DOI: 10.22146/jnteti.v14i4.21506

PLC Control System Implementation with IoT Based Industrial SCADA Development

Hendro Priyatman[1], Seno D. Panjaitan[1], Supriono[1], Muh. Revaldi Frizky[1]

[1] Electrical Engineering Study Program, Department of Electrical Engineering, Faculty of Engineering, Universitas Tanjungpura, Pontianak, Kalimantan Barat 78124, Indonesia

[Received: 3 July 2025, Revised: 19 September 2025, Accepted: 11 November 2025]
Corresponding Author: Hendro Priyatman (email: hendro.priyatman@ee.untan.ac.id)

ABSTRACT — Control system challenges such as remote access and long-distance command execution can be addressed using supervisory control and data acquisition (SCADA) systems. SCADA enables real-time monitoring and control of industrial processes, allowing remote decision-making, improving operational efficiency, and accelerating response to issues. It also enhances data transparency, enabling engineers to detect and analyze failures more effectively. However, SCADA implementation often involves high costs and requires skilled personnel for maintenance. To address these limitations, this research developed an affordable SCADA system tailored for small and medium-sized industries. The system used a programmable logic controller (PLC) for control and a human machine interface (HMI) on the remote terminal unit (RTU). Data from the RTU were sent to a master terminal unit (MTU), which was independently developed and enabled remote monitoring and control via industrial internet of things (IIoT) technology using the message queueing telemetry transport (MQTT) protocol. The system monitored two variables: temperature (RTU 1) and rotational speed (RTU 2), aiming to measure communication time and data accuracy. Testing showed sensor accuracy of 98.6% for RTU 1 and 100% for RTU 2, with 100% communication accuracy between layers. The average real-time communication duration was 0.324 s, demonstrating high efficiency and low latency. These results indicate that an IoT-based SCADA system is reliable, cost-effective, and suitable for small to medium industrial applications. It is also easy to replicate and reconfigure, making it a practical solution for broader industrial adoption.

KEYWORDS — SCADA, Programmable Logic Controller, Industrial Internet of Things, Remote Terminal Unit, Human Machine Interface, Message Queueing Telemetry Transport.

I. INTRODUCTION

The development of technology in our daily lives has significantly affected various sectors of life, such as the production process in the manufacturing industry. Massive technological advances in the industrial sector itself aim to increase the efficiency and effectiveness of the production process [1], which is often referred to as the Industrial Revolution. Currently, the Industrial Revolution 4.0, commonly referred to as Industry 4.0, is underway, starting with the implementation of the integration of artificial intelligence, cloud storage technology, and industrial internet of things (IIoT) technology [1], [2].

Industry 4.0 through IIoT and cloud storage for databases is a convergence between advances in information and communication technology, with the production sector in industrial areas [3]. The use of information and communication networks using IIoT between systems themselves makes the automation control system more sophisticated because it combines various controllers, sensors, and actuators in a single system that influences each other and can be controlled from a central terminal, like supervisory control and data acquisition (SCADA) [4]. However, Industry 4.0 is certainly a challenge for industries in Indonesia. IoT is defined as a group of infrastructures, interconnecting connected objects and allowing their management, data mining and the access to data they generate where connected objects are sensor(s) and/or actuator(s) carrying out a specific function that are able to communicate with other equipment [5], especially small and medium industries. The smaller the company, the more at risk of becoming a victim, rather than receiving benefits from this industrial revolution [6], [7]. This is because small and medium companies or start-up are less prepared to face Industry 4.0, one of which is the problem of funding capacity in procuring SCADA, even though this technology is a significant aspect in facing the Industrial Revolution 4.0. In addition, the intelligence of underlying equipment should be enhanced in order to meet the requirements of IoT. In the network layer, IIoT should support new protocols and new data format with high flexibility and scalability, whereas the industrial wireless sensor networks (IWSNs) bring new opportunities for development of industrial network [8].

Therefore, it is necessary to conduct research to develop a customized and open-source SCADA system at an affordable price for small and medium-sized industries. Considering the purpose of SCADA used at the industrial level, this study uses more than one system consisting of a programmable logic controller (PLC) as a controller, a human machine interface (HMI) as an interface and means to exchange data to the server.

Figure 1 exhibits the main architecture of SCADA. The sensor serves as a source for data collection, while the actuator can be controlled by the user where researchers develop a desktop application that can be used on a computer as a central terminal for data collection and a control center for the SCADA system.

The IIoT technology is an ecosystem of devices, sensors, applications, and network devices associated with collecting, monitoring, and analyzing data from industrial operations. Table I presents the comparison of IoT and IIoT. In technical terms, IoT and IIoT technology are closely related. However, IIoT is specifically designed for industrial automation purposes.

Previous study, the study aimed to develop an IoT device to replace RTU and integrate it with Amazon Web Services to

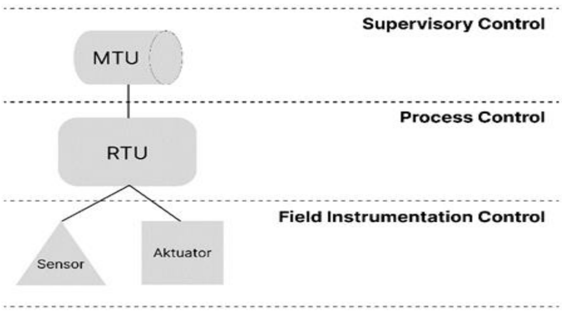

Figure 1. Main architecture of SCADA systems.

