Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X
Volume 4, Issue 4, 2026

A HAND WEARABLE GAMING CONTROLLER

Muzamil Adrees*[1], Muhammad Faheem Khan[2] Muhammad Yousaf[3], Muhammad Faizan Jillani [4], Ahmed Raza Asad[5], Ahmed Imran[6], Attiq-Ur-Rehman[7],

[*1,2,3,4,5,6,7] *BS Computer Science, [Times University], Pakistan*

DOI: https://doi.org/10.5281/zenodo.20675328

Keywords
Wearable Controller, MPU6050, Flex Sensors, Gaming, IoT Devices

Article History
Received: 28 February 2026
Accepted: 10 April 2026
Published: 27 April 2026

Copyright @Author

Corresponding Author: *
Muzamil Adrees

Abstract
We designed this project, named "A Hand Wearable Gaming Controller," on the Arduino microcontroller, using Bluetooth Low Energy (BLE) to mimic a personal computer gamepad. It translates natural hand movement through IMU sensors and flex sensors back into standard HID gamepad input. Think of it like a helpful IoT wearable a wireless bridge between your movements and game commands. The open-source BLEGamepad platform makes it responsive, eliminating latency while making the controller play well on any platform. Whether you're playing classic arcade games, the glove's ergonomic design frees up completely new ways of game control. Honestly, it changes the way you play. But here's what we're most excited about: Beyond games, it provides opportunities for accessibility and rehabilitation therapy. Patients may be able to use simple hand gestures for physical rehab exercises. We're keeping it lean and low-cost the whole setup costs under $40 to produce. That means both immersive gaming and medical motion therapy could become a whole lot more accessible.

1. INTRODUCTION

The Internet of Things (IoT) is an architecture that describes how everyday products, relative detectors, processors, and network connectivity can gather, add to, and respond to data with little or no human intervention. Wearable computing, which is typically used on the Internet of Things, describes technical devices worn on the human body that contain wireless communication modules as well as detectors incorporated. These devices can capture movements or physiological readings and report in real-time to other gear or cloud computing infrastructure. Intelligent eyeglasses and fitness trackers have been embodiments of these wearable computers; they can be employed to capture realistic video gameplay, user authentication, and health monitoring.

The MPU-6050 Motion Sensor, which features a 3-axis gyro and accelerometer with embedded Motion Fusion TM for accurate 6-axis tracking, is the central component of our glove controller (InvenSense Inc., 2013). Fraden (2016) goes into great detail on sensors in general how they convert physical signals into electronic data and why wearables require both analog front-ends and intelligent processing..

Modern videogame wearables such as the Oculus Touch demonstrate the advanced techniques of haptic feedback, ease of use, and capacitive touch (Oculus VR, 2020). Flex sensors are the preferred option for do-it-yourself finger bend detection; they act as a voltage filter into the Analog-to-Digital Converter (ADC) of your microcontroller and rest at about 30 kΩ flat before rising to 70 kΩ at a 90° bent (SparkFun Electronics, 2022).

The Internet of Things (IoT) is an architecture where everyday objects with sensors, processors, and networking connectivity can capture, disseminate, and process information with minimal human interaction. Wearable computing, in the context of the Internet of Things, is electronic computing devices worn on the body that incorporate wireless modules and sensors. The devices can capture motion or physiological information and send it in real-time to other devices or cloud platforms. Wearable computing examples include smart glasses and fitness trackers; the devices can support immersive gaming, user authentication, and health monitoring. In gaming, wearable controllers are used to convert hand gestures and movements into commands that computers or gaming consoles can understand. Such interfaces have a tendency to remove conventional input devices, such as keyboards, thus improving immersion by minimizing the cognitive dissonance between a player's natural gestures and on-screen responses.

We utilize a passivity-based dead-zone compensatory approach from robots to level out unresponsive sensing gaps earlier than they interfere with gaming in order to clear up noisy motion information (Chan-Zheng et al., 2022). The ESP32-BLE-Gamepad package then uses a conventional HID gamepad interface to send all that selective information over the Bluetooth Low Energy (BLE) so your PC just sees a regular joystick (BleGamepad Library, 2023).

Our work combines all of these elements to create a freely available, reasonably priced, BLE-compliant wearable glove that can be used on multiple platforms and assembled by anybody.

