DESIGN AND IMPLEMENTATION OF IOT BASED CRICKET BALL FOR PERFORMANCE IMPROVISATION OF THE PLAYER

A PROJECT REPORT
Submitted by

SHARAN SEKHAR (203001092)
SRIHARI KUMAR S (203001103)
PRAANESH R (203001069)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

ELECTRICAL AND ELECTRONICS ENGINEERING

Department of

Electrical and Electronics Engineering

Sri Sivasubramaniya Nadar College of Engineering

(An Autonomous Institution, Affiliated to Anna University)

Rajiv Gandhi Salai (OMR), Kalavakkam – 603 110

MAY 2024

Sri Sivasubramaniya Nadar College of Engineering

(An Autonomous Institution, Affiliated to Anna University)

BONAFIDE CERTIFICATE

Certified that this project titled "**DESIGN AND IMPLEMENTATION OF IOT BASED CRICKET BALL FOR PERFORMANCE IMPROVISATION OF THE PLAYER**" is the bonafide work of "**SHARAN SEKHAR** (203001092), **SRIHARI KUMAR S** (203001103) and **PRAANESH R** (203001069)", who carried out the project work under my supervision.

Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE	**SIGNATURE**
DR. V. RAJINI	**DR. R. RENGARAJ**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
	Associate Professor
Department of Electrical and Electronics Engineering	Department of Electrical and Electronics Engineering
SSN College of Engineering	SSN College of Engineering
Kalavakkam – 603 110	Kalavakkam – 603 110

Submitted for project viva-voce examination held on ……………

EXTERNAL EXAMINER	**INTERNAL EXAMINER**

ABSTRACT

Bowling in cricket demands a high level of proficiency in various aspects, both physical and mental. Among these critical skills, mastering the technique of delivering the ball stands out, and it can be enhanced through advanced training methods. These methods rely on sophisticated performance parameters. Creating an affordable smart cricket ball is a solution to address the limitations of existing systems, enabling the exploration of cricket bowling kinematics and dynamics. Historical challenges in studying cricket bowling kinematics with instrumented balls, due to technical constraints in sensor systems and electronics design, are overcome by this project. The primary objective is to develop a highly portable instrumented cricket ball capable of recording the ball's kinematics and calculating dynamic performance parameters from the kinematic data.

ACKNOWLEDGEMENT

We extend our gratitude to Padma Bhushan **DR. SHIV NADAR**, Chairman, SSN Institutions, **DR. KALA VIJAYAKUMAR**, President, SSN Institutions, **DR. V. E. ANNAMALAI**, Principal, SSN College of Engineering, and **DR. S. RADHA**, Vice Principal, SSN College of Engineering, for providing us the opportunity and facilities to work on our project and complete it.

We express our sincere thanks to **DR. V. RAJINI**, Professor and Head of Department, Electrical and Electronics Engineering for her persistent encouragement and support to work on this project.

We express our deepest gratitude to our guide, **DR. R. RENGARAJ**, Associate Professor, Department of Electrical and Electronics Engineering for his keen interest, inspiring guidance, constant encouragement of the work during the project work

We also thank our Project coordinators **DR. PREMANAND V. CHANDRAMANI**, Professor, and **DR. V. THIYAGARAJAN**, Associate Professor, for their valuable critics.

SHARAN SEKHAR

SRIHARI KUMAR S

PRAANESH R

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ADC	–	Analog to Digital Converter
DMP	–	Digital Motion Processing
EDA	–	Exploratory Data Analysis
IC	–	Integrated Circuit
IDE	–	Integrated Development Environment
I2C	–	Inter-Integrated Circuit
IoT	–	Internet of Things
LDR	–	Light-dependent Resistors
LED	–	Light Emitting Diode
ML	–	Machine Learning
RAM	–	Read Access Memory
ROM	–	Read Only Memory
SCL	–	Serial Clock Line
SDA	–	Serial Data Line
SPI	–	Serial Peripheral Interface
UART	–	Universal Asynchronous Receiver-Transmitter
USB	–	Universal Serial Bus

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

In the realm of cricket, precision and continuous skill enhancement are paramount for players seeking excellence. The "IoT-Based Cricket Ball for Performance Improvement of Players" project addresses this need by integrating Internet of Things technology into the traditional cricket ball, ushering in a new era of data-driven training and performance analysis.

Cricket, being a game of precision and strategy, demands continuous improvement in various aspects of a player's performance. Traditional training methods often lack real-time feedback and comprehensive data analysis, hindering players from reaching their full potential. This project addresses these challenges by introducing a smart cricket ball embedded with IoT capabilities.

The project's primary goal is to create a smart cricket ball equipped with sensors to capture and transmit real-time data on crucial performance metrics such as ball speed, spin, trajectory, and impact points. This live feedback loop ensures that players receive instantaneous insights into their performance during practice sessions and matches. The collected data is then subjected to in-depth analysis, generating comprehensive reports and statistics that aid players and coaches in identifying strengths, weaknesses, and areas for improvement.

The innovative features of the IoT-based cricket ball extend beyond traditional training methods. It facilitates interactive training sessions, offering players a dynamic and engaging learning experience. Coaches can tailor training programs based on individual player analytics, fostering targeted skill development. Moreover, the system aims to contribute to injury prevention by monitoring player fatigue and workload, providing insights to optimize training loads and reduce the risk of injuries.

Designed for seamless integration with existing training infrastructure, the smart cricket ball ensures accessibility for players of all levels. Compatibility with mobile applications and analysis tools further enhances its usability, making it a versatile tool for cricket enthusiasts and professionals alike.

This project represents a significant leap forward in cricket training technology, empowering players with the tools they need to refine their skills, make data-driven decisions, and ultimately elevate their performance on the field. The fusion of IoT and cricket promises an exciting future for the sport, fostering a new era of precision, analytics, and excellence.

OBJECTIVES

The development of the project "IoT-Based Cricket Ball for Performance Improvement of Players" is essential due to several compelling needs within the realm of cricket training and performance enhancement. These needs underscore the significance and potential impact of integrating IoT technology into cricket balls for the betterment of players. Here are some key reasons emphasizing the need for this study:

- Real-Time Performance Feedback: Traditional cricket training methods often lack immediate and accurate feedback on a player's performance. The IoT-based cricket ball addresses this gap by providing real-time data on critical performance metrics such as ball speed, spin, trajectory, and impact points.
- Data-Driven Decision-Making: In the competitive world of cricket, data-driven decision-making is becoming increasingly crucial for players and coaches. This project enables the collection and analysis of comprehensive data, empowering players to make informed decisions about their training routines and strategies during matches.
- Adaptability and Accessibility: The project ensures that the IoT-based cricket ball is designed to seamlessly integrate with existing training infrastructure. The adaptability and accessibility of this technology make it viable for players of all levels, democratizing advanced training methods in cricket.

ORGANIZATION OF REPORT

This report is structured into six chapters which are organized in the following manner

Chapter 1 contains the introduction and the motivation behind the thought process of this project. Additionally, the objectives of designing and implementing an IoT based cricket ball are presented.

Chapter 2 presents the literature review of the project and provides a brief summary for each Literature reviewed.

Chapter 3 includes the details about the hardware components that are being used. It shows the comparison of different components and explains the reason behind the choice of the component.

Chapter 4 introduces the information regarding the different types of data analysis present and the procedure required to analyze the data.

Chapter 5 proposes the implementation of the software along with the hardware to obtain results from the components and validate them.

Chapter 6 contains the conclusion of the project report summarizing the whole project and briefly explains the future scope of the project.

CHAPTER 2

LITERATURE SURVEY

2.1 INTRODUCTION

These literature surveys highlight the need for cost-effective smart cricket balls to enhance training. The proposed ball incorporates advanced technologies, including miniature sensors, impact-resistant materials, shock-absorbing foam, and a leather exterior for recording kinematics and calculating dynamic performance parameters. These researches reveals a comprehensive circuit with a power supply, a microcontroller with gyros a radio transceiver, and storage. These studies emphasizes the crucial gyro calibration process for aligning the ball's axes using a calibration board driven by an electric motor. Insights include the null bias of gyros and the alignment ensuring specific output voltages for counterclockwise and clockwise rotations. The continuous data is converted to revolutions per second (rps) and analyzed as a time series to understand angular velocity changes. These studies also provide intriguing insights into the efficiency of different bowling techniques, emphasizing the applicability of the bowling potential metric to spin bowlers rather than fast bowlers, for whom speed outweighs spin.

2.2 LITERATURE SURVEY

Doljin B and Fuss F.K (2015) has presented a smart cricket ball aimed at analyzing the kinematics and dynamics of bowling. It highlights the need for low cost smart cricket balls to improve training. The ball will record kinematics and calculate the dynamics performance parameters. The construction includes leather, impact resistant nylons, shock absorbing foam and miniaturized sensors. The useful extract from this paper is that the circuit consists of power supply, microcontroller 3 rate gyros which measure upto +50rps, radio transceiver and storage. Main components of the power supply circuit are Qi V1.0 wireless power receiver, proximity switch, voltage booster, 3.7V lipo battery and charger. The average power consumption of the whole circuit is about 30ma and total running time is ~5 hours based on 150mah battery capacity. The AVR microcontroller has a built-in 10 bit ADC which converts sensor analog output signal to digital format and stores it to a 64M bit flash memory. Data sampling frequency is set to 815Hz. The memory can hold upto 28 minutes. The firmware was developed as an analog voltage data logger to record output of the three high speed gyroscopes at sampling rate of 815Hz.

Moving to the field of calibration, Fuss F.K, Smith R.M and Subic A. (2012) has presented the methods to calibrate the axes of the ball to the orthogonal coordinate system using a calibration board. They published a compensated center of mass structure to yield balanced structure to prevent ball deviation on leveled ground. Before incorporating the gyros, they were aligned to the orthogonal coordinate system through a calibration board. The board was driven by an electric motor. Speed of the motor was measured and the accuracy was assessed at different rpm. After which, the position of its centre of mass was tested by the direction in which the ball deviated on level ground, and additional masses were added for balancing. The useful extract is that the gyro output is typically 2.5 V at zero angular velocity which is called the null bias. The axis of the gyros was aligned such that counterclockwise rotation returns an output voltage larger than 2.5 V and clockwise rotation returns an output voltage smaller than 2.5 V. They found out each gyro a value of 7.2 corresponded to 1 rps. This value is temperature dependent and increases linearly by 0.085 with every temperature rise of 10°C.

Franz Konstantin Fuss, Batdelger Doljin and René E. D. Ferdinands (2020) have contributed to a study related to profiling of bowlers in cricket, with a focus on assessing two key parameters: spin (angular velocity) and speed (translational velocity). They have evaluated bowlers based on these parameters without considering efficiency. The useful extract from this paper is that wrist-spin deliveries are more efficient than fingerspin ones, whereas backspin (fast bowling) deliveries are the least efficient. This explains why the average 100 Tmax/ωmax of wrist-spinners was slightly smaller than 1; the one of finger-spinners was slightly greater than 1, and the one of fast bowlers exceeded 1.5. The bowling potential is more applicable to spin bowlers rather than to fast bowlers. Fast bowlers were included in this study only for comparative reasons. For fast bowlers, speed is more important than spin.

In the comprehensive review by Yehuda Weizman et al. (2020) of a Frisbee equipped with a triaxial gyroscope thrown at low angular velocities, within the gyroscope's measurement limits. The gyroscope data analysis provided insights into the Frisbee's behavior. The reduction in wobble led to a decreased drag force, resulting in a smaller spin decay. Initial spin decay was in the range of 1.12 to 0.31 rev/s^2. The spin decay eventually asymptotes to a range of 0.11 to 0.01 rev/s^2. The information gained from this paper is that the study focuses on analyzing the flight behavior of a Frisbee using a triaxial gyroscope. To conduct the analysis, a Frisbee with specific characteristics was equipped with an IMU. The IMU was attached to the Frisbee's underside to measure angular velocities and other parameters. Data from four throws in still air conditions were collected and analyzed. Key parameters, including peak angular velocity, spin rate decay, angular acceleration, torque, power, and wobble, were calculated using specialized software.

Fuss F.K and Smith R.M (2014) focuses on seam bowling in cricket and the factors that affect the accuracy of the placement of the spin axis when releasing the ball. They mention that the spin vector rapidly processes into the torque vector, and the seam's orientation can lead to turbulence and affect aerodynamic forces, disrupting the swing. It found that the spin axis can deviate from the optimal position due to the amount of torque applied. The study used smart cricket balls with gyroscopes to analyze four spin bowlers' data. The takeaway is that the seam bowling in cricket involves the orientation of the seam of the ball relative to its flight path. In seam bowling, the seam introduces a roughness element, causing different flow regimes on either side of the ball, resulting in a side force that produces contrast swing. The angle of the seam relative to the flight path is crucial. If the seam is angled slightly towards the smooth side in seam bowling, it can disrupt the flow, decrease the side force, and prevent the ball from swinging effectively. To achieve optimal seam bowling, the spin axis should be perpendicular to the seam to prevent seam wobble. The torque vector should be 10-15 degrees off the pole to ensure the spin axis reaches the pole at the point of release. The collected data were processed to determine key parameters that influence seam bowling accuracy, including the deviation angle of the precessing spin vector.

Balbudhe Pravin, Khandelwal Brijesh and Solanki Sachin (2021) further contribute to the discussion of role of technology in game coaching, particularly in assisting coaches with player performance and accuracy assessment and delves into tracking technologies used in different sports. Lastly, outlines future research directions in the field of smart sports technology. The important keynotes is that the Surface mount components are selected to reduce the magnitude of printed circuit panels.The smart ball was powered up without having to open it using a magnetic proximity switch. Since the components i.e, Bluetooth transceiver, wireless charging & proximity switch are fully sealed, the smart cricket ball meets the IP67 norm. As a result, the ball can be used outside in any weather. Their research includes the comparative analysis of three methods i.e, Sensor, Camera, Wearable in terms of accuracy, cost, feasibility, speed of information detection and information accuracy. It is concluded that Sensor techniques have higher accuracy & feasibility and also they have lesser cost compared to camera and wearable methods. Camera methods have moderate accuracy and feasibility. Wearable methods have higher accuracy and viability but they have moderate cost.

