9

Artengo PR190



Ilustración 40: Pala Artengo PR190

Kelme Olímpica



Ilustración 41: Pala Kelme olímpica

8.3 Metodología de realización de cada ensayo

A continuación, se explica brevemente la metodología que se ha seguido para la realización de un ensayo completo, es decir para obtener todos los datos del golpeo de cada pala.

8.3.1 Estación de salida de bola

- **Colocación de los sensores en la pala:** Una vez se ha elegido la pala con la que se van a realizar los ensayos se colocan los dos acelerómetros en la pala mediante el uso de silicona con una pistola termofusible.

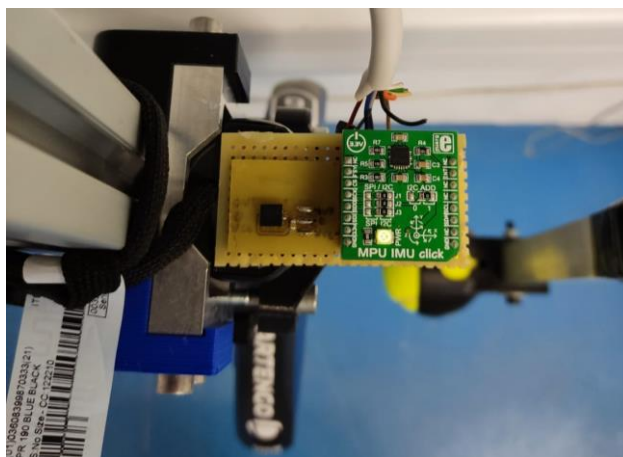


Ilustración 42: Colocación de los sensores en la pàña

- **Colocación de la pala en la estación de salida de bola:** Se coloca la pala en el péndulo de la estación de salida de bola de forma que, al caer, la bola golpee lo más centrada posible en el punto dulce de la pala.



Ilustración 43: Pala colocada en la estación de golpeo

- **Preparación y puesta en marcha del software de adquisición:** Se empieza cargando en los micro controladores, el programa de adquisición de IMU más acelerómetro analógico y en el receptor el correspondiente de recepción. Y se prepara la aplicación de LabVIEW para el almacenamiento de datos.



Ilustración 44: Estación preparada para medir

- **Preparación del ensayo:** Se coloca la pala en la posición extremo del péndulo, es decir lo más vertical posible, cuando toca con el final de carrera.



Ilustración 45: Preparación de la pala para el ensayo

- **Golpeo y adquisición:** Se presiona el botón de adquisición del prototipo, mientras el LED está parpadeando, se selecciona en la aplicación LabVIEW el modo lectura y se inicia el programa. Cuando deja de parpadear y se enciende en color verde inmediatamente se suelta la pala para que se realice el golpeo con la bola.
- **Recepción de los datos:** Se envían los datos del micro controlador de adquisición al de recepción, importante no desconectar nada durante ese proceso, termina cuando el LED rojo está encendido.
- **Cambio de modo de funcionamiento a solo acelerómetro analógico:** Después se cambian los programas de los micro controladores y las aplicaciones de LabVIEW, y se cargan los que solo se adquiere datos del acelerómetro analógico, para el muestreo más rápido. Y se repite el resto de pasos de manera idéntica.

Se realizan cinco adquisiciones para cada modo de funcionamiento.

8.3.2 Golpeo en pared

Sin quitar los sensores del mango de la pala, y intentando que se muevan lo mínimo posible lo siguiente es las pruebas de golpeo en la pared.

- **Preparación para el golpeo:** Se coloca el ordenador fuera del laboratorio, enfrente de la pared en una posición en la que se pueda interactuar por ejemplo en una silla, para comenzar pulsar los botones de comenzar a medir y observar si funciona todo, se coge la pala de tal forma que se pueda realizar un golpeo de abajo arriba, lo más similar posible a la estación de salida de bola, puesto que el objetivo es comparar estos dos golpes.
- **Preparación y puesta en marcha del software de adquisición:** Al igual que en el ensayo anterior primero se carga el programa de adquisición de la IMU y el acelerómetro analógico, y se prepara la aplicación de LabVIEW.
- **Preparación del golpe:** Se coge la pala de tal forma que se pueda realizar un golpeo natural de pádel.



Ilustración 46: Preparación del golpeo

- **Golpeo y adquisición:** Se pulsa el botón de adquisición y mientras el LED parpadea se pone la aplicación en modo recibir datos. Cuando deja de parpadear se realiza el golpeo.

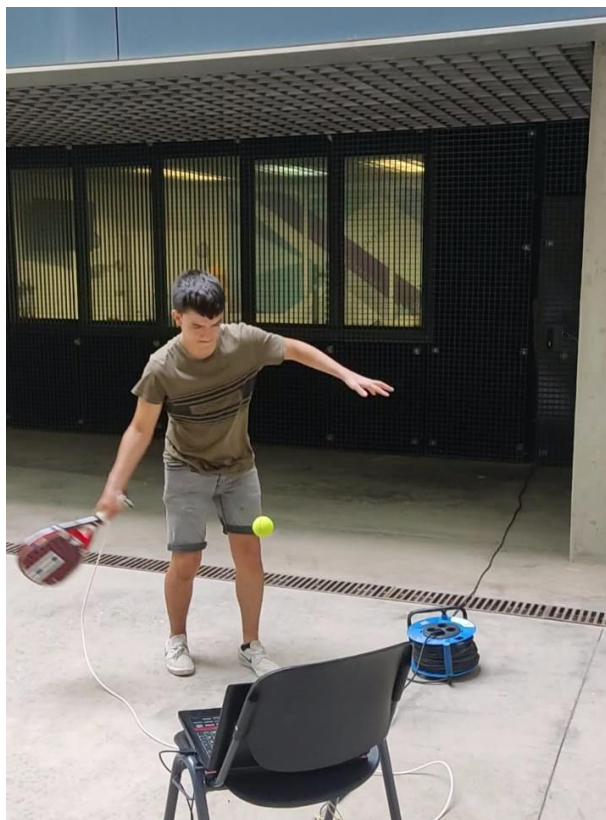


Ilustración 47: Golpeo ensayo en pared

- **Recepción de os datos:** Cuando se han adquirido todos los datos se envían de un micro controlador al otro, que posteriormente serán leídos por la aplicación.
- **Cambio de modo de funcionamiento a solo acelerómetro analógico:** Al igual que en el proceso anterior, se cambian los programas de los micro controladores y las aplicaciones de LabVIEW y se repite el proceso de manera idéntica.

Al igual que en la estación, como mínimo se realizan cinco adquisiciones de datos válidas para cada modo de funcionamiento.

Cuando ya se han realizado los dos tipos de ensayos y los dos modos de funcionamiento de cada uno, se puede decir que se han adquirido todos los datos que se tenían como objetivo para luego ser estudiados y comparados.



9. ANÁLISIS DE RESULTADOS

En este apartado se va a realizar el análisis de todos los datos que se han obtenido mediante los diferentes ensayos, los datos más importantes se han ido almacenando en diferentes documentos de Excel para cada pala y así poder establecer correlaciones, los datos tenidos en cuenta son los siguientes:

Ensayos con IMU y acelerómetro analógico

- Aceleración en cada eje en el golpe
- Aceleración máxima en cada eje en el ensayo
- Velocidad angular en cada eje en el golpe
- Velocidad angular en cada eje máxima en el ensayo
- Energía en cada eje de cada ensayo
- Energía/Velocidad angular en cada eje
- Módulo de aceleraciones en el golpeo
- Módulo de velocidades angulares en el golpeo
- Temperatura
- Armónicos de frecuencia en el golpeo

Ensayos solo con acelerómetro analógico

- Aceleración máxima del ensayo
- Aceleración mínima del ensayo
- Energía en el ensayo
- Temperatura
- Armónicos de frecuencia

Una vez recopilados todos los datos el objetivo es estudiar correlaciones en los diferentes ensayos.

Todos los datos se recopilan en tablas de Excel, las cuales se van rellenando según los datos obtenidos, la apariencia de las tablas que se rellenan es la siguiente:

Ensayo nº 1135												
Aceleración (g)			Golpe		Máxima		Energía		Módulo de aceleraciones golpes		Temperatura	
x	-0,6	16	x	14,3789			2,72685		24,31			
y	0,01	9,86	y	1,49513								
z	2,66	16	z	11,7535			Módulo de velocidades golpeo					
500 g	-1,61256	59,1662	500 g	14,3948			5,90269					
Velocidad			Golpe		Máxima		Energía/ velocidad angula		Frecuencias			
x	0,62	10,03	x	23,19177419	x	75,25						
y	0,02	0,33	y	74,7565	y	81,1						
z	5,87	25,72	z	2,00229983	z	0	138,98					
			500g		23,21741935		500 g		133,2			

Ilustración 48: Tabla de datos IMU Y acelerómetro analógico

Ensayo nº 1202						
Aceleración (g)		Mínima	Máxima	Energía		Temperatura
500 g		-73,1074	103,327	500 g 134,118		27,35
				Frecuencia		
				500 g		45,1 50,2

Ilustración 49: Tabla de datos solo acelerómetro analógico

Antes de comenzar a rellenar las tablas, se hace un estudio de la forma de onda resultante del golpeo.

No todas las ondas tienen la misma forma, por lo que es muy importante que para que los resultados tengan correlación entre sí elegir los golpes que generen gráficas con forma de onda similar.

Esto ha sido posible en las pruebas realizadas en la estación de salida de bola, ya que, en las pruebas realizadas en pared, han salido ondas con formas de onda bastante diferentes, esto es debido porque el golpeo en exterior no es siempre idéntico, cada vez se emplea una fuerza, una posición de golpeo y diferentes factores que afectarán a la forma de onda del golpeo.

9.1 Estudio y correlación de diferentes palas

Una vez se han reunido todos los datos de cada golpe en las tablas, se ha buscado en cuales se tiene una uniformidad para así poderse luego relacionar con otros ensayos.

Elegidos estos parámetros se ha realizado la media de los datos obtenidos para sacar conclusiones en función de la pala.

También se han calculado las desviaciones típicas de cada ensayo para ver así la repetibilidad de los mismos.

9.1.1 Resultados de los ensayos de la IMU y del acelerómetro analógico en la estación de salida de bola

ARTENGO PR190

Media de datos									
Aceleración (g)			Energía		Módulo de aceleraciones golpeo			Temperatura	
x	-0,632	16	x	14,66114	2,647436			24,5	
y	-	9,274	y	1,598322					
z	2,574	16	z	10,48466	Módulo de velocidades golpeo				
500 g	-	-	500 g	-	-				
Velocidad			Energía/ velocidad angula		Frecuencias				
x	0,784	11,938	x	-	x	74,9			
y	-	-	y	-	y	46,9375			
z	-	23,374	z	-	z	-			
			500g	-	500 g	-			

Ilustración 50: Datos finales pala Artengo

En los datos cuya representación es “-“, se debe a que no se ha establecido una relación uniforme en las diferentes pruebas.

Desviación típica de datos									
Aceleración (g)			Energía		Módulo de aceleraciones golpeo			Temperatura	
x	0,11077	0	x	0,870343012	0,870343012			0,137222447	
y	-	3,182449	y	0,368931007					
z	0,057706	0	z	1,384253744	Módulo de velocidades golpeo				
500 g	-	-	500 g	-	-				
Velocidad			Energía/ velocidad angula		Frecuencias				
x	0,366238	1,625783	x	-	x	1,126388			
y	-	-	y	-	y	15,52174			
z	-	1,86205	z	-	z	-			
			500g	-	500 g	-			

Ilustración 51: Desviaciones típicas pala Artengo



KELME OLÍMPICA

Media de datos									
Aceleración (g)			Energía		Módulo de aceleraciones golpeo		Temperatura		
x	-16	16	x	15,894675	20,819		25,4		
y	-3,555	12,6025	y	2,5837175					
z	-12,7725	16	z	9,7308375	Módulo de velocidades golpeo				
500 g	20,2415	20,784175	500 g	-	35,75635				
Velocidad			ergía/ velocidad angul		Frecuencias				
x	-33,6675	32,44	x	-	x	77,475			
y	-	0,2575	y	-	y	43,1			
z	-	59,7125	z	-	z	76,375			
			500g	-	500 g	127,1067			

Ilustración 52: Datos finales pala Kelme Olímpica

Desviación típica de datos									
Aceleración (g)			Energía		Módulo de aceleraciones golpeo		Temperatura		
x	0	0	x	0,5134951	0,574553583		0,506820152		
y	1,124885	3,95938021	y	0,3763438					
z	1,12787	0	z	1,3680969	Módulo de velocidades golpeo				
500 g	2,568871	3,64754084	500 g	-	7,783009154				
Velocidad			ergía/ velocidad angul		Frecuencias				
x	6,44972	3,11019292	x	-	x	1,141271			
y	-	0,06238322	y	-	y	16,13371			
z	-	4,70927011	z	-	z	0			
			500g	-	500 g	1,5115			

Ilustración 53: Desviación típica datos pala Kelme

ADIDAS ADIPOWER ATTK 1.9

Media de datos									
Aceleración (g)			Energía		Módulo de aceleraciones golpeo		Temperatura		
x	-	16	x	11,11562	-		24,7466667		
y	-0,6425	-	y	1,74846					
z	-	16	z	7,84471	Módulo de velocidades golpeo				
500 g	-	-	500 g	7,012508	-				
Velocidad			ergía/ velocidad angul		Frecuencias				
x	-	76,256	x	-	x	42,77			
y	-	-	y	-	y	36			
z	-	127,482	z	-	z	158,783			
			500g	-	500 g	-			

Ilustración 54: Datos finales pala Adidas

Desviación típica de datos									
Aceleración (g)			Energía		Módulo de aceleraciones golpeo		Temperatura		
x	-	0	x	0,1887367	-		0,02886751		
y	0,204022	-	y	0,4845502					
z	-	0	z	0,1389276	Módulo de velocidades golpeo				
500 g	-	-	500 g	2,7441143	-				
Velocidad			ergía/ velocidad angul		Frecuencias				
x	-	10,3711579	x	-	x	0,98209			
y	-	-	y	-	y	13,86651			
z	-	12,137214	z	-	z	69,55036			
			500g	-	500 g	-			

Ilustración 55: Desviación típica datos pala Adidas



Estos son los datos, haciendo la media de aquellos que siguen una línea, que se han obtenido de las tres palas en la estación de medida de salida de bola.

No en todos se ha obtenido correlación en los mismos datos.

Conclusiones

Primero, se ha observado que se obtiene bastante repetibilidad en los diferentes ensayos realizados en la estación ya que las desviaciones típicas son bajas respecto al valor medio, se ha visto calculando el cociente entre la desviación típica y el valor medio. Obteniendo que la mayoría de resultados están entre el 5 y el 10% de porcentaje de desviación. En algún caso se obtiene mayor porcentaje debido a que existe un error de medida de los sensores, que empeora la repetibilidad.

En la pala que mejores datos, más repetibles se obtuvieron fue en la Kelme, después en la Artengo y finalmente en la Adidas.

Las medidas que tienen mejor repetibilidad, las que menor desviación típica son las que se utilizan para estudiar luego las palas, debido a que los datos son fiables.

