Review Article

IoT-based Real-Time Monitoring System to Optimize Capacitor Bank Performance on Waste Chopping Machines in Cikawao Village, Bandung

[1*]Tjahjo Adiprabowo, [1]Dani Ramdani, [1]Andriana, [1]Ganjar Turesna, [1]Zulkarnain, [2]Raden Sukmana, [3]Rahma Nur Fauziah, [2]Lara Dwi Cinta Hendrawan, [4]Aura Anbiya Direja, [4]Tresna Intan Maulani Sambas, [3]Della Agni Resnitiani, [1]Prima Dhanureksa

1 Department of Electrical Engineering, Langlangbuana University, Bandung, West Java, Indonesia
2 Department of Informatic Engineering, Langlangbuana University, Bandung, West Java, Indonesia
3 Department of Management, Langlangbuana University, Bandung, West Java, Indonesia
4 Faculty of Law, Langlangbuana University, Bandung, West Java, Indonesia

* Corresponding Author: tjahjo.adiprabowo@unla.ac.id

Abstract: The use of waste chopping machines in Cikawao Village, Bandung depends on the electricity that supplies the machine. To supply electrical energy to the chopping machine, bank capacitors are needed which function to stabilize electrical power. A capacitor bank is a set of capacitors installed in parallel. Specifically, the function of the capacitor bank is to improve the power factor, reduce useless reactive power, reduce power losses on the network, increase the available power capacity, and also stabilize the system voltage. In this article review, the hope is that the voltage can be monitored in real-time using the Internet of Things (IoT) and focus on voltage sensors and current sensors and controllers that work like relays when the voltage is over, it will automatically turn off. And voltage (V) and current (I) can be monitored in real-time on the available monitors. In the process of building this IoT, a protocol is needed, to control and monitor voltage and electric current can use a microcontroller that can connect to the internet, which means using a WiFi module and connecting it to the Internet server using the MQTT Protocol.

Keywords: real-time monitoring, Voltage, Current, Internet of Things, waste chopping machines, Capacitor banks

check for updates

Citation: Adiprabowo, T., Ramdani, D., Andriana, Turesna, G., Zulkarnain, Sukmana, R., Fauziah, R. N., Hendrawan, L. D. C., Direja, A. A., Sambas, T. I. M., Resnitiani, D. A., & Dhanureksa, P. (2025). IoT-based real-time monitoring system to optimize capacitor bank performance on waste chopping machines in Cikawao Village, Bandung. *Iota*, 5(1). https://doi.org/10.31763/iota.v5i1.875

Academic Editor: Adi, P.D.P
Received: January 03, 2025
Accepted: January 20, 2025
Published: February 04, 2025

1. Introduction

Currently, the real-time monitoring system has become an important requirement for any business field. In this research case is the electrical stability factor of the garbage chopper machine in Cikawao village, Bandung. The problem is the need for unstable electrical power, so the most important thing is to ensure that the voltage and electric current are stable according to the needs of this waste shredder. The important component is the capacitor bank. Capacitor banks are installed in parallel and are used to improve the power factor in the function of the electrical system [1]. The function of the capacitor bank is to improve the power factor, reduce useless reactive power, reduce power losses on the network, increase the available power capacity, and stabilize the voltage on the system, in this case, the power needed for the waste chopper machine.

How capacitor banks work is When an inductive load (electric motor) operates, it requires a reactive power that causes a low power factor, and this capacitor bank supplies the reactive power needed by the inductive load, this can reduce the current that must be supplied from the main power source, as a result, power losses are reduced and system voltage improves. The chopping machine used uses an electric motor with a large power, during start-up, the motor requires a high starting current, and the capacitor bank helps to reduce the impact of this high starting current, as a result, voltage drop during start-up can be minimized. Moreover, the outline of the Capacitor Bank can be seen in Figure 1, and specifically the Capacitor Bank Chart can be seen in Figure 2.

