OPEN ACCESS
Saudi Journal of Engineering and Technology
Abbreviated Key Title: Saudi J Eng Technol
ISSN 2415-6272 (Print) |ISSN 2415-6264 (Online)
Scholars Middle East Publishers, Dubai, United Arab Emirates
Journal homepage: https://saudijournals.com

Review Article

Web-Based Air Conditioner Monitoring and Control

Theresia Ghozali[1*], Gabriella Ilena[2], Melisa Mulyadi[3], Linda Wijayanti[4]

[1]Lecturer, Telecommunications Research Group, Atma Jaya Catholic University of Indonesia, Indonesia
[2]Student, University of Stuttgart, Germany
[3]Lecturer, Telecommunications Research Group, Atma Jaya Catholic University of Indonesia, Indonesia
[4]Lecturer, Telecommunications Research Group, Atma Jaya Catholic University of Indonesia, Indonesia

DOI: 10.36348/sjet.2022.v07i08.006 | **Received:** 05.08.2022 | **Accepted:** 31.08.2022 | **Published:** 20.09.2022

***Corresponding author:** Theresia Ghozali
Lecturer, Telecommunications Research Group, Atma Jaya Catholic University of Indonesia, Indonesia

Abstract

Through advances of Internet-of-Things (IoT) technologies, smart home concepts have become increasingly prevalent, and they are gradually being adapted in society. Apart from cost efficiency and user convenience, implementing monitoring and control in a smart home has the potential to also increase energy efficiency. In this paper, a web-based monitoring and temperature control for an air conditioning (AC) system is proposed. The long-ranged AC control system consists of three ZigBee devices that enable wireless communication between the temperature, remote, and router modules which sends data over the internet using ESP8266. A web-based interface was also developed to display room temperature, last temperature of the AC, and a button to activate/deactivate the AC. The results showed that the prototypical system fulfilled the expected functionality, and that ZigBee is a robust choice of communication protocol.
Keywords: ZigBee, Website, ESP8266, IoT, Smartphone.

1. INTRODUCTION

In recent years, advancements of Internet technologies have been paving paths to increasing consumer adaptability for smart homes. The growth in consumer interest is mainly enabled by the more widespread use of networks and smartphones as well as reduced costs of sensors. A smart home is characterized by an intelligent environment, in which many activities are automated. The support of automation in providing service to the home inhabitants is perceived to increase user convenience, as well as enable energy monitoring and efficiency management services [1]. In particular, the monitoring and control of heating, ventilation, and air-conditioning system (HVAC) plays a central role in managing the energy usage.

Previously, Sastry and Kumar in [2] developed an infrared (IR) remote AC control with NodeMCU as the interfacing device with the Internet to send data over WiFi and serial communication between the NodeMCU and IR remote controller engine. David et al. also proposed a home automation device with Arduino in [3], which implements control and monitoring for various activities within a home, including the AC system, using the Bluetooth technology for sending sensor data to the Internet. Looking at the existing wireless technologies, there is potential improvement in implementing another communication protocol that has a higher range as WiFi but with lower cost and power, similar with Bluetooth. With this consideration, the ZigBee network protocol becomes a viable option. ZigBee covers a range of ten to a hundred meters and has relatively lower power and cost compared to Bluetooth and WiFi, based on Ohwo's survey of emerging wireless technologies [4].

In this paper, a web-based air conditioner (AC) monitoring and control prototype is proposed. The AC remote control system utilized the ZigBee protocol, with Arduino as the microcontroller unit and ESP8266 as the primary component to establish connection to the Internet. A graphical user interface (GUI) accessible from the user's smartphone was also developed to enable convenient long-range AC control, from which the user may activate or deactivate the AC and observe temperature values over a period.

The main requirements to be fulfilled by the proposed system were defined as the following:

a. Reliable communication and data transmission should be established between system modules

b. The user should be able to send control commands for the AC through the GUI
c. The system should respond to the commands by changing the AC temperature with corresponding trigger activation of the AC remote control
d. The response time to the command should not exceed 1 second for expected immediate system response [5].

Speed tests and range tests were done to evaluate the system's performance, as well as functional testing. The test results showed that the system covered a range of 150 meters with an average control latency of 1.44 to 5.72 seconds, depending on the available internet connection.

