WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

IoT-Based Real-Time Battery Health Monitoring System for Electric Vehicles

Dr. M. Vinay Kumar[1], N. Pragnya[2], K. Hemanth Kumar[3], N. Thanusri[4]

Department of Electrical and Electronics Engineering (EEE), GMR Institute of Technology (GMRIT), Rajam, Andhra Pradesh, India

ABSTRACT :

The performance, reliability and safe operation of an electric vehicle depend strongly on the condition of its battery pack. Continuous observation of electrical and thermal parameters can help identify abnormal operating conditions and support timely maintenance. This paper describes the development of a low-cost Internet of Things (IoT)-enabled battery monitoring prototype based on an ESP32 microcontroller. A resistor-divider circuit is used to measure pack voltage, while an INA219 sensor provides bus-voltage and current measurements through the I²C interface. A DS18B20 digital sensor is included in the design for temperature measurement. The controller processes the acquired readings, calculates instantaneous electrical power and estimates the battery State of Charge (SOC) using Coulomb counting. The measured information is made available through a browser-based dashboard hosted by the ESP32 and accessed over the local Wi-Fi network. The prototype was evaluated using a 3-series (3S) lithium-ion battery pack. During the reported test session, the dashboard displayed pack voltages of 11.74 V and 11.66 V, discharge currents of 0.09 A and 0.18 A, and corresponding power values of 1.09 W and 2.04 W. The estimated SOC remained at 80%. The temperature sensor was reported as disconnected during these measurements; therefore, temperature measurement and over-temperature protection were not experimentally validated. The results demonstrate the feasibility of browser-based monitoring of selected battery parameters using inexpensive embedded hardware.

Keywords— Internet of Things, electric vehicles, battery monitoring, ESP32, INA219, DS18B20, State of Charge, Coulomb counting, web dashboard.

I. INTRODUCTION

Electric vehicles (EVs) use rechargeable battery packs as their primary energy source. The battery influences vehicle range, acceleration capability, operating cost and overall safety. Lithium-ion batteries are widely used because of their favourable energy density and power characteristics. However, their performance can deteriorate when they are repeatedly exposed to unsuitable operating conditions, including excessive current, high temperature, overcharging and deep discharging.

A battery monitoring arrangement measures important operating quantities and presents them in a form that can be understood by a driver, technician or maintenance operator. Battery management systems commonly supervise voltage, current and temperature, and may estimate parameters such as State of Charge (SOC) and State of Health (SOH). SOC represents the remaining charge relative to the battery's usable capacity. SOH describes the battery's condition relative to a reference condition and generally requires additional measurements or a validated estimation method.

IoT connectivity can make battery information easier to access. A microcontroller equipped with wireless communication can collect sensor readings and publish them to a local or remote interface. In this prototype, an ESP32 performs both data acquisition and web-server functions. A user connected to the same Wi-Fi network can view the monitored values using a web browser, without requiring a separate gateway.

The work focuses on the implementation of a practical monitoring prototype rather than a complete automotive-grade battery management system. Its objectives are:

- to acquire battery voltage and current using low-cost sensing hardware;
- to provide a provision for battery temperature measurement;
- to calculate instantaneous power and estimate SOC from current measurements;
- to display live measurements and system status through an ESP32-hosted web interface; and
- to evaluate the prototype using measurements collected from a 3S lithium-ion battery pack.

The remainder of this paper is organised as follows. Section II discusses the technical background and related monitoring approaches. Section III explains the proposed architecture and implementation. Section IV presents the prototype observations. Section V discusses limitations, Section VI concludes the work, and Section VII identifies future improvements.

II. TECHNICAL BACKGROUND AND RELATED WORK

A. Battery Parameters

Battery voltage indicates the electrical potential of the pack and can provide useful information about its operating state. Current indicates the rate at which charge enters or leaves the battery. The product of voltage and current gives instantaneous electrical power. Temperature is also important because battery performance and safety are affected by thermal conditions.

