

SMART TRAFFIC CONTROL SYSTEM BASED ON VEHICLE DENSITY

Project Report submitted in partial fulfillment of the requirements for the Degree of

BACHELORS OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

Shobhit Agarwal : 121057

Shivi Bhatnagar : 121120

Pankaj Kumar : 121121

Under the guidance of

DR. MEENAKSHI SOOD



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

MAY, 2016

TABLE OF CONTENTS

TOPICS	Page Number
DECLARATION BY SCHOLAR.....	i
SUPERVISOR’S CERTIFICATE.....	ii
ACKNOWLEDGEMENT.....	iii
ABSTRACT.....	iv
LIST OF ACRONYMS.....	v
LIST OF FIGURES.....	vi
LIST OF TABLES.....	viii
CHAPTER 1	
INTRODUCTION.....	1
1.1 PRESENT SCENENARIO	2
1.2 MOTIVATION	5
1.3 OBJECTIVE	5
1.4 SCOPE OF THE PROJECT	6
CHAPTER 2	
HARDWARE COMPONENTS.....	7
2.1 ARDUINO UNO	7
2.1.1 ARDUINO PIN DIAGRAM	9
2.1.2 TECHNICAL SPECIFICATION	10
2.1.3 POWER SUPPLY	12

2.1.4	SOFTWARE	13
2.2	PIR SENSOR	14
2.2.1	WORKING	17
2.3	LCD	20
2.3.1	16X2 PIN OUT DIAGRAM	20
2.3.2	INTERFACING WITH ARDUINO UNO	22
2.4	RF RECEIVER TRANSMITTER PAIR	25
CHAPTER 3		
	DESIGN METHODOLOGY.....	27
3.1	PROPOSED METHODOLOGY	27
3.2	WORKING	29
3.2.1	MODULE 1	30
3.2.2	MODULE 2	31
CHAPTER 4		
	RESULTS AND CONCLUSION.....	32
4.1	RESULTS	32
4.2	CONCLUSION	35
4.3	FUTURE SCOPE	36
4.4	CHALLENGES FACED	40
	REFERENCES	41

DECLARATION BY THE SCHOLAR

We hereby declare that the work reported in the B-Tech project entitled “**Smart Traffic Control System Based on Vehicle Density**” submitted at **Jaypee University of Information Technology, Wagnaghat India**, is an authentic record of our work carried out under the supervision of **Dr. Meenakshi Sood**. We have not submitted this work elsewhere for any other degree or diploma.

SHOBHIT AGARWAL (121057): _____

SHIVI BHATNAGAR (121120) : _____

PANKAJ KUMAR (121121) : _____

Department of Electronics and Communication Engineering
Jaypee University of Information Technology, Wagnaghat
MAY, 2016



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY

(Established by H.P. State Legislative vide Act No. 14 of 2002)
P.O. Wagnaghat, Teh. Kandaghat, Distt. Solan - 173234 (H.P.) INDIA

Website: www.juit.ac.in

Phone No. (91) 01792-257999

Fax: +91-01792-245362

SUPERVISOR'S CERTIFICATE

This is to certify that the work reported in the B-Tech. project entitled “**Smart Traffic Control System Based on Vehicle Density**”, submitted by **Shobhit Agarwal – 121057, Shivi Bhatnagar – 121120 and Pankaj Kumar - 121121** at **Jaypee University of Information Technology, Wagnaghat, India**, is a bonafide record of his / her original work carried out under my supervision. This work has not been submitted elsewhere for any other degree or diploma.

Dr. Meenakshi Sood

Assistant Professor (Senior Grade)

Department of Electronics & Communication Engineering

Jaypee University of Information Technology, Wagnaghat, Solan.

विद्या तत्व ज्योतिसम्:

ACKNOWLEDGEMENT

A Project or task cannot be successfully completed without the help of those people who act as a guiding light and provide inspiration when you feel you have reached the spot you cannot seem to get out of. We are extremely grateful to **Dr. Meenakshi Sood**, who has been one such mentor to us. Our respected teacher has been a source of guidance and motivation from time to time. Her thinking and straightforward attitude has inspired us to work harder.

We are highly indebted to her for her initiative, encouragement, constructive criticism and familiarizing us with all technical aspects of this project over and above.

We would like to thank our **Dean, Prof. T.S. Lamba** and our Head of the Department **Prof. Sunil Bhooshan** for providing opportunity and the support required to work on this project.

Shobhit Agarwal : 121057

Shivi Bhatnagar : 121120

Pankaj Kumar : 121121

ABSTRACT

The project is designed to develop a SMART TRAFFIC CONTROL SYSTEM BASED ON VEHICLE DENSITY. The signal timing changes automatically on sensing the traffic density at the junction. There are several models for traffic simulation. In our project we focus on optimization of traffic in a city using IR sensor and visual monitoring using microcontroller ARDUINO UNO. Traffic light optimization is a complex problem. In this project, we propose an approach, to process the input data by Microcontroller and finally allotting a specific time to traffic light signals to control the Closed Loop System. The system contains PIR sensors which are mounted on the either sides of roads respectively. The IR system gets activated whenever any vehicle passes on road. Microcontroller controls the IR system and counts the number of vehicles passing on road. Microcontroller also store vehicles count in its memory. Based on different vehicles count, the microcontroller takes decision and updates the traffic light delays as a result. The system records vehicle count in its memory at user predefined recording interval on real time basis. This recorded vehicle count data can be used in future to analyze traffic condition at respective traffic lights connected to the system. Any central station computer can access traffic conditions of any approachable traffic lights to reduce traffic congestions to an extent. In future this system can be used to inform people about different places traffic condition through GSM or with the concept of Internet of Things.

LIST OF ACRONYMS

ASCII	American Standard Code for Information Interchange
ASK	Amplitude Shift Key
A/D	Analog to Digital
CAD	Computer-Aided Design
CCD	Charged Couple Device
EEPROM	Electrically Erasable Programmable Read Only Memory
GND	Ground
GPL	General Public License
GPS	Global Positioning System
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
IDE	Integrated Development Environment
I/O	Input / Output
IR	Infrared
ITS	Intelligent transportation systems
LCD	Liquid Crystal Display
LED	Light Emitting Diode
MIPS	Millions Instruction per Second
PIR	Passive Infrared
RF	Radio Frequency
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
TFT	Thin Film Transistor
USB	Universal Serial Bus

LIST OF FIGURES

FIGURE NO.	CAPTION	PAGE NO.
1.1	Vehicular Queuing	1
1.2	Smart Traffic Control Model	4
2.1	Arduino Uno	7
2.2	Arduino Pin Diagram	9
2.3	ATmega328	11
2.4	External Power	12
2.5	Pyroelectric Sensor	14
2.6	PIR Sensor Bottom View	15
2.7	PIR Sensor Top View	16
2.8	PIR Working	17
2.9	Fresnel Lens	18
2.10	Converging of rays from Fresnel lens	19
2.11	LCD pin out	20
2.12	Circuit Diagram – Arduino to 16X2 LCD	22
3.1	Block Diagram of Working Model	27

3.2	Decision-making Flowchart	29
4.1	Circuit Implementation	32
4.2a	Implementation of results obtained	33
4.2b	Implementation of results obtained	34
4.3	Lights and sensors running by Solar Panel	36
4.4	Priority Based Traffic Control System	38

LIST OF TABLES

TABLE NO.	CAPTION	PAGE NO.
2.1	Technical Specification	10
4.1	Observation for Different Vehicle Count	33

CHAPTER – 1

INTRODUCTION

Many problems are faced by us in this modern life one of which is traffic congestion and it is becoming more serious day after day. Traffic congestion is a severe problem in many major cities across the world and it has become a nightmare for the commuters in these cities. It is said that the high volume of vehicles, the inadequate infrastructure and the uneven distribution of the development are main reasons for increasing traffic jam. With affordability and increase in purchasing power, it has become very easy for a common person to own a vehicle. The number of cars sold last per year is increasing at a very high rate every year. Though this has led to a comfortable lifestyle, it also leads to problems such as road congestion and traffic pile up around our cities. Traffic congestion is a condition on road networks that occurs as increases in use of vehicles, and is characterized by slower speeds, longer trip times, and increased vehicular queuing as shown in figure 1.1.



Figure 1.1: Vehicular Queuing

1.1 PRESENT SCENARIO

When traffic demand is great enough then the interaction between vehicles slows the speed of the traffic stream which results in congestion. As soon as the demand becomes equals to the capacity of the road, extreme traffic congestion occurs. When vehicles are fully stopped for periods of time, this is colloquially known as a traffic jam or traffic snarl-up. Traffic congestion can lead to drivers' frustration and his/her engagement in road rage in order to avoid the congestion in the traffic.

Nowadays Traffic density on roads in big cities has become so high that it requires a lot of planning for smooth flow of the traffic on the roads. Normally the traffic lights are installed at almost all major traffic crossings in these cities. Traffic lights normally help a lot in managing traffic flow but the pattern of changing the traffic lights is a static repetitive task which does not take in consideration the real time traffic intensity on different ways of the crossing. Moreover during traffic jam, the emergency vehicles such as ambulance, fire brigade will be stuck at the traffic light junction which can prove out to be very critical problem.

Growth in urbanization and traffic congestion is massive in modern cities and it has generated the need of intelligent vision based traffic light controller to reduce chaos, traffic delays and travel time as the current automatic time based control is not realistic.

Conventional traffic light system is based on fixed time concept allotted to each side of the junction which cannot be varied depending upon the varying traffic density. The delay of respective light is hard coded in the traffic light and it does not dependent on traffic. Junction timings allotted are fixed. The intricate reasoning done by traffic lights that are in widespread use today is not efficient while deciding when to change the lights for the various road users waiting in different lanes. Even though today's methods are robust and work well for the evenly distributed traffic load across the lanes in the intersection but the systems are very inefficient because they are unable to handle various situations that arise throughout the day.

Traffic congestion is now considered as one of the biggest problems in urban cities. Traffic problems will increase more as an expected result of growing number of private transportation

means and low-quality infrastructure of the roads. Many studies and statistics were done in developing countries which proved that most of the road accidents occur due to very narrow roads and because of the destructive increase in the transportation means. The idea of creating a dynamic traffic light system in real time has attracted many researchers to work in this field with the goal of creating smart tool that can estimate the traffic congestion and based on this variable, the traffic light time interval is decided.

The heart of any system is the processing unit which controls the working of various sensors and hardware components. In our project we are using the Arduino UNO as the processing unit. The Arduino UNO is a microcontroller board based on the ATmega328. Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. The microcontroller board is programmed using the Arduino programming language under an open-source license; we are free to adapt them according to our needs. Arduino UNO microcontroller boards based on the ATmega328 are then programmed as per our need to perform the desired operations.

The Arduino UNO Microcontroller Development Board was selected as it satisfies our requirements. As was fundamental to the course, the assembly language was utilized for the project. The ARDUINO Programmer is based on the Processing IDE and uses a variation of the C and C++ programming languages .It was more than adequate to satisfy design objectives while enhancing level of understanding of the programming language.

For keeping a pace of ongoing traffic on road we can use PIR. These are mounted on the either sides of roads to act as vehicle detectors. This IR pair will act as an input to take data about ongoing vehicles and a queue estimator. Input which is obtained and optimized is send to Microcontroller which controls the IR system and takes decision based on density of vehicles. This complete model is shown in figure 1.2.

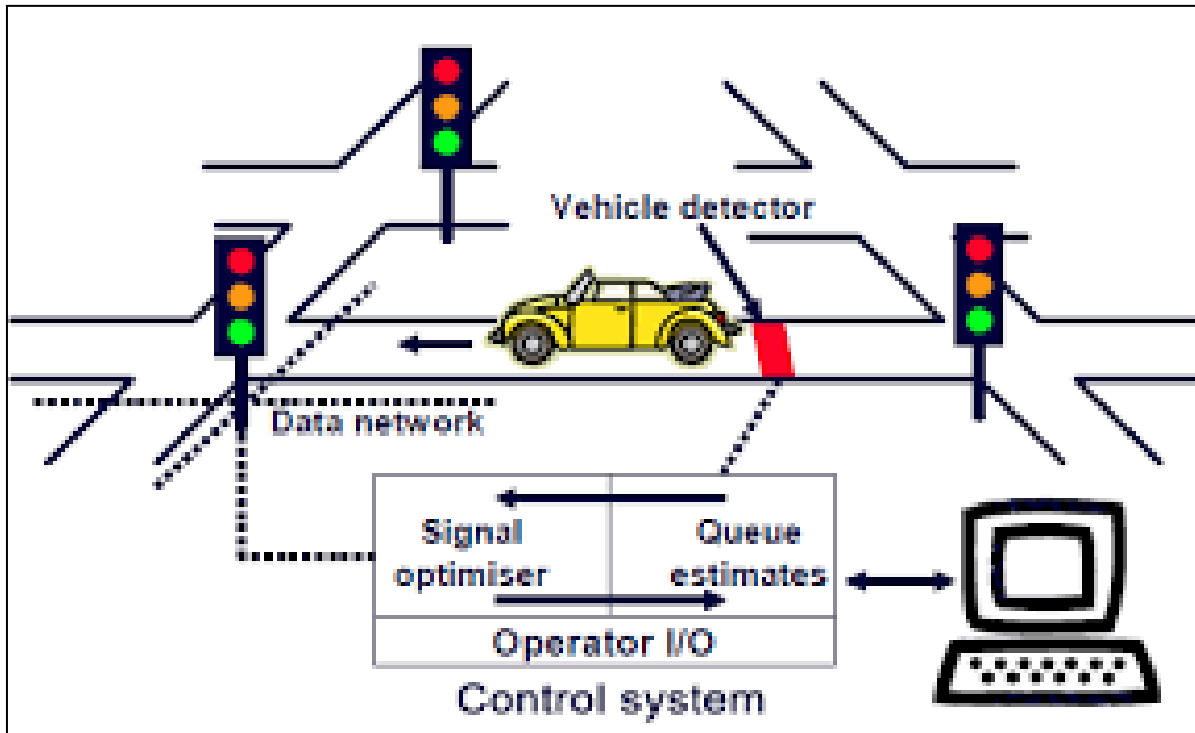


Figure 1.2: Smart Traffic Control Model

This traffic control system aims to optimize traffic flow of vehicles. As the number of road users increases exponentially, and current infrastructures provide resources which are limited thus intelligent control of traffic will become a very important issue in future as avoiding traffic jams will also be beneficial to both environment and economy.

However, the key to a successful traffic management is the use of data from different real-time sources and processing information obtained to take immediate decisions. It is the need of the hour to leverage enormous amount of data and create a more meaningful and smooth living for us.

1.2 MOTIVATION

- a) Smart City Mission launched by PM of India under Ministry Of Urban development.
- b) To fulfill the demands of Smart city model
- c) In modern life we have to face many problems one of which is traffic congestion, thus solution to this problem will solve many interlinked issues like fuel consumption, time management, ease of transportation and environmental pollution.
- d) Core infrastructure elements in a Smart City include “efficient urban mobility and transportation system”.

1.3 OBJECTIVES

This project is aimed towards designing a smart traffic control system using sensors and with the help of PIR sensor's input based on vehicle density and also to solve traffic congestion which is a severe problem in all modern cities all over the world. To solve this problem, we have a framework for a dynamic and automatic traffic light control system.

Generally, each traffic light on intersection is assigned a constant red and green signal time. It is possible to propose a dynamic real-time based coordination scheme where the red and green signal time of the traffic lights is assigned based on the present density of the traffic. In this project, we have adopted an approach to take data/input from vehicle and to process the input data using Microcontroller; after that red light and green light are assigned time based on the processing and finally displays it on the traffic light signal to control the Closed Loop System.

1.4 SCOPE OF THE PROJECT

This project is a basic step towards smart city plan of India. In today's world where basic electronics is finding its place in many day to day fields like home automation, car automation, automatic water storage system and so on, this will also take us one step further in Smart City plan. Recently government of India is spending round about 2 crores per city on 98 cities in India which are competing to become India's first Smart city thus this concept of traffic light control and transferring the traffic congestion scenarios of the city on real time basis on devices like smart phones can prove to be a good use of technology to solve day to day problems. The scope of this project is not just limited till controlling of traffic light timings based on density of vehicles but can go further by using GSM technology to transfer traffic density information to the mobile phones of travelers or controlling traffic based on priority.

CHAPTER 2

HARDWARE DESCRIPTION

2.1 ARDUINO UNO

Arduino is an open source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, hobbyists, and anyone interested in creating interactive objects or environments. It's an open-source physical computing platform based on a microcontroller board, and a development environment for writing software for the board and we can see the image of Arduino Uno in figure 2.1.



Figure 2.1: Arduino Uno [12]

In simple words, Arduino is a small microcontroller board with a USB plug to connect to your computer and a number of connection sockets that can be wired up to external electronics, such as motors, relays, light sensors, laser diodes, loudspeaker, microphones, etc., They can

either be powered through the USB connection from the computer or from a 9V battery. They can be controlled from the computer or programmed by the computer and then disconnected and allowed to work independently. Anyone can buy this device through online auction site or search engine. Since the Arduino is an open-source hardware designs and create their own clones of the Arduino and sell them, so the market for the boards is competitive. The name “Arduino” is reserved by the original makers. However, clone Arduino designs often have the letters “duino” on the end of their name, for example, Freeduino or DFRduino. The software for programming your Arduino is easy to use and also freely available for Windows, Mac, and LINUX computers at no cost.

Microcontroller can be described as a computer embedded on a rather small circuit board. To describe the function of a microcontroller more precisely, it is a single chip that can perform various calculations and tasks, and send/receive signals from other devices via the available pins. Precisely what tasks and communication with the world it does, is what is governed by what instructions we give to the Microcontroller. It is this job of telling the chip what to do, is what we refer to as programming on it. However, the microcontroller by itself cannot accomplish much; it needs several external inputs: power, for one; a steady clock signal, for another. So typically, a microcontroller is used along with a circuit which provides these things to it; this combination is called a microcontroller board. The Arduino Uno that we are using is one such microcontroller board. The actual microcontroller at its heart is the chip called Atmega328. The advantages that Arduino offers over other microcontroller boards are largely in terms of reliability of the circuit hardware as well as the ease of programming and using it.

Open-source hardware shares much of the principles and approach of free and open-source software. The founders of Arduino wanted people to study their hardware, to understand how it works, make changes to it, and share those changes with the world. To facilitate this, they release all of the original design files (Eagle CAD) for the Arduino hardware. These files are licensed under a Creative Common Attribution Share-Alike license, which allows for both personal and commercial derivative works, as long as they (people) credit Arduino and release their designs under the same license. The Arduino software is also open source. The source code for the Java environment is released under the GPL and the C/C++ microcontroller libraries are under the LGPL.

2.1.1 Arduino Pin Diagram

Pin diagram of Arduino Uno is shown in figure 2.2.

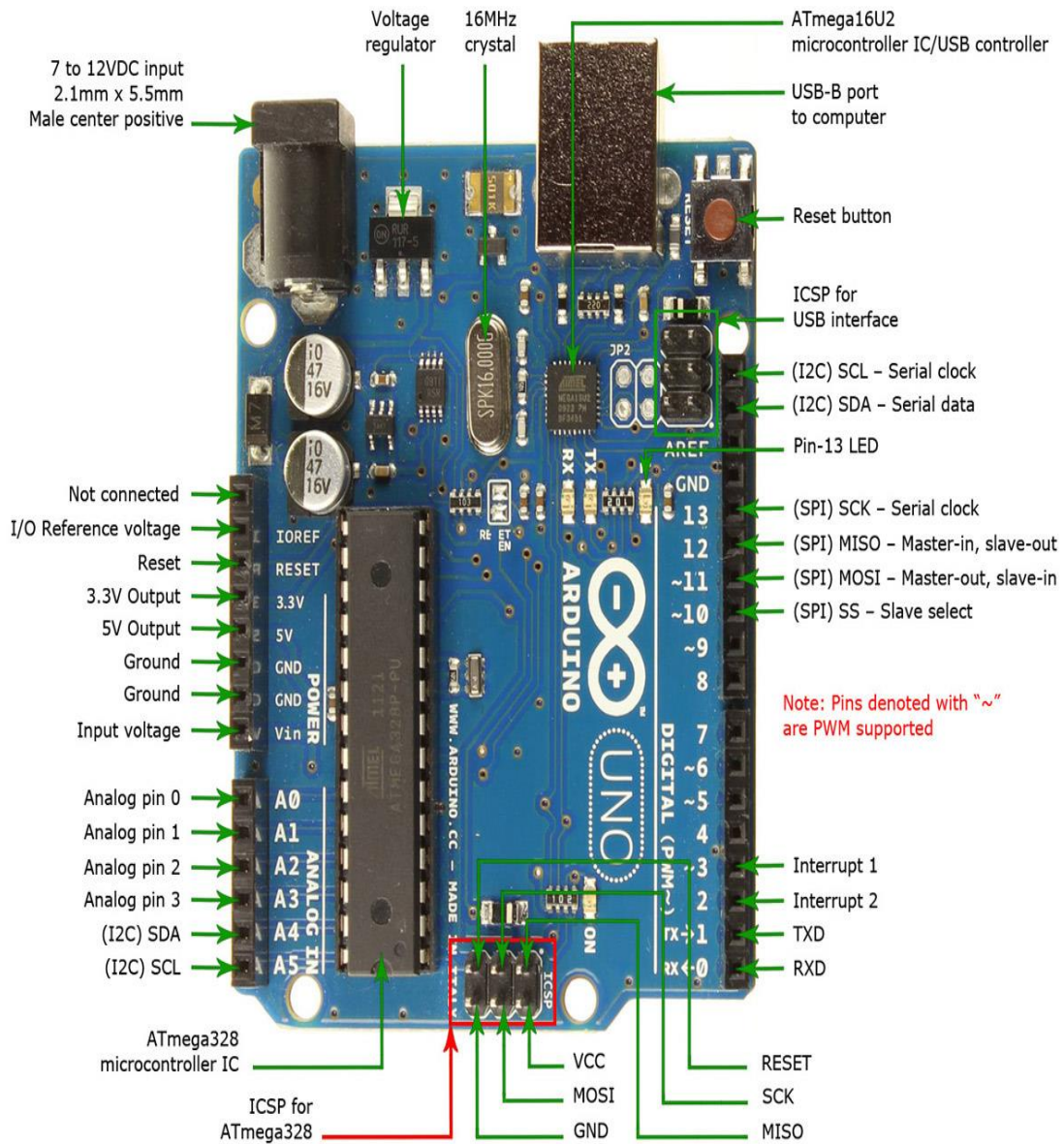


Figure 2.2: Arduino pin diagram [12]

2.1.2 Technical Specification

There are certain technical specifications of Arduino Uno as shown in table 2.1.

