ABSTRACT

In an era of smart living, people always try to improve existing technologies to achieve the concept of smart city. Street lighting systems is one of them. Most urban, towns and semi urban cities still use conventional lighting systems. But the use of these light system leads to huge demand and wastage of electricity between 20 to 40% of the total energy produced. Although a single unit of lighting consumes less power, their enormous quantity and long hours of operation requires more energy. So, energy saving is an essential need for smart lighting systems. Energy efficiency is achieved by this smart lighting system. It will provide light that varies intensity as per illumination requirement.The working principle is based on the requirement of luminous energy at a particular moment of time. An automatic system is designed using ARDUINO which will switch ON or OFF the street lights at given time and also depending on the intensity of the ambient light. This system also detects the movement of vehicles and interrupts the system to increase the intensity of light on the road.

Table of Contents

.

List of figures

CHAPTER 1

INTRODUCTION

1.1 Embedded Systems

Embedded system is a scaled down computer system which is designed to perform a specific task/operation. Unlike a general purpose computer system which is used for a variety of tasks, like playing music, games, surfing internet etc. The term embedded tells that whole system is embedded into an appliance.

A single chip contains both hardware and software (technically, firmware). It is designed to perform operations which minimize (or even completely avoid) need of human control. Basic Flow diagram for embedded computer systems can be shown as

Various ways we can Define……

• An Embedded system is a special-purpose computer system designed to perform one or a few dedicated functions, often with real-time computing constraints.

Or

• An Embedded system is a Software program on a H/W chip designed for a specific purpose and can also contain some mechanical moving parts. Ù As features get on added daily definition itself is evolving.

Or

An Embedded system is a special-purpose computer system designed to perform one or a few dedicated functions, often with real-time computing.

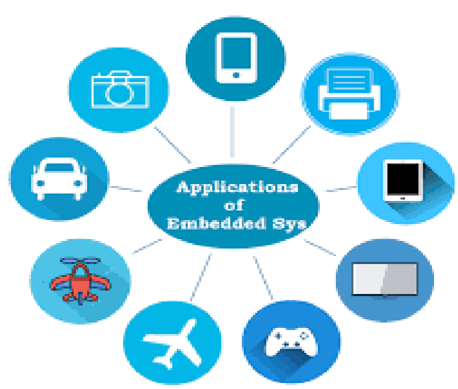

Fig 1: Embedded system application

Some important things to note about embedded systems:

1. Once an embedded hardware is programmed for a certain task, it is used forever for the same task. Changing the firmware afterwards is not possible.

2. Such systems are limited in computational resources like memory, CPU processing speed, I/O facilities but are still capable of performing the task given to them very efficiently.

3. Embedded systems can also be having a reduced functionality version of operating system called RTOS (Real Time Operating System) for highly specialized applications.

4. Is a system built to perform its duty, completely or partially independent of human intervention?

5. Is specially designed to perform a few tasks in the most efficient way.

6. Interacts with physical elements in our environment, viz. controlling and driving a motor, sensing temperature, etc.

Applications

1. Pen drives (for controlling the communication between P.C. and Flash Chip and also the small LED!)

2. Hard disks(again for the same purpose)

3. Mouse(Reads and Interprets the Sensors and send final result to P.C.),Keyboards

4. Printers: Ever opened a printer for installing ink cartridge? Then you must have seen the printed head. There are motors to control the print head and the paper movement. Your P.C. is not directly connected to them but there is built in MCU of printer to control all these. Your P.C. just sends the data (pixels) through the communication line (USB or parallel).But the MCU used here is fairly fast and has lots of RAM.

5. Automobiles

6. Calculators, Electronic wending machines, Electronic weighing scales, Phones(digital with LCD and phonebook)

7. Cell phones

8. Security System

9. Alarm system

10. Automobile system

1.2 Robotics

Robotics is the science of designing and building robots suitable for real-life applications in automated manufacturing and other non-manufacturing environments. Robots are the means of performing multifarious activities for mans welfare in most planned and integrated manner, maintaining their own flexibility to do any work, effecting enhanced productivity, guaranteeing quality, assuring reliability and assuring safety to workers. The word robot came from the Czechoslovakin word Robota which means a worker or a slave doing heavy work

Over the course of human history the emergence of certain new technologies have globally transformed life as we know it. Disruptive technologies like fire, the printing press, oil and television have dramatically changed both the planet we live on and mankind itself, most often in extraordinary and unpredictable ways. In pre-history these disruptions took place in less than a generation.

We are currently at the edge of one such event .In ten years robotic systems will fly our planes, grow our food, explore space, discover life saving drugs, fight our wars, sweep our homes and deliver our babies.

In the process, this robotics driven disruptive event will create a new 200 billion dollar global industry and change life as you now know it, forever. Just as the present generation cannot imagine world without electricity, the future generation will never know a world without robots.

