Scilab Textbook Companion for Engineering Circuit Analysis by W. Hayt, J. Kemmerly And S. Durbin[1]

Created by
Dhaivat Udayan Mandavia
B.E
Electronics Engineering
B.V.B.College of Engineering and Technology
College Teacher
Prof.sujata Kotabagi
Cross-Checked by

July 31, 2019

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Textbook Companion and Scilab codes written in it can be downloaded from the "Textbook Companion Project" section at the website http://scilab.in

Book Description

Title: Engineering Circuit Analysis

Author: W. Hayt, J. Kemmerly And S. Durbin

Publisher: Tata McGraw - Hill Education, New Delhi

Edition: 6

Year: 2008

ISBN: 0070611054

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes

List of Figures

Chapter 2

Basic components and Electric Circuits

Scilab code Exa 2.1 Power

```
//Example 2.1
//Computation of power absorbed by each part
//From figure 2.13a
V=2;I=3;
//We have Power(P)=V*I
P=V*I
printf("a) Power =%dW\n",P)
if P>0 then
    printf("Power is absorbed by the element\n")
else
    printf("Power is supplied by the element\n");
end

clear P;
//From figure 2.13b
V=-2;I=-3;
//We have Power(P)=V*I
P=V*I
printf("b) Power =%dW\n",P)
```

```
if P>0 then
    printf("Power is absorbed by the element\n")
else
    printf("Power is supplied by the element\n")
end

//From figure 2.13c
V=4;I=-5;
//We have Power(P)=V*I
P=V*I
printf("c) Power =%dW\n",P)
if P>0 then
    printf("Power is absorbed by the element\n")
else
    printf("Power is supplied by the element\n")
end
```

Scilab code Exa 2.2 `Dependent sources`

```
//Example 2.2
//Calculate vL
disp("Given")
disp("v2=3V")
v2=3;
//From figure 2.19b
disp("Considering the right hand part of the circuit
    ")
disp("vL=5v2")
vL=5*v2;
disp("On substitution")
printf("vL=%dV\n",vL);
```

Scilab code Exa 2.3 `Ohm law`

```
//Example 2.3
//Calculate the voltage and power dissipated acreoss
    the resistor terminals
//From figure 2.24b
disp("Given")
disp("R=560 ohm ; i=428mA")
R=560;i=428*10^-3;
//Voltage across a resistor is
disp("v=R*i")
v=R*i;
printf("Voltage across a resistor=%3.3fV\n",v)

//Power dissipated by the resistor is
disp("p=v*i")
p=v*i;
printf("Power dissipated by the resistor=%3.3fW\n",p
    )
```

Scilab code Exa 2.4 `Ohm law`

```
//Example 2.4
//Calculate the power dissipated within the wire
//From figure 2.27
disp("Given")
disp("Total length of the wire is 4000 feet")
disp("Current drawn by lamp is 100A")
//Considering American Wire Gauge system(AWG)
//Referring Table 2.4
disp("4AWG=0.2485ohms/1000ft")
l=4000; i=100 ; rl=0.2485/1000;
//Let R be the wire resistance
R=l*rl;
//Let p be the power dissipated within the wire
disp("p=i^2*R")
p=i^2*R
```

```
printf("Power dissipated within the wire=%dW\n",p)
```

