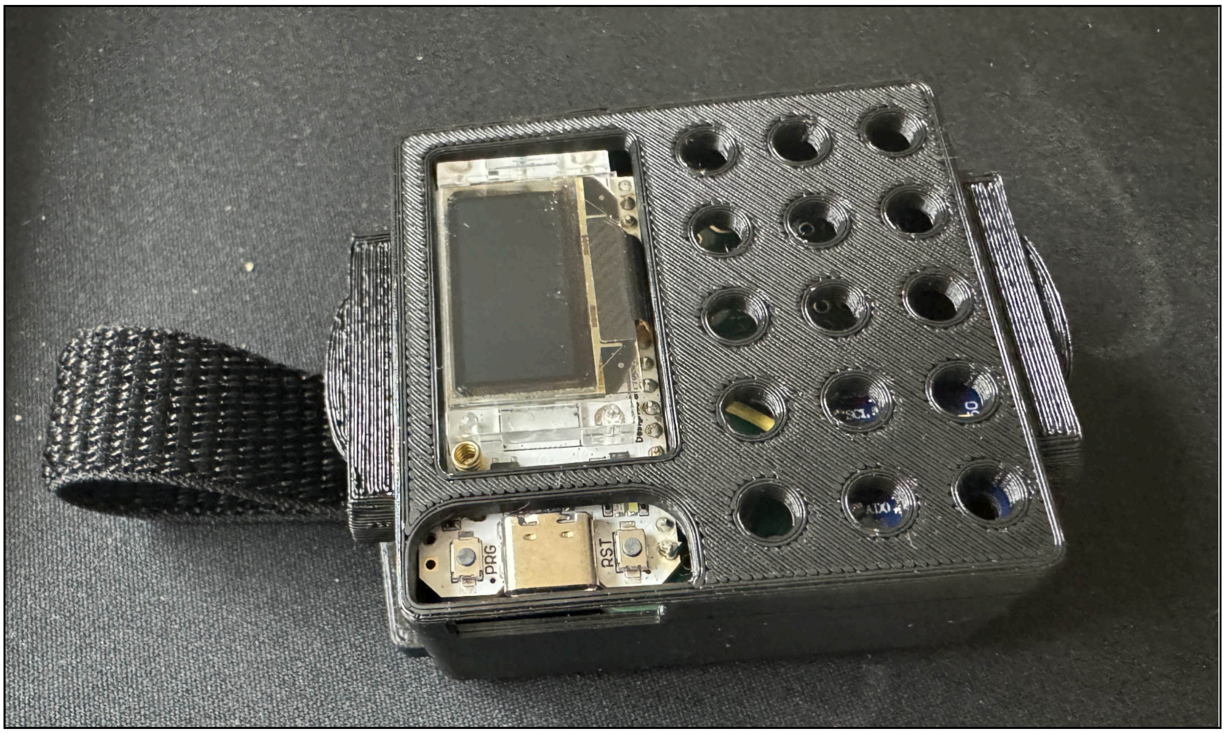
OnCourt: Wearable Joint Motion Tracker for Tennis Training

Enabling Accessible Skill Improvement Through Motion Analysis and Wearable Tech



Max Lee

Master of Engineering in Mechanical Engineering
University of California, Berkeley

ME292C: Product Design and Development

May 2025

Need Identification:

I identified several problems in my daily life, but one stood out: many teen athletes lack access to professional training due to limited connections or resources, which prevents them from staying competitive in their sport. To address this issue, our team developed a wearable product that mounts on the body to track wrist and joint motion. This data can then be compared to that of professional athletes, highlighting differences and suggesting improvements, enabling users to train effectively without a physical personal trainer.

Process of Product Identification:

I began by reviewing existing patents to see whether my idea already existed in the market. During this process, I found that many basketball personal training tools were similar to my concept. As a result, I shifted focus to tennis, a sport with fewer training tools and less market saturation.

Next, I interviewed several potential users (e.g., tennis and other sports players) to identify their key needs. Here are the top five:

1. The product must maintain natural hitting motions during extended training sessions.
2. It should accurately track joint angles and body positioning relative to professional form benchmarks.
3. It must be affordably priced for amateur players without access to professional coaching.
4. It should provide immediate, real-time feedback on swing mechanics.
5. It should measure serve and groundstroke speed to help users monitor milestones and set performance goals.

Based on these needs, I brainstormed 10 different product concepts. Each design was evaluated using criteria such as motion tracking accuracy, ease of use, comfort, feedback quality, durability, and manufacturing cost. After scoring and comparing the designs with my team, we selected the top five for further development.

Reflection on Concept Selection

I found that brainstorming everyday frustrations (or “bugs”) is a great way to generate product ideas. Products exist to solve problems, so identifying what bugs people is a strong starting point.

Interviewing potential customers was also highly valuable. It helped me avoid personal bias and identify real needs, allowing us to develop a product with broader appeal.

Working in a team facilitated idea generation. While many ideas were similar, unique contributions and different perspectives helped refine the final product.

Process of Product Design:

After identifying the top design, I researched available devices and compiled a bill of materials (BOM) to ensure timely ordering and team approval.

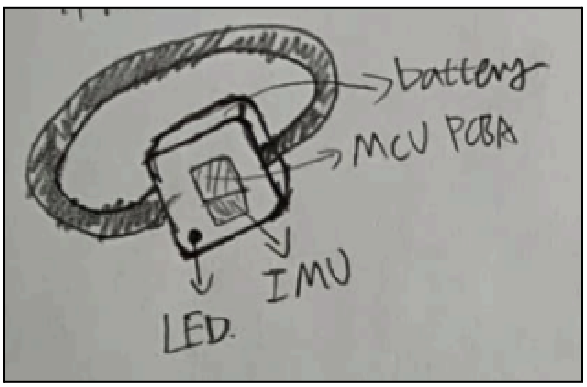


Figure: Wearable Watch-like device Sketch (Top Prototype)

Bill of Materials (BOM):

Items	Count	Cost
Mini PCB Prototype Board	1	$7.49
Breadboard and Jumper Wire kit	560 Pcs	$12.97
MPU-6050	3 Pcs	$10.99
MakerFocus ESP32 LoRa V3	1	$19.99
1100mAh Li Battery	2	$15.99
	Total Cost	$67.43

I created a CAD prototype in SolidWorks using dimensions from the manufacturers and my expectations for soldering the electronics.

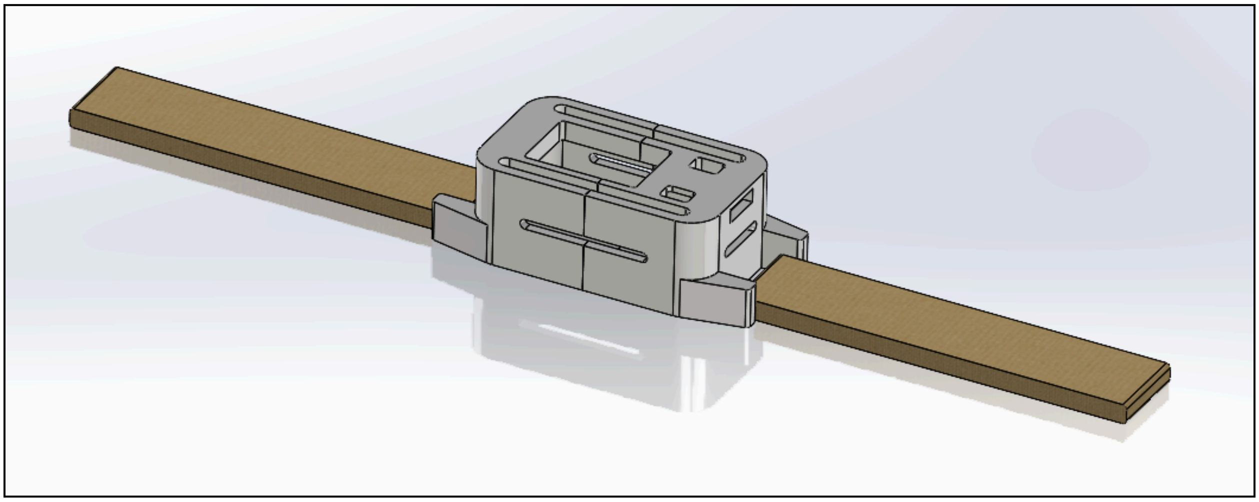


Figure: Wearable Watch-like device CAD (Top Prototype)

Reflection:

I realized that, due to my limited experience with soldering and electronics, my watch case design was not feasible. The configuration of the breadboard and MPU6050, as well as the screen orientation, needed revisions. For example, the screen should be rotated 90 degrees to allow users to read it naturally.

