Journal of Engineering Science and Technology
Vol. 20, No. 4 (2025) 1069 - 1088

IOT BASED INDUCTION MOTOR PID SPEED CONTROL SYSTEM USING ZIEGLER – NICHOLS METHOD

CHARLES RONALD HARAHAP*,
F. X. ARINTO SETYAWAN, CAHYA ANDIKA SALSABILA

Department of Electrical Engineering, Faculty of Engineering, University of Lampung,
Jalan Sumantri Brojonegoro No. 1, Bandar Lampung, Lampung, 35141, Indonesia
*Corresponding Author: charles.harahap69@gmail.com

Abstract

This research aimed to determine the Proportional Integral and Derivative (PID) parameters in a three-phase induction motor (TPIM) speed control system using the Ziegler-Nichols oscillation method. Presently, the three-phase inverter used for regulation, was designed with MOSFET (Metal Oxide Semiconductor Field Effect Transistor) through the PWM (Pulse Width Modulation). This method is unable to maintain a stable motor speed during a disturbance or when under a load, which led to the use of the Ziegler-Nichols oscillation method with efficiency in determining PID parameters. The Internet of Things (IoT) was implemented using the Blynk application with the NodeMCU ESP8266 module as a platform and microcontroller. The result led to the successful design of a TPIM speed control system. The closed-loop control system provided good speed response results, with an average rise time of 2.2 seconds without load and an overshoot of 1.06%. When disturbance is given to the TPIM, the recovery speed was immediately set to 2 seconds in accordance with the ideal characteristics for both step and disturbance response. The application of the IoT through the Blynk application successfully controlled and monitored the TPIM remotely. In conclusion, this research successfully designed the proposed system, and the experimental results were presented to prove the feasibility of the proposed method.

Keywords: Inverter, IoT, PID, Three-phase induction motor, Ziegler-Nichols.

1. Introduction

The Ziegler-Nichols method is used for tuning proportional integral and derivative (PID) controller parameters on three-phase induction motor (TPIM), including monitoring and controlling it remotely through smartphones in real time. TPIM is used in many industrial applications such as pumps, fans, compressors, conveyors, cranes, and elevators. It is also used in the transportation sector, namely electrical cars and trains. Meanwhile, induction motor is preferable due to the characteristics of the motor and operational activities which are simpler, reliable, and requires less maintenance when compared to the other types [1]. The industrial field requires a type of motor whose speed can be adjusted according to needs and desires. The speed of a TPIM can be fine-tuned by adjusting the frequency using an inverter [2], consisting of power electronic components. This includes MOSFET (Metal Oxide Semiconductor Field Effect Transistor), controlled by switching the PWM (Pulse Width Modulation). The technique is adopted where the output voltage can be adjusted as needed resulting in varied motor speed.

Based on this, regulating the speed of a TPIM using an inverter had not been able to provide a stable induction motor speed and when under load is affected by any disturbance. The motor performance tends to decrease, disrupting work in the industrial sector. Therefore, a closed loop control system must be applied to stabilize and maintain the desired speed of a TPIM when there is a disturbance or added load. In the present research, the Ziegler-Nichols method was applied to determine the values of Kp, Ki and Kd in the PID controller. This was due to the efficiency in determining suitable parameters by providing a more structured approach in a relatively short time, compared to the trial and error method which requires more experiments [3].

The Internet of Things (IoT) is currently the main pillar of industrial transformation [4], and the integration of a TPIM enabled remote monitoring and control. This entails the adoption of the NodeMCU ESP8266 module as an IoT platform and microcontroller, with the Blynk application used for controlling the Arduino and monitoring the data from the motor. In addition, this information can be accessed in real time from a mobile device, allowing better monitoring and management, compared to having to oversee and control the motor directly which is tiring and less efficient.

Several previous research had been conducted on PID controllers, and according to [5-7], these are applied to DC motors using mathematical models, namely Simulink/MATLAB. The research reported that PID controllers offered the best performance compared to PI and Fuzzy Logic Controllers. Prior research [2] used a PID controller to regulate the speed of a TPIM, comparing it to an open loop control using an inverter. Furthermore, performance analysis was carried out using Simulink/MATLAB. The research by [1, 8, 9] used a PID controller to monitor the direct torque of an induction motor. These controllers can be tuned using several PID methods such as the Ant Colony Optimization [10]. The research by Ferdiansyah et al. [3] used a combination of the Ziegler-Nichols oscillation and IFOC methods to control the speed of a TPIM. However, Premkumar et al. [11] combined fuzzy logic methods with PID to resolve windup problems in the control system. Bharti et al. [12] adopted a closed loop PI controller through Simulink/MATLAB to reduce rotor speed harmonics, as well as designed open loop control using an inverter. Several previous research had been conducted on IoT, for example Maity et al. [13] used the Blynk

application to control smart homes remotely with Arduino nano and Bluetooth module. Meanwhile, Othman and Zakaria [14] monitored the base of energy meter remotely through Blynk application using ESP32.

Based on previous research, the PID method was widely applied to TPIMs, through a simulation process using MATLAB software. Accurate results were obtained, because environmental variables not contained in the simulation model, were not considered. This led to the retesting and incorporation of previous findings into a TPIM speed controller hardware. The contribution of the research is the application of the PID control system to the induction motor speed controller hardware. In addition, IoT was applied to this system, enabling monitoring and speed control through smartphone with internet connection.

The research is organized in the following manner section 1 focused on introduction, describing the background, previous research, contributions, and novelty. Meanwhile, section 2 described the methods and system design used, including hardware and software implementation. Section 3 presented the experimental results and discussions, outlining the performance of the PID controller and the IoT-based monitoring system. Finally, Section 4 focused on conclusion, summary of the findings and recommendations for future work.

2. Methods

2.1. System design

Figure 1 shows the research flowchart outlining the sequential steps adopted in designing, implementing, and testing a TPIM control system.

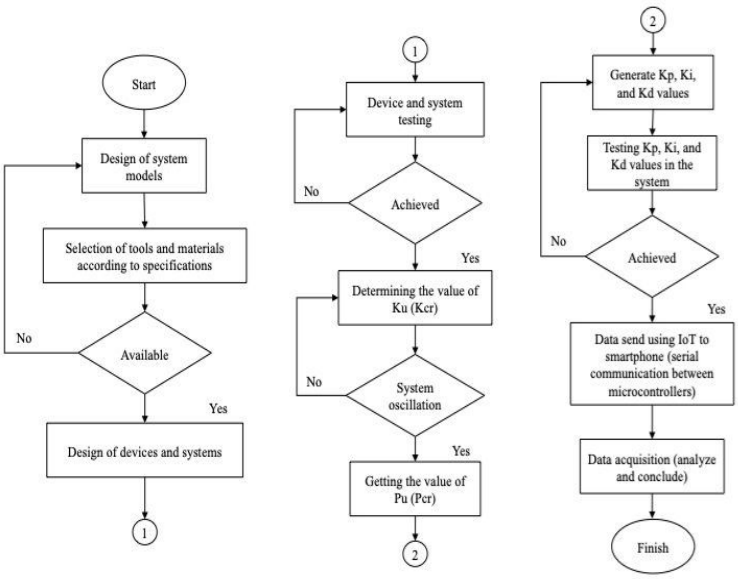


Fig. 1. Research flow chart.