TABLE I
COMPARISON OF IOT AND IIOT

Implementation	IoT	IIoT
Application	Supports optimization of energy consumption	Maximizing efficiency and seamless workflow
Orientation	Human-centered	Machine-centered
Reliability	Flexible latency-dependent	Designed to withstand low latency
Use Case	Small-scale automation	Large-scale automation and monitoring
Connectivity	Ad-Hoc (infrastructure cannot be tolerated, nodes may change)	Structured (nodes are fixed, centralized network management)
Criticality Level	Not strict	Strict (time, reliability, security, privacy)
System Size	Small to medium	Large to very large

replace the MTU [9]. As a result, in this study, the system was declared safe, lightweight, inexpensive, readily available, and easy to reconfigure. Another study opened the concept for this study to use a multiple PLC multiple HMI model, where two PLCs were used as RTUs, and two PCs were used as HMIs as MTUs [10]. The results showed that the designed system achieved good synchronization between the monitoring PC and the PLC, allowing the PC to monitor the plant. Furthermore, prior study aimed to design and implement a low-cost open-source SCADA system using the local IoT platform Thinger.IO server as MTU and the ESP32 thing microcontroller as RTU [11]. The local Thinger.IO server can store, monitor, and perform remote control. The results showed that the concept of a database in which data can be stored on the Thinger.IO server. Afterward, prior study aimed to design an IoT gateway allowing identical devices, based on Modbus RTU communication, to integrate with the IoT network [12]. This study brought the MQTT protocol to the SCADA system. The subsequent study produced a SCADA-based control panel simulator that used ESP32 as a controller and the Blynk internet as an intermediary to connect humans to the simulator [13]. The results showed that the data were successfully sent 100%, provided that the internet network was in good condition.

Previous studies have utilized hardware components that are not up to industry standards or do not use MQTT protocol for communication. In this study, standard PLC and HMI were used, ensuring that the developed tool was sufficiently capable for industrial application.

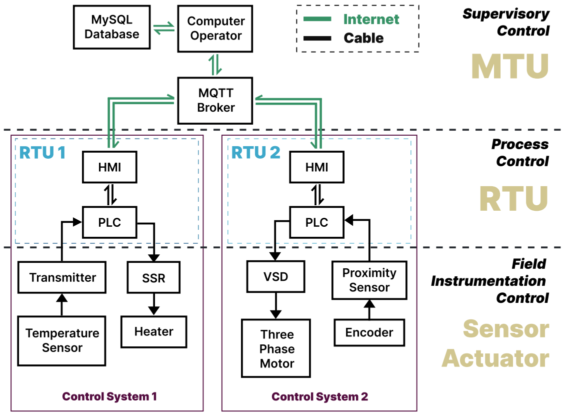

Figure 2. Block diagram of system.

II. SYSTEM DESIGN

The system design in this study utilized digital inputs and outputs with speed and heater sensors, analog inputs with temperature sensors, and Modbus communication. Each RTU was connected to the internet and communicated with the MQTT broker. Figure 2 shows the overall block diagram of system design, connected via internet and cable, which consists of the supervisory control layer, process control layer, and field instrumentation control layer in the designed SCADA system [14].

The wireless communication network design in the development of SCADA research based on IIoT used two types of protocols, the MQTT protocol between MTU and RTU and the wireless local area network (WLAN) on the MTU for SCADA applications that exchanged data with the database. In addition, between the control components in the system, there was communication with cables using the Modbus protocol type RS-485 for PLC communication with VSD to control the motor and the RS-232 type for data exchange between PLC and HMI.

A. CONTROL SYSTEM DESIGN

As shown in Figure 1, RTU 1 and RTU 2 served as the brains of their respective control systems, as they were directly connected to the field instruments. The control systems on both RTUs were managed by their respective PLCs, which handled different control system processes. The HMI on each RTU was used as an interface, and the HMI supported internet connection so that it could be used to communicate with the MQTT broker.

Figure 3 shows control system block diagram of each RTUs. The variable temperature output (*temp PV*) and the desired temperature reference value (*temp SV)* from and for RTU 1 were measured in degrees Celsius (°C). Then, the rotational speed output (*speed PV*) and the desired speed reference value (*speed SV*) were from and for RTU 2, measured in revolutions per minute (rpm). These values were used to process and obtain additional data needed for the study. Additional data were viewed for accuracy and in real-time to determine the percentage of data accuracy and its delivery delay.

B. HARDWARE DESIGN

Hardware design serves to facilitate the process of wiring flow and layout of the components used. The outline of the

 p-ISSN 2301–4156 | e-ISSN 2460–5719

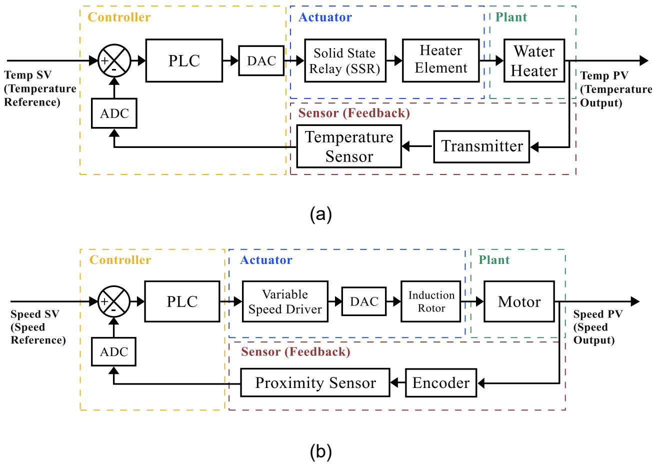

Figure 3. Control system block diagram, (a) RTU 1 and (b) RTU 2.

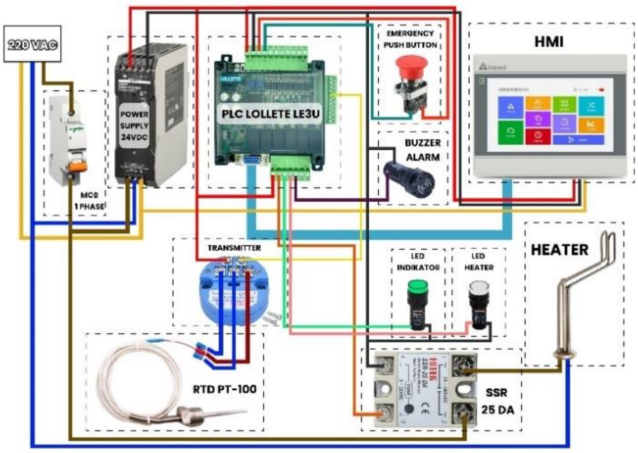

Figure 4. RTU 1 hardware design.

