

CN 4

Correction

Code Capytale : 9df2-7770217

```
Entrée[1]: import numpy as np
import matplotlib.pyplot as plt
from cmath import phase
```

Signal d'entrée

$$e(t) = E_0 + \frac{4A}{\pi} \left(\sin(\omega t) + \frac{1}{3} \sin(3\omega t) + \frac{1}{5} \sin(5\omega t) + \frac{1}{7} \sin(7\omega t) \dots \right)$$

Paramètres du signal d'entrée

```
Entrée[2]: A = 2      # Amplitude du signal carré
E0 = 4      # Valeur moyenne
f0 = 100    # Fréquence du signal carré
N = 50      # Nombre de termes
```

Tracé du spectre d'entrée en amplitude

```
Entrée[3]: coef_e=[]
f=[]
plt.figure(1)
plt.vlines(0,0,E0,'b',label='e(t)') # Tracé de la composante continue
for k in range(N):                  # Tracé des harmoniques
    coef_e.append((4*A)/(np.pi*(2*k+1)))
    f.append(f0*(2*k+1))
    plt.vlines(f[k],0,coef_e[k],'b')
plt.xlabel('f(Hz)')
plt.ylabel('amplitude(v)')
plt.title('Spectres')
```

Sortie[3]: Text(0.5, 1.0, 'Spectres')

Tracé de e(t)

```
Entrée[4]: plt.figure(2)
t = np.linspace(0, 2/f0, 2000) # 2 périodes
e=E0
for k in range(N):
    e+=coef_e[k]*np.sin(2*np.pi*f[k]*t)
plt.plot(t,e,'b',label='e(t)')
plt.xlabel('t(s)')
plt.grid()
```

Action d'un filtre passe-bas

$$\underline{H} = \frac{H_0}{1+j\frac{f}{f_c}}$$

Fonction de transfert

```
Entrée[5]: def H(H0,f,f_c):
    return H0/(1+1j*(f/f_c))
```

Paramètres du filtre

```
Entrée[6]: f_c=100
H0=1
```

Signal de sortie

Calcul et tracé du spectre en amplitude

```
Entrée[7]: S0 = E0*abs(H(H0,0,f_c)) # Valeur moyenne du signal de sortie
coef_s = []

plt.figure(1)
plt.vlines(0,0,S0,'r',label='s(t)') # Tracé de La composante continue
# Tracé des harmoniques
for k in range(N):
    coef_s.append(coef_e[k]*abs(H(H0,f[k],f_c))) # Calcul des amplitudes des harmoniques du signal de sortie
    plt.vlines(f[k],0,coef_s[k],'r')
plt.legend()
plt.show()
```

Tracé de s(t)

```
Entrée[8]: plt.figure(2)
t = np.linspace(0, 2/f0, 2000) # 2 périodes
s=[S0]
phi=[]
for k in range(N):
    phi.append(phase(H(H0,f[k],f_c))) # Calcul des phases à L'origine des harmoniques du signal de sortie
    s+=coef_s[k]*np.sin(2*np.pi*f[k]*t+phi[k])
plt.plot(t,s,'r',label='s(t)')
plt.legend()
plt.show()
```

Figure 1

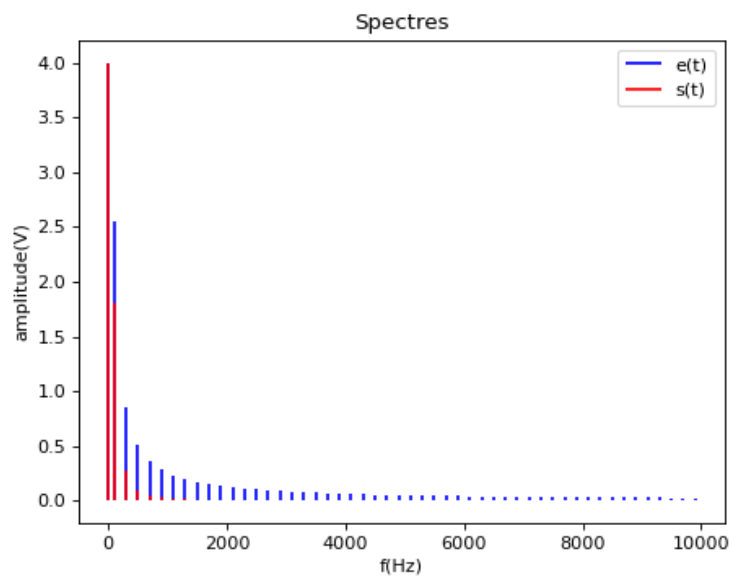


Figure 2

