

Scilab Textbook Companion for
Principle of Communication Engineering
by A. Singh and A. K. Chhabra¹

Created by
Dipak Tukaram Bagal
Bachelor of Engineering
Electronics Engineering
Mumbai University
College Teacher
Ms. Vijaya Patil
Cross-Checked by
K. V. P. Pradeep

May 25, 2016

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: Principle of Communication Engineering

Author: A. Singh and A. K. Chhabra

Publisher: S. Chand And Co. Ltd., New Delhi

Edition: 1

Year: 2003

ISBN: 81-219-0476-5

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	6
2 Switched Communication Systems	43
3 Modulation	54
5 Radio Transmission System	86
6 Radio Receivers	88
7 Noise	91
8 Transmission Line	103
9 Aerials	124
10 Propagation of Radio Waves	134
11 BroadBand Communication	139
15 Basic Information Theory	141

List of Scilab Codes

Exa 1.1	Signals Ex 1 1	6
Exa 1.2	Signals Ex 1 2	8
Exa 1.3	Signals Ex 1 3	8
Exa 1.4	Signals Ex 1 4	10
Exa 1.5	Signals Ex 1 5	11
Exa 1.6	Signals Ex 1 6	13
Exa 1.7	Signals Ex 1 7	14
Exa 1.8	Signals Ex 1 8	17
Exa 1.9	Signals Ex 1 9	22
Exa 1.10	Signals Ex 1 10	24
Exa 1.11.1	Signals Ex 1 11 Pg 39	25
Exa 1.11.2	Signals Ex 1 11 Pg43	28
Exa 1.12	Signals Ex 1 12	30
Exa 1.13	Signals Ex 1 13	30
Exa 1.14	Signals Ex 1 14	33
Exa 1.15	Signals Ex 1 15	36
Exa 1.16	Signals Ex 1 16	37
Exa 1.17	Signals Ex 1 17	38
Exa 1.18	Signals Ex 1 18	39
Exa 2.2	Chapter 2 Example 2 2	43
Exa 2.3	Chapter 2 Example 2 3	44
Exa 2.4.0	Chapter 2 Example 2 4 Pg126 Top	45
Exa 2.4.1	Chapter 2 Example 2 4 Pg126 Bottom	46
Exa 2.5	Chapter 2 Example 2 5	48
Exa 2.6	Chapter 2 Example 2 6	50
Exa 2.7	Chapter 2 Example 2 7	50
Exa 2.8	Chapter 2 Example 2 8	51
Exa 2.9	Chapter 2 Example 2 9	52

Exa 3.1	Modulation Ex 3 1	54
Exa 3.2	Modulation Ex 3 2	54
Exa 3.3	Modulation Ex 3 3	55
Exa 3.4	Modulation Ex 3 4	56
Exa 3.5	Modulation Ex 3 5	57
Exa 3.6	Modulation Ex 3 6	59
Exa 3.7	Modulation Ex 3 7	60
Exa 3.8	Modulation Ex 3 8	61
Exa 3.9	Modulation Ex 3 9	62
Exa 3.10	Modulation Ex 3 10	64
Exa 3.11	Modulation Ex 3 11	65
Exa 3.12	Modulation Ex 3 12	67
Exa 3.13	Modulation Ex 3 13	68
Exa 3.14	Modulation Ex 3 14	69
Exa 3.15	Modulation Ex 3 15	70
Exa 3.16	Modulation Ex 3 16	72
Exa 3.17	Modulation Ex 3 17	73
Exa 3.18	Modulation Ex 3 18	74
Exa 3.19	Modulation Ex 3 19	75
Exa 3.21	Modulation Ex 3 21	76
Exa 3.25	Modulation Ex 3 25	77
Exa 3.26	Modulation Ex 3 26	78
Exa 3.27	Modulation Ex 3 27	79
Exa 3.28	Modulation Ex 3 28	80
Exa 3.29	Modulation Ex 3 29	82
Exa 3.30	Modulation Ex 3 30	82
Exa 3.31	Modulation Ex 3 31	83
Exa 5.1	RTS Ex 5 1	86
Exa 6.1	Radio Receivers Ex 6 1	88
Exa 6.2	Radio Receivers Ex 6 2	88
Exa 7.2	Noise Ex 7 2	91
Exa 7.3	Noise Ex 7 3	92
Exa 7.4	Noise Ex 7 4	93
Exa 7.5	Noise Ex 7 5	94
Exa 7.7	Noise Ex 7 7	95
Exa 7.8	Noise Ex 7 8	96
Exa 7.9	Noise Ex 7 9	97
Exa 7.10	Noise Ex 7 10	98

