IoT Based Noise Pollution Monitoring System

sound 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. System uses air sensors to sense presence of harmful gases/compounds in the air and constantly transmit this data to microcontroller. 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. This allows authorities to monitor air pollution in different areas and take action against it. 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.

Presented by:

ARUN. C

SASIKANNAN.S

VIDHYASAGAR. K

NAVEENKUMAR.S

Presentation in:

VIDHYASAGAR. K

INTRODUCTION

The main objective of IOT based air and noise pollution monitoring system is that the air pollution is a rising issue these days. As a human we need fresh air to survive. Air is most important factor in humanslife.If there is any kind of air pollution it's harmful for human. Air pollution kill more than seven million people worldwide every year. Pollution is very harmful for those people who have any kind of internal diseases on this type of people pollution affect veryfastly.In atmosphere is the full of air which contain monoxide, smoke, alcohol, benzene, methane etc.

Between this gases some are good and some are harmful for environment for certain level some gases are good for human, animals, plants but beyondcertain level these created problem for services to overcome these problem system is useful because of this we can analyse the air and noise pollution means how many pollution level in atmosphere in three different levels.We use Internet Of Things (IOT). In this we use thingspeak we can analyse previous data also using this platform in graphical form.

PROBLEM STATEMENT

An effective natural observing framework is essential to screen and estimate the conditions in the event of surpassing endorsed level of parameter (for example, commotion, COand radiation levels). At the point whenthe items like condition furnished with sensor gadgets, smaller scale controller and different programming application turn into a self-securing and self-observing condition.

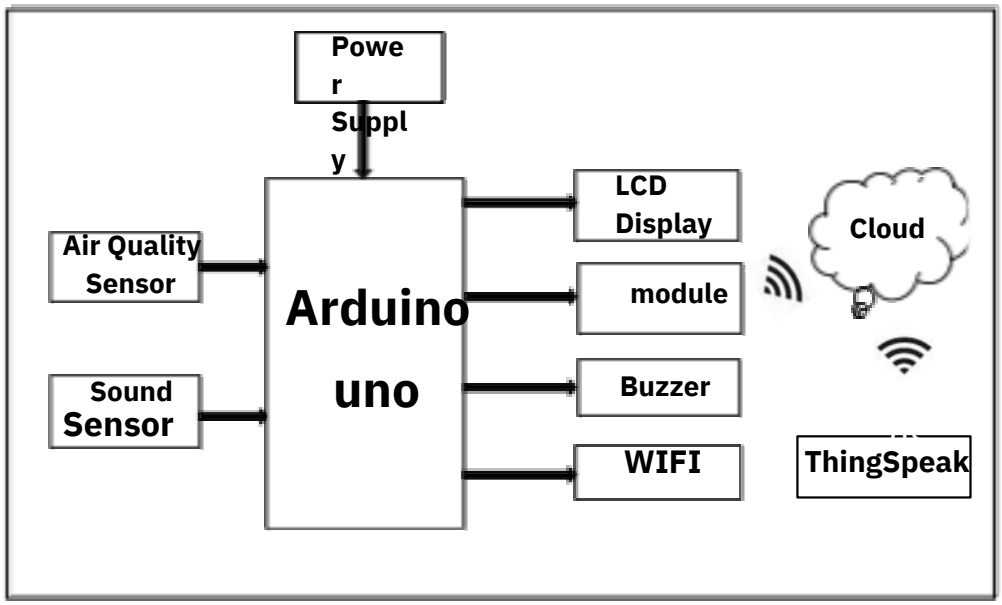
OBJECTIVES

1. To study the existingsystem.
2. To design the blockdiagram.
3. To decide the components specification &device insystem.
4. To design the circuit diagram and simulate it using suitablesoftware.
5. To design the PCB and implement hardware.
6. To test the circuit and observe the result.
7. To preparereport.

RELATED WORK

There are numerous works that have been done related to IOT based Projects.
Kavitha.B.C[1], Deepa Jose[2],Vallikannu.r KCG College of Technology, Chennai, India.[1] The idea can be realized by introducing raspberry pi & IOT shield . This project uses 3 different gas sensors, namely mq7, mq135 & DHT 11the air quality can be displayed on and as well as monitoring easy.Lcd display webpage whichmakes. Ajitesh Kumar, Mona Kumari, Harsh Gupta, GLA University Mathura, India.[2] They used nodemcu to monitoring air and noise which present in atmosphere MQ 2 gas sensor used to sense the smoke and MQ9 gas sensor used to sense carbon monoxide . PMS3003 G3 particle they used pm2.5 giving reading to node mcu processor and data to the rends. The dada internet. They also used a tod converter a dc to convert and leg data. To digital. They used OLED display, to display the pollution condition. All sensors will sends data to node mcu and and they node mcu send to think speak then graph shows on think speak and also data show onOLED.
V. Karthika1 S. Nandhini2, P. Preethy3 S. Sivapriya4, M.Thangabrindha5[3] In this system used PIC 16F877 Module for the Industrial Air monitoring system. The remote monitoring & controlling of Air quality of the room inside a building can designed the main purpose of constructing the proposed system.

BLOCK SCHEMATIC &WORKING



Block Diagram

As shown in fig. in system we use arduino as main controller. In system we use MQ135 gas sensor for detecting or sensing gases and also use sound sensor LM393 module for detect the sound pollution. Sensed data of sensor given to analog pin of the arduino then

digital output pin are connected to LCD, buzzer and LED. If air pollution is there then buzzer will start beeping and if sound pollution is there then LED will glow. All condition of pollution display on LED and we can also analyze past data using thingspeak in graphical form.

Arduino is an open source prototype. Software will operate in Arduino IDE Computer code can be written and upload to the physical board. Arduino board is a board that can be functioned via Arduino IDE by sending a set of instructions to the microcontroller on it. For controlling Sensors. For arduino programming we are going to use Embedded C. We are going to build project in Embedded C and for monitoring that project we are using Cloud.

COMPONENTS

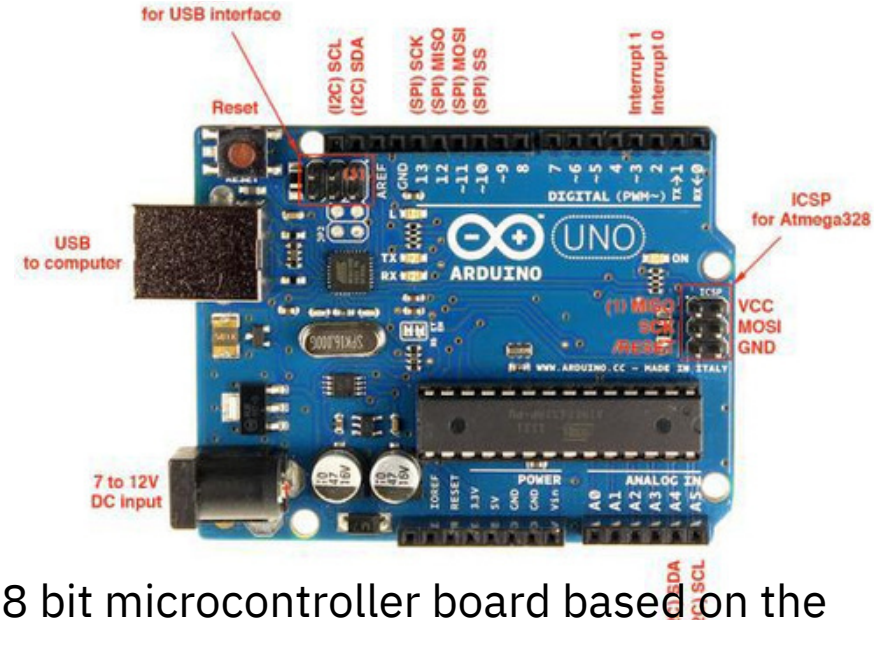
1. ArduinoUNO
2. MQ135 (Gassensor)
3. LM393 (Noisesensor) 4.

ESP8266 WIFIModule 5.

16*2 LCDDisplay

6. LED
7. Buzzer

Arduino UNO



Arduino is 8 bit microcontroller board based on the ATmega328P. The operating voltage is 5V. It has 14 pins digital input output pins (Of which can be used 6 as PWM output)

Oscillator frequency is 16 MHz It contains everything needed to support the microcontroller simply connect it to a computer with USB cable. It has 6 analog input pins.

Feature

- Operating voltage is5v.
- DC current per input pin is 40mA.
- Clock speed16MHz.
- DC current for 3.3v pin is 50mA.
- SPAM 2 KB
- EEPROM 1KB

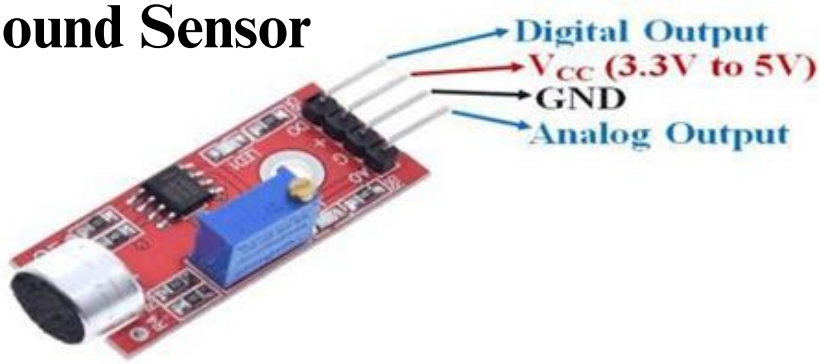
MQ135 Gas Sensor

The MQ135 is a gas sensor it used for detecting or sensing harmful gases in the atmosphere. It has wide detecting scope. It gives fast response and also it it high sensitivity sensor. It is simple and long life device. They are used in air quality control equipment for building offices are suitable for detecting of NH3, alcohol, benzene, smoke CO2 etc.

Feature

- Wide detecting scope
- Fast response and High sensitivity
- Stable and long life
- Operating Voltage is +5V
- Detect/Measure NH3, NOx, alcohol, Benzene, smoke, CO2, etc.
- Analog output voltage: 0V to 5V

LM393 Sound Sensor



The sound sensor module provide an easy way to detect sound and it generally used for detecting sound intensity. Module detect the sound has exceeded a threshold value. Sound is detected via microphone and fed into an LM393 opamp. The sound level adjust

through pot. The sound increases set value output is low. These module work on DC 3.3-5 voltage.

Feature

• Operating voltage 3.3V-5V
• Output model: digital switch outputs (0 and 1, high or low level)
• Voltage Gain 26dB
• Microphone Impedance 2.2kΩ
• Microphone Frequency 16.20 kHz

