

- Objectifs :**
- Mesurer une température à l'aide d'un capteur résistif (un conducteur de platine)
 - Produire et utiliser une courbe d'étalonnage reliant la résistance du conducteur de platine et la température à l'aide d'un langage de programmation (python)
 - Utiliser un dispositif avec microcontrôleur et capteur (python + Pyboard + conducteur de platine).

Contexte

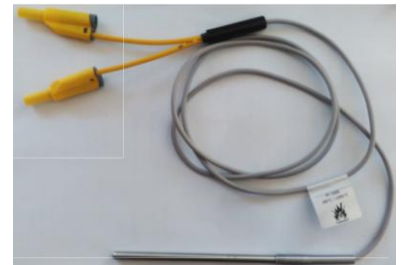
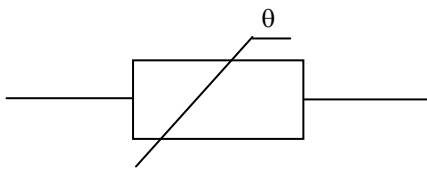
Vous disposez sur la paillasse d'un conducteur de platine, d'un microcontrôleur Pyboard et d'un générateur de courant.

Problème : Comment afficher la température de la salle de classe sur l'écran d'un ordinateur à l'aide d'un du conducteur de platine, d'un microcontrôleur et d'un programme Python.

COURBE D'ETALONNAGE DU CONDUCTEUR DE PLATINE

Document 1 : Le conducteur de platine « Pt1000 »

Symbole :



Questions :

ANALYSER

1) Vérifier à l'aide d'une manipulation simple que la résistance du conducteur de platine dépend de la température du milieu dans lequel elle se trouve.

Matériel disponible : * Conducteur de platine * Multimètre avec la fonction ohmmètre

2) On souhaite tracer la courbe d'étalonnage reliant la résistance **R** du conducteur de platine et la température **T** (c'est-à-dire une courbe qui fait la correspondance entre R et T) à l'aide d'un programme Python.

En utilisant le matériel disponible, proposer un protocole permettant de réaliser les mesures de R et T correspondantes.

Matériel : * chauffe ballon * Ballon
* conducteur de platine * Multimètre avec la fonction ohmmètre
* potence + noix de serrage + pinces

APPEL n° 1

Appeler le professeur pour lui montrer le protocole ou en cas de difficultés

REALISER

3) Réaliser les mesures en respectant les consignes de sécurité données par le professeur en limitant la température à 80 °C.

T (°C)	30	35	40	45	50	55	60	65	70	75	80
--------	----	----	----	----	----	----	----	----	----	----	----

↑
Température de départ

R (Ω)												
--------------	--	--	--	--	--	--	--	--	--	--	--	--

- 4) Copier le **dossier entier** « Conducteur de platine » situé dans L>ELEVES 2G>PC sur le lecteur U.
Ouvrir le fichier « Conducteur Pt1000 donnees.csv » à l'aide du bloc note. Entrer les mesures de R et T dans le fichier. Les données doivent être séparées par des « ; » et écrites lignes par ligne.
- 5) Ouvrir le logiciel IDLE python situé sur le bureau dans Logiciels de Math > python puis ouvrir le fichier « Conducteur Pt1000 traitement.py» situé dans le répertoire Conducteur de platine du lecteur U.
- 6) En vous appuyant sur les travaux précédents, ajouter les lignes de commandes permettant d'afficher la courbe T = f(R). Tester le programme avec la commande RUN>RUN MODULE.

APPEL n° 2	Appeler le professeur pour lui montrer le protocole ou en cas de difficultés	
------------	--	--

Document 2 : Aide python ...

* pour rechercher les paramètres d'une droite qui passe au plus près d'un nuage de points

a,b,r,c,d = stats.linregress(R,T)

R,T : grandeurs respectivement en abscisses et en ordonnée (lettres à modifier...)

a : coefficient directeur de la droite

b: ordonnée à l'origine de la droite

r : coefficient de corrélation ; plus ce coefficient se rapproche de 1 et meilleure est la modélisation

remarque : c et d ne seront pas utilisés

* pour tracer la droite de coefficient directeur **a** et d'ordonnée **b**

plt.plot(x,a*x+b,c='g') x : lettre attribuée à l'abscisse (à modifier) g : couleur exemple ici g pour green

- 7) En utilisant le document 2, compléter le programme afin qu'il puisse tracer la droite qui passe au plus près du nuage de points (droite de modélisation) et afficher l'équation de la droite à l'écran. Tester le programme avec la commande RUN>RUN MODULE.

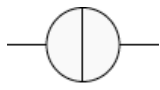
APPEL n° 3	Appeler le professeur pour lui montrer le protocole ou en cas de difficultés	
------------	--	--

Noter l'équation de la droite : T = × R


AFFICHAGE DE LA TEMPERATURE SUR L'ECRAN DE L'ORDINATEUR

On se propose d'afficher à l'écran la température de la salle de classe avec le conducteur de platine et en utilisant un programme Python.

Document 3 : Le générateur de courant (Générateur Jeulin)

Symbole : 

Cet appareil permet de générer un courant d'intensité constante I



Matériel disponible :

- * Conducteur de platine (précédemment étalonné)
- * Un microcontrôleur « Pyboard» qui renvoie un nombre N compris entre 0 et 4095 sur la pin X19 lorsqu'il mesure une tension sur cette pin.
- * Un générateur de courant d'intensité fixe $I_{pt} = 1,22 \text{ mA} = 1,22 \times 10^{-3} \text{ A}$ (à mesurer à l'ampèremètre).

Questions

On fait passer un courant d'intensité constante I_{pt} dans le conducteur de platine. La tension aux bornes de ce conducteur sera notée U_{pt} et sa résistance R_{pt} pour une température T_{pt} .

ANALYSER

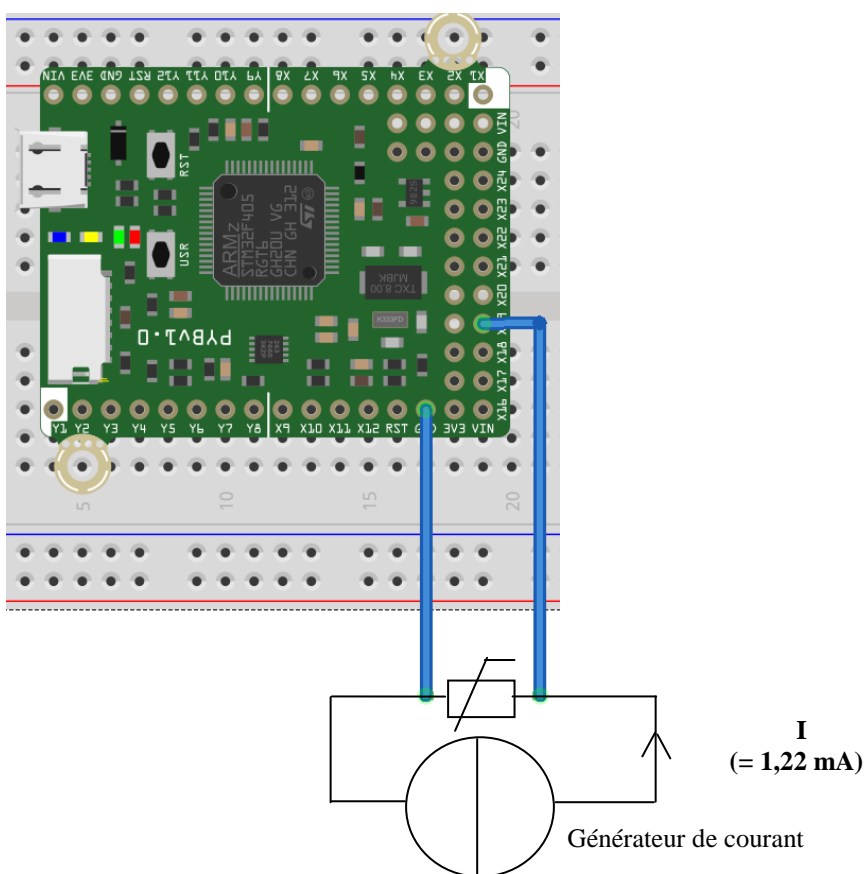
- 8) Sachant que ce conducteur suit la loi d'Ohm à toute température T , proposer une suite d'opérations à réaliser avec le programme Python pour passer de la tension U_{pt} à l'affichage de T en vous appuyant sur le résultats de la question 7.
- 9) Nous allons mesurer cette tension U_{pt} avec le microcontrôleur. En réalité le microcontrôleur renvoie au programme python un nombre entier compris entre 0 et 4095 et que nous appellerons $NPX19$. Ce nombre correspond à la tension U_{pt} entre la pin 1 et la masse GND.
Comment peut-on passer du nombre $NPX19$ à la tension U_{pt} sachant que le nombre maximal 4095 correspond à une tension de 3,3 V ?

REALISER

10 – Ouvrir le logiciel « **uPyCraft** »

11 – En utilisant les résultats des questions 8 et 9, compléter le programme « Conducteur Pt1000 microbit.py » pour qu'il puisse afficher la température de la salle de classe T_{pt} avec un chiffre après la virgule


12 – La carte n'étant pas connectée à l'ordinateur, réaliser le circuit ci-dessous.



13 – Connecter la carte Pyboard à l'ordinateur grâce au câble joint.

14 – Cliquer sur l'onglet  (le logiciel recherche la carte et le port utilisé automatiquement)

Les trois chevrons >>> doivent apparaitre en bas de l'écran

15 – Lancer le programme dans le microcontrôleur avec la touche 

commande  pour l'arrêter.

Pour aller plus loin

16 -

* On peut améliorer le programme pour qu'il puisse mesurer 20 fois la température avec un intervalle de temps de 1 s entre 2 mesures.

Aide : boucle « for » dont la syntaxe est :

```
for i in range(i début,i fin) : # boucle qui commence à i début et se termine à i fin -1
    action 1                   # listes des actions à effectuer pendant la boucle
    action 2
    ....
```

commande « pyb.delay(x) » # commande qui laisse un intervalle de temps de x millisecondes

* on peut allumer une diode si la température dépasse une certaine valeur...

remarque : on peut alimenter le générateur de courant directement avec le microcontrôleur ; dans ce cas le courant de sortie sera plus faible.

