**Lab 6: s-domain Circuit Analysis**

**Objective:**

To learn how to use MATLAB Symbolic Computation to solve the S-domain circuit problems.

**Prel-Lab:**

1. Find *v0(t)* for the circuit in Fig.6.1 using the following analysis methods in s-domain: Given *vs(t)* = 10u(t) V, *iL*(0) = -1A, *vc*(0) = 5V

a. Mesh Analysis

b. Superposition Theorem

2. Determine the transfer function H(s) for the circuit.

3. Solve AP13.9 from Nilsson & Riedel, assuming zero initial energy.

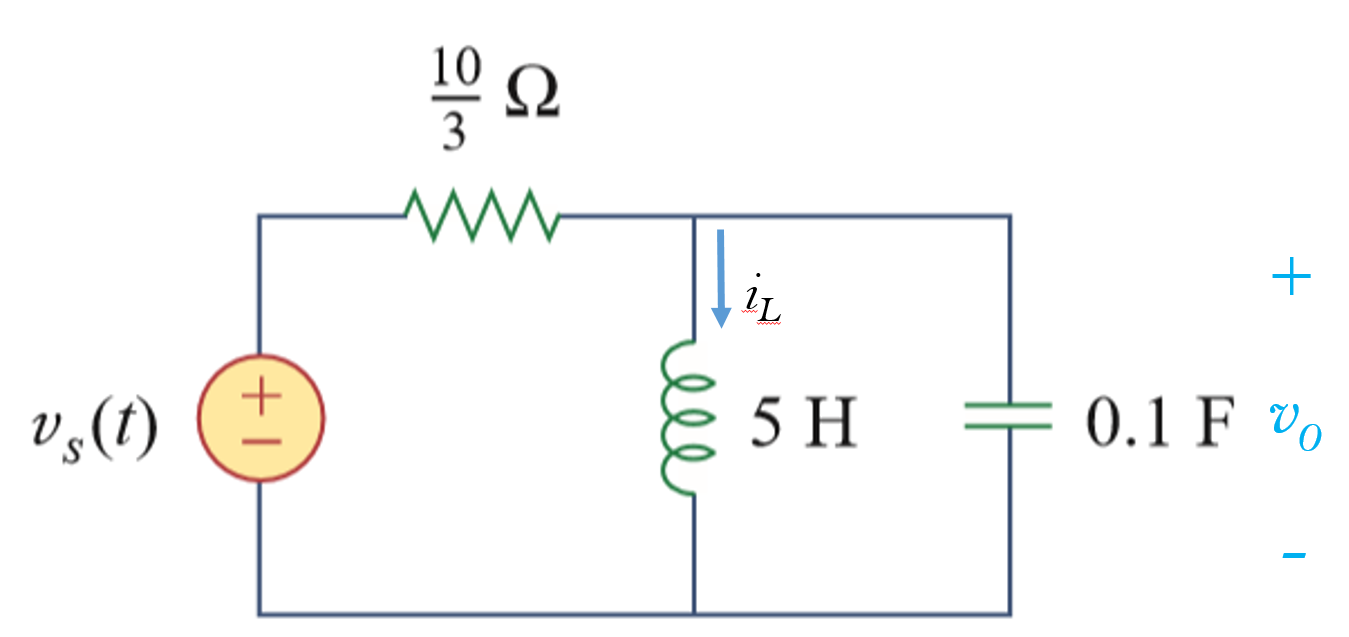
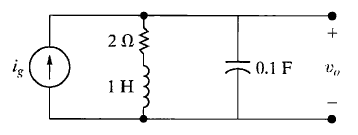
 

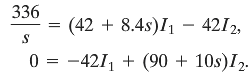
Fig. 6.1 RLC circuit Fig. 6.1 AP 13.9, 13.10

**Procedure:**

**PART 1: Use Symbolic Toolbox to find time-domain solution**

1. Solve the multiple mesh circuit in Figure 13.16 of Nilsson & Reidel as an example.

The book sets up a system of equations in the S-domain as



Write a short MATLAB symbolic script (see the Appendix) to solve the problem and find the time-domain solution. Compare your result with the book’s solution:



1. Update the Matlab code to accept user inputs for A and B. Using the code, solve the s-domain mesh analysis equations for the circuit in Figure 6.1.

*Hint:* Use 3 equations with 2 mesh currents and the voltage V0 as the variables.

1. Now, using your code, find the portion of *v0(t)* due to each of the s-domain inputs and use superposition theorem to find the total *v0(t).*

**PART 2: Find Unit Step and Impulse Responses using Transfer Function**

1. Solve AP13.9 and AP13.10 using transfer function in Matlab.

**AP13.9 (transfer function)**

First, derive the transfer function ***Vo/Ig***. Next, build the transfer function in Matlab.

The simplest way to do this to define the numerator and denominator coefficients as vectors and use **tf** as follows:

num=[0 1 2];

denom=[3 4 5];

H=tf(num,denom)

**AP13.10 (unit step and unit impulse responses)**

Matlab has built-in step and impulse response functionality. Just type

figure; step(H)

figure; impulse(H)

You will see plots showing the unit step and unit impulse responses.

***Note that the unit impulse response shows the transfer function in time-domain.***

1. Using the Matlab code, verify your transfer function from Pre-lab #2.

**Conclusion:**

Comment on your experience using Matlab to solve s-domain circuits and finding transfer function with respect to solving them by hand. What are the key things to keep in mind to generalize your code so that it may be applied to multiple circuits or systems?

**Appendix**

Suggested MATLAB code

clc; %reset the workspace command line

clear all; %clear all the variables

close all; %close all the plots

%%=============Part 1====================================

display('Lab6:Part1');

syms s %S-domain variable s is declared as a symbol

A = [42+8.4\*s, -42; -42, 90+10\*s]

B = [336/s; 0] %Define the linear system AX=B

X = inv(A)\*B %Solving the system for X(1),X(2)

%X=A\B % A\B does the same thing as inv(A)\*B

i1 = ilaplace(X(1));%Taking the inverse Laplace transform

i2 = ilaplace(X(2));

pretty(i1)

pretty(i2)

%%=============Part 2====================================

display('Lab6:Part2');

% build the transfer function

% type doc tf or help tf at the command prompt to learn more

num=input('Enter the numerator values (e.g. [1 2 3]): ');

denom=input('Enter the denominator values (e.g. [1 2 3]): ');

H=tf(num,denom)

figure;step(H)

figure;impulse(H)