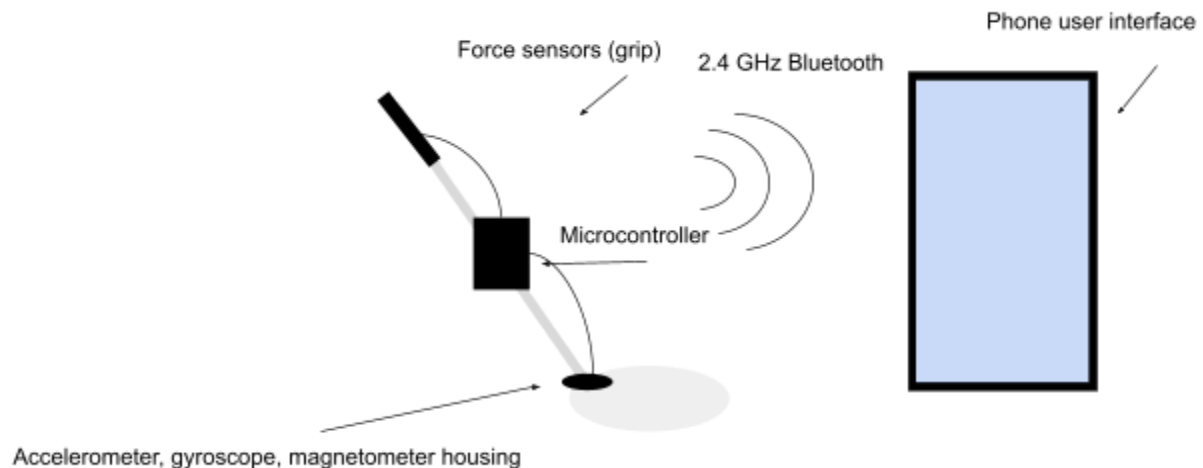# Real-Time Golf Swing Tracker

Team Members:
- tamirb2
- leuba2
- kijungk3

Problem

Mastering the golf swing is a complex challenge with nuances that can be difficult to grasp without precise feedback. Current training methods often rely on professional coaching and visual observation, which might not be readily accessible or affordable for all golfers. Additionally, the subtle mechanics of a golf swing, including swing path, speed, and force, are not easily quantifiable through mere observation. There's a growing need for a more accessible and scientific approach to golf training that leverages modern technology to provide real-time, detailed feedback directly to the golfer.

Solution

We propose to develop the Real-Time Golf Swing Tracker equipped with an integrated sensor system and a companion mobile application to analyze and improve golf swings. The core of our solution involves embedding accelerometers, gyroscopes, and force sensors within the grip of a standard golf club. These sensors will capture critical data points such as swing speed, angle, and grip pressure during each stroke. This data is then processed by a microcontroller that filters and interprets the raw sensor outputs. The processed information is wirelessly transmitted to a mobile application that provides the golfer with immediate visual feedback and historical data analysis.

High-Level Requirements:

- All sensors integrated into the Smart Golf Club, including accelerometers, gyroscopes, and force sensors, must maintain a tolerance of accuracy within 20%. This level of precision is essential to ensure that the data collected is consistently reliable and reflects true performance.
- The user interface of the Smart Golf Club's accompanying mobile application is designed to display data in real-time, ensuring that all information is updated and presented to the user within 5 seconds following each golf swing. This prompt update allows golfers to immediately see the impact of their swings on parameters such as swing speed, angle, and force, facilitating on-the-spot adjustments and learning.
- The total weight of the Smart Golf Club, after integrating all hardware systems including sensors, microcontroller, and power supply, should not exceed a 5% increase over the weight of a standard golf club. This stringent weight threshold is essential to ensure that the club retains a natural feel and balance, allowing golfers to swing with their usual technique without adaptation to added bulk. Maintaining this minimal weight increase is crucial for user acceptance and performance, ensuring that the technological enhancements enhance rather than hinder the golfing experience.

