



# Circuit Playground Express

## Développement, premier scripts



### Objectif :

Circuit Playground Express  
Micro Python



La carte est configurée, c'est le moment d'y mettre notre premier script.

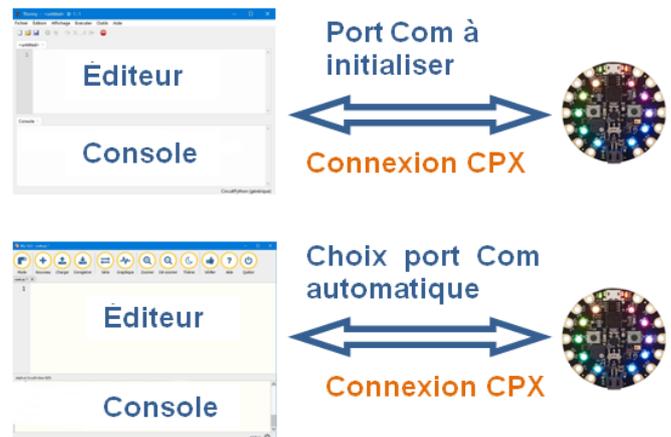
Deux environnements disponibles Mu  ou Thonny 

## 1 Principes généraux

Les deux environnements de développement, ou EDI, sont organisés de la même façon.

Chacun comporte deux zones :

- Une zone d'édition du code
- Une console en lien avec l'interpréteur micro python embarqué sur la carte CPX.



Le script qui fonctionne sur la carte doit s'appeler code.py

On démarre le script avec [F5] ou [CTRL D]

On arrête le script avec [CTRL C]

Les EDI communiquent avec la carte CPX via un port série, lié à une interface USB, il faut donc configurer cette connexion voir plus bas pour chacun des environnements.

C'est l'interface de commande REPL qui est à l'œuvre.



## 2 C'est quoi REPL Real Eval Print Loop ?

La nature interactive de MicroPython se base sur son interface de commande, officiellement appelée la fenêtre **REPL** (read-eval-print-loop), qui fonctionne généralement grâce à une connexion série reliant un PC hôte à une carte de développement de microcontrôleur.

### Principes généraux REPL / script

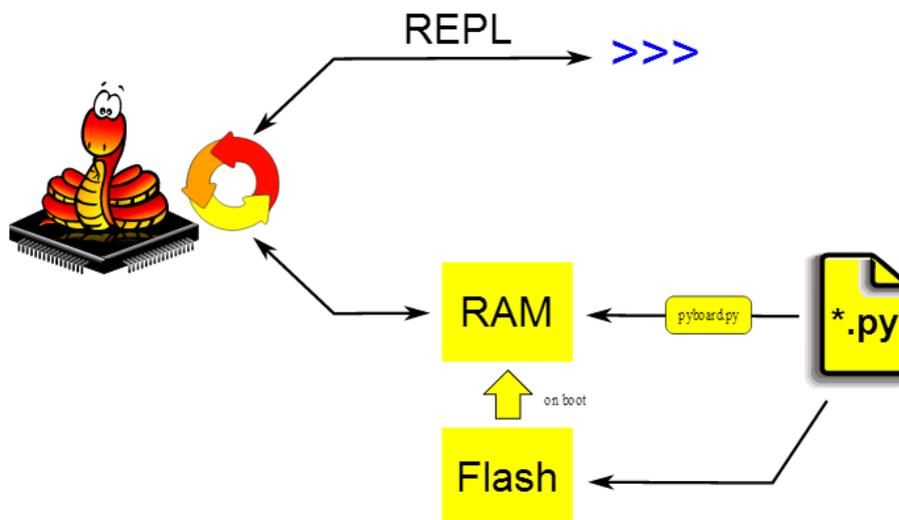
Bien que les cartes MicroPython soit des cartes à micro-contrôleur, au même titre qu'une carte Arduino, il y a quelques différences de concept qu'il faut bien intégrer dans la façon d'accéder à la carte, d'exécuter un programme.

### Python (et Micropython) : un langage interprété !

Fondamentalement, ce qu'il faut comprendre, c'est que MicroPython, tout comme le langage Python, est un **langage interprété**. Les instructions sont exécutées "à la volée" par le "moteur MicroPython" qui est le coeur actif que l'on doit avoir installé sur la carte avant toute utilisation.

A partir de là, on comprend aisément qu'il y a 2 modes de fonctionnement possibles :

- soit en envoi des instructions manuellement à la volée : c'est ce que l'on appelle le mode "interpréteur" ou REPL dans le cas de MicroPython. Concrètement, lorsque l'on utilise ce mode, on a l'équivalent d'une ligne de commande avec une invite au format `>>>` qui caractérise Python.
- soit on exécute un script, un code MicroPython, qui correspond ni plus ni moins à l'exécution automatique successive "à la volée" de toutes les instructions contenue dans le script. Noter que le script MicroPython peut appeler des fonctions placées dans d'autres scripts (que l'on va appeler "bibliothèques" ou "modules"), mais il faut bien comprendre que **dans tous les cas, c'est le coeur MicroPython qui interprète et exécute les instructions.**



Pour plus d'information voir ici :

[http://www.micropython.fr/06.technique/principes\\_generaux\\_repl\\_script/](http://www.micropython.fr/06.technique/principes_generaux_repl_script/)



### 3 Précautions à prendre pour la gestion des fichiers

Il faut sauvegarder ses fichiers sur le PC avec un nom correspondant au travail effectué dans un dossier personnel.

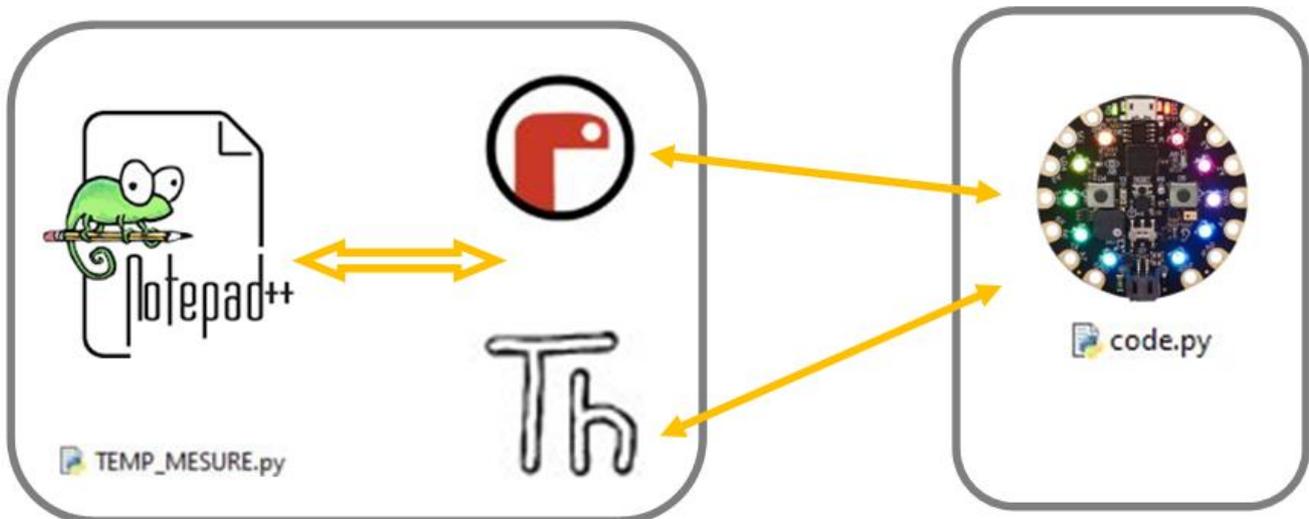
Par exemple le fichier  TEMP\_MESURE.py

Est utilisé sur la carte au travers de l'EDI dans  code.py

Les scripts seront donc développés avec leur nom sous NotePad++ puis recopiés dans l'EDI par copier coller dans le fichier code.py

De cette manière nous avons toujours une sauvegarde sur notre PC des scripts travaillés. Si nous réalisons dans la phase de mise au point des modifications directement dans l'EDI il suffira de faire un copier coller en sens inverse de l'EDI vers le PC pour conserver la dernière version des scripts.

