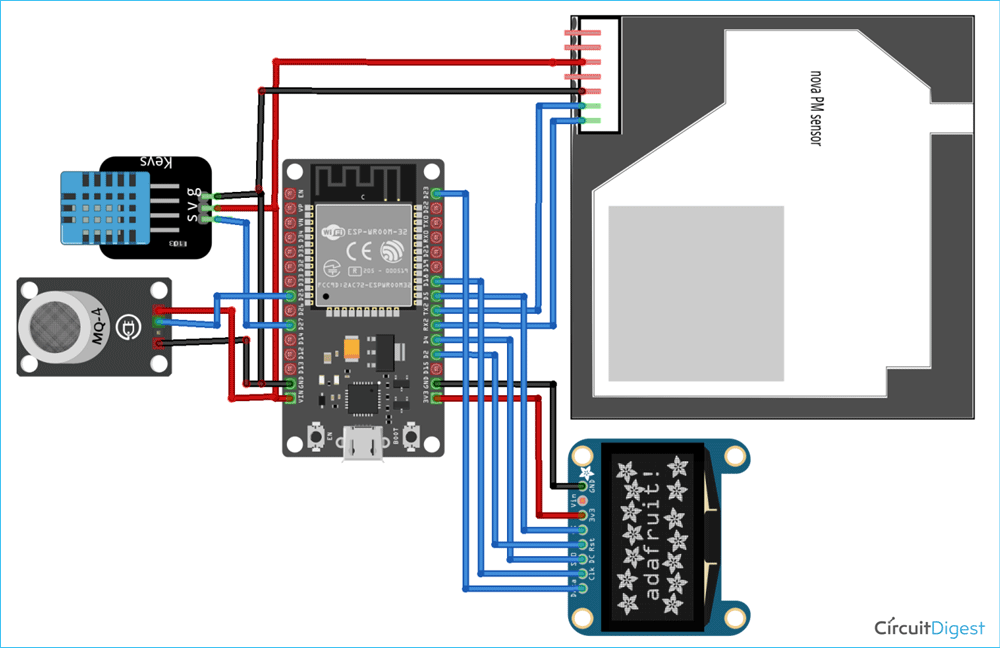
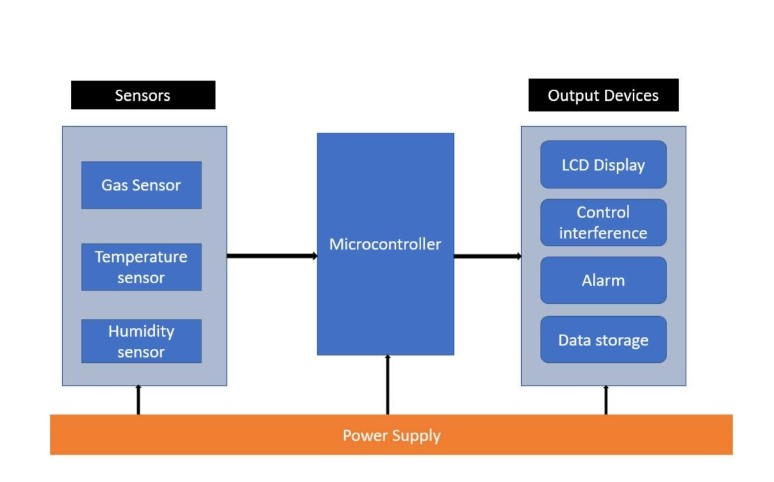
**Air Quality Monitoring**

` **Phase:-3**

# **Introduction:**

# In recent years, urban areas have experienced a significant increase in air pollution, leading to adverse health effects and environmental degradation. To address this critical issue, the City of Greenville implemented an advanced Air Quality Monitoring System (AQMS) aimed at real-time monitoring, analysis, and control of air pollutants. This case study explores the implementation process, challenges faced, and the impact of the AQMS on the city's environment and public health.

**Block diagram:**



**Block Diagram Description:**

1. Microcontroller/Main Processing Unit: This is the brain of the system, collecting data from sensors and processing it and managing output actions and displays.

2. Sensors: - Gas Sensor: Connects to the microcontroller and detects specific gasses. - Temperature Sensor: Connects to the microcontroller to measure temperature. - Humidity Sensor: Connects to the microcontroller to measure humidity.

3.Communication Interface: Enables the microcontroller to communicate with external devices or a computer. This can be Wi-Fi, Bluetooth, or wired connections like USB or Ethernet.

4. Display: Shows real-time data readings, alerts, or system status.

5. Alarm/Notification System: This can be an audible alarm, LED indicator, or any other signaling device that alerts the user when air quality goes outside the desired range.

6. Power Supply: Provides power to the entire system. This could be batteries, solar panels, or a direct power source.

7. Data Storage: Where the data can be logged for historical analysis. This could be an SD card, onboard memory, or cloud storage.

8. Control Buttons/Interface: Allows the user to interact with the system, set thresholds, or view historical data.

**Program**

import time

import machine

import dht

from machine import ADC, I2C, Pin

import ssd1306

# Initialize DHT11 sensor

dht\_sensor = dht.DHT11(machine.Pin(14))

# Initialize MQ-7 sensor (analog pin)

mq\_pin = ADC(Pin(34))

# Initialize OLED display

i2c = I2C(-1, Pin(22), Pin(21))

oled = ssd1306.SSD1306\_I2C(128, 64, i2c)

while True:

try:

# Read DHT11 sensor data

dht\_sensor.measure()

temperature = dht\_sensor.temperature()

humidity = dht\_sensor.humidity()

# Read MQ-7 sensor data

mq\_value = mq\_pin.read()

# Display data on OLED

oled.fill(0)

oled.text("Temperature: {}C".format(temperature), 0, 0)

oled.text("Humidity: {}%".format(humidity), 0, 20)

oled.text("CO Level: {}".format(mq\_value), 0, 40)

oled.show()

# Delay for 2 seconds

time.sleep(2)

except Exception as e:

print("Error: ", e)