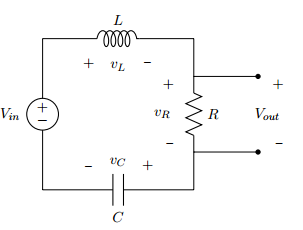
**EX3: RLC circuit with 3/8 4th Order Runge-Kutta**

The Runge-Kutta is an iterative numerical analysis, existing in form , with as the optimal representation of the gradient over the interval . A first step towards implementing numerical analysis on higher order ODEs is to express equation in terms of n coupled 1st ordered ODEs for an nth order ODE.



The RLC circuit can be simplified into equation below:

The formula above can be expressed as a system of two 1st order ODEs, and , whereas set as independent variables contribute to increments in and .

This solution written in MATLAB form is:

func1 = @(t,q,y) y;

func2 = @(t,q,y) (feval(Vin,ti)- q/C - R\*y)/L;

**Expected Results from Circuit Analysis**

Using circuit analysis methods on above circuit will help derive the following frequency response equation:

High frequency Asymptote of frequency response exists at and low frequency Asymptote exists at . Corner frequency exists at , . Damping factor is calculated using , giving for a step input. The peak frequency or resonant frequency exists roughly at , which calculates to be .

**How the code works**

The code is divided into three files for good structure. The program should run the plotting function, which has information of each used, and the details required to plot all 8 graphs. In other words, the plotting function sets up input voltage, unit step size, final time, and the name of the graph to be plotted. This is send to the main function, RLC\_script.m. This m file sets up the circuit with their initial conditions, calculate the number of times the Runge-Kutta process will be iterated, and sets up the arrays in preparation of the numerical analysis procedure. As observed in the MATLAB script, the for-loop will call RK4second.m as many times as needed to calculate every value needed to fill up the array. RK4second then uses the 3/8 Runge-Kutta method to evaluate the value at the next time step.

**Runge-Kutta 3/8**

k1 = h f(xi, yi),

k2 = h f(xi + h / 3, yi + k1 / 3 ),

k3 = h f(xi + 2 h / 3, yi - k1 / 3 + k2 ),

k4 = h f(xi + h, yi + k1 - k2 + k3 ),

yi+1 = yi + 1/8 ( k1 + 3 k2 + 3 k3 + k4 ),

is obtained iteratively.

After the array is filled up, the RLC\_script.m plots both input and output functions out in a graph. The program then goes back to the function plotting script to obtain the next set of and other parameters. After the execution, 8 graphs should be obtained as 8 different sets of input is given.

**RLC\_script.m**

function RLC\_script(Vin, h, t, tf, name)

q0 = 500e-9; %setting initial q condition

t0 = 0; %initial time t = 0;

y0 = 0; %initial condition, q'(0) = 0;

R = 250; %Resistor value

C = 3e-6; %Capacitor value

L = 650e-3; %Inductor Value

N = round((tf-t0)/h); %set size of arrays

qa = zeros (1,length(t));

ya = zeros (1,length(t));

qa(1) = q0; ya(1) = y0; %set up and initialize arrays

for i = 1: N-1 %loop for N steps

qi = qa(i); yi = ya(i); ti = t(i); %temporary names

[q,y] = RK4second(qi, yi,ti,L,h,C,R,Vin);%call RK4second for numerical analysis

qa(i+1) = q; %output from function saved into next space of array

ya(i+1) = y;

end

yy = R\*ya; %Vout = R\*q'

figure;

subplot (2,1,1), plot (t, Vin(t), 'b');%subplotting results

grid on;

title ('Input'); xlabel ('Time (s)'); ylabel ('Vin (V)');

subplot (2,1,2), plot (t, yy, 'g');

grid on;

title(name);

xlabel ('Time (s)');

ylabel('Vout (V)');

end

**Rk4-second.m**

function [q,y] = RK4second(qi, yi, ti,L,h, C, R, Vin)

func1 = @(t,q,y) y; %y = q' = func1

func2 = @(t,q,y) (feval(Vin,ti)- q/C - R\*y)/L; %y''=(Vin - q/c -Rq')/L

k1x = h\*feval(func1, ti,qi,yi); %evaluate func1

k1y = h\*feval(func2, ti,qi,yi); %evaluate func2

k2x = h\*feval(func1, ti+(h/3),qi+(k1x/3), yi+(k1y/3));%k1 gradient evaluated again based on Runga Kutta 3/8

k2y = h\*feval(func2, ti+(h/3),qi+(k1x/3), yi+(k1y/3));

k3x = h\*feval(func1, ti+(2\*h/3), qi-(k1x/3)+k2x, yi-(k1y/3)+k2y);

k3y = h\*feval(func2, ti+(2\*h/3), qi-(k1x/3)+k2x, yi-(k1y/3)+k2y);

k4x = h\*feval(func1, ti+h, qi+k1x-k2x+k3x, yi+k1y-k2y+k3y);

k4y = h\*feval(func2, ti+h, qi+k1x-k2x+k3x, yi+k1y-k2y+k3y);

y =yi+1/8\*(k1y+3\*k2y+3\*k3y+k4y);%optimal representation of gradient over the interval and

%next value of y calculated

q =qi+1/8\*(k1x+3\*k2x+3\*k3x+k4x);%similarly, next value of q calculated

end

**Plotting function (Run this instead of RLC\_script.m for all 8 graphs at the same time)**

%Maths Project, Ex 3---RLC circuit

clear;

close all;

tf = 0.08; %set final time

h = 0.0001; %set step;

t = 0:h:tf; %Created array for time

%Step input signal with amplitude Vin = 5V

%Input shifted to allow more precise comparison

Vin = @(t) 5\*heaviside(t-0.02);

name = 'RLC Vout with Step input'; %name of graph

RLC\_script (Vin, h, t, tf, name);%call function plotresult to plot result

%Impulsive input signal with decay

Vin = @(t) 5\*heaviside(t-0.02).\*exp(-((t-0.02).^2)/(3e-6));

name = 'RLC Vout with impulsive signal input with decay'; %name of graph

RLC\_script (Vin, h, t, tf, name);

%sine at 5, 100, and 500 Hz

Vin = @(t) 5\*sin(10\*pi\*t);

tf = 0.3;

t = 0:h:tf;

name = 'RLC Vout with sinusoidal input at f = 5Hz'; %name of graph

RLC\_script (Vin, h, t, tf, name);

Vin = @(t) 5\*sin(200\*pi\*t);

tf = 0.1;

t = 0:h:tf;

name = 'RLC Vout with sinusoidal input at f = 100Hz'; %name of graph

RLC\_script (Vin, h, t, tf, name);

Vin = @(t) 5\*sin(1000\*pi\*t);

h = 0.00001;

tf = 0.02;

t = 0:h:tf;

name = 'RLC Vout with sinusoidal input at f = 500Hz'; %name of graph

RLC\_script (Vin, h, t, tf, name);

%square at 5, 100, and 500 Hz

Vin = @(t) 5\*square(10\*pi\*t);

h = 0.0001;

tf = 0.3;

t = 0:h:tf;

name = 'RLC Vout with square input at f = 5Hz'; %name of graph

RLC\_script (Vin, h, t, tf, name);

Vin = @(t) 5\*square(200\*pi\*t);

tf = 0.1;

t = 0:h:tf;

name = 'RLC Vout with square input at f = 100Hz'; %name of graph

RLC\_script (Vin, h, t, tf, name);

Vin = @(t) 5\*square(1000\*pi\*t);

h = 0.00001;

tf = 0.02;

t = 0:h:tf;

name = 'RLC Vout with square input at f = 500Hz'; %name of graph

RLC\_script (Vin, h, t, tf, name);

General comments on all 8 output graphs

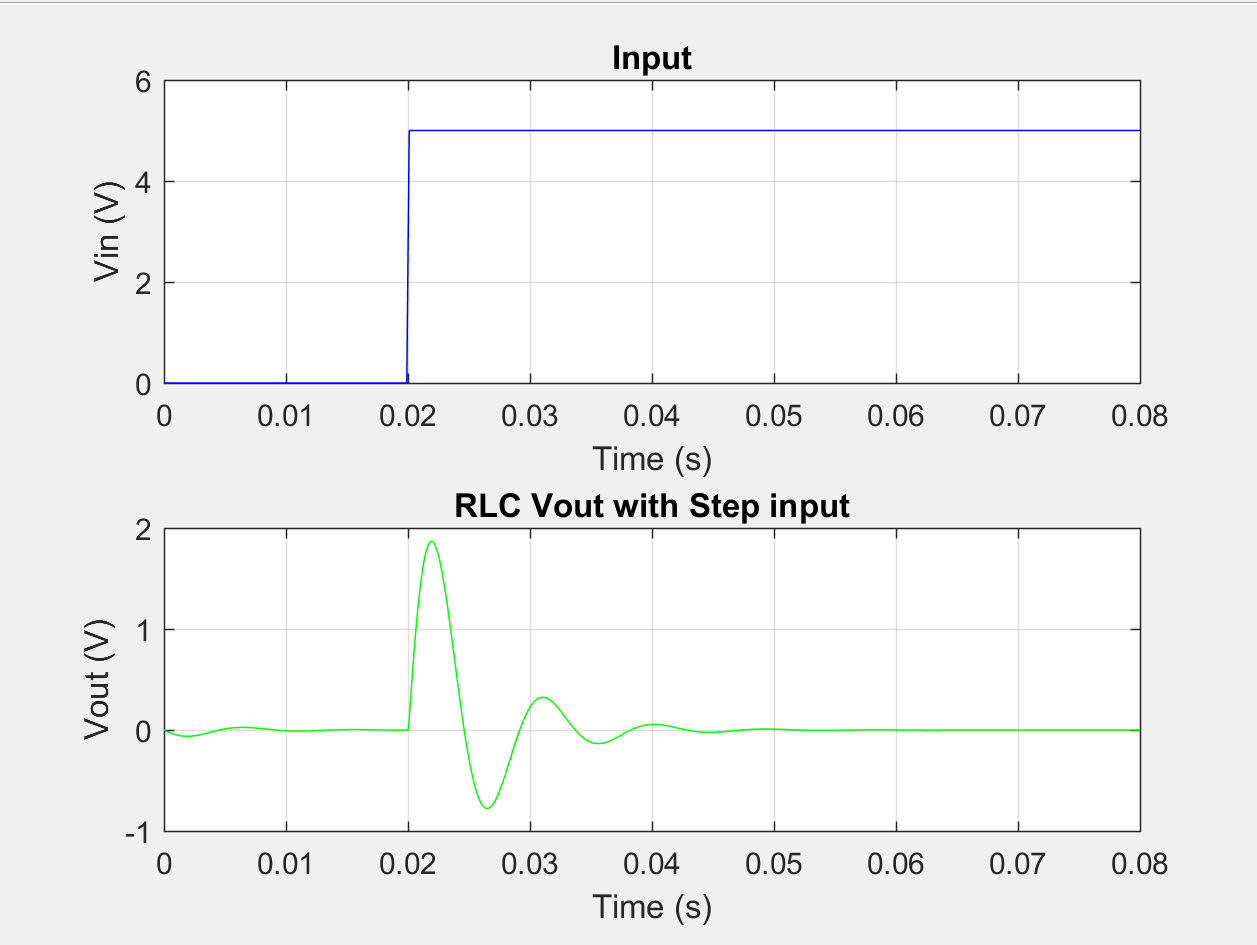
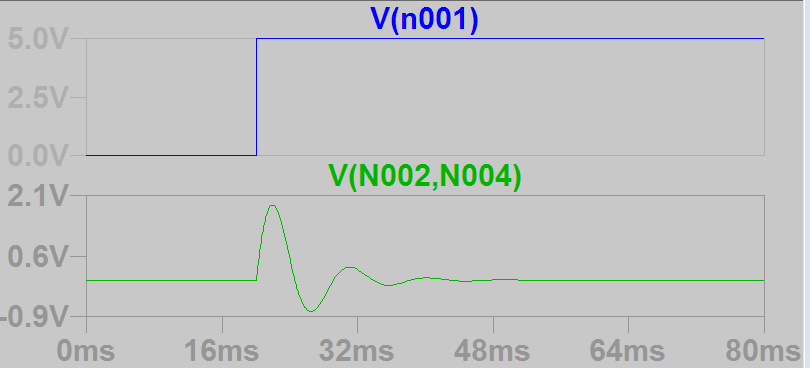
A small disturbance can be seen at the beginning of every output. This is likely due to the initial charge in the capacitor being non-zero, .

It takes times for the circuit to reach steady state. This initial difference is the most observable at higher frequencies. At higher frequencies, the length of time plotted is usually shorter to show the shape of the circuit’s response. The amplitude of this response is also smaller at higher frequencies as shown in the high frequency asymptote approximation. This means the relative size of the initial disturbance is more readily observed in the output graphs.

of RLC with a 5V impulse input

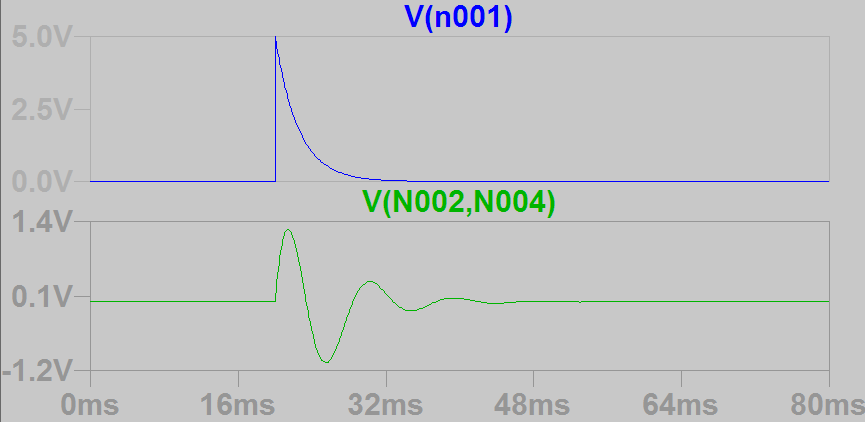
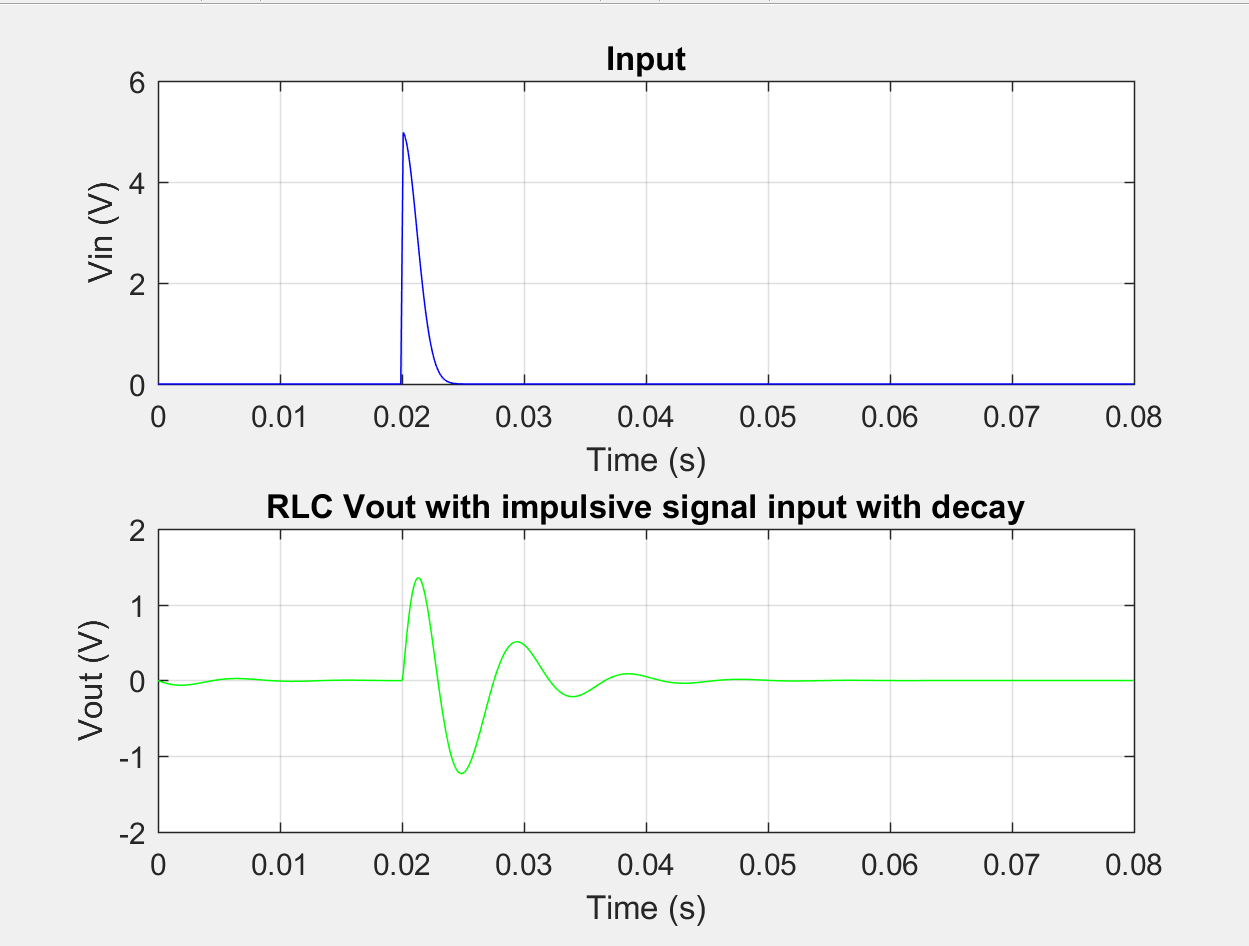
Studying ODE (calculated in figure…) as well as damping factor, a damping factor less than 1 would provide an underdamped response of the form:

The expected result should consist of a decaying sinusoid of oscillation and decaying rate as well as envelope determined by exponential.

**LTSpice simulation result (Step input of 5V)**  **Result from numerical analysis**

5V impulse input with exponential decay

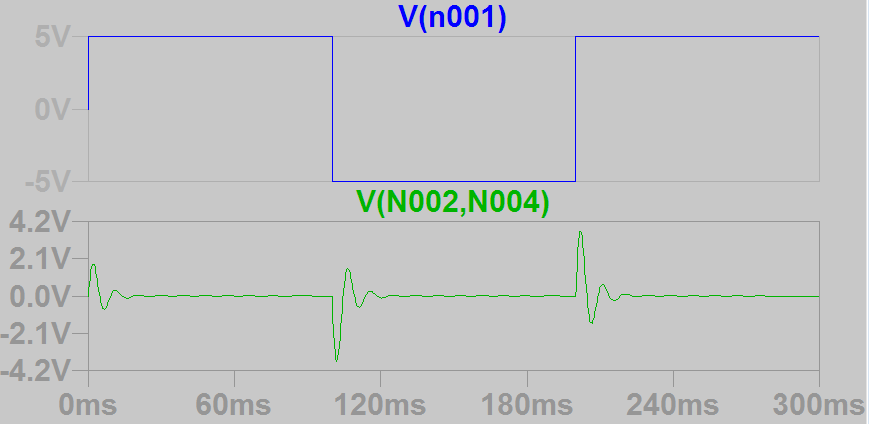
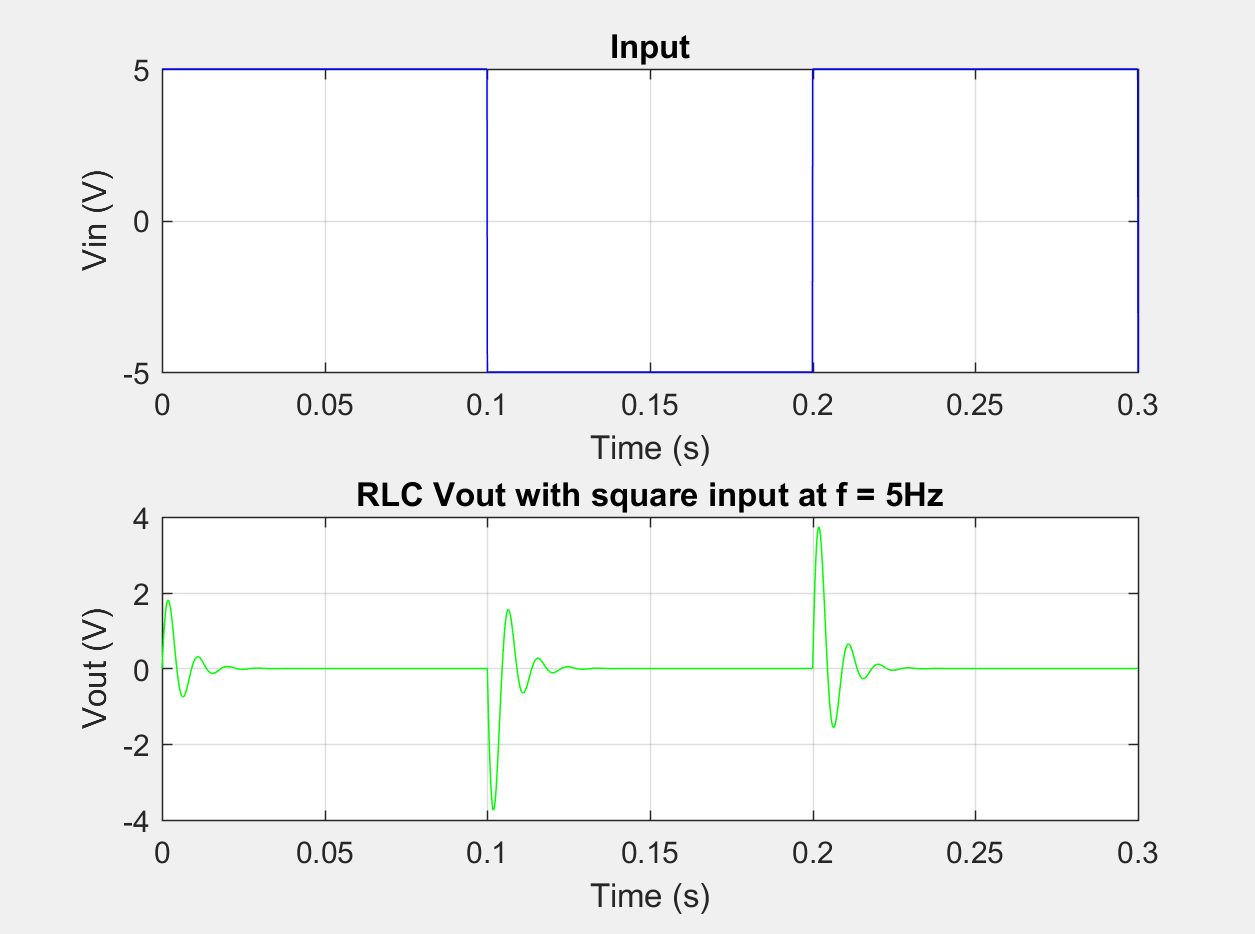
The output now has a relatively similar shape, but with a smaller amplitude and faster settling time.