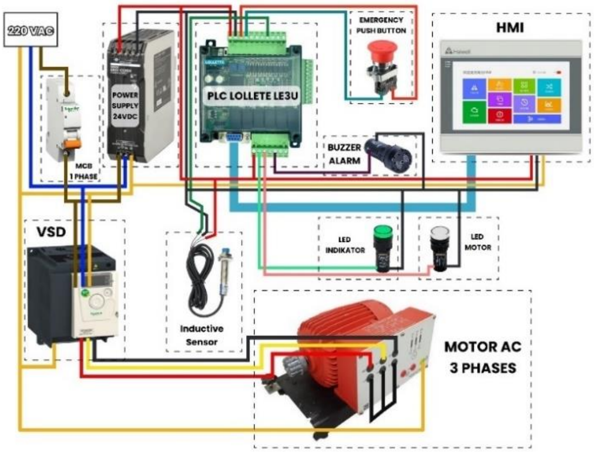

Figure 5. RTU 2 hardware design.

SCADA system wiring circuit scheme with IIoT technology is shown in Figure 4 and Figure 5 for both RTUs.

The design started from inputting 220 VAC power supply voltage into 24 VDC, which was the input source for PLC, HMI, and other control components that require 24 VDC voltage. The HMI communicated with the PLC and vice versa using an RS-232 data cable. Each RTU was connected to its respective sensors and actuators. RTU 1(Figure 1) and RTU 2 (Figure 2) were also equipped with an emergency button to turn off the system when needed, equipped with a buzzer when the alarm was on, accompanied by an indicator LED showing when the system was active and an indicator LED showing when the actuator turned on.

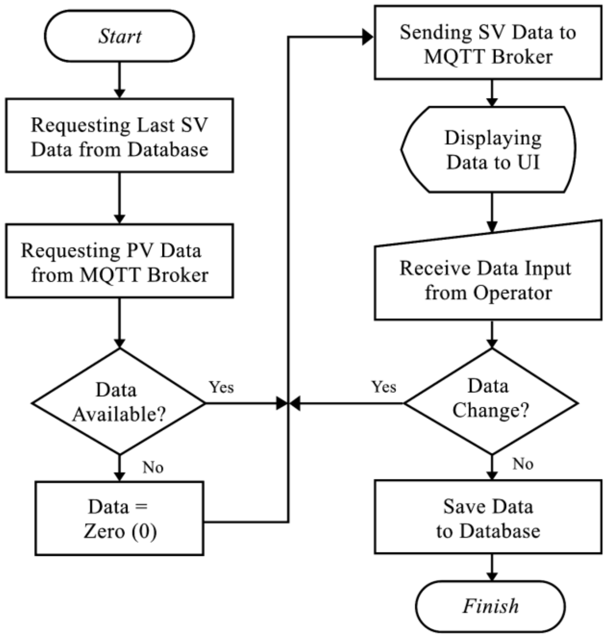

Figure 6. Flow diagram of MTU's system algorithm.

C. SYSTEM ALGORITHM DESIGN

The algorithm design for the system in this study was created separately for the MTU, RTU 1, and RTU 2. In MTU, the application desktop was developed using Visual Studio 2022 and RevaldiOiek Scada Solution (ROSS), consisting of six pages, each with its function, such as for database or control page. A software application, ROSS, was installed on the server computer, and its tasks basically included requesting and sending data to each RTU through a unique topic on the MQTT broker, storing and retrieving data from the database (as shown in Figure 6), and displaying the data on a user interface that can be understood by the operator.

Figure 7(a) and Figure 7(b) shows that, in general, RTU 1 and RTU 2 have similar algorithms because both use almost the same control system. The difference lies in the field instruments used between the two RTUs. The software design on RTU 1 and RTU 2 sent and received data from the MTU through unique topics customized on the MQTT broker. Each program on the RTU was created using GX Works2 for the Lollette LE3U PLC monitoring and control system, as well as Haiwell Cloud SCADA for the RTU interface and internet connection with the MQTT protocol. Figure 7 shows the software design scheme for RTU 1 and RTU 2.

To obtain *temp PV* data on RTU 1, a transmitter was required to convert the sensor output current into a current with a rating that could be understood by the PLC 12 bit. The transmitter used in this study converted the sensor output current into a current with a range of 4-20 mA representing 0°C to 100°C, which allowed the programming algorithm to convert the value to the actual temperature value using (1).

$$Temp\ PV = \frac{Ps - {}^{4mA}/_{20mA}(2^{12})}{2^{12} - {}^{4mA}/_{20mA}(2^{12})} \times 100°C \quad (1)$$

The heating element on RTU 1 used a 220 VAC voltage source, so a driver was needed so that a PLC with a 24 VDC rating could activate it. For the heating element, the most

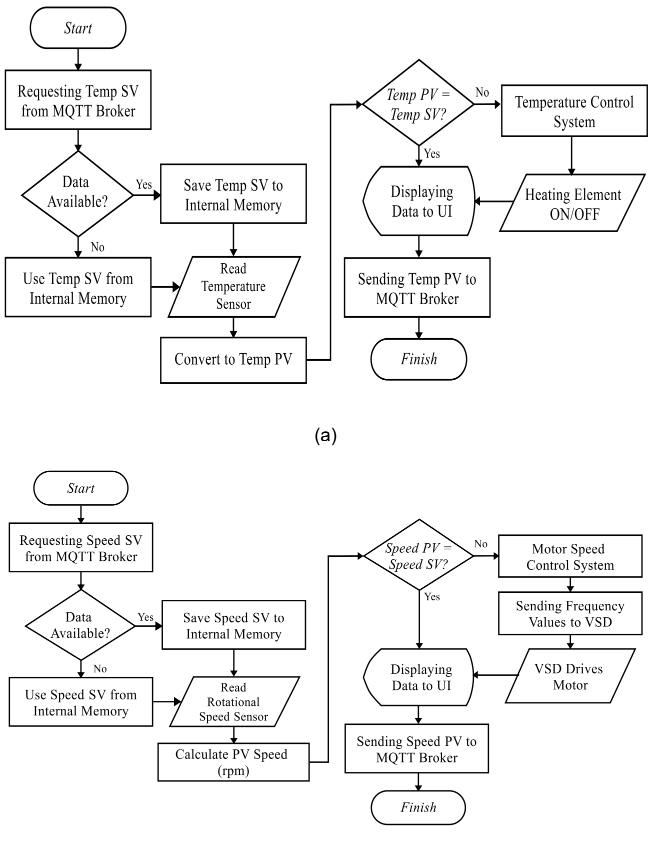

Figure 7. Flow diagram of the system algorithm, (a) RTU 1 and (b) RTU 2.

suitable driver was SSR. SSR has the same function as an electromechanical relay but has no moving parts because it works using the same principle as a transistor. The use of SSR is suitable for heating system control because in the heating system, there is often a very fast switching process between on and off to maintain the temperature.

