**Iot\_Phase4**

**Noise Pollution Monitoring system**

**Project Proposal :** Noise Pollution Monitoring using IoT devices.

**Problem Definition :**

The Smart Noise project aims to address the growing concern of noise pollution in urban environments by deploying IoT sensors on to measure noise levels in public areas. The project's primary goal is to provide real-time noise data to the public through a user-friendly platform or mobile app. By doing so, it seeks to raise awareness about noise pollution and empower individuals and communities to make informed decisions to mitigate its impact. Understanding the Project

**Objectives:**

* To create noise maps that visually represent noise pollution levels in different areas.
* Deploy IoT sensors strategically to monitor noise pollution levels in urban public areas.
* Develop a user-friendly platform or mobile app for real-time access to noise data.
* Raise awareness about the impact of noise pollution on health and well-being.
* Enable data-driven decision-making for noise reduction initiatives.
* Ensure the long-term sustainability and scalability of the noise monitoring system.

**Components required:**

Microcontroller-Raspberry Pi Pico

Sound Sensor-Mic

**How does the project Works? :**

Noise pollution monitoring using a Raspberry Pi Pico works by using the Pico as the brains of the operation to collect noise data from a sound sensor (like a microphone) and transmit that data for analysis or storage.

1. Device Setup

2. Data Acquisition

3. Data Processing

4. Data Transmission

5. Data Analysis

6.Visualization and Alerting

**Working of the Project:**

The noise pollution monitoring system operates by utilizing sensors, typically microphones or sound sensors, to constantly capture sound data from the environment. This data is then processed to measure the noise level in decibels (dB). The collected noise information can be transmitted to a central system or database using technologies such as the Internet of Things (IoT) for analysis and storage. Through data analysis, the system identifies noise patterns, peak levels, and trends over time, offering insights into when and where noise pollution is most pronounced. This information is crucial for decision-making, implementing noise control measures, enforcing legal regulations, and improving public awareness of noise-related issues to create quieter and more harmonious living environments.

**Main Program code :**

/\*

\* Microphone Noise Level Measurement in Decibels (dB SPL)

\*/

const int microphonePin = A0; // Analog pin connected to the microphone output

const float referenceVoltage = 3.3; // Reference voltage of the analog-to-digital converter (in volts)

const float sensitivity = 40; // Microphone sensitivity in mV/Pa (adjust this value based on your microphone's datasheet)

const float referencePressure = 20E-6; // Reference sound pressure level in Pa (corresponding to 0 dB SPL)

void setup() {

Serial.begin(115200);

pinMode(16, OUTPUT); // GPIO 16 on Raspberry Pi Pico

}

void loop() {

int sensorValue = analogRead(microphonePin); // Read analog input from the microphone

float voltage = (sensorValue / 1023.0) \* referenceVoltage; // Convert analog reading to voltage

// Calculate sound pressure level in dB SPL

float pressure = voltage / sensitivity; // Calculate sound pressure in Pa

float dBspl = 20 \* log10(pressure / referencePressure); // Calculate dB SPL

Serial.println("Sound Pressure Level: " + String(dBspl) + " dB SPL"); // Print dB SPL to the serial monitor

digitalWrite(16, sensorValue > 50); // Blink LED if noise level is above a threshold (adjust as needed)

delay(100); // Delay for a short duration before reading the microphone again

}

**Diagram Json Code :**

{

  "version": 1,

  "author": "premasagar K",

  "editor": "wokwi",

  "parts": [

    { "type": "wokwi-pi-pico", "id": "pico", "top": -41.55, "left": -34.8, "attrs": {} },

    { "type": "wokwi-microphone", "id": "mic", "top": -132.18, "left": -14.61, "attrs": {} }

  ],

  "connections": [

    [ "pico:A0", "mic:1", "blue", [ "v0" ] ],

    [ "pico:GND.1", "mic:2", "black", [ "v29", "h149" ] ],

    [ "mic:1", "pico:GP26", "green", [ "h76.8", "v144" ] ]