SOC is commonly expressed as a percentage of the estimated remaining charge relative to the nominal or usable capacity. Coulomb counting estimates SOC by integrating current over time. The method depends on a suitable initial SOC, an accurate current measurement and a known battery capacity. Measurement offset and capacity uncertainty can cause the estimate to drift.

SOH is different from SOC. SOC estimates how much charge is currently available, whereas SOH concerns longer-term degradation, such as capacity loss or increased internal resistance. The prototype described in this paper calculates SOC; it does not implement or validate a quantitative SOH estimation algorithm. Consequently, the term battery health monitoring in the system title refers to monitoring relevant operating parameters and displaying an SOC estimate, not to a fully validated SOH diagnosis.

B. IoT-Based Monitoring

An IoT monitoring system typically contains sensors, a processing unit, a communication link and a user interface. The sensors acquire physical measurements, the controller converts them into useful values, and the communication layer transfers the information to a display or storage platform. A local web dashboard is a simple implementation because it can be accessed by devices on the same network without installing a dedicated application.

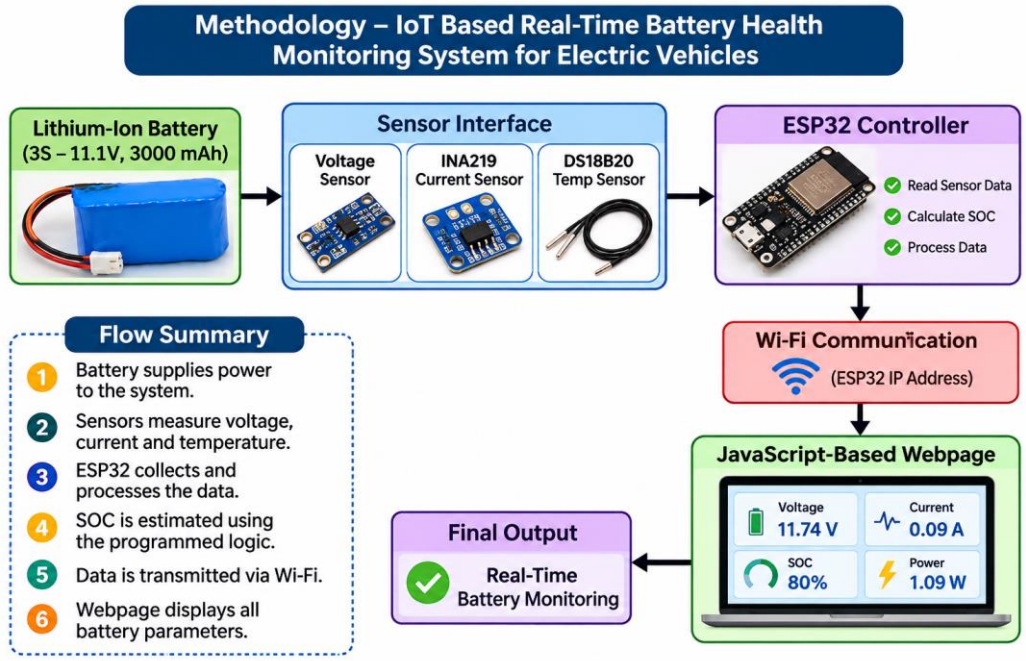
The ESP32 is suitable for this prototype because it provides digital interfaces, analogue input capability and integrated Wi-Fi connectivity. The INA219 is used for current and bus-voltage sensing over I²C. The DS18B20 provides a digital temperature-sensing interface using the 1-Wire protocol. These components form the sensing and communication basis of the proposed design.

III. PROPOSED SYSTEM AND IMPLEMENTATION

A. System Architecture

The proposed system is organised into four functional stages: battery input, sensor acquisition, embedded processing and wireless presentation. The 3S lithium-ion battery pack is the monitored source. The voltage-divider circuit scales the pack voltage to a level suitable for the ESP32 analogue input. The INA219 measures current and bus voltage through the I²C interface. The DS18B20 is provided for temperature measurement. The ESP32 reads the sensors, processes the measurements, estimates SOC and hosts the dashboard.

Fig. 1. System architecture and methodology of the proposed IoT-based battery monitoring prototype.



B. Hardware Components

- Battery pack: A 3S lithium-ion pack with a nominal voltage of approximately 11.1 V is used for prototype evaluation. A battery management/protection circuit is included with the pack.
- ESP32 development board: The ESP32 acquires sensor data, performs calculations and provides the Wi-Fi web server.

- Voltage sensing circuit: A resistor-divider module scales the battery voltage before it is applied to GPIO34, an ADC input. The dashboard configuration indicates a usable input range of 0–16.5 V.
- INA219 current sensor: The INA219 is connected through the I²C bus. It measures shunt voltage and bus voltage, from which current and electrical power are obtained.
- DS18B20 temperature sensor: This digital sensor is included on a 1-Wire connection to measure battery temperature. The firmware is configured to issue a warning above 45 °C; however, the sensor was disconnected during the reported test captures.
- Additional interface and control components: The prototype description includes a 16×2 I²C LCD, buzzer, LED indicators and a relay module for local indication and possible load or charger control. An LM2596 buck converter is used to derive a 5 V supply.

Component	Interface / connection	Role in the prototype
3S lithium-ion battery pack	Battery input / protected pack	Provides the monitored DC source; nominal pack voltage is about 11.1 V.
ESP32 development board	ADC, I²C, 1-Wire support, Wi-Fi	Acquires sensor data, performs calculations and hosts the local dashboard.
Voltage-divider module	GPIO34 ADC input	Scales pack voltage to a range suitable for the controller ADC.
INA219 sensor	I²C bus	Measures shunt voltage and bus voltage for current and power calculations.
DS18B20 sensor	1-Wire digital interface	Provides battery temperature measurement when connected.
LCD, buzzer and LEDs	I²C / digital outputs	Provide local display and status or warning indications.
Relay module	Controller output	Provides a switching interface for a load or charger-control function, subject to safe implementation.
LM2596 buck converter	DC–DC power conversion	Derives the 5 V supply used by the prototype electronics.

Table I summarizes the principal sensing and control components used in the prototype.

C. Sensor Acquisition and Processing

The controller samples the monitored parameters at an interval of 500 ms. Pack voltage is acquired through the resistor divider connected to GPIO34. The INA219 independently reports bus voltage and provides the shunt-voltage measurement used to determine current. Since the two voltage readings are obtained through different measurement paths, they can be compared to identify differences that may arise from calibration, wiring or measurement-point variation.

Instantaneous power is calculated from the measured voltage and current:

$$P(t) = V(t) \times I(t) \qquad (1)$$

where P is instantaneous power in watts, V is voltage in volts and I is current in amperes. The sign convention of current determines whether the battery is charging or discharging.

D. State-of-Charge Estimation

SOC is estimated using Coulomb counting. The method accumulates the current flowing into or out of the battery over time and relates the accumulated charge to the nominal battery capacity. For a discharge-current convention in which positive current represents charge leaving the battery, the estimate can be written as:

$$SOC(t) = SOC(t_0) - [1 / C_n] \int_{t_0}^{t} I(\tau)\, d\tau \qquad (2)$$

Here, $SOC(t_0)$ is the initial state-of-charge estimate, C_n is the nominal capacity in ampere-hours, and $I(\tau)$ is the measured discharge current. In a digital implementation, the integral is approximated by summing current samples over their sampling intervals. The accuracy of this estimate depends on initialisation, sensor calibration, sampling and the actual usable battery capacity. The dashboard therefore presents SOC as an estimate rather than a direct sensor measurement.

E. Web Dashboard and Communication

The ESP32 hosts a browser-based dashboard and communicates with a client over the same Wi-Fi network. The user opens the board's IP address in a browser to view the live information. The dashboard includes status cards for voltage, current, temperature, and SOC; a selectable live trend chart; an electrical-parameters panel; a system-alert log; and connection and Wi-Fi status indicators. Additional information includes hardware, network and controller status.

This local-network arrangement provides convenient access without a separate cloud service. It does not, by itself, provide monitoring from outside the local Wi-Fi network. Remote access requires a secure network or cloud communication layer.

IV. EXPERIMENTAL SETUP AND RESULTS

A. Prototype Test

The prototype was powered by a 3S lithium-ion battery pack, and the dashboard was accessed via the ESP32 IP address. Two dashboard captures were recorded during the same test session, at elapsed operating times of 6 min 52 s and 10 min 45 s. The two dashboard captures are shown in Figs. 3 and 4, and the corresponding readings are summarised in Table III. They represent two observations from one prototype session and should not be interpreted as a complete charge-discharge characterisation.

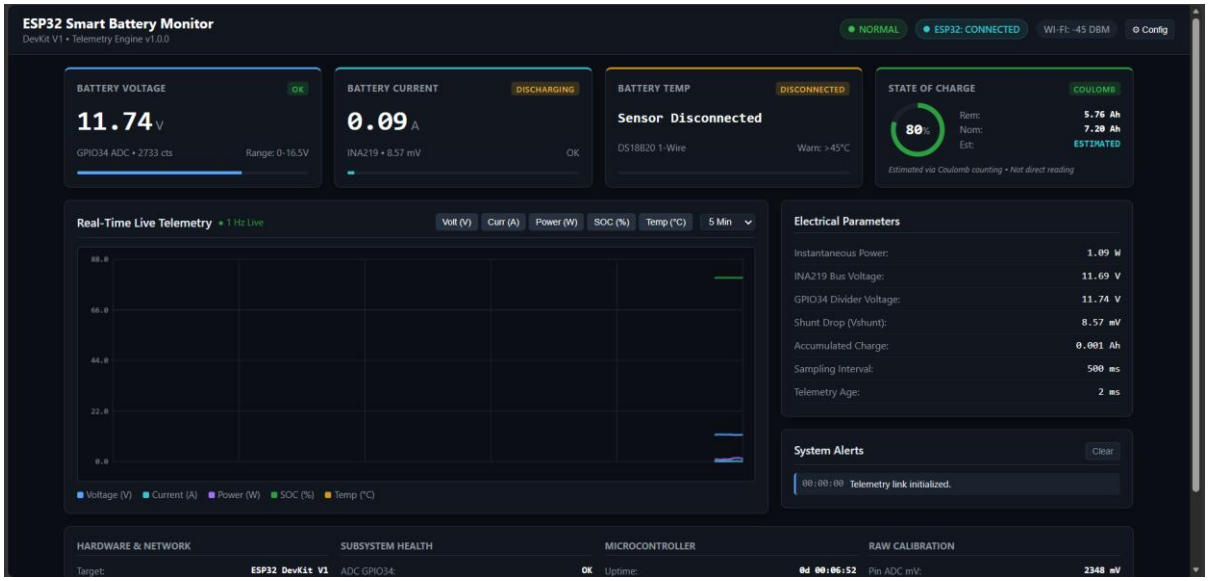


Fig. 3. Dashboard capture 1: 11.74 V pack voltage, 0.09 A discharge current and 80% estimated SOC.

