Development of a remote monitoring system for lithium-ion batteries by using IoT and real-time processing

Kosuke Shibuya

Dept. Electronic and Computer Engineering
Ritsumeikan University
Kusatsu, Japan

Masahiro Fukui

Dept. Electronic and Computer Engineering
Ritsumeikan University
Kusatsu, Japan
e-mail: mfukui@se.ritsumei.ac.jp

Abstract - In the context of the rapid proliferation of electric vehicles and challenges such as shortages and price increases in rare metals, there is a growing demand for reused batteries with significant performance variations. Furthermore, with the increasing importance of remote monitoring, encompassing overall lifecycle management, efficient utilization, and early detection of malfunctions, including both reused and new batteries, there is a high demand for the development of a remote monitoring and history management system for lithium-ion batteries. Against this background, the authors are actively engaged in the development of a system for dynamic charge state monitoring using IoT and the accumulation of historical data on the cloud. This paper presents a report on the development of dynamic charge state monitoring using Kalman Filter and data communication/management using MQTT.

I. Introduction

In recent years, global warming caused by greenhouse gases such as carbon dioxide has become a social issue, and achieving a decarbonized society is being required as a solution. Electric vehicles are becoming increasingly popular, which has led to a growing interest in energy storage battery technology. Lithium-ion batteries, due to their high voltage, high energy density, and high power output, have rapidly gained popularity in applications such as electric vehicles.

Storage batteries generate heat and can deteriorate quickly due to overcharging and discharging. Therefore, it is essential to have a battery management system (BMS) that can monitor the internal state of the battery. In recent years, there has been an increasing demand for optimal state management, effective utilization, early detection of malfunctions, and other requirements throughout the entire lifetime of new and reused batteries. Remote monitoring, real-time condition management, and historical data management have become crucial aspects of BMS. Therefore, the importance of BMS systems by using IoT, as shown in Fig. 1, is growing[1, 2].

Generally, on the edge side, systems are implemented using low-power, not very high-performance microcomputers. Therefore, functions that have light computational cost such as current, voltage, and temperature measurements are implemented. Whereas, on the host side, the estimation of internal states, and advanced and complex target (such as lifespan, anomalies, residual value, etc.) is performed by using Neural Networks (NN) [3-5]. State of Charge (SOC) and internal resistance are often estimated by using Kalman Filter

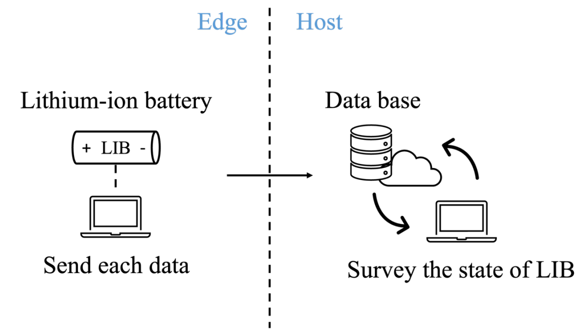


Fig. 1. Configuration of the remote monitoring BMS

or Recursive Least Squares (RLS) methods. However, the estimation was performed after completing to collect data for all timestamp like batch processing.

For the above backgrounds, the authors have undertaken several initiatives. (1) First initiative is the development of an advanced IoT-type BMS for functional verification. To enhance the efficiency of function and algorithm development, a Raspberry Pi 4B, Python, and I2C are employed as the edge-side computer, programing language, and communication protocol between sensors and Raspberry Pi 4B. While the Raspberry Pi 4B is not suitable for implementation in low-power IoT edge systems due to its high power consumption, it is suitable to elevate the functional design completeness as a design platform. Once algorithm implementation is sufficiently completed, detailed design and the implementation can be carried out on a low-power practical system using compilers of other efficient programing language such as C++. (2) Among the internal state estimation functions, the SOC estimation function is implemented using real-time processing on the edge side with a Kalman Filter. Since the Kalman Filter has a light computational cost, it can operate even when Raspberry Pi 4B is replaced with a low-power microcomputer.

In this paper, following an overview of the IoT-type BMS platform, we report on the real-time processing implementation of the Kalman Filter for estimating the SOC and the communication function with the host side using MQTT.

