Accurate SoC Estimation of Lithium Ion Battery

By
Sanjog Awasthi (21BEI0063)
Adit Tiwari (21BEI0121)
Aman Singh (21BEI0126)

Guide:
Dr. M.Geetha
Professor Grade -1
SELECT

Contents

Introduction

- The use of lithium-ion batteries is growing fast in electric vehicles, solar energy systems, and portable devices like phones and laptops.
- To ensure their **safe and efficient use**, we need **smart Battery Management Systems (BMS)**.
- One of the **most important features** of a BMS is estimating the **State of Charge (SoC)** — it tells us how much battery power is left.
- Accurate SoC estimation helps:
- **Prevent battery damage** (from overcharging or deep discharge)
- **Improve energy use** in power-limited devices
- **Extend battery life** and reduce electronic waste
- Traditional methods like using just voltage are not reliable due to battery behavior changes with load, temperature, and age.

Problem Definition

- As batteries are widely used in EVs, solar systems, gadgets, and medical devices, **accurate SoC (State of Charge) estimation** is crucial for safety and performance.
- **Current challenges in SoC estimation:**
- **Voltage-based method** is simple but becomes **inaccurate under load** due to voltage drops.
- **Coulomb counting** is real-time but can **drift over time** and needs regular calibration.
- **Lack of IoT** in many systems means no **remote monitoring or alerts**, reducing safety and reliability.
- There’s a **need for a smart, low-cost, and accurate SoC estimation system** that:
- Uses **multiple methods** to improve reliability
- Runs on **affordable microcontrollers** like Arduino/ESP32
- Supports **IoT-based monitoring** through smartphones or the cloud

Literature Survey

Piller, S., Perrin, M., & Jossen, A. "Methods for state-of-charge determination and their applications." *Journal of Power Sources*, 96(1), 113–120, 2001.

Key Insights:

- Estimates SoC based on terminal voltage mapped to an SoC-voltage curve.
- Simple, low-cost, and easy to implement on microcontrollers.
- Performance degrades under dynamic loads due to voltage drops.
- Highly sensitive to temperature, cell aging, and internal resistance.

Limitations:

- Accurate only under no-load or stable conditions.
- Poor standalone reliability under real-world usage.

Literature Survey

Kim, I.S. "The novel state of charge estimation method for lithium battery using sliding mode observer." *Journal of Power Sources*, 163(1), 584–590, 2006.

Key Insights:

- Integrates current flow over time to calculate net charge (in/out of battery).
- Suitable for real-time estimation.
- Commonly used in embedded battery systems.

Limitations:

- Long-term drift due to sensor inaccuracies, noise, and offset errors.
- Needs periodic calibration or correction via hybrid methods.

Literature Survey

He, H., Xiong, R., & Fan, J. “Evaluation of Lithium-Ion Battery Equivalent Circuit Models for State of Charge Estimation by an Experimental Approach.” *Energies*, 4(4), 582–598, 2011.

Key Insights:

- Uses model-based recursive estimation combining voltage/current with system dynamics.
- Dynamically corrects SoC with high robustness to noise and non-linearity.
- Performs well under fluctuating loads and environmental variability.

Limitations:

- More computationally intensive.
- Requires precise modeling of battery behavior.

Literature Survey

Noh, C., Park, K., & Choi, J. "Development of Battery Monitoring System Using IoT Technology." *Proceedings of the 2017 International Conference on Information and Communication Technology Convergence (ICTC)*.

Key Insights:

- IoT-based SoC systems enable remote monitoring, alerts, and real-time tracking.
- Enhances user interaction and system diagnostics via smartphones/web.
- Supports data logging and cloud-based analytics.

Limitations:

- Hybrid SoC estimation via IoT still under-researched.
- Scalability and affordability remain challenges for small applications.

Objectives

- Develop a real-time SoC estimation system for lithium-ion batteries.
- Implement hybrid estimation using Direct Voltage, Coulomb Counting, and Kalman Filter methods.
- Improve SoC accuracy under dynamic conditions and fluctuating loads.
- Ensure low-cost, low-power design using affordable microcontrollers and sensors.
- Create a scalable and deployable solution for research and small-scale applications.

Hardware Components

- **Arduino Uno**: Central microcontroller for data processing and wireless transmission.
- **ACS712**: Current sensor module for Coulomb counting.
- **Voltage sensor**: Measures battery voltage for direct SoC estimation.
- **16x2 LCD Display**: Displays live SoC values and system status.
- **Li-ion Battery (3.7V/18650)**: Target cell for SoC monitoring.
- **Resistors, Breadboard, Wires**: For circuit construction and interfacing.
- **USB Power Supply**: Powers the system during testing and deployment.

Arduino Uno

- **Role**: Main controller and processor.
- **Function**:
  - Reads analog voltage and current data from sensors.
  - Calculates State of Charge (SoC) using voltage-based, Coulomb Counting, and Kalman filter methods.
  - Displays the results on an LCD and sends data to the serial monitor.

ACS712

- **Role**: Measures the current drawn by or supplied to the battery.
- **Function**:
- Outputs an analog voltage proportional to the current.
- The zero-current output is 2.5V; deviation from this indicates current direction and magnitude.
- Arduino reads this voltage, subtracts the 2.5V offset, and converts it to current using sensitivity (e.g., 185 mV/A for ACS712-5A).

MH25V

•**Role**: Power source under test.
•**Function**:
•Its real-time voltage and current are continuously monitored.
•SoC is calculated based on its capacity (e.g., 2.2Ah), voltage range (e.g., 3.0V–3.65V), and discharge behavior.

LCD with I2C Module

•**Role**: Displays battery status in real time.
•**Function**:
•Shows voltage (V), current (A), and SoC (%) in a user-friendly format.
•Uses I2C interface (only two pins: SDA and SCL), leaving more pins free on the Arduino.

Sequence of Operation

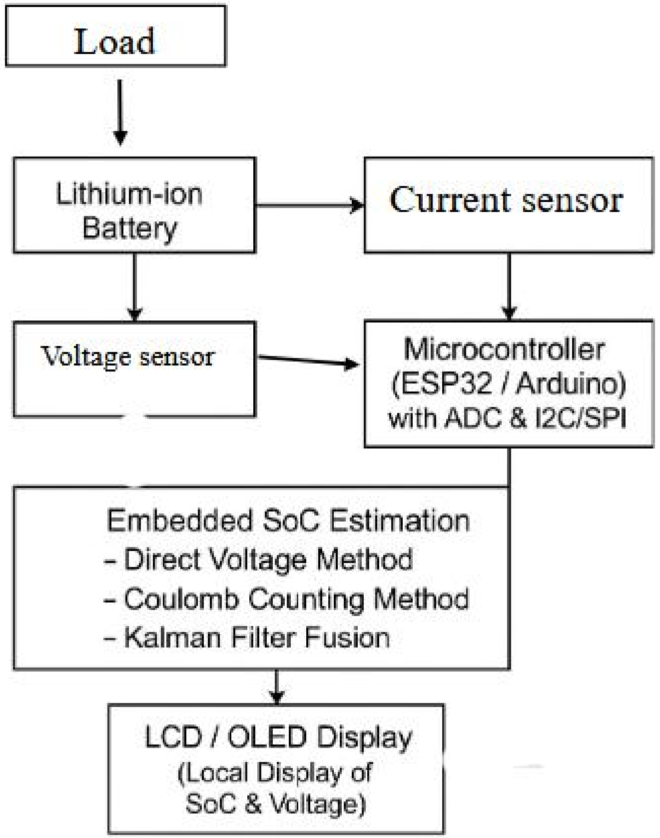


Figure 4.1: Block Diagram

• **Battery Monitoring**
The system begins by monitoring the **Li-ion battery** (3.7V/18650) connected to the circuit.

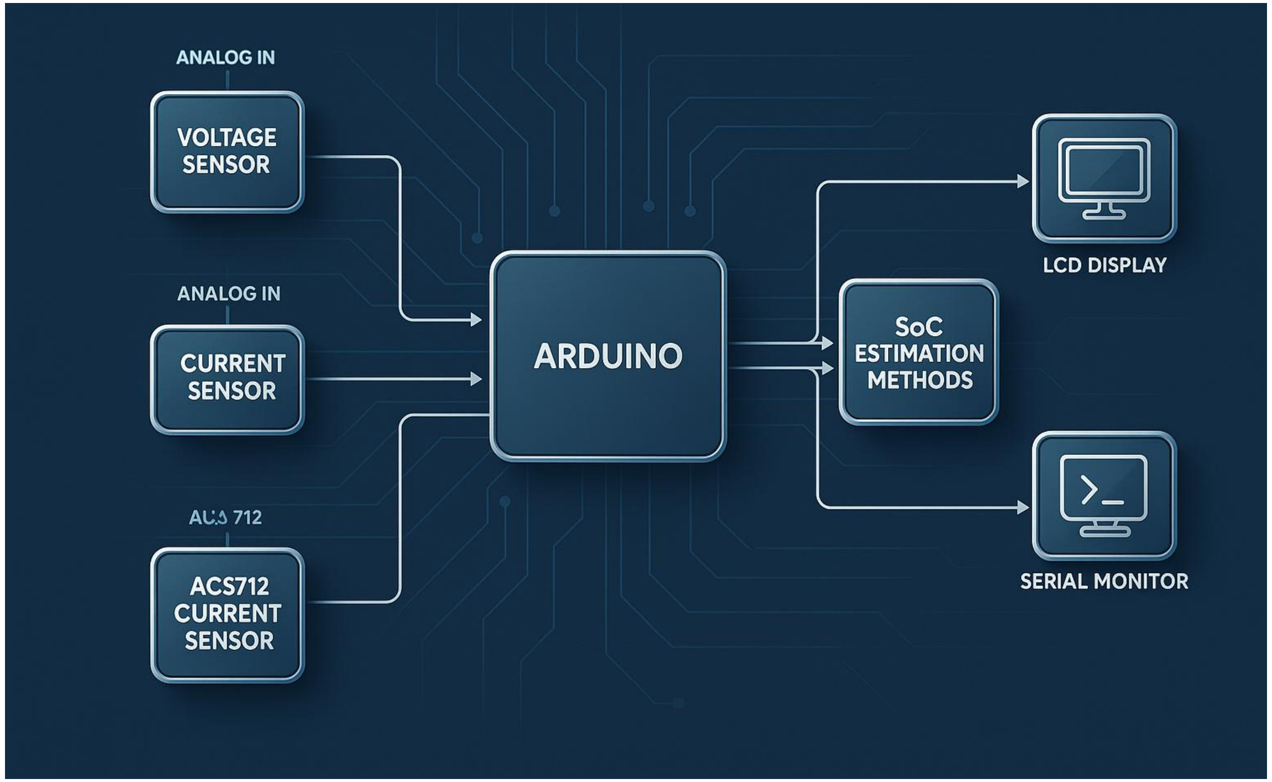
• **Voltage Measurement**
The **voltage divider circuit** measures the **open-circuit voltage** of the battery.
The voltage value is converted to **State of Charge (SoC)** using a calibrated lookup table or polynomial regression, primarily for **rest periods** or low-current load states.

