A Battery Monitoring System based on IoT for Electric Vehicles

NAGARAJA S R[1], REDDAM JAGADESWAR REDDY[2], VARIKUTI SAMPATH KUMAR REDDY[3], KALIMISETTY SUJITH[4], NITTURU NIKHIL SAI[5], DUMPA SRI VARDHAN REDDY[6]

[1]*Associate Professor, Presidency University, Bangalore*

[2, 3, 4, 5]*Department of CSE, Presidency University, Bangalore*

Abstract— Electric vehicles (EVs) have gained increasing prominence because of their environmental benefits, but monitoring the health of batteries is one big challenge. Hence, this paper proposes a real-time battery monitoring system for electric vehicles that measures and keeps track of important parameters like voltage and current. The Arduino UNO is integrated with current and voltage sensors to collect data and upload it to the ThingSpeak server. A machine learning model based on the Random Forest and Decision Tree algorithms is trained to estimate the health of the battery and its state of charge (SoC). Besides, a friendly user website interface is also created to show real-time voltage, battery health, and alerts. This holistic approach ensures better management of batteries, assures safety, and reduces operating costs.

Index Terms- Internet of Things, Electric Vehicles Battery, Grid, State of Charge, Message Queue Telemetry Transport (MQTT), Battery Management System.

I. INTRODUCTION

Electric vehicles have been gaining rapid popularity as a substitute for internal combustion engine-powered vehicles because they cause less damage to the environment and have lower operating costs. Nevertheless, one of the challenges EV makers and users face is ensuring the health and efficiency of the batteries as time progresses. A Battery Management System (BMS) plays an essential role in monitoring key parameters such as battery voltage, current, State of Charge (SoC), and State of Health (SoH). The precision of monitoring will ensure that the battery works effectively and also, at the required time, gives a signal to the user regarding whether it needs some sort of service or replacement. This research, therefore, proposes innovative hardware, software, and machine learning-based battery monitoring for real-time insight into the health of EV batteries. In the proposed system, Arduino UNO is the central controller that communicates with current and voltage sensors to gather parameters of the battery. The system uses ThingSpeak, an IoT platform, to upload the data for remote storage and analysis. Additionally, machine learning techniques are used to analyze the data gathered and predict the health and life cycle of the battery. This paper presents the methodology in the development of the system, including the sensors, data processing techniques, and machine learning models. The research also discusses the design and implementation of a user-friendly website that displays the health metrics of the battery and alerts when the battery needs maintenance or replacement.II. Monitoring the health and performance of the battery is important for ensuring its optimal operation. Traditional battery monitoring methods either call for periodic checks or depend on simple voltage and current measurements that do not allow in-depth insight into the general health of the battery. The rapid strides in sensor technology and cloud computing offer the prospect of making the monitoring of batteries a real-time, data-driven process. This paper describes an EV battery monitoring system integrating hardware, software, and machine learning to track battery parameters like voltage, current, and state of health (SoH).

II. LITERATURE SURVEY

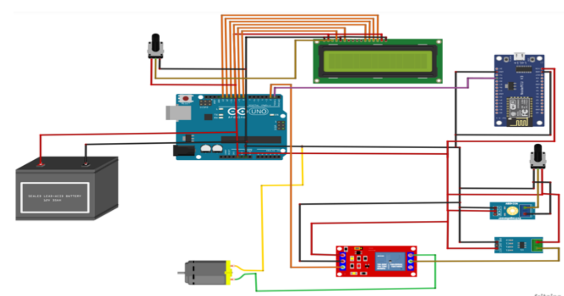
Battery health monitoring is one of the most critical aspects of EV development, and several different approaches have been explored in the literature. Traditional battery monitoring systems often depend solely on basic voltage and current readings to estimate the performance of the battery. These methods may be limited in their accuracy and fail to provide detailed insight into the overall health of the battery. More recent work with machine learning and

