# MINI PROJECT

# TRAFFIC MANAGEMENT SYSTEM

## OBJECTIVE:

The objective of this project is to develop a traffic management system by using sensors, raspberry pi and python programming to manage traffic situations on road efficiently and also decrease carbon emissions.

## ABSTRACT:

In today's world as the number of vehicles owned by a single family increasing as well as the standard living, resulting in more number of vehicles bought every year which are out on our streets resulting in traffic jams on roads. We have the current system of traffic signals where there is only a fixed timer for every lane regardless of the amount of vehicles on the lane. So we have developed a smart system where we will use ultrasonic sensor to detect which lane has more traffic and change the timer accordingly to ease traffic on that lane and the same for all the lanes at the intersection which will help in easing traffic based on the traffic on individual lanes. Also it will help in decreasing pollution as a vehicle is more polluting the environment when it has to stop and start and then run again once then to just keep driving it which will be facilitated by the increased timer resulting in less number of vehicles stopping at the traffic resulting in less pollution when seen at every intersection as there will be a huge number of vehicles which won't be stopping and then running which will help in lesser pollution too. This is how our smart traffic management system is useful as it eases traffic and also helps in reducing pollution.

## HARDWARE/SOFTWARE REQUIREMENTS:

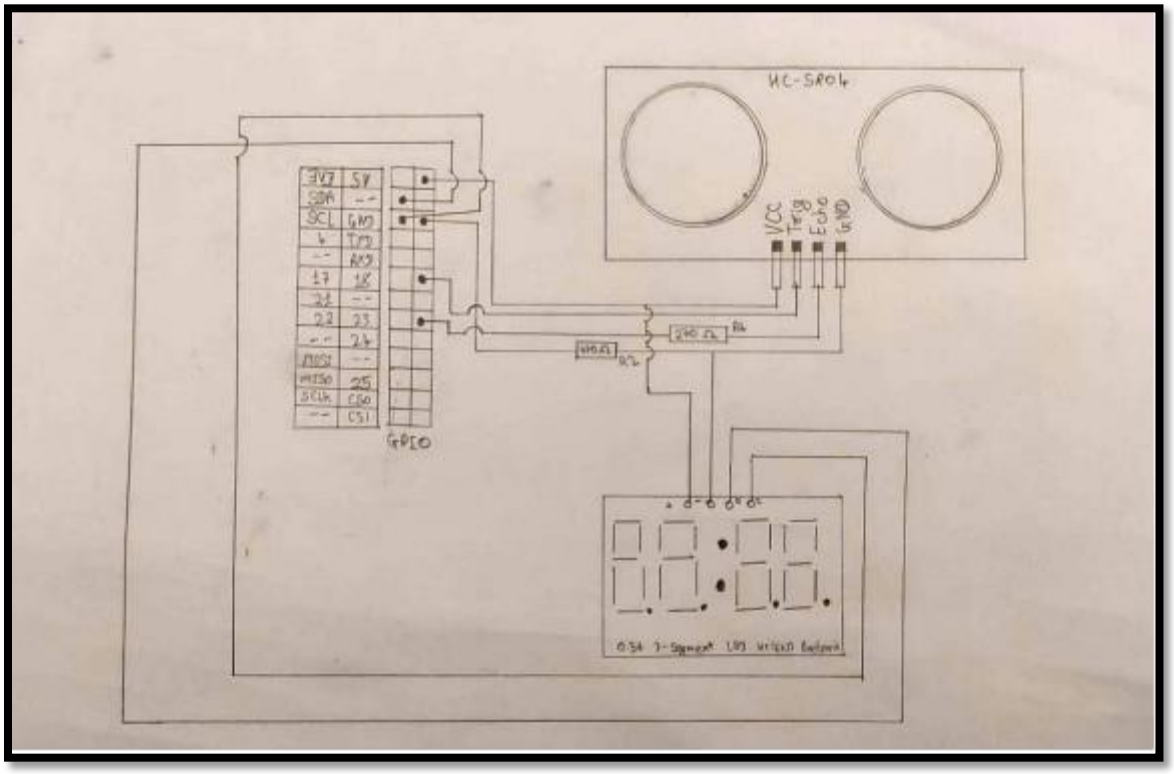
Hardware Requirement:

- Raspberry Pi Board
- Ultrasonic Sensor (HC-SR04)
- 0.56'' 7-segment LED HT16K33 Backpack
- Resistors (270 Ω, 470 Ω)
- Wires

Software Requirement:

- https://create.withcode.uk/ (for simulation purpose)
- https://app.diagrams.net/ (for making flowchart)

## BLOCK DIAGRAM / CONNECTION DIAGRAM :

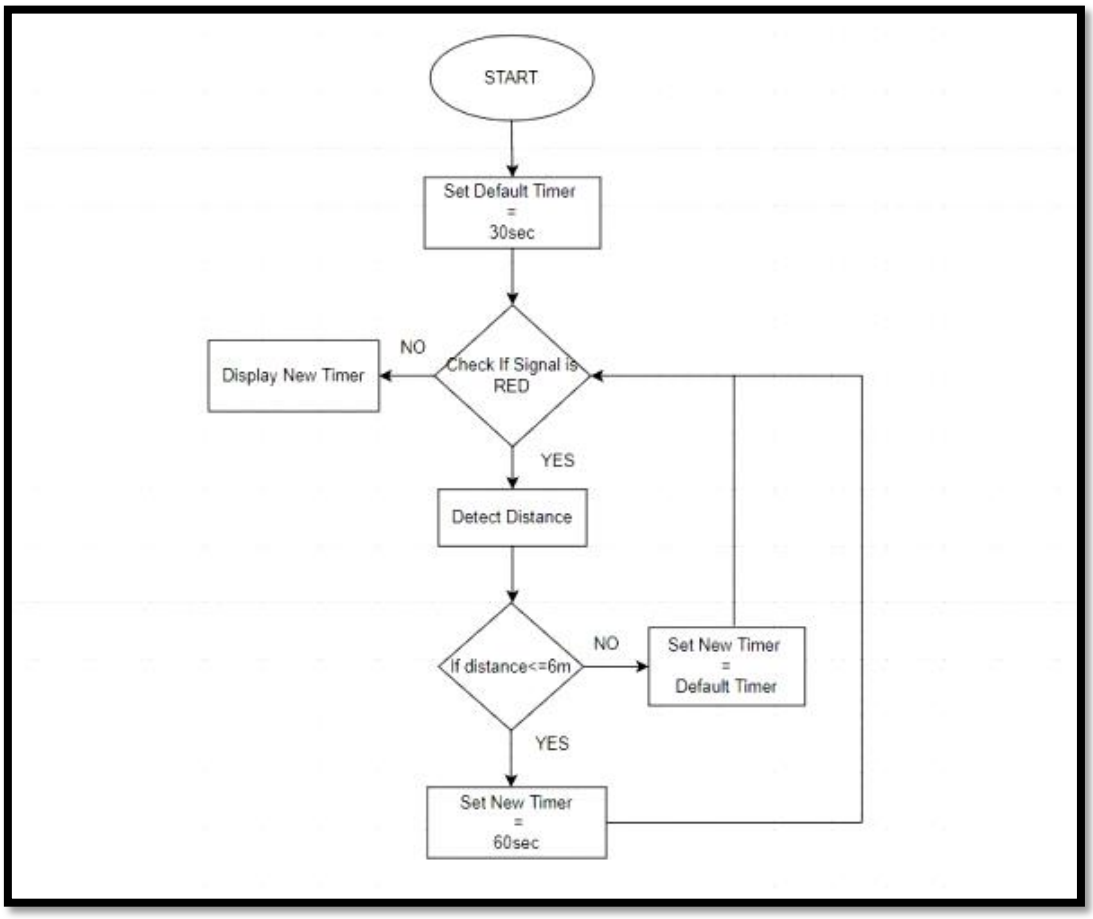


- The ultrasonic Sensor has four pins named Vcc, Trig, Echo and GND.
- The Vcc pin is connected to 5V GPIO of raspberry board so through this it will receive its input voltage .
- The Trig pin being connected to 18 GPIO pin and Echo pin is connected in series with 270Ω resistor and 23 GPIO pin of board.
- The GND pin of sensor is connected in series with 470Ω resistor and GND pin of board.
- The 7 segment LED display has four pins as positive terminal, negative terminal, D and C pin. The positive terminal of LED is connected in parallel with 5V GPIO pin of raspberry board.
- The negative terminal is connected in parallel with GND pin of board.
- The D pin is connected to SDA pin and C pin is connected to SCL pin od board respectively.

**ALGORITHM :**

- First we will start the system and import the required modules.
- We will set GPIO mode to GPIO.BOARD.
- Set pin 12 as output pin.
- Create a function checkDist() and return int input from user.
- newTimer=0
- Create another function named printTimer() with argument as newTimer.
- Inside the function run a for loop with range 0 to neewTimer.
- Inside the loop print(newTimer) and decrement it by 1 and sleep(0.1)
- Now the functions are completed and declare the main loop as while(1).
- Set defaultTimer=30.
- Create if(input("Enter the Red signal is ON or Off:")=="on"):
- For if true condition store distance from the checkDist() function in distance variable.
- Then set pin 12 as HIGH.
- Check if distance is less than or equal to 6am.
- If yes then set newTimer = 60 else store defaultTimer into newTimer.
- For else condition set 12 pin as LOW and call printTimer(newTimer) function.
- At last GPIO.cleanup().

## FLOW CHART:

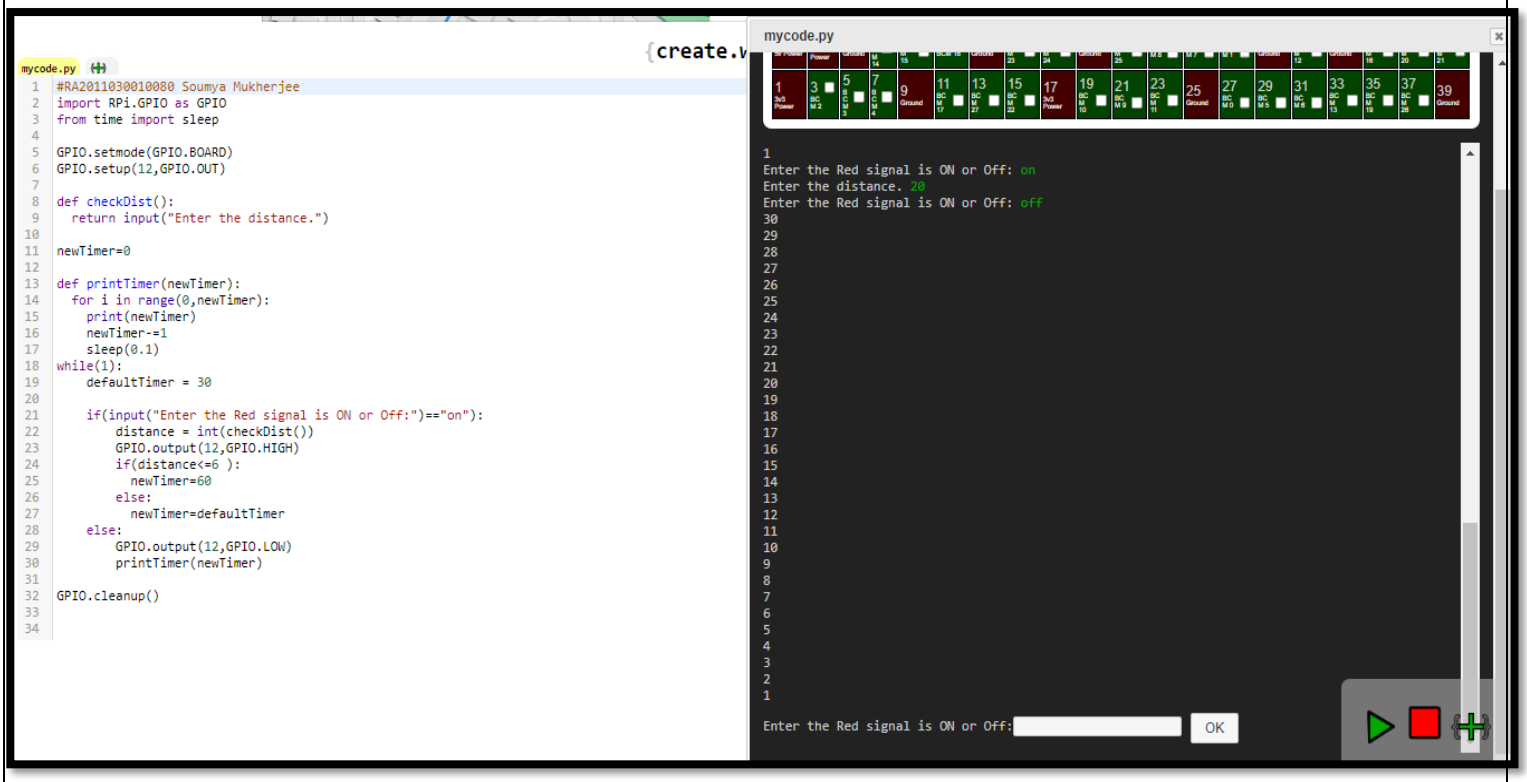



- First the system will start and we will specify defaultTimer as 30 seconds.
- Then we check if signal is red or not.
- If yes then we find the distance measured by ultrasonic sensor.
- Then we check if distance is less than or equal to 6 meters.
- If yes we set newTimer equal to 60 seconds.
- If no we set newTimer equal to defaultTimer.
- After checking for both case we go to the first if and check if, signal is red or not, and if it is off then it will start displaying timer as per the newTimer value and during that time other components will be turned off.

**PROGRAM :**

```
#RA2011030010080 SOUMYA MUKHERJEE

import RPi.GPIO as GPIO
from time import sleep

GPIO.setmode(GPIO.BOARD)
GPIO.setup(12,GPIO.OUT)

def checkDist():
        return input("Enter the distance.")

newTimer=0

def printTimer(newTimer):
        for i in range(0,newTimer):
                print(newTimer)
                newTimer-=1
                sleep(0.1)
while(1):
    defaultTimer = 30

if(input("Enter the Red signal is ON or Off:")=="on"):
          distance = int(checkDist())
          GPIO.output(12,GPIO.HIGH)
          if(distance<=6 ):
                newTimer=60
          else:
                newTimer=defaultTimer
else:
  GPIO.output(12,GPIO.LOW)
  printTimer(newTimer)

GPIO.cleanup()
```

## OUTPUT :




## REAL TIME CONSTRAINTS:

- We have assumed that we will place the ultrasonic sensor at 30m from the traffic signal to detect the presence of vehicle up to that point from traffic signal.
- We are assuming that the lane is 6m in width therefore we are using the condition if distance<=6m.
- We are assuming that by adding 30sec we will be able to successfully ease traffic by giving more time and decreasing the number of vehicles in lane efficiently.
- Since it is a simulation so we are taking the distance and traffic signal state, i.e , if red light is on or not.

## CONCLUSION :

- We have successfully designed and implemented smart traffic management system with the help of raspberry pi and ultrasonic sensor.
- We believe that by implementing this system in real world, we can ease traffic issues.
- Also the vehicles will be used efficiently which will eventually lead to less carbon emissions and protecting the mother nature.

**DEPARTMENT OF**
**ELECTRONICS AND COMMUNICATION ENGINEERING**
**College of Engineering and Technology**
**SRM Institute of Science and Technology**

MINI PROJECT REPORT
ODD Semester, 2022-2023

Lab code & Sub Name : 18ECO109J - Embedded System Design using Raspberry Pi

Year & Semester : 3rd / 5th

Project Title : TRAFFIC MANAGEMENT SYSTEM

Lab Supervisor : Dr. J.Subhashini

Team Members : 1. Soumya Mukherjee (RA2011030010080)

| **Particulars** | **Max.Marks** | **MarksObtained** |
|---|---|---|
| | | **Name: SOUMYA MUKHERJEE** |
| | | **Register No : RA2011030010080** |
| Program and Execution | 20 | |
| Demo verification & viva | 15 | |
| Project Report | 05 | |
| **Total** | **40** | |