MEASUREMENT AND INSTRUMENTATION

EEE2004

PROJECT REVIEW

Title: SMART AND SECURE HOME USING IoT

SUBMITTED TO- Prof.Thamilmaran

SUBMITTED BY-

1)Anagha Vivek Alurkar(17BEE0061)

2)B.Nikhil (17BEE0068)

3)Meghnaa Sunil Jaiswal (17BEE0096)

4)Shivani Kumari (17BEE0099)

5)Madhukumar S (17BEE0117)

6)Nooman Neha Shirin (17BEE0241)

SLOT-F2

CERTIFICATE

his is to be certified that Ms. Anagha Vivek Alurkar[17BEE0061],Mr. Nikhil[17BEE0068],Ms. Meghnaa Sunil Jalswal[17BEE0096], Ms.Shivani umari[17BEE0099], Mr.Madhukumar S[17BEE0117], Ms. Nooman Neha hirin[17BEE0241], students of II Year B.Tech, have successfully completed heirproject entitled **Smart and Secure Home using IoT** under the guidance of Prof.Thamilmaran A' during Fall-semester 2018-2019 of School of Electrical ngineering(SELECT),Vellore Institute of Technology, Vellore (TAMILNADU)

hey have submitted their Project Report in record for their own work for the artial fulfilment of the curriculum of the course 'Measurement and nstrumentation','EEE2004'.

Prof. 24/10/18

Designation AP Sr.

[DATE] 24/10/18

SMART AND SECURE HOME USING IoT

ANAGHA VIVEK ALURKAR (17BEE0061)

B. NIKHIL (17BEE0068)

MEGHNAA SUNIL JAISWAL (17BEE0096)

SHIVANI KUMARI (17BEE0099)

MADHUKUMAR S (17BEE0117)

NOOMAN NEHA SHIRIN (17BEE0241)

B.tech. Electrical and Electronics Department

Vellore Institute of Technology, Vellore, Tamil Nadu – 632014, India

I.ABSTRACT

In this modern era with increasing workload and hectic schedules Home Automation is being employed in many houses in order to save time, energy and money. In this paper we have tried to develop a Smart and Secure Home Automation System using Internet Of Things(IoT) which is affordable, safe and user friendly. This system consists of sensors (like PIR,MQ2 etc.) connected to Arduino UNO from which system is controlled with an user-interface that is interacted via Android App .The App would be sending a message to the user when there is an emergency so as to safeguard our smart home.

Keywords: Automation, IoT, Sensors, Nodemcu, Multiplexer.

A. INTRODUCTION

As we have entered the 21st century, usage of android phones have been increased rapidly, as a result its application is also increasing day by day. With many efficient development in the mobile networks and mobile devices better and innovative methods of Home Automation can be developed to make it easy and convenient for the user to access it from any place and at any time. Home Automation generally means removing as much human's physical effort as possible and replace them with electronic systems. In this paper various applications of Home Automation using Internet Of Things (IoT) are included like Security, Gas detection, Fire detection, Refrigerator control, Soil moisture control etc. By using of the Internet Of Things (IoT) the development of Home Automation is going to be more easy and more popular. Internet Of Things(IoTs) generally means connection of different kinds of objects such as smart phones, TVs, sensors etc to the internet. The devices are linked smartly as to allow communication between human and things and among the things themselves

B. COMPONENTS REQUIRED

Hardware components:

1)Arduino UNO

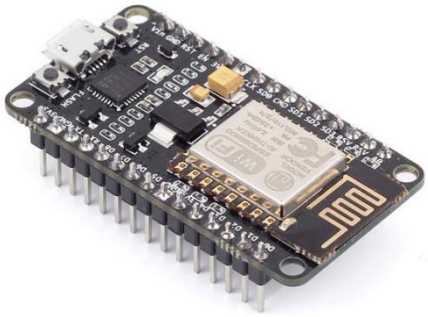
2)NodeMCU

3)Light Dependent Resistor
4)Thermistor
5) Soil moisture sensor
6)Gas sensor
7)MUX
8)Motor Driver
9)Buzzer
10)Jumper wires
11)Bread board