Se puede observar que:

- En todos los ensayos, el acelerómetro en el eje x y en el eje z satura, llega a 16 g que es el máximo rango del acelerómetro por lo que sería importante buscar algún acelerómetro triaxial con un rango mayor.
- Las velocidades angulares en el golpe son diferentes en todas las palas.
- El dato más importante es la energía que se libera en cada golpeo en las diferentes palas. Se puede ver, ante el mismo ensayo cuál es la pala que integra más energía. Llegando a las siguientes conclusiones:
 - La pala que más energía tiene es la Kelme, y por lo tanto la pala que más salida de bola tiene.
 - La segunda pala con más salida de bola es la Kelme con una salida de bola similar a la Artengo y la pala con menos salida de bola la Adidas.
- Se observa también que, en todos los ensayos, a la hora de estudiar la transformada de Fourier, comparten el mismo armónico de frecuencia en las ondas de aceleración:
 - Estos armónicos tienen similitud en la pala Artengo y Kelme ya que tienen una salida de bola parecida, siendo los armónicos:
 - Eje X: 74 (Artengo) y 77 (Kelme)
 - Eje Y: 40 (Artengo) y 43 (Kelme)
 - En la pala Adidas se han obtenido armónicos totalmente diferentes.

Se ha visto que en este caso utilizar el acelerómetro analógico junto con la IMU no ha generado unos resultados válidos puesto que la frecuencia de muestreo del acelerómetro analógico en este modo de funcionamiento no es suficiente para obtener unos datos fiables.

9.1.2 Resultados de los ensayos solo con el acelerómetro analógico en la estación de salida de bola

ARTENGO PR190

Media de datos						
Aceleración (g)	Mínima	Máxima		Energía		Temperatura
500 g	-101,4956667	66,6972		500 g	125,4501	27,3
					Frecuencia	
					500 g	50,44375

Ilustración 56: Datos finales para Artengo solo acelerómetro analógico

Desviación estándar de datos						
Aceleración (g)	Mínima	Máxima		Energía		Temperatura
500 g	13,68793447	3,951779		500 g	8,100369	0,054772256
					Frecuencia	
					500 g	0,37808

Ilustración 57: Desviación típica datos para Artengo

KELME OLÍMPICA

Media de datos						
Aceleración (g)	Mínima	Máxima		Energía		Temperatura
500 g	-45,5343	36,24843		500 g	93,43733	24,504
					Frecuencia	
					500 g	46,22383

Ilustración 58: Datos finales para Kelme solo acelerómetro analógico

Desviación típica de datos						
Aceleración (g)	Mínima	Máxima		Energía		Temperatura
500 g	10,99853	12,33661		500 g	21,64831	0,145830952
					Frecuencia	
				500 g	2,131017	

Ilustración 59: Desviación típica datos pala Kelme

ADIDAS ADIPOWER ATTK 1.9

Media de datos						
Aceleración (g)	Mínima	Máxima		Energía		Temperatura
500 g	-68,223	43,6513		500g	51,69183	24,4475
					Frecuencia	
				500 g	35,66613	

Ilustración 60: Datos finales pala Adidas solo acelerómetro analógico

Desviación típica de datos						
Aceleración (g)	Mínima	Máxima		Energía		Temperatura
500 g	25,696	4,359632		500g	11,7012	0,036855574
					Frecuencia	
				500 g	11,04899	

Ilustración 61: Desviación típica datos pala Adidas

Conclusiones

A la hora de utilizar el acelerómetro analógico únicamente se han obtenido las siguientes conclusiones:

- La pala que obtiene la máxima y la mínima aceleración es la Artengo.



- La pala que libera más energía en este modo de funcionamiento es la Artengo también, seguida de la Kelme y finalmente la Adidas.
- Al igual que en los ensayos con la IMU los armónicos de frecuencia son similares en la pala Artengo y Kelme ya que tienen similar salida de bola en 50 Hz Artengo y 46 Hz Kelme.

Los datos obtenidos en este ensayo se quedan cortos de información ya que solo se obtienen datos de un acelerómetro y en un eje por lo que se decide centrar la atención en los resultados obtenidos con la IMU ya que son más completos.

No son comparables estos datos con los de la IMU porque tienen diferentes rangos y diferente sensibilidad de medida, lo que hace que no se generen unos datos fiables para el estudio de los mismos.

9.1.3 Resultados de los ensayos realizados en pared

Dentro de este apartado se incluyen todos los ensayos realizados en pared, tanto de la IMU y acelerómetro analógico como solo del acelerómetro analógico.

Se ha llevado a cabo el mismo procedimiento que en los ensayos de la estación de salida de bola, pero después de estudiar los datos se ha visto ninguno de los parámetros sigue un patrón o una misma línea, por lo tanto no tienen nada de repetibilidad.

Esto es debido a que, al realizar el golpe, las condiciones no son idénticas, como si sucede en la estación de salida de bola. Cada vez se golpea con una fuerza, se mueve la pala de una forma y las condiciones no son idénticas.

Para ello habría que buscar un parámetro que relacione la velocidad angular de cada golpe con la aceleración y así intentar relacionarlo, se ha intentado dividir la aceleración entre la velocidad angular pero no ha dado un resultado adecuado.

9.2 Estudio del tipo de golpe

En este último apartado del análisis de resultados se estudia, qué tipo de golpe se ha realizado en función de la gráfica obtenida por el giróscopo, es decir la atención esta vez se centra más en el estudio de la forma de la gráfica que en los datos concretos de energía, velocidades y aceleraciones.

Para ello se han realizado diferentes pruebas golpeando a la bola de tres formas diferentes:

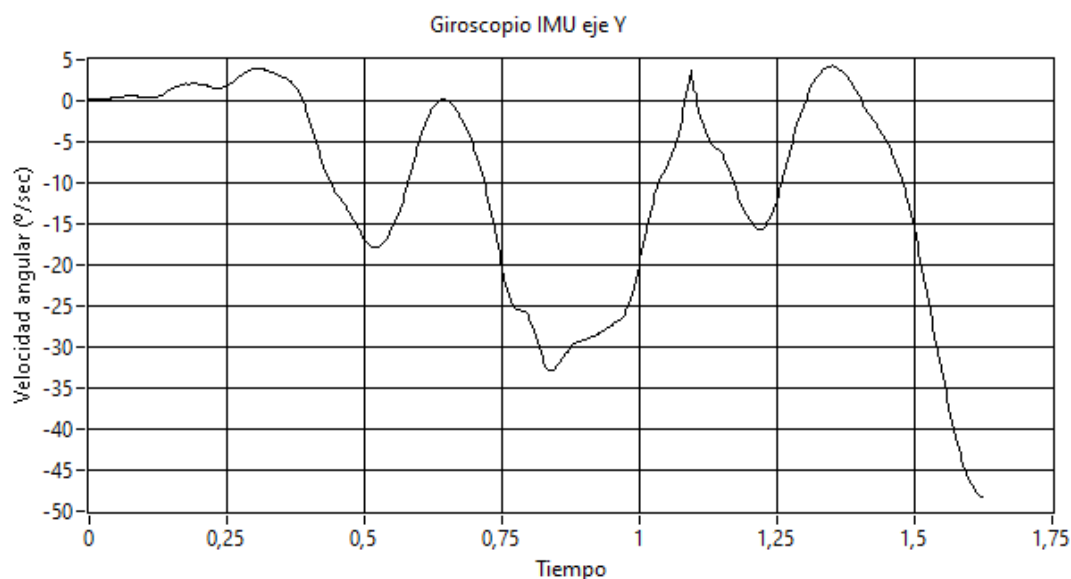
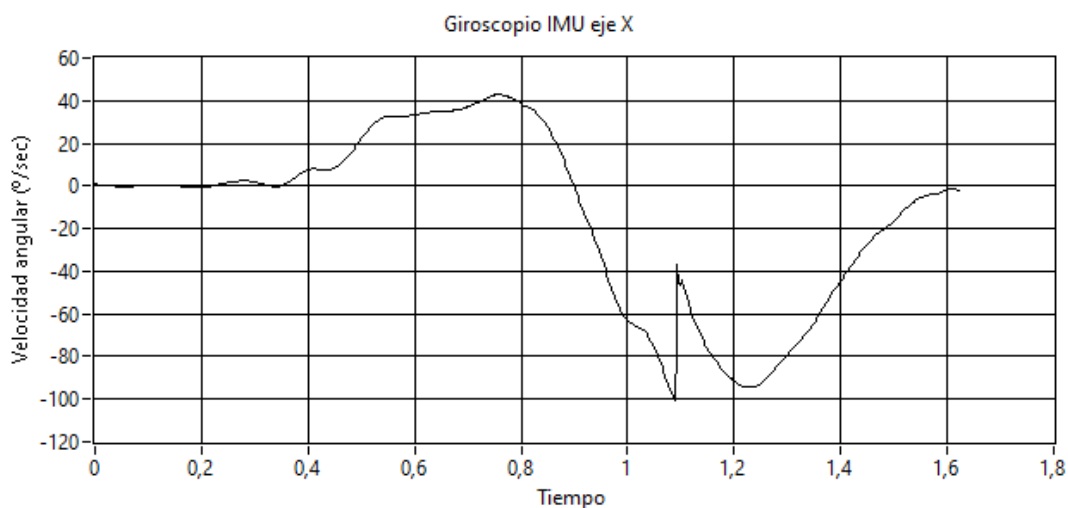
- **Golpe de derecha:** Es el golpe más común por los jugadores de pádel, se realiza por el mismo lado de la mano dominante del jugador, y la cara interior de la mano es la que apunta a la pelota a la hora de golpear.

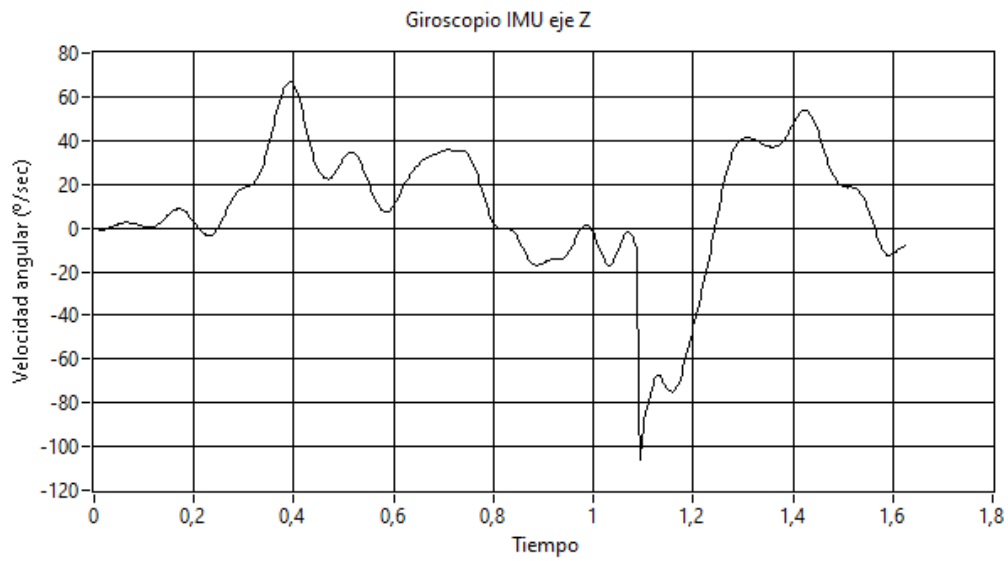
- **Golpe de Revés:** Es el segundo golpe más común en pádel, se realiza al contrario que el golpe de derecha, la pala bola se golpea con de forma que la cara exterior de la mano es la que apunta a la pelota a la hora de golpear.
- **Remate:** Se trata de un golpe ofensivo en que se golpea la pelota teniendo la pala en alto, y el golpeo se realiza de arriba abajo.

A continuación, se muestran las diferentes gráficas obtenidas en los diferentes golpes, como se comentó previamente las gráficas son de velocidad angular en los tres ejes.

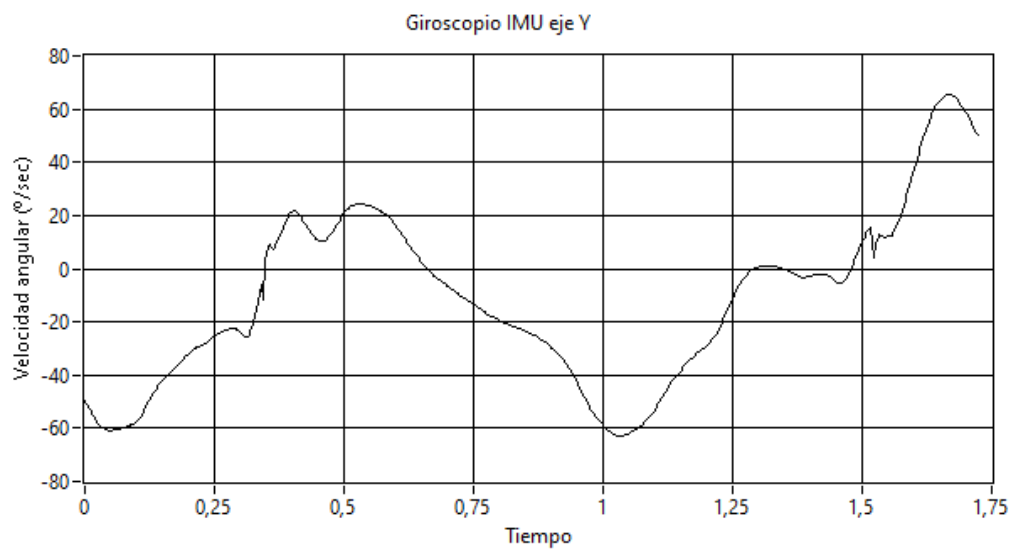
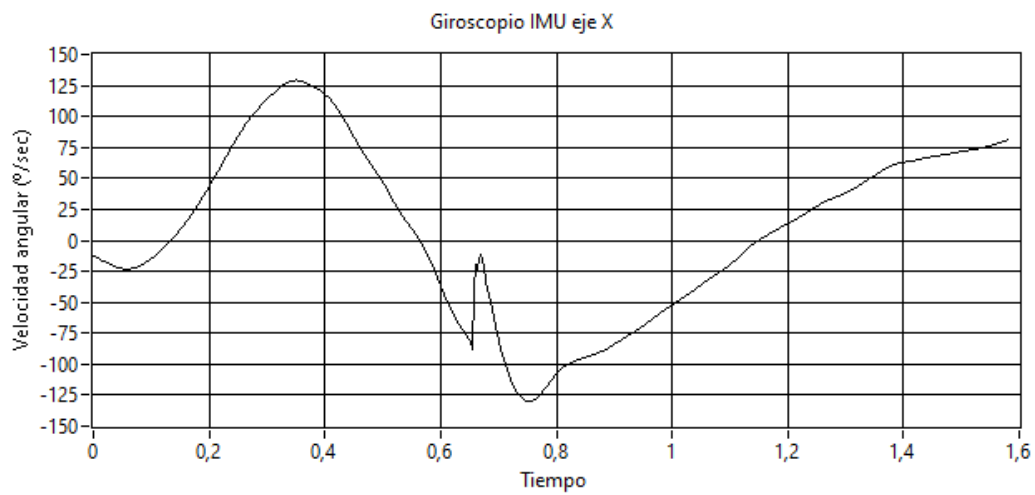
9.2.1 Gráficas golpes

