ESP32-Based External Storage Data Logger: A Cost Effective Solution for Scalable Industrial Data Management

Paradon Boonmeeruk and Kiattisak Wongsopanakul*

Department of Electrical & Biomedical Engineering, Prince of Songkla University, Songkhla, Thailand

Abstract: Data loggers play a crucial role in industrial operations by automating data recording processes and reducing the reliance on manual input, thereby improving operational efficiency. However, traditional data loggers often encounter scalability challenges when managing increasing data volumes, leading to high operational costs. To address these limitations, this paper presents the development of an economical and efficient external storage data logger system based on the ESP32 microcontroller and a simple Micro SD card. The proposed system supports data storage capacities up to 32 GB and is fully compatible with the Siemens S7-1200 industrial controller, making it suitable for extensive industrial applications. The system incorporates robust data backup mechanisms to ensure data integrity during device malfunctions, such as power failures or storage errors, thereby preventing data loss. By leveraging the cost-effective and versatile ESP32 microcontroller, the system achieves a scalable and reliable data logging solution while significantly reducing operational costs. Experimental results demonstrate the system's capability to meet industrial data management requirements while offering a low-cost alternative to conventional solutions. This research highlights the potential of microcontroller-based systems to enhance industrial data logging efficiency and reliability, contributing to more sustainable and cost-effective industrial automation practices.

***Keywords* — Data Logger, ESP32 Microcontroller, External Storage**

I. Introduction

A data logger is a device used to record data from measurement equipment or log the operation of machines within production processes. These devices typically consist of two primary components: a processing unit (microprocessor) and memory. In some cases, an integrated sensor is included to measure the surrounding environment. Besides recording data, data loggers significantly reduce the reliance on manual data entry by operators due to their ability to automatically manage and store data [7]. However, such devices often face limitations in memory capacity, restricting their ability to accommodate increasing data volumes [3]. For instance, the Siemens S7-1200 PLC provides approximately 4 MB of load memory, while devices like the Simatic HMI MTP700 offer 12 MB of internal memory. Expanding memory capacity to support more extensive datasets generally incurs high operational costs. This challenge has driven research into alternative, cost-effective solutions for data logging systems [5]. introduced the ALog system, an open-source data logger based on an Arduino microcontroller, capable of collecting environmental data such as weather conditions and glacier melting. Operating within a temperature range of -30°C to 60°C, the system uses low-energy alkaline batteries, ensuring energy efficiency. It also supports memory expansion up to 32 GB, enabling efficient data collection from connected measurement devices [2]. presented a data logger system leveraging Internet of Things (IoT) technology and embedded systems to create a scalable database for climate parameters. This system uploads extensive climate datasets seamlessly without requiring hardware modifications, ensuring flexibility and scalability [4]. developed a data logger system based on the Arduino Uno R3 microcontroller, integrating an SD card module and RTC module. Designed to monitor wind turbine systems, the system achieves high calibration accuracies of 99.99% for voltage measurements and 99.98% for current measurements. These advancements highlight the potential of microcontroller-based data loggers to reduce costs and enhance data handling capabilities. Building on this foundation, this article introduces an external storage data logger system that interfaces with Siemens S7-1200 industrial controllers. The proposed system addresses limitations in conventional data loggers, offering scalable storage, cost reduction, and enhanced functionality for industrial applications.

The manuscript received Sept. 9, 2024; revised Dec. 16, 2024; accepted Dec. 21, 2024; Date of publication Dec. 30, 2024

*Corresponding author: Kiattisak Wongsopanakul, Dept. of Electrical & Biomedical Engineering, Prince of Songkla University, Songkhla, Thailand, (E-mail: Kiattisak.w@psu.ac.th)

II. Related Theories on Data Logger Systems

the use of the Data Logger system [1], the overall structure of the Data Logger system is depicted in Fig. 1, Initially, environmental or physical quantities measured by equipment, such as temperature and relative humidity, are loaded into the Data Log Instruction section. The CPU then converts the stored data into CSV format and loads it into the Load Memory within the CPU. For additional data storage, the system can manage and store data on external memory (Memory Card). Data access to the CSV files on the device's memory is facilitated by a web server built into the CPU, allowing operators to access and download data [1].

