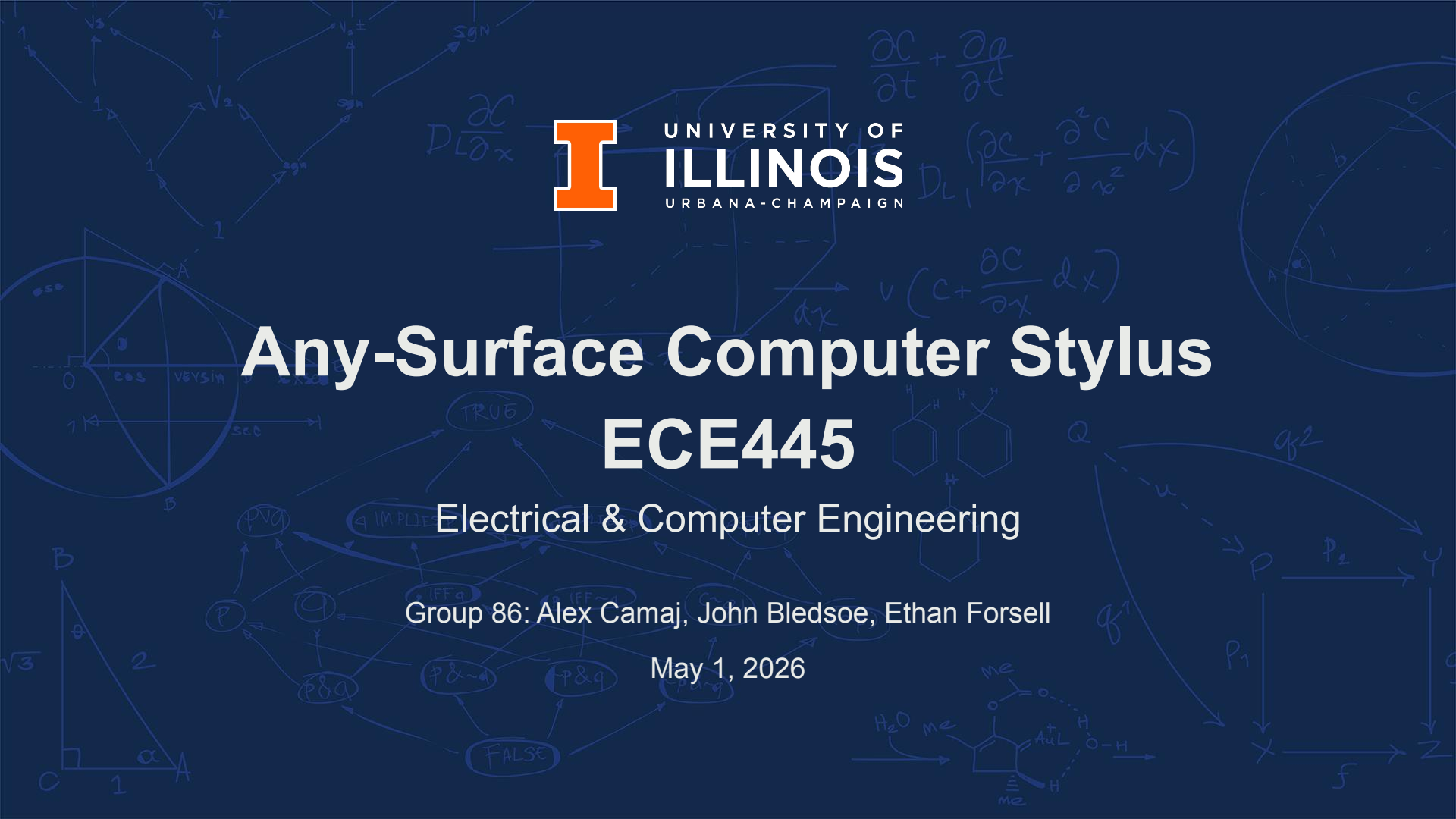


UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Any-Surface Computer Stylus

