2025 TEEP Progress Report Week-22

The Battery readings :

Charing curve with ESP32 and INA219.data

This project is a **Battery Management System (BMS)** built using an **ESP32 microcontroller**, an **INA219 current and voltage sensor**, and a **Li-ion battery**, with a **Flask-based web dashboard** for real-time monitoring. The ESP32 acts as the central controller, continuously collecting battery parameters such as voltage, current, and power from the INA219 sensor, which communicates with the ESP32 using the I^2C protocol. Using these measurements, the system can estimate battery status such as power consumption and overall battery condition, helping to prevent overuse and inefficient energy handling. The ESP32 then sends this data wirelessly (via Wi-Fi) to a Flask server running on a computer or local network. Flask is used to create a lightweight web application that displays the battery data in a clear and user-friendly dashboard, allowing users to monitor battery performance in real time through any web browser. This project demonstrates the integration of embedded systems, sensor data acquisition, wireless communication, and web development, making it a practical example of an IoT-based energy monitoring solution.

you can check following details with :

Battery monitoring system using flask server.pdf

Click the below link to watch the output video

- https://youtu.be/0CwqLIPdDpg

Code Files

Click bellow link and get code files :

- Battery monitoring system using flask server code files Github

click the below link to see the details monitor using ESP32

Charing curve with ESP32 and INA219.data

Battery Readings

This file includes **manual measurement and verification** of battery parameters during charging using a **TP4056 Li-ion charging module**. While charging, voltage and current values are measured **without using the ESP32**, relying instead on a **multimeter** to directly read and verify the battery and charging characteristics. This step helps validate the accuracy of the sensor-based readings and provides a better understanding of the charging behavior of the Li-ion battery. Overall, the project combines embedded hardware, sensor validation, safe battery charging, and web-based visualization, making it a comprehensive and practical battery monitoring and management solution.

Reading -1

Time (min)	Voltage (V)
0	3.12
5	3.45
10	3.62
17	3.74
21	3.83
25	3.90
30	3.98
40	4.07
46	4.11
55	4.17
60	4.19
65	4.21

click the bellow link to check the readings

[Battery Charging Readings]

