Controlling LED Using Micro controller and Web Server

Submitted By,

1.Danush Gupta (2019503012)

2.Jagadish (2019503018)

3.Premal Raj Vellaisamy (2019503032)

4.Sakthi Dasan (2019503555)

5.Sham Ganesh (2019503559)

Contents

Abstract

Technology is a never ending process. So to make our lives easier home automation can be developed. The aim of this project is to design and implement a low cost and flexible wireless home automation system.

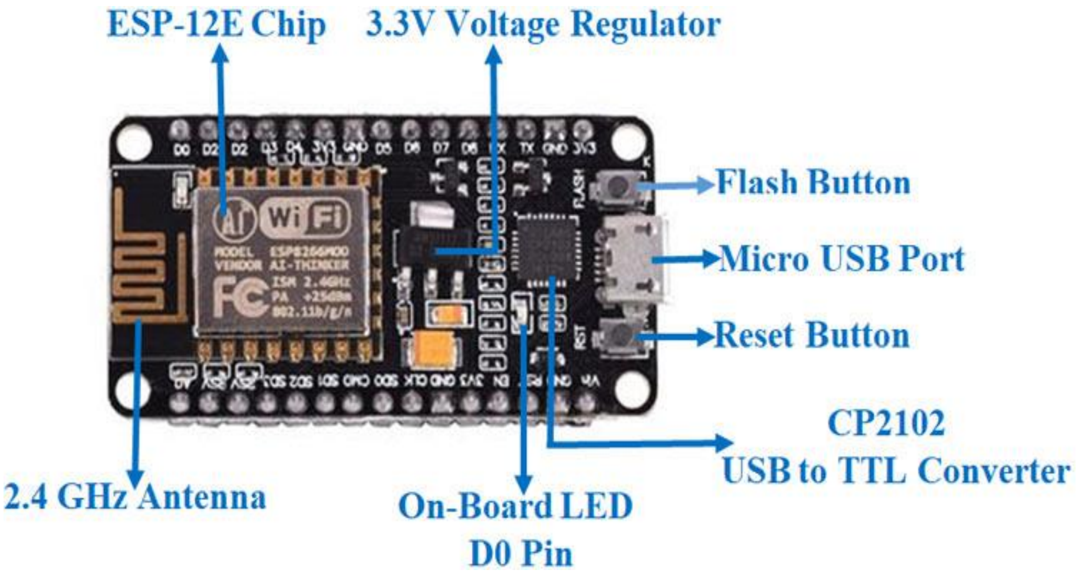
The basic aim of the project is to control the LED in the microcontroller which is the basic(first step) for home automation.

We are going to demonstrate 3 different tasks in order to understand and build home automation from the scratch:

- ➢ Making the built-in LED in the microcontroller to blink.
- ➢ Controlling the LED based on the light intensity in the room.
- ➢ Controlling the built-in LED using a web server.

Hardware Used

- NodeMCU (Microcontroller)



- LDR (Light Dependent Resistor)
- Jumper Wires
- USB Cable
- Smart Phone
- Laptop

Software Used

- Arduino C

What is a Microcontroller?