In Section 2, the hardware and software components used in developing the prototype are listed and their specifications are briefly defined. Section 3 encompasses the approach to designing the system, where the design of the hardware and software components of the system are further detailed. Section 4 explains the testing protocols used to evaluate the system's functionality, performance, and limitations. In Section 5, the implementation of the website interface is discussed. Finally, the takeaways and elaboration of potential future work for the project is detailed in Section 6.

2. COMPONENTS
2.1 Hardware
2.1.1 ZigBee
ZigBee is a network protocol that was developed by the ZigBee alliance which comprises of several well-known member companies, ranging from semiconductor and software developers to original equipment manufacturers [6]. The ZigBee protocol refers to the IEEE 802.15.4 standard and is characterized by its suitability for low-power, low-cost devices in wireless network technology.

Typically, ZigBee is implemented in smaller scale networks such as in a wireless personal area network (WPAN), where low-powered devices that do not require high bandwidths are often used. In such cases, ZigBee is often preferred over Bluetooth or WiFi [7]. A comparison between the wireless network technologies is given in Table 1.

Table 1: Comparison between ZigBee, WiFi, and Bluetooth technology based on [7]

	ZigBee	WiFi	Bluetooth
Standard	IEEE 802.15.4	IEEE 802.11	IEEE 802.15.1
Frequency range	2.4 GHz	2.4 GHz and 5 GHz	2.4 GHz to 2.483 GHz
Power consumption	Low	High	Low
Bandwidth	Low	High	Low
Signal range	291 meters	50 meters	77 meters

2.1.2 Arduino UNO
Arduino is an open-source electronic platform which contains an AVR microcontroller chip manufactured by Atmel. A microcontroller is a programmable chip or integrated circuit (IC) that receives input and processes it to deliver desired output. The Arduino board was previously designed mainly for learning, thereby introducing simple, inexpensive programming and control interface for electronic devices. It supports a wide variety of functions, including sending and receiving data over the internet. Arduino UNO is one of the more widely used variants in the Arduino product line, which has the ATMega328 microcontroller chip and works at 16 MHz clock speed [8].

2.1.3 ESP8266
The ESP8266 module is a system-on-chip that provides an access point for internet connection. It is designed to be compatible for use in Internet-of-Things (IoT) applications. ESP8266 is a low-power and low-cost device that is compliant with the IEEE 802.11 protocol and supports the TCP/IP protocol. This module can function as a client, an access point, or assume both roles at the same time [9].

2.1.4 DHT11 Temperature Sensor
The DHT11 sensor detects temperature and humidity by outputting analog voltage corresponding to changes in the temperature and humidity values. This sensor module is categorized as a resistive element. The advantage of this sensor module is its ability to detect temperature and humidity more responsively with less interference, and readings from this sensor are relatively accurate.

2.2 Software
The website interface was developed using JavaScript for the backend and HTML as well as CSS for the front-end. Sensor data were also stored in a MySQL database which allowed periodical monitoring of the temperature and humidity.

3. DESIGN
3.1 System Design
The remote AC control system received commands from the user via the web interface. The user had access to the interface through a smartphone which was connected to the internet. Information such as the room temperature was shown on the website, along with buttons to activate/deactivate and increase/decrease the AC temperature. These buttons were corresponding representations of buttons on an

actual, physical AC remote control device. The proposed design of the system can be described as consisting of three modules, each focusing on a certain task. The system design concept is illustrated in the block diagram shown on Figure 1.

