28-04-2024

BRAC UNIVERSITY

Inspiring Excellence

PROJECT REPORT

Group Members Name	Student ID
MD. ADNAN KHAN	21301286
TANAY PAUL	21301438

Name of the Project:

Home Automation System

Introduction:

The **Home Automation System** project integrates both IoT and manual control mechanisms to manage home appliances remotely. It utilizes an ESP-8266 or NodeMCU along with various components to achieve automation and manual switch control. This report provides a comprehensive overview of the project, including its applications, components, working principle, circuit diagram, setup, code, data flow, and conclusion.

Applications:

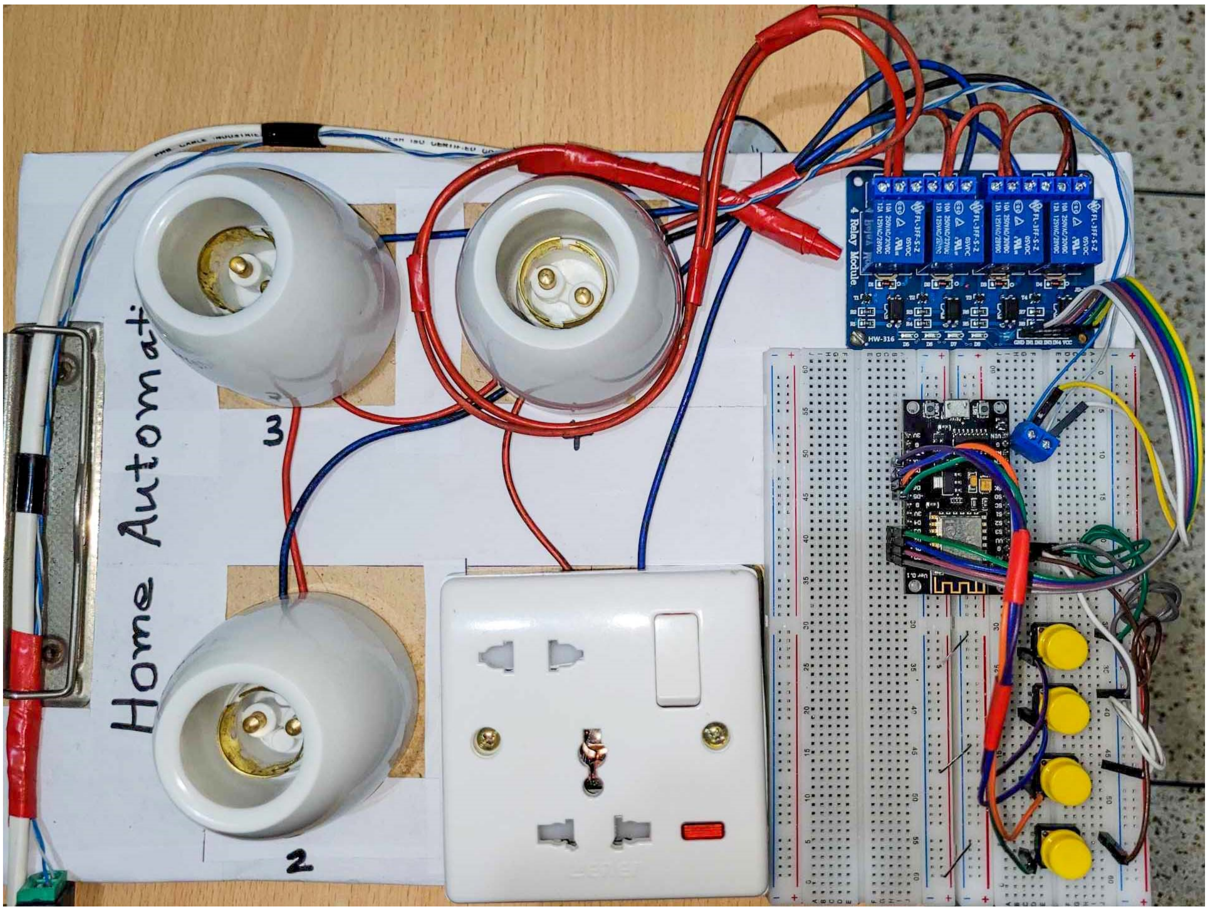
- Home automation for enhanced convenience and energy efficiency.
- Remote control of home appliances, providing convenience and security.
- Integration with IoT platforms for smart home management.

