Design and Implementation of a NodeMCU-Based Lamp Power Control and Energy-Metering Device Using the PZEM-004T Module

Sultan Aziz Syaifullah[a,1], Muhammad Arfah Asis[a,2]

[a] *Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Muslim Indonesia, Jl. Urip Sumoharjo km.05, Makassar dan 90231, Indonesia*
[1] *13020200073@umi.ac.id*; [2] *muh.arfah.asis@umi.ac.id*
**corresponding author*

ARTICLE INFORMATION

Received : 12 – 06 – 2024
Revised : 05 – 10 – 2025
Published : 29 – 10 – 2025

Keywords:
NodeMCU ESP8266
Internet of Things
PZEM-004T
Thinger.io

ABSTRACT

This study aims to design a NodeMCU- and PZEM-004T-based device for power control and energy metering, serving as the central unit for control and monitoring, thereby enabling users to remotely control and monitor household lamp power consumption. An experimental method was employed to evaluate the performance of the developed controller and power/energy meter. The NodeMCU ESP8266 functions as the microcontroller to execute the device's program and support remote control, while the PZEM-004T module performs lamp power measurements. Through wireless communication, the system delivers power (W), kilowatt-hours (kWh), and cost (IDR) information to a smartphone via the Thinger.io web interface. Users can set the lamp's ON–OFF status and obtain real-time power measurements. Based on testing results, the prototype supports real-time control and monitoring with an average power-measurement error of 0.10%.

I. Introduction

Electricity is one of today's fundamental human needs. At the industrial scale, electricity is utilized 24 hours a day; even a brief outage of a few seconds can cause losses amounting to millions or even billions. The Indonesian Retailers Association (APRINDO) recorded potential losses to its members of IDR 90–100 billion for every six hours of power outage [1]. In contrast to industry, household electricity does not need to be on continuously for 24 hours because it is typically switched on only when needed and turned off when not in use, especially for lighting. However, homeowners sometimes forget to turn off light switches in certain situations. This can lead to negative consequences such as wasteful electricity consumption and an increased risk of short circuits. A household's power usage depends on consumption patterns: the more appliances used, the greater the power drawn, which can result in excessive current load.

Rapid technological developments across various fields in recent years have significantly influenced societal habits and lifestyles. Electronic technology has advanced swiftly as well, particularly technologies related to control systems, with a persistent demand for easy-to-operate solutions. Remote control is no longer novel; consequently, many remote-control applications and scenarios have been explored.

Home automation—popularized for years under the term "smart home"—introduces automatic control of devices and appliances within residences and buildings, including remote control capabilities [2]. Several remote-control systems have been developed, each focusing on different applications and scenarios. Examples include an Android-based remote control system by Immanuel [3]; remote switch control via Short Message Service (SMS) by Affan Bachri and Nurhadiyan [4][5]; and Internet of Things (IoT)–based remote control of electrical switches by Berlianti and Amir using Arduino with a Wi-Fi module and the Blynk application [6][7].

Prior research primarily emphasized lamp control and did not incorporate an energy-metering feature. Therefore, this study proposes the "Design and Development of a NodeMCU-Based Lamp Power Control and Energy Meter Using the PZEM-004T Module", employing the Thinger.io web platform as the interface for remote access and monitoring.

II. Method

This study discusses the design and development of a controller and energy-metering device for lamps. The design encompasses both hardware and software; an experimental method is employed

A. Device Design

The device design constitutes a simple control system to operate lamp ON/OFF switching and perform electrical power/energy measurement. The overall circuit integrates a NodeMCU ESP8266, a PZEM-004T sensor, a relay, a lamp, and ancillary components. All parts are assembled into a single package referred to as the device design circuit. The complete circuit diagram of the device is shown in Figure 1.

