Scilab Textbook Companion for Analog and Digital Communications by H. P. Hsu[1]

Created by
K Anusha
B.Tech
Electronics Engineering
ABES Engineering College, Ghaziabad
College Teacher
None
Cross-Checked by
K. V. P. Pradeep

June 1, 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: Analog and Digital Communications

Author: H. P. Hsu

Publisher: Tata McGraw-Hill, New Delhi

Edition: 3

Year: 2010

ISBN: 978-0-07-015150-5

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

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

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

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

Contents

List of Scilab Codes

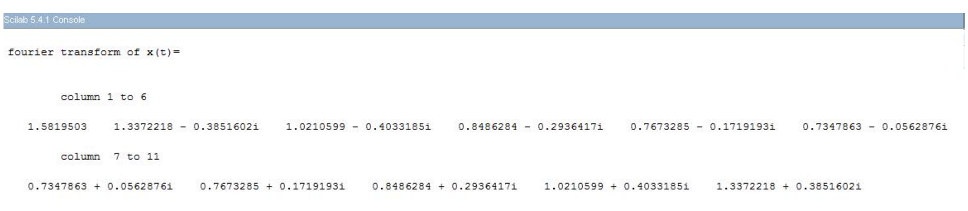
List of Figures

Chapter 1

Signals and Spectra

Scilab code Exa 1.9 FT of exponential with step function

```
//Page Number: 1.13
//Example 1.9
clc;
//Given,
//Signal is x(t)= e^(-at) * u(t)
//unity function u(t)=1 for 0 to infinity
//therefore
x=1;
//We assume 'infinity' value as 10 and the value of
```


```
Scilab 5.4.1 Console

fourier transform of x(t)=

      column 1 to 6

  1.5819503    1.3372218 - 0.3851602i    1.0210599 - 0.4033185i    0.8486284 - 0.2936417i    0.7673285 - 0.1719193i    0.7347863 - 0.0562876i

      column  7 to 11

  0.7347863 + 0.0562876i    0.7673285 + 0.1719193i    0.8486284 + 0.2936417i    1.0210599 + 0.4033185i    1.3372218 + 0.3851602i

-->
```

Figure 1.1: FT of exponential with step function

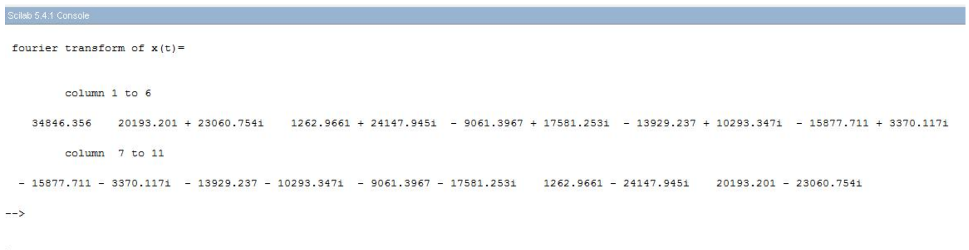


Figure 1.2: Exponential function

```
    'a' is 1
t= 0:1:10;
a=1;// a >0
z=((%e)^(-a*t) * x);
y=fft(z);
disp(y,'fourier transform of x(t)=');
```

Scilab code Exa 1.10 Exponential function

```
//Page Number: 1.14
//Example 1.10
clc;
//Given,
//Signal is x(t)= e^|-a|t * u(t)
//unity function u(t)=1 for 0 to infinity
//therefore
x=1;
//We assume 'infinity' value as 10 and the value of
    'a' is 1
t= 0:1:10;
a1=1;// For a >0
a2=-1; //For a <0
z=((%e)^(a2*t) * x)+((%e)^(a1*t) * x);
y=fft(z);
```

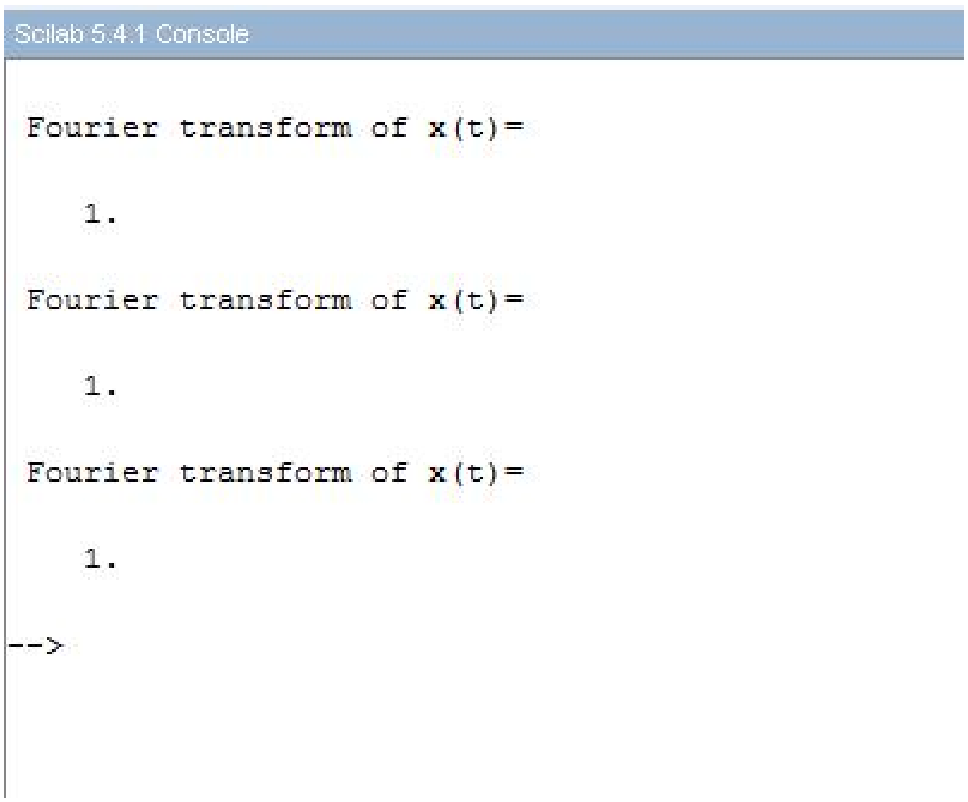


Figure 1.3: Rectangular Function

```
disp(y,'fourier transform of x(t)=');
```

Scilab code Exa 1.11 Rectangular Function

```
//Page Number: 1.14
//Example 1.11
clc;

//(a)
```

```

//Given
//Signal is x(t) = rect(t)
//rect(t) = 1 for -a< |t| < a and 0 elsewhere
//Therefore
//We find out fourier transform of x(t)= 1 for -a< |t| < a thus,
x=1;
a= 200; //Assume
t= -a : 1 : a; //range for fourier transform
y=fft(x);
disp(y,'Fourier transform of x(t)=');


//(b)

//Given
//Signal is x(t) = rect(t)
//rect(t) = 1 for -a/4< |t| < a/4 and 0 elsewhere
//Therefore
//We find out fourier transform of x(t)= 1 for -a/4< |t| < a/4 thus,
x=1;
a= 200; //Assume
t= -a/4 : 1 : a/4;//range for fourer transform
y=fft(x);
disp(y,'Fourier transform of x(t)=');

//(c)

//Given
//Signal is x(t) = rect(t)
//rect(t) = 1 for b < |t| < b + a/2 and 0 elsewhere
//Therefore
//We find out fourier transform of x(t)= 1 for b < |t| < b+ a/2 thus,
x=1;
a= 200; //Assume
```

```
b=100; //Assume
t= b : 1 : (b+(a/2));//range for fourer transform
y=fft(x);
disp(y,'Fourier transform of x(t)=');
```

Chapter 3

Amplitude Modulation

Scilab code Exa 3.7 Efficiency

```
//Page Number: 3.14
//Example 3.7
clc;
//Given

//(a)efficiency=((u^2)/(2+(u^2)))
u=0.5;
n=((u^2)/(2+(u^2)));
np=n*100;
disp("%", np, "Efficiney: ");

//(b)nmax,
//nmax occurs at u=1;
u1=1;
nmax=((u1^2)/(2+(u1^2)));
nmaxp=nmax*100;
disp("%", nmaxp, "Efficiney max: ");
```

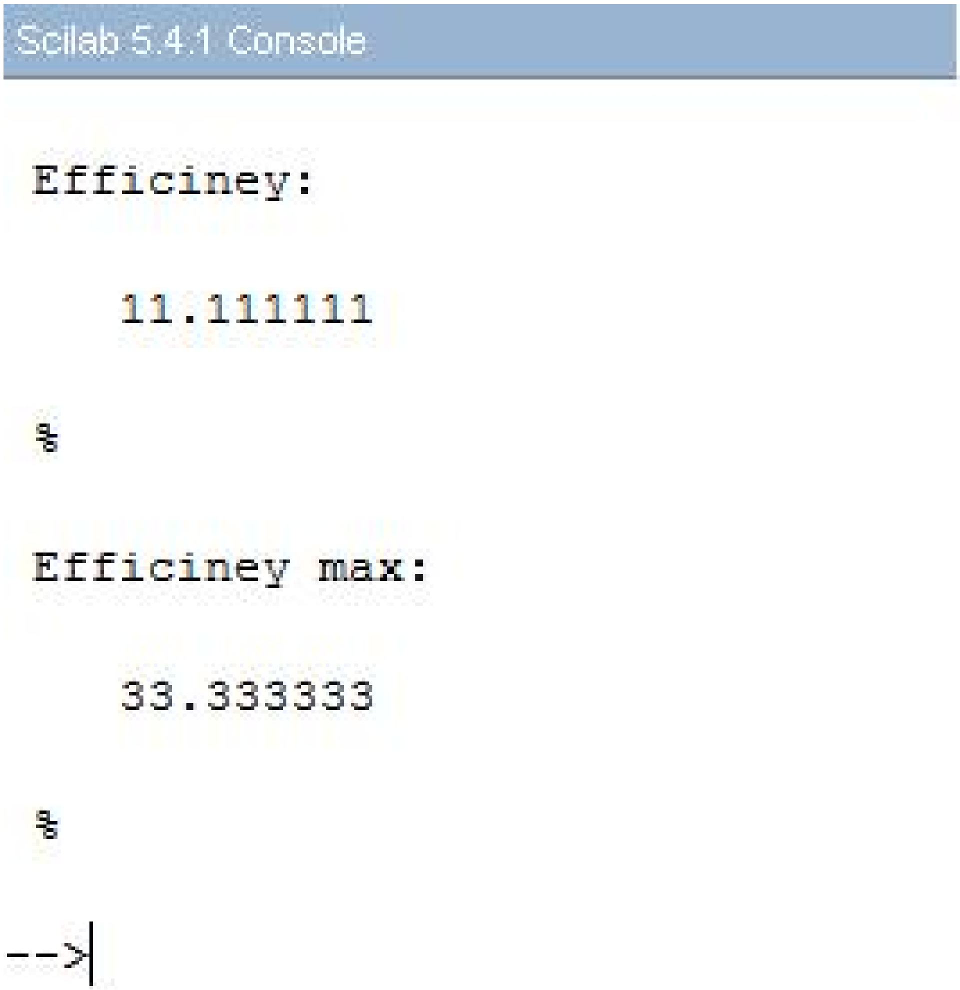


Figure 3.1: Efficiency

Scilab code Exa 3.8 Sine wave in AM broadcasting

```
//Page Number: 3.14
//Example 3.8
clc;
//Given
//From waveform
Amax=150;
Amin=30;

//(a)Modulation index
u=((Amax-Amin)/(Amax+Amin));
disp(u,"Modulation Index: ");

Ac=(Amax/(1+u));
//(b)
//Carrier Power
Pc=(Ac^2)/2;
disp("W",Pc,"Carrier Power: ");

//Side band Power
PSB=(Amin^2)/2;
disp("W",PSB,"USB=LSB Power: ");

//Total Average power
Pt=Pc+(2*PSB);
disp("W",Pt,"Total Average Power: ");

//(c)Peak Envelope Power
//Given
R=60; //Ohm
PEP=(Amax^2)/(2*R);
```

```
Scilab 5.4.1 Console

 Modulation Index:

    0.6666667

 Carrier Power:

    4050.

 W

 USB=LSB Power:

    450.

 W

 Total Average Power:

    4950.

 W

 Peak Envelope Power:

    187.5

 W

 Modulation efficieny:

    0.0909091

 A=

    210.
```



```
 A=

  - 15.

-->
```

```
disp("W",PEP,"Peak Envelope Power: ");

//(d) Modulation Efficieny
n=PSB/Pt;
disp(n,"Modulation efficieny: ");

//(e) Given
//(i) u=0.2
u1=0.2;
A1=(60/u1)-Ac;
disp(A1, "A=");

//(ii) u=0.8
u2=0.8;
A2=(60/u2)-Ac;
disp(A2, "A=");
```

Chapter 4

Angle Modulation

Scilab code Exa 4.3 Instantaneous frequency

```
//Page Number: 4.9
//Example 4.3
clc;
//Given
fc=1D+6; //Hz
kf=5;
mt=1D+5; //Hz

//(a) mi(t) with fm
mi=(fc+(kf*mt));
disp("Hz",mi,"Max, Inst. Frequency with FM");

kp=3;
//(b) mi2(t) with pm
mi2=fc+(mt*(kp/(2*%pi)));

disp("Hz",mi2,"Max, Inst. Frequency with PM");
```

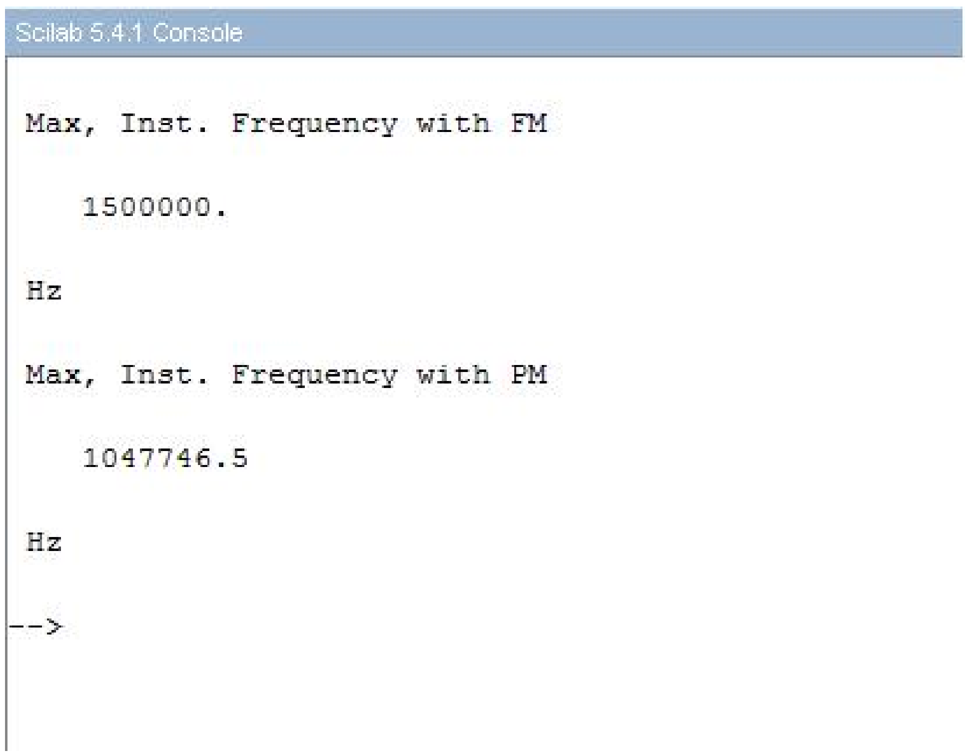


Figure 4.1: Instantaneous frequency

Scilab code Exa 4.9 Peak frequency

```
//Page Number: 4.13
//Example 4.9
clc;
//Given
delf=20D+3; //hz
fm=10D+3; //Hz

B=delf/fm;
disp(B, "Beta: ");
```

Scilab code Exa 4.13 Bandwidth

```
//Page Number: 4.16
//Example 4.13
clc;
//Given
//x(t)=10cos((2*pi*10^8*t)+(200cos(2*pi*10^3*t)))
//on differentiating
//wi=2*pi*(1D+8)-4*pi*sin(2*pi*(1D+3)*t)
//Therefore
delw=4*%pi*(1D+5);
wm=2*%pi*(1D+3);

B=delw/wm;
wb=2*(B+1)*wm;
fb=wb/2*%pi;
```

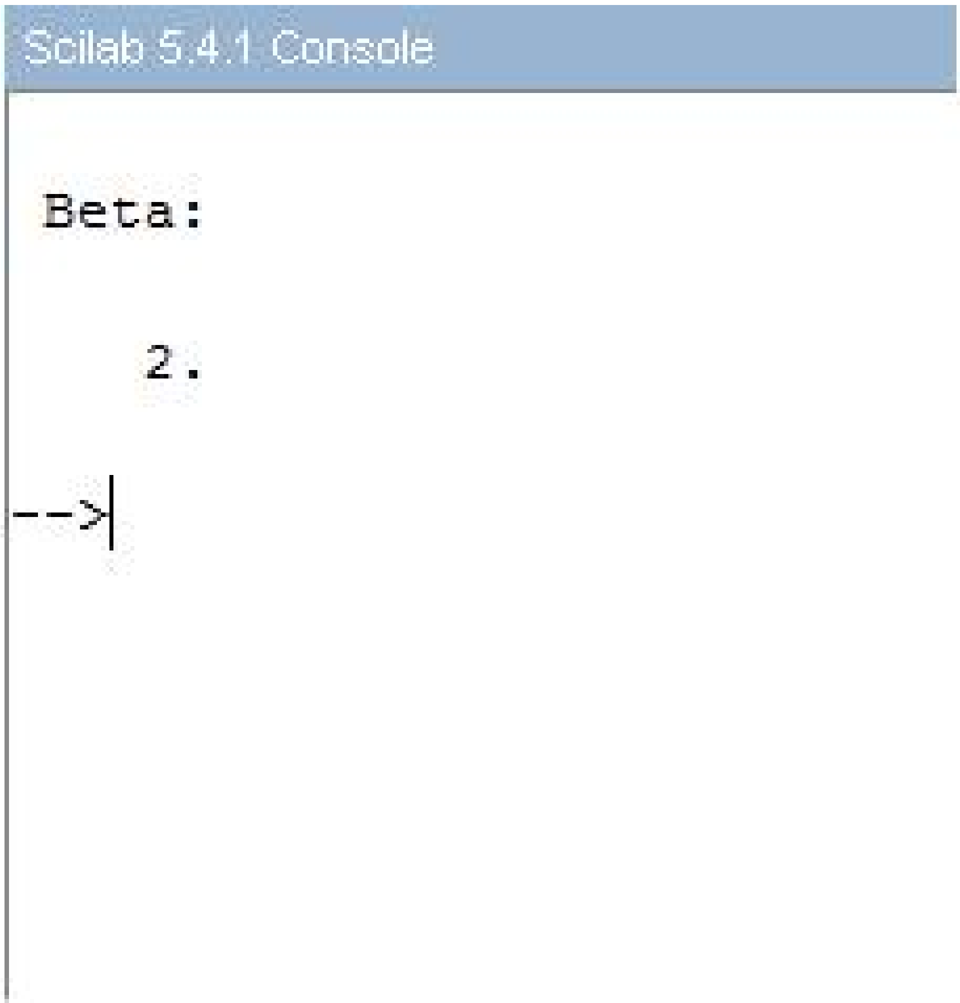


Figure 4.2: Peak frequency

```
Scilab 5.4.1 Console

 Wb

    2525840.5

 rad/s

 Fb

    3967581.

 Hz

-->
```

Figure 4.3: Bandwidth

```

disp("rad/s",wb, "Wb");

disp("Hz",fb, "Fb");
```

Scilab code Exa 4.14 Modulation Index

```
//Page Number: 4.17
//Example 4.14
clc;

//Given
delf=100D+3; //Hz
fc=20D+6; //Hz

//As B=delf/fm;
//(a) fm1=1D+3hz
disp('Part a')
fm1=1D+3; //Hz
B1=delf/fm1;
disp(B1,'Modulation Index');

fb1=2*delf;
disp('Hz',fb1,'Bandwidth');

//(b) fm2=100D+3hz
disp('Part b')
fm2=100D+3; //Hz
B2=delf/fm2;
disp(B2,'Modulation Index');

fb2=2*(B2+1)*fm2;
disp('Hz',fb2,'Bandwidth');
```

Scilab 5.4.1 Console

```
Part a

Modulation Index

   100.

Bandwidth

   200000.

Hz

Part b

Modulation Index

   1.

Bandwidth

   400000.

Hz

Part c

Modulation Index

   0.2

Bandwidth
```

```
//(c) fm3=500D+3hz
disp('Part c')
fm3=500D+3; //Hz
B3=delf/fm3;
disp(B3,'Modulation Index');

fb3=2*fm3;
disp('Hz',fb3,'Bandwidth');
```

Scilab code Exa 4.15 changed fm

```
//Page Number: 4.17
//Example 4.15
clc;
//Given
//x(t)=10cos(wct+3sinwmt)
//Comparing with standard equation
B=3;
fm=1D+3; //hz
fb=2*(B+1)*fm;

//(a)fm is doubled
fma=2*fm;
fba=2*(B+1)*fma;
disp(fba,"fb with 2fm: ");



//(b)fm is one halved
fmb=fm/2;
fbb=2*(B+1)*fmb;
disp(fbb,"fb with 0.5fm: ");
```

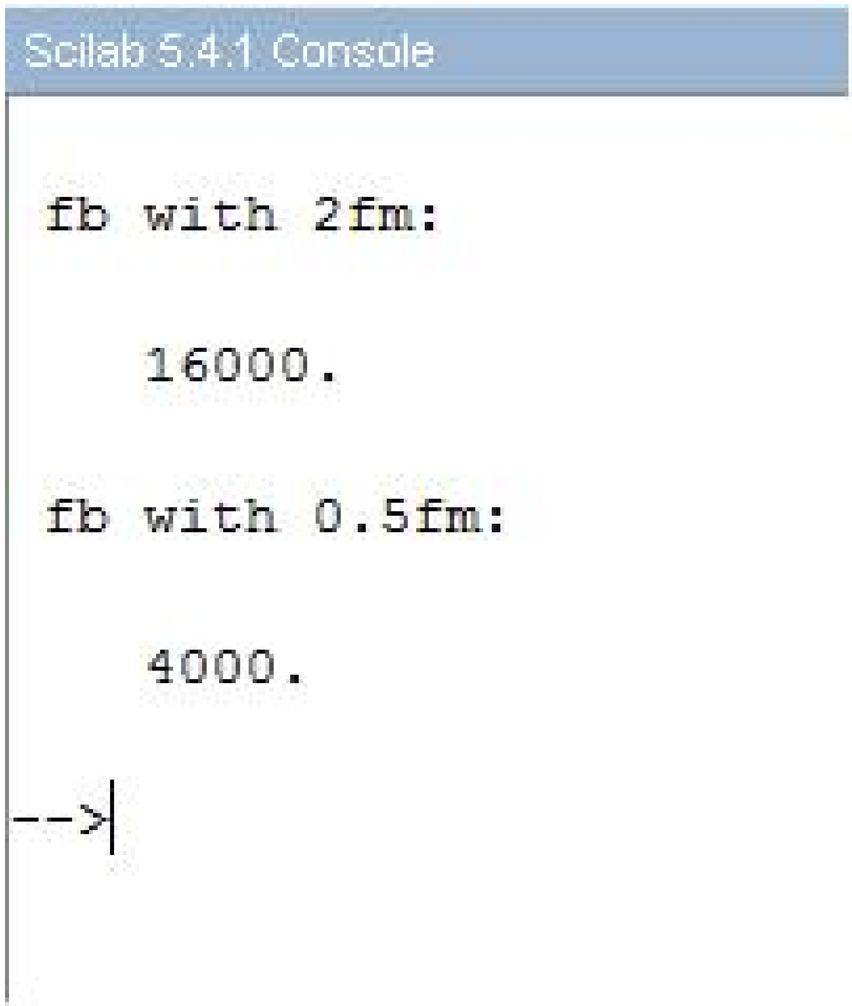


Figure 4.5: changed fm

Scilab code Exa 4.16 Using FM

```
//Page Number: 4.18
//Example 4.15
clc;
//Given
//x(t)=10cos(wct+3sinwmt)
//Comparing with standard equation of fm
B=3;
fm=1D+3; //hz
fb=2*(B+1)*fm;

//B is inversaly proportional to fm

//(a)fm is doubled
Ba=B/2;
fma=2*fm;;
fba=2*(Ba+1)*fma;
disp(fba,"fb with 2fm: ");



//(b)fm is one halved
Bb=2*B;
fmb=fm/2;
fbb=2*(Bb+1)*fmb;
disp(fbb,"fb with 0.5fm: ");
```

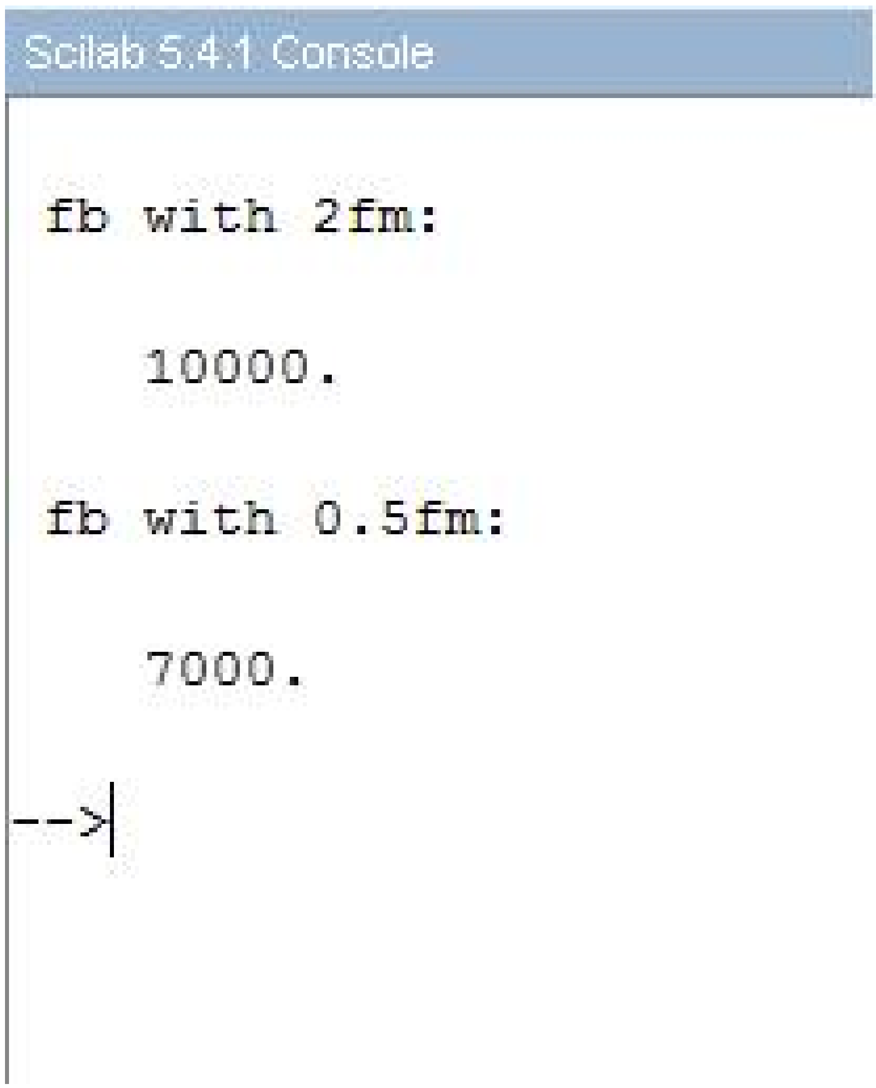


Figure 4.6: Using FM

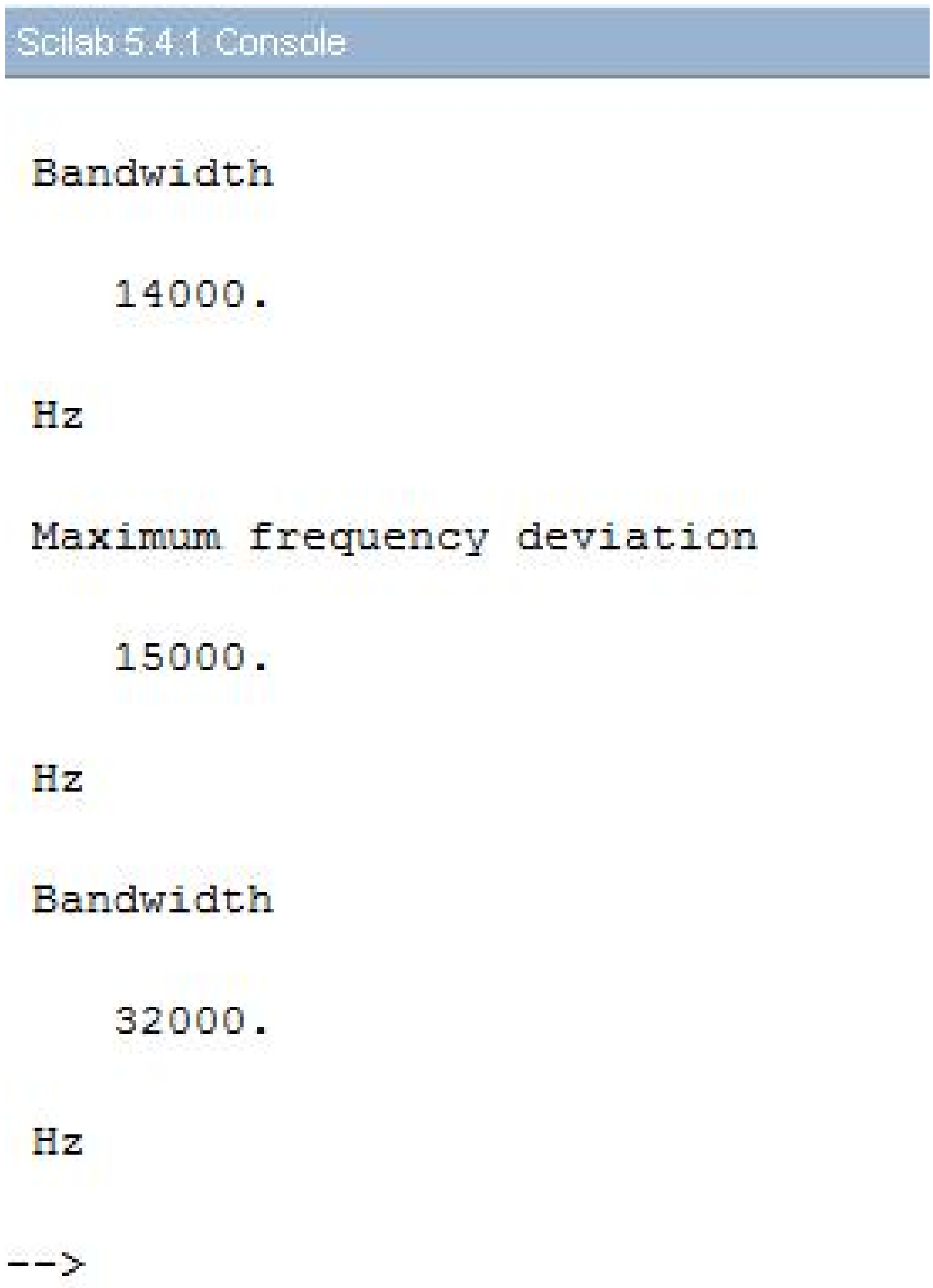


Figure 4.7: Frequency Modulated

Scilab code Exa 4.17 Frequency Modulated

```
//Page Number: 4.18
//Example 4.17
clc;
//Given
fm=2D+3; //Hz
delf=5D+3; //Hz

//(a) Bandwidth of modulated signal
B=delf/fm;

fb=2*(B+1)*fm;
disp('Hz',fb,'Bandwidth');

//(b)Max. frequency deviation and Bandwidth of new
    signal
//Given
fm1=fm-(1D+3);
delf1=3*delf;

B1=delf1/fm1;

fd=B1*fm1;
disp('Hz',fd,'Maximum frequency deviation');

fb1=2*(B1+1)*fm1;
disp('Hz',fb1,'Bandwidth');
```

Scilab code Exa 4.18 Carsons rule

```
//Page Number: 4.19
//Example 4.18
```

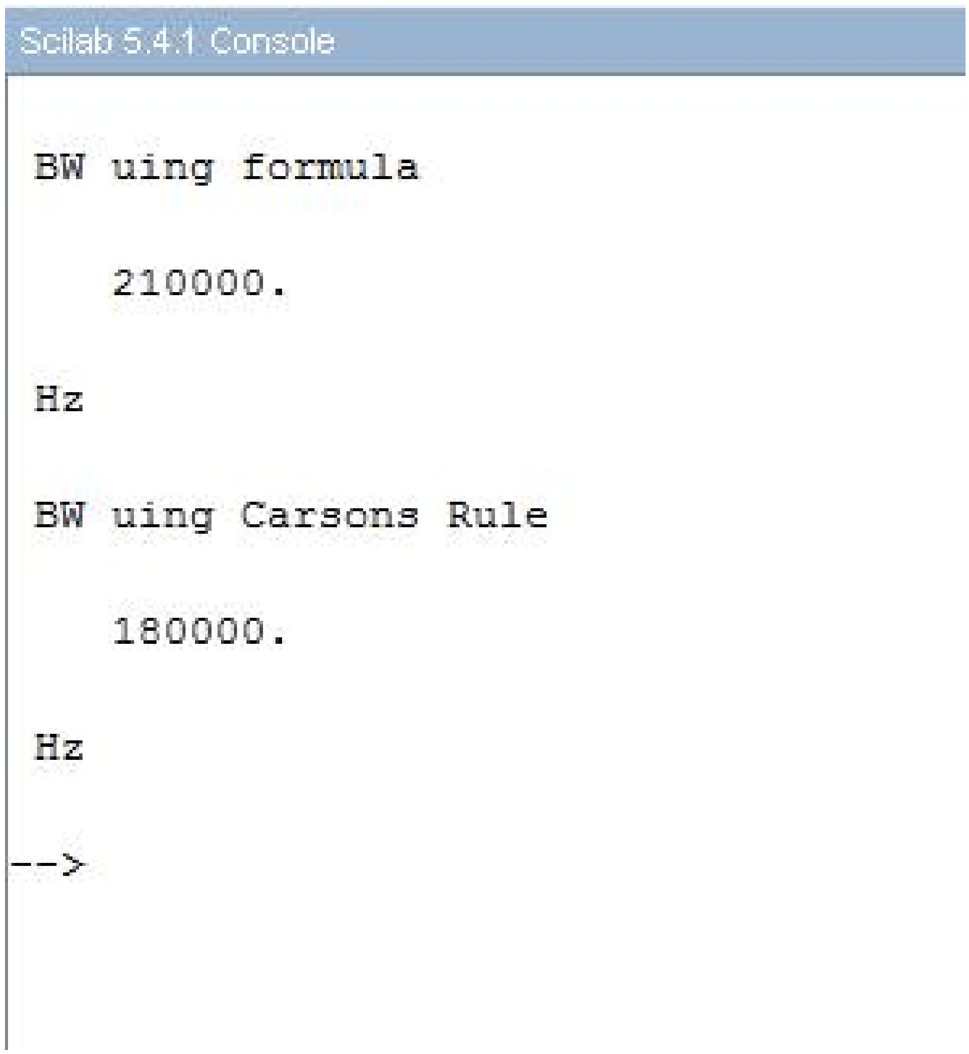


Figure 4.8: Carsons rule

```
clc;
//Given
delf=75D+3; //Hz
fM=15D+3; //Hz

D=delf/fM;
//Given formula fb=2(D+2)*fM
fb1=2*(D+2)*fM;
disp('Hz',fb1,'BW uing formula');

//Carsons Rule
fb2=2*(D+1)*fM;
disp('Hz',fb2,'BW uing Carsons Rule');

//High quality Fm radios require minimum 200kHz
//Therefore, carsons rule underestimates bandwidth
```

