College name: JP college of engineering

College code: 9512

Project code: proj_211936_team_1

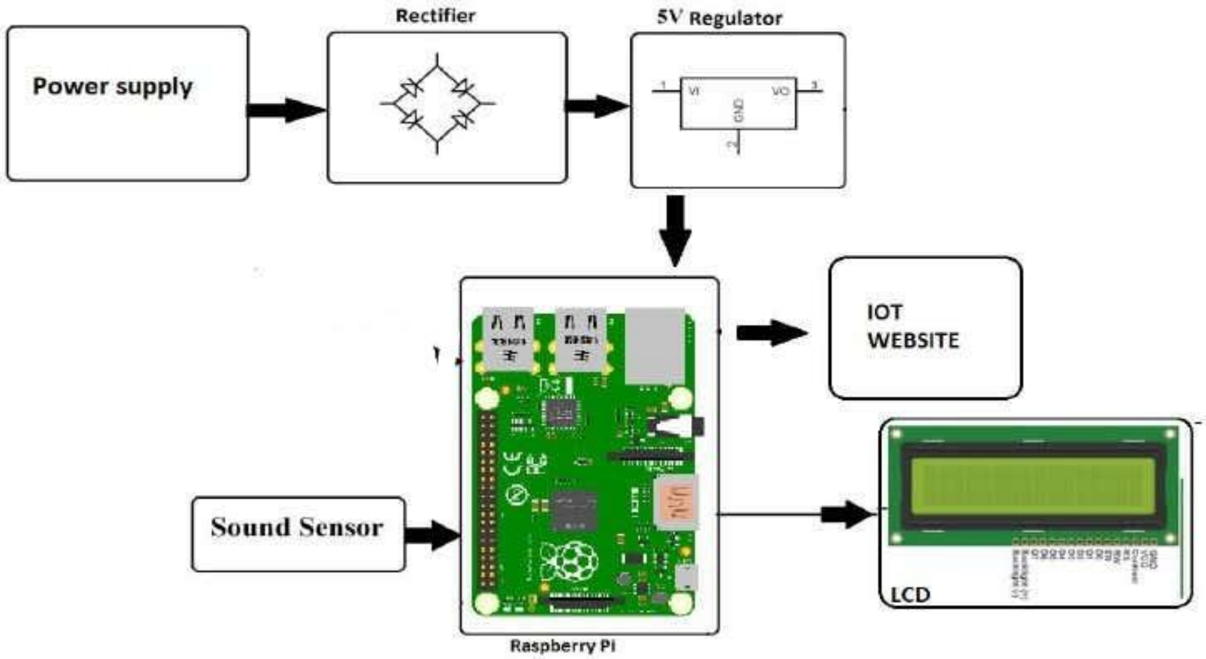
NOISE POLLUTION MONITORING

1. Santhiya.B (au951221106042)
2. Sangeeta.K (au951221106041)
3. Thirupathi.K (au951221106051)
4. Vijayalakshmi.A (au951221106054)
5. Veeralakshmi.K. (au951221106306)

IOT phase 4:

Noise pollution is a growing issue these days. It is necessary to monitor air quality and keep it under control for a better future and healthy living for all. Here we propose an air quality as well as sound pollution monitoring system that allows us to monitor and check live air quality as well as sound pollution in a particular areas through IOT. Also system keeps measuring sound level and reports it to the online server over IOT. The sensors interact with microcontroller which processes this data and transmits it over internet. Also authorities can keep a watch on the noise pollution near schools, hospitals and no honking areas, and if system detects air quality and noise issues it alerts authorities so they can take measures to control the issue.

Block diagram



Components:

Sound Sensors

Atmega 328 Microcontroller

Wifi ModeLp

LCD Display

LED’s

Resistors

Capacitors

Diodes

Program

```
Const int pingPin = 7;
Const int red=11;
Const int blue=10;
Int green=9;
Void setup() {
  // initialize serial communication:
  Serial.begin(9600);
  pinMode(red,OUTPUT);
  pinMode(blue,OUTPUT);
  pinMode(green,OUTPUT);
  pinMode(3, OUTPUT);
}
Void loop()
{
  digitalWrite(3, HIGH);
  delay(1000); // Wait for 1000 millisecond(s)
  digitalWrite(3, LOW);
  delay(1000); // Wait for 1000 millisecond(s)
  // establish variables for duration of the ping, and the distance result
  // in inches and centimeters:
  Long duration, inches, cm;
  // The PING))) is triggered by a HIGH pulse of 2 or more microseconds.
  // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
  pinMode(pingPin, OUTPUT);
  digitalWrite(pingPin, LOW);
  delayMicroseconds(2);
  digitalWrite(pingPin, HIGH);
  delayMicroseconds(5);
  digitalWrite(pingPin, LOW);

  // The same pin is used to read the signal from the PING))): a HIGH pulse
  // whose duration is the time (in microseconds) from the sending of the ping
  // to the reception of its echo off of an object.
  pinMode(pingPin, INPUT);
  duration = pulseIn(pingPin, HIGH);
```

