Design and Build an Internet of Things System to Monitor Electrical Energy in Electronic Devices

[1]Frans Davidson, [2]Maria Agustin
[1,2]Jurusan Tenik Informatika dan Komputer, Politeknik Negeri Jakarta, Depok, Jawa Barat, Indonesia
[1] frans.davidsonmangapulnababan.tik20@mhsw.pnj.ac.id
[2] maria.agustin@tik.pnj.ac.id

Abstract - Electrical power measurements are usually carried out using simple measuring instruments and manual recording so that the data can be accessed at any time and takes a long time to obtain. This research aims to implement an electrical energy monitoring system using Internet of Things (IOT) technology to obtain real time electrical energy measurement information that can be accessed via the internet network. The author uses the PZEM 004t sensor to get more accurate measurement results. To connect sensors to the IoT system, the author uses serial communication and ESP 32 to process data from sensors, so that the data can be sent using the MQTT communication protocol and displayed on the dashboard in real time. The test results show that the electrical energy monitoring system for home electronic devices can be implemented and runs well with end to end measurement results. From the results of measuring the current value (average 1 phase) with a voltage of 220 v with an error rate of 0.5-1% with an accuracy of 99%, service quality is measured with index 4 based on the TIPHON standard on delay, throughput and packet loss parameters, as well as index 3 is based on TIPHON standardization of jitter parameters

Keywords: Monitoring, Electrical Energy, PZEM 004T, MQTT, QoS (Quality Of Service).

I. INTRODUCTION

The increasing demand for electrical energy by households indicates the need for efficient solutions to monitor and manage electrical power consumption. Based on Energy and Mineral Resources (ESDM) records, electricity consumption in Indonesia will reach 183.41 million Barrels of Oil Equivalent (BOE) in 2022[1]. This amount has increased by 7.92% compared to the previous year which was 169.95 million BOE. So far, electrical power usage can only be seen using a PLN measuring instrument, the use of this instrument does not provide detailed information on how much electrical power is used which is connected to a KWH meter, making it difficult for users to know how much electrical power used[2].

The Internet of Things is becoming an increasingly popular trend, enabling the integration of devices to communicate with each other and providing better control to users. This provides an opportunity to improve electrical power management through IoT solutions. IoT-based Smart Energy Meters can overcome existing problems such as workforce reduction, energy monitoring, load management, electricity theft, etc. Smart Energy Meter is the same as a regular Energy Meter. This is an advanced technology for reading and controlling energy consumption[3],[4] .

Rising energy costs have individuals and companies looking for ways to optimize electrical power usage, so having an effective monitoring system can help reduce operational costs. Previously, research had been carried out using the pzem sensor and esp32. However, the electric power monitoring system still does not have a dashboard for monitoring and there is no Quality of Service measurement[5], [6].

Previously, research had been carried out on an electricity monitoring system using the Pzem 004t sensor and Arduino Uno. In this research, there is no dashboard for monitoring other than testing service quality such as delay, jitter, packet loss and throughput so it cannot be known whether the system can produce data. in real time[7].

The author proposes to conduct research related to Internet of Things-based electrical energy monitoring using the Pzem 004t sensor, where this sensor can accommodate and measure all electrical quantities. Apart from that, researchers also used ESP 32 which has a WiFi module so that the device can connect directly to the internet network. wirelessly and sending data via the MQTT protocol to be able to display data on the Node-RED dashboard and carry out analysis regarding the Quality of Service provided by the system[8],[9].

II. RESEARCH METHOD

A. Tools and Materials

In this research, several stages of how the tools used work will be explained.

1) Tools

a Computer/laptop

In this research, the computer or PC used was the Lenovo Ideapad 5i. Which uses an Intel I5 processor, has 8GB DDR4 memory. With 512 GB of Samsung M.2 NVme SSD storage and the operating system used is Windows 11. PCs with these specifications are used for making reports, system integration and measuring QoS (Quality Of Service).

Fig 1 Laptop

b Node Red

Node-RED is often used in situations that require rapid prototyping, visual development, and broad integration with various systems. This platform is famous for having an active and solid community, as well as providing various nodes that can be utilized for various needs, with contributions from users. In this research, the author uses Node-RED to integrate data from the PZEM 004T into a microcontroller and create a local dashboard[10].

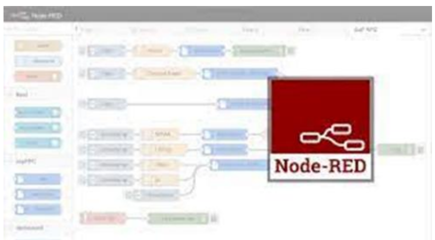

Fig 2 Node Red

c Wireshrak

Wireshark is a very useful tool in checking and improving network performance efficiency, as well as identifying possible problems or security threats. In this research the author used Wireshark software to measure service quality on the MQTT protocol[11].