This project aims to use an ESP32 microcontroller to build an electronic gamepad wearable on the hand. It is envisioned to build an affordable, fully programmable gamepad supporting multiple different platforms by combining data from IMUs and flex sensors to generate joystick and button inputs via BLE HID. The aim is to offer interactive gesture-based gaming and explore opportunities in motor rehabilitation and accessible gaming interfaces through sensor fusion, user-friendly layout, and BLE HID firmware.

Currently, existing wearable controllers are proprietary, platform-specific (console-specific, for example) or costly. There is no heterogenous (that works on multiple platforms), open-source and low-cost wearable controller that is simple to quickly prototype and customize.

I. Significance

Our controller will make gestural PC gaming accessible to everyone and serve as the basis for professional use in virtual reality (VR) or augmented reality (AR) and in medical rehabilitation, where natural hand movements can be used to enable therapy and patient engagement.

II. Objectives

- Design a low-cost, wearable hand controller using ESP32 and motion sensors to enhance gaming.
- Develop an intuitive gesture-to-mapping system to translate natural hand movements to precise gamepad input.
- Test performance of the controller in gaming and in latency, accuracy, and usability.

III. Research Questions

- How well can flex sensors and dual IMUs mimic typical gamepad inputs with gesture control?
- How is the under 20 ms latency of the controller impacting user immersion in gaming and motor-skill rehabilitation?
- Can an open-source, sub-$40 design hold its own against commercial ones in functionality and usability?

2. Literature Review

Smith (1997) gives a thorough introduction to digital filters, emphasizing their construction and use in audio systems. Phase preserving methods, frequency and time-domain analysis, and lowpass and comb filters are important subjects. It is useful for real-time signal processing tasks, like deciphering sensor data in wearable controllers, because of the useful MATLAB examples (Smith, 1997).

Mahony et al. (2008) recommend applying nonlinear complementary filters on the SO(3) group to generate more accurate attitude estimates from cheap IMUs that get impacted by noise and bias. Their direct and passive filter designs make sure that the system stays stable and can be used in real-time embedded systems. These filters are very useful for wearable gadgets that need MPU sensors for precisely tracking their orientation (Mahony et al., 2008).

Using a quaternion-based gradient descent algorithm, Madgwick (2010) produced an orientation filter for IMU and MARG sensor arrays to be easy on devices. The filter operates well when making up for gyroscope drift and magnetic distortion, offering it high accuracy with low processing demands. This renders it perfect for real-time, embedded gadgets like wearable motion controllers (Madgwick, 2010).

The MPU-6050 is an integrated circuit which incorporates a 3-axis gyroscope and a 3-axis accelerometer. It comes with a built-in Digital Motion Processor (DMP) that allows for real-time sensor combination with effective motion tracking. Its versatile register map lets you set the sensitivity and modes of communication exactly how you want them, making it optimal for wearable systems that need low-latency, 6-axis motion input (InvenSense Inc., 2013).

From simple sensors to sophisticated sensors with integrated processing, Fraden (2016) provides a thorough analysis of almost every kind of sensor you can imagine. The book describes how sensors function, how they link electronics and the actual environment, and why they are crucial to contemporary gadgets, particularly wearables. When working with flex and motion sensors, as we did in our hand controller project, it's a terrific resource (Fraden, 2016).

The Oculus Touch controller manual (Oculus VR, 2020) presents a comprehensive rundown of ways to configure, run, and debug the device. When designing user-friendly and responsive handheld or wearable controllers like ours, it highlights key design options like reactive touch, haptics, and ergonomic handling. In addition, it provides sensor configurations and usage of energy, which helped us comprehend the industry norms for VR input devices (Oculus VR, 2020).

In their research, Chan-Zheng et al. studied dead-zones in machines, which are simply the areas of hardware like motors or sensors which are not active. To even things , they devised a passivity-based method of control that included power and suspension changes. In real-world experiments using a 2-DOF artificial hand, the method they used behaved nicely so it is relevant to us given wearable controllers also have to cope with not linear input activity (Chan-Zheng et al., 2023).

