



# Microcontrôleurs - Les modules ES01(S)



[Mise à jour le 19/8/2023]

## Ressources

- Site [Espressif](#)
- [ESP-AT User Guide](#)
- JOY-IT : [Manuel d'utilisation de l'ESP8266](#)
- [Mise en marche du programmeur USB pour ESP8266](#)
- [Arduino IDE 2](#)

## Articles

- [Découverte des ESP8266 : le microcontrôleur connecté par Wifi pour 2€ au potentiel phénoménal](#)
- [Premiers pas avec l'ESP8266 \(ESP-01\)](#)
- [ESP8266 - Présentation du module ESP-01](#)

## Lectures connexes

- Wiki " [Matériels - Capteurs, afficheurs, préactionneurs, etc.](#) "
- Bibliothèques - [Arduino Library List](#)

## Distributeurs

- [GO TRONIC](#)

---

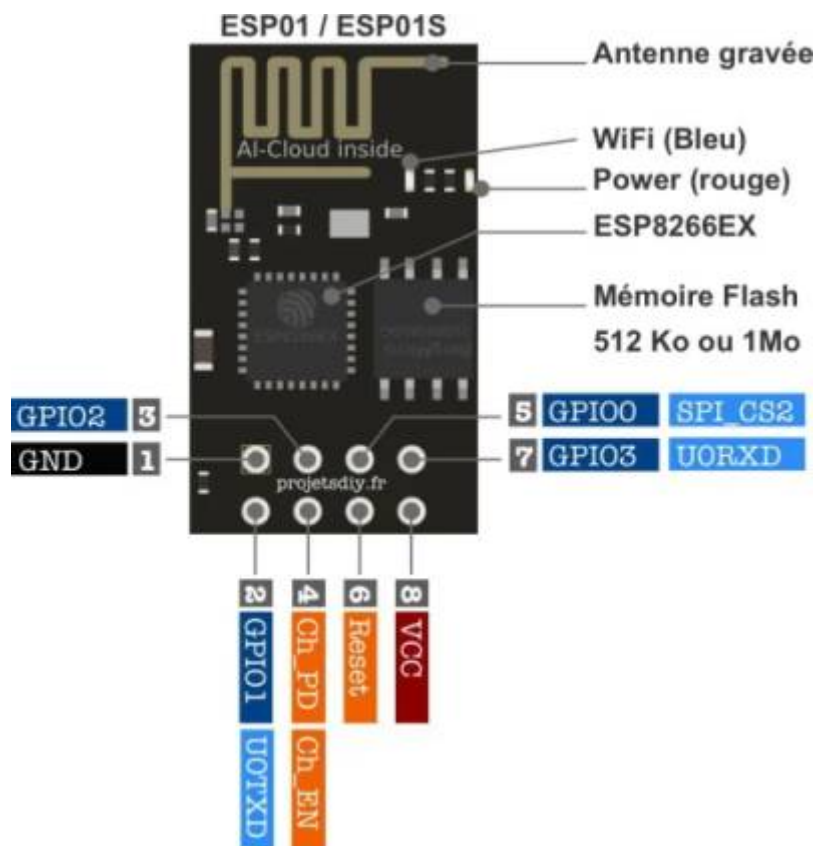
## 1. Généralités

L'**ESP-01** est une **version réduite de l'ESP8266** (ESPRESSIF) produite par AI-Thinker. De taille réduite, il permet de connecter un microcontrôleur à un réseau Wi-Fi et d'établir des connexions TCP/IP avec des commandes [Hayes](#).

## 2. Principales caractéristiques

- **Ressources** : [ESP-01 802.11 b/g/n Wi-Fi Module](#)
- **Modèle** : ESP01(S) ([ESP8266EX](#))
- **Processeur** : 32-bit CPU @ 160MHz
- **Wifi** 802.11 b/g/n (WEP/WPA-PSK/WPA2-PSK)
- **SPI Flash** : 512kB ou 1MB(S)
- **UART** : 300 à 4608000 Défaut : 115200
- **IO** : 2 (logique 3,3V)

## 3. Brochage



## 4. Accessoires



### 4.1 Adaptateur USB <--> Série et programmeur

Ce module permet la programmation du module WiFi ESP8266 via le port USB d'un PC.