ECE445

Electrical & Computer Engineering

Group 86: Alex Camaj, John Bledsoe, Ethan Forsell

May 1, 2026

The Problem with digital drawing

Usability

- Drawing ability of standard mice are limited
- Point and draw workflows are outdated
- Lack of natural feel

The Problem with digital drawing

Accessibility

- Drawing tablets are overkill for many users
- Nothing universal on the market.
- Existing peripherals require complex software or proprietary screens

The Stylus Product

Core Design

- Ergonomic pen-shaped stylus that works on any surface.
- Easy to use, writes like a pen

Portability

- USB-A plug and play; no drivers needed
- Lightweight Design: under 30g

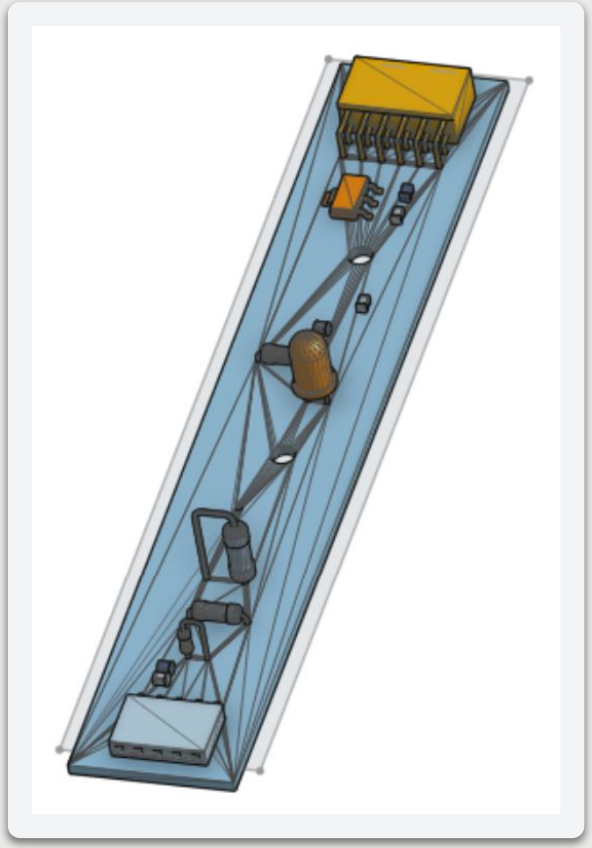
The Stylus Product

Tactile Controls

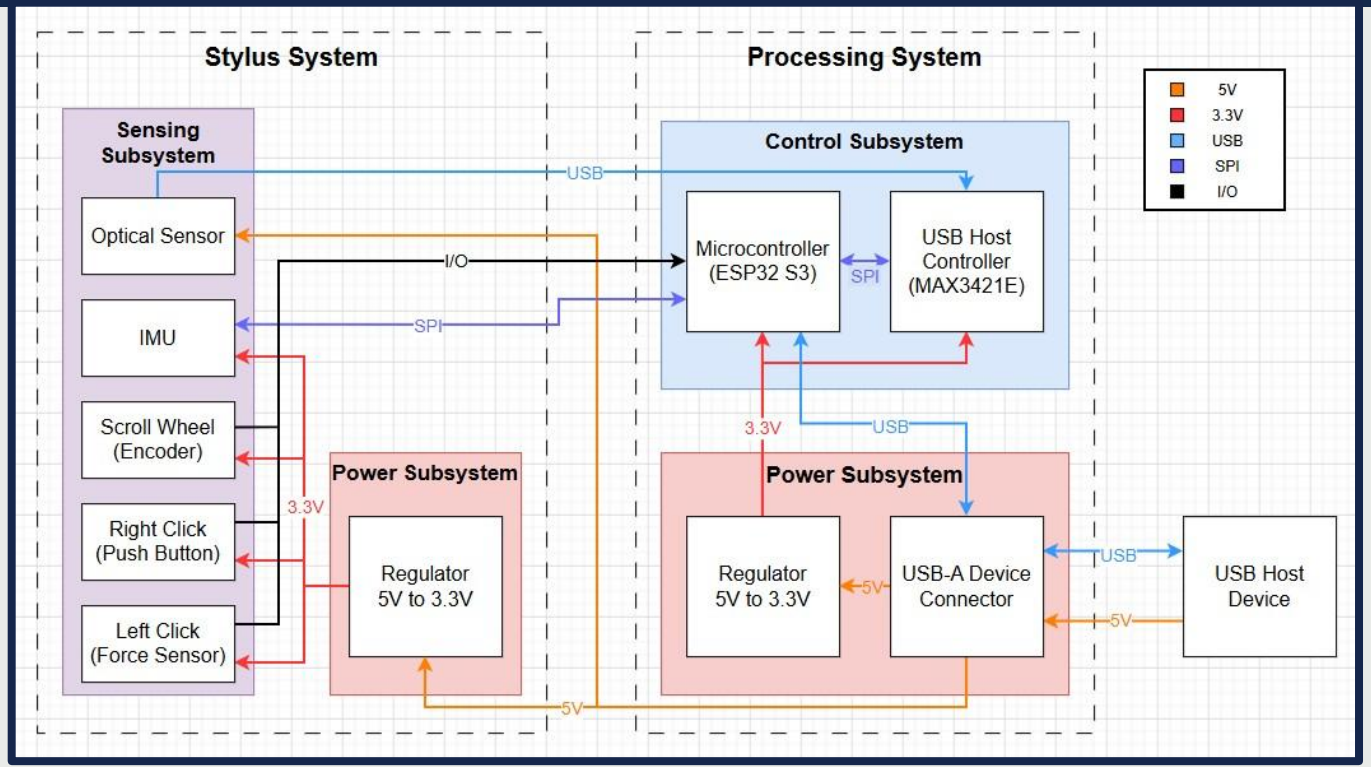
- Force sensitive tip for pressure-based left click.
- Side button for right clicks and scroll wheel.

