**A project report**

**on**

**HOME AUTOMATION USING ESPMODULE 8266**

***Submitted by***

**B.Gowrav Krishna 180040295**

**N.Sai Rakesh 180040526**

**P.Sai Gowtham 180040673**

**M.Sai Teja 180040293**

**N.Eswar 180040256**

**Sec - 7, Batch -6**

**I/IV BACHELOR OF TECHNOLOGY**

IN

Electronics and Communication engineering

**(SEMESTER-II)**

**WORKSHOP PRACTICE FOR ELECTRICAL & ELECTRONICS ENGINEERS**

**DEPARTMENT OF BASIC ENGINEERING SCIENCES - II**

**Koneru Lakshmaiah Education Foundation**

**VADDESWARAM-522 502**

April 2019

****

**Koneru Lakshmaiah Education Foundation**

**DEPARTMENT OF BASIC ENGINEERING SCIENCES - II**

**Workshop practice for Electrical & Electronics engineers (18EE1003)**

This is to certify that **Mr.B.Gowrav krishna-180040295, Mr.N.Sai Rakesh-180040526, Mr.P.Sai Gowtham-180040673, Ms.M.Sai Teja-180040293, & N.Eswar-18004025** of section ***7*** studying I/IV B.Tech in ***ECE*** has satisfactorily completed project ***Home Automation using ESPMODULE 8266*** in the semester ***II*** during the academic year 2018-2019

**Signature of Course Instructor Signature of Course Coordinator**

**Signature of HOD**

**ACKNOWLEDGEMENT**

We express great pleasure for me to express my gratitude to our honourable President **Sri. Koneru Satyanarayana**, for providing the opportunity and platform with facilities in accomplishing the project based laboratory. We express the sincere gratitude to our principal **Dr. N. Venkatram** for his administration towards our academic growth.

We express sincere gratitude to our Coordinator **Dr. V. Krishna Reddy** and Head of the department of BES - II **Dr. S. N. PADHI** for his leadership and constant motivation provided in successful completion of our academic semester. We record it as our privilege to deeply thank for providing us the efficient faculty and facilities to make our ideas into reality.

We express my sincere thanks to our project supervisor **Mr. M.V.Suman** for his novel association of ideas, encouragement, appreciation and intellectual zeal which motivated us to venture this project successfully.

We are pleased to acknowledge the indebtedness to our lab technicians who devoted themselves directly or indirectly to make this project success.

Last but not the least we express our deep gratitude and affection to our parents who stood behind us in all our endeavours.

**B.Gowrav Krishna 180040295**

**N.Sai Rakesh 180040526**

**P.Sai Gowtham 180040673**

**M.Sai Teja 180040293**

**N.Eswar 180040256**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| S.NO | CONTENTS | PAGE NO |
| 1. | Abstract | 5 |
| 2. | Architecture of project | 6 |
| 3. | Components required | 7-8 |
| 4. | Working | 9 |
| 5. | Code | 10-16 |
| 6. | Output | 17 |
| 7. | Advantages |  |
| 8.  9. | conclusion  Reference |  |
| . |  |  |
|  |  |  |
|  |  |  |
|  |  |
|  |  |

**ABSTRACT**

Smart homes refers to energy efficient, time

saving includes reduces the human efforts. Commercial

content distribution network (CDNs) has Video-on-demand

(VOD) services for on-demand capacity provisioning on

smart television (Home Automation). The content has full

control and customization, so they prefer self- managed

content infrastructure. The initial capital investment for

physical infrastructure was so expensive and quite difficult

for management. The objective of key providers was to

reduce the server rental cost by accessing through the

efficient and inexpensive ESP8266 chip to the cloud

platforms on which dynamic server provisioning to reduce

the server rental cost which was the key objective for

content providers. In this paper we addressed the

challenges of reducing the cost of accessing the cloud

platforms. We present the construction of ESP8266 with the

Arduino or microcontroller are attached to get the smart

television, where all our choice of videos can be stored and

retrieved at any time. Adafruit, Electrodragon, Sparkfun

helps to get the chip. Through the software development kit

(SDK) in ESP8266 that allow the chip to be programmed

which remove the need of the microcontroller. Generally

accessing cloud facility will produce high cost but in this

paper gives the replacement of the accessing the cloud

with the ESP8266 serial chip which operated in low

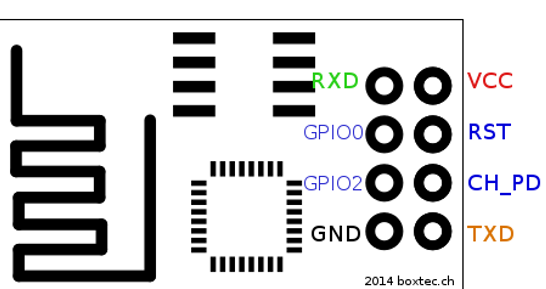
power consumption with the low cost because it has the