• **Current Measurement (Coulomb Counting)**
The **ACS712 current sensor** is used to measure the **current flowing in or out** of the battery.
The system performs **Coulomb counting**, integrating the current over time to calculate the SoC.

• **Kalman Filter Integration**
The **Kalman filter** is applied to predict and update the SoC using real-time data.
It minimizes errors, suppresses noise, and provides **accurate and stable SoC** estimation, especially under rapidly changing conditions.

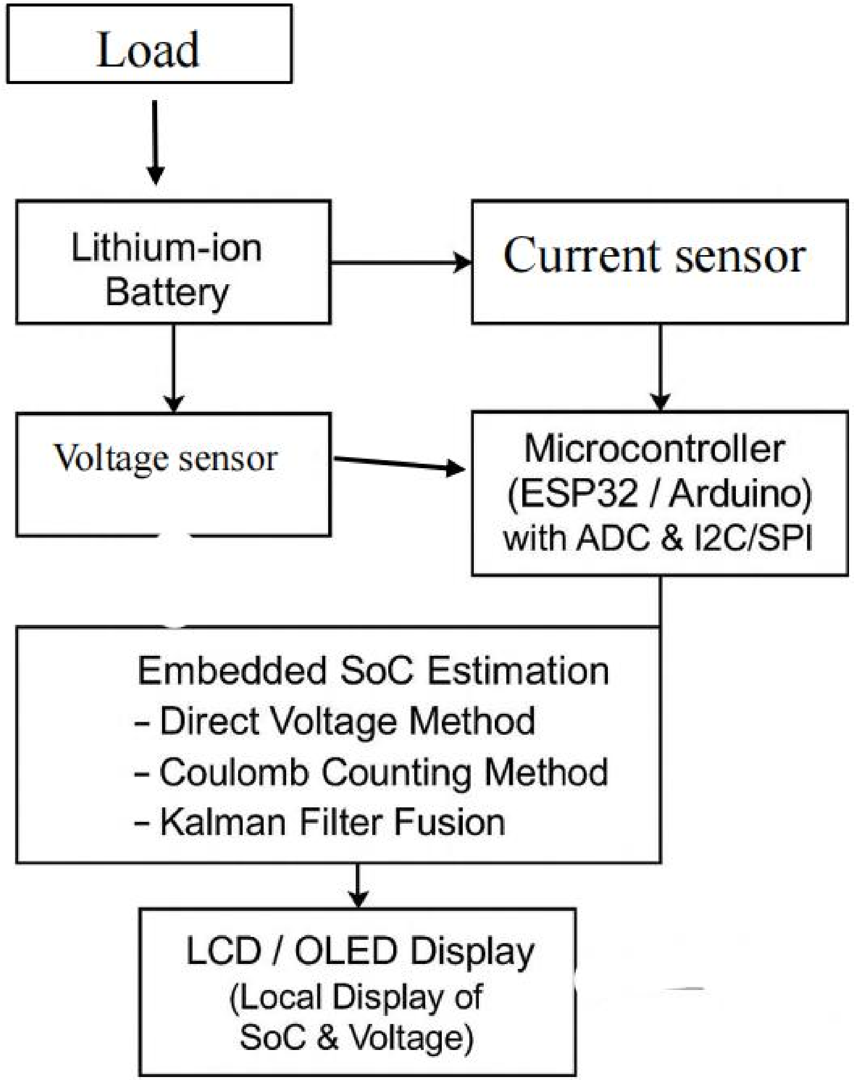
• **Display and Feedback**
16x2 LCD display shows the **live SoC values** and **system status**.
The user can monitor the battery's **current charge level** and **performance** through the display.