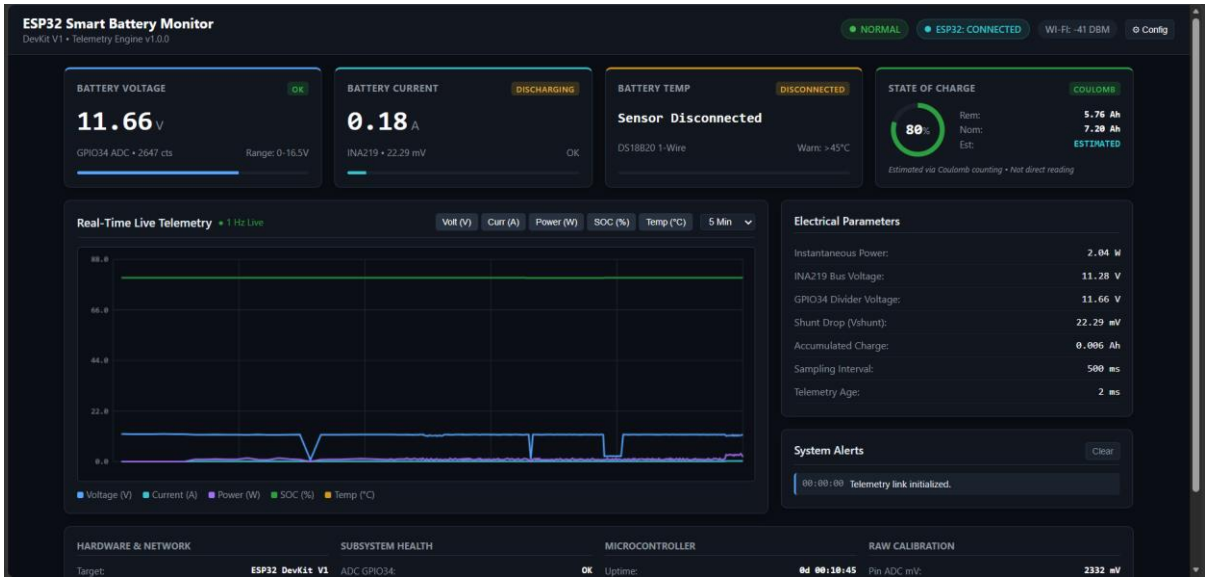


Fig. 4. Dashboard capture 2: 11.66 V pack voltage, 0.18 A discharge current and 80% estimated SOC.

Parameter	Capture 1	Capture 2
Battery voltage from GPIO34 ADC (V)	11.74	11.66
ADC pin voltage (mV)	2348	2332
INA219 bus voltage (V)	11.69	11.28
Battery current (A)	0.09	0.18
Shunt voltage (mV)	8.57	22.29
Instantaneous power (W)	1.09	2.04
Accumulated charge (Ah)	0.001	0.006

Estimated SOC (%)	80	80
Remaining / nominal capacity (Ah)	5.76 / 7.20	5.76 / 7.20
Wi-Fi signal strength (dBm)	−45	−41
Sampling interval (ms)	500	500
Telemetry age (ms)	2	2
Battery status	Discharging / Normal	Discharging / Normal
Temperature sensor status	Disconnected	Disconnected

Table II. Values reported by the ESP32 dashboard during the two prototype captures.

B. Interpretation of Observations

The voltage-divider channel displayed 11.74 V in the first capture and 11.66 V in the second. The corresponding ADC pin values were 2.348 V and 2.332 V. These pairs are consistent with the configured 5:1 scaling factor. The INA219 bus-voltage readings were 11.69 V and 11.28 V, respectively. The difference between the two voltage channels increased in the second capture, indicating that calibration and measurement-point checks are required.

The displayed discharge current increased from 0.09 A to 0.18 A. The reported shunt voltage also increased from 8.57 mV to 22.29 mV. The dashboard power values increased from 1.09 W to 2.04 W, which is consistent with the higher current at a similar pack voltage. These observations show that the monitoring interface responded to a change in the measured load.

The SOC estimate remained at 80% in both captures, corresponding to 5.76 Ah remaining from a configured nominal capacity of 7.20 Ah. The accumulated charge changed from 0.001 Ah to 0.006 Ah. The small difference in accumulated charge was not sufficient to change the displayed whole-number SOC value.

The dashboard reported a sampling interval of 500 ms and a telemetry age of 2 ms in both captures. The reported Wi-Fi signal strengths of −45 dBm and −41 dBm indicate that the ESP32 maintained a wireless connection during the observations. The status field displayed a normal discharging condition.

V. DISCUSSION AND LIMITATIONS

The prototype demonstrates an integrated approach for collecting selected battery measurements and presenting them through a local web dashboard. The ESP32 combines the sensing interface, basic calculations and Wi-Fi hosting in one controller. The measured voltage, current and power values were updated in the dashboard, and the SOC estimate was displayed alongside the configured capacity.

Some limitations must be considered when interpreting the results. First, the INA219 bus-voltage values did not exactly match the voltage-divider readings. The difference may be associated with the different measurement points, wiring losses, sensor tolerances or calibration. Both channels should be checked against a calibrated reference multimeter before quantitative accuracy is claimed.

Second, the DS18B20 was reported as disconnected in both test captures. Although the firmware indicated the missing sensor, the reported experiment does not validate temperature measurement or the 45 °C warning threshold. No temperature accuracy or thermal protection performance should therefore be inferred from these results.

Third, the SOC value is calculated using Coulomb counting and depends on the initial SOC and configured nominal capacity. The two dashboard captures do not independently validate the SOC estimate against a laboratory reference. Finally, the reported tests do not establish battery SOH, long-term reliability, protection response under fault conditions or automotive-grade safety compliance. These aspects require additional controlled experiments.

VI. CONCLUSION

An IoT-enabled battery monitoring prototype for electric vehicles has been described using an ESP32 controller, a resistor-divider voltage input and an INA219 current sensor, with a DS18B20 included for temperature sensing. The controller calculates instantaneous power, estimates SOC through Coulomb counting and presents the monitored information on a browser-based dashboard over local Wi-Fi. In the reported 3S battery test, the dashboard displayed pack voltages of 11.74 V and 11.66 V, discharge currents of 0.09 A and 0.18 A, and power values of 1.09 W and 2.04 W. The results support the feasibility of low-cost, local-network monitoring of selected electrical battery parameters. However, temperature sensing, SOC accuracy and SOH estimation require further validation before the design can be considered a comprehensive battery health assessment system.

VII. FUTURE SCOPE

- Connect the DS18B20 correctly and validate its readings using a calibrated temperature reference.
- Calibrate the voltage and current measurement channels and investigate the difference between the voltage-divider and INA219 readings.
- Conduct controlled full charge and discharge tests to evaluate SOC estimation error against a suitable reference method.
- Develop and validate an SOH estimation approach using capacity measurements, internal-resistance indicators or other suitable diagnostic features.
- Test overvoltage, undervoltage, overcurrent and over-temperature warning conditions, including the buzzer, LED, LCD and relay functions.
- Evaluate the system under different loads and operating temperatures, and assess long-duration data stability.

- Add secure cloud-based data logging if monitoring outside the local Wi-Fi network is required.

REFERENCES

[1] K. Sujit, K. C. Ramaswamy, S. R. Mathiyalagan, J. Giri, and M. Kanane, "An efficient battery management system for electric vehicles using IoT and blockchain," Results in Engineering, 2025, doi: 10.1016/j.rineng.2025.106284.

[2] R. R. Kumar, C. Bharatiraja, K. Udhayakumar, S. Devakirubakaran, K. S. Sekar, and L. Mihet-Popa, "Advances in batteries, battery modeling, battery management system, battery thermal management, SOC, SOH, and charge/discharge characteristics in EV applications," IEEE Access, vol. 11, 2023.

[3] F. Mohammadi and R. Rashidzadeh, "An overview of IoT-enabled monitoring and control systems for electric vehicles," IEEE Instrumentation & Measurement Magazine, 2021.

[4] R. Usha Rani, "IoT based battery monitoring system for electric vehicle using ESP32," in Proc. SmartTechCon, 2023, doi: 10.1109/SmartTechCon57526.2023.10391772.

[5] M. Ismail and R. Ahmed, "A comprehensive review of cloud-based lithium-ion battery management systems for electric vehicle applications," IEEE Access, 2024, doi: 10.1109/ACCESS.2024.3446880.

[6] S. Mishra, S. C. Swain, and R. K. Samantaray, "A review on battery management system and its application in electric vehicle," in Proc. ICACC, 2021, doi: 10.1109/ICACC-202152719.2021.9708114.