**IOT BASED REMOTE INTELLIGENT TRAFFIC CONTROL SYSTEM AND AIR POLLUTION MONITORING SYSTEM**

**A PROJECT REPORT**

**Submitted by**

**ARUN.A (810015104012)**

**MANIMARAN.G (810015104045)**

***In partial fulfillment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

***in***

**COMPUTER SCIENCE AND ENGINEERING**

****

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**UNIVERSITY COLLEGE OF ENGINEERING-BIT CAMPUS**

**ANNA UNIVERSITY TIRUCHIRAPPLLI**

**ANNA UNIVERSITY::CHENNAI 600 025**

**APRIL 2019**

**UNIVERSITY COLLEGE OF ENGINEERING,**

**BIT CAMPUS,**

**TIRUCHIRAPPALLI-620024**

**BONAFIDE CERTIFICATE**

Certified that this project report titled “**IOT BASED REMOTE INTELLIGENT TRAFFIC CONTROL SYSTEM AND AIR POLLUTION MONITORING SYSTEM”** is a bonafidework of **ARUN A (810015104012) & MANIMARAN G (810015104045)** who carried out the workunder my supervision, for the partial fulfillment of the requirements for the award of the degree of Bachelor of Engineering in computer science Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion.

**SIGNATURE OF HOD SIGNATURE OF GUIDE**

**Mr.D.VENKATESAN Mr.S.ANUVELAVAN**

Head of Department Assistant Professor

Department of CSE-IT Department of CSE

University College of Engineering, University College of Engineering,

BIT Campus, BIT Campus,

Tiruchirappalli -620 024 Tiruchirappalli -620 024

Submitted for the ANNA UNIVERSITY Viva-voce examination held on \_\_\_\_\_\_\_\_\_\_\_

**Internal Examiner** **External Examiner**

**DECLARATION**

I hereby declare that the work entitled “**IOT BASED REMOTE INTELLIGENT TRAFFIC CONTROL SYSTEM AND AIR POLLUTION MONITORING SYSTEM”** is submitted in partial fulfillment of the requirements for the awardof the degree in B.E , in University College of Engineering, BIT Campus, Anna University**,** Tiruchirappalli. It is record of the our own work carried out by us during the academic year 2018-2019 under the supervision and guidance of **Mr. S. Anuvelavan** , Assistant Professor, Department of Computer Science and Engineering, BIT Campus, Anna University, Tiruchirappalli. The extent and source of information are derived from the existing literature and have been indicated through the dissertation at the appropriate places. The matter embodied in this work is original and has not been submitted for the award of any other degree, either in this or any other university.

Signature of the candidates

ARUN A (810015104012)

MANIMARAN G (810015104082)

I certify that the declaration made above by the candidate is true.

Signature of the Guide

**Mr.S.ANUVELAVAN**

Assistant professor, Department of CSE/IT,

UCE-BIT Campus Anna University

Tiruchirappalli- 620 024

**ACKNOWLEDGEMENT**

We would like to thank our honorable Dean **Dr. T. SENTHIL KUMAR**, Professor for having provided us with all required facilities to complete our project without hurdles.

We would also like to express our sincere thanks to

**Mr.D.VENKATESAN**, Head of the Department of Computer Science and Engineering, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of this project work.

We would like to thank our Project Coordinator **SURESH KUMAR**, Teaching Fellow, Department of Computer Science for her kind support.

We would like to thank and express our deep sense of gratitude to our project guide, **Mr. S. ANUVELAVAN,** Assistant Professor, Department of Computer Science and Engineering, Anna University, Chennai, for his valuable guidance throughout the project. We also extend our thanks to all other teaching and non-teaching staff for their encouragement and support.

We thank our beloved parents and friends for their full support in the moral development of this project.

**ABSTRACT**

Nowadays traffic jams and congestion is a common issue because of the day by day increment of numerous vehicles.A Remote IOT device Intelligence traffic control system can be one of the solution to the above problem. To coordinate the traffic by keeping a check of its density from all the sides and there by controlling traffic signal intelligently. A new approach for controlling Traffic System is designed. And this projects includes ambient air pollution monitoring system based on Arduino micro controller using MQ-135 gas sensor for experiment purpose. Air pollution is on the rise due to number of human activities and its monitoring is of vital importance to mitigate certain measures to control it. Air pollution mainly comprises of harmful gases. MQ-135 is a SnO2 semiconductor based gas sensor capable of measuring concentration of gases such as CO, CO2, NH3, Butane. The data collected from the sensor is stored in a database. Which can be used for various process like reducing the pollution by preventing the vehicles in crowded area. A new approach for Traffic and Air pollution monitoring system is designed. These project may be used to notify the locals to take steps beneficial to their environment according to the amount of pollutants present in the area.

Keywords: IOT(Internet Of Things),ambient air monitoring. Intelligent traffic management

**LIST OF FIGURES**

**FIGURE NO NAME OF THE FIGURE PAGE NO**

**01 IOT based proposed system architecture**

**02 Sequence Diagram**

**03 Arduino Uno**

**04 Mq-135 gas sensor**

**05 Image Sensor(OV7670)**

**06 Traffic control system with arduino uno**

**07 Traffic light system**

**08 Connection between MQ-135 sensor with arduino uno**

**09 Interfacing of MQ-135 with Arduino uno**

**10 MQ-135 data calculations**

**11 Levels of various gases in atmosphere**

**12 Air pollution data in Trichy Areas**

**13 Data sheet for BUTANE gas using MQ-135**

**14 Data sheet for AMMONIA gas using MQ-135**

**15 Data sheet for CARBON DI OXIDE using MQ-135**

**16 Real time implementation for traffic control with air pollution monitoring system**

**TABLE OF CONTENTS**

**CHAPTER NO TITLE PAGE NO**

**ABSTRACT**

**LIST OF FIGURES**

**1 INTRODUCTION**

1.1 BACKGROUND AND MOTIVATION

1.2 INTERNET OF THINGS

1.3 ARDUINO UNO

**2 LITERATURE SURVEY**

**3 SYSTEM ANALYSIS**

* 1. EXISTING SYSTEM
  2. PROPOSED SYSTEM

**4 SYSTEM DESIGN**

4.1 SYSTEM ARCHITECTURE

4.2 ARCHITECTURE EXPLNATION

**5 UML DIAGRAMS**

5.1 SEQUENCE DIAGRAM

**6 SYSTEM REQUIREMENTS**

6.1 HARDWARE REQUIREMENTS

6.2 SOFTWARE REQUIREMENTS

**7 SYSTEM IMPLEMENTATION**

7.1 MODULE SPEIFICATONS

7.2 MODULE DESCRIPTION

**8 CONCLUSION**

**FUTURE ENHANCEMENT**

**APPENDIX**

**REFERENCES**

**CHAPTER 1**

**INTRODUCTION**

**1.1 BACKGROUND AND MOTIVATION**

**1.2 INTERNET OF THINGS**

The Internet of things (IoT) refers to the concept of extending Internet connectivity beyond conventional computing platforms such as personal computers and mobile devices, and into any range of traditionally "dumb" or non-internet-enabled physical devices and everyday objects. Embedded with electronics, Internet connectivity, and other forms of hardware (such as sensors), these devices can communicate and interact with others over the Internet, and they can be remotely monitored and controlled. The definition of the Internet of things has evolved due to convergence of multiple technologies, real-time analytics, machine learning, commodity sensors, and embedded systems. Traditional fields of embedded systems, wireless sensor networks, control systems, automation (including home and building automation), and others all contribute to enabling the Internet of things. In the consumer market, IoT technology is most synonymous with products pertaining to the concept of the "smart home", covering devices and appliances (such as lighting fixtures, thermostats, home security systems and cameras, and other home appliances) that support one or more common ecosystems, and can be controlled via devices associated with that ecosystem, such as smartphones and smart speakers. The IoT concept has faced prominent criticism, especially in regards to privacy and security concerns related to these devices and their intention of pervasive presence.

The most important features of IoT on which it works are connectivity, analyzing, integrating, active engagement, and many more. Some of them are listed below:

1) Connectivity: Connectivity refers to establish a proper connection between all the things of IoT to IoT platform it may be server or cloud. After connecting the IoT devices, it needs a high speed messaging between the devices and cloud to enable reliable, secure and bi-directional communication.

2) Analyzing: After connecting all the relevant things, it comes to real-time analyzing the data collected and use them to build effective business intelligence. If we have a good insight into data gathered from all these things, then we call our system has a smart system.

3) Integrating: IoT integrating the various models to improve the user experience as well.

4) Artificial Intelligence: IoT makes things smart and enhances life through the use of data. For example, if we have a coffee machine whose beans have going to end, then the coffee machine itself order the coffee beans of your choice from the retailer.

5) Sensing: The sensor devices used in IoT technologies detect and measure any change in the environment and report on their status. IoT technology brings passive networks to active networks. Without sensors, there could not hold an effective or true IoT environment.

6) Active Engagement: IoT makes the connected technology, product, or services to active engagement between each other.

Internet of things facilitates the several advantages in day-to-day life in the business sector. Some of its benefits are given below:

⦁ Efficient resource utilization: If we know the functionality and the way that how each device work we definitely increase the efficient resource utilization as well as monitor natural resources.

⦁ Minimize human effort: As the devices of IoT interact and communicate with each other and do lot of task for us, then they minimize the human effort.

⦁ Savetime: As it reduces the human effort then it definitely saves out time. Time is the primary factor which can save through IoT platform.

⦁ Enhance Data Collection:

⦁ Improve security: Now, if we have a system that all these things are interconnected then we can make the system more secure and efficient (SoC) which houses a 1.2 GHz Quad Core ARM Cortex-A53 processor.

**1.2 ARDUINO UNO**

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc.[2][3] The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits.[1] The board has 14 Digital pins, 6 Analog pins, and programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable.[4] It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo. [5][6] The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0.[1] The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases.[4] The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform.[3] The ATmega328 on the Arduino Uno comes preprogrammed with a bootloader that allows uploading new code to it without the use of an external hardware programmer.[3] It communicates using the original STK500 protocol.[1] The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

Technical specifications

Microcontroller: Microchip ATmega328P [7]

**Operating Voltage**: 5 Volts

**Input Voltage**: 7 to 20 Volts

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

**Analog Input Pins**: 6

DC Current per I/O Pin: 20 mA

DC Current for 3.3V Pin: 50 mA

Flash Memory: 32 KB of which 0.5 KB used by bootloader

SRAM: 2 KB

EEPROM: 1 KB

Clock Speed: 16 MHz

Length: 68.6 mm

Width: 53.4 mm

Weight: 25 g

General Pin functions

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

**VIN**: The input voltage to the Arduino Uno 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 - 20V), the USB connector (5V), or the VIN pin of the board (7-20V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage the board.

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

**GND**: Ground pins.

**IOREF**: This pin on the Arduino uno board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.

**Reset**: Typically used to add a reset button to shields which block the one on the board.[7]

Special Pin Functions

Each of the 14 digital pins and 6 Analog 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 20 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50k ohm. A maximum of 40mA is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller. 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.[7]

**In addition, some pins have specialized functions:**

**Serial / UART**: pins 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: pins 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.

**PWM (Pulse Width Modulation):** 3, 5, 6, 9, 10, and 11 Can provide 8-bit PWM output with the analogWrite() function.

**SPI (Serial Peripheral Interface):** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

TWI (Two Wire Interface) / I²C: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.

**AREF (Analog Reference)**: Reference voltage for the analog inputs.[7]

**CHAPTER 2**

**LITERATURE SURVEY**

1. Senthil Kumar Janahan1\*-"IOT BASED SMART TRAFFIC SIGNAL MONITORING SYSTEM USING VECHICLE COUNTS"

TMS -Traffic Monitoring Signal timing has been developed by using multiple features of hardware components in IOT. Traffic optimization is achieved using IOT platform for efficient utilizing allocating varying time to all traffic signal according to available .vehicles count in road path. TMS will helpful to client user to know timing arability and traffic flow count in any area of their nearby locality of any regions.

**DISADVANTAGE:**

1**.** Time delay cannot be recovered

2. Awareness of emergency vehicles is not considered.

2. Rashmi P. Nimkar1, Chandrashekhar N. Deshmukh2-"TRAFFIC DENSITY MONITORING AND CATTLE MENANCE ALERT SYSTEM USING IOT"-

The proposed system is used for controlling the traffic, avoiding traffic congestion. Also, the signals are monitored and the status of the traffic signal is updated at the server. Cattle menace is detected. The proposed system is very cost effective as it does not require installation of any additional devices, such as RFIDs, sensors etc. Because this method use Image Processing, Mat lab software and Thing Speak platform, it is feasible to implement it with low cost and with optimum accuracy. Using the proposed method we are only monitoring the number of vehicles present at the signal. This work can be enhanced further to detect the vehicles which disobey the traffic laws. This work can be enhanced further to identify the presence of emergency vehicles (like an ambulance or fire ) giving preference to those emergency vehicles. We can also use this method for garage management systems, for smoother parking of cars and vehicles.

**DISADVANTAGES:**

1. Emergency vehicles are identified but time delay unrecovered.

2. Air pollution cannot be monitored.

3. Prashant Jadhav -"SMART TRAFFIC CONTROL SYSTEM USING IMAGE PROCESSING"-

In this project, a method for estimating the traffic using Image Processing is presented. This is done by using the camera images captured from the highway and videos taken are converted to the image sequences. Each image is processed separately and the number of cars has been counted. If the number of cars exceeds a specific threshold, warning of heavy traffic will be shown automatically. The advantages of this new method include such benefits as use of image processing over sensors, low cost, easy setup and relatively good accuracy and speed. Because this method has been implemented using Image Processing and Mat lab software, production costs are low while achieving high speed and accuracy.

**DISADVANTAGES:**

1. Air pollution cannot be recovered.

2. Using of image processing technique is better solution but still traffic jams cannot be reduced.

4. Harshini Vijetha H-"IOT BASED INTELLIGENT TRAFFIC CONTROL SYSTEM"-

The proposed system results is as we expected. By this proposed system time management for signal lights is done which will reduce the traffic congestion problem. And the system has automatic and manual operation. When lost vehicle is identified a email will be dropped. Tracking of stolen automobile is done successfully and is fast. Clearance of traffic for emergency automobile is successfully implemented. Hence, many precious life would be saved. Further the same prototype can be added with accident message alert. And at present we have implemented the design for only one road of junction. This can be extended to more number of junctions. An 'app' can be designed which uses traffic status at different location from the control station database to display so that it helps normal people.

**DISADVANTAGES:**

1. In this still air pollution cannot be monitored

2. GPS tracking is not explained clearly.

3. Efficient Traffic system cannot be achieved.

5. Paul Jasmine Rani. L1\*, Khoushik Kumar. M2- "DYNAMIC TRAFFIC MANAGEMENT SYSTEM USING INFRARED (IR) AND INTERNET OF THINGS (IoT)"

The existing scheme faces a major demerits of changing the traffic controller in a clock-wise manner, it doesn't make note of the traffic denseness. The denseness of the traffic is calculated and the timer display is shift dynamically. This major advantage rules out the happening of ‘unwanted wait’ for the vehicles in the more crowded region.

**DISADVANTAGES:**

1. Unwanted waits cannot be recovered.

2. Air pollution cannot be monitored.

3. Still it cannot be implemented in real time.

6. Shubham N. Mahalank - "Device to Device Interaction Analysis in IoT based Smart Traffic Management System: An Experimental Approach"

The analysis of the four communication pattern depicts that sufficient investigations are carried out in device to device communication to develop suitable protocols. A closer examination of protocol development and standardization is the need in device to cloud infrastructure. Back end data sharing framework indicates the necessity for a secure communication in IoT applications. It is observed that the design pattern describes how different components of the IoT system interact. The device to device interaction has various data formats which are interesting to note. As a future work, the device to ALG communication protocols development is the need of the hour.

**DISADVANTAGES:**

1. GPS tracking system is not explained.

2. efficient traffic system cannot be achieved.

7. Walid Balid\* and Hazem H - "On The Development of Self-Powered IoT Sensor for Real-Time Traffic Monitoring in Smart Cities "

This work introduces a novel self-powered IoT sensor that can be used in various traffic surveillance applications. The sensor integrates state-of-the-art ultra-low-power electronics, smart sensors, WSN, and EH units to enable fully autonomous operation and indefinite sensor life. The sensor is easy-to install into roadway surfaces without intrusive roadwork and costs only $30 per node. These qualities make the technology a viable solution for a countless number of applications in both short-term (e.g., work zone safety, temporary roadway design studies, traffic management in atypical situations such as evacuation) and long-term deployments (e.g., turn movement, traffic management, automatic garage door and gates, drive their vehicle detector, smart parking lot management).

**DISADVANTAGES:**

1. Air quality cannot be monitored.

2. GPS tracking, emergency vehicles cannot be detected by this model.

**CHAPTER 3**

**SYSYTEM ANALYSIS**

**3.1 EXISTING SYSTEM**

Traffic jams and congestion is a common issue because of the day by day increment of numerous vehicles. Existing system includes both automatic and manual controlling system. In automatic system there is so much of problems there exist unwanted wait, time delay, awareness of emergency vehicles are the major issues and in manual the system is not properly used. To overcome this a remote IOT device Intelligence traffic control system with air pollution monitoring system can be one of the best solution. To coordinate the traffic by keeping a check of its density from all the sides and there by controlling traffic signal intelligently by using Image sensor(0v7670) and air pollution can be monitored using MQ-135 Gas Sensor.

**DISADVANTAGES**

1. Heavy traffic jams.
2. Still there is a time delay can’t be recovered.
3. And still air pollutions can’t be monitored.
4. Still Emergency vehicles problem is not considered.

**3.2 PROPOSED SYSTEM**

A new approach for controlling traffic and air pollution monitoring system is modelled using Arduino Uno micro controller. To control heavy traffic jams we use **Image sensor (0v7670)** to capture the vehicles on all sides and display which side the density of vehicles is more. Based on the density of vehicles respective signals will be provided so that unwanted delay can be restricted. If there is emergency vehicles on any path immediately green signal will be provided. We use **MQ-135 Gas Sensor** is use to detect the pollution and if it is more on any path it will detect and provide the red alert for the vehicles and the detect gas data can be extracted using PLX-DAQ excel sheet and stored in Database. All this approach are upload to a webserver which can be accessed by using a remote device or a PC.