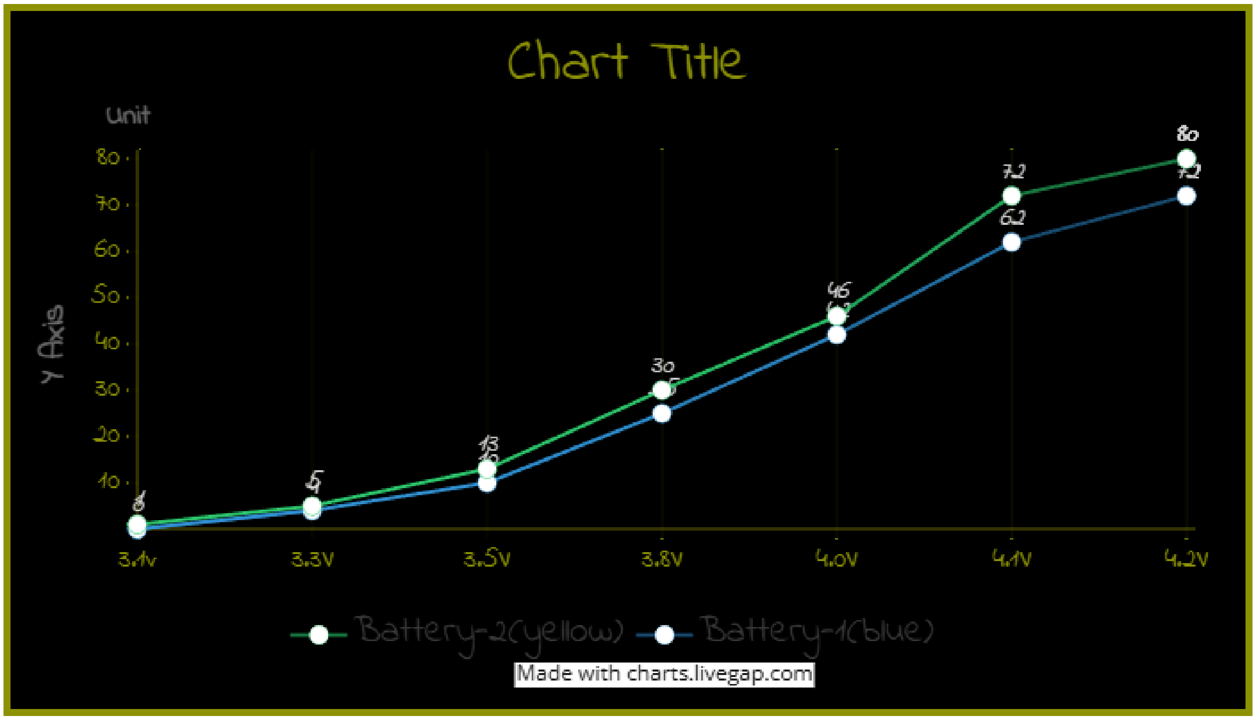
Reading- 2

Reading- 2

Intially battery at 3.08

Time (min)	Voltage (V)
0	3.08
5	3.39
15	3.55
20	3.68
28	3.79
35	3.88
45	3.95
59	4.01
65	4.05
79	4.10
86	4.22

The Graph based on readings :



Battery Monitoring Section with esp32:

In this project, battery monitoring is performed using an ESP32 microcontroller and an INA219 current and voltage sensor. The INA219 measures the battery voltage and current by connecting to the battery terminals and communicates these values to the ESP32 via the I²C interface. The ESP32 processes this data and transmits it wirelessly to a Flask-based web dashboard, where the battery status can be viewed in real time. This setup allows continuous monitoring of battery performance during system operation and discharge.

Battery Charging Section only TP4056:

Battery charging is handled separately using a TP4056 module that is directly connected to the single Li-ion battery. The TP4056 manages safe charging using an external 5 V supply, while voltage and current values are measured manually with a multimeter during charging. The ESP32 and INA219 are not involved in the charging process, which simplifies the circuit and ensures reliable and safe battery charging.

Using Tp4056 for series connections (Not recommonded)

Using **three TP4056 modules in series is not suitable for Li-ion charging**, and this is the main reason the boards were damaged. The TP4056 is **designed only for charging a single 3.7 V Li-ion cell**, with a maximum charge voltage of **4.2 V**. When multiple TP4056 modules are connected in series, the voltage levels become incorrect, causing **overvoltage, reverse current flow, or grounding conflicts**, which can permanently damage the charging ICs and protection circuits on the boards.

For multi-cell (series) battery systems, a **dedicated multi-cell BMS or balance charger** must be used instead of multiple TP4056 modules. These systems are specifically designed to handle **cell balancing, higher voltages, and safe charging of series-connected batteries**. In your project, using a **single TP4056 with one Li-ion battery** is the correct and safe approach, which is why your current setup works reliably without damaging the hardware.

Parallel Batteries with a Single TP4056 (Recommended)

Using **parallel-connected Li-ion batteries** with one TP4056 charger is generally acceptable **if the batteries are identical**. All cells must have the **same nominal voltage (3.7 V)**, **same capacity**, **same chemistry**, and ideally be of the **same brand and age**. Before connecting them in parallel, their voltages must be **nearly equal**; otherwise, a large equalization current can flow between cells, which may cause overheating or damage.

When connected correctly, parallel batteries behave like a **single larger-capacity battery** while maintaining the same voltage. The TP4056 will charge the battery pack to **4.2 V**, but the **charging current is shared among the cells**. This means charging will take longer, and the TP4056 charge current (set by its resistor) must be within safe limits for the combined battery configuration. Using TP4056 modules with **built-in protection (DW01 + MOSFET)** is strongly recommended.

However, for better safety and longevity, it is good practice to include **individual protection circuits or a parallel BMS**, especially when more than two cells are used. While parallel charging is much safer than series charging with TP4056, improper matching of cells or lack of protection can still lead to battery stress or failure.