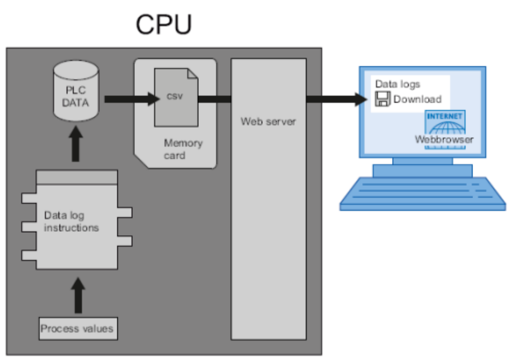

Fig. 1. Overview of the PLC Siemens Data Logger System

Direct data storage in a processor's memory, as previously described, exhibits significant limitations in terms of access flexibility and scalability. Data retrieval often requires access via a web server hosted by the processor, which depends on referencing the device's IP address. Additionally, expanding the processor’s memory to accommodate increased data from field devices is constrained by high costs and limited scalability. To address these challenges, implementing a data logger within a Supervisory Control and Data Acquisition (SCADA) system offers a viable alternative. As referenced in Siemens WinCC SCADA system documentation Fig. 2, the SCADA system functions as a centralized hub for collecting data from field devices, such as controllers or production line machinery, through industrial communication protocols. This data is subsequently stored in a structured SQL database, providing enhanced storage scalability and accessibility. The SCADA system's database supports two main data storage formats: Circular Log: This format deletes the oldest data entries in the logger when the specified storage limit is reached, enabling continuous logging of new data. Segmented Circular Log: In this format, data is distributed across multiple loggers. When the first logger reaches its capacity, data is pushed sequentially to the next logger, and the oldest data in the final logger is deleted to make space for new entries. The data writing mechanisms for these formats are depicted in Fig. 3 [6]. While SCADA-based data logging addresses the limitations of processor-based systems, it introduces its own challenges, primarily the high implementation and expansion costs required to support additional field devices.

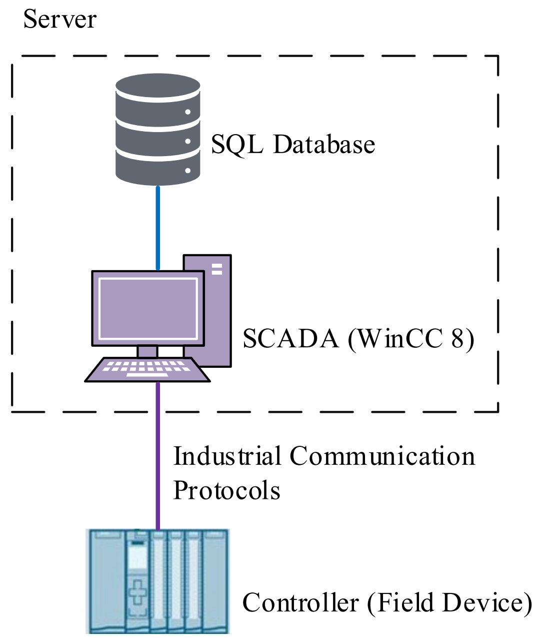

Fig. 2. Overview of SCADA-Based Data Logger System with SQL Database

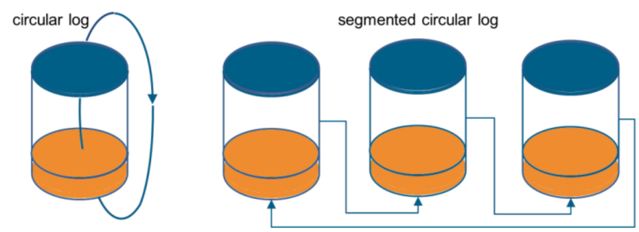

Fig. 3. Data Writing Format to SQL Database in the WinCC SCADA System

III. Design of the External Storage Data Logger System for PLC Siemens S7-1200

Fig. 4 illustrates the operation of the External Storage Data Logger system, which is seamlessly integrated with the PLC Siemens S7-1200 and ESP32 modules. The system is designed to efficiently manage and store critical industrial data through a combination of advanced processing and communication protocols. The first component, the PLC Siemens S7-1200, is tasked with reading DC voltage data within the range of 0-10 Vdc through its analog input port. This data is then scaled to a numerical range of 0-27648, adhering to the specified sampling time. The PLC is further responsible for configuring the memory address for data reading and writing, utilizing the Modbus TCP communication protocol to ensure smooth interfacing with the external memory of the ESP32 Server Node.