This is because uh

Square wave with amplitude Vin = 5 V and different frequencies f = 5 Hz, f = 100 Hz, f = 500 Hz:

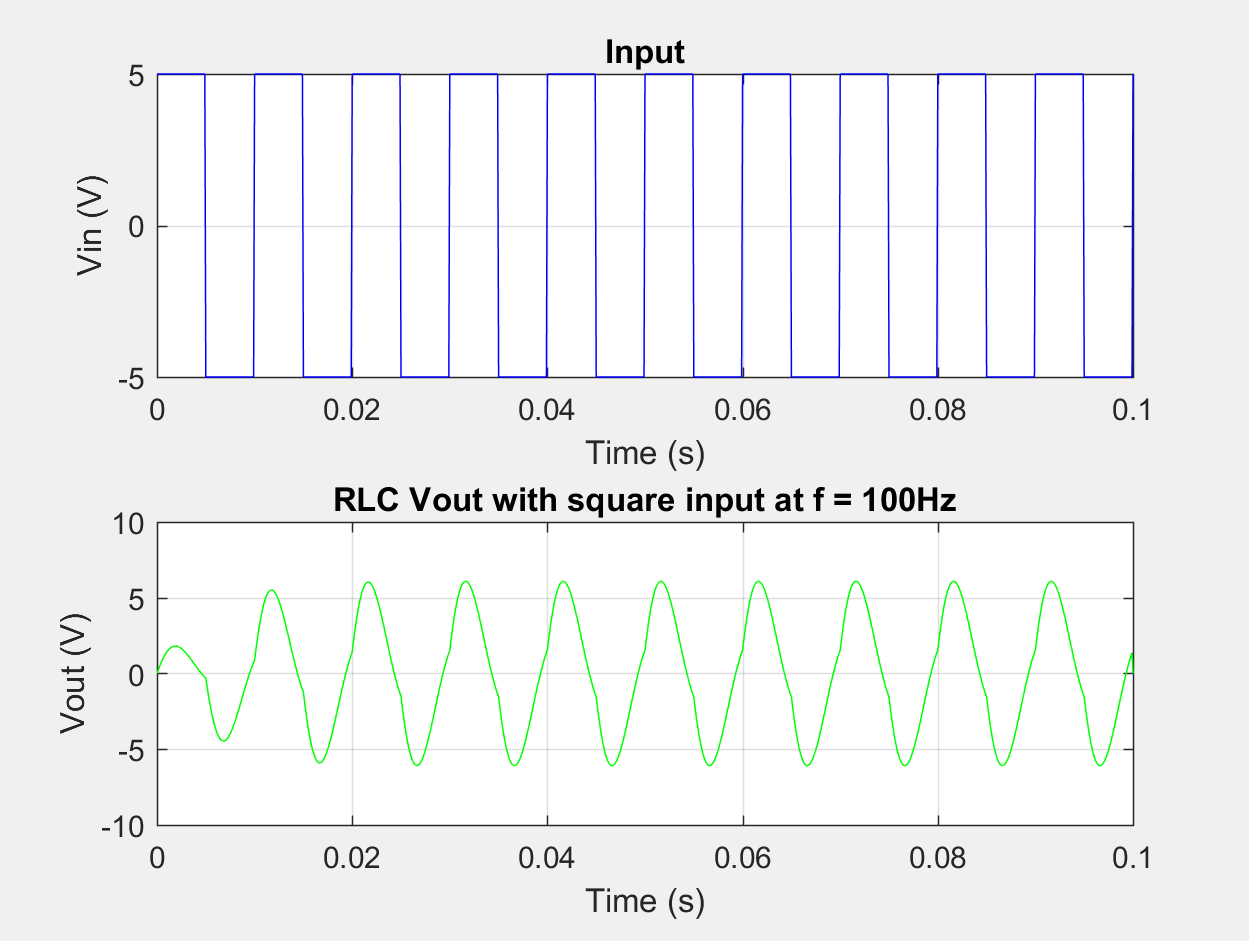
It can be observed that the amplitude is highest at f = 100Hz. This agrees with the calculated resonance frequency of 105 Hz.

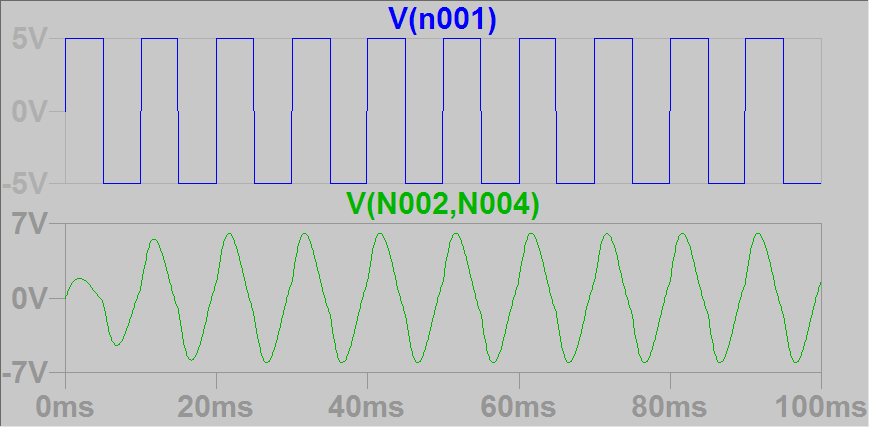
f=5Hz



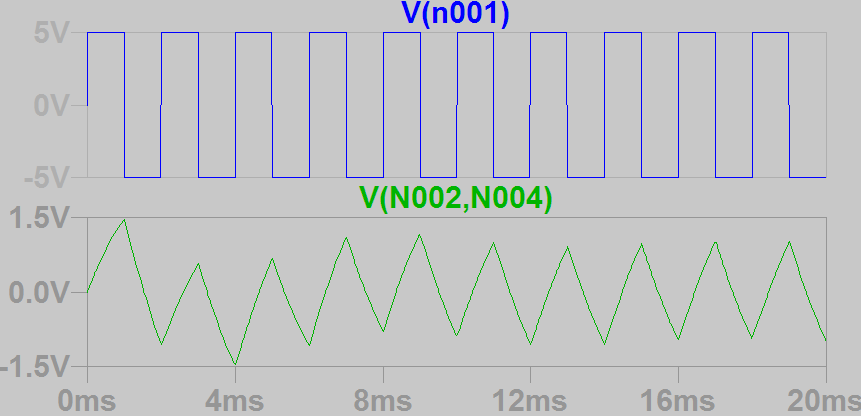
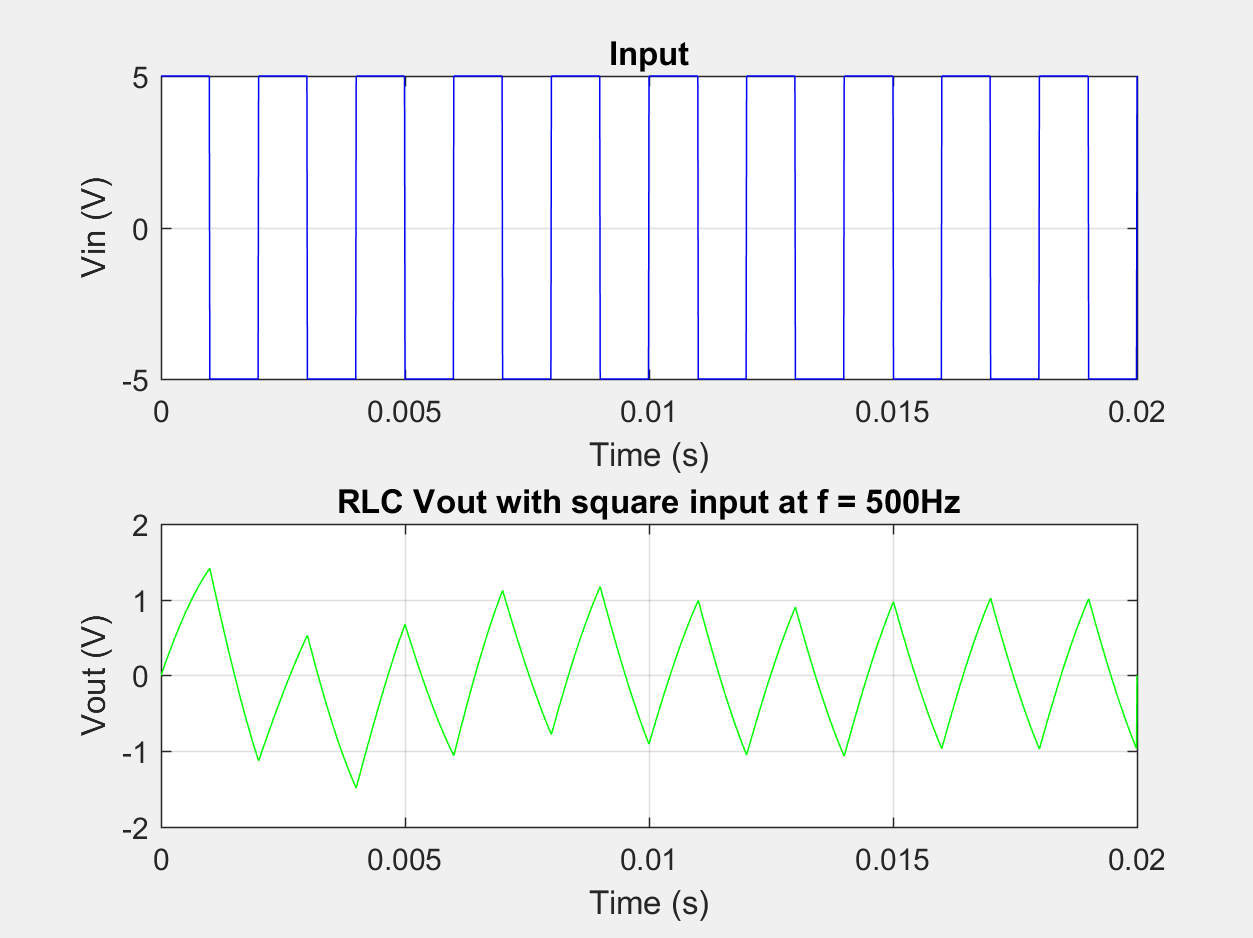
At low frequency, we see the circuit stays close to the steady state with an output voltage at 0V most of the time. Every 0.1s, the input switches. This causes a response very similar to a step response and it quickly dies down and reaches 0V again.

f=100Hz



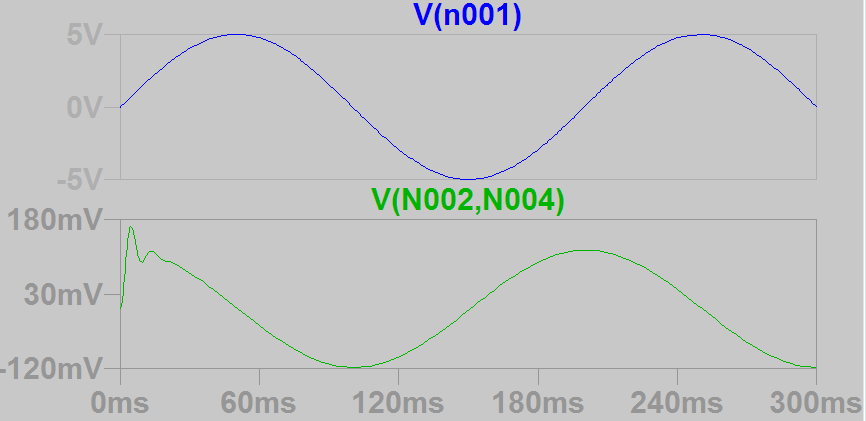
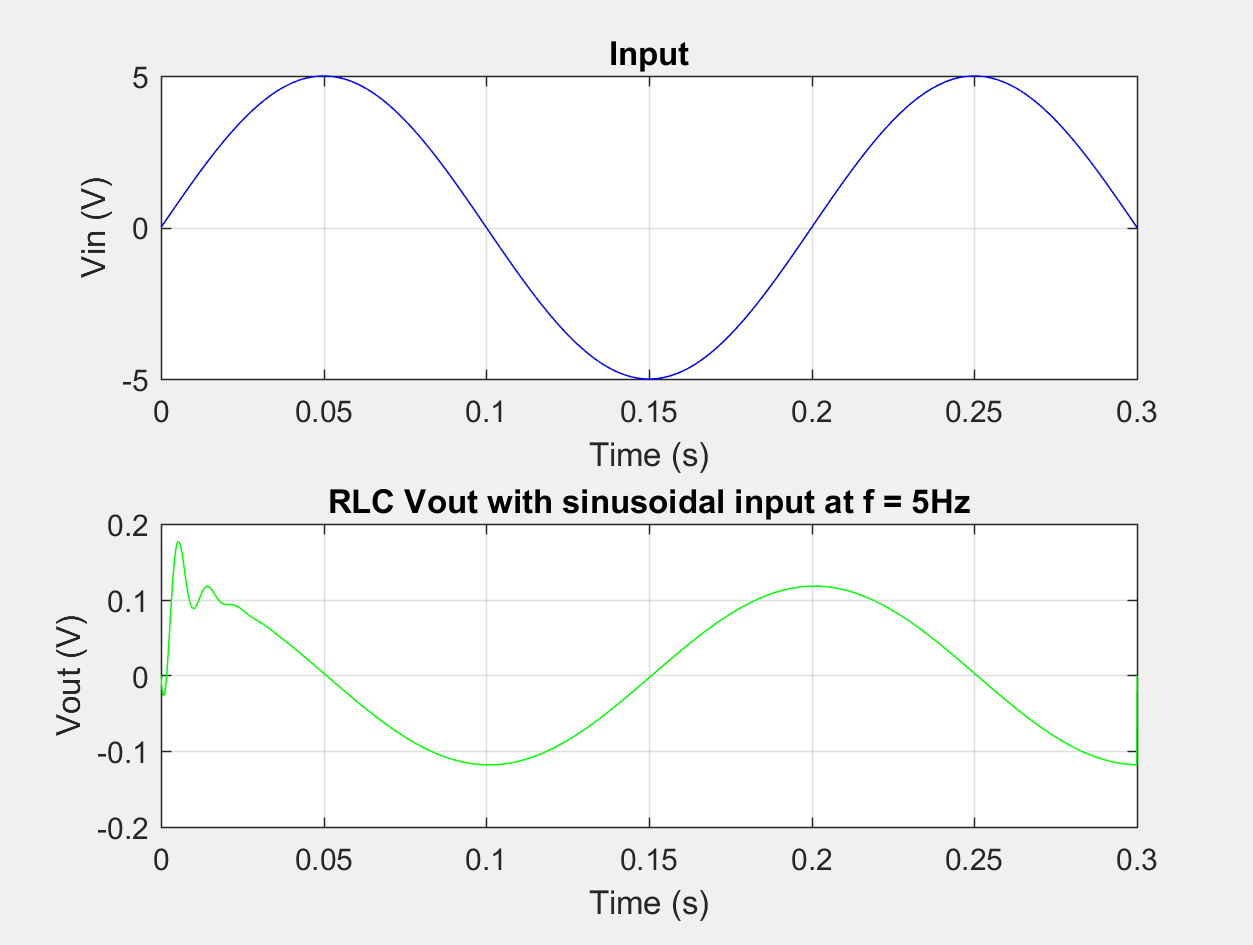


