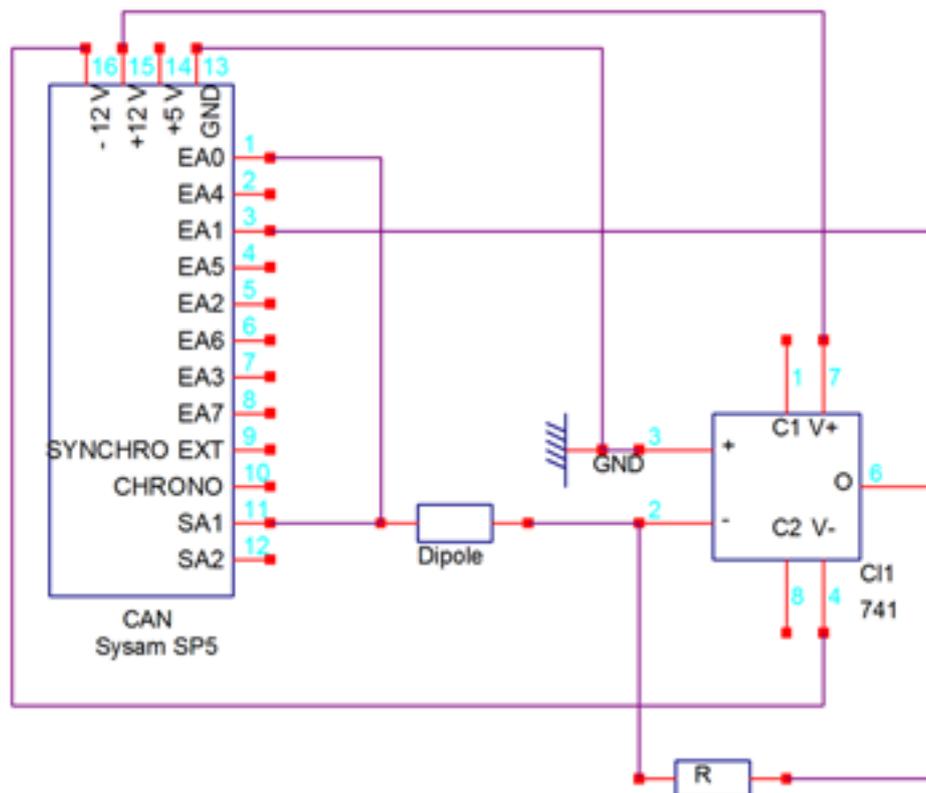


Caractéristique d'un dipôle

1. Obtention d'une caractéristique à faible courant

Pour des courants ne dépassant pas 20 mA, la caractéristique d'un dipôle peut être obtenue avec le montage suivant :



Le dipôle est alimenté directement par la sortie SA1 du CAN. Un convertisseur courant-tension permet de lire le courant sur la voie EA1, alors que la tension aux bornes du dipôle est lue sur la voie EA0. La résistance R effectue la conversion courant-tension ($i=u/R$). La résistance R est choisie en fonction du courant maximal supporté par le dipôle sous une tension de 10 V. L'amplificateur est alimenté par les bornes +/- 12 V du CAN.

Dans l'exemple ci-dessous, le dipôle est une thermistance d'environ 1 kOhm. La résistance R est choisie à 1 kOhm.

```
import pycan.main as pycan
import matplotlib.pyplot as plt
import numpy
import time
```

Pour acquérir un point de la caractéristique, il est préférable d'effectuer une acquisition échantillonnée que d'effectuer une mesure unique. Les valeurs de tension et de courant sont obtenues par moyennage des échantillons. L'écart-type permet d'obtenir les incertitudes dues aux fluctuations aléatoires. La fonction suivante effectue ces opérations. Une pause de 10 secondes est ici introduite pour laisser le temps à la thermistance d'atteindre un état thermique stationnaire.

```
def acquerirPoint(r,sys,tension):
    print("u = %f\n"%tension)
    sys.ecrire(1,tension,0,0)
    time.sleep(10)
    ne=1000
    te=1.0e-4
    sys.config_echantillon(te*1e6,ne)
    sys.acquerir()
    temps=sys.temps()
    tensions=sys.entrees()
    u = numpy.mean(tensions[0])
    du = numpy.std(tensions[0])
    i = -numpy.mean(tensions[1])/r
    di = numpy.std(tensions[1])/r
    print("u = %f, i= %f\n"%(u,i))
    return [u,du,i,di]
```

Lorsque le CAN SysamSP5 est branché et son câble USB connecté, les sorties analogiques SA1 et SA2 prennent une valeur de saturation +12 V ou -12 V. On commence donc par appliquer une tension nulle pour laisser la thermistance se refroidir :

```
sys=pycan.Sysam("SP5")
sys.ecrire(1,0.0,0,0)
time.sleep(30)
```

On configure les deux entrées utilisées, avec une tension maximale de 10 V :

```
sys.config_entrees([0,1],[10,10])
```

On donne les valeurs de la résistance R, du nombre de points à acquérir, des tensions minimale et maximale, puis on génère les différentes listes :

```
r=0.1 # résistance du convertisseur courant-tension en kOhm
np = 100
umin = 0
umax = -10
listeU = numpy.linspace(start=umin,stop=umax,num=np)
liste_u = numpy.zeros(np,dtype=numpy.float32)
liste_du = numpy.zeros(np,dtype=numpy.float32)
liste_i = numpy.zeros(np,dtype=numpy.float32)
liste_di = numpy.zeros(np,dtype=numpy.float32)
```

La boucle d'acquisition :

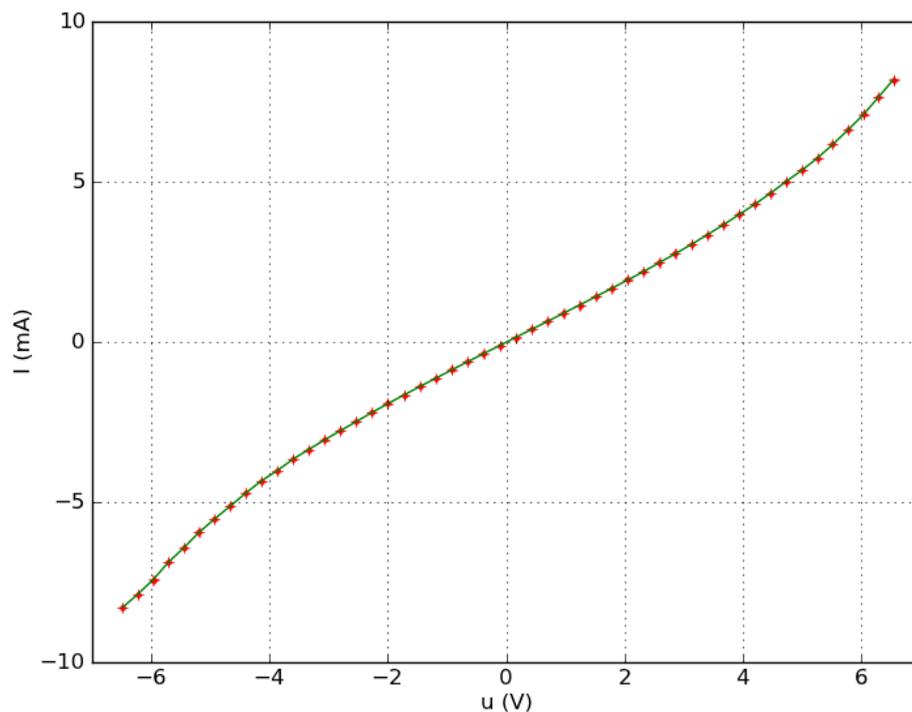
```
for k in range(np):  
    [liste_u[k],liste_du[k],liste_i[k],liste_di[k]] = acquerirPoint(r,sys,listeU[k])
```

On ferme l'interface puis on sauvegarde les données :

```
sys.fermer()  
numpy.savetxt('thermistance-10sec.txt',[liste_u,liste_du,liste_i,liste_di])
```

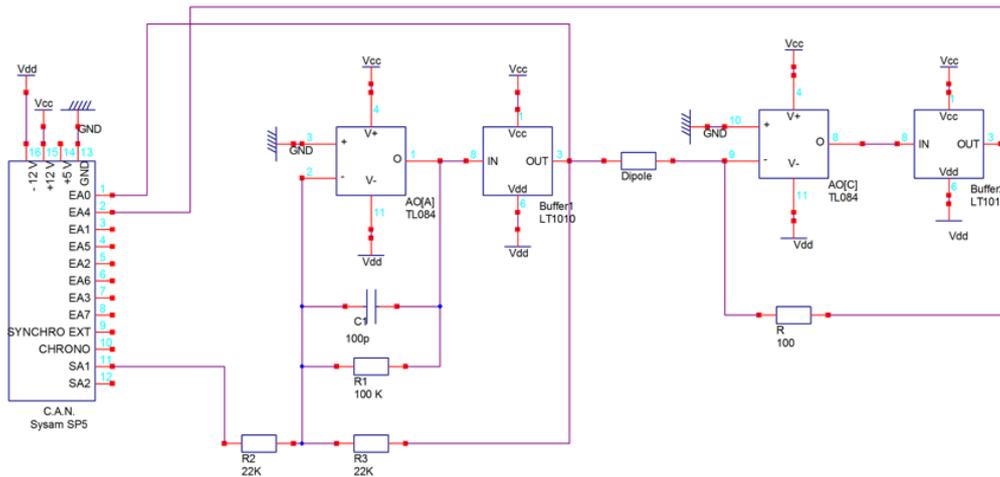
On effectue enfin le tracé de la caractéristique avec les barres d'incertitude :

```
plt.figure()  
plt.plot(liste_u,liste_i,marker='.',linestyle='',color='r')  
plt.errorbar(liste_u,liste_i,xerr=liste_du,yerr=liste_di,ecolor='r')  
plt.axis([0,10,-10/r,10/r])  
plt.xlabel("u (V)")  
plt.ylabel("I (mA)")  
plt.grid()  
plt.show()
```



2. Obtention d'une caractéristique à courant moyen

Pour des courants jusqu'à 150 mA, un buffer de puissance LT1010 peut-être utilisé. Il en faut un pour amplifier la sortie du CAN et un second pour le convertisseur courant-tension. Dans les deux cas, le buffer est placé dans la boucle de rétroaction de l'amplificateur.



Dans l'exemple ci-dessous, le dipôle est une lampe à incandescence 12 V 100 mA. La résistance du convertisseur courant-tension est de 100 Ohm (2 Watt). À chaque point, le temps de pause avant la mesure est de 10 secondes. Voici la caractéristique obtenue :

