

# Informatique embarqué et objets connectés - Plan de travail

---

SNT – avril 2023

Savoir-faire de la séquence

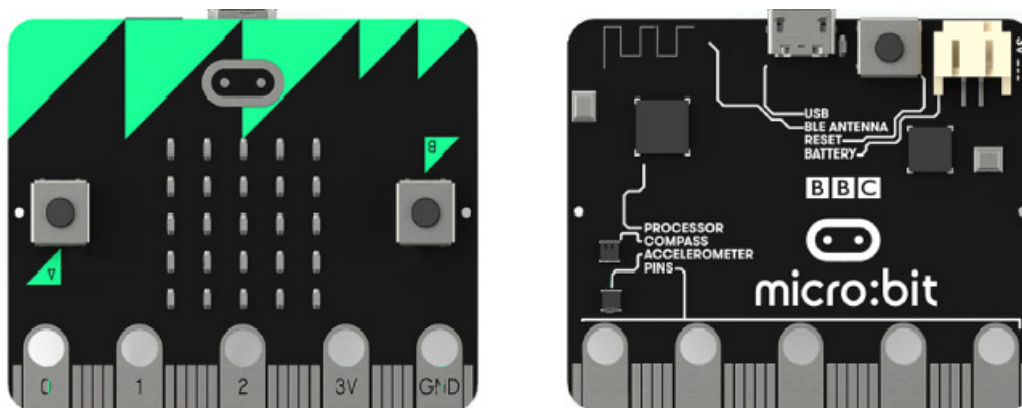
- 

Ordre des étapes à respecter

**1**

Exercice 1 : Découverte du Microbit ..... ☆☆☆☆☆

Ci-dessous, vous avez une image représentant la carte Micro:bit.



- Classer les éléments suivants dans les trois catégories (capteurs, actionneurs et transmetteurs)
 

<ul style="list-style-type: none"> <li>● Capteurs de lumière</li> <li>● Broches de connexion (pin ou GPIO)</li> <li>● Capteurs de température</li> <li>● Une boussole</li> <li>● Capteurs de mouvements (accéléromètre et boussole)</li> </ul>	<ul style="list-style-type: none"> <li>● Interface USB</li> <li>● 25 LEDs programmables individuellement</li> <li>● 2 boutons A et B</li> <li>● Communication sans fil, via Radio et Bluetooth</li> </ul>
--	---
- Repérer sur la carte le éléments précédemment listé.
- Imaginer 3 utilisations possibles de cette carte pour faire des objets connectés.

## Exercice 2

## Matrice Led

Dans cet exercice, vous apprendrez à utiliser la matrice de led.

Page de la documentation officiel traitant des images :

<https://microbit-micropython.readthedocs.io/fr/latest/tutorials/images.html>

La documentation nous invite à essayer ce premier programme

```
1 from microbit import *
2 display.show(Image.HAPPY)
```

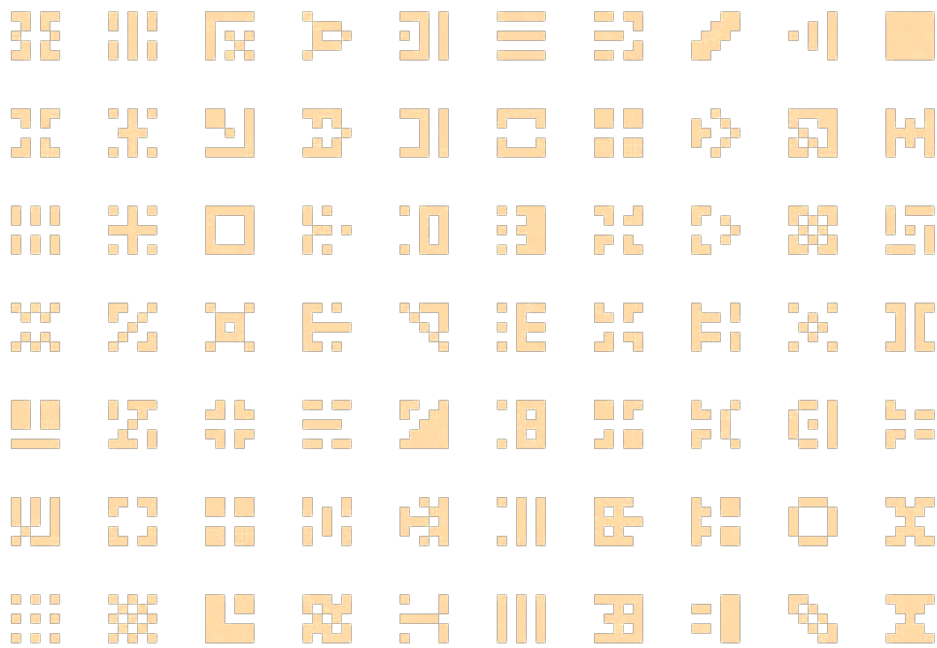
- Écrire le programme proposé et le déposer sur la carte pour voir le dessin affiché.
- Affichage d'images pré-dessinées
  - En vous basant sur la documentation, modifier le programme précédent pour afficher un visage de colère.
  - Pour afficher plusieurs images les unes à la suite des autres, il faut faire une pause entre.

```
1 from microbit import *
2 import time
3
4 display.show(Image.HAPPY)
5 time.sleep(1)
6 display.show(Image.SAD)
```

Modifier le programme précédent pour faire tourner une aiguille comme une montre.

- Affichage d'images personnelles

- Lire la suite de la documentation pour afficher des images personnelles.
- Choisir deux images de cette banque et les afficher sur la matrice de led.



- Affichage de texte : La méthode `display.scroll(...)` permet de faire défilé du texte. Faire défiler votre nom sur la matrice.

### Exercice 3

### Animations

Dans cet exercice, on propose de faire des animations sur la matrice de leds. Par exemple faire déplacer un point sur la matrice.

```

1 from microbit import *
2 import time
3
4 for col in range(5):
5     display.set_pixel(col, 0, 9)
6     # set_pixel(colonne, ligne, intensité)
7     time.sleep(0.5)

```

Modifier le programme précédent pour faire les animations suivantes :

- Éteindre la led allumée avant d'activer la suivante pour donner l'impression que c'est un point qui bouge.
- Faire clignoter deux fois chaque led de la première ligne avant de passer à la suivante.
- Faire parcourir au point toute la matrice led.

### Exercice 4

### Utiliser les capteurs

Dans le premier exercice, nous avons vu que la carte micro-bit contenait plusieurs capteurs. Voici un programme simple qui affiche la température.

```

1 from microbit import *
2 import time
3
4 while True:
5     temp = temperature()
6     display.scroll(temp)
7     time.sleep(1)

```

1. Programmer la carte avec le code ci-dessus et vérifier qu'il fonctionne.
2. Expliquer ce que fait chaque ligne du programme. À quoi peut servir la ligne `while True` ?

## Exercice 5

## Utiliser les boutons

Ci-dessous un exemple de programme utilisant les boutons.

```
1 from microbit import *
2 import time
3
4 while True:
5     if button_a.is_pressed():
6         display.show(Image.HAPPY)
7         time.sleep(1)
8         display.clear()
9     elif button_b.is_pressed():
10        display.show(Image.SAD)
11        time.sleep(1)
12        display.clear()
13    else:
14        display.show(Image.HEART)
```

1. Programmer la carte avec le code ci-dessus et vérifier qu'il fonctionne. Que fait ce programme ?
2. Transformer ce programme pour qu'il affiche la température quand le bouton A est pressé.
3. Écrire un programme qui allume la led centrale, qui la fait se déplacer à droite quand le B est pressé et vers la droite si c'est le bouton A.

## Exercice 6

## Jeu final - Pierre Feuille Ciseau

On souhaite créer un jeu de pierre feuille ciseau sur micro-bit pour jouer contre l'ordinateur.

1. Dessiner 3 icônes pierre, feuille et ciseaux puis les afficher les unes à la suite des autres sur l'écran du micro-bit.
2. Le programme ci-dessous permet de choisir au hasard entre "P" (pour pierre), "F" (pour feuille) et "C" (pour ciseau)

```
1 from microbit import *
2 from random import choice
3 import time
4
5 while True:
6     choix = choice(["P", "F", "C"])
7     if choix == "P":
8         display.scroll("Pierre")
9     elif choix == "F":
10        display.scroll("Feuille")
11    elif choix == "C":
12        display.scroll("Ciseaux")
13    time.sleep(1)
```

Adapter le programme pour qu'il affiche vos icônes à la place du nom du choix.

3. Ajouter un décompte (de 3 à 1) au début du programme (après le `while True`).

4. Avant de faire faire un choix au hasard au microbit, donner la possibilité à l'utilisateur de faire sélectionner son choix :

- Quand bouton A est pressé, c'est Ciseau
- Quand bouton B est pressé, c'est Pierre
- Quand bouton A et B sont pressés, c'est Feuille

Si aucun choix n'a été fait, le joueur perd.

Afficher le choix de l'utilisateur puis celui de l'ordi.

5. Après que le choix de l'utilisateur et celui du microbit sont affichés, afficher une icône pour savoir si on a gagné ou perdu.
6. Faire évoluer son programme pour que le décompte se fasse sur 5 secondes et est représenté avec les led de la première ligne qui s'éteignent une par une.
7. Compter les points et afficher le score du joueur à chaque début de parties (avant le décompte).
8. Bonus : adapter ce jeu pour que deux personnes sur micro-bit puissent s'affronter.