

Scilab Textbook Companion for
Analog and Digital Communications
by H. P. Hsu¹

Created by

K Anusha

B.Tech

Electronics Engineering

ABES Engineering College, Ghaziabad

College Teacher

None

Cross-Checked by

K. V. P. Pradeep

July 31, 2019

¹Funded 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	4
1 Signals and Spectra	6
3 Amplitude Modulation	11
4 Angle Modulation	16
5 Pulse Modulation Systems	41
6 Probability and Random Variables	70
8 Noise	82
9 Noise in Analog Communication System	117
10 Digital Modulation and Demodulation	137
11 Information Theory and Source Coding	157

List of Scilab Codes

Exa 1.9	FT of exponential with step function	6
Exa 1.10	Exponential function	7
Exa 1.11	Rectangular Function	8
Exa 3.7	Efficiency	11
Exa 3.8	Sine wave in AM broadcasting	13
Exa 4.3	Instantaneous frequency	16
Exa 4.9	Peak frequency	18
Exa 4.13	Bandwidth	18
Exa 4.14	Modulation Index	21
Exa 4.15	changed fm	23
Exa 4.16	Using FM	25
Exa 4.17	Frequency Modulated	28
Exa 4.18	Carsons rule	28
Exa 4.19	Frequency Multiplier	30
Exa 4.20	FM Transmitter	32
Exa 4.21	FM Modulator	34
Exa 4.22	FM generator	34
Exa 4.23	Multiplication Factor	37
Exa 4.29	Superheterodyne Receiver	39
Exa 5.9	Sampling rate	41
Exa 5.14	Binary Channel	42
Exa 5.15	Analog Signal to PCM System	44
Exa 5.18	Binary PCM System	46
Exa 5.19	CD Recording System	46
Exa 5.22	Audio Signal with Spectral Components	50
Exa 5.25	DM System	52
Exa 5.26	PCM Systems	52
Exa 5.27	DM system	55

Exa 5.32	TDM	55
Exa 5.33	TD Multiplexing	58
Exa 5.34	T1 Carrier System	60
Exa 5.35	TDM	62
Exa 5.37	Telephone Line	63
Exa 5.38	Roll Off Factor	66
Exa 5.39	Telemetry System	67
Exa 5.40	M ary PSK	67
Exa 6.7	Dot and dash	70
Exa 6.14	Binary Communication	73
Exa 6.16	Pulse Positions	74
Exa 6.17	Binary Source	75
Exa 6.20	Noisy Transmission Channel	77
Exa 6.23	Pdf of random variable	78
Exa 6.41	16 Binary Digits Transmission	80
Exa 8.1	Units Conversion	82
Exa 8.2	Rms Voltage	85
Exa 8.3	Output Noise Power	86
Exa 8.4	Noise Figure	87
Exa 8.6	Cascaded Amplifiers	89
Exa 8.7	Noise figure and temperature	92
Exa 8.9	WLAN Reciever	93
Exa 8.10	Receiver Noise Power	95
Exa 8.11	Cascade Amplifiers A and B	97
Exa 8.12	Distortion Component	100
Exa 8.13	Receiver Antenna System	102
Exa 8.14	TV Recievers	104
Exa 8.15	Reciever	107
Exa 8.18	Repeaters	110
Exa 8.23	Noise process	113
Exa 8.24	White Gaussian Noise	113
Exa 9.1	Output Noise Power	117
Exa 9.4	SNR of filters	120
Exa 9.8	AM with envelope detection	120
Exa 9.9	AM with additive thermal noise	122
Exa 9.10	Modulation Systems transmission Bandwidth	123
Exa 9.13	SSB System	127
Exa 9.14	FM broadcast System	128

Exa 9.17	Audio Signal Over RF Channel	130
Exa 9.18	Modulation Index	132
Exa 9.21	Normalized Message Signal	134
Exa 10.1	Binary Reciever	137
Exa 10.2	BPSK Reciever	139
Exa 10.6	Satellite Communication	141
Exa 10.11	Baseband Binary Transmission	141
Exa 10.17	On Off Binary System	144
Exa 10.18	Binary System	146
Exa 10.19	Binary Data Transmission	147
Exa 10.20	Binary Communication	149
Exa 10.24	Voice Transmission BPSK	149
Exa 10.25	Coheret Matched Reciever	153
Exa 10.27	M level PSK	154
Exa 11.2	DMS X with four symbols	157
Exa 11.3	Binary Memoryless Source	159
Exa 11.5	Black and white TV picture	159
Exa 11.6	Telegraph Source	162
Exa 11.7	Binary Channel	164
Exa 11.8	Binary Channel in cascade	166
Exa 11.9	Channel	168
Exa 11.16	BSC	168
Exa 11.20	Differential Entropy	172
Exa 11.23	AWGN Channel	174
Exa 11.24	DMS X with two symbols	174
Exa 11.25	DMS X	178
Exa 11.26	Second order extension of DMS	180
Exa 11.27	Kraft inequality	182
Exa 11.32	Shannon Fano	184
Exa 11.33	Shannon Fano and Huffman	186
Exa 11.34	DMS X with five symbols	190

List of Figures

1.1	FT of exponential with step function	6
1.2	Exponential function	7
1.3	Rectangular Function	8
3.1	Efficiency	12
3.2	Sine wave in AM broadcasting	14
4.1	Instantaneous frequency	17
4.2	Peak frequency	19
4.3	Bandwidth	20
4.4	Modulation Index	22
4.5	changed fm	24
4.6	Using FM	26
4.7	Frequency Modulated	27
4.8	Carsons rule	29
4.9	Frequency Multiplier	31
4.10	FM Transmitter	33
4.11	FM Modulator	35
4.12	FM generator	36
4.13	Multiplication Factor	38
4.14	Superheterodyne Receiver	40
5.1	Sampling rate	42
5.2	Binary Channel	43

5.3	Analog Signal to PCM System	45
5.4	Binary PCM System	47
5.5	CD Recording System	48
5.6	Audio Signal with Spectral Components	51
5.7	DM System	53
5.8	PCM Systems	54
5.9	DM system	56
5.10	TDM	57
5.11	TD Multiplexing	59
5.12	T1 Carrier System	61
5.13	TDM	62
5.14	Telephone Line	64
5.15	Roll Off Factor	65
5.16	Telemetry System	66
5.17	M ary PSK	68
6.1	Dot and dash	71
6.2	Binary Communication	72
6.3	Pulse Positions	74
6.4	Binary Source	76
6.5	Noisy Transmission Channel	77
6.6	Pdf of random variable	79
6.7	16 Binary Digits Transmission	80
8.1	Units Conversion	83
8.2	Rms Voltage	86
8.3	Output Noise Power	87
8.4	Noise Figure	88
8.5	Cascaded Amplifiers	90
8.6	Noise figure and temperature	92
8.7	WLAN Reciever	94
8.8	Receiver Noise Power	96
8.9	Cascade Amplifiers A and B	98
8.10	Distortion Component	101
8.11	Receiver Antenna System	103
8.12	TV Recievers	105
8.13	Reciever	108
8.14	Repeaters	111

8.15	Noise process	114
8.16	White Gaussian Noise	115
9.1	Output Noise Power	118
9.2	SNR of filters	119
9.3	AM with envelope detection	121
9.4	AM with additive thermal noise	122
9.5	Modulation Systems transmission Bandwidth	124
9.6	SSB System	126
9.7	FM broadcast System	129
9.8	Audio Signal Over RF Channel	131
9.9	Modulation Index	133
9.10	Normalized Message Signal	135
10.1	Binary Reciever	138
10.2	BPSK Reciever	140
10.3	Satellite Communication	142
10.4	Baseband Binary Transmission	143
10.5	On Off Binary System	145
10.6	Binary System	146
10.7	Binary Data Transmission	148
10.8	Binary Communication	150
10.9	Voice Transmission BPSK	151
10.10	Coheret Matched Reciever	152
10.11	M level PSK	155
11.1	DMS X with four symbols	158
11.2	Binary Memoryless Source	160
11.3	Black and white TV picture	161
11.4	Telegraph Source	163
11.5	Binary Channel	165
11.6	Binary Channel in cascade	167
11.7	Channel	169
11.8	BSC	170
11.9	Differential Entropy	173
11.10	AWGN Channel	175
11.11	DMS X with two symbols	176
11.12	DMS X	179

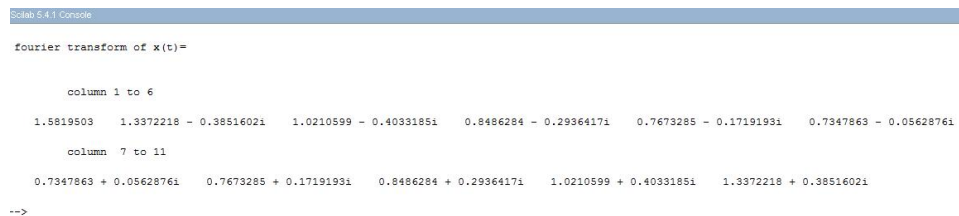
11.13	Second order extension of DMS	181
11.14	Kraft inequality	183
11.15	Shannon Fano	185
11.16	Shannon Fano and Huffman	187
11.17	DMS X with five symbols	189

Chapter 1

Signals and Spectra

Scilab code Exa 1.9 FT of exponential with step function

```
1 //Page Number: 1.13
2 //Example 1.9
3 clc;
4 //Given ,
5 //Signal is  $x(t) = e^{-at} * u(t)$ 
6 //unity function  $u(t) = 1$  for 0 to infinity
7 //therefore
8 x=1;
9 //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

```

Scilab 5.4.1 Console
fourier transform of x(t)=

column 1 to 6
34846.356 20193.201 + 23060.754i 1262.9661 + 24147.945i - 9061.3967 + 17581.253i - 13929.237 + 10293.347i - 15877.711 + 3370.117i
column 7 to 11
- 15877.711 - 3370.117i - 13929.237 - 10293.347i - 9061.3967 - 17581.253i 1262.9661 - 24147.945i 20193.201 - 23060.754i
-->

```

Figure 1.2: Exponential function

```

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

```

Scilab code Exa 1.10 Exponential function

```

1 //Page Number: 1.14
2 //Example 1.10
3 clc;
4 //Given,
5 //Signal is  $x(t) = e^{-|a|t} * u(t)$ 
6 //unity function  $u(t)=1$  for 0 to infinity
7 //therefore
8 x=1;
9 //We assume 'infinity' value as 10 and the value of
  'a' is 1
10 t= 0:1:10;
11 a1=1; // For a >0
12 a2=-1; //For a <0
13 z=((%e)^(a2*t) * x)+((%e)^(a1*t) * x);
14 y=fft(z);

```

```
Scilab 5.4.1 Console

Fourier transform of x(t)=

  1.

Fourier transform of x(t)=

  1.

Fourier transform of x(t)=

  1.

-->
```

Figure 1.3: Rectangular Function

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

Scilab code Exa 1.11 Rectangular Function

```
1 //Page Number: 1.14
2 //Example 1.11
3 clc;
4
5 //(a)
```

```

6
7 //Given
8 //Signal is  $x(t) = \text{rect}(t)$ 
9 // $\text{rect}(t) = 1$  for  $-a < |t| < a$  and 0 elsewhere
10 //Therefore
11 //We find out fourier transform of  $x(t) = 1$  for  $-a < |$ 
     $t| < a$  thus ,
12  $x=1$ ;
13  $a= 200$ ; //Assume
14  $t= -a : 1 : a$ ; //range for fourier transform
15  $y=\text{fft}(x)$ ;
16  $\text{disp}(y, \text{'Fourier transform of } x(t)=\text{'})$ ;
17
18
19 //(b)
20
21 //Given
22 //Signal is  $x(t) = \text{rect}(t)$ 
23 // $\text{rect}(t) = 1$  for  $-a/4 < |t| < a/4$  and 0 elsewhere
24 //Therefore
25 //We find out fourier transform of  $x(t) = 1$  for  $-a/4 <$ 
     $|t| < a/4$  thus ,
26  $x=1$ ;
27  $a= 200$ ; //Assume
28  $t= -a/4 : 1 : a/4$ ; //range for fourer transform
29  $y=\text{fft}(x)$ ;
30  $\text{disp}(y, \text{'Fourier transform of } x(t)=\text{'})$ ;
31
32 //(c)
33
34 //Given
35 //Signal is  $x(t) = \text{rect}(t)$ 
36 // $\text{rect}(t) = 1$  for  $b < |t| < b + a/2$  and 0 elsewhere
37 //Therefore
38 //We find out fourier transform of  $x(t) = 1$  for  $b < |$ 
     $t| < b + a/2$  thus ,
39  $x=1$ ;
40  $a= 200$ ; //Assume

```

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

Chapter 3

Amplitude Modulation

Scilab code Exa 3.7 Efficiency

```
1 //Page Number: 3.14
2 //Example 3.7
3 clc;
4 //Given
5
6 //(a) efficiency = ((u^2)/(2+(u^2)))
7 u=0.5;
8 n=((u^2)/(2+(u^2)));
9 np=n*100;
10 disp("%", np, "Efficiency: ");
11
12 //(b) nmax,
13 //nmax occurs at u=1;
14 u1=1;
15 nmax=((u1^2)/(2+(u1^2)));
16 nmaxp=nmax*100;
17 disp("%", nmaxp, "Efficiency max: ");
```

Scilab 5.4.1 Console

Efficiney:

11.111111

⌘

Efficiney max:

33.333333

⌘

-->|

Figure 3.1: Efficiency

Scilab code Exa 3.8 Sine wave in AM broadcasting

```
1 //Page Number: 3.14
2 //Example 3.8
3 clc;
4 //Given
5 //From waveform
6 Amax=150;
7 Amin=30;
8
9 //(a) Modulation index
10 u=((Amax-Amin)/(Amax+Amin));
11 disp(u,"Modulation Index: ");
12
13 Ac=(Amax/(1+u));
14 //(b)
15 //Carrier Power
16 Pc=(Ac^2)/2;
17 disp("W",Pc,"Carrier Power: ");
18
19 //Side band Power
20 PSB=(Amin^2)/2;
21 disp("W",PSB,"USB=LSB Power: ");
22
23 //Total Average power
24 Pt=Pc+(2*PSB);
25 disp("W",Pt,"Total Average Power: ");
26
27 //(c) Peak Envelope Power
28 //Given
29 R=60; //Ohm
30 PEP=(Amax^2)/(2*R);
```

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 efficiency:

0.0909091

A=

210.

A=

- 15.

-->

```
31 disp("W",PEP,"Peak Envelope Power: ");
32
33 //(d) Modulation Efficiency
34 n=PSB/Pt;
35 disp(n,"Modulation efficiency: ");
36
37 //(e) Given
38 //(i) u=0.2
39 u1=0.2;
40 A1=(60/u1)-Ac;
41 disp(A1, "A=");
42
43 //(ii) u=0.8
44 u2=0.8;
45 A2=(60/u2)-Ac;
46 disp(A2, "A=");
```

Chapter 4

Angle Modulation

Scilab code Exa 4.3 Instantaneous frequency

```
1 //Page Number: 4.9
2 //Example 4.3
3 clc;
4 //Given
5 fc=1D+6; //Hz
6 kf=5;
7 mt=1D+5; //Hz
8
9 //(a) mi(t) with fm
10 mi=(fc+(kf*mt));
11 disp("Hz",mi,"Max, Inst. Frequency with FM");
12
13 kp=3;
14 //(b) mi2(t) with pm
15 mi2=fc+(mt*(kp/(2*%pi)));
16
17 disp("Hz",mi2,"Max, Inst. Frequency with PM");
```

```
Scilab 5.4.1 Console

Max, Inst. Frequency with FM

    1500000.

