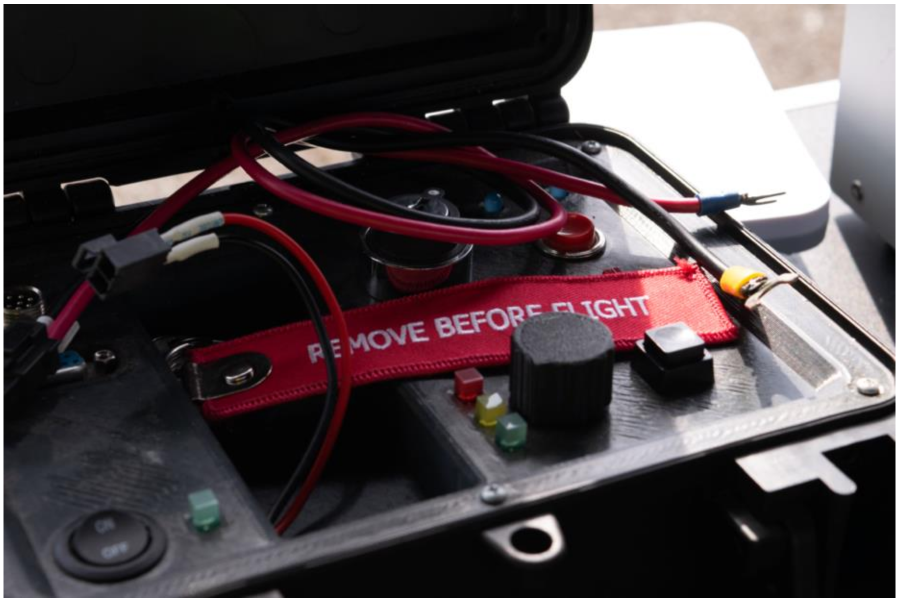
PROJECT I. – LAUNCHBOX STATUS MEASUREMENT

Introduction

This project primarily serves as a test of your abilities—most of you will find it a formality. However, it's not just a "shelf" project; we'll be using it regularly as a team. You may already be familiar with the ESP platform, which is essentially a more powerful Arduino with built-in WiFi and Bluetooth. While you won't be working with ESPs long-term, they act as training wheels for now, so we can ease you into the process without overwhelming you with the STM32 IDE. Meanwhile, our fellow hardware engineers and novices are working on a custom PCB, which you'll eventually use in future phases.

This document outlines both the project objectives (what to do) and the project execution (how to do it). The execution section includes a number of linked tutorials, so don't worry if you're unfamiliar with certain topics—you'll have plenty of opportunities to learn. If you get stuck, feel free to reach out to me via Teams or email. I'll be happy to help. Also, support one another as much as possible; the more members we have contributing, the better. Be sure to schedule weekly meetings to organize tasks and involve senior team members for guidance.



Requirements

The goal of this project is to develop a system that collects data from our launch box and wirelessly transmits it for use in other projects. The data to be collected includes:

- Firing status (red, yellow, green) with a timestamp
- Key position
- Battery voltage
- Launch button status

Hardware Requirements:

- ESP32
- Launch case: //Refer to the schematic
- Interconnection materials, resistors

Recomended project execution

1. **IDE setup:** Your first task is to set up an IDE that works with ESP32 chips. You can use either VSCode with PlatformIO or Arduino IDE. I strongly recommend using VSCode as it will be more useful for you in the long run. Start by flashing the built-in LED. Coordinate with your team since projects can't be directly backported from PlatformIO to Arduino.
2. **Git Setup:** The second task is to set up a Git account and create a repository for your project. We'll conduct a workshop to guide you through this.
3. **Understanding the Launch Case:** Your third task is to understand the wiring diagram of the briefcase. Take a multimeter and the briefcase to the office, and under supervision, measure the values within the case. Identify where to measure each data point. DO NOT DISCONNECT OR MODIFY ANYTHING IN THE CASE!
4. **Connecting the ESP32:** The fourth task is to connect your ESP32 to the case and start reading the values. To measure the battery, you'll need to use the ADC (Analog to Digital Converter). Be cautious—the ESP32 pins can only handle voltages within the 0–3.3V range, so you'll need to use a resistive divider to bring the voltage down. Ensure nothing burns up!
5. **Data Connectivity:** Your fifth task is to establish connectivity with other projects. You'll need to send the collected data in a suitable format (such as .csv or JSON) to your colleagues. Bluetooth will be used for this data transmission.

6. **Documentation:** The last task is to create documentation for the entire project. This should be a maximum of 3 A4 pages in English. I suggest you start drafting documentation during your meetings to keep things on track.

Literature

https://randomnerdtutorials.com/vs-code-platformio-ide-esp32-esp8266-arduino/

https://randomnerdtutorials.com/installing-esp32-arduino-ide-2-0/

https://randomnerdtutorials.com/esp32-digital-inputs-outputs-arduino

https://randomnerdtutorials.com/esp32-adc-analog-read-arduino-ide

https://ohmslawcalculator.com/voltage-divider-calculator

https://randomnerdtutorials.com/esp32-bluetooth-low-energy-ble-arduino-ide

Launchbox schematic - if the link doesnt work, message me on Teams

All other documentation can be found here