

## Colorimètre à Micro:bit

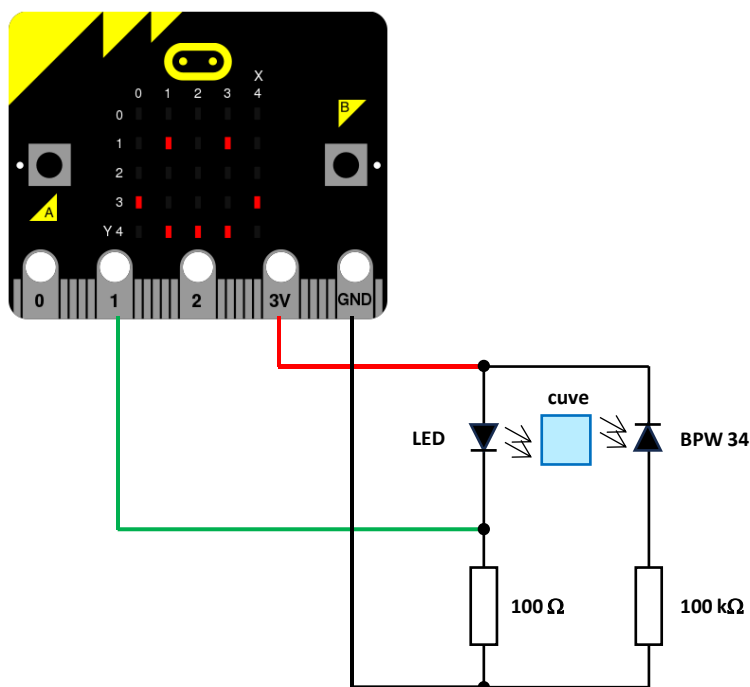
*But : réaliser un colorimètre simplifié à l'aide d'une carte Micro:bit et de quelques composants discrets, afin de réaliser des expériences de dosage d'une solution colorée par étalonnage, ou de suivi d'une cinétique.*

### I./ Réalisation du montage :

Le montage est actuellement réalisé sur une platine d'essai. La liste des composants est :

- Une LED haute intensité (couleur à déterminer en fonction de l'expérience).
- Une photodiode BPW 34
- Une résistance de  $100\ \Omega$
- Une résistance de  $100\ \text{k}\Omega$

Le schéma est le suivant :



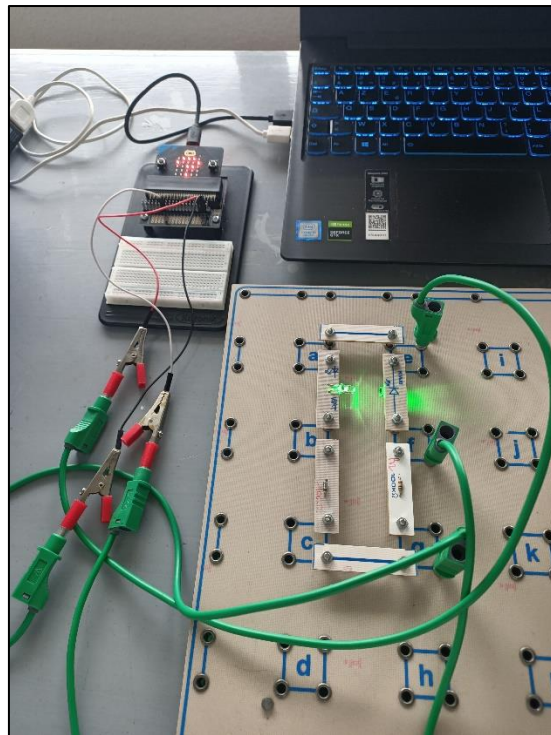
Nous avons à notre disposition 4 LED haute intensité, dont les longueurs d'onde, mesurées au spectromètre à fibre, sont :

Couleur	$\lambda$ (en nm)
Rouge	655
Jaune	580
Vert	525
Bleu	465

Ces 4 longueurs d'onde permettent de couvrir une grande partie du spectre visible.

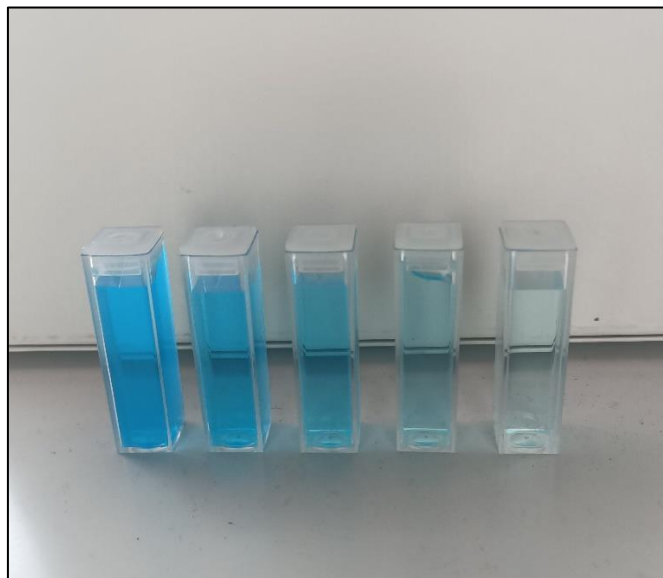
La cuve contenant l'échantillon est placée entre la LED et la photodiode. On couvre l'ensemble avec une boîte en carton sombre afin d'éviter au maximum les lumières parasites. Le dispositif est alimenté par la Micro:bit, elle-même reliée à un PC par USB, pour l'alimentation et la communication série pour récupérer les données.

En situation :



## II./ Dosage par étalonnage – échelle de teintes :

On réalise une échelle de teinte avec des solutions diluées de sulfate de cuivre :



Les concentrations des différentes solutions sont respectivement de  $1 \text{ mol}\cdot\text{L}^{-1}$ ,  $0,5 \text{ mol}\cdot\text{L}^{-1}$ ,  $0,25 \text{ mol}\cdot\text{L}^{-1}$ ,  $0,125 \text{ mol}\cdot\text{L}^{-1}$  et  $0,1 \text{ mol}\cdot\text{L}^{-1}$ .

On réalise les mesures en utilisant le programme Python suivant :

```

"""
Colorimètre Micro:bit
Mesures d'absorbance v1.0
Décembre 2023 par Bruno PEYROT
"""

from microbit import *
from math import *

zero = 512
display.clear()

while True:
    sleep(1000)
    display.show("?")
    print()
    print("A : Etalonnage")
    print("B : Mesures")
    print()
    while not(button_a.is_pressed()) and not(button_b.is_pressed()):
        pass
    if button_a.is_pressed():
        display.show("0")
        print()
        print("Veuillez inserer le blanc,")
        print("puis appuyez sur A")
        print()
        sleep(1000)
        while not(button_a.is_pressed()):
            lecture = pin1.read_analog()
            zero = lecture
        print()
        print("Etalonnage OK")
        print()
    if button_b.is_pressed():
        display.show("M")
        print()
        mesure = 0
        N = 1000
        for i in range(N):
            lecture = pin1.read_analog()
            mesure += log(zero / lecture) / log(10)
        A = int(mesure / N * 100) / 100
        print("Absorbance =", A)
        print()
        while button_b.is_pressed():
            pass

```

Pour la communication entre la Micro:bit et le PC, on utilise CoolTerm (<https://freeware.the-meiers.org/>) en réglant la vitesse du port série à 115 200 bauds.

Lorsqu'on lance le programme, l'écran intégré de la Micro:bit affiche « ? » et la console série affiche le menu suivant :

```

A : Etalonnage
B : Mesures

```

Pour réaliser l'étalonnage, on appuie sur le bouton A (gauche) de la Micro:bit. L'écran intégré affiche « 0 » et la console série affiche le message « Veuillez insérer le blanc puis appuyez sur A ». Il faut alors disposer la cuve d'eau distillée, puis appuyez à nouveau sur le bouton A. La console série affiche « Etalonnage OK », puis on revient au menu.

