International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 13 Issue: 08 | Aug 2026 www.irjet.net p-ISSN: 2395-0072

ESP-NOW Based Bidirectional Home Automation System with Real-Time Environmental Monitoring Using MicroPython

Kritish Mohapatra[1], Jyoti Prakash Sahoo[2]

[1]*Student, Dept. of Electrical Engineering, Odisha University of Technology and Research (OUTR), Bhubaneswar, Odisha, India*

[2]*Student, Dept. of Electrical Engineering, Odisha University of Technology and Research (OUTR), Bhubaneswar, Odisha, India*

---***---

Abstract - *Home automation systems today are usually based on Wi-Fi, internet and protocols such as MQTT, which require a stable router connection and introduce latency and points of failure. This paper describes an alternative low-latency home automation system that does not rely on a router for its operation. The core of the systems is based on the ESP-NOW protocol implemented on top of the MicroPython firmware for the ESP8266 microcontrollers. The system consists of two nodes: the sender, a control panel with four push buttons and a 0.96'' I2C OLED display, and a Receiver device that drives a 4-channel active low relay module and continuously samples a DHT11 sensor for temperature and humidity. The Sender node transmits Relay:State messages to the Receiver node's MAC address, in a compact form, every time that a button press occurs. The Receiver node executes the corresponding relay control command and then transmits the temperature and humidity readings, that were measured in the meantime, back to the Sender node, every two seconds. The temperature and humidity data are then displayed on the OLED display, in real time. The bidirectional communication between the two nodes results in low latency and low power consumption, while keeping the cost of the system low, being open source and easy to replicate.*

Key Words: ESP-NOW, MicroPython, Home Automation, Internet of Things, ESP32, ESP8266, Relay Control, Environmental Monitoring, Wireless Sensor Network

1. INTRODUCTION

Internet of Things (IoT) development led to the proliferation of home automation systems[2], mostly based on Wi-Fi based communication such as MQTT or HTTP like in the described application[4],[6]. Such means of communication are rather reasonable, as long as there is a stable internet connection and a Wi-Fi access point. However, the network introduces latency and additional power consumption, and, when offline, represents a single point of failure. In order to address this issue, Espressif offers a connectionless protocol ESP-NOW which allows the ESP32/ESP8266 modules to communicate directly with each other through their MAC addresses[1]. Thus, the protocol is suitable for local home automation purposes as it is capable of providing fast data transmission rates. Hence, ESP-NOW is perfectly suited for home automation applications which are supposed to work locally, efficiently and in real-time.

1.1 Motivation

The existing home automation systems offer complicated solutions for relatively simple tasks of mechanical activation inside the vicinity. Most of the researchers note that the reliable method of operating free of using a router while keeping the delay at a reasonable level would be possible with the help of cheap electronic components and MicroPython.

This system is intended for providing instant data about the conditions of heat and air moisture on the control panel. The system itself is not dependent upon the mobile phone application or internet access, which suggests that it is convenient in terms of operation and efficient in terms of power consumption.

1.2 Related Work

Prior home automation implementations largely use Wi-Fi with MQTT brokers, such as NodeMCU (ESP8266) based systems integrated with Home Assistant or Blynk. However, these solutions are dependent on a network to control the devices and read sensor readings[3]. ESP-NOW has been evaluated for its indoor performance and used for mesh sensor networks and peer-to-peer communication [5], but combining real-time bidirectional relay control with continuous OLED-based environmental feedback on low-cost hardware remains under-documented, motivating the present work.

2. PROPOSED SYSTEM AND METHODOLOGY

The proposed system uses two nodes programmed with MicroPython to communicate with one another via the ESP-NOW protocol. The Sender microcontroller is responsible for reading the states of four tactile push buttons (set up with internal pull-ups) and, every time the button is pressed, the sender transmits a message in the form RelayNumber:State (e.g., 1:1 for Relay 1 ON) directly to the Receiver's MAC address, which has been registered

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 13 Issue: 08 | Aug 2026 www.irjet.net p-ISSN: 2395-0072

as an ESP-NOW peer during setup. Once receiving the message, the Receiver decodes it and drives the corresponding channel of a 4-channel active-low relay module. In addition to receiving messages from the sender node, the receiver takes readings from DHT11 every two seconds and sends the obtained data to the last-known MAC address of the sender in the form T: Temperature, H: Humidity, which the Sender displays on a 0.96" I2C OLED screen.

