

DOSSIER PEDAGOGIQUE

Activité SIN1 - Chaîne d'acquisition sur jauge de contrainte

(Eléments de correction)



Problématique : Mettre en œuvre une expérimentation pour qualifier la tension dans le câble qui permet d'amarrer la génératrice au sol.



Extrait du programme



Banc d'essai du KiweeOne

Supports utilisés Activité SIN1



Capteur de force CZL301C.pdf



**Document technique du
KiweeOne**

Activité SIN1 - Chaîne d'acquisition sur jauge de contrainte		Première spécialité STI2D
Descriptif de l'activité :		2 x 2h
Objectifs pédagogiques : A travers l'analyse du système réel et par une expérimentation, qualifier la tension dans le câble de maintien de la génératrice.		
Problématique : Mettre en œuvre une expérimentation pour qualifier la tension dans le câble qui permet d'amarrer la génératrice au sol.		
Résumé des activités : 1 ^{ère} Partie : Analyse du système réel, positionnement du capteur de force. 2 ^{ème} Partie : Justification du choix du capteur. 3 ^{ème} Partie : Expérimenter la tension dans le câble à l'aide du capteur et le calibrer.		
Matériel Nécessaire : <ul style="list-style-type: none">• Banc de mesures Kitewinder - KW10• Poste informatique• Document technique du KiweeOne	Environnement logiciel : <ul style="list-style-type: none">• Interface Labview• Tableur (Excel)• Programmation (Spyder)	
Connaissances associées : <ul style="list-style-type: none">• CO3.1. Identifier et caractériser les fonctions et les constituants d'un produit ainsi que ses entrées/sorties• CO3.2. Identifier et caractériser l'agencement matériel et/ou logiciel d'un produit• CO6.1. Expliquer des éléments d'une modélisation multiphysique proposée relative au comportement de tout ou partie d'un système• CO7.2. Mettre en œuvre un scénario de validation devant intégrer un protocole d'essais, de mesures et/ou d'observations sur le prototype ou la maquette, interpréter les résultats et qualifier le produit	Compétences développées : <ul style="list-style-type: none">• 3.2.2. Concept d'équilibre• 4.3.5. Conception informationnelle des produits• 5.2.3. Transmetteurs des mouvements• 5.3.1. Capteurs, conditionneurs• 6.2. Expérimentations et essais	

1^{ère} Partie : Analyse du système réel, positionnement du capteur de force.

L'étude s'appuie sur le support *KiweeOne* (Figure 1) de la société *KITEWINDER*, une petite éolienne aéroportée permettant de produire localement de l'énergie électrique à hauteur de 100W, en allant chercher les vents constants jusqu'à 120 mètres d'altitude.

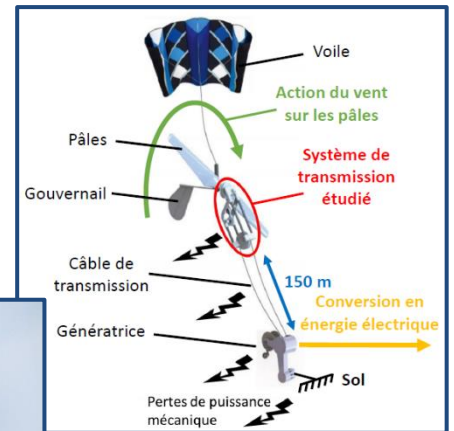


Figure 2 : Fonctionnement du KiweeOne

Figure 1 : Photo du KiweeOne

La lecture du dossier technique vous donnera plus d'informations sur le fonctionnement de l'éolienne aéroportée.

1.1. Présentation du KiweeOne

Le *KiweeOne* présente différents groupes de composants répondant à des exigences fonctionnelles spécifique. Dans cette étude, nous nous intéressons à la transmission de puissance et aux rendements associés. En particulier on se limitera à l'étude de la transmission mécanique entre l'axe de l'hélice et l'axe du câble de transmission de puissance jusqu'au sol (Figure 2). Une étude de l'environnement du *KiweeOne* est nécessaire pour définir les exigences associées à sa conception technique (Figure 3).

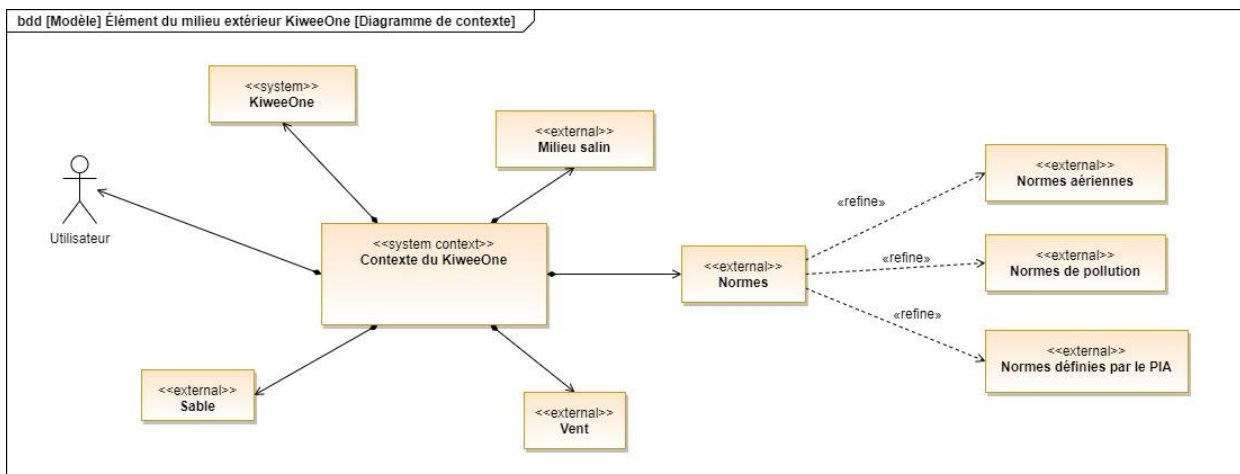


Figure 3 : Diagramme de contexte

1.2. Analyse du capteur d'effort

Un capteur de force est un transmetteur qui transforme une force physique (un poids, une tension, une compression ou une pression) en un signal électrique proportionnel qui peut être mesuré, converti et analysé. Lorsque la force appliquée au capteur augmente, le signal électrique change proportionnellement.

La technologie utilisée ici est un capteur à jauge de contrainte. Le fonctionnement et les caractéristiques du capteur de force sont décrites dans le document « **Capteur de force CZL301C.pdf** ».

La valeur brute délivrée par la jauge est codifiée en 12 bits.

Question 1 : En tenant compte des données fournies dans les caractéristiques du capteur, **quelle plage** de tension doit être utilisée ?

La plage de tension utilisée est comprise entre 0 et 12 Vcc.

Question 2 : **A quoi correspond** le signal de sortie ?

Il correspond à la valeur de la tension de sortie en fonction de la tension d'alimentation.

Question 3 : **Comment est réalisée** le calibrage du capteur avant son utilisation ?

Capteur alimenté en tension et sans charge appliquée, il faut relever la tension de sortie.
En appliquant une charge connue, rechercher le gain K tel que :

$$K = \frac{\text{Force}}{\text{Offset}}$$

Ce gain est propre à chaque capteur.

2^{ème} Partie : Justification du choix du capteur.

Le capteur est positionné entre la génératrice et le châssis du banc d'essai.

Question 4 : **Quel est** son rôle ?

Il permet de mesurer la charge que subit l'ensemble sous la tension générée par le vent sur la voile et la motrice.

Nous utiliserons, dans cette partie, le document ressource du capteur « **Capteur de force CZL301C.pdf** » et la notice d'utilisation « **Notice Utilisation KiweeOne (04.2021).pdf** »

Question 5 : **Quelles** sont les limites d'utilisation du KiweeOne d'un point de vue de la sécurité ?

La vitesse du vent ne peut excéder 25 km.h⁻¹.

La tension maximale dans le câble ne doit pas dépasser 20 kg.

Question 6 : **Pour quelle raison** est-il conseillé d'utiliser un support d'attache au moins 2 fois plus résistant que la force exercée par le cerf-volant ?

Avec un vent de 90 km.h⁻¹, en rafale, la charge sur le câble peut atteindre 20 kg. Avec les phénomènes de rafale de vent, des inerties importantes peuvent accentuer cette charge. Il est donc fortement conseiller de doubler la masse de soutien pour éviter le décrochage de l'ensemble.

Un extrait du programme de ce banc est proposé ci-dessous :

```
1  #include "stm32f10x.h"
2  #include "def.h"
3  #include "param.h"
4  #include "gpio.h"
5  #include "adc.h"
6
7  // Affiche ou non le debug dans ce source
8  #define _DEBUG_
9  #include "debug.h"
10 // modifier les valeurs de résistances sur le pont diviseur R42 / R155
11 // R42 = 1800 Ohms / R155 = 3000 Ohms
12
13 static u32 load;
14
15 #define A_DIVISOR      100000
16 #define B_DIVISOR      10000
17 #define C_DIVISOR      100
18
19 u32 _loadCellGrams(void)
20 {
21 return load;
22 }
23 void _loadCellRun ( void )
24 {
25 u32 tempLoad;
26 u32 adcValue=_adcValueGet(ADC_INPUT_ANA_V_LOADCELL);
27 tempLoad = (adcValue*adcValue*paramTable.a/A_DIVISOR)+(adcValue*paramTable.b/B_DIVISOR)+(paramTable.c/C_DIVISOR);
28 if (tempLoad>99999)
29 load=99999;
30 else
31 load =tempLoad;
32 }
```

Question 7 : A quelle ligne est récupérée l'information de valeur délivrée par le capteur ?

La valeur délivrée par le capteur est donnée en valeur brute ligne 26 :

```
26 u32 adcValue=_adcValueGet(ADC_INPUT_ANA_V_LOADCELL);
```

Question 8 : Sous quelle forme est la valeur délivrée ? Quel est sa valeur maximale ?

La valeur est une valeur décimale comprise entre 0 et 4096.

Question 9 : A partir de quelle ligne est établie l'équation polynomiale ? **Extraire** cette équation du programme et **détailler** les valeurs fournies.

L'équation est établie ligne 27 :

```
27 tempLoad = (adcValue*adcValue*paramTable.a/A_DIVISOR)+(adcValue*paramTable.b/B_DIVISOR)+(paramTable.c/C_DIVISOR);
```

$$\text{Lecture} = \frac{\text{Valeur brute}^2 \cdot a}{A_DIVISOR} + \frac{\text{Valeur brute} \cdot b}{B_DIVISOR} + \frac{c}{C_DIVISOR}$$

Avec $A_{DIVISOR} = 100\,000$, $B_{DIVISOR} = 10\,000$ et $C_{DIVISOR} = 100$

Question 10 : Le capteur **a-t-il été** bien choisi ?

0 à 12 V pour 0 à 4096 valeurs : la résolution est donc $\frac{12}{3 \cdot 10^{-3}} = 4\,000$ valeurs mesurables.


La charge maximale est de 100 kg.

Le capteur correspond à ce qui était recherché.

3ème Partie : Expérimenter la tension dans le câble à l'aide du capteur et le calibrer.

3.1. Mise en route du système



1. Avant toute mise en route du système, il faut que l'**interrupteur sectionneur** soit **ouvert** :
2. Dans un premier temps, la **présence du professeur est obligatoire** pour respecter les consignes.
3. Il faut **vérifier** que le fil de liaison **entre la tête et la génératrice** soit **tendu**.
4. **Raccorder** le cordon **USB** sur la prise « **liaison carte banc** » de l'armoire du système KW10 **au PC**.
5. **Raccorder** le cordon **USB** sur la prise « **liaison carte génératrice** » de l'armoire du système KW10 **au PC**.
6. **Mettre** le système **sous tension** : 
7. **Lancer l'exécutable LabView**.
8. À partir de l'onglet « Réglages », **attribuer les ports COM** préalablement identifiés de la **motrice** et de la **génératrice**. **Valider**
9. À partir de l'onglet « Motrice », **saisir** une consigne de vent de $6,5 \text{ m}\cdot\text{s}^{-1}$ et **valider**.
10. Sur l'armoire électrique, **maintenir** le bouton « Arrêt/Marche Génératrice » enfoncé durant 2 secondes minimales. (La génératrice va lancer la motrice qui va prendre alors le relais pour simuler l'hélice au vent)
11. En fonction de la charge dans la batterie, la génératrice va influencer sur la motrice et gérer la charge de la batterie afin d'optimiser la charge en fonction de la consigne vent.
12. Pour arrêter le système, **saisir** une consigne de vent nulle et **valider**. Lorsque la motrice ne tourne plus, **positionner l'interrupteur sectionneur** sur « **OFF** ».

3.2. Exportation des données

Nous allons réaliser une expérimentation sur le banc d'essai. Le banc d'essai doit être lancé ainsi que l'interface LabVIEW.

Le cahier des charges permet de fixer les limites d'utilisation du KiweeOne :

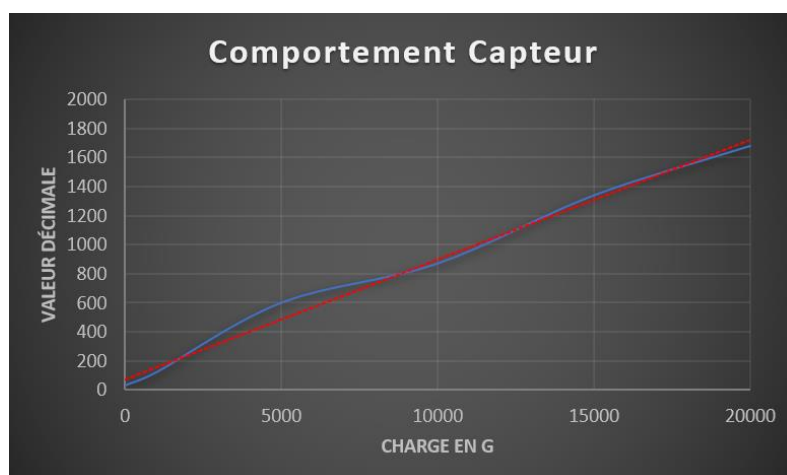
- 4kg pour un vent de 5,6 m/s
- 20kg pour un vent de 25 m/s

Question 11 : Relever sur l'interface les valeurs correspondants à des charges allant de 0 à 20 kg.

Charge en g	Valeur relevée en décimale	Tension associée en V
0	30	0
1 000	120	0,22
5 000	600	1,09
10 000	870	2,17
15 000	1340	3,62
20 000	1680	4,35

Question 12 : Que peut-on dire du comportement du capteur ?

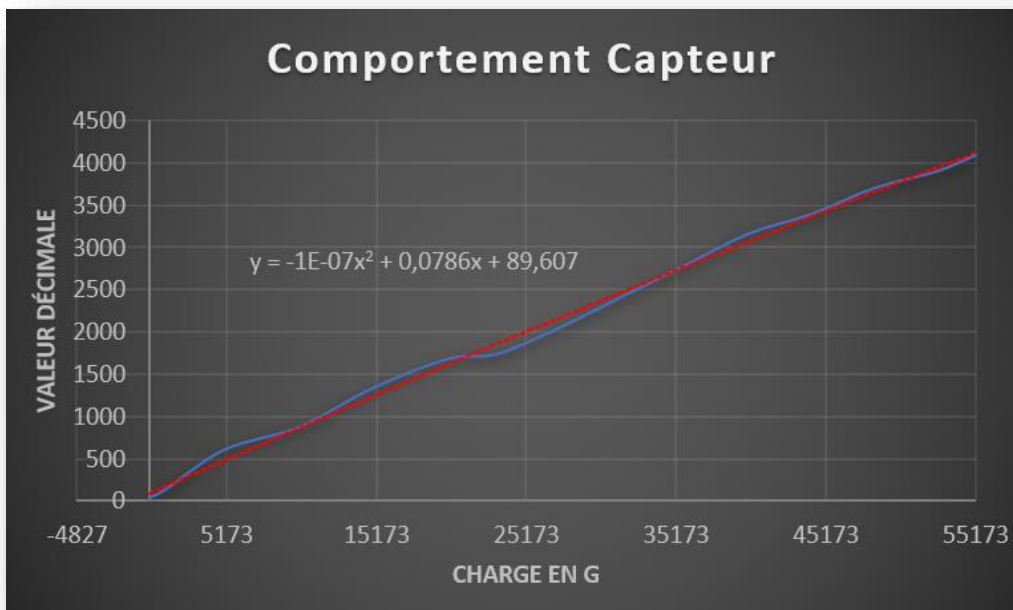
La fonction établie est une fonction linéaire



Question 13 : Quel peut-être la valeur maximale de la charge supportée par le capteur ? Pour cela établir un tableau sous Excel permettant de générer par extension des valeurs allant jusqu'à une valeur décimale admise par le capteur de 4096.

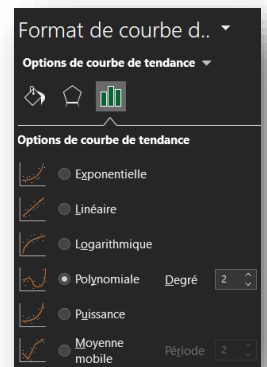
Nous pouvons observer sur ce graphique que la charge maximale peut atteindre près de 55 kg

Charge	Valeur décimale	Tension V
0	30	0
1 000	120	0,22
5 000	600	1,09
10 000	870	2,17
15 000	1 340	3,62
20 000	1 680	4,35
23 200	1 736	5,05
27 400	2 050	5,96
31 600	2 421	6,87
35 800	2 792	7,79
40 000	3 163	8,7
44 200	3 396	9,61
48 400	3 705	10,53
52 600	3 905	11,44
55 173	4 096	12



Question 14 : Extraire l'équation polynomiale correspondant en utilisant la fonction « Ajouter un élément graphique » d'excel.

$$y = -1.10^{-7} x^2 + 0,0786 x + 89,607$$



Question 15 : Nous pouvons donc calibrer le capteur sur l'interface LabView. **Donner** les valeurs retenues **a**, **b** et **c**.

$$a = -1.10^{-7}$$

$$b = 0,0786$$

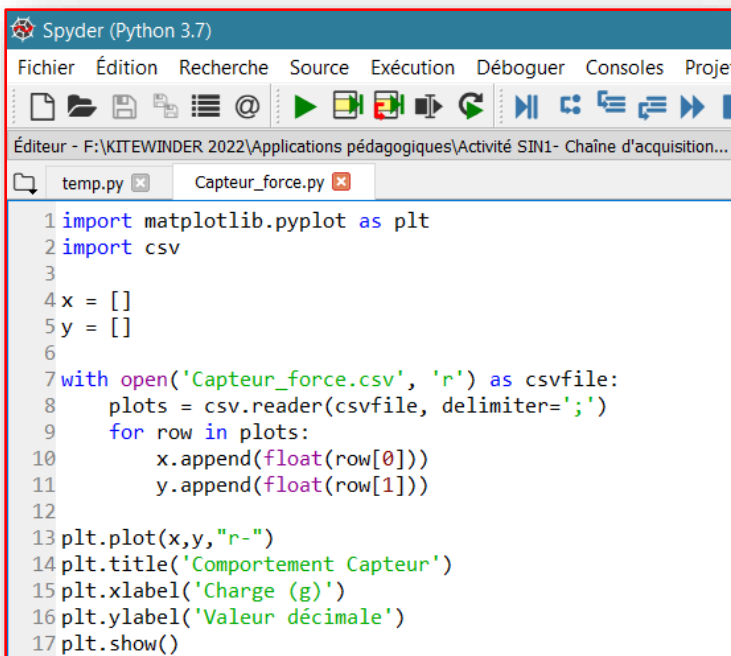
$$c = 89,607$$

Question 16 : Nous allons vérifier le bon fonctionnement du banc.

1. **Mettre** en route le banc d'essai.
2. Dans un premier temps nous allons mettre une charge de 1 kg.
3. Une fois la charge stabilisée sur l'interface, **appliquer** une force supplémentaire et progressive sur le câble.
4. Extraire le fichier de données du capteur sous le format « .csv » et renommer le fichier « Capteur_force.csv ».

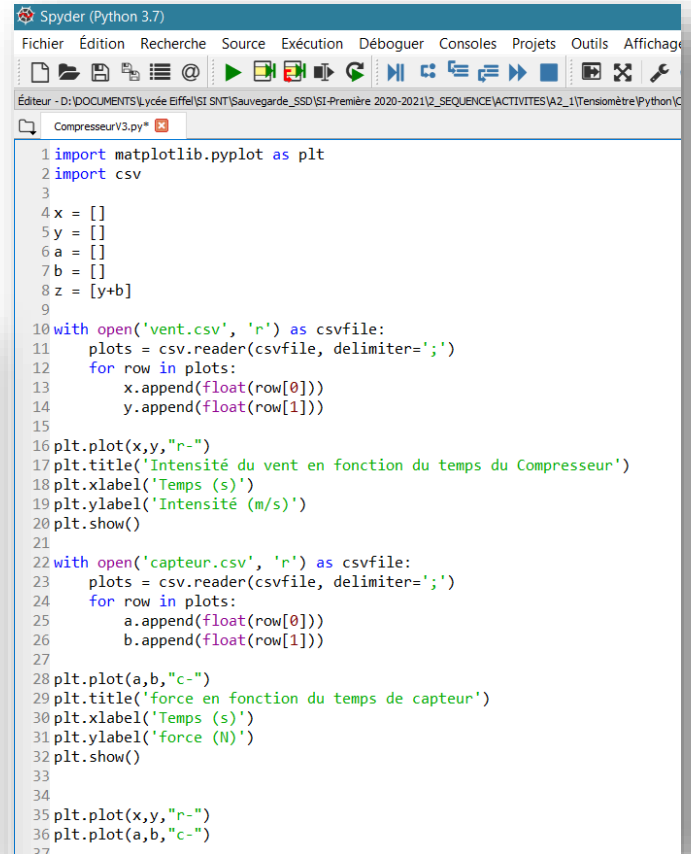
Nous souhaitons afficher les données ainsi générées au format CSV en utilisant une programmation en Python.

Question 17 : A partir du fichier en CSV **compléter** le programme d'affichage (logiciel Spyder) en vous appuyant sur l'exemple ci-contre. Nous souhaitons afficher le graphique de la valeur décimale relevée en fonction de la charge.



```
1 import matplotlib.pyplot as plt
2 import csv
3
4 x = []
5 y = []
6
7 with open('Capteur_force.csv', 'r') as csvfile:
8     plots = csv.reader(csvfile, delimiter=',')
9     for row in plots:
10         x.append(float(row[0]))
11         y.append(float(row[1]))
12
13 plt.plot(x,y,"r-")
14 plt.title('Comportement Capteur')
15 plt.xlabel('Charge (g)')
16 plt.ylabel('Valeur décimale')
17 plt.show()
```

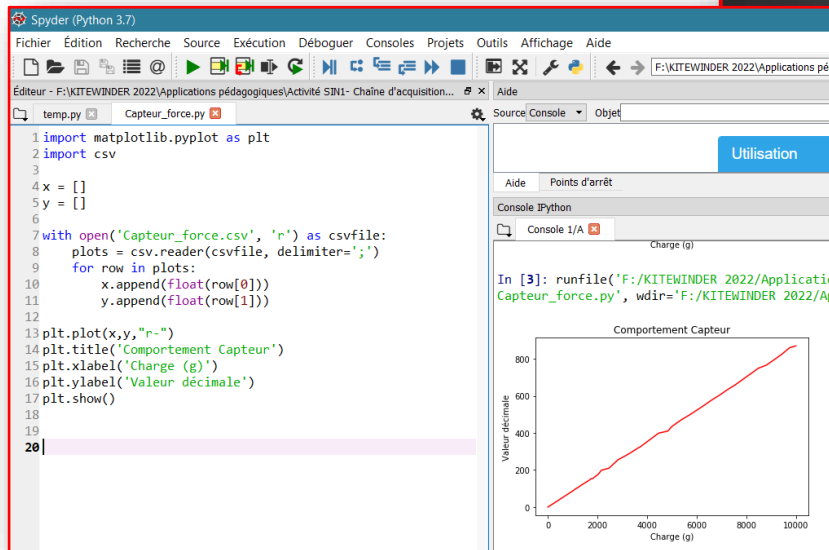
Programme proposé



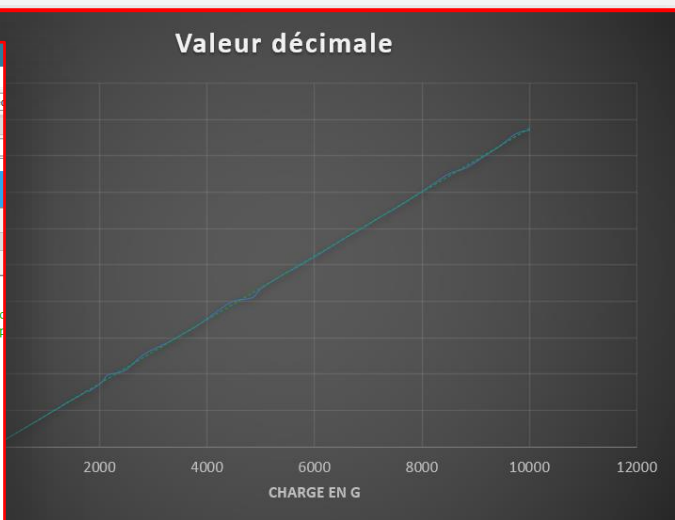
```
1 import matplotlib.pyplot as plt
2 import csv
3
4 x = []
5 y = []
6 a = []
7 b = []
8 z = [y+b]
9
10 with open('vent.csv', 'r') as csvfile:
11     plots = csv.reader(csvfile, delimiter=',')
12     for row in plots:
13         x.append(float(row[0]))
14         y.append(float(row[1]))
15
16 plt.plot(x,y,"r-")
17 plt.title('Intensité du vent en fonction du temps du Compresseur')
18 plt.xlabel('Temps (s)')
19 plt.ylabel('Intensité (m/s)')
20 plt.show()
21
22 with open('capteur.csv', 'r') as csvfile:
23     plots = csv.reader(csvfile, delimiter=',')
24     for row in plots:
25         a.append(float(row[0]))
26         b.append(float(row[1]))
27
28 plt.plot(a,b,"c-")
29 plt.title('force en fonction du temps de capteur')
30 plt.xlabel('Temps (s)')
31 plt.ylabel('force (N)')
32 plt.show()
33
34
35 plt.plot(x,y,"r-")
36 plt.plot(a,b,"c-")
37
```

Exemple de programmation en Python

Question 18 : Réaliser la saisie d'écran du graphique ainsi réalisé et le comparer avec celui que génère Excel.



Graphique sur Spyder



Graphique sur Excel

Question 19 : Quel est l'intérêt que nous avons à utiliser une programmation en python ,

La génération des données est sous format « .csv », c'est le format utilisé par nos applications sur smartphone pour extraire les données générées par les capteurs intégrés et les afficher sous forme graphique.