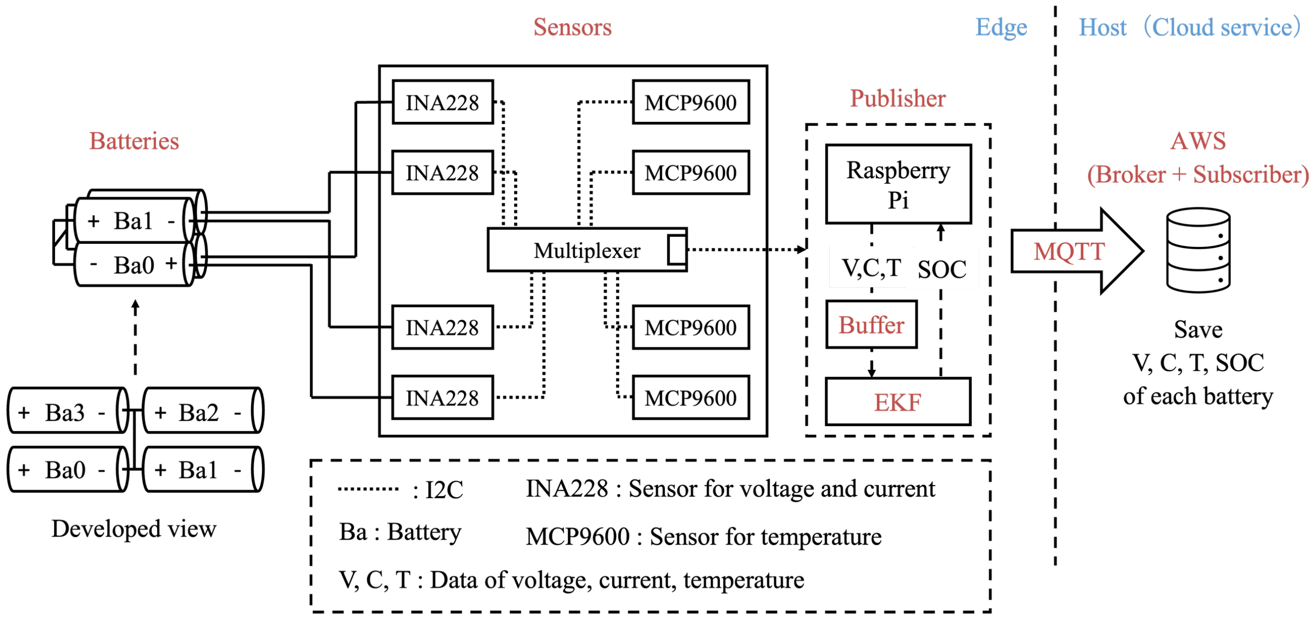


Fig. 2. The BMS architecture

II. System Scheme

For the system configuration enabling real-time remote state monitoring through IoT, this section describes the configuration for sensing, data communication, and real-time processing to achieve it.

A. The BMS architecture

Fig. 2 shows the BMS architecture, and Fig. 3 shows the actual picture of the BMS edge device. The BMS consists of six components: First, a battery pack composed of four lithium-ion batteries (Batteries). Second, sensors that measure voltage, current, and temperature from each lithium-ion battery (Sensors). Third, a temporary storage for the data measured by the sensors (Buffer). Fourth, an Extended Kalman Filter (EKF) [6] used to estimate SOC (State of Charge) based on the measured data. Fifth, a Publisher that sends voltage, current, temperature, and SOC data to AWS (Amazon Web Services). Sixth, AWS which stores and manages all the data.

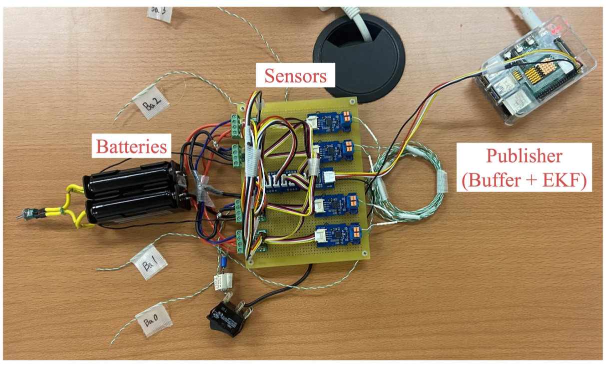


Fig. 3. The actual picture of the BMS edge device

The SOC estimated by the EKF and the measured voltage, current, and temperature data are recorded in AWS. In this process, MQTT (Message Queuing Telemetry Transport) is used for data communication with AWS.