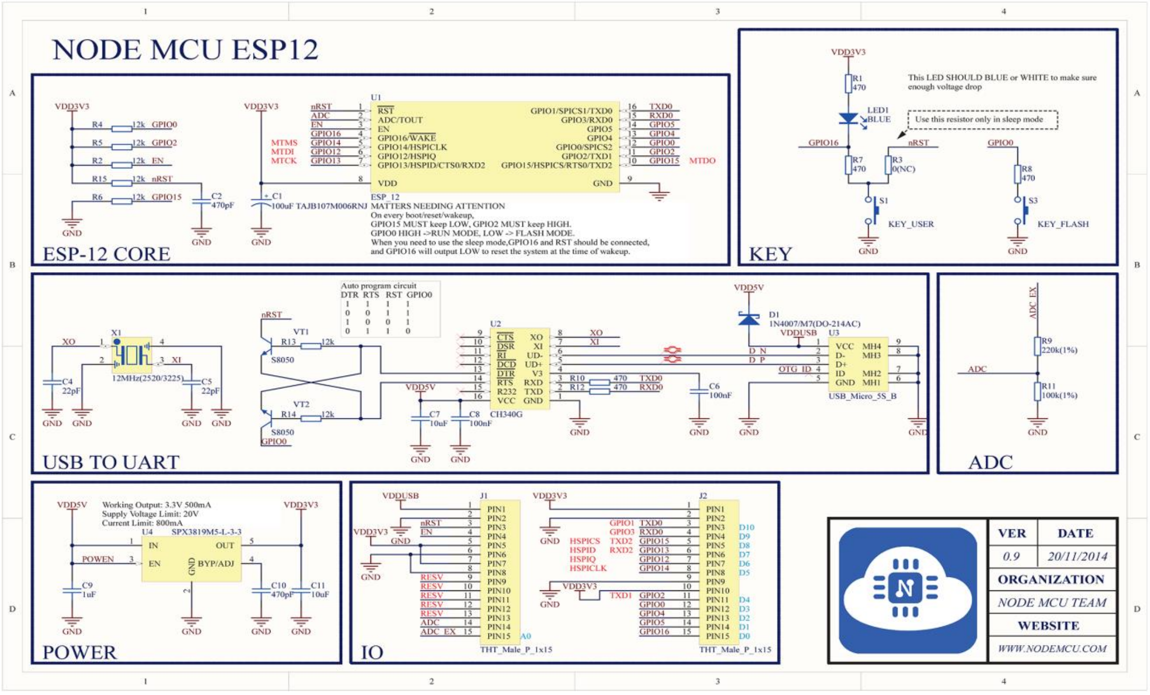
A microcontroller is a compact integrated circuit designed to govern a specific operation in an embedded system. A typical microcontroller includes a processor, memory and input/output (I/O) peripherals on a single chip.The temporary information that the microcontroller receives is stored in its data memory, where the processor accesses it and uses instructions stored in its program memory to decipher and apply the incoming data. It then uses its I/O peripherals to communicate and enact the appropriate action.

Specifications of NodeMCU

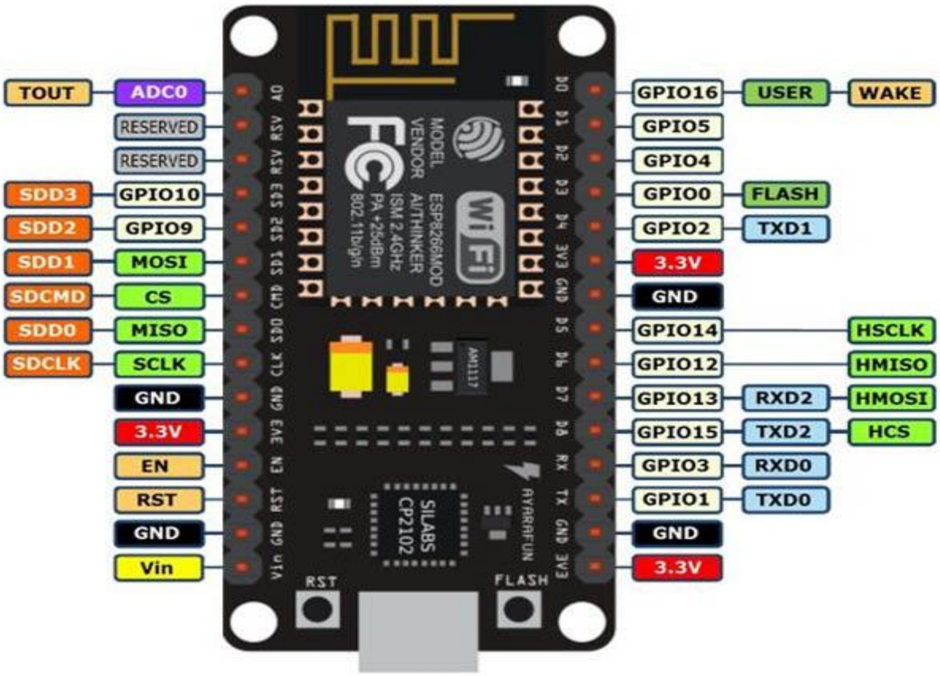
- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB
- SRAM: 64 KB
- Clock Speed: 80 MHz
- PCB Antenna
- Small Sized module to fit smartly inside your IoT projects

