

[ESP8266 Projects](#) [IoT Projects](#)

Gas Level Monitor On Internet Using ESP8266 & Gas Sensor

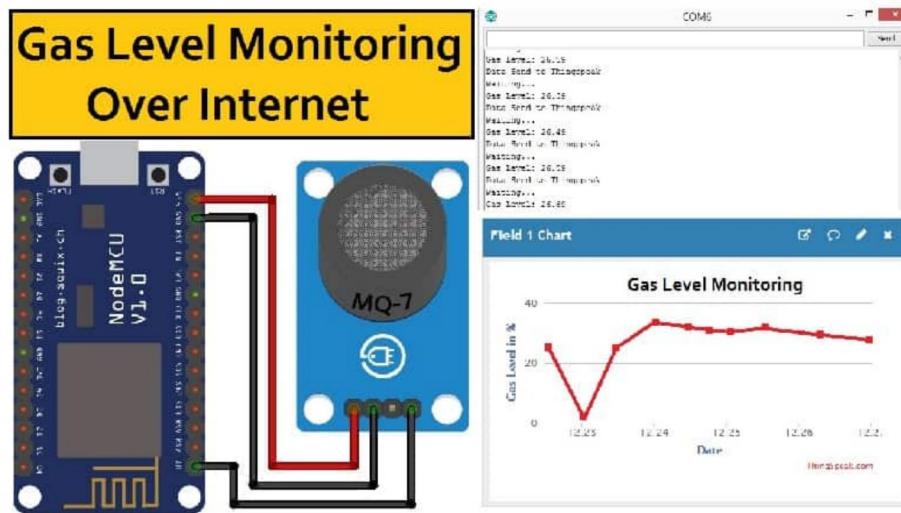
Admin Last Updated: November 18, 2021 10 16,264 3 minutes read

Table of Contents [hide]

- 1 Gas Level Monitoring Over Internet Using ESP8266 & Gas Sensor
- 2 Bill of Materials
- 3 MQ135 Gas/Smoke Sensor
 - 3.1 Description
 - 3.2 Features
- 4 Circuit Diagram & Connections
- 5 Setting Thingspeak & Getting API Key:
- 6 Source Code
- 7 Gas Level Monitor On Internet Using ESP8266 & Gas Sensor
- 8 Video Preview & Explanation

Gas Level Monitoring Over Internet Using ESP8266 & Gas Sensor

In this project, we will learn about **Gas Level Monitor On Internet Using ESP8266 & Gas Sensor Module**, i.e MQ135. We will measure the quantity of gas in percentage and send it over the internet using the **thingspeak server**. With this system, the data can be monitored remotely staying at any part of the world. We just need **gas/smoke/LPG sensor** like **MQ2/MQ3/MQ5/MQ7/MQ135** that is directly connected to **Nodemcu ESP8266-12E Module**.

ThingSpeak is an open-source Internet of Things (IoT) application and **API** to store and retrieve data from things using the HTTP protocol over the Internet or via a Local Area Network.

You can check our earlier projects while are similar to the current project:

1. [Alcohol Level Meter using Arduino & MQ-135 Alcohol/Gas Sensor](#)
2. [Arduino Smoke Level Detector using MQ-135 Sensor with Alert Alarm](#)
3. [Gas Leakage Detector with SMS Alert using GSM Module & Arduino](#)

Bill of Materials

The following are the components required for the project. All the components can be purchased from Amazon.

S.N.	COMPONENTS NAME	DESCRIPTION	QUANTITY	
------	--------------------	-------------	----------	---

S.N.	COMPONENTS NAME	DESCRIPTION	QUANTITY	
1	NodeMCU	ESP8266-12E Board	1	https://amzn.to/3LqfXWz
2	Air Quality Sensor	MQ135 Air Quality Sensor	1	https://amzn.to/3LqfXWz
3	Connecting Wires	Jumper Wires	10	https://amzn.to/3LqfXWz
4	Breadboard	-	1	https://amzn.to/3LqfXWz

MQ135 Gas/Smoke Sensor

Description

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.

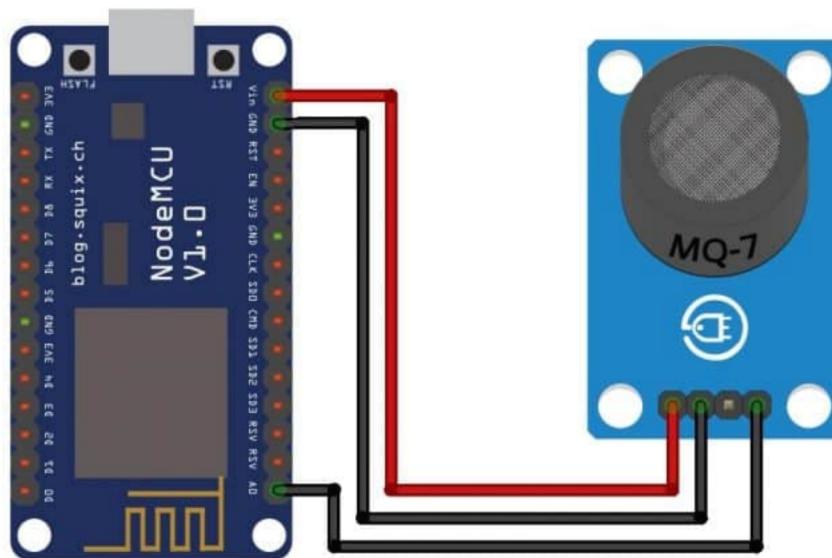
Features

- 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

Circuit Diagram & Connections

Make a connection as per the **circuit diagram** below. Connect the VCC pin of MQ135 to Vin of NodeMCU and GND to GND. Connect

analog pin A0 of MQ135 to Analog pin A0 of NodeMCU.



Setting Thingspeak & Getting API Key:

1. Go to <https://thingspeak.com/> and create an account if you do not have one. Login to your account.
2. Create a new channel by clicking on the button. Enter the basic details of the channel. Then Scroll down and save the channel.
You can follow the video guide below.

The screenshot shows the 'New Channel' creation interface on the ThingSpeak website. The 'Name' field is filled with 'Gas Level Monitoring'. The 'Field 1' section has the label 'Gas Level in %' and a checked checkbox. The 'Field 2' section is empty with an unchecked checkbox.

1. Then go to API keys copy and paste this key to a separate notepad file. You will need it later while programming.

Source Code

Here is a program for Gas Level Monitor On Internet Using ESP8266 & Gas Sensor MQ135. Copy this code and upload it to Nodemcu.

Change the wifi SSID, password, and thingspeak API key.

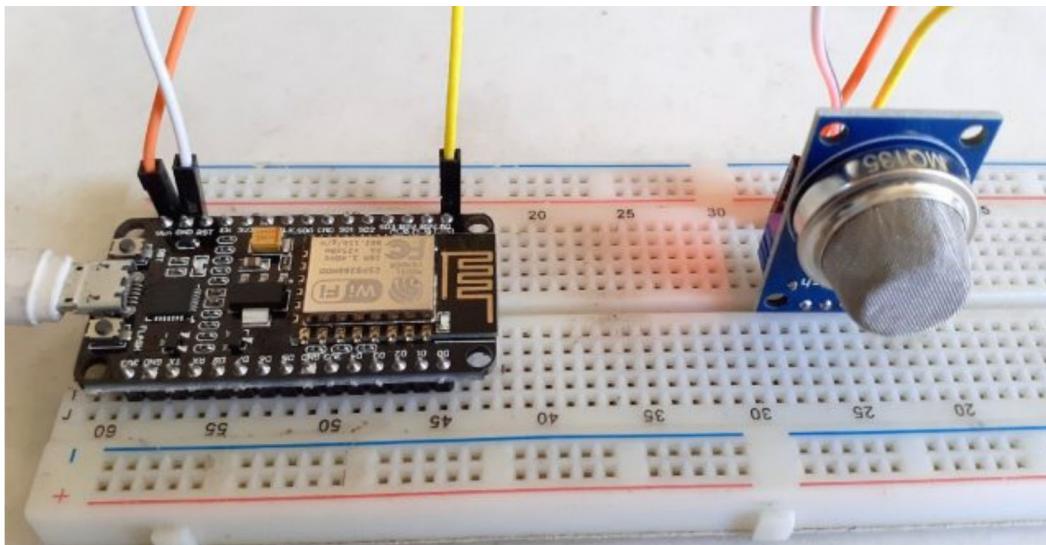
```
#include <ESP8266WiFi.h>
String apiKey = "SKP9YQY2CFVNK919"; // Enter your Write API key
const char *ssid = "Alexahome"; // replace with your wifi ssid
const char *pass = "hngzhouxiantan";
const char* server = "api.thingspeak.com";
WiFiClient client;
void setup()
{
Serial.begin(115200);
delay(10);
Serial.println("Connecting to ");
Serial.println(ssid);
WiFi.begin(ssid, pass);
while (WiFi.status() != WL_CONNECTED)
{
delay(500);
Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
}
void loop()
{
float h = analogRead(A0);
if (isnan(h))
{
Serial.println("Failed to read from MQ-5 sensor!");
return;
}

if (client.connect(server, 80)) // "184.106.153.149" or api.th
```