**Advantages**

1. There is no unwanted waits of vehicles.
2. There is no time delay.
3. Traffic jams can be reduced.
4. Pollution can be monitored and controlled.
5. Awareness can be created among human activities based on pollution.

**CHAPTER 4**

**SYSTEM DESIGN**

**4.1 SYSTEM ARCHITECTURE**

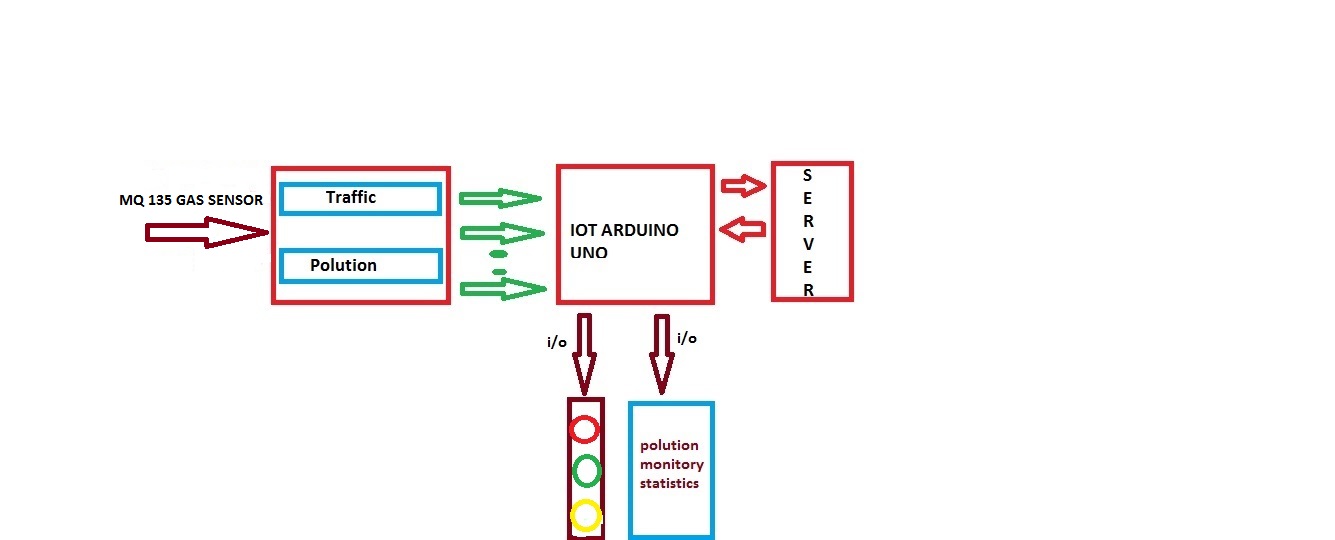
****

FIGURE 01: IOT BASED PROPOSED SYSTEM ARCHITECTURE

**4.2 ARCHITECTURE EXPLANATION:**

The proposed system consist of both Traffic control system and Air pollution monitoring system. These system can be accessed by using an IOT device arduino uno microcontroller. The Traffic System can be access by arduino uno and the program can be upload in Arduino using Arduino IDE software. This Traffic System is consist of image sensor OV7670. If will capture the Image from four side and send the image to the server and with that image density of vehicles can be calculated and priority is given based on the density of vehicles. The image sensor also capture the emergency vehicles and that can be noticed and green signal will be provided to that respective side. And main advantage is time delay and unwanted waits can be completely reduced. In these advanced traffic control system is achieved. Along with the Traffic control system MQ-135 sensor can be attached. The MQ-135 sensor detect the pollution gases on the surrounding atmosphere and alert can be given to public for awareness. The gases that are detected by the sensor are CO2, CO, Ammonia, Butane, Fresh air. MQ-135 sensor can be connected with arduino uno and respective code can be uploaded to that board and output can be extracted extract using PLX-DAQ. The data can be collected and effective gases can be classified based on the voltage. All these data are uploaded in a web server and that can be accessed or control by using pc or other remote devices. This is a project to help and solve the problem of congestion in our modern cities today. The understanding of the design and the implementation of this project will help save lives and make travelling experience wonderful.

**CHAPTER 5**

**UML DIAGRAM**

* 1. **SEQUENCE DIAGRAM**

A sequence diagram simply depicts interaction between objects in sequential order. The order in which these interactions take place. Here we have the sequence diagram .Which includes sequential order of Traffic Control System and Air pollution Monitoring System. Sequence diagram for our proposed system is given below. Here the admin is login to account then validate for accountant details after a message will be received on your system. All updates for every time recorded the information to the traffic control system. The image sensor is fixed on the traffic light control system it will capture the image and converted from grayscale to binary image. The image is convert to binary codes. After this binary codes will going on Morphological process. How much the number of vehicles in specific path is counted and also identifying the object by this process. MQ-135 electrochemical gas sensor is connected to the traffic control system then every time monitoring the pollution should be accuracy of calculated. So pollution overhead that surrounding alert is going to controller of traffic system via server. The sequence diagram is given below.

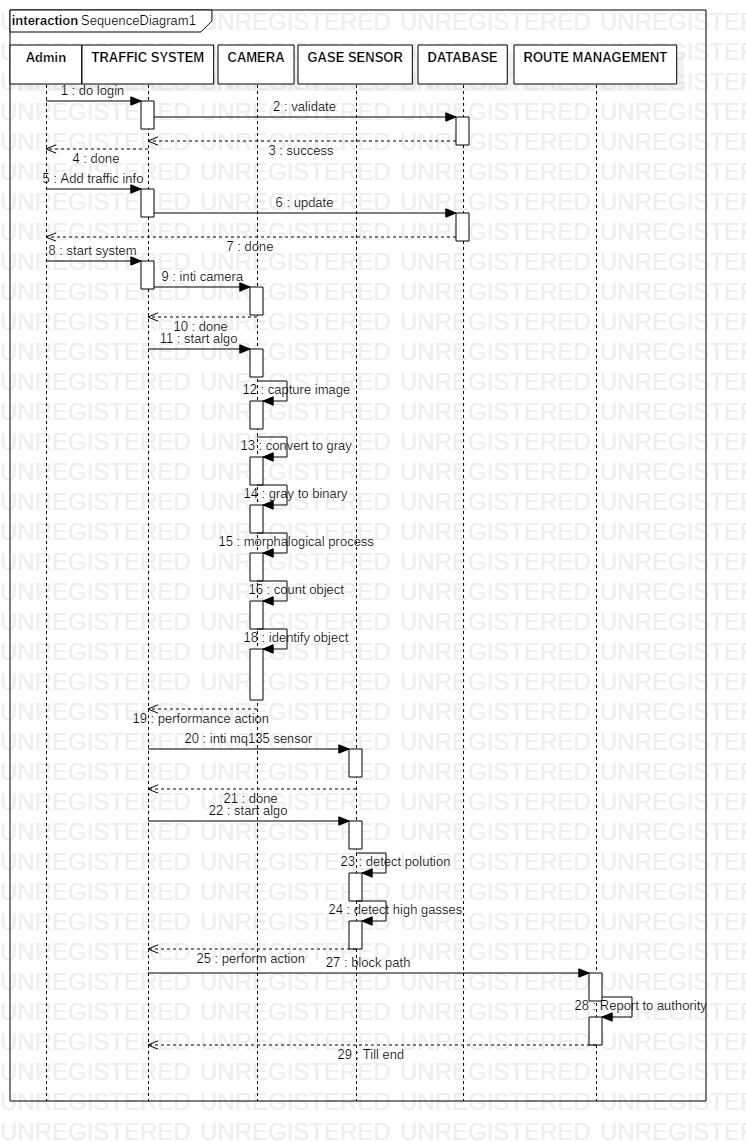
****

FIGURE 02: SEQUENCE DAIGRAM

**CHAPTER 6**

**SYSTEM REQUIREMENTS**

**6.1 HARDWARE REQUIREMENTS**

The hardware requirements may serve as the basis for a contact for the implementation of the system and should therefore be a complete and consistent specification of the whole system. The following hardware used in the system are.

**6.1.1 ARDUINO UNO**

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino Uno board are able to read inputs, light on a sensor, a finger on a button, or a Twitter message and turn it into an output activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the [Arduino programming language](https://www.arduino.cc/en/Reference/HomePage) (based on [Wiring](http://wiring.org.co/)), and [the Arduino Software (IDE)](https://www.arduino.cc/en/Main/Software), based on [Processing](https://processing.org/). It is used to write and upload computer code to the physical board.

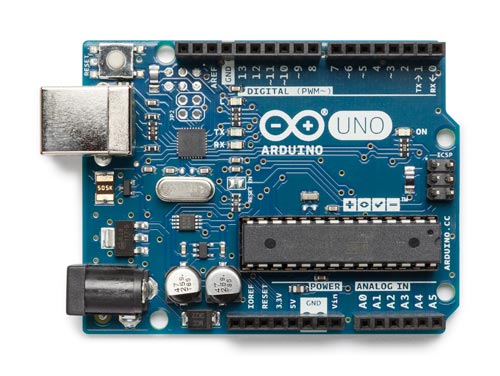


Figure 03: Arduino uno

**6.1.2 MQ-135 GAS SENSOR**

The gas sensor module consists of a steel exoskeleton under which a sensing element is housed. This sensing element is subjected to current through connecting leads. This current is known as heating current through it, the gases coming close to the sensing element get ionized and are absorbed by the sensing element. This changes the resistance of the sensing element which alters the value of the current going out of it.

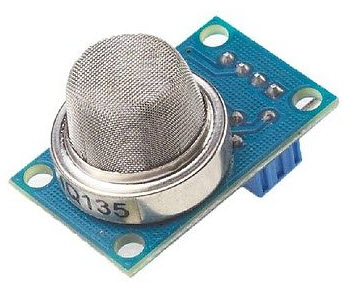


Figure 04: Mq-135 gas sensor

**6.1.3 IMAGE SENSOR(OV7670)**

High sensitivity suitable for illumination applications Low voltage suitable for embedded applications. Standard SCCB interface compatible with I2C interface. 50/60Hz automatic detection.



Figure 05: Image sensor (OV7670)

**6.2 SOFTWARE REQUIREMENTS**

The software requirements provide a basis for creating the software requirements specification.

Operating System - Windows

Software - Arduino IDE,

PLX-DAQ ,

IDLE PYTHON

**CHAPTER 7**

**SYSTEM IMPLEMENTATION**

**7.1 MODULE SPECIFICATIONS**

7.1.1 IOT BASED SMARTTRAFFIC LIGHT CONTROL SYSTEM

7.1.2 IOT OF INTERFACING OF MQ135 SENSOR WITH ARDUINO

7.1.3 AIR POLUTION MONITORING ENHANCED TRAFFIC MANAGEMENT

**7.2 MODULE DESCRIPTIONS:**

**7.2.1 IOT BASED SMARTTRAFFIC LIGHT CONTROL SYSTEM**

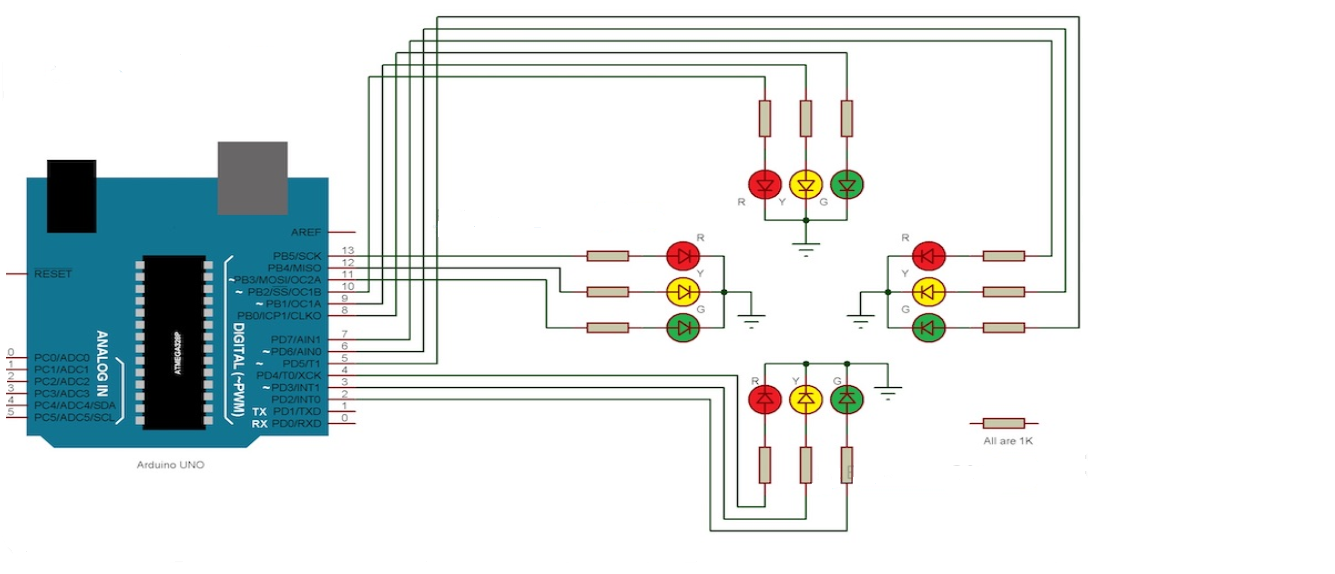
****

FIGURE 06: TRAFFIC LIGHT CONTROL SYSTEM WITH ARDUINO UNO

Since the project is a traffic light controller, the circuit consists of many LEDs (12 as a matter of fact) as we are implementing traffic lights at a 4 way intersection. The project is a simple representation of traffic light controller and hence no other extra components are used. We need three LEDs of Red, Yellow and Green colors at each intersection. The intersection is divided in to four lanes: Lane1, Lane 2 Lane 3 and Lane 4.All the LEDs are connected to the Arduino UNO’s digital I/O pins through respective current limiting resistors of 1KΩ. In the practical implementation of the project, we did not use the current limiting resistors as the current from each digital I/O pin of the Arduino UNO is only 20mA. This small current will not burn the LED. But it is advised to use the current limiting resistors of at least 220 Ω in series with each LED.Also note that Arduino UNO in this project acts as source of current for all the LED i.e. it provides the necessary current to turn ON the LED. Hence, a reliable power supply (like a DC adapter) to power the Arduino UNO must be used.

**WORKING AND EXPLANATION:**

The Automatic traffic light controller is a complex piece of equipment which consists of power cabinet, main controller or processor, relays, control panel with switches or keys, communication ports etc. In this project, a simple traffic light system for a 4 way intersection is implemented using Arduino UNO. Although it is not the ideal implementation for real life scenarios, it gives an idea of the process behind the traffic light control system The aim of the project is to implement a simple traffic light controller using Arduino UNO, where the traffic is controlled in a pre-defined timing system. The working of the project is very simple and is explained below. Consider the following gif image showing a loop of traffic light operations. The project is also implemented in the same manner.

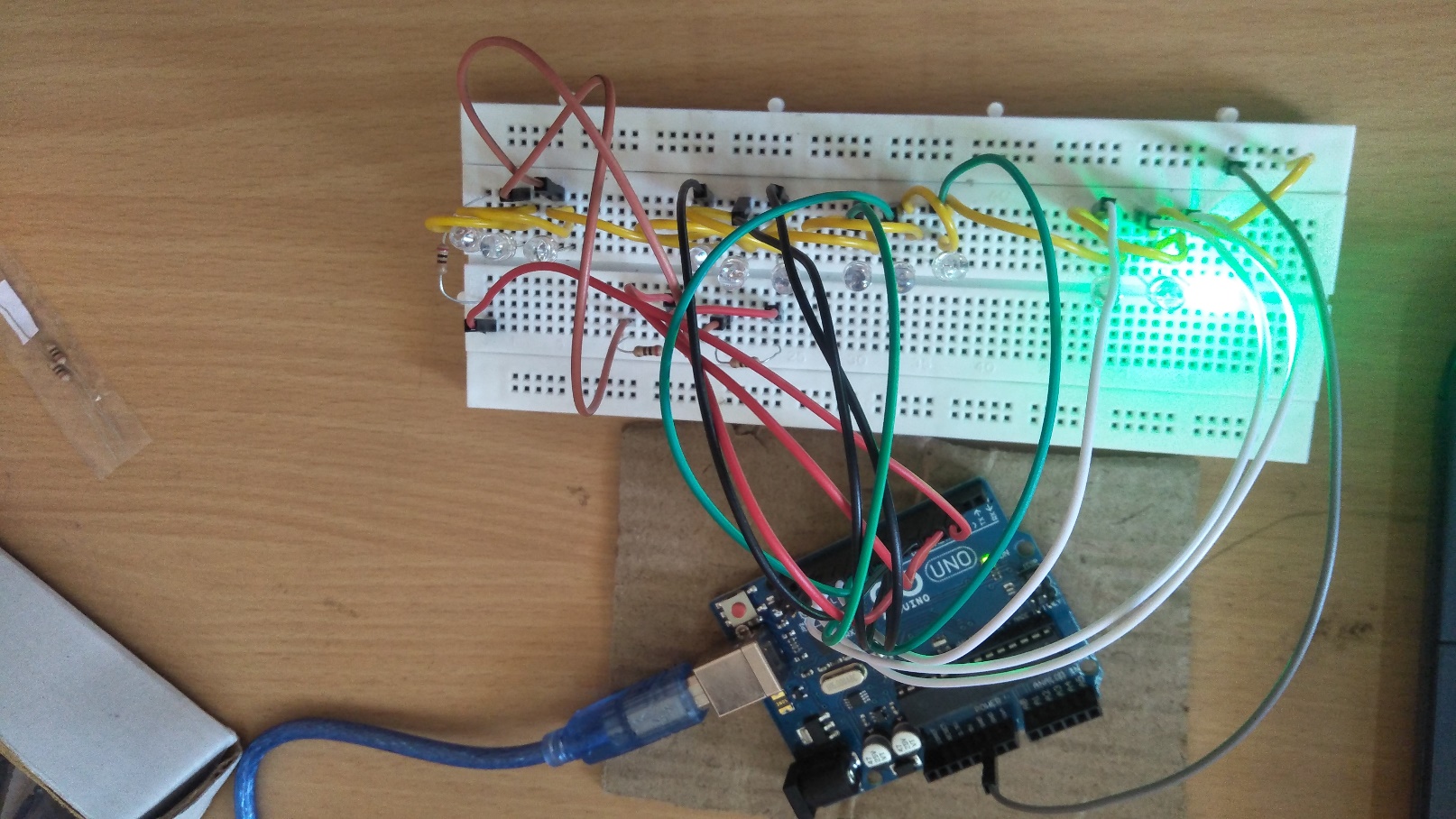


FIGURE O7: TRAFFIC LIGHT SYSTEM

In that, first the Lane 1 gets its Green light turned. Hence, in all the other Lanes, their corresponding Red lights are turned on. After a time delay of predefined time say 5 seconds, the Green light in the Lane 3 must be turned on and the Green light in the Lane 1 must be turned off. As a warning indicator, the Yellow light in Lane 1 is tuned on indicating that the red light is about to light up. Similarly, the yellow light in the Lane 3 is also turned as an indication that the green light about to be turned The yellow lights in Lanes 1 and 3 are turned for a small duration say 2 seconds after with the red light in the Lane 1 is turned on and green light in Lane 3 is also turned on. The green light in Lane 3 is also turned on for a predefined time and the process moves forward to Lane 4 and finally Lane 2.The system then loops back to Lane 1 where the process mentioned above will be repeated all over again.

**LIMITATIONS**

* The project is not suitable for actual implementation but just a demonstration of the process behind the system.
* Real time traffic light controller systems are generally run time programmable i.e. the operator (usually a policeman) can change the timings of each lane as per the intensity of the traffic in each lane.
* There will also be a provision for either manual operation or pre-programmed operation.

**APPLICATIONS**

* A simple traffic light controller is implemented in this project with a real chance of expansion.
* An external memory can be interface with the main controller so that the timings are not fixed during its programming but rather can be programmed during operation.
* An efficient traffic light controller system will include a pedestrian signaling system.

**7.2.2 IOT OF INTERFACING OF MQ135 SENSOR WITH ARDUINO**

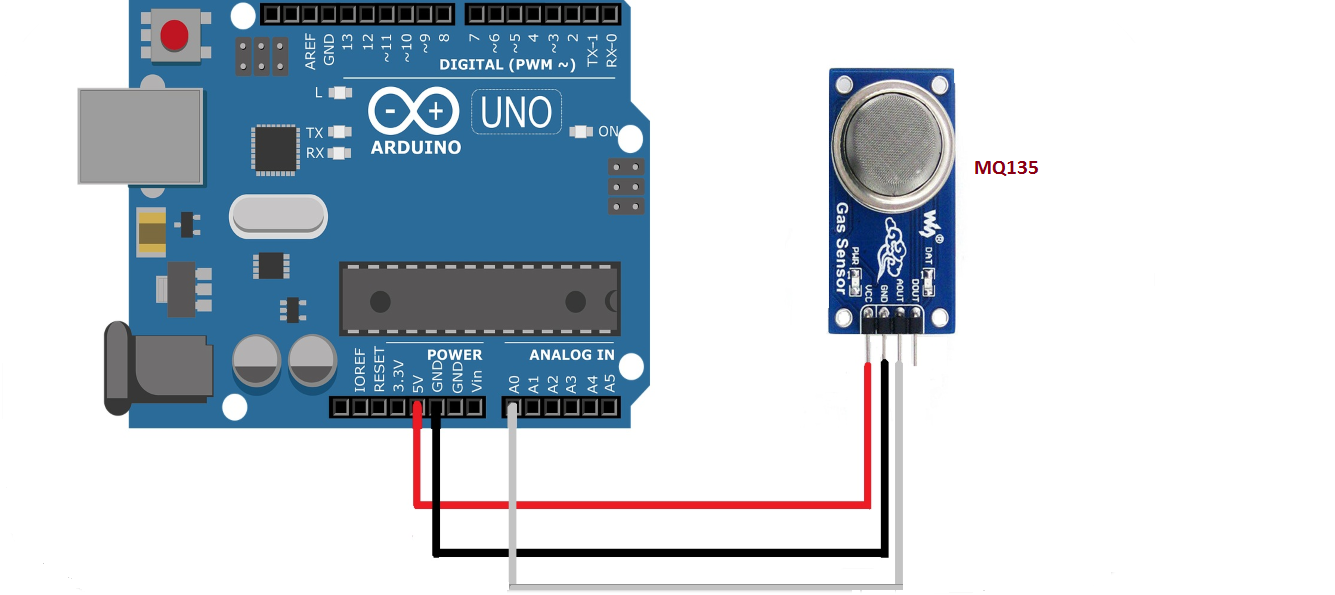
****

FIGURE 08: CONNECTION BETWEEN MQ135 SENSOR WITH ARDUINO UNO

### WORKING AND EXPLANATION:

The MQ135 sensor can sense NH3, BUTENE, CO2, CO, FRESH AIR, and some other gases, so it is perfect gas sensor for our Air Quality Monitoring project. When we will connect it to Arduino then it will sense the gases, and we will get the Pollution level in PPM (parts per million). MQ135 gas sensor gives the output in form of voltage levels and we need to convert it into PPM. So for converting the output in PPM, here we have used a library for MQ135 sensor, it is explained in detail in “Code Explanation” section below. Sensor was giving us value of 90 when there was no gas near it and the safe level of air quality is 350 PPM and it should not exceed 1000 PPM. When it exceeds the limit of 1000 PPM, then it starts cause Headaches, sleepiness and stagnant, stale, stuffy air and if exceeds beyond 2000 PPM then it can cause increased heart rate and many other diseases. When the value will be less than 1000 PPM, then the LCD and webpage will display “Fresh Air”.  Whenever the value will increase 1000 PPM, then the buzzer will start beeping and the LCD and webpage will display “Poor Air, Open Windows.

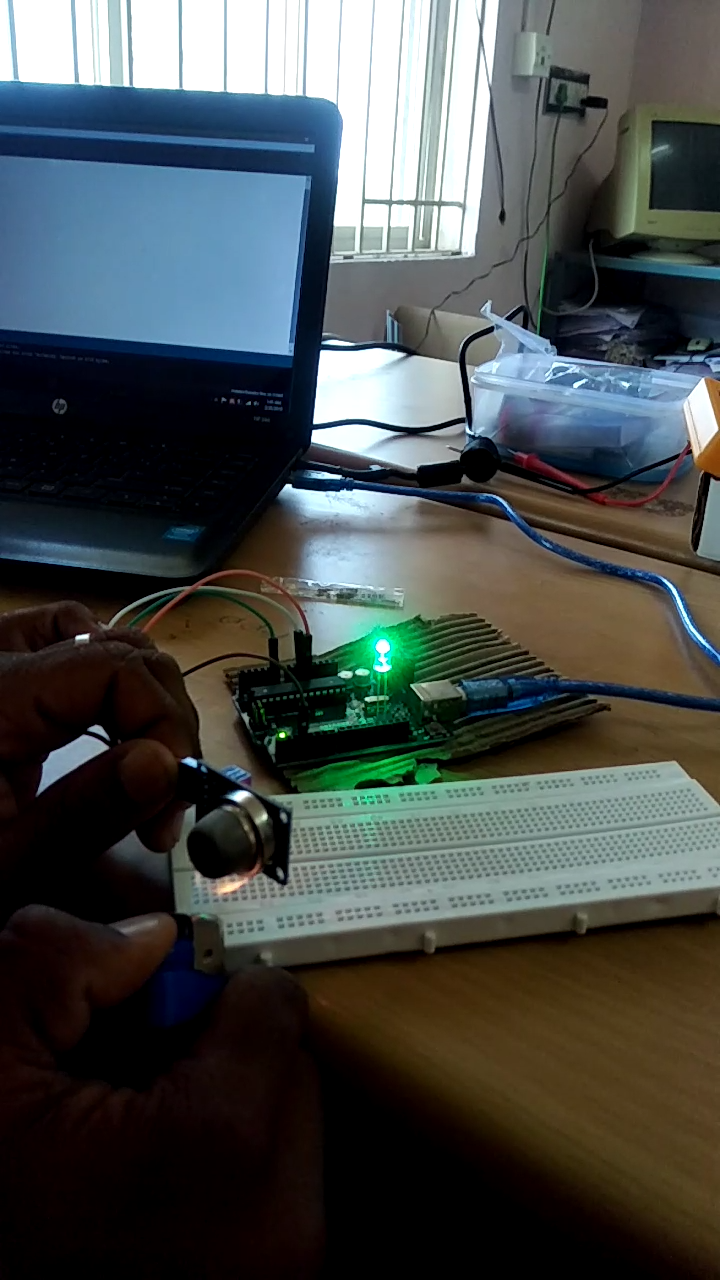


FIGURE 09: INTERFACING OF MQ135 WITH ARDUINO

**7.1.3 AIR POLUTION MONITORING ENHANCED TRAFFIC MANAGEMENT**

An ambient air quality monitoring system based on Arduino microcontroller using gas sensor MQ-135. Air pollution is on the rise due to number of human activities and its monitoring is of vital importance to mitigate certain measures to control it. We have put forward a low-cost and low-power sensor based system to monitor the air quality. Air pollutants mainly comprises of dust particles and harmful gases. MQ-135 is a SnO2 semiconductor based gas sensor capable of measuring concentration of gases such as CO, CO2, NH3 and other inert gases in the ambient air. The data collected from the sensors is stored into a database in a server, which can be accessed through a webpage. These data can be viewed in a tabular form or graphical form. These data can be useful to notify the locals to take steps beneficial to their environment according to the amount of pollutants present in the area. MQ-135 sensor sense various harmful gases, here we collect some gas data sensed by MQ-135 and respective graph can be plotted. The MQ-135 gas sensor provides the system with data which is calculated to find the concentration of gases like CO, CO2, NH3 and Butane. The calculations are performed using Arduino program. The final output is displayed into LCD display mounted on the system. The website contains information about the pollutants measured as well as the collected data in tabular and graphical format. The data is also sorted according to the certain interval of time.