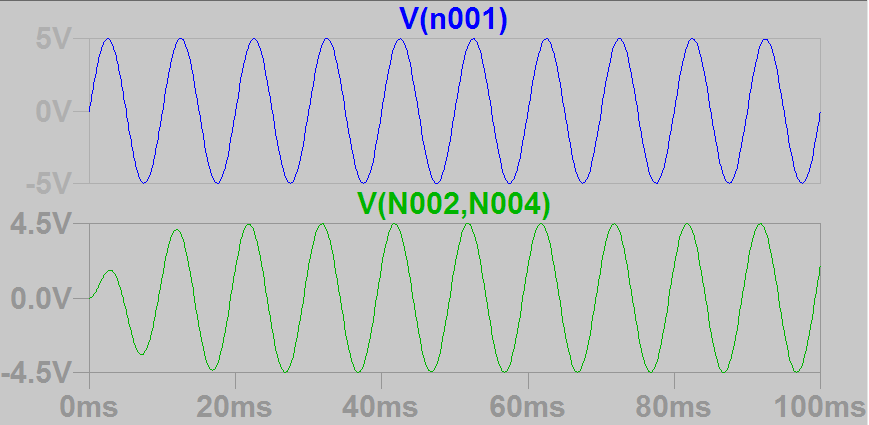
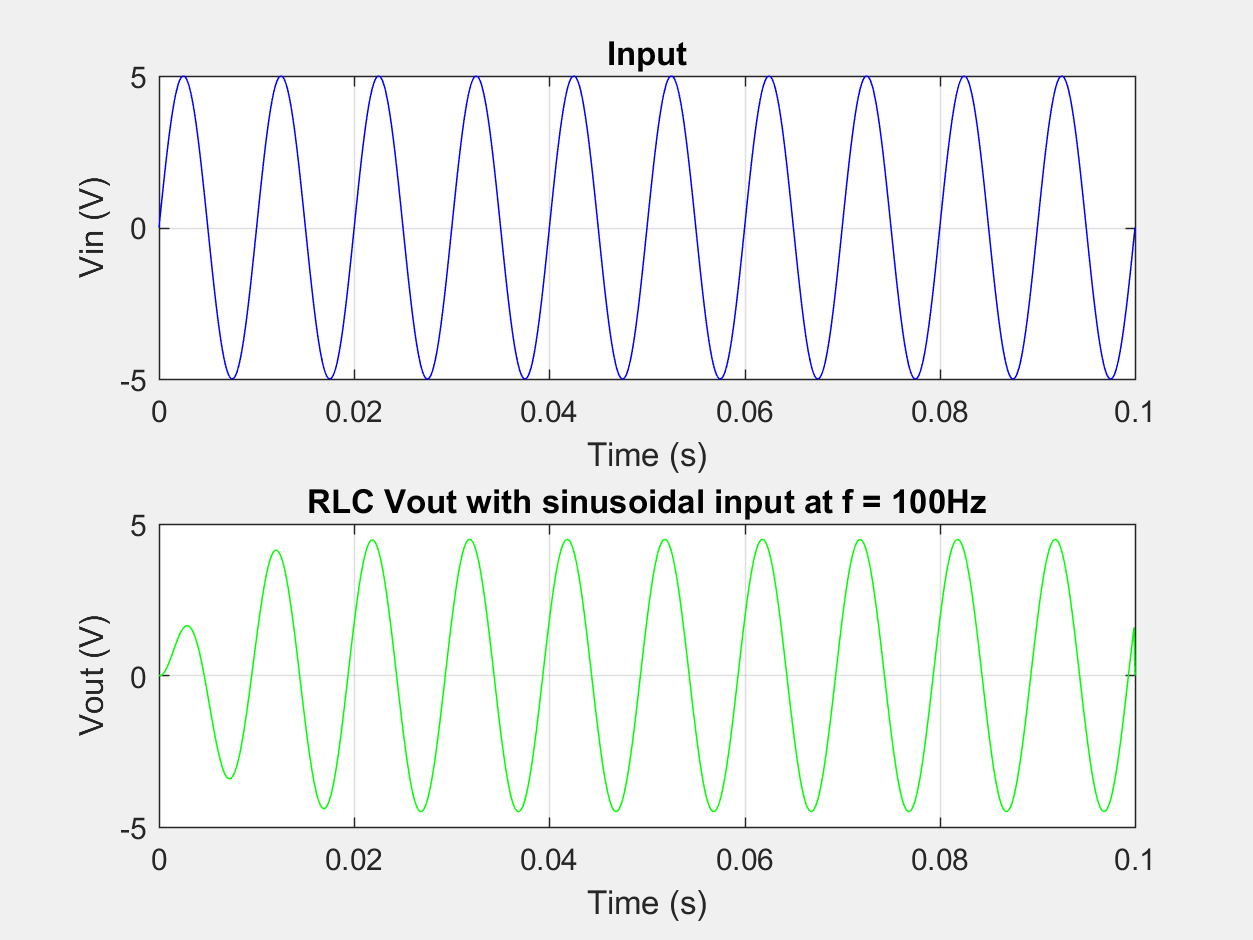
f=500Hz



Sine wave with amplitude Vin = 5 V and different frequencies f = 5 Hz, f = 100 Hz, f = 500 Hz.

It can be observed that the amplitude is highest at f = 100Hz. This agrees with the calculated resonance frequency of 105 Hz.

f=5

f=100

f=500