Il est également possible de passer via le gestionnaire de fichier de Windows la carte CPX est vue comme un lecteur USB externe CIRCUITPY (G:) > ci G :

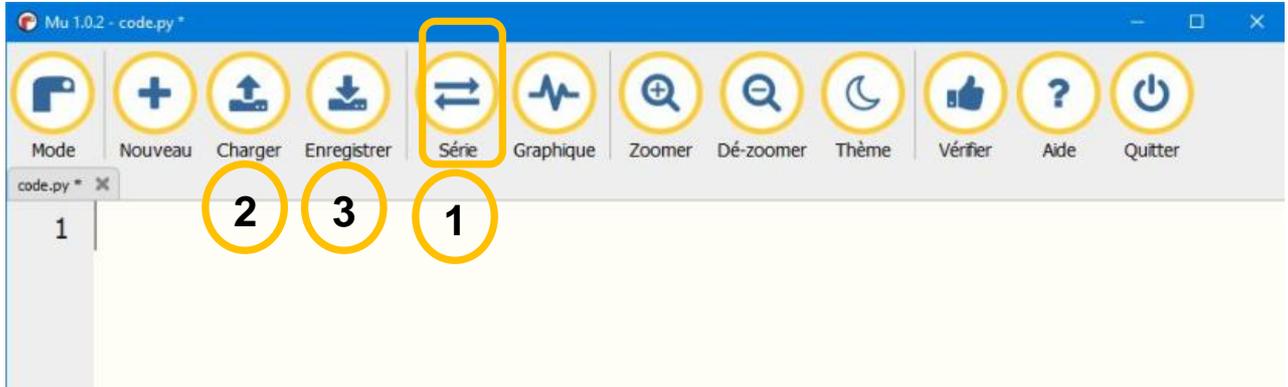




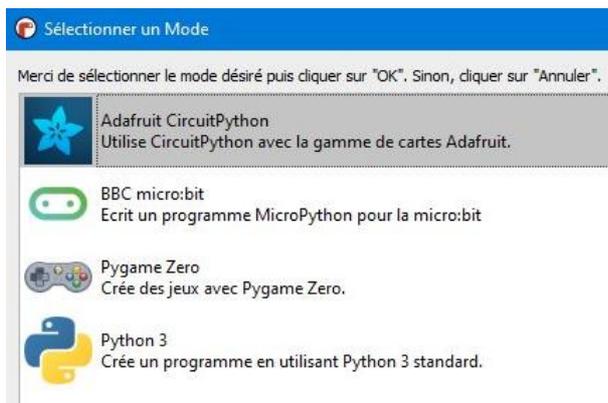
## 4 Utilisation de Mu

Pour utiliser l'EDI Mu.

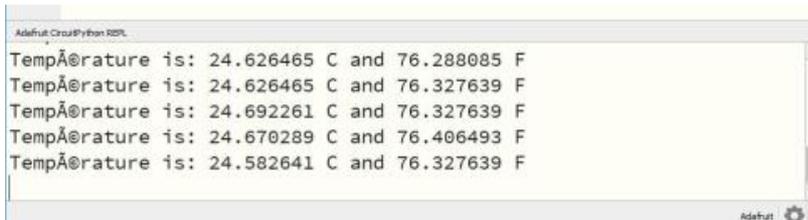
- Lancer le logiciel 



- Choisir le mode Adafruit CircuitPython (Normalement par défaut)



- Cliquer sur Série (1) pour connecter la carte  
La détection de la carte est automatique la console reçoit les messages du script en cours d'exécution.

