

Fiche méthode

Référentiel, compétences

- **Capacité** : Utiliser un dispositif avec microcontrôleur et capteur.
- Utiliser un langage de programmation.
- Prendre en main un capteur de pression Grove (fig.2) connecté sur l'entrée **IN1** du Hub (fig. 1).



Fig. 1

Commentaires de l'auteur

- Le Hub est une carte à microcontrôleur disposant d'entrées et de sorties déjà intégrées programmables à l'aide d'un langage de programmation. Il est également possible d'utiliser une large liste de capteurs additionnels grâce aux entrées IN 1, IN 2 et IN 3.
- Nous allons découvrir dans cette fiche-méthode comment effectuer une mesure de pression avec un capteur de pression (fig. 2) relié au TI-Innovator™ Hub, lui-même connecté à une calculatrice TI-83 Premium CE Edition Python.
- Nous verrons ensuite un programme permettant de vérifier expérimentalement la loi de Mariotte $P \times V = \text{constante}$.



Fig. 2

Matériel

- Calculatrice TI-83 premium CE Edition Python.
- TI-Innovator™ Hub et le câble USB de liaison.
- Kit de mesure de pression Grove MPX5700AP, avec tube silicone et seringue étanche. Il permet des mesures comprises entre 15 *kPa* et 700 *kPa*.

Prérequis

- Savoir utiliser les dispositifs d'entrée et de sortie intégrés au Hub. Consulter si besoin la fiche méthode intitulée « Découverte des dispositifs intégrés du TI-Innovator™ Hub ».



Premières mesures de pression

OBJECTIF : Mesurer la pression atmosphérique avec le dispositif complet. Le capteur Grove doit être étalonné pour afficher une pression en Pascal ou en kPa.

- Rappels :
 - ✓ $1 \text{ bar} = 10^5 \text{ Pa} = 10^3 \text{ hPa} = 10^2 \text{ kPa}$
 - ✓ $1 \text{ hPa} = 1 \text{ mbar}$
 - ✓ La pression atmosphérique normale est voisine de 1013 hPa .

- FONCTIONS PYTHON

- 1) Créer un SCRIPT nommé PRESSION dont le type est **6** : Projets STEM Hub (fig.3).
- 2) Importer le module analogique du Hub à l'aide du menu **2** : Dispositifs d'entrée... (fig.4a et 4b)

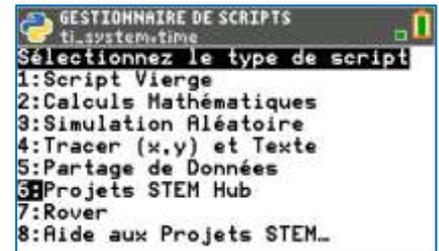


Fig. 3

Fig. 4a

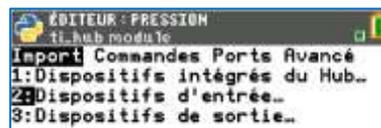


Fig. 4b



- 3) Écrire une fonction Python **p**, (fig.5), qui :
 - a. Configure le port **IN 1** du Hub en tant que dispositif d'entrée.
 - b. Effectue la mesure de la valeur de la pression atmosphérique.
 - c. Renvoie la pression **p1** exprimée en hPa. Les deux valeurs **0.04628** et **-30.26** sont empiriques et peuvent être modifiées si nécessaire. Elles sont nécessaires pour que la pression mesurée soit exprimée en *hPa*.



Fig. 5

- 4) L'appel à la fonction **p** renvoie la valeur de la pression atmosphérique, exprimée en hPa. (fig. 6).

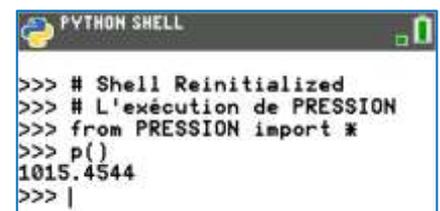


Fig. 6



Vérification de la loi de Mariotte : $P \times V = \text{constante}$

OBJECTIF

La loi de Mariotte s'énonce ainsi : « Au cours de l'évolution isotherme d'un gaz parfait, le produit $P \times V$ est constant ». La loi a d'abord été découverte en 1662 par l'Irlandais Robert Boyle puis par le Français Edme Mariotte en 1676.

Dans cette expérience, l'air sera assimilé à un gaz parfait.

PROTOCOLE EXPÉRIMENTAL

En faisant varier le volume V d'un gaz enfermé dans une seringue, on mesure sa pression à température constante.

- ✓ Le volume V de gaz sera lu sur la graduation de la seringue, très imprécise.
- ✓ La pression P du gaz sera mesurée avec le capteur de pression.
- ✓ On tracera ensuite la courbe $P = f(1/V)$.

Pour les besoins de l'expérience, la fonction **input** est *exceptionnellement* utilisée car elle permet une saisie au clavier du volume d'air enfermé dans la seringue, au cours de la détente du gaz. Il faut en effet un signal qui demande à n reprises à l'expérimentateur de tirer sur le piston de la seringue, afin de dilater le gaz.

Voici les captures d'écran du programme à utiliser (fig. 7, 8 et 9) :

```
EDITEUR : PRESSION
LIGNE DU SCRIPT 0001
# Projets STEM Hub
from ti_system import *
from time import *
from analogin import *
import ti_plotlib as plt
sensor=analog_in("IN 1")

def p():
    p=sensor.measurement()
    p1=(p*0.04628-30.26)*10
    return p1
```

Fig. 7

```
EDITEUR : PRESSION
LIGNE DU SCRIPT 0012
def mesures(n):
    l1surv=[]
    lp=[]
    for i in range(n):
        v=float(input("V="))
        p2=100*p()
        l1surv.append(1/v)
        lp.append(p2)
    return l1surv,lp
```

Fig. 8

```
EDITEUR : PRESSION
LIGNE DU SCRIPT 0025
def trace(x,y):
    plt.cis()
    plt.auto_window(x,y)
    plt.axes("on")
    plt.scatter(x,y,"+")
    plt.color(255,0,255)
    plt.lin_reg(x,y,"center",11)
    plt.show_plot()

l1surv,lp=mesures(5)
trace(l1surv,lp)
```

Fig. 9

COMMENTAIRES

Figure 7 : C'est le programme réalisé à la première partie (fig. 5).

Figure 8 : La fonction **mesures** va répéter n fois la mesure de la pression. Deux listes de valeurs sont alors construites :

- La liste **lp** qui contient les n valeurs successives de la pression, exprimée en Pa (cf **p2=100*p()**)
- La liste **l1surv** qui contient les n valeurs successives de $\frac{1}{V}$, exprimées en mL^{-1} .



Les deux instructions `l1surv`, `lp=mesures(6)` et `trace (l1surv,lp)` sont les deux appels de fonction qui rendent les mesures, puis leur tracé (avec régression linéaire) automatiques, dès le lancement du programme.

Voici les résultats obtenus pour une expérience de 6 mesures de volume (fig. 10 et fig. 11) :

- Le volume intérieur du tuyau est évalué à 3,4 mL (voir fig.12)
- Les volumes choisis sont les suivants :

V (mL)	13,4 = 10 + 3,4	15,4	17,4	19,4	21,4	23,4
--------	-----------------	------	------	------	------	------

```

PYTHON SHELL
>>>
>>> lp
[102054.52, 89049.84000000001, 7
9932.68000000001, 72157.64, 6530
8.2, 59708.32000000001]
>>> l1surv
[0.07462686567164178, 0.06493506
493506493, 0.0574712643678161, 0
.05154639175257732, 0.0467289719
6261683, 0.04273504273504274]
>>> |
Fns... a A # Outils Éditer Script
    
```

Fig. 10



Fig. 12 : sur cet exemple, le piston est sur la graduation 12 mL, ce qui correspond à un volume total d'air égal à 15,4 mL (12+3,4=15,4 mL)

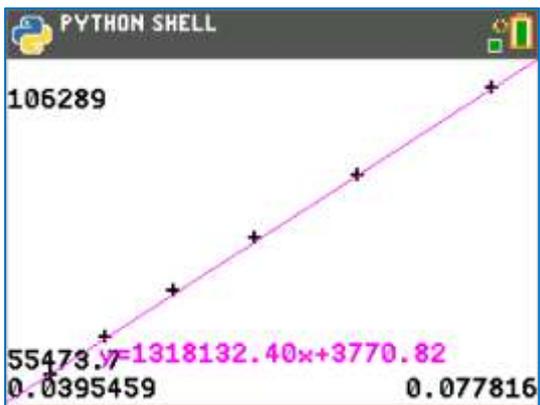


Fig. 11

CONCLUSION

La courbe $P = f\left(\frac{1}{V}\right)$ est une droite passant par l'origine du repère (fig.11). Le produit $P \times V$ est donc constant, aux incertitudes de mesure près, qui n'ont pas été évaluées ici. On aurait pu également tracer $P \times V$ en fonction de V , pour obtenir une fonction constante.

Télécharger le script PRESSION à l'adresse : <https://education.ti.com/fr/physique-chimie>.