Figure 1. Capacitor Bank (sources: galleoncy)

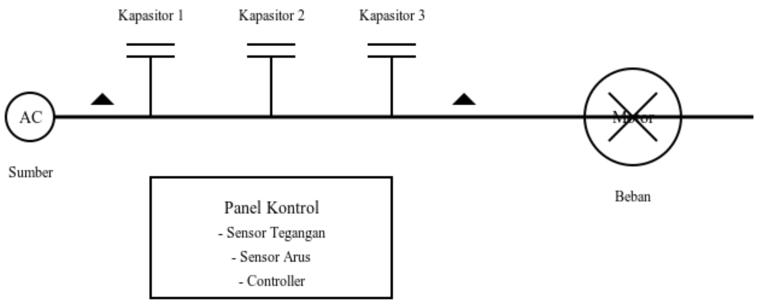

Figure 2. Capacitor Bank Chart

2. Literature Review

2.1 *Node MCU ESP 8266 & ESP WROOM 32*

NodeMCU is an open-source IoT platform, we can build IoT applications for example using MQTT Protocol to be able to display messages in the form of strings or characters from sensor data [13,14,15,16]. NodeMCU is made with the ESP8266 microcontroller whose main feature is built-in WiFi. Like the Arduino Microcontroller or ATMega 328p, NodeMCU also uses the Arduino IDE with the C++ programming language, making it very easy for users to carry out the calibration process and also read sensors by adding several libraries needed for voltage, current sensors, and also to run controllers such as relays.

2.2 *MQTT Protocol*

MQTT Protocol stands for Message Queuing Telemetry Transport is one of the protocol models that works with TCP/IP with the Publish-Subscribe model [2,3,4,5,6]. Designed for low bandwidth, high latency networks, and for IoT devices with limited resources. MQTT security uses SSL/TLS for encryption. The uses of MQTT include Home Automation, telemetry, and various IoT applications such as Smart Home, Smart Farming, industrial, education, etc. [7,8,9,10,11,12]. MQTT broker for example is Mosquitto and also various libraries that support it. For example, if we use Python, we need a library to be able to connect to MQTT, namely by using Paho-MQTT [17,18,19,20,21].

2.3 *Calculations on the electrical system of the Shredder Capacitor Bank: An Approach*

In this section, we will perform calculations of the system built by the chopping machine and also the electrical system built from the capacitor bank provided for the running process of this machine. The first is the calculation of the required Reactive Power (Q_C). with formulas such as Equation 1. Formula 1 involves P, PF_1, and PF_2.

Where P is the active power in the chopper motor in kW. While PF_1 is the initial power factor, for example, 0.6-0.8 for an uncorrected induction motor. While PF_2 is the target Power Factor usually 0.95.

$$Q_c = P\left(\sqrt{\frac{1}{PF_1^2}} - 1 - \sqrt{\frac{1}{PF_2^2}} - 1\right) \tag{1}$$

Moreover, the active motor power (P) is written as in equation 2. Where v is the source voltage (volts), I is the amperage (Amperes), and Cos ϕ is the motor starting power factor.

$$P = VxIx\cos\phi \tag{2}$$

To increase the power factor from Cos $\phi 1$ to Cos $\phi 2$, it is necessary to have the required reactive power Qc (VAR), as shown in equation 3. Where $\phi 1$ is the initial power factor angle, and $\phi 2$ is the target power factor angle.

$$\text{Qc} = P\ x\ (\tan\phi_1 - \phi_2) \tag{3}$$

Moreover, capacitor capacitance (C) is reactive power Qc (VAR) divided by 2phi x f x V^2. Where f is the Source Frequency (Hz) as shown in equation 4.

$$C = \frac{Q_c}{2\pi f V^2} \tag{4}$$

The motor starting capacitor can be estimated with equation 5. The starting capacitor value is generally 70-100 μF per horsepower (HP).

$$C_{start} \approx \frac{1000\ x\ P}{2\pi f V^2 x(1 - cos\phi_{start})} \tag{5}$$

The energy stored in the capacitor bank is according to equation 6. Where the Energy value is (Joules) and Ctotal is the total bank capacitance (Farads).

$$E = \frac{1}{2}C_{total}V^2 \tag{6}$$

The Parallel configuration of the Capacitor Bank can be written according to equation 7. If the capacitors are installed in parallel, the total capacitance increases. Conversely, if the capacitors are installed in a series configuration, the total capacitance decreases, as shown in Equation 8.

$$C_{total} = C_1 + C_2 + \cdots + C_n \tag{7}$$

$$\frac{1}{C_{total}} = \frac{1}{C_1} + \frac{1}{C_2} + \cdots \ldots + \frac{1}{C_n} \tag{8}$$