B. MQTT protocol

Fig. 4 shows the data communication scheme of the MQTT, which is a publish/subscribe communication model composed of publisher, subscriber, and broker. The publisher sends a data (message) to the broker and assigns an identifier (topic) to the message to distinguish it from other messages, as shown in Fig. 4 (1). The subscriber then selects which messages it wants to receive by specifying a topic, as shown in Fig. 4 (2). If there is a message with the topic the subscriber specifies, the subscriber can receive the message from the broker, as shown in Fig. 4 (3). The publisher and subscriber communicate only with the broker who passes messages between them. This allows for quick and efficient delivery of messages, even with multiple publishers and subscribers. This characteristic is expected to provide an advantage in efficiently managing data from each battery pack when the number of the battery pack monitored by the BMS increases in the future.

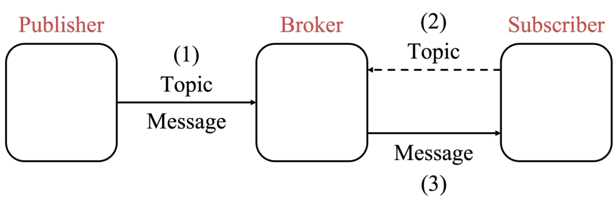


Fig. 4. Communication Scheme of the MQTT

C. The equivalent circuit model and Extended Kalman Filter

To describe the characteristics of a battery, the battery equivalent circuit model is used as Fig. 5. This model is composed of R_0, R_1, R_2, C_1, C_2, and u_{OCV}. R_0 is the resultant resistance consisting of electrolyte and electric double layer resistance. R_1, R_2 are diffusion resistance, and C_1, C_2 are capacitance of internal electrode. u_{OCV} is the value of open circuit voltage (OCV). In Fig. 5, i is the terminal current of the battery, u_L is the terminal voltage of the battery, u_1 (u_2) is the voltage of the parallel circuit consisting of R_1 (R_2) and C_1 (C_2).

In recent studies, the method for estimating SOC based on the Extended Kalman Filter (EKF) has been proposed. The EKF is an approximately optimal state estimator for a nonlinear stochastic process subject to Gaussian white noises using the state space model [7, 8].

$$x(k+1) = A(k)x(k) + B(k)u(k) + w(k) \tag{1}$$

$$y(k) = C(k)x(k) + v(k) \tag{2}$$

where (1) and (2) are the state and observation equations, respectively; $x(k)$, $y(k)$, and $u(k)$ are the state vector, the observation value, and the control input, respectively. The signal $w(k)$ is the system noise, and $v(k)$ is the observation noise. We assumed that both noises are zero-mean white Gaussian noise. For the battery equivalent circuit model, the detail of the state space model is shown as Table I.

The EKF algorithm is described as **Algorithm 1**. This algorithm consists of three steps: Initialization value, Prediction, and Filtering. Here, $\hat{x}^-$ is the estimation vector, $\hat{x}$ is the filtered estimation vector, P^- is the error covariance matrix, P is the filtered error covariance matrix. Then, as represented in equation (3), the $C(k)$ is the Jacobian matrix of $h(x(k))$ that represents the nonlinear relationship between OCV and SOC.

$$C(k) = \left.\frac{\partial h(x(k))}{\partial x(k)}\right|_{x(k)=\widehat{x(k)}} = \left[\left.\frac{dOCV}{dSOC}\right|_{SOC=\widehat{SOC}^-}, 1, 1, i(k)\right] \tag{3}$$

