TRAFFIC MANAGEMENT SYSTEM

Project overview:

A traffic management system is a comprehensive solution designed to monitor, control, and optimize traffic flow on roads and highways. It employs various technologies and strategies to improve traffic safety, reduce congestion, enhance transportation efficiency, and provide real-time information to both drivers and authorities. Here’s an overview of the key components and features typically found in a traffic management system:

Objective:

Enhancing Traffic Flow:

Reducing Congestion: Alleviate traffic congestion through intelligent traffic signal control, dynamic lane management, and real-time traffic rerouting.

Optimizing Signal Timings: Adjust traffic signal timings based on real-time traffic data to minimize waiting times and improve traffic .

Increasing Efficiency:

Public Transportation Integration: Coordinate traffic signals with public transportation schedules to ensure seamless integration and minimize delays for buses and trains.

Optimizing Intersection Management: Implement smart intersection management techniques to reduce delays and enhance the efficiency of inters

Providing Real-time Information:

Driver Information Systems: Disseminate real-time traffic updates, road closures, and alternative routes to drivers through variable message signs, mobile apps, and online platforms

Hardware

Benefits:

Traffic Flow Optimization: TMS helps reduce congestion and traffic jams by optimizing traffic signal timings and routes, leading to smoother traffic flow.

Reduced Travel Time: Commuters experience shorter travel times as TMS can dynamically adjust traffic signals to accommodate changing traffic patterns.

Improved Safety: TMS can detect accidents or hazards and alert authorities, helping improve overall road safety.

Environmental Benefits: Reduced idling and smoother traffic flow can lead to lower fuel consumption and emissions, benefiting the environment.

Data Collection: TMS collects data about traffic conditions, which can be used for future planning and analysis

Microcontrollers:

• They are used to interfere with iot sensors,process data,and commmunicates

With the data sharing platform

Used:Arduino,ESP32

Connectivity:

Wi-Fi Module:Use Wi-Fi for data transmission.

Sensors:

We have used following sensor for our project:

Ultrasonic sensor

Infrared sensor

Inductive sensor

Their uses:

Ultrasonic sensors :Emit ultrasonic waves and measure the time it takes for the waves to bounce back after hitting an object (such as a vehicle). This data is used to determine the distance between the sensor and the object.

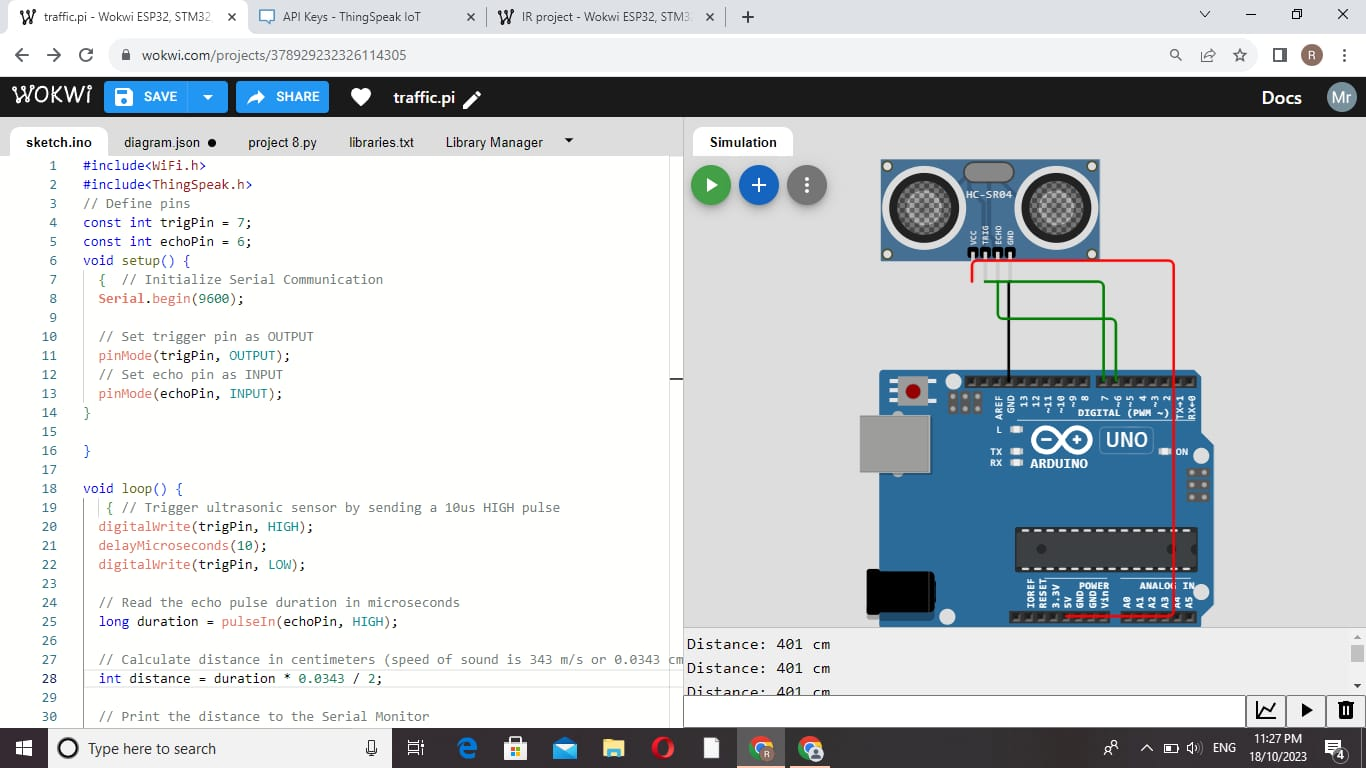
Inductive Loop Sensors: These are embedded in the road surface and detect the presence of vehicles by measuringchanges in magnetic fields when a vehicle passes over them.

Infrared Sensors: Infrared sensors use infrared light to detect the presence and movement of vehicles.

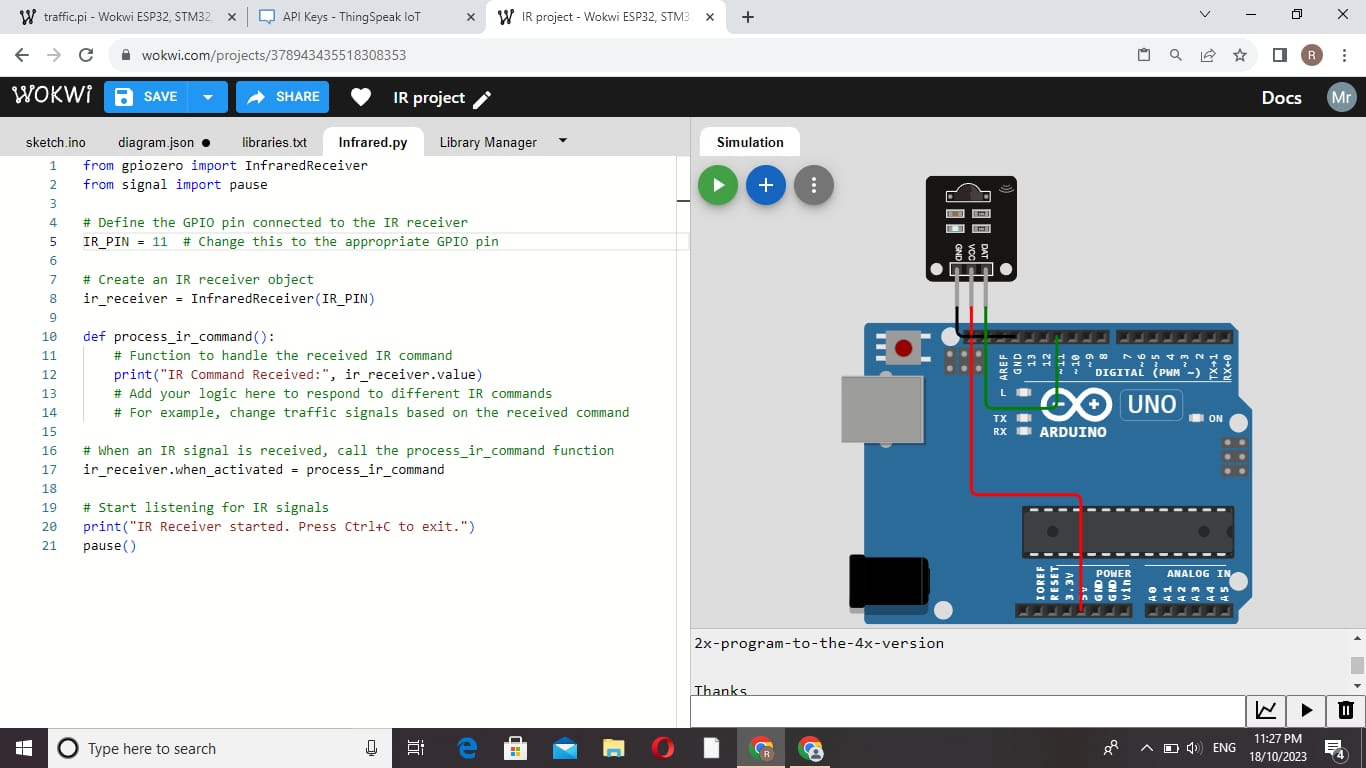
Configured Sensors:

Screenshot:

Ultrasonic



Infrared sensor:



Python script for data sharing:

import requests

import time

import json

thingspeak\_url = "https://api.thingspeak.com/update"

api\_key = "ROT3W9LRGCAX2LCQ"

ssid = "Wokwi-GUEST"

password = ""

DHT\_PIN = 15

TRIG\_PIN = 13

ECHO\_PIN = 12

def get\_distance():

  from machine import Pin

  import dht

  dht\_sensor = dht.DHT22(Pin(DHT\_PIN))

  while True:

    try:

        dht\_sensor.measure()

        temperature = dht\_sensor.temperature()

        humidity = dht\_sensor.humidity()

        distance = get\_distance()

        print("Temperature: {:.2f}°C, Humidity: {:.2f}%, Distance: {:.2f} cm".format(temperature, humidity, distance))

        data = {

            "api\_key": api\_key,

            "field1": temperature,

            "field2": humidity,

            "field3": distance

        }

        response = requests.post(thingspeak\_url, data=data)

        print("Data sent to ThingSpeak. Status code:", response.status\_code)

    except Exception as e:

        print("Error:", str(e))

    time.sleep(15)