A

Project Report

On

ANALYTICAL AND NUMERICAL SOLUTION OF LC network USING MATLAB

Submitted By:

SUJIT BARHATE

RITESH KUMAR

Under the guidance of

Mr. Vipin Palande

Department of Technology

Electrical and Electronics

Morning batch

Pune – 411007.

2015-2016

CERTIFICATE

This is to certify that the following students have completed the project entitled '**ANALYTICAL AND NUMERICAL SOLUTION OF LC NETWORK USING MATLAB** 'in a satisfactory means under guidance of Mr. Vipin Palande . The project is found to be completed in fulfillment for the subject of First Year Mathematics for the E.E (MB).

Submitted By:

SUJIT BARHATE (EE15MP05)
RITESH KUMAR (EE15M04)

Dr. Vipin Palande

ACKNOWLEDGEMENT

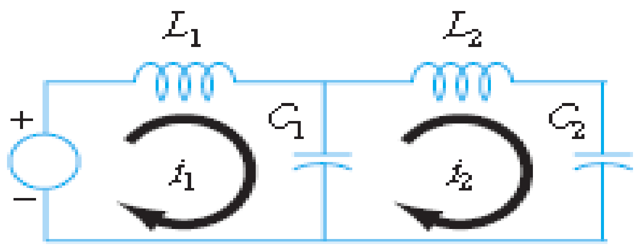
We take this opportunity to present our project report on '**ANALYTICAL AND NUMERICAL SOLUTION OF LC NETWORK USING MATLAB'**. We express our sincere thanks to our project guide **Mr. Vipin Palande** for his valuable help, guidance and the confidence, which he gave us at all stages of the project work.

INDEX

Contents Page No.

1. PROBLEM STATEMENT

Solve Inductance and capacitance system using analytical and numerical method. For analytical use Eigen values and Eigen vector system and numerical solve by Euler's method using matlab coding.



An LC network

In the above LC network.

No of component -

Inductor =2

Capacitor =2

Voltage source=1

2. Analytical solution

The network consists of inductor and capacitor with voltage source. There are two loops in the given network.

Applying KVL in both the loops –

Current i1 and i2 are following the network respectively.

Applying KVL in the 1st loop-

$$-L_1\frac{dI_1}{dt} - \frac{1}{C_1}\int_{-\infty}^{t} I_1 dt = 0$$

Applying KVL in the 2nd loop-

$$-L_2\frac{dI_2}{dt} - \frac{1}{C_2}\int_{-\infty}^{t} I_2 dt - \frac{1}{C_1}\int_{-\infty}^{t} (I_2 - I_1) dt = 0$$