Proof of Concept:

I researched how to use OpenPose for joint tracking (https://blog.roboflow.com/what-is-openpose/) and tested it on a YouTube video featuring Roger Federer. I wrote code to track his motion using OpenPose.

Figure: Demonstration of OpenPose Applied to a Video (https://www.youtube.com/watch?v=stEhSvoou4g)

Reflection:

Joint tracking worked reasonably well, but some joints were inaccurately tracked. As a result, the output graph was sometimes flawed. This is something to consider as we refine the design.

MPU6050 and ESP32 Setup

I researched Arduino libraries for both components. After wiring the MPU6050 to the ESP32 using I2C (connecting SCL and SDA to the appropriate GPIO pins), I was able to test each individually.

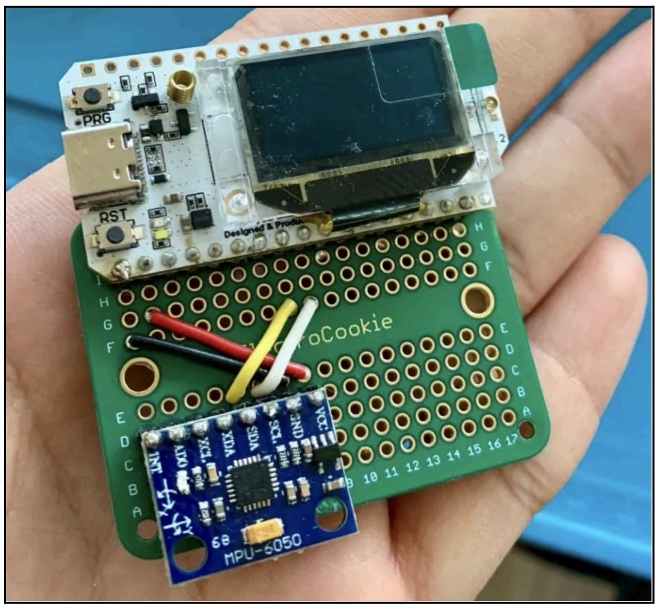


Figure: ESP32 microcontroller and MPU6050 sensor module connected via electronic wiring on a perfboard

Reflection:

To use the ESP32 effectively, it's essential to understand its datasheet. Each GPIO pin has specific functions, and understanding them was crucial for setting up the MPU6050 properly.

Testing IMU Accuracy

I tried generating acceleration outputs and applied double integration to estimate position. While stationary, the results were accurate. However, during motion, sensor noise caused significant error accumulation, resulting in inaccurate trajectories.

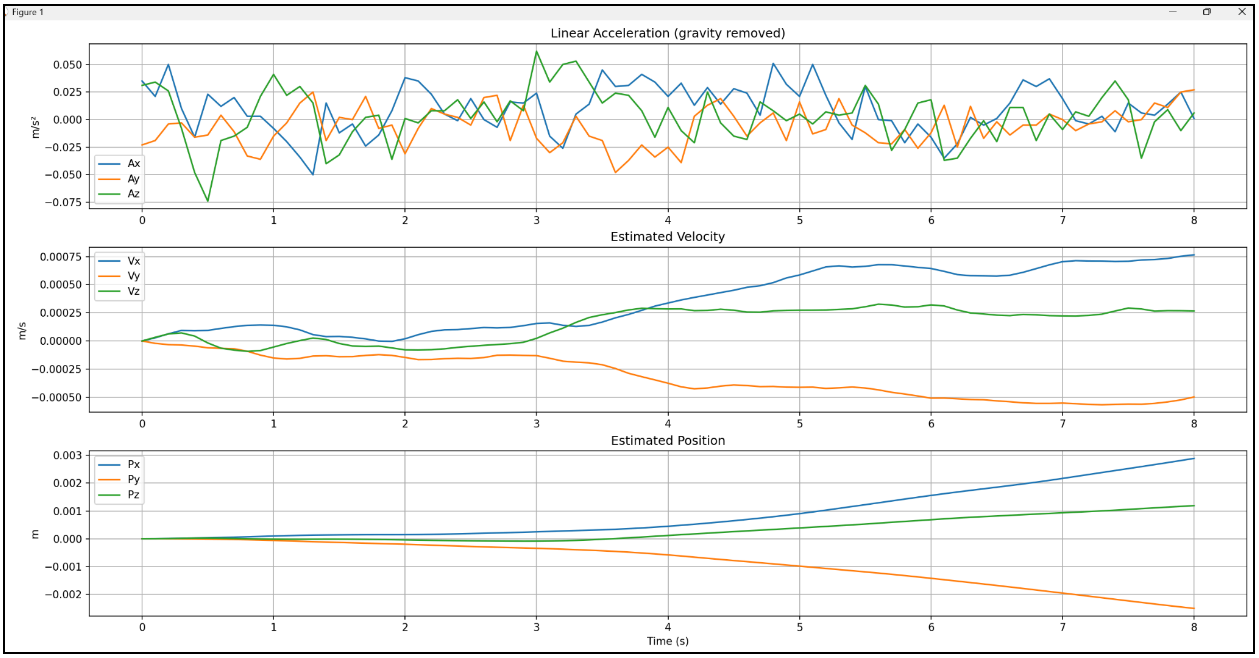


Figure: Acceleration, velocity, and position plots while the watch remains stationary