- **Ressources**

- Distributeur : [GO TRONIC](#)
- Documentation : [Installation du driver CH340/CH341](#)



#### 4.2 Câble adaptateur USB <--> Série TTL-232R-3V3

Les câbles TTL-232R-3V3 sont une famille de câbles convertisseurs USB vers UART série TTL intégrant le CI FT232RQ à interface USB vers UART série qui gère toute la signalisation et les protocoles USB. Les câbles offrent un moyen simple et rapide pour connecter des périphériques avec une interface série en niveau TTL vers USB.

- **Ressources**

- Distributeur : [Farnell](#)
- Documentation : [TTL-232R-3V3 Datasheet](#)



#### 4.3 MB102 Module 3.3V / 5V

- **Ressources**

- Distributeur : [AZ-Delivery](#)
- Documentation : [e-book](#)

- **Caractéristiques**

- Tension d'entrée: 6,5 - 12V DC ou via USB.
- Tension de sortie: 3,3V / 5V sélectionnable.
- Alimentation par port USB ou bloc d'alimentation externe

### 5. Commandes AT - Firmware d'origine

- **Ressources**

- [ESP-01. Comment réinstaller le firmware d'origine NodeMCU \(ESP8266\)](#)
- [ESP-AT User Guide](#)

### • Tests dans la console du logiciel Arduino

1. **Télécharger** le répertoire contenant les fichiers *esp8266\_flasher.exe* et *DOIT\_AT\_DOUT\_8Mbit.bin*
2. Placer l'ESP01 sur le programmeur USB ↔ UART et le commuter en mode **prog**.
3. Transférer le fichier .bin sur l'ESP01 avec *esp8266\_flasher.exe*. Débrancher le programmeur et le placer en mode **UART**.
4. Tester le programme en transmettant les **commandes AT** suivantes avec le moniteur

série du logiciel Arduino réglé sur :

Les deux, NL et CR ▾

115200 baud ▾

```
// 1a. Connexion au module
```

```
AT
```

```
réponse : OK
```

```
// 1b. Reset
```

```
AT+RST
```

```
Exemple de réponse partielle: OK
```

```
ets Jan 8 2013,rst cause:2, boot mode:(3,7)
```

```
load 0x40100000, len 2592, room 16
```

```
tail 0
```

```
checksum 0xf3
```

```
// 2. Version du firmware
```

```
AT+GMR
```

```
Exemple de réponse:
```

```
AT version:1.2.0.0(Jul 1 2016 20:04:45)
```

```
SDK version:1.5.4.1(39cb9a32)
```

```
www.doit.am
```

```
0.0.0.6
```

```
Build:1.5.4.1 Apr 17 2018 11:39:14
```

```
OK
```

```
// 3. Adresse IP du module (par défaut en mode point d'accès)
```

```
AT+CIFSR
```

```
réponse :
```

```
+CIFSR:APIP,"192.168.4.1"
```

```
+CIFSR:APMAC,"c6:5b:be:61:61:3d"
```

```
// 4. Réglage en mode STA(STATION) (2:AP (Point d'accès), 3:STA+AP)
```

```
AT+CWMODE=1
```

```
réponse : OK
```

```
// 5. Connexion à un réseau WIFI
```

```
AT+CWLAP="ssid","mot2passe"
```

```
Exemple de réponse:
```

```
connect to ssid SynBoxLAN, channel 5
```

```
OK
```

```
// 6. IP attribué par le routeur
```

```
AT+CIFSR
```

```
Exemple de réponse:
```

```
+CIFSR:STAIP,"192.168.200.189"  
+CIFSR:STAMAC,"c4:5b:be:61:61:3d"  
OK
```

## 6. Téléverser un programme avec Arduino ou PlatformIO

- **Ressources**

- Préparation de l'IDE Arduino : voir le wiki "[La carte ESP8266 Feather Huzzah](#)"

### 6.1 Premier programme

- **Ressources**

- [Premiers pas avec l'ESP8266 \(ESP-01\)](#)

- **Dans le logiciel Arduino**

1. Placer l'ESP01 sur l'adaptateur USB ↔ UART et le commuter en mode **prog.**
2. Sélectionner l'onglet Outils → Type de carte → Generic ESP8266 Module. Sélectionner le port COM.
3. Sélectionner l'onglet Fichier → exemple → ESP8266 → Blink et transférer.
4. Débrancher l'adaptateur et le placer en mode **UART**. Rebrancher.
5. La LED bleue clignote !



### 6.2 Wifi Web Serveur

- **Ressource**

- Distributeur : [AZ-Delivery](#)
- Documentation : [ESP8266-01S mit Relais](#)

- **Dans le logiciel Arduino**

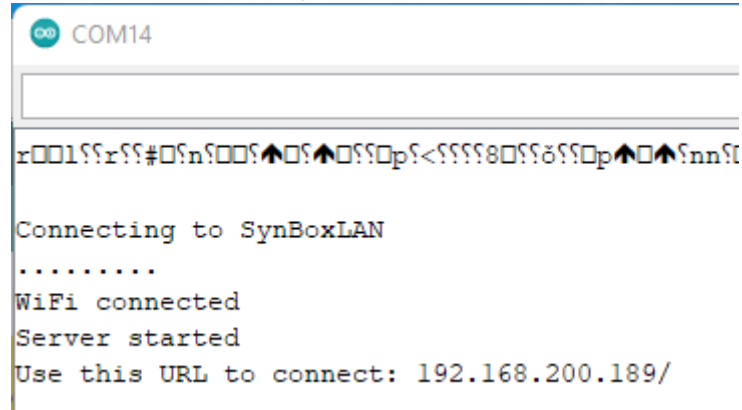
1. Placer l'ESP01 sur l'adaptateur USB ↔ UART et le commuter en mode **prog.**
2. Sélectionner l'onglet Outils → Type de carte → Generic ESP8266 Module. Sélectionner le port COM.
3. [Télécharger](#) le fichier **ESP01WebServer** et régler le SSID et le mot2passe du réseau puis transférer dans l'ESP01.

```
const char* ssid = "your SSID";  
const char* password = "your password";
```

4. Tests

## 1. Sur la **carte à relais**

1. Placer l'ESP01 sur la carte à relais.
2. Connecter un câble adaptateur USB ↔ Série à la carte **TTL-232R-3V3 : GND ↔ GND, TX ↔ RX, RX ↔ TX**
3. Ouvrir la console Arduino
4. Alimenter la carte sous 5V. Une fois la connexion au wifi effectuée l'ESP01 donne son IP comme dans l'exemple ci-dessous.



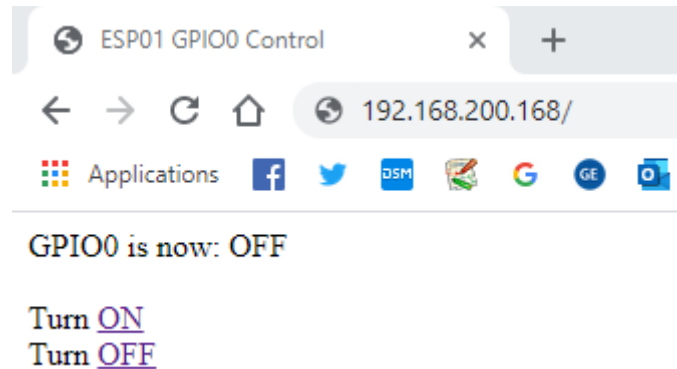
```

COM14

r001?r??#?n?□?▲?▲?□?p?<????8□?ö??p▲?▲?nn?

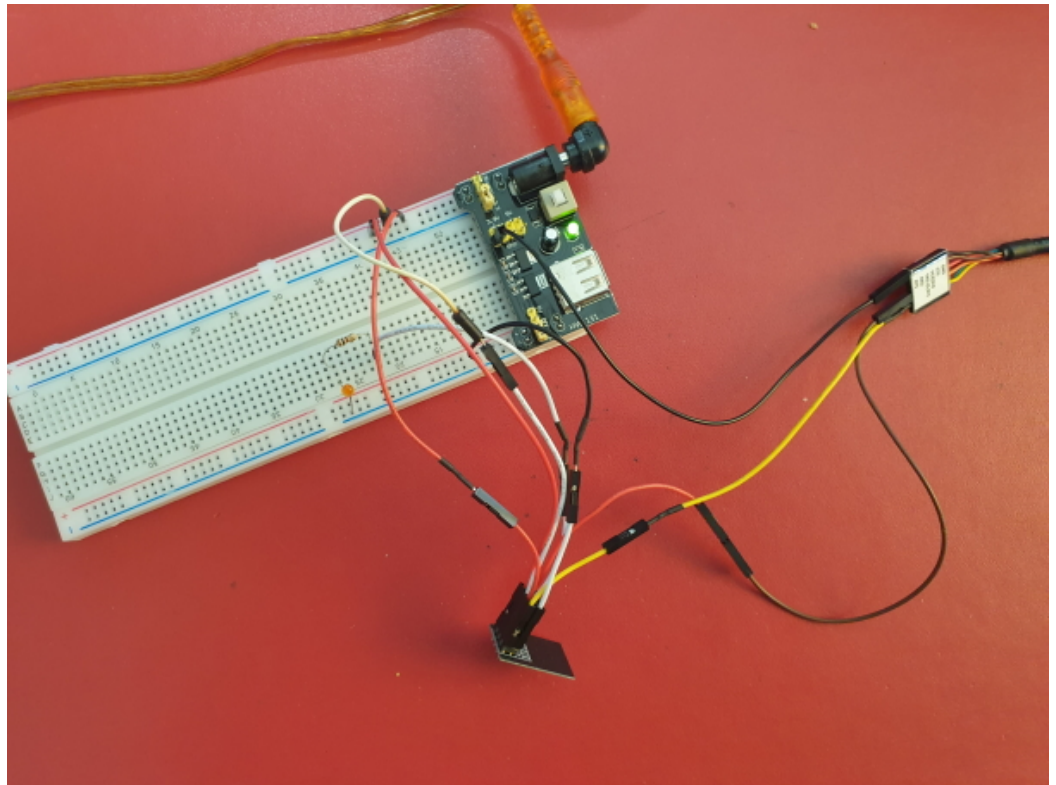
Connecting to SynBoxLAN
.....
WiFi connected
Server started
Use this URL to connect: 192.168.200.189/
  
```

5. Ouvrir un navigateur et entrer l'IP de l'ESP01 suivi de /. Changer l'état du relais en cliquant sur ON et OFF.



## 2. Sur une **breadboard**

1. Alimenter l'ESP01 avec un module MB102 : **GND ↔ GND, 3V3 ↔ 3V3, RST ↔ 3V3, EN ↔ 3V3**
2. Connecter **TX ↔ RX, RX ↔ TX** à un câble adaptateur USB ↔ Série tel que **TTL-232R-3V3** : (relier le GND de l'adaptateur au GND du MB102)
3. Relier la sortie IO0 à l'élément à commander (Led, etc.). Une Led et sa résistance sur le photo ci-dessous.
4. Vérifier le fonctionnement en reprenant la démarche ci-dessus à partir du a. III.



## Pour aller plus loin

- **ESP8266 SDK** - [Getting Started Guide](#)

From:

<http://webge.fr/dokuwiki/> - **WEBGE Wikis**

Permanent link:

<http://webge.fr/dokuwiki/doku.php?id=microc:uc:esp01>

Last update: **2023/08/19 11:05**