Scilab code Exa 4.19 Frequency Multiplier

```
//Page Number: 4.19
//Example 4.19
clc;
//Given
fm1=50; //Hz
fm2=15D+3; //Hz

delf=75D+3; //Hz

//As B=delf/fm
Bmin=delf/fm2;
Bmax=delf/fm1;

//Let B1=0.5
```

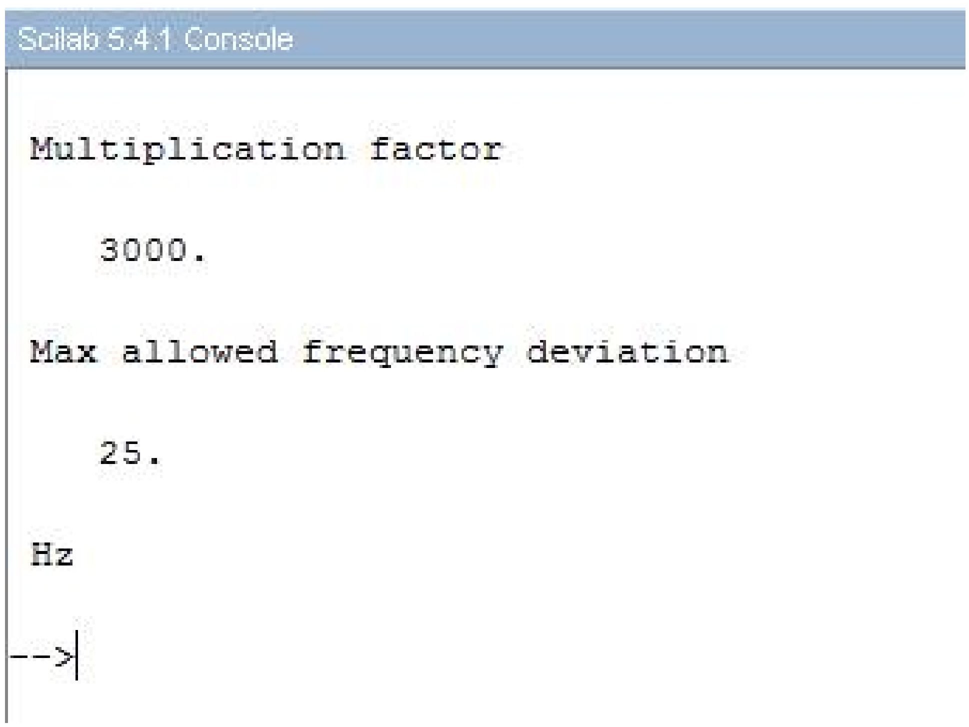


Figure 4.9: Frequency Multiplier

```
B1=0.5;
n=(Bmax/B1);
disp(n,'Multiplication factor');

delf1=(delf/n);
disp('Hz',delf1,'Max allowed frequency deviation');
```

Scilab code Exa 4.20 FM Transmitter

```
//Page Number: 4.20
//Example 4.20
clc;
//Given
f1=2D+5; //Hz
fLO=10.8D+6; //Hz
delf1=25; //Hz
n1=64;
n2=48;

delf=(delf1*n1*n2);
disp('Hz',delf,'Maximum frequency deviation');

f2=n1*f1;

f3a=f2+fLO;
f3b=f2-fLO;

//For f3a
fca=n2*f3a;
disp('Hz',fca,'Carrier frequency 1');

//For f3b
fcb=n2*f3b;
```

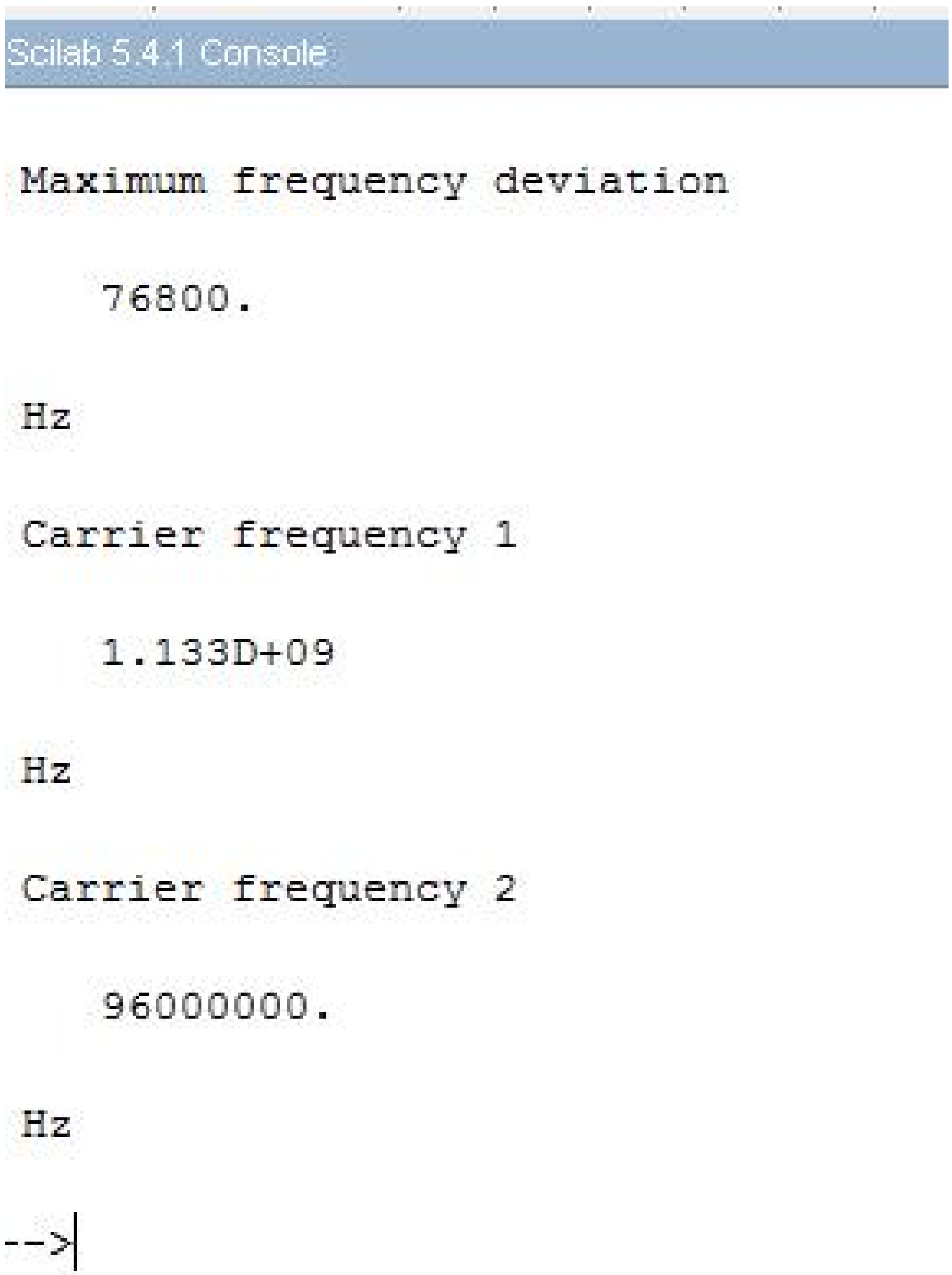


Figure 4.10: FM Transmitter

```
disp('Hz',fcb,'Carrier frequency 2');
```

Scilab code Exa 4.21 FM Modulator

```
//Page Number: 4.20
//Example 4.21
clc;
//Given
delf=20D+3; //Hz
fc=200D+3; //Hz
of=96D+6; //hz
//delf=n1*n2 and as only doublers are used, n1*n2
   has to be power of 2
//By trail and error, we find
n1=64;
n2=32;
//Output of first Multiplier
o1=n1*fc;
disp('Hz',o1,'Output of first multiplier: ');
i2=of/n2;
flo=o1-i2;
disp('Hz',flo,'fLO');
```

Scilab code Exa 4.22 FM generator

```
//Page Number: 4.20
//Example 4.22
clc;
//Given
```

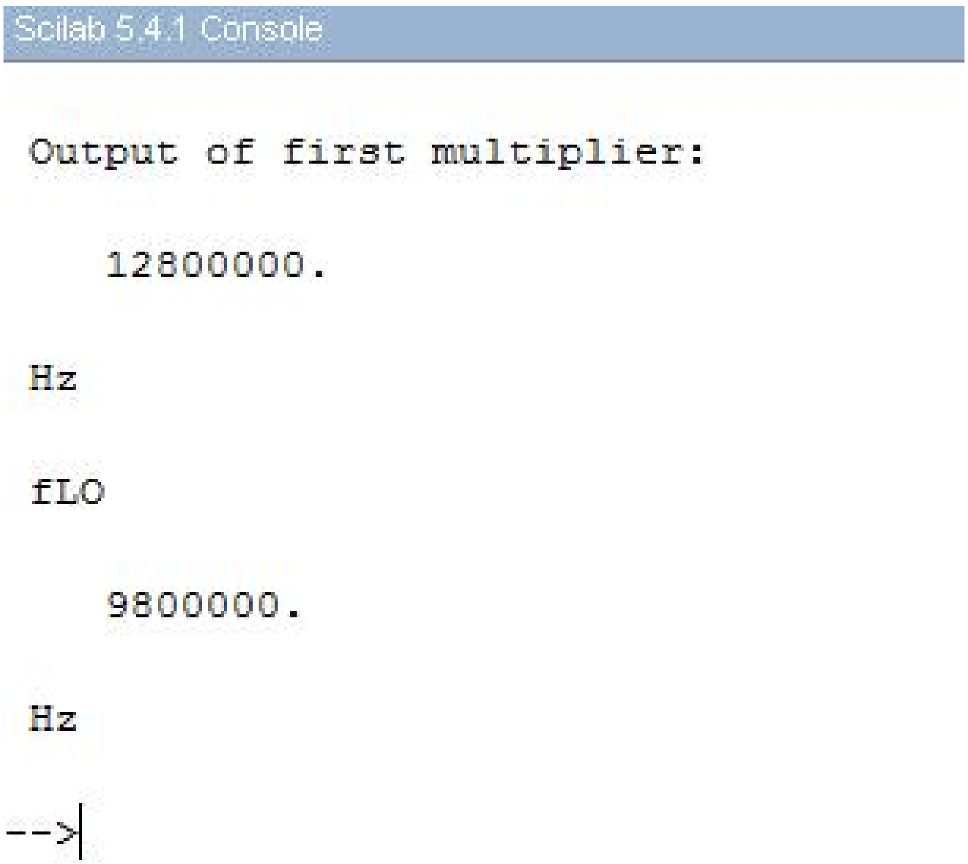


Figure 4.11: FM Modulator

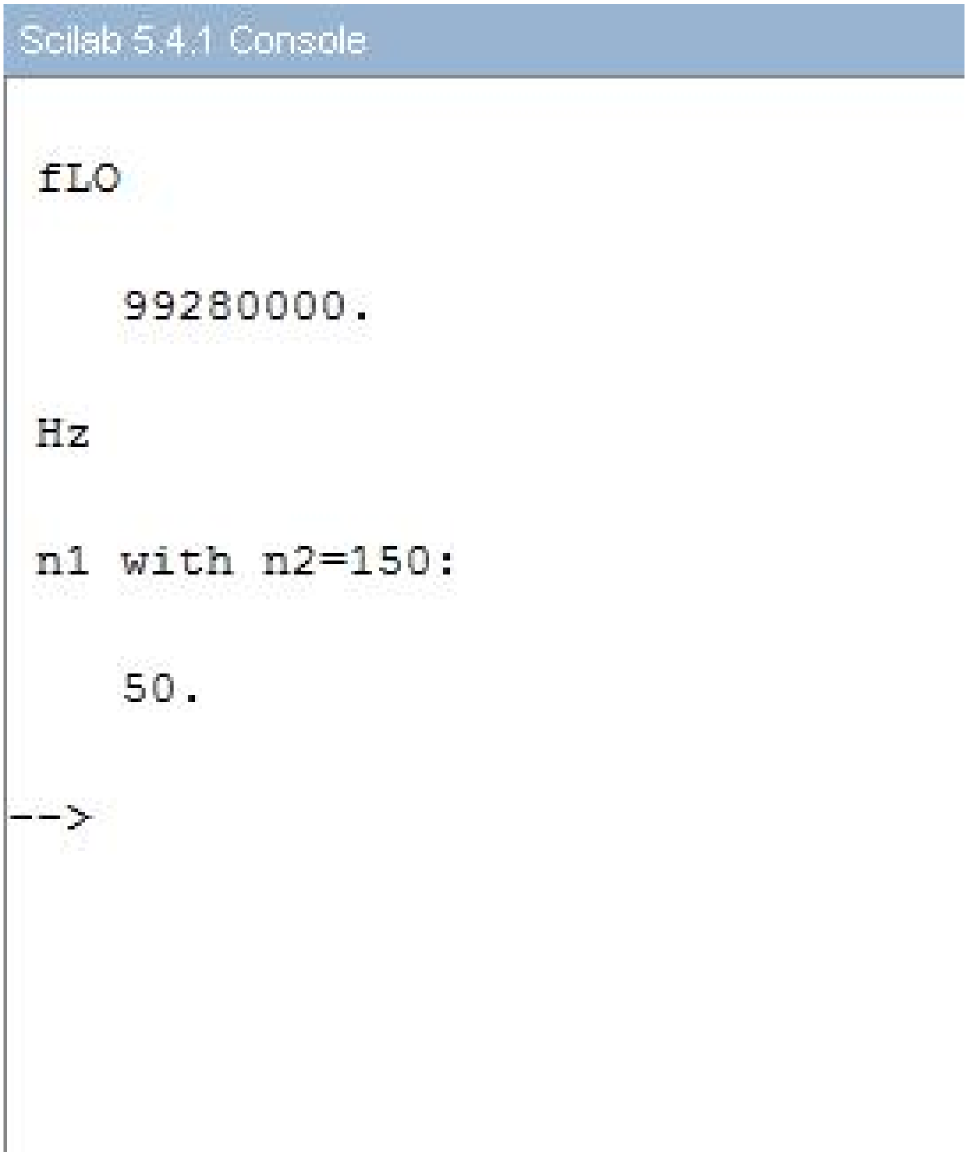


Figure 4.12: FM generator

```
B=0.2;
f1=200D+3; //Hz
fml=50; //Hz
fmh=15D+3; //Hz
delf=75D+3; //hz
fc=108D+6; //Hz

delf1=B*fml;
n1n2=delf/delf1;

//Let n2=150
n2=150;
flo=((delf*f1)-fc)/n2;
disp('Hz',flo,'fLO');

n1=n1n2/n2;
disp(n1, "n1 with n2=150:")
```

Scilab code Exa 4.23 Multiplication Factor

```
//Page Number: 4.21
//Example 4.23
clc;
//Given,

delfd1=50; //Hz
f1=120; //Hz

delfd2=20000; //Hz
f2=240; //Hz
//(a)PM
delf1=(f2/f1)*delfd1;
n1=delfd2/delf1;
```

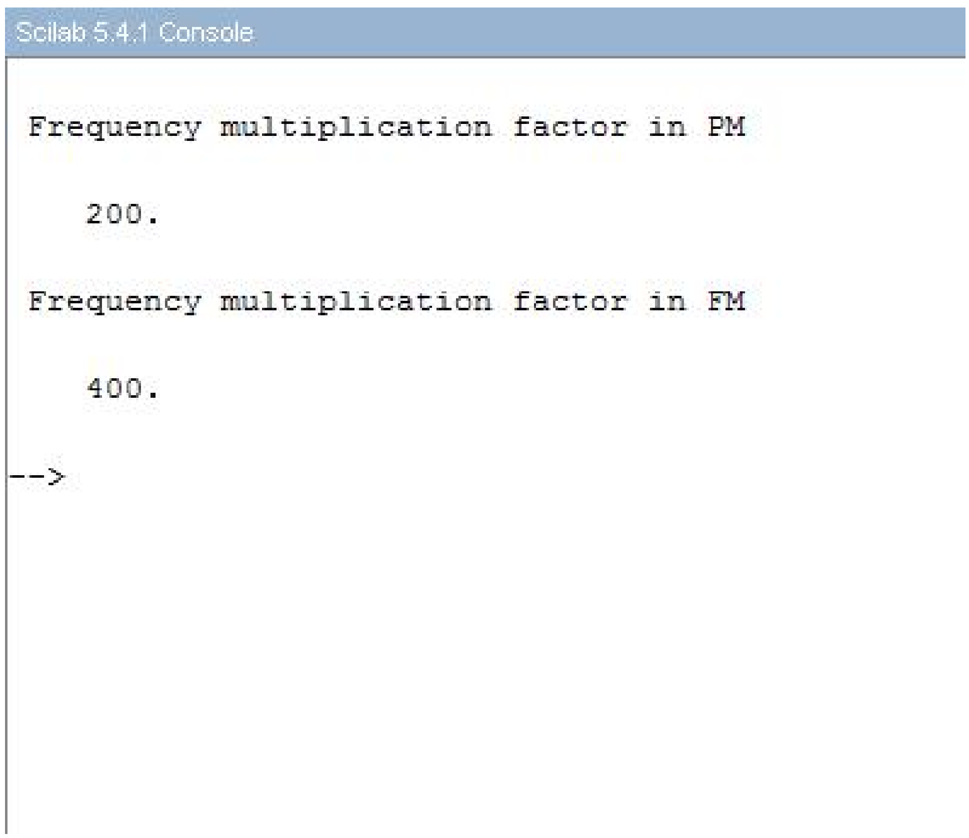


Figure 4.13: Multiplication Factor

```
disp(n1,'Frequency multiplication factor in PM');

//(b)FM
n2=delfd2/delfd1;
disp(n2,'Frequency multiplication factor in FM');
```

Scilab code Exa 4.29 Superheterodyne Receiver

```
//Page Number: 4.25
//Example 4.29
clc;
//Given,
f1=108; //MHz
f2=157; //MHz

//(a) Image frequency overlaps RF band
fIF=12; //MHz

fL01=f1-fIF;
disp('MHz',fL01,'fL01');
fim1=fL01-fIF;
disp('MHz',fim1,'fim1');

fL02=f2-fIF;
disp('MHz',fL02,'fL02');
fim2=fL02-fIF;
disp('MHz',fim2,'fim2');

//Clearly image and RF band overlap
```

Scilab 5.4.1 Console

```
fLO1

    96.

MHz

fim1

    84.

MHz

fLO2

    145.

MHz

fim2

    133.

MHz
```

Chapter 5

Pulse Modulation Systems

Scilab code Exa 5.9 Sampling rate

```
//Page Number: 5.26
//Example 5.9
clc;
//Given,
//m(t)=10cos(2000*pi*t)cos(8000*pi*t)
//or 5 cos (6000*pi*t) +5*cos(10000*pi*t)
//(a) Minimum sampling rate
//we have
fM=5000; //Hz
fs=2*fM;
disp('Hz',fs,'Minimum sampling rate');

//(b)bandpass sampling theoram
fu=fM;
fb=fM-3000; //Hz
//As fu/fb is 2.5
//We have
k=2;
fs2=(2*fu)/k;
```

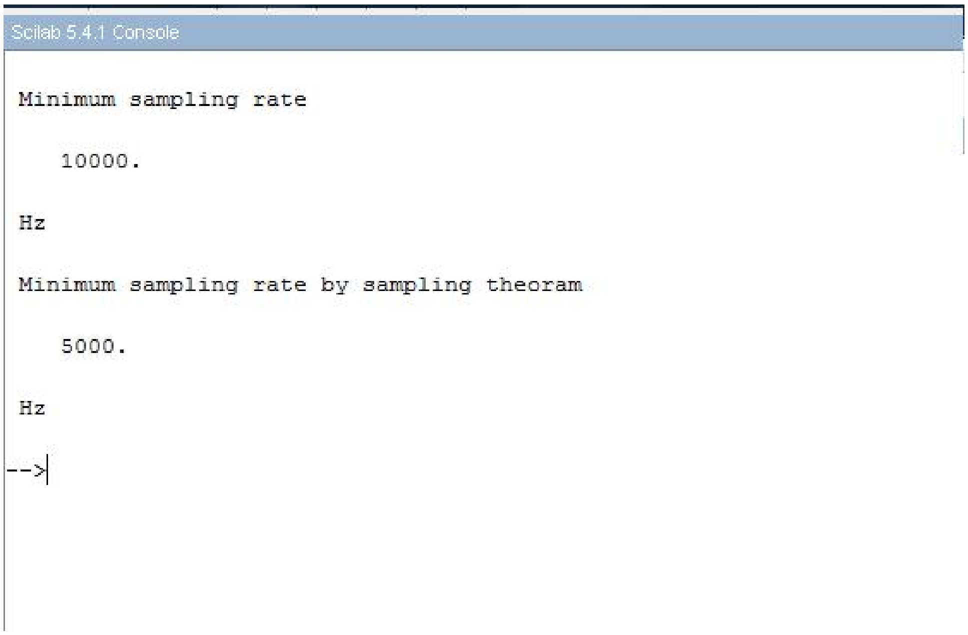


Figure 5.1: Sampling rate

```
disp('Hz',fs2,'Minimum sampling rate by sampling
   theoram');
```

Scilab code Exa 5.14 Binary Channel

```
//Page Number: 5.31
//Example 5.14
clc;
//Given,
Rb=36000; // (b/s)
fM=3200; //Hz
fs=2*fM;
n=Rb/fs;
//As n should be less than Rs/fs
```

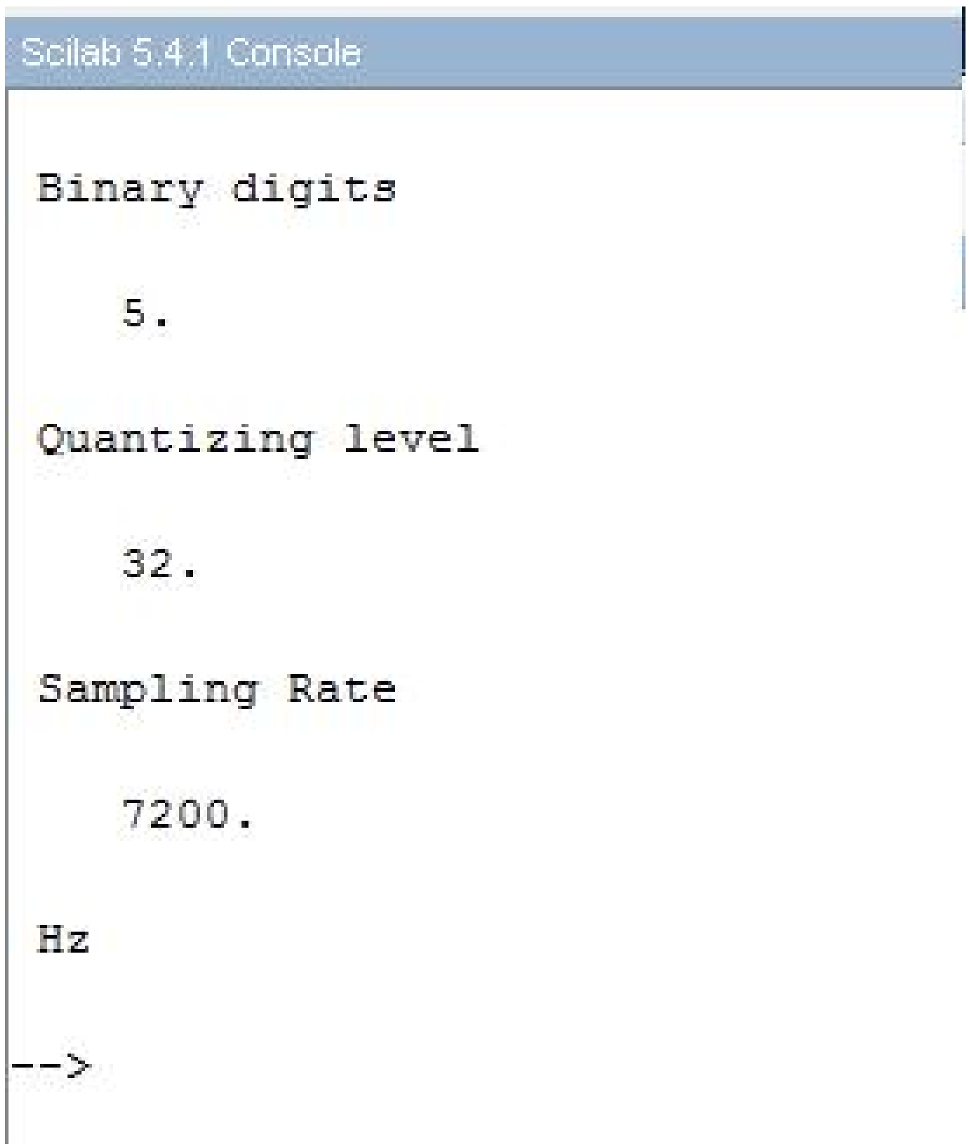


Figure 5.2: Binary Channel

```

nn=round(n)-1;
disp(nn,'Binary digits');

L=2^nn;
disp(L,'Quantizing level');

fs=Rb/nn;
disp('Hz',fs,'Sampling Rate');
```

Scilab code Exa 5.15 Analog Signal to PCM System

```
//Page Number: 5.31
//Example 5.15
clc;

mp=1; //Assume peak amplitude is unity
//Given
del=0.02*mp;

L=(mp*2)/del;

for (i=0:10)
  j=2^i;
  if(j>=L)
    L1=j;
    break;
    end
end
n=log2(L1);// bits per sample
disp(n,'Number of bits');
```

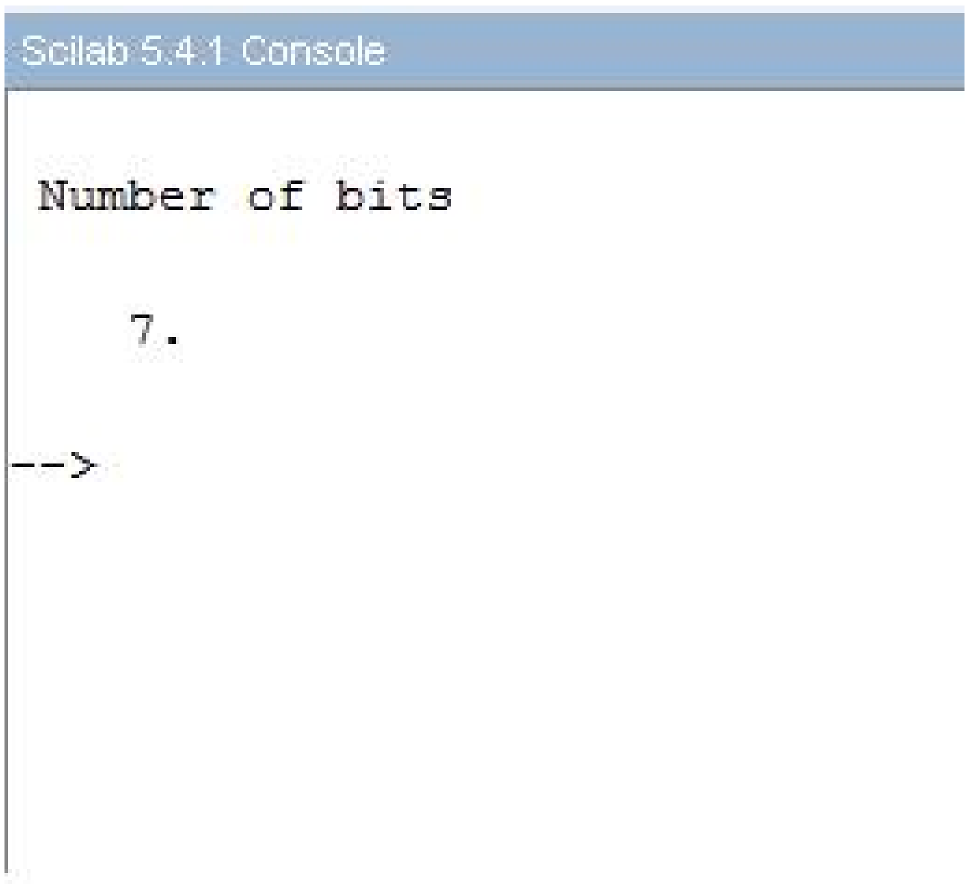


Figure 5.3: Analog Signal to PCM System

Scilab code Exa 5.18 Binary PCM System

```
//Page Number: 5.31
//Example 5.18
clc;
//Given,
SbyN=40; //db
SbyN0=10^(SbyN/10);

//As sbyn=3L^2/2
L=sqrt((2*(SbyN0))/3);
LL=round(L);

n=(log2(LL));

nn=(round(n))+1; //Upper limit

disp(nn,'Binary digits');
LL=2^nn;

disp(LL,'Number of levels');
//As SQN= 1.76+6.02(n)
SQN= 1.76+6.02*(nn);
disp('dB',SQN,'Signal to quantizin ratio');
```

Scilab code Exa 5.19 CD Recording System

```
//Page Number: 5.33
```

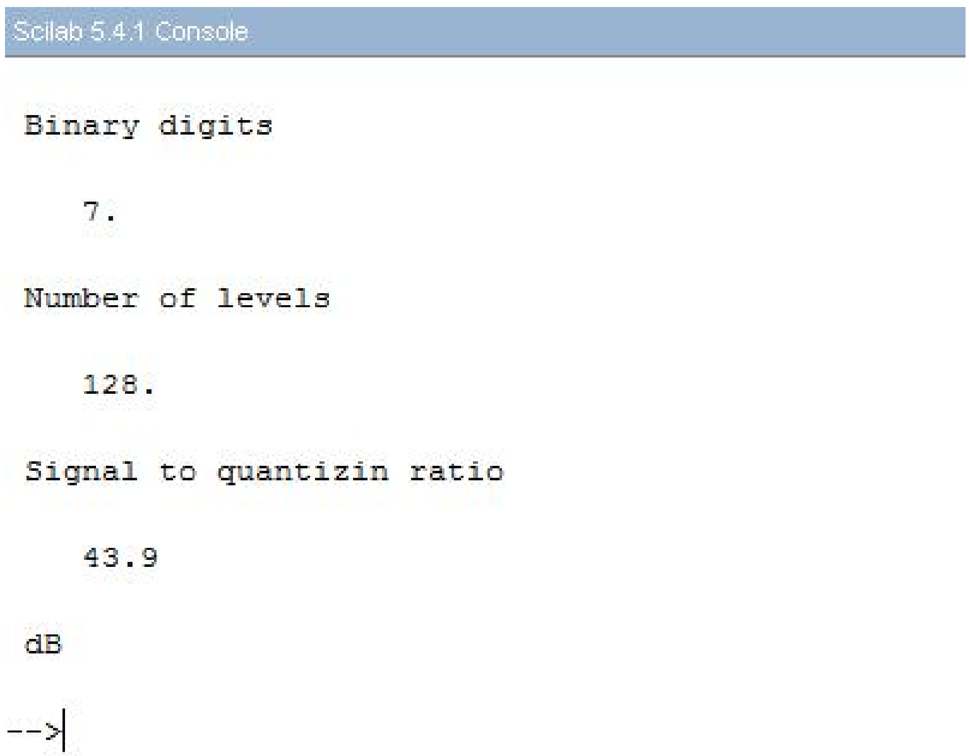


Figure 5.4: Binary PCM System

Scilab 5.4.1 Console

```
Output signal to quantizing ratio:

   98.08

dB

Input bit rate:

   1411200.

B/s

Output bit rate:

   2822400.

B/s

Number of bits recorded:

   1.016D+10

Bytes

Number of bits required

   1.008D+08

Bytes

Number of comparable books

   50.

-->
```

```
//Example 5.19
clc;
//Given,
n=16;
Rb=44100; //b/s
//(a) Output signal to quantizing ratio
SNQ=1.76+(6.02*n);
disp('dB',SNQ,'Output signal to quantizing ratio: ')
    ;

//(b)Input Bit Rate
IBR=2*Rb*n;
disp('B/s',IBR,'Input bit rate: ');
OBR=2*IBR;
disp('B/s',OBR,'Output bit rate: ');

//(c)Number of bits recorded
//Given, an hours time
//Therefore, time
t=60*60;
NBR=OBR*t;
disp('Bytes',NBR,'Number of bits recorded: ');

//(d) Dictionary
//Given
p=1500;
c=2;
l=100;
w=8;
let=6;
b=7;
d=p*c*l*w*let*b;
disp('Bytes',d,'Number of bits required');

x=NBR/(2*d);
y=round(x);
disp(y,'Number of comparable books');
```

Scilab code Exa 5.22 Audio Signal with Spectral Components

```
//Page Number: 5.35
//Example 5.22
clc;
//Given ,
f1=200; //Hz
f2=3300; //Hz
fs=8000; //Samples/s
SQN=30; //dB

//(a)Minimum number of quantizin levels and bits per
    sample
//From SQN=1.76+20log L
La=10^((SQN-1.76)/20);
L=round(La);
disp(L,'Minimum number of quantizing levels');
n=log2(L);
nn=round(n);
disp(nn,'Minimum number of bits per sample');

//(b)Minimum system bandwidth
Fpcm=(nn*fs)/2;
disp('Hz',Fpcm,'Minimum system Bandwidth');

//(c)For u=255
//SQN=20logL -10.1
La1=10^((SQN+10.1)/20);
L1=(round(La1))+1; //Upper Limit
disp(L1,'Minimum number of quantizing levels for u
   =255');
n1=log2(L1);
```

