

La loi d'Ohm (version polyfit et avec fonctions)

I(mA)	0	25	50	75	100	125
U(V)	0	1,7	3,4	5,1	6,8	8,5

tableau.png

In [1]:

```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:

```
# création de la fonction modelisation
# modélisation par une droite d'équation
# y=ax+b (polynôme d'ordre 1)

# la fonction polyfit permet de déterminer les
# coefficients a=coef[0] et b=coef[1] de la droite
# ymodel permet de déterminer les valeurs modélisées de y

def modelisation(x,y):
    coef=np.polyfit(x,y,1)
    ymodel=coef[0]*x+coef[1]
    print ('U= {0:.1f}'.format(coef[0]), 'x I')
    return (ymodel)
```

In [3]:

```
# fonction permettant de tracer le graphique avec les points
# expérimentaux et la courbe obtenue après modélisation

def courbemodelisee (x,y,ymodel) :
    fig = plt.figure(figsize=(12,10))
    plt.plot(x,y,'r+',label='U=f(I)')
    plt.plot(x,ymodel,'b',label='modèle linéaire')
    plt.legend()
    plt.xlabel("intensité I (A)")
    plt.ylabel("tension U (V)")
    plt.grid()
    plt.title("Caractéristique Intensité-Tension dun "
              "dipôle ohmique")
    plt.show()
```

In [4]:

```
# tableaux numpy obligatoires à cause de la
# fonction np.polyfit utilisée auparavant

I=np.array([0,25e-3,50e-3,75e-3,100e-3,125e-3])
```

```
U=np.array([0,1.7,3.4,5.1,6.8,8.5])
Umodel=modelisation(I,U)
courbemodelisee(I,U,Umodel)
```

$U = 68.0 \times I$

