**By : RISHABH SINGHAL**

**BTECH (YEAR 1)**

**ELECTRICAL**

**ROLL NO. 1700737**

***SMART STREET LIGHT SYSTEM USING IR SENSOR***

***BRIEF INTRODUCTION TO IDEA OF THIS PROJET***

Smart Street light is an automated system which automates the street. The main aim of Smart Street light is to reduce the power consumption when there are no vehicle movements on the road. The Smart street light will turn to be ON when there are vehicles on the road otherwise the lights will be switched OFF. With advancement of technology, things are becoming simpler and easier for everyone in the world today. Automation is the use of control systems and information technologies to reduce the need for human work in the production of goods and services. In the scope of industrialization, automation is a step beyond mechanization, whereas mechanization provided human operators with machinery to assist the users with the muscular requirements of work, automation greatly decreases the need for human sensory and mental requirements as well. Automation plays an increasingly important role in the world economy and in daily experience. Automatic systems are being preferred over manual system. The Smart street light provides a solution for energy saving which is achieved by sensing an approaching vehicle using the IR sensors and then switching ON a block of street lights ahead of the vehicle. As the vehicle passes by, the trailing lights switch OFF automatically.

***MATERIAL REQUIRED***

***( For making the main circuit )***

* 1 Arduino uno board
* 4 IR sensors
* 6 LED lights
* 1 Breadboard
* Male to male jumper wires
* Male to female jumper wires
* Female to female jumper wires

***What is Arduino uno?***

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, 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. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter. "Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduno, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform.

Microcontroller ATmega328

Operating Voltage 5V

Input Voltage (recommended) 7-12V

Input Voltage (limits) 6-20V

Digital I/O Pins 14 (of which 6 provide PWM output)

Analog Input Pins 6

DC Current per I/O Pin 40 mA

DC Current for 3.3V Pin 50 mA

Flash Memory 32 KB of which 0.5 KB used by bootloader

SRAM 2 KB

EEPROM 1 KB

Clock Speed 16 MHz



The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector. The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts. The power pins are as follows:

• VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

• 5V. The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.

• 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

• GND. Ground pins.

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

• Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. TThese pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .

• External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.

• PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.

• SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.

• LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. Additionally, some pins have specialized functionality:

• I 2C: 4 (SDA) and 5 (SCL). Support I2C (TWI) communication using the Wire library. There are a couple of other pins on the board:

• AREF. Reference voltage for the analog inputs. Used with analogReference().

• Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields

***Introduction to project***

Automation plays an increasingly important role in the world economy and in daily life. Automatic systems are being preferred over manual system. The research work shows automatic control of streetlights as a result of which power is saved to some extent. In the scope of industrialization, automation is a step beyond mechanization [1]. Whereas mechanization provided human operators with machinery to assist the users with muscular requirements of work, automation greatly decreases the need for human sensory and mental requirements as well. Basically, street lighting is one of the important parts. Therefore, the street lamps are relatively simple but with the development of urbanization, the number of streets increases rapidly with high traffic density. There are several factors need to be considered in order to design a good street lighting system such as night-time safety for community members and road users, provide public lighting at cost effective, the reduction of crime and minimizing it is effect on the environment. At the beginning, street lamps were controlled by manual control where a control switch is set in each of the street lamps which is called the first generation of the original street light. After that, another method that has been used was optical control method done using high pressure sodium lamp in their system. Nowadays, it is seen that the method is widely used in the country. The method operates by set up an optical control circuit, change the resistance by using of light sensitive device to control street lamps light up automatically at dusk and turn off automatically after dawn in the morning. Due to the technological development nowadays, road lighting can be categorized according to the installation area and performance, for an example, lighting for traffic routes, lighting for subsidiary roads and lighting for urban center and public amenity areas. The WSN helps in improving the network sensing for street lighting. Meanwhile, street light system can be classified according to the type of lamps used such as incandescent light, mercury vapor light, metal halide light, high pressure sodium light, low pressure sodium light, fluorescent light, compact fluorescent light, induction light and LED light. Different type of light technology used in lighting design with their luminous efficiency, lamp service life and their considerations. The LED is considered a promising solution to modern street lighting system due to its behavior and advantages. A part from that, the advantages of LED are likely to replace the traditional street lamps such as the incandescent lamp, fluorescent lamp and High Pressure Sodium Lamp in future but LED technology is an extremely difficult process that requires a combination of advanced production lines, top quality materials and high-precision manufacturing process.

The Smart street light control system adopts a dynamic control methodology. According to the proposed plan, initially when it becomes dark, all the street lights automatically glow for a few seconds and switches off. But throughout the night, only one streetlights remains switched on for security concerns. When a vehicle passes by, a block of street lights glows and as the vehicle moves forward, the next block of lights starts glowing where the previous block switches off.

***Existing System***

Industry of street lighting systems are growing rapidly and going to complex with rapid growth of industry and cities. Automation, Power consumption and Cost Effectiveness are the important considerations in the present field of electronics and electrical related technologies. To control and maintain complex street lighting system more economically, various street light control systems are developed. These systems are developed to control and reduce energy consumption of a town's public lighting system using different technologies. The existing work is done using HID lamps. Currently, the HID is used for urban street light based on principle of gas discharge, thus the intensity is not controlled by any voltage reduction method as the discharge path is broken. HID lamps[10] are a type of electrical gas discharge lamp which produces light by means of an electric arc between tungsten electrodes housed inside a translucent or transparent fused quartz or fused alumina arc tube. This tube is filled with both gas and metal salts. The gas facilitates the arc's initial strike. Once the arc is started, it heats and evaporates the metal salts forming plasma, which greatly increases the intensity of light produced by the arc and reduces its power consumption. High-intensity discharge lamps are a type of arc lamp.

***Disadvantages of Existing System:***

• HID lamps consume more power.

• The life time of the HID lamps is very less.

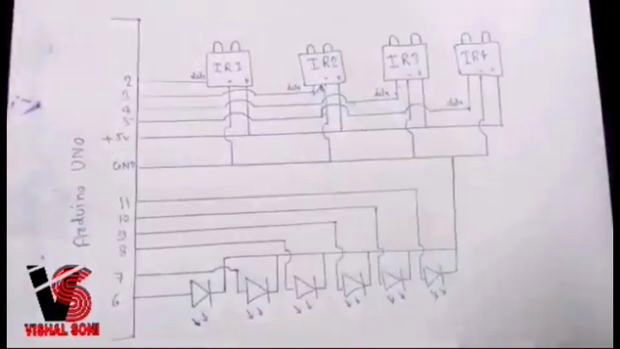
• It cannot be used in all outdoor applications.

• Brightness of the lights in the rear view mirrors which causes a problem for drivers in front of your vehicle.

***Proposed System***

Since the HID lamps are not cost effective and not reliable, smart street light system has overcome by replacing the HID lamps with LED. Due to automation, power consumption and cost effectiveness in the present field of electronics and electrical related technologies, industry of street lighting systems are growing rapidly and going to complex with rapid growth of industry and cities. To control and maintain complex street lighting system more economically, various street light control systems are developed. These systems are developed to control and reduce energy consumption of a town's public lighting system using different technologies which uses IR motion sensors to detect the vehicle movement after which the street light begins to glow. As the vehicle moves, the street light that was glowing switches off and the following lights begins to glow.

***Circuit diagram***



***Program uploaded on Arduino***

int ir1=2;

int ir2=3;

int ir3=4;

int ir4=5;

int led1=6;

int led2=7;

int led3=8;

int led4=9;

int led5=10;

int led6=11;

int red=12;

int green=13;

int prox1=0;

int prox2=0;

int prox3=0;

int prox4=0;

void setup()

{

pinMode(ir1,INPUT);

pinMode(ir2,INPUT);

pinMode(ir3,INPUT);

pinMode(ir4,INPUT);

pinMode(led1,OUTPUT);

pinMode(led2,OUTPUT);

pinMode(led3,OUTPUT);

pinMode(led4,OUTPUT);

pinMode(led5,OUTPUT);

pinMode(led6,OUTPUT);

pinMode(red,OUTPUT);

pinMode(green,OUTPUT);

}

void loop()

{

//digitalWrite(led6,LOW);

prox1=digitalRead(ir1);

prox2=digitalRead(ir2);

prox3=digitalRead(ir3);

prox4=digitalRead(ir4);

if(prox1==HIGH)

{

digitalWrite(led1,HIGH);

digitalWrite(led2,HIGH);

digitalWrite(led3,HIGH);

digitalWrite(green,LOW);

}

else

{

digitalWrite(led1,LOW);

digitalWrite(led2,LOW);

digitalWrite(led3,LOW);

}

if(prox2==HIGH)

{

digitalWrite(led2,HIGH);

digitalWrite(led3,HIGH);

digitalWrite(led4,HIGH);

}

else

{

digitalWrite(led2,LOW);

digitalWrite(led3,LOW);

digitalWrite(led4,LOW);

}

if(prox3==HIGH)

{

digitalWrite(led3,HIGH);

digitalWrite(led4,HIGH);

digitalWrite(led5,HIGH);

}

else

{

digitalWrite(led3,LOW);

digitalWrite(led4,LOW);

digitalWrite(led5,LOW);

}

if(prox4==HIGH)

{

digitalWrite(led4,HIGH);

digitalWrite(led5,HIGH);

digitalWrite(led6,HIGH);

digitalWrite(green,HIGH);

}

else

{

digitalWrite(led4,LOW);

digitalWrite(led5,LOW);

digitalWrite(led6,LOW);

}

/\* if(prox1==HIGH && prox2==HIGH && prox3==HIGH $$ prox4==HIGH)

\* {

\* digitalWrite(red,HIGH);

\* digitalWrite(green,LOW);

\* }

\* else

\* {

\* digitalWrite(red,LOW);

\* digitalWrite(green,HIGH);

\*/

}