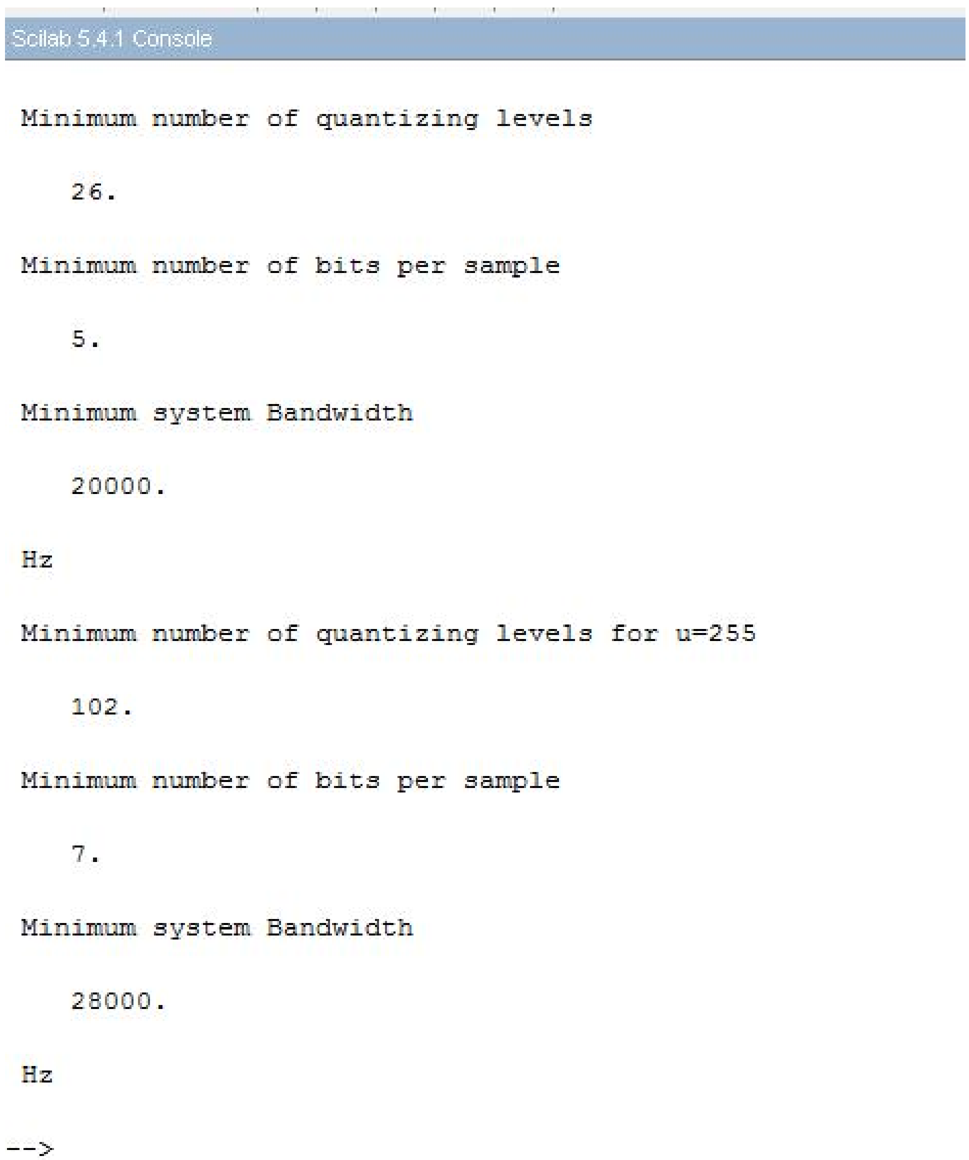


Figure 5.6: Audio Signal with Spectral Components

```
nn1=(round(n1));
disp(nn1,'Minimum number of bits per sample');

//Minimum system bandwidth
Fpcm1=(nn1*fs)/2;
disp('Hz',Fpcm1,'Minimum system Bandwidth');
```

Scilab code Exa 5.25 DM System

```
//Page Number: 5.37
//Example 5.25
clc;
//Given,
fs=32D+3; //Hz
fm=1000; //Hz
fM=4D+3; //Hz
//As SNR=(3*(fs^3))/(8*pi*pi*(fm^2)*fM)
SNR=(3*(fs^3))/(8*%pi*%pi*(fm^2)*fM);
SNRdb=(log10(SNR));
disp('dB',SNRdb,'Output SNR');
```

Scilab code Exa 5.26 PCM Systems

```
//Page Number: 5.38
//Example 5.26
clc;
//Given,
n=4;
SNQ=1.76+(6.02*n);
```

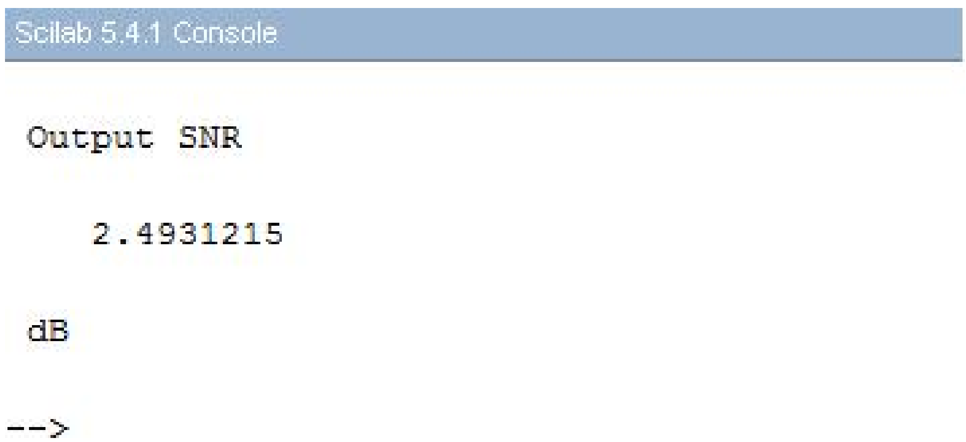


Figure 5.7: DM System

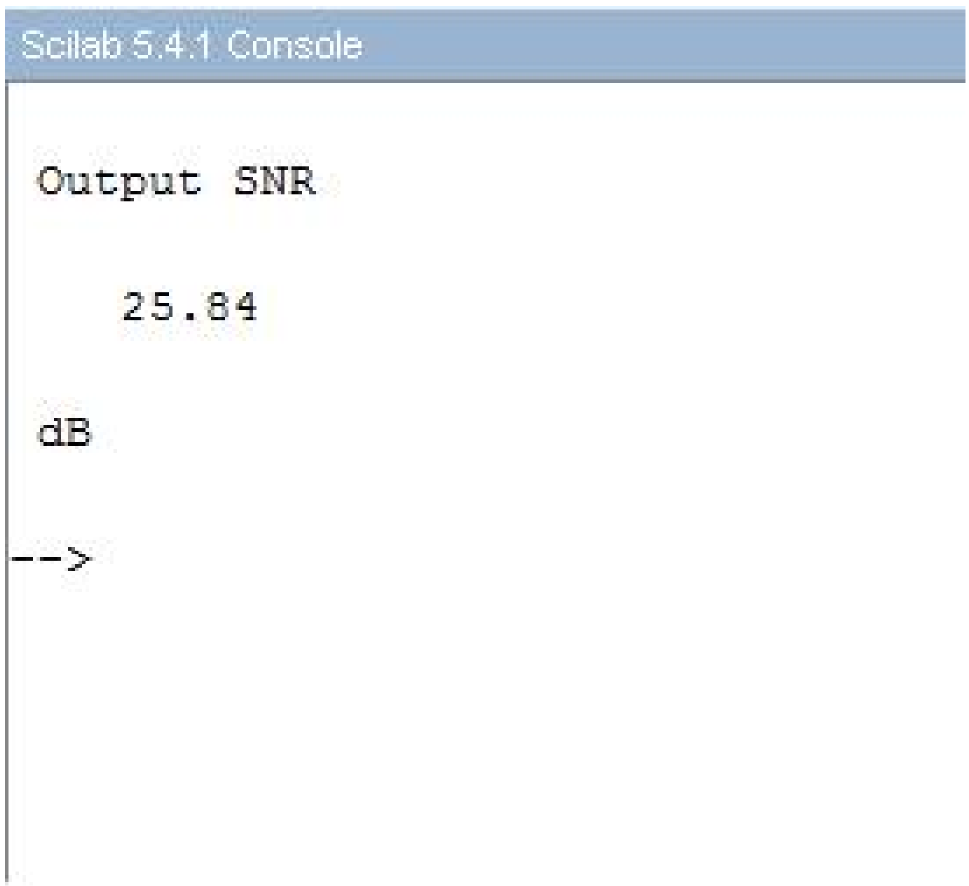


Figure 5.8: PCM Systems

```
disp('dB',SNQ,'Output SNR');
```

Scilab code Exa 5.27 DM system

```
//Page Number: 5.38
//Example 5.27
clc;
//Given,
bw=3D+3; //Hz
n=3;
fs=(n*2*bw);
del=250D-3; //mV
fm=1000; //Hz
//(a) Maximum amplitude
Amax=(del*fs)/(2*%pi*fm);
disp('V',Amax,'Maximum Amplitude');

//(b) Output signal to quantizing ratio
SNRO=(3*(fs^3))/(8*%pi*%pi*(fm^3));
SNRdb=10*(log10(SNRO));
disp('dB',SNRdb,'Output SNR');
```

Scilab code Exa 5.32 TDM

```
//Page Number: 5.40
//Example 5.32
clc;
//Given,
m1=3D+3; //Hz
```

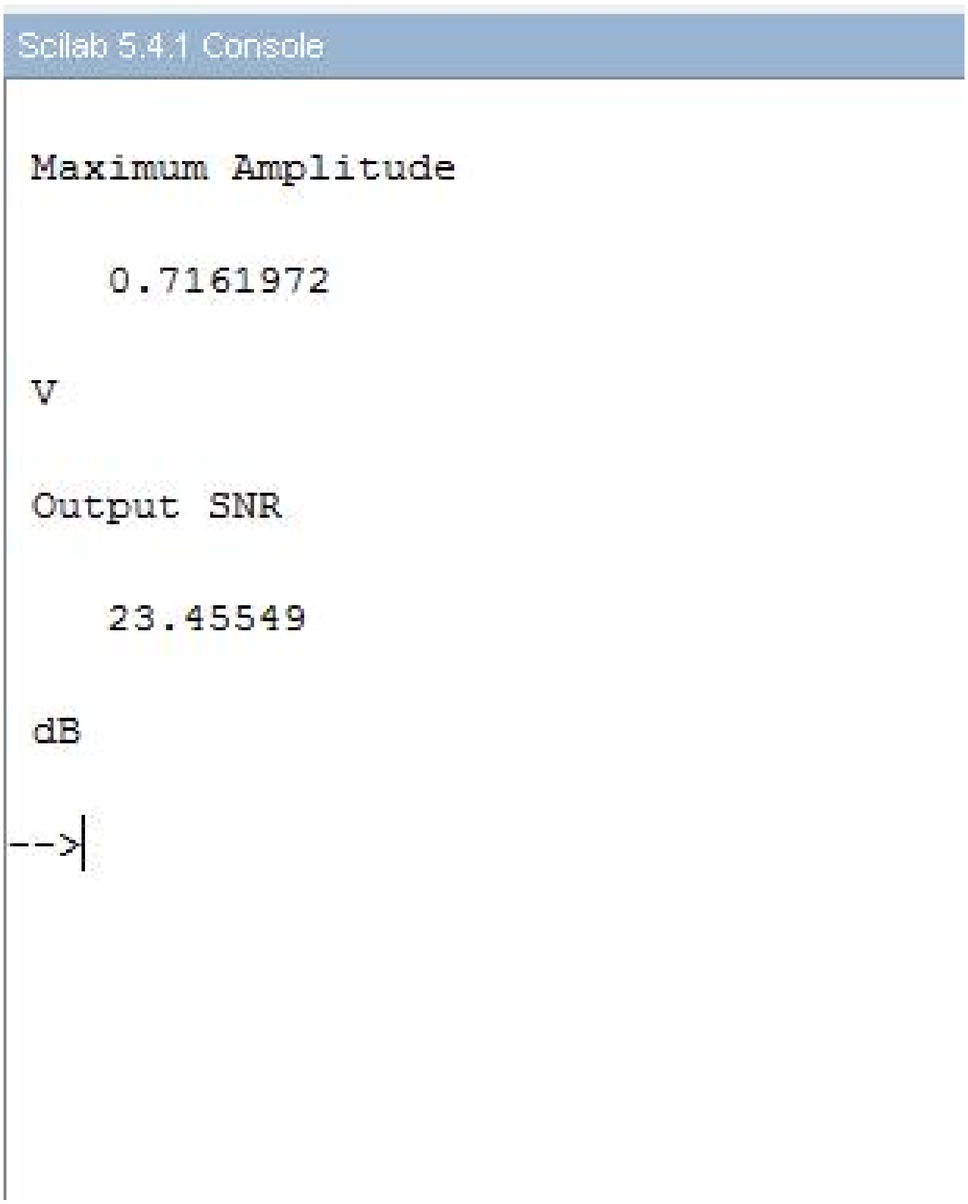


Figure 5.9: DM system

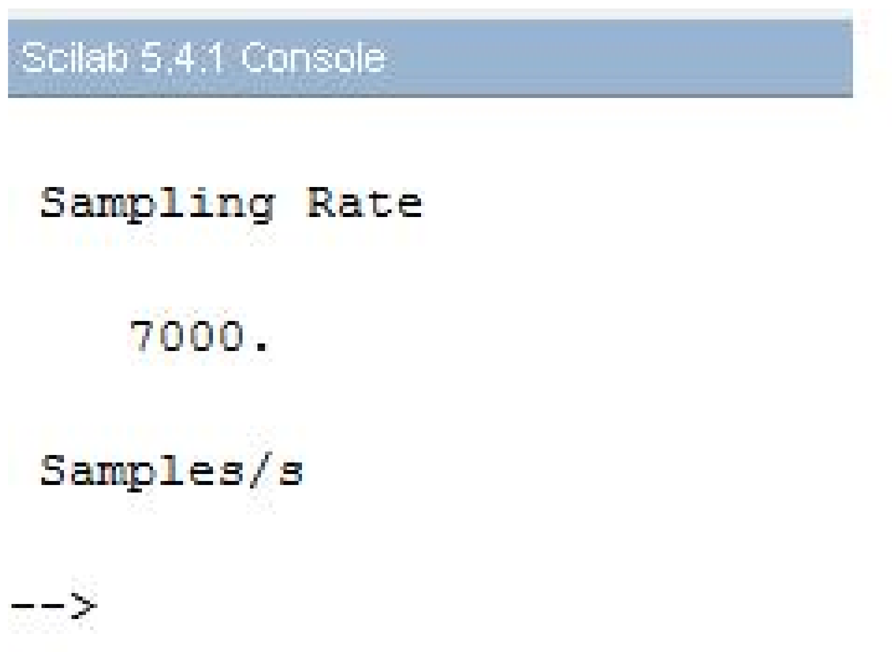


Figure 5.10: TDM

```
m2=3.5D+3; //Hz
//Since highest frequency is of m2
Sr=2*m2;
disp('Samples/s',Sr,'Sampling Rate');
```

Scilab code Exa 5.33 TD Multiplexing

```
//Page Number: 5.40
//Example 5.33
clc;
//Given,
m1=3.6D+3; //Hz
m2=1.2D+3; //Hz
m3=m2;
m4=m2;
//(a)Nyquist rate
nr1=2*m1;
disp(nr1,'Nyquist Rate of m1(t)');
nr2=2*m2;
disp(nr2,'Nyquist Rate of m2(t)');
nr3=2*m3;
disp(nr3,'Nyquist Rate of m3(t)');
nr4=2*m4;
disp(nr4,'Nyquist Rate of m4(t)');

//(b) Speed of commutator
c=nr1+nr2+nr3+nr4;
disp('samples/s',c,'Speed of commutator');

//(c)Output bit rate
//Given,
L=1024;
n=log2(L);
```

Scilab 5.4.1 Console

```
Nyquist Rate of m1(t)

   7200.

Nyquist Rate of m2(t)

   2400.

Nyquist Rate of m3(t)

   2400.

Nyquist Rate of m4(t)

   2400.

Speed of commutator

   14400.

samples/s

Output bit rate

   144000.

b/s

Minimum Channel Bandwidth

   7200.

Hz
```

```
OBR=n*c;
disp('b/s',OBR,'Output bit rate');

//(d)Minimum channel bandwidth
fB=c/2;
disp('Hz',fB,'Minimum Channel Bandwidth');
```

Scilab code Exa 5.34 T1 Carrier System

```
//Page Number: 5.41
//Example 5.34
clc;
//Given,
fs=8000; //Hz
m=24;
n=8;
//(a) Duration of each bit
t1=1/fs;
t2=(m*n)+1; // Extra bit for synchronization
Tb=t1/t2;
disp('seconds',Tb,'Duration of each bit');

//(b) Transmission Rate
Rb=1/Tb;
disp('b/s',Rb,'Transmission Rate');

//(c)Minimum transmission bandwidth
fT1=1/(2*Tb);
disp('Hz',fT1,'Minimum transmission bandwidth');
```

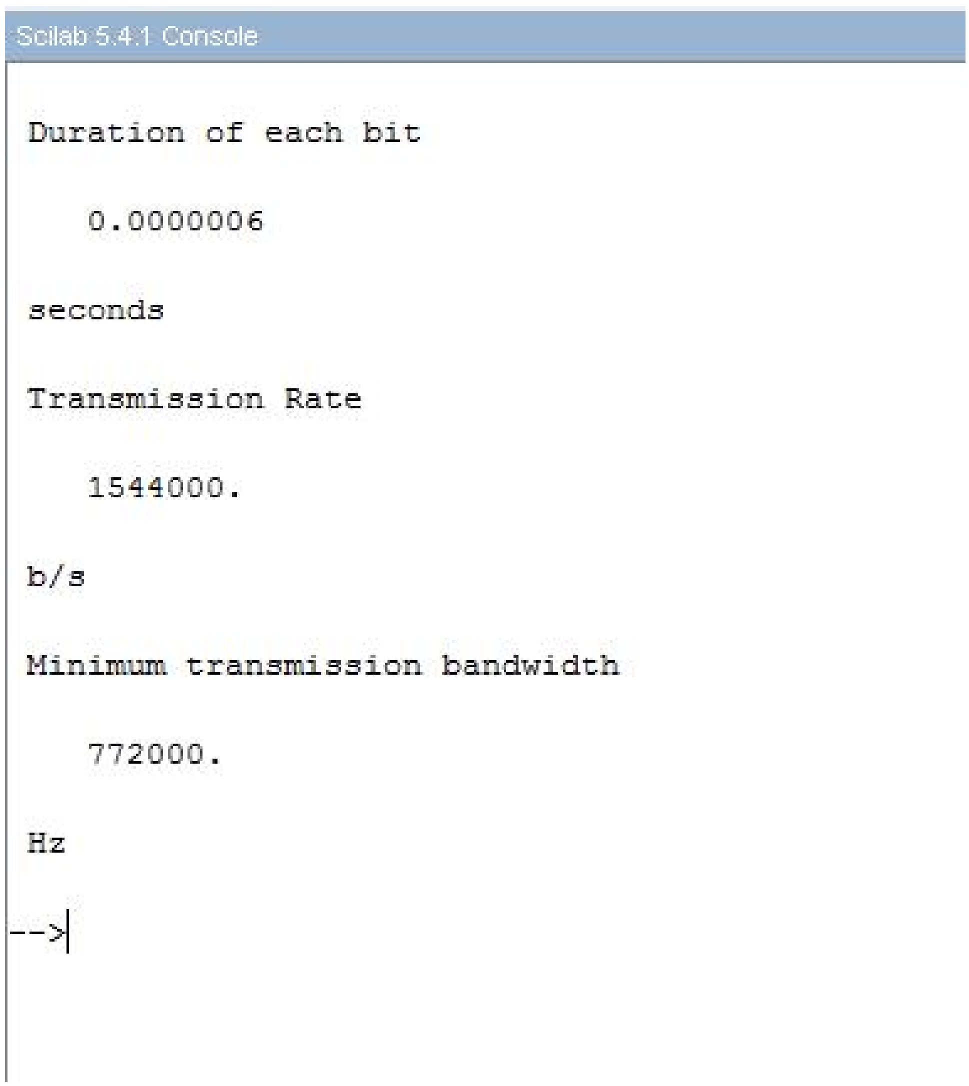


Figure 5.12: T1 Carrier System

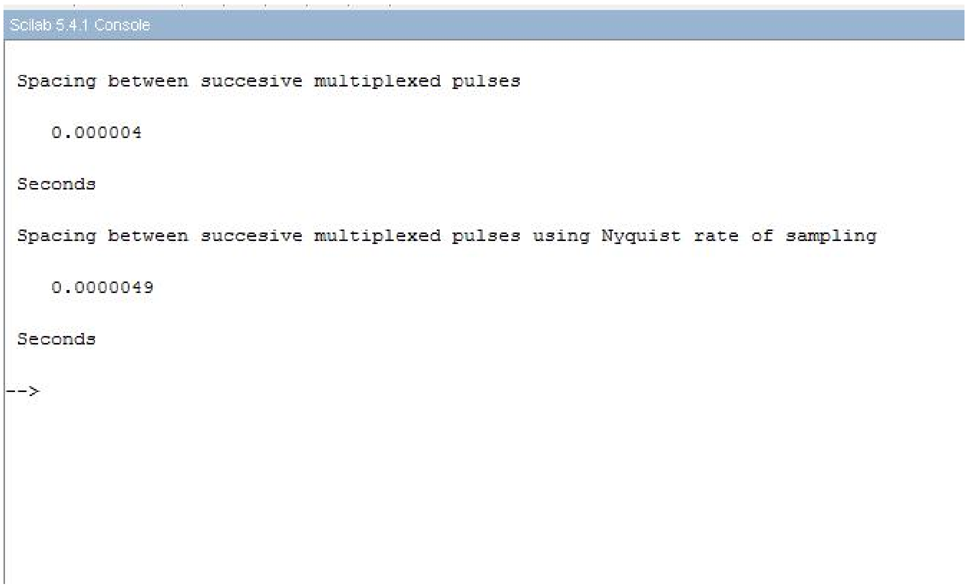


Figure 5.13: TDM

Scilab code Exa 5.35 TDM

```
//Page Number: 5.42
//Example 5.35
clc;
//Given ,
n=24;
f=3.4D+3; //Hz
ts=1D-6; //Second

//(a) Spacing between succesive multiplexed pulses
fs=8000; //Samples per second
t1=1/fs;
t2=n+1; //One synchronizing bit

Tb=t1/t2;
```

```
//Actual Tb, as actual duration of each pulse is 1us
ATb=Tb-ts;
disp('Seconds',ATb,'Spacing between succesive
    multiplexed pulses');


//(b) Nyquist Rate of Sampling
f1=2*f;
T=1/f1; //Seconds

Tb1=T/t2;

ATb1=Tb1-ts;
disp('Seconds',ATb1,'Spacing between succesive
    multiplexed pulses using Nyquist rate of sampling
    ');
```

Scilab code Exa 5.37 Telephone Line

```
//Page Number: 5.43
//Example 5.37
clc;
//Given,
bw=3.5D+3; //Hz
//Roll off factor
a=0.25;
Rb=(2*bw)/(1+a);
disp('b/s',Rb,'Data Rate');
```

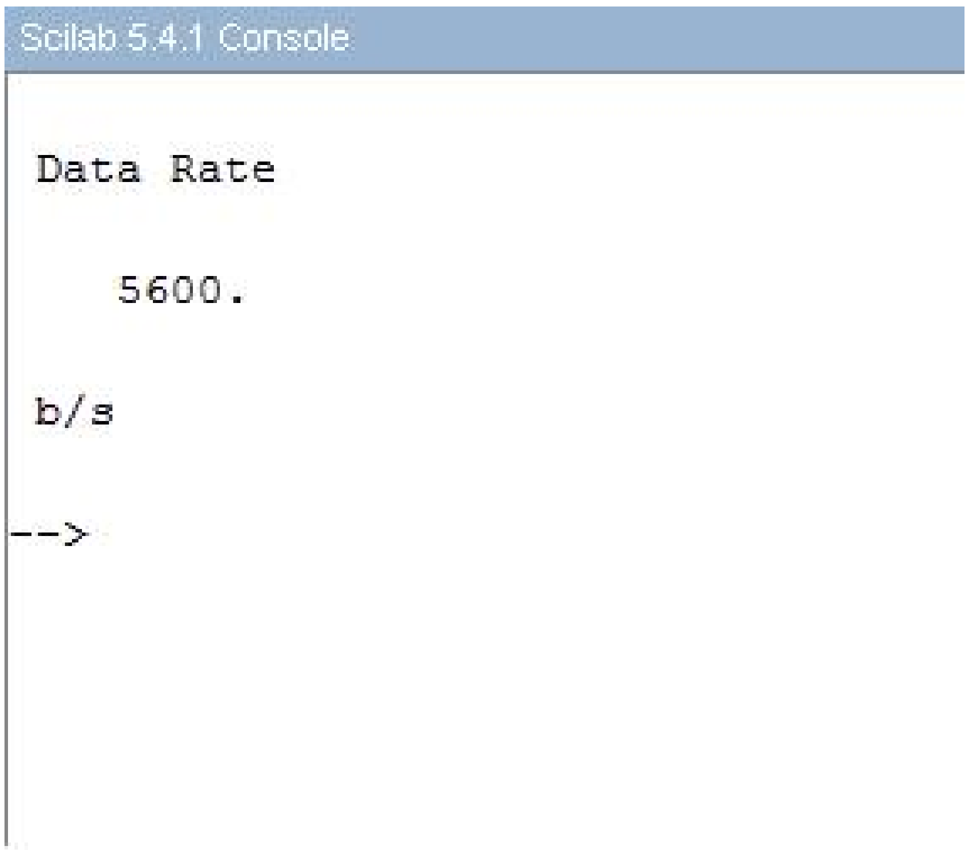


Figure 5.14: Telephone Line

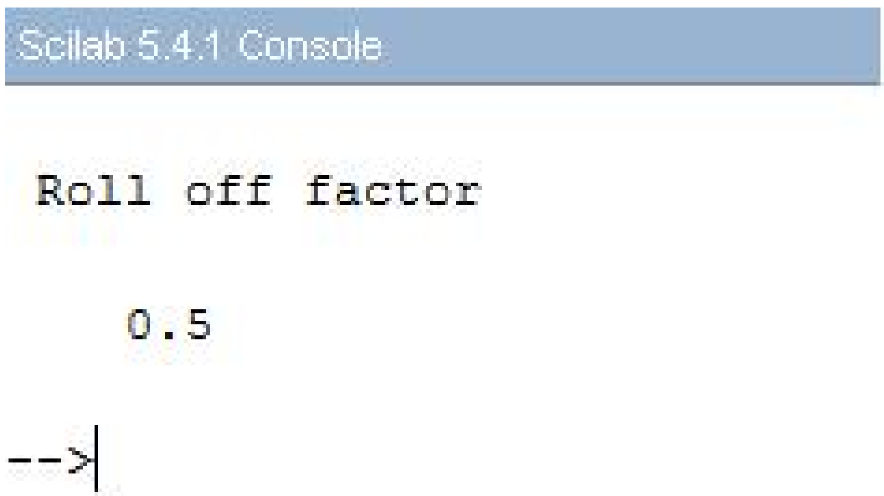


Figure 5.15: Roll Off Factor

Figure 5.16: Telemetry System

Scilab code Exa 5.38 Roll Off Factor

```
//Page Number: 5.43
//Example 5.38
clc;
//Given,
fB=75D+3; //Hz
Rb=0.1D+6; //B/s
Tb=1/Rb;
a=(2*fB*Tb)-1;
disp(a,'Roll off factor');
```

Scilab code Exa 5.39 Telemetry System

```
//Page Number: 5.43
//Example 5.39
clc;
//Given,
m=8;
fM=2D+3; //Hz
a=0.2;
//Here we choose L=128;
L=128;
n=log2(L);
Sr=2*fM;
fs=1.25*Sr;

//For n tdm signals
x=m*fs;

//Resultant bit rate
br=7*x;

//Minimum Transmission bandwidth
fB=((1+a)*br)/2;
disp('Hz', fB,'Minimum Transmission bandwidth');
```

Scilab code Exa 5.40 M ary PSK

```
//Page Number: 5.44
//Example 5.40
clc;
//Given,
M=16;
sr=40000;
```

```
Scilab 5.4.1 Console

 Bit Rate:

    160000.

 b/s

 Value of M

    4.

 Band Rate

    80000.

 Symbols/s

 Spectral efficiency

    1.5384615

 b/s Hz

-->
```


Figure 5.17: M ary PSK

```
a=0.3;
//(a) Bit Rate
n=log2(M);
br=sr*n;
disp('b/s',br,'Bit Rate: ');

//(b)
//As 2*fB=(1+a)*R/log2M
//Given
bw=110D+3; //Hz  //=2*fB

M=2^(((1+a)*br)/bw);
MM=round(M);
disp(MM,'Value of M');

//(c)Band Rate
band=br/(log2(n));
disp('Symbols/s',band,'Band Rate');

//(d) Spectral efficiency
BT=((1+a)*br)/2;
Eff=br/BT;
disp('b/s Hz',Eff,'Spectral efficiency');
```

Chapter 6

Probability and Random Variables

Scilab code Exa 6.7 Dot and dash

```
//Page Number: 6.16
//Example 6.7
clc;
//Given
//Pdot=2*Pdash and Pdot+Pdash=1
//Therfore, on solving using linear equations
a=[1 -2;1 1];
c=[0;1];
b=inv(a)*c;
Pdash=b(1,1);
Pdot=b(2,1);
disp(Pdot,'Pdot:');
disp(Pdash,'Pdash:')
```

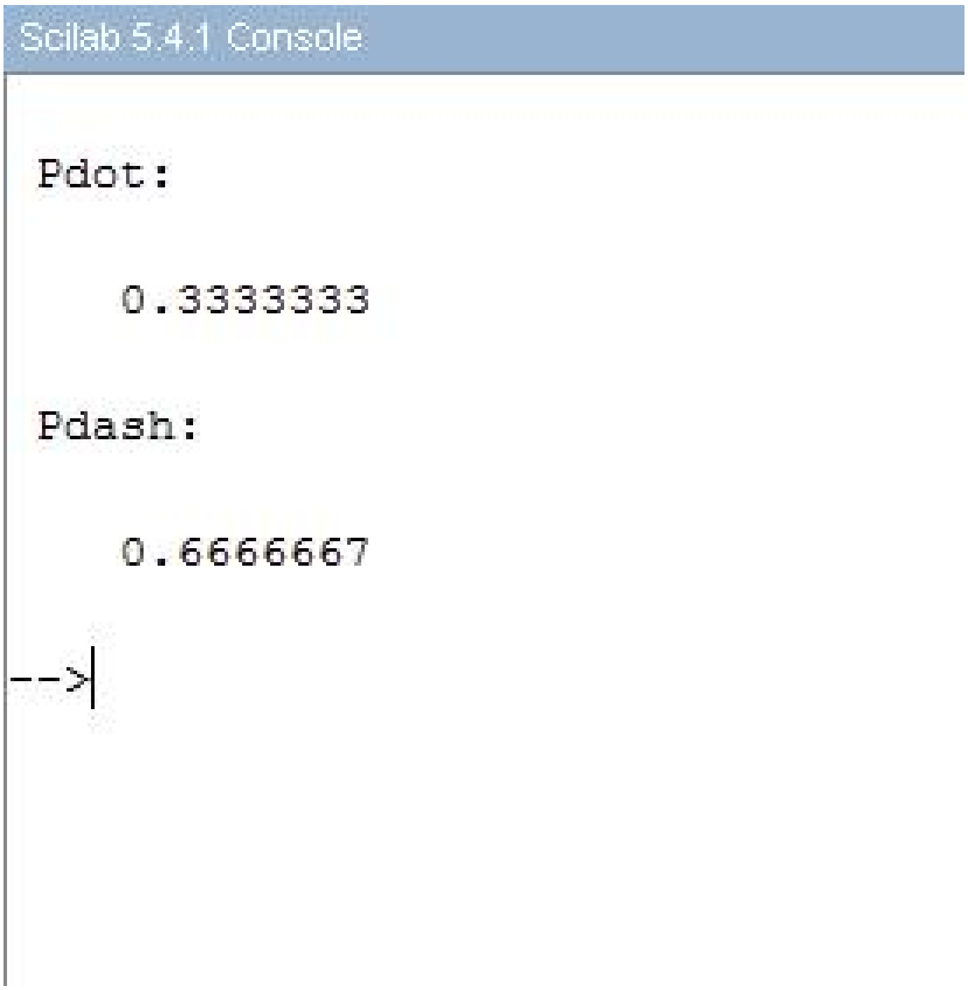


Figure 6.1: Dot and dash

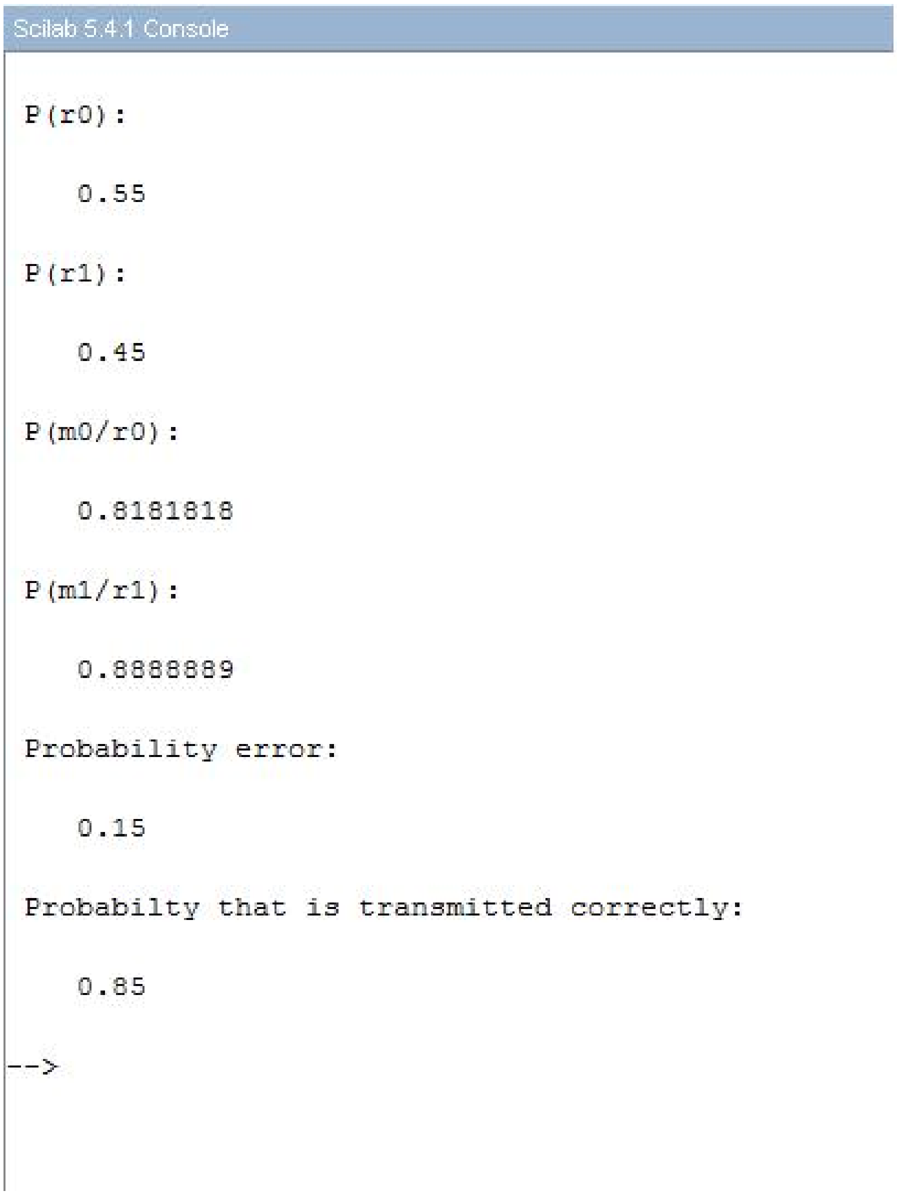


Figure 6.2: Binary Communication

Scilab code Exa 6.14 Binary Communication

```
//Page Number: 6.18
//Example 6.14
clc;
//Given
p=0.1;
q=0.2;
Pm0=0.5;
Pr1bym0=p;
Pr0bym1=q;

//(a) Find Pr0 and Pr1

Pm1=1-Pm0;
Pr0bym0=1-Pr1bym0;
Pr1bym1=1-Pr0bym1;

//By formula
//P(r0)=(P(r0/m0)*P(m0))+(P(r0/m1)*P(m1);
//P(r1)=(P(r1/m0)*P(m0))+(P(r1/m1)*P(m1);

Pr0=(Pr0bym0*Pm0)+(Pr0bym1*Pm1);
Pr1=(Pr1bym0*Pm0)+(Pr1bym1*Pm1);
disp(Pr0,'P(r0):');
disp(Pr1,'P(r1):');

//(b)P(m0/r0)
//Using Bayes Rule
//P(m0/r0)=(P(m0)*P(r0/m0)/P(r0))
Pm0byr0=(Pm0*Pr0bym0)/Pr0;
disp(Pm0byr0,'P(m0/r0):');

//(c)P(m1/r1)
```