Software components:
1)Arduino IDE
2)Ubidots IoT platform

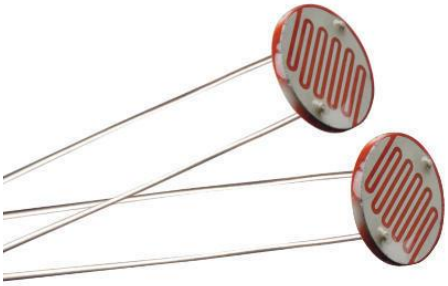
Description of the components:
1)Arduino UNO:

The Arduino UNO is a widely used open source micro-controller board based on the ATmega328P micro-controller and developedby Arduino.cc. It is used to interface with sensors ,buzzer and water pump.
2)NodeMCU(ESP8266)



NodeMCU is an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module. We used this development board to interface with all 6 analog sensors and to upload the data from received from sensors to Ubidots.
3)LDR

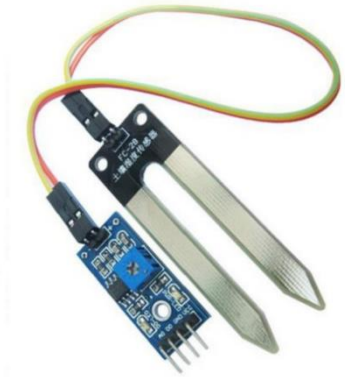
It is a light-controlled variable resistor also called as photo resistor which works on the principle photoconductivity, which means that with increase in light intensity the resistance of a LDR decreases.



4)NTC Thermistor

A thermistor is a temperature controlled variable resistor. NTC thermistors are resistors with a negative temperature coefficient, which indicates that with increase in temperature the resistance decreases.

5)FC-28 (Soil moisture sensor)



This sensor is a simple breakout for measuring the moisture present in the soil and similar materials. The soil moisture sensor is very easy and simple to use. The two large exposed pads functions as probes for the sensor, together acting as a variable resistor. The more the water in the soil means better the conductivity between the pads which finally results in a lower resistance and a higher AOUT.

6)MQ 2 Gas Sensor

MQ2 gas sensor can be used to detect the presence of LPG, propane and Hydrogen, methane and other combustible steam, it is of low cost and suitable for different applications .It is sensitive to flammable gas and smoke. It gives analog output.

7)MUX(16:1)
Multiplexer is used when we have many analog sensors and no sufficient analog to digital convertor pins. Multiplexer will take input from a single channel at a time and it gives to the analog to digital convertor pins of NodeMCU.

Limitations: The major disadvantage in its use is that MUX can transfer only one value at a time to the NodeMCU. It would take some time for the rest of the values to be sent and detected. So it is not advisable to be used in application where immediate action is required such as any sudden danger is detected.
8)L293D Motor Driver
This motor driver is used to interface the water pump with Arduino UNO. We can't connect water motor to Arduino directly as Arduino can't meet the current requirements for the water pump to run.

Software Components
UBIDOTS

It is an IoT platform, where we can upload the physical quantities measured by the sensors and store the data from 1 day to 1 year and also trigger actions like sending the mails, message alerts, voice call based on sensor data.

I. METHODODLOGY

A. CONCEPT REVIEW

Earlier home automation based projects were performed on Arduino kits. In our project along with the Arduino NodeMCU is also used, NodeMCU has huge advantages over the arduino, NodeMCU unlike arduino allows inbuilt Wi-Fi functionality.

Our project mainly focuses on the automation of appliances with the help of android application. In today's world, the main goal for any system is optimisation. Any system developed aims mainly at reducing the human efforts and our system also does the same.

Home automation systems, give homeowners the power to control their electricity, temperature and security etc at anytime from anyplace. This also makes the user perform specific functions without the need of physical involvement.

Home automation can also be very useful for the elderly or disabled. They can perform household activities with much more ease. The automation of appliances can reduce the need of physical involvement for these set of people who are not able to perform the activities due to their inabilities.

Our project mainly aims at implementing the following:

1)Main Door Security system

During night when an unauthorised person tries entering the house the buzzer will start alarming, the owner will automatically get an alert message and he can also watch the live updates on the dashboard.

2)Automatic sprinkler system:

If the soil in the garden is dry then the sprinkler motor will be automatically switched on and a message will be sent to the owner and similarly if the soil has got enough water then the motor will be switched off automatically and an alert message will be sent to the owner. In addition to this the owner can also see the updates on the dashboard.