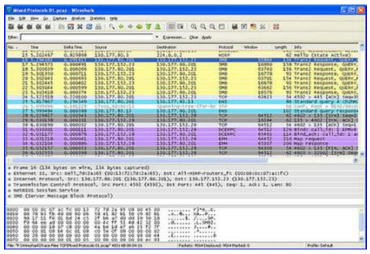

Fig 3 Wireshark

2) Materials

a PZEM004T

In this research, the author uses the PZEM 004T sensor as a tool to measure the implemented electrical parameters. PZEM 004T is a digital sensor used to measure electricity consumption and load management. This module can measure various electrical parameters in AC systems, including current, voltage, power, energy, power factor (pf), and frequency. This sensor has a maximum current measurement capacity of up to 100 A. At a low cost, the PZEM 004T offers the features needed to combine electricity consumption[12]. Some of the features of the PZEM 004T are listed below.

a) can measure from 80 – 260V with a resolution of 0.1 V and an accuracy of 0.5%.
b) The current sensor can measure from 0-10A and 0-100A with a resolution of 0.001A and an accuracy of 0.5%.
c) Active Power Sensor can be measured from 0-2.3 KW and 0-23 KW with a resolution of 0.1 W and an accuracy of 0.5%.
d) The Power Factor Sensor can measure 0.00 – 1.00 pf with a resolution of 0.01pf and an accuracy of 1%.
e) The frequency sensor can measure 45Hz - 65 HZ with a resolution of 0.1 HZ and an accuracy of 0.5%.
f) Active Energy Sensor can measure 0-9999.99 kWh with a resolution of 1Wh and an accuracy of 0.5%[13].

Fig 4 PZEM 004T

Fig 5 CT (Current Tranformer) PZEM 004T

b ESP 32

ESP 32 has very useful features such as TCP/IP, HTTP and FTP, this module is also equipped with features that support analog signal processing, support for sensors and also has enough digital I/O ports. Here are the ESP 32 specifications, namely[14]:

a) Processor: Xtensa dual-core (or single-core) 32-bit LX6 microprocessor, operating at 160 or 240 MHz.
b) Memory: 520 KB SRAM
c) Wireless connectivity: Wi-Fi 802.11 b/g/n, Bluetooth v4.2 BR/EDR and BLE (radio sharing with Wi-Fi)
d) Peripheral I/O: 12-bit SAR ADC (up to 18 channels), 2x 8-bit DACs, 10x touch sensors (capacitive sensing GPIO), 4x SPI, 2x I2S interfaces, 2x I2C interfaces, 3x UART, SD/SDIO / CE-ATA/MMC/eMMC host controller, SDIO/SPI slave controller, Ethernet MAC interface, CAN bus 2.0, infrared remote controller (TX/RX, up to 8 channels), PWM motor, PWM LED (up to 16 channels), hall effect sensor, very low power analog pre-amplifier[15].

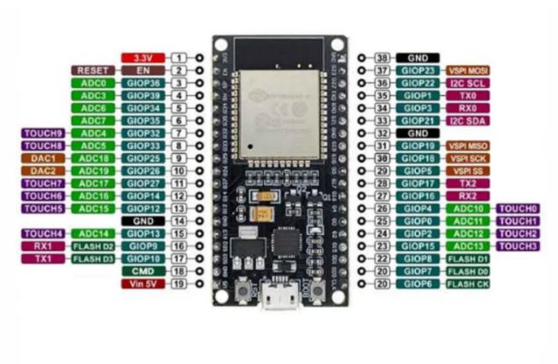

Fig 6 ESP 32

B. Research Flow

The research was carried out in the form of a prototype with a quantitative experimental research method consisting of several stages, namely stages starting from determining literature studies, system design, system testing, and analysis of the test results that have been carried out.

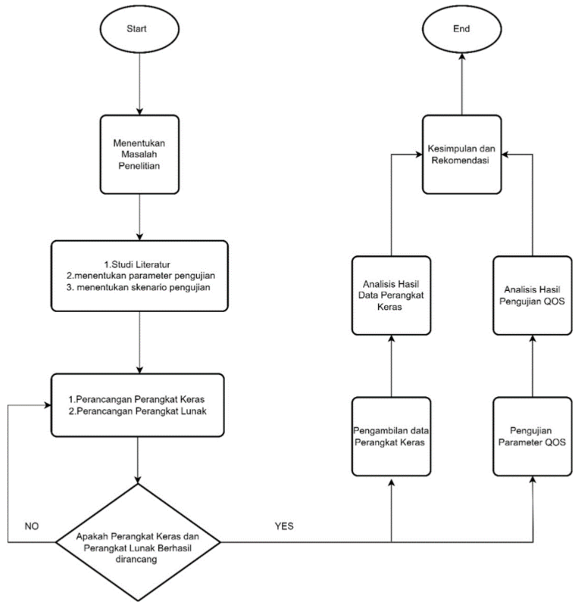

Fig 7 Research Flow

Setting these steps is necessary to achieve optimal levels of performance in conducting research. The initial step begins with searching for sources from previous research, using references from scientific journals, proceedings and books. The function of the literature study is to provide references that enable the identification of new concepts that can be developed within the framework of this research. Furthermore, based on the literature study the author determines the research problem that will be developed. After the research problem has been determined, the author determines what parameters will be tested regarding the system being used. will be developed.

