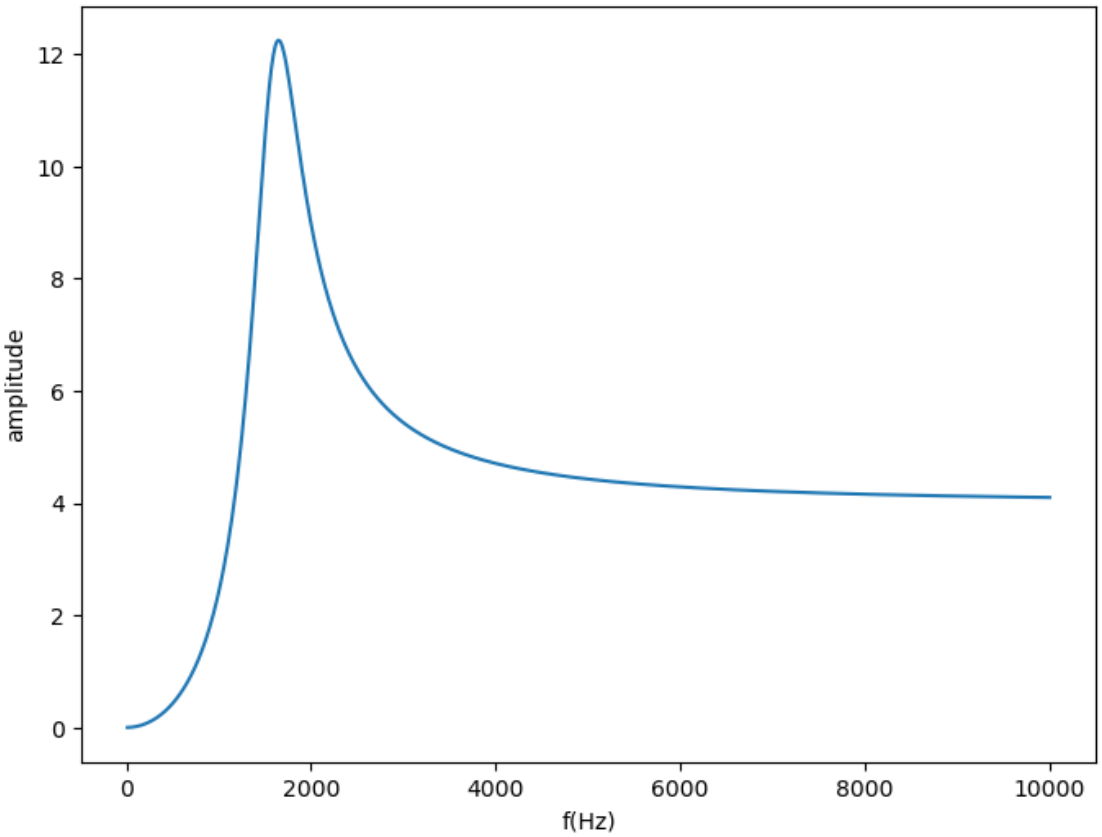
Résonances

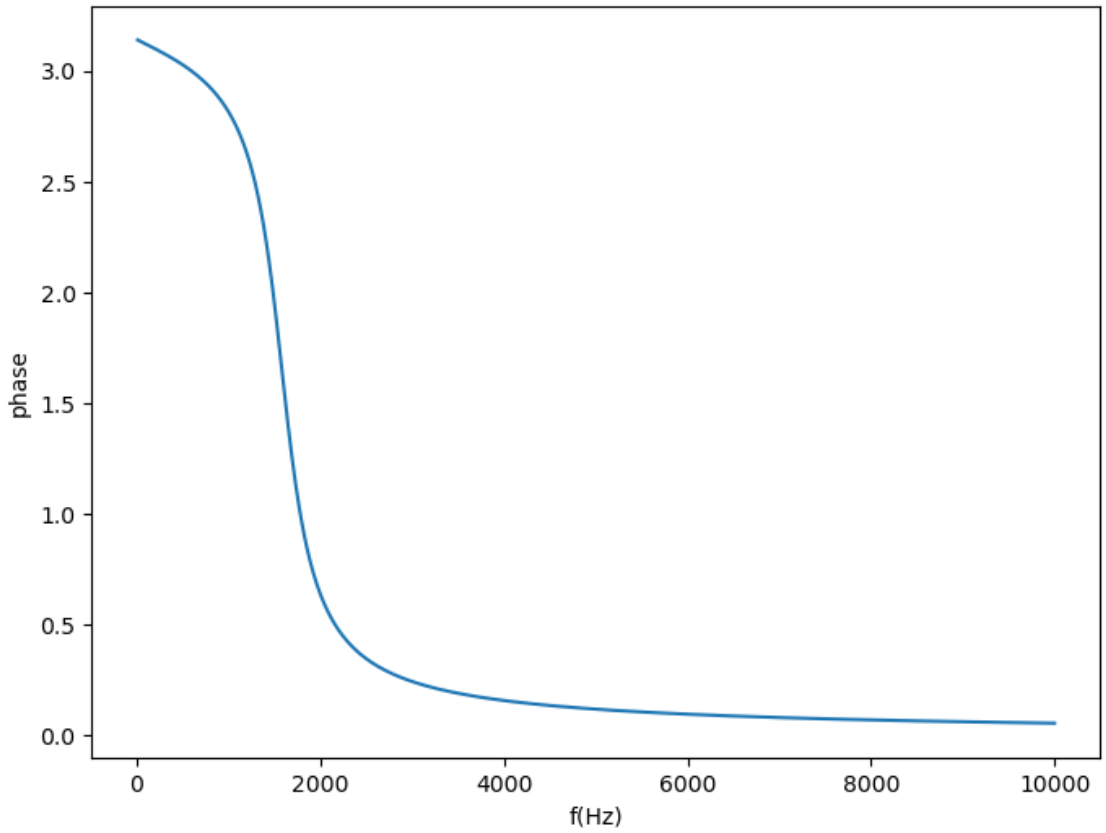
1 Circuit RLC

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

[2]:
```
R,L,C=70,21e-3,470e-9
E=4
def _uL(f):
    w=2*np.pi*f
    return 1j*L*w*E/(R+1j*L*w+1/(1j*C*w))
```

[3]:
```
tab_f=np.linspace(10,10000,1000)
tab__uL=_uL(tab_f)
tab_module_uL=np.abs(tab__uL)
tab_phase_uL=np.angle(tab__uL)
plt.figure(figsize=(8,6))
plt.plot(tab_f,tab_module_uL)
plt.xlabel("f(Hz)")
plt.ylabel("amplitude")
plt.show()
plt.figure(figsize=(8,6))
plt.plot(tab_f,tab_phase_uL)
plt.xlabel("f(Hz)")
plt.ylabel("phase")
plt.show()
```