The research reported that the TPIM operated under balanced voltage and load conditions. External disturbances were limited to step changes in load and

temperature variations, considered negligible. Additionally, the system model was used to approximate the motor as a linear time-invariant, neglecting minor nonlinearities in the rotor dynamics to simplify the PID controller design.

Figure 2 shows the overall system block diagram, with the TPIM (731404 Class 0.1) coupled to the load, which is a DC generator. A rotary encoder with an LM393 module considered as a speed sensor, was placed on the DC generator rotor. The output of the generator was connected to the load in the form of three 5 Volt DC lamps with a total power of 15 watts.

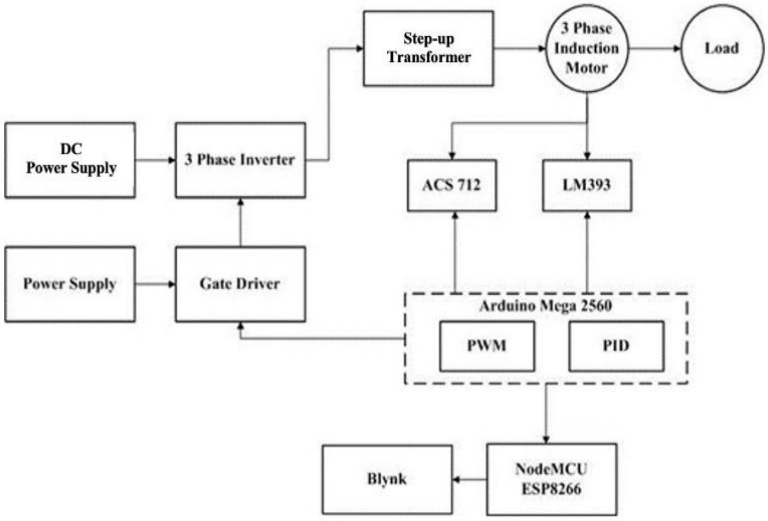


Fig. 2. System block diagram.

Several system modules, such as the DC power supply, gate driver, and three-phase inverter were designed. These were realized using the Easy EDA application, printed in the form of PCB.

The system is started with a 220 Volt power source, lowered by the supply module to 15 Volts, considered the input voltage for the IC on the gate driver. Additionally, it is connected to the Arduino Mega 2560 circuit, which has a switching program installed. The switching arrangement moves towards the three-phase inverter, with power supplied from another DC, namely the MOSFET working voltage in the inverter. The switching process and speed regulation of the induction motor occurred in the three-phase inverter through changing frequency settings. Furthermore, the step-up transformer increased the output voltage of the three-phase inverter to supply the TPIM. The rotation speed, read by the sensor, provided feedback used by the PID control system. The ACS 712 current sensor monitored the current value in the system, with similar program also installed on the Arduino Mega 2560. This was also connected through serial communication with the NodeMCU ESP8266, linked to the internet network. The NodeMCU ESP8266 was also connected to the Blynk IoT application for controlling the TPIM remotely.

2.2. Model of PID control

The research adopted a control system that combined three components, namely proportional, integral, and derivative [15-17]. The PID control model used is represented in Fig. 3, showing the relationship between the proportional, integral, and derivative components in the system. This model served as the basis for determining the PID controller parameters using the second Ziegler-Nichols tuning method, commonly referred to as the oscillation method. Based on Fig. 3, the input signal is the desired reference speed, compared with the actual speed of the motor. The difference between the reference and actual speed is called the error, which serves as the input of the PID controller. The output generated is sent to the TPIM as the speed controller input. The motor tries to adjust the speed according to the reference speed. Meanwhile, the speed sensor (Encoder) is used to read the actual speed of the motor, providing feedback to the system. The first step in carrying out this method entailed setting the Ki and Kd values to zero (Ki = 0, Kd = 0), because the system works only with the proportional controller (Kp). Additionally, Kp is set from 0 to a value where the system reaches an oscillation condition with relatively the same amplitude consistently [18]. This condition is referred to as sustained oscillation, and the Kp value that causes oscillation is called the critical Ku (ultimate gain), while the period of sustained oscillation is termed Pu (ultimate period) [19].

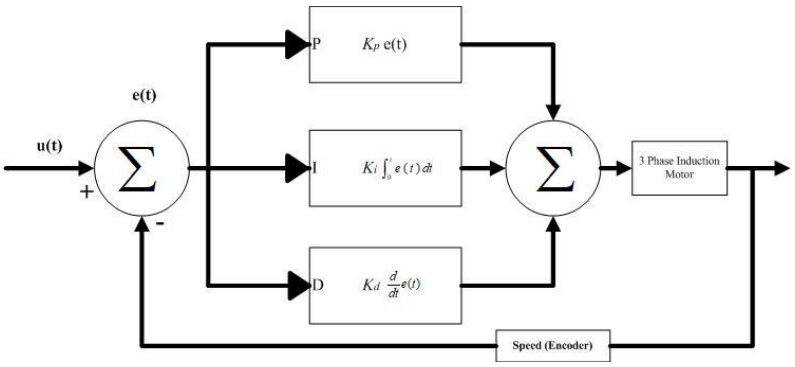


Fig. 3. Diagram of PID control [2].

After obtaining the values of Ku (temporary Kp) and Pu, the Kp, Ti, and Td values were calculated using the following Ziegler-Nichols second method PID determination shown in Table 1 [19, 20].

Table 1. PID controller parameters.

Controller Settings	Ziegler – Nichols Closed loop Oscillation Method
Kp	0.6 Ku
Ti	0.5 Ku
Td	0.125 Pu

However, after obtaining the values of Kp, Ti and Td, PID control algorithm was calculated using Eq. (1) [21, 22].

$$u(t) = k_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t)dt + T_d \frac{de(t)}{dt} \right] \quad (1)$$

where k_p is proportional gain, T_i is integral time constant, $u(t)$ is output control quantity, $e(t)$ is deviation therefore, with the Ki and Kd values calculated using Eq. (2).

$$k_i = k_p \times \frac{T}{T_i};\ k_d = k_p \times \frac{T_d}{T} \quad (2)$$

2.3. Specification of motor and load

The induction motor used as a plant or controlled object in this research is a TPIM Type 731404 Class 0.1. In addition, the adopted parameters are shown in Table 2.

Table 2. Specifications of TPIM.

Parameter	Value (Unit)
Voltage Rating	400V (Y) /230V (Δ)
Current Rating	0.45A (Y) /0.78A (Δ)
Power Rating	0.12 Kw
Frequency Rating	50 Hz
Speed Rating	1380 U/min
Cos φ	0.67

Source: Three-phase Induction Motor Type 731404 class 0.1 at the Electrical Energy Conversion Laboratory, Faculty of Engineering, Department of Electrical Engineering, University of Lampung.

The induction motor was coupled to a DC generator (Type 73121) as the load, with total output power of 15W, represented by three 5V DC lamps. Furthermore, the parameters of the DC generator used are shown in Table 3.

Table 3. Specifications of DC generator.

Parameter	Value (Unit)
Voltage Rating	220 V
Current Rating	0.7 A
Power Rating	0.1 kW
Speed Rating	2000 U/min
Cos φ	0.67

Source: DC Generator Type 73121 at the Electrical Energy Conversion Laboratory, Faculty of Engineering, Department of Electrical Engineering, University of Lampung.

2.4. Design of hardware control of induction motor

In this research, a TPIM and gate driver module were designed to connect the voltage control signal (G) and MOSFET source (S). These were designed using several components such as IC Optocoupler HCPL 3120, 47 Ω resistor, 220 Ω resistor, and 100nF Mylar capacitor. The inverter module comprised six IRFP460 MOSFET components, and Elco 2200 μF 180 V capacitors. The HCPL 3120 Optocoupler IC causes the voltage to function based on the DC power supply module. This was designed from the following components, a 220/15V 1 Ampere CT transformer, FR207 diode, LM7815 regulator IC, 3300 μF and 100 μF capacitors. Additionally, all the components and circuits were designed using the easy EDA application, as shown in Fig. 4, and three phase inverter circuit is shown in Fig. 5.

Based on Fig. 5, the MOSFET components used in the three-phase inverter circuit were arranged in a bridge configuration. Table 4 shows the inverter switching configuration, with a conduction mode of 180 degrees. In line with Table 4, the MOSFET switching scheme and inverter output voltage are shown in Table 5.

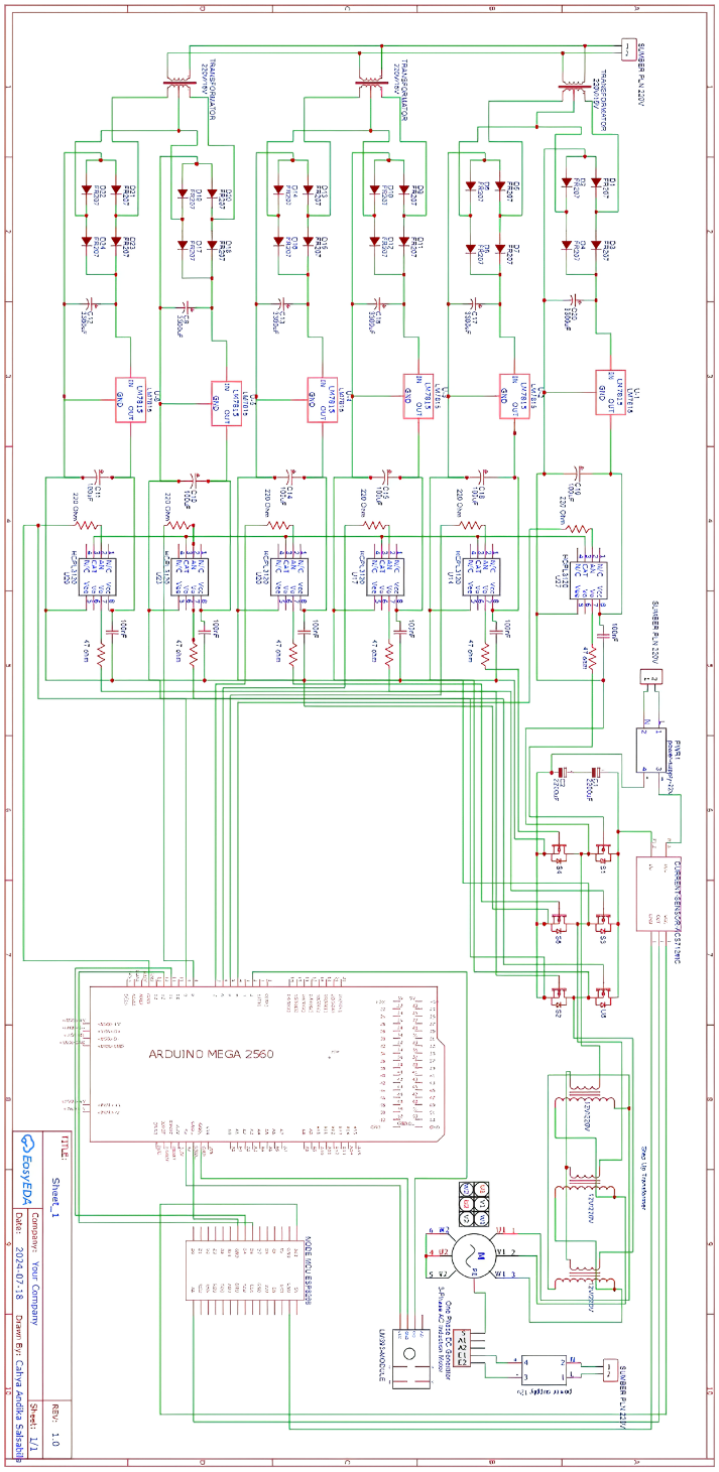


Fig. 4. One-line diagram of hardware control of induction motor.

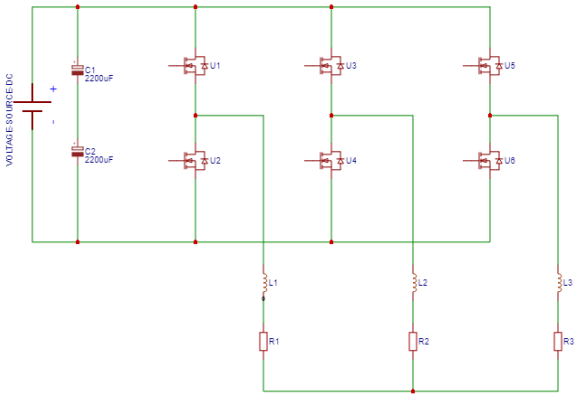


Fig. 5. Three-phase inverter.

Table 4. Three-phase inverter switching configuration.

S1	S2	S3	S4	S5	S6
1	0	0	0	1	1
1	1	0	0	0	1
1	1	1	0	0	0
0	1	1	1	0	0
0	0	1	1	1	0
0	0	0	1	1	1

Table 5. PWM inverter switching schematic.