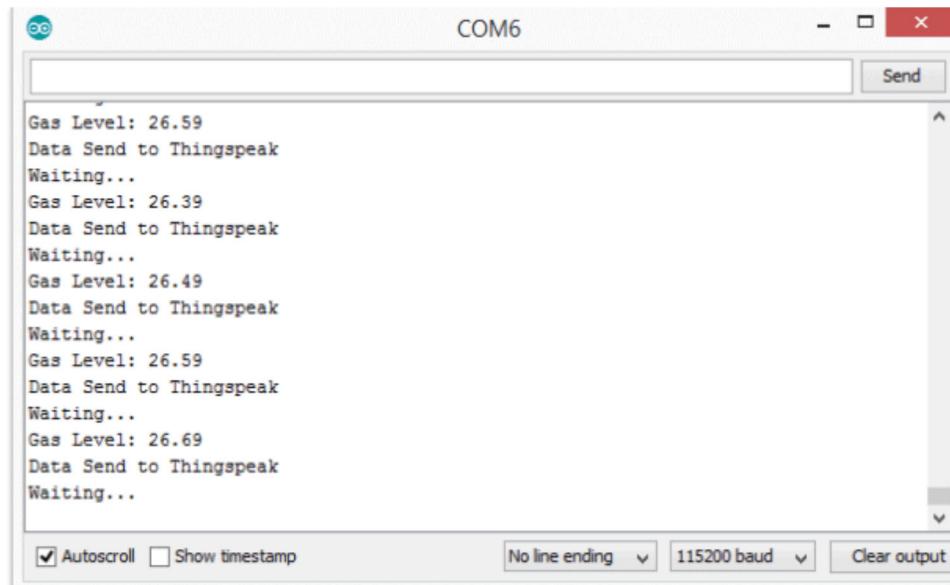
```
{  
String postStr = apiKey;  
postStr += "&field1=";  
postStr += String(h/1023*100);  
postStr += "r\n";  
client.print("POST /update HTTP/1.1\n");  
client.print("Host: api.thingspeak.com\n");  
client.print("Connection: close\n");  
client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n");  
client.print("Content-Type: application/x-www-form-urlencoded\n");  
client.print("Content-Length: ");  
client.print(postStr.length());  
client.print("\n\n");  
client.print(postStr);  
Serial.print("Gas Level: ");  
Serial.println(h/1023*100);  
Serial.println("Data Send to Thingspeak");  
}  
delay(500);  
client.stop();  
Serial.println("Waiting...");  
  
// thingspeak needs minimum 15 sec delay between updates.  
delay(1500);  
}
```

Gas Level Monitor On Internet Using ESP8266 & Gas Sensor

After the circuit is assembled on a breadboard, code should be uploaded.

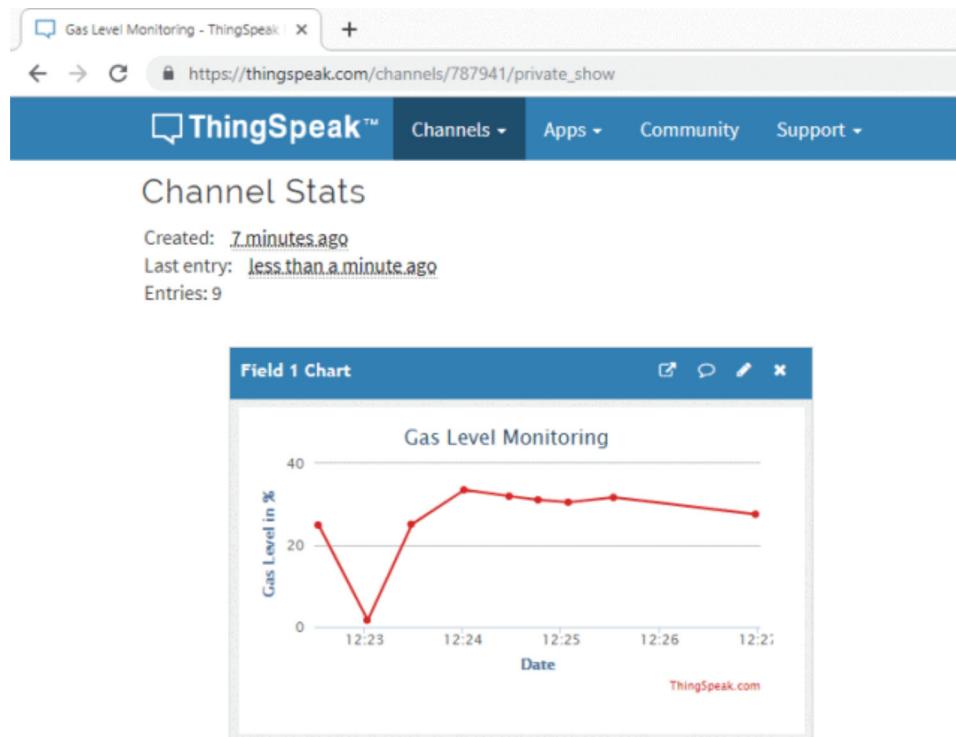


Once code uploading is done open the serial monitor to see whether the wifi is connected or not. Make sure the baud rate should be 115200. If wifi is connected then you can see the gas level displayed in percentage and data will be sent to thingspeak.



You can now introduce gas or smoke or perfume at the MQ135 and see the changes in a rise in the percentage value of gas level.

Open thingspeak channel and select public/private view. Here you can see the data uploaded after the interval of 15 seconds.



Video Preview & Explanation

IOT Based Gas Level Monitoring over Internet Using ESP8266

Watch this video [on YouTube](#).

10 Comments

**grishma**

October 1, 2019 at 7:00 PM

i uoloaded this code but nothing comes in serial monitor.please help me.

Loading...

Reply

**prince sharma**

May 6, 2020 at 8:48 PM

Didn't show the graph in thingspeak

Loading...

Reply

**Mario**

June 4, 2020 at 10:57 PM

I have the same issue. In monitor it works but not in Thingspeak

Loading...

Reply

**Aparna**

November 22, 2020 at 12:24 PM

How can we calibrate MQ135

Loading...

Reply

Todd Sharrard

January 30, 2021 at 8:31 PM

not a good lab to do – author has not updated code to work correctly with Thinkspeak

Loading...

Reply

Nishit

May 2, 2021 at 2:23 PM

same problem

Loading...

Reply

Rakesh

May 12, 2021 at 8:49 PM

Senosr data is displayed in serial monitor ,but the data is not same in JSON format in thingspeak?? it is showing null why
please me the solution

Loading...

Reply

Wade Alex

October 21, 2021 at 8:05 PM

Anyone solved the issue with thingspeak not showing the values? I can see my entries, but nothing on the graph.

Loading...

Reply

Wade GG

October 21, 2021 at 9:25 PM

UPDATE:

I solved the issue and here is the link for the code :

https://docs.google.com/document/d/1Xt4i5q6X_nxIBbx6qLLffofYwJqJ592F4WzdNPaJl8o/edit?usp=sharing

Loading...

Reply

abreham

October 23, 2021 at 1:58 PM

where is the scheme?

Loading...

Reply