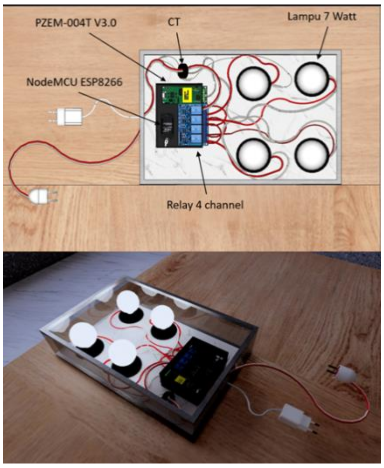


Figure 1. Tool Design

1) Internet of Things (IoT)

The Internet of Things (IoT) has emerged as a distinct research domain alongside advances in the internet and other communication media. As technological needs continue to grow, so too does the volume of related research. IoT represents one such line of inquiry, optimizing access to in-room devices for smart-home applications [8].

2) Thinger.io Web Server

Thinger.io is a platform that has attracted considerable interest within the scientific and technology communities. It provides a ready-to-use cloud service for connecting devices to the Internet, enabling remote sensing and actuation over the web. The platform offers a free tier for connecting a limited number of devices, and it also supports self-hosted deployments, allowing users to manage their own data and devices without any imposed limits [9][10].

3) NodeMCU ESP8266

The ESP8266—part of Espressif Systems' ESP series—can run firmware that uses the Lua scripting language. The term "NodeMCU" originally referred to the firmware rather than the development hardware; in practice, it is also used to denote the ESP8266-based development board (often likened to an Arduino-style board for the ESP8266). Beyond Lua, NodeMCU is supported in the Arduino IDE by adding the appropriate board URL via the Board Manager [11][12][13].

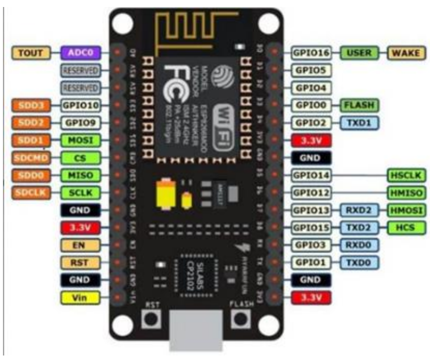


Figure 2. Mapping Pin NodeMCU ESP 8266[14]

4) Arduino IDE

IDE stands for *Integrated Development Environment*, i.e., an integrated environment used for software development. It is called an “environment” because it provides the tools through which Arduino can perform embedded tasks via a programming syntax. Arduino uses a language similar to C. The Arduino programming language (Sketch) has been adapted to simplify programming for beginners compared to the original language. Before entering the market, Arduino microcontroller ICs are preloaded with a program called a *bootloader*, which acts as an intermediary between the Arduino compiler and the microcontroller [15].

5) Relay Module

A relay is a type of switch actuated by current, comprising a low-voltage coil wound around a core. When current flows through the coil, an iron armature is attracted toward the core. This armature is linked to a spring-loaded lever. As the armature is pulled in, the common contact switches from the normally-closed (NC) to the normally-open (NO) contact. Relays enable control of AC motors with a DC control circuit, or other loads where the control-circuit voltage differs from the load voltage [16].

Figure 3. Relay Module [17]

6) PZEM-004T Module

The PZEM-004T is a multifunction sensor module designed to measure power, voltage, current, and energy in an electrical line. It integrates both a voltage sensor and a current transformer (CT). The module is intended for indoor use, and the connected load must not exceed the specified rating [18].

Figure 4. PZEM-004T V3.0 [19]

III. Results and Discussion

A. Research Results

1) System Block Diagram

This system is designed to control a lamp using a relay as the ON/OFF switch and to measure the lamp’s electrical power using the PZEM-004T, whose readings are processed by the NodeMCU ESP8266. The processed data are then presented via the Thinger.io web interface. The basic concept is as follows.

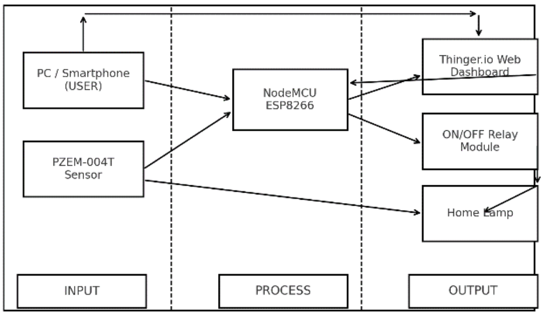


Figure 5. Block Diagram

2) Flowchart of the Lamp Control and Power-Metering Device

The workflow begins with inputs, proceeds through processing steps, and produces the desired outputs. To clarify the design and the analysis of the system workflow, the general flowchart is presented below.