Interval	Switch (*ON*)	Phase Voltage			Interphase Voltage		
		Va	Vb	Vc	Vab	Vbc	Vca
1	1,6,5	1/3 Vdc	-2/3 Vdc	1/3 Vdc	Vdc	-Vdc	0
2	1,6,2	2/3 Vdc	-1/3 Vdc	-1/3 Vdc	Vdc	0	-Vdc
3	1,3,2	1/3 Vdc	1/3 Vdc	-2/3 Vdc	0	Vdc	-Vdc
4	4,3,2	-1/3 Vdc	2/3 Vdc	-1/3 Vdc	-Vdc	Vdc	0
5	4,3,5	-2/3 Vdc	1/3 Vdc	1/3 Vdc	-Vdc	0	Vdc
6	4,6,5	-1/3 Vdc	-1/3 Vdc	2/3 Vdc	0	-Vdc	Vdc

2.5. Internet of things

The research adopted NodeMCU, in connection with Arduino Mega 2560, and an internet/Wi-Fi network to connect with the Blynk cloud server. In addition, smartphone devices served as monitoring and control media. The device also issues commands to the NodeMCU, which then sends the data to the Arduino Mega microcontroller for processing. The application of IoT in motor control addressed the limitations of traditional methods, which required on-site supervision, and lacked real-time monitoring capabilities. The integration also enabled remote monitoring and control of the three-phase induction motor through the Blynk application. This provided a user-friendly interface for adjusting motor speed and observing the following critical parameters speed and current in real time. The integration of IoT was particularly beneficial in industrial settings where efficiency, scalability, and quick troubleshooting are crucial. Furthermore, the adoption process is in line with

the current transformation to Industry 4.0, where interconnected systems played a relevant role in enhancing productivity and operational insights.

The IoT [23, 24] in this research undergoes a process to ensure the functionality, as shown in Fig. 6. In the Physical layer [25], the ACS712 current [26, 27] and LM393 speed sensors [28, 29] were connected to the microcontroller [30, 31]. The ACS712-20A sensor was connected to the NodeMCU microcontroller through the Vcc pin with the 3.3 V pin, the OUT pin with the NodeMCU A0 pin, and the GND pin of the sensor and NodeMCU were interconnected. The LM393 sensor was connected to the Arduino Mega microcontroller through the Vcc pin with the 5V pin, the D0 pin with pin 2, while the GND pin of the sensor and Arduino Mega were interconnected, with the data read by the sensor processed in the next stage. This occurred in the Gateway Layer section, where two microcontrollers [31] namely Arduino Mega 2560 and NodeMCU ESP8266 [32-34] were connected through serial communication. Additionally, pins 11 and 12 Arduino Mega were connected with pin RxD5 and TxD6 NodeMCU, with the GND pin connecting Arduino Mega and NodeMCU. The data contained in Arduino Mega was subsequently sent to NodeMCU. In NodeMCU ESP8266, a Wi-Fi module [30, 35] was installed to send the data to the software through the internet network. Furthermore, the Middleware Layer consisted of the Blynk Cloud Server [36] which functioned as a data storage sent from the NodeMCU ESP8266 and a bridge connecting the equipment to the user through the Blynk application. The final part or aspect of the Application Layer was the software used namely Blynk [37].

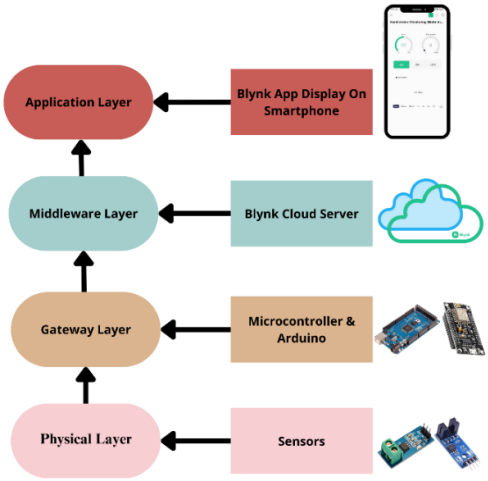


Fig. 6. Architecture of IoT modelling.

Figure 7 shows the overall circuit of the IoT design, on the microcontroller, the program was embedded through the Arduino IDE software [32, 38]. In order to ensure the functionality of Blynk [25], the Blynk and Wi-Fi libraries [39] must be installed on the Arduino IDE, passing through the token obtained from the server after designing a project [33] on Blynk. After the connection process and the program is successfully running, the user can interact with the equipment, from anywhere and at any time. The user can also control and monitor the speed of the TPIM using the internet network through the application.

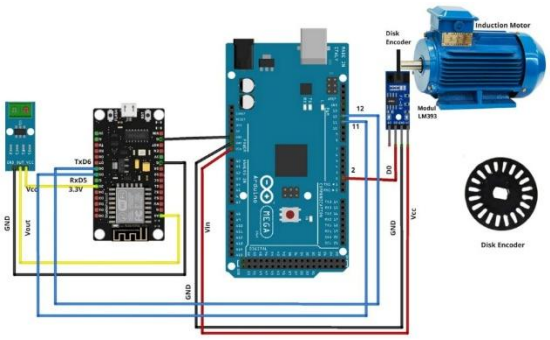


Fig. 7. Internet of things-based system control and monitoring circuit.

The IoT system adopted username and password-based authentication to ensure only authorized user access and control the induction motor through the Blynk IoT application. Each user must enter unique credentials before being able to access the monitoring and control features.

3. Result and Discussion

3.1. Experimental set up

The induction motor used, and the overall circuit design of the experimental setup are shown in Figs. 8 and 9, respectively. Additionally, these diagrams depicted the integration of various components used in the system.

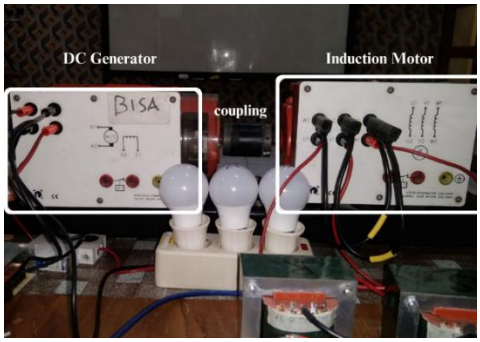


Fig. 8. Induction motor.

Figure 9 shows that the overall circuit design, consisting of a power supply module was connected to a gate driver. This module connects the power supply with a three-phase inverter circuit, with an output in the form of AC voltage raised by a step-up transformer to drive a TPIM. The motor was coupled with a load circuit, namely a single-phase DC generator. Furthermore, the output of the DC generator was connected to the load in the form of three DC lamps with a total power of 15 watts. The rotation speed of the plant was detected by the LM393 speed sensor with data processed as feedback for the PID controller. The current in the system was also detected by an ACS712 current sensor monitored to avoid early surges.

