International Journal of Application on Sciences, Technology and Engineering
(IJASTE)
Volume 1, Issue 1, 2023.ISSN:2987-2499

# Monitoring for Temperature of Three Phase Induction Motor at Chemical Industry Using Internet of Things

Z. Zaini a) and Lidwina Natasha Chendra b)

Electrical Engineering, Andalas University, Padang, West Sumatera, Indonesia
b) Corresponding author: lidwinanatashachendra@gmail.com
a) Electronic mail: zaini@eng.unand.ac.id

*Submitted: November-December 2022, Revised: January 2023, Accepted: February 23, 2023*

**Abstract.** Induction motor is the most motor applicable in the world. especially three phase induction motor. Furthermore 95% of induction motor are used as propulsion in industry. However, maintenance costs are large enough more than 50% of operating costs of the industry or the factory. To reduce these costs is to detect early faults in induction motors. One way to detect initial faults is to measure and analyze the temperature of the induction motor. In this research, the authors design temperature monitoring of a three-phase induction motor using the internet of things to make it easier for users to monitor the temperature of an existing three-phase induction motor and to schedule maintenance if temperature reach the operating limit. The field system elements consist of SoC ESP32 to collect data from 2D temperature, to drive lead screw and to send data to cloud. The SoC ESP 32 is connected to the AMG8833 temperature sensor array which collect the frame of temperature of a three-phase induction motor in the form of an 8x8 array. In addition, the SoC ESP32 is also connected to the motor driver L293D to regulate the movement of geared DC motor. Geared DC motor is used to drive the lead screw which already has a mount for placing the AMG8833. From the test on 18.5 kW, 360V induction motor installed in chemical industry, it is obtained the sensor can measure temperature accurately and it travel smoothly above frame. In the storage using MongoDB database, the data transmission fulfilled the QoS requirement.

## 1. INTRODUCTION

Induction motor is the most widely used type of motor in the world. Specifically, the induction motor used is a three-phase induction motor. Induction motors are used for industrial purposes such as carpentry machines, blowers, elevators, compressors, mining industry, automotive industry, chemical industry, and others [1].

As many as 95% of induction motors are used as propulsion in industry. However, the costs used for maintenance are large enough to cost more than 50% of the operating costs of the industry or the factory itself [2]. This maintenance is important because induction motors are often required to work without shopping. With regular maintenance, fatal damage can be avoided and can prevent financial losses [1].

Detecting several faults that occur in induction motors is one way to prevent financial losses. For example, by detecting vibrations in an induction motor, you can find out whether it is a machine error or not because the machine works mechanically. In addition, by analysing the electric current can find out whether the induction motor has an electrical error. However, based on [3] most of the induction motor faults are in the bearing (41%) and winding (31%) [3].

Broadly speaking, there are two causes of damage to induction motors, namely electrical faults and mechanical faults. In general, the cause of damage is electrical faults such as stator winding, rotor winding, and others [4]. 35% to 40% failure of motor winding insulation is the cause of induction motor failure. Assuming that every time there is an increase in temperature of 100C from the stator winding, the service life of the induction motor can be reduced by 50% [5].

Faults that occur in bearings and windings can be classified as fatal errors because they can cause an increase in the temperature of the induction motor and if left unchecked can results in a short circuit. However, the temperature rise in the induction motor can also be caused by a failure in the cooling system used [3].

In the induction motor error that causes an increase in the temperature of the induction motor can be detected using a temperature sensor. The temperature sensors used in general are the resistance temperature detector (RTD) and the thermal array sensor in which there in an infrared sensor [1-3].

Based on the number of uses of induction motors [2], it is necessary to make it easier to monitor temperature. The author chooses to use the Internet of Things (IoT) so that users can monitor remotely. This is related to the industrial revolution 4.0 that the industrial internet is enabling technology for efficiency improvement in digital era [6].

## 2. BACKGROUND

### 2.1 Induction Motor Fault

Machines that have an amortisseur winding are called induction machines. The machine is called an induction machine due no external excitation occurred the magnetic field is generated in the rotor according to voltage induced by mismatch between rotating field speed with rotor speed. The thing that distinguishes induction motors from induction machines is that induction machines do not require DC current to run the machine [7].

Induction motors have high resistance to stress factors caused by unpredictable things. However, changing situ-ations and internal or external factors can affect not only the motor construction. Stress factors can interfere with the smooth and reliable operation of an induction motor. If it is not diagnosed properly, the induction motor will be damaged [8].

In induction motors, the most common faults are electrical faults and mechanical faults. Errors are categorized as electrical if they are related to electric current. Meanwhile, the fault is categorized as mechanical if the main cause of the disturbance is mechanical. Induction motor fault detection can be done through voltage, current, torque, speed, magnetic flux, vibration, and temperature.

https://doi.org/10.24912/ijaste.v1.i1.140-162

## 2.2 Induction Motor Temperature

In general, thermal stress is caused by thermal aging or overload. Due to the increase in temperature of $10^0$C, the lifetime of the insulation is reduced by two times. A better type of insulation is recommended if the motor is used in a high pressure environment. Environmental stress is caused by external factors and pollution. For example, pollution that can cause improper thermal exchange between the motor and the surrounding environment. This causes an increase in the temperature of the motor which if left unchecked will increase the risk of failure of the electric coil or winding [8].

To prevent the winding insulation from being damaged by overheating, it is necessary to limit the winding temperature. However, this has an impact on the limited maximum power that can be continuously supplied by the engine. There is a standard temperature limit on engine insulation regulated by the National Electrical Manufactures Association (NEMA) which determines the insulation system based on the existing class. In general, there are three classes of NEMA insulation for motors, namely B, F, and H. Each class requires that the permissible winding temperature be higher than the previous temperature. For example, the increase in the temperature of the armature winding above the ambient temperature in a type of induction motor that operates continuously must be limited to $80^0$C for class B, $105^0$C for class F, and $125^0$C for class H [7]. In addition, according to NEMA, the highest value for the induction motor's ambient temperature is $40^0$C.

One of the existing over-temperature protection is using the ANSI code49RMS relay. The protection carried out by this relay is to give a disconnection order when there is an increase in temperature (E). Calculation according to the equivalent current measurement ($I_eq$) greater than the set point ($E_s$) [9]. The largest continuous current allowed can be seen in equation 1.

$$ \overset{p}{I} = I_b \quad \dot{E}_s \tag{1} $$

According to equation 1, $I_b$ is the base current and $E_s$ is the heat rise. Set points in the hot state, when the function is used to protect the motor, the set points are designed to detect the hot state used by the start function. The graph of the heat rise and cooling time constant can be seen in Figure 1 and the characteristics of the time constant can be seen in Table 1.

TABLE I. Time Constant Characteristic

:

| Time Constant | | Rate 1 | Rate 2 |
|---|---|---|---|
| Setting | T1 running (heat rise) | 1 mn to 600 mn | 1 mn to 600 mn |
| | T2 stopped (cooling) | 5 mn to 600 mn | 5 mn to 600 mn |

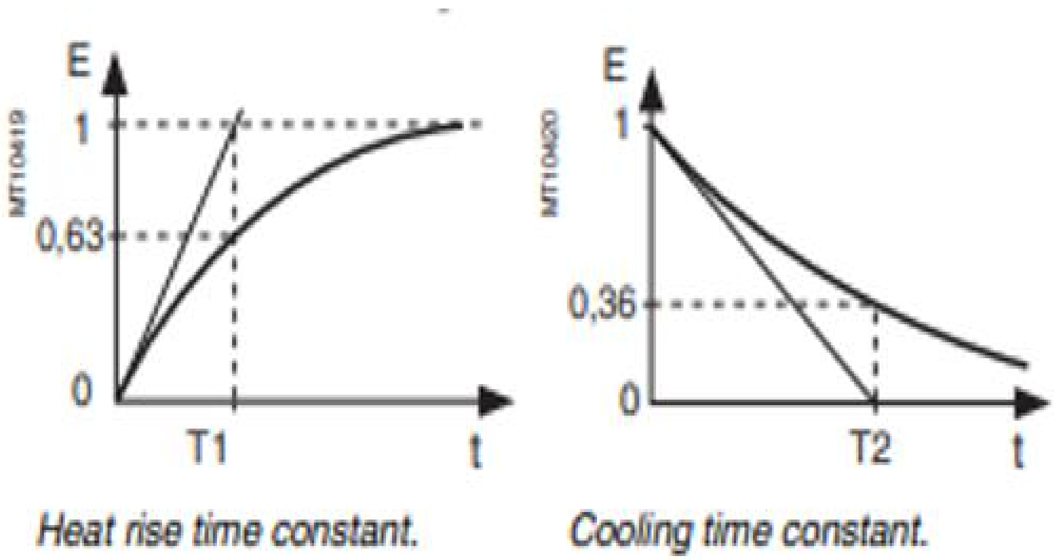


FIGURE 1. Graphics of Heat Rise and Cooling Time Constant

Based on Table 1, Rate 1 is the base current $I_b$ which is defined as a general Sepam parameter, used to calculate the temperature rise in the equipment, while rate 2 is the base current $I_b$-rate 2, the protection setting from a certain temperature overload, is used to calculate the temperature rise on equipment.

## 2.3 Components Used

There are several components used in compiling this research, such as thermal infrared array sensor (AMG8833), system on chip (SoC) ESP32, driver motor (L293D), geared DC motor (JGA25-371)

## Thermal Infrared Array Sensor (AMG8833)

The working principle of AMG8833 is to absorb heat generated by an object and convert to digital data they can be used by microprocessor into body temperature in 2D axis [10]. The results of the detection of AMG8833 can be seen in Figure 2 where AMG8833 is applied to detect temperature as a medical diagnosis [11].

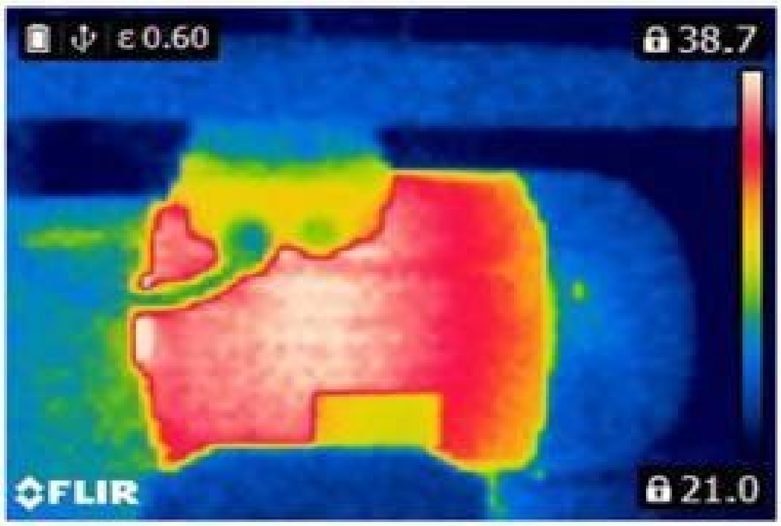


FIGURE 2. Detection Result from Thermal Array Sensor

This sensor has 64 arrays consisting of 8 × 8 pixels and works with an input voltage of 3.3 V.DC with a detected temperature range of 0$^0$C to 80$^0$C. The amplification factor of the AMG8833 is high performance type high gain.

The purpose of knowing this amplification factor is to make it easier to know the advanced specifications of the sensors used [10].

## SoC ESP32

The ESP32 is a SoC (System on Chip) microcontroller integrated with Wi-Fi 802.11 b/g/n, dual mode Bluetooth version 4.2 and various peripherals. ESP32 is a continuation of the ESP8266 chip with the implementation of two cores clocked with different versions up to 240 MHz. If this chip is compared to its predecessor, on this chip there is an increase in the number of GPIO pins from the original 17 to 36 with the number of PWM channels per 16 and is equipped with 4 MB of flash memory [11]. The specifications of this component can be seen in Table II [11].

TABLE II. Specification of ESP32

| Specification | description |
|---|---|
| SoC | ESP32 |
| Wi-Fi Protocol | 802.11 b/g/n |
| Input voltage | 3.0 3.6 V |
| Digital pin I/O | 36 |
| PWM pin | 4 |
| Communication | SPI, I2S, I2C, CAN, UART, Ethernet MAC, and IR |
| Flash memory | 4MB |
| Clock speed | 240 MHz |

## Driver Motor (L293D) and Geared DC Motor

The L293D is a high-voltage, four-channel monolithic integrated device designed to accept standard DTL or TTL logic levels and drive inductive loads (such as solenoid relays, DC, and stepping motors), and switching power transistors [12].In this reseaerch, L293D is used to control the performance of a geared DC motor (JGA25-371). Geared motor is a motor that uses a set of gears to convert the original high speed and low torque motor into a low speed high torque motor.

To simplify use as two bridges, each channel is equipped with a possible input. A separate input supply has logic that allows lower voltage operation and an internal clamp is included [12]. The specifications of the L293D can be seen in Table III.

TABLE III. Specification of L293D

:

| Specification | description |
| --- | --- |
| Supply voltage | 4.5 Volt to 36 Volt |
| Logic supply voltage | 36 Volt |
| Input voltage | 7 Volt |
| Enable voltage | 7 Volt |
| Pin total | 4 input pins, 4 output pins, 4 ground pins, 1 logic supply pin, 1 supply voltage pin, and 2 enable voltage pins |

## 2.4 Software

The software used in this research is Arduino IDE to manage the performance of the system and MongoDB as a database to store data that has been obtained from the device.

## Arduino IDE

Arduino provides a simple code structure called a sketch, written in C language and only consists of two functions, namely setup and loop. The setup function only runs once at the start when the system starts, while the loop function is executed repeatedly while the system is on, User can write as many functions as required for the connected software, but must call the function in the setup function or loop. For more details, it can be seen in Figure 2.8 where there is a depiction of a typical Arduino sketch structure, which can be compiled and uploaded by the Arduino IDE to the intended processor by simply clicking a simple button [13].

## Database and MongoDB

Database is a structured collection of records or data stored in a computer system and arranged in such a way that it can be searched quickly and information can be retrieved quickly [14]. The database can be created using MongoDB software with the programming language that can be used is BSON [15].

MongoDB is a database management system designed to connect to the internet and based on web applications with a maximum storage capacity of 16 megabytes [16]. MongoDB can be classified as a document-based NoSQL database. The data model and persistence strategy is built for read and write throughput and the ability to scale easily using automatic failover [15]. The MongoDB document data model makes it easy to build as it has built-in support for unstructured data and does not require expensive and time-consuming migrations as application requirements change [15].

MongoDB documents are encoded in a JavaScript Object Natation (JSON)-like format called BSON. BSON lends itself well to modern object-oriented programming methodologies and is lightweight, fast, and traversable. MongoDB uses BSON as the network transfer format for documents. BSON looks like BLOB at first, but there is an important difference that the MongoDB database understands BSON internals. This means that MongoDB can span within BSON objects, even nested objects, using dot notation. This allows MongoDB to build indexes and match objects with query expressions on nested and top-level BSON keys.

MongoDB is a cross-platform, document provisioning oriented database, high performance, high availability, and easy scalability. MongoDB works with collections and documents concepts. A database is a physical container for collections. Each database has its own set of documents in the document system. One MongoDB server usually has multiple databases.

## 2.5 Quality of Service (QoS)

In modern society, complex software systems have become highly distributed, component-based, and service-oriented. These service-oriented systems consist of a large number of software components that can be found at run-time and run on many unknown and heterogeneous hardware and network platforms. Distributed software components are usually implemented as web services. Web services are software components designed to support machine-to-machine interactions that can be operated over a network, which is the main technique for building service-oriented systems [17].

To ensure that service-oriented systems remain reliable, efficient, and effective, service quality assurance techniques are needed. Quality of Service (QoS) is one of the most important aspects to determine software quality. In modern service-oriented systems, QoS includes a number of parameters such as throughput, latency, packet loss [18].

### a. Throughput

Throughput is the actual ability to determine the time it takes for a data packet from the time it is sent by the transmitter to the time it is received by the receiver from a network [18]. In addition, throughput can also be referred to as the data transfer rate [19]. The throughput calculation equation can be seen in equation 2 and the throughput category itself can be seen in Table IV [19].

$$Throughput(\%) = \frac{\text{Amount of data received(bit)}}{\text{Data delivery time(second)}} \quad 100\% \qquad (2)$$

TABLE IV. Throughput Standardisation

| Throughput category | Throughput | Index |
|---|---|---|
| Very good | 100% | 4 |
| Good | 75% | 3 |
| Moderate | 50% | 2 |
| Poor | <25% | 1 |

b. Latency

Latency is the time it takes for data to arrive from the origin to the destination. Latency can be affected by physical media distance, congestion, or long processing times [19].
The latency calculation equation can be seen in equation 3 and the latency category can be seen in Table V.

$$\text{Latency(ms)} = \frac{\text{thelengtho f timethedatapacketreachestherecipient(ms)}}{\text{totaldatapacketsreceived}} \quad (3)$$

TABLE V. Latency Standardisation

| Latency category | Latency | Index |
|---|---|---|
| Very good | <150 ms | 4 |
| Good | 150 to 300 ms | 3 |
| Moderate | 300 to 450 ms | 2 |
| Poor | >450 ms | 1 |

c. Packet loss

Packet loss is the amount of data lost during the transmission process to the destination [18]. Packet loss can also be interpreted as the number of data packets that fail to reach the destination where the data packets were sent [19].
The equation for calculating packet loss can be seen in equation 4 and for standardization in the packet loss category, it can be seen in table VI.

$$\text{Packetloss}(\%) = \frac{\text{Data packetssent datapacketsreceived} \quad 100\%}{\text{Data packets sent}} \tag{4}$$

TABLE VI. Packet Loss Standardisation

| Packet loss category | Packet loss | Index |
|---|---|---|
| Very good | 0% | 4 |
| Good | 3% | 3 |
| Moderate | 15% | 2 |
| Poor | 25% | 1 |

The QoS value can also be affected by the density of network traffic.

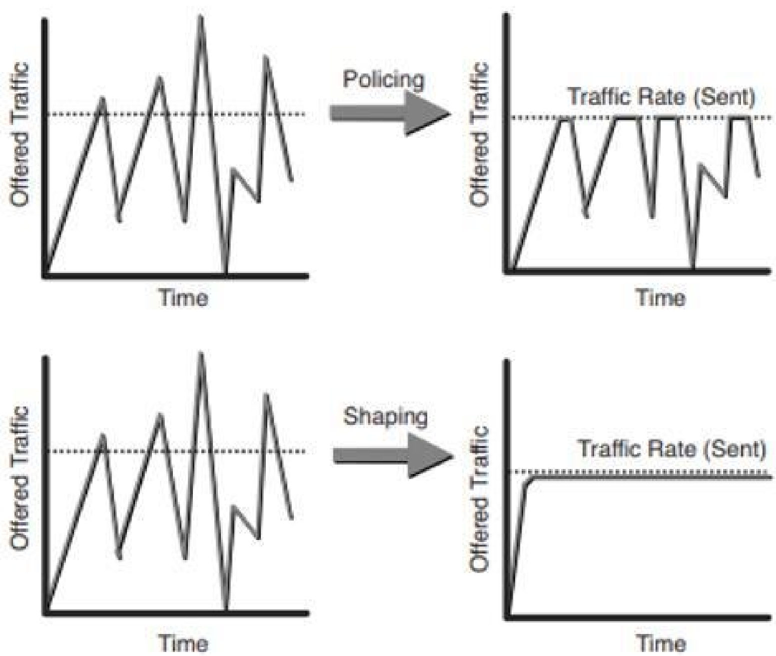


FIGURE 3. Comparison between Uncontrolled Network and Controlled Network

So traffic formation is needed to control the network so that the performance obtained is optimal, has low latency, and also optimal bandwidth [20]. This comparison can be seen from the peak value between the network traffic that has not been controlled and the network traffic that has been controlled in Figure 3 [21].

Based on Figure 3, it can be seen that there are two methods for managing network traffic, namely by policing and shaping. Policing is carried out by policers by checking traffic violations against the configuration speed and taking actions such as reducing or re-creating excess traffic. In this action, only network traffic checks and action are taken [21].

# 3. METHODS

This research uses two approaches, namely a qualitative approach and a quantitative approach. A qualitative approach is used by analyzing the data obtained. A qualitative approach is used in testing the array temperature sensor (AMG8833) whether the sensor is working well or not. The temperature sensor is categorized as working well if the sensor's accuracy in detecting temperature has a difference in the range of $2{:}5^0$C when compared to a thermogenic. If the sensor has a difference of more than $2{:}5^0$C, the temperature sensor array measurement can be categorized as poor. This qualitative approach can also be used on IoT platforms where the reference can be seen in the Quality of Service (QoS) section. References to be considered here are latency and packet loss. The two references consist of 4 categories, namely very good, good, moderate, and bad.

A quantitative approach is used by collecting data from tests conducted in the field. A quantitative approach can be used to see the performance of the lead screw. In this approach, it can be seen from the speed where the sensor is placed whether the speed value changes or not. In addition, a quantitative approach can also be used to calculate the amount of Quality of Service (QoS), which consists of latency and packet loss.

## 3.1 System Design

Making a three-phase induction motor temperature monitoring system using the AMG8833 temperature sensor array, DC motor, L293D motor driver, and ESP32 SoC. The AMG8833 temperature sensor reads temperature data in the form of 8×8 pixels. The DC motor is connected to the L293D motor driver where this motor driver regulates the performance of the DC motor. The temperature sensor and motor driver are connected to the ESP32 SoC. The results of the AMG8833 sensor readings in the form of temperature data in the form of an array are sent to the server via a Wi-Fi connection. The server will process the data into a database so that the webserver can provide information to users that can be seen by users through the website that has been created. The block diagram design of the system can be seen in Figure 4.

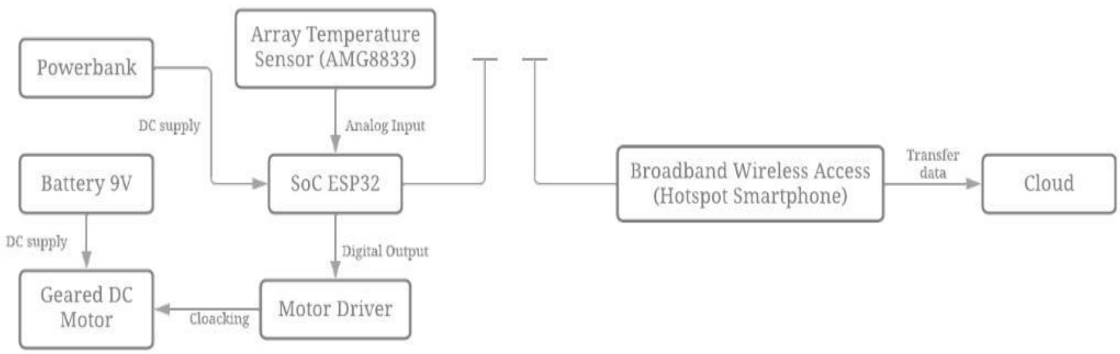


FIGURE 4. Tool Design

The workflow of the three-phase induction motor temperature monitoring system can be seen in Figure V, the AMG8833 temperature sensor array and the L293D motor driver connected to the ESP32 SoC will continue to work to obtain temperature data and regulate the motion of the DC motor. The ESP32 that has been connected to the server via the API database will send the data obtained from the sensor via the connected Wi-Fi, then the data will be processed and stored in the MongoDB database. The stored data is then displayed by the webserver on the server computer or other computers connected by the server by accessing the website that has been created.

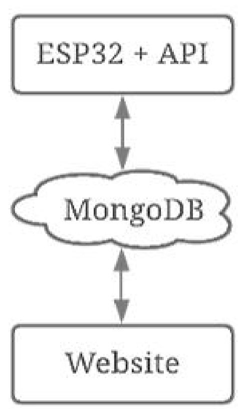


FIGURE 5. Three Phase Induction Motor Temperature Monitoring System Diagram

## 3.2 Software Design

It can be seen in Figure 6, the first step is to initialize the ESP32, temperature sensor, and motor driver. Furthermore, ESP32 is connected with Wi-Fi. If the ESP32 condition can be connected or can be categorized as "Yes", then the motor driver will drive the DC motor, then the temperature sensor array will detect the temperature of the three-phase induction motor and the reading will be sent to the webserver. If the condition of the ESP32 cannot be connected or can be categorized as "No", then the ESP32 will continue to loop until it can connect to Wi-Fi.

