**Bluetooth Voice Based Home Automation**

**Aim:**

To design a home automation system that allows users to control electrical appliances using voice commands sent via Bluetooth to an ESP32-C6 which activates a relay module accordingly.

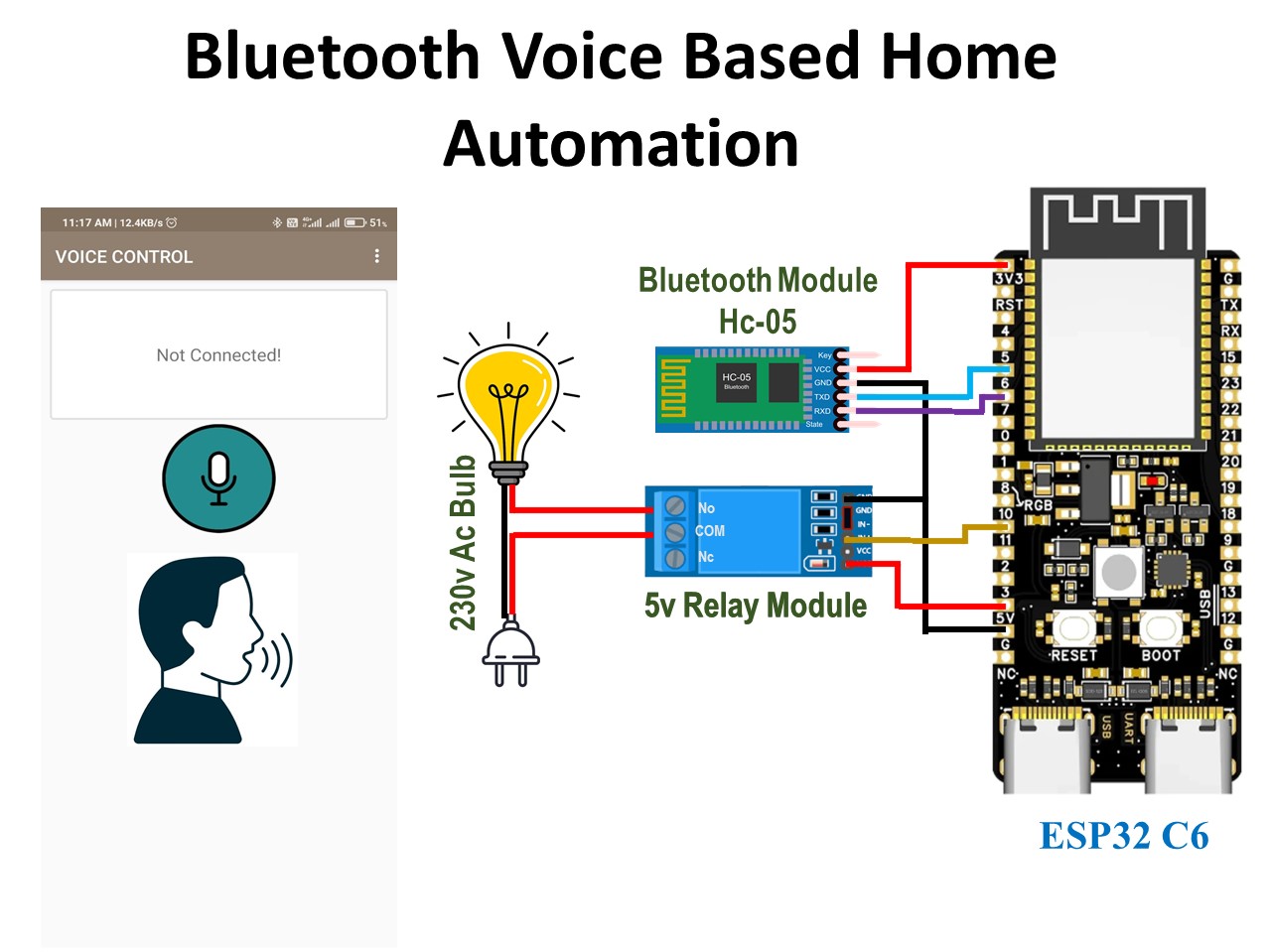
**Apparatus Required:**

|  |  |  |
| --- | --- | --- |
| **S. No** | **Component** | **Quantity** |
|  | ESP32-C6 Dev Board | 1 |
|  | Relay module (5V) | 1 |
|  | HC-05 Bluetooth module | 1 |
|  | Breadboard | 1 |
|  | Jumper Wires | As needed |
|  | USB Type-C Cable | 1 |
|  | Computer with Arduino IDE | 1 |
|  | Ac 230v Blub | 1 |
|  | Blub Holder | 1 |
|  | Wire | As needed |

**Theory:**

Bluetooth-based home automation leverages short-range wireless communication (2.4 GHz) to allow devices to be remotely controlled. In this project, voice commands are converted into text by a mobile app and transmitted via Bluetooth. The ESP32-C6 reads these commands via its second UART interface (Serial2), then processes them to switch a relay ON/OFF, which in turn controls appliances.

**Circuit Diagram:**

****

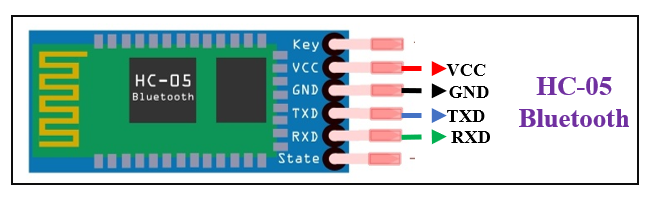
**Procedure:**

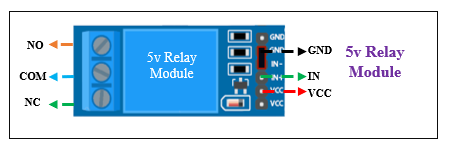
1. Connect the Bluetooth module (HC-05/HC-06) to ESP32-C6 using UART2 (GPIO6 as RX, GPIO7 as TX).
2. Connect a relay module to GPIO10 through a transistor circuit (if needed) or directly if the relay is logic-compatible.
3. Upload the below code to ESP32-C6 via USB.
4. Install a Bluetooth voice command app on your phone (e.g., “Bluetooth Voice Control”).
5. Pair your phone with the Bluetooth module.
6. Use voice commands like "turn on" or "turn off".
7. The ESP32-C6 reads the command and toggles the relay accordingly.

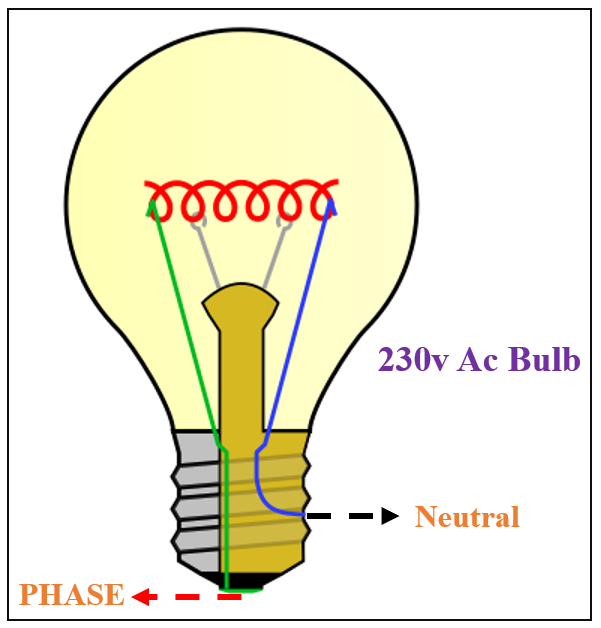
**Connection Pin:**

|  |  |
| --- | --- |
| **Component** | **GPIO** |
| Bluetooth RX | 7 |
| Bluetooth TX | 6 |
| Relay Module IN | 10 |
| VCC (Relay) | 5v |
| GND | GND |

**Pinout:**

****

****

****

**Code:**

#include <HardwareSerial.h>

#define RXD2 6

#define TXD2 5

#define LED\_PIN 10

HardwareSerial BTSerial(1); // Use UART1 (you can name it anything)

void setup() {

  Serial.begin(115200); // USB serial monitor

  BTSerial.begin(9600, SERIAL\_8N1, RXD2, TXD2); // Bluetooth serial

  pinMode(LED\_PIN, OUTPUT);

  digitalWrite(LED\_PIN, LOW);

  Serial.println("Bluetooth LED Control Ready");

}

void loop() {

  if (BTSerial.available()) {

    String cmd = BTSerial.readStringUntil('\n');

    cmd.trim();

    Serial.print("Received: ");

    Serial.println(cmd);

    if (cmd == "turn on") {

      digitalWrite(LED\_PIN, HIGH);

      BTSerial.println("turn on");

    } else if (cmd == "turn off") {

      digitalWrite(LED\_PIN, LOW);

      BTSerial.println("turn off");

    } else {

      BTSerial.println("Unknown command");

    }

  }

}