Traffic management system

IOT_phase 3

TRAFFIC MANAGEMENT SYSTEM:

Phase 3: Development part 1

Introduction:

Identifying the vehicles using camera and sensors and as per that the lights will glow .

It reduces the traffic and accidents due to traffic jam .

On working as per the following programs, the traffic jam must be reduced.


MOJO

PYTHON program using Arduino UNO:

```
#include <IRremote.h>
#define signal 11
int R1=4 , G1=5 , Y1=12 , R2=6 , G2=7 ,Y2=13 , R3=8 , G3=9 ,Y3=10  , R4=2 , G4=3 ,Y4=A4 ;

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  IrReceiver.begin(signal);
  for(int i=2 ; i<=9 ; i++){
     pinMode(i, OUTPUT);
  }
}
 void Alert(int code){
  if(code==48){
     digitalWrite(R1, LOW);
     digitalWrite(G1, HIGH);
     digitalWrite(Y1, LOW);
     digitalWrite(R2, LOW);
     digitalWrite(G2, LOW);
     digitalWrite(Y2, HIGH);
```

```
digitalWrite(R3, HIGH);
    digitalWrite(G3, LOW);
    digitalWrite(Y3, LOW);
    digitalWrite(G1, LOW);
    digitalWrite(R4, HIGH);
    digitalWrite(G4, LOW);
    digitalWrite(Y4, LOW);
    delay(3000);
  }
  else if(code==24){
    digitalWrite(R1, HIGH);
    digitalWrite(G1, LOW);
    digitalWrite(Y1, LOW);
    digitalWrite(R2, LOW);
    digitalWrite(G2, HIGH);
    digitalWrite(Y2, LOW);
    digitalWrite(R3, LOW);
    digitalWrite(G3, LOW);
    digitalWrite(Y3, HIGH);
    digitalWrite(R4, HIGH);
    digitalWrite(G4, LOW);
    digitalWrite(Y4, LOW);
    delay(3000);
  }
```

```
else if(code==122){
   digitalWrite(R1, HIGH);
   digitalWrite(G1, LOW);
   digitalWrite(Y1, LOW);
   digitalWrite(R2, HIGH);
   digitalWrite(G2, LOW);
   digitalWrite(Y2, LOW);
   digitalWrite(R3, LOW);
   digitalWrite(G3, HIGH);
   digitalWrite(Y3, LOW);
   digitalWrite(R4, LOW);
   digitalWrite(G4, LOW);
   digitalWrite(Y4, HIGH);
   delay(3000);
 }
 else if(code==16){
   digitalWrite(R1, LOW);
   digitalWrite(G1, LOW);
   digitalWrite(Y1, HIGH);
   digitalWrite(R2, HIGH);
   digitalWrite(G2, LOW);
   digitalWrite(Y2, LOW);
```

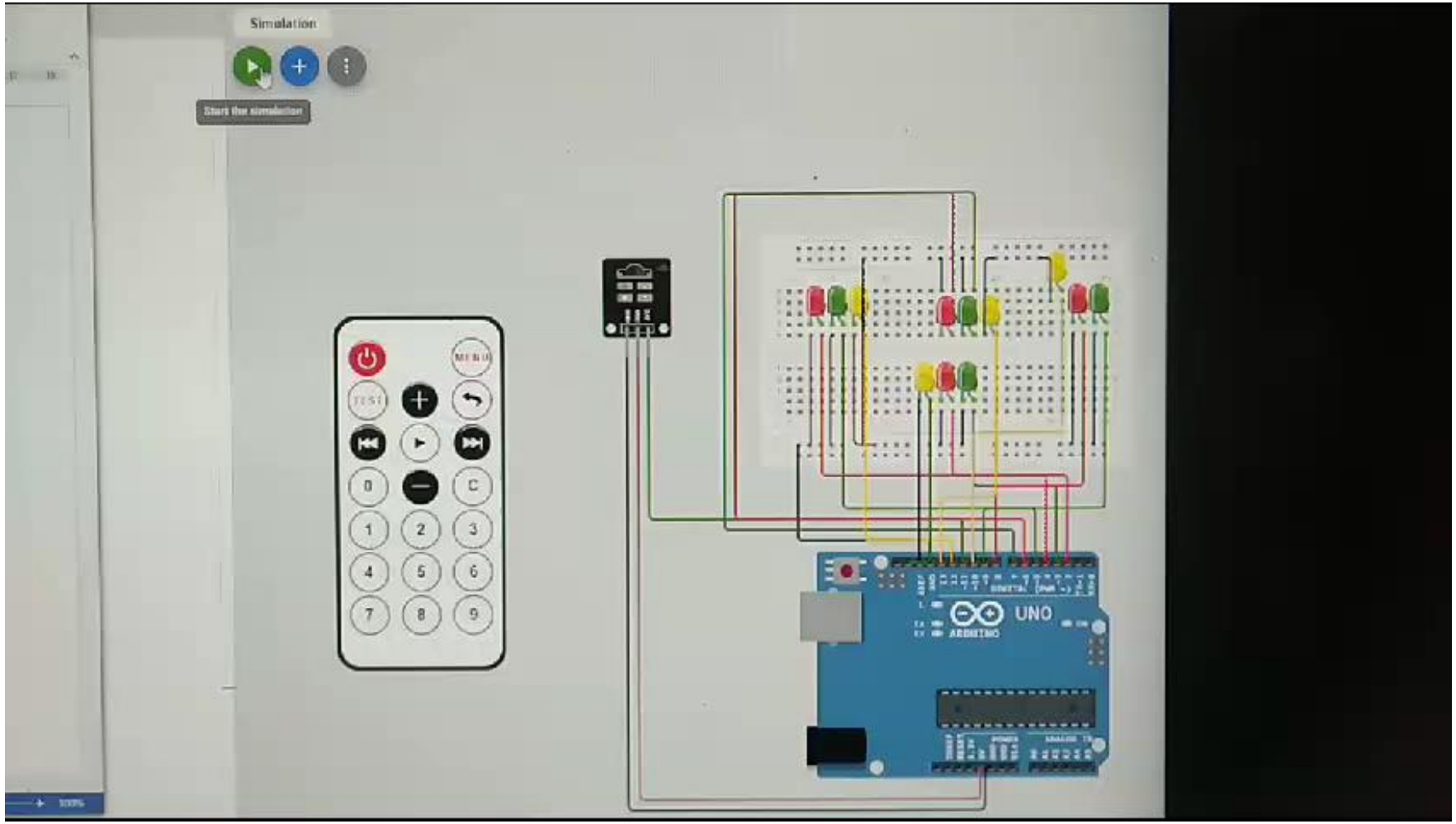
```
  digitalWrite(R3, HIGH);
     digitalWrite(G3, LOW);
     digitalWrite(Y3, LOW);
     digitalWrite(R4, LOW);
     digitalWrite(G4, HIGH);
     digitalWrite(Y4, LOW);
     delay(3000);
    }
}
int code=0;
void loop() {
  // put your main code here, to run repeatedly:
  if(IrReceiver.decode()){
    IrReceiver.resume();
    code=IrReceiver.decodedIRData.command;
    Alert(code);
  }
  digitalWrite(R1, LOW);
  digitalWrite(G1, HIGH);
  digitalWrite(Y1, LOW);
  digitalWrite(R2, LOW);
  digitalWrite(G2, LOW);
```

```
digitalWrite(Y2, HIGH);
digitalWrite(R3, HIGH);
digitalWrite(G3, LOW);
digitalWrite(Y3, LOW);
digitalWrite(R4, HIGH);
digitalWrite(G4, LOW);
digitalWrite(Y4, LOW);
delay(3000);
if(IrReceiver.decode()){
  IrReceiver.resume();
  code=IrReceiver.decodedIRData.command;
  Alert(code);
}
digitalWrite(R1, HIGH);
digitalWrite(G1, LOW);
digitalWrite(Y1, LOW);
digitalWrite(R2, LOW);
digitalWrite(G2, HIGH);
digitalWrite(Y2, LOW);
digitalWrite(R3, LOW);
digitalWrite(G3, LOW);
digitalWrite(Y3, HIGH);
```

```
digitalWrite(R4, HIGH);
digitalWrite(G4, LOW);
digitalWrite(Y4, LOW);
delay(3000);
if(IrReceiver.decode()){
  IrReceiver.resume();
  code=IrReceiver.decodedIRData.command;
  Alert(code);
}
digitalWrite(R1, HIGH);
digitalWrite(G1, LOW);
digitalWrite(Y1, LOW);
digitalWrite(R2, HIGH);
digitalWrite(G2, LOW);
digitalWrite(Y2, LOW);
digitalWrite(R3, LOW);
digitalWrite(G3, HIGH);
digitalWrite(Y3, LOW);
digitalWrite(R4, LOW);
digitalWrite(G4, LOW);
digitalWrite(Y4, HIGH);
delay(3000);
```

```
 if(IrReceiver.decode()){
    IrReceiver.resume();
    code=IrReceiver.decodedIRData.command;
    Alert(code);
  }
  digitalWrite(R1, LOW);
  digitalWrite(G1, LOW);
  digitalWrite(Y1, HIGH);
  digitalWrite(R2, HIGH);
  digitalWrite(G2, LOW);
  digitalWrite(Y2, LOW);
  digitalWrite(R3, HIGH);
  digitalWrite(G3, LOW);
  digitalWrite(Y3, LOW);
  digitalWrite(R4, LOW);
  digitalWrite(G4, HIGH);
  digitalWrite(Y4, LOW);
  delay(3000);
}
```

Output(video):



PROGRAM 2 using Raspbeery pi 3:

```
import RPi.GPIO as gp
from time import sleep,
gp.setup(12,gp.IN)
 gp.setup(32,gp.OUT)
 gp.setup(36,gp.OUT)
 from picamera import PiCamera camera = PiCamera() time.
Traffic lights=R1=14  ,  G1= 15 , Y1= 23 , R2=8 , G2= 7 , Y2=12  , R3=5  ,Y3= 20 , G3=26, R4=21, G4=3, Y4= 13;
While true:
{
If gp.input(sensor)
print “Vehicle Detected“
Camera.start_recording(camera1);
while gp.input(sensor):
```

```
{
gp.setup(R1,LOW);
gp.setup(G1,LOW);
gp.setup(Y1,HIGH);
gp.setup(R2,HIGH);
gp.setup(Y2,LOW);
gp.setup(G2,LOW);
gp.setup(R3,HIGH;
gp.setup(Y3,LOW);
gp.setup(G3,LOW);
gp.setup(R4,LOW);
gp.setup(Y4,LOW);
gp.setup(G4,HIGH);
};
```

```
If gp.input(sensor)
print “Vehicle Detected"
while gp.input(sensor):
{
gp.setup(R1, LOW);
gp.setup(G1, HIGH);
gp.setup(Y1,LOW);
gp.setup(R2,LOW);
gp.setup(Y2, HIGH);
gp.setup(G2,LOW);
gp.setup(R3,HIGH;
gp.setup(Y3,LOW);
gp.setup(G3,LOW);
gp.setup(R4, HIGH);
gp.setup(Y4,LOW);
gp.setup(G4,LOW);
};
```

```
If gp.input(sensor)
print “Vehicle Detected"
while gp.input(sensor):
{
gp.setup(R1, HIGH);
gp.setup(G1,LOW);
gp.setup(Y1,LOW);
gp.setup(R2,LOW);
gp.setup(Y2,LOW);
gp.setup(G2, HIGH);
gp.setup(R3,LOW);
gp.setup(Y3, HIGH);
gp.setup(G3,LOW);
gp.setup(R4,HIGH);
gp.setup(Y4,LOW);
gp.setup(G4,LOW);
};
```

```
If gp.input(sensor)
print “Vehicle Detected"
while gp.input(sensor):
{
gp.setup(R1,HIGH);
gp.setup(G1,LOW);
gp.setup(Y1,LOW);
gp.setup(R2,HIGH);
gp.setup(Y2,LOW);
gp.setup(G2,LOW);
gp.setup(R3,LOW);
gp.setup(Y3,LOW);
gp.setup(G3,HIGH);
gp.setup(R4LOW);
gp.setup(Y4, HIGH);
gp.setup(G4,LOW);
};
```

Circuit diagram: