

Caractéristique électrique d'une pile

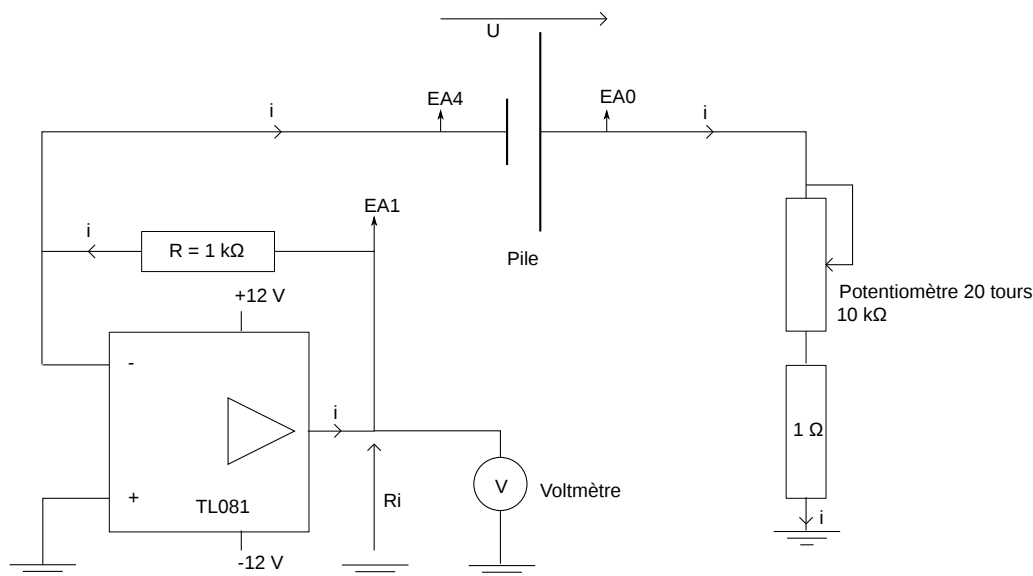
1. Introduction

Cette page montre des expériences de relevé de caractéristiques courant-tension de différentes piles. Pour les piles dont l'électrolyte liquide est accessible, on fait aussi un relevé des potentiels d'oxydoréduction des deux électrodes avec une électrode de référence (ECS), ce qui permet de tracer les courbes courant-potentiel de ces électrodes.

2. Circuits de mesure

2.a. Caractéristique courant-tension à faible courant

Le circuit suivant permet de faire des mesures pour des courants jusqu'à environ 10 mA .



L'amplificateur opérationnel maintient la borne moins de la pile à un potentiel nul. Le courant i débité par la pile passe dans la résistance R . La sortie de l'amplificateur est donc au potentiel Ri . Le voltmètre permet donc de mesurer le courant en milliampères. Le potentiomètre multitours permet de faire varier la résistance entre la borne plus de la pile et la masse, ce qui fait varier le courant.

La carte d'acquisition SysamSP5 est utilisée pour faire les mesures. L'amplificateur est alimenté avec l'alimentation double $-12/0/12\text{ V}$ de la carte. La tension U aux bornes de la pile est mesurée en différentiel entre les bornes EA0 et EA4. En principe, la borne EA4 reste à la masse mais on fait quand même une mesure différentielle car l'amplificateur peut introduire un léger décalage de cette tension par rapport à la masse. La tension Ri , qui donne accès au courant, est acquise sur la voie EA1.

Le script python suivant effectue les mesures point par point et enregistre le tableau dans un fichier. Chaque mesure est en fait une acquisition de 100 ms comportant 1000

échantillons. Pour faire une mesure, on doit appuyer sur entrée. Pour terminer, on doit entrer N ou n.

[courantTensionPile.py](#)

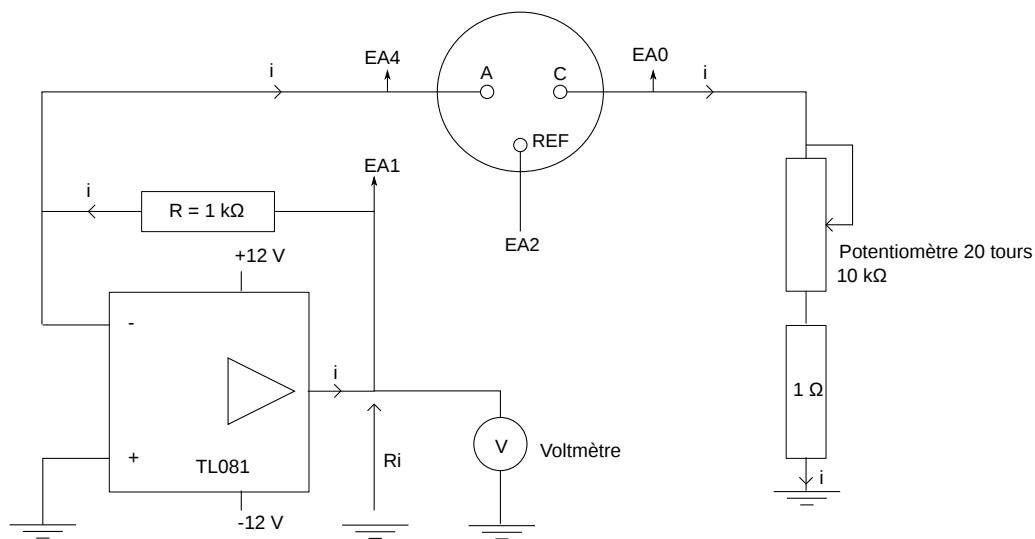
```
import pycan.main as pycan
import numpy

sys = pycan.Sysam("SP5")
sys.config_entrees([0,1],[2.0,10.0],[0])
fe = 10000.0
te = 1.0/fe
ne = 1000
sys.config_echantillon(te*1e6,ne)
R = 1000.0 # resistance du convertisseur courant tension
liste_I = []
liste_U = []
while True:
    r = raw_input("Continuer [O/N] ?")
    if r=="N" or r=="n":
        break
    sys.acquerir()
    u=sys.entrees()
    u0 = u[0]
    u1 = u[1]
    I = u1.mean()/R*1000
    U = u0.mean()
    liste_I.append(I)
    liste_U.append(U)
    print("U = %f, I (mA) = %f"%(U,I))
sys.fermer()
numpy.savetxt("alcaline-1.txt",[liste_U,liste_I])
```

Au début de l'expérience, on place le potentiomètre dans la position de plus grande résistance ($10\text{ k}\Omega$). On tourne la vis du potentiomètre pour réduire sa résistance avant de déclencher une mesure. Le courant est contrôlé avec le voltmètre.

2.b. Caractéristique courant-tension à faible courant avec mesure du potentiel d'oxydoréduction

Il s'agit d'obtenir la caractéristique courant-tension d'une pile dont l'électrolyte liquide est accessible. Une électrode référence (ECS) plongée dans l'électrolyte permet d'accéder en plus aux potentiels d'oxydoréduction des deux électrodes, et de tracer les courbes courant-potentiel correspondantes.



L'anode (A) est l'électrode sur laquelle une réaction d'oxydation a lieu. La cathode (C) est le siège d'une réduction. L'électrode de référence (REF) est reliée à l'entrée EA2 de la carte d'acquisition.

Voici le script python, analogue au précédent, mais enregistrant en plus le potentiel redox de l'anode.

[courantTensionPile2.py](#)

```
import pycan.main as pycan
import numpy

sys = pycan.Sysam("SP5")
sys.config_entrees([0,1,2],[2.0,10.0,2.0],[0])
fe = 10000.0
te = 1.0/fe
ne = 1000
sys.config_echantillon(te*1e6,ne)
R = 1000.0 # resistance du convertisseur courant tension
liste_I = []
liste_U = []
liste_EA = []
while True:
    r = raw_input("Continuer [O/N] ?")
    if r=="N" or r=="n":
        break
    sys.acquerir()
    u=sys.entrees()
    u0 = u[0]
    u1 = u[1]
    u2 = u[2]
    I = u1.mean()/R*1000
```

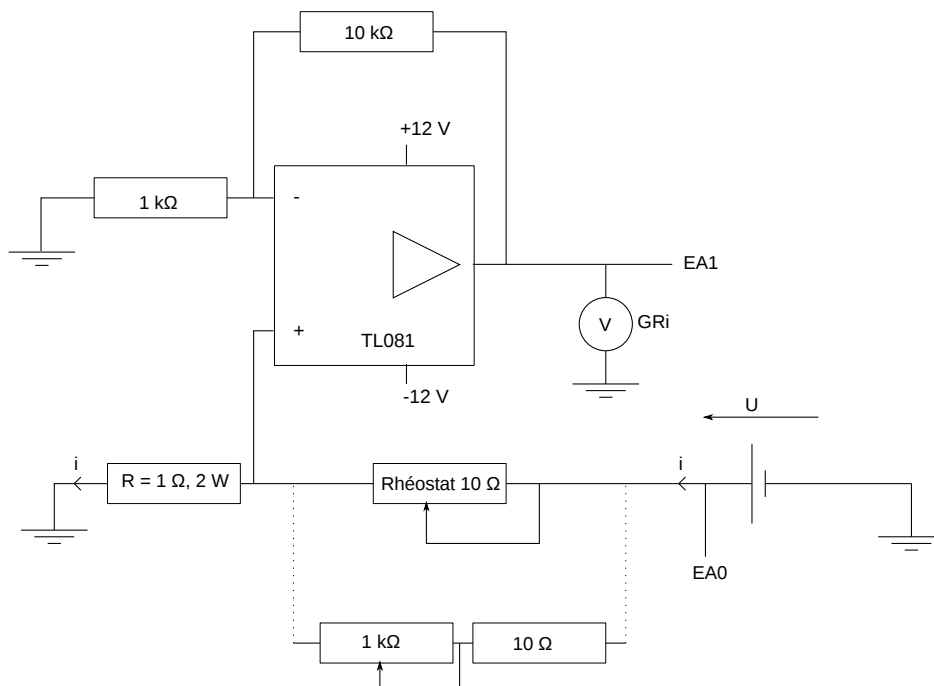
```

U = u0.mean()
EA = 0.25-u2.mean() # potentiel de l'anode
liste_I.append(I)
liste_U.append(U)
liste_EA.append(EA)
print("U (V) = %f, I (mA) = %f, E(A) (V) = %f"%(U,I,EA))
sys.fermer()
numpy.savetxt("pile.txt",[liste_U,liste_I,liste_EA])

```

2.c. Caractéristique courant-tension à fort courant

Le circuit suivant permet d'atteindre un courant d'intensité 1 A.



Le rhéostat permet de faire varier le courant débité par la pile. Ce courant traverse une résistance $R = 1\Omega$ qui doit supporter un courant de 1 A sous 1 V. L'amplificateur non inverseur multiplie la tension Ri par un gain $G = 11$. La tension aux bornes de la pile est lue sur l'entrée EA0 de la carte d'acquisition. La tension GRi est lue sur l'entrée EA1. Un voltmètre peut être utilisé pour lire cette tension lorsqu'on tourne le rhéostat.

Les fils de liaison du circuit de puissance parcouru par le courant i doivent être le plus court possible, en particulier les fils de liaison à la masse.

Afin de réduire le courant, le rhéostat est alterné avec un potentiomètre multitours de $1\text{ k}\Omega$. Il est alors possible d'abaisser le courant jusqu'à environ 1 mA , ce qui fait une tension lue sur la voie EA2 de 10 mV .

[courantTensionPile3.py](#)

```

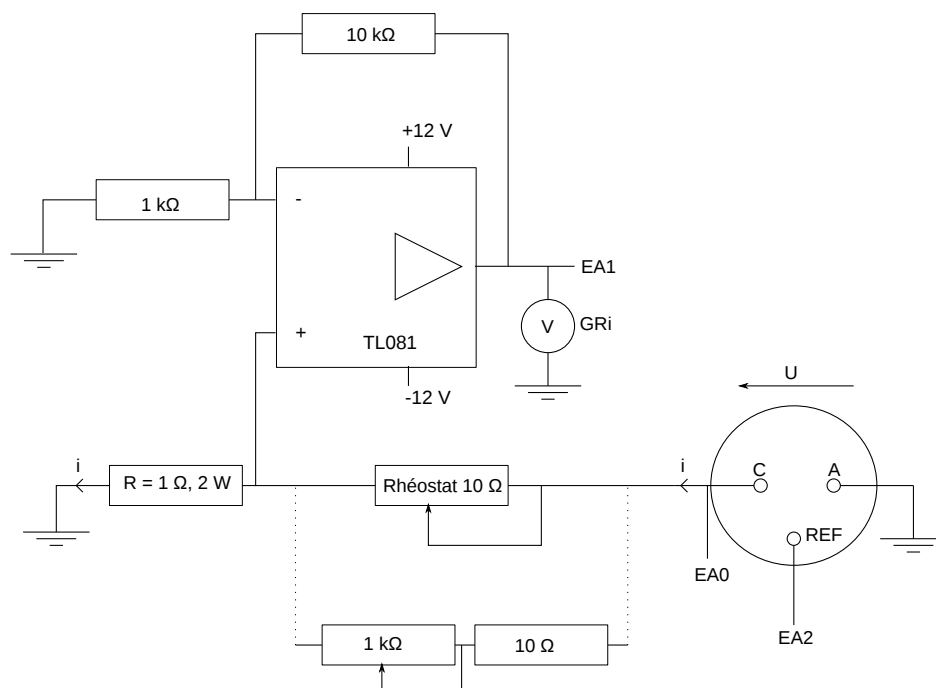
import pycan.main as pycan
import numpy

```

```
sys = pycan.Sysam("SP5")
sys.config_entrees([0,1],[10.0,10.0])
fe = 10000.0
te = 1.0/fe
ne = 1000
sys.config_echantillon(te*1e6,ne)
R = 1.1 # resistance du convertisseur courant tension
G = 11 # gain de l'ampli
liste_I = []
liste_U = []
while True:
    r = raw_input("Fin [O/N] ?")
    if r=="N" or r=="n":
        break
    sys.acquerir()
    u=sys.entrees()
    u0 = u[0]
    u1 = u[1]/G
    I = u1.mean()/R*1000
    U = u0.mean()
    liste_I.append(I)
    liste_U.append(U)
    print("U = %f, I (mA) = %f"%(U,I))
sys.fermer()
numpy.savetxt("pile.txt",[liste_U,liste_I])
```

2.d. Caractéristique courant-tension à fort courant avec mesure du potentiel d'oxydoréduction

Le circuit est identique au précédent, avec en plus la mesure du potentiel de l'électrode de référence sur la voie EA2. L'anode, sur laquelle l'oxydation a lieu, est reliée à la masse. On peut néanmoins inverser l'anode et la cathode, car l'amplificateur fonctionne en bipolaire.



Voici le script python, qui enregistre la tension, le courant, et le potentiel redox de l'anode.

[courantTensionPile4.py](#)

```
import pycan.main as pycan
import numpy

sys = pycan.Sysam("SP5")
sys.config_entrees([0,1,2],[10.0,1.0,5.0])
fe = 10000.0
te = 1.0/fe
ne = 1000
sys.config_echantillon(te*1e6,ne)
R = 1.1 # resistance du convertisseur courant tension
G = 11 # gain de l'ampli
liste_I = []
liste_U = []
liste_EA = []
while True:
    r = raw_input("Fin [O/N] ?")
    if r=="N" or r=="n":
        break
    sys.acquerir()
    u=sys.entrees()
    u0 = u[0]
    u1 = u[1]/G
    u2 = u[2]
```

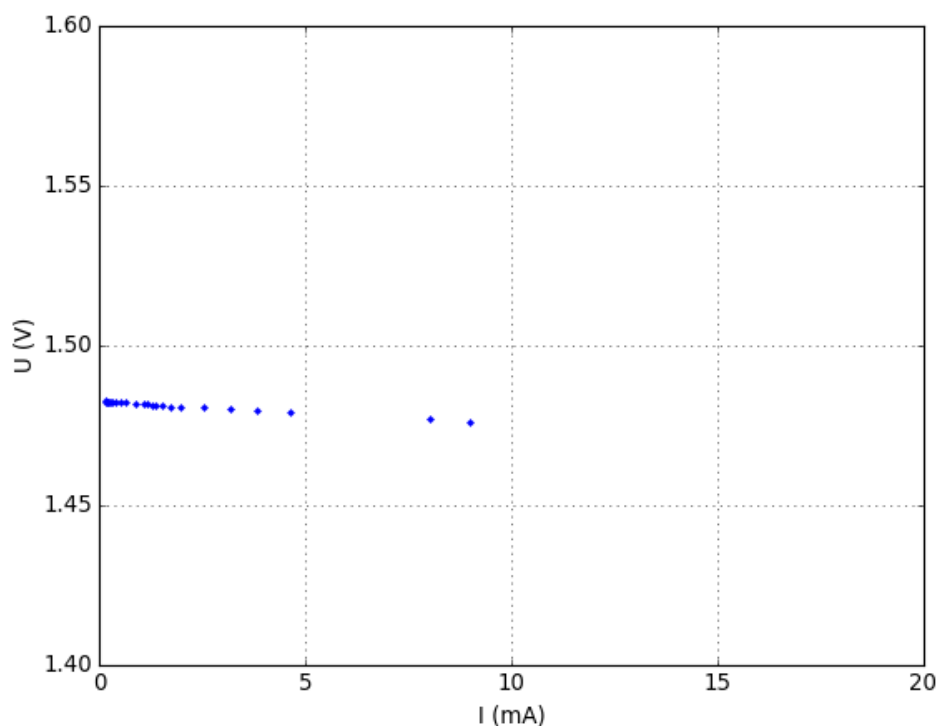
```
EA = 0.25-u2.mean()
I = u1.mean()/R*1000
U = u0.mean()
liste_I.append(I)
liste_U.append(U)
liste_EA.append(EA)
print("U = %f, I (mA) = %f, EA (V) = %f"%(U,I,EA))
sys.fomer()
numpy.savetxt("pile.txt",[liste_U,liste_I,liste_EA])
```

3. Pile alcaline

Voici les résultats obtenus en faible courant avec une pile commerciale AA zinc-oxyde de manganèse, appelée communément pile alcaline :

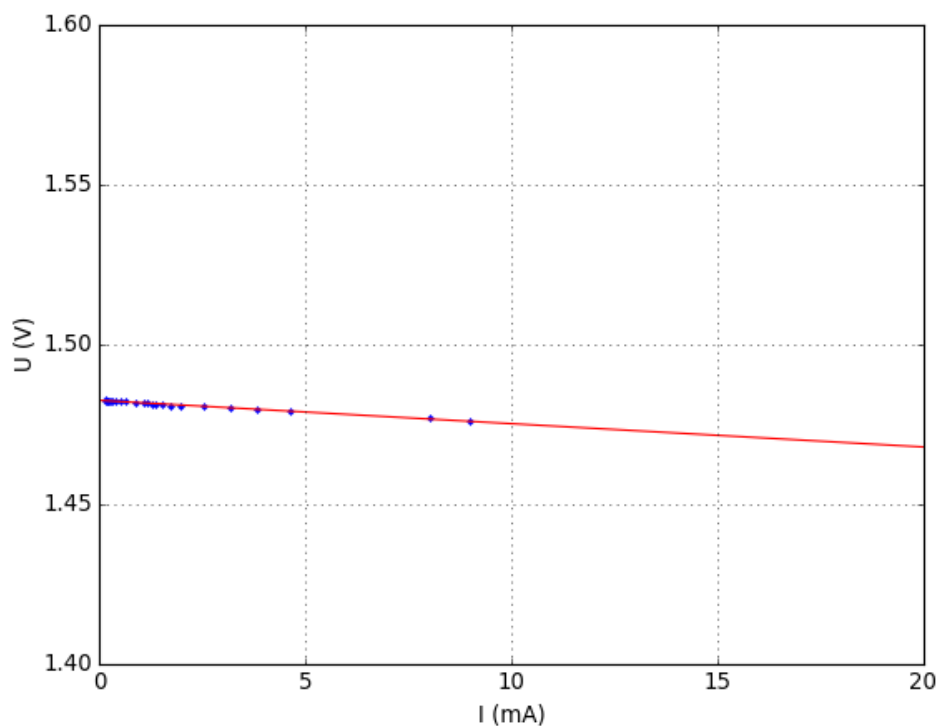
```
import numpy
from matplotlib.pyplot import *

[U,I] = numpy.loadtxt("alcaline-4.txt")
figure()
plot(I,U,'.')
xlabel("I (mA)")
ylabel("U (V)")
axis([0,20,1.4,1.6])
grid()
```



La caractéristique courant-tension semble linéaire. On fait une régression linéaire pour obtenir la force électromotrice et la résistance interne :

```
from scipy import stats
a,b,r_value,p_value,std_err = stats.linregress(I,U)
plot([0,-b/a],[b,0], 'r')
E = b
R = -a*1000
```



Force électromotrice :

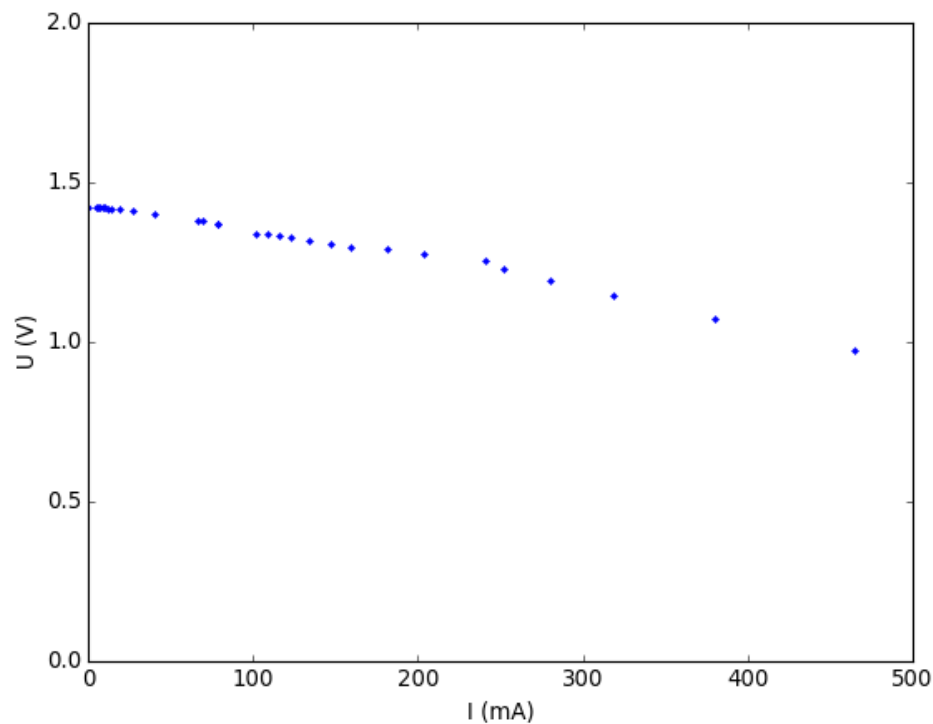
```
print(E)
--> 1.4825235383414135
```

Résistance interne :

```
print(R)
--> 0.7296086266714531
```

Voici les résultats en fort courant :

```
[U,I] = numpy.loadtxt("alkaline-power-4.txt")
figure()
plot(I,U, 'r.')
xlabel("I (mA)")
ylabel("U (V)")
axis([0,500,0,2.0])
```

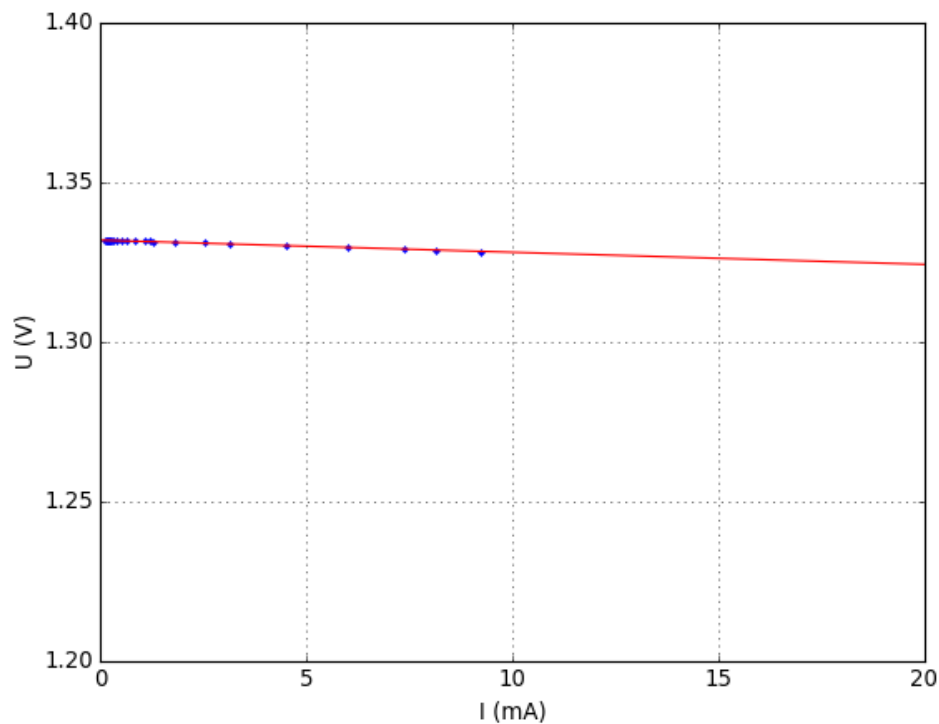



4. Accumulateur Ni-MH

Voici les résultat avec un accumulateur Nickel-hydrure métallique (format AA), rechargé depuis peu :

```
import numpy
from matplotlib.pyplot import *

[U,I] = numpy.loadtxt("NiMH-1.txt")
figure()
plot(I,U,'.')
xlabel("I (mA)")
ylabel("U (V)")
axis([0,20,1.2,1.4])
grid()
a,b,r_value,p_value,std_err = stats.linregress(I,U)
plot([0,-b/a],[b,0],'r')
E = b
R = -a*1000
```



Force électromotrice :

```
print(E)
--> 1.3319041635283797
```

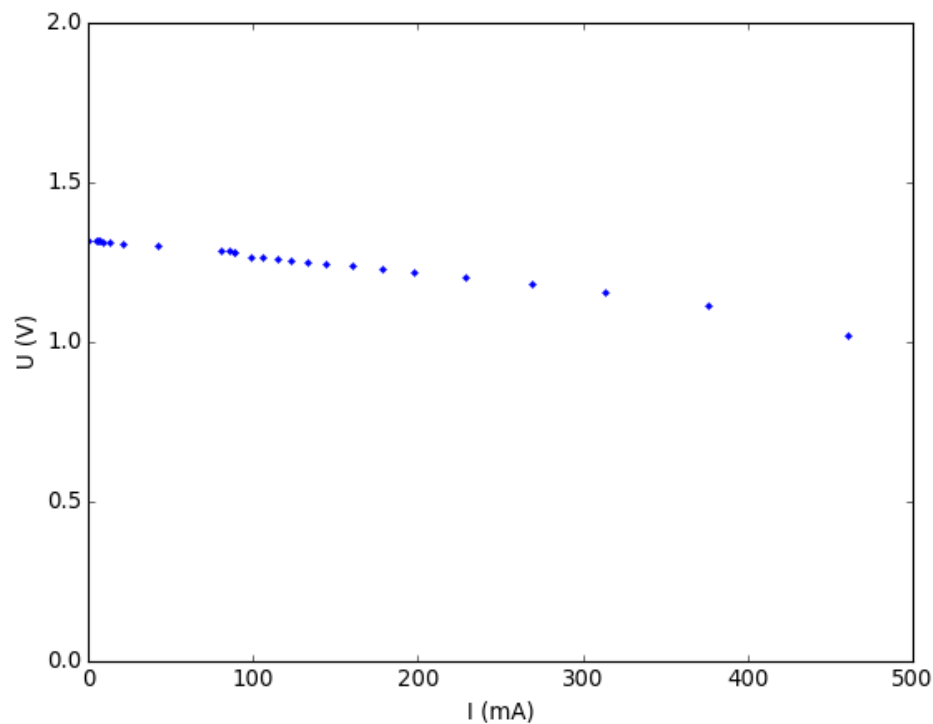
Résistance interne :

```
print(R)
--> 0.37909794498297456
```

La force électromotrice est nettement plus basse que pour la pile alcaline (la valeur spécifiée sur l'accumulateur est 1,2 V), mais la résistance interne est plus faible.

Voici les résultats en fort courant :

```
[U,I] = numpy.loadtxt("Ni-MH-power-3.txt")
figure()
plot(I,U,'.')
xlabel("I (mA)")
ylabel("U (V)")
axis([0,500,0,2.0])
```



5. Pile zinc-fer

5.a. Solution NaCl

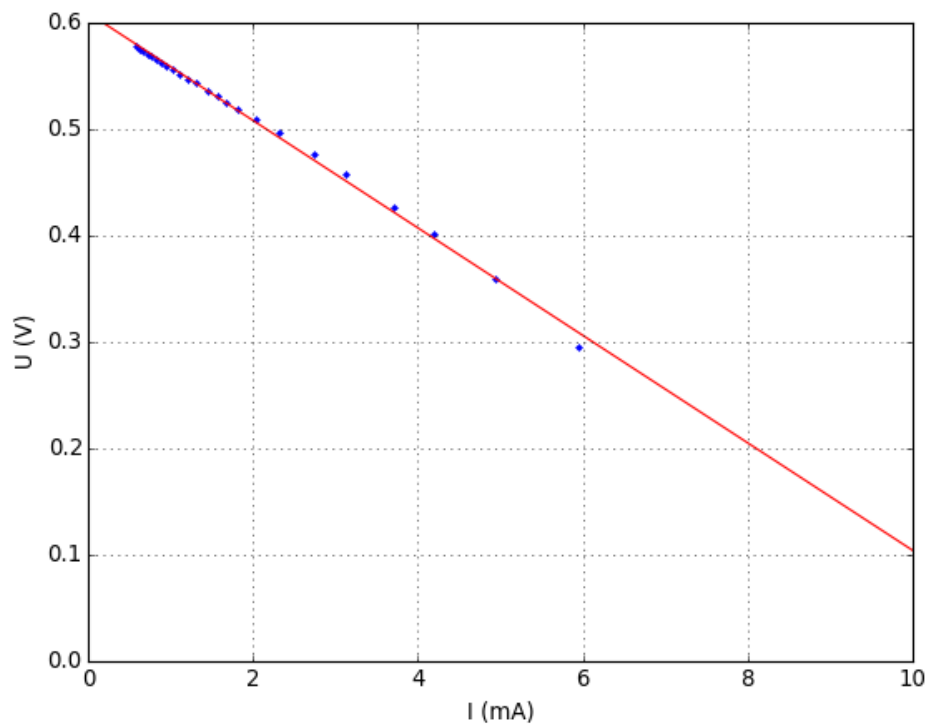
Cette pile n'est pas utilisée pour convertir l'énergie mais comme pile de corrosion, pour protéger le fer contre la corrosion. L'anode est l'électrode de zinc. L'électrolyte est une solution aqueuse de chlorure de sodium à 30 g/L (concentration de l'eau de mer).

L'expérience suivante est réalisée avec deux plaques de fer et de zinc d'environ 3 cm^2 par 10 cm de long, plongées dans 500 mL d'eau salée.

```
import numpy

from matplotlib.pyplot import *

figure()
[U,I,Ezn] = numpy.loadtxt("Fe-Zn-9.txt")
plot(I,U,'. ')
xlabel("I (mA)")
ylabel("U (V)")
grid()
axis([0,10,0,0.6])
a,b,r_value,p_value,std_err = stats.linregress(I,U)
plot([0,-b/a],[b,0], 'r')
E = b
R = -a*1000
```



Force électromotrice :

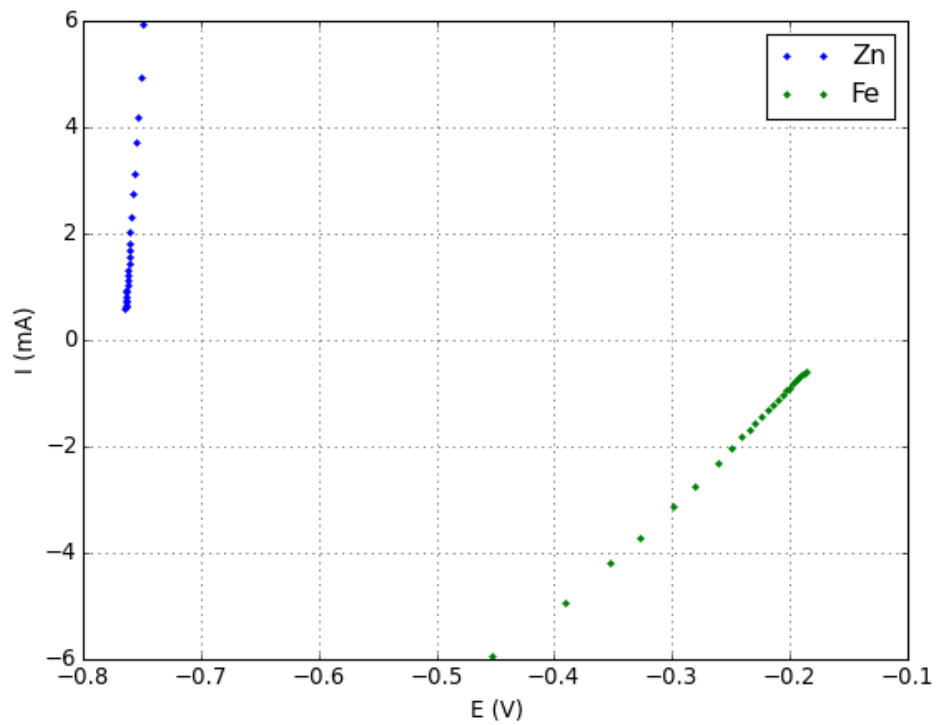
```
print(E)
--> 0.60932677133438373
```

Résistance interne :

```
print(R)
--> 50.512737959358375
```

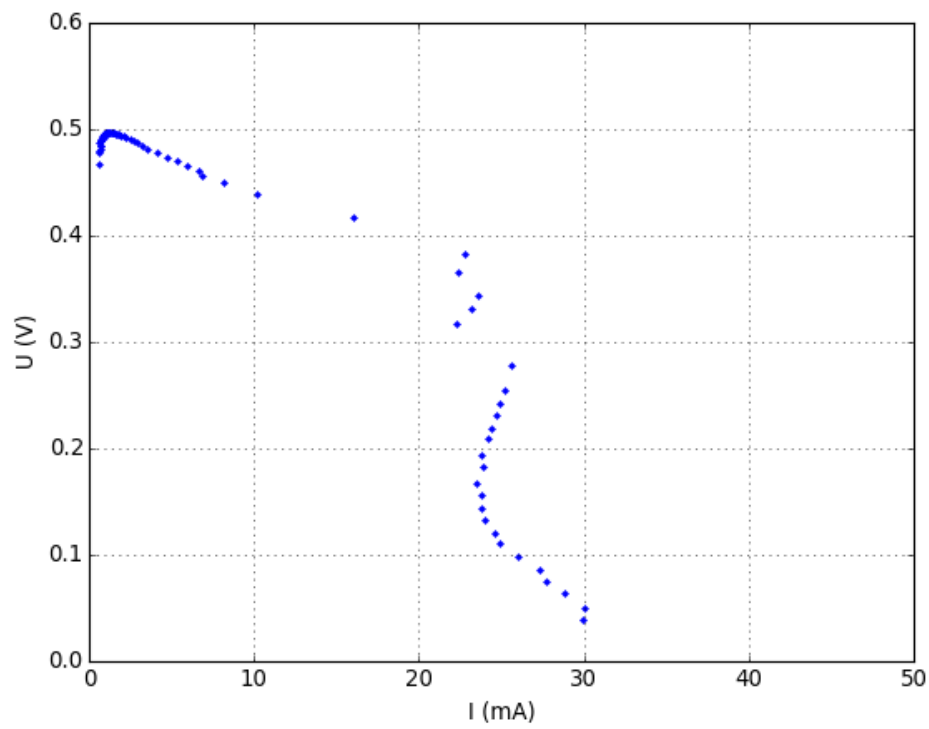
On trace les courbes courant-potentielle pour chaque électrode :

```
figure()
plot(Ezn, I, '.', label='Zn')
Efe = Ezn + U
plot(Efe, -I, '.', label='Fe')
xlabel("E (V)")
ylabel("I (mA)")
legend(loc='upper right')
grid()
```

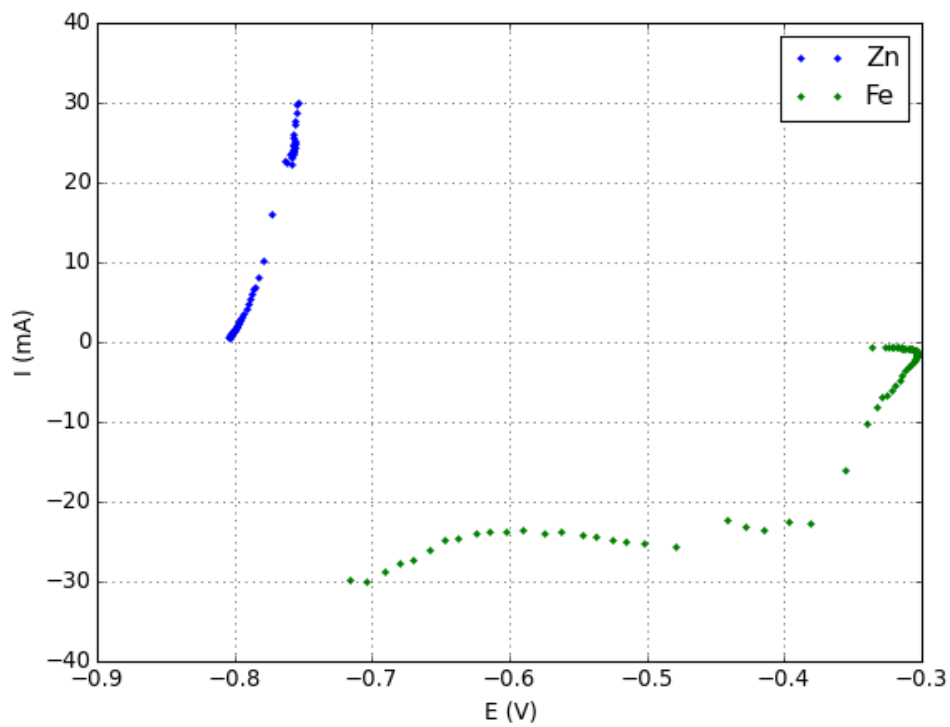


Voici des résultats obtenus avec le montage à fort courant :

```
figure()
[U,I,Ezn] = numpy.loadtxt("Zn-Fe-NaCl-power-5.txt")
plot(I,U,'.')
xlabel("I (mA)")
ylabel("U (V)")
grid()
axis([0,50,0,0.6])
```



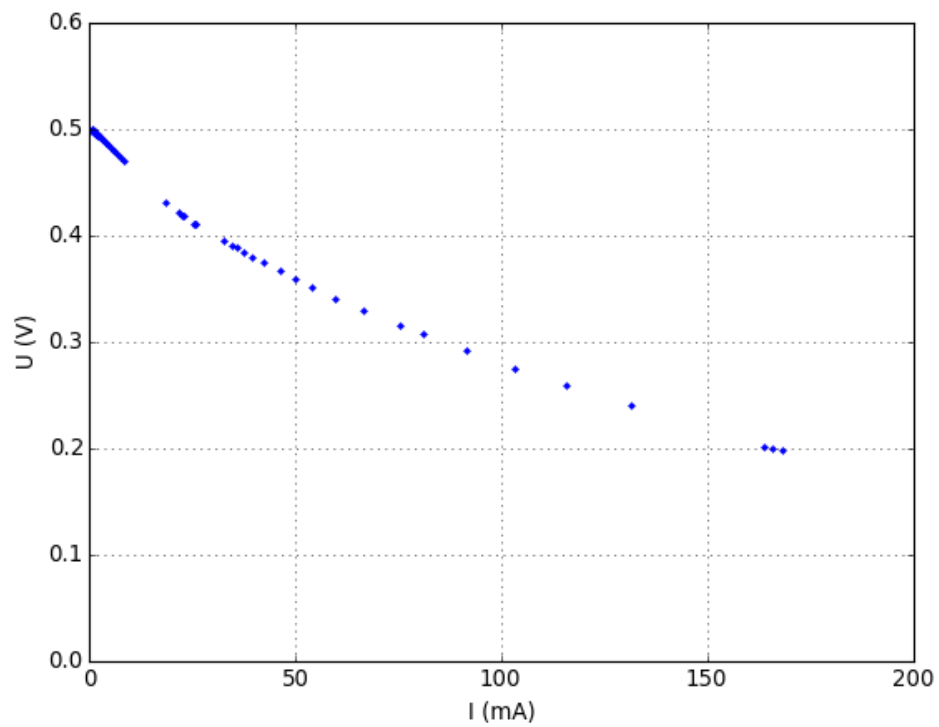
```
figure()
plot(Ezn,I, '.',label='Zn')
Efe = Ezn+U
plot(Efe,-I, '.',label='Fe')
xlabel("E (V)")
ylabel("I (mA)")
legend(loc='upper right')
grid()
```



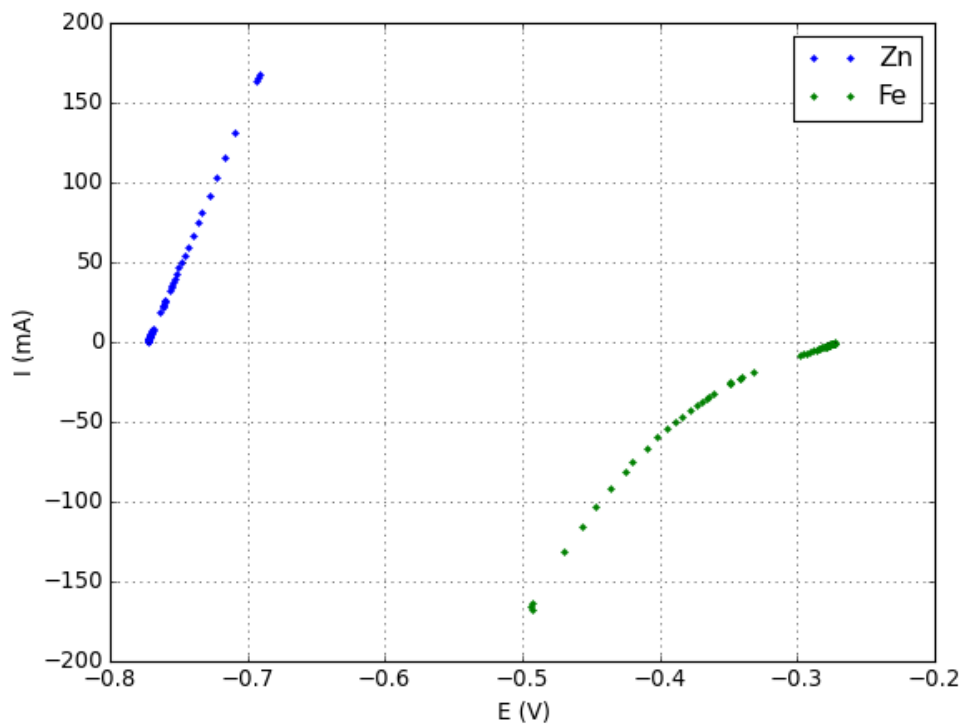
5.b. Solution NaCl acidifiée

La solution aqueuse a une concentration de $0,05 \text{ mol/L}$ en H_2SO_4 et 30 g/L en NaCl . Le circuit de mesure en fort courant est utilisé. Les électrodes sont poncées avant l'expérience.

```
figure()
[U,I,Ezn] = numpy.loadtxt("Zn-Fe-H2SO4-NaCl-power-6.txt")
plot(I,U,'.')
xlabel("I (mA)")
ylabel("U (V)")
grid()
axis([0,200,0,0.6])
```



```
figure()
plot(Ezn,I,'.',label='Zn')
Efe = Ezn+U
plot(Efe,-I,'.',label='Fe')
xlabel("E (V)")
ylabel("I (mA)")
legend(loc='upper right')
grid()
```

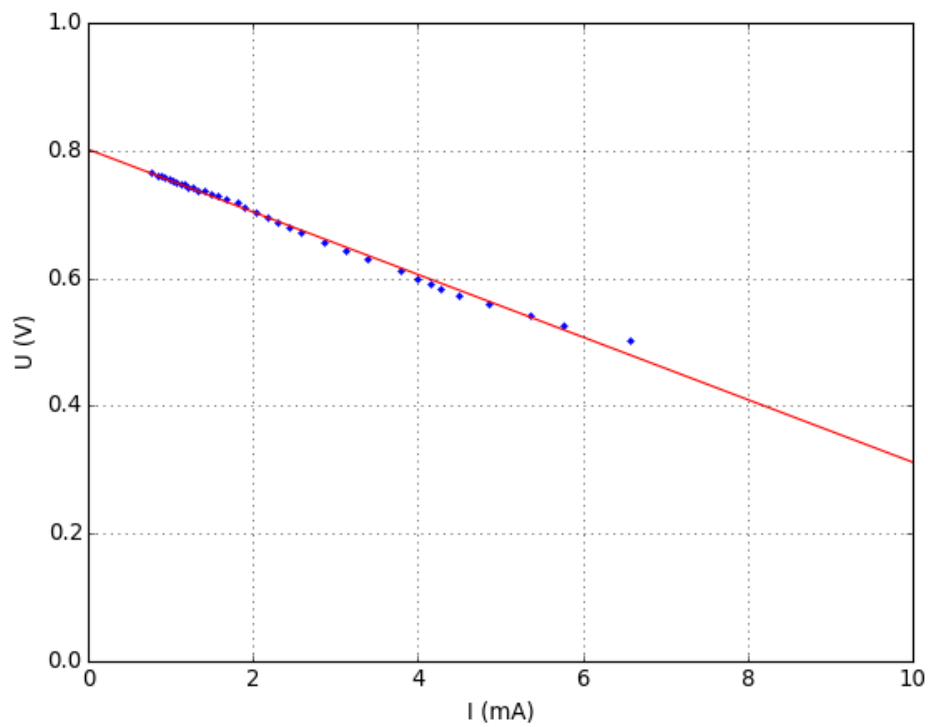



6. Pile zinc-cuivre (pile Volta)

6.a. Solution NaCl

L'électrode en cuivre est un tube de 1 cm de diamètre. L'électrode en zinc est une lame cylindrique entourant complètement l'électrode de cuivre. L'électrolyte est une solution aqueuse de chlorure de sodium de concentration 30 g/L. L'anode est le zinc. La première pile construite par Volta était de ce type : les rondelles de cuivre et de zinc étaient séparées par un tissu imbibé d'eau salée.

```
figure()
[U,I,Ezn] = numpy.loadtxt("Cu-Zn-2.txt")
plot(I,U,'. ')
xlabel("I (mA)")
ylabel("U (V)")
grid()
axis([0,10,0,1.0])
a,b,r_value,p_value,std_err = stats.linregress(I,U)
plot([0,-b/a],[b,0],'r')
E = b
R = -a*1000
```



Force électromotrice :

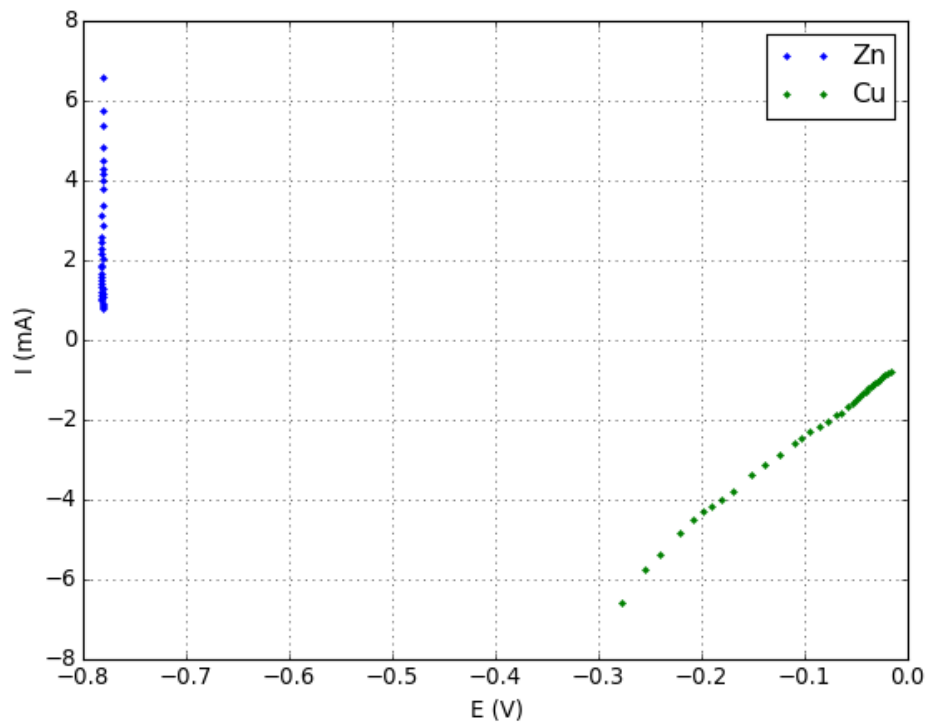
```
print(E)
--> 0.80179602297235375
```

Résistance interne :

```
print(R)
--> 49.001297331864691
```

On trace les courbes courant-potentielle pour chaque électrode :

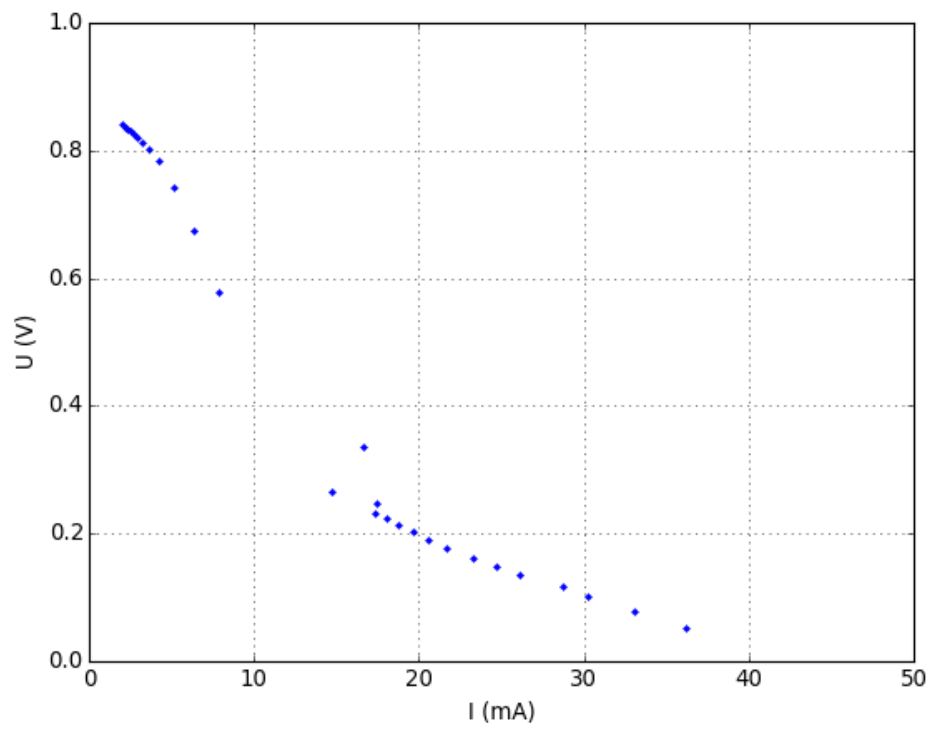
```
figure()
plot(Ezn, I, '.', label='Zn')
Ecu = Ezn+U
plot(Ecu, -I, '.', label='Cu')
xlabel("E (V)")
ylabel("I (mA)")
legend(loc='upper right')
grid()
```



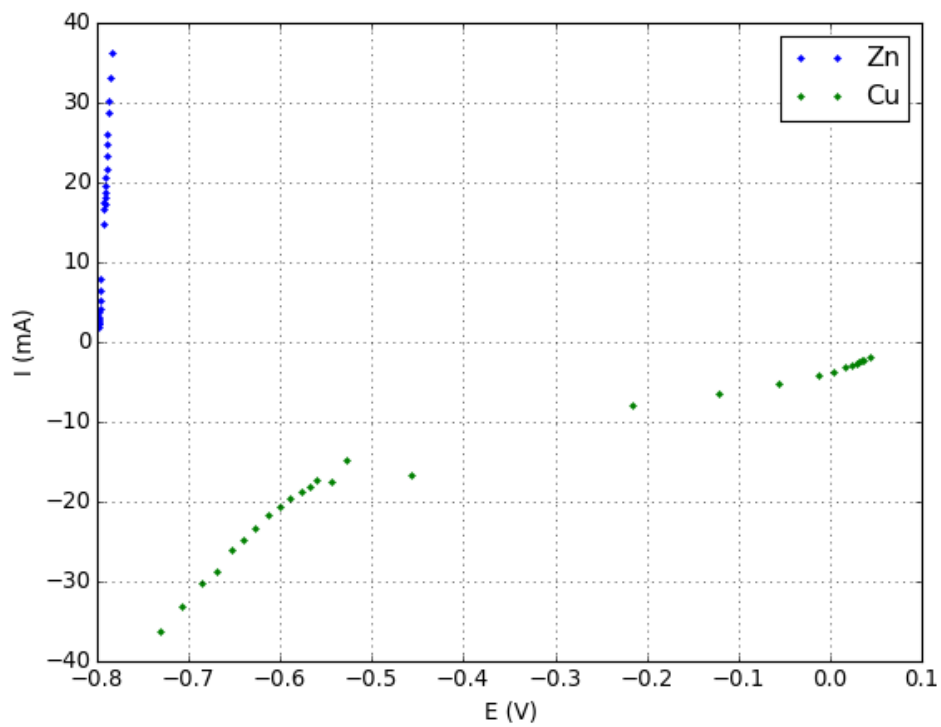
6.b. Solution NaCl acidifiée

Voici les résultats en fort courant, avec une solution H_2SO_4 $0,025 \text{ mol/L}$ et NaCl 30 g/L :

```
figure()
[U,I,Ezn] = numpy.loadtxt("Zn-Cu-H2SO4-NaCl-power-2.txt")
plot(I,U,'.')
xlabel("I (mA)")
ylabel("U (V)")
grid()
axis([0,50,0,1.0])
```

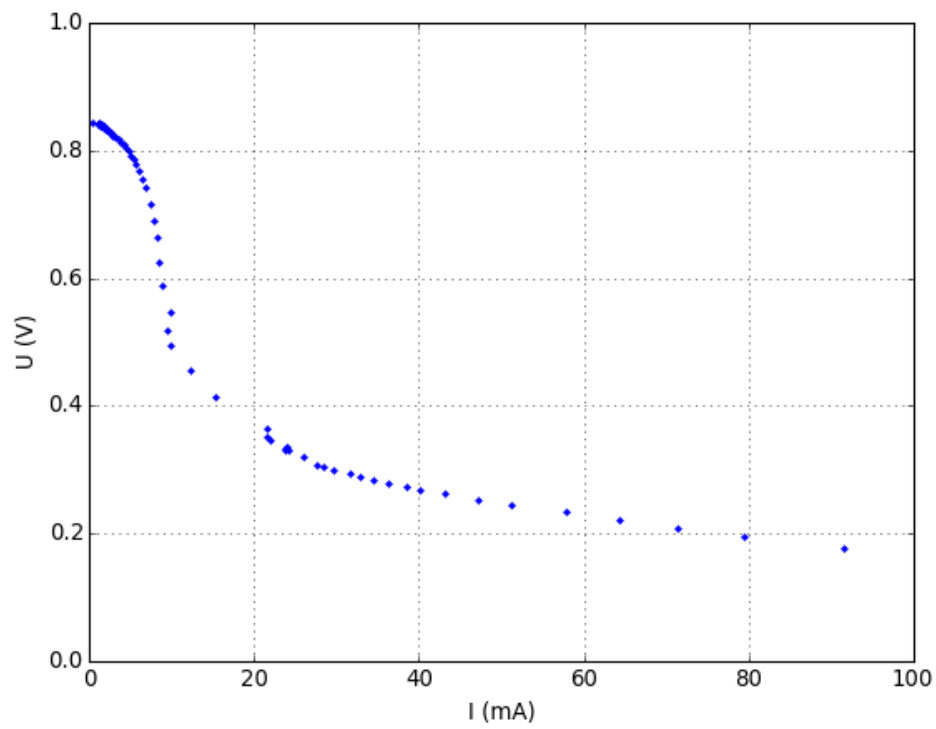


```
figure()
plot(Ezn,I,'.',label='Zn')
Ecu = Ezn+U
plot(Ecu,-I,'.',label='Cu')
xlabel("E (V)")
ylabel("I (mA)")
legend(loc='upper right')
grid()
```



Voici une autre expérience réalisée avec une solution H_2SO_4 0,05 mol/L et NaCl 30 g/L :

```
figure()
[U,I,Ezn] = numpy.loadtxt("Zn-Cu-H2SO4-NaCl-power-5.txt")
plot(I,U,'.')
xlabel("I (mA)")
ylabel("U (V)")
grid()
axis([0,100,0,1.0])
```



```
figure()
plot(Ezn,I,'.',label='Zn')
Ecu = Ezn+U
plot(Ecu,-I,'.',label='Cu')
xlabel("E (V)")
ylabel("I (mA)")
legend(loc='upper right')
grid()
```

