BLE-Based Smart Home Automation System

Abdulwahab Al Furaidan   
Department of Network and Communication, College of Computing Sciences and Information Technology (CCSIT), King Faisal University (KFU)

Saudi Arabia  
ID: 222407732

Abdullah Alnasser  
Department of Network and Communication, College of Computing Sciences and Information Technology (CCSIT), King Faisal University (KFU)

Saudi Arabia  
ID: 222433435

Dr. Abdullah Albuali  
Department of Network and Communication, College of Computing Sciences and Information Technology (CCSIT), King Faisal University (KFU)

Saudi Arabia Faisal Samer Musallam  
Department of Network and Communication, College of Computing Sciences and Information Technology (CCSIT), King Faisal University (KFU)

Saudi Arabia  
ID: 222453423

*Abstract*— This paper describes the design of BLE-Based Smart Home Automation System with the aim of solving the problems that Wi-Fi systems have with efficiency in power consumption and interferences. The system utilizes the ESP32 and Bluetooth Low Energy (BLE) technology for efficient, secure, and energy-saving home automation. The system contains DHT11 for measuring temperature and humidity, BH1750 for measuring light intensity, and a relay to control the lamp connected. Using BLE, the data is processed and sent to a mobile application developed with MIT App Inventor which provides real-time monitoring and control features of the devices connected.

In addition, the system integrates with Arduino IoT Cloud for remote data logging and device management via a dashboard where users can monitor environmental conditions and manually control the lamp. Improving user comfort and operational efficiency. The results prove the reliability of the system in real-time data monitoring and actuator control while maintaining low power consumption and almost low interference. This paper highlights the potential of BLE in smart home applications, offering a scalable and sustainable solution to modern home automation challenges.

Keywords— Automation of Smart Home, Bluetooth Low Energy (BLE), ESP32

# Introduction

The evolution of mobile and wireless networks paves the way for new innovative solutions in smart home automation, ensuring the priority of convenience, efficiency, and security of the system. Usually, smart home systems come with Wi-Fi, which is power inefficient and experiences frequent interference in densely connected environments. Therefore, these two disadvantages often come with higher operation costs and reduce the reliability of applications, hindering their adoption to widespread use.

This project shows the use of BLE technology in smart home automation as an alternative to Wi-Fi systems. BLE has several very clear advantages, such as: low power consumption, low interference, and secure communication.

Also, the project aims to design and implement a BLE-based smart home using these advantages to ensure seamless integration of sensors, actuators, and user interface integration while saving energy as much as possible.

The system uses the ESP32 microcontroller integrated with a DHT11 sensor for temperature and humidity monitoring, a BH1750 sensor to measure light intensity, and a relay that controls a connected lamp. The mobile application is developed using MIT App Inventor to provide users with real-time environmental data and control through BLE. Also, it is also integrated with Arduino IoT Cloud for remote monitoring, data logging and device management, which makes it scalable and accessible.

The motivation behind this project is to satisfy the ever-growing demands for energy-efficient and reliable home automation systems in a smart city environment. This project tries to show the capabilities of BLE technology to improve user experience by pointing out some of the weaknesses concerning Wi-Fi-based systems and contributes to the wide area of mobile and wireless networks. Basically, the proposed solution will have an impact on the ability to upgrade the technology of sustainable and scalable smart homes, making it a valuable contribution within the IoT landscape.

# Detailed System Description

The system integrates a hardware platform with intuitive software design to achieve effective, user-friendly home automation. The implementation included integrating sensors and actuators into an ESP32 microcontroller, programming the system to support BLE communication, and developing a mobile application to interact with the system.

The hardware platform of the system consists of one ESP32 microcontroller that serves as a central processing unit and BLE server. The ESP32 is connected with a DHT11 sensor for temperature and humidity monitoring, a BH1750 sensor to measure light intensity, and a relay module for the lamp. The ESP32 will handle the transmission of data from sensors, process the inputs, and send the results using BLE. It also controls the relay module to allow manual and automated control of the lamp. Power management considerations were a priority because the system leverages BLE's low power consumption for energy efficiency.

On the software side, the ESP32 was programmed using the Arduino IDE. The firmware included BLE configurations with custom UUIDs, which allowed for communication between the microcontroller and the mobile application. Real-time sensor readings were transmitted as BLE characteristics, while the lamp was controlled via commands sent from the app. Integration with the Arduino IoT Cloud allowed remote monitoring, data logging and device management through a dashboard, extending the system's functionality beyond local BLE interactions.

The mobile application was developed using MIT App Inventor, which enables the user to interact with the system through BLE. In the application, it displays real-time data of temperature, humidity, and light intensity, and has a button for switching of the lamp. The development of this application was one of the most challenging aspects of this project. Establishing a stable BLE connection with the ESP32 was difficult, with many attempts resulting in disconnections. This issue was resolved by making sure the BLE configuration in the ESP32 firmware is in sync with the MIT app inventor in terms of UUID mismatch and permissions configuration.

Another challenge was displaying sensor data accurately on the app interface. This was sorted out with consistent formatting of BLE characteristics in the ESP32 firmware and some careful debugging in the App Inventor.

In the end, the proposed system successfully delivers reliable real-time monitoring and control. The BLE-based system could bring scalability and energy efficiency using BLE and cloud connectivity. It provides a scalable and energy-efficient solution for smart home automation, demonstrating the potential of BLE-based systems in the Internet of Things field.

# Network Design

For a BLE-based home automation system, a star topology is ideal, with devices like lights and sensors connected to a central hub. The range of BLE is around 100 meters, but obstacles can reduce it. Security can be ensured using Bluetooth’s encryption features, and a BLE mesh network can extend coverage. VLANs can be used to separate different devices for better security and performance. For example, one VLAN for home automation devices and another for user devices. This isolates traffic, improves security, and can prioritize critical devices. Subnetting helps organize devices and control traffic. Devices can be grouped into subnets, one for IoT devices, another for user devices to manage resources and enhance security. To improve communication, selecting the best BLE channel, placing devices strategically, and using mesh networking helps avoid interference. Low-power modes on the ESP32 extend battery life, and regular software updates ensure efficient operation. Utilizing OAT(Over the Air) to update the system remotely to reduce maintenance time.

# Wireless and Mobile Concepts

The ESP32 supports various wireless communication protocols for home automation. Wi-Fi enables internet connectivity for cloud integration and remote control, though it consumes more power. Bluetooth/BLE is ideal for local, low-power applications. ESP-NOW allows direct, low-power communication between ESP devices without a router. LoRa and Zigbee, both requiring external modules, offer long-range or mesh networking for extended coverage, while MQTT facilitates lightweight messaging, typically over Wi-Fi. Wi-Fi is ideal for cloud systems, BLE and ESP-NOW suit local setups, and LoRa or Zigbee work for larger networks.  
  
For Wi-Fi implementation, the ESP32 connects to the local network and controls devices like relays and sensors via GPIO pins. Communication can be achieved through Arduino IoT Cloud. Wi-Fi excels in remote monitoring, and appliance control. Reliability, and encryption are crucial for secure and efficient operation.  
  
For BLE implementation, the ESP32 acts as a BLE server, using BLE characteristics to send or receive commands or sensor data. BLE supports local device control through smartphones, short-range monitoring, and mesh networks for wider coverage. Its low power consumption and independence from the internet make it ideal for battery-powered systems. Combining Wi-Fi for cloud access and BLE for localized control makes the ESP32 a perfect choice for smart home systems.

# Validation

The communication performance of the system was tested using both BLE and Wi-Fi. Inside the room, BLE had a signal strength of -43 dBm with a latency of 45 ms, while outside the room, about 4 meters away, the signal strength was weaker at -75 dBm with a latency of 80 ms. On the other hand, Wi-Fi had a signal strength of -50 dBm inside the room with a latency of 120 ms, and outside, the signal dropped to -74 dBm with a latency of 177 ms. For power consumption, Wi-Fi draws 200 mA, while BLE only draws 100 mA. While Wi-Fi offers more range, it comes with higher power consumption. BLE, on the other hand, provides lower latency and more efficient power usage.

GITHUB:

https://github.com/Abdulwahab65/BLEBasedHomeAutomationSystem

##### References

1. M. Afaneh, *Intro to Bluetooth Low Energy*. Fishers Novel Bits August, 2018.
2. mo-thunderz, “GitHub - mo-thunderz/BleMobileAppMitAppInventor: MIT App Inventor Mobile BLE controller app,” *GitHub*, 2023. https://github.com/mo-thunderz/BleMobileAppMitAppInventor (accessed Dec. 13, 2024).
3. “ESP32 BLE Server and Client (Bluetooth Low Energy) | Random Nerd Tutorials,” Nov. 11, 2021. https://randomnerdtutorials.com/esp32-ble-server-client/
4. “ESP32 Relay Module - Control AC Appliances (Web Server) | Random Nerd Tutorials,” Dec. 18, 2019. https://randomnerdtutorials.com/esp32-relay-module-ac-web-server/
5. “ESP32 with DHT11/DHT22 Temperature and Humidity Sensor using Arduino IDE | Random Nerd Tutorials,” *Random Nerd Tutorials*, Apr. 25, 2019. https://randomnerdtutorials.com/esp32-dht11-dht22-temperature-humidity-sensor-arduino-ide/
6. “ESP32 with BH1750 Ambient Light Sensor | Random Nerd Tutorials,” Mar. 08, 2022. https://randomnerdtutorials.com/esp32-bh1750-ambient-light-sensor/
7. *Arduino.cc*, 2024. https://docs.arduino.cc/arduino-cloud/guides/esp32/

[8] “ESP32 Wireless Communication Protocols | Random Nerd Tutorials,” Oct. 28, 2022. <https://randomnerdtutorials.com/esp32-wireless-communication-protocols/>

[9] https://www.facebook.com/techrmcom, “Practical guide to ESP32 communication protocols - Techrm,” Techrm, Jun. 29, 2024. <https://www.techrm.com/practical-guide-to-esp32-communication-protocols/>

[10] Espressif, “ESP-NOW Wireless Communication Protocol | Espressif Systems,” www.espressif.com. <https://www.espressif.com/en/solutions/low-power-solutions/esp-now>

[11] 2- Chrysafides, C., Wang, Y., & Xu, L. (2019). Power management in BLE devices for home automation. IEEE Transactions on Industrial Electronics, 66(9), 7238-7247.

[12] Gao, W., Zhang, Z., & Jiang, Y. (2020). Optimizing BLE communication for IoT applications. Wireless Communications and Mobile Computing, 2020, 1-10.

**I**