Scilab Textbook Companion for Electric Circuits by M. Navhi And J. A. Edminister[1]

Created by
Dhaivat Udayan Mandavia
B.E. (pursuing)
Electronics Engineering
B. V. B. C. E. T., Hubli
College Teacher
Prof. Sujata Kotabagi, B. V. B. C. E. T., Hubli
Cross-Checked by
sonanya tatikola, IITB

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: Electric Circuits

Author: M. Navhi And J. A. Edminister

Publisher: Tata McGraw - Hill Publishing Co. Ltd., New Delhi

Edition: 4

Year: 2007

ISBN: 0070601739

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 4

1 Introduction 6

2 Circuit Concepts 9

3 Circuit Laws 17

5 Analysis Methods 22

6 Amplifiers and Operational Amplifiers 27

7 Waveforms and Signals 33

8 First order Circuits 45

9 Higher order circuits and Complex frequency 57

10 Sinusoidal Steady state Circuit Analysis 66

11 AC Power 70

12 Polyphase Circuits 84

13 Frequency Response Filters and Resonance 90

14 Two Port Networks **93**

15 Mutual Inductance and Transformers **100**

17 The Laplace Transform Method **105**

List of Scilab Codes

Exa 1.1	Force Work and Power	6
Exa 1.2	Electric Charge and Current	7
Exa 1.3	Electric Potential	7
Exa 1.4	Energy and Electrical Power	8
Exa 2.1	Resistance	9
Exa 2.2	Inductance	10
Exa 2.3	Capacitance	12
Exa 2.4	Nonlinear Resistors	14
Exa 2.5	Nonlinear Resistors	15
Exa 3.3	Circuit elements in series	17
Exa 3.4	Circuit elements in series	17
Exa 3.5	Circuit elements in parallel	18
Exa 3.6	Circuit elements in parallel	19
Exa 3.7	Voltage division	19
Exa 3.8	Current division	20
Exa 5.8	Thevenin and Norton theorem	22
Exa 6.8	Operational amplifiers	27
Exa 6.10	Summing circuit	28
Exa 7.1	Sinusoidal Functions	33
Exa 7.2	Sinusoidal Functions	35
Exa 7.3	Time Shift and Phase Shift	36
Exa 7.5	Combinations of Periodic Functions	37
Exa 7.13	The Average and Effective RMS values	38
Exa 7.17	The Unit Impulse Function	38
Exa 7.22	The Exponential Function	40
Exa 7.23	The Exponential Function	40
Exa 7.25	Random Signals	42
Exa 7.26	Random Signals	43

Exa 8.1	Capacitor Discharge in a Resistor	45
Exa 8.10	Transitions at Switching Time	54
Exa 9.6	Generalized Impedance RLC in s domain	58
Exa 9.7	Generalized Impedance RLC in s domain	60
Exa 9.8	Network function and Pole zero plots	64
Exa 9.9	The Forced Response	65
Exa 9.11	The Natural Response	65
Exa 10.4	Phasors	66
Exa 10.5	Impedance and Admittance	67
Exa 10.7	Superposition of AC sources	68
Exa 11.1	Power in time domain	70
Exa 11.2	Power in time domain	71
Exa 11.4	Average or Real Power	73
Exa 11.5	Reactive Power	74
Exa 11.10	Complex power Apparent power and Power triangle	74
Exa 11.11	Complex power Apparent power and Power triangle	76
Exa 11.12	Parallel connected Networks	77
Exa 11.13	Power factor improvement	78
Exa 11.14	Power factor improvement	78
Exa 11.16	Maximum power transfer	79
Exa 11.17	Superposition of average powers	81
Exa 12.2	Balanced Delta connected load	84
Exa 12.5	Unbalanced Delta connected load	86
Exa 12.6	Unbalanced Wye connected load	87
Exa 13.2	High pass and Low pass networks	90
Exa 13.7	Bandpass filters and Resonance	91
Exa 13.8	Bandpass filters and Resonance	92
Exa 14.1	Z parameters	93
Exa 14.2	Reciprocal and non reciprocal networks	94
Exa 14.4	Y parameters	95
Exa 14.6	Conversion between Z and Y parameters	96
Exa 14.7	H parameters	96
Exa 14.8	g parameters	97
Exa 14.10	Choice of parameter type	98
Exa 15.4	Energy in a pair of coupled coils	100
Exa 15.7	Ampere Turn Dot rule	101

Exa 15.8	Reflected Impedance	102
Exa 15.9	Reflected Impedance	103
Exa 17.2	Convergence of the integral	105
Exa 17.4	Partial Fractions Expansions	105

List of Figures

2.1 Resistance 11
2.2 Inductance 13
2.3 Capacitance 15

5.1 The branch current method 23
5.2 The branch current method 24
5.3 Network reduction 25
5.4 Network reduction 25
5.5 Superposition 26
5.6 Superposition 26

6.1 Analysis of circuits containing ideal op amps 28
6.2 Analysis of circuits containing ideal op amps 29
6.3 Noninverting circuit 30
6.4 Noninverting circuit 30
6.5 Circuits containing several Op amps 31
6.6 Circuits containing several Op amps 32

7.1 Sinusoidal Functions 35
7.2 Sinusoidal Functions 36
7.3 Time Shift and Phase Shift 37
7.4 The Exponential Function 41
7.5 The Exponential Function 42
7.6 The Exponential Function 43

8.1 Capacitor Discharge in a Resistor 46
8.2 Capacitor Discharge in a Resistor 47
8.3 Establishing a DC Voltage across a Capacitor 48
8.4 Establishing a DC Voltage across a Capacitor 48
8.5 The Source free RL Circuit 49
8.6 The Source free RL Circuit 49
8.7 Complex first order RL and RC Circuits 50
8.8 Complex first order RL and RC Circuits 50
8.9 Complex first order RL and RC Circuits 51
8.10 Complex first order RL and RC Circuits 52
8.11 DC Steady state in Inductors and Capacitors 52
8.12 DC Steady state in Inductors and Capacitors 53
8.13 DC Steady state in Inductors and Capacitors 53
8.14 DC Steady state in Inductors and Capacitors 54
8.15 Transitions at Switching Time 55
8.16 Transitions at Switching Time 55

9.1 Series RLC Circuit 58
9.2 Series RLC Circuit 59
9.3 Series RLC Circuit 60
9.4 Series RLC Circuit 61
9.5 Parallel RLC circuit 62
9.6 Parallel RLC circuit 63

11.1 Power in time domain 71
11.2 Power in time domain 72
11.3 Power in time domain 73

Chapter 1

Introduction

Scilab code Exa 1.1 Force Work and Power

```
clc
disp("Example 1.1")
printf("\n")
printf("Given")
disp("Acceleration is 2.0m/s^2")
disp("Mass is 10kg")
m=10;a=2;
disp("a)")
disp("F=m*a")
F=m*a
printf("Force is %dN\n",F)
disp("b)")
printf("time=4s\n")
t=4;
x=(a*t*t)/2
KE=(F*x)
P=KE/t
printf("Position is %dm\n",x)
printf("Kinetic energy =%3.1fJ\n",KE)
printf("Power =%3.1fW\n",P)
```

Scilab code Exa 1.2 Electric Charge and Current

```
clc
disp("Example 1.2")
printf("\n")
printf("Given")
disp("Current flow is 5A")
disp("Time is 1 minute")
i=5;
//As electroms/min is asked so we need to convert A(C/s) to C/min
i1=5*60;
//Let e be electronic charge
e=1.602*10^-19
n=(i1/e)
printf("Number of electrons =%3.2f electrons/min\n",n)
```

Scilab code Exa 1.3 Electric Potential

```
clc
disp("Example 1.3")
printf("\n")
printf("Given")
disp("Energy is 9.25uJ")
disp("Charge to be transferred is 0.5uC")
E=9.25*10^-6;q=0.5*10^-6;
//1 volt is 1 joule per coulomb
V=E/q;
printf("Potential difference between two points a and b is %3.1fV\n",V)
```

Scilab code Exa 1.4 Energy and Electrical Power

```
clc
disp("Example 1.4")
printf("\n")
printf("Given")
disp("Potential difference is 50V")
disp("Charge per minute is 120C/min")
V=50;x=120;
//As Electrical energy is to be calculated charge
   per minute is to be converted in charge per
   second
//Charge per second is nothing but the current
i=x/60;
P=i*V;
//Since is 1W=1J/s
printf("Rate of energy conversion is %dJ/s\n",P)
```

Chapter 2

Circuit Concepts

Scilab code Exa 2.1 Resistance

```

disp("Example 2.1")
printf("\n")
printf("Given")
disp("Resistance used is 4 ohm")
disp("Current flow is i=2.5*sin(w*t)")
disp("Angular frequency(w)=500 rad/s")

R=4;
iamp=2.5;w=500;
t=0:0.001:0.012566
i=2.5*sin(w*t)


Vamp=iamp*R;
printf("v=%d*sin(%d*t)(V)\n",Vamp,w)

pamp=iamp*iamp*R;
printf("p=%d(sin(%d*t))^2(W)\n",pamp,w)
p=pamp*sin(w*t)^2;

```

```
//On integrating p with respect to t
W=25*(t/2-sin(2*w*t)/(4*w))

function p=f(t),p=pamp*sin(w*t)^2,endfunction
w1=intg(0,2*%pi/w,f);


subplot(221)
plot(t,i)
xtitle ('i vs wt','wt','i ');

subplot(222)
plot(t,p)
xtitle ('p vs wt','wt','p ');



subplot(223)
plot(t,W)
xtitle ('w vs wt','wt','w ');
```

Scilab code Exa 2.2 Inductance

```
clc
disp("Example 2.2")
printf("\n")
printf("Given")
disp("Inductance used is 30mH")
disp("Current flow is i=10*sin(50*t)")
L=30*10^-3;iamp=10;
t=0:0.01:0.06283;
i=10*sin(50*t)
//v=L*d/dt(i)
```

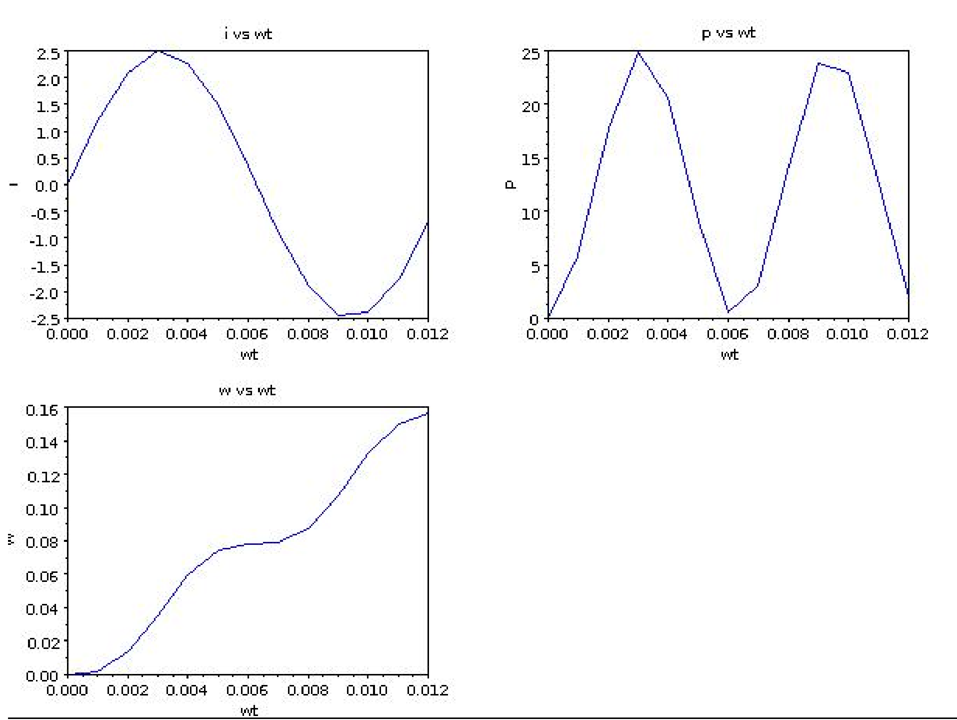

Figure 2.1: Resistance

```
//d/dt(sin 50t)=50*cos t
vamp=L*iamp*50;
v=vamp*cos(50*t)

//sinA*cosB=(sin(A+B)+sin(A-B))/2

pamp=vamp*iamp/2;
p=pamp*sin(100*t)
//On integrating 'p' w.r.t  t

wL=0.75*(1-cos(100*t));



subplot(221)
plot(t,i)
xtitle ('i vs wt','wt','i');

subplot(222)
plot(t,v)
xtitle ('v vs wt','wt','v ');

subplot(223)
plot(t,p)
xtitle ('p vs wt','wt','p ');

subplot(224)
plot(t,wL)
xtitle ('wL vs wt','wt','wL ');
```

Scilab code Exa 2.3 Capacitance

```
clc
disp("Example 2.3")
```

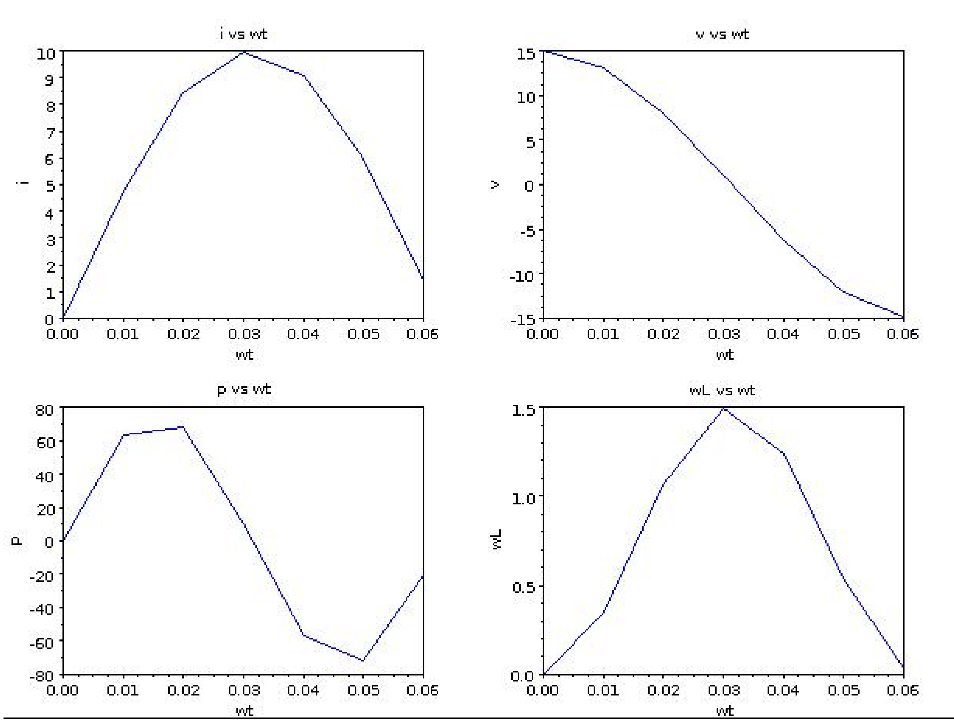

Figure 2.2: Inductance

```
printf("\n")
printf("Given")
disp("Capacitance used is 20uF")
disp("Voltage is v=50*sin(200*t)")
C=20*10^-6;
// Given that v=50*sin(200*t);
vamp=50;
t=0:0.001:0.015;
//q=C*v
qamp=vamp*C
q=qamp*sin(200*t)
//i=C*d/dt(v)
//d/dt(sin 200t)=200*cos t
iamp=C*vamp*200;
i=iamp*cos(200*t)

//sinA*cosB=(sin(A+B)+sin(A-B))/2

pamp=vamp*iamp/2;
p=pamp*sin(400*t)

//On integrating 'p' w.r.t  t

wC=12.5*(1-cos(400*t));

figure
a= gca ();
plot(t,wC)
xtitle ('wC vs wt','wt','wC (mJ)');
a. thickness = 2;
```

Scilab code Exa 2.4 Nonlinear Resistors

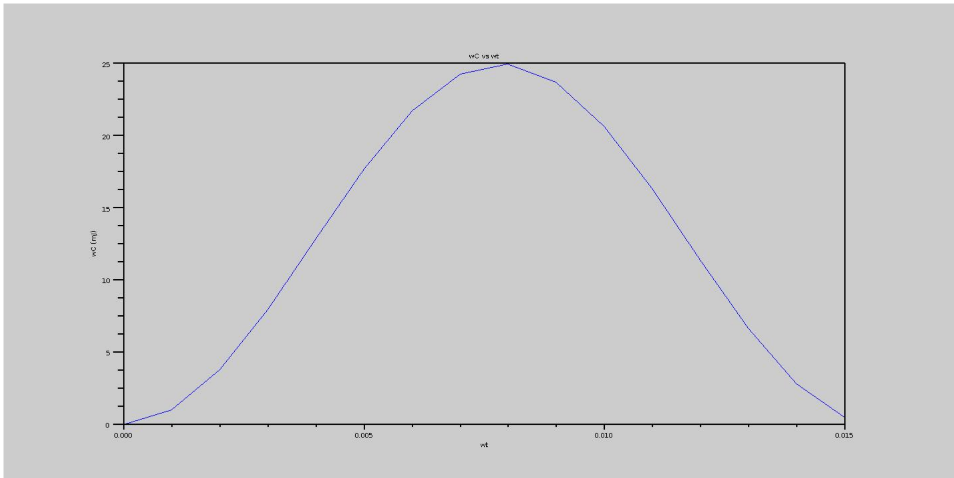

Figure 2.3: Capacitance

```
clc
disp("Example 2.4")
printf("\n")
printf("Given")
disp("Current through diode is 30mA")
//From the table the nearest value is at v=0.74V
V=0.74;I=28.7*10^-3;
R=V/I;
delV=0.75-0.73
delI=42.7*10^-3-19.2*10^-3
r=delV/delI
p=(V*I)*10^3
printf("\n \n Static resistance is %3.2fohm\n",R)
printf("Dynamic resistance is %3.2fohm\n",r)
printf("Power consumption is %3.2fmW\n",p)
```

