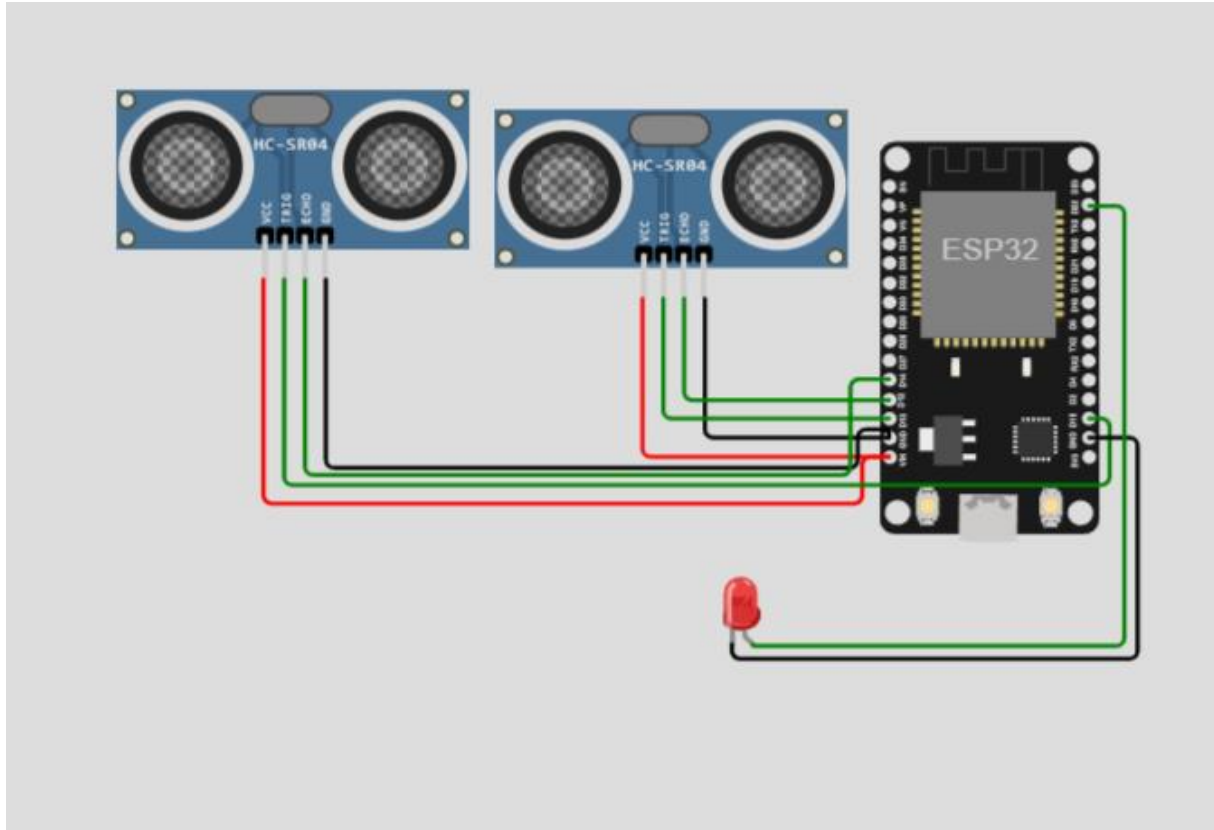


b

PUBLIC TRANSPORT OPTIMIZATION



PUBLIC TRANSPORT OPTIMIZATION

Public transport optimization refers to the process of improving the efficiency, effectiveness, and sustainability of public transportation systems within a city or region. The goal is to provide convenient, reliable, and cost-effective transportation options to the public while minimizing negative environmental and social impact

PUBLIC TRANSPORT OPTIMIZATION CODE:

```
BLYNK_TEMPLATE_ID = "TMPL26V4fGv5q"
```

```
BLYNK_TEMPLATE_NAME = "Test"
```

b

```
BLYNK_AUTH_TOKEN = "XEHxNF_Ur1Nt2p7wB5B20dNI1ZUwj34P"
```

```
import WiFi
```

```
import WiFiClient
```

```
import BlynkSimpleEsp32
```

```
duration1 = 0
```

```
distance1 = 0
```

```
duration2 = 0
```

```
distance2 = 0
```

```
dis1 = 0
```

```
dis2 = 0
```

```
dis_new1 = 0
```

```
dis_new2 = 0
```

```
entered = 0
```

```
left = 0
```

```
inside = 0
```

```
LED = 2
```

```
PIN_TRIG1 = 15
```

```
PIN_ECHO1 = 14
```

```
PIN_TRIG2 = 13
```

```
PIN_ECHO2 = 12
```

```
timer = BlynkTimer()
```

```
auth = BLYNK_AUTH_TOKEN
```

```
ssid = "Wokwi-GUEST"
```

```
pass = ""
```

b

```
BLYNK_PRINT = Serial
```

```
def get_distance1():
```

```
    digitalWrite(PIN_TRIG1, HIGH)
```

```
    delayMicroseconds(10)
```

```
    digitalWrite(PIN_TRIG1, LOW)
```

```
    duration1 = pulseIn(PIN_ECHO1, HIGH)
```

```
    distance1 = duration1 / 58
```

```
    return distance1
```

```
def get_distance2():
```

```
    digitalWrite(PIN_TRIG2, HIGH)
```

```
    delayMicroseconds(10)
```

```
    digitalWrite(PIN_TRIG2, LOW)
```

```
    duration2 = pulseIn(PIN_ECHO2, HIGH)
```

```
    distance2 = duration2 / 58
```

```
    return distance2
```

```
def myTimer():
```

```
    Serial.println("100")
```

```
    dis_new1 = get_distance1()
```

```
    dis_new2 = get_distance2()
```

```
    if dis1 != dis_new1 or dis2 != dis_new2:
```

```
        Serial.println("200")
```

```
        if dis1 < dis2:
```

```
            Serial.println("Enter loop")
```

COMPONENTS:

ESP32 Simulation:

The ESP32 is a popular WiFi and Bluetooth-enabled microcontroller, widely used for IoT Projects. Wokwi simulates the ESP32, ESP32-C3, ESP32-S2, ESP32-S3, ESP32-C6 (beta), and ESP32-H2 (alpha).

VIN GND D13 D12 D14 D27 D26 D25 D33 D32 D35 D34 VN VP EN 3V3 GND D15 D2 D4 RX2 TX2 D5 D18 D19 D21 RX0 TX0 D22 D23

HC-SR04 Ultrasonic Distance Sensor:

The TRIG pin to high for 10uS or more. Then wait until the ECHO pin goes high, and count the time it stays high (pulse length). The length of the ECHO high pulse is proportional to the distance.

Standard 5mm LED.

wokwi-led Reference

	Name	Description
A		Anode (positive pin)
C		Cathode (negative pin)

we Connected two HC-SR04 Ultrasonic Distance Sensor by using a wire .if we connected between a 2 UDS and connected to ESP32 Simulation.

And one Uds can be join by wire and sent to ESP32 Simulation and another UDS can join to same ESP32 Simulation and both are connected through wire. All are finally connected with a LED bulb.As