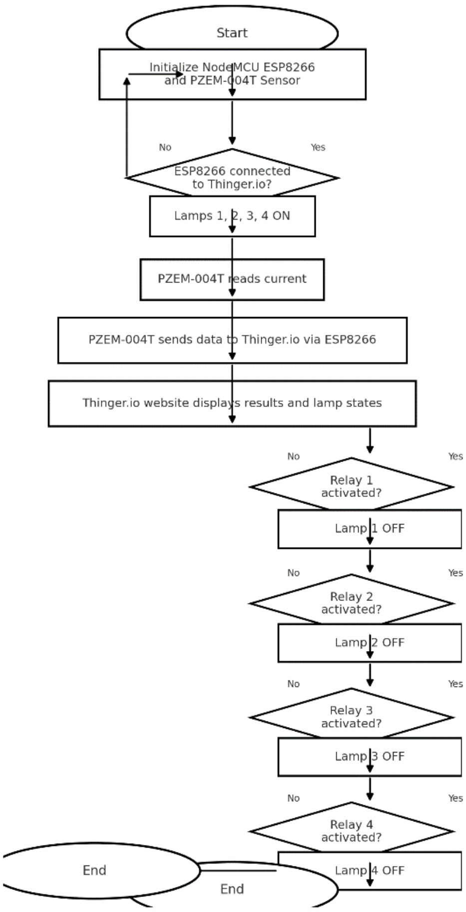


Figure 6. Flowchart of the Lamp Control and Power-Metering Device

3) Design Outcome: NodeMCU- and PZEM-004T-Based Lamp Power Controller and Energy Meter

This research produces a design comprising several stages, starting from component assembly to end-to-end testing. The overall testing evaluates the performance and success rate of the design. Based on the results, we analyze how each interconnected component operates to form the NodeMCU ESP8266– and PZEM-004T–based lamp control and power-metering system. The design consists of a 5 VDC (USB) power supply, a NodeMCU ESP8266, a 4-channel relay, a PZEM-004T module, and a lamp as the target load.

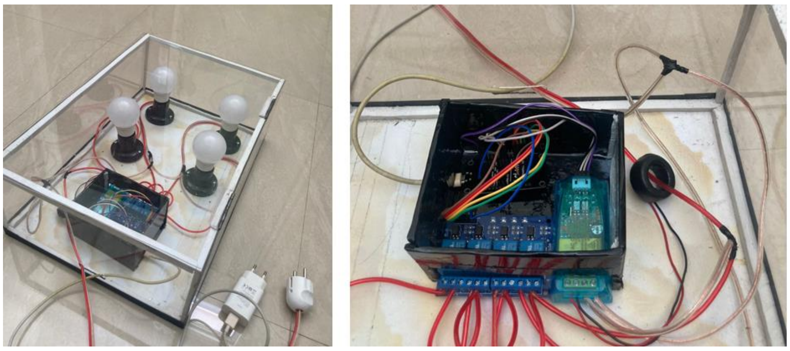


Figure 7. External and internal views of the device

4) Hardware Circuit of the NodeMCU- and PZEM-004T-Based Lamp Power Controller and Energy Meter

The detailed hardware circuit of the NodeMCU- and PZEM-004T-based lamp power controller and energy meter is shown in Figure 8.

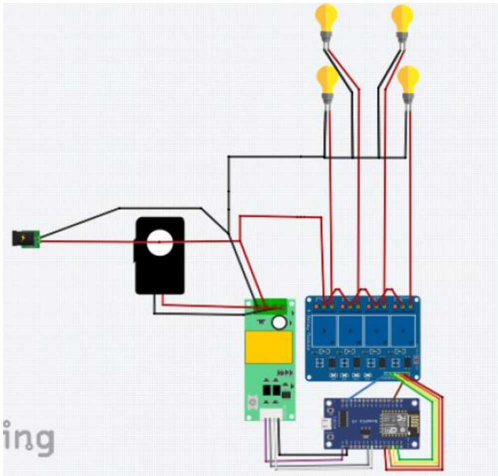


Figure 8. Hardware circuit of the NodeMCU- and PZEM-004T-based lamp power controller and energy meter

B. Discussion

a. Device Testing

Using the arithmetic mean, we obtained the average delay values for switching the lamp OFF and ON, as summarized in Table 1.

$$\text{Average Delay} = \frac{Total\ delay}{Total\ delay\ received} \quad (1)$$

Table 1. Lamp Control Delays

No	Lamp	Delay(s)	Lamp Condition
1	Lamp 1	3,00	On
2	Lamp 2	1,28	On
3	Lamp 3	0,98	On
4	Lamp 4	1,03	On
5	Lamp 1	1,15	Off
6	Lamp 2	2,56	Off
7	Lamp 3	2,63	Off
8	Lamp 4	1,11	Off
Average Delay		1,71	

Table 1 presents the test results for lamp control. The highest observed delay was 3.00 seconds for Lamp 1 in the ON state, while the lowest delay was 0.98 seconds for Lamp 3 in the ON state. The overall average delay across Lamps 1, 2, 3, and 4 (for both ON and OFF states) was 1.71 seconds.

Table 2. PZEM-004T Test Results (Lamp 1 ON)

Time (2 Minutes)	Cost (Rp)	Power (Watt)	Actual Power (Watt)	Energy (kWh)
2:49:27 AM	24.97	6.80	7	0.0166
2:51:16 AM	72.98	6.80		0.0486
2:53:05 AM	120.08	6.69		0.0800
2:54:54 AM	169.18	6.69		0.1127
2:56:43 AM	217.77	6.69		0.1451
2:58:33 AM	266.87	6.69		0.1779
Average power measurement results		6,72	-	-

Based on Table 2—PZEM-004T measurements with Lamp 1 ON—the average power over a 10-minute observation window was 6.72 W.

Table 3. PZEM-004T Test Results (Lamps 1 and 2 ON)

Time (2 Minutes)	Cost (Rp)	Power (Watt)	Actual Power (Watt)	Energy (kWh)
3:00:22 AM	315.59	13.30	14	0.2103
3:02:12 AM	364.84	13.19		0.2432
3:04:00 AM	413.08	13.10		0.2753
3:05:49 AM	461.46	13		0.3076
3:07:38 AM	509.84	13		0.3398
3:09:27 AM	557.81	13.10		0.3718
Average power measurement results		13,11	-	-

From Table 3—PZEM-004T measurements with Lamps 1 and 2 ON—the average power over 10 minutes was 13.11 W.

Table 4. PZEM-004T Test Results (Lamps 1, 2, and 3 ON)

Time (2 Minutes)	Cost (Rp)	Power (Watt)	Actual Power (Watt)	Energy (kWh)
3:11:16 AM	606.84	19.60	21	0.4045
3:13:06 AM	655.46	19.5		0.4369
3:14:55 AM	704.64	19.39		0.4697
3:16:44 AM	753.94	19.39		0.5026
3:18:34 AM	803.29	19.29		0.5355
3:20:22 AM	852.25	19.29		0.5681
Average power measurement results		19,41	-	-

According to Table 4—PZEM-004T measurements with Lamps 1, 2, and 3 ON—the average power over 10 minutes was 19.41 W.

Table 5. PZEM-004T Test Results (Lamps 1, 2, 3, and 4 ON)

Time (2 Minutes)	Cost (Rp)	Power (Watt)	Actual Power (Watt)	Energy (kWh)
3:22:12 AM	901.80	25.70	28	0.6012
3:24:01 AM	950.95	25.60		0.6339
3:25:50 AM	1000.20	25.60		0.6668
3:27:40 AM	1050.61	25.60		0.7004
3:29:29 AM	1100.11	25.60		0.7334

Time (2 Minutes)	Cost (Rp)	Power (Watt)	Actual Power (Watt)	Energy (kWh)
3:31:18 AM	1150.24	25.60		0.7668
Average power measurement results		25,61	-	-

As shown in Table 5—PZEM-004T measurements with Lamps 1–4 ON—the average power over 10 minutes was 25.61 W.

Table 6. Overall PZEM-004T Test Results

Time (2 Minutes)	Cost (Rp)	Power (Watt)	Actual Power (Watt)	Energy (kWh)
6:24:20 PM	11.72	25	28	0.0078
6:26:20 PM	12.29	25		0.0081
6:28:19 PM	12.85	25		0.0085
6:32:20 PM	13.99	24.89		0.0093
6:30:20 PM	13.42	25		0.0089
6:34:21 PM	14.56	24.89		0.0097
6:38:23 PM	15.70	24.89		0.0104
6:36:21 PM	15.13	24.89		0.0100
6:40:23 PM	16.27	24.89		0.0108
6:42:24 PM	16.85	24.89		0.0112

From Table 6, the measured power reported by the device and the nominal power of the load show slight differences. Accordingly, the overall percentage measurement error can be computed as follows [20]: (formula omitted here, as in the source).

$$\%\text{Power measurement error } 1 = \frac{25-28}{28} \times 100\% = 0{,}10\%$$

$$\%\text{Power measurement error } 2 = \frac{25-28}{28} \times 100\% = 0{,}10\%$$

$$\%\text{Power measurement error } 3 = \frac{25-28}{28} \times 100\% = 0{,}10\%$$

$$\%\text{Power measurement error } 4 = \frac{24{,}89-28}{28} \times 100\% = 0{,}11\%$$

$$\%\text{Power measurement error } 5 = \frac{25-28}{28} \times 100\% = 0{,}10\%$$

$$\%\text{Power measurement error } 6 = \frac{24{,}89-28}{28} \times 100\% = 0{,}11\%$$

$$\%\text{Power measurement error } 7 = \frac{24{,}89-28}{28} \times 100\% = 0{,}11\%$$

$$\%\text{Power measurement error } 8 = \frac{24{,}89-28}{28} \times 100\% = 0{,}11\%$$

$$\%\text{Power measurement error } 9 = \frac{24{,}89-28}{28} \times 100\% = 0{,}11\%$$

$$\%\text{Power measurement error } 10 = \frac{24{,}89-28}{28} \times 100\% = 0{,}11\%$$

Based on the calculation of the power measurement error above, the average value of the measurement error can be calculated as follows:

$$= \frac{0{,}10+0{,}10+0{,}10+0{,}11+0{,}10+0{,}11+0{,}11+0{,}11+0{,}11+0{,}11}{10}$$

$$= \frac{1{,}06}{10}$$

$$= 0{,}10\%$$

b. Testing via the Thinger.io Website

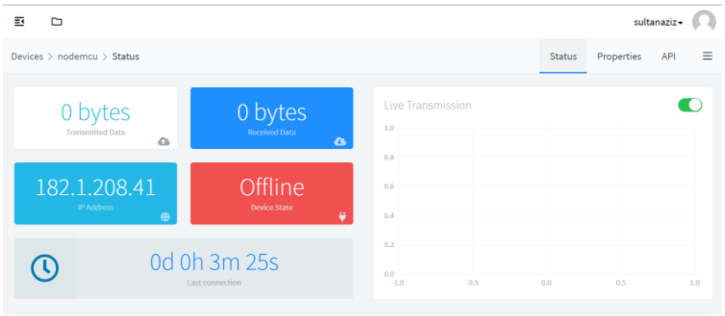


Figure 9. Offline dashboard view

When the system is offline, the dashboard displays: white box—transmitted data: 0 bytes; dark-blue box—received data: 0 bytes; light-blue box—IP address: 182.1.208.4; red box—device state: offline.

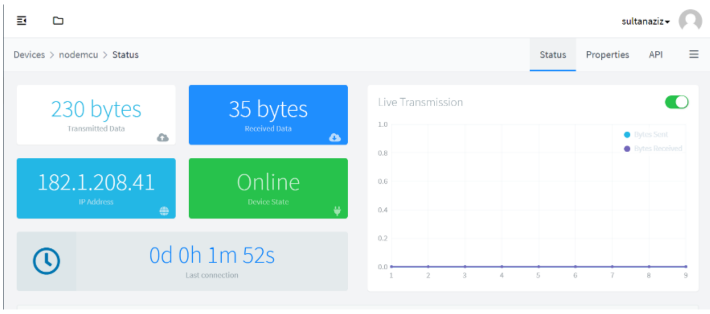


Figure 10. Online dashboard view

When the system is online, the dashboard shows: white box—transmitted data: 230 bytes; dark-blue box—received data: 35 bytes; light-blue box—IP address: 182.1.208.41; red box—device state: online.

