**PUNE VIDYARTHI GRIHA's**

**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**AND G K PATE (WANI) INSTITUTE OF MANAGEMENT,**

**PUNE – 411 009**

**PROJECT REPORT**

**ON**

# *Real Time Traffic Signal Control*

**SUBMITTED BY**

***Rajwardhan Shivraj Ghorpade (B190072520)***

***Lalit Suresh Chaudhari (B190072513)***

***Saurav Pandharinath Palve (B19007251)***

***Abhishek Surendra Sharma (B190072567)***

**Class**

**BE (ELECTRICAL)**

**Year**

*2022-23*

**GUIDED BY**

***Dr. M. P. Bhawalkar***

**DEPARTMENT OF ELECTRICAL ENGINEERING**

**PVG's COET & GKPIM, PUNE-9**

**SAVITRIBAI PHULE PUNE UNIVERSITY,**

**PUNE VIDYARTHI GRIHA's**

**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**AND G K PATE (WANI) INSTITUTE OF MANAGEMENT,**

**PUNE – 411 009**

# CERTIFICATE

This is to certify that the following students, studying in B. E. (Electrical) and having respective exam numbers, have satisfactorily completed the work for their project under my guidance, in the following topic:

## *Real Time Traffic Signal Control*

The report is submitted as fulfillment of the requirement of the Under Graduate degree course in Electrical Engineering, Savitribai Phule Pune University, during the academic year 2022-23.

1. ***Rajwardhan Shivraj Ghorpade (B190072520)***
2. ***Lalit Suresh Chaudhari (B190072513)***
3. ***Saurav Pandharinath Palve (B19007251)***
4. ***Abhishek Surendra Sharma (B190072567)***

**Dr. M. P. Bhawalkar**
**Assistant Professor**
**Dept. of Electrical Engg.**
**PVG's COET & GKPIM,**
**Pune-9**

**Dr. M. S. Thakare**
**Head of Department**
**Dept. of Electrical Engg.**
**PVG's COET & GKPIM,**
**Pune-9**

# ACKNOWLEDGEMENT

We would like to express our gratitude and appreciation to everyone who helped us while working on the project. Special thanks are due to our project guide **Dr. M. P. Bhawalkar** for his stimulating suggestions and encouragement.

Any words of dictum are not enough for the contribution of our head of department **Dr. M. S. Thakare** has been constant support and inspired us to work on new and emerging technologies.

We have taken efforts in this project. However, it would have been not possible without the kind support and help of many individuals and websites. I would like to extend my sincere thanks to all of them. Our thanks and appreciation also go to our college for letting us work on this project.

# ABSTRACT

Heavy and constant traffic jams is one the many and major problems faced by urban cities. Cities like Pune, Mumbai and Bengaluru are infamously known for their traffic problem. Traffic is a random function. It changes time to time. It is different in morning, noon, evening and night. It follows completely different pattern on weekdays and weekends. It can be very random on national holidays. It can vary due to any reasons. Traffic patterns cannot be continuously predicted. Even though our traffic is random, our traffic management system, signals and traffic police, are fixed and limited. No matter what the condition, the timers of our signals remain same. Commuters have to deal with the same signal throughout the year.

This project intends to solve the above problem. The intention is to convert the existing static traffic system into a dynamic and real time operation. To simply explain the project, controller will count the number of vehicles on street and adjust the timer of signals accordingly. The traffic signal will have different timing adjustments for every possible situation. This project will dynamically adjust of signal timers and ensure a smooth and quick movement of vehicles.

The primary objective is to reduce redundancy in day-to-day life. Office workers can reach home early and spend quality time with their loved ones, children can reach home early from school and invest their time in co-curricular activities, college students can stop blaming traffic for being late for lectures, etc. Possibilities of what people can do with their free time are endless.

# INDEX


| Chapter | Sub-chapter | Content | | Page No. |
|---|---|---|---|---|
| 1 | | Introduction | - | 1 |
| | 1 | Problem statement | - | 1 |
| | 2 | Objective | - | 1 |
| | 3 | Impact | - | 2 |
| 2 | | Literature Survey | - | 3 |
| 3 | | Methodology | - | 4 |
| | 1 | Working principle | - | 4 |
| | 2 | Working | - | 4 |
| | | 1. Logic to count vehicles | - | 4 |
| | | 2. Logic to determine timer value (ToD and Difference based timer Adjustment) | - | 6 |
| | | 3. Design a user-oriented product | - | 8 |
| 4 | | Project prototype | - | 9 |
| | 1 | Components | - | 10 |
| 5 | | Project competition participation summary | - | 14 |
| 6 | | Future scope and additional features | - | 15 |
| 7 | | Conclusion | - | 16 |
| 8 | | References | - | 17 |
| 9 | | Annexure | - | 18 |
| | 1 A | Project code | - | 18 |

| | | | |
|---|---|---|---|
| 2 B | MMCOE Project Competition Certificates | - | 23 |
| 3 C | PICT Project Competition Certificates | - | 25 |
| 4 D | JSPM Project Competition Certificates | - | 27 |
| 5 E | Plagiarism Report | - | 29 |

# DIAGRAM INDEX


| Chapter | Sub-chapter | Dig No. | Content | | Pg no. |
|---|---|---|---|---|---|
| 3 | 2.0 | 1 | Block diagram of project | - | 4 |
| 3 | 3.1 | 2 | Counting number of vehicles on street | - | 5 |
| 3 | 3.2 | 3 | T junction | - | 7 |
| 4 | | 4 | Project prototype | - | 9 |
| 4 | 1.0 | 5 | Standard Firmata code in Arduino IDE | - | 12 |

# CHAPTER 1: INTRODUCTION

## 1.1 Problem Statement [9]:

Traffic on roads is a highly dynamic function that experiences fluctuations throughout the day. Understanding the patterns and dynamics of traffic is crucial for effective transportation planning and management. On a typical workday, traffic tends to peak around 9 am when office hours commence. This surge in traffic is primarily due to the influx of commuters heading to their workplaces. As the day progresses, the traffic volume remains at a medium level, with variations depending on factors such as specific locations and local conditions.

In the evening, as office hours conclude, traffic experiences another peak. This surge occurs as individuals are now free to travel, running errands, or returning home after work. The traffic flow during this time can be particularly heavy, leading to congestion in various areas.

However, during weekends and other holidays, the traffic pattern deviates significantly. With fewer people commuting for work, the traffic dynamics change. Weekends often witness increased recreational travel, with people visiting shopping centres, tourist destinations, or engaging in leisure activities. As a result, traffic patterns on weekends can be less predictable and vary significantly from those observed on workdays.

Even though traffic is dynamic real time function, our existing traffic management system, signals & traffic control police, are fixed and limited. No matter what the condition, the timers of traffic signals remain same. In severe traffic conditions, traffic police are deployed but they are not enough. It is impossible to deploy traffic police at each place where traffic congestion is formed.

This project intends to make traffic management a real time and dynamic operation. It will create a traffic signal system such that it will adjust according to the need of the real time traffic. In cases of high traffic, it will automatically change its timer to a value which is higher than the usual timer of a particular signal. In cases of low traffic, it will change to a different pattern. This project will bring a change and give a new direction as to how current traffic is managed.

## 1.2 Objective of the project is as follows:

**Improve existing traffic management system and thus improve the quality of human life.**

The objective of our project is to enhance the existing traffic management system with the aim of improving the quality of human life. By implementing innovative solutions and strategies, the project aims to alleviate traffic congestion, reduce travel times, and enhance overall transportation efficiency. By addressing the challenges associated with traffic management, such as improving traffic flow, minimizing congestion, and enhancing road safety, the project seeks to create a more sustainable and livable environment for individuals, ultimately enhancing their quality of life.

## 1.3 Possible impact of Real Time Traffic Signal Control:

In a world where all traffic signals operate based on real-time data, transportation and traffic management would undergo a transformative change. The utilization of real-time data would revolutionize traffic flow, safety, planning, and the overall quality of life for individuals.

By integrating real-time data into traffic signal operations, the flow of vehicles would become more dynamic and responsive. Traffic signals would adjust their timing and phasing based on current traffic conditions, resulting in optimized traffic flow and reduced congestion. This would lead to shorter travel times, smoother journeys, and improved overall transportation efficiency.

The adaptive nature of signal control would ensure that signals continuously adapt to changing traffic patterns. Prioritization of high-volume routes and dynamic allocation of green signal time based on real-time demand would lead to more streamlined traffic and reduced delays. Additionally, the ability to prioritize pedestrian crossings, emergency vehicle routes, and school zones in real time would enhance safety for all road users.

Real-time data-driven traffic signal operation would also provide valuable insights for traffic planning and management. Transportation authorities could analyse real-time data to identify peak hours, congestion hotspots, and other traffic patterns. This information would enable the development of more effective traffic management strategies, targeted interventions to alleviate bottlenecks, and optimized road infrastructure planning.

The environmental impact of real-time data-based signal operation would be significant. The reduction in congestion and smoother traffic flow would result in lower fuel consumption, decreased emissions, and improved air quality. This would contribute to creating more sustainable and environmentally friendly cities.

Overall, a world where all traffic signals operate based on real-time data would bring about a more efficient, safer, and sustainable transportation system. Commuters would experience improved travel times, reduced congestion, and enhanced predictability, while cities would benefit from better traffic management, optimized infrastructure, and a positive environmental impact.

# CHAPTER 2: LITERATURE SURVEY

The modern electric traffic light is an American invention. As early as1912 in Salt Lake City, Utah, policeman Lester Wire invented the first red-green electric traffic light. The first automatic experimental traffic lights in England were deployed in Wolverhampton in 1927. From the past decades, management of traffic has been one of the biggest issues of modernization. Researchers have followed a long way to overcome the traffic crises.

K. Vidhya [1] proposed a density-based traffic signal system which changes the signal timings automatically by sensing the traffic density at the junctions. She analyzed the image sequence and then estimate traffic congestion and finally predict the traffic light timings. Raspberry pi microcontroller is used to sense the traffic density and provides the signal timings.

A Ms. Pramila Sinhar [2] proposed an intelligent traffic light and density control using IR sensor and microcontroller which optimizes the traffic light control using microcontroller. When a vehicle is passed on a road between IR transmitter and IR receiver, the IR system automatically gets activated and counts the number of vehicles present and store in its memory. Based on the vehicles count the microcontroller takes decision for the traffic signal timings.

