REPORT

ON

Traffic Intensity Calculator

SUBMITTED

TO

VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY, PUNE

For the PBL of Digital Electronics

IN

DEPARTMENT OF ELECTRONICS AND TELECOMMUNICATION

AY 2023-24

Semester II

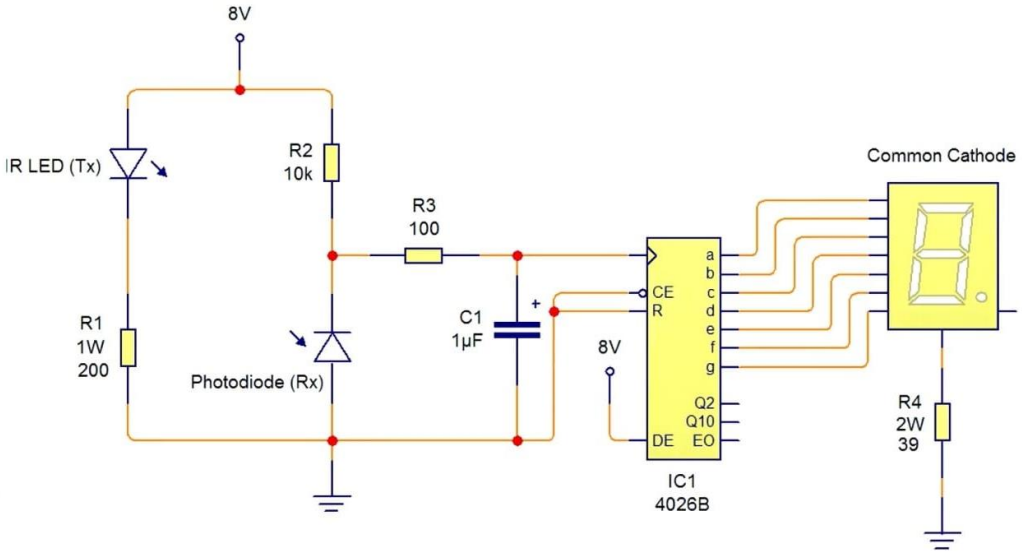
BY

P. R. No.	Roll No.	Name of Student	Mobile No.	VIIT Email ID
22311251	101145	Tanushri Chetan Rajput	8446029351	tanushri.22311251@viit.ac.in
22311496	101149	Atharva Rajendra Joshi	9527043551	atharva.22311496@viit.ac.in
22311517	101153	Revati Tushar Aute	8208092760	revati.22311517@viit.ac.in
22311608	101158	Manas Girish Kulkarni	7972470825	manas.22311608@viit.ac.in

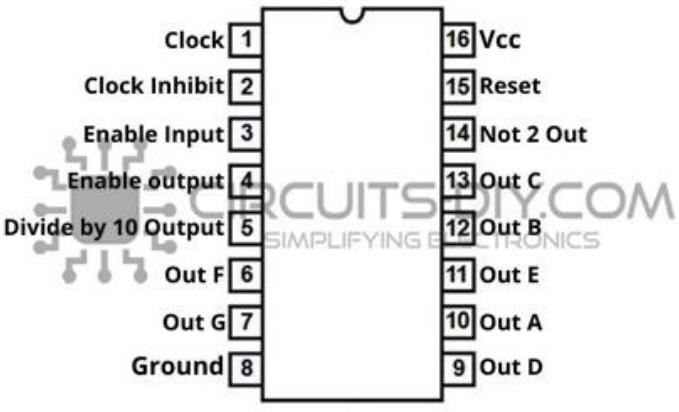
Class: F. Y. B. Tech. **Division: A** **Batch: A3**

Batch Incharge
Mrs. Rupa Kawchale

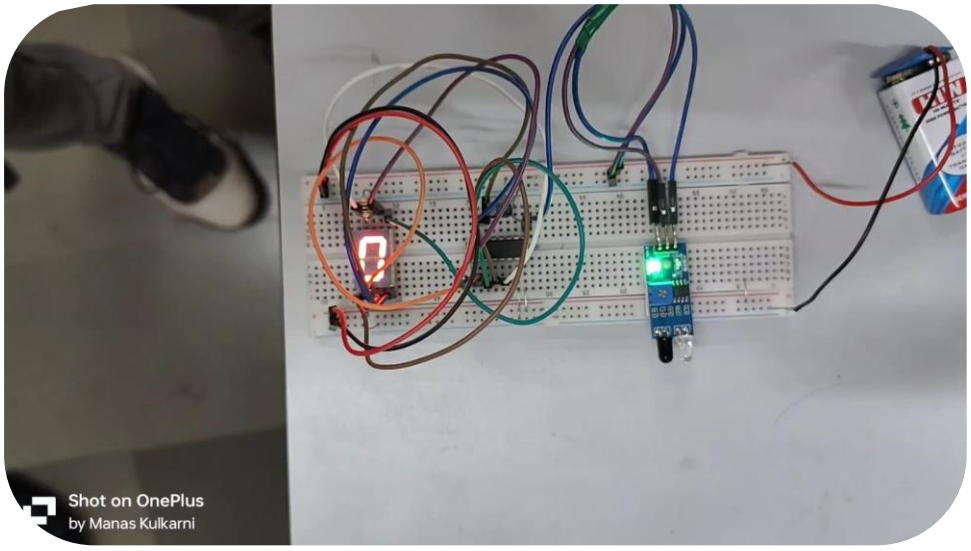
Circuit Diagram

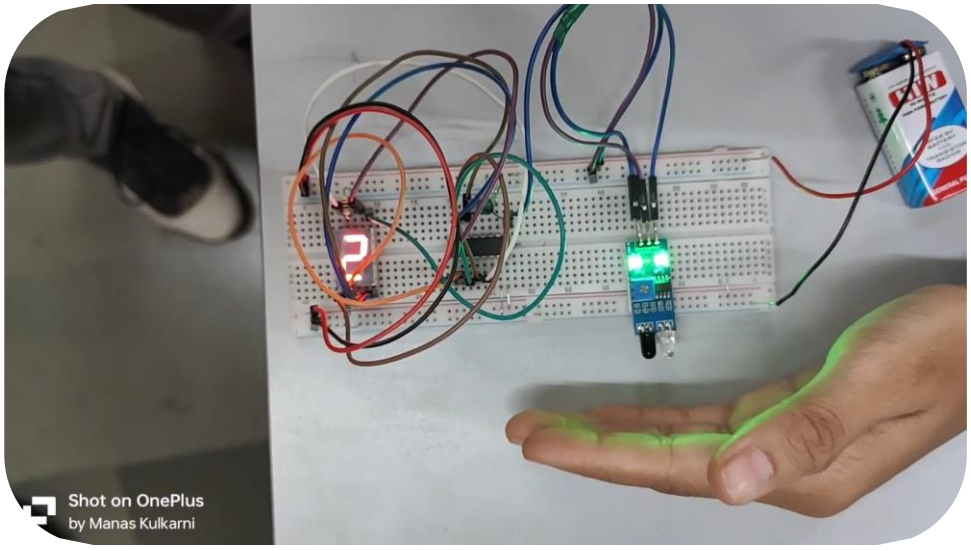


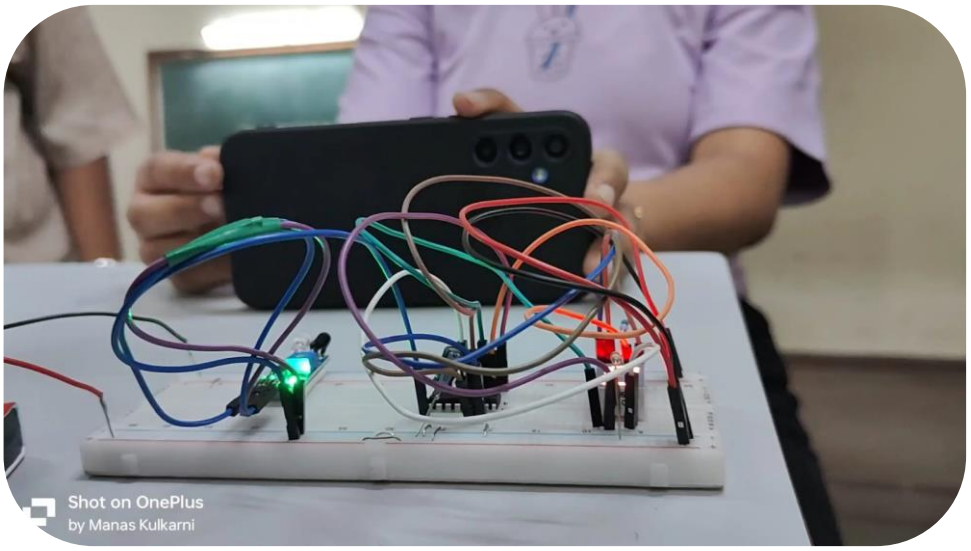
CD4026 Pinout Diagram



Images of Model



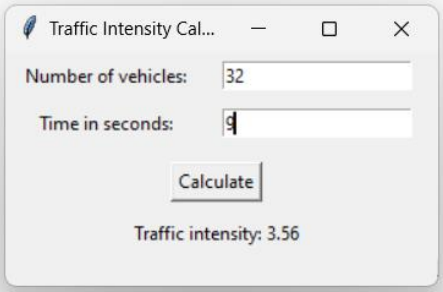




Code (Python – Tkinter)

```
import tkinter as tk

def calculate_traffic_intensity():
    num_vehicles = int(entry_vehicles.get())
    time_seconds = int(entry_time.get())
    intensity = num_vehicles / time_seconds
    result_label.config(text="Traffic intensity: {:.2f}".format(intensity))

# Create main window
root = tk.Tk()
root.title("Traffic Intensity Calculator")

# Create input fields
label_vehicles = tk.Label(root, text="Number of vehicles:")
label_vehicles.grid(row=0, column=0, padx=10, pady=5)
entry_vehicles = tk.Entry(root)
entry_vehicles.grid(row=0, column=1, padx=10, pady=5)

label_time = tk.Label(root, text="Time in seconds:")
label_time.grid(row=1, column=0, padx=10, pady=5)
entry_time = tk.Entry(root)
entry_time.grid(row=1, column=1, padx=10, pady=5)

# Create calculate button
calculate_button = tk.Button(root, text="Calculate",
command=calculate_traffic_intensity)
calculate_button.grid(row=2, column=0, columnspan=2, pady=10)

# Create result label
result_label = tk.Label(root, text="")
result_label.grid(row=3, column=0, columnspan=2)

root.mainloop()
```

Output



Working of Project

CD4026 IC:

- The CD4026 is a **CMOS counter IC** with a built-in **7-segment output driver**.
- It is capable of driving **common cathode LED displays** directly.
- The IC has a **5-stage Johnson decade counter** and an output decoder.
- The output decoder transforms the Johnson code into a **7-segment decoded output**.
- The IC increments its count each time it receives a **high clock pulse**.
- In our project, this pulse is generated when a vehicle interrupts the infrared beam of the IR module.
- The **clock inhibit pin (pin 2)** is held low (ground/0V) so that the clock signals can be sent to the IC.
- The **Enable Input pin (pin 3)** is made high (+5V) to activate the output pins.
- The IC can work from **3V to 15V**, but we typically power it with +9V to the Vdd/Vcc pin.
- The **Ground/Vss pin** is connected to the ground.
- The **7-segment pins** will increment the count by one number each time when the clock pin (pin 1) is made high.

Python Tkinter:

- We have used Python Tkinter to create a **simple interface** that displays the current count of vehicles from the CD4026.
- By dividing the total count of vehicles by the elapsed time (in seconds), we calculate the **vehicle intensity** (vehicles per second).
- This value is then displayed on the Tkinter interface.

Project Integration:

- We have developed a system that not only counts vehicles but also calculates and displays vehicle intensity.
- This is achieved by integrating the hardware components (CD4026 IC and IR module) with the Python Tkinter interface.
- This combination of hardware and software components makes our project a robust and efficient solution for vehicle counting and intensity calculation.