



Experiment - 2

Programme	: B.Tech. CSE	Semester	: Winter 2023-24
Course Title	: Internet of Things Lab	Code	: BECE352E
		Slot	: L23-24
Register Number	: 21BAI1106	Name	: Ojas Patil
Faculty (s)	: Dr. Manimaran P	Date	: 25 th Feb 2024

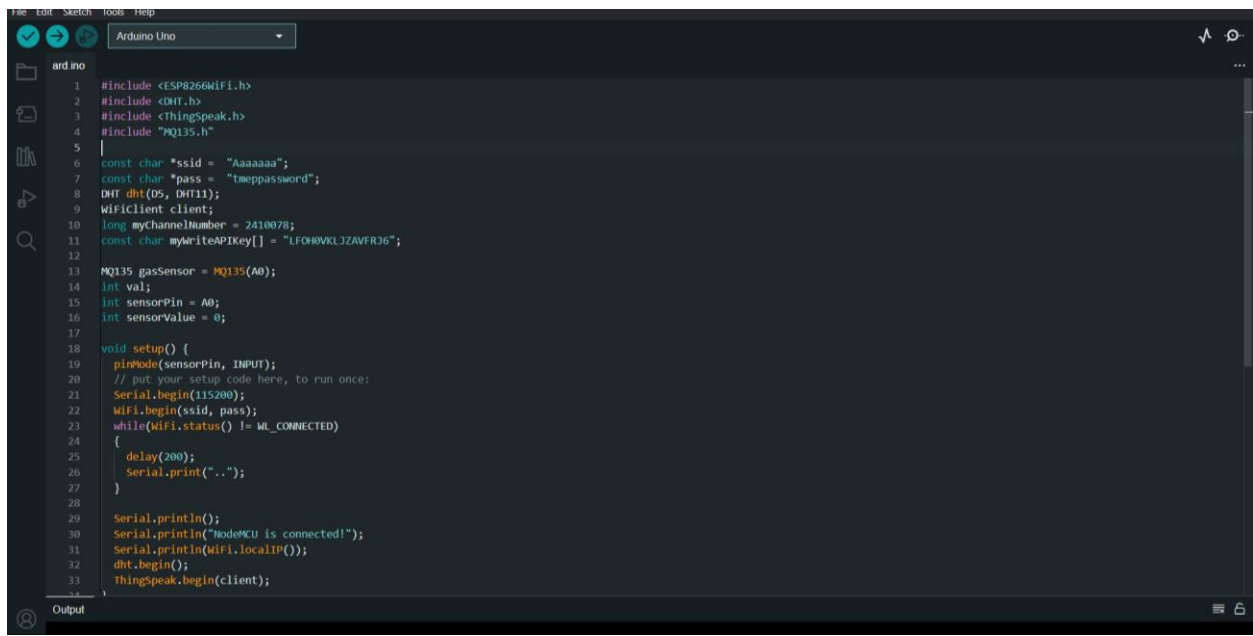
AIM

To develop an Air Quality Monitoring System using ESP8266 – NODEMCU with MQ135 and DHT 11 utilizing ThingSpeak Cloud Computing.

Steps to Follow

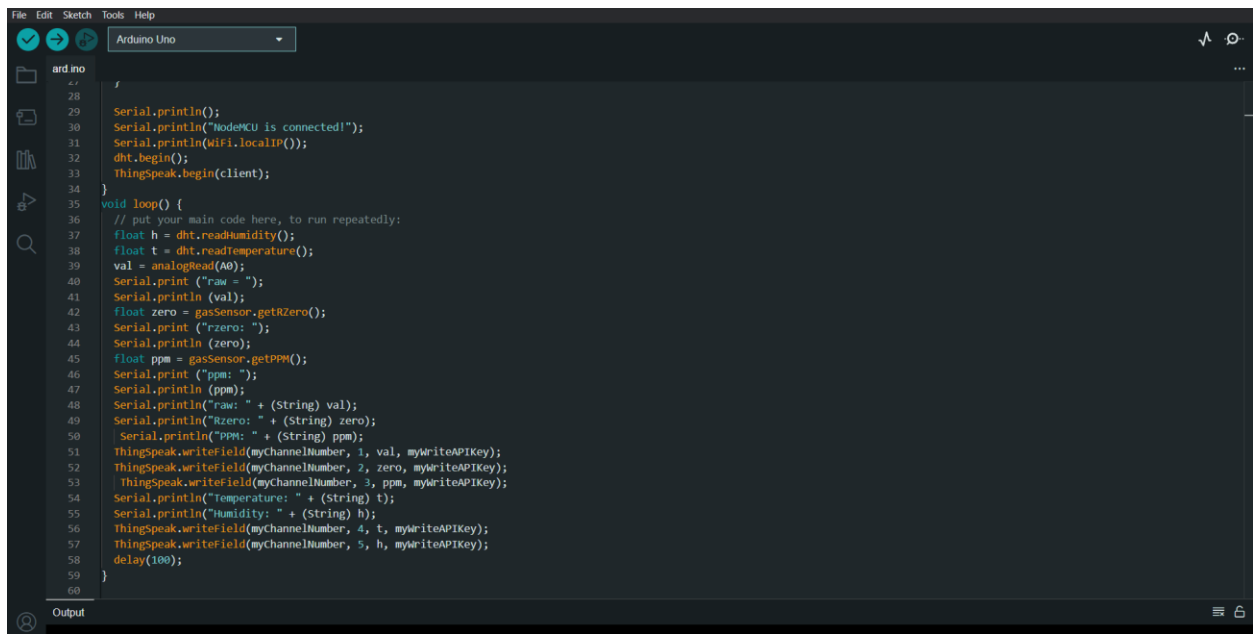
- 1) Initialize: Include libraries, define WiFi credentials, create sensor & communication objects, set sensor pin as input, initialize serial communication, connect to WiFi, initialize DHT sensor.
- 2) Main Loop: Read temperature, humidity, and raw analog value from sensors; calculate Rzero and PPM; print sensor readings (numerical & string) to serial monitor.
- 3) Serial Printing: Print raw sensor values and calculated values in both numerical and string formats to the serial monitor.
- 4) WiFi Connection: Attempt to connect to WiFi network, wait for connection & print dots, print confirmation message and local IP address upon successful connection.
- 5) ThingSpeak Communication: Use ThingSpeak library to write sensor readings (raw, Rzero, PPM, temperature, and humidity) to specific fields in your channel using channel number and API key.
- 6) Delay: Introduce a delay of 100 milliseconds before repeating the loop.

Code Screenshots



```
File Edit Sketch Tools Help
Arduino Uno

ard uno
1 #include <ESP8266WiFi.h>
2 #include <DHT.h>
3 #include <ThingSpeak.h>
4 #include "MQ135.h"
5
6 const char *ssid = "Aaaaaa";
7 const char *pass = "tmeppassword";
8 DHT dht(D5, DHT11);
9 WiFiClient client;
10 long myChannelNumber = 2410078;
11 const char myWriteAPIKey[] = "LFQ80VKL3ZAVFRJ6";
12
13 MQ135 gasSensor = MQ135(A0);
14 int val;
15 int sensorPin = A0;
16 int sensorValue = 0;
17
18 void setup() {
19   pinMode(sensorPin, INPUT);
20   // put your setup code here, to run once:
21   Serial.begin(115200);
22   WiFi.begin(ssid, pass);
23   while(WiFi.status() != WL_CONNECTED)
24   {
25     delay(200);
26     Serial.print(".");
27   }
28
29   Serial.println();
30   Serial.println("NodeMCU is connected!");
31   Serial.println(WiFi.localIP());
32   dht.begin();
33   ThingSpeak.begin(client);
34 }
```



```
File Edit Sketch Tools Help
Arduino Uno

ard uno
28
29 Serial.println();
30 Serial.println("NodeMCU is connected!");
31 Serial.println(WiFi.localIP());
32 dht.begin();
33 ThingSpeak.begin(client);
34 }
35 void loop() {
36   // put your main code here, to run repeatedly:
37   float h = dht.readHumidity();
38   float t = dht.readTemperature();
39   val = analogRead(A0);
40   Serial.print ("raw = ");
41   Serial.println (val);
42   float zero = gasSensor.getRZero();
43   Serial.print ("rzero: ");
44   Serial.println (zero);
45   float ppm = gasSensor.getPPM();
46   Serial.print ("ppm: ");
47   Serial.println (ppm);
48   Serial.println("raw: " + (String) val);
49   Serial.println("Rzero: " + (String) zero);
50   Serial.println("PPM: " + (String) ppm);
51   ThingSpeak.writeField(myChannelNumber, 1, val, myWriteAPIKey);
52   ThingSpeak.writeField(myChannelNumber, 2, zero, myWriteAPIKey);
53   ThingSpeak.writeField(myChannelNumber, 3, ppm, myWriteAPIKey);
54   Serial.println("Temperature: " + (String) t);
55   Serial.println("Humidity: " + (String) h);
56   ThingSpeak.writeField(myChannelNumber, 4, t, myWriteAPIKey);
57   ThingSpeak.writeField(myChannelNumber, 5, h, myWriteAPIKey);
58   delay(100);
59 }
60 }
```

Source Code

```
#include <ESP8266WiFi.h>
#include <DHT.h>
#include <ThingSpeak.h>
#include "MQ135.h"
```

```

const char *ssid = "Aaaaaaa";
const char *pass = "temppassword";
DHT dht(D5, DHT11);
WiFiClient client;
long myChannelNumber = 2410078;
const char myWriteAPIKey[] = "LFOH0VKLJZAVFRJ6";
MQ135 gasSensor = MQ135(A0);
int val;
int sensorPin = A0;
int sensorValue = 0;

void setup() {
  pinMode(sensorPin, INPUT);
  // put your setup code here, to run once:
  Serial.begin(115200);
  WiFi.begin(ssid, pass);
  while(WiFi.status() != WL_CONNECTED)
  {
    delay(200);
    Serial.print("..");
  }
  Serial.println();
  Serial.println("NodeMCU is connected!");
  Serial.println(WiFi.localIP());
  dht.begin();
  ThingSpeak.begin(client);
}

void loop() {
  // put your main code here, to run repeatedly:
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  val = analogRead(A0);
  Serial.print ("raw = ");
  Serial.println (val);
  float zero = gasSensor.getRZero();
  Serial.print ("rzero: ");
  Serial.println (zero);
  float ppm = gasSensor.getPPM();
  Serial.print ("ppm: ");
  Serial.println (ppm);
  Serial.println("raw: " + (String) val);
  Serial.println("Rzero: " + (String) zero);
  Serial.println("PPM: " + (String) ppm);
  ThingSpeak.writeField(myChannelNumber, 1, val, myWriteAPIKey);
  ThingSpeak.writeField(myChannelNumber, 2, zero, myWriteAPIKey);
}

```

```

ThingSpeak.writeField(myChannelNumber, 3, ppm, myWriteAPIKey);
Serial.println("Temperature: " + (String) t);
Serial.println("Humidity: " + (String) h);
ThingSpeak.writeField(myChannelNumber, 4, t, myWriteAPIKey);
ThingSpeak.writeField(myChannelNumber, 5, h, myWriteAPIKey);
delay(100);
}

```

Features and Description

Sensor	NodeMCU 12E	Sensor	NodeMCU 12E
DHT11	PIN in NodeMCU	MQ135	PIN in NodeMCU
+Vcc (+)	+3.3V	+Vcc (+)	+3.3V
Dout/OUT	GPIO2 (D4) hardware connection In software: GPIO2 is Referred as PIN2 for digital input	A0 (analog out)	A0 In software: A0 is Referred as Analog output
GND (-)	GND (G)		

Output Screenshot

Channels ▾ Apps ▾ Devices ▾ Support ▾
Commercial Use How to Buy

New Channel

Name

Description

Field 1 ☒

Field 2 ☒

Field 3 ☒

Field 4 ☒

Field 5 ☒

Field 6 ☐

Field 7 ☐

Field 8 ☐

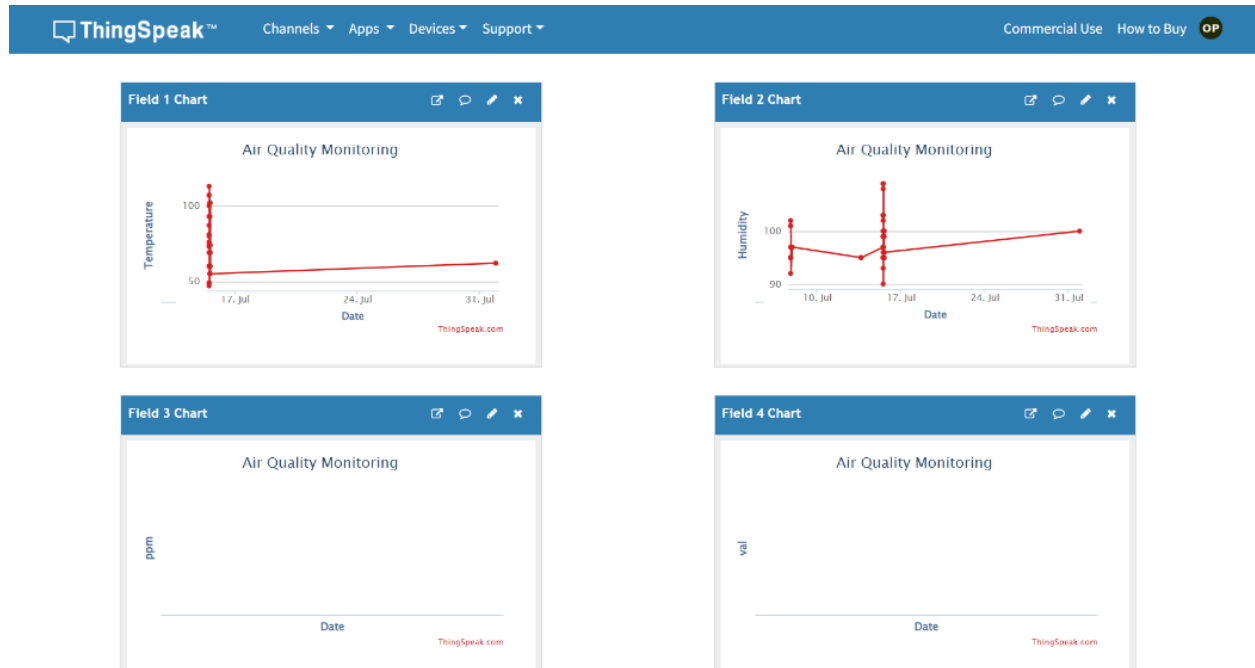
Metadata

Help

Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.

Channel Settings

- Percentage complete:** Calculated based on data entered into the various fields of a channel. Enter the name, description, location, URL, video, and tags to complete your channel.
- Channel Name:** Enter a unique name for the ThingSpeak channel.
- Description:** Enter a description of the ThingSpeak channel.
- Fields:** Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
- Metadata:** Enter information about channel data, including JSON, XML, or CSV data.
- Tags:** Enter keywords that identify the channel. Separate tags with commas.
- Link to External Site:** If you have a website that contains information about your ThingSpeak channel, specify the URL.
- Show Channel Location:**
 - Latitude:** Specify the latitude position in decimal degrees. For example, the latitude of the city of London is 51.5072.
 - Longitude:** Specify the longitude position in decimal degrees. For example, the longitude of the city of London is -0.1275.
 - Elevation:** Specify the elevation position meters. For example, the elevation of the city of London is 35.052.



Results

Thus, an Air Quality Monitoring System using ESP8266 – NODEMCU with MQ135 and DHT 11 utilizing ThingSpeak Cloud Computing has been developed and verified.