Chapter 3

Voltage and Current laws

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

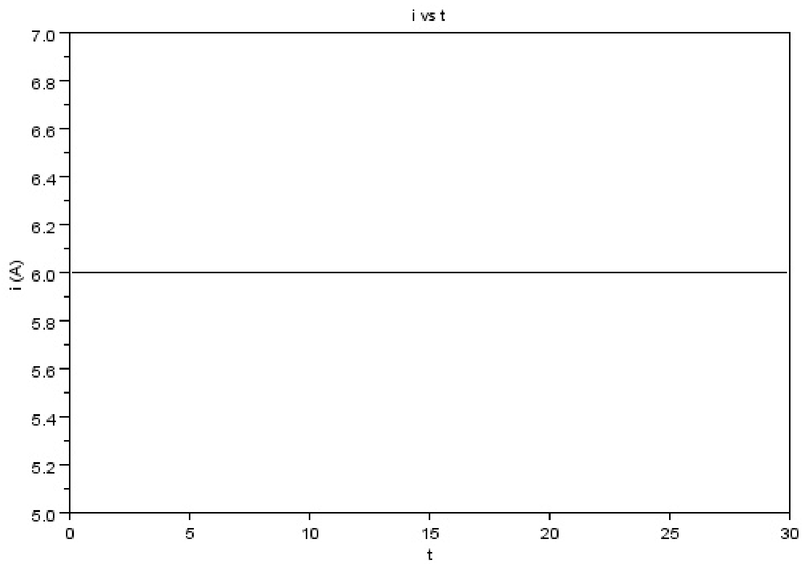

Figure 3.1: Kirchoff current law

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

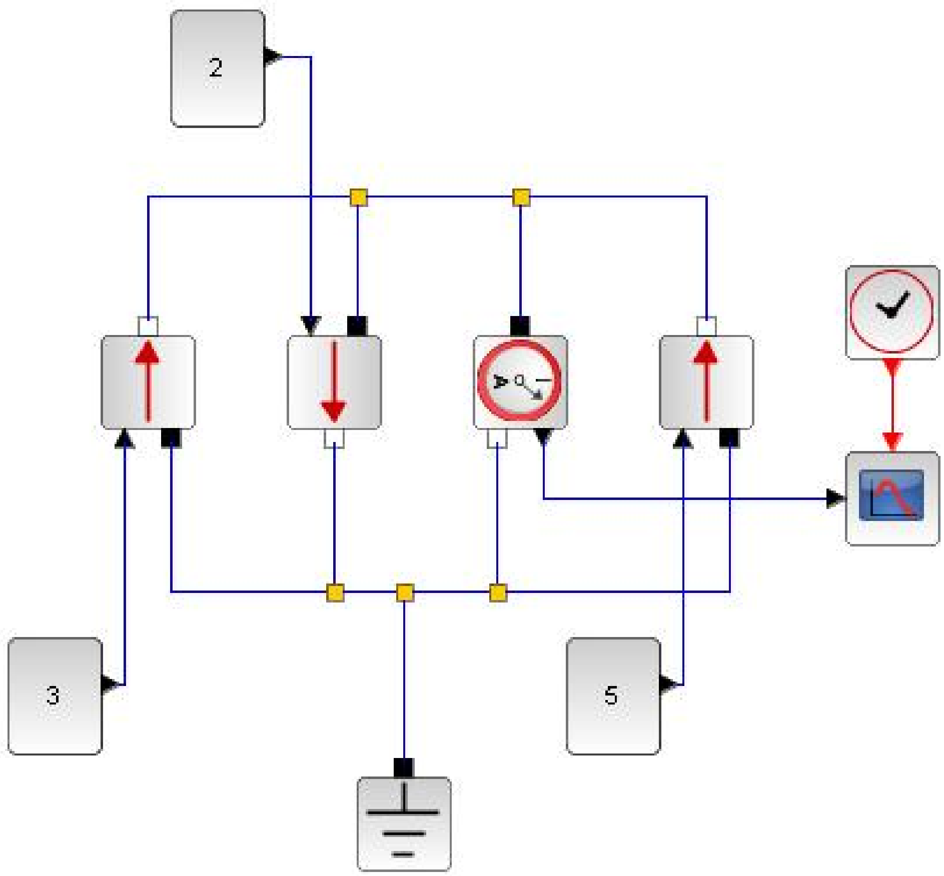

Figure 3.2: Kirchoff current law

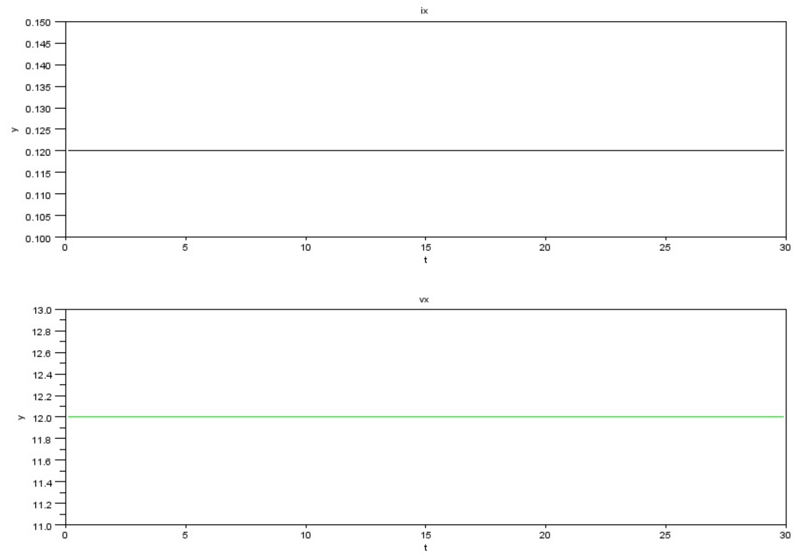

Figure 3.3: Kirchoff voltage law

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

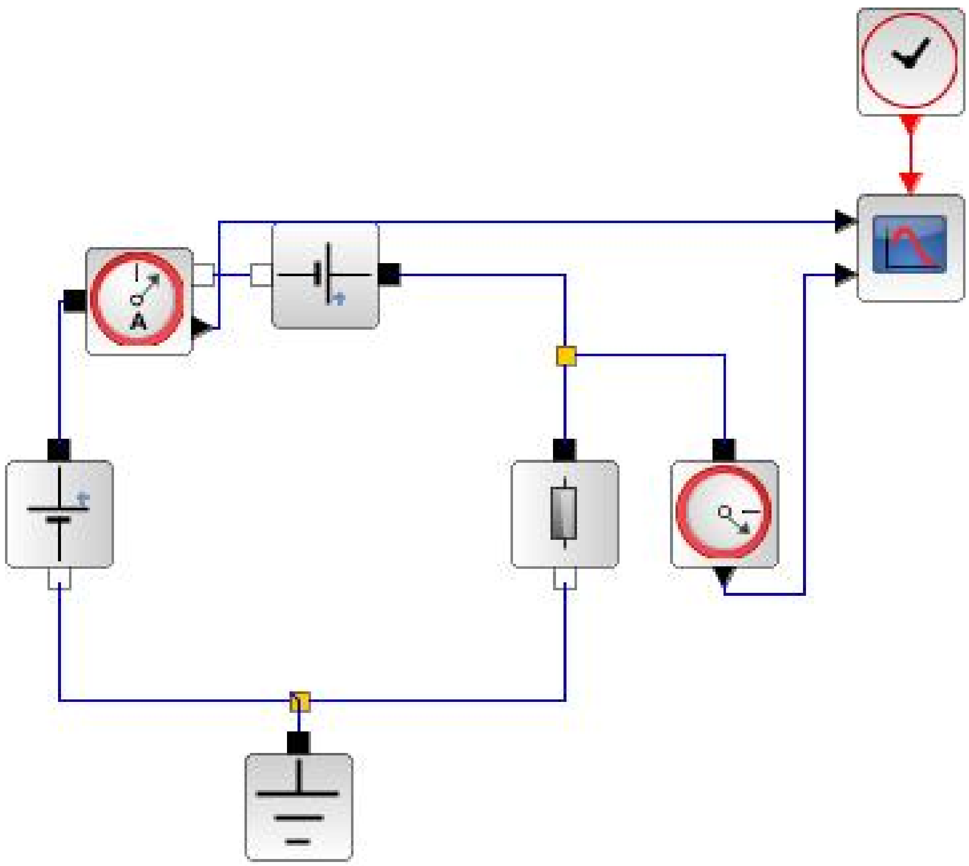

Figure 3.4: Kirchoff voltage law

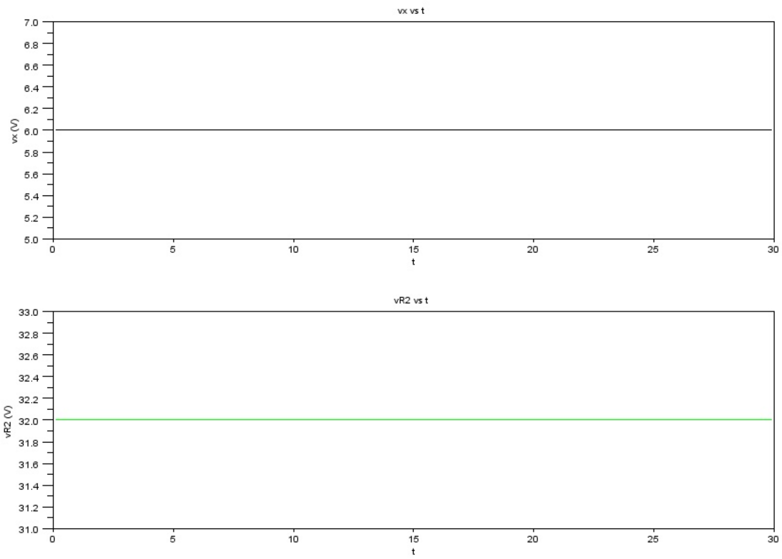

Figure 3.5: Kirchoff voltage law

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

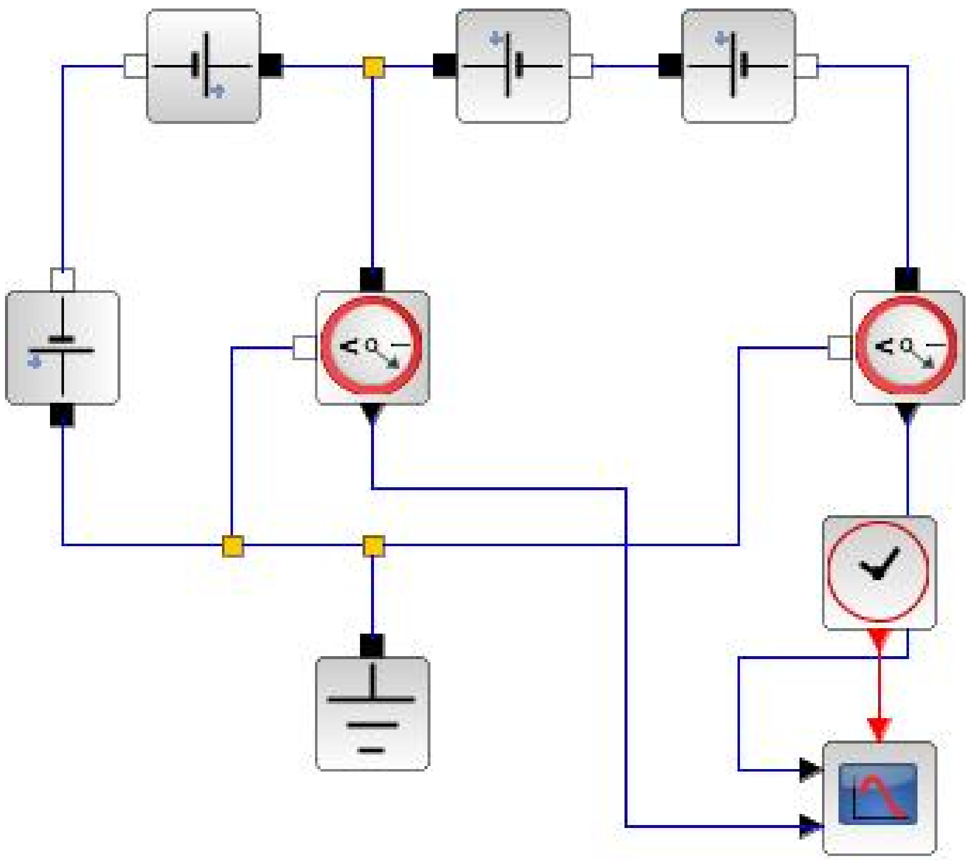

Figure 3.6: Kirchoff voltage law

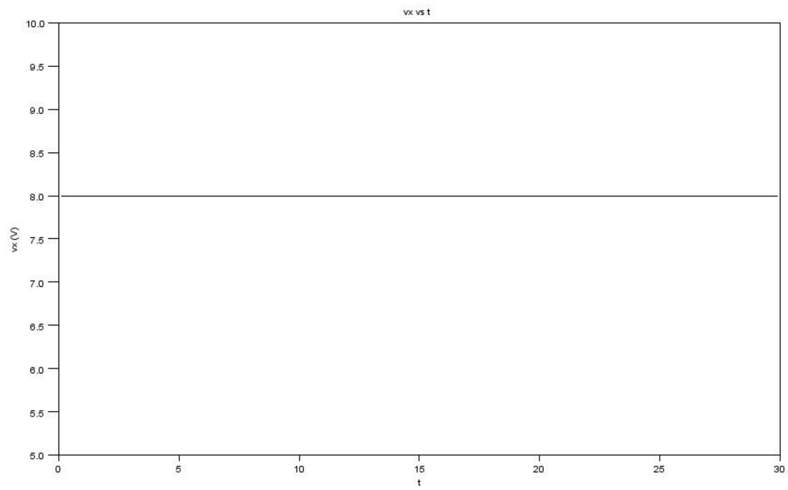

Figure 3.7: Kirchoff voltage law

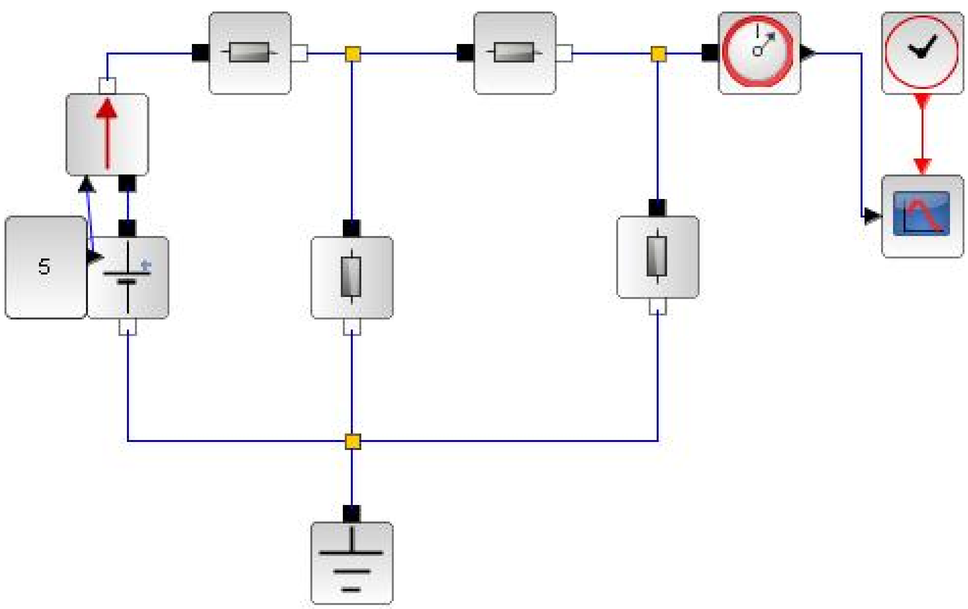

Figure 3.8: Kirchoff voltage law

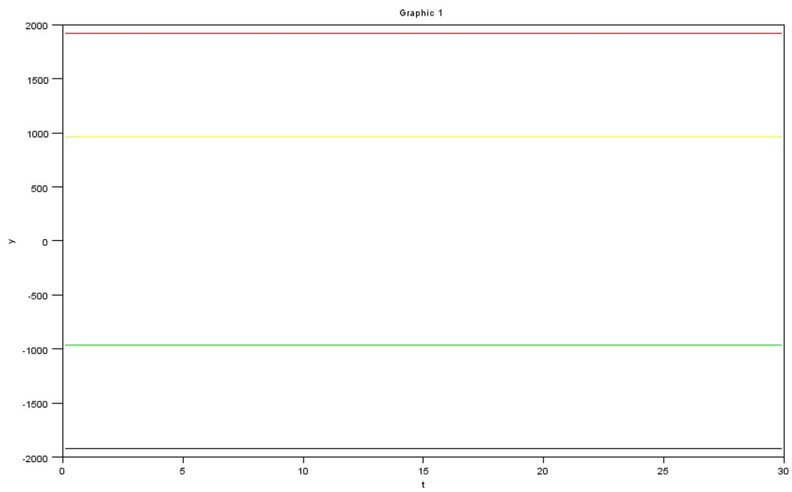

Figure 3.9: The Single Loop Circuit

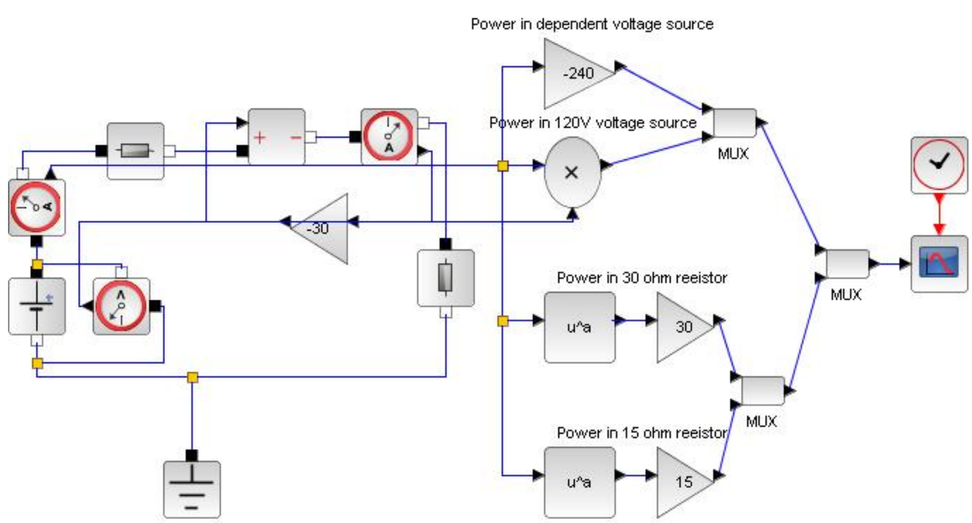

Figure 3.10: The Single Loop Circuit

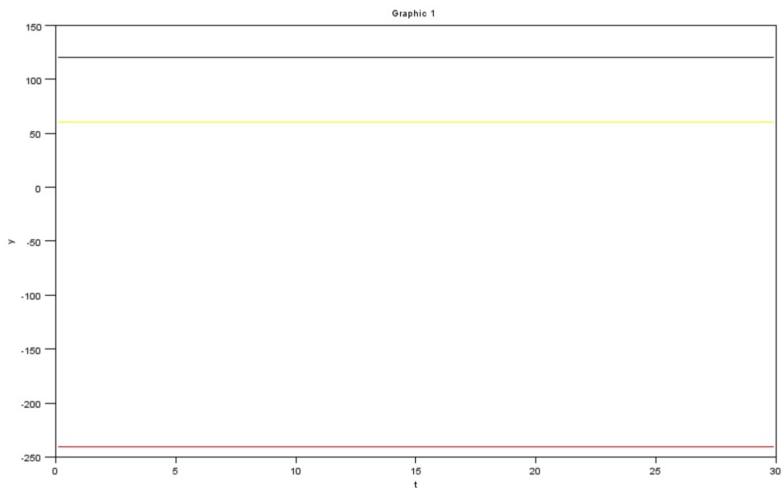

Figure 3.11: The single node pair circuit

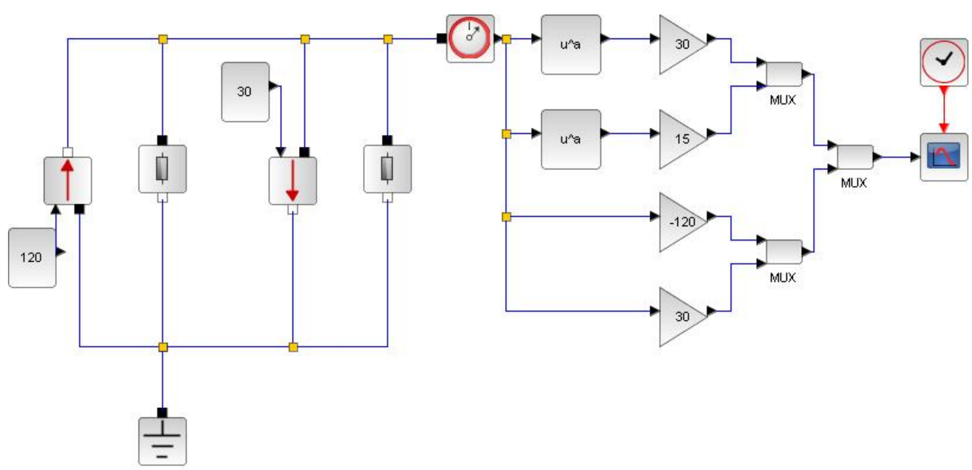

Figure 3.12: The single node pair circuit

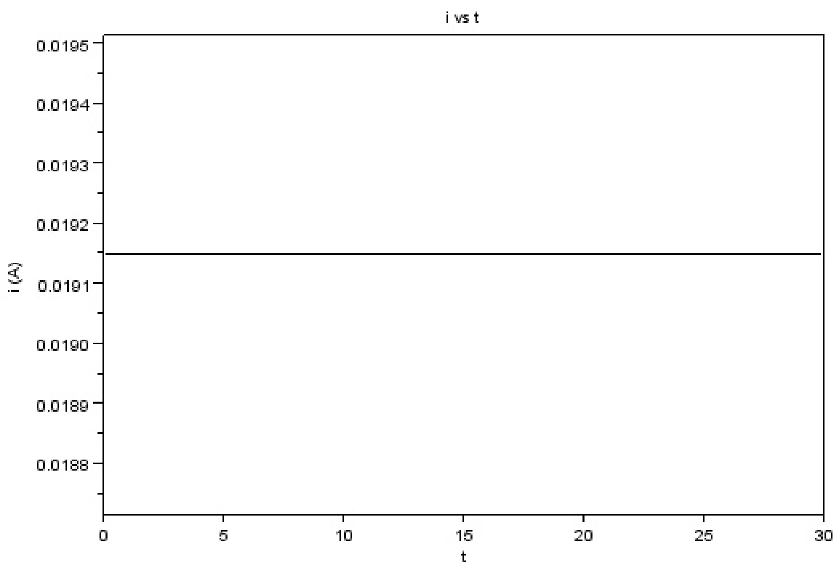

Figure 3.13: Series and Parallel connected sources

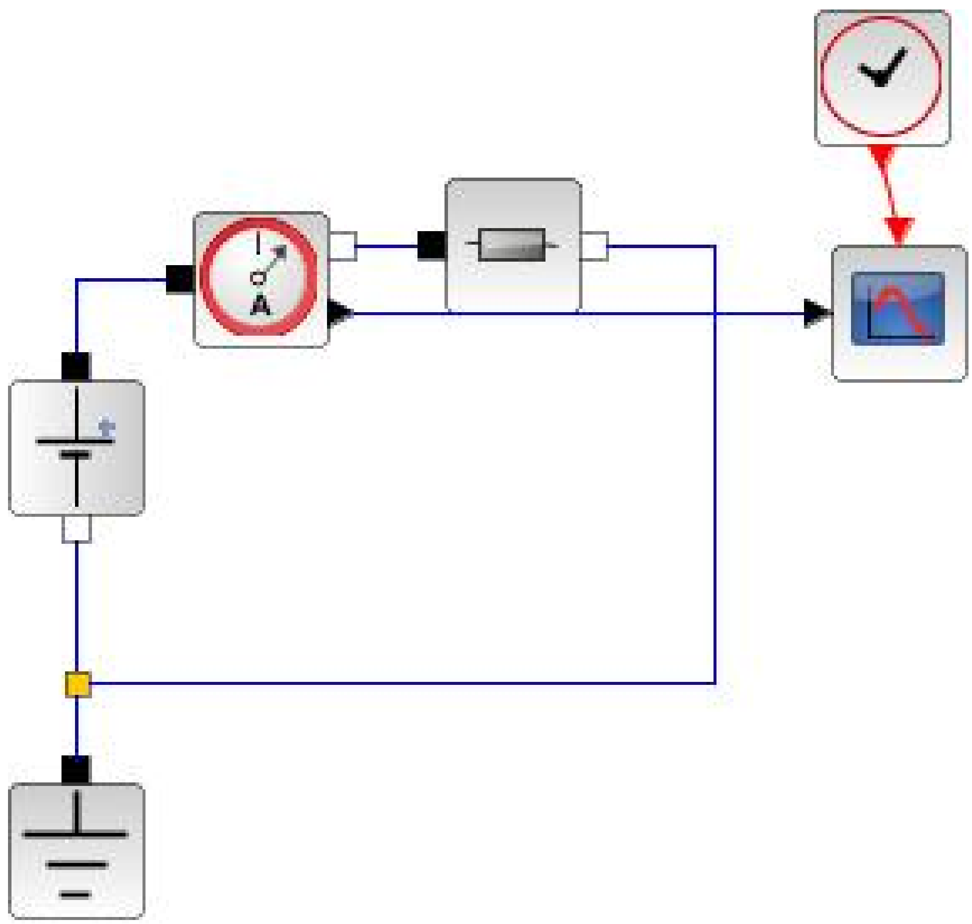

Figure 3.14: Series and Parallel connected sources

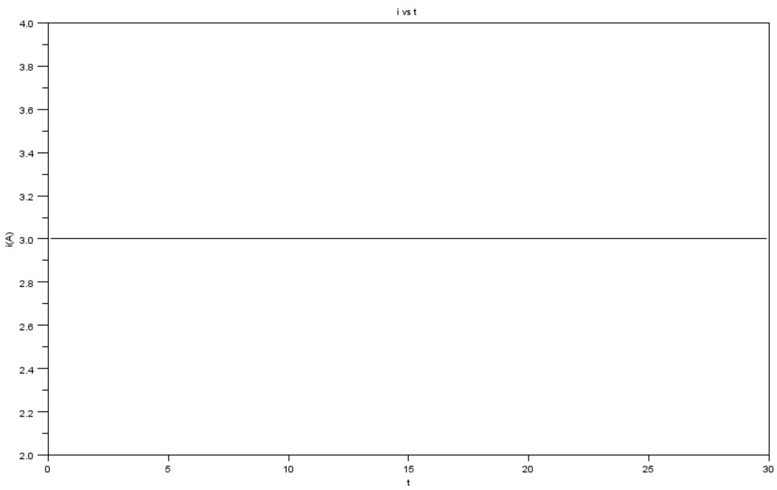

Figure 3.15: Resistors in series and parallel

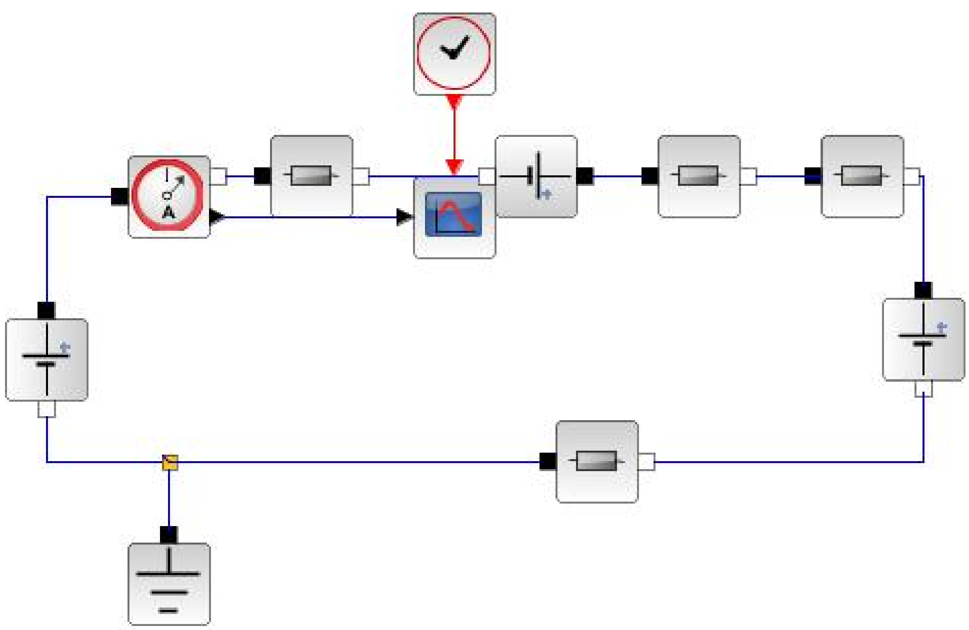

Figure 3.16: Resistors in series and parallel

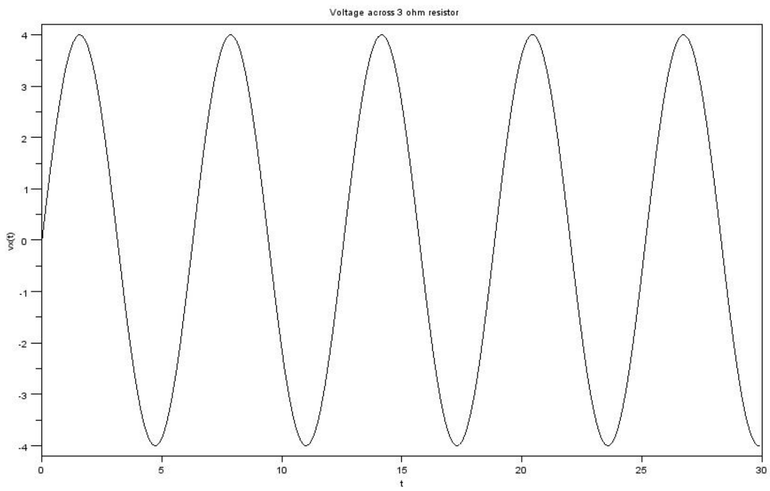

Figure 3.17: Voltage and Current division

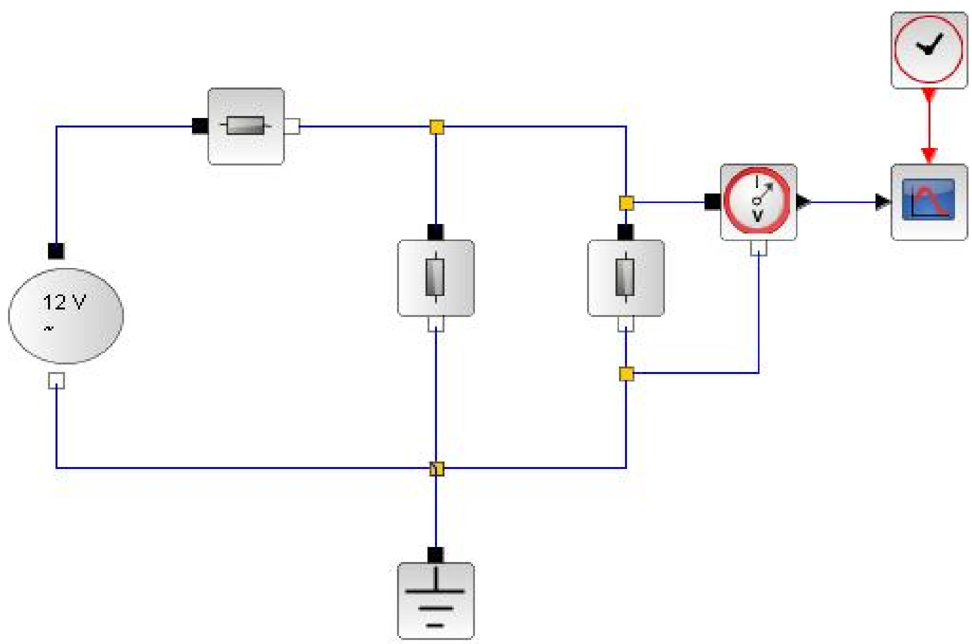

Figure 3.18: Voltage and Current division

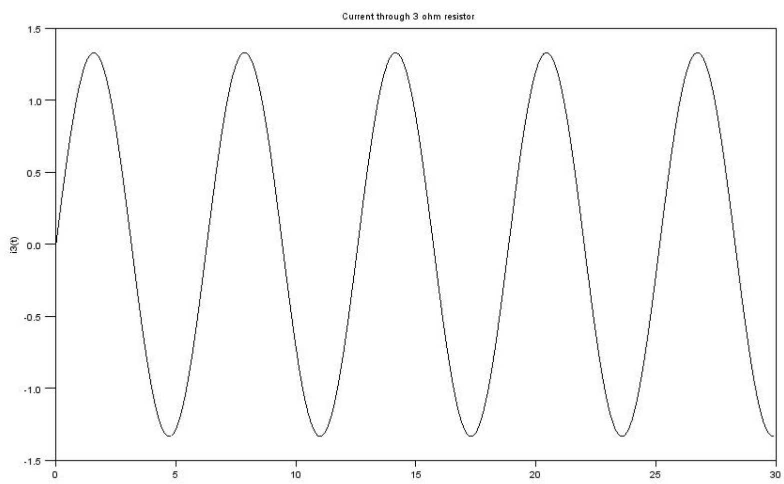

Figure 3.19: Voltage and Current division

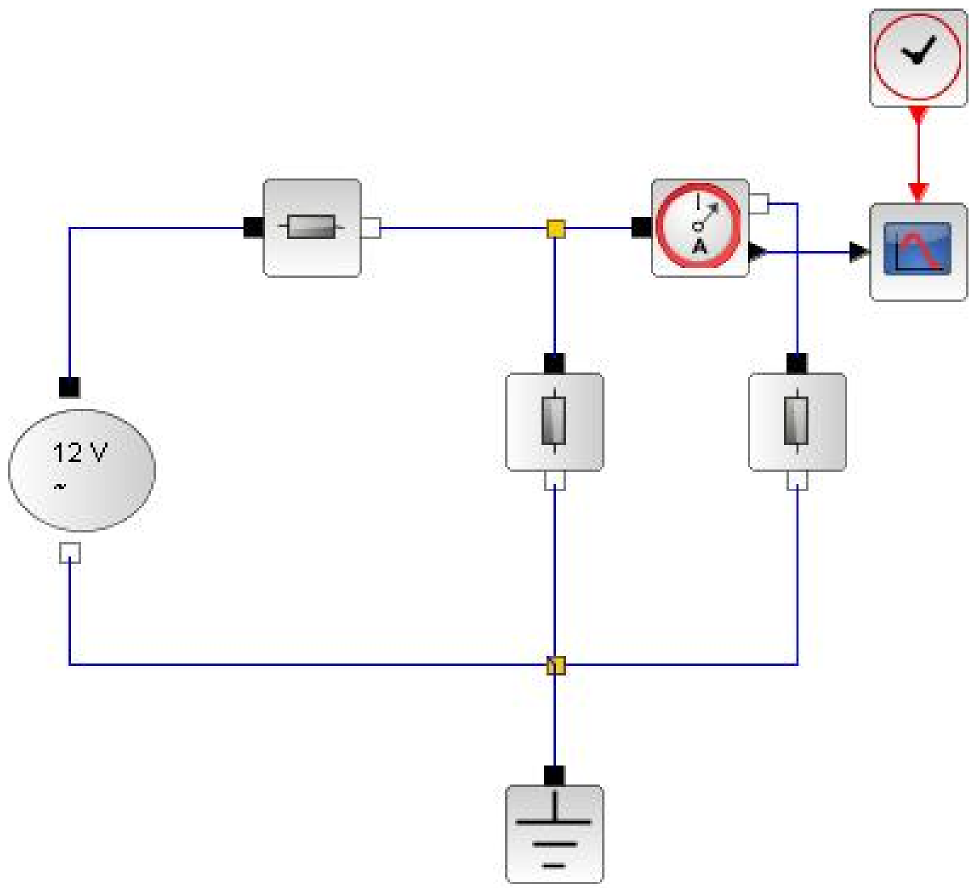

Figure 3.20: Voltage and Current division

Chapter 4

Basic Nodal and Mesh Analysis

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

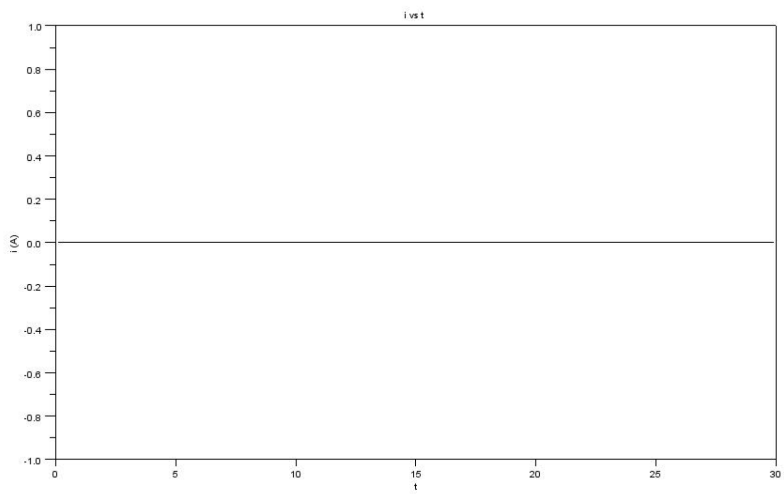

Figure 4.1: Nodal analysis

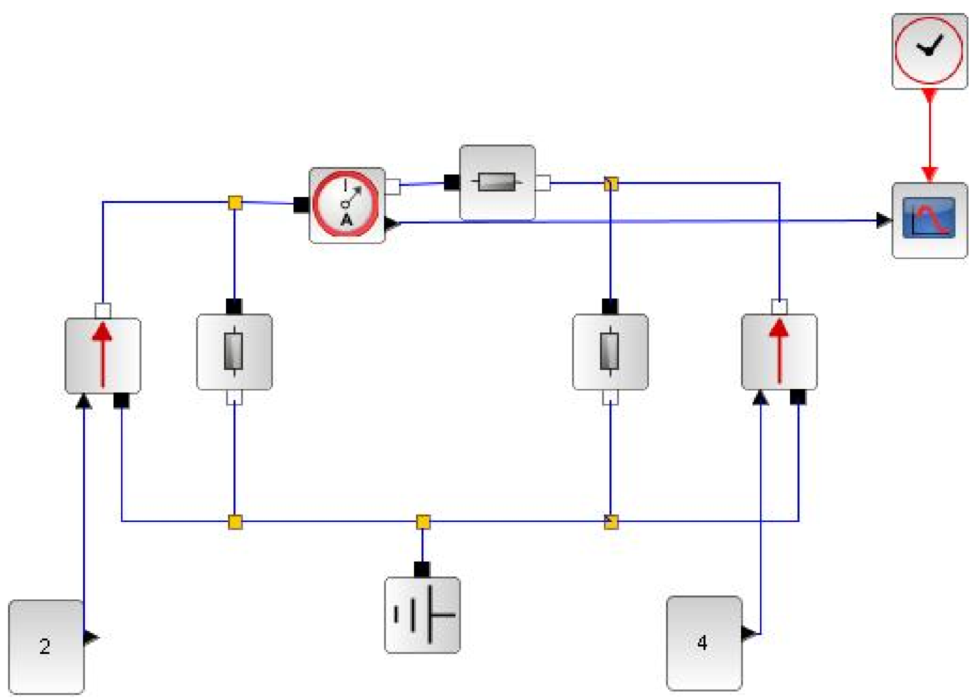

Figure 4.2: Nodal analysis

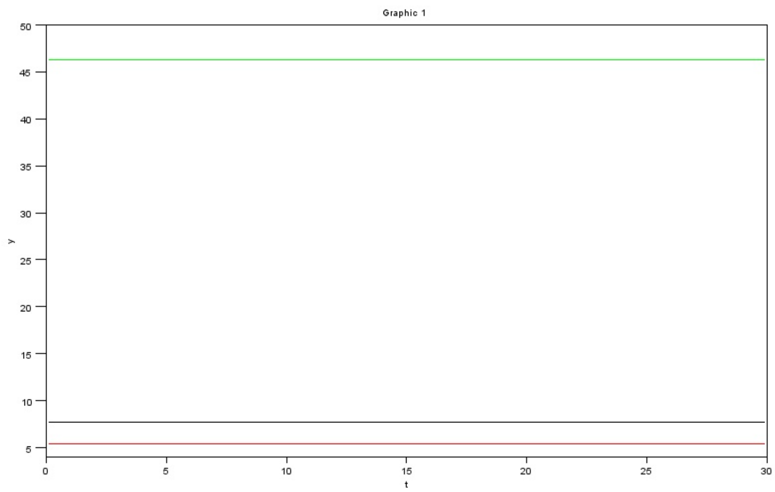

Figure 4.3: Nodal analysis

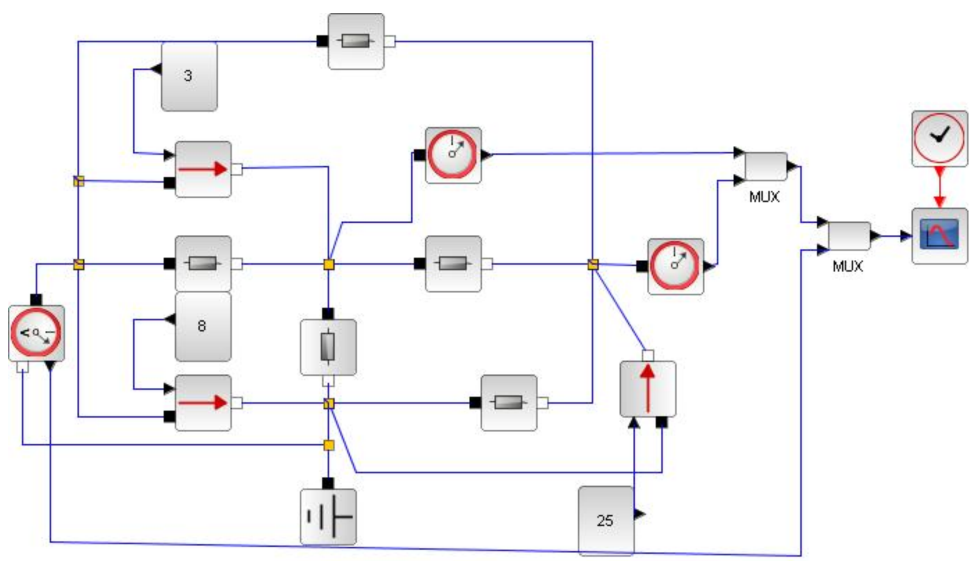

Figure 4.4: Nodal analysis

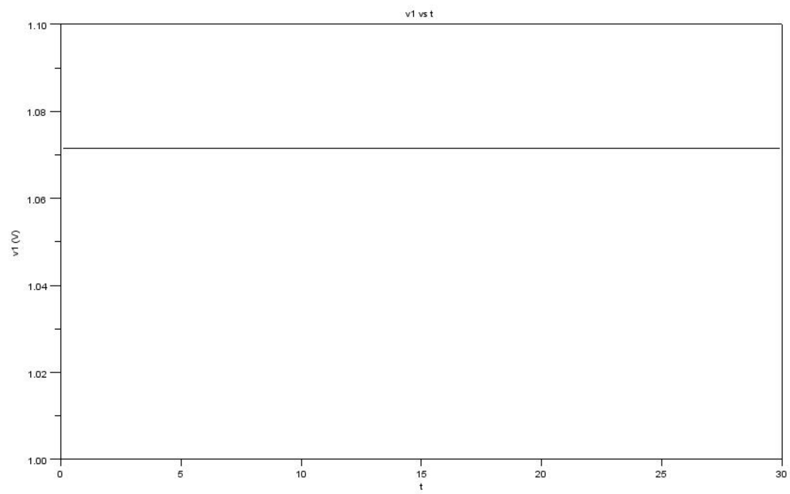

Figure 4.5: The supernode

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

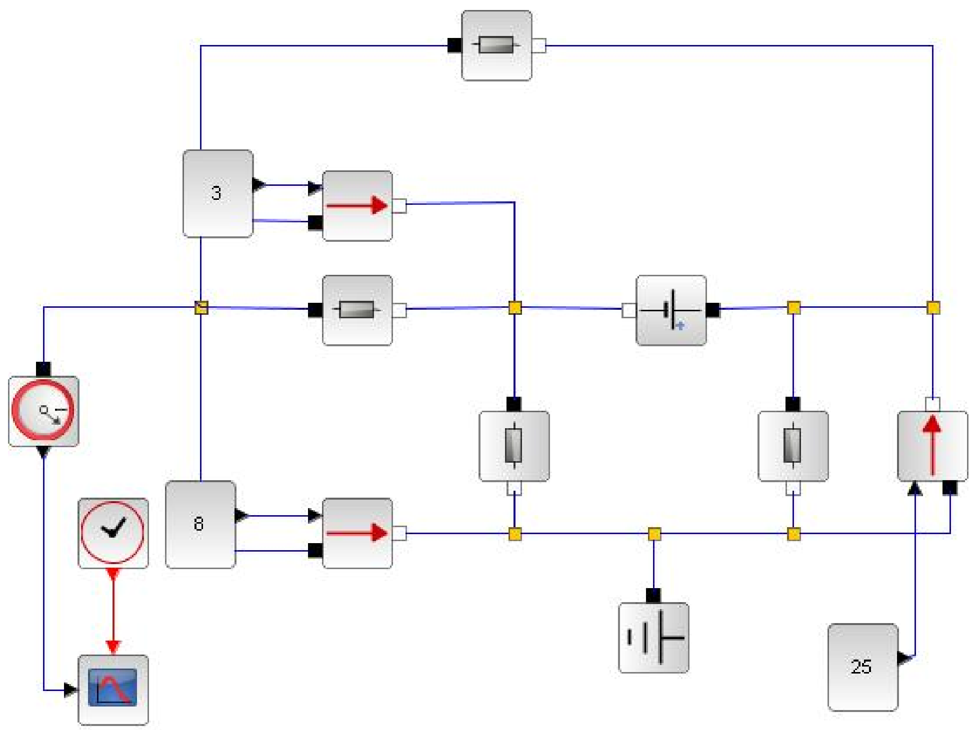

Figure 4.6: The supernode

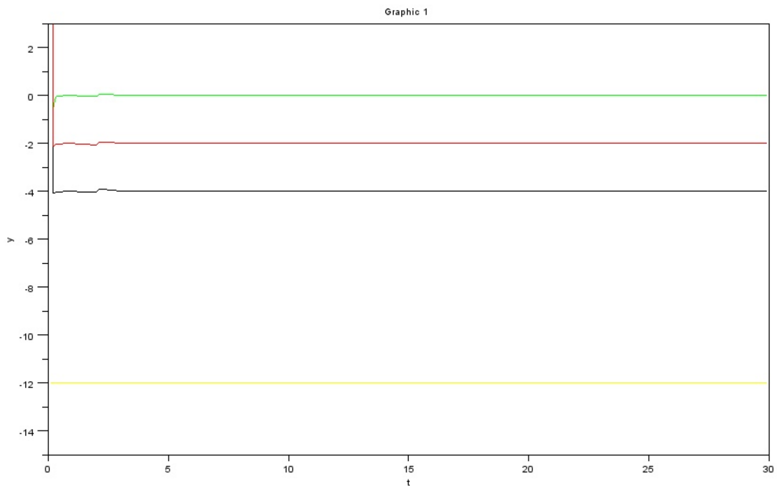

Figure 4.7: The supernode

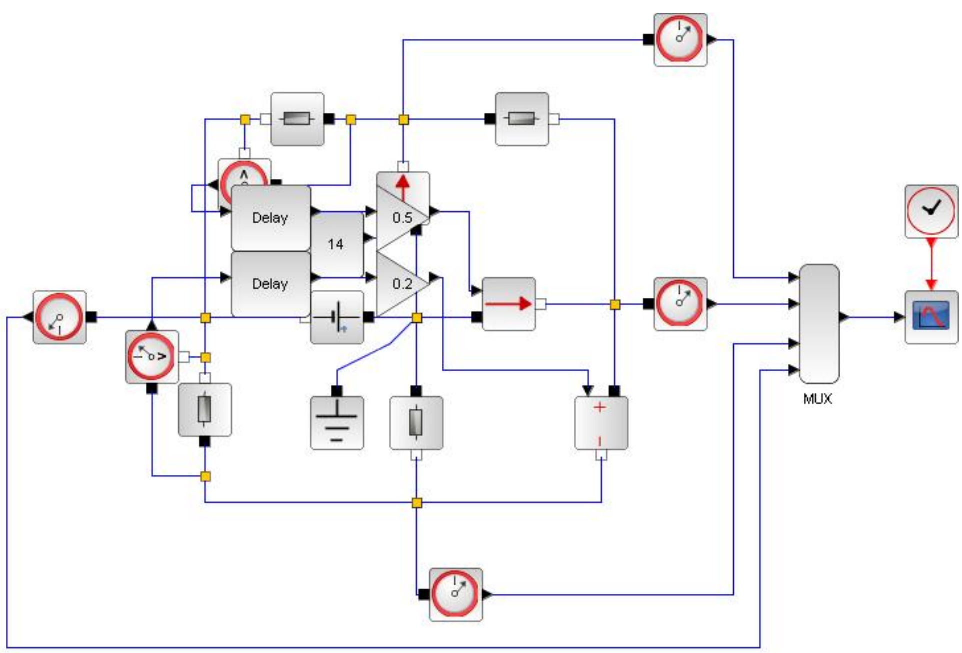

Figure 4.8: The supernode

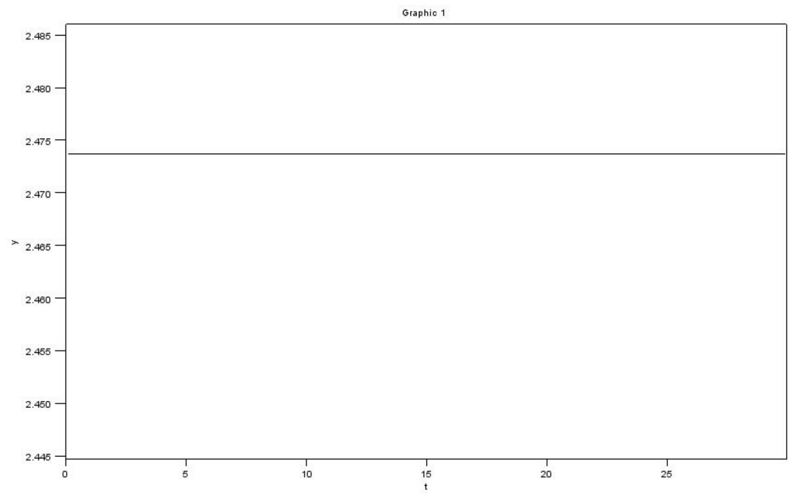

Figure 4.9: Mesh analysis

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

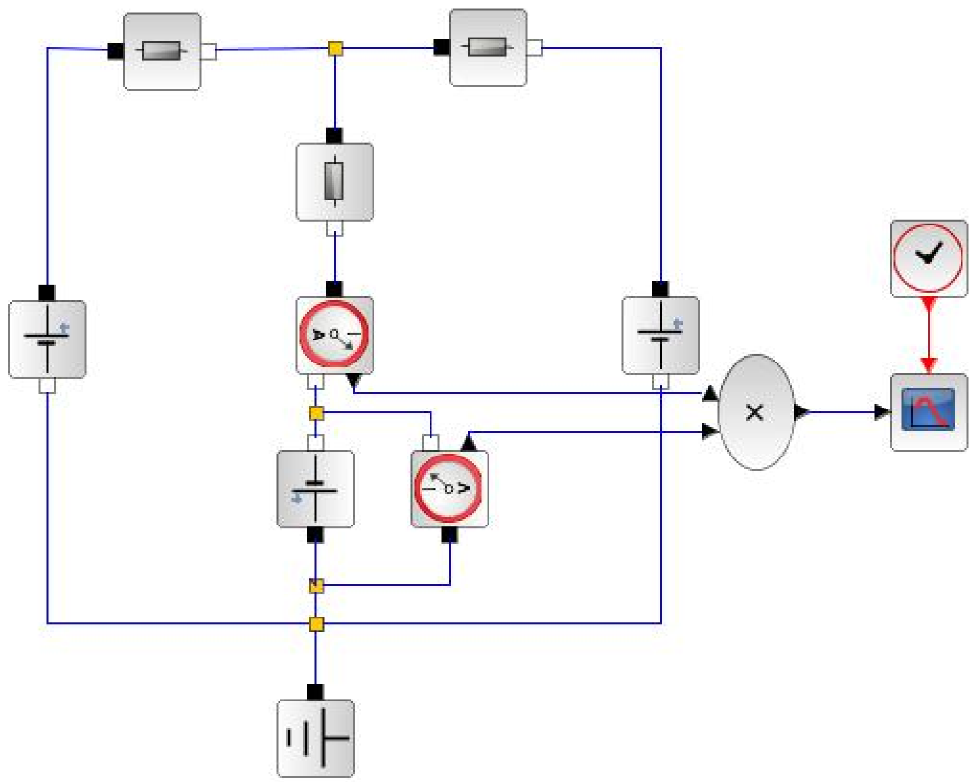

Figure 4.10: Mesh analysis

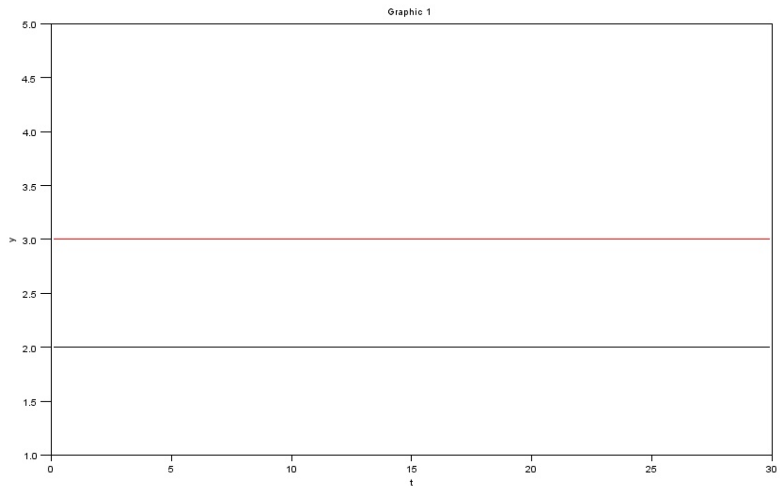

Figure 4.11: Mesh analysis

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

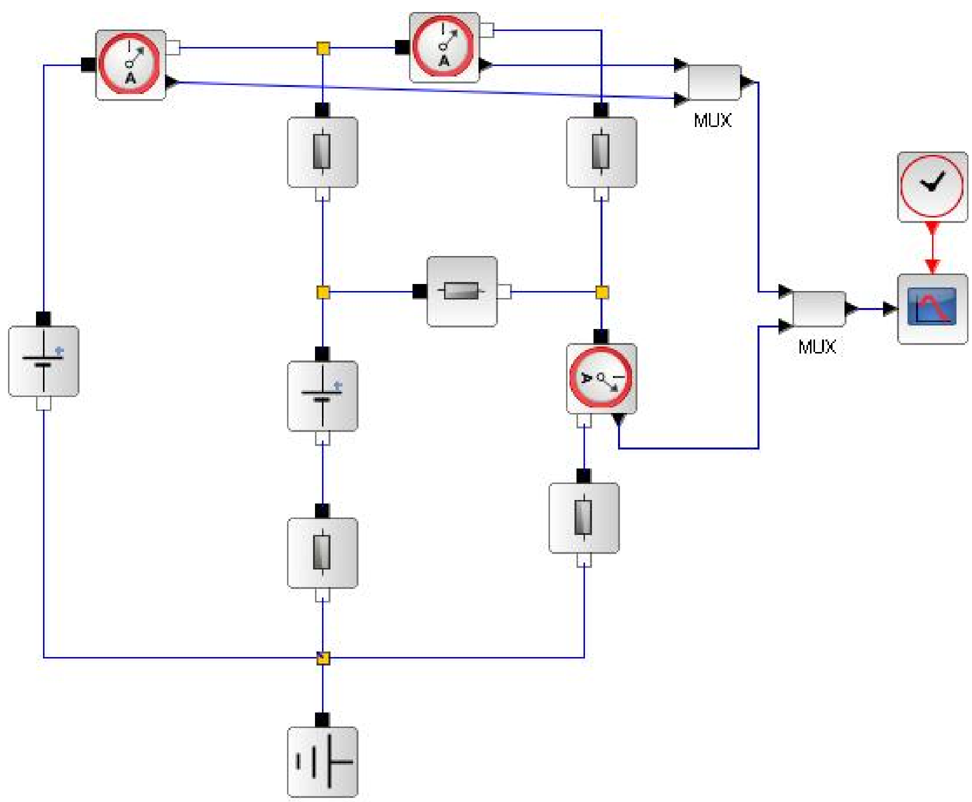

Figure 4.12: Mesh analysis

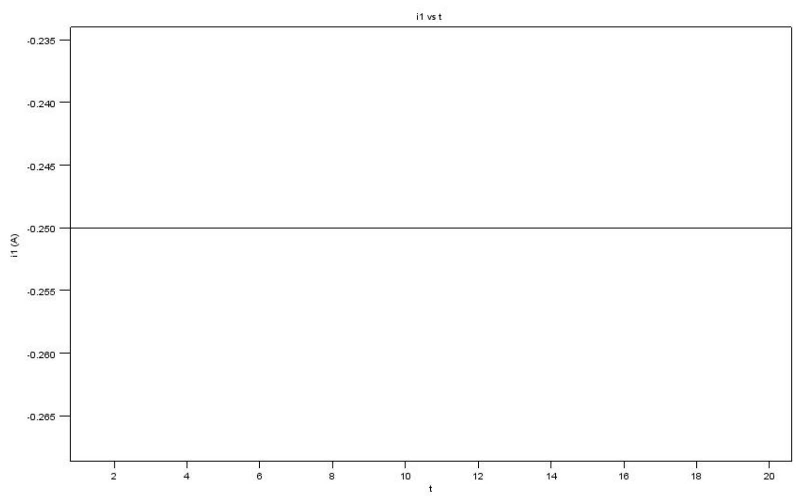

Figure 4.13: Mesh analysis

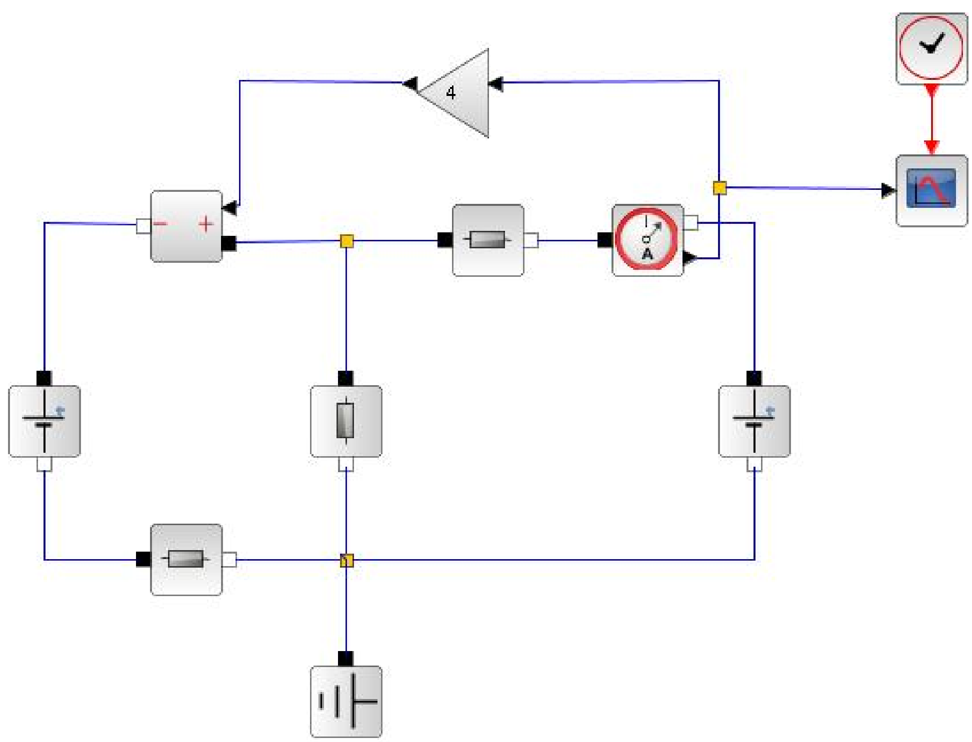

Figure 4.14: Mesh analysis

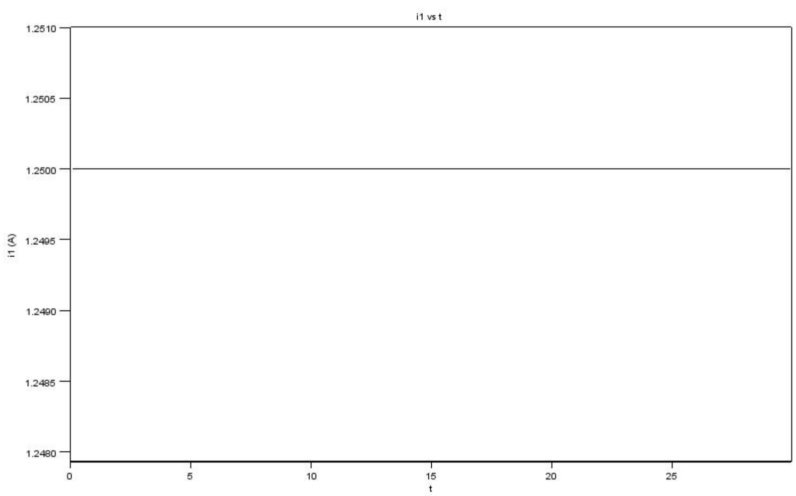

Figure 4.15: Mesh analysis

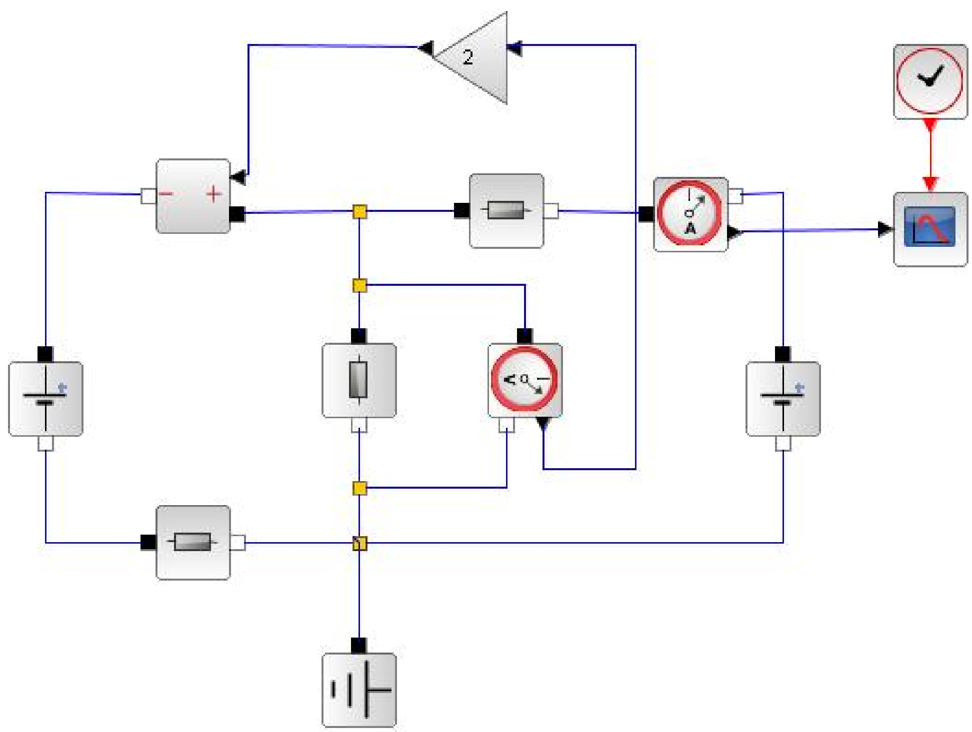

Figure 4.16: Mesh analysis

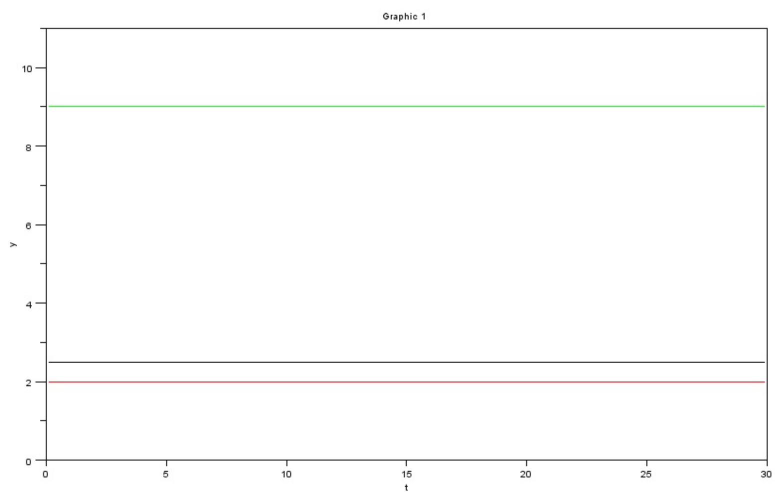

Figure 4.17: The Supermesh

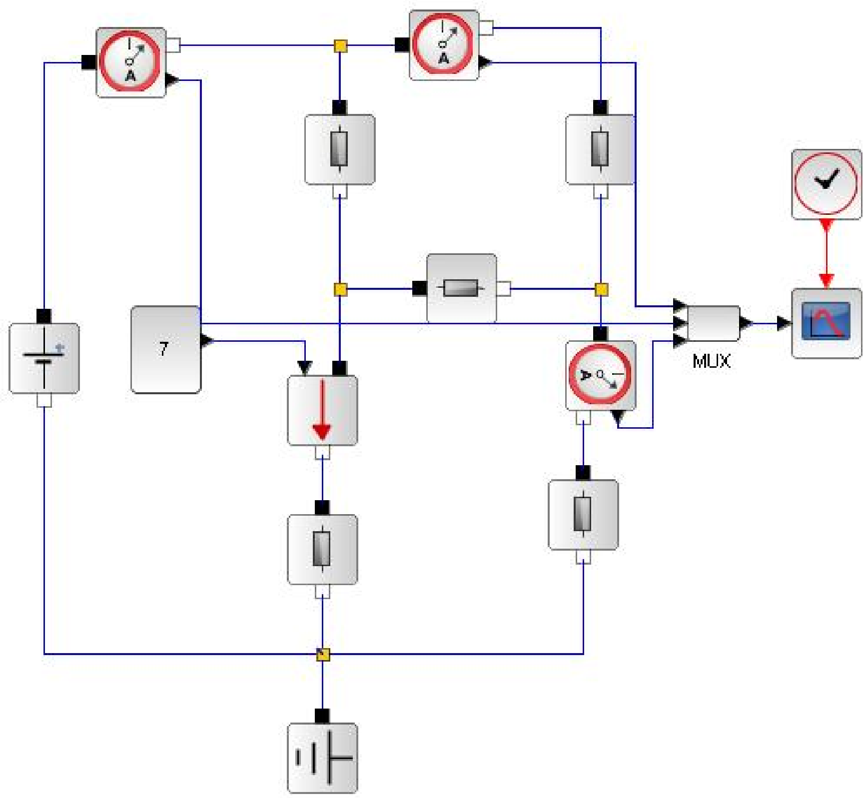

Figure 4.18: The Supermesh

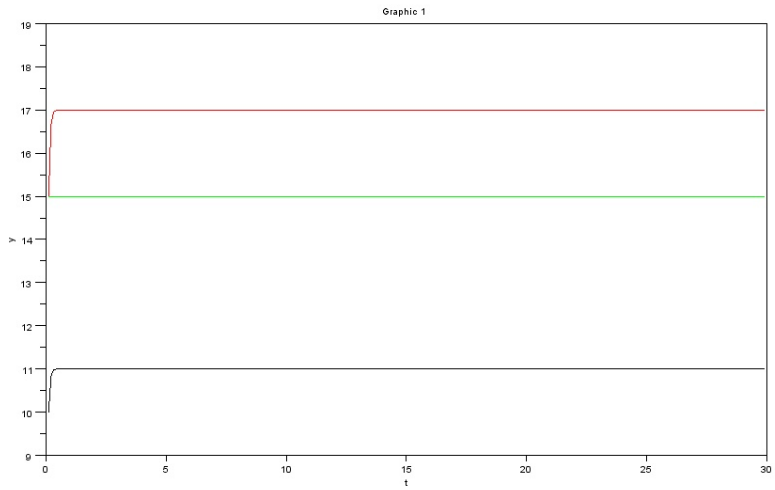

Figure 4.19: The Supermesh

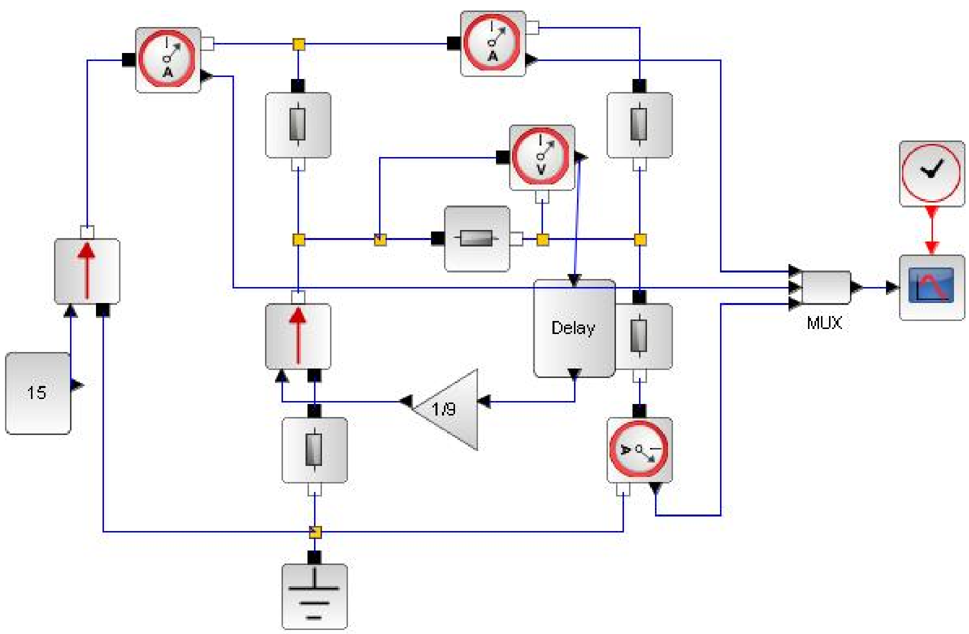

Figure 4.20: The Supermesh

Chapter 6

Network Theorems and useful Circuit Analysis Techniques

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

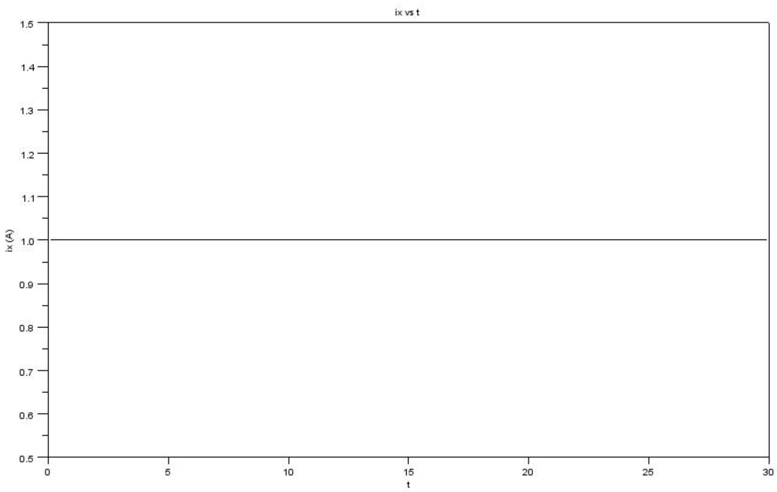

Figure 6.1: The Superposition principle

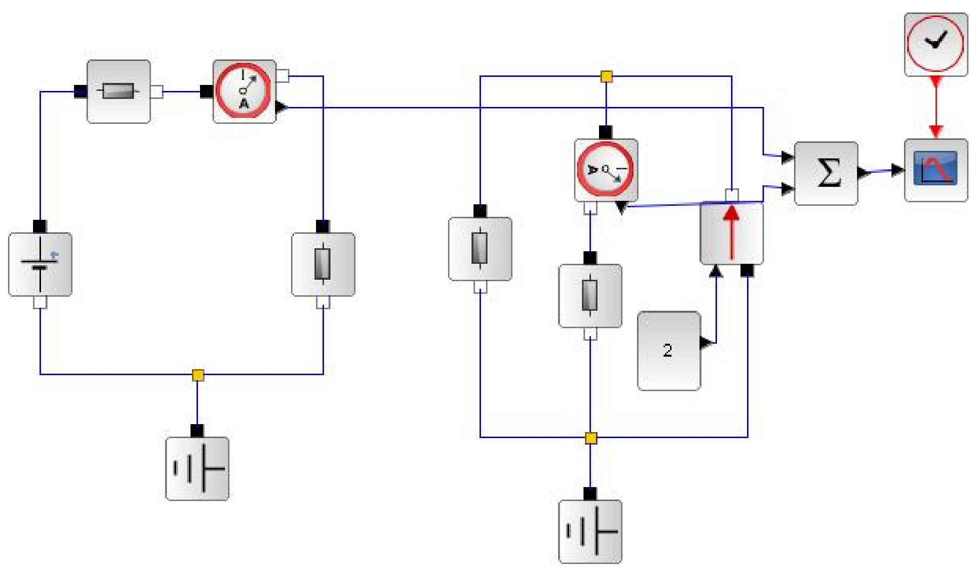

Figure 6.2: The Superposition principle

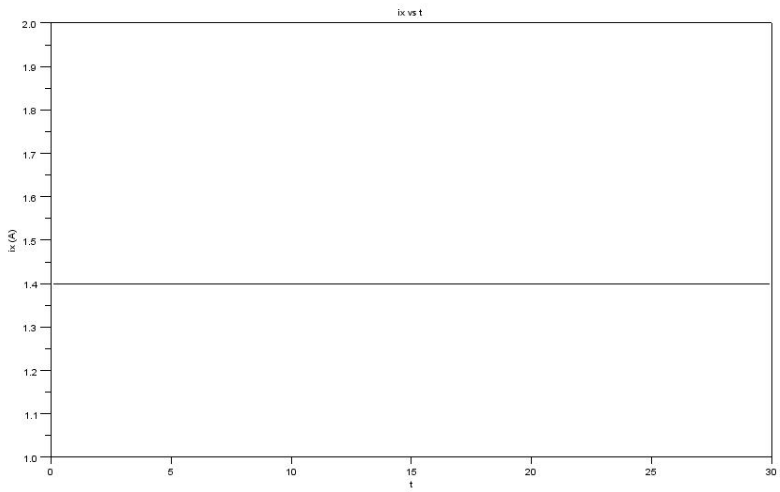

Figure 6.3: The Superposition principle

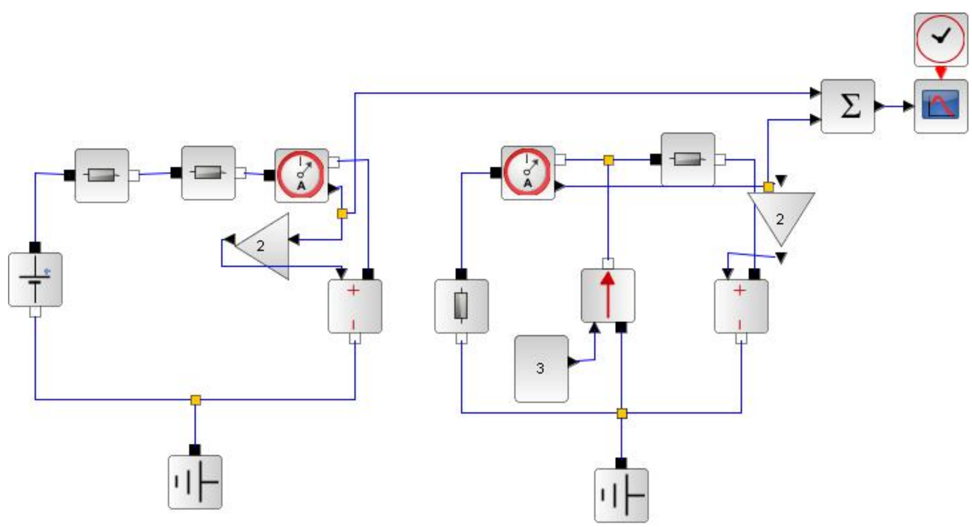

Figure 6.4: The Superposition principle

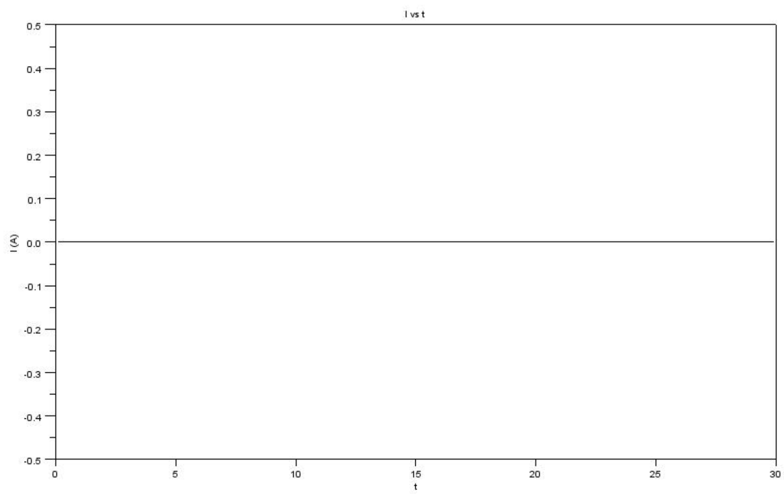

Figure 6.5: The Superposition principle

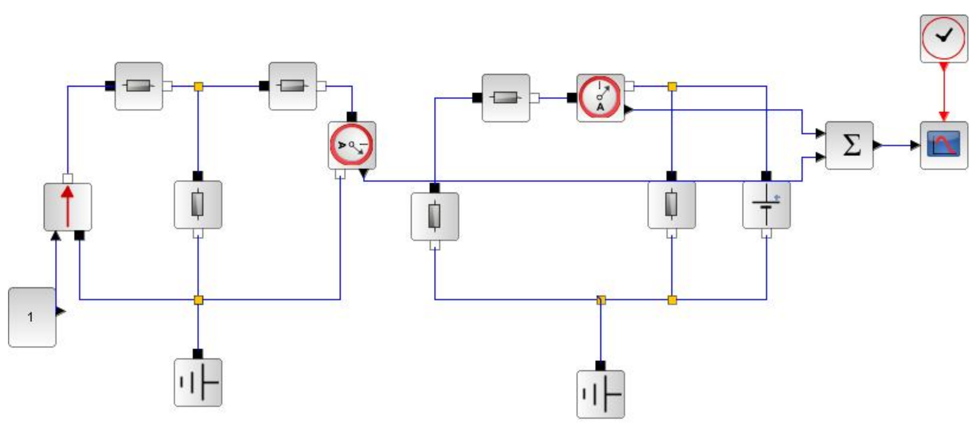

Figure 6.6: The Superposition principle

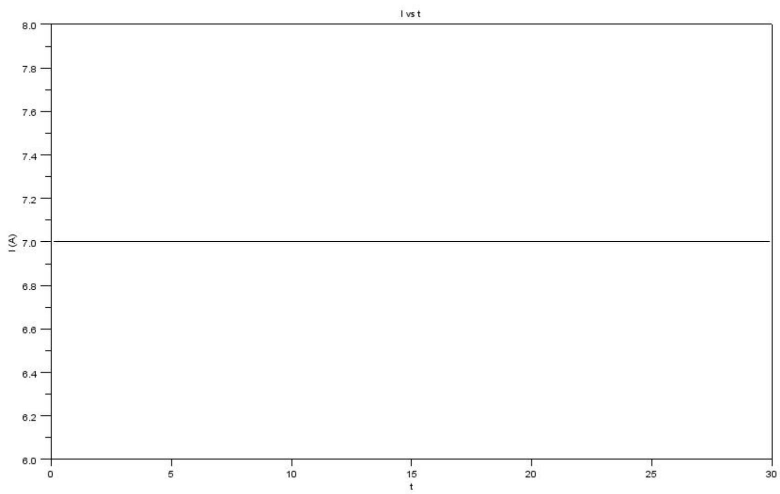

Figure 6.7: The Superposition principle

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

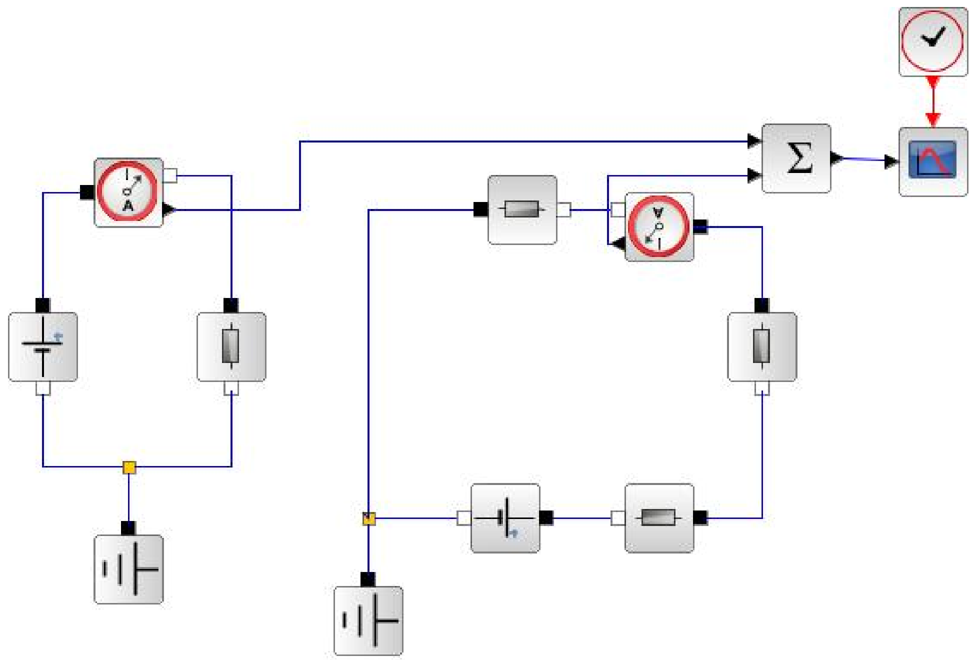

Figure 6.8: The Superposition principle

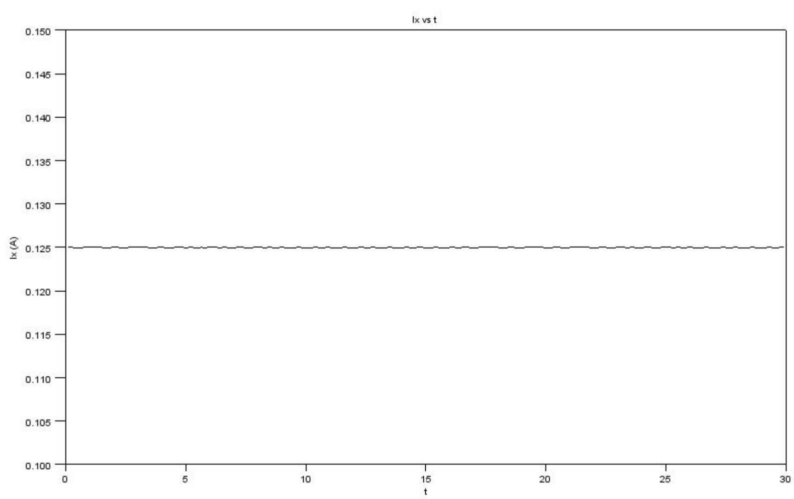

Figure 6.9: The Superposition principle

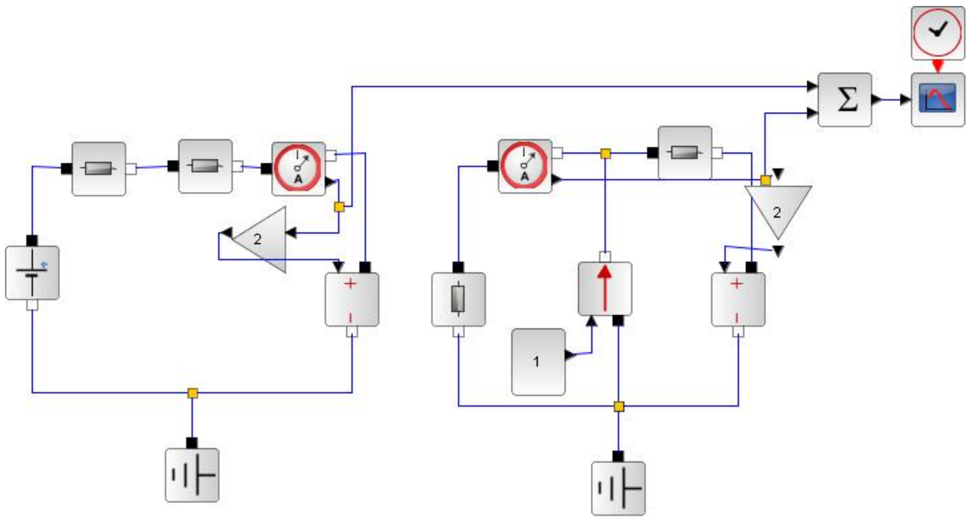

Figure 6.10: The Superposition principle

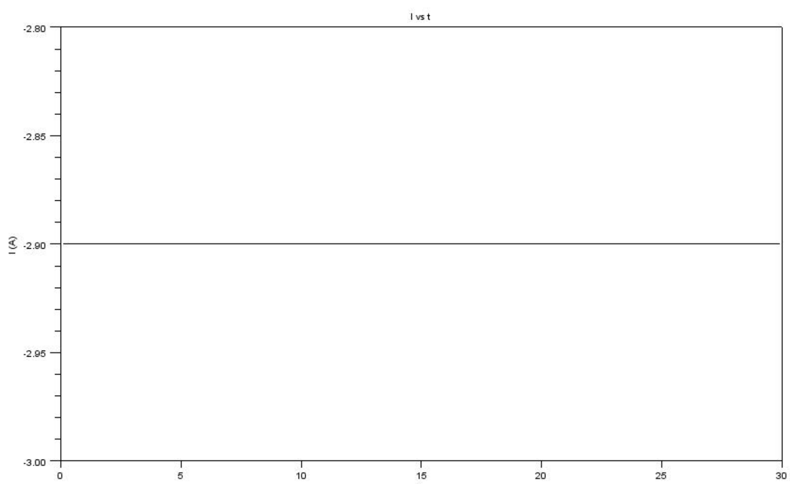

Figure 6.11: The Superposition principle

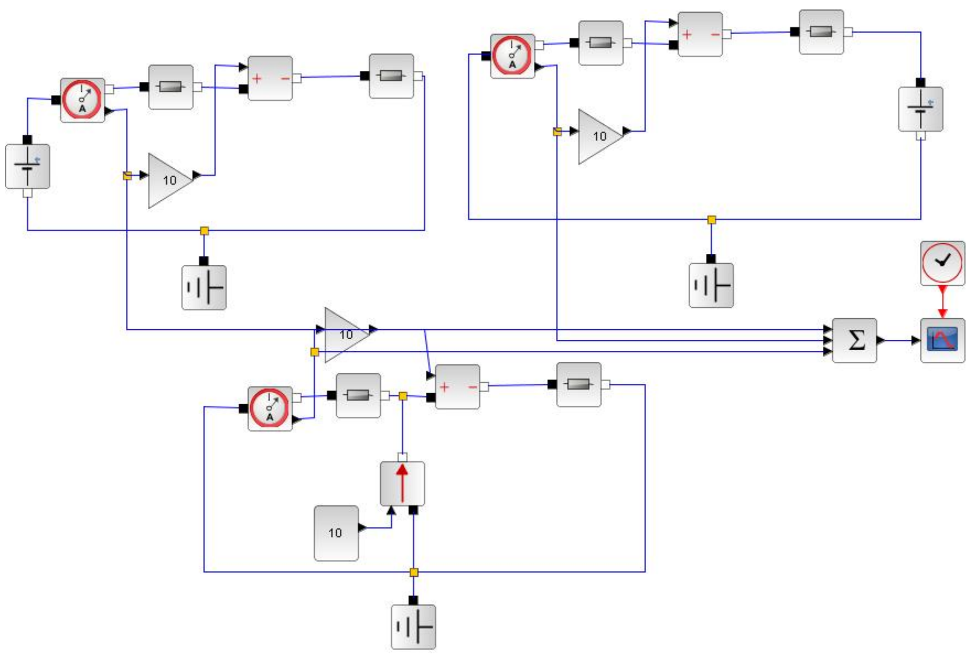

Figure 6.12: The Superposition principle

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 6.10 The Superposition principle

```
clc
//Example 6.10
//Calculate the voltage across 20 ohm capacitor
//Consider the circuit to be solved by superposition principle
disp('Consider the current source 2(90 deg)only')
//From figure 6.32
//Let I1 be the current through -i*4 capacitive reactance
Imag=2;Iph=90;
i=%i
x=Imag * cos (( Iph * %pi ) /180) ;
y=Imag * sin (( Iph * %pi ) /180) ;
I= complex (x,y)
I1=(I*(i*15))/(i*5+i*15-i*4)
//Let V20 be the voltage across -i*4 capacitive reactance
V200=(-i*4)*I1
printf("V20=%3.2fV \n",V200)
disp('Consider the 20 V voltage source only')
V=20;
//From figure 6.35
//let V201 be the voltage across -i*5 capacitive reactance
V201=-V
printf("V201=%d V \n",V201)
disp('Consider the current source 1(90 deg)only')
I1mag=1;I1ang=90;
//From figure 6.37
```

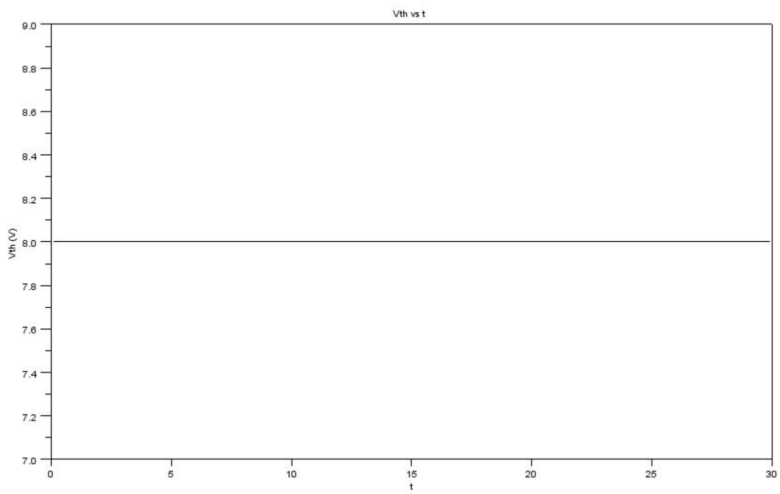

Figure 6.13: Thevenin and Norton Equivalent circuit

```
//Let V202 be the voltage across -i*5 capacitive
    reactance
V202=(-i*5)*I1mag*i
printf("V202=%3.2fV \n",V202)
//Let V20 be the voltage across -i*20 capacitive
    reactance
V20=V200+V201+V202
printf("\n V20=%3.2fV \n",V20)
```

This code can be downloaded from the website wwww.scilab.in

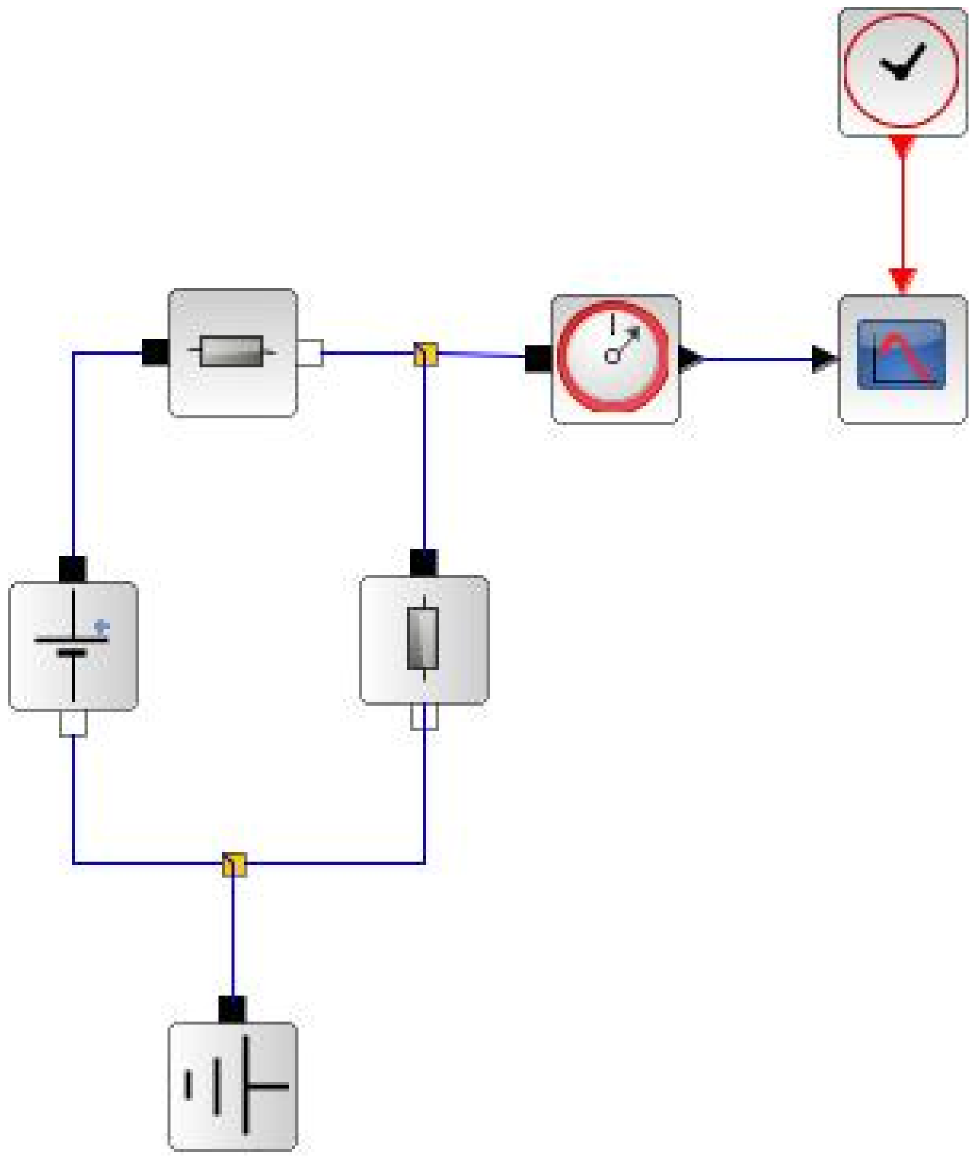

Figure 6.14: Thevenin and Norton Equivalent circuit

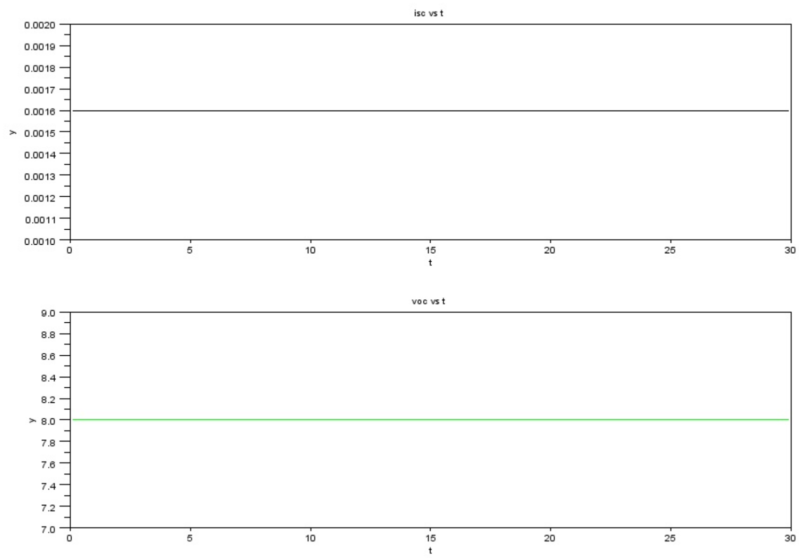

Figure 6.15: Thevenin and Norton Equivalent circuit

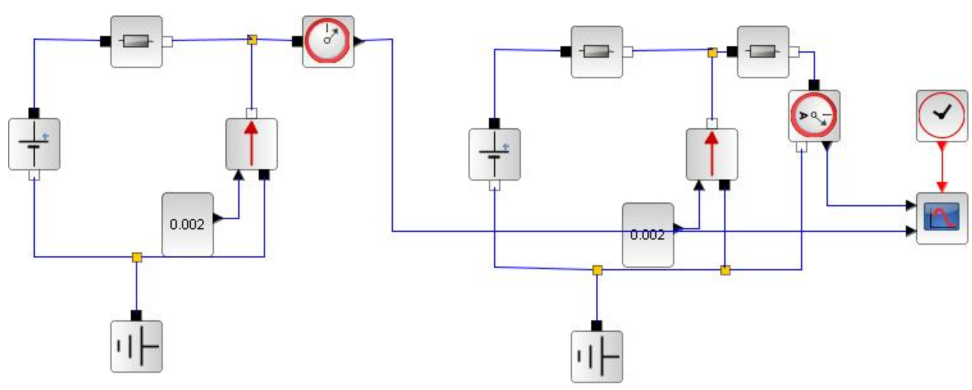

Figure 6.16: Thevenin and Norton Equivalent circuit

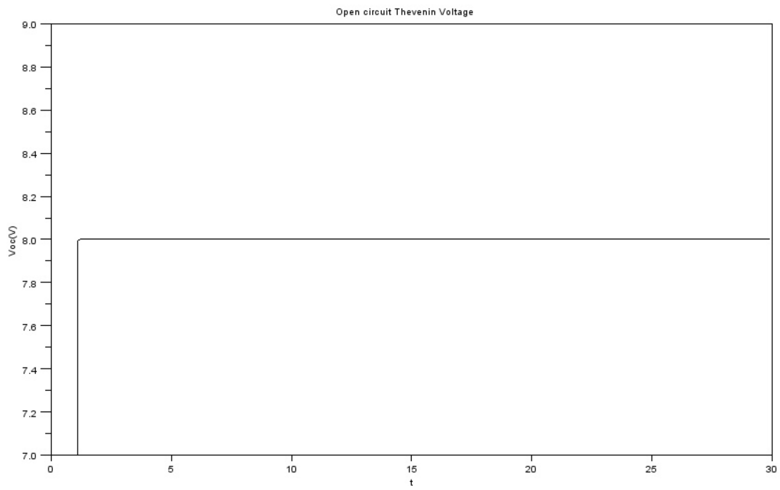

Figure 6.17: Thevenin and Norton Equivalent circuit

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

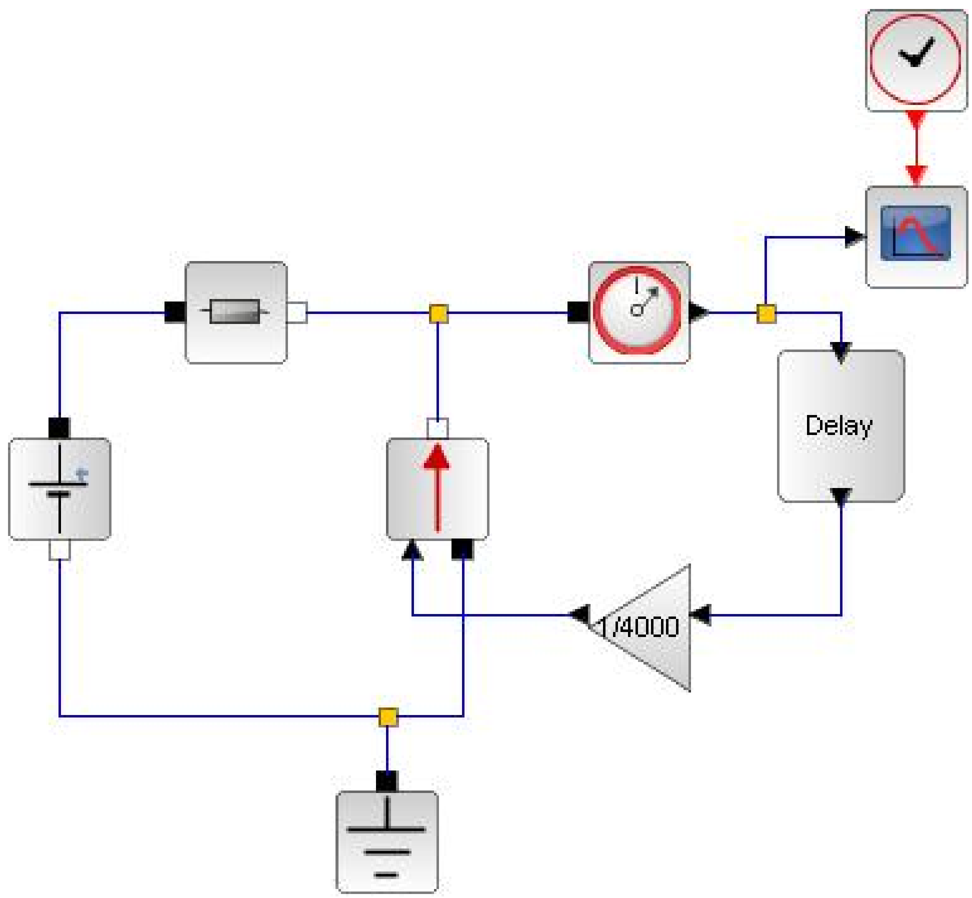

Figure 6.18: Thevenin and Norton Equivalent circuit

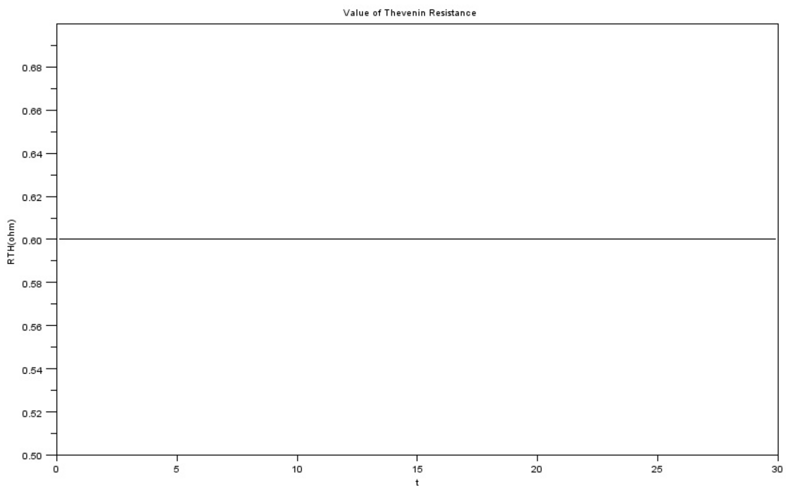

Figure 6.19: Thevenin and Norton Equivalent circuit

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 6.17 `Reciprocity Theorem`

```
clc
//Example 6.17
//Verification of Reciprocity theorem
I=20
//From figure 6.59
disp('The current divides between the two parallel impedances')
//Let I2 be the current through i5 ohm
I2=(20*%i*(10+%i*5))/(10+%i*5+%i*5-%i*2)
```