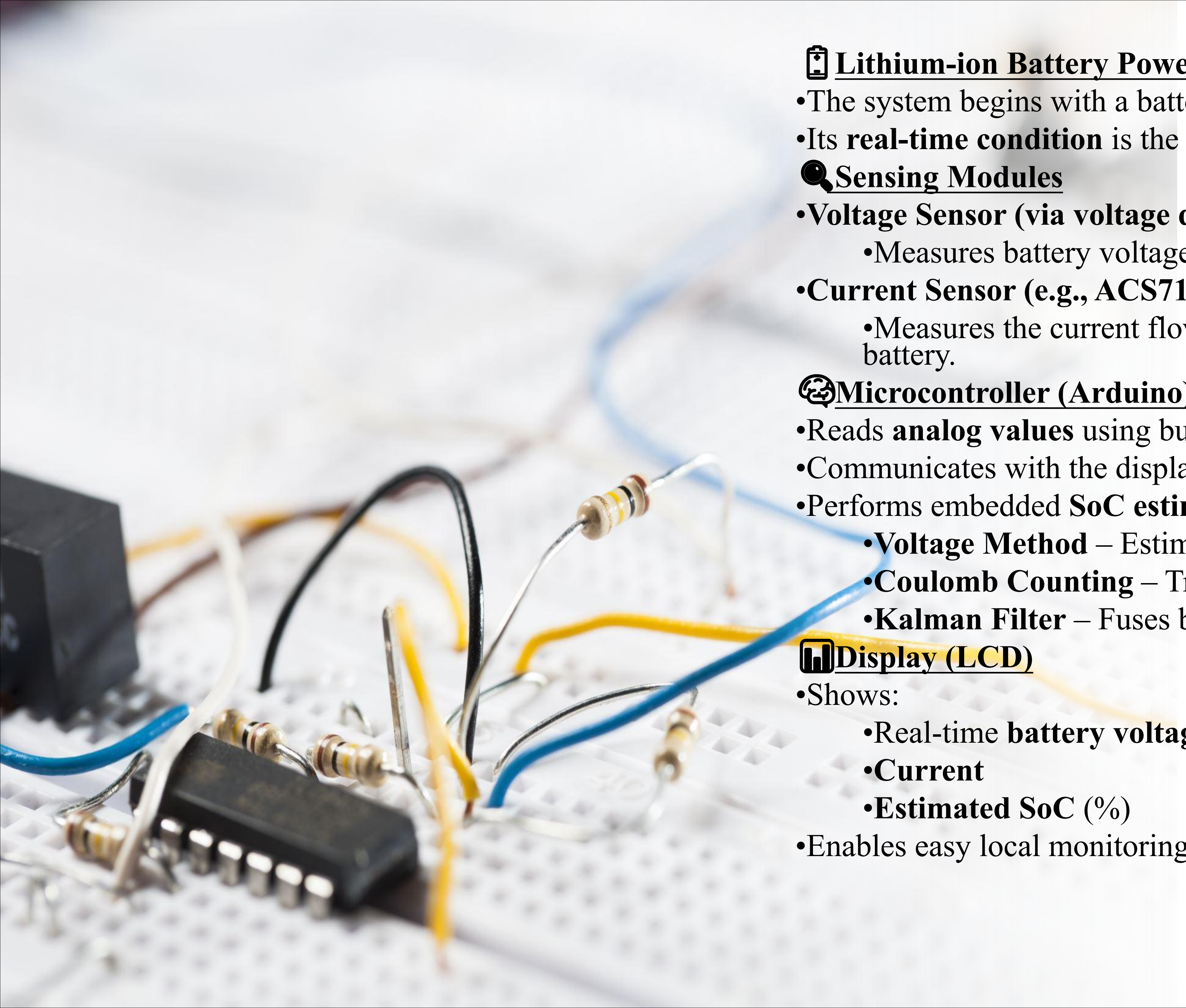
Programming



- **Inputs:** Reads battery voltage and current using analog pins.
- **Processing:** Converts readings, removes noise, and assumes discharge current.
- **SoC Estimation:** Uses three methods — voltage-based, coulomb counting, and Kalman filter.
- **Output:** Displays voltage, current, and SoC on LCD and serial monitor.
- **Purpose:** Estimates battery State of Charge (SoC) accurately using IoT sensors.

Design Approach





🔋 Lithium-ion Battery Powers the Load

- The system begins with a battery supplying current to a connected load.
- Its **real-time condition** is the primary focus of monitoring.

🔍 Sensing Modules

- **Voltage Sensor (via voltage divider)**:
  - Measures battery voltage and feeds it to the microcontroller.
- **Current Sensor (e.g., ACS712)**:
  - Measures the current flow (typically discharge current) from the battery.

