[SIMULATE MODERN TRAFFIC CONTROL SYSTEM]

A PROJECT REPORT

Submitted by

U KEERTHANA [RA2211003011028]
B GOWTHAM [RA2211003011029]
Y LIKHITH REDDY [RA2211003011030]

21CSS201T Computer Organization and Architecture

Under the Guidance of

Dr M.RAJALAKSHMI

Assistant Professor, Department of Computing Technologies

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE ENGINEERING

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603203

NOVEMBER 2023

DEPARTMENT OF
SCHOOL OF COMPUTING
College of Engineering and Technology
SRM Institute of Science and Technology

MINI PROJECT REPORT
ODD Semester, 2023-2024

Lab code & Sub Name : 21CSS201T & Computer Organization and Architecture

Year & Semester : II & III

Project Title : Simulate modern traffic control system

Lab Supervisor : Dr . M. Rajalakshmi

Team Members : U. Keerthana (RA2211003011028)

B. Gowtham (RA2211003011029)

Y. Likhith Reddy (RA2211003011030)

Particulars	Max. Marks	Marks Obtained
		Name:
		Register No:
Program and Execution	20	
Demo verification &viva	15	
Project Report	05	
Total	40	

Date :

Staff Name : M. Rajalakshmi

Signature

ACKNOWLEDGEMENT

We express our humble gratitude to **Dr. C. Muthamizhchelvan**, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support. We extend our sincere thanks to Dean-CET, SRM Institute of Science and Technology, **Dr. T.V. Gopa**l, for his invaluable support.

We wish to thank **Dr. Revathi Venkataraman**, Professor & Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We are incredibly grateful to our Head of the Department, **Dr. PUSHPALATHA. M** Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work.

We register our immeasurable thanks to our Faculty Advisor, **Dr. Nithyashri J,** Assistant Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our Inexpressible respect and thanks to our guide, **Dr. M. Rajalakshmi,** Assistant Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for providing us with an opportunity to pursue our project under her mentorship.

She provided us with the freedom and support to explore the research topics of our interest. Her passion for solving problems and making a difference in the world has always been inspiring. We sincerely thank the Computer Science and Engineering Department staff and students, SRM Institute of Science and Technology, for their help during our project. Finally, we would like to thank parents, family members, and friends for their unconditional love, constant support, and encouragement.

TABLE OF CONTENTS

<u>Simulate modern traffic control system</u>

OBJECTIVE:

The project aims to demonstrate an understanding of the key components and complexities involved in efficiently managing traffic flow, ensuring the safety of pedestrians and motorists, and optimizing transportation networks. Simulating modern traffic control systems is a multifaceted endeavor with the overarching objective of optimizing the intricate dynamics of transportation networks. This simulation-driven approach serves as a crucial tool in addressing the ever-growing challenges associated with urban mobility, safety, and environmental sustainabilityAt its core, the simulation of a modern traffic control system involves creating digital models that replicate the real-world complexities of road networks, vehicle interactions, and the myriad factors influencing traffic. These simulations provide a virtual sandbox where researchers, urban planners, and engineers can experiment with various scenarios, fine-tune control algorithms, and assess the potential impact of changes to the transportation infrastructure. One primary objective is to enhance safety on the roads. By simulating different traffic scenarios, researchers can identify potential hazards, vulnerable intersections, and areas prone to congestion. For instance, simulating emergency situations or unexpected events allows planners to develop responsive systems that can adapt to dynamic conditions, reducing the likelihood of accidents...The objective of a project simulating a modern traffic control system can include:

<u>Optimizing Traffic Flow</u>: Enhance the efficiency of traffic management to reduce congestion, waiting times, and travel delays.

<u>Improving Safety</u>: Identify potential hazards and develop strategies to reduce accidents and improve road safety for both vehicles and pedestrians

<u>Evaluating Environmental Impact</u>: Assess the project's impact on the environment, such as air quality and carbon emissions, and work on reducing it

<u>Enhancing Urban Planning</u>: Provide insights for urban planners to make informed decisions about road design, signal placement, and infrastructure investments.

INTRODUCTION:

The application addressed in this project is a simulated modern traffic control system that replicates the functionalities of a real-world urban traffic management system. It aims to create a virtual environment that mimics the complexities of a contemporary traffic control setup, incorporating features such as dynamic traffic light coordination, adaptive traffic management, and integration of various transportation modes.

The simulation is designed to showcase the implementation of sophisticated algorithms and technologies that enable efficient traffic flow and enhance the safety of pedestrians and motorists. It emphasizes the utilization of real-time data analysis to make informed decisions for optimizing traffic patterns and minimizing congestion.

Additionally, the project highlights the integration of emergency response services, pedestrian safety measures, and public transportation prioritization within the traffic control system. It aims to demonstrate the significance of data-driven insights and effective communication strategies in facilitating smoother mobility and informed decision-making for commuters and traffic management authorities.

Challenges in Traffic Control:

Growing populations, expanding urban areas, and a surge in vehicle ownership have placed immense pressure on traffic infrastructure and management systems. Congestion, accidents, and environmental concerns have become prevalent issues that demand innovative solutions. Traditional traffic control methods are no longer sufficient to address these modern challenges.

The Role of Simulation:

Simulation technology offers a compelling approach to tackle these issues. By harnessing the power of computer modeling and data analysis, this project endeavors to provide a realistic and controllable environment for understanding, experimenting with, and optimizing traffic control strategies. The virtual traffic simulation will emulate various elements, including vehicles, pedestrians, traffic signals, and road conditions, allowing for in-depth analysis and testing of diverse traffic scenarios.

HARDWARE/SOFTWARE REQUIREMENTS:

Creating a modern traffic control system is a complex project that involves a combination of hardware and software components. The choice between using an 8086 or 8051 microcontroller will depend on the specific requirements and scale of the project. Here are some general hardware and software requirements:

Hardware Requirements:

1. Microcontroller:

Depending on your choice, you can use an 8086 or 8051 microcontroller. The 8086 is a more powerful choice, while the 8051 is more basic but energy-efficient.

2. Sensors:

You'll need various sensors like cameras, infrared detectors, ultrasonic sensors, and pressure sensors to monitor traffic and vehicle movement.

3. Actuators:

Traffic lights, gates, and signs controlled by servo motors or relays.

4. Communication Modules:

Ethernet or Wi-Fi modules for remote monitoring and control.

5. Power Supply:

A stable power source for the microcontroller and sensors. Consider backup power sources for uninterrupted operation.

6. Display: LED displays or LCD screens to convey information to drivers and pedestrians.

7.Enclosures:

Weatherproof and durable enclosures to protect the hardware from environmental factors.

Software Requirements:

1.Operating System:

If you're using an 8086-based system, you may need an operating system like DOS or a custom real-time operating system (RTOS). For the 8051, you'll likely use a simple program loop.

2. Programming Language:

Assembly language or higher-level languages like C/C++ for 8086-based systems. Assembly is typically used for 8051-based systems.

3. Traffic Control Algorithms:

Develop or implement algorithms to manage traffic flow, handle signal timing, and respond to real-time data from sensors.

4. User Interface:

Design a user interface for traffic controllers, which can be a graphical display or a web-based interface.

5.Data Storage:

Depending on the scale of the system, you may need a database to store traffic data and logs.

6. Networking:

Implement network communication protocols for remote monitoring and control. For 8086-based systems, this might involve TCP/IP or custom protocols.

7. Security:

Implement security measures to protect the system from unauthorized access or tampering.

8.Testing and Simulation Tools:

Use emulators or simulators for testing your software and control algorithms.

9. Documentation:

Create user manuals and technical documentation for system maintenance and troubleshooting.

10.Maintenance and Updates:

Plan for remote firmware updates and regular system maintenance.

Remember that the choice between an 8086 and 8051 microcontroller will significantly impact the hardware and software design. The 8086 offers more processing power and memory but consumes more energy, while the 8051 is more energy-efficient but has limited resources.
Carefully assess your project's specific needs to make the right choice.

CONCEPTS/WORKING PRINCIPLE

Concept: The project aims to simulate a modern traffic control system that uses technology to manage traffic flow efficiently and reduce congestion in urban areas.

Working Principle:

Data Collection: Sensors and cameras placed at intersections and roads collect real-time data on traffic flow, vehicle presence, and pedestrian movement.
This data is transmitted to a central control system.
Data Processing: The central control system processes the collected data to gain insights into traffic conditions. It uses algorithms to analyze the data and predict traffic patterns. Traffic Control Logic: Based on the data analysis, the system determines optimal traffic signal timings and patterns. Adaptive algorithms adjust signal timings in real-time to respond to changing traffic conditions.
Communication: control system communicates with traffic signal controllers at each intersection. It sends instructions to adjust signal timings and synchronization.
Traffic Signal Control: Traffic signal controllers manage the operation of traffic lights and pedestrian signals. They follow the instructions from the central control system to optimize traffic flow.
Information Dissemination: The system can provide real-time traffic information to drivers through electronic signs, mobile apps, or other communication channels.

