

Scilab Manual for  
Electronic Design and Automation  
by Prof Kavya Manohar  
Electronics Engineering  
Aryanet Institute of Technology<sup>1</sup>

Solutions provided by  
Mr NILESH GUPTA  
Others  
Mumbai University/PVPPCOE

January 31, 2025

<sup>1</sup>Funded by a grant from the National Mission on Education through ICT, <http://spoken-tutorial.org/NMEICT-Intro>. This Scilab Manual and Scilab codes written in it can be downloaded from the "Migrated Labs" section at the website <http://scilab.in>



# Contents

List of Scilab Solutions	3
1 Solve quadratic equations containing complex numbers and matrix multiplication	5
2 Generate and plot various signals like sine square, pulse in same window.	7
3 Obtain different types of plots (2D/3D, surface plot, polar plot)	9
4 Plot the diode/transistor characteristics.	11
5 Solve node, mesh and loop equations of simple electrical/network circuits.	13
6 Find the poles and zeros hence plot the transfer functions/polynomials	15
7 Read floating point numbers from a formatted file and sort it in ascending order and save in another text file	17
8 Plot a full wave rectified waveform using Fourier series.	19

# List of Experiments

Solution 1.0	Experiment number 1 . . . . .	5
Solution 2.0	Experiment number 2 . . . . .	7
Solution 3.0	Experiment number 3 . . . . .	9
Solution 4.0	Experiment number 4 . . . . .	11
Solution 5.0	Experiment Number 5 . . . . .	13
Solution 6.0	Experiment number 6 . . . . .	15
Solution 7.0	Experiment Number 7 . . . . .	17
Solution 8.0	Experiment Number 8 . . . . .	19
AP 1	text file for Experiment number 7 . . . . .	21
AP 2	Circuit Diagram For Experiment 5 . . . . .	22

# List of Figures

# Experiment: 1

## Solve quadratic equations containing complex numbers and matrix multiplication

Scilab code Solution 1.0 Experiment number 1

```
1 //AIM: Solve quadratic equations containing complex
   numbers & matrix
2 //multiplications
3 //Software version Scilab 5.5.2
4 //OS Windows 7
5
6 clc;
7 clear;
8
9 //consider quadratic equation  $a(x^2)+(b*x)+c=0$  to be
   solved.
10 x=poly(0,"x")
11 a=input('Enter the coefficients of quadratic
   equation(Eg: To solve  $a(x^2)+(b*x)+c=0$ , enter [a
   b c]) ');
12 X=[x^2 x 1]
13 h=a*X'
```

```

14 //h=a*(x^2)+(b*x)+c
15 disp(h)
16 root1=(-a(2))+sqrt((a(2)^2)-4*a(1)*a(3))/(2*a(1))
17 disp(root1,"1st root of the equation is =")
18 root2=(-a(2))-sqrt((a(2)^2)-4*a(1)*a(3))/(2*a(1))
19 disp(root2,"2nd root of the equation is")
20
21 //Solving Linear equations in two variables
22 //let two equations be
23 //1: a1.x+b1.y=c1
24 //2: a2.x-b2.y=c2
25 C=input('Enter the coefficients of first linear
           equation in two variables (a1*x+b1*y=c1 may be
           entered as [a1 b1 -c1]: ')
26 D=input('Enter the coefficients of second linear
           equation in two variables (a2*x+b2*y=c2 may be
           entered as [a2 b2 -c2]: ')
27 a=[C(1:2);D(1:2)];
28 d=[C(3);D(3)];
29 X=linsolve(a,d);
30 mprintf("X=%f\n",X);

```

---

## Experiment: 2

Generate and plot various signals like sine square, pulse in same window.

Scilab code Solution 2.0 Experiment number 2

```
1 //Aim:Generate and plot various signals like sine ,
   square ,pulse in same window.
2 //Software version Scilab 5.5.2
3 //OS Windows 7
4
5 clc;
6 clear;
7
8 t=-10:0.05:10;
9 a=1;
10 b=t;
11 //To plot the sine wave
12 x1=sin(t);
13 subplot(3,1,1)
14 plot(t,x1)
15 xlabel('time')
16 ylabel('x1[t]')
```



```
17 title('sine wave')
18
19 //To plot the square wave
20 x2=squarewave(t);
21 subplot(3,1,2)
22 plot(t,x2)
23 xlabel('time')
24 ylabel('x2[t]')
25 title('square wave')
26
27 //To plot the impulse function
28 x3=a*(t==0)
29 subplot(3,1,3)
30 plot(t,x3)
31 xlabel('time')
32 ylabel('x3[t]')
33 title('impulse function')
```

---

## Experiment: 3

### Obtain different types of plots (2D/3D, surface plot, polar plot)

Scilab code Solution 3.0 Experiment number 3

```
1 //Aim:Obtain different types of plots (2D,surface
   plot ,polar plot).
2 //Software version Scilab 5.5.2
3 //OS Windows 7
4
5 clear;
6 clc;
7 //plot for 2D
8 figure()
9 x =-10: 0.1 : 10;// Define your x-values
10 y =cos(x).*sinc(x);// Define your function
11 plot(x, y)
12 title('2D plot')
13 xlabel('x axis');
14 ylabel('y axis');
15 legend('y =cos(x).*sinc(x)');
16
```

```
17 //plot for polar plot
18 figure()
19 t = linspace(0, 2*pi, 100);
20 r =sqrt(abs(2 * sinc(5*t)));
21 title('polar plot')
22 polarplot(t, r);
23
24 //plot for surface plot
25 figure()
26 t=[0:0.3:2*pi]';
27 z=sin(t)*cos(t');
28 xlabel('x axis');
29 ylabel('y axis');
30 zlabel('z axis');
31 title('surface plot');
32 plot3d(t,t,z)
```

---

# Experiment: 4

## Plot the diode/transistor characteristics.

Scilab code Solution 4.0 Experiment number 4

```
1 //AIM:Plot the diode/transistor characteristics
2 //Software version Scilab 5.5.2
3 //OS Windows 7
4
5 clear;
6 clc;
7
8 Vd=0:0.1:1//diode voltage
9 Is=10^(-15)//assume reverse biased saturation
   current
10 n=1//Emission coefficient
11 Temp=32//temperature in degrees
12 T=Temp+273//temperature in Kelvin
13 k=1.38*10^(-23)//Boltzmann's Constant
14 q=1.6*10^(-19)//Charge of electron
15 Vt=(k*T)/q//thermal voltage
16 Id=Is*(exp(Vd/(n*Vt))-1)//Diode current
17 subplot(1,2,1)
18 plot(Vd,Id)
```

```
19 xlabel("Vd in volts")
20 ylabel("Id in mA")
21 title("Characteristic of diode when forward biased")
```

---

## Experiment: 5

# Solve node, mesh and loop equations of simple electrical/network circuits.

Scilab code Solution 5.0 Experiment Number 5

```
1 //Aim: Solve node, mesh and loop equations of simple
   electrical/network circuits.
2 //Software version : Scilab 5.5.2
3 //OS: Windows 7
4 //Toolbox: SIVP_0.5.3.2
5
6 clc;
7 clear;
8
9 //**PLEASE REFER THE DEPENDENCY FILE i.e. the
   circuit **
10
11 //Mesh/loop analysis
12
13 V=10//in volts
14 R1=1000//in ohms
15 R3=1000//in ohms
```

```

16 R4=2000 //in ohms
17 R2=2000 //in ohms
18 R22=(R2*R4)/(R2+R4)
19 disp(" Ohms" ,R22 , " R22=")
20 //10==(I*R1)+(I*R2)+(I*R3)
21 I=V/(R1+R22+R3)
22 disp(" Ampere" ,I , " I=")
23 I1=I*(R2/(R4+R2))
24 disp(" Ampere" ,I1 , " I1=")
25
26 //Nodal analysis
27 //((Va-10)/(1000+1000))+(Va/2000)+(Va/2000)==0
28 Va=(V/(R1+R3))/((1/R2)+(1/R4)+(1/(R1+R3)))
29 disp(" Volts" ,Va , " Va=")

```

---

check Appendix [AP 2](#) for dependency:

CircuitForExp5.jpg

## Experiment: 6

Find the poles and zeros hence plot the transfer functions/polynomials

Scilab code Solution 6.0 Experiment number 6

```
1 //AIM:Find the poles and zeros hence plot the
   transfer functions/polynomials
2 //Software version Scilab 5.5.2
3 //OS Windows 7
4
5 clear;
6 clc;
7
8 //Consider the transfer function
9 //T.F=(s*(s+2))/(s^2+2*s+2)
10
11 x=poly(0,"x")
12 a1=input("Enter the value of coefficient of a=");
13 b1=input("Enter the value of coefficient of b=");
14 c1=input("Enter the value of coefficient of z=");
15 zero1=(-b1)+sqrt((b1^2)-4*a1*c1)/(2*a1)
16 printf("1st zero = %g\n",zero1)
```



```

17 zero2=(-b1)-sqrt((b1^2)-4*a1*c1)/(2*a1)
18 printf("2nd zero = %g\n\n",zero2)
19
20 y=poly(0,"y")
21 a2=input("Enter the value of coefficient of a=");
22 b2=input("Enter the value of coefficient of b=");
23 c2=input("Enter the value of coefficient of z=");
24
25 root1=(-b2)+sqrt((b2^2)-4*a2*c2)/(2*a2)
26 disp(root1,"1st pole =")
27 root2=(-b2)-sqrt((b2^2)-4*a2*c2)/(2*a2)
28 disp(root2,"2nd pole =")
29 disp('Transfer function = (s*(s+2))/(s^2+2*s+2)')

```

---

## Experiment: 7

**Read floating point numbers from a formatted file and sort it in ascending order and save in another text file**

Scilab code Solution 7.0 Experiment Number 7

```
1 //Aim:Read floating point numbers from a formatted
   file and sort it in ascending
2 // order and save in another text file
3 //Software version Scilab 5.5.2
4 //OS Windows 7
5
6 clear;
7 clc;
8 fid=mopen("C:\Program Files\scilab-5.5.2\contrib\
   text.txt","r");//reading the file
9 if(fid==-1)
10     error("cannot open file\n")
11 else
12     printf("Scilab can read selected file\n")
13 end
```

```

14 num_yet=0
15 done_yet=0
16 while(done_yet==0)
17     [num_read, val(1), val(2), val(3), val(4)]=mfscanf(
18         fid, "%f%f%f%f")
19     if(num_yet <=0)
20         done_yet=1;
21 end
22 file("close", fid)
23 printf("The floating point numbers from a formatted
24     file\n")
25 disp(val)
26 l=length(val)
27 printf("Values in ascending order copied in another
28     text file with name new in same directory\n")
29 for i=1:(l-1)
30     for j=1:(l-1)
31         if(val(j)>val(j+1))
32             temp=val(j)
33             val(j)=val(j+1)
34             val(j+1)=temp
35         end
36     end
37 disp(val)
38 fis=mopen("new.txt", "w") //
39     creating the new file
40 mfprintf(fis, "%f\n%f\n%f\n%f", val(1), val(2), val(3),
41     val(4))
42 file("close", fis)

```

---

check Appendix [AP 1](#) for dependency:

text.txt

# Experiment: 8

## Plot a full wave rectified waveform using Fourier series.

Scilab code Solution 8.0 Experiment Number 8

```
1 //Aim:Plot a full wave rectified waveform using
   Fourier series
2 //Software version Scilab 5.5.2
3 //OS Windows 7
4
5 clc;
6 clear;
7 funcprot(0); //creating funtion
   for Fourier series
8 function y=f(x),y=1/%pi*sin(x),endfunction
   //calculating a0
9   a0=intg(0,%pi,f) //integrating from 0 to
   %pi for calculating a0
10 function [a0,A]=fourier(l,n,f)
11   for i=1:n
12     function b=f1(x,f)
13       b=f(x)*cos(i*%pi*x/l)*sin(i*%pi*x/l);
   //declaring function An
14   endfunction
```

```

15         A(i)=1/l*intg(0,%pi,f1);           //
           calculating value of An
16     end
17     function series=solution(x)
18         series=a0/2
19         for i=1:n
20             series=series+A(i)*cos(i*%pi*x/l); //
           suming the serice
21         end
22     endfunction
23     x=0:0.1:6*l;
24     plot2d(x,10*(solution(x)-0.068));
25     title("output voltage of full wave rectifier")
26     ylabel("amplitude")
27     xlabel("time")
28 endfunction
29 deff('a=f(x)', 'a=x')
30 [a0,A]=fourier(%pi,1,f) //calling function by
    asinging value of l and n

```

---

# Appendix

Scilab code AP 11 33.22

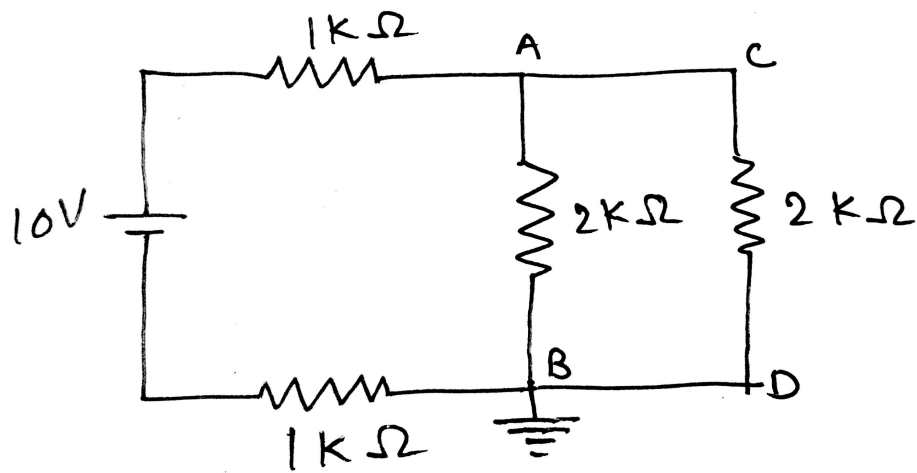
2 462.

3 27.11

4 383.22

text file for Experiment number 7

---



Circuit Diagram For Experiment 5

Cir-