Table 2.1: Technical Specification

Microcontroller	<u>ATmega328P</u>
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

Microcontroller

The ATmega328 is a single chip microcontroller created by Atmel as shown in figure 2.3. As of 2013 the ATmega328 is commonly used in many projects and autonomous systems where a simple, low-powered, low-cost micro-controller is needed. Perhaps the most common implementation of this chip is on the popular Arduino development platform, namely the Arduino Uno and Arduino Nano models. A common alternative to the ATmega328 is the "picoPower" ATmega328P. A comprehensive list of all other member of the megaAVR series can be found on the Atmel website.

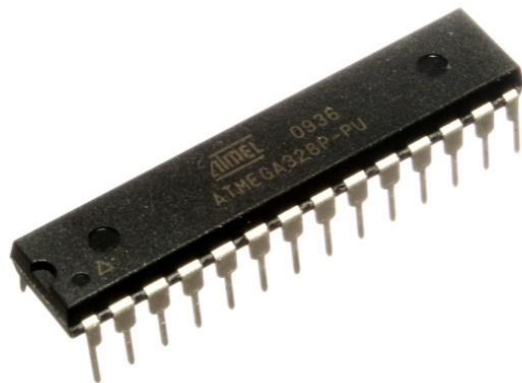


Figure 2.3: ATmega328 (Microcontroller) [13]

The Atmel 8-bit AVR RISC-based microcontroller combines 32 kB flash memory with read-while-write capabilities, 1 kB EEPROM, 2 kB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, 6-channel 10-bit A/D converter, programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts. The device achieves throughput approaching 1 MIPS per MHz

2.1.3 Power Supply

The Arduino's on-board regulator can actually handle up to 20V or more, so you can actually use an adapter that puts out 20V DC. The reasons you don't want to do that are twofold: you'll lose most of that voltage in heat, which is terribly inefficient. Secondly, the nice 9V pin on the Arduino board will actually be putting out 20V or so, which could lead to potential disaster when you connect something expensive to what you thought was the 9V pin. Our advice is to stick with the 9V or 12V DC adapter.

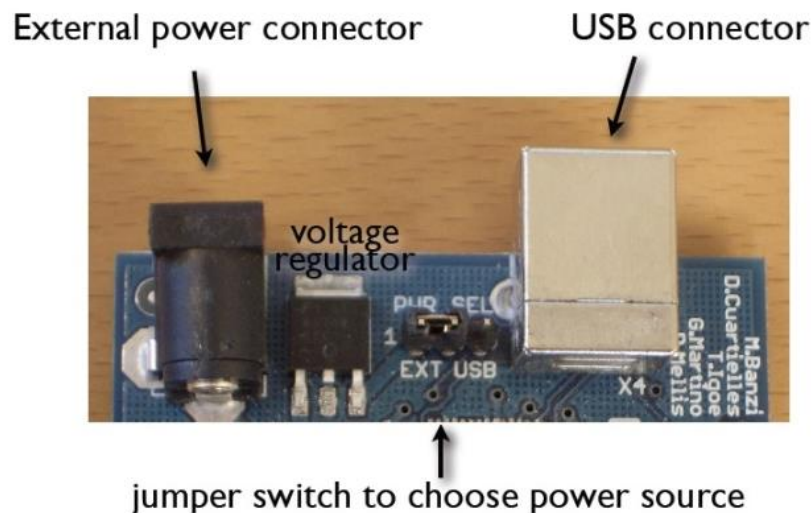


Figure 2.4: External Power

The power requirement for ARDUINO is 9 to 12V DC, 250mA or more, 2.1mm plug, center pin positive.

The OFF-the shelf adapter

- must be a DC adapter (i.e. it has to put out DC, not AC)
- should be between 9V and 12V DC
- must be rated for a minimum of 250mA current output, although you will likely want something more like 500mA or 1A output, as it gives you the current necessary to power a servo or twenty LEDs if you want to.
- must have a 2.1mm power plug on the Arduino end, and

- The plug must be "centre positive", that is, the middle pin of the plug has to be the + connection.

Current rating: Since you'll probably be connecting other things to the Arduino (LEDs, LCDs, servos) you should get an adapter that can supply at least 500mA, or even 1000 mA (1 ampere). That way you can be sure you have enough juice to make each component of the circuit function reliably.

2.1.4 Software

The software used by the Arduino is Arduino IDE .he Arduino IDE is a cross-platform application written in Java, and is derived from the IDE for the Processing programming language and the Wiring project. It is designed to introduce programming to artists and other newcomers unfamiliar with software development. It includes a code editor with features such as syntax highlighting, brace matching, and automatic indentation, and is also capable of compiling and uploading programs to the board with a single click. There is typically no need to edit make files or run programs on a command-line interface. Although building on command-line is possible if required with some third-party tools such as ino.

The Arduino IDE comes with a C/C++ library called "Wiring" (from the project of the same name), which makes many common input/output operations much easier. Arduino programs are written in C/C++, although users only need define two functions to make a runnable program.

2.2 PIR SENSOR

PIR sensors allow you to sense motion, almost always used to detect whether a object has moved in or out of the sensors range. Infrared radiation exists in the electromagnetic spectrum at a wavelength that is longer than visible light. Infrared radiation cannot be seen but it can be detected. Objects that generate heat also generate infrared radiation including animals and the human body whose radiation is strongest at a wavelength of 9.4mm. 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 as shown in figure 2.5.

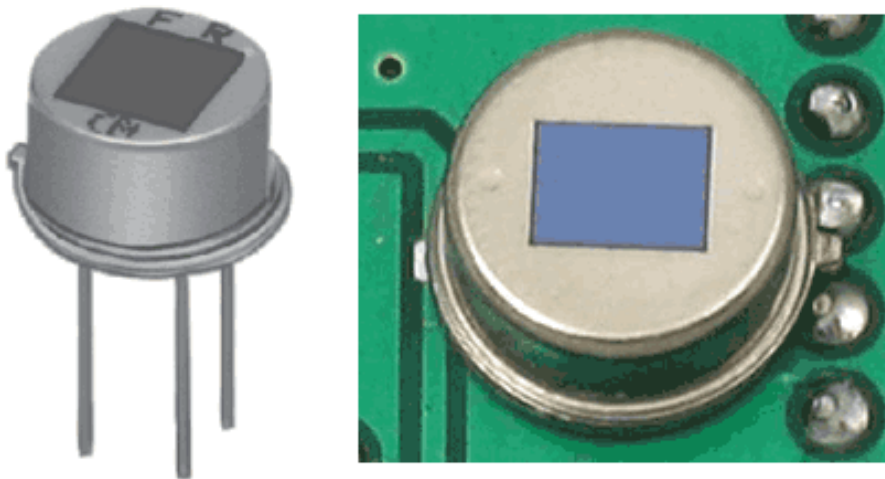


Figure 2.5: Pyroelectric Sensor [14]

PIRs are basically made of a pyroelectric sensor (which you can see above 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.

The module has three more pins with a jumper between two of them as shown in fig 2.6. These pins are for selecting the trigger modes. The first one is called “non-repeatable trigger” and works like this: when the sensor output is high and the delay time is over, the output will automatically change from high to low level. The other mode called “repeatable trigger” will keep the output high all the time until the detected object is present in sensor’s range.

An increased sensitivity can be beneficial for when using a PIR for long range, say up to 20 feet (6 meters) or more. This can however also cause false positives in smaller spaces – i.e. the PIR might trigger when it should not. A decreased sensitivity is good for a short range; say half of the maximum range or up to 10 feet (3 meters). Which in turn could miss movement at a longer range? Objects further away may need more motion to be detected.

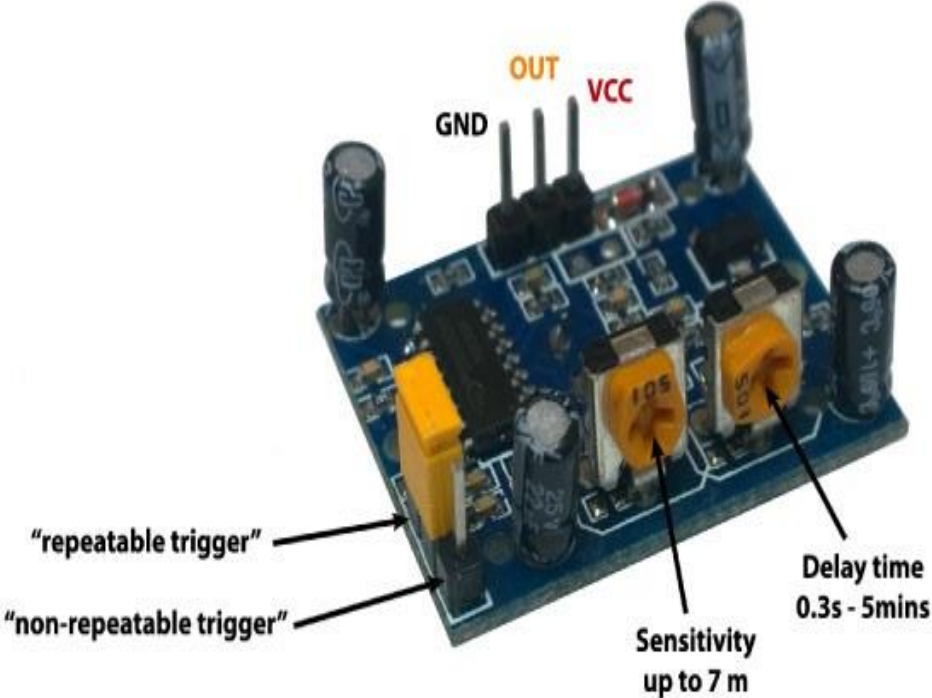


Figure 2.6: PIR Sensor (Bottom View)

The “L” setting seems to behave a little erratic – when you move in front of the PIR, you will see that it occasionally switches the LED ON and then OFF again. This is called “non-retriggering” – this setting does not or just barely retrigger. When using the “H” (default)

setting, you will see that the LED remains ON when moving in front of the PIR. This is what is called “retriggering” and it seems that this would be the most used setting as this seems to behave the nicest.