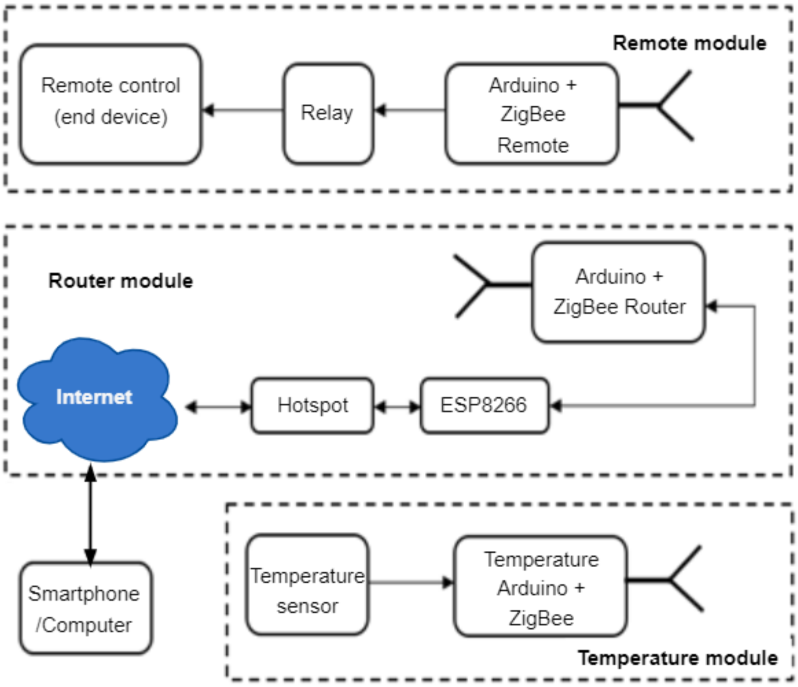

Figure 1: System block diagram

There were three ZigBee modules in the entire system, with one ZigBee placed in each module. The ZigBee modules come from the XB24C product family with type S2C. It was also necessary to ensure that each ZigBee had the same firmware version so that communication between them was possible. Each ZigBee module was identified with a certain address, in this project the addresses were the following:

- Temperature module ZigBee: 13A20041673403
- Router module ZigBee: 13A2004154F50F
- Remote module ZigBee: 13A2004167340D

The XCTU software, which is a free software provided by the developers of XBee to set-up, configure, and test ZigBee devices, was used as means to configure the ZigBee modules. All ZigBee modules had the same 9600 baud rate and 8 data bits configuration, with no parity. Communication between each ZigBee was defined by setting the destination address; the temperature ZigBee pointed to the router ZigBee, and the router ZigBee pointed to the remote ZigBee. Join notification was enabled for the router ZigBee, allowing two-way communication in the router module. Since the three modules were placed in the same personal area network (PAN), the same unique PAN ID were defined for all the ZigBee modules. In this project, the PAN ID 1234 was used to identify the network.

3.1.1 Temperature module

This module was responsible for the data acquisition, taking the sensor readings and processing them before sending the measurements to the communication (router) module and finally to the end device, which was the AC remote control. The room temperature was monitored using a 3-pin DHT11 sensor. The sensor readings were then processed by the Arduino and were sent through the connected ZigBee module to the ZigBee router. The schematics of this module and a simple flowchart are shown in Figure 2 and Figure 3, respectively.

3.1.2 Router module

The router module consisted of another pair of Arduino and ZigBee which oversaw the communication traffic between the temperature module and the remote module, which primarily showed the data to the user on the website via the internet and sent commands to the AC remote control. The Arduino and ZigBee pair were connected to an ESP8266 module to enable connection to the internet, where packets of data were being sent to and stored into the MySQL database in the backend.

The schematics of this module and the respective flowchart are shown in Figure 4 and Figure 5.

