Evolution in Electrical and Electronic Engineering Vol. 4 No. 1 (2023) 229-235

EEEE

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/eeee
e-ISSN : 2756-8458

Design and Implementation of a Low-cost IoT-based SCADA System using ESP32

Mohammad Amin Johari[1], Chew Chan Choong[1]*

[1]Faculty of Electrical and Electronic Engineering,
Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Johor, MALAYSIA

*Corresponding Author Designation

DOI: https://doi.org/10.30880/eeee.2023.04.01.027
Received 15 January 2023; Accepted 11 April 2023; Available online 30 April 2023

Abstract: First, temperature is one of the most monitored variables in many commercial and industrial settings. Temperature sensors used in the Internet of Things (IoT) are typically relatively small, require very little power, and can gauge relative humidity. In an Internet of Things system, it is necessary to acquire temperature information that is both timely and accurate. The performance of an Internet of Things temperature system is compared with that of an industrial temperature system in this paper. In this project, an Internet of Things temperature monitoring system has been created by combining a Raspberry Pi with a DHT11 temperature sensor to simulate an Internet of Things temperature monitoring system. The primary purpose of this report is to design an Internet of Things (IoT)–based ESP32 device that can monitor environmental parameters and provide details regarding its implementation. It also offers knowledge regarding the Internet of Things and other potential solutions for this project. In addition to using the finished prototype, this report analysed several possible approaches. The Internet of Things temperature monitoring system that uses the DHT11 sensor offers accurate measurements up to degrees Celsius but has a slower response time when it comes to detecting temperature changes. This discovery could be a reference point when designing an Internet of Things system that incorporates temperature monitoring.

Keywords: Temperature, Humidity, IoT

1. Introduction

The "Internet of Things" (IoT) is a network of physical things`` equipped with software and hardware, allowing them to collect and share data. These various objects are interconnected through something known as the "Internet of Things," these various objects are interconnected. The (IoT) is increasingly important in numerous industries, including the financial, educational, and healthcare. Supervisory Control and Data Acquisition (SCADA) is the best instrument for managing such places in remote places. A SCADA system must monitor, regulate, and gather data from them. Type SCADA system, data from sensors, actuators, and controllers situated throughout the facility can be collected efficiently, the system can be operated remotely in real-time, and the output voltage, current, and power can be remotely monitored and maintained.

Corresponding author: chewcc@uthm.edu.my

SCADA is a "system that combines telemetry and data capture." The relationship between the SCADA system, IoT, and Raspberry Pi 4 is that Raspberry Pi 4 can be used as a remote device in a SCADA system to gather data from the sensor and send it to central system devices. Raspberry Pi 4 can be integrated with the SCADA system to provide more data, location, and flexibility. Additionally, Raspberry Pi 4 can remotely monitor and control process facilities for local operators.

One challenge in monitoring SCADA systems and IoT is effectively gathering and analysing data from many remote devices and sensors to ensure smooth operations and security. This includes identifying and addressing potential issues or malfunctions as detecting and preventing security. Secondly, a large amount of data must be collected, analysed, and stored, especially for systems with many devices and sensors. Additionally, the data from these devices may be in different formats and need to be integrated and analysed in real time. This requires monitoring for access controls.

Due to IoT and the integrated platform, it is now possible to link various machines and devices. The IoT is one component that facilitates the management and monitoring of ordinary and traditional home appliances. A monitoring system can be categorised into a wired or wireless system. In wireless communication, the connectivity will be more convenient and user-friendly. Thus, the weather monitoring system would not need the responsible person to be at the location [1]. Wireless communication is also the transfer of information or data over a distance without wires from the transmitter to the receiver.

1.1 Literature Review

The number of companies and organisations employing IoT applications has expanded at a rate that has never been seen before in a range of industries, including the automotive, utility, health, and logistics industries, as well as the home automation industry [2]. There is an urgent need to develop cloud-based middleware capable of managing many sensors and actuators as the number of connected devices continues to increase rapidly. IoT-Frameworks are constructed using software and hardware components available under an open-source license. These frameworks are designed to process and distribute data.

The term "smart" is derived from the concept of a "smart city," where everything can be connected to the internet. The IoT is a term used to describe the phenomenon of connecting objects or things to the internet, which has increased the IoT. The author reinscribed Mark Weiser's views on technology: "The most powerful technologies are those that disappear. There is not one design for the IoT that is universally accepted as the consensus. Instead, researchers have come up with various innovative architectural concepts. Architectures of IoT with Three and Five Layers, respectively A three-layer architecture [3] is the most basic form of architecture [4]. It was first utilised back when the research into this field was in its infancy. It comprises three layers the perception network and the application layer.