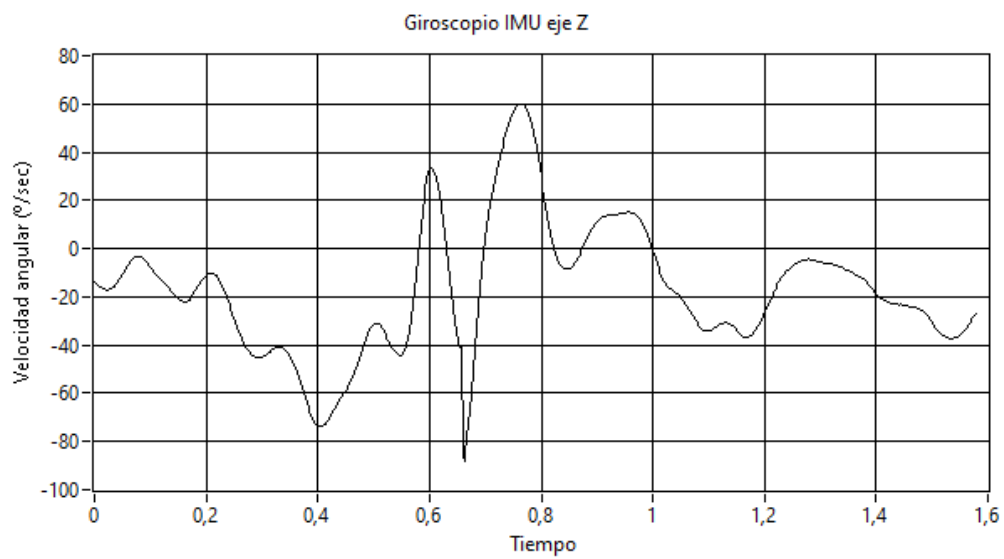
GOLPE 1 DERECHA



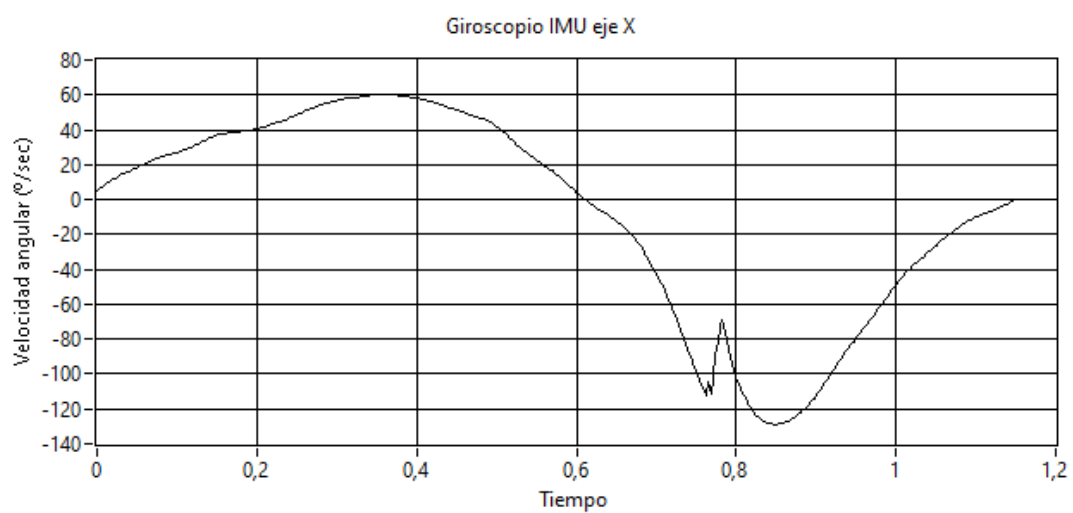


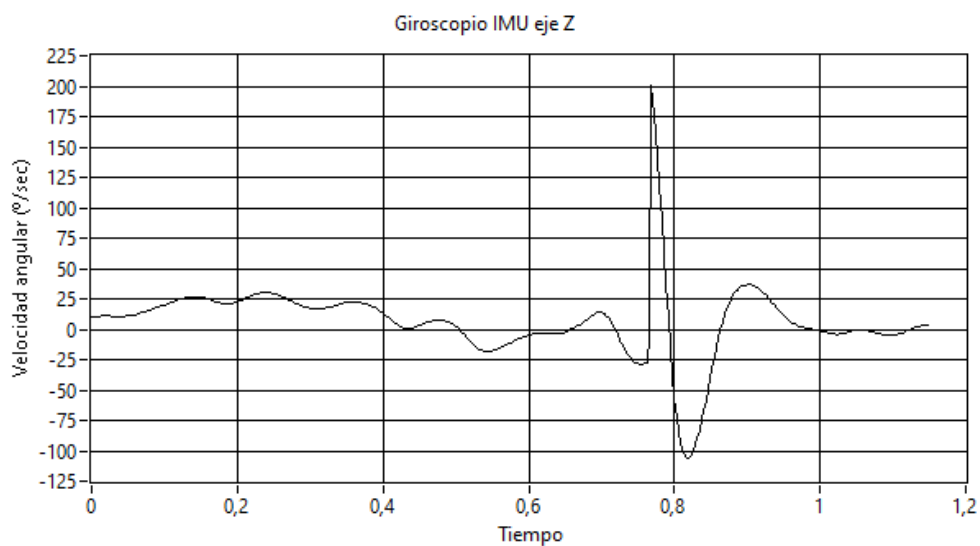
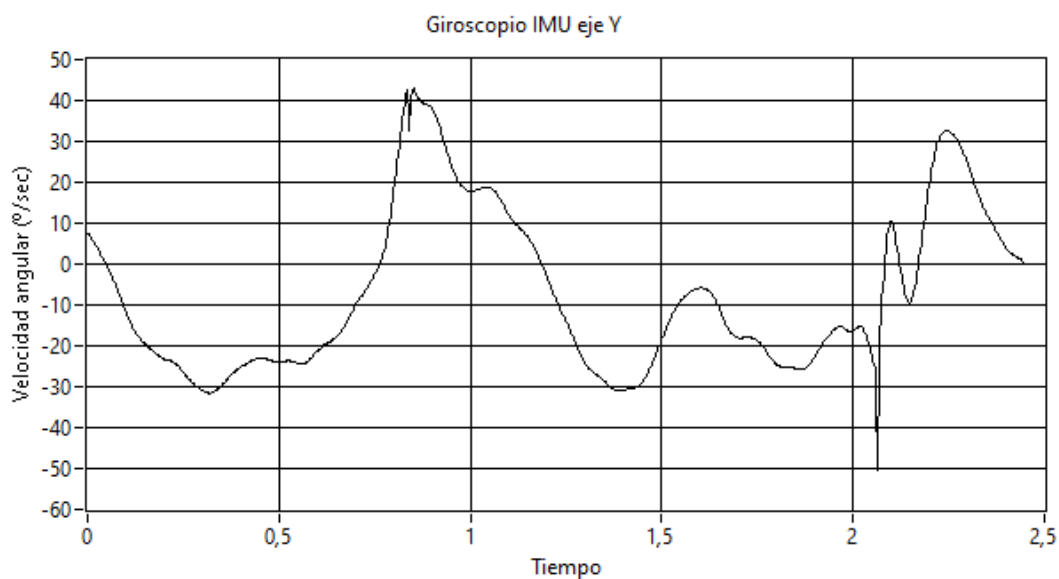
GOLPE 2 DERECHA



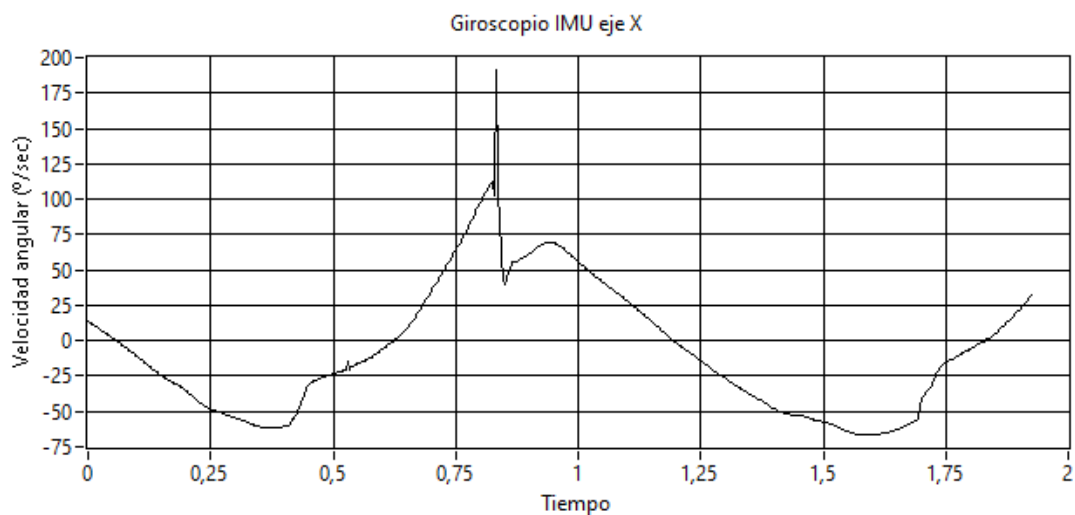


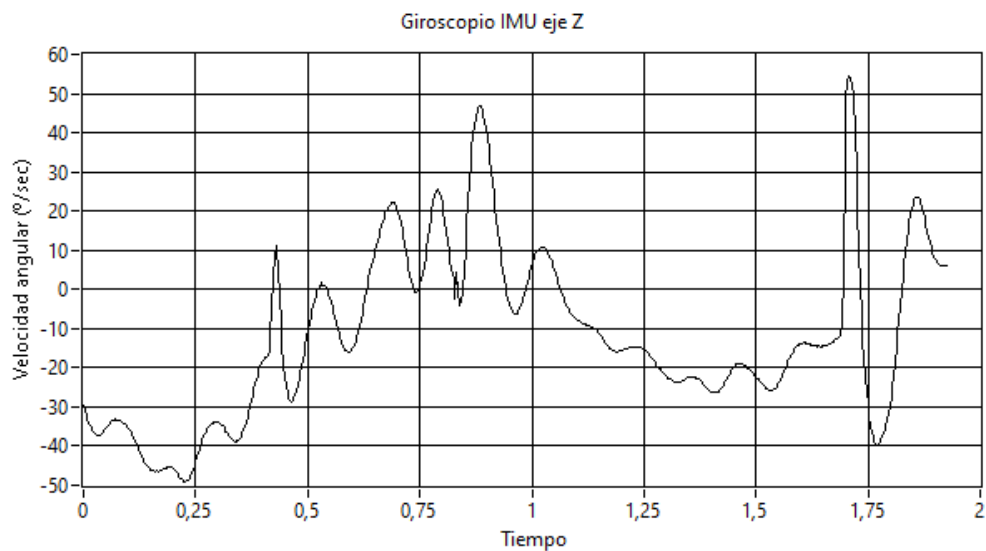
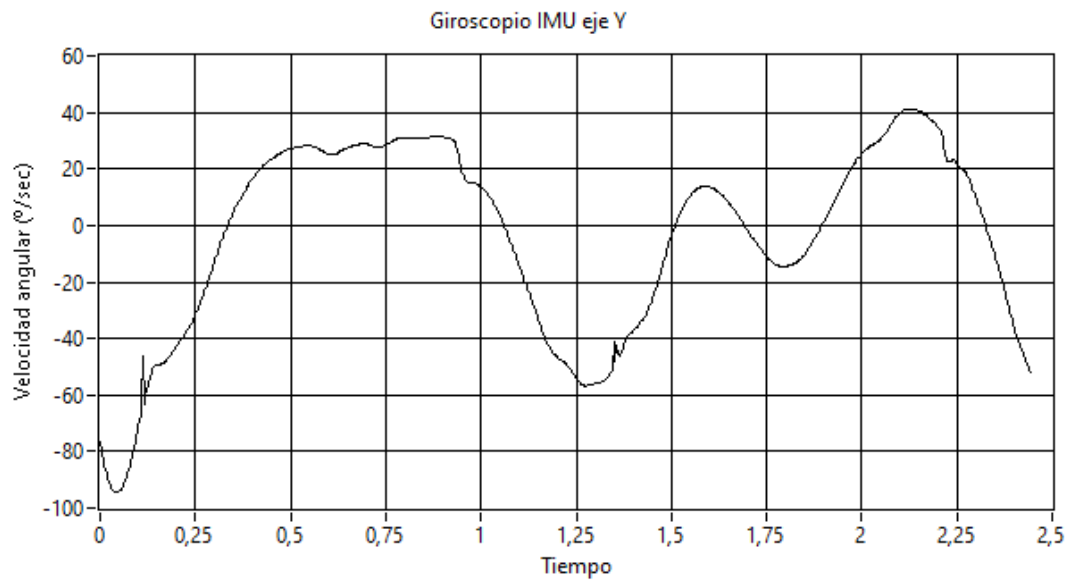
GOLPE 3 DERECHA



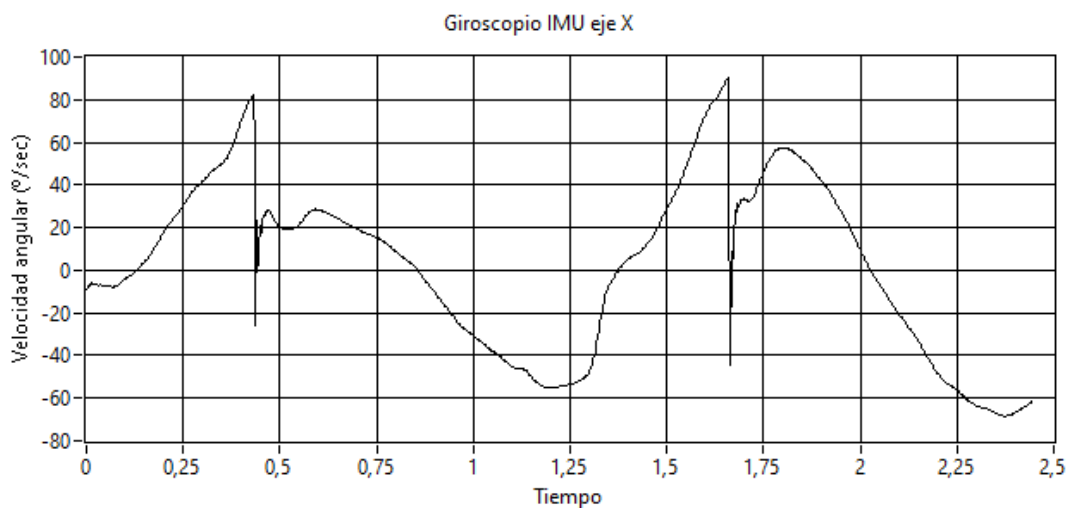


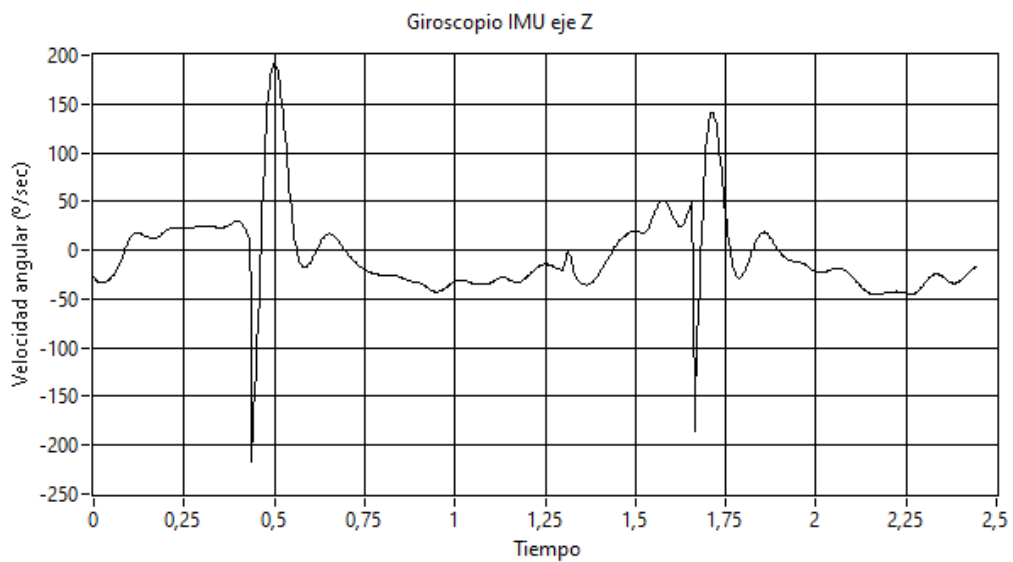
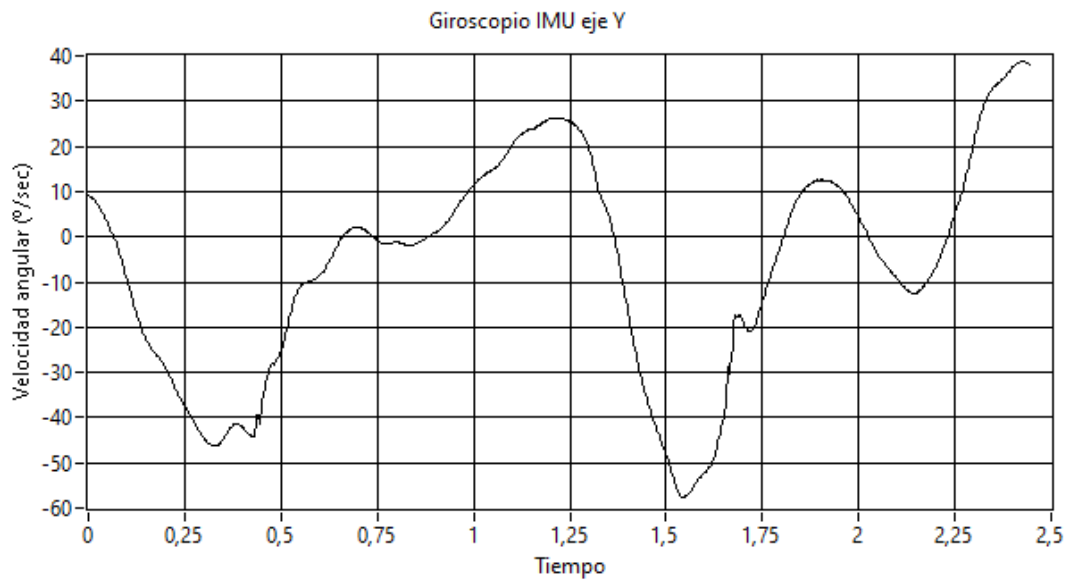
GOLPE 1 REVÉS



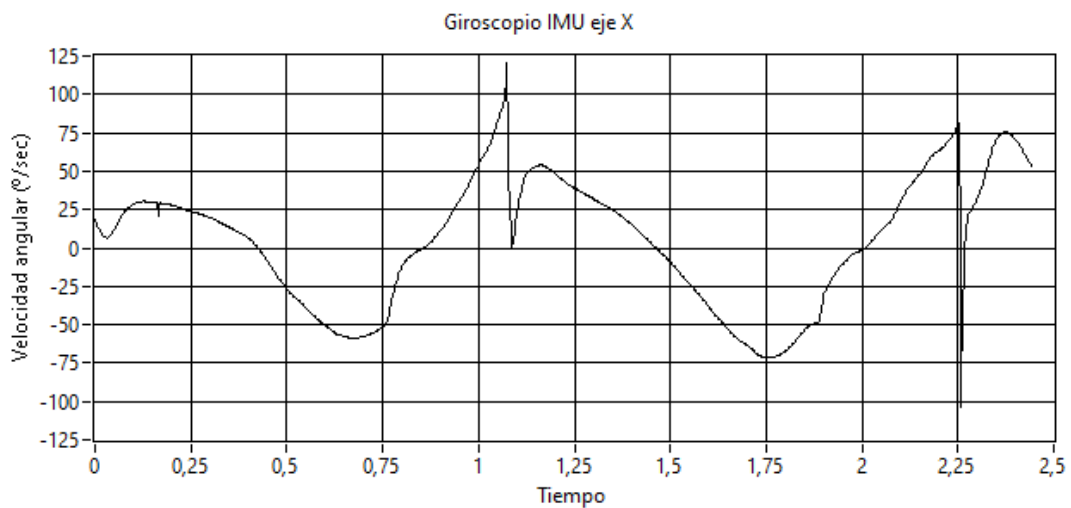


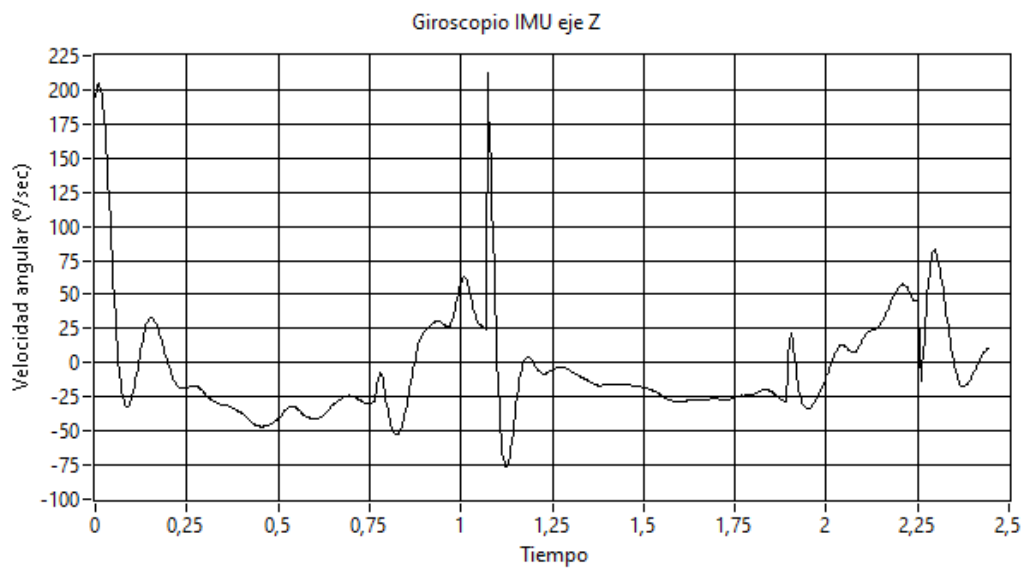
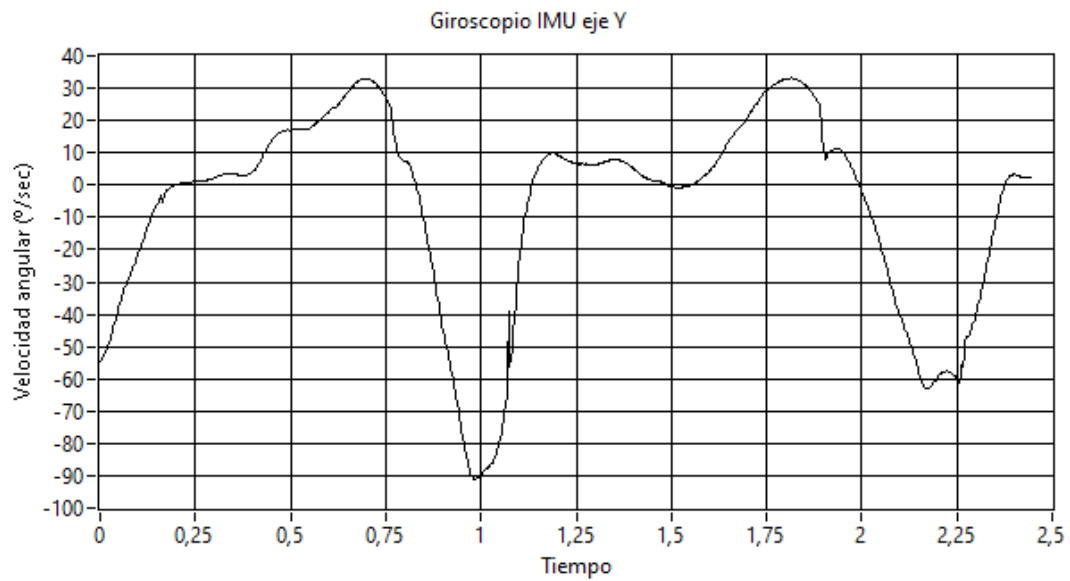
GOLPE 2 REVÉS



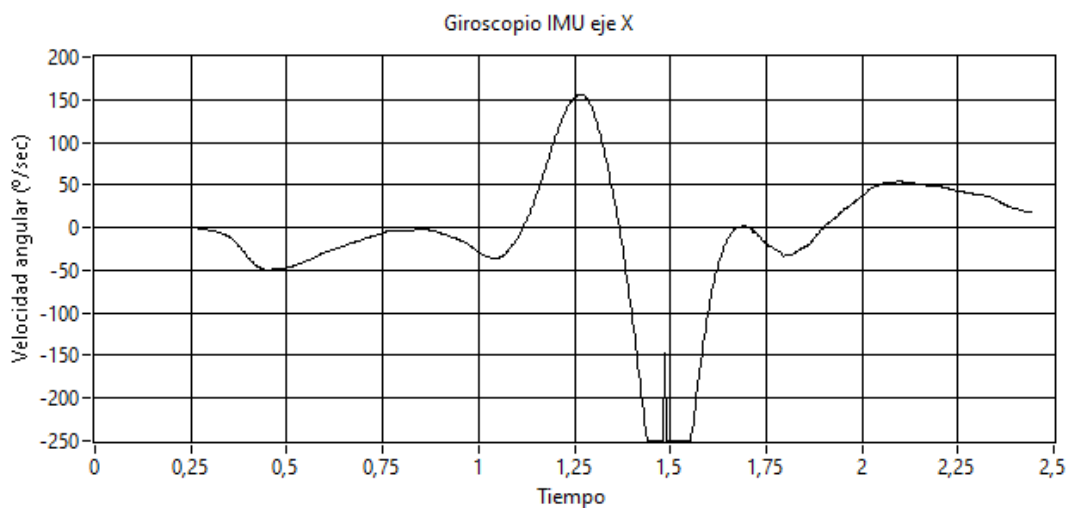


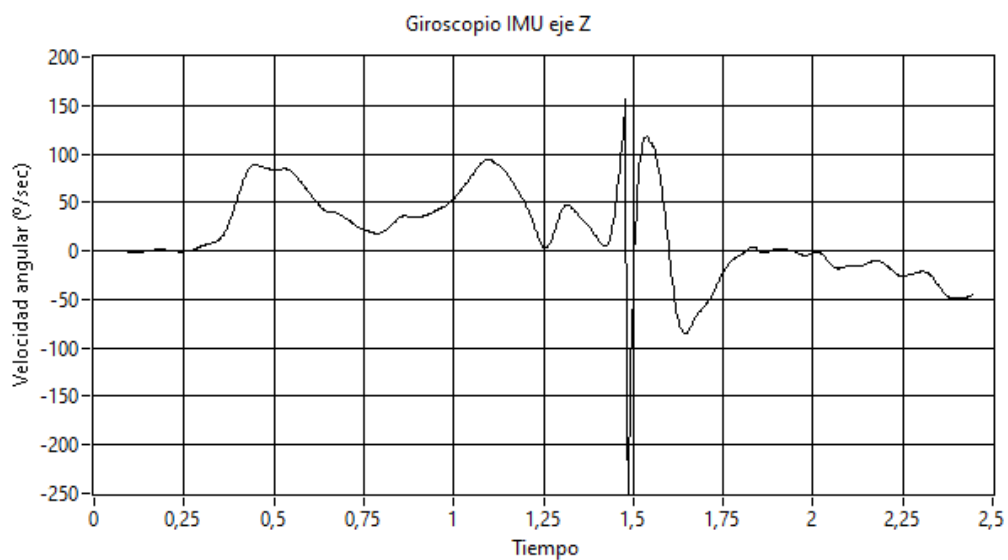
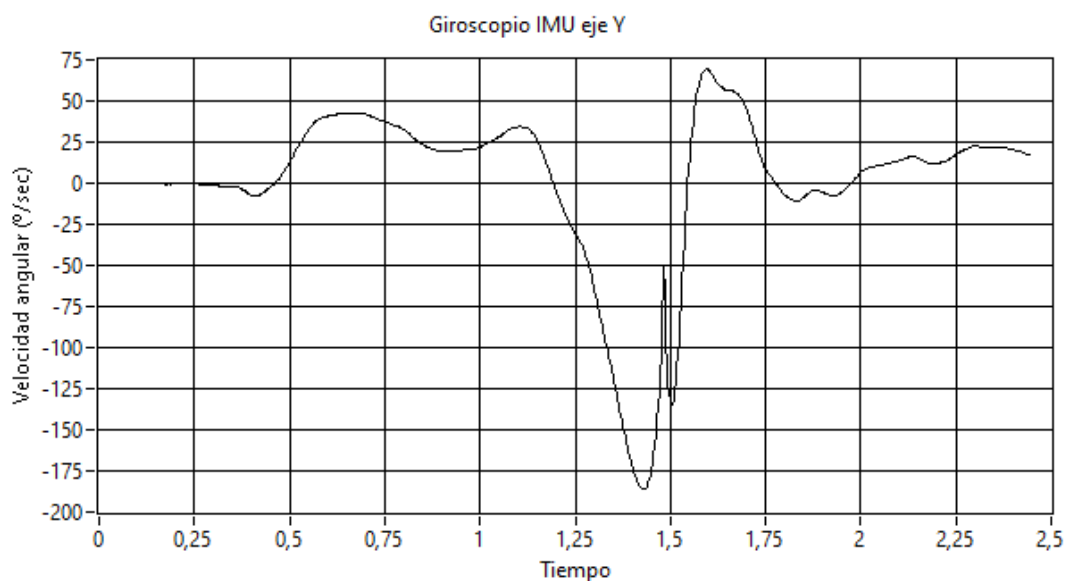
GOLPE 3 REVÉS



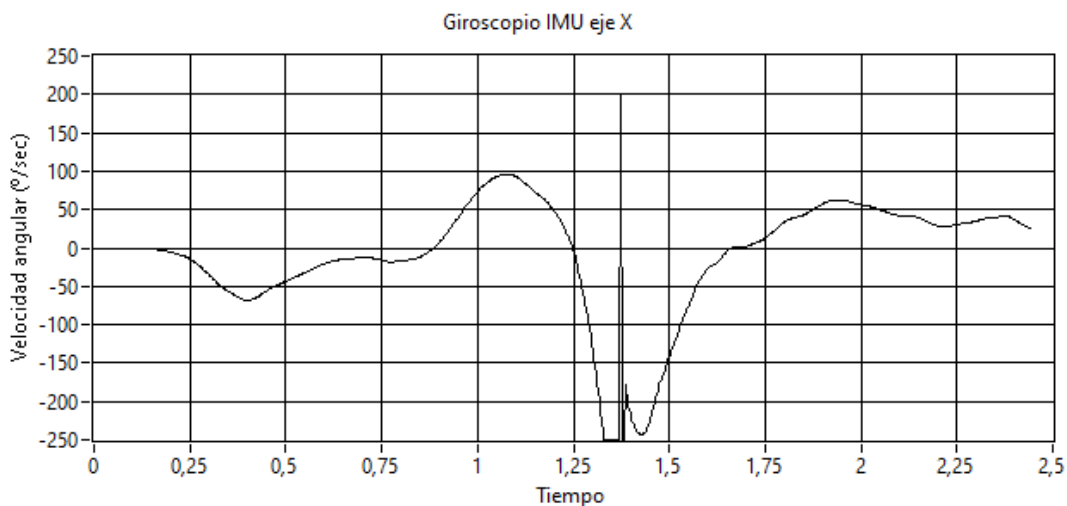


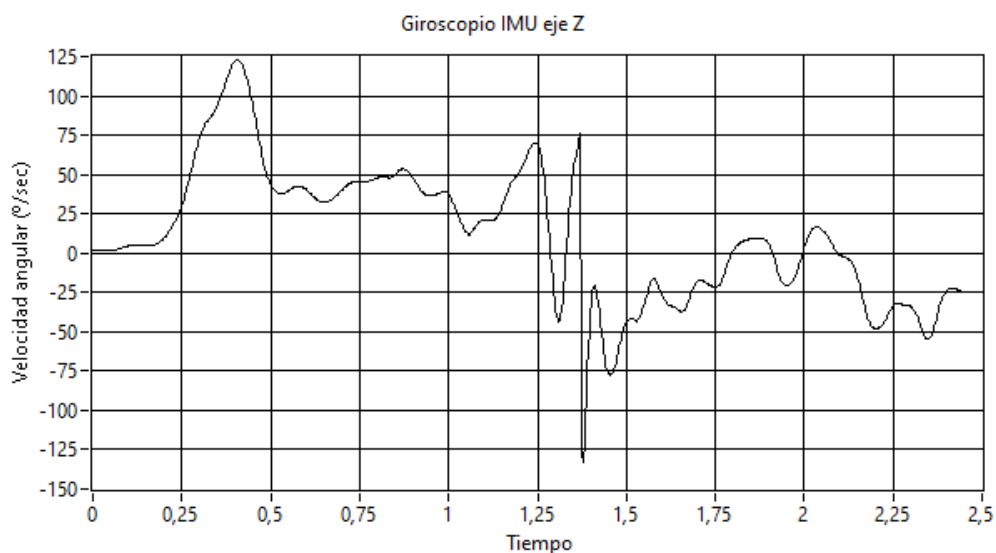
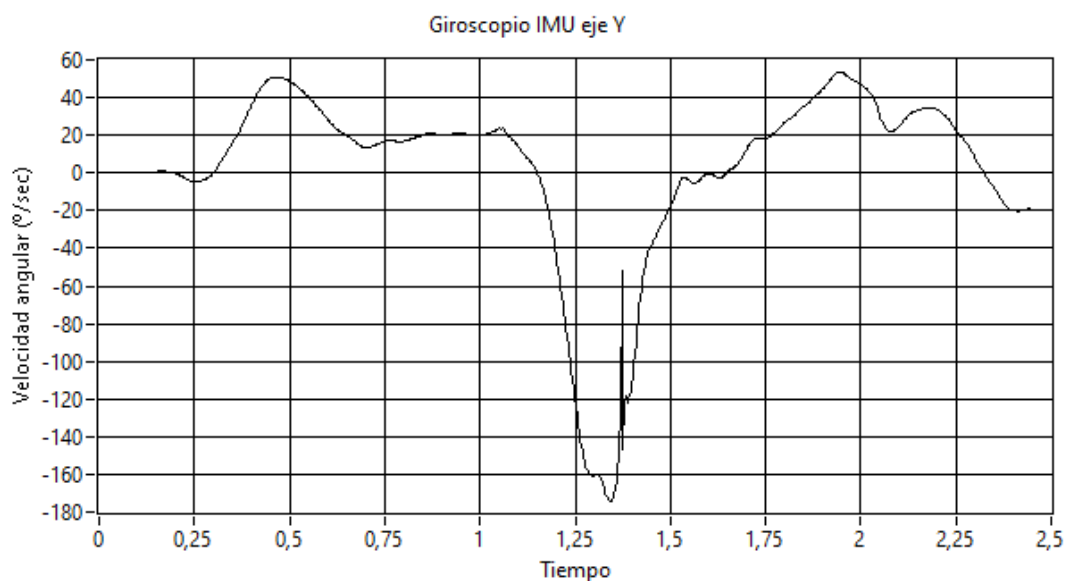
GOLPE 1 REMATE



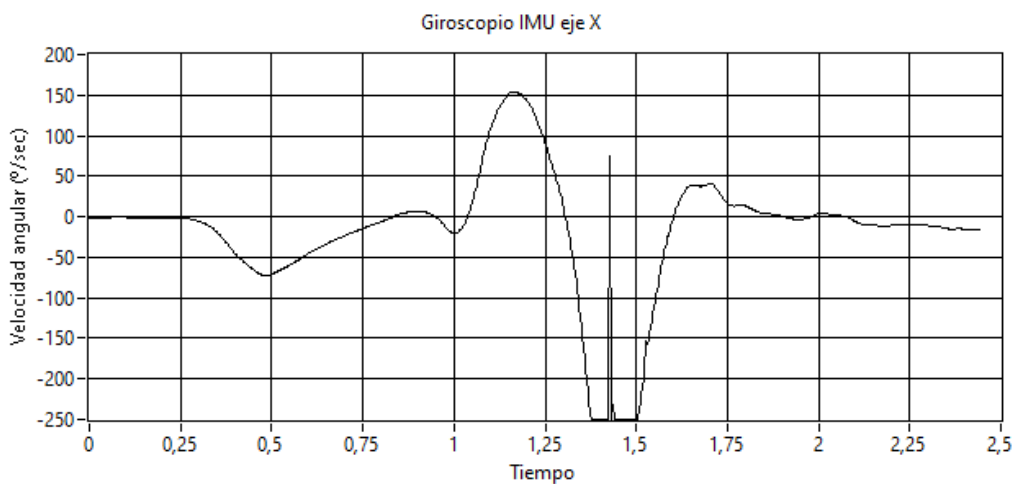


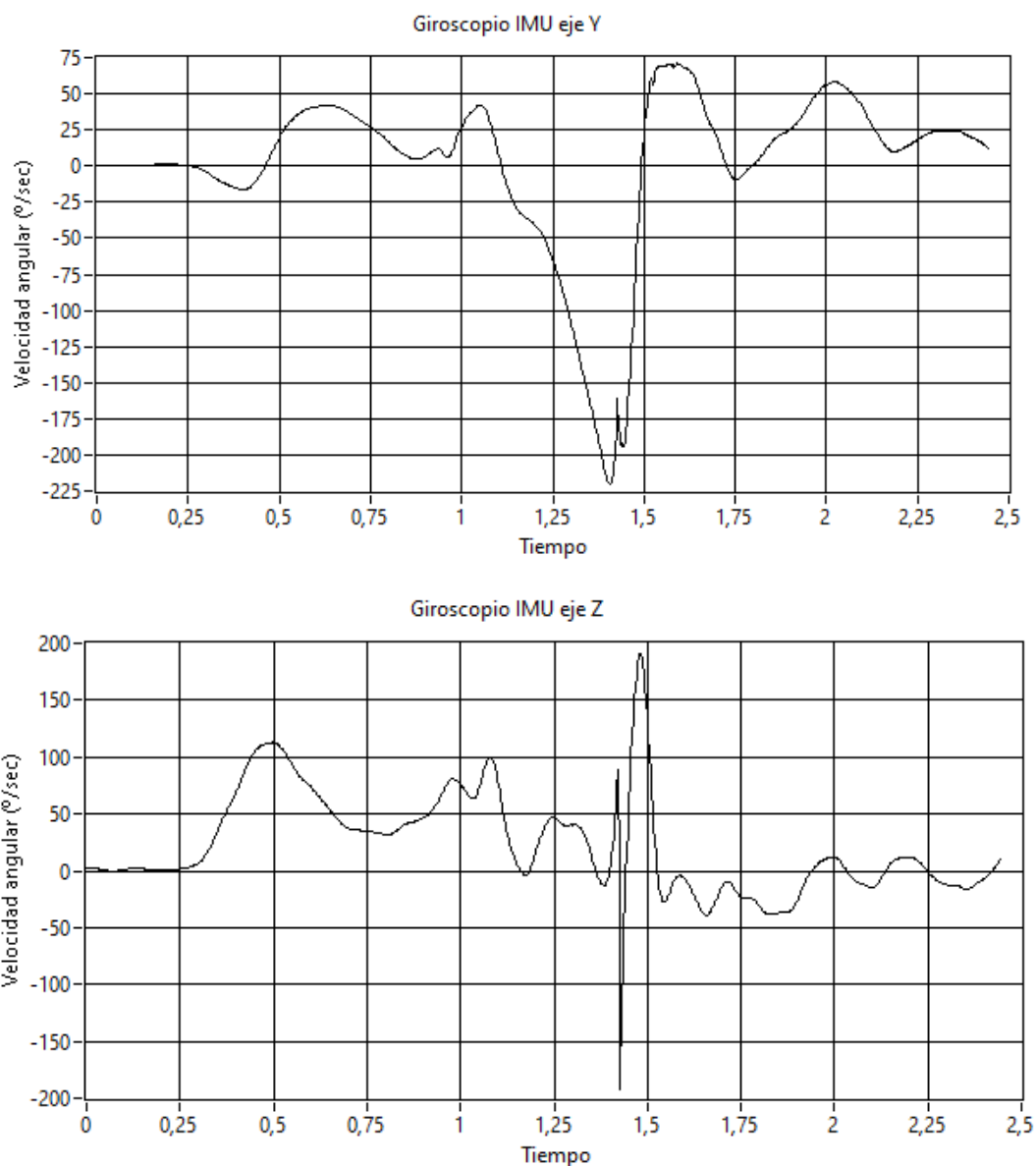
GOLPE 2 REMATE





GOLPE 3 REMATE





9.2.2 Conclusiones tipo de golpe

Una vez obtenidas las gráficas de la velocidad angular en cada tipo de golpe se pasa a realizar un estudio para identificar qué tipo de golpe se ha realizado.

Donde mejor se aprecia el tipo de golpe es en la velocidad angular en el eje X, que en cada tipo de golpe sigue el mismo patrón, el cual se mostrará a continuación.

Golpeo de derecha

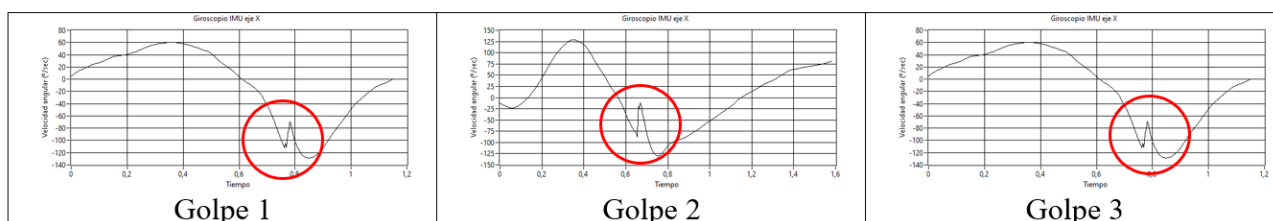


Ilustración 62: Comparativa golpeo de derecha eje X

Cuando se obtiene esta forma de onda en la velocidad angular del eje X se puede afirmar que se ha obtenido un golpeo de derecha, primero la tiene forma ascendente para preparar el golpeo, a continuación, va disminuyendo progresivamente hasta que se produce un pico de aceleración que nos indica que se ha producido el golpeo.

Golpeo de revés

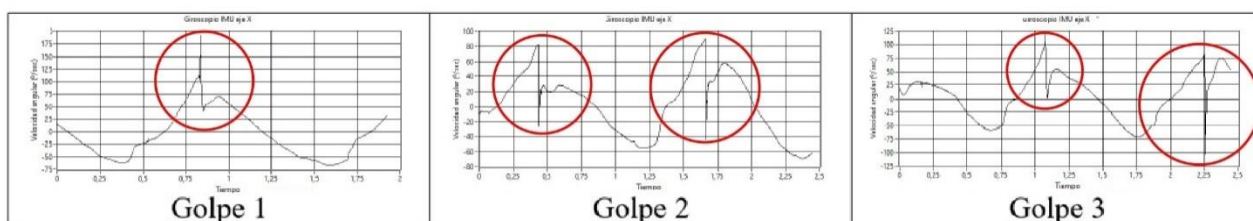


Ilustración 63: Comparativa golpeo de revés eje X

En esta gráfica se puede observar la forma que onda que se consigue cuando se realiza un golpeo de revés, la velocidad angular va aumentando antes de realizar el golpeo, lo que consistiría en la preparación del golpeo y cuando se produce el golpeo cae bruscamente en forma de pico, que es lo que representa al golpeo.

Golpeo de remate

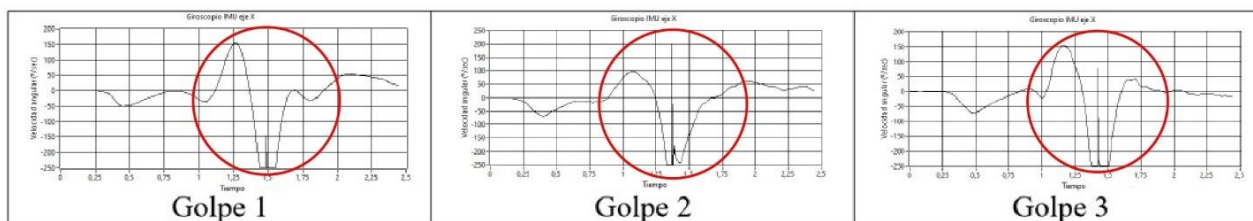


Ilustración 64: Comparativa golpeo de remate eje X

Para terminar, esta es la morfología que tiene la onda al realizar el remate, la velocidad angular va aumentando a medida que se prepara el golpeo, pero en este caso al ir realizando el golpeo va disminuyendo. Pero llega un punto en el que el giroscopio se satura ya que



ha llegado al máximo rango de medida para el que está configurado, pero se puede apreciar como en el golpeo se produce un pico ascendente elevado al igual que en el golpeo de derecha.

