ECE 445 - Fall 2025

Design Document

Glove Controlled Mouse

Team 20

Khushi Kalra (khushik3), Vallabh Nadgir (vnadgir2), Vihaansh Majithia (vjm5)

TA: Frey Zhao

Table of Contents:

1. Introduction

1.1. Motivation

For digital artists, designers, and even casual computer users, traditional input devices such as mousepads, trackpads, and external mice impose fundamental limitations on natural hand interaction. These devices restrict user motion to a small, flat surface, which disrupts gestures that people use for sketching, painting, or writing. As a result, tasks that require precision and fluid motion, like free-form drawing, digital sculpting, or handwriting, become awkward and fatiguing. While styluses and pen tablets try to address these issues, they often require specialized hardware, are limited to specific surfaces, and can cost hundreds of dollars. Furthermore, they can only be used in dedicated setups, reducing portability and convenience for artists who want to work anywhere.

Existing gesture-based input systems, such as those that use external cameras, like Kinect, introduce a different set of challenges. These systems require users to remain within the camera's line of sight, depend on lighting conditions in the location, and occupy significant desk space. Moreover, the setup and calibration process can be tedious, and its high cost limits usage for hobbyists or students. As a result, the current ecosystem of input devices is fragmented between low-cost, restrictive devices (like mice) and high-cost, very complicated setups (like camera or pen-tablet systems). There is a clear gap in the market for a portable and affordable input device that seamlessly translates hand gestures into digital actions, without relying on external cameras or very specific surfaces

This problem is not limited to artists only. In everyday computing, people naturally move their hands in three-dimensional space to point, pinch, and rotate. A wearable device that directly understands hand motion into digital control could change human-computer interaction by bridging the gap between physical and virtual input, empowering users to interact with their devices more freely and intuitively.

1.2. Solution

To address these limitations, we propose a wearable glove-based input system that enables users to control a computer cursor and execute click actions using natural hand and finger movements. There are no external cameras or tracking stations required to run this glove. This glove integrates a suite of onboard motion sensors to capture the user’s hand orientation, rotation, and movement in real time. These raw sensor readings are processed on a microcontroller (e.g., an ESP32) to compute motion vectors, which are then transmitted to a computer via Bluetooth. The result is smooth cursor control that mimics the natural motion of a hand guiding a mouse, but without being tethered to a flat surface.

Mouse clicks are added through pinch-based gestures, detected via pinching your fingers. This allows for “click” and “drag” actions that feel natural. Additionally, the glove adds haptic feedback modules, such as small vibration motors, so that users receive tactile confirmation of their actions. This eliminates the need for constant visual attention to the cursor, which is especially beneficial during artistic tasks or immersive applications like 3D modeling and virtual reality.

The system’s architecture prioritises portability and affordability. Unlike camera-based systems, this solution works in any environment like indoors, outdoors, or on the go, and does not require calibration for lighting or positioning. It brings the freedom of 3D hand movement to ordinary laptops and desktops, merging the precision of a mouse with the fluidity of gesture control. Beyond digital art, this system has potential applications in gaming, accessibility devices for users with mobility challenges, VR/AR interfaces, and educational tools that leverage natural motion for interaction. By combining low-cost hardware with intuitive design, the glove represents a practical step toward more human-centered, spatially aware computing interfaces.

