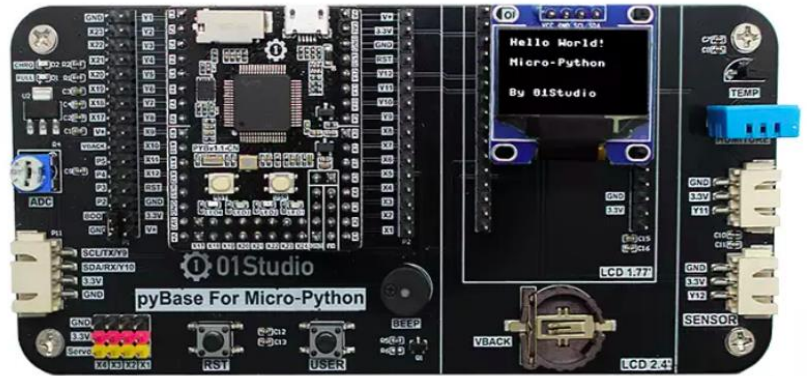


## TP prise en main de la PyBoard

Il s'agit d'une carte constituée d'un microcontrôleur de type Micro-Python, d'un écran OLED de 0,9', et d'un certain nombre de capteurs et connexions prémontés.

Il supporte le langage Python, et vous pourrez ainsi récupérer des valeurs des différents capteurs et programmer leur affichage par exemple.

**Il sera important de prendre le temps de bien comprendre le code que vous rentrez...**



Commençons par le Hello World des cartes : allumer une LED !

Nous allons allumer la LED 4 bleu.

Brancher grâce à l'USB votre carte à l'ordinateur.

Pour cela dans Thonny, rentrez le code suivant :

```
import pyb
pyb.LED(4).on()
```

Sauvegarder ce programme sous le nom **main.py**.

Il faudra toujours utiliser ce nom pour vos programmes avec cette carte !

Copier le sur la mémoire flash de votre carte (PYBFLASH).

**ATTENDRE QUE LA LED ROUGE 1 S'ÉTEIGNE**

Appuyer sur le bouton RST de la carte.

La LED 4 doit s'allumer en bleu.

Tenter le code suivant :

```
import pyb
pyb.LED(2).on()
```

Maintenant, nous allons voir comment faire allumer et éteindre les LEDs les unes après les autres :

Voici le code :

**Conseil : vérifiez bien que les décalages (indentations) soient les mêmes : une indentation = 1 TAB = 4 espaces.**

```
from pyb import LED,delay
#Extinction des LEDs
LED(2).off()
LED(3).off()
LED(4).off()
#while True signifie boucle infinie
while True:
#LED2 allumée durant 1s
    LED(2).on()
    delay(1000)
    LED(2).off()
#LED3 allumée durant 1s
    LED(3).on()
    delay(1000)
    LED(3).off()
#LED4 allumée durant 1s
    LED(4).on()
    delay(1000)
    LED(4).off()
```

Ce code peut être raccourci à l'aide d'une boucle :

```
from pyb import LED,delay
for i in range(2,5):
LED(i).off()
```

```
while True:
    #Use for loop
    for i in range(2,5):
        LED(i).on()
        delay(1000) #delay 1000ms(1second)
        LED(i).off()
```

Explication :

```
for i in range(2,5):
```

Cela permet de faire une boucle pour i variant de 2 à 4 (pas 5). Ainsi, on peut allumer les LEDs comme nous le voulons...

### Utilisation du bouton USER :

Les boutons sont importants dans l'interface utilisateur/machine.

Nous avons un bouton le bouton sur la carte : USER

Voici le code :

```
from pyb import LED,Switch
```

```
sw = Switch() #Défini l'objet Switch comme sw
# Quand le bouton USER est pressé la LED4 s'allume ou s'éteint
sw.callback(lambda:LED(4).toggle())
```

La commande `sw.callback` permet ensuite de mettre ce que vous voulez qu'il soit réalisé une fois le bouton pressé.

### Utilisation du bouton USER :

Voici un exemple de code pour mieux comprendre. Nous allons faire un code pour allumer une Led quand on appuie sur le bouton USER et l'éteindre s'il n'est pas appuyé.

**Conseil : vérifiez bien que les décalages (indentations) soient les mêmes : une indentation = 1 TAB = 4 espaces.**

```
from pyb import Pin
#Paramètre la Led4 (B4) en mode push pull sortie
p_out=Pin('B4',Pin.OUT_PP)
#Paramètre le bouton USER (X17) en mode entrée
p_in = Pin('X17', Pin.IN, Pin.PULL_UP)
while True:
    if p_in.value()==0: #Usr est pressé
        p_out.high() #Allumage de la Led4
    else:
        p_out.low() #Extinction de la Led4
```

### Utilisation de l'écran OLED

Votre carte dispose d'un écran OLED qui permet l'affichage de données de capteurs par exemple ou de messages.

Commandes Python :

Nous utiliserons des éléments du module Machine : I2C et Pin et du module `ssd1306` pour driver l'écran.

*Remarque :*

*I2C est un bus informatique (dispositif de transmission de données) devenu très commun. La connexion est réalisée à l'aide de 2 lignes : SDA (Serial Data Line) et SCL (Serial Clock Line).*

Voici un exemple de code :

```
from machine import I2C,Pin # importation des modules I2C et Pin
from ssd1306 import SSD1306_I2C # importation du module ssd1306
# initialisation de I2C :sda --> Y8, scl --> Y6
i2c = I2C(sda=Pin("Y8"), scl=Pin("Y6"))
```

```
#Init de l'écran:128*64, l'adresse est 0x3c
oled = SSD1306_I2C(128, 64, i2c, addr=0x3c)
oled.text("Hello World!", 0, 0)      # 1ere ligne
oled.text("Club Prog", 0, 20)       # 2eme ligne
oled.text("LAB", 0, 50)             # 3eme ligne
oled.show()    # affichage à l'écran
```