2. Materials and Methods

The proposed system's entire algorithm is shown in Figure 1. Initially, the ESP WIFI module and the Blynk server are initialised, the sensors are connected, and the values are read from the sensors. The working algorithm of the humidity sensor and temperature sensor are explained in a flowchart. The temperature sensor reads the temperature's analogue values, sends them into the Blynk server, and then updates the serial monitor.

2.1 Flowchart

Figure 1 shows a flowchart outlining the project development and divided into two parts: software and hardware. The software section's main issue is coding the temperature and humidity sensor, ESP32

and Cloud Server. Meanwhile, the components and project structure are detailed in the hardware section.

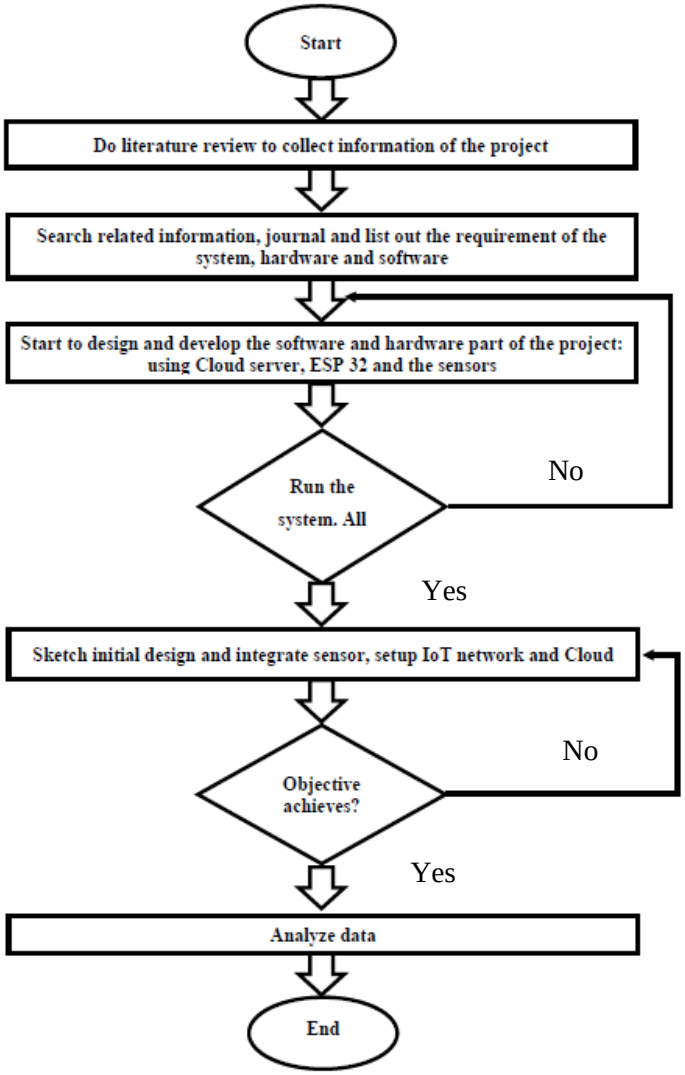

Figure 1: Flowchart of project development

2.2 Block Diagram

The proposed system in Figure 2 uses DHT 11 sensor. The microcontroller unit ESP32 Wi-Fi module and the primary processor module (NodeMCU). Due to its small size and low power consumption, the microcontroller unit is essential for measuring temperature and humidity sensors.

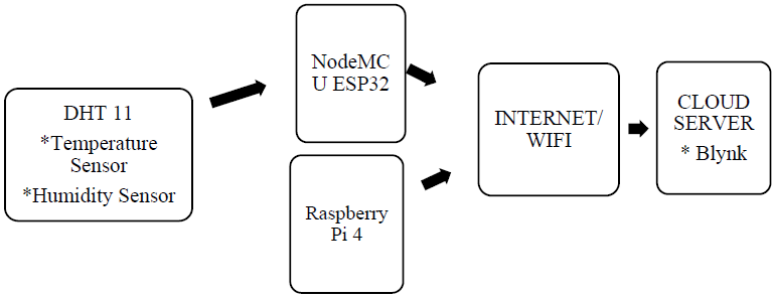

Figure 2: Block Diagram

2.3 Connection of Blynk Cloud and ESP32

Equations and the rough draft of the code for the access point have been created with the help of the Arduino IDE software. Adding a board manager to a preference is required to properly configure the library for use with the Arduino IDE software. After selecting the sketch component, choose the ESP32 Board from the library management menu. The code design in Figure 3 links the ESP32 Board to Wi-Fi and the Blynk Cloud. To finish configuring the Blynk token, which may be retrieved when the Blynk Cloud has been set up and running. To obtain the IP Address that is necessary to connect to the Wi-Fi and Monitor Blynk Applications, it is necessary to provide not only the name of the Wi-Fi network but also the password for that network.