IoT has enabled more sophisticated methods of battery health monitoring. For example, Wang et al. (2019) developed an IoT-based BMS that uses real-time data from sensors to monitor the voltage, current, and temperature of a battery. Their system used cloud storage for data analysis and developed predictive models for battery health forecasting using machine learning techniques.Another study by Chen et al. (2020) demonstrated the application of deep learning algorithms in predicting the remaining useful life of lithium-ion batteries and proved that machine learning algorithms can outperform the traditional methods with regard to accuracy and in predictive power. Our study, therefore, focuses on the integration of the simpler machine-learning models like Random Forest and Decision Trees, which had been effective for smaller-scale IoT applications for battery health prediction. In addition, our study integrates an easier-to-use interface that users can view on a website and understand the insights of the batteries' performance in real time—a feature highly sought after when data visualization is to be presented for practical applications.The application of Random Forest and Decision Trees in battery health prediction has been explored in various research studies, such as by Zhang et al. (2021), who applied these algorithms to predict battery life and assess charging/discharging cycles. These models provide the flexibility and robustness required to handle complex data inputs from multiple sensors, making them suitable for this type of monitoring system.

III. PROPOSED SYSTEM

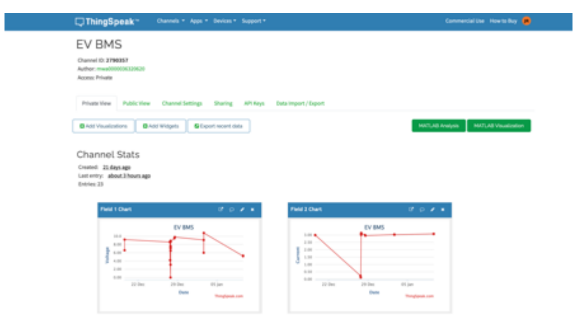
The proposed EV battery monitoring system is composed of various interconnected sub-components: data acquisition hardware, cloud storage for data transmission, machine learning models for analyzing battery health, and a frontend web application for data visualization. The Arduino UNO microcontroller is the beginning point of the system, which interacts with current and voltage sensors attached to the EV battery. These sensors sample real-time voltage and current values from the battery, which are critical in monitoring the performance of the battery in calculating parameters such as State of Charge (SoC) and State of Health (SoH).The data collected from the sensors is transmitted to Thingspeak, which is a cloud platform for Internet of Things (IoT) data analytics in real time. Thingspeak allows for the continuous monitoring and storage of sensor data and provides an easy way to visualize trends and patterns. The system also connects to a trained Machine Learning model that has been developed on historical data collected from various battery performance datasets. The model utilizes the algorithms of Random Forest and Decision Tree to predict the future health of batteries and determines when it is likely to need maintenance or replacement. The frontend website is developed in React JS, which retrieves data from the Thingspeak server and represents it in a more friendly, intuitive form. The proposed system allows for more informed decision-making regarding battery usage, maintenance schedules, and best practices for charging that will ultimately lead to extended battery life and lower costs for EV users. The uploaded data is then analyzed by the machine learning models—Random Forest and Decision Tree algorithms—for predicting the SoC and SoH of the battery. These models are trained with historical data in order to build a predictive framework that can precisely predict the future performance of the battery based on its present state. The user interface, which is developed as a website, displays real-time metrics including battery voltage, current, and health status.

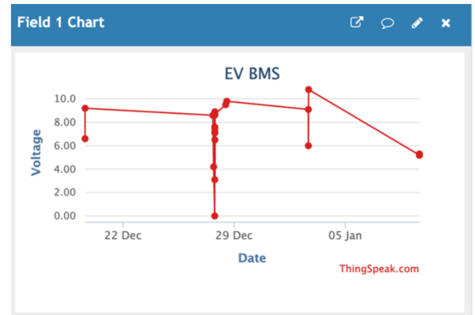
IV. BLOCK DIAGRAM

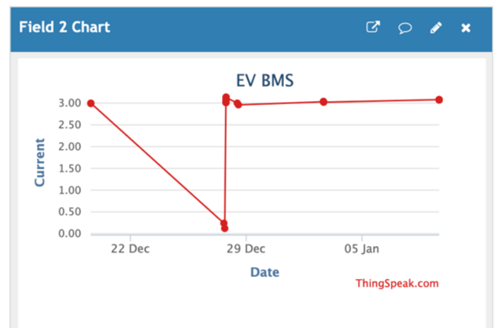


V. DATACOLLECTION PROCESS FROM ARDUINO CURRENT AND VOLTAGE SENSORS AND SENDING TO THINGSPEAK SERVER.

The data collection process begins with the connection of current and voltage sensors to the EV battery through the Arduino UNO microcontroller. These sensors are capable of measuring the real-time voltage and current values of the battery, which are crucial indicators of battery performance and health. The voltage sensor measures the potential difference across the battery terminals, while the current sensor tracks the rate of flow of electric charge. These readings are important for calculating various battery parameters, including the State of Charge (SoC), which indicates the remaining capacity of the battery, and the State of Health (SoH), which reflects the battery's ability to hold charge relative to its original capacity.After collecting the data, it is transmitted by the Arduino UNO through a Wi-Fi module to the Thingspeak server for storage and analysis. Thingspeak is a cloud-based platform that supports real-time data monitoring and analysis through APIs. The data is uploaded to the Thingspeak server every few seconds, hence the performance of the battery is monitored continuously. The server stores the data in a structured format, which is accessible for further processing and visualization.In parallel, the system also processes the data to calculate the SoC and SoH of the battery using predefined algorithms. These values are then sent to the cloud, where they are stored along with the raw sensor data. Machine learning algorithms like Random Forest and Decision Trees analyze historical data from the battery to forecast its future health.







VI. HARDWARE DESCRIPTION

A. Aurdino Uno Microcontroller:

An Arudino UNO is a microcontroller board with one of the frequently used ATmega328P quads. An Arduino UNO is one of the most preferred boards of development for most people as it has a wide range of applications including sensor interfacing, automation, robotics among many others. It can also accept inputs from sensors and provide outputs to devices including motors, relays and screens.

B. LCD (Liquid Crystal Display):

An LCD is a display that consists of liquid crystals which are used to convey information. They are commonly found in electronic devices for outputting texts or data like reading and status messages. Inside your project, this can aid you in outputting live data such as the voltage or current alongside other information.

Fig. 4. Liquid Crystal Display

C. Voltage Sensor:

Voltage Sensor: A voltage sensor is a device that determines the amount of energy between two circuit points. This device enables a circuit such as the Arduino to track voltage information efficiently. This sensor is particularly useful for monitoring the voltages of the batteries embedded within your project aiding in SoC monitoring.

Fig. 3. Voltage Sensor

D. Current Sensor:

Current Sensor measures current flowing through a circuitry by outputting a signal in proportion to the flow which the microcontroller then interprets. In consideration of your project, this sensor aids in monitoring the current running in the battery which enables you to estimate the batteries health and current status.

E. Relay

A relay is an electrically operated switch that lets a low-power signal control a high-power circuit. It acts as an interface between the low-power microcontroller and the high-power components such as the motor or other actuators. In your project, a relay can be used to control the power flow to various components based on sensor readings or alerts.

F.DC Motor:

This is an electric motor that runs using DC electricity. It will change electrical energy into mechanical motion that can be applied toa project in the sense of making or controlling real motion For example, it can simulate the motion of a vehicle part moving around, or check whether a battery is powerful enough to power a motor in a car.

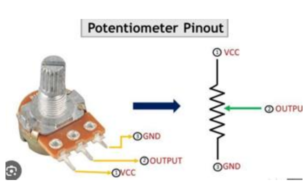
G.Buzzer:

It is an audio-signaling device that produces sound when powered; it is broadly used for alert or warning purposes related to certain conditions. Use a buzzer in the project to alert the users to thecritical oc currences like running low on the battery and notify the userto take care of its health.

H. Battery 12V: It is a DC 12V battery which gives an output voltage of 12V.This is t he power supply to many electronic devices. It's crucial to your project as it powers the sensors, Arduino board, and all other parts. The health and charge of the battery need to be checked to ensure the vehicle runs efficiently.

I.Node MCU: It is an open-source development board, based on the ESP8266 Wi-Fi module. It is used to add Internet connectivity to any IoT projects. In your project, the Node MCU can be used to send sensor data (voltage, current) to the cloud and allow remote monitoring via an app or web interface.

J.Potentiometer: A potentiometer is a variable resistor. It can be used to change the resistance within a circuit and thereby control the amountof current flow. It could be used for adjusting parameters like voltage or current in your project. It might also be useful to manually adjust a parameter related to the battery or motor.



VII. SOFTWARE DESCRIPTION

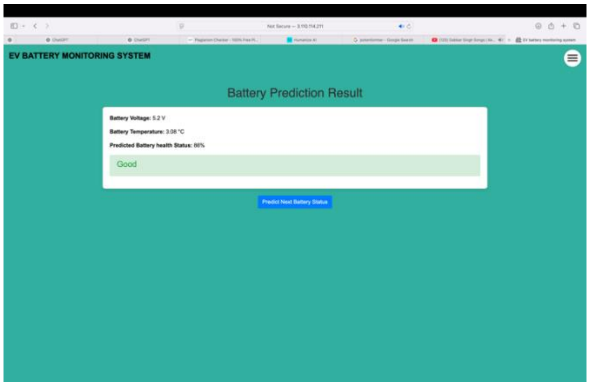
Frontend website displaying current battery voltage, present battery health:

The frontend user interface was designed to show a very clear and concise view of the status of the battery. It is developed in React JS, which fetches real-time data from the Thingspeak server, where sensor readings of voltage, current, SoC, and SoH are stored. The website presents this data in a very aesthetically pleasing and understandable manner, showing real-time graphs, battery status indicators, and predictive health alerts.

The website has a dashboard that shows the current voltage and current of the battery. It also supports graphical representation of the SoC and SoH. All the metrics update in real-time, and users get to see the performance and health of the battery at any time. The website also has an alerting system, showing a message to the user when the battery health has reached a critical level, meaning it needs to be replaced or serviced.

The system supports better management of EV battery health, reduces the risk of unexpected failure, and also extends the life of batteries by combining cloud data storage with real-time analytics and an intuitive user interface.

It also contains features like history view of the battery's performance over time, with trends in battery voltage and health. This feature allows users to understand the long-term behavior of their EV batteries and take proactive measures to avoid potential failures.



VIII. CHALLENGES AND CONSIDERARTIONS

Though the proposed EV battery monitoring system offers a number of advantages, its successful implementation has been associated with various challenges. For instance, sensor data accuracy is one of the prime concerns. Precise measurements will depend on correct calibrations of the voltage and current sensors. Inaccurate predictions regarding the health of the battery may be produced due to inappropriate sensor data. Furthermore, because sensors deteriorate over time, the reliability of the system would decrease, and maintenance or recalibration may be required more frequently.

Reliable communication between the Arduino microcontroller and the Thingspeak server is another challenge. Data loss or delays due to connectivity issues, particularly in remote areas, can jeopardize the monitoring system's timeliness. This may be addressed through the inclusion of failover mechanisms or local data storage so that data will not be lost in case of communication disruption..

IX. RESULTS AND DISCUSSION

The system has been implemented successfully, and initial tests have shown promising results. The Arduino UNO in combination with the voltage and current sensors has given accurate real-time data, which has been transmitted to the Thingspeak server successfully. The Machine Learning model trained on various datasets of battery performance has demonstrated its ability to predict the future health of the battery with reasonable accuracy and identified the potential issues much before they become critical.

The frontend website has been a very efficient tool in showing the real-time data and battery health information. It is easy to access and understand the data presented, which informs the users' decisions on matters related to the maintenance of batteries. The system's predictive alerts have been able to prevent failures in batteries, which gives users adequate time to replace or service their batteries before encountering significant problems.

Results were promising but required further adjustments to increase the accuracy and reliability of the system. This would mean more diversification in datasets as well as how real-time data is collected. The last step would be to scale the tests to other EV models and configurations.

CONCLUSION

This paper presents a new EV battery monitoring system that utilizes hardware, software, and machine learning to monitor the health of the battery in real time. The system therefore monitors key parameters such as voltage, current, and State of Health, thereby making it possible for predictive maintenance to be sent for the extension of the lifespan of EV batteries. Data will be collected by using Arduino. It will have Thingspeak-based cloud storage. User interface is built using React JS, for easy and effective management of a battery.

The proposed system has a lot of promise concerning the efficiency of electric vehicle battery monitoring in terms of proper timely maintenance and cost-effective operations. Accuracy challenges regarding sensors, communication between data, and model optimization, however, have to be overcome for improved functioning of the system. In the future, the system will be highly scalable, accurate, and user-friendly, and more advanced and reliable electric vehicle battery management systems can be developed.

REFERENCES

[1] S. Yonghua, Y. Yuexi, H. Zechun, "Present Status and Development Trend of Batteries for Electric Vehicles," Power System Technology, Vol. 35, No. 4, pp. 1-7, 2011.

[2] X. Kuang, K. Li, Y. Xie, C. Wu, P. Wang, X. Wang, C. Fu., "Research on Control Strategy for a Battery Thermal Management System for Electric Vehicles Based on Secondary Loop Cooling," in IEEE Access, vol. 8, pp. 73475-73493, 2020.

[3] K. Mansiri, S. Sukchai and C. Sirisamphanwong, "Fuzzy Control Algorithm for Battery Storage and Demand Side Power Management for Economic Operation of the Smart Grid System at Naresuan University, Thailand," in IEEE Access, vol. 6, pp. 32440-32449, 2018.

[4] M. A. Hannan, M. M. Hoque, A. Hussain, Y. Yusof and P. J. Ker,

[5] "State of-the-Art and Energy Management System of Lithium-Ion Batteries in Electric Vehicle Applications: Issues and Recommendations," in IEEE Access, vol. 6, pp. 19362-19378, 2018.

[6] Fang, R.; Chen, K.; Yin, L.; Sun, Z.; Li, F.; Cheng, H.M., "The Regulating Role of Carbon Nanotubes and Graphene in Lithium-Ion and Lithium-Sulphur Batteries," Adv. Mater. 2018.

[7] F. Zhu, G. Liu, C. Tao, K. Wang and K. Jiang, "Battery management system for Li-ion battery," in The Journal of Engineering, vol. 2017, no.13, pp. 1437-1440, 2017.

[8] M. Luo, Y. Xiao, W. M. Sun, and Z. Wang, "Online battery monitoring system based on GPRS for electric vehicles" Proceedings - 2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics, IHMSC 2013, Vol. 1, pp. 122–125, 2013.

[9] A. Rahman, M. Rahman and M. Rashid, "Wireless battery management system of electric transport," IOP Conf. Ser. Mater. Sci. Eng. 2017.

[10] N. Hossein Motlagh, M. Mohammadrezaei, J. Hunt, and B. Zakeri,

[11] "Internet of Things (IoT) and the Energy Sector," Energies, vol. 13, no. 2, p. 494, Jan. 2020.

[12] K. Kumar, and B. Jaipal, "The Role of Energy Storage with Renewable Electricity Generation," in Electric Grid Modernization. London, United Kingdom: Intech Open, 2022[Online].

[13] S. K. Panda and A. Ghosh, "A Low Ripple Load Regulation Scheme for Grid Connected Microgrid Systems," IEEE 8th Power India International Conference (PIICON) 2018.

[14] J. Meher and A. Ghosh, "Comparative Study of DC/DC Bidirectional SEPIC Converter with Different Controllers", EEE 8th Power India International Conference (PIICON) 2018.

[15] Ghosh, S. Banerjee, M. K. Sarkar, and P. Dutta, "Design and implementation of type-II and type-III controller for DC–DC switched mode boost converter by using K-factor approach and