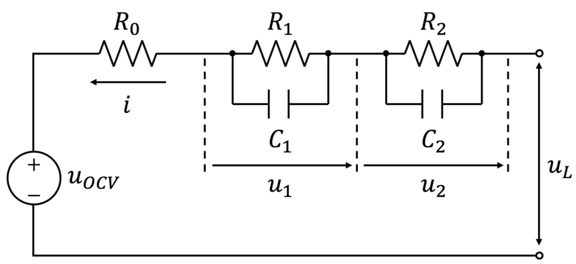


Fig. 5. Equivalent circuit model of Li-ion battery

TABLE I
The Details of State Space Model

$A(k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \left(1-\frac{\Delta t}{R_1C_1}\right) & 0 & 0 \\ 0 & 0 & \left(1-\frac{\Delta t}{R_2C_2}\right) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$B(k) = \begin{bmatrix} \frac{\Delta t}{FCC} \\ \frac{\Delta t}{C_1} \\ \frac{\Delta t}{C_2} \\ 0 \end{bmatrix}$
$h(x(k)) = u_{OCV} + u_1 + u_2 + iR_0$	
State Vector	$x(k) = \begin{bmatrix} SOC(k) \\ u_1(k) \\ u_2(k) \\ R_0(k) \end{bmatrix}$
Control Input	$u(k) = i(k)$
Observation Value	$y(k) = u_L(k)$

Algorithm 1 EKF

Initialization Value
$\hat{x}_0^-$, P_0^-

Filtering Step
$g_{k+1} = P_{k+1}^- C^T / (CP_{k+1}^- C^T + \sigma_v^2)$
$\hat{x}_{k+1} = \hat{x}_{k+1}^- + g_{k+1}(y_{k+1} - \hat{y}_{k+1}^-)$
$P_{k+1} = (1 - g_{k+1}C)P_{k+1}^-$

Prediction Step
$\hat{x}_{k+1}^- = A\hat{x}_k + Bu_k$
$P_{k+1}^- = AP_kA^T + \sigma_w^2$

D. Missing value imputation mechanism

Depending on the environment where the pack batteries and BMS are installed, there is the possibility of missing values due to sensor malfunctions. If the acquired data from sensor includes missing values, SOC estimation cannot perform, so it is necessary to impute missing values with other numerical values.

Fig. 6 shows the missing value imputation mechanism we implemented. In the missing value imputation process, the

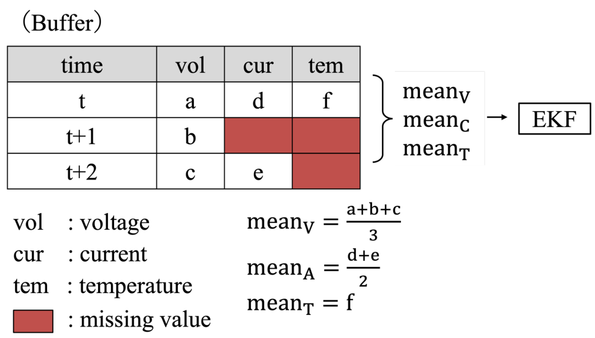
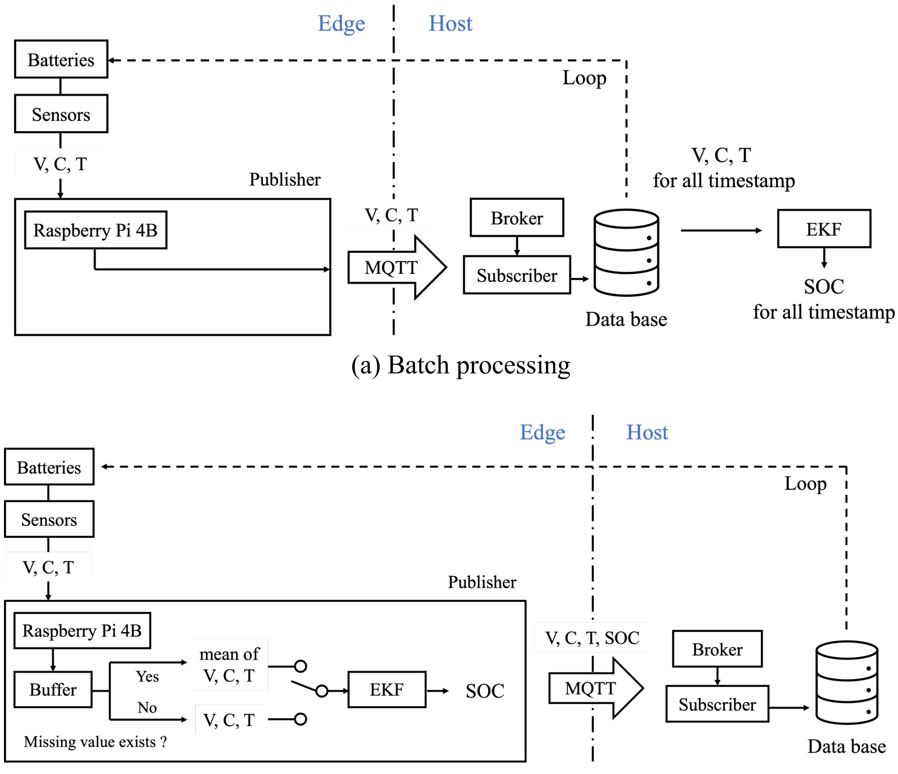


Fig. 6. The missing value imputation mechanism



(a) Batch processing

(b) Real-time processing (proposed processing)

Fig. 7. The comparison of batch processing and real-time processing

average of each data ($mean_V$, $mean_C$, $mean_T$) are calculated from the data recorded in the buffer for a few seconds, and missing values are imputed by the average. The calculation of the average is initially performed and then repeated every second. However, when calculating the average after accumulating a certain amount of data in the buffer, a time lag occurs between the actual measurement and the acquired data. Therefore, in this development, missing value imputation processing is performed only when the data acquired at the current time is missing. If the data is not missing, the acquired data is used directly for SOC estimation.

E. Batch processing and real-time processing

Fig. 7 shows the comparison of the batch processing and the real-time processing. In the batch processing, as shown in Fig. 7 (a), the SOC estimation is performed on the host side after a series of processes, including sending and recording voltage (V), current (C), and temperature (T) data acquired at each time to the database are completed. In the proposed real-time processing, as shown in Fig. 7 (b), both data acquisition and SOC estimation are performed on the edge side. The data including SOC for each timestamp is sent to the database and recorded. In this development, as shown in Fig. 2, real-time processing including SOC estimation on the edge side is adopted. Therefore, compared to the batch processing, dynamic state management of the battery pack becomes possible, and it can be applied to dynamic diagnosis of deterioration and other applications.

III. Experiments and Results

Fig. 8 shows the voltage and current in the charge and discharge patterns used in this development. Fig. 9 shows the actual data of terminal voltage, terminal current, and temperature of each battery measured by the BMS and the estimated SOC results using EKF from these data. Additionally, Fig. 10 shows the actual operating screen of the Publisher. From Fig. 8 and 9, it is confirmed that the BMS can measure voltage and current similar to the charge and discharge pattern, which indicates that the BMS sensing is performing correctly. The SOC was dynamically observed using the EKF on proposed BMS architecture. Further verification will be conducted for the accuracy of each sensor in the future.