Hz

Max, Inst. Frequency with PM

    1047746.5

Hz

-->
```

Figure 4.1: Instantaneous frequency

Scilab code Exa 4.9 Peak frequency

```
1 //Page Number: 4.13
2 //Example 4.9
3 clc;
4 //Given
5 delf=20D+3; //hz
6 fm=10D+3; //Hz
7
8 B=delf/fm;
9 disp(B, "Beta: ");
```

Scilab code Exa 4.13 Bandwidth

```
1 //Page Number: 4.16
2 //Example 4.13
3 clc;
4 //Given
5 //x(t)=10cos((2*pi*10^8*t)+(200cos(2*pi*10^3*t)))
6 //on differentiating
7 //wi=2*pi*(1D+8)-4*pi*sin(2*pi*(1D+3)*t)
8 //Therefore
9 delw=4*%pi*(1D+5);
10 wm=2*%pi*(1D+3);
11
12 B=delw/wm;
13 wb=2*(B+1)*wm;
14 fb=wb/2*%pi;
```



```
Scilab 5.4.1 Console  
  
Beta :  
  
    2.  
  
->
```

Figure 4.2: Peak frequency

Scilab 5.4.1 Console

Wb

2525840.5

rad/s

Fb

3967581.

Hz

-->|

Figure 4.3: Bandwidth

```
15
16 disp(" rad/s",wb, "Wb");
17
18 disp(" Hz",fb, "Fb");
```

Scilab code Exa 4.14 Modulation Index

```
1 //Page Number: 4.17
2 //Example 4.14
3 clc;
4
5 //Given
6 delf=100D+3; //Hz
7 fc=20D+6; //Hz
8
9 //As B=delf/fm;
10 //(a) fm1=1D+3hz
11 disp('Part a')
12 fm1=1D+3; //Hz
13 B1=delf/fm1;
14 disp(B1,'Modulation Index');
15
16 fb1=2*delf;
17 disp('Hz',fb1,'Bandwidth');
18
19 //(b) fm2=100D+3hz
20 disp('Part b')
21 fm2=100D+3; //Hz
22 B2=delf/fm2;
23 disp(B2,'Modulation Index');
24
25 fb2=2*(B2+1)*fm2;
26 disp('Hz',fb2,'Bandwidth');
```

Part a

Modulation Index

100.

Bandwidth

200000.

Hz

Part b

Modulation Index

1.

Bandwidth

400000.

Hz

Part c

Modulation Index

0.2

Bandwidth

1000000.

```

27
28 // (c) fm3=500D+3hz
29 disp('Part c')
30 fm3=500D+3; //Hz
31 B3=delf/fm3;
32 disp(B3,'Modulation Index');
33
34 fb3=2*fm3;
35 disp('Hz',fb3,'Bandwidth');

```

Scilab code Exa 4.15 changed fm

```

1 //Page Number: 4.17
2 //Example 4.15
3 clc;
4 //Given
5 //x(t)=10cos(wct+3sinwmt)
6 //Comparing with standard equation
7 B=3;
8 fm=1D+3; //hz
9 fb=2*(B+1)*fm;
10
11 //(a)fm is doubled
12 fma=2*fm;
13 fba=2*(B+1)*fma;
14 disp(fba,"fb with 2fm: ");
15
16
17
18 //(b)fm is one halved
19 fmb=fm/2;
20 fbb=2*(B+1)*fmb;
21 disp(fbb,"fb with 0.5fm: ");

```

Scilab 5.4.1 Console

```
fb with 2fm:
```

```
16000.
```

```
fb with 0.5fm:
```

```
4000.
```

```
-->
```

Figure 4.5: changed fm

Scilab code Exa 4.16 Using FM

```
1 //Page Number: 4.18
2 //Example 4.15
3 clc;
4 //Given
5 //x(t)=10cos(wct+3sinwmt)
6 //Comparing with standard equation of fm
7 B=3;
8 fm=10+3; //hz
9 fb=2*(B+1)*fm;
10
11 //B is inversely proportional to fm
12
13 //(a)fm is doubled
14 Ba=B/2;
15 fma=2*fm;;
16 fba=2*(Ba+1)*fma;
17 disp(fba,"fb with 2fm: ");
18
19
20
21 //(b)fm is one halved
22 Bb=2*B;
23 fmb=fm/2;
24 fbb=2*(Bb+1)*fmb;
25 disp(fbb,"fb with 0.5fm: ");
```

Scilab 5.4.1 Console

```
fb with 2fm:
```

```
10000.
```

```
fb with 0.5fm:
```

```
7000.
```

```
-->
```

Figure 4.6: Using FM

Scilab 5.4.1 Console

Bandwidth

14000.

Hz

Maximum frequency deviation

15000.

Hz

Bandwidth

32000.

Hz

-->

Figure 4.7: Frequency Modulated

Scilab code Exa 4.17 Frequency Modulated

```
1 //Page Number: 4.18
2 //Example 4.17
3 clc;
4 //Given
5 fm=2D+3; //Hz
6 delf=5D+3; //Hz
7
8 //(a) Bandwidth of modulated signal
9 B=delf/fm;
10
11 fb=2*(B+1)*fm;
12 disp('Hz',fb,'Bandwidth');
13
14 //(b)Max. frequency deviation and Bandwidth of new
    signal
15 //Given
16 fm1=fm-(1D+3);
17 delf1=3*delf;
18
19 B1=delf1/fm1;
20
21 fd=B1*fm1;
22 disp('Hz',fd,'Maximum frequency deviation');
23
24 fb1=2*(B1+1)*fm1;
25 disp('Hz',fb1,'Bandwidth');
```

Scilab code Exa 4.18 Carsons rule

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

Scilab 5.4.1 Console

```
BW using formula
```

```
210000.
```

```
Hz
```

```
BW using Carsons Rule
```

```
180000.
```

```
Hz
```

```
-->
```

Figure 4.8: Carsons rule

```

3  clc;
4  //Given
5  delf=75D+3; //Hz
6  fM=15D+3; //Hz
7
8  D=delf/fM;
9  //Given formula fb=2(D+2)*fM
10 fb1=2*(D+2)*fM;
11 disp('Hz',fb1,'BW uing formula');
12
13 //Carsons Rule
14 fb2=2*(D+1)*fM;
15 disp('Hz',fb2,'BW uing Carsons Rule');
16
17 //High quality Fm radios require minimum 200kHz
18 //Therefore, carsons rule underestimates bandwidth

```

Scilab code Exa 4.19 Frequency Multiplier

```

1  //Page Number: 4.19
2  //Example 4.19
3  clc;
4  //Given
5  fm1=50; //Hz
6  fm2=15D+3; //Hz
7
8  delf=75D+3; //Hz
9
10 //As B=delf/fm
11 Bmin=delf/fm2;
12 Bmax=delf/fm1;
13
14 //Let B1=0.5

```

```
Scilab 5.4.1 Console

Multiplication factor

    3000.

Max allowed frequency deviation

    25.

Hz

-->
```

Figure 4.9: Frequency Multiplier

```

15 B1=0.5;
16 n=(Bmax/B1);
17 disp(n, 'Multiplication factor ');
18
19 delf1=(delf/n);
20 disp('Hz',delf1, 'Max allowed frequency deviation ');

```

Scilab code Exa 4.20 FM Transmitter

```

1 //Page Number: 4.20
2 //Example 4.20
3 clc;
4 //Given
5 f1=2D+5; //Hz
6 fL0=10.8D+6; //Hz
7 delf1=25; //Hz
8 n1=64;
9 n2=48;
10
11 delf=(delf1*n1*n2);
12 disp('Hz',delf, 'Maximum frequency deviation ');
13
14 f2=n1*f1;
15
16 f3a=f2+fL0;
17 f3b=f2-fL0;
18
19 //For f3a
20 fca=n2*f3a;
21 disp('Hz',fca, 'Carrier frequency 1 ');
22
23 //For f3b
24 fcb=n2*f3b;

```

Scilab 5.4.1 Console

```
Maximum frequency deviation
```

```
76800.
```

```
Hz
```

```
Carrier frequency 1
```

```
1.133D+09
```

```
Hz
```

```
Carrier frequency 2
```

```
96000000.
```

```
Hz
```

```
-->|
```

Figure 4.10: FM Transmitter

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

Scilab code Exa 4.21 FM Modulator

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

Scilab code Exa 4.22 FM generator

```
1 //Page Number: 4.20
2 //Example 4.22
3 clc;
4 //Given
```


Scilab 5.4.1 Console

```
Output of first multiplier:
```

```
12800000.
```

```
Hz
```

```
fLO
```

```
9800000.
```

```
Hz
```

```
-->|
```

Figure 4.11: FM Modulator

Scilab 5.4.1 Console

```
fLO
```

```
99280000.
```

```
Hz
```

```
n1 with n2=150:
```

```
50.
```

```
-->
```

Figure 4.12: FM generator

```

5 B=0.2;
6 f1=200D+3; //Hz
7 fml=50; //Hz
8 fmh=15D+3; //Hz
9 delf=75D+3; //hz
10 fc=108D+6; //Hz
11
12 delf1=B*fml;
13 n1n2=delf/delf1;
14
15 //Let n2=150
16 n2=150;
17 flo=((delf*f1)-fc)/n2;
18 disp('Hz',flo,'fLO');
19
20 n1=n1n2/n2;
21 disp(n1, "n1 with n2=150:")

```

Scilab code Exa 4.23 Multiplication Factor

```

1 //Page Number: 4.21
2 //Example 4.23
3 clc;
4 //Given ,
5
6 delfd1=50; //Hz
7 f1=120; //Hz
8
9 delfd2=20000; //Hz
10 f2=240; //Hz
11 //(a)PM
12 delf1=(f2/f1)*delfd1;
13 n1=delfd2/delf1;

```

```
Scilab 5.4.1 Console

Frequency multiplication factor in PM

200.

Frequency multiplication factor in FM

400.

-->
```

Figure 4.13: Multiplication Factor

```
14 disp(n1, 'Frequency multiplication factor in PM');
15
16 //(b)FM
17 n2=delfd2/delfd1;
18 disp(n2, 'Frequency multiplication factor in FM');
```

Scilab code Exa 4.29 Superheterodyne Receiver

```
1 //Page Number: 4.25
2 //Example 4.29
3 clc;
4 //Given ,
5 f1=108; //MHz
6 f2=157; //MHz
7
8 //(a) Image frequency overlaps RF band
9 fIF=12; //MHz
10
11 fL01=f1-fIF;
12 disp('MHz',fL01,'fL01');
13 fim1=fL01-fIF;
14 disp('MHz',fim1,'fim1');
15
16 fL02=f2-fIF;
17 disp('MHz',fL02,'fL02');
18 fim2=fL02-fIF;
19 disp('MHz',fim2,'fim2');
20
21 //Clearly image and RF band overlap
```

fL01

96.

MHz

fim1

84.

MHz

fL02

145.

MHz

fim2

133.

MHz

Chapter 5

Pulse Modulation Systems

Scilab code Exa 5.9 Sampling rate

```
1 //Page Number: 5.26
2 //Example 5.9
3 clc;
4 //Given ,
5 //m(t)=10cos(2000*pi*t)cos(8000*pi*t)
6 //or 5 cos (6000*pi*t) +5*cos(10000*pi*t)
7 //(a) Minimum sampling rate
8 //we have
9 fM=5000; //Hz
10 fs=2*fM;
11 disp('Hz',fs,'Minimum sampling rate');
12
13 //(b)bandpass sampling theorem
14 fu=fM;
15 fb=fM-3000; //Hz
16 //As fu/fb is 2.5
17 //We have
18 k=2;
19 fs2=(2*fu)/k;
```

```
Scilab 5.4.1 Console

Minimum sampling rate

    10000.

Hz

Minimum sampling rate by sampling theorem

    5000.

Hz

-->
```

Figure 5.1: Sampling rate

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

Scilab code Exa 5.14 Binary Channel

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



```
Scilab 5.4.1 Console

Binary digits

    5.

Quantizing level

    32.

Sampling Rate

    7200.

Hz

-->
```

Figure 5.2: Binary Channel

```

10
11 nn=round(n)-1;
12 disp(nn,'Binary digits ');
13
14 L=2^nn;
15 disp(L,'Quantizing level ');
16
17 fs=Rb/nn;
18 disp('Hz',fs,'Sampling Rate ');

```

Scilab code Exa 5.15 Analog Signal to PCM System

```

1 //Page Number: 5.31
2 //Example 5.15
3 clc;
4
5 mp=1; //Assume peak amplitude is unity
6 //Given
7 del=0.02*mp;
8
9 L=(mp*2)/del;
10
11 for (i=0:10)
12     j=2^i;
13     if(j>=L)
14         L1=j;
15         break;
16     end
17 end
18 n=log2(L1); // bits per sample
19 disp(n,'Number of bits ');

```

```
Scilab 5.4.1 Console

Number of bits

7.

-->
```

Figure 5.3: Analog Signal to PCM System

Scilab code Exa 5.18 Binary PCM System

```
1 //Page Number: 5.31
2 //Example 5.18
3 clc;
4 //Given ,
5 SbyN=40; //db
6 SbyN0=10^(SbyN/10);
7
8 //As sbyn=3L^2/2
9 L=sqrt((2*(SbyN0))/3);
10 LL=round(L);
11
12 n=(log2(LL));
13
14 nn=(round(n))+1; //Upper limit
15
16 disp(nn, 'Binary digits ');
17 LL=2^nn;
18
19 disp(LL, 'Number of levels ');
20 //As SQN= 1.76+6.02(n)
21 SQN= 1.76+6.02*(nn);
22 disp('dB',SQN, 'Signal to quantizin ratio');
```

Scilab code Exa 5.19 CD Recording System

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

Scilab 5.4.1 Console

Binary digits

7.

Number of levels

128.

Signal to quantizin ratio

43.9

dB

-->|

Figure 5.4: Binary PCM System

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

53

Number of comparable books

50.

```

2 //Example 5.19
3 clc;
4 //Given,
5 n=16;
6 Rb=44100; //b/s
7 //(a) Output signal to quantizing ratio
8 SNQ=1.76+(6.02*n);
9 disp('dB',SNQ,'Output signal to quantizing ratio: ')
   ;
10
11 //(b)Input Bit Rate
12 IBR=2*Rb*n;
13 disp('B/s',IBR,'Input bit rate: ');
14 OBR=2*IBR;
15 disp('B/s',OBR,'Output bit rate: ');
16
17 //(c)Number of bits recorded
18 //Given, an hours time
19 //Therefore, time
20 t=60*60;
21 NBR=OBR*t;
22 disp('Bytes',NBR,'Number of bits recorded: ');
23
24 //(d) Dictionary
25 //Given
26 p=1500;
27 c=2;
28 l=100;
29 w=8;
30 let=6;
31 b=7;
32 d=p*c*l*w*let*b;
33 disp('Bytes',d,'Number of bits required');
34
35 x=NBR/(2*d);
36 y=round(x);
37 disp(y,'Number of comparable books');

```

Scilab code Exa 5.22 Audio Signal with Spectral Components

```
1 //Page Number: 5.35
2 //Example 5.22
3 clc;
4 //Given ,
5 f1=200; //Hz
6 f2=3300; //Hz
7 fs=8000; //Samples/s
8 SQN=30; //dB
9
10 //(a)Minimum number of quantizin levels and bits per
    sample
11 //From  $SQN=1.76+20\log L$ 
12 La=10(((SQN-1.76)/20));
13 L=round(La);
14 disp(L,'Minimum number of quantizing levels');
15 n=log2(L);
16 nn=round(n);
17 disp(nn,'Minimum number of bits per sample');
18
19 //(b)Minimum system bandwidth
20 Fpcm=(nn*fs)/2;
21 disp('Hz',Fpcm,'Minimum system Bandwidth');
22
23 //(c)For u=255
24 //SQN=20logL - 10.1
25 La1=10(((SQN+10.1)/20));
26 L1=(round(La1))+1; //Upper Limit
27 disp(L1,'Minimum number of quantizing levels for u
    =255');
28 n1=log2(L1);
```


Scilab 5.4.1 Console

Minimum number of quantizing levels

26.

Minimum number of bits per sample

5.

Minimum system Bandwidth

20000.

Hz

Minimum number of quantizing levels for u=255

102.

Minimum number of bits per sample

7.

Minimum system Bandwidth

28000.

Hz

-->

Figure 5.6: Audio Signal with Spectral Components

```

29 nn1=(round(n1));
30 disp(nn1, 'Minimum number of bits per sample');
31
32 //Minimum system bandwidth
33 Fpcm1=(nn1*fs)/2;
34 disp('Hz', Fpcm1, 'Minimum system Bandwidth');

```

Scilab code Exa 5.25 DM System

```

1 //Page Number: 5.37
2 //Example 5.25
3 clc;
4 //Given,
5 fs=32D+3; //Hz
6 fm=1000; //Hz
7 fM=4D+3; //Hz
8 //As  $SNR = \frac{3 * (fs^3)}{8 * \pi * \pi * (fm^2) * fM}$ 
9  $SNR = \frac{3 * (fs^3)}{8 * \pi * \pi * (fm^2) * fM}$ ;
10 SNRdb=(log10(SNR));
11 disp('dB', SNRdb, 'Output SNR');

```

Scilab code Exa 5.26 PCM Systems

```

1 //Page Number: 5.38
2 //Example 5.26
3 clc;
4 //Given,
5 n=4;
6 SNQ=1.76+(6.02*n);

```

Scilab 5.4.1 Console

Output SNR

2.4931215

dB

-->

Figure 5.7: DM System

```
Scilab 5.4.1 Console

Output SNR

    25.84

dB

-->
```

Figure 5.8: PCM Systems

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

Scilab code Exa 5.27 DM system

```
1 //Page Number: 5.38
2 //Example 5.27
3 clc;
4 //Given ,
5 bw=3D+3; //Hz
6 n=3;
7 fs=(n*2*bw);
8 del=250D-3; //mV
9 fm=1000; //Hz
10 //(a) Maximum amplitude
11 Amax=(del*fs)/(2*pi*fm);
12 disp('V',Amax,'Maximum Amplitude');
13
14 //(b) Output signal to quantizing ratio
15 SNRO=(3*(fs^3))/(8*pi*pi*(fm^3));
16 SNRdb=10*(log10(SNRO));
17 disp('dB',SNRdb,'Output SNR');
```

Scilab code Exa 5.32 TDM

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

```
Scilab 5.4.1 Console

Maximum Amplitude

    0.7161972

V

Output SNR

    23.45549

dB

-->
```

Figure 5.9: DM system

Scilab 5.4.1 Console

Sampling Rate

7000.

Samples/s

-->

Figure 5.10: TDM

```

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

```

Scilab code Exa 5.33 TD Multiplexing

```

1 //Page Number: 5.40
2 //Example 5.33
3 clc;
4 //Given ,
5 m1=3.6D+3; //Hz
6 m2=1.2D+3; //Hz
7 m3=m2;
8 m4=m2;
9 //(a)Nyquist rate
10 nr1=2*m1;
11 disp(nr1,'Nyquist Rate of m1(t)');
12 nr2=2*m2;
13 disp(nr2,'Nyquist Rate of m2(t)');
14 nr3=2*m3;
15 disp(nr3,'Nyquist Rate of m3(t)');
16 nr4=2*m4;
17 disp(nr4,'Nyquist Rate of m4(t)');
18
19 //(b) Speed of commutator
20 c=nr1+nr2+nr3+nr4;
21 disp('samples/s',c,'Speed of commutator');
22
23 //(c)Output bit rate
24 //Given ,
25 L=1024;
26 n=log2(L);

```


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 Bandwidth⁶⁴

7200.

Hz

```

27 OBR=n*c;
28 disp('b/s',OBR,'Output bit rate');
29
30 //(d)Minimum channel bandwidth
31 fB=c/2;
32 disp('Hz',fB,'Minimum Channel Bandwidth');

```

Scilab code Exa 5.34 T1 Carrier System

```

1 //Page Number: 5.41
2 //Example 5.34
3 clc;
4 //Given ,
5 fs=8000; //Hz
6 m=24;
7 n=8;
8 //(a) Duration of each bit
9 t1=1/fs;
10 t2=(m*n)+1; // Extra bit for synchronization
11 Tb=t1/t2;
12 disp('seconds',Tb,'Duration of each bit');
13
14 //(b) Transmission Rate
15 Rb=1/Tb;
16 disp('b/s',Rb,'Transmission Rate');
17
18 //(c)Minimum transmission bandwidth
19 fT1=1/(2*Tb);
20 disp('Hz',fT1,'Minimum transmission bandwidth');

```

```
Scilab 5.4.1 Console

Duration of each bit

    0.0000006

seconds

Transmission Rate

    1544000.

b/s

Minimum transmission bandwidth

    772000.

Hz

-->
```

Figure 5.12: T1 Carrier System

```
Scilab 5.4.1 Console

Spacing between successive multiplexed pulses

0.000004

Seconds

Spacing between successive multiplexed pulses using Nyquist rate of sampling

0.0000049

Seconds

-->
```

Figure 5.13: TDM

Scilab code Exa 5.35 TDM

```
1 //Page Number: 5.42
2 //Example 5.35
3 clc;
4 //Given ,
5 n=24;
6 f=3.4D+3; //Hz
7 ts=1D-6; //Second
8
9 //(a) Spacing between successive multiplexed pulses
10 fs=8000; //Samples per second
11 t1=1/fs;
12 t2=n+1; //One synchronizing bit
13
14 Tb=t1/t2;
```

```

15 //Actual Tb, as actual duration of each pulse is 1us
16 ATb=Tb-ts;
17 disp('Seconds',ATb,'Spacing between successive
      multiplexed pulses');
18
19
20 //(b) Nyquist Rate of Sampling
21 f1=2*f;
22 T=1/f1; //Seconds
23
24 Tb1=T/t2;
25
26 ATb1=Tb1-ts;
27 disp('Seconds',ATb1,'Spacing between successive
      multiplexed pulses using Nyquist rate of sampling
      ');

```

Scilab code Exa 5.37 Telephone Line

```

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

```

```
Scilab 5.4.1 Console

Data Rate

    5600.

b/s

-->
```

Figure 5.14: Telephone Line

Scilab 5.4.1 Console

```
Roll off factor
```

```
0.5
```

```
-->
```

Figure 5.15: Roll Off Factor

Scilab 5.4.1 Console

```
Minimum Transmission bandwidth
```

```
168000.
```

```
Hz
```

```
-->
```

Figure 5.16: Telemetry System

Scilab code Exa 5.38 Roll Off Factor

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

Scilab code Exa 5.39 Telemetry System

```
1 //Page Number: 5.43
2 //Example 5.39
3 clc;
4 //Given ,
5 m=8;
6 fM=2D+3; //Hz
7 a=0.2;
8 //Here we choose L=128;
9 L=128;
10 n=log2(L);
11 Sr=2*fM;
12 fs=1.25*Sr;
13
14 //For n tdm signals
15 x=m*fs;
16
17 //Resultant bit rate
18 br=7*x;
19
20 //Minimum Transmission bandwidth
21 fB=((1+a)*br)/2;
22 disp('Hz', fB, 'Minimum Transmission bandwidth');
```

Scilab code Exa 5.40 M ary PSK

```
1 //Page Number: 5.44
2 //Example 5.40
3 clc;
4 //Given ,
5 M=16;
6 sr=40000;
```

Bit Rate:

160000.

b/s

Value of M

4.

Band Rate

80000.

Symbols/s

Spectral efficiency

1.5384615

b/s Hz

-->

```

7 a=0.3;
8 //(a) Bit Rate
9 n=log2(M);
10 br=sr*n;
11 disp('b/s',br,'Bit Rate: ');
12
13 //(b)
14 //As  $2*fB=(1+a)*R/\log_2M$ 
15 //Given
16 bw=110D+3; //Hz //= $2*fB$ 
17
18  $M=2^{((1+a)*br)/bw}$ ;
19 MM=round(M);
20 disp(MM,'Value of M');
21
22 //(c) Band Rate
23 band=br/(log2(n));
24 disp('Symbols/s',band,'Band Rate');
25
26 //(d) Spectral efficiency
27  $BT=((1+a)*br)/2$ ;
28 Eff=br/BT;
29 disp('b/s Hz',Eff,'Spectral efficiency');

```

Chapter 6

Probability and Random Variables

Scilab code Exa 6.7 Dot and dash

```
1 //Page Number: 6.16
2 //Example 6.7
3 clc;
4 //Given
5 //Pdot=2*Pdash and Pdot+Pdash=1
6 //Therefore, on solving using linear equations
7 a=[1 -2;1 1];
8 c=[0;1];
9 b=inv(a)*c;
10 Pdash=b(1,1);
11 Pdot=b(2,1);
12 disp(Pdot, 'Pdot: ');
13 disp(Pdash, 'Pdash: ')
```

```
Scilab 5.4.1 Console  
  
Pdot :  
  
    0.3333333  
  
Pdash :  
  
    0.6666667  
  
-->|
```

Figure 6.1: Dot and dash

```
Scilab 5.4.1 Console

P(r0):

    0.55

P(r1):

    0.45

P(m0/r0):

    0.8181818

P(m1/r1):

    0.8888889

Probability error:

    0.15

Probabilty that is transmitted correctly:

    0.85

-->
```

Figure 6.2: Binary Communication

Scilab code Exa 6.14 Binary Communication

```
1 //Page Number: 6.18
2 //Example 6.14
3 clc;
4 //Given
5 p=0.1;
6 q=0.2;
7 Pm0=0.5;
8 Pr1bym0=p;
9 Pr0bym1=q;
10
11 //(a) Find Pr0 and Pr1
12
13 Pm1=1-Pm0;
14 Pr0bym0=1-Pr1bym0;
15 Pr1bym1=1-Pr0bym1;
16
17 //By formula
18 //P(r0)=(P(r0/m0)*P(m0))+P(r0/m1)*P(m1);
19 //P(r1)=(P(r1/m0)*P(m0))+P(r1/m1)*P(m1);
20
21 Pr0=(Pr0bym0*Pm0)+(Pr0bym1*Pm1);
22 Pr1=(Pr1bym0*Pm0)+(Pr1bym1*Pm1);
23 disp(Pr0, 'P(r0): ');
24 disp(Pr1, 'P(r1): ');
25
26 //(b)P(m0/r0)
27 //Using Bayes Rule
28 //P(m0/r0)=(P(m0)*P(r0/m0)/P(r0))
29 Pm0byr0=(Pm0*Pr0bym0)/Pr0;
30 disp(Pm0byr0, 'P(m0/r0): ');
31
32 //(c)P(m1/r1)
```

```

Probability that all pulses are positive:

0.004096

Probability that all pulses are positive ,positive, positive, zero,zero,negative:

0.001728

-->|

```

Figure 6.3: Pulse Positions

```

33 //Using Bayes Rule
34 //P(m1/r1)=(P(m1)*P(r1/m1)/P(r1))
35 Pm1byr1=(Pm1*Pr1bym1)/Pr1;
36 disp(Pm1byr1, 'P(m1/r1): ');
37
38 //(d) Probabilty error
39 //As Pe=(P(r1/m0)*P(m0))+P(r0/m1)*P(m1)
40 Pe=(Pr1bym0*Pm0)+(Pr0bym1*Pm1);
41 disp(Pe, 'Probability error: ');
42
43 //(e) Probabilty that is transmitted correctly
44 //As Pc=(P(r0/m0)*P(m0))+Pr1bym1*Pm1
45
46 Pc=(Pr0bym0*Pm0)+(Pr1bym1*Pm1);
47 disp(Pc, 'Probabilty that is transmitted correctly: ')
    ;

```

Scilab code Exa 6.16 Pulse Positions

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