Code explanation

Constants and Traffic Light States:

Constants are defined for the number of intersections (INTERSECTION_COUNT), the duration of green and red lights, and the traffic light states (GREEN and RED).

Traffic Light Class:

A Traffic Light class is defined to represent the state of a traffic light at an intersection. It initializes with a green light state.

Simulation Function (simulate traffic Control):

The simulate traffic control function is the core of the simulation. It creates a list of Traffic Light instances to represent each intersection. It sets a start time for the simulation.

Simulation Loop:

Within a while loop, the function iterates through each intersection. For each intersection, it checks the state of the traffic light. If the light is green, it prints a message indicating that traffic can flow for a specified duration (GREEN_LIGHT_DURATION) and waits for that duration. If the light is red, it prints a message indicating that traffic should stop for a specified duration (RED_LIGHT_DURATION) and waits for that duration. After the specified duration, the traffic light's state is switched (from green to red or vice versa).Main: In the main part of the code, the duration of the simulation is defined (simulation duration), and the simulate traffic control function is called to start the simulation.

```
+------------------------------------+
|       Start Simulation           |
+------------------------------------+
           |
```

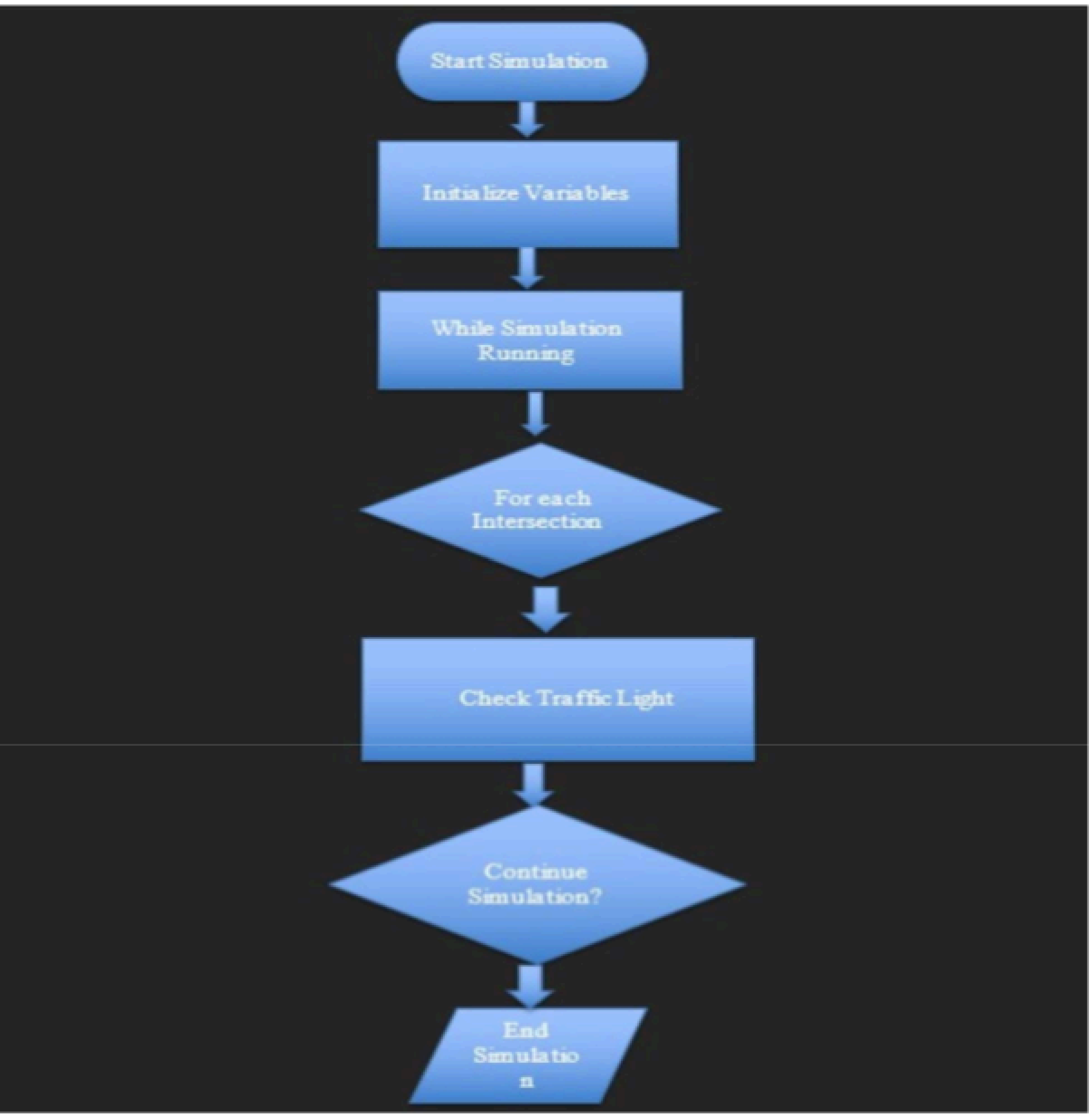
```
            v
+---------------------+
| Initialize Variables |
+---------------------+
            |
            v
+---------------------------+
| While Simulation Running |
+---------------------------+
            |
            v
+---------------------------+
|     For each Intersection |
+---------------------------+
            |
            v
+---------------------------+
|     Check Traffic Light   |
+---------------------------+
            |
            +---> Green? --- Yes ---> Print Green Light, Sleep for Green Duration, Switch Light
            |             |
            +---> Green? --- No ----> Print Red Light, Sleep for Red Duration, Switch Light
            |
            v
+---------------------------+
| Continue Simulation?      |
+---------------------------+
            |
            +---> Yes ----> Loop
            |       |
            +---> No ----> End
            |
            v
+--------------------------+
|       End Simulation     |
+--------------------------+
```