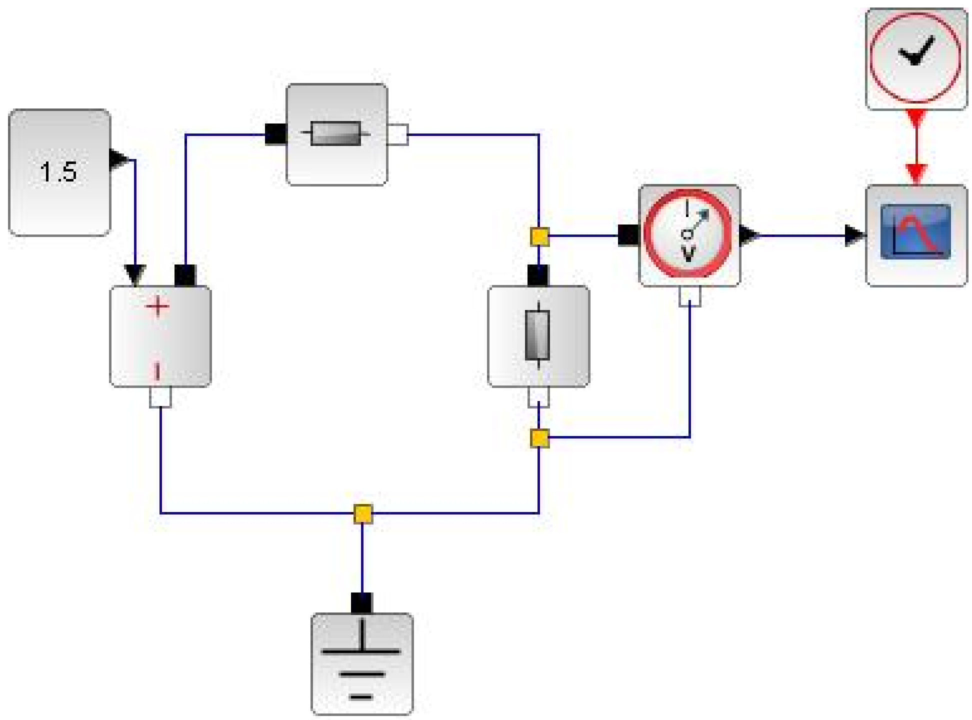

Figure 6.20: Thevenin and Norton Equivalent circuit

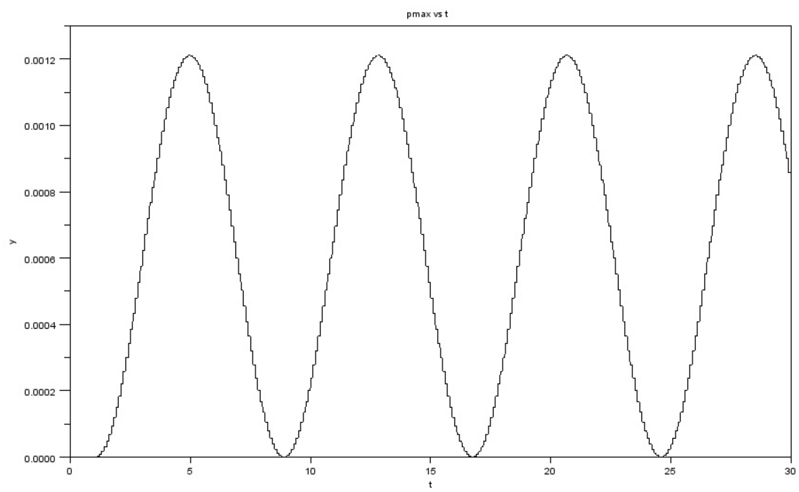

Figure 6.21: Maximum power transfer

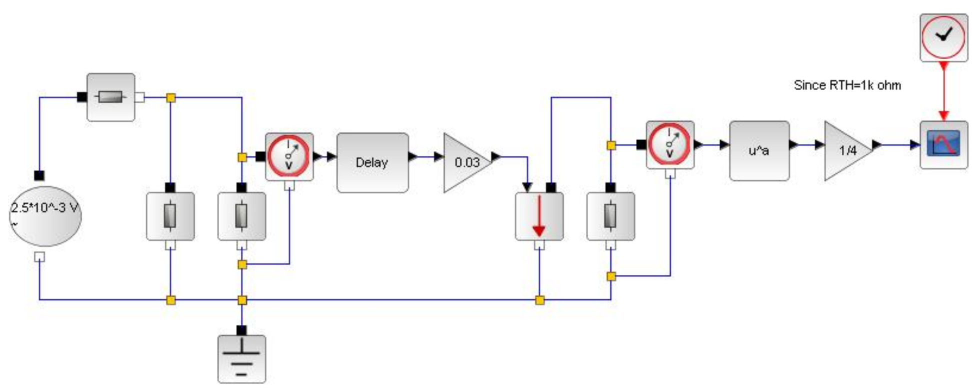

Figure 6.22: Maximum power transfer

```
//Let Vx be the voltage across −i2 ohm capacitive
   reactance
Vx=I2*(-%i*2)
[Vxmag Vxang]=polar(Vx)
printf("Vx=%3.2f(%3.2f deg)V \n",Vxmag,(Vxang*180)/
   %pi)
//To verify Reciprocity theorem remove the current
   source and place it parallel with −i2 ohm
   capacitive reactance
//From figure 6.60
//Let I2 be the current flowing through resistor of
   10 ohm
I2=(20*%i*(-%i*2))/(10+%i*5+%i*5-%i*2)
//let Vx1 be the deired output voltage across 10 ohm
    resistor and i5 inductive reactance
Vx1=I2*(10+%i*5)
[Vx1mag Vx1ang]=polar(Vx1)
printf("Vx1=%3.2f(%3.2f deg)V \n",Vx1mag,(Vx1ang
   *180)/%pi)
//Comparing the values of Vx and Vx1
disp('Vx=Vx1')
disp('Hence Reciprocity theorem is verified')
```

Scilab code Exa 6.18 Reciprocity Theorem

```
clc
//Example 6.18
//Verification of Reciprocity theorem
I=10
//From figure 6.61
disp('The current divides between the two parallel
   impedances')
//Let I2 be the current through 4 ohm
I2=(10*5)/(4-%i*4+5)
//Let Vx be the voltage across −i4 ohm capacitive
```

```
   reactance
Vx=I2*(-%i*4)
[Vxmag Vxang]=polar(Vx)
printf("Vx=%3.2f(%3.2f deg)V \n",Vxmag,(Vxang*180)/%pi)
//To verify Reciprocity theorem remove the current source and place it parallel with −i4 ohm capacitive reactance
//From figure 6.62
//Let I1 be the current flowing through resistor of 5 ohm
I1=(10*(-%i*4))/(5+4-%i*4)
//let Vx1 be the deired output voltage across 5 ohm resistor
Vx1=I1*5
[Vx1mag Vx1ang]=polar(Vx1)
printf("Vx1=%3.2f(%3.2f deg)V \n",Vx1mag,(Vx1ang*180)/%pi)
//Comparing the values of Vx and Vx1
disp('Vx=Vx1')
disp('Hence Reciprocity theorem is verified')
```

Scilab code Exa 6.19 Millman Theorem

```
clc
//Example 6.19
//Calculate total current through load
//On applying source transformation
//From figure 6.65
i=%i
V1=10;V2mag=5;V2ph=90;V3mag=14.4;V3ph=225;
x=V2mag * cos (( V2ph * %pi ) /180) ;
y=V2mag * sin (( V2ph * %pi ) /180) ;
V2= complex (x,y)
a=V3mag * cos (( V3ph * %pi ) /180) ;
```

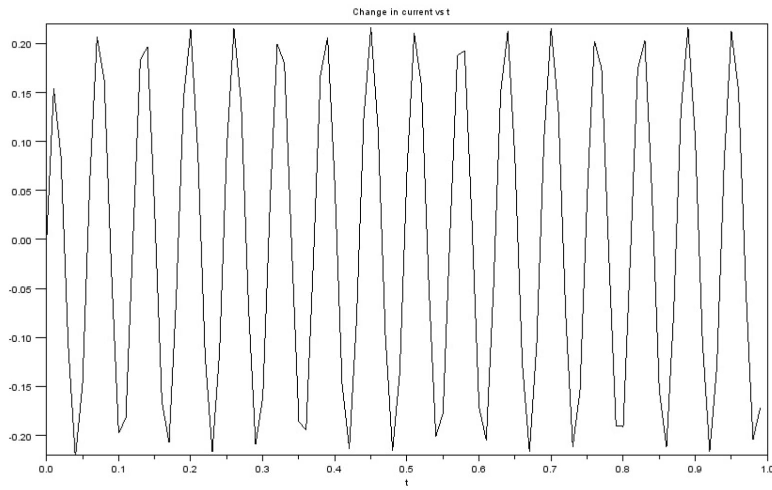

Figure 6.23: Compensation Theorem

```
b=V3mag * sin (( V3ph * %pi ) /180) ;
V3= complex (a,b)
G1=1/2;G2=1/(2+i*3);G3=1/(2-i*2);
//By applying Millman Theorem
disp('V=((V1*G1)+(V2*G2)+(V3*G3))/(G1+G2+G3)')
V=((V1*G1)+(V2*G2)+(V3*G3))/(G1+G2+G3)
[Vmag Vang]=polar(V)
R=1/(G1+G2+G3)
printf("V=%3.2f(%3.2f deg)V",Vmag,(Vang*180)/%pi)
disp(R,'R=')
//Consider the resultant circuit from figure 6.66
disp('Let the total current through 3+i4 be I')
//Applying KVL to the circuit
I=V/(3+i*4+R)
[Imag Iang]=polar(I)
printf("I=%3.2f(%3.2f deg)V",Imag,(Iang*180)/%pi)
```

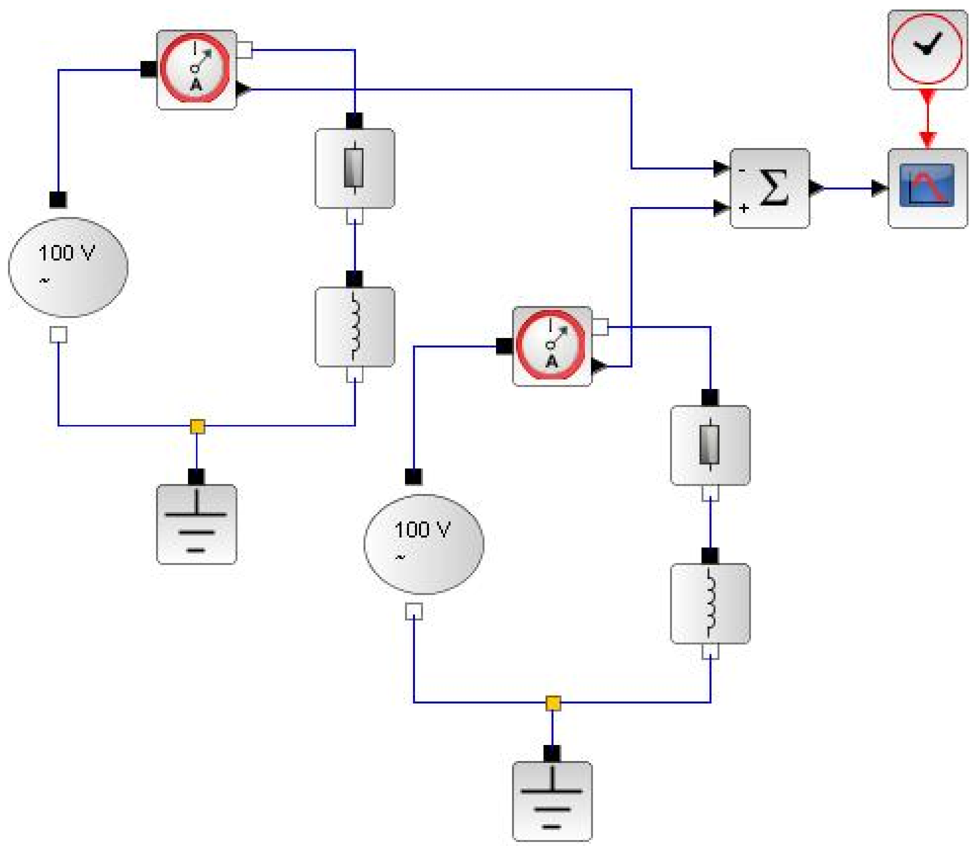

Figure 6.24: Compensation Theorem

This code can be downloaded from the website wwww.scilab.in

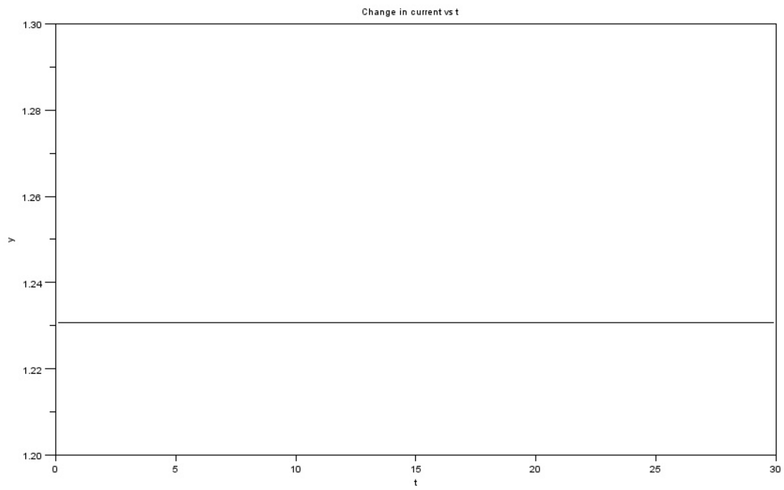

Figure 6.25: Compensation Theorem

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 6.22 `Tellegen Theorem`

```
clc
//Example 6.22
//Writing KVL for the circuit
disp('10*i=30')
//On solving
i=3;R=10;V1=25;V2=5;
printf("Power absorbed by 10 ohm resistor is %d W \n
    ",i^2*R)
printf("Power delivered by 25 V source is %d W \n",
    V1*i)
printf("Power delivered by 5 V source is %d W \n",V2
```

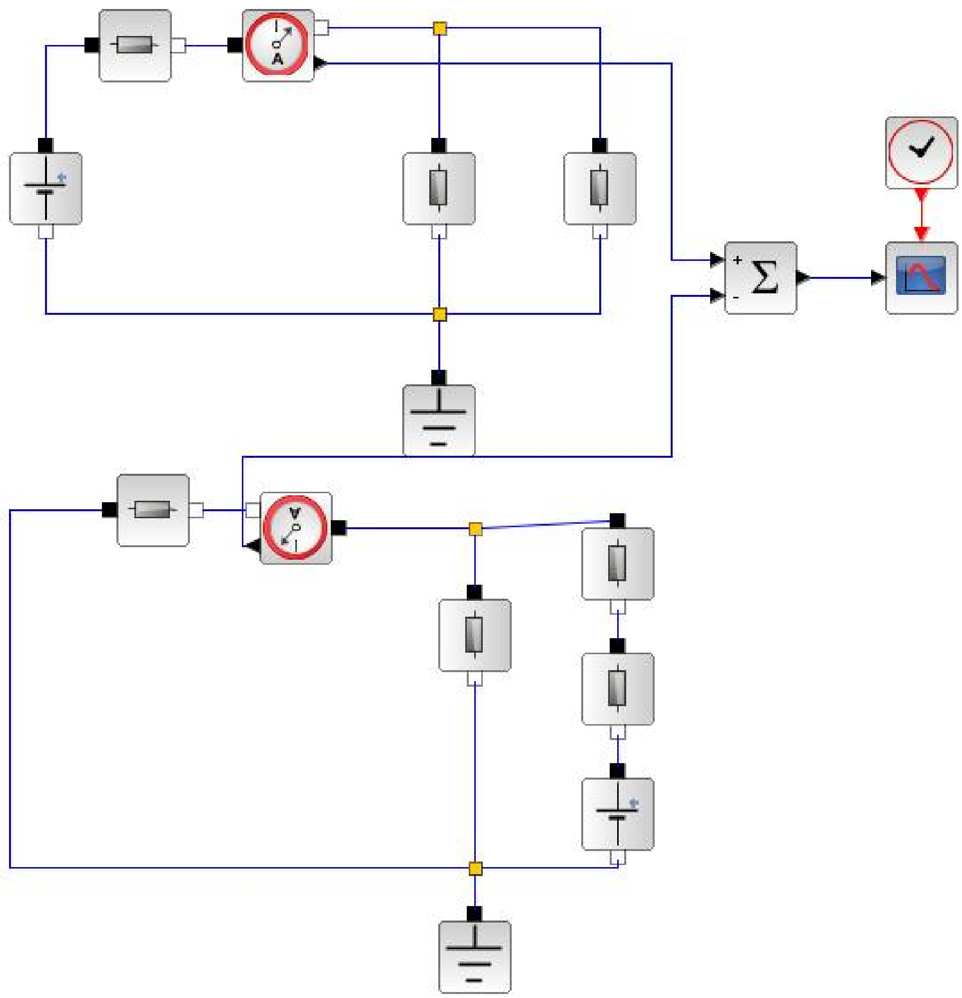

Figure 6.26: Compensation Theorem

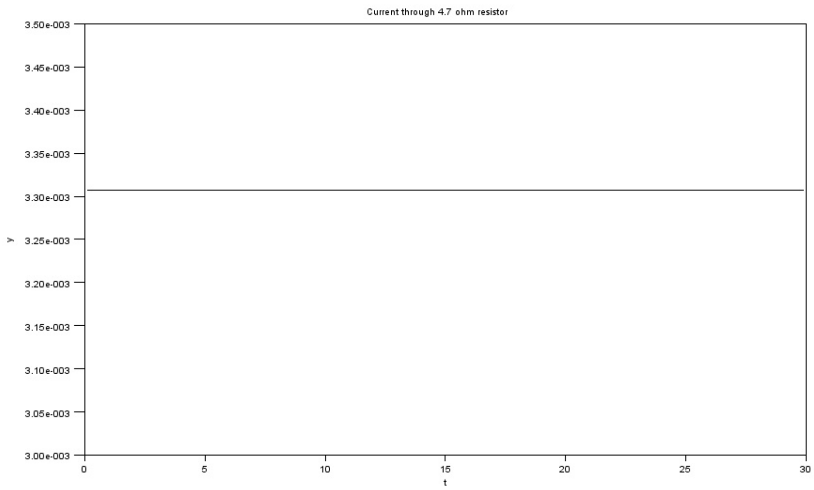

Figure 6.27: Source Transformations

```
   *i)
//Let P be the total power
P=i^2*R-(V1*i+V2*i)
if P==0 then
    disp('Tellegen theorem is valid')
else
    disp('Tellegen theorem is not valid')
end
```

This code can be downloaded from the website wwww.scilab.in

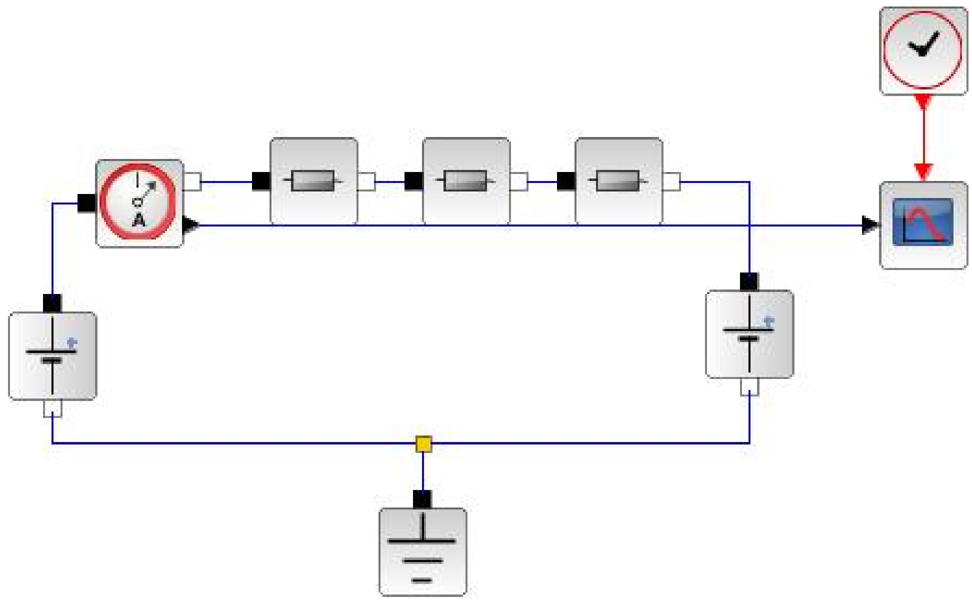

Figure 6.28: Source Transformations

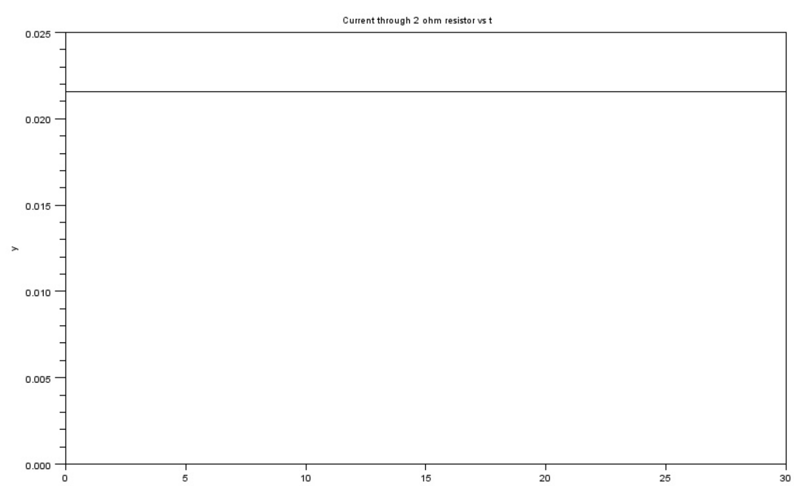

Figure 6.29: Source Transformations

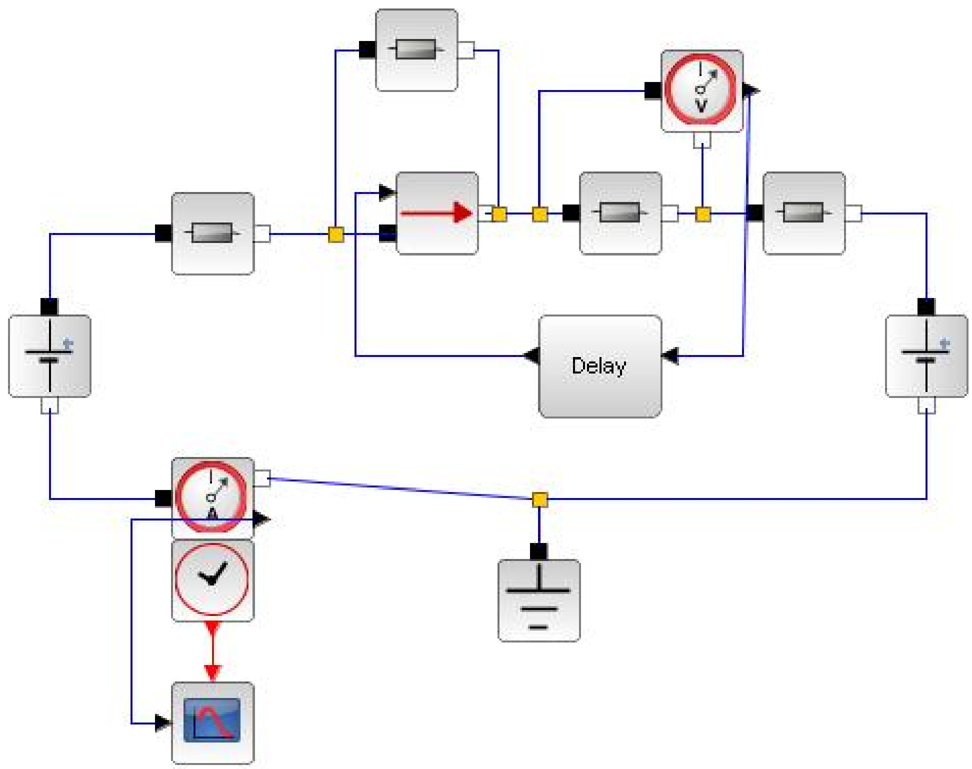

Figure 6.30: Source Transformations

This code can be downloaded from the website wwww.scilab.in

Chapter 7

Capacitors and Inductors

Scilab code Exa 7.1 The Capacitor

```
clc
syms s t
//part(a)
i=diff(5*s^0,s)
disp(i,'i=')
//prt(b)
i1=diff(4*sin(3*t),t)
t=-2:.1:5
 plot (t,12*cos(3*t))
xtitle('i vs t','t(s)','i(A)')
```

Scilab code Exa 7.2 The Capacitor

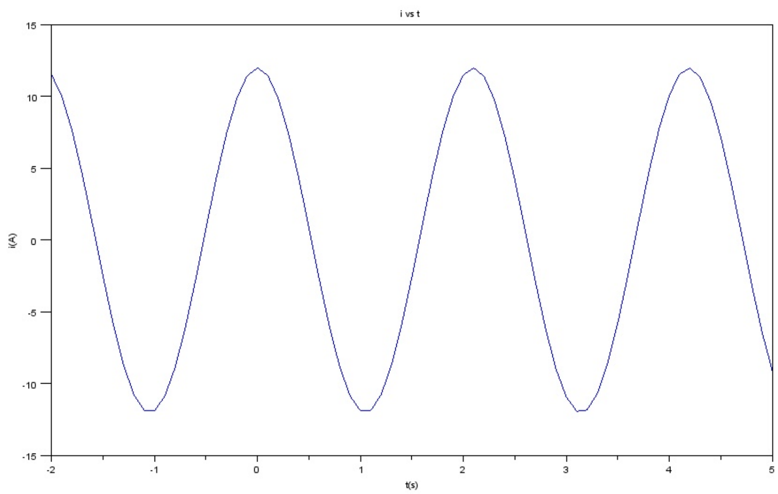

Figure 7.1: The Capacitor

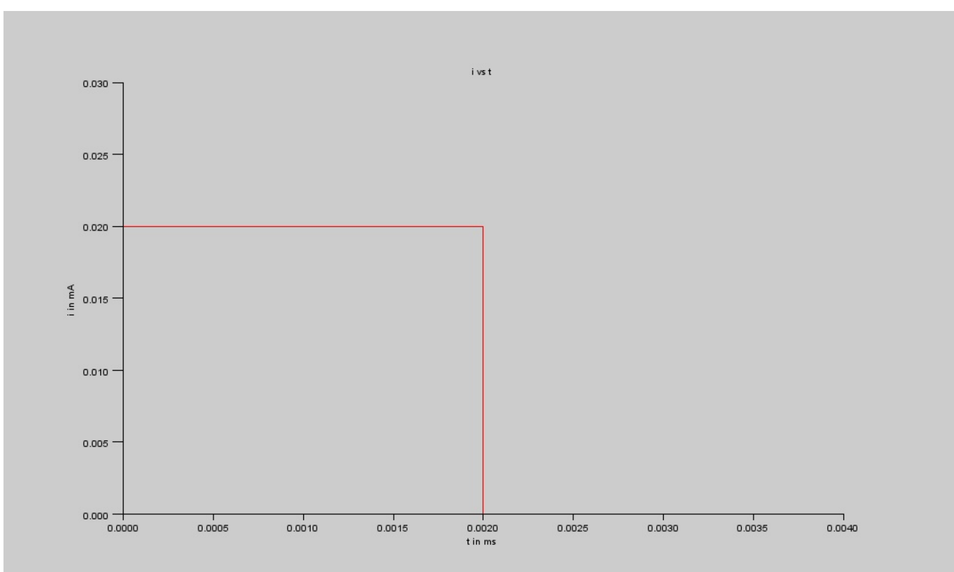

Figure 7.2: The Capacitor

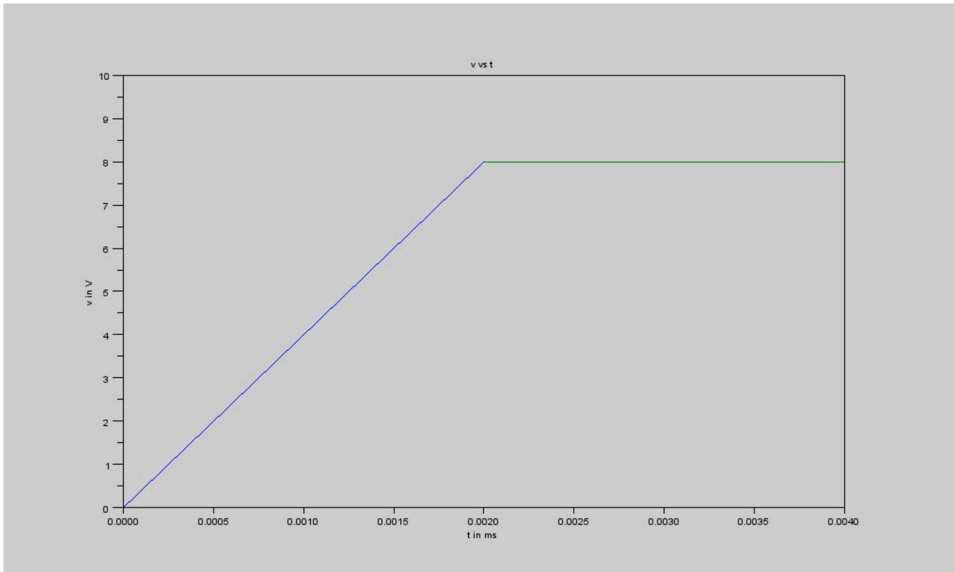

Figure 7.3: The Capacitor

```
clc
t = 0:0.000001:0.002;
deff('y=u(t)','y=1*(t>=0)');
y =0.02*(u(t) - u(t-0.002));
figure
a=gca()
subplot(111)
plot2d(t,y,5,rect=[0 0 0.004 0.03])
xtitle('i vs t','t in ms','i in mA')

syms s
//For t<=0 ms
v=0
//For the region in the rectangular pulse i.e 0<t<=2
    ms
v=integ(s^0,s)*4000
//For t>2 ms
v=8
s=0:0.000001:0.002
```

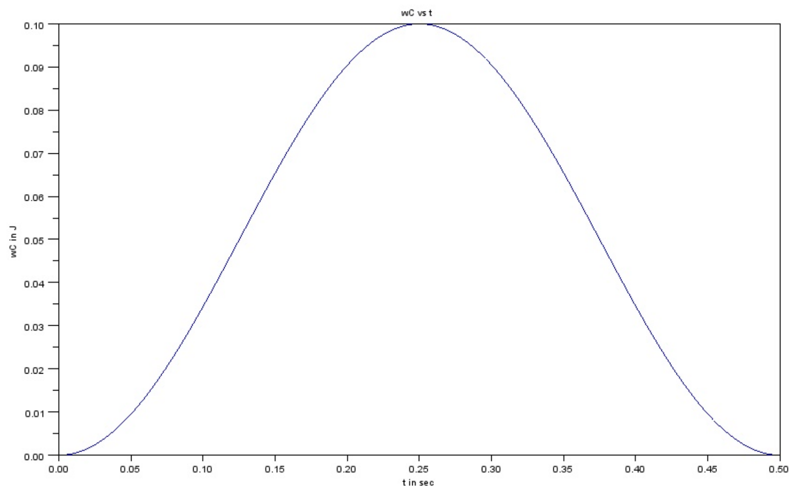

Figure 7.4: The Capacitor

```

figure
a=gca()
subplot(111)
plot(s,(4000*s),s+0.002,8)
xtitle('v vs t','t in ms','v in V')
```

Scilab code Exa 7.3 The Capacitor

```
clc
//Example 7.3
//Let wc be the energy stored in capacitor
C=20*10^-6; R=10^6;
t=0:0.001:0.5
v=100*sin(2*%pi*t)
```

```
wc=0.5*C*v^2
plot(t,wc)
xtitle('wC vs t','t in sec','wC in J')
//Let iR be the current in the resistor
iR=v/R
//Let pR be the power dissipated in the resistor
pR=iR^2*R
//If wR is the energy dissipated in the resistor
syms s
wR=integ(100*(sin(2*%pi*s))^2,s,0,0.5)
disp(wR,'wR=')
```

Scilab code Exa 7.7 The Inductor

```
clc
//Example 7.7
printf("Given")
disp('i=12*sin(%pi*t/6),R=0.1 ohm,L=3H')
t=0:.1:6
i=12*sin(%pi*t/6),R=0.1;L=3;
//Let wL be the energy stored in the inductor
wL=0.5*L*i^2
plot(t,wL)
//From the above graph
wLmax=216;tmax=3;
printf("Maximum value at %d J at %d sec",wLmax,tmax)
//Let pR be the power dissipated in the resistor
pR=i^2*R
//Energy converted to heat in 6 sec interval in the
    resistor is
syms s
wR=integ(14.4*(sin(%pi/6*s))^2,s,0,6)
disp(wR,'wR')
```

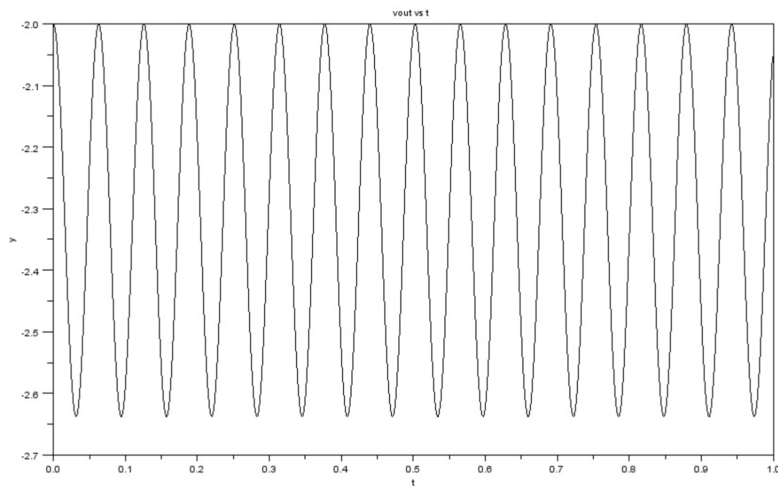

Figure 7.5: Modeling Capacitors and Inductors

This code can be downloaded from the website wwww.scilab.in

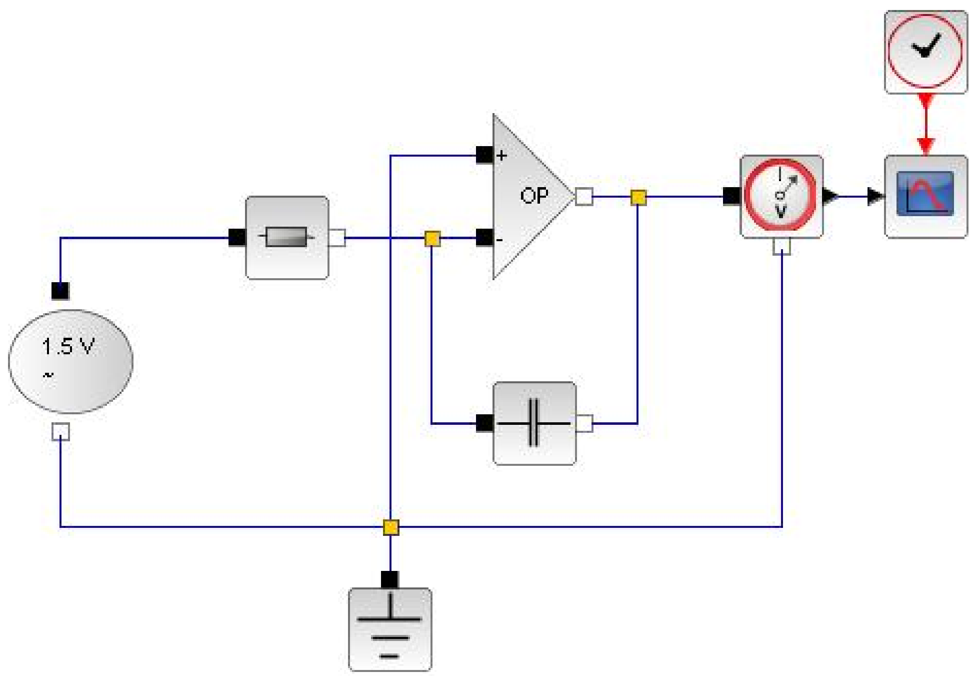

Figure 7.6: Modeling Capacitors and Inductors

Chapter 8

Basic RL and RC circuits

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

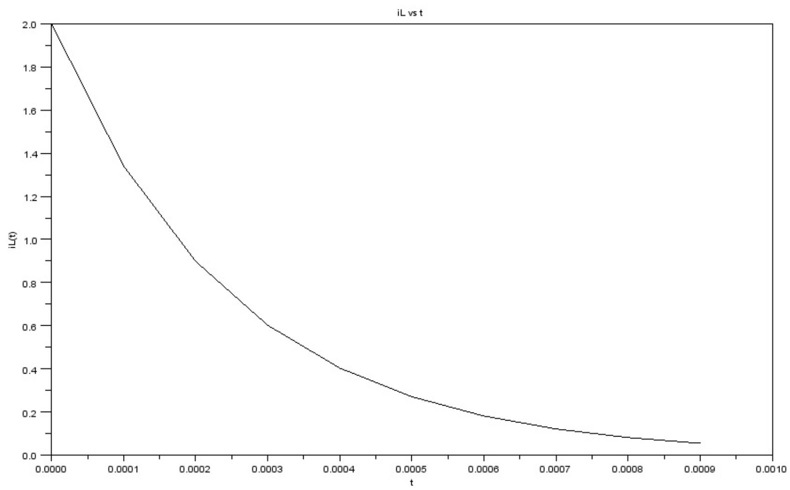

Figure 8.1: The Source free RL Circuit

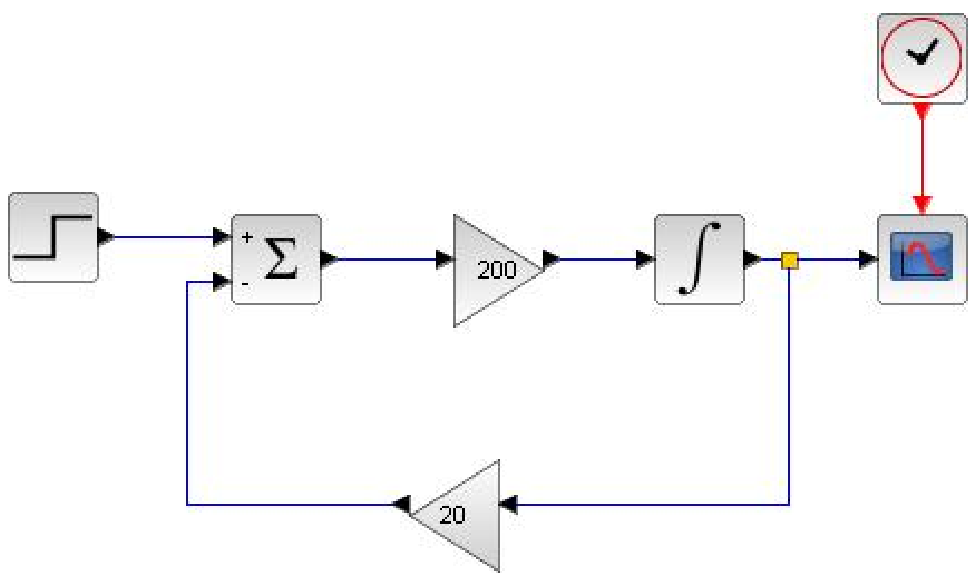

Figure 8.2: The Source free RL Circuit

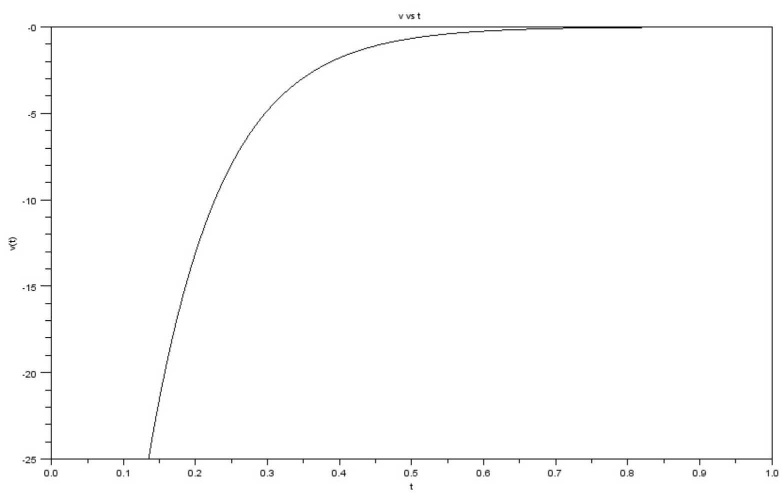

Figure 8.3: The Source free RL Circuit

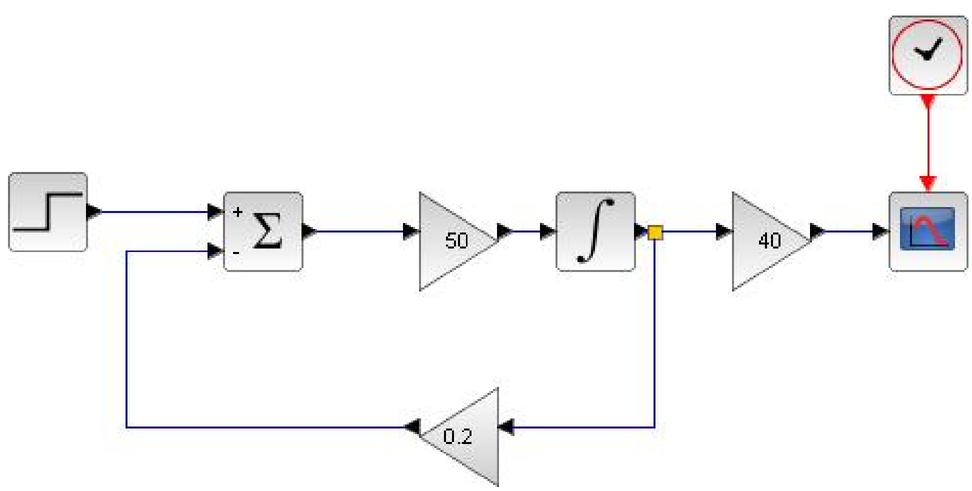

Figure 8.4: The Source free RL Circuit

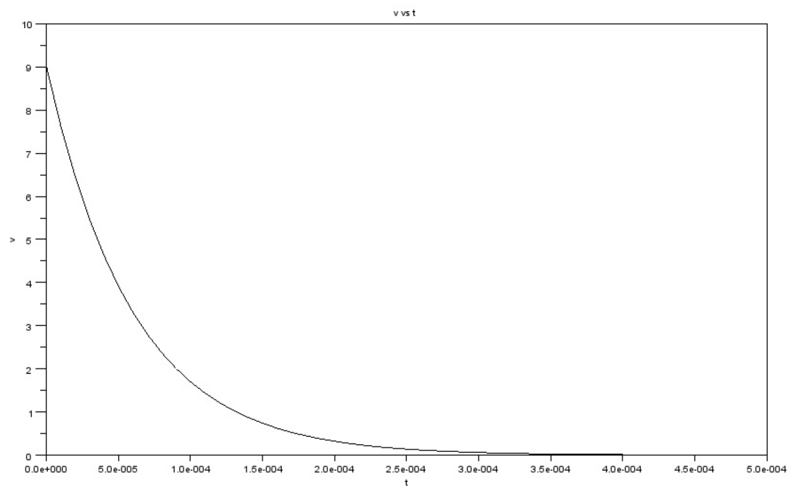

Figure 8.5: The Source free RC Circuit

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

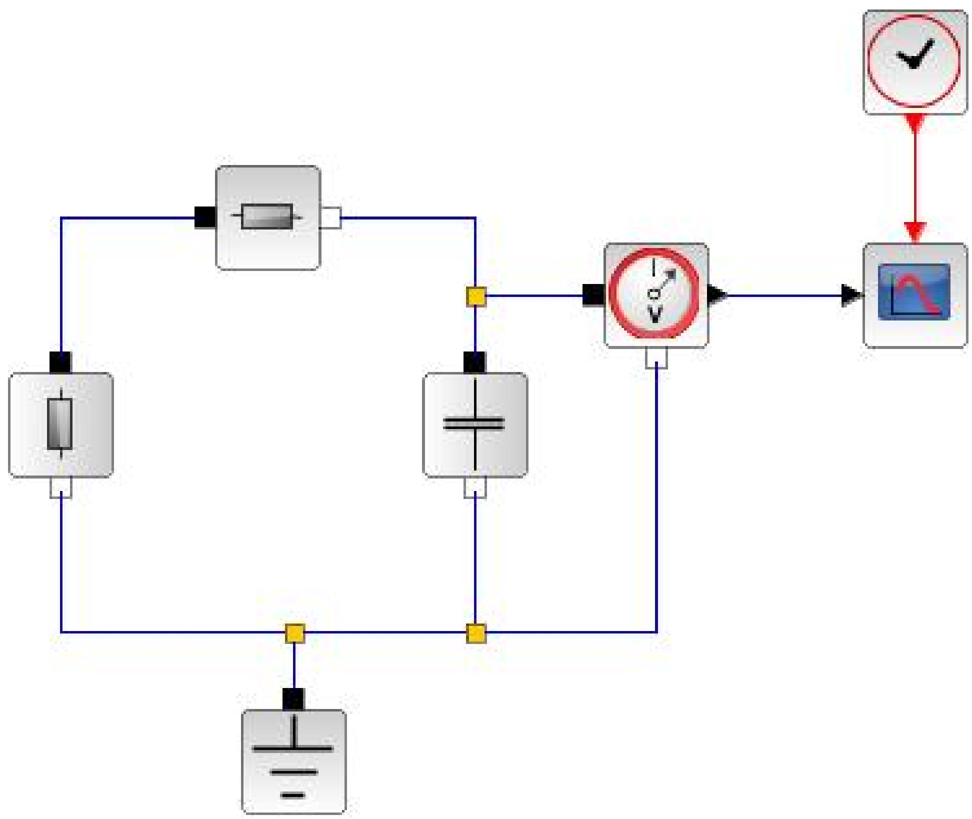

Figure 8.6: The Source free RC Circuit

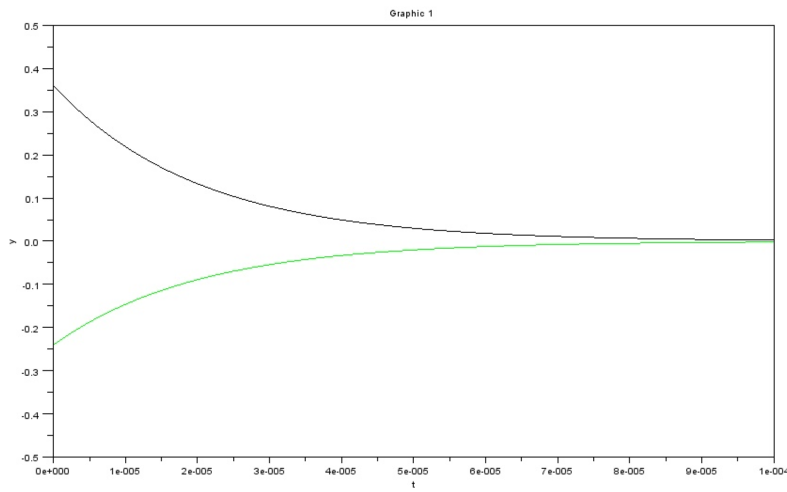

Figure 8.7: A more General Perspective

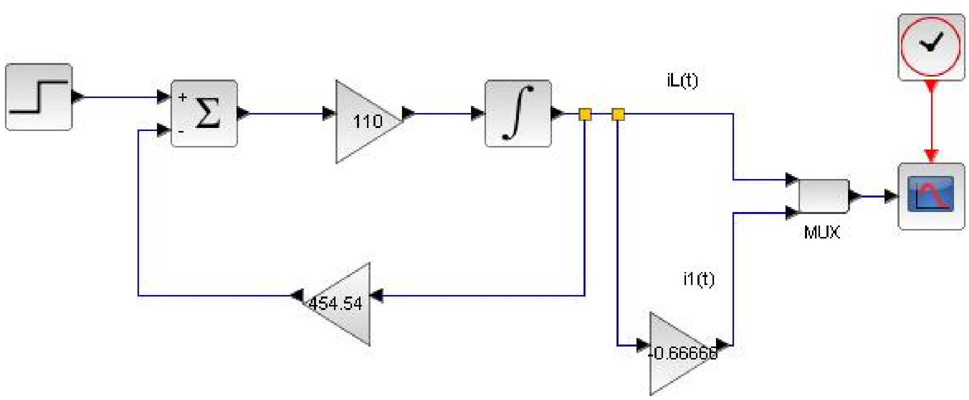

Figure 8.8: A more General Perspective

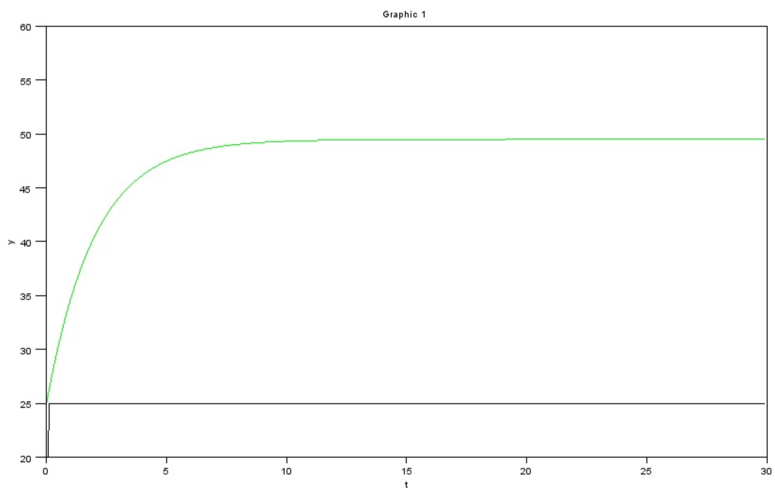

Figure 8.9: Natural and Forced Response

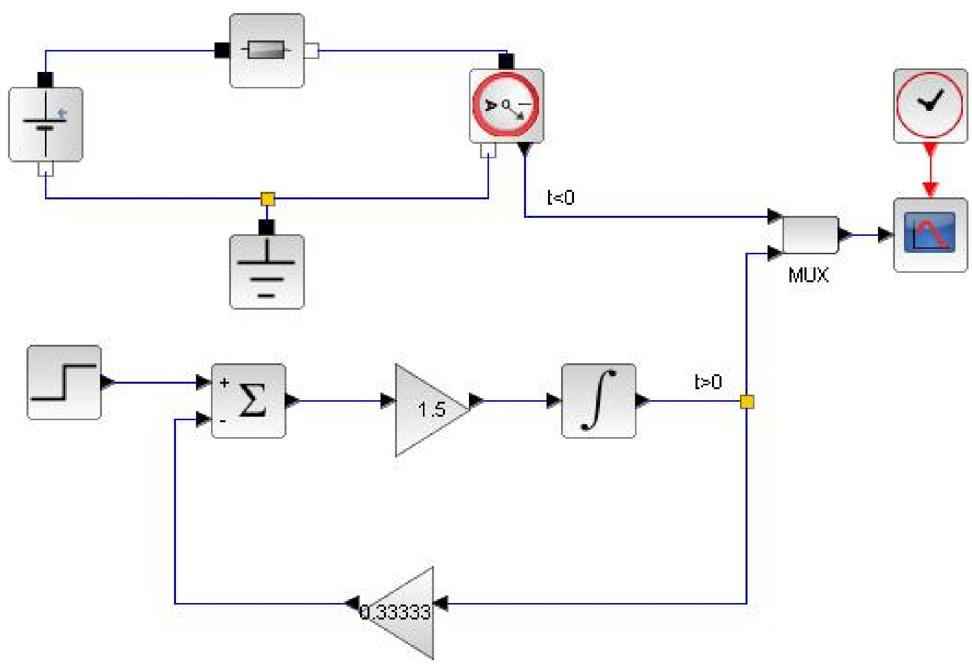

Figure 8.10: Natural and Forced Response

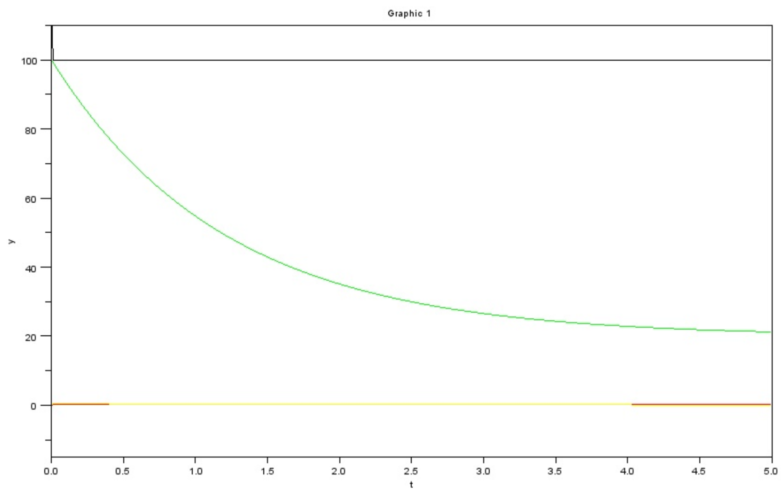

Figure 8.11: Driven RC circuits

This code can be downloaded from the website wwww.scilab.in

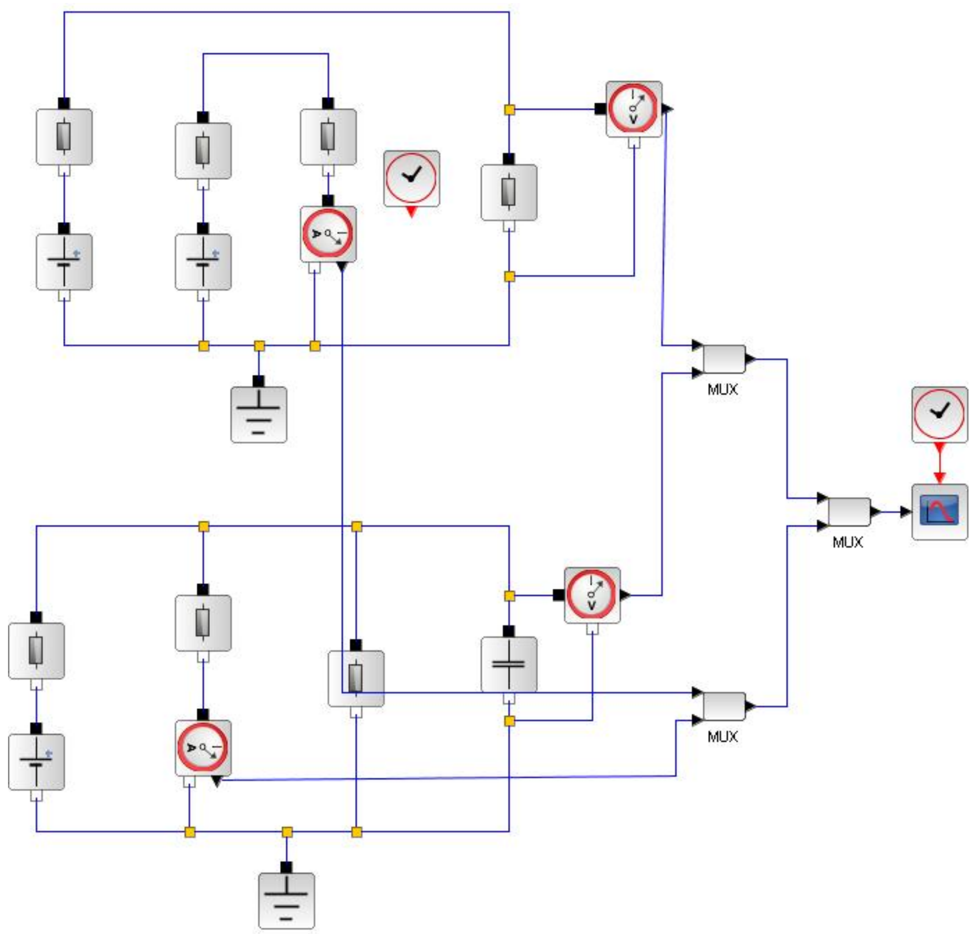

Figure 8.12: Driven RC circuits

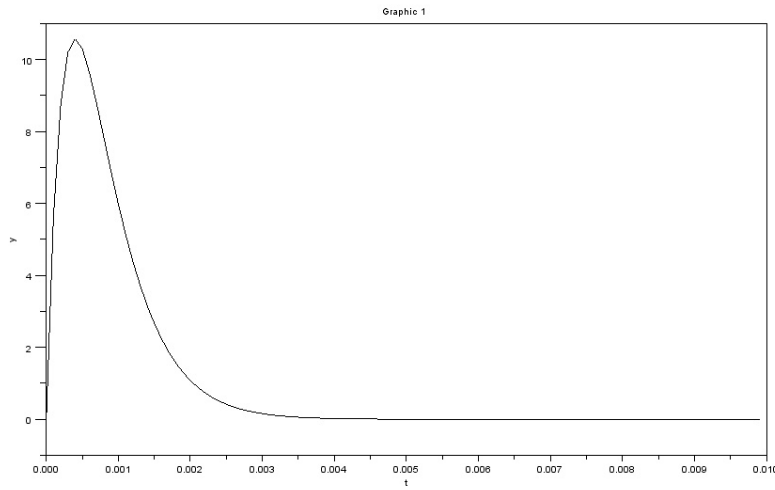

Figure 8.13: Driven RC circuits

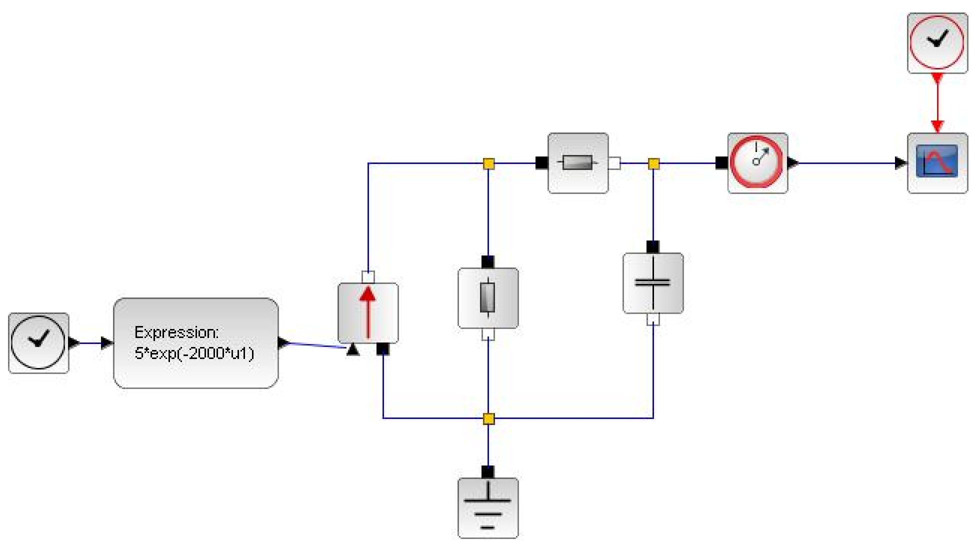

Figure 8.14: Driven RC circuits

Chapter 9

The RLC Circuit

Scilab code Exa 9.1 The Source free parallel circuit

```
//Example 9.1
//Calculate resistor values for underdamped and
   overdamped responses
printf("Given")
disp('L=10mH and C=100uF')
L=10*10^-3;C=100*10^-6
w0=sqrt(1/(L*C))
printf("w0=%drad/s\n",w0)
//alpha(a)=1/(2*R*C)
disp('For an overdamped response')
disp('a > w0')
//On solving
disp('Hence')
disp('R<5ohm')
disp('For an underdamped response')
disp('a < w0')
//On solving
disp('Hence')
disp('R>5ohm')
```

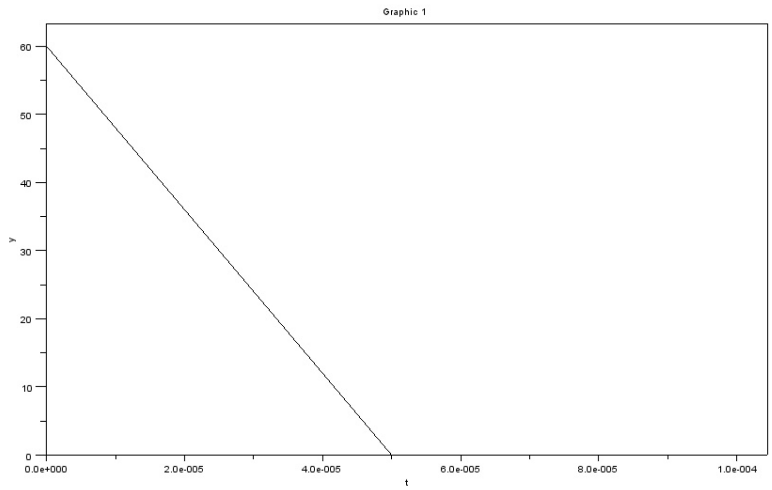

Figure 9.1: The Overdamped parallel circuit

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

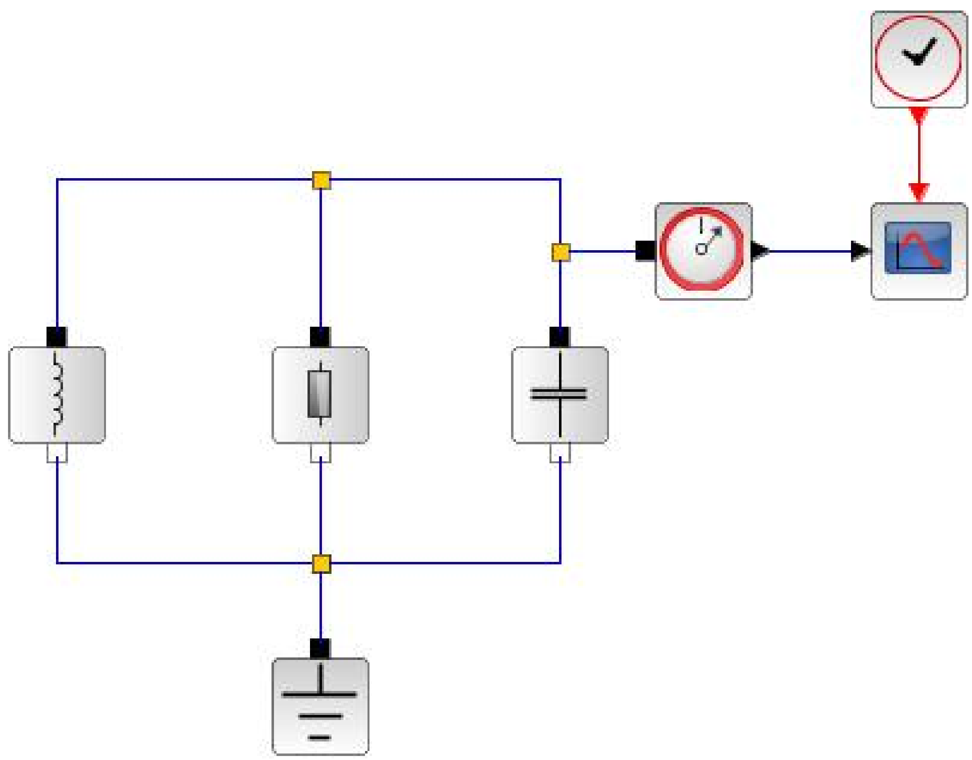

Figure 9.2: The Overdamped parallel circuit

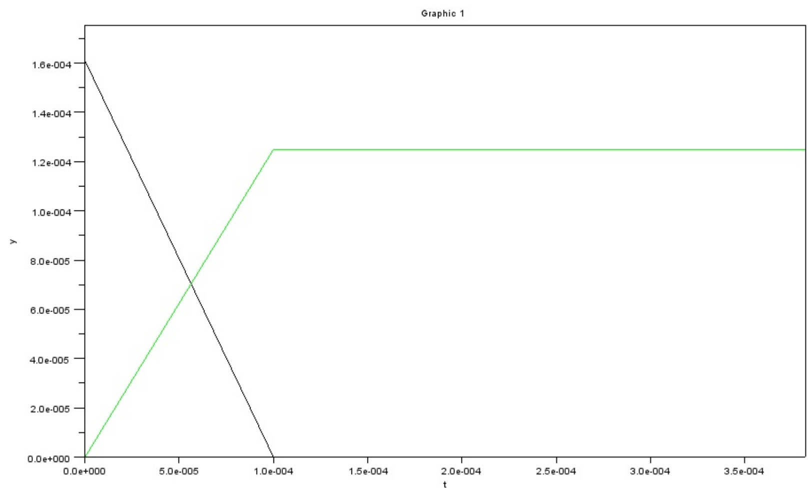

Figure 9.3: The Overdamped parallel circuit

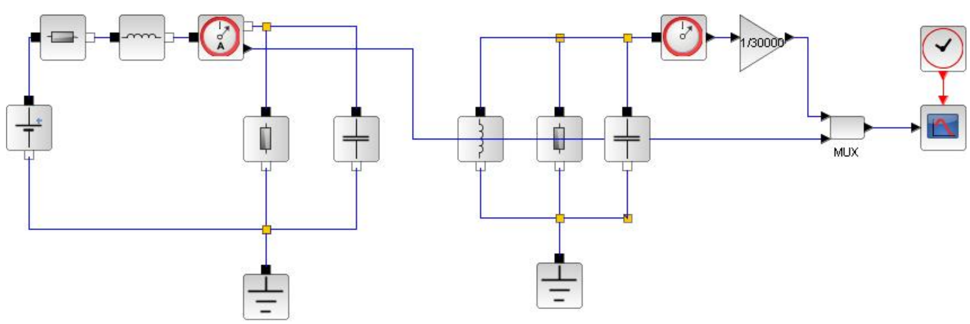

Figure 9.4: The Overdamped parallel circuit

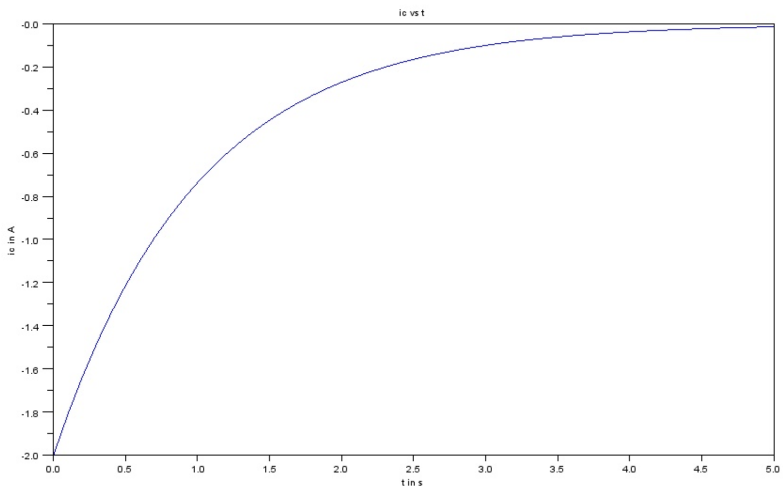

Figure 9.5: Graphical Representation of Overdamped response

Scilab code Exa 9.4 `Graphical Representation of Overdamped response`

```
clc
//Example 9.4
//Calculate settling time
t=0:0.1:5
ic=2*exp(-t)-4*exp(-t)
plot(t,ic)
xtitle('ic vs t','t in s','ic in A')
//Let ts be the settling time
//From the graph the maximum value is|-2|=2A
//'ts' is the time when ic has decreased to 0.02A
//On solving for 'ts'
ts=-log(0.02/4)
printf("ts=%3.2f s\n",ts)
```

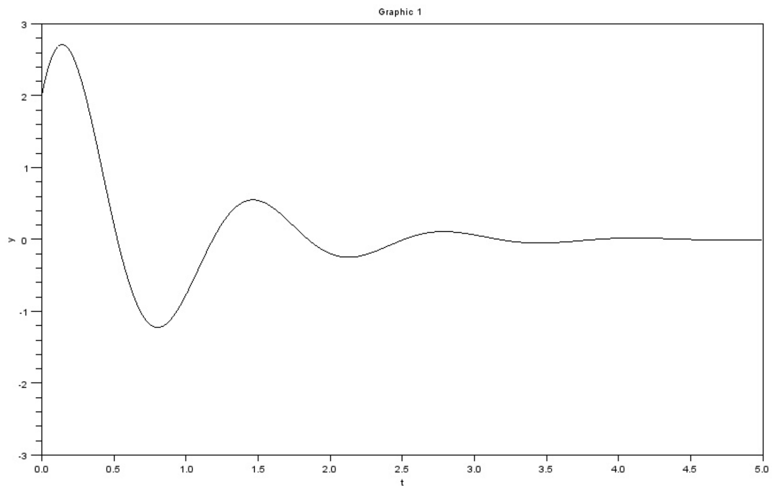

Figure 9.6: The Underdamped parallel RLC circuit

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

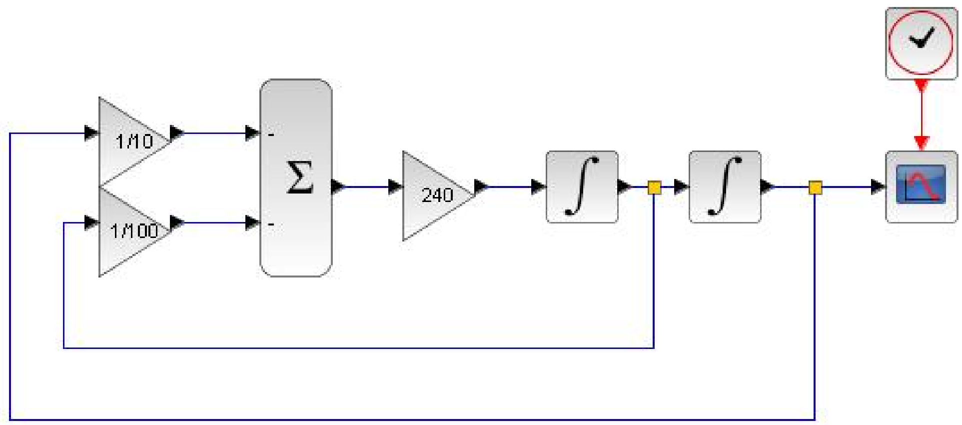

Figure 9.7: The Underdamped parallel RLC circuit

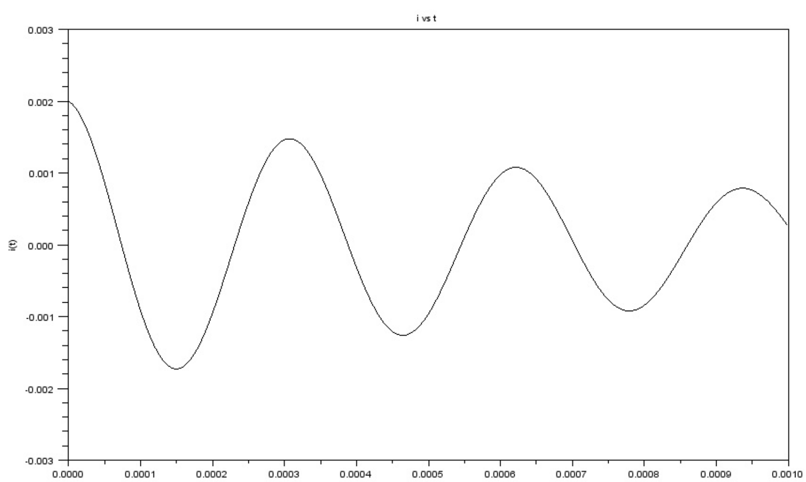

Figure 9.8: The Source free series RLC Circuit

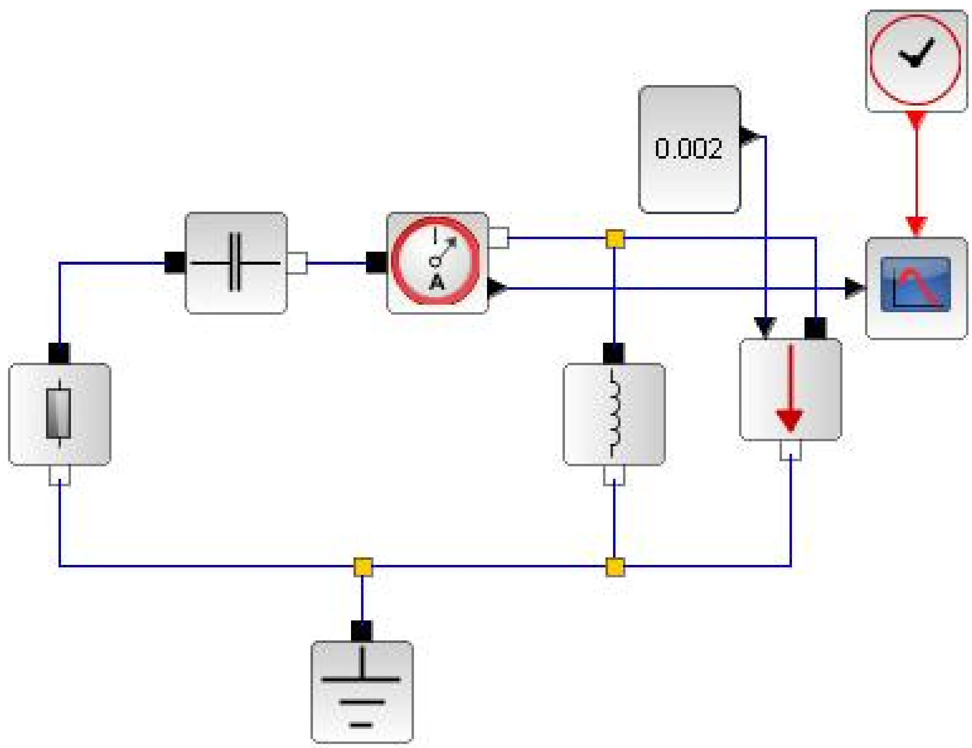

Figure 9.9: The Source free series RLC Circuit

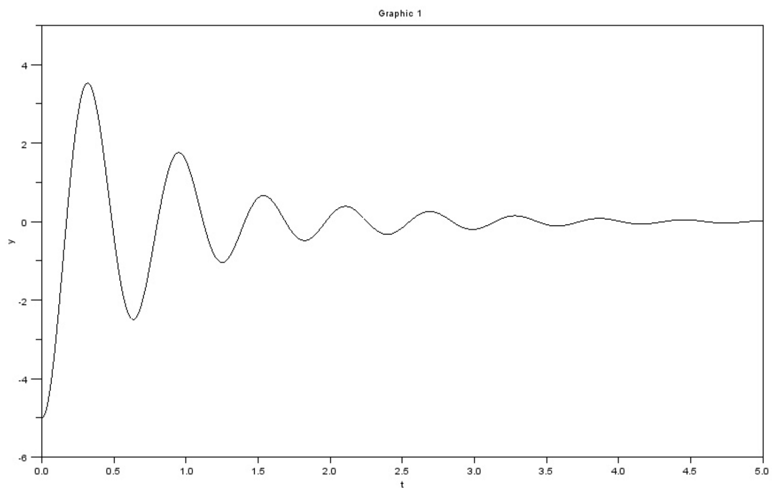

Figure 9.10: The Source free series RLC Circuit

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

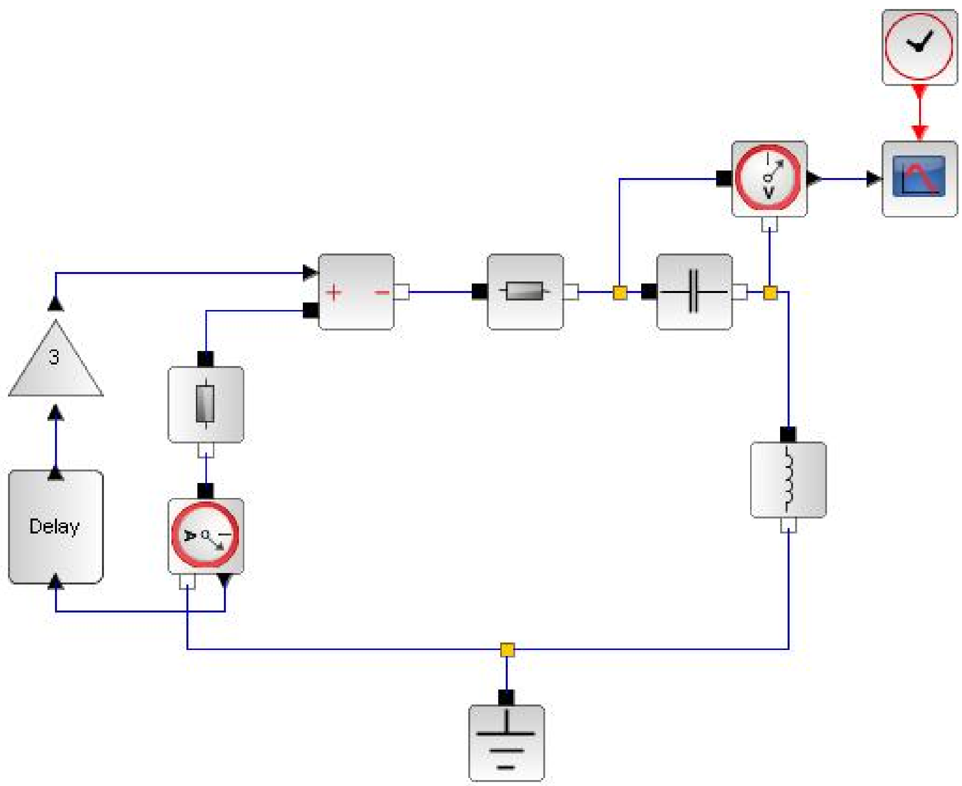

Figure 9.11: The Source free series RLC Circuit

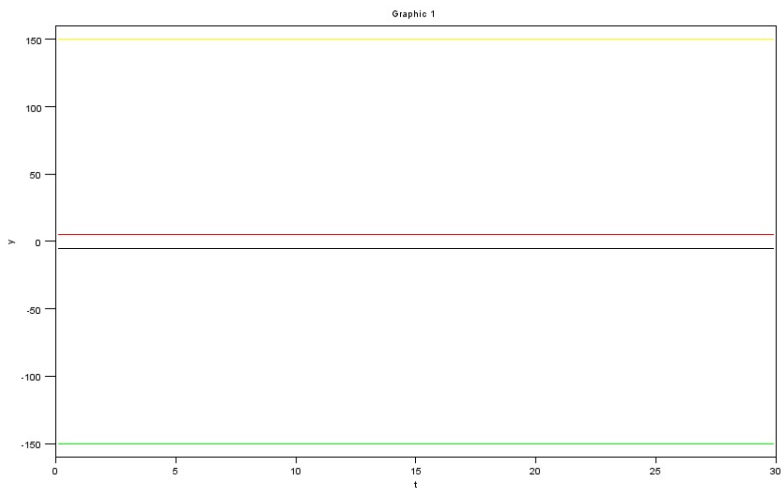

Figure 9.12: The Complete response of RLC circuit

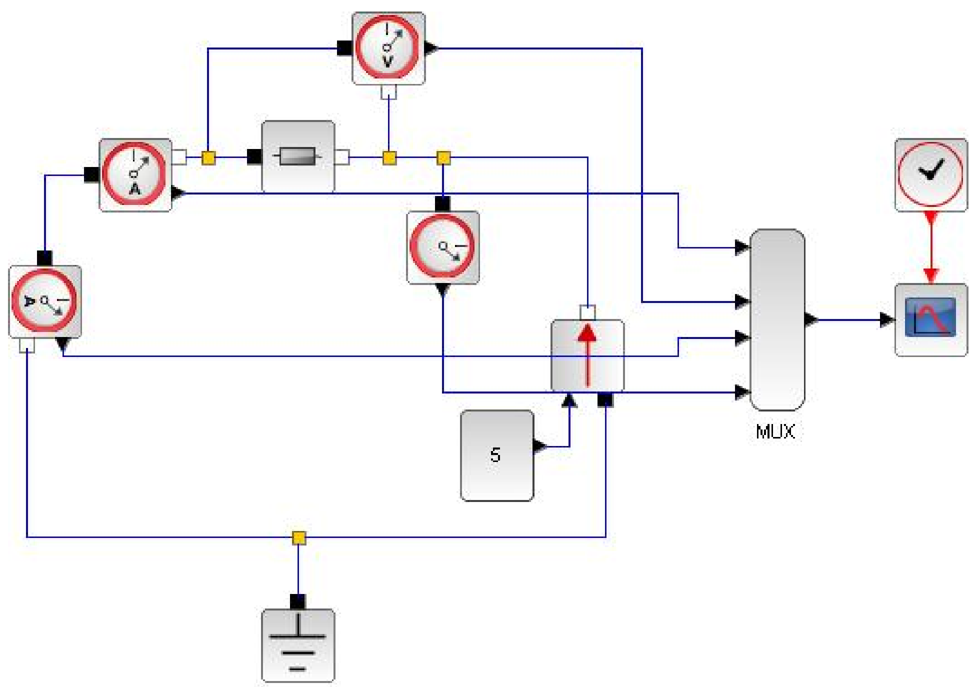

Figure 9.13: The Complete response of RLC circuit

Chapter 10

Sinusoidal Steady state Analysis

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 10.4 `The Inductor`

```
clc
//Example 10.4
//Determine phasor current and time-domain current
printf("Given")
disp('Voltage is 8(-50 deg),Frequency is 100rad/s,
   Inductance is 4H')
L=4;
w=100;
Vamp=8;Vang=-50;
//Let current be I
Iamp=Vamp/(w*L)
```

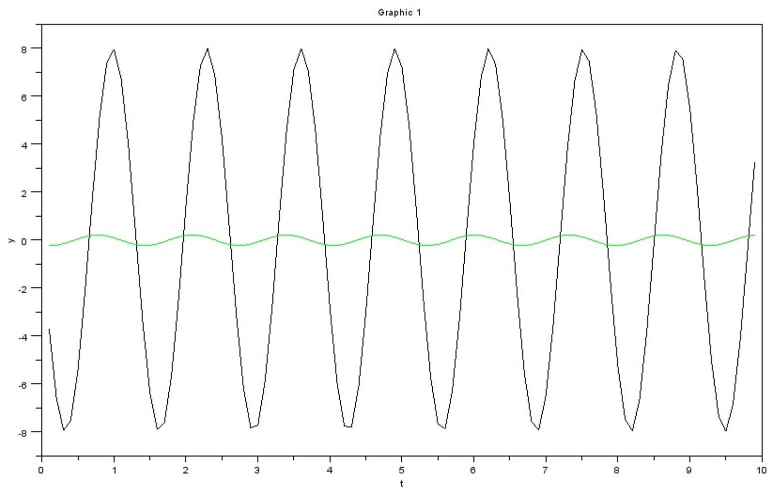

Figure 10.1: Forced Response to sinusoidal functions

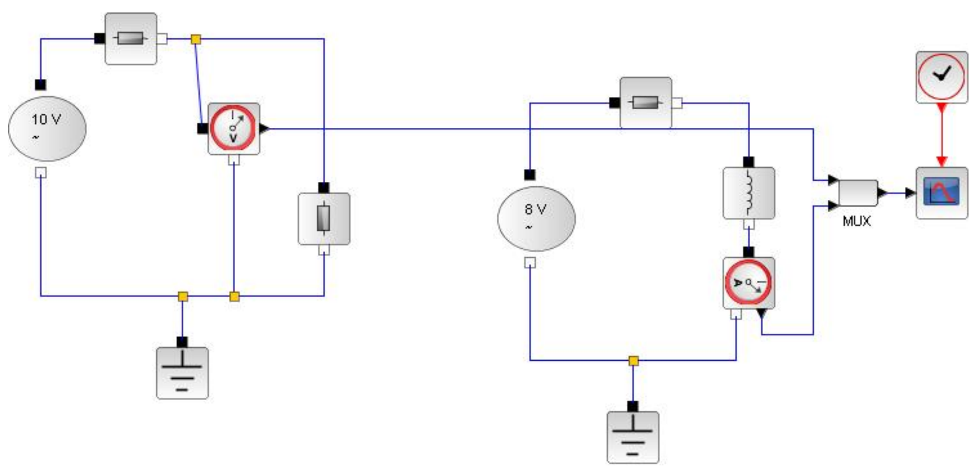

Figure 10.2: Forced Response to sinusoidal functions

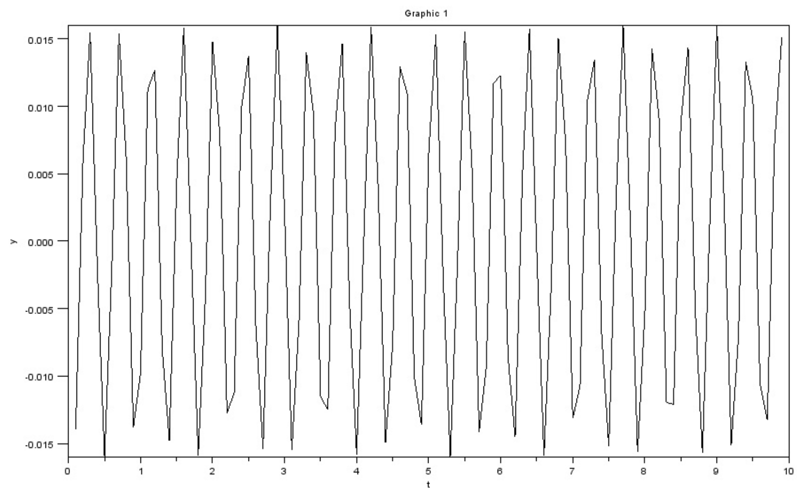

Figure 10.3: Impedance

```
Iang=-90+Vang
printf("I=%3.2f(%d deg) A \n",Iamp,Iang)
//In time domain
printf("i(t)=%3.2f *cos(%d*t%d) A",Iamp,w,Iang);
```

This code can be downloaded from the website wwww.scilab.in

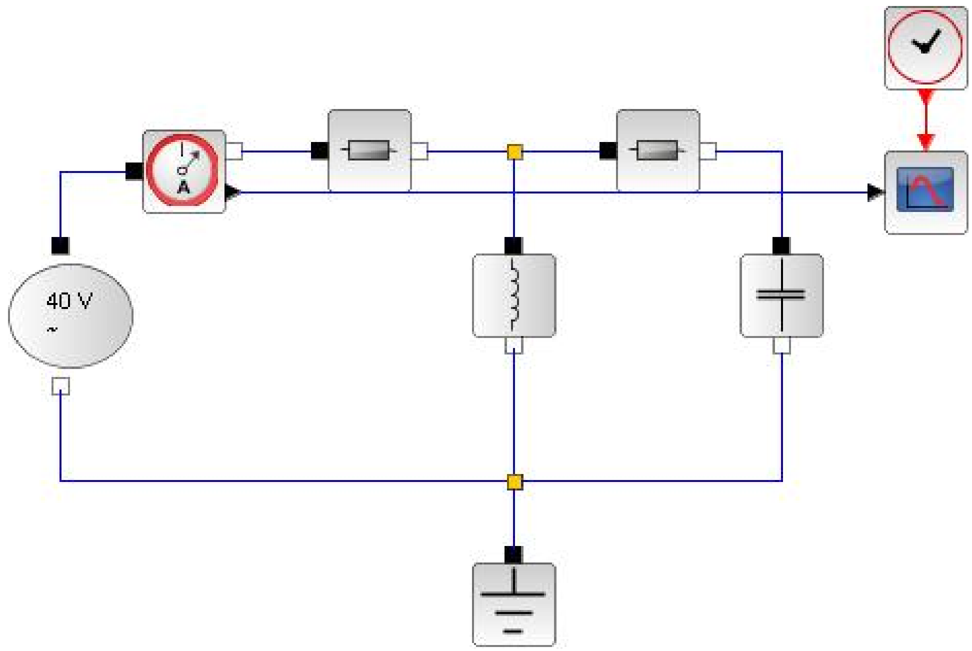

Figure 10.4: Impedance

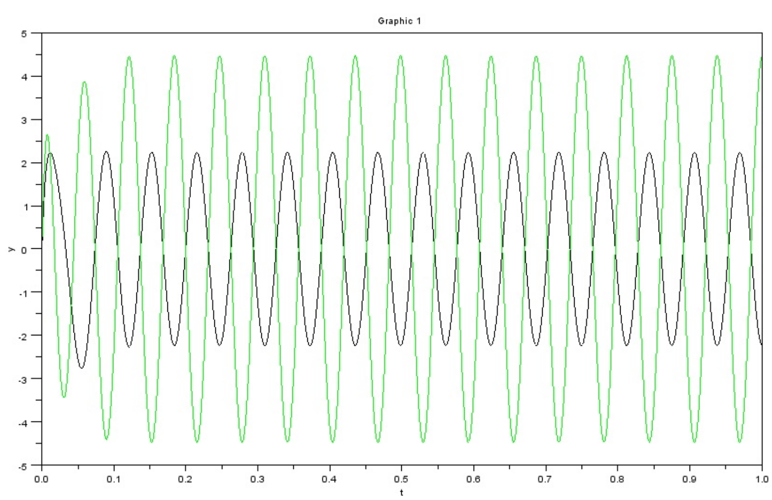

Figure 10.5: Nodal and Mesh analysis

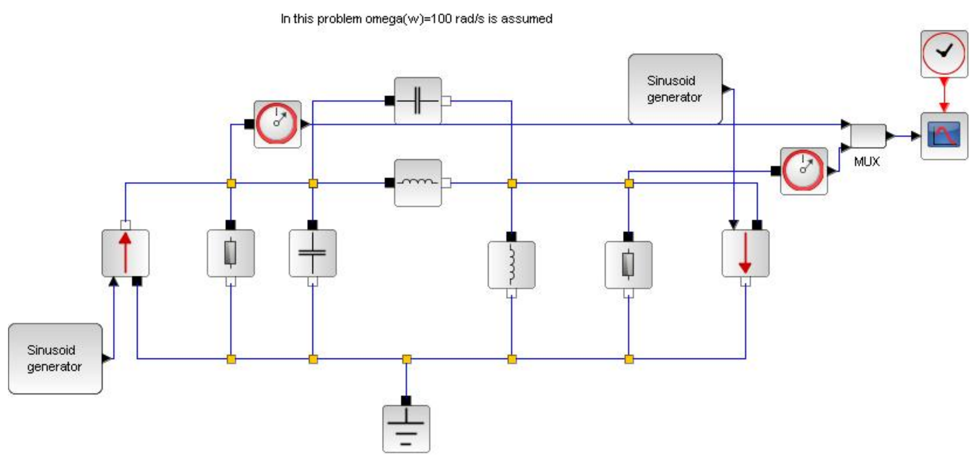

Figure 10.6: Nodal and Mesh analysis

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

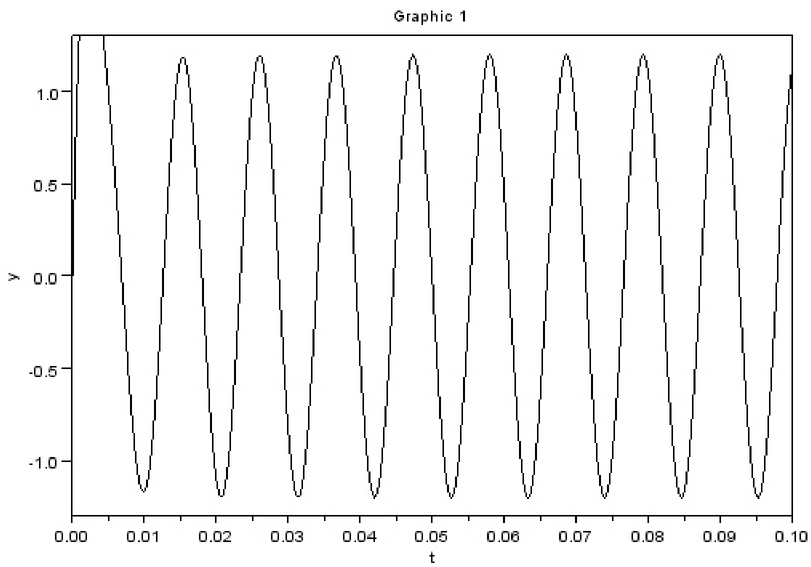

Figure 10.7: Nodal and Mesh analysis

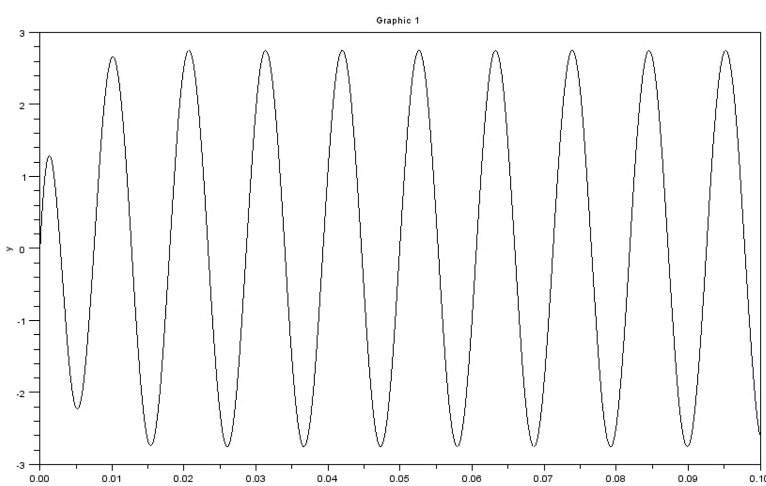

Figure 10.8: Nodal and Mesh analysis

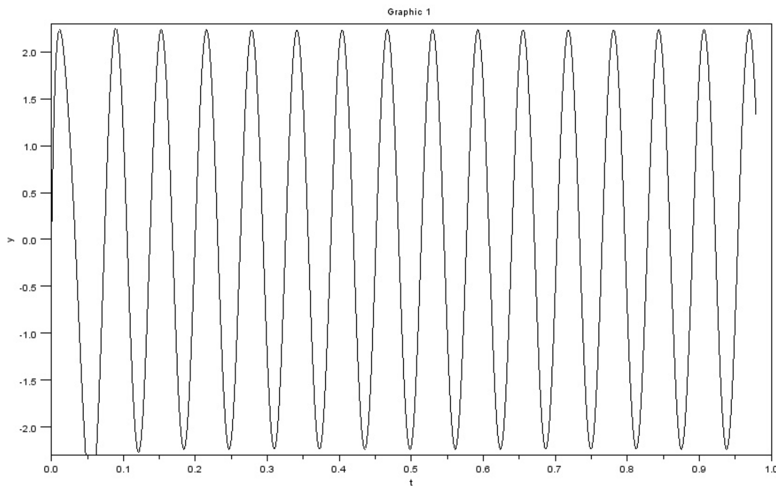

Figure 10.9: Superposition Source Transformations and Thevenin theorem

This code can be downloaded from the website wwww.scilab.in

This code can be downloaded from the website wwww.scilab.in

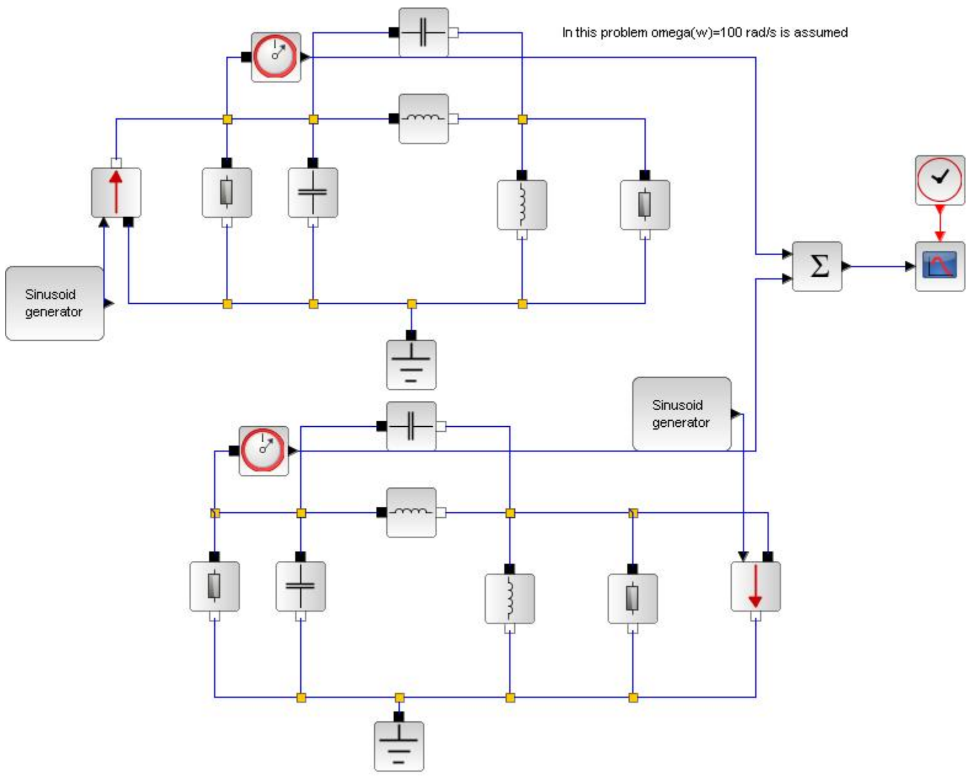

Figure 10.10: Superposition Source Transformations and Thevenin theorem

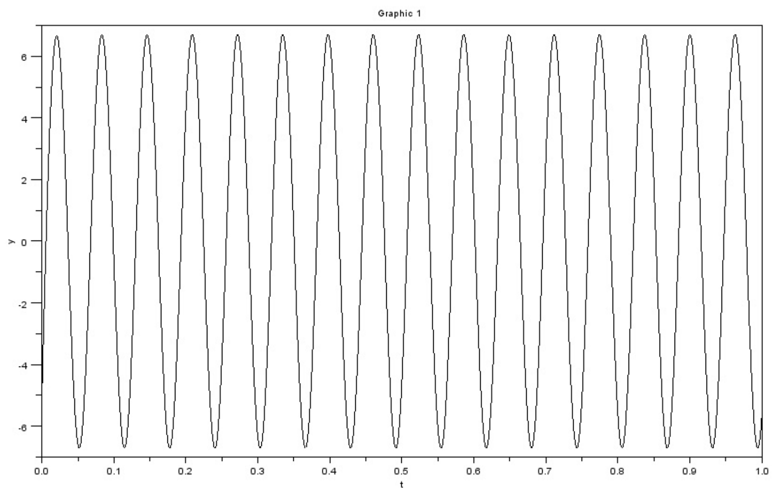

Figure 10.11: Superposition Source Transformations and Thevenin theorem

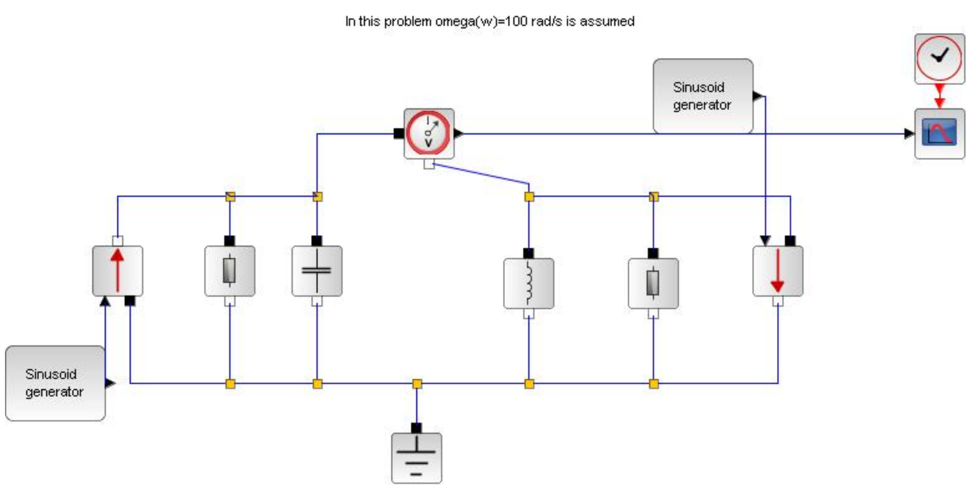

Figure 10.12: Superposition Source Transformations and Thevenin theorem

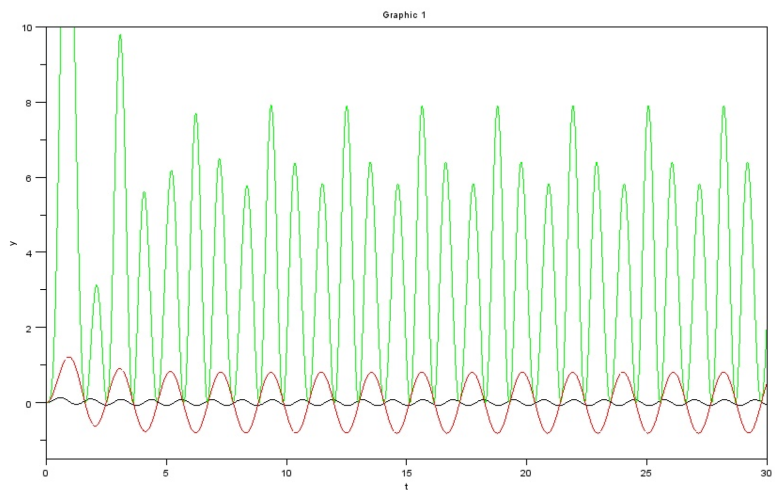

Figure 10.13: Superposition Source Transformations and Thevenin theorem

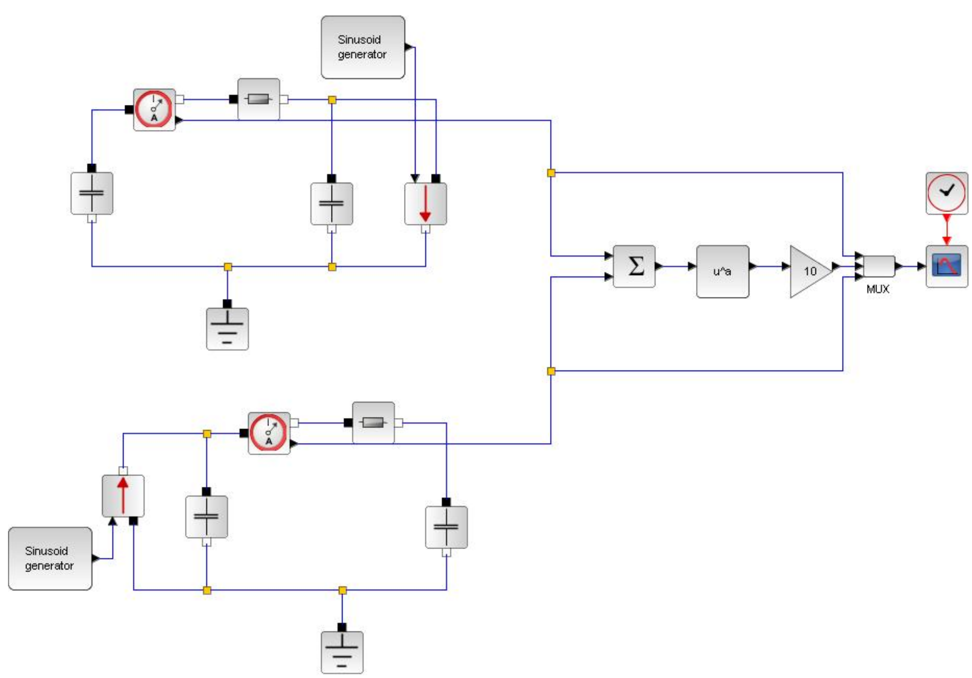

Figure 10.14: Superposition Source Transformations and Thevenin theorem

Chapter 11

AC Circuit Power Analysis

Scilab code Exa 11.1 Instantaneous Power

```
clc
//Example 11.1
//Calculate the powerr absorbed by capacitor and
   resistor
printf("Given")
disp('Capacitor 5uF, Resistor 200 ohm, Voltage
   source is 40+60*u(t)')
C=5*10^-6;R=200;
//For t<0 the value of u(t) is zero hence at t=0-
   the value of voltage is 40V
//For t=0+ the voltage is 100V
//At t=0+ the capacitor cannot charge
   instantaneously hence resistor voltage is 60V
disp('For t=0+')
VR=60;
i0=VR/R
T=R*C
t=1.2*10^-3
disp('The value of current is i(t)=i0*exp(-t/T)')
ival=i0*exp(-t/T)
printf("Value of resistor current at 1.2ms=%3.2f mA
```

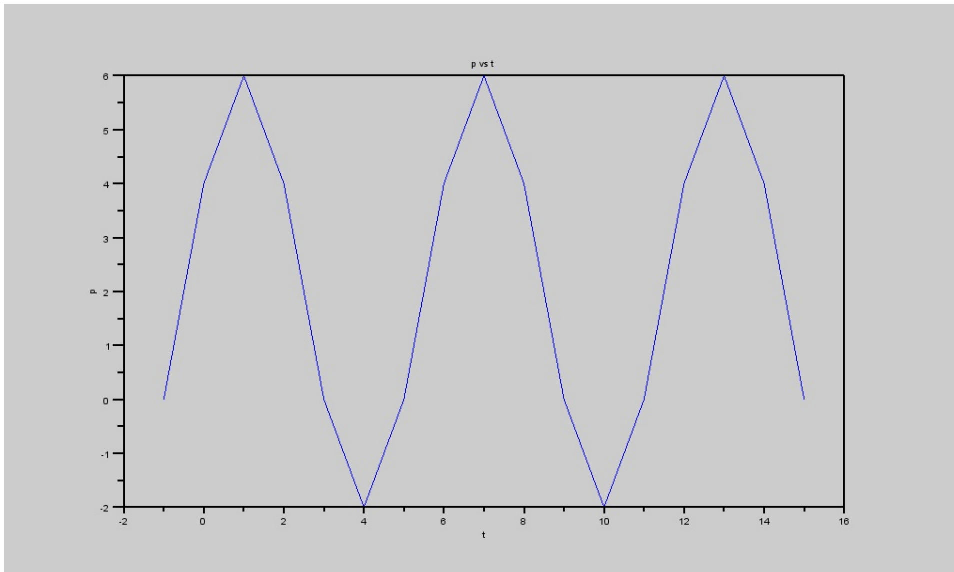

Figure 11.1: Average Power

```
    \n",ival*10^3)
//Let PR be the power absorbed by the resistor
PR=ival^2*R
printf("Value of resistive power at 1.2ms=%3.2f W \n
    ",PR)
//Out of the 100V available at t>0 the voltage
    across the capacitor is
disp('vC(t)=100-60*exp(-t/T)')
vCval=100-60*exp(-t/T)
printf("Value of capacitor voltage at 1.2ms=%3.2f V
    \n",vCval)
//Let PC be the power absorbed by the capacitor
PC=ival*vCval
printf("Value of capacitive power at 1.2ms=%3.2f W \
    n",PC)
```

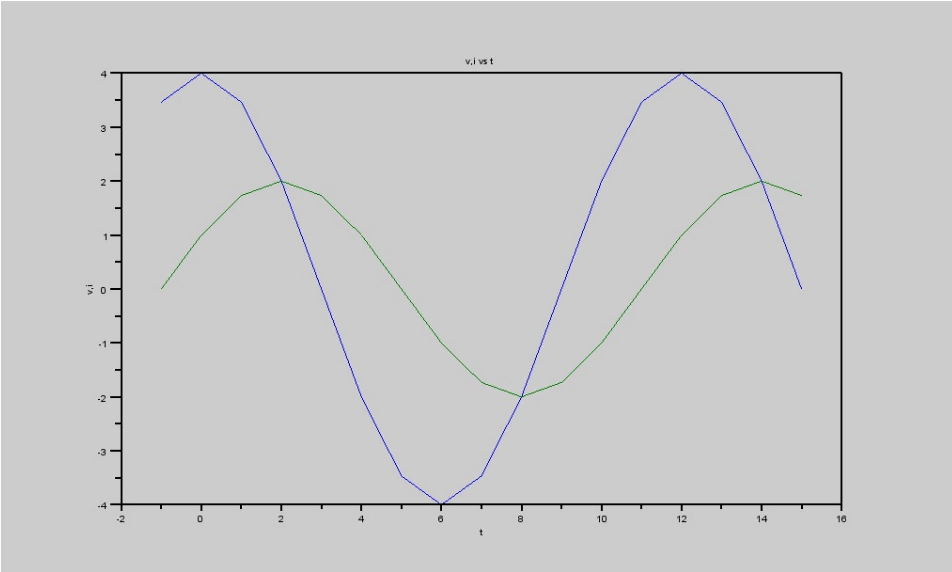

Figure 11.2: Average Power

Scilab code Exa 11.2 `Average Power`

```
clc
//Example 11.2
//Calculate the average power
printf("Given")
disp('v=4*cos(%pi/6*t), V=4(0 deg), Z=2(60 deg)')
Vamp=4;Vang=0;Zamp=2;Zang=60;
//Let I be the phasor current
Iamp=Vamp/Zamp
Iang=Vang-Zang
P=0.5*Vamp*Zamp*cos((Zang*%pi)/180)
printf("P=%d W \n",P);
t=-1:1:15
```

```
t1=-3:1:12
v=Vamp*cos(%pi/6*t)
//i=2*cos((%pi/6)*t-(%pi/3))
i=Iamp*cos(%pi/6*t+((Iang*%pi)/180))
figure
a= gca ();
plot (t,v,t,i)
xtitle ('v,i vs t' ,'t' ,'v,i');
a. thickness = 2;
//Instantaneous power p=v*i
//On solving
p=2+4*cos(%pi/3*t+((Iang*%pi)/180))
figure
a= gca ();
plot (t,p)
xtitle ('p vs t' ,'t' ,'p');
a. thickness = 2;
```

Scilab code Exa 11.3 Average Power

```
clc
//Example 11.3
//Calculate the Average Power
printf("Given")
disp('ZL=8-i*11 ohm, I=5(20 deg)A')
R=8;Iamp=5;
//We need to calculate the average power
//In the calculation of average power the resistance
    part of impedace only occurs
//Let P be the average power
P=0.5*Iamp^2*R
printf("Average Power=%d W \n",P)
```

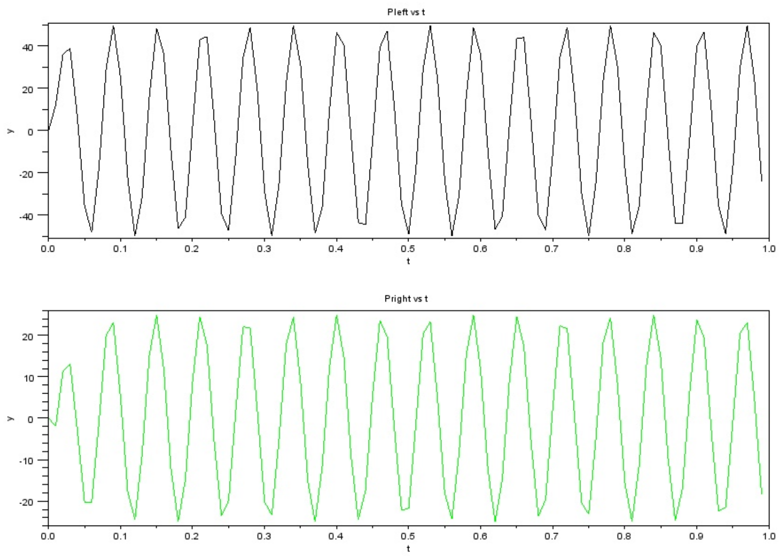

Figure 11.3: Average Power

Scilab code Exa 11.4 `Average Power`

```
clc
//Example 11.4
//Calculate the Average power absorbed and average
    power supplied by source
//From figure 11.6
//By applying mesh analysis
I1mag=11.18;I1ang=-63.43;I2mag=7.071;I2ang=-45;R=2;
    Vleft=20;Vright=10;
//Current through 2 ohm resistor
printf("I1-I2=%d(%d ang) A \n",5,-90)
//Average power absorbed by resistor
PR=0.5*5^2*R
```

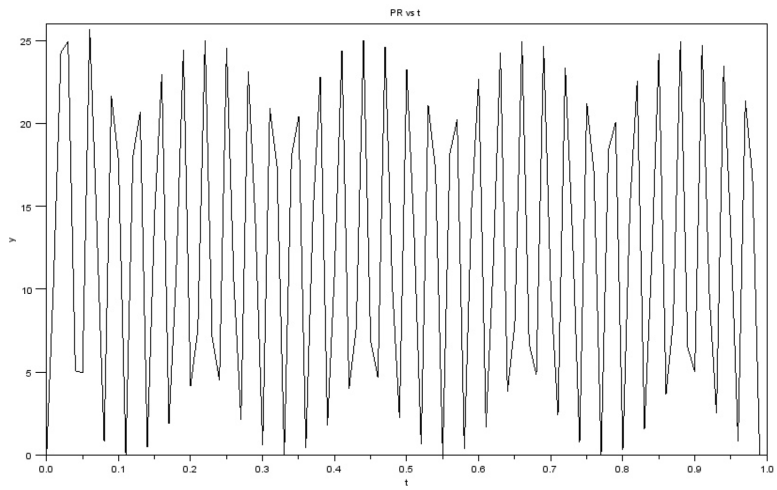

Figure 11.4: Average Power

```
printf("Average power absorbed by resistor=%d W \n",
   PR)
//Power supplied by left source
Pleft=0.5*Vleft*I1mag*cos(0-I1ang*%pi/180)
//Power supplied by right source
Pright=0.5*Vright*I2mag*cos(0+I2ang*%pi/180)
printf("Power supplied by sources \t Pleft=%d W \t
   Pright=%3.1f W",Pleft,Pright);
```

Scilab code Exa 11.6 `Average Power for Non periodic Functions`

```
clc
//Example 11.6
//Calculate the Average power
printf("Given")
disp('Resistor value is 4 ohm, i1=2*cos(10t)-3*cos
   (20t) A')
```

```
R=4;im1=2;im2=-3;
//Let P be the average power delievered
P=0.5*im1^2*R+0.5*im2^2*R
printf("Average power=%d W",P)
```

Scilab code Exa 11.7 Average Power for Non periodic Functions

```
clc
//Example 11.7
//Calculate the Average power
printf("Given")
disp('Resistor value is 4 ohm, i2=2*cos(10t)-3*cos
    (10t) A')
disp('On solving we get i2=-cos(10t)')
R=4;im=-1
//Let P be the average power delievered
P=0.5*im^2*R
printf("Average power=%d W",P)
```

Scilab code Exa 11.8 Apparent Power and Power factor

```
clc
//Example 11.8
//Calculate average power, power supplied by source
    and the power factor
printf("Given")
disp('Voltage source is 60 V,Load values are 2-i ohm
     and 1+5i ohm')
Vamp=60;Vang=0;
//Let Z be the cobined resistance
Z=2-%i+1+5*%i
[Zmag Zph]=polar(Z)
Isamp=Vamp/Zmag;
```

```
Isang=Vang-Zph;
printf("Ieff=%3.0f A rms and angle of Is is %3.2f
    degree\n",Isamp,(Isang*180)/%pi);
//Let Pupper be the power delievered to the upper
    load
Rtop=2;
Pupper=Isamp^2*Rtop
printf("Average Power delievered to the top load=%3
    .0f W \n",Pupper)
//Let Plower be the power delievered to the lower
    load
Rright=1;
Plower=Isamp^2*Rright
printf("Average Power delievered to the right load=
    %3.0f W \n",Plower)
//Let Papp be the apparent power
Papp=Vamp*Isamp
printf("Apparent Power =%3.0f VA \n",Papp)
//Let pf be the power factor
pf=(Pupper+Plower)/Papp
printf("power factor=%3.1f lag \n",pf)
```

Scilab code Exa 11.9 Complex Power

```
clc
//Example 11.9
printf("Given")
disp('Power of induction motor=50kW ,power factor is
     0.8 lag,Source voltage is 230V')
disp('The wish of the consumer is to raise the power
     factor to 0.95 lag')
//Let S1 be the complex power supplied to the
    indiction motor
V=230;Pmag=50*10^3;pf=0.8;
Pang=(acos(pf)*180)/%pi
```

```
S1mag=Pmag/pf
S1ph=Pang
x=S1mag * cos (( Pang * %pi ) /180) ;
y=S1mag * sin (( Pang * %pi ) /180) ;
z= complex (x,y)
disp(z ,'S1=')
//To achieve a power factor of 0.95
pf1=0.95
//Now the total complex power be S
P1ang=(acos(pf1)*180)/%pi
Smag=Pmag/pf1
Sph=P1ang
a=Smag * cos (( P1ang * %pi ) /180) ;
b=Smag * sin (( P1ang * %pi ) /180) ;
c= complex (a,b)
disp(c,'S=')
//Let S2 be the complex power drawn by the
   corrective load
S2=c-z
disp(S2,'S2=')
disp('Let a phase angle of voltage source selected
   be 0 degree')
//Let I2 be the current
I2=-S2/V
//Let Z2 be the impedance of corrective load
Z2=V/I2
disp(Z2,'Z2=')
```

Chapter 12

Polyphase Circuits

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 12.2 `Three phase Wye connection`

```
clc
//Example 12.2
//Calculate total power dissipated
disp('Given')
disp('Van=200 with angle 0 degree and Zp=100with angle 60 degree')
Zpamp=100;Zpang=60
//Since one of the phase voltage is given, we need to find other phase voltages
Vanamp=200;Vbnamp=200 ; Vcnamp=200;
Vanang=0;Vbnang=-120;Vcnang=-240;
disp('The phase voltages are')
printf("Van=%d /_%d deg V\tVbn=%d /_%d deg V\tVcn=%d /_%d deg V\t",Vanamp,Vanang,Vbnamp,Vbnang,Vcnamp,Vcnang)
```

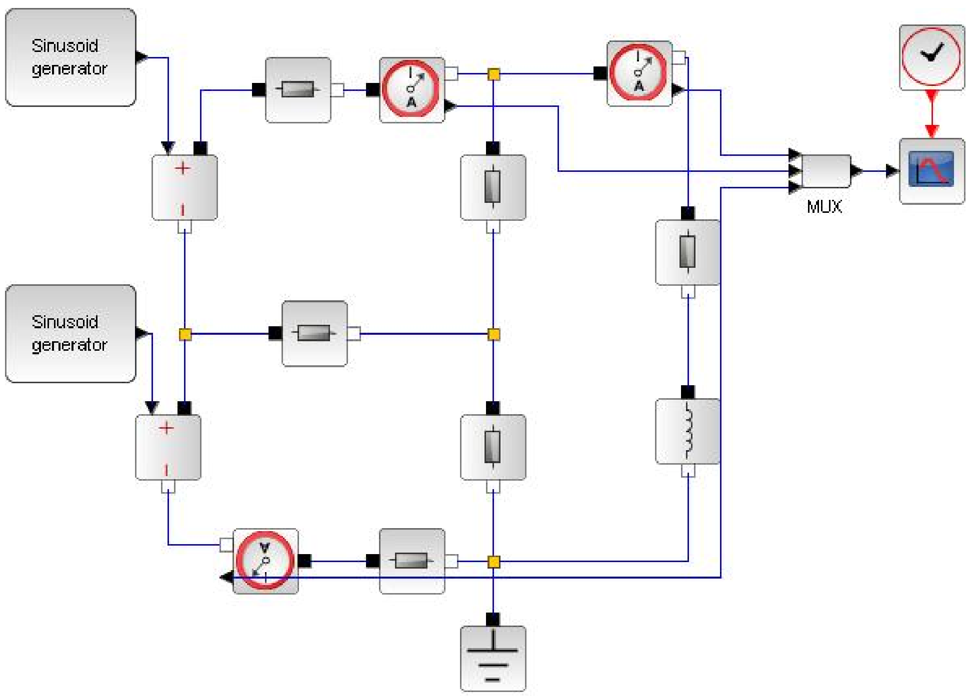

Figure 12.1: Single phase three wire systems

```
//Now we will find line voltages
//Let line voltage be Vline
Vline=200*sqrt(3)
//By constructing a phasor diagram
disp('The line voltages are')
printf("\n Vab=%d /_%d deg V\tVbc=%d /_%d deg V\tVca=%d /_%d deg V\t",Vline,30,Vline,-90,Vline,-210)

//Let the line current be IaA
IaAamp=Vanamp/Zpamp
IaAang=Vanang-Zpang
//Since the given system is a balanced three phase system
//From phasor diagram as shown in figure 12.16
disp('The line currents are')
printf("\n IaA=%d /_%d deg V\tIbB=%d /_%d deg V\tIcC=%d /_%d deg V\t",IaAamp,IaAang,IaAamp,IaAang-120,IaAamp,IaAang-240)
//Let power absorbeed by phase A is PAN
PAN=Vanamp*IaAamp*cos(((Vanang+IaAang)*%pi)/180)
printf("\n Total average power = %d W",3*PAN)
```

Scilab code Exa 12.3 Three phase Wye connection

```
clc
//Example 12.3
//Calculate the line current and phase impedance
disp('Given')
disp('Line voltage = 300V, Power factor=0.8(lead), Phase power = 1200W')
Vline=300;pf=0.8;PW=1200;
Vp=Vline/sqrt(3)
PerPhpower=PW/3;
//Line current can be found as
```

```
IL=PerPhpower/(pf*Vp)
printf("Line current= %3.2f A \n",IL)
//Let Zp be the phase impedance
Zpmag=Vp/IL
//Sice power factor is 'leading'
Zpang=-(acos(0.8)*180)/%pi
printf("Phase impedance = %d/_%3.2f deg ohm",Zpmag,
   Zpang);
```

Scilab code Exa 12.4 Three phase Wye connection

```
clc
//Example 12.4
//Calculate the line current
//Continuing from example 12.3
Vp=300/sqrt(3);
IL=2.89;pf=0.8
disp('A balanced 600W lighting load is added in
   parallel with the existing load')
disp('600W if balanced then 200W will be consumed by
    each phase')
Vpadd=200;
//From figure 12.17
I1=Vpadd/Vp
disp('Load current is unchanged')
I2mag=IL
I2ph=(acos(pf)*180)/%pi
x=I2mag * cos (( I2ph * %pi ) /180) ;
y=I2mag * sin (( I2ph * %pi ) /180) ;
z= complex (x,y)
disp(z)
ILnew=I1+z
[ILmag ILph]=polar(ILnew)
printf("Line current=%3.2f /_%3.2f deg A \n ",ILmag,
   ILph*(180/%pi));
```

Scilab code Exa 12.5 The Delta connection

```
clc
//Example 12.5
//Calculate amplitude of line current
disp('Given')
disp('Line voltage = 300V, Power factor=0.8(lag),
   Phase power = 1200W')
Vline=300;pf=0.8;PW=1200;
disp('1200W will be consumed as 400W in each phase')
Vp=400
//Phase current be Ip
Ip=Vp/(Vline*pf)
//Let amplitude of line current be IL
IL=Ip*sqrt(3)
printf("Line current=%3.2f A \n",IL)
//Let Zp be the phase impedance
Zpmag=Vline/Ip
//Sice power factor is 'lagging'
Zpang=(acos(0.8)*180)/%pi
printf("Phase impedance = %d(%3.2f deg)ohm",Zpmag,
   Zpang);
```

Scilab code Exa 12.6 The Delta connection

```
clc
//Example 12.6
//Calculate amplitude of line current
disp('Given')
disp('Line voltage = 300V, Power factor=0.8(lag),
   Phase power = 1200W')
```

```
Vline=300;pf=0.8;PW=1200;
Vph=Vline/sqrt(3)
disp('1200W will be consumed as 400W in each phase')
Vp=400
//Let phase current be Ip
Ip=Vp/(Vph*pf)
printf("Phase current=%3.2f A \n",Ip)
//Let Zp be the phase impedance
Zpmag=Vph/Ip
//Sice power factor is 'lagging'
Zpang=(acos(0.8)*180)/%pi
printf("Phase impedance = %d(%3.2f deg)ohm\n",Zpmag,
    Zpang);
//PW=sqrt(3)*VL*IL*pf
IL=PW/(sqrt(3)*Vline*pf)
printf("Line current=%3.2f A \n",IL)
```

Scilab code Exa 12.7 Power measurement in three phase systems

```
clc
//Example 12.7
//Determine wattmeter reading and total power drawn
    by the load
disp('Given')
disp('Vab=230(0 deg)V')
Vline=230
//Since positive phase sequence is used
disp('The line voltages are')
printf("\n Vab=%d (%d deg)V\tVbc=%d (%d deg) V\tVca=
    %d (%d deg)V\t",Vline,0,Vline,-120,Vline,120)
Vacamp=Vline;
Vacang=-60;
Vbcamp=Vline;
Vbcang=-120;
//Now we will evaluate phase current
```

```
//Let IaA be the phase current
Vanamp=Vline/sqrt(3)
Vanph=-30
//From figure 12.28
Zph=4+%i*15
[Zphmag Zphang]=polar(Zph)
IaAamp=Vanamp/Zphmag
IaAang=Vanph-(Zphang*180)/%pi
IbBang=IaAang+240
printf("\nIaA=%3.2f(%3.2f deg)A\n",IaAamp,IaAang);
//Power rating of each wattmeter is now calculated
//Power measured by wattmeter #1
P1=Vline*IaAamp*cos(((Vacang-IaAang)*%pi)/180)
printf("P1=%d W \n",P1)
//Power measured by wattmeter #2
P2=Vline*IaAamp*cos(((Vbcang-IbBang)*%pi)/180)
printf("P2=%3.2f W \n",P2)
//Net power be P
P=P1+P2
printf("P=%3.2f W \n",P)
```

Chapter 13

Magnetically coupled circuits

Scilab code Exa 13.2 `Mutual Inductance`

```
clc
//Example 13.2
disp('Given')
disp('Input voltage is 10V')
Viamp=10
//From figure 13.7
//Writing the left mesh equations
disp('(1+10i)*I1-90i*I2=10')
//Writing the right mesh equations
disp('(400+1000i)*I2-90i*I1=0')
i=%i
A=[1+10*i -90*i;-90*i 400+1000*i]
i2mat=[1+10*i 10; -90*i 0]
//Find i2
i2=det(i2mat)/det(A)
[mag Theta]=polar(i2)
Theta=(Theta*180)/%pi
//The value of resistor is 400 ohm
R=400;
//Let V=V2/V1
Vamp=R*mag/Viamp
```

```
printf("Ratio of output voltage to input is %3.2f with angle %3.2f degrees",Vamp,Theta);
```

Scilab code Exa 13.4 Energy considerations

```
clc
//Example 13.4
disp('Given')
disp('L1=0.4H L2=2.5H k=0.6 i1=4i2=20*cos(500t-20)mA')
L1=0.4;L2=2.5;k=0.6;
disp('a)')
t=0;
i2=5*cos(500*t-(20*%pi)/180)
printf("i2(0)=%3.2f mA \n",i2)
disp('b)')
M=k*sqrt(L1*L2)
//v1(t)=L1*d/dt(i1)+M*d/dt(i2)
v1=-L1*20*500*10^-3*sin(500*t-(20*%pi)/180)-M*5*500*10^-3*sin(500*t-(20*%pi)/180)
printf("v1(0)=%3.2f V \n",v1)
disp('c')
//The total energy can be found as
w=(L1*(4*i2)^2)/2+ (L2*(i2)^2)/2+M*(4*i2)*(i2)
printf("w=%3.2f uJ \n",w)
```

Scilab code Exa 13.5 T and PI equivalent networks

```
clc
//Example 13.5
printf("Given")
disp('L1=30 mH L2=60 mH M=40 mH')
L1=30*10^-3; L2=60*10^-3; M=40*10^-3;
```

```
//The equivalent T network is
UL=L1-M
UR=L2-M
CS=M
printf("The T network has \n")
printf("%d mH in the upper left arm\n",UL*10^3)
printf("%3.0f mH in the upper right arm\n",UR*10^3)
printf("%d mH in the center stem\n",CS*10^3)
```

Scilab code Exa 13.6 T and PI equivalent networks

```
clc
//Example 13.6
printf("Given")
disp('L1=30 mH L2=60 mH M=40 mH')
L1=30*10^-3; L2=60*10^-3; M=40*10^-3;
//Let X=L1*L2-M^2
X=L1*L2-M^2
//The equivalent PI network is
LA=X/(L2-M)
LB=X/M
LC=X/(L1-M)
printf("The PI network has \n")
printf("LA=%3.0f mH\n",LA*10^3)
printf("LB=%3.0f mH \n",LB*10^3)
printf("LC=%3.0f mH\n",LC*10^3)
```

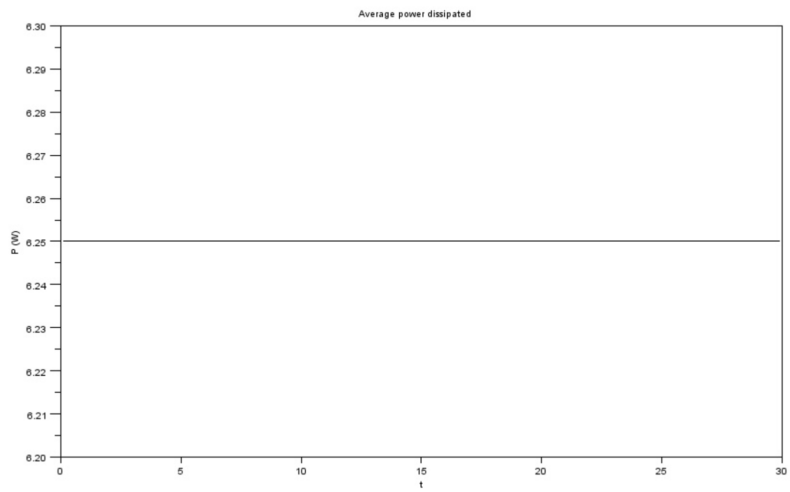

Figure 13.1: The Ideal Transformer

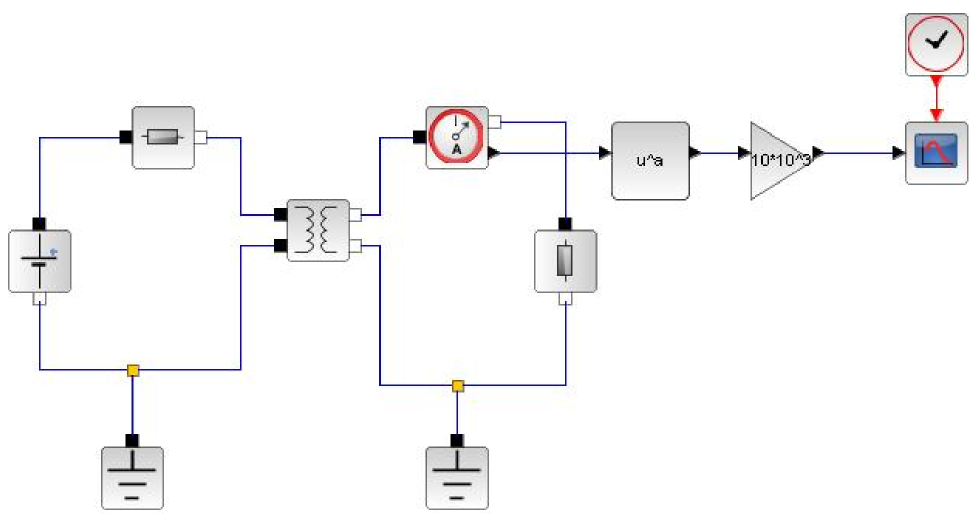

Figure 13.2: The Ideal Transformer

Scilab code Exa 13.8 Equivalent Circuits

```
clc
//Example 13.8
disp('Given')
disp('Vin=50V Zg=100 ohm')
Vin=50;Zg=100;
//From figure 13.32
disp('When the secondary circuit and ideal
    transformer is replaced by a Thevenin equivalent
    then the primary circuit sees a 100 ohm impedance
    ')
//The turns ratio is a
a=10;
disp('We place the secondary circuit and ideal
    transformer by a Thevenin equivalent circuit')
Vth=-a*Vin
Zth=(-a)^2*Zg
printf("The secondary circuit has voltage source %d
    V rms with %d kohm resistance in series with it
    along with %d kohm load resistance",Vth,Zth
    *10^-3,10)
```

Chapter 14

Complex frequency and the Laplace Transform

Scilab code Exa 14.2 Definition of the Laplace Transform

```
//Example 14.2
//Install Symbolic toolbox
//Find the Laplace transform
syms t s
clc
z=integ(2*exp(-s*t),t,3,%inf)
//The second term will result in zero
disp(z,'F(s)=')
```

Scilab code Exa 14.3 Inverse Transform Techniques

```
clc
//Example 14.3
//Install Symbolic toolbox
//Find the Inverse Laplace transform
syms s
```

```
a=7/s
b=31/(s+17)
x=ilaplace(a)
y=ilaplace(b)
g=x-y
disp(g,'g(t)=')
```

Scilab code Exa 14.4 Inverse Transform Techniques

```
clc
//Example 14.4
//Install Symbolic toolbox
//Find the Inverse Laplace transform
syms s t
a=2
b=4/s
x=ilaplace(b)
//Inverse laplace transform of a constant is
disp('inverse laplace(2)=2*delta(t)')
disp('Answer is')
disp(x+'2*delta(t)')
```

Scilab code Exa 14.5 Inverse Transform Techniques

```
clc
//Example 14.5
//Install Symbolic toolbox
//Find the Inverse Laplace transform
syms s
s=%s;
P =(7*s+5)/(s^2+s);
Pp=pfss (P)
p1=ilaplace (Pp(1))
```

```
p2=ilaplace (Pp(2))
p=p1+p2
disp(p,'p(t)=');
```

Scilab code Exa 14.6 Inverse Transform Techniques

```
clc
//Example 14.6
//Install Symbolic toolbox
//Find the Inverse Laplace transform
syms s
s=%s;
V =2/(s^3+12*s^2+36*s);
Vp=pfss (V)
v1=ilaplace (Vp(1))
v2=ilaplace (Vp(2))
v=v1+v2
disp(v,'v(t)=');
```

Scilab code Exa 14.7 Basic Theorems for the Laplace Transform

```
clc
//Example 14.7
//Install Symbolic toolbox
//Find the current through 5 ohm resistor
syms s
s=%s
//From figure 14.3
//Writing the KVL equation and taking the Laplace
   transform
I=1.5/(s*(s+2))+5/(s+2)
I1=1.5/(s*(s+2))
I2=5/(s+2)
```

```
I1p=pfss(I1)
i1=ilaplace(I1p(1))
i2=ilaplace(I1p(2)+I2)
i=i1+i2
disp(i,'i(t)=')
```

Scilab code Exa 14.8 Basic Theorems for the Laplace Transform

```
clc
//Example 14.8
//Install Symbolic toolbox
//Find the current for t>0
syms s
s=%s
//From figure 14.5
//Writing the KVL equation and taking the Laplace
   transform
I=-2/(s+4)
i=ilaplace(I)
disp(i,'i(t)=')
```

Scilab code Exa 14.9 Basic Theorems for the Laplace Transform

```
clc
//Example 14.9
//Install Symbolic toolbox
//Find the voltage v(t)
syms s
s=%s
//From figure 14.6
//Writing the KCL equation and taking the Laplace
   transform
V=4/(s*(s+4))+9/(s+4)
```

```
V1=4/(s*(s+4))
V2=9/(s+4)
V1p=pfss(V1)
v1=ilaplace(V1p(1))
v2=ilaplace(V1p(2)+V2)
v=v1+v2
disp(v,'v(t)=')
```

Scilab code Exa 14.10 The time shift theorem

```
clc
//Example 14.10
//Install Symbolic toolbox
//Determine the transform of rectangular pulse
syms t s
v=integ(exp(-s*t),t,2,%inf)-integ(exp(-s*t),t,5,%inf
  )
disp(v,'V(s)=')
```

Scilab code Exa 14.11 The Initial and Final value theorems

```
clc
//Example 14.11
//Install Symbolic toolbox
//Calculate f(inf)
syms s t ;
disp('Given function is f(t)=1-exp(-a*t)')
u=laplace(1)
v=laplace(exp(-2*t))
F=u-v
x=s*F
//From final value theorem
y=limit(x,s,0)
```

```
disp(y,'f(inf)=')
```

Chapter 15

Circuit Analysis in the s domain

Scilab code Exa 15.1 Modeling Inductors in the s domain

```
clc
//Example 15.1
//Install Symbolic toolbox
//Calculate the voltage
//From figure 15.3
//Writing the KVL equation for the voltage and
   taking the Laplace transform
syms s
s=%s
disp('V=(2*s*(s+9.5)/((s+8)*(s+0.5)))-2')
//On solving
V=(2*s-8)/((s+8)*(s+0.5))
Vp=pfss (V)
Vp1=ilaplace(Vp(1))
Vp2=ilaplace(Vp(2))
v=Vp1+Vp2
disp(v,'v(t)=')
```

Scilab code Exa 15.2 Modeling capacitors in the s domain

```
clc
//Example 15.2
//Install Symbolic toolbox
//Calculate the voltage
//Selecting the current based model
//From figure 15.6(b)
//Writing the KCL equation for the voltage and
   taking the Laplace transform
syms s
s=%s
Vc=-2*(s-3)/(s*(s+2/3))
Vcp=pfss (Vc)
Vcp1=ilaplace(Vcp(1))
Vcp2=ilaplace(Vcp(2))
vc=Vcp1+Vcp2
disp(vc,'vc=')
```

Scilab code Exa 15.4 Nodal and Mesh analysis in s domain

```
clc
//Example 15.4
//Install Symbolic toolbox
//Calculate the voltage
//From figure 15.9
//Applying nodal equation and solving for vx
syms s
s=%s
Vx=(10*s^2+4)/(s*(2*s^2+4*s+1))
Vxp=pfss (Vx)
Vxp1= ilaplace (Vxp(1))
```

```
Vxp2= ilaplace (Vxp(2))
vx=Vxp1+Vxp2
disp(vx,'vx=')
```

Scilab code Exa 15.6 Additional circuit analysis techniques

```
clc
//Example 15.6
//Install Symbolic toolbox
//Calculate the voltage
//Performing source transformatiom on the s-domain
   circuit
//Solving for V(s)
syms s
s=%s
V=(180*s^4)/((s^2+9)*(90*s^3+18*s^2+40*s+4))
Vp=pfss (V)
Vp1=ilaplace(Vp(1))
Vp2=ilaplace(Vp(2))
Vp3=ilaplace(Vp(3))
v=Vp1+Vp2+Vp3
disp(v,'v(t)=')
```

Scilab code Exa 15.9 Convolution and Laplace Transform

```
clc
//Example 15.9
//Install Symbolic toolbox
//Find the inverse Laplace transform
syms s
s=%s
//Let a=1 and b=3
a=1;b=3;
```

```
V=1/((s+a)*(s+b))
Vp=pfss (V)
Vp1=ilaplace(Vp(1))
Vp2=ilaplace(Vp(2))
v=Vp1+Vp2
disp(v,'v(t)=')
```

Scilab code Exa 15.10 Convolution and Laplace Transform

```
clc
//Example 15.10
//Since the input function is given the Laplace transform is found
syms s t
s=%s
vin=6*exp(-t)
Vin=laplace(vin)
//Connecting the impulse voltage pulse to the circuit and converting to s-domain
//If vin=delta(t)..the impulse source
V0=2/((2/s)+2)
//As source voltage is 1V
H=V0
V=Vin*H
Vp=pfss ((6*s)/(s+1)^2)
Vp1=ilaplace(Vp(1))
v0=Vp1
disp(v0,'v0(t)=')
```

Chapter 16

Frequency Response

Scilab code Exa 16.1 Parallel Resonance

```
clc
//Example 16.1
disp('Given')
disp('L=2.5mH Q0=5 C=0.01uF')
L=2.5*10^-3; Q0=5; C=0.01*10^-6;
w0=1/sqrt(L*C)
printf("w0= %3.1f krad/s \n",w0*10^-3);
f0=w0/(2*%pi)
alpha=w0/(2*Q0)
printf("alpha= %3.1f Np/s \n",alpha);
wd=sqrt(w0^2-alpha^2)
printf("wd= %3.1f krad/s \n",wd*10^-3);
R=Q0/(w0*C)
printf("R= %3.2f ohm \n",R*10^-3);
```

Scilab code Exa 16.2 Bandwidth and high Q circuits

```
clc
```

```
//Example 16.2
disp('Given')
disp('R=40Kohm L=1H C=1/64 uF w=8.2krad/s')
R=40*10^3; L=1; C=1/64 *10^-6; w=8.2*10^3;
//The value of Q0 must be at least 5
Q0=5;
w0=1/sqrt(L*C)
printf("w0= %3.1f krad/s \n",w0*10^-3);
f0=w0/(2*%pi)
B=w0/Q0
printf("Bandwidth= %3.1f krad/s \n",B*10^-3);
//Number of half bandwidths be N
N=2*(w-w0)/B
disp(N)
//Admittance Y(s)=(1+i*N)/R
//Finding the magnitude and angle
magY=sqrt(1+N^2)/R
angY=atan(N)*(180/%pi)
disp(angY,'angY=')
printf("admittance value=%3.2f uS",magY*10^6)
```

Scilab code Exa 16.3 Series Resonance

```
clc
//Example 16.3
disp('Given')
disp('R=10 ohm L=2mH C=200 nF w=48 krad/s vs=100*cos
    (wt) mV')
R=10; L=2*10^-3; C=200*10^-9; w=48*10^3;
vsamp=100;
w0=1/sqrt(L*C)
printf("w0= %3.1f krad/s \n",w0*10^-3);
Q0=w0*L/R
printf("Q0=%d \n",Q0)
B=w0/Q0
```

```
printf("Bandwidth= %3.1f krad/s \n",B*10^-3);
//Number of half bandwidths be N
N=2*(w-w0)/B
disp(N)
//Impedance Z(s)=(1+i*N)*R
//Finding the magnitude and angle
magZ=sqrt(1+N^2)*R
angZ=atan(N)*(180/%pi)
disp(angZ,'angZ=')
printf("Equivalent impedance value=%3.2f ohm \n", magZ)
//Approx current magnitude is
Iamp=vsamp/magZ
printf("\n Approx current magnitude= %3.2f mA \n", Iamp);
```

Scilab code Exa 16.4 `Other resonant forms`

```
clc
//Example 16.4
disp('Given')
disp('R1=2 ohm R2=3 ohm L=1H C=125mF')
R1=2;R2=3 ; L=1;C=125*10^-3;
w0=sqrt(1/(L*C)-(R1/L)^2)
printf("w0=%d rad/s \n",w0)
//Input admittance is 1/R2+i*w*C+1/(R+I*w*L)
Y=1/3+%i/4+1/(2+%i*2)
printf("Y= %3.4f S \n",Y)
//Now input impedance at resonance
Z=1/Y
printf("Z= %3.4f ohm \n",Z)
//Resonant frequency f=1/sqrt(L*C)
f=1/sqrt(L*C)
printf("f=%3.2f rad/s \n",f);
```

Scilab code Exa 16.5 Equivalent Series and parallel combination

```
clc
//Example 16.5
disp('Given')
disp('R=5 ohm L=100mH w=100 rad/s')
Rs=5; Ls=100*10^-3 ;w=100;
//Let Xs be the capacitive and inductive reactance
Xs=w*Ls
Q=Xs/Rs
//As Q is greater than 5 we can approximate as
Rp=Q^2*Rs
Lp=Ls
printf("The parallel equivalent is \n");
printf("Rp= %d ohm \t Lp=%d mH",Rp,Lp*10^3);
```

Scilab code Exa 16.6 Scaling

```
clc
//Example 16.6
disp('Given')
disp('Km=20 Kf=50')
Km=20; Kf=50;
s=poly(0,'s')
//From figure 16.20(a)
C=0.05; L=0.5;
//Performing magnitude as well as frequency scaling
    simultaneously
Cscaled =C/(Km*Kf)
Lscaled = L*Km/Kf
printf("Scaled values are \n")
```

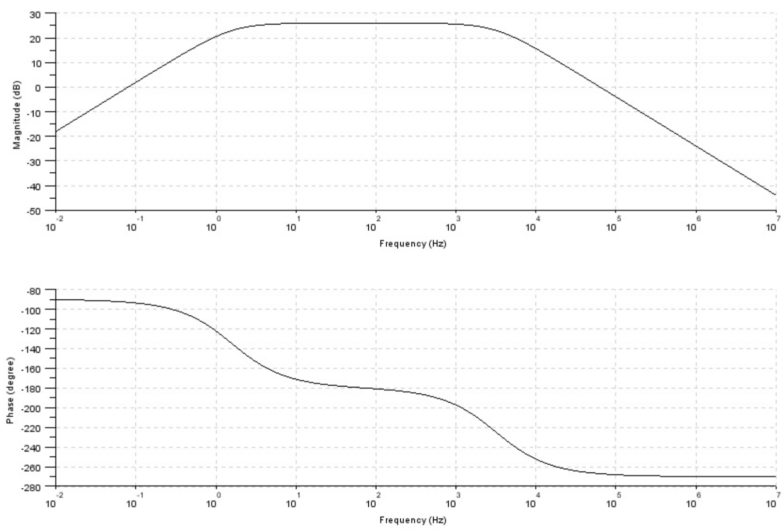

Figure 16.1: Bode diagrams

```
printf("Cscaled =%d uF \t Lscaled =%d mH \n",Cscaled
    *10^6,Lscaled*10^3)
//Converting the Laplace transform of the circuit
//From figure 16.20(c)
disp('Vin=V1+0.5s*(1-0.2*V1)')
disp('V1=20/s')
//On substituting V1 in equation of Vin

Zin=(s^2-4*s+40)/(2*s)
disp(Zin,'Zin=')
//Now we need to scale Zin
//We will multiply Zin by Km and replace s by s/Kf
Zinscaled=horner(Km*Zin,s/Kf)
disp(Zinscaled,'Zinscaled')
```

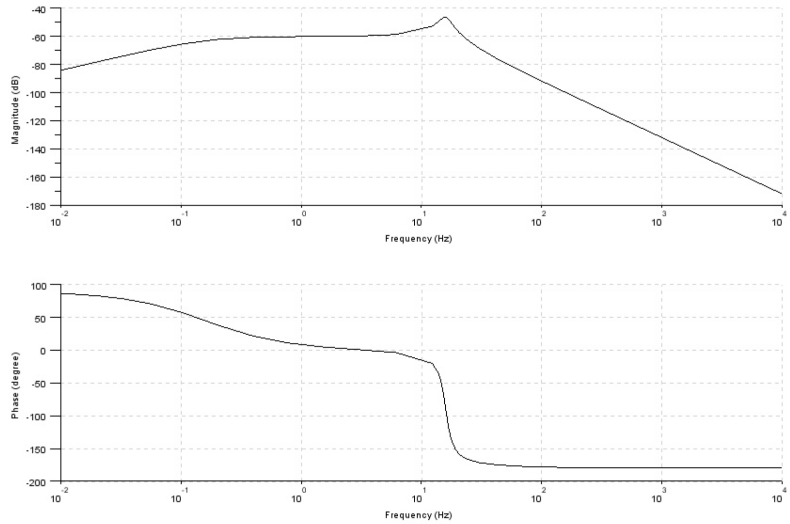

Figure 16.2: Bode diagrams

Scilab code Exa 16.8 `Bode diagrams`

```
clc
//Example 16.8
//From figure 16.26
disp('Writing the expression for voltage gain')
disp('Vout/Vin=4000*(-1/200)*(5000*10^8/s)
    /((5000+10^8/s)*(5000+10^6/20s))')
//On simplification
s=poly(0,'s')
h=syslin('c',(-2*s)/((1+s/10)*(1+s/20000)))
disp(h)
fmin=0.01
fmax=10^7
scf(1);clf;
bode(h,fmin,fmax)
```

Scilab code Exa 16.10 Bode diagrams

```
clc
//Example 16.10
s=poly(0,'s')
h=syslin('c',(10*s)/((1+s)*(s^2+20*s+10000)))
disp(h)
fmin=0.01
fmax=10^4
scf(1);clf;
//Calculate Bode plot
bode(h,fmin,fmax)
```

Scilab code Exa 16.11 Filters

```
clc
//Example 16.11
disp('Given')
disp('A high pass filter with cutoff frequency of 3k
    Hz')
//Cutoff frequency(wc)=1/(R*C)
//Let us select some standard value of resistor
disp('Let R=4.7k ohm')
fc=3*10^3;R=4.7*10^3;
wc=2*%pi*fc
C=1/(R*wc)
printf("\n C=%3.2f nF ",C*10^9);
s=poly(0,'s')
h=syslin('c',(R*C*s)/((1+s*R*C)))
disp(h)
HW = frmag(h,512);
w=0: %pi /511: %pi ;
```

```
plot(w,HW)
```

Scilab code Exa 16.12 Filters

```
clc
//Example 16.12
disp('Given')
disp('Bandwidth = 1M Hz and high frequency cutoff =
   1.1M Hz')
B=10^6;fH=1.1*10^6
//B=fH-fL
fL=fH-B
printf("Low frequency cutoff fL= %d kHz \n",fL
   *10^-3);
wL=2*%pi*fL
printf("wL= %3.2f krad/s \n",wL*10^-3);
wH=2*%pi*fH
printf("wH= %3.3f Mrad/s \n",wH*10^-6);
//Now we need to find values for R,L and C
//Let X=1/LC
B=2*%pi*(fH-fL)
X=(wH-B/2)^2-(B^2/4)
disp(X)
disp('Let L=1H')
L=1;
C=1/(L*X)
disp(C,'C=')
//B=R/L
R=L*B
printf("R= %3.3f Mohm \n",R*10^-6);
```

Scilab code Exa 16.13 Filters

```
clc
//Example 16.13
disp('Given')
disp('Voltage gain = 40dB and cutoff frequency = 10k
    Hz')
Av_dB=40
Av=10^(Av_dB/20)
f=10*10^3
B=2*%pi*f
//From figure 16.41(a)
disp('1+Rf/R1=100(Gain)')
//From figure 16.41(b)
//The transfer function is
disp('V+= Vi*(1/sC)/(1+1/sC)')
//Combining two transfer functions
disp('V0 = Vi*(1/sC)/(1+1/sC)*(1+Rf/R1)')
//The maximum value of the combined transfer
   function is
disp('Maximum value is V0 = Vi*(1+Rf/R1)')
disp('Let R1=1k ohm')
R1=10^3
Rf=(Av-1)*R1
printf("Rf= %d kohm \n",Rf*10^-3);
disp('C=1 uF')
C=10^-6
//B=1/(R2*C)
R2=1/(C*B)
printf("R2= %3.2f ohm \n",R2);
```

Chapter 17

Two Port Networks

Scilab code Exa 17.1 `One port networks`

```
//Example 17.1
clc
//From figure 17.3
disp('The mesh equations are')
disp('V1=10*I1-10*I2')
disp('0=-10*I1+17*I2-2*I3-5*I4')
disp('0=-2*I2+7*I3-I4')
disp('0=-5*I2-I3+26*I4')
//We need to find input impedance
disp('Zin=delz/del11')
//In matrix form
A=[10 -10 0 0 ;-10 17 -2 -5; 0 -2 7 -1;0 -5 -1 26]
delz=det(A)
printf("\n delz=%f ohm^4",delz);
//Eliminating first row and first column to find del11
B=[17 -2 -5;-2 7 -1;-5 -1 26]
del11=det(B)
printf("\n del11=%f ohm^3",del11);
Zin=delz/del11
printf("\n Zin=%f ohm",Zin);
```

Scilab code Exa 17.2 One port networks

```
//Example 17.2
clc
//From figure 17.5
disp('The mesh equations are')
disp('V1=10*I1-10*I2')
disp('0=-10*I1+17*I2-2*I3-5*I4')
disp('0=-2*I2+7*I3-I4')
disp('0=-0.5*I3+1.5*I4')
//We need to find input impedance
disp('Zin=delz/del11')
//In matrix form
A=[10 -10 0 0 ;-10 17 -2 -5; 0 -2 7 -1;0 0 -0.5 1.5]
delz=det(A)
printf("\n delz=%f ohm^3",delz);
//Eliminating first row and first column to find
    del11
B=[17 -2 -5;-2 7 -1;0 -0.5 1.5]
del11=det(B)
printf("\n del11=%f ohm^2",del11);
Zin=delz/del11
printf("\n Zin=%f ohm",Zin);
```

Scilab code Exa 17.3 One port networks

```
//Example 17.3
clc
//From figure 17.7
disp('The nodal equations are')
disp('I1=0.35*V1-0.2*V2-0.05*V3')
```

```
disp('I2=-0.2*V1+1.7*V2-1*V3')
disp('I3=-0.05*V1-1*V2+1.3*I3')
//We need to find input impedance
disp('Yin=dely/del11')
disp('Zin=1/Yin')
//In matrix form
A=[0.35 -0.2 -0.05;-0.2 1.7 -1;-0.05 -1 1.3]
dely=det(A)
printf("\n dely=%f S^3",dely);
//Eliminating first row and first column to find
    del11
B=[1.7 -1;-1 1.3]
del11=det(B)
printf("\n del11=%f S^2",del11);
Yin=dely/del11
printf("\n Yin=%f S",Yin);
Zin=1/Yin
printf("\n Zin=%f ohm",Zin);
```

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 17.7 `Some equivalent networks`

```
clc
//Example 17.7
//From figure 17.16
disp('Given a linear model of a transistor we need
    not explicitly find the aadmittance parameters ')
disp('-y12 corresponds to admittance of 2k ohm
    resistor')
```

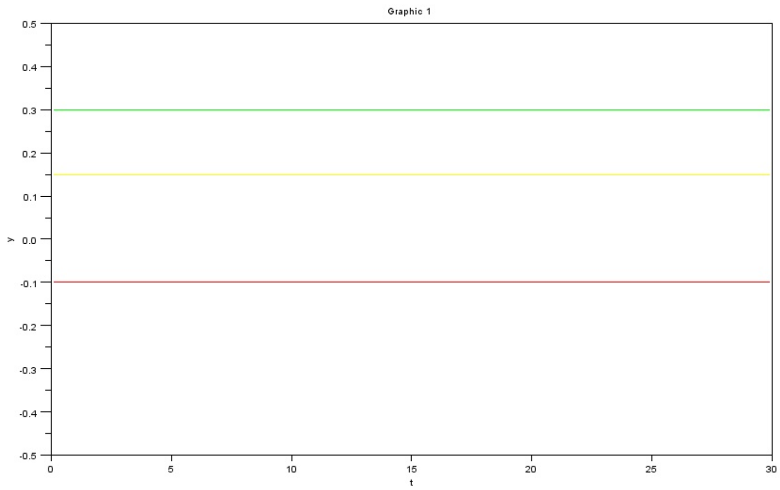

Figure 17.1: Admittance parameters

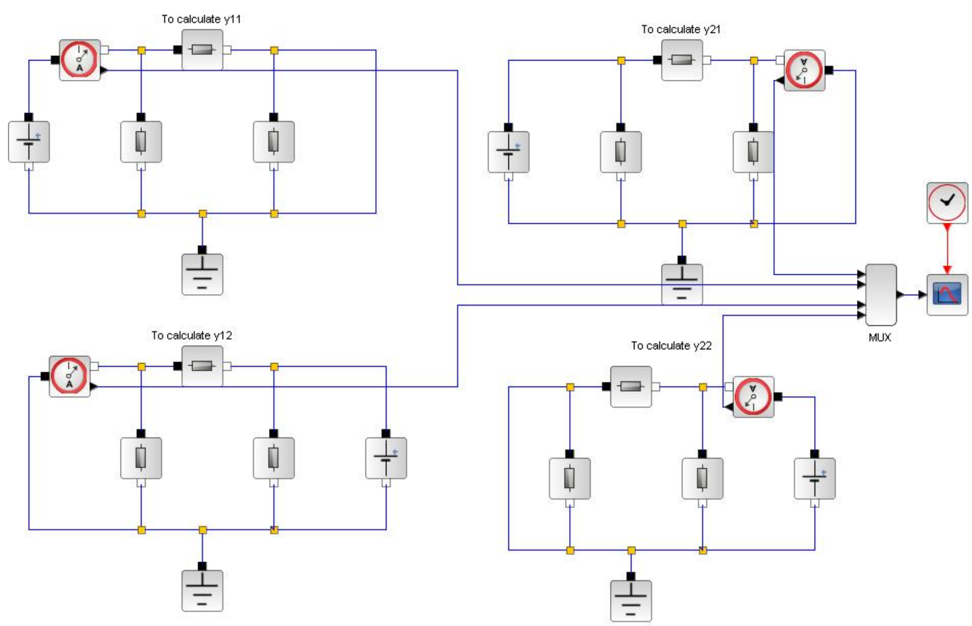

Figure 17.2: Admittance parameters

```
disp('y11+y12 corresponds to admittance of 500 ohm
    resistor')
disp('y21-y12 correponds to gain of dependent
    voltage source')
disp('y22+y12 corresponds to admittance of 10k ohm
    resistor')
//Writing down in equation form
y12=-1/2000
y11=1/500-y12
y21=0.0395+y12
y22=1/10000-y12
printf("\n y11= %3.2f mS \n y12= %3.2f mS \n y21= %3
    .2f mS \n y22= %3.2f mS",y11*10^3,y12*10^3,y21
    *10^3,y22*10^3);
```

Scilab code Exa 17.8 Impedance parameters

```
clc
//Example 17.8
Vs = poly(0,'Vs')
disp('Given')
disp('Z=[10^3 10;-10^6 10^4 ]')
z11=10^3 ; z12=10;z21=-10^6;z22=10^4
//Using the given matrix we can write the mesh
    equations as
disp('V1=10^3*I1+10*I2')
disp('V2=-10^6*I1+10^4*I2')
//The input to an two port network is an ideal
    sinusoidal voltage source in series with 500 ohm
//Mathematically
disp('The characterizing equations are')
disp('Vs=500*I1+V1')
//The output to an two port network is a 10k ohm
    resistor
//Mathematically
```

```
disp('V2=-10^4*I2')
Zg=500;
//Expressing V1,V2,I1,I2 in terms of Vs
V1=0.75*Vs
I1=Vs/2000
V2=-250*Vs
I2=Vs/40
disp('Voltage gain Gv=V2/V1')
Gv=V2/V1
disp(Gv,'Gv=')
disp('Current gain Gi=I2/I1')
Gi=I2/I1
disp(Gi,'Gi=')
disp('Power gain Gp=Real[-0.5*V2*I2*]/Real[0..5*V1*
    I1*]')
Gp=(-0.5*V2*I2)/(0.5*V1*I1)
disp(Gp,'Gp=')
disp('Input impedance is Zin=V1/I1')
Zin=V1/I1
disp('Output impedance is Zout=z22-((z12*z21)/(z11+
    Zg))')
Zout=z22-((z12*z21)/(z11+Zg))
printf("\n Zout= %3.2f kohm",Zout*10^-3)
```

Scilab code Exa 17.9 Hybrid parameters

```
clc
//Example 17.9
//From figure 17.27
//Writing the mesh equations
disp('V1=5*I1+4*I2')
disp('V2=4*I1+10*I2')
//Arranging in the standard form
//V1=h11*I1+h12*V2
//I2=h21*I1+h22*V2
```

```
//Therefore h parameters are
h11=3.4;h12=0.4;h21=-0.4;h22=0.1;
h=[h11 h12;h21 h22]
disp(h)
```

Scilab code Exa 17.10 `Transmission parameters`

```
clc
//Example 17.10
//From figure 17.32
disp('Consider Network A')
//Writing the mesh equations
disp('V1=12*I1+10*I2')
disp('V2=10*I1+14*I2')
//Arranging in the standard form
//V1=t11*V2-t12*I2
//I1=t21*V2-t22*I2
//Therefore t parameters of Network A is
t11A=1.2;t12A=6.8;t21A=0.1;t22A=1.4;
disp('Consider Network B')
//Writing the mesh equations
disp('V1=24*I1+20*I2')
disp('V2=20*I1+28*I2')
//Arranging in the standard form
//V1=t11*V2-t12*I2
//I1=t21*V2-t22*I2
//Therefore t parameters of Network B is
t11B=1.2;t12B=13.6;t21B=0.05;t22B=1.4;
tA=[1.2 6.8;0.1 1.4]
tB=[1.2 13.6;0.05 1.4]
disp('t parameters of cascaded network is t=tA*tB')
t=tA*tB
disp(t)
```

Chapter 18

Fourier Circuit Analysis

Scilab code Exa 18.1 Trigonometric form of the Fourier Series

```
clear
close
clc
//Example 18.1
//From the figure 18.2
disp('The equation of v(t) considering one period can be written as')
disp('v(t)=Vm*cos(5*%pi*t) for -0.1&lt;=t&lt;=0.1')
disp('v(t)=0 for 0.1&lt;=t&lt;=0.3')
//Assuming the value of Vm is 1

//Evaluating the constants an and bn
//bn=0 for all n
//an=(2*Vm*cos(n*%pi/2))/(%pi*(1-n^2))
//a0=Vm/%pi
t=-1:0.02:1;
Vm=ones(1,length(t));
v0t=Vm/%pi;
v1t=(Vm.*cos(5*%pi*t)).*0.5;
```

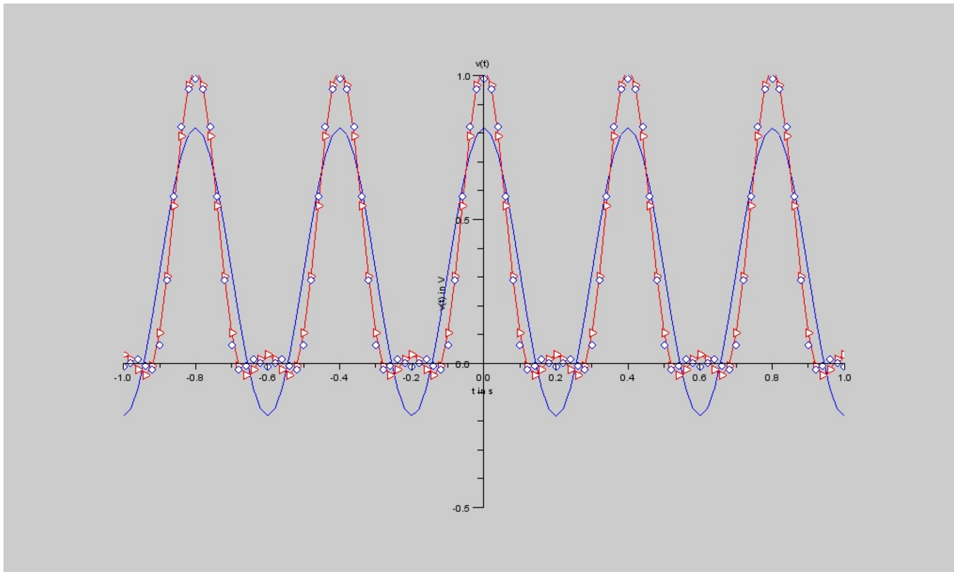

Figure 18.1: Trigonometric form of the Fourier Series

```
v0t_v1t=v0t+v1t;
v2t=(2/(3*%pi))*(Vm.*cos(10*%pi*t));
v0t_v1t_v2t=v0t+v1t+v2t;
v3t=(2/(15*%pi))*(Vm.*cos(20*%pi*t));
v0t_v1t_v2t_v3t=v0t+v1t+v2t-v3t;
figure
a = gca ();
a. y_location = "origin";
a. x_location = "origin";
a. data_bounds =[ -1,0;1 0.5];
plot (t,v0t)
xtitle('vot vs t','t in s','vot')
figure
a = gca ();
a. y_location = "origin";
a. x_location = "origin";
a. data_bounds =[ -1,-0.5;1 0.5];
plot (t,v0t_v1t)
a. y_location = "origin";
```

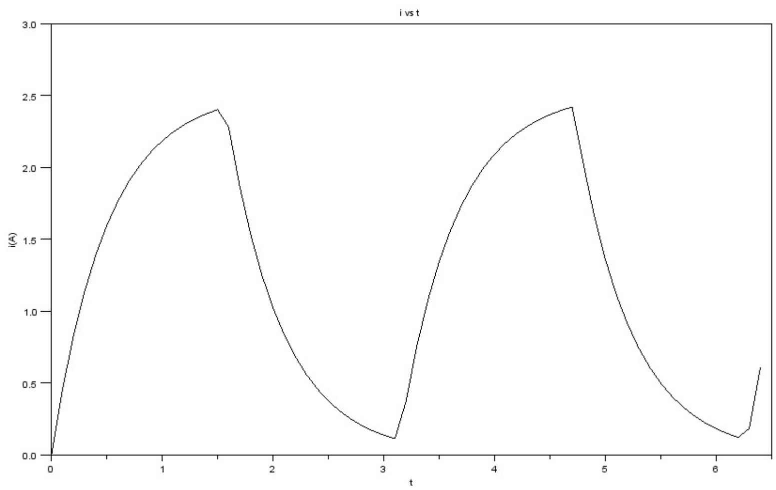

Figure 18.2: Complete Response to periodic Forcing Functions

```
a. x_location = "origin";
a. data_bounds =[ -1,-0.5;1 0.5];
plot (t,v0t_v1t_v2t,'r.->')
a. y_location = "origin";
a. x_location = "origin";
a. data_bounds =[ -1,-0.5;1 0.5];
plot (t,v0t_v1t_v2t_v3t,'d')
xtitle('v(t)','t in s','v(t) in V')
```

This code can be downloaded from the website wwww.scilab.in

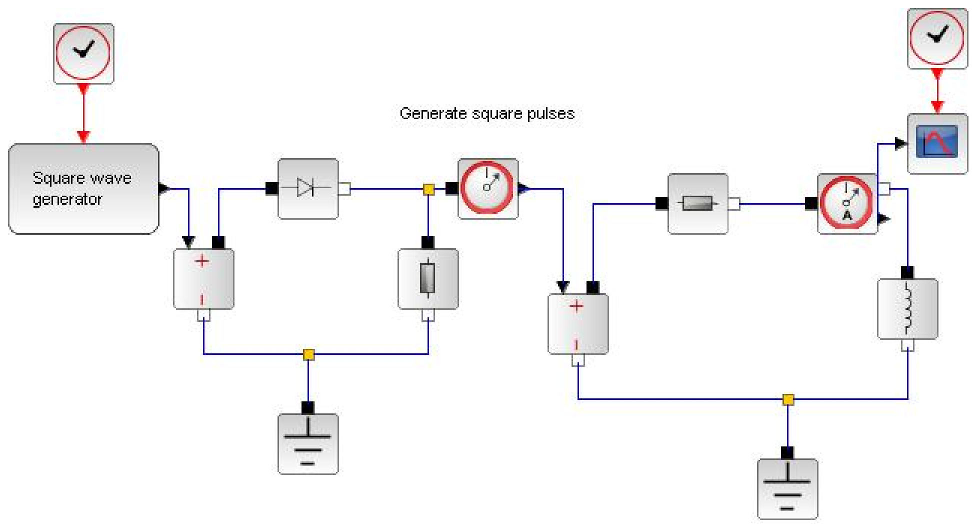

Figure 18.3: Complete Response to periodic Forcing Functions

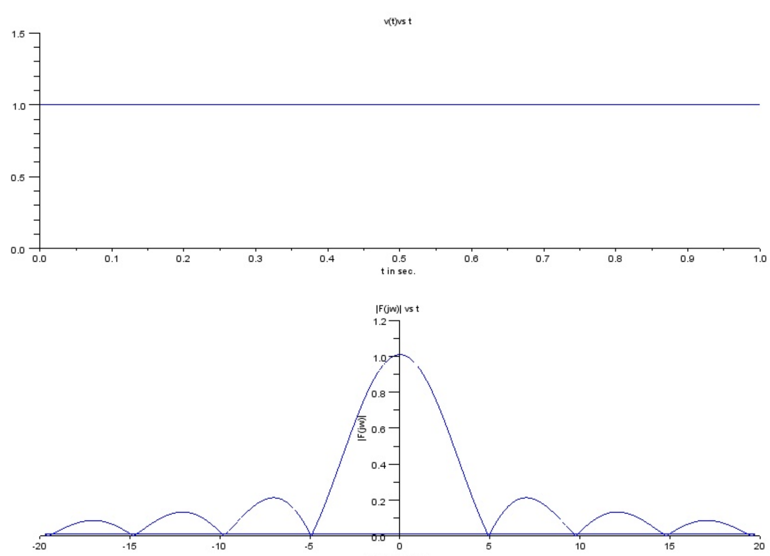

Figure 18.4: Definition of the Fourier Transform

Scilab code Exa 18.5 Definition of the Fourier Transform

```
clc;
//Example 18.5
//Let amplitude be 1
A=1;
Dt=0.01;
T1=4;
t=0:Dt:T1/4;
for i=1:length(t)
    xt(i)=A
end
//Calculate Fourier Transform
Wmax=2*%pi*1;
K=4;
k=-(2*K):(K/1000):(2*K);
W=k*Wmax/K;
xt=xt';
XW=xt*exp(-sqrt(-1)*t'*W)*Dt;
XW_Mag=real(XW);
W=[-mtlb_fliplr(W),W(2:1001)];
XW_Mag=[mtlb_fliplr(XW_Mag),XW_Mag(2:1001)];
subplot(2,1,1);
a=gca();
a.data_bounds=[0,0;1,1.5];
a.y_location="origin";
plot(t,xt);
xlabel('t in sec.');
title('v(t)vs t');
subplot(2,1,2);
a=gca();
a.y_location="origin";
plot(W*%pi/2,abs (XW_Mag));
xlabel('Freq in rad/sec');
ylabel('|F(jw)|')
title('|F(jw)| vs t');
```

Scilab code Exa 18.6 Physical significance of Fourier Transform

```
clc
syms s t
printf("Given")
disp('v(t)=4*exp(-3*t)*u(t)')
v=4*exp(-3*t)

F=4*(integ(exp(-(3+%i*1)*s),s,0,%inf))
//The secind term tends to zero
disp(F,'F=')
//Let W be the total 1 ohm energy in the input signal
W=integ(v^2,t,0,%inf)
disp(W,'W=')
//Let Wo be the total energy
//As the frequency range is given as 1 Hz<|f|<2 Hz
//Considering symmetry
Wo=(1/%pi)*integ((16/(9+s^2)),s,2*%pi,4*%pi)
disp(Wo,'Wo=')
```

This code can be downloaded from the website wwww.scilab.in

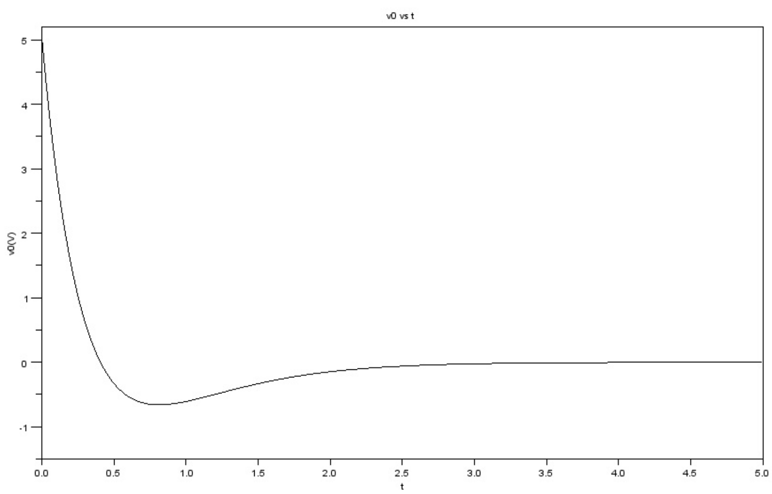

Figure 18.5: The physical significance of system function

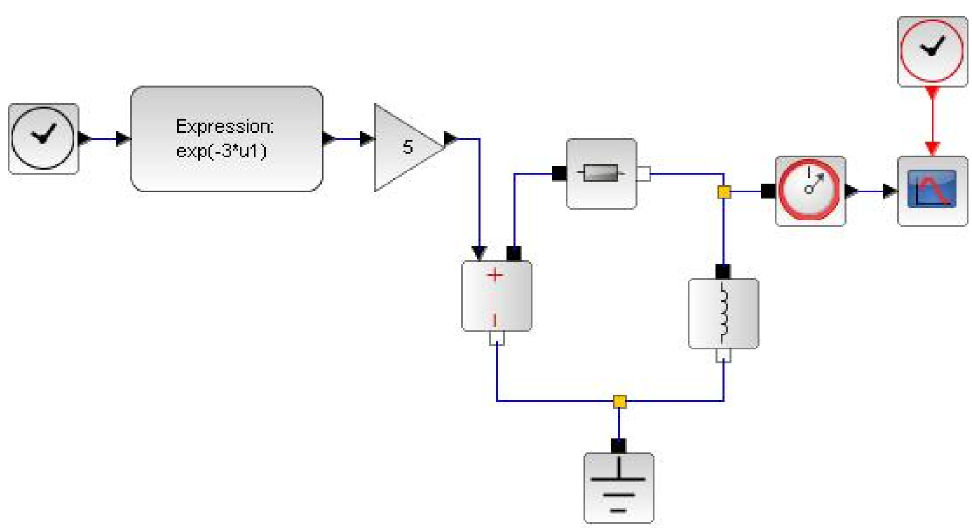

Figure 18.6: The physical significance of system function