

Activité

Programmation en python d'une carte Micro:Bit

Objectifs :

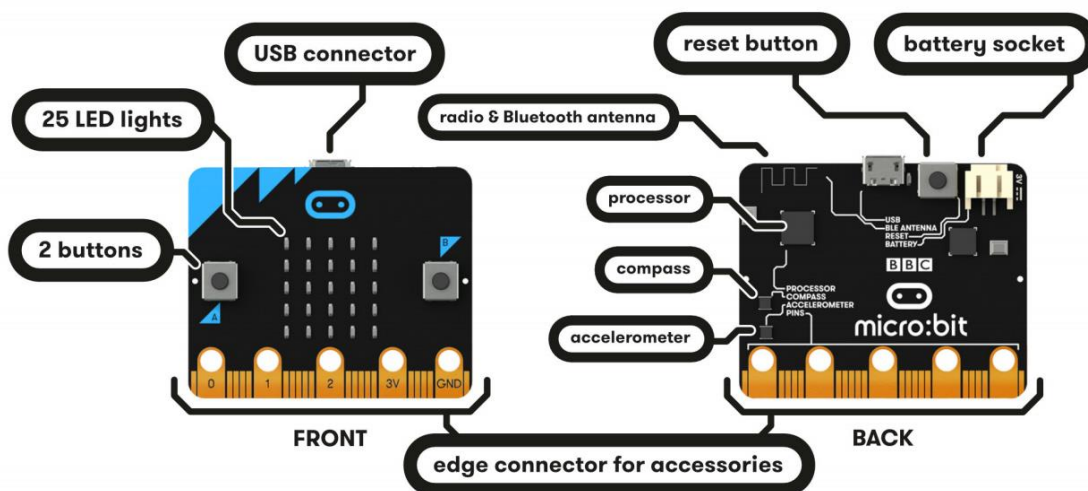
Le but de ce programme est de se familiariser avec la programmation sur carte micro:bit, à savoir :

- Comment « charger » un programme sur la carte ?
- Comment déboguer un programme (identifier et corriger les erreurs) ?
- Comment exploiter les capacités d'entrées-sorties d'une carte électronique ?

Présentation de la carte micro:bit

La carte micro:bit conçue au Royaume-Uni par la BBC est un nano-ordinateur qui peut équiper un système informatique embarqué comme par exemple dans le domaine de la robotique.

Elle est munie d'un microprocesseur et de plusieurs capteurs et interfaces de connexion. Le guide de présentation en ligne est disponible sur <https://microbit.org/fr>.



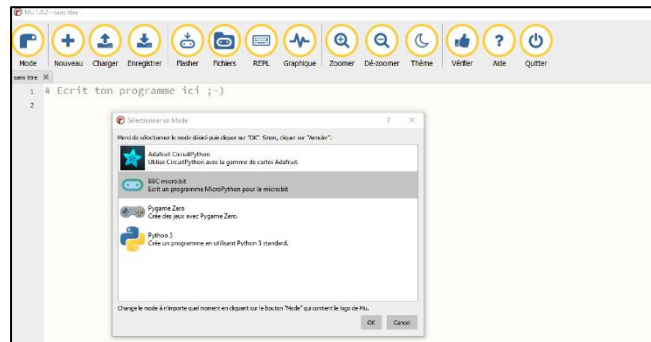
Nous utiliserons uniquement la carte en la connectant à un ordinateur avec le câble USB fourni qui assure la liaison de communication et d'alimentation en énergie électrique. Si on veut intégrer la carte dans un système embarqué, il est possible de la connecter à une alimentation externe par piles.

Lorsque la communication entre l'ordinateur et la carte échoue ou si vous souhaitez relancer un programme vous utiliserez le bouton *reset* situé au verso.

Programmes à saisir et à tester.

Nous programmerons la carte avec le langage Python et son module micro:bit.

Lancer l'environnement de programmation Mu depuis votre ordinateur et sélectionner le mode micro:bit.



BBC

a. Sélectionner *Nouveau* dans la barre de menu pour créer un nouveau programme puis **enregistrer** le fichier sous le nom programme1.py dans un dossier de votre espace personnel sur le réseau pédagogique.

b. Saisir dans l'éditeur de texte le code suivant en respectant bien l'indentation c'est-à-dire l'espacement par rapport à la marge de gauche. Ce programme utilise la structure « répéter toujours », ainsi les trois lignes de code sont répétées en boucle.

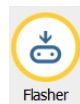
```
1 from microbit import *
2
3 while True:
4     display.scroll("Coucou")
5     display.show(Image.HAPPY)
6     sleep(1000)
7
```

Vérifier votre saisie avec le bouton *Vérifier* :



Corrigez si nécessaire puis

Télécharger le programme dans la carte Micro:Bit avec le bouton *Flasher* :



D'autres images sont disponibles sur [la liste en ligne](#), testez-en quelques-unes sur votre programme.

c. Ecrire puis tester un programme capable de réaliser la séquence suivante :

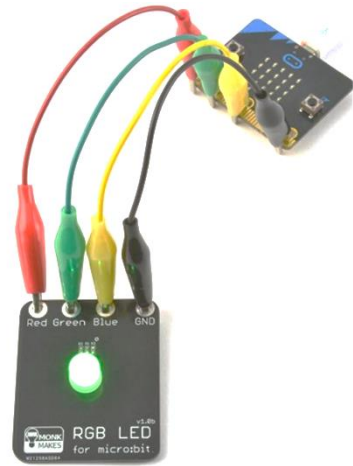
- afficher votre prénom
- afficher un triangle pendant 0,5 s
- afficher un carrée pendant 1 s
- afficher « Au revoir »
- attendre 2 secondes

d. Tester ce programme qui utilise la structure « Si Alors ».

```
1 from microbit import *
2
3 while True:
4     if button_a.is_pressed():
5         display.show(Image.HAPPY)
6     if button_b.is_pressed():
7         display.show(Image.SAD)
```

e. **Câbler** le module de LED sur la carte à l'aide des fils en respectant les liens suivants :

- Red → broche 0
- Green → broche 1
- Bleu → broche 2
- GND → GND



Programmer la carte en utilisant le code suivant puis tester-le.

```
from microbit import *

def prog_couleur(R, V, B):
    pin0.write_analog(R*4)
    pin1.write_analog(V*4)
    pin2.write_analog(B*4)

while True:
    prog_couleur (255, 0, 0)
    sleep(1000)
    prog_couleur (0, 255, 0)
    sleep(1000)
    prog_couleur(0, 0, 255)
    sleep(1000)
```

L'intensité lumineuse de chaque couleur est définie par la fonction *prog_couleur(R,V,B)* où les variables R, V et B peuvent prendre toutes les valeurs de 0 à 255.

Modifier le programme afin de générer le cycle de couleur suivant :

{ rouge, vert, bleu, magenta, cyan, jaune, blanc }

f. **Programmer** la micro:bit afin de respecter le comportement suivant :

Si l'on actionne le bouton A :

Ecrire « France » sur la micro:bit puis faire défiler les couleurs bleu, blanc, rouge sur la Led.

Si l'on actionne le bouton B :

Ecrire « Italie » sur la micro:bit puis faire défiler les couleurs vert, blanc, rouge sur la Led.