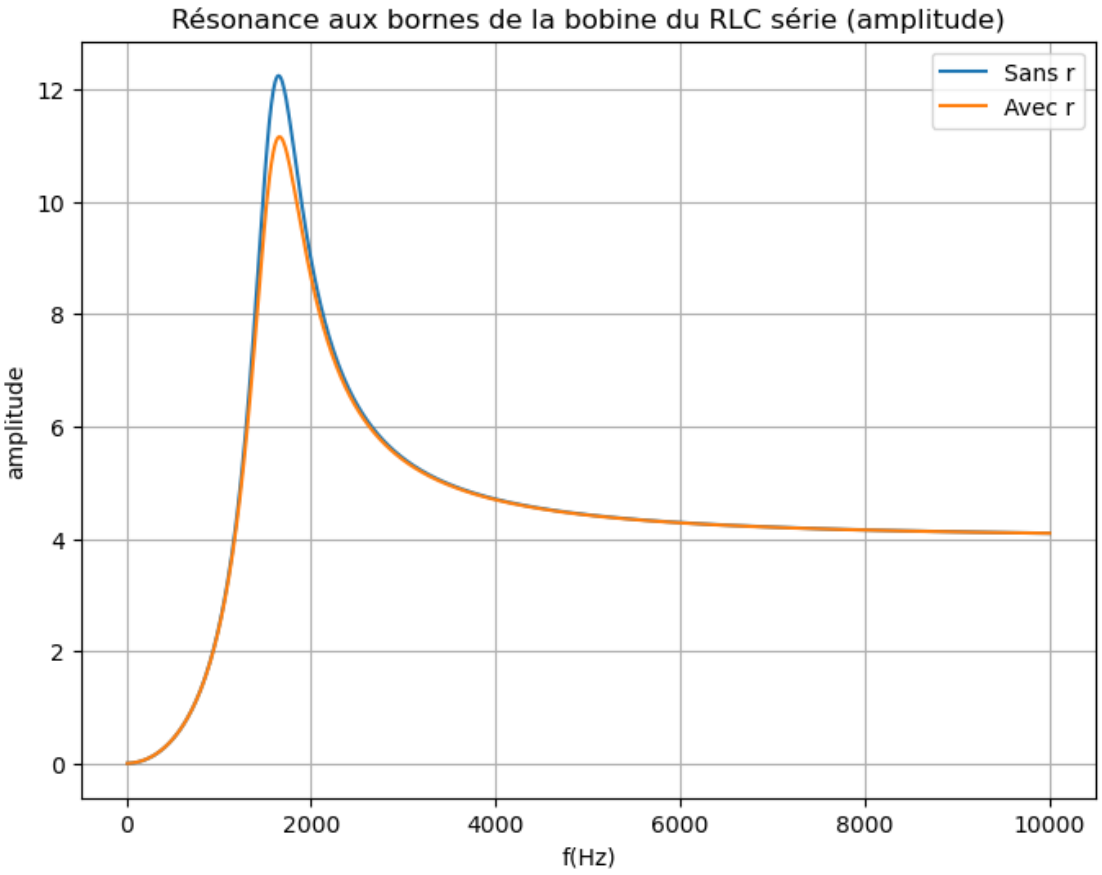

amplitude
f(Hz)
0
2
4
6
8
10
12
0
2000
4000
6000
8000
10000

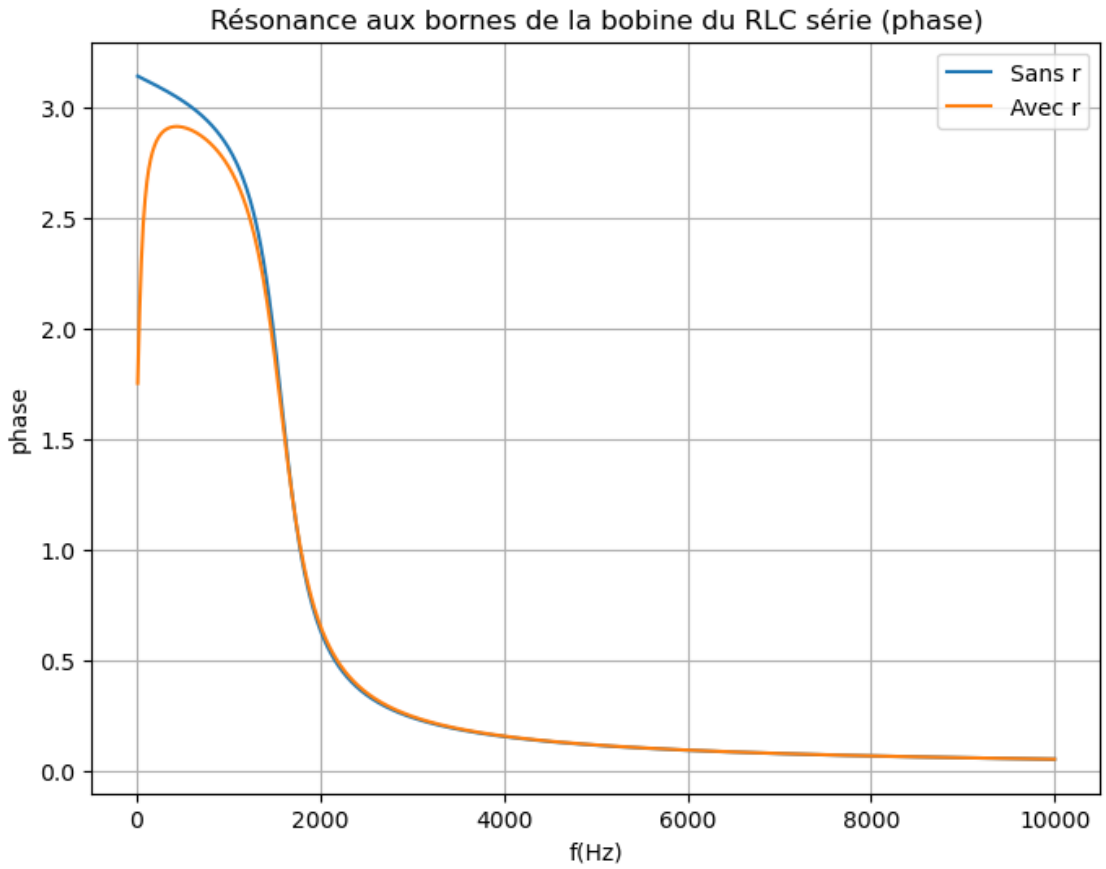


[4]:
```
R,r=70,7.1
def _uL2(f):
    w=2*np.pi*f
    return (r+1j*L*w)*E/(R+r+1j*L*w+1/(1j*C*w))
```

[5]:
```
tab__uL2=_uL2(tab_f)
tab_module_uL2=np.abs(tab__uL2)
tab_phase_uL2=np.angle(tab__uL2)
plt.figure(figsize=(8,6))
plt.plot(tab_f,tab_module_uL,label="Sans r")
plt.plot(tab_f,tab_module_uL2,label="Avec r")
plt.xlabel("f(Hz)")
plt.ylabel("amplitude")
plt.title("Résonance aux bornes de la bobine du RLC série (amplitude)")
plt.legend()
plt.grid()
plt.show()
plt.figure(figsize=(8,6))
plt.plot(tab_f,tab_phase_uL,label="Sans r")
plt.plot(tab_f,tab_phase_uL2,label="Avec r")
```

```
plt.xlabel("f(Hz)")
plt.ylabel("phase")
plt.title("Résonance aux bornes de la bobine du RLC série (phase)")
plt.legend()
plt.grid()
plt.show()
```