3. Method

3.1 *Methodology Flowchart*

The method of this research is shown in Figure 3, where the development of the IoT system starts from the selection of sensors & hardware, firmware development, and cloud/server integration system. After the IoT system development process, optimization and development of the algorithm system are carried out. Which consists of Power Quality analysis, Automatic Control, and Failure Prediction. Furthermore, three things need to be considered, in the method built, the things that are done are ensuring that the IoT-based Real-time monitoring system works well, optimizing the performance of the capacitor bank, and managing the waste shredding machine application. The method developed is Research and Development (R&D) with an experimental approach. So there is a trial and error process, requiring system testing and validation, real-time measurement and testing, there are performance parameters that must be optimized which involves the implementation of technology on existing systems.

The stages required are needs analysis including the process of identifying capacitor bank parameters that need to be monitored, specifications of waste chopping machines, the needs of the Internet of Things system, system design including IoT architecture design, sensor selection and components, monitoring system design, and interface design. The implementation is the creation of prototypes, integration of sensors with capacitor banks, development of monitoring software, and system calibration. The testing and optimization process consists of testing monitoring accuracy, real-time data validation, optimization of capacitor bank performance, and efficiency analysis. The evaluation process consists of analyzing system performance, measuring the success rate, and documenting the test results.

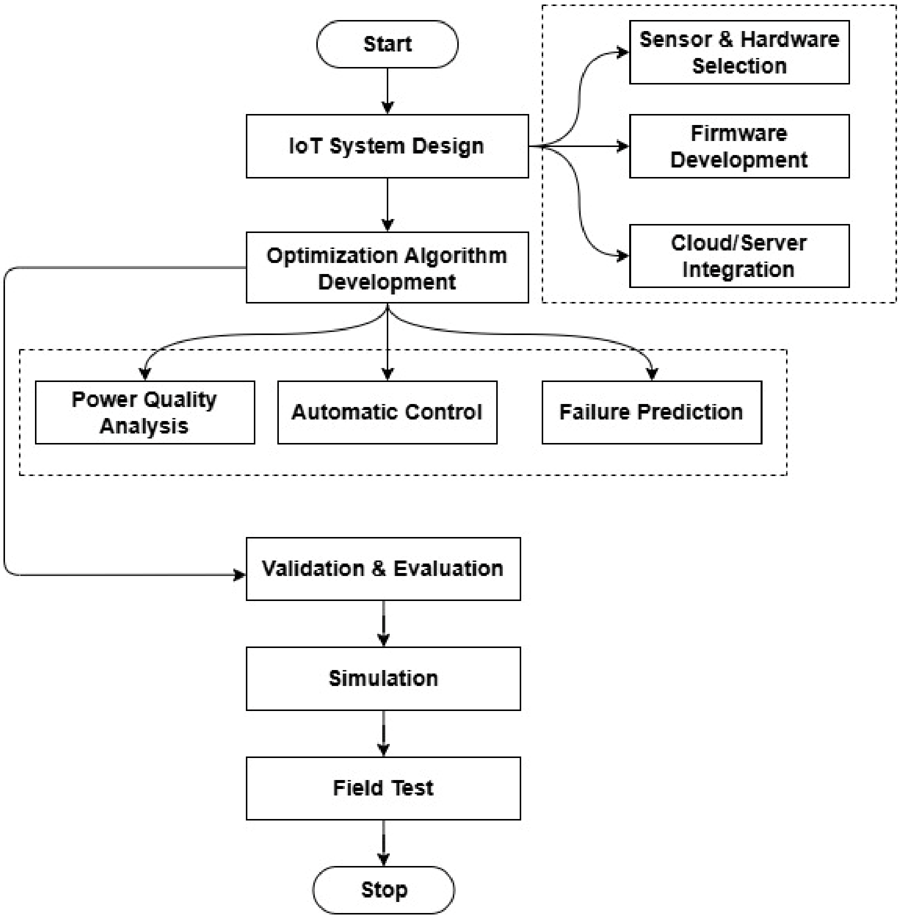

Figure 3. Methodology Flowchart

4. Result and Analyzes

4.1 Monitoring the Voltage (V) and Current (A) of the Waste Shredding Machine on Capacitor Banks

In the results and analysis section, an analysis will be made to see the surge and also the conditions at the time of starting the waste chopper machine indicated by the induction motor, so that it can be seen how the capacitor bank works in handling over voltages and also see in detail the condition of the electric current. Moreover, from the simulation, the parameters will be determined first, this is important to show the output of voltage and current on the waste shredder using and without a capacitor bank. We will set the frequency at 50 Hz and the voltage source is 220 Volts AC.

After initializing the Voltage and Frequency, proceed with initializing the Motor, namely Motor Inductance of 0.5 Henry, and Motor Resistance of 10 ohms, while the power Factor without a capacitor is 0.7. then continued with the calculation of impedance, calculation of required capacitors, calculation of Voltage and Current, calculation using capacitor banks, Starting Current Simulation, and accompanied by an output display in the form of voltage and current graphs. The complete pseudocode can be seen as follows:

```
BEGIN PROGRAM BankCapacitorSimulation

// Simulation Parameter Initialization
SET timeArray = CREATE_LINEAR_ARRAY(start=0, end=0.1, points=1000)
SET frequency = 50 Hz
SET sourceVoltage = 220 V

// Motor parameter
SET motorInductance = 0.5 H
SET motorResistance = 10 Ω
SET powerFactorWithoutCapacitor = 0.7

// Motor Impedance Calculation
SET motorImpedance = SQRT(motorResistance^2 + (2π × frequency ×
motorInductance)^2)
SET motorPhaseAngle = ARCCOS(powerFactorWithoutCapacitor)

// Calculation of Required Capacitors
SET motorReactivePower = (sourceVoltage^2 × SIN(motorPhaseAngle)) /
motorImpedance
SET requiredCapacitance = motorReactivePower / (2π × frequency ×
sourceVoltage^2)

// Voltage and Current Calculations
FOR each time IN timeArray DO
    // Tanpa Bank Kapasitor
    SET voltage[time] = sourceVoltage × SIN(2π × frequency × time)
    SET motorCurrent[time] = (sourceVoltage/motorImpedance) ×
                            SIN(2π × frequency × time - motorPhaseAngle)

// With Capacitor Banks
    SET capacitorCurrent[time] = 2π × frequency × requiredCapacitance ×
sourceVoltage ×
                                SIN(2π × frequency × time + π/2)
    SET totalCurrent[time] = motorCurrent[time] + capacitorCurrent[time]
END FOR

// Starting Current Simulation
SET startupDuration = 20% of timeArray length
SET startingMultiplier = 6
FOR time = 0 TO startupDuration DO
    SET motorCurrent[time] = motorCurrent[time] × startingMultiplier
    SET totalCurrent[time] = motorCurrent[time] + capacitorCurrent[time]
END FOR

// Data Visualization
CREATE Figure with 2 subplots

// Plot 1: Voltage
PLOT timeArray vs voltage
SET title = "System Voltage and Currents with/without Capacitor Bank"
SET ylabel = "Voltage (V)"
ADD grid
```

```
// Plot 2: Current
PLOT timeArray vs motorCurrent
PLOT timeArray vs totalCurrent
SET xlabel = "Time (ms)"
SET ylabel = "Current (A)"
ADD grid
ADD legend

// Calculate and Display Power Factor
SET powerFactorWithCapacitor = CALCULATE_POWER_FACTOR(totalCurrent, voltage)
DISPLAY "Power Factor without Capacitor: " + powerFactorWithoutCapacitor
DISPLAY "Power Factor with Capacitor: " + powerFactorWithCapacitor

// Show System Parameters
DISPLAY "System Parameters:"
DISPLAY "Source Voltage: " + sourceVoltage + " V"
DISPLAY "Motor Inductance: " + motorInductance + " H"
DISPLAY "Motor Resistance: " + motorResistance + " Ω"
DISPLAY "Calculated Capacitance: " + requiredCapacitance + " µF"

SHOW plots

END PROGRAM
```

Moreover, The output of the simulation can be seen in Figure 4, Voltage (V) can be seen oscillating as in Figure 4, given the value of Power Factor without a capacitor is 0.7, while with a capacitor is 1. while the picture below is a comparison between Current without Capasitor Bank with red line, and with Capasitor Bank with a green line. The surge is higher without using a capacitor bank, but by using a capacitor bank, the surge voltage is lower.

