Assignment-5

ELP - 720 Telecommunication Networks Laboratory

Sudhanshu Chaudhary

2019JTM2207
2020-21

A report presented for the assignment on

ESP32 with Arduino IDE

Bharti School Of

Telecommunication Technology and Management

IIT Delhi

India

February 11, 2020

Contents

1 Problem Statement 1 **3**

1.1 Problem Statement 3

1.2 Assumptions 3

1.3 Algorithm and Implementation 4

1.4 Program Structure 5

1.5 Screenshots 6

2 Appendix **10**

2.1 Appendix-A : Code 10

List of Figures

1 Google Assistant Turn on LED 6
2 Google Assistant Turn off LED 6
3 Serial Monitor WIFI connection 7
4 Temperature Alert mail 7
5 Blynk App interface 8
6 Blynk App interface temperature sensor 9

1 Problem Statement 1

1.1 Problem Statement

The concept of Smart Homes has made our living spaces more interactive and responsive to the needs of users. The Home automation concept covers a wide range of functionality right from operating your lightnings, your home security, easy opening of your garage doors, instant availability of coffee as and when you need it and almost everything that comes across your daily activities.
Let us design our own Home Automation device using Mobile App and Google Assistant!

Your Home automation app should be designed in such a way, that you can control two electrical appliances, monitor the temperature of your home and monitor the power consumed by these electrical appliances:

- Connect your ESP32 to IITD WIFI
- You have limited "Energy" in the Blynk app, so use it carefully according to your assignment.
- The app must perform the following things:
  - Control 2 appliances using Buttons and Google Assistant and read the status of appliances using led widget on the app. Note: Use LEDs instead of electrical appliances and led widget shows the status of electrical appliances not the button status
  - Display the reading of the temperature (in) **Note:** Use touch sensor as the temperature sensor is not available
  - Display the total power consumed by these appliances (consider one LED is consuming 9-11 watt of power while the other is consuming 19-21 watt of power)
  - Calculate the total power consumption cost (convert to watt-hour for this) Note: 1 unit = 1 watt-hour and each unit costs Rs. 100 (very expensive)
  - If temperature value is greater than the threshold value, you get email alerting you regarding the rise in temperature. Note: Choose threshold value using a slider in the app (slider makes the threshold value variable in the app)

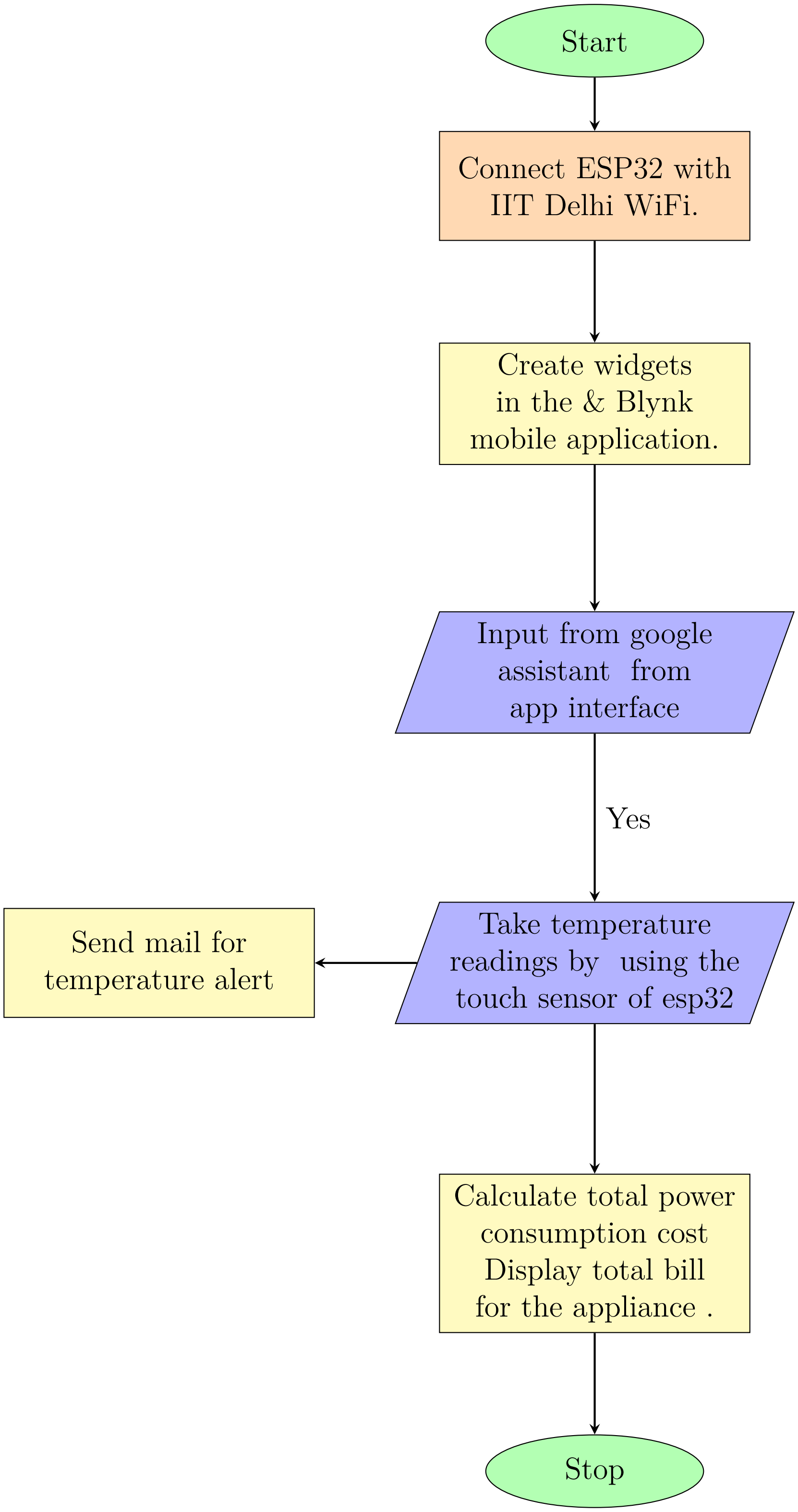
1.2 Assumptions

- Power values of appliance random number between 9 to 22.
- Touch sensor used in place of temperature sensor.

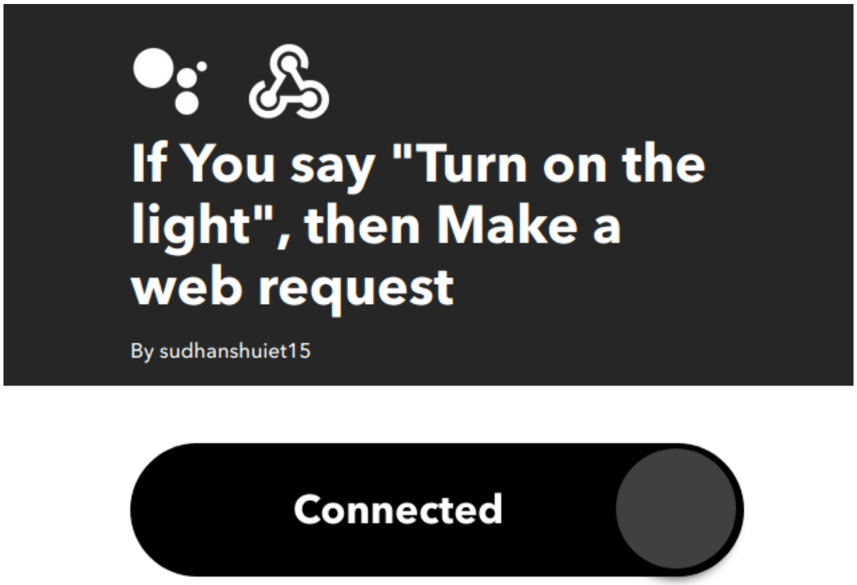
1.3 Algorithm and Implementation

- Connect ESP32 with IIT Delhi WiFi.
- Create widgets in the Blynk mobile application.
- Connect google assistant of mobile with Blynk using IFTTT .
- Display status of the appliances by using led's in the app.
- Take temperature readings by using the touch sensor of esp32.
- If temperature exceeds a certain threshold, send a alert mail.
- Calculate the time for which any appliance was left ON.
- Calculate total power consumption cost.
- Display total bill for the appliance .

1.4 Program Structure


Start
Connect ESP32 with IIT Delhi WiFi.
Create widgets in the & Blynk mobile application.
Input from google assistant from app interface
Yes
Take temperature readings by using the touch sensor of esp32
Send mail for temperature alert
Calculate total power consumption cost Display total bill for the appliance .
Stop

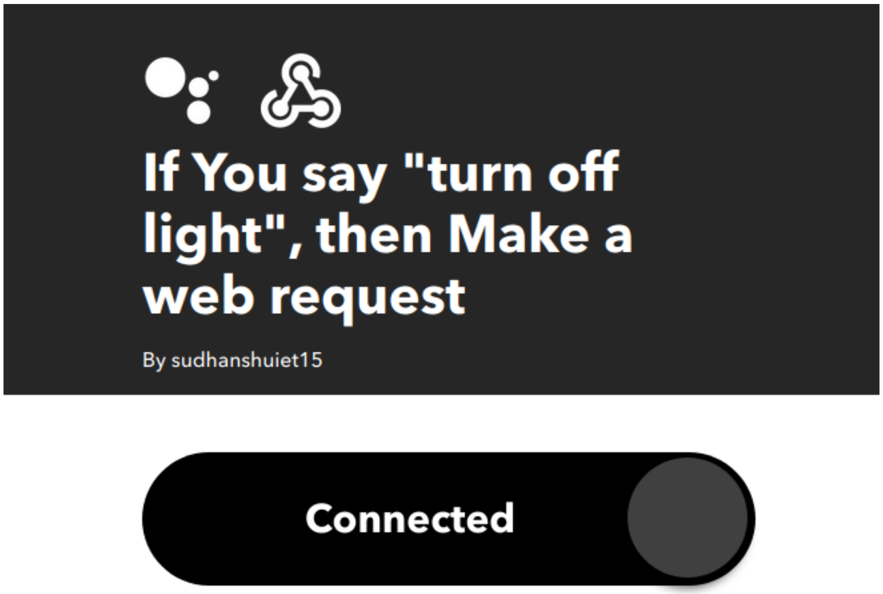
1.5 Screenshots



If You say "Turn on the light", then Make a web request

By sudhanshuiet15

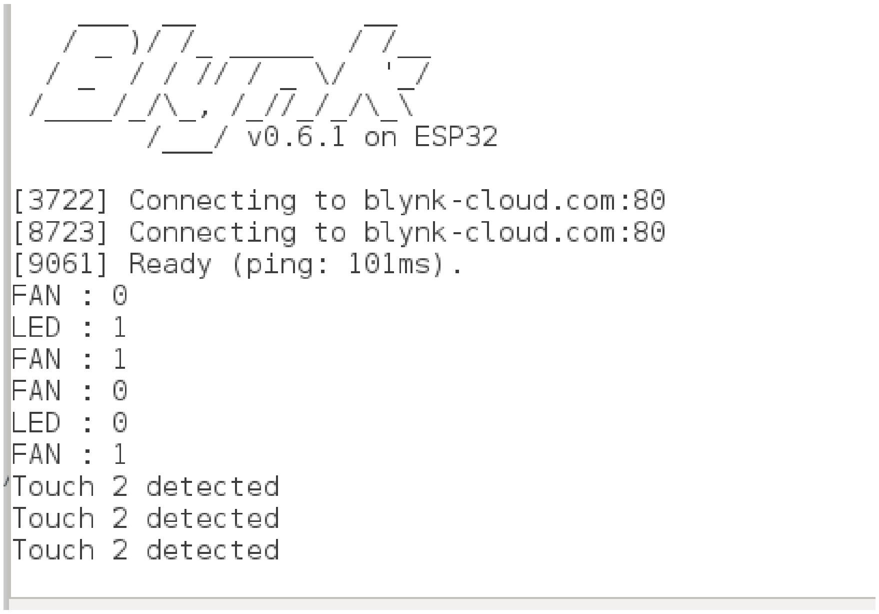
Figure 1: Google Assistant Turn on LED



If You say "turn off light", then Make a web request

By sudhanshuiet15

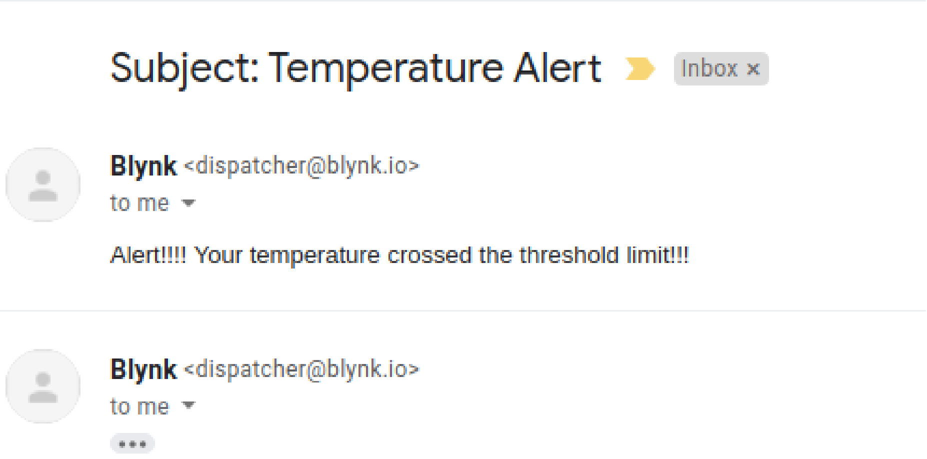
Figure 2: Google Assistant Turn off LED



```
    ___  __          __
   / _ )/ /_ _____  / /__
  / _  / / // / _ \/  '_/
 /____/_/\_, /_//_/_/\_\
        /___/ v0.6.1 on ESP32

[3722] Connecting to blynk-cloud.com:80
[8723] Connecting to blynk-cloud.com:80
[9061] Ready (ping: 101ms).
FAN : 0
LED : 1
FAN : 1
FAN : 0
LED : 0
FAN : 1
Touch 2 detected
Touch 2 detected
Touch 2 detected
```

Figure 3: Serial Monitor WIFI connection



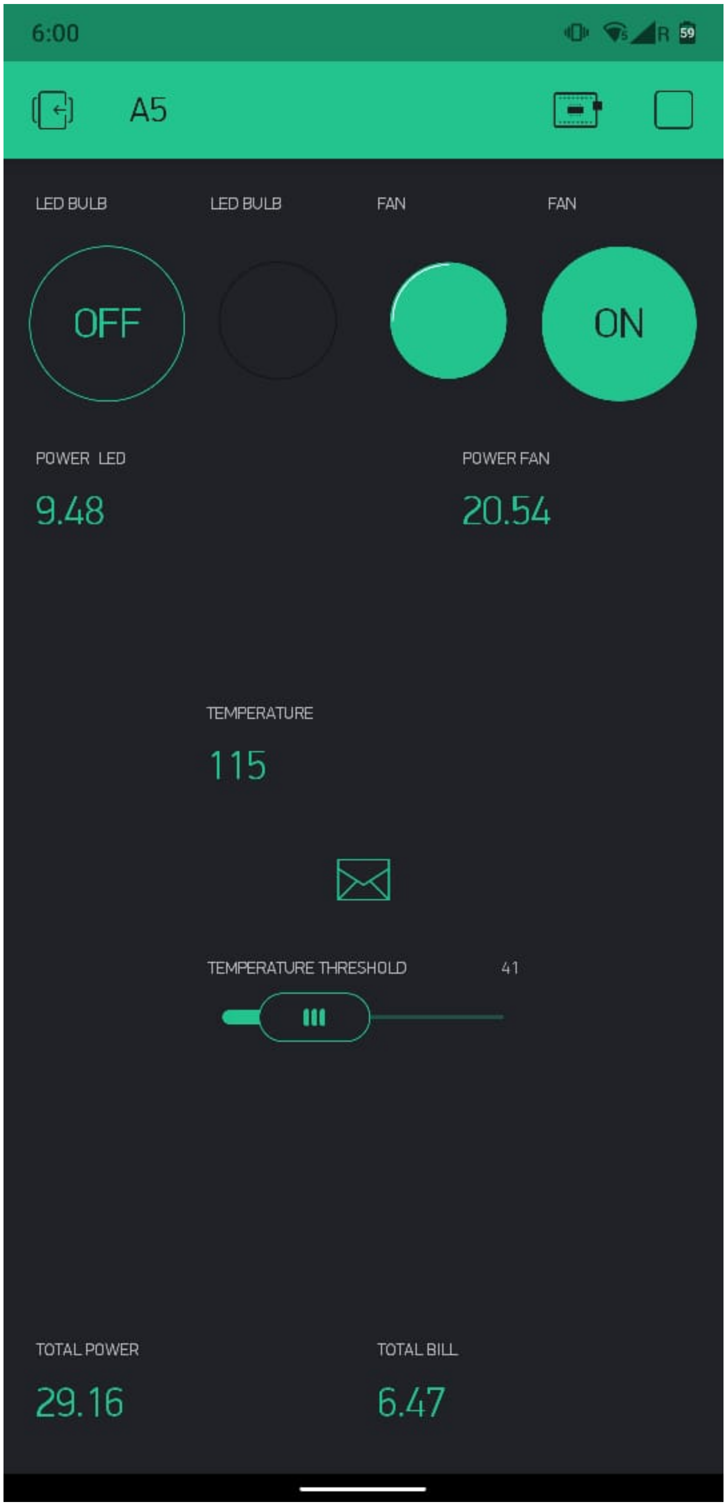
Subject: Temperature Alert

Blynk <dispatcher@blynk.io>
to me

Alert!!!! Your temperature crossed the threshold limit!!!

