Scilab Textbook Companion for Signals And Systems by V. Krishnaveni And A. Rajeswari[1]

Created by
S D Vinodhini
Bachelor of Engineering
Electronics Engineering
PSG College of Technology
College Teacher
V. Krishnaveni
Cross-Checked by
Bhavani

May 31, 2016

[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: Signals And Systems

Author: V. Krishnaveni And A. Rajeswari

Publisher: Wiley India Private Limited, New Delhi-110 002

Edition: 1

Year: 2012

ISBN: 9788126522897

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

Chapter 1

Overview of Signals and systems

Scilab code Exa 1.1 Plotting a dc signal

```
clf
clc
clear
t=[-20:0.01:20];
for i=1:length(t)
    x(i)=2;
end
plot(t,x);
xtitle("x(t)=2 for all t","time","amplitude");
xgrid(5)
```

Scilab code Exa 1.2 plotting basic signals

```
clf
clear
clc
```

```
t=[-20:0.01:20];
for i=1:length(t)
    x=2*t;
end
plot(t,x);
xtitle("x(t)=2*t for all t","time","amplitude");
xgrid(5)
```

Scilab code Exa 1.3 Plotting Basic signals

```
clc
clear
clf
interval=input('enter the value of time interval T between two samples');
t=(-20*interval):interval:(20*interval);
for i=1:length(t)
    if t(i)<0 then
        x(i)=-1;
    elseif t(i)>0 then
        x(i)=1;
    else
        x(i)=0;
    end
end
plot(t,x,".");
xtitle("x(t)=1 for positive values of t..., x(t)=0 for t=0...., x(t)=-1 for negative values of t","time","amplitude");
xgrid(5)
```

Scilab code Exa 1.4 Plotting basic signals

```
clear
clf
clc
t=-20:0.01:20;
for i=1:length(t)
    if t(i)>0 then
        x1(i)=0.5;
    else
        x1(i)=-0.5;
    end
end
subplot(3,1,1)
plot(t,x1);
xtitle("x1(t)=-0.5 for t<0 and x1(t)=0.5 for t>0","
   time","amplitude");
xgrid(5);
subplot(3,1,2)
for i=1:length(t)
    x2(i)=-t(i);
end
plot(t,x2);
xtitle("x2(t)=-t for all t","time","amplitude");
xgrid(5);
subplot(3,1,3)
for i=1:length(t)
    x3(i)=t(i).^2;
end
plot(t,x3);
xtitle("x3(t)=t^2 for all t","time","amplitude");
xgrid(5);
```

Scilab code Exa 1.5 Plotting basic signals

```
clear
clf
```

```
clc
n=-20:1:20;
for i=1:length(n)
    if n(i)>=0 then
        x(i)=2;
    else
        x(i)=0;
    end
end
plot(n,x,".");
xtitle("x(n)=0 for n<0 and x(n)=2 for n>=0","number of samples","amplitude");
xgrid(5)
```

Scilab code Exa 1.6 Plotting basic signals

```
clc
clear
clf
n1=-2:1:2;
x1=-2:1:2;
subplot(3,1,1)
plot(n1,x1,".");
xtitle("x1(n)","n","x1(n)");
xgrid(5)
n=-5:1:5;
for i=1:length(n)
   x2(i)=n(i);
   x3(i)=2-n(i);
end
subplot(3,1,2);
plot(n,x2,".");
xtitle("x2(n)","n","x2(n)");
xgrid(5);
subplot(3,1,3);
```

```
plot(n,x3,".");
xtitle("x3(n)","n","x3(n)");
xgrid(5);
```

Scilab code Exa 1.7 Plotting basic signals

```
clear
clf
clc
interval=input('enter the sampling interval');
n=[-20:1:20];
t=n*interval
for i=1:length(t)
    x(i)=2*t(i);
end
plot(t,x,".");
xtitle("sampled function of x(t)=2*t for all t","
   number of samples","amplitude");
```

Scilab code Exa 1.8 Plotting basic signals

```
x=poly([-4 2 1],'t','c')
a=horner(x,0)
b=horner(x,-2)
disp(a)
disp(b)
```

Scilab code Exa 1.10 Plotting basic signals

```
clear
```

```
clf
clc
n=-20:1:20;
for i=1:length(n)
        x(i)=0.5;
end
subplot(2,1,1)
plot(n,x,".");
xtitle("x(n)=0.5 for all n","number of samples","
   amplitude");
xgrid(5)
y=0.5*x;
subplot(2,1,2)
plot(n,y,".");
xtitle("y(n)=0.5*x(n) for all n","number of samples"
   ,"amplitude");
xgrid(5)
```

Chapter 2

Continuoustime and discretetime signals

Scilab code Exa 2.1 Signal Operations

```
//Example 2.1
clf
clear
clc
t=[-10:0.01:10];
for i=1:length(t)
    if t(i)>= -0.5 & t(i)<= 0.5 then
        x(i)=t(i)+0.5;
    elseif t(i)>0.5 & t(i)<=1.5 then
            x(i)=1.5-t(i);
        else
            x(i)=0;
        end
end
subplot(3,1,1);
plot2d(t,x,rect=[-4 0 4 2]);
xtitle("x(t) vs t","t in sec","x(t)");
subplot(3,1,2);
plot2d(t-1,x,rect=[-4 0 4 2]);
```

```
xtitle("x(t+1) vs t","t in sec","x(t+1)");
subplot(3,1,3);
plot2d(t+2,x,rect=[-4 0 4 2]);
xtitle("x(t-2) vs t","t in sec","x(t-2)");
xset('window',1);
subplot(3,1,1);
plot2d(-t,x,rect=[-4 0 4 2]);
xtitle("x(-t) vs t","t in sec","x(-t)");
subplot(3,1,2);
plot2d(t/2,x,rect=[-4 0 4 2]);
xtitle("x(2t) vs t","t in sec","x(2t)");
subplot(3,1,3);
plot2d(t*2,x,rect=[-4 0 4 2]);
xtitle("x(t/2) vs t","t in sec","x(t/2)");
xset('window',2);
subplot(3,1,1);
plot2d(-t-1,x,rect=[-4 0 4 2]);
xtitle("x(-t+1) vs t","t in sec","x(-t+1)");
subplot(3,1,2);
plot2d(-t+2,x,rect=[-4 0 4 2]);
xtitle("x(-t-2) vs t","t in sec","x(-t-2)");
subplot(3,1,3);
plot2d(-t/2,x,rect=[-4 0 4 2]);
xtitle("x(-2t) vs t","t in sec","x(-2t)");
xset('window',3);
subplot(3,1,1);
plot2d(-t*2,x,rect=[-4 0 4 2]);
xtitle("x(-t/2) vs t","t in sec","x(-t/2)");
subplot(3,1,2);
plot2d(-(t-1)/2,x,rect=[-4 0 4 2]);
xtitle("x(-2t+1) vs t","t in sec","x(-2t+1)");
subplot(3,1,3);
plot2d(-(t+2)/2,x,rect=[-4 0 4 2]);
xtitle("x(-2t-2) vs t","t in sec","x(-2t-2)");
```

Scilab code Exa 2.2 Signal operations

```
//Example 2.2
clf
clear
clc
t=[-10:0.01:10];
for i=1:length(t)
    if t(i)>= -2 & t(i)<=4 then
        x(i)=(t(i)+2)/6;
        else
            x(i)=0;
        end
end
subplot(3,1,1);
plot2d(t,x,rect=[-10 0 10 2]);
xtitle("x(t) vs t","t in sec","x(t)");
subplot(3,1,2);
plot2d(t-1,x,rect=[-10 0 10 2]);
xtitle("x(t+1) vs t","t in sec","x(t+1)");
subplot(3,1,3);
plot2d(t+1,x,rect=[-10 0 10 2]);
xtitle("x(t-1) vs t","t in sec","x(t-1)");
xset('window',1);
subplot(3,1,1);
plot2d(t/2,x,rect=[-10 0 10 2]);
xtitle("x(2t) vs t","t in sec","x(2t)");
subplot(3,1,2);
plot2d(2*t,x,rect=[-10 0 10 2]);
xtitle("x(t/2) vs t","t in sec","x(t/2)");
subplot(3,1,3);
plot2d(-t/3,x,rect=[-10 0 10 2]);
xtitle("x(-3t) vs t","t in sec","x(-3t)");
xset('window',2);
subplot(3,1,1);
plot2d(-(t-3),x,rect=[-10 0 10 2]);
xtitle("x(3-t) vs t","t in sec","x(3-t)");
subplot(3,1,2);
```

```
plot2d(-(t-2)/2,x,rect=[-10 0 10 2]);
xtitle("x(-2t+2) vs t","t in sec","x(-2t+2)");
```

Scilab code Exa 2.3 Signal operations

```
//Example 2.3
clear
clc
clf
n=-20:1:20;
x=[zeros(1,19),1,1,2,3,4,0.5, zeros(1,16)];
subplot(3,1,1);
plot(n,x,".");
xtitle("x(n) vs n","n","x(n)");
subplot(3,1,2);
plot(n+3,x,'.');
xtitle("x(n-3) vs n","n","x(n-3)");
subplot(3,1,3);
plot(n-2,x,'.');
xtitle("x(n+2) vs n","n","x(n+2)");
figure(1)
subplot(3,1,1);
plot(-n,x,'.');
xtitle("x(-n) vs n","n","x(-n)");
subplot(3,1,2);
plot(-n+2,x,'.');
xtitle("x(-n+2) vs n","n","x(-n+2)");
subplot(3,1,3)
plot(-n-3,x,'.')
xtitle("x(-n-3) vs n","n","x(-n-3)");
```