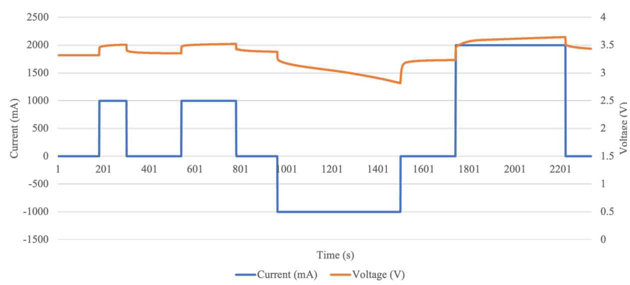
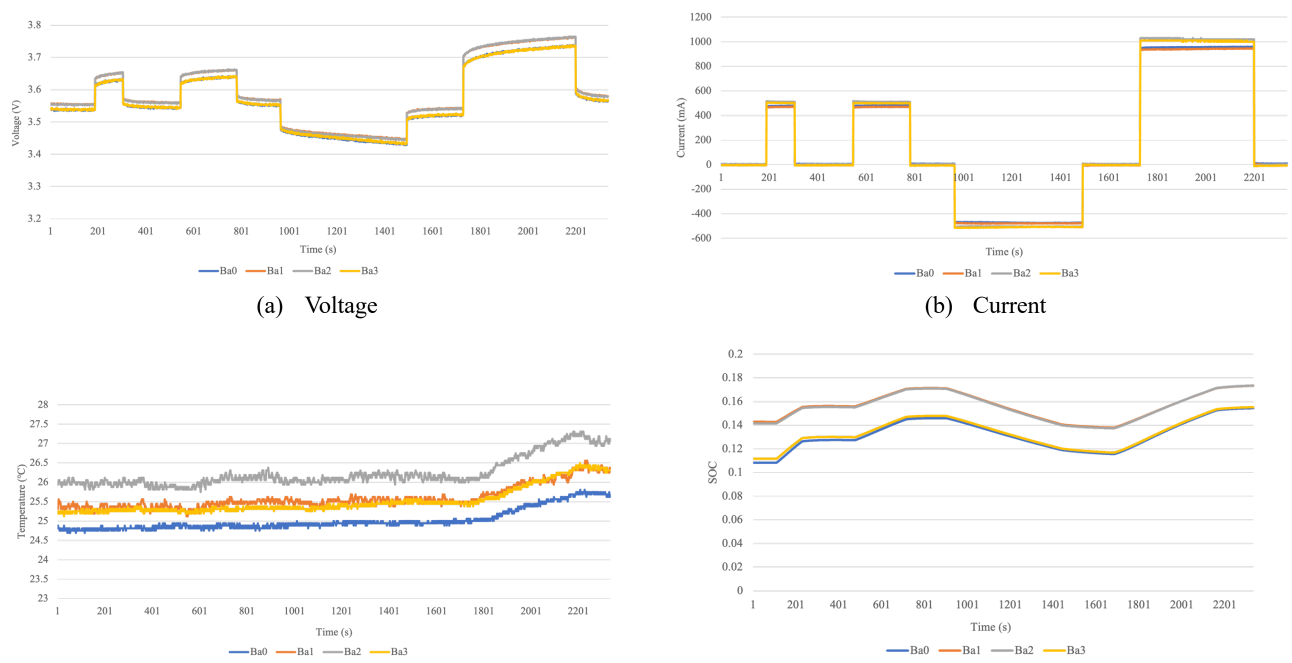


Fig. 8. Charge and discharge pattern of voltage and current



(a) Voltage (b) Current

(c) Temperature (d) SOC

Fig. 9. Measured data and estimated SOC of each battery by the BMS

```
----------------------------
Time: 2023-12-22T17:42:52.762326
Ba0 -> Voltage:  3.47852,       Current: 486.40625,     Temperature: 21.87500,  SOC:  0.04788
Ba1 -> Voltage:  3.67207,       Current: 468.59375,     Temperature: 22.37500,  SOC:  0.29658
Ba2 -> Voltage:  3.67344,       Current: 514.53125,     Temperature: 22.43750,  SOC:  0.28757
Ba3 -> Voltage:  3.47598,       Current: 494.68750,     Temperature: 22.43750,  SOC:  0.08386
----------------------------
Time: 2023-12-22T17:42:53.763417
Ba0 -> Voltage:  3.47656,       Current: 485.15625,     Temperature: 21.87500,  SOC:  0.04757
Ba1 -> Voltage:  3.67344,       Current: 467.96875,     Temperature: 22.43750,  SOC:  0.29704
Ba2 -> Voltage:  3.67480,       Current: 514.21875,     Temperature: 22.43750,  SOC:  0.28800
Ba3 -> Voltage:  3.47617,       Current: 494.53125,     Temperature: 22.43750,  SOC:  0.08387
----------------------------
Time: 2023-12-22T17:42:54.764549
Ba0 -> Voltage:  3.47773,       Current: 485.93750,     Temperature: 21.87500,  SOC:  0.04808
Ba1 -> Voltage:  3.67480,       Current: 468.12500,     Temperature: 22.37500,  SOC:  0.29713
Ba2 -> Voltage:  3.67500,       Current: 510.93750,     Temperature: 22.43750,  SOC:  0.28801
Ba3 -> Voltage:  3.47637,       Current: 495.00000,     Temperature: 22.37500,  SOC:  0.08385
----------------------------
```

Fig. 10. Actual operation screen of the Publisher

IV. Conclusions and Discussions

We reported on the development of a BMS with dynamic SOC estimation and communication function using MQTT as proposed development.

We developed a functional verification platform for IoT-type BMS and confirmed its operation. To improve the efficiency of function and algorithm development, the edge-side system used a Raspberry Pi 4B as computer, Python as programming language, and I2C as communication protocol between sensors and Raspberry Pi 4B. Although Raspberry Pi 4B is not suitable for implementation on the edge side due to its high power consumption, it enhances the functional design completion level as a design platform. Additionally, we implemented the SOC estimation function using the Kalman Filter on the edge side in real-time processing. Processing on the edge side reduced the burden on the host side and enabled real-time state monitoring.

Looking ahead, for the long-term use of batteries, monitoring the SOC alone is insufficient; it is essential to monitor the deterioration state of the battery. RLS is a method for estimating the internal parameters of batteries, and by estimating the internal resistance of the battery using RLS, effective diagnosis of deterioration is possible [9]. With the dynamic estimation function in the BMS we proposed, dynamic deterioration diagnosis using RLS is enable. Moreover, by dynamically estimating the internal parameters of each battery using RLS, it is possible to dynamically configure circuit models for battery internal state estimation using estimation techniques such as EKF [10].

NN is effective for more complex internal state estimation, such as estimating remaining life and residual value of batteries. However, NN requires a large amount of charge-discharge data for estimation and diagnosis. Also, it consumes a lot of memory and has long calculation times, so it needs to be implemented on the host-side computer shown in Fig. 1. Because of these factors, the proposed BMS with communication function using MQTT is believed to be effective for making big data for NN estimation and diagnosis.

Acknowledgements

This paper is based on results acquired from a project, JPNP10020, commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

References

[1] P. Weicker, “A Systems Approach to Lithium-Ion Battery Management,” Artech House, 2013.

[2] K. Liu, K. Li, Q. Peng, and C. Zhang, “A brief review on key technologies in the battery management system of electric vehicles,” Front. Mech. Eng., vol.14, no.1, pp.47-64, 2019.

[3] S. Irikuchi, H. Ito, and M. Fukui, “Storage battery open-circuit voltage estimation method using recurrent neural network and its evaluation”, The Committee of Battery Technology, vol.62, Nov.2021.

[4] Y. Harada, L. Lin, and M. Fukui, “Consideration of an Internal Impedance Estimation Method for Energy Storage Systems Using Recurrent Neural Networks”, Kansai-section Joint Convention of Institutes of Electrical Engineering, vol.2020, Nov.2020.

[5] S. Yamazi, F. Saito, K. Suzuki, and M. Fukui, “Consideration of Anomaly Detection Method for Internal Gas Generation in Lithium-ion Batteries”, Technical Committee on Circuits and Systems, Dec.2022.

[6] G. L. Plett, “Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 2. Modeling and identification,” J. Power Sources, vol.134, no.2, pp.262-276, 2004.

[7] M. Fukui, M. Arima, “Smart Grid and Battery Management Technology”, CORONA Publishing company, Tokyo, 2022.

[8] L. Lin, N. Kawarabayashi, M. Fukui, S. Tsukiyama, and I. Shirakawa, "A practical and accurate SOC estimation system for lithium-ion batteries," in Proc. 17th International Meeting on Lithium Batteries, p. 632, 2014.

[9] L. Lin and M. Fukui, "A stable equivalent circuit identification algorithm for Li-ion batteries," in Proc. 22nd Workshop on Synthesis and System Integration of Mixed Technologies, 2019.

[10] L. Lin, M. Arima, and M. Fukui, “Simultaneous and Stable Estimation Method for Circuit Parameters and Remaining Capacity in Lithium-Ion Batteries”, The Institute of Electronics, Information and Communication Engineers Transactions, vol.J104-B, no.3, Mar.2021.

[11] W. He, N. Williard, C. Chen, and M. Pecht, “State of charge estimation for Li-ion batteries using neural network modeling and unscented Kalman filter-based error cancellation,” Int. J. Electrical. Power & Energy Systems., vol.62, pp.783-791, 2014.

[12] S. Irikuchi, H. Ito, Y. Nishimura, and M. Fukui, “Long short-term memory-networks for accurate open circuit voltage estimation of Li-ion batteries,” in Proc. RISP International Workshop on Nonlinear Circuits, Communications and Signal Processing, 2022.