Khalil M. Yousef and et al [3] proposed an intelligent traffic light flow control system using wireless sensors networks. An adaptive traffic control system using wireless sensor networks are dynamically used for both single and multiple intersection to control the traffic conditions. Their future work can simulate the human behaviors and package the entire system using FPGA technology and also, different types of intersection and different types of crossing directions also considered.

A. D. Jadhav, et al. [4] proposed an Intelligent Traffic Light Control System which aims at reducing the delay on roads by reducing the amount of traffic. The proposed system tries to minimize the possibilities of traffic jams to some extent by clearing the road with higher density of vehicles. The road which is recorded with more traffic density then other roads will be assigned with a green signal and all other roads will be assigned with red.

Article in Times of India ‘PCMC’s Rs 9 crore traffic control system is on the wrong roads.’[10]

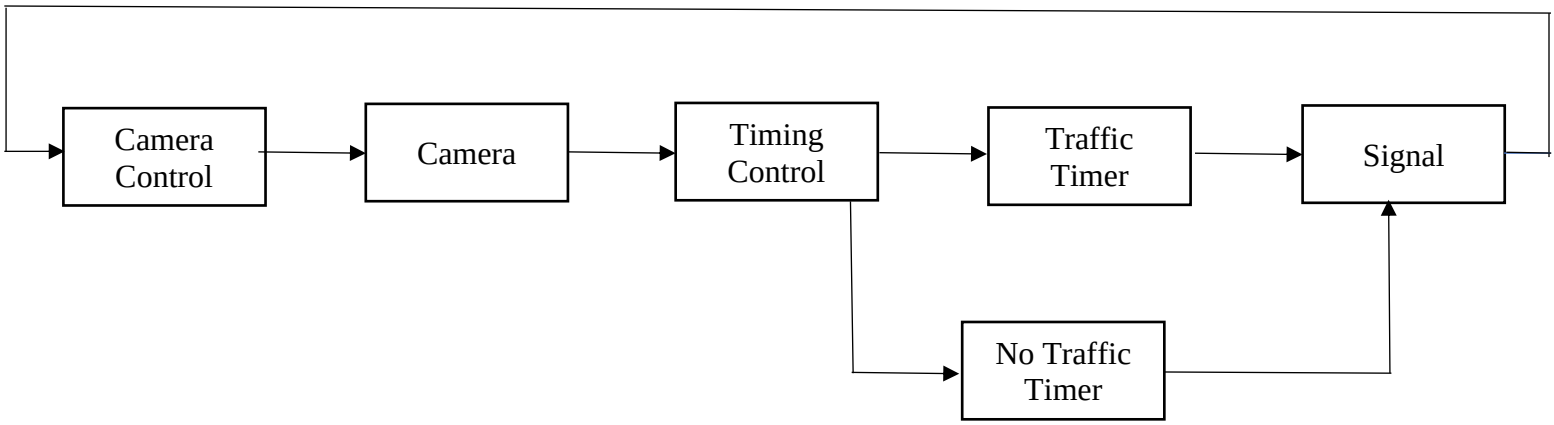
# CHAPTER 3: METHODOLOGY

## 3.1 Working principle:

A camera placed at the traffic junction captures images of vehicles on a particular side of the road. Processor counts the number of vehicles in the image using OpenCV libraries and Machine Learning algorithms. Based on the count of vehicles, timer for green signal will be either increased or decreased.

Based on the type of junction, multiple cameras, timers, and signals will work synchronism with each other to obtain an optimized and continuous flow of vehicles.

## 3.2 Block diagram of project:




Dig 1: Block diagram of Real Time Traffic Control

## 3.3 Working:

The project work is divided into 3 sections as follows:

1. Design a logic to count vehicles visible in the image.
2. Design a logic to select a suitable timer for green signal based on count of vehicles.
3. Designing a user-oriented code to make vehicle counting logic and timer selection logic to work in synchronism.

### 3.3.1 Design a logic to count vehicles visible in the image:

The image capturing process, as well as the storage and vehicle counting tasks, are controlled by a central controller. To establish a connection between the camera and the controller, we offer two options: wireless and wired connections. In the wireless setup, the user can input the camera's IP address into the controller. On the other hand, for a wired connection, the user can input the camera's port number into the controller.

For the crucial task of vehicle detection and counting, we employ the YOLOV4 model, which has shown impressive performance in object detection tasks. YOLOV4 stands for "You Only Look Once" and is a state-of-the-art real-time object detection algorithm. It is known for its speed and accuracy in detecting various objects, including vehicles. The model has been trained on the COCO (Common Objects in Context) dataset, which is a widely used benchmark for object detection.

The COCO dataset contains a vast collection of images with 80 different object categories, including vehicles. By training YOLOV4 on this dataset, the model has learned to recognize and classify vehicles accurately. Leveraging the model's capabilities, our system can effectively detect and count vehicles present in the captured image.

Using YOLOV4 and the COCO dataset, our project provides real-time vehicle counting, enabling traffic authorities to gather valuable data on traffic flow and congestion patterns. This information can aid in optimizing traffic signal timings and implementing efficient traffic management strategies.

By combining advanced deep learning techniques, such as YOLOV4, and leveraging the rich and diverse COCO dataset, our real-time traffic signal control project offers an effective solution for vehicle detection and counting. The system's ability to accurately monitor vehicle movements can contribute to improved traffic management, enhanced safety, and more efficient transportation infrastructure.


Dig 2: Output of program counting number of vehicles on street.

### 3.3.2 Design a logic to select a suitable timer for green signal based on count of vehicles:

A combination of two different logics is used to select a suitable timer. They are as follows:

3.3.2.1 Time of Day Control / ToD Control
3.3.2.2 Difference based timer adjustment

#### 3.3.2.1 ToD Control:

In our project, we address the dynamic nature of traffic by allowing users to input base timers for different hours of the day. This feature enables the system to adapt the duration of green signals based on the specific time periods. For instance, during the early morning hours from 6:00 am to 9:00 am when traffic is generally low, the user can set a base timer of 15 seconds for the green signal. However, during peak traffic periods from 9:00 am to 1:00 pm, which are typically office starting hours, the user can set a longer base timer of 30 seconds.

By inputting different timer values for different hours of the day during the installation phase, users can establish customized signal timing patterns that align with the traffic flow patterns at specific times. This approach allows for more efficient traffic management and reduces congestion during peak hours.

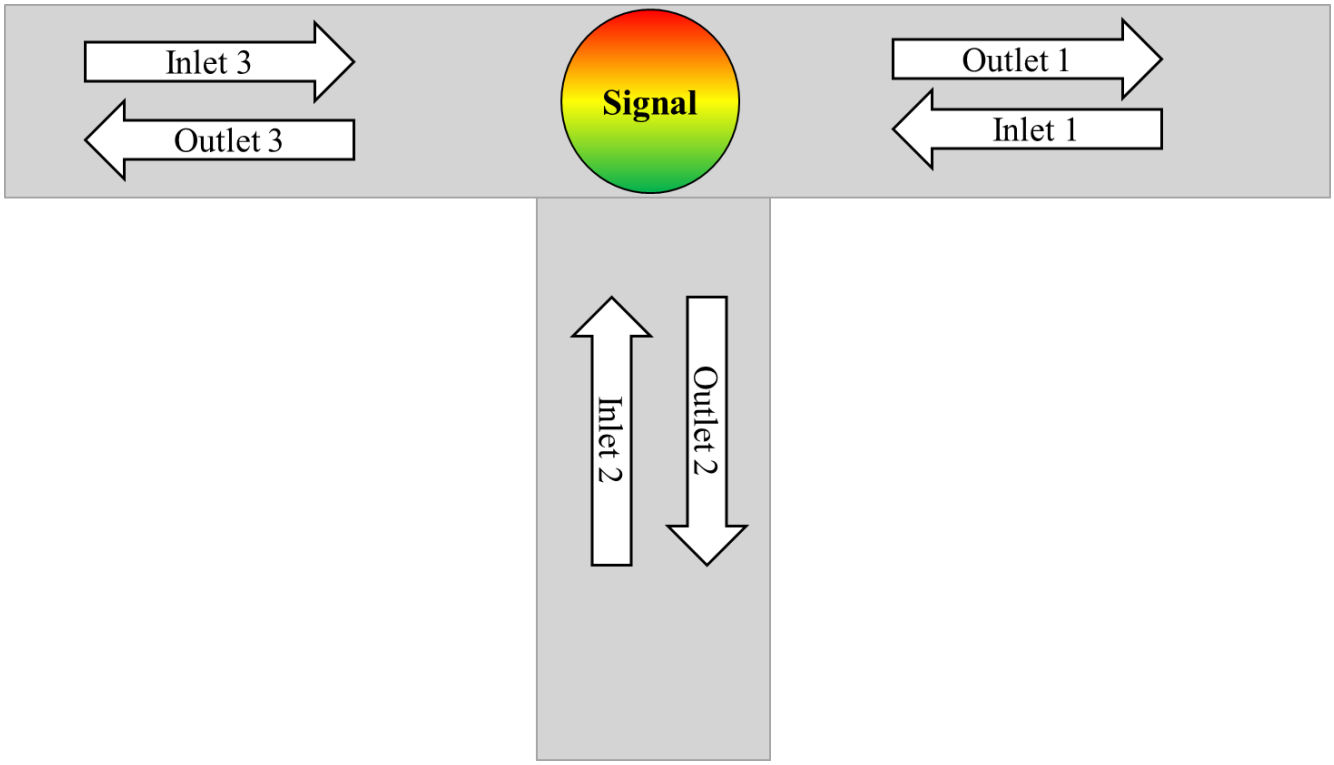
The user's input of base timers for different hours of the day is a one-time setup, ensuring that the system automatically adjusts the green signal duration accordingly. This not only simplifies the operation but also enhances the system's ability to adapt to varying traffic conditions throughout the day.

By providing users with the flexibility to customize signal timing based on different hours of the day, our project aims to improve traffic flow, reduce delays, and enhance overall transportation efficiency.

### 3.3.2.2 Difference based timer adjustment:

This part of the logic determines whether the timer should be increased, decreased, or remain unchanged. It compares the traffic on the main lane, which is the side where the green signal is starting, with the traffic on the next lane of the junction. The value of the timer is then determined based on the difference in vehicle counts.