Chapter 3

continuoustime and discretetime systems

Scilab code Exa 3.10 properties of a system

```
//Example 3.10
clc
clear all
t=0:0.01:20;
function y=signal(x)
    y=x
endfunction
//Assume v(t) as ramp signal
v1=t;
v2=t+2;
//Assume R=1
i1=signal(v1)
i2=signal(v2)
a=2;
b=2;
subplot(4,2,1)
plot(t,a*v1)
xtitle("a*v1(t)","t in sec","a*v1(t)");
subplot(4,2,2)
```

```
plot(t,signal(a*v1))
xtitle("a*i1(t)","t in sec","a*i1(t)");
subplot(4,2,3)
plot(t,b*v2)
xtitle("b*v2(t)","t in sec","b*v2(t)");
subplot(4,2,4)
plot(t,signal(b*v2))
xtitle("b*i2(t)","t in sec","b*i2(t)");
c=(a*v1)+(b*v2);
subplot(4,2,5)
plot(t,c)
xtitle("v3(t)","t in sec","v3(t)");
subplot(4,2,6)
plot(t,signal(c))
xtitle("i3(t)","t in sec","i3(t)");
subplot(4,2,8)
plot(t,signal(a*v1)+signal(b*v2))
xtitle("LINEAR SYSTEM","t in sec","a*i1(t)+b*i2(t)")
    ;
```

Scilab code Exa 3.11 properties of a system

```
//Example 3.11
clc
clear all
t=0:0.001:0.5;
function i=signal(v)
    i=exp(v);
endfunction
//Assume v(t) as ramp signal
x1=2*ones(1,length(t));
x2=t+2;
//Assume R=1
y1=signal(x1)
y2=signal(x2)
```

```
a=2;
b=2;
subplot(4,2,1)
plot(t,a*x1)
xtitle("a*x1(t)","t in sec","a*x1(t)");
subplot(4,2,2)
plot(t,signal(a*x1))
xtitle("a*y1(t)","t in sec","a*y1(t)");
subplot(4,2,3)
plot(t,b*x2)
xtitle("b*x2(t)","t in sec","b*x2(t)");
subplot(4,2,4)
plot(t,signal(b*x2))
xtitle("b*y2(t)","t in sec","b*y2(t)");
c=(a*x1)+(b*x2);
subplot(4,2,5)
plot(t,c)
xtitle("x3(t)","t in sec","x3(t)");
subplot(4,2,6)
plot(t,signal(c))
xtitle("y3(t)","t in sec","y3(t)");
subplot(4,2,8)
plot(t,signal(a*x1)+signal(b*x2))
xtitle("NON-LINEAR SYSTEM","t in sec","a*y1(t)+b*y2(
    t)");
```

Scilab code Exa 3.13 properties of a system

```
//Example 3.13
clear;
clc;
x1 = [1,1,1,1];
x2 = [2,2,2,2];
a = 1;
b = 1;
```

```
for n = 1:length(x1)
  x3(n) = a*x1(n)+b*x2(n);
end
for n = 1:length(x1)
  y1(n) = x1(n)^2;
  y2(n) = x2(n)^2;
  y3(n) = x3(n)^2;
end
for n = 1:length(y1)
  z(n) = a*y1(n)+b*y2(n);
end
count = 0;
for n =1:length(y1)
  if(y3(n)== z(n))
    count = count+1;
  end
end
if(count == length(y3))
   disp('Since It satisifies the superposition
      principle')
   disp('The given system is a Linear system')
   y3
   z
  else
    disp('Since It does not satisify the
       superposition principle')
    disp('The given system is a Non-Linear system')
end
```

Scilab code Exa 3.14 properties of a system

```
//Example 3.14
clear;
clc;
to = 2; //Assume the amount of time shift  =2
```

```
T=10;
t=0:0.1:T;
for i=1:length(t)
    if (t(i)>=0 & t(i)<1)
       x1(i) = t(i);
       x2(i)=0;
    elseif (t(i)>=1 & t(i)<2) then
        x1(i)=1;
        x2(i)=t(i)-1;
    elseif (t(i)>=2 & t(i)<3) then
        x1(i)=2;
        x2(i)=1;
    elseif (t(i)>=3 & t(i)<4)
        x1(i)=0;
        x2(i)=2;
    else
        x1(i)=0;
        x2(i)=0;
    end
y1(i) = 2*(x1(i));
y2(i)=2*x2(i);
end
figure(0);
subplot(2,1,1);
plot(t,x1);
xtitle("x1(t)","t","x1(t)");
subplot(2,1,2);
plot(t,y1);
xtitle("y1(t)=2*x1(t)","t","y1(t)");
figure(1);
subplot(2,1,1);
plot(t,x2);
xtitle("x2(t)","t","x2(t)");
subplot(2,1,2);
plot(t,y2);
xtitle("y2(t)=2*x2(t)=2*x1(t-1)=y1(t-1)","t","y2(t)");
//First shift the input signal only
```

```
Input_shift = 2*(x1(T-to));
Output_shift = y1(T-to);
if(Input_shift == Output_shift)
  disp('The given system is a Time In-variant system
     ');
else
  disp('The given system is a Time Variant system');
end
```

Scilab code Exa 3.15 properties of a system

```
//Example 3.15
clear;
clc;
to = 2; //Assume the amount of time shift =2
T=10;
for t = 1:0.01:T
  x(t) = sin(t);
  y(t) = sin(2*t);
end
//First shift the input signal only
Input_shift = x(T-to);
Output_shift = y(T-to);
if(Input_shift == Output_shift)
  disp('The given system is a Time In-variant system
     ');
else
  disp('The given system is a Time Variant system');
end
```

Scilab code Exa 3.16 properties of a system

```
//Example 3.16
```

```
clear;
clc;
no = 2; //Assume the amount of time shift  =2
L = 10; //Length of given  signal
for n = 1:L
    x(n)=sin(n);
  end
  n=2;
  for i=1:L
      y(i)=x(n-1);
      n=n+1;
  end
//First shift the input signal only
Input_shift = x(L-no);
Output_shift = y(L-no);
if(Input_shift == Output_shift)
  disp('The given discrete system is a Time In−
     variant system');
else
  disp('The given discrete system is a Time Variant
     system');
end
```

Scilab code Exa 3.17 properties of a system

```
//Example 3.17
clear;
clc;
no = 2; //Assume the amount of time shift  =2
L = 10; //Length of given  signal
for n = 1:L
    x(n)=sin(n);
    y(n)=-sin(n);
  end

```

```
//First shift the input signal only
Input_shift = x(L-no);
Output_shift = y(L-no);
if(Input_shift == Output_shift)
  disp('The given discrete system is a Time In−
     variant system');
else
  disp('The given discrete system is a Time Variant
     system');
end
```

Scilab code Exa 3.18 properties of a system

```
//Example 3.18
clc
clear
t=0:0.01:10;
for i=1:length(t)
    x(i)=sin(i);
    y(i)=2*x(i);
    z(i)=0.5*y(i);
end
if (x==z) then
    disp("The given system is invertible");
else
    disp("the Given system is non−invertible");
end
```

Chapter 4

Linear Time Invariant Systems

Scilab code Exa 4.1 Convolution

```
//Example 4.1
//Convolution sum of x[n] and h[n]
clc
clear
n=[0 1 2];
n1=0:4;
x=[0.5 0.5 0.5];
h=[3 2 1];
y=convol(h,x)
disp("Convolution of x[n] and h[n] is...")
disp(y)
subplot(3,1,1)
xtitle("","....................n","x[n]");
plot2d3('gnn',n,x,5);
subplot(3,1,2)
xtitle("","....................n","h[n]");
plot2d3('gnn',n,h,5);
subplot(3,1,3)
xtitle("",".............................n","y[n]");
plot2d3('gnn',n1,y,5);
```

Scilab code Exa 4.2 Convolution

```
//Example 4.2
//Convolution sum of x[n] and h[n]
clc
clear
n=-1:1;
n1=-2:2;
x=[0.5 0.5 0.5];
h=[3 2 1];
y=coeff(poly(h,'z','c')*poly(x,'z','c'))
disp("Convolution of x[n] and h[n] is...")
disp(y)
```

Scilab code Exa 4.5 Convolution

```
//Example 4.5
//Convolution sum of x[n] and h[n]
clc
clear
n=0:2;
n1=0:4;
x=[0.5 0.5 0.5];
h=[3 2 1];
y=coeff(poly(h,'z','c')*poly(x,'z','c'))
disp("Convolution of x[n] and h[n] is...")
disp(y)
subplot(3,1,1)
xtitle("input signal x(n)","...................n","
   x[n]");
plot(n,x,'.');
subplot(3,1,2)
```

```
xtitle("system response h(n)",".....................n
    ","h[n]");
plot(n,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","
    .............................n","y[n]");
plot(n1,y,'.');
```

Scilab code Exa 4.6 convolution

```
//Example 4.6
//Convolution sum of x[n] and h[n]
clc
clear
n=-1:1;
n1=-2:2;
x=[0.5 0.5 0.5];
h=[3 2 1];
y=coeff(poly(h,'z','c')*poly(x,'z','c'))
disp("Convolution of x[n] and h[n] is...")
disp(y)
subplot(3,1,1)
xtitle("input signal x(n)","....................n","
    x[n]");
plot(n,x,'.');
subplot(3,1,2)
xtitle("system response h(n)",".....................n
    ","h[n]");
plot(n,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","
    .............................n","y[n]");
plot(n1,y,'.');
```

Scilab code Exa 4.10 Convolution

```
//Example 4.10
//Convolution sum of x[n] and h[n]
clc
clear
n=0:2;
n1=0:4;
x=[0.5 0.5 0.5];
h=[3 2 1];
A=[x 0 0;0 x 0; 0 0 x];
y=A'*h'
disp("Convolution of x[n] and h[n] is...")
disp(y)
subplot(3,1,1)
xtitle("input signal x(n)","....................n","
    x[n]");
plot(n,x,'.');
subplot(3,1,2)
xtitle("system response h(n)",".....................n
    ","h[n]");
plot(n,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","
    .............................n","y[n]");
plot(n1,y,'.');
```

Scilab code Exa 4.11 Convolution

```
//Example 4.11
//Convolution sum of x[n] and h[n]
clc
```

```
clear
n=-1:1;
n1=-2:2;
x=[0.5 0.5 0.5];
h=[3 2 1];
A=[x 0 0;0 x 0; 0 0 x];
y=A'*h'
disp("Convolution of x[n] and h[n] is...")
disp(y)
subplot(3,1,1)
xtitle("input signal x(n)","....................n","x[n]");
plot2d3('gnn',n,x,5);
subplot(3,1,2)
xtitle("system response h(n)","....................n","h[n]");
plot2d3('gnn',n,h,5);
subplot(3,1,3)
xtitle("output signal y(n)","............................n","y[n]");
plot2d3('gnn',n1,y,5);
```

Scilab code Exa 4.13 Convolution

```
//Example 4.13
//Convolution sum of x[n] and h[n]
clc
clear
n=0:100;
n1=0:200;
for i=1:length(n)
    x(i)=n(i);
    h(i)=1;
end
y=convol(x,h);
```

```
disp(y)
subplot(3,1,1)
xtitle("input signal x(n)","...................n","x[n]");
plot(n,x,'.');
subplot(3,1,2)
xtitle("system response h(n)","....................n","h[n]");
plot(n,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","...........................n","y[n]");
plot(n1,y,'.');
```

Scilab code Exa 4.14 Convolution

```
//Example 4.14
//Convolution sum of x[n] and h[n]
clc
clear
n=0:100;
n1=0:200;
a=0.7//assume the constant a=0.7
for i=1:length(n)
    x(i)=a^n(i);
    h(i)=1;
end
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(n)","...................n","x[n]");
plot(n,x,'.');
subplot(3,1,2)
xtitle("system response h(n)","....................n","h[n]");
```

```
plot(n,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","
    ..............................n","y[n]");
plot(n1,y,'.');
```

Scilab code Exa 4.15 Convolution

```
//Example 4.15
//Convolution sum of x[n] and h[n]
clc
clear
n1=-100:1:0;
n2=0:100;
n3=-100:100;
a=0.5//assume the constant a=0.5
for i=1:length(n1)
    x(i)=a^-n1(i);
    h(i)=a^n1(i);
end
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(n)","....................n","
    x[n]");
plot(n1,x,'.');
subplot(3,1,2)
xtitle("system response h(n)",".....................n
    ","h[n]");
plot(n2,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","
    ..............................n","y[n]");
plot(n3,y,'.');
```

Scilab code Exa 4.16 convolution

```
//Example 4.16
//Convolution sum of x[n] and h[n]
clc
clear
n1=-100:-2;
n2=2:100;
n3=-98:98;
a=1/2//assume the constant a=1/2
for i=1:length(n1)
    x(i)=a^-n1(i);
    h(i)=1;
end
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(n)","....................n","
  x[n]");
plot(n1,x,'.');
subplot(3,1,2)
xtitle("system response h(n)","....................n
  ","h[n]");
plot(n2,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","
  ..............................n","y[n]");
plot(n3,y,'.');
```

Scilab code Exa 4.17 Convolution

```
//Example 4.17
//Convolution sum of x[n] and h[n]
```

```
clc
clear
n1=2:12;
n2=4:14;
n3=6:26;
a=1/3//assume the constant a=1/3
for i=1:length(n1)
    x(i)=a^-n1(i);
    h(i)=1;
end
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(n)","....................n","x[n]");
plot(n1,x,'.');
subplot(3,1,2)
xtitle("system response h(n)","....................n","h[n]");
plot(n2,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)",".............................n","y[n]");
plot(n3,y,'.');
```

Scilab code Exa 4.18 Convolution

```
//Example 4.18
//Convolution sum of x[n] and h[n]
clc
clear
n1=-100:0;
n2=0:100;
n3=-100:100;
b=0.8//assume the constant b=0.4
a=0.8//assume the constant a=0.8
```

```
for i=1:length(n1)
    x(i)=a^n1(i);
    h(i)=b^n1(i);
end
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(n)","....................n","
  x[n]");
plot(n1,x,'.');
subplot(3,1,2)
xtitle("system response h(n)","....................n
  ","h[n]");
plot(n2,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","
   ............................n","y[n]");
plot(n3,y,'.');
```

Scilab code Exa 4.19 Convolution

```
//Example 4.19
//Convolution sum of x[n] and h[n]
clc
clear
n1=0:9;
n2=0:100;
n3=0:109;
b=0.8//assume the constant b=0.4
a=0.8//assume the constant a=0.8
for i=1:length(n1)
    x(i)=a^n1(i);
end
for j=1:length(n2)
 h(j)=b^n2(j);
end
```

```
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(n)","...................n","
   x[n]");
plot(n1,x,'.');
subplot(3,1,2)
xtitle("system response h(n)",".....................n
   ","h[n]");
plot(n2,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","
   ..............................n","y[n]");
plot(n3,y,'.');
```

Scilab code Exa 4.20 Convolution

```
//Example 4.20
//Convolution sum of x[n] and h[n]
clc
clear
n1=0:5;
n2=0:7;
n3=0:12;
a=0.8//assume the constant a=0.8
for i=1:length(n1)
    x(i)=1;
end
for j=1:length(n2)
 h(j)=a^n2(j);
end
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(n)","...................n","
   x[n]");
plot(n1,x,'.');
```

```
subplot(3,1,2)
xtitle("system response h(n)","...................n
   ","h[n]");
plot(n2,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)","
    ..........................n","y[n]");
plot(n3,y,'.');
```

Scilab code Exa 4.21 Convolution

```
//Example 4.21
//Convolution of x(t) and h(t)
clc
clear
t1=0:0.01:5;
t2=0:0.01:2;
t3=0:0.01:7;
for i=1:length(t1)
    x(i)=1;
end
for j=1:length(t2)
 h(j)=1;
end
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(t)","..................t","
   x[t]");
plot(t1,x);
subplot(3,1,2)
xtitle("system response h(t)","..................t
   ","h[t]");
plot(t2,h);
subplot(3,1,3)
xtitle("output signal y(t)","
```

```
.............................t","y[t]");
plot(t3,y);
```

Scilab code Exa 4.23 Convolution

```
//Example 4.23
//Convolution of x(t) and h(t)
clc
clear
t1=0:0.01:20;
t2=0:0.01:20;
t3=0:0.01:40;
a1=0.5;//constants a and b are equal
b1=0.5;
a2=0.8;// constants a and b are unequal
b2=0.3;
for i=1:length(t1)
    x1(i)=exp(-a1*t1(i));
    x2(i)=exp(-a2*t1(i));
end
for j=1:length(t2)
     h1(j)=exp(-b1*t2(j));
    h2(j)=exp(-b2*t2(j));
end
//case 1: a & b are equal
y1=convol(x1,h1);
subplot(3,1,1)
xtitle("input signal x(t)","....................t","x[t]");
plot(t1,x1);
subplot(3,1,2)
xtitle("system response h(t)","....................t","h[t]");
plot(t2,h1);
subplot(3,1,3)
```

```
xtitle("output signal y(t)","............................t","y[t]");
plot(t3,y1);
//case 2: a& b are unequal
figure(1)
y2=convol(x2,h2);
subplot(3,1,1)
xtitle("input signal x(t)","....................t","x[t]");
plot(t1,x2);
subplot(3,1,2)
xtitle("system response h(t)",".....................t","h[t]");
plot(t2,h2);
subplot(3,1,3)
xtitle("output signal y(t)","............................t","y[t]");
plot(t3,y2);
```

Scilab code Exa 4.24 Convolution

```
//Example 4.24
//Convolution of x(t) and h(t)
clc
clear
t1=-3:0.01:10;
t2=1:0.01:10;
t3=-2:0.01:20;
a=0.5//assume a=0.5
for i=1:length(t1)
    x(i)=exp(-a*t1(i));
end
for j=1:length(t2)
 h(j)=exp(-a*t2(j));
end
```

```
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(t)","....................t","x[t]");
plot(t1,x);
subplot(3,1,2)
xtitle("system response h(t)",".....................t","h[t]");
plot(t2,h);
subplot(3,1,3)
xtitle("output signal y(t)","..............................t","y[t]");
plot(t3,y);
```

