**Practical - 4**

**Aim:**  Upload Temperature and Humidity Sensor Data

Use appropriate sensors to collect the temperature and humidity data and display the same data in web browser of laptop/mobile over WiFi.

**Components:** LED, Temperature and Humidity sensor DHT11, Jumper wires, ESP 8266 NodeMCU

**Procedure:**

* Library installation:

-Include the DHT library files (https://github.com/adafruit/DHT-sensor-library)

-Add "Adafruit\_Sensor.h" header file

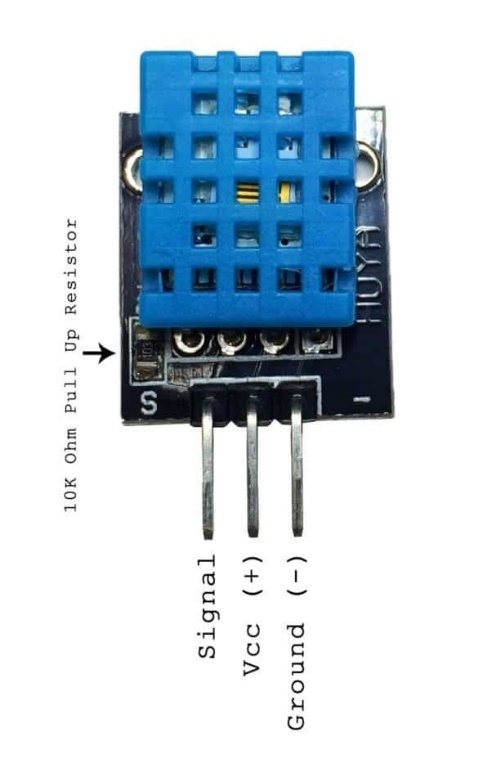
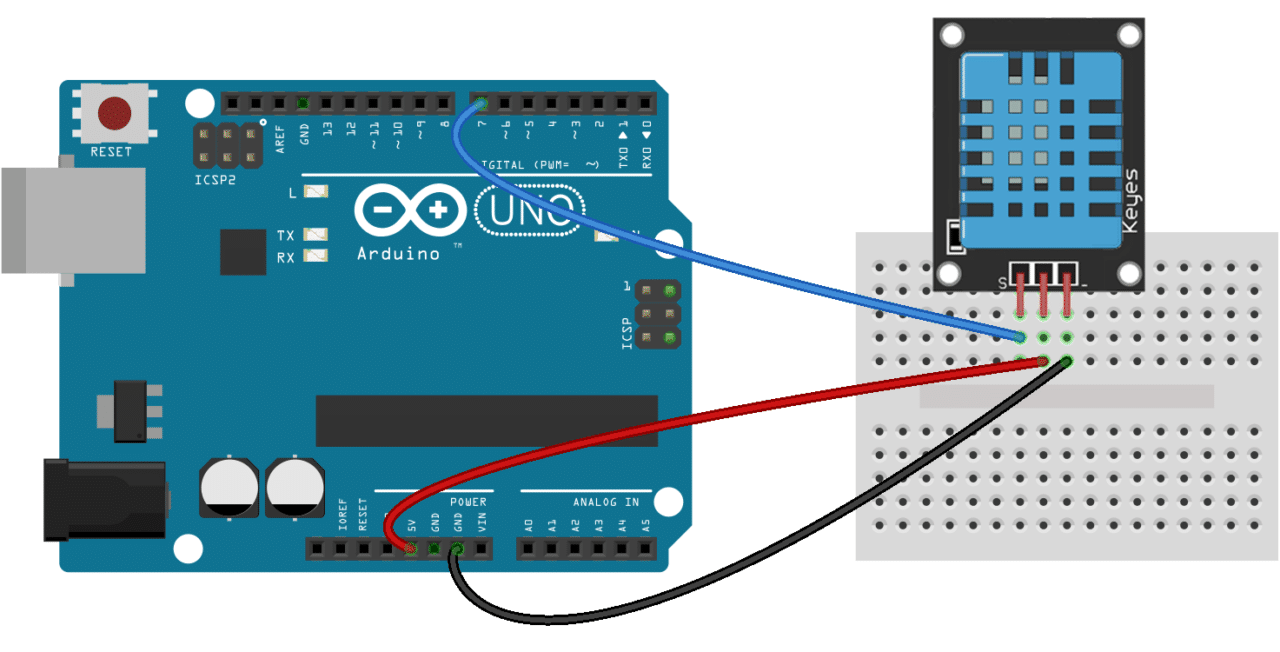
* Connections for ESP8266:

Sensor’s -Ve to GND

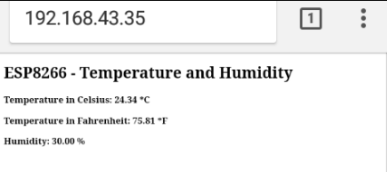
Sensor’s +Ve to 3v3

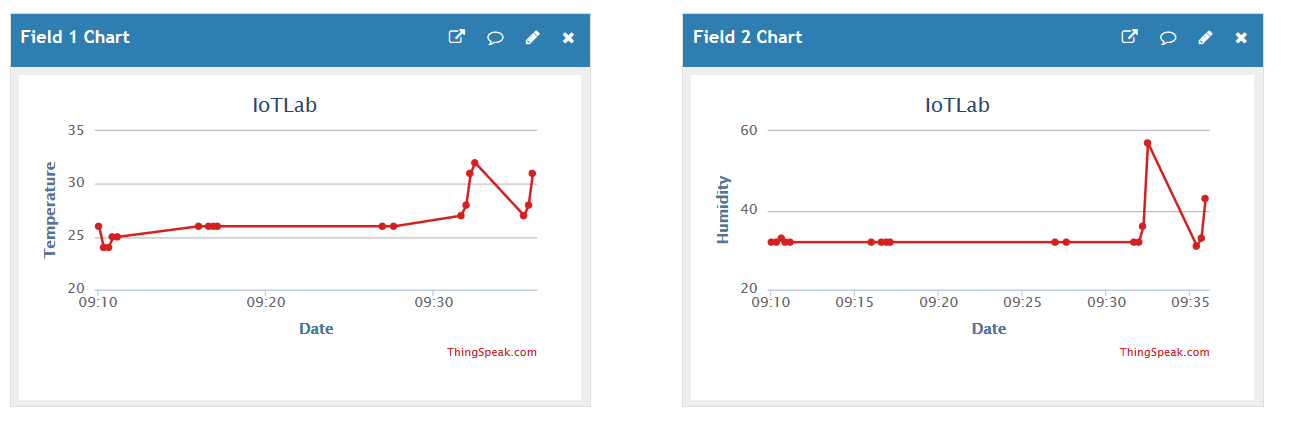
Sensor’s output to D7 (pin 13)

* Implementation in diagram



Reading of Temperature and Humidity Sensor:





* Code:

#include <ESP8266WiFi.h>

#include "Adafruit\_Sensor.h"

#include "DHT.h"

#define DHTTYPE DHT11 // DHT 11

const char\* ssid = "cs"; // Replace with your network details

const char\* password = "cscs1234";

WiFiServer server(80);

// DHT Sensor

const int DHTPin = 13;

// Initialize DHT sensor.

DHT dht(DHTPin, DHTTYPE);

// Temporary variables

static char celsiusTemp[7];

static char fahrenheitTemp[7];

static char humidityTemp[7];

void setup() {

// Initializing serial port for debugging purposes

Serial.begin(115200);

delay(10);

dht.begin();

// Connecting to WiFi network

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");

// Starting the web server

server.begin();

Serial.println("Web server running. Waiting for the ESP IP...");

delay(10000);

// Printing the ESP IP address

Serial.println(WiFi.localIP());

}

// runs over and over again

void loop() {

// Listenning for new clients

WiFiClient client = server.available();

if (client) {

Serial.println("New client");

// bolean to locate when the http request ends

boolean blank\_line = true;

while (client.connected()) {

if (client.available()) {

char c = client.read();

if (c == '\n' && blank\_line) {

// Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)

float h = dht.readHumidity();

// Read temperature as Celsius (the default)

float t = dht.readTemperature();

// Read temperature as Fahrenheit (isFahrenheit = true)

float f = dht.readTemperature(true);

// Check if any reads failed and exit early (to try again).

if (isnan(h) || isnan(t) || isnan(f)) {

Serial.println("Failed to read from DHT sensor!");

strcpy(celsiusTemp,"Failed");

strcpy(fahrenheitTemp, "Failed");

strcpy(humidityTemp, "Failed");

}

else{

// Computes temperature values in Celsius + Fahrenheit and Humidity

float hic = dht.computeHeatIndex(t, h, false);

dtostrf(hic, 6, 2, celsiusTemp);

float hif = dht.computeHeatIndex(f, h); // compute temperature using Steadman’s equation

dtostrf(hif, 6, 2, fahrenheitTemp); // converts float or double to char array (float,width,precision,char)

dtostrf(h, 6, 2, humidityTemp);

// You can delete the following Serial.print's, it's just for debugging purposes

Serial.print("Humidity: ");

Serial.print(h);

Serial.print(" %\t Temperature: ");

Serial.print(t);

Serial.print(" \*C ");

Serial.print(f);

Serial.print(" \*F\t Heat index: ");

Serial.print(hic);

Serial.print(" \*C ");

Serial.print(hif);

Serial.print(" \*F");

Serial.print("Humidity: ");

Serial.print(h);

Serial.print(" %\t Temperature: ");

Serial.print(t);

Serial.print(" \*C ");

Serial.print(f);

Serial.print(" \*F\t Heat index: ");

Serial.print(hic);

Serial.print(" \*C ");

Serial.print(hif);

Serial.println(" \*F");

}

client.println("HTTP/1.1 200 OK");

client.println("Content-Type: text/html");

client.println("Connection: close");

client.println();

// your actual web page that displays temperature and humidity

client.println("<!DOCTYPE HTML>");

client.println("<html>");

client.println("<head></head>

<body><h1>ESP8266 - Temperature and Humidity</h1><h3>Temperature in Celsius: ");

client.println(celsiusTemp);

client.println("\*C</h3><h3>Temperature in Fahrenheit: ");

client.println(fahrenheitTemp);

client.println("\*F</h3><h3>Humidity: ");

client.println(humidityTemp);

client.println("%</h3><h3>");

client.println("</body></html>");

break;

}

if (c == '\n') {

// when starts reading a new line

blank\_line = true;

}

else if (c != '\r') {

// when finds a character on the current line

blank\_line = false;

}

}

}

// closing the client connection

delay(1);

client.stop();

Serial.println("Client disconnected.");

}

}