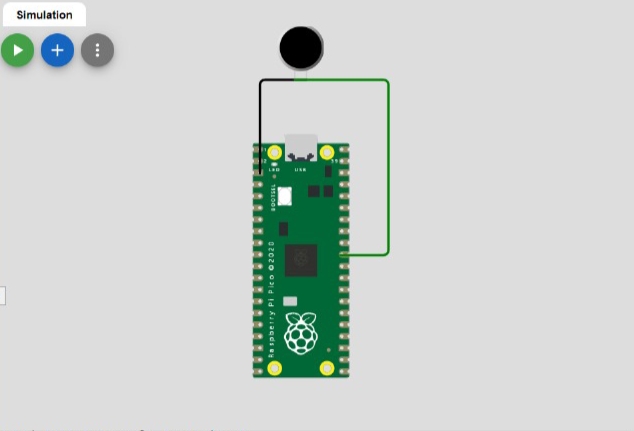
  ],

  "serialMonitor": { "display": "plotter" },

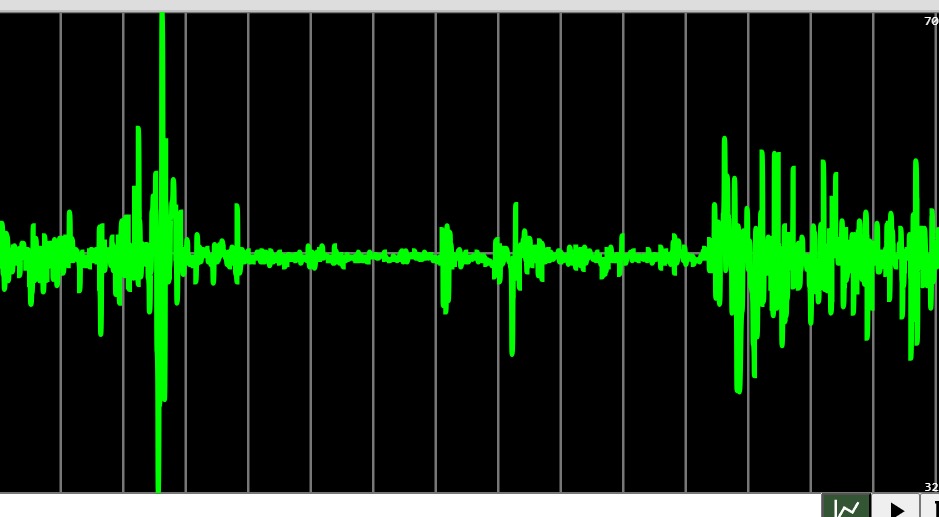
  "dependencies": {}

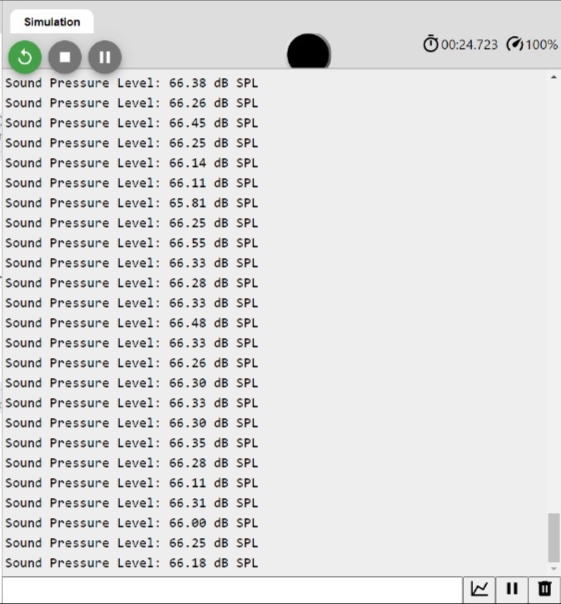
}

**Device simulation:**



**Output:**





**Conclusion:**

Noise pollution monitoring serves as a vital guardian of our well-being and the environment, fulfilling multiple roles. It assesses our exposure to excessive noise, ensuring regulatory compliance and pinpointing noise sources, while also raising public awareness and fuelling research for evidence-based policies. Through its vigilant presence, it aids in the prevention of harm and the creation of quieter, more peaceful living spaces for future generations.