

Filtry Pasywne

1. Wstep

Filtrem częstotliwości nazywamy układ o strukturze czwórnika (czwórnik to układ mający cztery zaciski – jedna z par zacisków pełni rolę wejścia, zaś druga wyjścia), który przepuszcza bez tłumienia lub z małym tłumieniem napięcia i prądy o określonym paśmie częstotliwości, a tłumi napięcia i prądy leżące poza tym pasmem. Filtry częstotliwości mają głównie zastosowanie w urządzeniach elektronicznych i energetycznych.

Wyróżniamy filtry aktywne, czyli takie w których układzie wykorzystuje się elementy aktywne takie jak wzmacniacze operacyjne. Na zajęciach zajmowaliśmy się filtrami pasywnymi zbudowanymi z samych elementów pasywnych.

```
In [2]: import pandas as pd
import numpy as np
import scipy.signal as signal
import matplotlib.pyplot as plt
from IPython.display import clear_output
```

Filtr dolnoprzepustowy

Wartości teoretyczne jakie wybraliśmy to rezystancja $10k\Omega$ natomiast pojemność $3.3nF$. Do pomiarów wykorzystaliśmy kondensator o pojemności $3.3nF$ oraz rezystor o rezystancji $9,9664k\Omega$.

```
In [9]: R = 9.9664 * 1e3
C = 3.3 * 1e-9

# Czestotliwosc graniczna naszego filtra
borderFreq = 1 / (2 * np.pi * R * C)
print(f"Czestotliwosc graniczna uzytego filtra wynosi: {borderFreq}")
print(f"Czestosc graniczna uzytego filtra wynosi: {borderFreq*2*np.pi}")
```

```
Czestotliwosc graniczna uzytego filtra wynosi: 4839.136562239894
Czestosc graniczna uzytego filtra wynosi: 30405.191747301236
```

Transmitancja filtra dolno przepustowego

$$G(s) = \frac{1}{1 + RCs}$$

$$G(s) = \frac{1}{1 + 3.288912 \cdot 10^{-5} s}$$

```
In [4]: dolnoPrzepustowy = pd.read_csv("dolno_przepustowy.csv", sep=";")
dolnoPrzepustowy["G[db]"] = 20 * np.log10(dolnoPrzepustowy["Vpp_wyj"] / dolnoPrzepustowy["Vpp_wej"])
```

```
Out[4]:
```

	Vpp_wyj	Vpp_wej	Freq	G[db]
0	1.04	1.02	1.0	0.168663
1	1.04	1.02	1.3	0.168663
2	1.04	1.02	1.7	0.168663
3	1.04	1.02	2.1	0.168663
4	1.02	1.02	2.8	0.000000
5	1.02	1.02	3.6	0.000000
6	1.02	1.02	4.6	0.000000
7	1.02	1.02	6.0	0.000000
8	1.00	1.02	7.7	-0.172003
9	1.00	1.02	10.0	-0.172003
10	0.96	1.02	13.0	-0.526579
11	0.94	1.02	17.0	-0.709446
12	0.90	1.02	21.0	-1.087153
13	0.82	1.02	28.0	-1.895726
14	0.76	1.02	36.0	-2.555732
15	0.66	1.02	46.0	-3.781125
16	0.56	1.02	60.0	-5.208243
17	0.48	1.02	77.0	-6.547179
18	0.38	1.02	100.0	-8.576332

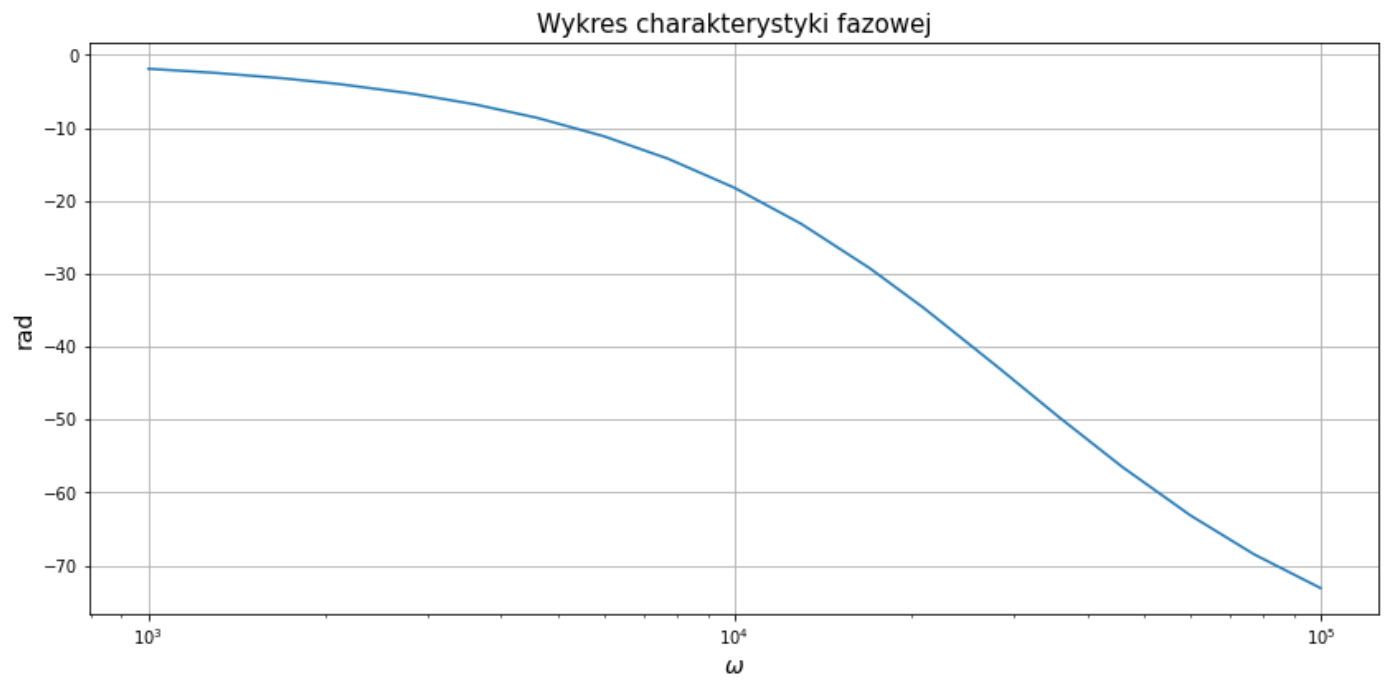
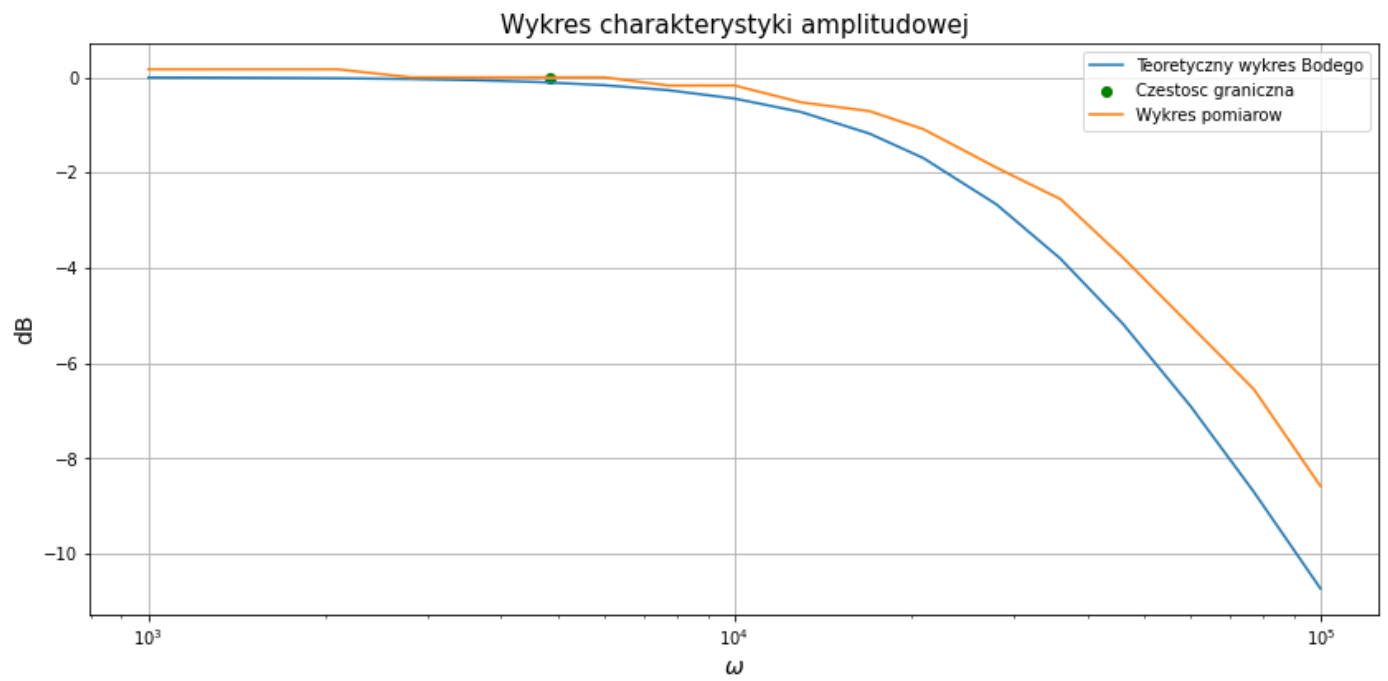
```
In [5]: num = [0, 1]
den = [R*C, 1]
s1 = signal.TransferFunction(num, den)

w, mag, phase = signal.bode(s1, w=dolnoPrzepustowy["Freq"]*1000)
clear_output()

fig, (ax1, ax2) = plt.subplots(2, 1, figsize=(14, 14))
fig.suptitle("Charakterystyka Bodego", fontsize=20)
ax1.semilogx(w, mag, label="Teoretyczny wykres Bodego")
ax1.scatter(borderFreq, 0, color="green", label="Czestosc graniczna")
ax1.semilogx(w, dolnoPrzepustowy["G[db]"], label="Wykres pomiarow")
ax1.set_title("Wykres charakterystyki amplitudowej", fontsize=15)
ax1.set_xlabel("$\omega$", fontsize=14)
ax1.set_ylabel("dB", fontsize=14)
ax1.grid()
ax1.legend()

ax2.semilogx(w, phase)
# ax2.set_xlim(10e-10, 10e10)
ax2.set_title("Wykres charakterystyki fazowej", fontsize=15)
ax2.grid()
ax2.set_xlabel("$\omega$", fontsize=14)
ax2.set_ylabel("rad", fontsize=14)
plt.show()
```

Charakterystyka Bodego



Filtr gorno przepustowy

$$G(s) = \frac{RC}{RC \cdot s + 1}$$

```
In [6]: R = 9.06 * 1e3
C = 1.134 * 1e-9

# Czystotliwosc graniczna naszego filtra
borderFreq = 1 / (2 * np.pi * R * C)
print(f"Czystotliwosc graniczna uzytego filtra gorno przepustowego wynosi: {borderFreq}")
print(f"Czystosc graniczna uzytego filtra gornoprzepustowego wynosi: {borderFreq*2*np.pi}
```

Czystotliwosc graniczna uzytego filtra gorno przepustowego wynosi: 15490.979506785583
Czystosc graniczna uzytego filtra gornoprzepustowego wynosi: 97332.69483085525

```
In [7]: gornoPrzepustowy = pd.read_csv("gorno_przepustowy.csv", sep=";")
gornoPrzepustowy["Vpp_wyj"] = gornoPrzepustowy["Vpp_wyj"] * 1e-3
gornoPrzepustowy["Freq"] = gornoPrzepustowy["Freq"] * 1e3
gornoPrzepustowy["G[db]"] = 20 * np.log10(gornoPrzepustowy["Vpp_wyj"] / gornoPrzepustowy
```

```
Out[7]:
```

	Vpp_wyj	Vpp_wej	Freq	G[db]
0	0.010	1.04	1000.0	-40.340667
1	0.032	1.04	1300.0	-30.237667
2	0.072	1.04	1700.0	-23.194017
3	0.104	1.04	2100.0	-20.000000
4	0.168	1.04	2800.0	-15.834481
5	0.216	1.04	3600.0	-13.651592
6	0.264	1.04	4600.0	-11.908588
7	0.336	1.04	6000.0	-9.813881
8	0.400	1.04	7700.0	-8.299467
9	0.488	1.04	10000.0	-6.572270
10	0.568	1.04	13000.0	-5.253700
11	0.568	1.04	17000.0	-5.253700
12	0.648	1.04	21000.0	-4.109167
13	0.704	1.04	28000.0	-3.389214
14	0.760	1.04	36000.0	-2.724395
15	0.820	1.04	46000.0	-2.064390
16	0.860	1.04	60000.0	-1.650698
17	0.880	1.04	77000.0	-1.451013
18	0.900	1.04	100000.0	-1.255817

```
In [8]: num = [R*C, 0]
den = [R*C, 1]
s1 = signal.TransferFunction(num, den)

w, mag, phase = signal.bode(s1, w=gornoPrzepustowy["Freq"])
clear_output()

fig, (ax1, ax2) = plt.subplots(2, 1, figsize=(14, 14))
```

```

fig.suptitle("Charakterystyka Bodego", fontsize=20)
ax1.semilogx(w, mag, label="Teoretyczny wykres")
ax1.scatter(borderFreq*2*np.pi, 0, color="green", label="Czestosc graniczna")
ax1.semilogx(w, gornoPrzepustowy["G[db]"], label="Wykres naszych pomiarow")
ax1.set_title("Wykres charakterystyki amplitudowej", fontsize=15)
ax1.set_xlabel("$\omega$", fontsize=14)
ax1.set_ylabel("dB", fontsize=14)
ax1.grid()
ax1.legend()

ax2.semilogx(w, phase)
ax2.set_title("Wykres charakterystyki fazowej", fontsize=15)
ax2.grid()
ax2.set_xlabel("$\omega$", fontsize=14)
ax2.set_ylabel("rad", fontsize=14)
plt.show()

```

Charakterystyka Bodego