2. Block Diagram (TODO: INVERT THIS COLOR; use snipping tool)
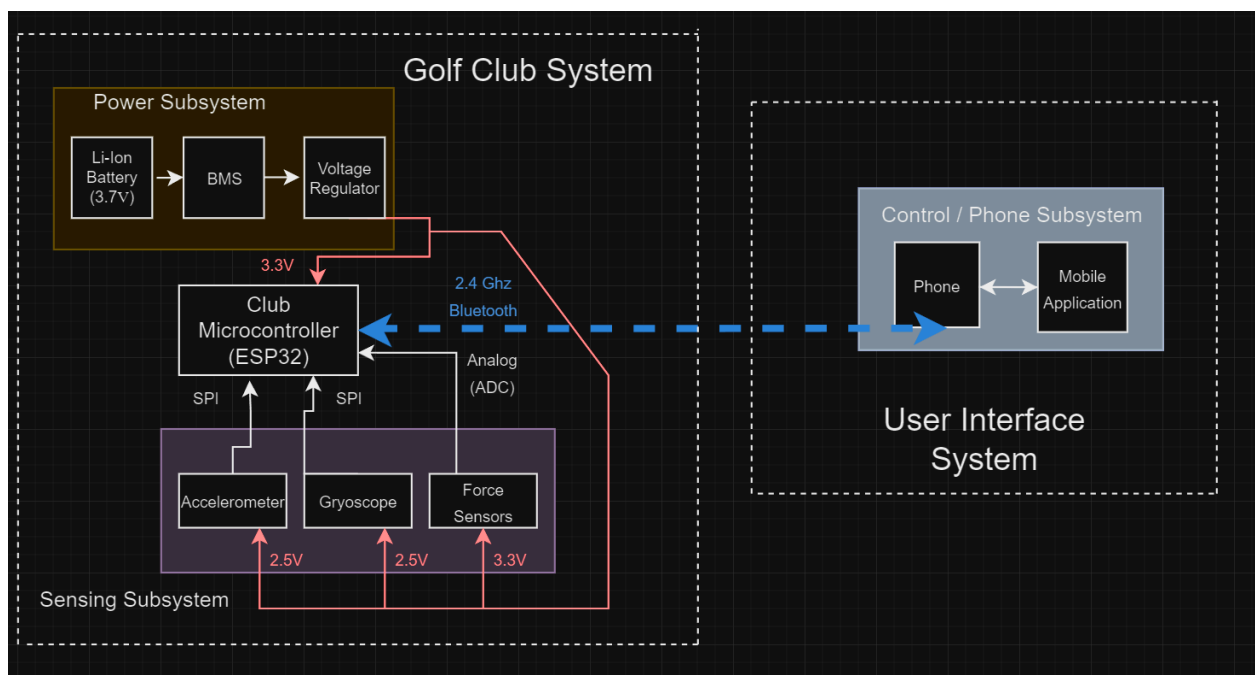
Subsystem overview

Sensor Subsystem

This subsystem includes accelerometers, gyroscopes, and force sensors integrated into the golf club's grip. These sensors capture real-time data on swing speed, angle, and grip pressure. We plan to use MPU9250, which includes the gyroscope and the accelerometer together. This sensor will be attached right above the golf head, allowing for accurate sensing when the club swings up and down. Also, the FSR06BE sensors will be utilized to sense the grip force from our hands to the golf grip. We would insert sensors under the grip such that the pressure resulting from our hands will be transmitted to the microcontroller, which will be on the golf shaft.

–**EDIT: Force sensors can be potentially changed to** ->
https://www.tekscan.com/sites/default/files/FLX-Datasheet-A201-RevI.pdf

- Accelerometer : Measures the acceleration and deceleration of the golf club to track swing speed.
- Gyroscope : Tracks the orientation and angular velocity to track the angle of the club throughout the swing.
-Magnetometer : Tracks absolute orientation in relation to Earth's magnetic field to work in tandem with the gyroscope and gather a more accurate reading.
- Force Sensor : Monitors the grip pressure applied by the golfer throughout the swing. If the scope of the project is too small, we also plan to add force sensors on the golf club face to measure the point of contact made between the club and the ball.

Microcontroller Subsystem

The microcontroller subsystem processes data from the sensors, executes filtering algorithms, and manages wireless data transmission to the mobile application.

- Microcontroller (ESP32-S3-WROOM-1 MCU): Manages real-time data processing from all sensors and supports Bluetooth communication to transmit swing data to the mobile application. Sensors will be connected to the microcontroller through the GPIO and ADC allowing for digital and analog transmission. The microcontroller/PCB will most likely be placed & screwed on the shaft of the club to allow for even distribution of wires between grip sensors and club sensors.

Power Subsystem

This subsystem ensures that all electronic components within the golf club are adequately powered during use.