Scilab code Exa 4.26 Convolution

```
//Interconnectiuion of LTI systems
n=-10:10;
for i=1:length(n)
    if(n(i)==0)
        h1(i)=2;
    else
        h1(i)=1;
    end
end
for i=1:length(n)
    if(n(i)==2)
        h2(i)=1;
    else
        h2(i)=0;
    end
end
for i=1:length(n)
    if(n(i)>=1)
        h3(i)=1;
```

```
    else
        h3(i)=0;
end
end
for i=1:length(n)
    if(n(i)>= -1)
        h4(i)=1;
    else
        h4(i)=0;
    end
end
for i=1:length(n)
    h5(i)=n(i);
    h6(i)=1;
end
h23=h2.*h3;
h234=h4+h23;
h1234=h1.*h234;
h56=h5.*h6;
h=h56+h1234;
x=[1 -0.5];
n1=[0 1];
y=convol(x,h);
n2=-10:11;
subplot(3,1,1)
xtitle("input signal x(n)","....................n","x[n]");
plot(n1,x,'.');
subplot(3,1,2)
xtitle("system response h(n)","....................n","h[n]");
plot(n,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)",".............................n","y[n]");
plot(n2,y,'.');
```

Scilab code Exa 4.27 Convolution

```
//Example 4.27
//Interconnectiuion of LTI systems
n2=0:18;
h1=[1 5 10 11 8 4 1];
h2=[1 1 zeros(1,5)];
h3=[1 1 zeros(1,5)];
a=convol(h1,h2);
h=convol(a,h3);
x=[1 -1];
n1=[0 1];
n3=0:19;
y=convol(x,h);
subplot(3,1,1)
xtitle("input signal x(n)","...................n","x[n]");
plot(n1,x,'.');
subplot(3,1,2)
xtitle("system response h(n)","....................n","h[n]");
plot(n2,h,'.');
subplot(3,1,3)
xtitle("output signal y(n)",".............................n","y[n]");
plot(n3,y,'.');
```

Scilab code Exa 4.30 Convolution

```
//Example 4.30
//Cascade connection of systems
clc
```

```
clear
n=0:10;
h11=[1 -0.5];
for i=1:length(n)
    h2(i)=0.5^n(i);
    if (n(i)==0) then
        h1(i)=1;
    elseif n(i)==1 then
            h1(i)=-0.5
    else
        h1(i)=0;
    end
end
h=convol(h11,h2);
n2=0:11;
//assume x[n]=[1 1 1]
n1=0:2;
x=[1 1 1];
n3=0:13;
y=convol(x,h);
subplot(3,1,1);
plot(n1,x,'.');
xtitle("Input Signal x[n]","n","x[n]")
subplot(3,1,2);
plot(n2,h,'.');
xtitle("Impulse Response h[n]","n","h[n]")
subplot(3,1,3);
plot(n3,y,'.');
xtitle("Output Signal y[n]","n","y[n]")
disp("the given system is an invertible system");
```

Chapter 5

Fourier Analysis of Continuoustime signals and systems

Scilab code Exa 5.1 Fourier Series representation

```
//Continuous Time Fourier Series Coefficients of
//a periodic signal x(t) = sin(2*Wot)
clear;
close;
clc;
t = 0:0.01:1;
T = 1;
Wo = 2*%pi/T;
xt = sin(2*Wo*t);
for k =0:4
  C(k+1,:) = exp(-sqrt(-1)*Wo*t.*k);
  a(k+1) = xt*C(k+1,:)'/length(t);
```

```
  if(abs(a(k+1))<=0.01)
    a(k+1)=0;
  end
end
a =a';
ak = [-a,a(2:$)]
for i=1:length(ak)
    if real(ak(i))== 0 then
        phase(i)=0;
    else
         if i<length(ak)/2 then
            phase(i)= atan(imag(ak(i))/real(ak(i)));
         else
            phase(i)= -atan(imag(ak(i))/real(ak(i)));
         end
    end
end
disp("The fourier series coefficients are...")
disp(ak)
disp("magnitude of Fourier series coefficient")
disp(abs(ak))
disp("Phase of Fourier series coefficient in radians")
disp(phase)
n=-4:4;
subplot(2,1,1)
plot(n,abs(ak),'.');
xtitle("|ak|","k","|ak|");
subplot(2,1,2)
for i=1:length(n)
    if n(i)== -2 then
    phase(i)=3.14/2;
    elseif n(i)== 2 then
    phase(i)= -3.14/2;
    else
    phase(i)=0;
end
```

```
end
plot(n,phase,'.');
xtitle("/_ak","k","/_ak");
```

Scilab code Exa 5.2 Fourier Series representation

```
//Continuous Time Fourier Series Coefficients of
//a periodic signal x(t) = cos(Wot)
clear;
close;
clc;
t = 0:0.01:1;
T = 1;
Wo = 2*%pi/T;
xt = cos(Wo*t);
x1t=cos(Wo*-t);
for k =0:2
  C(k+1,:) = exp(-sqrt(-1)*Wo*t.*k);
  a(k+1) = xt*C(k+1,:)'/length(t);
  if(abs(a(k+1))<=0.01)
    a(k+1)=0;
  end
end
a =a';
ak = [-a,a(2:$)]
disp("The fourier series coefficients are...")
disp(ak)
disp("magnitude of Fourier series coefficient")
disp(abs(ak))
n=-2:2;
subplot(2,1,1)
plot(n,abs(ak),'.');
xtitle("Magnitude Spectrum","k","|ak|");
if xt== x1t then
    disp("The Given signal is even. It has no phase
```

```
        spectrum");
else
for i=1:length(ak)
    if real(ak(i))== 0 then
        phase(i)=0;
    else
        phase(i)=atan(imag(ak(i))/real(ak(i)));
    end
end
disp("Phase of Fourier series coefficient in radians
   ")
disp(phase)
subplot(2,1,2)
plot(n,phase,'.');
xtitle("Phase Spectrum","k","ak in radians");
end
```

Scilab code Exa 5.3 Fourier series representation

```
//Continuous Time Fourier Series Coefficients of
//a periodic signal x(t) = 5*cos((%pi/2*t)+(%pi/6))
clear;
close;
clc;
t = 0:0.01:1;
T = 1;
Wo = 2*%pi/T;
xt = cos((%pi/2*t)+(%pi/6))
x1t=cos((%pi/2*-t)+(%pi/6))

//x(t) is expanded according to Euler's theorem
x=5/2*(exp(%i*(%pi/2*t+%pi/6))+exp(-%i*(%pi/2*t+%pi
   /6)));
a1=5/2*exp(%i*%pi/6);
a_1=5/2*exp(-%i*%pi/6);
```

```
ak=[zeros(1,5) a_1 0 a1 zeros(1,5)];
k=-6:6;
disp("The fourier series coefficients are...")
disp(ak)
disp("magnitude of Fourier series coefficient")
disp(abs(ak))
subplot(2,1,1)
plot(k,abs(ak),'.');
xtitle("Magnitude Spectrum","k","|ak|");
if xt== x1t then
    disp("The Given signal is even. It has no phase
       spectrum");
else
    phase=[zeros(1,5) atan(imag(a_1)/real(a_1)) 0
       atan(imag(a1)/real(a1))  zeros(1,5)];
    disp("Phase of Fourier series coefficient in
       radians")
    disp(phase)
    subplot(2,1,2)
    plot(k,phase,'.');
    xtitle("Phase Spectrum","k","ak in radians");
end
```

Scilab code Exa 5.4 Fourier series representation

```
//Continuous Time Fourier Series Coefficients of
//a periodic signal x(t) = 1+sin(6t)+cos(4t)
clear;
close;
clc;
t = 0:0.01:1;
xt = 1+sin(6*t)+cos(4*t);
x_t = 1+sin(6*-t)+cos(4*-t);

//x(t) is expanded according to Euler's theorem
```

```
x=1+(1/2)*exp(%i*4*t)+(1/2)*exp(-%i*4*t)+(1/(2*%i))*exp(%i*6*t)-(1/(2*%i))*exp(-%i*6*t);
a0=1;
a2=(1/2)
a_2=(1/2)
a3=(1/(2*%i));
a_3=-(1/(2*%i));
ak=[zeros(1,5) a_3 a_2 0 a2 a3 zeros(1,5)];
k=-7:7;
disp("The fourier series coefficients are...")
disp(ak)
disp("magnitude of Fourier series coefficient")
disp(abs(ak))
subplot(2,1,1)
plot(k,abs(ak),'.');
xtitle("Magnitude Spectrum","k","|ak|");
if xt== x_t then
    disp("The Given signal is even. It has no phase spectrum");
else
    phase=[zeros(1,6) %pi/2 0 -%pi/2  zeros(1,6)];
    disp("Phase of Fourier series coefficient in radians")
    disp(phase)
    subplot(2,1,2)
    plot(k,phase,'.');
    xtitle("Phase Spectrum","k","ak in radians");
end
```