Figure 3: Code connection of Blynk

3. Results and Discussion

The hardware of this project has been developed by referring to the circuit design. Figure 4 shows the view of the project hardware.

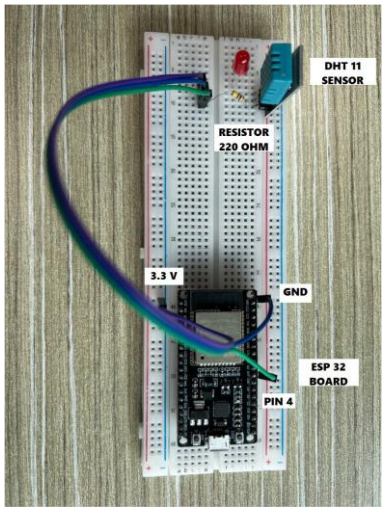

Figure 4: Prototype connection

3.1 Results of the monitoring system

Figure 5 and 6 show the mean data of temperature and humidity that were collected for three days.

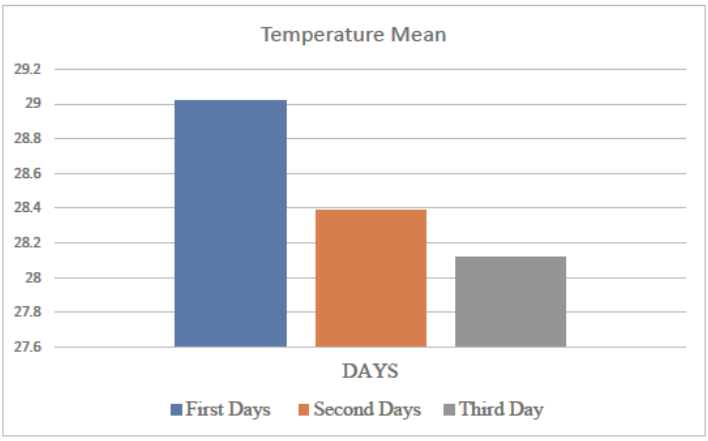

Figure 5: Temperature Mean comparison for Three Day

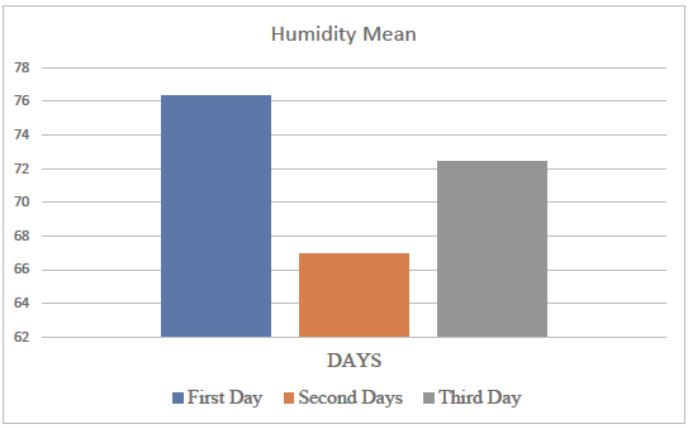

Figure 6: Humidity Mean comparison for Three day

3.2 Blynk 2.0

This section discusses the Blynk application results that appear on a mobile phone. The experiment is carried out to evaluate the system's performance, especially when the system sends the notification to the user's mobile phone wirelessly based on the reading obtained from the sensors. Figure 7 to 9 show the graphical user interface (GUI) of the Blynk application. On the Blynk application dashboard, the data such as temperature and humidity readings appear on the screen so that the user can monitor their indoor condition remotely via mobile phone.