Precision Tracking

- USB-A plug and play; no drivers needed
- Lightweight Design: under 30g

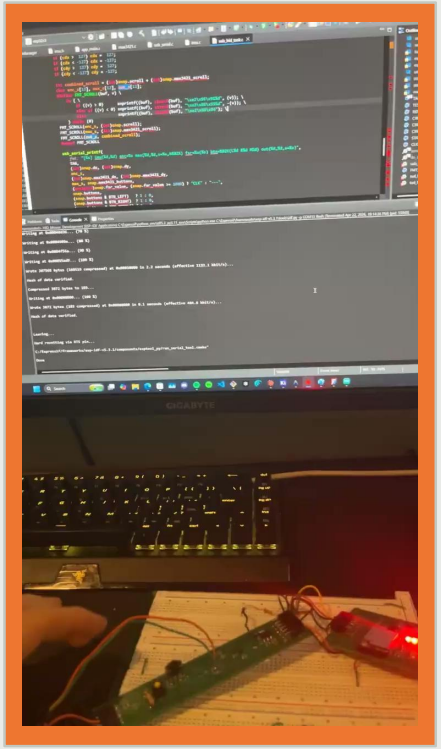


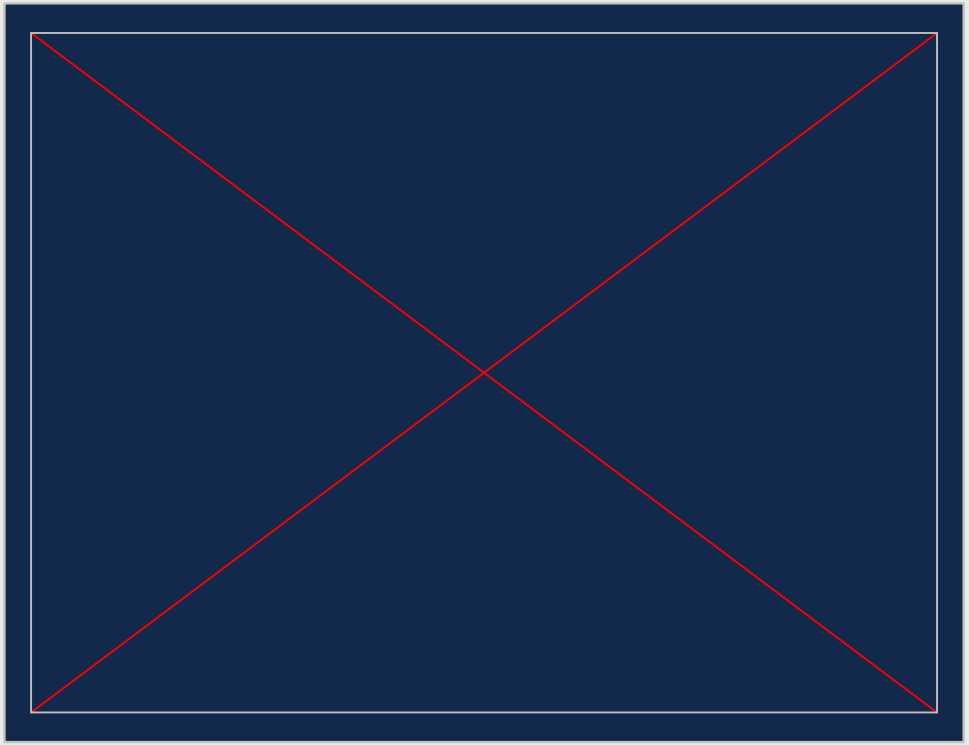
Block Diagram



The system consists of two separate PCBs: the stylus houses the sensors and the processing system contains the microprocessor.

IMU vs Optical Sensor





There are 2 main ways of control through a IMU tilt or optical sensor.

Software Architecture: Dual Core

Core 1: Sensor Task

- Poll IMU, MAX3421E, & FSR
- Read Button & Encoder
- **Pack & Push to Queue:**

```
report.buttons = btns;
report.dx = (int16_t)acc_dx;
report.dy = (int16_t)acc_dy;
report.scroll = 0;
report.max3421_dx = (int16_t)acc_max_dx;
report.max3421_dy = (int16_t)acc_max_dy;
```

Delay for 8ms timing loop.

Queue

Core 0: USB HID Task

- **Runs every 8ms:** Precise report interval.
- **Queue Management:** Grabs latest sensor data.
- **Logic:** Decides which sensor output to utilize.
- **Reporting:** Sends data to host device.

HID & CDC

Host Device

COM Port

Mouse Device

System interprets inputs as standard HID mouse movements and serial data.

Software Architecture: State Machine

USB HID Implementation

Uses **TinyUSB stack** in ESP-IDF. Reports at **125 Hz** for smoothness.

HID Report (4 bytes):

- Left Click: FSR pen-down
- Right Click: Push button
- X/Y: State machine outputs
- Scroll: Accumulator value

State Framework

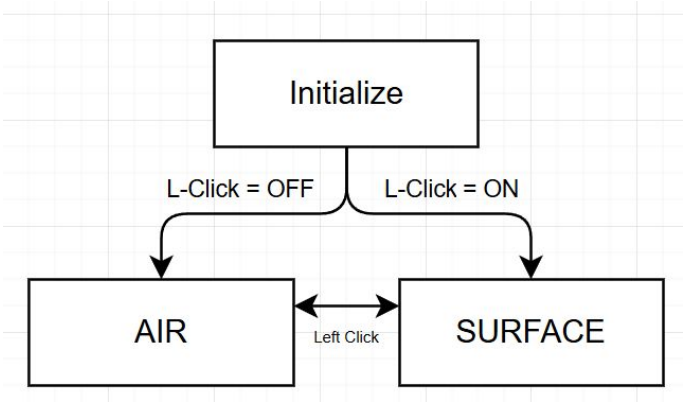
INITIALIZE: Establishing host-device link.

SURFACE: Optical sensor controls cursor.

AIR: IMU takes control when FSR is below threshold.

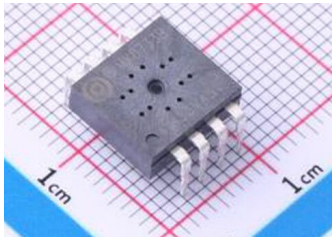
Timer Interrupt: Every 8ms for timing restraints.

System State Flow




```
[usb_hid_task] imu(3,-1) enc= max(0,0,s=-, b00)
fsr=4012(---) btn=00(L0 R0 M0) out(3,-1,s=-)
[sensor] conn: imu= ok max3421= ok
```

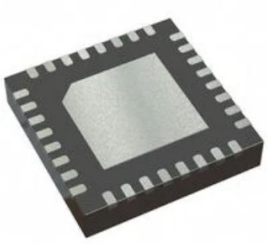
Part Selection



Optical Sensor

MX8733B - SoC

Provides Differntial signals for mouse movement and acts as LED driver. Lens provided. 1000DPI.



USB Interfacing

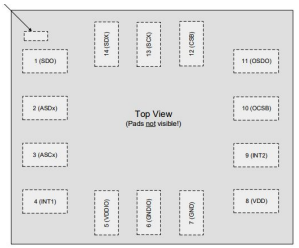
MAX3421

Converts between SPI and differential signals for USB applications

Microcontroller

ESP32-S3

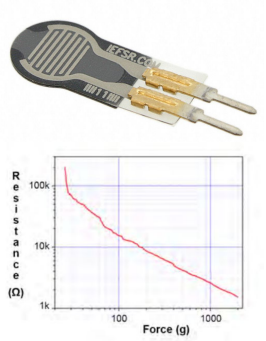
Multi-Core processor with sufficient memory, Internal pull-ups, analog and digital inputs



IMU (Inertial Measurement Unit)

BMI160

Compact 6-axis - Tilt and Acceleration, SPI communication

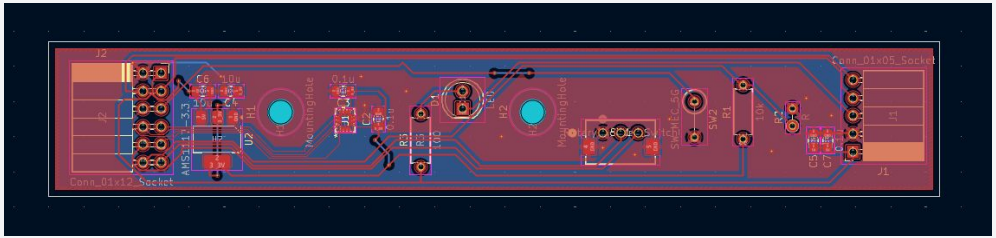


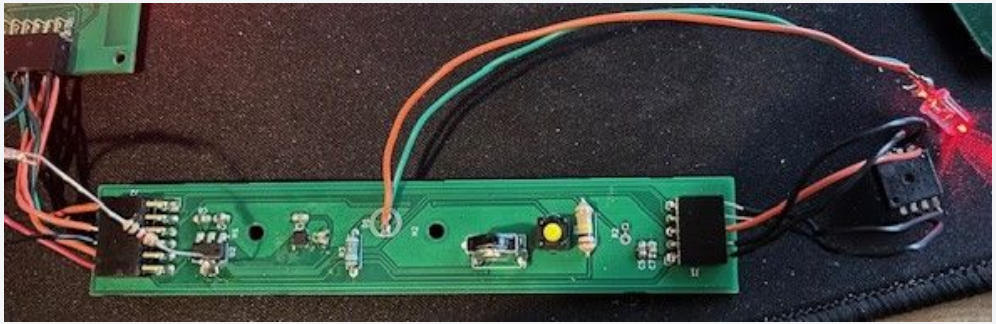
Force Sensitive Resistor

34-00004

Compact, simple, most sensitive between 0g-100g

Project Build - Stylus



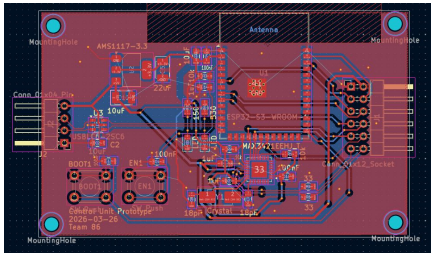


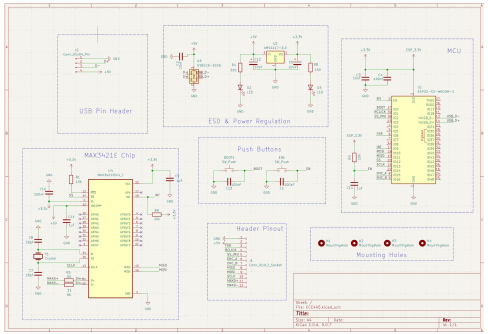
Stylus PCB

6-AXIS IMU – Optical Sensor – FSR – Rotary Encoder – Button – LED – Regulator – Capacitors

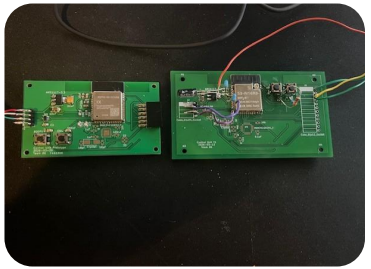
Project Build - Control Box