Exa 7.11	Noise Ex 7 11	99
Exa 7.12	Noise Ex 7 12	100
Exa 8.1	Transmission Line Ex 8 1	103
Exa 8.2	Transmission Line Ex 8 2	104
Exa 8.3	Transmission Line Ex 8 3	105
Exa 8.4	Transmission Line Ex 8 4	108
Exa 8.5	Transmission Line Ex 8 5	108
Exa 8.6	Transmission Line Ex 8 6	109
Exa 8.15	Transmission Line Ex 8 15 Pg334	111
Exa 8.15.1	Transmission Line Ex 8 15 Pg348	112
Exa 8.16	Transmission Line Ex 8 16	114
Exa 8.17	Transmission Line Ex 8 17	115
Exa 8.18	Transmission Line Ex 8 18	116
Exa 8.19	Transmission Line Ex 8 19	117
Exa 8.20	Transmission Line Ex 8 20	118
Exa 8.22	Transmission Line Ex 8 22	119
Exa 8.23	Transmission Line Ex 8 23	120
Exa 8.24	Transmission Line Ex 8 24	121
Exa 8.25	Transmission Line Ex 8 25	122
Exa 9.1	Aerials Ex 9 1	124
Exa 9.2	Aerials Ex 9 2	124
Exa 9.3	Aerials Ex 9 3	125
Exa 9.4	Aerials Ex 9 4	126
Exa 9.5	Aerials Ex 9 5	127
Exa 9.6	Aerials Ex 9 6	128
Exa 9.7	Aerials Ex 9 7	129
Exa 9.8	Aerials Ex 9 8	130
Exa 9.9	Aerials Ex 9 9	131
Exa 10.1	PoRW Ex 10 1	134
Exa 10.2	PoRW Ex 10 2	134
Exa 10.3	PoRW Ex 10 3	135
Exa 10.4	PoRW Ex 10 4	136
Exa 11.1	BBC Ex 11 1	139
Exa 15.1	BIT Ex 15 1	141
Exa 15.2	BIT Ex 15 2	141
Exa 15.3	BIT Ex 15 3	142

List of Figures

1.1	Signals Ex 1 1	7
1.2	Signals Ex 1 2	7
1.3	Signals Ex 1 3	8
1.4	Signals Ex 1 4	10
1.5	Signals Ex 1 5	12
1.6	Signals Ex 1 6	13
1.7	Signals Ex 1 7	15
1.8	Signals Ex 1 8	17
1.9	Signals Ex 1 9	20
1.10	Signals Ex 1 9	21
1.11	Signals Ex 1 10	25
1.12	Signals Ex 1 11 Pg 39	26
1.13	Signals Ex 1 11 Pg43	28
1.14	Signals Ex 1 12	30
1.15	Signals Ex 1 13	31
1.16	Signals Ex 1 14	33
1.17	Signals Ex 1 14	34
1.18	Signals Ex 1 15	36
1.19	Signals Ex 1 16	37
1.20	Signals Ex 1 17	38
1.21	Signals Ex 1 18	40
2.1	Chapter 2 Example 2 2	44
2.2	Chapter 2 Example 2 3	45
2.3	Chapter 2 Example 2 4 Pg126 Top	46
2.4	Chapter 2 Example 2 4 Pg126 Bottom	47
2.5	Chapter 2 Example 2 5	48
2.6	Chapter 2 Example 2 6	49
2.7	Chapter 2 Example 2 7	51

2.8	Chapter 2 Example 2 8	52
2.9	Chapter 2 Example 2 9	53
3.1	Modulation Ex 3 1	55
3.2	Modulation Ex 3 2	56
3.3	Modulation Ex 3 3	57
3.4	Modulation Ex 3 4	58
3.5	Modulation Ex 3 5	59
3.6	Modulation Ex 3 6	60
3.7	Modulation Ex 3 7	62
3.8	Modulation Ex 3 8	63
3.9	Modulation Ex 3 9	64
3.10	Modulation Ex 3 10	66
3.11	Modulation Ex 3 11	67
3.12	Modulation Ex 3 11	68
3.13	Modulation Ex 3 12	69
3.14	Modulation Ex 3 13	70
3.15	Modulation Ex 3 14	71
3.16	Modulation Ex 3 15	72
3.17	Modulation Ex 3 16	73
3.18	Modulation Ex 3 17	74
3.19	Modulation Ex 3 18	75
3.20	Modulation Ex 3 19	76
3.21	Modulation Ex 3 21	77
3.22	Modulation Ex 3 25	78
3.23	Modulation Ex 3 25	79
3.24	Modulation Ex 3 26	80
3.25	Modulation Ex 3 27	81
3.26	Modulation Ex 3 28	82
3.27	Modulation Ex 3 29	83
3.28	Modulation Ex 3 30	84
3.29	Modulation Ex 3 31	85
5.1	RTS Ex 5 1	87
6.1	Radio Receivers Ex 6 1	89
6.2	Radio Receivers Ex 6 2	90
7.1	Noise Ex 7 2	92