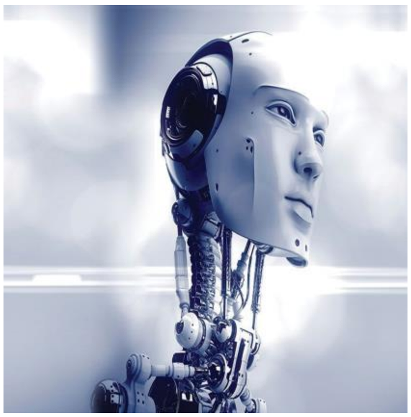

Fig2: Robot

1.3 AIM AND OBJECTIVE OF THE PROJECT

This project aims at designing and executing the advanced development in embedded systems for energy saving of street lights with light depending resistor. Nowadays, human has become too busy and he is unable to find time even to switch the lights wherever not necessary. This can be seen more effectively in the case of street lights. The present system is like, the street lights will be switched on in the evening before the sun sets and they are switched off the next day morning after there is sufficient light on the roads. But the actual timings for these street lights to be switched on are when there is absolute darkness. With this, the power will be wasted up to some extent. This project gives the best solution for electrical power wastage.

Also the manual operation of the lighting system is completely eliminated. In our project we are using LDR, which varies according to the amount of light falling on its surface, this give an indication for us whether it is a day/night time. We have placed IR sensors in both sides of the road, which can be controlled by Micro controller (AT89C51).The IR‘s will be activated only on the night time. If any obstacle crosses the IR, automatically particular light will be ON, for few seconds. In our project we use regulated 5V, 500mA power supply. 7805 three terminal voltage regulator is used for voltage regulation. Bridge type full wave rectifier is used to rectify the ac output of secondary of 230/12V step down transformer. The programming language used for developing the software to the microcontroller is Embedded/Assembly. The KEIL cross compiler is used to edit, compile and debug this program. Here in our application we are using AT89C51 microcontroller which is Flash Programmable IC.AT represents the Atmel Corporation ‘C’ represents CMOS technology is used for designing the IC. We believe that our idea provides better than the existing system.

Objective of the project:-

The main considerations in the present field technologies are Automation, Power consumption and cost effectiveness. Automation is intended to reduce man power with the help of intelligent systems. Power saving is the main consideration forever as the source of the power(Thermal, Hydro etc.,)are getting diminished due to various reasons. The main aim of the project is Automatic street power saving system with LDR, this is to save the power. We want to save power automatically instead of doing manual. So it’s easy to make cost effectiveness. This saved power can be used in some other cases. So in villages, towns etc we can design intelligent systems for the usage of street lights.

1.3.a) Specifications of the project

1. The smsl system encompasses a combinational infrastructure for

Communications

2. In proposed system ,we can control the intensity of the light .

3. The pole controller turns on/off lamps.

4. A Passive Infrared Detector is installed on the closest pole to the segment controller to detect vehicle and pedestrian motion.

5. The segment controller is the local interface between the lighting pole controllers in the field and the supervisory system.

6. The Local controller acts as an actuator and applies the received commands.

7. The controller inherently responses to lamp fault. Moreover, pole electrical parameters and status of the lamp and its capacitor is transmitted to the intermediate controller.

8. The supervisory controller which is installed on a server monitors the whole system and sends appropriate commands to the segment controller based on WiMAX.