Components:

The project utilizes the following components:

- Solderless Breadboard
- ESP-8266 (NodeMCU)
- Push Buttons * 4
- 4-Channel 5V Relay Module
- Male to Male Jumper Wires
- Male to Female Jumper Wires
- Bulb Holders * 3
- 220V LED Bulbs * 3
- Multi Socket Board
- 5V 2A Power Adapter

Picture:



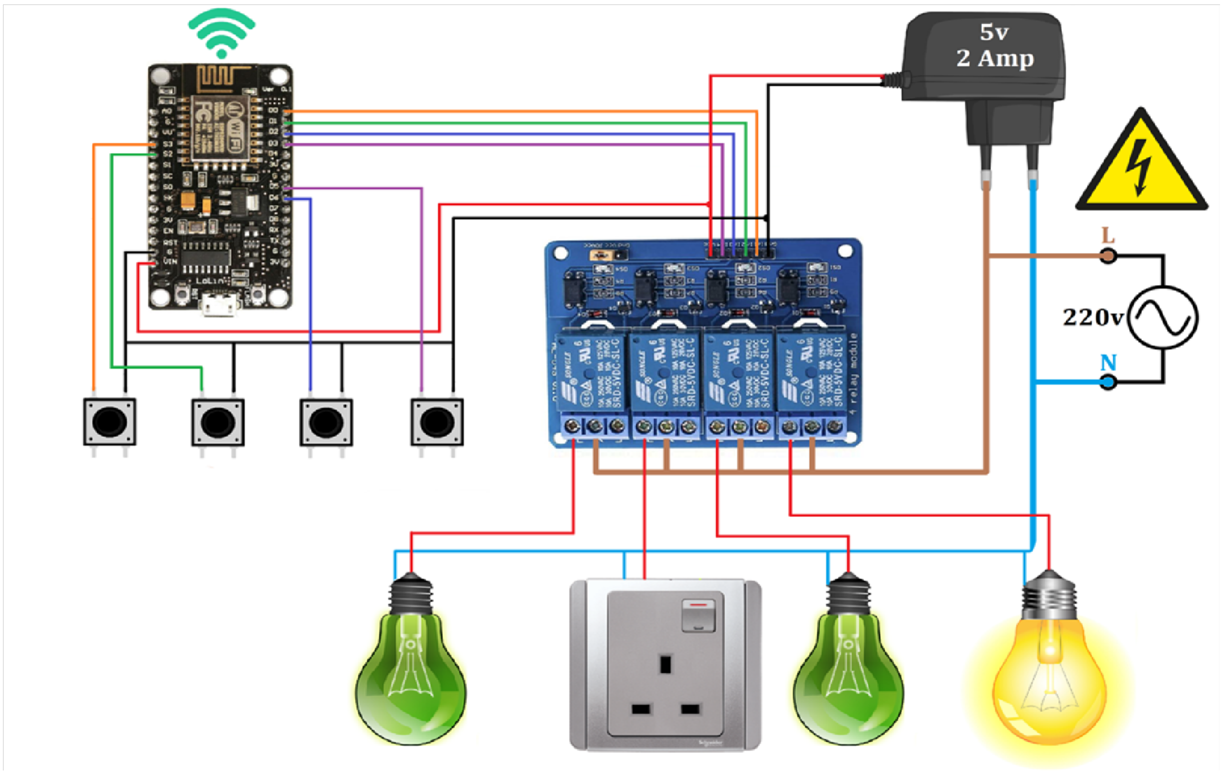
Working Principle:

The Home Automation System operates based on the following working principles:

- **Manual Control-** Push buttons are utilized for manual control of appliances. Pressing a button toggles the state of the corresponding relay, thereby turning the associated appliance on or off. This method does not require any internet connection.

- **IoT Control-** The system is integrated with the Blynk platform for IoT control. Through the Blynk app, users can current device state, remotely toggle the relays to enable or disable the connected appliances from anywhere with internet connectivity.

Circuit Diagram:



Circuit Setup:

1. Firstly, we connected the ESP-8266 module to the breadboard.
2. Then we connected the 4 channel Relay module to the board.
3. After that we added the push buttons to the board.
4. Then we connected the LED holders, Multi Socket to the relay module.
5. After adding all the components we did the wiring according to our circuit diagram.
5. Finally, we powered the whole circuit with a 5V 2Amp power adapter and also connected the relay with 220v AC line.