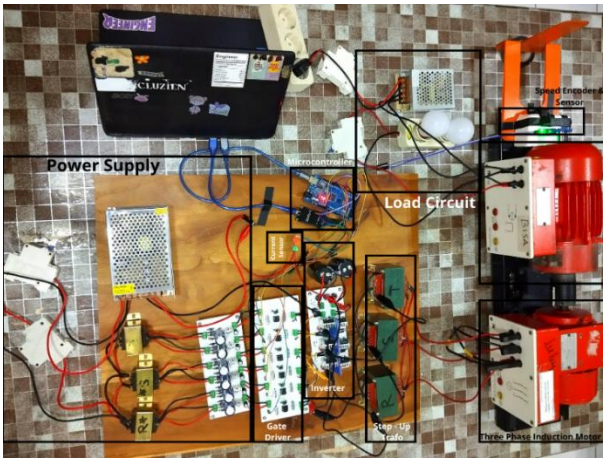


Fig. 9. Overall circuit design.

Based on this, induction motors had more complex non-linear characteristics than DC, with proportional and integral control insufficient to handle significant load variations or sudden disturbances. As a result, derivative components were often required to maintain system stability under various operating conditions. The PID parameters in this research were determined using the Ziegler- Nichols second method.

These parameters were determined to obtain the Ku (Ultimate gain) or Kcr (critical) value responsible for producing stable oscillating waves. However, from the tuning process, the resulting value Ku = 0.03121.

Using Ku = 0.03121, the system response was obtained as shown in Fig. 10, with the Ultimate Period (Pu) value determined for two seconds. This value was calculated using the Ziegler-Nichols formula stated in Table 1. In addition, Eq. (2) was then used to obtain the value in Table 6.

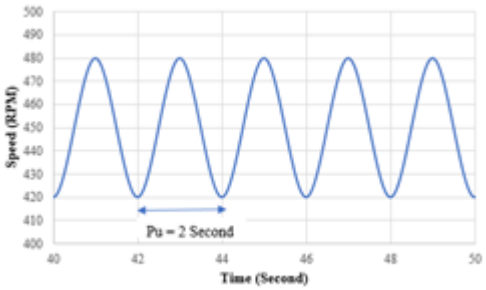


Fig. 10. Motor speed oscillation period.

Table 6. PID controller parameters.

Parameter	Value
Kp	0.018726
Ki	0.18726
Kd	0.0046815

3.2. Experimental results

The microcontroller Arduino Mega 2560 was used to generate a PWM signal [40], adopted for switching on a multilevel inverter with 16 MOSFETs. In this research, PWM was used to perform inverter switching with a total of six MOSFETs. The following digital pins four, five, six, seven, eight, and nine, were used to issue switching commands. The output voltage of the Arduino Mega 2560 is 5V serving as the voltage source of the gate driver IC to issue on/off commands to the MOSFET. This test was carried out by examining the PWM wave and inverter output frequencies of 20 Hz, 30 Hz, and 40 Hz. The frequency selection was based on determining the set point, with the PWM output waveform shown in Fig. 11.

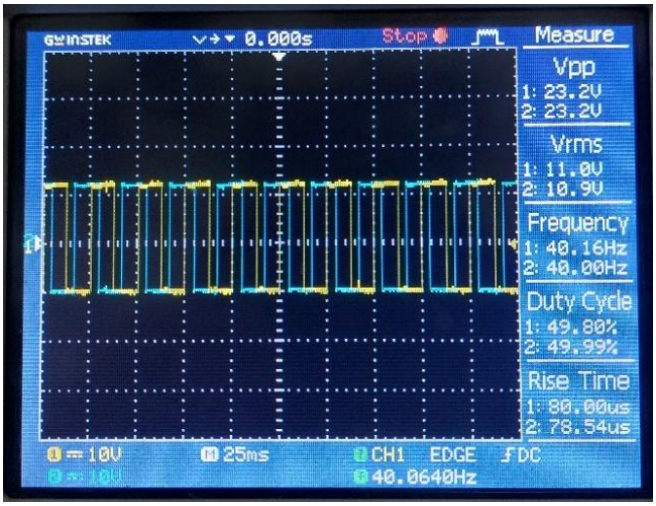
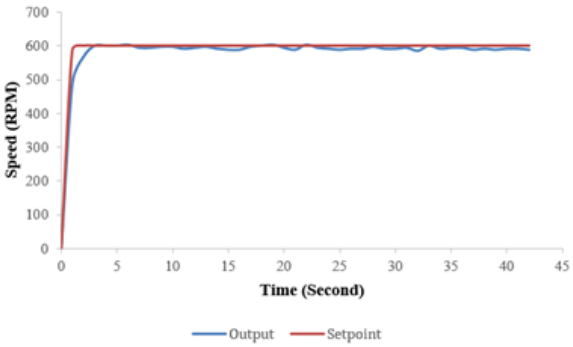


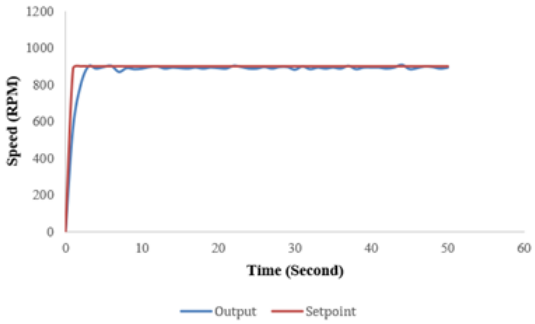
Fig. 11. PWM output wave.

The PID controller parameter testing was conducted at the following set-points, 600 rpm, 900 rpm, and 1200 rpm. This was realized by observing the response of the system to the motor speed under no-load and load conditions by changing the set-point values. The speed response is shown in Fig. 12.

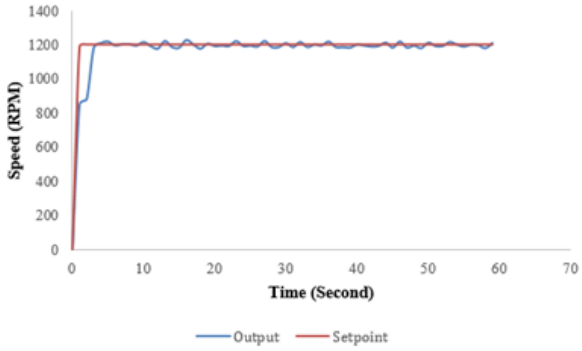
Based on the speed response in Fig. 12, rise, and settling time, including overshoot data are shown in Table 7. In Fig. 12(a) the speed response graph of the TPIM without load when the set point is 600 rpm, shows the motor response in respect to PID control. The results showed that when the set point is at 600 rpm, the system has a rise and settling time of 1.53 seconds, and 3.6 seconds, respectively with a maximum overshoot and undershoot of 0.5% and 0%. In Fig. 12(b) the speed response graph of the TPIM without load when the set point is at 900 rpm, shows the motor response when given PID control. The results implied that when the set point is at 900 rpm, the system has a rise and settling time of 2.56 seconds, and 3.9 seconds, with a maximum overshoot and undershoot of 1% and 0%, respectively. Figure 12(c) shows the speed response graph of a TPIM without load when the set point is at 1200 rpm, depicting the motor response when given PID control. The results implied that when the set point is 1200 rpm, the system has a rise and settling time of 2.52 seconds, and 3.9 seconds, with a maximum overshoot and undershoot of 1.75% and 0%, respectively.



(a)



(b)



(c)

Fig. 12. Step response of induction motor speed without load (a) at 600 rpm (b) at 900 rpm (c) at 1200 rpm.

Table 7. PID step response of three-phase induction motor speed.

Set Point (RPM)	Rise Time (sec)	Settling Time (sec)	Overshoot (%)
600	1.53	3.6	0.5
900	2.56	3.9	1
1200	2.52	3.9	1.7
Average	**2.2**	**3.8**	**1.06**

In line with the step response of the induction motor, Fig. 13 shows the speed response graph of a TPIM when given a load. The test was conducted by controlling the speed of the TPIM at a set point of 600 rpm, which was later changed at 1200 rpm. When the speed is 1200 rpm, the TPIM was disturbed at the 24th second of three dc lights with a total load of 15 watts. As a result, the motor speed decreases, with the speed of the TPIM returning to the set-point for 2 seconds.

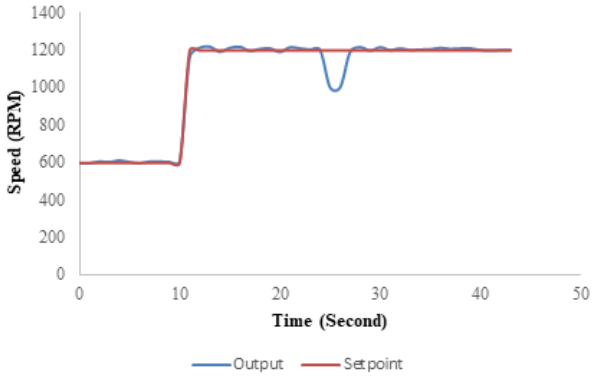


Fig. 13. Step response of induction motor speed with load.

Examining the system output graph, the Ziegler-Nichols method was adopted for this system, because it is easier to use compared to the trial and error method. The second Ziegler-Nichols or oscillation method was used to determine the values of Kp, Ki, and Kd. This method is brief, and the requirement fulfilled by analysing the oscillation of the output graph, in accordance with the PID tuning provisions.

Based on the results obtained, the speed response generated was consistent with the ideal response. The IoT can be applied to control and monitor the induction motor in this research. Furthermore, the Ziegler-Nichols tuning method was easily used to determine the values of Kp, Ki, and Kd.

Speed control and current monitoring were observed using the Blynk IoT software. The application of IoT was implemented to monitor and control the speed of a TPIM, rather than tuning PID remotely. The designed Blynk software is shown in Fig. 14. From the application, speed (RPM) and current (Ampere) monitoring can be implemented. In addition, speed tend to be controlled by pressing the selection button. The application was also used to plot the step response graph of TPIM speed.

In accordance with several previous research, the application of the PID controller was carried out using MATLAB Simulink. Accurate results were obtained, but the limitation was that the outcome seemed invalid when affected by environmental factors. This research implemented PID on TPIM hardware, with the results obtained shown in Table 7. The novelty is the application of the PID control system designed using the Ziegler-Nichols method. The induction motor speed controller hardware was used to provide a speed response in the form of step and disturbance responses. Previous research designed PID using Ziegler-Nichols only on step response. In addition, the novelty included designing a PID controller with Ziegler-Nichols method based on control and monitoring using the IoT, which produced a speed response in the form of step and disturbance responses.

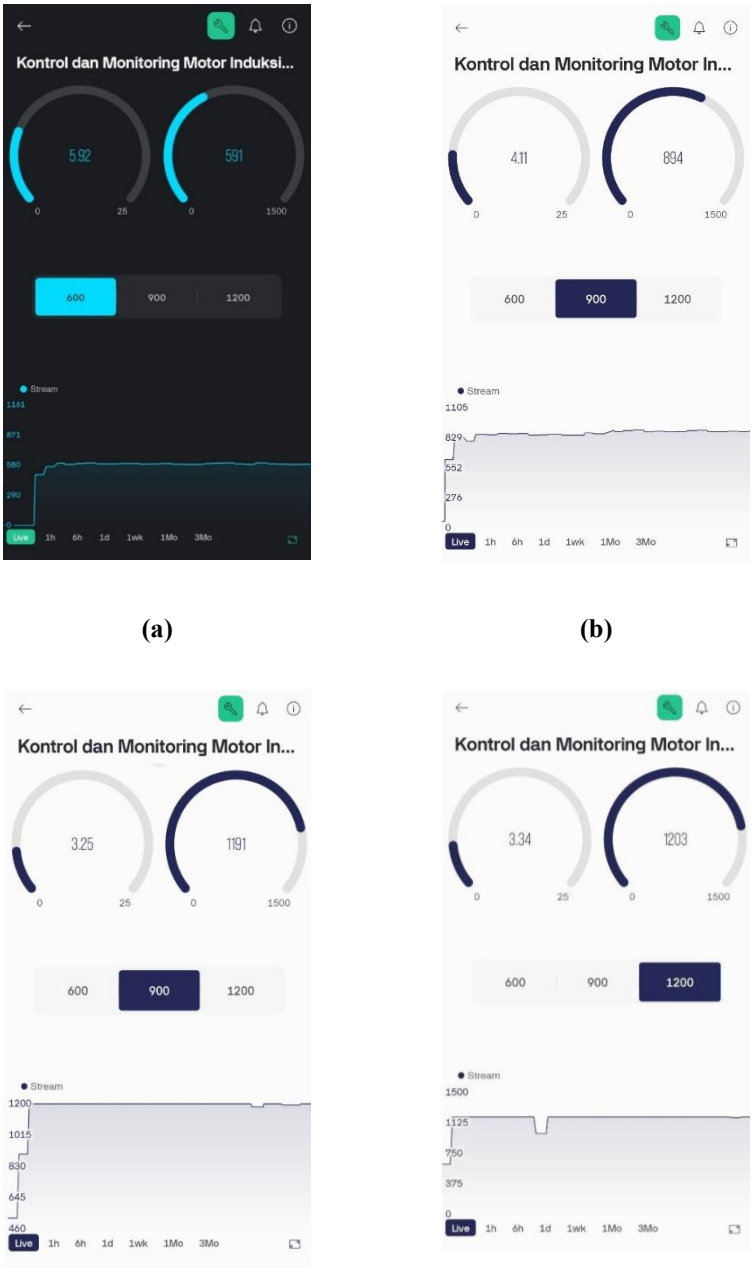


Fig. 14. Display Blynk (a) at Set-point 600 rpm, (b) at Set-point 900 rpm, (c) at Set-point 1200 rpm, (d) under load testing.

The title of the GUI, *Kontrol dan Monitoring Motor Induksi Tiga Fasa*, is translated to Control and Monitoring of Three-Phase Induction Motor in English. This explanation aimed to provide clarity for native speakers while maintaining the original language for easy understanding by users. The error difference between the Blynk application and real-time display under load and no-load conditions was caused by changes in the internet network. This led to a slight error, with the results displayed in real-time.

This drive system was designed to operate within the base speed of a TPIM. However, with appropriate modifications to the control algorithm and inverter parameters, the system had the potential to support field weakening operation enabling speeds outside the nominal range.

Some of the experimental equipment used was personally designed, for example the power supply, inverter, and gate driver modules, including disk encoder, resulting in efficient research costs.

The Ziegler-Nichols PID control and IoT-based induction motor speed control applications were successfully implemented, although the system had certain limitations. It was designed to control only motor speed, in one direction, and the implementation of speed reversal also required additional modifications to the inverter including control algorithms to manage safe transitions between positive and negative speeds. The vector control system was not used in this system, so an inner current loop was absent from the control system. Furthermore, the system relied on a stable internet connection for remote control and monitoring, limiting the application in areas with poor network coverage.

The proposed IoT-based induction motor PID speed control system showed significant potential for practical applications. In industrial automation, the system was used for the precise speed control of conveyor belts, pumps, and fans, ensuring operational efficiency and reliability. Additionally, the ability to provide real-time monitoring and remote control made it ideal for energy management systems. This was particularly evident in water treatment plants and power distribution networks, where optimized motor performance led to substantial energy savings.

The association with Industry 4.0 principles facilitated the development of smart manufacturing environments, using interconnected systems to enhance productivity and decision-making. These applications focused on the versatility of the proposed system, offering solutions for diverse sectors demanding reliable and efficient motor control.

4. Conclusions

In conclusion, a TPIM speed control and monitoring system using the Ziegler-Nichols PID controller was successfully designed with the tuning outcome resulting in a parameter value of Kp = 0.018726, Ki = 0.018726, and Kd = 0.0046815. This was aimed to generate an exceptional TPIM speed response step. During the occurrence of additional load and disturbance, the PID controller quickly fixed the motor rotation back to the set point with an average rise time and overshoot of 2.2 seconds and 1.06%, respectively. The IoT was also successfully carried out using the Blynk IoT application, ensuring speed monitoring and controlling of a TPIM was performed remotely using a smartphone. Although this

research focused on a TPIM, the proposed Ziegler-Nichols PID tuning method was adapted for other motor types, such as DC, BLDC, and synchronous motors. With necessary adjustments based on the dynamic characteristics of each motor type, this method had the potential to deliver optimal performance and stability in various applications. Future research should combine the PID controller method with other types such as fuzzy logic controller.

Nomenclatures

$e(t)$	Deviation
K_d	Derivative Gain
K_i	Integral Gain
K_p	Proportional Gain
T_d	Derivative Time Constant
T_i	Integral Time Constant
$u(t)$	Output Control Quantity

Abbreviations

IoT	Internet of Things
MATLAB	MATrix LABoratory
PID	Proportional Integral Derivative
TPIM	Three-Phase Induction Motor

References

1. Abdullah, A.N.; and Ali, M.H. (2020). Direct torque control of IM using PID controller. *International Journal of Electrical and Computer Engineering*, 10(1), 617-625.
2. Hartono, H.; Sudjoko, R.I.; and Iswahyudi, P. (2019). Speed control of three phase induction motor using universal bridge and PID controller. *Journal of Physics: Conference Series*, 1381(1), 012053.
3. Ferdiansyah, I.; Raharja, L.P.S; Yanaratri, D.S.; and Purwanto, E. (2019). Design of PID controllers for speed control of three phase induction motor based on direct-axis current (Id) coordinate using IFOC. *Proceedings of the* 4th *International Conference on Information Technology, Information Systems and Electrical Engineering* (*ICITISEE*), Yogyakarta, Indonesia, 369-372.
4. Malik, P.K. et al. (2021). Industrial internet of things and its applications in industry 4.0: state of the art. *Computer Communications*, 166, 125-139.
5. Mahmud, M.; Motakabber, S.M.A.; Alam, A.H.M.Z.; and Nordin, A.N. (2020). Control BLDC motor speed using PID controller. *International Journal of Advanced Computer Science and Applications*, 11(3), 477-481.
6. Hammoodi, S.J.; Flayyih, K.S.; and Hamad, A.R. (2020). Design and implementation speed control system of DC Motor based on PID control and Matlab Simulink. *International Journal of Power Electronics and Drive Systems*, 11(1), 127-134.
7. Koca, Y.B.; Y. Aslan, Y.; and Gokce, B. (2021). Speed control based PID configuration of a DC motor for an unmanned agricultural vehicle.

Proceedings of the 8th *International Conference on Electrical and Electronics Engineering* (*ICEEE*), Antalya, Turkey, 117-120.