Scilab code Exa 5.5 Fourier Series Coefficients

```
//Fourier Series coefficients of the signal x(t)
//Assume the period of the signal T=10
clc
clear
```

```
close
T=1;
To=1/4;
//Assume the magnitude of the signal A=1
A=1;
t=-10:0.01:10;
for i=1:length(t)
    if t>To & t<-To then
        x(i)=0;
    else
        x(i)=A;
    end
end

Wo=2*%pi;

k=-5:5
for i=1:length(k)
    if k(i)==0 then
        ak(i)=1.5;
    else
        ak(i)=(sin(k(i)*%pi/2))/(k(i)*%pi);
    end
end

disp("The fourier series coefficients are...")
disp(ak)
disp("magnitude of Fourier series coefficient")
disp(abs(ak))
disp("the givem signal is even and so it has no phase spectrum")
//PLotting frequency spectrum
subplot(2,1,2)
plot(k,abs(ak),'.');
xtitle("Magnitude Spectrum","k","|ak|");
subplot(2,1,1)
plot(k,ak,'.');
xtitle("Ak","k","ak");
```

Scilab code Exa 5.6 Fourier series Coefficients

```
//Fourier Series coefficients for Impulse train
clc
clear
close
//Assume period of the impulse train T=2
T=2;
t=-5*T:T:5*T;
for i=1:length(t)
    x(i)=1;
end
//Using sifting property of the impulse signal
k=-10:10
for i=1:length(k)
    ak(i)=1/T;
end
subplot(2,1,1)
plot(t,x,'.')
xtitle("Impulse train","t","x(t)")
subplot(2,1,2)
plot(k,ak,'.')
xtitle("Fourier coefficients of impulse train","k","
   ak")
```

Scilab code Exa 5.7 Fourier series coefficients of halfwave rectified signal

```
//Fourier Series coefficients of half-wave rectifier
    output
//Assume the period of the signal T=1
t=-0.5:0.01:1;
```

```
for i=1:length(t)
    if t(i)<T/2 then
        x(i)=sin(2*%pi*t(i));
    else
        x(i)=0;
    end
end
k=-10:10;
for i=1:length(k)
    if k(i)==1 then
        ak(i)=1/(4*%i);
    elseif k(i)==-1
        ak(i)=-1/(4*%i);
    else
        ak(i)=(cos(k(i)*%pi/2)*exp(-k(i)*%pi/2*-%i))/(%pi-(%pi*k(i)*k(i)));
    end
end


disp("The fourier series coefficients are...")
disp(ak)
disp("magnitude of Fourier series coefficient")
disp(abs(ak))
//PLotting frequency spectrum
subplot(2,1,1)
plot(k,abs(ak),'.');
xtitle("Magnitude Spectrum","k","|ak|");
for i=1:length(k)
    if k(i)==0 | k(i)==3 | k(i)==-3 | k(i)==-5 |k(i)==5 then
        phase(i)=0;
    elseif k(i)==-1 then
            phase(i)=%pi/2;
    elseif k(i)==1  then
             phase(i)=-%pi/2;
    elseif k(i)==-2 | k(i)==-4
          phase(i)=%pi;
```

```
    elseif k(i)==2 | k(i)==4
            phase(i)=-%pi;

end
end
subplot(2,1,2)
plot(k,phase,'.');
xtitle("Phase Spectrum","k","angle(ak)");
disp(phase)
```

Scilab code Exa 5.8 Fourier series coefficients

```
//Fourier Series coefficients of half-wave rectifier
   output
//Assume the period of the signal T=1
t=-0.5:0.01:0.5;
for i=1:length(t)
    if t(i)<-0.25 & t(i)>0.25 then
        x(i)=-1;
    else
        x(i)=1;
    end
end
k=-10:10;
for i=1:length(k)
    if k(i)==0 then
        ak(i)=0;
    else
        ak(i)=(%i*((2-(-1)^k(i))*exp(-%i*k(i)*%pi/2)
           -exp(%i*k(i)*%pi/2)))/(k(i)*2*%pi);
    end
end

disp("The fourier series coefficients are...")
disp(ak)
```

```
plot(k,ak,'.')
xtitle("Fourier Coefficients","k","ak")
```

Scilab code Exa 5.14 Continuoustime Fourier Transform

```
//Continuous Time Signal x(t)= exp(−B∗t)u(t), t>0
clear;
clc;
close;
B =1;
Dt = 0.005;
t = 0:Dt:10;
xt = exp(-B*t);
Wmax = 2*%pi*1;
K = 4;
k = 0:(K/1000):K;
W = k*Wmax/K;
XW = xt* exp(-sqrt(-1)*t'*W) * Dt;
XW_Mag = abs(XW);
W = [-mtlb_fliplr(W), W(2:1001)];
XW_Mag = [mtlb_fliplr(XW_Mag), XW_Mag(2:1001)];
[XW_Phase,db] = phasemag(XW);
XW_Phase = [-mtlb_fliplr(XW_Phase),XW_Phase(2:1001)
   ];
//Plotting Continuous Time Signal
figure(1)
plot(t,xt);
xlabel('t in sec.');
ylabel('x(t)')
title('Continuous Time Signal')
figure(2)
//Plotting Magnitude Response of CTS
subplot(2,1,1);
plot(W,XW_Mag);
xlabel('Frequency in Radians/Seconds−−−> W');
```

```
ylabel('abs(X(jW))')
title('Magnitude Response (CTFT)')
//Plotting Phase Reponse of CTS
subplot(2,1,2);
plot(W,XW_Phase*%pi/180);
xlabel('                                Frequency in Radians/Seconds---> W');
ylabel('                                                            <X(jW)')
title('Phase Response(CTFT) in Radians')
```

Scilab code Exa 5.15 Continuoustime Fourier Transform

```
//Continuous Time Signal x(t)= exp(B*t)u(-t), t>0
clear;
clc;
close;
B =1;
Dt = 0.005;
t = -10:Dt:0;
xt = exp(B*t);
Wmax = 2*%pi*1;
K = 4;
k = 0:(K/1000):K;
W = k*Wmax/K;
XW = xt* exp(-sqrt(-1)*t'*W) * Dt;
XW_Mag = abs(XW);
W = [-mtlb_fliplr(W), W(2:1001)];
XW_Mag = [mtlb_fliplr(XW_Mag), XW_Mag(2:1001)];
[XW_Phase,db] = phasemag(XW);
XW_Phase = [-mtlb_fliplr(XW_Phase),XW_Phase(2:1001)
    ];
//Plotting Continuous Time Signal
figure(1)
```

```
plot(t,xt);
xlabel('t in sec.');
ylabel('x(t)')
title('Continuous Time Signal')
figure(2)
//Plotting Magnitude Response of CTS
subplot(2,1,1);
plot(W,XW_Mag);
xlabel('Frequency in Radians/Seconds---> W');
ylabel('abs(X(jW))')
title('Magnitude Response (CTFT)')
//Plotting Phase Reponse of CTS
subplot(2,1,2);
plot(W,XW_Phase*%pi/180);
xlabel('                              Frequency in
   Radians/Seconds---> W');
ylabel('
                                                          <X
   (jW)')
title('Phase Response(CTFT) in Radians')
```

Scilab code Exa 5.16 Continuoustime Fourier Transform

```
//Continuous Time Signal x(t)= exp(-B*abs(t))
clear;
clc;
close;
B =1;
Dt = 0.005;
t = -4.5:Dt:4.5;
xt = exp(-B*abs(t));
Wmax = 2*%pi*1;
K = 4;
k = 0:(K/1000):K;
W = k*Wmax/K;
```

```
XW = xt* exp(-sqrt(-1)*t'*W) * Dt;
XW = real(XW);
W = [-mtlb_fliplr(W), W(2:1001)];
XW = [mtlb_fliplr(XW), XW(2:1001)];
disp("The given signal is even and it has no phase
    spectrum")
subplot(2,1,1);
plot(t,xt);
xlabel('t in sec.');
ylabel('x(t)')
title('Continuous Time Signal')
subplot(2,1,2);
plot(W,XW);
xlabel('Frequency in Radians/Seconds W');
ylabel('X(jW)')
title('Continuous-time Fourier Transform')
```

Scilab code Exa 5.17 Continuoustime Fourier Transform

```
//Frequency Response of a Rectangular Waveform
// x(t)= A, from -T1 to T1
clear;
clc;
close;
A =1;
Dt = 0.005;
T1 = 4;
t = -T1/2:Dt:T1/2;
for i = 1:length(t)
  xt(i) = A;
end
Wmax = 2*%pi*1;
K = 4;
k = 0:(K/1000):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);
plot(t,xt);
xlabel('t in sec.');
title('Contiuous Time Signal x(t)')
subplot(2,1,2);
plot(W,XW_Mag);
xlabel('Frequency in Radians/Seconds');
title('Continuous-time Fourier Transform X(jW)')
```

Scilab code Exa 5.18 Inverse Fourier transform

```
// Inverse Continuous Time Fourier Transform
// X(jW)= 1, from -T1 to T1
clear;
clc;
close;
// CTFT
A =1;
Dw = 0.005;
W1 = 4;
w = -W1/2:Dw:W1/2;
for i = 1:length(w)
  XW(i) = A;
end
XW = XW';
//Inverse Continuous-time Fourier Transform
t = -3*%pi:%pi/length(w):3*%pi;
xt =(1/(2*%pi))*XW *exp(sqrt(-1)*w'*t)*Dw;
xt = real(xt);
figure
```

```
plot(t,xt);
xlabel('            t Sec');
title('Time domain signal x(t)')
```