2.3 SUMMARY

This chapter includes brief information about the papers referred and studied for the development of our project under the literature survey. These studies collectively contribute to the evolving landscape of cricket and its decision making system, showcasing the advancements in ML and IoT integration, optimal feature selection, and application of accurate performance analysis of the player.

CHAPTER 3

HARDWARE IMPLEMENTATION OF SMART-CRICKET BALL

3.1 INTRODUCTION

In the aim of studying the design and implementation of an IOT-based smart cricket-ball and its effectiveness over other methodologies, the study of materials and components to be used is essential. After a careful evaluation and selection of components that not give accurate results but also fall within the economical constraints, we have come to the conclusion of using the components that are discussed in this chapter.

3.2 MICROCONTROLLER

A microcontroller is a compact IC that contains a processor core, memory (RAM and/or ROM), and programmable input/output peripherals. It is designed to execute specific tasks and control electronic devices based on the instructions programmed into its memory.

Microcontrollers find widespread use in various applications, from simple embedded systems to complex control systems. In our application, we have to consider a microcontroller that is small enough to fit inside a ball and as well as be fast enough to sample data up to 500KHz, process the data, and simultaneously send the data wirelessly to another computer without losing data or precision.

3.2.1 ESP8266

The ESP8266 is a powerful and versatile microcontroller module that builds upon the success of the ESP32. It incorporates a range of features suitable for a variety of applications, especially in the IoT space. Here are some key features of the ESP8266:

- Integrated Wi-Fi: The ESP8266 module comes with built-in Wi-Fi connectivity, allowing devices to connect to wireless networks and communicate over the internet.
- Clock Speed: Typically operates at clock speeds ranging from 80 MHz to 160 MHz, offering a balance between performance and power consumption.
- Memory: The ESP8266 features both program memory (Flash) and data memory. Flash memory is used to store the program code, and RAM is available for runtime data storage.
- Low Power Consumption: Designed to be power-efficient, making it suitable for battery-powered and energy-conscious applications.
- Communication Interfaces: Supports serial communication through UART and can be programmed with the Arduino IDE. It also supports SPI and I2C communication protocols.
- Cost-Effective: The ESP8266 is known for its affordability, making it a popular choice for projects with budget constraints.

The following fig 3.1 and fig 3.2 show the module and the pinout diagram of ESP8266 respectively.

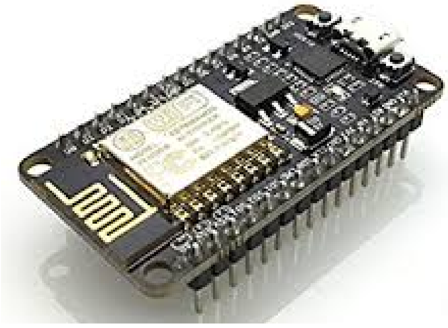


Figure 3.1 ESP8266

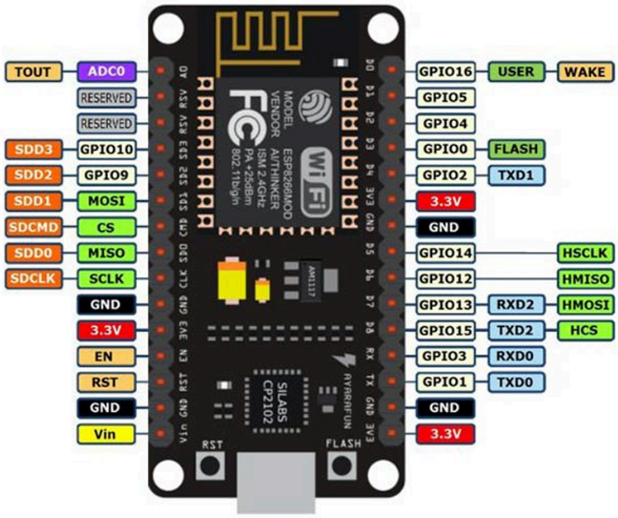


Figure 3.2 ESP8266 Pinout Configuration

3.2.2 Comparison with other Modules

From table 3.2, we can come to the clear conclusion that ESP32 is much more powerful compared to ESP8266, but the power required by our standards is well met by ESP8266 itself. Therefore, to avoid excess battery usage and expenses, we decided to implement the smart cricket-ball using ESP8266.

Table 3.1 Comparison between ESP8266 and ESP32

MCU	ESP8266	ESP32
Processor	Xtensa Single-core 32-bit L106	Xtensa Dual-Core 32-bit LX6 with 600 DMIPS
Bluetooth	No	Bluetooth 4.2 and BLE
802.11 b/g/n Wi-Fi	HT20	HT40
Typical Frequency	80 MHz	160 MHz
SRAM	No	Yes
Flash	No	Yes
GPIO	17	36
Hardware /Software PWM	None / 8 channels	None / 16 channels
SPI/I2C/I2S/ UART	2/1/2/2	4/2/2/2
ADC	10-bit	12-bit
CAN	No	Yes
Ethernet MAC Interface	No	Yes
Touch Sensor	No	Yes
Temperature Sensor	No	Yes
Hall effect sensor	No	Yes

3.3 SENSORS

Sensors are devices or instruments that detect and measure physical properties or changes in the environment and convert this information into signals that can be interpreted, displayed, or used for various applications. Sensors play a crucial role in collecting data for a wide range of fields, including science, industry, healthcare, and consumer electronics. There are numerous types of sensors, each designed to measure specific physical parameters. Here are some common types of sensors and the properties they measure:

- Temperature Sensors: Measure the temperature of an object or environment. Examples include thermocouples, thermistors, and infrared sensors.
- Pressure Sensors: Detect changes in pressure. Common types include barometers and piezoelectric sensors.
- Proximity Sensors: Detect the presence or absence of an object without physical contact. Examples include infrared sensors, ultrasonic sensors, and capacitive sensors.
- Motion Sensors: Detect movement or acceleration. Types include accelerometers, gyroscopes, and motion detectors.
- Light Sensors: Measure the intensity of light in the surrounding environment. Examples include photodiodes, phototransistors, and LDRs.

Sensors are essential components in the IoT era, where they are widely used to gather data for smart devices, environmental monitoring, industrial automation, healthcare, and various other applications. Advances in sensor technology continue to drive innovation across diverse industries, enabling improved data collection, automation, and decision-making processes.

3.3.1 MPU-6050

The MPU-6050 is a popular Inertial Measurement Unit that combines a 3-axis gyroscope and a 3-axis accelerometer into a single chip. It's used to measure motion, orientation, and acceleration of an object. The MPU-6050 is commonly used in various applications, including robotics, drones, wearable devices, and motion sensing applications. Here are the key features of the MPU-6050:

- 3-Axis Gyroscope: Measures angular rate (rotational speed) around the X, Y, and Z axes.
- 3-Axis Accelerometer: Measures acceleration along the X, Y, and Z axes.
- Digital Motion Processor: The MPU-6050 includes a Digital Motion Processor, which offloads complex sensor fusion calculations, reducing the load on the host microcontroller.
- Communication Interface: Typically communicates with a microcontroller or other devices using the I2C communication protocol.
- Gyroscope Sensitivity and Accelerometer Range: The sensitivity of the gyroscope and the range of the accelerometer can often be configured through software settings.
- Motion Processing: Capable of providing quaternion data for orientation estimation, as well as raw sensor data for gyroscope and accelerometer.
- Applications: Used in a variety of applications, including motion sensing for consumer electronics and stabilization systems.

The fig 3.3 and fig 3.4 show the MPU-6050 and MPU-9250 modules respectively. It can be clearly seen that there are more pins in MPU-9250 compared to the MPU-6050. Additionally, fig 3.5 shows the pin diagram of the MPU-6050 used.

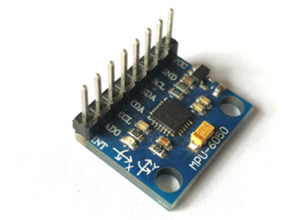


Figure 3.3 MPU-6050

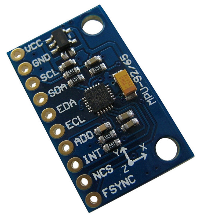


Figure 3.4 MPU-9250

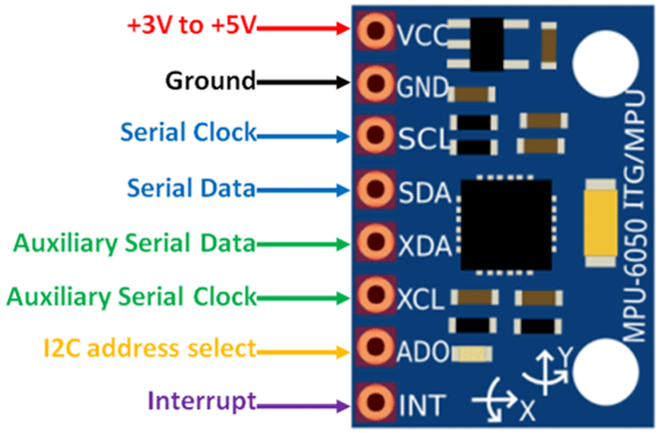


Figure 3.5 MPU6050 Pinout Configuration

MPU6050 Features:-

- MEMS 3-axis accelerometer and 3-axis gyroscope values combined
- Power Supply: 3-5V
- Communication : I2C protocol

- Built-in 16-bit ADC provides high accuracy
- Built-in DMP provides high computational power
- Can be used to interface with other IIC devices like magnetometer
- Configurable I2C Address
- In-built Temperature sensor

3.3.2 Comparison with other Modules

Some key specifications and features of the MPU-6050 and MPU-9250 are mentioned in table 3.3. Depending on your specific project requirements, you may choose one over the other based on factors such as communication protocol, gyroscope range, accelerometer range, and the inclusion of a magnetometer.

Table 3.2 Comparison between MPU-6050 and MPU-6500

Feature	MPU-6050	MPU-6500
Gyroscope range	±250, ±500, ±1000, ±2000 dps	±250, ±500, ±1000, ±2000, ±4000, ±8000, ±12000 dps
Accelerometer range	±2g, ±4g, ±8g, ±16g	±2g, ±4g, ±8g, ±16g
Communication Protocol	I2C	I2C,SPI

3.4 INTERFACING ESP8266 WITH MPU-6050

As you can see from fig 3.6, the MPU6050 has 8 terminals but we will only require the first four pins highlighted in yellow. These are VCC, GND, SCL, and SDA. The table shows the connections between the two modules. The VCC pin is connected with the 3.3V from the ESP8266 module to power up. Both the grounds of the two devices are connected in common. The SCL pin of MPU6050 is connected with the default SCL pin of ESP8266 NodeMCU. The table 3.4 shows the connections of pins from MPU6050 Module to ESP8266.

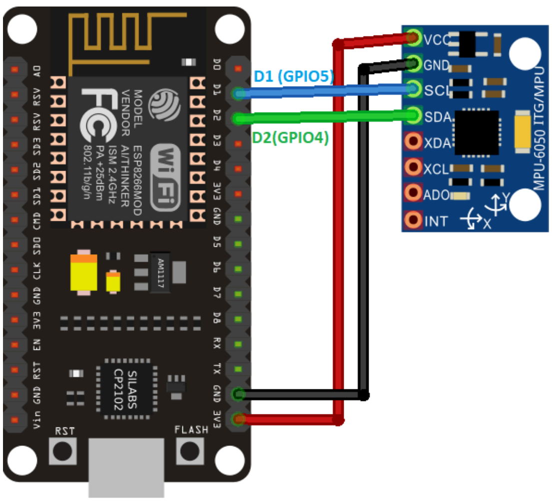


Figure 3.6 MPU-6050 Interfacing with ESP8266 MCU

Table 3.3 Connection pins between Sensor and Microcontroller

MPU6050 Module	ESP8266 NodeMCU
VCC	3.3V
GND	GND (common ground)
SCL	GPIO5 (I2C SCL)
SDA	GPIO4 (I2C SDA)

Once the connection is done the microcontroller and sensor are both placed together with the help strap so that there won't be any unnecessary vibration. After strapping the microcontroller and sensor are kept inside the prototype ball shown in fig 3.7.

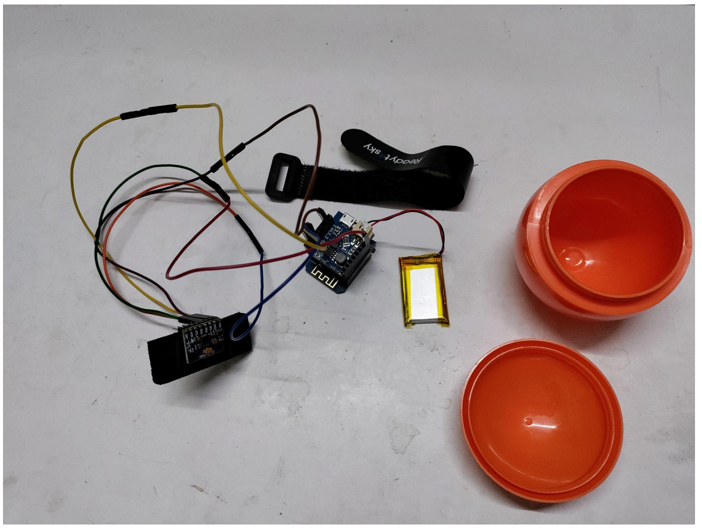


Figure 3.7 All components with prototype

3.5 SUMMARY

This chapter tells about the functions and features of the microcontroller (ESP8266/ESP32) and the sensors (MPU-6050/MPU-6500) and uses of both. It then proceeds to compare the available choices and depending on the requirements of the project the final choice of component to be used is made. It also gives an idea on how the interfacing of ESP8266 and MPU-6050 through connections diagrams and port connections using I2C protocol lines.

CHAPTER 4

SOFTWARE IMPLEMENTATION OF SMART CRICKET BALL

4.1 INTRODUCTION

In this chapter the process of implementing the code and software related tasks has been mentioned. The first step in our project is the development of the hardware to let us collect the data. For which, all the parts mentioned earlier are connected together and the necessary code is written to obtain the data and later the results are validated. Before beginning with the implementation of the hardware circuits, simulation has been done in order to verify the connections and the working of the system. The project has been built under the base of the block diagram that is shown in fig 4.1.

