A Project Report

on

AUTOMATIC ILLUMINATION(Using Arduino)

Submitted in partial fulfillment of the

requirement for the award of the degree of

B.Tech CSE with Specialization in IOT & AIML

Under The Supervision of
Name of Supervisor: Dr. KM Baalamurugan
Assistant Professor

Submitted By

Javin Singh
19SCSE1130003
Somya Agnihotri
19SCSE1180022

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
December, 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

We hereby certify that the work which is being presented in the project, entitled **"AUTOMATIC ILLUMINATION"** in partial fulfillment of the requirements for the award of the B.Tech submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of October 2021 to December 2021, under the supervision of Dr. KM Baalamurugan Assistant Professor, Department of Computer Science and Engineering of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the project has not been submitted by us for the award of any other degree of this or any other places.

Javin Singh (19SCSE1130003)

Somya Agnihotri (19SCSE1180022)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr. KM Baalamurugan
Assistant Professor

CERTIFICATE

The Final Project Viva-Voce examination of Javin Singh,Somya Agnihotri has been held on _____ and his/her work is recommended for the award of B.Tech in CSE Iot & Aiml.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date: December,2021
Place: Greater Noida

Acknowledgement

This is the matter of great privilege for all of us to submit this project entitled "**Automatic Illumination**" We take pleasure in expressing our deep sense of gratitude for providing necessary guidance to Dr. KM Baalamurugan in the department of Computer Science & Engineering at GALGOTIAS UNIVERSITY, Greater Noida for his kind and constant encouragement made it possible for us to complete this project work.

He has given us a pragmatic sense to look into the matter and we are also highly obliged for his persistence in making the project complete. It gives us great pleasure to extend our sincere thanks to our colleagues, who helped us directly or indirectly to complete the project.

Last but not the least, we would like to express our gratitude towards our parents for their kind co- operation and encouragement which helped us in completion of this project.

Javin Singh (19SCSE1130003)
Somya Agnihotri (19SCSE1180022)

Abstract

Considering the present scenario, how the developing technology is shaping the future of mankind and the amount of manpower required to fulfil these tasks. Major manpower means massive buildings and offices. As of now how everybody owns their personal vehicle, these buildings consists of huge and confusing parking systems. Most of these systems being underground, it is mostly dark and require a major lighting system which consumes most of the electrical resource. This project aims to rectify this issue. A system that illuminates only if a vehicle passes by or by some respective movement. When driving the illumination of the motion sensor zone as well as front and rear zone is switched on.

***Keywords*: Arduino, PIR Sensor, Parking system, smart devices, automation illumination**

List of Figures

Table Of Contents

Introduction:

Humans shape technology, technology shapes the future of mankind and the baseline for technology is electrical energy. A major number of advancements have been done to preserve this energy, moreover on the methods on how to transport this energy. Major advancements need manpower and huge manpower needs a place to reside and these places are massive buildings with complex parking systems. In these parking systems a lot of electrical energy is wasted.

As major of these parking systems are underground, it needs an illuminating lighting resource, which is turned on for the whole day or night, which we look practically is a waste of the energy, as people just park their vehicles either in beginning or end of the day, same can be considered for malls and complex during the lunch time.

This project aims to save the electrical energy that is lost in the majority of the parking systems prevalent. All the advancements that are done in the field of home automation have somewhat or the other helped to achieve greater things and at the same time preserve our natural as well as man-made resources.

In the system being developed, we will use transmitters. The transmitters with help of sensors and timers, with coordination of receivers will provide us with an output that will help us define the load control.

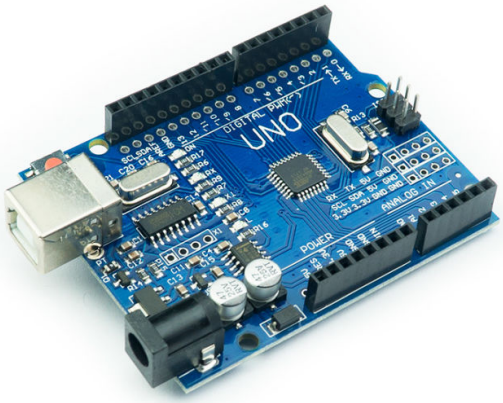
The control of light in an underground parking or a room area. So while walking across the sensors or driving across the sensors, the motion sensor will detect the presence of a human being or a vehicle and automatically illuminate that particular area. The area in which the object is present, with its rear zone as well as front zone, will be illuminated efficiently.

This will do the job and at the same time help us to preserve the electrical resource which is the baseline of almost every technology. Hence, basic need for us as humans to develop and shape our bright future. So, when the car or person has passed the motion sensor signal is reset and accordingly the illumination of the front and rear zones is switched off.

In addition to this, the sensor zone which is illuminated will be turned off by a timer. Also a led will indicate the status of PIR sensor, which is the motion sensor and the output signal for the light illuminating object used.

Module Description

Arduino



Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language , and the Arduino Software (IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments.

All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many others offer similar functionality. All of these tools take the messy details of microcontroller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

- Inexpensive - Arduino boards are relatively inexpensive compared to other

microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50

- Cross-platform - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.

- Simple, clear programming environment - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.

- Open source and extensible software - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.

- Open source and extensible hardware - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money.

PIR Sensor

Passive InfraRed sensors can detect movement of objects that radiate IR light (like human bodies). Therefore, using these sensors to detect human movement or occupancy in security systems is very common. Initial setup and calibration of these sensors takes about 10 to 60 seconds.

The HC-SR501's infrared imaging sensor is an efficient, inexpensive and adjustable module for detecting motion in the environment.

The small size and physical design of this module allow you to easily use it in your project.

The output of the PIR motion detection sensor can be connected directly to one of the Arduino (or any microcontroller) digital pins. If any motion is detected by the sensor, this pin value will be set to “1”. The two potentiometers on the board allow you to adjust the sensitivity and delay time after detecting a movement.

PIR modules have a passive infrared sensor that detects the occupancy and movement from the infrared radiated from the human body. You can use this module in security systems, smart lighting systems, automation, etc. There are different PIR modules available in the market, but all of them are basically the same.

They all have at least a Vcc pin, GND pin, and digital output. In some of these modules, there is a ball like a lens on the sensor that improves the viewing angle.

The motion sensor: Able to detect the movement of people or objects. In most applications, these sensors are mainly used to detect human activities in a specific area.

- **Converts motion into electrical signals**: the sensor either emits stimuli and monitors any changes reflected back, or acquires signals from the moving object itself.

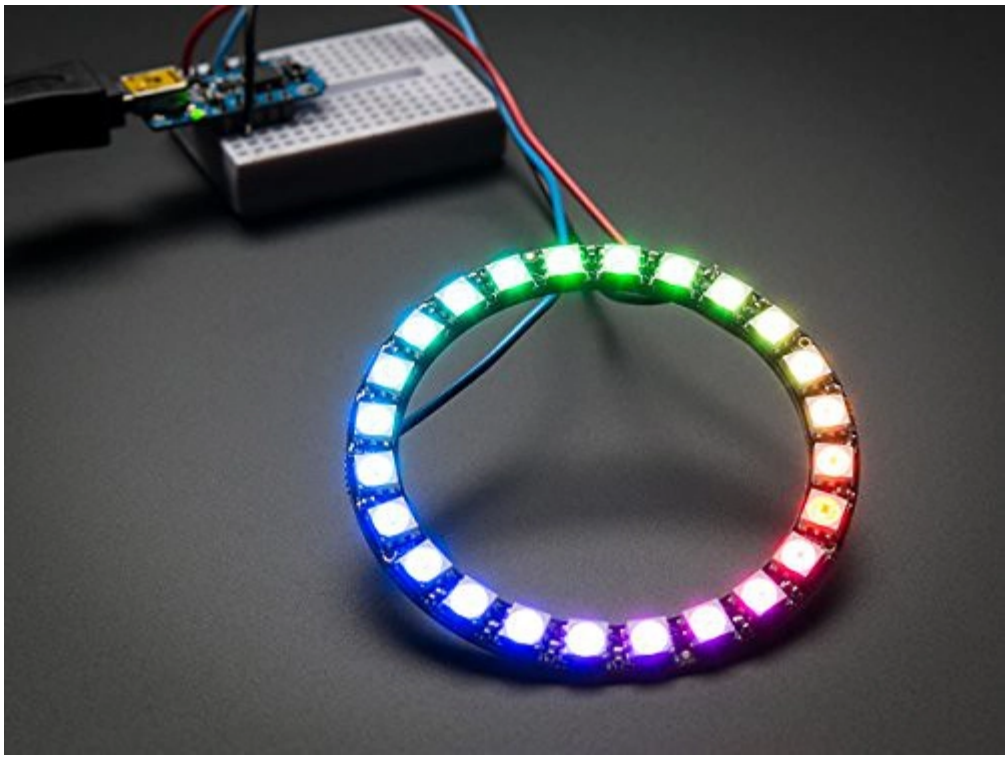
- **Alarm**: Sounds the alarm when people or other objects invade and break the normal state, while others will alarm when they return to normal state after the invasion.

- **Usage**: Security systems all over the world rely on motion sensors to trigger alarms and/or automatic lighting switches, which are usually placed in relatively easy access to buildings, such as windows and gates.

PIR is only one of the technical methods to detect motion, so we will say PIR sensor is a **subset of the motion sensor**.

PIR sensor is small in size, cheap in price, low-power consumption and very easy to understand, which makes it quite popular. A lot of merchants will add “motion” between PIR sensor for the convenience of beginners.

NeoPixel Lights



Incorporating scads of LEDs into an electronic project used to be a hairy prospect, a veritable rat's nest of wires and code. The arrival of dedicated LED driver chips brought welcome relief, offloading grunt work from the microcontroller and allowing one to focus on the application. Much simpler, but still not "Christmas light" simple.

The WS2812 Integrated Light Source — or *NeoPixel* in Adafruit parlance — is the latest advance in the quest for a simple, scalable and affordable full-color LED. Red, green and blue LEDs are integrated alongside a driver chip into a tiny surface-mount package controlled through a single wire. They can be used individually, chained into

NeoPixels don’t just light up on their own; **they require a microcontroller** (such as Arduino) and some programming. We provide some sample code to get you started. To create your own effects and animation, you’ll need some programming practice. If this is a new experience, work through some of the beginning Arduino tutorials to get a feel for the language.

These are **individually addressable LEDs** all housed on a string that can be controlled from a **single pin** on a microcontroller. This means one pin can control all of the LEDs colors and which LEDs are on at any given time. When compared to a **normal RGB LED** you'll notice that we need 3 pins to control the **RedGreen** and **Blue** value and all LEDs have to remain **on** or **off**. So as you can see using individually addressable can help create some cool effects.

Lots of companies sell strips of addressable LEDs however the most popular is the **NeoPixel** by Adafruit. Adafruit produces a line of addressable LED strip and also supplies the library to control them, this is called "Adafruit NeoPixel", there are other companies that have produced libraries to control these addressable LEDs.

We can use that to modify each LED in a sequence – or every other LED, or whatever you like.

- So, first, we tell the sketch to include Adafruit's library.
- Then, we define two variables: the data pin we're using (any PWM pin), and the

number of pixels in the strip.

- In the next bit, we initialize the strip as a new object, pixels . (You can call it whatever you want, of course.)
- Then, we set a delay value, which will be used later to pause after lighting up each LED.

There's a little bit of board-specific code in the setup code, and then we tell the NeoPixel library to start communicating with this strip.

Loop time:-

Loop is kicked off when the variable i is equal to 0.

Note that this is actually where we're defining the variable, as well – so it starts as 0. You can tell because of the int marker, and also because there's only a single = sign. (One = sets the first thing equivalent to the second, as in i=0 ; two == checks to see whether the two sides are equivalent.)

So, this for loop says:

- If i is () when you hit this block of code (and it is), then as long as i is less than the declared number of pixels, bump it up by 1...
- then, set the color of the first pixel in the pixel strand to green...
- then, actually push that color to the pixel...
- then, wait 500ms (from the delayval setting)...
- and repeat for each successive pixel, until you max out the strand.

Literature Survey

Internet of things is a system that involves multiple devices working together without any human intervention over the Internet. Iot has evolved in multiple ways just to provide comfort to human beings. Making their lifestyle more efficient and private. The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.

A *thing* in the internet of things can be a person with a heart monitor implant, a farm animal with a biochip transponder, an automobile that has built-in sensors to alert the driver when tire pressure is low or any other natural or man-made object that can be assigned an Internet Protocol (IP) address and is able to transfer data over a network.

Increasingly, organizations in a variety of industries are using IoT to operate more efficiently, better understand customers to deliver enhanced customer service, improve decision-making and increase the value of the business.

An IoT ecosystem consists of web-enabled smart devices that use embedded systems, such as processors, sensors and communication hardware, to collect, send and act on data they acquire from their environments. IoT devices share the sensor data they collect by connecting to an IoT gateway or other edge device where data is either sent

to the cloud to be analyzed or analyzed locally. Sometimes, these devices communicate with other related devices and act on the information they get from one another. The devices do most of the work without human intervention, although people can interact with the devices -- for instance, to set them up, give them instructions or access the data.

The internet of things helps people live and work smarter, as well as gain complete control over their lives. In addition to offering smart devices to automate homes, IoT is essential to business. IoT provides businesses with a real-time look into how their systems really work, delivering insights into everything from the performance of machines to supply chain and logistics operations.

IoT enables companies to automate processes and reduce labor costs. It also cuts down on waste and improves service delivery, making it less expensive to manufacture and deliver goods, as well as offering transparency into customer transactions.

As such, IoT is one of the most important technologies of everyday life, and it will continue to pick up steam as more businesses realize the potential of connected devices to keep them competitive.

The baseline provided by Iot and smart systems to make lifestyle efficient was home automation. It involved the use of sensors and actuators. Sensors did all the sensing stuff, sensing the environment and providing us with data.

With the help of this data, actuators performed their job, may it be turning on/off a fan or any light. Apart from the intelligent systems, the cloud infrastructure with the help of various protocols such as MQTT, CoaP, all the data was transferred with the utmost privacy and security.

With the help of various sensors, the Internet of things has overcome boundaries very easily giving rise to technology no one could ever even think of. Starting from an ATM machine to fully developed home-automation systems.

The most major sensor is the distance sensor or the motion sensor. It detects all the activity that is happening in the environment. This particular sensor has helped to develop smart systems.

For example, a system that assists in car parking, when the car is about to touch the wall, the led turns on to warn about the obstacle, which is mainly used in almost all the cars that are now being developed.

The second sensor is the temperature sensor, it has been used in smart agriculture systems. If we talk about the water preserving application, the temperature sensor has a major role in it, as it is able to identify the moisture of the soil and accordingly, its water was automatically provided.

Apart from the other sensors, if we particularly take about a sensor which has been used at major innovative ideas, it's the motion sensor, recent technology includes the smart cars that are built by Tesla.

Working:

→Transmitter - sensors and timers, receiver - load control.

→Lighting control for underground parking or a room area.

→When driving, the illumination of the motion sensor zone, as well as the front and rear zones, is switched on.

→When the motion sensor signal is reset, the illumination of the front and rear zones is switched off, and the sensor zone is turned off by a timer.

→LED indicators for PIR sensor status and output signals for lamps.

Step by step procedure:-

→Include the library code:

#include <Wire.h>

Then define the library for controlling single-wire-based LED pixels and strip.
#include <Adafruit_NeoPixel.h>

→Then add the transmit-receive variables
byte addressSlaveDevice = 1;
bool i2cReceiver = 0;

```
int  ledReceiver = 0;
int  markerDataI2C = 0;
```

After that initialize the library with the numbers of the interface pins-
This is used to define the sensors and the arduino connecting pin-modes, that will make the connection between them possible.

→Then we will define the LED strip pins.

(i) For this an arduino pin assignment for LED strips needs to be declared, which will decide the primary connection of the pin.
(ii) Following with the number of dots needed in the LED strip

→Then Create an object -
This object will for work with the LED strip

```
Adafruit_NeoPixel pixels = Adafruit_NeoPixel(NUMBERPIXELS, IN_pin, NEO_GRB+NEO_KHZ800);
```

```
//output pins lamp
const int lampPin0= Num;
const int lampPin1= Num;
const int lampPin2= Num;
const int lampPin3= Num;
const int lampPin4= Num;
```

```
const int lampPin5= Num;
const int lampPin6= Num;
const int lampPin7= Num;
int
lampPin[]={lampPin0,lampPin1,lampPin2,lampPin3,lampPin4,lampPin5,lampPin6,la
mpPin7};

// setting position LED
// Designation of the position of the LED strip to display the state of the PIR sensors.
#define ledShow_0         0    // status - led_0
#define ledShow_1         8    // status - led_1
#define ledShow_2         15   // status - led_2
#define ledShow_3         22   // status - led_3
#define ledShow_4         29   // status - led_4
#define ledShow_5         36   // status - led_5
#define ledShow_6         43   // status - led_6
#define ledShow_7         50   // status - led_7
#define ledShow_8         57   // status - led_8

#define ledNumber   9

int ledShow[ledNumber] = {ledShow_0, ledShow_1, ledShow_2,
                    ledShow_3, ledShow_4, ledShow_5,
                    ledShow_6, ledShow_7, ledShow_8} ;

//-----------------------------------------------------
#define ledDelta     1
```

```
#define ledLampMax       7
//-------------------------------------------------------
//variables for the color palette of the LED strip
int red = 0;
int green = 0;
int blue = 0;
int ledLight = 0;

//-------------------------------------------------------
//function to select the color palette of the LED strip
void setColorPalette(int idPalette){
  switch (idPalette) {
    case 0: red = 254; green = 254; blue = 254;  break;   //white

    case 1: red = 255; green = 0;   blue = 0;    break;   //red

    case 2: red = 0;   green = 255; blue = 0;    break;   //green

    case 3: red = 0;   green = 0;   blue = 255;  break;   //blue

    case 4: red = 250; green = 127; blue = 0;    break;   //orange

    case 5: red = 255; green = 255; blue = 0;    break;   //yellow

    case 6: red = 84;  green = 104; blue = 200;  break;   //indigo

    case 7: red = 140; green = 0;   blue = 255;  break;   //violet
```

```
    case 8: red = 0;   green = 0;   blue = 0;    break;   //
  }
}

void setup()
{
  Serial.begin(19200);
  Serial.println("Start code");

  Wire.begin(addressSlaveDevice);    // join i2c bus with address #1
  Wire.onReceive(receiveEvent);

 //OUTPUT parameter pin
  for (int i=0; i<8; i++)  {
    PRIVATE
    digitalWrite(lampPin[i], HIGH);
  }
```

→Initialization of the LED strip to show the operation of the LED strip in the alarm test mode.

```
 pixels.begin();
  combination_Right(1,ledShow_0,ledShow_0+8);
```

```
  for (int j=0; j<ledNumber; j++)  {
    combination_Right(1,ledShow[j],ledShow[j] + ledLampMax);

//
combination_Right(1,ledShow[j]+i,ledShow[j]+i+1);//ledLampMax
    //delay(20);

 }

  delay(5);

    // initialize the LED strip and show the operation of the LED strip in normal operation.
  combination_Right(8,ledShow_0,ledShow_0+8);
  combination_Right(8,ledShow_1,ledShow_8+7);

   delay(5);
}


void loop()  {
static bool PIRLed[8]={0,0,0,0,0,0,0,0};
static int colorLed = 0;

// data display
```
PRIVATE

```
    i2cReceiver = 0;

    for (int i=0; i<8; i++)  {
    PRIVATE

      //switch On/Off Lamps

      digitalWrite(lampPin[i],!outState);

      //switch On/Off LED strip

      if (outState != PIRLed[i])  {

        PIRLed[i] = outState;

        PIRLed[i] == 1 ? colorLed = 1 : colorLed = 8;

        PRIVATE

        PIRLed[i] == 1 ? colorLed = i : colorLed = 8;

        combination_Right(colorLed,ledShow[i+1],ledShow[i+1]+ ledLampMax); //
      }
    }
  }
}
```

```
void receiveEvent(int howMany)  {

  byte receiverByte[2] = {0,0};

  //receive data
  for (byte i=0; i<2; i++)  {
    PRIVATE
  }

  //data processing
    ledReceiver = (receiverByte[1]<<8) | receiverByte[0];

  //data display
  i2cReceiver = 1;
}
```

Private

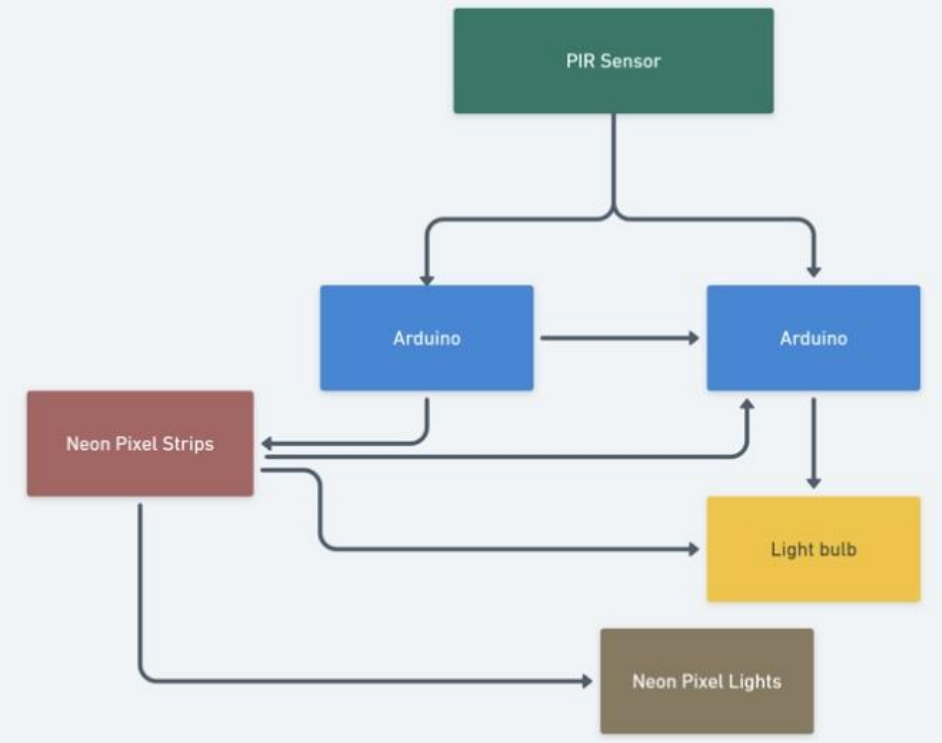
```
  }
  pixels.show();
}
```

Conclusion:

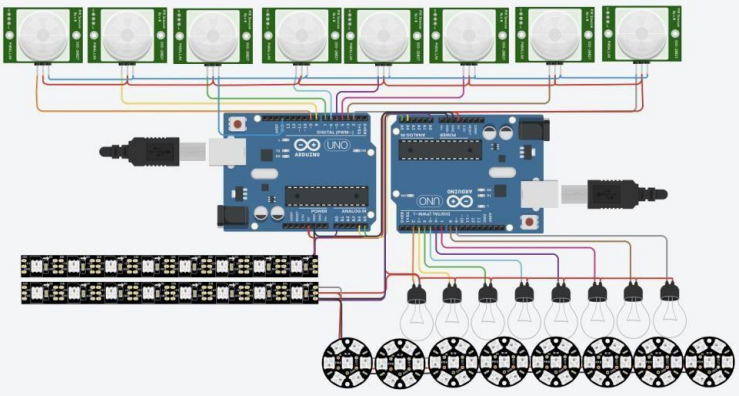
In this project, we will use this motion sensor only or the PIR sensor, which would help us to detect the motion of the incoming object. After analysing the incoming object, the system would illuminate the zone and do the necessary task that needs to be done. This will help to make the most of the resource, as light will not be turned on when it is not required.

The presence of the vehicle and the person will decide if the area needs to be illuminated or not. If we consider that the person has parked the car and left, the illumination then also can be handled efficiently. And accordingly, the light will be turned on/off after a given particular time, which can be modified according to the wish of the owner or user. The area in which the object is present, with its rear zone as well as front zone, will be illuminated efficiently. This will do the job and at the same time help us to preserve the electrical resource which is the baseline of almost every technology And this will result in preserving the electrical energy. And again, Iot helping us achieve wonders.

Data Flow Diagram:



Circuit Diagram:



References :

1. M. Works Ecuador, "82 years building roadworks in Ecuador," 2011.
   C. Rodrigues, P. Almeida, G. Soares, J. Jorge, D.
2. Pinto, and H. Braga, “An experimental comparison between different technologies arising for public lighting: LED luminaires replacing high pressure sodium lamps,” in Industrial Electronics (ISIE), 2011 IEEE International Symposium on, pp. 141– 146, 2011. R. Barraza Garcia, G. Velazquez Angulo, J. Romero
3. Gonzalez, E. Flores Tavizon, and J. Huertas Cardozo, “LED street lighting as a strategy for climate change mitigation at local government level,” in Global Humanitarian Technology
4. Conference (GHTC), 2014 IEEE, pp. 345–349, 2014
5. G. Horvat, D. Sostaric, and D. Zagar, “Using radio irregularity for vehicle detection in adaptive roadway lighting,” in MIPRO, 2012 Proceedings of the 35th International Convention, pp. 748–753, 2012.
6. M. C. Rubio, "Roads that shine with their own light," Roads that shine with their own light, 2014
7. Elecfreaks, “Ultrasonic ranging module hc-sr04
8. Instructables → “Circuits, NeoPixel, Arduino,PIR Sensor.
9. G. Horvat, D. Sostaric, and D. Zagar, “Using radio irregularity for vehicle detection in adaptive roadway lighting,” in MIPRO, 2012 Proceedings of the 35th International Convention, pp. 748–753, 2012.
10. Ravi Kishore Kodali, Research gate, research paper on NeoPixel.