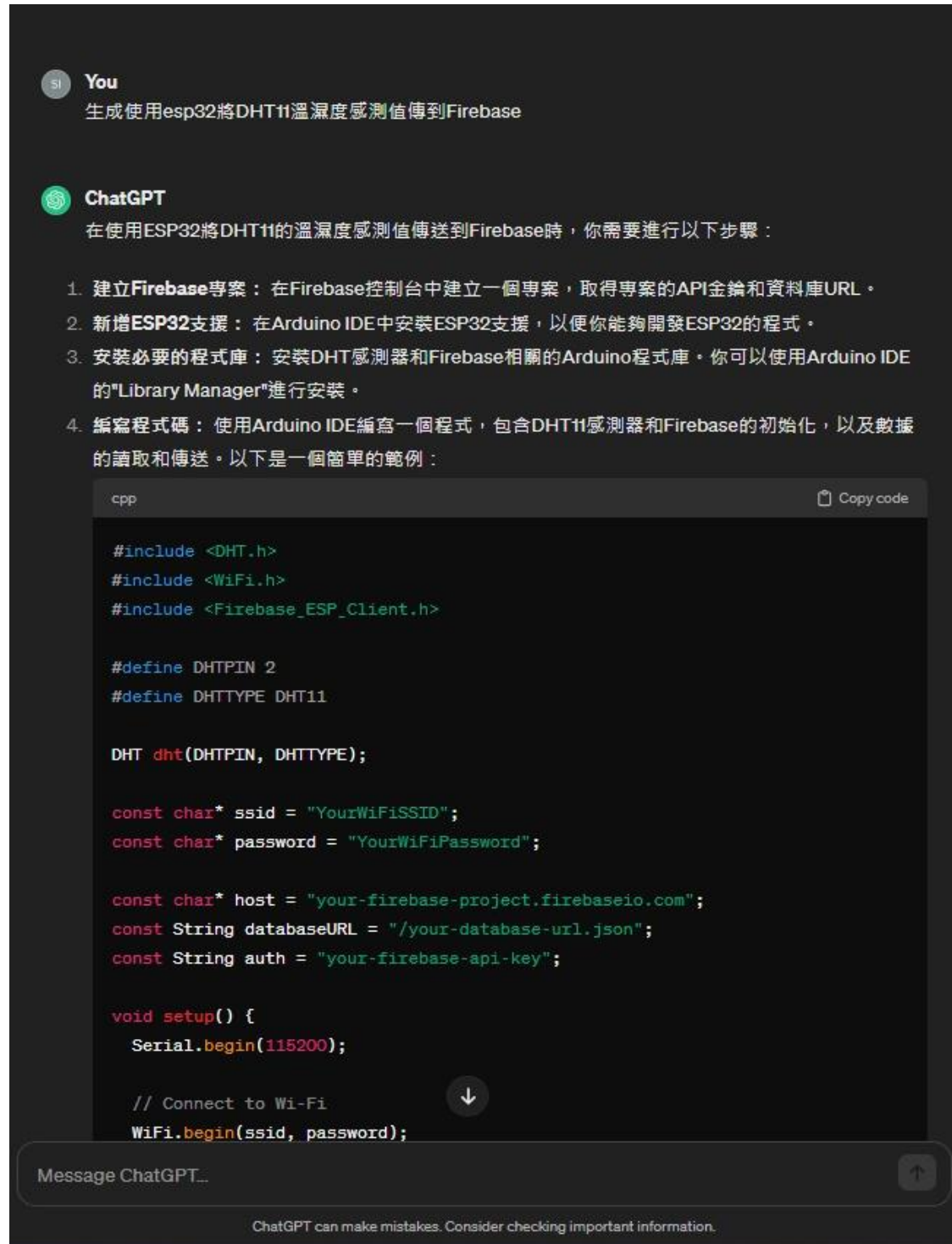


HW2

用 ChatGPT prompt 生成使用 esp32 將 DHT11 溫濕度感測值傳到 Firebase，將過程和 code 結果截圖並實作。

prompt 過程：



The screenshot shows a ChatGPT interface with a dark theme. The user's prompt is: "生成使用esp32將DHT11溫濕度感測值傳到Firebase". The ChatGPT response provides a step-by-step guide and a C++ code snippet.

You
生成使用esp32將DHT11溫濕度感測值傳到Firebase

ChatGPT
在使用ESP32將DHT11的溫濕度感測值傳送到Firebase時，你需要進行以下步驟：

1. 建立**Firebase**專案：在Firebase控制台中建立一個專案，取得專案的API金鑰和資料庫URL。
2. 新增**ESP32**支援：在Arduino IDE中安裝ESP32支援，以便你能夠開發ESP32的程式。
3. 安裝必要的程式庫：安裝DHT感測器和Firebase相關的Arduino程式庫。你可以使用Arduino IDE的"Library Manager"進行安裝。
4. 編寫程式碼：使用Arduino IDE編寫一個程式，包含DHT11感測器和Firebase的初始化，以及數據的讀取和傳送。以下是一個簡單的範例：

```
cpp Copy code

#include <DHT.h>
#include <WiFi.h>
#include <Firebase_ESP_Client.h>

#define DHTPIN 2
#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

const char* ssid = "YourWiFiSSID";
const char* password = "YourWiFiPassword";

const char* host = "your-firebase-project.firebaseio.com";
const String databaseURL = "/your-database-url.json";
const String auth = "your-firebase-api-key";

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

  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
```

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

```
const String auth = "your-firebase-api-key";

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

  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");

  // Initialize Firebase
  Firebase.begin(host, auth, databaseURL);

  // Initialize DHT sensor
  dht.begin();
}

void loop() {
  // Read sensor values
  float temperature = dht.readTemperature();
  float humidity = dht.readHumidity();

  // Send data to Firebase
  Firebase.setFloat("temperature", temperature);
  Firebase.setFloat("humidity", humidity);

  delay(5000); // Delay for 5 seconds before sending next data
}
```

請替換程式中的"YourWiFiSSID"、"YourWiFiPassword"、"your-firebase-project"、"your-database-url"和"your-firebase-api-key"等資訊為你的實際數據。

5. 燒錄程式：使用Arduino IDE將你的程式燒錄到ESP32板上。

這樣，你的ESP32就應該能夠讀取DHT11的溫度數據，並將其傳送到Firebase資料庫中。

Message ChatGPT...

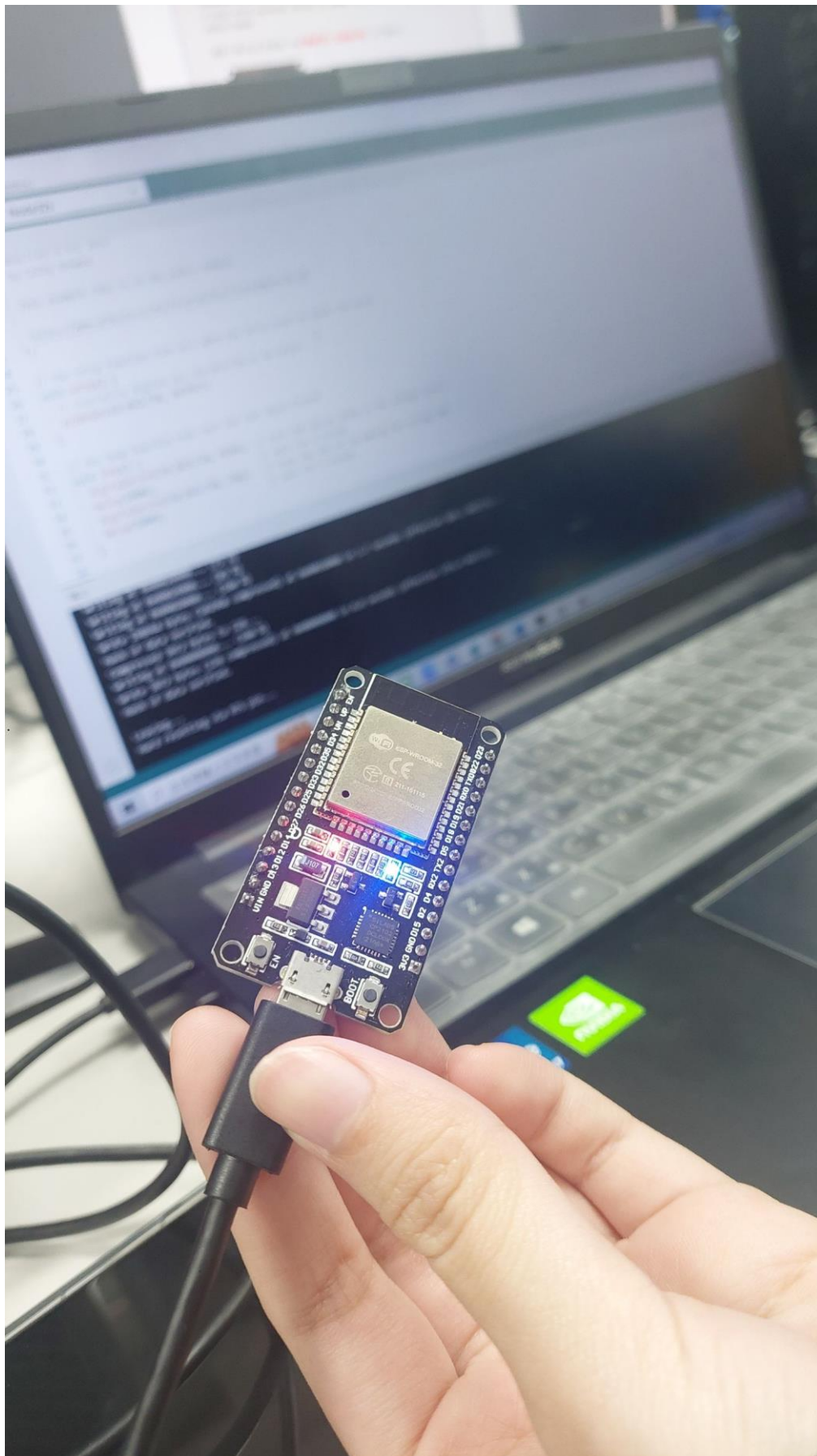


ChatGPT can make mistakes. Consider checking important information.

實驗 code :

```
1  #include <ESP32Firebase.h>
2  #include <WiFi.h>
3  #include <SimpleDHT.h>
4
5  #define DHTPIN 4
6  // #define DHTTYPE SimpleDHT11
7
8  SimpleDHT11 dht11(DHTPIN);
9
10 const char* ssid = "123";
11 const char* password = "878787";
12
13 const char* host = "https://test2-7a221-default-rtdb.firebaseio.com/";
14 // const String databaseURL = "/test2-7a221-default-rtdb-export.json";
15
16 Firebase firebase(host);
17
18 void setup() {
19     Serial.begin(9600);
20     // Connect to Wi-Fi
21     WiFi.begin(ssid, password);
22     while (WiFi.status() != WL_CONNECTED) {
23         delay(1000);
24         Serial.println("Connecting to WiFi...");
25     }
26     Serial.println("Connected to WiFi");
27 }
28
29 void loop() {
30     byte temperature = 0;
31     byte humidity = 0;
32     int err = SimpleDHTErrSuccess;
33     // start working...
34     Serial.println("=====");
35     if ((err = dht11.read(DHTPIN, &temperature, &humidity, NULL)) != SimpleDHTErrSuccess) {
36         Serial.print("Read DHT11 failed, err="); Serial.println(err); delay(1000);
37         return;
38     }
39     Serial.print("Humidity = ");
40     Serial.print((int)humidity);
41     Serial.print("% , ");
42     Serial.print("Temperature = ");
43     Serial.print((int)temperature);
44     Serial.println("C ");
45
46     String pushData = "temperature : " + String((int)temperature) + " / humidity : " + String((int)humidity);
47     firebase.pushString("test", pushData);
48
49     delay(1000); //每3秒顯示一次
50 }
```

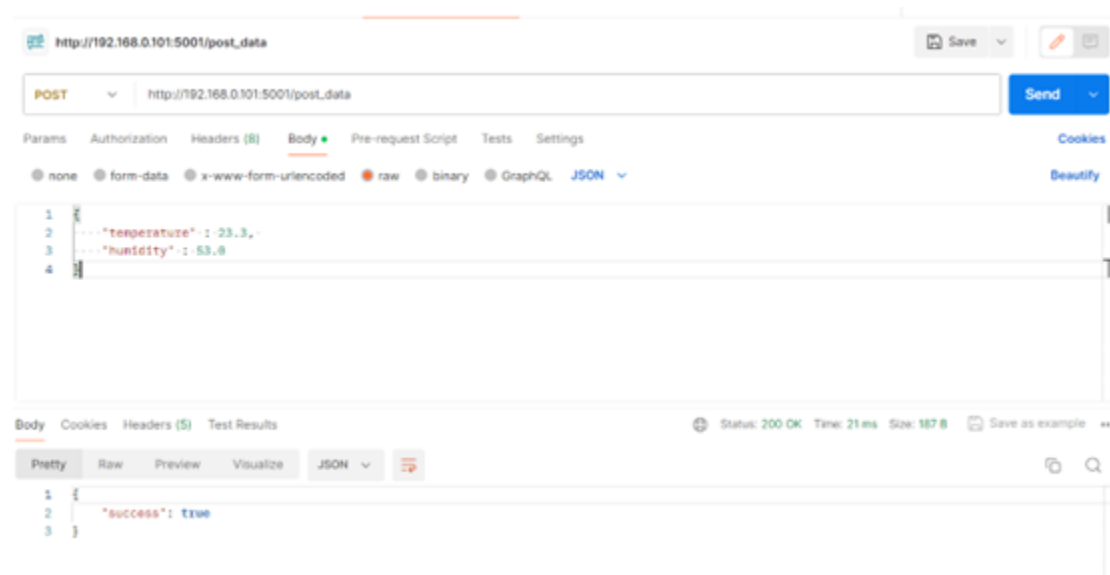
接線拍照：



請描述本次實驗的整個架構：

1. 硬體：
 - 使用 ESP32 微控制器和 DHT11 溫濕度感測器。ESP32 是一個廣泛使用的 Wi-Fi 和藍牙模組，而 DHT11 是一種廉價的數字溫濕度感測器。
2. 建立 Firebase 專案：
 - 在 Firebase 上建立一個專案，這將提供一個用於儲存感測數據的雲端數據庫。
3. 寫 ESP32 的 code：
 - 編寫 ESP32 的程式碼，以初始化 Wi-Fi 連接和 DHT11 感測器，並獲取溫濕度數據。
 - 將數據傳輸到 Firebase。這需要使用 Firebase 的 API 來建立連接並上傳數據。
4. ChatGPT：
 - 使用 ChatGPT 模型來生成代碼。可能的 prompt 可以包括關於如何初始化 ESP32、如何讀取 DHT11 數據、如何建立與 Firebase 的連接等。
5. 生成 code：
 - 使用 ChatGPT 提供的代碼生成功能，根據 prompt 生成獲得的建議代碼。
 - 確保生成的代碼符合 ESP32 和 Firebase 的庫和 API。
6. 實作和 debug：
 - 將生成的代碼實作到 ESP32 上，並進行測試和調試。
 - 確保 ESP32 能夠正確地讀取 DHT11 數據並將其上傳到 Firebase。

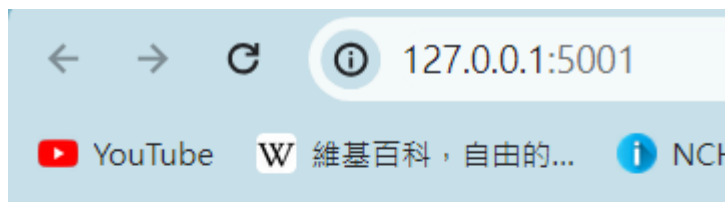
POSTMAN:



Arduino:

```
Temperature: 24 °C      Humidity: 54 %  
Server Address: http://192.168.0.104:5001/post_data  
HTTP Request failed: connection refused
```

frontend:



Sensor Data

Temperature: °C

Humidity: %