1.3. Visual-Aid

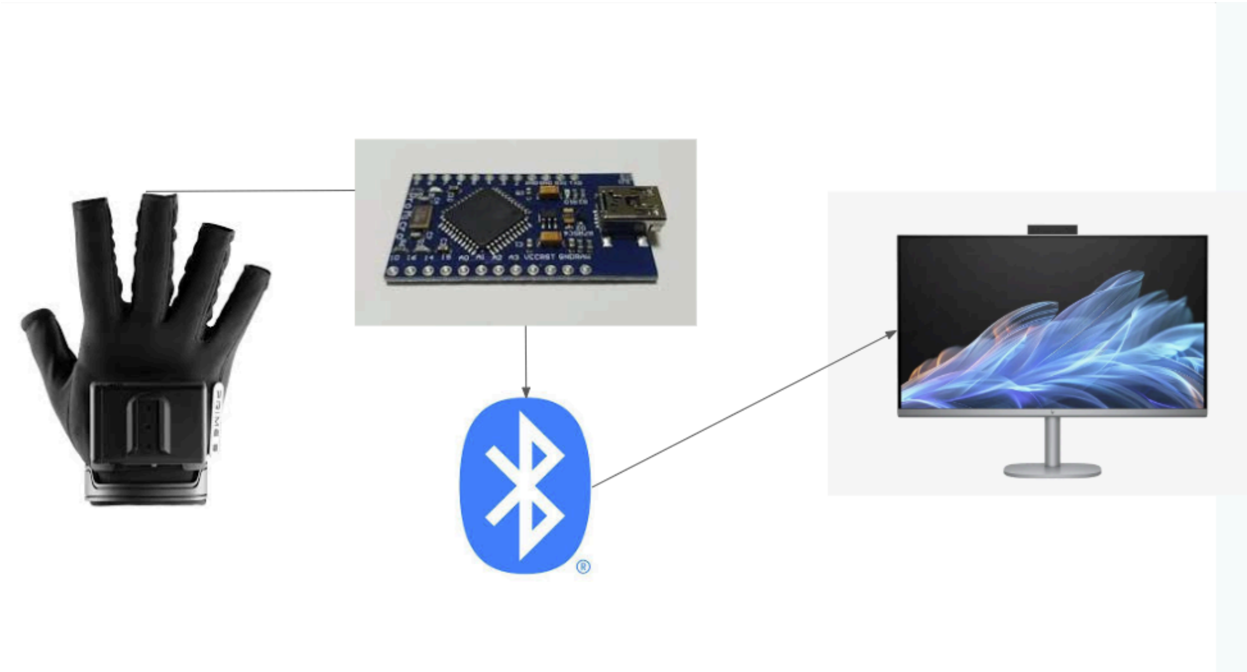

Figure 1: Visual Aid

1.4. High-Level Requirements

To consider our project successful, our safety suite must fulfill the following:

1) The glove must achieve an end-to-end latency under 100 milliseconds between motion or gesture input and corresponding cursor or haptic output.

2) The motion tracking subsystem must provide orientation updates at a minimum rate of 100 Hz.

3) The gesture detection subsystem must recognize intentional touch activations with at least 97% accuracy while maintaining a false-positive rate below 2%.

2. Design

2.1. Block Diagram

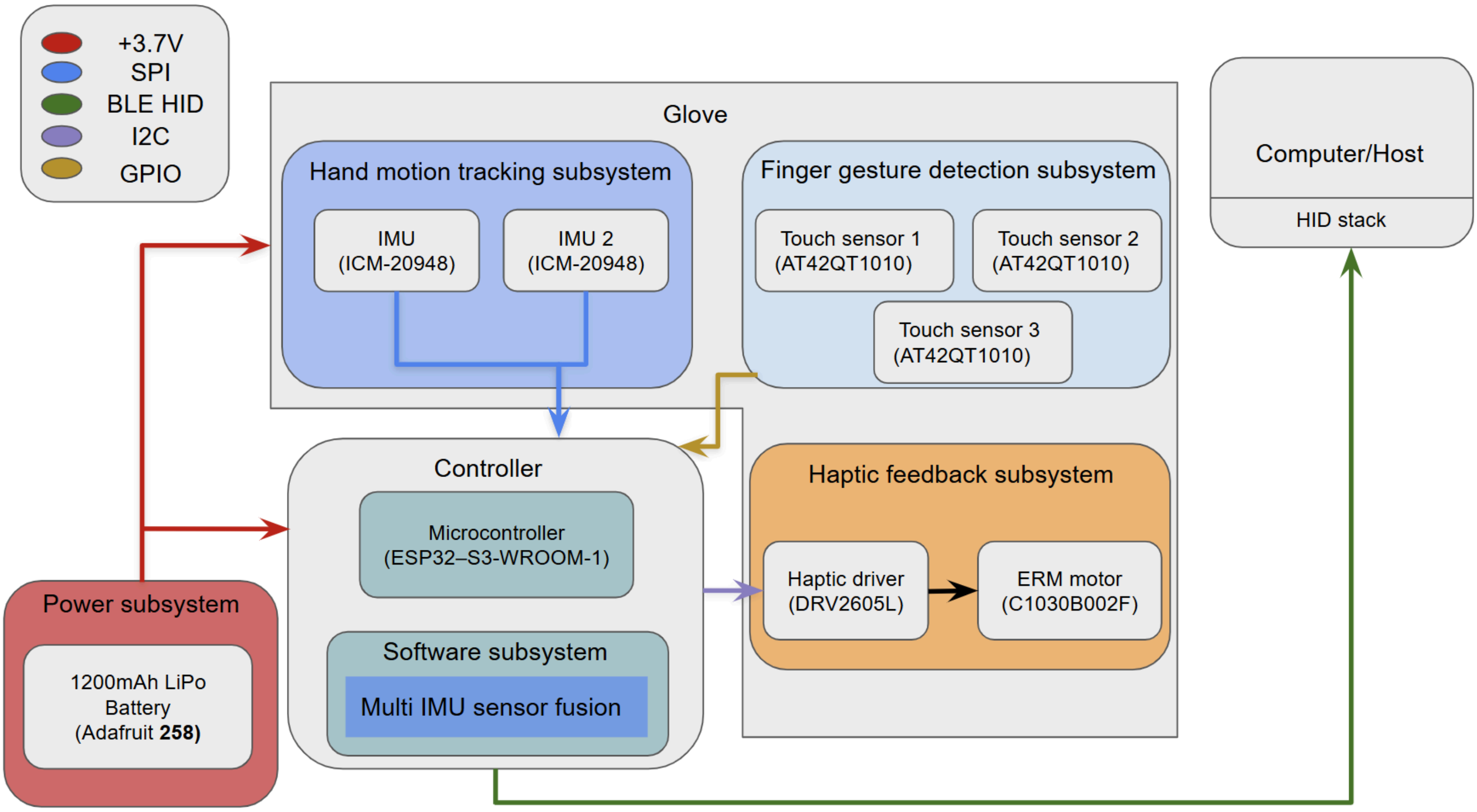

Figure 2: Block Diagram

2.2. Physical Design

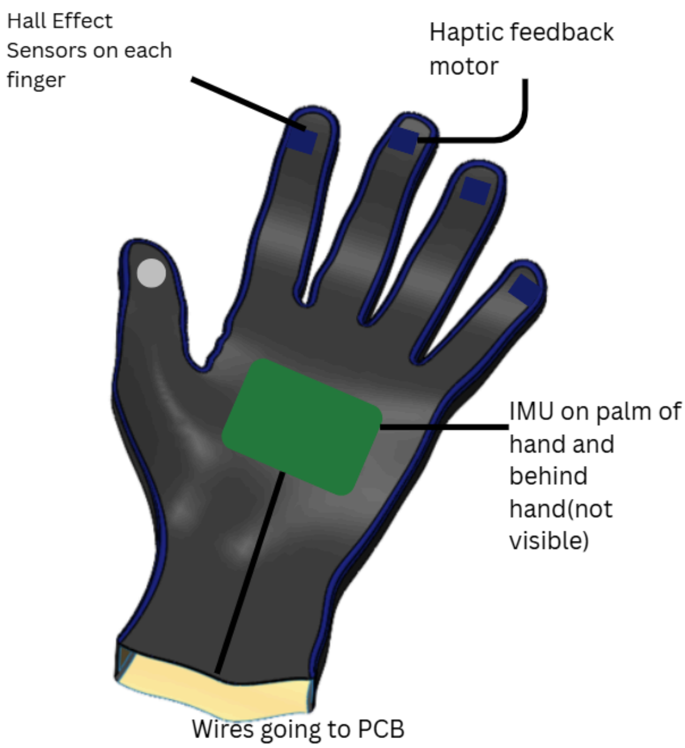

Figure 3: Physical glove design

The physical layout of the glove is shown in Figure 3. The system consists of four primary components: the Hall effect sensors, the IMU, haptic feedback motor, and the microcontroller connection wiring. Each of these components is strategically placed to ensure accurate sensing during operation.

The Hall effect sensors are mounted at the fingertips of the glove, one on each finger. These sensors are used to detect finger pinches and gestures by measuring the magnetic field variations caused when two fingers come close together. By embedding small magnets on opposing fingertips, the sensors can determine the distance between fingers with high precision, allowing the software to reliably interpret pinch gestures as mouse clicks. Each Hall effect sensor is secured to the fabric of the glove.

The IMU module is placed on the palm of the hand and a secondary unit behind the hand (not visible in the figure). This configuration ensures that both rotational and translational hand movements are captured accurately, regardless of hand orientation. The IMUs continuously track hand movement, which is processed to control the on-screen cursor position. The placement on the palm provides a stable surface for motion tracking while maintaining user comfort and freedom of movement.The haptic feedback motor is located near the base of the index finger, as shown in Figure 3, and serves as the tactile output for the glove. When a pinch gesture or click event is recognized, the motor briefly vibrates to signal successful input. This provides non-visual feedback, allowing the user to focus on the screen without needing to confirm their gestures manually. The motor is small and lightweight to prevent discomfort during prolonged use, and its placement ensures that the vibration is felt clearly without hindering hand motion.

The wires converge toward the main PCB, which is mounted near the arm and contains the microcontroller and power regulation circuitry. The PCB transmits sensor data wirelessly to a computer via Bluetooth, enabling real-time interaction without any external hardware or cameras. The glove is powered by a rechargeable 3.7 V Li-Po battery enclosed in a small, lightweight casing attached to the wrist section of the glove for balance and convenience.

2.3. Subsystems

2.3.1. Hand Motion Tracking Subsystem

Overview:

The Hand Motion Tracking Module is responsible for making the cursor move naturally as a function of hand movement. It uses IMUs to measure angular and linear acceleration, which are processed to produce relative cursor movements. This Module contains two ICM-20948 9-axis IMUs, mounted at strategic points on the glove to capture angular rate and acceleration. Each sensor is configured with gyroscope full-scale ±1000 dps and accelerometer full-scale ±4 g, operating at ≥400 Hz. The IMUs provide the raw data needed to compute smooth cursor deltas. Data is transmitted to the controller subsystem via SPI for low-latency performance. First, we define a navigation frame to be a 2D coordinate frame with one axis corresponding to the direction opposite of gravity and the other pointing towards the right (and perpendicular to the first axis) from first person view. Data is processed using an Extended Kalman Filter (EKF) [8] to estimate hand velocity (up/down and left/right motion in a predefined **navigation frame**) while rejecting drift via Zero-Velocity Updates (ZUPT) when the user presses the ZUPT button, which are transmitted via Bluetooth to the host.

