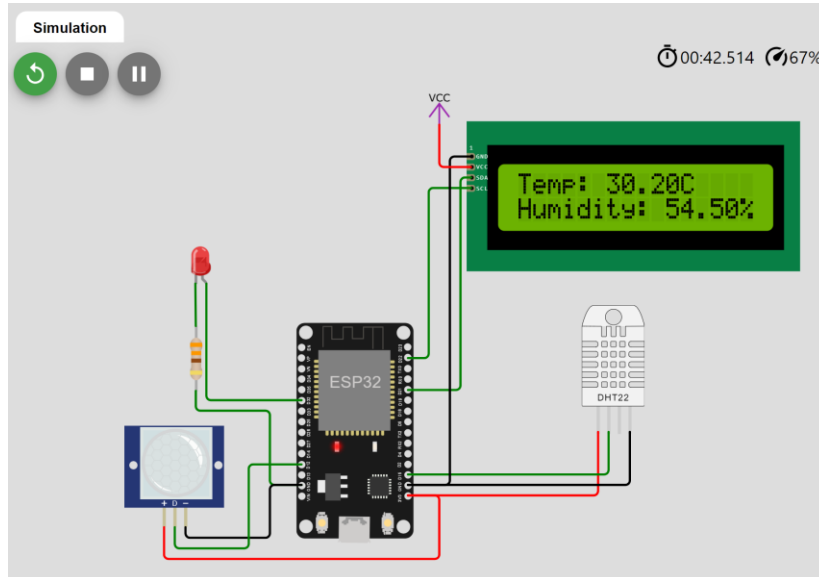


Bài 5. ESP32 gửi dữ liệu HTTP request

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:

- Gửi dữ liệu HTTP get request, dữ liệu đóng gói url-encoded
- Gửi dữ liệu HTTP post request, dữ liệu đóng gói url-encoded
- 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>

Yêu cầu nộp bài:

Viết báo cáo mô tả công việc thực hiện gồm:

- Code của chương trình
- Link bài mô phỏng trên wokwi
- Chụp ảnh màn hình kết quả thực hiện mô phỏng.

Submit: file báo cáo (kèm link wokwi bài mô phỏng)

Hướng dẫn tham khảo:

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

```
#include <WiFi.h>
#include <HTTPClient.h>
String serverName = "https://postman-echo.com/get";
void setup() {
    //setup for serial communication
    Serial.begin(9600);
    Serial.print("Connecting to WiFi");
    //setup for WiFi connection
    WiFi.begin("Wokwi-GUEST", "", 6);
    while (WiFi.status() != WL_CONNECTED) {
        delay(100);
        Serial.print(".");
    }
    Serial.println("WiFi Connected!");
}
void loop() {
    if (WiFi.status() == WL_CONNECTED) {
        HTTPClient http;
        String serverPath = serverName + "?temp=24.7&humid=30";
        http.begin(serverPath.c_str()); // Send HTTP GET request
        int httpResponseCode = http.GET();
        if (httpResponseCode > 0) {
            Serial.print("HTTP Response code: "); Serial.println(httpResponseCode);
            String payload = http.getString();
            Serial.println(payload);
        }
        else {
            Serial.print("Error code: "); Serial.println(httpResponseCode);
        }
        http.end();
    }
    else {
        Serial.println("WiFi Disconnected");
    }
    delay(3000);
}
```

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

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

```

String serverName = "https://postman-echo.com/post";

void setup() {
  //setup for serial communication
  //.....
}

void loop() {
  if (WiFi.status() == WL_CONNECTED) {
    HTTPClient http;
    float x = 30.5; //get temperature value
    float y = 78;   //get humidity value

    http.begin(serverName);
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");
    String httpRequestData = "&temp=" + String(x) + "&humid=" + String(y);
    Serial.println(httpRequestData);
    int httpResponseCode = http.POST(httpRequestData);

    if (httpResponseCode > 0) {
      Serial.print("HTTP Response code: "); Serial.println(httpResponseCode);
      String payload = http.getString();
      Serial.println(payload);
    }
    else {
      Serial.print("Error code: "); Serial.println(httpResponseCode);
    }
    http.end();
  }
  else {
    Serial.println("WiFi Disconnected");
  }
  delay(3000);
}

```

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

```

#include <WiFi.h>
#include <HTTPClient.h>
#include <ArduinoJson.h>

String serverName = "https://postman-echo.com/post";

void setup() {

```

```

//setup for serial communication
Serial.begin(9600);
Serial.print("Connecting to WiFi");
//setup for WiFi connection
WiFi.begin("Wokwi-GUEST", "", 6);
while (WiFi.status() != WL_CONNECTED) {
    delay(100);
    Serial.print(".");
}
Serial.println("WiFi Connected!");
}

void loop() {
    if (WiFi.status() == WL_CONNECTED) {
        HTTPClient http;
        float x = 30.5; //get temperature value
        float y = 78;   //get humidity value

        http.begin(serverName);
        http.addHeader("Content-Type", "application/json");
        String httpRequestData = "{\"temp\": 24.25,\"humid\": 49.54 }";
        //String httpRequestData = "{\"temp\": " + String(x) + ",\"humid\": " +
String(y) + "}";
        Serial.println(httpRequestData);
        int httpResponseCode = http.POST(httpRequestData);

        if (httpResponseCode > 0) {
            Serial.print("HTTP Response code: "); Serial.println(httpResponseCode);
            String payload = http.getString();
            Serial.println(payload);
        }
        else {
            Serial.print("Error code: "); Serial.println(httpResponseCode);
        }
        http.end();
    }
    else {
        Serial.println("WiFi Disconnected");
    }
    delay(3000);
}

```

```
float x = 30.5;
float y = 78;

DynamicJsonDocument doc(1024);
String jsonstr;
JsonObject root = doc.to<JsonObject>();
root["temperature"] = x;
root["humidity"] = y;
serializeJson(doc, jsonstr);
String httpRequestData = jsonstr;
```

<https://arduinojson.org/v6/example/string/>