

Primeiro Relatório de Lab de Eletronica 1

Henrique da Silva
henrique.pedro@ufpe.br

25 de junho de 2023

Sumário

1 Introdução

2 Análise preliminar

2.1	O circuito	
2.2	Análise simbólica	
2.3	Análise numérica	
2.3.1	Circuito 1	
2.3.2	Circuito 2	
2.3.3	Gráfico dos exemplos	

3 Medições em laboratório

3.1	Circuito 1	
3.1.1	Valores dos componentes	
3.1.2	Frequência de corte	
3.1.3	Valores medidos	
3.1.4	Fotos do osciloscópio	
3.2	Circuito 2	
3.2.1	Valores dos componentes	
3.2.2	Frequência de corte	
3.2.3	Valores medidos	
3.2.4	Fotos do osciloscópio	

4 Análise dos resultados

4.1	Circuito 1	
4.1.1	Ajuste de curva	
4.1.2	Gráfico de Bode	
4.2	Circuito 2	
4.2.1	Ajuste de curva	
4.2.2	Gráfico de Bode	
4.3	Gráfico de Bode de ambos circuitos.	

5 Conclusões

6 Apêndice

7 Anexos

1 Introdução

Neste relatório, vamos discutir e analisar um circuito com amp op, e veremos que ele se comporta como um filtro passa-baixa inversor.

Todos arquivos utilizados para criar este relatório, e o relatório em si estão em: https://github.com/Shapis/ufpe_ee/tree/main/6thsemester/Eletronica1/

O código utilizado para a análise numérica também se encontra no anexo ao final do relatório.

2 Análise preliminar

Utilizarei a biblioteca *sympy* do *Python* para fazer a análise simbólica e numérica do circuito antes de montá-lo fisicamente.

Após terminar as análises compararei os resultados obtidos nas análises numéricas e em laboratório para verificar sua coerência.

2.1 O circuito

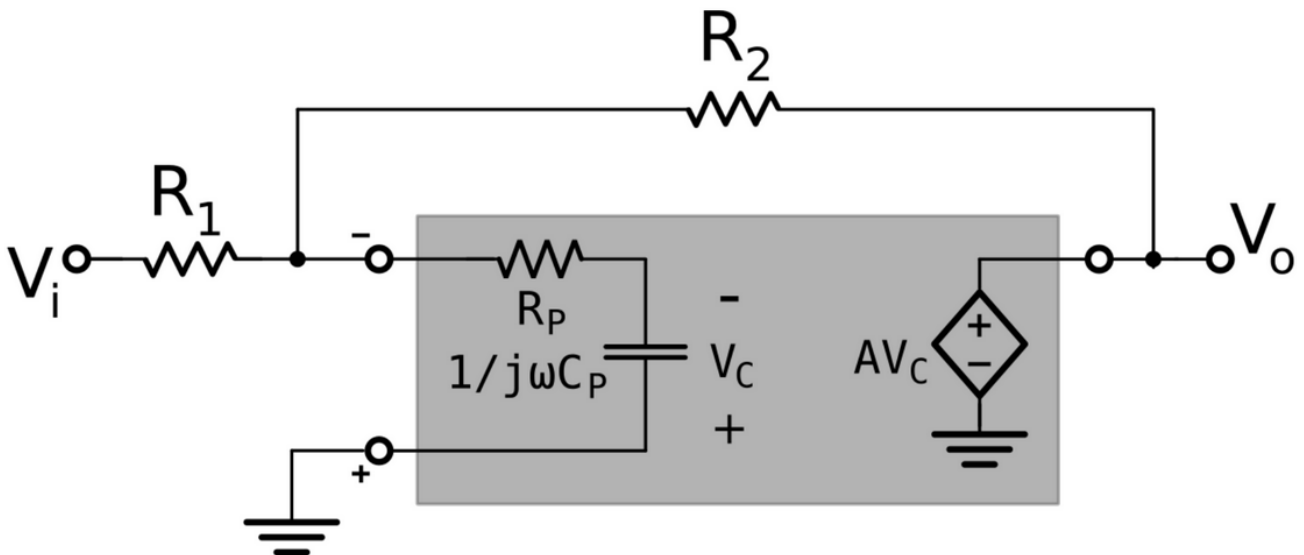


Figura 1: Circuito com amp op em configuração inversora.

2.2 Análise simbólica

Podemos realizar a análise do circuito utilizando análise nodal.

$$\begin{aligned} 0 &= \frac{V_a}{R_p + \frac{1}{C_p jw}} + \frac{V_a - V_o}{R_2} + \frac{V_a - V_i}{R_1} \\ -V_c &= \frac{V_a}{C_p jw \left(R_p + \frac{1}{C_p jw} \right)} \\ V_o &= AV_c \end{aligned} \quad (1)$$

Resolvendo as equações para V_o obtemos que V_o é dado por:

$$-\frac{AR_2 V_i}{AR_1 + C_p R_1 R_2 jw + C_p R_1 R_p jw + C_p R_2 R_p jw + R_1 + R_2} \quad (2)$$

Aqui fazemos a seguinte simplificação $R_p \gg R_1$, $R_p \gg R_2$, e $A \gg 1$:

$$V_o = -\frac{AR_2 V_i}{AR_1 + C_p R_p jw (R_1 + R_2)} \quad (3)$$

Com esta simplificação fazemos $\frac{V_o}{V_i}$ para obter a função transferência $H(jw)$:

$$H(jw) = -\frac{AR_2}{AR_1 + C_p R_p jw (R_1 + R_2)} \quad (4)$$

Agora podemos reorganiza-la no formato de um filtro passa-baixa para achar o ganho K , e a frequência de corte w_c :

$$\begin{aligned} H(jw) &= -\frac{K w_c}{jw + w_c} \\ w_p &= \frac{1}{R_p C_p} \\ K &= \frac{R_2}{R_1} \\ w_c &= \frac{A w_p}{1 + K} \end{aligned} \quad (5)$$

Podemos também achar o valor absoluto de $H(jw)$:

$$|H(jw)| = \frac{|K w_c|}{\sqrt{w^2 + w_c^2}} \quad (6)$$

2.3 Análise numérica

Aqui utilizaremos as equações (5) e (6) para implementar o circuito discutido acima com dois conjuntos de valores, os conjuntos diferem apenas em seu R_2

Para ambos circuitos utilizaremos os seguintes valores:

$$\begin{aligned} R_1 &= 4.7k\Omega \\ w_p &= 2\pi 1k\text{rad/s} \\ A &= 10^5 \end{aligned} \quad (7)$$

2.3.1 Circuito 1

Neste utilizaremos o valor $R_2 = 22k$:

Isto nos dá:

$$\begin{aligned} K &= 4.68 \\ w_c &= 1.10 \times 10^6 \text{rad/s} \end{aligned} \tag{8}$$

freq rad/s	freq Hz	$ H(jw) $
$0.5w_c$	$88014.98Hz$	4.19
w_c	$176029.96Hz$	3.31
$2w_c$	$352059.93Hz$	2.09
$4w_c$	$704119.85Hz$	1.14
$10w_c$	$1760299.63Hz$	0.47
$20w_c$	$3520599.25Hz$	0.23
$40w_c$	$7041198.50Hz$	0.12

2.3.2 Circuito 2

Neste utilizaremos o valor $R_2 = 560k$:

Isto nos dá:

$$\begin{aligned} K &= 119.15 \\ w_c &= 52295 \text{rad/s} \end{aligned} \tag{9}$$

freq rad/s	freq Hz	$ H(jw) $
$0.5w_c$	$4161.50Hz$	106.57
w_c	$8323Hz$	84.25
$5w_c$	$41615Hz$	23.37
$20w_c$	$166460Hz$	5.95
$50w_c$	$416150Hz$	2.38
$200w_c$	$1664600Hz$	0.60
$500w_c$	$4161501Hz$	0.24
$1000w_c$	$8323003Hz$	0.12

2.3.3 Gráfico dos exemplos

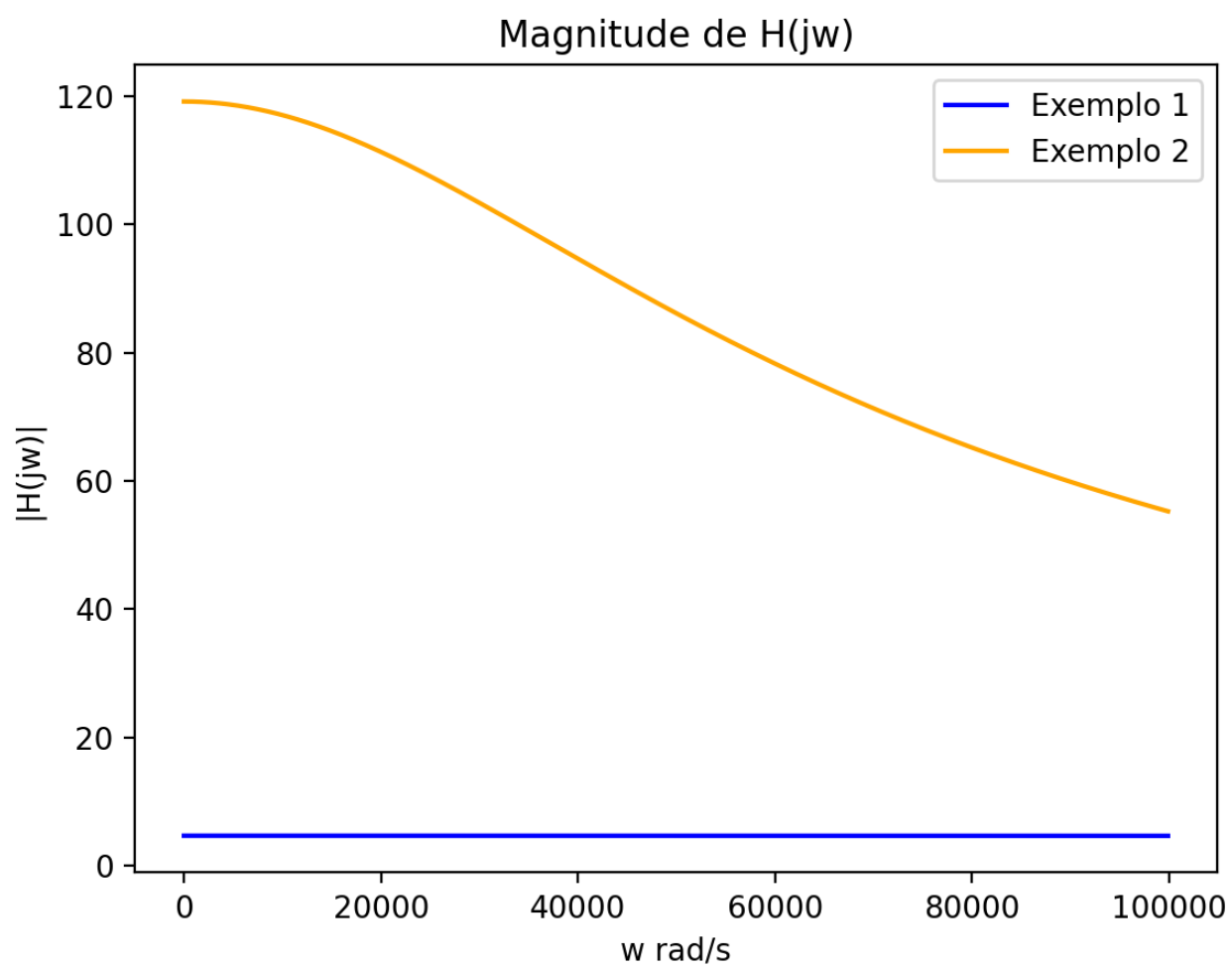


Figura 2: Gráfico da magnitude pela frequência dos exemplos 1 e 2.

3 Medições em laboratório

Montaremos os dois circuitos discutidos acima em laboratório, e mediremos a tensão de entrada e saída para várias frequências, e com isto obteremos a magnitude da função transferência para frequências diversas.

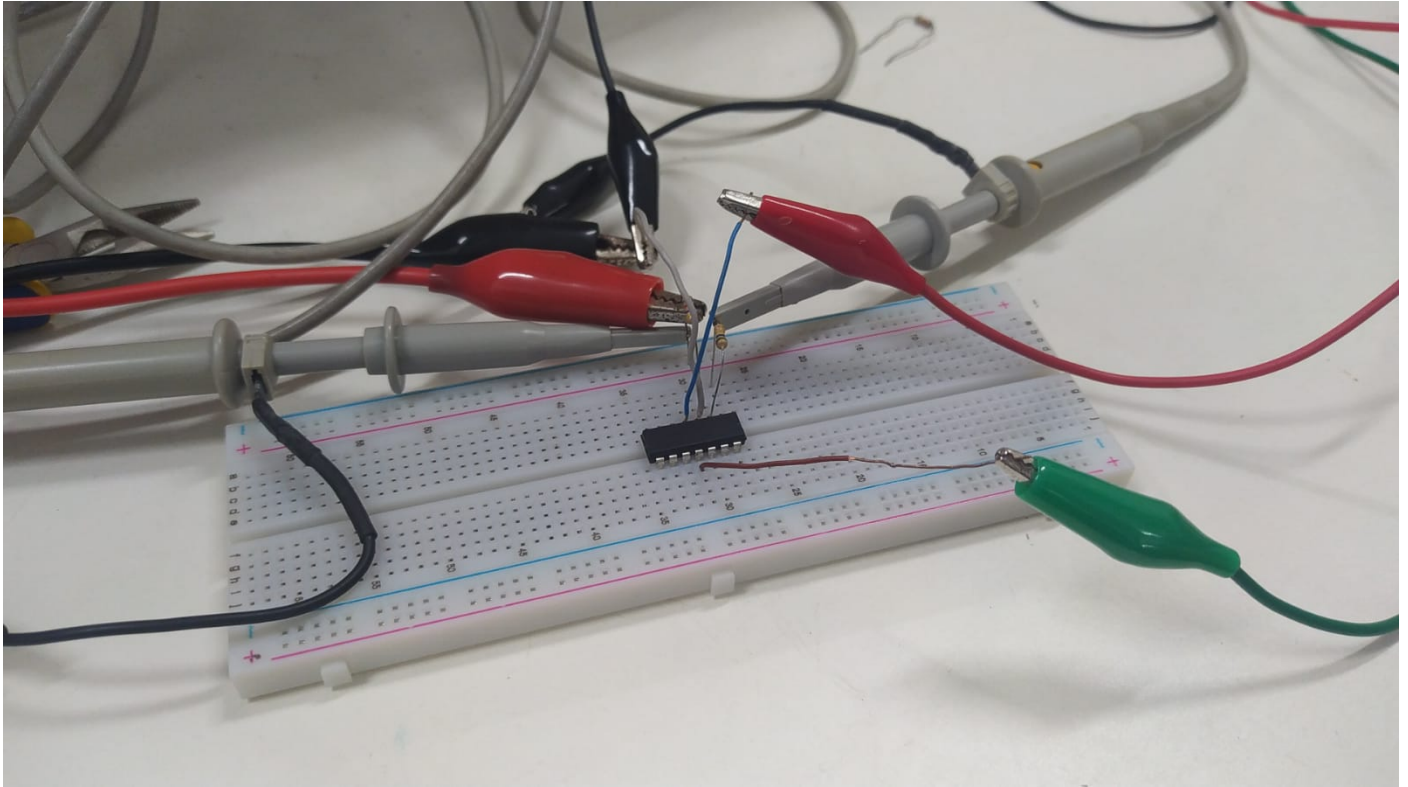


Figura 3: Foto do circuito montado em laboratório.

3.1 Circuito 1

3.1.1 Valores dos componentes

$$\begin{aligned} R_1 &= 4.65k\Omega \\ R_2 &= 21.9k\Omega \end{aligned} \tag{10}$$

3.1.2 Frequência de corte

Identificamos a frequência de corte como sendo $f_c = 210kHz$.

