

LOGICIEL MU

Mu est un éditeur de code Python destiné aux programmeurs débutants.

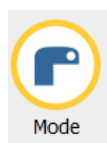
Logiciel Mu à télécharger sur : <https://codewith.mu>

Attention, seule la version Alpha permet de programmer les cartes Micro:bit et ESP8266 / ESP32

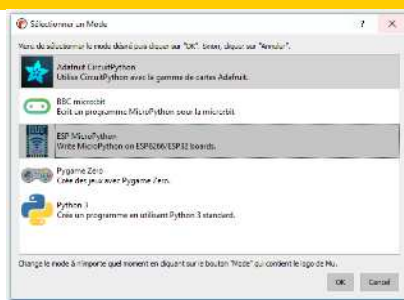


Les menus de Mu

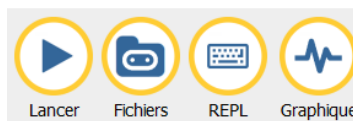
1. Choix du mode



Choisir la carte utilisée



2. Opérations de communication avec les ESP8266 ESP32



- **FLASHER** : pour programmer dans le micro-contrôleur le programme MicroPython présent dans la fenêtre d'édition.

- **FICHIERS** : permet d'accéder au système de fichier du micro-contrôleur pour :

- stocker un programme, une bibliothèque
- enregistrer des couples date/valeur

- **REPL** : (Read-Evaluate- Print Loop) : interpréteur interactif, qui permet d'envoyer vers la carte des instructions.

Raccourci clavier :


- CTRL-C : Interrompre un programme en exécution ;
- CTRL-D : Sur une ligne vide, réinitialisation logicielle de la carte.

Commandes :

- `dir()` : Indique les variables, les modules et les fonctions utilisées. `dir(nom)` : indique les attributs sur la variable, le module ou la fonction nommée.
- `help()` : Donne des informations sur Mu et la carte. `help(nom)` : Donne des informations sur la variable, le module ou la fonction nommée.

- **GRAPHIQUE** : permet de visualiser graphiquement des valeurs en temps réel.

L'outil graphique

L'outil graphique  permet de visualiser graphiquement des valeurs en temps réel.

Il faut utiliser la commande **print(tuple)**.

Un **tuple** est une sorte de liste délimitée par des parenthèses.

A chaque élément du tuple correspondra une courbe.

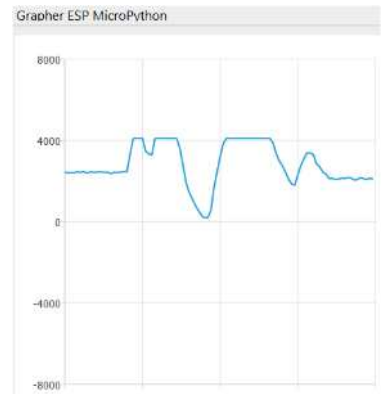
- 1 courbe : **print((temp,))**
- 2 courbes : **print((temp1, temp2))**
- 3 courbes : **print((temp1, temp2, temp3))**

Exemple de programme pour afficher la valeur du potentiomètre connecté sur la broche 34 (A3)

```
from machine import ADC, Pin
from time import sleep

can = ADC(Pin(34))          # crée un objet ADC sur la broche 34 (A3)

while True:
    pot = can.read()
    print((pot,))           # affichage sur l'outil graphique
    sleep(0.1)
```



Installer et utiliser une bibliothèque MicroPython avec Mu

Plusieurs possibilités pour trouver des bibliothèques : Le fabricant du matériel, moteur de recherche, GitHub.

Exemple pour l'afficheur Grove LCD RGB :

1. **Télécharger** la bibliothèque qui vous intéresse (exemple : de l'afficheur ici fournie dans la zone mise en œuvre [Afficheur LCD RGB](#))

2. **Dézipper** les fichiers.

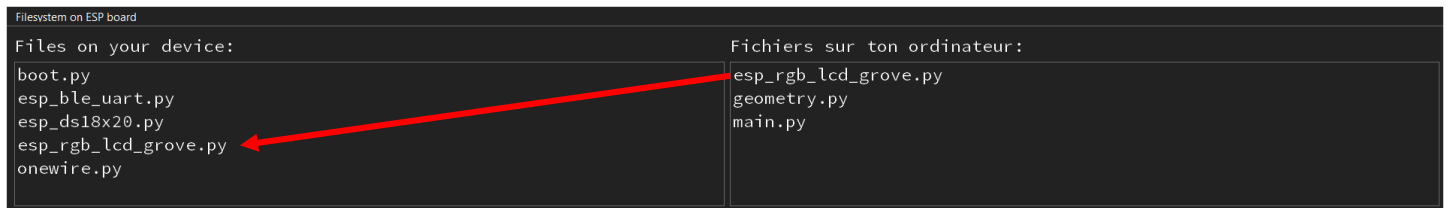
3. **Coller** les fichiers dans le répertoire d'enregistrement de votre code Mu python (probablement dans votre espace personnel\mu_code) :



4. Dans **Mu**, cliquer sur **Fichiers** (Attention, il ne faut pas que **REPL** soit sélectionné)

Un panneau s'ouvre montrant deux zones : à gauche les fichiers présents sur la carte, à droite ceux dans le répertoire **mu_code**

5. **Copier** le fichier bibliothèque souhaité (exemple : **esp_rgb_lcd_grove.py**) en le déplaçant à l'aide de la souris de la zone « Fichiers sur ton ordinateur » vers la zone carte « Fichier sur ta "carte" ».



6. **Tester** la présence de la bibliothèque (par exemple en utilisant le fichier **hello_world.py** qui utilise la bibliothèque **esp_rgb_lcd_grove.py**)

Transférer dans la carte un code qui démarre à la mise sous tension de la carte

1. **Enregistrer** le code python que vous venez de tester avec le REPL sous le nom de **main.py** dans le répertoire **mu_code**

2. **Transférer** le fichier **main.py** dans la carte (voir la section sur l'installation d'une bibliothèque).

3. **Rebooter** la carte à l'aide de son bouton RESET ou couper puis remettre l'alimentation de la carte et le tour est joué. Le programme **main.py** démarre à la mise sous tension de la carte.

Tutoriels Mu

Le site officiel : [Mu Tutorials](#).