Scilab code Exa 5.19 Continuoustime Fourier Transform

```
//frequency response of impulse signal
clear;
clc;
close;
A =1;
Dt = 0.005;
T1 = 4;
Wo=2//Assume Wo=2
t = -T1/2:Dt:T1/2;
for i = 1:length(t)
    xt(i)=sin(Wo*t(i));
end
Wmax = 2*%pi*1;
K = 4;
k = 0:(K/1000):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);
plot(t,xt);
xlabel('t in sec.');
title('Contiuous Time Signal x(t)')
subplot(2,1,2);
plot(W,XW_Mag);
xlabel('Frequency in Radians/Seconds');
title('Continuous-time Fourier Transform X(jW)')
```

Scilab code Exa 5.20 Inverse Fourier transform

```
// Inverse Continuous Time Fourier Transform
// X(jW)= 2*pi , at W=0
clear;
clc;
close;
// CTFT
A =1;
Dw = 0.005;
W1 = 4;
w = -W1/2:Dw:W1/2;
for i = 1:length(w)
    if w(i)==0 then
  XW(i) = 2*%pi;
else
    XW(i)=0;
end
end
XW = XW';
subplot(2,1,1)
plot(w,XW)
//Inverse Continuous-time Fourier Transform
t = -3*%pi:%pi/length(w):3*%pi;
xt =(1/(2*%pi))*XW *exp(sqrt(-1)*w'*t)*Dw;
xt = real(1+xt);
subplot(2,1,2)
plot(t,xt);
xlabel('             t Sec');
title('Time domain signal x(t)')
```

Scilab code Exa 5.21 Inverse Fourier Transform

```
// Inverse Continuous Time Fourier Transform
// X(jW)= 2*pi , at W=Wo
clear;
clc;
close;
// CTFT
A =1;
Dw = 0.005;
W1 = 4;
Wo=2//Assume Wo=2
w = -W1/2:Dw:W1/2;
for i = 1:length(w)
    if w(i)==Wo then
  XW(i) = 2*%pi;
else
    XW(i)=0;
end
end
XW = XW';
//Inverse Continuous-time Fourier Transform
t = -3*%pi:%pi/length(w):3*%pi;
xt =(1/(2*%pi))*XW *exp(sqrt(-1)*w'*t)*Dw;
xt = real(1+xt);
plot(t,xt);
xlabel('            t Sec');
title('Time domain signal x(t)')
```

Scilab code Exa 5.22 Inverse fourier transform

```
// Inverse Continuous Time Fourier Transform
// X(jW)= 2*pi , at W=-Wo
clear;
clc;
close;
// CTFT
```

```
A =1;
Dw = 0.005;
W1 = 4;
Wo=2//Assume Wo=2
w = -W1/2:Dw:W1/2;
for i = 1:length(w)
    if w(i)==-Wo then
  XW(i) = 2*%pi;
else
    XW(i)=0;
end
end
XW = XW';
//Inverse Continuous-time Fourier Transform
t = -3*%pi:%pi/length(w):3*%pi;
xt =(1/(2*%pi))*XW *exp(sqrt(-1)*w'*t)*Dw;
xt = real(1+xt);
plot(t,xt);
xlabel('              t Sec');
title('Time domain signal x(t)')
```

Scilab code Exa 5.24 Fourier Transform of periodic sinusoid

```
// Continuous Time Fourier Transforms of
// Sinusoidal waveforms sin(Wot)
clear
clc;
close;
T1 = 2;
T = 4*T1;
Wo = 2*%pi/T;
W = [-Wo,0,Wo];
ak = (2*%pi*Wo*T1/%pi)/sqrt(-1);
XW = [-ak,0,ak];
plot(W,-imag(XW),'.');
```

```
xlabel('
  W');
xtitle('CTFT of sin(Wot)','W','X(jW)')
```

Scilab code Exa 5.25 Fourier Transform of periodic signal

```
// Continuous Time Fourier Transforms of
// Sinusoidal waveforms cos(Wot)
clear;
clc;
close;
// CTFT
T1 = 2;
T = 4*T1;
Wo = 2*%pi/T;
W = [-Wo,0,Wo];
ak = (2*%pi*Wo*T1/%pi);
XW =[ak,0,ak];
plot(W,abs(XW),'.');
xlabel('                                W');
xtitle('CTFT of cos(Wot)','W','X(jW)')
```

Scilab code Exa 5.32 Fourier transform of impulse train

```
//CTFT of Periodic Impulse Train
clear;
clc;
close;
// CTFT
T = -4:4;;
T1 = 1;
xt = ones(1,length(T));
```

```
ak = 1/T1;
XW = 2*%pi*ak*ones(1,length(T));
Wo = 2*%pi/T1;
W = Wo*T;
figure
subplot(2,1,1)
plot2d3('gnn',T,xt);
xlabel('                              t');
title('Periodic Impulse Train')
subplot(2,1,2)
plot2d3('gnn',W,XW);
xlabel('                              t');
title('CTFT of Periodic Impulse Train')
```

Scilab code Exa 5.37 Frequency response of the system

```
//Continuous Time Signal x(t)= 0.5*exp(-B*t*0.5)u(t), t>0
clear;
clc;
close;
B =1;
Dt = 0.005;
t = 0:Dt:10;
h = 0.5*exp(-B*t*0.5);
Wmax = 2*%pi*1;
K = 4;
k = 0:(K/1000):K;
W = k*Wmax/K;
XW = h* exp(-sqrt(-1)*t'*W) * Dt;
XW_Mag = abs(XW);
W = [-mtlb_fliplr(W), W(2:1001)];
XW_Mag = [mtlb_fliplr(XW_Mag), XW_Mag(2:1001)];
[XW_Phase,db] = phasemag(XW);
XW_Phase = [-mtlb_fliplr(XW_Phase),XW_Phase(2:1001)
```

```
    ];
//Plotting Continuous Time Signal
figure(1)
plot(t,h);
xlabel('t in sec.');
ylabel('x(t)')
title('Continuous Time Signal')
figure(2)
//Plotting Magnitude Response of CTS
subplot(2,1,1);
plot(W,XW_Mag);
xlabel('Frequency in Radians/Seconds---> W');
ylabel('abs(X(jW))')
title('Magnitude Response (CTFT)')
//Plotting Phase Reponse of CTS
subplot(2,1,2);
plot(W,XW_Phase*%pi/180);
xlabel('                                  Frequency in
    Radians/Seconds---> W');
ylabel('
                                                          <X
    (jW)')
title('Phase Response(CTFT) in Radians')
```

Chapter 6

Sampling

Scilab code Exa 6.1 Sampling

```
//Sampling the CT signals
clc
clear
close
t=-0.3:0.0001:0.3;
x1=2*cos(2*%pi*20*t);//F1=20Hz
x2=2*cos(2*%pi*80*t);//F2=80Hz
figure(1)
subplot(2,1,1)
plot(t,x1);
xtitle("CT Signal X1(t)","t","x1(t)");
subplot(2,1,2)
plot(t,x2)
xtitle("CT Signal X2(t)","t","x2(t)");
//Given Sampling frequency Fs=60Hz
Fs=60;
n=-10:1:10;
Ts=1/60;//Sampling interval Ts=1/Fs
x1_n=2*cos(2*%pi*20*n*Ts);
x2_n=2*cos(2*%pi*80*n*Ts);
figure(2)
```

```
subplot(2,1,1)
plot2d3('gnn',n,x1_n,3);
xtitle("Sampled signal x1[n]","n","x1[n]")
subplot(2,1,2)
plot2d3('gnn',n,x2_n,3);
xtitle("Sampled signal x2[n]","n","x2[n]")
```

Scilab code Exa 6.2 Sampling

```
//Sampling the CT signals
clc
clear
close
t=-10:0.01:10;
x=sin(%pi*t);
figure(1)
subplot(2,1,1)
plot(t,x);
xtitle("CT Signal sin(pi*t)","t","x(t)");
Wb=%pi;//Given Sampling frequency is Pi radians
Ws=2*Wb;
Fs=Ws/(2*%pi);
n=-100:1:100;
Ts=1/Fs;//Sampling interval Ts=1/Fs
x_n=sin(%pi*n*Ts);
subplot(2,1,2)
plot2d(n,x_n,rect=[-100 -2 100 2]);
xtitle("Sampled signal x[n]","n","x[n]")
```

Scilab code Exa 6.3 Sampling

```
//Sampling the CT signals
clc
```

```
clear
close
t=-0.3:0.0001:0.3;
x=5*sin(10*%pi*t);
figure(1)
plot(t,x);
xtitle("CT Signal x(t)","t","x(t)");
//Given Sampling frequency (a) Fs=15Hz (b) Fs=6Hz
Fs1=15;
Fs2=6;
n=-10:1:10;
Ts1=1/Fs1;//Sampling interval Ts=1/Fs
Ts2=1/Fs2;
x1=5*sin(%pi*10*n*Ts1);
x2=5*sin(%pi*10*n*Ts2);
figure(2)
subplot(2,1,1)
plot2d3('gnn',n,x1);
xtitle("Sampled signal Fs=15Hz","n","x1[n]")
subplot(2,1,2)
plot2d3('gnn',n,x2);
xtitle("Sampled signal Fs=6Hz","n","x2[n]")
```