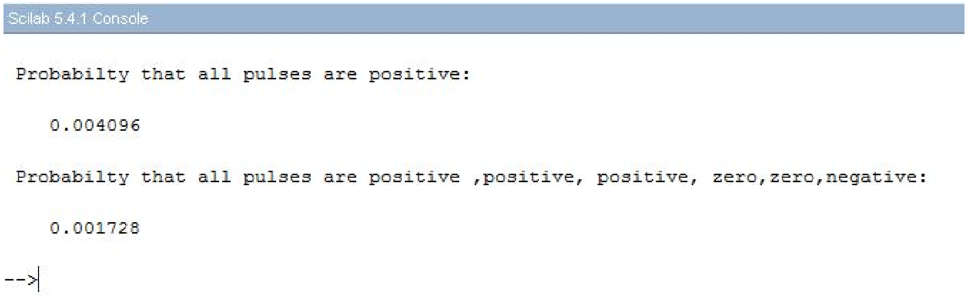


Figure 6.3: Pulse Positions

```
//Using Bayes Rule
//P(m1/r1)=(P(m1)*P(r1/m1)/P(r1))
Pm1byr1=(Pm1*Pr1bym1)/Pr1;
disp(Pm1byr1,'P(m1/r1):');

//(d)Probabilty error
//As Pe=(P(r1/m0)*P(m0))+(P(r0/m1)*P(m1))
Pe=(Pr1bym0*Pm0)+(Pr0bym1*Pm1);
disp(Pe,'Probability error:');

//(e)Probabilty that is transmitted correctly
//As Pc=(P(r0/m0)*P(m0))+(Pr1bym1*Pm1)

Pc=(Pr0bym0*Pm0)+(Pr1bym1*Pm1);
disp(Pc,'Probabilty that is transmitted correctly:')
    ;
```

Scilab code Exa 6.16 Pulse Positions

```
//Page Number: 6.21
```

```
//Example 6.16
clc;
//Given
p=0.4;
Pp=p;
q=0.3;
Pn=q;
a=1; //i start value
b=6; //i end value
//(a)Probabilty that all pulses are positive
s=1;
for i=a:b
    s=s*Pp;
    end
disp(s,'Probabilty that all pulses are positive:');

//(b)Pulses are positive ,positive, positive, zero,
   zero,negative
Pz=1-(p+q);
s1=(Pp^3)*(Pz^2)*Pn;
disp(s1,'Probabilty that all pulses are positive ,
   positive, positive, zero,zero,negative:');
```

Scilab code Exa 6.17 Binary Source

```
//Page Number: 6.21
//Example 6.17
clc;
//Given
P1=0.6;
P0=0.4;
n=5; //Five digit sequence
j=2; //two outcomes 0 and 1
```

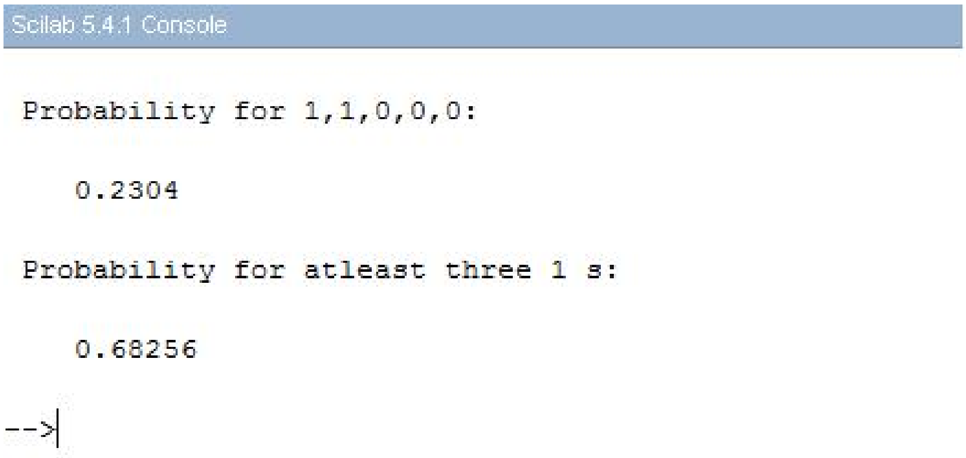


Figure 6.4: Binary Source

```

//(a)1,1,0,0,0
xf=(factorial(n))/((factorial(j))*(factorial(n-j)));
s=xf*(P1^j)*(P0^(n-j));
disp(s,'Probability for 1,1,0,0,0:');

//(b)atleast 3 1's
//P(X>=3)=1-P(X<=2)
//Here y=1-x
x=0;
for k=0:2
    f=(factorial(n))/((factorial(k))*(factorial(n-k)));
    x=x+{f*((P1^k)*(P0^(n-k)))};
end
y=1-x;
disp(y,'Probability for atleast three 1 s:');
```

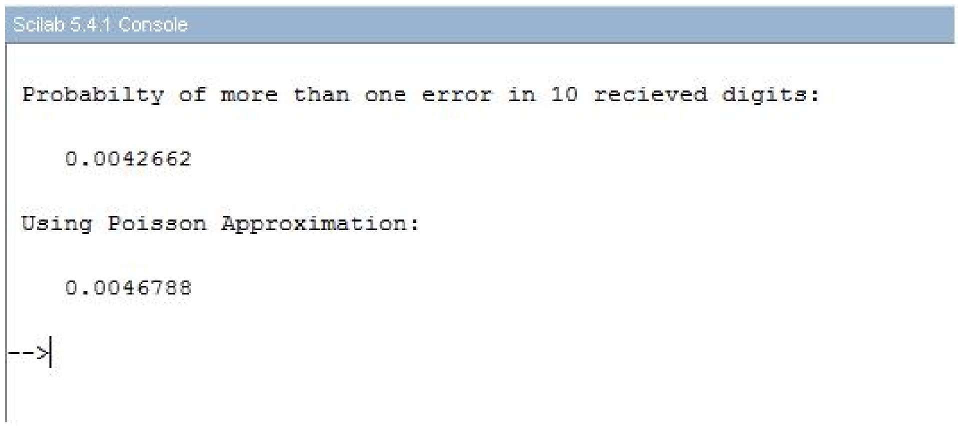


Figure 6.5: Noisy Transmission Channel

Scilab code Exa 6.20 Noisy Transmission Channel

```
//Page Number: 6.23
//Example 6.20
clc;
//Given
pe=0.01; //Error probability

//(a) Probabilty of more than one error in 10 recieved digits
n=10;
//As P(X>1)=1-P(X=0)-P(X=1)
//Let x=P(X>1)
//s=P(X=0)+P(X=1)
s=0;
for t=0:1
    f=(factorial(n))/((factorial(t))*(factorial(n-t)));
```

```
    s=s+{f*(pe^t)*((1-pe)^(n-t))};
end
x=1-s;
disp(x,'Probabilty of more than one error in 10
   recieved digits:');

//(b)Using Poisson approximation
//P(X=k)~[{(%exp)^(-n*p)}*{((n*p)^k)}]/k factorial
s1=0;
for k=0:1
    j=factorial(k);
    s1=s1+[{exp(-n*pe)}*{((n*pe)^k)}]/j;
end
x1=1-s1;
disp(x1,'Using Poisson Approximation:');
```

Scilab code Exa 6.23 Pdf of random variable

```
//Page Number: 6.23
//Example 6.23
clc;
//We find , k=1/(b-a)
//(b) if a=1 and b=2,P(|x|<c) where c=1/2
a=-1;
b=2;
c=1/2;

k=1/(b-a);
//P(|x|<c) = P(-c<=x<=c)
//Let y
x0=-c;x1=c;
y=integrate('1','x',x0,x1);
y1=k*y;
```

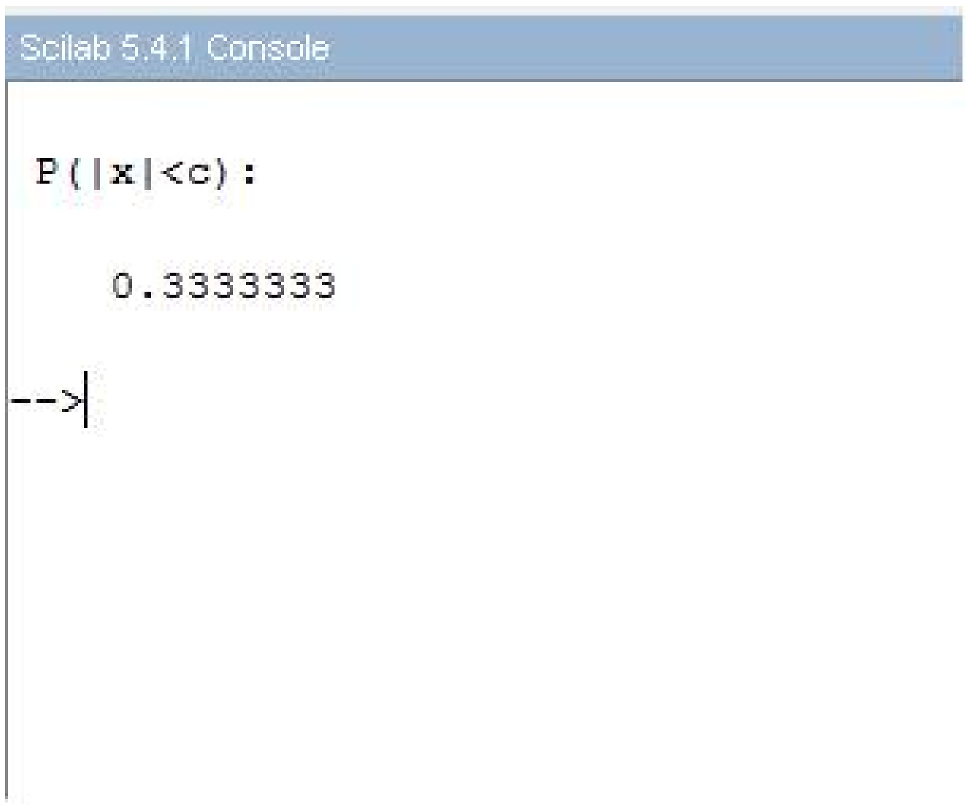


Figure 6.6: Pdf of random variable

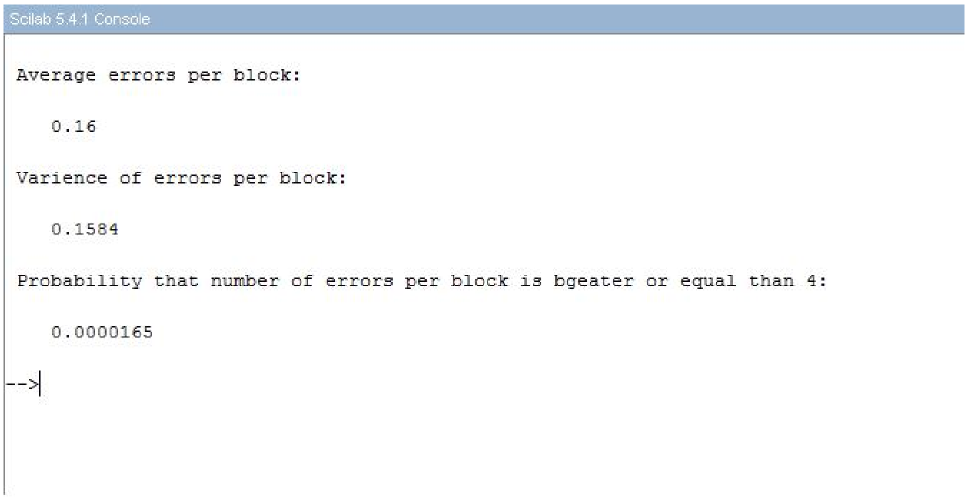


Figure 6.7: 16 Binary Digits Transmission

```
disp(y1,'P(|x|<c):');
```

Scilab code Exa 6.41 16 Binary Digits Transmission

```
//Page Number: 6.34
//Example 6.41
clc;
//Given
n=16; //binary digits
p=0.01; //Probabilty error due to noise

//(a) Average errors per block
//E(X)=n*p
EofX=n*p;
disp(EofX,'Average errors per block:');
```

```
//(b)Varience of errors of block
//s=n*p*(1-p)
s=n*p*(1-p);
disp(s,'Varience of errors per block:');


//(c) Probability that number of errors per block is
    bgeater or equal than 4
i=4;
//AsP(X>=4)=1=P(X<=3)
P3=0;
for k=0:3;
    f=(factorial(n))/((factorial(k))*(factorial(n-k)
       ));
    P3=P3+(f*(p^k)*((1-p)^(n-k)));
end
P4=1-P3;
disp(P4,'Probability that number of errors per block
    is bgeater or equal than 4:');
```

Chapter 8

Noise

Scilab code Exa 8.1 Units Conversion

```
//Page Number: 8.6
//Example 8.1

clc;
disp('Part a');
//(a)Given, u1=1W and u2=1mW
//Change to dBW and dBm
u1=1D-3;
u2=1;
//(i)470mW
a=470D-3;
y1=(10*log10(a/u1));
disp('dBm',y1);

y2=(10*log10(a/u2));
disp('dBW',y2);

//(ii)1W
b=1;
```

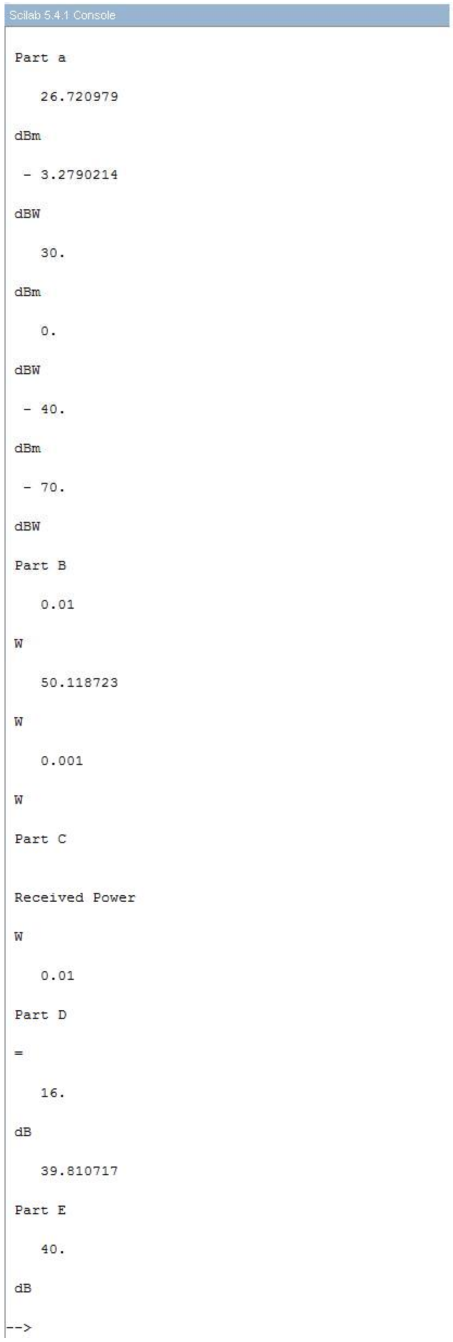


Figure 8.1: Units Conversion

```
z1=(10*log10(b/u1));
disp('dBm',z1);

z2=(10*log10(b/u2));
disp('dBW',z2);

//(iii)100nW
c=100D-9;
x1=(10*log10(c/u1));
disp('dBm',x1);

x2=(10*log10(c/u2));
disp('dBW',x2);

disp('Part B');
//(b)Here u1=1W (for dBW)and u2=1mW (for dBm)
//Change to powers to watts
//(i)-20dBW
a=-20;
k1=u2*(10^(a/10));
disp('W',k1);


//(ii)47dBm
b=47;
k2=u1*(10^(b/10));
disp('W',k2);


//(ii)0dBm
c=0;
k3=u1*(10^(c/10));
disp('W',k3);


disp('Part C');
//(c)Given, channel loss=20dB and Pt=1W
l=-20;
```

```
PT=1;
PR=10^(1/10);
disp(PR,'W','Received Power',);

disp('Part D');
//(d)Given, channel loss=30dB when signal=3dB and
    overall loss=20dB
l1=-30;
s=-3;
l2=-20;
q=-l1-s-s+l2;
d1=10^(q/10);
disp('dB',q,'=');
disp(d1);

disp('Part E');
//(e)Given,
Si=0; //dBm
S1=1D-3*(10^(Si/10));
Ni=1D-7; //W

Osnr=S1/Ni;
Odb=(10*(log10(Osnr)));
disp('dB',Odb);
```

Scilab code Exa 8.2 Rms Voltage

```
//Page Number: 8.7
//Example 8.2
clc;
//Given,
R=1000;
T=27; //degree celsius
```

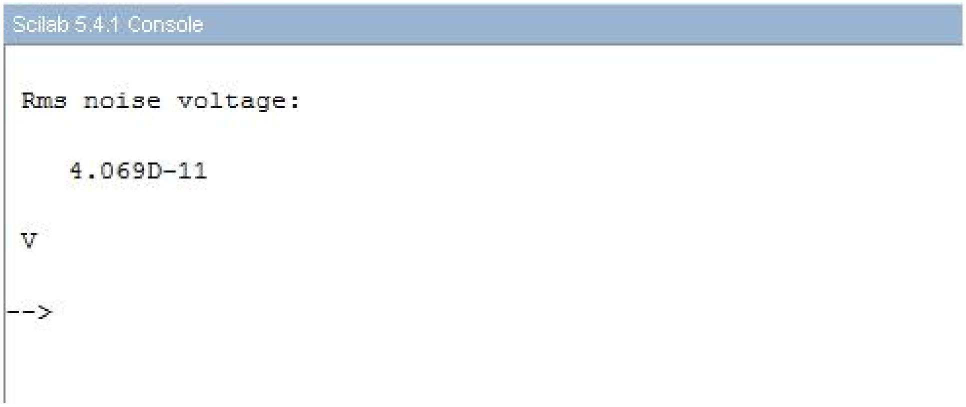


Figure 8.2: Rms Voltage

```
TK=T+273; //kelvin

// We know, rms noise voltage is 4RKTB
K=1.38D-28;
B=10;

V=sqrt(4*R*K*TK*B);
disp('V',V,'Rms noise voltage:');
```

Scilab code Exa 8.3 Output Noise Power

```
//Page Number: 8.8
//Example 8.3
clc;

//Given,
G=100;
G1=(10^(G/10));

T=30;
```

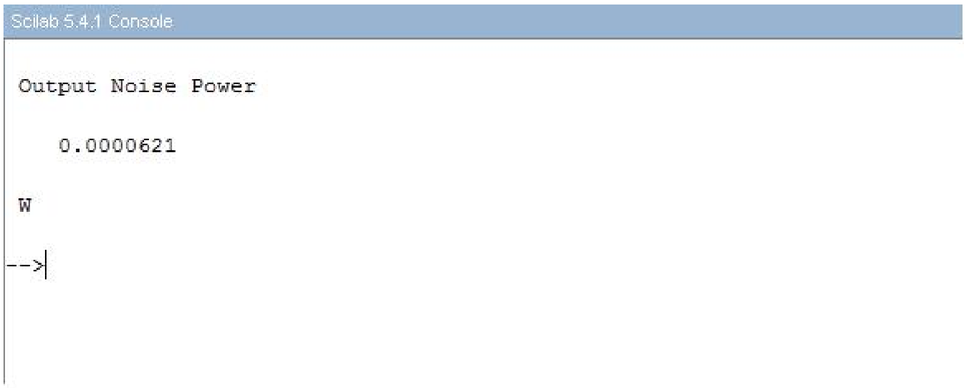


Figure 8.3: Output Noise Power

```
Te=270;

//We know, output noise power=GKB(T+Te)
K=1.38D-23;
B=1.5D+6;

No=G1*1.38D-23*1.5D+6*(T+Te);
disp('W',No,'Output Noise Power');
```

Scilab code Exa 8.4 Noise Figure

```
//Page Number: 8.8
//Example 8.4
clc;

//Given,
R=50;
osnr=0;
```

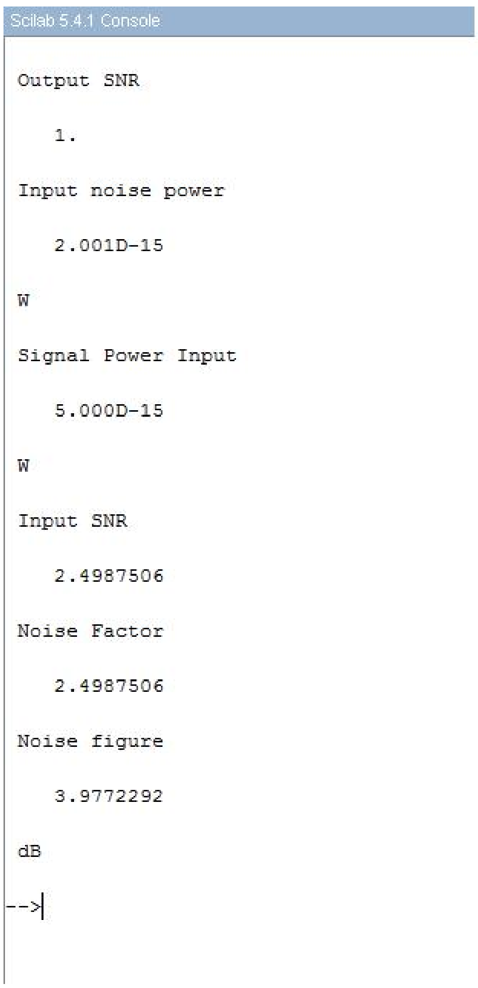


Figure 8.4: Noise Figure

```
SNRo=(10^(osnr/10));
disp(SNRo,'Output SNR');

//As Pni=KTB
K=1.38D-23;
T=290;
B=5D+5;
Pni=K*T*B;
disp('W',Pni,'Input noise power');

//Psi=V^2/R
//Given V=5*10^-6V
V=0.5D-6;
Psi=(V^2)/R;
disp('W',Psi,'Signal Power Input');

isnr=(Psi/Pni);
disp(isnr,'Input SNR');

F=(isnr/SNRo);
disp(F,'Noise Factor');

NF=10*log10(F);
disp('dB',NF,'Noise figure');
```

Scilab code Exa 8.6 Cascaded Amplifiers

```
//Page Number: 8.9
//Example 8.6
clc;

//Given,Stage 1
SNRo=120;
```

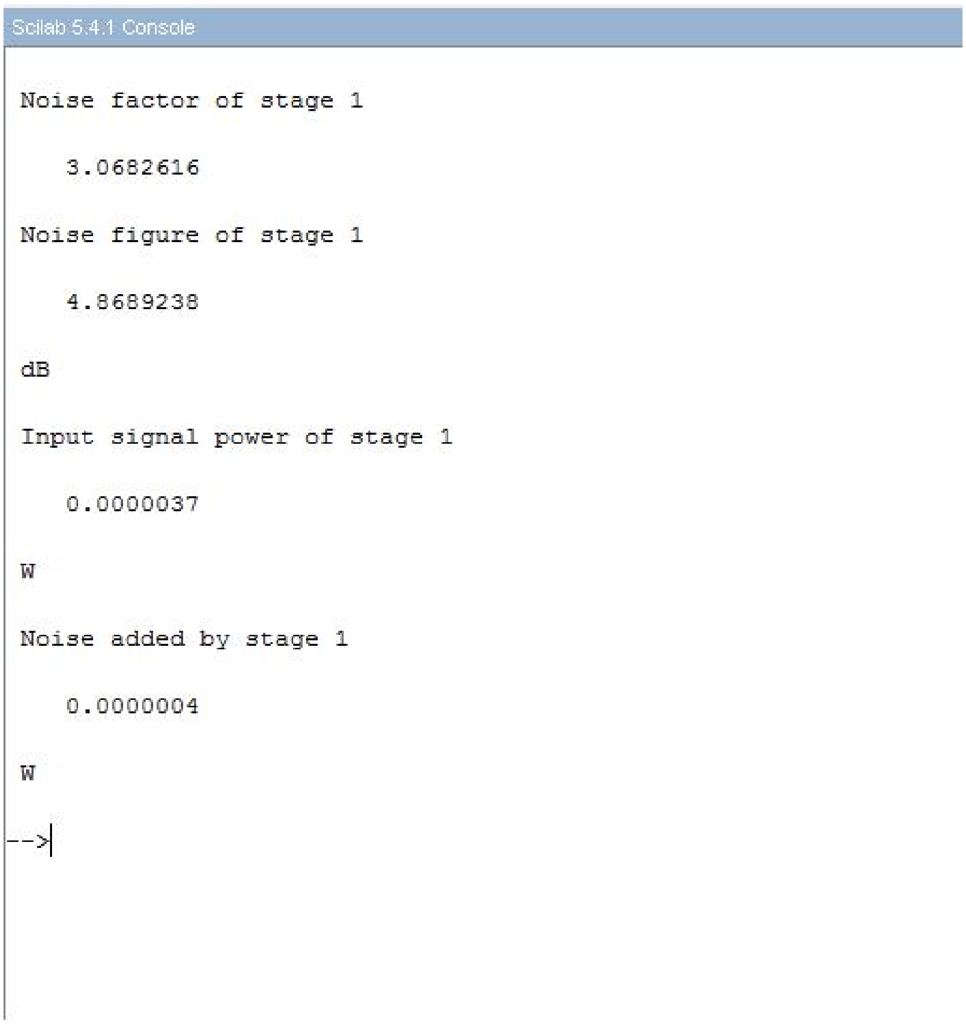


Figure 8.5: Cascaded Amplifiers

```
Pni=0.01D-6; //W
G1=20;

//Stage 2
F2=12; //dB
FF2=(10^(F2/10));
G2=30;

//Stage 3
F3=9.3; //dB
FF3=(10^(F3/10));
G3=35;

//(a)Nosie factor and noise figure of Stage 1
F=5.6; //dB
FF=(10^(F/10));

//As F=F1-((F2-1)/G1)-((F3-1)*(G1G2));
Fa=FF-((F2-1)/G1)-((FF3-1)/(G1*G2));
disp(Fa,'Noise factor of stage 1');

FadB=(10*(log10(Fa))); //dB
disp('dB',FadB,'Noise figure of stage 1');


//(b)Input signal power of stage 1
Psi=Pni*Fa*SNRo;
disp('W',Psi,'Input signal power of stage 1');


//(c)Nosie added by stage 1
N=(Fa-1)*G1*Pni;
disp('W',N,'Noise added by stage 1');
```

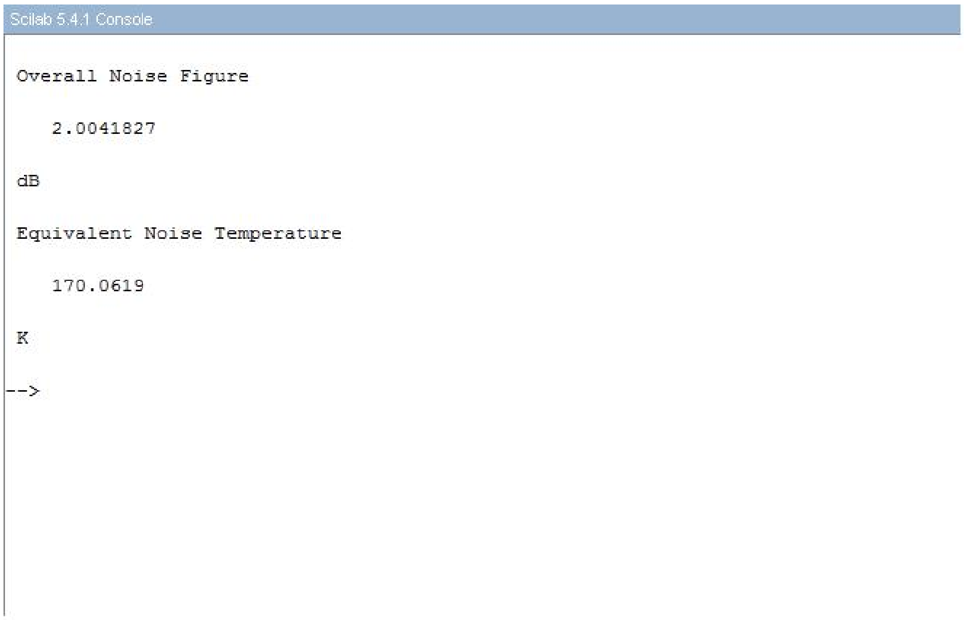


Figure 8.6: Noise figure and temperature

Scilab code Exa 8.7 Noise figure and temperature

```
//Page Number: 8.10
//Example 8.7
clc;

//Given
Te=127;//Kelvin
T=290;//Kelvin

G1=100;

F2dB=12;//db
F2=(10^(F2dB/10));

F1=1+(Te/T);

F=F1+((F2-1)/G1);
```

```
FF=(10*log10(F));
disp('dB',FF,'Overall Noise Figure');

//Equivalent Noise Temperature TE
TE=(F-1)*T;
disp('K',TE,'Equivalent Noise Temperature');
```

Scilab code Exa 8.9 WLAN Reciever

```
//Page Number: 8.11;
//Example 8.9
clc;

//(a)Noise Figure
//Given
//Loss Fcator
IL=1.5; //dB
IL1=(10^(IL/10));

//Noise figure
F1=1.41;
G1=1/F1;

G2=10;
GG2=(10*log10(G2)); //dB

G3=100;
GG3=(10*log10(G3)); //dB

F2=2; //dB
F3=2; //dB

FF2=(10^(F2/10));
```

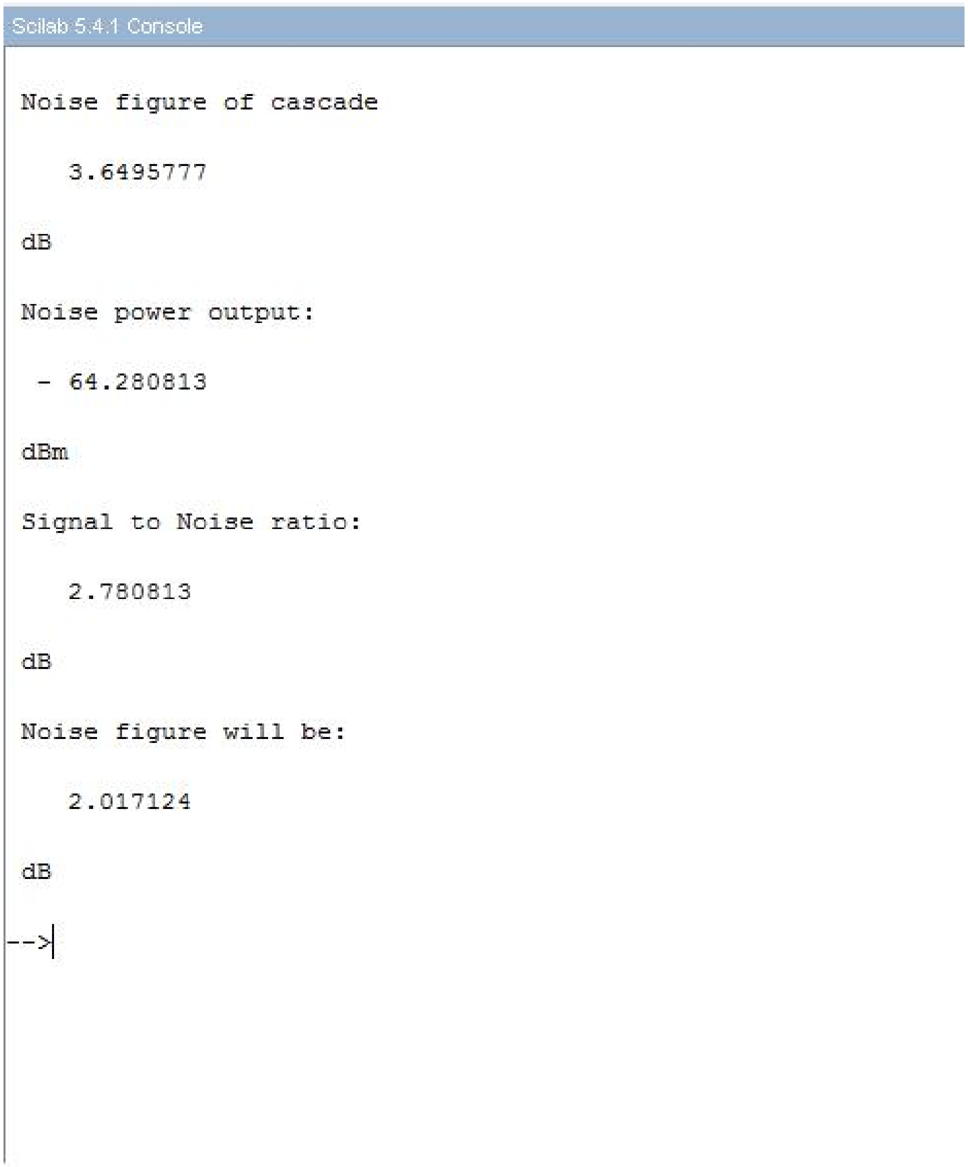


Figure 8.7: WLAN Reciever

```
FF3=(10^(F3/10));

F=(F1+((FF2-1)/G1)+((FF3-1)/(G1*GG2)));
FF=(10*log10(F));
disp('dB',FF,'Noise figure of cascade');

//(b) SNR at output
//Given
Pin=-90; //dBm
Pout=Pin-IL+GG2+GG3; //dBm

//Pn=Gcas*K*Te*B (cascade)
K=1.38D-23;
To=290; //Kelvin
B=1D+8;
Gcas=GG2+GG3-IL;
Gcas1=(10^(Gcas/10));
Pn=K*To*(F-1)*B*Gcas1; //W

Pn1=(10*(log10(Pn/1D-3)));
disp('dBm',Pn1,'Noise power output:');

SNR=Pout-Pn1;
disp('dB',SNR,'Signal to Noise ratio:');

//(c)Best Noise Figure
//G1 after G2 after IL

Fcas=(FF2+((FF3-1)/G3)+((IL1-1)/(G3*G2)));
Fcas1=(10*(log10(Fcas)));

disp('dB',Fcas1,'Noise figure will be:');
```

