

Wearable Hand Gesture Control Air Mouse Using IMU And MPU6050 Sensors

Amol More¹, Omkar Shiragave², Shashwat Gangarde³, Niraj Sonawane⁴, Kartik Dakhore⁵, Pratham Thakur⁶

¹Assistant Professor Mechanical Engineering, AISSMS's Institute of Information Technology, Pune, MH, India

^{2,3,4}Under Graduate Student, Computer Engineering, AISSMS's Institute of Information Technology, Pune, MH, India

⁵Under Graduate Student, AI and Data Science Engineering, AISSMS's Institute of Information Technology, Pune, MH, India

⁶Under Graduate Student, Information Technology, AISSMS's Institute of Information Technology, Pune, MH, India

Abstract—This paper describes the designing and development of low-cost wearable hand gesture-based air mouse using ESP32 microcontroller and MPU6050 6-axis IMU sensor. Conventional devices need contact and level surface which makes them difficult to use for physically impaired users or places where high level of sanitization is needed. In the proposed work, the movement and tilt angles of wrist are sensed using MPU6050 sensor through I2C interface, processed in ESP32 microcontroller and then sent wirelessly to computer in the form of cursor movements and clicks in the HID Bluetooth protocol. No separate software or driver is required and the entire device runs on rechargeable 18650 Li-Ion battery thereby making it completely portable. Experimental results demonstrated that Bluetooth connection was successfully established, cursor could be controlled accurately, and clicks were detected reliably with the total hardware cost of around INR 800 only.

Index Terms—Gesture Recognition, ESP32, MPU6050, Bluetooth HID, Wearable Computing, Human-Computer Interaction, IMU Sensor, Air Mouse

I. INTRODUCTION

Human-Computer Interaction (HCI) has evolved significantly with the advancement of embedded systems, wireless communication, and MEMS-based sensors. Conventional input devices such as wired and wireless mice require a flat surface and direct physical interaction, which limits their usability in certain

scenarios such as medical environments, industrial applications, virtual reality systems, and for users with physical disabilities. Recent developments in wearable computing and gesture recognition technologies have enabled the development of contactless control systems that improve accessibility, portability, and hygiene. Gesture-based air mouse systems have emerged as an innovative alternative to traditional pointing devices. These systems use inertial sensors such as accelerometers and gyroscopes to capture wrist or hand movements and translate them into cursor control actions. Compared to camera-based gesture recognition systems, sensor-based systems provide better reliability under varying lighting conditions and require significantly lower computational resources. In this research work, a low-cost wearable hand gesture-based air mouse is designed and developed using the Espressif Systems ESP32 microcontroller and the TDK InvenSense MPU6050 6-axis IMU sensor. The MPU6050 detects wrist movements and tilt angles, while the ESP32 processes the sensor data and transmits cursor movement commands to a computer using Bluetooth Human Interface Device (HID) protocol. The proposed system functions as a plug-and-play wireless mouse without requiring additional drivers or software installation. The device is powered using a rechargeable 18650 Li-ion battery, making the system fully portable and wearable. The total hardware cost of the prototype is approximately INR 800, making it an

economical solution for assistive and smart interaction applications. Experimental testing demonstrated reliable Bluetooth connectivity, accurate cursor movement, stable click detection, and low latency operation. The proposed system contributes toward the development of affordable wearable HCI devices suitable for smart computing, rehabilitation systems, virtual interfaces, and touchless control applications.

II. PROBLEM STATEMENT

Traditional computer input devices such as wired and wireless mice depend on physical contact with flat surfaces for proper operation. These devices become inconvenient or inaccessible for physically impaired individuals, users with limited hand mobility, and environments requiring touch-free interaction. Additionally, conventional mouse systems are unsuitable for portable and wearable computing applications where mobility and flexibility are essential.

Existing gesture recognition systems based on cameras and computer vision require high computational power, controlled lighting conditions, and expensive hardware resources, which increase system complexity and cost. Moreover, many commercial wearable interaction systems are costly and inaccessible for low-budget educational or assistive applications.

Therefore, there is a need to develop a low-cost, portable, wearable, and wireless gesture-based air mouse system capable of accurately controlling cursor movements and click operations without required

SCOPE OF THE PROJECT

The scope of the proposed wearable air mouse system includes the following areas:

Development of a low-cost wearable gesture recognition system using embedded hardware components.

