Task 5: To interface DHT11 sensor with Node MCU and write a program to find the weather conditions and control the air conditioner.

Aim:

To interface DHT11 for measure Temperature and humidity using NodeMCU and display in ThingSpeak Cloud.

Apparatus:
1.NodeMCU
2.Jumper wires
3.DHT11
4.Bread board

Procedure:

Step 1: Connect the DHT11 with NodeMCU using jumper wires.

Step 2: Give the connections from DHT11 to NodeMCU by

VCC to 3v3 pin of NodeMCU

GND to GND of NodeMCU

DATA pin to D4 of NodeMCU

Step 3: After giving the connections as per schematic diagram, the USB port of NodeMCU could be connected

to PC or laptop.

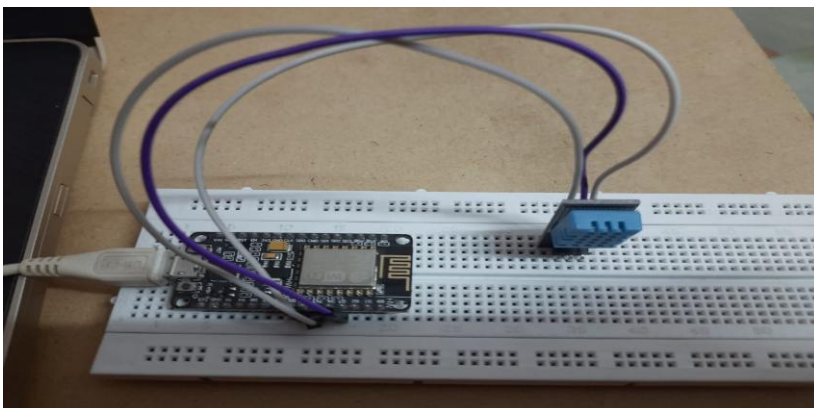Step 4:Create a new channel in thingspeak.

Step 5:Provide your channel id and key in the code

Step 6: Upload the code to the NodeMCU from Arduino.

Step 7: After the program is uploaded to NodeMCU open the serial monitor for the output.

Step 8:output can be seen in the Thingspeak as graph.

CIRCUIT DIAGRAM:

Code:

```
#include <DHT.h>
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ThingSpeak.h>
#define DHTPIN 0
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
const char* ssid = "saisrinivas";
const char* password = "Gayathris";
WiFiClient client;
unsigned long myChannelNumber = 1400536;
const char * myWriteAPIKey = "9FAQFX9YPJPXKFTG";
uint8_t temperature, humidity, k=0, l=0;
void setup()
{
 Serial.begin(115200);
dht.begin();
delay(10);
 // Connect to WiFi network
 Serial.println();
 Serial.println();
 Serial.print("Connecting to ");
 Serial.println(ssid);
 WiFi.begin(ssid, password);
 while (WiFi.status() != WL_CONNECTED)
 {
delay(500);
  Serial.print(".");
 }
 Serial.println("");
 Serial.println("WiFi connected");
 // Print the IP address
 Serial.println(WiFi.localIP());
 ThingSpeak.begin(client);
```

```cpp
}
void loop()
{
  static boolean data_state = false;
  temperature = dht.readTemperature();
  humidity = dht.readHumidity();
  Serial.print("Temperature Value is :");
  Serial.print(temperature);
  Serial.println("C");
  Serial.print("Humidity Value is :");
  Serial.print(humidity);
  Serial.println("%");
  // Write to ThingSpeak. There are up to 8 fields in a channel, allowing you to store up to 8 different
  // pieces of information in a channel. Here, we write to field 1.
if(temperature < 255)
  {
     k=temperature;
  }
if(humidity < 255)
  {
     l=humidity;
  }
if( data_state )
  {
   ThingSpeak.writeField(myChannelNumber, 1, k, myWriteAPIKey);
   data_state = false;
  }
  else
  {
   ThingSpeak.writeField(myChannelNumber, 2, l, myWriteAPIKey);
   data_state = true;
  }
delay(3000); // ThingSpeak will only accept updates every 15 seconds.
  }
```
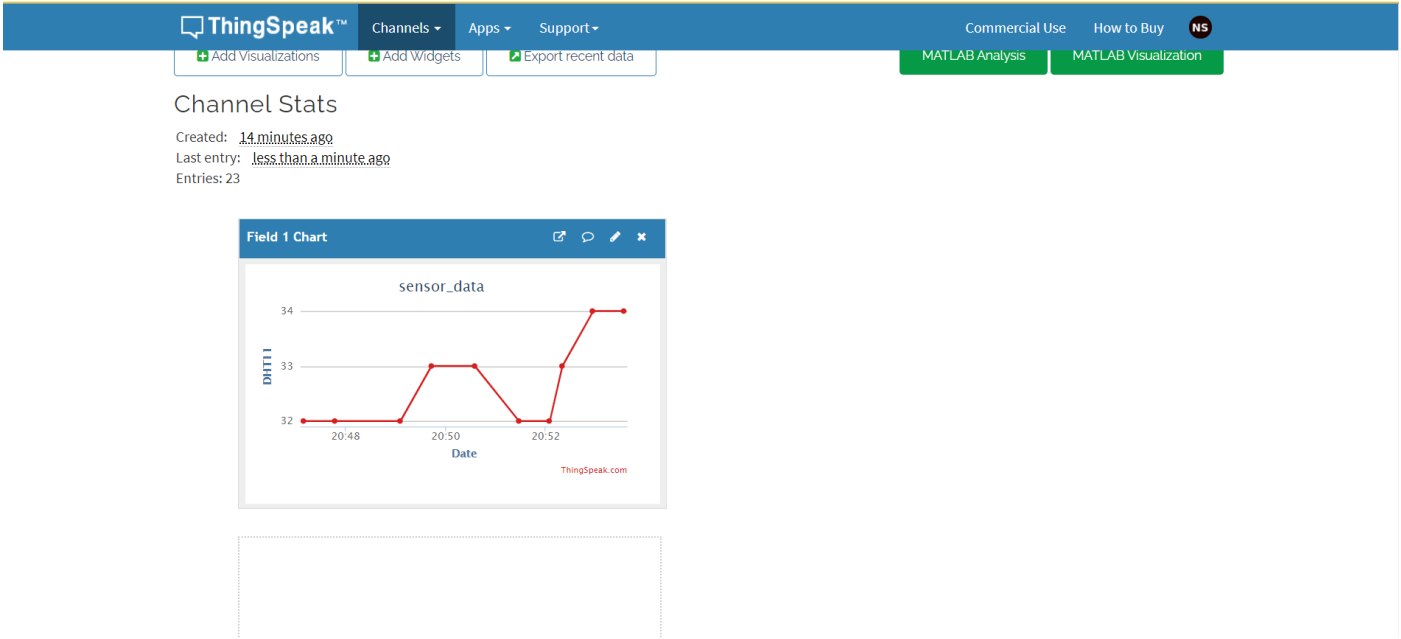
OUTPUT:

RESULT:

Thus the execution to display the DHT11 information in thingspeak is done successfully