Diagram

Circuit Diagram


NODE MCU ESP12
ESP-12 CORE
KEY
USB TO UART
ADC
POWER
IO
VDD3V3
VDD5V
VDDUSB
GND
U1
ESP_12
This LED SHOULD BLUE or WHITE to make sure enough voltage drop
Use this resistor only in sleep mode
KEY_USER
KEY_FLASH
LED1 BLUE
MATTERS NEEDING ATTENTION
On every boot/reset/wakeup,
GPIO15 MUST keep LOW, GPIO2 MUST keep HIGH.
GPIO0 HIGH ->RUN MODE, LOW -> FLASH MODE.
When you need to use the sleep mode,GPIO16 and RST should be connected,
and GPIO16 will output LOW to reset the system at the time of wakeup.
Auto program circuit
DTR RTS RST GPIO0
12MHz(2520/3225)
S8050
CH340G
1N4007/M7(DO-214AC)
USB_Micro_5S_B
Working Output: 3.3V 500mA
Supply Voltage Limit: 20V
Current Limit: 800mA
SPX3819M5-L-3-3
THT_Male_P_1x15
VER
DATE
0.9
20/11/2014
ORGANIZATION
NODE MCU TEAM
WEBSITE
WWW.NODEMCU.COM

Pin Diagram



About Project

- Task 1:Making the built-in LED in microcontroller blink:

Source Code:

```
void setup() {
  // put your setup code here, to run once:
pinMode(LED_BUILTIN,OUTPUT);
Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
digitalWrite(LED_BUILTIN,LOW);
Serial.println("LED is on");
delay(1500);
digitalWrite(LED_BUILTIN,HIGH);
Serial.println("LED is off");
delay(1500);
}
```

- **Applications:**
  - Used in indicators
  - Can be used for any visual sign indication in highways
  - Also used as a flashing beacon

➢ Task 2: Controlling the LED based on the light intensity in the room:

Source Code:

```
void setup() {
  // put your setup code here, to run once:
Serial.begin(115200);
pinMode(A0,INPUT);
pinMode(LED_BUILTIN,OUTPUT);
}
void loop() {
  // put your main code here, to run repeatedly:
int a;
  a=analogRead(A0);
delay(200);
Serial.print(a);
if(a<40)
  {
digitalWrite(LED_BUILTIN,LOW);
Serial.println(" => LED is on");
  }
else
  {
digitalWrite(LED_BUILTIN,HIGH);
Serial.println(" => LED is off");
  }
}
```

❖ **Applications:**

- Used to implement automatic street lights.
- Used in mobile phones for auto brightness mode
- Also used in camera shutter control and flash control
- It can also be implemented in alarm clocks
- Mainly used in light intensity meters and burglar alarm circuits

➢ Task 3: Controlling the built-in LED using a web server:

Source Code:

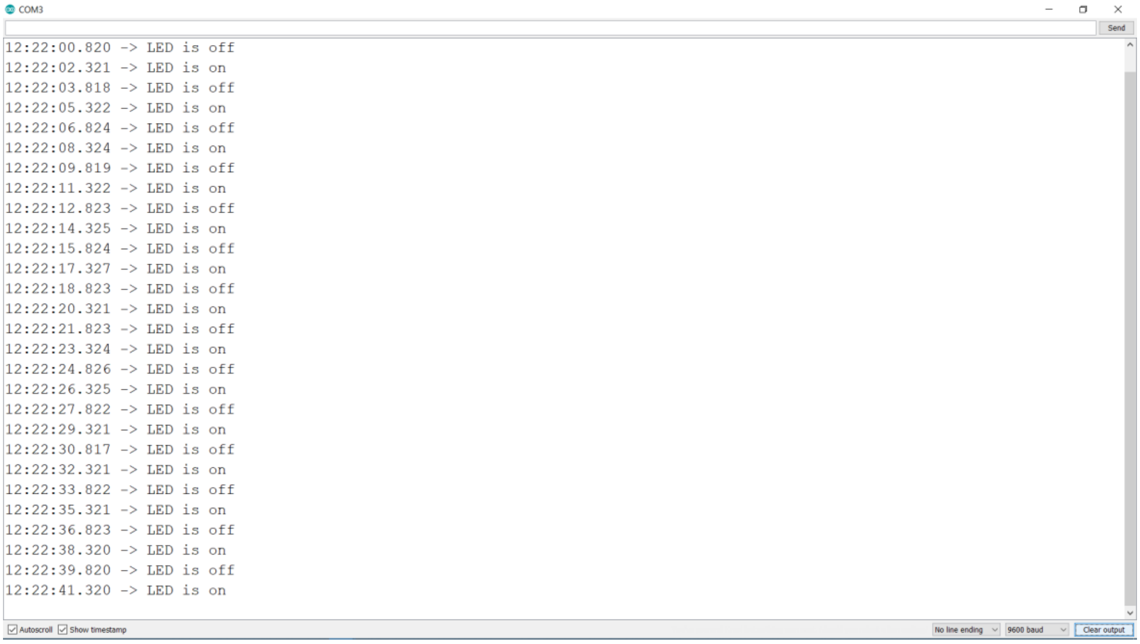
```
#include<ESP8266WebServer.h>

#define username "project"

#define password "12345678"

ESP8266WebServer server;

String ledwebpage="<!DOCTYPE html><html><head><title>LED Controller</title><h1><center>LED Controller</center></h1></head><body><form><center><button type=\"submit\" name=\"state\" value=\"0\"><b>LED ON</b><br><br></button><button type=\"submit\" name=\"state\" value=\"1\"><b>LED OFF</b></button></center></form></body></html>";


void setup() {

  // put your setup code here, to run once:

Serial.begin(115200);

pinMode(LED_BUILTIN,OUTPUT);

WiFi.softAP(username,password);
Serial.println(" ");
Serial.println("IP Address ");
Serial.println("||");
Serial.println("\\/");
Serial.println(WiFi.softAPIP());
server.begin();
server.on("/led",led);
}
void loop() {
  // put your main code here, to run repeatedly:
server.handleClient();
}
void led(){
server.send(200,"text/html",ledwebpage);
if((server.arg("state")=="0"))
  {
digitalWrite(LED_BUILTIN,LOW);
  }
else
  {
digitalWrite(LED_BUILTIN,HIGH);
  }
}
```

❖ **Applications:**

- Can control any appliance from anywhere around the globe

Experimental Output

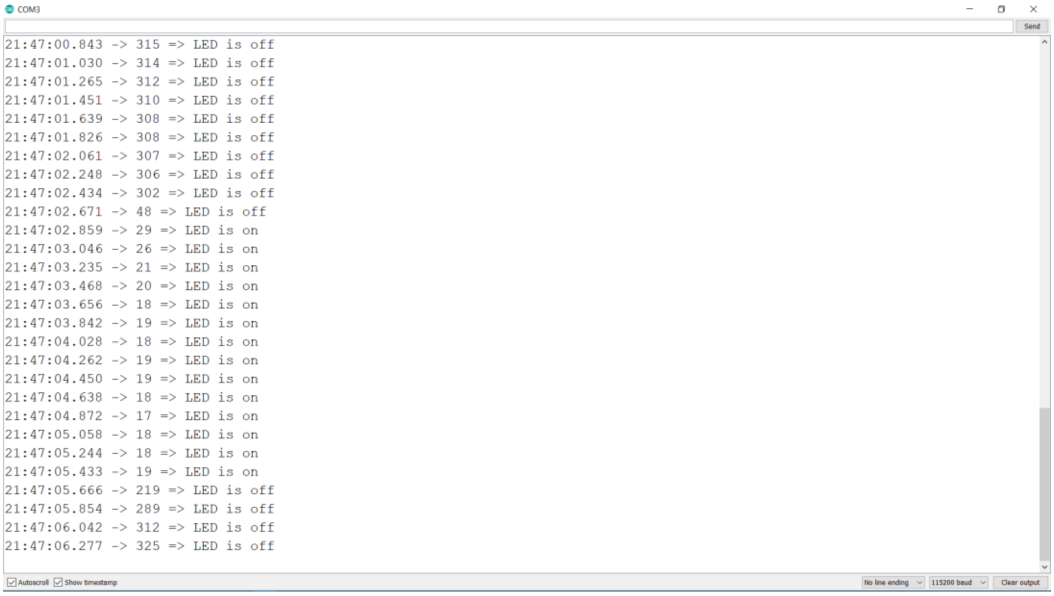
- Task 1:Making the built-in LED in microcontroller blink:



```
12:22:00.820 -> LED is off
12:22:02.321 -> LED is on
12:22:03.818 -> LED is off
12:22:05.322 -> LED is on
12:22:06.824 -> LED is off
12:22:08.324 -> LED is on
12:22:09.819 -> LED is off
12:22:11.322 -> LED is on
12:22:12.823 -> LED is off
12:22:14.325 -> LED is on
12:22:15.824 -> LED is off
12:22:17.327 -> LED is on
12:22:18.823 -> LED is off
12:22:20.321 -> LED is on
12:22:21.823 -> LED is off
12:22:23.324 -> LED is on
12:22:24.826 -> LED is off
12:22:26.325 -> LED is on
12:22:27.822 -> LED is off
12:22:29.321 -> LED is on
12:22:30.817 -> LED is off
12:22:32.321 -> LED is on
12:22:33.822 -> LED is off
12:22:35.321 -> LED is on
12:22:36.823 -> LED is off
12:22:38.320 -> LED is on
12:22:39.820 -> LED is off
12:22:41.320 -> LED is on
```

☑ Autoscroll ☑ Show timestamp No line ending 9600 baud

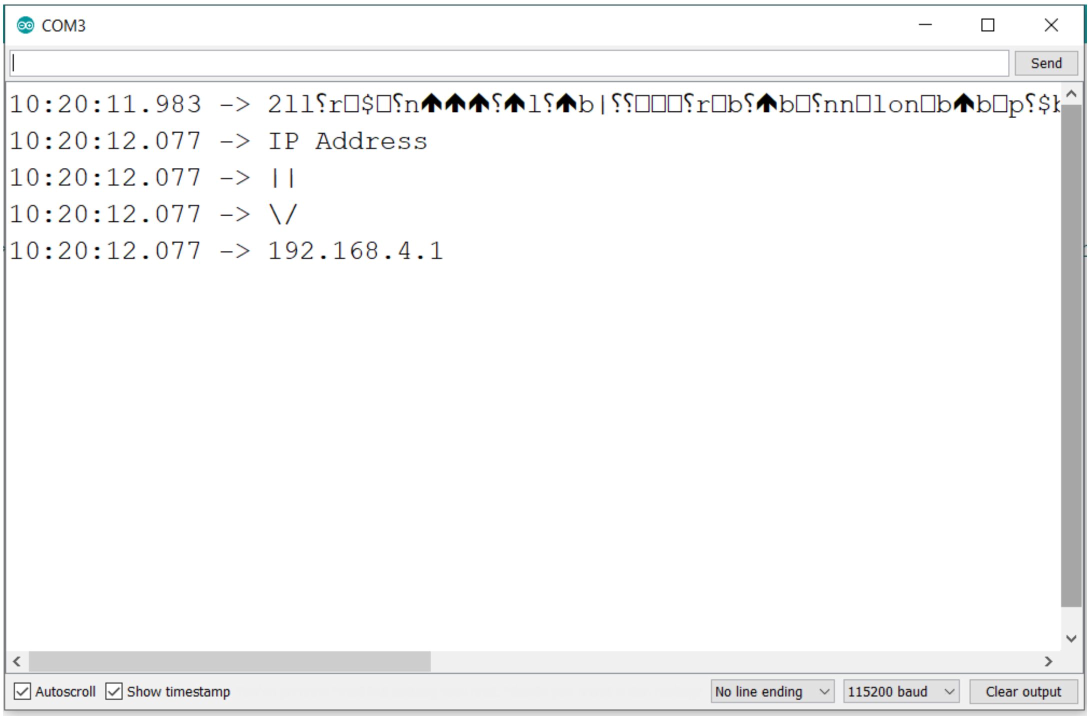
- Task 2: Controlling the LED based on the light intensity in the room:



```
21:47:00.843 -> 315 => LED is off
21:47:01.030 -> 314 => LED is off
21:47:01.265 -> 312 => LED is off
21:47:01.451 -> 310 => LED is off
21:47:01.639 -> 308 => LED is off
21:47:01.826 -> 308 => LED is off
21:47:02.061 -> 307 => LED is off
21:47:02.248 -> 306 => LED is off
21:47:02.434 -> 302 => LED is off
21:47:02.671 -> 48 => LED is off
21:47:02.859 -> 29 => LED is on
21:47:03.046 -> 26 => LED is on
21:47:03.235 -> 21 => LED is on
21:47:03.468 -> 20 => LED is on
21:47:03.656 -> 18 => LED is on
21:47:03.842 -> 19 => LED is on
21:47:04.028 -> 18 => LED is on
21:47:04.262 -> 19 => LED is on
21:47:04.450 -> 19 => LED is on
21:47:04.638 -> 18 => LED is on
21:47:04.872 -> 17 => LED is on
21:47:05.058 -> 18 => LED is on
21:47:05.244 -> 18 => LED is on
21:47:05.433 -> 19 => LED is on
21:47:05.666 -> 219 => LED is off
21:47:05.854 -> 289 => LED is off
21:47:06.042 -> 312 => LED is off
21:47:06.277 -> 325 => LED is off
```

☑ Autoscroll ☑ Show timestamp No line ending 115200 baud

➢ Task 3: Controlling the built-in LED using a web server:

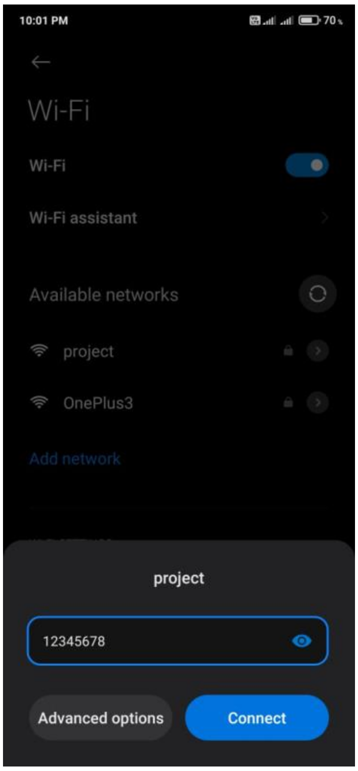


```
10:20:11.983 -> 211ʕr□$□ʕn🡅🡅🡅ʕ🡅1ʕ🡅b|ʕʕ□□□ʕr□bʕ🡅b□ʕnn□lon□b🡅b□pʕ$k
10:20:12.077 -> IP Address
10:20:12.077 -> ||
10:20:12.077 -> \/
10:20:12.077 -> 192.168.4.1
```

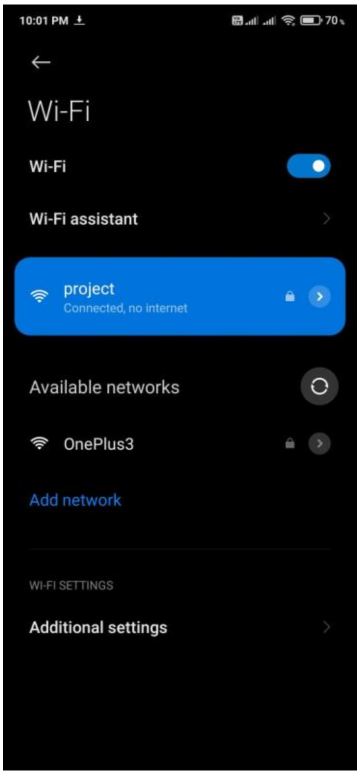
☑ Autoscroll ☑ Show timestamp

No line ending

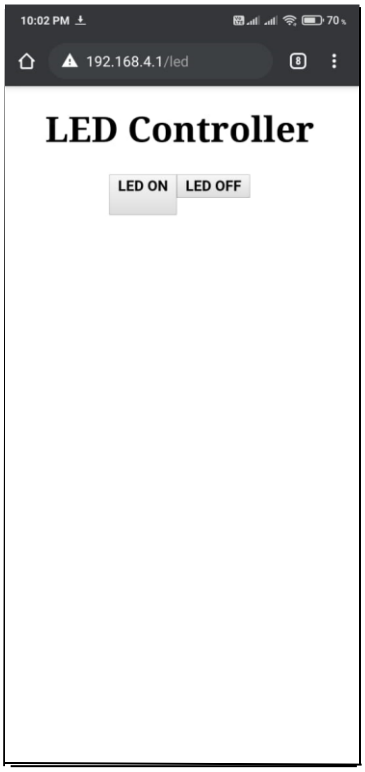
115200 baud



12345678



project
Connected, no internet



Conclusion

Starting from the basics of home automation, we have shown the design and implementation of a low cost, flexible and wireless solution to home automation. The system is also secure for access as users are expected to know the password for pairing. It also conserves energy since the appliances are turned on only when needed. This system can be used a test bed for any appliance that requires on-off switching applications.

References

1. https://internetofthingsagenda.techtarget.com/definition/microcontroller
2. https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet