**CO2 GAS LEVEL DETECTOR**

**EMBEDDED SYSTEM LAB PROJECT**

**Submitted by**

**DIVYADHARSHINI.B**

**(2018115027)**

**MONIKA.V**

**(2018115058)**

# 1.INTRODUCTION

## CO2 level is increasing day by day and the inevitable increase in industries and urbanization would add up to make it worse. Since air pollution is not visible to naked eyes it is important to regulate monitoring systems to actually aware the people of a city by measuring the pollutant parameters in real time and sending a notification whenever the CO2 level goes down beyond a certain level. Here, we are going to make an IOT based CO2 level Monitoring System.

## 1.2 PROBLEM STATEMENT

The goal of this project is to develop a device which can monitor PPM in air in real time, tell the quality of air and log data to a remote server(web application). The air monitoring device developed in this project is based on NodeMCU. The NodeMCU board connects with web application using ESP8266 Wi-Fi module through HTTP socket connection. The sensor used for monitoring the air pollution is MQ-135 gas sensor. The sensor value is sent to the NodeMCU. From the NodeMCU, the sensor value is sent to the backend text file by creating an URL to place the sensor data in the text file. The IP address of the Wi-Fi to which the ESP8266 Wi-Fi module is connected is displayed in the serial monitor. The sensor data is also displayed on a serial monitor.

**1.3 EXISTING WORKS**

CO2 gas level detector is already done with Arduino board and the sensor value from the

MQ-135 gas sensor is stored in an online database called ThingSpeak with the help of separate Wi-Fi module.

## 1.4 OBJECTIVES

The main objective of our project is to develop a CO2 gas level detector using MQ-135 gas sensor and NodeMCU and store the values from the gas sensor in a remote text file. The MQ-135 gas sensor senses the gases like ammonia nitrogen, oxygen, carbon-di-oxide, alcohols, aromatic compounds, sulfide and smoke. The operating voltage of this gas sensor is from 2.5V to 5.0V. MQ-135 gas sensor can be implementation to detect the smoke, benzene, steam and other harmful gases. It has potential to detect different harmful gases. The MQ-135 alcohol sensor consists of a tin dioxide (SnO2), a perspective layer inside aluminium oxide micro tubes (measuring electrodes) and a heating element inside a tubular casing. The end face of the sensor is enclosed by a stainless steel net and the back side holds the connection terminals. MQ 135 sensor is 4-pin multi-use sensor. The Vcc pin of MQ135 sensor is connected with Vin pin of NodeMCU, and GND pin is connected with NodeMCU’s GND pin. While the A0 pin is connected with A0 pin of NodeMCU as shown in the circuit diagram above. The complete set-up will be powered by the micro-usb port of NodeMCU through a USB cable. We first create a web server with 000webhost. There will be a text file with name “datastorage.txt”. We will store our sensor values in this file. ESP8266 Wi-Fi module can write the values in this file. Then GET request is added to get the text after the ‘=’ sign in the URL and that text will be saved in the variable named ‘val’. PHP variables start with ‘$’ character. file\_put\_contents () is a function used to push the contents in the file. Filename, data and flag(FILE\_APPEND) are arguments of this function. This function returns a status flag i.e. True or False. IF flag is true, means Data is stored successfully. In case of False, there might be some error. The ESP8266HTTPClient library provides function to begin HTTP connection with any website and send GET/POST requests. The ESP8266WiFi library can also be used for connecting Wi-Fi module with Internet. Setup function will be the trivial one containing the code to connect with Internet and display ‘Connected’. After that the sensor values are sent in the same manner.

**1.5 INTERFACES**

There are two interfaces in the CO2 level detector. First, is the interface between NodeMCU and the MQ-135 gas sensor, which is to sense the current air quality and send to the microcontroller. This interface is made with the help of jumper wires specifically female to female jumper wires. Second, is the interface between the NodeMCU and the laptop, to send the received sensor values to the computer for further analysis. This interface is primarily used to power up the NodeMCU which in turn powers up the gas sensor. This interfacing is made with the USB cable.

### 1.6 HARDWARE

The initial connections were made in the breadboard with normal connecting wires. As it lead to power loss, it was further rectified with jumper wires. Later, as the connections were simple, breadboard was removed and the connections were made directly between NodeMC and gas sensor. Also, there were some loose contact in the connections which could not be easily identified. The gas sensor wasn’t calibrated initially, which lead to improper readings. So it was calibrated.

### 1.7 SOFTWARE

Arduino IDE was installed and the sketch was typed, compiled, and uploaded in the arduino IDE. The sketch was uploaded in the NodeMCU 1.0(ESP-12E module) board in the port COM5. The header files included are ESP8266WiFi.h and ESP8266HTTPClient.h. ESP8266WiFi.h is to connect the microcontroller to a specified Wi-Fi network. This is achieved by specifying the SSID (Service Set IDentifier) and the password for the specified SSID.ESP8266HTTPClient.h is for providing HTTP socket connection.