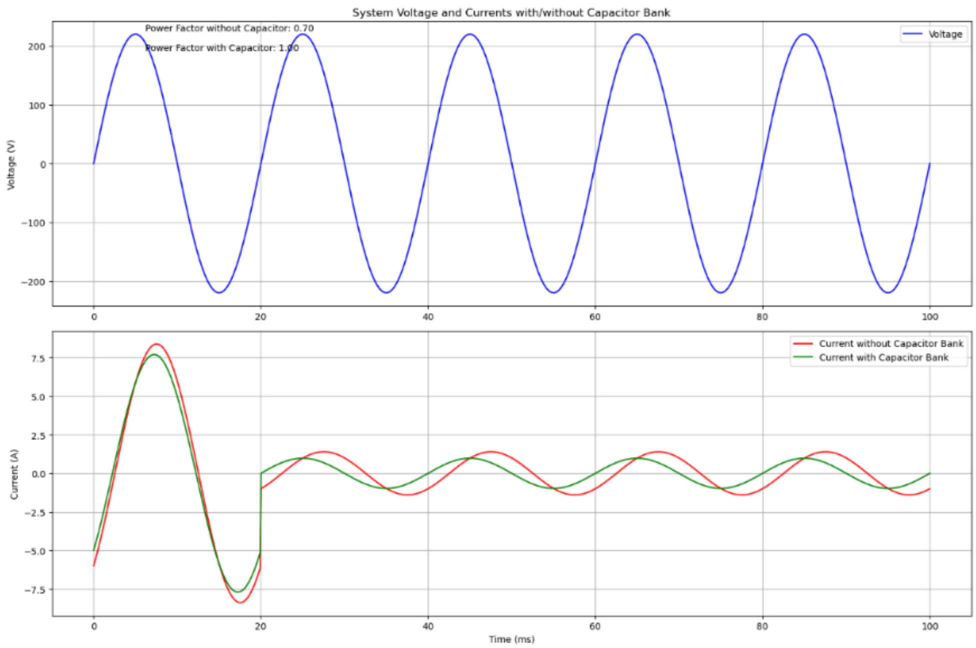

Figure 4. System Voltage and Currents with/without Capacitor Bank

4.2 Real measurement results of electric motors

Moreover, In this research, the specifications of the waste shredder used are shown in Figure 5. While the type of capacitor is shown in Figure 6. Next, we input all the parameters of the electric motor and run it in Python Code.

Figure 5. Specifications of waste shredding machine

Figure 6. Type of capacitor used

Moreover, from Figure 5, it is obtained that the motor specifications are Bologna Electric Motors with complete specifications as follows:

- Type: BLY-112
- Power: 1.5 KW / 2 HP
- Voltage: 220V
- Current: 9.44A
- RPM: 2800
- Frequency: 50 Hz
- Poles: 2
- Power Factor (cos φ): 0.85 (typical for this type)

Moreover, Figures 7 and 8 outline the characteristics of the Start-up current, speed, Torque Characteristics, and Efficiency vs load of the BLY-112 electric motor. From the measurement results or Motor Analysis results, the following parameter values are obtained:

- Apparent Power (S) = 2076.80 VA
- Active Power(P) = 1500.00 W
- Reactive Power(Q) = 1094.02 VAR
- Nominal Torque= 5.12 N.m

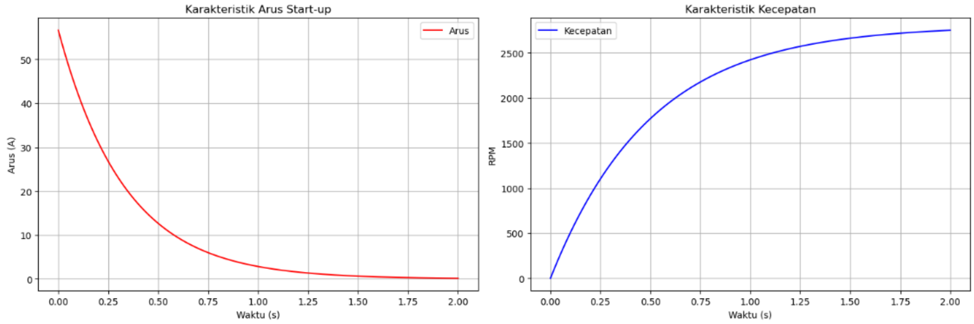

Figure 7. Characteristics of Start-up Current (A) and speed (RPM)