3.1.3 Valores medidos

Amplitudes	V_i	V_o	$ H(\omega) $
0.002F0	0.2975	1.404	4.7193277310924
0.01F0	0.297	1.405	4.7306397306397
0.05F0	0.298	1.403	4.7080536912752
0.2F0	0.297	1.393	4.6902356902357
0.5F0	0.298	1.306	4.3825503355705
0.8F0	0.301	1.124	3.734219269103
F0	0.297	0.975	3.2828282828283
2F0	0.297	0.54	1.8181818181818
4F0	0.3	0.276	0.92
10F0	0.302	0.114	0.3774834437086
20F0	0.301	0.057	0.1893687707641
40F0	0.315	0.031	0.0984126984127

Figura 4: Tabela de magnitude para uma gama de valores de frequência.

3.1.4 Fotos do osciloscópio

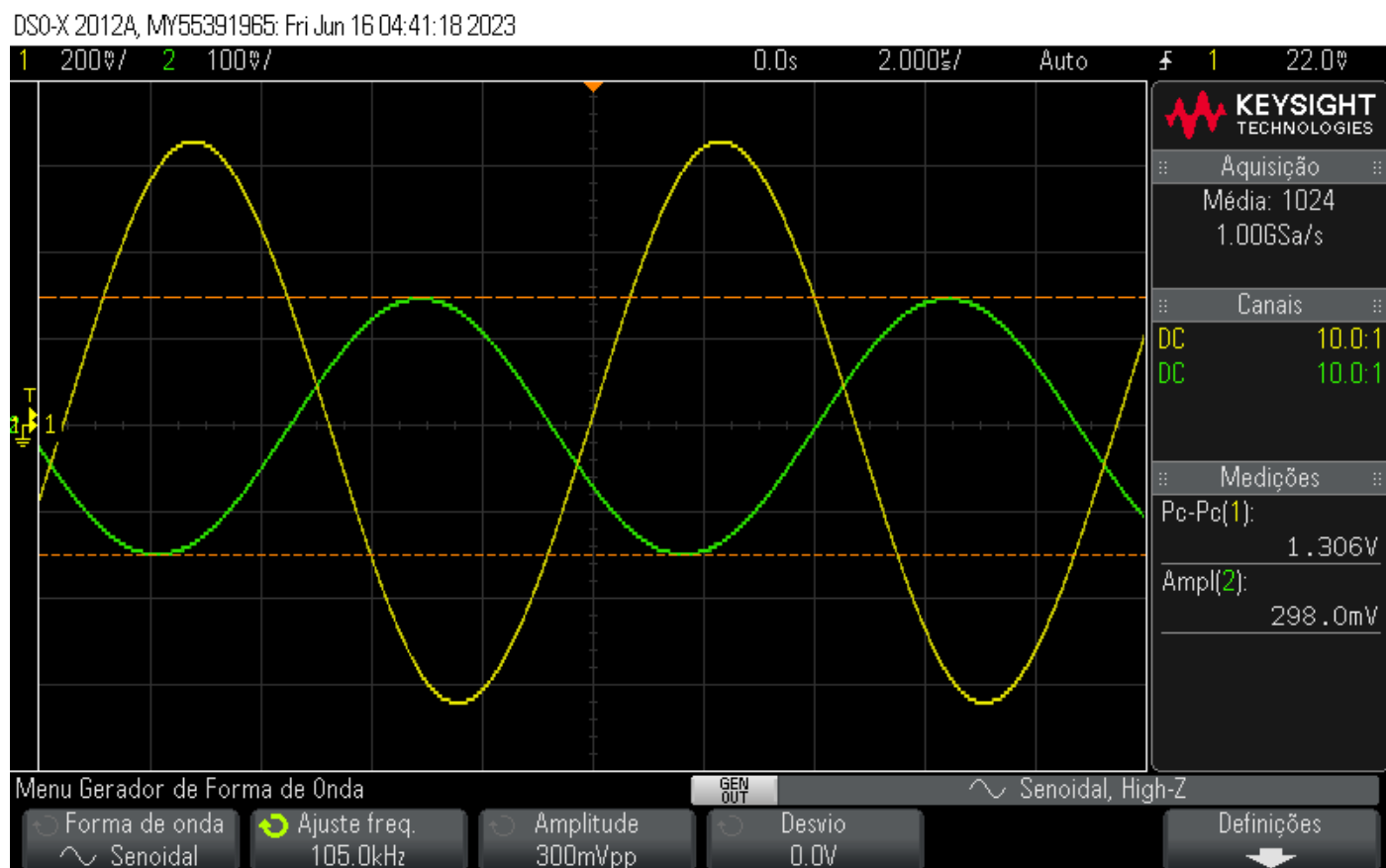


Figura 5: Imagem da onda no osciloscópio para $0.5 f_c$.

DSO-X 2012A, MY55391965: Fri Jun 16 04:43:00 2023

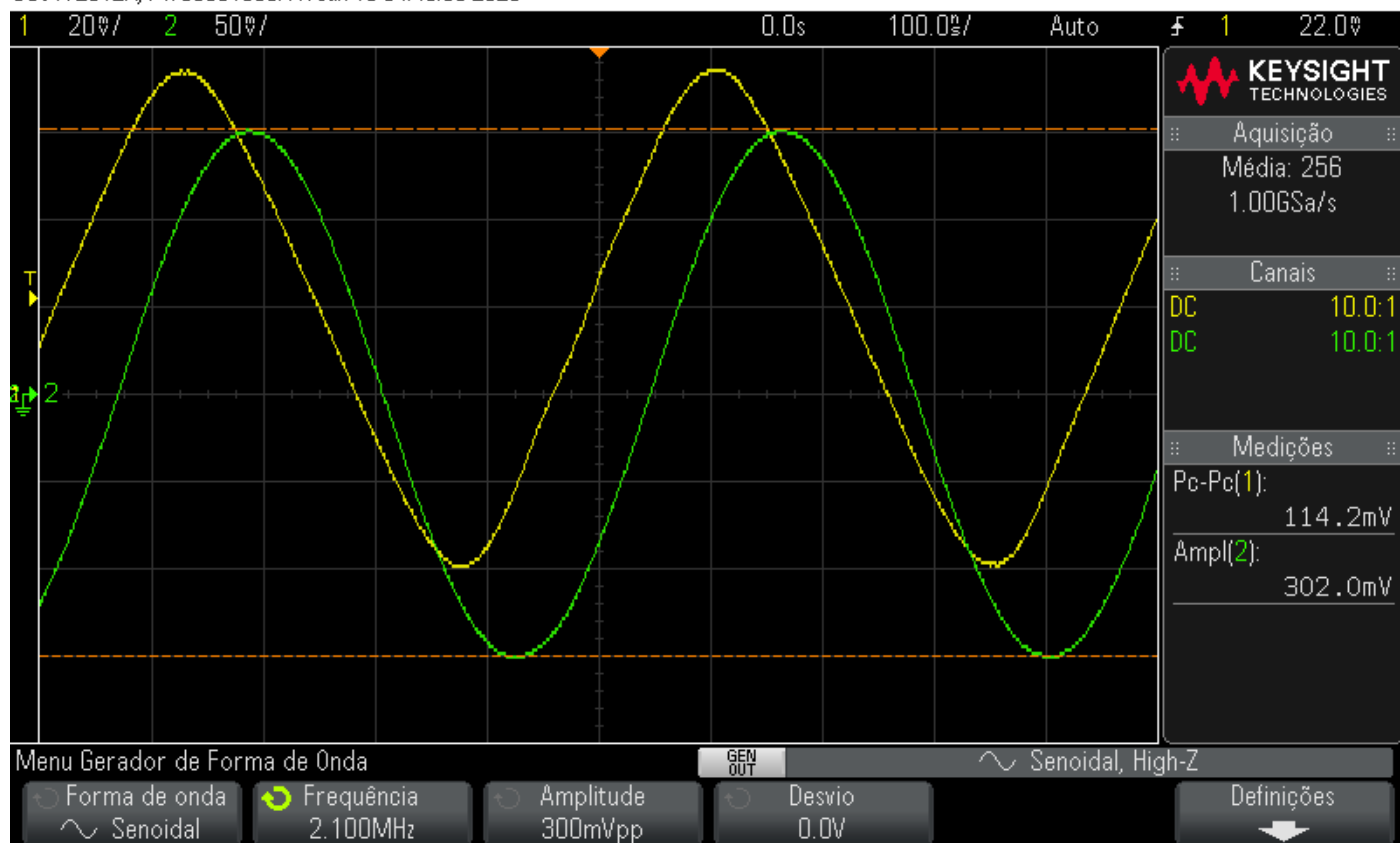


Figura 6: Imagem da onda no osciloscópio para $10 f_c$.

DSO-X 2012A, MY55391965: Fri Jun 16 04:43:23 2023

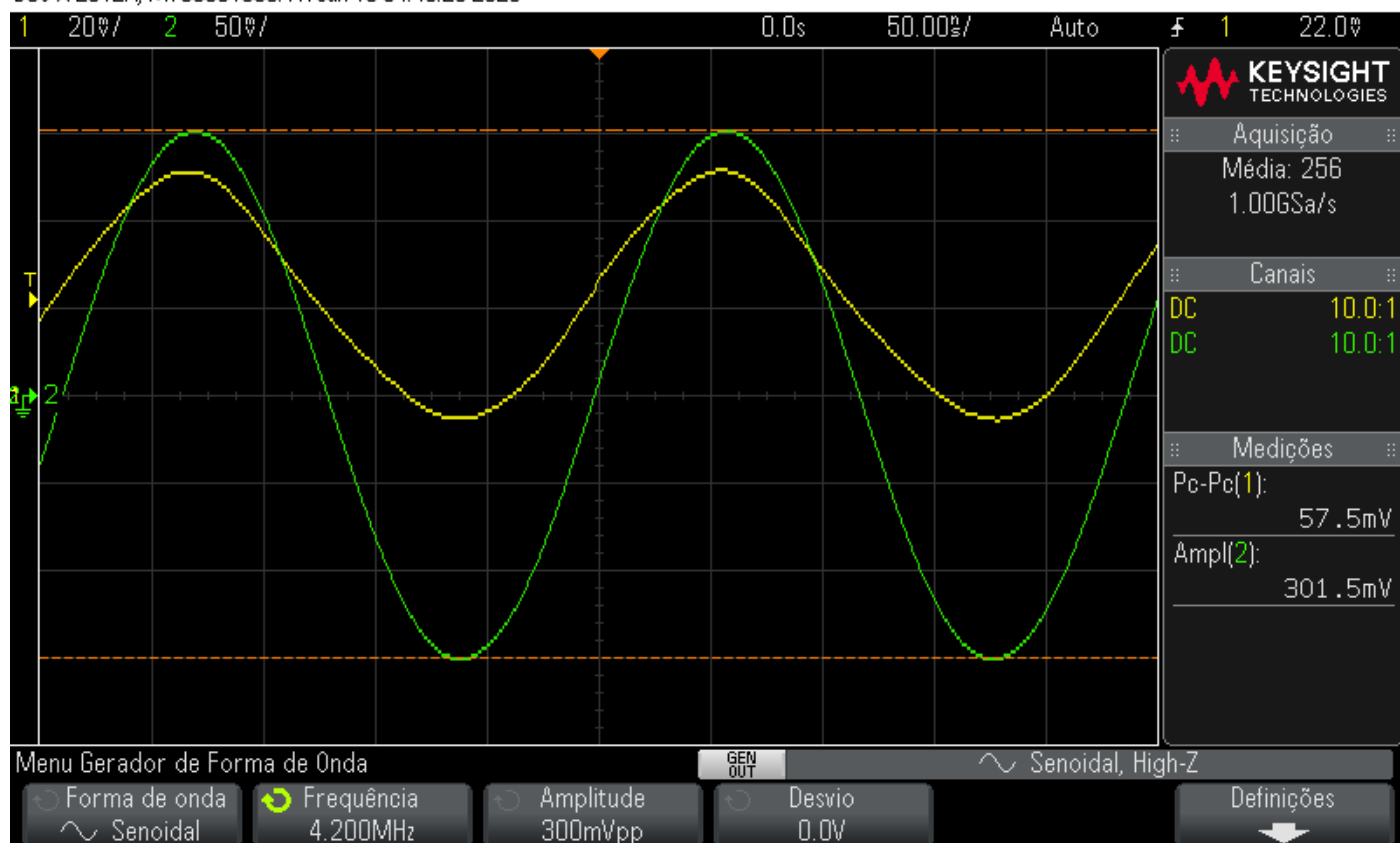


Figura 7: Imagem da onda no osciloscópio para $20 f_c$.

3.2 Circuito 2

3.2.1 Valores dos componentes

$$\begin{aligned}R_1 &= 4.65k\Omega \\ R_2 &= 550k\Omega\end{aligned}\tag{11}$$

3.2.2 Frequência de corte

Identificamos a frequência de corte como sendo $f_c = 11800kHz$.

3.2.3 Valores medidos

Amplitudes	Vi	Vo	H(ω)
0.05F0	0.1191	14.01	117.6322418136
0.1F0	0.1199	13.95	116.3469557965
0.2F0	0.12	13.77	114.75
0.5F0	0.121	12.61	104.21487603306
0.8F0	0.122	10.85	88.934426229508
F0	0.122	9.8	80.327868852459
2F0	0.1218	5.92	48.604269293925
5F0	0.123	2.548	20.715447154472
20F0	0.123	0.645	5.2439024390244
50F0	0.121	0.258	2.1322314049587
200F0	0.121	0.0652	0.5388429752066
500F0	0.125	0.0205	0.164
1000F0	0.151	0.0122	0.0807947019868

Figura 8: Tabela de magnitude para uma gama de valores de frequência.

3.2.4 Fotos do osciloscópio

DSO-X 2012A, MY55391965: Fri Jun 16 05:06:22 2023

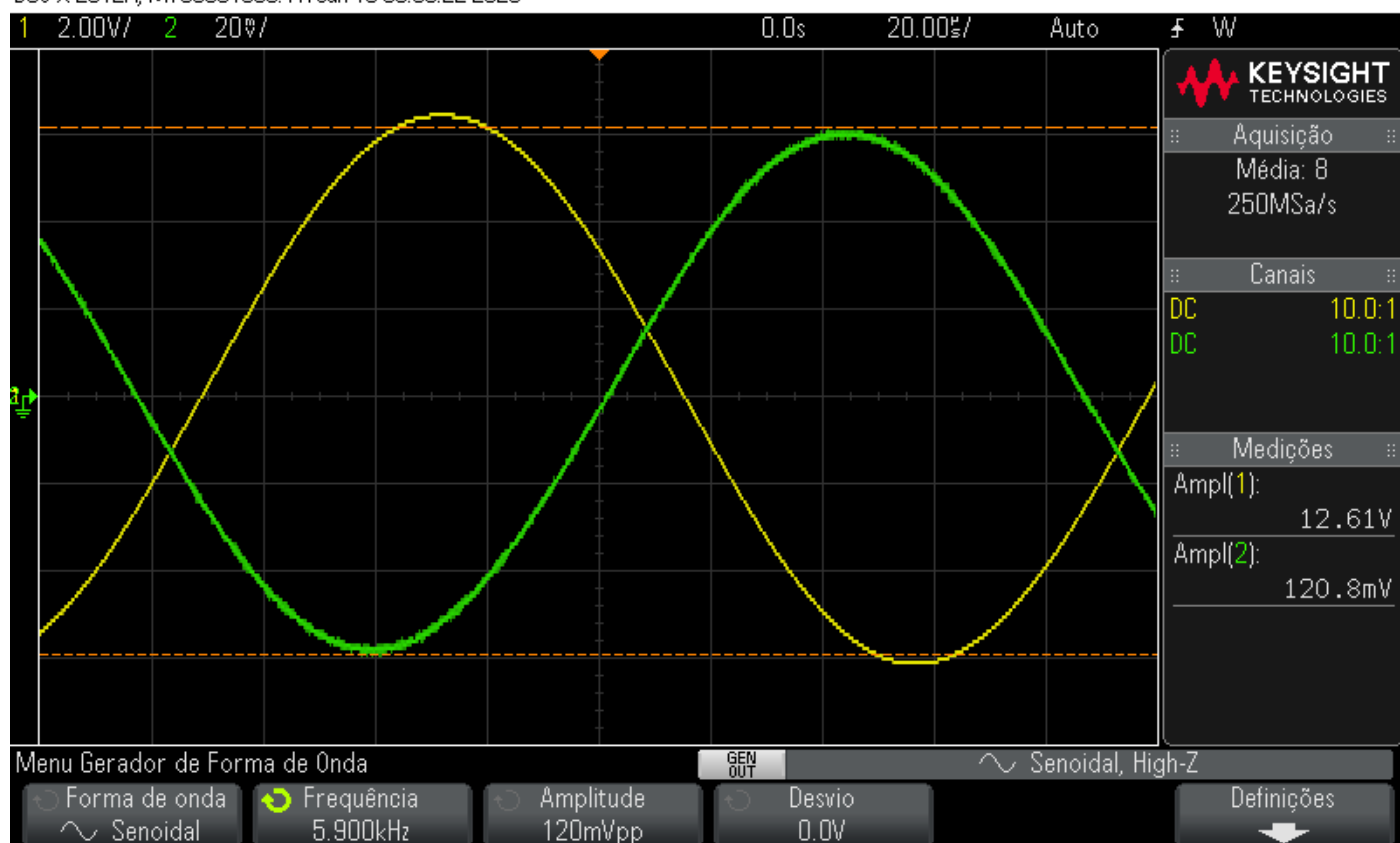


Figura 9: Imagem da onda no osciloscópio para $0.5 f_c$.

DSO-X 2012A, MY55391965: Fri Jun 16 05:10:32 2023

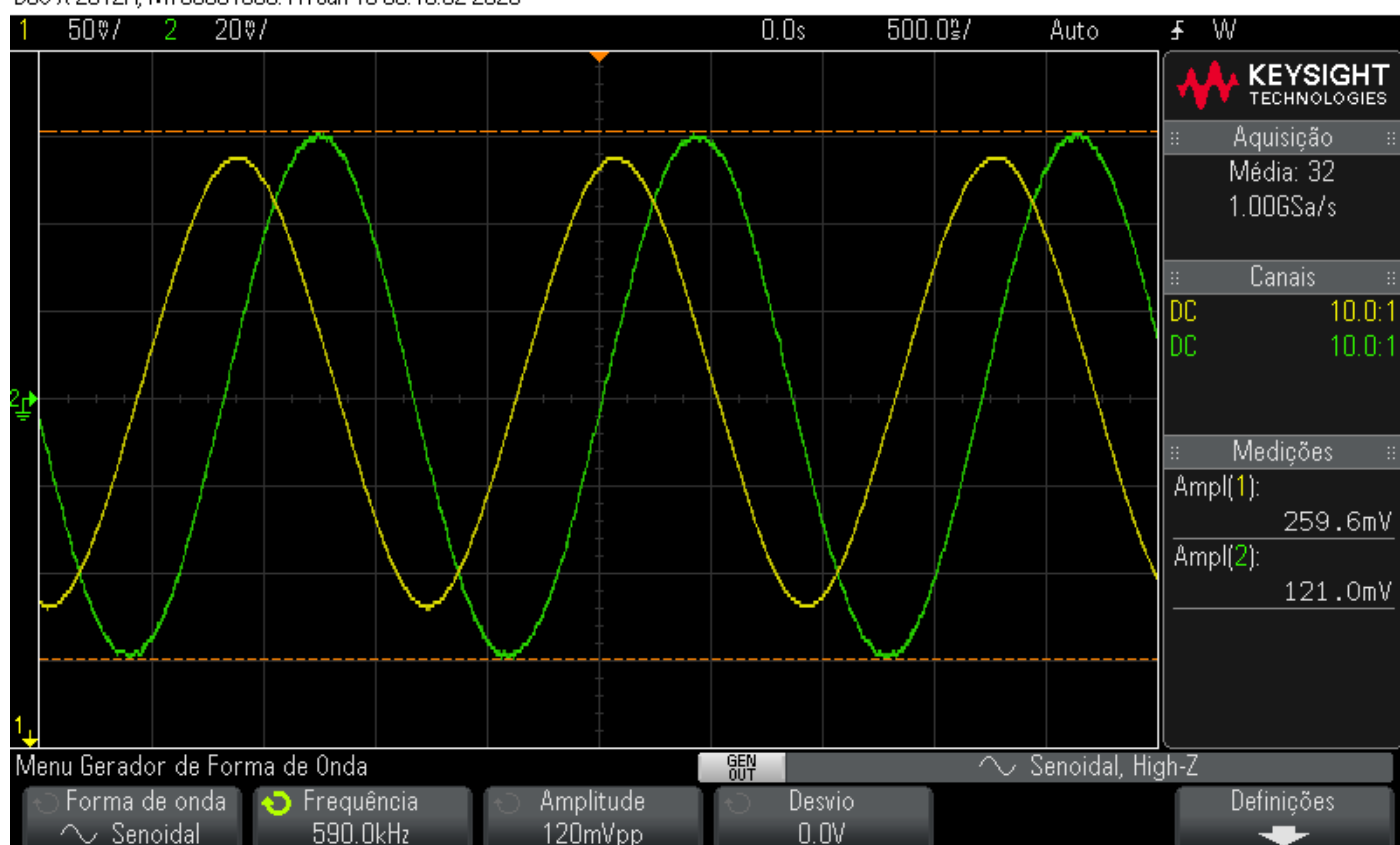


Figura 10: Imagem da onda no osciloscópio para $50 f_c$.

DSO-X 2012A, MY55391965: Fri Jun 16 05:12:54 2023

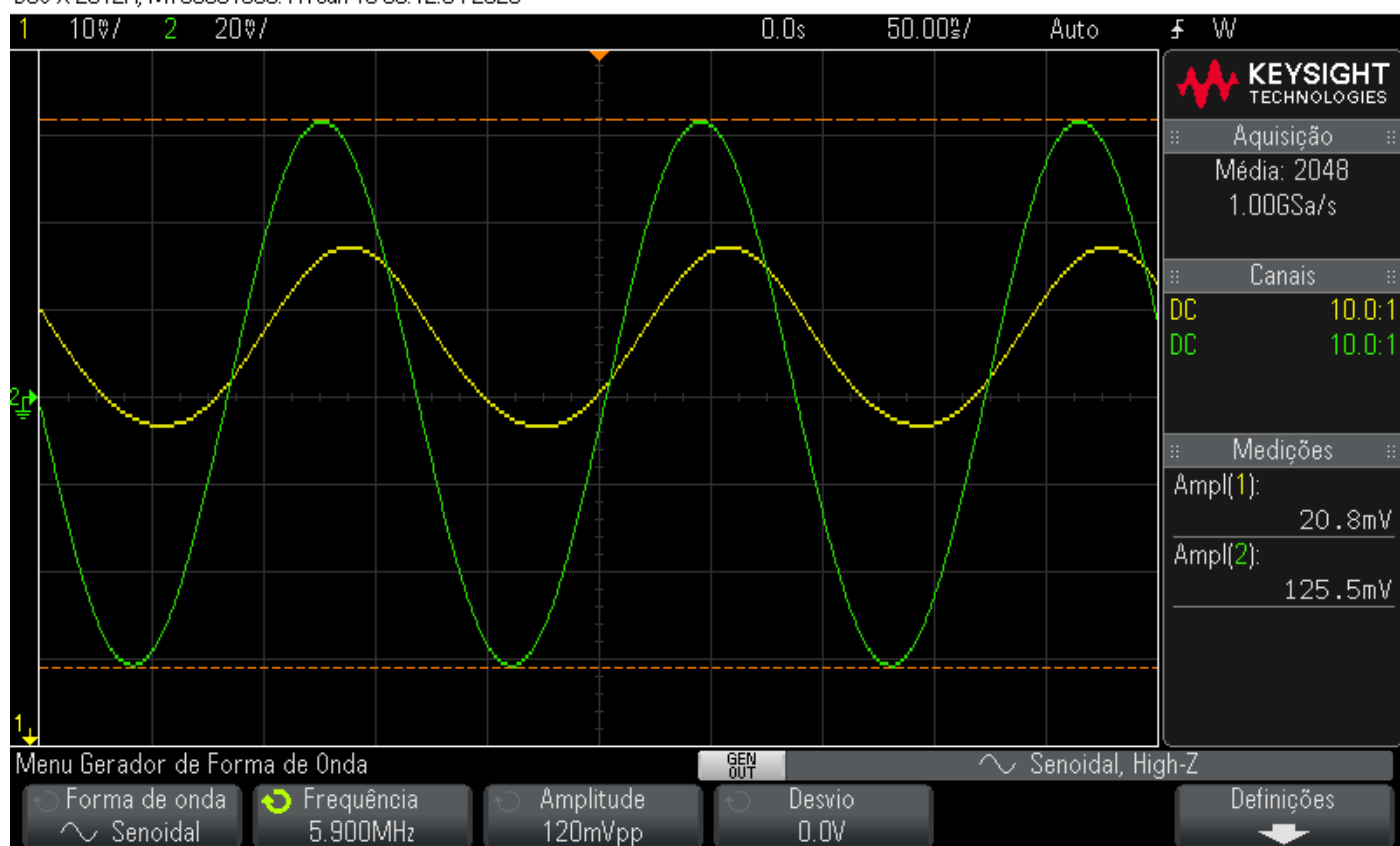


Figura 11: Imagem da onda no osciloscópio para $500 f_c$.

4 Análise dos resultados

Obtivemos as magnitudes de $H(jw)$ dos exemplos nas figuras (3) e (6), as utilizarei no algoritmo de ajuste de curva que foi disponibilizado no Classroom.

4.1 Circuito 1

4.1.1 Ajuste de curva

Pelo código de ajuste de curva obtivemos os seguintes valores:

$$\begin{aligned} K &= -4.9 \\ f_c &= 168452.6 \text{ Hz} = 1.06 * 10^6 \text{ rad/s} \end{aligned} \tag{12}$$

Que são coerentes e próximos com os valores achados anteriormente na equação (8).

4.1.2 Gráfico de Bode

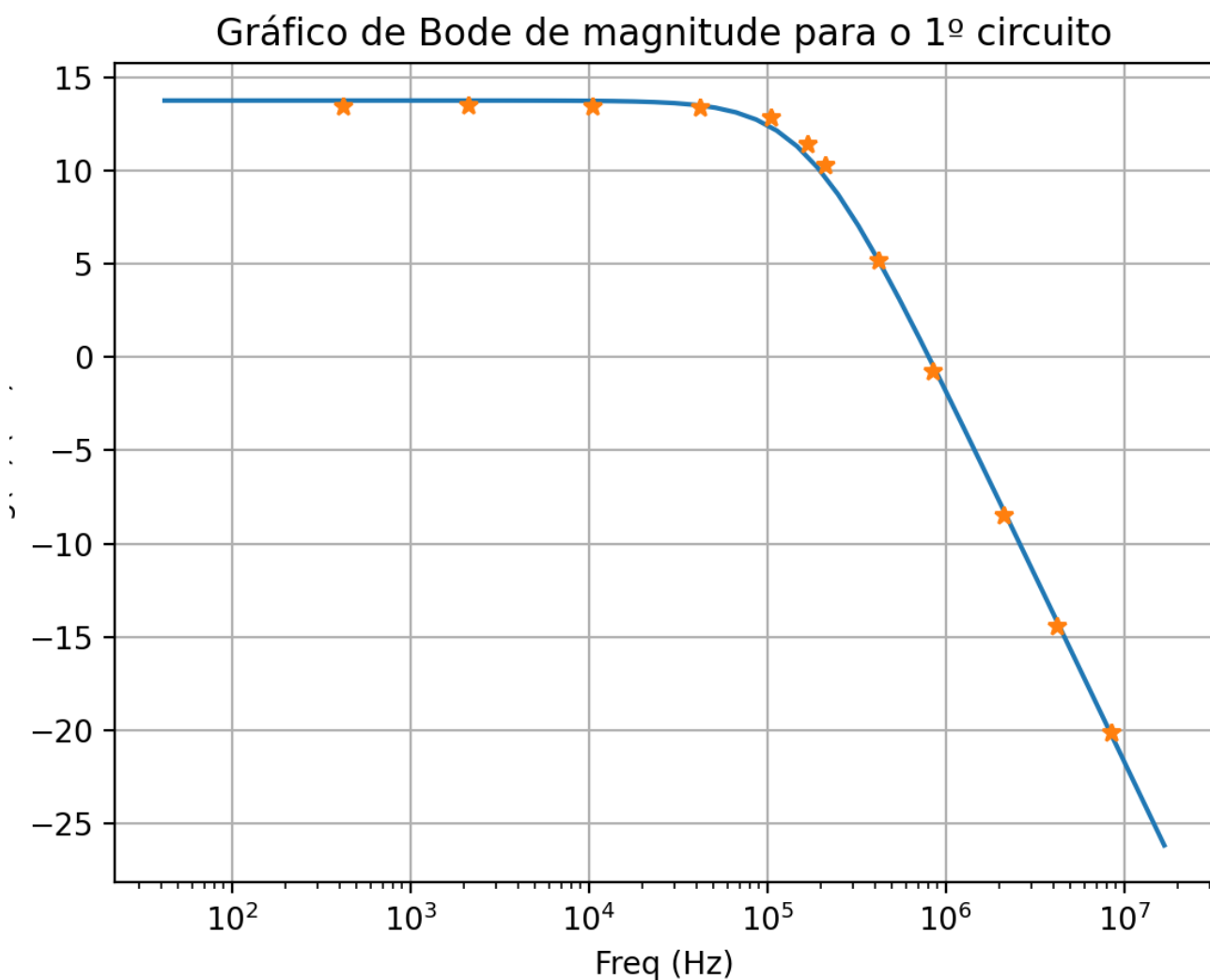


Figura 12: Gráfico de Bode para o primeiro circuito.

4.2 Circuito 2

4.2.1 Ajuste de curva

Pelo código de ajuste de curva obtivemos os seguintes valores:

$$\begin{aligned} K &= -120.1 \\ f_c &= 9701.2\text{Hz} = 60954\text{rad/s} \end{aligned} \tag{13}$$

Que são coerentes e próximos com os valores achados anteriormente na equação (9).

4.2.2 Gráfico de Bode

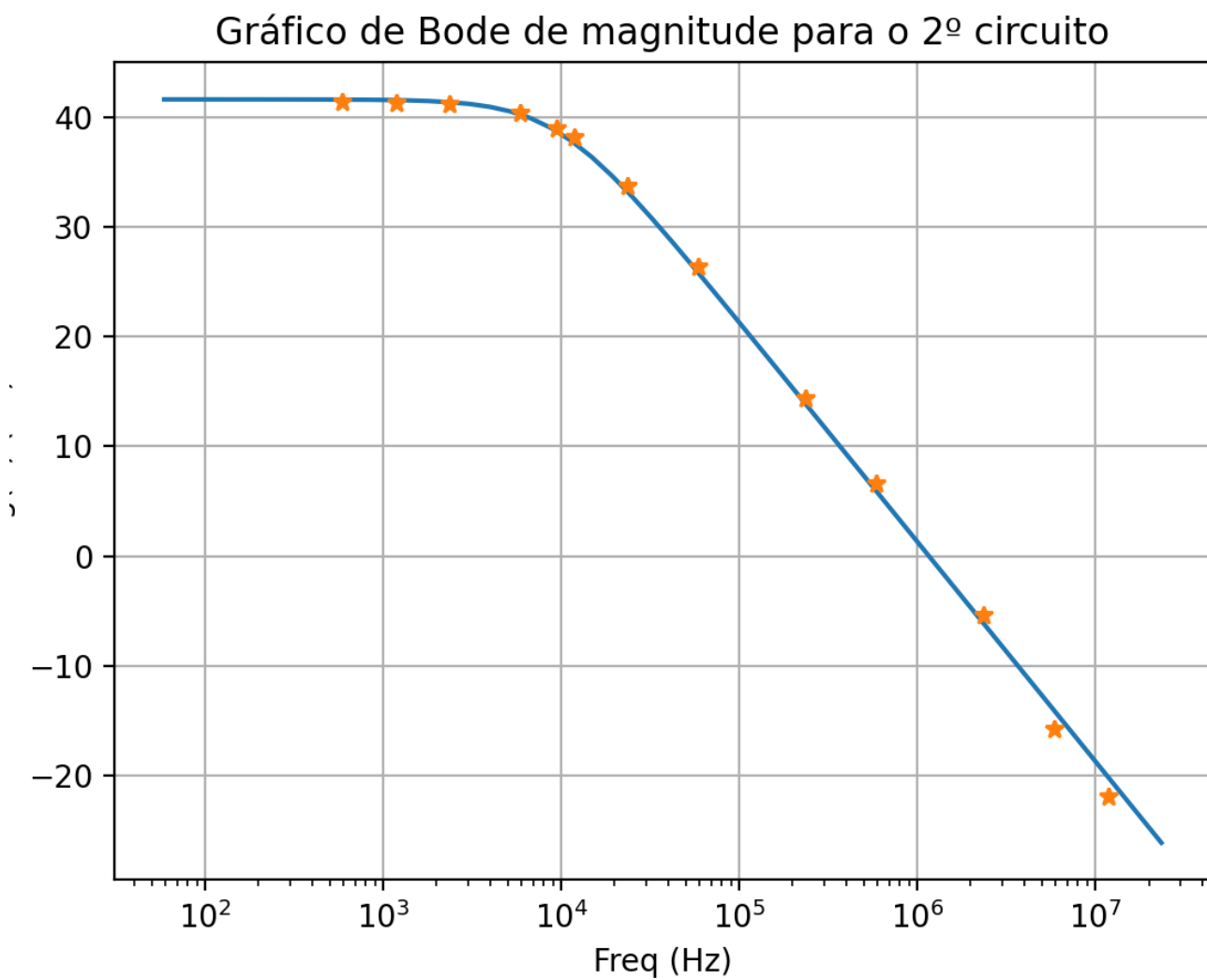


Figura 13: Gráfico de Bode para o segundo circuito.

4.3 Gráfico de Bode de ambos circuitos.

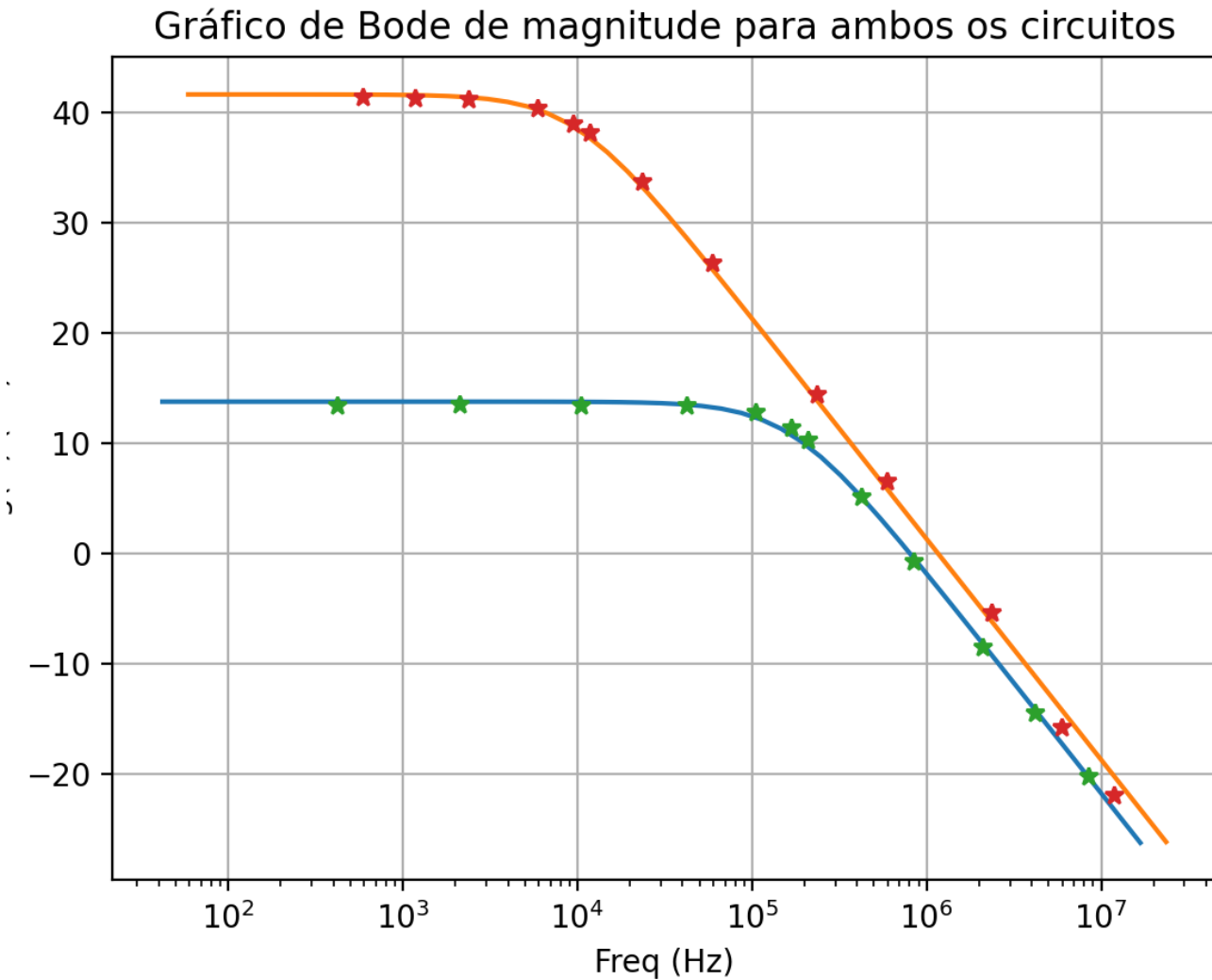


Figura 14: Gráfico de Bode para ambos circuitos sobrepostos.

5 Conclusões

Conseguimos com sucesso fazer a análise simbólica e numérica com a biblioteca sympy do Python, e comparamos os resultados com os obtidos experimentalmente.

Nos resultados práticos, obtivemos parâmetros K e w_c bastante similares aos obtidos numericamente.

Observamos que ambos circuitos se comportam como filtros passa-baixa inversores de primeira ordem, que era esperado devido a forma da equação que encontramos na análise preliminar.

Em suma creio que tivemos sucesso em nos familiarizar com as ferramentas de análise de circuitos eletrônicos, e métodos para análise numérica e simbólica.

6 Apêndice

Abaixo se encontra o código utilizado para a análise simbólica e numérica do circuito.

```
import matplotlib.pyplot as plt
import sympy as smp
from sympy import *

# Definindo as variaveis simbolicas
Vo, Vi, Va, Vc, R1, R2, Rp, Cp, A, w, j, Hwj, wp, wc, K = smp.symbols(
    'V_o V_i V_a V_c R_1 R_2 R_p C_p A w j H_jw w_p, w_c, K', real=True)

# Analise nodal do circuito

eq1 = smp.Eq((Va - Vi)/R1 + (Va - Vo)/R2 + Va/(Rp + (1/(j * w * Cp))), 0)
eq2 = smp.Eq(Va * ((1/(j * w * Cp)))/(((1/(j * w * Cp)) + Rp), -Vc)
eq3 = smp.Eq(A * Vc, Vo)
# print('Equacoes em latex:')
# smp.pprint(smp.latex(eq1))
# smp.pprint(smp.latex(eq2))
# smp.pprint(smp.latex(eq3))
# print("")

print("Equacoes do circuito:")
print("Equacao 1:")
smp.pprint(eq1)
print("Equacao 2:")
smp.pprint(eq2)
print("Equacao 3:")
smp.pprint(eq3)
print("")

sols = smp.solve([eq1, eq2, eq3], [Va, Vc, Vo])

print("Solucao para Vo:")
smp.pprint(sols[Vo])
# print("")
# smp.pprint(smp.latex(sols[Vo]))
print("")

print("Aqui fazemos a seguinte simplificacao:")
print("Rp >> R1 , Rp >> R2, e A >> 1")

Vo = (-A * R2 * Vi)/((Rp*(R1 + R2))*j*w*Cp + A*R1)

print("Equacao de Vo simplificada:")
smp.pprint(Vo)
# smp.pprint(smp.latex(Vo))
print("")

eqHwj = smp.Eq(Hwj, Vo/Vi)

print("Equacao de Hwj:")
smp.pprint(eqHwj)
print("")

print("Resolvendo a equacao Hwj e colocando no formato canonico da um filtro
passa baixa  $(-K \omega_p / (j\omega + \omega_p))$ 
obtemos o seguinte:")

eqHwj = smp.Eq(Hwj, -(K * wc) / (I * w + wc))

Hwj = -(K * wc) / (I * w + wc)
```

```

smp.pprint(eqHjw)

print("")

eqwp = smp.Eq(wp, 1/(Rp*Cp))
smp.pprint(eqwp)

eqK = smp.Eq(K, R2/R1)
smp.pprint(eqK)

eqwc = smp.Eq(wc, (A*wp)/(1 + K))
smp.pprint(eqwc)

print("")

# Hjw = Hjw.subs({K: R2/R1, wc: A*wp/(1+K)})

print("Valor absoluto de Hjw:")
Hjw_abs = smp.Abs(Hjw)
smp.pprint(Hjw_abs)
# smp.pprint(smp.latex(Hjw_abs))
print("")

print("Exemplo 1:")
print("Para R1 = 4.7E3 ohms, R2 = 2.2E4 ohms, wp = 2E1 pi e A = 1E5\n")

eqK1 = eqK.subs({R1: 4.7E3, R2: 2.2E4})
K1 = smp.solve(eqK1, K)[0]
smp.pprint(eqK1)
print("")

eqwc1 = eqwc.subs({A: 1E5, wp: 20 * smp.pi, K: K1}).evalf()
wc1 = smp.solve(eqwc1, wc)[0]
smp.pprint(eqwc1)
print("")

print("Valores absolutos para w = [0.5, 1, 2, 4, 10, 20, 40]*wc:")

# ex1interval = [0.02, 0.01, 0.05, 0.2, 0.5, 1, 2, 4, 10, 20, 40]
ex1interval = [0.5, 1, 2, 4, 10, 20, 40]

ex1Vals = []

for val in ex1interval:
    temp = Hjw_abs.subs({w: wc1 * val, wc: wc1, K: K1})
    ex1Vals.append(temp)
    print('para freq:', round(((val*wc1)/(2*smp.pi)).evalf(), 2),
          '\ttemos:', round(temp, 2))

print("")

print("Exemplo 2:")
print("Para R1 = 4.7E3 ohms, R2 = 5.6E5 ohms, wp = 2E1 pi e A = 1E5\n")

eqK2 = eqK.subs({R1: 4.7E3, R2: 5.6E5})
K2 = smp.solve(eqK2, K)[0]
smp.pprint(eqK2)
print("")

eqwc2 = eqwc.subs({A: 1E5, wp: 20 * smp.pi, K: K2}).evalf()
wc2 = smp.solve(eqwc2, wc)[0]
smp.pprint(eqwc2)
print("")

```



```

ex2interval = [0.5, 1, 5, 20, 50, 200, 500, 1000]
ex2Vals = []

for val in ex2interval:
    temp = Hjwt_abs.subs({w: wc2 * val, wc: wc2, K: K2})
    ex2Vals.append(temp)
    print('para freq:', round(((val*wc2)/(2*smp.pi)).evalf(), 2),
          '\ttemos:', round(temp, 2))
    print("")

# Plotando os graficos

frequencias_plot = [i for i in range(1, 100000, 100)]

plotH1 = [Hjwt_abs.subs({w: i, wc: wc1, K: K1}) for i in frequencias_plot]
plotH2 = [Hjwt_abs.subs({w: i, wc: wc2, K: K2}) for i in frequencias_plot]

fig, ax = plt.subplots()

ax.plot(frequencias_plot, plotH1, color='blue', label='Exemplo 1')
ax.plot(frequencias_plot, plotH2, color='orange', label='Exemplo 2')
ax.legend(['Exemplo 1', 'Exemplo 2'])
plt.xlabel('w rad/s')
plt.ylabel('|H(jw)|')
plt.title('Magnitude de H(jw)')

plt.show()

# Medicoes na pratica

# Exemplo1

```

7 Anexos

Código utilizado para geração de gráficos de bode, e análise utilizando as frequências de cortes obtidas experimentalmente.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit

fc1 = 210000
fc2 = 11800

freqs1 = np.array([0.002*fc1, 0.01*fc1, 0.05*fc1, 0.2*fc1, 0.5*fc1,
                    0.8*fc1, fc1, 2*fc1, 4*fc1, 10*fc1, 20*fc1, 40*fc1])
vin1 = np.array([0.2975, 0.297, 0.298, 0.297, 0.298,
                  0.301, 0.297, 0.297, 0.3, 0.302, 0.301, 0.315])
vout1 = np.array([1.404, 1.405, 1.403, 1.393, 1.306, 1.124,
                  0.975, 0.54, 0.276, 0.114, 0.057, 0.031])

freqs2 = np.array([0.05*fc2, 0.1*fc2, 0.2*fc2, 0.5*fc2, 0.8*fc2, fc2,
                    2*fc2, 5*fc2, 20*fc2, 50*fc2, 200*fc2, 500*fc2, 1000*fc2])
vin2 = np.array([0.1191, 0.1199, 0.12, 0.121, 0.122, 0.122,
                  0.1218, 0.123, 0.123, 0.121, 0.121, 0.125, 0.151])
vout2 = np.array([14.01, 13.95, 13.77, 12.61, 10.85, 9.8, 5.92,
                  2.548, 0.645, 0.258, 0.0652, 0.0205, 0.0122])

def mag_sqr_fun(f, K, fc):
    return (K*fc)**2/(f**2 + fc**2)

def dB(m):
    return 20*np.log10(m)

(K1, fc1), _ = curve_fit(lambda f, K, fc: dB(mag_sqr_fun(f, K, fc)),
                        freqs1, 2*dB(vout1/vin1))

print(f"""O ganho K eh {round(K1, 1)}
A frequencia de corte eh {round(fc1, 1)} Hz""")

f1 = np.logspace(np.log10(freqs1[0]) - 1, np.log10(freqs1[-1]) + 0.3)
mag1 = dB(mag_sqr_fun(f1, K1, fc1))/2

plt.semilogx(f1, mag1)
plt.semilogx(freqs1, dB(vout1/vin1), "*")
plt.xlabel("Freq (Hz)")
plt.ylabel("Mag(H) (dB)")
plt.title("Grafico de Bode de magnitude para o 1 circuito")
plt.grid()
plt.savefig("figura1.png")
plt.show()
```

```

(K2, fc2), _ = curve_fit(lambda f, K, fc: dB(mag_sqr_fun(f, K, fc)),
freqs2, 2*dB(vout2/vin2))

print("\n\nResultados para o 2 circuito:\n")
print(f""O ganho K eh {round(K2, 1)}
A frequencia de corte eh {round(fc2, 1)} Hz""")

f2 = np.logspace(np.log10(freqs2[0]) - 1, np.log10(freqs2[-1]) + 0.3)
mag2 = dB(mag_sqr_fun(f2, K2, fc2))/2

plt.semilogx(f2, mag2)
plt.semilogx(freqs2, dB(vout2/vin2), "*")
plt.xlabel("Freq (Hz)")
plt.ylabel("Mag(H) (dB)")
plt.title("Grafico de Bode de magnitude para o 2 circuito")
plt.grid()
plt.savefig("figura2.png")
plt.show()


plt.semilogx(f1, mag1)
plt.semilogx(f2, mag2)
plt.semilogx(freqs1, dB(vout1/vin1), "*")
plt.semilogx(freqs2, dB(vout2/vin2), "*")
plt.xlabel("Freq (Hz)")
plt.ylabel("Mag(H) (dB)")
plt.title("Grafico de Bode de magnitude para ambos os circuitos")
plt.grid()
plt.savefig("figura3.png")
plt.show()

```