

PROTOTYPE AIR CONDITIONING MONITORING AND CONTROL SYSTEM FOR SMART CLASSROOM BASED ON THE INTERNET OF THINGS

Samuel Beta¹, Rindang Reynaldi², Gisnaya Faridatul Avisyah³

^{1,2,3} *Telecommunication Engineering, Electrical Engineering Department, Politeknik Negeri Semarang*

Abstract— Electricity wastage is a growing concern, often resulting from human behavior. One common scenario is the continuous operation of a high-powered air conditioner (AC) even when it's unnecessary. Moreover, extended AC usage can accelerate wear and tear, potentially damaging the unit. To address these issues, a remote monitoring and control device based on the Internet of Things (IoT) concept is crucial. This device can connect to smartphones, allowing users to remotely monitor and control AC usage. Through an AC simulator, the system enables monitoring and control via a telegram system linked to the smartphone. Developed using the waterfall method and utilizing the NodeMCU ESP8266 microcontroller, the prototype proves effective in assisting users in monitoring and controlling AC usage. Particularly useful when users are away from the AC location and inadvertently forget to turn it off, the system mitigates unnecessary electricity wastage. By implementing this solution, energy efficiency is promoted, and the lifespan of AC units is extended. With the potential to reduce electricity wastage, this technology contributes to a more sustainable future.

Keywords— Internet of Things, Air Conditioner, Telegram, NodeMCU ESP8266, Arduino Uno.

1. Introduction

Electricity is an important resource in all sectors of human life. However, the limited source of electricity causes humans to be required to do energy savings. Energy efficiency is defined as an effort to save the energy needed. Waste of electricity is one of the important issues to consider. One technology that helps this problem is the Internet of Things or commonly known as IoT.

Internet of Things (IoT) refers to the network of physical objects or devices that are connected to the internet and can communicate and exchange data with each other. These objects can include everyday items such as appliances, vehicles, wearable devices, industrial equipment, and more. The core concept behind IoT is to enable these objects to collect and share data, as well as interact with their environment or other connected devices, to provide new functionalities, automation, and insights. This is achieved through the integration of sensors, actuators, software, and network connectivity into these objects. IoT uses several technologies that are largely combined into one unity including sensors as data readers, internet connections with various network topologies, radio frequency identification (RFID), wireless sensor network (WSN) and technologies that will continue to grow according to the needs [1]. IoT in its application can identify, track, monitor objects, trigger an event and provide notifications automatically and in real time [2]. Therefore, IoT has become a solution to addressing the

main factor of electricity waste, which is human negligence. For example, one of the high-powered electrical devices is the air conditioner. (AC). Sometimes one forgets to turn off the AC when not in use. The AC that is continuously on without being turned off will significantly increase the waste and cost of electricity. In addition, air conditioning that is left to light for a long time will speed up the service life or quickly damage the air conditioner itself. With the use of IoT technology, the AC control system can be made remotely so that users who are not in the location where the AC is can control by turning on or turning off the AC.

The research aims to create prototypes of remote AC control systems using smartphones on telegram applications. The system can be used in the classroom to address the problem of the student's lack of awareness to care about the use of air conditioning and the lack of control of the room officer to check the condition of the air condition. Officers who are outside the classroom can control by turning on or turning off the AC in the classrooms when no one is using it. Then officers can also find out if the AC status is still on or dead simply by checking the status using a smartphone that can be monitored remotely. This results in an efficient use of electricity in the classroom and AC lifetime because it does not light up continuously.

In this study, the authors were inspired and referenced from previous research related to the background of this research, namely research conducted by [3] and [4]. The

research conducted by [3] discussed the integrated control system of electrical equipment and room security based on the Internet of Things using the Arduino Mega 2560 microcontroller. The system is built to detect AC flash status using the main optocoupler component which is a photo transistor which is mounted parallel to the LED indicator on the AC. AC uses relay drives that are installed parallel to the manual switches on the AC. When the Arduino Mega 2560 output pin connected to the relay drive input is activated, the relay contact will move to normally open (NO) for a moment, which causes the AC to turn off when the AC is on and the AC will turn on when the Air is off. Arduino connects to the Internet using the W5100 Ethernet Shield module so it requires a RJ-45 cable. The system uses Blynk as an interface app on the smartphone.

The company developed a prototype system for remote control of lights and AC with an Internet network using the NodeMCU-based telegram application ESP8266. [4] The system is built using the Internet of Things based on NodeMCU ESP8266 to do by connecting AC to the relay and electric power, then open the telegram application and log in to the already created telegram bot. If you send a command to turn on the AC, then the AC will light up, and when you send the deadly AC command, then AC will off.

From previous research it can be concluded that the use of the Arduino Mega 2560 as in [3] is not necessary in this study because of the too many spins and the more expensive price. The NodeMCU ESP8266 microcontroller feels suitable for use. Then on [4] the system was made only for remote control while the system should be made without eliminating the manual way to turn off and turn on the AC i.e. pressing the AC button or using a remote AC. Further to the study, a PIR sensor was added to tell officers if there was any activity in the classroom with notifications every 30 minutes using a telegram app. The use of the telegram app is also used to see the AC flash status directly as well as shut down and turn on the AC. The prototype used an AC simulator system, which is a series of input controls on the Arduino UNO with a push button to turn on and turn off the LED.

The difference between the research carried out and the previous research lies in the system created. In previous studies, the system could only be monitored and controlled using a telegram bot with no manual control. If there is an error in the system, the AC cannot be turned on. This system can be controlled not only automatically but also manually.

2. Method

A. Method of Research

In this study, the chosen methodology for the development process is the widely recognized and structured waterfall method [5]. This approach consists of several distinct stages that progress sequentially, ensuring a systematic and well-defined process for creating the intended system. Each stage serves a specific purpose and contributes to the overall success of the project.

a. Requirement Analysis

The first stage involves a thorough examination of the data and information relevant to the system being developed. Through careful analysis and data collection, the specific needs and requirements of the system are identified and documented.

b. Design of System

Moving on to the design stage, the focus shifts to studying the requirements established in the previous phase. In this stage, in-depth analysis and detailed planning are conducted to prepare the system design. This phase aids in determining the necessary hardware components and plays a vital role in defining the overall system architecture.

c. Implementation

Next comes the implementation stage, where the actual development of applications takes place. Skilled programmers utilize specific programming languages to write application code that aligns with the previously analyzed requirements. This phase is crucial in translating the design into functional software components.

d. Verification

Once the implementation is complete, the verification stage takes center stage. Rigorous testing and system integration activities are conducted to ensure that the developed systems adhere to the specified specifications. Any discrepancies or errors detected during this phase are thoroughly addressed and rectified before proceeding further.

e. Maintenance

Lastly, the maintenance stage covers system installation and testing, ensuring the smooth operation and performance of the developed system. It encompasses ongoing support, periodic maintenance, and the implementation of necessary system enhancements. This phase ensures the longevity and optimal functioning of the system.

By employing the waterfall method, this study ensures a systematic and well-structured approach that follows a logical progression from requirement analysis to system design, implementation, verification, and maintenance. This adherence to the waterfall method provides a robust framework for the research endeavor, guiding the project toward successful outcomes. The stages outlined above align with the waterfall method as illustrated in Figure 1, reinforcing the systematic and rigorous nature of the study.

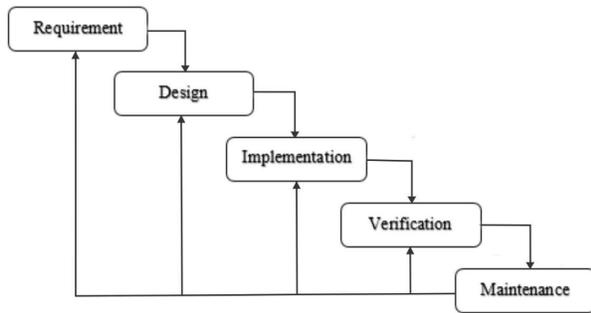


Figure 1 Phase of Research

B. Requirements

Several components, device, and software are required for this research, including: NodeMCU ESP8266, PIR Module, LDR Module, Relay 5V, AC Simulator, Arduino IDE, and Telegram Bot.

a. NodeMCU ESP8266

The NodeMCU ESP8266 is an open-source development board that combines the ESP8266 Wi-Fi module with a microcontroller unit (MCU). It provides an easy and affordable way to add Wi-Fi connectivity to your projects. The ESP8266 module is based on the ESP8266EX system-on-a-chip (SoC), which includes a Wi-Fi radio and a microcontroller with GPIO pins.

The NodeMCU ESP8266 board is popular among hobbyists, makers, and IoT enthusiasts due to its low cost, ease of use, and versatility. It has been widely adopted for projects ranging from simple home automation to complex IoT solutions.

The ESP8266 is a microcontroller developed by Espressif Systems. It serves as an all-in-one WiFi networking solution, enabling existing microcontrollers to connect to WiFi and run independent applications. Equipped with a built-in USB connector and a wide selection of pin-outs, the module can be easily connected to a laptop using a micro USB cable for convenient flashing, similar to Arduino. Furthermore, it is designed to be compatible with breadboards right away, facilitating rapid prototyping. Figure 2 illustrates the NodeMCU ESP8266 device and the pin definitions are displayed in Figure 3.

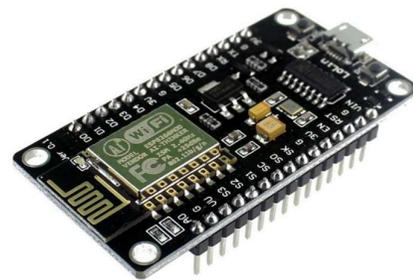


Figure 2 NodeMCU ESP8266

Specification:

- Voltage: 3.3V.
- Wi-Fi Direct (P2P), soft-AP.
- Current consumption: 10uA~170mA.
- Flash memory attachable: 16MB max (512K normal).
- Integrated TCP/IP protocol stack.
- Processor: Tensilica L106 32-bit.
- Processor speed: 80~160MHz.
- RAM: 32K + 80K.
- GPIOs: 17 (multiplexed with other functions).
- Analog to Digital: 1 input with 1024 step resolution.
- +19.5dBm output power in 802.11b mode.
- 802.11 support: b/g/n.
- Maximum concurrent TCP connections: 5.