Scilab code Exa 2.5 `Nonlinear Resistors`

```
clc
disp("Example 2.5")
```

```
printf("\n")
printf("Given")
disp("a)")
disp("Current through diode is 10mA")
//From the table the  value is at v=2.5V
V=2.5;I=10*10^-3;
R=V/I;
delV=3-2
delI=11*10^-3-9*10^-3
r=delV/delI
p=(V*I)*10^3
printf("\n \n Static resistance is %3.2fohm\n",R)
printf("Dynamic resistance is %3.2fohm\n",r)
printf("Power consumption is %3.2fmW\n",p)

disp("b)")
disp("Current through diode is 15mA")
//From the table the  value is at v=5V
V=5;I=15*10^-3;
R=V/I;
delV=5.5-4.5
delI=16*10^-3-14*10^-3
r=delV/delI
p=(V*I)*10^3
printf("\n \n Static resistance is %3.2fohm\n",R)
printf("Dynamic resistance is %3.2fohm\n",r)
printf("Power consumption is %3.2fmW\n",p)
```

Chapter 3

Circuit Laws

Scilab code Exa 3.3 `Circuit elements in series`

```
clc
disp("Example 3.3")
printf("\n")
printf("Given")
disp("Equivalent resistance of three resistors is
   750 ohm")
disp("values of two resistors are 40 ohm and 410 ohm
   ")
Req=750;R1=40;R2=410;

//For series resistance
disp("Req=R1+R2+R3")
//On solving for R3
R3=Req-R1-R2
printf("Value of third ohmic resistor is %dohm\n",R3
   )
```

Scilab code Exa 3.4 `Circuit elements in series`

```
clc
disp("Example 3.4")
printf("\n")
printf("Given")
disp("values of two capacitors are 2uF and 10uF")
C1=2*10^-6;C2=10*10^-6;
//For two capacitors in series
disp("Ceq=(C1*C2)/(C1+C2)")
//On solving for Ceq
Ceq=((C1*C2)/(C1+C2))*10^6
printf("Value of equivalent capacitance is %3.2fuF\n
   ",Ceq)

disp("If C2=10pF")
C2=10*10^-12;

Ceq=((C1*C2)/(C1+C2))*10^12
printf("Value of equivalent capacitance is %3.2fpF\n
   ",Ceq)
```

Scilab code Exa 3.5 Circuit elements in parallel

```
clc
disp("Example 3.5")
printf("\n")
printf("Given")
disp("a)")
disp("values of two resistors are 60 ohm and 60 ohm"
   )
R1=60;R2=60;
disp("If resistors are parallel")
Req=(R1*R2)/(R1+R2)
printf("Value of equivalent resistance is %dohm\n",
   Req)

```

```
disp("b)")
disp("values of three equal resistors are 60 ohm")
R1=60;R2=60;R3=60;
disp("If resistors are parallel")
x=1/R1+1/R2+1/R3
Req=1/x;
printf("Value of equivalent resistance is %dohm\n",
   Req)
```

Scilab code Exa 3.6 `Circuit elements in parallel`

```
clc
disp("Example 3.6")
printf("\n")
printf("Given")

disp("values of two inductors are 3mH and 6 mH")
L1=3*10^-3;L2=6*10^-3;
disp("If inductors are parallel")
Leq=((L1*L2)/(L1+L2))*10^3
printf("Value of equivalent inductance is %3.1fmH\n"
   ,Leq)
```

Scilab code Exa 3.7 `Voltage division`

```
clc
disp("Example 3.7")
printf("\n")
printf("Given")
disp("Total resistance of three resistors is 50 ohm"
   )
R=50;
```

```
disp("Output voltage is 10 percent of the input voltage")
//Let v be input voltage and v1 be output voltage
//Let v1/v=V
V=0.1;
//As V=R1/(Total resistance)
//Solving for R1
R1=V*R;
//As R=R1+R2
//Solving for R2
R2=R-R1;
printf("R1=%dohm\n R2=%dohm\n",R1,R2)
```

Scilab code Exa 3.8 Current division

```
clc
disp("Example 3.8")
printf("\n")
printf("Given")
disp("Total current is 30mA")
disp("Branch currents are 20mA and 10mA")
disp("Equivalent resistance is equal to or greater than 10 ohm")

//From Fig 3.6
//Current flowing through R1 be i1 and let it be equal to 10mA
//Current flowing through R2 be i2 and let it be equal to 20mA
i1=10*10^-3;i2=20*10^-3;
i=30*10^-3;

//Let R1/(R1+R2)=X1          (1)
//Let R2/(R1+R2)=X2          (2)
X1=i1/i;
```

```
X2=i2/i;
//Let R1*R2(R1+R2)=Y          (3)
//Given that
printf("\n Given")
disp("R1*R2(R1+R2)>=10")
//Solving (1),(2) and (3) we get
printf("R1>=%dohm\nR2>=%dohm\n",15,30)
```

Chapter 5

Analysis Methods

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

Scilab code Exa 5.8 `Thevenin and Norton theorem`

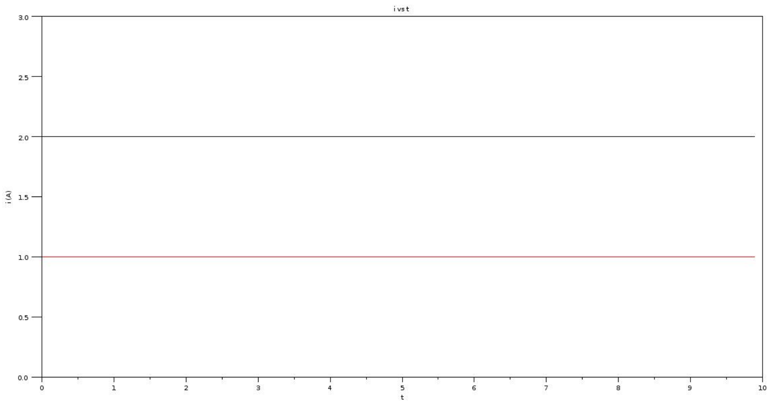

Figure 5.1: The branch current method

```
clc
//From figure 5.13(a)
//Applying KVL equation to the loop
I=(20+10)/(3+6)
//As current will not flow in upper 3 ohm resistor
    so Thevenin voltage is equal to either of the two
     parallel branches
V1=20-I*3
printf("Thevenin voltage = %dV\n",V1)

// Left 3 ohm and 6 ohm resistor are in parallel and
     their equivalent is in series with 3 ohm
R1=3+(3*6)/(3+6)
printf("Thevenin resistance =%dohm\n",R1)

//Now to find Norton's equivalent
I1=V1/R1
printf("\n Norton current =%dA\n",I1)
disp("The value of resistance in Norton equivalent
    will not change but will come in parallel with
    current source")
```

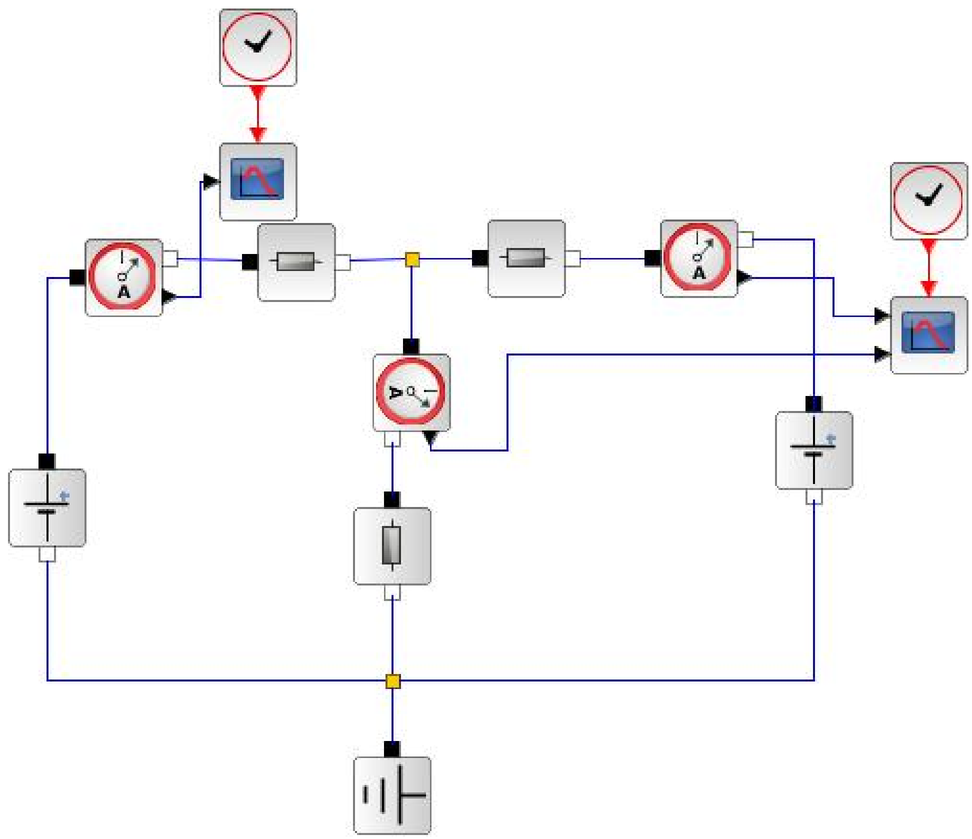

Figure 5.2: The branch current method

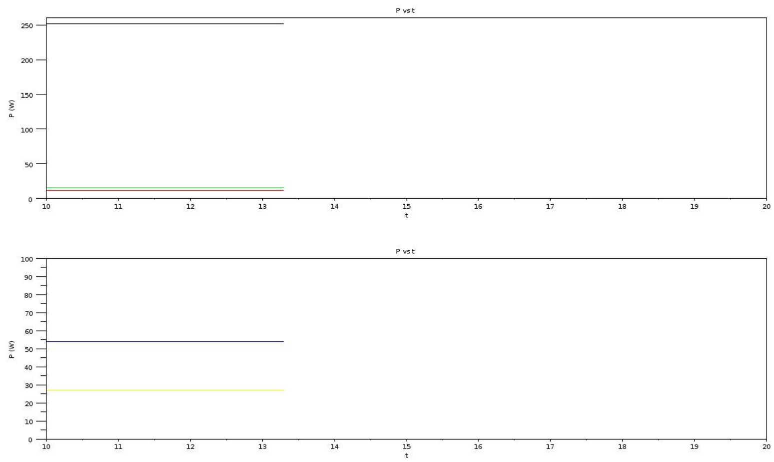

Figure 5.3: Network reduction

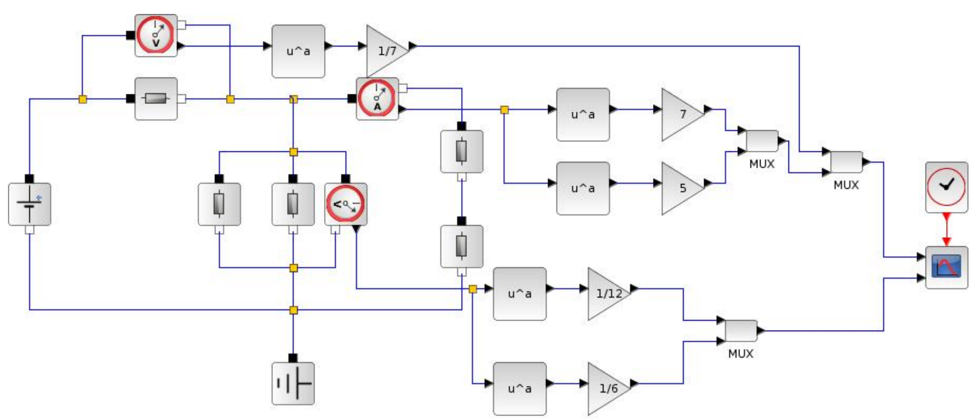

Figure 5.4: Network reduction

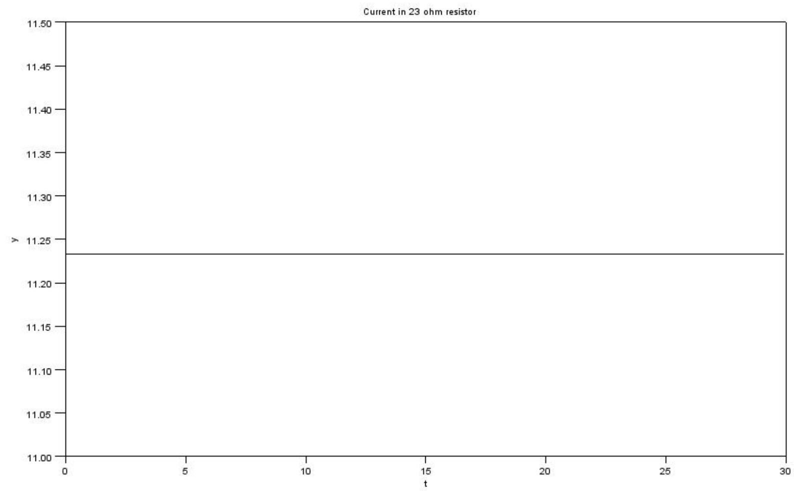

Figure 5.5: Superposition

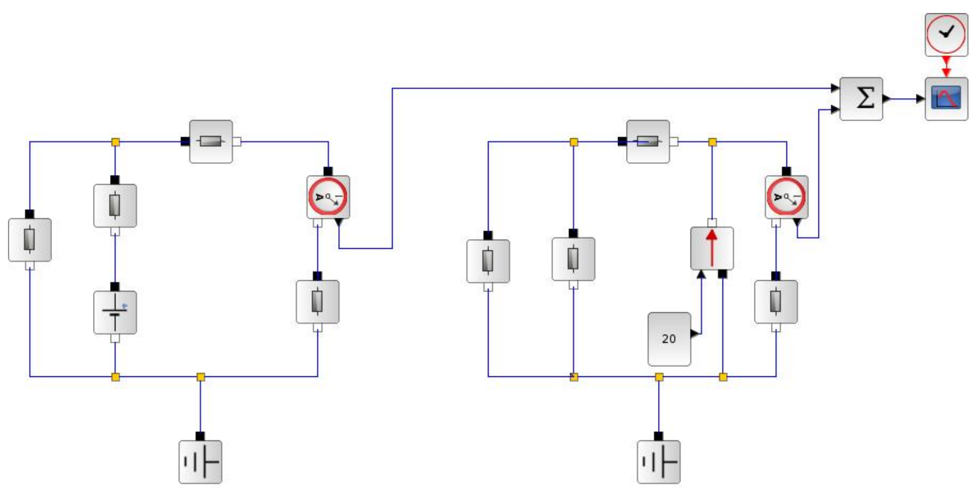

Figure 5.6: Superposition

Chapter 6

Amplifiers and Operational Amplifiers

Scilab code Exa 6.8 `Operational amplifiers`

```
clc
disp("Example 6.8")
printf("Given")
disp("R1= 10kohm R2=50kohm Ri=500kohm R0=0")
disp("Open loop gain (A)=10^5")
A=10^5;R1=10*10^3;R2=50*10^3;Ri=500*10^3;
//From figure 6.11
//Applying KCL equation at node B
disp("(v1+vd)/10+ (v2+vd)/50+ vd/500=0          (1)")
//Since R0=0
disp("v2=A*vd")
//Solving for vd
disp("vd=10^-5*v2        (2)")
//Substituting (2) in (1) we get
printf("v2/v1=%d\n",-5)
```

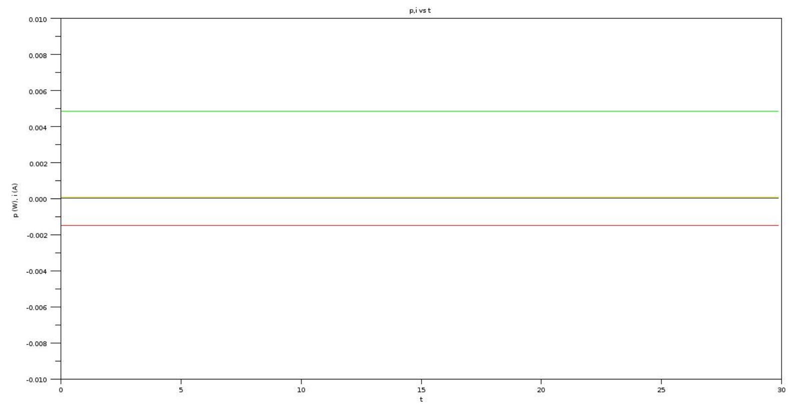

Figure 6.1: Analysis of circuits containing ideal op amps

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

Scilab code Exa 6.10 `Summing circuit`

```
clc
printf("Given")
disp("R1=1 ohm;R2=1/2 ohm;R3=1/4 ohm;R4=1/8 ohm")
disp("Rf=1 ohm")
//From figure 6.14
//THe output of summing circuit can be written as
disp("v0=-((Rf/R1)*v1+(Rf/R2)*v2+(Rf/R3)*v3+......")
//From above equation
disp("v0=-(8v4+4v3+2v2+v1)------------(1)")
disp("a)")
```

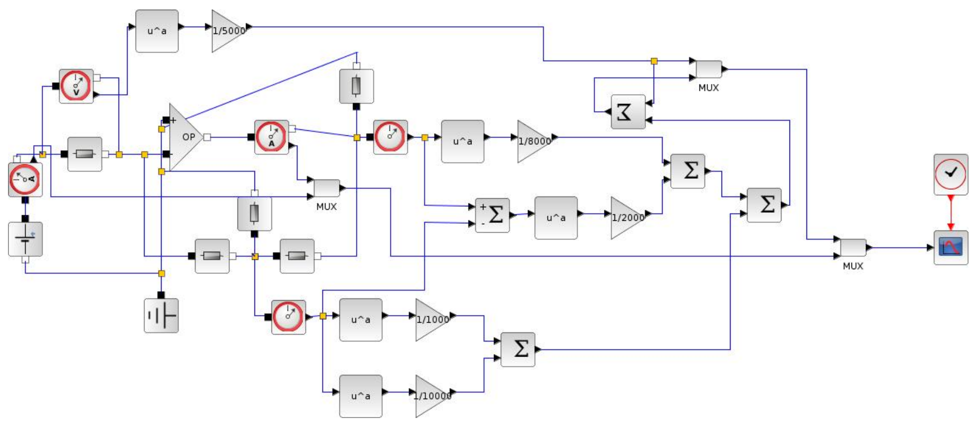

Figure 6.2: Analysis of circuits containing ideal op amps

```
v1=1;v2=0;v3=0;v4=1;
//Substituting in equation (1)
v0=-(8*v4+4*v3+2*v2+v1)
printf("v0=%dV\n",v0);

disp("b)")
v1=0;v2=1;v3=1;v4=1;
//Substituting in equation (1)
v0=-(8*v4+4*v3+2*v2+v1)
printf("v0=%dV\n",v0);
```

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

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

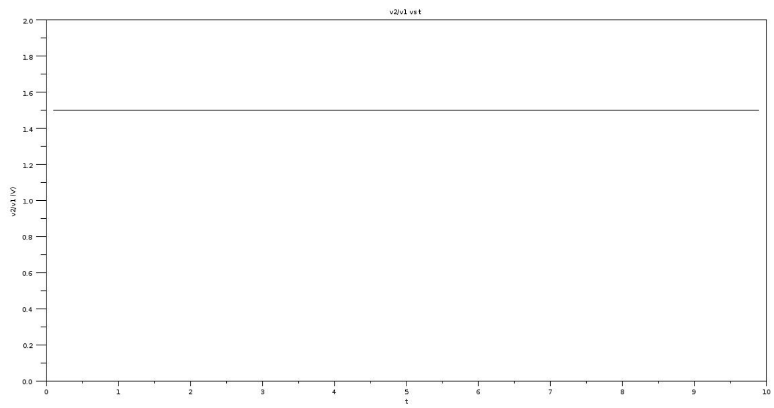

Figure 6.3: Noninverting circuit

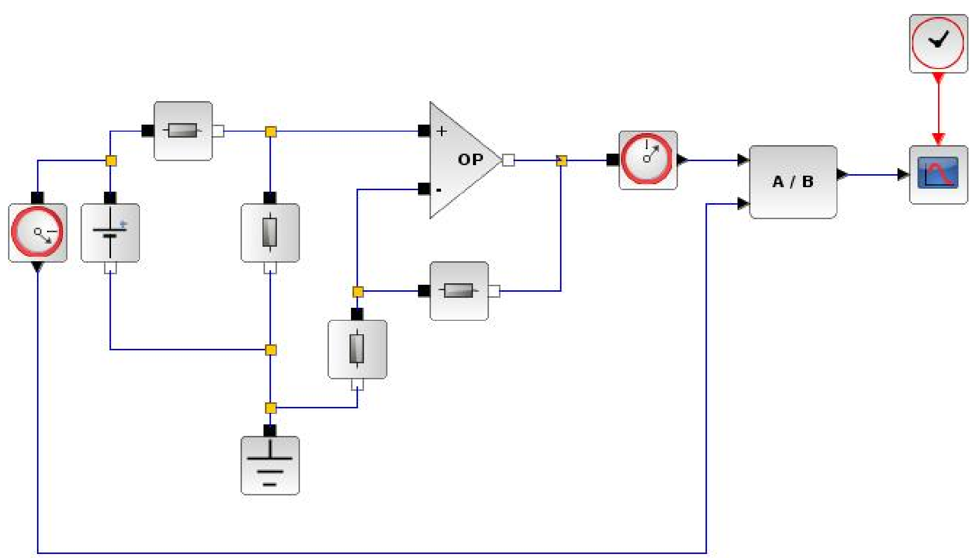

Figure 6.4: Noninverting circuit

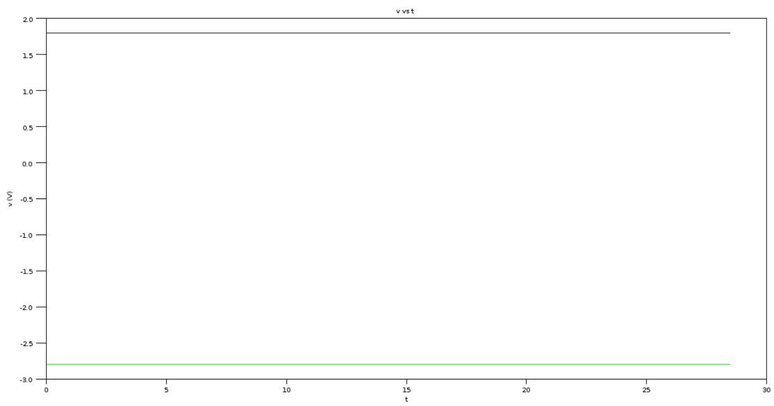

Figure 6.5: Circuits containing several Op amps

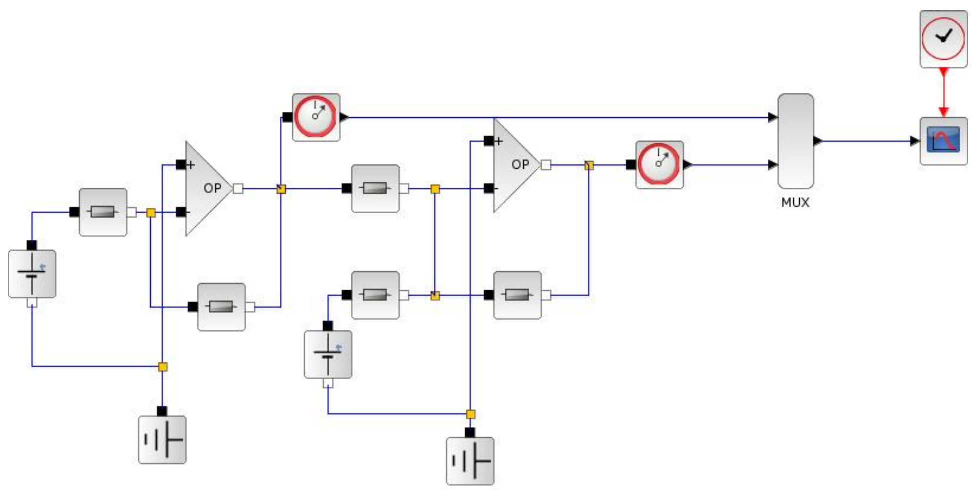

Figure 6.6: Circuits containing several Op amps

Chapter 7

Waveforms and Signals

Scilab code Exa 7.1 Sinusoidal Functions

```
clc
disp("Example 7.1")
printf("\n")

t1=-5:0.5:8
v1=cos (t1)
figure
a= gca ();
plot(t1,v1)
xtitle ('v1 vs t1','t1','v1 ');
a. thickness = 2;
//From the graph
printf("Time period1= %3.3fs\n Frequency 1=%0.3fHz\n
   ",6.2832,0.159)

t2=-4:0.5:10
v2=sin (t2)
figure
a= gca ();
plot(t2,v2)
xtitle ('v2 vs t2','t2','v2 ');
```

```
a. thickness = 2;
//From the graph
printf("Time period 2= %3.3fs\n Frequency 2=%0.3fHz\
   n",6.2832,0.159)

t3=-1:0.05:1.5
v3=2*cos (2*%pi*t3)
figure
a= gca ();
plot(t3,v3)
xtitle ('v3 vs t3','t3','v3 ');
a. thickness = 2;
//From the graph
printf("Time period 3= %ds\n Frequency 3=%dHz\n"
   ,1,1)

t4=-5:0.5:12
v4=2*cos (%pi*t4/4-%pi/4)
figure
a= gca ();
plot(t4,v4)
xtitle ('v4 vs t4','t4','v4 ');
a. thickness = 2;
//From the graph
printf("Time period 4= %ds\n Frequency 4=%0.3fHz\n"
   ,8,0.125)

t5=-1:0.005:1
v5=5*cos (10*t5+%pi/3)
figure
a= gca ();
plot(t5,v5)
xtitle ('v5 vs t5','t5','v5 ');
a. thickness = 2;
//From the graph
printf("Time period 5= %0.3fs\n Frequency 5=%3.2fHz\
   n",.62832,1.59)
```

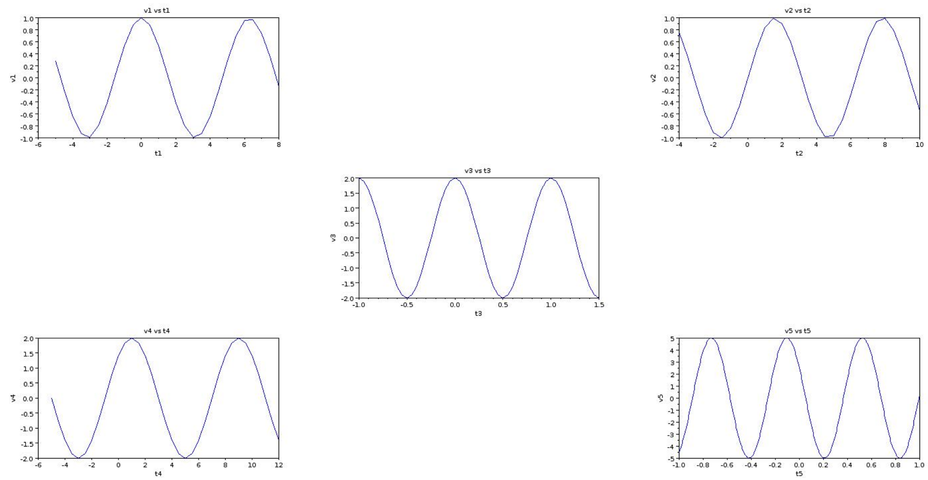

Figure 7.1: Sinusoidal Functions

Scilab code Exa 7.2 `Sinusoidal Functions`

```
clc
disp("Example 7.2")
printf("\n")

//Let wt=q
q=-8:0.5:8
v=5*cos (q)
figure
a= gca ();
plot(q,v)
xtitle ('v vs wt','wt','v ');
a. thickness = 2;
```

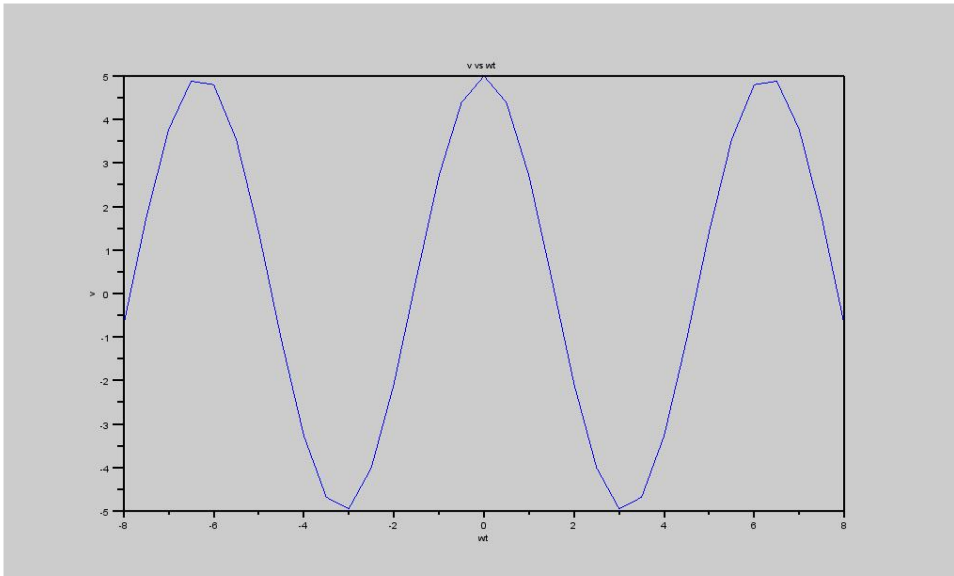

Figure 7.2: Sinusoidal Functions

Scilab code Exa 7.3 Time Shift and Phase Shift

```
clc
disp("Example 7.3")
printf("\n")

t1=-10:0.05:10
v=5*cos (%pi*t1/6+%pi/6)
figure
a= gca ();
plot(t1,v)
xtitle ('v vs %pi*t/6','%pi*t/6','v ');
a. thickness = 2;
```

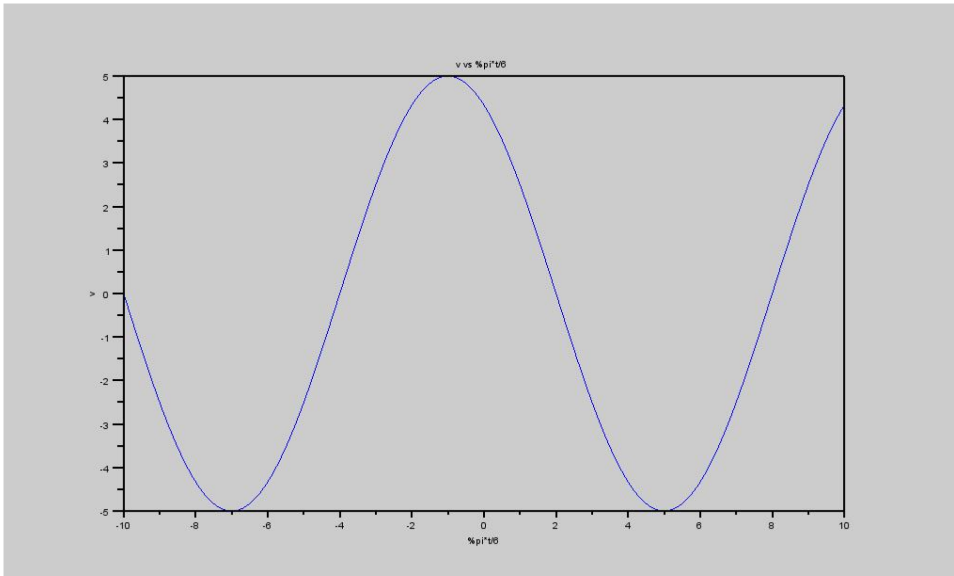

Figure 7.3: Time Shift and Phase Shift

Scilab code Exa 7.5 `Combinations of Periodic Functions`

```
clc
disp("Example 7.5")
printf("\n")

printf("Given")
disp("v(t)=cos5t+3sin(3t+45)")
//Finding the periods of individual terms
disp("Period of cos5t=2*%pi/5")
disp("Period of 3*sin(3t+45)=2*%pi/3")
//If T=2*%pi
T=2*%pi;
disp("Now T=5*T1=3*T2")
//Now the relation for T is the smallest common
   integral multiple of T1 and T2
```

```
printf("Period = %3.2fs\n",T)
```

Scilab code Exa 7.13 The Average and Effective RMS values

```
clc
disp("Example 7.13")
printf("\n")

printf("Given")
disp("Capacitance is 1uF")
C=1*10^-6;
disp("a)")
//Let k=1 which results in t=5ms
t=5*10^-3;
vac=(integrate('.004','t',0,0.003)-integrate('.002','t',0.003,0.005))/C;
printf("vac=%dV\n",vac);

//In general
disp("At t=5k voltage follows as v=8k ms")

disp("b)")
//As vdc=1/C*integrate(Idc*dt)
//On solving for Idc
vdc=vac
Idc=(1/((integrate('1/vac','t',0,0.005))/C))*10^3
printf("Idc=%3.2fmA\n",Idc);
disp("Idc is equal to <i(t)> in the period of 5ms")
```

Scilab code Exa 7.17 The Unit Impulse Function

```
clc
disp("Example 7.17")
```

```
printf("\n")

printf("Given")
disp("Capacitance is 100nF")
disp("The voltage across capacitor increases
    linearly from 0 to 10V")
C=100*10^-9;
//From figure 7.10(a)
disp("a)")
//At t=T voltage across capacitor =10V
vc=10;
Q=C*vc;
printf("Charge across capacitor is %fC\n",Q)
disp("b)")
//The waveform shown in fig 7.10(a) can be written
    as
disp("0                    t<0")
disp("I0=10^-6/T           0<t<T")
disp("0                    t>T")


//For T=1s;
T=1;
I0=10^-6/T;
printf("I0(1s)=%fA\n",I0);

//For T=1ms;
T=1*10^-3;
I0=10^-6/T;
printf("I0(1ms)=%0.3fA\n",I0);

//For T=1us;
T=1*10^-6;
I0=10^-6/T;
printf("I0(1us)=%dA",I0);
```

Scilab code Exa 7.22 The Exponential Function

```
clc
disp("Example 7.22")
printf("\n")

//The general equation of exponential decay function
    is given by
disp("v(t)=A*e(-t/T)+B")
//We need to solve A and B
//At t=0 we get v(0)=A+B      (1)
//at t=inf we get B=1         (2)
//Solving (1) and (2)
A=4;B=1;
T=3;
t=0:0.05:10
v=4*exp(-t/T)+1;
figure
a= gca ();
plot(t,v)
xtitle ('v vs t','t','v');
a. thickness = 2;
```

Scilab code Exa 7.23 The Exponential Function

```
clc
disp("Example 7.23")
printf("\n")

//Sketch voltage 'v'
```

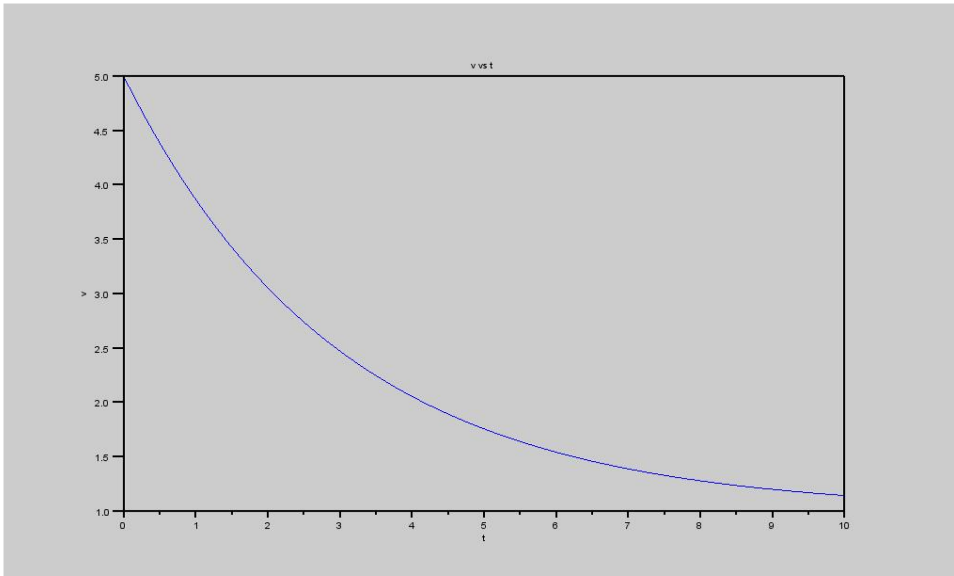

Figure 7.4: The Exponential Function

```
t=-.001:0.00005:0
t1=0:0.00005:.001
T=1*10^-3;
V0=10;
v=V0*exp(t/T)
v1=V0*exp(-t1/T)
figure
a= gca ();
plot(t,v)
plot(t1,v1)
xtitle ('v vs t','t (ms)','v ');
a. thickness = 2;

//Sketch current 'i'
t=-.001:0.00005:0
t1=0:0.00005:.001
T=1*10^-3;
I0=10*10^-3;
i=I0*exp(t/T)
```

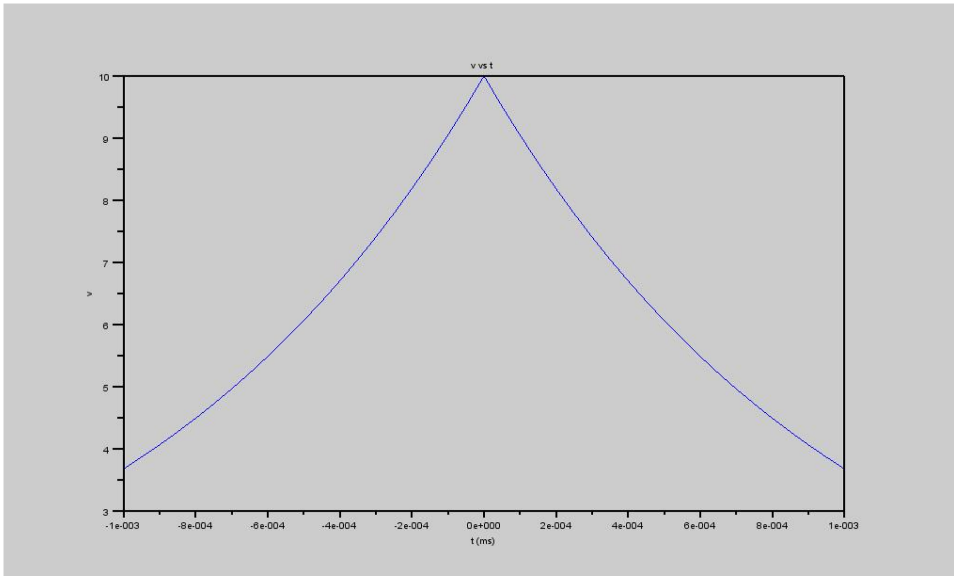

Figure 7.5: The Exponential Function

```
i1=-I0*exp(-t1/T)
figure
a= gca ();
plot(t,i)
plot(t1,i1)
xtitle ('i vs wt','t (ms)','i (mA)');
a. thickness = 2;
```

Scilab code Exa 7.25 `Random Signals`

```
clc
disp("Example 7.25")
printf("\n")
```

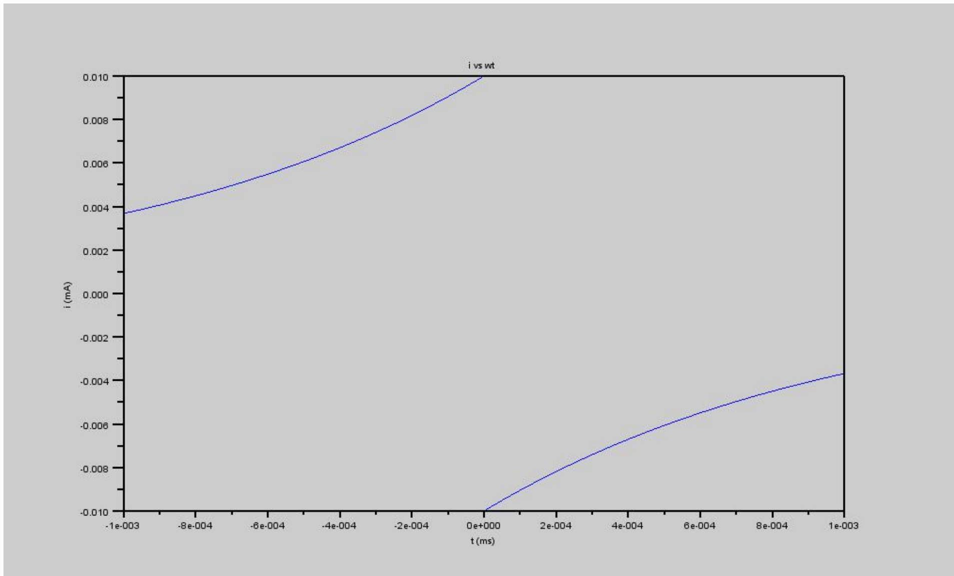

Figure 7.6: The Exponential Function

```

Xavg=(2+4+11+5+7+6+9+10+3+6+8+4+1+3+5+12)/16;
//Let X=X^2eff
X
   =(2^2+4^2+11^2+5^2+7^2+6^2+9^2+10^2+3^2+6^2+8^2+4^2+1^2+3^2+5^2+1
   /16
Xeff=sqrt(X);
printf("Xavg=%d\n  Xeff=%3.2f\n",Xavg,Xeff)
```

Scilab code Exa 7.26 `Random Signals`

```
clc
disp("Example 7.26")
printf("\n")

printf("Given")
disp("Period =10s")
```

```
disp("Interval is 1ms")
disp("Voltage of binary signal is either 0.5 or -0.5
   ")
T=10;
//During 10s period there are 10000 intervals of 1ms
    each
//For calculating average equal number of intervals
   are considered at 0.5V and -0.5V
vavg=(0.5*5000-0.5*5000)/10000
//The effective value of v(t) is
//Let V=V^2eff
V=(0.5^2*5000+(-0.5)^2*5000)/10000
Veff=sqrt(V)
printf("vavg=%dV\nVeff=%3.2fV\n",vavg,Veff)
```

Chapter 8

First order Circuits

Scilab code Exa 8.1 `Capacitor Discharge in a Resistor`

```
clc
disp("Example 8.1")
printf("\n")

printf("Given")
disp("Capacitance is 1uF")
disp("Resistance is 1Mohm")
disp("Voltage across capacitor is 10V")
R=1*10^6;C=1*10^-6;V=10
//Let T be time constant
T=R*C
//v(t)=V*exp(-t/T)
disp("v(t)=10*exp(-t)        (1)")
//Substituting value of t=5 in (1)
v5=10*exp(-5)
printf("Time constant is %ds\n",T)
printf("v(5)=%0.3fV\n",v5)
```

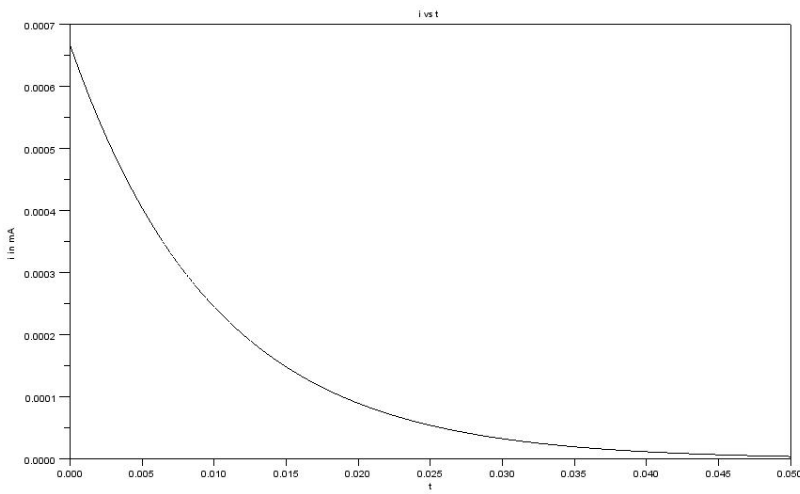

Figure 8.1: Capacitor Discharge in a Resistor

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

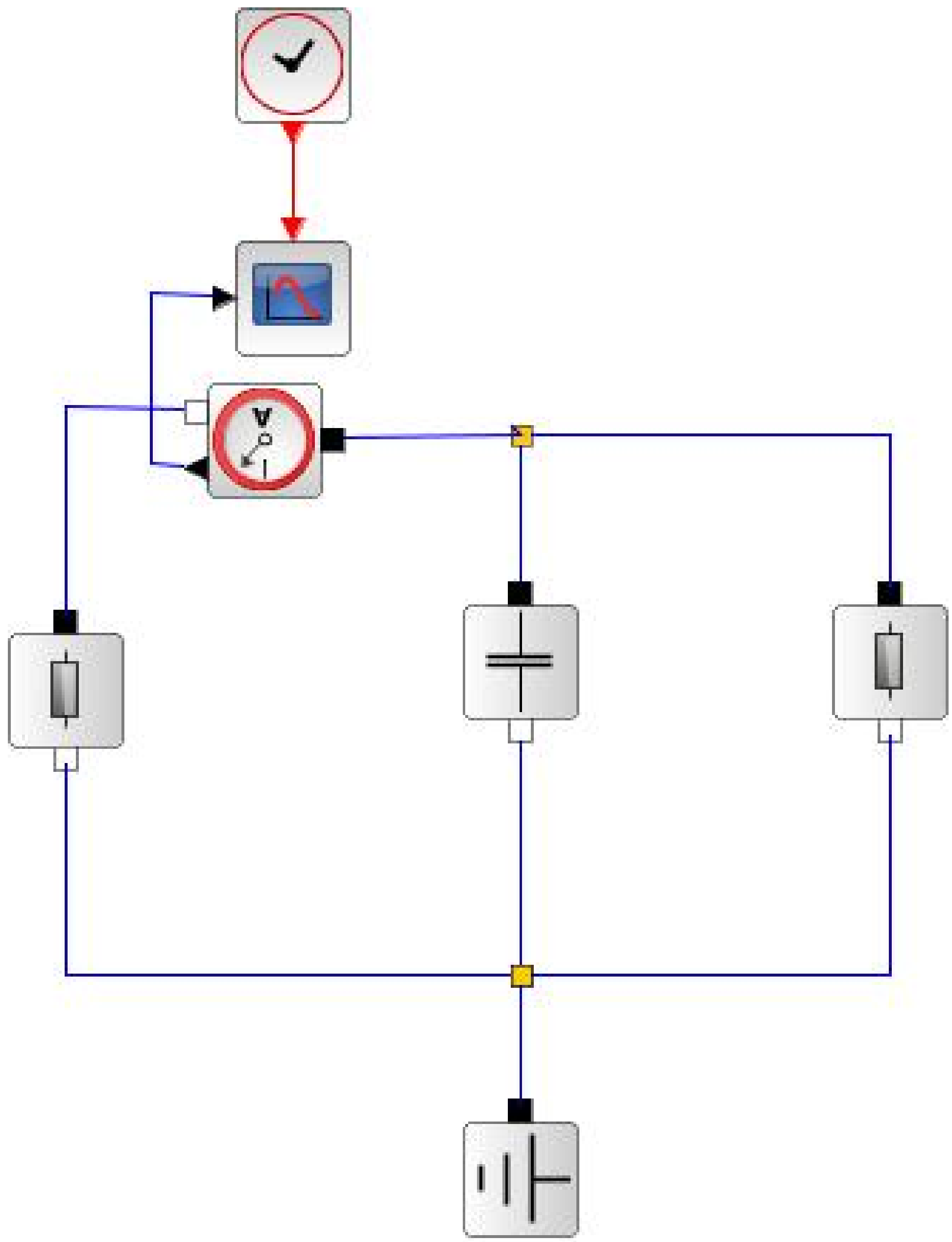

Figure 8.2: Capacitor Discharge in a Resistor

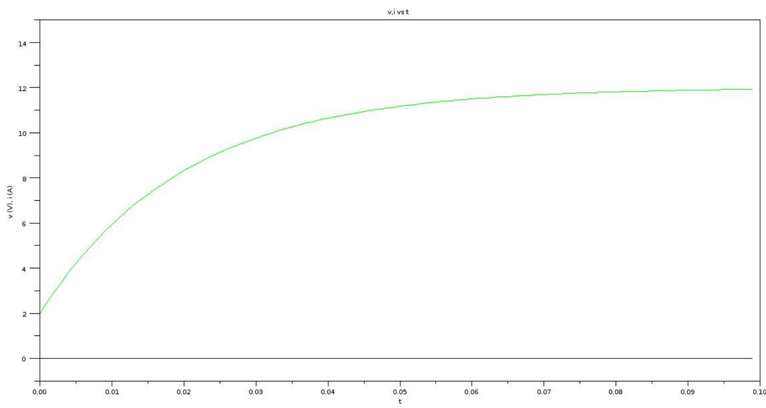

Figure 8.3: Establishing a DC Voltage across a Capacitor

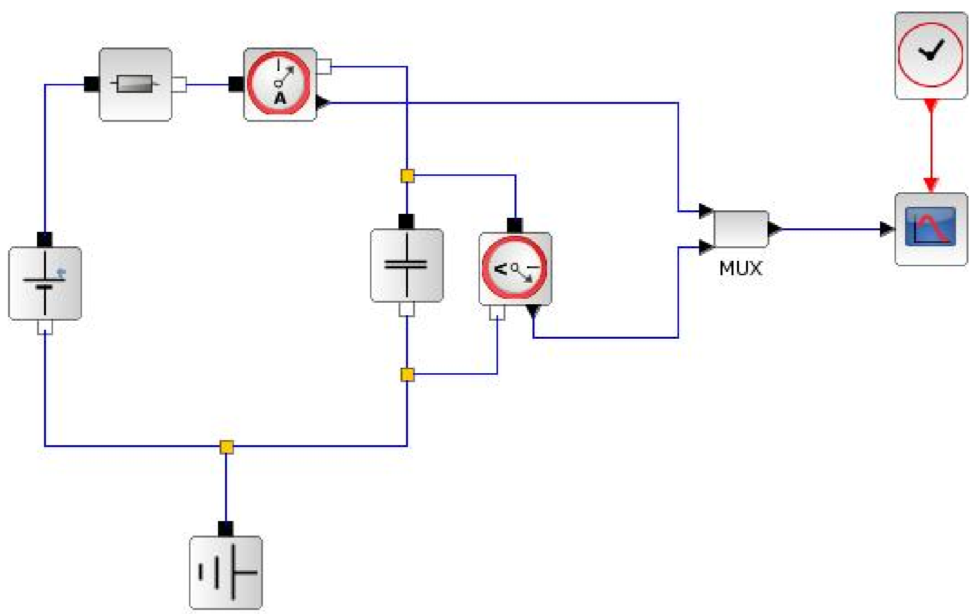

Figure 8.4: Establishing a DC Voltage across a Capacitor

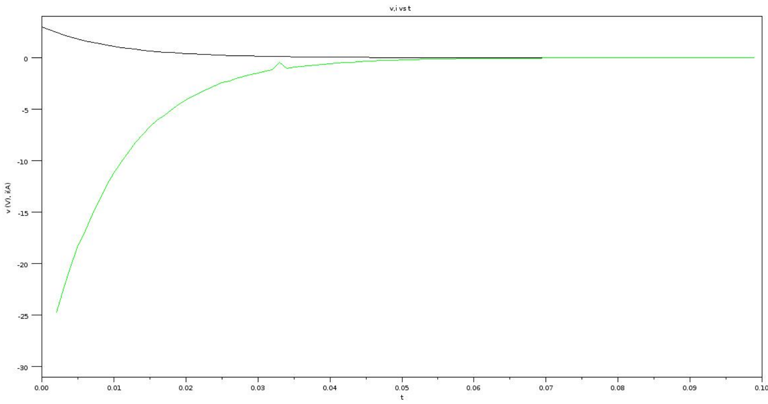

Figure 8.5: The Source free RL Circuit

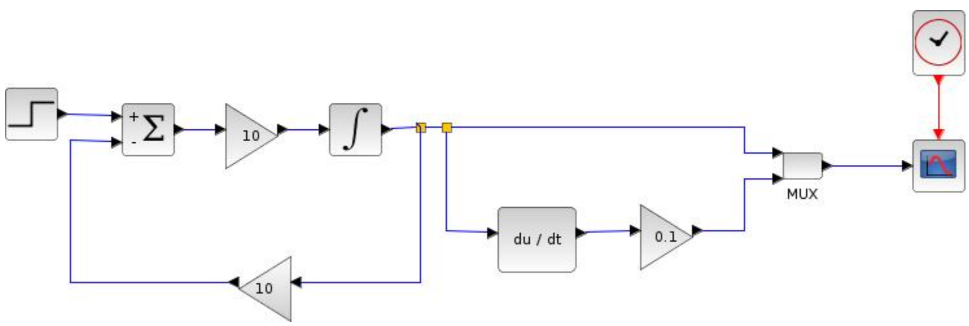

Figure 8.6: The Source free RL Circuit

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

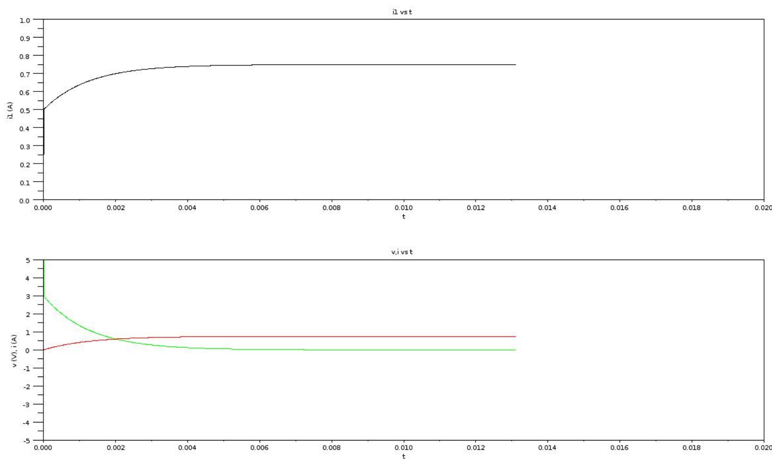

Figure 8.7: Complex first order RL and RC Circuits

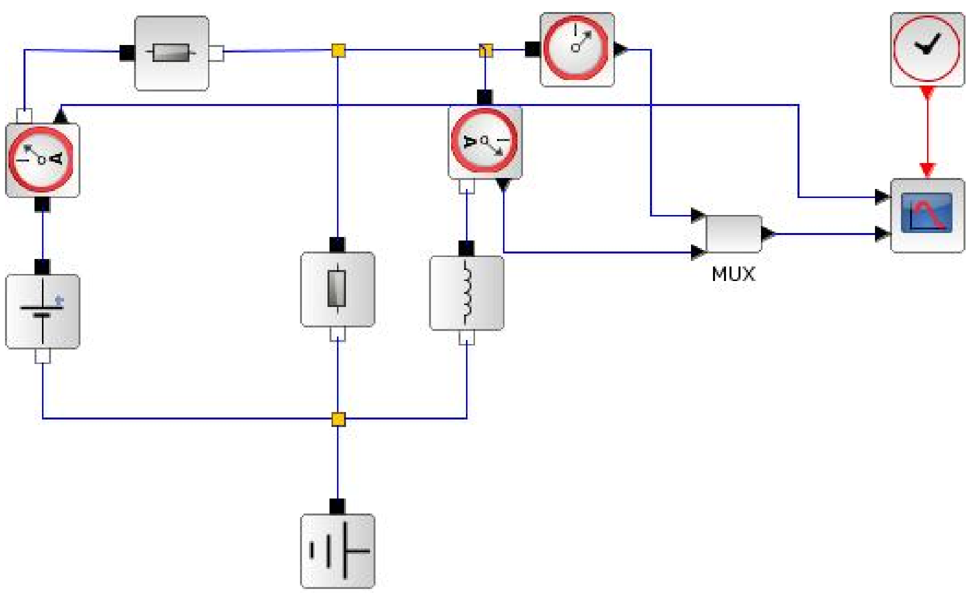

Figure 8.8: Complex first order RL and RC Circuits

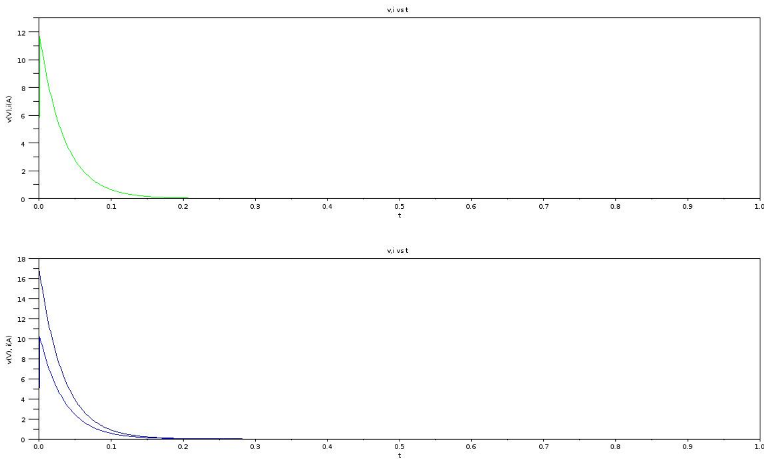

Figure 8.9: Complex first order RL and RC Circuits

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

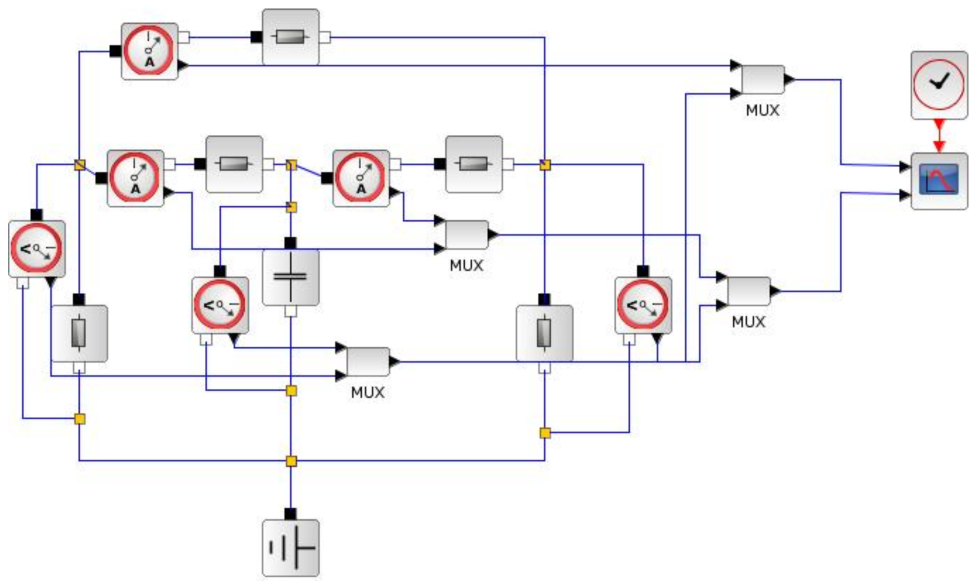

Figure 8.10: Complex first order RL and RC Circuits

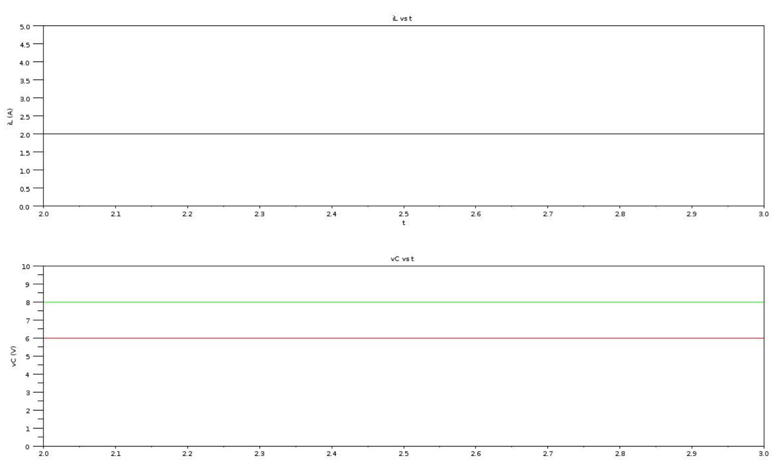

Figure 8.11: DC Steady state in Inductors and Capacitors

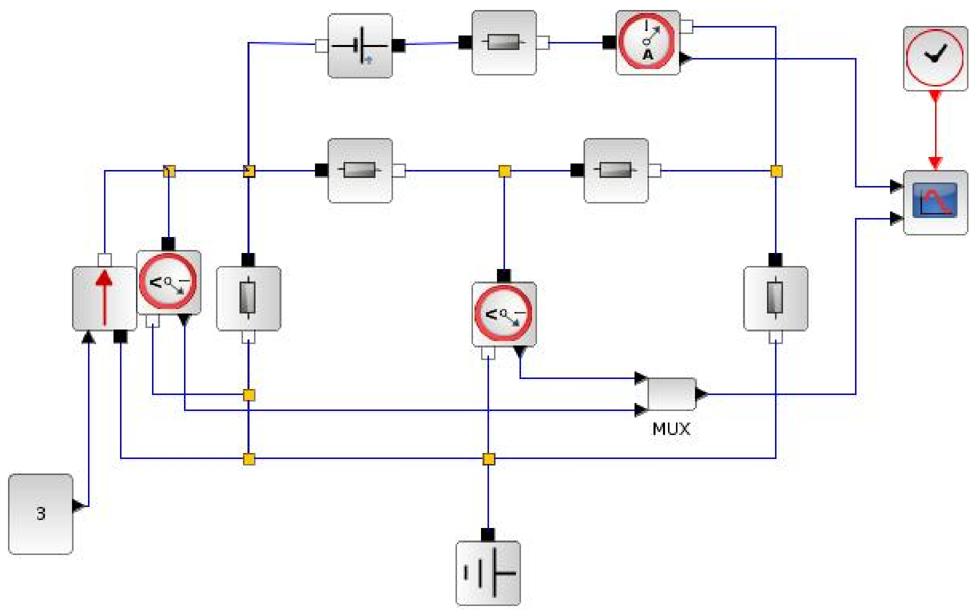

Figure 8.12: DC Steady state in Inductors and Capacitors

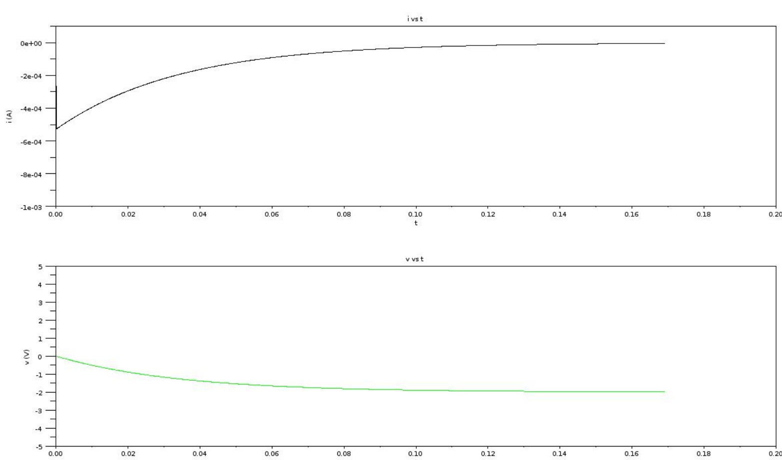

Figure 8.13: DC Steady state in Inductors and Capacitors

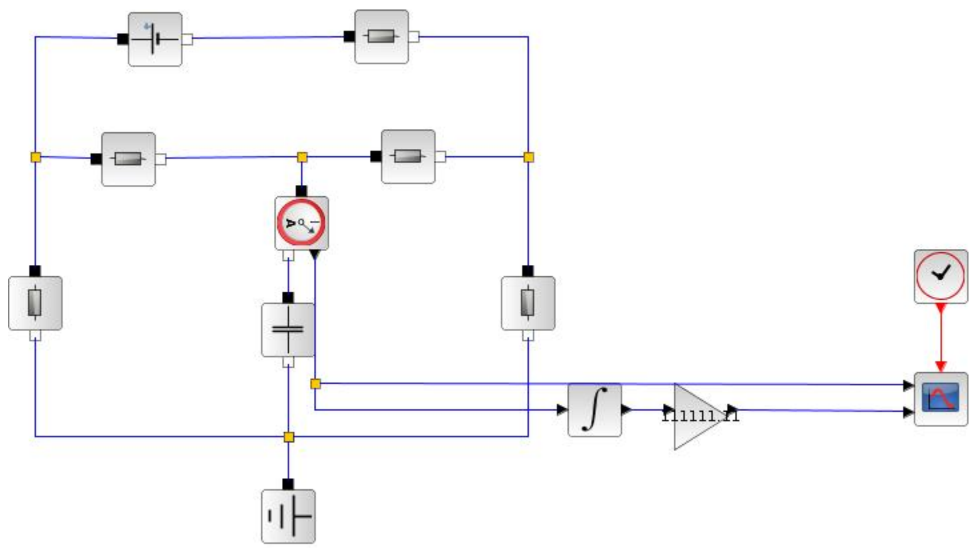

Figure 8.14: DC Steady state in Inductors and Capacitors

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

Scilab code Exa 8.10 `Transitions at Switching Time`

```
clc
disp("Example  8.10")
printf("\n")

printf("Given")
disp("vs=  5V             t<0")
disp("vs=5*sin(w*t)    t>0")
```

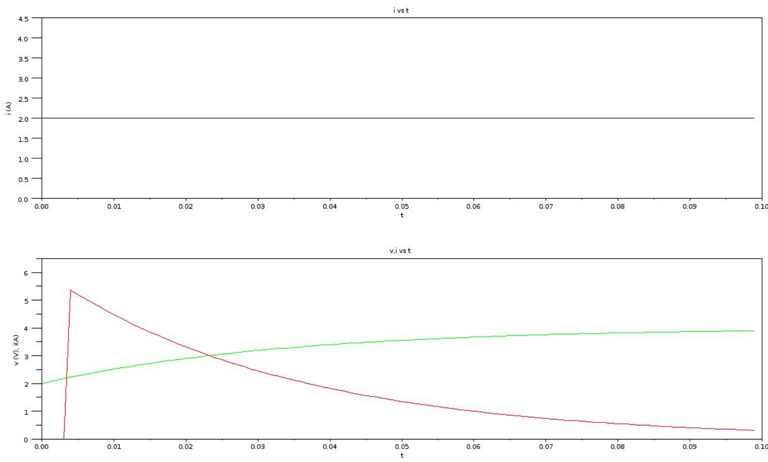

Figure 8.15: Transitions at Switching Time

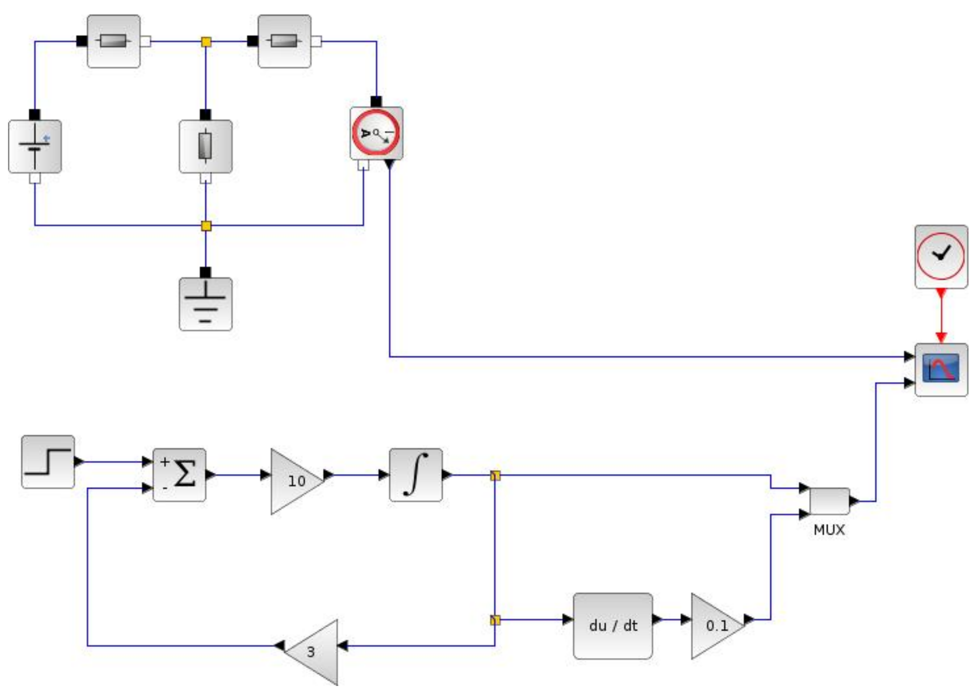

Figure 8.16: Transitions at Switching Time

```
vs=5;
R=5;L=10*10^-3;
//At t<0
//Inductor behaves as a short circuit
//Let i(0-)=i
i=vs/R;
printf("i(0-)=%dA\n",i)
//During the transition from t=0- to t=0+
//Let i(0+)=i1
i1=i
printf("i(0+)=%dA\n",i1)
//Applying KVL equation to the loop
disp("vs=i*R+v")
//Let v(0+)=v1 ; vs(0+)=vs1
//From given vs(0+)=0
vs1=0;
v1=vs1-i*R
printf("\nv(0+)=%dV\n",v1)
```

Chapter 9

Higher order circuits and Complex frequency

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

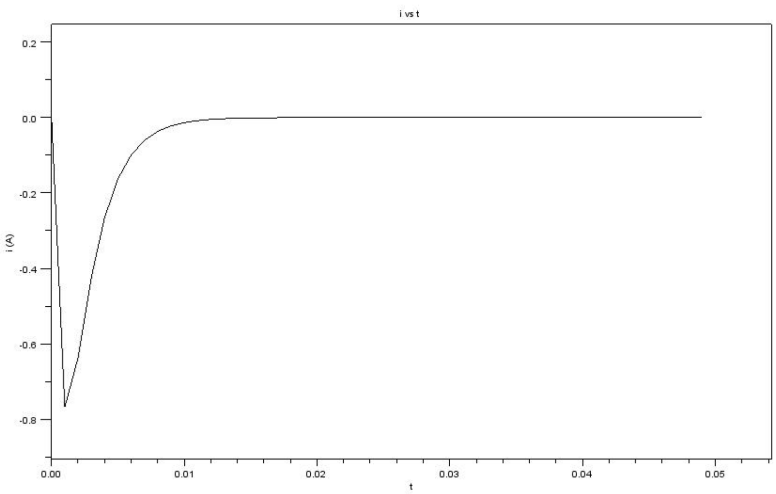

Figure 9.1: Series RLC Circuit

Scilab code Exa 9.6 `Generalized Impedance RLC in s domain`

```
clc
disp("Example 9.6")
printf("\n")

printf("Given")
disp("Resistance is 10ohm and inductance is 2H")
disp("Applied voltage is 10*exp(-2*t)*cos(10*t+30)")
s=%s;
//For a RL circuit
//Applying KVL equation
//v=i*R+L*d/dt(i)      (1)
//As v=10(30 deg)      (2)
//Equating (1) and (2)
// Let i=I*exp(s*t)      (3)
// 10(30 deg)*exp(s*t)=10*I*exp(s*t)+2*s*I*exp(s*t)
   ")
//Solving for I
```

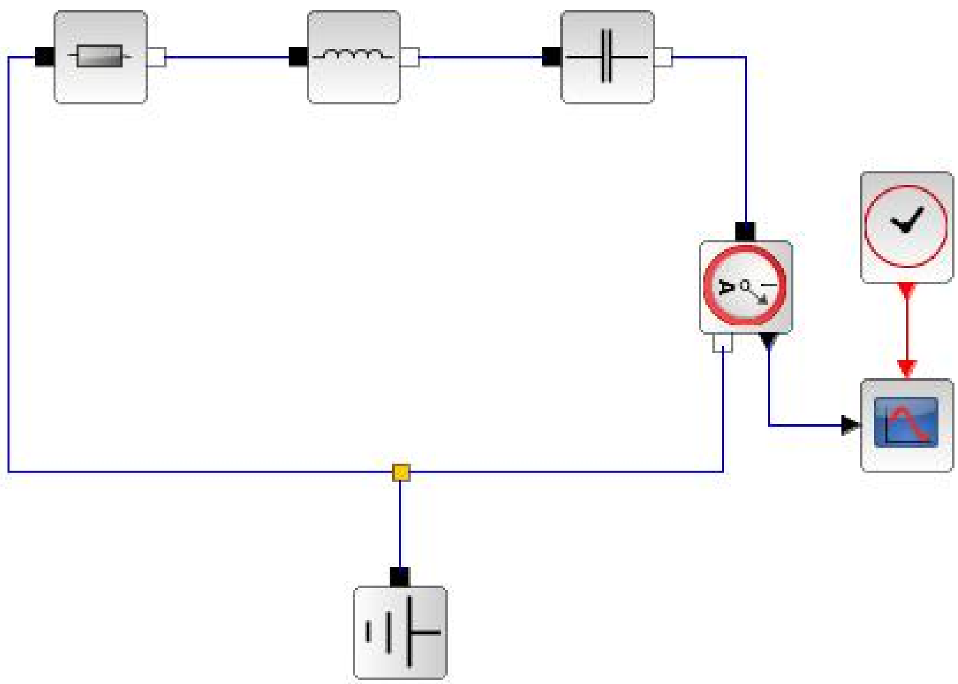

Figure 9.2: Series RLC Circuit

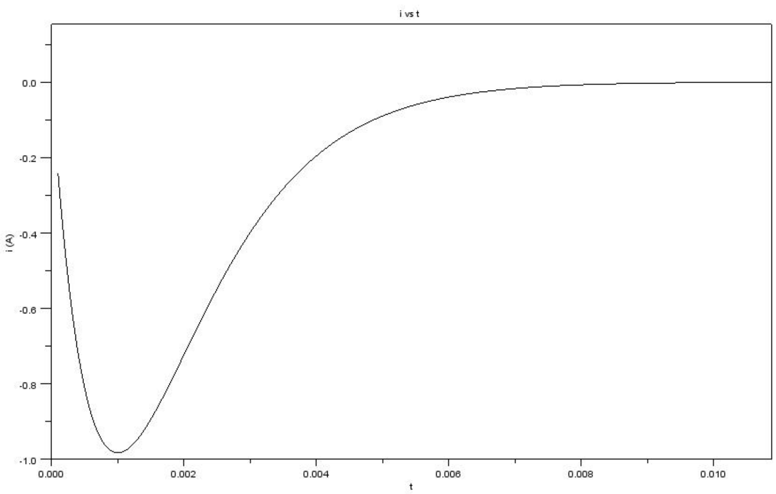

Figure 9.3: Series RLC Circuit

```
disp("I=10(30 deg)/10+2*s")
s=-2+%i*10
a=10+2*s
x=10*cos((30*%pi)/180);
y=10*sin((30*%pi)/180);
z=complex(x,y)
I=z/a
b=real(I);
c=imag(I);
magn=sqrt(b^2+c^2)
ph=(atan(c/b)*180)/%pi
//From (3)
printf("\ni=%0.2f*exp(-2*t)*cos(10t%3.1f deg) (A)\n"
    ,magn,ph);
```

Scilab code Exa 9.7 `Generalized Impedance RLC in s domain`

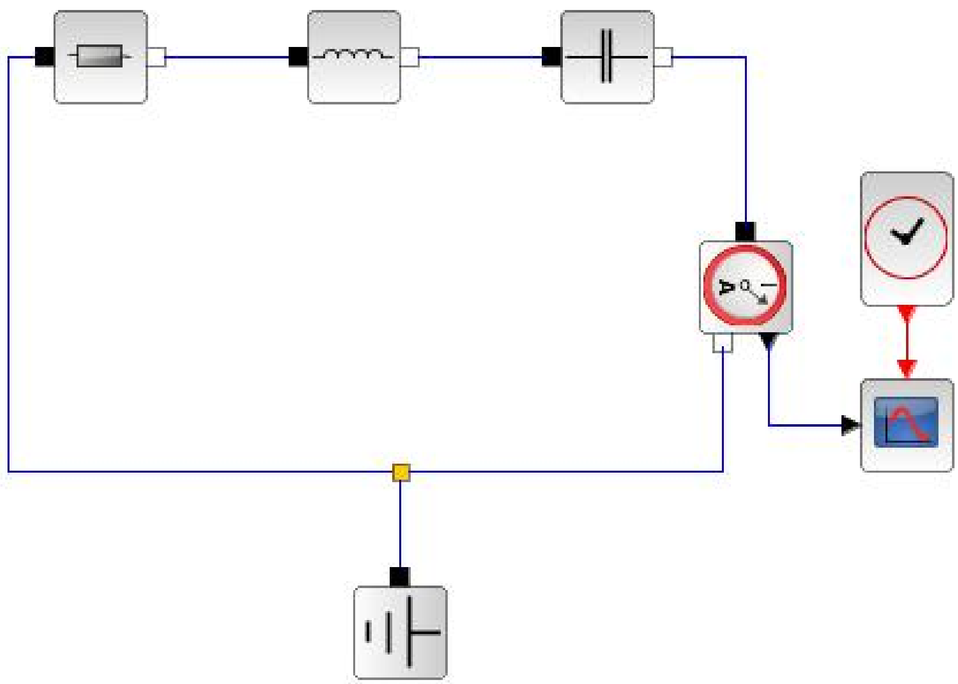

Figure 9.4: Series RLC Circuit

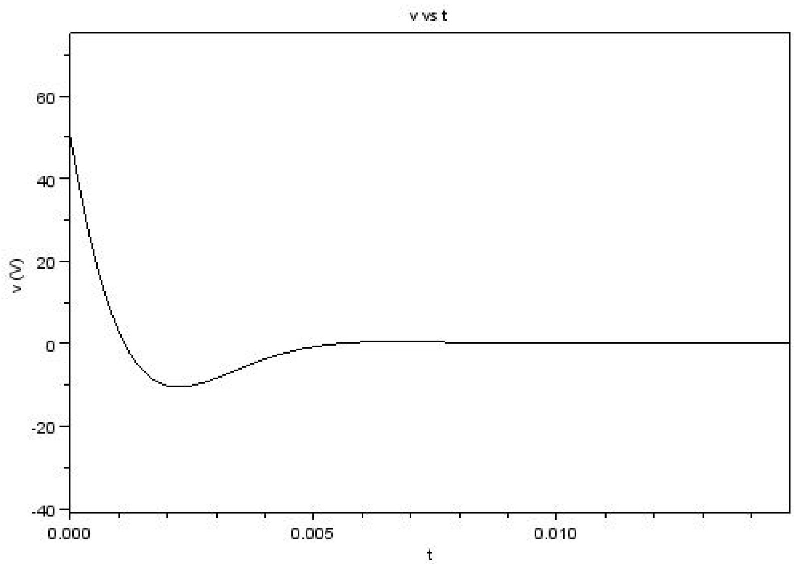

Figure 9.5: Parallel RLC circuit

```
clc
disp("Example 9.7")
printf("\n")

printf("Given")
disp("Resistance is 10ohm and Capacitance is 0.2F")
disp("Applied voltage is 10*exp(-2*t)*cos(10*t+30)")
s=%s;
//For a RC circuit
//Applying KVL equation
//v=i*R+(1/C)*integrate(i*dt)      (1)
//As v=10(30 deg)      (2)
//Equating (1) and (2)
// Let i=I*exp(s*t)      (3)
// 10(30 deg)*exp(s*t)=10*I*exp(s*t)+(5/s)*I*exp(s*t
    )")
//Solving for I
disp("I=10(30 deg)/10+(5/s)")
```

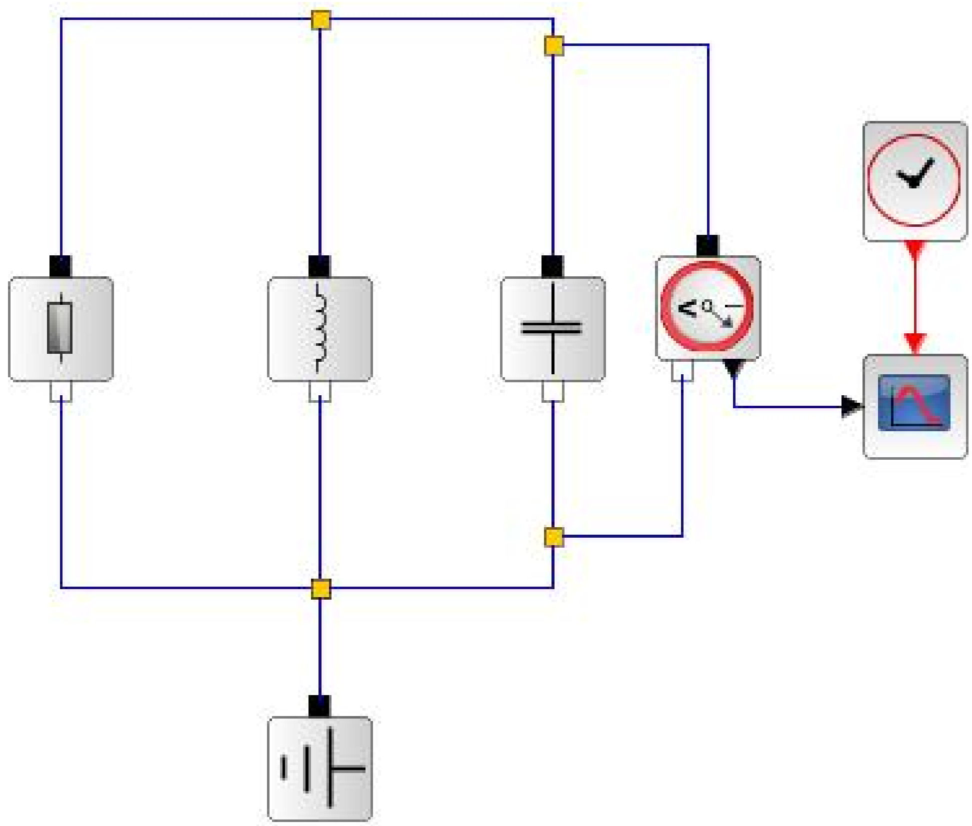

Figure 9.6: Parallel RLC circuit

```
s=-2+%i*10
a=10+(5/s)
x=10*cos((30*%pi)/180);
y=10*sin((30*%pi)/180);
z=complex(x,y)
I=z/a
b=real(I);
c=imag(I);
magn=sqrt(b^2+c^2)
ph=(atan(c/b)*180)/%pi
//From (3)
printf("\ni=%0.2f*exp(-2*t)*cos(10t+%3.1f deg) (A)\n
    ",magn,ph);
```

Scilab code Exa 9.8 `Network function and Pole zero plots`

```
clc
disp("Example 9.8")
printf("\n")

s=%s ;

//From figure 9.13
disp("Z(s)=(2.5+((5*s/3)*(20/s))/(5*s/3+20/s))")
//On solving
z1=poly([12 8 1],'s','coeff')
z2=poly([12 0 1],'s','coeff')
Z=2.5*(z1/z2)
disp(Z,"Z(s)")
//H(s)=I(s)/Z(s)
//Let I(s)=1 the H(s)=1/Z(s)
H=(1/2.5)*(z2/z1)
disp(H,"H(s)")
```

Scilab code Exa 9.9 The Forced Response

```
clc
disp("Example 9.9")
printf("\n")

s=%s ;
H=syslin ( 'c',(0.4*(s^2+12))/((s+2)*(s+6) ) ) ;
evans (H,1)
//If s=1Np/s
H1=0.4*(1+12)/((1+2)*(1+6))
printf("H(1)=%0.3f\n",H1)
```

Scilab code Exa 9.11 The Natural Response

```
clc
disp("Example 9.11")
printf("\n")

//From figure 9.16
//H(s)=V(s)/I(s)=Z(s)
//Let V(s)=1 the H(s)=Z(s)
s=%s
z1=(1/2.5)+(3/(5*s))+(s/20)
Z=1/z1
Dem=Z('den')
//The roots are
q=roots(Dem)
disp(q,"Poles are")
```

Chapter 10

Sinusoidal Steady state Circuit Analysis

Scilab code Exa 10.4 Phasors

```
clc
disp("Problem 10.4")
printf("\n")

//For V1
Ro1=25
Theta1=143.13
//For V1
Ro2=11.2
Theta2=26.57
//We need to find V1/V2
//Let V=V1/V2
Vmag=(Ro1/Ro2)
Vph=Theta1-Theta2
x=Vmag*cos((Vph*%pi)/180);
y=Vmag*sin((Vph*%pi)/180);
z=complex(x,y)
//Let V1+V2=V12
x1=Ro1*cos((Theta1*%pi)/180);
```

```
y1=Ro1*sin((Theta1*%pi)/180);
z1=complex(x1,y1)
x2=Ro2*cos((Theta2*%pi)/180);
y2=Ro2*sin((Theta2*%pi)/180);
z2=complex(x2,y2)
V12=z1+z2
[R,Theta]=polar(V12)
printf("V1/V2=%0.2f+i*%3.2f \nV1+V2=%3.2f(%3.2f deg)
    ",x,y,R,(Theta*180)/%pi)
```

Scilab code Exa 10.5 Impedance and Admittance

```
clc
disp("Problem 10.5")
printf("\n")

printf("Given")
disp("Voltage is 100(45 deg)")
disp("Current is 5(15 deg)")
//For V
Ro1=100
Theta1=45
//For I
Ro2=5
Theta2=15
//We need to find V/I=Z

Zmag=(Ro1/Ro2)
Zph=Theta1-Theta2
x=Zmag*cos((Zph*%pi)/180);
y=Zmag*sin((Zph*%pi)/180);
z=complex(x,y)
//Let Y=1/Z
Ymag=(Ro2/Ro1)
Yph=Theta2-Theta1
```

```
x1=Ymag*cos((Yph*%pi)/180);
y1=Ymag*sin((Yph*%pi)/180);
z1=complex(x1,y1)

printf("R=%3.2fohm XL=%3.2fH \nG=%0.3fS BL=%0.3fS",x
    ,y,x1,abs(y1));
```

Scilab code Exa 10.7 Superposition of AC sources

```
clc
disp("Problem 10.7")
printf("\n")

printf("Voltage v1=5*cos(w1*t)")
printf("Voltage v2=10*cos(w2*t+60)")
//The circuit is modeled as
disp("Resistance is 10ohm and inductance is 5mH")
R=10;L=5*10^-3;
disp("a)")
w1=2000;w2=2000;
//Let Z be the impedance of the coil
Z1=R+%i*L*w1
Z2=R+%i*L*w2
//Let V be phasor voltage between the terminals
Vmag=10;
Vph=60;
x=Vmag*cos((Vph*%pi)/180);
y=Vmag*sin((Vph*%pi)/180);
z=complex(x,y)
v=5-z;
//Let I be the current
I=v/Z1
[R,Theta]=polar(I)
printf("i=%0.2f*cos(%dt%d deg)",R,w1,(Theta*180)/%pi
    );
```

```

disp("b)")
R=10;L=5*10^-3;
w1=2000;w2=4000;
//Let Z be the impedance of the coil
Z1=R+%i*L*w1
Z2=R+%i*L*w2
V1=5;
//By applying superposition i=i1-i2
I1=V1/Z1
[R,Theta]=polar(I1)
printf("i1=%0.2f*cos(%dt%d deg)\n",R,w1,(Theta*180)/%pi);
V2mag=10;V2ph=60;
I2=z/Z2
[R1,Theta1]=polar(I2)
printf("i2=%0.2f*cos(%dt%3.2f deg)\n",R1,w2,(Theta1*180)/%pi);
//i=i1-i2
printf("i=%0.2f*cos(%dt%d deg)-%0.2f*cos(%dt%3.2f deg)\n",R,w1,(Theta*180)/%pi,R1,w2,(Theta1*180)/%pi)
```

Chapter 11

AC Power

Scilab code Exa 11.1 `Power in time domain`

```
clc
disp("Problem 11.1")
printf("\n")

printf("Given")
disp("Resistance =1000ohm")
t=0:0.5:1;
i=ones(length(t),1) ;i1=-1;
figure
a=gca()
plot(t,i,t+1,i1,t+2,i,t+3,i1)
xtitle("i vs t",'t in ms','i in mA')
i=1*10^-3;R=1000;
//p=i^2*R
p=i^2*R*ones(length(t),1) ;
figure
a=gca()
plot(t,p)
xtitle("p vs t",'t in ms','p in mW')
```

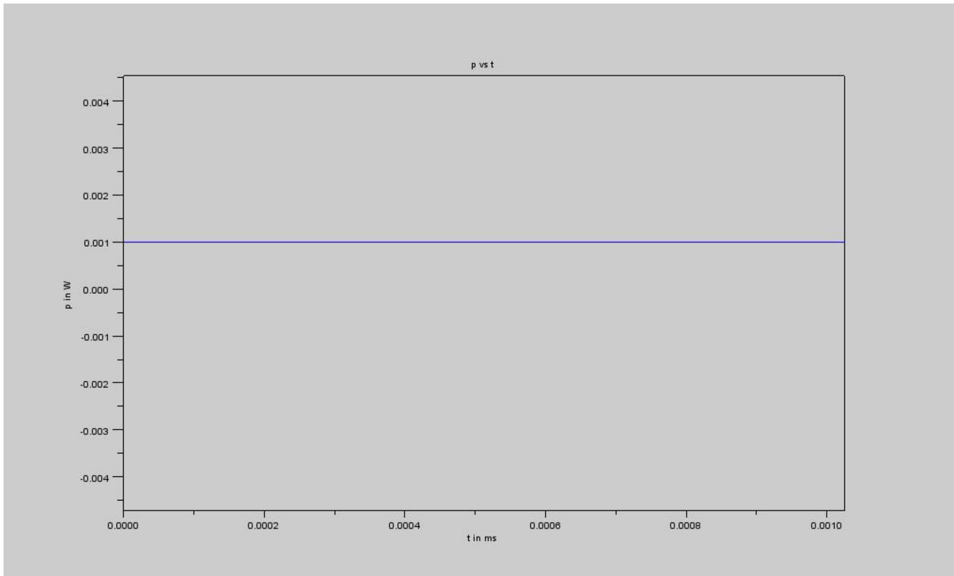

Figure 11.1: Power in time domain

Scilab code Exa 11.2 `Power in time domain`

```
clc
disp("Problem 11.2")
printf("\n")

t=0:0.5:1;
i=ones(length(t),1);i1=-1;
figure
a=gca()
plot(t,i,t+1,i1)
xtitle("i vs t",'t in ms','i in mA')
```

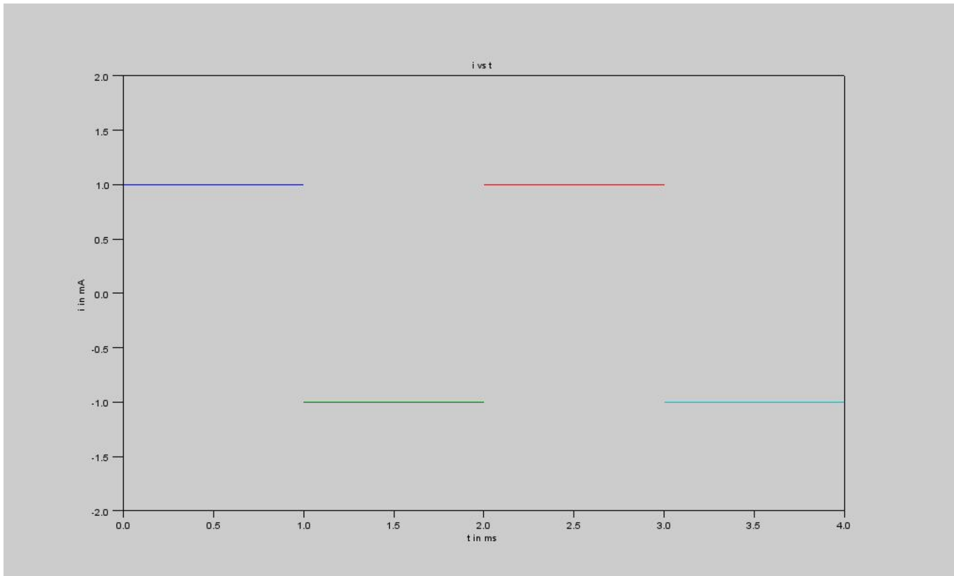

Figure 11.2: Power in time domain

```
//Voltage across capacitor vC=(1/C)*integrate(i*dt)
//On integration
t=0:0.0005:0.001
v=2000*t
v1=2-v;
figure
a=gca()
plot(t,v,t+0.001,v1,t+0.002,v,t+0.003,v1)
xtitle("v vs t",'t in ms','v in V')

//Power is p=v*i
t=0:.0005:.001
p=2000*t
p1=p-2;
figure
a=gca()
plot(t,p,t+0.001,p1,t+0.002,p,t+0.003,p1)
xtitle("p vs t",'t in ms','p in W')

```

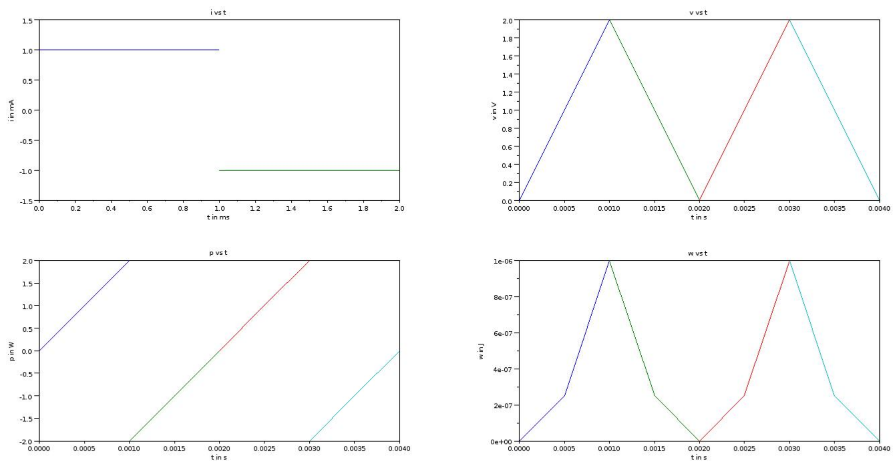

Figure 11.3: Power in time domain

```
//Work is (C*v^2)/2
t=0:.0005:.001
w=t.^2
w1=t.^2+1*10^-6-(2*10^-3*t);
figure
a=gca()
plot(t,w,t+0.001,w1,t+0.002,w,t+0.003,w1)
xtitle("w vs t",'t in ms','w in J')
```

Scilab code Exa 11.4 `Average or Real Power`

```
clc
disp("Problem 11.4")
printf("\n")

printf("Given")
disp("Veff=110V Z=10+i8 ohm")
Veff=110;
```

```
Z=10+%i*8;
R = 10;
Zmag=sqrt(10^2+8^2)
Zph=(atan(8/10)*180)/%pi
P=(Veff^2*R)/(Zmag^2)
pf=cos((Zph*%pi)/180)

disp(pf,"Power factor is")
```

Scilab code Exa 11.5 Reactive Power

```
clc
disp("Problem 11.5")
printf("\n")

printf("Given")
disp("Veff=110V Ieff=20(-50 deg)")
Imagn=20;Iph =-50;
Veff=110;

P=Veff*Imagn*cos((abs(Iph)*%pi)/180)
Q=Veff*Imagn*sin((abs(Iph)*%pi)/180)
printf("Average power is %3.1fW\n",P)
printf("Reactive power is %3.1fvar\n",Q)
```

Scilab code Exa 11.10 Complex power Apparent power and Power triangle

```
clc
disp("Problem 11.10")
printf("\n")

printf("Given")
disp("Veff=10V v=10*sqrt(2)*cos(w*t)");
```

```
Veff=10;vmag=10*1.414

disp("a)")
Z1=1+%i
[R,Theta]=polar(Z1)
printf("i1=%d*cos(w*t-%d)\n",(vmag/R),Theta)
I1eff=(vmag/R)/1.414
//p1(t)=100*sqrt(2)*cos(wt)*cos(wt-45)
//On solving
disp("p1(t)=50+50*sqrt(2)*cos(2*w*t-45) W")
P1=Veff*I1eff*cos(Theta)
Q1=Veff*I1eff*sin(Theta)
S1=P1+%i*Q1
S1mag=sqrt(P1^2+Q1^2)
pf1=P1/S1mag
printf("P1=%dW\nQ1=%dvar\npf1=%0.4f(lag)\n",P1,Q1,pf1)


disp("b)")
Z2=1-%i
[R,Theta]=polar(Z2)
printf("i2=%d*cos(w*t%d)\n",(vmag/R),Theta)
I2eff=(vmag/R)/1.414
//p2(t)=100*sqrt(2)*cos(wt)*cos(wt+45)
//On solving
disp("p2(t)=50+50*sqrt(2)*cos(2*w*t+45) W")
P2=Veff*I2eff*cos(Theta)
Q2=Veff*I2eff*sin(Theta)
S2=P2+%i*Q2
S2mag=sqrt(P2^2+Q2^2)
pf2=P2/S2mag
printf("P2=%dW\nQ2=%dvar\npf2=%0.4f(lag)\n",P2,Q2,pf2)

disp("c)")
Zmag=(Z1*Z2)/(Z1+Z2)
printf("i=%d*cos(w*t)\n",(vmag/Zmag))
```

```
Ieff=(vmag/Zmag)/1.414
//p(t)=100*sqrt(2)*sqrt(2)*cos(wt)*cos(wt)
//On solving
disp("p2(t)=200*cos(w*t)^2 W")
P=Veff*Ieff
Q=0
S=P+%i*Q
Smag=sqrt(P^2+Q^2)
pf=P/Smag
printf("P=%dW\nQ=%dvar\npf=%0.4f\n",P,Q,pf)
```

Scilab code Exa 11.11 Complex power Apparent power and Power triangle

```
clc
disp("Problem 11.11")
printf("\n")

printf("Given")
disp("v=42.5*cos(1000*t+30 deg)V Z=3+i4 ohm")
Vmag=42.5;
Z=3+%i*4;
R=sqrt(3^2+4^2)
Theta=atan(4/3)*(180/%pi)
Veffm=Vmag/sqrt(2)
Veffph=30
Ieffm=Veffm/R
Ieffph=30-Theta

Smag=Veffm*Ieffm
Sph=Veffph-Ieffph
x=Smag*cos((Sph*%pi)/180)
y=Smag*sin((Sph*%pi)/180)
z=complex(x,y)
pf=cos((Theta*%pi)/180);

```

```
printf("Real Power is %fW\n",x)
printf("Reactive Power is %fvar(inductive)\n",y)
printf("Complex Power is %fVA\n",Smag)
printf("Power factor is %3.1f(lag)\n",pf)
```

Scilab code Exa 11.12 Parallel connected Networks

```
clc
disp("Problem 11.12")
printf("\n")

printf("Given")
disp("pf1=1 ; pf2=0.5 ; pf3=0.5")
disp("P1=10kW;P2=20kW;P3=15kW")
disp("Power supply is 6kV")
P1=10000;P2=20000;P3=15000;
Veff=6000;
pf1=1   //implifies that theta1=0
t1=0
Q1=P1*t1

pf2=0.5   //implifies that theta1=60
t2=1.73;
Q2=P2*t2

pf3=1   //implifies that theta1=53.13
t3=1.33;
Q3=P3*t3

PT=P1+P2+P3
QT=Q1+Q2+Q3
ST=sqrt(PT^2+QT^2)
pfT=PT/ST
Ieff=ST/Veff
Ieffph=acos(pfT)*(180/%pi)
```

```
printf("PT=%dW\nQT=%dvar\nST=%dVA\npf=%0.2f(lag)\
    nIeff=%3.1f(%3.2f deg)\n",PT,QT,ST,pfT,Ieff,
    Ieffph)
```

Scilab code Exa 11.13 Power factor improvement

```
clc
disp("Problem 11.13")
printf("\n")

printf("Given")
disp("Power factor is 0.95(lag)")
vmag=240;Zmag=3.5;Zph=25;
I1mag=vmag/Zmag;iph=0-Zph;
//Smag=Veff*Ieff
Smag=(vmag/sqrt(2))*(I1mag/sqrt(2))
Sph=0+abs(iph)
x=Smag*cos((Sph*%pi)/180)
y=Smag*sin((Sph*%pi)/180)
z=complex(x,y)
pf=0.95
theta=acos(0.95)*(180/%pi)
//From fig 11.11
//Solving for Qc
Qc=y-(tan((theta*%pi)/180)*x)
printf("\n Qc=%dvar(Capacitive )\n",Qc)
```

Scilab code Exa 11.14 Power factor improvement

```
clc
disp("Problem 11.14")
printf("\n")

```

```
printf("Given")
disp("Power =1000kW ; pf=0.5(lag)")
disp("Voltage source is 5kV")
disp("Improved power factor is 0.8")

//Before improvement
P=1000*10^3;
pf=0.5;V=5*10^3;
S=(P/pf)*10^-3
I=S/V

//After improvement
P=1000*10^3;
pf=0.8;V=5*10^3;
S=(P/pf)*10^-3
I1=S/V

disp("Current is reduced by ")
red=((I-I1)/I)*100
printf("Percentage reduction in current is %3.1
   fpercent\n",red)
```

Scilab code Exa 11.16 Maximum power transfer

```
clc
disp("Problem 11.16")
printf("\n")

printf("Given")
disp("Vg=100V(rms)")
disp("Zg=1+i Z1=2")
Vg=100;

disp("a)")
Zg=1+%i;
```

```
Z1=2
Z=Z1+Zg
Zmag=sqrt(real(Z)^2+imag(Z)^2)
I=Vg/Zmag
PZ1=real(Z1)*(I^2)
Pg=real(Zg)*(I^2)
PT=PZ1+Pg
printf("PZ=%dW\n Pg=%dW\n PT=%dW\n",PZ1,Pg,PT);

disp("b)")
//If Z2=a+i*b
//Zg*=1-i
//Given that
//(Z1*Z2)/(Z1+Z2)=1-i
//As Z1=2 and solving for Z2
  disp(-%i,"Z2=")

disp("c)")
//If Z2 is taken the value as calculated in b) then
   Z=1-i
Zg=1+%i;
Z1=2;
Z=1-%i;
Zt=Z+Zg
Zmag=sqrt(real(Zt)^2+imag(Zt)^2)
I=Vg/Zmag
PZ=real(Z)*(I^2)
Pg=real(Zg)*(I^2)
//To calculate PZ1 and PZ2 we need to first
   calculate IZ1 nad IZ2
VZ=I*(1-%i)
IZ1=VZ/Z1
IZ1mag=sqrt(real(IZ1)^2+imag(IZ1)^2)
PZ1=real(Z1)*(IZ1mag^2)
PZ2=PZ-PZ1
PT=PZ1+PZ2+Pg
printf("PZ=%dW\n Pg=%dW\n PT=%dW\n",PZ,Pg,PT);
```

Scilab code Exa 11.17 Superposition of average powers

```
clc
disp("Problem 11.17")
printf("\n")

printf("Given")
disp("v1=5*cos(w1*t)   v2=10*cos(w2*t+60)")
//The circuit is modeled as
disp("Resistance is 10ohm and inductance is 5mH")
R=10;L=5*10^-3;
//Let V be phasor voltage between the terminals
Vmag=10;
Vph=60;
x=Vmag*cos((Vph*%pi)/180);
y=Vmag*sin((Vph*%pi)/180);
z=complex(x,y)

disp("a)")
w1=2000;w2=4000;
//Let Z be the impedance of the coil
Z1=R+%i*L*w1
Z2=R+%i*L*w2
V1=5;
//By applying superposition i=i1-i2
I1=V1/Z1
[R1,Theta]=polar(I1)
printf("i1=%0.2f*cos(%dt%d deg)\n",R1,w1,(Theta*180)/%pi);
P1=(R*R1^2)/2

V2mag=10;V2ph=60;
I2=z/Z2
[R2,Theta1]=polar(I2)
```

```
printf("i2=%0.2f*cos(%dt%3.2f deg)\n",R2,w2,(Theta1
    *180)/%pi);
P2=(R*R2^2)/2

//i=i1-i2
printf("i=%0.2f*cos(%dt%d deg)-%0.2f*cos(%dt%3.2f
    deg)\n",R1,w1,(Theta*180)/%pi,R2,w2,(Theta1*180)/
    %pi)

printf("P1=%0.3fW\nP2=%3.1fW\nTotal power(P)=%3.3fW\
    n",P1,P2,(P1+P2))

disp("b)")
//From problem 10.7
imagn=0.61
P=(R*imagn^2)/2
printf("Power dissipated in the coil=%3.3fW\n",P)

disp("c)")
w1=2000;w2=1414;
//Let Z be the impedance of the coil
Z1=R+%i*L*w1
Z2=R+%i*L*w2
V1=5;
//By applying superposition i=i1-i2
I1=V1/Z1
[R1,Theta]=polar(I1)
printf("i1=%0.2f*cos(%dt%d deg)\n",R1,w1,(Theta*180)
    /%pi);
P1=(R*R1^2)/2

V2mag=10;V2ph=60;
x1=V2mag*cos((V2ph*%pi)/180);
y1=V2mag*sin((V2ph*%pi)/180);
z1=complex(x1,y1)
I3=z1/Z2
[R3,Theta3]=polar(I3)
printf("i2=%0.2f*cos(%dt+%3.2f deg)\n",R3,w2,(Theta3
```

```
    *180)/%pi);
P3=(R*R3^2)/2

//i=i1-i2
printf("i=%0.2f*cos(%dt%d deg)-%0.2f*cos(%dt+%3.2f
    deg)\n",R1,w1,(Theta*180)/%pi,R3,w2,(Theta3*180)/
    %pi)

printf("P1=%0.3fW\nP2=%3.1fW\nTotal power(P)=%3.3fW\
    n",P1,P3,(P1+P3))
```

Chapter 12

Polyphase Circuits

Scilab code Exa 12.2 Balanced Delta connected load

```
clc
disp("Example 12.2")
printf("\n")

printf("Given")
disp("The system ABC is DELTA connected")
disp("Effective line voltage is 120V")
disp("The three impedances are 5(45 deg)")
Zmag=5;Zph=45;
//Let maximum line voltage is Vmax
Vmax=120*sqrt(2)
//From fig 12.7(a)
//VAB=Vmax(120 deg)
//VBC=Vmax(0 deg)
//VCA=Vmax(240 deg)

//From figure 12.8
IABmag=Vmax/Zmag
IABph=120-Zph
printf("IAB=%3.2f(%d deg)\n",IABmag,IABph);

```

```
IBCmag=Vmax/Zmag
IBCph=0-Zph
printf("IBC=%3.2f(%d deg)\n",IBCmag,IBCph);

ICAmag=Vmax/Zmag
ICAph=240-Zph
printf("ICA=%3.2f(%d deg)\n",ICAmag,ICAph);

//Applying KCL equation
//IA=IAB+IAC
//IB=IBC+IBA
//IC=ICA+ICB

x=IABmag*cos((IABph*%pi)/180);
y=IABmag*sin((IABph*%pi)/180);
z=complex(x,y)

x1=ICAmag*cos((ICAph*%pi)/180);
y1=ICAmag*sin((ICAph*%pi)/180);
z1=complex(x1,y1)

x2=IBCmag*cos((IBCph*%pi)/180);
y2=IBCmag*sin((IBCph*%pi)/180);
z2=complex(x2,y2)

IA=z-z1;
[RA,ThetaA]=polar(IA)

IB=z2-z;
[RB,ThetaB]=polar(IB)

IC=z1-z2
[RC,ThetaC]=polar(IC)

disp("Therefore")

printf("\nIA=%3.2f(%d deg)A\n",RA,ThetaA*(180/%pi));
printf("\nIB=%3.2f(%d deg)A\n",RB,ThetaB*(180/%pi));
```

```
printf("\nIC=%3.2f(%d deg)A\n",RC,ThetaC*(180/%pi));
```

Scilab code Exa 12.5 Unbalanced Delta connected load

```
clc
disp("Example 12.5")
printf("\n")

printf("Given")
disp("The system ABC is DELTA connected")
disp("Maximum line voltage is 339.4V")
disp("The three impedances are 10(0 deg),10(30 deg)
    ,15(-30 deg)")

ZABmag=10;ZABph=0;
ZBCmag=10;ZBCph=30;
ZCAmag=15;ZCAph=-30;
//Let maximum line voltage is Vmax
Vmax=339.4
//From fig 12.7(a)
//VAB=Vmax(120 deg)
//VBC=Vmax(0 deg)
//VCA=Vmax(240 deg)

//From figure 12.15
IABmag=Vmax/ZABmag
IABph=120-ZABph
printf("IAB=%3.2f(%d deg)\n",IABmag,IABph);

IBCmag=Vmax/ZBCmag
IBCph=0-ZBCph
printf("IBC=%3.2f(%d deg)\n",IBCmag,IBCph);

ICAmag=Vmax/ZCAmag
ICAph=240-ZCAph
```

```
printf("ICA=%3.2f(%d deg)\n",ICAmag,ICAph);

//Applying KCL equation
//IA=IAB+IAC
//IB=IBC+IBA
//IC=ICA+ICB

x=IABmag*cos((IABph*%pi)/180);
y=IABmag*sin((IABph*%pi)/180);
z=complex(x,y)

x1=ICAmag*cos((ICAph*%pi)/180);
y1=ICAmag*sin((ICAph*%pi)/180);
z1=complex(x1,y1)

x2=IBCmag*cos((IBCph*%pi)/180);
y2=IBCmag*sin((IBCph*%pi)/180);
z2=complex(x2,y2)

IA=z-z1;
[RA,ThetaA]=polar(IA)

IB=z2-z;
[RB,ThetaB]=polar(IB)

IC=z1-z2
[RC,ThetaC]=polar(IC)

disp("Therefore")

printf("\nIA=%3.2f(%d deg)A\n",RA,ThetaA*(180/%pi));
printf("\nIB=%3.2f(%d deg)A\n",RB,ThetaB*(180/%pi));
printf("\nIC=%3.2f(%d deg)A\n",RC,ThetaC*(180/%pi));
```

Scilab code Exa 12.6 Unbalanced Wye connected load

```
clc
disp("Example 12.6")
printf("\n")

printf("Given")
disp("The system CBA is WYE connected")
disp("Maximum line voltage is 150V")
disp("The three impedances are 6(0 deg),6(30 deg)
    ,5(45 deg)")
ZAmag=6;ZAph=0;
ZBmag=6;ZBph=30;
ZCmag=5;ZCph=45;
//Let maximum line voltage is Vmax
Vmax=150
//Let the line to neutral voltage magnitude be Vn
Vn=Vmax/sqrt(3)
//From fig 12.7(b)
//VAN=Vn(-90 deg)
//VBN=Vn(30 deg)
//VCN=Vn(150 deg)

//From figure 12.16
IAmag=Vn/ZAmag
IAph=-90-ZAph
printf("\nIA=%3.2f(%d deg)A\n",IAmag,IAph);

IBmag=Vn/ZBmag
IBph=30-ZBph
printf("\nIB=%3.2f(%d deg)A\n",IBmag,IBph);

ICmag=Vn/ZCmag
ICph=150-ZCph
printf("\nIC=%3.2f(%d deg)A\n",ICmag,ICph);

//Now to calculate IN
//IN=-(IA+IB+IC)
x=IAmag*cos((IAph*%pi)/180);
y=IAmag*sin((IAph*%pi)/180);
```

```
z=complex(x,y)

x1=ICmag*cos((ICph*%pi)/180);
y1=ICmag*sin((ICph*%pi)/180);
z1=complex(x1,y1)

x2=IBmag*cos((IBph*%pi)/180);
y2=IBmag*sin((IBph*%pi)/180);
z2=complex(x2,y2)

IN=-(z+z1+z2)

[R,Theta]=polar(IN)

printf("\nIN=%3.2f(%d deg)A\n",R,Theta*(180/%pi));
```

Chapter 13

Frequency Response Filters and Resonance

Scilab code Exa 13.2 `High pass and Low pass networks`

```
clc
disp("Problem 13.2")
printf("\n")

printf("Given")
disp("|Hv|=1/sqrt(2)                    (1)")
disp("Resistance R1=5kohm")
R1=5000;
disp("Hv(w)=1/1+%i*(w/wx)               (2)")
//wx=1/(R1*C2)
//On solving we get
disp("wx=2*10^-4/C2                     (3)")

disp("a)")
C2=10*10^-9;
//Taking modulus of (2)
disp("|Hv(w)|=1/sqrt(1+(w/wx)^2)")
//Equating (1) and (2)
wx=2*10^-4/C2;
```

```
fx=(wx/(2*%pi))*10^-3
printf("Frequency(a) is %3.2fkHz\n",fx)

disp("b)")
C2b=1*10^-9;
//As frequency is inversely proportional to C2 (from (3))
fx1=(C2/C2b)*fx
printf("Frequency(b) is %3.2fkHz\n",fx1)
```

Scilab code Exa 13.7 Bandpass filters and Resonance

```
clc
disp("Problem 13.7")
printf("\n")

s=%s;
printf("Given")
H=(10*s)/(s^2+300*s+10^6)
disp(H,"H(s)=")
//From the above transfer function
//Comparing the denominator with s^2+a*s+b with w=sqrt(b)
a=300;b=10^6;
//Therefore center frequency is
w0=sqrt(10^6)
//The lower and upper frequencies are
wl=sqrt(a^2/4+b)-a/2
wh=sqrt(a^2/4+b)+a/2
B=wh-wl          //It can be inferred that B=a
Q=sqrt(b)/a
printf("\nCenter frequency= %drad/s\n",w0);
printf("Low power frequency = %3.2frad/s\nHigh power frequency = %3.2frad/s\n",wl,wh);
printf("Bandwidth= %drad/s\nQuality factor =%3.2f\n"
```

```
,B,Q)
```

Scilab code Exa 13.8 Bandpass filters and Resonance

```
clc
disp("Problem 13.8")
printf("\n")

s=%s;
printf("Given")
H=(10*s)/(s^2+30*s+10^6)
disp(H,"H(s)=")
//From the above transfer function
//Comparing the denominator with s^2+a*s+b with w=sqrt(b)
a=30;b=10^6;
//Therefore center frequency is
w0=sqrt(10^6)
//The lower and upper frequencies are
wl=sqrt(a^2/4+b)-a/2
wh=sqrt(a^2/4+b)+a/2
B=wh-wl
Q=sqrt(b)/a
printf("\nCenter frequency= %drad/s\n",w0);
printf("Low power frequency = %3.2frad/s\nHigh power frequency = %3.2frad/s\n",wl,wh);
printf("Bandwidth= %drad/s\nQuality factor =%3.2f\n",B,Q)
```

Chapter 14

Two Port Networks

Scilab code Exa 14.1 Z parameters

```
clc
disp("Example 14.1")
printf("\n")

s=%s;
//Applying KVL equation to the two loops we get
//V1=2*I1+s*(I1+I2)
//V2=3*I2+s*(I1+I2)

//On solving we get
disp("(s+2)*I1+s*I2=V1          (1)");
disp("s*I1+(s+3)*I2=V2          (2)");

//The equations which contain Z parameters are
//V1=Z11*I1+Z12*I2
//V2=Z21*I1+Z22*I2

//On comparing (1) and (2) with above equations
Z11=s+2;
Z12=s;
Z21=s;
```

```
Z22=s+3;

disp(Z11,"Z11=")
disp(Z12,"Z12=")
disp(Z21,"Z21=")
disp(Z22,"Z22=")
```

Scilab code Exa 14.2 Reciprocal and non reciprocal networks

```
clc
disp("Example 14.2")
printf("\n")

s=%s;
//Applying KVL equation to the two loops we get
//V1=2*I1+s*(I1+I2)-I2
//V2=3*I2+s*(I1+I2)

//On solving we get
disp("(s+2)*I1+(s-1)*I2=V1      (1)");
disp("s*I1+(s+3)*I2=V2          (2)");

//The equations which contain Z parameters are
//V1=Z11*I1+Z12*I2
//V2=Z21*I1+Z22*I2

//On comparing (1) and (2) with above equations
Z11=s+2;
Z12=s-1;
Z21=s;
Z22=s+3;

disp(Z11,"Z11=")
disp(Z12,"Z12=")
disp(Z21,"Z21=")
```

```
disp(Z22,"Z22=")
```

Scilab code Exa 14.4 Y parameters

```
clc
disp("Example 14.4")
printf("\n")

s=%s;
Ya=3/(5*s+6);
Yb=2/(5*s+6);
Yc=s/(5*s+6);

//Writing KCL equations
disp("I1=(Ya+Yc)*V1-Yc*V2          (1)")
disp("I2=-Yc*V1+(Yb+Yc)*V2         (2)")

//The equations which contain Y parameters are
//I1=Y11*V1+Y12*V2
//I2=Y21*V1+Y22*V2

//On comparing (1) and (2) with above equations
disp("Y11=Ya+Yc")
disp("Y12=-Yc=Y21")
disp("Y22=Yb+Yc")

//Substituting Ya , Yb and Yc
Y11=Ya+Yc
Y12=-Yc
Y21=-Yc
Y22=Yb+Yc

disp(Y11,"Y11=")
disp(Y12,"Y12=")
disp(Y21,"Y21=")
```

```
disp(Y22,"Y22=")
```

Scilab code Exa 14.6 Conversion between Z and Y parameters

```
clc
disp("Example 14.6")
printf("\n")

s=%s;
//From example 14.4

Y11=(3 + s)/(5*s+6)
Y12=- s/(6 + 5*s)
Y21=- s/(6 + 5*s)
Y22=(2+s)/(6+5*s)

DYY=Y11*Y22-Y12*Y21

Z11=Y22/DYY;
Z12=-Y12/DYY;
Z21=-Y21/DYY;
Z22=Y11/DYY;

disp(Z11,"Z11=")
disp(Z12,"Z12=")
disp(Z21,"Z21=")
disp(Z22,"Z22=")
```

Scilab code Exa 14.7 H parameters

```
clc
disp("Example 14.7")
printf("\n")
```

```

//From figure 14.9
disp("V1=50*I1          (1)");
disp("I2=300*I1         (2)");

//The equations which contain h parameters are
//V1=h11*I1+h12*V2
//I2=h21*I1+h22*V2

//On comparing (1) and (2) with above equations

printf("\nh11=%d\n",50);
printf("h12=%d\n",0);
printf("h21=%d\n",300);
printf("h22=%d\n",0);
```

Scilab code Exa 14.8 g parameters

```
clc
disp("Example 14.8")
printf("\n")

//From figure 14.10
//By inspection
//V1=10^9*I1
//V2=10(I2-10^-3*V1)


//On solving we get
disp("I1=10^-9*V1                 (1)");
disp("V2=10*I2-10^-2*V1           (2)");


//The equations which contain g parameters are
//I1=g11*V1+g12*I2
```

```
//V2=g21*V1+g22*I2

//On comparing (1) and (2) with above equations

printf("\ng11=%2.1e\n",10^-9);
printf("g12=%d\n",0);
printf("g21=%2.1e\n",-10^-2);
printf("g22=%d\n",10);
```

Scilab code Exa 14.10 Choice of parameter type

```
clc
disp("Example 14.10")
printf("\n")

s=%s;
//Applying KVL equation to the two loops we get
//V1=3*I1+3*(I1+I2)
//V2=7*I1+3*(I1+I2)+2*I2

//On solving we get
disp("6*I1+3*I2=V1           (1)");
disp("10*I1+5*I2=V2          (2)");

//The equations which contain Z parameters are
//V1=Z11*I1+Z12*I2
//V2=Z21*I1+Z22*I2

//On comparing (1) and (2) with above equations
Z11=6;
Z12=3;
Z21=10;
Z22=5;

disp(Z11,"Z11=")
```

```
disp(Z12,"Z12=")
disp(Z21,"Z21=")
disp(Z22,"Z22=")

disp("As DZZ results in zero(0) therefore Y
    parameters are not defined ")
```

Chapter 15

Mutual Inductance and Transformers

Scilab code Exa 15.4 Energy in a pair of coupled coils

```
clc
disp("Example 15.4")
printf("\n")

printf("Given")
disp("L1=0.1H L2=0.2H")
disp("i1=4A i2=10A")
L1=0.1;L2=0.2
i1=4;i2=10;
//The energy stored in coupled coils is
disp("W=(L1*i1^2)/2+(L2*i2^2)/2+M*i1*i2")

disp("a)")
M=0.1;
W=(L1*i1^2)/2+(L2*i2^2)/2+M*i1*i2;
printf("Total Energy in the coils=%3.2fJ\n",W);

disp("b)")
M=sqrt(2)/10;
```

```
W=(L1*i1^2)/2+(L2*i2^2)/2+M*i1*i2;
printf("Total Energy in the coils=%3.2fJ\n",W);

disp("c)")
M=-0.1;
W=(L1*i1^2)/2+(L2*i2^2)/2+M*i1*i2;
printf("Total Energy in the coils=%3.2fJ\n",W);

disp("a)")
M=-sqrt(2)/10;
W=(L1*i1^2)/2+(L2*i2^2)/2+M*i1*i2;
printf("Total Energy in the coils=%3.2fJ\n",W);
```

Scilab code Exa 15.7 Ampere Turn Dot rule

```
clc
disp("Example 15.7")
printf("\n")

printf("Given")
disp("N1=20 N2=N3=10")
disp("I2=10(-53.13 deg) I3=10(-45 deg)")
N1=20;N2=10;N3=10;
I2mag=10;I2ph=-53.13;
I3mag=10;I3ph=-45;
//From figure 15.14
disp("N1*I1-N2*I2-N3*I3=0")
//Solving for I1
Xmag=N2*I2mag
Xph=I2ph
x=Xmag*cos((Xph*%pi)/180);
y=Xmag*sin((Xph*%pi)/180);
z=complex(x,y)

Ymag=N3*I3mag
```

```
Yph=I3ph
x1=Ymag*cos((Yph*%pi)/180);
y1=Ymag*sin((Yph*%pi)/180);
z1=complex(x1,y1)

I1=(z+z1)/N1
[R,Theta]=polar(I1);
printf("I1=%3.2f(%3.2f deg) A\n",R,(Theta*180)/%pi);
```

Scilab code Exa 15.8 Reflected Impedance

```
clc
disp("Example 15.8")
printf("\n")

printf("Given")
disp("L1=0.2H L2=0.1H")
disp("M=0.1H R=10ohm")
disp("v1=142.3*sin(100*t)")
L1=0.2;L2=0.1
M=0.1;R=10;
v1mag=142.3;
w=100;
//Let Input impedance be Z1 and can be calculated as
//From the equations in 15.10
disp("Z1=%i*w*L1+((M*w)^2)/(Z2+%i*w*L2)")
Z1=%i*w*L1+((M*w)^2)/(R+%i*w*L2)
[R,Theta]=polar(Z1)
//If I1 is the input current
I1mag=v1mag/R
I1ph=-(Theta*180)/%pi
//In time domain form
printf("i1=%3.1f*sin(%d*t%3.1f deg) (A)",I1mag,w,
   I1ph);
```

Scilab code Exa 15.9 `Reflected Impedance`

```
clc
disp("Example 15.9")
printf("\n")

s=%s;
printf("Given")
disp("L1=0.2H L2=0.1H")
disp("M=0.1H R=10ohm")
disp("v1=u(t) a unit step function")
L1=0.2;L2=0.1
M=0.1;R=10;
v1=1;
w=100;
//Let Input impedance be Z1 and can be calculated as
//From the equations in 15.10
disp("Z1(s)=L1*s-((M*s)^2)/(R+L2*s)")
Z1=L1*s-(((M*s)^2)/(R+L2*s))
//Proper rearranging of co-efficients
Num=Z1('num')/0.01
Den=Z1('den')*100

disp(Num/Den,"Z1(s)")
Y1=1/Z1
disp(Den/Num,"Y1(s)")

//As the input is unit step function the value is 1V for t>0
//In exponential form the value is represented as exp(s*t) with s=0 as the pole of Y1(s)

//Therefore forced response
k=1/L1;
```

```
printf("Forced response i1,f=(%d*t) (A)\n",k);
```

Chapter 17

The Laplace Transform Method

check Appendix AP 2 for dependency:

ch17_2.sce

Scilab code Exa 17.2 Convergence of the integral

```
syms t s ;
 x=laplace ('3*%e^(2*t)' , t , s ) ;
 disp (x , " X(s)=" )
```

---

check Appendix AP 1 for dependency:

ch17_4.sce

Scilab code Exa 17.4 Partial Fractions Expansions

```
clc
syms t
s=%s;
//Factorizing the denominator
I=(s-10)/((s^2)*(s-%i)*(s+%i));
```

```
disp(I,"I(s)=")
//The principal part at s=0 is
//B1/s+B2/s^2
//Taking the limit s->0 to (s-10)/((s-%i)*(s+%i))

B2=-10

//Taking the limit s->0 to (s*(s-10))/(s^2)*(s^2+1)+(10/s)

B1=1

//The principal part at s=%i is
//A/(s-%i)
//Taking the limit s->%i to (s-10)/((s^2)*(s+%i))

A=(-0.5-%i*5)

//As the other co-efficient is conjugate of the above we can write the partial fraction expansion of I(s)
I=(1/s)-(10/s^2)-(0.5+%i*5)/(s-%i)-(0.5-%i*5)/(s+%i);
//Taking inverse of each term
I1=ilaplace('1/s',s,t)
I2=ilaplace('10/s^2',s,t)
I3=ilaplace('(0.5+%i*5)/(s-%i)',s,t)
I4=ilaplace('(0.5-%i*5)/(s+%i)',s,t)
I=I1-I2-I3-I4
disp(I,"i(t)=")
```

Appendix

Scilab code AP 1 Partial Fractions Expansions

```
clc
syms t
s=%s;
//Factorizing the denominator
I=(s-10)/((s^2)*(s-%i)*(s+%i));
disp(I,"I(s)=")
//The principal part at s=0 is
//B1/s+B2/s^2
//Taking the limit s->0 to (s-10)/((s-%i)*(s+%i))

B2=-10

//Taking the limit s->0 to (s*(s-10))/(s^2)*(s^2+1)+(10/s)

B1=1

//The principal part at s=%i is
//A/(s-%i)
//Taking the limit s->%i to (s-10)/((s^2)*(s+%i))

A=(-0.5-%i*5)

//As the other co-efficient is conjugate of the above we can write the partial fraction expansion of I(s)
```

```
I=(1/s)-(10/s^2)-(0.5+%i*5)/(s-%i)-(0.5-%i*5)/(s+%i)
    ;
//Taking inverse of each term
I1=ilaplace('1/s',s,t)
I2=ilaplace('10/s^2',s,t)
I3=ilaplace('(0.5+%i*5)/(s-%i)',s,t)
I4=ilaplace('(0.5-%i*5)/(s+%i)',s,t)
I=I1-I2-I3-I4
disp(I,"i(t)=")
```

Scilab code AP 2 Convergence of the integral

```
syms t s ;
 x=laplace ('3*%e^(2*t)' , t , s ) ;
 disp (x , " X(s)=" )
```