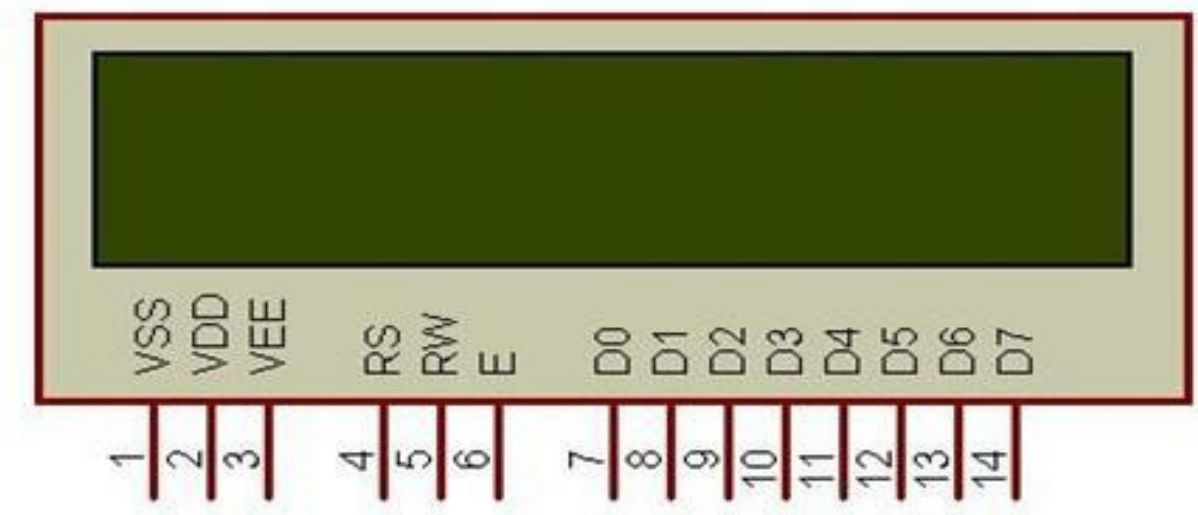
ESP8266 WIFI Module

The esp8266 WIFI module is a self containedsoc with integrated TCP/IP protocol stack that can give any microcontroller access to your WIFI network. The esp8266 is capable of either hosting an application or offloading all WIFI networking functions from another application processor.

Feature

2.4 GHz Wi-Fi (802.11 b/g/n supporting WPA/WPA2).
General-purpose input/output (16 GPIO).
Inter-Integrated Circuit (I²C) serial communication protocol.
Analog-to-digital conversion (10-bit ADC).
Serial Peripheral Interface (SPI) serial communication protocol.

16*2 LCD Display



LCD is used for to display the condition there are three conditions in air pollution and three conditions in noise pollution means air and sound is clear, moderately polluted or highly polluted that is displayed on LED.

Feature

Operating Voltage is 4.7V to 5.3V

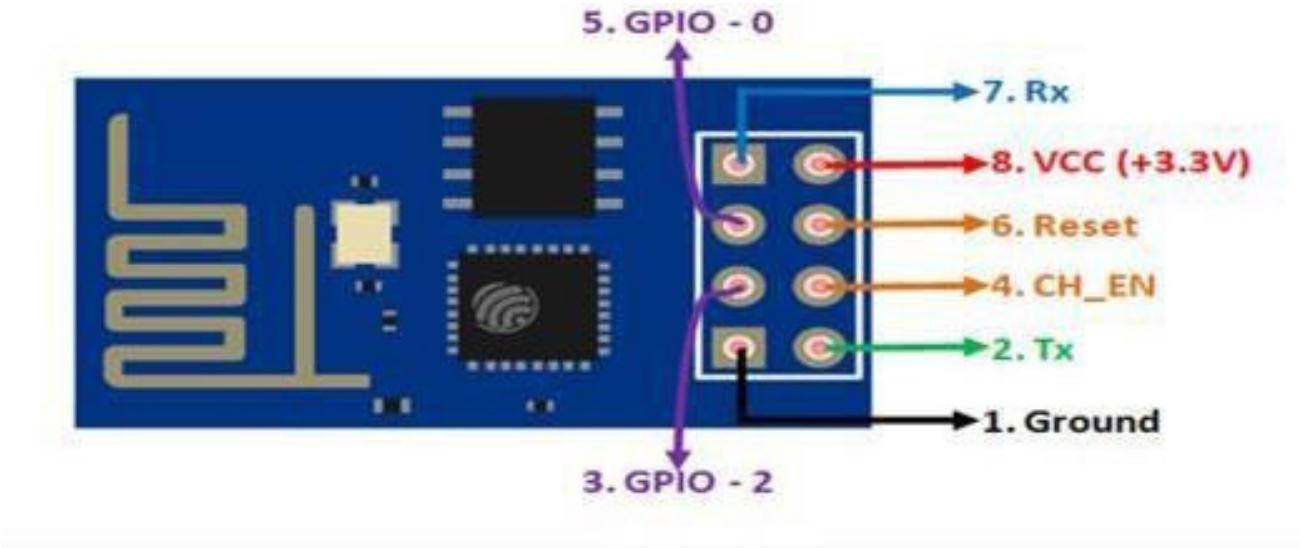
Current consumption is 1mA without backlight

Alphanumeric LCD display module, meaning can display alphabets and numbers

Consists of two rows and each row can print 16 characters.

Each character is build by a 5×8 pixel box

Can work on both 8-bit and 4-bit mode



ALGORITHM

1. Start Arduino UNO.
2. initialize LCD, gas sensor & noise sensor
3. Establish WITI connections.
4. If connection successful. next step else go to step 1
5. Read sensor values.
6. If Sensor value available establish TCP connection else read values again.
7. Establish TCP connection
8. If TCP connections successful. send data to server (Thingspeak). else set go to step 7
9. If TCP connections successful. send data to server (ThingSpeak). else set gob to step 7
10. Check for acknowledgement.
11. If acknowledgement received. go to step some time & else wait for go to step no 5.

PROGRAM

Step 1: Set Up Your IoT Device

Before you start with MIT App Inventor, ensure that you have your IoT device (e.g., Raspberry Pi with a noise sensor) set up to collect noise level data and send it to a server or a database. The IoT device should have an API to communicate with the mobile app.

Step 2: Create the Mobile App in MIT App Inventor

Go to the MIT App Inventor website (https://appinventor.mit.edu) and log in with your Google account.

Create a new project and name it something like "NoiseMonitorApp."

Design the app's user interface:

Add components like Labels, Buttons, and a Web component to create a user-friendly interface.

Use Labels to display the noise level and other information.

Use a Button to trigger data retrieval.

Set up the Web component:

In the "Blocks" section, add the Web component.

Configure it to make HTTP requests to your IoT device's API endpoint.

Step 3: Coding with Blocks in MIT App Inventor

Now, let's create the logic for your app using blocks:

When the Button is clicked, use the Web component to make an HTTP request to your IoT device's API endpoint to retrieve noise level data.

When the response is received, parse the data (usually in JSON format) using blocks to extract relevant information (e.g., noise level, location, timestamp).

Update the Label components on your app's interface to display the retrieved information.

Implement any additional functionality you require, such as historical data storage, charts, or user settings.

Step 4: Test and Deploy the App

After you've designed and coded your app, you can use the built-in emulator in MIT App Inventor to test your app. Once you are satisfied with the functionality, you can package the app for Android and distribute it to users via the Google Play Store or other means.

Step 5: IoT Device Integration

On the IoT device, you need to have a service running that listens for requests from your mobile app and responds with the noise level data. This service should expose an API endpoint for the mobile app to access.

You can implement this service using a programming language like Python (using Flask or Django for the web server) or Node.js, depending on your IoT device's capabilities.

Here's a simplified example of a Flask-based Python server:

python

Copy code

```
from flask import Flask, request, jsonify

app = Flask(__name)

# Endpoint to retrieve noise level data
@app.route('/get_noise_level', methods=['GET'])
```

```
def get_noise_level():
    # Replace this with your logic to retrieve noise level data from your IoT sensor
noise_level = 75.5
return jsonify({'noise_level': noise_level})


if __name__ == '__main':
app.run(host='0.0.0.0', port=5000)
```

This is a basic example to get you started. You will need to adapt and extend the functionality to meet your specific requirements for noise pollution monitoring.

ADVANTAGES

1. Sensors are easily available.
2. Sensors are effortlessly accessible.
3. Detecting of wide range of gases.
4. Simple, compact and easy to handle. 5. Sensors have long life time.
6. Low cost
7. Data can be used to control pollution.

APPLICATIONS

1. To estimate the pollution.
2. Indoor Air Quality Monitoring.

3. To design server and upload data on that server with date and time. 4. We can use it at industrial area as there is lot of noise pollution

5. In city roads traffic noise.
6. Activities like shooting, open air events, football and cricket matches.
7. At small level, in schools and colleges we can use this device.
8. Automation Automation.

FUTURE SCOPE

In future we modify the system to notify a user about the air quality and noise level it reaches beyond permissible level through sms or app.
We can monitor air and sound pollution level at any place of the world.

CONCLUSION

By using this project each and every variation we can analyze and inform nearby people in time. We can also analyze data form home using thingspeak.
The most important factor of this system is that it is small, cost efficient and portable. Sensors are available easily anywhere.
This system fully helpful to save the lives and overcome all the problem related to environment.

THANK YOU