For RTU 2, the encoder works by reading how many times the mark is read by the proximity sensor [15]. When the motor complete one full rotation, the sensor will read as many different marks as used. Based on this concept, and with a correct algorithm, the proximity sensor that can read the difference in marks can be utilized as a motor speed sensor by comparing the sensor reading frequency, divided by the number of marks on the encoder, over a unit of time, then converting the results in to rpm for *speed PV,* as shown in (2).

$$Speed\ PV = \frac{Read\ marks / Number\ of\ markers}{\Delta t} \tag{2}$$

The design in this study used six marks with a sampling time of one second $\left(1/_{60}\ \text{minute}\right)$. Thus, by applying (2), the rotational speed can be calculated using (3).

$$Speed\ PV = \frac{Read\ marks / 6}{1/60} \tag{3}$$

Controlling the angular synchronous speed of the motor in this study did not involve slip since the motor did not have a large load. The speed could be adjusted by changing the frequency applied to the motor, as shown in (4) [16], [17]. The number of poles of the three-phase AC motor used in this study was four.

$$\text{Speed SV} = \frac{120 \times f}{4} \tag{4}$$

The calculation result using (4) indicated that the rotational speed of the induction motor on system created was thirty times the frequency applied to the induction motor. This frequency was regulated by the variable speed drive (VSD) to regulate the motor speed [18]. VSD works with single- or three-phase AC inputs converted into DC voltages with rectifier circuits. Then, this DC voltage with a circuit consisting of six transistors was converted into a voltage signal resembling a three-phase AC voltage. Timing with precise computation by the control unit to the six transistors forwarded the DC voltage to form three phases of current signals with a certain frequency as desired: the narrower the timing/period value, the greater the output frequency value.

The frequency data signal to the VSD was controlled by the PLC which was communicated via the RS-485 port. The communication setting that needed to be configured were baud rate, data length, stop bit, and parity bit. In addition to the PLC, it was also necessary to configure the communication protocol as a master and specify the slave id value from the VSD. Meanwhile, on the VSD, the slave id was adjusted to the PLC.

D. MQTT COMMUNICATION PROTOCOL

Table II shows the communicated variables and topics between the ROSS application and the RTUs. MQTT is a protocol designed for IIoT applications [19], allowing for implementation in various scenarios both in device configuration and services provided. It can be implemented in limited conditions in terms of devices and network constraints. For this study, the communication between the ROSS application and both RTUs used the MQTT protocol. The ROSS application and RTUs were connected because they employed the same topics per variable at MQTT broker. The topics in Table II were then entered into the ROSS application program code on the MTU computer and the HMI on each RTU. By using identical topic addresses, both layers could be connected via an MQTT broker.

E. MYSQL DATABASE

In this study, the MySQL database was developed using the phpMyAdmin web-based application. As shown in Figure 8, four tables are used with different functions for each table. MySQL works by allowing users to create databases and tables, store data in them, and retrieve information from the database with commands [20].

Meanwhile, to ensure network security aspects and data from cyber threats and attacks, in accordance with the principles of confidentiality, integrity, and availability (CIA Triad), MQTT supports username/password-based authentication. In addition, communication can be secured with TLS/SSL encryption. Access to MTU and MySQL databases can be protected with firewalls and two-factor authentication. For real-world implementation, industrial security protocols (e.g., IEC 62443) can be integrated.

The server is a computer that stores the MySQL software and database, which in this study was the MTU hardware where the SCADA application was located. The tables named ross_rtu1heater and ross_rtu2motor function to store any data that changed. In other words, the table saved the latest value based on the variable RTU address. Furthermore, the tables named ross_rtu1con and ross_rtu2con functioned to store data with a gap of every ten seconds. This table, updated every ten

TABLE II
TOPIC OF EACH VARIABLE

Variable	Declaration	Topic
Output temperature	TempPV	data/ROSS/RTU1TempPV/123
Reference temperature	TempSV	data/ROSS/RTU1TempSV/123
RTU 1 actuator status	StaAkt1	data/ROSS/RTU1AktuStat/123
RTU 1 access permission	RTUAks1	data/ROSS/RTU1Aks/123
RTU 1 alarm limit	Range1	data/ROSS/RTU1RangeTempSV/123
RTU 1 alarm setting	AlarmEnDis1	data/ROSS/RTU1AlarmSet/123
Output speed	Speed PV	data/ROSS/RTU2Speed PV/123
Reference speed	Speed SV	data/ROSS/RTU2Speed SV/123
RTU 2 actuator status	StaAkt2	data/ROSS/RTU2AktuStat/123
RTU 2 access permission	RTUAks2	data/ROSS/RTU2Aks/123
RTU 2 alarm limit	Range2	data/ROSS/RTU2RangeTempSV/123
RTU 2 alarm setting	AlarmEnDis2	data/ROSS/RTU2AlarmSet/123

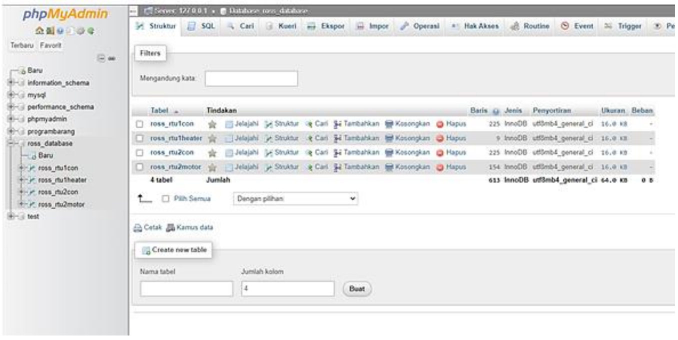

Figure 8. Four database table for the ROSS application.

seconds, was sent by the ROSS application depending on the RTU address of the data, which was intended for analytics purposes.

III. RESULTS AND DISCUSSION

A. DATA ACCURACY TESTING OF MEASURING INSTRUMENTS AND SYSTEMS

1) RTU 1 – WATER HEATER SYSTEMS

This measurement was carried out at room temperature, with a range of every 20±5°C from 0 to 100°C. Based on the test results, the results were stable although not perfectly linear. The largest error value was 3.6°C but only appears when the temperature was almost 10°C. The average error from the 24 tests was 0.4 °C. The average error percentage for each data test performed was 1.4%, so the percentage of sensor reading accuracy was 98.6%. This percentage accuracy value was high enough so that this system could be relied on. The distribution of temperature measurement data from RTU 1 can be seen in

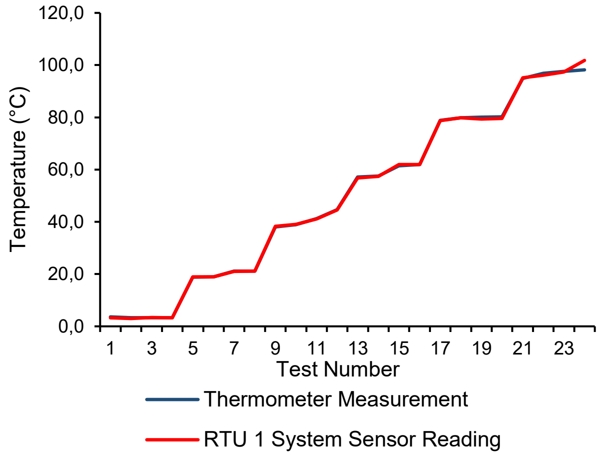

Figure 9. RTU 1 sensor testing comparison.

Figure 9. This figure shows that the largest error occurred in the last test when the temperature approached 100°C. This occurred because 100°C is the maximum limit of the sensor reading on the RTU 1 system, causing its accuracy to decrease as the temperature approached that threshold. However, the sensor in this system was stable because when the water heater (plant) experienced a temperature change, as it was still able to follow the temperature change.

2) RTU 2 – MOTOR SPEED CONTROL

Measurements ranged from 0 rpm to 1500 rpm, with a range of 60 rpm or 2 Hz (as shown in (7)), resulting a total of 26 data obtained during the rotational speed sensor test on RTU 2. The test results showed that the measurement results were stable and linear, indicating that the error value of the sensor reading between the RTU 2 system and the tachometer of the entire test was 0 rpm, so the error percentage was 0%. Thus, the average error percentage of the 26 tests carried out was 0%, with an accuracy percentage of 100%. The distribution of rotational speed measurement data from RTU 2 can be seen in Figure 10.

This system had an error of 0 rpm and an error percentage of 0% in each test representing a value between 0 and 50 Hz, so the sensor readings of this system were very stable, linear, and reliable for use. Based on the calculation using (5), the rotational speed sensor on RTU 2 had an accuracy of 10 rpm. This result indicated that the sensor was not accurate if the actual speed did not include a multiplication of 10. Nevertheless, according to (7), the motor speed could only be set at a multiplication of 30 rpm, and since 30 corresponded to a factorial value of 10 (10 x 3), the accuracy of the sensor reading on RTU 2 was sufficient and did not need to be improved. In other words, the motor on the RTU 2 system could not be set at a rotational speed outside the multiplication of 30 rpm precisely and nor it could be read outside the multiplication of 10 rpm. However, for further research, if the three-phase motor can be set at a speed with an accuracy below 10 rpm, then the sensor on RTU 2 needs to be improved by increasing the number of plates on the encoder and change the read calculation.

B. RTU AND MTU DATA COMMUNICATION TESTING

In testing of data communication between RTU and MTU, both communication with the MQTT protocol between RTU and MTU, and between the ROSS application and the database

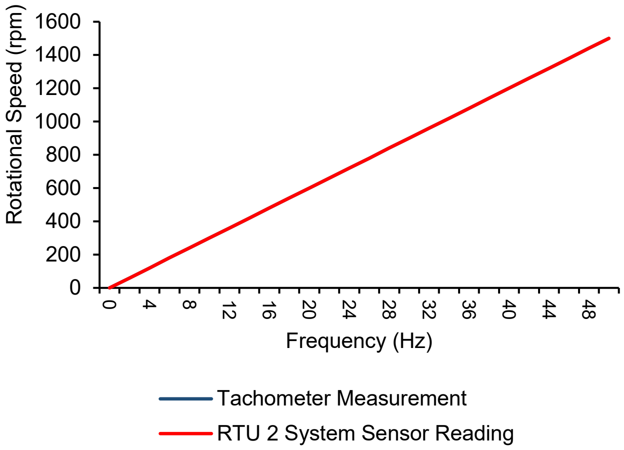

Figure 10. RTU 2 sensor testing comparison.

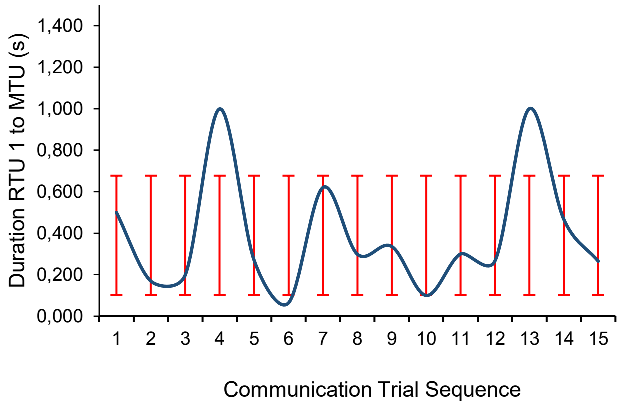

Figure 11. RTU 1 to MTU communication duration.