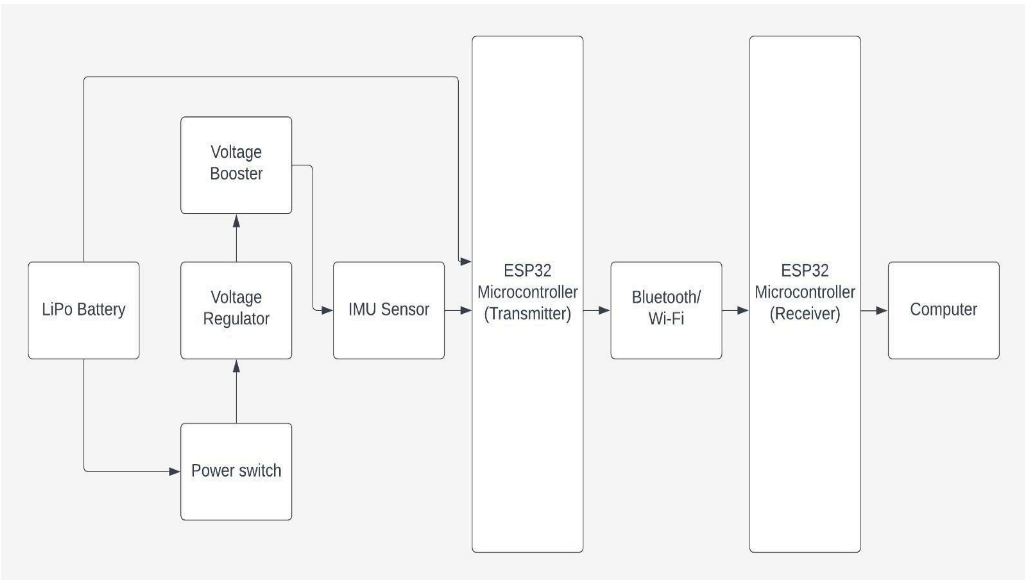


Figure 4.1 Block diagram of circuit

4.2 SIMULATION

The basic wiring diagram and the simulation of the circuit involving 2 ESP8266 (Microcontroller), MPU6050 (Gyroscope Sensor), an LED Display, and the required wires has been done and is illustrated below in the fig 4.2. We have used 2 microcontrollers in order to wirelessly transfer data between them. The yellow wire and Blue wires correspond to the SCL and SDA pins used for communication between the sensor and the microcontroller. The black and red wire is used for ground and supply. For a simple scenario, we have used a LED display to display the accelerometer reading alone that has been transmitted wirelessly to the other ESP8266.

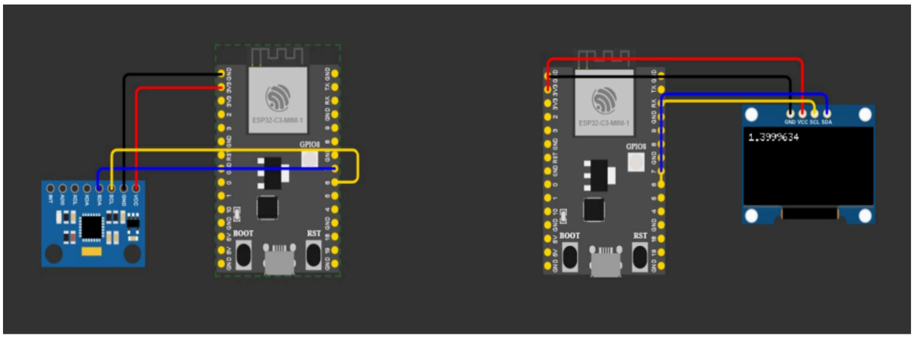


Figure 4.2 Basic Wiring and simulation diagram

4.3 IMPLEMENTATION

After successful implementation of the simulation, the next big step is to implement the same with real hardware components and collect the data. The implementation is made up of hardware implementation and Software implementation. The hardware implementation is explained in Chapter 3 along with the components and connections required for this project. This is then followed up by the software implementation. The software implementation consists of collection of data and analyzing the data for the improvement analysis of the player. This whole process has been divided into multiple steps.

4.3.1 Connections and Pin Diagram

The connections for this project are fairly simple mainly due to the usage of I2C protocol for transmitting and receiving data between the sensor and the microcontroller. I2C is a widely used serial communication protocol that facilitates communication between integrated circuits or other peripherals in embedded systems. I2C is a synchronous, multi-master, and multi-slave communication protocol. Additionally, I2C supports multiple masters on the same bus, enabling a flexible and scalable network of interconnected devices. The protocol is known for its simplicity and ease of implementation, making it accessible for both hardware designers and software developers. I2C also facilitates communication at relatively high speeds, contributing to efficient data transfer between devices. Overall, the simplicity, versatility, and scalability of I2C make it a preferred choice for inter-device communication in a wide range of applications, from consumer electronics to industrial systems.

It requires only 2 pins for communication and 2 additional pins for supply and ground. The pins diagram of ESP8266 and MPU6050 has been shown in fig 4.3 and fig 4.4 respectively. The SCL and SDA of MPU6050 is given to the GPIO4 and GPIO5 pins of the ESP8266 to establish the I2C communication. The Vcc and Gnd pins are taken from the 3.3V and Gnd pins of the ESP8266 respectively.

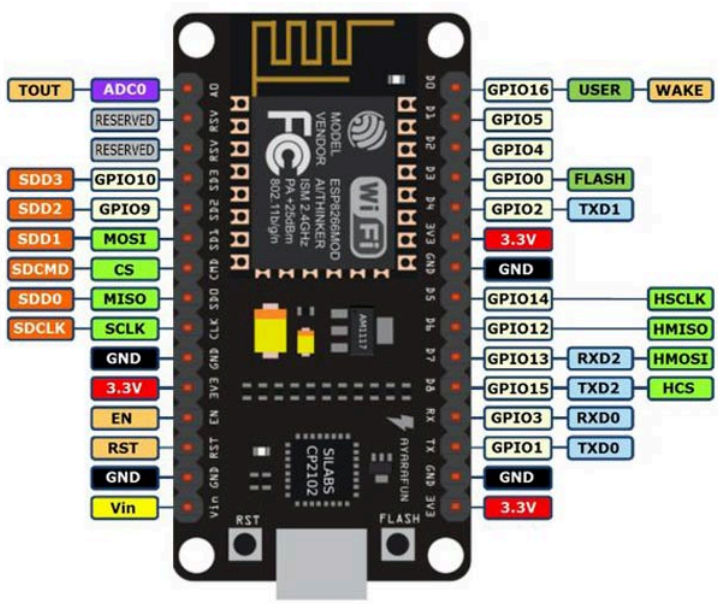


Figure 4.3 ESP8266 Pin Diagram

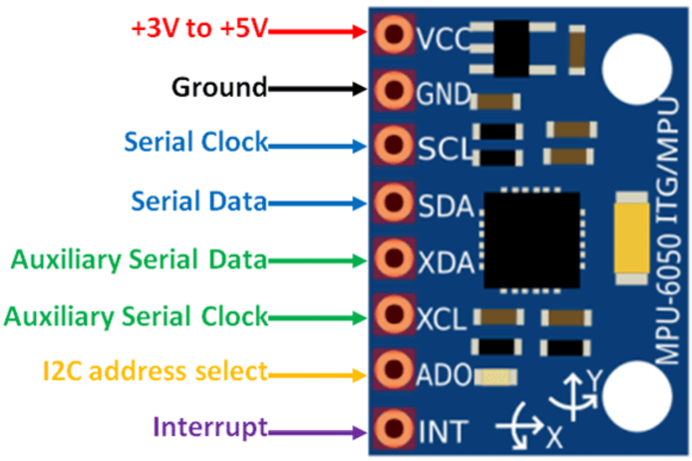


Figure 4.4 MPU6050 Pin Diagram

4.3.2 Collection of Raw Data and Converting into Useful data

Once the connections have been made, it is important to code in order to get useful data out of the components. But before coding, it is necessary to import all necessary libraries in order for the components to work properly. In our case we will be using the “Adafruit Unified Sensor” libraries. After which the code is written and uploaded to the ESP8266 through USB cable. On successful upload of the code, we can read the values being output from the sensor in the Serial Monitor window of the Arduino IDE which is shown in fig 4.5.

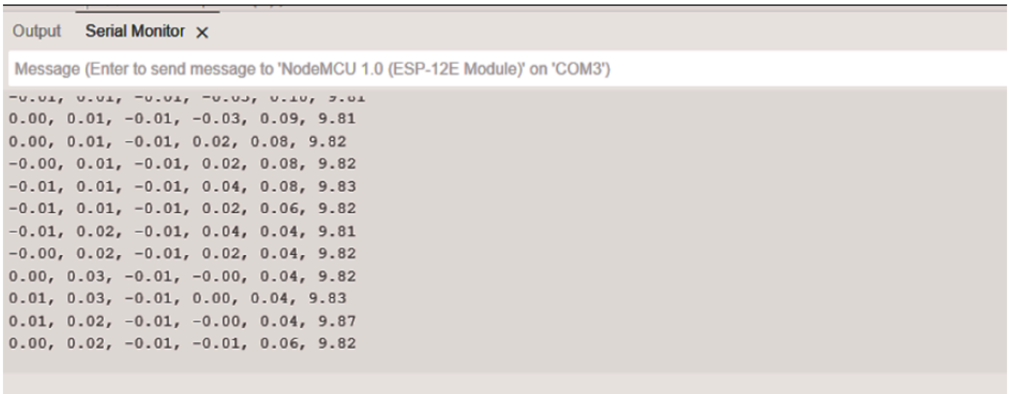


Figure 4.5 Sensor readings print in Serial Monitor

There are totally 6 readings provided by the 6-axis Gyroscope. First three readings correspond to the 3 axis readings (X, Y, Z) from the Gyroscope and the following three readings are the 3 axis readings (X, Y, Z) from the accelerometer. You can see that the Z axis reading of the Accelerometer will be close to 10 due to gravity.

Initially, these readings are not in terms of m/s^2 and radians. These are rather converted to standard units by multiplying with standard conversion measurement units to actually obtain the useful data.

4.3.3 Calibration

Any device or instrument requires calibration before use and its importance lies in ensuring the accuracy, reliability, and traceability of measurements and instruments. Here, in order to know the starting point, or rather the reference axis to measure from, calibration is done.

There are multiple ways to calibrate but here we rather go with a simple offset method. During idle or stationary position, the sensor is ideally supposed to produce zero movement and zero error readings on all its axes except for the Z-axis of the Accelerometer due to presence of gravity. The graph before calibration is shown in fig 4.6.

The first few seconds after initialization of the ESP8266 is dedicated for the process of calibration of the sensor. For the first 5 seconds, the sensor collects as many samples as possible and averages the readings to determine the offset. Any reading that is not zero contributes to error and offset and this averaged offset is subtracted from the concurrent readings that are measured during actual application.

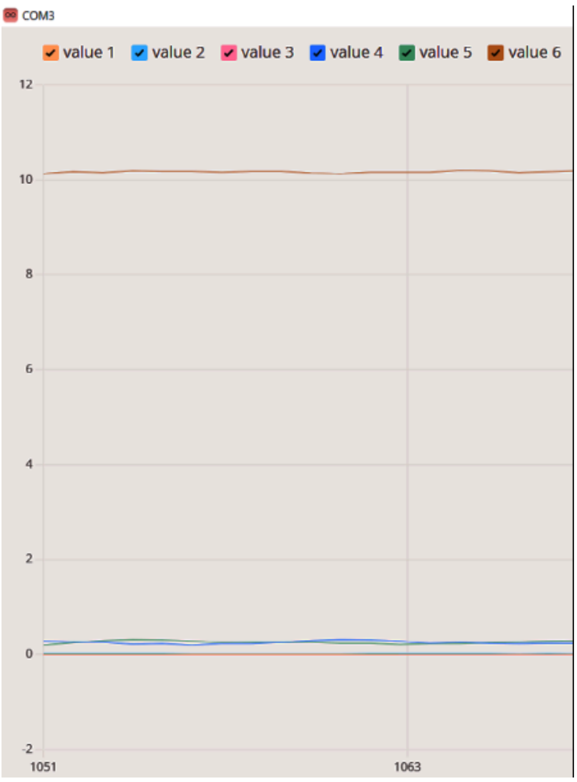


Figure 4.6 Uncalibrated results of MPU6050

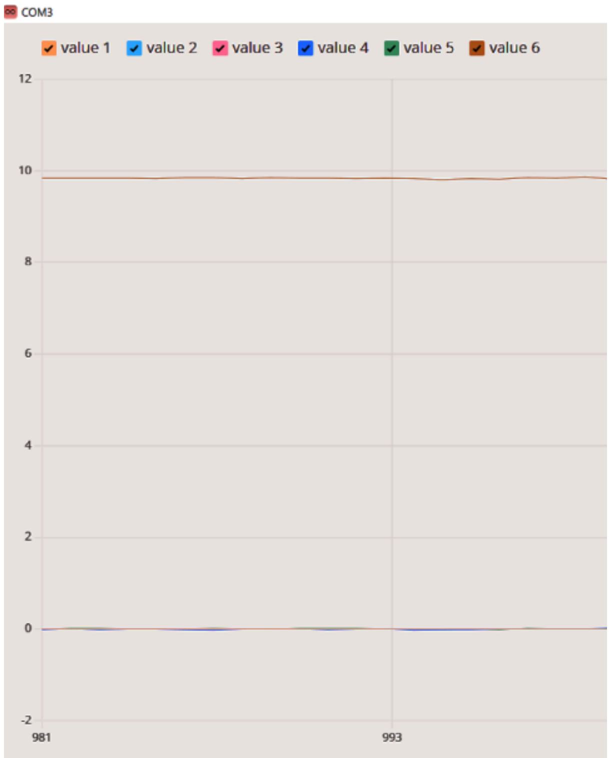


Figure 4.7 Calibrated results of MPU6050

From fig 4.7, we can clearly see that the calibrated readings show values much closer to their idle values and this can help prevent errors or mismatch in readings during application.

4.3.4 Filters

After calibration, it is important to apply filters to the sensor since it is prone to lots of noise due to nearby vibrations and also due to the presence of drift within the sensor leading to inaccurate readings. These filters are not physical filters but perhaps just a few lines of code that remove readings that seem anomalous from the remaining set of data values. Therefore, to enhance the performance of the MPU6050, various filters can be applied. Common primary filters that are used with the MPU6050 are mentioned below:

- Low-Pass Filter: The MPU6050 features an onboard low-pass filter that can be configured to reduce high-frequency noise in the accelerometer and gyroscope data. It helps in eliminating high-frequency vibrations and noise, resulting in smoother and more accurate sensor readings.Users can adjust the cutoff frequency based on their specific application requirements.

- Digital Motion Processing: This is a feature of the MPU6050 that includes built-in algorithms for sensor fusion, which combines data from the accelerometer and gyroscope to calculate a more accurate and stable orientation estimate. It can handle sensor calibration, filtering, and fusion internally, providing a convenient solution for obtaining quaternion or Euler angle outputs.
- Kalman Filter: The Kalman filter is a widely used algorithm for sensor fusion that combines noisy sensor measurements to estimate a more accurate and reliable state. It is often employed in conjunction with the MPU6050 to improve the accuracy of orientation data by considering the noise characteristics of both the accelerometer and gyroscope.
- Complementary Filter: The complementary filter is another sensor fusion technique that combines accelerometer and gyroscope data. It uses a weighted combination of accelerometer and gyroscope readings, leveraging the high-pass characteristics of the gyroscope to handle short-term changes and the low-pass characteristics of the accelerometer for long-term stability.

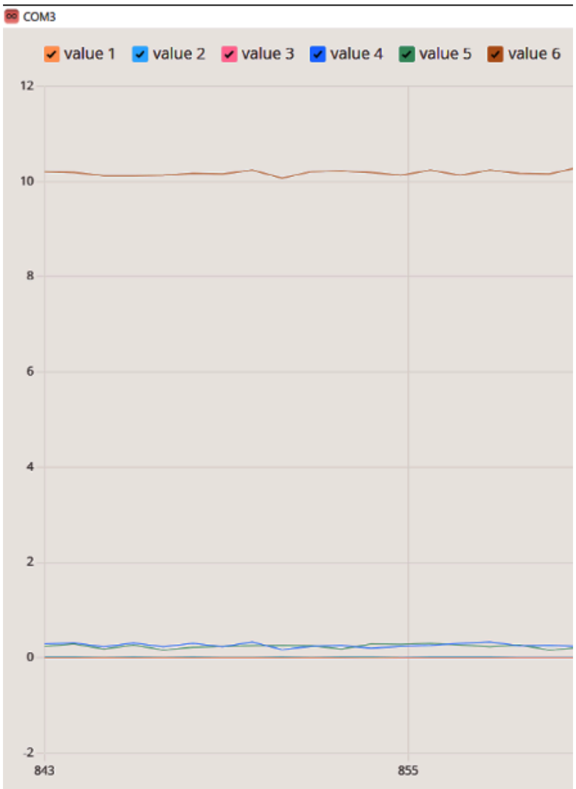


Figure 4.8 Sensor readings without filters

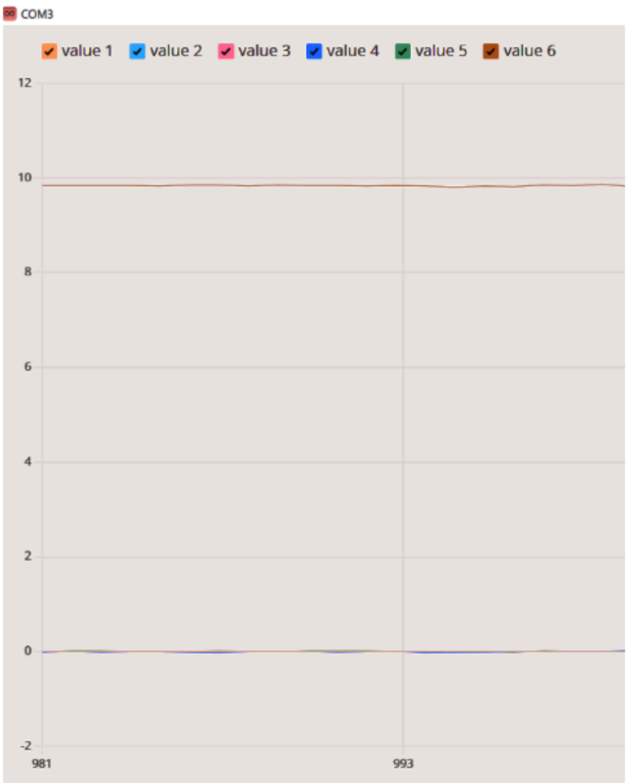


Figure 4.9 Sensor readings with LPF filter and calibration

The choice of the "best" filter depends on the specific requirements and constraints of the application. The DMP feature is convenient for users who want a ready-made solution for sensor fusion. The Kalman filter and complementary filter, on the other hand, provide more flexibility for customization but may require additional implementation effort.

In our case we have gone with a simple Low Pass Filter, since it requires low data processing requirements and the results are decently accurate. The difference in the smoothness of the curves can be seen in fig 4.8 and fig 4.9, which show before and after applying filters, respectively.

4.3.5 Storing Data in CSV File

The data that is being output from the sensor is live data and is seen through the Serial Monitor instantaneously. In order to process the data and analyze it for further use, we need to store the data. In order to store the continuous stream of data, a python script has been written to sample the required number of data that is being fed through the COM port of the computer and store it in an Excel file.

Once the data has been stored, basic preprocessing techniques such as cleaning the data to remove erroneous data or missing data from the Excel is done. The fig 4.10 below shows the Excel file created by running the python script application, which is shown in fig 4.11.

Packet number	Gyroscope X (deg/s)	Gyroscope Y (deg/s)	Gyroscope Z (deg/s)	Accelerometer X (g)	Accelerometer Y (g)	Accelerometer Z (g)	Magnetometer X (G)	Magnetometer Y (G)	Magnetometer Z (G)
1	0	-0.01	0.02	0.03	0.07	9.83	0	0	0
2	0	0	0.02	0.01	0.05	9.84	0	0	0
3	0	-0.01	0.02	0.01	0.06	9.84	0	0	0
4	0	-0.01	0.02	0.06	0.06	9.85	0	0	0
5	0	-0.01	0.04	0.04	0.06	9.87	0	0	0
6	0	-0.01	0.06	0.08	0.1	9.83	0	0	0
7	0	-0.01	0.08	0.08	0.12	9.82	0	0	0
8	0	-0.02	0.1	-0.02	0.12	9.82	0	0	0
9	0	-0.01	0.11	0.02	0.11	9.84	0	0	0
10	0	-0.01	0.09	0.03	0.12	9.85	0	0	0
11	0	0	0.09	-0.03	0.05	9.86	0	0	0
12	0	0	0.09	-0.05	0.02	9.86	0	0	0
13	0	-0.01	0.11	-0.06	0.01	9.85	0	0	0
14	-0.01	0	0.12	-0.01	0.08	9.82	0	0	0
15	-0.01	0.01	0.1	0.05	0.08	9.8	0	0	0
16	-0.02	0.02	0.09	0.12	0.2	9.84	0	0	0
17	-0.03	0.02	0.06	0.06	0.11	9.84	0	0	0
18	-0.05	0.03	0.06	0.09	0.13	9.87	0	0	0
19	-0.04	0.03	0.05	0.08	0.06	9.8	0	0	0
20	-0.05	0.02	0.05	0.11	0.09	9.87	0	0	0
21	-0.05	0.01	0.04	0.04	0.13	9.84	0	0	0
22	-0.05	0.01	0.03	-0.01	0.1	9.85	0	0	0
23	-0.05	0.02	0.01	-0.02	0.13	9.84	0	0	0
24	-0.11	0.01	-0.08	-0.44	0.18	9.92	0	0	0
25	-0.12	0.02	-0.02	0.18	-0.09	9.69	0	0	0
26	-0.06	0.03	-0.03	0.12	-0.01	9.76	0	0	0
27	-0.04	0.02	-0.01	0.01	-0.02	9.78	0	0	0
28	-0.04	0.03	0	0.01	0.03	9.83	0	0	0
29	-0.03	0.04	0.01	0.06	0.05	9.81	0	0	0
30	-0.02	0.04	0.02	0.05	0.06	9.85	0	0	0

LoggedData_CalInertialAndMag1

Figure 4.10 Excel file with stored data

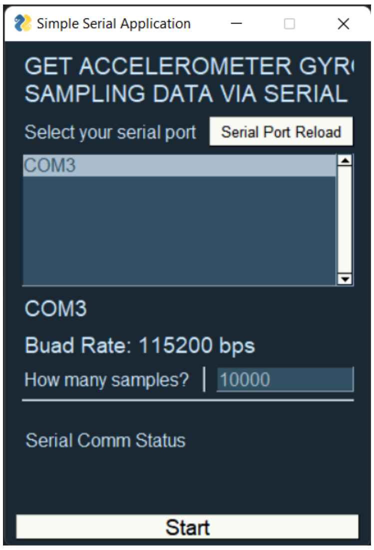


Figure 4.11 Python Script application to sample the data

4.3.6 Analysis of Motion in MATLAB

With the data being stored in an Excel file, we can import the excel file into MATLAB for preprocessing the data, further removing noises. First the data is loaded into MATLAB where gyroscope and accelerometer values are plotted as shown in fig 4.12 and fig 4.13 respectively.

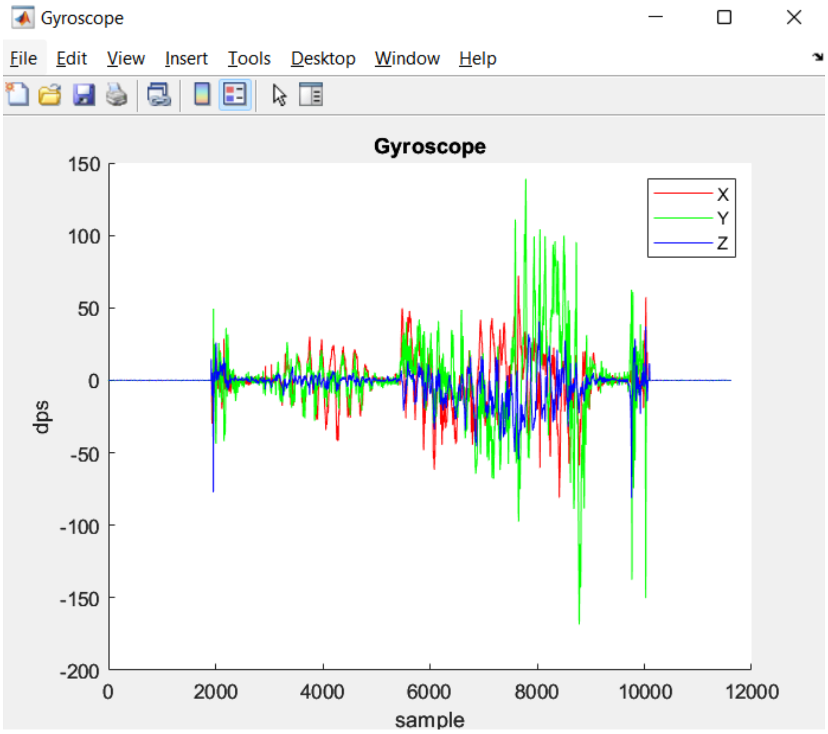


Figure 4.12 Gyroscope readings graph

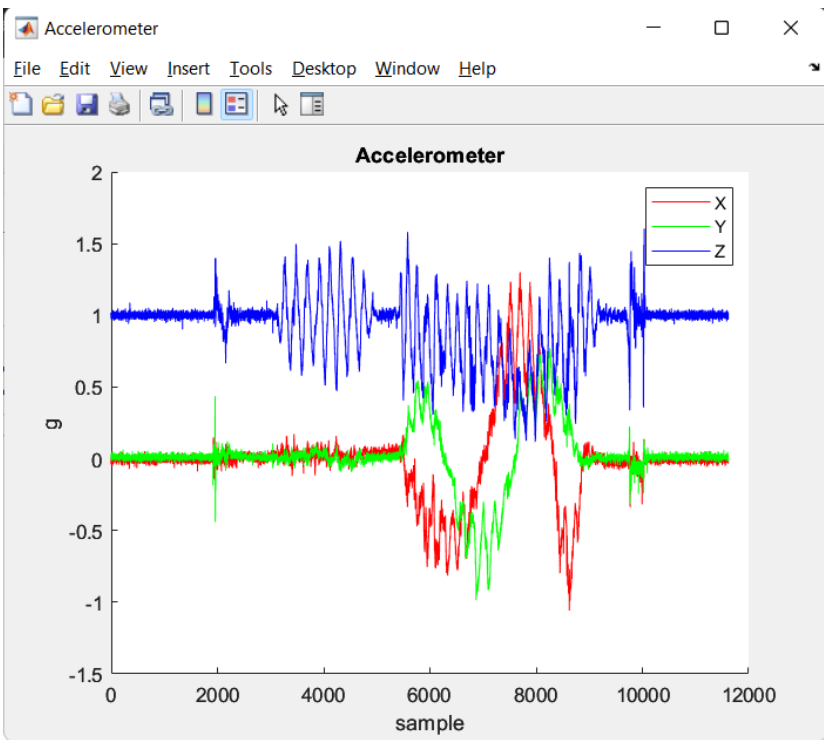


Figure 4.13 Accelerometer readings graph

With this set of readings, we can proceed to use AHRS (Altitude and Heading Reference System) algorithm to convert to linear values. This involves integration of acceleration values in fig 4.14. In simple terms integration means summation of the values and here, the values are continually summed to produce linear velocity as shown in fig 4.15. And by once again integrating the linear velocity, we can arrive at linear displacement or linear position values. From this then the values are used to convert into 3-D space and finally an 6DOF animation is made with the help of linear position values to produce orientation tracking for n number of samples. Through this, we can visualize the movement of the ball in 3D space as shown in fig 4.17.

