IJCSPUB.ORG ISSN : 2250-1770
INTERNATIONAL JOURNAL OF CURRENT SCIENCE (IJCSPUB)
IJCS PUBLICATION (IJCSPUB.ORG) An International Open Access, Peer-reviewed, Refereed Journal

IOT – BASED AIR QUALITY MONITORING SYSTEM

[1] Dr. R. NANDHAKUMAR, [2] K. KANCHANA,

[1]Assistant Professor, [2] Post Graduate,
Department of Computer Science (S.F),
Nallamuthu Gounder Mahalingam College of Arts and Science, Pollachi-642001, India

Abstract: Due to the increased emissions of hazardous gases from vehicles and industries, the density of polluted air increases in the surroundings. It leads to human health problems such as asthma, lung disease, bronchitis and even premature death if the polluted air conditions are severe. In addition, polluted air can cause global warming and acid rain that causes unhealthy crop growth and the pollution of water supply sources. Therefore, a smart monitoring system is essential to check the air quality. This paper proposes a device of air quality monitoring system to efficiently monitor air conditions in real-time. The device comprises ESP8266 module Node MCU board as microcontrollers MQ-135detection gasses sensors such as Smoke, Toluene, Acetone, Carbon-dioxide (co2), NH4. The proposed system is designed with a user-friendly. The monitoring results can be displayed in a Thingspeak cloud with integrated real-time data transmitted by connecting to Wi-Fi network and GSM through ESP8266. In this paper, the Internet of Things (IoT) based in the air quality monitoring system plays an important part. Accordingly, the users can receive updates from everywhere.

Index Terms **- IOT, Node MCU, MQ - 135 Gas Sensor, Thingspeak.**

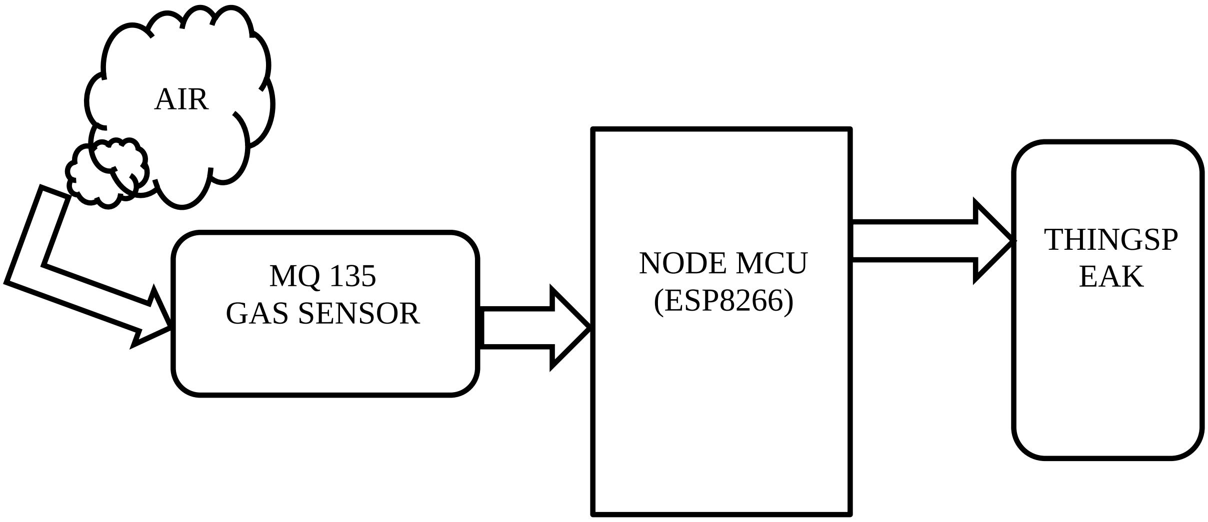
I. INTRODUCTION

The discharge of poisonous gases by industry, automobile emissions, and higher concentrations of dangerous chemicals and particulate matter in the atmosphere all contribute to air pollution. Pollution levels are fast rising as a result of causes such as industry, urbanization, population growth, and automobile usage, all of which can have a negative impact on human health. Particulate matter is one of the most important parameters, contributing significantly to the growth in air pollution. This necessitates the measurement and analysis of real-time air quality monitoring so that necessary choices may be made in a timely manner. I propose an air quality monitoring system that allows us to monitor and check live air quality and air pollution in an area using the Internet of Things (IoT). It employs air sensors (Gas Sensor MQ135) to detect the presence of dangerous gases/compounds in the air and continuously transmits the data. Furthermore, the system continues to measure and report the air level. The sensors communicate with the NodeMCU (Microcontroller), which analyses the data and communicates it to the application. This allows authorities to monitor and combat air pollution in various places. The configuration will display the air quality in PPM, allowing us to conveniently monitor it.

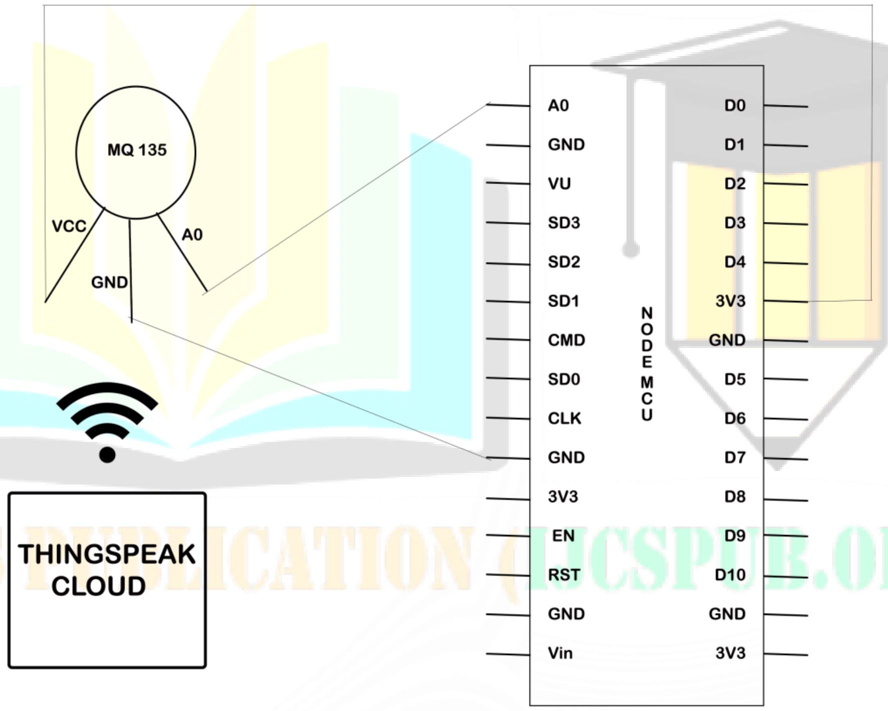
II. METHODOLOGY

2.1 Project Development

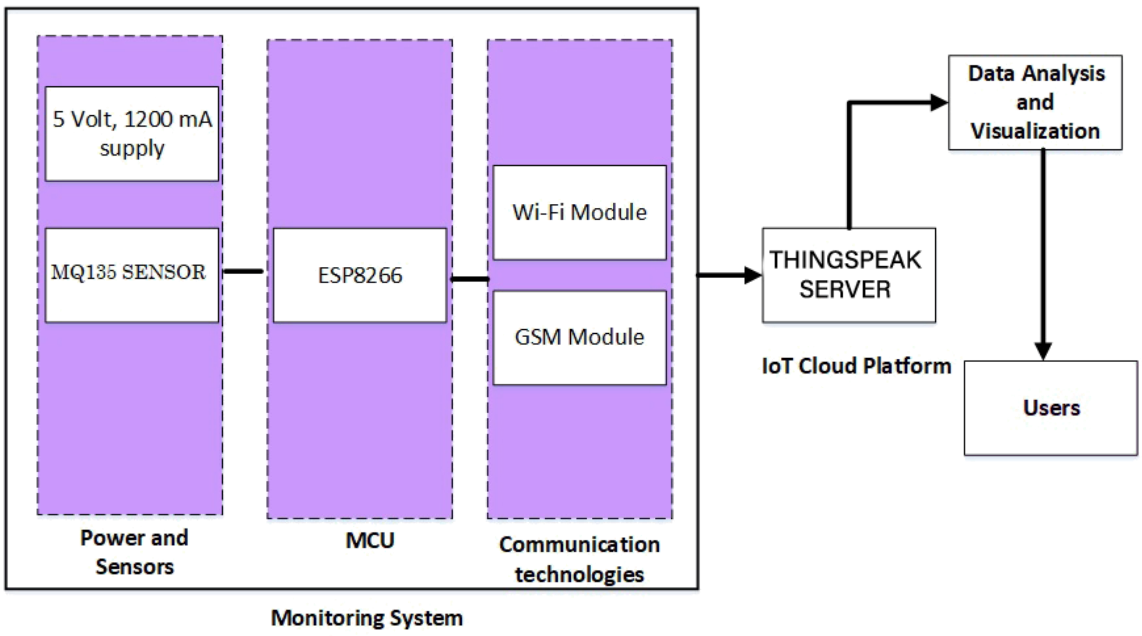
The Internet of Things (IOT) is primarily concerned with connecting intelligent devices to the Internet using the OSI layered architecture. As part of this work, we propose a cluster of MQ135 air quality monitoring gas sensors used to measure the concentration of air pollutants in the atmosphere. The MQ135 gas sensor is an interface with a small fixed platform with other features. I mainly used the Node MCU, which is an open source development board with a built-in Wi-Fi module. The MQ135 gas sensor is used to obtain gas concentration measurements. The Thingspeak mobile application is used to control IoT devices and display the collected data. This sensor data is captured and sent to a node MCU for IoT (Internet of Things) -based data collection, and displayed in the Thingspeak.



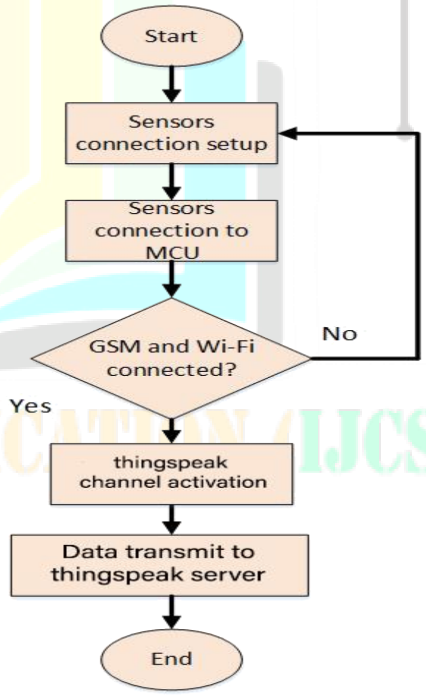
2.2 CIRCUIT DIAGRAM



2.3 BLOCK DIAGRAM FOR THE WORKING MODEL OF THE SYSTEM



2.4 WORKING PROCESS



It describes the functionality of the system. Firstly, the power supply at 5 V with 1200 mA is required to initialize the system. It ensures the connected sensor in active mode, so that the data from the sensors can be sent to the NodeMCU ESP8266 microcontroller. All the data from sensors being forwarded to Thingspeak server over a GSM and Wi-Fi network. ThingSpeak is an open-source Internet of Things application and API to store and retrieve data from things using the HTTP and MQTT protocol over the Internet or via a Local Area Network. When the communication network connection cannot be detected, the process is repeated starting from the sensor placement inspection until the communication network connection problem is resolved. Then, the collected data are analyzed and employed on data visualization to receive information updates of the air quality conditions. Finally, the information results are sent through Wi-Fi for detailed monitoring by the end-users.

III. MODELING AND ANALYSIS

NODEMCU (Microcontroller board): (in Fig 3.1)

Nodemcu V3 is an open-source firmware and development kit that plays a vital role in designing your own IOT product using a few Lua script lines. Multiple GPIO pins on the board allow you to connect the board with other peripherals and are capable of generating PWM, I2C, SPI, and UART serial communications.

- The interface of the module is mainly divided into two parts including both Firmware and Hardware where former runs on the ESP8266 Wi-Fi soc and later is based on the ESP-12 module.

The firmware is based on Lua – A scripting language that is easy to learn, giving a simple programming environment layered with a fast scripting language that connects you with a well-known developer community. And open source firmware gives you the flexibility to edit, modify and rebuilt the existing module and keep changing the entire interface until you succeed in optimizing the module as per your requirements.

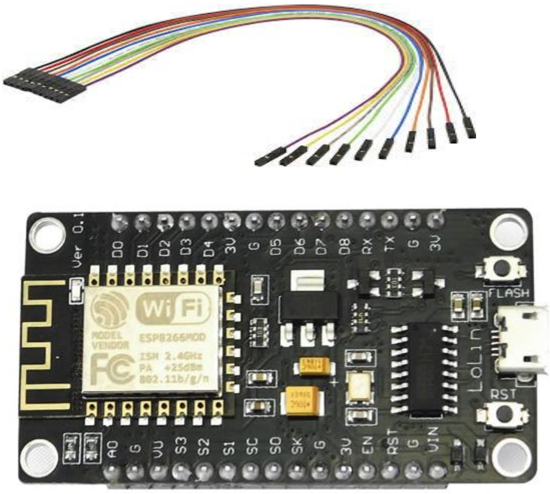


Fig 3.1

Setting Thingspeak & Getting API Key:

ThingSpeak is an open-source Internet of Things application and API to store and retrieve data from things using the HTTP and MQTT protocol over the Internet or via a Local Area Network. Thingspeak enables you to collect, store, analyze, visualize, and act on data from sensors.

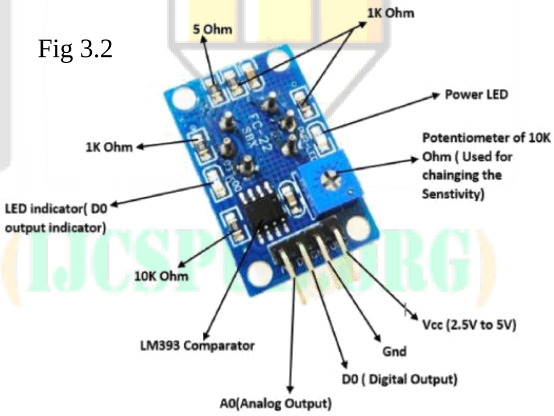
- To setup the Thingspeak Server, visit https://thingspeak.com/. Create an account or simply sign in if created the account earlier.
- Create a new channel by clicking on the button. Enter the basic details of the channel. Then Scroll down and save the channel.
- Then go to API keys copy and paste this key to a separate notepad file. We will need it later while programming.

GAS SENSOR MQ135: (in Fig 3.2)

The **MQ-135** gas sensor senses gases like **ammonia nitrogen, oxygen, alcohols, aromatic compounds, sulfide, and smoke**. The MQ-3 gas sensor has a lower conductivity to clean the air as a gas sensing material. In the atmosphere, we can find polluting gases, but the conductivity of the gas sensor increases as the concentration of polluting gas increases. MQ-135 gas sensor can be implemented to detect the **smoke, benzene, steam, and other harmful gases**. It has the potential to detect different harmful gases. It is with low cost and is particularly suitable for Air quality monitoring applications.

The **MQ135 sensor** is a signal output indicator instruction. It has two outputs: analog output and TTL output. The **TTL output** is a low signal light that can be accessed through the IO ports on the Microcontroller. The **analog output** is a concentration, i.e. increasing voltage is directly proportional to increasing concentration. This sensor has a long life and reliable stability as well.

Fig 3.2



Features

- High Sensitivity
- High sensitivity to Ammonia, Sulfide, and Benze
- Stable and Long Life
- Detection Range: 10 – 300 ppm NH3, 10 – 1000 ppm Benzene, 10 – 300 Alcohol
- Heater Voltage: 5.0V
- Dimensions: 18mm Diameter, 17mm High excluding pins, Pins – 6mm High
- Long life and low cost

Micro USB Cable: (in Fig 3.3)

The Micro USB cable allows you to connect your NodeMCU to your computer for programming. It also supplies power to the device. The NodeMCU only works with specific cables. Some USB cables are 'charging only', and have only 2 wires inside, meaning they can only provide power and can't transfer data. Cables with 4 wires can transfer data, which is what we need. In addition, you need a cable that can provide enough current to power the NodeMCU. Look for high quality, high speed, CE certified USB 2.0+ cable with at least 1 Amp of current and thickness of 28 AWG.

Fig 3.3

JUMPER WIRE : (in Fig 3.4)

A jump wire is an electrical wire, or group of them in a cable, with a connector or pin at each end, which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering.

Fig 3.4

3.2 SETUP THE HARDWARE OF THIS PROJECT

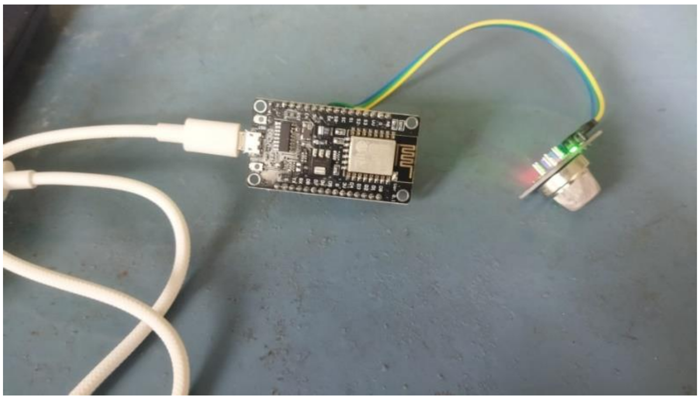


Fig 3.5

- After the circuit is assembled like in fig 3.5, code should be uploaded.
- Once code uploading is done open the serial monitor to see whether the Wi-Fi is connected or not. Make sure the baud rate should be 115200. If Wi-Fi is connected then you can see the gas level displayed in percentage and Air Quality Index in PPM then, data will be sent to thingspeak.
- Open thingspeak channel and select public /private view. Here we can see the data uploaded after the interval of 15 seconds.

IV. RESULT AND DISCUSSION

The data identified from the sensor used in the air quality monitoring system is displayed in the serial monitor are shown in Figure 4.2. Figure 4.1.1 & 4.1.2 shows the graphical user interface (GUI) display result in the Thingspeak when the MQ-135 gas sensor detected the gases in the surroundings. The real-time data collected by the sensor when the gas density increases according to the gas level conditions are also shown in serial monitor. Both serial monitor and Thingspeak interfaces displayed the updated data instantaneously.

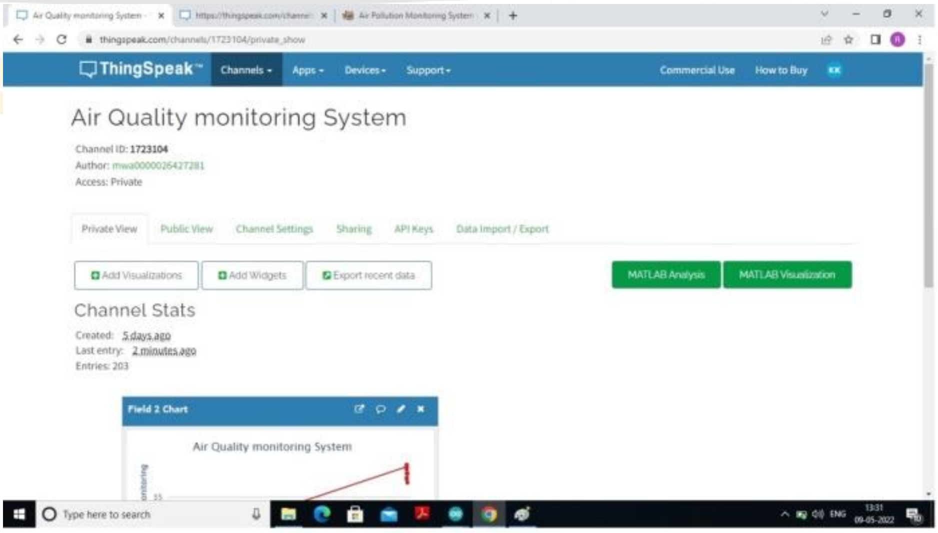


Fig 4.1.1

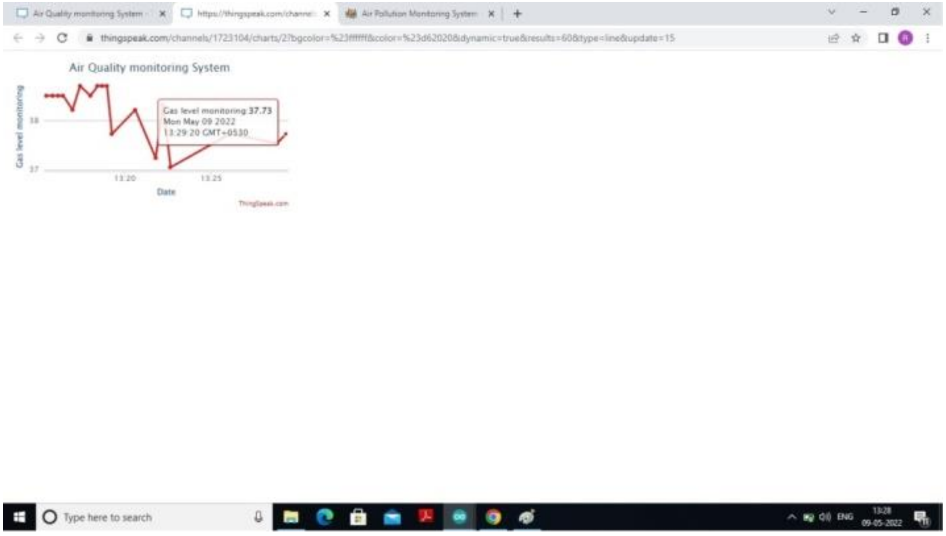


Fig 4.1.2

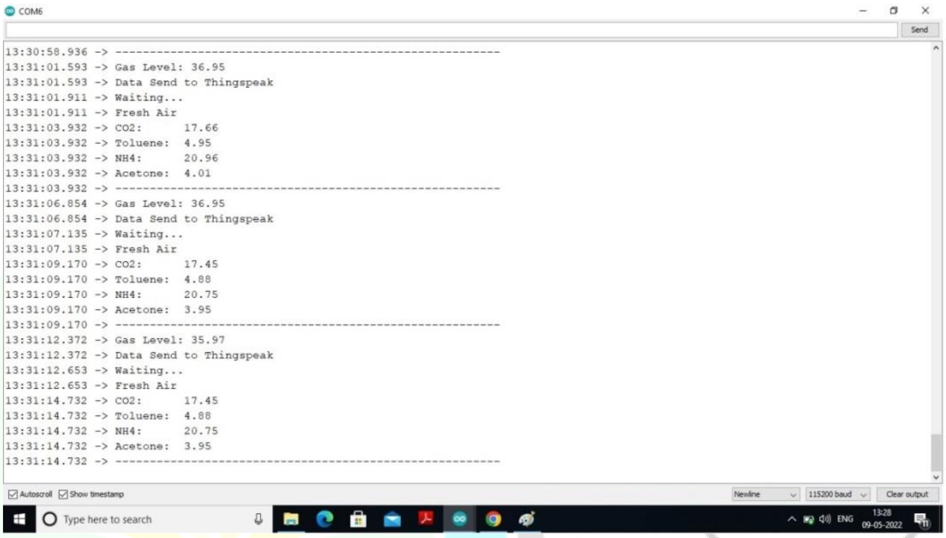


Fig 4.2

REFERENCES

[1] https://thingspeak.com/pages/learn_more
[2] https://en.wikipedia.org/wiki/NodeMCU
[3] https://www.instructables.com/How-to-Program-NodeMCU-on-Arduino-IDE/
[4] https://www.youtube.com/watch?v=Ku4Y29XPyyo
[5] https://www.scpe.org/index.php/scpe/article/view/1561
[6] https://www.slideshare.net/TejaTeja20/iot-based-air-quality-and-monitoring-by-using-arduino
[7] https://how2electronics.com/gas-level-monitor-esp8266-gas-sensor/
[8] https://circuitdigest.com/microcontroller-projects/iot-air-pollution-monitoring-using-arduino
[9] https://ieeexplore.ieee.org/abstract/document/9182330/
[10] https://how2electronics.com/iot-air-quality-index-monitoring-esp8266/
[11] https://create.arduino.cc
[12] https://www.instructables.com/Air-Qualiy-Monitoring/
[13] https://www3.epa.gov/airquality/montring.html
[14] https://create.arduino.cc/projecthub/abid_hossain/air-analyzer-using-almost-all-kinds-of-mq-sensors-01c2a4
[15] https://create.arduino.cc/projecthub/east-west-university/indoor-air-quality-monitoring-system-5b5244