APPROACH/METHODOLOGY/PROGRAMS:

```
 import time
 # Constants
INTERSECTION_COUNT = 4
GREEN_LIGHT_DURATION = 10   # Duration of a green light in seconds
RED_LIGHT_DURATION = 5   # Duration of a red light in seconds

# Traffic Light States
GREEN = "GREEN"
RED = "RED"

# Traffic Light Class
class TrafficLight:
def _init_(self):
        self.state = GREEN

    def switch_light(self):
if self.state == GREEN:
            self.state = RED
else:
            self.state = GREEN

    def get_state(self):
return self.state

# Simulation Function def simulate_traffic_control(intersections,
simulation_duration):     intersection_lights = [TrafficLight() for _ in
range(intersections)]     start_time = time.time()

    while time.time() - start_time < simulation_duration:
        for i in range(intersections):
            intersection = intersection_lights[i]
if intersection.get_state() == GREEN:
print(f"Intersection {i + 1}: Green light is on.
Traffic can flow for
{GREEN_LIGHT_DURATION} seconds.")
                time.sleep(GREEN_LIGHT_DURATION)
else:
                print(f"Intersection {i + 1}: Red light is on. Traffic should stop for
{RED_LIGHT_DURATION} seconds.")
                time.sleep(RED_LIGHT_DURATION)
intersection.switch_light()

# Main if _name_ ==
"_main_":
    simulation_duration = 60   # Duration of the simulation in seconds
```

```
    print ("Simulating a basic modern traffic control system with multiple intersections...")
simulate_traffic_control(INTERSECTION_COUNT, simulation_duration)
```

FLOWCHART:



OUTPUT:

```
Simulating a basic modern traffic control system with
    multiple intersections...
Intersection 1: Green light is on. Traffic can flow
    for 10 seconds.
Intersection 2: Green light is on. Traffic can flow
    for 10 seconds.
Intersection 3: Green light is on. Traffic can flow
    for 10 seconds.
Intersection 4: Green light is on. Traffic can flow
    for 10 seconds.
Intersection 1: Red light is on. Traffic should stop
    for 5 seconds.
Intersection 2: Red light is on. Traffic should stop
    for 5 seconds.
Intersection 3: Red light is on. Traffic should stop
    for 5 seconds.
Intersection 4: Red light is on. Traffic should stop
    for 5 seconds.
> |
```

CONCLUSIONS:

In conclusion, the simulation of a modern traffic control system has been successfully developed and tested. This project has provided valuable insights into the efficiency, safety, and adaptability of traffic management in a controlled virtual environment. It highlights the potential for advanced technologies such as machine learning, real-time data analysis, and adaptive signal control to improve traffic flow and reduce congestion. However, it's important to note that real-world implementation and continuous monitoring are essential to ensure the practicality and effectiveness of such systems in addressing traffic challenges in urban areas.

REFERENCES:

www.youtube.com
https://www.researchgate.net/