Scilab code Exa 6.4 Sampling

```
// Continuous Time Fourier Transforms of
// Sinusoidal waveforms 3cos(2*pi*t)
clear;
clc;
close;
// CTFT
t=-10:0.01:10;
x=3*cos(2*%pi*t);
subplot(2,1,1)
plot(t,x);
```

```
xtitle("CT signal x(t)","t","x(t)");
T1 = 2;
T = 4*T1;
Wo = 6*%pi/T;
W = [-Wo,0,Wo];
ak = (2*%pi*Wo*T1/%pi);
XW =[ak,0,ak];
subplot(2,1,2)
plot2d3('gnn',W,real(XW));
xlabel('                                    W');
xtitle('CTFT of cos(Wot)','W','X(jW)')
n=-10:10;
W1=4*%pi;
W2=8*%pi;
W3=3*%pi;
T1=(2*%pi)/W1;
T2=(2*%pi)/W2;
T3=(2*%pi)/W3;
x1=3*cos(2*%pi*n*T1);
x2=3*cos(2*%pi*n*T2);
x3=3*cos(2*%pi*n*T3);
figure(1)
subplot(3,1,1)
plot2d3('gnn',n,x1);
xtitle("X(t) sampled at Ws=4*pi","n","x1[n]");
subplot(3,1,2)
plot2d3('gnn',n,x2);
xtitle("X(t) sampled at Ws=8*pi","n","x2[n]");
subplot(3,1,3)
plot2d3('gnn',n,x3);
xtitle("X(t) sampled at Ws=3*pi","n","x3[n]");
```

Scilab code Exa 6.6 Sampling

```
//Sampling the signal at nyquist rate
```

```
clear;
clc;
close;
t=-1:0.01:1;
x=2*cos(200*%pi*t)+3*sin(100*%pi*t)-4*sin(500*%pi*t);
f1=100;
f2=50;
f3=250;
fb=max(f1,f2,f3);
Fs=2*fb;
Ts=1/Fs;
n=-10:10;
x_n=2*cos(200*%pi*n*Ts)+3*sin(100*%pi*n*Ts)-4*sin(500*%pi*n*Ts);
plot2d3('gnn',n,x_n)
xtitle("DT Signal x(n) sampled at nyquist rate","n","x[n]");
```

Scilab code Exa 6.7 Sampling

```
//Determining nyquist rate for the signals
clc
clear
close
Wb1=4*%pi;
Wb2=10*%pi;
Wbs=max(Wb1,Wb2);
Ws=2*Wbs;
//Bandlimited frequency doesnt change by Amplitude scaling
//(a) 2*x1(t)
Wa=2*Wb1
disp("Wa=")
disp(Wa)
```

```
//Timing shifting doesnt affect the magnitude spectrum
//(b) x1(t-1)
Wb=2*Wb1
disp("Wb=")
disp(Wb)
//Adding two band-limited spectrums will not sampling frequency
//(c) 2*x1(t)+x1(t-1)
Wc=2*Wb1
disp("Wc=")
disp(Wc)
//Compressing time axis expands frequency axis by the same factor
//(d) x2(2t)
Wd=2*2*Wb2
disp("Wd=")
disp(Wd)
//Expanding the time axis compresses the frequency axis by same factor
//(e) x2(t/2)
We=1/2*2*Wb2
disp("We=")
disp(We)
//(f) x2(2t)+x2(t/2)
Wf=max(Wd,We)
disp("Wf=")
disp(Wf)
//x1(t)x2(t)
Wg=2*(Wb1+Wb2)
disp("Wg=")
disp(Wg)
//x1(t)*x2(t)
Wh=2*min(Wb1,Wb2)
disp("Wh=")
disp(Wh)
//x1(t)*cos(2*%pi*t)
Wi=2*(Wb1+2*%pi)
```

```
disp("Wi=")
disp(Wi)
//x1'(t)
Wj=2*Wb1
disp("Wj=")
disp(Wj)
```

Chapter 7

Fourier Analysis of discretetime signals and systems

Scilab code Exa 7.3 Fourier series representation of DT signal

```
//DTFS of x[n] =2cos((pi/3)*n+(pi/6))
clear;
close;
clc;
n = -3:3;
N = 6;
Wo = 2*%pi/N;
xn = 2*cos((%pi/3)*n+(%pi/6));
//By euler's theorem X[n] can be represented
x_n=exp(%i*(%pi*n/3)+%pi/6)+exp(-%i*(%pi*n/3)+%pi/6)
for i=1:length(n)
    if n(i)==1
        a(i)=exp(%i*%pi/6);
    elseif n(i)==-1
        a(i)=exp(-%i*%pi/6);
    else
        a(i)=0;
    end
end
```

```
for i=1:length(a)
    if real(a(i))==0 then
        phase(i)=0;
    else
    phase(i)=atan(imag(a(i))/real(a(i)));
end
end
subplot(2,1,1)
plot2d3('gnn',n,abs(a))
xtitle("MAgnitude spectrum","k","|ak|")
subplot(2,1,2)
plot2d3('gnn',n,phase)
xtitle("Phase spectrum","k","angle(ak)")
```

Scilab code Exa 7.4 Fourier series representation of DT signal

```
//Fouries series representation of combinarion of
   signals
//x[n]=1+sin(pi*n/2)+cos(%pi*n/4)
clc
clear
close
n=-3:3;
x=1+sin(%pi*n/2)+cos(%pi*n/4);
w1=%pi/2;
w2=%pi/4;
N1=2*%pi/w1;
N2=2*%pi/w2;
N=max(N1,N2);
wo=2*%pi/N;
//Expanding x[n] by Euler's theorem
xn=1+0.5*exp(%i*wo*n)+0.5*exp(-%i*wo*n)-0.5*%i*exp(
   %i*2*wo*n)-0.5*%i*exp(-%i*2*wo*n);
a0=1;
a1=0.5;
```

```
a_1=0.5;
a2=1/2*%i;
a_2=-1/2*%i;
a=[a_2 a_1 a0 a1 a2];
a1=[0 a 0];
phase=[%pi/2 0 0 0 -%pi/2]
phase=[0 phase 0]
subplot(2,1,1)
plot(n,abs(a1),'.')
xtitle("magnitude spectrum","k","ak")
subplot(2,1,2)
plot(n,phase,'.')
xtitle("Phase spectrum","k","ak")
```

Scilab code Exa 7.5 Fourier series representation of DT signal

```
//DTFS coefficients of periodic square wave
clear;
close;
clc;
N = 10;
N1 = 2;
Wo = 2*%pi/N;
xn = ones(1,length(N));
n = -(2*N1+1):(2*N1+1);
a(1) = (2*N1+1)/N;
for k =1:2*N1
  a(k+1) = sin((2*%pi*k*(N1+0.5))/N)/sin(%pi*k/N);
  a(k+1) = a(k+1)/N;
  if(abs(a(k+1))<=0.1)
    a(k+1) =0;
  end
end
a =a';
a_conj =conj(a);
```

```
ak = [a_conj($:-1:1),a(2:$)];
k = -2*N1:2*N1;
plot2d3('gnn',k,abs(ak))
xtitle('Magnitude spectrum','k','|ak|')
```

Scilab code Exa 7.6 Fourier series representation of DT signal

```
//DTFS of a periodic sequence
clc
clear
close
n=-4:3;
x=[0 1 2 3 0 1 2 3];
N=4;
k=0:3;;
wo=2*%pi/N;
a0=1.5;
a1=-0.5+0.5*%i;
a2=-0.5;
a3=-0.5-0.5*%i;
a=[a0,a1,a2,a3]
for i=1:length(a)
    phase(i)=atan(imag(a(i))/real(a(i)));
end
subplot(2,1,1)
plot(k,abs(a),'.');
xtitle("magnitude spectrum","k","ak");
subplot(2,1,2)
plot(k,phase,'.');
xtitle("phase spectrum","k","ak");
```

Scilab code Exa 7.7 Fourier series representation of DT signal

```
//DTFS of discrete periodic signal
clc
clear
close
N=2//asume N=2
n=-2*N:2*N
for i=1:length(n)
    if modulo(n(i),N)==0 then
    x(i)=1;
else
    x(i)=0;
end
end
subplot(2,1,1)
plot(n,x,'.')
xtitle("Input signal x[n]","n","x[n]");
k=-5:5;
for i=1:length(k)
    ak(i)=1/N;
end
subplot(2,1,2)
plot(k,ak,'.')
xtitle("Frequency spectrum","k","ak")
```

Scilab code Exa 7.8 Fourier series representation of DT signal

```
//x[n] = 1+sin(2*%pi/N)n+3cos(2*%pi/N)n+cos[(4*%pi/N
   )n+%pi/4]
clear;
close;
clc;
N = 10;
n = 0:0.01:N;
Wo = 2*%pi/N;
xn =ones(1,length(n))+sin(Wo*n)+3*cos(Wo*n)+cos(2*Wo
```

```
    *n+%pi/4);
for k =0:N-2
  C(k+1,:) = exp(-sqrt(-1)*Wo*n.*k);
  a(k+1) = xn*C(k+1,:)'/length(n);
  if(abs(a(k+1))<=0.1)
    a(k+1)=0;
  end
end
a =a';
a_conj =conj(a);
ak = [a_conj($:-1:1),a(2:$)];
Mag_ak = abs(ak);
for i = 1:length(a)
  Phase_ak(i) = atan(imag(ak(i))/(real(ak(i))
      +0.0001));
end
Phase_ak = Phase_ak'
Phase_ak = [Phase_ak(1:$-1) -Phase_ak($:-1:1)];
k = -(N-2):(N-2);
subplot(2,1,1)
plot2d3('gnn',k,Mag_ak,5)
xtitle('abs(ak)','k','ak')
subplot(2,1,2)
plot2d3('gnn',k,Phase_ak,5)
xtitle('phase(ak)','k','ak')
```

Scilab code Exa 7.9 Fourier series representation of DT signal

```
//x[n] = 1+sin(4*%pi/N)n+cos(10*%pi/N)n
clear;
close;
clc;
N = 21;
n = 0:0.01:N;
Wo = 2*%pi/N;
```

```
xn =ones(1,length(n))+sin(2*Wo*n)+cos(5*Wo*n);
for k =0:N-2
  C(k+1,:) = exp(-sqrt(-1)*Wo*n.*k);
  a(k+1) = xn*C(k+1,:)'/length(n);
  if(abs(a(k+1))<=0.1)
    a(k+1)=0;
  end
end
a =a';
a_conj =conj(a);
ak = [a_conj($:-1:1),a(2:$)];
Mag_ak = abs(ak);
for i = 1:length(a)
  Phase_ak(i) = atan(imag(ak(i))/(real(ak(i))
     +0.0001));
end
Phase_ak = Phase_ak'
Phase_ak = [Phase_ak(1:$-1) -Phase_ak($:-1:1)];
k = -(N-2):(N-2);
subplot(2,1,1)
plot2d3('gnn',k,Mag_ak,5)
xtitle('abs(ak)','k','ak')
subplot(2,1,2)
plot2d3('gnn',k,Phase_ak,5)
xtitle('phase(ak)','k','ak')
```

Scilab code Exa 7.10 DTFSrepresentation

```
//x[n] = 0.5+0.5*cos(2*%pi/N)n
clear;
close;
clc;
N = 8;
n = 0:0.01:N;
Wo = 2*%pi/N;
```

```
xn =0.5*ones(1,length(n))+0.5*cos(Wo*n);
for k =0:N-2
  C(k+1,:) = exp(-sqrt(-1)*Wo*n.*k);
  a(k+1) = xn*C(k+1,:)'/length(n);
  if(abs(a(k+1))<=0.1)
    a(k+1)=0;
  end
end
a =a';
a_conj =conj(a);
ak = [a_conj($:-1:1),a(2:$)];
Mag_ak = abs(ak);
k = -(N-2):(N-2);
plot2d3('gnn',k,Mag_ak,5)
xtitle('abs(ak)','k','ak')
```

Scilab code Exa 7.16 Discretetime fourier transform

```
//Discrete Time Fourier Transform of discrete
   sequence
//x[n]= (a^n).u[n], |a|<1
clear;
clc;
close;
a1 = 0.5;
max_limit = 10;
for n = 0:max_limit-1
  x1(n+1) = (a1^n);
end
n = 0:max_limit-1;
Wmax = 2*%pi;
K = 4;
k = 0:(K/1000):K;
W = k*Wmax/K;
x1 = x1';
```

```
XW1 = x1* exp(-sqrt(-1)*n'*W);
XW1_Mag = abs(XW1);
W = [-mtlb_fliplr(W), W(2:1001)]; // Omega from -
   Wmax to Wmax
XW1_Mag = 2.5*[mtlb_fliplr(XW1_Mag), XW1_Mag(2:1001)
   ];
[XW1_Phase,db] = phasemag(XW1);
XW1_Phase = (1/30)*[-mtlb_fliplr(XW1_Phase),
   XW1_Phase(2:1001)];
subplot(3,1,1);
plot2d3('gnn',n,x1);
xtitle('Discrete Time Sequence x[n]')
subplot(3,1,2);
plot2d(W,XW1_Mag);
title('Magnitude Response abs(X(jW))')
subplot(3,1,3);
plot2d(W,XW1_Phase);
title('Phase Response <(X(jW))')
```

Scilab code Exa 7.17 Discretetime fourier transform

```
//Discrete Time Fourier Transform of discrete
   sequence
//x[n]= (a^n).u[-n], |a|>1
clear;
clc;
close;
a1 = 3;
min_limit = -20;
n = min_limit:0
for i=1:length(n)
  x1(i) = (a1^n(i));
end
Wmax = 2*%pi;
K = 4;
```

```
k = 0:(K/1000):K;
W = k*Wmax/K;
x1 = x1';
XW1 = x1* exp(-sqrt(-1)*n'*W);
XW1_Mag = abs(XW1);
W = [-mtlb_fliplr(W), W(2:1001)]; // Omega from -
    Wmax to Wmax
XW1_Mag = [mtlb_fliplr(XW1_Mag), XW1_Mag(2:1001)];
[XW1_Phase,db] = phasemag(XW1);
XW1_Phase = [-mtlb_fliplr(XW1_Phase),XW1_Phase
    (2:1001)];
subplot(3,1,1);
plot2d3('gnn',n,x1);
xtitle('Discrete Time Sequence x[n]','n','x[n]')
subplot(3,1,2);
plot2d(W,XW1_Mag);
xtitle('Magnitude Response abs(X(jW))','w','|X(jW)|'
    )
subplot(3,1,3);
plot2d(W,XW1_Phase);
xtitle('Phase Response <(X(jW))','w','<(X(jW))')
```

Scilab code Exa 7.18 Discretetime fourier transform

```
//Discrete Time Fourier Transform of
//x[n]= (a^abs(n)) |a|<1
clear;
clc;
close;
// DTS Signal
a = 0.5;
max_limit = 10;
n = -max_limit+1:max_limit-1;
x = a^abs(n);
// Discrete-time Fourier Transform
```

```
Wmax = 2*%pi;
K = 4;
k = 0:(K/1000):K;
W = k*Wmax/K;
XW = x* exp(-sqrt(-1)*n'*W);
XW_Mag = real(XW);
W = [-mtlb_fliplr(W), W(2:1001)]; // Omega from -
    Wmax to Wmax
XW_Mag = [mtlb_fliplr(XW_Mag), XW_Mag(2:1001)];
//plot for abs(a)<1
figure
subplot(2,1,1);
plot2d3('gnn',n,x);
xtitle('Discrete Time Sequence x[n] for a>0','n','x[
    n]')
subplot(2,1,2);
plot2d(W,XW_Mag);
xtitle('Discrete Time Fourier Transform X(exp(jW))',
    'w','|X(exp(jW))|')
```

Scilab code Exa 7.19 Discretetime fourier transform

```
//Discrete Time Fourier Transform of
//x[n]= 1 , abs(n)<=M1
clear;
clc;
close;
// DTS Signal
M1 = 2;
n = -M1:M1;
x = ones(1,length(n));
Wmax = 2*%pi;
K = 4;
k = 0:(K/1000):K;
W = k*Wmax/K;
```

```
XW = x* exp(-sqrt(-1)*n'*W);
XW_Mag = real(XW);
W = [-mtlb_fliplr(W), W(2:1001)]; // Omega from -Wmax to Wmax
XW_Mag = [mtlb_fliplr(XW_Mag), XW_Mag(2:1001)];
//plot for abs(a)<1
figure
subplot(2,1,1);
plot2d3('gnn',n,x,2);
xtitle('Discrete Time Sequence x[n]','n','x[n]')
subplot(2,1,2);
plot2d(W,XW_Mag);
xtitle('Discrete Time Fourier Transform X(exp(jW))','w','|X(exp(jW))|')
```

Scilab code Exa 7.24 Fourier transform

```
//Discrete Time Fourier Transform of
// Periodic Impulse Train
clear;
clc;
close;
N = 5;
N1 = -3*N:3*N;
xn = [zeros(1,N-1),1];
x = [1 xn xn xn xn xn xn];
ak = 1/N;
XW = 2*%pi*ak*ones(1,2*N);
Wo = 2*%pi/N;
n  = -N:N-1;
W = Wo*n;
figure
subplot(2,1,1)
plot2d3('gnn',N1,x,2);
xtitle('Periodic Impulse Train','n','x[n]')
```

```
subplot(2,1,2)
plot2d3('gnn',W,XW,2);
xtitle('DTFT of Periodic Impulse Train','w','|X(exp(
    jw))|')
disp(Wo)
```

Scilab code Exa 7.26 Discretetime fourier transform

```
//Discrete Time Fourier Transform of discrete
    sequence
//x[n]= 1, n=2
clear;
clc;
close;
a1 = 1/8;
max_limit = 10;
for n = 0:max_limit-1
    if n==2 then
    x1(n+1) = 1;
else
    x1(n+1) = 0;
end
end
n = 0:max_limit-1;
Wmax = 2*%pi;
K = 4;
k = 0:(K/1000):K;
W = k*Wmax/K;
x1 = x1';
XW1 = x1* exp(-sqrt(-1)*n'*W);
XW1_Mag = abs(XW1);
W = [-mtlb_fliplr(W), W(2:1001)]; // Omega from -
    Wmax to Wmax
XW1_Mag = [mtlb_fliplr(XW1_Mag), XW1_Mag(2:1001)];
[XW1_Phase,db] = phasemag(XW1);
```

```
XW1_Phase = [-mtlb_fliplr(XW1_Phase),XW1_Phase
   (2:1001)];
subplot(3,1,1);
plot2d3('gnn',n,x1);
xtitle('Discrete Time Sequence x[n]')
subplot(3,1,2);
plot2d(W,XW1_Mag);
title('Magnitude Response abs(X(jW))')
subplot(3,1,3);
plot2d(W,XW1_Phase);
title('Phase Response <(X(jW))')
```