For example, consider a T junction, as shown below:




Dig 3 T junction.

Assuming the signal at Inlet 1 is about to start, the controller will analyse the difference in the number of vehicles present at Inlet 1 and Inlet 2. If it determines that the number of vehicles on Inlet 2 is significantly higher than Inlet 1, it will decrease the timer at Inlet 1 to reduce the time required to handle the traffic at Inlet 2. Conversely, if Inlet 1 has higher traffic compared to Inlet 2, the controller will increase the green timer at Inlet 1 to address the congestion at that inlet.

The difference in vehicles between Inlet 1 and Inlet 2 determines whether the timer should be increased or decreased. The user sets the tolerance values for this difference during the setup process.
Difference in vehicles = Difference in vehicles at Inlet 1 - Difference in vehicles at Inlet 2
If the difference is significantly positive, it indicates heavy traffic at Inlet 1. Conversely, if the difference is significantly negative, it indicates heavy traffic at Inlet 2. If the difference falls within the tolerance value, it implies an equal amount of traffic on both sides, and therefore, there is no change in the timer.

### 3.3.3 Designing a user-oriented code to make vehicle counting logic and timer selection logic to work in synchronism:

The final step is to select a suitable controller, cameras, signal display LEDs, writing a correct code, ensuring that all necessary is considered and designing a backup system.

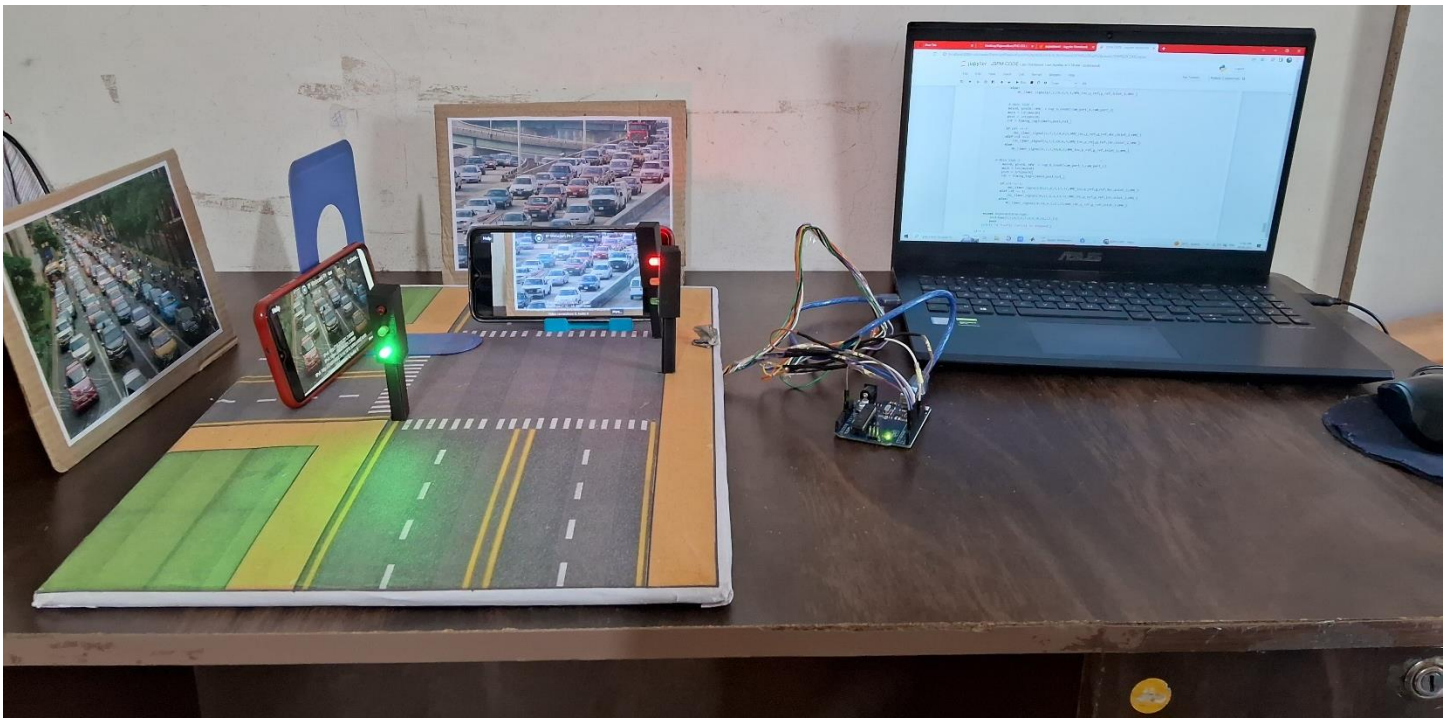
Recommended controllers are as follows [11]:

1. Nvidia Jetson TX1
2. Nvidia Jetson TK1
3. Raspberry Pi 3
4. Beagleboard: Beagle Bone Black
5. Orange Pi
6. ODROID-C2

Existing traffic cameras can be integrated into the system. But it is recommended to use cameras that can capture images of the highest quality. High quality images are essential in image processing. Also recommended to use wired connections for cameras. This will ensure that the system doesn't fail due to network issues.

The code for T junction demonstration is attached as an annexure at the end.

# CHAPTER 4: PROJECT PROTOTYPE



Dig 4: Project prototype.

To demonstrate our project, we built a small-scale demonstration of real time traffic control at T junction. Here we used Laptop of image processing. It captures images from mobile cameras wirelessly. Mobile had an app installed 'IP Webcam' which enabled the mobile to behave like CCTVs with an IP address. Traffic signal LEDs are controlled via Arduino from laptop through serial communication using Pyfirmata. Code for firmata protocol is added at the end. The standard fermata code is uploaded on Arduino UNO.

We took this project to multiple competitions where we were given valuable feedback.

## 4.1 Components in project prototype:

1. **Python 3:**

Python logo [16]

Python is a versatile programming language that is high-level and general-purpose. It prioritizes code readability through significant indentation in its design philosophy. Python is dynamically typed and features garbage collection. It offers support for various programming paradigms such as structured, object-oriented, and functional programming. Additionally, it is utilized in the model for tasks such as image capturing, storage, and processing.

2. **Embedded C:**

Embedded C refers to a collection of language extensions developed by the C Standards Committee specifically for the C programming language. Its purpose is to tackle common challenges found in C extensions for various embedded systems. When it comes to embedded C programming, it often involves the utilization of nonstandard extensions to the C language. These extensions enable support for advanced microprocessor functionalities like fixed-point arithmetic, multiple separate memory banks, and fundamental input/output operations. In the context of the model, Embedded C is employed to manage signals using the Arduino UNO platform.

3. **OpenCV:**


Open CV logo [16]

OpenCV is a programming library that primarily focuses on providing a wide range of functions for real-time computer vision tasks. Initially created by Intel, it received subsequent support from Willow Garage and Itrez (which was later acquired by Intel). This library is designed to be compatible with multiple platforms and is freely available under the open-source Apache 2 License. In the context at hand, OpenCV is utilized for executing various image processing tasks.

### 4. YOLO v4:

YOLO (You Only Look Once) is an advanced object detection algorithm introduced by Joseph Redmon in 2015, which remains a leading-edge solution to this day. Before the development of YOLO, detection algorithms relied on classifiers to detect objects. These algorithms involved applying the model to an image at various locations and scales, focusing on the regions with the highest scores for detection. However, this approach proved computationally intensive and was unable to deliver real-time performance on standard laptops.

In the present scenario, YOLO is employed specifically for identifying vehicles captured within an image.

### 5. COCO Dataset:

COCO, an acronym for Common Objects in Context, is an image dataset designed to advance image recognition. It provides computer vision with challenging and high-quality visual datasets, primarily used by state-of-the-art neural networks. COCO serves as a benchmark for evaluating real-time object detection algorithms. Advanced neural network libraries can automatically interpret the format of the COCO dataset. The MS COCO images dataset is licensed under the Creative Commons Attribution 4.0 License, allowing users to distribute, remix, tweak, and build upon the work, even for commercial purposes, as long as proper credit is given to the original creator. [17]

### 6. Arduino IDE:


Arduino IDE logo [16]

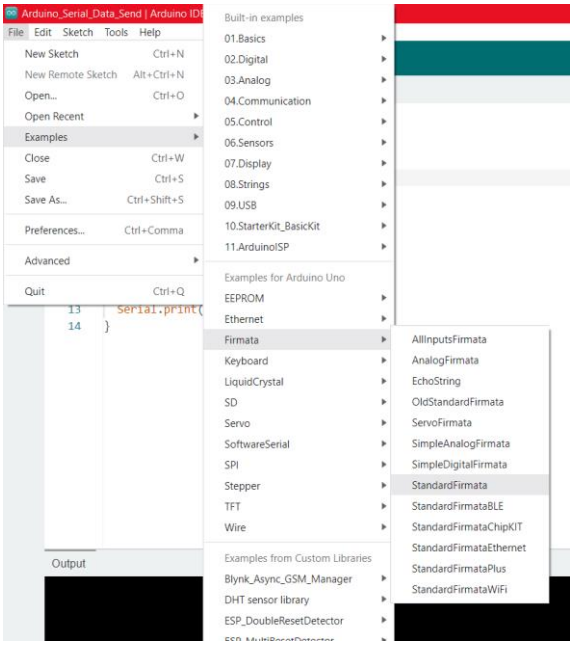
The Arduino Integrated Development Environment, also known as Arduino Software (IDE), comprises various components such as a code editor, a message area, a text console, a toolbar with commonly used functions, and a set of menus. It establishes a connection with Arduino hardware for program uploading and communication.

The programs written using Arduino Software (IDE) are referred to as sketches. These sketches are created in the code editor and saved with the file extension '.ino'. The editor provides features like cut, paste, search, and replace for manipulating text. The message area provides feedback during saving, exporting, and displays any encountered errors. The console presents text output from the Arduino Software (IDE), including detailed error messages and other relevant information. The bottom right corner of the window indicates the selected board and serial port. The toolbar buttons enable users to verify and upload programs, create, open, and save sketches, as well as access the serial monitor. [18]

Here, it is used to control LEDs which represent traffic signals.