Design & Schematics





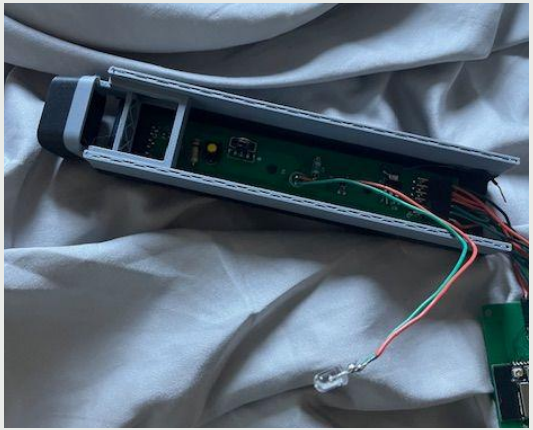
Physical Assembly

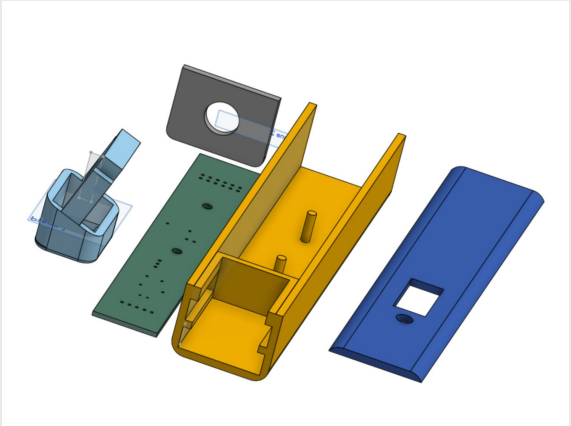


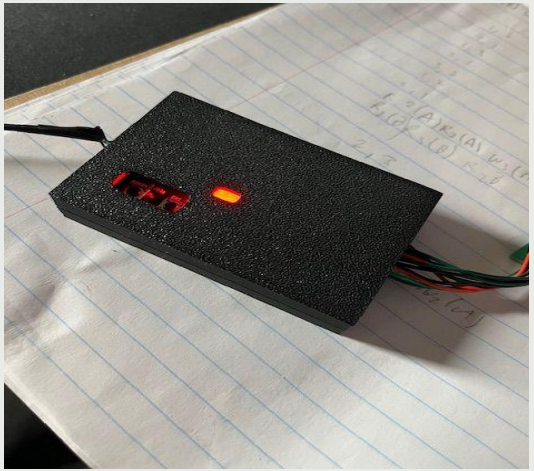
Control PCB Components

ESP32-S3 – MAX3421 – Regulator – Crystal Oscillator – Capacitors – Processor Buttons – LEDs – USB Connector

Project Build - Integration & Housing







Stylus Tip – Stylus Body – Control Housing

Requirements & Verification

HIGH LEVEL REQUIREMENTS

1. Writing Speed

Within 25% of regular pen speed.

2. Pen Writing Range

Track movement when lifted up to 0.5 inches.

3. Ergonomics

Easy to manipulate; mass < 30g.

SENSING SUBSYSTEM

1. X/Y Accuracy (Error < 5%)

±5mm match for 100mm travel.

2. Lift Detection (3.0mm ± 0.5mm)

Verify zero displacement at 3mm.

3. IMU Tilt Control (> 3mm)

45° tilt match cursor direction.

4. Click Latency (< 100ms)

GPIO to USB event timing.

5. Scroll Wheel Fidelity

1:1 physical click to host event.

6. Pressure Sensitivity (FSR)

Stable voltage during drag.

CONTROL SUBSYSTEM

1. USB Enumeration (< 5s)

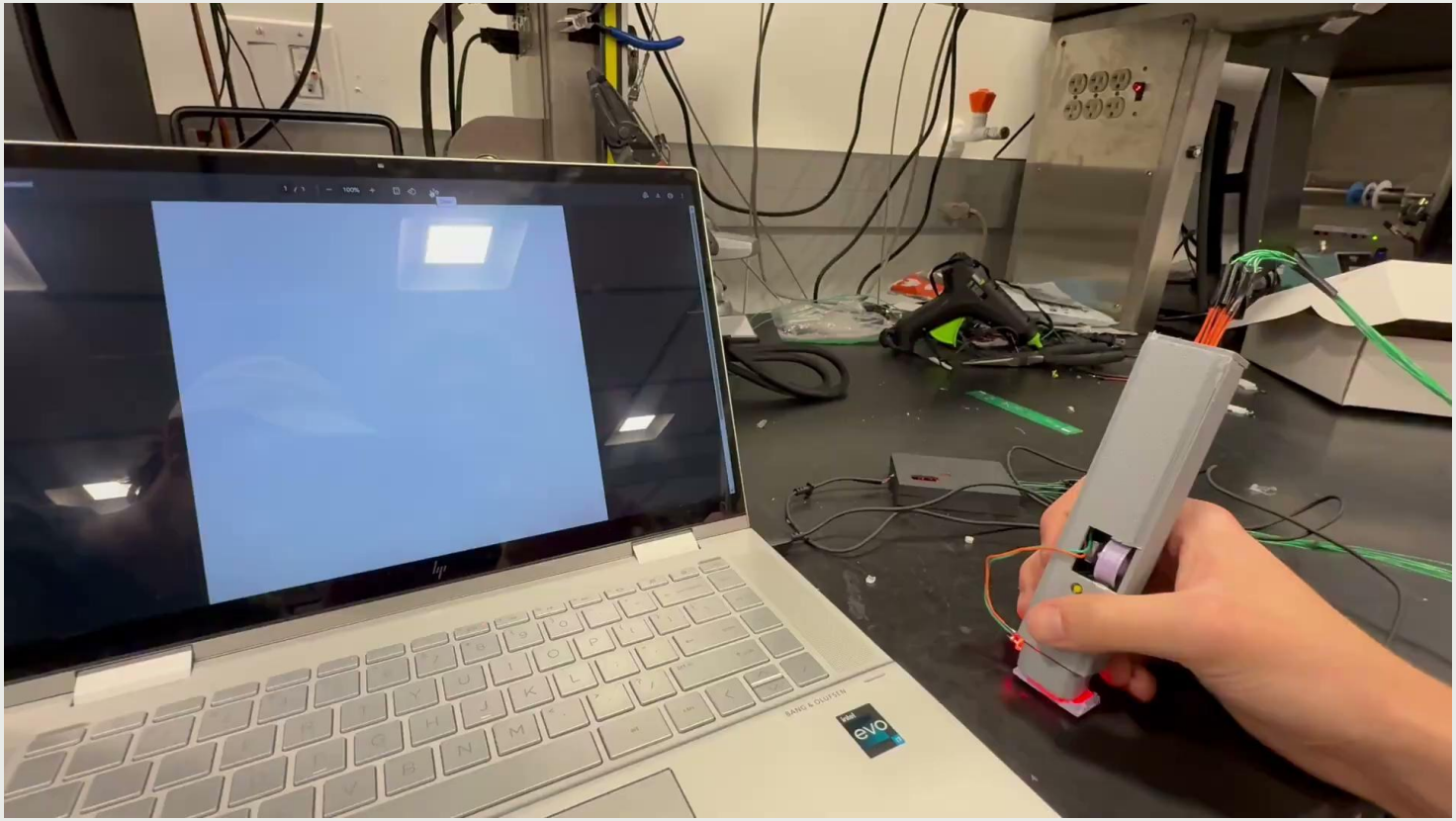
Plug-in to Manager appearance.

2. Seamless Handover

Optical to IMU switch at 3mm.

3. Data Fidelity (< 5% Loss)

Minimal loss over 60s log.



Testing the Stylus Design. Demonstrating IMU control, a click, optical sensor control, and drawing. Not demonstrated: scroll and right click functionality.

Project Build - Bill of Materials

Part Name	Qty	Cost (total)	Description
ESP32-S3	1	$6.00	Microprocessor
MAX3421	1	$12.29	USB Peripheral
ECS-120	1	$0.39	Crystal Oscillator 12MHz
4518-USBLC6	1	$0.39	Diode for USB
USBC-31	1	$0.46	USB Port
SS14	1	$0.44	Schottky Diode
T491B106K010AT	1	$0.66	Tantalum Cap 10uF
BMI160	1	$4.17	IMU
34-00004 FSR	1	$7.12	FSR
MX8733B	1	$1.00	Optical Sensor + Lens
EC10E	1	$0.92	Rotary Encoder
PTS636	3	$0.63	Button
N/A	3	$0.22	LED
AMS1117	2	$0.20	Regulator 5 to 3.3v LDO
N/A	15	$0.20	Capacitors

Not Included:

Cost of PCBs
Cost of Plastic
Cost of Wire

Improving the Product

If we restarted today:

- Design a smaller, more compact stylus PCB
- Add orientation tracking
- Redesign housing and integration for smaller size and greater reliability.
- Continue to tune and calibrate sensors, add state to read optical input without left-click

Ideas for commercialization:

- Design a tiny custom lens to make the tip of the pen smaller
- Swap wired connection for Bluetooth
- Restructure case and button quality
- Create a custom optical sensor with increased range.

Thank You!

Requirements & Verification

HIGH LEVEL REQUIREMENTS

1. Writing Speed:
We want the pen to write as fast as possible. Within 25% of the writing speed of a regular pen.

2. Pen Writing Range:
The pen must be able to still track movement when lifted above 0.5 inches from the surface.

3. Ergonomics and Easy of Use:
The pen should be easy to manipulate and be less than 30g in mass.

SENSING SUBSYSTEM

1. X/Y Accuracy (Error < 5%)
Verify 100mm stylus travel matches ESP32 counts within ±5mm.

2. Lift Detection (3.0mm ± 0.5mm)
Use 3mm spacer to verify zero displacement output on serial monitor.

3. IMU Tilt Control (> 3mm)
Tilt pen 45° in air; confirm cursor direction matches tilt angle.

4. Click Latency (< 100ms)
Record GPIO edge to USB event for 10 trials.

5. Scroll Wheel Fidelity
Confirm exactly 1 scroll event per physical click on host monitor.

6. Pressure Sensitivity (FSR)
Constant pressure drag; verify uninterrupted left-click voltage.

CONTROL SUBSYSTEM

1. USB Enumeration (< 5s)
Measure time from plug-in to device appearance in Manager.

2. Seamless Handover
Verify cursor data switches from Optical to IMU at 3mm lift.

3. Data Fidelity (< 5% Loss)
Log 60s of sensor data; verify packet loss is minimal via analyzer.