three configurable sleep modes

**Architecture of the project:**

**A circuit board

Description automatically generated**

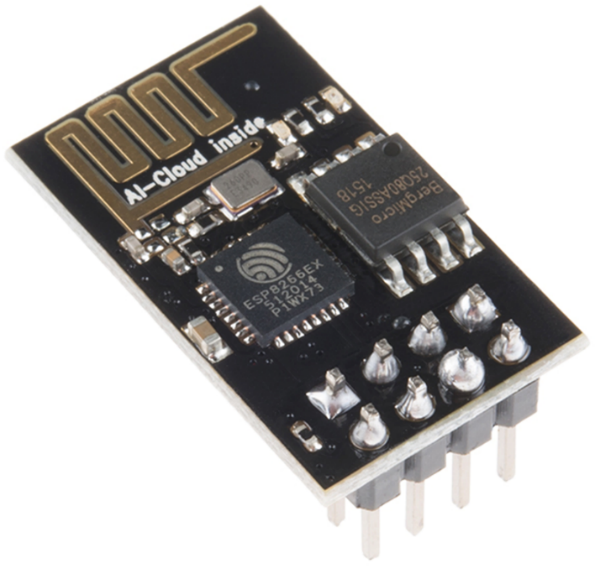
****

**Procedure:**

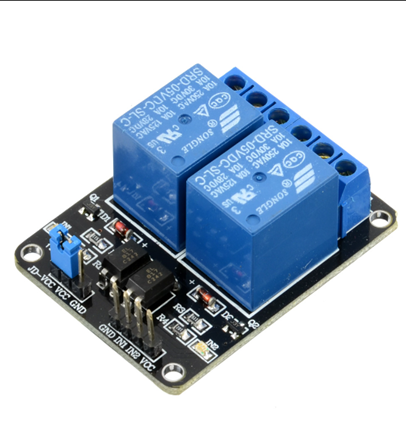
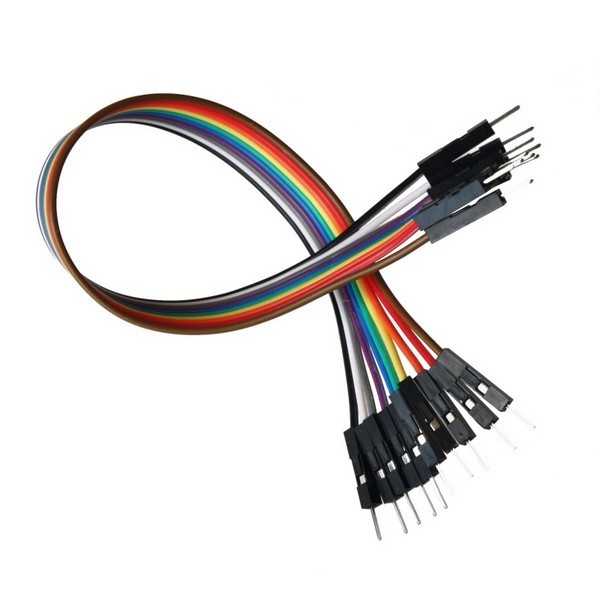
**Components Required:**

In this project we have used ESPMODULE 8266,Jumper Wires,Relay Module92 channel),LED,Mobile phone.

**# ESP MODULE 8266:**



**# Jumper Wires:** #**Relay Module:**



**WORKING:**

Home automation is commonly called smart home

**\***.It involves the control and handle the things like

light, fan, security purposes etc. All the things are connected

to the internet and all of them can accessed at any place

and any time.

**\*.**Are all incompatible with other.

The web server is simultaneously updated

by sensing the status of the things which are

connected to the network. The status of the

appliances is controlled by the switch it either ON or

OFF using the computer technology. It also provides

the security, energy efficient.

**\***.And ease of use, hence it is adopted more.

It also helps by providing to control and monitoring on

web browser.All the devices can be used in our own sitting

place itself. The problem overcome by this paper is about

that home automation is generally implemented by using

Bluetooth through our PC

**Code:**

#include <ESP8266WiFi.h>

#include "Adafruit\_MQTT.h"

#include "Adafruit\_MQTT\_Client.h"

#define WLAN\_SSID       "gowrav\_krishna"

#define WLAN\_PASS       "gowrav143"

#define AIO\_SERVER      "[io.adafruit.com](http://io.adafruit.com/)"

#define AIO\_SERVERPORT  1883

#define AIO\_USERNAME    "gowrav143"

#define AIO\_KEY         "790682310a1d4dd69f5b058098b752f4"

WiFiClient client;

Adafruit\_MQTT\_Client mqtt(&client, AIO\_SERVER, AIO\_SERVERPORT, AIO\_USERNAME, AIO\_KEY);

Adafruit\_MQTT\_Publish shut\_down = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/shutdown");

Adafruit\_MQTT\_Subscribe shut\_down1 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/shutdown");

Adafruit\_MQTT\_Publish tpl = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/turnall");

Adafruit\_MQTT\_Subscribe tl = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/turnall");

Adafruit\_MQTT\_Subscribe l1 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/l1");

Adafruit\_MQTT\_Publish lp1 = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/l1");

Adafruit\_MQTT\_Subscribe l2 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/l2");

Adafruit\_MQTT\_Publish lp2 = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/l2");

Adafruit\_MQTT\_Subscribe l3 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/l3");

Adafruit\_MQTT\_Publish lp3 = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/l3");

Adafruit\_MQTT\_Subscribe l4 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/l4");

Adafruit\_MQTT\_Publish lp4 = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/l4");

Adafruit\_MQTT\_Subscribe f1 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/f1");

Adafruit\_MQTT\_Publish fp1 = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/f1");

Adafruit\_MQTT\_Subscribe f2 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/f2");

Adafruit\_MQTT\_Publish fp2 = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/f2");

Adafruit\_MQTT\_Subscribe f3 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/f3");

Adafruit\_MQTT\_Publish fp3 = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/f3");

Adafruit\_MQTT\_Subscribe f4 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/f4");

Adafruit\_MQTT\_Publish fp4 = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/f4");

int dude = 0;

void MQTT\_connect();

void setup() {

  Serial.begin(115200);

  delay(10);

  pinMode(D0, OUTPUT);

  pinMode(D1, OUTPUT);

  pinMode(D2, OUTPUT);

  pinMode(D3, OUTPUT);

  pinMode(D4, OUTPUT);

  pinMode(D5, OUTPUT);

  pinMode(D6, OUTPUT);

  pinMode(D7, OUTPUT);

  Serial.println(F("Adafruit MQTT demo"));

  Serial.println(); Serial.println();

  Serial.print("Connecting to ");

  Serial.println(WLAN\_SSID);

  WiFi.begin(WLAN\_SSID, WLAN\_PASS);

  while (WiFi.status() != WL\_CONNECTED) {

    delay(500);

    Serial.print(".");

  }

  Serial.println();

  Serial.println("WiFi connected");

  Serial.println("IP address: "); Serial.println(WiFi.localIP());

digitalWrite(D0,HIGH);

digitalWrite(D1,HIGH);

digitalWrite(D2,HIGH);

digitalWrite(D3,HIGH);

digitalWrite(D4,HIGH);

digitalWrite(D5,HIGH);

digitalWrite(D6,HIGH);

digitalWrite(D7,HIGH);

  mqtt.subscribe(&l1);

  mqtt.subscribe(&l2);

  mqtt.subscribe(&l3);

  mqtt.subscribe(&l4);

  mqtt.subscribe(&f1);

  mqtt.subscribe(&f2);

  mqtt.subscribe(&f3);

  mqtt.subscribe(&f4);

  mqtt.subscribe(&shut\_down1);

  mqtt.subscribe(&tl);

}

uint32\_t x = 0;

void loop() {

  MQTT\_connect();

String g, o, w, r, a, v, k, i;

  Adafruit\_MQTT\_Subscribe \*subscription;

  while ((subscription = mqtt.readSubscription(5000))) {

    if (subscription == &l1 || subscription == &l2 || subscription == &l3 || subscription == &l4 || subscription == &f1 || subscription == &f2 || subscription == &f3 || subscription == &f4) {

      g = (char\*)l1.lastread;

      o = (char\*)l2.lastread;

      r = (char\*)l4.lastread;

      a = (char\*)f1.lastread;

      v = (char\*)f2.lastread;

      k = (char\*)f3.lastread;

      i = (char\*)f4.lastread;

      if (g == "ON")

        digitalWrite(D0, LOW);

      else if (g == "OFF")

        digitalWrite(D0, HIGH);

      if (o == "ON")

        digitalWrite(D1,LOW);

      else if (o == "OFF")

        digitalWrite(D1, HIGH);

      if (w == "ON")

        digitalWrite(D7,LOW);

      else if (w == "OFF")

        digitalWrite(D7, HIGH);

      if (r == "ON")

        digitalWrite(D2,LOW);

      else if (r == "OFF")

        digitalWrite(D2, HIGH);

      if (a == "ON")

        digitalWrite(D3,LOW);

      else if (a == "OFF")

        digitalWrite(D3, HIGH);

      if (v == "ON")

        digitalWrite(D4,LOW);

      else if (v == "OFF")

        digitalWrite(D4, HIGH);

      if (k == "ON")

        digitalWrite(D5,LOW);

      else if (k == "OFF")

        digitalWrite(D5, HIGH);

      if (i == "ON")

        digitalWrite(D6,LOW);

      else if (i == "OFF")

        digitalWrite(D6, HIGH);

    }

    else if (subscription == &shut\_down1)

      {

        String ling = (char\*)shut\_down1.lastread;

        if (ling == "ON")

        {

          dude++;

        }

      }

    else if (subscription == &tl)

      {

        String ling = (char\*)tl.lastread;

        if (ling == "ON")

        {

          digitalWrite(D0,LOW);

    digitalWrite(D1,LOW);

    digitalWrite(D2,LOW);

    digitalWrite(D3,LOW);

    digitalWrite(D4,LOW);

    digitalWrite(D5,LOW);

    digitalWrite(D6,LOW);

    digitalWrite(D7,LOW);

    delay(1000);

    lp1.publish("ON");

    delay(1500);

    lp2.publish("ON");

    delay(1500);

    lp4.publish("ON");

    delay(1500);

    fp1.publish("ON");

    delay(1500);

    fp2.publish("ON");

    delay(1500);

    fp3.publish("ON");

    delay(1500);

    fp4.publish("ON");

    delay(1500);

    tpl.publish("OFF");

    delay(1500);

        }

      }

     }

  if (dude)

  {

    digitalWrite(D0, HIGH);

    digitalWrite(D1, HIGH);

    digitalWrite(D2, HIGH);

    digitalWrite(D3, HIGH);

    digitalWrite(D4, HIGH);

    digitalWrite(D5, HIGH);

    digitalWrite(D6, HIGH);

    digitalWrite(D7, HIGH);

    delay(1000);

    lp1.publish("OFF");

    delay(1500);

    lp2.publish("OFF");

    delay(1500);

    lp4.publish("OFF");

    delay(1500);

    fp1.publish("OFF");

    delay(1500);

    fp2.publish("OFF");

    delay(1500);

    fp3.publish("OFF");

    delay(1500);

    fp4.publish("OFF");

    delay(1500);

    shut\_down.publish("OFF");

    delay(1500);

    dude = 0;

  }

}

void MQTT\_connect() {

  int8\_t ret;

 if (mqtt.connected()) {

    return;

  }

  Serial.print("Connecting to MQTT... ");

  uint8\_t retries = 3;

  while ((ret = mqtt.connect()) != 0) {

    Serial.println(mqtt.connectErrorString(ret));

    Serial.println("Retrying MQTT connection in 5 seconds...");

    mqtt.disconnect();

    delay(5000);

    retries--;

    if (retries == 0) {

      while (1);

    }

  }

  Serial.println("MQTT Connected!");}

**Output:** Screenshot

A screenshot of a computer

Description automatically generated

****

**Advantages:**

1. High quality

2. Low cost

3. No extra power supply

4. Software with many functions

5. Control electrical devices according weekday/date/time

6. Create timers or pulses with our software

**Conclusions:**

This paper gives the basic knowledge of IOT,

where all the appliances are controlled and can also

update the status of the device. ESP8266 is very

useful in IOT industry where it is cost efficient,

security, power is saved , compact in design and its

performance is reliable,where all the devices are easily

control when the users are in out of town. The convenience

is increased by accessing wherever and saves our

valuable time, cost. ESP8266 would be no longer

burned by the users heating bill and also it is cheap in

cost for our home automation.

**References:**

**1.**https://www.researchgate.net/publication/316926871\_HOME\_AUTOMATION\_ON\_ESP8266

2. <http://en.wikipedia.org>

3. http://www.iot-playground.com