- Rechargeable Cable/Battery (3.3 - 5V) : A lightweight, durable battery capable of providing consistent power to necessary subsystems for extended periods, ensuring usability through multiple rounds of golf. A USB port will be utilized to allow for the battery to be recharged.

Mobile Application and Data Analysis Subsystem

A comprehensive app that receives data from the golf club's microcontroller. The user interface displays real-time analytics and historical trend analysis to help golfers understand and improve their swing techniques. High-level/process-intensive code will be run from the mobile application to perform any algorithms or potential ML to analyze golf swings. The application will most likely be exclusively hosted as an Android application for easier development.

Criterion For Success

- **Precision**: The sensor data must be accurate to within a few degrees or percentage points, ensuring that feedback is reliable.
- **User Interface**: The mobile application must be intuitive and easy to use, providing clear and actionable insights without overwhelming the user. A section will be dedicated to user data & numerics so that users can quickly digest raw data.
- **Durability**: The Real-Time Golf Swing Tracker must withstand regular use in various weather conditions without sensor or system failure.

Subsystem Requirements

Power Subsystem

The Power Subsystem is responsible for supplying consistent and reliable power to all electronic components within the Smart Golf Club, including the microcontroller and various sensors. It primarily consists of a rechargeable lithium-ion battery and a voltage regulator to ensure a steady output.

Interfaces:
Outputs a regulated 3.3V to the Microcontroller and Sensing Subsystem.
Receives charging input via a USB-C port

Requirements:
Must supply at least 500mA continuously at 3.3V ± 0.1V to the ESP32 microcontroller. According to the link below (use APA citation), the voltage is 3.3 Volts being driven at .5 amps

- Citation:
  https://www.espressif.com/sites/default/files/documentation/esp32-s3-wroom-1_wroom-1u_datasheet_en.pdf

Must integrate a charging circuit capable of fully charging the battery within 2 hours.
The battery should last for at least 10 holes of golf on a single charge without performance degradation.

Sensing Subsystem

The Sensing Subsystem includes accelerometers, gyroscopes, and force sensors, each responsible for collecting specific data about the golf swing. The sensors will be placed along the grip to capture user data such as grip strength, swing speed, and club angle. Sensor placement may be extended by placements on the golf head to track the location the golf club hits the golf ball.

Interfaces:
Communicates with the Microcontroller via I2C or SPI

Requirements:
Sensors must maintain a tolerance of accuracy within 7% to ensure reliable data
Must function effectively under the environmental conditions typically experienced on a golf course
Accelerometers and gyroscope must update their readings at least 1000 times per second to capture dynamic swing data accurately.

Microcontroller Subsystem

The microcontroller serves as the central processing unit of the Smart Golf Club, handling input from the sensors via SPI, processing the data, and transmitting it to the Mobile Application via Wifi or Bluetooth. Placement of microcontroller will most likely be in the middle of the golf club to reduce distance wires have to travel (in the scenario we have sensors both on the grip and the club head)

Interfaces:
Receives sensor data via I2C/SPI.
Transmits processed data to the Mobile Application via Bluetooth.

Requirements:

Must have a high enough sampling rate to capture critical points during the downswing and backswing during the users golf swing. According to:
https://blastconnect.com/training_center/golf/metrics/golf-full-swing#gref
, pro golfers can have a backswing time of 0.6 seconds and a downswing time around ⅓ of their backswing. In total we expect golfers to take roughly 1 second for their swing. Considering the short amount of time for the entire swing, we want to have a sampling rate of roughly 1kHz so we can get enough data to ensure we capture all critical points with accuracy (top speed, angles at peak of backswing etc).

(TODO, ReEvaluate these requirements)
Must have sufficient processing power to handle data from multiple sensors without lag, ensuring data processing latency of less than 500 milliseconds.
Requires embedded Bluetooth capability with a range of at least 10 meters to maintain connection with the mobile application.

Phone Subsystem

The Mobile Application interfaces with the Smart Golf Club to display data in a user-friendly format and stores historical data for progress tracking. This subsystem may be extended by incorporating camera features to record golfers and track their form/technique with pre-compiled data. This may involve machine learning.