🧠 Microcontroller (Arduino)

- Reads **analog values** using built-in ADC.
- Communicates with the display through **I2C/SPI**.
- Performs embedded **SoC estimation** using:
  - **Voltage Method** – Estimates SoC from voltage level.
  - **Coulomb Counting** – Tracks charge consumed over time.
  - **Kalman Filter** – Fuses both methods for improved accuracy.

📊 Display (LCD)

- Shows:
  - Real-time **battery voltage**
  - **Current**
  - **Estimated SoC** (%)
- Enables easy local monitoring without needing external interfaces.

Results

- **Voltage-based Method**:
  - Simple and quick but suffers from voltage sag under load.
  - Accurate when the battery is idle but inaccurate during active charge/discharge.
- **Coulomb Counting Method**:
  - Provides real-time SoC tracking with accurate short-term estimations.
  - Prone to cumulative errors and drift over time; requires periodic resets.
- **Kalman Filter Method**:
  - Fuses voltage-based and Coulomb counting methods for improved accuracy.
  - Minimizes drift, offset errors, and improves stability under varying loads.
  - Optimal Kalman gain range: 0.4–0.6 for balancing responsiveness and noise suppression.
- **IoT Integration**:
  - Enabled real-time monitoring with smooth data transmission and mobile notifications.
- **Overall Outcome**:
  - The hybrid approach (Kalman filter + IoT) provides a balanced, scalable solution for real-time battery monitoring.

The LCD displays a battery voltage of **3.92V**, current of **0.0A**, and a **State of Charge (SoC)** of **90%**, indicating that the battery is nearly full and not currently discharging.

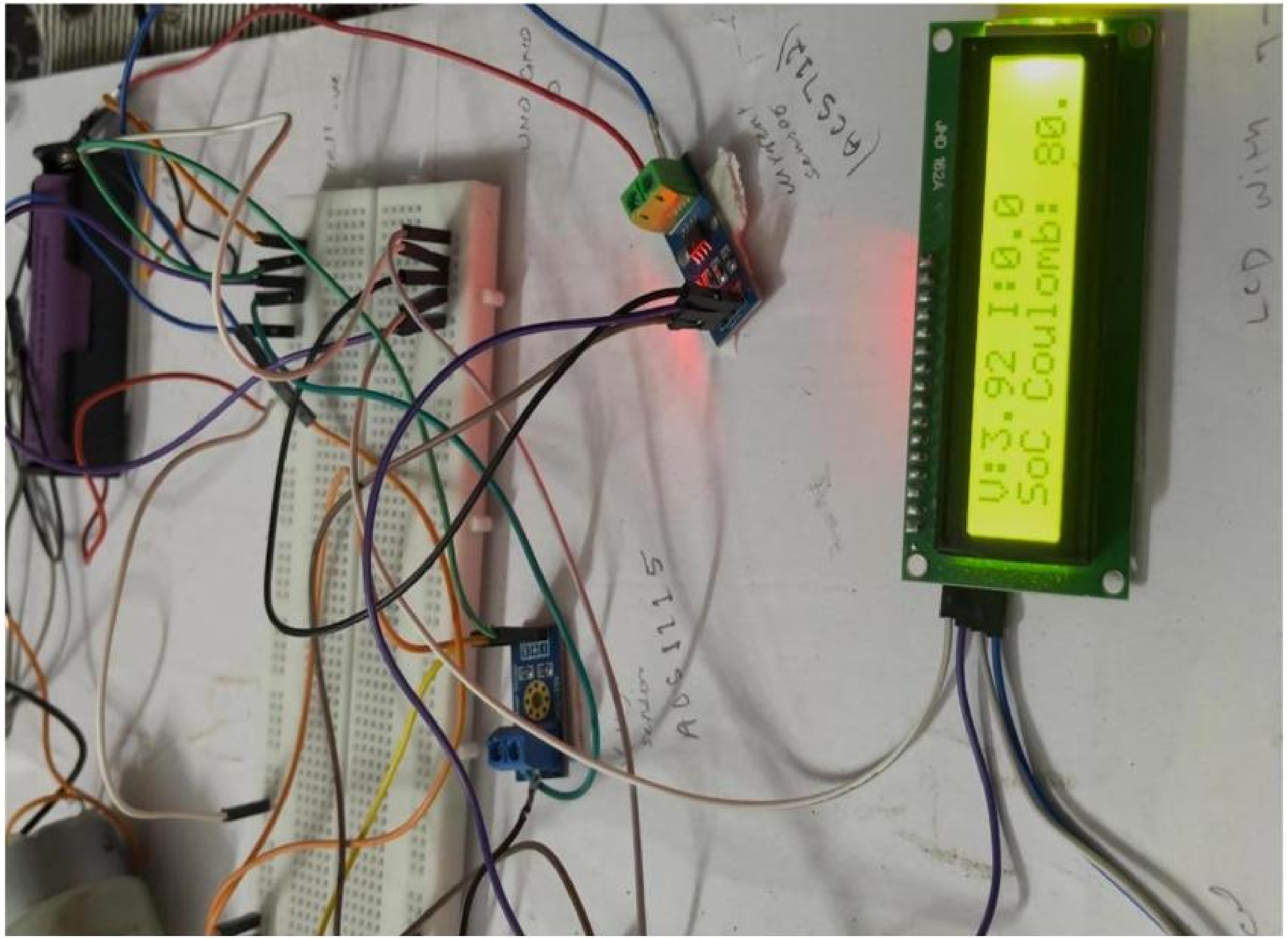


Figure 8.1.1: Hardware setup for Coulomb counting method

The LCD shows a battery voltage of **3.98V**, current of **0.0A**, and a **Kalman Filter-based SoC** of **92.1%**, indicating a well-charged battery with refined estimation accuracy using sensor fusion.

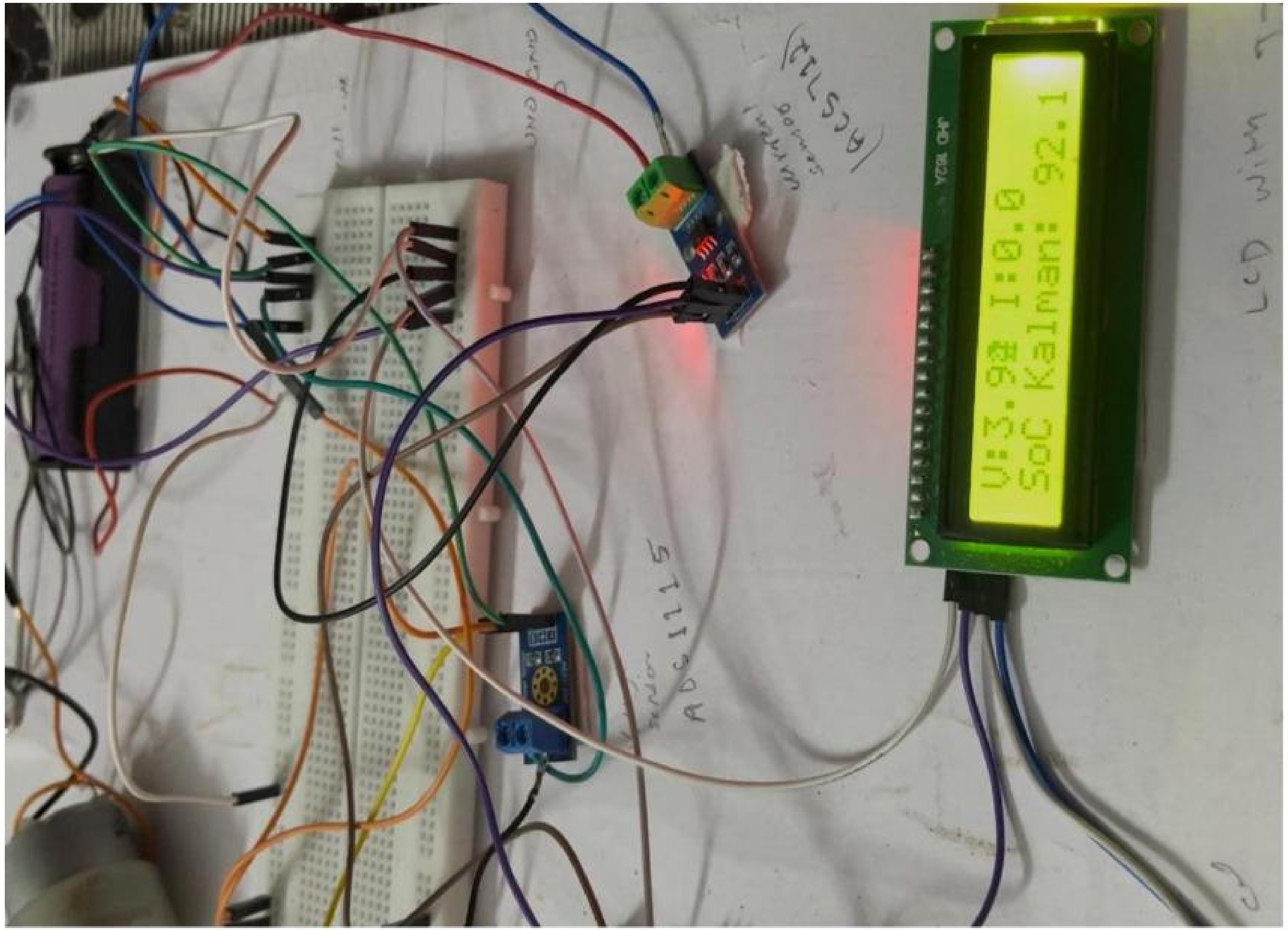


Figure 8.1.2: Hardware setup for Kalman Filter

- The LCD displays a battery voltage of **3.93V**, current of **0.0A**, and a **State of Charge (SoC)** of **100%** based on the **Open Circuit Voltage (OCV) method**, indicating a fully charged battery when at rest.