For many basic projects or products that need to detect when a person has left or entered the area, or has approached, PIR sensors are great. They are low power and low cost, pretty rugged, have a wide lens range, and are easy to interface with.



Figure 2.7: PIR Sensor (Top View)

A top view of PIR sensor as shown in fig 2.7 contain a Fresnel lens which is a Plano Convex lens that has been collapsed on itself to form a flat lens that retains its optical characteristics but is much smaller in size and has less absorption losses.

2.2.1 Working

The PIR sensor itself has two slots in it; each slot is made of a special material that is sensitive to IR. The lens used here is not really doing much and so we see that the two slots can 'see' out past some distance (basically the sensitivity of the sensor). When the sensor is idle, both slots detect the same amount of IR, the ambient amount radiated from the room or walls or outdoors. When a warm body passes by then it first intercepts one half of the PIR sensor, which causes a positive differential change between the two halves. When the warm body leaves the sensing area, the reverse happens, whereby the sensor generates a negative differential change. These change pulses are what is detected as shown in figure 2.8.

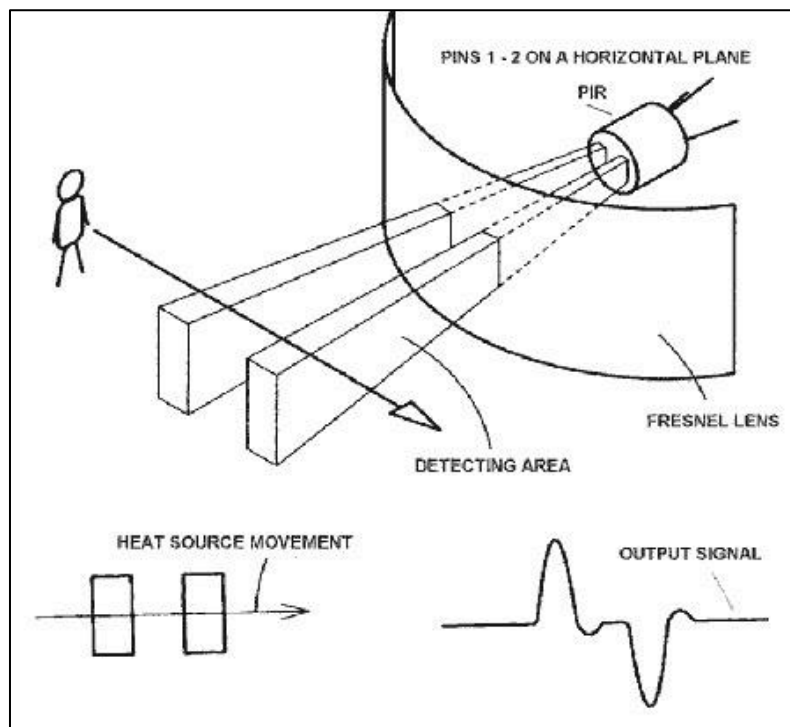


Figure 2.8: PIR working [14]

The IR sensor itself is housed in a hermetically sealed metal can to improve noise/temperature/humidity immunity. There is a window made of IR-transmissive material (typically coated silicon since that is very easy to come by) that protects the sensing element. Behind the window are the two balanced sensors.

PIR sensors are rather generic and for the most part vary only in price and sensitivity. Most of the real magic happens with the optics. This is a pretty good idea for manufacturing: the PIR sensor and circuitry is fixed and costs a few dollars. The lens costs only a few cents and can change the breadth, range, sensing pattern, very easily.

The lens is just a piece of plastic, but that means that the detection area is just two rectangles. Usually we'd like to have a detection area that is much larger. To do that, we use a simple lens such as those found in a camera: they condense a large area (such as a landscape) into a small one (on film or a CCD sensor). For reasons that will be apparent soon, we would like to make the PIR lenses small and thin and moldable from cheap plastic, even though it may add distortion. For this reason the sensors are actually Fresnel lenses as shown in figure 2.9

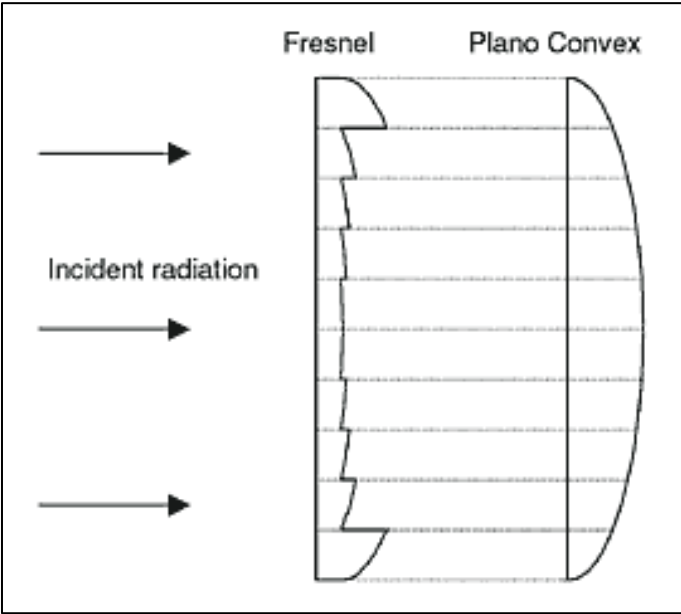


Figure 2.9: Fresnel lens [14]

The Fresnel lens condenses light (shown in figure 2.10), providing a larger range of IR to the sensor. If once the sensor gets warmed up, then the output remains low until it detects motion. If once it detects the motion, then the output goes high for a couple of seconds and then returns to a normal state or low.

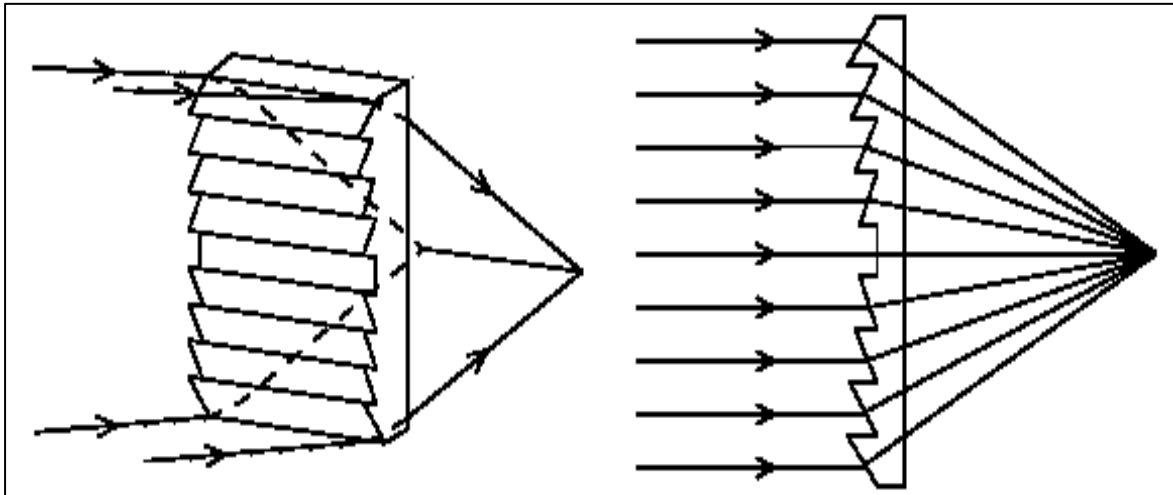


Figure 2.10: Converging of rays from Fresnel lens [14]

A Fresnel lens is usually thin and flexible and is about 0.015 inch (0.38mm) thick with grooves molded into one surface. The groove side of the lens usually faces the PIR sensor. A Fresnel lens both captures more IR radiation and focuses it to a small point. This focal point moves across the sensor as the IR source moves and exposes one element at a time. A Fresnel lens can extend detection range to about 100 feet. It focuses the infrared signals onto the pyroelectric sensor.

2.3 LIQUID CRYSTAL DISPLAY (LCD)

A Liquid Crystal Display commonly abbreviated as LCD is basically a display unit built using Liquid Crystal technology. When we build real life/real world electronics based projects, we need a medium/device to display output values and messages. The most basic form of electronic display available is 7 segment display – which has its own limitations. The next best available option is Liquid Crystal Displays which comes in different size specifications. Out of all available LCD modules in market, the most commonly used one is 16×2 LCD Module which can display 32 ASCII characters in 2 lines (16 characters in 1 line). Here one must understand that in each character there are 5x10=50 pixels so to display one character all 50 pixels must work together. But we need not to worry about that because there is another controller (HD44780) in the display unit which does the job of controlling the pixels. Other commonly used LCD displays are 20×4 Character LCD, Nokia 5110 LCD module, 128×64 Graphical LCD Display and 2.4 inch TFT Touch screen LCD display.

2.3.1 16×2 Pin out Diagram

The JHD162A LCD module has 16 pins and can be operated in 4-bit mode or 8-bit mode. Here we are using the LCD module in 4-bit mode. Before going in to the details of the project, let's have a look at the JHD162A LCD module. The schematic of a JHD162A LCD pin diagram as shown in figure 2.11

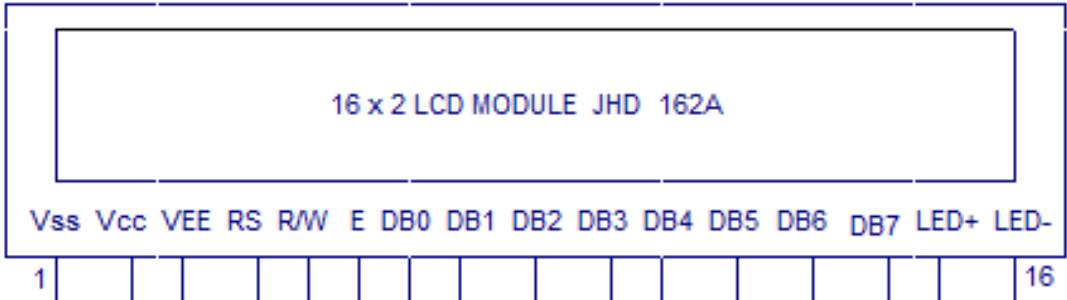


Figure 2.11: LCD pin out

The name and functions of each pin of the 16×2 LCD module is given below.

Pin1 (Vss): Ground pin of the LCD module.

Pin2 (Vcc): Power to LCD module (+5V supply is given to this pin)

Pin3 (VEE): Contrast adjustment pin. This is done by connecting the ends of a 10K potentiometer to +5V and ground and then connecting the slider pin to the VEE pin. The voltage at the VEE pin defines the contrast. The normal setting is between 0.4 and 0.9V.

Pin4 (RS): Register select pin. The JHD162A has two registers namely command register and data register. Logic HIGH at RS pin selects data register and logic LOW at RS pin selects command register. If we make the RS pin HIGH and feed an input to the data lines (DB0 to DB7), this input will be treated as data to display on LCD screen. If we make the RS pin LOW and feed an input to the data lines, then this will be treated as a command (a command to be written to LCD controller – like positioning cursor or clear screen or scroll).

Pin5 (R/W): Read/Write modes. This pin is used for selecting between read and write modes. Logic HIGH at this pin activates read mode and logic LOW at this pin activates write mode.

Pin6 (E): This pin is meant for enabling the LCD module. A HIGH to LOW signal at this pin will enable the module.

Pin7 (DB0) to Pin14 (DB7): These are data pins. The commands and data are fed to the LCD module through these pins.

Pin15 (LED+): Anode of the back light LED. When operated on 5V, a 560 ohm resistor should be connected in series to this pin. In Arduino based projects the back light LED can be powered from the 3.3V source on the Arduino board.

Pin16 (LED-): Cathode of the back light LED.

2.3.2 Interfacing with Arduino Uno

The circuit diagram of interfacing LCD to arduino for displaying a text message is shown in figure 2.12

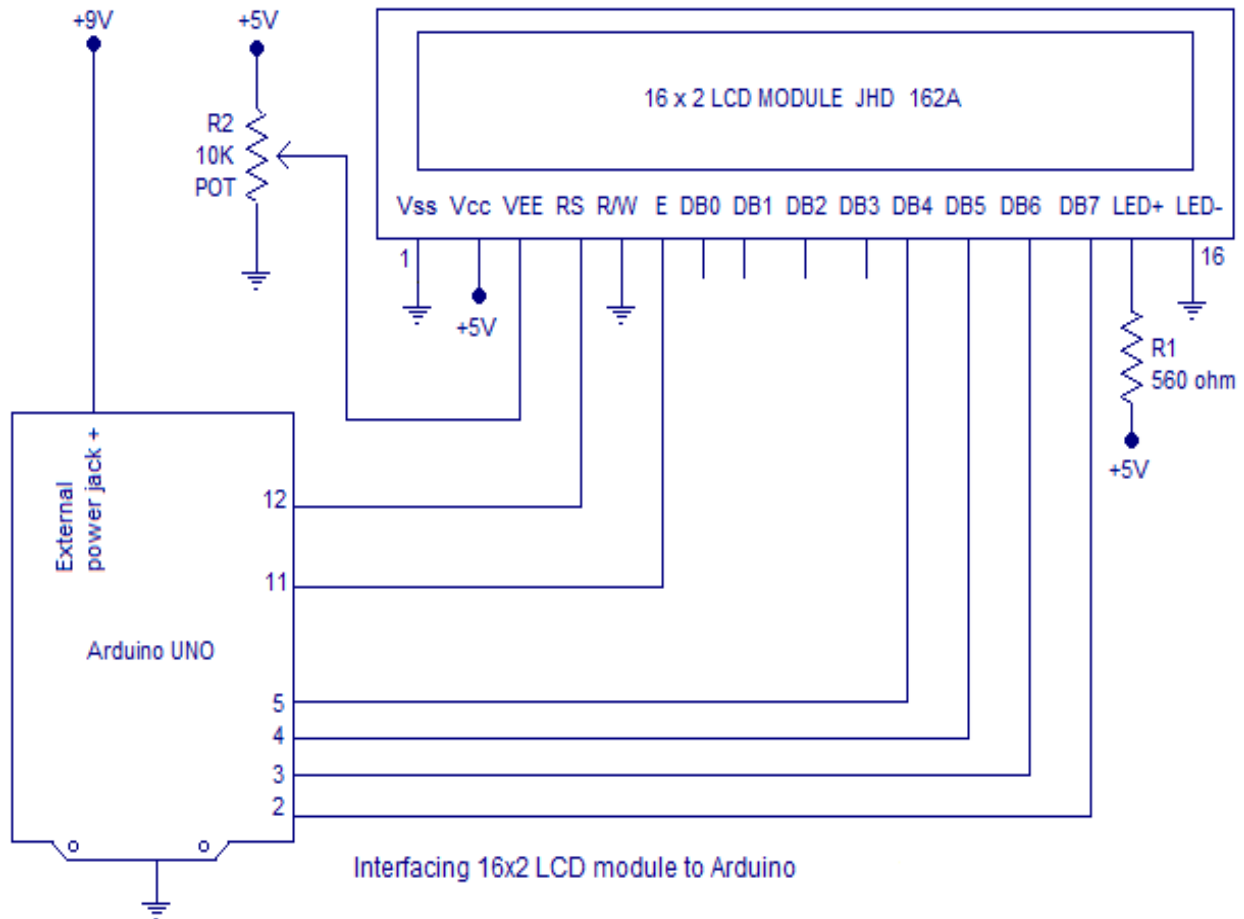


Figure 2.12: Circuit diagram – Arduino to 16×2 LCD

RS pin of the LCD module is connected to digital pin 12 of the Arduino. R/W pin of the LCD is grounded. Enable pin of the LCD module is connected to digital pin 11 of the Arduino. In this project, the LCD module and Arduino are interfaced in the 4-bit mode. This means only four of the digital input lines (DB4 to DB7) of the LCD are used.

This method is very simple, requires less connections and you can almost utilize the full potential of the LCD module. Digital lines DB4, DB5, DB6 and DB7 are interfaced to digital pins 5, 4, 3 and 2 of the Arduino. The 10K potentiometer is used for adjusting the contrast of the display. 560 ohm resistor R1 limits the current through the back light LED. The arduino can be powered through the external power jack provided on the board. +5V required in some other parts of the circuit can be tapped from the 5V source on the arduino board. The arduino can be also powered from the PC through the USB port.

The connections which are done for LCD are given below:

PIN1 or VSS to ground

PIN2 or VDD or VCC to +5v power

PIN3 or VEE to ground (gives maximum contrast best for a beginner)

PIN4 or RS (Register Selection) to PIN0 of ARDUINO UNO

PIN5 or RW (Read/Write) to ground (puts LCD in read mode eases the communication for user)

PIN6 or E (Enable) to PIN1 of ARDUINO UNO

PIN11 or D4 to PIN8 of ARDUINO UNO

PIN12 or D5 to PIN9 of ARDUINO UNO

PIN13 or D6 to PIN10 of ARDUINO UNO

PIN14 or D7 to PIN11 of ARDUINO UNO

To facilitate communication between Arduino and LCD module, we make use of a built in library in Arduino <LiquidCrystal.h> – which is written for LCD modules making use of the Hitachi HD44780 chipset (or a compatible chipset). This library can handle both 4 bit mode and 8 bit mode wiring of LCD.

Library “LiquidCrystal.h” is used for easily controlling the LCD module using Arduino board with the help of built in methods defined inside the library For example, data string can be printed on the LCD module by merely calling a method `lcd.print()`. If you want to print “Hello World” at row 1, starting from column3; first set the cursor at the desired position using method `lcd.setCursor(1,3)` and then write the command to print the characters as

lcd.print("Hello World"). The library is readily available with the Arduino IDE (as it's a pre-installed standard library).

Any library can be accessed manually through the "Import library" in the "sketch" tab in the main menu bar. The LiquidCrystal.h library provides functions/methods for almost all applications like printing a string, setting the cursor, initializing the LCD, scrolling the display, auto scroll, clear LCD, blink cursor etc.

LiquidCrystal lcd() – is a constructor used to declare a variable of its type. Here 'lcd' is the variable declared using the constructor and is used to invoke methods defined inside the library LiquidCrystal.h (Example – lcd.print(); lcd.setCursor() and other methods)

lcd.begin() – is called to initialize the lcd screen and to pass the dimension of lcd screen (columns, rows) as parameters of the invoked method.

Scroll() - A simple program for scrolling a text message on the LCD screen using arduino is shown here. This is done using the method defined inside LiquidCrystal.h library.

lcd.scrollDisplayRight() - will scroll the display to right

lcd.scrollDisplayLeft() - will scroll the display to left. A "for" loop is used for selecting the number of positions to scroll at a time.

2.4 RF RECEIVER AND TRANSMITTER PAIR

A wireless radio frequency (RF) transmitter and receiver can be easily made using Decoder, Encoder and ASK RF Module. Wireless transmission can be done by using 433 MHz or 315MHz ASK RF Transmitter and Receiver modules. In these modules digital data is represented by different amplitudes of the carrier wave, hence this modulation is known as Amplitude Shift Keying (ASK). Radio Frequency (RF) transmission is more strong and reliable than Infrared (IR) transmission due to following reasons:

- Radio Frequency signals can travel longer distances than Infrared.
- Only line of sight communication is possible through Infrared while radio frequency signals can be transmitted even when there are obstacles.
- Infrared signals will get interfered by other IR sources but signals on one frequency band in RF will not interfered by other frequency RF signals.

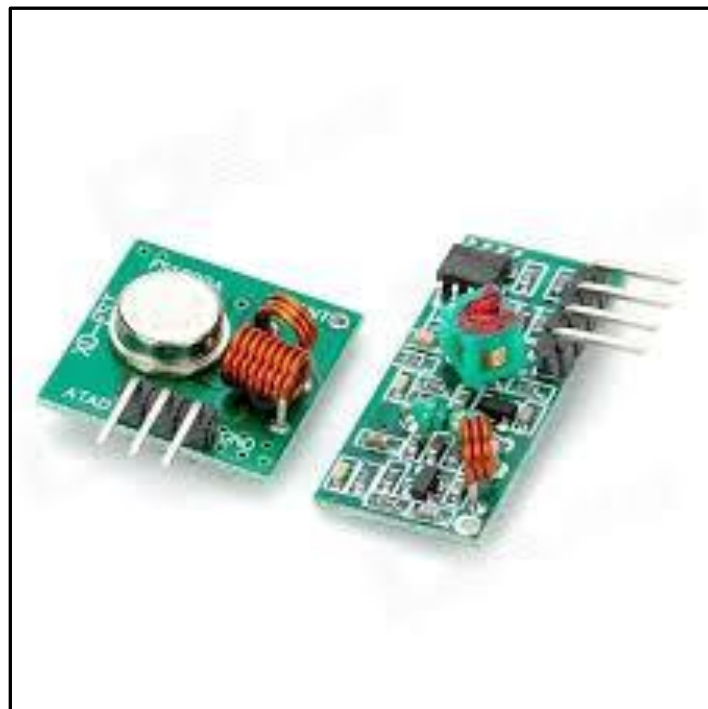


Figure 2.13 RF Receiver and Transmitter pair

Receiver:

Working voltage: 5.0VDC +0.5V

Working current: $\leq 5.5\text{mA}$ max

Working method: OOK/ASK

Working frequency: 315MHz-433.92MHz

Bandwidth: 2MHz

Sensitivity: excel -100dBm (50Ω)

Transmitting velocity: $< 9.6\text{Kbps}$ (at 315MHz and -95dBm)

Transmitter:

Working voltage: 3V - 12V for max. power use 12V

Working current: max Less than 40mA max , and min 9mA

Modulation mode: ASK

Working frequency: 315MHz Or 433MHz

Transmission power: 25mW (315MHz at 12V)

Frequency error: +150kHz (max)

Velocity : less than 10Kbps

CHAPTER 3

DESIGN METHODOLOGY

3.1 PROPOSED METHODOLOGY

Basic Block diagram of proposed model is composed of a microcontroller, sensors, LED lights (in place of traffic lights), memory elements to store the data sensed, LCD and power supply as shown in figure 3.1

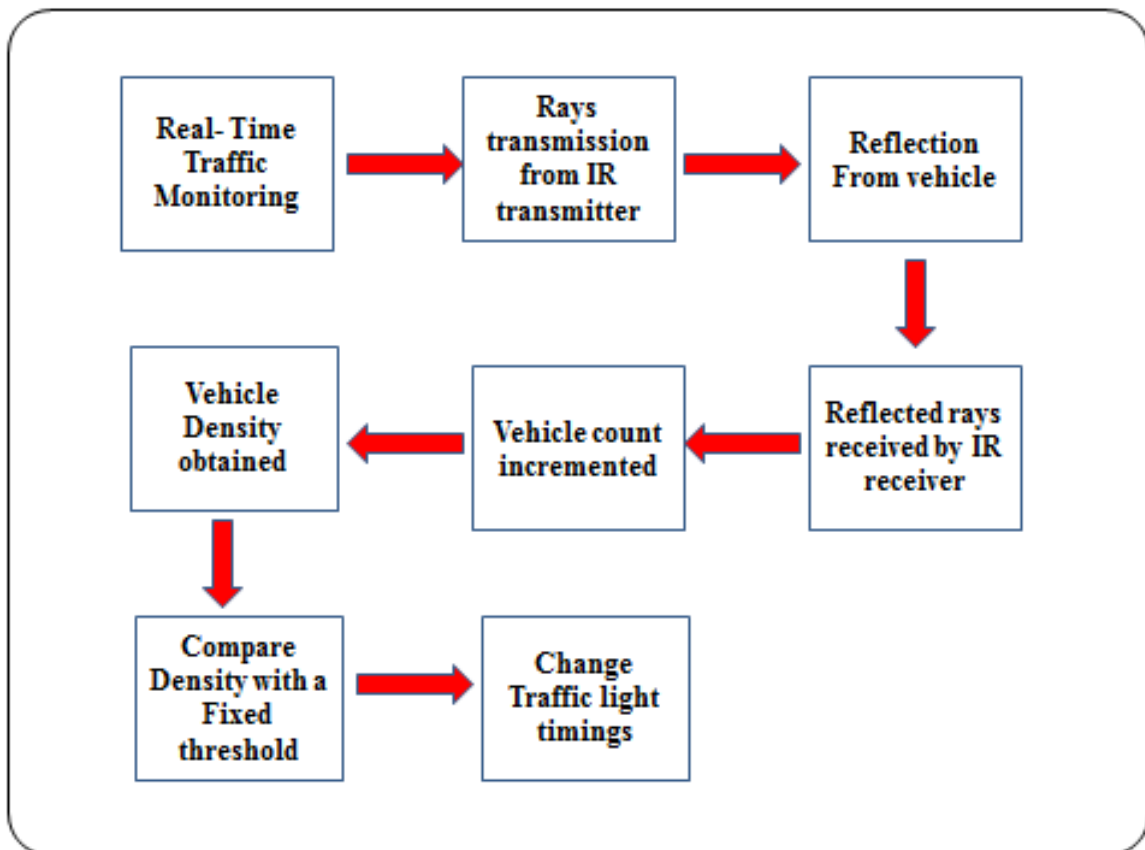


Figure 3.1: Block Diagram of Working Model

Project requires real time monitoring of the road. There is continuous ray transmission from PIR sensors when an object crosses in front of PIR sensor then it detects the heat emitted by it. For vehicles this heat is emitted from the engine. PIR sensors include repeatable and non-repeatable trigger through which we can control the counting of vehicles in such a way that the sensor detects only vehicle in motion. As soon as the reflected rays are sensed, vehicle count is incremented.

This counting process is done for a specific time interval and after that density of vehicles crossed is obtained which is then compared with a fixed threshold and according to which various timings are allotted to red and green lights. During waiting time and green light timing, the counting process keeps on going and the decision on that density obtained is thus taken in next cycle and so on.

There is a wide range of sensor technologies available for vehicle detection. We are using “Passive IR Sensors” in our project.

This sensor module has great adaptive capability of the ambient light and an infrared emitting tube to emit a certain frequency, encounters an obstacle detection direction (reflecting surface), infrared reflected back to the receiver tube receiving. When the sensor is idle, both slots detect the same amount of IR, the ambient amount radiated from the room or walls or outdoors.

When a warm body passes by, it first intercepts one half of the PIR sensor, which causes a positive differential change between the two halves. When the warm body leaves the sensing area, the reverse happens, whereby the sensor generates a negative differential change. These change pulses are what is detected. And thus an object is detected.

PIR sensors are digital sensors whenever an object is sensed they give output “1” else “0”. Once the traffic density is obtained then after a certain fixed interval of time depending upon general vehicle rush and width of the road, this vehicle density is compared with a certain threshold and based on that further processing is done.

3.2 WORKING

Count is initialized to zero after a certain time. Numbers of vehicles are counted continuously and after every cycle of traffic signal i.e. from Red to Green next timings run based on previous density obtained.

LCD display is controlled with the help of potentiometer, by adjusting its resistance we can the brightness of the screen of LCD. The whole project is divided into two modules.

Decision-making is done as shown in figure 3.2

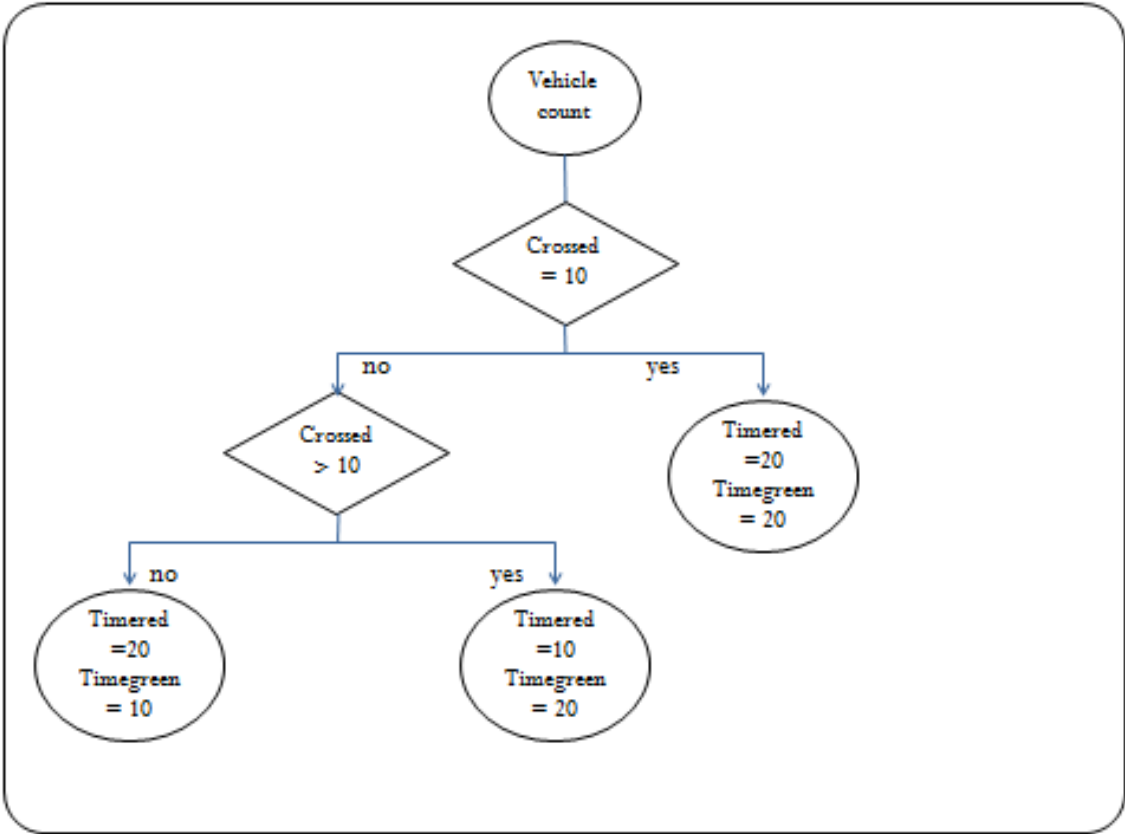


Figure 3.2: Decision making flowchart

3.2.1 Module 1

Module 1 is based on data obtained by sensors. This data is send to microcontroller as an input and based on this data various timing are given to green and red lights based on the flowchart shown in figure 3.2.

CASE 1: When density of vehicles is equal to 10

This is the general case assuming that normal vehicle density on road is 10. When this case is satisfied then time assigned to red light is 20 seconds and green light is also 20 seconds.

CASE 2: When density of vehicles is greater than 10

This is the case for high vehicle density on road. When this case is satisfied then time assigned to red light is 10 seconds and green light is also 20 seconds. This is done as we want the vehicles to stop for less time because stopping for more time will lead to more traffic congestion at the back.

Greater time is assigned to green light because we want more vehicles to pass when signal is green to decrease the probability of traffic congestion for coming vehicles and at the junction.

CASE 3: When density of vehicles is less than 10

This is the case when we assume that vehicle density is less than normal. When this case is satisfied then time assigned to red light is 20 seconds and green light is also 10 seconds.

To make the vehicle density constant on the road we have assigned 20 seconds time to red light as waiting for normal time will not lead to traffic congestion because density is low. Then green light is on for 10 seconds only because less vehicles are there so they will take less time to pass the traffic light junction.

3.2.2 Module 2

Module 2 deals about transferring the information recorded by sensor to a distantly placed microcontroller. In our previous module we have assumed that the sensors and microcontroller are at the same place and the decision making is done at that point itself.

For this module we are using RF transmitter receiver pair. RF transmitter part will be attached to sensors and receiver part will be attached to Arduino Uno to transfer the data wirelessly. This is basically done to signify more practically working smart traffic model and to increase efficiency of the system as that one microcontroller can act as hub and take decisions for more than one road sensor.

CHAPTER 4

RESULTS AND CONCLUSION

4.1 RESULTS

With the help of ARDUINO UNO and PIR sensor obstacle detector, the code helped us to count the number of objects sensed. Whenever an object which has reflecting surface comes in front of the IR receiver transmitter, the value sensed by IR pair is 1 and when not then it is 0. Circuit implementation and the output obtained on serial monitor window is shown in figure 4.1

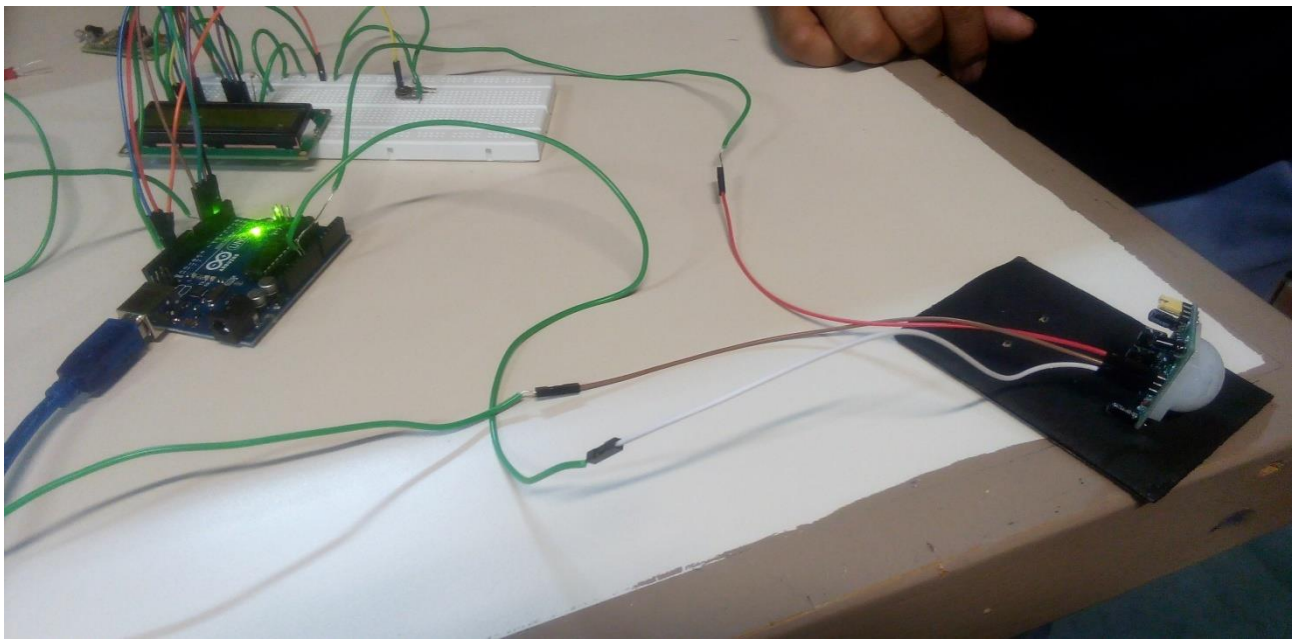


Figure 4.1: Circuit Implementation

With the help of the circuit as shown figure 4.1 we have implemented all three cases which are possible according to our project.

We have run various tests for each case to verify our results which are shown in table 4.1

Table 4.1: Observation for different vehicle count

NUMBER OF VEHICLES CROSSED	RED LIGHT TIMING (IN SECONDS)	GREEN LIGHT TIMING (IN SECONDS)
1. 10	20	20
2. 7	20	10
3. 12	10	20
4. 5	20	10

Various timings which are allotted to red and green light according to the traffic density are shown using LCD. For the allotted time red and green LEDs lit up as shown in figure 4.2a and figure 4.2b.

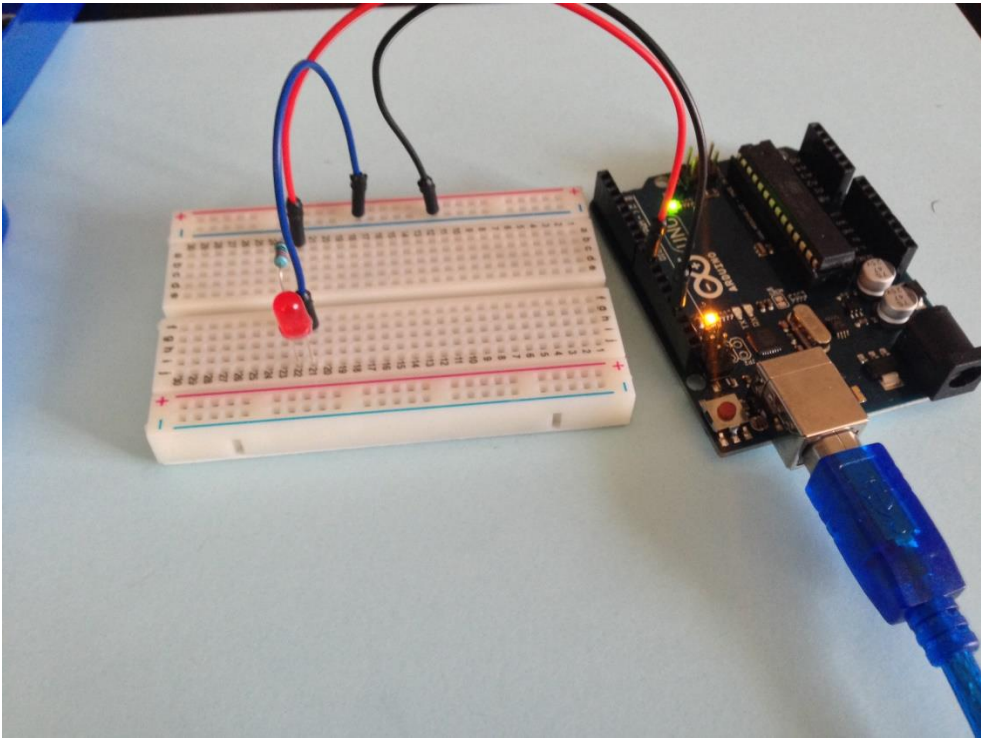


Figure 4.2a: Implementation of the results obtained

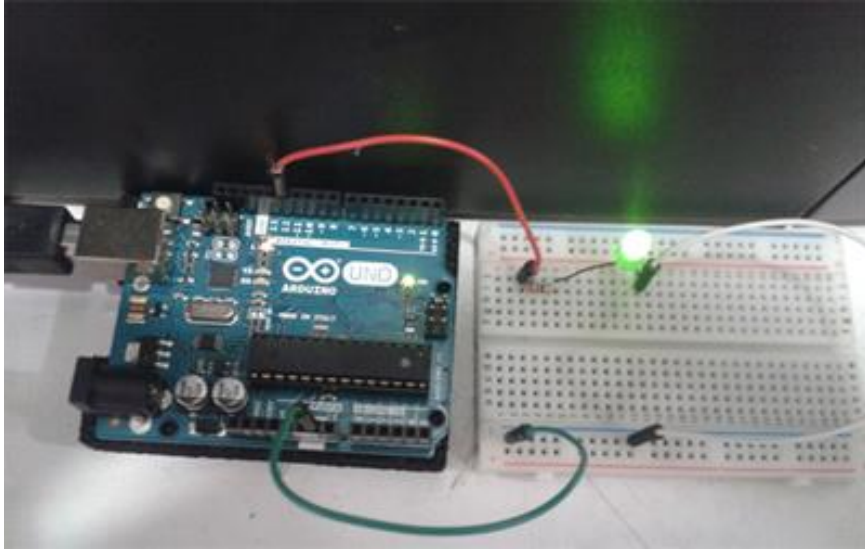


Figure 4.2b: Implementation of the results obtained

Now further aim is to transfer this information wirelessly to a hub. This is implemented by Arduino to Arduino information transfer using RF receiver transmitter pair. In a practical model this will help us to use a single authority for a particular area, thus all traffic lights will report to hub and all decision making will be done at that place only and timings are changed according to algorithm applied at hub.

4.2 CONCLUSION

While working on this project, we came across the world of sensors and microcontrollers. Theoretical study of above mentioned topics was an easy task but implementation of algorithm using microcontroller and sensors took us for a ride. We have fulfilled our objective and done a successful completion of smart traffic control model. We have completed the concept of “SMART TRAFFIC CONTROL SYSTEM BASED ON VEHICLE DENSITY” with the help of hardware and software.

The system designed can prove to be a step forward in designing a dynamic traffic control model which can help to save time and fuel both. There are wide range of IR sensors which can be used practically and much more range than that which we have used and can be a better option than other sensors as they fixed on either side of the road which can make their maintenance and replacement easy. Similarly, there are many RF receiver transmitter pair module which can work efficiently on higher range. This model is just a sample designed to work in a room only hence we have used components with short ranges.

4.3 FUTURE SCOPE

ENERGY EFFICIENT

Sensor Networks deployed along road can be utilized to control the traffic load on roads and at traffic intersections. Sensors are placed on either side of roads at intersection points. These sensors run on both solar energy as well as battery. During bright and sunny conditions sensors have the capability to draw solar energy from sun light as shown in figure 4.2, and use battery power for functioning during night and cloud and foggy condition. By using solar energy we can save a lot of energy.



Figure 4.2: Lights and sensors running by solar panel [15]

UTILISATION OF TECHNOLOGIES

It is based on evaluation of an integrated traffic control and route diversion strategy. Traffic density management using GPS is an attempt for the creation of a smooth driving environment through a variety of recent technologies such as GPS and WI-FI. With these technologies we can make people not to wait for a long time in traffic signals when there are fewer vehicles on the road. This technology makes the signal system to work on the vehicle density basis whereas the existing system is fixed.

This system can be extended to personnel vehicles also. This technology can serve its purpose by use GPS as one can use it to choose shortest path between two place and can know traffic density of any road in the city so that he/she can decide the route to the destination accordingly. This technology can play a vital role in fuel savings and pollution control.

PRIORITY BASED TRAFFIC SYSTEM

Consider a scenario of highly congested area where many vehicles such as personal transport, public transport and emergency vehicles (Ambulance, Fire brigade, VIP cars and other rescue vehicles) have to wait for long for the change of traffic signals at intersection points. Existing traffic light systems have timers that are set at regular intervals. This leads to the wastage of precious time especially in case of rescue vehicles for emergency conditions.

In order to control this situation, we can propose a system consisting of two parts: Smart Traffic Light Control System (STLC) and Smart Congestion Avoidance System (SCA) during emergencies. STLC System controls the change of traffic lights at intersection points giving high priority to emergency vehicles. SCA System is a smart traffic routing system that chooses the shortest routes having the least congestions.

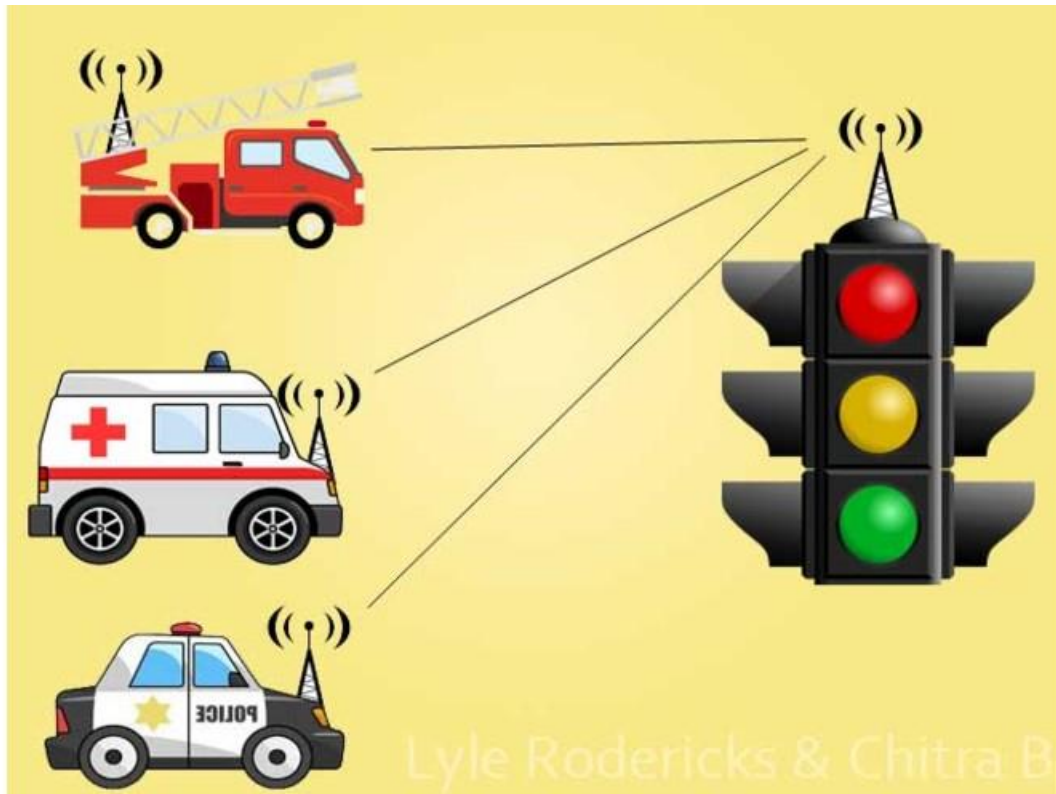


Figure 4.3: Priority based traffic control System

The vehicles are fitted with GPS as shown in figure 4.3, that will transmit the vehicle geo-location which will then be received by the road side unit at signal and it will start the green light and this light will be in ON state till the vehicle is within the GPS range. This will be useful for the emergency vehicles such as Fire Brigade, Ambulance, Police cars.

SMART ROUTING SYSTEM

In the last decade, intelligent transportation systems have progressed at a rapid rate, which aim to improve transportation activities in terms of safety and efficiency. Car-to-car and car-to-infrastructure communications are important components of the ITS architecture. GSM modules for communication and location tracking, speed and acceleration sensors for speed monitoring and display module for real time traffic pattern display Communication between cars and traffic lights is one of the important applications which help to have dynamic and automatic traffic lights that can create several benefits such as minimizing the traffic jam, reducing fuel consumption and emissions, etc. Here it deals with decreasing the response time of the emergency cars by changing the traffic lights status with employing the communication technologies.

Additionally, several statistics about traffic simulation is created for each car such as traveling time, waiting time, emissions, fuel consumption; or complete amount of car emissions in the street during the simulation, fuel consumption, and number of vehicles and so on, for each street. Here it presents evaluation of an integrated traffic control and route diversion strategy. By using all this data we can provide information about the traffic congestion and can provide the best suitable route for their destination which can save a lot of time and fuel too.

4.4 CHALLENGES FACED

1. Choice of an appropriate sensor for this model was an issue as various heat, weight and light sensors have different constraints in day and night.
2. Initially we planned to work on image processing but due to overlapping and night constraints the plan was dropped.
3. Configuring PIR sensor was a great deal. First we implemented a small module for counting number of vehicles crossed using an IR Proximity Module. Later on due to better performance and range we shifted to PIR sensors but due to its higher range and sensitivity it was tough to configure PIR sensor as it can detect anything in motion nearby.
4. Implementation of wireless transmission of data from sensor location to a hub requires Arduino to Arduino transmission using RF receiver transmitter pair. Configuration of two Arduino and serial communication between them took a lot of time.

REFERENCES

- [1] Ashwini Basavaraju, Senhalata Doddigarla, Navitha Naidu and Shruti Malgatti, “Vehicle density sensor system to manage traffic”, Volume: 03 Issue: 03, Mar-2014
- [2] Promila Sinhmar, “Intelligent traffic light and density control using IR sensors and microcontroller (IJATER)”, Volume 2, Issue 2, March 2012
- [3] Suhail Ahmad Bhat, Faiyqa Naqashbandi, Isna Khan, Naazira Badar, Rabaie Malik, "Smart Traffic Mess Control and Avoidance System", International Journal of Engineering Research & Technology (IJERT), Vol. 3, Issue 2, February – 2014
- [4] K.Vidhya, A.BazilaBanu, "Density Based Traffic Signal System", International Journal of Innovative Research in Science, Engineering and Technology, 2014 International Conference on Innovations in Engineering and Technology (ICIET'14), Volume 3, Special Issue 3, March 2014
- [5] RoufUIAlam Bhat, Suhail Ahmad Bhat, Faiyqa Naqashbandi, Isna Khan, NaaziraBadar, Rabaie Malik, “Smart Traffic Mess Control and Avoidance System”, International Journal of Engineering Research & Technology (IJERT), Vol. 3, Issue 2, February – 2014
- [6] Mario Collotta, Giovanni Pau, Gianfranco Scata, Tiziana Campisi, “a dynamic traffic light management system based on wireless sensor networks for the reduction of the red-light running phenomenon”, Transport and Telecommunication Institute, Lomonosova1, Riga, LV-1019, Latvia Transport and Telecommunication, 2014, volume 15,no 1, 1–11
- [7] Rashid Hussian, Sandhya Sharma, Vinita Sharma, Sandhya Sharma, “WSN Applications: Automated Intelligent Traffic Control System Using Sensors”, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-3, Issue-3, July 2013

[8] Mohit Dev Srivastava, Prerna, Shubhendu Sachin, Sumedha Sharma, Utkarsh Tyagi, “SMART TRAFFIC CONTROL SYSTEM USING PLC”, International Journal of Innovative Research in Science, Engineering and Technology, Vol. 1, Issue 2, December 2012

[9] Khalil m. Yousef, Jamal n. Al-karakil and Ali m. Shatnawi, “Intelligent Traffic Light Flow Control System Using Wireless Sensors Networks”, journal of information science and engineering 26, (2010)

[10] Vishakha S.Thakare, Snehal R.Jadhav, Sananaj G.Sayed, Poonam V. Pawar, “Design of Smart Traffic Light Controller Using Embedded System”, IOSR Journal of Computer Engineering (IOSR-JCE), Volume 10, Issue 1, (Mar. - Apr. 2013)

[11] Megha A. Tank, Hardik Mewada, Viraj Choksi, M.B. Potdar, “Review on Smart Traffic Control for Emergency Vehicles”, International Journal of Computer Applications (0975 – 8887), Volume 112 – No. 7, February 2015

[12]

https://www.google.co.uk/search?q=arduino+uno+images&espv=2&biw=1366&bih=667&tbm=isch&imgil=Cbj4eEjlm_kj3M%253A%253BfTDyyFfFKmI2M%253Bhttp%25253A%25252F%25252Fwww.intorobotics.com%25252Farduino-uno-setup-and-programming-tutorials%25252F&source=iu&pf=m&fir=Cbj4eEjlm_kj3M%253A%252Cf-TDyyFfFKmI2M%252C_&usg=__FB1BoUkLR-ohkskLjDte_GBVQGY%3D&ved=0ahUKEwiLjrSqgflMAhUJLMAKHfhuA2sQyjcINQ&ei=yepDV8uFDonYgAb43Y3YBg#imgrc=deuPRxaZTIWhsM%3A

[13]

https://www.google.co.uk/search?q=ATMega328+microcontroller+images&espv=2&biw=1366&bih=667&tbm=isch&imgil=GzMynJPXEjKpSM%253A%253BZGPLjDwHNqSkTM%253Bhttps%25253A%25252F%25252Fen.wikipedia.org%25252Fwiki%25252FATmega328&source=iu&pf=m&fir=GzMynJPXEjKpSM%253A%252CZGPLjDwHNqSkTM%252C_&usg=__rc6G0FAwiwMp42v3NFFvTC6tQOw%3D&ved=0ahUKEwimsP-GhPLMAhUkB8AKHdV0D1cQyjcINQ&ei=pO1DV6bOEKSOgAbV6b24BQ#imgrc=GzMynJPXEjKpSM%3A

[14] <https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor/how-pirs-work>

[15]

https://www.google.co.uk/search?q=traffic+lights+running+by+solar+panel&espv=2&biw=1366&bih=667&tbm=isch&tbo=u&source=univ&sa=X&ved=0ahUKEwiX_6WyhfLMAhXIDsAKHYHOAgMQsAQIKw#tbm=isch&q=traffic+lights++and+sensors+running+by+solar+panel&imgsrc=7vm5Xfi2yZpniM%3A