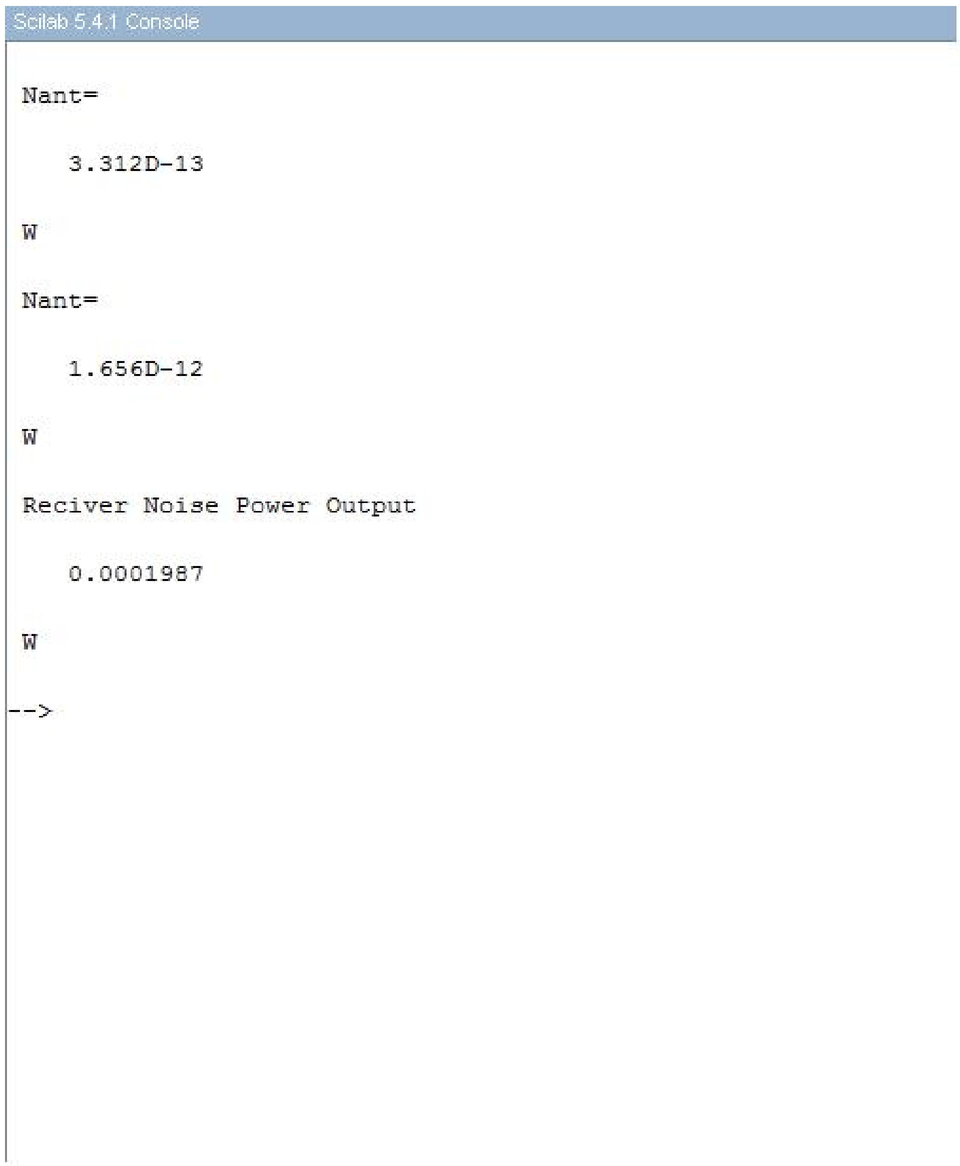


Figure 8.8: Receiver Noise Power

Scilab code Exa 8.10 Receiver Noise Power

```
//Page Number: 8.12
//Example 8.10
clc;


//Given
K=1.38D-23;
B=40D+6;

Tant=600; //Kelvin
Trec=3000; //Kelvin

G=80; //dB
GG=(10^(G/10));

//Input noise power from antenna
Nant=K*Tant*B; //W
disp('W',Nant,'Nant=');

Nrec=K*Trec*B; //W
disp('W',Nrec,'Nant=');

Nout=(Nant+Nrec)*GG;
disp('W',Nout,'Reciver Noise Power Output');
```

Scilab code Exa 8.11 Cascade Amplifiers A and B

```
//Page Number: 8.12
//Example 8.11
clc;

```

```
Scilab 5.4.1 Console
 GA  =

    10.
 GB  =

    15.

 Gain when Amp A followed by Amp B

    1.5995262

 Gain when Amp B followed by Amp A

    2.0110737
 T0  =

    20.

 Noise temperataure when Amp A followed by Amp B

    175.66119

 Noise temperataure when Amp B followed by Amp A

    296.2446

-->
```

Figure 8.9: Cascade Amplifiers A and B

```
//We use, F=(F1+(F2-1)/G1)
//Given
FA=1.5;

GA=10, //dB
GAA=(10^(GA/10));

FB=3; //dB
FBB=(10^(FB/10));

GB=15, //dB
GBB=(10^(GB/10));

//Case 1: Amp A followed by Amp B
F11=FA;
F12=FBB;
G11=GAA;

F1=(F11+(F12-1)/G11);
disp(F1,'Gain when Amp A followed by Amp B');

//Case 2: Amp B followed by Amp A
F21=FBB;
F22=FA;
G21=GBB;

F2=(F21+(F22-1)/G21);
disp(F2,'Gain when Amp B followed by Amp A');

//As F1<F2, Case 1 gives lowest Noise

//Also given,
T0=20 //degree celsius
T=T0+273; //Kelvin

//For amplifier A
TA=((FA-1)*T);

```

```
//For amplifier B
TB=((FBB-1)*T);

//When A is followed by B
Te1=(F1-1)*T;
disp(Te1,'Noise temperataure when Amp A followed by
    Amp B');

//When B is followed by A
Te2=(F2-1)*T;
disp(Te2,'Noise temperataure when Amp B followed by
    Amp A');
```

Scilab code Exa 8.12 Distortion Component

```
//Page Number: 8.13
//Example 8.12
clc;
//Given, Noise figure
NF=2; //dB
F=(10^(NF/10));

AG=12; //dB
A=(10^(AG/10));

//(a)Total Output Noise Power

//Also given,Input signal power
Pi=1; //W

//Input Noise power Pni
Pni=100D-3; //W

```

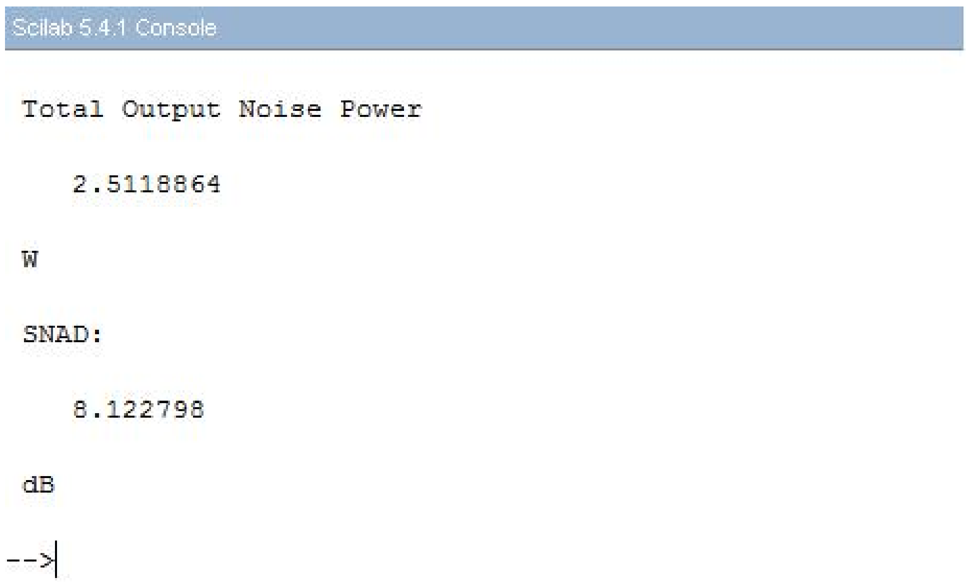


Figure 8.10: Distortion Component

```
//Input SNR
Isnr=Pi/Pni;

//Output SNR
Osnr=Isnr/F;

//Total output signal power
Po=Pi*A; //W

//Total output noise power
N=Po/Osnr; //W
disp('W',N,'Total Output Noise Power');

//(b)Signal to Noise and disortion ratio

//Given. 2% is disortion
Di=2/100;

//Total disortion
D=Di*A; //W

//Useful Power
S=(1-Di)*A; //W

//As given,SNAD=10*(log10(S+N+D)/(N+D));
SNAD=10*(log10((S+N+D)/(N+D)));
disp('dB',SNAD,'SNAD:');
```

Scilab code Exa 8.13 Receiver Antenna System

```
//Page Number: 8.14
//Example 8.13
clc;
```

```
Scilab 5.4.1 Console

 Noise tempertaure of antenna

    362.31884

 K

 Effective Noise tempertaure

    129.17753

 K

 Output Noise:

  - 68.675706

 dBm

-->
```

Figure 8.11: Receiver Antenna System

```
//Given
Pni=-100; //dBm
PniW=((1D-3)*(10^(Pni/10)));

To=290; //K

F=1.6; //dB
NF=(10^(F/10));

//(a) Noise tempertaure of antenna
//As Te=Pni/K*B;
K=1.38D-23;
B=20D+6;
Te=(PniW/(K*B));
disp('K',Te,'Noise tempertaure of antenna');

//(b)Effective noise tempertaure
//Given,
G=30; //dB
GdB=(10^(G/10));

Tef=((NF-1)*To);
disp('K',Tef,'Effective Noise tempertaure');

//Output Noise Pno=K*T(Te+Tef)*B*GdB

Pno=K*(Te+Tef)*B*GdB; //W
Pno1=(10*(log10(Pno/1D-3)));
disp('dBm',Pno1,'Output Noise: ');
```

Scilab code Exa 8.14 TV Recievers

```
//Page Number: 8.14
```

```
Scilab 5.4.1 Console

 Overall Noise Figure:

    6.7809936

 db

 Overall Noise Figure:

    19.

 db

 Overall Noise Figure:

    9.4039983

 db

-->
```

Figure 8.12: TV Recievers

```
//Example 8.14
clc;
//Given
GG1=20;//dB
G1=(10^(GG1/10));

FF1=6;//dB
F1=(10^(FF1/10));

GG2=60;//dB
G2=(10^(GG2/10));

FF2=16;//dB
F2=(10^(FF2/10));

LF=3; //dB
FC=(10^(LF/10));
GC=1/FC;

//(a)Overall Noise Figure
//Usinng F=(F1+((F2-1)/G1)+((F3-1)(G1*G2)));

Fa=(F1+((FC-1)/G1)+((F2-1)/(G1*GC)));
FadB=(10*(log10(Fa)));
disp('db',FadB,'Overall Noise Figure:');


//(b)Noise figure, if pre-amplifier is removed and
   gain increased by 20dB

Fb=FC+((F2-1)/GC);
FbdB=(10*(log10(Fb)));
disp('db',FbdB,'Overall Noise Figure:');

//(c)Change in noise figure
//Again usinng F=(F1+((F2-1)/G1)+((F3-1)(G1*G2)));
Fc=(FC+((F1-1)/GC)+((F2-1)/(G1*GC)));
FcdB=(10*(log10(Fc)));
```

```

disp('db',FcdB,'Overall Noise Figure:');
```

Scilab code Exa 8.15 Reciever

```
//Page Number: 8.15
//Example 8.15
clc;

//Given Noise figure
NF=5; //dB
F=(10^(NF/10));

Ta=1050; //Kelvin

//(a) Overall Noise Figure
T=20; // degree Celsius
To=T+273; //Kelvin

//Effective Noise temperature
Te=((F-1)*To);
disp('K',Te,'Effective Noise temperature');

//Overall effective Noise Temperature
TIN=Ta+Te;
disp('K',TIN,'Overall Effective Noise temperature');

//Overall noise figure
ONF=(1+(TIN/To));
ONFdB=(10*(log10(ONF))); //dB

disp('dB',ONFdB,'Overall Noise Figure:');

```

```
Scilab 5.4.1 Console

Effective Noise temperature

   633.54735

K

Overall Effective Noise temperature

   1683.5474

K

Overall Noise Figure:

   8.290396

dB

Input signal Power:

 - 135.05886

dBW

Equivalent Noise temperature:

   418.02117

K

Effective Noise temperature:

   1468.0212

K

Overall Noise Figure:

   8.290396

W

Input Signal Power:

   6.982D-14

W

Minimum detectable signal Vmin:
```

```
//(b)Input Signal Power
//Given Output SNR
Outsnr=6; //dB
Osnr=(10^(Outsnr/10));

Isnr=ONF*Osnr;

//Input Noise Power=KTB
K=1.38D-23;
B=50000;
Pni=K*TIN*B; //W

//Input signal Power
Psi=Isnr*Pni; //W
PsidBW=(10*(log10(Psi/1))); //dBW
disp('dBW',PsidBW,'Input signal Power:');


//(c)Minimum detectable signal Vmin
//Given
Osnr=10; //dB
R=50; //Ohms

FF1=3; //dB
F1=(10^(FF1/10));

FF2=5; //dB
F2=(10^(FF2/10));

GG1=7;
G1=(10^(GG1/10));
//Using F=F1+((F2-1)/G1)

Fa=F1+((F2-1)/G1);

Fa1=(10*(log10(Fa)));

```

```
//Equivalent Noise Tempertaure
Te1=((Fa-1)*To);
disp('K',Te1,'Equivalent Noise temperature:');

//Overall effective Noise Temperature
TIN1=Ta+Te1;
disp('K',TIN1,'Effective Noise temperature:');

//Input Noise Power=KTB
Pni1=K*TIN1*B; //W

//Overall noise figure
ONF1=(1+(TIN1/To));
ONFdB1=(10*(log10(ONF)));
disp('W',ONFdB1,'Overall Noise Figure:');

//Input SNR
Isnr1=ONF1*Osnr;

//Input signal Power
Psi1=Isnr1*Pni; //W
disp('W',Psi1,'Input Signal Power:');

//Now as Vmin^2/R=Psi1
//Therefore
Vmin=sqrt(Psi1*R);
disp('V',Vmin,'Minimum detectable signal Vmin:');
```

Scilab code Exa 8.18 Repeaters

```
//Page Number: 8.18
//Example 8.18
clc;
```

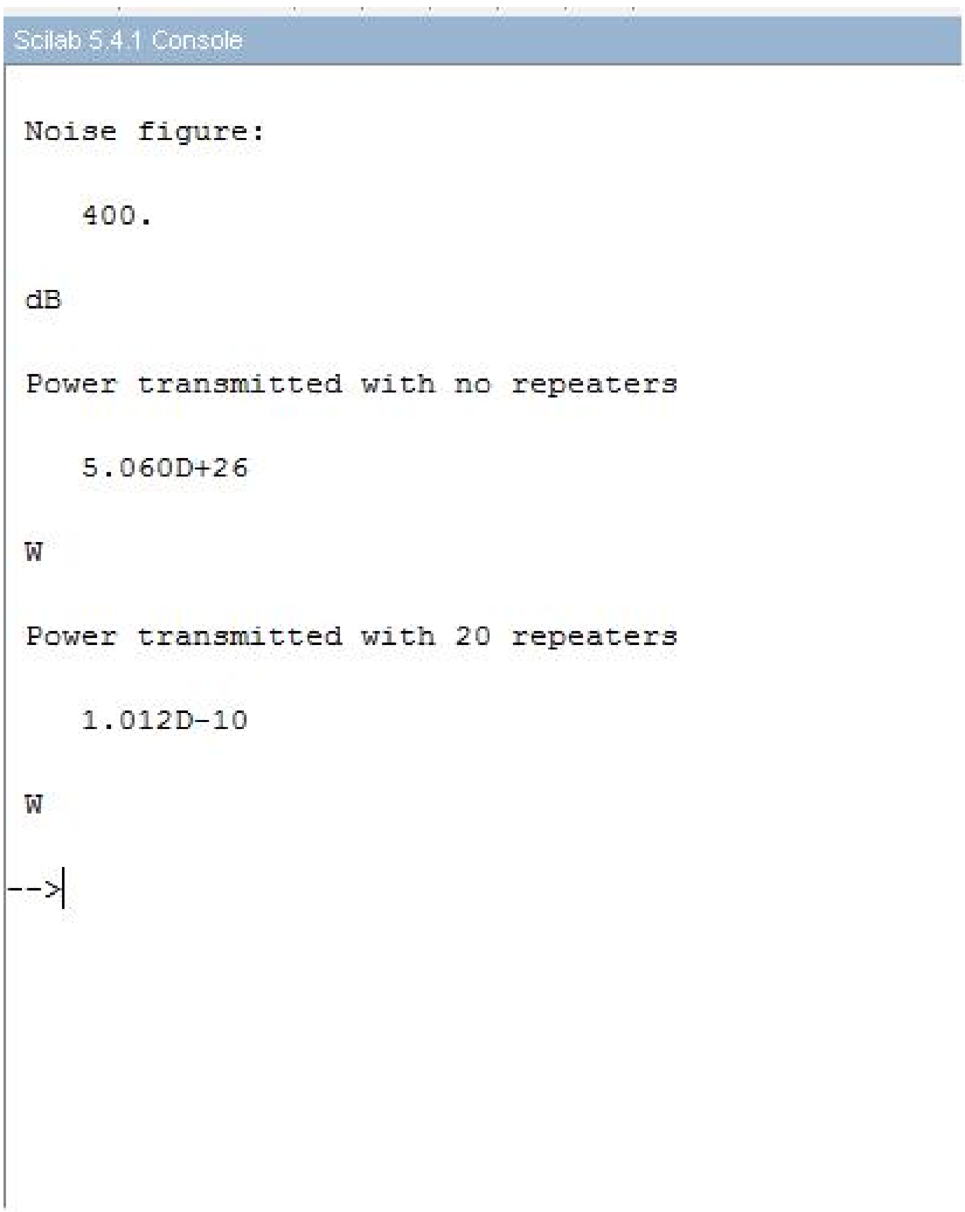


Figure 8.14: Repeaters

```
//Given,

Fa=5; //dB
d=200; //Km
a=2; //dB/Km
No=4D-21; //W/Hz
BW=4000;
Osnr=30; //dB
//(a) No repeaters used
L=d*a; //dB
disp('dB',L,'Noise figure:');

//As Output SNR=InputSNR/F where F=L*Fa
//And Input SNR=(Pt/(No*B))
//Therefore,PT=Output SNR+L+Fa+(No*B)

NoB=10*log10(No*BW);

//Power Transmitted
Pt=Osnr+L+Fa+(NoB);

PtdB=10^(Pt/10);
disp('W',PtdB,'Power transmitted with no repeaters')
    ;

//(b)20 repeaters are employed
n=20;
//F becomes 20F
//Output SNR=InputSNR/20*F where F=L*Fa
L1=L/n; //dB per segment

//Power Transmitted
Pt1=Osnr+L1+Fa+(NoB)+(10*(log10(n)));

PtdB1=10^(Pt1/10);
disp('W',PtdB1,'Power transmitted with 20 repeaters'
    );
```

Scilab code Exa 8.23 Noise process

```
//Page Number: 8.23
//Example 8.23
clc;
//Given,
// S=10D−8*(1−(|f|/10D+8));


//(a)Power contenet of output noise
//Bandwidth of 2MHz centered at 50MHz
//Therefore, first limits will be

x0=-51D+6;
x1=-49D+6;
P1=integrate('1+(f/10^8)','f',x0,x1);

//And,second limits will be

x2=49D+6;
x3=51D+6;

P2=integrate('1−(f/10^8)','f',x2,x3);

P=10D-8*(P1+P2);
disp('W',P,'Power content:');
```

Scilab code Exa 8.24 White Gaussian Noise

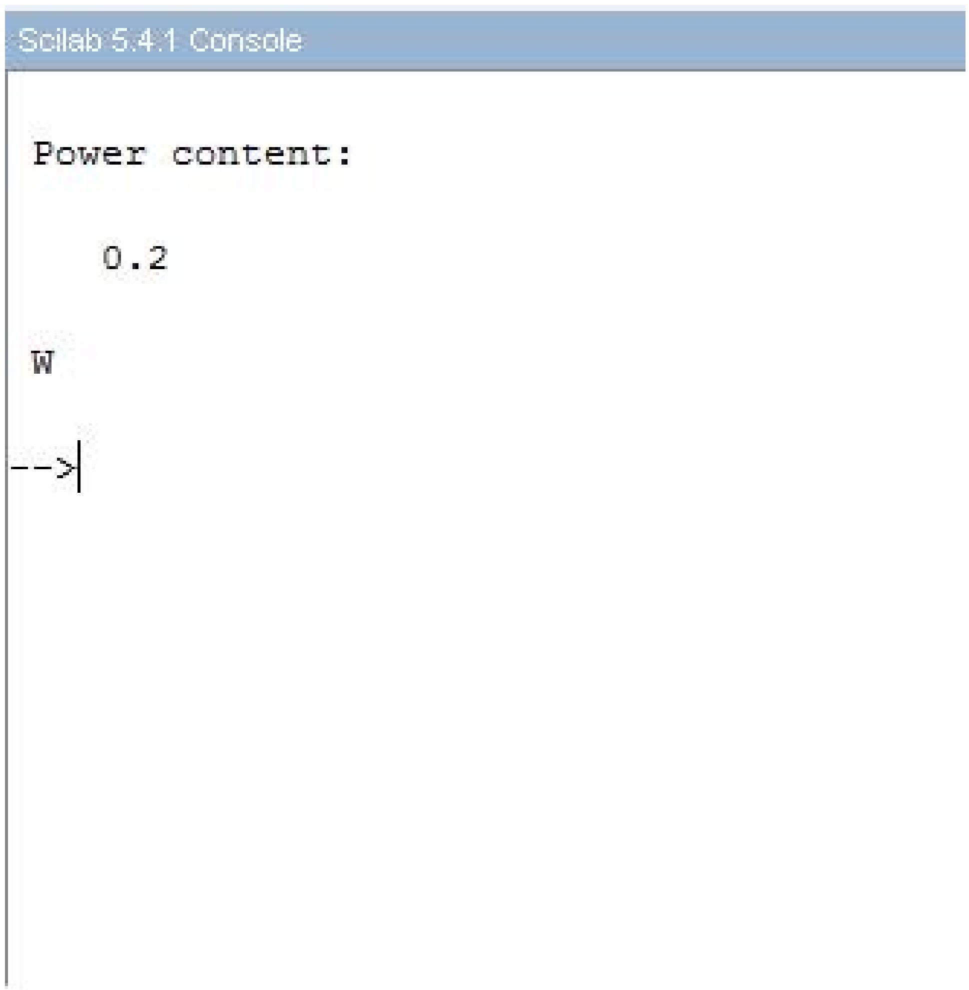


Figure 8.15: Noise process

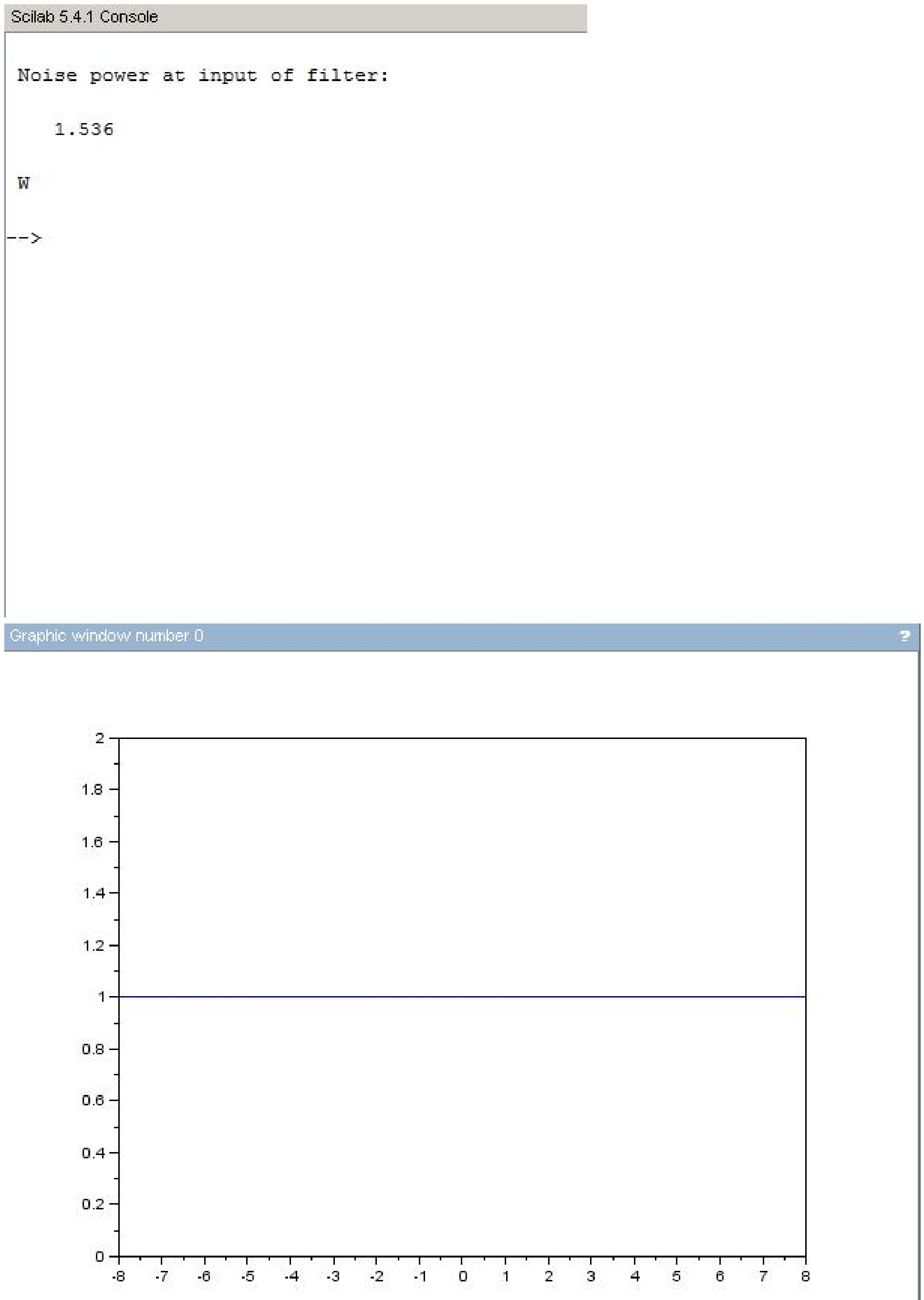


Figure 8.16: White Gaussian Noise

```
//Page Number: 8.25
//Example 8.24
clc;
//Given, band limited gaussian noise with psd,
S=9.6D-5; //W/Hz for |f|<8kHz
L=100D-3; //H
R=100; //Ohms

//(a) Noise power at input of filter

x0=-8000;
x1=8000;
Pni=S*(integrate('1','f',x0,x1));
disp('W',Pni,'Noise power at input of filter:');

//Plot
x=linspace(-8,8,3);
y=linspace(0,1,1);

plot(x,y);
```

Chapter 9

Noise in Analog Communication System

Scilab code Exa 9.1 Output Noise Power

```
//Page Number: 9.10
//Example 9.1
clc;
//Given
wo=2*%pi*8000;
n=2D-9;
//N0=(n/4*%pi)integrate('1/(1+((w/w0)^2))','w',-%inf
    ,+%inf)
//Which yields
//Output Noise Power
N0=(wo*n)/4;
disp("W",N0,"Output Noise Power: ");
```

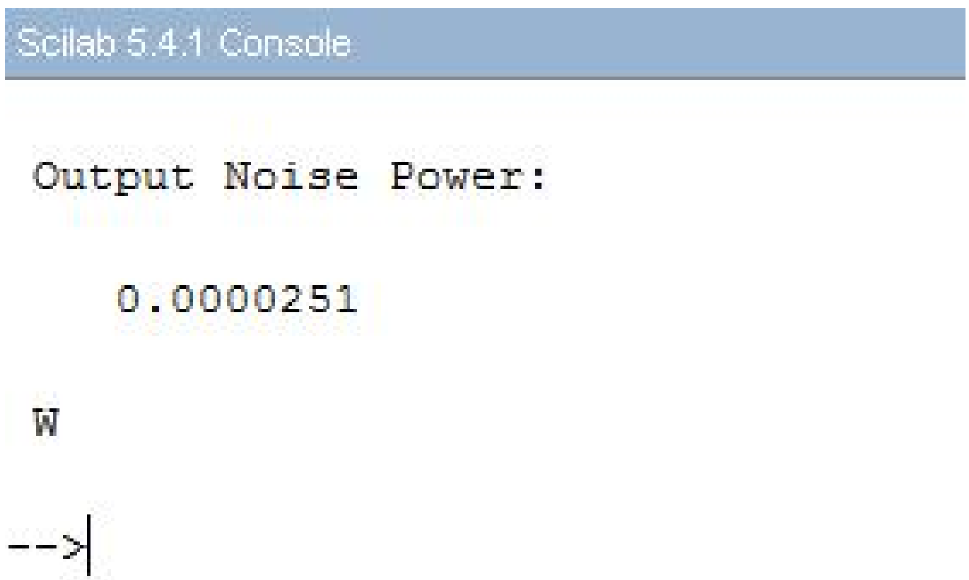


Figure 9.1: Output Noise Power

Figure 9.2: SNR of filters

Scilab code Exa 9.4 SNR of filters

```
//Page Number: 9.12
//Example 9.4
clc;
//Given
//(a)
H1=1;
H2=(1/2);
S0=(H1^2)/2;
w0=-2*%pi;
w1=2*%pi;
N0=(0.1/(2*%pi))*2*(integrate('1','w',w0,w1));
SNR=S0/N0;
disp(SNR,"SNR: ");

//(b)
S01=(H1^2*H2^2)/2;
N01=0.101;
SNR1=S01/N01;
disp(SNR1,"SNR1: ");
```

Scilab code Exa 9.8 AM with envelope detection

```
//Page Number: 9.15
//Example 9.8
clc;
//Given
p=0.99;
u=1;
q=1-p;
//As exp(-Ac^2/4*n*B)=1-p
//AndAC^2/2*n*B=S/N
```

