**DESIGNING AN IOT-BASED AIR QUALITY MONITORING SYSTEM**

**Abstract:**

The air quality in an area very influential on the state of the population in an area because of the quality can affect both the health of humans, animals or plants. Therefore, it is necessary to periodically monitor air quality conditions in an area. In this study an IoT-based air quality monitoring system was designed to determine the air quality conditions in an area. The system will monitor using sensors to see the levels of several substances in the aur including 03, SO2, CO and particulates. Reading sensor data using an Arduino microcontroller. Then the data sent to the ThingSpeak cloud system uses a WIFI module on Arduino by accessing the API provided by the ThingSpeak cloud service. The monitoring results will be visible through a web page provided by the Thing Speak cloud service.

1. **Introduction**

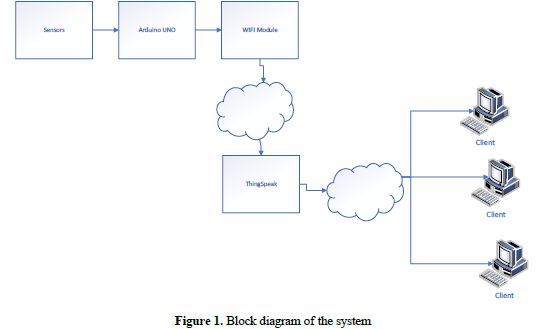
The air quality in an area very influential on the state of the population in an area because of the quality can affect both the health of humans, animals or plants[1-3]. Poor air quality can impact on health is not good for all living organisms. To prevent something that can worsen health conditions, it is necessary to know the condition of air quality in an area. Therefore, it is necessary to periodically monitor air quality conditions in an area.

Air quality monitoring can be done by measuring several parameters including temperature, humidity and air-contained compounds including 03 (Ozone), SO2 (Sulfur Dioxide), CO (Carbon Monoxide) and particulates [4–6]. At this time, advances in sensor technology can help us to measure the proficiency level parameters to determine the condition of the air quality[7,8]. In addition, the development of IoT (Internet of things) also helps with remote monitoring technology[9,10]. In this study we design an IoT- based monitoring system to monitor air quality in an area on a regular basis. Monitoring is done by collecting data from several sensors that can be accessed remotely. This research was conducted based on several previous existing studies. Previously there were studies that conducted control and monitoring of air quality in the room[11,12]. This research is also based on our research in designing remote communication for monitoring air quality[13]. In contrast to previous studies we designed an outdoor air quality monitoring system. The system will measure several compounds in the air including 03, SO2, CO and particles. Air quality will be monitored remotely from web pages.

**2. Method**

**2.1. System overview**

General overview designed system can be seen in the block diagram in Figure 1. Figure 1 shows a system designed to collect air quality data from various sensors that will measure the levels of 03. SO2, CO and particles in the air. The measurement results of these sensors will be read by the Arduino Uno controller. Arduino Uno is an ATmega328 based microcontroller board which is widely used for automatic control and monitoring[14,15,16]. Arduino Uno has 14 digital input/output pins (of which 6 can be used as PWM output), 6 analog inputs, 16 MHz ceramic resonator, USB connection, power jack, ICSP header, and reset button[17,18].



Furthermore, Arduino Uno will send data to the cloud system using WIFI module. We use ThingSpeak for cloud systems. ThingSpeak is an open source IoT application platform and API for storing and retrieving data from things using the HTTP protocol over the Internet or through a Local Area Network[19]. The monitoring results can be seen on the web page provided by ThingSpeak in graphical form.

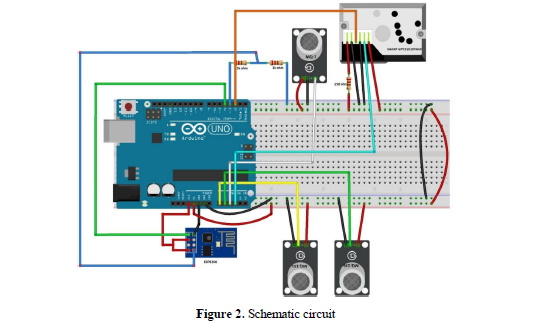
**2.2. The design of the system**

The first thing to do in system design is to make a schematic range as shown in Figure 2. We use 4 types of sensors to measure several types that are in the air, namely MQ-7, MQ1-131, MQ-135 and Pm10. Then do the coding on Arduino Uno using Arduino IDE. Arduino IDE is an editor used to write programs to Arduino [20]. Furthermore, create a channel in ThingSpeak to collect measurement data sent by the Arduino. Data transmission from Arduino is done by accessing the ThingSpeak API URL obtained after we create a channel to get data. Arduino accesses the ThingSpeak API using the WIFI ESP8266 module[21].

The MQ-7 sensor is used to read the levels of CO compounds in the air. MQ-7 is an analog sensor. The features of the MQ7 gas sensor are having high sensitivity to CO, stable, and long life[22]. This sensor uses heater power supply. SV AC/DC and uses circuit power supply: SVDC, measuring distance: 20- 2000 ppm to be able to measure carbon monoxide gas. In Figure 2 MQ-7 is connected to the A2 analog pin on Arduino.

To measure Ozone levels, we use MQ-131 which is an analog sensor to measure Ozone levels in the air[23]. MQ-131 works with a power supply of 5V (VCC) which is connected to a VCC pin on the microcontroller. If the sensor detects 03 gas in the air, the output voltage on the sensor will increase, so that the gas concentration will decrease and a deoxidation process will occur. 03 gas concentration value is obtained by calculating the ratio between the sensor resistance value and sensor resistance when the air is clean. In Figure 2 the MQ-131 is connected to the A0 analog pin on Arduino.

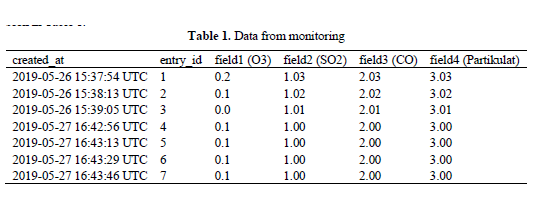
SO2 levels in the air are measured using the MQ-135 sensor which is connected to an analog Al pin on Arduino. MQ-135 sensor is a gas sensor whose working principle is to detect the quality of air in the room so that the characteristics of the sensor as a resistor will turn into a semiconductor or voltage conductor[24] Besides SO2 MQ-135 can also be used to measure ammonia, aromatics, benzene vapor, and other gases.

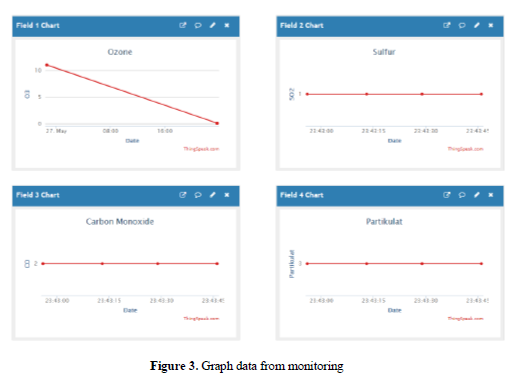


In addition to measuring the levels of some gases, we also measure particles in the air using PM10. PM10 sensor is a sensor that can measure airbome particles smaller than 10 microns (micrometres) [25]. In the schematic of the circuit in Figure 2 the Pm10 sensor is connected to an A3 analog pin on Arduino.

**3. Results and discussion**

From the results of the design that has been done, a system trial has been carried out that has been designed. The trial was carried out in the campus environment to monitor air quality conditions on the campus. From the results of the trial monitored data of compounds in the air that were measured were data on levels of Ozone, SO2, CO and PM10 particles. Data is received by ThingSpeak every one minute. Data received by Thing Speak can be monitored on the channel pages provided. On the web page data is presented in graphical form. Web pages can be accessed from the internet network by accessing the monitoring channel web pages as shown in Figure 3. The monitoring results can also be exported to table forms in the excel, xml or Json file format. Examples of export data in tables can be seen in Table 1.





**4. Conclusion**

From the trials conducted, the results of the design pan work well. It seems that the data obtained from the sensor can be seen from the webpage provided. Data changes can be monitored every once a minute. From the results obtained, further development can be done by making a mobile-based client application

Data can also be exported into Excel data format, XML or in Json

This is supported because ThingSpeak can also provide API access to get data that has been collected

on the server.

GITHUB LINK :https://github.com/Saravanankodi/Naan-mudhalvan-IOT-group-4-Team-6.

*SUBMITTED*

*BY*

*TEAM MEMBERS*

|  |
| --- |
| SARAVANAN.K(au821121104048) |
| MURUGANANTHAM.P(au821121104036) |
| MOHEMMED ASICK.A(au8211211032) |
| MANIBHARATHI.V.S(au821121104029) |
| GOVINDHAVASAN.K(au821121104301) |