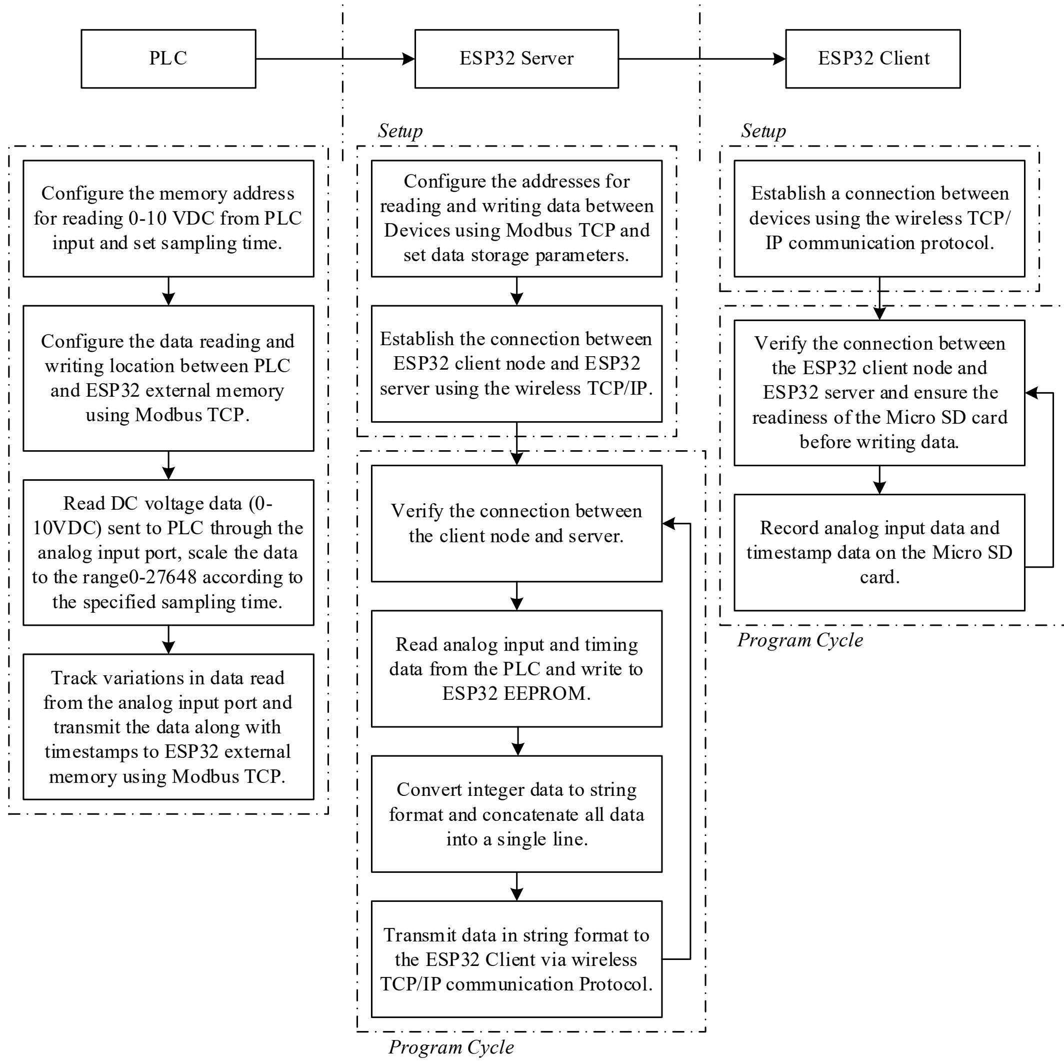

Fig. 4. Diagram illustrating the operation of the External Storage Data Logger system with PLC Siemens S7-1200.

In the ESP32 Server Node, after establishing a connection with the PLC using the Modbus TCP/IP communication protocol, the control program commands the Server Node to retrieve data from the PLC based on the specified data location shared between the Modbus Client and Modbus Server. Once the data is retrieved, it is stored in the Server Node's EEPROM to safeguard against data loss during interruptions, such as power failures during operation. Following the data storage process, the data is reorganized into a standardized format, and its type is converted to text, as depicted in Fig. 5. The final step in the Server Node involves checking the connection with the Client Node via the wireless TCP/IP communication protocol. If the connection is stable and uninterrupted, the organized data is transmitted to the Client Node for further processing, specifically writing the data to the MicroSD card.

In the ESP32 Client Node, upon completing the connection verification process with the Server Node, the system performs a readiness check on the MicroSD card to ensure it is available for data writing. Once the MicroSD card is confirmed to be operational the Client Node downloads the data transmitted by the Server Node and writes it to the MicroSD card for storage. After completing the data logging process, the Client Node terminates its connection with the Server Node and enters a standby state, awaiting the next data transmission.

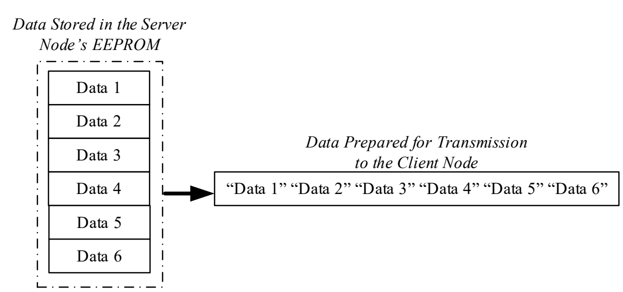

Fig. 5. Data Arrangement Format in Server Node for Transmission to Client Node and Storage on MicroSD Card

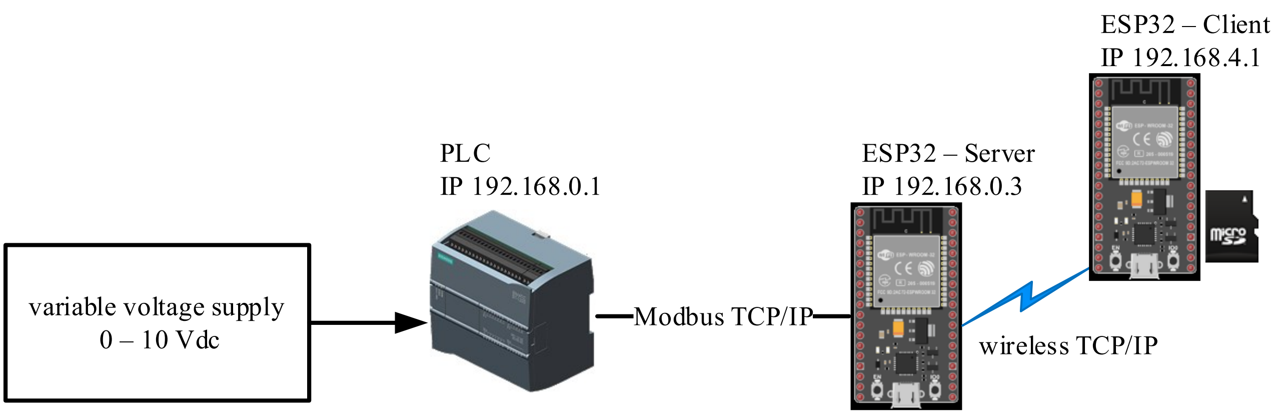

Fig. 6. Schematic Representation of External Storage Data Logger System Testing

IV. PROCEDURE FOR TESTING THE EXTERNAL STORAGE DATA LOGGER SYSTEM

Fig. 6 presents an overview of the External Storage Data Logger system testing. The test simulates environmental changes by adjusting a DC voltage within the 0-10 Vdc range. This process creates a cycle of analog input data with values between 0 and 27648. The Modbus TCP communication protocol transmits time-stamped analog input data for storage in the ESP32 Server Node's EEPROM as a backup before writing to the MicroSD card. Each data set written to the ESP32 Server Node's EEPROM has a size of 28 bytes and a format detailed in Fig. 8. For writing data to the MicroSD card, the system utilizes the internally installed ESP32 CAM. The MicroSD card is configured as an ESP32 Client, enabling wireless TCP/IP communication with the ESP32 Server Node. This allows the logging data stored in the ESP32 Server Node's EE-PROM to be transferred to the MicroSD card, as illustrated in Fig. 7.