Que se sature, es algo normal puesto que el remate es un golpeo ofensivo, y la fuerza que se le aplica al golpeo es mayor que en la derecha y en el revés.

Una vez analizados los tres golpes, se puede observar que el pico que se produce en el golpeo en la derecha y en el remate es un pico ascendente y en el revés es un pico descendente, es, al contrario.

Tiene sentido porque la posición de la pala y la dirección del movimiento son opuestos, por ejemplo, en un jugador diestro el golpeo de derecha y de remate tiene una dirección de derecha a izquierda y el revés de izquierda a derecha. En la derecha se golpea con la cara interior de la mano y en el revés con la cara exterior.

Para poder reconocer automáticamente cada tipo de golpe, se propone crear una red neuronal convolucional de reconocimiento de patrones, ya que cada uno es claramente diferenciado del resto.



10. MEJORAS Y TRABAJOS FUTUROS

En este apartado se va a explicar las mejoras y posibles trabajos futuros que se pueden realizar en este proyecto en base a lo que se ha ido observando durante la realización del mismo.

10.1 Hardware

A nivel de hardware se proponen diferentes mejoras:

- Utilizar un dispositivo de control con más potencia de procesamiento, para muestrear más rápido los sensores, así como con más memoria RAM para poder almacenar más bloques de datos al realizar los envíos y poder adquirir más golpes en cada prueba.
- El sensor con la IMU y el acelerómetro analógico tiene una frecuencia máxima de muestreo de 1 KHz, menor que la frecuencia de muestreo del acelerómetro analógico, por lo que lo limita. Sería interesante buscar un acelerómetro y giróscopo con una frecuencia de muestreo más elevada para así evitar la limitación al acelerómetro de 500 g.
- La última propuesta en cuanto a hardware, sería reducir el tamaño del dispositivo en general, para evitar tener que sujetar el dispositivo de control con una muñequera y los acelerómetros en la base del mango, lo ideal sería elegir sensores y micro controlador con un tamaño más reducido para así integrarlos todos en una placa que se adhiera a la base del mango.

10.2 Comunicaciones

A nivel de comunicaciones se propone realizar una búsqueda para intentar encontrar un protocolo de comunicaciones en el cual se puedan enviar más rápido los bloques de datos y así evitar los dos, tres minutos de envíos de cada ensayo.

10.3 Software

En cuanto al software, las mejoras propuestas son principalmente de cara al análisis de datos.

Un trabajo futuro consistiría en entrenar una red neuronal convolucional para así poder estudiar qué tipo de golpeo es en función del patrón de la gráfica obtenida y que así, cuando se analicen los datos, automáticamente se extraiga el tipo de golpeo.



También se propone crear una aplicación móvil para así poder visualizar los datos sin la necesidad de utilizar el ordenador con el software LabVIEW.



11. CONCLUSIONES

Para finalizar la memoria se va a hablar sobre las conclusiones a las que se ha llegado una vez el proyecto se ha terminado, se pueden dividir en la parte de instrumentación y en los resultados obtenidos.

En cuanto a la parte de instrumentación, el dispositivo es capaz de monitorizar todas las señales que se habían planteado al inicio del proyecto con una frecuencia de muestreo suficiente.

Al contar con dos modos de funcionamiento:

- IMU y el acelerómetro analógico
- Solo con el acelerómetro analógico

Se concluye que cuando se utiliza la IMU se recopila más información, pero solo se ha visto que al utilizar los dos sensores solo se puede utilizar correctamente la IMU ya que al utilizar los dos sensores se muestrea a una frecuencia no suficiente para poder obtener una medida válida del acelerómetro analógico.

El otro modo, solo con el acelerómetro analógico se ha llegado a la conclusión de que no es adecuado puesto que se queda corto de información ya que no hay nada de información del giróscopo.

En cuanto a la parte de resultados, se va a hablar de los dos tipos de ensayos, en exteriores y en la estación.

Ensayos en exteriores: Estos han mostrado que los ensayos son muy diferentes entre sí, es decir, que sacar conclusiones para la caracterización de palas requiere más ensayos y un análisis más estricto debido a la falta de igualdad en los ensayos.

Pero sí que permiten caracterizar los distintos golpes, lo que permite saber si un golpe es de derecha, revés o remate a partir de los patrones de la señal.

Ensayos en estación de salida de bola: En este tipo de ensayos se ha llegado a la conclusión de que se consiguen datos repetibles, por lo tanto, fiables. Lo que permite hacer una comparativa de las diferentes palas.

En pared no se ha llegado a esta repetibilidad debido a que la entrada, el golpeo es muy diferente cada vez y depende mucho de como golpee la persona al contrario que en la estación.



No se ha conseguido llegar a establecer una correlación entre los datos de la estación y los datos de exteriores porque se requieren muchos más datos, muchos más análisis y posiblemente algún sensor más para poder detectar la velocidad de la pala en el golpeo.

También habría que tener en cuenta que el agarre cada vez es diferente, es por eso que cada vez se han obtenido unos datos diferentes en exteriores y con respecto a la estación porque es muy diferente el agarre de la estación con el de una persona al realizar cualquier tipo de golpeo.

12. REFERENCIAS

- [1] «El pádel se erige como el deporte con mayor crecimiento a nivel mundial - CMD Sport». <https://www.cmdsport.com/esencial/cmd-raqueta/padel-se-erige-deporte-mayor-crecimiento-nivel-mundial/> (accedido ago. 08, 2022).
- [2] «Smart Tennis Sensor, más que un cuantificador un corrector que te ayuda a mejorar tu juego». <https://www.xatakamovil.com/espacio-sony/smart-tenis-sensor-o-como-convertirte-en-rafa-nadal-con-ayuda-de-la-tecnologia> (accedido nov. 08, 2021).
- [3] «Tennis Zepp | Analice y mejore su servicio y golpe». <http://www.zepplabs.com/en-us/tennis/freeapp/> (accedido nov. 11, 2021).
- [4] «Tennis Sensor – HEAD». https://www.head.com/es_ES/sensor (accedido nov. 11, 2021).
- [5] «Babolat Play». <https://es.babolatplay.com/> (accedido nov. 19, 2021).
- [6] «Babolat Play Pure Drive review. Smart tennis racket - CNET - CNET». <https://www.cnet.com/reviews/babolat-play-pure-drive-review/> (accedido nov. 19, 2021).
- [7] «Sensor de tenis Artengo - Tecnología de tenis». <http://tennis-technology.com/artengo-tennis-sensor/> (accedido nov. 11, 2021).
- [8] «Babolat Pop, una pulsera que registra todos tus movimientos como tenista». <https://www.vitonica.com/equipamiento/babolat-pop-una-pulsera-que-registra-todos-tus-movimientos-como-tenista> (accedido nov. 19, 2021).
- [9] «Arduino y solo Arduino - Todo lo que necesitas lo encontrarás aquí.: ¿Qué es un ESP32?» <https://soloarduino.blogspot.com/2017/03/que-es-un-esp32.html> (accedido may 31, 2021).
- [10] J. Manuel, S. Navarro, y J. S. Muro, «UNIVERSIDAD POLITÉCNICA DE CARTAGENA».
- [11] «Matlab - csic.es». <http://sitios.csic.es/web/calculo-cientifico/matlab> (accedido ago. 11, 2022).
- [12] «Labview». <https://sdei.unican.es/Paginas/servicios/software/Labview.aspx> (accedido ago. 11, 2022).
- [13] F. Blanes y J. L. Posadas, «Unidad Temática 4: Protocolos Nivel de Transporte».



- [14] «ESP-Now conecta dos o más ESP32/ESP8266 - ElectroSoftCloud». <https://www.electrosoftcloud.com/esp-now-conecta-dos-o-mas-esp32-esp8266/> (accedido ago. 11, 2022).
- [15] «Protocolo ESPNOW para la comunicación entre dos dispositivos NodeMcu Esp8266.» <https://electrodiy.cl/2022/04/04/protocolo-espnw-para-la-comunicacion-entre-dos-dispositivos-nodemcu-esp8266/> (accedido ago. 11, 2022).
- [16] C. G. de la Peña, «Teoría del proyecto técnico», pp. 1-48.
- [17] M. de Trabajo, «Disposición 14977 del BOE núm. 251 de 2019», 2019. Accedido: may 26, 2021. [En línea]. Available: <http://www.boe.es>
- [18] «Seguridad Social: Cotización / Recaudación de Trabajadores». <https://www.seg-social.es/wps/portal/wss/internet/Trabajadores/CotizacionRecaudacionTrabajadores/10721/10957/583> (accedido may 26, 2021).
- [19] «Tarifa de primas por contingencias profesionales (Accidente de Trabajo y Enfermedad Profesional) 2019 | Iberley». <https://www.iberley.es/temas/tarifa-primas-contingencias-profesionales-accidente-trabajo-enfermedad-profesional-2019-6571> (accedido may 26, 2021).

ANEXO N° 1 PRESUPUESTOS

El objetivo del presente documento tiene como finalidad proporcionar información acerca del coste de la realización del proyecto.

Este documento está constituido por tres partes claramente diferenciadas:

En primer lugar, la determinación de las mediciones de cada una de las unidades de obra del proyecto.

En segundo lugar, la relación de los precios de dichas unidades de obra y partidas alzadas.

Por último, los presupuestos finales, que son resultado de la suma de todas las mediciones por los precios y de la aplicación de ciertos coeficientes.

Antes de las tres partes principales del documento, se realiza una justificación de los precios.

1.1 Justificación de los precios

Para comenzar se realizará una justificación de los precios del proyecto, para ello se calculan los precios de cada unidad de obra y partidas alzadas, precios que se utilizarán para elaborar los cuadros de precios.

1.1.1 Costes

En el presupuesto del proyecto se tienen en cuenta dos tipos de costes, que se evalúan en la Justificación de precios, los costes directos e indirectos.

1.1.1.1 Costes Directos

Estos costes son aquellos correspondientes a los distintos elementos que intervienen directamente en la ejecución de cada una de las unidades de obra.

Están constituidos por el coste de la mano de obra, materiales, gastos de personal combustible, energía y conservación de la maquinaria.

Coste horario de la mano de obra

Este coste se rige por la siguiente ecuación:

$$C = A + B + K * A$$

Siendo

A: Cantidades percibidas por el trabajo con carácter salarial, sujetas a cotización.

B: Cantidades percibidas por el trabajador con carácter no salarial, no sujetas a cotización.

K: Cantidades devengadas por el empresario como cotización a la seguridad social y seguro de accidentes, es el resultado de aplicar los porcentajes reglamentarios a las cantidades sujetas a dicha cotización.

C, A y B se expresan en euros/hora y **K** en tanto por uno.[16]

Para calcular las cantidades percibidas con carácter salarial (A), las cantidades sujetas a cotización se recurre a la resolución del 7 de octubre de 2019, de la Dirección General del Trabajo, del XIX convenio colectivo nacional de empresas de ingeniería y oficinas de estudios técnicos.

	Niveles	Tabla salarial según art. 33		Plus convenio anual según art. 38 convenio	Total anual
		Mes x 14	Anual		
1	LICENCIADOS Y TITULADOS 2.º Y 3.º CICLO UNIVERSITARIO Y ANALISTA	1.712,42	23.973,88	2.349,69	26.323,57
2	DIPLOMADOS Y TITULADOS 1.º CICLO UNIVERSITARIO. JEFE SUPERIOR	1.291,04	18.074,56	2.349,69	20.424,25
3	TÉCNICO DE CÁLCULO O DISEÑO, JEFE DE 1.º Y PROGRAMADOR DE ORDENADOR	1.244,93	17.429,02	2.349,69	19.778,71
4	DELINEANTE-PROYECTISTA, JEFE DE 2.º Y PROGRAMADOR DE MAQ. AUXILIARES	1.141,36	15.979,04	2.349,69	18.328,73
5	DELINEANTE, TÉCNICO DE 1.º, OFICIAL 1.º ADMTVO. Y OPERADOR DE ORDENADOR	1.019,82	14.277,48	2.349,69	16.627,17
6	DIBUJANTE, TÉCNICO DE 2.º, OFICIAL 2.º ADMTVO., PERFORISTA, GRABADOR Y CONSERJE	878,63	12.300,82	2.349,69	14.650,51
7	TELEFONISTA-RECEPCIONISTA, OFICIAL 1.º OFICIOS VARIOS Y VIGILANTE	849,16	11.888,24	2.349,69	14.237,93
8	AUXILIAR TÉCNICO, AUX. ADMTVO., TELEFONISTA, ORDENANZA, PERSONAL DE LIMPIEZA Y OFICIAL 2.º OFICIOS VARIOS	790,36	11.065,04	2.349,69	13.414,73
9	AYUDANTE OFICIOS VARIOS	757,29	10.602,06	2.349,69	12.951,75

Tabla

1: Tablas salariales según convenio año 2020 [17]

El nivel salarial que corresponde al proyecto es el Nivel 2: *Diplomados y Titulados 1º ciclo universitario, jefe superior*, cuyo sueldo anual es de 20.424,25 €/ año.

Se lleva a cabo el cálculo del coste horario por mano de obra:

- Teniendo en cuenta que un ingeniero por convenio trabaja un máximo de 1792 horas al año, trabajando 8 horas al día:

Primero se calculan los días laborables al año.

$$\text{Días laborables} = \frac{1792 \text{ horas/año}}{8 \text{ horas/día}} = 224 \frac{\text{días}}{\text{año}}$$

$$A (\text{precio por hora}) = \frac{\text{Salario base al año}}{\text{Horas trabajadas al año}} = \frac{20424,25 \frac{\text{€}}{\text{año}}}{1792 \frac{\text{horas}}{\text{año}}} = 11,39 \frac{\text{€}}{\text{h}}$$

Las cuotas a ingresar a la Seguridad social se calculan aplicando a la base de cotización del trabajador el porcentaje que corresponde a cada contingencia protegida.[18]

Tipo de cotización:

Concepto	Empresa	Trabajador	Total
Contingencias comunes	23,6	4,7	28,3
Desempleo	5,5	1,55	7,05
Fondo de garantía social	0,2	No cotiza	0,2
Formación profesional	0,6	0,1	0,7
Horas extraordinarias fuerza mayor	12	2	14
Horas extraordinarias comunes	23,6	4,7	28,3

Tabla 2: Cuotas a ingresar a la seguridad social según tipo. [18]

El porcentaje de cotización por accidentes de trabajo y enfermedades laborales depende de la actividad que se realice, en este caso corresponde con la fabricación de equipos electrónicos e informáticos que se encuadra dentro de la categoría 26 de los códigos CNAE-2009.[19]

Código CNAE-2009	Denominación	IT (incapacidad temporal)	IMS (Invalidez, muerte y supervivencia)	Total
26	Fabricación de productos informáticos, electrónicos y ópticos	1.5	1.1	2.6

Tabla 3: Porcentaje de cotización por accidentes CNAE-2009 26 [4]

El porcentaje de horas extra no se tiene en cuenta ya que es un dato que antes de hacer el proyecto es muy complicado de estimar.