1.3.b) IMPLEMENTATION OF THE PROJECT

PROPOSED BLOCK DIAGRAM MODEL

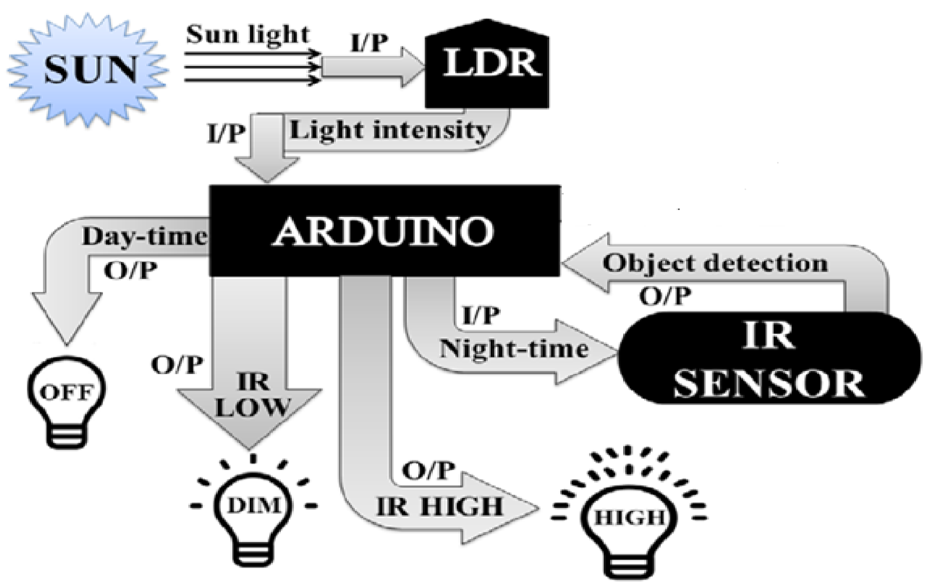

Fig 3 : Block diagram of the project

1.3.c) Working

For the simplicity of discussion, Figure3 illustrates the overall working mechanism and the features of the proposed lighting concept. I/P and O/P represents the input and output, respectively. LDR first senses the intensity value of sunlight and sends it to Arduino. After receiving the data, Arduino converts it into different discrete values from 0–1023, and judges whether the received value is above the threshold level (a limit value that is set independently by the user from the range of discrete values: 0–1023); it will then be considered as day-time, and the LEDs will remain OFF; if the received value is below the threshold level, Arduino will consider it as night-time. During night-time, if the value of IR obstacle avoidance sensor is LOW and detects no object, then DIM LEDs will glow, or if the IR obstacle avoidance value is HIGH and identifies any object, then HIGH LEDs will shine.

1.3.d) FLOW CHART

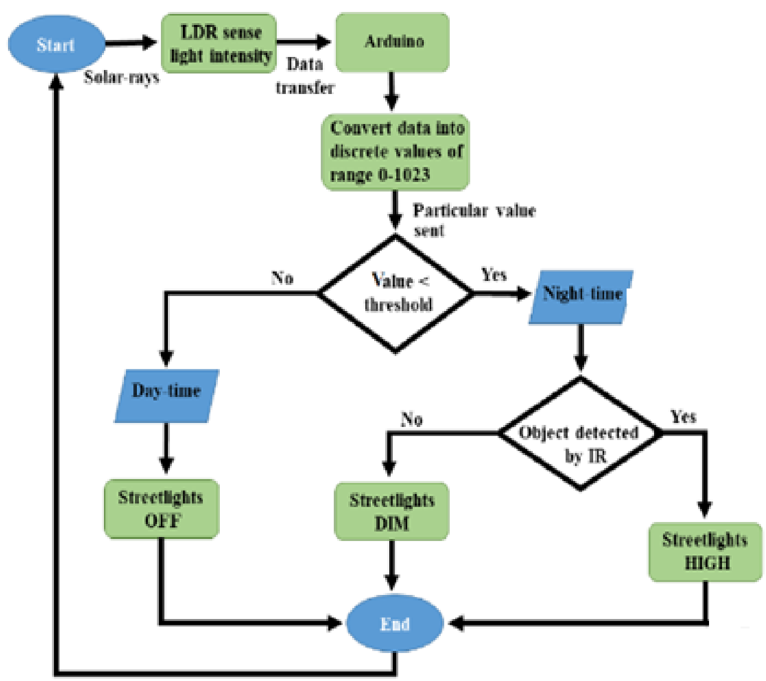

Fig 4 : Flow chart of the project

1.3.e) CIRUIT DIAGRAM

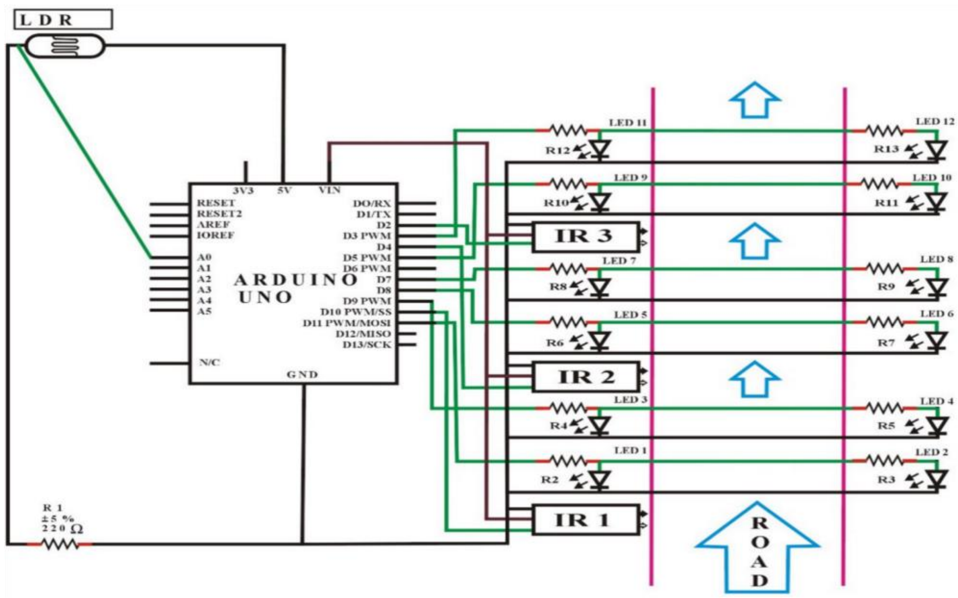

Fig 5: circuit diagram of the project

1.3.f) Connections

Figure 5 shows the circuit design of an automatic street light control system based on object detection using Arduino Uno, with DIM light capability. In this scenario, the lights will turn to HIGH only with the detection of an object; otherwise the lights will remain OFF at day-time, and DIM at night-time.

In this task, an LDR sensor, 12 LEDs, 13 resistors, three IR obstacle avoidance sensors, and a single Arduino Uno were used. One leg of the LDR sensor was connected to Arduino analog pin number A0, and another leg to a 5 V pin, and the same with a resistor to the GND port of the Arduino. Besides, the threshold value for the LDR was adjusted to 200 from the discrete values (0–1023) for understanding whether it is day or night. After that, all the positive terminals of the LED set were connected with resistors to pin numbers D11, D9, D8, D7, D5, and D3 as the outputs of the Arduino signals. Here, one set of LEDs consists of two individual LEDs. Furthermore, the GNDs of all the LEDs were connected to the GND port, as shown in the circuit diagram

CHAPTER 2

HARDWARE SPECIFICATION

2.1 MICROCONTROLLER

A Brief History of Arduino Uno

Arduino/Genuino Uno is a micro controller board based on the ATmega328P (datasheet). 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 micro controller; 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 worring 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 boards..

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.

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.This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

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

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

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. These 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 using the SPI library.

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, labeled A0 through A5, 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:

TWI: A4 or SDA pin and A5 or SCL pin. Support 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 which block the one on the board. See also the mapping between Arduino pins and ATmega328 ports. The mapping for the Atmega8, 168, and 328 is identical.

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other micro controllers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital

pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software Serial library allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

The Arduino Uno can be programmed with the Arduino software (download). Select "Arduino Uno from the Tools > Board menu (according to the microcontroller on your board). For details, see the reference and tutorials. The ATmega328 on the Arduino Uno comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files). You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details. The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available . The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by: On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode. You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See this user-contributed tutorial for more information.

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip.

The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following halfsecond or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened.

If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data. The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

The Arduino Uno has a resettable polyfuse that protects your computer's

USB ports from shorts and over current. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

Fig 6 : Arduino Uno Circuit Board

2.2 INFRARED SENSOR

An infrared sensor is an electronic device, that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion.These types of sensors measures only infrared radiation, rather than emitting it that is called as a passive IR sensor. Usually in the infrared spectrum, all the objects radiate some form of thermal radiations. These types of radiations are invisible to our eyes, that can be detected by an infrared sensor .The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode which is sensitive to IR light of the same wavelength as that emitted by the IR LED. When IR light falls on the photodiode, The resistances and these output voltages, change in proportion to the magnitude of the IR light received.

2.2.a Introduction

Infrared technology addresses a wide variety of wireless applications. The main areas are sensing and remote controls. In the electromagnetic spectrum, the infrared portion is divided into three regions: near infrared region, mid infrared region and far infrared region.

The wavelengths of these regions and their applications are shown below.

1. Near infrared region — 700 nm to 1400 nm — IR sensors, fiber optic
2. Mid infrared region — 1400 nm to 3000 nm — Heat sensing
3. Far infrared region — 3000 nm to 1 mm — Thermal imaging

The frequency range of infrared is higher than microwave and lesser than visible light.

For optical sensing and optical communication, photo optics technologies are used in the near infrared region as the light is less complex than RF when implemented as a source of signal. Optical wireless communication is done with IR data transmission for short range applications.

An infrared sensor emits and/or detects infrared radiation to sense its surroundings.

The working of any Infrared sensor is governed by three laws: Planck's Radiation law, Stephen – Boltzmann law and Wien's Displacement law.

Planck's law states that "every object emits radiation at a temperature not equal to 00K". Stephen – Boltzmann law states that "at all wavelengths, the total energy emitted by a black body is proportional to the fourth power of the absolute temperature". According to Wien's Displacement law, "the radiation curve of a black body for different temperatures will reach its peak at a wavelength inversely proportional to the temperature".

The basic concept of an Infrared Sensor which is used as Obstacle detector is to transmit an infrared signal, this infrared signal bounces from the surface of an object and the signal is received at the infrared receiver.

There are five basic elements used in a typical infrared detection system:

infrared source, a transmission medium, optical component, infrared detectors or receivers and signal processing. Infrared lasers and Infrared LED's of specific wavelength can be used as infrared sources. The three main types of media used for infrared transmission are vacuum, atmosphere and optical fibers. Optical components are used to focus the infrared radiation or to limit the spectral

Optical lenses made of Quartz, Germanium and Silicon are used to focus the infrared radiation. Infrared receivers can be photodiodes, phototransistors etc. some important specifications of infrared receivers are photosensitivity, detectivity and noise equivalent power. Signal processing is done by amplifiers as the output of infrared detector is very small.

2.2.b Types of IR sensor

Infrared sensors can be passive or active. Passive infrared sensors are basically Infrared detectors. Passive infrared sensors do not use any infrared source and detects energy emitted by obstacles in the field of view. They are of two types: quantum and thermal. Thermal infrared sensors use infrared energy as the source of heat and are independent of wavelength. Thermocouples, pyroelectric detectors and bolometers are the common types of thermal infrared detectors.

Quantum type infrared detectors offer higher detection performance and are faster than thermal type infrared detectors. The photosensitivity of quantum type detectors is wavelength dependent. Quantum type detectors are further classified into two types: intrinsic and extrinsic types. Intrinsic type quantum detectors are photoconductive cells and photovoltaic cells.

Active infrared sensors consist of two elements: infrared source and infrared detector. Infrared sources include an LED or infrared laser diode. Infrared detectors include photodiodes or phototransistors. The energy emitted by the infrared source is reflected by an object and falls on the infrared detector.

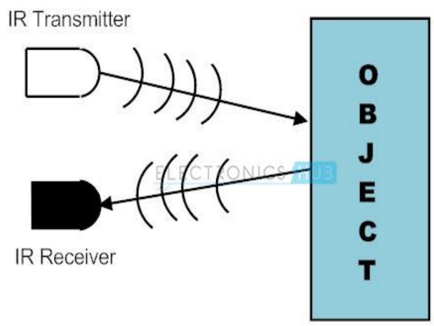

Fig 7 : IR SENSORS

2.2.c IR Transmitter

Infrared Transmitter is a light emitting diode (LED) which emits infrared radiations. Hence, they are called IR LED's. Even though an IR LED looks like a normal LED, the radiation emitted by it is invisible to the human eye.

The picture of a typical Infrared LED is shown below.

Fig 8 : IR TRANSMITTER

There are different types of infrared transmitters depending on their wavelengths, output power and response time.

A simple infrared transmitter can be constructed using an infrared LED, a current limiting resistor and a power supply. The schematic of a typical IR transmitter is shown below.

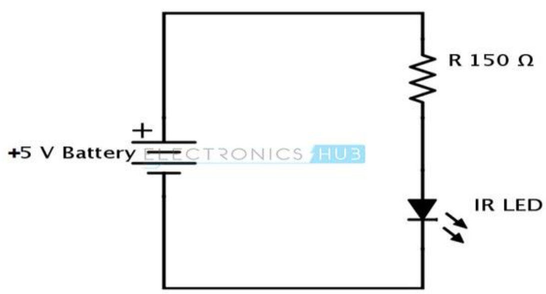

Fig 9: Circuit Diagram of IR Transmitter

When operated at a supply of 5V, the IR transmitter consumes about 3 to 5 mA of current. Infrared transmitters can be modulated to produce a particular frequency of infrared light. The most commonly used modulation is OOK (ON – OFF – KEYING) modulation.

IR transmitters can be found in several applications. Some applications require infrared heat and the best infrared source is infrared transmitter. When infrared emitters are used with Quartz, solar cells can be made.

2.2.d IR Receiver

Infrared receivers are also called as infrared sensors as they detect the radiation from an IR transmitter. IR receivers come in the form of photodiodes and phototransistors. Infrared Photodiodes are different from normal photo diodes as they detect only infrared radiation. The picture of a typical IR receiver or a photodiode is shown below Different types of IR receivers exist based on the wavelength, voltage, package, etc. When used in an infrare

Fig 10: IR RECIVER

transmitter – receiver combination, the wavelength of the receiver should match with that of the transmitter.

A typical infrared receiver circuit using a phototransistor is shown below.

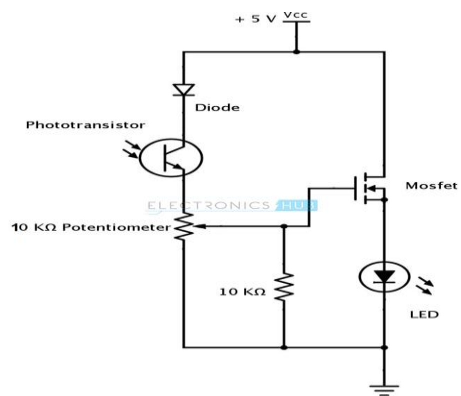

Fig11: Circuit Diagram of IR Receiver

It consists of an IR phototransistor, a diode, a MOSFET, a potentiometer and an LED. When the phototransistor receives any infrared radiation, current flows through it and MOSFET turns on. This in turn lights up the LED which acts as a load. The potentiometer is used to control the sensitivity of the phototransistor.

2.2.e Principle of Working

The principle of an IR sensor working as an Object Detection Sensor can be explained using the following figure. An IR sensor consists of an IR LED and an IR

Photodiode; together they are called as Photo – Coupler or Opto – Coupler. When the IR transmitter emits radiation, it reaches the object and some of the radiation reflects back to the IR receiver. Based on the intensity of the reception by the IR receiver, the output of the sensor is defined.

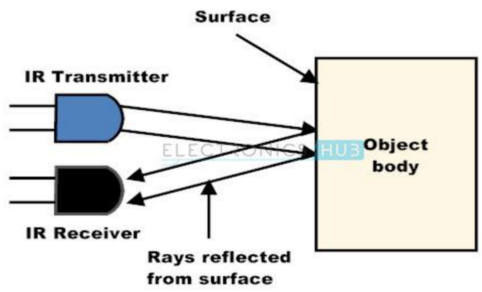

Fig 12 : SENSING DIAGRAM

2.3 LDR SENSOR

A Light Dependent Resistor (LDR) is also called a photoresistor or a cadmium sulfide (CdS) cell. It is also called a photoconductor. It is basically a photocell that works on the principle of photoconductivity. The passive component is basically a resistor whose resistance value decreases when the intensity of light decreases. This optoelectronic device is mostly used in light varying sensor circuit, and light and dark activated switching circuits. Some of its applications include camera light meters, street lights, clock radios, light beam alarms, reflective smoke alarms, and outdoor clocks.

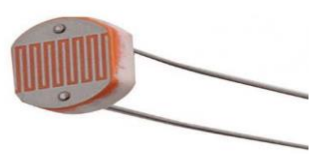

Fig 13 : LDR SENSOR

2.3.a LDR Sensor Working

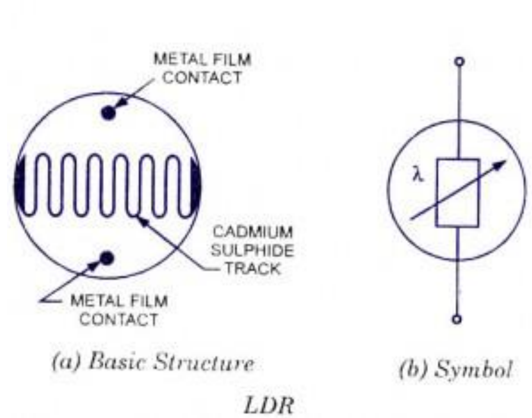

Fig 14: LDR SENSING DIAGRAM

The snake like track shown below is the Cadmium Sulphide (CdS) film which also passes through the sides. On the top and bottom are metal films which are connected to the terminal leads. It is designed in such a way as to provide maximum possible contact area with the two metal films. The structure is housed in a clear plastic or resin case, to provide free access to external light. As explained above, the main component for the construction of LDR is cadmium sulphide (CdS), which is used as the photoconductor and contains no or very few electrons when not illuminated. In the absence of light it is designed to have a high resistance inthe range of megaohms. As soon as light falls on the sensor, the electrons are liberated and the conductivity of the material increases. When the light intensity exceeds a certain frequency, the photons absorbed by the semiconductor give band electrons the energy required to jump into the conduction band. This causes the free electrons or holes to conduct electricity and thus dropping the resistance dramatically (< 1 Kiloohm).

The equation to show the relation between resistance and illumination can be written as

$R = A.E^a$

Where

E–Illumination(lux)

R–Resistance(Ohms)

A,a – constants

The value of ‘a’ depends on the CdS used and on the manufacturing process. Values usually range betwee 0.7 and 0.9.