Testing the ESP32 Server Node's backup system, designed for data redundancy in case of offline status or Client connection issues, involves supplying a logic "1" input to port D15 (Fig. 9). This action triggers the ESP32 Server Node to enter a mode where it transmits data from the EEPROM to the ESP32 Client while in Setup mode.

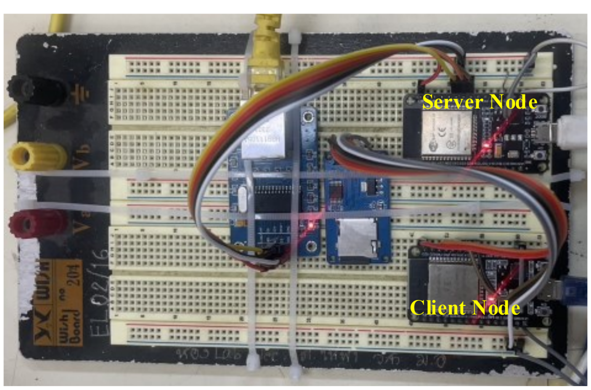

Fig. 7. Hardware Configuration: ESP32 Client Node and Server

Date (Int)	Month (Int)	Year (Int)	Hr (Int)	Sec (Int)	Analog Input (Int)
4 Bytes	4 Bytes	4 Bytes	4 Bytes	4 Bytes	4 Bytes

Fig. 8. Data Format for ESP32 Client Node EEPROM

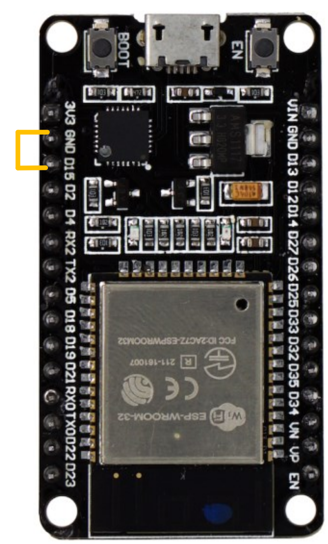

Fig. 9. Activating ESP32 Server Node for EEPROM Data Transmission to ESP32 Client

V. EVALUATION OF EXTERNAL STORAGE DATA LOGGER SYSTEM TEST RESULTS

The test results of the External Storage Data Logger system are illustrated in Fig. 10A and Fig. 10B. In Fig. 10A, the data read by the ESP32 Server Node from the PLC using the Modbus TCP/IP communication protocol is shown. This data is first recorded in the Server Node's EEPROM and subsequently arranged into structured data sets for transmission to the Client Node. Fig. 10B highlights the data received by the Client Node from the Server Node via the Wireless TCP/IP communication protocol. The Client Node records this data to the MicroSD card. Under normal operating conditions, once the data has been successfully recorded to the MicroSD card, the Client Node displays a confirmation message, as depicted in Fig. 10B.

Date	Month	Year	Hour	Minute	Second	Data
12	12	2024	19	42	44	6296
12	12	2024	19	42	44	6757
12	12	2024	19	42	44	5102
12	12	2024	19	43	8	4661
12	12	2024	19	43	8	4645

(A) Data from ESP32- Server Node

```
12     12     2024     19     42     44     6296
Appending to file: /Logger.txt
File appended successfully
12     12     2024     19     42     44     6757
Appending to file: /Logger.txt
File appended successfully
12     12     2024     19     42     44     5102
Appending to file: /Logger.txt
File appended successfully
12     12     2024     19     43     8     4661
Appending to file: /Logger.txt
File appended successfully
12     12     2024     19     43     8     4645
Appending to file: /Logger.txt
File appended successfully
```

(B) Data from MicroSD card on ESP32 – Client Node

Fig. 10. Data Logger System Operation Under Normal Conditions (Online Mode)

The External Storage Data Logger system supports the generation of files compatible with data analysis software, such as Microsoft Excel. Logger data stored on the MicroSD card can be processed and analyzed using these tools. Specifically, when the recorded data is imported into Microsoft Excel, the system generates a .CSV file format containing the data, as illustrated in Fig. 11. This feature enhances the usability of the logger data for further analysis and reporting.

Date	Month	Year	Hour	Minute	Second	Data
12	12	2024	19	42	44	6296
12	12	2024	19	42	44	6757
12	12	2024	19	42	44	5102
12	12	2024	19	43	8	4661
12	12	2024	19	43	8	4645

Fig. 11. .CSV File Generated by the External Storage Data Logger System for Use with Microsoft Excel

The graph in Fig. 12 represents data retrieved from the EEPROM of the Server Node, which includes the data received from the PLC before transmission to the Client Node for recording on the MicroSD card. A comparison was made between this data and the data stored in the MicroSD card, representing the recorded output of the Data Logger system. Upon analyzing the two graphs, it was observed that the data points in both are positioned identically, and the patterns of change in the graph characteristics align precisely. This consistency between the data retrieved from the Server Node's EEPROM and that recorded on the MicroSD card demonstrates the system's capability to preserve data integrity throughout the data logging process. The results confirm that the proposed External Storage Data Logger system operates in compliance with standard data logging principles, ensuring accurate data recording and reliable performance in industrial applications.

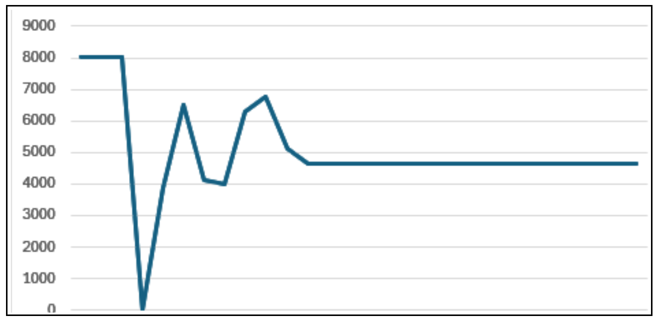

Fig. 12. Analog Input Data Retrieved from PLC Before Storage on MicroSD Card (Server Node Data)

The test results demonstrate the ability of the Server Node to back up data to its EEPROM, ensuring data integrity in the event of a malfunction. When the Server Node enters the data retrieval mode from its EEPROM, as shown in Fig. 9, the data recorded prior to the malfunction can be accessed. Upon restarting the Server Node, the pre-failure data is successfully downloaded, as illustrated in Fig. 13. Once the Server Node reestablishes a connection with the Client Node, all retrieved data is transmitted to the Client Node for storage on the MicroSD card, as depicted in Fig. 14. These results confirm the system's capability to back up data during malfunctions, safeguarding against data loss. However, this capability is constrained by the size of the Server Node EEPROM, which has a maximum capacity of 28 bytes. Consequently, the backup functionality is limited to handling data sets up to 28 bytes in size. Future enhancements could focus on increasing the EEPROM capacity or integrating external storage solutions to improve the robustness of the data backup mechanism.

```
    Date Month Year Hour Minute Second Data
-> 12 12 2024 19 49 29 4643
-> 12 12 2024 19 49 29 4641
-> 12 12 2024 19 49 40 4643
-> 12 12 2024 19 49 42 4641
-> 12 12 2024 19 50 2 4642
-> 12 12 2024 19 50 8 4641
```

Fig. 13. Data Retrieved from Server Node's EEPROM Prior to System Malfunction

```
-> 12    12    2024    19    49    29    4643
-> Appending to file: /Logger.txt
-> File appended successfully
-> 12    12    2024    19    49    29    4641
-> Appending to file: /Logger.txt
-> File appended successfully
-> 12    12    2024    19    49    40    4643
-> Appending to file: /Logger.txt
-> File appended successfully
-> 12    12    2024    19    49    42    4641
-> Appending to file: /Logger.txt
-> File appended successfully
-> 12    12    2024    19    50    2    4642
-> Appending to file: /Logger.txt
-> File appended successfully
-> 12    12    2024    19    50    8    4641
-> Appending to file: /Logger.txt
-> File appended successfully
```

Fig. 14. Data Transferred from Server Node's EEPROM to MicroSD Card Following System Recovery

VI. Conclusion

The ESP32-based data logging system offers a cost effective solution with low-cost memory expansion, demonstrated using 32 GB of memory, to accommodate growing data volumes. It also provides critical data backup

functionality, storing essential data on a MicroSD card to prevent loss in case of ESP32 Server Node malfunctions, such as power failures or data saving issues.

For future work, several enhancements are identified to broaden the system's capabilities. These include improving scalability and performance to support larger data volumes and more complex environments, along with implementing robust encryption measures to ensure data security. Integrating real-time analytics could enable predictive maintenance, enhancing operational efficiency. Additionally, further integration with industrial systems like SCADA and MES will provide a more comprehensive solution for industrial automation and IoT applications. Improving energy efficiency, particularly in remote or resource constrained environments, and developing a more intuitive user interface for easier system configuration and monitoring are also important areas for future research. These advancements will enhance the system's robustness, sustainability, and usability, making it better suited for a wide range of industrial applications.

References

[1] Siemens AG, "SIMATIC S7-1200 Data Logging," Siemens Industry Online Support, [Online]. Available:https://cache.industry.-siemens.com/dl/files/156/64396156/att_992804/v1/64396156_S71200_DataLogging_DOC_v3d0d1_en.pdf. Accessed: Feb. 12, 2024.

[2] J. Mabrouki, M. Azrour, D. Dhiba, Y. Farhaoui, and S. El Hajjaji, "IoT-based data logger for weather monitoring using arduino-based wireless sensor networks with remote graphical application and alerts," Big Data Mining and Analytics, vol. 4, no. 1, pp. 25-32, 2021.

[3] D. L. Wilson, K. N. Williams, and A. Pillarisetti, "An integrated sensor data logging, survey, and analytics platform for field research and its application in HAPIN, a multi-center household energy intervention trial," Sustainability, vol. 12, no. 5, p. 1805, 2020.

[4] S. Murti, P. Megantoro, G. D. B. Silva, and A. Maseleno, "The design and analysis of DC electrical voltage-current datalogger device implemented on wind turbine control system," Journal of Robotics and Control (JRC), vol. 1, no. 3, pp. 75-80, 2020.

[5] A. D. Wickert, C. T. Sandell, B. Schulz, and G.-H. C. Ng, "Open-source Arduino-compatible data loggers designed for field research," Hydrology and Earth System Sciences, vol. 23, no. 4, pp. 2065-2076, 2019.

[6] Siemens AG, " Logging Process Values and Alarms with WinCC (TIA Portal)" Siemens Industry Online Support, [Online]. https://support.industry.siemens.com-/cs/document/109746939/logging-process-values-and-alarms-with-wincc-(tia-portal) Accessed: JUN. 12, 2019.

[7] Z. W. Siew, C. H. Wong, S. E. Tan, H. P. Yoong, and K. T. K. Teo, "Design and development of a tablet based real time wireless data logger," in 2012 IEEE Global High Tech Congress on Electronics, 2012: IEEE, pp. 111-116.

Paradon Boonmeeruk, He received a B.S. degree in Electronics Engineering Technology Instrumentation and Control from King Mongkut's University of Technology North Bangkok in 2020. He is currently pursuing a master's degree in Electrical & Biomedical Engineering at the Faculty of Engineering, Prince of Songkla University, Songkhla, Thailand.

Kiattisak Wongsopanakul, He received a B.S. degree in Electrical Engineering from the Faculty of Engineering, Prince of Songkla University, Songkhla, Thailand, an M.S. degree in Electrical Engineering from the New York Institute of Technology, USA, and an M.S. degree in Business Administration from the Faculty of Management Sciences, Prince of Songkla University, Songkhla, Thailand. He earned his Ph.D. degree in Electrical Engineering from Wayne State University, USA. At present, Dr. Kiattisak Wongsopanakul serves as a lecturer in Electrical & Biomedical Engineering at Prince of Songkla University, Songkhla, Thailand.