$I_j = A_j \sin \omega t \quad \ldots \ldots \ldots \ldots \ldots (I)$

$V_C = \frac{1}{C} \int_{-\infty}^{t} I \, dt$

$\frac{dI_j}{dt} = A_j \omega \cos \omega t$

$$\frac{d^2 I_j}{dt^2} = -A_j \omega^2 \sin \omega t = -I_j \omega^2$$

Differentiate equations

$$-L_1 \frac{d^2 I_1}{dt^2} - \frac{I_1}{C_1} = 0$$

$$-L_2 \frac{d^2 I_2}{dt^2} - \frac{I_2}{C_2} - \frac{(I_2 - I_1)}{C_1} = 0$$

Rearranging the equations

$$L_1 \frac{d^2 I_1}{dt^2} + \frac{I_1}{C_1} = 0$$

$$L_2 \frac{d^2 I_2}{dt^2} + \frac{I_2}{C_2} + \frac{(I_2 - I_1)}{C_1} = 0$$

$\left(\frac{1}{c1} - w^2 L1\right) A1 = 0$ -----1

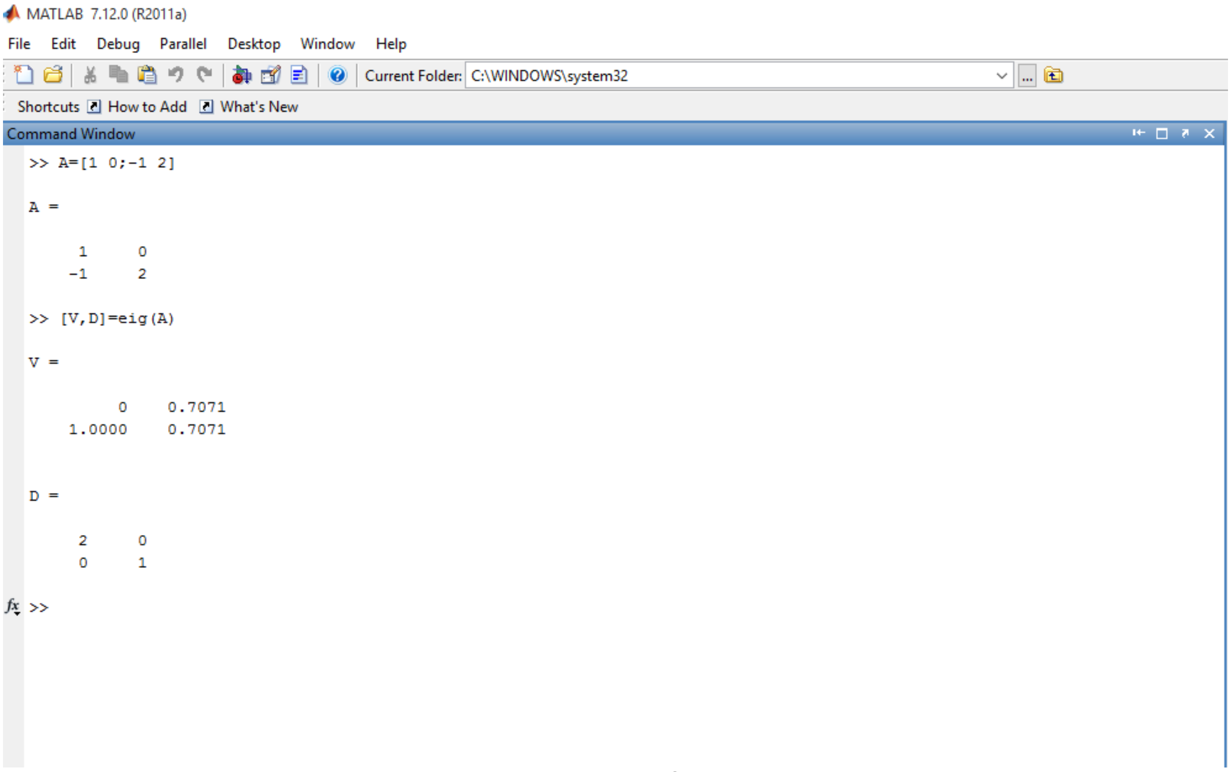
$-\frac{A1}{c1} + \left(\frac{1}{c1} + \frac{1}{c2} - w^2 L2\right) A2 = 0$ -------2

Solving equation 1 and 2 by Eigen value and Eigen vector

A = $\begin{matrix} 1 & 0 \\ -1 & 2 \end{matrix}$

$\begin{bmatrix} 1-\lambda & 0 \\ -1 & 2-\lambda \end{bmatrix} \begin{matrix} A1 \\ A2 \end{matrix} = 0$

Finding Eigen value and Eigen vector by MATLAB:-



$$\lambda = LCW^2$$

$$\frac{\lambda}{LC} = W^2$$

$$\sqrt[2]{\frac{\lambda}{LC}} = W$$

3. SOLVE USING EULER'S METHOD

In order to solve above problem using Euler's method, the equations are rearranged as follows:

Lets consider equation,

$$L_1 \frac{d^2 I_1}{dt^2} + \frac{I_1}{C_1} = 0$$

Let,

$$\frac{dI_1}{dt} = I_1^* \qquad \text{---------------------- 3}$$

$$\frac{dI_1^*}{dt} = \frac{d^2 I_1}{dt^2} = -\frac{I_1}{L_1 C_1} \qquad \text{----------------------4}$$

