TRAINING SMARTFUSION

MINI PROJECT 1:

Title: Read Sensor Data and Log It to Cloud (ThingSpeak)

Submitted By:

Kashish Gujral

(2302584)

GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA

INTRODUCTION :

In this mini project, I developed an IoT system using an ESP32 microcontroller and a DHT22 sensor to read temperature and humidity values. I connected the ESP32 to Wi-Fi and uploaded the data to ThingSpeak, a cloud-based IoT analytics platform. The project was simulated using the Wokwi IoT Simulator.

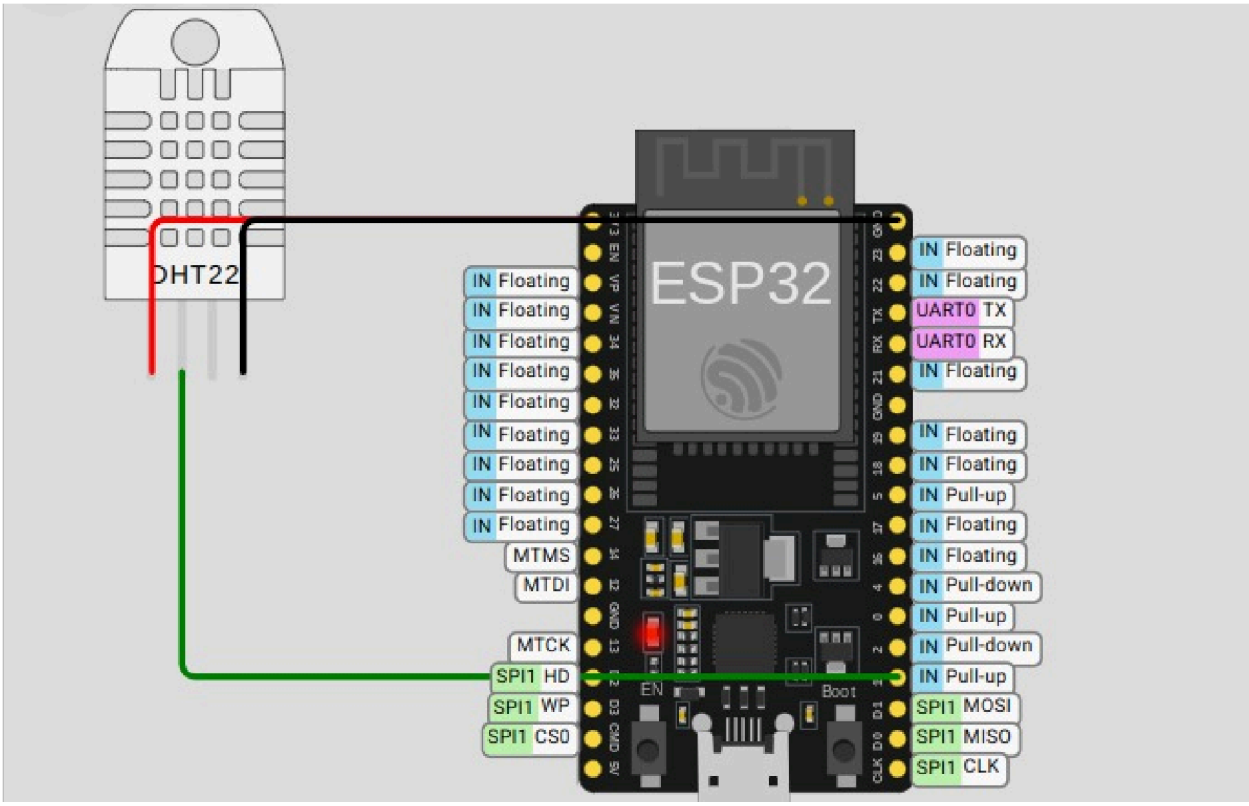
COMPONENT AND CONNECTIONS :

S. No.	Component	Description
1.	ESP32	WiFi enabled
2.	DHT22 sensor	Temp and humidity
3.	Wokwi Simulator	Online simulation
4.	ThingSpeak	Cloud platform

PIN CONNECTIONS:

DHT22	ESP32 Pin

VCC	3.3V
GND	GND
DATA	GPIO15



CODE:

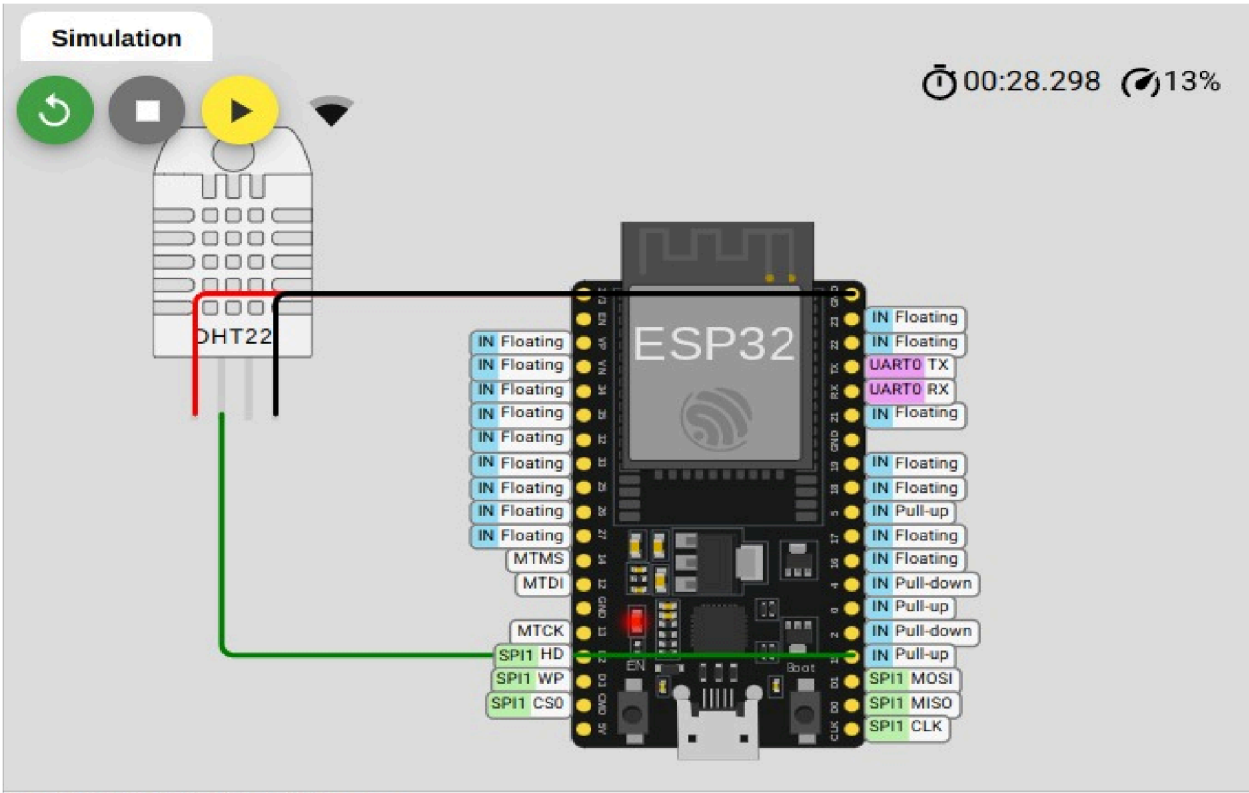
```
#include <WiFi.h>
#include "DHT.h"
#include <HTTPClient.h>
```

```
#define DHTPIN 15         // GPIO where data pin of DHT22 is connected
#define DHTTYPE DHT22     // Sensor type

const char* ssid = "Wokwi-GUEST"; // Wokwi WiFi const
char* password = "";

String apiKey = "1KTX1L0K197G17J5";  // Replace with your ThingSpeak Write API Key

const char* server = "http://api.thingspeak.com/update";

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);   dht.begin();

  WiFi.begin(ssid, password);
  Serial.print("Connecting to WiFi");   while (WiFi.status() != WL_CONNECTED) {
delay(500);

    Serial.print(".");
  }
  Serial.println("\nWiFi connected");
}

void loop() {
```

```
  float temperature = dht.readTemperature();   float
humidity = dht.readHumidity();


  if (isnan(temperature) || isnan(humidity)) {
Serial.println("Failed to read from DHT sensor!");     return;

  }

  Serial.println("Temp: " + String(temperature) + " °C");
  Serial.println("Humidity: " + String(humidity) + " %");


  if (WiFi.status() == WL_CONNECTED) {
    HTTPClient http;
    String url = server + String("?api_key=") + apiKey +
                 "&field1=" + String(temperature) +
"&field2=" + String(humidity);     http.begin(url);

    int httpResponseCode = http.GET();     if
(httpResponseCode > 0) {

      Serial.println("Data sent to ThingSpeak!");
    } else {
      Serial.println("Error sending data.");
    }
    http.end();
  }
```

```
  delay(15000);  // Send data every 15 seconds
}
```

Output:




```
Connecting to WiFi...
WiFi connected
Temp: 24.00 °C
Humidity: 40.00 %
Data sent to ThingSpeak!
```

ThingSpeak Graphs :