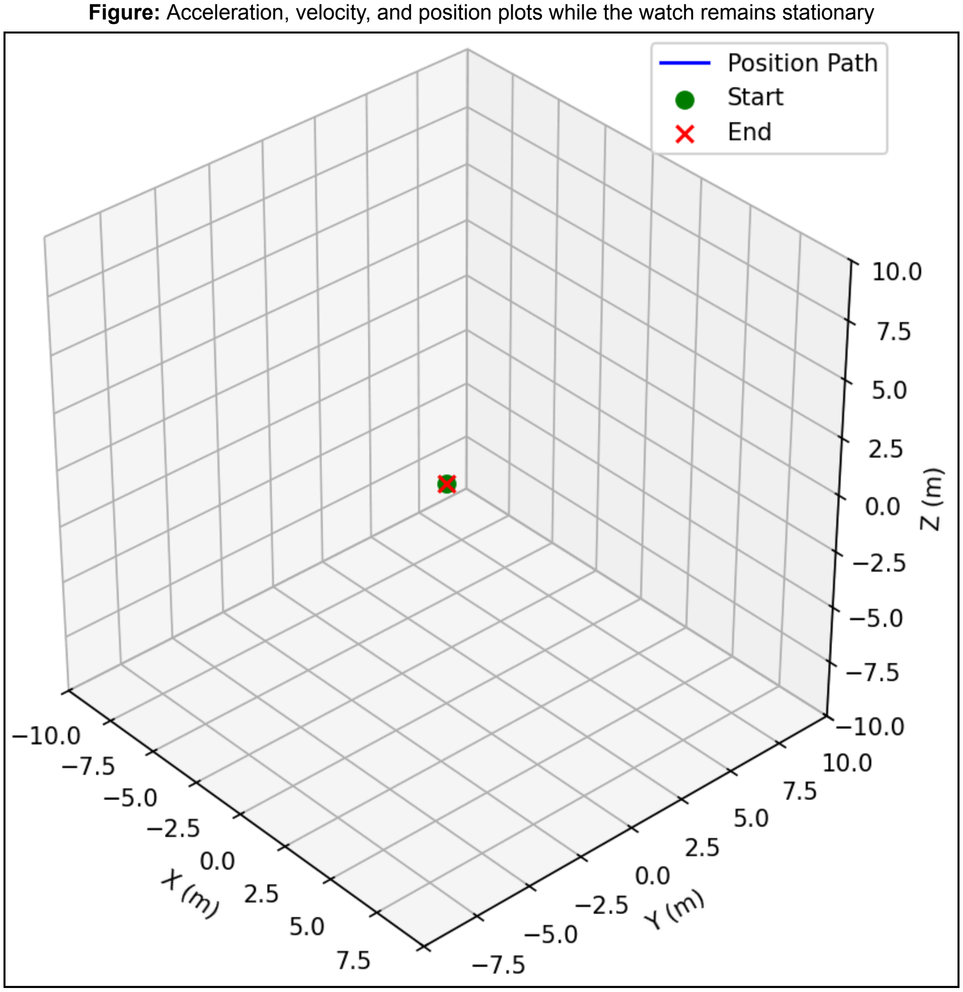


Figure: 3D plot of the watch's estimated trajectory while stationary

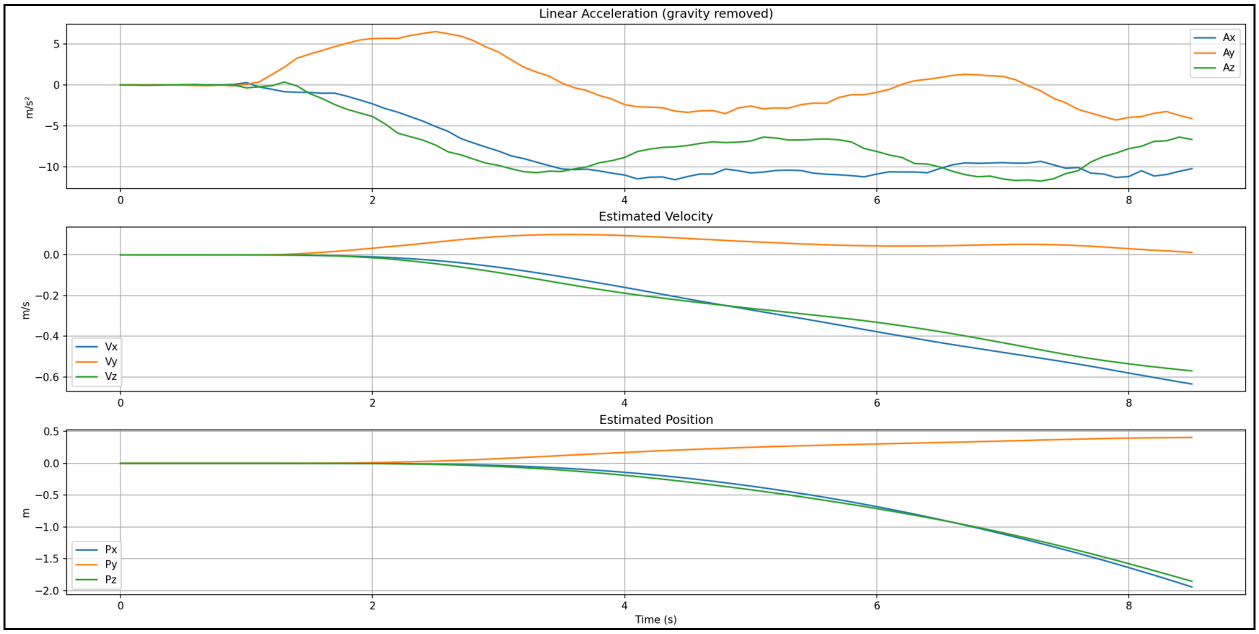


Figure: Acceleration, velocity, and position plots while the watch is worn on a moving arm.

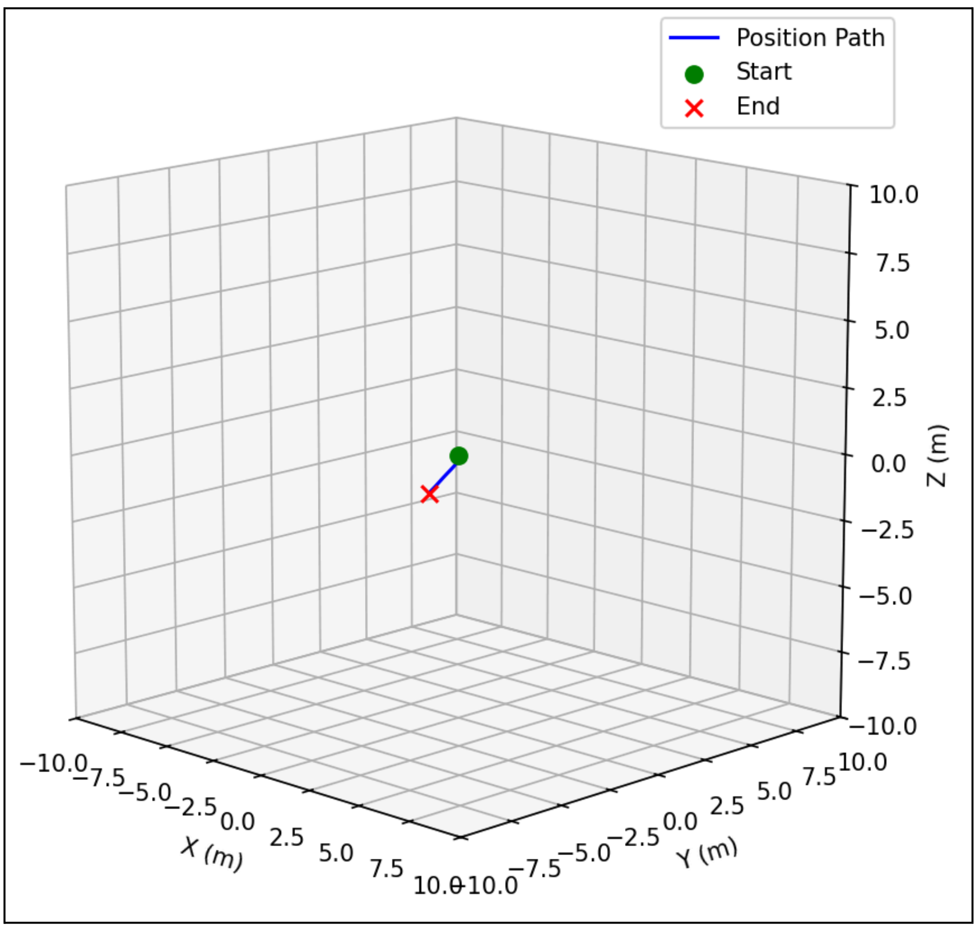


Figure: 3D plot of the watch's estimated trajectory while the watch is worn on a moving arm.

Reflection:

Sensor noise heavily affected the acceleration data, and double integration amplified the error. I attempted to use a Kalman filter to reduce this noise, but it didn't improve accuracy significantly. As a result, I shifted focus to comparing rotation angles instead of position, since it requires one less derivative.

OLED Screen Testing

I attempted to display IMU data on the OLED screen but encountered communication issues—both the IMU and OLED used the same protocol, leading to conflicts. However, I found I could run the IMU first and then output the final data to the OLED.

Reflection:

I should have tested communication compatibility before soldering. This would have helped ensure both components could operate simultaneously using the given wiring configuration.

Final CAD Design

Using PLA 3D printing, I applied design-for-assembly techniques—splitting the case into top and bottom parts for easy battery access. Angled extrusions enabled a push-fit mechanism to keep the case securely closed.

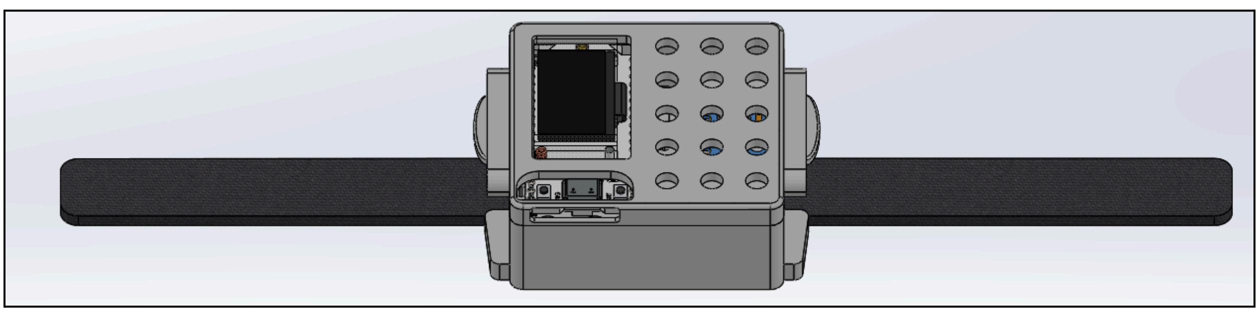


Figure: Final Clock Assembly CAD

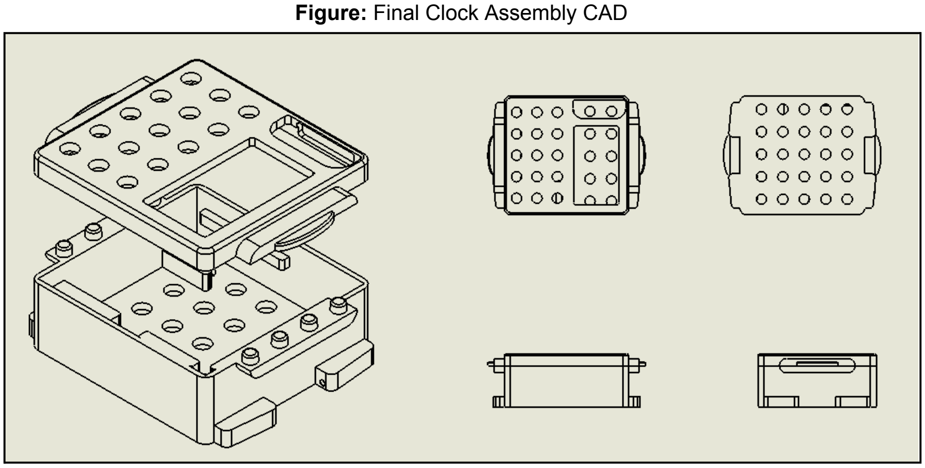


Figure: Final Clock Assembly CAD Drawing

Reflection:

3D printing the case was successful after several iterations. I improved the fit and connector tolerances to accommodate the ESP32 and MPU6050.

Final Testing

I found a beginner tennis player and compared their wrist rotation angles with mine (as a more experienced player). The comparison revealed that their arm was less stable, and the plotted data helped highlight specific time steps where improvements could be made.

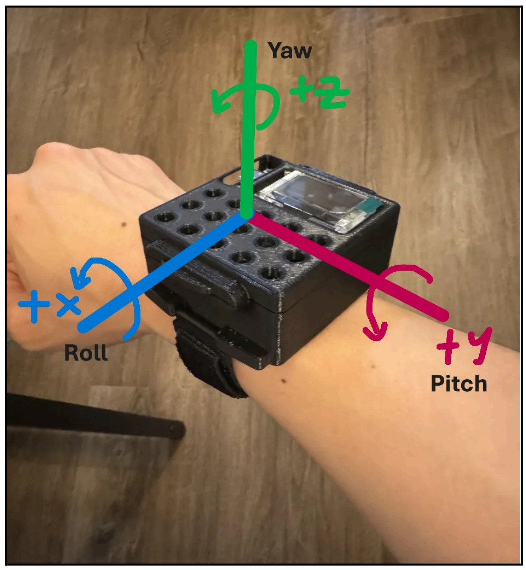


Figure: Euler Angles (On Wrist)

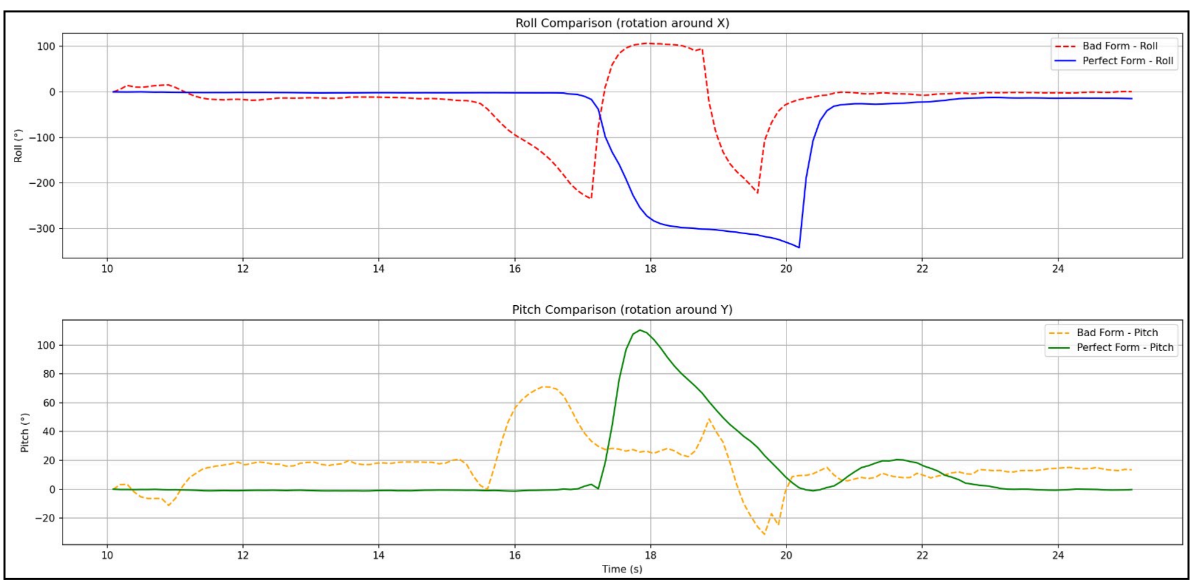


Figure: Data Comparison Between Experienced and Inexperienced Players

Final Reflection:

This project taught me how to navigate the full product development cycle—from identifying a problem to prototyping and user testing. Many unexpected challenges required quick pivots, helping me develop problem-solving and design iteration skills crucial in real-world product development.