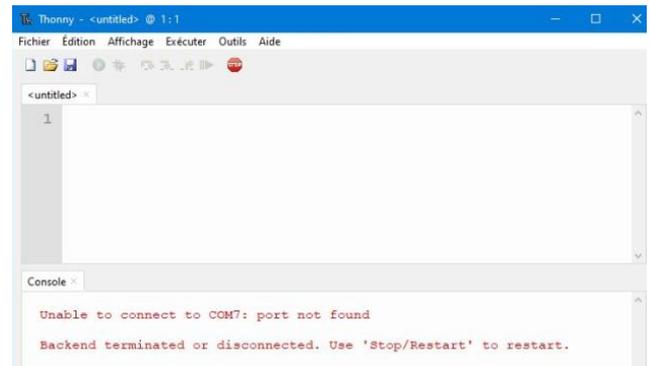
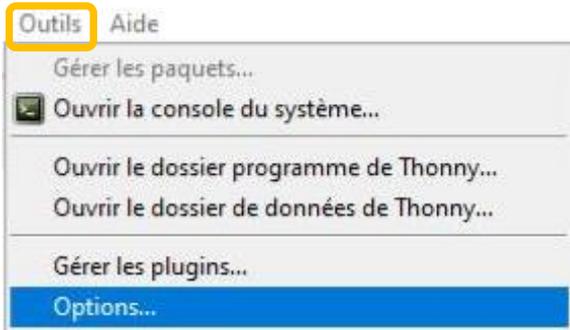


- Cliquer sur Charger (2) ou Enregistrer (3) pour lire - écrire sur la carte CPX

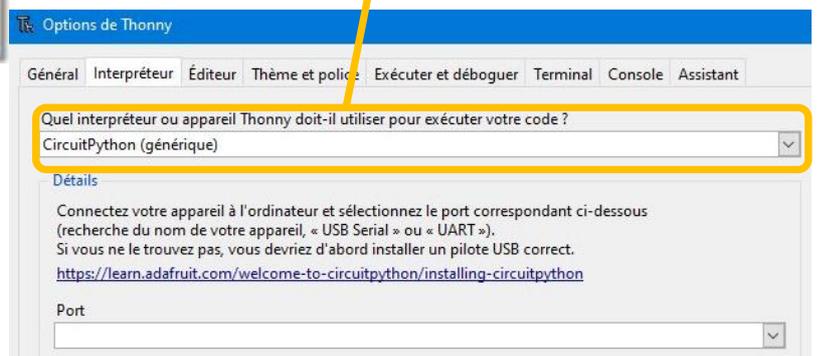


## 5 Utilisation de Thonny

- Lancement de Thonny  
Ici la carte n'est pas détectée sur la liaison série, port Com, du PC
- Sélection du port Com  
(Une carte CPX doit être connectée.)



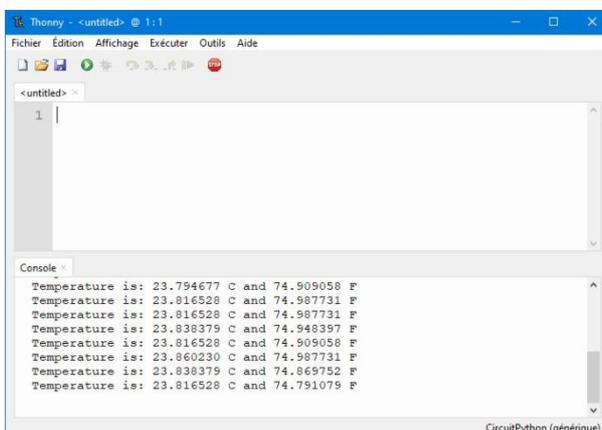
1) Choix de l'interpréteur



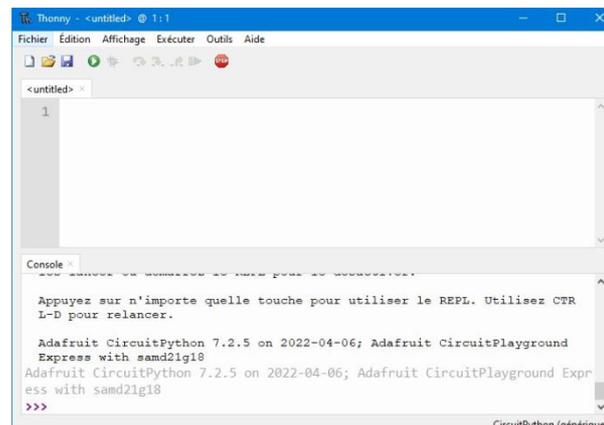
Choix du port Com



La carte est connectée les informations sont reçues dans la console.

