Journal of IoT in Social, Mobile, Analytics, and Cloud (ISSN: 2582-1369)
www.irojournals.com/iroismac/

A Hybrid Smart Home Automation System Using ESP32 with Edge AI and IoT Integration

Ravinder K.[1], Khaja Mujeebuddin Quadry[2], Romeela Rose P.[3], Sravani R.[4]

Department of Electronics and Communication Engineering, Joginpally B R Engineering College, Hyderabad, India.

E-mail: [1]kravinderece@jbrec.edu.in, [2]drkhajamujeebuddinece@jbrec.edu.in, [3]romeelaroseece@jbrec.edu.in, [4]rsravaniece@jbrec.edu.in

Abstract

This paper describes an approach for developing a hybrid smart home automation system through the incorporation of the ESP32 microcontroller along with IoT, Edge Artificial Intelligence (Edge AI), MQTT communication protocol, and Firebase cloud. In essence, this approach has been formulated by focusing on the development of an intelligent, fast and scalable system capable of automating different household appliances through real-time monitoring and control. In this regard, sensors like PIR, DHT11, and MQ2 have been used for motion detection, temperature monitoring, and gas leakage detection respectively. On its part, the ESP32 microcontroller has been used for performing edge-based processing in order to minimize communication delay and enhance system reliability. Meanwhile, the use of MQTT protocol will facilitate lightweight real-time communication between different devices and cloud servers. Similarly, Firebase cloud service will facilitate remote monitoring as well as data synchronization of the system in question using web or mobile interfaces. Experimentation reveals that the proposed system offers fast and reliable performance in addition to scalable multi-device operation. From the experiments, it has been established that the hybrid approach has resulted in efficient and scalable performance compared to other approaches commonly used for home automation.

DOI: https://doi.org/10.36548/jismac.2026.2.005
Extended version of article received from **National Conference on Electrical, Electronics & Communication Technologies (NCEET 2026)**
Received: 10.03.2026, received in revised form: 13.04.2026, accepted: 02.05.2026, published: 15.05.2026

Keywords: ESP32, Internet of Things (IoT), Smart Home Automation, MQTT (Message Queuing Telemetry Transport) Protocol, Passive Infrared (PIR) Sensor, Edge Artificial Intelligence (Edge AI), Firebase Cloud.

1. Introduction

Intelligent home automation using embedded systems and IoT has become a prominent use case that enables intelligent monitoring and controlling of appliances within homes through interconnections among devices. The evolution of wireless communication, cloud computing, and edge computing has considerably improved the efficiency and flexibility of current home automation systems. Traditional automation systems used wired communication and central control mechanisms, which made them relatively complex, costly, and difficult to maintain. The use of wireless communication technologies, such as Wi-Fi and Bluetooth, provided remote accessibilities and simpler connections, hence improving user convenience and operational flexibility [1][2].

Microcontroller-based IoT home automation systems have become very popular due to their inexpensive nature, ability to provide real-time monitoring of systems, and ease of implementing them. Among the many available controllers, the ESP32 controller has become very popular owing to its built-in communication capabilities via Wi-Fi and Bluetooth, as well as efficient processing capability and cloud-edge computing capability [3][4]. Current intelligent home automation systems include cloud-based platforms, lightweight communication protocols, and intelligent processing algorithms for real-time communication and automated control [5].

In this research work, a hybrid smart home automation system using ESP32 technology that includes IoT connectivity and edge processing, along with MQTT protocol and cloud computing has been described. It has been designed to offer the advantages of real-time monitoring, intelligent device operation, enhanced scalability, and multi-device communication. Through integration of edge processing with cloud computing technology, the system aims at reducing delay in communication and ensuring reliability and convenience for the users.

2. Literature Survey

The current trend in automation in smart homes revolves around the utilization of Internet of Things (IoT), wireless technologies, cloud computing, and intelligent control systems in order to increase automation efficiency and user comfortability [11]. The initial automation systems in smart homes were based on wireless communications involving Bluetooth and FPGA-based communication networks that would control appliances in the house. Despite being functional and performing their tasks effectively, such systems had shortcomings in terms of communication range and scalability and lacked internet connectivity [6].