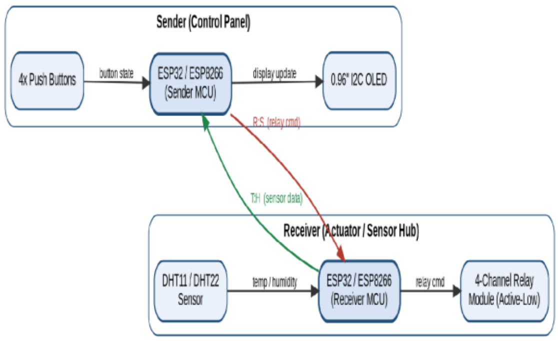

Fig -1: System Block Diagram

Table -1: Hardware Components Used

Component	Function
ESP32/ESP8266 (Sender)	Control panel MCU
0.96" I2C OLED (SSD1306)	Status display
4x Push Buttons	Relay toggle input
4x 10kΩ Resistors	Button pull-up
ESP32/ESP8266 (Receiver)	Actuator/sensor hub MCU
4-Channel Relay Module	Relay control (active-low)
DHT11/DHT22 Sensor	Temp & humidity sensing
Jumper Wires	Interconnection

This system uses the ESP-NOW protocol frame for both control and telemetry, avoiding the overhead of a broker or IP stack. Relay commands are unicast from Sender to Receiver, while sensor telemetry is unicast from Receiver back to the Sender using the MAC address of the last device it received a message from, allowing the pair to communicate without static network configuration beyond the initial peer registration.

The MAC address of the Receiver's station is read once using the MicroPython REPL and hardcoded as an ESP-NOW registered peer on the Sender's side. The two devices are both programmed with MicroPython and have separate event loops - the Sender checks the status of the buttons and updates the display screen, while the Receiver checks the DHT and waits for relay commands.

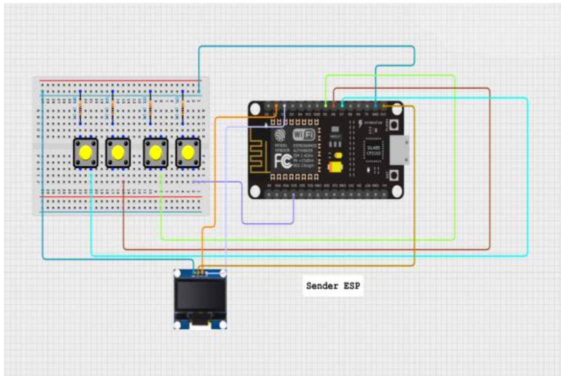

Fig -2: Sender Circuit Diagram (Control Panel)

The Sender board has the OLED display connected to the ESP8266 GPIO 5 (SCL) and GPIO 4 (SDA). The push button switches are connected to the GPIO 10, 14, 12 and 13 (each with a pull-up resistor to 3.3V).

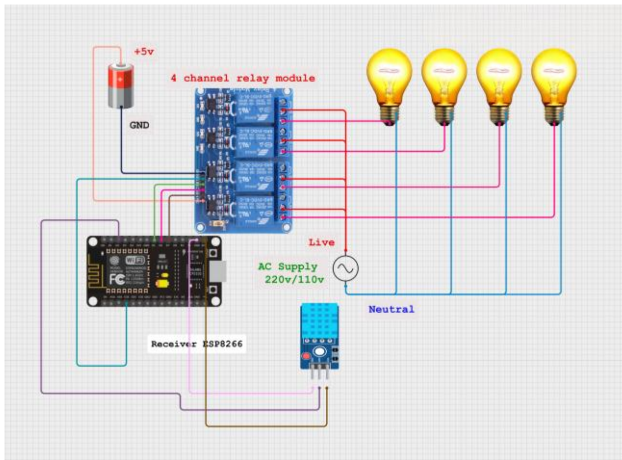

Fig -3: Receiver Circuit Diagram (Sensor Hub)

The Receiver board drives the four relay channels from GPIO 10, 14, 12 and 13, matching the Sender's button-to-relay mapping, while the DHT sensor's data line is connected to GPIO 4.

