Electronic Devices & Circuits

Project

On

P.I.R.Motion Sensing Energy Saving Lamp

(Modelling)

Submitted towards partial fulfillment of the requirement for

the award of the degree of

Bachelor of Technology

In

Electrical Engineering

Submitted by :

Mudit Tripathi
2K20/EE/171

Rishu Yadav
2K20/EE/218

Under the Supervision Of

Prof.Ashish Rajeshwar Kulkarni

Department of Electrical Engineering

Delhi Technical University
Bawana Road, Delhi - 110042

Candidate's Declaration

We hereby certify that the work, which is presented in Project entitled **P.I.R.Motion Sensing Lamp Energy Saving Lamp (Modelling)** in fulfilment of the requirement for the award of the Degree of Bachelor of Technology in Electrical Engineering and submitted to the Department of Electrical Engineering, Delhi Technological University, Delhi is the record of my/our own, carried out under the supervision of **Prof.Ashish Rajeshwar Kulkarni**

The work presented in this report has not been submitted and not under consideration for the award for any other course/degree of this or any other Institute/University.

Mudit Tripathi

2K20/EE/171

B. Tech, Electrical Engineering

Rishu Yadav

2K20/EE/218

B. Tech, Electrical Engineering

Subject Teacher Certificate

To the best of my knowledge, the report comprises original work and has not been submitted in part or full for any Course/Degree to this university or elsewhere as per the candidate's declaration.

Place: Delhi Subject: EDC

Teacher Name: **Prof.Ashish Rajeshwar Kulkarni**

Signature Date: 7 Nov. 2021

Acknowledgement

The success & outcome of this work required a lot of guidance & assistance from many books and websites, we are delighted to have got this all along with the completion of the project. All that is done is only due to such supervision and assistance & we shall not forget to thank them. We respect and thank**, Department of Electrical Engineering**, Delhi Technological University, for providing an opportunity to do the project work and give us all support and guidance, which made us complete the project duly. We are incredibly thankful to **Prof.Ashish Rajeshwar Kulkarni** for providing such vested support and guidance. We are thankful & fortunate enough to get constant encouragement, support and guidance from the department which helped us in doing the work till now. Also, we would like to extend our sincere esteems to all the valuable suggestions put in by the peers and for their timely support.

We would like to extend our gratitude to Mr. Ashish Rajeshwar Kulkarni who gave us the golden opportunity to do this project on the topic “Energy Saving Street light ”, which also helped us in doing a lot of Research.

Mudit Tripathi

2K20/EE/171

B. Tech, Electrical Engineering

Rishu Yadav

2K20/EE/218

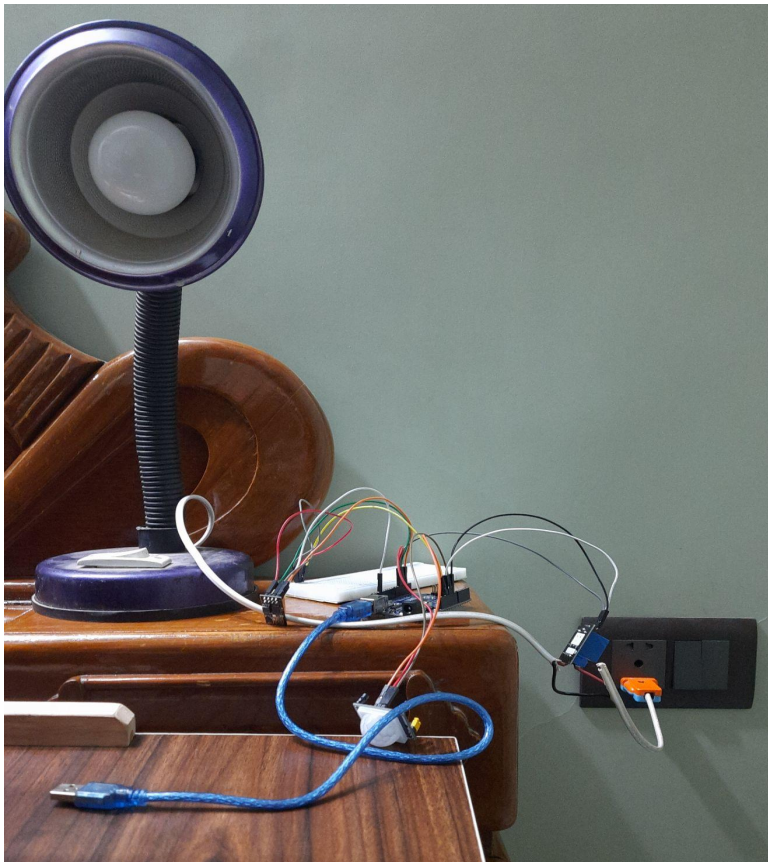
B. Tech, Electrical Engineering

Table of Contents

Abstract

In this project, we have tried to make a prototype of a streetlight that will glow in the night and it will glow in the morning when it encounters any motion. The component used are : Study lamp, Arduino,ESP8266 Module, HC-SR501 PIR motion sensor , Relay 250 V AC, Breadboard,Jumper wires

The main objective of this project is to save electrical energy by using an I.R motion sensor that will allow the lamp to glow whenever it encounters any motion.



INTRODUCTION

1.1 Motivation

In Some parts of our college there are few roads on which the streetlight continuously glows even in day ,which eventually causes the waste of electricity ,So to rectify that we made this “Energy Saving Lamp” from which we can think of a streetlight which will glow at night when any motion will take place and in the daylight it will automatically turn off.

The existing system of Street lights is a manual switching system.Sometimes people forget to turn off these lights due to which they glow day and night.

1.2 Energy Saving Lamp

This Energy saving lamp/streetlight plays a very important role in every field; it can be used in our homes which can help to lower down the electricity bills. In school/college so that the light will only shine when there will be motion and when there will no motion , it will not glow .Moreover, we can easily adjust the sensitivity and the duration of the light.

1.2.1 Advantages

·OFF when not in Use.

·Less electricity consumption.

