Par défaut, le microcontrôleur ESP8266 contient un firmware de gestion des commandes AT. Afin de programmer le microcontrôleur avec l'IDE Arduino, il est nécessaire de changer le firmware.

Étape n°1 : configuration

La première étape est la configuration de l'IDE Arduino pour la prise en compte de la carte Esp8266.

- Ouvrir l'IDE Arduino.
- Ensuite :
 - Fichier->Préférences



• Dans la boite de dialogue des préférences, au champ intitulé « *URL de gestionnaire de cartes supplémentaires* « , vous pouvez coller l'adresse suivante :

http://arduino.esp8266.com/stable/package_esp8266com_index.json

Préférences			×		
Paramètres Réseau					
Emplacement du carnet de croquis					
C:\Users\nico\Documents\Arduino			Parcourir		
Choix de la langue :	Langue du système	(nécessite un redémarrage d'Arduino)			
Taille de police de l'éditeur :	12				
Interface scale:	Automatic 100 芸 % (nécessite un redémarrage d'A	rduino)			
Afficher les résultats détaillés pendant : 🦵 compilation 🖵 téléversement					
Avertissement du compilateur:	Rien				
🔲 Afficher les numéros de ligne					
🗖 Activer le repli de code					
Vérifier le code après téléversement					
🔲 Utiliser un éditeur externe					
🔽 Vérifier les mises à jour au démarrag	je				
🔽 Mettre à jour l'extension des fichier	s croquis lors de la sauvegarde (.pde -> .ino)				
Sauvegarder pendant la vérification	ou le transfert				
URL de gestionnaire de cartes suppléme	ntaires http://arduino.esp8266.com/stable/package_esp826	6com_index.json			
Davantage de preferences peavone Sare	Call Constant and a California				
C:\Users\nico\AppData\Local\Arduino15	\preferences.txt				
(éditer uniquement lorsque Arduino ne s	'exécute pas)				
		1			
		OK	Annuler		

• Validez cette modification avec le bouton OK.

Étape n°2 : installation

• Puis, dans le menu : Outils->Type de carte->Gestionnaire de cartes :



• Ensuite, dans le champs de recherche, recherchez « *esp8266* » :

🥯 Ge	estionnaire de carte	×
Туре	e Tout esp8266	
es Ca Ge Ad ES <u>Or</u>	i p8266 by ESP8266 Community artes incluses dans se paquet: eneric ESP8266 Module, Olimex MOD-WIFI-ESP8266(-DEV), NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), Jafruit HUZZAH ESP8266 (ESP-12), ESPresso Lite 1.0, ESPresso Lite 2.0, Phoenix 1.0, Phoenix 2.0, SparkFun Thing, SweetPea iP-210, WeMos D1, WeMos D1 mini, ESPino (ESP-12 Module), ESPino (WROOM-02 Module), WifInfo, ESPDuino. nline help ore info	-
	2.3.0 Installer	
		-
,	Ferm	ier

• Installez « esp8266 by ESP8266 Community » :



Installation en cours

Étape n°3 : test

Après quelque minutes d'installation, vous pouvez sélectionner la nouvelle carte dans :

• Outils->Type de carte

🧙 sketch_jul25c Ardu	ino 1.6.8		
Fichier Édition Croquis	Outils Aide		
sketch jul25c	Formatage automatique Archiver le croquis Réparer encodage & recharger	Ctrl+T	Arduino/Genuino Micro Arduino Esplora
<pre>void setup() { // put your setur</pre>	Moniteur série Traceur série	Ctrl+Maj+M Ctrl+Maj+L	Arduino Mini Arduino Ethernet Arduino Eio
<pre>} void loop() {</pre>	Type de carte: "Arduino Nano" Processeur: "ATmega328" Port) 	Arduino BT LilyPad Arduino USB LilyPad Arduino
// put your main	Programmateur: "AVRISP mkII" Graver la séquence d'initialisation	•	Arduino Pro or Pro Mini Arduino NG or older
,			Arduino Robot Control Arduino Robot Motor Arduino Gemma
			ESP8266 Modules Generic ESP8266 Module Generic ESP8285 Module
			ESPDuino (ESP-13 Module) Adafruit HUZZAH ESP8266 ESPresso Lite 1.0
			Phoenix 1.0 Phoenix 2.0 NodeMCU 0.9 (ESP-12 Module)
			NodeMCU 1.0 (ESP-12E Module) Olimex MOD-WIFI-ESP8266(-DEV) SparkFun ESP8266 Thing

Maintenant, vous pouvez tester un exemple conçu pour le microcontrôleur ESP8266.



Un petit exemple pour faire clignoter une led :

```
💿 Blink | Arduino 1.6.8
                                                                                                    Fichier Édition Croquis Outils Aide
        Ð
                                                                                                      Ø
 v )
  Blink
 ESP8266 Blink by Simon Peter
 Blink the blue LED on the ESP-Ol module
 This example code is in the public domain
 The blue LED on the ESP-01 module is connected to GPI01
 (which is also the TXD pin; so we cannot use Serial.print() at the same time)
 Note that this sketch uses LED_BUILTIN to find the pin with the internal LED
*7
void setup() {
 pinMode(LED_BUILTIN, OUTPUT); // Initialize the LED_BUILTIN pin as an output
}
// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, LOW); // Turn the LED on (Note that LOW is the voltage level
                                   // but actually the LED is on; this is because
                                   // it is acive low on the ESP-01)
                                   // Wait for a second
  delay(1000);
  digitalWrite(LED_BUILTIN, HIGH); // Turn the LED off by making the voltage HIGH
  delay(2000);
                                   // Wait for two seconds (to demonstrate the active low LED)
}
```

 Vérifier & télécharger le programme après avoir sélectionner le bon port.

Téléversement terminé	
Le croquis utilise 222 201 octets (21%) de l'espace de stockage de programmes. L Les variables globales utilisent 31 576 octets (38%) de mémoire dynamique, ce qu	e maximum est de 1 044 464 oc i laisse 50 344 octets pour 1
Uploading 226352 bytes from C:\Users\nico\AppData\Local\Temp\build8cf7c6dal487b3	561b274c204972ed8f.tmp/Blink.
	[36%] [72%]
	[100%]

Voilà, fin du tuto !

Sources : www.esp8266.com