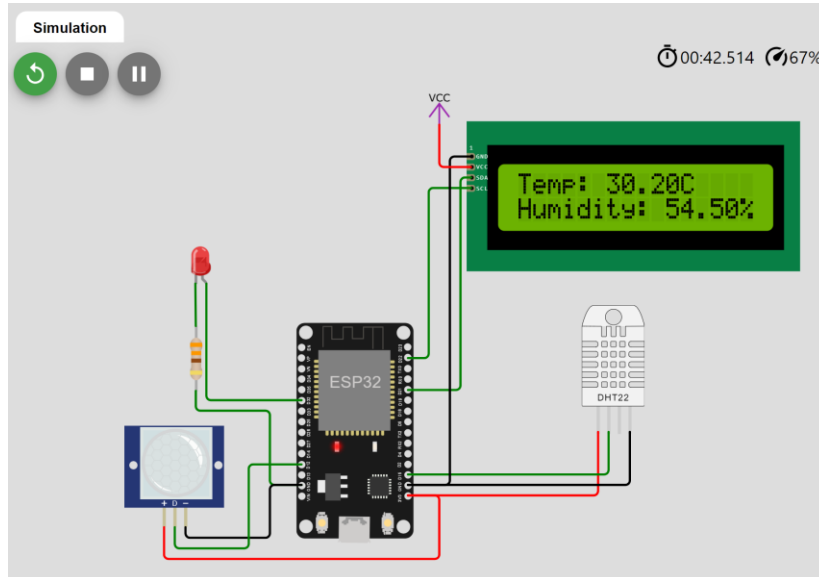# Bài 5. ESP32 gửi dữ liệu HTTP request

Mục đích:

Lập trình cho ESP32 gửi dữ liệu (nhiệt độ, độ ẩm như đã thu thập ở bài 4) lên server

Sử dụng công cụ mô phỏng wokwi https://wokwi.com/

Hoạt động của chương trình:

a) Gửi dữ liệu HTTP get request, dữ liệu đóng gói url-encoded

b) Gửi dữ liệu HTTP post request, dữ liệu đóng gói url-encoded

c) Gửi dữ liệu HTTP post request, dữ liệu đóng gói json (trong request body)

Sử dụng server có sẵn:

https://postman-echo.com/get

https://postman-echo.com/post

Bài làm

Họ và tên: Lê Đức Minh

MSSV: 20200395

a) Gửi dữ liệu HTTP get request, dữ liệu đóng gói url-encoded

Link: https://wokwi.com/projects/381808207244992513

Code:

sketch.ino   diagram.json   libraries.txt   Library Manager ▼

```cpp
1   #include <WiFi.h>
2   #include <HTTPClient.h>
3   #include "DHTesp.h"
4   #define DHT_PIN 15
5
6   String serverName = "https://postman-echo.com/get";
7   DHTesp dhtSensor;
8
9   void setup() {
10    dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
11    //setup for serial communication
12    Serial.begin(9600);
13    Serial.print("Connecting to WiFi");
14    //setup for WiFi connection
15    WiFi.begin("Wokwi-GUEST", "", 6);
16    while (WiFi.status() != WL_CONNECTED) {
17      delay(100);
18      Serial.print(".");
19    }
20    Serial.println("WiFi Connected!");
21  }
22
```

```
22
23  void loop() {
24    static uint32_t timer_for_dht = millis();
25    uint32_t cur_time = millis();
26
27    if (cur_time - timer_for_dht > 10000 && WiFi.status() == WL_CONNECTED) {
28      HTTPClient http;
29      TempAndHumidity data = dhtSensor.getTempAndHumidity();
30      float temp = data.temperature;
31      float humid = data.humidity;
32      String serverPath = serverName + "?temp=" + String(temp) + "&humid=" + String(humid);
33      http.begin(serverPath.c_str()); // Send HTTP GET request
34
35      int httpResponseCode = http.GET();
36
37      if (httpResponseCode > 0) {
38        Serial.print("HTTP Response code: ");
39        Serial.println(httpResponseCode);
40        String payload = http.getString();
41        Serial.println(payload);
42      }
43      else {
44        Serial.print("Error code: ");
45        Serial.println(httpResponseCode);
46      }
47      http.end();
48      timer_for_dht = cur_time; // restart timer
49    }
50    else if (WiFi.status() != WL_CONNECTED) {
51      Serial.println("WiFi disconnected");
52    }
53  }
54
```

Kết quả:

```
Connecting to WiFi.............WiFi Connected!
HTTP Response code: 200
{
  "args": {
    "temp": "24.00",
    "humid": "40.00"
  },
  "headers": {
    "x-forwarded-proto": "https",
    "x-forwarded-port": "443",
    "host": "postman-echo.com",
    "x-amzn-trace-id": "Root=1-6559c5cb-530ad01247b0706568633028",
    "user-agent": "ESP32HTTPClient",
    "accept-encoding": "identity;q=1,chunked;q=0.1,*;q=0"
  },
  "url": "https://postman-echo.com/get?temp=24.00&humid=40.00"
}
```

b) Gửi dữ liệu HTTP POST request, dữ liệu đóng gói url-encoded

Link: https://wokwi.com/projects/381811633909777409

Code:

sketch.ino   diagram.json   libraries.txt   Library Manager ▼

```cpp
1    #include <WiFi.h>
2    #include <HTTPClient.h>
3    #include "DHTesp.h"
4    String serverName = "https://postman-echo.com/post";
5    #define DHT_PIN 15
6    DHTesp dhtSensor;
7
8    void setup() {
9      dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
10     //setup for serial communication
11     Serial.begin(9600);
12     Serial.print("Connecting to WiFi");
13     //setup for WiFi connection
14     WiFi.begin("Wokwi-GUEST", "", 6);
15     while (WiFi.status() != WL_CONNECTED) {
16       delay(100);
17       Serial.print(".");
18     }
19     Serial.println("WiFi Connected!");
20   }
21
```

```
22  void loop() {
23    static uint32_t timer_for_dht = millis();
24    uint32_t cur_time = millis();
25
26    if (cur_time - timer_for_dht > 10000 && WiFi.status() == WL_CONNECTED) {
27      HTTPClient http;
28      TempAndHumidity data = dhtSensor.getTempAndHumidity();
29      float temp = data.temperature;
30      float humid = data.humidity;
31      String serverPath = serverName + "?temp=" + String(temp) + "&humid=" + String(humi
32
33      http.begin(serverPath.c_str()); // Send HTTP GET request
34      http.addHeader("Content-Type", "IoT/HW5");
35      String httpRequestData = "&temp=" + String(temp) + "&humid=" + String(humid);
36      int httpResponseCode = http.POST(httpRequestData);
37      if (httpResponseCode > 0) {
38        Serial.print("HTTP Response code: "); Serial.println(httpResponseCode);
39        String payload = http.getString();
40        Serial.println(payload);
41      }
42      else {
43        Serial.print("Error code: ");
44        Serial.println(httpResponseCode);
45      }
46      http.end();
47      timer_for_dht = cur_time; // restart timer
48    }
49    else if(WiFi.status()!=WL_CONNECTED){
50      Serial.println("WiFi disconnected");
51    }
52  }
53
```

Kết quả:

```
HTTP Response code: 200
{
  "args": {
    "temp": "24.00",
    "humid": "40.00"
  },
  "data": {},
  "files": {},
  "form": {},
  "headers": {
    "x-forwarded-proto": "https",
    "x-forwarded-port": "443",
    "host": "postman-echo.com",
    "x-amzn-trace-id": "Root=1-6559cb3d-5eaab94166b837094514029d",
    "content-length": "23",
    "user-agent": "ESP32HTTPClient",
    "accept-encoding": "identity;q=1,chunked;q=0.1,*;q=0",
    "content-type": "IoT/HW5"
  },
  "json": null,
  "url": "https://postman-echo.com/post?temp=24.00&humid=40.00"
}
```

c) Gửi dữ liệu qua HTTP POST request, dữ liệu đóng gói json trong body request

Link: https://wokwi.com/projects/381812596364173313

Code:

sketch.ino   diagram.json   libraries.txt   Library Manager   ▼

```arduino
1   #include <WiFi.h>
2   #include <HTTPClient.h>
3   #include "DHTesp.h"
4   String serverName = "https://postman-echo.com/post";
5   #define DHT_PIN 15
6   DHTesp dhtSensor;
7
8   void setup() {
9     dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
10    //setup for serial communication
11    Serial.begin(9600);
12    Serial.print("Connecting to WiFi");
13    //setup for WiFi connection
14    WiFi.begin("Wokwi-GUEST", "", 6);
15    while (WiFi.status() != WL_CONNECTED) {
16      delay(100);
17      Serial.print(".");
18    }
19    Serial.println("WiFi Connected!");
20  }
21
22  void loop() {
23    static uint32_t timer_for_dht = millis();
24    uint32_t cur_time = millis();
25
26
```

```
22  void loop() {
23    static uint32_t timer_for_dht = millis();
24    uint32_t cur_time = millis();
25
26
27    if (cur_time - timer_for_dht > 10000 && WiFi.status() == WL_CONNECTED) {
28      HTTPClient http;
29      TempAndHumidity data = dhtSensor.getTempAndHumidity();
30      float temp = data.temperature;
31      float humid = data.humidity;
32
33      http.begin(serverName);
34      http.addHeader("Content-Type", "application/json");
35      String httpRequestData = "{\"temp\":" + String(temp) + ",\"humid\": "+ String(humi
36      int httpResponseCode = http.POST(httpRequestData);
37
38      if (httpResponseCode > 0) {
39        Serial.print("HTTP Response code: ");
40        Serial.println(httpResponseCode);
41        String payload = http.getString();
42        Serial.println(payload);
43      }
44      else {
45        Serial.print("Error code: ");
46        Serial.println(httpResponseCode);
47      }
48      http.end();
49      timer_for_dht = cur_time; // restart timer
50    }
51    else if(WiFi.status()!=WL_CONNECTED){
52      Serial.println("WiFi disconnected");
53    }
54  }
```

Kết quả:

```
HTTP Response code: 200
{
  "args": {},
  "data": {
    "temp": 24,
    "humid": 40
  },
  "files": {},
  "form": {},
  "headers": {
    "x-forwarded-proto": "https",
    "x-forwarded-port": "443",
    "host": "postman-echo.com",
    "x-amzn-trace-id": "Root=1-6559cd60-434ec6161449d4675cf7b2d2",
    "content-length": "30",
    "user-agent": "ESP32HTTPClient",
    "accept-encoding": "identity;q=1,chunked;q=0.1,*;q=0",
    "content-type": "application/json"
  },
  "json": {
    "temp": 24,
    "humid": 40
  },
  "url": "https://postman-echo.com/post"
}
```