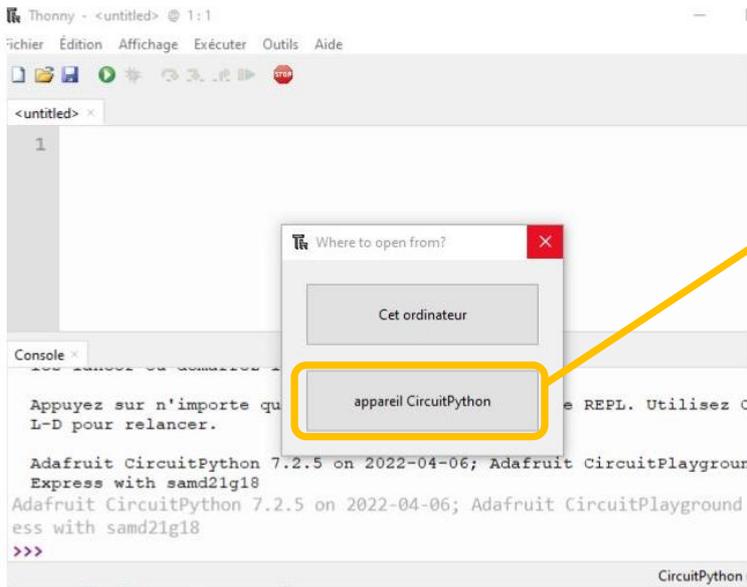


Arrêt du script pour pouvoir le télécharger [CTRL C]

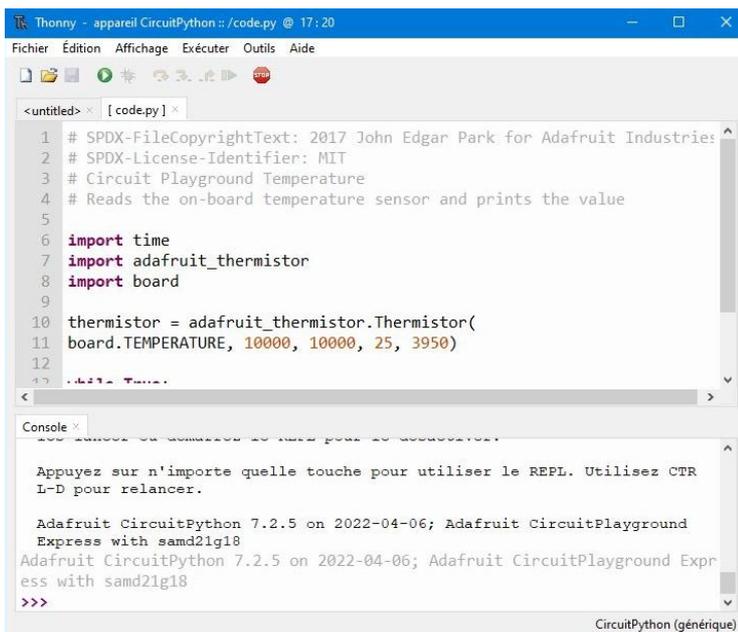
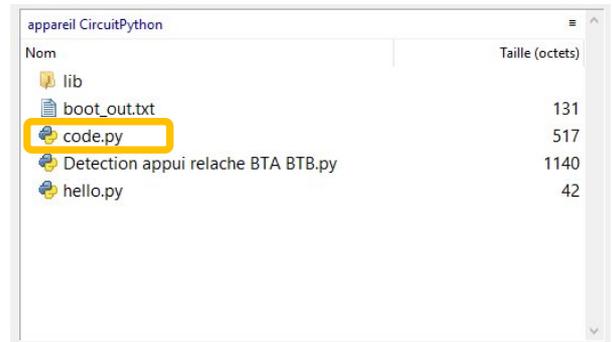




## Chargement du script code.py



Choix de la source ici la carte CPX correspond à l'appareil CircuitPython



Le script est chargé on relance le programme avec [CTRL D]