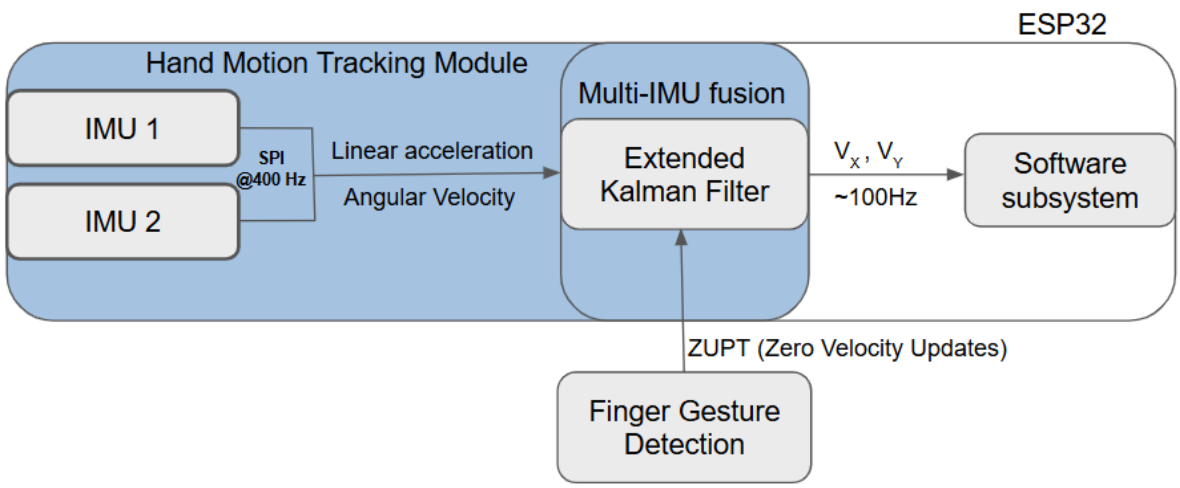

Figure 4: Overview of the Hand Motion Tracking Subsystem

State Definition for the EKF

The EKF state vector is designed to capture the minimal information needed for vertical motion tracking and will have the following elements:

1) **Orientation of the hand, represented as a unit quaternion:**

(orientation of the body/sensor frame with respect to a navigation frame fixed to the Earth). This is used to know which direction is "up" (vertical) regardless of how the user's hand is oriented.

2) **Vertical velocity** of the hand (in the navigation frame, along the global vertical axis). This is the quantity that will be mapped to cursor vertical movement.
3) **Horizontal velocity** of the hand (in the navigation frame). This is the quantity that will be mapped to cursor horizontal movement.
4) **Sensor bias terms** such as gyroscope bias and accelerometer bias.

Noise and Jitter Filtering:
The ICM-20948 IMUs have built-in configurable low-pass filters on both the accelerometer and gyroscope signal paths. We will configure these filters to remove vibrations and noise above the bandwidth of typical hand motions.

Requirements	Verification
• Update Rate: The motion tracking subsystem shall provide fused orientation/velocity updates at ≥ 100 Hz (10 ms or faster per update).	• Log time stamps in software for 100 consecutive cycles to ensure the average period is ~10 ms and variance is low.
• Drift Stability: When the hand is stationary, the cumulative vertical or horizontal position drift of the cursor shall not exceed 1 cm over 30 s.	• Drift Test: Place the glove on a stable surface (motionless) and initialize the system. Leave it stationary for at least 30 seconds. Measure the cursor movement on the screen (or internal recorded position) during this interval. It should remain within an equivalent of 1 cm of movement (for example, if 1 cm corresponds to X pixels based on screen DPI, the cursor should stay within that range). If the cursor moves slightly but stays under the threshold, requirement is met.
• Orientation Accuracy (Tilt Compensation): The system shall accurately compute the tilt of the hand so that vertical movement is correctly identified regardless of hand orientation (within a 5° tilt error during dynamic motion).	• Tilt Motion Test*:* Start with the glove upright, move it straight up/down, and confirm cursor moves purely vertically. Then tilt the hand by 30° in various directions (roll/pitch) and repeat an up/down motion. Verify that the cursor still moves predominantly vertically (with no more than a small horizontal drift, indicating tilt is correctly accounted for). Repeat the same for the other axis.
• ZUPT Activation: On a pinch gesture at the corresponding sensor, the system shall	• ZUPT Response Test*:* Induce a known velocity bias in the system by modifying the

reset the vertical velocity estimate to zero within one update cycle and eliminate any accumulated drift velocity.	the module's output by hand. Then press the ZUPT button on the palm.. Verify via logs that within the next 10 ms cycle the EKF's velocity state goes to ~0.
• Synchronization of Dual IMUs: The two IMU sensors' data shall be time-aligned to within 1 ms to ensure the fusion algorithm treats them as simultaneous readings.	• Synchronization Test: Shake the glove rapidly for a certain amount of time and record the raw signals from both IMUs. Compute the cross correlation between the two signals and find the maxima of the cross-correlation. Ensure that the maxima is between -1 and 1 ms.

2.3.2. Finger Gesture Detection Subsystem

Overview:
This subsystem allows the glove to detect which finger the thumb is touching by using a small magnet on the thumb and four Hall-effect sensors placed on the index, middle, ring, and little fingers. Each Hall sensor measures the magnetic field strength from the thumb magnet. When the thumb comes close to a finger, the sensor reading changes noticeably, and the software interprets that as a gesture.

Each finger is mapped to a specific mouse action:
1) Index finger: Left click
2) Middle finger: Right click
3) Ring finger: Scroll down
4) Little finger: Scroll up

Each Hall sensor continuously measures magnetic field strength and sends data to the ESP32 microcontroller. When the thumb approaches a finger, the field strength crosses a threshold value. The software subsystem detects this change and applies a short debounce delay to confirm the gesture before processing the gesture.

Requirements	**Verification**
• The subsystem must correctly identify which finger the thumb is touching and trigger the matching mouse action.	• The user will perform ten pinches with each finger. The recorded output will be compared to the expected action, and all gestures must match correctly.
• The subsystem must not falsely trigger	• The glove will be held in a relaxed

when the thumb is not near any finger.	position for one minute without any pinches. No gestures should be registered during this time.
• Each finger gesture must be recognized at least 95 % of the time in normal use.	• The user will perform 100 random gestures, including pinches and unrelated hand motions. The detection accuracy will be calculated and must meet or exceed 95%.
• Gestures on one finger must not interfere with other fingers' sensors.	• While pinching each finger, readings from all sensors will be logged. Only one finger should show a large change, and no other sensor should exceed its threshold.

2.3.3. Haptic Feedback Subsystem

The Haptic Feedback Subsystem is responsible for providing tactile responses to the user whenever an action, such as a click recognition, is successfully registered by the system. This subsystem enhances user interaction by giving immediate confirmation through vibration, removing the need for the user to constantly look at the screen to verify that their input has been detected. The inclusion of this subsystem significantly improves the glove's usability, making it more natural and responsive to human behavior.

The subsystem consists primarily of a vibration motor module integrated into the glove's fabric and controlled by the main microcontroller. When a valid gesture, such as a finger pinch or tap, is detected by the sensing subsystem, the signal is processed by the microcontroller, which then triggers the haptic driver circuit. This circuit momentarily powers the vibration motor using the same power source as the rest of the glove. The duration and intensity of the vibration are finely tuned in software to provide feedback that is perceptible but not distracting or uncomfortable.

To ensure consistent performance, the haptic motor is positioned in a part of the glove where vibration can be easily felt by the user, typically on the back of the hand. The subsystem's design also accounts for energy efficiency, as haptic activation is brief and only occurs when necessary to confirm user actions. The haptic motor's control signal is transmitted via a low-power PWM (Pulse Width Modulation) pin on the microcontroller, allowing smooth intensity control without additional hardware complexity.

Requirements	**Verification**

• When the Haptic Feedback Subsystem receives a signal from the microcontroller indicating that a pinch or gesture input has been recognized, it must activate the vibration motor within 100 milliseconds.	• Perform a finger pinch gesture and measure the time delay between gesture recognition and haptic vibration using an oscilloscope. • Confirm that the vibration motor activates within 100 ms of the input being recognized.
• When the Haptic Feedback Subsystem is inactive (no gestures detected), the vibration motor must remain off to conserve battery power.	• Observe the glove in an idle state with no user gestures for a duration of at least 60 seconds. • Verify through current measurement that no power is being supplied to the haptic motor driver during this period. • Confirm that no unintentional vibrations occur.

2.3.4. Software Subsystem

Overview

The Software Subsystem is responsible for the link between the EKF's instantaneous velocity outputs from the Hand Motion Tracking subsystem to visible mouse cursor movements on the host screen.The software subsystem acts as the central interface between the glove's sensor-fusion outputs and the computer's cursor control. It receives real-time velocity estimates from the Extended Kalman Filter (EKF) in the Hand Motion Tracking Subsystem and converts those values into smooth, proportional cursor movements transmitted wirelessly over Bluetooth. In addition, it processes finger gesture inputs (clicks and scrolls) and integrates all subsystems into a cohesive control loop recognized by the host computer as a standard Bluetooth mouse.

Data Flow and Processing:

The EKF on the ESP32-S3 microcontroller provides updated 2D velocity estimates, representing horizontal (right/left) and vertical (up/down) hand motion, at a rate of approximately 100 Hz. This frequency ensures responsiveness while maintaining computational efficiency on the embedded processor. At each cycle, the software receives the new velocity vector and the previous one from the prior cycle. Using a trapezoidal integration step, it computes the small relative displacement of the hand since the last update as the product of the timestep and the average instantaneous velocity of the current and previous time step.These displacements represent relative motion, similar to how an optical mouse tracks surface movement, and not absolute position, which prevents cumulative drift.

To achieve natural cursor movement, the software applies a scaling factor that converts the small physical displacement (in meters) to on-screen pixel displacement. This sensitivity is calibrated so that a comfortable hand motion range corresponds to the full screen width and height. For example, if 0.5 m of horizontal motion should move the cursor across a 1920 px-wide screen, the scale factor would be roughly 3840 px/m. The scaling constants can be tuned by the user until cursor speed feels natural. To prevent jitter, any very small movements (below a set threshold) are ignored, and the scaled displacement is smoothed slightly over time to ensure fluid motion.

The software also monitors for pinch gestures detected by the Finger Gesture Detection Subsystem. Each detected gesture triggers an event code (left click, right click, scroll up, scroll down), which the software combines with the motion data in a single Bluetooth report.

Motion constraints:
In addition to ZUPT, we also add motion constraints to counter drift. Our mapping structure from velocity to mouse deltas complements the usage of these constraints. Details are mentioned in the Tolerance Analysis section.

Bluetooth Transmission and Host Interaction

After computing the motion deltas (Δx, Δy) and processing any gesture events, the software packages them into a standard Bluetooth HID [9] (Human Interface Device) mouse report. Each report contains fields for:

1) Relative X and Y motion (signed 8-bit integers)
2) Button states (1 bit each for left, right, middle, scroll up, scroll down)

The ESP32-S3 is configured in Bluetooth Classic HID mode, which provides direct compatibility with Windows, macOS, and Linux computers without requiring special drivers. Alternatively, BLE HID over GATT can also be used for lower power consumption if desired. We hope to transmit the report at atleast 80Hz.

Requirements	**Verification**
• The subsystem must convert velocity into proportional cursor motion using correct scaling and integration.	• Test by moving the glove along a known distance (e.g., 0.5 m) and measuring corresponding on-screen movement. The displacement should match within ±10%.
• The Bluetooth HID connection must successfully transmit relative motion and button data to a standard host computer.	• Pair the glove with a Windows or Mac host and confirm the OS recognizes it as a “Bluetooth Mouse.” Move and pinch to verify cursor and click functions.

• The system must correctly map pinch gestures to mouse actions (left, right, scroll).	• Perform ten pinches for each finger gesture. Confirm all corresponding mouse actions execute correctly on the host.
• The Bluetooth connection must remain stable and maintain at least 80 Hz update rate.	• Measure Bluetooth report intervals using Putty. Verify average interval ~15 ms, no packet losses for ≥1 minute continuous operation.

2.3.5. Power Subsystem

Overview: The Power Subsystem provides stable and continuous power to all electronic components in the glove, including the IMUs, ESP32-S3 microcontroller, gesture sensors, and haptic feedback module. It is built around a 1200 mAh 3.7 V Li-Po rechargeable battery, which is chosen for its high energy density and low weight. The subsystem includes a TP4056-based charging and protection circuit, which provides overcharge, over-discharge, and short-circuit protection to ensure user safety and extend battery lifespan. A low-dropout (LDO) regulator ensures voltage stability within ±0.1 V even under transient current loads from the IMUs or haptic driver. To minimize electrical noise coupling between subsystems, a star-ground topology is used, isolating high-current haptic return paths from sensitive analog sensors. The overall subsystem must deliver consistent power for up to 4 hours of continuous operation without performance degradation.

Design Decisions and Justification

1. **Battery Type:** Li-Po was selected over Li-ion for its thin profile and lightweight form factor, improving comfort and flexibility for wearable applications.
2. **Capacity (1200 mAh):** Chosen to support up to 4 hours of use at an average current draw of ~250 mA (IMUs + MCU + haptics).
3. **Protection Circuit:** Integrated protection module prevents unsafe battery conditions, compliant with IEC 62133 safety standards.
4. **Voltage Regulation:** A low-noise LDO regulator (e.g., MCP1700 or AP2112) ensures the ESP32 and IMUs receive stable power even under fast load transitions from haptic activations.
5. **Thermal Safety:** The subsystem dissipates <0.5 W under continuous operation, keeping surface temperatures <40 °C for skin contact safety.

Requirement	**Verification Method**	**Verification Procedure**

1. Must supply **3.7 V ± 0.1 V** continuously under a 300 mA load.	Measured using a digital multimeter and oscilloscope.	Power the subsystem under simulated load equivalent to total system current draw and record output voltage stability.
2. Must tolerate **current transients up to 150 mA** from haptic motor activation without output droop > 0.1 V.	Oscilloscope transient response test.	Trigger haptic motor activation and monitor voltage response at output rail for undershoot or instability.
3. Must include **overcharge and short-circuit protection** for the Li-Po cell.	Functional test of charger circuit.	Manually short battery leads through 1 Ω resistor and verify cutoff within <100 ms; test automatic charge cutoff at 4.2 V.
4. Must provide **minimum 3.5 hours of operation** on full charge.	Endurance test.	Fully charge the battery, run the glove system continuously, and measure time until undervoltage protection activates.
5. Surface temperature must remain **<40 °C** during operation.	Thermal imaging / IR thermometer.	Operate the glove under normal load for 1 hour and verify external surface temperature below threshold.

Supporting Calculations

Subsystem	Typical Current (mA)	Peak (mA)	Power @ 3.7 V (mW)
ESP32-S3 MCU	80	240	888
Two IMUs	30	50	185
Four Hall Effect Sensors	8	15	56

Haptic Driver + ERM Motor	0	150	555
Total	**125 mA avg**	**460 mA peak**	**~1.7 W peak**

For a **1200 mAh battery**, expected runtime ≈
(1200 mAh)/(125 mA))≈ 9.6h under ideal conditions.
Accounting for regulator inefficiency and peak current draw, real-world runtime is ~4–5 hours, satisfying system requirements.

Risk Mitigation

1. **Voltage Drop:** Use low-ESR capacitors (≥47 µF) near IMUs and haptic drivers to absorb transient spikes.
2. **Thermal Protection:** Include a temperature sensor (NTC thermistor) in the battery circuit to monitor overheating.
3. **User Safety:** Full insulation between conductive lines and glove fabric to prevent skin contact with any live wire.

2.4. Tolerance Analysis

2.4.1. IMU Pose Drift Analysis

Integrating IMU acceleration twice (to get position) is notoriously unstable: even tiny accelerometer bias and tilt error (mis-projecting gravity) produce errors that grow quadratically with time

The glove's Extended Kalman Filter (EKF) runs at roughly 100 Hz and outputs 2-D hand velocities (v_x, v_y) representing right/left and up/down motion.
At each update we compute a small relative cursor movement:

$$\Delta x = 0.5 \times (v_x\_prev + v_x\_now) \times \Delta t$$
$$\Delta y = 0.5 \times (v_y\_prev + v_y\_now) \times \Delta t$$

with $\Delta t \approx 0.01$ s.
These displacements are scaled by a constant sensitivity (pixels per meter) and sent as relative Bluetooth-HID mouse deltas. Because the cursor motion is relative rather than absolute, accumulated integration error is automatically limited.

In addition to ZUPT, we use **motion constraints**:
When the IMU says the hand is still ($\|a\| \approx g$ and $\|\omega\|$ is small for ~0.5–1.0 s), the software forces the EKF velocity to zero (and gently trims bias). This caps any drift between "resets" so small errors never accumulate into visible cursor creep. We keep using the relative (velocity→delta) mapping, so the cursor only moves while the glove moves.

If we map position (i.e., double integration to get absolute hand pose) and try the same “reset when still” trick, we hit a fundamental problem:

1) With position mapping, a reset means changing the absolute origin. If you zero the pose while your hand is off to the side, the cursor’s notion of “center” jumps too.
2) Over time, these resets warp the workspace: parts of the screen become unreachable without re-centering because your physical hand range is limited but the screen is large.
3) Position mapping needs a true external reference (like a camera or marker) to anchor pose. Without that, you either drift forever or you recenter and break the mapping.

3. Cost Analysis & Schedule

Component	**Manufacturer / Model**	**Quantity**	**Unit Cost (USD)**	**Total (USD)**
ESP32-S3-WROOM-1 Microcontroller	Espressif	1	\$6.13	\$6.13
ICM-20948 9-Axis IMU	TDK InvenSense	2	\$14.95	\$29.90
AT42QT1010 Capacitive Touch IC	Microchip	3	\$0.77	\$2.31
DRV2605L Haptic Driver	Texas Instruments	1	\$3.25	\$3.35

ERM Coin Vibration Motor (C1030B002F)	Jinlong Machinery	1	$5.58	$5.58
Li-Po Battery (3.7 V, 1200 mAh)	Adafruit / SparkFun	1	$9.95	$9.95
TP4056 Charger + Protection Module	Generic	1	$3.	$3.00
LDO Regulator (e.g., AP2112)	Diodes Inc.	1	$0.22	$0.22
Flexible PCB / Wiring Harness	Custom	1	$11.95	$11.95
Conductive Glove Fabric + Materials	Generic	1	$20	$20
Miscellaneous (headers, connectors, resistors, LEDs)	—	—	$5.00	$5.00

Team Member	**Hourly Rate ($/hr)**	**Estimated Hours**	**2.5× Multiplier**	**Total ($)**
Khushi Kalra	$40	120	2.5	$12,000

Vihaansh Majithia	$40	120	2.5	$12,000
Vallabh Nadgir	$40	120	2.5	$12,000
Total Labor Cost	—	—	—	**$36,000**

Schedule

Week	Task	Person
09/22	Proposal	Everyone
09/29	Proposal adjustment + presentation prep	Everyone
10/06	Breadboard demo: bluetooth connection, hall effect sensors	Everyone
10/13	Dsign document	Everyone
10/20	Plan for breadboard demo #2	Everyone
10/27	Breadboard demo #2 + PCB designing	Everyone
11/03	PCB Designing	Everyone
11/10	Assembling the PCB and breadboard	Everyone
11/17	Getting everything on the glove	Everyone
11/24	Testing	Everyone
12/01	Last minute tweaks + changes	Everyone

4. Ethics and Safety

Electrical and Battery Safety

The glove operates at 3.7 V DC, which is below hazardous voltage levels, but its Li-Po battery poses risks of overheating, overcharging, or puncture if mishandled. To ensure user and developer safety, the following safeguards are implemented:

1. A TP4056-based protection circuit will provide overcharge, over-discharge, and short-circuit protection to prevent unsafe battery operation.
2. The charging input is limited to 5 V, 500 mA via a USB interface, preventing excessive charging currents.
3. All wires and PCBs are fully insulated and routed within the glove's fabric layers to eliminate skin contact with conductors.
4. A fuse and reverse-polarity protection diode are included in the power line to protect against electrical faults.
5. During bench testing and charging, batteries will be handled on non-flammable surfaces with supervision and UL-certified charging adapters.