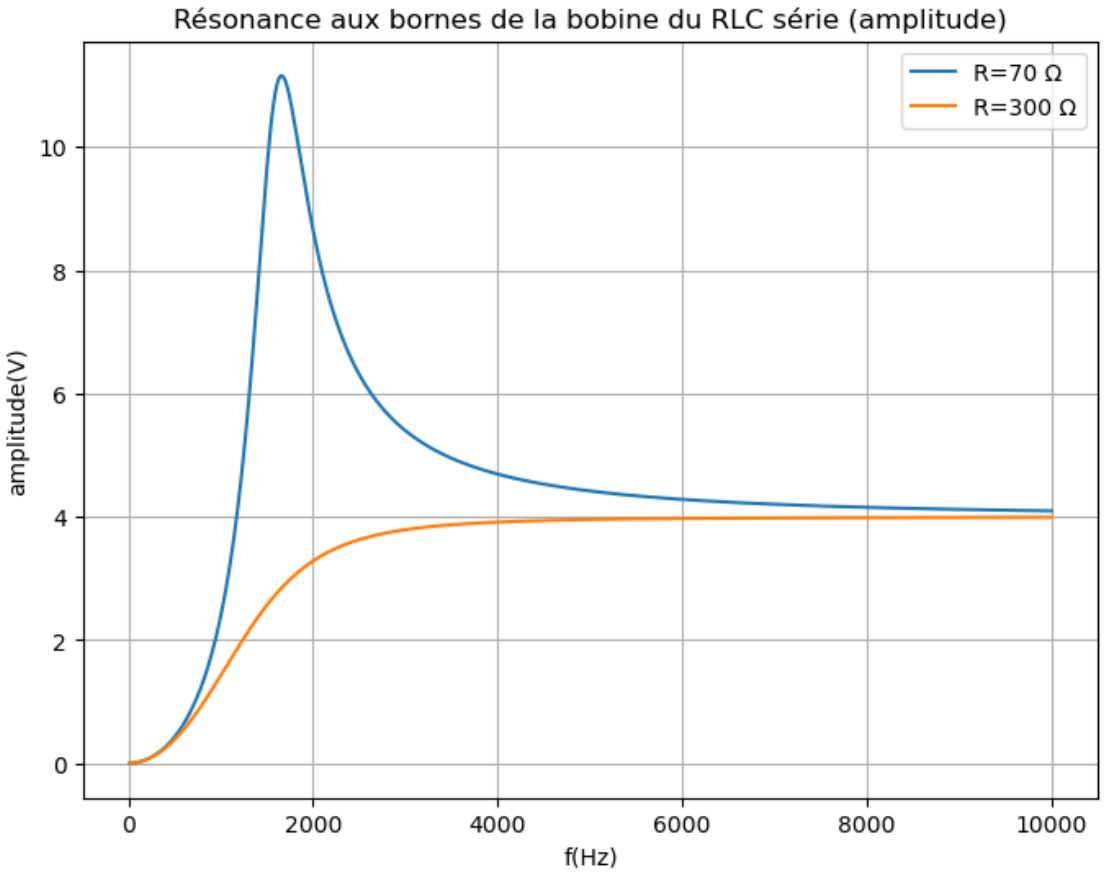


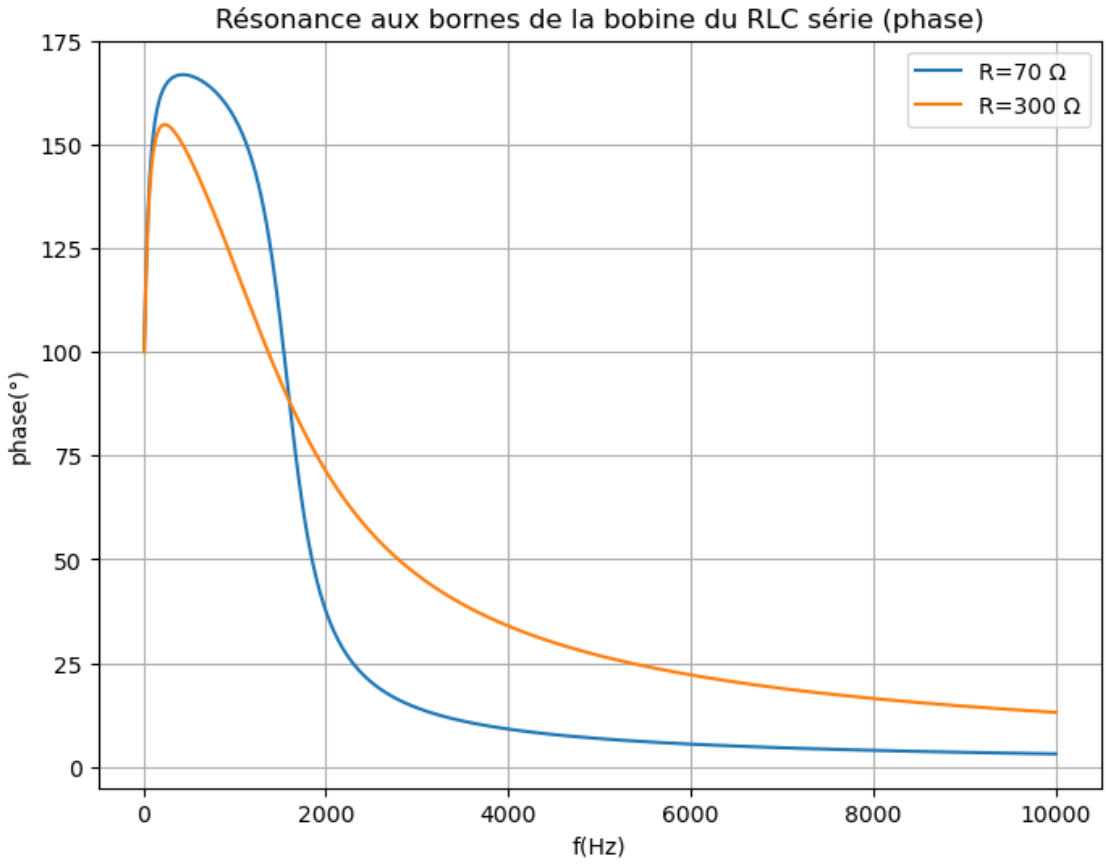


[6]:
```
ulmax=np.max(tab_module_uL)
print(ulmax)
def equation(f):
    return np.abs(_uL(f))-ulmax/np.sqrt(2)
from scipy.optimize import bisect
print(bisect(equation,1000,1700))
print(bisect(equation,1700,5000))
ulmax2=np.max(tab_module_uL2)
def equation2(f):
    return np.abs(_uL2(f))-ulmax2/np.sqrt(2)
print(bisect(equation2,1000,1700))
print(bisect(equation2,1700,5000))
```

```
12.247472402729956
1420.5500368664925
2036.905931979406
1409.2402279969979
2112.688951299816
```

[7]:
```
tab_f=np.linspace(10,10000,1000)
R=300
tab__uL2b=_uL2(tab_f)
tab_module_uL2b=np.abs(tab__uL2b)
plt.figure(figsize=(8,6))
plt.plot(tab_f,tab_module_uL2,label="R=70 Ω")
plt.plot(tab_f,tab_module_uL2b,label="R=300 Ω")
plt.xlabel("f(Hz)")
plt.ylabel("amplitude(V)")
plt.legend()
plt.grid()
plt.title("Résonance aux bornes de la bobine du RLC série (amplitude)")
plt.show()
plt.figure(figsize=(8,6))
plt.plot(tab_f,np.rad2deg(np.angle(tab__uL2)),label="R=70 Ω")
plt.plot(tab_f,np.rad2deg(np.angle(tab__uL2b)),label="R=300 Ω")
plt.xlabel("f(Hz)")
plt.ylabel("phase(°)")
plt.legend()
plt.grid()
plt.title("Résonance aux bornes de la bobine du RLC série (phase)")
plt.show()
```