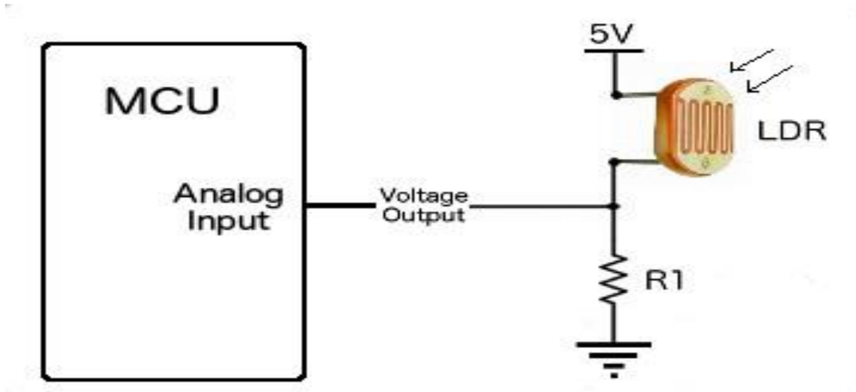

Fig15: LDR Sensor Arduino Circuit

CHAPTER 3

RESULTS AND CONCLUSIONS

3.1 RESULT

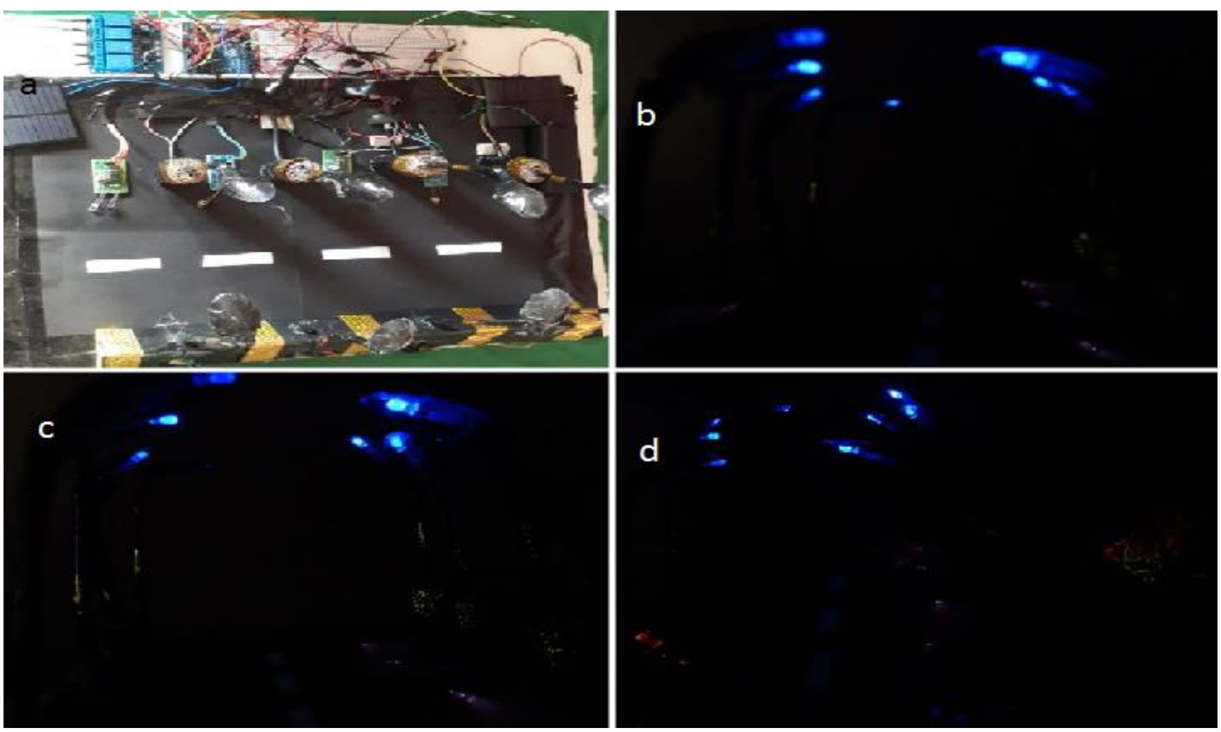

Fig16 : result of the project

Result diagrams of the automatic streetlight control system that switches to DIM at night and HIGH at object detection. (**a**) In the day-time simulation, the LEDs are not glowing. (**b**) In the night-time representation, the DIM LEDs are glowing. (**c**) An object in front of first IR obstacle avoidance sensor; the first set of HIGH LEDs are glowing while the remainder are in DIM mode. (**d**) Motion in front of the second IR obstacle avoidance sensor; only the second set of LEDs are glowing HIGH, and all the others are in a DIM state.

COMPONENT	**WORKING**	**OUTCOME**
LDR	Works based on light intensity.	**Threshold value:** a) <200 – Night time. b) >200 – Day time
PIR(Proximity IR Sensor)	Detects the movement of obstacle.	a) when no obstacle is detected ,light will be in dim condition. b)when obstacle is detected the light will have high intensity after passage of that obstacle it will come back to its original position.

3.2 CONCLUSIONS

This paper discusses the technical aspect of smart street light system and the possible energy saved by implementing this proposed system. The current problem with the conventional system is the long hour operational time which cause a lot of electricity cost. This is a huge waste if it is not taken seriously. Thus, this project proposed the solution to save the energy consumption of street light. Two sensors were used in this proposed smart street light system which is PIR Sensor and LDR sensor. By using PIR sensor to detect

obstacle, it can control the light intensity level which lead to saving energy. Besides, LED bulb used in this paper is also able to control the power consumption use by street light and saves the energy up to 40% to 45% per month. Other than that, it also reduces carbon dioxide (CO_2) emission. In this system we are using solar panels which helps in recharging the battery . All these were achieved in this paper.

3.3 BENEFITS

1. Reduced Maintenance Costs. ...
2. Real-time Reports and Controllability. ...
3. Environmental Impact. ...
4. Light Intensity Control. ...
5. Smart Mapping. ...
6. Life span….
7. Power saving.

3.4 APLICATIONS

Street light management can control the brightness of the streetlights in real time based on environmental dynamics or for special events. This real-time control provides a platform for **smart** city **applications** that can be paired with many public data sources such as traffic, weather, and other IoT devices.

CHAPTER 4

CODE

```
const int ledPin1 = 9;
const int ledPin2 = 10;
const int ledPin3 = 11;
const int ledPin4 = 5;
int obstaclePin1 = 4;// This is our input pin
int obstaclePin2 = 2;
int obstaclePin3 = 6;
int obstaclePin4 = 7;
int hasObstacle1 = LOW;
int hasObstacle2 = LOW;
int hasObstacle3 = LOW;
int hasObstacle4 = LOW; // HIGH MEANS NO OBSTACLE
const int ldrPin = A0;
int RELAY1 = A1;
int RELAY2 = A2;
int RELAY3 = A3;
int RELAY4 = A4;
int fadeAmount = 3;
int brightness = 50;
int delayValue = 500;
void setup() {

Serial.begin(9600);

  pinMode(RELAY1, OUTPUT);
  pinMode(RELAY2, OUTPUT);
  pinMode(RELAY3, OUTPUT);
  pinMode(RELAY4, OUTPUT);
```

```
  digitalWrite(RELAY1, HIGH);
  digitalWrite(RELAY2, HIGH);
  digitalWrite(RELAY3, HIGH);
  digitalWrite(RELAY4, HIGH);
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(ledPin3, OUTPUT);
  pinMode(ledPin4, OUTPUT);
  pinMode(obstaclePin1, INPUT);
  pinMode(obstaclePin2, INPUT);
  pinMode(obstaclePin3, INPUT);
  pinMode(obstaclePin4, INPUT);

  pinMode(ldrPin, INPUT);



}

void loop() {

int ldrStatus = analogRead(ldrPin);

if (ldrStatus >= 200) {

digitalWrite(ledPin1, LOW);
digitalWrite(ledPin2, LOW);
digitalWrite(ledPin3, LOW);
digitalWrite(ledPin4, LOW);
digitalWrite(RELAY1, HIGH);
digitalWrite(RELAY2, HIGH);
```

```
digitalWrite(RELAY3, HIGH);
digitalWrite(RELAY4, HIGH);


} else {



if (hasObstacle1 == HIGH )
{
  digitalWrite(RELAY1, LOW);
  digitalWrite(ledPin1, HIGH);
}
else
{
  digitalWrite(RELAY1, HIGH);
  analogWrite(ledPin1, brightness);
}

if (hasObstacle2 == HIGH )
{
  digitalWrite(RELAY2, LOW);
}
else
{
  digitalWrite(RELAY2, HIGH);
}

hasObstacle1 = digitalRead(obstaclePin1);
  if (hasObstacle1 == HIGH)
```

```
{
  //Serial.println("Stop something is ahead!!");
  digitalWrite(RELAY1, LOW);
  digitalWrite(ledPin1, HIGH);

}
else
{
  //Serial.println("Path is clear");
  digitalWrite(RELAY1, HIGH);
  //digitalWrite(ledPin1, LOW);
  analogWrite(ledPin1, brightness);
}
//delay(200);
hasObstacle2 = digitalRead(obstaclePin2);
if (hasObstacle2 == HIGH)
{
  //Serial.println("Stop something is ahead!!");
  digitalWrite(RELAY2, LOW);
  digitalWrite(ledPin2, HIGH);
}
else
{
  //Serial.println("Path is clear");
  digitalWrite(RELAY2, HIGH);
  analogWrite(ledPin2, brightness);
}
//delay(200);
hasObstacle3 = digitalRead(obstaclePin3);
if (hasObstacle3 == HIGH)
```

```
    {
      //Serial.println("Stop something is ahead!!");
      digitalWrite(RELAY3, LOW);
      digitalWrite(ledPin3, HIGH);
    }
    else
    {
      //Serial.println("Path is clear");
      digitalWrite(RELAY3, HIGH);
      analogWrite(ledPin3, brightness);
    }
    //delay(200);
    hasObstacle4 = digitalRead(obstaclePin4);
    if (hasObstacle4 == HIGH)
    {
      //Serial.println("Stop something is ahead!!");
      digitalWrite(RELAY4, LOW);
      digitalWrite(ledPin4, HIGH);
    }
    else
    {
      //Serial.println("Path is clear");
      digitalWrite(RELAY4, HIGH);
      analogWrite(ledPin4, brightness);
    }
    //delay(200);
  }

}
```

5. FUTURE ENHANCEMENTS

1. In future the project can be modified to recharge battery by using dyanamo under speed breaker.
2. It can also modify the intensity level according to the obstacle.