```

2 //Example 6.16
3 clc;
4 //Given
5 p=0.4;
6 Pp=p;
7 q=0.3;
8 Pn=q;
9 a=1; //i start value
10 b=6; //i end value
11 //(a) Probabilty that all pulses are positive
12 s=1;
13 for i=a:b
14     s=s*Pp;
15 end
16 disp(s, 'Probabilty that all pulses are positive:');
17
18 //(b)Pulses are positive ,positive , positive , zero ,
    zero ,negative
19 Pz=1-(p+q);
20 s1=(Pp^3)*(Pz^2)*Pn;
21 disp(s1, 'Probabilty that all pulses are positive ,
    positive , positive , zero ,zero ,negative:');

```

Scilab code Exa 6.17 Binary Source

```

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

```

```
Probability for 1,1,0,0,0:

0.2304

Probability for atleast three 1 s:

0.68256

-->
```

Figure 6.4: Binary Source

```
9
10 // (a) 1,1,0,0,0
11 xf=(factorial(n))/((factorial(j))*(factorial(n-j)));
12 s=xf*(P1^j)*(P0^(n-j));
13 disp(s,'Probability for 1,1,0,0,0:');
14
15 // (b) atleast 3 1's
16 // P(X>=3)=1-P(X<=2)
17 // Here y=1-x
18 x=0;
19 for k=0:2
20     f=(factorial(n))/((factorial(k))*(factorial(n-k)
21         ));
22     x=x+{f*((P1^k)*(P0^(n-k)))};
23 end
24 disp(y,'Probability for atleast three 1 s:');
```

```
Scilab 5.4.1 Console

Probabilty of more than one error in 10 recieved digits:

    0.0042662

Using Poisson Approximation:

    0.0046788

-->
```

Figure 6.5: Noisy Transmission Channel

Scilab code Exa 6.20 Noisy Transmission Channel

```
1 //Page Number: 6.23
2 //Example 6.20
3 clc;
4 //Given
5 pe=0.01; //Error probability
6
7 //(a) Probabilty of more than one error in 10
   recieved digits
8 n=10;
9 //As  $P(X>1)=1-P(X=0)-P(X=1)$ 
10 //Let  $x=P(X>1)$ 
11 //s= $P(X=0)+P(X=1)$ 
12 s=0;
13 for t=0:1
14     f=(factorial(n))/((factorial(t))*(factorial(n-t)
       ));
```

```

15     s=s+{f*(pe^t)*((1-pe)^(n-t))};
16 end
17 x=1-s;
18 disp(x,'Probabilty of more than one error in 10
    recieved digits:');
19
20 //(b) Using Poisson approximation
21 //P(X=k) ~ [ {( %exp)^(-n*p) } * { ((n*p)^k) } ] / k factorial
22 s1=0;
23 for k=0:1
24     j=factorial(k);
25     s1=s1+[ { exp(-n*pe) } * { ((n*pe)^k) } ] / j;
26 end
27 x1=1-s1;
28 disp(x1,'Using Poisson Approximation:');

```

Scilab code Exa 6.23 Pdf of random variable

```

1 //Page Number: 6.23
2 //Example 6.23
3 clc;
4 //We find , k=1/(b-a)
5 //(b) if a=1 and b=2,P(|x|<c) where c=1/2
6 a=-1;
7 b=2;
8 c=1/2;
9
10 k=1/(b-a);
11 //P(|x|<c) = P(-c<=x<=c)
12 //Let y
13 x0=-c;x1=c;
14 y=integrate('1','x',x0,x1);
15 y1=k*y;

```

```
Scilab 5.4.1 Console  
  
P ( |x| < c ) :  
  
    0.3333333  
  
-->|
```

Figure 6.6: Pdf of random variable

```
Scilab 5.4.1 Console

Average errors per block:

    0.16

Variance of errors per block:

    0.1584

Probability that number of errors per block is bgeater or equal than 4:

    0.0000165

-->
```

Figure 6.7: 16 Binary Digits Transmission

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

Scilab code Exa 6.41 16 Binary Digits Transmission

```
1 //Page Number: 6.34
2 //Example 6.41
3 clc;
4 //Given
5 n=16; //binary digits
6 p=0.01; //Probabilty error due to noise
7
8 //(a) Average errors per block
9 //E(X)=n*p
10 EofX=n*p;
11 disp(EofX, 'Average errors per block:');
```

```

12
13 //(b) Variance of errors of block
14 //s=n*p*(1-p)
15 s=n*p*(1-p);
16 disp(s, 'Variance of errors per block:');
17
18
19 //(c) Probability that number of errors per block is
    bgeater or equal than 4
20 i=4;
21 //AsP(X>=4)=1-P(X<=3)
22 P3=0;
23 for k=0:3;
24     f=(factorial(n))/((factorial(k))*(factorial(n-k)
        ));
25     P3=P3+(f*(p^k)*((1-p)^(n-k)));
26 end
27 P4=1-P3;
28 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

```
1 //Page Number: 8.6
2 //Example 8.1
3
4 clc;
5 disp('Part a');
6 //(a) Given, u1=1W and u2=1mW
7 //Change to dBW and dBm
8 u1=1D-3;
9 u2=1;
10 //(i) 470mW
11 a=470D-3;
12 y1=(10*log10(a/u1));
13 disp('dBm',y1);
14
15 y2=(10*log10(a/u2));
16 disp('dBW',y2);
17
18 //(ii) 1W
19 b=1;
```



```
Scilab 5.4.1 Console

Part a
  26.720979
dBm
- 3.2790214
dBW
  30.
dBm
  0.
dBW
- 40.
dBm
- 70.
dBW

Part B
  0.01
W
  50.118723
W
  0.001
W

Part C
Received Power
W
  0.01

Part D
=
  16.
dB
  39.810717

Part E
  40.
dB
-->
```

Figure 8.1: Units Conversion

```

20 z1=(10*log10(b/u1));
21 disp('dBm',z1);
22
23 z2=(10*log10(b/u2));
24 disp('dBW',z2);
25
26 //(iii)100mW
27 c=100D-9;
28 x1=(10*log10(c/u1));
29 disp('dBm',x1);
30
31 x2=(10*log10(c/u2));
32 disp('dBW',x2);
33
34 disp('Part B');
35 //(b)Here u1=1W (for dBW)and u2=1mW (for dBm)
36 //Change to powers to watts
37 //(i)-20dBW
38 a=-20;
39 k1=u2*(10^(a/10));
40 disp('W',k1);
41
42
43 //(ii)47dBm
44 b=47;
45 k2=u1*(10^(b/10));
46 disp('W',k2);
47
48
49 //(ii)0dBm
50 c=0;
51 k3=u1*(10^(c/10));
52 disp('W',k3);
53
54
55 disp('Part C');
56 //(c)Given, channel loss=20dB and Pt=1W
57 l=-20;

```

```

58 PT=1;
59 PR=10^(1/10);
60 disp(PR, 'W', 'Received Power ',);
61
62 disp('Part D');
63 //(d) Given, channel loss=30dB when signal=3dB and
    overall loss=20dB
64 l1=-30;
65 s=-3;
66 l2=-20;
67 q=-l1-s-s+l2;
68 d1=10^(q/10);
69 disp('dB', q, '=');
70 disp(d1);
71
72 disp('Part E');
73 //(e) Given,
74 Si=0; //dBm
75 S1=1D-3*(10^(Si/10));
76 Ni=1D-7; //W
77
78 Osnr=S1/Ni;
79 Odb=(10*(log10(Osnr)));
80 disp('dB', Odb);

```

Scilab code Exa 8.2 Rms Voltage

```

1 //Page Number: 8.7
2 //Example 8.2
3 clc;
4 //Given,
5 R=1000;
6 T=27; //degree celsius

```

```
Scilab 5.4.1 Console

Rms noise voltage:

    4.069D-11

V
-->
```

Figure 8.2: Rms Voltage

```
7 TK=T+273; //kelvin
8
9 // We know, rms noise voltage is 4RKT B
10 K=1.38D-28;
11 B=10;
12
13 V=sqrt(4*R*K*TK*B);
14 disp('V',V,'Rms noise voltage:');
```

Scilab code Exa 8.3 Output Noise Power

```
1 //Page Number: 8.8
2 //Example 8.3
3 clc;
4
5 //Given ,
6 G=100;
7 G1=(10^(G/10));
8
9 T=30;
```

```
Scilab 5.4.1 Console

Output Noise Power

0.0000621

W
-->
```

Figure 8.3: Output Noise Power

```
10 Te=270;
11
12 //We know, output noise power=GKB(T+Te)
13 K=1.38D-23;
14 B=1.5D+6;
15
16 No=G1*1.38D-23*1.5D+6*(T+Te);
17 disp('W',No,'Output Noise Power');
```

Scilab code Exa 8.4 Noise Figure

```
1 //Page Number: 8.8
2 //Example 8.4
3 clc;
4
5 //Given ,
6 R=50;
7 osnr=0;
```

```
Scilab 5.4.1 Console

Output SNR

    1.

Input noise power

    2.001D-15

W

Signal Power Input

    5.000D-15

W

Input SNR

    2.4987506

Noise Factor

    2.4987506

Noise figure

    3.9772292

dB

-->
```

Figure 8.4: Noise Figure

```

8 SNRo=(10^(osnr/10));
9 disp(SNRo,'Output SNR');
10
11 //As Pni=KTB
12 K=1.38D-23;
13 T=290;
14 B=5D+5;
15 Pni=K*T*B;
16 disp('W',Pni,'Input noise power');
17
18 //Psi=V^2/R
19 //Given V=5*10^-6V
20 V=0.5D-6;
21 Psi=(V^2)/R;
22 disp('W',Psi,'Signal Power Input');
23
24 isnr=(Psi/Pni);
25 disp(isnr,'Input SNR');
26
27 F=(isnr/SNRo);
28 disp(F,'Noise Factor');
29
30 NF=10*log10(F);
31 disp('dB',NF,'Noise figure');

```

Scilab code Exa 8.6 Cascaded Amplifiers

```

1 //Page Number: 8.9
2 //Example 8.6
3 clc;
4
5 //Given, Stage 1
6 SNRo=120;

```

```
Scilab 5.4.1 Console

Noise factor of stage 1

    3.0682616

Noise figure of stage 1

    4.8689238

dB

Input signal power of stage 1

    0.0000037

W

Noise added by stage 1

    0.0000004

W

-->
```

Figure 8.5: Cascaded Amplifiers


```

7 Pni=0.01D-6; //W
8 G1=20;
9
10 //Stage 2
11 F2=12; //dB
12 FF2=(10^(F2/10));
13 G2=30;
14
15 //Stage 3
16 F3=9.3; //dB
17 FF3=(10^(F3/10));
18 G3=35;
19
20 //(a)Nosie factor and noise figure of Stage 1
21 F=5.6; //dB
22 FF=(10^(F/10));
23
24 //As  $F=F_1 - ((F_2-1)/G_1) - ((F_3-1)*(G_1G_2))$ ;
25  $F_a = FF - ((F_2-1)/G_1) - ((FF_3-1)/(G_1*G_2))$ ;
26 disp(Fa, 'Noise factor of stage 1');
27
28 FadB=(10*(log10(Fa))); //dB
29 disp('dB',FadB, 'Noise figure of stage 1');
30
31
32 //(b)Input signal power of stage 1
33 Psi=Pni*Fa*SNRo;
34 disp('W',Psi, 'Input signal power of stage 1');
35
36
37 //(c)Nosie added by stage 1
38 N=(Fa-1)*G1*Pni;
39 disp('W',N, 'Noise added by stage 1');

```

```
Scilab 5.4.1 Console

Overall Noise Figure

    2.0041827

dB

Equivalent Noise Temperature

    170.0619

K

-->
```

Figure 8.6: Noise figure and temperature

Scilab code Exa 8.7 Noise figure and temperature

```
1 //Page Number: 8.10
2 //Example 8.7
3 clc;
4
5 //Given
6 Te=127; //Kelvin
7 T=290; //Kelvin
8
9 G1=100;
10
11 F2dB=12; //db
12 F2=(10^(F2dB/10));
13
14 F1=1+(Te/T);
15
16 F=F1+((F2-1)/G1);
```

```

17 FF=(10*log10(F));
18 disp('dB',FF,'Overall Noise Figure');
19
20 //Equivalent Noise Temperature TE
21 TE=(F-1)*T;
22 disp('K',TE,'Equivalent Noise Temperature');

```

Scilab code Exa 8.9 WLAN Receiver

```

1 //Page Number: 8.11;
2 //Example 8.9
3 clc;
4
5 //(a) Noise Figure
6 //Given
7 //Loss Factor
8 IL=1.5; //dB
9 IL1=(10^(IL/10));
10
11 //Noise figure
12 F1=1.41;
13 G1=1/F1;
14
15 G2=10;
16 GG2=(10*log10(G2)); //dB
17
18 G3=100;
19 GG3=(10*log10(G3)); //dB
20
21 F2=2; //dB
22 F3=2; //dB
23
24 FF2=(10^(F2/10));

```

Noise figure of cascade

3.6495777

dB

Noise power output:

- 64.280813

dBm

Signal to Noise ratio:

2.780813

dB

Noise figure will be:

2.017124

dB

-->

Figure 8.7: WLAN Receiver

```

25 FF3=(10^(F3/10));
26
27 F=(F1+((FF2-1)/G1)+((FF3-1)/(G1*GG2)));
28 FF=(10*log10(F));
29 disp('dB',FF,'Noise figure of cascade');
30
31 //(b) SNR at output
32 //Given
33 Pin=-90; //dBm
34 Pout=Pin-IL+GG2+GG3; //dBm
35
36 //Pn=Gcas*K*Te*B (cascade)
37 K=1.38D-23;
38 To=290; //Kelvin
39 B=1D+8;
40 Gcas=GG2+GG3-IL;
41 Gcas1=(10^(Gcas/10));
42 Pn=K*To*(F-1)*B*Gcas1; //W
43
44 Pn1=(10*(log10(Pn/1D-3)));
45 disp('dBm',Pn1,'Noise power output:');
46
47 SNR=Pout-Pn1;
48 disp('dB',SNR,'Signal to Noise ratio:');
49
50 //(c) Best Noise Figure
51 //G1 after G2 after IL
52
53 Fcas=(FF2+((FF3-1)/G3)+((IL1-1)/(G3*G2)));
54 Fcas1=(10*(log10(Fcas)));
55
56 disp('dB',Fcas1,'Noise figure will be:');

```

```
Scilab 5.4.1 Console

Nant=

    3.312D-13

W

Nant=

    1.656D-12

W

Receiver Noise Power Output

    0.0001987

W

-->
```

Figure 8.8: Receiver Noise Power

Scilab code Exa 8.10 Receiver Noise Power

```
1 //Page Number: 8.12
2 //Example 8.10
3 clc;
4
5
6 //Given
7 K=1.38D-23;
8 B=40D+6;
9
10 Tant=600; //Kelvin
11 Trec=3000; //Kelvin
12
13 G=80; //dB
14 GG=(10^(G/10));
15
16 //Input noise power from antenna
17 Nant=K*Tant*B; //W
18 disp('W',Nant,'Nant=');
19
20 Nrec=K*Trec*B; //W
21 disp('W',Nrec,'Nant=');
22
23 Nout=(Nant+Nrec)*GG;
24 disp('W',Nout,'Reciver Noise Power Output');
```

Scilab code Exa 8.11 Cascade Amplifiers A and B

```
1 //Page Number: 8.12
2 //Example 8.11
3 clc;
4
```

```
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


```

5 //We use ,  $F=(F1+(F2-1)/G1)$ 
6 //Given
7 FA=1.5;
8
9 GA=10 , //dB
10 GAA=(10^(GA/10));
11
12 FB=3; //dB
13 FBB=(10^(FB/10));
14
15 GB=15 , //dB
16 GBB=(10^(GB/10));
17
18 //Case 1: Amp A followed by Amp B
19 F11=FA;
20 F12=FBB;
21 G11=GAA;
22
23 F1=(F11+(F12-1)/G11);
24 disp(F1, 'Gain when Amp A followed by Amp B');
25
26 //Case 2: Amp B followed by Amp A
27 F21=FBB;
28 F22=FA;
29 G21=GBB;
30
31 F2=(F21+(F22-1)/G21);
32 disp(F2, 'Gain when Amp B followed by Amp A');
33
34 //As  $F1 < F2$ , Case 1 gives lowest Noise
35
36 //Also given ,
37 T0=20 //degree celsius
38 T=T0+273; //Kelvin
39
40 //For amplifier A
41 TA=((FA-1)*T);
42

```

```

43 //For amplifier B
44 TB=((FBB-1)*T);
45
46 //When A is followed by B
47 Te1=(F1-1)*T;
48 disp(Te1,'Noise temperataure when Amp A followed by
      Amp B');
49
50 //When B is followed by A
51 Te2=(F2-1)*T;
52 disp(Te2,'Noise temperataure when Amp B followed by
      Amp A');

```

Scilab code Exa 8.12 Distortion Component

```

1 //Page Number: 8.13
2 //Example 8.12
3 clc;
4 //Given, Noise figure
5 NF=2; //dB
6 F=(10^(NF/10));
7
8 AG=12; //dB
9 A=(10^(AG/10));
10
11 //(a) Total Output Noise Power
12
13 //Also given, Input signal power
14 Pi=1; //W
15
16 //Input Noise power Pni
17 Pni=100D-3; //W
18

```

Scilab 5.4.1 Console

Total Output Noise Power

2.5118864

W

SNAD:

8.122798

dB

-->

Figure 8.10: Distortion Component

```

19 //Input SNR
20 Isnr=Pi/Pni;
21
22 //Output SNR
23 Osnr=Isnr/F;
24
25 //Total output signal power
26 Po=Pi*A; //W
27
28 //Total output noise power
29 N=Po/Osnr; //W
30 disp( 'W',N, 'Total Output Noise Power');
31
32 //(b)Signal to Noise and disortion ratio
33
34 //Given. 2% is disortion
35 Di=2/100;
36
37 //Total disortion
38 D=Di*A; //W
39
40 //Useful Power
41 S=(1-Di)*A; //W
42
43 //As given ,SNAD=10*(log10 (S+N+D)/(N+D));
44 SNAD=10*(log10((S+N+D)/(N+D)));
45 disp( 'dB',SNAD, 'SNAD: ');

```

Scilab code Exa 8.13 Receiver Antenna System

```

1 //Page Number: 8.14
2 //Example 8.13
3 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

```

4 //Given
5 Pni=-100; //dBm
6 PniW=((1D-3)*(10^(Pni/10)));
7
8 To=290; //K
9
10 F=1.6; //dB
11 NF=(10^(F/10));
12
13 //(a) Noise tempertaure of antenna
14 //As Te=Pni/K*B;
15 K=1.38D-23;
16 B=20D+6;
17 Te=(PniW/(K*B));
18 disp('K',Te,'Noise tempertaure of antenna');
19
20 //(b) Effective noise tempertaure
21 //Given,
22 G=30; //dB
23 GdB=(10^(G/10));
24
25 Tef=((NF-1)*To);
26 disp('K',Tef,'Effective Noise tempertaure');
27
28 //Output Noise Pno=K*T(Te+Tef)*B*GdB
29
30 Pno=K*(Te+Tef)*B*GdB; //W
31 Pno1=(10*(log10(Pno/1D-3)));
32 disp('dBm',Pno1,'Output Noise: ');

```

Scilab code Exa 8.14 TV Recievers

1 //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 Receivers

```

2 //Example 8.14
3 clc;
4 //Given
5 GG1=20; //dB
6 G1=(10^(GG1/10));
7
8 FF1=6; //dB
9 F1=(10^(FF1/10));
10
11 GG2=60; //dB
12 G2=(10^(GG2/10));
13
14 FF2=16; //dB
15 F2=(10^(FF2/10));
16
17 LF=3; //dB
18 FC=(10^(LF/10));
19 GC=1/FC;
20
21 //(a) Overall Noise Figure
22 //Usinng F=(F1+((F2-1)/G1)+((F3-1)(G1*G2)));
23
24 Fa=(F1+((FC-1)/G1)+((F2-1)/(G1*GC)));
25 Fadb=(10*(log10(Fa)));
26 disp('db',Fadb,'Overall Noise Figure:');
27
28
29 //(b)Noise figure, if pre-amplifier is removed and
    gain increased by 20dB
30
31 Fb=FC+((F2-1)/GC);
32 Fbdb=(10*(log10(Fb)));
33 disp('db',Fbdb,'Overall Noise Figure:');
34
35 //(c)Change in noise figure
36 //Again usinng F=(F1+((F2-1)/G1)+((F3-1)(G1*G2)));
37 Fc=(FC+((F1-1)/GC)+((F2-1)/(G1*GC)));
38 Fcdb=(10*(log10(Fc)));

```


39

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

Scilab code Exa 8.15 Reciever

```
1 //Page Number: 8.15
2 //Example 8.15
3 clc;
4
5 //Given Noise figure
6 NF=5; //dB
7 F=(10^(NF/10));
8
9 Ta=1050; //Kelvin
10
11 //(a) Overall Noise Figure
12 T=20; // degree Celsius
13 To=T+273; //Kelvin
14
15 //Effective Noise temperature
16 Te=((F-1)*To);
17 disp('K',Te,'Effective Noise temperature');
18
19 //Overall effective Noise Temperature
20 TIN=Ta+Te;
21 disp('K',TIN,'Overall Effective Noise temperature');
22
23 //Overall noise figure
24 ONF=(1+(TIN/To));
25 ONFdB=(10*(log10(ONF))); //dB
26
27 disp('dB',ONFdB,'Overall Noise Figure:');
28
```

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:

```

29
30 //(b)Input Signal Power
31 //Given Output SNR
32 Outsnr=6; //dB
33 Osnr=(10^(Outsnr/10));
34
35 Isnr=ONF*Osnr;
36
37 //Input Noise Power=KTB
38 K=1.38D-23;
39 B=50000;
40 Pni=K*TIN*B; //W
41
42 //Input signal Power
43 Psi=Ismr*Pni; //W
44 PsidBW=(10*(log10(Psi/1))); //dBW
45 disp('dBW',PsidBW,'Input signal Power:');
46
47
48 //(c)Minimum detectable signal Vmin
49 //Given
50 Osnr=10; //dB
51 R=50; //Ohms
52
53 FF1=3; //dB
54 F1=(10^(FF1/10));
55
56 FF2=5; //dB
57 F2=(10^(FF2/10));
58
59 GG1=7;
60 G1=(10^(GG1/10));
61 //Using F=F1+((F2-1)/G1)
62
63 Fa=F1+((F2-1)/G1);
64
65 Fa1=(10*(log10(Fa)));
66

```

```

67 //Equivalent Noise Tempertaure
68 Te1=((Fa-1)*To);
69 disp('K',Te1,'Equivalent Noise temperature:');
70
71 //Overall effective Noise Temperature
72 TIN1=Ta+Te1;
73 disp('K',TIN1,'Effective Noise temperature:');
74
75 //Input Noise Power=KTB
76 Pni1=K*TIN1*B; //W
77
78 //Overall noise figure
79 ONF1=(1+(TIN1/To));
80 ONFdB1=(10*(log10(ONF)));
81 disp('W',ONFdB1,'Overall Noise Figure:');
82
83 //Input SNR
84 Isn1=ONF1*Osnr;
85
86 //Input signal Power
87 Psi1=Isn1*Pni; //W
88 disp('W',Psi1,'Input Signal Power:');
89
90 //Now as  $V_{min}^2/R=Psi1$ 
91 //Therefore
92 Vmin=sqrt(Psi1*R);
93 disp('V',Vmin,'Minimum detectable signal Vmin:');

```

Scilab code Exa 8.18 Repeaters

```

1 //Page Number: 8.18
2 //Example 8.18
3 clc;

```

```
Scilab 5.4.1 Console

Noise figure:

    400.

dB

Power transmitted with no repeaters

    5.060D+26

W

Power transmitted with 20 repeaters

    1.012D-10

W

-->
```

Figure 8.14: Repeaters

```

4 //Given ,
5
6 Fa=5; //dB
7 d=200; //Km
8 a=2; //dB/Km
9 No=4D-21; //W/Hz
10 BW=4000;
11 0snr=30; //dB
12 //(a) No repeaters used
13 L=d*a; //dB
14 disp('dB',L,'Noise figure:');
15
16 //As Output SNR=InputSNR/F where F=L*Fa
17 //And Input SNR=(Pt/(No*B))
18 //Therefore ,PT=Output SNR+L+Fa+(No*B)
19
20 NoB=10*log10(No*BW);
21
22 //Power Transmitted
23 Pt=0snr+L+Fa+(NoB);
24
25 PtdB=10^(Pt/10);
26 disp('W',PtdB,'Power transmitted with no repeaters')
    ;
27
28 //(b) 20 repeaters are employed
29 n=20;
30 //F becomes 20F
31 //Output SNR=InputSNR/20*F where F=L*Fa
32 L1=L/n; //dB per segment
33
34 //Power Transmitted
35 Pt1=0snr+L1+Fa+(NoB)+(10*(log10(n)));
36
37 PtdB1=10^(Pt1/10);
38 disp('W',PtdB1,'Power transmitted with 20 repeaters')
    );

```

Scilab code Exa 8.23 Noise process

```
1 //Page Number: 8.23
2 //Example 8.23
3 clc;
4 //Given ,
5 //  $S=10D-8*(1-(|f|/10D+8))$ ;
6
7
8 //(a)Power content of output noise
9 //Bandwidth of 2MHz centered at 50MHz
10 //Therefore , first limits will be
11
12  $x0=-51D+6$ ;
13  $x1=-49D+6$ ;
14  $P1=\text{integrate}('1+(f/10^8)', 'f', x0, x1)$ ;
15
16 //And, second limits will be
17
18  $x2=49D+6$ ;
19  $x3=51D+6$ ;
20
21  $P2=\text{integrate}('1-(f/10^8)', 'f', x2, x3)$ ;
22
23  $P=10D-8*(P1+P2)$ ;
24 disp('W',P, 'Power content:');
```

Scilab code Exa 8.24 White Gaussian Noise

Scilab 5.4.1 Console

Power content:

0.2

W

-->

Figure 8.15: Noise process


```
Scilab 5.4.1 Console

Noise power at input of filter:

    1.536

W

-->
```

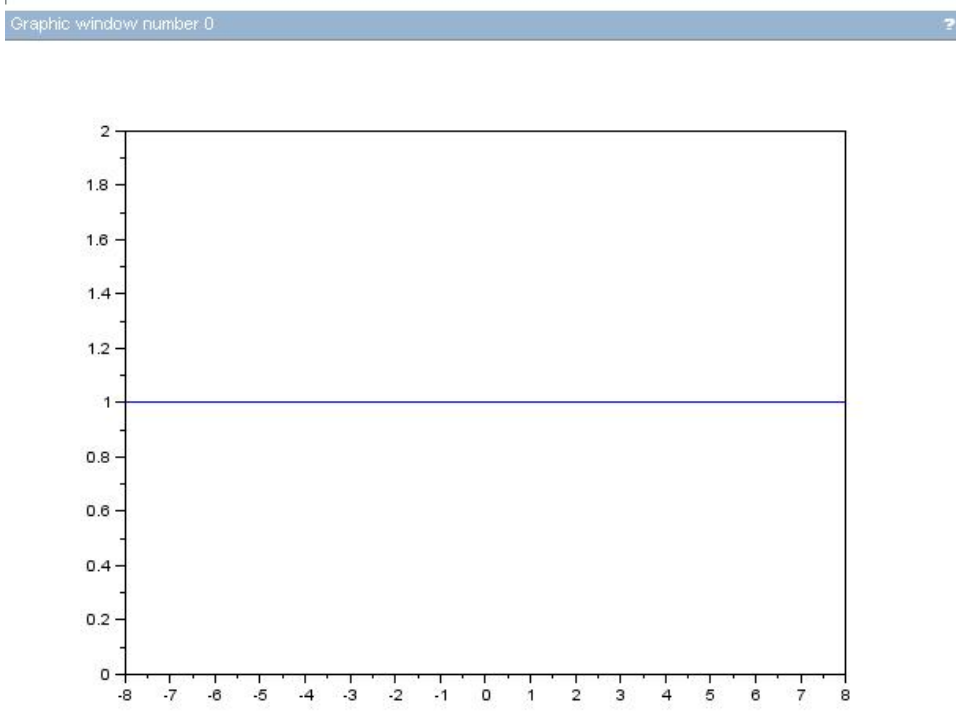


Figure 8.16: White Gaussian Noise

```

1 //Page Number: 8.25
2 //Example 8.24
3 clc;
4 //Given, band limited gaussian noise with psd,
5 S=9.6D-5; //W/Hz for  $|f| < 8\text{kHz}$ 
6 L=100D-3; //H
7 R=100; //Ohms
8
9 //(a) Noise power at input of filter
10
11 x0=-8000;
12 x1=8000;
13 Pni=S*(integrate('1','f',x0,x1));
14 disp('W',Pni, 'Noise power at input of filter:');
15
16 //Plot
17 x=linspace(-8,8,3);
18 y=linspace(0,1,1)*ones(1,length(x));
19
20 plot(x,y);

```

Chapter 9

Noise in Analog Communication System

Scilab code Exa 9.1 Output Noise Power

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

Scilab 5.4.1 Console

Output Noise Power:

0.0000251

W

-->

Figure 9.1: Output Noise Power

```
Scilab 5.4.1 Console

SNR:

    1.25

SNR1:

    1.2376238

-->
```

Figure 9.2: SNR of filters

Scilab code Exa 9.4 SNR of filters

```
1 //Page Number: 9.12
2 //Example 9.4
3 clc;
4 //Given
5 //(a)
6 H1=1;
7 H2=(1/2);
8 S0=(H1^2)/2;
9 w0=-2*%pi;
10 w1=2*%pi;
11 N0=(0.1/(2*%pi))*2*(integrate('1','w',w0,w1));
12 SNR=S0/N0;
13 disp(SNR,"SNR: ");
14
15 //(b)
16 S01=(H1^2*H2^2)/2;
17 N01=0.101;
18 SNR1=S01/N01;
19 disp(SNR1,"SNR1: ");
```

Scilab code Exa 9.8 AM with envelope detection

```
1 //Page Number: 9.15
2 //Example 9.8
3 clc;
4 //Given
5 p=0.99;
6 u=1;
7 q=1-p;
8 //As  $\exp(-Ac^2/4*n*B)=1-p$ 
9 //And  $AC^2/2*n*B=S/N$ 
```

```
Scilab 5.4.1 Console

S/N:

    10.

db

-->|
```

Figure 9.3: AM with envelope detection

```
Minimum Value of Ac:
```

```
0.0309839
```

```
v
```

```
Minimum Value of Ac at Threshold:
```

```
0.0005237
```

```
v
```

```
-->
```

Figure 9.4: AM with additive thermal noise

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

Scilab code Exa 9.9 AM with additive thermal noise


```

1 //Page Number: 9.16
2 //Example 9.9
3 clc;
4 //Given
5 Si=7D+4;
6 u=1;
7 BW=4D+3; //Hz
8 n=2D-12; //W/Hz
9
10 //(a)Minimum value of Ac
11 SbyN=40; //dB
12 SN=10^(SbyN/10);
13
14 Sx=2*[integrate('(x^2)*(-x+1)', 'x', 0, 1)];
15 //Now
16 g=SN/(Sx/(1+Sx));
17 //And
18 //Ac=sqrt((2*n*BW*g)/(1+(u^2*Sx)))
19 //We have
20 Ac=sqrt((2*n*BW*g)/(1+(u^2*Sx)));
21 disp('V', Ac, 'Minimum Value of Ac: ');
22
23 //(b)Threshold value of Ac
24 //AS S/N at threshold is 10dB
25 SNT=10; //dB
26 gT=2*SNT;
27 AcT=sqrt((2*n*BW*gT)/(1+(u^2*Sx)));
28 disp('V', AcT, 'Minimum Value of Ac at Threshold: ');

```

Scilab code Exa 9.10 Modulation Systems transmission Bandwidth

```

1 //Page Number: 9.17
2 //Example 9.10

```

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

```

3  clc;
4  //Given
5  BW=10D+3; //Hz
6  SNR=40; //dB
7  SNRO=10^(SNR/10);
8  P=40; //dB
9  PL=10^(P/10);
10 n=2D-9; //W/Hz
11 USx2=0.5;
12 //For DSB, AM and SSB bandwidth requirement
13 BTD=2*BW;
14 BTA=2*BW;
15 BTS=BW;
16 disp('Hz',BTD,'Transmission Bandwidth for DSB:');
17 disp('Hz',BTA,'Transmission Bandwidth for AM:');
18 disp('Hz',BTS,'Transmission Bandwidth for SSB:');
19
20
21 //Pt for DSB and SSB
22 //As SNRO=Si/nBW
23 Si=n*BW*SNRO; //W
24 //Considering Channel loss
25 ST=Si*PL;
26 disp('W',ST,'Power transmission for DSB and SSB:');
27
28 //Pt for AM
29 //As SNRO=x*Si/nBW
30 //x=USx2/(1+USx)
31 x=USx2/(1+USx2);
32 Si1=(n*BW*SNRO)/x; //W
33 //Considering Channel loss
34 ST1=Si1*PL;
35 disp('W',ST1,'Power transmission for AM:');

```

Ac for $s(t)=100\text{mw}$:

3.6514837

V

Power in absence of Noise:

0.025

W

Power:

0.00015

W

SNR at output for SSB:

22.218487

dB

131

SNR at output for DSB:

25.228787

Scilab code Exa 9.13 SSB System

```
1 //Page Number: 9.20
2 //Example 9.13
3 clc;
4 //Given
5 Mf1=0.003;
6 //for f<=1.5D+3 Hz
7 f1=1.5D+3; //Hz
8
9 Mf2=0.001;
10 //for 1.5D+3 <=f<=3D+3 Hz
11 f2=3D+3; //Hz
12
13 Mf3=0;
14 //for f>3D+3 Hz
15
16
17 //(a) Ac that power transmitted is 100mW
18 St=100D-3; //W
19 //As  $St=2*[\{f1*(Mf1*Ac/2)^2\}+\{f1*(Mf2*Ac/2)^2\}+\{f2*(Mf3*Ac/2)^2\}]$ 
20 //Neglecting Mf3 as zero
21 Ac=sqrt((4*St)/(2*f1*(Mf1^2+Mf2^2)));
22 disp('V',Ac,'Ac for s(t)=100mw:');
23
24 //(b)Power in absence of noise
25 Zt=2*[\{f1*(((Mf1*Ac)/4)^2)\}+\{f1*(((Mf2*Ac)/4)^2)\}+\{f2*(((Mf3*Ac)/4)^2)\}];
26 disp('W',Zt,'Power in absence of Noise:');
27
28 //(c)
29 //Given
30 N0=0.0001D-3; //W/Hz
31 //Psd=N0/4
32 //Pt=2*f1*N0/4
33 Pt=(2*f2*N0)/4;
34 disp('W',Pt,'Power:');
```

```

35
36 //(d) SNR at output
37 SNR=Zt/Pt;
38 SNR0=10*log10(SNR);
39 disp('dB',SNR0,'SNR at output for SSB:');
40
41 //(e) For DSB
42 St1=100D-3; //W
43 //As St=4*[{f1*(Mf1*Ac/2)^2}+{f1*(Mf2*Ac/2)^2}+{f2*(
    Mf3*Ac/2)^2}]
44 //Neglecting Mf3 as zero
45 Ac1=sqrt((4*St)/(4*f1*(Mf1^2+Mf3^2)));
46
47 Zt1=4*[{f1*(((Mf1*Ac)/4)^2)}+{f1*(((Mf2*Ac)/4)^2)}+{
    f2*(((Mf3*Ac)/4)^2)}];
48
49 //SNR at output
50 SNR1=Zt1/Pt;
51 SNR01=10*log10(SNR1);
52 disp('dB',SNR01,'SNR at output for DSB:');
53
54 //3dB increase in SNR
55 //DSB has higher SNR but SSB os spectarally
    efficient

```

Scilab code Exa 9.14 FM broadcast System

```

1 //Page Number: 9.22
2 //Example 9.14
3 clc;
4 //Given
5 delf=75D+3; //Hz
6 W=15D+3; //Hz

```

```
Scilab 5.4.1 Console

Output SNR:

    15.740313

dB

-->|
```

Figure 9.7: FM broadcast System

```

7 Sx=1/2;
8 //As SNRO=3(delf/W)^2*Sx*g
9 // Assume g=1
10 g=1;
11
12 SNRO=3*(delf/W)^2*Sx*g;
13 SNdB=10*log10(SNRO);
14 disp('dB',SNdB,'Output SNR:');
15
16 //Hence it is SNdB times better

```

Scilab code Exa 9.17 Audio Signal Over RF Channel

```

1 //Page Number: 9.23
2 //Example 9.17
3 clc;
4 //Given
5 oSNR=40; //dB
6 SNRO=10^(oSNR/10);
7 n=2D-10; //W/Hz
8 l=50; //dB
9 PL=10^(l/10);
10 B=15D+3; //Hz
11 Sx=1/2;
12 //(a) DSB Modulation
13 BTD=2*B;
14 disp('Hz',BTD,'Transmission bandwidth for DSB:');
15 //As SNRO=Si/(n*B)
16 SiD=SNRO*n*B;
17 STD=SiD*PL;
18 disp('W',STD,'Average Power transmitted for DSB:');
19
20 //(b) AM

```


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:

```

21 U=1;
22 U2Sx=U*U*Sx;
23 BTA=2*B;
24 disp('Hz',BTA,'Transmission bandwidth for AM:');
25 //As  $SNRO=x*Si/(n*B)$ 
26 //where  $x=USx/(1+USx)$ 
27 x=U2Sx/(1+U2Sx);
28 SiA=(SNRO*n*B)/x;
29 STA=SiA*PL;
30 disp('W',STA,'Average Power transmitted for AM:');
31
32 //(c)PM
33 kp=3;
34 BTP=2*(kp+1)*B;
35 disp('Hz',BTP,'Transmission bandwidth for PM:');
36 //As  $SNRO=kp^2*Sx*Si/(n*B)$ 
37 SiP=(SNRO*n*B)/(Sx*(kp^2));
38 STP=SiP*PL;
39 disp('W',STP,'Average Power transmitted for PM:');
40
41 //(d)FM
42 D=5;
43 BTF=2*(D+1)*B;
44 disp('Hz',BTF,'Transmission bandwidth for FM:');
45 //As  $SNRO=3*D^2*Sx*Si/(n*B)$ 
46 SiF=(SNRO*n*B)/(3*(D^2)*Sx);
47 STF=SiF*PL;
48 disp('W',STF,'Average Power transmitted for FM:');

```

Scilab code Exa 9.18 Modulation Index

```

1 //Page Number: 9.24
2 //Example 9.18

```

```
Scilab 5.4.1 Console

Modulation index:

    2.917128

-->
```

Figure 9.9: Modulation Index

```

3  clc;
4  //(b) Modulation index b
5  // Given
6  SNdB=30; //dB
7  SNRO=10^(SNdB/10);
8  //As SNRO=30*b^2*(b+1)
9  // Therefore
10 p2=poly(0, 'x');
11 p3 =30*(p2^3)+30*(p2^2)-1000;
12 r=roots(p3);
13 t=r(3,1);
14 disp(t, 'Modulation index:');

```

Scilab code Exa 9.21 Normalized Message Signal

```

1  //Page Number: 9.27
2  //Example 9.21
3  clc;
4  // Given
5  BW=5000; //Hz
6  P=0.1; //W
7  CBW=100D+3; //Hz
8  A=80; //dB
9  A1=10^(-A/10);
10 N0=2*(0.5D-12); //W/Hz
11 Pt=10D+3; //W
12
13 //We know, CBW=2*(D+1)*BW
14 // Therefore
15 D=(CBW/(2*BW))-1;
16 kp=D;
17 Si=Pt*A1;
18 Sx=P;

```

Scilab 5.4.1 Console

```
SNR at output:
```

```
52.09515
```

```
dB
```

```
-->
```

Figure 9.10: Normalized Message Signal

```
19 //We know
20 //SNR=((kp^2)*Si*Sx)/(N0*BW);
21
22 SNR=((kp^2)*Si*Sx)/(N0*BW);
23 SNR1=10*log10(SNR);
24 disp("dB", SNR1,"SNR at output: ");
```

Chapter 10

Digital Modulation and Demodulation

Scilab code Exa 10.1 Binary Receiver

```
1 //Page Number: 10.18
2 //Example 10.1
3 clc;
4 //Given
5 Rb=1D+6; //b/s
6 A=5D-3; //V
7 N0=0.5D-11; //W/Hz
8
9 Tb=1/Rb;
10 Eb=(A*A*Tb)/2;
11
12 //(a) ASK
13 //Pe=Q(x)
14 //where
15 xA=sqrt(Eb/N0);
16
17 PeA=(1/2)*erfc(xA/1.414);
```

Scilab 5.4.1 Console

For ASK:

0.0568959

For PSK:

0.0126626

For FSK:

0.0568959

-->

Figure 10.1: Binary Receiver


```

18 disp(PeA, 'For ASK: ');
19
20 //(b) PSK
21 //Pe=Q(x)
22 //where
23 xP=sqrt((2*Eb)/N0);
24
25 PeP=(1/2)*erfc(xP/1.414);
26 disp(PeP, 'For PSK: ');
27
28 //(c) FSK
29 //Pe=Q(x)
30 //where
31 xF=sqrt(Eb/N0);
32
33 PeF=(1/2)*erfc(xF/1.414);
34 disp(PeF, 'For FSK: ');

```

Scilab code Exa 10.2 BPSK Reciever

```

1 //Page Number: 10.19
2 //Example 10.2
3 clc;
4 //Given
5 Rb=1D+6; //b/s
6
7 //(a)
8 Aa=1D-3; //V
9 N0=1D-11; //W/Hz
10
11 Tb=1/Rb;
12 Eba=(Aa*Aa*Tb)/2;
13

```

```

For Average bit error probability:

    0.3758967

Average Power:

    9.171D-09

V

For Average bit error probability for bipolar antipodal signals:

    0.00001

-->

```

Figure 10.2: BPSK Receiver

```

14 //Pe=Q(z)
15 //where
16 za=sqrt((2*Eba)/N0);
17
18 Peb=(1/2)*erfc(za/1.414);
19 disp(Peb,'For Average bit error probability:');
20
21 //(b) Maintain Pb=2D-3
22 //From table
23 zb=2.9;
24 Ebb=((zb^2)*N0)/2;
25 P=Ebb*Tb;
26 Ab=sqrt((2*P));
27 disp('V',Ab,'Average Power:');
28
29 //(c)
30 Ac=100; //V
31 Rbc=1D+5; //p/s

```

```

32 N01=1D-2;; //W/Hz
33 Tbc=1/Rbc;
34 Ebc=(Ac*Ac*Tbc);
35
36 zc=sqrt(((2*Ebc)/N01));
37
38 Pec=(1/2)*erfc(zc/1.414);
39 //Pec=0.0000039
40 //nearly 10^-5
41 disp(10^-5, 'For Average bit error probability for
    bipolar antipodal signals:');

```

Scilab code Exa 10.6 Satellite Communication

```

1 //Page Number: 10.23
2 //Example 10.6
3 clc;
4 //Given
5 p=0.1; //dB
6 p1=10^(-p/10);
7 p2=sqrt(p1);
8 t=acosd(p2);
9 a=round(t);
10 disp('degrees',a,'System cannot tolerate more than:');

```

Scilab code Exa 10.11 Baseband Binary Transmission

```

1 //Page Number: 10.28

```

Scilab 5.4.1 Console

```
System cannot tolerate more than:
```

```
  9.
```

```
degrees
```

```
-->|
```

Figure 10.3: Satellite Communication

Scilab 5.4.1 Console

Required value of A:

0.6010408

Bandwidth:

100000.