Lets consider another equation,

$$L_2 \frac{d^2 I_2}{dt^2} + \frac{I_2}{C_2} + \frac{(I_2 - I_1)}{C_1} = 0$$

Let,

$$\frac{dI_2}{dt} = I_2^* \qquad \text{---------------------5}$$

$$\frac{dI_2^*}{dt} = \frac{d^2 I_2}{dt^2} = -\frac{1}{L_2}\left[\frac{I_2}{C_2} + \frac{(I_2 - I_1)}{C_1}\right] \qquad \text{-------------6}$$

Let L_1, L_2, C_1,C_2 = 1

Substituting values in equations 3, 4, 5 & 6

$$\frac{dI_1}{dt} = I_1^* \qquad \text{---------------------7}$$

$$\frac{dI_1^*}{dt} = -I_1 \qquad \text{---------------------8}$$

$$\frac{dI_2}{dt} = I_2^* \qquad \text{---------------------9}$$

$$\frac{dI_2^*}{dt} = -2I_2 - I_1 \qquad \text{---------------------10}$$

Equations 7 & 8 and 9 & 10 can be solved using Euler's method.

Consider initial conditions $I_1 = 1, I_2 = 2, I_1^* = 2, I_2^* = 1$, Step = 0.1
50 iterations are performed using Matlab programs.

4. MATLAB CODE AND SOLUTIONS

4.1 Matlab code to solve equations 7 & 8

```
%Program to solve Differential equation using Euler's method
%The euation is: dI1/dt = I1*
%Mapping with the equations from network to the program:
%I = I1*
%Consider initial value of I as 2 and performing 50 iterations to solve the
%equation.

clear all;
I0= 2; %Initial value of I

iterations = 50;
dt = 0.1; %Delta t as Step Size

I = zeros(iterations,1); % this initializes the vector I to being all zeros
t = zeros(iterations,1);

I(1) = I0; % the initial condition
t(1) = 0.0;

for step=1:iterations-1 % Performing iterations
I(step+1) = I(step) + dt*(I(step));
t(step+1) = t(step) + dt;
end

I %print values of I

plot(t,I,'b'); %Plotting the result
title('Solution of First Differential Equation')
xlabel('Time')
ylabel('Current')
hold on; %keep the previously plotted lines

%Program to solve Differential equation using Euler's method
%The euation is: dI1/dt = -I1
%Mapping with the equations from network to the program:
%I = I1
%Consider initial value of I as 2 and performing 50 iterations to solve the
%equation.

Iini= 1; %Initial value of I

I1 = zeros(iterations,1); % this initializes the vector I to being all zeros
t = zeros(iterations,1);

I1(1) = Iini; % the initial condition
t(1) = 0.0;

for step=1:iterations-1 % Performing iterations
I1(step+1) = I1(step) - dt*(I1(step));
t(step+1) = t(step) + dt;
end
```

```
I1 %print values of I

plot(t,I1,'r'); %Plotting the result
title('Solution of First Differential Equation')
xlabel('Time')
ylabel('Current')
hold on; %keep the previously plotted lines
```

Output of the Program

Values of $\mathbf{I}_1^*$

```
Command Window
I =

    2.0000
    2.2000
    2.4200      23.8364
    2.6620      26.2200
    2.9282      28.8420
    3.2210      31.7262
    3.5431      34.8988
    3.8974      38.3887
    4.2872      42.2276
    4.7159      46.4503
    5.1875      51.0953
    5.7062      56.2049
    6.2769      61.8254
    6.9045      68.0079
    7.5950      74.8087
    8.3545      82.2896
    9.1899      90.5185
   10.1089      99.5704
   11.1198     109.5274
   12.2318     120.4801
   13.4550     132.5282
   14.8005     145.7810
   16.2805     160.3591
   17.9086     176.3950
   19.6995     194.0345
   21.6694     213.4379
   23.8364
```