The SparkFun Flex Sensor is basically a resistor that changes its resistance depending on how much it's bent the more you bend it, the higher the resistance goes. They've actually been used in stuff like the Nintendo Power Glove, which is pretty cool. Flex sensors come in two sizes (2.2" and 4.5") and can reach up to 70kΩ resistance at a 90° bend. To get useful data out of them, they're usually combined with a regular resistor to make a voltage divider, which then feeds into a microcontroller's analog pin. The guide also suggests using simple stuff like an Arduino, breadboard, and jumper wires to get started with testing (SparkFun Electronics, 2022).

Similar to a tiny storage device within the board's microcontroller, the Arduino EEPROM module allows you to store content that is retained even after the board is turned off. Storage sizes vary depending on the chip (for example, the ATmega328P has 1024 bytes, whereas the ATmega2560 has 4 KB). Basic functions like read(), which allows you to retrieve data from a specified memory region, are included in the library; however, any area that hasn't been written to previously often reads 255. All you need to do is add <EEPROM.h> to the code (Arduino, 2023).

The ESP32-BLE-Gamepad library lets an ESP32 act as a Bluetooth game controller with support for 128 buttons, multiple axes, sliders, and D-pads. It uses the efficient NimBLE library and allows full customization of HID settings and BLE info. Compatible with Windows, Android, Linux, and macOS (but not iOS) (BLEGamepad Library, 2023).

How Bluetooth-enabled devices find and connect to Human Interface Devices (HIDs), such as keyboards, mouse, or bespoke controllers, is specified in the Bluetooth HID Profile v1.1.1.

To guarantee correct compliance and functioning, it operates over the L2CAP layer and incorporates required fixes (such as Errata 23224) (Bluetooth SIG, 2023). The design, functionality, and programming techniques of the hardware modules of the ESP32 SoC are covered in detail in the ESP32 Technical Reference Manual v4.8. Most notably, it describes the ULP coprocessor, a low-power device optimal for ultra-low-power applications that operates under deep sleep to perform functions like reading sensors or RTC data without waking the main CPU (Espressif Systems, 2024).

Few open-source, low-cost solutions combine dual motion sensors and flex sensors to emulate gamepad inputs over Bluetooth for PC games, let alone support medical-grade rehabilitation exercises.

3. Methodology

In our project we are using multiple tools to create A Hand Wearable Gaming Controller. The parts we are using are given below.

Research Gap

Table 1: Table for Hand Wearable Controller

Sr#	Equipment	Use	Model
1	ESP32 Microcontroller	Main controller for BLE HID	ESP32 DevKit V4
2	MPU-6050 (x2)	Motion sensing for both hands	MPU-6050
3	Flex Sensors (x4)	Detect finger bending	2.2″ Bend Sensors
4	Tactile Push-Buttons (x4)	Standard face buttons	6×6mm Tactile Switch
5	Firmware Library	BLE HID gamepad interface	BLEGamepad (Arduino)
6	ESP32 Microcontroller	Main controller for BLE HID	ESP32 DevKit V4
7	MPU-6050 (x2)	Motion sensing for both hands	MPU-6050

Institute for Excellence in Education & Research

SDLC Model

Two-week sprint Agile methodology, culminating in a working demo:

- **Sprint 1:** Sensor Calibration and Data Collection
- **Sprint 2:** BLEGamepad integration and basic button mapping
- **Sprint 3:** Merging dual-stick motion & drift compensation
- **Sprint 4:** Force calibration & integration of flex-sensor
- **Sprint 5:** Enclosure design and usability testing

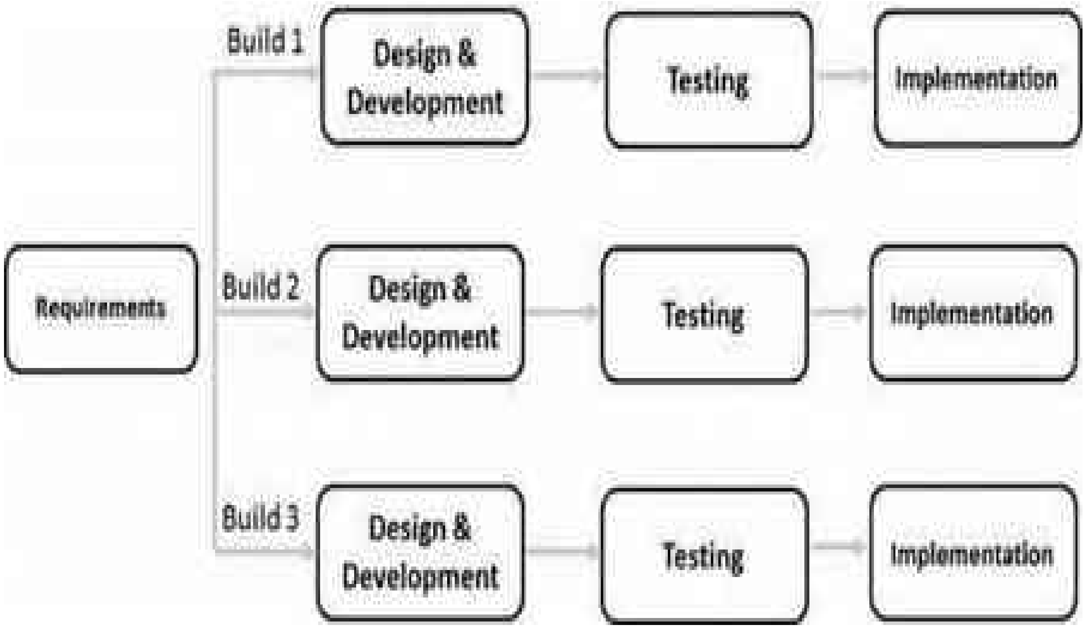

Figure 1: Iterative SDLC Model

Data Flow Diagram

Hand and finger movements are recognized and translated into joystick and button input. They are sent wirelessly via Bluetooth to the game, allowing for real-time.

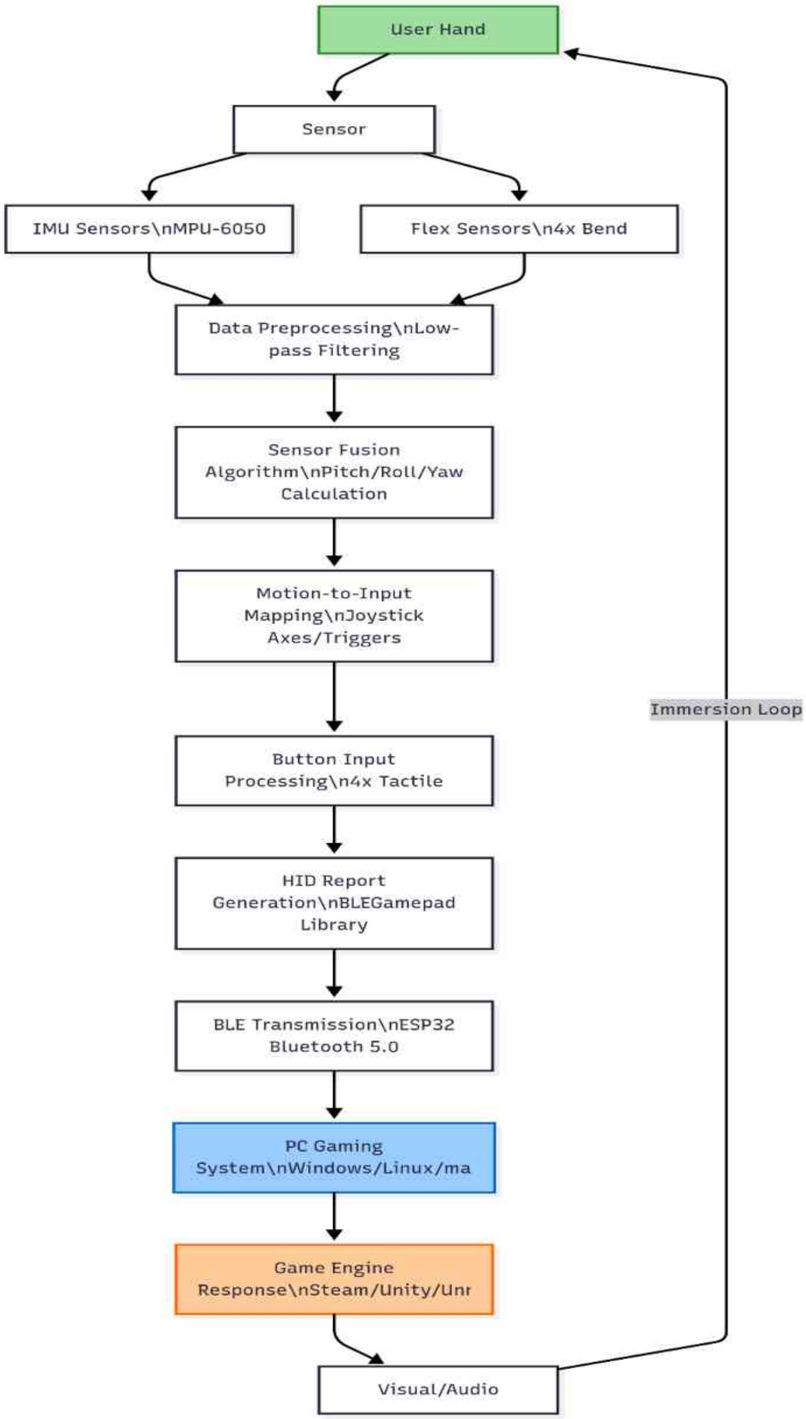

Figure 3: DFD Level-0 for Hand Wearable Controller

Work Flow Diagram

Read Sensors → Check Drift → Fuse Data → Map Gestures → Transmit HID

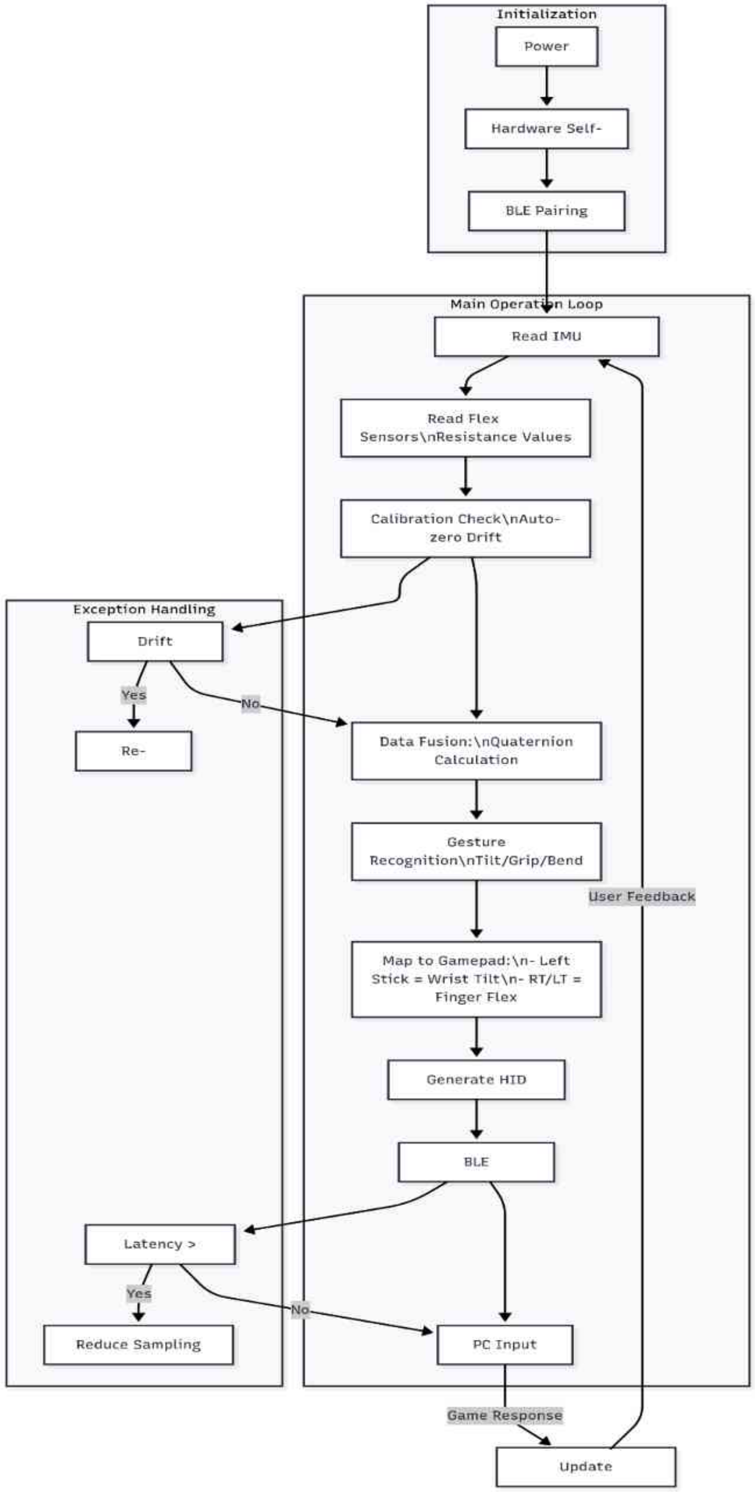

Figure 3: Work Flow Diagram for Hand Wearable Controller

UML Diagram
Wearable Controller: The central coordinator system.
MPU6050: Manages motion data (pitch/roll/yaw)
Flex Sensor: Converts bending to activation values.
HID Mapper: Maps gestures to game controls
BLEGamepad: Manages wireless transmission
Relationships: Structure (1 controller to 2 IMUs and 4 flex sensors)

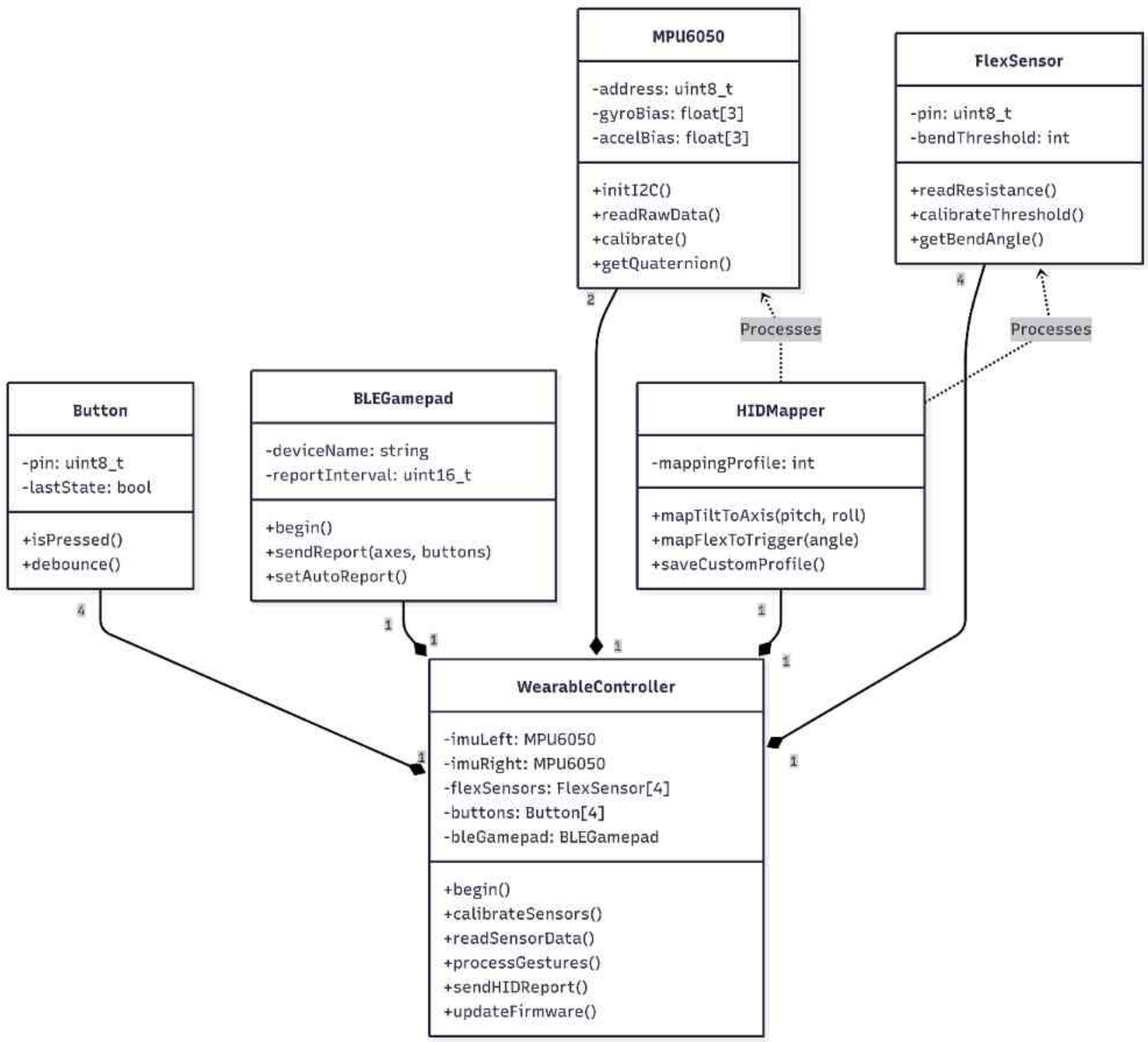

Figure 4: UML Diagram for Hand Wearable Controller.

REFERENCES

[1] Arduino EEPROM Library. (2023). *EEPROM write/read best practices*. Arduino Documentation. https://docs.arduino.cc/learn/built-in-libraries/eeprom

[2] BleGamepad Library. (2023). *ESP32-BLE-Gamepad* [GitHub Repository]. https://github.com/lemmingDev/ESP32-BLE-Gamepad

[3] Bluetooth SIG. (2023). *HID service specification v1.1.1*. https://www.bluetooth.com/specifications/specs/human-interface-device-service-1-1-1/

[4] Chen-Zheng, C., Borja, P., & Scherpen, J. M. A. (2022). *Dead zone compensation via passivity-based control for a class of mechanical systems*. In J. Trumpf & R. Mahony (Eds.), *12th IFAC Symposium on Nonlinear Control Systems NOLCOS 2022* (pp. 319–324). https://doi.org/10.1016/j.ifacol.2023.02.054

[5] Espressif Systems. (2024). *ESP32 technical reference manual v4.8*. https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf

[6] Fraden, J. (2016). *Handbook of modern sensors* (5th ed.). Springer. https://doi.org/10.1007/978-3-319-19303-8

[7] InvenSense Inc. (2013). *MPU-6050 register map and descriptions (Rev 4.2)*. TDK Corporation. https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6050-Register-Map1.pdf

[8] Mahony, R., Hamel, T., & Pflimlin, J.-M. (2008). Nonlinear complementary filters on SO(3). *IEEE Transactions on Automatic Control*, *53*(5), 1203–1218. https://doi.org/10.1109/TAC.2008.923738.

[9] Madgwick, S. O. H. (2010). An efficient orientation filter for inertial/magnetic sensor arrays. In *Proceedings of the International Conference on Rehabilitation Robotics (ICORR)*. https://doi.org/10.1109/ICORR.2010.5645866

[10] Oculus VR. (2020). *Oculus Touch controller overview*. Oculus Developer Documentation.

[11] Smith, J. O. (1997). *Introduction to digital filters with audio applications*. W3K Publishing. https://ccrma.stanford.edu/~jos/filters/

[12] SparkFun Electronics. (2022). *Flex sensor hookup guide*. https://learn.sparkfun.com/tutorials/flex-sensor-hookup-guide

[13] Khurshid, R. P., Fitter, N. T., Fedalei, E. A., & Kuchenbecker, K. J. (2017). Effects of grip-force, contact, and acceleration feedback on a teleoperated pick-and-place task. IEEE Transactions on Haptics, 10(1), 40–53. https://doi.org/10.1109/TOH.2016.2573301

[14] Kim, M., Jang, I., Lee, Y., Lee, Y., & Lee, D. (2016, August). *Wearable 3-DOF cutaneous haptic device with integrated IMU-based finger tracking*. 2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Xi'an, China. 649–649. https://doi.org/10.1109/URAI.2016.7733974

[15] Lee, J., & Bae, J. (2015, August). Design of a hand exoskeleton for biomechanical analysis of the stroke hand. 2015 IEEE International Conference on Rehabilitation Robotics (ICORR), Singapore. 484–489. https://doi.org/10.1109/ICORR.2015.7281246