These practices align with IEEE Code of Ethics #1, prioritizing public safety, and ECE lab safety standards for battery-operated devices.

Sensor and Magnetic Field Safety

The system uses four Hall Effect sensors to detect finger pinches instead of capacitive touch sensors. Hall sensors operate at extremely low magnetic field strengths (<100 mT), which are safe for human exposure.

1. The embedded neodymium magnets are small (≤3 mm diameter) and enclosed within fabric pockets to prevent skin contact or ingestion risk.
2. Magnetic field levels are far below any exposure limits defined by ICNIRP and IEEE C95.1 standards for wearable electronics.
3. All magnets are firmly bonded or sewn into the glove to prevent detachment during use.

This ensures gesture detection is achieved safely without creating mechanical or electromagnetic hazards.

Wearability and Ergonomics

Since this is a body-worn electronic system, comfort and safety during long-term use are paramount:

1. The glove is designed with breathable, stretchable fabric to reduce sweat accumulation and improve comfort.
2. Weight distribution is optimized so the battery and electronics are placed on the wrist rather than the fingers, preventing motion restriction.
3. PCB edges and sensor mounts are rounded and insulated to avoid irritation, poking, or skin abrasion.
4. The glove remains lightweight (<100 g), allowing extended use without strain.

Proper ergonomics reduce the risk of repetitive strain and ensure inclusive usability for various hand sizes.

Data Privacy and Security

The glove communicates via Bluetooth HID, transmitting only motion and gesture data. No personally identifiable or biometric information.

1. No data is stored locally or sent to cloud servers.
2. Device pairing requires standard Bluetooth authentication to prevent unauthorized connections.
3. All transmissions are limited to necessary input signals, ensuring user privacy and data minimization.

This adheres to IEEE Code of Ethics #7, protecting user confidentiality and respecting privacy boundaries.

Environmental and Sustainability Considerations

To reduce environmental impact and electronic waste:

1. All electronic components are RoHS-compliant, free from hazardous substances such as lead, mercury, and cadmium.
2. The Li-Po battery is rechargeable, significantly lowering waste compared to disposable batteries.
3. End-of-life disposal instructions will be provided, including proper Li-Po recycling procedures at certified e-waste centers.
4. Whenever possible, components are sourced from manufacturers with sustainability certifications, and reused during prototyping to minimize material waste.

Ethical Compliance Summary

This project adheres to the IEEE Code of Ethics through the following principles:

1. Prioritizing safety and welfare of users by implementing multiple electrical and mechanical safeguards.

2. Avoiding environmental harm by using recyclable, RoHS-compliant materials.
3. Ensuring honesty and transparency by transmitting only necessary motion and gesture data.
4. Designing inclusively and responsibly by considering ergonomics, accessibility, and user comfort.
5. Promoting responsible development through careful testing, documentation, and adherence to ECE laboratory safety standards.

By integrating these ethical and safety considerations, the Wearable Glove Cursor Controller provides a safe, sustainable, and ethically responsible human–computer interaction solution.

5. REFERENCES

[1] TDK InvenSense, *ICM-20948 9-Axis MotionTracking Device Datasheet*, rev. 1.3, 2019. [Online]. Available: https://invensense.tdk.com/products/motion-tracking/9-axis/icm-20948/

[2] Espressif Systems, *ESP32-S3 Series Datasheet*, v2.0, 2023. [Online]. Available: https://www.espressif.com/en/support/documents/technical-documents

[3] Espressif Systems, *ESP32-S3-WROOM-1/-1U Module Datasheet*, 2023. [Online]. Available: https://www.espressif.com/en/support/documents/technical-documents

[4] Texas Instruments, *DRV2605L Haptic Driver for ERM/LRA Datasheet*, rev. C, 2016. [Online]. Available: https://www.ti.com/product/DRV2605L

[5] Intertek, “IEC 62133 Safety Testing for Lithium-Ion Batteries Overview,” 2021. [Online]. Available: https://www.intertek.com/batteries/iec-62133/

[6] IEC, *IEC 62133-2:2017+A1:2021 Secondary lithium cells and batteries for portable applications*, 2021. [Online]. Available: https://webstore.iec.ch/

[7] Jinlong Machinery & Electronics, *10-mm Coin ERM Vibration Motor C10xx Series Specifications*, 2020. [Online]. Available: https://www.nfpmotor.com/coin-vibration-motor/

[8] Welch, G., and Bishop, G. "An Introduction to the Kalman Filter." TR 95-041, University of North Carolina at Chapel Hill, Department of Computer Science, 2006. Available: https://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf

[9] Bluetooth SIG. "Human Interface Device Profile (HID) 1.2 Specification." Bluetooth SIG Standard, Adopted Nov. 22, 2022.

[10] Espressif Systems, *Bluetooth API ESP-IDF Programming Guide (ESP32)*, v5.5.1, 2025. [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/bluetooth/

[11] Espressif Systems, *Bluetooth Low Energy (BLE) API Reference (ESP32)*, v5.5.1, 2025. [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/bluetooth/bt_le.html