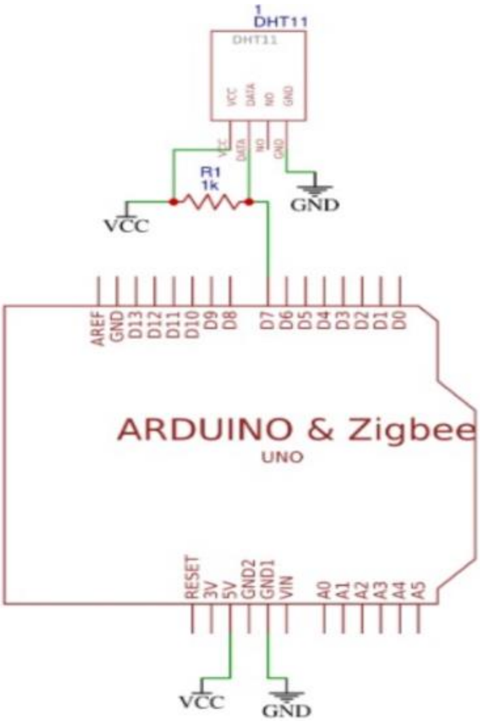

Figure 2: Schematics of the temperature module

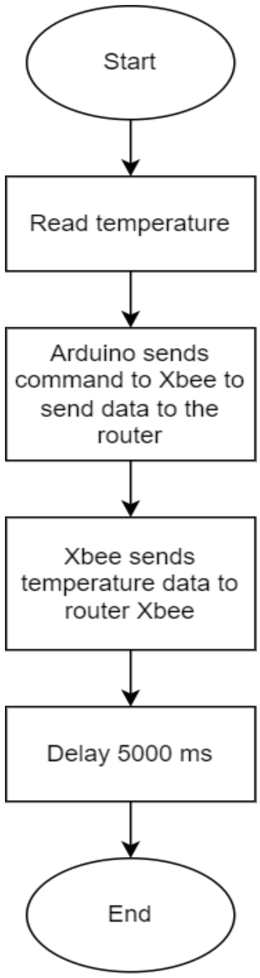

Figure 3: Flowchart of the temperature module

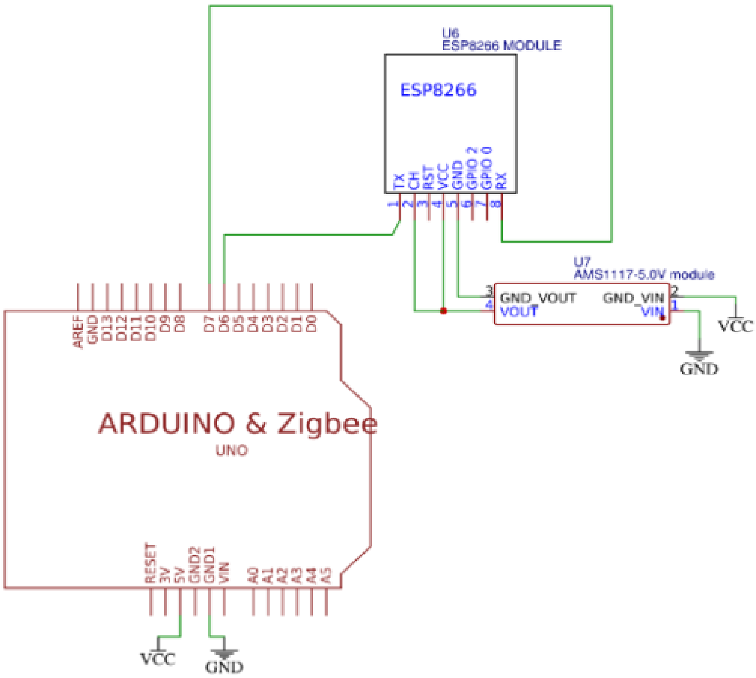

Figure 4: Schematics of the router module

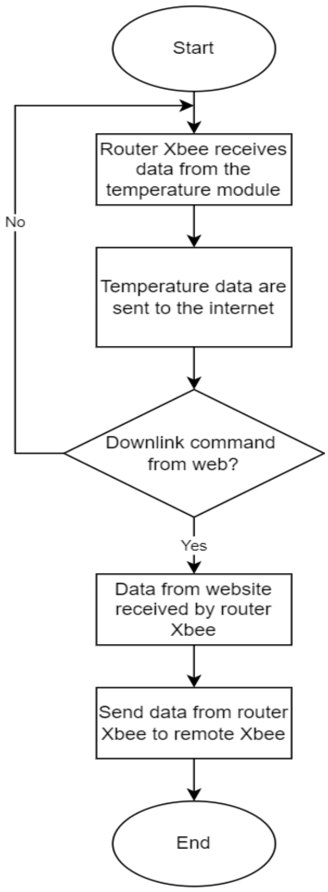

Figure 5: Flowchart of the router module

3.1.3 Remote module

From the router module, data packets sent from the router were not only shown to the user on the web interface, but also sent to the end device which proceeded to emit infrared signal to the AC corresponding to the instruction or command given from the user through the web interface.

In this module, the Arduino and ZigBee pair retrieved and processed data from the temperature module and the web interface. The pair was connected to four relays, three of which were connected to each button on the remote-control device: the power button, increase button, and decrease button. The relays acted as switches that could trigger the activation of the designated button.