· Easier to manage , automate the on/off.

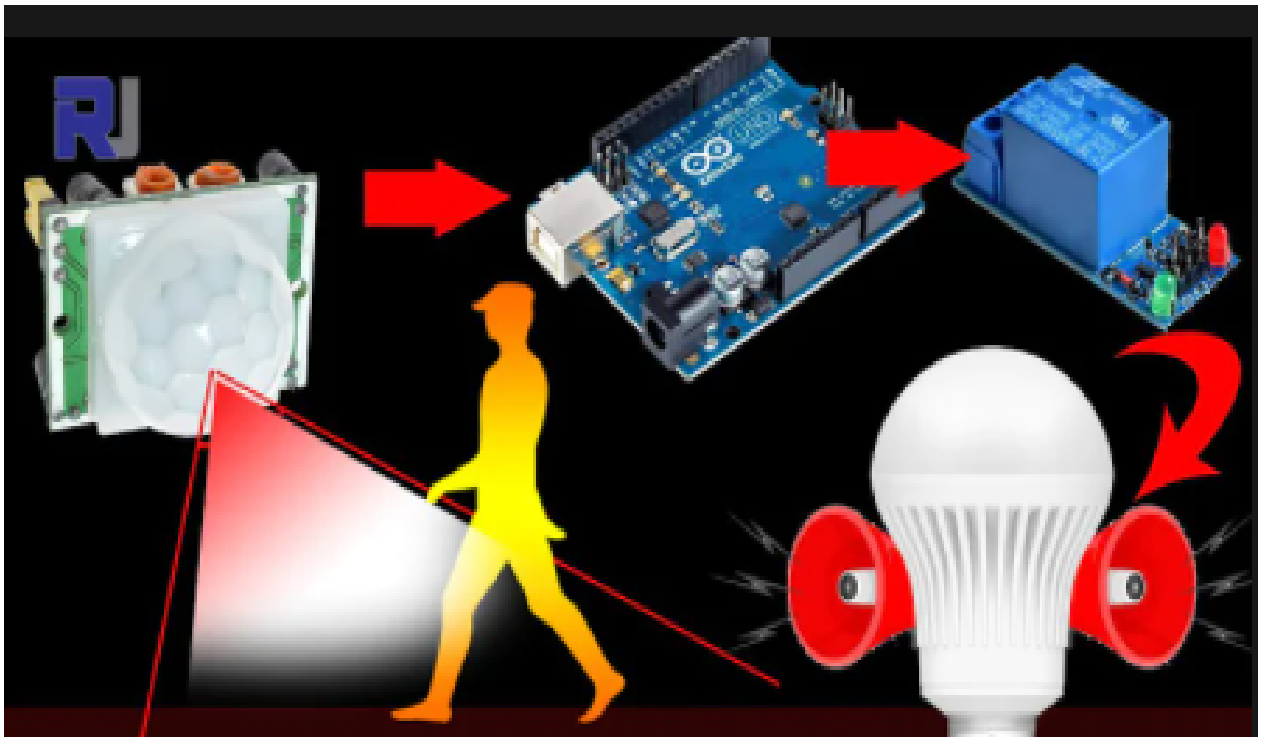
· Cost effective.

· Avoid time theft

· Real time data

4) WORKING

Whenever any motion is detected by the HC-SR501 PIR motion sensor then it will send signal to the arduino microcontroller and arduino will send that signal to the relay and relay will turn on/off the streetlight



INSPIRATION

We often encounter some of the street light in specific area That glows even in the morning which leads to the unusual electricity wastage so to minimize this kind of electricity wastage we have made a prototype of a street light using motion sensor which will glow in the night and in the morning it will glow only when there is any motion.

Steps How the streetlight will glow

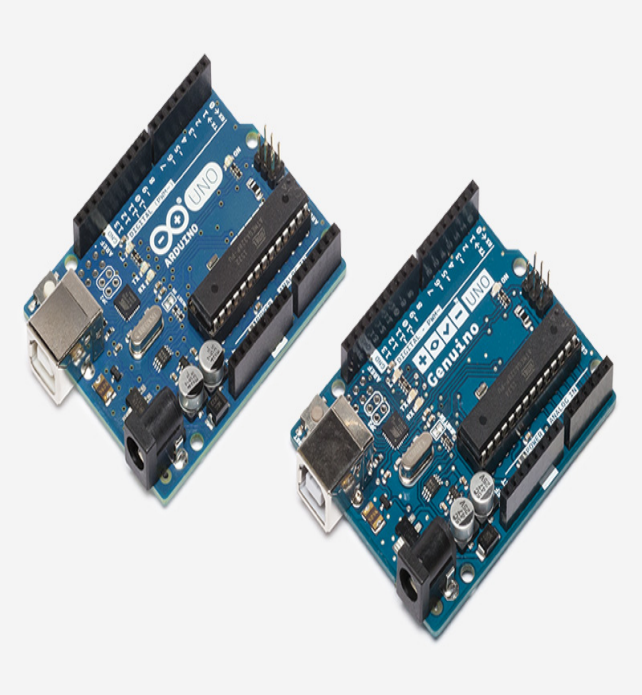
- IR sensor will sense the motion and will send to arduino
- Arduino will send the signal to relay
- Relay will let the current to flow
- This is how the streetlight will glow when the motion happens

