

## 1. Objectifs

Écrire des programmes simples d'acquisition de données ou de commande d'un actionneur.

## 2. Manipulation de capteurs : les boutons

```
1 from microbit import *
2
3 happy = Image("00000:09090:00000:90009:09990")
4
5 while True:
6     if button_a.is_pressed():
7         display.show(happy)
8     else:
9         display.clear()
```

✂ À Faire 1 : Effectuer les actions suivantes :

1. Connecter la micro:bit à l'ordinateur via le câble USB,
2. Se rendre à l'adresse <https://python.microbit.org/v/3>
3. Copier le code ci-dessus dans l'interface Web et cliquer sur « Send to micro:bit ».
4. Décrire l'effet du programme sur la carte. Une interaction est-elle possible ?

5. Quelle ligne fait référence à un capteur ?

✂ À Faire 2 : Modifier le programme du À Faire 1 afin qu'il :

- affiche un visage triste lorsque le bouton B est pressé,
- affiche un ? lorsqu'aucun bouton n'est pressé.

```
1 from microbit import *
2
3
4
5
6
7
8
9
10
11
12
```

✂ À Faire 3 : Effectuer les actions suivantes :

```

1 from microbit import *
2 from random import choice
3
4 plus = Image("00900:00900:99999:00900:00900")
5 mult = Image("90009:09090:00900:09090:90009")
6 moins = Image("00000:00000:99999:00000:00000")
7
8 coup = choice([plus, mult, moins])
9 display.show(coup)

```

1. Copier le code ci-dessus dans l'interface Web et cliquer sur « Send to micro:bit ».
2. Décrire l'effet du programme sur la carte après plusieurs exécutions.

3. Préciser le rôle de l'instruction de la ligne 8.

✂ À Faire 4 : Modifier le programme du À Faire 3 afin qu'il simule le jeu du Pierre-Feuille-Ciseaux:

- affiche une image au hasard (Pierre, Feuille ou Ciseaux) lorsque le bouton A est pressé,
- vide la matrice de LED lorsque le bouton B est pressé.

```

1 from microbit import *
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16

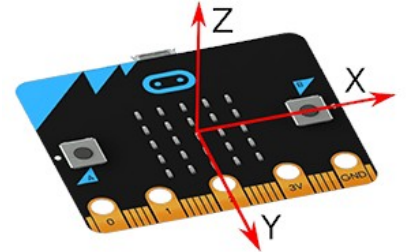
```

### 3. Manipulation de capteurs : l'accéléromètre

La carte micro :bit est livrée avec un accéléromètre. Comme sur les smartphones, elle est capable de détecter son orientation. La carte dispose de trois accéléromètres suivant les axes  $x$ ,  $y$  et  $z$ .

Trois méthodes permettent de récupérer les valeurs de ces accéléromètres en milli-g : `accelerometer.get_x()`, `accelerometer.get_y()`, `accelerometer.get_z()`.

Elles renvoient un nombre entre -1000 et 1000 correspondant à l'inclinaison de l'axe, respectivement  $x$ ,  $y$  et  $z$ .



**À Faire 5 :** Effectuer les actions suivantes :

```
1 from microbit import *
2 x = 0
3 y = 2
4 while True:
5     dx = accelerometer.get_x()
6     dy = accelerometer.get_y()
7
8     if dx > 600:
9         x = 4
10    elif dx > 300:
11        x = 3
12    elif dx < -600:
13        x = 0
14    elif dx < -300:
15        x = 1
16    else:
17        x = 2
18    # Espace à modifier pour le A Faire 6
19    display.set_pixel(x, y, 9)
20    sleep(20)
21    display.clear()
```

1. Copier le code ci-dessus dans l'interface Web et cliquer sur « Send to micro:bit ».
2. Décrire l'effet du programme sur la carte. Une interaction est-elle possible ?

3. Préciser le rôle du bloc d'instruction de la ligne 8 à 17.

**À Faire 6 :** Modifier le programme du **À Faire 5** pour que le pixel affiché puisse aussi se déplacer verticalement (axe  $y$ ).

L'accéléromètre sait aussi interpréter les données d'accélération en gestes prédéfinis.

✂ À Faire 7 : Effectuer les actions suivantes :

```
1 from microbit import *
2
3 while True:
4     gesture = accelerometer.current_gesture()
5     if gesture == "shake":
6         display.show(Image.HAPPY)
7         sleep(100)
8     display.clear()
```

1. Copier le code ci-dessus dans l'interface Web et cliquer sur « Send to micro:bit ».
2. Décrire l'effet du programme sur la carte après plusieurs « secouages » de la carte.

3. Préciser le rôle de l'instruction de la ligne 4.

✂ À Faire 8 : Compléter le programme suivant pour créer un « dé virtuel ». Quand la carte est secouée, une face de dé est tirée au hasard et est affichée.

```
1 from microbit import *
2 from random import choice
3
4 faces = [1, 2, 3, 4, 5, 6]
5
6 while True:
7     geste = accelerometer.current_gesture()
8
9
10
11
```

## 4. Synthèse

✂ À Faire 9 : Compléter le texte suivant avec les termes adéquats.

En considérant la carte micro:bit comme un système embarqué, les boutons et l'accéléromètre sont des .

Il existe plusieurs commandes pour les manipuler :

- `button_a.is_pressed()` :
- `accelerometer.get_x()` :
- `accelerometer.current_gesture()` :