Interfaces:
Receives data from the Microcontroller via Bluetooth

–EDITED FOR DESIGN DOC–
Requirements:
Must update and display data within 45 seconds of receiving it to ensure real-time feedback. (TODO, place the +- 15 second deviation here)
ESP32 BLE has an expected transmission rate of 400-700 kbps. Considering the microcontroller will need to transmit roughly 40 bytes. Utilizing worst case estimates, we can get an expected transmission time of: 40/(400/8)^3 which is roughly 0.00032 seconds or .32 ms. However, packet sizes for BLE protocol can get up to 250 bytes with the header, leading to a transmission time of 0.002 seconds or 2 ms. However, we expect that the transmission of data can be slower than this expected value, and that the time for the mobile application to perform the calculations can grow large with more complex tasks such as machine learning. Therefore, we want to have the desired time for the data/analysis to be displayed on the frontend UI.
-Calculation:
Force sensor -> 4 bytes
Accelerometer->3 axes * 4 = 12 bytes

Magnetometer->3 axes * 4 = 12 bytes
Gyroscope->3 axes * 4 = 12 bytes

Tolerance Analysis
(TODO, explain why 7%, maybe bring it up to 10)
Sensors
The force pressure sensor has a part-to-part repeatability of around 4% and 2% uncertainty of force readings from a single component. Given that, the uneven pressure given by a golf grip, and the variance of data through the microcontroller being received over 2.4GHz RF, we hope to expect a 7% accuracy tolerance of the force sensors.

On the MPU-9250, we can expect a range of 16g for the accelerometer, 2000 dps for the gyroscope, and 4800µT for the magnetometer. Given our application, we can hope to expect a 5%tolerance.Specsheet:
https://invensense.tdk.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf
Needs some math to explain why this tolerance

Wireless communication and user interface

We want at least 1 Mbps, a latency of 20-30 ms, and a less than 1% packet loss from our bluetooth transmission and our android app to crash less than once per hour of use.

Power

We would aim for our li-ion and voltage regulator to be within 3% range of 3.3V and 10% range for necessary current draw.

Alternatives

Current training aids mostly involve static tools like swing trainers or mats that do not provide dynamic, real-time feedback. While some digital solutions exist, such as swing analyzers that attach to a club, they often require additional devices or do not integrate seamlessly. Our solution

improves upon these by integrating all necessary technologies directly into the club and accompanying app, providing a more holistic and user-friendly experience.

https://courses.grainger.illinois.edu/ece445/guidelines/proposal.asp

_____

Requirements for first TA meeting:

1. Block diagram:
2. Three high-level requirements:
    a. There must be a 20% tolerance of accuracy for all sensors (CHATGPT these to make it longer)
    b. User interface should display data in real-time, data should be updated within 5 seconds of each golf swing
    c. Maintain a 5% threshold of weight for the golf club after hardware systems are attached
3. One subsystem requirement:
    a. Sensors must maintain a tolerance of accuracy within 20% to ensure reliable data
    b. Must function effectively under the environmental conditions typically experienced on a golf course
4. Tolerance analysis
    a. Sensors
        i. The force pressure sensor has a part-to-part repeatability of around 4% and 2% uncertainty of force readings from a single component. Given that, the uneven pressure given by a golf grip, and the variance of data through the microcontroller being received over 2.4GHz RF, we hope to expect a 20% accuracy tolerance of the force sensors.
        ii. On the MPU-9250, we can expect a range of 16g for the accelerometer, 2000 dps for the gyroscope, and 4800μT for the magnetometer. Given our application, we can hope to expect a 5% tolerance.
    b. Wireless communication and user interface
        i. We want at least 1 Mbps, a latency of 20-30 ms, and a less than 1% packet loss from our bluetooth transmission and our android app to crash less than once per hour of use.
    c. Power
        i. We would aim for our li-ion and voltage regulator to be within 3% range of 3.3V and 10% range for necessary current draw.
5. Rough Draft (maybe reuse RFA but tweak a bit)
6. Git repo for notebook (someone verify) https://gitlab.engr.illinois.edu/ece445/example-project
7. Proof of completion

Bullet point high-level requirements

Instead of soldering, try baking the accelerometer (too small)

Label any arrows

Identify pin arrangement (UART …)

Fa 22 first project : good example

Phone subsystem can be just user interface subsystem

Might need 5V to actually utilize voltage regulator

Tolerance analysis → voltage regulator, temperature control

09/23
Use wires on the golf club and if it doesn't work, we need to inform the professor/ta that the weight of the wires were too much to be added on to the club

Build the chip on the pcb, cause three won't be much on the pcb itself

Need to make a force sensor, since it will be mostly analog (resistive)

Start making schematic on kiCAD this week → takes a long time ordering + receiving, so start as early as possible
        Try having the schematic on the design document (professor would appreciate it)

Build accelerometer on pcb

Esp32 → programming circuit → most likely need to replicate it (on website)
  ● linear voltage regulator, esp32, esp32 programming circuit, force sensor, accelerometer on pcb
  ●
Two microcontrollers may be hard (transceiver + receiver)

ECE 445

Senior Design Laboratory
Project Proposal

Real-Time Golf Swing Tracker

Team No. 15

Tamir Battsogt

(tamirb2@illinois.edu)

Ryan Leuba

(leuba2@illinois.edu)

Ki Jung (Ben) Kim

(kijungk3@illinois.edu)

TA: Sanjana Pingali

Professor: Arne Fliflet

September 19, 2024

Contents

1 Introduction

1.1 Problem

Mastering the golf swing is a complex challenge with nuances that can be difficult to grasp without precise feedback. Current training methods often rely on professional coaching and visual observation, which might not be readily accessible or affordable for all golfers. Additionally, the subtle mechanics of a golf swing, including swing path, speed, and force, are not easily quantifiable through mere observation. There's a growing need for a more accessible and scientific approach to golf training that leverages modern technology to provide real-time, detailed feedback directly to the golfer.

1.2 Solution

We propose to develop the Real-Time Golf Swing Tracker equipped with an integrated sensor system and a companion mobile application to analyze and improve golf swings. The core of our solution involves embedding accelerometers, gyroscopes, and force sensors within the grip of a standard golf club. These sensors will capture critical data points such as swing speed, angle, and grip pressure during each stroke. This data is then processed by a microcontroller that filters and interprets the raw sensor outputs. The processed information is wirelessly transmitted to a mobile application that provides the golfer with immediate visual feedback and historical data analysis.

1.3 Visual Aid

We provide a visual aid of our Real-Time Golf Swing Tracker in Figure 1. Data gathered from the swing will be transferred into the mobile app through bluetooth.
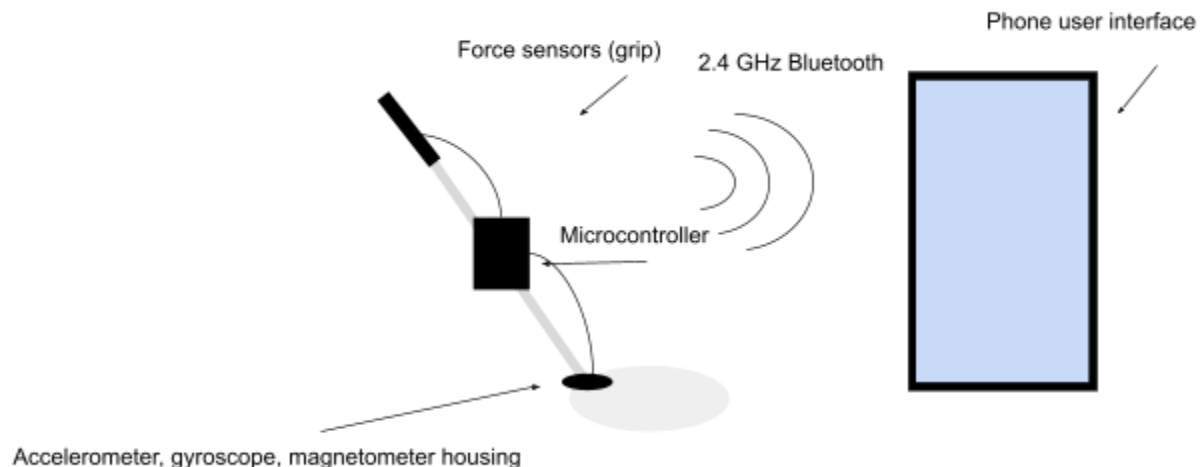
Figure 1: Visual Overview of Real-Time Golf Swing Tracker

1.4 High Level Requirements

a. All sensors integrated into the Smart Golf Club, including accelerometers, gyroscopes, and force sensors, must maintain a tolerance of accuracy within 20%. This level of precision is essential to ensure that the data collected is consistently reliable and reflects true performance.

b. The user interface of the Smart Golf Club's accompanying mobile application is designed to display data in real-time, ensuring that all information is updated and presented to the user within 5 seconds following each golf swing. This prompt update allows golfers to immediately see the impact of their swings on parameters such as swing speed, angle, and force, facilitating on-the-spot adjustments and learning.

c. The total weight of the Smart Golf Club, after integrating all hardware systems including sensors, microcontroller, and power supply, should not exceed a 5% increase over the weight of a standard golf club. This stringent weight threshold is essential to ensure that the club retains a natural feel and balance, allowing golfers to swing with their usual technique without adaptation to added bulk.

2   Design

2.1   Block Design
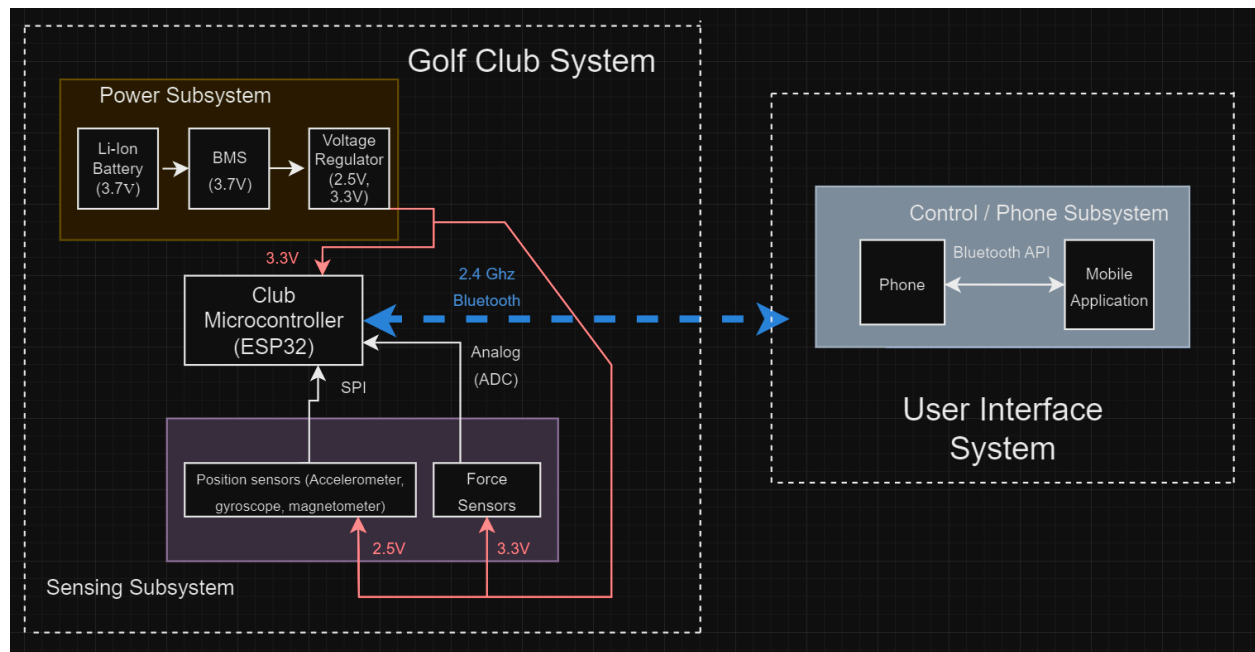
2.2   Subsystem Overview

Our project is divided into a total of four subsystems as outlined below.

2.2.1   Power Subsystem

This subsystem ensures that all electronic components within the golf club are adequately powered during use. We will be using a lightweight, durable battery capable of providing consistent power to necessary subsystems for extended periods, ensuring usability through multiple rounds of golf. This battery cell will be replaceable.

2.2.2   Sensor Subsystem

This subsystem includes accelerometers, gyroscopes, and force sensors integrated into the golf club's grip. These sensors capture real-time data on swing speed, angle, and grip pressure. They will be attached right above the golf head, allowing for accurate sensing when the club swings up and down. Also, force sensors will be utilized to sense the grip force from our hands to the golf grip. We would insert sensors under the grip such that the pressure resulting from our hands will be transmitted to the microcontroller.

2.2.3   Microcontroller Subsystem

The microcontroller subsystem processes data from the sensors, executes filtering algorithms, and manages wireless data transmission to the mobile application. The ESP32 microcontroller will manage real-time data processing from all sensors while supporting Bluetooth communication for swing data transmission to the mobile application. Sensors will be connected to the microcontroller through the GPIO and ADC allowing for digital and analog transmission.

2.2.4 Mobile Application and Data Analysis Subsystem

A comprehensive app that receives data from the golf club's microcontroller. The user interface displays real-time analytics and historical trend analysis to help golfers understand and improve their swing techniques. High-level/process-intensive code will be run from the mobile application to perform any algorithms or potential ML to analyze golf swings. The application will most likely be exclusively hosted as an Android application for easier development.

2.3 Subsystem Requirements

2.2.1 Power Subsystem

a. The subsystem must supply at least 500mA, continuously at 3.3V ± 0.1V.
b. The subsystem must integrate a charging circuit capable of fully charging the battery within 2 hours.
c. The battery should last for at least 10 holes of golf on a single charge without performance degradation.

2.2.2 Sensor Subsystem

a. All sensors must maintain a tolerance of accuracy defined in tolerance analysis (2.3) to ensure reliable data.
b. This subsystem must function effectively under the environmental conditions typically experienced on a golf course (rain, strong wind).
c. Accelerometers and gyroscopes must update their readings at least 1000 times per second to capture dynamic swing data accurately.

2.2.3 Microcontroller Subsystem

a. The microcontroller subsystem must have sufficient processing power to handle data from multiple sensors without lag, ensuring data processing latency of less than 500 milliseconds.

b. This subsystem requires embedded Bluetooth capability with a range of at least 10 meters to maintain connection with the mobile application.

2.2.4 Mobile Application and Data Analysis Subsystem

a. This subsystem must update and display data within 5 seconds of receiving it to ensure real-time feedback.
b. It requires robust error handling to manage potential data transmission errors over Bluetooth.
c. Our subsystem should be compatible with major mobile operating systems (iOS and Android) to ensure wide accessibility.

2.3 Tolerance Analysis

a. Sensors
   i. The force pressure sensor has a part-to-part repeatability of around 4% and 2% uncertainty of force readings from a single component. Given that, the uneven pressure given by a golf grip, and the variance of data through the microcontroller being received over 2.4GHz RF, we hope to expect a 10% accuracy tolerance of the force sensors.
   ii. On the MPU-9250, we can expect a range of 16g for the accelerometer, 2000 dps for the gyroscope, and 4800μT for the magnetometer. Given our application, we can hope to expect a 5% tolerance.
b. Wireless communication and user interface
   i. We want at least 100 Kbps, a latency of 20-30 ms, and a less than 1% packet loss from our bluetooth transmission and our android app to crash less than once per hour of use.
c. Power
   i. We would aim for our li-ion and voltage regulator to be within 3% range of 3.3V and 10% range for current draw across all subsystems.

There are two main systems that require analysis in our project, the wireless communication latency with the amount of data we are processing, and ensuring we sample our sensors at an adequate rate to reach the information processing desired while also not putting too large of a strain on the ESP32's small digital memory.

The MPU9250 has 3 axes on each of its sensors; accelerometer, gyroscope, and magnetometer. We are given that BLE4.2 has a throughput of 700 kb/s, the 9 sensors on our IMU operate at 32 kHz, and the ESP32 GPIO pins can reach a theoretical 240 MHz rate which is plenty to reach the desired data rate. The force sensors are resistive so it is through the ADC pin, which is 6 μs on the ESP32. Given that we can find the frequency using

$$fmax_{ADC} = \frac{1}{6*10^{-6}} = 166.67 kHz$$

Which is also well above the desired range of frequency we want to send to the web-app. With all this in mind and given the fact that each data point will be 4 bytes as a float, the hardware involved is viable to collect data fast enough to reach our specifications.

3 Ethics and Safety

Regarding ethics, our group will follow the IEEE Code of Ethics:

1. **"To Uphold the highest standards of integrity, responsible behavior, and ethical conduct in professional activities"** [1]
   In order for this code to be effective, our group will maintain a strict schedule of weekly meetings to review and discuss accomplishments and improvements. This way, we will be able to continue to keep a high standard of professionalism.

2. **"To treat all persons fairly and with respect, to not engage in harassment or discrimination, and to avoid injuring others"** [1]
   To ensure that no one gets harmed throughout the semester, we created a group chat in our phone / discord so that no one gets left behind. Also, when only two people can make it to meetings, we have created a Github repository to keep track of what was done to store information for the next meetings. Moreover, since we are handling a golf club that may lead to dangerous situations, we need to ensure enough distance between members when the swinging happens and that no one is swinging alone, but with a team member present in the same room for secondary assistance.

4   References

[1] IEEE. ""IEEE Code of Ethics"." (2016), [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html (visited on 09/19/2024).