In this research, several parameter tests will be carried out starting from measuring data collection from the PZEM 004T sensor for each parameter and testing the quality of the QOS system service on the MQTT protocol, to find out how good the protocol used in system design is. After the test scenario was obtained, the author then carried out hardware and software design using the tools and materials previously described.

C. System Design

System design is carried out in several stages, starting from determining tools and materials, creating a system design block diagram, creating the input flow, process and system output, designing the sensor interface with outlets and electrical loads and creating a communication schematic between the microcontroller and the PZEM 004T sensor.

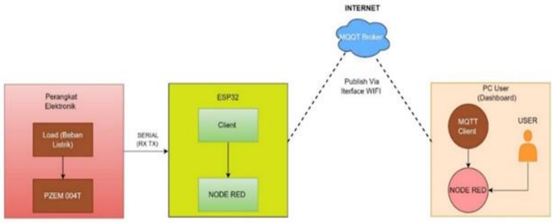

Fig 8 System Design

In figure 8, the system explains that the design of the tool begins with installing the PZEM 004T sensor into an electrical device (load). Then the sensor device is connected to the microcontroller using serial communication. ESP 32 uses RX TX to retrieve data from the PZEM 004T sensor after the sensor data is obtained, then ESP 32 forwards the data to the MQTT broker using the publish method via a WIFI interface that has been connected to the Internet using Node-RED. After that, PC users who are connected to the internet can access the data locally by subscribing using Node-RED and displaying it on the dashboard

Fig 9 Input Output System

From figure 9 it can be seen that the input for this research is the value of the electrical quantity read by the PZEM 004T sensor from an electronic device connected to the sensor. After the parameters can be read by the PZEM 004T sensor, the data will then be processed by the ESP 32 microcontroller. The ESP 32 acts as a bridge that sends parameter data values and transfers the data to the internet via a WiFi network. The data will then be displayed on a simple dashboard locally to display the output of this research

D. Hardware Design

The hardware design of this system consists of the entire schematic drawing of the interface design between the PZEM 004T sensor and the electrical outlet and the PZEM 004T sensor interface with the ESP32 microcontroller

a Scheme pzem 004t with load

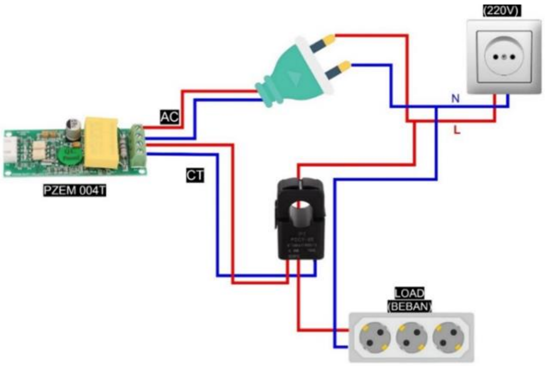

Fig 10 Scheme pzem 004t with load

In Figure 10 above is an illustration of the wiring connectivity that connects the outlet and electrical load to the PZEM 004T sensor. CT (Current Transformers) placement and Current Transformers connectivity to the PZEM 004T sensor. The port for each connection can be seen in Table 2.1 below

Table 1 Sensor connection port with load and CT

Pzem	Load	CT
Nout	L (Line)	Nout
Nin	N(Netral)	Nin
Lin		Nout
Lout		Nin

b Scheme pzem 004t with loading

To connect the PZEM 004T sensor with the ESP 32 microcontroller, wiring using serial communication is required to convert the PZEM 004T interface with the ESP 32

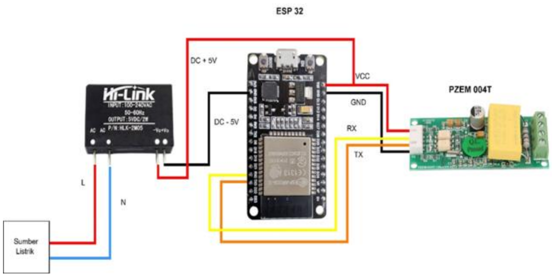

Fig 11 Scheme pzem 004t with ESP32

As in Figure 11 above, port connection details can be seen in Table 2.2 below

Table 2 Pin koneksi pzem 004t dengan esp 32

Pzem 004T	ESP 32
VCC	VCC
GND	GND

RX	TX
TX	RX

E. Software Design

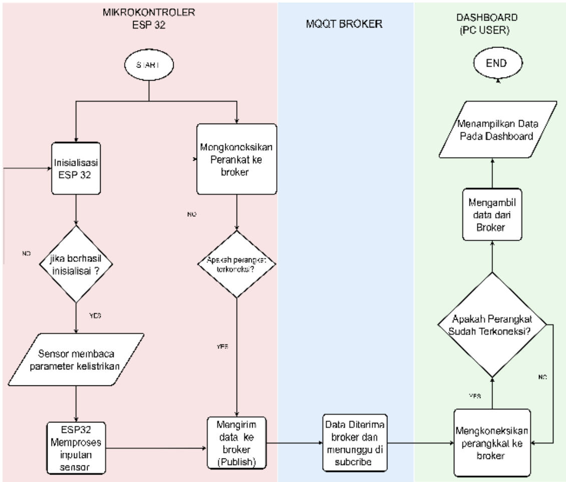

Fig 12 Flow Software Design

In Figure 12 above, the flowchart diagram of the software design created for the system is explained. There are 3 main parts in software design, namely the ESP32 microcontroller, MQQT broker, and dashboard (user's PC). The flow starts with initializing the ESP 32. If it is initialized, then the sensor will start reading the electrical parameters, after the sensor has successfully read the electrical parameters, Current, Voltage, Power, Power Factor, Frequency, then the microcontroller will process the electrical parameter data, after the data for each parameter is obtained then the data is put together so that it becomes one message package which will be sent to the server using the MQTT protocol, then the data will be subscribed using Node Red and displayed on the dashboard.