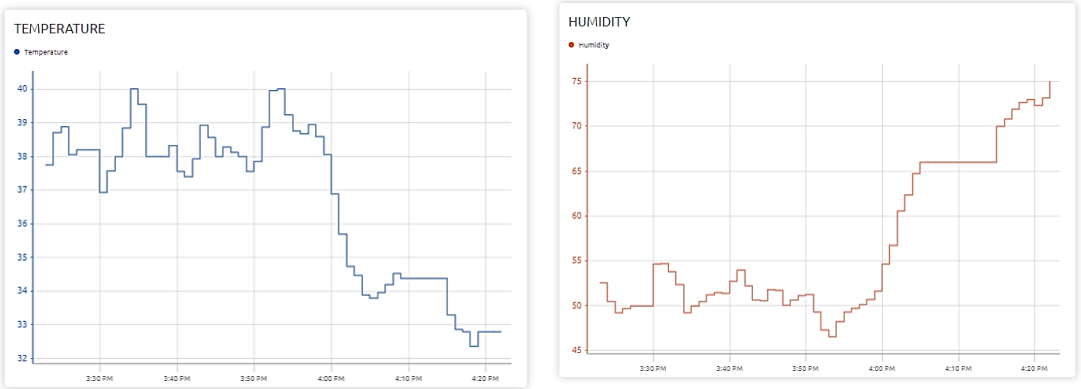

Figure 7: Temperature and Humidity sensor for One Hour

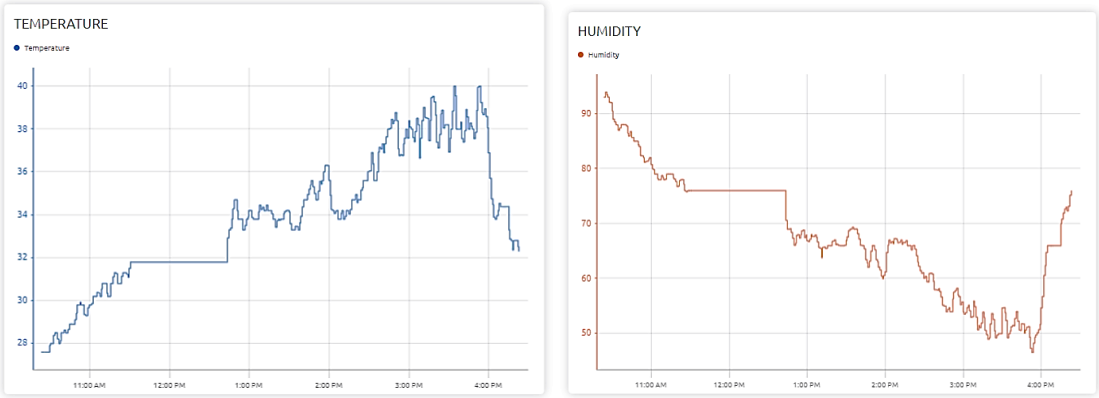

Figure 8: Temperature and humidity sensor for one week

Figure 9: Blynk IoT Smartphone Interface

4. Conclusion

In conclusion, the monitoring project's primary focus is providing data, including temperature and humidity readings. This particular type of sensor was utilised to obtain all the assignments for this project. The temperature and humidity sensor, known as DHT 11, is what this sensor is used for. Because the coding for each sensor is unique, it is necessary to combine it to make all of the sensors function as part of a single system. The Internet of Things network uses ESP32 as a controller and a Wi-Fi access point.

This project uses the Blynk mobile application for the Internet of Things platform, and all of the data collected is linked to his mobile application via his home Wi-Fi network. The connection can be established by entering the system's coding, network name (SSID), and Wi-Fi password. Compilation of the code takes place on the controller board. A connection with the Wi-Fi network is established while the controller board is powered up and operational. Since ESP32 can only operate at 2.4 GHz, that is the frequency used by the Wi-Fi network. It will take some time for the controller board to

connect with Wi-Fi because of the physical distance between the system and the Wi-Fi modem, as well as the signal strength of the network.

Acknowledgement

The authors would like to thank the Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, for its support.

References

[1] B. Varghese and R Buyya "Next generation Cloud Computing: New Trends and Research direction", future generation computer system, vol 79, pp 849-861,2018

[2] L. Bass, H. Fadhil, "The Perception of Information Security Threats Surrounding the Cloud Computing Environment." International Journal of Computing and Digital Systems 7, no. 06 (2018): 375-380

[3] L. Stein, B. S.S. Tejesh and S. Neeraja, "A Smart Home Automation system using IoT and Open- Source Hardware," International Journal of Engineering & Technology, vol. 7, no. 27, p. 428, 2018. Available: 10.14419/jet. v7i2.7.10856.

[4] Wei Yuantong, Dai yawn, Design of Industrial Monitoring ad hoc Network Protocol System Based on LoRa, Computer Measurement & Contro, 2019,27(02):225-228