

ESP8266Sketch3_ConnectionWiFi

Ce programme utilise un module **ESP8266** pour se connecter à un réseau **WiFi** en mode station (client), obtenir une adresse IP et afficher des informations sur la connexion réseau dans le moniteur série. Il est utile comme point de départ pour toute application nécessitant une connexion WiFi, comme l'accès à des serveurs web ou l'envoi de données à distance.

```
#include <ESP8266WiFi.h>

const char *SSID = "XXXXXXX";
const char *PASSWORD = "123456";
void setup() {
  Serial.begin(115200);
  delay(100);
  WiFi.mode(WIFI_STA);
  WiFi.begin(SSID, PASSWORD);
  Serial.println("\n");
  Serial.print("Connexion en cours ");

  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
  }

  Serial.println("\n");

  if (WiFi.isConnected()) {
    Serial.println("ESP8266 connecté au réseau WIFI");
    Serial.println("Adresse IP : " + WiFi.localIP().toString());
    Serial.println("Passerelle IP : " + WiFi.gatewayIP().toString());
    Serial.println("DNS IP : " + WiFi.dnsIP().toString());
    Serial.print("Puissance de réception : ");
    Serial.println(WiFi.RSSI());
  }
  else {
    Serial.println("ESP8266 non connecté au réseau WIFI");
  }
  delay(2000);
}

void loop() {
}
```

```
#include <ESP8266WiFi.h>
```

- Bibliothèque spécifique à l'ESP8266 pour gérer les connexions WiFi. Elle permet d'utiliser des fonctions comme `WiFi.begin()` pour établir une connexion et d'autres fonctions pour obtenir des informations réseau.

```
const char *SSID = "XXXXXXXX";  
const char *PASSWORD = "123456";
```

- **SSID** : Constante de type pointeur char qui contient le nom du réseau WiFi auquel l'ESP8266 doit se connecter.
- **PASSWORD** : Constante contenant le mot de passe du réseau WiFi. Il doit être correct pour que le module puisse se connecter au réseau.

```
WiFi.mode(WIFI_STA);  
WiFi.begin(SSID, PASSWORD);
```

- **WiFi.mode(WIFI_STA)** : Cette ligne configure le module ESP8266 en mode **station** (client). Cela signifie qu'il se connectera à un réseau WiFi existant, comme un ordinateur ou un smartphone qui se connecte à un routeur.
- **WiFi.begin(SSID, PASSWORD)** : Cette fonction démarre le processus de connexion au réseau WiFi en utilisant le **SSID** et le **mot de passe** définis précédemment. Elle permet de se connecter au réseau spécifié.

```
while (WiFi.status() != WL_CONNECTED) {  
    delay(1000);  
    Serial.print(".");  
}
```

- **while (WiFi.status() != WL_CONNECTED)** : Cette boucle while attend que l'ESP8266 soit connecté au réseau WiFi. La fonction **WiFi.status()** renvoie l'état de la connexion WiFi. Si cet état n'est pas **WL_CONNECTED** (connecté), la boucle continue à attendre.
- **delay(1000)** : Attendre 1 seconde avant de vérifier à nouveau l'état de la connexion.
- **Serial.print(".")** : Affiche un point dans le moniteur série pour indiquer que le processus de connexion est toujours en cours.

```
if (WiFi.isConnected()) {
  Serial.println("ESP8266 connecté au réseau WIFI");
  Serial.println("Adresse IP : " + WiFi.localIP().toString());
  Serial.println("Passerelle IP : " + WiFi.gatewayIP().toString());
  Serial.println("DNS IP : " + WiFi.dnsIP().toString());
  Serial.print("Puissance de réception : ");
  Serial.println(WiFi.RSSI());
}
else {
  Serial.println("ESP8266 non connecté au réseau WIFI");
}
delay(2000);
```

- **WiFi.isConnected()** : Vérifie si l'ESP8266 est bien connecté au réseau WiFi. Si la connexion est réussie, les informations suivantes sont affichées :
 - **WiFi.localIP()** : Récupère l'adresse IP attribuée au module ESP8266 par le routeur.
 - **WiFi.gatewayIP()** : Affiche l'adresse IP de la passerelle, c'est-à-dire le routeur auquel l'ESP8266 est connecté.
 - **WiFi.dnsIP()** : Affiche l'adresse IP du serveur DNS, qui est utilisé pour la résolution des noms de domaine.
 - **WiFi.RSSI()** : Cette fonction renvoie la puissance du signal WiFi en **dBm** (décibels milliwatt), appelée **RSSI** (Received Signal Strength Indicator). Un RSSI négatif élevé (par exemple, -30 dBm) signifie un signal fort, tandis qu'un RSSI plus proche de -100 dBm indique un signal faible.