2 Quartz

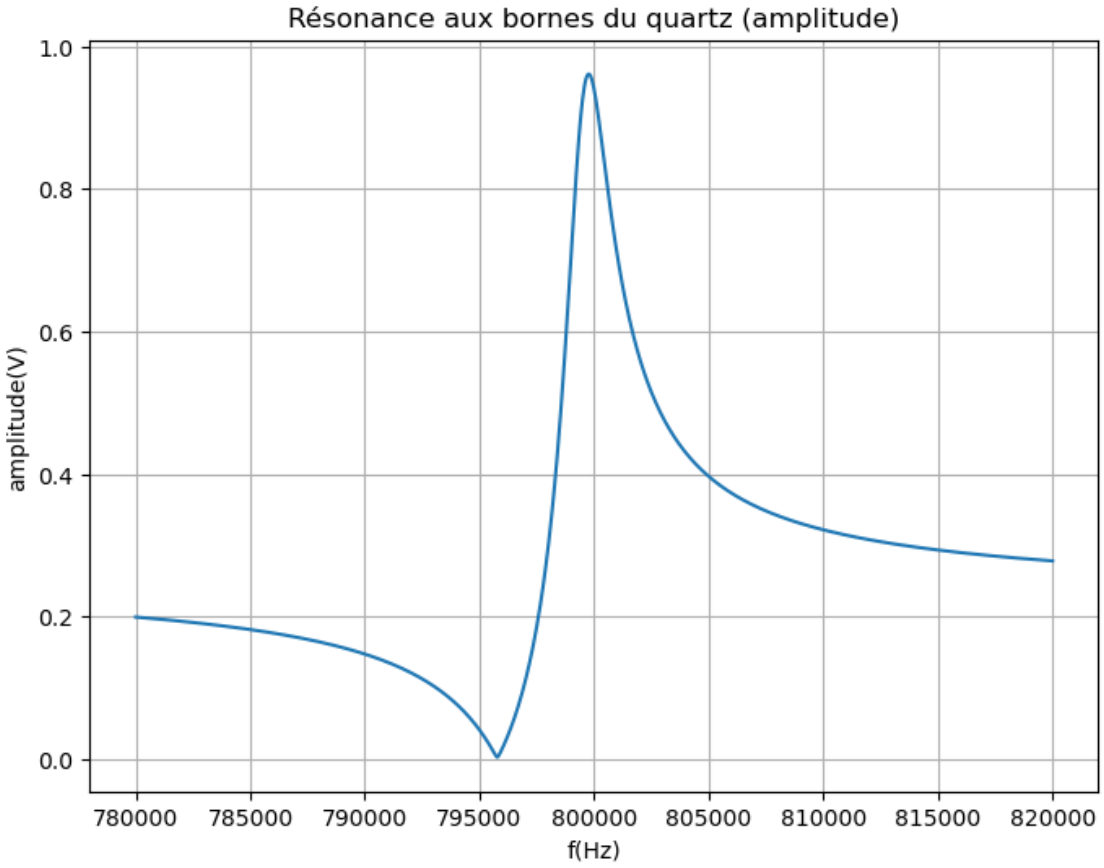
La branche du bas est équivalente à $\underline{Z_1} = R + jL\omega + \frac{1}{jC_s\omega}$. Ensuite, par association en parallèle:

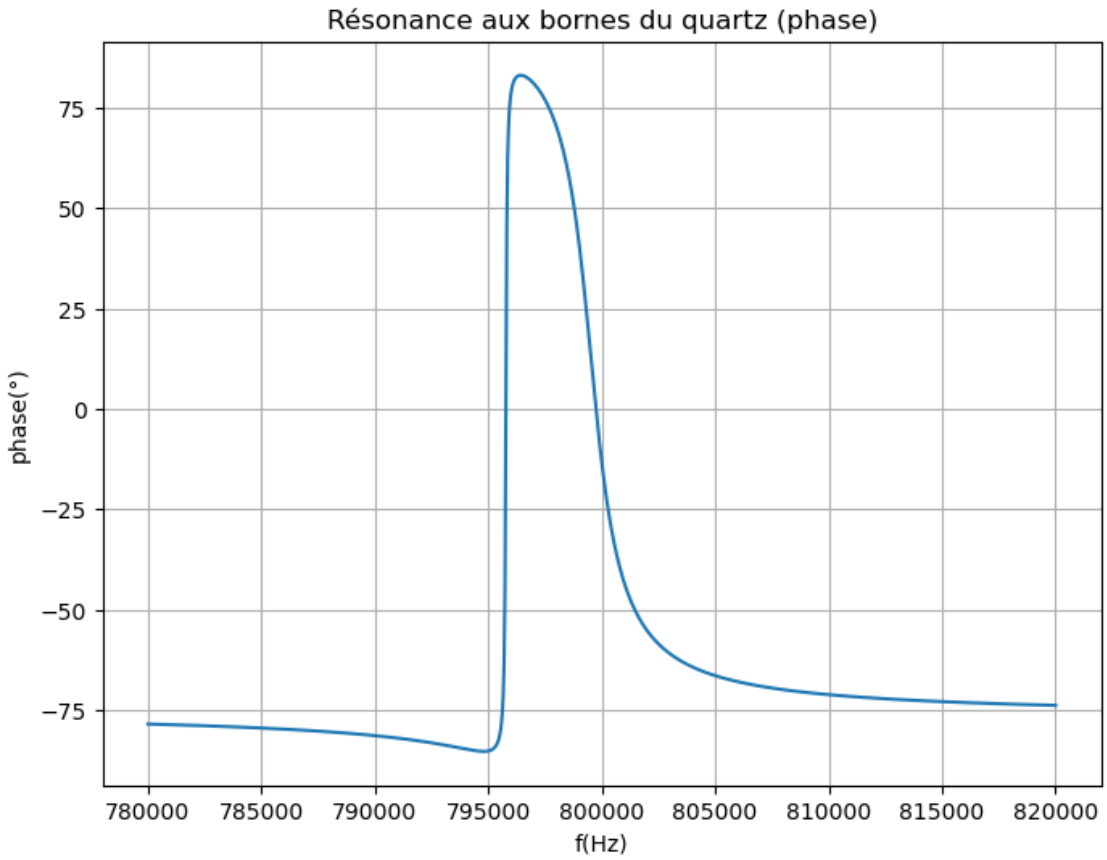
$$\underline{Z} = \frac{1}{jC_p\omega + \frac{1}{R+jL\omega+\frac{1}{jC_s\omega}}}$$

[8]:
```
R,L,Cs,Cp=250,500e-3,0.08e-12,8e-12
def _Z(f):
    w=2*np.pi*f
    _Z1=R+1j*L*w+1/(1j*Cs*w)
    return 1/(1j*Cp*w+1/_Z1)
```

[9]:
```
Rp=100e3
def _uQ(f):
    ZZ=_Z(f)
```

```
    return ZZ/(Rp+ZZ)
tab_f=np.linspace(780e3,820e3,1000)
tab__uQ=_uQ(tab_f)
tab_module_uQ=np.abs(tab__uQ)
plt.figure(figsize=(8,6))
plt.plot(tab_f,tab_module_uQ)
plt.xlabel("f(Hz)")
plt.ylabel("amplitude(V)")
plt.title("Résonance aux bornes du quartz (amplitude)")
plt.grid()
plt.show()
plt.figure(figsize=(8,6))
plt.plot(tab_f,np.rad2deg(np.angle(tab__uQ)))
plt.xlabel("f(Hz)")
plt.ylabel("phase(°)")
plt.title("Résonance aux bornes du quartz (phase)")
plt.grid()
plt.show()
```





[13]:
```
uQmax=np.max(tab_module_uQ)
fres=tab_f[np.argmax(tab_module_uQ)]
print(uQmax)
def equationQ(f):
    return np.abs(_uQ(f))-uQmax/np.sqrt(2)
from scipy.optimize import bisect
a=bisect(equationQ,795000,800000)
b=bisect(equationQ,800000,805000)
print(a,b)
Q=fres/(b-a)
print(Q)
```

```
0.9612418210519935
798942.9031953791 801153.4927782106
361.79478361395525
```

[ ]: