Experiment No: 5

Node MCU Cloud interfacing and programming using Thingspeak.

Name of the Student: Bhavin Patil

Div.: D

Roll No.: 78

Aim: Uploading the data on cloud using Thingspeak.

Components Required:

1) Node MCU – 1
2) Micro USB Cable – 1
3) PC/Laptop – 1
4) Connecting Wires
5) Bread Board – 1
6) Temperature Sensor LM 35

Software Required:

Arduino IDE

Theory: We all are observed LM 35 temperature sensor during practical no. 3.

Procedure:

Step 1: Include Wi-Fi and ThingSpeak directories.

Step 2: Note that we already have wi-fi directory installed. If not, install it.

Step 3: Installing Thingspeak. Goto Sketch---Include Library---Manage Libraries---Write in Library Manage 'Thingspeak'---Install latest version

Step 4: Goto www.google.com. Googlr -thingspeak login---Sign-in

Step 5: Create account if you don't have one. Use vit.edu mailID. Location India. Your name etc--- Continue

Step 6: Goto Channels --- My Channels--- New Channel

Step 7: Write Channel name, description (not mandatory)---Create two fields. Field 1- temp in degrees Celsius. Field 2- temp in Fahrenheit. Save.

Step 8: Copy Channel ID and paste it in the code. long myChannelNumber = 1587542;

Step 9: Goto API keys. Copy API key (Write API Key) and paste it in the code. const char myWriteAPIKey[] = "OMVXC2R3UOKGBNV1";

Step 10: Enter the wifi login and password in the code. (Same as in Expt 4).
WiFi.begin("Login","Password");

Step 11: Write code for reading data from LM35 temperature sensor (Same as in Expt 3). ThingSpeak.begin(client); ------ Starts thingspeak ThingSpeak.writeField (myChannelNumber, 1, tempc, myWriteAPIKey); ----- Displays temp in the field in thingspeak.

Step 12: Make hardware connections using node MCU and LM35, to sense and measure temperature.

Step 13: Upload sketch. The data (temp) will be displayed in the serial monitor. Also it will be collected and uploaded on cloud and displayed in the two fields.

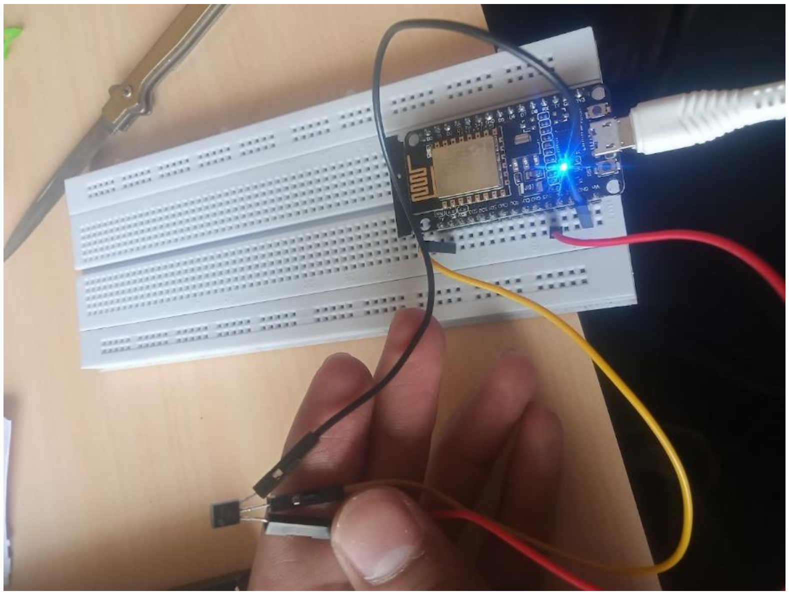
Step 13: Observe the outputs.

<u>Code:</u>

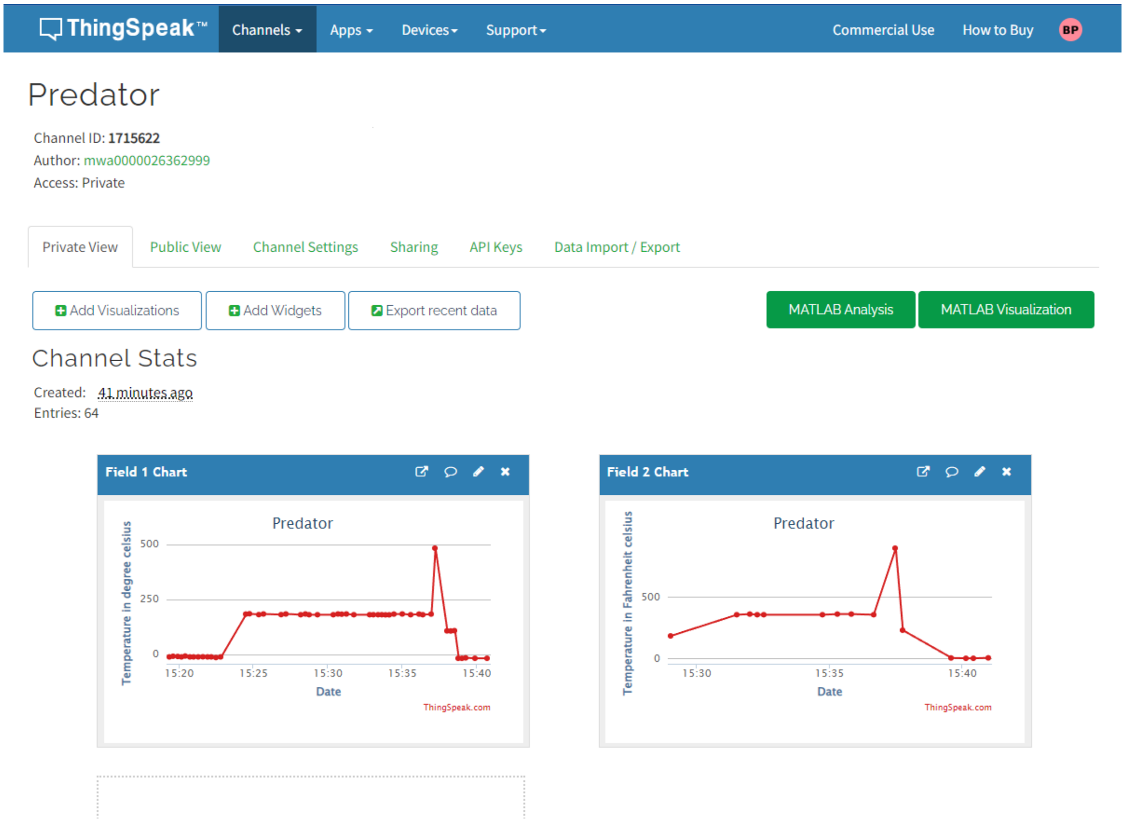
```
#include <ESP8266WiFi.h>

#include <ThingSpeak.h>

WiFiClient client;w

long myChannelNumber = 1715622;

const char myWriteAPIKey[] = "QDULOD8O4YKIDYLF";

const int sensor=A0; // Assigning analog pin A0 to variable 'sensor'
```

```
float tempc;  //variable to store temperature in degree Celsius
float tempf;  //variable to store temperature in Fahreinheit
float vout;  //temporary variable to hold sensor reading

void setup() {
  // put your setup code here, to run once:
Serial.begin(9600);
WiFi.begin("realme C25","QWERTYOP");
while(WiFi.status() !=WL_CONNECTED)
{delay(100);
Serial.print(".");
}
Serial.println();
Serial.println("NodeMCU is Connected");
Serial.println(WiFi.localIP());
ThingSpeak.begin(client);
pinMode(sensor,INPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
vout=analogRead(sensor);
vout=(vout*500)/1024-18;

tempc=vout; // Storing value in Degree Celsius

tempf=(vout*1.8)+32; // Converting to Fahrenheit

  Serial.println("Temperature in C: " + (String) tempc);

  Serial.println("Temperature in F: " + (String) tempf);

  ThingSpeak.writeField(myChannelNumber, 1, tempc, myWriteAPIKey);

  ThingSpeak.writeField(myChannelNumber, 2, tempf, myWriteAPIKey);

  delay(2000);

}
```

Result:



Setup



Predator

Channel ID: **1715622**
Author: mwa0000026362999
Access: Private

Channel Stats

Created: 41 minutes ago
Entries: 64

Thingspeak dashboard

<u>Conclusion:</u>

ThingSpeak is a platform that allows us to visualize data from sensors in a graphical format, making it easier to interpret data. This technology can also be utilized to keep an eye on our system from faraway.