Blynk <dispatcher@blynk.io>
to me

Figure 4: Temperature Alert mail



POWER LED
9.48

POWER FAN
20.54

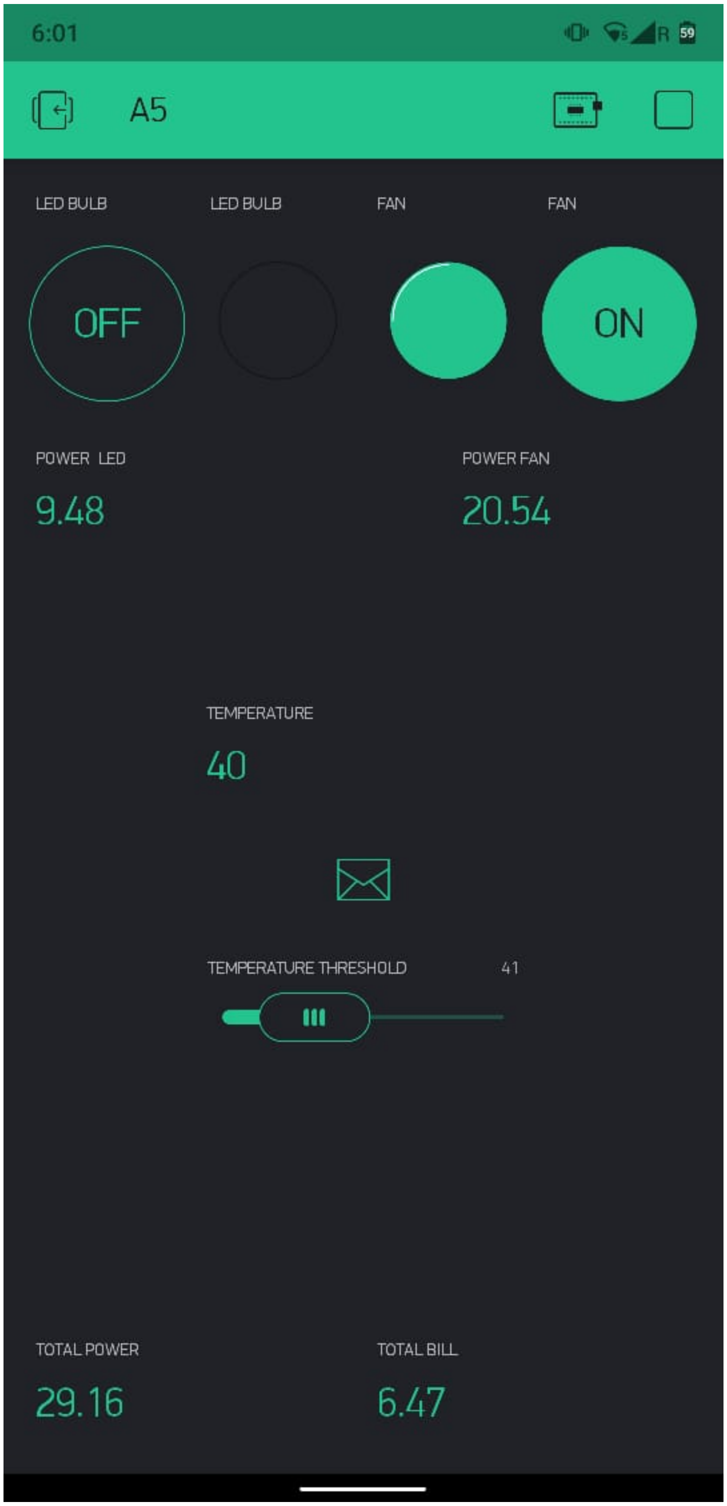
TEMPERATURE
115

TEMPERATURE THRESHOLD
41

TOTAL POWER
29.16

TOTAL BILL
6.47

Figure 5: Blynk App interface



POWER LED
9.48

POWER FAN
20.54

TEMPERATURE
40

TEMPERATURE THRESHOLD 41

TOTAL POWER
29.16

TOTAL BILL
6.47

Figure 6: Blynk App interface temperature sensor

2 Appendix

2.1 Appendix-A : Code

```

#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include "esp_wpa2.h"

#define EAP_IDENTITY "jtm192207"
#define EAP_PASSWORD "jtm22072908"
const char* ssid = "IITD_WIFI";
const char* host = "https://proxy62.iitd.ac.in/cgi-bin/proxy.cgi";
int pin = 2;
float wattled= random(900,1200)/100.0;
float wattfan= random(1900,2200)/100.0;
float totalpower = 0;
float watt=0;
float watt1=0;
float total=0;
int currenttime1=0;
int elapsedtime1=0;
float timeon1=0;
float price1=0;
int currenttime=0;
int elapsedtime=0;
float price=0;
float timeon=0;

/////////////////////////////////////////////////////////////////////////////
BlynkTimer timer;
WidgetLED led1(V2);
WidgetLED led2(V3);

BLYNK_WRITE(V0)
{
  int pinValue1 = param.asInt(); // assigning incoming value from pin V1 to a variable
  // You can also use:
  // String i = param.asStr();
  // double d = param.asDouble();
  Serial.print("LED : ");
  Serial.println(pinValue1);
  if (pinValue1==1)
  { currenttime1=millis();
    led1.on();
    watt1=random(900,1100)/100.0;
    Blynk.virtualWrite(V6, watt1);
    totalpower = totalpower + watt1;
  }
  else
  {  led1.off();
   elapsedtime1=millis();
    timeon1=((elapsedtime1-currenttime1)/(1000.0*3600.0));
    price1=watt1*timeon1*100;
    total=total+price1;
    Blynk.virtualWrite(V9, total);
      Blynk.virtualWrite(V8, totalpower);
  }
}
//////////////////////////////////////////////////////////////////////////
BLYNK_WRITE(V1)
{
  int pinValue2 = param.asInt(); // assigning incoming value from pin V1 to a variable
  // You can also use:
  // String i = param.asStr();
  // double d = param.asDouble();
  Serial.print(”FAN : ”);
  Serial.println(pinValue2);
  if (pinValue2==1)
  { currenttime=millis();
    led2.on();
    watt=random(1900,2100)/100.0;
    Blynk.virtualWrite(V7, watt);
     totalpower = totalpower + watt;
  }
  else
  {  led2.off();
      elapsedtime=millis();
    timeon=((elapsedtime-currenttime)/(1000.0*3600.0));
    price=watt*timeon*100;
    total=total+price;
    Blynk.virtualWrite(V9, total);
    Blynk.virtualWrite(V8, totalpower);
  }
}
//////////////////////////////////////////////////////////////////////////////


//////////////////////////////////////////////////////////////////////////////
int threshold = 40;
bool touch2detected = false;
BLYNK_WRITE(V4)
{
threshold = param.asInt();
}
void gotTouch2(){
 touch2detected = true;
}



////////////////////////////////////////////////////////////////////////////
void setup()
```

```
{
  pinMode(pin, OUTPUT);
  pinMode(pin, HIGH);
  Serial.begin(115200);
  delay(10);
  Serial.print("Connecting to ");
  Serial.println(ssid);

  touchAttachInterrupt(T3, gotTouch2, threshold);

  WiFi.disconnect(true);
  WiFi.mode(WIFI_STA);
  esp_wifi_sta_wpa2_ent_set_identity((uint8_t *)EAP_IDENTITY, strlen(EAP_IDENTITY));
  esp_wifi_sta_wpa2_ent_set_username((uint8_t *)EAP_IDENTITY, strlen(EAP_IDENTITY));
  esp_wifi_sta_wpa2_ent_set_password((uint8_t *)EAP_PASSWORD, strlen(EAP_PASSWORD));
  esp_wpa2_config_t config = WPA2_CONFIG_INIT_DEFAULT();
  esp_wifi_sta_wpa2_ent_enable(&config);
  WiFi.begin(ssid);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
  Serial.println("WiFi connected");
  Serial.println("IP address set: ");
  Serial.println(WiFi.localIP());
  Blynk.begin(" 2PjbSS5zvRf2Bpwu_ccVt9pbur4-DkFf", ssid, EAP_PASSWORD);

   Blynk.virtualWrite(V5, "115");



//////////////////////////////////////////////////////////////////////////
}
void loop()
{
  Blynk.run();
//  total=total+(((totaltime*watt)+(totaltime1*watt1))*100);
//  Blynk.virtualWrite(V9, total);
   if(touch2detected){
    Blynk.virtualWrite(V5, threshold);
    Blynk.email("sudhanshuiet15@gmail.com", "Subject: Temperature Alert", "Alert!!!! Your temperature crossed the threshold limit!!!");
    delay(4000);
    touch2detected = false;
    Blynk.virtualWrite(V5, "115");
    Serial.println("Touch 2 detected");

   }


}
```

References

[1] *Blynk*. https://blynk.io/.

[2] *Blynk examples*. https://examples.blynk.cc/?board=ESP8266&shield=ESP8266%20WiFi&example=GettingStarted%2FPushData.

[3] *Control appliances using esp32 / esp8266 in AP mode*. https://www.youtube.com/watch?v=weGGCJGn2H8.

[4] *Getting Started with the ESP32 Development Board*. https://randomnerdtutorials.com/getting-started-with-esp32/.

[5] *GOOGLE ASSISTANT CONTROLLED SWITCH USING NODEMCU*. https://www.instructables.com/id/GOOGLE-ASSISTANT-CONTROLLED-SWITCH-USING-NODEMCU/.

[6] *How to display ANY sensor data in Blynk app*. http://help.blynk.cc/en/articles/512056-how-to-display-any-sensor-data-in-blynk-app.

[7] *IFTTT*. https://ifttt.com/.

[8] *Use Google Assistant to control an ESP8266 using Blynk and IFTTT*. http://www.whatimade.today/control-an-esp8266-verbally-using-google-assistant/.