Assignment No. 6: Report

-Akash Reddy A, EE17B001

March 19, 2019

1 Overview

Assignment 6 deals with the Laplace Transform.

This assignment uses the Laplace Transform to analyse **Linear Time-Invariant Systems** or **LTI systems**. LTI systems can be used to model a number of real-life systems that are of interest to Electrical Engineers, such as linear circuits and communication channels.

Assignment 6, however, deals only with mechanical systems, and serves as an introduction to the usage of the Laplace Transform to solve for electrical systems.

Python provides us with the Signals toolbox, which largely simplifies calculations that pertain to signal processing and calculation.

2 The Assignment

Relevant Theory

A Linear Time-Invariant System is one whose output-input relationship:

- possesses the properties of *additivity* and *homogeneity* (making system linear).
  - **Additivity** of a system means that when the system is defined as:

    $$x(t) \longleftrightarrow y(t)$$

    it also satisfies the following:

    $$x_1(t) + x_2(t) \longleftrightarrow y_1(t) + y_2(t)$$

    In words, this means that one gets the same thing from an additive system by taking two input signals, passing them through the system and adding the outputs, *and* by adding the two input signals, passing the sum through the system, and obtaining the output.
  - **Homogeneity** of a system means that when the system is defined as:

    $$x(t) \longleftrightarrow y(t)$$

    it also satisfies the following:

    $$c \cdot x(t) \longleftrightarrow c \cdot y(t)$$

    In words, this means that one gets the same thing from a homogeneous system by taking an input signal, passing it through the system and multiplying the output by a scaling factor c, *and* by scaling the input signal by c, passing the result through the system, and obtaining the output.

Both the above conditions have to hold true for linearity to hold.

- does not depend on the time at which the input is given to the system (making system time-invariant).
  - **Time-invariance** of a system means that when the system is defined as:

    $$x(t) \longleftrightarrow y(t)$$

    it also satisfies the following:

    $$x(t - t_0) \longleftrightarrow y(t - t_0)$$

    In words, this means that one gets the same thing from a time-invariant system by taking an input signal, passing it through the system and time-shifting the output, *and* by time-shifting the input signal, passing the time-shifted input through the system, and obtaining the output.

Now, an LTI system always gives the same output for an input $x(t) = \delta(t)$, at all values of t because of its time-invariance. Since this output now depends only on the nature of the system, it can be used to characterise an LTI system. Indeed, this response is used to characterise an LTI system, and is called its **Impulse Response** denoted by $h(t)$.

Any system can be expressed as an integral (or sum in the discrete-time case) of shifted and scaled impulses. Therefore, we can write:

$$x(t) = \int_{-\infty}^{\infty} x(\tau)\delta(t - \tau)\, d\tau$$

Now, we have:

$$\begin{aligned}
\delta(t) &\longleftrightarrow h(t) \\
\delta(\tau) &\longleftrightarrow h(\tau) \\
\delta(t - \tau) &\longleftrightarrow h(t - \tau) \\
x(t)\delta(t - \tau) &\longleftrightarrow x(t)h(t - \tau) \\
\int_{-\infty}^{\infty} x(\tau)\delta(t - \tau)\, d\tau &\longleftrightarrow \int_{-\infty}^{\infty} x(\tau)h(t - \tau)\, d\tau \\
x(t) &\longleftrightarrow \int_{-\infty}^{\infty} x(\tau)h(t - \tau)\, d\tau \\
&\qquad\qquad = y(t)
\end{aligned}$$

The RHS in the above result $y(t) = x(t) \longleftrightarrow \int_{-\infty}^{\infty} x(\tau)h(t - \tau)\, d\tau$ is known as the **convolution integral** of the input $x(t)$ with the impulse response $h(t)$ and is the output given any input and system (characterised by its impulse response).