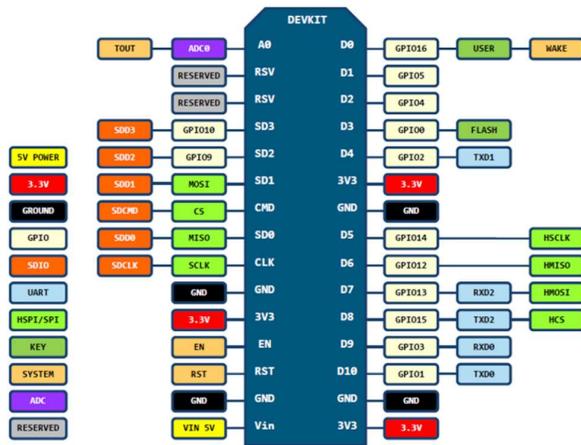


Figure 3 NodeMCU ESP8266 pin definition

Some important points to keep in mind:

- GPIO0 (D3) and GPIO2 (D4) have internal pull-up resistors, which can affect the boot process of the NodeMCU board. GPIO0 is also used to put the board into programming mode (flash mode) by pulling it low during startup.
- GPIO1 (TX) and GPIO3 (RX) are connected to the built-in UART interface, which is useful for serial communication with other devices.
- GPIO6 (D0) to GPIO11 (D5) are sometimes referred to as the "SDIO" pins, and they have additional functions related to the SPI flash memory on the ESP8266 module.
- GPIO16 (D14) is often used as the wake-up pin from deep sleep mode.

b. PIR Module

A PIR (Passive Infrared) module, also known as a PIR sensor or motion sensor, is a device that detects infrared radiation emitted by objects in its field of view. It is commonly used for motion detection applications, such as security systems, automatic lighting, and energy-saving applications.

PIR modules are easy to use and can be interfaced with various microcontrollers, development boards, or other electronic devices. By combining PIR modules with other components and programming logic, you can create motion-activated systems for a wide range of applications.

PIR sensors provide the capability to detect motion, typically used to determine if a person has entered or exited the sensor's range. These sensors are compact, affordable, energy-efficient, user-friendly, and have a long lifespan. Due to these advantages, they are frequently employed in household and commercial devices. They are commonly known as PIR sensors, which stands for "Passive Infrared," "Pyroelectric," or "IR motion" sensors. The PIR module and its component definitions can be seen in Figure 4.

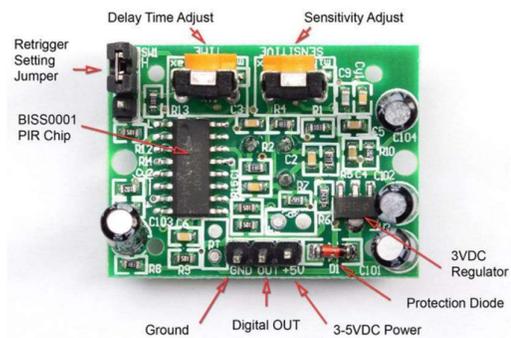


Figure 4 PIR Module

Specification:

- Output: The sensor provides a digital pulse that is high (3V) when triggered (motion detected) and digital low when idle (no motion detected). The length of the pulses is determined by resistors and capacitors on the PCB and can vary between different sensor models.
- Sensitivity range: The sensor can detect motion within a range of up to 20 feet (6 meters) with a detection angle of 110° x 70°.
- Power supply: Most modules can accept input voltages ranging from 5V to 12V (they have a built-in 3.3V regulator), but it is recommended to use a 5V power supply to ensure compatibility with the regulator's specifications.

c. LDR Module

An LDR (Light Dependent Resistor) module, also known as a photoresistor module, is a sensor module that detects the intensity of light in its surroundings. It is commonly used in applications such as automatic lighting control, burglar alarm systems, and light-sensitive robots.

LDR modules are relatively simple to use and can be integrated into various electronic circuits. By monitoring the output voltage of an LDR module, you can control lighting systems, adjust display brightness, or activate/deactivate devices based on the ambient light conditions.

The LDR sensor module in Figure 5 is an affordable module that functions as a digital sensor. It is designed to measure and detect light intensity and is commonly referred to as a Photoresistor sensor. The module is equipped with an LDR (Light Dependent Resistor) that enables it to sense light. It has three terminals, with the "DO" pin serving as a digital output. The module produces a high output when there is no light and a low output when there is light present. The sensitivity of the sensor can be adjusted using the built-in potentiometer on the module.

