

## Construire un niveau numérique

### Référentiel, Compétences

Lycée :

- Mesurer une grandeur physique à l'aide d'un capteur.
- Utiliser un dispositif comportant un microcontrôleur.
- Variabilité de la mesure d'une grandeur physique.

Lycée Professionnel :

- Echelles caractéristiques d'un système.
- Etudier expérimentalement le basculement d'un solide posé sur un plan.
- Calculs trigonométriques (lien avec les mathématiques)

Compétences :

- **S'approprier** : Représenter une situation par un schéma.
- **Analyser Raisonner** : Procéder à des analogies.
- **Réaliser** : Mettre en œuvre les étapes d'une démarche.
- **Valider** : Proposer d'éventuelles améliorations d'une démarche.
- **Communiquer** : Présenter une démarche de manière argumentée. Echanger entre pairs.

### Situation déclenchante

Comment peut-on s'assurer de l'horizontalité ou de la verticalité d'un tableau, d'une conduite de câbles électriques ?

Nous allons voir dans cette activité, qu'avec l'aide d'une carte BBC micro:bit, on peut aisément construire un niveau, qui permettra de déterminer la verticale et l'horizontale d'un lieu.



# Construire un niveau numérique

## Problématique

Comment fonctionne l'accéléromètre de la carte BBC micro:bit ?

Comment utiliser la carte BBC micro:bit afin de fabriquer un niveau à bulle numérique ?



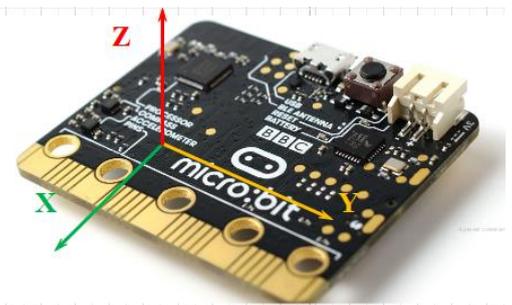
## Fiche méthode

### L'accéléromètre de la carte BBC micro:bit

La carte BBC micro:bit est équipée d'un circuit accéléromètre MMA8652 produit par la société NXP/Freescale. Il permet d'obtenir une mesure de l'accélération suivant les trois axes X, Y, Z, positionnés comme sur la figure suivante :

La connaissance de ces trois composantes de l'accélération permet de calculer l'angle d'inclinaison du dispositif.

Attention, le sens de la carte à une importance. Respecter cette configuration.



### Fonctionnement du capteur

Chaque capteur fournit une information sur l'accélération enregistrée sur l'axe donné. Pour simplifier, l'accélération est une information sur la variation de la vitesse.

Les valeurs sont accessibles dans le code en utilisant les fonctions Python suivantes :

**acceleration.get\_x()** : pour obtenir la valeur de l'accélération suivant l'axe X.

**acceleration.get\_y()** : pour obtenir la valeur de l'accélération suivant l'axe Y.

**acceleration.get\_z()** : pour obtenir la valeur de l'accélération suivant l'axe Z.

**acceleration.get\_values()** : pour obtenir les 3 valeurs dans un tuple (**accX** , **accY**, **accZ**)

La valeur renvoyée par ces 3 fonctions est l'accélération en mg (milli-g), sachant que :

1g=1000 mg.

Le 'g' est l'unité d'accélération utilisée dans l'industrie de l'aéronautique, ou l'automobile par exemple. En physique l'unité du système international pour l'accélération est le mètre par seconde carré ( $m.s^{-2}$ ).

On a la correspondance suivante :

1g = 9,81  $m.s^{-2}$ , il correspond à l'accélération de la pesanteur utilisée dans la célèbre formule

$P = m.g$  !



## Fiche méthode

### Matériel nécessaire

- Une carte BBC microbit.
- Un fil à plomb (éventuellement) ou un niveau à bulle pour contrôler.
- Une calculatrice graphique TI-83 Premium CE Edition Python.
- Un câble USB mini – USB micro.

### Réalisation d'un niveau électronique

On souhaite réaliser un système embarqué qui fonctionne comme un niveau. Ce type de système est particulièrement utile dans les domaines du bâtiment, ou encore de l'aménagement intérieur.

Rechercher l'inclinaison revient à mesurer l'angle que fait un plan du système avec l'axe vertical.

Un élément de masse  $m$ , exprimée en kilogrammes, subit une force d'attraction vers la Terre. Cette force est donnée par la relation :  $P = m \times g$ .

Lorsqu'elle est statique, la carte micro:bit mesure l'accélération du champ de pesanteur, soit  $1g$ . En effet, l'accélération est la même quelle que soit la masse de l'élément qui subit la force de gravitation. Cette mesure est donnée en millième de  $g$ .

Si on positionne un des axes de l'accéléromètre vers le bas, nous devons trouver  $1000 \text{ mg} = 1g$ . C'est-à-dire le champ de pesanteur terrestre classique.

On vérifiera qu'en positionnant la carte micro:bit à plat et que l'on affiche l'accélération suivant l'axe Z, on obtient la valeur  $z = -1000$ .

Cette valeur est négative car l'axe Z est orienté vers le haut, alors que le champ de pesanteur est dirigé quant à lui vers le bas (centre de la Terre).



### Remarque

Il est possible de tester les trois axes et de vérifier ainsi que l'on peut mesurer le champ de pesanteur en positionnant correctement l'axe de l'un des trois capteurs verticalement.

## Fiche méthode

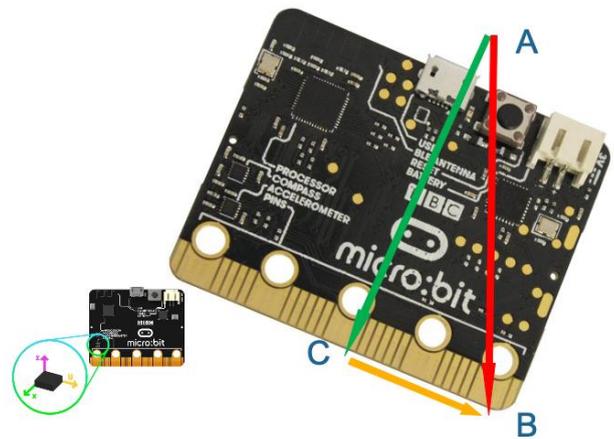
### Mise en œuvre

Le triangle  $ABC$  est rectangle en  $C$ . L'angle  $\alpha$  est donc défini par les deux mesures de l'accélération sur les axes X et Y.

$$\tan(\alpha) = \frac{\text{accélération sur l'axe Y}}{\text{accélération sur l'axe X}}$$

L'algorithme du script est le suivant :

```
Acc_x ← capteur accélération axe X
Acc_y ← capteur accélération axe Y
Inclinaison ←  $\tan^{-1} \alpha$ 
Afficher inclinaison en degrés
```



- Chargement des modules de la carte microbit.
- La fonction niv déclenche la mesure de l'accélération selon X et Y, puis renvoie le calcul de l'inclinaison.
- Affichage de l'inclinaison en degrés.

```
ÉDITEUR : NIVEAU
LIGNE DU SCRIPT 0001
from math import *
import ti_plotlib as plt
from microbit import *
from mb_sensr import *
from ti_system import *
def niv():
    acc_x=accelerometer.get_x()
    acc_y=accelerometer.get_y()
    inclinaison=int(atan2(acc_y,ac
    c_x)*180/pi)
    incl=round(inclinaison,1)
    return incl
Fns... | a | A | # | Outils | Exéc | Script
```

## Fiche méthode



```
PYTHON SHELL
>>>
>>> niv()
90
>>> |
```

Affichage en continu de l'inclinaison.

Créer une fonction **aff**.

- Insérer une boucle **Tant que**.
- Utiliser **sleep** pour avoir un délai d'affichage (en seconde).
- Affecter la valeur de l'inclinaison à la variable **l** en utilisant la fonction **niv**.
- **plt.cls** effacer l'écran.
- Afficher le message **"Inclinaison"** à la ligne 7 de l'écran et au centre.

```
ÉDITEUR : NIVEAU
LIGNE DU SCRIPT 0022
def aff():
    while not escape():
        sleep(1)
        l=niv()
        plt.cls()
        msg="Inclinaison deg =%.1f"%
        l
        plt.text_at(7,msg,"center")
```

```
PYTHON SHELL

Inclinaison =90.0 deg
```

Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus !

