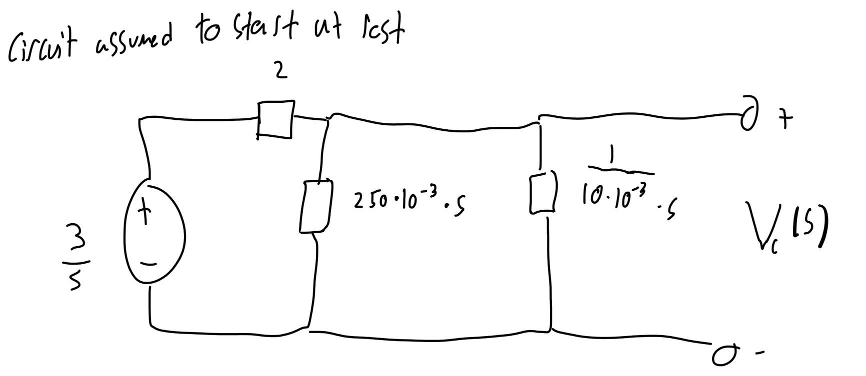
Lab 9

Tuesday, November 26, 2024 6:37 PM

1)

Circuit assumed to start at rest



$$Z_{eq} = 2 + \left(\frac{s}{4} \,\|\, \frac{100}{s}\right)$$

Voltage division: $\dfrac{\left(\frac{s}{4} \,\|\, \frac{100}{s}\right)}{2+\left(\frac{s}{4} \,\|\, \frac{100}{s}\right)} \cdot \dfrac{3}{s} = V_c(s)$

$$V_c(s) = \frac{3}{s} \frac{\left(\frac{4}{s}+\frac{s}{100}\right)^{-1}}{2+\left(\frac{4}{s}+\frac{s}{100}\right)^{-1}} = \frac{3}{s} \frac{1}{\left(\frac{4}{s}+\frac{s}{100}\right)\left(2+\frac{1}{4/s+s/100}\right)} = \frac{1}{\frac{8}{s}+\frac{2s}{100}+1} \frac{3}{s} = \frac{3}{\frac{s^2}{50}+s+8} = \frac{150}{s^2+50s+400}$$

$$\mathcal{L}^{-1}\{V_c(s)\} = u(t)\left[5e^{-10t} - 5e^{-40t}\right] = V_c(t)$$

2)

$$V_{out} = \underset{\text{Starting } V}{V_{test}} - \underset{\text{drop over } R_3}{I_{test} \cdot R_3}$$

KCL at V^-: $\dfrac{V_{test} - V_{out}}{R_2} + \dfrac{V_{test}}{R_1} = 0$

$$V_{out} = V_{test} + \frac{R_2 V_{test}}{R_1}$$

$$V_{out} = V_{test}\left(1+\frac{R_2}{R_1}\right)$$

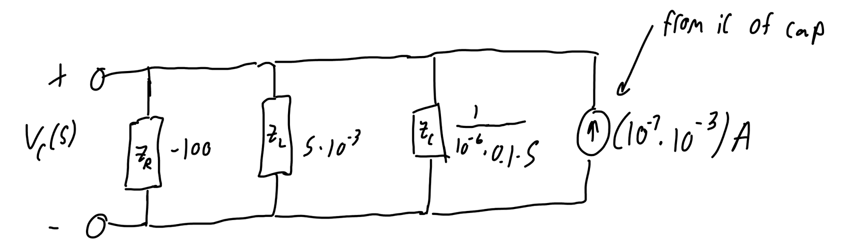
$$V_{test}\left(1+\frac{R_2}{R_1}\right) = V_{test} - I_{test} \cdot R_3$$

$$\cancel{1} + \frac{R_2}{R_1} = \cancel{1} - \frac{I_{test}}{V_{test}} \cdot R_3$$

$$\frac{-R_2}{R_1 R_3} = \frac{I_{test}}{V_{test}}$$

$$\boxed{\frac{V_{test}}{I_{test}} = R_{th} = \frac{-R_1 R_3}{R_2}}$$

3.

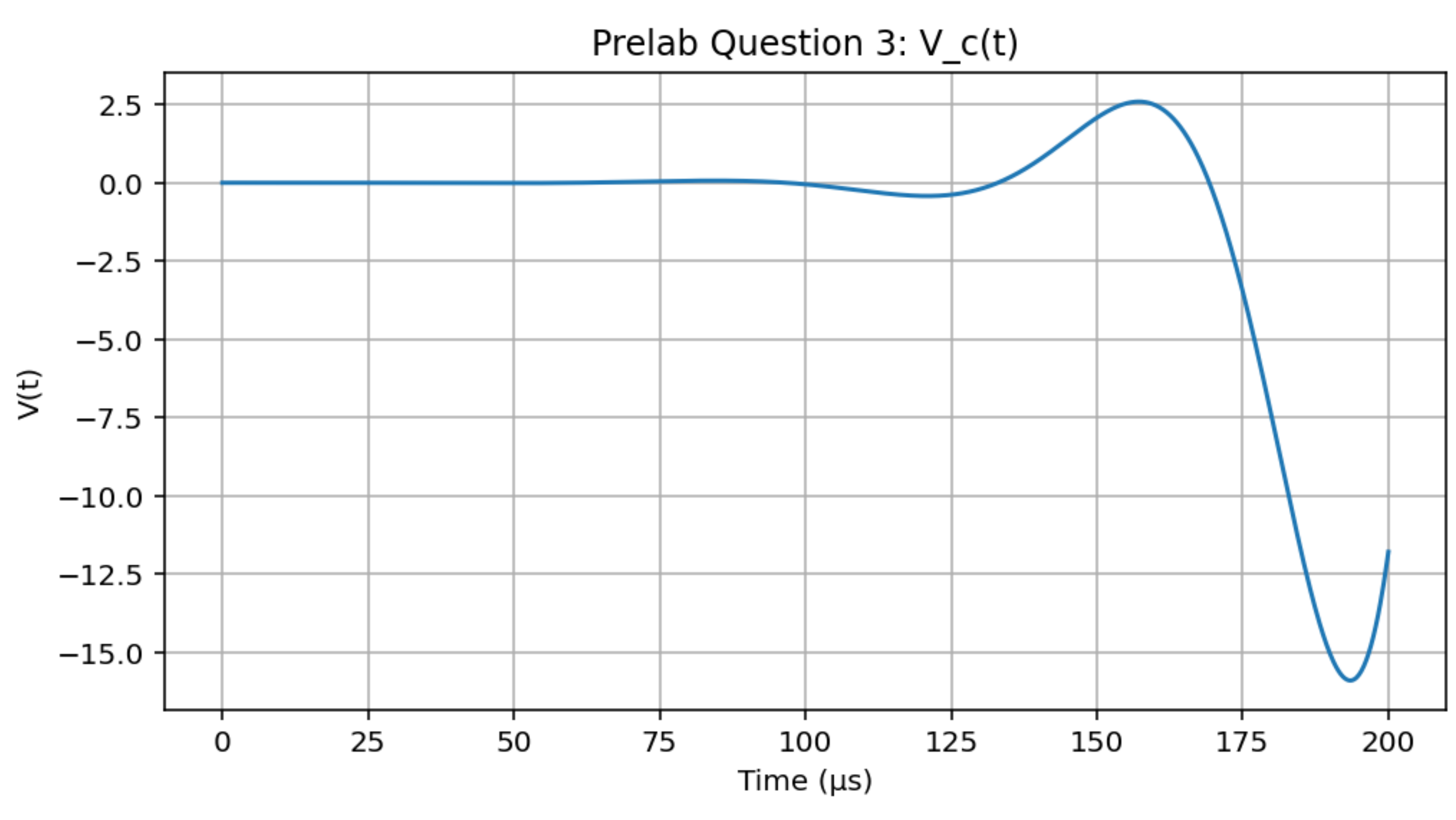


$$Z_{eq}(s) = -100 \,\|\, s \cdot 10^{-3} \,\|\, \frac{1}{10^{-7} \cdot s} = \left(\frac{-1}{100} + \frac{1}{s \cdot 10^{-3}} + 10^{-7} \cdot s\right)^{-1} = \frac{100s}{10^{-5}s^2 - s + 10^5}$$

$$I_{in}(s) = 10^{-10} A$$

$$V_c(s) = Z_{eq}(s) \cdot I_{in}(s) = 10^{-10} \cdot \frac{100s}{10^{-5}s^2 - s + 10^5}$$

$$\mathcal{L}^{-1}\{V_c(s)\} = V_c(t)$$




```
import numpy as np
import matplotlib.pyplot as plt
from sympy import  inverse_laplace_transform, simplify
from sympy.utilities.lambdify import lambdify
from sympy.abc import s, t


V_s = 1e-10 * (100 * s) / (1e-5 * s**2 - s + 1e5)


v_t = simplify(inverse_laplace_transform(V_s, s, t))


print(f"Time-domain function: {v_t}")


t_vals = np.linspace(0, 200e-6, 1000)  # 1000 points between 0 and 200 µs


v_func = lambdify(t, v_t, 'numpy')  # Converts sympy function to numpy function


v_vals = v_func(t_vals)


plt.figure(figsize=(8, 4))
plt.plot(t_vals * 1e6, v_vals)
plt.title('Prelab Question 3: V_c(t)')
plt.xlabel('Time (µs)')
plt.ylabel('V(t)')
plt.grid()
plt.show()
```