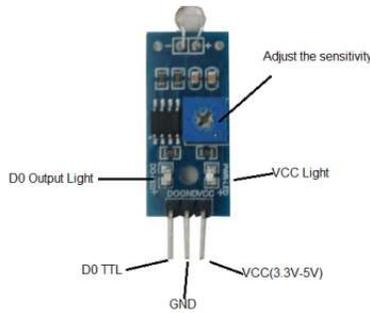


Figure 5 LDR Module

Specification:

- Operating voltage: The LDR sensor module can operate at either 5V or 3.3V DC.
- Comparator chip: The module utilizes the LM393 comparator chip.
- Module pins: It has 3 pins for connections.
- Output type: The sensor provides digital outputs (D0).
- Sensitivity: The sensitivity of the sensor is adjustable.
- Indicator LED: The module includes an indicator LED for output and power indication.
- PCB size: The size of the printed circuit board (PCB) is 3cm * 1.6cm.
- Fixed Hole Diameter: The module features fixed holes with a diameter of 3mm.

d. Relay 5V

A 5V relay is an electronic switch that can be controlled by a 5V signal. It is commonly used to control high-power devices or circuits using a low-power control signal from a microcontroller, Arduino board, or other electronic devices. When the 5V signal is applied to the relay coil, it energizes the coil and causes the switch contacts to change state. This, in turn, controls the connected high-power device or circuit.

5V 1-channel relay interface board designed for low-level signals requires a driver current of 15-20mA. The board is suitable for controlling appliances and equipment with high currents. It features high-current relays capable of operating under AC250V 10A or DC30V 10A. The interface is standard and allows direct control by a microcontroller. For safety purposes and to prevent ground loops when connected to a microcontroller, the module is optically

isolated from the high voltage side. Figure 6 shows a 5V single-channel relay.

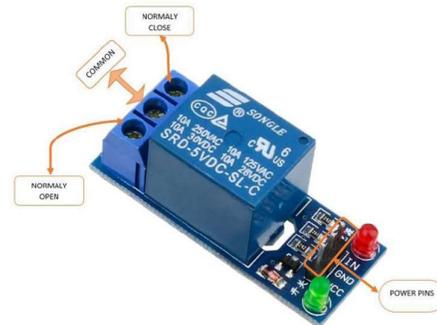


Figure 6 1-Channel 5V Relay Module

Specification:

- Operating Voltage: 5Vdc.
- Maximum Relay Output: DC 30V/10A, AC 250V/10A.
- 1 Channel Relay Module with Opto-coupler: This module serves as a LOW Level Trigger expansion board, compatible with Arduino control boards.
- Standard interface: The module can be directly controlled by various microcontrollers such as 8051, AVR, PIC, DSP, ARM, MSP430, TTL logic.
- High-quality Relay: The module features high-quality, low-noise SPDT relays with a common terminal, a normally open terminal, and a normally closed terminal.
- Opto-Coupler Isolation: The module utilizes opto-coupler isolation to ensure high voltage safety and prevent ground loops when interfacing with a microcontroller.

e. Arduino IDE

Arduino IDE (Integrated Development Environment) is an open-source software platform used for programming Arduino boards. It provides a user-friendly interface for writing, compiling, and uploading code to Arduino microcontrollers. The Arduino IDE is available for Windows, Mac OS X, and Linux operating systems.

The Arduino IDE is widely used by beginners and experienced users alike due to its simplicity and ease of use. However, alternative development environments like PlatformIO and Visual Studio Code with Arduino extensions are also available, offering additional features and flexibility for advanced users.

The Arduino Integrated Development Environment, also known as the Arduino Software (IDE), comprises a code editor, a message area, a text console, a toolbar with commonly used function buttons, and a set of menus. It establishes a connection with Arduino hardware to facilitate program uploading and communication with the devices.

```

23 int sensor_ledStatus;
24 unsigned long aCStarted = 0;
25 const long interval = 30000;
26 unsigned long bot_lastTime; // last time messages' scan has been done
27 const unsigned long BOT_TIME = 3000; // when time between scan messages
28 //=====
29
30 void setup() {
31   Serial.begin(115200);
32   pinMode(PIR_Sensor, INPUT);
33   pinMode(LED, OUTPUT); // LED pin as output.
34   pinMode(LDR_Sensor, INPUT);
35   WiFi.mode(WIFI_STA);
36   WiFi.begin(ssid, password);
37   delay(2000);
38   Serial.println("Connecting WiFi: ");
39   Serial.println(ssid);
40   WiFi.begin(ssid, password);
41   while (WiFi.status() != WL_CONNECTED) {
42     Serial.print(".");
43     delay(500);
44   }
45   Serial.println("");
46   Serial.println("WiFi connected");
47   Serial.println("IP address: ");
48   IPAddress ip = WiFi.localIP();
49   Serial.println(ip);
50   IPAddress ipAddress = ip.toString();
51   Client myClient(ipAddress);
52   digitalWrite(LED, HIGH); // turn the LED on
53   ledStatus = digitalRead(LDR_Sensor);