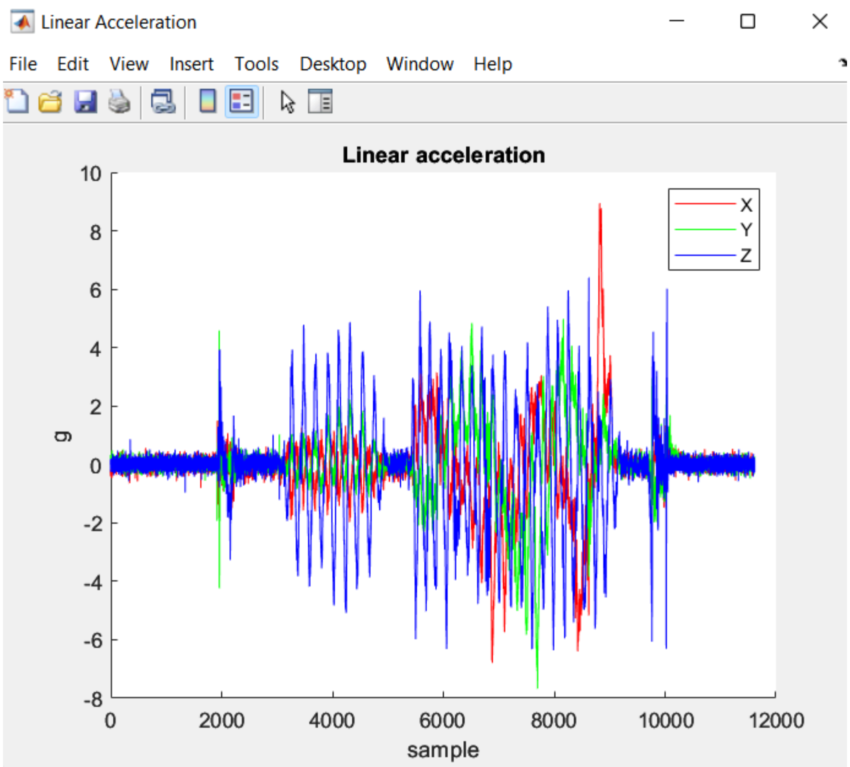


Figure 4.14 Linear acceleration

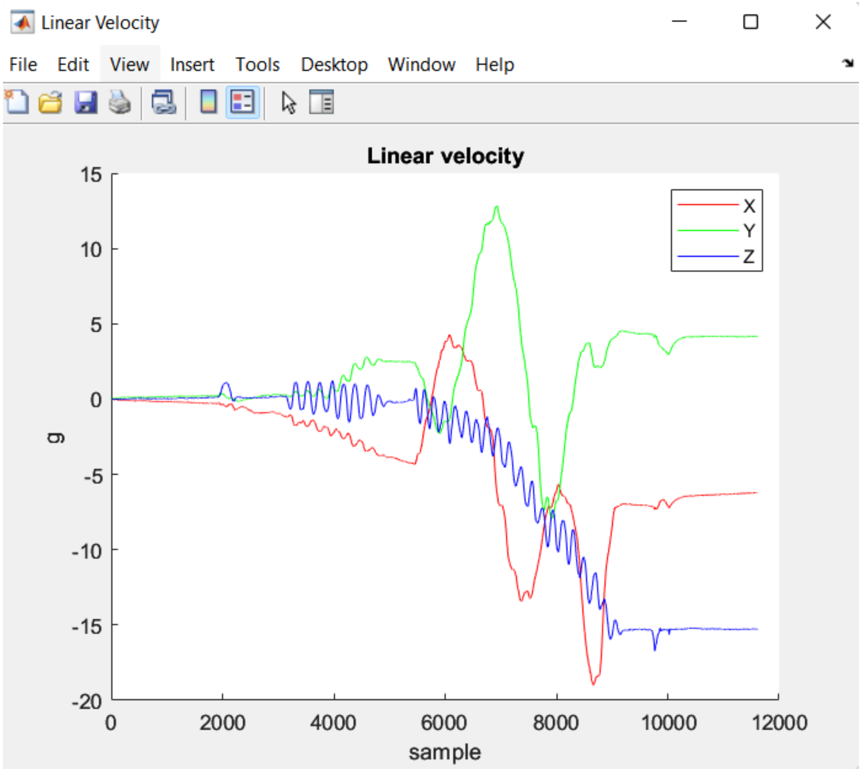


Figure 4.15 Linear Velocity

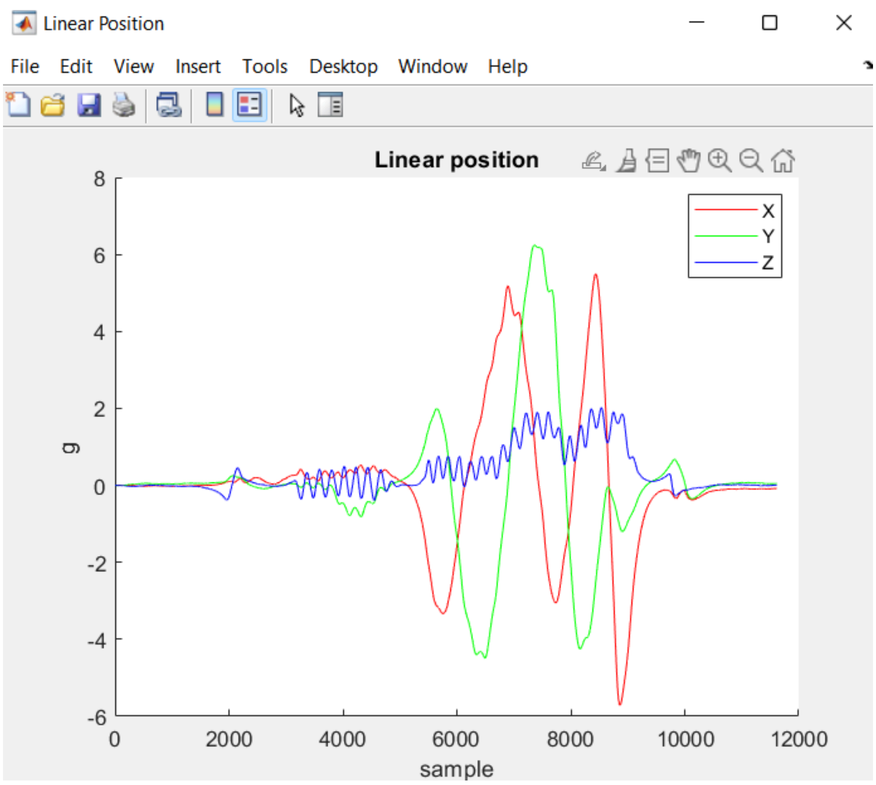


Figure 4.16 Linear position

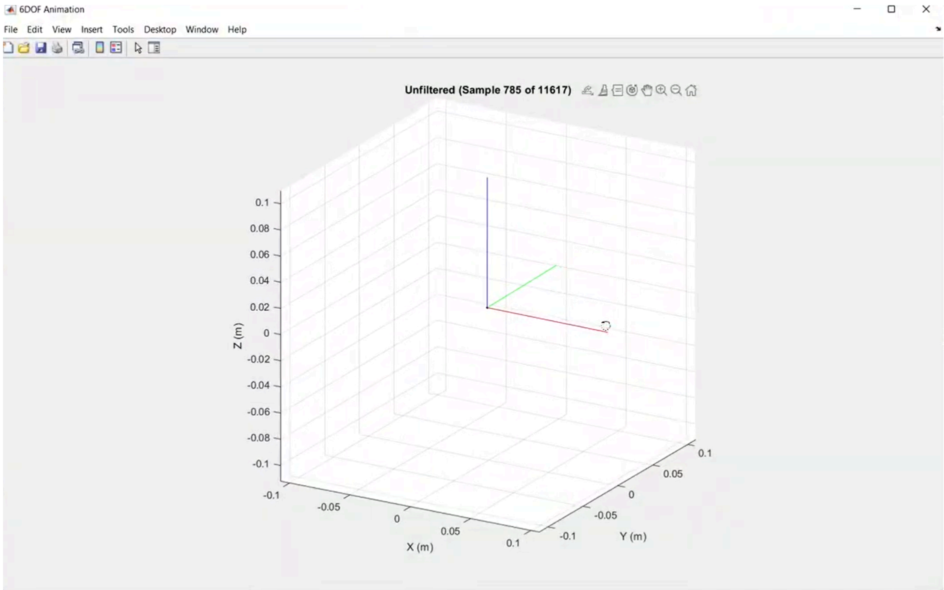


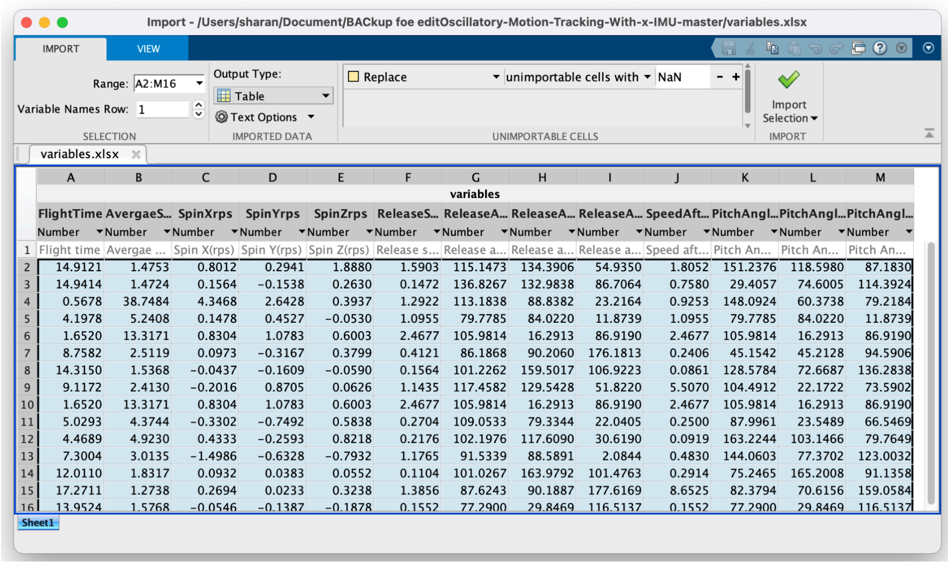
Figure 4.17 Visualization of ball in 3D Space

4.4 PARAMETERS

In sports and beyond, refining throwing technique is crucial for success. This process involves collecting and processing data on throwing motions, calculating key parameters such as release point and pitch angle, and fine-tuning them through trial and error. The resulting dataset is then analyzed to identify trends and areas for improvement. This systematic approach enables athletes and coaches to optimize throwing performance for peak efficiency and effectiveness.

- Processing Data: Initially, raw data from the throw is collected, which could include various metrics like velocity, angle of release, trajectory, etc. This raw data needs refinement and analysis to extract meaningful insights.
- Calculating Parameters: The processed data is then utilized to calculate specific parameters relevant to the throw. This could involve computations to determine factors such as the point of release, pitch, or other performance-related metrics.

- Trial and Error Optimization: To enhance accuracy, the process involves refining the calculations using gain, threshold, and peak values. These values are determined through experimentation and adjustment until they provide the most accurate representation of the throw's parameters.
- Determining Release Point and Pitch: With the aid of the refined parameters, particularly gain, threshold, and peak values, the point of release and pitch can be accurately determined.



	A	B	C	D	E	F	G	H	I	J	K	L	M
	FlightTime	AvergaeS...	SpinXrps	SpinYrps	SpinZrps	ReleaseS...	ReleaseA...	ReleaseA...	ReleaseA...	SpeedAft...	PitchAngl...	PitchAngl...	PitchAngl...
	Number	Number	Number	Number	Number	Number	Number	Number	Number	Number	Number	Number	Number
1	Flight time	Avergae ...	Spin X(rps)	Spin Y(rps)	Spin Z(rps)	Release s...	Release a...	Release a...	Release a...	Speed aft...	Pitch An...	Pitch An...	Pitch An...
2	14.9121	1.4753	0.8012	0.2941	1.8880	1.5903	115.1473	134.3906	54.9350	1.8052	151.2376	118.5980	87.1830
3	14.9414	1.4724	0.1564	−0.1538	0.2630	0.1472	136.8267	132.9838	86.7064	0.7580	29.4057	74.6005	114.3924
4	0.5678	38.7484	4.3468	2.6428	0.3937	1.2922	113.1838	88.8382	23.2164	0.9253	148.0924	60.3738	79.2184
5	4.1978	5.2408	0.1478	0.4527	−0.0530	1.0955	79.7785	84.0220	11.8739	1.0955	79.7785	84.0220	11.8739
6	1.6520	13.3171	0.8304	1.0783	0.6003	2.4677	105.9814	16.2913	86.9190	2.4677	105.9814	16.2913	86.9190
7	8.7582	2.5119	0.0973	−0.3167	0.3799	0.4121	86.1868	90.2060	176.1813	0.2406	45.1542	45.2128	94.5906
8	14.3150	1.5368	−0.0437	−0.1609	−0.0590	0.1564	101.2262	159.5017	106.9223	0.0861	128.5784	72.6687	136.2838
9	9.1172	2.4130	−0.2016	0.8705	0.0626	1.1435	117.4582	129.5428	51.8220	5.5070	104.4912	22.1722	73.5902
10	1.6520	13.3171	0.8304	1.0783	0.6003	2.4677	105.9814	16.2913	86.9190	2.4677	105.9814	16.2913	86.9190
11	5.0293	4.3744	−0.3302	−0.7492	0.5838	0.2704	109.0533	79.3344	22.0405	0.2500	87.9961	23.5489	66.5469
12	4.4689	4.9230	0.4333	−0.2593	0.8218	0.2176	102.1976	117.6090	30.6190	0.0919	163.2244	103.1466	79.7649
13	7.3004	3.0135	−1.4986	−0.6328	−0.7932	1.1765	91.5339	88.5891	2.0844	0.4830	144.0603	77.3702	123.0032
14	12.0110	1.8317	0.0932	0.0383	0.0552	0.1104	101.0267	163.9792	101.4763	0.2914	75.2465	165.2008	91.1358
15	17.2711	1.2738	0.2694	0.0233	0.3238	1.3856	87.6243	90.1887	177.6169	8.6525	82.3794	70.6156	159.0584
16	13.9524	1.5768	−0.0546	−0.1387	−0.1878	0.1552	77.2900	29.8469	116.5137	0.1552	77.2900	29.8469	116.5137

Figure 4.18 Parameters of the ball

4.5 SUMMARY

This chapter includes a brief summary of the process involved in the software implementation of the project which involves coding of the sensor and the microcontroller in order to obtain readings from it. Further coding and data processing which involves cleaning and visualizing the data through various softwares is shown with the respective code and graphs.

CHAPTER 5

DATA ANALYSIS IN SMART CRICKET BALL

5.1 INTRODUCTION

Smart cricket ball plays an important role in a cricket game which is used to improve the performance of the player. Smart cricket balls are equipped with sensors that capture data during a cricket match. Analyzing data from a smart cricket ball involves extracting and interpreting various metrics to gain insights into player performance, ball behavior, and match dynamics.

5.2 DATA ANALYSIS

Data analysis is the process of cleaning, changing, and processing raw data and extracting actionable, relevant information that helps businesses make informed decisions. The procedure helps reduce the risks inherent in decision-making by providing useful insights and statistics, often presented in charts, images, tables, and graphs.

5.2.1 Importance of Data Analysis

- Better Customer Targeting: You don’t want to waste your business’s precious time, resources, and money putting together advertising campaigns targeted at demographic groups that have little to no interest in the goods and services you offer. Data analysis helps you see where you should be focusing your advertising and marketing efforts.
- You Will Know Your Target Customers Better: Data analysis tracks how well your products and campaigns are performing within your target

demographic. Through data analysis, your business can get a better idea of your target audience's spending habits, disposable income, and most likely areas of interest.

- Reduce Operational Costs: Data analysis shows you which areas in your business need more resources and money, and which areas are not producing and thus should be scaled back or eliminated outright.
- Better Problem-Solving Methods: Informed decisions are more likely to be successful decisions. Data provides businesses with information. You can see where this progression is leading. Data analysis helps businesses make the right choices and avoid costly pitfalls.
- You Get More Accurate Data: If you want to make informed decisions, you need data, but there's more to it. The data in question must be accurate. Data analysis helps businesses acquire relevant, accurate information, suitable for developing future marketing strategies, business plans, and realigning the company's vision or mission

5.3 DATA ANALYSIS PROCESS

The process of data analysis, or alternately, data analysis steps, involves gathering all the information, processing it, exploring the data, and using it to find patterns and other insights. The process of data analysis consists of:

- **Data Requirement Gathering:** Ask yourself why you're doing this analysis, what type of data you want to use, and what data you plan to analyze.
- **Data collection**: Guided by your identified requirements, it's time to collect the data from your sources. Sources include case studies, surveys, interviews, questionnaires, direct observation, and focus groups. Make sure to organize the collected data for analysis.
- **Data cleaning** : Not all of the data you collect will be useful, so it's time to clean it up. This process is where you remove white spaces,

duplicate records, and basic errors. Data cleaning is mandatory before sending the information on for analysis.

- **Data analysis:** Here is where you use data analysis software and others tools to help you interpret and understand the data and arrive at conclusions. Data analysis tools include Excel, Python, R, Looker, Rapid miner, Metabase, Redash and Microsoft Power BI.
- **Data visualization**: Data visualization is a fancy way of saying, "graphically show your information in a way that people can read and understand it." You can use charts, graphs, maps, bullet points, or a host of other methods. Visualization helps you derive valuable insights by helping you compare datasets and observe relationships.

5.4. TYPES OF DATA ANALYSIS

There are many types of data analysis are available today, commonly used in technology and business are:

- **Descriptive analysis**: Descriptive analysis involves summarizing and describing the main features of a dataset. It focuses on organizing and presenting the data in a meaningful way, often using measures such as mean, median, mode, and standard deviation. It provides an overview of the data and helps identify patterns or trends.
- **Inferential analysis**: Inferential analysis aims to make inferences or predictions about a larger population based on sample data. It involves applying statistical techniques such as hypothesis testing, confidence intervals, and regression analysis. It helps generalize findings from a sample to a larger population.
- **Exploratory Data analysis**: EDA focuses on exploring and understanding the data without preconceived hypotheses. It involves visualizations, summary statistics, and data profiling techniques to uncover patterns, relationships, and interesting features. It helps to

generate hypotheses for further analysis.

- **Diagnostic analysis**: Diagnostic analysis aims to understand the cause-and-effect relationships within the data. It investigates the factors or variables that contribute to specific outcomes or behaviors. Techniques such as regression analysis, Analysis of Variance, or correlation analysis are commonly used in diagnostic analysis.
- **Predictive analysis:** Predictive analysis involves using historical data to make predictions or forecasts about future outcomes. It utilizes statistical modeling techniques, machine learning algorithms, and time series analysis to identify patterns and build predictive models. It is often used for forecasting sales, predicting customer behavior, or estimating risk.
- **Prescriptive analysis**: Prescriptive analysis goes beyond predictive analysis by recommending actions or decisions based on the predictions. It combines historical data, optimization algorithms, and business rules to provide actionable insights and optimize outcomes. It helps in decision-making and resource allocation.

5.5. DATA ANALYSIS METHODS

Although there are many data analysis methods available, they all fall into one of two primary types: Qualitative and Quantitative Data analysis.

Qualitative Data analysis:

The qualitative data analysis method derives data via words, symbols, pictures, and observations. This method doesn’t use statistics. The most common qualitative methods include:

- Content Analysis, for analyzing behavioral and verbal data.
- Narrative Analysis, for working with data culled from interviews, diaries, surveys.

- Grounded Theory, for developing causal explanations of a given event by studying and extrapolating from one or more past cases.

The key Qualitative methods:

- **Text analysis**: Text analysis, also known in the industry as text mining, works by taking large sets of textual data and arranging them in a way that makes it easier to manage. By working through this cleansing process in stringent detail, you will be able to extract the data that is truly relevant to your organization and use it to develop actionable insights that will propel you forward.
- **Content analysis**: This is a straightforward and very popular method that examines the presence and frequency of certain words, concepts, and subjects in different content formats such as text, image, audio, or video.
- **Narrative analysis**: A bit more complex in nature than the two previous ones, narrative analysis is used to explore the meaning behind the stories that people tell and most importantly, how they tell them. By looking into the words that people use to describe a situation you can extract valuable conclusions.

Quantitative Data analysis:

Also known as statistical data analysis methods collect raw data and process it into numerical data. Quantitative analysis methods include:

- Hypotheses Testing for assessing the truth of a given hypothesis or theory for a data set or demographic.
- Mean, or average, determines a subject’s overall trend by dividing the sum of a list of numbers by the number of items on the list.
- Sample Size Determination uses a small sample taken from a larger group of people and analyzed. The results gained are considered representative of the entire body.

The key Quantitative methods:

- **Cluster analysis**: The action of grouping a set of data elements in a way that said elements are more similar (in a particular sense) to each other than to those in other groups – hence the term 'cluster.' Since there is no target variable when clustering, the method is often used to find hidden patterns in the data. The approach is also used to provide additional context to a trend.
- **Cohort analysis**: This type of data analysis approach uses historical data to examine and compare a determined segment of users' behavior, which can then be grouped with others with similar characteristics. By using this methodology, it's possible to gain a wealth of insight into consumer needs or a firm understanding of a broader target group.
- **Regression analysis:** Regression uses historical data to understand how a dependent variable's value is affected when one (linear regression) or more independent variables (multiple regression) change or stay the same. By understanding each variable's relationship and how it developed in the past, you can anticipate possible outcomes and make better decisions in the future.
- **Factor analysis**: The factor analysis also called "dimension reduction" is a type of data analysis used to describe variability among observed, correlated variables in terms of a potentially lower number of unobserved variables called factors. The aim here is to uncover independent latent variables, an ideal method for streamlining specific segments.
- **Data mining**: A method of data analysis that is the umbrella term for engineering metrics and insights for additional value, direction, and context. By using exploratory statistical evaluation, data mining aims to identify dependencies, relations, patterns, and trends to generate advanced knowledge.
- **Decision Trees**: The decision tree analysis aims to act as a support tool to make smart and strategic decisions. By visually displaying potential

outcomes, consequences, and costs in a tree-like model, researchers and company users can easily evaluate all factors involved and choose the best course of action. Decision trees are helpful to analyze quantitative data and they allow for an improved decision-making process by helping you spot improvement opportunities, reduce costs, and enhance operational efficiency and production.

Apart from qualitative and quantitative categories, there are also other types of data that you should be aware of before dividing into complex data analysis processes. These categories include:

- **Big data:** Refers to massive data sets that need to be analyzed using advanced software to reveal patterns and trends. It is considered to be one of the best analytical assets as it provides larger volumes of data at a faster rate.
- **Metadata:** Putting it simply, metadata is data that provides insights about other data. It summarizes key information about specific data that makes it easier to find and reuse for later purposes.
- **Real time data:** As its name suggests, real time data is presented as soon as it is acquired. From an organizational perspective, this is the most valuable data as it can help you make important decisions based on the latest developments.
- **Machine data:** This is more complex data that is generated solely by a machine such as phones, computers, or even websites and embedded systems, without previous human interaction.

5.6. ROLE OF DATA ANALYSIS IN CRICKET

The role of data analysis in cricket using smart balls is multifaceted, encompassing various aspects of the game, player performance, and strategic decision-making. Smart cricket balls are equipped with sensors that capture data during the game, providing valuable insights for players, coaches, and analysts. Some key roles are:

Bowling Analysis:

- Trajectory and Spin: Smart balls can capture data on the trajectory and spin of the ball. Data analysis helps in understanding how the ball moves through the air, its spin rate, and variations in trajectory.
- Release Point and Speed: Analysis of the release point and speed of the ball provides crucial information about a bowler's performance. Coaches can identify inconsistencies or areas for improvement in a bowler's action.
- Consistency and Accuracy: Data analysis helps assess the consistency and accuracy of a bowler's deliveries, enabling targeted training to enhance precision.

Strategic Decision-Making:

- Opponent Analysis: Smart ball data can be used to analyze the performance of opponents, identifying their strengths, weaknesses, and patterns. This information helps teams formulate game strategies and make informed decisions during matches.
- In-Game Adjustments: Real-time data analysis allows coaches to make in-game adjustments based on the performance of their own team and the opposition. This adaptability can be a significant advantage during a match.

Training Optimization:

- Personalized Training Programs: Individualized player data helps coaches design personalized training programs, focusing on specific areas of improvement for each player.
- Workload Management: Smart ball data assists in managing player workloads, ensuring that training intensity aligns with the demands of the game while minimizing the risk of injuries.

Performance Prediction:

- Predictive Analytics: Utilizing historical data, machine learning algorithms can be applied to predict player and team performance, helping in strategic planning and decision-making.

5.7. EXPLORATORY DATA ANALYSIS

Exploratory data analysis (EDA) is an approach to analyzing data sets to summarize their main characteristics, often employing visual methods. The primary goal is to understand the data's structure, identify patterns, detect anomalies, and formulate hypotheses that can be further investigated. EDA typically involves techniques such as summary statistics, data visualization (e.g., histograms, scatter plots), and data mining techniques. It is an important initial step in the data analysis process, helping analysts gain insights and guide subsequent analyses or modeling efforts. EDA is particularly useful when dealing with unfamiliar or large data sets, as it can provide an overview and guide further investigation.

The context is a table of data related to a series of ball throws or pitches, with various metrics recorded for each throw which is shown in fig 5.1. The table contains following columns:

- Ball No: A unique identifier for each ball throw, starting from 0 and incrementing by 1 for each subsequent throw.
- Flight time (s): The duration (in seconds) from the release of the ball to when it is pitched or thrown.
- Average speed (m/s): The average speed of the ball during its flight, measured in meters per second.
- Spin X (rps): The rotational speed (in revolutions per second) of the ball around the X-axis (typically corresponding to the left-right axis).
- Release speed (Rs): The initial speed of the ball at the moment it is released, measured in the same units as the average speed.

- Speed after Pitch (Rp): The speed of the ball immediately after it has been pitched or thrown, measured in the same units as the average speed.

	Ball No.	Flight time(s)	Avergae speed(m\s)	Spin X(rps)	Release speed(Rs)	Speed after Pitch(Rp)
0	1	14.912088	1.475313	0.801191	1.590267	1.805218
1	2	14.941392	1.472420	0.156379	0.147248	0.757985
2	3	4.468864	4.922951	0.433282	0.217610	0.091950
3	4	4.197802	5.240838	0.147782	1.095549	1.095549
4	5	1.652015	13.317073	0.830387	2.467704	2.467704
5	6	8.758242	2.511920	0.097282	0.412122	0.240570
6	7	14.315018	1.536847	-0.043660	0.156436	0.086134
7	8	9.117216	2.413017	-0.201633	1.143451	5.507025
8	9	1.652015	13.317073	0.830387	2.467704	2.467704
9	10	5.029304	4.374363	-0.330194	0.270416	0.250049
10	11	4.468864	4.922951	0.433282	0.217610	0.091950
11	12	7.300366	3.013547	-1.498620	1.176500	0.482951
12	13	12.010989	1.831656	0.093244	0.110389	0.291427
13	14	17.271062	1.273807	0.269414	1.385631	8.652466
14	15	13.952381	1.576792	-0.054611	0.155214	0.155214
15	16	1.058608	20.782007	0.433282	0.207425	0.091950
16	17	0.567766	38.748387	4.346757	1.292221	0.925295

Figure 5.1 Parameters

Following data import, parameters are assigned specific weightage for analysis. The resulting data distribution is visualized using a violin plot, as seen in Fig 5.2. This plot offers a detailed representation of parameter distribution, aiding in the identification of patterns and anomalies. Leveraging the violin plot enhances data interpretability and facilitates deeper insights. This visualization technique enables users to make informed decisions based on the nuanced understanding of the dataset.