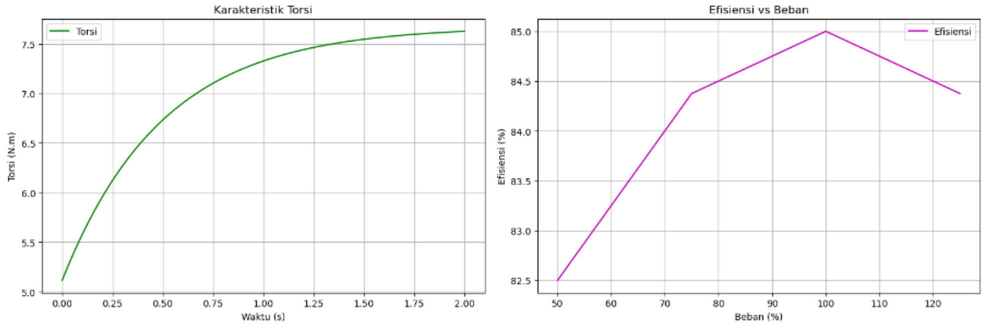

Figure 8. Torque Characteristics (N.m) and Efficiency vs. Load Comparison (%)

Moreover, to simulate the Internet of Things (IoT), a component called paho-mqtt is required. As shown in Figure 9. Several compatible Python components or libraries for the process of converting field data into real-time data are needed to increase the effectiveness, speed, and accuracy of obtaining data in real-time.

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.26100.2894]
(c) Microsoft Corporation. All rights reserved.

C:\Users\BRIN>pip install paho-mqtt
Defaulting to user installation because normal site-packages is not writeable
Collecting paho-mqtt
  Downloading paho_mqtt-2.1.0-py3-none-any.whl.metadata (23 kB)
Downloading paho_mqtt-2.1.0-py3-none-any.whl (67 kB)
Installing collected packages: paho-mqtt
Successfully installed paho-mqtt-2.1.0

C:\Users\BRIN>
```

Figure 9. paho-mqtt installation

Then it needs initial configuration as following this Pseudocode:

1. **Initial Configuration**
   MQTT_BROKER ← "broker.hivemq.com" // Broker MQTT publik
   MQTT_TOPIC ← "mesin_pencacah/bank_kapasitor" // Topik MQTT
   MQTT_PORT ← 1883 // Port MQTT standar

2. **Function to Calculate Stored Energy**
   FUNGSI calculate_energy(capacitance, voltage)
   energy ← 0.5 * capacitance * (voltage ^ 2)
   KEMBALIKAN energy
   AKHIR FUNGSI

3. Data Publication Function to MQTT

```
FUNGSI publish_data(client, voltage, current, energy, status)
    payload ← {
        "tegangan": voltage,
        "arus": current,
        "energi_tersimpan": energy,
        "status_mesin": status
    }
    client.publish(MQTT_TOPIC, payload)   // Kirim data ke broker MQTT
    CETAK("Data terkirim:", payload)
AKHIR FUNGSI
```

4. Main Functions of Simulation

```
FUNGSI simulate_bank_capacitor()
    // Inisialisasi MQTT Client
    client ← mqtt.Client()
    client.connect(MQTT_BROKER, MQTT_PORT, 60)
    CETAK("Terhubung ke broker MQTT:", MQTT_BROKER)

    // Parameter Awal
    capacitance ← 0.413   // Kapasitansi dalam Farad
    voltage ← 220         // Tegangan awal dalam Volt
    current ← 0           // Arus awal dalam Ampere
    status ← "OFF"        // Status mesin awal

    COBA:
        SELAMA BENAR:
            // Simulasi Perubahan Data
            voltage ← voltage + RANDOM(-5, 5)   // Fluktuasi tegangan
            voltage ← BATAS(voltage, 0, 240)    // Batasi antara 0-240V
            current ← RANDOM(0, 10)             // Simulasi arus
            energy ← calculate_energy(capacitance, voltage)   // Hitung energi

            // Simulasi Status Mesin
            JIKA energy > 5000 MAKA
                status ← "ON"
            LAINNYA
                status ← "OFF"
            AKHIR JIKA

            // Publikasikan Data ke MQTT
            publish_data(client, ROUND(voltage, 2), ROUND(current, 2), ROUND(energy,
2), status)

            // Delay untuk simulasi
            TUNGGU(5 detik)