3)Gas Leakage detection:

When there is a leakage of gas in the house it will automatically send an alert message to the owner and an alarming sound will be heard from the buzzer. The owner can also watch the live updates on the dashboard.

4)Fire detection:

When there is a fire accident in the house an alert message will be sent to the owner and he can also check the updates on the dashboard and there will be an alarm to alert the people in the house.

5)Automated refrigerator system:

Often people forget to close the refrigerator door or don't close it properly which may lead to many severe problems further, so in order to check whether the door is closed this system has been developed where the owner gets a message if the door is not closed and he can also check the updates on the dashboard.

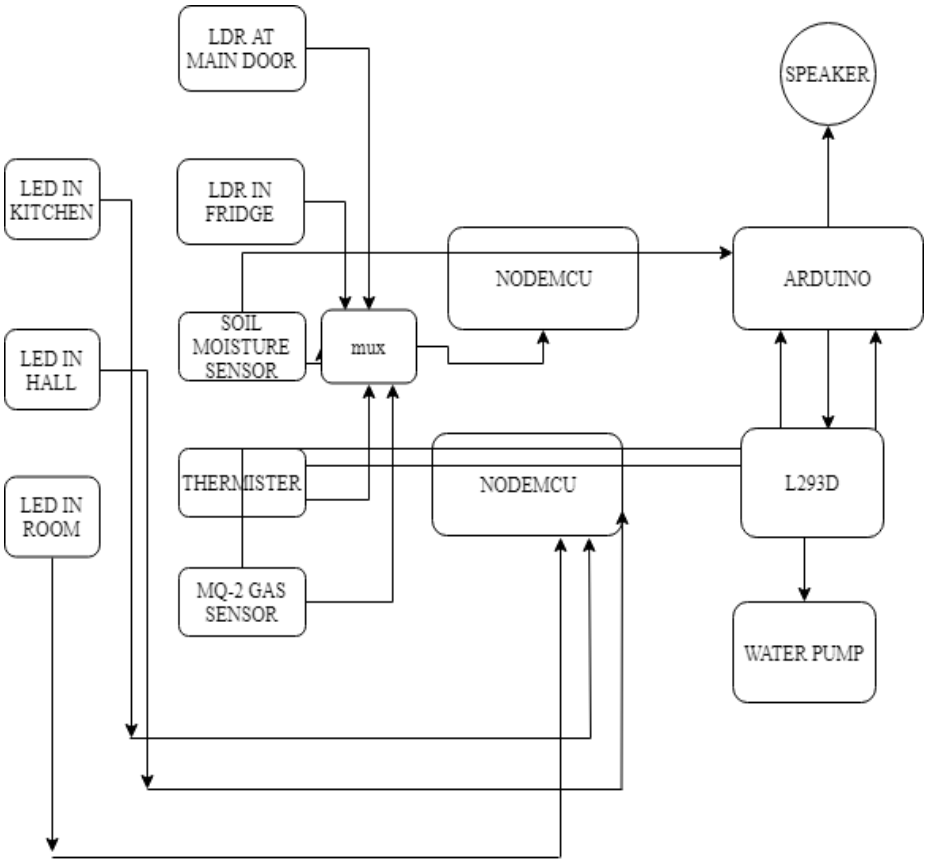
6)Light controlled rooms:

This system enables the owner to switch on and switch off the lights from anyplace across the globe.

In our project we are using several types of sensors (like MQ2,PIR etc) in which 6 are analog sensors, these sensors are connected to multiplexer as NodeMCU has only one analog to digital convertor pin, whereas we need 6 pins, hence the multiplexer takes the input from a single channel at a time and it gives to the analog digital pin of NodeMCU. The NodeMCU gets the data from the sensors and uploads the data to the IoT platform.

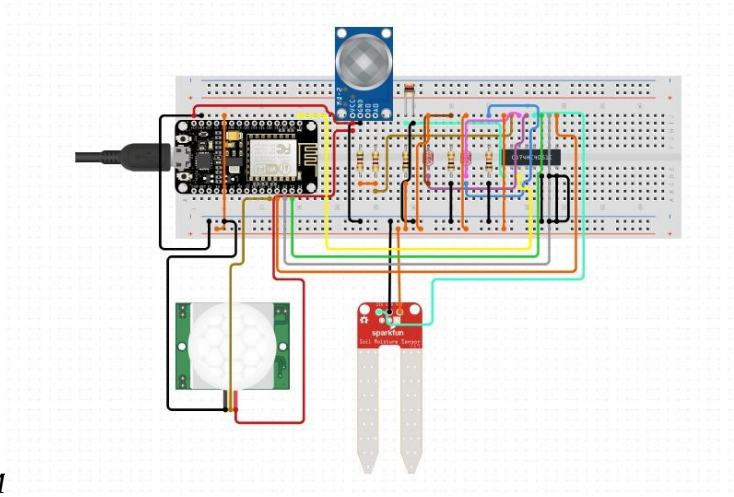
Here Arduino is used as a interface between water pump and speaker. Though this can be done by the NodeMCU but as we have already used it for other purpose if we again use it for connecting speakers and motors there will be huge delay in uploading the data to IoT platform.

B. PROPOSED SYSTEM



C. HARDWARE IMPLEMENTATION:

To make the hardware of the system to work we should make the connections according to the circuit shown *D.*



CIRCUIT DIAGRAM

1)Laser Security At Main Door

We used LDR in this system, the LDR analog output is connected to NodeMCU's 10 bit AD convertor. So the principle is when the laser light is falling on the LDR the ADC value will be in the range of 10-50.If the laser light is blocked then the LDR ADC value increases to the range of 600-900. In the arduino code and Ubidots we have

given the conditions that if LDR value is greater than or equal to 600 it sends an alert message to the owner and alarm is also given by the buzzer

2)Garden automation
We used FC-28 soil moisture sensor in this system. Whenever the soil is dry the ADC value of the sensor will be in the range of 900-1024 then an automatic command will be given to Arduino to switch on the motor and Ubidots sends alert message to the owner and vice versa.

3)Gas leakage detection

We used MQ2 gas sensor in this system , it will detect LPG gas. When there is no leakage of gas the ADC value of MQ2 sensor will be around 200. When there is either LPG, methane or propane gas the ADC value would increase upto 1024.We used the condition if MQ2 sensor ADC value is greater than or equal to 400 then it will send an alert message to the owner and also an alarm beeps.

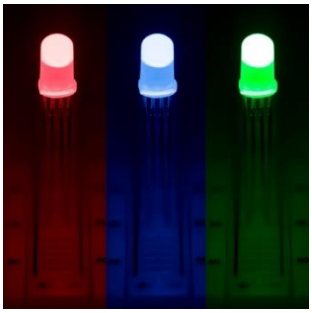
4)Fire detection

We used 10K thermistor(NTC), when there is no fire the ADC value of the thermistor would be around 500 and when there is a fire the ADC value of the thermistor will decrease and would reach around 100.We used the condition if ADC value of thermistor is less than or equal to 200 it will send an alert message to the owner along with an alarming sound.

5)Refrigerator door

We have used LDR inside the refrigerator. When the door is closed the ADC value of the LDR will be in the range of 800 to 1024, when it is open light starts falling on the LDR and the ADC value reduces to the range of 0-50,hence we used the condition if value of LDR is less than or equal to 100 send an alert message to the owner.

6)Light controlled room



All three LED's are connected to NodeMCU digital pins. The NodeMCU's will receive commands from IoT platform which is nothing but the Ubidots. There will be three virtual switches in the Ubidots from which owner can switch on and off the lights from anywhere.

D. SOFTWARE IMPLEMENTATION:

The code implemented for the above system using C programming.

NodeMCU code : To measure the sensor data and to upload the corresponding values to the Ubidots.

```
#include <UbidotsMicroESP8266.h>

#define TOKEN  "A1E-fEZr6tGhxt7tq0uvAVKJUiNaHpQULx"  // Put here your Ubidots TOKEN
#define ref   "5bcdbf9dc03f970bbe05ble4"
#define laser "5bcdbf7dc03f970bbe05bld4"
#define gas   "5bcdbf74c03f970b3a083bc8"
#define soil  "5bcdbf94c03f970bbe05ble1"
#define temp  "5bcdbf88c03f970bbe05blde"

#define WIFISSID "madhul" // Put here your Wi-Fi SSID
#define PASSWORD "madhuiot" // Put here your Wi-Fi password
int vref,vlaser,vgas,vsoil,vtemp;
Ubidots client(TOKEN);


#define S0 D0
#define S1 D1
#define S2 D2
#define S3 D3

#define analogpin 0
 void setup() {
  Serial.begin(115200);
  client.wifiConnection(WIFISSID, PASSWORD);
  pinMode(analogpin, INPUT);
  pinMode(S0,OUTPUT);
  pinMode(S1,OUTPUT);
  pinMode(S2,OUTPUT);
```

```
  digitalWrite(S0,LOW);
  digitalWrite(S1,LOW);
  digitalWrite(S2,LOW);
  digitalWrite(S3,LOW);

  Serial.begin(115200);
   // put your setup code here, to run once:
}
 void loop() {
  digitalWrite(S0,LOW); // MQ2 GAS SENSOR
  digitalWrite(S1,LOW);
  digitalWrite(S2,LOW);
  digitalWrite(S3,LOW);
  delay(100);
  vgas=analogRead(analogpin);
  client.add(gas,vgas );
    client.sendAll(false);
  Serial.print("gas");Serial.println(vgas);


  digitalWrite(S0,HIGH); // laser
  digitalWrite(S1,LOW);
  digitalWrite(S2,LOW);
  digitalWrite(S3,LOW);
  delay(100);
  vlaser=analogRead(analogpin);
  client.add(laser,vlaser );

  client.sendAll(false);
Serial.print("laser");Serial.println(vlaser);


digitalWrite(S0,LOW);      //temp
digitalWrite(S1,HIGH);
digitalWrite(S2,LOW);
digitalWrite(S3,LOW);
delay(100);
vtemp=analogRead(analogpin);
client.add(temp,vtemp );
  client.sendAll(false);
Serial.print("temp");Serial.println(vtemp);


digitalWrite(S0,HIGH); //FC-28
digitalWrite(S1,HIGH);
digitalWrite(S2,LOW);
digitalWrite(S3,LOW);
delay(100);
vsoil=analogRead(analogpin);
client.add(soil,1024-vsoil );
  client.sendAll(false);
Serial.print("soil");Serial.println(vsoil);

digitalWrite(S0,LOW); //REFRIGERATOR
digitalWrite(S1,LOW);
```

```
digitalWrite(S1,LOW);
digitalWrite(S2,HIGH);
digitalWrite(S3,LOW);
delay(100);
vref=analogRead(analogpin);
client.add(ref,vref );
  client.sendAll(false);
Serial.print("ref");Serial.println(vref);
delay(100);
```

EXPLANATION

```
#include <UbidotsMicroESP8266.h>
```

This is a very important library which is used to call the functions that are useful in uploading the sensor data to Ubidots.

```
#define TOKEN  "A1E-fEZr6tGhxt7tq0uvAVKJUiNaHpQULx"  // Put here your Ubidots TOKEN
#define ref   "5bcdbf9dc03f970bbe05b1e4"
#define laser "5bcdbf7dc03f970bbe05b1d4"
#define gas   "5bcdbf74c03f970b3a083bc8"
#define soil  "5bcdbf94c03f970bbe05b1e1"
#define temp  "5bcdbf88c03f970bbe05b1de"
```

In Ubidots for each and every account there will be a unique token generated and for each and every data variable there will be a unique ID generated, which can be known to the owner only.

```
#define WIFISSID "madhul" // Put here your Wi-Fi SSID
#define PASSWORD "madhuiot" // Put here your Wi-Fi password
int vref,vlaser,vgas,vsoil,vtemp;
Ubidots client(TOKEN);
```

Wi-Fi credentials and global declaration of the variables.

```
#define S0 D0
#define S1 D1
#define S2 D2
#define S3 D3
```

Defining the select lines for 16:1 MUX

```
client.wifiConnection(WIFISSID, PASSWORD);
pinMode(analogpin, INPUT);
```

Connecting to Wi-Fi

```
client.add(gas,vgas );
  client.sendAll(false);
```

To send the sensor datas

And the rest of the code is to just obtain the analog values from sensors via MUX

Code for Arduino UNO

Arduino is used to interface water pump and speaker .We could have used NodeMCU but as we are already using it for getting the data from sensors and uploading the data to IoT platform if we use it again to connect speakers and motors there may be a huge delay in uploading the data

```
#define button 2
#define  led 3
#define  temp 0 //analog
#define  speak 4
#define  gas 4 //analog
void setup() {
  // put your setup code here, to run once:
 pinMode(button,INPUT);
 pinMode(temp,INPUT);
 pinMode(led,OUTPUT);
 pinMode(speak,OUTPUT);
 digitalWrite(led,LOW);
 Serial.begin(9600);

}


 void loop() {
   // put your main code here, to run repeatedly:
   int buttonr=digitalRead(button);  //ref
   Serial.println(buttonr);
   delay(100);
   if(buttonr==1)
   { digitalWrite(led,1);
     }
   if(buttonr==0)
   { digitalWrite(led,0);
     }
```

```
//////////////////////////////////////
 int tempr=analogRead(temp); //ther
 Serial.println(tempr);
 delay(100);


 //////////////////////////////////////
int gasr=analogRead(gas); //gas
 Serial.println(gasr);
 delay(100);
  if(gasr<=350 || tempr>200)
 { digitalWrite(speak,0);
   }
 if(gasr>350 || tempr<=200 )
 { digitalWrite(speak,1);
   }
  ----------------------------------------
```

Code for Light controlled rooms

```
#include <UbidotsMicroESP8266.h>

#define TOKEN  "A1E-fEZr6tGhxt7tq0uvAVKJUiNaHpQULx"  // Put her
#define HALLID   "5bcdbfb3c03f970bbe05b1eb"
#define ROOMID   "5bcdbfa9c03f970bdc72bbad"
#define KITCHENID "5bcdbfbfc03f970bbe05b1f3"
#define WIFISSID "madhul" // Put here your Wi-Fi SSID
#define PASSWORD "madhuiot" // Put here your Wi-Fi password
Ubidots client(TOKEN);


void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  client.wifiConnection(WIFISSID, PASSWORD);
  pinMode(D0,OUTPUT);
  pinMode(D1,OUTPUT);
  pinMode(D2,OUTPUT);
 digitalWrite(D0,LOW);
 digitalWrite(D1,LOW);
 digitalWrite(D2,LOW);
  delay(100);


}
```

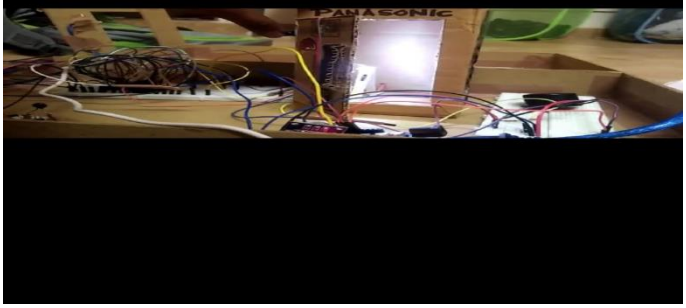
```
void loop() {
  // put your main code here, to run repeatedly:
  int x=client.getValue(HALLID); //recieve data command
  if(x==1)
  {
  digitalWrite(D0,1);
  }
  if(x==0)
  {
  digitalWrite(D0,0);
  }


  int y=client.getValue(ROOMID); //recieve data command
  if(y==1){
  digitalWrite(D1,1);
  }
  if(y==0)
  {
  digitalWrite(D1,0);
  }

   int z=client.getValue(KITCHENID); //recieve data command
   if(z==1){
  digitalWrite(D2,1);
```

```
 int z=client.getValue(KITCHENID); //recieve data command
 if(z==1){
digitalWrite(D2,1);
 }
  if(z==0){
digitalWrite(D2,0);
 }
```

In the above code **Client.getvalue** is the function used to get the value from Ubidots

II. SIMULATION AND RESULT:

Once all the circuit connections are made, codes are implemented and then the stimulation took place and the result was obtained.



III. CONCLUSION

Home automation system will bring more convenience and comfort to people's life. In the next few years we will come across many developed automation system. This android-based smart home application will also be implemented widely. The proposed system allows the owner of the house to take decisions instantly from any place around the world. This system minimizes the wastage of electricity, it consumes less time, also it helps the old aged and physically challenged people in doing the basic physical household works on their own.

IV. ACKNOWLEDGEMENT

We would like to thank Mr.Thamilmaran, for his kind efforts and continuous guidance which helped us in accomplishing the project.

V. REFERENCES

http://esp8266.com

http://arduino.cc

http://arduino.org

http://openhomeautomation.com

https://www.irjet.net/archives/V4/i3/IRJET-V4I3655.pdf

http://inpressco.com/wp-content/uploads/2017/05/Paper33962-967.pdf