

Caractéristique d'un conducteur ohmique

Loi d'Ohm

1- Documents

Document 1 - Caractéristique tension intensité

Afin de définir les propriétés électriques d'un dipôle, il est nécessaire d'étudier sa caractéristique tension-intensité $U=f(I)$.

Cette caractéristique est la courbe représentant la tension U entre les bornes du dipôle étudié en fonction de l'intensité I du courant qui le traverse.

Document 2 - La loi d'Ohm

La loi d'Ohm est une loi physique empirique qui lie l'intensité du courant I traversant un dipôle électrique à la tension U à ses bornes.

La loi d'Ohm établit que la tension est proportionnelle à l'intensité:

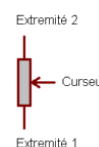
$$U = R \times I$$

- U : Tension aux bornes de la résistance en volt (V)
- I : Intensité du courant qui circule à travers la résistance en ampère (A)
- R : Résistance en Ohm (Ω)

Un dipôle électrique vérifiant la loi d'Ohm est appelé un conducteur ohmique.

Document 3 - La résistance variable

Un potentiomètre est une résistance variable ou rhéostat à trois bornes, dont une est reliée à un curseur qui peut se déplacer: on peut donc faire varier la valeur ohmique entre deux points, par simple action mécanique sur un axe rotatif ou rectiligne.



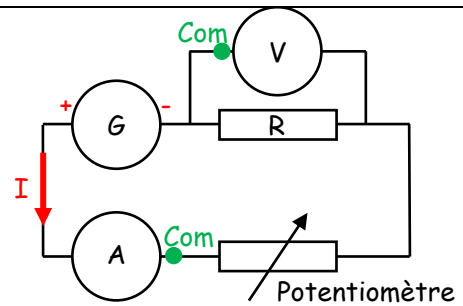
Document 4 - Matériel

- Générateur de tension continue
- Plaque pour composants
- Résistances de $220\ \Omega$, $330\ \Omega$ et $470\ \Omega$
- Potentiomètre de $1\ k\Omega$
- Multimètres (Voltmètre et Ampèremètre)
- Cordons électriques
- Ordinateur avec Spyder
- Programme "Loi d'Ohm.py" ([http://www.prof-tc.fr/Lycees/file/Python/Loi Ohm.zip](http://www.prof-tc.fr/Lycees/file/Python/Loi%20Ohm.zip))

Document 5

Montage expérimental - Pont diviseur de tension

Le pont diviseur de tension est un montage électronique simple qui permet de diviser une tension d'entrée, constitué par exemple de deux résistances en série.



2- Python

Python est un langage de programmation interprété (les instructions que vous lui envoyez sont "transcrites" en langage machine au fur et à mesure de leur lecture), à ne pas confondre avec un langage compilé, (où avant de pouvoir les exécuter, le logiciel se charge de transformer le code du programme en langage machine).

Il est possible d'associer des bibliothèques à Python afin d'étendre ses possibilités

Il est portable, c'est à dire qu'il peut fonctionner sous différents systèmes d'exploitation (Windows, Linux, Mac OS X,...).

Document 6 - Graphe des points avec une régression linéaire en Python

Le programme à utiliser se nomme: "Loi Ohm.py"

```
# Prof-TC
# Decembre 2020
# Loi d'Ohm
# Tracé de U=f(I) pour une résistance

#Affichage du nom du programme
print("_____")
print("loi d'Ohm")
print("_____")

#Importation des bibliothèques
import matplotlib.pyplot as plt
import numpy as np

#Valeurs expérimentales à modifier selon l'expérience (U en volt et I en ampère)
print("*****")
I=[0, 0.0068, 0.0082, 0.0095, 0.0109, 0.0123, 0.0136, 0.0150, 0.0164, 0.0177,
0.0191, 0.0205, 0.0218, 0.0232, 0.0245, 0.0259, 0.0273]
U=[0, 1.5, 1.8, 2.1, 2.4, 2.7, 3, 3.3, 3.6, 3.9, 4.2, 4.5, 4.8, 5.1, 5.4, 5.7, 6]
print("*****")

#Affichage des valeurs expérimentales et du nombre de points
print("valeurs de I: ",I)
print("valeurs de U: ",U)
print("Il y a ",len(I)," couples de points")
print("*****")

#Initialisation des variables
Imoy=0.0
Umay=0.0
N=0.0
D=0.0
a=0.0
b=0.0
Imin=I[0]
Umin=U[0]
Imax=I[len(I)-1]
Umax=U[len(U)-1]

# Initialisation de la figure
fig, ax = plt.subplots()

#Calcul des coefficients de la droite représentant U=f(I)
for i in range (len(I)):
    Imoy=Imoy+I[i]
    Umay=Umay+U[i]
    N=N+(I[i]-Imoy)*(U[i]-Umay)
    D=D+(I[i]-Imoy)*(I[i]-Imoy)

a=N/D #Coefficient directeur de la droite
R=a #Valeur de la résistance
b=Umay-a*Imoy #Ordonnée à l'origine de la droite

#Mise en écriture scientifique des valeurs de R et b pour l'affichage de l'équation
Rs=np.format_float_scientific(R, precision = 1, exp_digits=1)
bs=np.format_float_scientific(b, precision = 1, exp_digits=1)

#Tracé des axes, labels et titre du graphique
plt.axis([Imin, Imax, 0, Umax])
plt.xlabel('I (A)',color='green', fontsize=20)
plt.ylabel('U (V)',color='green', fontsize=20)
plt.title("Caractéristique d'un conducteur Ohmique - Loi d'Ohm", color='red', fontsize=10)
plt.grid()

#Tracé des valeurs expérimentales
plt.scatter(I,U,marker='o',color='r',linewidth = 4)

#Tracé de la droite
x=np.linspace(0,Imax+1,200)
y=R*x+b
plt.plot(x,y,linewidth = 2)

#Ecriture de l'équation de la droite
plt.text((Imax)/2, Umax-Umax*0.1, "U = "+str(Rs)+" x I + "+str(bs), color='green',
fontsize=10,horizontalalignment = 'center')

#Affichage du graphique
plt.show()

# Sauvegarde de la figure dans le dossier où se trouve le programme
fig.savefig("Loi d'Ohm")

print("*****")
print("L'équation de la droite est: U = ",Rs," x I + ",bs)
print("*****")

#Fin du programme
```

3- Expérience

- Réaliser le montage expérimental sur la plaque, en respectant la position des composants et les couleurs pour les fils.
- Faire vérifier le montage par le professeur.
- Allumer le générateur de tension et le régler sur une tension de 6V.
- Réaliser les mesures de la tension U et de l'intensité I du courant en faisant varier la résistance du potentiomètre.
- Noter les valeurs dans le tableau ci-dessous.

I (mA)												
U (V)												

4- Exploitation des résultats

- En suivant les instructions du professeur, utiliser le programme "Loi Ohm.py" afin de rentrer les valeurs.
- Exécuter le programme pour tracer la courbe et relever l'équation de la courbe.
- Imprimer la courbe et la coller ci-dessous.

- Les affirmations suivantes sont-elles justes ou fausses?

	Vrai	Faux
U dépend de I		
U est proportionnelle à I		
I diminue quand U augmente		
I dépend de U		
$U = a + b.I$ ($a \neq 0$)		
C'est le graphe d'une fonction affine		
Le rapport U/I est constant		
C'est le graphe d'une fonction linéaire		
$U_{AB} = k.I$ avec k un nombre constant		
U_{AB} est une droite		

- Quelle loi vient-on de retrouver?

.....

.....

.....

.....

- Ecrire la relation concernant cette loi en précisant les unités.

.....

.....

.....

.....

.....

.....

.....

.....