Through IoT technologies, there has been much focus on ESP32-based smart home systems that offer affordable solutions that involve both Wi-Fi and Bluetooth wireless connections and perform well. ESP32-based IoT home automation systems provide effective appliance control and remote monitoring. Nevertheless, most systems that use ESP32 rely heavily on pre-configured logic in performing automation activities.

Cloud-enabled smart home management systems took another step ahead to increase their accessibility and device coordination by introducing cloud platforms along with ESP32 controllers. The cloud platforms enable monitoring and control of the environmental parameters as well as household appliances; nevertheless, they still suffer from limitations associated with scalability, communication delay, and lack of dynamic and interactive communication in real-time [8]. Another recent development has included home automation schemes based on MQTT to ensure real-time and efficient communication between IoT devices and the cloud servers. However, such communication schemes decrease scalability since they rely on centralized processing techniques [9].

Finally, some research efforts have been made to develop low-cost ESP32-based IoT gateways for establishing a seamless connection between edge devices and the cloud servers [10]. Though such schemes can help improve system scalability and reduce costs in the process, they do not concentrate on intelligent automation and real-time multi-way communication. Besides, cloud-based IoT applications have already been used to build a sustainable environment for intelligent smart homes [12]. Recent research is thus increasingly focusing on developing new IoT systems that incorporate edge computing, local decision-making algorithms, and hybrid communication schemes.

3. Methodology

The proposed system for smart home automation makes use of an IoT-based framework that incorporates edge processing and cloud communication in order to provide effective monitoring and control of appliances. The entire system is designed based on the ESP32 microcontroller that works as the main processor that is used for collecting information from sensors, making decisions, communicating, and controlling the appliances.

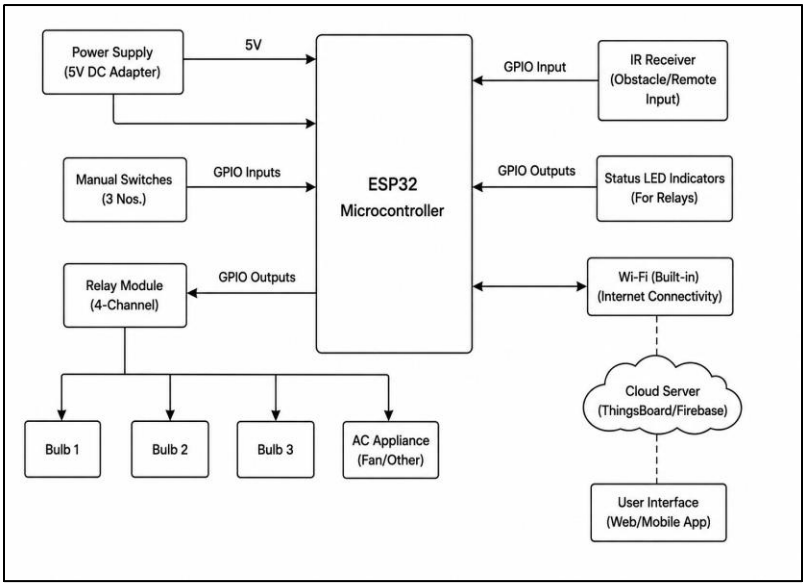

Figure 1. Block Diagram of the Proposed Hybrid Smart Home Automation System

The block diagram of the proposed system is illustrated in Figure 1. The system uses the ESP32 microcontroller to connect to various components, which include relays, switches, and environmental sensors via GPIO pins. Relay switches are used to control the functioning of electrical home appliances like lighting and fans. In addition, WiFi communication protocol has been integrated into the system for establishing internet connectivity with the use of Firebase cloud platform. Interaction with users can be achieved using the web or mobile application interface.

Environmental sensing is done continuously in the suggested system. Motion detection has been done using a PIR sensor, while temperature sensing has been done using the DHT11 sensor. Gas detection and alert have been done continuously using the MQ2 gas sensor. Any alert is raised if the gas concentration level goes above the predefined threshold value. The sensed values from all the sensors are processed using predefined logical conditions within the ESP32 microcontroller.

However, the proposed methodology focuses primarily on local edge processing rather than depending fully on the cloud. Such methodology will help improve communication delays as well as increase response effectiveness while running real-time. In case the ESP32 collects the sensor input, the controller will switch on the relays without involving cloud-based processing constantly. As for the communication system design, the methodology suggests using MQTT over Wi-Fi communication protocol for lightweight data transfers. Cloud-based server will synchronize device and monitor its status remotely, whereas the user can control the appliance via his mobile or web-based user interface. Manual switch option will also be considered to ensure the continued appliance functioning despite the communication problems.

4. Implementation

The suggested smart home system design based on the hybrid model of smart home automation system was designed using the ESP32 processor along with IoT technology, cloud services, relay modules, and environmental sensors. The suggested design aims at implementing real-time appliance management and control using improved communication efficiency and flexibility. The ESP32 acts as the controller and communicator in the design by taking input from sensors, performing automation tasks and controlling other connected appliances by relay modules. The Wi-Fi and Bluetooth connectivity provided in ESP32 make communication possible with the sensors, cloud platform, and user interface. Below Table 1 illustrates the components used in the suggested design. Below Table 1 summarizes the hardware components, software tools, and communication technologies used in the proposed implementation.

Table 1. Components Used in the Proposed System

Component/Technology	Function
ESP32 Microcontroller	Main controller with Wi-Fi and Bluetooth support
PIR Sensor	Motion detection for automatic lighting
DHT11 Sensor	Temperature and humidity monitoring
MQ2 Gas Sensor	Gas leakage and smoke detection
Relay Module	Switching control for appliances
Power Supply Unit	Provides regulated power supply
Arduino IDE	Programming environment for ESP32

MQTT Protocol	Real-time lightweight communication
Firebase Cloud Platform	Cloud storage and remote monitoring
Wi-Fi Communication	Internet-based remote connectivity
Bluetooth Communication	Local wireless communication backup

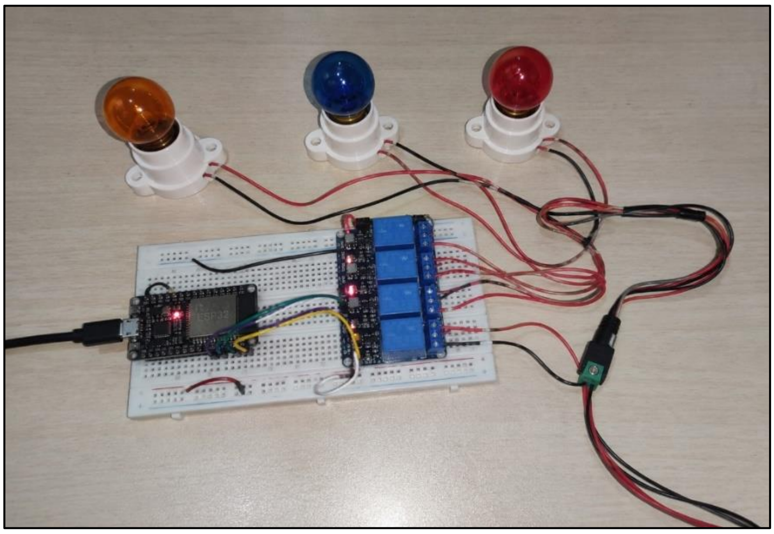

Figure 2. Experimental Setup

Figure 2 displays an experimental setup that includes an ESP32 development board coupled with a 4-channel relay module, power supply circuits, and several AC lighting loads. The relay module was interfaced via GPIO pins on the ESP32 for controlling domestic appliances like bulbs and fans. Switches were employed for manual control of appliances when communication was disrupted. The designed system further involved status LEDs to monitor relay switching during experiments.

The proposed design incorporates a PIR sensor, DHT11 sensor, and MQ2 sensor for detecting motion, environmental parameters, and gas leakage, respectively. Sensors' data is collected continuously and analyzed locally in the ESP32 using pre-defined automation rules. Edge computing in the system reduces reliance on the cloud and enhances efficiency during real-time operation. When temperature surpasses a certain level, the fan turns on, while gas leakage activates alert notifications.

The communication model was implemented using the MQTT protocol through Wi-Fi communication to ensure light-weight, real-time data exchange between the ESP32 and cloud server. The Firebase cloud platform was employed to provide remote data synchronization, device monitoring, and user authentication functionality. Web/Mobile-based user interface was developed to enable users to remotely monitor the status of appliances and carry out ON/OFF control actions. The web/Mobile-based user interface ensures real-time synchronization between the cloud database and connected devices.

Experimental evaluation was conducted under different operating conditions to analyze system performance in terms of response time, communication reliability, scalability, and automation accuracy.

The response time of the proposed system was evaluated using the difference between command initiation time and appliance execution time:

$$T_r = T_e - T_s$$

where T_r represents the system response time, T_e denotes appliance execution time, and T_s represents command initiation time.

The automation accuracy of the system was calculated using:

$$\text{Accuracy}(\%) = \frac{N_c}{N_t} \times 100$$

where N_c represents the number of correctly executed automation events and N_t represents the total number of experimental events.

The temperature-based fan control mechanism was implemented according to the threshold condition:

$$F = \begin{cases} 1, & T \geq T_{th} \\ 0, & T < T_{th} \end{cases}$$

where F represents fan status, T is the measured temperature, and T_{th} denotes the predefined threshold temperature.

Similarly, gas leakage detection using the MQ2 sensor was implemented using:

$$A = \begin{cases} 1, & G \geq G_{th} \\ 0, & G < G_{th} \end{cases}$$

where A represents alert status, G is the measured gar concentration, and G_{th} denotes the predefined gas threshold value.

5. Results and Discussion

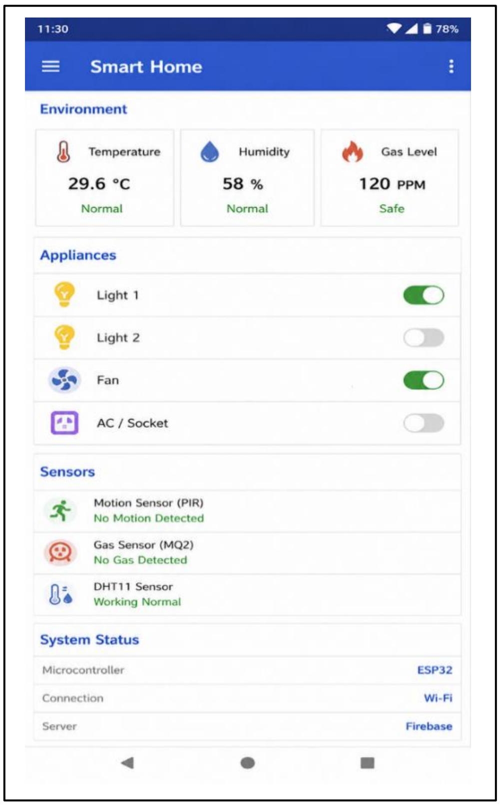

Figure 3. A Sample User Interface

The hybrid smart home automation system has been experimented upon under varied operating environments in order to study the response time, reliability, and scalability of the system. Figure 3 depicts the user interface for remotely controlling appliances from the proposed smart home automation system. Using this interface, users can check status and turn appliances ON/OFF by communicating with them via the cloud.

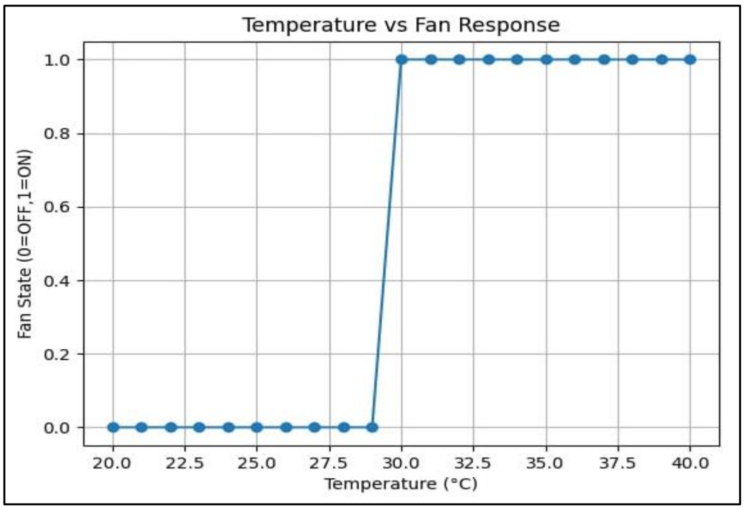

Figure 4. Temperature Versus Fan Response Characteristics

Figure 4 depicts the fan response due to temperature fluctuation. The proposed system was successful in activating the fan as soon as the sensed temperature exceeds the predefined threshold value. It was noticed that the system responds to changes quickly and reliably, proving the utility of local edge-level processing for environmental monitoring systems. Furthermore, no delay was observed while switching ON/OFF the fan relay.

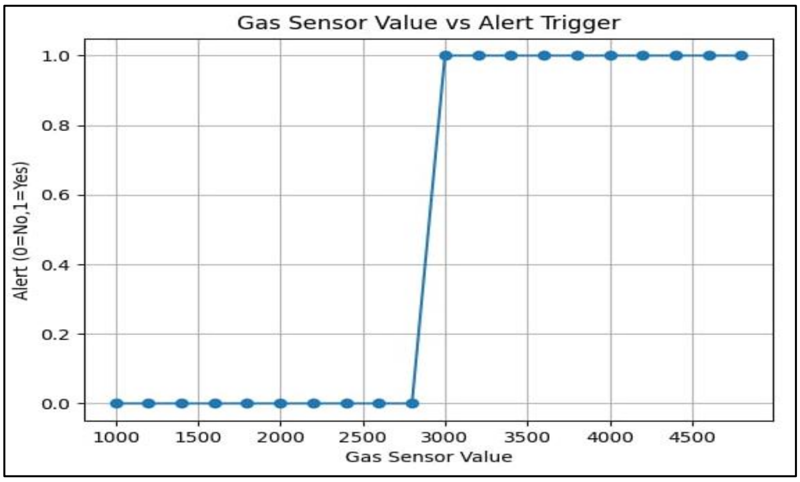

Figure 5. Gas Sensor Response and Alert Triggering Performance Using the MQ2 Sensor

Figure 5 represents the performance of the gas leakage detection module using the MQ2 sensor. As expected, the system triggers an alarm whenever the sensed gas concentration exceeds the defined threshold limit. The obtained result is indicative of the accuracy and reliability of the proposed system for household safety applications. The edge-level processing carried out by the ESP32 module resulted in minimizing the response time.

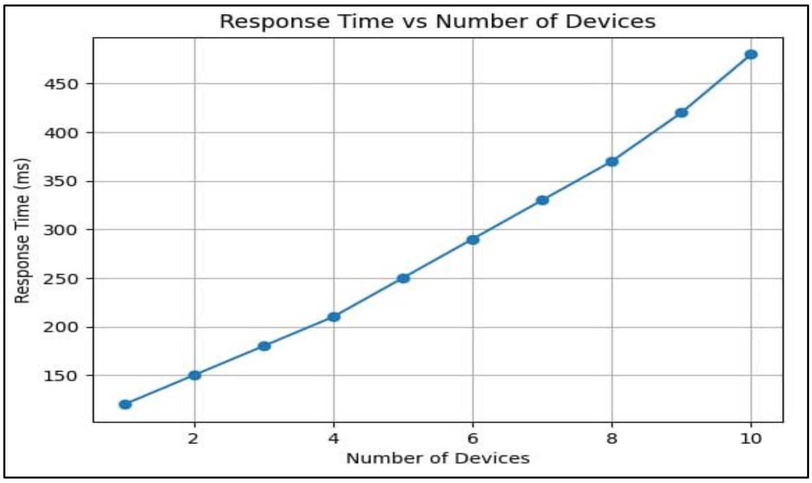

Figure 6. Variation of System Response Time

Figure 6 represents the impact of the number of devices on the system's response time. It can be seen from the figure that the response time increases as more and more devices are added to the existing setup. However, it was noticed that the response time does not exceed one second even with multiple devices operating at a time.

Overall, the results of the experiment show that the designed system offers reliable real-time automation with stable communication and effective functioning on multiple devices. The incorporation of edge computing and MQTT communications increases the speed of implementation, minimizes the cost of communications, and ensures scalability of the system. When compared with the traditional Bluetooth-enabled home automation systems, the developed system offers enhanced communication range, higher responsiveness, and greater flexibility of use.

6. Conclusion

The proposed hybrid smart home automation system effectively integrates ESP32, IoT communication techniques, edge AI processing, MQTT protocol and cloud monitoring system to create an intelligent and efficient home automation system. The developed smart home automation system is capable of providing reliable real-time monitoring and automation for household devices through environmental sensing and local decision-making processes. According to the experimental results obtained during this study, the system was found to deliver reliable performance characterized by minimal response time, stable communication performance, and efficient multiple devices management. Furthermore, PIR, DHT11, and MQ2 sensors have been utilized in order to enhance the capabilities of the smart home automation system, allowing its users to control lighting, temperature-based fan operations, and gas leakage. The use of MQTT protocol allows minimizing the communication overhead and providing reliable real-time data exchange. Meanwhile, Firebase cloud platform enables remote access and synchronization of all the devices connected to the network. Overall, the developed architecture provides a cost-effective, reliable, and intelligent smart home solution suitable for modern IoT-enabled living environments and future smart city applications.

References

[1] Wilson, Charlie, Tom Hargreaves, and Richard Hauxwell-Baldwin. "Smart Homes and Their Users: A Systematic Analysis and Key Challenges." Personal and ubiquitous computing 2015, vol. 19, no. 2: 463-476.

[2] Al-Qutayri, M. A. "Smart Home Systems." 2010, In InTech eBooks. https://doi.org/10.5772/182.

[3] Heath, S. "Embedded Systems Design" (Second edition). Newnes. https://api.pageplace.de/preview/DT0400.9780080477565_A23516682/preview-9780080477565_A23516682.pdf.

[4] Ibrahim, Usman Isah, Henry Ohize, Usman Aaze Umar, and Yusuf Aliyu. "Design and Construction of Voice-Controlled Home Automation Using Arduino." ABUAD Journal of Engineering Research and Development 2024, vol. 7, no. 1: 144-152.

[5] Singh, Simar, Sourabh Anand, and M. K. Satyarthi. "A Comprehensive Review of Smart Home Automation Systems." Advances in computer science and information technology (ACSIT) 2023, vol. 10, no. 2: 61-66.

[6] Krishna, B. Murali, V. Narasimaha Nayak, K. Reddy, B. Rakesh, P. Kumar, and N. Sandhya. "Bluetooth Based Wireless Home Automation System Using FPGA." Journal of Theoretical & Applied Information Technology 2015, vol. 77, no. 3: 411-420.

[7] Anusha, K., S. Aravind, N. U. Rao, and T. Ravisagar. "Internet of Things (IoT) Based Home Automation Using ESP-32." Int. J. Res. Appl. Sci. Eng. Technol. (IJRASET) 2024, vol.12, Issue 5, 1672-1676.

[8] Imam, Abubakar Surajo, Abdussalam Garba, A. S. Isah, I. F. Chukwu, Muhammad Ahmad Baballe, and Sagir Shehu. "Design of an ESP32-Based IoT Smart Home Automation Management System." Global Journal of Research in Engineering & Computer Sciences 2024, vol. 4, no. 4: 121-130.

[9] Arigela, Arun Kumar, Chandini Banapuram, and Nookala Venu. "Remote Based Home Automation with MQTT: ESP32 Nodes and Node-RED on Raspberry Pi." In 2024 8th international conference on I-SMAC (IoT in social, mobile, analytics and cloud), 313-318.

[10] Serepas, Filippos, Ioannis Papias, Konstantinos Christakis, Nikos Dimitropoulos, and Vangelis Marinakis. "Lightweight Embedded IoT gateway for Smart Homes Based on an ESP32 Microcontroller." Computers 2025, vol. 14, no. 9: 391.

[11] Ashton, Kevin. "That 'Internet of Things' Thing." RFID journal 2009, vol. 22, no. 7: 97-114.

[12] Etuk, Ubong E., Gabriel Omenaru, Saviour Joshua Inyang, and Imeh Umoren. "Towards Sustainable Smart Living: Cloud-Based IoT Solutions for Home Automation." Journal of Information Systems and Informatics 2023, vol. 5, no. 4: 1743-1763.