However, when we convert the above to the Laplace Domain, we have a property that greatly simplifies this:

$$\begin{aligned}
\int_{-\infty}^{\infty} x(\tau)h(t - \tau)\, d\tau &= y(t) \\
\implies X(s) \cdot H(s) &= Y(s)
\end{aligned}$$

Additionally, we have the differentiation property of the Laplace Transform:

$$\begin{aligned} x(t) &\longleftrightarrow X(s) \\ \frac{d\,x(t)}{dt} &\longleftrightarrow sX(s) - x(0^-) \\ \frac{d^2\,x(t)}{dt^2} &\longleftrightarrow s^2X(s) - sx(0^-) - \dot{x}(0^-) \end{aligned}$$

Keeping all the above theory in mind, we use the following functions provided by `numpy` and `scipy.signal`:

- `numpy.polyadd()`: Used to add polynomials which are in the form of coefficient arrays.
- `numpy.poylmul()`: Used to multiply polynomials which are in the form of coefficient arrays.
- `scipy.signal.lti(Num, Den)`: Defines a transfer function.
- `scipy.signal.impulse()`: Takes the transfer function and the time vector values as arguments, and calculates impulse response of the transfer function at the given time points.
- `scipy.signal.lsim()`: Takes input time-domain signal, impulse response of the system, and time vector values as arguments, and calculates the convolution/output from the system.

This report tackles the Assignment in a question-by-question approach.

Question 1

In this question,

$$f(t) = cos(1.5t)e^{-0.5t}u_0(t)$$

is the input to the spring system, and $F(s)$ (Laplace Domain equivalent) is:

$$F(s) = \frac{(s+0.5)}{(s+0.5)^2 + 2.25}$$

The equation that is satisfied by the spring is:

$$\ddot{x} + 2.25x = f(t)$$

We write this equation in the Laplace Domain to be able to easily solve for the output. In order to do this, we make use of the differentiation property of the Laplace Transform. We have the initial conditions $x(0^-) = 0$ and $\dot{x}(0^-) = 0$.

$$\ddot{x} + 2.25x = f(t)$$

$$s^2X(s) - sx(0^-) - \dot{x}(0^-) + 2.25X(s) = F(s)$$

$$s^2X(s) + 2.25X(s) = F(s)$$

$$\frac{X(s)}{F(s)} = \frac{1}{s^2 + 2.25} = H(s)$$

which is the Transfer Function of the system. Now, we have:

$$\begin{aligned} X(s) &= F(s) \cdot H(s) \\ &= \frac{(s+0.5)}{((s+0.5)^2+2.25)\cdot(s^2+2.25)} \end{aligned}$$

We have the `signal.impulse` function which calculates the impulse response $h(t)$ of a system given its transfer function $H(s)$. This means that this function can be used for Laplace Domain-Time Domain conversion by assuming the Laplace Domain function we have to be the "transfer function" of a system and finding the corresponding "impulse response". We do this for X(s) to obtain the required output $x(t)$ and then plot it.

Relevant Code:

```
H=sp.lti([1],[1,0,2.25])
F=sp.lti([1,0.5],polyadd(polymul([1,0.5],[1,0.5]),[2.25]))
X=sp.lti(polymul(F.num,H.num), polymul(F.den,H.den))
t,x=sp.impulse(X,None,np.linspace(0,50,501))
```

The above code first generates $H(s)$ and $F(s)$, solves for $X(s)$ and then uses the `signal.impulse` function to evaluate x(t).

Plot Generated:

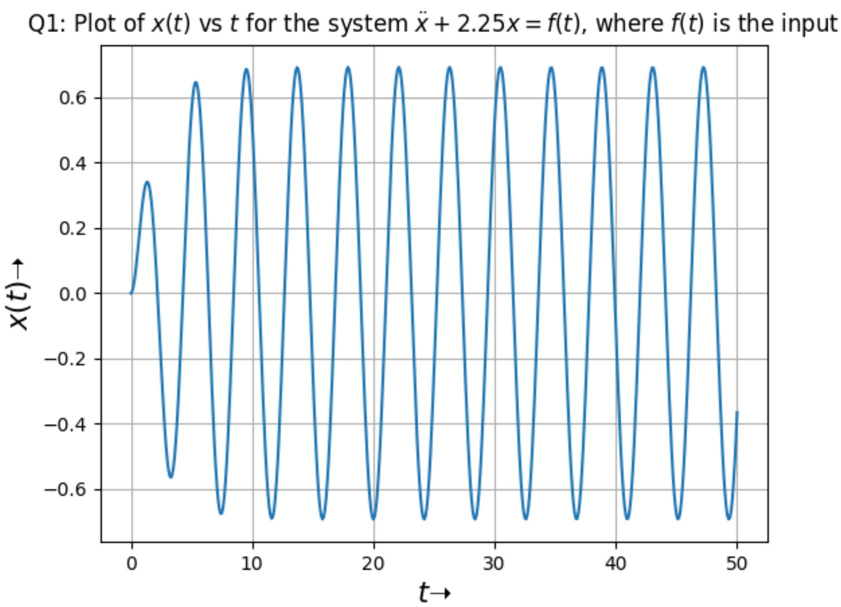

Figure 1: Plot of $x(t)$ vs t for the system $\ddot{x} + 2.25x = f(t)$, where f(t) is the input

We notice that the first oscillation has a smaller amplitude, and the next one has an amplitude almost equal to the constant amplitude, after which the output nearly reaches constant amplitude. This is because the applied force $f(t)$ has a much faster decay rate, and constant amplitude is reached much faster as a result.

Question 2

The same as Question 1 is done, except $f(t)$ has a much smaller decay. As a result of this, we can see the effect of the applied force $f(t)$ last a long time on the system. **The amplitude varies according in a decaying sinusoidal manner until $f(t)$ decays to nearly zero.** This time, we have:

$$f(t) = cos(1.5t)e^{-0.05t}u_0(t)$$

and therefore:

$$F(s) = \frac{(s + 0.05)}{(s + 0.05)^2 + 2.25}$$

We modify the code accordingly and plot.

Relevant Code:

```
H=sp.lti([1],[1,0,2.25])
F=sp.lti([1,0.05],polyadd(polymul([1,0.05],[1,0.05]),[2.25]))
X=sp.lti(polymul(F.num,H.num), polymul(F.den,H.den))
t,x=sp.impulse(X,None,np.linspace(0,50,501))
```

Plot Generated:

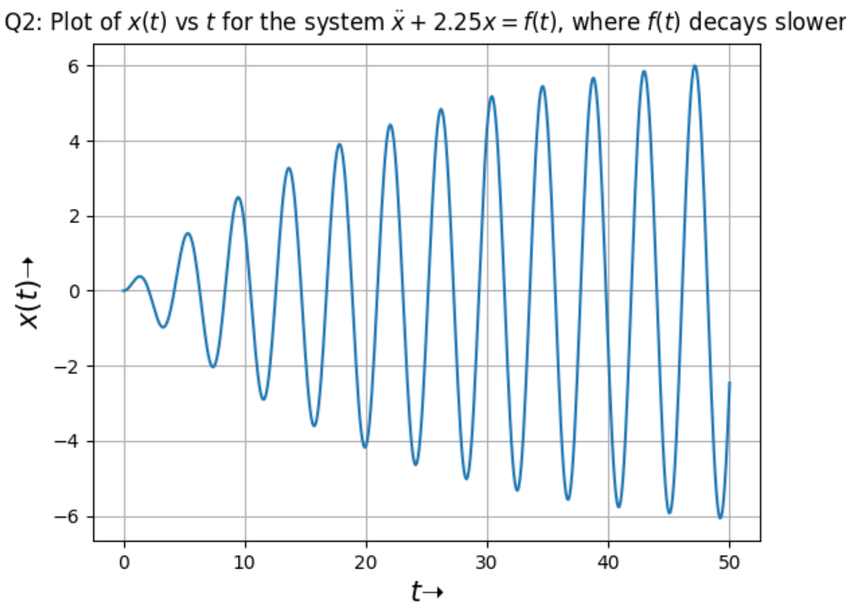

Figure 2: Plot of $x(t)$ vs t for the system $\ddot{x} + 2.25x = f(t)$, where f(t) decays slower

In this plot, we notice that the highest amplitude reached is also much higher than in Question 1, apart from the general trend of longer-lasting increase of amplitude. This is because the input force $f(t)$ dies out slower and lasts much longer, and leads to much larger amplitudes and slower arrival at constant amplitude.

Question 3

The same as Question 2 is done, except we solve and plot for multiple frequencies ω. Here, it is asked to solve using `signal.lsim` as opposed to `signal.impulse` previously, so the same is done.

In this process, we do not convert both the input and the impulse response to the Laplace Domain, multiply the two and take the inverse to get the output. Instead, the input is taken in the Time Domain by `signal.lsim` along with the transfer function, and the Time Domain output is directly obtained.

Relevant Code:

```
freq=np.arange(1.4,1.61,0.05)
for i in freq:
    f=np.cos(i*t)*np.exp(-0.05*t)*np.heaviside(t, 0.5)
    t,x,svec=sp.lsim(H, f, t)
    plot(t,x, label='$\omega=$'+str(i))
```

We have already obtained $H(t)$ from earlier calculations, so `H` can be directly used.

Plot Generated:

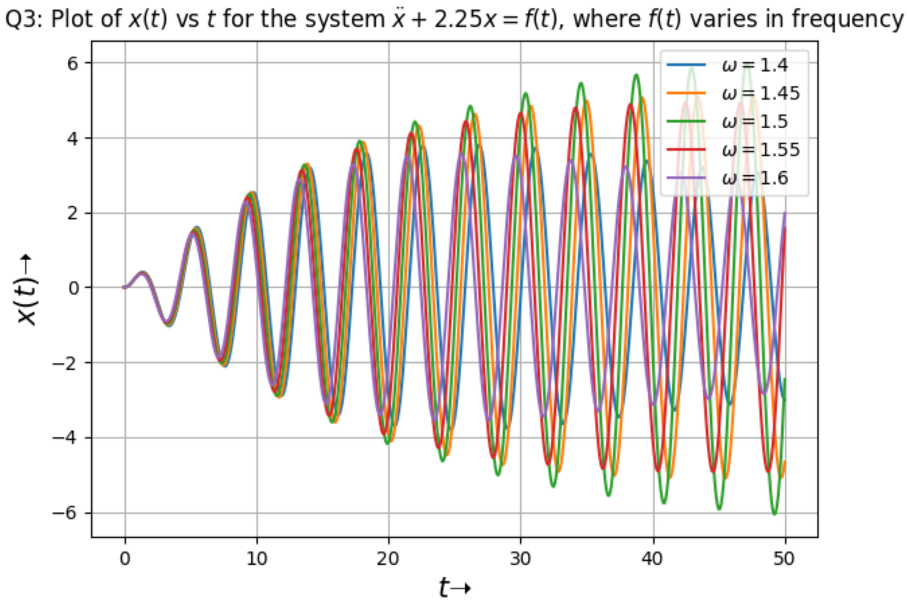

Figure 3: Plot of $x(t)$ vs t for the system $\ddot{x} + 2.25x = f(t)$, where f(t) varies in frequency

We plot all the curves with ω varying in steps of 0.05 between 1.4 and 1.6. All of them follow a similar trend of large amplitudes and delayed constant amplitude.

However, from the $\omega = 1.4$ curve to the $\omega = 1.6$ curve, the speed of oscillations gradually increases (obviously so, as the frequency increases). Additionally, the envelopes of the functions start decreasing in a sinusoidal manner for higher values of t, for ω farther from $\omega = 1.5$. The maximum value of the envelopes also varies in the same manner, being the highest for $\omega = 1.5$.

For $\omega = 1.5$, we see a *constant increase of the envelope*. However for $\omega = 1.4$ and $\omega = 1.6$, we can clearly see that the envelopes reach a maximum at nearly the same point and proceed to decrease in a sinusoidal fashion.

All of the above occurs because of a phenomenon known as **resonance**, which is the reason for the continuously increasing amplitude at $\omega = 1.5$. Resonance occurs when the frequency of an applied force is equal to the natural frequency of the system.

The **natural frequency** of a system is the frequency at which the system tends to oscillate in the absence of any driving or damping force.

In our equation of motion:

$$\ddot{x} + 2.25x = f(t)$$

if the applied force is removed, we get:

$$\ddot{x} + 2.25x = 0$$

This is a simple harmonic motion equation, with $\omega = \sqrt{2.25} = 1.5$.

Therefore, the natural frequency is $\omega = 1.5$ and, as observed above, **resonance occurs at** $\omega = 1.5$.

Question 4

This question solves the coupled spring equations:

$$\ddot{x} + (x - y) = 0$$
$$\ddot{y} + 2(y - x) = 0$$

We write these equations in the Laplace Domain to be able to easily solve for the output. In order to do this, we make use of the differentiation property of the Laplace Transform. We have the initial conditions $x(0^-) = 1$ and $\dot{x}(0^-) = \dot{y}(0^-) = y(0^-) = 0$.

$$s^2X(s) - sx(0^-) - \dot{x}(0^-) + X(s) - Y(s) = 0$$
$$s^2Y(s) - sy(0^-) - \dot{y}(0^-) + 2Y(s) - 2X(s) = 0$$

Substituting initial conditions, we get:

$$s^2X(s) - s + X(s) - Y(s) = 0$$
$$s^2Y(s) + 2Y(s) - 2X(s) = 0$$

and solving, we get:

$$X(s) = \frac{s^3 + 2s}{s^4 + 3s^2}$$
$$Y(s) = \frac{2s}{s^4 + 3s^2}$$

We now use `signal.impulse` to solve for the Time Domain functions $x(t)$ and $y(t)$, which are then plotted for $0 \leq t \leq 20$.

Relevant Code:

```
X=sp.lti([1,0,2,0],[1,0,3,0,0])
Y=sp.lti([2,0],[1,0,3,0,0])
t,x=sp.impulse(X,None,np.linspace(0,20,201))
t,y=sp.impulse(Y,None,np.linspace(0,20,201))
```

Plot Generated:

As we can see, the initial conditions are all satisfied. The slopes of both functions at $t = 0$ are 0, and we also have $x(0^-) = 1$ and $y(0^-) = 0$.

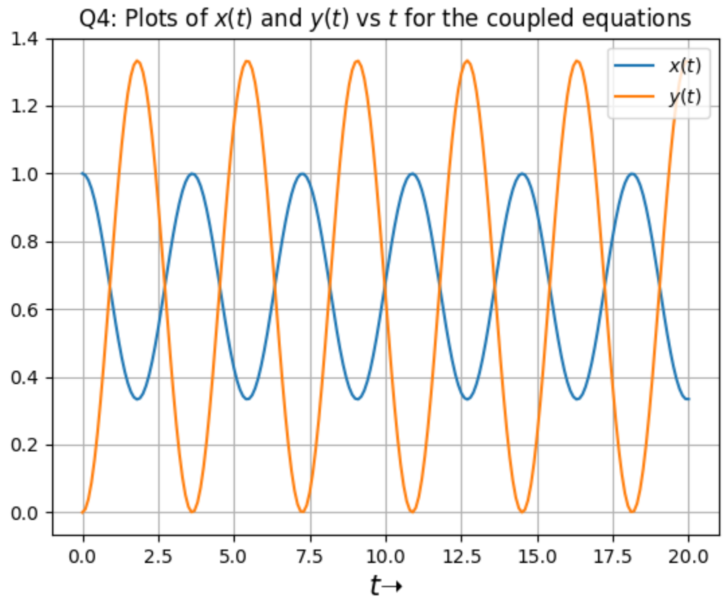

Figure 4: Plot of $x(t)$ vs t for the system $\ddot{x} + 2.25x = f(t)$, where f(t) varies in frequency

Question 5

We analyse our circuit in the Laplace Domain.

In the Laplace Domain, all capacitors C_i turn into impedances $\frac{1}{sC_i}$, and inductors L_i turn into impedances sL_i. Then, Kirchoff's laws are used to carry out circuit analysis in the frequency domain.

In the given circuit, we have:

$$\frac{V_i}{R + \frac{1}{sC} + sL} \cdot \frac{1}{sC} = V_o$$

$$\frac{V_i}{s^2LC + sCR + 1} = V_o$$

$$\implies \frac{V_i}{V_o} = \frac{1}{s^2LC + sCR + 1} = H(s)$$

Substituting our values, we have:

$$H(s) = \frac{1}{10^{-12}s^2 + 10^{-4}s + 1}$$

or

$$H(s) = \frac{10^{12}}{s^2 + 10^8 s + 10^{12}}$$

This transfer function is created using `signal.lti` and the corresponding magnitude and phase response are plotted. The Bode Plot tells us that the poles occur at $|\omega| = 10^4$ and $|\omega| = 10^8$, which can be verified to be equal to the poles of the transfer function (zeroes of the denominator).

Relevant Code:

```
H=sp.lti([10**12],[1,10**8,10**12])
w,S,phi=H.bode()
```

We plot using a semilog(x) axis as we always do for Bode Plots.

Plot Generated:

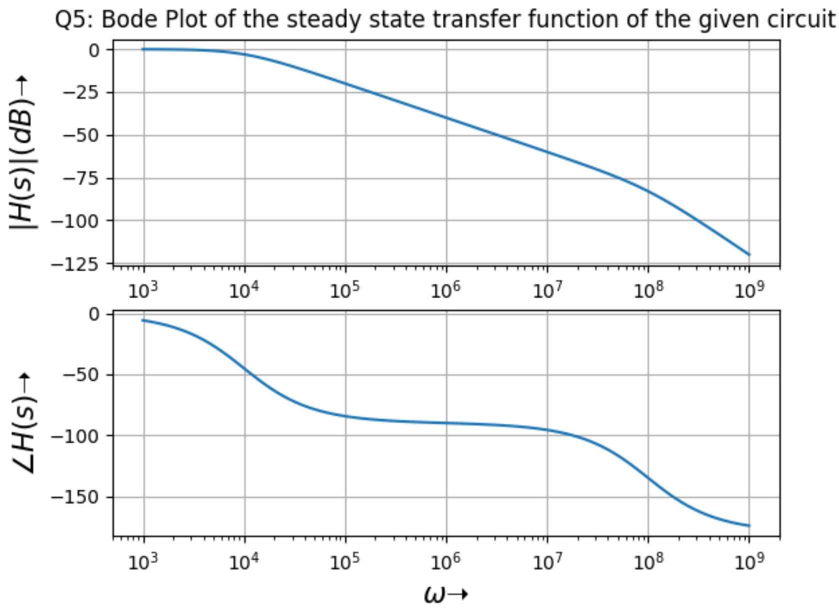

Figure 5: Bode Plot of the steady state transfer function of the given circuit

Question 6

This question is a continuation to Question 5, where a specific input signal $v_i(t)$ given by:

$$v_i(t) = \cos(10^3 t)u(t) - \cos(10^6 t)u(t)$$

The output is obtained by using the input signal and the transfer function of the system earlier obtained as arguments to `signal.lsim`. We first plot the output $v_o(t)$ for 10 ms, and then plot the first 30 μs specifically to observe the trend of the output voltage in this time.

Relevant Code:

```
t=np.linspace(0,10**-2,50001)
vi=(np.cos(10**3*t)-np.cos(10**6*t))*np.heaviside(t, 0.5)
t,vo,svec=sp.lsim(H, vi, t)
f=figure(1)
plot(t,vo)

t=np.linspace(0,3*10**-5,3001)
vi=(np.cos(10**3*t)-np.cos(10**6*t))*np.heaviside(t, 0.5)
t,vo,svec=sp.lsim(H, vi, t)
g=figure(2)
plot(t,vo)
```

Plots Generated:

We plot the input and the output together in each case as subplots of each plot, for easier comparison and qualitative analysis.

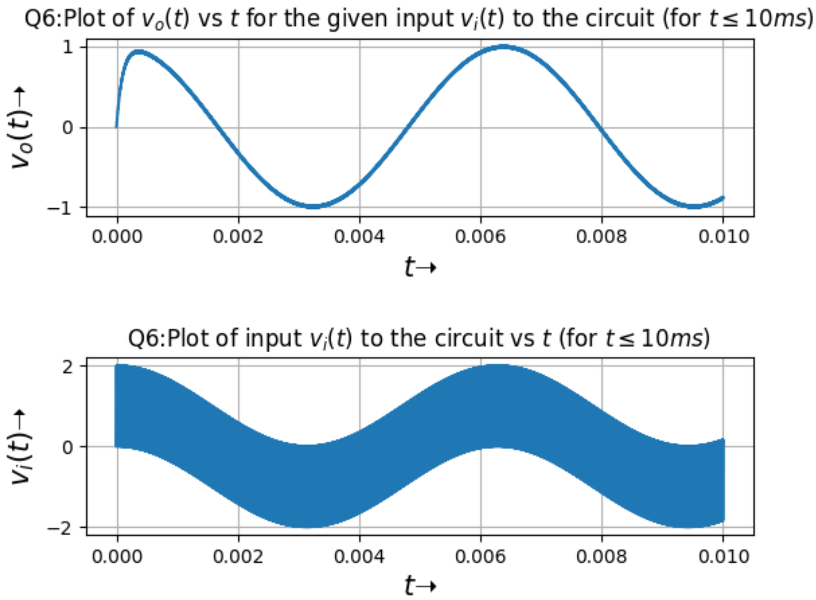

Figure 6: Output $v_o(t)$ and input $v_i(t)$ vs t (for $t \leq 10ms$)

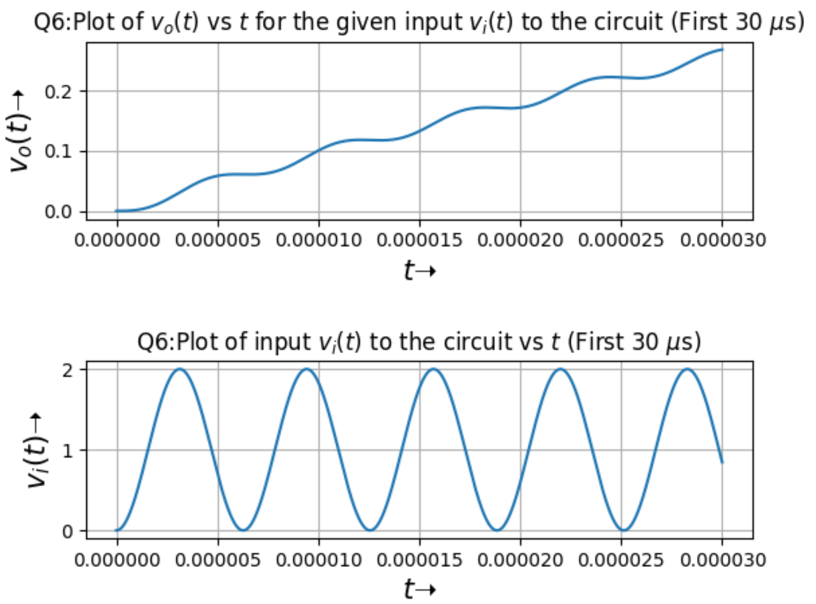

Figure 7: Output $v_o(t)$ and input $v_i(t)$ vs t (First 30 μs)

The input signal, as we know, is the superposition of two sinusoidal signals of largely different frequencies, one much higher than the other ($10^6 >> 10^3$). What we send as the input $v_i(t)$ is essentially a $\cos(10^6 t)$ function superimposed on top of a $\cos(10^3 t)$ wave.

Therefore, we get a **sinusoidal band** as the input as seen in subplot 2 of Figure 6. The band nature is because of the rapidly oscillating $\cos(10^6 t)$ wave, and the sinusoidal nature of the whole band itself is due to the $\cos(10^3 t)$ wave.

Now, we observe from the Bode Plot or use the transfer function to realise that the value of $|H(s)| \approx 10^{\frac{0}{20}} = 1$ for $\omega = 10^3$ and $\approx 10^{\frac{-40}{20}} = 10^{-2}$ for $\omega = 10^6$.

What this means is that the gain for the $\cos(10^3 t)$ component is nearly equal to 1, therefore the output will be nearly the same as the input. However, the gain for the $\cos(10^6 t)$ component is nearly equal to $\frac{1}{100}$, therefore the output for this component will be scaled down 100 times. This means that the high frequency component is highly reduced (almost removed) during passage through the system, and allows the low frequency component to pass untouched.

Therefore, the system is a low pass filter (LPF).

However, the filter is non-ideal as the high frequency component has not vanished completely.

This very well explains the long-term output over a time duration of 10 ms visible in Figure 6. The input is a high frequency band oscillating on a sinusoidal base signal. The system filters out the high frequency component and allows only the low frequency signal (the base sinusoid) to pass through. Therefore, the input *sinusoidal band* changes into an output *sinusoidal wave*.

Now, in the interval $0 \leq t \leq 30\mu s$, we have the initial condition that $i(0^-) = 0$. Therefore, there is no potential drop iR across the resistor. There is also no charge on the capacitor and therefore the potential drop across it $v_C = \frac{Q}{C} = 0$. This means that the entire voltage $v_i(0)$ drops across the inductor. Therefore,

$$\begin{aligned} &v_L \text{ is large} \\ \implies &L \cdot \tfrac{di}{dt} \text{ is large.} \end{aligned}$$

Therefore, there is a large initial rate of increase of current $\implies$ current grows rapidly, and the rate at which charge on the capacitor grows increases rapidly $\implies$ v_C grows rapidly. However, this rate of growth of v_C decreases until it reaches the steady state value.

This leads to the rapid growth of $v_o(t)$ in the initial few microseconds.

The bumps in the increasing $v_o(t)$ in the first 30 μs are because of the scaled down $\cos(10^6 t)$ present in the output.

3 Conclusion:

From this Assignment, one learns that:

- The Laplace Domain is a powerful tool to solve complex situations such as systems with higher order differential equations, including circuits that have capacitive and inductive impedances.

- `scipy` has the powerful `signal` toolbox which simplifies any signal-related processing to a matter of a few lines.

- Upon applying an external alternating force to a spring block system of different frequencies and different decay rates, we get different outputs with varying frequencies and times taken to reach constant amplitude.

- In the given series LCR circuit, for the output voltage $v_o(t)$, there is a short term response and a long term response that we observe. The two are qualitatively different.

* *