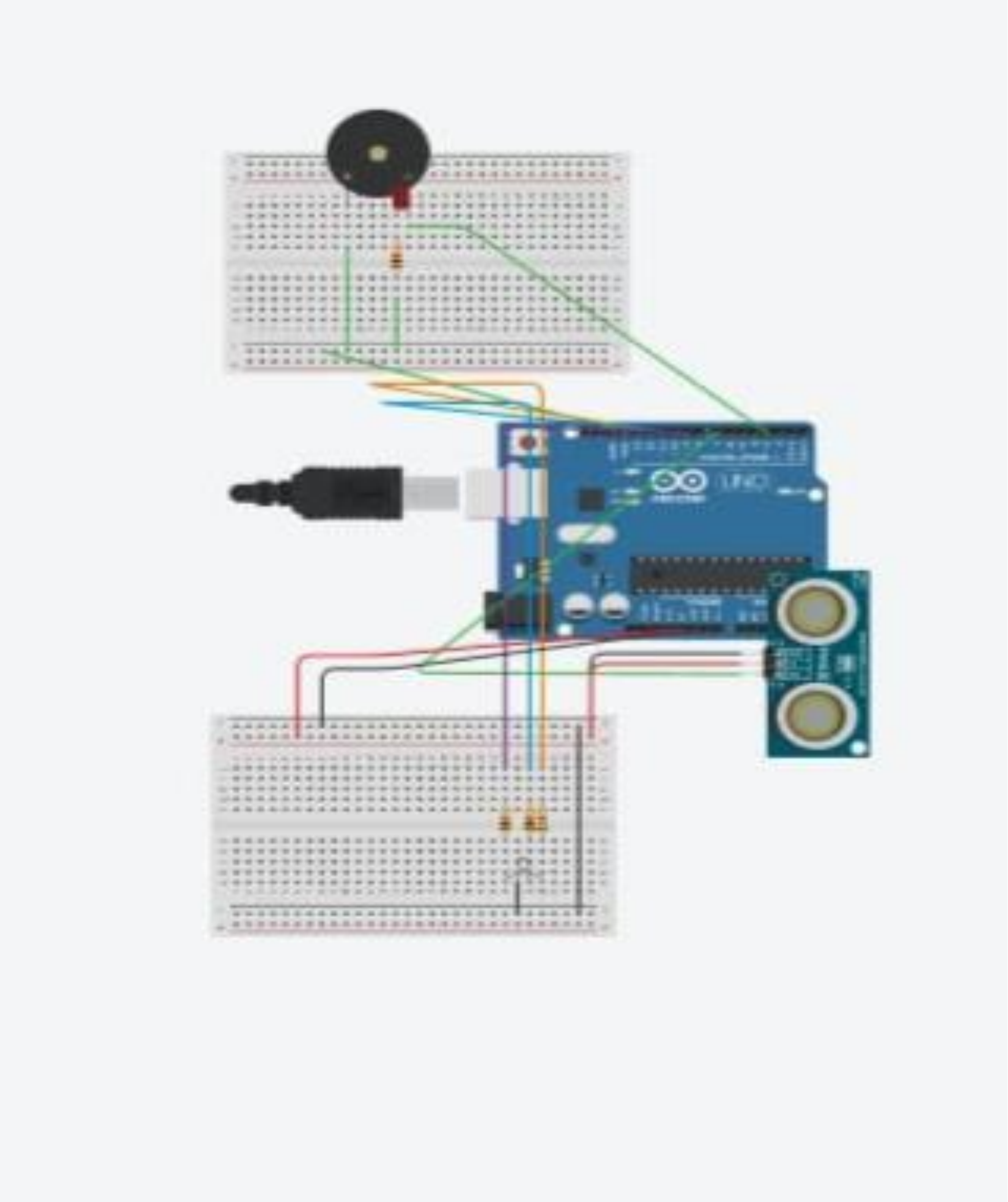
```
// convert the time into a distance
Inches = microsecondsToInches(duration);
Cm = microsecondsToCentimeters(duration);
Serial.print(inches);
Serial.print(“in, “);
Serial.print(cm);
Serial.print(“cm”);
Serial.println();

If(cm<256){
  analogWrite(red,cm);
  analogWrite(blue,255-cm);
  analogWrite(green,inches);
}
Else{
  analogWrite(red,0);
```

```
    analogWrite(blue,0);
    analogWrite(green,0);}
  delay(100);
}

Long microsecondsToInches(long microseconds) {
  // According to Parallax’s datasheet for the PING))),
there are 73.746
  // microseconds per inch (i.e. sound travels at 1130
feet per second).
  // This gives the distance travelled by the ping,
outbound and return,
  // so we divide by 2 to get the distance of the
obstacle.
  // See:
  Return microseconds / 74 / 2;
}
```

```
Long microsecondsToCentimeters(long microseconds) {
  // The speed of sound is 340 m/s or 29 microseconds per centimeter.
  // The ping travels out and back, so to find the distance of the object we
  // take half of the distance travelled.
  Return microseconds / 29 / 2;
}
```



Output

Real-Time Noise Levels: Continuous monitoring and display of current noise levels in decibels (dB) for specific locations in the monitored area.

These outputs are valuable for mitigating noise pollution, making informed decisions, and maintaining compliance with noise regulations.