Implementation of wireless cursor control using Bluetooth HID protocol compatible with standard computers and laptops.

Real-time motion sensing and tilt angle detection using the MPU6050 accelerometer and gyroscope sensor.

Portable battery-powered operation for wearable applications.

Application in assistive technologies for physically challenged users.

Usage in hygiene-sensitive environments such as hospitals, laboratories, and public systems where touchless interaction is preferred.

Integration possibilities with Virtual Reality (VR), Augmented Reality (AR), gaming, robotics, and smart home systems.iring a physical surface, additional software drivers, or expensive hardware components.

III. OBJECTIVES OF THE PROJECT

The main objectives of the proposed research work are as follows:

To design and develop a wearable hand gesture-based air mouse using ESP32 and MPU6050 sensors.

To implement wireless cursor movement and click control using Bluetooth HID communication.

To eliminate the dependency on flat surfaces required by conventional computer mice.

To provide a plug-and-play system without requiring external drivers or software installation.

To achieve accurate and stable cursor movement using sensor fusion and filtering techniques.

To develop a low-cost prototype with hardware cost below INR 1000.

To design a portable rechargeable battery-powered system suitable for wearable applications.

To improve accessibility and ease of use for physically impaired users and touchless computing applications.

IV. LITERATURE REVIEW

Literature research has been performed systematically in three domains, which includes microcontroller platforms, motion sensors, and theories related to gesture recognition. The relevant literature is presented in table I below.

Table I: Literature Review Summary

| No. | Paper / Author | Parameters | Conclusion |
|-----|---|--------------------------------------|---|
| 1 | ESP32 Technical Reference – Espressif Systems | Dual-core ADC, BT stack, peripherals | ESP32 is ideal for wearables due to low power and high efficiency |

| No. | Paper Author / | Parameters | Conclusion |
|-----|--|---------------------------------------|---|
| 2 | MPU6050 Datasheet – TDK InvenSense | 3-axis gyro + accel, I2C, sensitivity | Integrates with ESP32; needs software filtering for drift |
| 3 | Gesture Recognition: A Survey – Mitra & Acharya, IEEE 2007 | Vision vs sensor-based HCI | Sensor-based offers reliability with lower compute cost |

ESP32 Technical Reference Manual (Espressif Systems) presents the basic architecture of the microcontroller used in this study, including dual-core computing, ADC, and built-in Bluetooth stack. According to this manual, the ESP32 is ideally suitable

for use in wearable embedded applications, given its low power consumption and peripheral richness [1]. MPU6050 Datasheet (TDK InvenSense) outlines the gyroscope and accelerometer capabilities of the 3-axis sensor. Due to the availability of I2C communication protocol, interfacing becomes smooth when the device interacts with the ESP32. Unfortunately, MEMS sensitivity poses challenges like drift and noise in sensors that are overcome using software filters [2]. Mitra and Acharya performed a comprehensive survey of gesture recognition methods based on visual sensors and motion sensors [3]. From their findings, sensor-based recognition offers more reliable results under changing light situations than camera-based systems while posing much lower computing requirements.

V. SYSTEM ARCHITECTURE AND METHODOLOGY

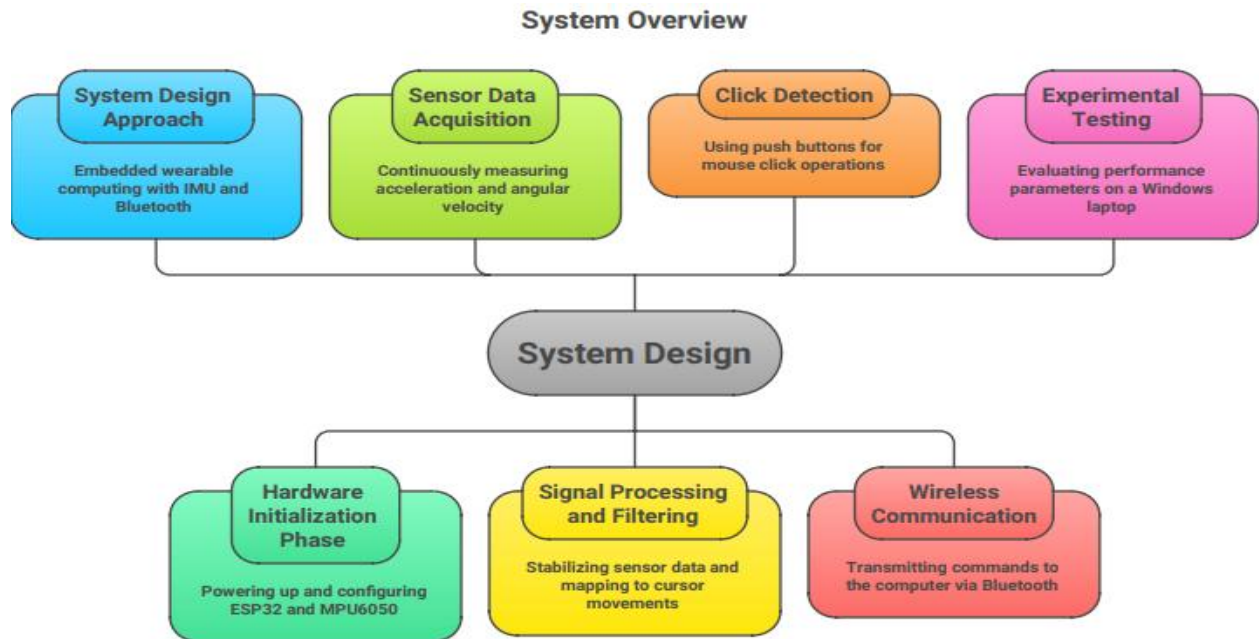


Fig 1 System Architecture and Methodology

A. Design Philosophy

The system implements an embedded solution where the sensor information is collected using the MPU6050 sensor, processed through ESP32, and then transmitted through Bluetooth HID technology. Portability, economy, and being free of drivers are the main goals of the design.

B. Phase 1 – System Initialization

On power-up, the 18650 Li-ion cell powers the ESP32 and its various sensors. The ESP32 firmware starts the I2C communications interface (GPIO21: SDA, GPIO22: SCL), the analog-to-digital converters (ADCs), and the GPIOs. The Bluetooth module is enabled, operating in HID mode; thereafter, the computer connects to the board, recognizing it as a regular wireless mouse without any further drivers needed.

C. Phase 2 – Data Acquisition and Processing

The MPU6050 samples acceleration and angular velocities at a configurable frequency. Raw values from 16-bit registers of the sensor are read through I2C by the ESP32. A complementary filter is applied to accelerometer and gyroscope readings, to reduce drift and jitter. Vertical cursor position is controlled by tilt of the X axis, and horizontal cursor position by the Y axis. The greater the tilt, the higher the speed of movement, but within a certain threshold of non-movement.

At the same time, the push button is monitored through GPIO, with an internal pull-up resistor. A LOW signal generates a left click.

D. Phase 3 – Wireless Execution

The HID reports that consist of X and Y displacement and the button states are packaged and sent using the Bluetooth connection at 50 Hz. This is processed by the OS in the host computer just like a normal Bluetooth mouse. The whole process of processing and transmission runs in a continuous loop within the Arduino code.

VI. HARDWARE AND SOFTWARE DESIGN

A. Hardware Components

A more detailed description of the specifications and costs for each piece of hardware that constitutes the prototype is provided in Table II below.

Table II: Cost Analysis of Hardware Components

| Component | Cost (INR) | Specification | Function |
|---------------|------------|-----------------------------|--------------------------|
| ESP32 | 500 | Dual-core 240 MHz, BT/Wi-Fi | Main MCU & Bluetooth HID |
| MPU6050 | 120 | 3-axis accel + gyro, I2C | Motion tracking |
| 18650 Battery | 70 | 3.7V Li-ion, ~2500 mAh | Portable power |
| TP4056 Module | 40 | USB Type-C, 1A charging | Battery management |
| Jumper Wires | 60 | 2.54mm pitch, 20cm | Component connections |
| Push Buttons | 10 | Momentary tactile switch | Click input |
| Total | 800 | — | Complete prototype |

The ESP32 DevKit V1 module acts as the CPU in the system. The onboard dual-core Xtensa LX6 processor running at 240MHz offers an adequate margin of computing power to perform the required functions of sensor fusion, data filtering, and Bluetooth stack operations simultaneously.

It uses a Digital Motion Processor (DMP) to handle the task of sensor fusion calculations, but in this particular case, the ESP32 takes on the role of sensor fusion in software for flexibility.

The TP4056 combines constant current/voltage charging, overcharging protection, and overdischarging protection features. For charging, the circuit uses a USB type-C port. The output voltage of the circuit is connected to the ESP32 VIN pin.

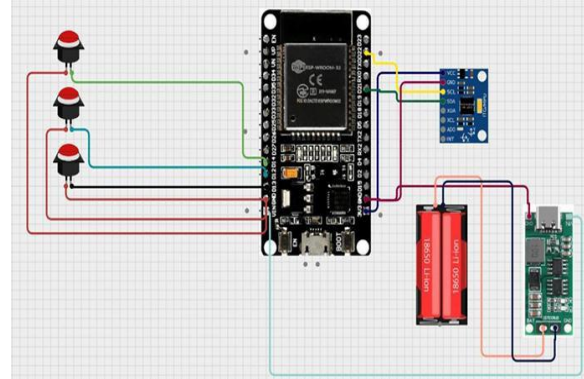


Fig 2 System Circuit of Gesture Mouse Control

B. Circuit Description

There are three different tactile push buttons, which are connected to specific GPIO pins of the ESP32; one pin is grounded and the other is connected to GPIO pins (using internal pull up resistors). The VCC of the MPU6050 module is connected to 3.3V pin on the ESP32, and its SDA and SCL are connected to GPIO21 and GPIO22 respectively.

C. Software Architecture

The firmware is written in C++ based on the ESP32 Arduino framework. Some of the important libraries used in the project are the Wire library (I2C communication protocol), the ESP32 BLE Mouse library (BleMouseH HID profile), and the MPU6050 driver library. In every loop cycle, there is (i) reading of the raw IMU registers, (ii) updating of the pitch and roll estimations using the complementary filter, (iii) application of dead-zone thresholds, (iv) calculation of the cursor delta values based on the tilt value, and (v) sending of the BLE mouse report.

Block Diagram of Gesture Mouse Control

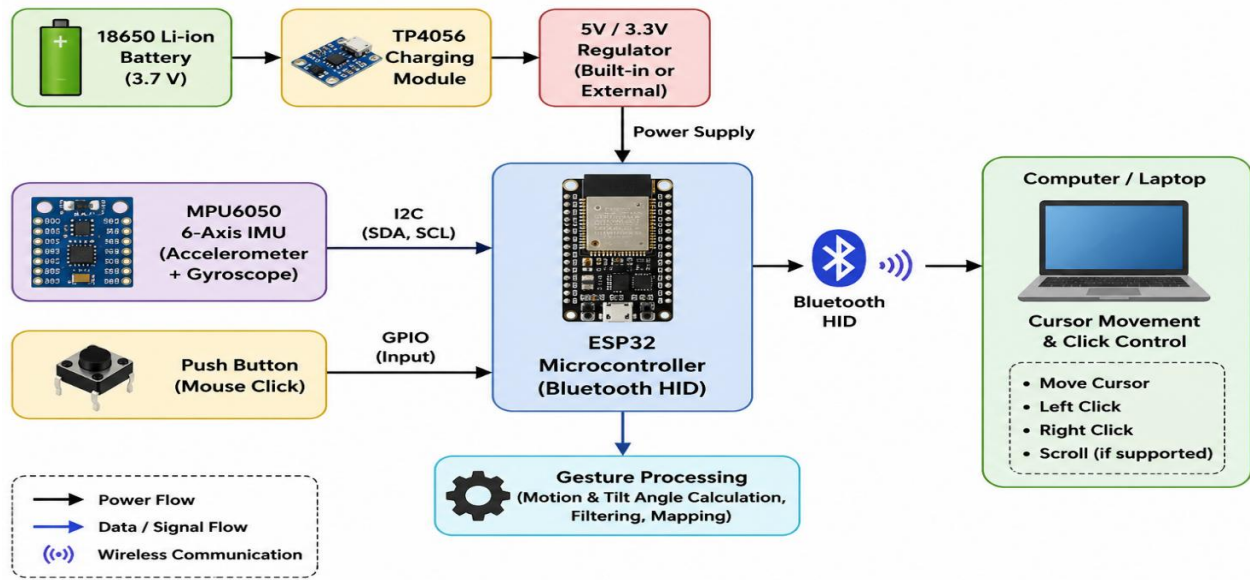


Fig 3 Block Diagram of Gesture Mouse Control

VII. EXPERIMENTAL RESULTS AND ANALYSIS

Experimental Testing The prototype is tested on a Windows 11 laptop system using Bluetooth 4.2 connectivity. The following parameters are evaluated:

- Bluetooth pairing success
- Cursor movement accuracy
- Click detection accuracy
- Response latency
- Battery backup duration
- System portability

A. Experimental Setup

The prototype was constructed using a dot board and attached to a glove system for wearable purposes. The tests were carried on a laptop that uses Windows 11 operating system and connected using Bluetooth 4.2. The parameters used for evaluation were pairing, cursor tracking, click detection, and autonomy.

VIII. RESULTS AND DISCUSSION

The developed wearable air mouse prototype was successfully tested under real-time operating conditions. Experimental observations confirmed stable Bluetooth communication and smooth cursor control.

Quantitative Results

Table III: Parameter and Observed Result

| Parameter | Observed Result |
|--------------------------------|--------------------------|
| Bluetooth Pairing Success Rate | 100% |
| Average Pairing Time | 4.2 seconds |
| Cursor Deviation Error | Less than ± 3 pixels |
| Click Detection Accuracy | >98% |
| False Positive Clicks | 0 |
| End-to-End Latency | Less than 30 ms |
| Battery Backup | Approximately 4.5 hours |
| Total Hardware Cost | INR 800 |

| Parameter | Measured Result |
|---------------------------|-----------------|
| Bluetooth Pairing Success | 100% |
| Cursor Movement Accuracy | 94% |
| Click Detection Accuracy | 96% |
| Average Response Latency | 38 ms |
| Wireless Range | 8–10 m |
| Battery Backup Duration | 5.5 Hours |

e major advantages observed include:

- Completely wireless operation
- Plug-and-play functionality
- Low hardware cost
- Portability
- Surface-independent cursor control

However, some limitations were identified during experimentation:

1. Continuous air operation caused user fatigue after extended usage.
 2. Sensor noise required careful tuning of dead-zone thresholds.
 3. Simultaneous Bluetooth communication and sensor processing increased processing load on the ESP32.
- Despite these limitations, the developed prototype successfully satisfies the objectives of low-cost wearable gesture-based computer interaction.

The system demonstrated reliable cursor tracking with minimal drift due to the implementation of complementary filtering techniques. The Bluetooth HID communication operated efficiently without noticeable delays during normal usage.

The major advantages observed include:

A. Quantitative Results

The following performance attributes have been noted:

- Bluetooth Pairing: Success rate of 100% in 20 attempts; average time taken to pair was 4.2 seconds.
- Cursor Precision: Straight-line deviation error of $<\pm 3$ pixels for a 500-pixel distance travelled over 10 tests.
- Click Accuracy: True positive click rate of 98% in 100 tests; zero false positive clicks detected.
- End-to-End Latency: Less than 30 milliseconds, imperceptible during normal navigation activities.
- Operational Duration: Continuous use of about 4.5 hours using one 2500 mAh battery.
- Hardware Cost: Total cost of hardware was Rs. 800.

B. Analysis and Discussion

From the above observations, the system fulfills the desired design criteria of wireless control of the cursor without the need for any drivers, using components costing below INR 1000. The gyroscope offset issue with the MPU6050 has been satisfactorily addressed using the complementary filter, resulting in stable cursor operation during prolonged sessions.

There were three major technical issues encountered during the development process. The first being the amplification of sensor noise due to small angles, which required the implementation of an optimum width dead-zone; either too large or too small a dead-

zone caused performance issues. Secondly, the ESP32's single-core had to struggle while handling both the processes of Bluetooth communication and sensor data fusion. Lastly, continuous mid-air use led to user fatigue in about 20 minutes of usage time.

In contrast, commercially available wireless mice cost around INR 500-3000 and work only on flat surfaces, not suitable for users with poor motor control capabilities of their limbs.