    TANGKAP KeyboardInterrupt:
        CETAK("Simulasi dihentikan.")
        client.disconnect()
AKHIR FUNGSI
```

5. Run Simulation

```
JIKA PROGRAM UTAMA MAKA
    simulate_bank_capacitor()
AKHIR JIKA
```

Figure 10. Data output in MQTT Broker

5. Conclusions

The process of monitoring the waste chopper in Cikawao Village, Bandung is expected to be able to provide feedback in dealing with problems that occur, namely unstable voltage and current and the use of capacitor banks and monitoring effectively to produce a valid and close loop real-time monitoring system that provides feedback in the form of a control system that can stop the engine and turn it back on based on the conditions it receives. The process of monitoring the waste shredding machine needs to be changed towards the Internet of Things (IoT) to make real-time data that can be received by users or checkers quickly. Even overshoot or excess voltage or current and vice versa can be resolved quickly. In this article, simulation and theoretical calculations are carried out so that the data collection process and running tools can be carried out with comprehensive approaches in terms of equations, simulations, and also practical approaches towards MQTT Brokers to be able to build real IoT applications.

Acknowledgments: Thank you to all academic staff, students, and lecturers at the Department of Electrical Engineering, Langlangbuana University, Bandung, hopefully, this research can continue to be developed. In particular, the MQTT-based IoT monitoring system and its development.

Author contributions: The authors were responsible for building Conceptualization, Methodology, analysis, investigation, data curation, writing—original draft preparation, writing—review and editing, visualization, supervision of project administration, funding acquisition, and have read and agreed to the published version of the manuscript.

Funding: The study was conducted without any financial support from external sources.

Availability of data and Materials: All data are available from the authors.

Conflicts of Interest: The authors declare no conflict of interest.

Additional Information: No Additional Information from the authors.

References

1. M. E. Borisova, A. M. Kamalov and Y. K. Osina, "Absorption Phenomena in Capacitors Based on PPS Films," 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Saint Petersburg and Moscow, Russia, 2019, pp. 84-86, doi: 10.1109/EIConRus.2019.8657265.
2. W. Wang, Y. Zhao, Y. Liu, G. Liu, F. Zheng and C. Sun, "MQTT Protocol and Implementation of Equipment Management System for Industrial Internet of Things," 2024 43rd Chinese Control Conference (CCC), Kunming, China, 2024, pp. 6139-6144, doi: 10.23919/CCC63176.2024.10662474.

3. R. Doshi, S. Inamdar, T. Karmarkar and M. Wakode, "Distributed MQTT Broker: A Load-Balanced Redis-Based Architecture," 2024 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 2024, pp. 1-6, doi: 10.1109/ESCI59607.2024.10497427.
4. M. Nast, M. Rethfeldt, F. Golatowski and C. Haubelt, "Improving the Real-Time Capability of MQTT for Sensor Networks (MQTT-SN) Using PREEMPT_RT," 2024 IEEE 20th International Conference on Factory Communication Systems (WFCS), Toulouse, France, 2024, pp. 1-8, doi: 10.1109/WFCS60972.2024.10540812.
5. S. U. A. Laghari, W. Li, S. Manickam, P. Nanda, A. K. Al-Ani and S. Karuppayah, "Securing MQTT Ecosystem: Exploring Vulnerabilities, Mitigations, and Future Trajectories," in IEEE Access, vol. 12, pp. 139273-139289, 2024, doi: 10.1109/ACCESS.2024.3412030.
6. K. Morise, T. Toyohara and H. Nishi, "Proposal of Differential Privacy Anonymization for IoT Applications Using MQTT Broker," 2024 IEEE 21st Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 2024, pp. 634-635, doi: 10.1109/CCNC51664.2024.10454877.
7. B. H. Variant Wahono and R. M. Ijtihadie, "Caching Cluster of Distributed MQTT Broker on Containerized Architecture," 2024 International Conference on Electrical Engineering and Computer Science (ICECOS), Palembang, Indonesia, 2024, pp. 349-353, doi: 10.1109/ICECOS63900.2024.10791203.
8. W. Alsabbagh, S. Amogbonjaye, C. Kim and P. Langendörfer, "Pirates of the MQTT: Raiding IIoT Systems with a Rogue Client," 2024 8th Cyber Security in Networking Conference (CSNet), Paris, France, 2024, pp. 248-253, doi: 10.1109/CSNet64211.2024.10851733.
9. K. Kusumi and H. Koide, "MQTT-MTD: Integrating Moving Target Defense into MQTT Protocol as an Alternative to TLS," 2024 7th International Conference on Advanced Communication Technologies and Networking (CommNet), Rabat, Morocco, 2024, pp. 1-8, doi: 10.1109/CommNet63022.2024.10793300.
10. A. Prayudi, M. F. Arief, M. A. Zainuddin, S. Sukaridhoto and H. Rante, "Wireless Edge Controller for Robot Education Using MQTT on Dobot Magician Lite," 2024 IEEE International Symposium on Consumer Technology (ISCT), Kuta, Bali, Indonesia, 2024, pp. 517-521, doi: 10.1109/ISCT62336.2024.10791077.
11. P. D. P. Adi, A. Kitagawa and J. Akita, "Finger Robotic control use M5Stack board and MQTT Protocol based," 2020 7th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), Semarang, Indonesia, 2020, pp. 1-6, doi: 10.1109/ICITACEE50144.2020.9239170.
12. P. D. P. Adi and A. Kitagawa, "A Review of the Blockly Programming on M5Stack Board and MQTT Based for Programming Education," 2019 IEEE 11th International Conference on Engineering Education (ICEED), Kanazawa, Japan, 2019, pp. 102-107, doi: 10.1109/ICEED47294.2019.8994922.
13. B. M. Agostinho, S. Chessa, R. Perego, M. A. Ribeiro Dantas and A. Sandro Roschildt Pinto, "MQTT-Chain: An MQTT approach using blockchain and smart contracts to achieve a new Quality of Service level," 2024 11th International Conference on Future Internet of Things and Cloud (FiCloud), Vienna, Austria, 2024, pp. 316-323, doi: 10.1109/FiCloud62933.2024.00056.
14. J. Kaur, J. Singh and K. Kaur, "A Review on Machine Learning Based Techniques for MQTT Based Attacks for the IoT Systems," 2024 International Conference on Cybernation and Computation (CYBERCOM), Dehradun, India, 2024, pp. 461-465, doi: 10.1109/CYBERCOM63683.2024.10803264.
15. A. Li, "MQTT Subscription Tree Optimization and Performance Analysis," 2024 IEEE 7th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 2024, pp. 1613-1617, doi: 10.1109/IAEAC59436.2024.10503937.
16. Q. Wu, T. Zhang and Z. Yang, "A Secure Communication Protocol Based on Blockchain and MQTT for IOT," 2024 IEEE 7th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 2024, pp. 1419-1423, doi: 10.1109/IAEAC59436.2024.10503637.
17. S. Banerjee, P. Kumari and T. Maity, "Development of MQTT Protocol-Based Sensor Data Subscription Using Raspberry-Pi as a Server Mode for IIoT Application," 2024 IEEE International Conference on Smart Power Control and Renewable Energy (ICSPCRE), Rourkela, India, 2024, pp. 1-5, doi: 10.1109/ICSPCRE62303.2024.10675281.
18. A. Miu, L. Ruse, R. Deaconescu and D. Tudose, "Integrating TLS/SSL with MQTT in NuttX Operating System," 2024 23rd RoEduNet Conference: Networking in Education and Research (RoEduNet), Bucharest, Romania, 2024, pp. 1-6, doi: 10.1109/RoEduNet64292.2024.10722675.
19. A. Kumar Arigela, C. Banapuram and N. Venu, "Remote based Home Automation with MQTT: ESP32 Nodes and Node-RED on Raspberry Pi," 2024 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Kirtipur, Nepal, 2024, pp. 313-318, doi: 10.1109/I-SMAC61858.2024.10714655.
20. Rizky Rahmatullah, A., et.al. 2025. Multinode LoRa-MQTT of Design Architecture and Analyze Performance for Dual Protocol Network IoT. International Journal of Advanced Computer Science and Applications(IJACSA), 16(1), 2025. http://dx.doi.org/10.14569/IJACSA.2025.0160129
21. Rahmatullah, A., et.al. 2023. Analyze Transmission Data from a Multi-Node Patient's Respiratory FMCW Radar to the Internet of Things. International Journal of Advanced Computer Science and Applications(IJACSA), 14(5), 2023. http://dx.doi.org/10.14569/IJACSA.2023.0140521