The calculated value are as follows;

AIR (10, 3.58), (200, 3.58)

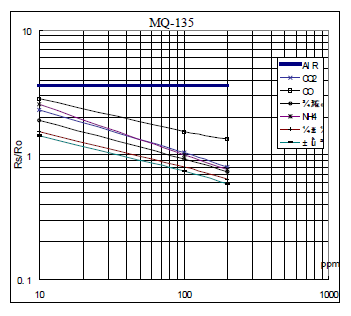
CO (10, 2.8351), (200, 1.3464)

CO2 (10, 2.3), (200, 0.8)

BUTANE (10, 1.90), (200, 0.73)

NH3 (10, 2.55), (200, 0.765)

MQ-135 values are plotted,

**Fresh air (21% )**

**Nitrogen(78%)**

**Carbon di oxide(0.004%) Ammonia(0.0003%)**

Figure 10 : MQ 135 DATA CALCULATIONS

Air pollution monitoring data for various gases in the atmosphere are calculated and ppm values for co2,butane,sulphur,freshair, ammonia values are plotted below.

Figure 11: LEVELS OF VARIOUS GASES IN ATMOSPHERE

**Trichy traffic junction data :**

The air pollution statistics are collected from certain traffic system areas in trichy and pollution data can be graphically drawned.

Figure 12 : AIR POLLUTION DATA IN TRICHY AREA(CO2 GAS)

**EXTRACTION OF DATA FOR VARIOUS GASES USING MQ135:**

MQ 135gas sensor applies has a low conductivity in the clear air as a gas material. In an atmosphere where there may be polluting gas, the conductivity of gas sensor raises along with the concentration of the polluting gas increases. Mq135 performs a good detection to smoke and other harmful gas, especially sensitive to ammonia, co2 and other gases. Here the data for various gas can be extracted through PLX-DAQ and computation time, ppm, volts data are collected. The extracted data sheet for various gas can be given below.

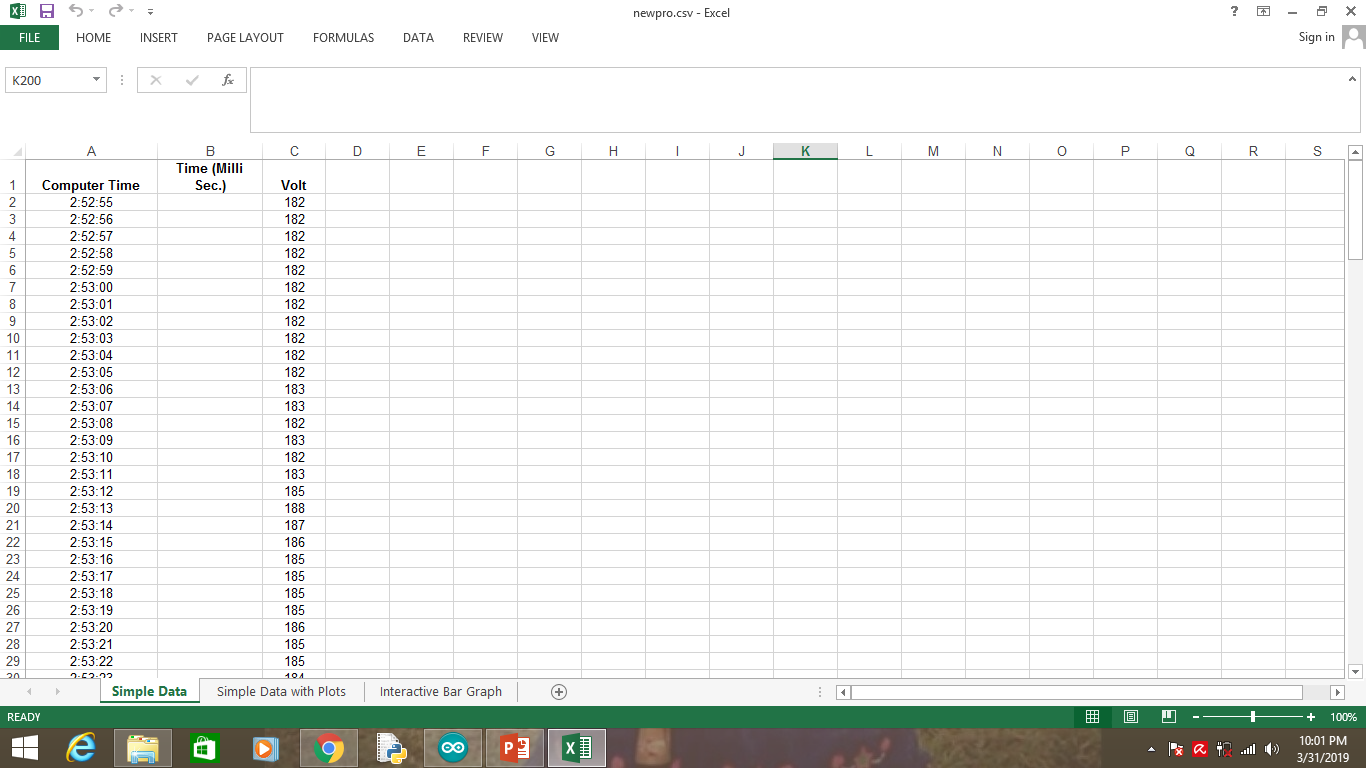
**DATA SHEET FOR BUTENE:**

FIGURE 13: DATA SHEET FOR BUTANE GAS USING MQ135

(COMPUTATION TIME AND VOLT)

**DATA SHEET FOR AMMONIA IN SERIAL MONITOR:**

**AMMONIA RANGE BETWEEN- 17PPM TO 35 PMM ON SHORT TIME EXPOSURE.**

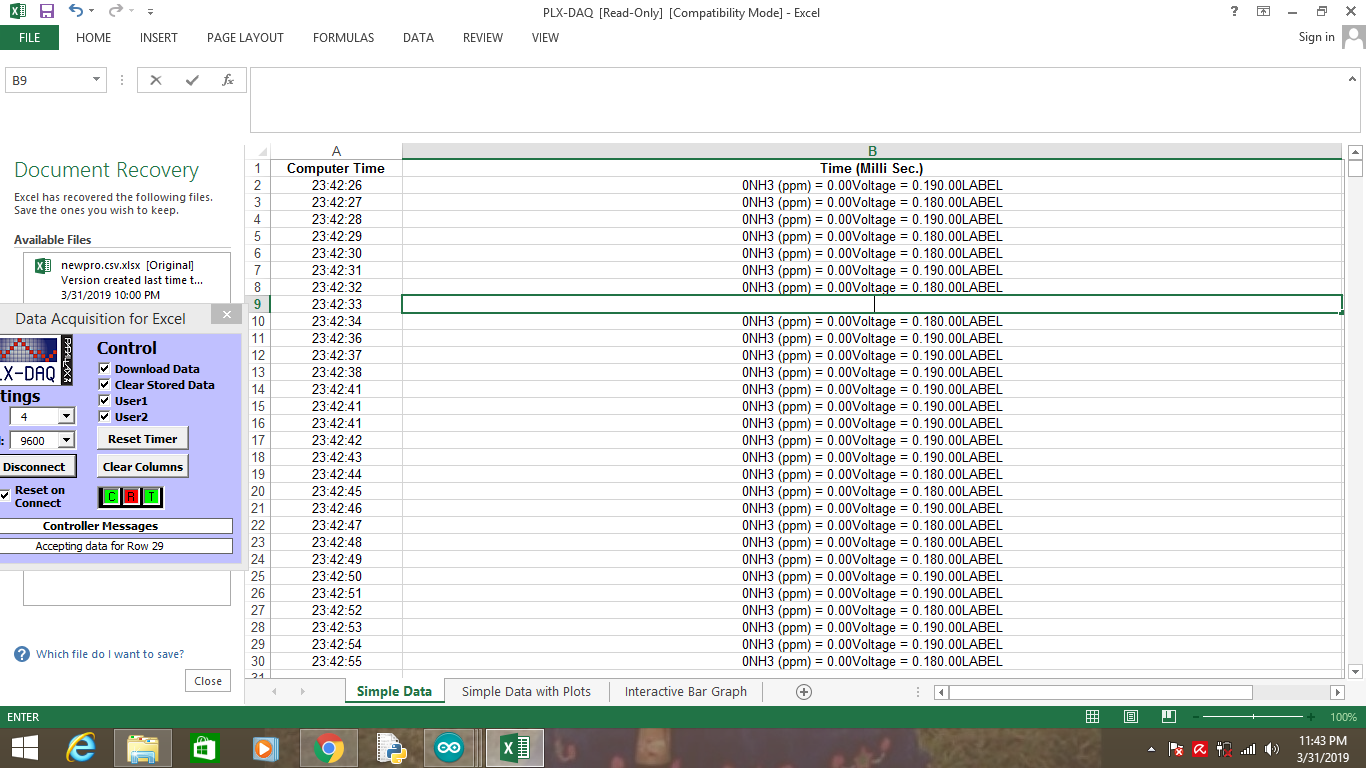
****

FIGURE 14: DATA SHEET FOR AMMONIA USING MQ135

(COMPUTATION TIME AND VOLTS)

**DATA SHEET FOR CO2 IN SERIAL MONITOR:**

**CARBON DI OXIDE RANGE BETWEEN – 300 TO 700K**

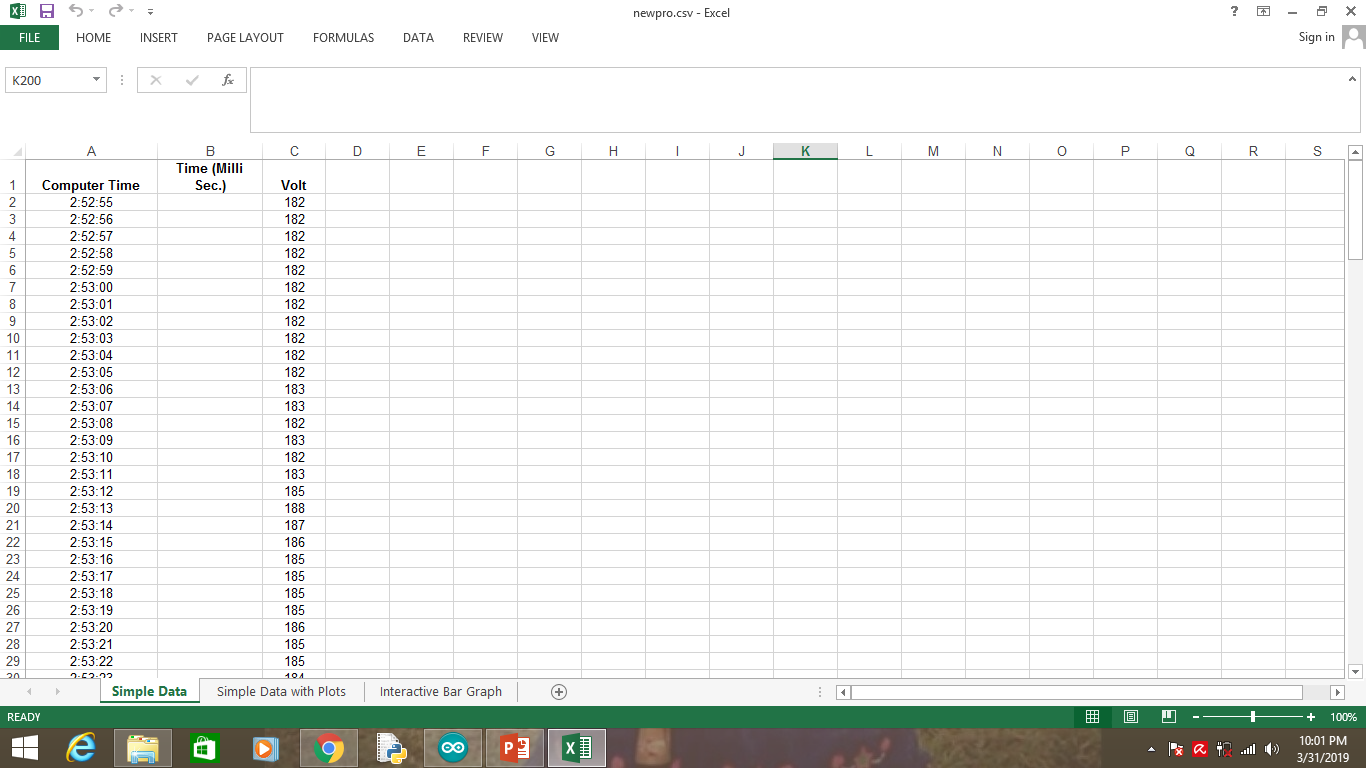
****

FIGURE 15: DATA SHEET FOR CO2 USING MQ135

(COMPUTATION TIME AND VOLTS)

**RESULT AND EXPECTED OUTCOMES:**

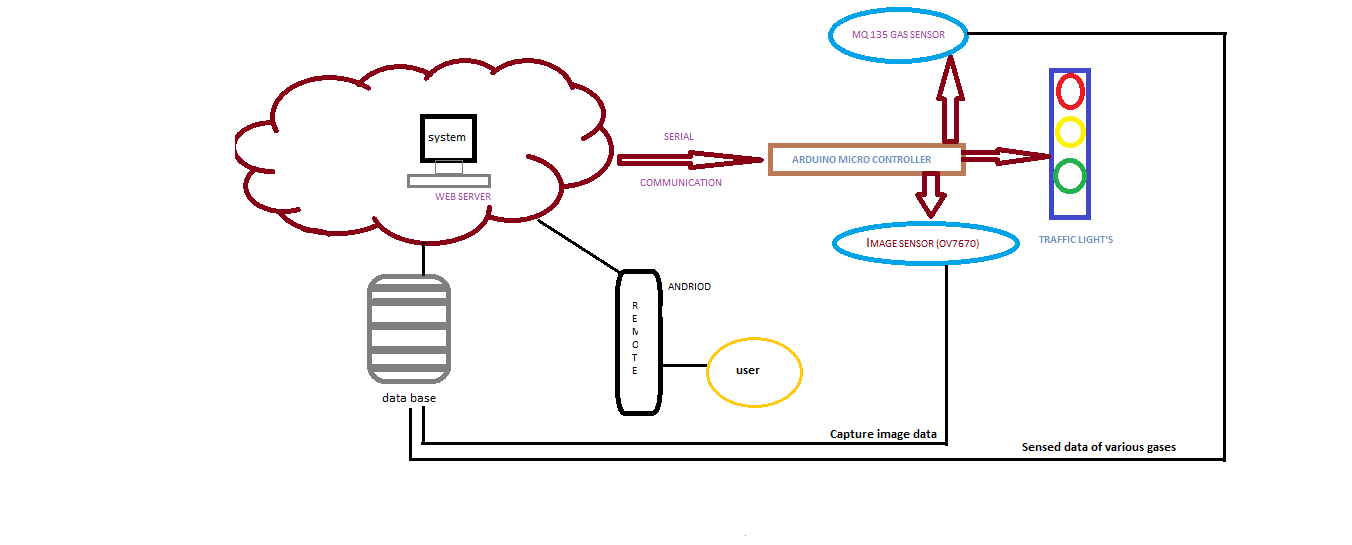
****

FIGURE 15: REAL TIME IMPLEMENTATION OF TRAFFIC CONTROL WITH AIR POLLUTION MONITORING SYSTEM

From the above explained data we have modelled the smart intelligence traffic system with air pollution monitoring system using Arduino Uno micro controller. To control heavy traffic jams we use **Image sensor (0v7670)** to capture the vehicles on all sides and display which side the density of vehicles is more. Based on the density of vehicles respective signals will be provided so that unwanted delay can be restricted. If there is emergency vehicles on any path immediately green signal will be provided. All the collected data and captured images are received and stored as a database and upload to a web server from this necessary priority has been given to lanes and also emergency vehicles awareness can be achieved by using this OV7670 sensor. And to control the air pollution on surrounding traffic areas ,its better to use **MQ-135 Gas Sensor** ,it is use to detect the pollution and if it is more on any path it will detect and provide the red alert for the vehicles and the detect gas data can be extracted using PLX-DAQ excel sheet and stored in Database. All this approach are upload to a webserver which can be accessed by using a remote device or a PC accessed by an user. From this a smart traffic system can be provided for big cities.

**CHAPTER 8**

**CONCLUSION AND FUTURE ENHANCEMENT**

**CONCLUSION**

The Remote intelligent traffic control with air pollution monitoring system has been developed by using multiple future of hardware components in IOT. Traffic control system is achieved using IOT platform for efficient by providing varying time to all traffic signals and priority to emergency vehicles. Air pollution on the traffic areas can be monitored and public awareness can be created with help of MQ 135 sensor. By this in future a better traffic system can be provided to all the countries and provided a pollution free journey for people.

**FUTURE ENHANCEMENT**

In future, along with this proposed system image sensor can be fixed and by this smart traffic control system can be achieved and provide a pollution free environment can be obtained.

**APPENDIX**

**ARDUINO CODE:**

**Traffic control system:**

int Lane1[] = {13,12,11}; // Lane 1 Red, Yellow and Green

int Lane2[] = {10,9,8};// Lane 2 Red, Yellow and Green

int Lane3[] = {7,6,5};// Lane 3 Red, Yellow and Green

int Lane4[] = {4,3,2};// Lane 4 Red, Yellow and Green

void setup()

{

for (int i = 0; i < 3; i++)

{

pinMode(Lane1[i], OUTPUT);

pinMode(Lane2[i], OUTPUT);

pinMode(Lane3[i], OUTPUT);

pinMode(Lane4[i], OUTPUT);

}

for (int i = 0; i < 3; i++)

{

digitalWrite(Lane1[i], LOW);

digitalWrite(Lane2[i], LOW);

digitalWrite(Lane3[i], LOW);

digitalWrite(Lane4[i], LOW);

}

}

void loop()

{

digitalWrite(Lane1[2], HIGH);

digitalWrite(Lane3[0], HIGH);

digitalWrite(Lane4[0], HIGH);

digitalWrite(Lane2[0], HIGH);

delay(7000);

digitalWrite(Lane1[2], LOW);

digitalWrite(Lane3[0], LOW);

digitalWrite(Lane1[1], HIGH);

digitalWrite(Lane3[1], HIGH);

delay(3000);

digitalWrite(Lane1[1], LOW);

digitalWrite(Lane3[1], LOW);

digitalWrite(Lane1[0], HIGH);

digitalWrite(Lane3[2], HIGH);

delay(7000);

digitalWrite(Lane3[2], LOW);

digitalWrite(Lane4[0], LOW);

digitalWrite(Lane3[1], HIGH);

digitalWrite(Lane4[1], HIGH);

delay(3000);

digitalWrite(Lane3[1], LOW);

digitalWrite(Lane4[1], LOW);

digitalWrite(Lane3[0], HIGH);

digitalWrite(Lane4[2], HIGH);

delay(7000);

digitalWrite(Lane4[2], LOW);

digitalWrite(Lane2[0], LOW);

digitalWrite(Lane4[1], HIGH);

digitalWrite(Lane2[1], HIGH);

delay(3000);

digitalWrite(Lane4[1], LOW);

digitalWrite(Lane2[1], LOW);

digitalWrite(Lane4[0], HIGH);

digitalWrite(Lane2[2], HIGH);

delay(7000);

digitalWrite(Lane1[0], LOW);

digitalWrite(Lane2[2], LOW);

digitalWrite(Lane1[1], HIGH);

digitalWrite(Lane2[1], HIGH);

delay(3000);

digitalWrite(Lane2[1], LOW);

digitalWrite(Lane1[1], LOW);

}

**ARDUINO CODE FOR CO2:**

int sensorValue;

int pin8 = 8;

int digitalvalue;

int pin3 = 3;

void setup()

{

Serial.begin(9600);

pinMode(pin8, OUTPUT);

pinMode(pin3, INPUT);

Serial.println("CLEARDATA");

}

void loop()

{

sensorValue = analogRead(0);

Serial.print(sensorValue, DEC);

Serial.print("DATA,TIME,");

Serial.print("");

Serial.print(",");

Serial.println("LABEL,Computer Time,Time (Milli Sec.),");

// prints the value read

Serial.println("ppm");

Serial.println(sensorValue);

if (sensorValue > 500) {

// Activate digital output pin 8 - the LED will light up

digitalWrite(pin8, HIGH);

Serial.println("LABEL,Computer Time,Time (Milli Sec.),");

}

else {

// Deactivate digital output pin 8 - the LED will not light up

digitalWrite(pin8, LOW);

}

delay(1000);

}

**ARDUINO CODE FOR AMMONIA**

#define RL 47 //The value of resistor RL is 47K

#define m -0.263 //Enter calculated Slope

#define b 0.42 //Enter calculated intercept

#define Ro 20 //Enter found Ro value

#define MQ\_sensor A0

//Sensor is connected to A4

#include <LiquidCrystal.h>

const int rs = 8, en = 9, d4 = 10, d5 = 11, d6 = 12, d7 = 13; //Pins to which LCD is connected

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

int sensorValue;

int digitalValue;

void setup()

{

{

Serial.begin(9600);

Serial.println("CLEARDATA");

}

lcd.begin(16, 2); //We are using a 16\*2 LCD display

lcd.print("NH3 in PPM"); //Display a intro message

lcd.setCursor(0, 1); // set the cursor to column 0, line 1

lcd.print("-CircuitDigest"); //Display a intro message

delay(1000); //Wait for display to show info

lcd.clear(); //Then clean it

}

void loop()

{

Serial.println("LABEL,Computer Time,Time (Milli Sec.),Volt ,VRL,R0");

float analog\_value;

Serial.print("DATA,TIME,");

Serial.print(sensorValue);

float VRL; //Voltage drop across the MQ sensor

float Rs; //Sensor resistance at gas concentration

float ratio; //Define variable for ratio

VRL = analogRead(MQ\_sensor)\*(5.0/1023.0); //Measure the voltage drop and convert to 0-5V

Rs = ((5.0\*RL)/VRL)-RL; //Use formula to get Rs value

ratio = Rs/Ro; // find ratio Rs/Ro

float ppm = pow(10, ((log10(ratio)-b)/m));

Serial.print("NH3 (ppm) = "); //Display a ammonia in ppm

Serial.print(ppm);

lcd.setCursor(0, 1); // set the cursor to column 0, line 1

Serial.print("Voltage = "); //Display a intro message

Serial.print(VRL);

Serial.print(ppm);

delay(1000);

lcd.clear(); //Then clean it

}

**ARDUINO CODE FOR BUTENE:**

void setup()

{

Serial.begin(9600); // sets the serial port to 9600

pinMode(13, OUTPUT);

pinMode( 3, INPUT);

Serial.println("CLEARDATA");

Serial.println("LABEL,Computer Time,Time (Milli Sec.),Volt ,ppm");

}

void loop()

{

sensorValue = analogRead(0); // read analog input pin 0

Serial.print("DATA,TIME,");

Serial.print("");

Serial.print(",");

Serial.println(sensorValue);

digitalValue = digitalRead(2);

if(sensorValue>400)

{

digitalWrite(13, HIGH);

}

else

digitalWrite(13, LOW);

Serial.println(sensorValue, DEC); // prints the value read

Serial.println(digitalValue, DEC);

delay(1000); // wait 100ms for next reading

}

**REFERENCES:**

[1] Li Z, Shahidehpour M, Bahramirad S & Khodaei A, “Optimizing traffic signal settings in smart cities”, IEEE Transactions on Smart Grid, Vol.8, No.5, (2017), pp.2382-2393.

[2] Singh L, Tripathi S & Arora H, “Time optimization for traffic signal control using genetic algorithm”, International Journal of Recent Trends in Engineering, Vol.2, No.2, (2009),pp.4-6.

[3] Pable SN, Welekar A & Gaikwad-Patil T, “Implementation on Priority Based Signal Management in Traffic System”.

[4] Keertikumar B Malagund, Shubham N Mahalank, R.M.Banakar,” IoT based smart city traffic alert system design”

[5] Koushik Mandal, Arindam Sen, Abhijnan Chakraborty and Siuli Roy,” Road Traffic Congestion Monitoring and Measurement using Active RFID and GSM Technology”, 2011 14th International IEEE Conference on Intelligent Transportation Systems Washington, DC, USA. October 5-7, 2011 .

[6] F. Wu, C. Rüdiger, and M. Yuce, “Real-Time Performance of a SelfPowered Environmental IoT Sensor Network System,” Sensors, vol. 17, no. 2, p. 282, Feb. 2017.

[7] W. Balid, “Fully-Autonomous Self-Powered Intelligent Wireless Sensor for Real-Time Traffic Surveillance in Smart Cities,” Ph.D. dissertation, Electrical and Computer Engineering Dept., University of Oklahoma, 2016, https://shareok.org/handle/11244/51828.

[8] Xi Yu, Fuquan Sun, Xu Cheng, et.al.,” Intelligent Urba n Traffic Management System Based on Cloud Computing and Internet of Things”, Dalian High-tech Zone Innovation of Science and Technology Plan: 20113006.

[9]Chen Hui,WangXianghui, Zhang Xiqiang, ZhangShaol et.al., ,“TheevaluationofChineseurbantrafficmanagementSystemApplicati onBasedo intelligent traffic control technology, 2014 7thInternational Conference on Intelligent Computation Technology and Automation 978147996636-3/14 $31.00 © 2014 IEEEDOI 10.1109/ICICTA.2014.191791.

[10] Y. Peng and J. M. Chang, “A novel mobility management scheme for integration of vehicular ad hoc networks and ?xed IP networks,” Mobile Netw. Appl., vol. 15, no. 1, pp. 112–125, 2010.

[11] S. Rashwand, J. Miši´c, and H. Khazaei, “IEEE 802.15.6 under saturation: Some problems to be expected,” J. Commun. Netw., vol. 13, no. 2, pp. 142–148, Apr. 2011.

[12] „Traffic Light Control System Using Image Processing?, International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 5, October 2014.

[13] Ms.PallaviChoudkalet. al„Real time traffic light control using imageprocessing?, Indian journal of Computer Science and Engineering(IJCSE) ISSN:0976-5166, Vol 2, No.1,pp 6-10 .