8. Mahfoud, S.; Derouich, A.; Ouanjli, N.E.; Mahfoud, M.E.; and Taoussi, M. (2021). A new strategy-based PID controller optimized by genetic algorithm for DTC of the doubly fed induction motor. *Systems*, 9(2), 37.
9. Boukhalfa, G.; Belkacem, S.; Chikhi, A.; and Benaggoune, S. (2019). Genetic algorithm and particle swarm optimization tuned fuzzy PID controller on direct torque control of dual star induction motor. *Journal of Central South University*, 26, 1886-1896.
10. Dhieb, Y.; Yaich, M.; Guermazi, A.; and Ghariani, M. (2019). PID controller tuning using ant colony optimization for induction motor. *Journal of Electrical Systems*, 15(1), 133-141.
11. Premkumar, K.; Thamizhselvan, T.; Priya, M.V.; Carter, S.B.R.; and Sivakumar, L.P. (2019). Fuzzy anti-windup PID controlled induction motor. *International Journal of Engineering and Advanced Technology*, 9(1), 184-189.
12. Bharti, R.; Kumar, M.; and Prasad, B.M. (2019). V/F control of three phase induction motor. *Proceedings of the International Conference on Vision Towards Emerging Trends in Communication and Networking* (*ViTECoN*), Vellore, India, 1-4.
13. Maity, S.; Bagchi, P.; Patra, S.; Maity, S.; and Mukherjee, S. (2021). IoT based home appliance control system using proteus simulation software and blynk server. *International Journal for Research in Applied Science and Engineering Technology*, 9(vii), 3158-3165.
14. Othman, A.; and Zakaria, N.H. (2020). Energy meter based wireless monitoring system using blynk application via smartphone. *Proceedings of the IEEE* 2*nd International Conference on Artificial Intelligence in Engineering and Technology* (*IICAIET*, 2020), Kota Kinabalu, Malaysia, 1-5.
15. Allu, N.; and Toding, A. (2020). Tuning with Ziegler Nichols method for design PID controller at rotate speed DC Motor. *IOP Conference Series: Materials Science and Engineering*, Padang, Indonesia, 846(1), 012046.
16. Saputra, D.; Ma'arif, A.; Maghfiroh, H.; Chotikunnan, P.; and Rahmadhia, S.N. (2023). Design and application of PLC-based speed control for DC motor using PID with identification system and MATLAB tuner. *International Journal of Robotics and Control Systems*, 3(2), 233-244.
17. Ghith, E.S.; and Tolba, F.A.A. (2022). Design and optimization of PID controller using various algorithms for micro-robotics system. *Journal of Robotics and Control*, 3(3), 244-256.
18. Brito, A.G. (2019). On the misunderstanding of the Ziegler Nichols formulae usage. *IEEE/CAA Journal of Automatica Sinica*, 6(1), 142-147.
19. Bharat, S. et al. (2019). A review on tuning methods for PID controller. *Asian Journal of Convergence in Technology*, V(I), 1-4.
20. Pande, U.; and Hote, Y.V. (2020). Real time design of PID controller for process control system. *Proceedings of the International Conference on Smart Technologies in Computing, Electrical and Electronics* (*ICSTCEE*), Bengaluru, India, 278-283.

21. Okelola, M.O.; Aborisade, D.O.; and Adewuyi, P.A. (2021). Performance and configuration analysis of tracking time anti-windup PID controllers. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, 6(2), 20-29.
22. Imani, A.; and Montazeri-Gh, M. (2020). Stability analysis of override logic system containing state feedback regulators and its application to gas turbine engines. *European Journal of Control*, 52, 97-107.
23. Ramalingam, S.; Baskaran, K.; and Kalaiarasan, D. (2019). IoT enabled smart industrial pollution monitoring and control system using raspberry Pi with BLYNK server. *Proceedings of the International Conference on Communication and Electronics Systems* (*ICCES*, 2019), Coimbatore, India, 2030-2034.
24. Sharma, P.; and Parveen, K. (2020). Realization of an IoT system for real-time remote surveillance of multi-sensor network and control of multiple appliances using 'Blynk' cloud server. *International Research Journal of Engineering and Technology*, 7(10), 1367-1372.
25. Media's, E.; Syufrijal; and Rif'an, M. (2019). Internet of Things (IoT): BLYNK framework for smart home. *KnE social sciences*, 3(12), 579-586.
26. Romero-Perigault, J.; Flores-Fuentes, W.; Jo, K-H.; and Hernandez, D.C. (2019). Wireless current monitoring for autonomous robot navigation. *Proceedings of IEEE* 28th *International Symposium on Industrial Electronics* (*ISIE*), Vancouver, BC, Canada, 1717-1722.
27. Mubarok, H.; and Ardiansyah, A. (2020). Prototype design of IoT (Internet of Things)-based load monitoring system. *Proceedings of the* 3rd *International Seminar on Research of Information Technology and Intelligent Systems* (*ISRITI*), Yogyakarta, Indonesia, 377-382.
28. Vignesh, N.; Manoj, M.; Reddy, S.P.; and Reddy, K. (2024). Smart parking system using IOT with blynk and RFID. *International Research Journal on Advanced Engineering Hub*, 2(4), 814-822.
29. Ashraf, A.; Ahsan, H.; and Arshad, M.S. (2021). Motor speed synchronization of mobile robot using PI controller. *Proceedings of the International Conference on Digital Futures and Transformative Technologies* (*ICoDT2*), Islamabad, Pakistan, 1-6.
30. Karrupusamy, P. (2020). A sensor based IoT monitoring system for electrical devices using blynk framework. *Journal of Electronics and Informatics*, 2(3), 182-187.
31. Al-Sheikh, M.A.; and Ameen, I.A. (2020). Design of mobile healthcare monitoring system using IoT technology and cloud computing. *IOP Conference Series: Material Science and Engineering*, 881(1), 1-17.
32. Sugapriya, M. (2020). Heist Tracking and Prevention in ATM Utilizing IOT and Blynk Server. *International Journal of Scientific Research & Engineering Trends*, 6(5), 2907-2911.
33. Sharma, P.; and Kantha, P. (2020). 'Blynk' cloud server based monitoring and control using 'NodeMCU'. *International Research Journal of Engineering and Technology*, 7(10), 1362-1366.
34. Venkateshappa; Chethan, H.; Jayaraj, C.L.; Jatinjayasimha V.N.; and Srihari D.S. (2021). Home automation with blynk and nodemcu. *Turkish Journal of Computer and Mathematics Education*, 12(12), 2669-2674.

35. Hasan, D.; and Ismaeel, A.G. (2020). Designing ECG monitoring healthcare system based on internet of things blynk application. *Journal of Applied Science and Technology Trends*, 1(2), 106-111.
36. Assaf, R.; and Ishaq, I. (2020). Improving irrigation by using a cloud based IoT system. *Proceedings of the International Conference on Promising Electronic Technologies* (*ICPET*), Jerusalem, Palestina, 28-31.
37. Markovic, M.; Maljkovic, M.; and Hasanah, R.N. (2020). Smart home heating control using raspberry Pi and blynk IoT platform. *Proceedings of the* 10^{th} *Electrical Power, Electronics, Communications, Controls and Informatics Seminar* (*EECCIS*, 2020), Malang, Indonesia, 188-192.
38. Etuk, U.E.; Omenaru, G.; Inyang, S.J.; and Umoren, I. (2023). Towards sustainable smart living: cloud-based IoT solutions for home automation. *Journal of Information Systems and Informatics*, 5(4), 1743-1763.
39. Prabu, S. (2023). Blynk 2.0 based smart electricity monitoring meter. *International Journal for Research in Applied Science and Engineering Technology*, 11(1), 1312-1323.
40. Harahap, C.R.; Setyawan, F.X.A.; and Sanjaya, D. (2023). IoT based monitoring and speed control of induction motor using nine level cascaded H-Bridge multilevel inverter. *Proceedings of the International Conference on Converging Technology in Electrical and Information Engineering* (*ICCTEIE*), Bandar Lampung, Indonesia, 16-20.