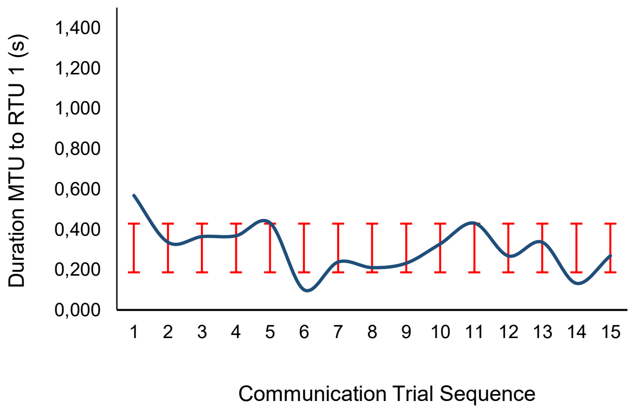

Figure 12. MTU to RTU 1 communication duration.

on the MTU were carried out with different network types at different times. All data communicated were not lost, so that the data sent and received were 100% successful.

The scalability of the MQTT broker-based system architecture enables high scalability, as the MQTT protocol supports the addition of new RTUs/PLCs simply by adding relevant communication topics. The system can be expanded from two RTUs to dozens of RTUs without significant changes to the MTU or ROSS application design. SCADA systems, especially with distributed control system (DCS) platforms for very large and complex applications, as well as high-specification PLCs such as the ControlLogix series from Rockwell Automation, offer the modularity and processing power required.

C. REAL-TIME DURATION TESTING OF DATA COMMUNICATION

Data communication duration testing was carried out to see whether the system that had been built performed well for real-time operation.

1) RTU 1 – WATER HEATER SYSTEMS

Temp PV was used to represent the data sent by RTU 1 to the MTU in the SCADA application. Throughout fifteen tests, the average communication duration from RTU 1 to MTU was 0.390 s, with a standard deviation of 0.287 s. Variations in Wi-Fi network speeds can cause significant differences between the fastest and slowest delivery times. Figure 11 illustrates the data distribution and standard deviations for these tests.

The variable representing the data sent by the MTU in the SCADA application to RTU 1 was the *temp SV*. Testing of fifteen data obtained the duration of communication from MTU to RTU 1, which had an average duration of 0.307 s. The standard deviation of the test was quite low at 0.121 s. The distribution of data and standard deviation on the average test can be seen in Figure 12. Based on these data, the duration of data communication from MTU to RTU 1 or RTU 1 to MTU was very fast and reliable because it could work in real time.

2) RTU 2 – MOTOR SPEED CONTROL

The *speed PV* variable is a variable used to represent data delivery by RTU 2 to MTU in SCADA applications. In fifteen communication duration tests from RTU 2 to MTU, the average duration was 0.227 s, and the standard deviation of the test was only 0.115 s. The distribution of data and their standard deviation on the average duration can be seen in Figure 13.

The distribution of data and standard deviation on the average of this test can be seen in Figure 14. One of the variables sent from the MTU in the SCADA application to RTU 2 was the *speed SV*. Based on fifteen data of the communication duration test results from the MTU to RTU 2, an average duration was 0.373, with a relatively low standard deviation of 0.141 s. Based on these data, the duration of data communication from MTU to RTU 2 and RTU 2 to MTU was very fast and reliable because they could work in real-time.

Overall, modern SCADA systems utilizing wired networks and efficient communication protocols required communication times of less than 1 s. Meanwhile, communication times for systems that used wireless or satellite networks could reach several seconds, sometimes exceeding 5 s. The average duration required to communicate was 0.324.

D. CONTROL SYSTEM TESTING

Each RTU owned a control system, which was controlled by a PLC based on input from references and output from sensor readings. The control system in RTU 1's water heating system operated using a proportional integral (PI) control system. It was adjusted so that when the reference temperature was significantly different, the actuator worked optimally. On the other hand, if the reference was closer, the actuator worked by utilizing SSR switching. This approach ensures efficient temperature regulation and energy usage, allowing for quick adjustments in response to varying conditions. Additionally, the system's design promotes reliability and minimizes wear on the actuator components, extending their operational lifespan. Figure 8 shows that the water, which was initially at a

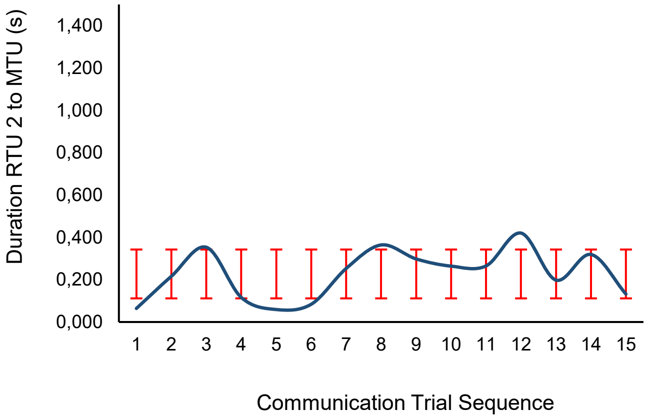

Figure 13. RTU 1 to MTU communication duration.

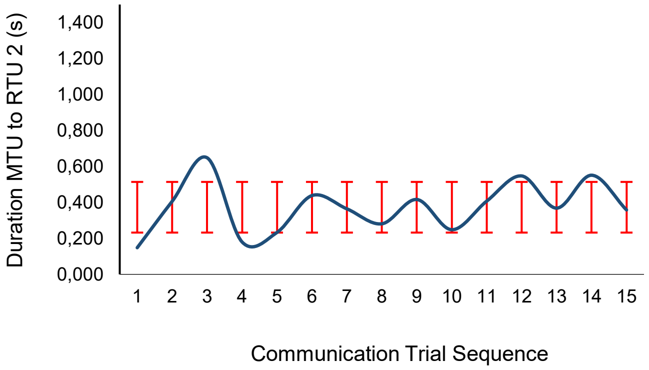

Figure 14. MTU to RTU 2 communication duration.

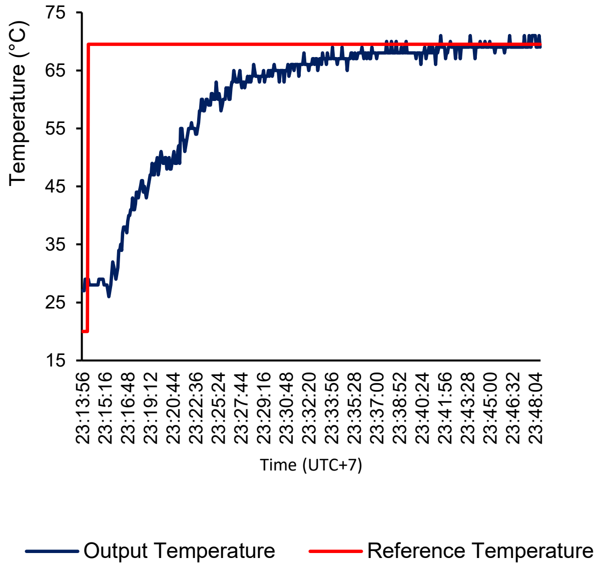

Figure 15. RTU 1 control test.

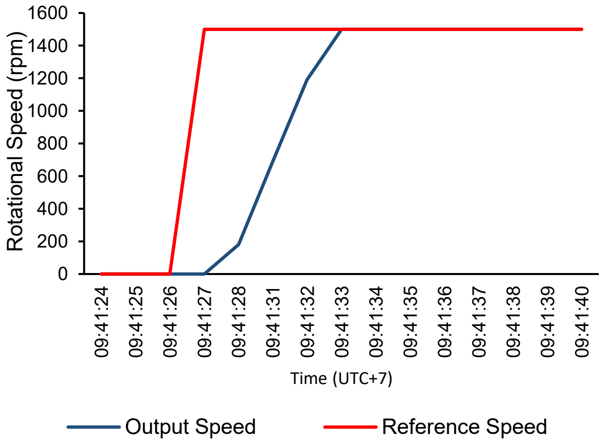

Figure 16. RTU 2 control test.

temperature of approximately 27°C at 23:13:56 UTC+7, reached approximately 70°C at 23:40:24 UTC+7. It can be observed that when the temperature was far from the reference, the temperature increased rapidly. However, when the error between the reference and the output was smaller, the temperature did not significantly increase. This behavior illustrates the efficiency of the system in achieving its target temperature while maintaining stability. Such characteristics are crucial in applications where precise temperature control is essential for optimal performance and safety. From 23:40:24 UTC+7 to 23:48:24 UTC+7, the system maintained the output temperature relative to the reference temperature. This was achieved by utilizing SSR switching, which alternately turned the heater on and off.

The control system in RTU 2, namely motor speed control, used proportional integral differential (PID). It was regulated by VSD, while its value was controlled by PLC. Based on Figure 15 and Figure 16, VSD drove the motor at the start with a soft starter concept, to make the motor more durable and the initial power draw was not excessive (overshoot). When the motor rotated, the motor acceleration was increased by the VSD so that the motor speed increased sharply. As the motor approached the reference, its acceleration decreased again to avoid speed overshoot, so that the motor speed did not exceed its reference.

E. SYSTEM DEVELOPMENT COST ANALYSIS

The system used in this study used the most cost-efficient components that could be obtained. By prioritizing affordability without sacrificing quality, the study aimed to provide a practical solution for similar future projects. The SCADA system built was not only a network for SCADA communication like other conventional SCADA systems, but also a control system from each RTU using PLC class components and other industry standard components. This innovative approach enables real-time monitoring and control, enhancing operational efficiency while minimizing costs. Additionally, it allows for greater flexibility in system expansion and integration with emerging technologies. Compared to conventional SCADA, this system is 80 % cheaper and offers flexibility: it is easy to replicate and reconfigure.

The system developed is nearly ready for use and industrial implementation, with no major changes required. The final testing phase is underway, and early feedback suggests that user experience will be significantly enhanced. Stakeholders are optimistic about the potential impact on productivity and efficiency across various sectors. Based on cost calculations, this system is significantly cheaper than SCADA systems with similar features and customization options, which can cost up to US$5,000.00 [21], or with a cost efficiency of around 80.6%.

For long-term reliability, this study focuses on the reliability of SCADA systems that have been designed, developed, and tested repeatedly under various communication testing conditions, with no packet loss. However, for perfect long-term reliability, further research in the form of endurance tests (stress tests) lasting several weeks/months is needed to observe sensor degradation, MQTT communication stability, and the durability of PLCs and HMIs.

However, conventional SCADA systems on the market have certainly undergone extensive research and systems testing, making them more stable. In the real-world implementation of this research, the proposed system was implemented in the pilot project laboratory of the Electrical Engineering Department a t Tanjungpura University with a real load consisting of a water heater and a 3-phase AC motor. The system was designed to replicate small industrial conditions. At present, direct application in industry is currently in the offering stage.

IV. CONCLUSION

Based on the results of the tests and analysis, in conclusion, this study is successful and meets the desires, with accuracy between measuring instruments on RTU 1 reaching 98.6% and on RTU 2 up to 100%. No packet loss occurred during the communication between SCADA layers, meaning that the MQTT protocol with IIoT technology in this study did not experience any packet loss. Real-time testing by measuring the duration of communication between RTU and MTU depended on the type of data and network, but the duration of delivery between RTU 1 and MTU and RTU 2 and MTU was very small, around 0.324 s. The control systems of both RTUs can be trusted, because their output can reach the user's reference and are stable, and the cost efficiency of building a SCADA system compared to a conventional SCADA system is very large, around 80.6%. The built SCADA system can also be replicated and reconfigured.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest in the research and preparation of this paper.

AUTHORS' CONTRIBUTIONS

Conceptualization, Hendro Priyatman, Supriono, and Seno D. Panjaitan; methodology, Seno D. Panjaitan; software, Hendro Priyatman; validation, Seno D. Panjaitan; formal analysis, Supriono, Muh. Revaldi Frizky; investigation, Muh. Revaldi Frizky; writing-preparation of initial draft, Muh. Revaldi Frizky; writing-review and editing, Supriono; visualization, Muh. Revaldi Frizky; supervision, Seno D. Panjaitan, Hendro Priyatman, Supriono and Muh.Revaldi Frizky.

ACKNOWLEDGMENT

The authors would like to thank the Control Engineering Laboratory of the Faculty of Engineering, Tanjungpura University Pontianak for providing the necessary support equipment for this research.

REFERENCES

[1] A.S. Marsudi and Y. Widjaja, "Industri 4.0 dan dampaknya terhadap financial technology serta kesiapan tenaga kerja di Indonesia," *Ikra-Ith Ekonomika*, vol. 2, no. 2, pp. 1–10, Jul. 2019.

[2] A.W. Colombo *et al.*, "A 70-year industrial electronics society evolution through industrial revolutions: The rise and flourishing of information and communication technologies," *IEEE Ind. Electron. Mag.*, vol. 15, no. 1, pp. 115–126, Mar. 2021, doi: 10.1109/MIE.2020.3028058.

[3] M. Hermann, T. Pentek, and B. Otto, "Design principles for industrie 4.0 scenarios," in *2016 49th Hawaii Int. Conf. Syst. Sci. (HICSS)*, 2016, pp. 3928–3937, doi: 10.1109/HICSS.2016.488.

[4] T. Thepmanee, S. Pongswatd, F. Asadi, and P. Ukakimaparn, "Implementation of control and SCADA system: Case study of Allen Bradley PLC by using WirelessHART to temperature control and device diagnostic," *Energy Rep.,* vol. 8, pp. 934-941, Apr. 2022, doi: 10.1016/j.egyr.2021.11.163.

[5] H. Boyes, B. Hallaq, J. Cunningham, and T. Watson, "The industrial Internet of things (IIoT): An analysis framework," *Comput. Ind.,* vol. 101, pp. 1–12, Oct. 2018, doi: 10.1016/j.compind.2018.04.015.

[6] M. Alabadi, A. Habbal, and X. Wei, "Industrial Internet of things: Requirements, architecture, challenges, and future research directions," *IEEE Access*, vol. 10, pp. 66374–66400, Jun. 2022, doi: 10.1109/ACCESS.2022.3185049.

[7] E. Sisinni *et al.*, "Industrial Internet of things: Challenges, opportunities, and directions," *IEEE Trans. Ind. Inform.*, vol. 14, no. 11, pp. 4724–4734, Nov. 2018, doi: 10.1109/TII.2018.2852491.

[8] B.Chen *et al.*, "Smart factory of industry 4.0: Key technologies, application case, and challenges," *IEEE Access*, vol. 6, pp. 6505–6519, Dec. 2017, doi: 10.1109/access.2017.2783682.

[9] R. Dugyala, N.H. Reddy, and S. Kumar, "Implementation of SCADA through cloud based IoT devices – Initial design steps," in *2019 5th Int. Conf. Image Inf. Process. (ICIIP)*, 2019, pp. 367–372, doi: 10.1109/ICIIP47207.2019.8985966.

[10] Hasan, I. Suharto, and W. Heryawan, "Rancang bangun jaringan komunikasi PLC menggunakan sistem WLAN untuk memonitor proses kontrol berbasis human machine interface SCADA INFO U," *Vokasi*, vol. 14, no. 2, pp. 87–94, Dec. 2019.

[11] L.O. Aghenta and M.T. Iqbal, "Low-cost, open source IoT-based SCADA system design using Thinger.IO and ESP32 thing," *Electronics*, vol. 8, no. 8, pp. 1–24, Aug. 2019, doi: 10.3390/electronics8080822.

[12] I. Harjanto, "IoT gateway menggunakan protokol MQTT pada perangkat kendali berbasis modbus-RTU," *JITEK (J. Ilm. Teknosains)*, vol. 6, no. 1, pp. 12–19, May 2020, doi: 10.26877/jitek.v6i1.5957.

[13] A. Supriyono, M.J. Afroni, and O. Melfazen, "Penerapan SCADA berbasis IoT untuk simulator kontrol panel pada contoh kasus pembangkit listrik tenaga mikrohidro (PLTM)," *JE-Unisla*, vol. 7, no. 2, pp. 119–126, Sep. 2022, doi: 10.30736/je-unisla.v7i2.889.

[14] M. Endi, Y.Z. Elhalwagy, and A. Hashad, "Three-layer PLC/SCADA system architecture in process automation and data monitoring," in *2010 2nd Int. Conf. Comput. Autom. Eng. (ICCAE)*, 2010, pp. 774–779, doi: 10.1109/ICCAE.2010.5451799.

[15] K.-M. Lee, T. Gu, and Y.-B. Bang, "Analysis of accuracy and measuring range of dual absolute encoder system," *IEEE Sens. J.*, vol. 20, no. 6, pp. 2997–3004, Mar. 2020, doi: 10.1109/JSEN.2019.2955381.

[16] A.W. Aditya, R.M. Utomo, N.R. Alham, and Hilmansyah, "The industrial IoT control design of three phase induction motor using conventional VF method," *Elkha, J. Tek. Elekt*., vol. 15, no. 1, pp. 67–72, Apr. 2023, doi: 10.26418/elkha.v15i1.63817.

[17] Z. Anthony, "Pengaruh perubahan frekuensi dalam sistem pengendalian kecepatan motor induksi 3-fasa terhadap efisiensi dan arus kumparan motor," *J. Tek. Elekt*., vol. 1, no. 1, pp. 25–29, Jan. 2018.

[18] A. Spahiu, D. Bizhga, and L. Dhamo, "Reduction of electricity consumption and water cost in pump application," in *2020 IEEE 3rd Int. Conf. Workshop Óbuda Electr. Power Eng. (CANDO-EPE)*, 2020, pp. 000137–000142, doi: 10.1109/CANDO-EPE51100.2020.9337777.

[19] R.A. Atmoko and D. Yang, "Online monitoring & controlling industrial arm robot using MQTT protocol," in *2018 IEEE Int. Conf. Robot. Biomim. Intell. Comput. Syst. (Robionetics)*, 2018, pp. 12–16, doi: 10.1109/ROBIONETICS.2018.8674672.

[20] G. Ongo and G.P. Kusuma, "Hybrid database system of MySQL and MongoDB in web application development," in *2018 Int. Conf. Inf. Manag. Technol. (ICIMTech)*, 2018, pp. 256–260, doi: 10.1109/ICIMTech.2018.8528120.

[21] DPS Telecom "Take control of your remote monitoring." Access date: 16-Sep-2014. [Online]. Available: https://www.dpstele.com/

This page is intentionally left blank