Hz

-->

Figure 10.4: Baseband Binary Transmission

```

2 //Example 10.11
3 clc;
4 //Given
5 Rb=1D+6; //b/s
6 T=1D-5;
7 N0=2*0.5D-7; //W/Hz
8
9 //From table for  $Q(z)=10^{-5}$ 
10 z=4.25;
11
12 //As  $z=\sqrt{A*A*T/2*N0}$ 
13 x=((z^2)*2*N0)/T;
14 A=sqrt(x);
15 disp(A,'Required value of A:');
16
17 //(b) Bandwidth
18 //B=1/(2*(T/2))
19 //Therefore B=1/T
20 B=(1/T);
21 disp('Hz',B,'Bandwidth:');

```

Scilab code Exa 10.17 On Off Binary System

```

1 //Page Number: 10.33
2 //Example 10.17
3 clc;
4 //Given
5 A=0.2D-3; //V
6 T=2D-6; //s
7 n=2*1D-15; //W/Hz
8
9
10 //Pe=Q(z)

```

```
Scilab 5.4.1 Console

Error probability:
0.0007814

-->
```

Figure 10.5: On Off Binary System

```
Scilab 5.4.1 Console

Error probability:

    0.0000038

-->
```

Figure 10.6: Binary System

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

Scilab code Exa 10.18 Binary System

```
1 //Page Number: 10.34
2 //Example 10.18
3 clc;
4 //Given
5 A=0.2D-3; //V
6 A1=A/1.414;
7
8 T=2D-6; //s
9 n=2*1D-15; //W/Hz
10
11
12 //Pe=Q(z)
13 //where
14 x=(A1*A1*T)/(n);
15 z=sqrt(x);
16 Pe=(1/2)*erfc(z/1.414);
17 disp(Pe, 'Error probability:');
```

Scilab code Exa 10.19 Binary Data Transmission

```
1 //Page Number: 10.35
2 //Example 10.19
3 clc;
4 //Given
5 d12sqr=8;
6 N0=2*0.5; //W/Hz
7
8 //(c)
9 //As for two equiprobables
10 //Pe=Q(z)
11 //where z=sqrt(d12^2)/sqrt(2*N0)
```

Scilab 5.4.1 Console

Probability error:

0.0227338

-->|

Figure 10.7: Binary Data Transmission

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

Scilab code Exa 10.20 Binary Communication

```
1 //Page Number: 10.36
2 //Example 10.20
3 clc;
4 //Given
5 c=sqrt(2);
6 A=sqrt(5);
7 N0=1; //W/Hz
8
9 d12sqr=4*A*A;
10
11 //As for two equiprobables
12 //Pe=Q(z)
13 //where z=sqrt(d12^2)/(2*N0)
14 z=sqrt((d12sqr)/(sqrt(2*N0)));
15 Pe=(1/2)*erfc(z/1.414);
16 disp(Pe,'Probabilty error:')
```

Scilab code Exa 10.24 Voice Transmission BPSK

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

Scilab 5.4.1 Console

Probability error:

0.0000846

-->

Figure 10.8: Binary Communication

Scilab 5.4.1 Console

Number of constellation points

256.

Bandwidth efficiency

8.

bps/hz

-->

Figure 10.9: Voice Transmission BPSK

```
Probabilty of bit error:  
  
0.2701151  
  
Probabilty error for fs=50Hz:  
  
0.0062031  
  
-->
```

Figure 10.10: Coheret Matched Reciever

```
5 //Given  
6 Rs=2400; //bps  
7 Rb=19.2D+3; //bps  
8  
9 //As Rs=Rb/log2M  
10 M=2^(Rb/Rs);  
11 disp(M,'Number of constellation points');  
12  
13 //(b) Bandwidth efficiency  
14 BT=2400; //Symbols/second  
15 n=Rb/BT;  
16 disp('bps/hz',n,'Bandwidth efficiency');
```

Scilab code Exa 10.25 Coherent Matched Receiver

```
1 //Page Number: 10.41
2 //Example 10.25
3 clc;
4 //Given
5 A1=0.5;
6 A2=0.5;
7 T=0.01; //sec
8 N0=2*0.0001; //W/Hz
9 f=50; //Hz
10
11 //(a) Probability of bit error
12 Es1=(A1^2*T)/2;
13 Es2=(A2^2*T)/2;
14
15 Eb=(Es1+Es2)/2;
16 //As  $PE=Q\sqrt{(E_p+E_q-2E_{pq})/2N_0}$ 
17 //In this case  $E_p=E_q=E_b$ 
18 //Therefore  $PE=Q\sqrt{(E_b(1-p))/N_0}$ 
19 //where  $p=E_{pq}/E_b$ 
20
21 //p=(1/Eb)*integrate('0.5*cos(2000*%pi*t)*0.5*cos
    (2020*%pi*t)', 't', 0, T);
22 //We get
23 p=0.94;
24 q=1-p;
25 //As  $Pe=Q(z)$ 
26 //where  $z=\sqrt{(Eb/N_0)}$ 
27 z=sqrt((Eb*q)/N0);
28 Pe=(1/2)*erfc(z/1.414);
29 disp(Pe, 'Probability of bit error:')
30
31 //(b)
```

```

32 //Given
33 fs=50; //Hz
34 //or fs=1/2T where T=0.001
35 //This implies y=tone spacing will be orthogonal
36 //Therefore p=0
37
38 //As Pe=Q(z)
39 //where z=sqrt(Eb/N0)
40 zb=sqrt(Eb/N0);
41 PB=(1/2)*erfc(zb/1.414);
42 disp(PB,'Probabilty error for fs=50Hz:')

```

Scilab code Exa 10.27 M level PSK

```

1 //Page Number: 10.43
2 //Example 10.27
3 clc;
4 //Given
5 Rb=4.8D+3; //b/s
6 bw=3.2D+3; //Hz
7
8 //BPSK can give maximum spectral efficiency of 1bps/
  Hz, therefore not suitable
9 //QPSK can give twice spectral efficiency ,2bps/Hz,
  therefore
10 qpsk=2*bw;
11 //PSK can give thrice spectral efficiency ,3bps/Hz,
  therefore
12 psk=3*bw;
13
14 //QPSK is most suitable
15 Rs=Rb/2;
16 //Roll off Factor

```


Scilab 5.4.1 Console

```
Roll off factor:
```

```
33.333333
```

```
⊞
```

```
-->
```

Figure 10.11: M level PSK

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

Chapter 11

Information Theory and Source Coding

Scilab code Exa 11.2 DMS X with four symbols

```
1 //Page Number: 11.12
2 //Example 11.2
3 clc;
4 //Given
5 //Probabilities of four symbols
6 Px=[0.4 0.3 0.2 0.1];
7
8 //(a) H(X)
9 //As  $H(X) = -\sum (P(x_i) \log_2 P(x_i))$ 
10 //Where  $i=0$  to  $n$ ;
11 HofX=0;
12 for i=1:4
13     HofX=HofX+(Px(i)*log2(Px(i)));
14 end
15 disp('b/symbol',-HofX,'H(X): ');
16
17 //(b) Amount of information in  $x_1x_2x_1x_3$  and  $x_4x_3x_3x_2$ 
```

H (X) :

1.8464393

b/symbol

Ix1x2x1x3:

6.7027499

b/symbol

Ix4x3x3x2:

9.7027499

b/symbol

-->

Figure 11.1: DMS X with four symbols

```

18 Px1x2x1x3=Px(1)*Px(2)*Px(1)*Px(3);
19 Ix1x2x1x3=-log2(Px1x2x1x3);
20 disp('b/symbol',Ix1x2x1x3,'Ix1x2x1x3:');
21
22 Px4x3x3x2=Px(4)*Px(3)*Px(3)*Px(2);
23 Ix4x3x3x2=-log2(Px4x3x3x2);
24 disp('b/symbol',Ix4x3x3x2,'Ix4x3x3x2:');

```

Scilab code Exa 11.3 Binary Memoryless Source

```

1 //Page Number: 11.13
2 //Example 11.3
3 clc;
4 //As H(X) is maximum when
5 //Px1=Px2=1/2
6 Px=[0.5 0.5];
7
8 //As H(X)=-Sum of [P(xi)log2P(xi)]
9 //Where i=0 to n;
10 HofX=0;
11 for i=1:2
12     HofX=HofX+(Px(i)*log2(Px(i)));
13 end
14 disp('b/symbol',-HofX,'Maximum H(X):');

```

Scilab code Exa 11.5 Black and white TV picture

```

1 //Page Number: 11.15
2 //Example 11.5

```

```
Scilab 5.4.1 Console

Maximum H (X) :

    1.

b/symbol

-->|
```

Figure 11.2: Binary Memoryless Source

```
Scilab 5.4.1 Console  
  
Average rate of information conveyed:  
  
    2.560D+08  
  
b/symbol  
-->|
```

Figure 11.3: Black and white TV picture

```

3  clc;
4  //Given
5  //Picture elements
6  pe=2D+6;
7  //Brightness levels
8  l=16;
9  //Rate of repeation
10 rr=32; //Per second
11
12
13 //As  $H(X)=-\text{Sum of } [P(x_i) \log_2 P(x_i)]$ 
14 //Where  $i=0$  to  $n$ ;
15 HofX=(-1)*l*[(1/l)*log2(1/l)];
16
17 r=pe*rr;
18
19 //As  $R=r*H(X)$ 
20 R=r*HofX;
21 disp('b/symbol',R,'Average rate of information
        conveyed: ');

```

Scilab code Exa 11.6 Telegraph Source

```

1  //Page Number: 11.15
2  //Example 11.6
3  clc;
4  //Given
5  //Pdot-2*Pdash and Pdot+Pdash=1
6  //Therefore , on solving
7  Pdot=2/3;
8  Pdash=1/3;
9
10 tdot=0.2; //Sec

```


Scilab 5.4.1 Console

Average information rate :

1.7218047

b/symbol

-->

Figure 11.4: Telegraph Source

```

11 tdash=0.6; //Sec
12 tspace=0.2; //Sec
13
14 //Finding H(X)
15 //As H(X)=Sum of [P(xi)log2P(xi)]
16 //Where i=0 to n;
17 HofX=(-1)*[{Pdot*log2(Pdot)}+{Pdash*log2(Pdash)}];
18
19 //Average time per symbol
20 Ts=(Pdot*tdot)+(Pdash*tdash)+tspace;
21
22 //Average Symbol Rate
23 r=1/Ts;
24
25 //Average information rate
26 R=r*HofX;
27 disp('b/symbol',R,'Average information rate :');

```

Scilab code Exa 11.7 Binary Channel

```

1 //Page Number: 11.15
2 //Example 11.7
3 clc;
4
5 //(a) Channel Matrix
6 //Given
7 Py1byx1=0.9;
8 Py2byx1=0.1;
9 Py1byx2=0.2;
10 Py2byx2=0.8;
11 PYbyX=[Py1byx1 Py2byx1;Py1byx2 Py2byx2];
12 disp(PYbyX,'Channel Matrix,P(Y/X):');
13

```

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

```

14 // (b) Py1 and Py2
15 // Given
16 Px1=0.5;
17 Px2=Px1;
18 // As  $P(Y)=P(X)*P(Y/X)$ 
19 PX=[Px1 Px2];
20 PY=PX*PYbyX;
21 disp(PY, 'P(y1) P(y2): ');
22
23 // (c) Joint Probabilities P(x1,y2) and P(x2,y1)
24 // Diagonalizing PX
25 PXd=diag(PX);
26 PXY=PXd*PYbyX;
27 disp(PXY(2,1), PXY(1,2), 'P(x1,y2) P(x2,y1) ');

```

Scilab code Exa 11.8 Binary Channel in cascade

```

1 // Page Number: 11.16
2 // Example 11.8
3 clc;
4 // (a) Channel Matrix
5 // Given
6 PYbyX=[0.9 0.1;0.2 0.8];
7 PZbyY=[0.9 0.1;0.2 0.8];
8
9 // As  $P(Z/X)=P(Y/X)*P(Z/Y)$ 
10 PZbyX=PYbyX*PZbyY;
11 disp(PZbyX, 'Channel Matrix ');
12
13 // (b) Pz1 and Pz2
14 // Given
15 Px1=0.5;
16 Px2=Px1;

```

```
Scilab 5.4.1 Console

Channel Matrix

    0.83    0.17
    0.34    0.66

P(z1) P(z2) :

    0.585    0.415

-->
```

Figure 11.6: Binary Channel in cascade

```

17 //As  $P(Z)=P(X)*P(Z/X)$ 
18
19 //P(X) matrix
20 PX=[Px1 Px2];
21 PZ=PX*PZbyX;
22 disp(PZ, 'P(z1) P(z2): ');

```

Scilab code Exa 11.9 Channel

```

1 //Page Number: 11.17
2 //Example 11.9
3 clc;
4 //Given
5 p=0.2;
6 Px1=0.5;
7 Px2=0.5;
8 //P(X) Matrix
9 PX=[Px1 Px2];
10 //Given
11 PYbyX=[(1-p) p 0;0 p (1-p)];
12 //P(y)=
13 PY=PX*PYbyX;
14 disp(PY, 'P(y1) P(y2) P(y3): ');

```

Scilab code Exa 11.16 BSC

```

1 //Page Number: 11.21
2 //Example 11.16
3 clc;

```

```
Scilab 5.4.1 Console

P (y1)  P (y2)  P (y3) :
      0.4      0.2      0.4

-->
```

Figure 11.7: Channel

Scilab 5.4.1 Console

```
I(X;Y) for a=0.5 and p=0.1:
```

```
0.5310044
```

```
I(X;Y) for a=0.5 and p=0.5:
```

```
0.
```

```
-->
```

Figure 11.8: BSC


```

4 // (b) I(X;Y)
5 // Given
6 a=0.5;
7 p=0.1;
8 // As we know
9 // P(Y)=P(X)*P(Y/X)
10 // We have
11 PX=[a (1-a)];
12 PYbyX=[(1-p) p;p (1-p)];
13 PY=PX*PYbyX;
14
15 // As H(Y)=-Sum of [P(yi) log2 P(yi)]
16 // Where i=0 to n;
17 HofY=0;
18 for i=1:2
19     HofY=HofY+(PY(i)*log2(PY(i)));
20 end
21
22 // For BSC, I(X;Y)=H(Y)+p log2(p)+(1-p) log2(1-p)
23 IXY=-HofY+[(p*log2(p))+((1-p)*log2(1-p))];
24 disp(IXY, 'I(X;Y) for a=0.5 and p=0.1:');
25
26
27 // (c) I1(X;Y)
28 // Given
29 a1=0.5;
30 p1=0.5;
31 // As we know
32 // P(Y)=P(X)*P(Y/X)
33 // We have
34 PX1=[a1 (1-a1)];
35 PYbyX1=[(1-p1) p1;p1 (1-p1)];
36 PY1=PX1*PYbyX1;
37
38 // As H(Y)=-Sum of [P(yi) log2 P(yi)]
39 // Where i=0 to n;
40 HofY1=0;
41 for i=1:2

```

```

42     HofY1=HofY1+(PY1(i)*log2(PY1(i)));
43 end
44
45
46 //For BSC, I(X;Y)=H(Y)+plog2(p)+(1-p)log2(1-p)
47 IXY1=-HofY1+(p1*log2(p1))+((1-p1)*log2(1-p1));
48 disp(IXY1, 'I(X;Y) for a=0.5 and p=0.5: ');

```

Scilab code Exa 11.20 Differential Entropy

```

1 //Page Number: 11.24
2 //Example 11.20
3 clc;
4 //Given
5 //f(x)=1/a for x from 0 to a
6 //0, otherwise
7
8 //We have
9 //H(X)=-integrate [ f(x)*log2f(x) ] dx
10 //Here, f(x)=1/a for limits 0 to a
11 //H(X)=-integrate (1/a)*log2(1/a)dx for 0 to a
12 //H(X)=log2(a)
13
14 //(a) a1=1
15 a1=1;
16 y1=log2(a1);
17 disp(y1, 'For a=1, H(X): ');
18
19 //(b) a2=2
20 a2=2;
21 y2=log2(a2);
22 disp(y2, 'For a=2, H(X): ');
23

```

Scilab 5.4.1 Console

For $a=1$, $H(X)$:

0.

For $a=2$, $H(X)$:

1.

For $a=1/2$, $H(X)$:

- 1.

-->|

Figure 11.9: Differential Entropy

```

24
25 // (c) a3=1/2
26 a3=1/2;
27 y3=log2(a3);
28 disp(y3, 'For a=1/2, H(X): ');

```

Scilab code Exa 11.23 AWGN Channel

```

1 //Page Number: 11.26
2 //Example 11.23
3 clc;
4 //Given
5 B=4D+3; //Hz
6 S=0.1D-3; //W
7 n=2*(1D-12); //W/hz
8
9 N=n*B;
10 SN=S/N;
11 //As Channel Capacity
12 //C=B*(log2(1+(S/N)));
13 C=B*(log2(1+(S/N)));
14 disp('b/s',C, 'Channel Capacity');

```

Scilab code Exa 11.24 DMS X with two symbols

```

1 //Page Number: 11.26
2 //Example 11.24
3 clc;
4

```

```
Scilab 5.4.1 Console
```

```
Channel Capacity
```

```
54439.024
```

```
b/s
```

```
-->|
```

Figure 11.10: AWGN Channel

Information Rate:

80000.

b/s

Channel Capacity:

66582.115

b/s

For error free transmission S/N:

24.065402

dB

Bandwidth for error free transmission:

12015.239

Hz

-->|

Figure 11.11: DMS X with two symbols

```

5 // (a) Information Rate
6 // Given
7 n=1.25; // times
8 l=256; // Levels
9 fM=4D+3; // Hz // Bandwidth
10 Nr=2*fM; // Nyquist Rate
11 r=Nr*n;
12 HofX=log2(1);
13 // Information rate
14 R=r*HofX;
15 disp('b/s',R,'Information Rate:');
16
17 // (b)
18 // As Channel Capacity
19 // C=B*(log2(1+(S/N)));
20 B=10^4; // Hz
21 SNdB=20; // dB
22 SN=10^(SNdB/10);
23 C=B*(log2(1+(SN)));
24 disp('b/s',C,'Channel Capacity:');
25
26 // As R>C, error free transmission isnt possible
27
28 // (c) For error free transmission
29 C1=R;
30 // Therefore S/N
31 SN1={2^(C1/B)}-1;
32 SN1dB=10*(log10(SN1));
33 disp('dB',SN1dB,'For error free transmission S/N:');
34
35 // (d) Bandwidth for error free transmission
36 SN2dB=20; // dB
37 SN2=10^(SN2dB/10);
38 // As Channel Capacity
39 // C=B*(log2(1+(S/N)));
40 B=C1/(log2(1+(SN2)));
41 disp('Hz',B,'Bandwidth for error free transmission:');
    );

```

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

Scilab code Exa 11.25 DMS X

```
1 //Page Number: 11.27
2 //Example 11.25
3 clc;
4 //Given
5 p=0.9;
6 Px=[p (1-p)];
7
8 n=1;
9 //Average Code length
10 //L=Summation(P(xi)ni)
11 L=0;
12 for i=1:2
13     L=L+(Px(i)*n);
14 end
15
16 //As H(X)=-Sum of [P(xi)log2P(xi)]
17 //Where i=0 to n;
18 HofX=0;
19 for i=1:2
20     HofX=HofX+(Px(i)*log2(Px(i)));
21 end
22
23 // Efficiency=H(X)/L
24 n=-HofX/L;
25 np=n*100;
26 disp('%',np,'Code efficiency:');
27
28 //Redundancy
```


Scilab 5.4.1 Console

Code efficiency:

46.899559

␣

Code redundancy:

53.100441

␣

—v|

Figure 11.12: DMS X

```

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

```

Scilab code Exa 11.26 Second order extension of DMS

```

1 //Page Number: 11.28
2 //Example 11.26
3 clc;
4 //Given
5 Pa=[0.81 0.09 0.09 0.01];
6 n=[1 2 3 4];
7
8 //Average Code length
9 //L=Summation(P(xi)ni)
10 L=0;
11 for i=1:4
12     L=L+(Pa(i)*n(i));
13 end
14
15 //Entropy of second order extension
16 //As  $H(X^2)=-\text{Sum of } [P(ai)\log_2P(ai)]$ 
17 //Where i=0 to n;
18 HofX2=0;
19 for i=1:4
20     HofX2=HofX2+(Pa(i)*log2(Pa(i)));
21 end
22 //b/s
23
24 //Efficiency= $H(X^2)/L$ 
25 n=-HofX2/L;
26 np=n*100;
27 disp('%',np,'Code efficiency:');

```

```
Scilab 5.4.1 Console

Code efficiency:

    72.153168

%

Code redundancy:

    27.846832

%

-->|
```

Figure 11.13: Second order extension of DMS

```
28
29 //Redundancy
30 g=1-n;
31 gp=g*100;
32 disp('%',gp,'Code redundancy:');
```

Scilab code Exa 11.27 Kraft inequality

```
1 //Page Number: 11.28
2 //Example 11.27
3 clc;
4 //As Kraft inequality
5 //K=summation(2^(-n))
6 //where i from 0 to 4
7 //As i=1,2,3,4
8 //Given
9
10 //For Code A
11 na=[2 2 2 2];
12 KA=0;
13 for i=1:4
14     KA=KA+(2^(-na(i)));
15 end
16 disp(KA,'For Code A:');
17
18 //For Code B
19 nb=[1 2 2 3];
20 KB=0;
21 for i=1:4
22     KB=KB+(2^(-nb(i)));
23 end
24 disp(KB,'For Code B:');
25
```

For Code A:

1.

For Code B:

1.125

For Code C:

1.

For Code D:

0.875

-->|

Figure 11.14: Kraft inequality

```

26 //For Code C
27 nc=[1 2 3 3];
28 KC=0;
29 for i=1:4
30     KC=KC+(2^(-nc(i)));
31 end
32 disp(KC, 'For Code C: ');
33
34 //For Code D
35 nd=[1 3 3 3];
36 KD=0;
37 for i=1:4
38     KD=KD+(2^(-nd(i)));
39 end
40 disp(KD, 'For Code D: ');
41
42 //All codes except Code B satisfy Kraft inequality

```

Scilab code Exa 11.32 Shannon Fano

```

1 //Page Number: 11.31
2 //Example 11.32
3 clc;
4 //Given
5 Px=[1/2 1/4 1/8 1/8];
6
7 //As  $I(x_i) = -\log_2(P_{xi})$ 
8 for i=1:4
9     Ix(i)=-log2(Px(i));
10    n(i)=Ix(i);
11 end
12
13 //As  $H(X) = -\text{Sum of } [P(x_i) \log_2 P(x_i)]$ 

```

Scilab 5.4.1 Console

```
Code efficiency:
```

```
100.
```

```
⊞
```

```
-->
```

Figure 11.15: Shannon Fano

```

14 //and I(xi)=-log2p(xi)
15 //Where i=0 to n;
16 HofX=0;
17 for i=1:4
18     HofX=HofX+(Px(i)*Ix(i));
19 end
20
21 //Average Code length
22 //L=Summation(P(xi)ni)
23 L=0;
24 for i=1:4
25     L=L+(Px(i)*n(i));
26 end
27
28 //Efficiency=H(X)/L
29 n=HofX/L;
30 np=n*100;
31 disp('%',np,'Code efficiency:');
32
33 //Hence, efficiency is 100%

```

Scilab code Exa 11.33 Shannon Fano and Huffman

```

1 //Page Number: 11.32
2 //Example 11.33
3 clc;
4 //Given
5 //(a)Efficiency of code
6 Px=[0.2 0.2 0.2 0.2 0.2];
7 na=[2 2 2 3 3];
8
9 //As H(X)=-Sum of [P(ai)log2P(ai)]
10 //Where i=0 to n;

```



```
Code efficiency for Shannon code 1:
```

```
96.747004
```

```
⌘
```

```
Code efficiency for Shannon code 2:
```

```
96.747004
```

```
⌘
```

```
Code efficiency for Hauffman code:
```

```
96.747004
```

```
⌘
```

```
-->
```

Figure 11.16: Shannon Fano and Huffman

```

11 HofX=0;
12 for i=1:5
13     HofX=HofX+(Px(i)*log2(Px(i)));
14 end
15
16 //Average Code length
17 //L=Summation(P(xi)ni)
18 La=0;
19 for i=1:5
20     La=La+(Px(i)*na(i));
21 end
22
23 //Efficiency=H(X)/L
24 ea=-HofX/La;
25 npa=ea*100;
26 disp('%',npa,'Code efficiency for Shannon code 1:');
27
28 //(b) Another Shannon Fano Code
29 nb=[2 3 3 2 2];
30
31 //Average Code length
32 //L=Summation(P(xi)ni)
33 Lb=0;
34 for i=1:5
35     Lb=Lb+(Px(i)*nb(i));
36 end
37
38 //Efficiency=H(X)/L
39 eb=-HofX/Lb;
40 npb=eb*100;
41 disp('%',npb,'Code efficiency for Shannon code 2:');
42
43 //(c) Hauffman Code
44 nc=[2 3 3 2 2];
45
46 //Average Code length
47 //L=Summation(P(xi)ni)
48 Lc=0;

```

```
Scilab 5.4.1 Console

Code efficiency for shannon fanon:

    95.544548

%

Code efficiency for hauffman:

    97.716015

%

-->
```

Figure 11.17: DMS X with five symbols

```
49 for i=1:5
50     Lc=Lc+(Px(i)*nc(i));
51 end
52
53 // Efficiency=H(X)/L
54 ec=-HofX/Lc;
55 npc=ec*100;
56 disp('%',npc,'Code efficiency for Hauffman code:');
57
58 //Efficiency of all codes is same
```

Scilab code Exa 11.34 DMS X with five symbols

```
1 //Page Number: 11.33
2 //Example 11.34
3 clc;
4 //Given
5 //(a) For Shannon Fano Code
6 Px=[0.4 0.19 0.16 0.15 0.1];
7 n=[2 2 2 3 3];
8
9 //Average Code length
10 //L=Summation(P(xi)ni)
11 L=0;
12 for i=1:5
13     L=L+(Px(i)*n(i));
14 end
15
16 //As  $H(X)=-\text{Sum of } [P(x_i) \log_2 P(x_i)]$ 
17 //Where i=0 to n;
18 HofX=0;
19 for i=1:5
20     HofX=HofX+(Px(i)*log2(Px(i)));
21 end
22
23 //Efficiency= $H(X)/L$ 
24 n=-HofX/L;
25 np=n*100;
26 disp('%',np,'Code efficiency for shannon fanon:');
27
28 //(b) For Huffman Code
29 nh=[1 3 3 3 3];
30
31 //Average Code length
32 //L=Summation(P(xi)ni)
```

```
33 Lh=0;
34 for i=1:5
35     Lh=Lh+(Px(i)*nh(i));
36 end
37
38 // Efficiency=H(X)/L
39 n1=-HofX/Lh;
40 np1=n1*100;
41 disp('%',np1,'Code efficiency for hauffman:');
```