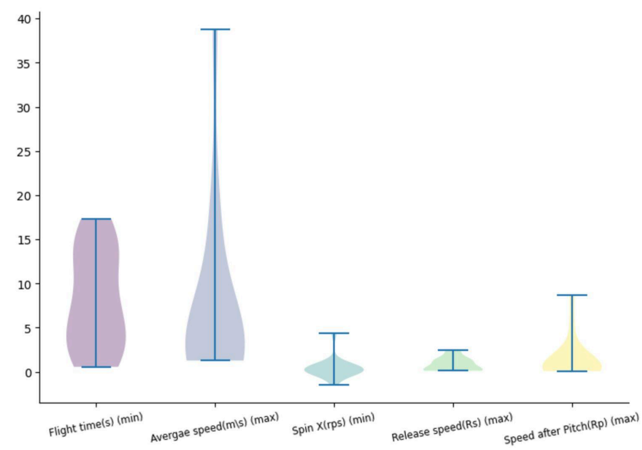


Figure 5.2 Violin plot

A heat map is a 2-dimensional data visualization technique that represents the magnitude of individual values within a dataset as a color. Once normalization of data is done then a heat map form of representation is displayed for those five parameters shown in fig 5.3.

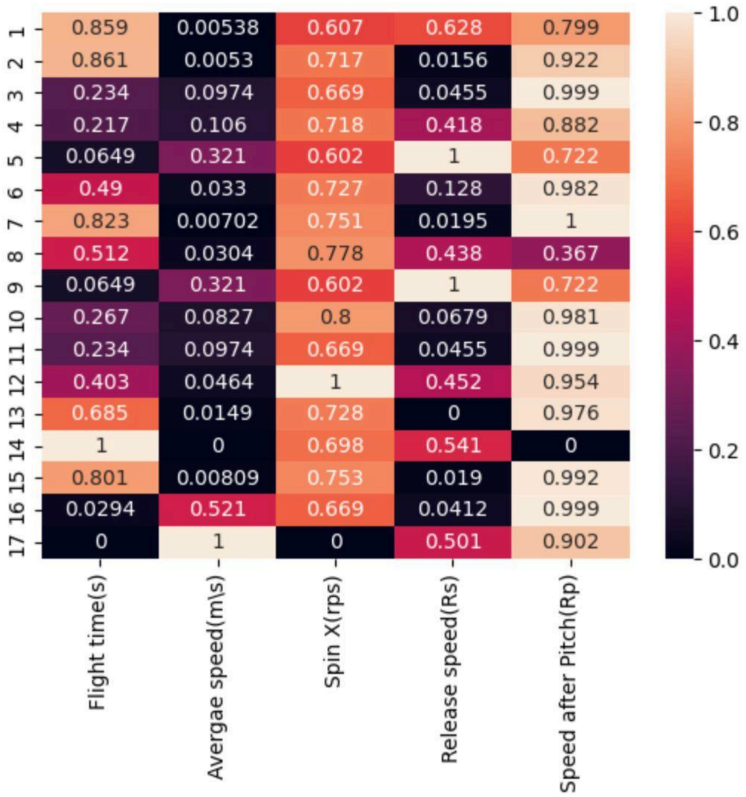


Figure 5.3 Heat map of the parameters

With the help of normalization of the data, using Minimisation and Maximisation norm, the data has been ranked according to the parameters input. Additionally, based on the weightage of each parameter given, the ranking algorithm produces an output table ranked accordingly as shown in the fig 5.4.

WeightedSum (mnorm=sum, wnorm=sum) - Solution:

ALT./CRIT.	Flight time(s) (min)	Avergae speed(m\s) (max)	Spin X(rps) (min)	Release speed(Rs) (max)	Speed after Pitch(Rp) (max)	Rank
1	14.9121	1.47531	0.801191	1.59027	1.80522	12
2	14.9414	1.47242	0.156379	0.147248	0.757985	8
3	4.46886	4.92295	0.433282	0.21761	0.0919499	9
4	4.1978	5.24084	0.147782	1.09555	1.09555	7
5	1.65201	13.3171	0.830387	2.4677	2.4677	3
6	8.75824	2.51192	0.0972819	0.412122	0.24057	5
7	14.315	1.53685	-0.0436596	0.156436	0.0861339	17
8	9.11722	2.41302	-0.201633	1.14345	5.50702	15
9	1.65201	13.3171	0.830387	2.4677	2.4677	4
10	5.0293	4.37436	-0.330194	0.270416	0.250049	14
11	4.46886	4.92295	0.433282	0.21761	0.0919499	10
12	7.30037	3.01355	-1.49862	1.1765	0.482951	13
13	12.011	1.83166	0.093244	0.110389	0.291427	6
14	17.2711	1.27381	0.269414	1.38563	8.65247	11
15	13.9524	1.57679	-0.0546111	0.155214	0.155214	16
16	1.05861	20.782	0.433282	0.207425	0.0919499	2
17	0.567766	38.7484	4.34676	1.29222	0.925295	1

Figure 5.4 Ranking of Data

5.8. FLOW DIAGRAM

The context provided by the flow diagram in the fig 5.5 outlines the general steps involved in building a machine learning model for classification tasks. Here is a brief explanation of each step:

- **Collect Dataset**: This involves gathering data relevant to the problem you are trying to solve. The data could be in the form of structured or unstructured data, such as text, images, or numerical data.
- **Pre-processing**: This step involves cleaning and transforming the raw data into a format that can be used for machine learning. This may include handling missing values, removing outliers, and normalizing the data.
- **Feature Extraction:** This step involves identifying and extracting the most relevant features from the data that will be used to train the machine learning model. Feature extraction can help reduce the dimensionality of the data and improve the accuracy of the model.
- **Convert Categorical Data into Numerical Data:** Many machine learning algorithms require numerical data as input. Therefore, categorical data needs to be converted into numerical data before it can

be used for training the model. This can be done using techniques such as one-hot encoding or label encoding.

- **Partition Dataset into Training and Test Data:** This step involves splitting the dataset into two parts: a training set and a test set. The training set is used to train the machine learning model, while the test set is used to evaluate the performance of the model.
- **Training:** This step involves using the training set to train the machine learning model. The model learns to identify patterns in the data and make predictions based on those patterns.
- **Knowledge Base:** This refers to the information and insights that the machine learning model has learned from the training data.
- **Classification:** This is the process of using the trained machine learning model to make predictions on new, unseen data. In the context of classification tasks, the model will predict which category or class a new data point belongs to.
- **Results:** This refers to the output of the machine learning model, which includes the predictions made by the model and any metrics used to evaluate the performance of the model.

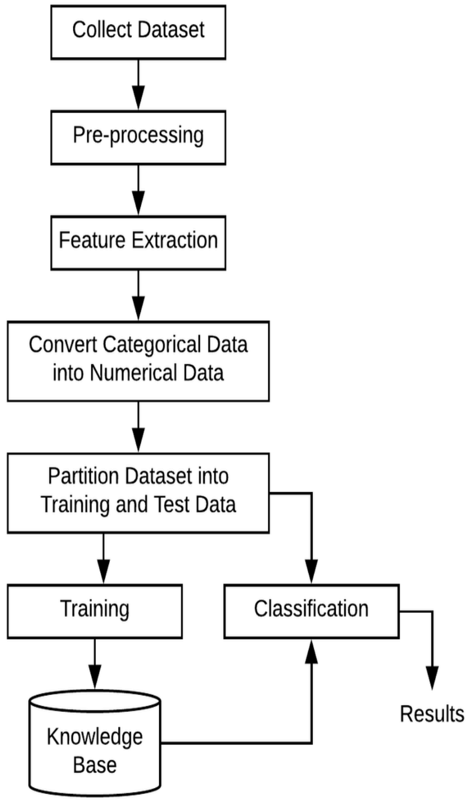


Figure 5.5 Flow diagram

5.9. SUMMARY

This chapter talks about data analysis, importance of data analysis, various types of data analysis, methods and techniques which can be used to analyze the data. It also briefs about the role of data analysis in cricket with the help of smart ball and use of exploratory data analysis method for detailed analysis of the collected data.

CHAPTER 6

WEB APPLICATION

6.1 INTRODUCTION

A created web application empowers users to control a microcontroller and monitor received data effortlessly. Leveraging websockets, data transmission occurs via a locally established web server, accessible across any connected Wi-Fi network through standard web browsers on devices like phones, laptops, or tablets. Within the webpage's backend, gyroscope and accelerometer readings, alongside operational flags, are displayed in a dynamic console log. This data, formatted as JSON strings, is decoded by the receiving unit for further analysis, facilitating seamless communication between transmitting and receiving units. Additionally, the application stores previous data locally, enabling comparison with current results to yield valuable insights.

6.2 WEB PAGE

The webpages are run on a localhost machine, which is connected to the ESP8266 Web-server. With the help of flask and ajax, we send the webpages and update them live. The Index page in figure 6.1 includes the following features:

- *Start New Log* - This feature allows users to start a new log.
- *Last Recorded Log* - This feature displays the last recorded log.
- *View Raw Data* - This feature allows users to view raw data.

- *Visual Report* - This feature generates a visual report based on the data.
- *Analyze Data* - This feature analyzes and provides insights.

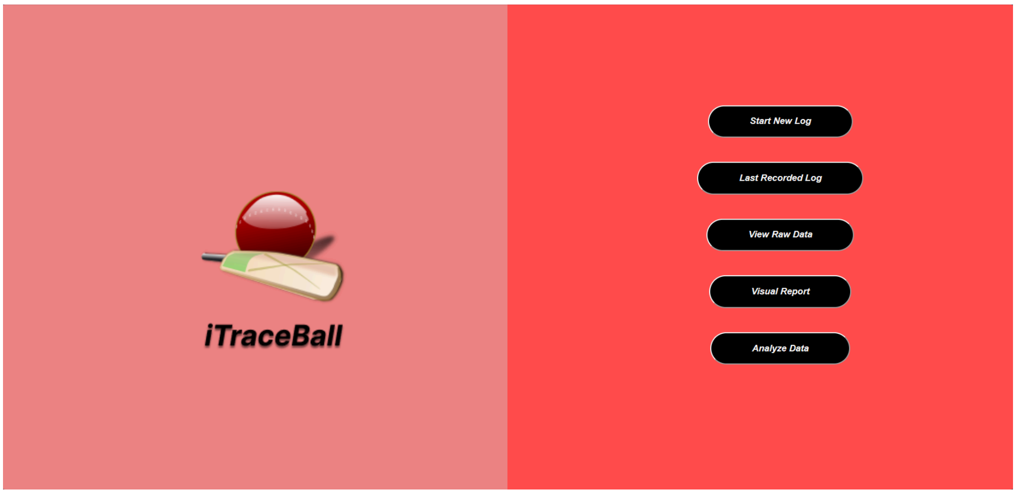


Figure 6.1 Index Page

When the "Start New Log" button is clicked, a pop-up message appears saying "Your data has started recording successfully." This message suggests that the web page has a function to record user data, possibly for logging or analytics purposes. Furthermore, when the "Calibrate" button is clicked, it initiates a calibration process for the data and gives a pop-up message indicating data being calibrated. After calibration, the "Record Data" function is likely to start recording data according to the calibration settings shown in fig 6.2.

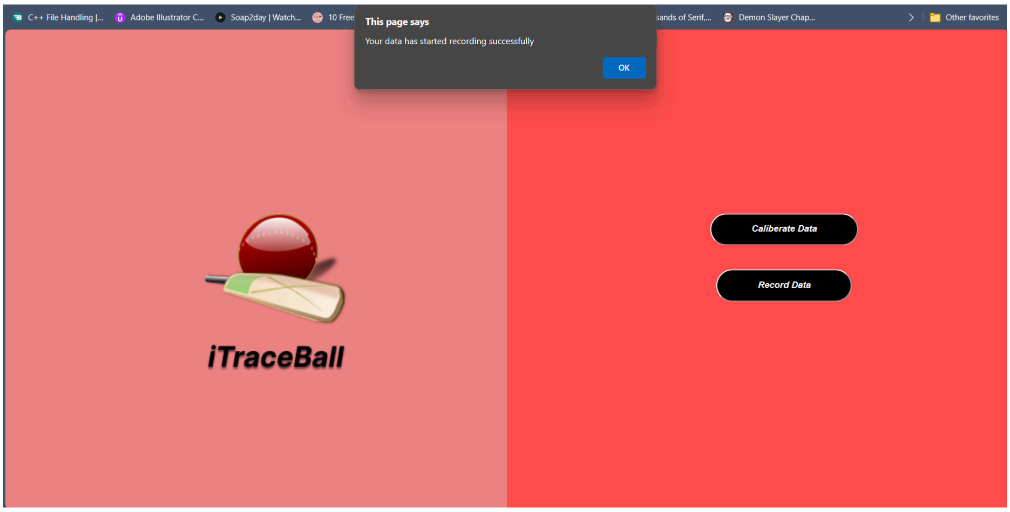


Figure 6.2 Start New Log Page

The integration of the "Record Data" and "View Raw Data" buttons streamlines the user experience by providing easy access to collected sensor data from both the index page and a dedicated view page. In Fig 6.3a, the presence of the File Chooser button enhances user control, allowing them to manage selected files efficiently. Despite the absence of a selected file currently, users can readily manipulate their data viewing experience. The detailed insights provided by the Raw Data View, encompassing magnetometer, gyroscope, and accelerometer readings, empower users to gain a comprehensive understanding of the ball sensor's behavior. With the packet number counter continuously updating, users can track the influx of data packets in real-time, facilitating accurate data analysis and interpretation. The inclusion of a Clear Data button further enhances user autonomy by providing a convenient means to reset the view, ensuring a clutter-free interface for uninterrupted data exploration. This webpage epitomizes user-centric design, prioritizing seamless data visualization and management for optimal user engagement and satisfaction.

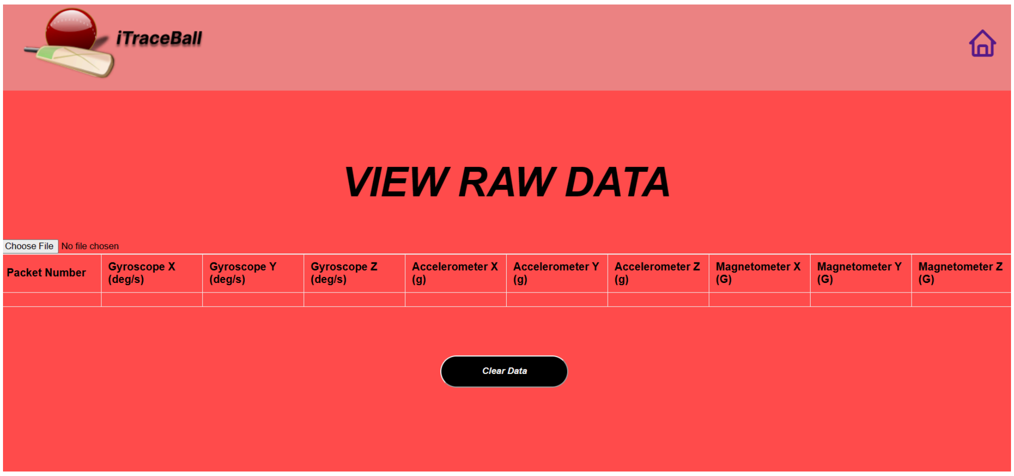


Figure 6.3a View Raw data Page

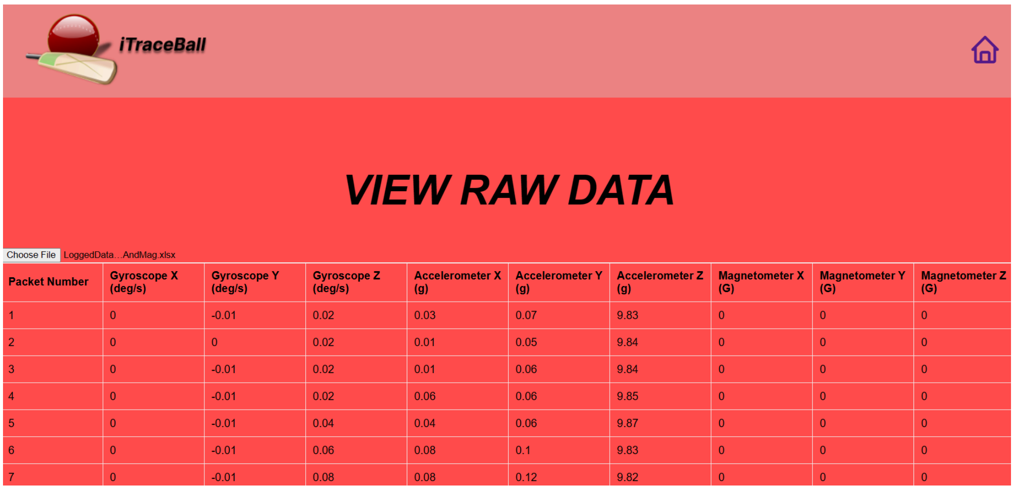


Figure 6.3b View raw data page with data

Next is the Last Recorded Log page displaying the latest recorded event from the sensor is shown in fig 6.4. The recorded data includes Average Velocity which is the average speed of the ball, measured in meters per second (m/s), Revolutions is the number of rotations the ball made per second, measured in revolutions per second (rps), Total Time which tells the duration of the event in seconds (sec), and Distance that is traveled by the ball during the event, measured in yards.

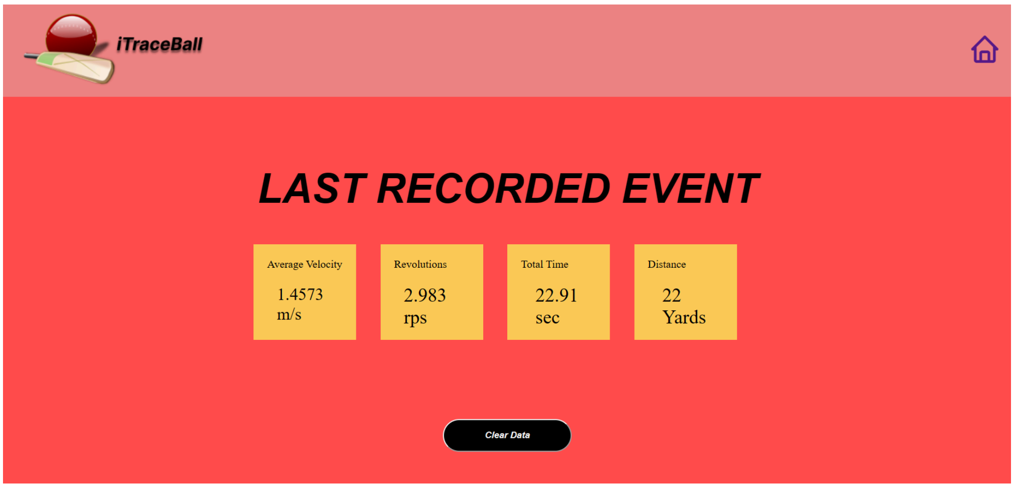


Figure 6.4 Last Recorded Log Page

The Visual Report page shown in fig 6.5 displays data visualization of the collected data from the sensor. It shows correlation between parameters that are obtained. The page contains 6-7 slides of visualization.

Figure 6.5 Visual Report Page

In our exploratory data analysis we are considerings five parameters that are flight time, average speed, spin, release speed and speed after pitch. These parameters can either be minimized or maximized in order to obtain the right throw style. This is usually decided by the coach. Additionally, each parameter has a weightage on how much minimisation and maximization is required. These parameters are collected through the Analyze data page shown in fig 6.6 and once entered, it sends the data to a python script to analyze the throws according to your style.

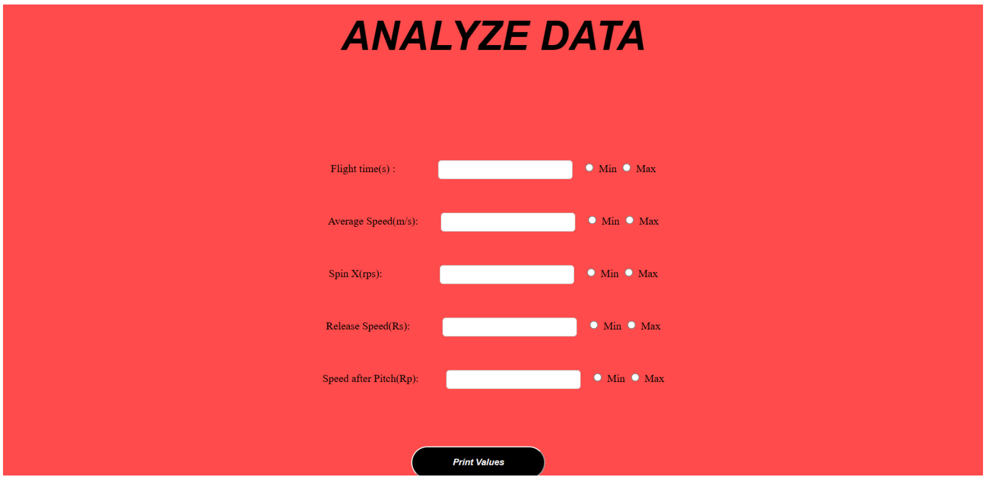


Figure 6.6 Analyze data Page

6.3 SUMMARY

This chapter includes a brief summary about the webpage features a "Start New Log" button for data recording, confirmed by a pop-up message. Calibration is initiated by clicking "Calibrate," with a corresponding ongoing calibration message. "Record Data" and "View Raw Data" buttons manage sensor data, focusing on magnetometer, gyroscope, and accelerometer readings. The "Last Recorded Log" page displays key parameters like Average Velocity and Distance traveled. This analysis considers parameters like flight time and spin for throw optimization, sent to a Python script for further analysis based on user preferences.

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

During the course of this project, simulations and functioning of hardware components were performed. Initially, with the ESP8266 which is a wifi module used to receive and transfer data. Then the MPU-6050 sensor was studied and used to collect data from the ball. MPU-6050 and ESP8266 are connected in order to capture and transmit real-time data on crucial performance metrics such as ball speed, spin, trajectory, and impact points and upload it into a table to analyze for performance improvisation of the player. This live feedback loop ensures that players receive instantaneous insights into their performance during practice sessions and matches.

The collected data is then subjected to in-depth analysis, generating comprehensive reports and statistics that aid players and coaches in identifying strengths, weaknesses, and areas for improvement.The innovative features of the IoT-based cricket ball extend beyond traditional training methods. It facilitates interactive training sessions, offering players a dynamic and engaging learning experience. Coaches can tailor training programs based on individual player analytics, fostering targeted skill development. This project represents a significant leap forward in cricket training technology, empowering players with the tools they need to refine their skills, make data-driven decisions, and ultimately elevate their performance on the field.

7.2 FUTURE SCOPE

In the next phase of the project, software implementation of smart balls will be performed after which analyzed data will be used for performance improvisation of the player. The integration of smart technology in cricket, including the use of a smart ball, opens up several exciting possibilities for the future. Here are some potential future scopes for smart balls in cricket:

- Performance Analytics - Smart balls can provide detailed performance analytics for bowlers. Data on speed, spin, swing, and seam movement can be collected and analyzed in real-time, offering valuable insights into a bowler's performance.
- Umpire Assistance - Smart balls could assist umpires in making more accurate decisions. Sensors in the ball could detect edges, no-balls, and other crucial events, reducing the chances of human error.
- Player Development - Coaches can use smart ball data to analyze players' strengths and weaknesses. This information can be used to tailor training programs, correct technical flaws, and enhance overall performance.
- Ball Condition - Smart balls could be equipped with sensors to monitor the health and condition of the ball. This could include data on wear and tear, allowing for timely replacement to maintain fair play.
- Integration with Wearables - Players could use wearable devices that communicate with smart balls to gather comprehensive data about their performance during practice sessions and matches. This can contribute to a more holistic understanding of player fitness and skill development

The future scope for smart balls in cricket is vast and holds the potential to revolutionize the game in terms of player development, fan engagement, and overall cricketing experience.

7.3 SOCIETAL IMPACT

Smart ball technology enhances the fan experience by providing real-time data and insights. This increased engagement can lead to a broader and more passionate fan base, fostering a sense of community among cricket enthusiasts. The use of smart ball technology provides an educational platform for cricket enthusiasts, students, and aspiring players. The detailed analytics and data generated can be used for educational purposes, allowing individuals to learn more about the technical aspects of the game.

The integration of advanced technology in cricket introduces fans and players to new and sophisticated tools. This can contribute to an increased level of technological literacy within the cricketing community and beyond. The development and implementation of smart ball technology can inspire innovation in other sports and industries. The success of such technological advancements in cricket may encourage the exploration of similar applications in different domains. The growth of smart ball technology creates new career opportunities in fields such as sports technology, data analytics, and sports science. This could attract individuals with diverse skill sets to contribute to the development and maintenance of such technologies. Smart ball technology can revolutionize the way players are trained and developed. This not only benefits professional players but also has a trickle-down effect on grassroots cricket, improving coaching methods and player development at all levels.

As smart ball technology becomes more prevalent, its accessibility may increase. This democratization of technology could potentially benefit players and teams with limited resources, allowing them to access advanced training and performance analysis tools. The societal impact of smart ball technology in cricket goes beyond the game itself, influencing education, innovation, career opportunities, and the overall viewing experience. As with any technological advancement, careful consideration of the ethical and social implications is crucial to ensure a positive and responsible integration into the fabric of society.

REFERENCES

1. Doljin B and Fuss F.K(2015),‘Development of a smart cricket ball for advanced performance analysis of bowling’ ,Procedia Technology 20, 133–137

2. Doljin B, Jeong K, Kim Y.K and Fuss F.K(2020),‘Profiling of a pitcher's performance with a smart baseball: A case report.’ Proceedings,49, 103.

3. Espinosa, H.G., Öchsner, A., Mirnajafizadeh, A., James, D.A.(2016),‘On the preliminary design of an instrumented ball for sports applications’, Hum. Technol. 1, 5–6.

4. Espinosa, H.G., Shepherd, J.B., Thiel, D.V., Worsey,(2019),‘M.T.O. Anytime, anywhere! Inertial sensors monitor sports performance’,IEEE Potentials , 38, 11–16

5. Espinosa, H.G., Lee, J., James, D.A.(2015), ‘The inertial sensor: A base platform for wider adoption in sports science applications’, J. Fitness Res.4, 1–8.

6. Fuss F.K, Lythgo, N., Smith, R., Benson, A., Gordon, B.(2011). ‘Identification of key performance parameters during off-spin bowling with a smart cricket ball’,Sports Technology 4, 159-163.

7. Fuss F.K,Smith R.M and Subic, A.(2012),‘Determination of spin rate and axes with an instrumented cricket ball’,Procedia Eng. 34, 128–133

8. Fuss F.K, Smith R.M, (2014),‘Accuracy Performance Parameters of Seam Bowling, Measured with a Smart Cricket Ball’, Procedia Engineering 72, 435-440.

9. Franz Konstantin Fuss, Batdelger Doljin and René E. D. Ferdinands(2020),‘Bowling Performance Assessed with a Smart Cricket Ball: A Novel Way of Profiling Bowlers’,Proceedings, 49(1), 141

10. Pravin O. Balbudhe, Dr. Brijesh Khandelwal, Dr. Sachin Solanki(2022), ‘Automated Training Techniques and ElectronicsSensors Role in Cricket: A Review’, Journal of Physics : Conference series 2286(1)

11. Rowlands, D., James, D.A., Thiel, D.V.(2009),‘Bowler analysis in cricket using center of mass inertial monitoring’,Sports Technol. 2, 39–42.

12. Yehuda Weizman, Adin Ming Tan and Franz Konstantin Fuss(2020). ‘Measurement of Flight Dynamics of a Frisbee Using a Triaxial MEMS Gyroscope’,Proceedings .