



---

# CONNEXION DE L'ESP32 A EDUROAM

---

Réalisé par Bruno PIQUEMAL

24 septembre 2020

**Ce tutoriel a pour but de vous présenter une méthode pour vous  
connecter au réseau Wifi Eduroam via votre ESP32.**



---

# Introduction

Certains projets peuvent amener les étudiants à vouloir connecter leur microcontrôleur à la WiFi. Pour ce faire, il peut être judicieux d'opter pour un ESP32 qui possède un module WiFi intégré. Dans ce tutoriel, nous verrons ainsi comment connecter un ESP32 au réseau Eduroam.

Dans un premier temps, nous nous assurerons d'avoir tous les éléments mis en place afin de programmer la carte ESP32 avec l'IDE d'Arduino.



---

## Table des matières

<b>A</b>	<b>Inclure toutes les cartes ESP32 dans l'IDE</b>	<b>3</b>
<b>B</b>	<b>Problèmes éventuels</b>	<b>6</b>
<b>C</b>	<b>La connexion au Wifi</b>	<b>7</b>

\* \* \*



---

## A Inclure toutes les cartes ESP32 dans l'IDE

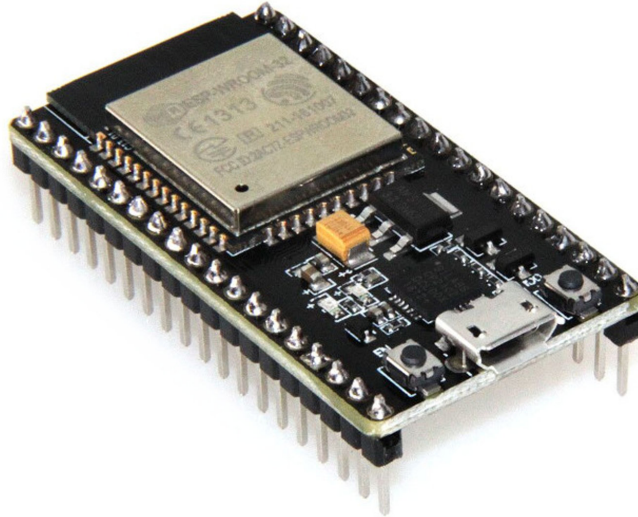


FIGURE 1 – ESP32 Node MCU WROOM-32

Afin de pouvoir programmer l'ESP32, il faut ajouter une source supplémentaire dans le "Gestionnaire de cartes supplémentaires" de l'IDE. Il suffit de copier/coller un lien qui pointe sur un dossier JSON qui facilite le dialogue entre CPU.

Les instructions à suivre sont les suivantes :

- Cliquez sur le menu Fichier dans la barre d'outils
- Cliquez dans "Préférences", une boîte de dialogue devrait s'ouvrir
- Si vous n'êtes pas sur l'onglet "Paramètres", veuillez vous y rendre
- Cherchez l'emplacement où est marqué "URL de gestionnaire de cartes supplémentaires"
- Si jamais il y a déjà un lien, mettez une virgule puis ajoutez celui-ci

[https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\\_esp32\\_index.json](https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json)

- Cliquez sur "OK" pour valider la saisie.

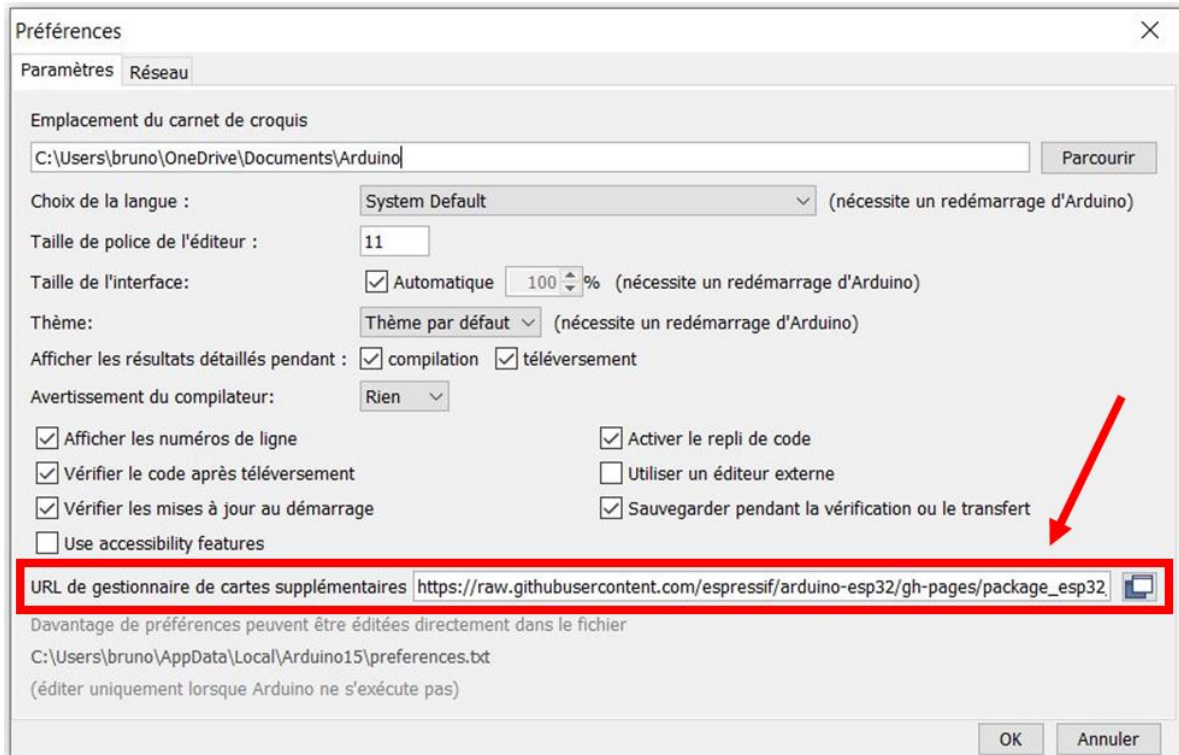


FIGURE 2 – Fenêtre explicative 1

Ensuite, il va falloir ajouter les cartes ESP032 à l'IDE de l'Arduino. Les instructions à suivre sont les suivantes :

- Dans l'IDE d'Arduino, cliquez sur "Outils" dans la barre d'outils
- Descendez jusqu'à "Type de carte" et laissez que le menu se déroule
- Cliquez sur "Gestionnaire de cartes"
- Dans la barre de recherches, mettez "ESP32"
- Installez, comme sur l'exemple, la librairie d'Expressif Systems

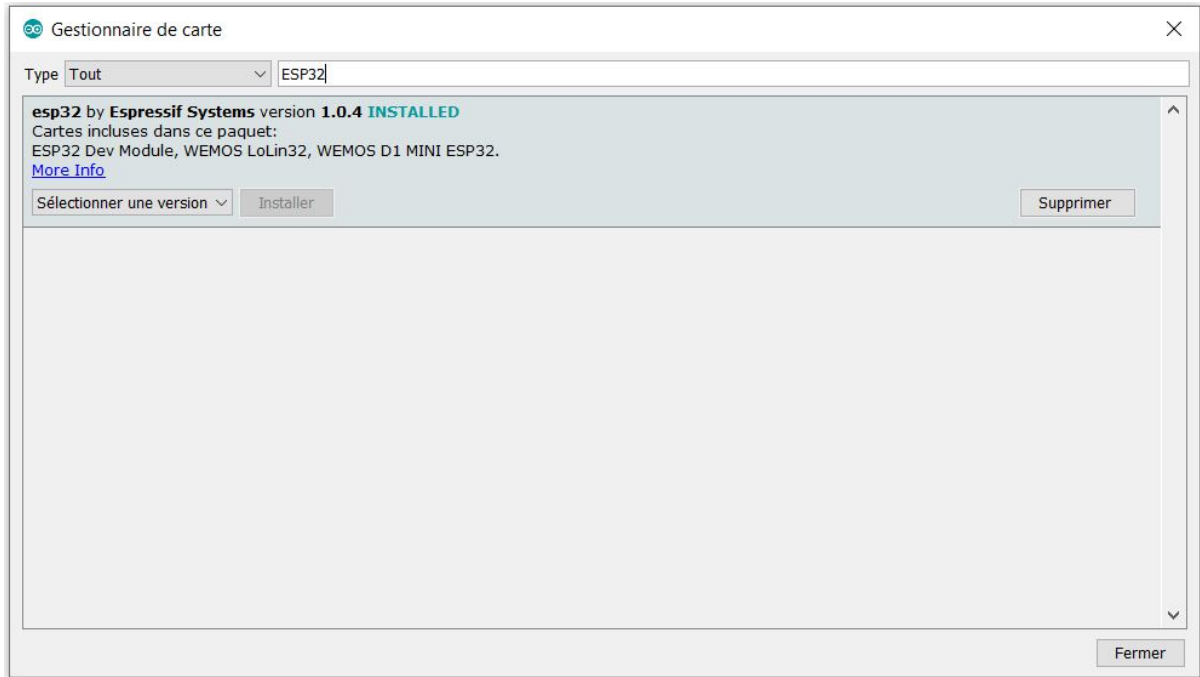


FIGURE 3 – Fenêtre explicative 2

Si vous regardez dans l'onglet "Outils" puis "Type de carte", vous devriez voir pleins de cartes ESP32. **Il est impératif** de sélectionner la carte correspondante ou équivalente à la votre.

Dans mon cas j'utilise la carte ESP32S Node MCU donc je sélectionne la carte "Node32s" parmi celles proposées.

A ce stade, il est possible que la configuration soit presque totalement faite. "Presque" parce qu'il se peut que votre ordinateur rencontre encore quelques soucis.



---

## **B Problèmes éventuels**

Il se peut que des problèmes subsistent malgré le processus d'installation que vous venez de réaliser.

→ Si l'ordinateur ne détecte pas la carte lorsque celle-ci est connectée en USB, il y a de fortes chances que le câble soit défectueux. Ayez en tête que la qualité des câbles influe directement dans la vitesse de transfert des données entre l'ESP32 et l'ordinateur. De plus, certains câbles ne sont vendus que pour charger les téléphones portables (ou autres appareils) et donc qu'il n'y a pas de fils permettant de faire du transfert d'informations.

→ **"No module named serial"**

Si ce message apparaît lors du transfert de votre programme, c'est que vous devez télécharger le programme "pyserial". Veuillez suivre les instructions dans ce lien :

<https://learn.adafruit.com/arduino-lesson-17-email-sending-movement-de-installing-python-and-pyserial>

→ Lorsque vous téléversez votre programme depuis l'IDE d'Arduino, il ne faut pas oublier d'appuyer sur le bouton **"IO0"** qui se trouve sur l'ESP32 Node MCU et qui permet de faire transiter les données. Le bouton **"EN"** permet de faire redémarrer la carte ESP32.



## C La connexion au Wifi



### Codage

```
1 #include <Wire.h>
2 #include <WiFi.h> //Wifi library
3 #include "esp_wpa2.h" //wpa2 library for connections to Enterprise networks
4 #define EAP_IDENTITY "login_utc@utc.frr" //if connecting from another
   corporation, use identity@organisation.domain in Eduroam
5 #define EAP_PASSWORD "mon_password" //your Eduroam password (cest la mot
   de passe pour se connecter IUTC)
6 const char* ssid = "eduroam"; //Eduroam SSID
7 const char* host = "www.utc.fr"; //external server domain for HTTP
   connection after authentication
8 int counter = 0;
```

Il faut inclure les différentes bibliothèques qui nous seront utiles pour ce projet. On définit ensuite le nom et le mot de passe qui correspondent au nom du réseau auquel on veut se connecter et à son éventuel mot de passe.

On arrive ensuite à la boucle setup. Dans cette boucle on va retrouver la vitesse de communication, l'initialisation de la librairie Wifi.h et la configuration de notre connexion en mode client (Station mode → STA à la ligne 7).

Toutes les instructions commençant par "esp\_wifi\_sta\_wpa2\_ent\_" sont nécessaires pour se connecter à un réseau entreprise (Enterprise connexion).

Ensuite, on essaie de se connecter au réseau pendant 30 secondes, si on y arrive on écrit "Wifi connected", sinon on redémarre l'ESP32. N'oubliez pas de mettre le bon "baud rate" (vitesse de transfert de données) qui est de 115200 sur le moniteur série.





## Codage

```
1 void setup() {
2   Serial.begin(115200);
3   delay(10);
4   Serial.print("Connecting to network: ");
5   Serial.println(ssid);
6   WiFi.disconnect(true); //disconnect form wifi to set new wifi connection
7   WiFi.mode(WIFI_STA); //init wifi mode
8   esp_wifi_sta_wpa2_ent_set_identity((uint8_t *)EAP_IDENTITY,
   strlen(EAP_IDENTITY)); //provide identity
9   esp_wifi_sta_wpa2_ent_set_username((uint8_t *)EAP_IDENTITY,
   strlen(EAP_IDENTITY)); //provide username is for identity and username is
   same
10  esp_wifi_sta_wpa2_ent_set_password((uint8_t*)EAP_PASSWORD,
   strlen(EAP_PASSWORD)); //provide password
11  esp_wpa2_config_t config = WPA2_CONFIG_INIT_DEFAULT(); //set
   config settings to default
12  esp_wifi_sta_wpa2_ent_enable(&config); //set config settings to enable
   function
13  WiFi.begin(ssid); //connect to wifi
14  while (WiFi.status() != WL_CONNECTED) {
15    delay(500);
16    Serial.print(".");
17    counter++;
18    if (counter >= 60) { //after 30 seconds timeout—reset board
19      ESP.restart();
20    }
21  }
22  Serial.println("");
23  Serial.println("WiFi connected");
24  Serial.println("IP address set: ");
25  Serial.println(WiFi.localIP()); //print LAN IP
26 }
```

En ce qui concerne la boucle loop(), si l'on se connecte au réseau, on affiche des données le concernant sur le moniteur série, autrement on écrit "Connection unsuccessful".



## Codage

```
1 void loop() {
2   if (WiFi.status() == WL_CONNECTED) { //if we are connected to Eduroam
3     network
4     counter = 0; //reset counter
5     Serial.println("Wifi is still connected with IP: ");
6     Serial.println(WiFi.localIP()); //inform user about his IP address
7   } else if (WiFi.status() != WL_CONNECTED) { //if we lost connection,
8     retry
9     WiFi.begin(ssid);
10  }
11  while (WiFi.status() != WL_CONNECTED) { //during lost connection, print
12    dots
13    delay(500);
14    Serial.print(".");
15    counter++;
16    if (counter >= 60) { //30 seconds timeout – reset board
17      ESP.restart();
18    }
19  }
20  Serial.print("Connecting to website: ");
21  Serial.println(host);
22  WiFiClient client;
23  if (client.connect(host, 80)) {
24    String url = "/rele/rele1.txt";
25    client.print(String("GET ") + url + " HTTP/1.1\r\n" + "Host: "
26      + host + "\r\n" + "User-Agent: ESP32\r\n" + "
27      Connection: close\r\n\r\n");
28    while (client.connected()) {
29      String line = client.readStringUntil('\n');
30      if (line == "\r") {
31        break;
32      }
33    }
34    String line = client.readStringUntil('\n');
35    Serial.println(line);
36  } else {
37    Serial.println("Connection unsuccessful");
38  }
39  Serial.println();
40  delay(2000);
41 }
```