III. RESULT AND DISCUSSION

1 System Design Result

In this research there are several designs, including the results of hardware design including the assembly of the entire component. Then the software design includes the display on the dashboard. The installation result can bee seen in figure 13.

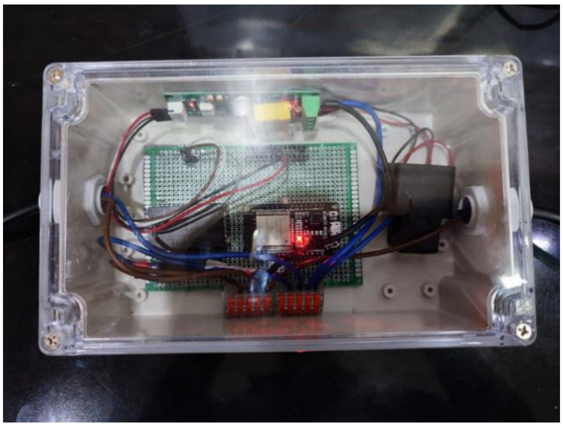

Fig 13 Hardware Installation Resul

1) Results of installation of ESP 32 interface and power source.

In Figure 14 the installation of the ESP32 interface and power source, each component functions as follows

a) The ESP32 blue box functions as a microcontroller
b) The HI-Link green box functions as an AC to DC converter
c) The orange box is VCC and GND functions as power with a DC 5V input
d) The red box is a connector that functions as an AC electricity input
e) The yellow box is the RX TX which functions as a pin for serial communication between the ESP 32 and PZEM 004T

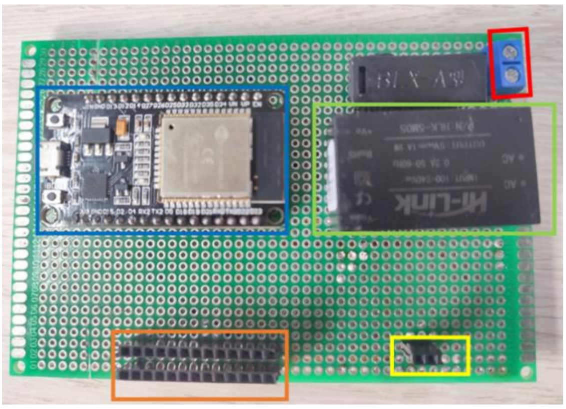

Fig 14 ESP 32 Installation Results with Power Source

2) Result of installation of PZEM 004T interface with ESP 32.

In Figure 15 is the result of installing the PZEM 004T interface with ESP32, each component functions as follows:

a) The orange box is the PZEM 004T which functions to process the measured electrical parameter data.
b) The green contact of the Current Transformer functions as a tool for measuring electrical parameters.

c) The Wago connector functions as a connector for the L&N cable to AC electricity.

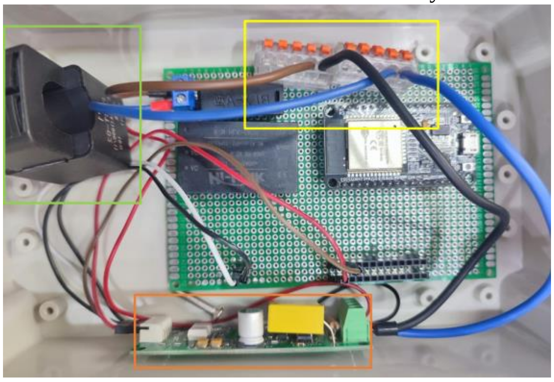

Fig 15 ESP 32 Installation Results with Power Source

3) Flow Node-Red And MQTT Server

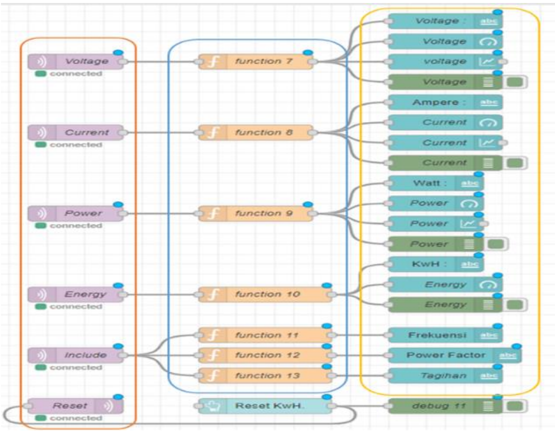

Fig 16 logic flow dashboard on Node-Red

Flow design and logic configuration on Node Red. Based on Figure 16, there are 3 segments which can be explained as follows:

Segment 1: Using the MQTT Subscribe Node which functions to subscribe to data which is measured using the ESP 32 module with serial communication connected to the PZEM 004T.