Figure 11. Measurement display with Lamp 1 ON

With Lamp 1 ON, the readings are power = 6.70 W, energy = 0.1933 kWh (from PZEM-004T), and cost = IDR 304.04 per kWh.

Figure 12. Measurement display with Lamps 1 and 2 On

With Lamps 1 and 2 ON, the readings are power = 13.10 W, energy = 0.3857 kWh, and cost = IDR 578.54 per kWh.

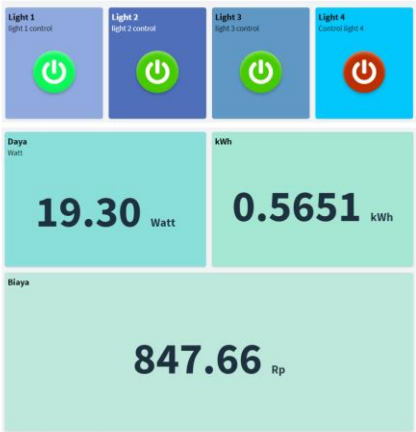


Figure 13. Measurement display with Lamps 1, 2, and 3 ON

With Lamps 1–3 ON, the readings are power = 19.30 W, energy = 0.5651 kWh, and cost = IDR 847.66 per kWh.

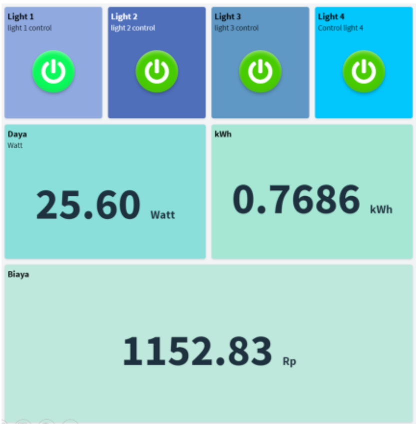
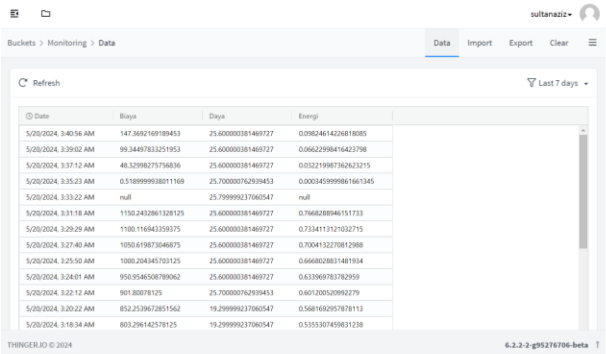


Figure 14. Measurement display with Lamps 1, 2, 3, and 4 ON

With Lamps 1–4 ON, the readings are power = 25.60 W, energy = 0.7686 kWh, and cost = IDR 1,152.83 per kWh.

Figure 15. Measurement display with all lamps OFF

When all lamps are OFF, a small residual power is still detected. This occurs because internal components—namely the PZEM-004T (sensor) and the relay (control circuit)—remain powered even when the lamp load is switched off, causing a small, persistent power draw.



Date	Biaya	Daya	Energi
5/20/2024, 3:40:56 AM	147.3692169189453	25.600000381469727	0.09824614226818085
5/20/2024, 3:39:02 AM	99.34497833251953	25.600000381469727	0.06622998416423798
5/20/2024, 3:37:12 AM	48.32998275756836	25.600000381469727	0.03221987362623215
5/20/2024, 3:35:23 AM	0.5189999938011169	25.700000762939453	0.0003459999861661345
5/20/2024, 3:33:22 AM	null	25.799999237060547	null
5/20/2024, 3:31:18 AM	1150.2432861328125	25.600000381469727	0.7668288946151733
5/20/2024, 3:29:29 AM	1100.116943359375	25.600000381469727	0.7334113121032715
5/20/2024, 3:27:40 AM	1050.619873046875	25.600000381469727	0.7004132270812988
5/20/2024, 3:25:50 AM	1000.204345703125	25.600000381469727	0.6668028831481934
5/20/2024, 3:24:01 AM	950.9546508789062	25.600000381469727	0.633969783782959
5/20/2024, 3:22:12 AM	901.80078125	25.700000762939453	0.601200520992279
5/20/2024, 3:20:22 AM	852.2539672851562	19.299999237060547	0.5681692957878113
5/20/2024, 3:18:34 AM	803.296142578125	19.299999237060547	0.5355307459831238

Figure 16. PZEM-004T records in Thinger.io Data Buckets

The Thinger.io data buckets present timestamped records (month, date, year, time) with cost (IDR), power (W), and energy (kWh).

Figure 17. Condition with mains connected (relay OFF)

Figure 18. Condition with mains disconnected (relay ON)

Relay testing verifies that the relay responds to programmed commands issued from the Thinger.io web interface: when the relay button is pressed, the relay disconnects the mains; pressing the button again reconnects the mains. The outcomes are illustrated in Figures 17 and 18.

Conclusion

Based on the results of research conducted on the design of a control system and electricity meter for lamps based on NodeMCU and the PZEM-004T Module, it can be concluded that it was successfully built with an average lamp control delay of 1.71 seconds and an average power measurement error of 0.10%. The sensor is able to detect the power (Watts), energy (kWh), and cost (Rp) of electricity expenditure of the lamps and directly sends data to the Thunger.io website as a medium for direct control and monitoring that can be accessed using a smartphone/PC. With the NodeMCU, which has a Wi-Fi module feature, it makes it easier for users to control and monitor anywhere and anytime. However, the design of this tool still has shortcomings, namely the device must be continuously connected to a Wi-Fi network to always work.

Based on the test results, this tool still has many shortcomings. Several things must be considered, therefore the author suggests several things in the construction of this tool, developing the design of the controller and electricity meter for lamps on a wider scale, such as being applied to large companies and offices. Adding an automatic control feature that allows the NodeMCU to adjust the brightness or timing of lights based on measured energy consumption data to help save energy. The Thinger.io website is used as the medium, and it is hoped that it can be developed using a dedicated application for greater safety and effectiveness.

References

[1] N. Sari, "5 Fakta Listrik Padam, Kerugian Diperkirakan Triliunan Rupiah hingga Tuntut Ganti Rugi Rp 5.000.," *Kompas.com*, 2019. https://megapolitan.kompas.com/read/2019/08/07/06113891/5-fakta-listrik-padam-kerugian-diperkirakan-triliunan-rupiah-hingga (accessed Oct. 20, 2023).

[2] R. A. Ramlee *et al.*, "Bluetooth Remote Home Automation System Using Android Application," *Int. J. Eng. Sci.*, vol. 2, no. 1, pp. 149–153, 2013.

[3] I. Warangkiran, I. S. T. G. Kaunang, A. S. M. Lumenta, and A. M. R. St, "Perancangan Kendali Lampu Berbasis Android," *J. Tek. Elektro dan Komput.*, vol. 3, no. 1, pp. 1–8, 2014.

[4] A. Bachri, "Rancang Bangun Smart Kontrol Lampu Penerangan Jalan Umum Berbasis SMS Gateway," *J. JE-UNISLA Electron. Control. Telecomunication, Comput. Inf. Power Syst.*, vol. 4, no. 2, pp. 256–264, 2019, [Online]. Available: https://jurnalteknik.unisla.ac.id/index.php/elektronika/article/view/363

[5] M. T. Nurhadiyan and A. E. Saputro, "Sistem Kendali Saklar Lampu Jarak Jauh Menggunakan Sms Berbasis Mikrokontroleratmega328/Arduino Uno," *Prosisko*, vol. 6, no. 2, pp. 144–152, 2019, [Online]. Available: http://e-jurnal.lppmunsera.org/index.php/PROSISKO/article/view/1635

[6] R. Berlianti and Fibriyanti, "Perancangan Alat Pengontrolan Beban Listrik Satu Phasa Jarak Jauh Menggunakan Aplikasi Blynk Berbasis Arduino Mega," *Sain, Energi Teknol. Ind.*, vol. 5, no. 1, pp. 17–26, 2020.

[7] A. Amir, "Rancang Bangun Purwarupa Alat Monitoring dan Kontrol Beban Satu Fasa Berbasis IoT (INTERNET OF THINGS)," *J. Tek. Elektro Univ. Islam Sultan Agung*, vol. 20, no. 1, pp. 29–33, 2018.

[8] S. Windiyati, Satrianovari, "Prototype Smarthome Berbasis IoT (INTERNET OF THINGS)," *Jik*, vol. 14, no. 1, pp. 28–36, 2023.

[9] M. A. P. and J. M. M. Alvaro Luis Bustamante, "Thinger.io: An open source platform for deploying data fusion applications in IoT environments," *Sensors (Switzerland)*, vol. 19, no. 1044, p. 4, 2019, doi: 10.3390/s19051044.

[10] D. Indra, E. I. Alwi, and M. Al Mubarak, "Prototipe Sistem Kontrol Pemadam Kebakaran Pada Rumah Berbasis Arduino Uno dan ESP8266," *Komputika J. Sist. Komput.*, vol. 11, no. 1, pp. 1–8, 2021, doi: 10.34010/komputika.v11i1.4801.

[11] A. Rachmansyah, R. Satra, and M. A. Mude, "Perancangan Alat Pemberi Makan dan Monitoring Sisa Pakan Hewan Pemeliharaan Berbasis Microcontroller," *Bul. Sist. Inf. dan Teknol. Islam*, vol. 3, no. 1, pp. 26–33, 2022.

[12] Z. Fahmi, R. Satra, and F. Fattah, "Monitoring Ketinggian Air Berbasis NodeMCU dengan Menggunakan Web Resposive," *Bul. Sist. Inf. dan Teknol. Islam*, vol. 2, no. 1, pp. 1–5, 2021, doi: 10.33096/busiti.v2i1.688.

[13] A. R. Manga, D. Indra, and N. Nurhalima, "Sistem Monitoring Infus Menggunakan ESP8266," *Bul. Sist. Inf. dan Teknol. Islam*, vol. 1, no. 2, pp. 118–123, 2021, doi: 10.33096/busiti.v1i2.927.

[14] Microcontrollerslab.com, "ESP8266 NodeMCU ADC using Arduino IDE – Measure Analog Voltage," *microcontrollerslab.com*, 2022. https://microcontrollerslab.com/esp8266-nodemcu-adc-tutorial-measure-analog-voltage/ (accessed Jun. 03, 2024).

[15] S. P. Aji, "Alat Monitoring Tetesan Infus Menggunakan Web Secara Online Berbasis ESP8266 Dengan Pemrograman Arduino IDE," *J. Tek Elektron. Fak. Tek. UNY*, vol. 6, no. 8, p. 3, 2017.

[16] D. A. O. Turang, "Pengembangan Sistem Relay Pengendalian dan Penghematan Pemakaian Lampu Berbasis Mobile," *Semin. Nas. Inform.*, p. 78, 2015.

[17] Kuongshun, "Modul Relay 4 Channel," *kuongshun.com*, 2024. https://id.szks-kuongshun.com/uno/uno-board-shield/4-channel-relay-module-with-light-coupling-5v.html

[18] M. Nursamsi Adiwiranto and C. Budi Waluyo, "Prototipe Sistem Monitoring Konsumsi Energi Listrik Serta Estimasi Biaya Pada Peralatan Rumah Tangga Berbasis Internet of Things," *ELECTRON J. Ilm. Tek. Elektro*, vol. 2, no. 2, pp. 13–22, 2021, doi: 10.33019/electron.v2i2.2.

[19] InivatorGuru, "Modul Komunikasi AC PZEM-004T V3.0 Panduan Pengguna," *InivatorGuru*. https://manuals.plus/id/innovatorsguru/ac-communication-module-pzem-004t-v3-0-manual

[20] T. H. Solin, "Rancang Bangun Monitoring Kerja Konverter Listrik Pada Pengisian Baterai Aki Berbasis Internet of Things," Universitas Muhammadiyah Sumatera Utara, 2023.