**SCEEM Outreach Work Experience 2022**

**Air Quality Project Helpers Cheat Sheet**



**MQ-135**  - [Datasheet](https://www.olimex.com/Products/Components/Sensors/Gas/SNS-MQ135/resources/SNS-MQ135.pdf) - [Library](https://github.com/ViliusKraujutis/MQ135) - [Code Guide](https://docs.google.com/document/d/1-IcmL0PFMLWscSxfrf83bC4acDqrkmR5jGGRL1-vqUM/edit?usp=sharing)

measuresgeneral air quality - NH3, NOx, alcohol, Benzene, smoke, CO2, etc  
Read via analog pin (for the library) and runs on 5V.   
The sensor takes around 60 seconds to properly heat up and get roughly correct readings. If you breathe on the sensor you should see the value change as you’re increasing CO2 on it.

We’re looking for numbers in the hundreds (eg- 700ish)

Wiring:

MQ-135 -> Wemos,

1. A0 -> A0
2. Ignore D0.
3. GND -> GND
4. VCC -> 5V

Code to read sensor:

#include "MQ135.h"

MQ135 mq135\_sensor = MQ135(A0);

float temperature = 21.0;

float humidity = 25.0;

void setup() {

// put your setup code here, to run once:

Serial.begin(115200);

}

void loop() {

// put your main code here, to run repeatedly:

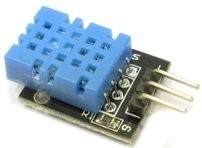
float airquality = mq135\_sensor.getCorrectedPPM(temperature, humidity);

Serial.print("Air Quality (PPM): ");

Serial.println(airquality);

delay(5000);

}

**DHT11** - [Datasheet](https://components101.com/sites/default/files/component_datasheet/DHT11-Temperature-Sensor.pdf) - [Library](https://github.com/beegee-tokyo/DHTesp) - [Code Guide](https://docs.google.com/document/d/1Er8DJXRYOe-j82qNtZJ3iiSVNrDEQ1L5Dd7ooW7rPTk/edit?usp=sharing)

measures temperature / humidity

It can run on 3.3 or 5V pin. It sends single wire data which should be connected to a digital pin on the wemos.

You can test the sensor by breathing on it - you should see the humidity and temperature going up.  
We’re looking for humidity in the 50s and Temperature in the 20s.

wiring:

DHT11 -> wemos;

1. + -> 3V3 or 5V
2. Out -> D7
3. - -> GND

Code to read sensor:

#include "DHTesp.h"

DHTesp dht;

float temperature;

float humidity;

void setup(){

Serial.begin(115200);

dht.setup(D7, DHTesp::DHT11);

}

void loop(){

humidity = dht.getHumidity();

temperature = dht.getTemperature();

Serial.print("Humidity (%): ");

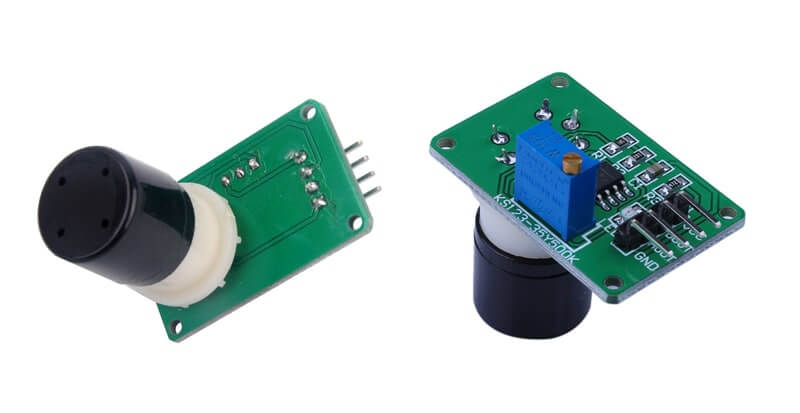
Serial.print(humidity);

Serial.print(", Temperature (C): ");

Serial.println(temperature);

delay(5000);

}

**MQ-131** - [Datasheet](https://cdn.sparkfun.com/assets/9/9/6/e/4/mq131-datasheet-low.pdf) - [Library](https://github.com/ostaquet/Arduino-MQ131-driver) - [Code Guide](https://docs.google.com/document/u/0/d/1xAPgOu6WURDJY_ohFiXQO--e6pOUWQmvoSLwsl4P7ls/edit)

Measures Ozone

Read via analog pin (for the library) and runs on 5V.

This one is super slow to calibrate and read so the groups will have to be patient! Blowing in to the sensor should increase the value that comes out.

Wiring:

MQ-131 -> Wemos,

1. A0 -> A0
2. Ignore D0.
3. GND -> GND
4. VCC -> 5V

Code to read sensor:

#include <MQ131.h>

void setup(){

Serial.begin(115200);

MQ131.begin(2,A0, LOW\_CONCENTRATION, 1000000);

MQ131.calibrate();

Serial.println("Calibration done!");

}

void loop(){

MQ131.sample();

float ozone = MQ131.getO3(PPM);

Serial.print("Concentration O3 (PPM): ");

Serial.println(ozone);

delay(60000);

}

**MQ-7** - [Datasheet](https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf) - [Library](https://github.com/fjebaker/MQ7) - [Code Guide](https://docs.google.com/document/u/0/d/1SOxqunTeVBkYko7gWsXOzr5aM6tMGpSdKJTmaW5Ckjg/edit)

Measures Carbon Monoxide

Read via analog pin (for the library) and runs on 5V.

Blowing in to the sensor should lower the value that comes out.

Wiring:

MQ-131 -> Wemos,

1. A0 -> A0
2. Ignore D0.
3. GND -> GND

VCC -> 5V

Code to read sensor:

#include "MQ7.h"

MQ7 mq7(0, 5); // set Pin A0, Voltage 5V

void setup(){

Serial.begin(115200);

mq7.calibrate();

Serial.println("Calibration done!");

}

void loop(){

float co = mq7.readPpm();

Serial.print("Concentration CO (PPM): ");

Serial.println(co);

delay(5000);

}

**Sending data to thingspeak code**

They’ll need to add in chunks of code from their specific sensor code - these bits are in red:

#include “ESP8266WiFi.h”

String writeApiKey = "write api key here";

const char\* ssid = "wifi name here";

const char\* password = "wifi password here";

const char\* server = "api.thingspeak.com";

WiFiClient client;

// sensor library etc here

void setup(){

Serial.begin(115200);

delay(10);

WiFi.begin(ssid, password);

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED){

delay(500);

Serial.print(".");

}

Serial.println("WiFi connected");

// sensor setup / calibration code here

}

void loop()

{

if (client.connect(server,80)) {

// sensor(s) reading to float(s) here

String postStr = writeApiKey;

postStr +="&field1=";

postStr += String(FLOATNAMEHERE);

client.print("POST /update HTTP/1.1\n");

client.print("Host: api.thingspeak.com\n");

client.print("Connection: close\n");

client.print("X-THINGSPEAKAPIKEY: "+writeApiKey+"\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 prints here

Serial.println("Sending data to Thingspeak");

}

client.stop();

Serial.println("Waiting 20 secs");

// thingspeak needs at least a 15 sec delay between updates

// 20 seconds to be safe - change this to 60 if using MQ-131

delay(20000);

}

Joe’s notes, not needed for helpers necessarily!

WHO & EU Limits: <https://www.hackair.eu/air-quality/>

| **Pollutant** | **WHO limit** | **EU limit** |
| --- | --- | --- |
| **Particulate matter PM10** | **50 µg/m³ 24-hour mean  20 µg/m³ annual mean** | **50 µg/m³ 24-hour mean 40 µg/m³ annual mean** |
| **Particulate matter PM2.5** | **25 µg/m³ 24-hour mean 10 µg/m³ annual mean** | **25 µg/m³ annual mean** |
| **Carbon monoxide (CO)** | **-** | **10 mg/m³ 8-hour mean** |
| **Ozone (O3)** | **100 µg/m³ 8-hour mean** | **120 µg/m³ 8-hour mean** |
| **Nitrogen dioxide (NO2)** | **200 µg/m³ 1-hour mean  40 µg/m³ annual mean** | **200 µg/m³ hourly mean 40 µg/m³ annual mean** |
| **Sulphur dioxide (SO2)** | **500 µg/m³ 10-minute mean 20 µg/m³ 24-hour mean** | **350 µg/m³ hourly mean  125 µg/m³ 24-hour mean** |

**Particulate matter (PM2.5/PM10)** - Sensor: PMS5003

Road dust, wear and tear of brakes and tyres on vehicles, construction activities, sea salt, forest fires, pollen, etc.

**Carbon monoxide (CO)** - Sensor: MQ-7 x 3

Produced during incomplete combustion of any fuel like natural gas, charcoal, gasoline, coal.

**Ozone (O3)** - Sensor: MQ-131 x4 (black sensor, green pcb)

Created by chemical reactions between oxides of nitrogen (NOx) and volatile organic compounds (VOCs). Emissions from industrial facilities, motor vehicle exhaust and chemical solvents are some of the major sources of NOx and VOCs.

**Nitrogen dioxide (NO2)** - Sensor: CJMCU-4541 x 2

Motor vehicles and any combustion of fossil fuels.

**Sulphur dioxide (SO2)** -Sensor: MQ-135 x 14 (does general “air quality”)

Combustion of fossil fuels, volcanic eruptions.

**Bonus sensors:  
Wemos x 16**

DHT11 x 11

**Not using:**

MH-Z19B - CO2