Project Title: Smart Home Automation System Using Embedded System & IoT

CSE234: Embedded Systems and IoT Lab

Submitted To

Md. Moniruzzaman Hemal
Lecturer
Department of CSE
Daffodil International University

Submitted By

Name	Student ID
Md. Fahimur Rahman	221-15-5953
Md. Jahidul Islam (Shuvo)	221-15-5003
Afia Benta Aziz Tonima	221-15-5115
Murshed Al- Faruq	221-15-5001
Ferdous Mahamud	221-15-4908

Section: 61_P2
Department of CSE
Daffodil International University

Date of Submission: 17/08/202

Table of Contents

1. Introduction & Problem Statement

The rapid evolution of the Internet of Things (IoT) has ushered in a new era of home automation, transforming how individuals interact with their living spaces. By integrating embedded systems with IoT technologies, smart home systems enable seamless control and monitoring of household appliances, offering enhanced convenience, energy efficiency, and security. This project develops a Smart Home Automation System that leverages an Arduino Uno microcontroller, Bluetooth communication, and relay modules to control appliances such as lights, fans, and power sockets remotely via a mobile application. The system is designed to be cost-effective, scalable, and user-friendly, addressing the growing demand for intelligent home management solutions in modern households.

The problem addressed by this project stems from the inefficiencies and limitations of traditional home management systems, which often require manual intervention and lack remote accessibility. With increasing energy costs, security concerns, and the need for convenience in fast-paced lifestyles, there is a pressing need for an automated system that can intelligently manage home appliances while being accessible to users with varying technical expertise. Additionally, many existing smart home solutions are expensive or complex, limiting their adoption in resource-constrained environments. This project aims to bridge these gaps by developing a prototype that combines affordability with functionality, paving the way for scalable and future-ready home automation.

Key Objectives of the Project:

- Develop a cost-effective and scalable IoT-based home automation system using embedded systems.
- Enable remote control of home appliances through a user-friendly mobile application.
- Ensure secure and reliable communication between the system components and the user interface.
- Lay the foundation for future enhancements, such as sensor-based automation and integration with advanced technologies like voice assistants and machine learning.
- Enhance energy efficiency by optimizing appliance control and reducing unnecessary power consumption.

Problem Statement Highlights:

- **Manual Control Limitations**: Traditional home appliances require physical interaction, which is inconvenient for users with mobility issues or those away from home.
- **Energy Inefficiency**: Lack of automated control leads to wasted energy, increasing utility costs and environmental impact.
- **Security Concerns**: Conventional homes lack real-time monitoring, making them vulnerable to security breaches like unauthorized access.
- **High Costs of Existing Solutions**: Many commercial smart home systems are expensive, limiting accessibility for middle- and low-income households.

- **Scalability Issues**: Existing systems often lack flexibility to integrate additional devices or advanced features like predictive automation.

By addressing these challenges, this project delivers a practical prototype that demonstrates the potential of IoT and embedded systems in creating intelligent, accessible, and efficient home environments.

---

2. System Architecture / Design

The system architecture ensures seamless communication between hardware components and a mobile application via Bluetooth. The core components include an Arduino Uno microcontroller, an HC-05 Bluetooth module, a 4-channel relay module, and controlled appliances (fan, red light, pink light, and power socket). A status LED indicates the system's operational state.

2.1 Components

- **Arduino Uno**: Central microcontroller for processing commands and controlling appliances.
- **HC-05 Bluetooth Module**: Facilitates wireless communication with the mobile app.
- **4-Channel Relay Module**: Controls appliances by switching them on/off.
- **Appliances**: Fan, red light, pink light, and power socket.

2.2 Block Diagram

The block diagram illustrates the flow of control, where the HC-05 module receives commands from the mobile app, processed by the Arduino Uno, which then controls the appliances via the relay module.