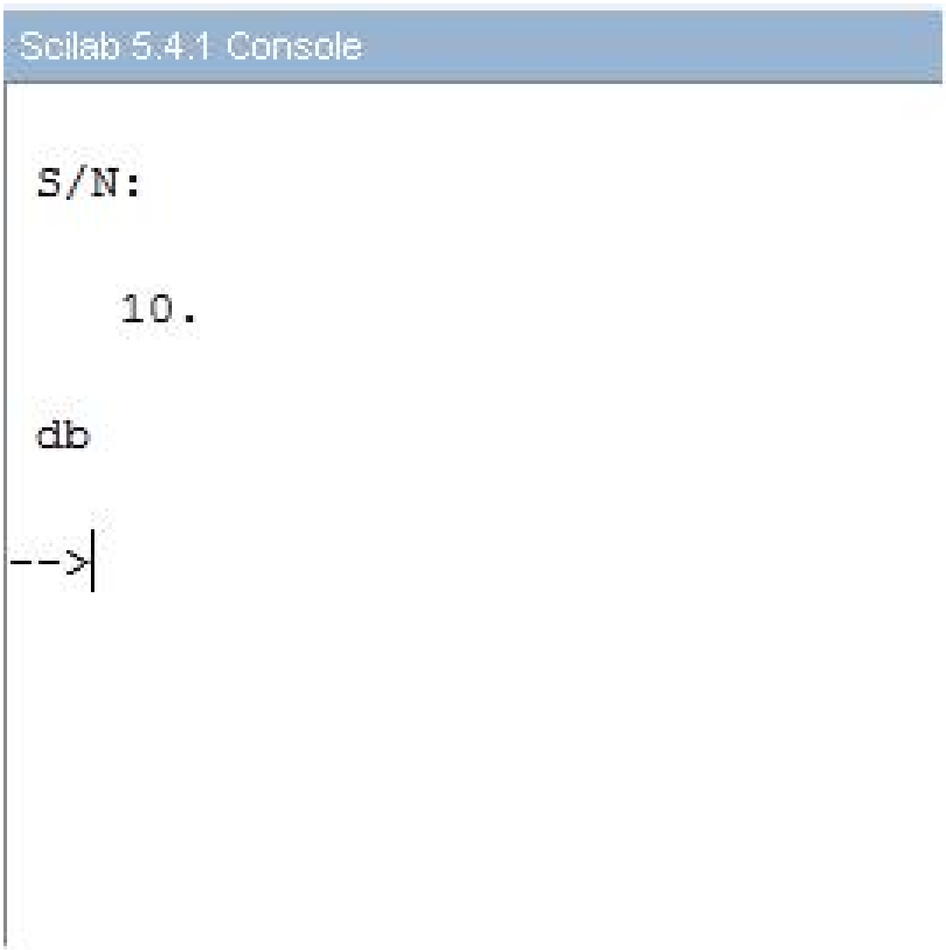


Figure 9.3: AM with envelope detection

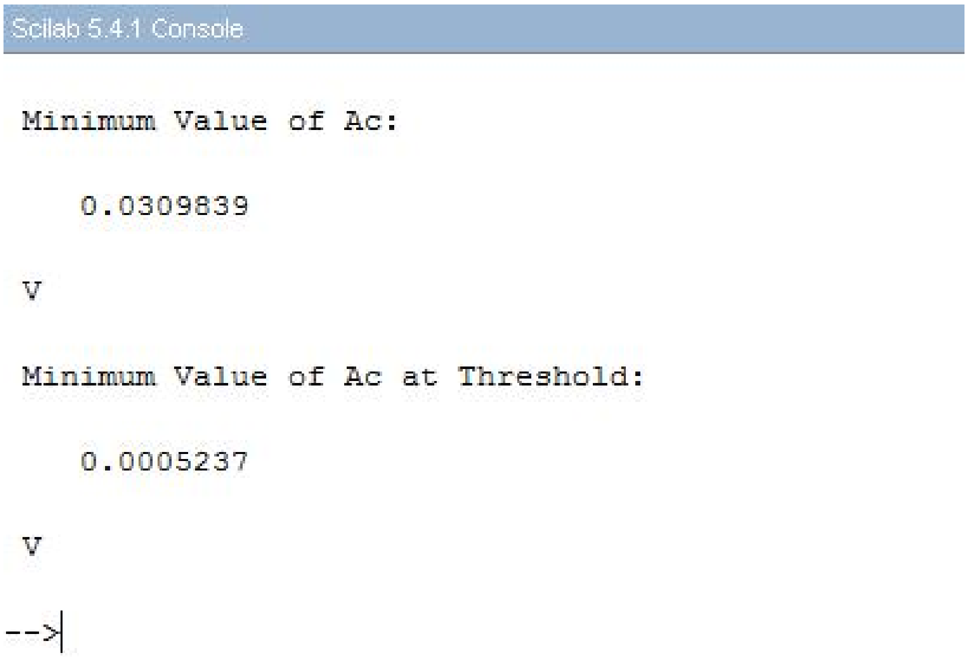


Figure 9.4: AM with additive thermal noise

```
//Therefore exp(-(1/2)*(S/N))=1-p
SN=2*(log(1/q));
SN1=(round(SN)+1); //Upper limit
disp('db',SN1,'S/N:');
//Hence proved
```

Scilab code Exa 9.9 AM with additive thermal noise

```
//Page Number: 9.16
//Example 9.9
clc;
//Given
Si=7D+4;
u=1;
BW=4D+3; //Hz
n=2D-12; //W/Hz

//(a)Minimum value of Ac
SbyN=40; //dB
SN=10^(SbyN/10);

Sx=2*[integrate('(x^2)*(-x+1)','x',0,1)];
//Now
g=SN/(Sx/(1+Sx));
//And
//Ac=sqrt((2*n*BW*g)/(1+(u^2*Sx))
//We have
Ac=sqrt((2*n*BW*g)/(1+(u^2*Sx)));
disp('V',Ac,'Minimum Value of Ac:');

//(b)Threshold value of Ac
//AS S/N at threshold is 10dB
SNT=10; //dB
gT=2*SNT;
AcT=sqrt((2*n*BW*gT)/(1+(u^2*Sx)));
disp('V',AcT,'Minimum Value of Ac at Threshold:');
```

Scilab code Exa 9.10 Modulation Systems transmission Bandwidth

```
//Page Number: 9.17
//Example 9.10
```

Scilab 5.4.1 Console

```
 Transmission Bandwidth for DSB:

    20000.

 Hz

 Transmission Bandwidth for AM:

    20000.

 Hz

 Transmission Bandwidth for SSB:

    10000.

 Hz

 Power transmission for DSB and SSB:

    2000.

 W

 Power transmission for AM:

    6000.

 W

-->
```

```
clc;
//Given
BW=10D+3; //Hz
SNR=40; //dB
SNRO=10^(SNR/10);
P=40; //dB
PL=10^(P/10);
n=2D-9; //W/Hz
USx2=0.5;
//For DSB, AM and SSB bandwidth requirement
BTD=2*BW;
BTA=2*BW;
BTS=BW;
disp('Hz',BTD,'Transmission Bandwidth for DSB:');
disp('Hz',BTA,'Transmission Bandwidth for AM:');
disp('Hz',BTS,'Transmission Bandwidth for SSB:');


//Pt for DSB and SSB
//As SNRO=Si/nBW
Si=n*BW*SNRO; //W
//Considering Channel loss
ST=Si*PL;
disp('W',ST,'Power transmission for DSB and SSB:');

//Pt for AM
//As SNRO=x*Si/nBW
//x=USx2/(1+USx)
x=USx2/(1+USx2);
Si1=(n*BW*SNRO)/x; //W
//Considering Channel loss
ST1=Si1*PL;
disp('W',ST1,'Power transmission for AM:');
```

Scilab 5.4.1 Console

```
 Ac for s(t)=100mw:

    3.6514837

 V

 Power in absence of Noise:

    0.025

 W

 Power:

    0.00015

 W

 SNR at output for SSB:

    22.218487

 dB
```



```
 SNR at output for DSB:

    25.228787
```

Scilab code Exa 9.13 SSB System

```
//Page Number: 9.20
//Example 9.13
clc;
//Given
Mf1=0.003;
//for f<=1.5D+3 Hz
f1=1.5D+3; //Hz

Mf2=0.001;
//for 1.5D+3 <=f<=3D+3 Hz
f2=3D+3; //Hz

Mf3=0;
//for f>3D+3 Hz


//(a) Ac that power transmitted is 100mW
St=100D-3; //W
//As St=2*[{f1*(Mf1*Ac/2)^2}+{f1*(Mf2*Ac/2)^2}+{f2*(
    Mf3*Ac/2)^2}]
//Neglecting Mf3 as zero
Ac=sqrt((4*St)/(2*f1*(Mf1^2+Mf2^2)));
disp('V',Ac,'Ac for s(t)=100mw:');

//(b)Power in abscence of noise
Zt=2*[{f1*(((Mf1*Ac)/4)^2)}+{f1*(((Mf2*Ac)/4)^2)}+{
    f2*(((Mf3*Ac)/4)^2)}];
disp('W',Zt,'Power in absence of Noise:');

//(c)
//Given
N0=0.0001D-3; //W/Hz
//Psd=N0/4
//Pt=2*f1*N0/4
Pt=(2*f2*N0)/4;
disp('W',Pt,'Power:');
```

```

//(d) SNR at output
SNR=Zt/Pt;
SNRO=10*log10(SNR);
disp('dB',SNRO,'SNR at output for SSB:');

//(e)For DSB
St1=100D-3; //W
//As St=4*[{f1*(Mf1*Ac/2)^2}+{f1*(Mf2*Ac/2)^2}+{f2*(Mf3*Ac/2)^2}]
//Neglecting Mf3 as zero
Ac1=sqrt((4*St)/(4*f1*(Mf1^2+Mf3^2)));

Zt1=4*[{f1*(((Mf1*Ac)/4)^2)}+{f1*(((Mf2*Ac)/4)^2)}+{f2*(((Mf3*Ac)/4)^2)}];

//SNR at output
SNR1=Zt1/Pt;
SNRO1=10*log10(SNR1);
disp('dB',SNRO1,'SNR at output for DSB:');

//3dB increase in SNR
//DSB has higher SNR but SSB os spectarally efficient
```

Scilab code Exa 9.14 FM broadcast System

```
//Page Number: 9.22
//Example 9.14
clc;
//Given
delf=75D+3; //Hz
W=15D+3; //Hz
```

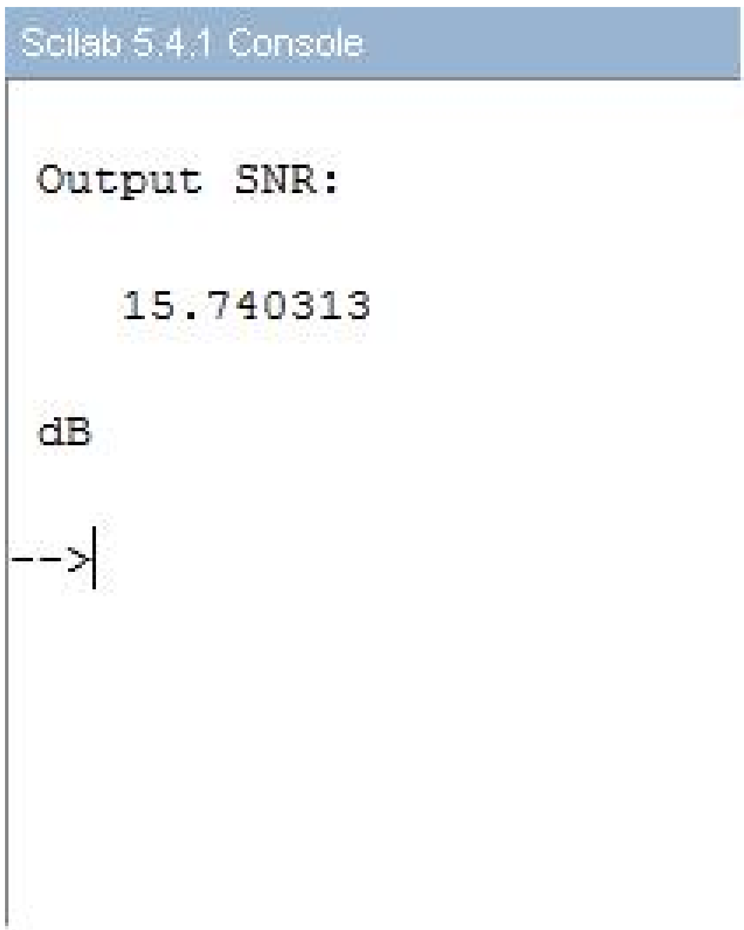


Figure 9.7: FM broadcast System

```
Sx=1/2;
//As SNRO=3(delf/W)^2*Sx*g
// Assume g=1
g=1;

SNRO=3*(delf/W)^2*Sx*g;
SNdB=10*log10(SNRO);
disp('dB',SNdB,'Output SNR:');

//Hence it is SNdB times better
```

Scilab code Exa 9.17 Audio Signal Over RF Channel

```
//Page Number: 9.23
//Example 9.17
clc;
//Given
oSNR=40; //dB
SNRO=10^(oSNR/10);
n=2D-10; //W/Hz
l=50; //dB
PL=10^(l/10);
B=15D+3; //Hz
Sx=1/2;
//(a) DSB Modulation
BTD=2*B;
disp('Hz',BTD,'Transmission bandwidth for DSB:');
//As SNRO=Si/(n*B)
SiD=SNRO*n*B;
STD=SiD*PL;
disp('W',STD,'Average Power transmitted for DSB:');

//(b) AM
```

```
Scilab 5.4.1 Console

Transmission bandwidth for DSB:

   30000.

Hz

Average Power transmitted for DSB:

   3000.

W

Transmission bandwidth for AM:

   30000.

Hz

Average Power transmitted for AM:

   9000.

W

Transmission bandwidth for PM:

   120000.

Hz

Average Power transmitted for PM:

   666.66667

W

Transmission bandwidth for FM:
```

```
U=1;
U2Sx=U*U*Sx;
BTA=2*B;
disp('Hz',BTA,'Transmission bandwidth for AM:');
//As SNRO=x*Si/(n*B)
//where x=USx/(1+USx)
x=U2Sx/(1+U2Sx);
SiA=(SNRO*n*B)/x;
STA=SiA*PL;
disp('W',STA,'Average Power transmitted for AM:');

//(c)PM
kp=3;
BTP=2*(kp+1)*B;
disp('Hz',BTP,'Transmission bandwidth for PM:');
//As SNRO=kp^2*Sx*Si/(n*B)
SiP=(SNRO*n*B)/(Sx*(kp^2));
STP=SiP*PL;
disp('W',STP,'Average Power transmitted for PM:');

//(d)FM
D=5;
BTF=2*(D+1)*B;
disp('Hz',BTF,'Transmission bandwidth for FM:');
//As SNRO=3*D^2*Sx*Si/(n*B)
SiF=(SNRO*n*B)/(3*(D^2)*Sx);
STF=SiF*PL;
disp('W',STF,'Average Power transmitted for FM:');
```

Scilab code Exa 9.18 Modulation Index

```
//Page Number: 9.24
//Example 9.18
```

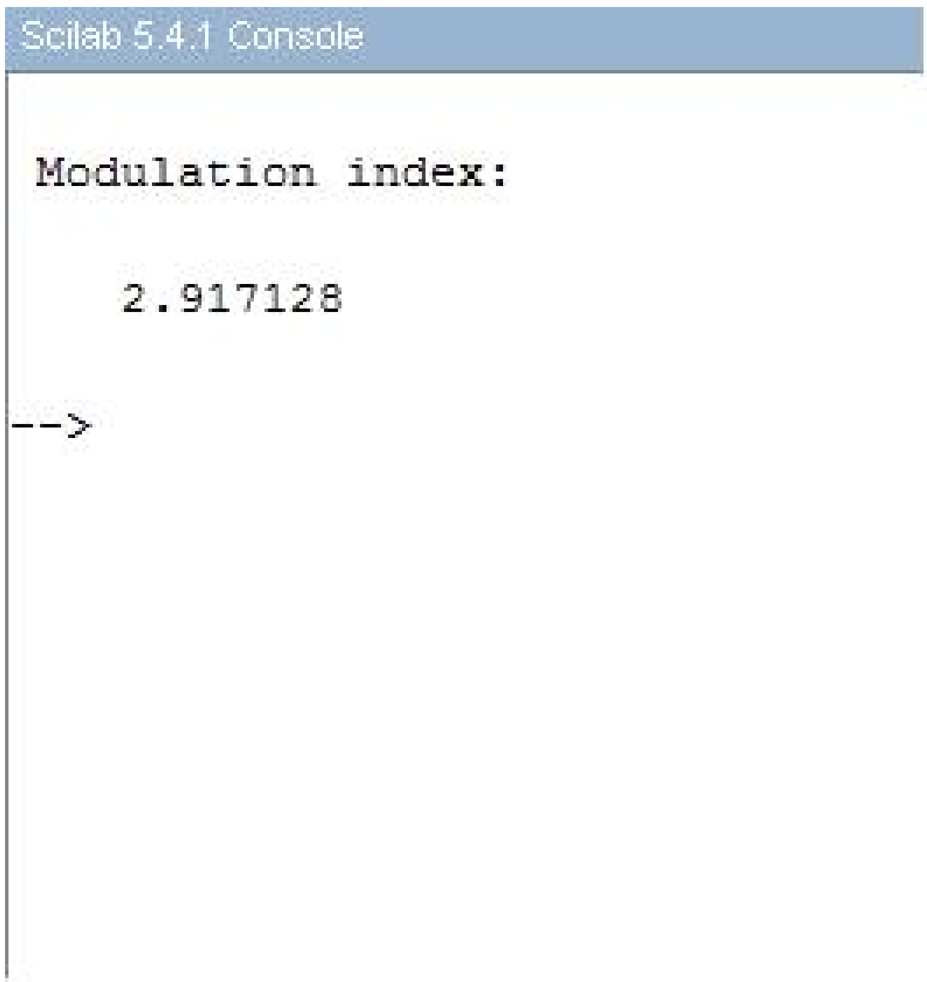


Figure 9.9: Modulation Index

```
clc;
//(b)Modulation index b
//Given
SNdB=30; //dB
SNRO=10^(SNdB/10);
//As SNRO=30*b^2*(b+1)
//Therefore
p2=poly(0,'x');
p3 =30*(p2^3)+30*(p2^2)-1000;
r=roots(p3);
t=r(3,1);
disp(t,'Modulation index:');
```

Scilab code Exa 9.21 Normalized Message Signal

```
//Page Number: 9.27
//Example 9.21
clc;
//Given
BW=5000; //Hz
P=0.1; //W
CBW=100D+3; //Hz
A=80; //dB
A1=10^(-A/10);
N0=2*(0.5D-12); //W/Hz
Pt=10D+3; //W

//We know, CBW=2*(D+1)*BW
//Therefore
D=(CBW/(2*BW))-1;
kp=D;
Si=Pt*A1;
Sx=P;
```

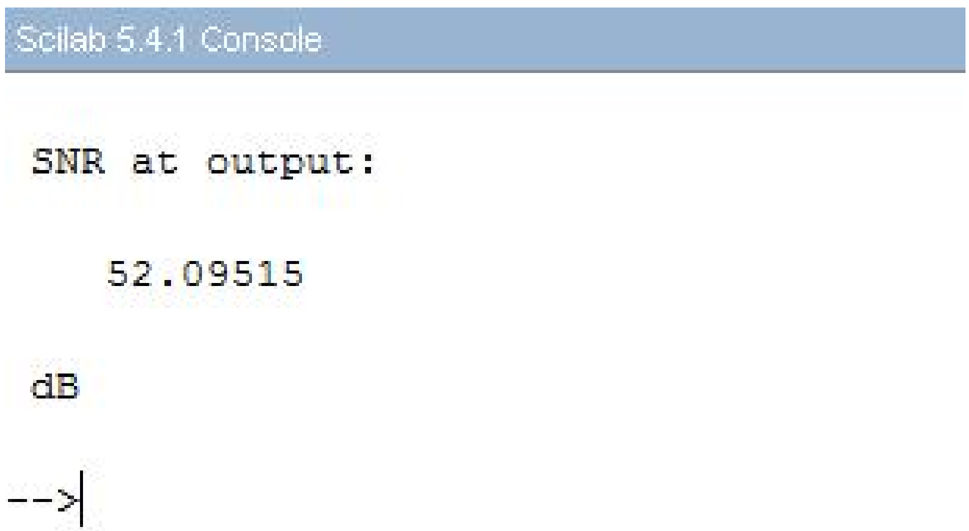


Figure 9.10: Normalized Message Signal

```
//We know
//SNR=((kp^2)*Si*Sx)/(N0*BW);

SNR=((kp^2)*Si*Sx)/(N0*BW);
SNR1=10*log10(SNR);
disp("dB", SNR1,"SNR at output: ");
```

Chapter 10

Digital Modulation and Demodulation

Scilab code Exa 10.1 Binary Reciever

```
//Page Number: 10.18
//Example 10.1
clc;
//Given
Rb=1D+6; //b/s
A=5D-3; //V
N0=0.5D-11; //W/Hz

Tb=1/Rb;
Eb=(A*A*Tb)/2;

//(a) ASK
//Pe=Q(x)
//where
xA=sqrt(Eb/N0);

PeA=(1/2)*erfc(xA/1.414);
```

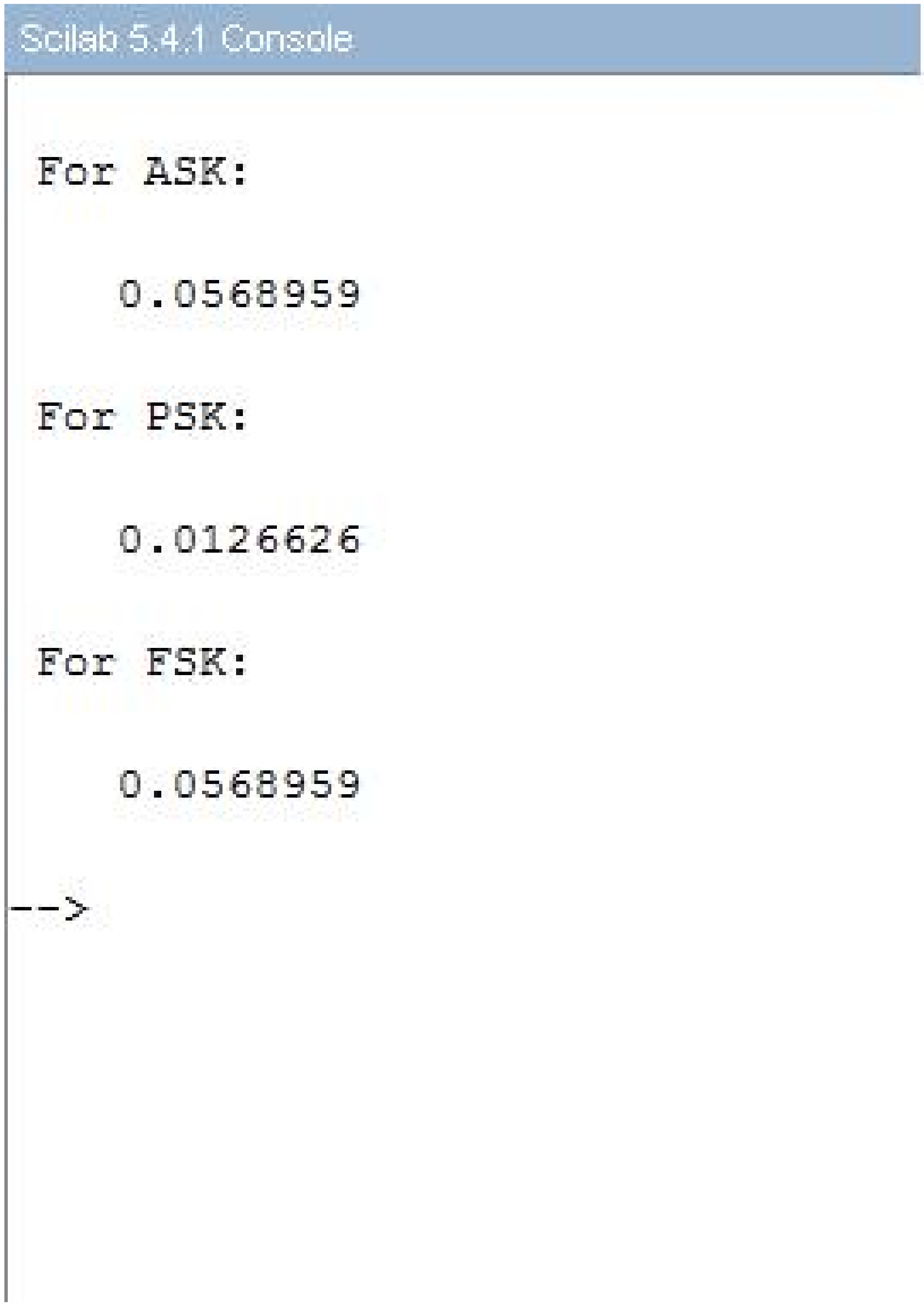


Figure 10.1: Binary Reciever

```
disp(PeA,'For ASK:');

//(b) PSK
//Pe=Q(x)
//where
xP=sqrt((2*Eb)/N0);

PeP=(1/2)*erfc(xP/1.414);
disp(PeP,'For PSK:');

//(c) FSK
//Pe=Q(x)
//where
xF=sqrt(Eb/N0);

PeF=(1/2)*erfc(xF/1.414);
disp(PeF,'For FSK:');
```

Scilab code Exa 10.2 BPSK Reciever

```
//Page Number: 10.19
//Example 10.2
clc;
//Given
Rb=1D+6; //b/s

//(a)
Aa=1D-3; //V
N0=1D-11; //W/Hz

Tb=1/Rb;
Eba=(Aa*Aa*Tb)/2;

```

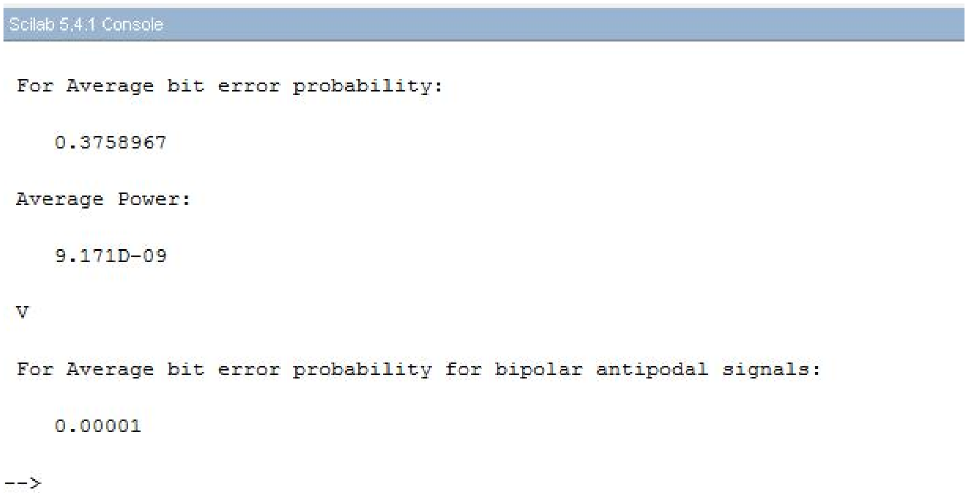


Figure 10.2: BPSK Reciever

```
//Pe=Q(z)
//where
za=sqrt((2*Eba)/N0);

Peb=(1/2)*erfc(za/1.414);
disp(Peb,'For Average bit error probability:');

//(b) Maintain Pb=2D-3
//From table
zb=2.9;
Ebb=((zb^2)*N0)/2;
P=Ebb*Tb;
Ab=sqrt((2*P));
disp('V',Ab,'Average Power:');

//(c)
Ac=100; //V
Rbc=1D+5; //p/s
```

```
N01=1D-2;; //W/Hz
Tbc=1/Rbc;
Ebc=(Ac*Ac*Tbc);

zc=sqrt(((2*Ebc)/N01));

Pec=(1/2)*erfc(zc/1.414);
//Pec=0.0000039
//nearly 10^-5
disp(10^-5,'For Average bit error probability for
    bipolar antipodal signals:');
```

Scilab code Exa 10.6 Satellite Communication

```
//Page Number: 10.23
//Example 10.6
clc;
//Given
p=0.1; //dB
p1=10^(-p/10);
p2=sqrt(p1);
t=acosd(p2);
a=round(t);
disp('degrees',a,'System cannot tolerate more than:'
    );
```

Scilab code Exa 10.11 Baseband Binary Transmission

```
//Page Number: 10.28
```

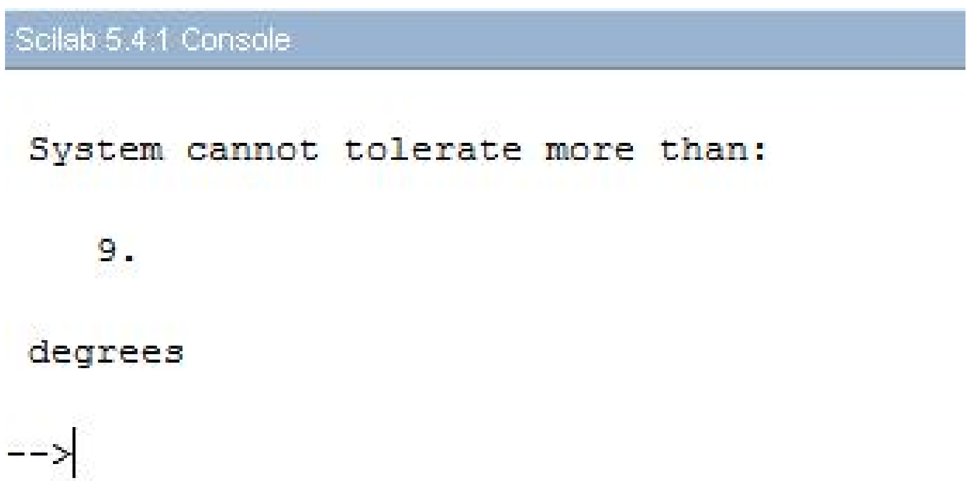


Figure 10.3: Satellite Communication

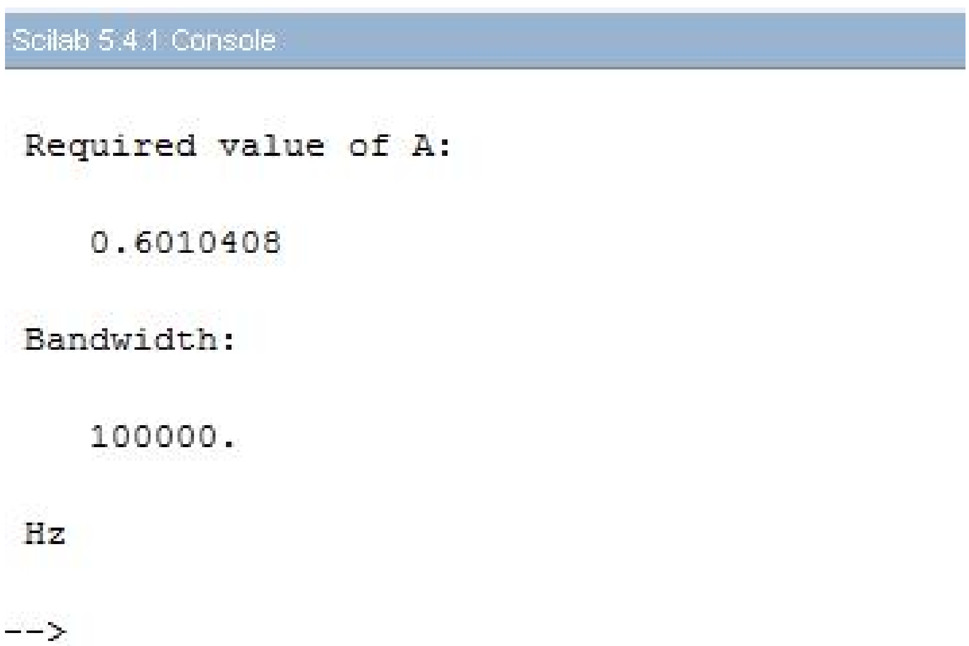


Figure 10.4: Baseband Binary Transmission

```
//Example 10.11
clc;
//Given
Rb=1D+6; //b/s
T=1D-5;
N0=2*0.5D-7; //W/Hz

//From table for Q(z)=10^-5
z=4.25;

//As z=sqrt(A*A*T/2*N0)
x=((z^2)*2*N0)/T;
A=sqrt(x);
disp(A,'Required value of A:');

//(b) Bandwidth
//B=1/(2*(T/2))
//Therefore B=1/T
B=(1/T);
disp('Hz',B,'Bandwidth:');
```

Scilab code Exa 10.17 On Off Binary System

```
//Page Number: 10.33
//Example 10.17
clc;
//Given
A=0.2D-3; //V
T=2D-6; //s
n=2*1D-15; //W/Hz


//Pe=Q(z)
```

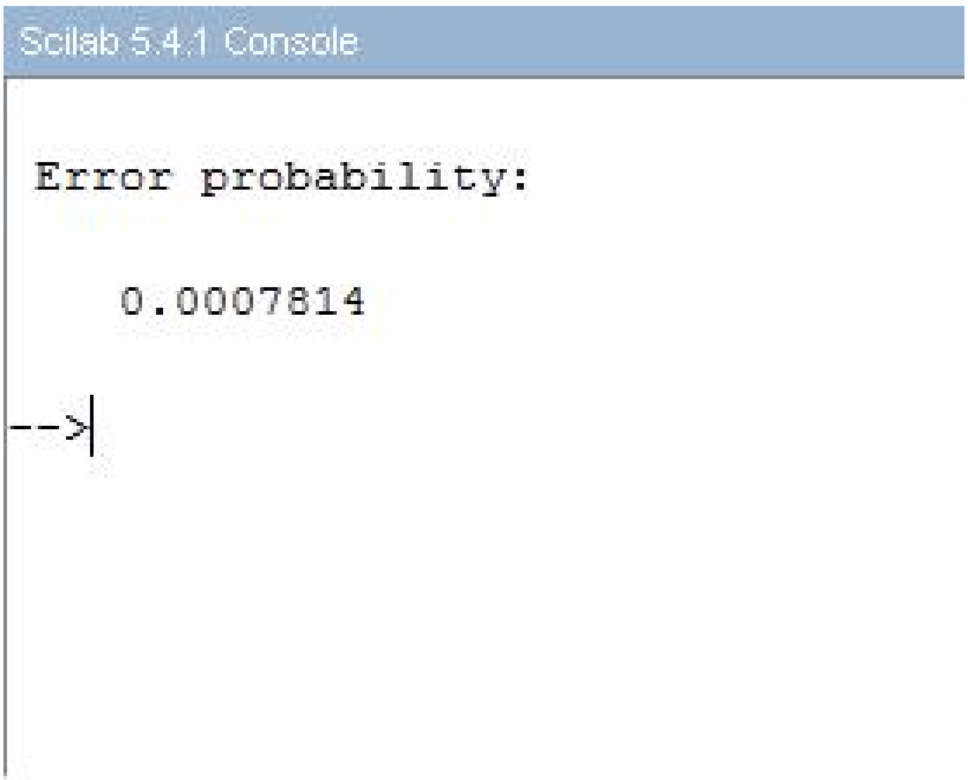


Figure 10.5: On Off Binary System

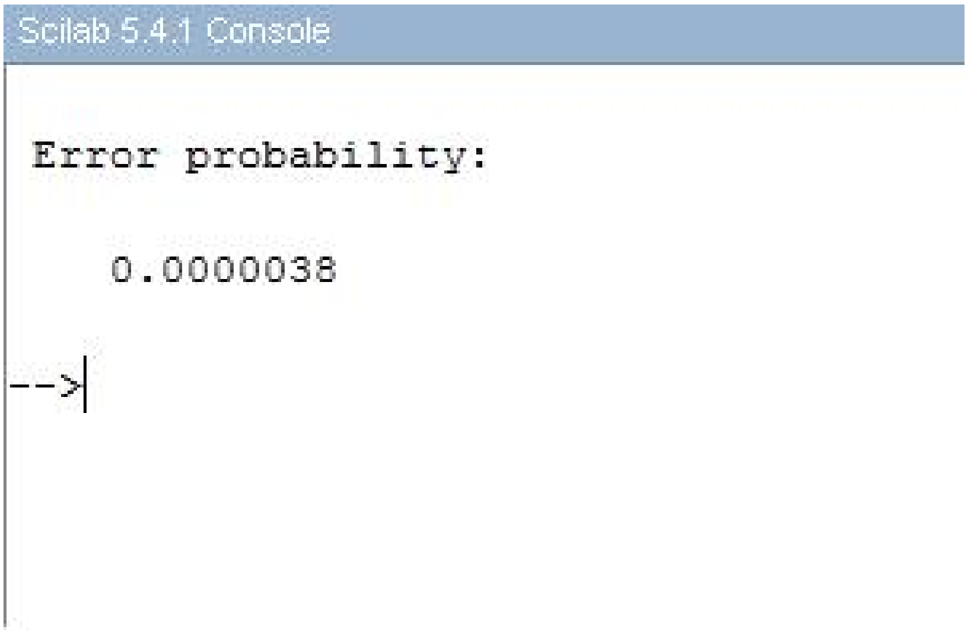


Figure 10.6: Binary System

```
//where
x=(A*A*T)/(4*n);
z=sqrt(x);
Pe=(1/2)*erfc(z/1.414);
disp(Pe,'Error probability:');
```

Scilab code Exa 10.18 Binary System

```
//Page Number: 10.34
//Example 10.18
clc;
//Given
A=0.2D-3; //V
A1=A/1.414;

T=2D-6; //s
n=2*1D-15; //W/Hz


//Pe=Q(z)
//where
x=(A1*A1*T)/(n);
z=sqrt(x);
Pe=(1/2)*erfc(z/1.414);
disp(Pe,'Error probability:');
```

Scilab code Exa 10.19 Binary Data Transmission

```
//Page Number: 10.35
//Example 10.19
clc;
//Given
d12sqr=8;
N0=2*0.5; //W/Hz

//(c)
//As for two equiprobables
//Pe=Q(z)
//where z=sqrt(d12^2)/sqrt(2*N0)
```

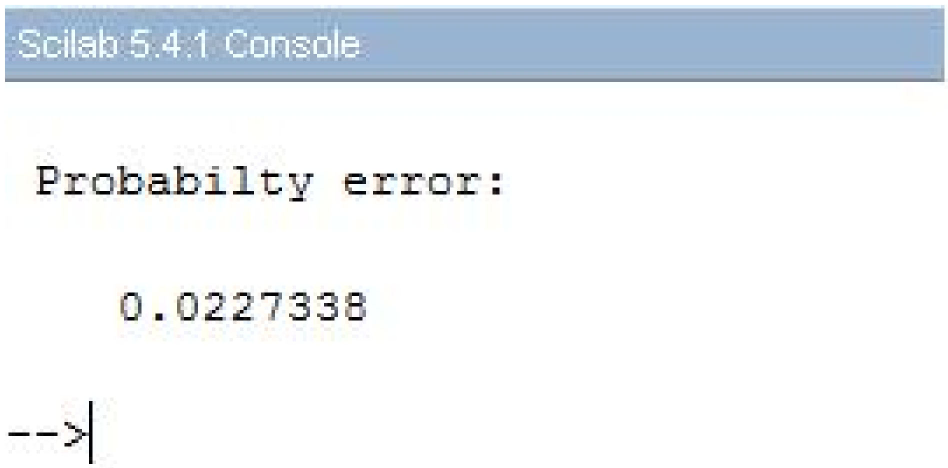


Figure 10.7: Binary Data Transmission

```
z=sqrt((d12sqr)/(2*N0));
Pe=(1/2)*erfc(z/1.414);
disp(Pe,'Probabilty error:')
```

Scilab code Exa 10.20 Binary Communication

```
//Page Number: 10.36
//Example 10.20
clc;
//Given
c=sqrt(2);
A=sqrt(5);
N0=1; //W/Hz

d12sqr=4*A*A;

//As for two equiprobables
//Pe=Q(z)
//where z=sqrt(d12^2)/(2*N0)
z=sqrt((d12sqr)/(sqrt(2*N0)));
Pe=(1/2)*erfc(z/1.414);
disp(Pe,'Probabilty error:')
```

Scilab code Exa 10.24 Voice Transmission BPSK

```
//Page Number: 10.41
//Example 10.24
clc;
//(a) Number of constellation Points
```

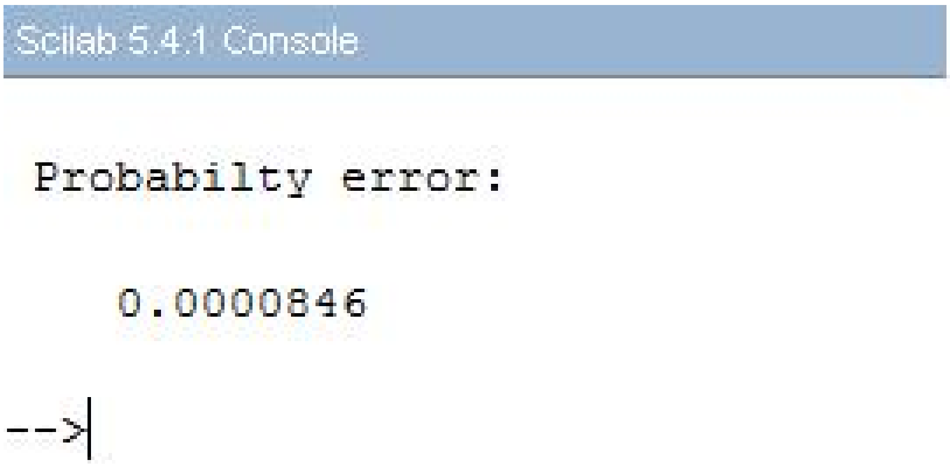


Figure 10.8: Binary Communication

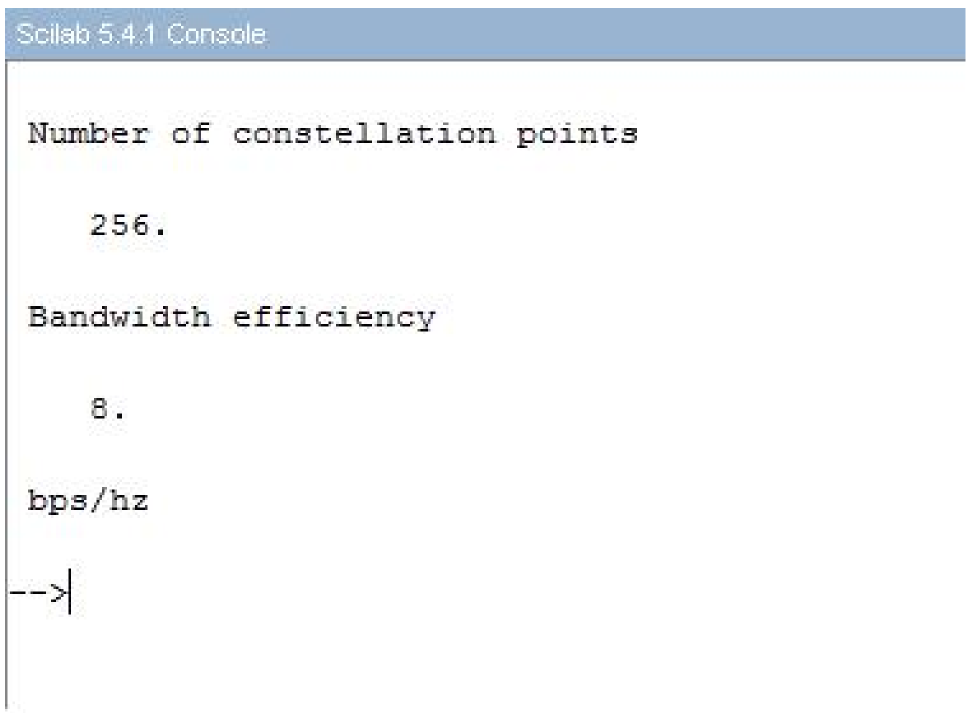


Figure 10.9: Voice Transmission BPSK

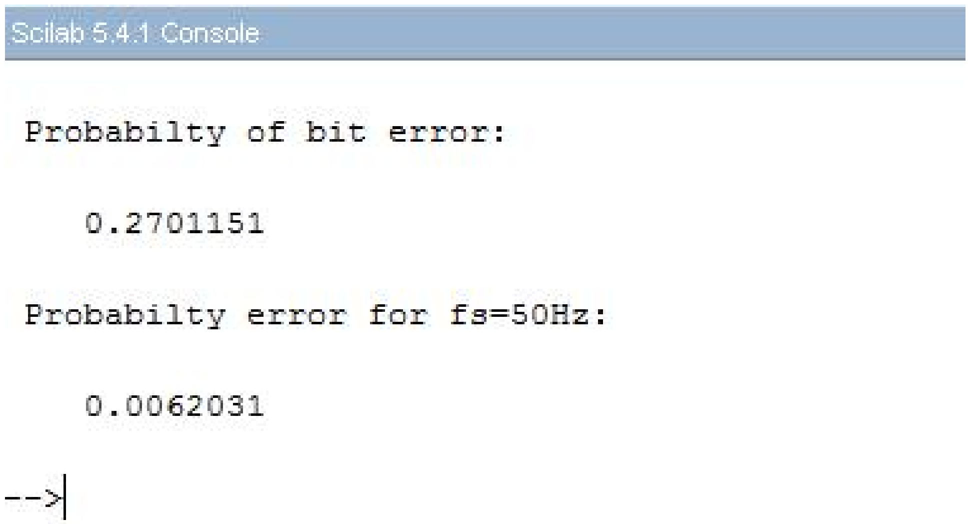


Figure 10.10: Cohereht Matched Reciever

```
//Given
Rs=2400; //bps
Rb=19.2D+3; //bps

//As Rs=Rb/log2M
M=2^(Rb/Rs);
disp(M,'Number of constellation points');

//(b) Bandwidth efficiency
BT=2400; //Symbols/second
n=Rb/BT;
disp('bps/hz',n,'Bandwidth efficiency');
```

Scilab code Exa 10.25 Cohereht Matched Reciever

```
//Page Number: 10.41
//Example 10.25
clc;
//Given
A1=0.5;
A2=0.5;
T=0.01; //sec
N0=2*0.0001; //W/Hz
f=50; //Hz

//(a) Probability of bit error
Es1=(A1^2*T)/2;
Es2=(A2^2*T)/2;

Eb=(Es1+Es2)/2;
//As PE=Qsqrt(Ep+Eq-2Epq/2N0)
//In this case Ep=Eq=Eb
//Therefore PE=Qsqrt(Eb(1-p)/N0)
//where p=Epq/Eb

//p=(1/Eb)*integrate('0.5*cos(2000*%pi*t)*0.5*cos
    (2020*%pi*t)','t',0,T);
//We get
p=0.94;
q=1-p;
//As Pe=Q(z)
//where z=sqrt(Eb/N0)
z=sqrt((Eb*q)/N0);
Pe=(1/2)*erfc(z/1.414);
disp(Pe,'Probabilty of bit error:')

//(b)
```

```
//Given
fs=50; //Hz
//or fs=1/2T where T=0.001
//This implies y=tone spacing will be orthogonal
//Therefor p=0

//As Pe=Q(z)
//where z=sqrt(Eb/N0)
zb=sqrt(Eb/N0);
PB=(1/2)*erfc(zb/1.414);
disp(PB,'Probabilty error for fs=50Hz:')
```

Scilab code Exa 10.27 M level PSK

```
//Page Number: 10.43
//Example 10.27
clc;
//Given
Rb=4.8D+3; //b/s
bw=3.2D+3; //Hz

//BPSK can give maximum spectral efficiency of 1bps/
   Hz, therefore not suitable
//QPSK can give twice spectral efficiency ,2bps/Hz,
   therefore
qpsk=2*bw;
//PSK can give thrice spectral efficiency ,3bps/Hz,
   therefore
psk=3*bw;

//QPSK is most suitable
Rs=Rb/2;
//Roll off Factor
```

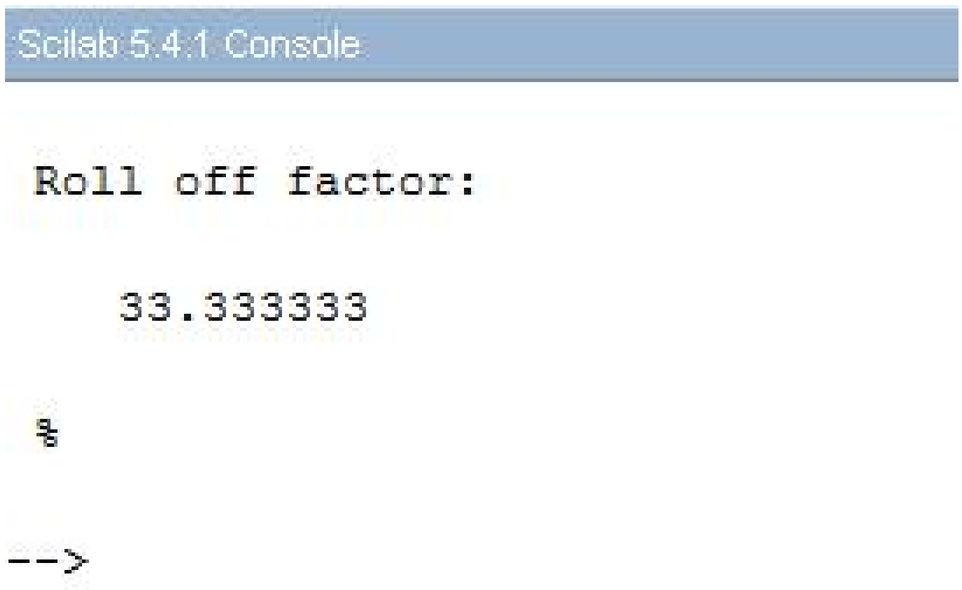


Figure 10.11: M level PSK

```
a=(bw/Rs)-1;
ap=a*100;
disp('%',ap,'Roll off factor:');
```

Chapter 11

Information Theory and Source Coding

Scilab code Exa 11.2 DMS X with four symbols

```
//Page Number: 11.12
//Example 11.2
clc;
//Given
//Probabilities of four symbols
Px=[0.4 0.3 0.2 0.1];

//(a) H(X)
//As H(X)=-Sum of(P(xi)log2P(xi))
//Where i=0 to n;
HofX=0;
for i=1:4
    HofX=HofX+(Px(i)*log2(Px(i)));
end
disp('b/symbol',-HofX,'H(X):');

//(b)Amount of information in x1x2x1x3 and x4x3x3x2
```

```
Scilab 5.4.1 Console

 H(X):

    1.8464393

 b/symbol

 Ix1x2x1x3:

    6.7027499

 b/symbol

 Ix4x3x3x2:

    9.7027499

 b/symbol

-->
```

Figure 11.1: DMS X with four symbols

```
Px1x2x1x3=Px(1)*Px(2)*Px(1)*Px(3);
Ix1x2x1x3=-log2(Px1x2x1x3);
disp('b/symbol',Ix1x2x1x3,'Ix1x2x1x3:');

Px4x3x3x2=Px(4)*Px(3)*Px(3)*Px(2);
Ix4x3x3x2=-log2(Px4x3x3x2);
disp('b/symbol',Ix4x3x3x2,'Ix4x3x3x2:');
```

Scilab code Exa 11.3 Binary Memoryless Source

```
//Page Number: 11.13
//Example 11.3
clc;
//As H(X) is maximum when
//Px1=Px2=1/2
Px=[0.5 0.5];

//As H(X)=-Sum of[P(xi)log2P(xi)]
//Where i=0 to n;
HofX=0;
for i=1:2
    HofX=HofX+(Px(i)*log2(Px(i)));
end
disp('b/symbol',-HofX,'Maximum H(X):');
```

Scilab code Exa 11.5 Black and white TV picture

```
//Page Number: 11.15
//Example 11.5
```

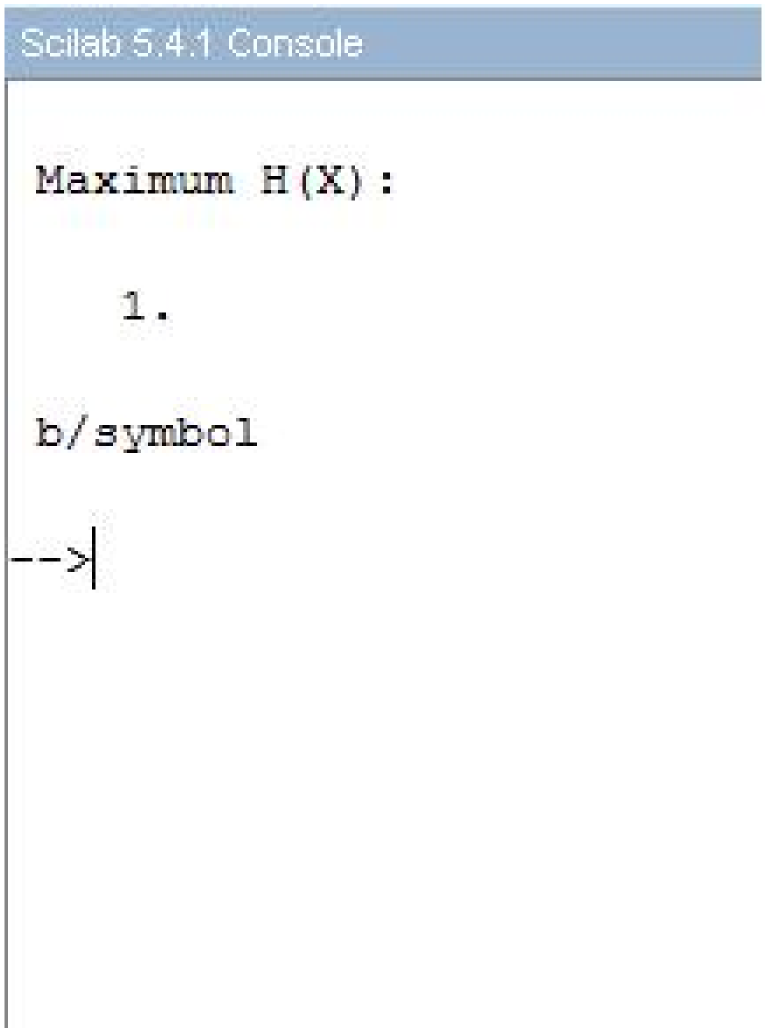


Figure 11.2: Binary Memoryless Source

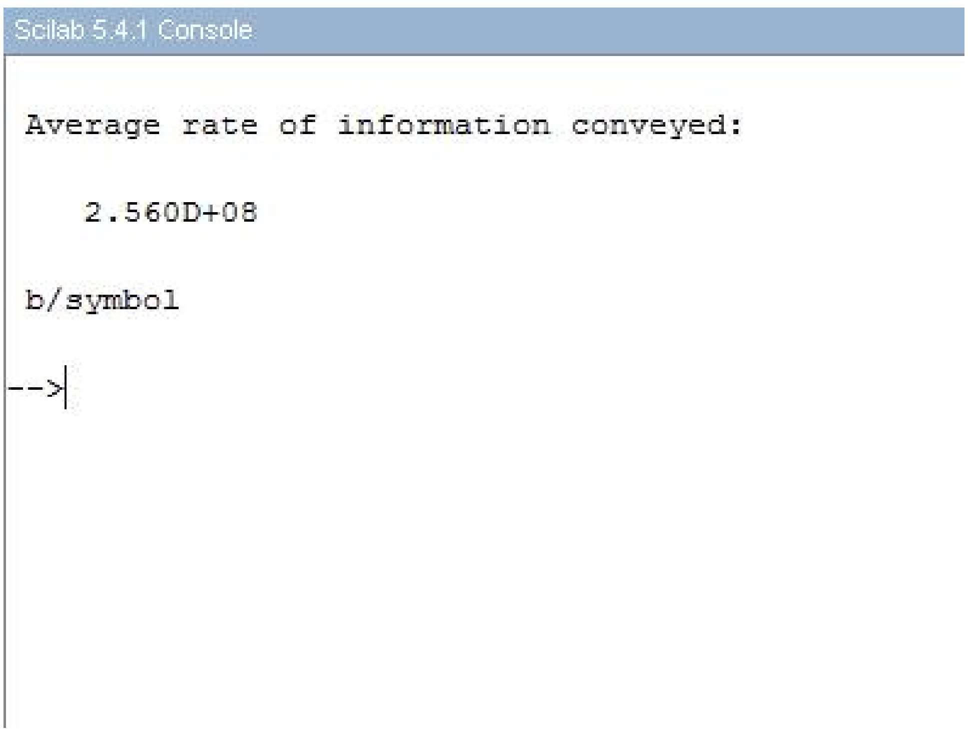


Figure 11.3: Black and white TV picture

```
clc;
//Given
//Picture elements
pe=2D+6;
//Brightness levels
l=16;
//Rate of repeatation
rr=32; //Per second


//As H(X)=-Sum of[P(xi)log2P(xi)]
//Where i=0 to n;
HofX=(-1)*l*[(1/l)*log2(1/l)];

r=pe*rr;

//As R=r*H(X)
R=r*HofX;
disp('b/symbol',R,'Average rate of information
   conveyed:');
```

Scilab code Exa 11.6 Telegraph Source

```
//Page Number: 11.15
//Example 11.6
clc;
//Given
//Pdot-2*Pdash and Pdot+Pdash=1
//Therfore , on solving
Pdot=2/3;
Pdash=1/3;

tdot=0.2; //Sec
```

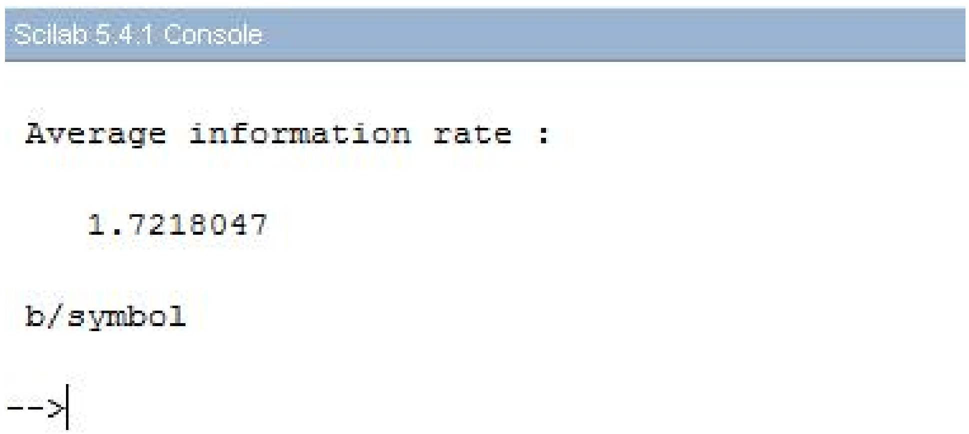


Figure 11.4: Telegraph Source

```
tdash=0.6; //Sec
tspace=0.2; //Sec

//Finding H(X)
//As H(X)=-Sum of[P(xi)log2P(xi)]
//Where i=0 to n;
HofX=(-1)*[{Pdot*log2(Pdot)}+{Pdash*log2(Pdash)}];

//Average time per symbol
Ts=(Pdot*tdot)+(Pdash*tdash)+tspace;

//Average Symbol Rate
r=1/Ts;

//Average information rate
R=r*HofX;
disp('b/symbol',R,'Average information rate :');
```

Scilab code Exa 11.7 Binary Channel

```
//Page Number: 11.15
//Example 11.7
clc;

//(a)Channel Matrix
//Given
Py1byx1=0.9;
Py2byx1=0.1;
Py1byx2=0.2;
Py2byx2=0.8;
PYbyX=[Py1byx1 Py2byx1;Py1byx2 Py2byx2];
disp(PYbyX,'Channel Matrix,P(Y/X):');

```

```
Scilab 5.4.1 Console

 Channel Matrix,P(Y/X):

    0.9    0.1
    0.2    0.8

 P(y1) P(y2):

    0.55    0.45

 P(x1,y2) P(x2,y1)

    0.05

    0.1

-->
```

Figure 11.5: Binary Channel

```
//(b)Py1 and Py2
//Given
Px1=0.5;
Px2=Px1;
//As P(Y)=P(X)*P(Y/X)
PX=[Px1 Px2];
PY=PX*PYbyX;
disp(PY,'P(y1) P(y2):');

//(c)Joint Probabilities P(x1,y2) and P(x2,y1)
//Diagonalizing PX
PXd=diag(PX);
PXY=PXd*PYbyX;
disp(PXY(2,1),PXY(1,2),'P(x1,y2) P(x2,y1)');
```

Scilab code Exa 11.8 Binary Channel in cascade

```
//Page Number: 11.16
//Example 11.8
clc;
//(a) Channel Matrix
//Given
PYbyX=[0.9 0.1;0.2 0.8];
PZbyY=[0.9 0.1;0.2 0.8];

//As P(Z/X)=P(Y/X)*P(Z/Y)
PZbyX=PYbyX*PZbyY;
disp(PZbyX,'Channel Matrix');

//(b)Pz1 and Pz2
//Given
Px1=0.5;
Px2=Px1;
```

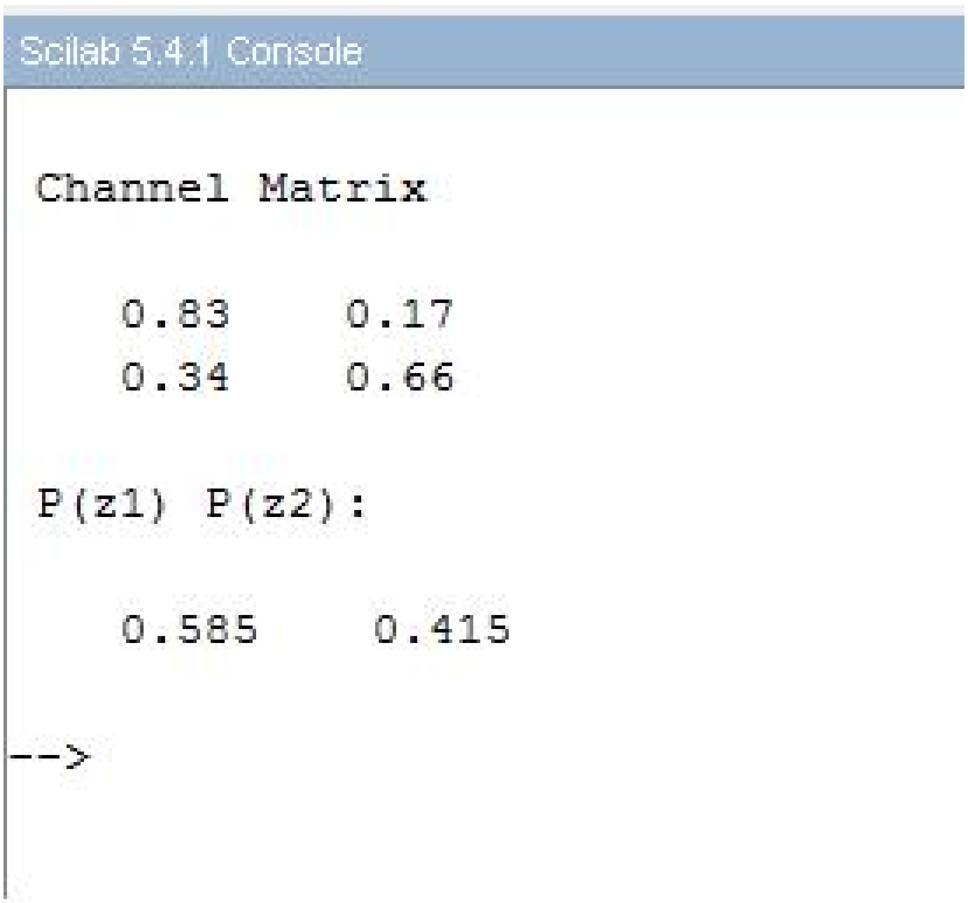


Figure 11.6: Binary Channel in cascade

```
//As P(Z)=P(X)*P(Z/X)

//P(X) matrix
PX=[Px1 Px2];
PZ=PX*PZbyX;
disp(PZ,'P(z1) P(z2):');
```

Scilab code Exa 11.9 Channel

```
//Page Number: 11.17
//Example 11.9
clc;
//Given
p=0.2;
Px1=0.5;
Px2=0.5;
//P(X) Matrix
PX=[Px1 Px2];
//Given
PYbyX=[(1-p) p 0;0 p (1-p)];
//P(y)=
PY=PX*PYbyX;
disp(PY,'P(y1) P(y2) P(y3):');
```

Scilab code Exa 11.16 BSC

```
//Page Number: 11.21
//Example 11.16
clc;
```

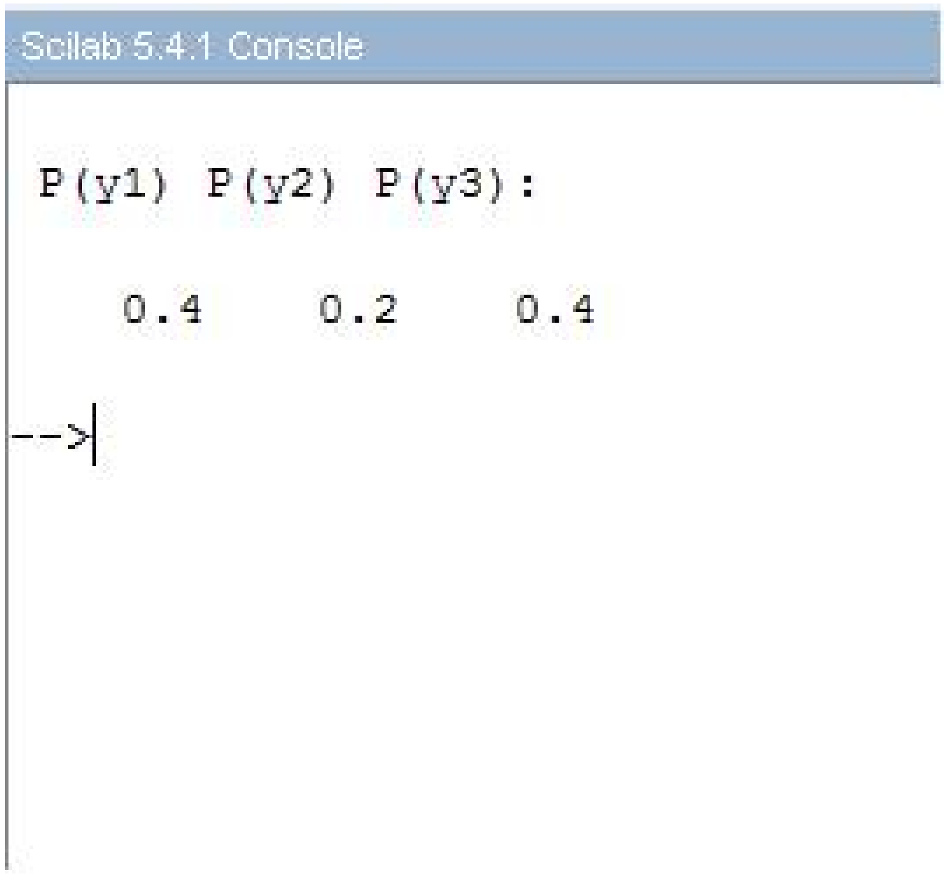


Figure 11.7: Channel

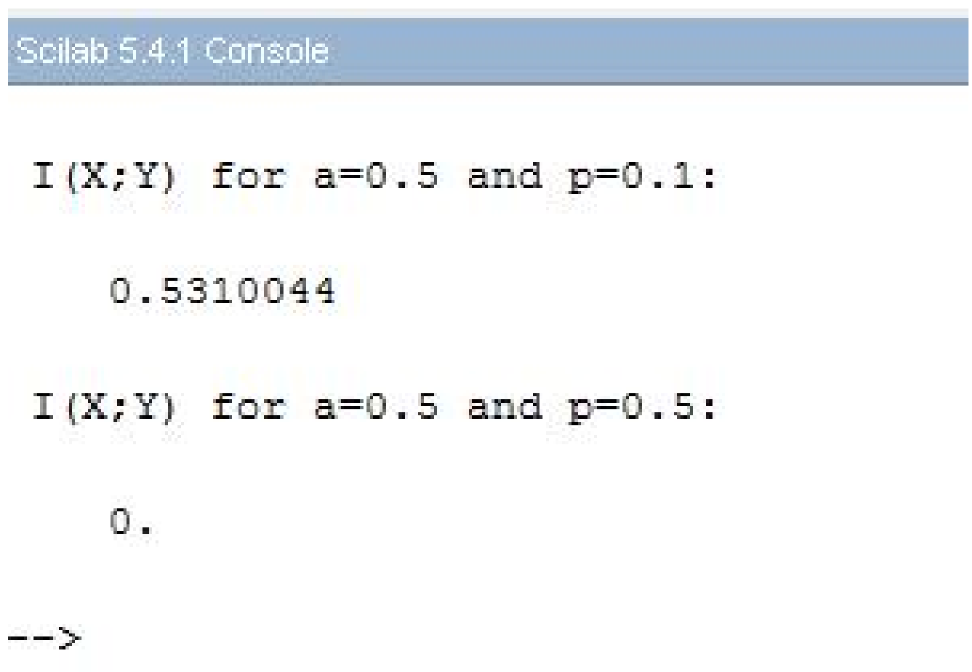


Figure 11.8: BSC

```
//(b)I(X;Y)
//Given
a=0.5;
p=0.1;
//As we know
//P(Y)=P(X)*P(Y/X)
//We have
PX=[a (1-a)];
PYbyX=[(1-p) p;p (1-p)];
PY=PX*PYbyX;

//As H(Y)=-Sum of[P(yi)log2P(yi)]
//Where i=0 to n;
HofY=0;
for i=1:2
    HofY=HofY+(PY(i)*log2(PY(i)));
end

//For BSC, I(X;Y)=H(Y)+plog2(p)+(1-p)log2(1-p)
IXY=-HofY+[(p*log2(p))+((1-p)*log2(1-p))];
disp(IXY,'I(X;Y) for a=0.5 and p=0.1:');


//(c)I1(X;Y)
//Given
a1=0.5;
p1=0.5;
//As we know
//P(Y)=P(X)*P(Y/X)
//We have
PX1=[a1 (1-a1)];
PYbyX1=[(1-p1) p1;p1 (1-p1)];
PY1=PX1*PYbyX1;

//As H(Y)=-Sum of[P(yi)log2P(yi)]
//Where i=0 to n;
HofY1=0;
for i=1:2
```

```
    HofY1=HofY1+(PY1(i)*log2(PY1(i)));
end


//For BSC, I(X;Y)=H(Y)+plog2(p)+(1-p)log2(1-p)
IXY1=-HofY1+(p1*log2(p1))+((1-p1)*log2(1-p1));
disp(IXY1,'I(X;Y) for a=0.5 and p=0.5:');
```

Scilab code Exa 11.20 Differential Entropy

```
//Page Number: 11.24
//Example 11.20
clc;
//Given
//f(x)=1/a for x from 0 to a
//0, otherwise

//We have
//H(X)=-integrate[f(x))*log2f(x)]dx
//Here, f(x)=1/a for limits 0 to a
//H(X)=-integrate(1/a)*log2(1/a)dx for 0 to a
//H(X)=log2(a)

//(a)a1=1
a1=1;
y1=log2(a1);
disp(y1,'For a=1, H(X):');

//(b)a2=2
a2=2;
y2=log2(a2);
disp(y2,'For a=2, H(X):');

```

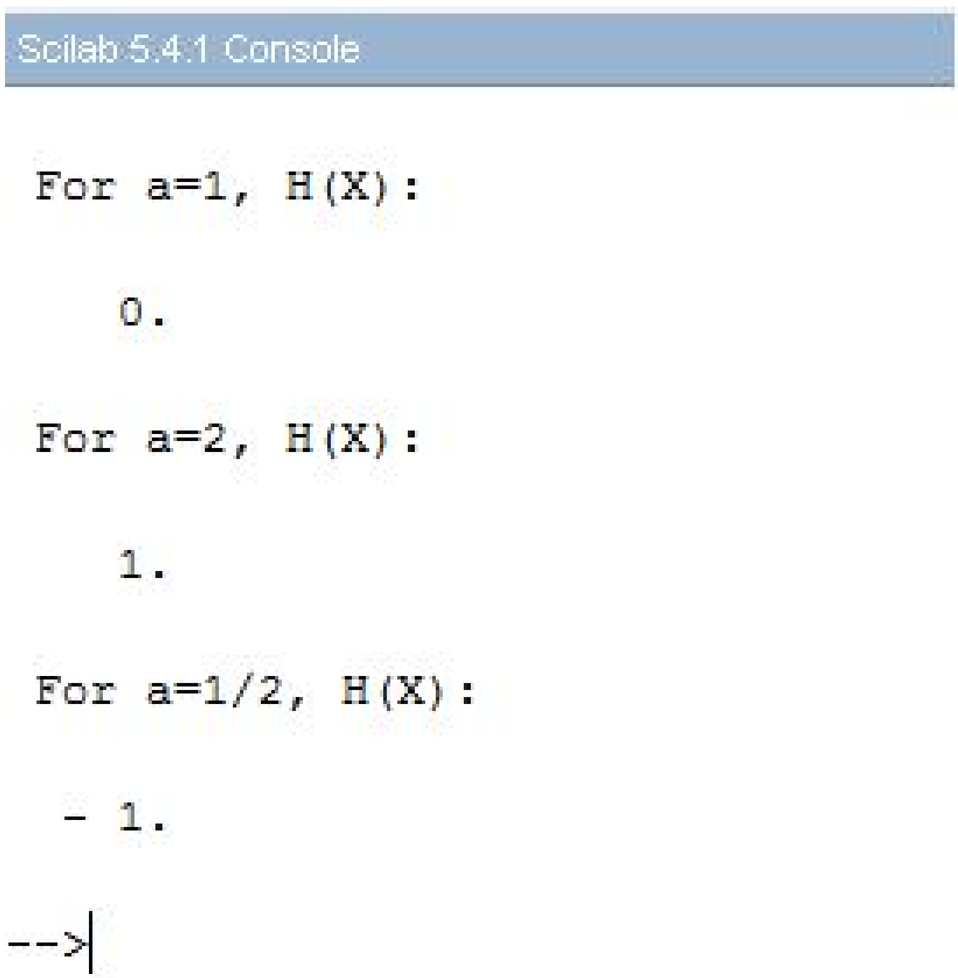


Figure 11.9: Differential Entropy

```
//(c)a3=1/2
a3=1/2;
y3=log2(a3);
disp(y3,'For a=1/2, H(X):');
```

Scilab code Exa 11.23 AWGN Channel

```
//Page Number: 11.26
//Example 11.23
clc;
//Given
B=4D+3; //Hz
S=0.1D-3; //W
n=2*(1D-12); //W/hz

N=n*B;
SN=S/N;
//As Channel Capacity
//C=B*(log2(1+(S/N)));
C=B*(log2(1+(S/N)));
disp('b/s',C,'Channel Capacity');
```

Scilab code Exa 11.24 DMS X with two symbols

```
//Page Number: 11.26
//Example 11.24
clc;

```

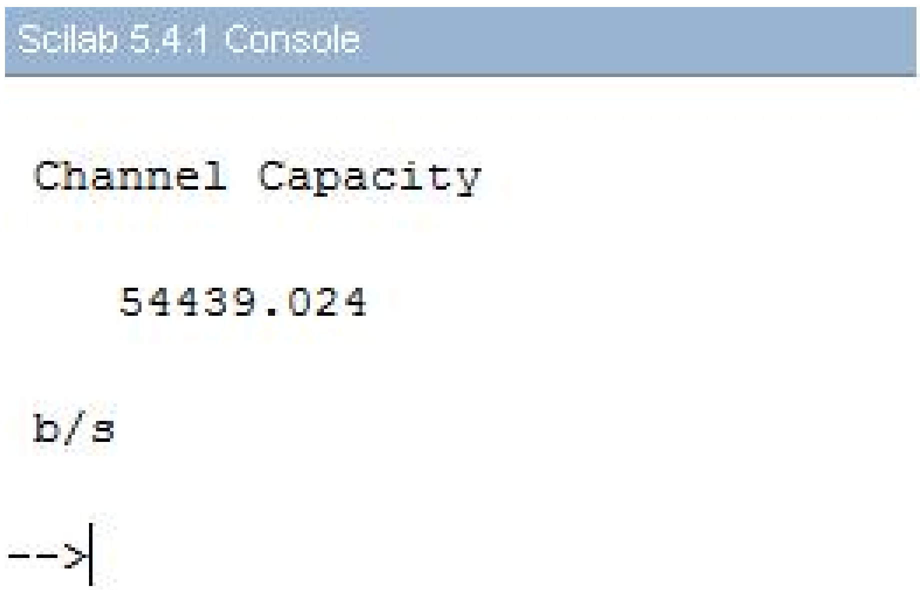


Figure 11.10: AWGN Channel

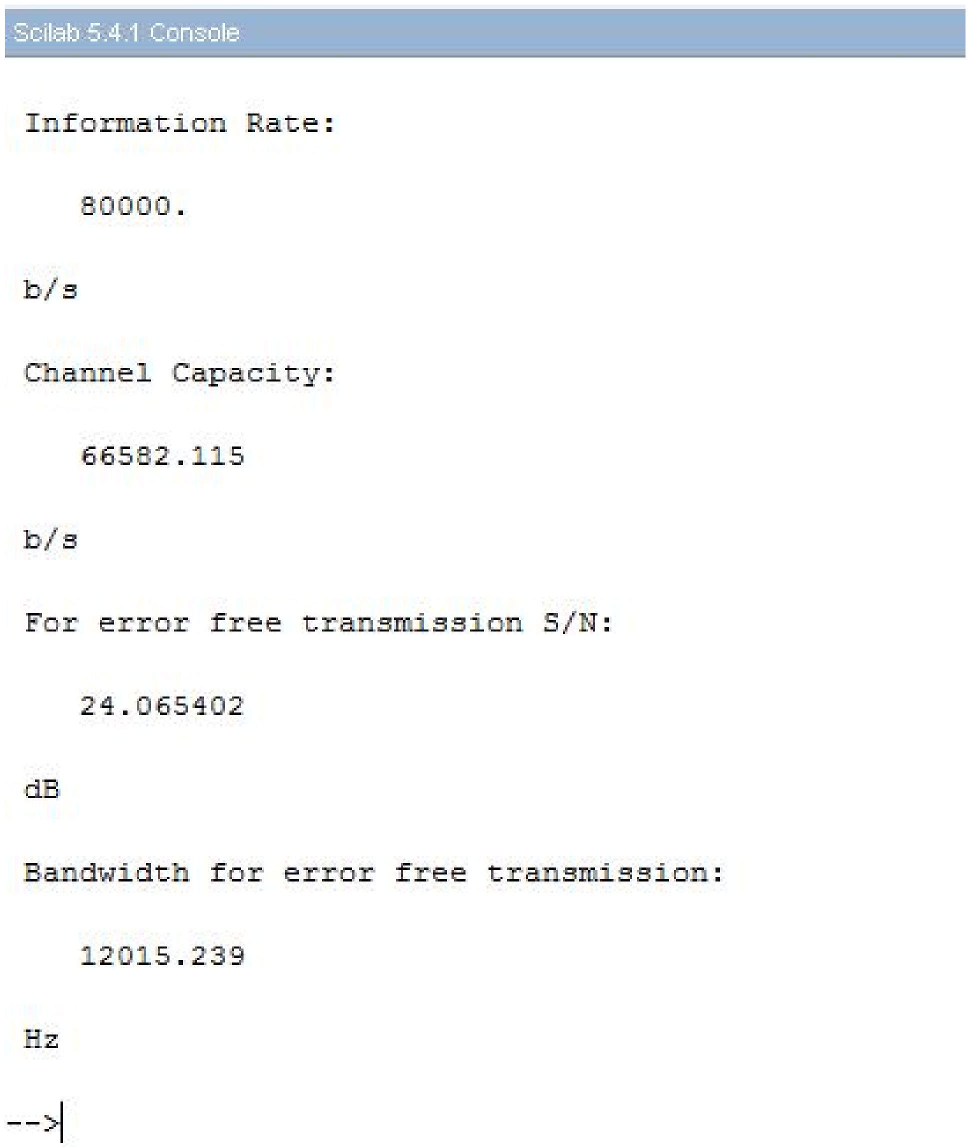


Figure 11.11: DMS X with two symbols

```
//(a) Information Rate
//Given
n=1.25; //times
l=256; //Levels
fM=4D+3; //Hz //Bandwidth
Nr=2*fM; //Nyquist Rate
r=Nr*n;
HofX=log2(l);
//Information rate
R=r*HofX;
disp('b/s',R,'Information Rate:');

//(b)
//As Channel Capacity
//C=B*(log2(1+(S/N)));
B=10^4; //Hz
SNdB=20; //dB
SN=10^(SNdB/10);
C=B*(log2(1+(SN)));
disp('b/s',C,'Channel Capacity:');

//As R>C, error free transmission isnt possible

//(c)For error free transmission
C1=R;
//Therfore S/N
SN1={2^(C1/B)}-1;
SN1dB=10*(log10(SN1));
disp('dB',SN1dB,'For error free transmission S/N:');

//(d)Bandwidth for error free transmission
SN2dB=20; //dB
SN2=10^(SN2dB/10);
//As Channel Capacity
//C=B*(log2(1+(S/N)));
B=C1/(log2(1+(SN2)));
disp('Hz',B,'Bandwidth for error free transmission:'
   );
```

```
//Therefore bandwidth should be greater than or
   equal to B
```

Scilab code Exa 11.25 DMS X

```
//Page Number: 11.27
//Example 11.25
clc;
//Given
p=0.9;
Px=[p (1-p)];

n=1;
//Average Code length
//L=Summation(P(xi)ni)
L=0;
for i=1:2
    L=L+(Px(i)*n);
end

//As H(X)=-Sum of[P(xi)log2P(xi)]
//Where i=0 to n;
HofX=0;
for i=1:2
    HofX=HofX+(Px(i)*log2(Px(i)));
end

//Efficiency=H(X)/L
n=-HofX/L;
np=n*100;
disp('%',np,'Code efficiency:');

//Redundancy
```

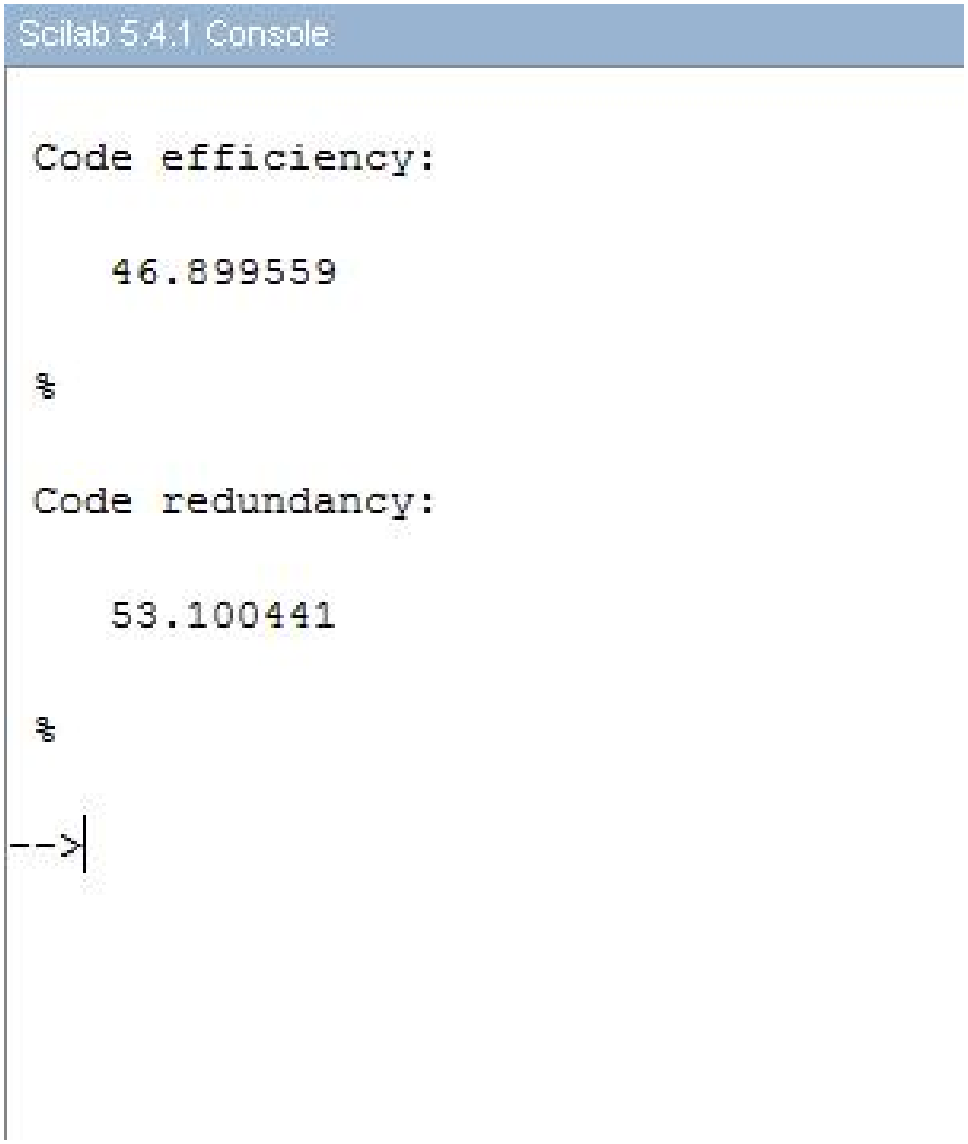


Figure 11.12: DMS X

```
g=1-n;
gp=g*100;
disp('%',gp,'Code redundancy:');
```

Scilab code Exa 11.26 Second order extension of DMS

```
//Page Number: 11.28
//Example 11.26
clc;
//Given
Pa=[0.81 0.09 0.09 0.01];
n=[1 2 3 4];

//Average Code length
//L=Summation(P(xi)ni)
L=0;
for i=1:4
    L=L+(Pa(i)*n(i));
end

//Entropy of second order extension
//As H(X^2)=-Sum of[P(ai)log2P(ai)]
//Where i=0 to n;
HofX2=0;
for i=1:4
    HofX2=HofX2+(Pa(i)*log2(Pa(i)));
end
//b/s

//Efficiency=H(X^2)/L
n=-HofX2/L;
np=n*100;
disp('%',np,'Code efficiency:');
```

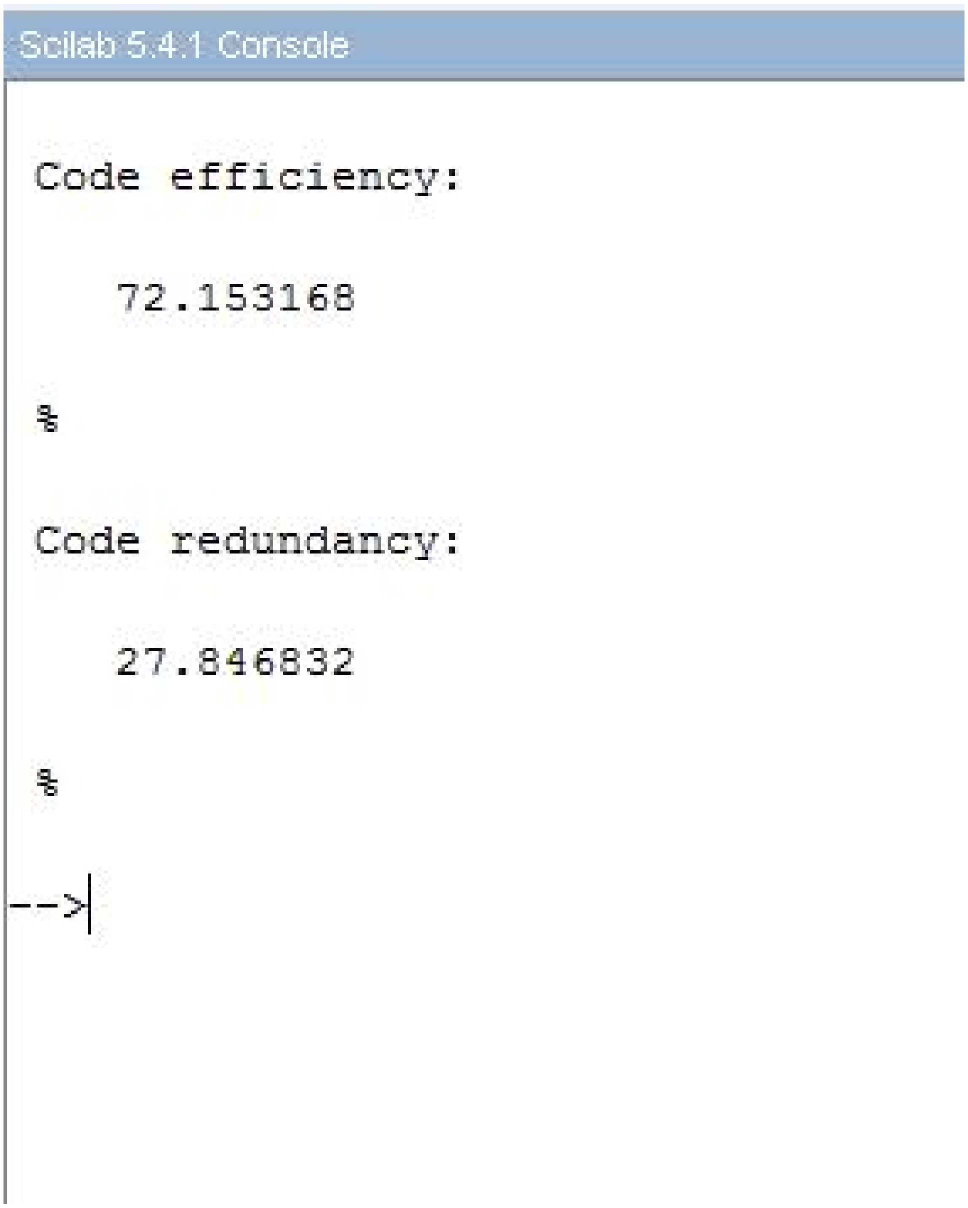


Figure 11.13: Second order extension of DMS

```
//Redundancy
g=1-n;
gp=g*100;
disp('%',gp,'Code redundancy:');
```

Scilab code Exa 11.27 Kraft inequality

```
//Page Number: 11.28
//Example 11.27
clc;
//As Kraft inequlity
//K=summation(2^(-n))
//where i from 0 to 4
//As i=1,2,3,4
//Given

//For Code A
na=[2 2 2 2];
KA=0;
for i=1:4
    KA=KA+(2^(-na(i)));
end
disp(KA,'For Code A:');

//For Code B
nb=[1 2 2 3];
KB=0;
for i=1:4
    KB=KB+(2^(-nb(i)));
end
disp(KB,'For Code B:');

```

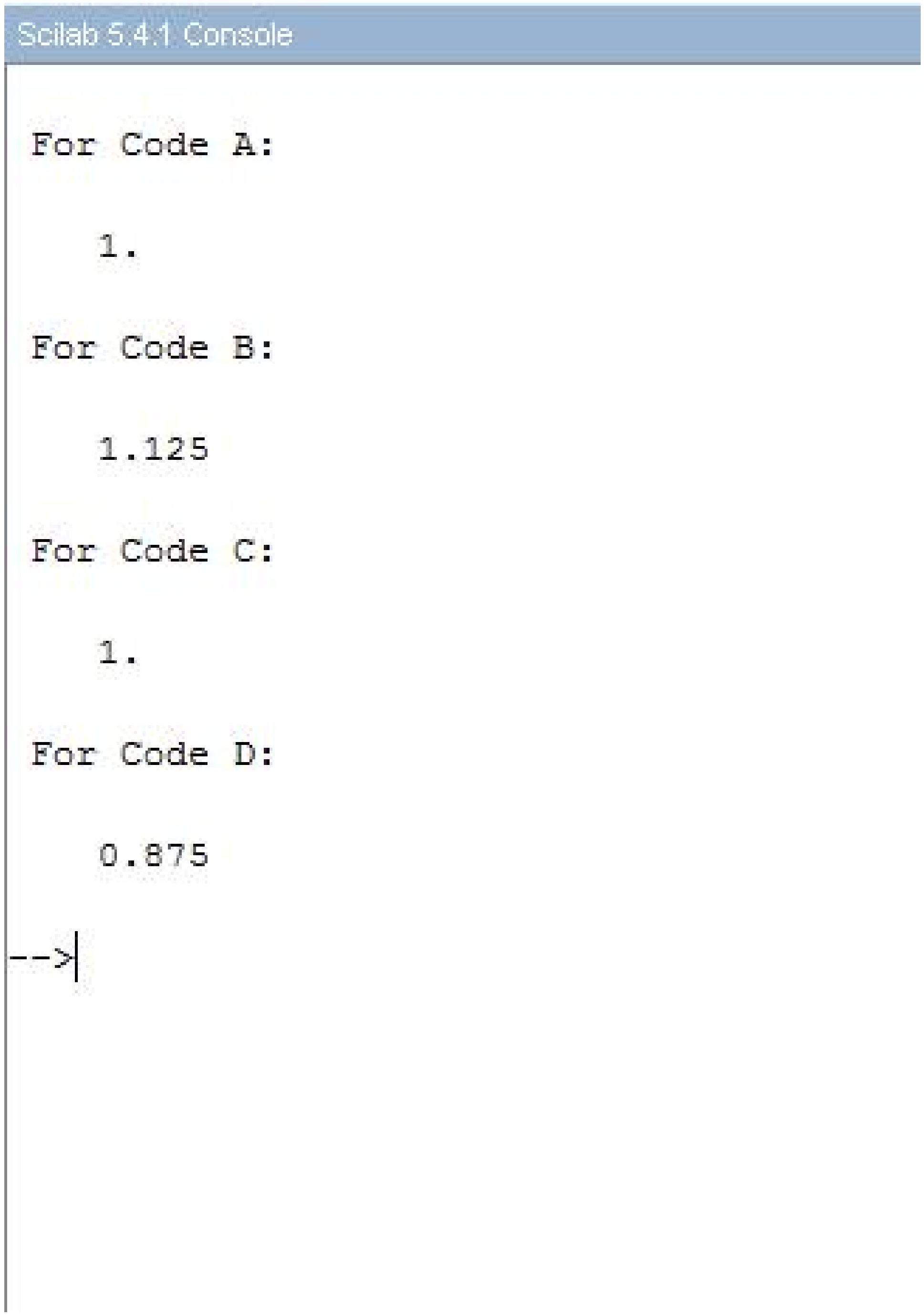


Figure 11.14: Kraft inequality

```
//For Code C
nc=[1 2 3 3];
KC=0;
for i=1:4
    KC=KC+(2^(-nc(i)));
end
disp(KC,'For Code C:');

//For Code D
nd=[1 3 3 3];
KD=0;
for i=1:4
    KD=KD+(2^(-nd(i)));
end
disp(KD,'For Code D:');

//All codes except Code B satisfy Kraft inequality
```

Scilab code Exa 11.32 Shannon Fano

```
//Page Number: 11.31
//Example 11.32
clc;
//Given
Px=[1/2 1/4 1/8 1/8];

//As I(xi)=-log2(Pxi)
for i=1:4
Ix(i)=-log2(Px(i));
n(i)=Ix(i);
end

//As H(X)=-Sum of[P(xi)log2P(xi)]
```

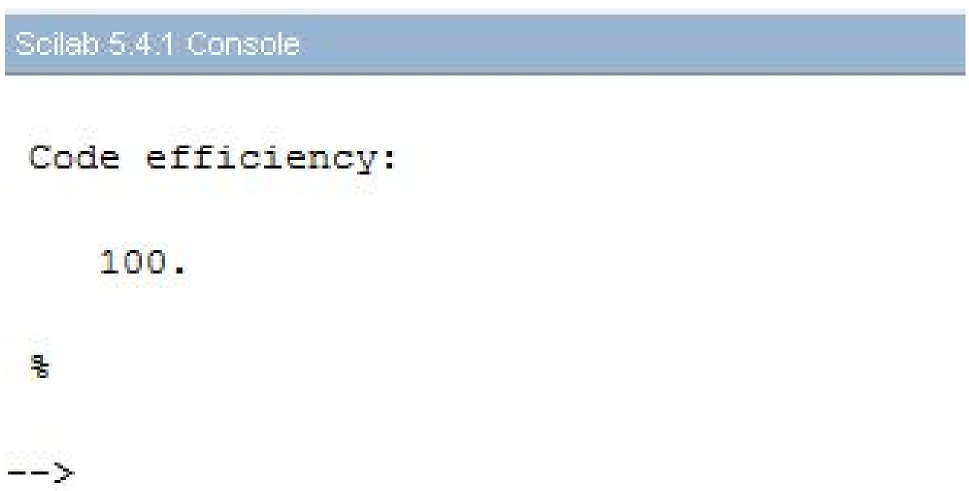


Figure 11.15: Shannon Fano

```
//and I(xi)=-log2p(xi)
//Where i=0 to n;
HofX=0;
for i=1:4
    HofX=HofX+(Px(i)*Ix(i));
end

//Average Code length
//L=Summation(P(xi)ni)
L=0;
for i=1:4
    L=L+(Px(i)*n(i));
end

//Efficiency=H(X)/L
n=HofX/L;
np=n*100;
disp('%',np,'Code efficiency:');

//Hence, efficiency is 100%
```

Scilab code Exa 11.33 Shannon Fano and Huffman

```
//Page Number: 11.32
//Example 11.33
clc;
//Given
//(a)Efficiency of code
Px=[0.2 0.2 0.2 0.2 0.2];
na=[2 2 2 3 3];

//As H(X)=-Sum of[P(ai)log2P(ai)]
//Where i=0 to n;
```

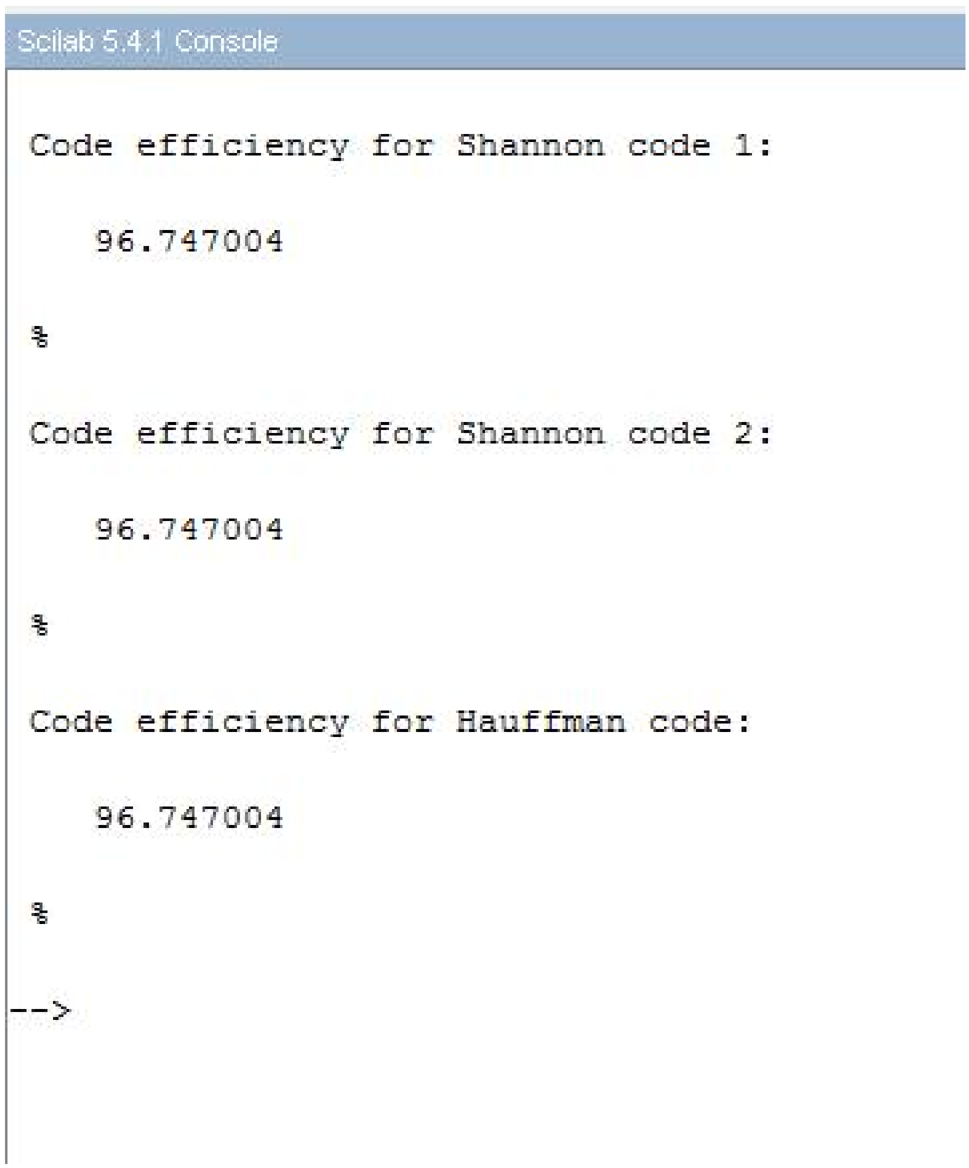


Figure 11.16: Shannon Fano and Huffman

```
HofX=0;
for i=1:5
    HofX=HofX+(Px(i)*log2(Px(i)));
end

//Average Code length
//L=Summation(P(xi)ni)
La=0;
for i=1:5
    La=La+(Px(i)*na(i));
end

//Efficiency=H(X)/L
ea=-HofX/La;
npa=ea*100;
disp('%',npa,'Code efficiency for Shannon code 1:');

//(b) Another Shannon Fano Code
nb=[2 3 3 2 2];

//Average Code length
//L=Summation(P(xi)ni)
Lb=0;
for i=1:5
    Lb=Lb+(Px(i)*nb(i));
end

//Efficiency=H(X)/L
eb=-HofX/Lb;
npb=eb*100;
disp('%',npb,'Code efficiency for Shannon code 2:');

//(c) Hauffman Code
nc=[2 3 3 2 2];

//Average Code length
//L=Summation(P(xi)ni)
Lc=0;
```

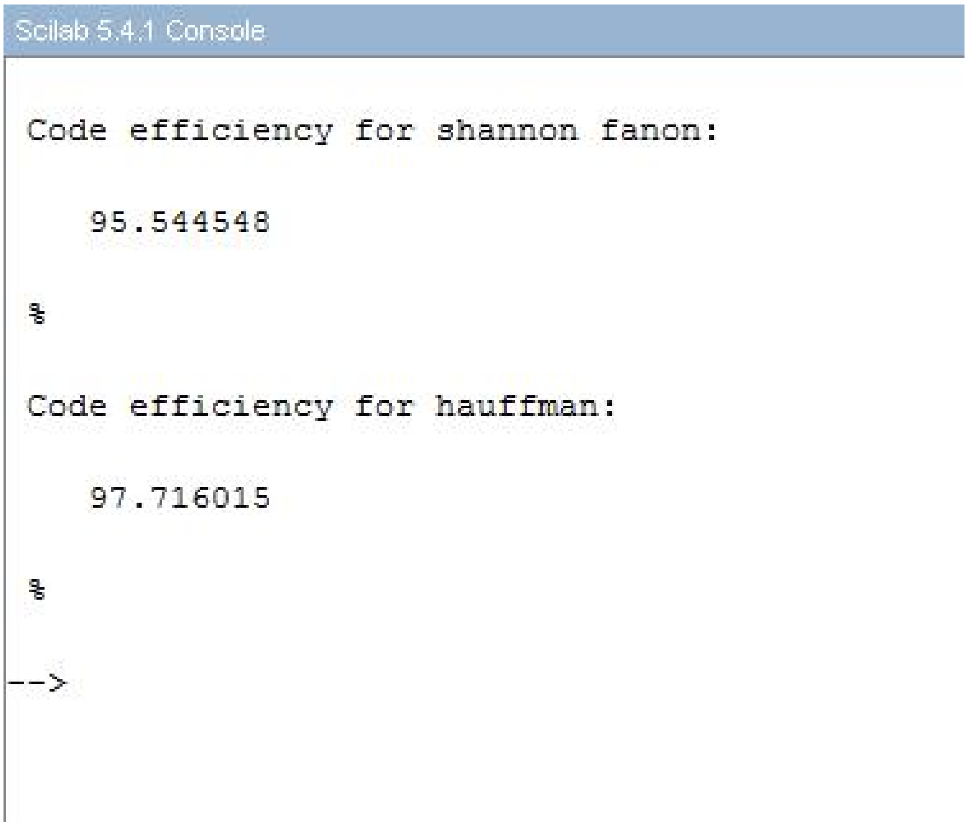


Figure 11.17: DMS X with five symbols

```
for i=1:5
    Lc=Lc+(Px(i)*nc(i));
end

//Efficiency=H(X)/L
ec=-HofX/Lc;
npc=ec*100;
disp('%',npc,'Code efficiency for Hauffman code:');

//Efficiency of all codes is same
```

Scilab code Exa 11.34 DMS X with five symbols

```
//Page Number: 11.33
//Example 11.34
clc;
//Given
//(a) For Shannon Fano Code
Px=[0.4 0.19 0.16 0.15 0.1];
n=[2 2 2 3 3];

//Average Code length
//L=Summation(P(xi)ni)
L=0;
for i=1:5
    L=L+(Px(i)*n(i));
end

//As H(X)=-Sum of[P(xi)log2P(xi)]
//Where i=0 to n;
HofX=0;
for i=1:5
    HofX=HofX+(Px(i)*log2(Px(i)));
end

//Efficiency=H(X)/L
n=-HofX/L;
np=n*100;
disp('%',np,'Code efficiency for shannon fanon:');

//(b) For Huffman Code
nh=[1 3 3 3 3];

//Average Code length
//L=Summation(P(xi)ni)
```

```
Lh=0;
for i=1:5
    Lh=Lh+(Px(i)*nh(i));
end

//Efficiency=H(X)/L
n1=-HofX/Lh;
np1=n1*100;
disp('%',np1,'Code efficiency for hauffman:');
```