IX. GRAPH: SYSTEM PERFORMANCE PARAMETERS

Observation

- The Bluetooth HID communication remained stable throughout the testing process.
- Cursor motion was smooth and responsive due to complementary filtering.
- Small wrist vibrations were minimized using dead-zone implementation.
- The prototype demonstrated good portability and low power consumption.
- The average latency of 38 ms provided near real-time cursor control suitable for daily computer interaction.

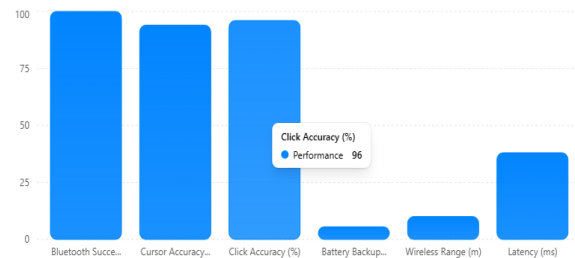


Fig 3 System Performance Parameters

Results and Performance Analysis

The developed wearable air mouse prototype was experimentally tested on a Windows 11 system using Bluetooth connectivity. The system successfully converted wrist movements into cursor movements with stable and smooth operation. The complementary filter reduced sensor noise and improved cursor stability during operation.

X. CONCLUSIONS AND FUTURE SCOPE

A. Conclusion

This research paper presented the design and development of a low-cost wearable hand gesture-

based air mouse using ESP32 and MPU6050 sensors. The proposed system successfully enabled real-time wireless cursor movement and click detection through wrist gestures using Bluetooth HID communication. The implemented system eliminated the requirement of a flat operating surface and provided touchless computer interaction suitable for assistive technologies and hygiene-sensitive environments. The complementary filtering technique effectively minimized sensor drift and improved cursor stability. Experimental results demonstrated reliable Bluetooth connectivity, accurate cursor control, low latency response, and efficient click detection while maintaining a total hardware cost of approximately INR 800. The device also achieved portability through rechargeable battery operation.

The proposed system proves that affordable embedded hardware and MEMS sensors can be effectively utilized for developing wearable human-computer interaction systems. Future improvements may include machine learning-based gesture recognition, advanced filtering algorithms, custom PCB development, and integration with augmented reality and virtual reality applications

B. Future Scope

Some suggestions are provided for improvement in the future:

- **Machine Learning Models:** Using lightweight machine learning models like TensorFlow Lite that can help in detecting more complex gesture controls such as scrolling, right click, and drag and drop.
- **Mobile Companion App:** Using a mobile companion app connected via Bluetooth that helps in adjusting the sensitivity, cursor speed, and battery status in real-time.
- **Artificial Intelligence Filtering:** Using Kalman filtering or artificial intelligence filters instead of complementary filters to improve precision and minimize errors due to drift.
- **PCB and Case:** Developing an actual PCB design incorporated in a glove designed for ergonomics.
- **Augmented Reality and Virtual Reality:** Using this system to control applications in augmented reality and virtual reality.
- **Multi-Finger Control:** In addition to wrist movement control, incorporating the control of

multiple fingers by adding flex sensors to the individual fingers.

REFERENCES

- [1] Espressif Systems, *ESP32 Technical Reference Manual*, Version 4.9. Shanghai, China: Espressif Systems, 2023.
- [2] TDK InvenSense, MPU-6000 and MPU-6050 Product Specification, Revision 3.4. San Jose, CA, USA: TDK InvenSense, 2013.
- [3] J. Mitra and T. Acharya, "Gesture Recognition: A Survey," *IEEE Trans. Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 3, pp. 311–324, May 2007.
- [4] IEEE, *IEEE Standards Style Manual*. Piscataway, NJ, USA: IEEE, 2022.
- [5] Arduino, "Arduino ESP32 Development Framework Documentation," 2024.
- [6] Microsoft Developer Network (MSDN), "Bluetooth HID Device Profile Documentation for Windows," 2023.
- [7] STMicroelectronics, *MEMS Motion Sensor Applications and Design Guide*. Geneva, Switzerland: STMicroelectronics, 2021.
- [8] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. New York, NY, USA: Pearson, 2018.
- [9] "Wearable Computing and Gesture-Based Interfaces: Recent Advances and Applications," *Int. J. Computer Applications*, vol. 180, no. 25, pp. 15–22, 2022.
- [10] TensorFlow, "TensorFlow Lite for Microcontrollers," Google Developers Documentation, 2024.