The schematics and flowchart of the remote module are provided in Figure 6 and Figure 7.

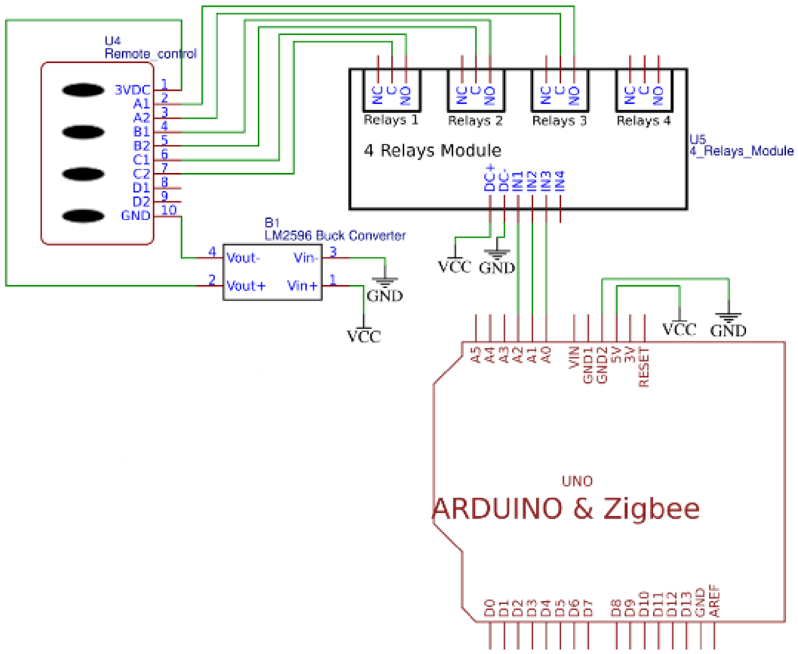

Figure 6: Schematics of the remote module

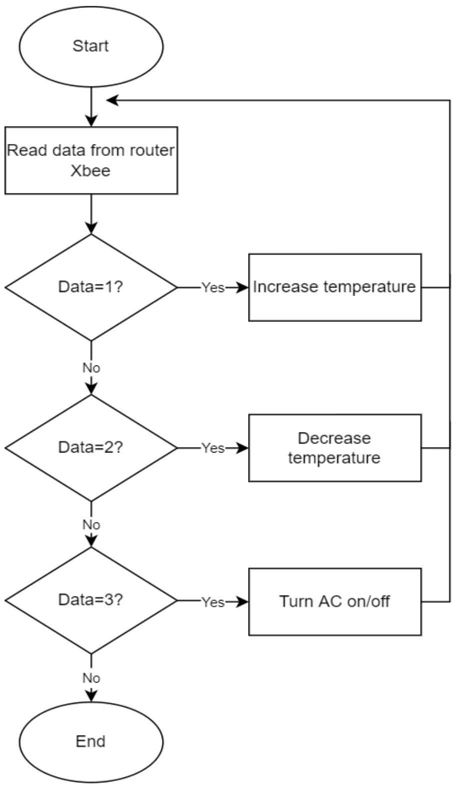

Figure 7: Flowchart of the remote module

3.2 Website Interface Design

The website interface was created using 000webhost, a free web-hosting service that provides support for PHP and MySQL. The graphics for the website's front-end were defined with JavaScript, HTML, and CSS. This service was selected because of its compatibility with the chosen database engine, MySQL, and cost reasons that were suitable for building a simple prototype for this project.

When accessing the website, users were prompted to login with credentials such as username and password. After logging in, the user would be able to see the interface of the remote AC control. On the website, information such as the last temperature, the current AC temperature, the AC status, and how the temperature changed overtime was shown. Buttons to turn on/off the AC, increase temperature, or decrease the temperature were also provided on the website. When the user clicked on the appropriate button, the command would be sent to the Arduino and ZigBee pair in the router module.

3.3 Database Design

A simple database design of using MySQL was developed for this project. MySQL was chosen as it is a mature database engine with stable support and proper documentation available. Two unrelated tables were designed to store real-time temperature measurements and the other to store the status of the AC. The table *temperature_data* stored the temperature measurements with their respective timestamps, while the table *status* stored the power state of the AC (on/off) and the last measured temperature shown on the web interface. A simple schema depicting the table attributes is shown in Figure 8.

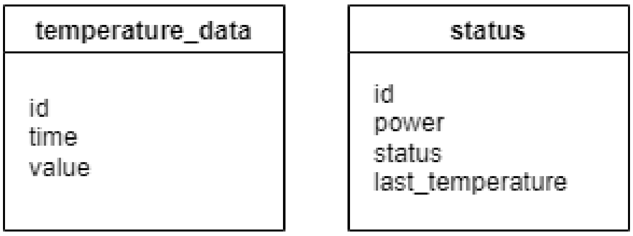

Figure 8: Table attributes in MySQL

4. TESTING PROTOCOLS AND RESULTS

In this project, functionality testing was done for individual modules of the system. The functionality testing objective was defined as the successful establishment of communication between each module and the communication between the system and the Internet. The testing was conducted with the following tests:

- Connection test between ZigBee modules
- Connection test between system to the Internet
- Connection test between ZigBee modules with varying ranges
- Latency test for data transfer between ZigBee modules

The test protocol emphasizes mainly on the successful communication as the goal of the system was ensured by it, because the transfer of commands from the web interface to and from the remote module relies heavily on the connection between ZigBee devices and the Internet.

4.1 ZigBee connection test

The ZigBee connection test was done with the XCTU software to configure the data transfer between ZigBee devices. With the XCTU software, a single data packet consisting of the byte 1 was sent from the temperature module ZigBee to the router module ZigBee. The same step was done to send a data packet from the remote module ZigBee to the router module. A snippet of the test on the XCTU software is shown in Figure 9. In this figure below, the same data packet was shown on both ZigBee modules, indicating that they were able to communicate successfully.

4.2 Internet connection test

The internet connection test was done to evaluate if the system was able to send data packets from the router module to the database and to the website interface. This test also established a threshold of the required internet speed to ensure successful transfer of data. The internet speed depended on the internet service provider, and in this project the connection to internet was tested using the 3G and LTE mobile network. The test showed that internet connection with a download speed below 6 Mbps would cause very slow data transfer or even cause data loss, hence no control commands from the web interface would reach the remote end device and the system would not be able to trigger the AC.

The test was done by evaluating the upload and download speed of both means for internet connection (3G and LTE). The results in Figure 10 and Figure 11 showed that connection with the 3G network was not sufficient to ensure reliable internet connection in the area where it was tested indoors.

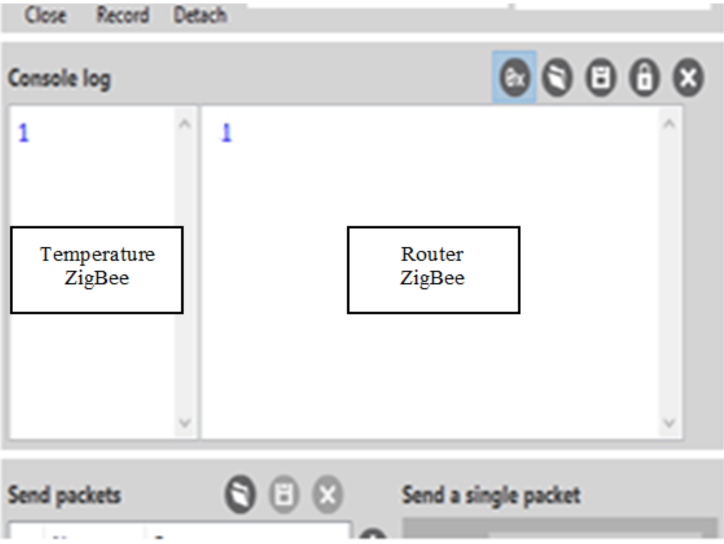

Figure 9: Connection test between temperature ZigBee and router ZigBee

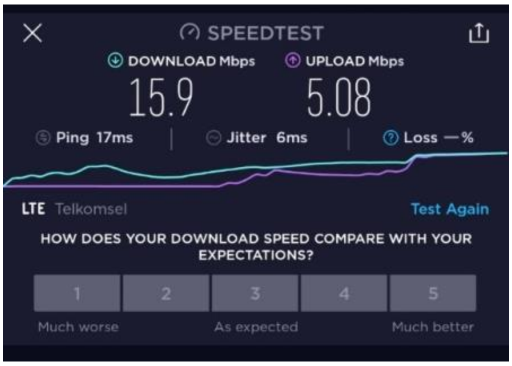

Figure 10: Internet connection upload/download speed with LTE network

Figure 11: Internet connection upload/download speed with 3G network

4.3 ZigBee range test

Maximum range between ZigBee modules was an important aspect to consider, since placing the ZigBee modules too far away from one another may cause communication between them to fail. This test was conducted to define the scope of possible ranges between ZigBee modules. The test results are shown in Table 2. In this test, the range (distance between ZigBee devices) was varied to evaluate if sending data packets between ZigBee devices was successful, from the temperature module to the router module and the router module to the remote module.

Table 2: ZigBee range test results

Number	Range (m)	Status
1	10	Success
2	30	Success
3	50	Success
4	70	Success
5	90	Success
6	110	Success
7	130	Success
8	150	Success
9	160	Fail
10	170	Fail

From the above table, it was determined that distance between ZigBee modules of more than 150 meters would cause failed transmission of data packets.

4.4 ZigBee latency test

In Doherty's survey of defining system response time with regards to user experience, for systems that are expected to have immediate response, the response time should fall between 300 milliseconds to 1 second [5]. Using this as a reference, a latency test was conducted to evaluate the delay in data transmission from the router module to the remote module. In other words, this test evaluated the time taken for a command sent through the web interface to activate the end device.

Because latency was affected heavily by the upload/download speed of the internet connection, the test was conducted under two internet speed conditions: download speed of less than 6 Mbps and more than 6 Mbps. The test results are shown in Table 3. From the results, an average response time of 5.72 seconds under low download speed and 1.44 seconds under high download speed could be inferred.

The average response time even in good internet condition was slightly longer than the expected requirement of maximum 1 second. This latency could possibly be improved with a higher download speed.

Table 3: ZigBee latency test results

#	Response time in seconds (speed < 6 Mbps)	Response time in seconds (speed > 6 Mbps)
1	5.4	1.3
2	5	1.1
3	6.2	1.5
4	5.2	2
5	6.8	1.3
6	7	1.3

5. WEBSITE IMPLEMENTATION

After ensuring that all modules were able to communicate reliably, the relevant information was shown on a web interface accessible via smartphone or computer. The implemented design of the GUI is shown in Figure 11, which shows the last measured temperature of the room, the remote device temperature, and the status of the AC device. The remote device temperature was also shown to provide information on whether the end device was overheating. A plot depicting how the temperature had changed overtime was also shown. This plot provided users with information on the system's capability of maintaining the desired temperature; the functionality requirement would be satisfied when the system was able to respond promptly to the given command and adjust the room temperature.

The GUI showed that the system could deliver its expected functionality, which was to monitor the room temperature. Below the temperature plot, users were able to click on the power button, increase button, or decrease button to control the AC status and the desired temperature. Clicking a button will send a corresponding trigger to the end device—the AC remote control—which would pass the command to the AC itself. This function was also achieved successfully within a run-through of the system.

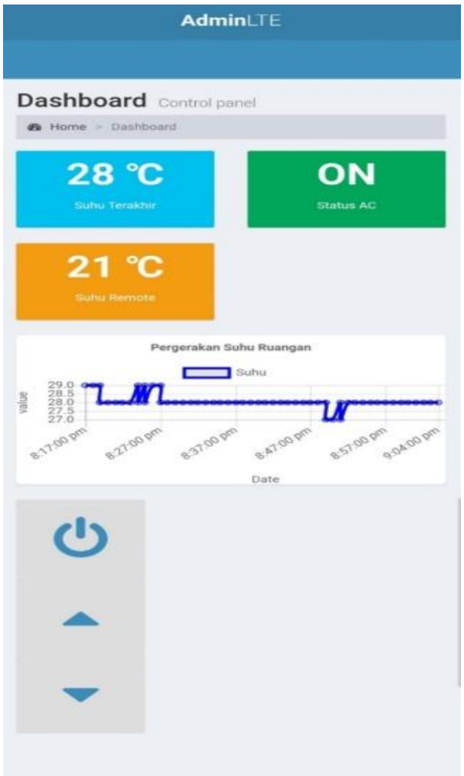

Figure 12: Web interface of the remote AC control. The blue box showed the last measured room temperature, the orange box showed the remote device temperature, and the green box showed the status of the AC

In the backend side, data was also successfully transmitted and stored in the database.

6. CONCLUSION AND FUTURE WORK

In this project, a prototypical design and implementation of an AC remote monitoring and control with ZigBee was proposed. System evaluation with the testing protocols showed that the system could perform the expected functions under adequate internet connection and placement of ZigBee modules. To ensure stable connection and data transfer, the internet download speed must not be lower than 6 Mbps, and the proximity between ZigBee modules should not exceed 150 meters. Considering the use of the system in mainly indoor settings within a WPAN, this maximum distance of 150 meters is deducted to be reasonable and cover enough area.

This project showed that the usage of ZigBee in WPAN for smart home use cases is viable and thus potentially enabling higher cost efficiency, compared to using WiFi or Bluetooth, while still maintaining robust communication between modules and devices in the network. ZigBee devices are generally more low-powered and require less bandwidth than the other two wireless network technologies and cover even wider ranges. The smart home use case could also be extended to other use cases that employ monitoring and control, such as smart manufacturing. With proper monitoring and control, energy efficiency can be achieved as the AC is not left turned on all the time.

Further improvements to this system can also be considered. One such improvement is to fully automate the AC control process by comparing the last measured temperature with a set-point temperature and send commands directly to the remote end device based on a control logic. This control logic could, for example, turn on the AC only if the last measured temperature is higher than the desired set-point. A more in-depth study into the energy expenditure with and without the system also needs to be done to provide a comparison baseline and to further strengthen the proof-of-concept that having monitoring and control can increase energy efficiency.

Another possible expansion is to utilize the data collected by the sensor to examine faults within the system, for example, checking if the temperature sensor yielded readings within an accepted range. Detecting early faults allow users to promptly repair the AC and keep it running reliably. In [10], detecting early faults would prevent leakages of fluorocarbon emissions from the freon coolant to help reduce global warming effects.

REFERENCES

1. Chang, S., & Nam, K. (2021). Smart Home Adoption: The Impact of User Characteristics and Differences in Perception of Benefits. *Buildings*, *11*(9), 393.
2. Sastry, K. R. K., & Kumar, P. P. (2019). Internet Enabled Remote Control Device for Air Conditioners. *International Journal of Innovative*

Technology and Exploring Engineering, *8*(10), 1901-1905.

3. David, N., Chima, A., Ugochukwu, A., & Obinna, E. (2015). Design of a home automation system using arduino. *International Journal of Scientific & Engineering Research*, *6*(6), 795-801.
4. Ohwo, O. B., & Olujimi, A. D. (2020). A Comparative Review of Emerging Wireless Technology. *International Journal of Scientific Research in Computer Science Engineering and Information Technology*, 7, 163-175.
5. Doherty, R. A., & Sorenson, P. (2015). Keeping users in the flow: mapping system responsiveness with user experience. *Procedia Manufacturing*, *3*, 4384-4391.
6. Ramya, C. M., Shanmugaraj, M., & Prabakaran, R. (2011, April). Study on ZigBee technology. In *2011 3rd international conference on electronics computer technology* (Vol. 6, pp. 297-301). IEEE.
7. Abhishek, T., Shivanee, K., Amith, R., Varun, C., Kiran, C., & Vishwas, K. V. (2021). ZigBee, It's Applications and Comparison with Other Short Range Network Technologies, *International Journal of Engineering and Technical Research (IJERT)*, 10, 891-897.
8. Leo, L. (2018). Working Principle of Arduino and Using it as a Tool for Study and Research, *International Journal of Control, Automation, Communication and Systems*, 1.
9. Mesquita, J., Guimarães, D., Pereira, C., Santos, F., & Almeida, L. (2018, September). Assessing the ESP8266 WiFi module for the Internet of Things. In *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)* (Vol. 1, pp. 784-791). IEEE. 10.1109/ETFA.2018.8502562.
10. Yoshiaki, B. (2020). Development of Remote Monitoring/Predictive Diagnosis Service for Air Conditioning, *Hitachi Review*, 69(1).