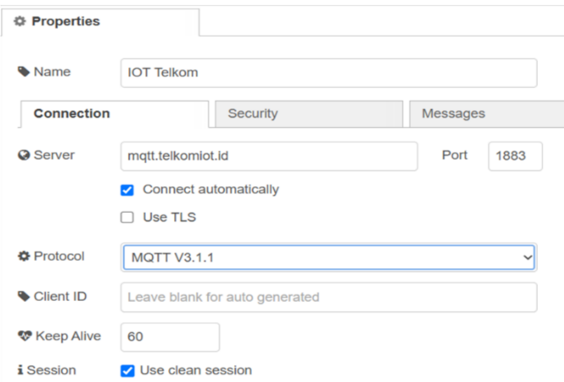

Fig 17 Server Node-Red Configuration

In the MQTT Subscribe configuration, it consists of server name, server port, protocol and server protocol. The complete configuration of segment 1 can be seen in table 3.

Table 3 Node-Red Configuration

Parameter	Protocol	Port	Server
Voltage(V)			
Current(I)			
Power(P)			
Energy(E)			Mqqt.telkomiot.id
Power Factor(PF)	MQTT V3.1.1	1883	
Tagihan (Rp)			
Reset Kwh			

Segment 2: Using the Node Function which functions to convert buffer values and call parameters that have been published on the server to be displayed on the dashboard. Function configuration can be seen in Figure17

Segment 3: uses the Gauge, Chart and Debug nodes which function to display data for each parameter with Gauge and Chart displays on the dashboard. Meanwhile, debug functions to display connected data, namely data in segment 1.

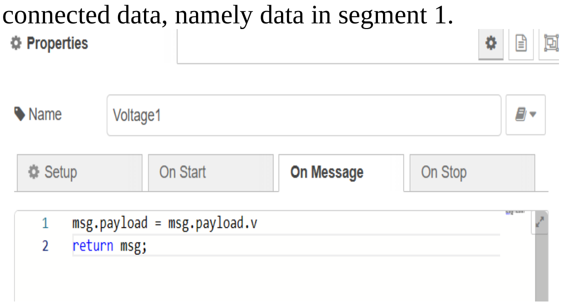

Fig 18 Function Node-Red.

Based on Figure 18, there are several menus for their functions. To convert buffer values on the server, you only need to use the On Message menu, which calls the parameters into a payload so that the data parameters on the server can be displayed on the dashboard

A. System Test Result

1) Results of Validation of Electrical Parameter Values on the Device

Validation is carried out by comparing the dashboard reading results on the device with the values on the server with a duration of 1 minute and a data transmission duration of every 5 seconds.

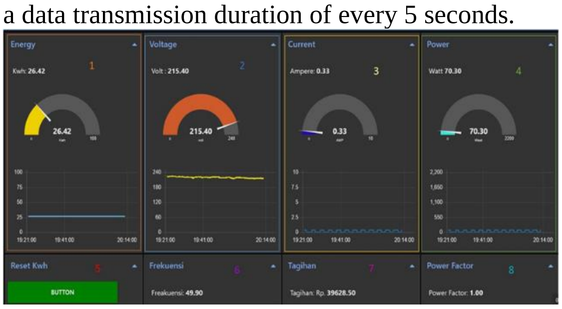

Fig 19 Dashboard Monitoring

In Figure 19, the values of the parameters that can be detected can be seen based on the number boxes

1 Energy Total
2 Voltage (Average 1 phase)
3 Current (Average 1 phase)
4 Power
5 Frequency
6 Power Factor
7 Reset Kwh

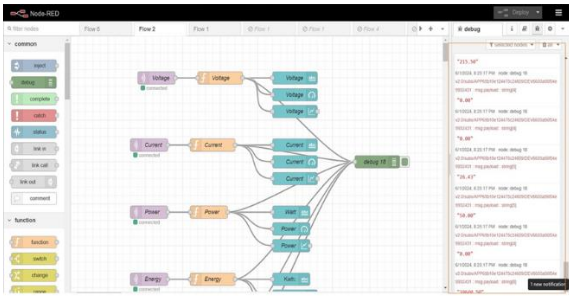

Fig 20 Data Flow on Node-RED

Based on Figure 20, the researchers obtained 10-12 data in 1 minute with the data sending process every 5 seconds. This data is the result sent by the server to Node-RED via the MQTT protocol using the subscription method.

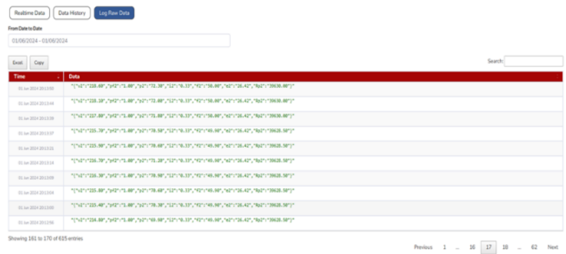

Fig 21 Data Flow on the Server

In Figure 21, the data values received by the server from the device are sent via the MQTT protocol. This data is obtained from the sensor data obtained by the PZEM 004T.