Graph of I_1^*

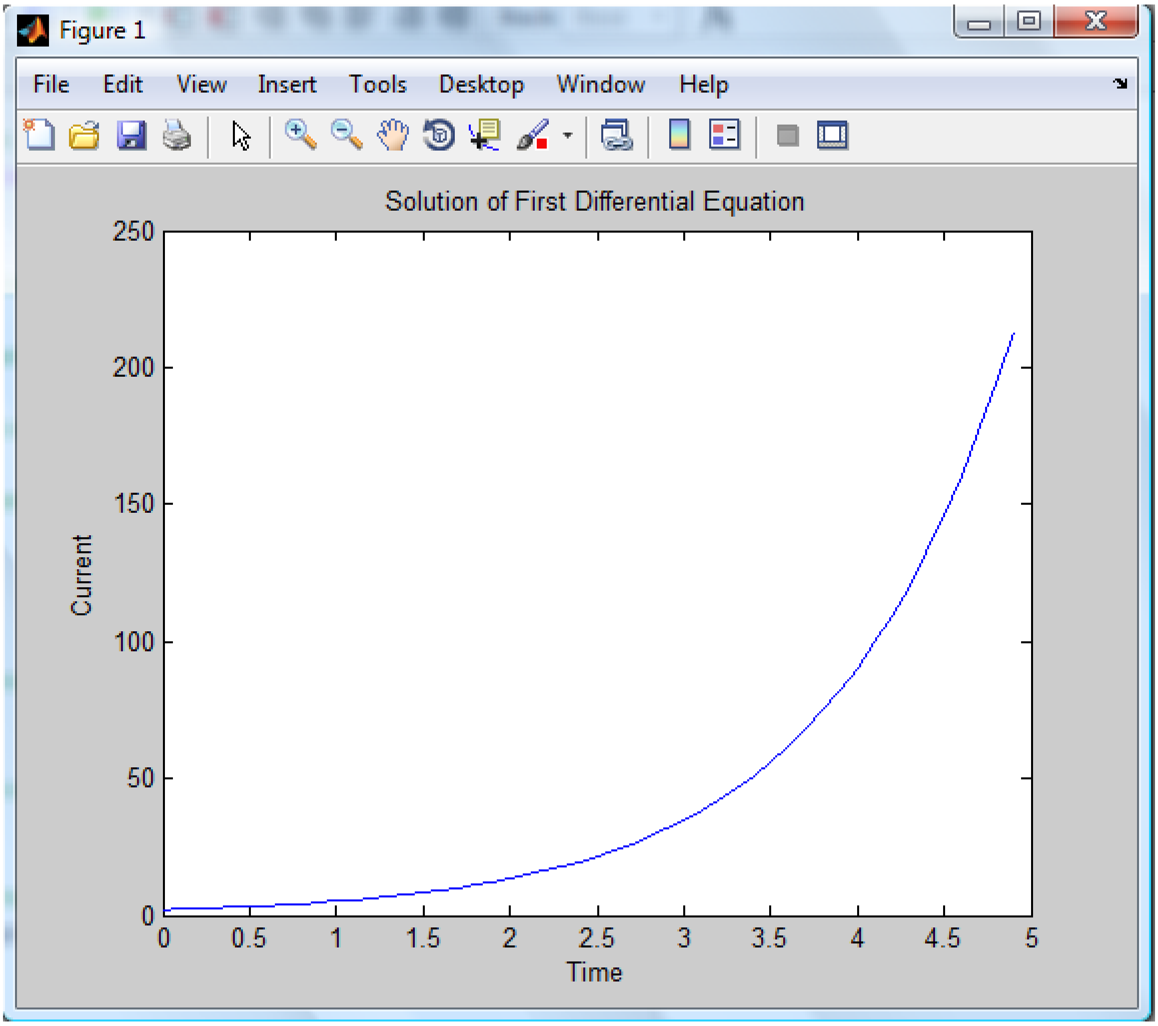


Figure 1
File Edit View Insert Tools Desktop Window Help
Solution of First Differential Equation
Current
Time
0
50
100
150
200
250
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

Values of I_1

Command Window

```
I1 =

    1.0000
    0.9000
    0.8100
    0.7290
    0.6561
    0.5905
    0.5314
    0.4783
    0.4305
    0.3874
    0.3487
    0.3138
    0.2824
    0.2542
    0.2288
    0.2059
    0.1853
    0.1668
    0.1501
    0.1351
    0.1216
    0.1094
    0.0985
    0.0886
    0.0798
    0.0718
    0.0646
    0.0581
    0.0523
    0.0471
    0.0424
    0.0382
    0.0343
    0.0309
    0.0278
    0.0250
    0.0225
    0.0203
    0.0182
    0.0164
    0.0148
    0.0133
    0.0120
    0.0108
    0.0097
    0.0087
    0.0079
    0.0071
    0.0064
    0.0057
```

Graph for I_1

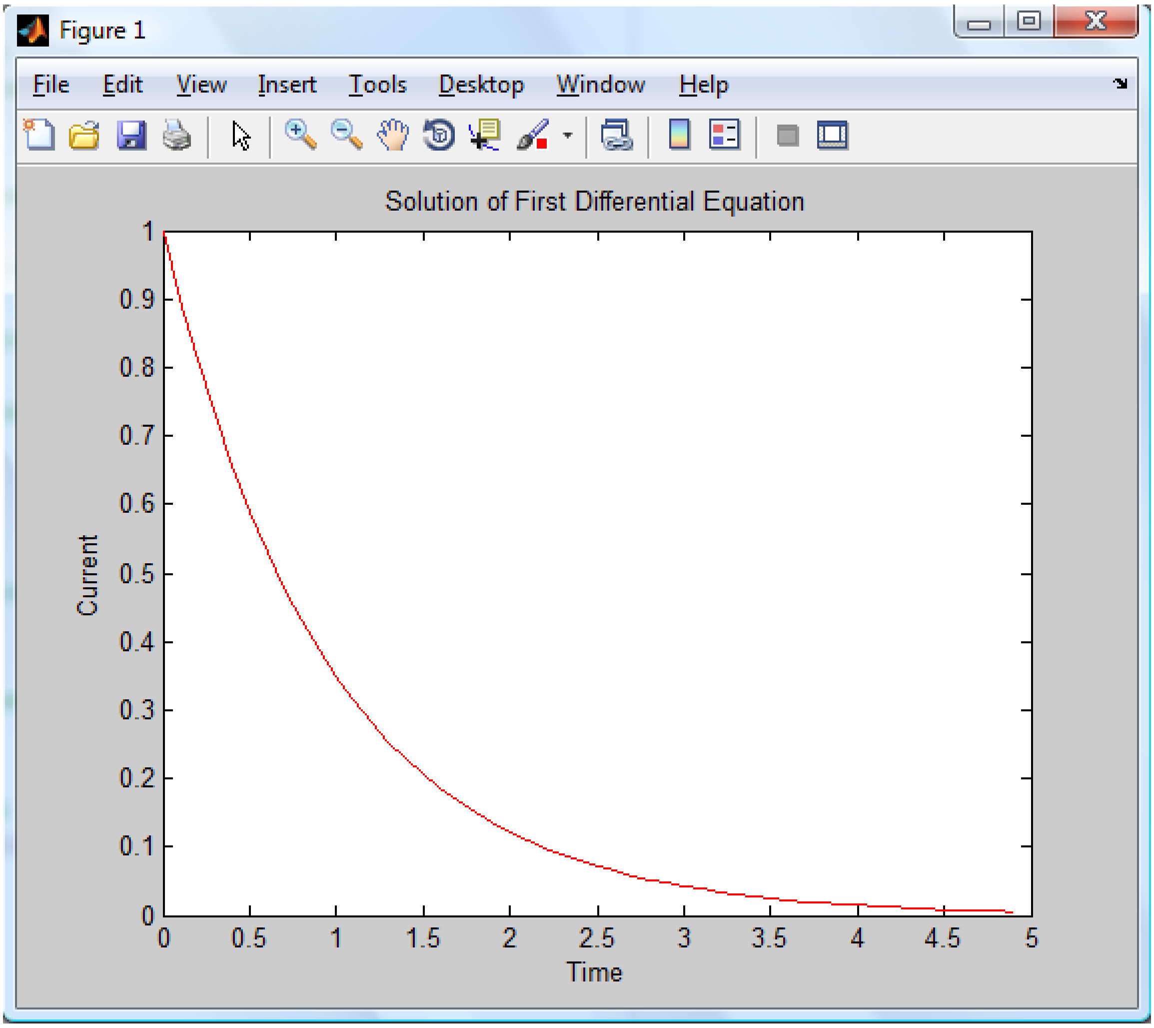
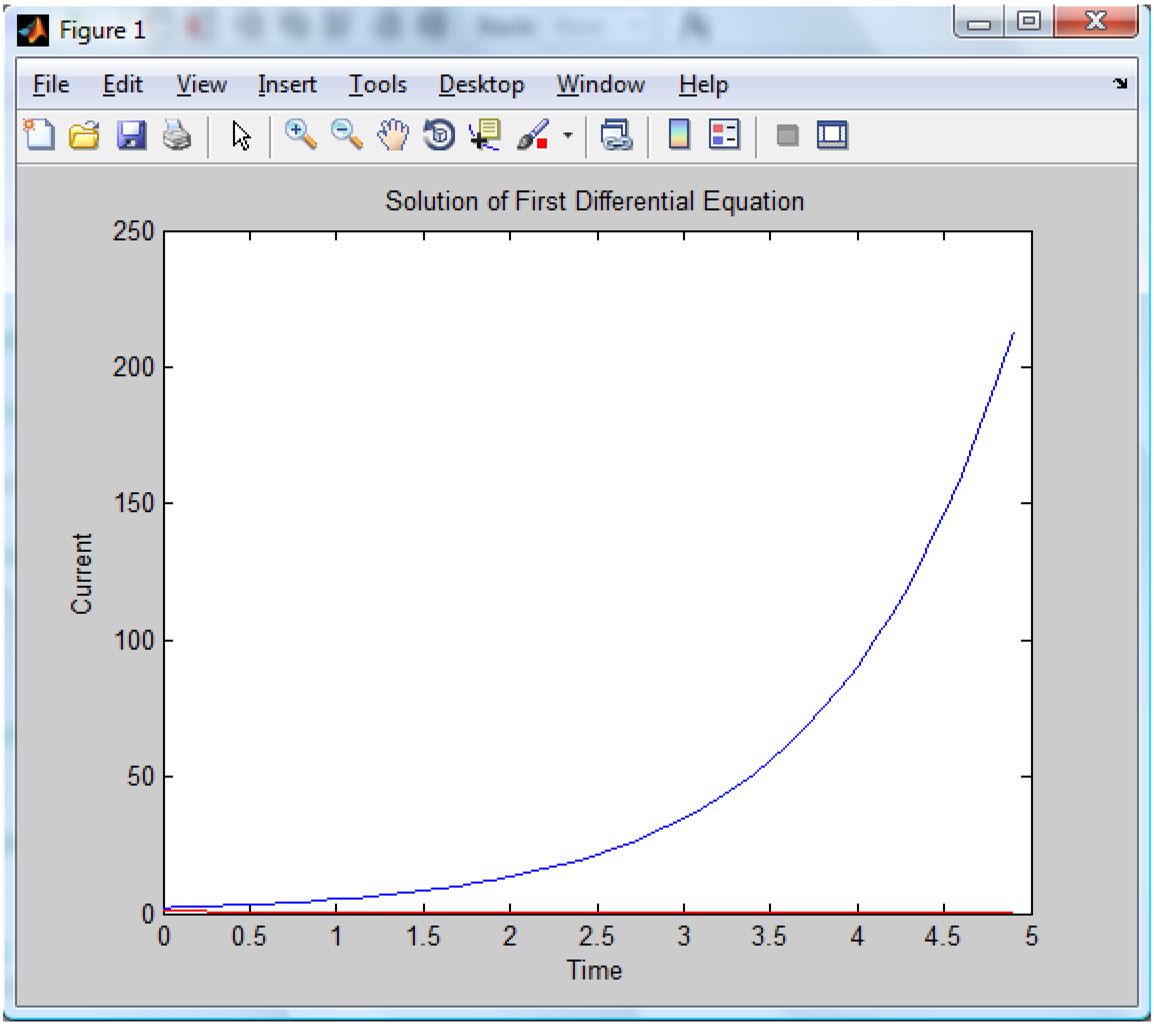


Figure 1
File Edit View Insert Tools Desktop Window Help
Solution of First Differential Equation
Current
Time
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0.5
1.5
2
2.5
3
3.5
4
4.5
5

Combined Graph of I_1^* and I_1



4.2 Matlab code to solve equations 9 & 10

```
%Program to solve Differential equation using Euler's method
%The euation is: dI2/dt = I2*
%Mapping with the equations from network to the program:
%dI2*/dt = -2I2-I1
%Consider initial value of I as 2 and performing 50 iterations to solve the
%equation.

clear all;
clc;

%Find vector I representing I1

I0= 1; %Initial value of I

iterations = 50;
dt = 0.1; %Delta t as Step Size

I = zeros(iterations,1); % this initializes the vector I to being all zeros
t = zeros(iterations,1);

I(1) = I0; % the initial condition
t(1) = 0.0;

for step=1:iterations-1 % Performing iterations
I(step+1) = I(step) + dt*(I(step));
t(step+1) = t(step) + dt;
end
I %print values of I
% Find current vector I1 representing I2 using vector I
% dI/dt = -2I1-I

Iini= 1; %Initial value of I1

%Program to solve Differential equation using Euler's method
%The euation is: dI2/dt = -2I2-I1
%Mapping with the equations from network to the program:
%dI2*/dt = -2I2-I1
%Consider initial value of I1 as 1 and performing 50 iterations to solve the
%equation.


I1 = zeros(iterations,1); % this initializes the vector I to being all zeros
t = zeros(iterations,1);

I1(1) = Iini; % the initial condition
t(1) = 0.0;

for step=1:iterations-1 % Performing iterations
I1(step+1) = I1(step) + dt*((-2*I1(step))-I(step));
t(step+1) = t(step) + dt;
end

I1 %print values of I1

plot(t,I1,'b',t,I,'r'); %Plotting the result
title('Solution of Second current loop Differential Equations')
```

```
xlabel('Time')
ylabel('Current')
```

OUTPUT OF THE PROGRAM

Values of $I=I_2^*$ and I_1

```
Command Window

I =

    1.0000
    1.1000
    1.2100
    1.3310
    1.4641
    1.6105
    1.7716
    1.9487
    2.1436
    2.3579
    2.5937
    2.8531
    3.1384
    3.4523
    3.7975
    4.1772
    4.5950
    5.0545
    5.5599
    6.1159
    6.7275
    7.4002
    8.1403
    8.9543
    9.8497
   10.8347
   11.9182
   13.1100
   14.4210
   15.8631
   17.4494
   19.1943
   21.1138
   23.2252
   25.5477
   28.1024
   30.9127
   34.0039
   37.4043
   41.1448
   45.2593
   49.7852
   54.7637
   60.2401
   66.2641
   72.8905
   80.1795
   88.1975
   97.0172
  106.7190

I1 =

    1.0000
    0.7000
    0.4500
    0.2390
    0.0581
   -0.0999
   -0.2410
   -0.3700
   -0.4908
   -0.6070
   -0.7214
   -0.8365
   -0.9545
   -1.0775
   -1.2072
   -1.3455
   -1.4941
   -1.6548
   -1.8293
   -2.0194
   -2.2271
   -2.4545
   -2.7036
   -2.9769
   -3.2769
   -3.6065
   -3.9687
   -4.3668
   -4.8044
   -5.2856
   -5.8148
   -6.3968
   -7.0369
   -7.7409
   -8.5152
   -9.3669
  -10.3038
  -11.3343
  -12.4678
  -13.7147
  -15.0862
  -16.5949
  -18.2545
  -20.0799
  -22.0880
  -24.2968
  -26.7265
  -29.3991
  -32.3390
  -35.5730
```

Graph

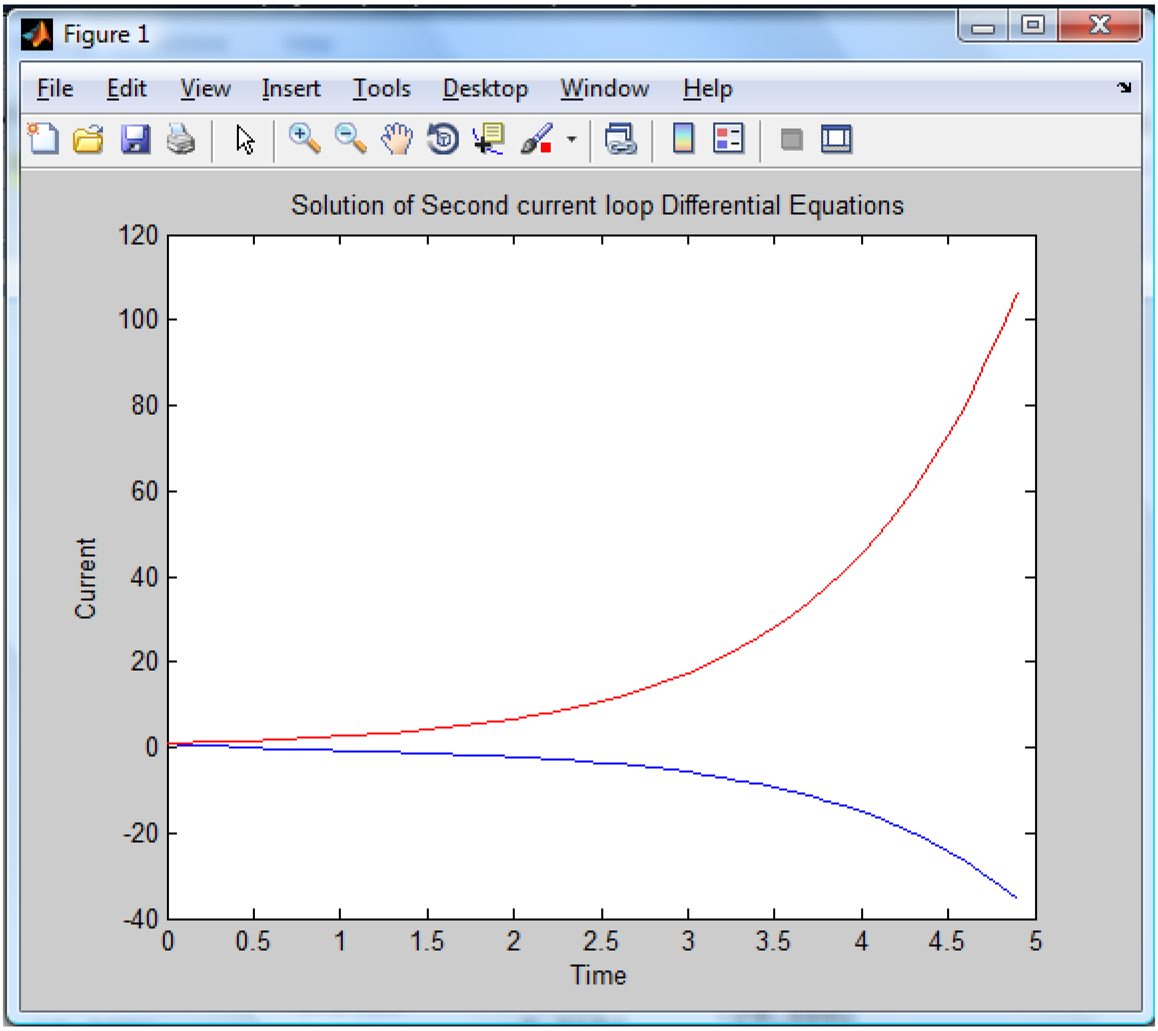


Figure 1
File
Edit
View
Insert
Tools
Desktop
Window
Help
Solution of Second current loop Differential Equations
Current
Time
120
100
80
60
40
20
0
-20
-40
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

5. CONCLUSION:

In the project, we have solved LC network using analytical and numerical method.
In the analytical method we have used Eigen value and Eigen vector method to calculate current in the given network.
For numerical method we have used Euler's method in the MATLAB to calculate the current in the given network. In the MATLAB program we also calculated graph between current and time.

6. REFERENCES:

1. Numerical Methods for Engineers (6th edition) by Steven C. Chapra and Raumond P. Canale
2. Advanced Engineering Mathematics (10th Edition) by Erwin Kreyszig
3. Numerical Computing with Matlab by Cleve Moler