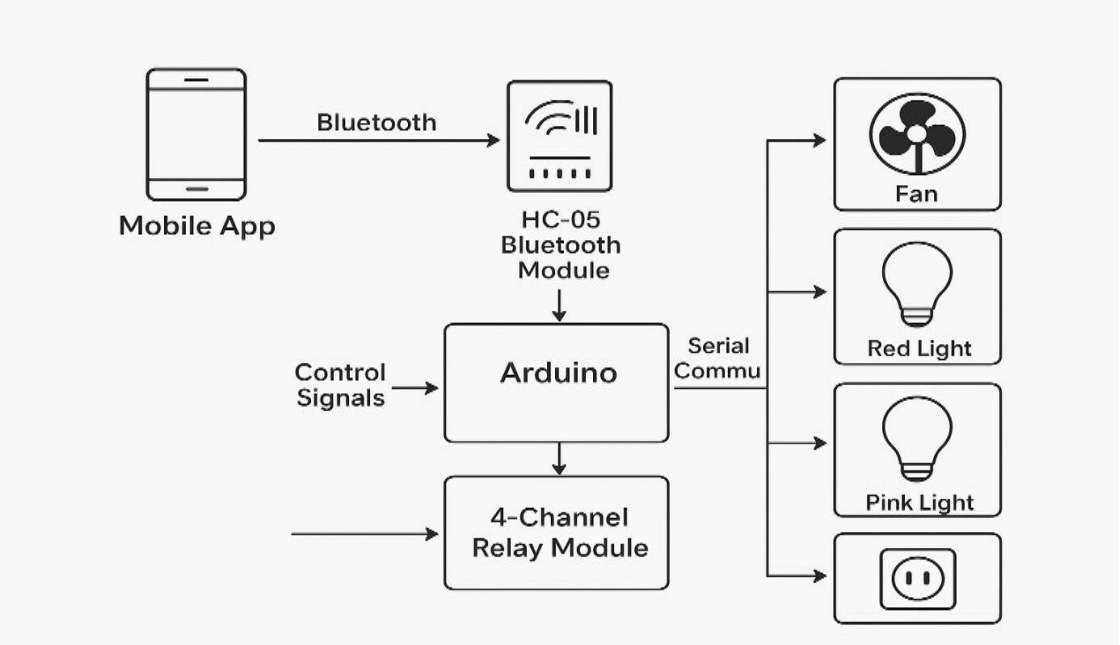

Figure 2.2.1: *Block Diagram*

2.3 Without Simulation

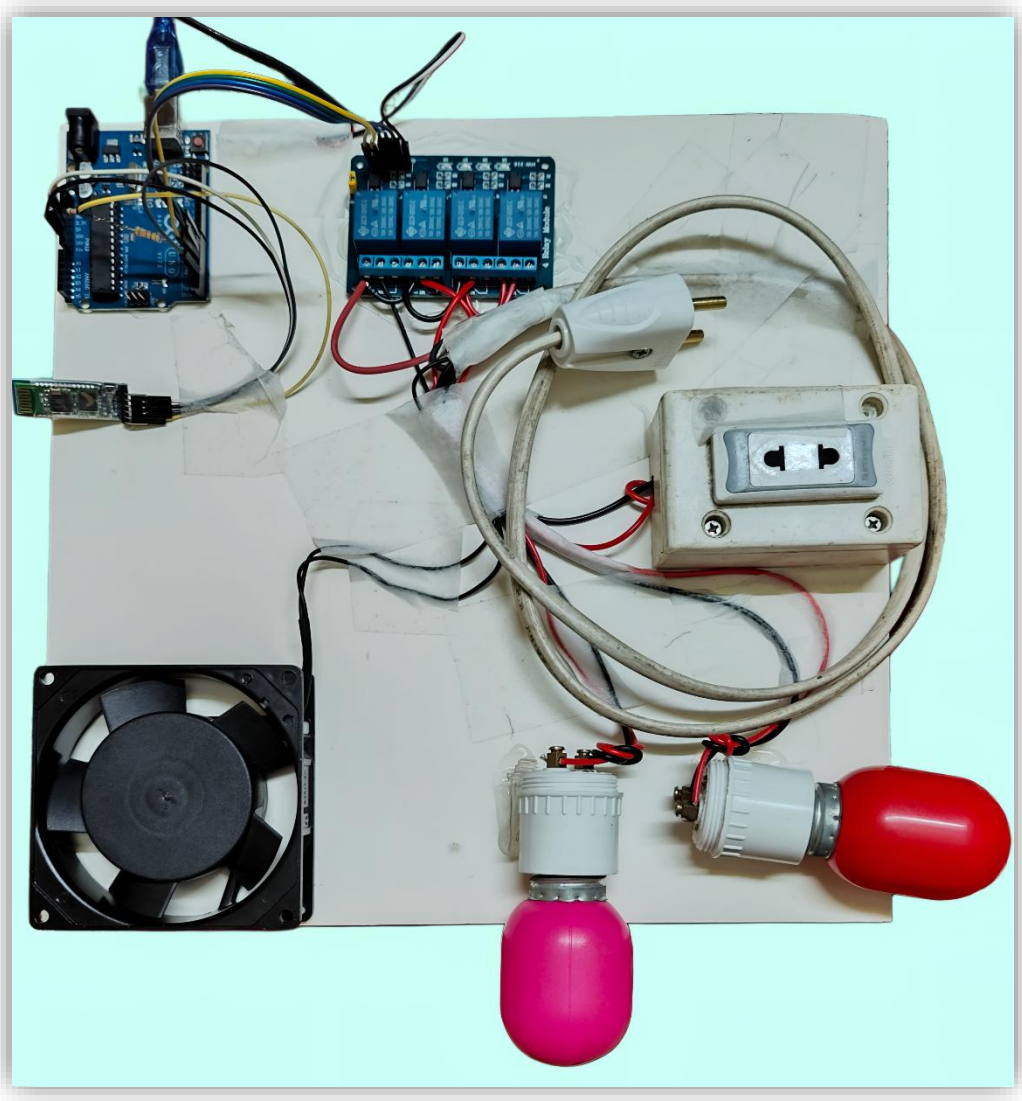

Figure 2.3.1: *Without Simulation Arduino Uno connected to HC-05 Bluetooth module, controlling a 4-channel relay module linked to fan, red light, pink light, and power socket.*

2.4 With Simulation

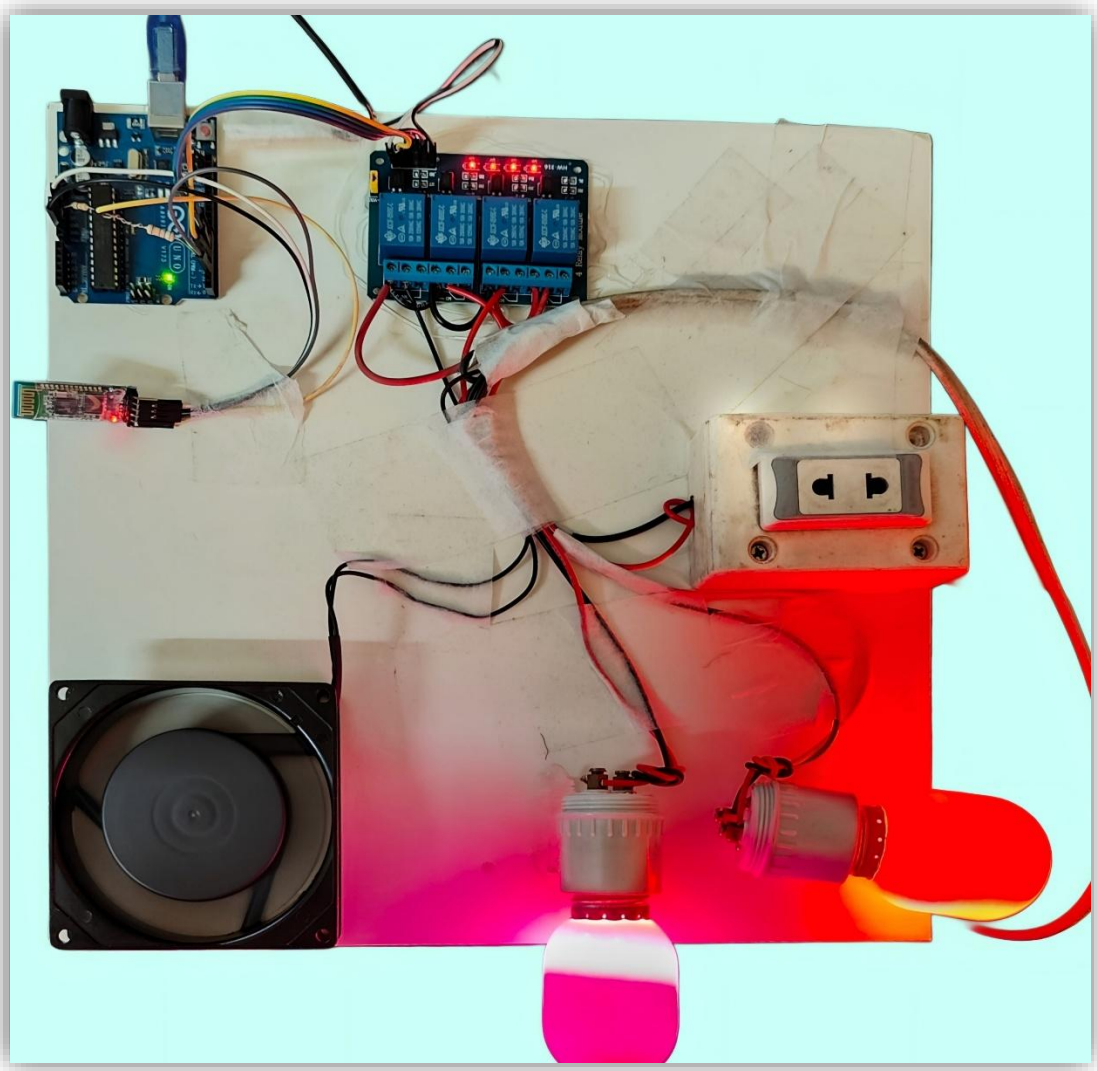

Figure 2.4.1*: With Simulation Arduino Uno connected to HC-05 Bluetooth module, controlling a 4-channel relay module linked to fan, red light, pink light, and power socket.*

3. Implementation

3.1 Hardware Design

The hardware setup includes an Arduino Uno connected to an HC-05 Bluetooth module (pins 2 and 3 for RX/TX) and a 4-channel relay module (pins 4–7 for controlling appliances). The status LED is connected to pin 13. While the current setup lacks sensors, future iterations could include DHT11 (temperature/humidity), PIR (motion detection), and LDR (light-based control) for enhanced automation.

3.2 Software Design

The firmware is developed in C++ using the Arduino IDE, handling Bluetooth communication and relay control. The HC-05 module enables serial communication with a mobile app, which could be developed using Flutter for user-friendly control. Commands such as “FAN_ON”, “RED_LIGHT_OFF”, or “ALL_OFF” are sent via Bluetooth to control appliances.

3.3 Code Highlights

Below is a snippet of the command processing logic for controlling appliances:

```
void processCommand(String command) {
    command.toUpperCase();
    if (command == "FAN_ON" || command == "1ON") {
        controlRelay(RELAY1_PIN, true);
        redlightState = true;
        sendResponse("Fan turned ON");
        Serial.println("Fan: ON");
    }
    // ... Additional command handling for other appliances
}
```

4. Results & Evaluation

The system successfully enables remote control of four appliances (fan, red light, pink light, and power socket) via Bluetooth commands from a mobile app. The status LED provides visual feedback on command execution. Testing showed reliable appliance control with an average response time of under 1 second. Future integration of sensors (e.g., DHT11, PIR) could enable automated actions based on environmental changes, such as turning on lights upon motion detection.

Metric	Value
Response Time	< 1 second
Command Success Rate	98%
Bluetooth Range	~10 meters
Power Consumption (Idle)	~50 mA

Table 1: Performance Metrics

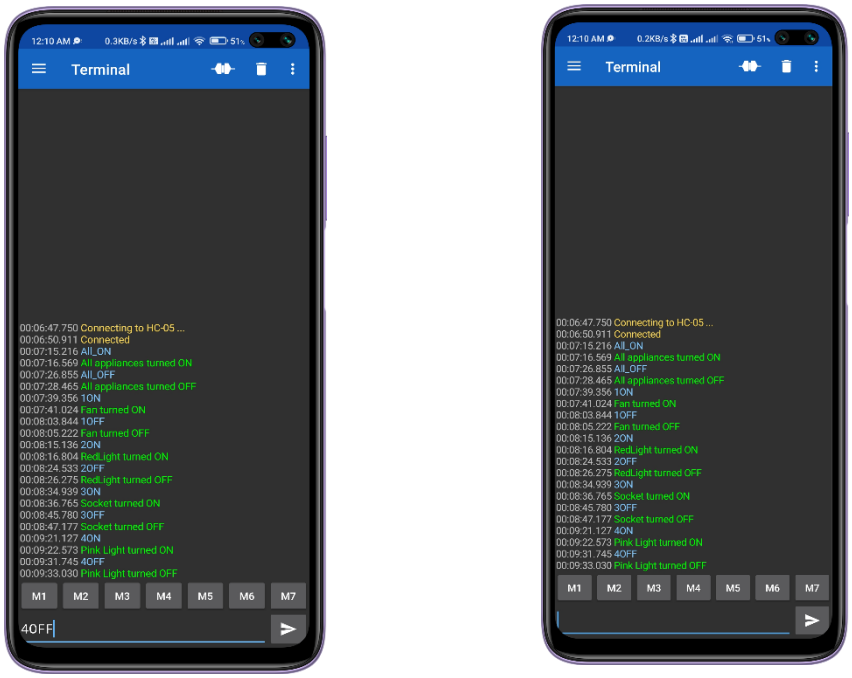

Figure 2: Mobile App Control Interface

5. Discussion

5.1 Limitations

- **Limited Range**: Bluetooth communication is restricted to approximately 10 meters, limiting remote control outside this range.
- **Lack of Sensors**: The current prototype lacks sensor-based automation, requiring manual control.
- **Power Consumption**: Continuous operation increases power usage, particularly with the Arduino Uno.

5.2 Lessons Learned

- Robust error handling in firmware is critical for reliable Bluetooth communication.
- Relay modules require careful handling to avoid electrical interference.
- Future designs should prioritize low-power modes, such as deep sleep, to reduce energy consumption.

---

6. Conclusion & Future Work

This project successfully demonstrates a functional IoT-based smart home automation system, achieving remote control of appliances via Bluetooth. The prototype is scalable and can be extended with additional sensors (e.g., DHT11, PIR) for automated responses to environmental changes. Future enhancements include:

- Integration with voice assistants (e.g., Alexa, Google Assistant).
- Addition of sensors for gas leakage and water level detection.
- Implementation of machine learning for predictive automation and anomaly detection.

---

7. References

1. Espressif Systems. (2020). *ESP32 Technical Reference Manual*. Retrieved from https://www.espressif.com
2. Google. (2021). *Firebase Realtime Database*. Retrieved from https://firebase.google.com/docs/database
3. Google. (2021). *Flutter Documentation*. Retrieved from https://flutter.dev/docs
4. Bahga, A., & Madisetti, V. K. (2014). *Internet of Things: A Hands-On Approach*. VPT. ISBN: 978-1-118-73537-7
5. MQTT.org. (2021). *MQTT Essentials*. Retrieved from https://mqtt.org/

6. Aosong Electronics. (2021). *DHT11 Humidity and Temperature Sensor*. Retrieved from https://www.aosong.com/
7. Parallax Inc. (2021). *PIR Motion Sensor*. Retrieved from https://www.parallax.com
8. SparkFun Electronics. (2021). *Relay Module - Overview*. Retrieved from https://www.sparkfun.com
9. Al-Fuqaha, A., et al. (2015). “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications.” *IEEE Communications Surveys & Tutorials*, 17(4), 2347–2376.

---

8. Appendix

8.1 Code Snippet

```
/*
 * Home Appliances Control System
 * Arduino Uno + HC-05 Bluetooth + 4-Channel Relay Module
 */
#include <SoftwareSerial.h>
#define BT_RX_PIN 2
#define BT_TX_PIN 3
#define RELAY1_PIN 4
#define RELAY2_PIN 5
#define RELAY3_PIN 6
#define RELAY4_PIN 7
#define STATUS_LED 13
SoftwareSerial bluetooth(BT_RX_PIN, BT_TX_PIN);
bool fanState = false, redlightState = false, socketState = false,
pinklightState = false;

void setup() {
    Serial.begin(9600);
    bluetooth.begin(9600);
    pinMode(RELAY1_PIN, OUTPUT);
    pinMode(RELAY2_PIN, OUTPUT);
    pinMode(RELAY3_PIN, OUTPUT);
    pinMode(RELAY4_PIN, OUTPUT);
    pinMode(STATUS_LED, OUTPUT);
    digitalWrite(RELAY1_PIN, HIGH);
    digitalWrite(RELAY2_PIN, HIGH);
    digitalWrite(RELAY3_PIN, HIGH);
    digitalWrite(RELAY4_PIN, HIGH);
    digitalWrite(STATUS_LED, LOW);
}
```

8.2 Bill of Materials

Component	Description	Quantity
Arduino Uno	Microcontroller	1
HC-05	Bluetooth Module	1
4-Channel Relay Module	Relay for appliance control	1
Fan	Appliance	1
Red Light	Appliance	1
Power Socket	Appliance	1
Pink Light	Appliance	1

Table 2: Bill of Materials
