

Tutoriel de Contrôle de LED avec ESP32 et Blynk

Enseignant Mustapha jaballah

November 15, 2024

1 Introduction

Dans ce projet, nous allons connecter un ESP32 à l'application Blynk pour contrôler des LED. Nous utiliserons deux réseaux Wi-Fi pour garantir une connexion stable et nous afficherons le code bien commenté pour faciliter la compréhension.

2 Code

Voici le code complet avec des commentaires pour expliquer chaque étape :

```
1 #define BLYNK_TEMPLATE_ID "TMPL20-2CmbzB" // ID du mod le Blynk
2 #define BLYNK_TEMPLATE_NAME "esp32" // Nom du mod le Blynk
3 #define BLYNK_AUTH_TOKEN "ZIhNsW3ZNKN8ig-v5FSdR1oqEakOUYAr" //
  Token d'authentification pour Blynk
4
5 #include <WiFi.h> // Biblioth que Wi-Fi pour ESP32
6 #include <WiFiClient.h> // Biblioth que Wi-Fi client
7 #include <BlynkSimpleEsp32.h> // Biblioth que Blynk pour ESP32
8
9 String ssid1 = "Galaxy A1273B2"; // Premier SSID Wi-Fi
10 String pass1 = "awat3625"; // Mot de passe du premier Wi-Fi
11 String ssid2 = "Caf magreb"; // Deuxi me SSID Wi-Fi
12 String pass2 = "87654321"; // Mot de passe du deuxi me Wi-
  Fi
13
14 int ledPin = 2; // Pin de la premi re LED (GPIO 2)
15 int ledPin1 = 17; // Pin de la deuxi me LED (GPIO 17)
16
17 void setup() {
18   Serial.begin(115200); // Initialisation du port
19   s rie pour d bogage
20   pinMode(ledPin, OUTPUT); // D finir la pin de la LED 1
  comme sortie
21   pinMode(ledPin1, OUTPUT); // D finir la pin de la LED 2
  comme sortie
```

```

22   connectToWiFi(); // Tentative de connexion aux
      r seaux Wi-Fi
23   Blynk.begin(BLYNK_AUTH_TOKEN, WiFi.SSID().c_str(), WiFi.psk().
      c_str()); // Connexion Blynk
24 }
25
26 // Fonction pour connecter l'ESP32 un r seau Wi-Fi
27 void connectToWiFi() {
28   Serial.println("Tentative de connexion au Wi-Fi...");
29
30   WiFi.begin(ssid1.c_str(), pass1.c_str()); // Connexion au
      premier r seau Wi-Fi
31   if (tryWiFiConnection()) {
32     Serial.println("Connect au r seau Wi-Fi : " + ssid1);
33     return;
34   }
35
36   WiFi.begin(ssid2.c_str(), pass2.c_str()); // Connexion au
      second r seau Wi-Fi
37   if (tryWiFiConnection()) {
38     Serial.println("Connect au r seau Wi-Fi : " + ssid2);
39     return;
40   }
41
42   Serial.println(" chec de la connexion un r seau Wi-Fi.");
43 }
44
45 // Fonction pour tester la connexion Wi-Fi
46 bool tryWiFiConnection() {
47   int attempts = 0;
48   while (WiFi.status() != WL_CONNECTED && attempts < 10) { //
      Essayer 10 fois
49     delay(1000);
50     Serial.print("."); // Afficher un point pour chaque seconde
      attempts++;
51   }
52   return WiFi.status() == WL_CONNECTED;
53 }
54
55
56 // Fonction pour g rer le bouton dans l'application Blynk
57 BLYNK_WRITE(V1) {
58   int pinValue = param.asInt(); // Lire la valeur du bouton (0 ou
      1)
59
60   Serial.print("Bouton de l'application Blynk : ");
61   Serial.println(pinValue);
62
63   if (pinValue == 1 || pinValue == 255) { // Si le bouton est
      appuy
64     digitalWrite(ledPin, HIGH); // Allumer la LED 1
65     digitalWrite(ledPin1, HIGH); // Allumer la LED 2
66     Serial.println("LED allum e");
67   } else { // Si le bouton est rel ch
68     digitalWrite(ledPin, LOW); // teindre la LED 1
69     digitalWrite(ledPin1, LOW); // teindre la LED 2
70     Serial.println("LED teinte ");
71   }

```

```
72 }  
73  
74 void loop() {  
75     Blynk.run(); // Maintenir la connexion avec Blynk  
76 }
```

Listing 1: Contrôle de LED avec ESP32 et Blynk

3 Explications

- **BLYNK_AUTH_TOKEN** : C'est le jeton d'authentification que Blynk génère pour votre projet. Il permet de connecter l'ESP32 à l'application.
- **WiFi.begin()** : Cette fonction est utilisée pour se connecter à un réseau Wi-Fi. Si la connexion échoue, le code tentera de se connecter au deuxième réseau.
- **BLYNK_WRITE(V1)** : Cette fonction est déclenchée lorsque vous appuyez sur le bouton de l'application Blynk. Elle lit la valeur du bouton et allume ou éteint les LED en conséquence.
- **Blynk.run()** : Permet de maintenir la connexion active avec l'application Blynk.