2) End To End Test Results on Each Electronic Device

End to end testing is carried out to find out what electrical parameter values are obtained by the sensor when the electronic device is working and in standby state.

a. Rice Cooker

Fig 22 Rice Cooker Data Display on Dashboard

In figure 22 is a display on the dashboard where the dashboard displays all the electrical parameters used by the dispenser when it is in standby. Based on the data on the dashboard, the number of each parameter that can be detected is voltage 216V, current 1.78A, power 385.50

Fig 23 Testing Rice cooker Using an Avometer

In Figure 23 is a display of the test results using a avometer tester. The meter pliers are used to measure the voltage and current used by the Rice Cooker when it is in standby mode. Based on data obtained when researchers tested the electrical parameters needed for the Rice Cooker when working, namely a voltage of 216V, a current of 1.92A and a power of 414.72 W.

Table 4 Error and Accuracy Rice Cooker Testing Result

	System	Testing	Error	Accuracy
Voltage	216	216	1%	99%
Current	1,78	1,92	1%	99%
Power	385,5	414,7	1%	99%

In Table 4 are the results of the error rate values for rice cooker electronic devices under working conditions where the voltage has an error rate of 1%, current 1% and watts 1%. so that based on device data it has an accuracy value of 99% in testing on rice cooker devices

b Dispensers

Fig 24 Dispensers Data Display on Dashboard

In Figure 24 is a display on the dashboard where the dashboard displays all the electrical parameters used by the dispenser when working. Based on the data on the dashboard, the number of each parameter that can be detected is voltage 229V, current 1.93A, power 429.40W.

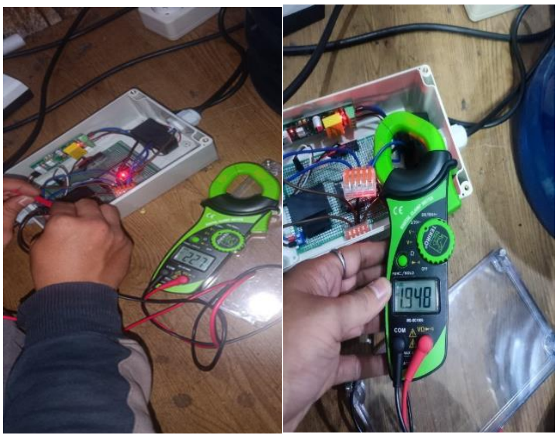

Fig 25 Testing Dispensers Using an Avometer

Figure 25 shows the results of direct testing using a tool in the form of measuring pliers which is used to determine the electrical data used by the dispenser when working. Based on data obtained when researchers tested the electrical parameters required for the dispenser, namely voltage 227V, current 1.95A and power 422.65W.

Table 5 Error and Accuracy Dispensers Testing Result

	System	Testing	Error	Accuracy
Voltage	229	227	1%	99%
Current	1,93	1,95	1%	99%
Power	429,4	422,6	1%	99%

In Table 5 are the results of the error rate values for electronic dispensers in working conditions where the voltage has an error rate of 1%, current 1% and wattage 1%. so that based on device data it has an accuracy value of 99% in testing on dispensers devices.

c. Water Heater

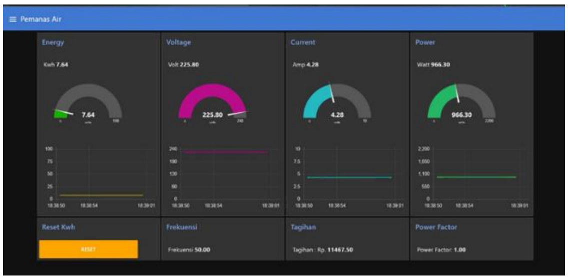

Fig 26 Water Heater Data Display on Dashboard

In Figure 26 is the display on the dashboard where the dashboard displays all the electrical parameters used by the water heater when working. Based on the data on the dashboard, the number of each parameter that can be detected is voltage 225.80V, amperage 4.28A, and power 996.30W.

Figure 27 shows the results of direct testing using tools in the form of meter pliers which are used to determine the electrical data used by the water heater when it is working. Based on data obtained when researchers tested the electrical parameters needed for air heaters when working, namely voltage 224V, amperage 4.37A and power 978.88 W.

Fig 27 Testing Water Heater Using an Avometer

Table 6 Error and Accuracy Ware Heater Testing Result

	System	Testing	Error	Accuracy
Voltage	225	224	1%	99%
Current	4,33	4,51	1%	99%
Power	966,7	1010,24	1%	99%

In Table 6 are the results of the error rate values for electronic water heaters under working conditions where the voltage has an error rate of 1%, current 1% and watts 1%. so that based on device data it has an accuracy value of 99% in testing water heating devices.