Desde la posición de proyectista los porcentajes que se tienen en cuenta son aquellos pertenecientes a la empresa.

Por ello será:



$$K = \frac{23.6 + 5.5 + 0.2 + 0.6 + 2.6}{100} = 0.325$$

Finalmente, los costes de mano de obra son:

$$C = 11.39 \frac{\text{€}}{h} + 0 + 0.325 * 11.39 \frac{\text{€}}{h} = 15,09 \frac{\text{€}}{h}$$

Costes horarios de la maquinaria

En este apartado se procede a determinar el coste derivado de la utilización de cada una de las máquinas que intervienen en las Unidades de Obra.

En este apartado se procede a determinar el coste derivado de la utilización de cada una de las máquinas que intervienen en las Unidades de Obra.

Concepto	Precio	Precio por hora
Uso del ordenador Msi GP62 6QF leopard pro: procesador Intel core i7 de sexta generación, 16 GB de memoria RAM, 512 de SSD y tarjeta gráfica NVIDIA GTX960M	989 €	0,122 €/h
Uso del software LabVIEW	6600 €	0,982 €/h
Uso del software Arduino IDE Software libre	0,00 €	0,00 €/h

Tabla 4: Costes horarios de la maquinaria

Coste horario del ordenador

El ordenador que se ha utilizado para realizar el proyecto es un MSI, modelo GP62 6QF Leopard pro, con un procesador Intel core i7 de sexta generación, 16 GB de memoria RAM, 512 de SSD y tarjeta gráfica NVIDIA GTX960M. Su precio fue 989 €.

Se considera una vida útil de 6 años, se estima un uso medio de 6 horas por día laborable.

$$Coste_{ordenador} = \frac{989 \text{ €}}{6 \text{ años}} * \frac{1 \text{ año}}{224 \text{ días laborables}} * \frac{1 \text{ día laborable}}{6 \text{ horas}} = 0,982 \frac{\text{€}}{h}$$



Coste del software LabVIEW

El software empleado para el crear la aplicación de procesado y muestra de los planos LabVIEW, cuya licencia tiene un valor de 1.875 €, con una duración de 1 años. Se establece que se utiliza un de 6 horas al día.

$$Coste_{solidWorks} = \frac{1875 \text{ €}}{1 \text{ año}} * \frac{1 \text{ año}}{224 \text{ días laborables}} * \frac{1 \text{ día laborable}}{6 \text{ horas}} = 1,395 \frac{\text{€}}{h}$$

1.1.1.2 Costes Indirectos

Estos son aquellos costes que se producen por el mero hecho de la ejecución de la obra, es decir, que no se pueden imputar a ninguna unidad de obra en particular, se aplican al conjunto.

Son denominados gastos generales, y se añadirán al final del presupuesto.

1.2 MEDICIONES

A continuación, se muestra una tabla correspondiente a cada unidad de obra que conforma el proyecto.

Designación de la unidad	Nombre	Volumen Parcial	Volumen Total
Nº 1	Diseño y montaje del dispositivo	1	1
Nº 2	Programación del dispositivo	1	1
Nº3	Programación de la aplicación	1	1
Nº4	Ensayos y recogida de datos	1	1
Nº5	Redacción del proyecto y estudio de los datos	1	1
Nº6	Control de calidad, incluye prueba de funcionamiento	1	1

Tabla 5: Mediciones unidades de obra



1.3 CUADROS DE PRECIOS

1.3.1 Cuadro de precios N°1

Se muestra una tabla con los precios finales de todas las unidades de obra que conforman el proyecto.

Nº de Orden de la Unidad	Designación de la Unidad	Precio en letra	Precio en cifra
1	Diseño y montaje del dispositivo	Setecientos sesenta y cinco euros con cincuenta y cuatro céntimos	765,54 €
2	Programación del dispositivo	Mil novecientos un euro con cincuenta céntimos	1901,50 €
3	Programación de la aplicación	Quinientos ochenta y un euros con doscientos cuarenta y cinco céntimos	581,245 €
4	Ensayos y recogida de datos	Quinientos treinta y un euros con cuatrocientos veinticuatro céntimos	531,424 €
5	Redacción del proyecto y estudio de los datos	Mil novecientos nueve euros con ochenta y cinco céntimos	1909,85 €
6	Control de calidad, incluye prueba de funcionamiento	Ciento siete euros con novecientos cuarenta y cinco céntimos	107,945 €

Tabla 6: Cuadro de precios N° 1

1.3.2 Cuadro de precios N°2

Se mostrará a continuación los precios detallados de cada unidad de obra incluyendo sus costes indirectos.

1.3.2.1 Unidad de obra N°1 Diseño y montaje del dispositivo

Materiales:

Componente	Precio unitario (PVP)	uds	Precio Total
ESP32-S2-DevKit-Lipo	9,95 €	2	19,90 €
MikroElektronika MPU IMU Click - MIKROE-1577	50,12 €	1	50,12 €
ADXL1004BCPZ-RL7	66,03 €	1	66,03 €
Pila recargable de Polímero de Litio, 3,7 V 1,8 Ah	15,42 €	1	15,42 €
Placa perforada pruebas	2,92 €	1	2,92 €
Placa soldadura	2,67 €	1	2,67 €
Coste total de materiales			157,06 €

Tabla 7: Coste de materiales Unidad de obra N°1

Mano de obra:

Personal	Precio/hora	Horas	Precio Total
Diplomados y titulados 1er ciclo universitario	15,09 €/h	40	603,60 €
Coste total de la mano de obra			603,60 €

Tabla 8: Coste de mano de obra Unidad de obra N°1

Maquinaria:

Denominación	Precio por hora	Horas	Precio Total
Uso del ordenador	0,122 €/h	40	4,88 €
Uso del software Arduino IDE	0 €/h	40	0 €
Coste total de la maquinaria			4,88 €

Tabla 9: Coste de maquinaria Unidad de obra N° 1



Coste total materiales.....	157,06 €
Coste total mano de obra.....	603,60 €
Coste total maquinaria.....	4,88 €
Total costes directos.....	765,54 €
PRECIO UNITARIO.....	765,54 €

1.3.2.2 Unidad de obra N°2 Programación del dispositivo

Mano de obra:

Personal	Precio/hora	Horas	Precio Total
Diplomados y titulados 1er ciclo universitario	15,09 €/h	125	1886,25 €
Coste total de la mano de obra			1886,25 €

Tabla 6: Coste de mano de obra Unidad de obra N°2

Maquinaria:

Denominación	Precio por hora	Horas	Precio Total
Uso del ordenador	0,122 €/h	125	15,25 €
Uso del software Arduino	0 €/h	125	0 €
Coste total de la maquinaria			15,25 €

Tabla 7: Coste de maquinaria Unidad de obra N°3

Coste total materiales.....	0,00 €
Coste total mano de obra.....	1886,25 €
Coste total maquinaria.....	15,25 €
Total costes directos.....	1901,50 €
PRECIO UNITARIO.....	1901,50 €



1.3.2.3 Unidad de obra N°3 Programación de la aplicación

Mano de obra:

Personal	Precio/hora	Horas	Precio Total
Diplomados y titulados 1er ciclo universitario	15,09 €/h	35	528,15 €
Coste total de la mano de obra			528,15 €

Tabla 10: Coste de mano de obra Unidad de obra N°3

Maquinaria:

Denominación	Precio por hora	Horas	Precio Total
Uso del ordenador	0,122 €/h	35	4,27 €
Uso del software LabVIEW	1,395 €/h	35	48,825 €
Coste total de la maquinaria			53,095 €

Tabla 11: Coste de maquinaria Unidad de obra N°3

Coste total materiales.....	0,00 €
Coste total mano de obra.....	528,15 €
Coste total maquinaria.....	53,095 €
Total costes directos.....	581,245 €
PRECIO UNITARIO.....	581,245 €

1.3.2.4 Unidad de obra N°4 Ensayos y recogida de datos

Mano de obra:

Personal	Precio/hora	Horas	Precio Total
Diplomados y titulados 1er ciclo universitario	15,09 €/h	32	482,88 €
Coste total de la mano de obra			482,88 €

Tabla 12: Coste de mano de obra Unidad de obra N°4



Maquinaria:

Denominación	Precio por hora	Horas	Precio Total
Uso del ordenador	0,122 €/h	32	3,904 €
Uso del software LabVIEW	1,395 €/h	32	44,64 €
Coste total de la maquinaria			48,544 €

Tabla 13: Coste de maquinaria Unidad de obra N°4

Coste total materiales.....	0,00 €
Coste total mano de obra.....	482,88 €
Coste total maquinaria.....	48,544 €
Total costes directos.....	531,424 €
PRECIO UNITARIO.....	531,424 €

1.3.2.5 Unidad de obra N°5 Redacción del proyecto y estudio de los datos

Mano de obra:

Personal	Precio/hora	Horas	Precio Total
Diplomados y titulados 1er ciclo universitario	15,09 €/h	115	1735,35 €
Coste total de la mano de obra			1735,35 €

Tabla 14: Coste de mano de obra Unidad de obra N°4

Maquinaria:

Denominación	Precio por hora	Horas	Precio Total
Uso del ordenador	0,122 €/h	115	14,03 €
Uso del software LabVIEW	1,395 €/h	115	160,425 €
Coste total de la maquinaria			174,455 €

Tabla 15: Coste de maquinaria Unidad de obra N°4

Coste total materiales.....	0,00 €
Coste total mano de obra.....	1735,35 €
Coste total maquinaria.....	174,455 €
Total costes directos.....	1909,805 €
PRECIO UNITARIO.....	1909,805 €



1.3.2.6 Unidad de obra Nº6 Control de calidad, con prueba de funcionamiento

Mano de obra:

Personal	Precio/hora	Horas	Precio Total
Diplomados y titulados 1er ciclo universitario	15,09 €/h	6,5	98,085 €
Coste total de la mano de obra			98,085 €

Tabla 16: Coste de mano de obra Unidad de obra Nº2

Maquinaria:

Denominación	Precio por hora	Horas	Precio Total
Uso del ordenador	0,122 €/h	6,5	0,793 €
Uso del software LabVIEW	1,395 €/h	6,5	9,0675 €
Coste total de la maquinaria			9,8605 €

Tabla 17: Coste de maquinaria Unidad de obra Nº2

Coste total materiales.....	0,00 €
Coste total mano de obra.....	98,085 €
Coste total maquinaria.....	9,8605 €
Total costes directos.....	107,945 €
PRECIO UNITARIO.....	107,945 €

1.4 PRESUPUESTOS

Este apartado tiene la finalidad de dar una idea de lo que costará el proyecto, existen tres presupuestos:

- Presupuesto de Ejecución Material
- Presupuesto de Ejecución por Contrata
- Presupuesto Total



1.4.1 Presupuesto de Ejecución Material

Este presupuesto es la suma de los productos de los volúmenes a ejecutar de las distintas unidades de obra por sus correspondientes precios unitarios.

Nº de Orden de la Unidad	Designación de la Unidad	Medición unidades	Precio unitario (€)	Presupuesto (€)
1	Diseño y montaje del dispositivo	1	765,54 €	765,54 €
2	Programación del dispositivo	1	1901,50 €	1901,50 €
3	Programación de la aplicación	1	581,245 €	581,245 €
4	Ensayos y recogida de datos	1	531,424 €	531,424 €
5	Redacción del proyecto y estudio de los datos	1	1992,84 €	1909,805 €
6	Control de calidad, incluye prueba de funcionamiento	1	107,945 €	107,945 €
PRECIO TOTAL				5797,46 €

Tabla 18: Presupuesto de ejecución Material

El presupuesto de Ejecución Material asciende a CINCO MIL SETECIENTOS NOVENTA Y SIETE CON CUARENTA Y SEIS CÉNTIOS, 5797,46 €.

1.4.2 Presupuesto de Ejecución por Contrata

Presupuesto de Ejecución Material.....	5797,46 €
Gastos generales (10%).....	579,746 €
Beneficio industrial (6%).....	347,846 €
TOTAL PARCIAL.....	6725,0536 €
IVA(21%).....	1412,26 €
TOTAL PRESUPUESTO DE EJECUCIÓN POR CONTRATA.....	8137,315 €

El presupuesto de Ejecución por Contrata asciende a la cantidad de OCHO MIL CIENTO TREINTA Y SIETE CON TRESCIENTOS QUINCE CÉNTIOS, 8137,315 €.

ANEXO Nº2 CÓDIGO DE LOS MICRO CONTROLADORES

2.1 Código del dispositivo de adquisición, Acelerómetro analógico e IMU

```
/*
Santiago Bujedo Saiz
Adquisición de datos IMU y acelerómetro analógico
*/
#include "I2Cdev.h"
#include "MPU6050.h"
#include <esp_now.h>
#include <WiFi.h>
#include <Adafruit_NeoPixel.h>

#define NUMPIXELS      1
#define PIN_NEOPIXEL  18
Adafruit_NeoPixel pixels(NUMPIXELS, PIN_NEOPIXEL, NEO_GRB +
NEO_KHZ800);

MPU6050 accelgyro(0x69); // <-- use for ADO high
uint8_t slaveAddress[] = {0x7C, 0xDF, 0xA1, 0x00, 0x75, 0x56};
int16_t ax, ay, az;
int16_t gx, gy, gz;
float ax_m_s2, ay_m_s2, az_m_s2 ,gx_deg_s, gy_deg_s,
gz_deg_s, gScaled;
int16_t temp_raw;
float temp;
unsigned long previo, actual, interrupcion, tiempo;
int cont=0;
const int PINac500 = 12;
float ref_V=1.25; // dependerá del divisor de tensión aqui poner el
valor que te de el acelerometro alimentado a 3.3 a 0 g
float sensibilidad=0.002; // son 2.7 mv/g pasados por el divisor de
tension
int resolution=12;
int var=0;
String dataMessage;
int accx;
int periodo_de_muestreo;
const int num_vectores=850;
int16_t acc_x [num_vectores];
int16_t acc_y[num_vectores];
int16_t acc_z[num_vectores];
int16_t gy_x[num_vectores];
int16_t gy_y[num_vectores];
int16_t gy_z[num_vectores];
int16_t ac_500[num_vectores];

typedef struct valores {
float acx;
float acy;
float acz;
float gyrox;
float gyroy;
float gyroz;
float temp;
float ac500g;
bool var_temperatura;
};
```



```
valores MPUdata;

typedef struct temperatura {
    float temp;
};
temperatura TEMPP;

void lecturaVALORES ()
{
    accelgyro.getAcceleration(&ax, &ay, &az);
    accelgyro.getRotation(&gx, &gy, &gz);
    accx = analogRead(PINac500);
}

void lecturaTEMP ()
{
    temp_raw = accelgyro.getTemperature();
    temp = float(temp_raw )/340 + 36.53;
}

void IRAM_ATTR empezar_enviar() {
    var=1;
}

void setup() {

    #if defined(NEOPIXEL_POWER)
        pinMode(NEOPIXEL_POWER, OUTPUT);
        digitalWrite(NEOPIXEL_POWER, HIGH);
    #endif

    pixels.begin();
    pixels.setBrightness(100);
    pixels.setColor(0, 0, 0);
    pixels.show();

    #if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
        Wire.begin();
    #elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTWIRE
        Fastwire::setup(400, true);
    #endif
    Serial.begin(115200);
    analogReadResolution(resolution);

    accelgyro.initialize();

    accelgyro.setXAccelOffset(-271);
    accelgyro.setYAccelOffset(319);
    accelgyro.setZAccelOffset(1885);
    accelgyro.setXGyroOffset(125);
    accelgyro.setYGyroOffset(64);
    accelgyro.setZGyroOffset(14);
```




```
Serial.println("Initializing I2C devices...");
accelgyro.initialize();

// verify connection
Serial.println("Testing device connections...");
Serial.println(accelgyro.testConnection() ? "MPU6050 connection
successful" : "MPU6050 connection failed");

Serial.println("Updating internal sensor offsets...");
Serial.print(accelgyro.getXAccelOffset());
//Serial.print("\t"); // -76
Serial.print(accelgyro.getYAccelOffset());
//Serial.print("\t"); // -2359
Serial.print(accelgyro.getZAccelOffset());
//Serial.print("\t"); // 1688
Serial.print(accelgyro.getXGyroOffset()); //Serial.print("\t");
// 0
Serial.print(accelgyro.getYGyroOffset()); //Serial.print("\t");
// 0
Serial.print(accelgyro.getZGyroOffset()); //Serial.print("\t");
// 0
Serial.print("\n");
accelgyro.setXAccelOffset(-271);
accelgyro.setYAccelOffset(319);
accelgyro.setZAccelOffset(1885);
accelgyro.setXGyroOffset(125);
accelgyro.setYGyroOffset(64);
accelgyro.setZGyroOffset(14);
Serial.print(accelgyro.getXAccelOffset());
//Serial.print("\t"); // -76
Serial.print(accelgyro.getYAccelOffset());
//Serial.print("\t"); // -2359
Serial.print(accelgyro.getZAccelOffset());
//Serial.print("\t"); // 1688
Serial.print(accelgyro.getXGyroOffset()); //Serial.print("\t");
// 0
Serial.print(accelgyro.getYGyroOffset()); //Serial.print("\t");
// 0
Serial.print(accelgyro.getZGyroOffset()); //Serial.print("\t");
// 0
Serial.print("\n");

pinMode(0, INPUT_PULLUP);
pinMode(34, OUTPUT);
attachInterrupt(digitalPinToInterrupt(0), empezar_enviar,
FALLING);

WiFi.mode(WIFI_STA);

if (esp_now_init() != ESP_OK) {
Serial.println("There was an error initializing ESP-NOW");
return;
}

esp_now_peer_info_t slaveInfo;
memcpy(slaveInfo.peer_addr, slaveAddress, 6);
slaveInfo.channel = 0;
```



```
slaveInfo.encrypt = false;

if (esp_now_add_peer(&slaveInfo) != ESP_OK) {
    Serial.println("There was an error registering the slave");
    return;
}

}

void OnSent(const uint8_t *mac_addr, esp_now_send_status_t status)
{
    Serial.print("\r\nSend message status:\t");
    Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Sent
Successfully" : "Sent Failed");
}

void loop(){
    String dataString = "";

    if (var==1 ){
        for (int i=0; i<5; i++)
        {

            Serial.println("espero un poquinin ");
            pixels.setPixelColor(0, pixels.Color(0, 0, 255));
            pixels.show();
            delay(1000);
            pixels.fill(0x000000);
            pixels.show();
            delay(1000);

        }
        Serial.println("EMPIEZOOOOOOOOOOOOOOOOOOOOOOOO");
        pixels.setPixelColor(0, pixels.Color(0, 255, 0));
        pixels.show();
        previo = micros();

        for (int i=0;i<num_vectores;i++)
        {

            previo = micros();
            lecturaVALORES(); // tarda 2.5 ms
            acc_x[i]=ax;
            acc_y[i]=ay;
            acc_z[i]=az;
            gy_x[i]=gx;
            gy_y[i]=gy;
            gy_z[i]=gz;
            ac_500[i]=accx;

        }
        Serial.print("La función tardo:");
        Serial.println( actual - previo );
        digitalWrite(34,LOW);
        lecturaTEMP();
        MPUdata.temp=temp;
        MPUdata.var_temperatura=1;
    }
}
```



```
    esp_err_t result = esp_now_send(slaveAddress, (uint8_t *)
&MPUdata, sizeof(MPUdata));
    Serial.println(" ENVIO LA TEMPERATURA");
    pixels.setPixelColor(0, pixels.Color(255, 0, 0));
    pixels.show();
    for(int i=0; i<num_vectores;i++)
    {

        MPUdata.acx= acc_x[i];
        MPUdata.acy= acc_y[i];
        MPUdata.acz= acc_z[i];
        MPUdata.gyrox= gy_x[i];
        MPUdata.gyroy= gy_y[i];
        MPUdata.gyroz= gy_z[i];
        MPUdata.ac500g= ac_500[i];
        MPUdata.var_temperatura=0;

        esp_err_t result = esp_now_send(slaveAddress, (uint8_t *)
&MPUdata, sizeof(MPUdata));
        delay(110);
        if (result == ESP_OK) {
            Serial.println("The message was sent sucessfully.");
        }
        else {
            Serial.println("There was an error sending the message.");
        }
        acc_x[i]=0;
        acc_y[i]=0;
        acc_z[i]=0;
        gy_x[i]=0;
        gy_y[i]=0;
        gy_z[i]=0;
        ac_500[i]=0;
        var=0;
    }
    pixels.setPixelColor(0, pixels.Color(0, 0, 0));
    pixels.show();
}
}
```



2.2 Código del dispositivo de adquisición solo acelerómetro analógico

```
/*
Santiago Bujedo Saiz
Adquisición de datos solo acelerómetro analógico
*/

#include "I2Cdev.h"
#include "MPU6050.h"
#include <esp_now.h>
#include <WiFi.h>
#include <Adafruit_NeoPixel.h>

#define NUMPIXELS 1
#define PIN_NEOPIXEL 18
Adafruit_NeoPixel pixels(NUMPIXELS, PIN_NEOPIXEL, NEO_GRB +
NEO_KHZ800);

MPU6050 accelgyro(0x69); // <-- use for ADO high
uint8_t slaveAddress[] = {0x7C, 0xDF, 0xA1, 0x00, 0x75, 0x56};
//uint8_t slaveAddress[] = {0x0C, 0xB8, 0x15, 0xF5, 0x1E, 0xE4};
int16_t ax, ay, az;
int16_t gx, gy, gz;
float ax_m_s2, ay_m_s2, az_m_s2 ,gx_deg_s, gy_deg_s,
gz_deg_s, gScaled;
int16_t temp_raw;
float temp;
unsigned long previo, actual, interrupcion, tiempo;
int cont=0;
const int PINAc500 = 12;
float ref_V=1.25; // dependerá del divisor de tensión aquí poner el
valor que te de el acelerometro alimentado a 3.3 a 0 g
float sensibilidad=0.002; // son 2.7 mv/g pasados por el divisor de
tension
int resolution=12;
int var=0;
String dataMessage;
int accx;
const int num_vectores=25000;

int16_t ac_500[num_vectores];
#define OUTPUT_READABLE_ACCELGYRO

typedef struct valores {
    float temp;
    float ac500g;
    bool var_temperatura;
};
valores MPUdata;

typedef struct temperatura {
    float temp;
};
temperatura TEMPP;

void lecturaVALORES ()
```



```
{
  accx = analogRead(PINac500);
}

void lecturaTEMP()
{
  temp_raw = accelgyro.getTemperature();
  temp = float(temp_raw)/340 + 36.53;
}

void IRAM_ATTR empezar_enviar(){
  var=1;
}

void setup(){

#ifdef NEOPIXEL_POWER
  pinMode(NEOPIXEL_POWER, OUTPUT);
  digitalWrite(NEOPIXEL_POWER, HIGH);
#endif

  pixels.begin(); // INITIALIZE NeoPixel strip object (REQUIRED)
  pixels.setBrightness(100); // not so bright
  pixels.setPixelColor(0, pixels.Color(0, 0, 0));
  pixels.show();

  #if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
    Wire.begin();
  #elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTWIRE
    Fastwire::setup(400, true);
  #endif
  Serial.begin(115200);
  analogReadResolution(resolution);

  accelgyro.initialize();

  accelgyro.setXAccelOffset(-271);
  accelgyro.setYAccelOffset(319);
  accelgyro.setZAccelOffset(1885);
  accelgyro.setXGyroOffset(125);
  accelgyro.setYGyroOffset(64);
  accelgyro.setZGyroOffset(14);

  pinMode(0, INPUT_PULLUP);
  pinMode(34, OUTPUT);
  attachInterrupt(digitalPinToInterrupt(0), empezar_enviar,
  FALLING);

  WiFi.mode(WIFI_STA);

  if (esp_now_init() != ESP_OK) {
    Serial.println("There was an error initializing ESP-NOW");
  }
}
```



```
    return;
}

esp_now_peer_info_t slaveInfo;
memcpy(slaveInfo.peer_addr, slaveAddress, 6);
slaveInfo.channel = 0;
slaveInfo.encrypt = false;

if (esp_now_add_peer(&slaveInfo) != ESP_OK){
    Serial.println("There was an error registering the slave");
    return;
}

}

void OnSent(const uint8_t *mac_addr, esp_now_send_status_t status)
{
    Serial.print("\r\nSend message status:\t");
    Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Sent
Successfully" : "Sent Failed");
}

void loop(){
    String dataString = "";

    if (var==1 ){

        MPUdata.var_temperatura=1;
        Serial.println(" ENVIO LA TEMPERATURA");
        for (int i=0; i<5; i++)
        {
            Serial.println("espero un poquinin");
            pixels.setPixelColor(0, pixels.Color(0, 255, 0));
            pixels.show();
            delay(1000);
            pixels.fill(0x000000);
            pixels.show();
            delay(1000);
        }
        Serial.println("EMPIEZOOOOOOOOOOOOOOOOOOOOOO");
        pixels.setPixelColor(0, pixels.Color(0, 0, 255));
        pixels.show();
        previo = micros();

        for (int i=0;i<num_vectores;i++)
        {

            lecturaVALORES();

            ac_500[i]=accx;

        }
        Serial.print("La función tardó:");
        Serial.println(actual - previo );
    }
}
```



```
pixels.setColor(0, pixels.Color(255, 0, 0));
pixels.show();
lecturaTEMP();
MPUdata.temp=temp;
MPUdata.var_temperatura=1;
esp_err_t result = esp_now_send(slaveAddress, (uint8_t *)
&MPUdata, sizeof(MPUdata));
for(int i=0; i<num_vectores;i++)
{

    MPUdata.ac500g= ac_500[i];
    MPUdata.var_temperatura=0;

    esp_err_t result = esp_now_send(slaveAddress, (uint8_t *)
&MPUdata, sizeof(MPUdata));
    delay(8);
}
```



2.3 Código del dispositivo receptor Acelerómetro analógico e IMU

```
/*
Santiago Bujedo Saiz
Recepción de datos IMU acelerómetro analógico
*/

#include <esp_now.h>
#include <WiFi.h>
int i =0;
float ref_V=1.375;// dependerá del divisor de tensión aqui poner el
valor que te de el acelerometro alimentado a 3.3 a 0 g
float sensibilidad=0.002;
float gScaled;

typedef struct valores {
    float acx;
    float acy;
    float acz;
    float gyrox;
    float gyroy;
    float gyroz;
    float temp;
    float ac500g;
    bool var_temperatura;
    // int envios;
};
valores MPUdata;

typedef struct temperatura {
    float temp;
};
temperatura TEMPP;

void OnRecv(const uint8_t * mac, const uint8_t *incomingData, int
len) {
    memcpy(&MPUdata, incomingData, sizeof(MPUdata));
    i++;

    if(MPUdata.var_temperatura ==0){

        Serial.print(MPUdata.acx*(1/2048.0));
        Serial.print("; ");
        Serial.print(MPUdata.acy*(1/2048.0));
        Serial.print("; ");
        Serial.print(MPUdata.acz*(1/2048.0));
        Serial.print("; ");

        Serial.print(MPUdata.gyrox*(250.0/32768.0));
        Serial.print("; ");
        Serial.print(MPUdata.gyroy*(250.0/32768.0));
        Serial.print("; ");
        Serial.print(MPUdata.gyroz*(250.0/32768.0));
        Serial.print("; ");
        float VScaled =(MPUdata.ac500g*((2.5)/(4095)));
```




```
gScaled = (VScaled-ref_V)/sensibilidad;
Serial.println((MPUdata.ac500g-114));

}
else
{

Serial.print(MPUdata.temp);
Serial.print("; ");
Serial.print(1000);
  Serial.print("; ");
Serial.print(1000);
  Serial.print("; ");

Serial.print(1000);
  Serial.print("; ");
Serial.print(1000);
  Serial.print("; ");
Serial.print(1000);
  Serial.print("; ");
Serial.println(1000);

}

}
void setup() {
  Serial.begin(9600);
  Serial.println("aaaaaaaaaaaaaaaaaaaa");
  WiFi.mode(WIFI_STA);
  if (esp_now_init() != ESP_OK) {
    Serial.println("There was an error initializing ESP-NOW");
    return;
  }
  else {
    Serial.println("Tfasdfasdfsdfsdfializing ESP-NOW");
  }
}

esp_now_register_recv_cb(OnRecv);
}
void loop() {
}
```



2.4 Código del dispositivo receptor solo acelerómetro analógico

```
/*
Santiago Bujedo Saiz
Recepción de datos IMU acelerómetro analógico
*/
#include <esp_now.h>
#include <WiFi.h>

typedef struct valores {
    float temp;
    float ac500g;
    bool var_temperatura;
};

valores MPUdata;

typedef struct temperatura {
    float temp;
};

temperatura TEMPP;

void OnRecv0(const uint8_t * mac, const uint8_t *incomingData, int len) {
    memcpy(&TEMPP, incomingData, sizeof(TEMPP));
    Serial.print("Bytes received: ");
    Serial.println(len);
    Serial.print("temperatura: ");
    Serial.println(TEMPP.temp);
}

void OnRecv(const uint8_t * mac, const uint8_t *incomingData, int len) {
    memcpy(&MPUdata, incomingData, sizeof(MPUdata));
    // i++;
    if(MPUdata.var_temperatura ==0){
        Serial.println((MPUdata.ac500g-114));
    }
    else
    {
        Serial.println(MPUdata.temp);
    }
}

void setup() {
    Serial.begin(9600);

    WiFi.mode(WIFI_STA);
    if (esp_now_init() != ESP_OK) {
        return;
    }
    else {

    }

    esp_now_register_recv_cb(OnRecv);
}

void loop() {
}
```