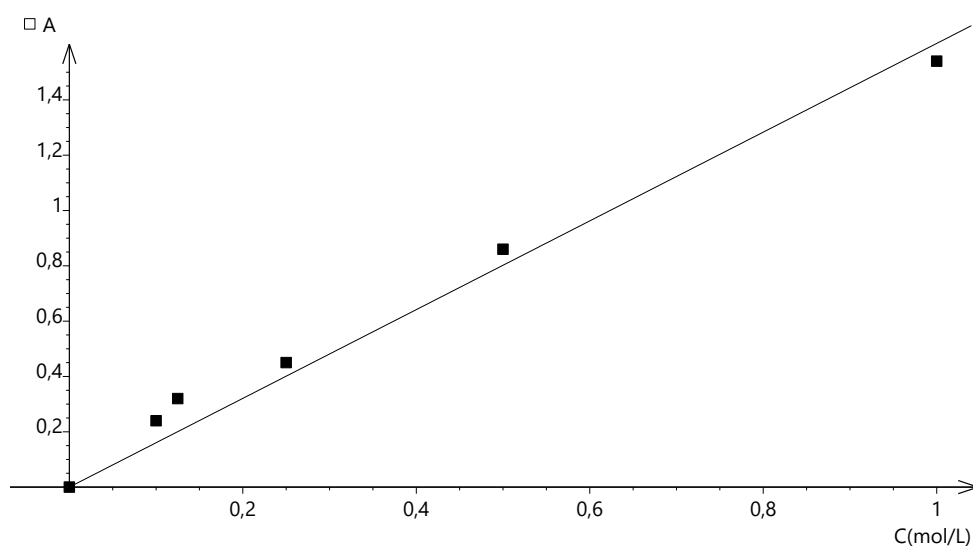
Pour réaliser la mesure, on met en place la cuve à mesurer, et on appuie sur la bouton B (droite) de la Micro:bit. L'écran intégré affiche « M » pendant environ une seconde, et la console série affiche la valeur de l'absorbance.

Pour éviter les fluctuations, la Micro:bit réalise un millier de mesure, et calcule ensuite une moyenne.

Nous avons réalisé la mesure de l'absorbance des différentes solutions de sulfate de cuivre après avoir fait le blanc avec de l'eau distillée. Le maximum d'absorption du sulfate de cuivre se situant aux alentours de 800 nm, nous avons utilisé la LED rouge. On obtient les résultats suivants :

Concentration (en mol·L <sup>-1</sup> )	Absorbance
0	0,00
0,1	0,24
0,125	0,32
0,25	0,45
0,5	0,86
1	1,54

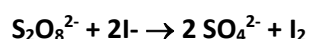
On obtient la courbe suivante :



On retrouve la loi de Beer-Lambert ( $A = k \times C$ ) avec une valeur de  $k$  égale à 1,60 avec les données présentées. La réponse de la photodiode a l'air linéaire, mais il y a un écart certain.

### III./ Suivi d'une cinétique :

On étudie la réaction se produisant entre les ions peroxodisulfate et les ions iodure :



Après avoir réalisé l'étalonnage du colorimètre avec une cuve d'eau distillée, on mélange dans la cuve 1 mL de solution de peroxodisulfate de potassium à une concentration de  $1 \times 10^{-2} \text{ mol} \cdot \text{L}^{-1}$  avec 3 mL de solution d'iodure de potassium à une concentration de  $5,0 \times 10^{-2} \text{ mol} \cdot \text{L}^{-1}$ . Le maximum d'absorption du diiode se situant aux alentours de 470 nm, on utilise la LED bleue.

On réalise le suivi cinétique à l'aide du programme Python suivant :

```

"""
Colorimètre Micro:bit
Suivi de cinétique v1.0
Décembre 2023 par Bruno PEYROT
"""

from microbit import *
from math import *

zero = 512
display.clear()

while True:
    sleep(1000)
    display.show("?")
    print()
    print("A : Etalonnage")
    print("B : Mesures")
    print()
    while not(button_a.is_pressed()) and not(button_b.is_pressed()):
        pass
    if button_a.is_pressed():
        display.show("0")
        print()
        print("Veuillez inserer le blanc,")
        print("puis appuyez sur A")
        print()
        sleep(1000)
        while not(button_a.is_pressed()):
            lecture = pin1.read_analog()
            zero = lecture
        print()
        print("Etalonnage OK")
        print()
    if button_b.is_pressed():
        display.show("M")
        print()
        print("Debut des mesures :")
        print("Appuyez sur B pour arreter")
        print()
        sleep(1000)
        temps_zero = running_time()
        while not(button_b.is_pressed()):
            lecture = pin1.read_analog()
            temps = (running_time() - temps_zero) / 1000
            A = log(zero / lecture) / log(10)
            print(temps, A)
            sleep(1000)
        while button_b.is_pressed():
            pass

```

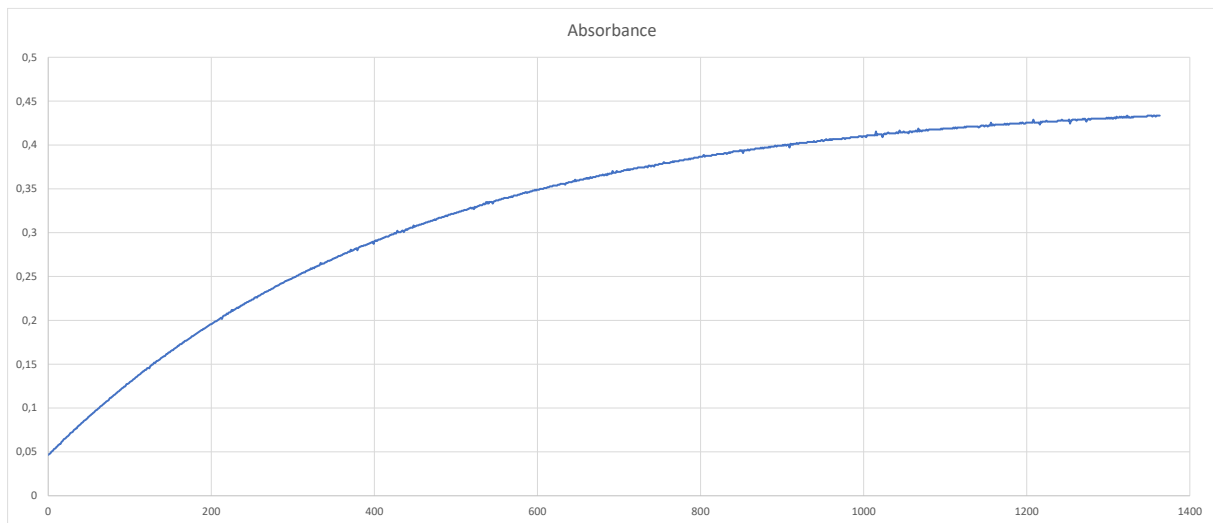
L'étalonnage se déroule de la même manière que le programme précédent. Lorsqu'on veut lancer le suivi de la cinétique, on met en place la cuve à mesurer puis on appuie sur le bouton B de la Micro:bit lorsqu'on se trouve devant le menu. L'écran intégré affiche « M » et la console série affiche « Début des mesures : appuyez sur B pour arreter ». La Micro:bit prend alors une mesure par seconde et affiche pour chaque mesure le temps et la valeur de l'absorbance dans la console série. On arrête l'acquisition en appuyant à nouveau sur B.

Afin de sauvegarder ses résultats, on peut capturer le flux de la console série dans un fichier texte. Avec CoolTerm, il suffit de faire « Connection », puis « Capture to Text/Binary File » et « Start... ».

Le fichier texte obtenu peut-être importé dans un tableur grapheur. Après avoir laissé tourner l'acquisition pendant un peu plus d'une vingtaine de minutes, on obtient de la page suivante.

La courbe est « bruitée », et cela peut venir du fait que l'on prend une mesure par seconde. Ce choix a été fait car pour des raisons de simplicité, j'ai utilisé la fonction « Sleep » pour attendre entre deux mesures. On pourrait mettre une valeur plus grande que 1000, pour séparer deux mesures

consécutives de plus d'une seconde, mais le problème serait alors l'interception de l'appui sur le bouton B afin de stopper l'acquisition.



#### IV./ Prolongements et améliorations :

- Si on se limite à seulement trois longueurs d'onde, il pourrait être intéressant d'utiliser une LED tricolore car cela permettrait de ne pas avoir à changer de LED en fonction de l'expérience.
- Réaliser le montage sur une petite platine avec un connecteur permettant de brancher la Micro:bit afin d'avoir un ensemble plus compact.
- Réaliser un boîtier permettant de positionner facilement la cuve par rapport à la LED et à la photodiode (impression 3D ?)
- Au niveau logiciel, permettre de changer la période d'acquisition.

#### V./ Liens :



Vidéo Youtube montrant l'utilisation du colorimètre

Github contenant les programmes Python et la documentation du projet