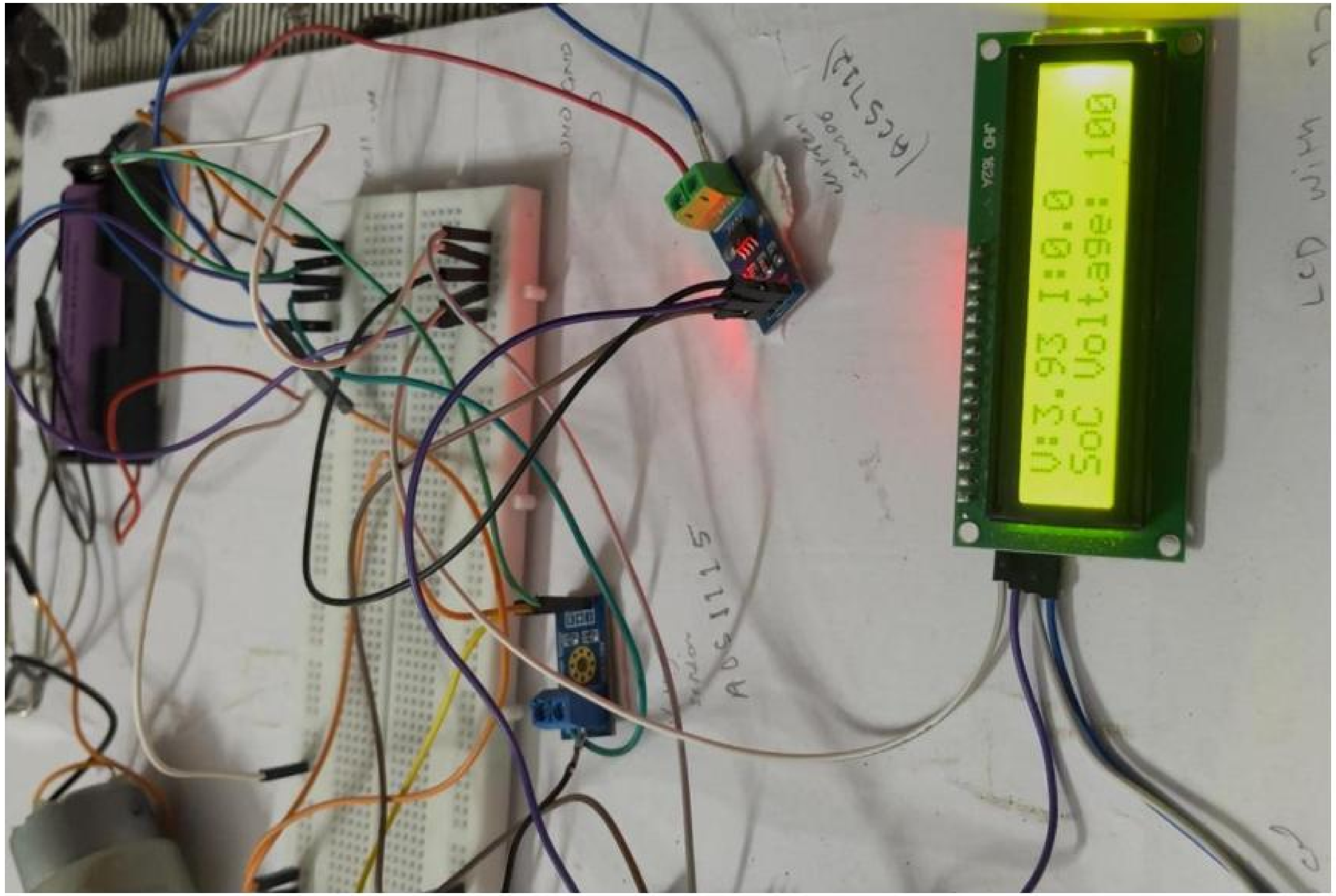


Figure 8.1.3: Hardware setup for OCV method

MATLAB Simulation Output

•Compares **actual vs. predicted** values using an **LSTM (Long Short-Term Memory)** network.

•The close alignment between the blue (actual) and red (predicted) curves confirms the model’s high accuracy in forecasting battery SoC trends.

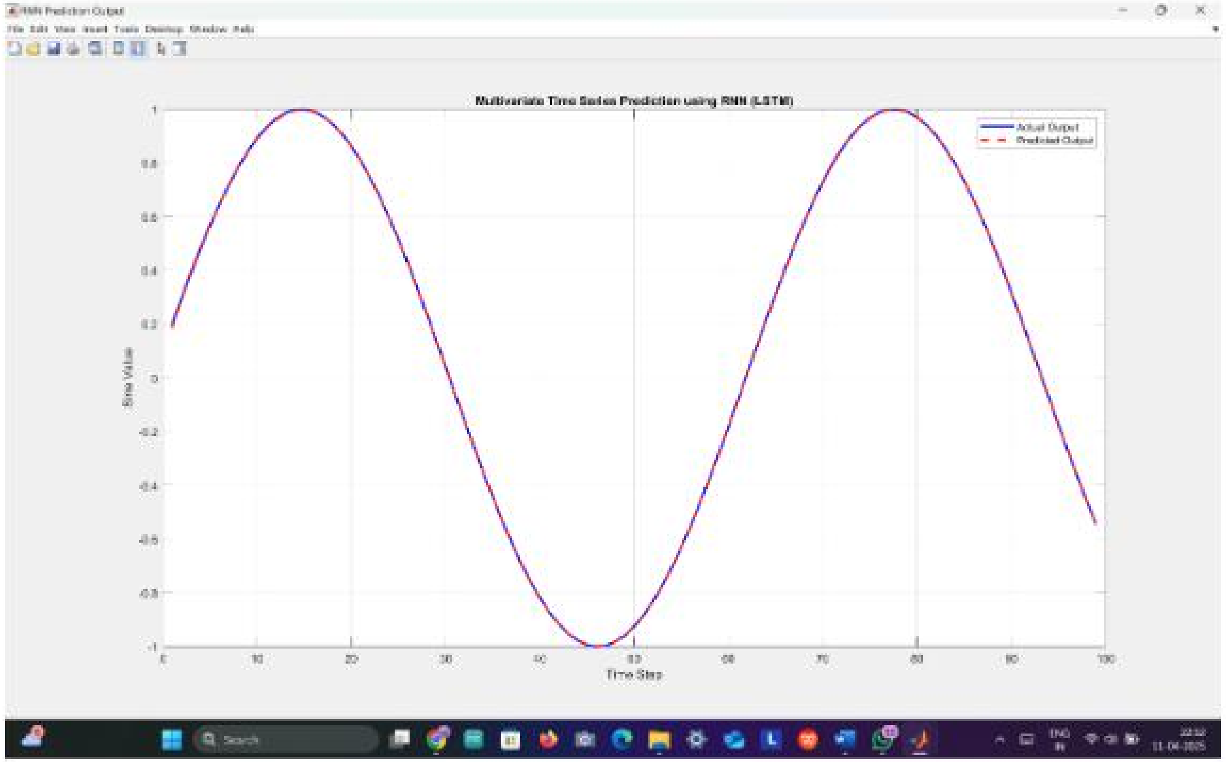


Figure 8.3.3: MATLAB output graph for multivariable time series predictions

Table 8.4 – Result Summary

Feature	Accuracy	Response Time	Remarks
Voltage Estimation	±5% at rest	Instantaneous	Affected by load conditions
Coulomb Counting	±3% (short term)	Real-time	Drift over long use, reset periodically
Kalman Filter	±2%	1 second	Best combined accuracy, load-insensitive
IoT Dashboard	100% update rate	~1–2 sec delay	Mobile notifications and data logs successful
System Integration	Stable	Continuous	No lag; modules synced through MCU

Conclusion

- **System Development**:
    - Successful development of a real-time embedded system for battery SoC estimation.
- **Estimation Techniques**:
    - Integrated voltage-based, Coulomb counting, and Kalman filter methods.
    - Voltage-based: Fast but limited to open-circuit conditions.
    - Coulomb counting: Real-time tracking but prone to drift.
    - Kalman filter: High accuracy and stability under dynamic conditions.
- **Performance**:
    - Achieved SoC estimation accuracy within ±2–3%.
- **Usability**:
    - SoC displayed on local LCD and cloud-connected IoT dashboard for real-time monitoring.

References

- S. Piller, M. Perrin, and A. Jossen, "Methods for state-of-charge determination and their applications," Journal of Power Sources, vol. 96, no. 1, pp. 113–120, 2001.
- G. L. Plett, "Extended Kalman Filtering for Battery Management Systems of LiPB-Based HEV Battery Packs: Part 1. Background," Journal of Power Sources, vol. 134, no. 2, pp. 252–261, 2004.
- C. Zhang, K. Li, J. Deng, and J. Gong, "A novel approach for estimating the state of charge of lithium-ion batteries based on a hybrid model," Applied Energy, vol. 105, pp. 142–151, 2013.
- S. M. Noh et al., "An IoT-based smart battery monitoring system," in IEEE Sensors Journal, vol. 17, no. 17, pp. 5617–5625, 2017.
- A. Hannan, M. Lipu, A. Hussain, and A. Mohamed, "A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations," Renewable and Sustainable Energy Reviews, vol.78, pp. 834–854, 2017.
- Blynk Inc., "IoT App Builder Platform," [Online]. Available: https://www.blynk.io
- MathWorks, "Battery State of Charge Estimation," [Online]. Available:

https://www.mathworks.com