Code:

```
#define BLYNK_TEMPLATE_ID "TMPL6wzEpd_58"
#define BLYNK_TEMPLATE_NAME "Home Automation"
#define BLYNK_AUTH_TOKEN "dFV42nBGQlDPPW6-8i08sR-k371A0ZSD"
```

```
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "No Internet!";
char pass[] = "error404!";

// Define the push button pins
#define BUTTON1_PIN D5
#define BUTTON2_PIN D6
#define BUTTON3_PIN D7
#define BUTTON4_PIN D9

// Initialize the button states
int button1State = 0; int
button2State = 0; int
button3State = 0; int
button4State = 0;

// Initialize the last button states
int lastButton1State = HIGH; int
lastButton2State = HIGH; int
lastButton3State = HIGH; int
lastButton4State = HIGH;
BLYNK_WRITE(V0)
{   int value =
param.asInt();
Serial.println(value);
  if(value == 1)
  {
    digitalWrite(D0, LOW);
Serial.println("Relay 1 ON");
    Blynk.virtualWrite(V0, 255); // Update status in Blynk app
  }   if(value
== 0)
  {
     digitalWrite(D0, HIGH);
Serial.println("Relay 1 OFF");
     Blynk.virtualWrite(V0, 0); // Update status in Blynk app
  }
}
```

```
BLYNK_WRITE(V1)
{   int value =
param.asInt();
Serial.println(value);
  if(value == 1)
  {
    digitalWrite(D1, LOW);
Serial.println("Relay 2 ON");
    Blynk.virtualWrite(V1, 255); // Update status in Blynk app
  }   if(value
== 0)
  {
     digitalWrite(D1, HIGH);
Serial.println("Relay 2 OFF");
     Blynk.virtualWrite(V1, 0); // Update status in Blynk app
  }
}

BLYNK_WRITE(V2)
{   int value =
param.asInt();
Serial.println(value);
  if(value == 1)
  {
    digitalWrite(D2, LOW);
Serial.println("Relay 3 ON");
    Blynk.virtualWrite(V2, 255); // Update status in Blynk app
  }   if(value
== 0)
  {
     digitalWrite(D2, HIGH);
Serial.println("Relay 3 OFF");
     Blynk.virtualWrite(V2, 0); // Update status in Blynk app
  }
}

BLYNK_WRITE(V3)
{   int value =
param.asInt();
Serial.println(value);
  if(value == 1)
  {
    digitalWrite(D3, LOW);
    Serial.println("Relay 4 ON");
    Blynk.virtualWrite(V3, 255); // Update status in Blynk app
  }   if(value
== 0)
  {
     digitalWrite(D3, HIGH);
Serial.println("Relay 4 OFF");
     Blynk.virtualWrite(V3, 0); // Update status in Blynk app
  } } void setup() {
Serial.begin(115200);
  Blynk.begin(auth, ssid, pass);

  // Initialize relay pins to OFF state
digitalWrite(D0, HIGH);   digitalWrite(D1,
HIGH);   digitalWrite(D2, HIGH);
digitalWrite(D3, HIGH);

  pinMode(D0, OUTPUT);
pinMode(D1, OUTPUT);   pinMode(D2,
OUTPUT);   pinMode(D3, OUTPUT);

  // Initialize the button pins as inputs with internal pull-up resistors
  pinMode(BUTTON1_PIN, INPUT_PULLUP);
pinMode(BUTTON2_PIN, INPUT_PULLUP);   pinMode(BUTTON3_PIN,
INPUT_PULLUP);   pinMode(BUTTON4_PIN, INPUT_PULLUP);
} void loop()
{
Blynk.run();

  // Read the state of the buttons   button1State
= digitalRead(BUTTON1_PIN);   button2State
= digitalRead(BUTTON2_PIN);   button3State
= digitalRead(BUTTON3_PIN);   button4State
= digitalRead(BUTTON4_PIN);

  // Check if button1 state has changed
if (button1State != lastButton1State) {
if (button1State == LOW) {
      // Toggle relay1
digitalWrite(D0, !digitalRead(D0));
      // Update status in Blynk app
      Blynk.virtualWrite(V0, digitalRead(D0) == LOW ? 255 : 0);
    }
    // Delay a little bit to avoid bouncing
delay(50);
  }
  // Check if button2 state has changed
if (button2State != lastButton2State) {
if (button2State == LOW) {       //
Toggle relay2       digitalWrite(D1,
!digitalRead(D1));
      // Update status in Blynk app
      Blynk.virtualWrite(V1, digitalRead(D1) == LOW ? 255 : 0);
    }
    // Delay a little bit to avoid bouncing
delay(50);
  }
  // Check if button3 state has changed
if (button3State != lastButton3State) {
if (button3State == LOW) {       //
Toggle relay3       digitalWrite(D2,
!digitalRead(D2));
      // Update status in Blynk app
      Blynk.virtualWrite(V2, digitalRead(D2) == LOW ? 255 : 0);
    }
    // Delay a little bit to avoid bouncing
delay(50);
  }
  // Check if button4 state has changed
if (button4State != lastButton4State) {
if (button4State == LOW) {       //
Toggle relay4       digitalWrite(D3,
!digitalRead(D3));
      // Update status in Blynk app
      Blynk.virtualWrite(V3, digitalRead(D3) == LOW ? 255 : 0);
    }
    // Delay a little bit to avoid bouncing
delay(50);
  }
  // Update the last button states
lastButton1State = button1State;
lastButton2State = button2State;
lastButton3State = button3State;
lastButton4State = button4State;
}
```

Data Flow:

- User input is received either through manual button presses or via the Blynk app.
- Button presses are detected by the ESP-8266 module's GPIO pins.
- Based on the button pressed, the corresponding relay is toggled.
- Relay state changes are reflected in the connected appliances, turning them ON or OFF.

Conclusion:

The Home Automation System project successfully demonstrates the integration of IoT and manual control mechanisms for Home Automation. By leveraging the NodeMCU and Blynk platform, users can conveniently manage their home appliances remotely. The project showcases the potential of IoT in enhancing everyday tasks and improving energy efficiency.

Project Video:

https://www.youtube.com/watch?v=dXv8TOY6WUU