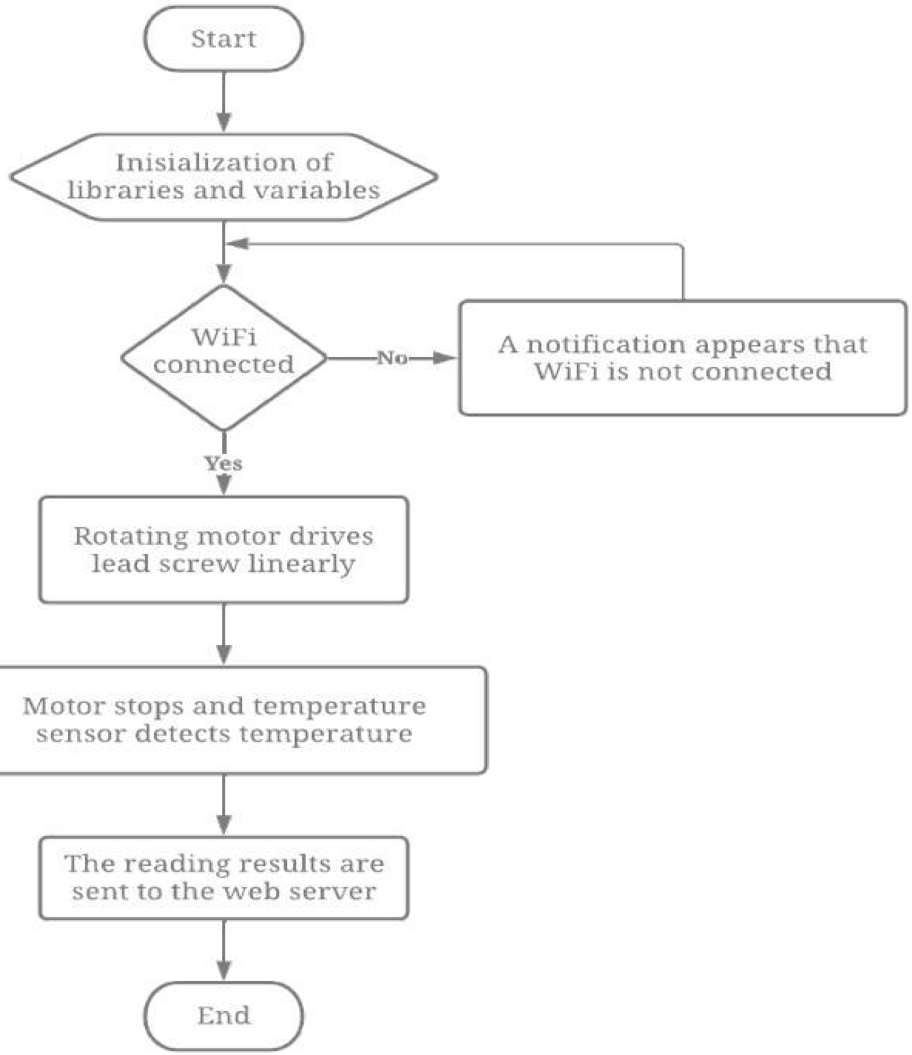


FIGURE 6. Program Flowchart

## 3.3 Hardware Design

The hardware design consists of several main components, namely ESP32, AMG8833 temperature sensor array, L293D motor driver, and DC motor whose diagram can be seen in Figure 7. An array temperature sensor is a temperature reading of the induction motor. Motor driver to regulate the motion of the DC motor. ESP32 is the SoC that regulates the performance of the motor driver, receives the temperature sensor readings, and sends the readings to the user via a Wi-Fi connection.

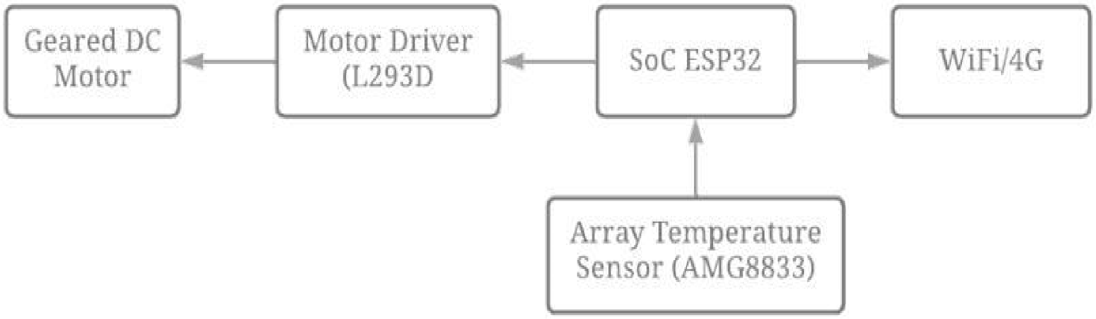


FIGURE 7. Hardware Design

# 4. RESULTS

## 4.1 System Design Results

The results of the system design can be seen in the diagram listed in Figure 8 where after the components are assembled as shown in Figure 8 they are connected to the lead screw which has been supported by an iron rod and an acrylic stand.

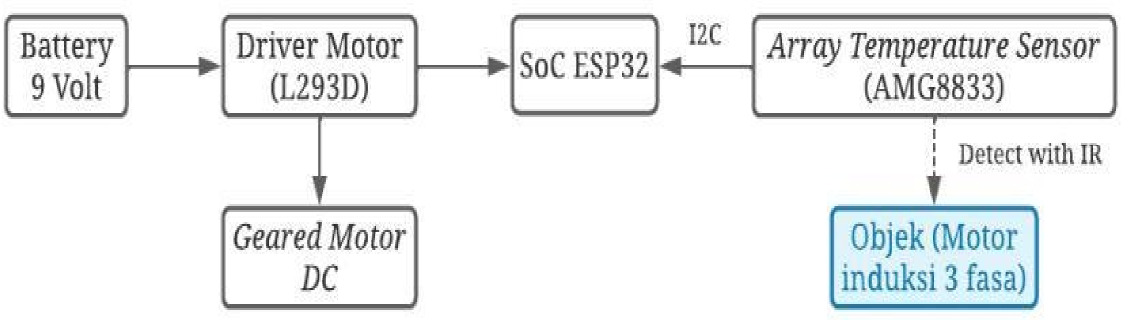


FIGURE 8. Data Acquisition System Design Result Diagram

This series of hardware will send data via WLAN to the API which then the API sends data to MongoDB. The data stored on MongoDB can be displayed by the website through an API which will access the database through a request to the web server. The display of the tool from the results of the system design can be seen in Figure 9 for the front view for the rear view.

## 4.2 Temperature Monitoring System Testing

The testing phase is carried out to determine whether the designed system can work well. System testing is divided into three parts, namely temperature sensor testing, lead screw testing, and network Quality of Service testing.

a. Array Temperature Sensor Testing

At this stage, the temperature monitoring system is tested by comparing the results obtained from the AMG8833 temperature sensor array with the results read by the infrared thermometer. The comparison can be seen in Figure 10. If seen in Figure 10 on the left, there is line 6, line 8, when compared to the reading results in Figure 10 on the right, there is a difference in line 6 of $0{:}9^{0}$C. Based on the datasheet from AMG8833, the difference between the AMG8833

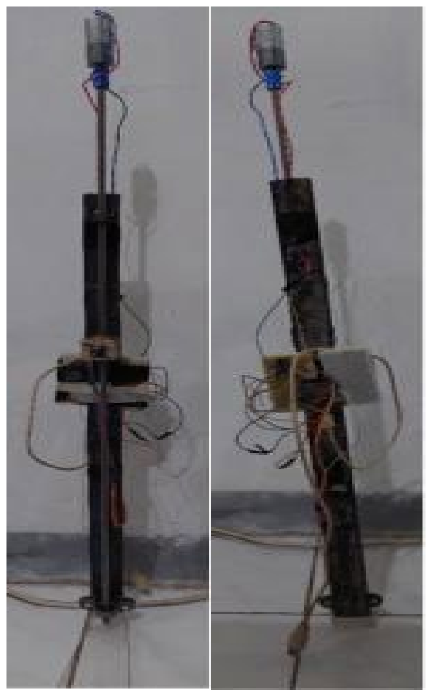

FIGURE 9. (from left to right) Front View, Rear View


FIGURE 10. (from left to right) AMG8833 Readings and Fluke 61 Readings at 42:60C

readings and the Fluke 61 readings, the AMG8833 readings can be categorized as accurate because they are in the sensor sensitivity range of 2:5$^0$C.

b. Lead screw Mechanical Test

The lead screw test is carried out to determine whether the lead screw used can work properly or not. The indicator of this lead screw test is the constant speed generated from the place where the temperature sensor is placed. If the resulting speed is constant then the lead screw used is working well.

The lead screw test was carried out under sunny weather conditions for about 1 hour. The lead screw is driven by a geared motor which is connected to the L293D motor driver. The condition of the geared motor installation connected to the motor driver can be seen in Figure 10. The speed at which the sensor is placed can be seen in Table IV with the speed retrieval indicator carried out every 15 minutes.

Based on the data obtained in Table VII, the speed value obtained is not constant. The average speed obtained is

0.796 cm/s or close to 0.80 cm/s. The results obtained show that the lead screw used is not good. This is caused by

https://doi.org/10.24912/ijaste.v1.i1.140-162

TABLE VII. Temperature Sensor Laying Speed

:

| Take to- | Distances(cm) | Time (s) | Speed (m/s) |
|---|---|---|---|
| 1 | 23,6 | 30 | 0,79 |
| 2 | 25,1 | 30 | 0,84 |
| 3 | 22,1 | 30 | 0,74 |
| 4 | 24,1 | 30 | 0,80 |
| 5 | 24,3 | 30 | 0,81 |

many factors. The first factor is the condition of the lead screw that is not straight or has curvature. The second factor is that the supporting pole of the tool used is not new so that the resulting distance between the support pole and the placement of the sensor is not appropriate. The resulting distance between the support posts and the placement of the sensor should be ± 2 mm. If the distance is more than that, then the movement of the temperature sensor will be disturbed. In addition, the surface of the support posts must be flat. The third factor is the placement of the geared motor must also have a support so that when the geared motor rotates, it will move the lead screw, not the geared motor that rotates without turning the lead screw.

c. Quality of Service (QoS) Testing

At this stage, QoS testing is carried out to determine whether the system used in making this final project is reliable, efficient, and effective. References for QoS testing can be seen in section 2.5 regarding Quality of Service (QoS). The things analyzed consist of 2 things, namely the latency between data from ESP32 to the website and packet loss that occurs during the data transmission process.

Latency is the lag between the time the data is sent and the time the data is received at the destination. The results of the latency calculation can be seen in Table VIII.

https://doi.org/10.24912/ijaste.v1.i1.140-162

TABLE VIII. Latency Value

:

| Take to- | Data Received (field) | Data transmission time (ms) | Latency (ms) |
|---|---|---|---|
| 1 | 64 | 194 | 3.03 |
| 2 | 64 | 175 | 2.73 |
| 3 | 64 | 154 | 2.41 |
| 4 | 64 | 172 | 2.69 |

Based on Table VIII, there is a latency range for data retrieval between 2.41 ms to 3.03 ms and an average of 4 times the data retrieval, the amount of latency obtained is 2.72 ms. If the results obtained are compared with the literature listed in Table VIII, the latency that occurs in data collection for this final project can be categorized as very good because it is smaller than 150 ms and has an index of 4.

Packet loss is the amount of data lost during data transmission from origin to destination. The results of the calculation of packet loss in the test can be seen in Table IX.

TABLE IX. Packet Loss Value

:

| Take to- | Data Sent (field) | Data Received (filed) | Packet Loss (%) |
|---|---|---|---|
| 1 | 64 | 64 | 0 |
| 2 | 64 | 64 | 0 |
| 3 | 64 | 64 | 0 |
| 4 | 64 | 64 | 0 |

Based on Table IX, the value of packet loss is 0%. If the results of packet loss in Table IX are compared with the theory listed in Table VI, then the packet loss obtained can be classified as very good because there is no data loss and can be classified at index 4. The results of the QoS parameters can be seen in Table X. Based on Table X, it is found that the monitoring network created has a very good QoS category with an average index of 4. This value indicates that the network created is classified as reliable.

TABLE X. QoS Parameter Value

| QoS Parameter | Index | Category |
|---|---|---|
| Latency | 4 | Verry Good |
| Latency | 4 | Verry Good |
| Average | 4 | Verry Good |

## CONCLUSION

From the results and tests that have been carried out on a three-phase induction motor temperature monitoring system using IoT, the following conclusions can be drawn:

1. The component used, namely AMG8833, can be used for temperature monitoring and works well. It can be seen that the accuracy of AMG8833, when compared to infrared thermometers, is still in the range of 2.50C.
2. The lead screw used is not in good condition so it affects the movement speed and position of the temperature sensor array. This is caused by the condition of the lead screw that is not straight, the support is not flat, the distance between the support and the placement of the temperature sensor is not constant and the geared motor has no support.
3. The Internet of Things system used in the three-phase induction motor temperature monitoring system is running very well. This can be seen using the QoS parameters.

## ACKNOWLEDGMENTS

Thanks to related parties who have helped the author in compiling this journal. The author apologizes if there are still shortcomings in this journal and is willing to accept input for the sake of making this journal better.

## REFERENCES

1. A. Glowacz and Z. Glowacz, "Diagnosis of the three-phase induction motor using thermal imaging," Infrared Physics Technology 81, 7–16 (2017).
2. N. Saad, M. Irfan, and R. Ibrahim, "Condition monitoring and faults diagnosis of induction motors," (CRC Press, New York, 2018) Chap. 1, pp. 1–2, 1st ed.
3. G. Singh, T. C. A. Kumar, and V. N. A. Naikan, "Fault diagnosis of induction motor cooling system using infrared thermography," IEEE 6th International Conference on Power System (ICPS) , 1–4 (2016).
4. R. H. C. Palacios, I. N. D. Silva, and W. F. Godoy, "A novel multi-agent approach to identify faults in line connected three-phase induction motors," Applied Soft Computing 45, 1–10 (2016).
5. E. A. Elisabeth and Jebaseeli, "Monitoring the thermal behavior of induction motor using regression technique," Innovations in Power and Advanced Computing Technologies , 1–4 (2017).
6. W. Gu, Q. Huang, S. He, G. Ma, and Q. Wang, "Industrial intelligent optimization system based on industrial internet platform," IEEE 11th International Conference on Software Engineering and Service Science (ICSESS) , 523–526 (2020).
7. Chapman and S. J., "Electric machinery fundamental," (McGraw-Hill, New York, 2012).

8. J. Faiz, V. Ghorbanian, and G. Joksimovic, “Fault diagnosis of induction motors,” (The Institution of Engineering and Technology, London, 2017).

9. A. B. Haripiya, K. A. Sunitha, and B. Mahima, “Development of low-cost thermal imaging system as a preliminary screening instrument,” Procedia Computer Science 172, 283–288 (2020).

10. U. Jayalatsumi, A. F. Naaz, K. Sravani, A. Anusha, and A. Vasavi, “A low cost thermal imaging system for medical diagnostic applications,” International Journal of Engineering Technology 7, 314–317 (2018).

11. P. Smutny, M. Babiuch, and P. Foltynek, “Using the esp32 microcontroller for data processing,” 20th International Carpathian Control Con-ference (ICCC) , 1–6 (2019).

12. “L293d push-pull four channel driver with diodes,” STMicroelectronics (datasheet).

13. D. K. Halim, N. M. Song, and D. Hartono, “Arduino-based ide for embedded multi-processor system-on-chip,” 5th International Conference on New Media Studies (CONMEDIA) , 135–138 (2019).

14. Nixon and Robin, “Learning php, mysql javascript: With jquery, css html5,” (O’Reilly Media, USA, 2014) 4th ed.

15. H. Krishnan, M. S. Elayidom, and T. Santhanakri, “Mongo-db–a comparison with nosql databases,” International Journal of Scientific and Engineering Research 7, 1035–1037 (2016).

16. “Mongodb,” https://docs.mongodb.com/manual/core/document (Accessed 12 January 2022).

17. J. Xu, Z. Zheng, and M. R. Lyu, “Web service personalized quality of service prediction via reputation-based matrix factorization,” IEEE Transactions on Reliability 65, 28–37 (2016).

18. Fahmi and Hasanul, “Analisis qos (quality of service) pengukuran delay, jitter, packet loss dan throughput untuk mendapatkan kualitas kerja radio streaming yang baik,” Teknologi Informasi dan Komunikasi 7, 98–105 (2018).

19. S. W. Pamungkas, Kuseini, and E. Pramono, “Analisis quality of service (qos) pada jaringan hotspot sma negeri xyz,” Sistem Informasi dan Teknologi Informasi 7, 142–152 (2018).

20. G. Ash, B. Davie, J. Evans, A. Farrel, C. Filsfils, P. Loshin, D. Medhi, M. Morrow, R. M. Perea, L. L. Peterson, K. Ramasamy, J. Strassner, K. Vijayananda, and Z. Wang, “Network quality of service know it all,” (Elsevier, Burlington, 2009).

21. T. Szigeti, R. Barton, and C. Hattingh, “End-to-end qos network design,” (Cisco, USA, 2014) 2nd ed.

22.