```

Figure 7 Programming using Arduino IDE

Sketches are the term used for programs written using the Arduino Software (IDE). These sketches are created in the IDE's text editor and are saved with the .ino file extension. Within the editor, you can utilize features like cut/paste and search/replace for editing the code. The message area provides feedback during tasks like saving and exporting, and it also displays any encountered errors. The console section shows the textual output from the Arduino Software (IDE), including detailed error messages and other relevant information. In the bottom right corner of the window, you can see the selected board and serial port configurations. The toolbar contains buttons that enable you to verify and upload programs, create, open, and save sketches, and access the serial monitor. Figure 7 is an example of programming Arduino using the Arduino IDE.

f. Telegram Bot

This research utilizes the Arduino library called Universal Bot Library. It is an Arduino IDE library designed specifically for using the Telegram Bot API. The library is compatible with multiple Arduino architectures and provides an interface for interacting with the Telegram Bot API.

Telegram is an instant messaging service that enables the creation of bots. Bots are compact applications that operate entirely within the Telegram app. Users engage with bots through versatile interfaces capable of accommodating various tasks or services. Users have the ability to send bot messages of different types, encompassing text, files, locations, stickers, voice messages, and even dice if they are feeling fortunate. Nevertheless, Telegram bots provide a wide array of additional tools to construct adaptable interfaces customized to specific requirements, including:

- o Commands: These are highlighted in messages and can be chosen from a list after typing "/".
- o Keyboards: They replace the user's keyboard with predefined options for answers.
- o Buttons: These are displayed alongside messages from the bot.

Bots can be configured to send and receive messages. This feature is beneficial for Arduino projects as it allows you to

receive updates from your project or send commands to it through the Telegram app, regardless of your location.

g. AC Simulator

This research utilizes Arduino Uno, a switch, and an LED as a simulator for an air conditioner (AC). The Arduino is programmed to control the on and off functionality of the LED, simulating the operation of turning on and off an AC unit.

C. Design of System

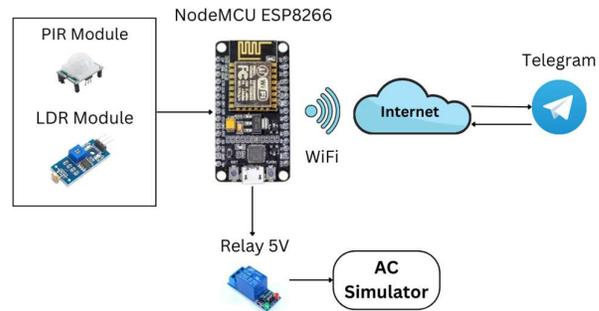


Figure 8 Architectural System

The monitoring and control system of the air conditioner simulator employs the NodeMCU ESP8266 microcontroller as the central controller. Within the AC simulator, a set of light and dark detectors, including an LDR positioned in front of the LED indicator, is utilized. Based on the circuit configuration, the circuit's output is set to a logical '0' when the AC LED indicator is on and a logical '1' when the LED is off. To detect activity within the simulated environment, a PIR module is utilized at 30-minute intervals. The microcontroller interfaces with a relay module to simulate the AC turning on or off. Figure 8 illustrates the architecture of the system.

The system starts when the device is connected to the voltage source and then awaits to connect to the internet. Once connected to the Internet, the system will initialize the telegram application on Android. Then when the system is still in activity mode, the system will always read the commands given. The schematic network can be seen in Figure 9.

The series consists of NodeMCU ESP8266, the LDR module, the PIR sensor, the relay module and the AC simulator. The AC simulator in this study is a series of input controls on the Arduino UNO with a push button to turn on and off the LED. The PIR sensor and the LDR module become inputs in sequence on the D5 and D1 pins then the relay module becomes the output on the NodeMCU ESP8266 microcontroller D6 pins. The VCC pin on the PIR sensor is connected with the 3V NodeMCU ESP8266 pin while the relay module uses an external power of 5V. Further for the AC simulator use the LED as the output on the D6

pin and the push button as the input on the d2 pin that is paralleled with the NO pin on the relay module so that the initial position condition is open or the electric current is cut off. The entire GND pin is connected to the ground.

The program flow diagram can be seen in Figure 10. The AC monitoring and control system is made using the telegram app on the smartphone to turn on and off the AC. In addition, on the telegram application can monitor the condition of the air condition in the state of lighting or off. On the smartphone can monitor the current state of the AC which then when to turn off the AC from a distance then the smartphone sends a command that then sent to NodeMCU. After that, NodeMCU will give a trigger to the relay to turn off the AC. If the data integer is 1 then NodeMCU will give a trigger to the relay to turn on the AC, whereas when the data Integer is 0 then NODEMCU will provide a triggers to the relay to turn off the AC. But in addition to being controlled using the Android app, AC can also be controlled manually in the classroom.

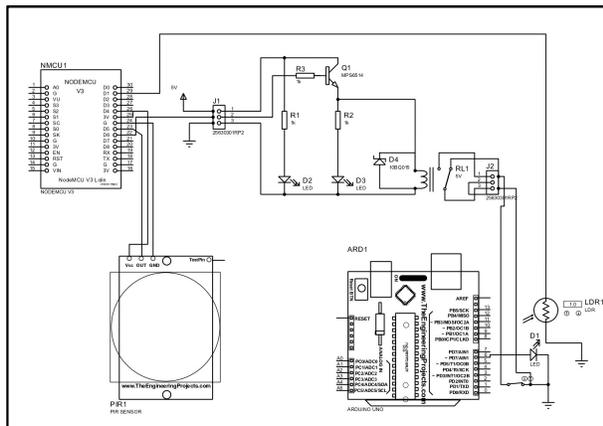


Figure 9 Circuit Schematic Diagram

D. Research

In the production of this paper, the testing of the device was carried out in the State Polytechnic Electronics Laboratory Semarang. The parameters tested included testing the failure percentage on the control function using the buttons on the smartphone and testing the response delay on the device.

The percentage of failure in testing the control function with the smartphone button can be calculated by the equation (1).

$$Error (\%) = (\Sigma Error / \Sigma examination) \times 100\% \tag{1}$$

While the response delay on the control system can be calculated by the equation (2).

$$Delay (ms) = t_{order} - t_{response} \tag{2}$$

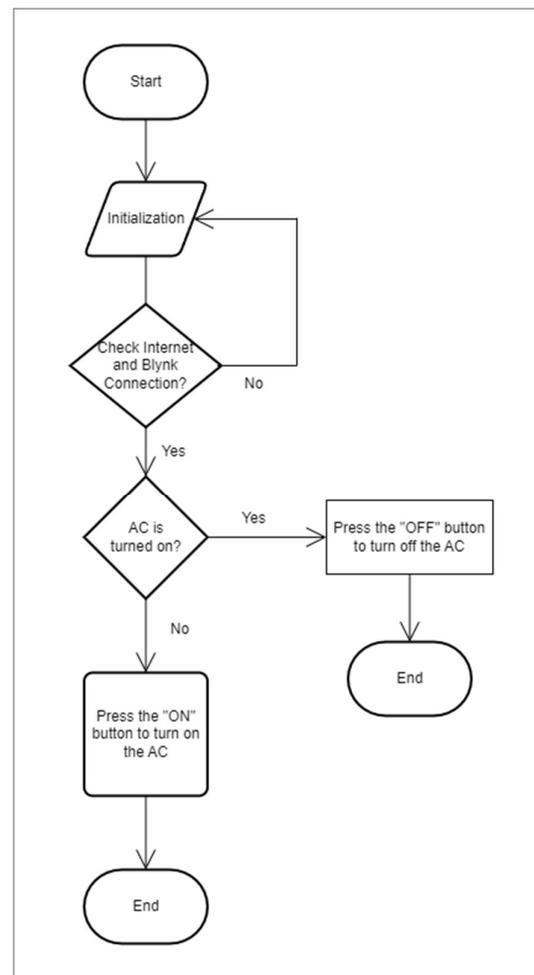


Figure 10 Flowchart

3. Result and discussion

After the system that has been created can be completed and without any problems, then the next step is the trial process. Trials are carried out repeatedly to get the appropriate results and test whether the system can run properly. System testing is done manually by pressing the On Off button on the simulator and automatically by using the telegram bot application.

The first scenario test was carried out by taking a sample from the manual delivery of the switch button commands to the BuiltIn LED on the Arduino Uno. This test is performed ten times by pressing the switch button alternately. This test has a 100% success rate. This can be seen from the data collection table in Table 1. The result of the graph is the percentage success of the first scenario control in Figure 11.

Table 1 The First Scenario Data

Test	Press The Button	Feedback LED
1	1	ON
2	2	OFF
3	3	ON
4	4	OFF
5	5	ON
6	6	OFF
7	7	ON
8	8	OFF
9	9	ON
10	10	OFF

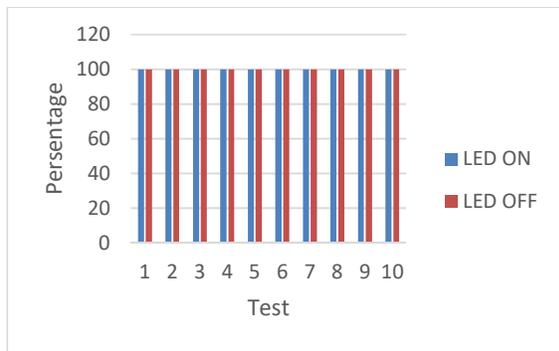


Figure 11 The First Graphic Scenario

The second scenario test is performed by taking a sample from the delivery of orders on a bot telegram. This test was performed ten times sending commands on as many as five times and off as many than five times. This test has a 100% success rate. This can be seen from the data collection table in Table 2. The result of the graph is the percentage success of the second scenario control in Figure 12.

Table 2 The Second Scenario Data

Test	Giving Commands	Feedback LED
1	/ONOFF	ON
2	/ONOFF	OFF
3	/ONOFF	ON
4	/ONOFF	OFF
5	/ONOFF	ON
6	/ONOFF	OFF
7	/ONOFF	ON
8	/ONOFF	OFF
9	/ONOFF	ON
10	/ONOFF	OFF

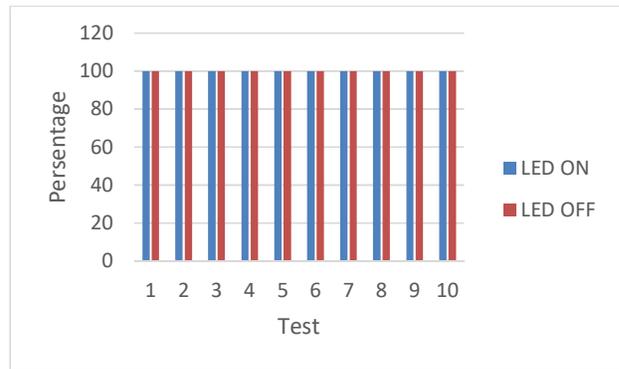


Figure 12 The Second Graphic Scenario

The third scenario test is carried out by taking a sample from the delivery of orders on a bot telegram. This test was performed five times by checking the movement in front of the PIR sensor. This test has a 100% success rate. This can be seen from the data collection table in Table 3. The result of the graph is the percentage success of the second scenario control in Figure 13.

Table 3 The Third Scenario Data

Test	Movement	Feedback Telegram
1	1	Active
2	2	Active
3	3	Active
4	4	Active
5	5	Active

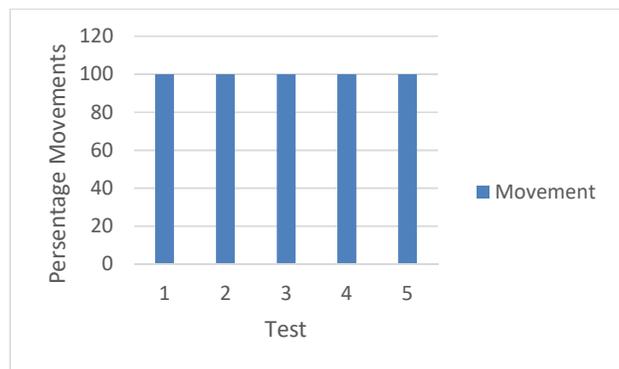


Figure 13 The Third Graphic Scenario

The fourth scenario test is carried out by calculating the delay of the feedback result sent using a way to calculate the time difference at the time the order was sent. The test was performed five times on each feedback from the output of the lighting delivery, the outcome of the deadly delivery of

the lamp and the outcomes of the motion of the sensor on the telegram. This can be seen from the data collection table in Table 4. The result of the graph is the percentage success of the control of the fourth scenario in Figure 14. On the delivery of telegram orders using 4G cellular data existing on the smartphone.

Table 4 The Fourth Data Scenario

Test	Commands	Delay (s)
1	/ONOFF	1
2	/ONOFF	1,4
3	/ONOFF	0,6
4	/ONOFF	0,7
5	/ONOFF	1,5
6	/ONOFF	0,4
7	/ONOFF	0,5
8	/ONOFF	0,2
9	/ONOFF	1,4
10	/ONOFF	0,8
11	Sensor Movements	2,3
12	Sensor Movements	1,8
13	Sensor Movements	1,3
14	Sensor Movements	2
15	Sensor Movements	1,7

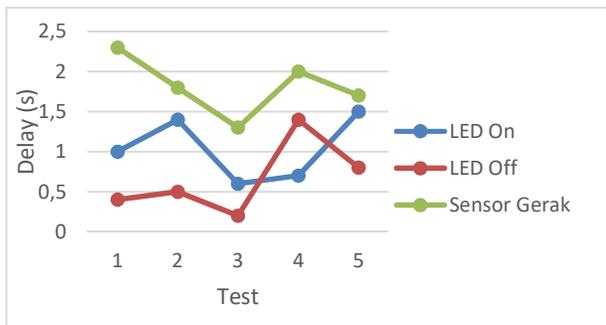


Figure 14 Results of The Fourth Scenario

The results of the design of the telegram bot application used on the system are in accordance with Figure 15. In the telegram bot, it is made using a simple command. This is because it makes it easier for users to use it.

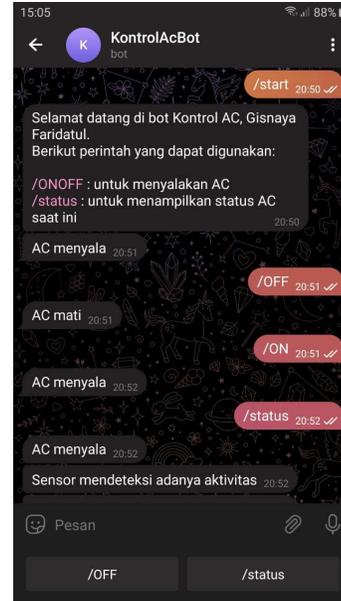


Figure 15 Telegram Bot Application Mockup

In the telegram bot application, you can monitor and control the air conditioner in the smart classroom. This is useful to know the status of the AC is on and not in the room. In addition, the application can detect movement in the room so that it can find out whether the classroom is being used or not.

The telegram bot command can be started with /start then it will get a response in the form of an opening greeting and a guide for turning on and off the air conditioner. The command to find out the status of the air conditioner in the room is on or off by pressing the command /status. In this command if the AC in the classroom is on, then the user will get the response 'AC menyala' and vice versa. The next command is in the form of /ONOFF which is a command to turn on or turn off the air conditioner. if the AC condition is actually on, then after pressing the command /ONOFF it will respond 'AC menyala' and vice versa.

On the system that has been made will detect motion if there is movement. The PIR sensor will detect movement when movement has occurred. Then the motion sensor will send information to Arduino which will then provide information to the telegram bot in the form of the command 'Sensor mendeteksi adanya aktivitas'.

The system that has been created has advantages compared to other systems. The advantage of this system is that it can be done in two ways, namely automatically and manually. So that users can monitor and control AC remotely and in real time using telegram bots. But users can also monitor and control the AC manually in the classroom.

4. Conclusion

Functionally, the system that has been built can respond to commands that have been placed on a bot telegram with a 100% success rate. However, status feedback sent and received to users is delayed because a stable internet is one of the indicators of delay or non-delay in the delivery of orders and the recipient of the orders. The system that has been created can be useful for monitoring and controlling the air conditioner installed in the smart classroom. Each user can monitor and control using the Telegram bot according to what has been made. This can aim to reduce the level of waste of electrical energy that is wasted when the classroom is not in use. Because the system has motion sensors that can be monitored via telegram bot. The system that has been created has the advantage that it can be monitored and controlled automatically and manually. Automatically via telegram bot and manually in the classroom. Besides that the system has not been proven in its energy saving. Further research on this system can be carried out directly installed on the smart classroom and directly on the AC without using a simulator.

References

- [1] A. Junaidi, "Internet Of Things, Sejarah, Teknologi Dan Penerapannya : Review," *J. Ilm. Teknol. Inf.*, vol. IV, no. 3, pp. 62–66, 2015.
- [2] Q. Zhou and J. Zhang, "Internet of things and geography review and prospect," *Proc. - 2011 Int. Conf. Multimed. Signal Process. C. 2011*, vol. 2, pp. 47–51, 2011, doi: 10.1109/CMSP.2011.101.
- [3] A. D. Kurniasari, R. A. Fatansyah, and S. B. Kuntardjo, "SISTEM KONTROL PERALATAN LISTRIK DAN KEAMANAN RUANGAN TERPADU BERBASIS INTERNET OF THINGS".
- [4] A. M. Ibrahim and D. Setiyadi, "PROTOTYPE PENGENDALIAN LAMPU DAN AC JARAK JAUH DENGAN JARINGAN INTERNET MENGGUNAKAN APLIKASI TELEGRAM BERBASIS NODEMCU ESP8266," vol. 7, no. 1, pp. 27–34, 2021.
- [5] N. Triwahyuni and S. Beta, "Running Text Information System Design Internet-Based for Small Outlets," *J. Appl. Inf. Commun. Technol.*, vol. 7, no. 2, p. 2022, 2022.
- [6] F. Rahmasari, "Integrasi sistem pemantauan dan pengendalian smart agriculture dan smart farm berbasis android," 2020.
- [7] M. I. Kurniawan, U. Sunarya, and R. Tulloh, "Internet of Things : Sistem Keamanan Rumah berbasis Raspberry Pi dan Telegram Messenger," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 6, no. 1, p. 1, 2018, doi: 10.26760/elkomika.v6i1.1.
- [8] R. Rahmatina, M. N. Aripin, M. Iqbal, and A. Deolika, "Implementasi Transistor BD139 dan Rangkaian Relay pada Mesin Air," vol. 3, no. 1, pp. 11–18, 2023.
- [9] P. W. Rusimamto, Endryansyah, and M. S. Zuhrie, "Perancangan Dan Implementasi Fuzzy Logic Control Untuk Pengaturan Kestabilan Gerak Pada Two Wheels Self Balancing," *J. Tek. Elektro*, no. Vol 9 No 3 (2020): SEPTEMBER 2020, pp. 717–723, 2020, [Online]. Available: <https://ejournal.unesa.ac.id/index.php/JTE/article/view/39158/34330>
- [10] M. U. Fahri, "Implementasi Channel Bot Telegram (Real Time) COVID-19 di Kalimantan Barat dengan Memanfaatkan API," *J. Ilmu Komput. dan Desain Komun. Vis.*, vol. 5, no. 2, pp. 77–84, 2020.