**Smart parking**

**Description :**

**Smart parking solutions detect parking space availability in real-time, helping to optimize on-street parking in cities and in parking garages or surface parking lots such as those in shopping malls, train stations, corporate campuses, and more.**

**Program :**

#define ECHO\_PIN1 15 //Pins for Sensor 1

#define TRIG\_PIN1 2 //Pins for Sensor 1

#define ECHO\_PIN2 5    //Pins for Sensor 2

#define TRIG\_PIN2 18   //Pins for Sensor 2

#define ECHO\_PIN3 26  //Pins for Sensor 3

#define TRIG\_PIN3 27   //Pins for Sensor 3

int LEDPIN1 = 13;

int LEDPIN2 = 12;

int LEDPIN3 = 14;

void setup() {

**Serial**.begin(115200);

  pinMode(LEDPIN1, OUTPUT);

  pinMode(TRIG\_PIN1, OUTPUT);

  pinMode(ECHO\_PIN1, INPUT);

   pinMode(LEDPIN2, OUTPUT);

  pinMode(TRIG\_PIN2, OUTPUT);

  pinMode(ECHO\_PIN2, INPUT);

   pinMode(LEDPIN3, OUTPUT);

  pinMode(TRIG\_PIN3, OUTPUT);

  pinMode(ECHO\_PIN3, INPUT);

}

float readDistance1CM() {

  digitalWrite(TRIG\_PIN1, LOW);

  delayMicroseconds(2);

  digitalWrite(TRIG\_PIN1, HIGH);

  delayMicroseconds(10);

  digitalWrite(TRIG\_PIN1, LOW);

  int duration = pulseIn(ECHO\_PIN1, HIGH);

  return duration \* 0.034 /2 ;

}

float readDistance2CM() {

  digitalWrite(TRIG\_PIN2, LOW);

  delayMicroseconds(2);

  digitalWrite(TRIG\_PIN2, HIGH);

  delayMicroseconds(10);

  digitalWrite(TRIG\_PIN2, LOW);

  int duration = pulseIn(ECHO\_PIN2, HIGH);

  return duration \* 0.034 / 2;

}

float readDistance3CM() {

  digitalWrite(TRIG\_PIN3, LOW);

  delayMicroseconds(2);

  digitalWrite(TRIG\_PIN3, HIGH);

  delayMicroseconds(10);

  digitalWrite(TRIG\_PIN3, LOW);

  int duration = pulseIn(ECHO\_PIN3, HIGH);

  return duration \* 0.034 / 2;

}

void loop() {

  float distance1 = readDistance1CM();

  float distance2 = readDistance2CM();

  float distance3 = readDistance3CM();

  bool isNearby1 = distance1 > 200;

  digitalWrite(LEDPIN1, isNearby1);

  bool isNearby2 = distance2 > 200;

  digitalWrite(LEDPIN2, isNearby2);

  bool isNearby3 = distance3 > 200;

  digitalWrite(LEDPIN3, isNearby3);

**Serial**.print("Measured distance: ");

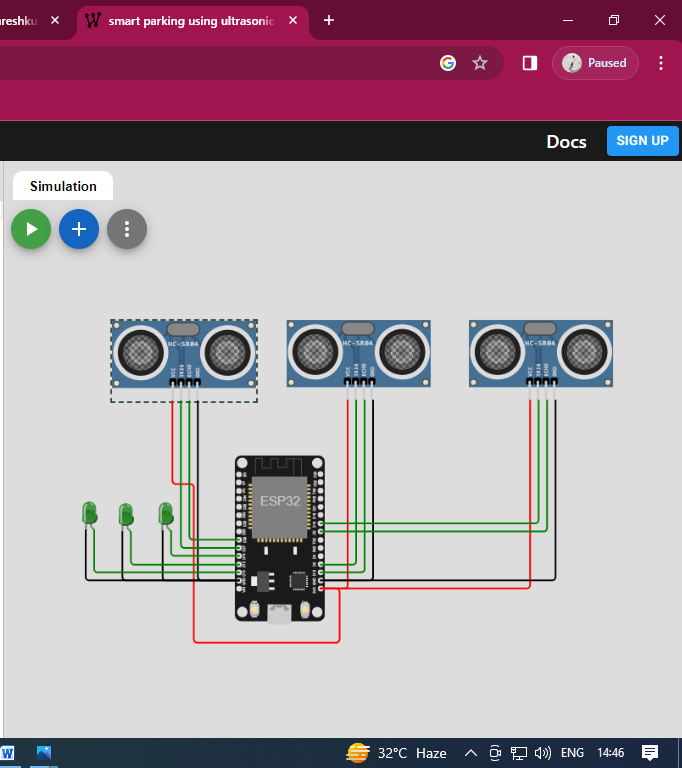
**Serial**.println(readDistance1CM());

**Serial**.println(readDistance2CM());

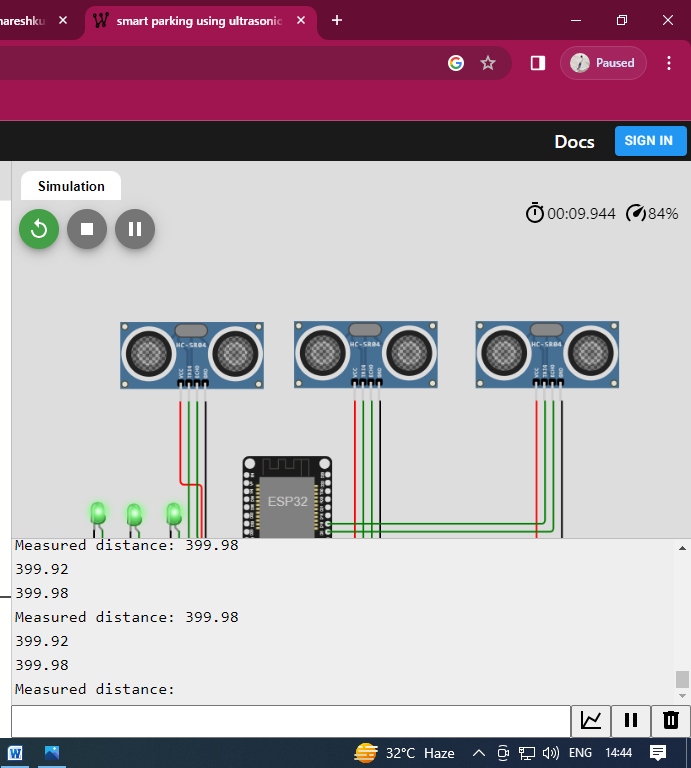
**Serial**.println(readDistance3CM());

  delay(100);

}

**Sensor :**

**Output :**

****