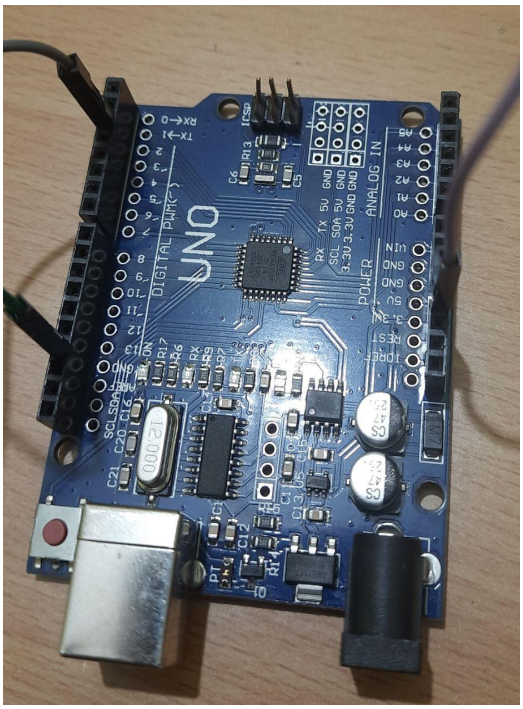
5) COMPONENTS USED

Material Used
1) Arduino Uno
2) P.I.R motion Sensor

3) Esp8266
4) Relays
5) Cable
6) Jumper - Wires & Breadboard

Arduino



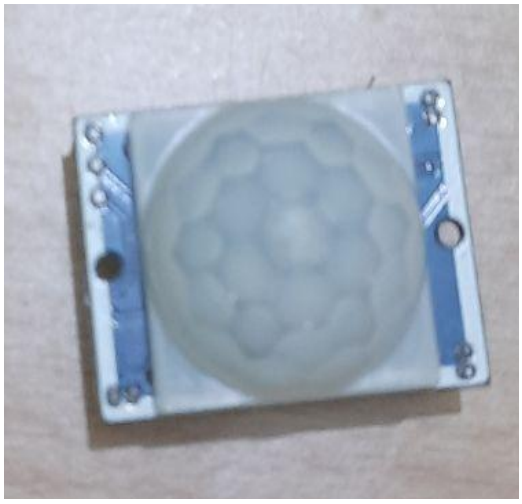


Arduino/Genuino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB

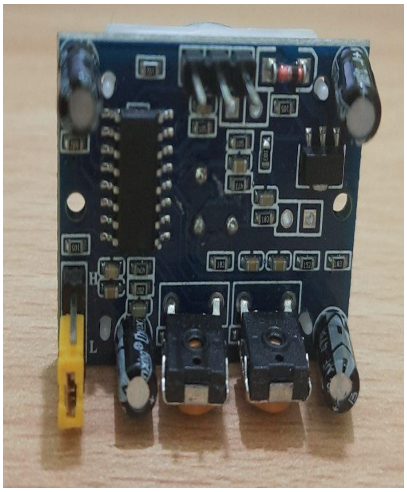
connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

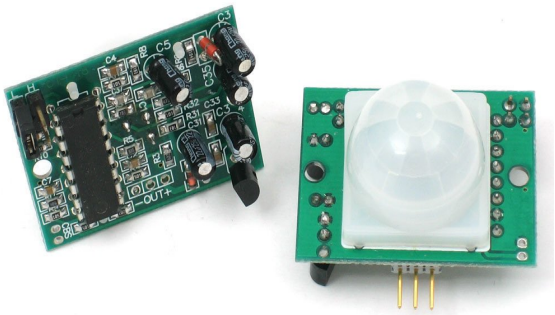
"Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of board.

P.I.R Motion Sensor



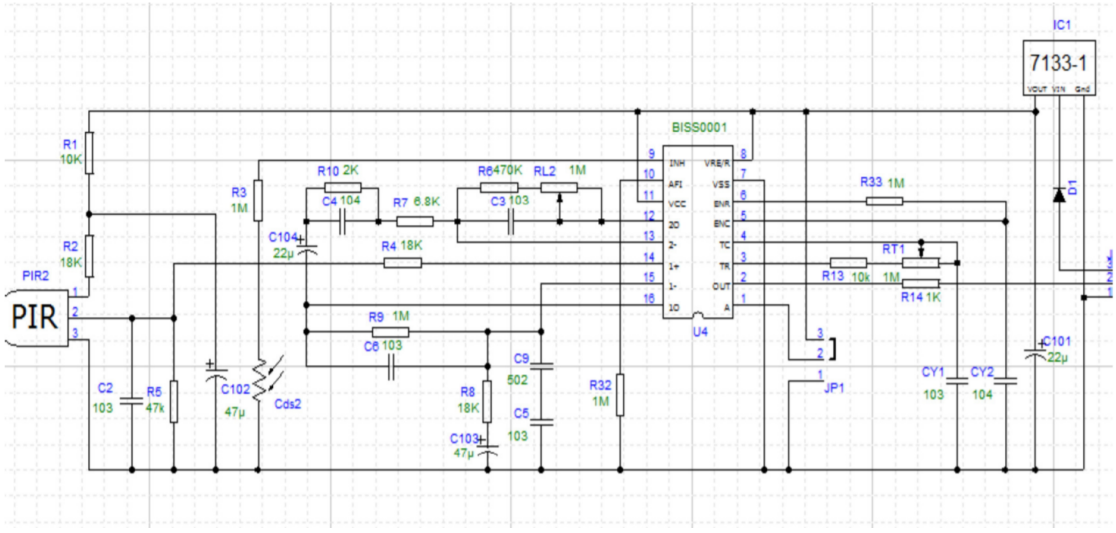
PIR sensors allow you to sense motion, almost always used to detect whether a human has moved in or out of the sensor's range. They are small, inexpensive, low-power, easy to use and don't wear out. For that reason they are commonly found in appliances and gadgets used in homes or businesses. They are often referred to as PIR, "Passive Infrared", "Pyroelectric", or "IR motion" sensors.



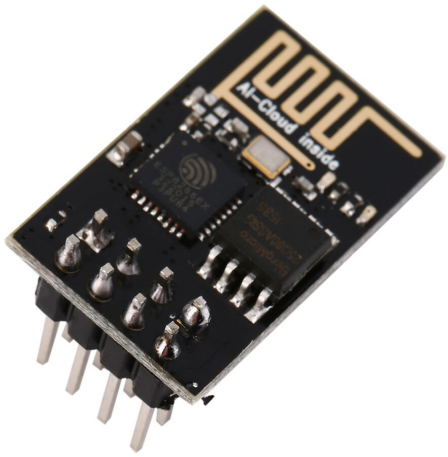


PIRs are basically made of a pyroelectric sensor (which you can see below as the round metal can with a rectangular crystal in the center), which can detect levels of infrared radiation. Everything emits some low level radiation, and the hotter something is, the more radiation is emitted. The sensor in a motion detector is actually split in two halves. The reason for that is that we are looking to detect motion (change) not average IR levels. The two halves are wired up so that they cancel each other out. If one half sees more or less IR radiation than the other, the output will swing high or low.

HC-SR501 PIR MOTION DETECTOR



ESP8266

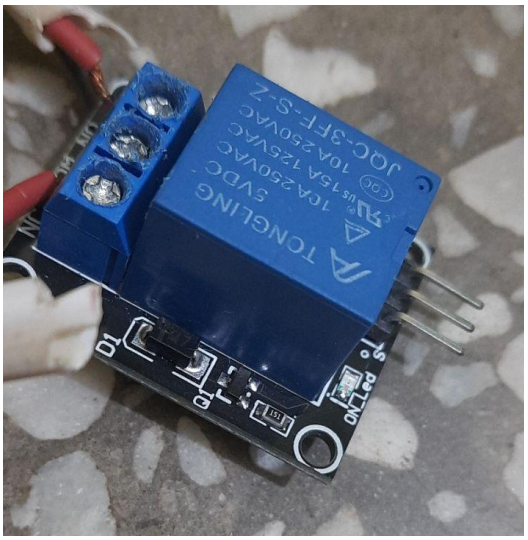


The ESP8266 is a low-cost Wi-Fi microchip, with built-in TCP/IP networking software, and microcontroller capability, produced by Espressif Systems[1] in Shanghai, China.

The chip first came to the attention of Western makers in August 2014 with the ESP-01 module, made by a third-party manufacturer Ai-Thinker. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands. However, at first, there was almost no English-language documentation on the chip and the commands it accepted.The very low price and the fact that there were very few external components on the module, which suggested that it could eventually be very inexpensive in volume, attracted many hackers to explore the module, the chip, and the software on it, as well as to translate the Chinese documentation.

The ESP8285 is a similar chip with a built-in 1 MiB flash memory, allowing the design of single-chip devices capable of connecting via Wi-Fi.

Relays



A relay is an electrically operated switch. It consists of a set of input terminals for a single or multiple control signals, and a set of operating contact terminals. The switch may have any number of contacts in multiple contact forms, such as make contacts, break contacts, or combinations thereof.

Relays are used where it is necessary to control a circuit by an independent low-power signal, or where several circuits must be controlled by one signal. Relays were first used in long-distance telegraph circuits as signal repeaters: they refresh the signal coming in from one circuit by transmitting it on another circuit. Relays were used extensively in telephone exchanges and early computers to perform logical operations.

The traditional form of a relay uses an electromagnet to close or open the contacts, but other operating principles have been invented, such as in solid-state relays which use semiconductor properties for control without relying on moving parts. Relays with calibrated operating characteristics and sometimes multiple operating coils are used to protect electrical circuits from overload or faults; in modern electric power systems these functions are performed by digital instruments still called protective relay

LAMP / Demo-Street light

- **Shade:** directs and controls light from the bulb.
- **Pole/tube:** Structure providing height to which shade and bulb attach.
- **Adjuster:** Swivel component to adjust direction of shade.
- **Break:** Component that secures / loosens top pole from bottom poll to adjust floor lamp height.
- **Husk:** Attaches poll to floor lamp base.
- **Base:** Usually fairly heavy and wide (in relation to lamp poll) to anchor the floor lamp and keep it upright.

What Does **Port 80** Mean in the code?

Port 80 is the port number assigned to commonly used internet communication protocol, Hypertext Transfer Protocol (HTTP). It is the port from which a computer sends and receives Web client-based communication and messages from a Web server and is used to send and receive HTML pages or data.

- https://www.electronicshub.org/esp8266-at-commands/#:~:text=The%20AT%20Commands%20of%20the%20ESP8266%20WiFi%20Module,least%20a%20few%20important%20ones%20if%20not%20all.

- https://www.espressif.com/sites/default/files/4b-esp8266_at_command_examples_en_v1.3.pdf#:~:text=0%20means%20that%20remote%20IP%20and%20port%20is,0%2C%20it%20will%20send%20to%20a%20new%20PC.

Some AT commands used in this project for esp8266

1) Single Connection as TCP Client

- **Set WiFi mode**:

```
AT+CWMODE=3      // softAP+station mode
Response :OK
```

- **Connect to router**:

```
AT+CWJAP="SSID", "password"      // SSID and password of router
Response :OK
```

- **Query device's IP**:

```
AT+CIFSR
Response :192.168.3.106   // Device got an IP from router.
```

- **ESP8266 connect to server as a client:**

```
AT+CIPSTART="TCP", "192.168.3.116", 8080  //protocol、server IP & port
Response :OK
```

- **Send data:**

```
AT+CIPSEND=4   // set date length which will be sent,  such as 4 bytes

>DGFY         // enter the data,  no CR
Response :SEND OK
```

Void setup

- void setup () The function, setup (), as the name implies, is used to set up the Arduino board. The Arduino executes all the code that is contained between the curly braces of setup () only once. Typical things that happen in setup () are setting the modes of pins, starting

Arduino Code

```
//This project is done by Mudit Tripathi 2K20/EE/171 & Rishu Yadav 2K20/EE/218
//Under the guidance of Prof.Ashish Kulkarni dept. of EE
#include <SoftwareSerial.h>
#define RX 2 //RX reffer to the receiver signal
```

```
#define TX 3 //TX reffer to the Tranfer signal
const int SENSOR_PIN = 5;// the Arduino pin connected to output (middle) wire of sensor
const int RELAY_PIN = 4;// the Arduino pin which is connected to control relay
String AP = "tushar";     // wifi Name that will be Needed by Esp8266 to push Data
String PASS = "yahoo123"; // AP PASSWORD
String API = "SOQOJM05H891EL8H";   // Write API KEY
String HOST = "api.thingspeak.com"; //Site Name
String PORT = "80"; //most commonly used port numbers in the Transmission Control Protocol (TCP) suite
String field = "field1";
int countTrueCommand;
int countTimeCommand;
boolean found = false; //Bool variable
int valSensor = 1; //Integer variable
SoftwareSerial esp8266(RX,TX);

void setup() {  //Void setup function to setup arduino board
  Serial.begin(9600); //"9600 baud" means that the serial port is capable of transferring a maximum of 9600 bits per second.
```

```
 esp8266.begin(115200);
 pinMode(SENSOR_PIN, INPUT);// Define SENSOR_PIN as Input
from sensor
 pinMode(RELAY_PIN, OUTPUT);// Define RELAY_PIN as
OUTPUT for relay
 sendCommand("AT",5,"OK");
 sendCommand("AT+CWMODE=1",5,"OK"); //Setting up the arduino
to station mode
 sendCommand("AT+CWJAP=\""+ AP +"\",\""+ PASS +"\"",20,"OK");
//AT+CWJAP ->  connect to an Access Point (like a router).
}

void loop() { //in the void loop you'll write your main program, knowing
that the initialization is already done
 valSensor = getSensorData(); // Calling a function by the name of
getsensor data
 String getData = "GET /update?api_key="+ API +"&"+ field
+"="+String(valSensor); //Storing all data in a string
sendCommand("AT+CIPMUX=1",5,"OK");//enable or disable multiple
TCP Connections ,AT+CIPMUX =1 for Multiple connection
```

```
 sendCommand("AT+CIPSTART=0,\"TCP\",\""+ HOST +"\","+
PORT,15,"OK"); //establishing one of the three connections: TCP,
UDP or SSL.
 sendCommand("AT+CIPSEND=0," +String(getData.length()+4),4,">");
//Start sending data
 esp8266.println(getData);delay(1500);countTrueCommand++;
 sendCommand("AT+CIPCLOSE=0",5,"OK"); //closes TCP connection
}

int getSensorData(){ //Function that is called in the upper section
  int motion =digitalRead(SENSOR_PIN);// read the sensor pin and
stores it in "motion" variable

  // if motion is detected
  if(motion){
    Serial.println("Motion detected");
    digitalWrite(RELAY_PIN, HIGH);// Turn the relay ON
    return 0;
  // if motion is not detected
  }else{
     Serial.println("===nothing moves");   //To print on the  serial monitor
"Noting moves"
```

```
    digitalWrite(RELAY_PIN,LOW);// Turn the relay OFF
    return 500;
  }
  delay(500);  //Delaying for 0.5 second or 500 milliseconds
}

void sendCommand(String command, int maxTime, char readReplay[]) {
  Serial.print(countTrueCommand);
  Serial.print(". at command => ");
  Serial.print(command);
  Serial.print(" ");
  while(countTimeCommand < (maxTime*1))
  {
    esp8266.println(command);//at+cipsend
    if(esp8266.find(readReplay))//ok
    {
      found = true;
      break;
    }

    countTimeCommand++;
  }
```

```
if(found == true)
{
  Serial.println("OYI");
  countTrueCommand++;
  countTimeCommand = 0;
}

if(found == false)
{
  Serial.println("Fail");
  countTrueCommand = 0;
  countTimeCommand = 0;
}

 found = false;
}
```

Output in the Serial Monitor

COM5

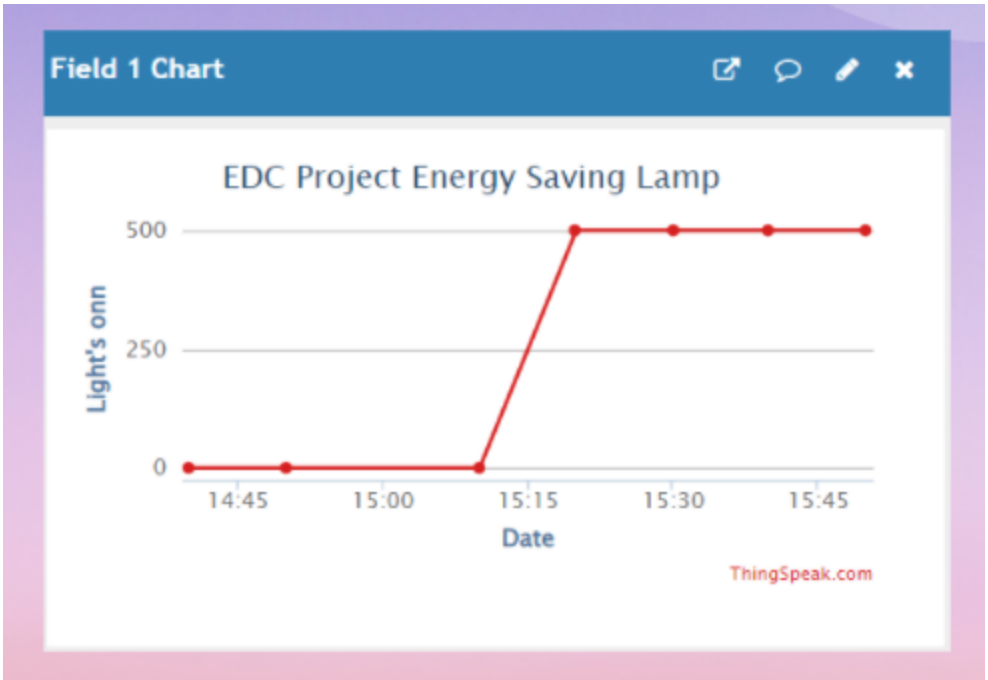
```
4. at command => AT+CIPSEND=0,49 OYI
6. at command => AT+CIPCLOSE=0 OYI
Motion detected
7. at command => AT+CIPMUX=1 OYI
8. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
9. at command => AT+CIPSEND=0,51 OYI
11. at command => AT+CIPCLOSE=0 Fail
===nothing moves
0. at command => AT+CIPMUX=1 OYI
1. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
2. at command => AT+CIPSEND=0,49 OYI
4. at command => AT+CIPCLOSE=0 Fail
===nothing moves
0. at command => AT+CIPMUX=1 OYI
1. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
2. at command => AT+CIPSEND=0,49 OYI
4. at command => AT+CIPCLOSE=0 Fail
===nothing moves
0. at command => AT+CIPMUX=1 OYI
1. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
2. at command => AT+CIPSEND=0,49 OYI
4. at command => AT+CIPCLOSE=0 Fail
===nothing moves
0. at command => AT+CIPMUX=1 OYI
1. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
2. at command => AT+CIPSEND=0,49 OYI
4. at command => AT+CIPCLOSE=0 OYI
Motion detected
5. at command => AT+CIPMUX=1 OYI
6. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
7. at command => AT+CIPSEND=0,51 OYI
9. at command => AT+CIPCLOSE=0
```

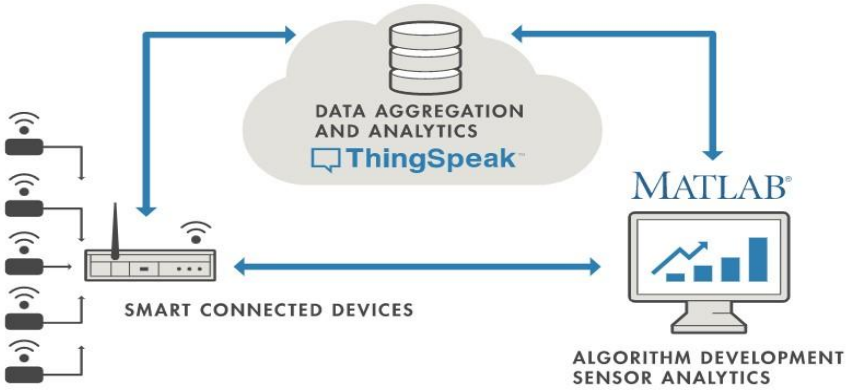
Send

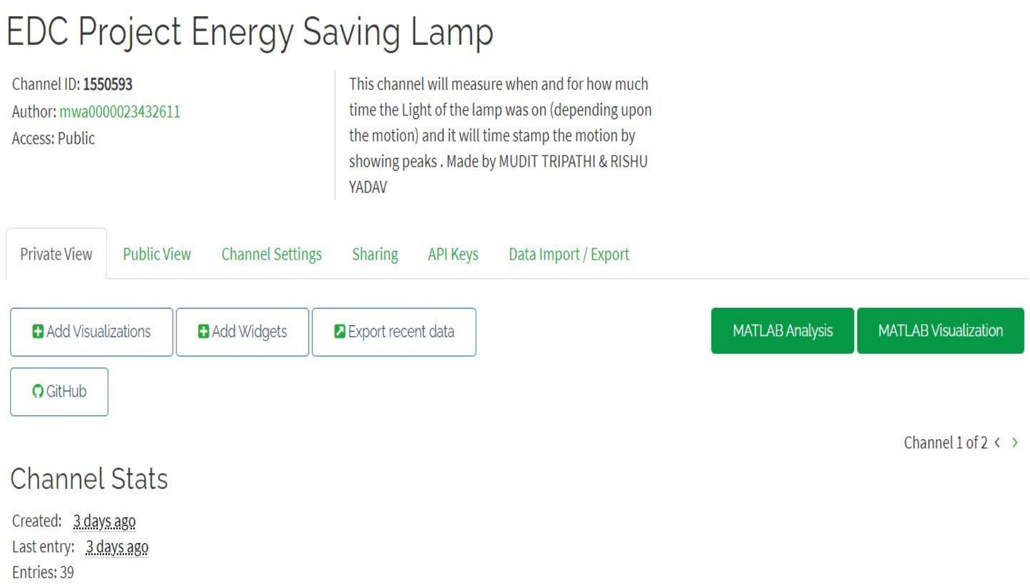
Autoscroll | Show timestamp | Newline | 9600 baud | Clear output

Innovatively Plotting Thingspeak Data Using ESP 8266

Link to the Project Channel:





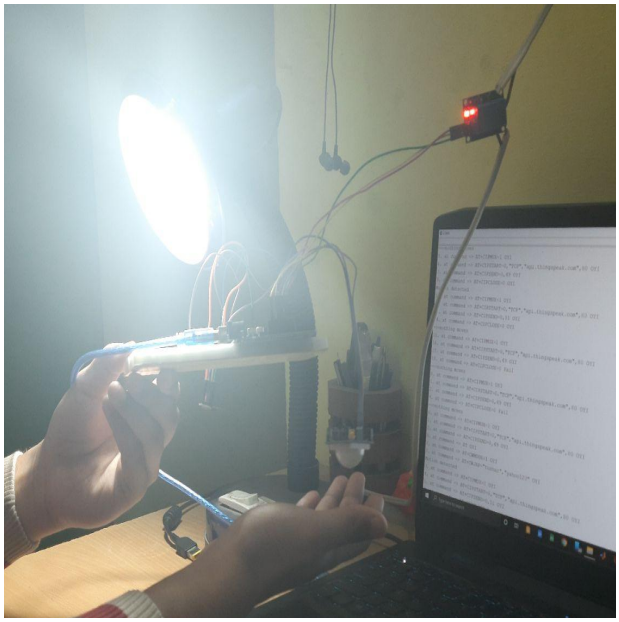

EDC Project Energy Saving Lamp
Channel ID: 1550593
Author: mwa0000023432611
Access: Public
This channel will measure when and for how much time the Light of the lamp was on (depending upon the motion) and it will time stamp the motion by showing peaks . Made by MUDIT TRIPATHI & RISHU YADAV
Private View
Public View
Channel Settings
Sharing
API Keys
Data Import / Export
Add Visualizations
Add Widgets
Export recent data
MATLAB Analysis
MATLAB Visualization
GitHub
Channel 1 of 2
Channel Stats
Created: 3 days ago
Last entry: 3 days ago
Entries: 39

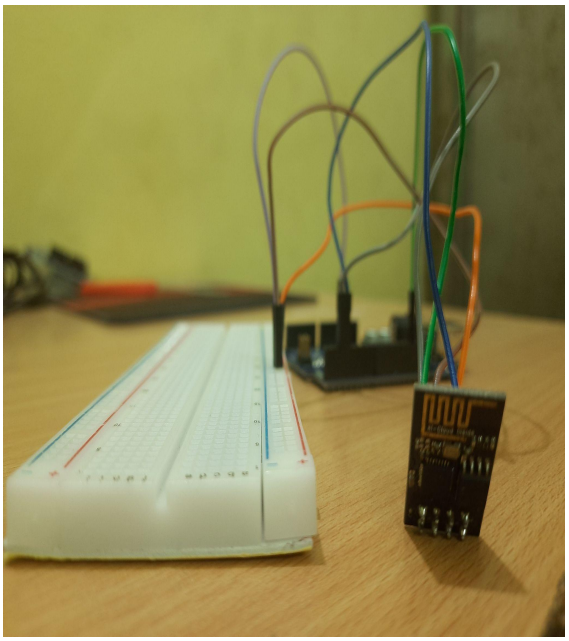

COM5
Send
4. at command => AT+CIPSEND=0,49 OYI
6. at command => AT+CIPCLOSE=0 OYI
Motion detected
7. at command => AT+CIPMUX=1 OYI
8. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
9. at command => AT+CIPSEND=0,51 OYI
11. at command => AT+CIPCLOSE=0 Fail
===nothing moves
0. at command => AT+CIPMUX=1 OYI
1. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
2. at command => AT+CIPSEND=0,49 OYI
4. at command => AT+CIPCLOSE=0 Fail
===nothing moves
0. at command => AT+CIPMUX=1 OYI
1. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
2. at command => AT+CIPSEND=0,49 OYI
4. at command => AT+CIPCLOSE=0 Fail
===nothing moves
0. at command => AT+CIPMUX=1 OYI
1. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
2. at command => AT+CIPSEND=0,49 OYI
4. at command => AT+CIPCLOSE=0 Fail
===nothing moves
0. at command => AT+CIPMUX=1 OYI
1. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
2. at command => AT+CIPSEND=0,49 OYI
4. at command => AT+CIPCLOSE=0 OYI
Motion detected
5. at command => AT+CIPMUX=1 OYI
6. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OYI
7. at command => AT+CIPSEND=0,51 OYI
9. at command => AT+CIPCLOSE=0
Autoscroll
Show timestamp
Newline
9600 baud
Clear output

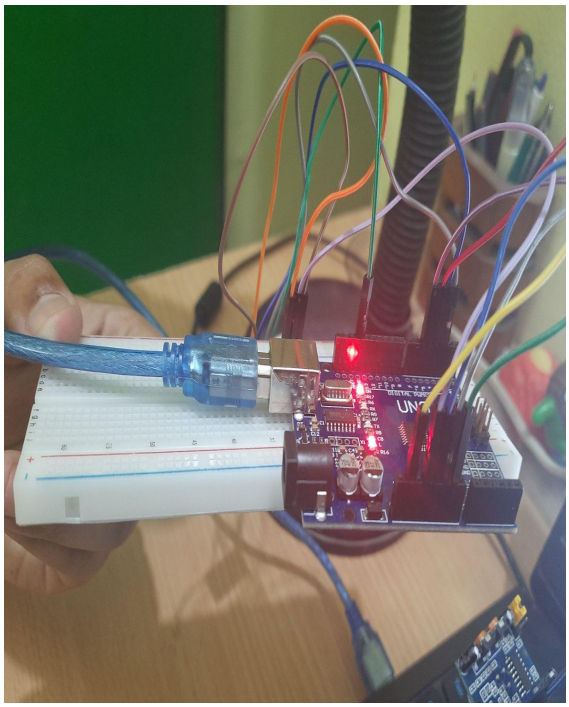
DATASHEETS

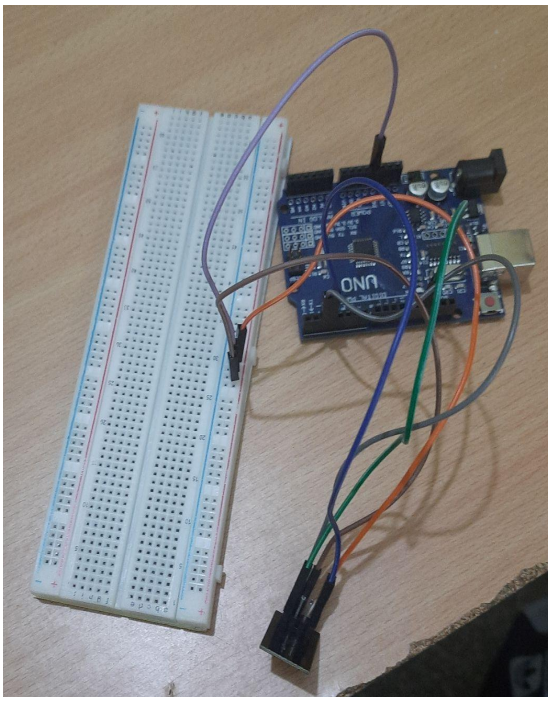
- HC-SR501 P.I.R Motion sensor
- ESP8266 DATASHEET
- Arduino DATASHEET
- Relay DATASHEET

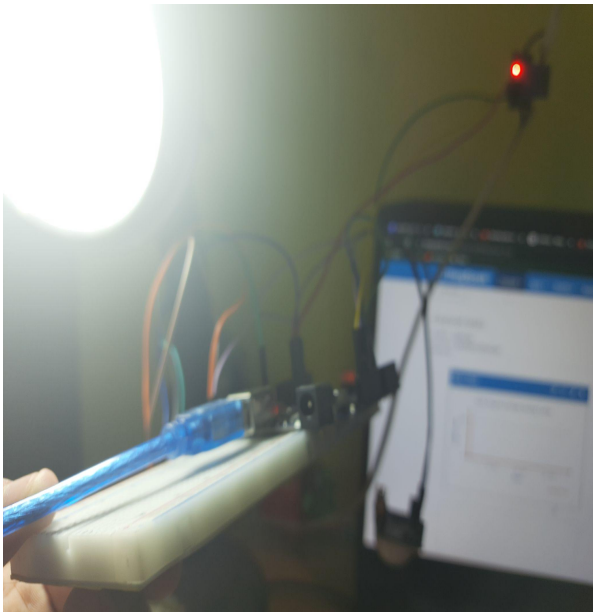
Circuit Diagram & Images of Working Model

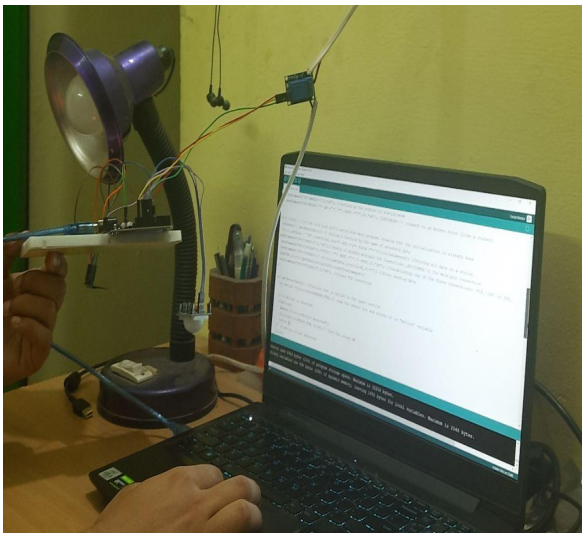


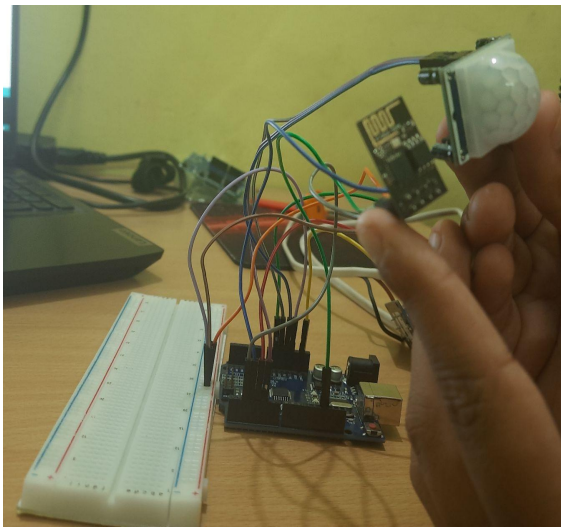


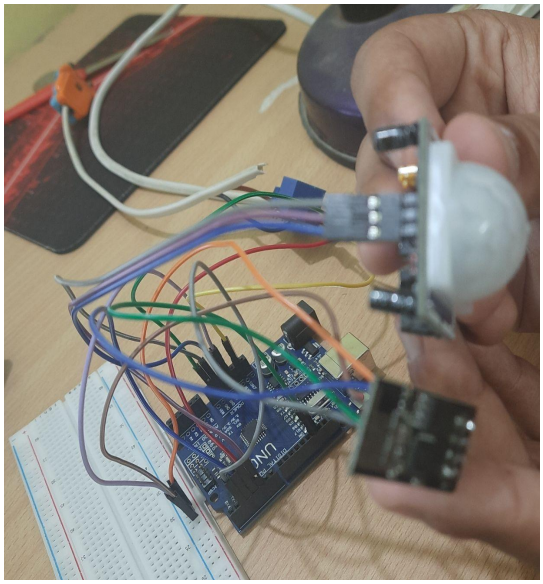


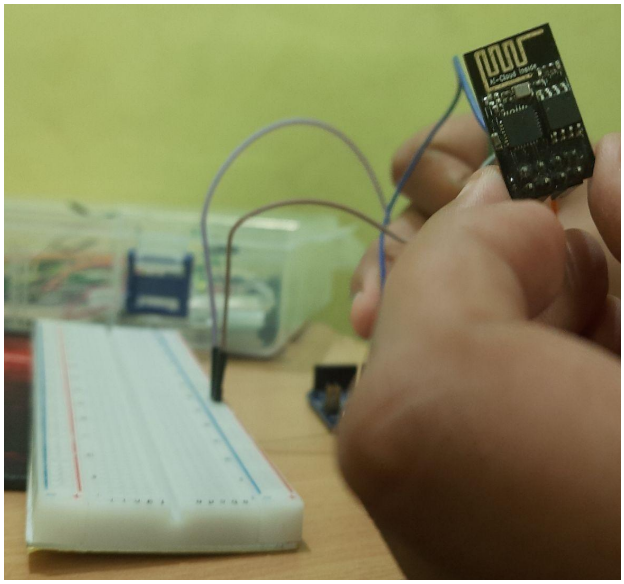

UNO











Conclusion & Future Scope

In this project we have made it glow when the motion occurs but we have to set the street light to glow even the night and for that we require a LDR

.Photoresistors work based off of the principle of photoconductivity. Photoconductivity is an optical phenomenon in which the material's conductivity is increased when light is absorbed by the material.
When light falls i.e. when the photons fall on the device, the electrons in the valence band of the semiconductor material are excited to the conduction band. These photons in the incident light should have energy greater than the bandgap of the semiconductor material to make the electrons jump from the valence band to the conduction band.

Hence when light having enough energy strikes on the device, more and more electrons are excited to the conduction band which results in a large number of charge carriers. The result of this process is more and more current starts flowing through the device when the circuit is closed and hence it is said that the resistance of the device has been decreased. This is the most common working principle of LDR . This is how we will complete our project so that it will glow in daylight and even in the night.

References

- http://robojax.com/sites/default/files/docs/robojax-HC-SR501-motion-LHI778-manual.pdf

- http://robojax.com/sites/default/files/docs/robojax-HC-SR501-motion-manual.pdf

- https://thingspeak.com/channels/1550593/private_show