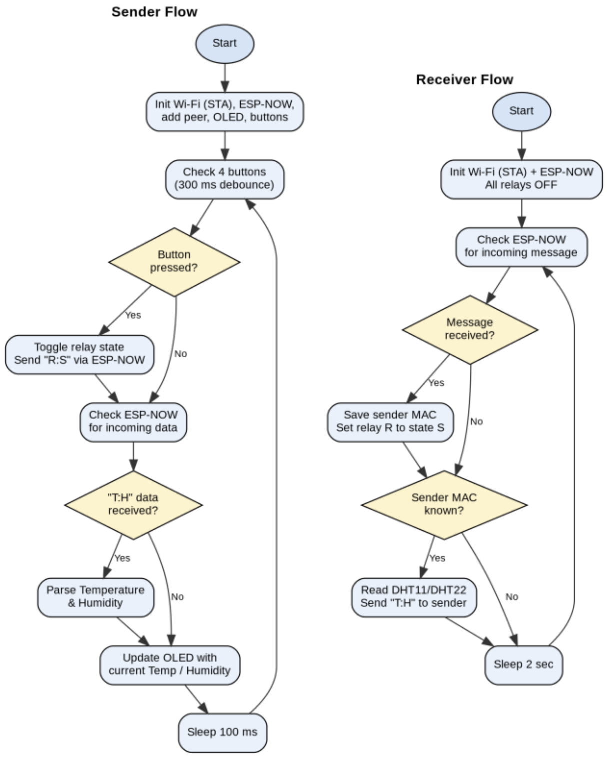

Fig -4: Sender and Receiver Operation Flowcharts

In testing, relay actuation following a button press was effectively instantaneous, consistent with the low, single-digit millisecond latency reported for ESP-NOW in prior indoor evaluations [2]. The temperature and humidity readings were sent from the Receiver to the Sender every two seconds and displayed on the OLED without any noticeable drift or missed packets over extended operation. Additionally, the system was tested without a Wi-Fi router connected to the network and the devices continued to work properly, confirming the system could function independently of the router.

3. CONCLUSIONS

The paper presents the home automation system with MicroPython scripts, based on the ESP-NOW protocol, establishing reliable communication between nodes. It enables bidirectional, low-latency and local (router-independent) control, with the real-time display of temperature and humidity on OLED and the possibility to manage four relays. The system is developed on the open-source MIT-licensed firmware, and thus can be considered a viable alternative for local home and building automation. The system can be extended in multiple ways, most notably, by adding more nodes, logging the data on the SD card, establishing voice or mobile application control while retaining the router independence.

ACKNOWLEDGEMENT

The authors would like to thank the Department of Electrical Engineering, Odisha University of Technology and Research (OUTR), Bhubaneswar, for providing the resources and environment to carry out this project.

REFERENCES

[1] Espressif Systems, "ESP-NOW User Guide," Espressif Systems Documentation.

[2] D. Kanellopoulos, V. K. Sharma, T. Panagiotakopoulos, and A. Kameas, "Networking Architectures and Protocols for IoT Applications in Smart Cities: Recent Developments and Perspectives," Electronics, vol. 12, no. 11, 2023.

[3] J. A. Cujilema Paguay, G. A. Hidalgo Brito, D. L. Hernandez Rojas, and J. J. Cartuche Calva, "Secure home automation system based on ESP-NOW mesh network, MQTT and Home Assistant platform," IEEE Latin America Transactions, vol. 21, no. 7, pp. 829-838, Jul. 2023, doi: 10.1109/TLA.2023.10244182.

[4] "IOT Based Smart Home Automation Using ESP32," IEEE Conference Publication, IEEE Xplore.

[5] "Indoor Performance Evaluation of ESP-NOW," in Proc. Wireless IoT Performance Studies.

[6] "A step towards Home Automation using IOT," IEEE Conference Publication, IEEE Xplore.