B. Quality Of Service Test Result

a. Delay

Delay is a time delay when a data packet is sent through the transmission process, from one point to another. To calculate the delay value, researchers used Microsoft Excel, by exporting data from Wireshark into CSV file format.

Table 6 MQTT Protocol Data Sample

NO	Time	Source	Destination	Protocol
1	0	192.168.137.	103.67.79.6	TCP
2	0.016585	103.67.79.6	192.168.137.	MQTT
3	1,476541	192.168.137.	103.67.79.6	TCP
4	5,064255	103.67.79.6	192.168.137.	MQTT
5	5,071537	192.168.137.	103.67.79.6	TCP

From Table 6 above, to calculate the delay, what we need is time or time information. To find out the difference in delivery time between packages, make a calculation table as follows

Table 7 Calculation of Delay on Ms. Excel

T2	T1	Delay
0,016585	0,01659	0,01659
1,476541	1,45996	1,45996
5,064255	3,58771	3,58771
5,071537	0,00728	0,00728
	Delay Avg :	1,26788

Based on the results of calculating the average delay from all data obtained from the capture results in Wireshark software with predetermined distance variations, it can be seen in Table 7.

Table 8 Average Delay

	Average value of delay (ms)		
Range (M)	5	10	15
Delay	7,804	7,952	8,761

When compared with the TIPHON standard, the quality of delay parameter service produced by the MQTT subscription protocol is very good. Based on Table 8, the average delay produced by the MQTT protocol with a distance variation of 5 – 15 meters is below 150 ms, which shows that this delay is very good.

b. Jitter

Table 9 Calculation of Jitter on Ms. Excel

delay2	delay1	Jitter
1,459956	-1,443371	2,903327
3,587714	-2,127758	5,715472
0,007282	3,580432	-3,57315
Total Jitter :		1,681883
Jitter Avg :		0,4204708

calculation to calculate jitter is to subtract the delay value 2 from delay 1. After determining the delay value between packets, the next step is to calculate the total delay, namely the total delay divided by the number of packets received minus 1.

Table 10 Average Jitter

Average value of delay (ms)			
Range (M)	5	10	15
Jitter(Ms)	7,899	8,044	8,868

When compared with the TIPHON standardization, the results of the service quality of jitter parameters produced by the MQTT protocol are good. It can be seen in Table 10 that the average jitter value produced by the MQTT protocol at a distance variation of 5-15 meters has a value between 0-75 ms, which indicates that the jitter is good

c. Throughput

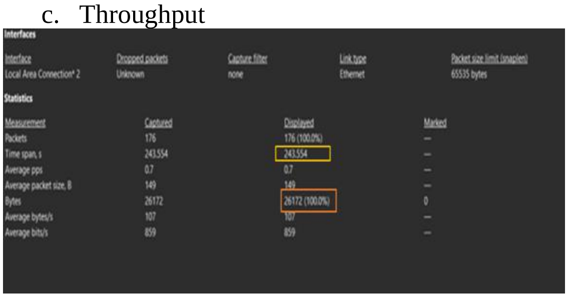

Fig 28 Throughput Calculations in Wireshark

The throughput value can be seen in average bytes in Wireshark, the average bytes value is the result of dividing bytes by timespan. However, the calculation results are still in bytes, so calculations using a formula are required

Table 11 Average Throughput

Average Throughput				
Jarak	Time(s)	Bytes	AVG Bytes	bps
5M	152,339	19610	128,726065	1029,80852
10M	197,907	22756	114,9833	919,866402
15M	243,554	26172	107,458716	859,669724

Based on the values in Table 11 when compared with the TIPHON standardization, the service quality results for throughput parameters produced by the MQTT protocol with distance variations of 5-15 meters are very good. It can be seen in the table that the throughput value produced by the MQTT protocol is above 100 bps, which indicates that the throughput is very good.

d. Packet Loss

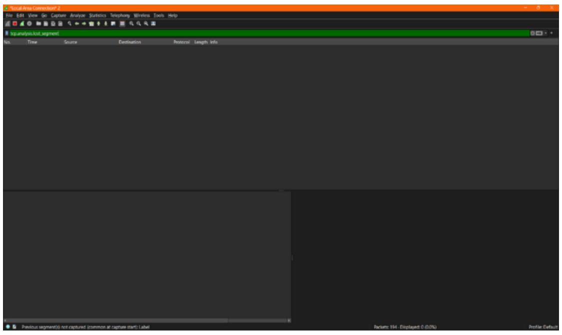

Fig 29 Packet Loss

Figure 29 shows that no packets were lost. Previously, in throughput, jitter and delay testing, each variation in distance experienced fluctuations in network quality. However, in each of these tests, no packet loss or missing packets were found. This is caused by the absence of ACK data feedback from the platform to Wireshark, so the detected packet loss value is 0%. In addition, packet loss usually occurs when sending large data, which will only reduce the packet size, not cause packet loss.

When compared with the TIPHON standardization, the service quality results for packet loss parameters produced by the MQTT protocol are very good. It can be seen in the picture that the packet loss value produced by the MQTT protocol

IV. CONCLUSION

Based on the discussion regarding the Design and Development of an Internet of Things System for Monitoring Electrical Energy in Home Electronic Devices, several conclusions are obtained as follows.

1 Design of an Internet of Things System for Monitoring Electrical Energy in Home Electronic Devices has succeeded in sending end to end data with the following value configuration: current (average 1 phase) of 2-5 Amp, voltage of 220-230V.

2 The accuracy quality of the device is very high with an average accuracy value of 99% and an average error rate of 1%. This is proven by the calculations that researchers carried out during end to end testing, these results are also very relevant to the specifications of the sensor used.

3 Service quality in the delay, throughput and packet loss categories using Wireshark software has an index value of 4 in accordance with the TIPHON standard, while for the jitter category it has an index value of 3 in accordance with the TIPHON standard

V. REFERENCES

[1] Shilvina Widi, “Konsumsi Energi Listrik Indonesia Kembali Meningkat pada 2022.”

Accessed: Nov. 27, 2023. [Online]. Available: https://dataindonesia.id/energi-sda/detail/konsumsi-listrik-indonesia-capai-18341-juta-boe-pada-2022

[2] D. A. Putra and R. Mukhaiyar, “Monitoring Daya Listrik Secara Real Time,” *Voteteknika (Vocational Tek. Elektron. dan Inform.*, vol. 8, no. 2, p. 26, 2020, doi: 10.24036/voteteknika.v8i2.109138.

[3] G. B and D. Sabu George Asso, “IJERT_IoT_Based_Smart_Energy_Management,” *J. Int. Ris. Teknol. Tek.*, 2021.

[4] N. P. Windryani, N. Bogi, and R. Mayasari, “Analisa Perbandingan Protokol Mqtt Dengan Http Pada Iot Platform Patriot Comparison Analysis Between Mqtt and Http Protocol in Patriot Iot Platform,” *-Proceeding Eng.*, vol. 6, no. 2, pp. 3192–3199, 2019.

[5] D. Azizi and V. Arinal, “Sistem Monitoring Daya Listrik Menggunakan Internet of Thing (Iot) Berbasis Mobile,” *J. Indones. Manaj. Inform. dan Komun.*, vol. 4, no. 3, pp. 1808–1813, 2023, doi: 10.35870/jimik.v4i3.409.

[6] S. George Asso and C. Dept, “IoT Based Smart Energy Management System using Pzem-004t Sensor & Node MCU,” 2021. [Online]. Available: https://www.amazon.in/Easy-Electronics-16x2-

[7] P. Harahap, F. I. Pasaribu, and M. Adam, “Prototype Measuring Device for Electric Load in Households Using the Pzem-004T Sensor,” 2020, doi: 10.33258/birex.v2i3.1074.

[8] R. Z. Pratama and H. Nurwarsito, “Monitoring Penggunaan Daya Listrik menggunakan Protokol MQTT berbasis Web,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 11, pp. 10820–10826, 2019, [Online]. Available: https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/6799

[9] A. D. Prakoso, “Analisis Perbandingan Kualitas Layanan Sistem Antara Protokol HTTP dan MQTT Pada Monitoring Kelembaban Tanah,” *PRotek J. Ilm. Tek. Elektro*, vol. 7, no. 2, pp. 55–59, 2020, doi: 10.33387/protk.v7i2.1658.

[10] D. Torres, J. P. Dias, A. Restivo, and H. S. Ferreira, “Real-timeTorres, D., Dias, J. P., Restivo, A., & Ferreira, H. S. (2020). Real-time Feedback in Node-RED for IoT Development: An Empirical Study. Proceedings of the 2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applicat,” *Proc. 2020 IEEE/ACM 24th Int. Symp. Distrib. Simul. Real Time Appl. DS-RT 2020*, 2020.

[11] B. Dodiya and U. K. Singh, “Malicious Traffic analysis using Wireshark by collection of Indicators of Compromise,” *Int. J. Comput. Appl.*, vol. 183, no. 53, pp. 1–6, 2022, doi: 10.5120/ijca2022921876.

[12] W. Chaiyong and S. Sonasang, “Applications of energy monitoring using the IoT,” *SNRU J. Sci. Technol.*, vol. 14, no. 2, p. 245041, 2022, doi: 10.55674/snrujst.v14i2.245041.

[13] Lulu Sabillah and R. Hidayat, “Sistem Monitoring Pemakaian Energi Listrik Pada Kamar Kost Menggunakan Aplikasi Blynk Berbasis Internet of Things,” *J. Komput. dan Elektro Sains*, vol. 1, no. 2, pp. 25–29, 2023, doi: 10.58291/komets.v1i2.104.

[14] A. Imran and M. Rasul, “Pengembangan Tempat Sampah Pintar Menggunakan Esp32,” *J. Media Elektr.*, vol. 17, no. 2, pp. 2721–9100, 2020, [Online]. Available: https://ojs.unm.ac.id/mediaelektrik/article/view/14193

[15] T. Venkatesh, M. Sireesha, P. Suresh, K. Ganesh Kumar, and B. RameshNaidu, “Pengukur Energi Otomatis menggunakan ESP-32 berkemampuan WiFi,” 2021. [Online]. Available: www.ijatest.org