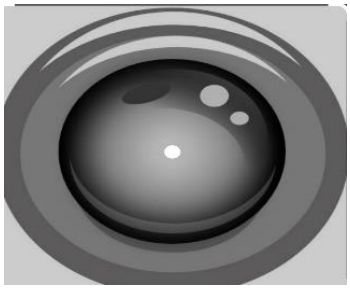
## 7. Pyfirmata:

pyFirmata is a Python interface for the Firmata protocol. It enables the laptop to control the LEDs connected to Arduino from Laptop. The code used for using fermata protocol can be found in the Arduino IDE as seen in the image below:




Dig 5: Standard Firmata code location in Arduino IDE.

## 8. IP Webcam:



IP Webcam logo [14]

An IP webcam (Internet Protocol webcam) is a type of camera that captures video and sends it over the internet or a local network using the Internet Protocol (IP). It allows users to view live video feeds or recorded footage from a remote Location using a computer, smartphone, or tablet.[14]

## 9. Arduino Uno:

Arduino UNO [15]

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc and initially released in 2010. 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. The board has 14 digital I/O pins (six capable of PWM output), 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable.

## 10. Laptop:

Due to cost restrictions and unavailability of desired processors, laptop is used to perform important part of the project.

## 11. Miscellaneous

LEDs were used to represent signals. Resistors were used to limit current flowing the circuit. Jumper wires were used for connections. Cardboard was used as base to develop the T junction model.

# CHAPTER 5: PROJECT COMPETITION PARTICIPATION SUMMARY

## ELECTFAIR 2K23:

This competition was organized by Marathwada Mitramandal's College of Engineering, Karve Nagar Pune in the month of March 2023. Here we were instructed by the judges to increase accuracy of the image processing algorithm. After this suggestion was given, we devised a plan to use 2 cameras placed at 150m to get an accurate depth of traffic. After this we created the ToD and Difference Based Timer Adjustment logic.

## Impetus and Concepts '23:

This competition was organized by PICT, Pune in the month of April 2023. Here we were instructed by the judges to make the project user oriented. They said what we had designed cannot be used as a product. They said design the code such that user can give input at the time of set up so that his needs can be satisfied. After this we rectified the project. We adjusted the code such that all the required data is given by the user.

## TECHPRO – 2K23:

This competition was organized by JSPM, Wagholi, Pune in the month of April 2023. Here we were instructed by the judges to fix the green signal timer to a certain limit. This will ensure that user doesn't mistakenly put impractical timer values like 30 mins.

Participation certificates are attached in form annexure at the end.

# CHAPTER 6: FUTURE SCOPE AND ADDITIONAL FEATURES

The ability of the system to collect real time data opens gateways to large number of customizations and data analysis. While implementing the project on large scale following points are figured out which can be implemented by state and central government depending on different type of junction present across the city

- Trucks & Buses take longer to pass through a junction than cars. Program can be modified such, if there are buses are trucks detected, the green timer can be increased by few seconds.

- Pedestrians can also be counted. This can help adjust pedestrian signals in regions where zebra crossing has huge importance. This can be beneficial for schools or colleges. Just before school starts or after school ends, it would observe that there is huge density of students on nearby roads. If pedestrian signal can accommodate to these conditions, it will help provide a much safer environment.

- With further research, system can be upgraded to detect emergency vehicles like ambulances and fire-trucks. The system can create a temporary green corridor for such vehicles.

- Training machine learning algorithm of signal on the images captured at the traffic junction can help increase accuracy of counting.

It should be noted that the system collects data. In our age, data is given huge importance. Negative forces could use the data collected. It can also be possible to hack into the traffic system. To prevent such disasters, proper cyber security measures must be taken.

# CHAPTER 7: CONCLUSION

In conclusion, this project aims to revolutionize traffic management by implementing a real-time and dynamic traffic signal control system. The existing traffic management systems, such as fixed timers and limited deployment of traffic police, are insufficient to handle the dynamic nature of traffic. By integrating real-time data and advanced technologies, this project offers innovative solutions to improve traffic flow, reduce congestion, and enhance overall transportation efficiency.

The project's objectives focus on enhancing the existing traffic management system to improve the quality of human life. By implementing real-time traffic signal control, the project aims to alleviate traffic congestion, reduce travel times, and create a more sustainable and livable environment. The impact of real-time traffic signal control is significant, providing optimized traffic flow, enhanced road safety, and valuable insights for traffic planning and management.

The work of the project involves designing a logic to count and detect vehicles using advanced deep learning techniques, specifically the YOLOV4 model. Real-time vehicle counting enables the collection of valuable data on traffic flow and congestion patterns. The project also incorporates logic for selecting suitable timers based on vehicle counts, including time-of-day control and difference-based timer adjustment. By combining these logics, the system adapts to varying traffic conditions, optimizing signal timings and reducing congestion.

To demonstrate the project's functionality, a small-scale demonstration was built, showcasing real-time traffic control at a T-junction. The system's architecture involved the use of a laptop for image processing, which captured images wirelessly from mobile cameras. The mobile cameras, acting as CCTV cameras with an IP address, provided high-quality images for accurate vehicle detection. The traffic signal LEDs were controlled via an Arduino board connected to the laptop through serial communication using Pyfirmata. This setup demonstrated the feasibility and effectiveness of the real-time traffic signal control system.

Throughout the development of the project, valuable feedback was received from multiple competitions. This feedback helped refine the project and address any shortcomings, ensuring its effectiveness and practicality. The participation in competitions also highlighted the project's potential and garnered recognition for its innovative approach to traffic management.

By implementing real-time traffic signal control, this project has the potential to transform traffic management and create a more efficient and livable transportation system. The integration of real-time data, advanced deep learning techniques, and customized timer logics allows for adaptive and responsive traffic signal control. Commuters can experience improved travel times, reduced congestion, and enhanced predictability. Furthermore, cities can benefit from better traffic management strategies, optimized infrastructure planning, and a positive environmental impact.

In conclusion, the implementation of real-time traffic signal control through this project brings forth a transformative approach to traffic management. By leveraging real-time data, advanced technologies, and customized timer logics, the project offers a solution to address the dynamic nature of traffic and improve the quality of human life. With its potential to optimize traffic flow, enhance road safety, and contribute to sustainable development, the real-time traffic signal control system holds great promise for the future of transportation and urban planning.

# Chapter 8: References


[1] A. B. B. K. Vidhya, “Density based traffic signal system,” International Journal of Innovative Research in Science, Engineering and Technology.

[2] Z. A. Promila Sinhmar, “Intelligent traffic light and density control using if sensors and microcontroller,” International Journal of Advanced Technology & Engineering Research.

[3] A. M. S. Khalil M. Yousef, Jamal N.Al-Karakai, “Intelligent traffic light flow con-trol system using wireless sensors networks,” Journal of Information Science and Engineering.

[4] T. K. A.D. Jadhav, Bhor Madhuri, “Intelligent traffic light control system,” 4th IRF International Conference ,ISBN: 978-93-82702-66-5.

[5] Python: https://assets.stickpng.com/images/5848152fcef1014c0b5e4967.png

[6] Arduino IDE: https://images.g2crowd.com/uploads/product/image/social_landscape/social_landscape_690e90c5de441951cf5715b08b1d5420/arduino-ide.png

[7] Open CV: https://upload.wikimedia.org/wikipedia/commons/thumb/3/32/OpenCV_Logo_with_text_svg_version.svg/1200px-OpenCV_Logo_with_text_svg_version.svg.png

[8] Jumper Wires: https://upload.wikimedia.org/wikipedia/commons/5/5c/A_few_Jumper_Wires.jpg

[9] https://en.wikipedia.org/wiki/Traffic_congestion

[10] https://timesofindia.indiatimes.com/city/pune/pcmcs-rs-9-crore-traffic-control-system-is-on-the-wrong-roads/articleshow/98955888.cms?from=mdr

[11] https://learnopencv.com/embedded-computer-vision-which-device-should-you-choose/

[12] https://en.wikipedia.org/wiki/Embedded_C

[13] https://medium.com/@buzzind99/using-the-state-of-the-art-yolov4-with-coco-dataset-to-detect-objects-6ee017016cd0

[14] https://play.google.com/store/apps/details?id=com.pas.webcam&hl=en&gl=US

[15] https://en.wikipedia.org/wiki/Arduino_Uno

[16] https://www.wikipedia.org/

[17] https://viso.ai/computer-vision/coco-dataset/

[18] https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics

# CHAPTER 9: ANNEXURES

## 9.1 Annexure A: Code

```
]: #Import Libraries
import numpy
import cv2
import os
import matplotlib.pyplot as plt
import cvlib as cv
import time as t
import sys
import keyboard
import serial
import pyfirmata
import threading
import datetime
#from datetime import datetime, time

# Import required methods from libraries

from cv2 import *
from cvlib.object_detection import draw_bbox
```

```
]: #Functions for 3 way signal

#Shuts down operation
def shutdown(r1,y1,gr1,gl1,r2,y2,gr2,gl2,r3,y3,gr3,gl3):
    board.digital[r1].write(0)
    board.digital[r2].write(0)
    board.digital[r3].write(0)
    board.digital[y1].write(0)
    board.digital[y2].write(0)
    board.digital[y3].write(0)
    board.digital[gr1].write(0)
    board.digital[gr2].write(0)
    board.digital[gr3].write(0)
    board.digital[gl1].write(0)
    board.digital[gl2].write(0)
    board.digital[gl3].write(0)

def NC_timer_signal(r,y,r_pre,r_post,g_r,g_l,hmv_delay,y_base,g_base,outlet_avail,hmv_logic):
    if hmv_logic == 0:
        g_base = g_base + 0
    else:
        g_base = hmv_delay + g_base
    g_base += hmv_delay
    board.digital[r_post].write(1)
    board.digital[r_pre].write(1)
    board.digital[r].write(0)
    board.digital[y].write(0)
    if outlet_avail[0] == 1:
        board.digital[g_l].write(1)
    else:
        board.digital[g_l].write(0)
    if outlet_avail[1] == 1:
        board.digital[g_r].write(1)
    else:
        board.digital[g_r].write(0)
    t.sleep(g_base)
    board.digital[g_l].write(0)
    board.digital[g_r].write(0)
    board.digital[y].write(1)
    t.sleep(y_base)
    board.digital[y].write(0)
    board.digital[r].write(1)

def inc_timer_signal(r,y,r_pre,r_post,g_r,g_l,hmv_delay,y_base,g_base,incr,outlet_avail,hmv_logic):
    if hmv_logic == 0:
        g_base = g_base+incr
    else:
        g_base = hmv_delay+g_base+incr
    board.digital[r_post].write(1)
    board.digital[r_pre].write(1)
    board.digital[r].write(0)
    board.digital[y].write(0)
    if outlet_avail[0] == 1:
        board.digital[g_l].write(1)
    else:
        board.digital[g_l].write(0)
    if outlet_avail[1] == 1:
        board.digital[g_r].write(1)
    else:
        board.digital[g_r].write(0)

    t.sleep(g_base)
    board.digital[g_l].write(0)
    board.digital[g_r].write(0)
    board.digital[y].write(1)
    t.sleep(y_base)
    board.digital[y].write(0)
    board.digital[r].write(1)


#d
def dec_timer_signal(r,y,r_pre,r_post,g_r,g_l,hmv_delay,y_base,g_base,decr,outlet_avail,hmv_logic):
    if hmv_logic == 0:
        g_base = g_base-decr
    else:
        g_base = hmv_delay+g_base+decr
    board.digital[r_post].write(1)
    board.digital[r_pre].write(1)
    board.digital[r].write(0)
    board.digital[y].write(0)
    if outlet_avail[0] == 1:
```

```
        board.digital[g_l].write(1)
    else:
        board.digital[g_l].write(0)
    if outlet_avail[1] == 1:
        board.digital[g_r].write(1)
    else:
        board.digital[g_r].write(0)
    t.sleep(g_base)
    board.digital[g_l].write(0)
    board.digital[g_r].write(0)
    board.digital[y].write(1)
    t.sleep(y_base)
    board.digital[y].write(0)
    board.digital[r].write(1)

def Off_period(r1,r2,r3,y1,y2,y3,g1,g2,g3):
    board.digital[y1].write(1)
    board.digital[y2].write(1)
    board.digital[y3].write(1)
    board.digital[r1].write(0)
    board.digital[r2].write(0)
    board.digital[r3].write(0)
    board.digital[g1].write(0)
    board.digital[g2].write(0)
    board.digital[g3].write(0)

#Function for Capture & Count
def cap_n_count(IP_main,IP_post):

    #Cap n Count Main side
    cam_main = cv2.VideoCapture(IP_main)
    result_main, image_main = cam_main.read()
    cv2.imwrite("a.png",image_main)
    im_main = cv2.imread("a.png")
    bbox, label, conf = cv.detect_common_objects(im_main)
    print('detecting object')
    p_main = int(label.count("person"))
    count_main = int(label.count("car"))+int(label.count("truck"))+int(label.count("bus"))+int(label.count("motorcycle"))
    hmv_count = int(label.count("truck"))+int(label.count("bus"))
    print('\n Number of cars on main side is '+ str(count_main))
    print('\n Number of people on main side is '+ str(p_main))
    print('\n Number of HMVs on main side is '+ str(hmv_count))
    if hmv_count == 0:
        hmv_logic = 0
    else:
        hmv_logic = 1

    #Cap n Count Post side
    cam_post = cv2.VideoCapture(IP_post)
    result_post, image_post = cam_post.read()
    cv2.imwrite("b.png",image_post)
    im_post = cv2.imread("b.png")
    bbox, label, conf = cv.detect_common_objects(im_post)
    count_post = int(label.count("car"))+int(label.count("truck"))+int(label.count("bus"))+int(label.count("motorcycle"))
    hmv_count = int(label.count("truck"))+int(label.count("bus"))
    print('\n Number of cars on next side is '+ str(count_post))

    return count_main,count_post,hmv_logic
    print("\n Image capturr and count complete.")

# Traffic Prioity & Tolerance & Timing Logic
def timing_logic(main_sig,post_sig,tol):
    if main_sig - post_sig < (tol*-1):
        print('\nAs traffic is heavy on next signal.Priority is given to next signal.\nTimer of current signal is decremented.')
        count = -1
    elif main_sig - post_sig > tol:
        print('\nAs traffic is heavy on current signal.Priority is given to current signal.\nTimer of current signal is incremented.')
        count = 1
    else:
        print('\nSimilar amount of traffic is present on both sides. \Hence no change in timer.')
        count = 0
    return count
```

[ ]:
```
#Initialize UNO
UNO_port_num = input('Enter port number of UNO: ')
UNO_port = 'COM' + UNO_port_num
board = pyfirmata.Arduino(UNO_port)
#Lalit COM5
#Rajwardhan COM6
```

[ ]:
```
#Select type of junction
print('\nWhich of the following best describes your junction?')
print('\n 1. 3 way \n 2. 4 way \n 3. 5 way')
junc_type = int(input('Enter corresponding number which describes your junction: '))
if junc_type == 1 or junc_type == 2 or junc_type == 3:
    if junc_type == 1:
        print('\n 3 way Junction type selected')
    if junc_type == 2:
        print('\n 4 way Junction type selected')
        print('Sorry we dont have that option. Run the code again.')
    if junc_type == 3:
        print('\n 5 way Junction type selected')
        print('Sorry we dont have that option. Run the code again.')
else:
    print('Sorry we dont have that option. Run the code again.')
```

[ ]:
```
#Get Camera address:
print('\n Please give the correct IP address for your cameras')
print('\n Type no if you do not have a 3rd, 4th or 5th inlet')
if junc_type == 1:
    cam_port_1 = input('Enter IP address for camera at inlet 1: ')
    cam_port_2 = input('Enter IP address for camera at inlet 2: ')
```

```
    cam_port_3 = input('Enter IP address for camera at inlet 3: ')

elif junc_type == 2:
    cam_port_1 = input('Enter IP address for camera at inlet 1: ')
    cam_port_2 = input('Enter IP address for camera at inlet 2: ')
    cam_port_3 = input('Enter IP address for camera at inlet 3: ')
    cam_port_4 = input('Enter IP address for camera at inlet 4: ')
elif junc_type == 3:
    cam_port_1 = input('Enter IP address for camera at inlet 1: ')
    cam_port_2 = input('Enter IP address for camera at inlet 2: ')
    cam_port_3 = input('Enter IP address for camera at inlet 3: ')
    cam_port_4 = input('Enter IP address for camera at inlet 4: ')
    cam_port_5 = input('Enter IP address for camera at inlet 5: ')
```

```
[ ]: # Directions for each signal
inlet_1 = []
inlet_2 = []
inlet_3 = []
inlet_4 = []
inlet_5 = []

if  junc_type == 1:

    print('\nVisualize the 3 way junction in your head. Mark each inlet and outlet numerically in a clockwise order.')
    print('Consider markings as shown in above image as a representation of your junction')
    print('\nPlease provide on which directions vehicles can go in at each inlet.\nProvide this data in the following format.')
    print('\n1 stands for yes and 0 for no.')
    print('\nFor e.g. When it asks, can Vehicles at inlet 3 turn on left. Type in 1 for yes. 0 for no')
    #Directions at inlet 1
    out_1 = int(input('At inlet 1 Can vehicle go out through 1st outlet on left? '))
    inlet_1.append(out_1)
    out_2 = int(input('At inlet 1 Can vehicle go out through 2nd outlet on left? '))
    inlet_1.append(out_2)


    #Directions at inlet 2
    out_1 = int(input('At inlet 2 Can vehicle go out through 1st outlet on left? '))
    inlet_2.append(out_1)
    out_2 = int(input('At inlet 2 Can vehicle go out through 2nd outlet on left? '))
    inlet_2.append(out_2)


    #Directions at inlet 3
    out_1 = int(input('At inlet 3 Can vehicle go out through 1st outlet on left? '))
    inlet_3.append(out_1)
    out_2 = int(input('At inlet 3 Can vehicle go out through 2nd outlet on left? '))
    inlet_3.append(out_2)



elif junc_type == 2:
    print('\nVisualize the 4 way junction in your head. Mark each inlet and outlet numerically in a cyclic order.\nAbove image can help you.')
    print('\nPlease provide on which directions vehicles can go in at each inlet.\nProvide this data in the following format.')
    print('\n1 stands for yes and 0 for no.')
    print('\nFor e.g. Consider at a 4 way junction if a vehicle at an particular inlet can go in left and forward but not right direction.')
    print('\nIn that case give input in follwing format: \n1,1,0')
    print('\n1st index stands for left, 2nd for forward aka 1st direction after left and 3rd for right direction 2nd direction after left')

    #Directions at inlet 1
    out_1 = int(input('At inlet 1 Can vehicle go out through 1st outlet on left? '))
    inlet_1.append(out_1)
    out_2 = int(input('At inlet 1 Can vehicle go out through 2nd outlet on left? '))
    inlet_1.append(out_2)
    out_3 = int(input('At inlet 1 Can vehicle go out through 3rd outlet on left? '))
    inlet_1.append(out_3)


    #Directions at inlet 2
    out_1 = int(input('At inlet 2 Can vehicle go out through 1st outlet on left? '))
    inlet_1.append(out_1)
    out_2 = int(input('At inlet 2 Can vehicle go out through 2nd outlet on left? '))
    inlet_1.append(out_2)
    out_3 = int(input('At inlet 2 Can vehicle go out through 3rd outlet on left? '))
    inlet_1.append(out_3)


    #Directions at inlet 3
    out_1 = int(input('At inlet 3 Can vehicle go out through 1st outlet on left? '))
    inlet_1.append(out_1)
    out_2 = int(input('At inlet 3 Can vehicle go out through 2nd outlet on left? '))
    inlet_1.append(out_2)
    out_3 = int(input('At inlet 3 Can vehicle go out through 3rd outlet on left? '))
    inlet_1.append(out_3)


    #Directions at inlet 4
    out_1 = int(input('At inlet 4 Can vehicle go out through 1st outlet on left? '))
    inlet_1.append(out_1)
    out_2 = int(input('At inlet 4 Can vehicle go out through 2nd outlet on left? '))
    inlet_1.append(out_2)
    out_3 = int(input('At inlet 4 Can vehicle go out through 3rd outlet on left? '))
    inlet_1.append(out_3)


elif  junc_type == 3:

    print('\nVisualize the 5 way junction in your head. Mark each inlet and outlet numerically in a cyclic order.')
    print('\nPlease provide on which directions vehicles can go in at each inlet.\nProvide this data in the following format.')
    print('\n1 stands for yes and 0 for no.')
    print(' \nFor e.g. Consider at a 4 way junction if a vehicle at an particular inlet can go in left and forward but not right direction.')
    print('\n1,1,0')
    print('\n1st index stands for left, 2nd for forward aka 1st direction after left and 3rd for right direction 2nd direction after left')
```

```
    #Directions at inlet 1
    out_1 = int(input('At inlet 1 Can vehicle go out through 1st outlet on left? '))
    inlet_1.append(out_1)
    out_2 = int(input('At inlet 1 Can vehicle go out through 2nd outlet on left? '))
    inlet_1.append(out_2)
    out_3 = int(input('At inlet 1 Can vehicle go out through 3rd outlet on left? '))
    inlet_1.append(out_3)
    out_4 = int(input('At inlet 1 Can vehicle go out through 4th outlet on left? '))
    inlet_1.append(out_4)

    #Directions at inlet 2
    out_1 = int(input('At inlet 2 Can vehicle go out through 1st outlet on left? '))
    inlet_2.append(out_1)
    out_2 = int(input('At inlet 2 Can vehicle go out through 2nd outlet on left? '))
    inlet_2.append(out_2)
    out_3 = int(input('At inlet 2 Can vehicle go out through 3rd outlet on left? '))
    inlet_2.append(out_3)
    out_4 = int(input('At inlet 2 Can vehicle go out through 4th outlet on left? '))
    inlet_1.append(out_4)

    #Directions at inlet 3
    out_1 = int(input('At inlet 3 Can vehicle go out through 1st outlet on left? '))
    inlet_3.append(out_1)
    out_2 = int(input('At inlet 3 Can vehicle go out through 2nd outlet on left? '))
    inlet_3.append(out_2)
    out_3 = int(input('At inlet 3 Can vehicle go out through 3rd outlet on left? '))
    inlet_3.append(out_3)
    out_4 = int(input('At inlet 3 Can vehicle go out through 4th outlet on left? '))
    inlet_1.append(out_4)

    #Directions at inlet 4
    out_1 = int(input('At inlet 4 Can vehicle go out through 1st outlet on left? '))
    inlet_4.append(out_1)
    out_2 = int(input('At inlet 4 Can vehicle go out through 2nd outlet on left? '))
    inlet_4.append(out_2)
    out_3 = int(input('At inlet 4 Can vehicle go out through 3rd outlet on left? '))
    inlet_4.append(out_3)
    out_4 = int(input('At inlet 4 Can vehicle go out through 4th outlet on left? '))
    inlet_4.append(out_4)

    #Directions at inlet 5
    out_1 = int(input('At inlet 5 Can vehicle go out through 1st outlet on left? '))
    inlet_5.append(out_1)
    out_2 = int(input('At inlet 5 Can vehicle go out through 2nd outlet on left? '))
    inlet_5.append(out_2)
    out_3 = int(input('At inlet 5 Can vehicle go out through 3rd outlet on left? '))
    inlet_5.append(out_3)
    out_4 = int(input('At inlet 5 Can vehicle go out through 4th outlet on left? '))
    inlet_5.append(out_4)
```

]:

]:
```
#Increment or decrement value
print('\nIncrement or decrement value should be selected on basis of following parameters')
print('\n 1. Distance between Inlet and outlet. \n 2. Width of road.')
inc = int(input('\nIf traffic is detected on current signal, increment timer by how many seconds ?'))
dec = int(input('\nIf traffic is detected on next signals, decrement timer by how many seconds? '))
HMV_inc = int(input('\nIf HMVs are detected on current signal, increment timer by how many seconds? \n Put zero if you dont want increment'))
tol_ = int(input('\nWhat is the tolerance to assign/remove priority of main signal? '))
```

]:
```
#Set base value
g_ref_1 = int(input(print('What should be base timer for green signal during 5:00 to 9:00 ?')))
g_ref_2 = int(input(print('What should be base timer for green signal during 9:00 to 13:00 ?')))
g_ref_3 = int(input(print('What should be base timer for green signal during 13:00 to 16:00 ?')))
g_ref_4 = int(input(print('What should be base timer for green signal during 16:00 to 21:00 ?')))
g_ref_5 = int(input(print('What should be base timer for green signal during 21:00 to 23:00?')))
```

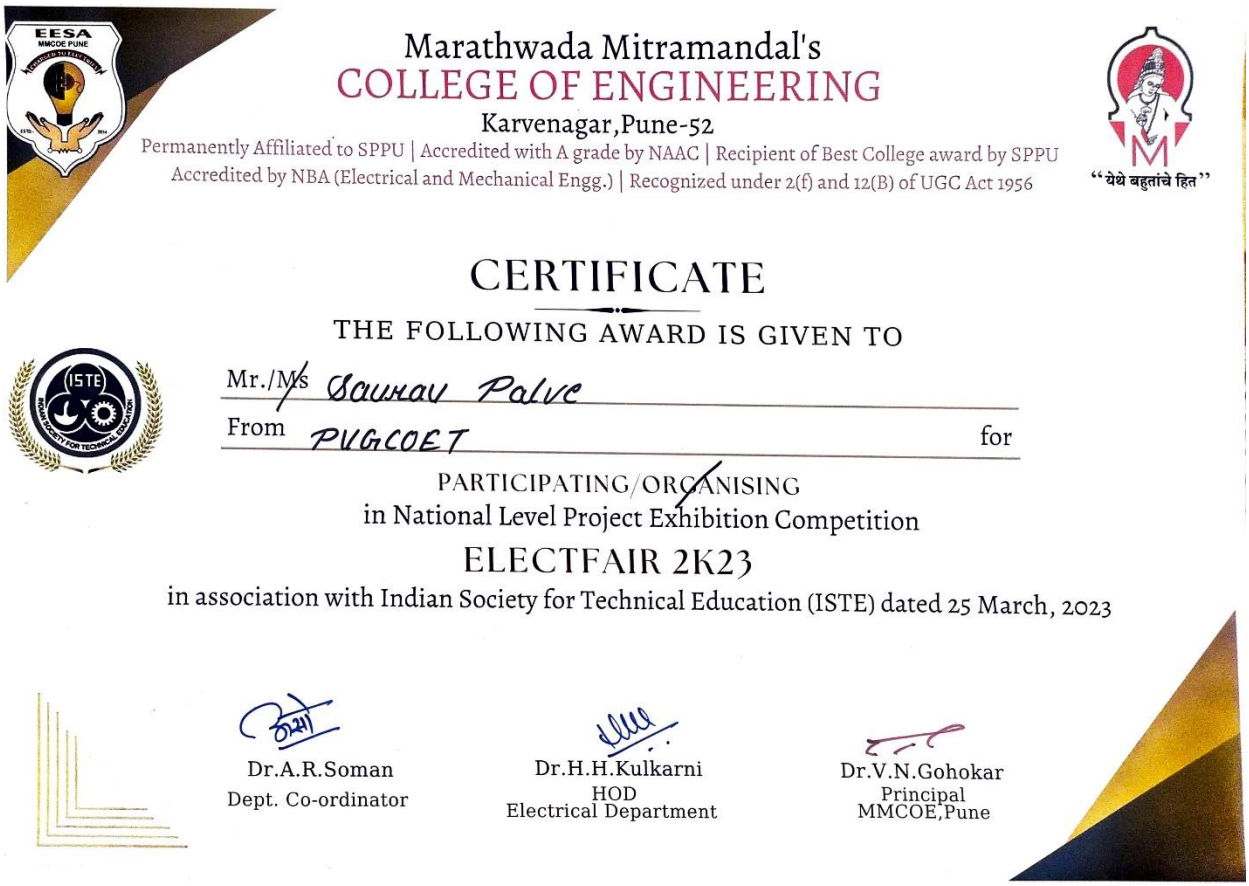
## CODE FOR 3 WAY JUNCTION

]:
```
if junc_type ==1:
    direction = 3
    try:
        while True:

        #Time Of Day Adaption Traffic Control
            current_time = datetime.datetime.now()

            if datetime.time(5, 0) <= current_time.time() < datetime.time(9, 0):
                r_ref = 10
                y_ref = 5
                g_ref = g_ref_1
                print('slot 1')
            elif datetime.time(9, 0) <= current_time.time() < datetime.time(13, 0):
                r_ref = 10
                y_ref = 5
                g_ref = g_ref_2
                print('slot 2')
            elif datetime.time(13, 0) <= current_time.time() < datetime.time(16, 0):
                r_ref = 10
                y_ref = 5
                g_ref = g_ref_3
                print('slot 3')
            elif datetime.time(16, 0) <= current_time.time() < datetime.time(21, 0):
                r_ref = 10
                y_ref = 5
                g_ref = g_ref_4
                print('slot 4')
            elif datetime.time(21, 0) <= current_time.time() < datetime.time(23,0):
```

```
            r_ref = 10
            y_ref = 5
            g_ref = g_ref_5
            print('slot 5')
        else:
            Off_period(13,10,7,12,9,6,11,8,5)
            print('slot 6')
            break



        # Main Side 1
        main0, post0, HMV_ = cap_n_count(cam_port_1,cam_port_2)
        main = int(main0)
        post = int(post0)
        cnt = timing_logic(main,post,tol_)

        #Cannot use digital pins 0 & 1, hence, red and yellow of inlet 1 are connected to pins 5&6
        if cnt ==-1:
            dec_timer_signal(2,3,10,6,5,4,HMV_inc,y_ref,g_ref,dec,inlet_1,HMV_)
        elif cnt ==1:
            inc_timer_signal(2,3,10,6,5,4,HMV_inc,y_ref,g_ref,inc,inlet_1,HMV_)
        else:
            NC_timer_signal(2,3,10,6,5,4,HMV_inc,y_ref,g_ref,inlet_1,HMV_)


        # Main Side 2
        main0, post0, HMV_ = cap_n_count(cam_port_2,cam_port_3)
        main = int(main0)
        post = int(post0)
        cnt = timing_logic(main,post,tol_)

        if cnt ==-1:
            dec_timer_signal(6,7,2,10,8,9,HMV_inc,y_ref,g_ref,dec,inlet_2,HMV_)
        elif cnt ==1:
            inc_timer_signal(6,7,2,10,8,9,HMV_inc,y_ref,g_ref,inc,inlet_2,HMV_)
        else:
            NC_timer_signal(6,7,2,10,8,9,HMV_inc,y_ref,g_ref,inlet_2,HMV_)


    # Main Side 3
        main0, post0, HMV_ = cap_n_count(cam_port_3,cam_port_1)
        main = int(main0)
        post = int(post0)
        cnt = timing_logic(main,post,tol_)

        if cnt ==-1:
            dec_timer_signal(10,11,6,2,13,12,HMV_inc,y_ref,g_ref,dec,inlet_3,HMV_)
        elif cnt == 1:
            inc_timer_signal(10,11,6,2,13,12,HMV_inc,y_ref,g_ref,inc,inlet_3,HMV_)
        else:
            NC_timer_signal(10,11,6,2,13,12,HMV_inc,y_ref,g_ref,inlet_3,HMV_)


except KeyboardInterrupt:
    shutdown(2,3,4,5,6,7,8,9,10,11,12,13)
    pass
print('\n Traffic Control is Stopped')
```

]:

## 9.2 Annexure B: Participation certificates (MMCOE)



Marathwada Mitramandal's
COLLEGE OF ENGINEERING
Karvenagar, Pune-52
Permanently Affiliated to SPPU | Accredited with A grade by NAAC | Recipient of Best College award by SPPU
Accredited by NBA (Electrical and Mechanical Engg.) | Recognized under 2(f) and 12(B) of UGC Act 1956

"येथे बहुतांचे हित"

CERTIFICATE

THE FOLLOWING AWARD IS GIVEN TO

Mr./Ms Saurav Palve

From PUGCOET for

PARTICIPATING/ORGANISING
in National Level Project Exhibition Competition

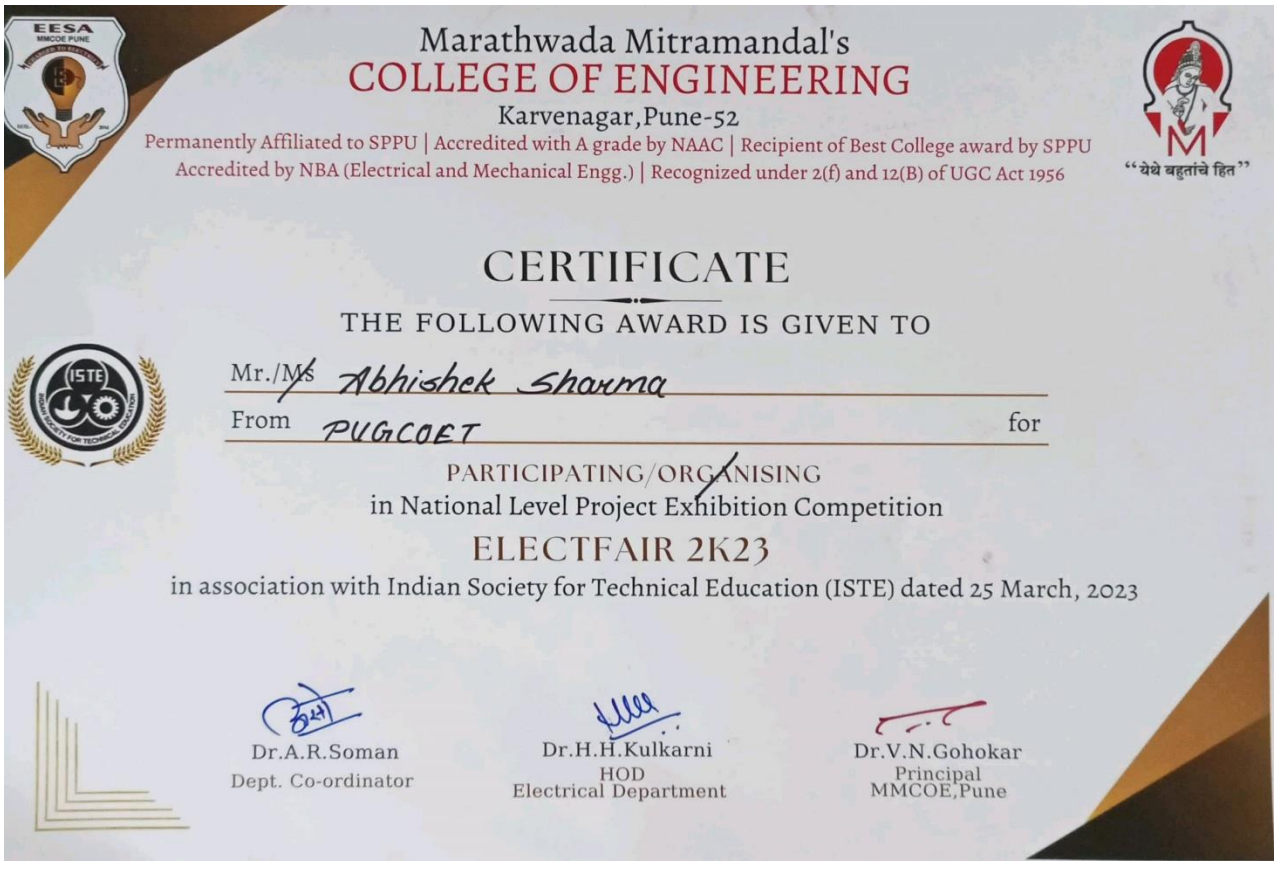
ELECTFAIR 2K23

in association with Indian Society for Technical Education (ISTE) dated 25 March, 2023

Dr.A.R.Soman
Dept. Co-ordinator

Dr.H.H.Kulkarni
HOD
Electrical Department

Dr.V.N.Gohokar
Principal
MMCOE,Pune



Marathwada Mitramandal's
COLLEGE OF ENGINEERING
Karvenagar, Pune-52
Permanently Affiliated to SPPU | Accredited with A grade by NAAC | Recipient of Best College award by SPPU
Accredited by NBA (Electrical and Mechanical Engg.) | Recognized under 2(f) and 12(B) of UGC Act 1956

"येथे बहुतांचे हित"

CERTIFICATE

THE FOLLOWING AWARD IS GIVEN TO

Mr./Ms Abhishek Sharma

From PUGCOET for

PARTICIPATING/ORGANISING
in National Level Project Exhibition Competition

ELECTFAIR 2K23

in association with Indian Society for Technical Education (ISTE) dated 25 March, 2023

Dr.A.R.Soman
Dept. Co-ordinator

Dr.H.H.Kulkarni
HOD
Electrical Department

Dr.V.N.Gohokar
Principal
MMCOE,Pune

Marathwada Mitramandal's
# COLLEGE OF ENGINEERING
Karvenagar, Pune-52
Permanently Affiliated to SPPU | Accredited with A grade by NAAC | Recipient of Best College award by SPPU
Accredited by NBA (Electrical and Mechanical Engg.) | Recognized under 2(f) and 12(B) of UGC Act 1956

"येथे बहुतांचे हित"

## CERTIFICATE

THE FOLLOWING AWARD IS GIVEN TO

Mr./Ms Lalit Suresh Chaudhari

From PVGCOET for

PARTICIPATING/ORGANISING

in National Level Project Exhibition Competition

ELECTFAIR 2K23

in association with Indian Society for Technical Education (ISTE) dated 25 March, 2023

Dr.A.R.Soman
Dept. Co-ordinator

Dr.H.H.Kulkarni
HOD
Electrical Department

Dr.V.N.Gohokar
Principal
MMCOE, Pune

Marathwada Mitramandal's
COLLEGE OF ENGINEERING
Karvenagar, Pune-52
Permanently Affiliated to SPPU | Accredited with A grade by NAAC | Recipient of Best College award by SPPU
Accredited by NBA (Electrical and Mechanical Engg.) | Recognized under 2(f) and 12(B) of UGC Act 1956

"येथे बहुतांचे हित"

CERTIFICATE

THE FOLLOWING AWARD IS GIVEN TO

Mr./Ms Rajwardhan Ghorpade

From PVGCOET for

PARTICIPATING/ORGANISING

in National Level Project Exhibition Competition

ELECTFAIR 2K23

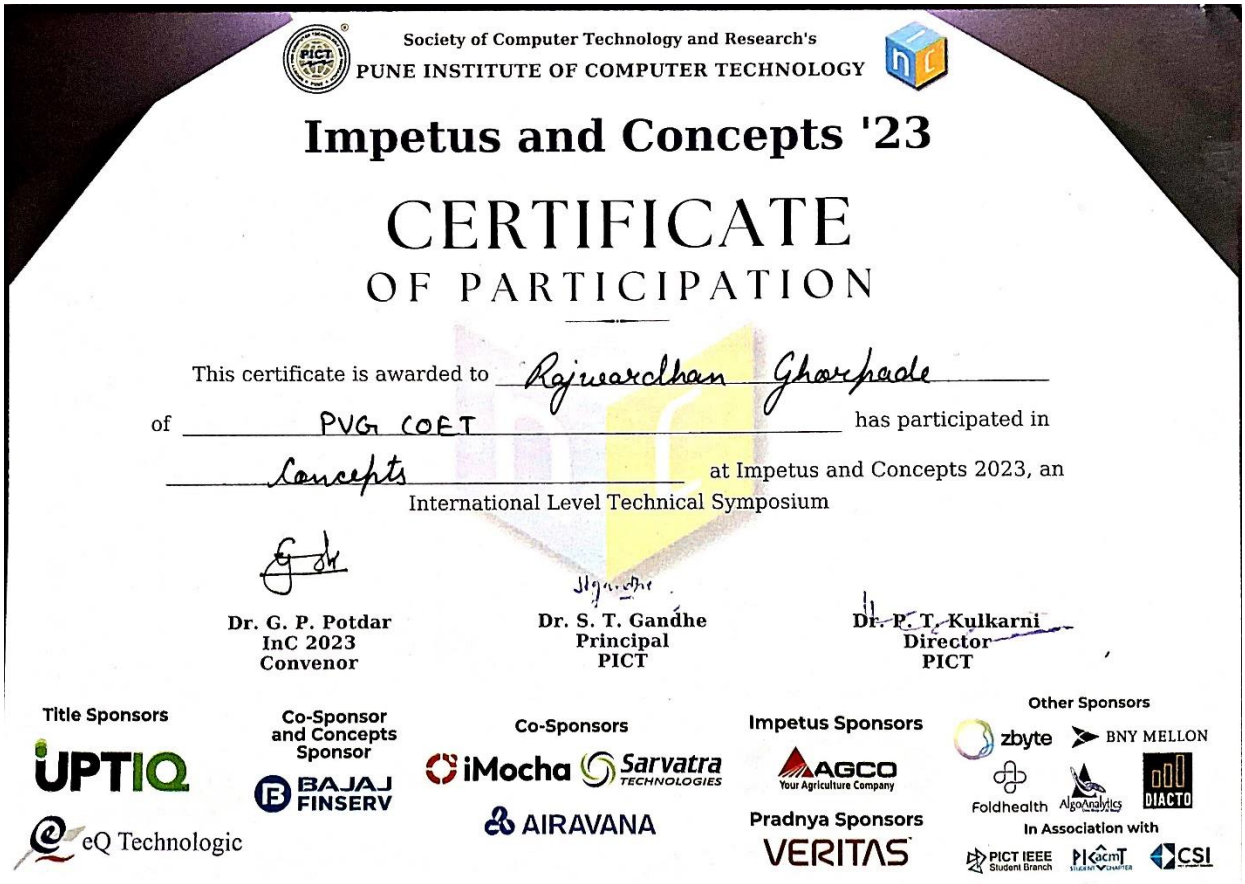
in association with Indian Society for Technical Education (ISTE) dated 25 March, 2023

Dr.A.R.Soman
Dept. Co-ordinator

Dr.H.H.Kulkarni
HOD
Electrical Department

Dr.V.N.Gohokar
Principal
MMCOE, Pune

## 9.3 Annexure C: Participation certificates (PICT)



Society of Computer Technology and Research's
PUNE INSTITUTE OF COMPUTER TECHNOLOGY

**Impetus and Concepts '23**

CERTIFICATE
OF PARTICIPATION

This certificate is awarded to Rajvardhan Ghorpade
of PVG COET has participated in
Concepts at Impetus and Concepts 2023, an
International Level Technical Symposium

Dr. G. P. Potdar
InC 2023
Convenor

Dr. S. T. Gandhe
Principal
PICT

Dr. P. T. Kulkarni
Director
PICT

Title Sponsors: UPTIQ, eQ Technologic
Co-Sponsor and Concepts Sponsor: BAJAJ FINSERV
Co-Sponsors: iMocha, Sarvatra Technologies, AIRAVANA
Impetus Sponsors: AGCO Your Agriculture Company
Pradnya Sponsors: VERITAS
Other Sponsors: zbyte, BNY MELLON, Foldhealth, AlgoAnalytics, DIACTO
In Association with: PICT IEEE Student Branch, PICT acmT Student Chapter, CSI

Society of Computer Technology and Research's
PUNE INSTITUTE OF COMPUTER TECHNOLOGY

**Impetus and Concepts '23**

CERTIFICATE
OF PARTICIPATION

This certificate is awarded to Saurav Palve
of PVGCOET has participated in
Concepts at Impetus and Concepts 2023, an
International Level Technical Symposium

Dr. G. P. Potdar
InC 2023
Convenor

Dr. S. T. Gandhe
Principal
PICT

Dr. P. T. Kulkarni
Director
PICT

Co-Sponsor and Concepts Sponsor: BAJAJ FINSERV
Co-Sponsors: iMocha, Sarvatra Technologies, AIRAVANA
Impetus Sponsors: AGCO Your Agriculture Company
Pradnya Sponsors: VERITAS
Other Sponsors: zbyte, BNY MELLON, Foldhealth, AlgoAnalytics, DIACTO
In Association with: PICT IEEE Student Branch, PICT acmT Student Chapter, CSI

Society of Computer Technology and Research's
PUNE INSTITUTE OF COMPUTER TECHNOLOGY

# Impetus and Concepts '23

## CERTIFICATE
### OF PARTICIPATION

This certificate is awarded to Lalit Suresh Chaudhari

of PVG COET has participated in

Concepts at Impetus and Concepts 2023, an International Level Technical Symposium

Dr. G. P. Potdar
InC 2023
Convenor

Dr. S. T. Gandhe
Principal
PICT

Dr. P. T. Kulkarni
Director
PICT

Title Sponsors: UPTIQ, eQ Technologic
Co-Sponsor and Concepts Sponsor: BAJAJ FINSERV
Co-Sponsors: iMocha, Sarvatra Technologies, AIRAVANA
Impetus Sponsors: AGCO
Pradnya Sponsors: VERITAS
Other Sponsors: zbyte, BNY MELLON, Foldhealth, AlgoAnalytics, DIACTO
In Association with: PICT IEEE Student Branch, ACM, CSI

Society of Computer Technology and Research's
PUNE INSTITUTE OF COMPUTER TECHNOLOGY

# Impetus and Concepts '23

## CERTIFICATE
### OF PARTICIPATION

This certificate is awarded to Abhishek Sharma

of PVG COET has participated in

Concepts at Impetus and Concepts 2023, an International Level Technical Symposium

Dr. G. P. Potdar
InC 2023
Convenor

Dr. S. T. Gandhe
Principal
PICT

Dr. P. T. Kulkarni
Director
PICT

Title Sponsors: UPTIQ, eQ Technologic
Co-Sponsor and Concepts Sponsor: BAJAJ FINSERV
Co-Sponsors: iMocha, Sarvatra Technologies, AIRAVANA
Impetus Sponsors: AGCO
Pradnya Sponsors: VERITAS
Other Sponsors: zbyte, BNY MELLON, Foldhealth, AlgoAnalytics, DIACTO
In Association with: PICT IEEE Student Branch, ACM, CSI

# 9.4 Annexure D: Participation certificates (JSPM)

JAYAWANT SHIKSHAN PRASARAK MANDAL'S

BHIVARABAI SAWANT INSTITUTE OF TECHNOLOGY AND RESEARCH
WAGHOLI, PUNE.

TECHPRO-2K23

Organized by

DEPARTMENT OF ELECTRICAL ENGINEERING

(Accredited by NBA, NAAC & Recognized by AICTE)

This certificate is proudly presented to

Mr./~~Ms.~~ LALIT SURESH CHAUDHARI for project Titled REAL TIME TRAFFIC SIGNAL CONTROL of PUNE VIDYARTHI GRIHAS COLLEGE OF ENGINEERING. for Participating in TECHPRO-2K23 a National Level Project Competition Held on April 26, 2023.

Dr. N. N. Ghuge
CONVENER

Dr. T. K. Nagaraj
PRINCIPAL

Sponsored by

Electronics Study Centre | Net Protector NP AV Cyber Security | SYPROTECH SOLUTIONS Your Success Partner

JAYAWANT SHIKSHAN PRASARAK MANDAL'S

BHIVARABAI SAWANT INSTITUTE OF TECHNOLOGY AND RESEARCH
WAGHOLI, PUNE.

TECHPRO-2K23

Organized by

DEPARTMENT OF ELECTRICAL ENGINEERING

(Accredited by NBA, NAAC & Recognized by AICTE)

This certificate is proudly presented to

Mr./~~Ms.~~ RAJWARDHAN SHIVRAJ GHORPADE for project Titled REAL TIME TRAFFIC SIGNAL CONTROL of PUNE VIDYARTHI GRIHAS COLLEGE OF ENGINEERING for Participating in TECHPRO-2K23 a National Level Project Competition Held on April 26, 2023.

Dr. N. N. Ghuge
CONVENER

Dr. T. K. Nagaraj
PRINCIPAL

Sponsored by

Electronics Study Centre | Net Protector NP AV Cyber Security | SYPROTECH SOLUTIONS Your Success Partner

JAYAWANT SHIKSHAN PRASARAK MANDAL'S

BHIVARABAI SAWANT INSTITUTE OF TECHNOLOGY AND RESEARCH
WAGHOLI, PUNE.

# TECHPRO-2K23

Organized by

**DEPARTMENT OF ELECTRICAL ENGINEERING**

(Accredited by NBA, NAAC & Recognized by AICTE)

This certificate is proudly presented to

Mr./~~Ms.~~ SAURAV PANDHARINATH PALVE for project Titled REAL TIME TRAFFIC SIGNAL CONTROL of PUNE VIDYARTHI GRIHAS COLLEGE OF ENGINEERING for Participating in **TECHPRO-2K23** a National Level Project Competition Held on April 26, 2023.

Dr. N. N. Ghuge
CONVENER

Dr. T. K. Nagaraj
PRINCIPAL

**Sponsored by**

Electronics Study Centre | Net Protector NPAV Cyber Security | SYPROTECH SOLUTIONS Your Success Partner

JAYAWANT SHIKSHAN PRASARAK MANDAL'S

BHIVARABAI SAWANT INSTITUTE OF TECHNOLOGY AND RESEARCH
WAGHOLI, PUNE.

# TECHPRO-2K23

Organized by

**DEPARTMENT OF ELECTRICAL ENGINEERING**

(Accredited by NBA, NAAC & Recognized by AICTE)

This certificate is proudly presented to

Mr./~~Ms.~~ ABHISHEK SURENDRA SHARMA for project Titled REAL TIME TRAFFIC SIGNAL CONTROL of PUNE VIDYARTHI GRIHAS COLLEGE OF ENGINEERING for Participating in **TECHPRO-2K23** a National Level Project Competition Held on April 26, 2023.

Dr. N. N. Ghuge
CONVENER

Dr. T. K. Nagaraj
PRINCIPAL

**Sponsored by**

Electronics Study Centre | Net Protector NPAV Cyber Security | SYPROTECH SOLUTIONS Your Success Partner

## 9.5 Annexure E: Plagiarism Report

# Plagiarism Checker X - Report

Originality Assessment

**1%**

**Overall Similarity**

**Date:** May 30, 2023
**Matches:** 26 / 4887 words
**Sources:** 2

**Remarks:** Low similarity detected, check with your supervisor if changes are required.

**Verify Report:**
Scan this QR Code

v 8.0.11 - WML 4
FILE - FINAL REPORT.PDF