7.2	Noise Ex 7 3	93
7.3	Noise Ex 7 4	94
7.4	Noise Ex 7 5	95
7.5	Noise Ex 7 7	97
7.6	Noise Ex 7 8	98
7.7	Noise Ex 7 9	99
7.8	Noise Ex 7 10	100
7.9	Noise Ex 7 11	101
7.10	Noise Ex 7 12	102
8.1	Transmission Line Ex 8 1	104
8.2	Transmission Line Ex 8 2	105
8.3	Transmission Line Ex 8 3	106
8.4	Transmission Line Ex 8 4	107
8.5	Transmission Line Ex 8 5	109
8.6	Transmission Line Ex 8 6	110
8.7	Transmission Line Ex 8 15 Pg334	111
8.8	Transmission Line Ex 8 15 Pg348	113
8.9	Transmission Line Ex 8 16	114
8.10	Transmission Line Ex 8 17	115
8.11	Transmission Line Ex 8 18	116
8.12	Transmission Line Ex 8 19	117
8.13	Transmission Line Ex 8 20	119
8.14	Transmission Line Ex 8 22	120
8.15	Transmission Line Ex 8 23	121
8.16	Transmission Line Ex 8 24	122
8.17	Transmission Line Ex 8 25	123
9.1	Aerials Ex 9 1	125
9.2	Aerials Ex 9 2	126
9.3	Aerials Ex 9 3	127
9.4	Aerials Ex 9 4	128
9.5	Aerials Ex 9 5	129
9.6	Aerials Ex 9 6	130
9.7	Aerials Ex 9 7	131
9.8	Aerials Ex 9 8	132
9.9	Aerials Ex 9 9	133

10.1 PoRW Ex 10 1	135
10.2 PoRW Ex 10 2	136
10.3 PoRW Ex 10 3	137
10.4 PoRW Ex 10 4	138
11.1 BBC Ex 11 1	140
15.1 BIT Ex 15 1	142
15.2 BIT Ex 15 2	143
15.3 BIT Ex 15 3	144

Chapter 1

Signals

Scilab code Exa 1.1 Signals Ex 1 1

```
1  clc
2  //Chapter1: Signals
3  //Example1, page no12
4  //Given:
5  n=round(rand())//any integers
6  m=round(rand())//any integers
7  wo=2*(n+m)*%pi//Angular Freq
8  t=0:0.01:2*%pi/wo
9  to=0,t1=2*%pi/wo
10 C= integrate('sin(n*wo*t)*cos(m*wo*t)', 't', to, t1) //
    integrating sin(n*wo*t)*cos(m*wo*t) function
11 mprintf('The value of the above integral is:C=%d\n
    Since C=%d, the two functions: \n f(t)=sin(n*wo*
    t)\n g(t)=cos(n*wo*t) are Orthogonal', C, C)
```

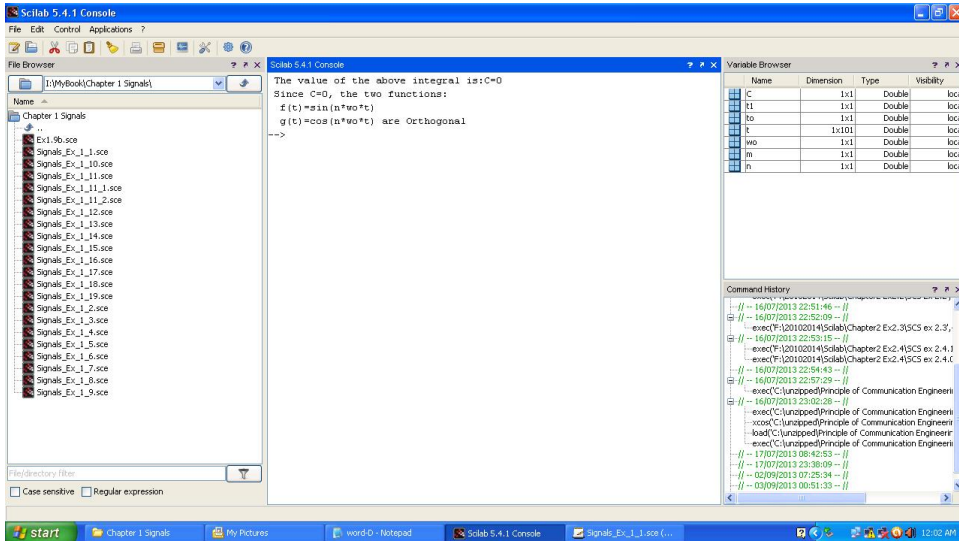


Figure 1.1: Signals Ex 1 1

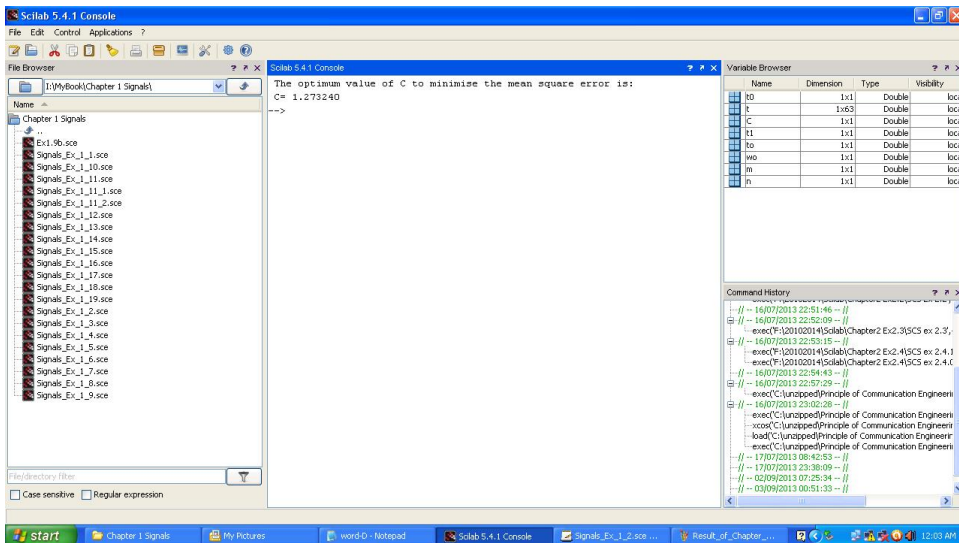


Figure 1.2: Signals Ex 1 2

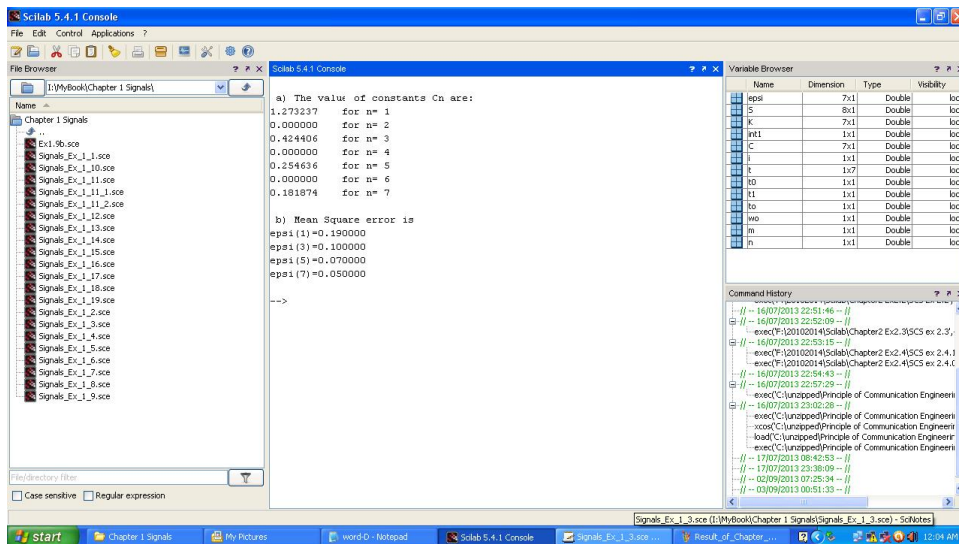


Figure 1.3: Signals Ex 1 3

Scilab code Exa 1.2 Signals Ex 1 2

```

1  clc
2  //Chapter1: Signals
3  //Example1, page no 12
4  //Given:
5  // Curve on page no 9.... fig 1.6
6  t=0:0.1:2*%pi, t0=0, t1=2*%pi
7  C=((integrate('sin(t)', 't', t0, t1/2)-integrate('sin(t)
   ')', 't', t1/2, t1))/integrate('(sin(t))^2', 't', t0, t1
   ))
8
9  mprintf('The optimum value of C to minimise the mean
   square error is:\n C= %f', C)

```

Scilab code Exa 1.3 Signals Ex 1 3

```

1  clc
2  //Chapter1: Signals
3  //Example2, page no12
4  //Given:
5  //a // Referance Figure on page no 9.. (1.6d)
6  t=0:2*3.14,t0=0,t1=2*3.14
7  disp('a) The value of constants Cn are:')
8  for i=1:7
9      C(i)=((integrate('sin(i*t)', 't', t0, t1/2)-
              integrate('sin(i*t)', 't', t1/2, t1))/integrate(
              '(sin(i*t))^2', 't', t0, t1))
10     if C(i)<=0.01 then C(i)=0
11     end
12     mprintf('%f    for n= %d\n',C(i),i)
13 end
14 //b Mean Square error
15
16 int1=integrate('(1)^2', 't', t0, t1)
17 for n=1:7
18     if modulo(n,2)==0 then
19         C(n)=0
20     else
21         C(n)=4/(n*pi)
22     end
23 end
24 for n=1:7
25
26     K(n)=integrate('(sin(n*t))^2', 't', t0, t1)
27
28 end
29 K(n)=pi
30
31 for n=1:7
32 S(1)=0
33     S(n+1)=S(n)+(((C(n))^2)*K(n))
34
35 end
36 //Mean Square error

```

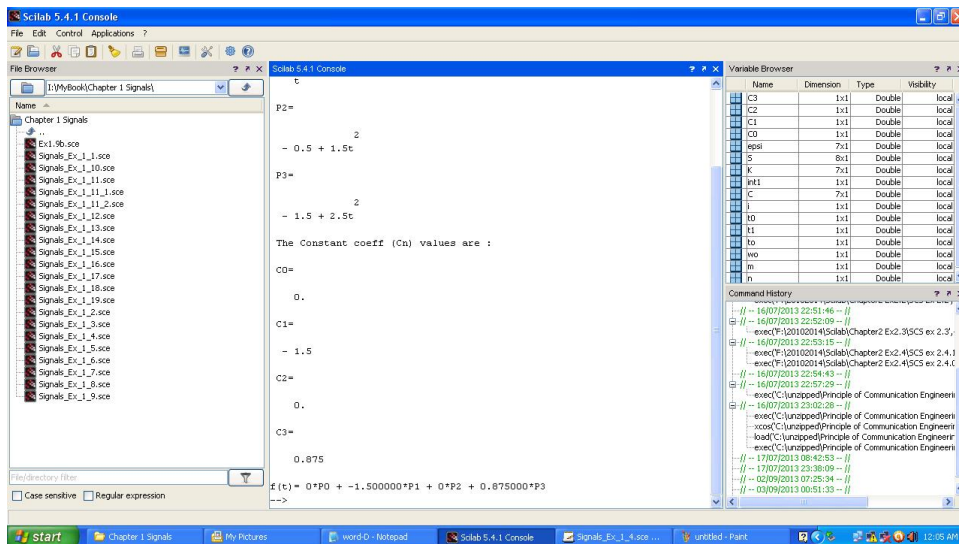


Figure 1.4: Signals Ex 1 4

```

37 for n=1:7
38     epsi(n)=(1/(t1-t0)*(int1-S(n+1)))
39 end
40 disp('b) Mean Square error is ')
41 for n=1:2:7
42     mprintf('epsi (%d)=%f\n',n,round(100*epsi(n))/100)
43
44 end

```

Scilab code Exa 1.4 Signals Ex 1 4

```

1 clc
2 //Chapter1: Signals
3 //Example1, page no12
4 //Given:
5 t=-1:0.01:1, t0=-1, t1=1

```



```

6 // Legendre Polynomial
7 t=poly(0,"t")
8 P0=poly(1,"t","c")
9 P1=poly([0,1],"t","c")
10 P2=poly([-0.5,0,1.5],"t","c")
11 P3=poly([-1.5,0,2.5,0],"t","c")
12 disp(P3,'P3=',P2,'P2=',P1,'P1=',P0,'P0=')
13 //The Constant coeff (Cn)
14 C0=0.5*(integrate('1','t',-1,0)+integrate('-1','t',
    ,0,1))
15 C1=1.5*(integrate('t','t',-1,0)+integrate('-t','t',
    ,0,1))
16 C2=2.5*(integrate('(1.5*t^2)-0.5','t',-1,0)+
    integrate('-(1.5*t^2)+0.5','t',0,1))
17 C3=3.5*(integrate('(2.5*t^3)-(1.5*t)','t',-1,0)+
    integrate('-(2.5*t^3)+(1.5*t)','t',0,1))
18 disp('The Constant coeff (Cn) values are :')
19 disp(C3,'C3=',C2,'C2=',C1,'C1=',C0,'C0=')
20 mprintf('\nf(t)= %d*%s + %f*%s + %d*%s + %f*%s',C0,"
    P0",C1,"P1",C2,"P2",C3,"P3")

```

Scilab code Exa 1.5 Signals Ex 1 5

```

1 clc
2 //Chapter 1 Signals
3 //Example 1.5, page no 19
4 //given
5 T=1,t0=0,wo=2*%pi
6 P=1
7 t=0:0.001:1
8 f=P*t
9 //The trigonometric Fourier series coeff for given
    function

```

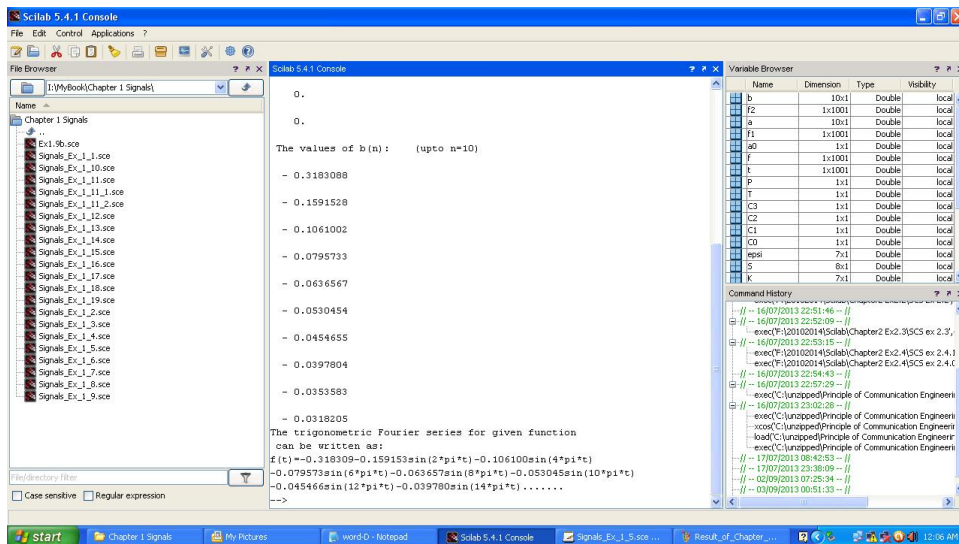


Figure 1.5: Signals Ex 1 5

```

10 a0=(1/T)*inttrap(t,f)
11
12 for n=1:10
13     f1=(P*t).*cos(wo*n*t)
14     a(n)=(2/T)*inttrap(t,f1)
15     if a(n)<0.01 then
16         a(n)=0
17     end
18 end
19 for n=1:10
20     f2=(P*t).*sin(2*%pi*(1/T)*n*t)
21     b(n)=(2/T)*inttrap(t,f2)
22 end
23 // Displaying trigonometric Fourier series coeff
24 mprintf('The value of a0 is: %f\n',a0)
25 disp('The values of a(n): (upto n=10)')
26 for n=1:10
27     disp(a(n))
28 end
29 disp('The values of b(n): (upto n=10)')
  
```

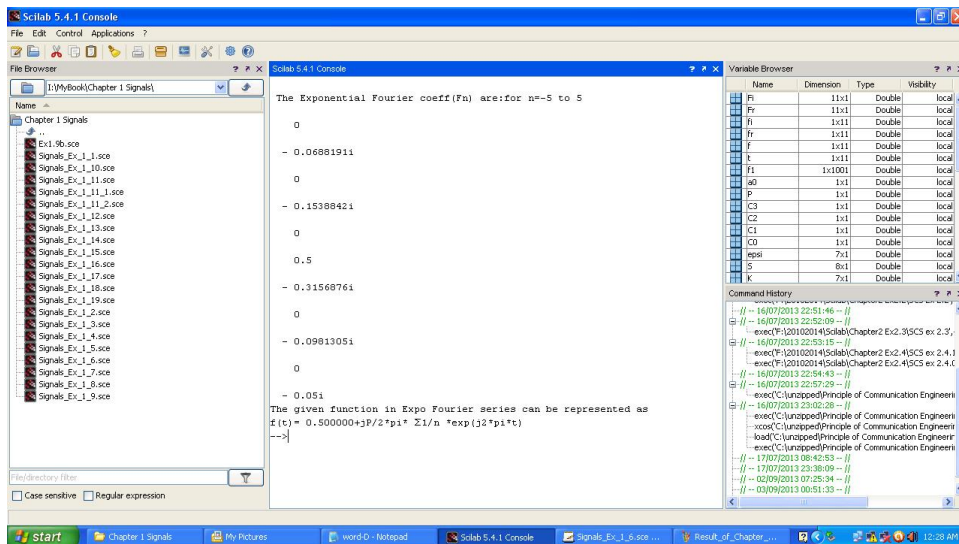


Figure 1.6: Signals Ex 1 6

```

30 for n=1:10
31     disp(b(n))
32 end
33 mprintf('The trigonometric Fourier series for given
function\n can be written as:\n')
34 mprintf(' f(t)=%f%fsin(2*pi*t)%fsin(4*pi*t)\n%fsin(6*
pi*t)%fsin(8*pi*t)%fsin(10*pi*t)\n%fsin(12*pi*t)
%fsin(14*pi*t) . . . . . ',b(1),b(2),b(3),b(4),b(5),b
(6),b(7),b(8))

```

Scilab code Exa 1.6 Signals Ex 1 6

```

1 clc
2 //Chapter 1 Signals
3 //Example 1.6, page no 21
4 //given

```

```

5
6 t0=1,T=1,w0=2*3.14/T,P=1
7 t=0:0.1:1
8 f=P*t// function f(t)=P*t, 0<t<1
9 a=1
10 disp('The Exponential Fourier coeff(Fn) are:for n=-5
      to 5')
11 for n=-5:5// Calculating the fourier coeff
12     fr=f.*cos(%pi*n*t/T)
13     Fr(a)=inttrap(t,fr)
14     fi=f.*sin(%pi*n*t/T)
15     Fi(a)=inttrap(t,fi)
16     if Fr(a)<0.01 then Fr(a)=0
17     end
18     if Fi(a)<0.01 then
19         Fi(a)=0
20     end
21     disp(Fr(a)-%i*Fi(a))
22     a=a+1
23 end
24 mprintf('The given function in Expo Fourier series
      can be represented as \n')
25 mprintf('f(t)= %f+jP/2*pi*      1 /n *exp(j2*pi*t)',P
      /2)

```

Scilab code Exa 1.7 Signals Ex 1 7

```

1 clc
2 close
3 clear
4 //Chapter 1 Signals
5 //Example 1.7, page no 22
6 //given

```

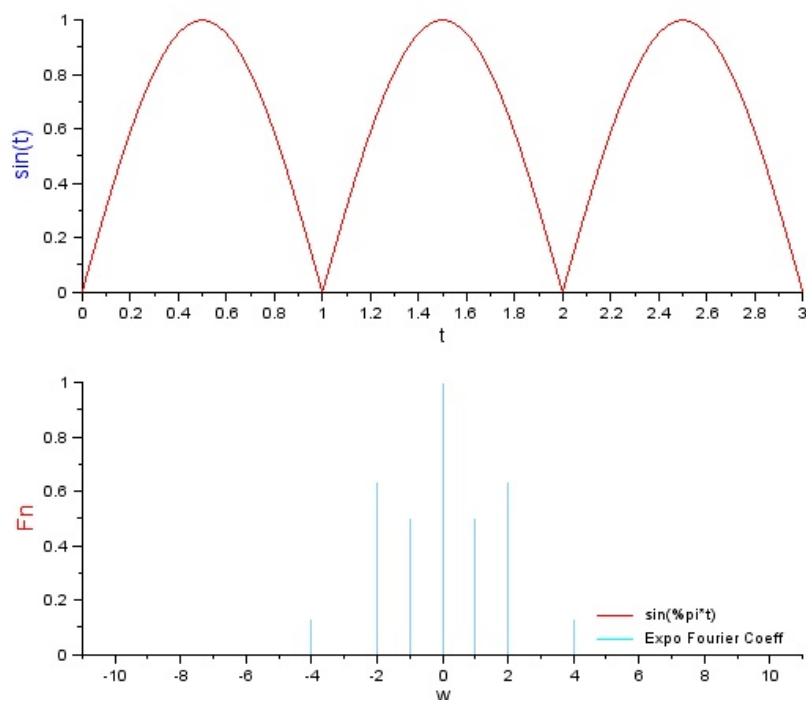


Figure 1.7: Signals Ex 1 7

```

7 V=1
8 t0=1,T=1,w0=2*3.14/T,P=1
9 t=0:0.01:3
10 f=V*abs(sin(%pi*t))
11 //The Expo fourier series coeff
12 disp('The Expo fourier series coeff are: for n=-5 to
      5')
13 a=1
14 for n=-5:5
15     fr=f.*cos(%pi*n*t/T)
16     Fr(a)=inttrap(t,fr)
17     fi=f.*sin(%pi*n*t/T)
18     Fi(a)=inttrap(t,fi)
19
20     mag(a)=abs(Fr(a)+%i*Fi(a))
21
22     disp(Fr(a)-(%i*Fi(a)))
23     x(1 ,size(t,2))=0
24     x=x+((Fr(a))-%i*Fi(a)).*(cos(%pi*n*t/T)+%i*sin(
      %pi*n*t/T))
25     a=a+1
26 end
27 mprintf('The given function in Expo Fourier series
      can be represented as \n')
28 mprintf('f(t)= 2V/pi -2V*exp(j2*pi*t)/3*pi -2V*exp(
      j2*pi*t)/15*pi\n          -2V*exp(j2*pi*t)/35*pi ... \n
      n          -2V*exp(-j2*pi*t)/3*pi -2V*exp(-j2*pi*t)
      /15*pi ... ')
29 n=-5:5
30 subplot(2,1,1),plot2d(t,f,style=5) // Rectified sine
      function Plot
31 xlabel("t", "fontsize", 2);
32 ylabel("sin(t)", "fontsize", 3, "color", "blue");
33 subplot(2,1,2),plot2d3(n,mag,12,rect=[-11,0,11,1],
      style=4) //Plot of the magnitude of the Fourier
      coeff
34 xlabel("w", "fontsize", 2);
35 ylabel("Fn", "fontsize", 3, "color", "red");

```

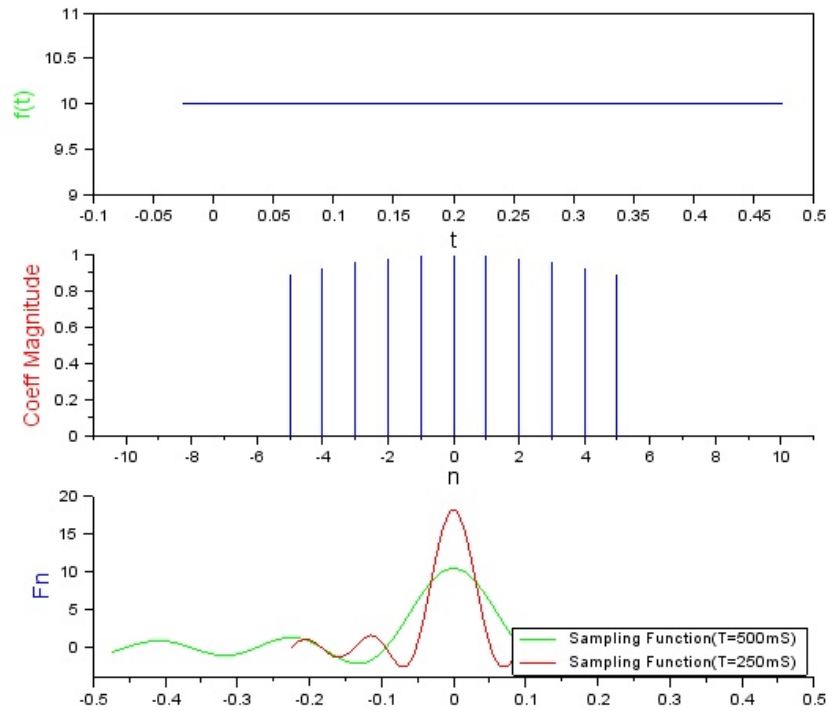


Figure 1.8: Signals Ex 1 8

```
36 legends([" sin (%pi*t)"; "Expo Fourier Coeff"], [5,4],
    with_box=%f, opt="lr" )
```

Scilab code Exa 1.8 Signals Ex 1 8

```
1 clc
2 clear
3 close
4 //Chapter 1 Signals
5 //Example 1.8, page no 24
6 //given
```

```

7
8 T=500e-3,w0=2*pi/T,d=50e-3,A=10
9 t=-d/2:0.01:T-d/2
10 t1=-d/2:0.01:d/2
11 f1=A
12 t2=d/2:0.01:T-(d/2)
13 f2=0
14 a=1
15 disp('The fourier series coeff Fn are:')
16 for n=-5:5
17     if n==0 then
18         Fr(a)=1,Fi(a)=0
19     else
20         fa1=f1.*cos(%pi*n*t1/T)
21         fa2=f2.*cos(%pi*n*t2/T)
22         fb1=f1.*sin(%pi*n*t1/T)
23         fb2=f2.*sin(%pi*n*t2/T)
24     end
25     Fr(a)=1/T*(inttrap(t1,fa1)+inttrap(t2,fa2))
26     Fi(a)=1/T*(inttrap(t1,fb1)+inttrap(t2,fb2))
27     mag(a)=abs(Fr(a)+%i*Fi(a))
28     // disp(mag(a))
29     disp(Fr(a)-%i*Fi(a))
30     x(1,size(t,2))=0
31     x=x+((Fr(a))-%i*Fi(a)).*(cos(%pi*n*t/T)+%i*sin(
        %pi*n*t/T))
32     a=a+1
33
34 end
35 n=-5:5
36 subplot(3,1,1),plot(t,f1)
37 xlabel("t", "fontsize", 3);
38 ylabel("f(t)", "fontsize", 3, "color", "green");
39 subplot(3,1,2),plot2d3(n,mag,2,rect=[-11,0,11,1]) //
    expo fourier series coeff
40 xlabel("n", "fontsize", 3);
41 ylabel("Coeff Magnitude", "fontsize", 3, "color", "
    red");

```



```

42 subplot(3,1,3),plot2d(t,x,style=3),plot2d(-t,x,style
    =3) // one sided spectrum with T=500ms
43 xlabel("w", "fontsize", 3);
44 ylabel("Fn", "fontsize", 3, "color", "blue");
45 legends(['Sampling Function(T=500mS)'; 'Sampling
    Function(T=250mS)'],[3,5],opt="lr")
46 T1=T/2
47 t=-d/2:0.01:T1-d/2
48 t1=-d/2:0.01:d/2
49 f1=A
50 t2=d/2:0.01:T1-(d/2)
51 f2=0
52 //The Expo fourier series coeff
53 for n=-5:5
54     if n==0 then
55         Fr1(a)=1,Fi1(a)=0
56     else
57         fr1=f1.*cos(%pi*n*t1/T1)
58         fr2=f2.*cos(%pi*n*t2/T1)
59         fi1=f1.*sin(%pi*n*t1/T1)
60         fi2=f2.*sin(%pi*n*t2/T1)
61     end
62     Fr1(a)=1/T1*(inttrap(t1,fr1)+inttrap(t2,fr2))
63     Fi1(a)=1/T1*(inttrap(t1,fi1)+inttrap(t2,fi2))
64     mag(a)=abs(Fr1(a)+%i*Fi1(a))
65     disp(Fr1(a)-%i*Fi1(a))
66     y(1,size(t,2))=0
67     y=y+((Fr1(a))-%i*Fi1(a)).*(cos(%pi*n*t/T1)+%i*
        sin(%pi*n*t/T1))
68     a=a+1
69 end
70 plot2d(t,y, style=5),plot2d(-t,y,style=5) // double
    sided spectrum with T=250ms

```

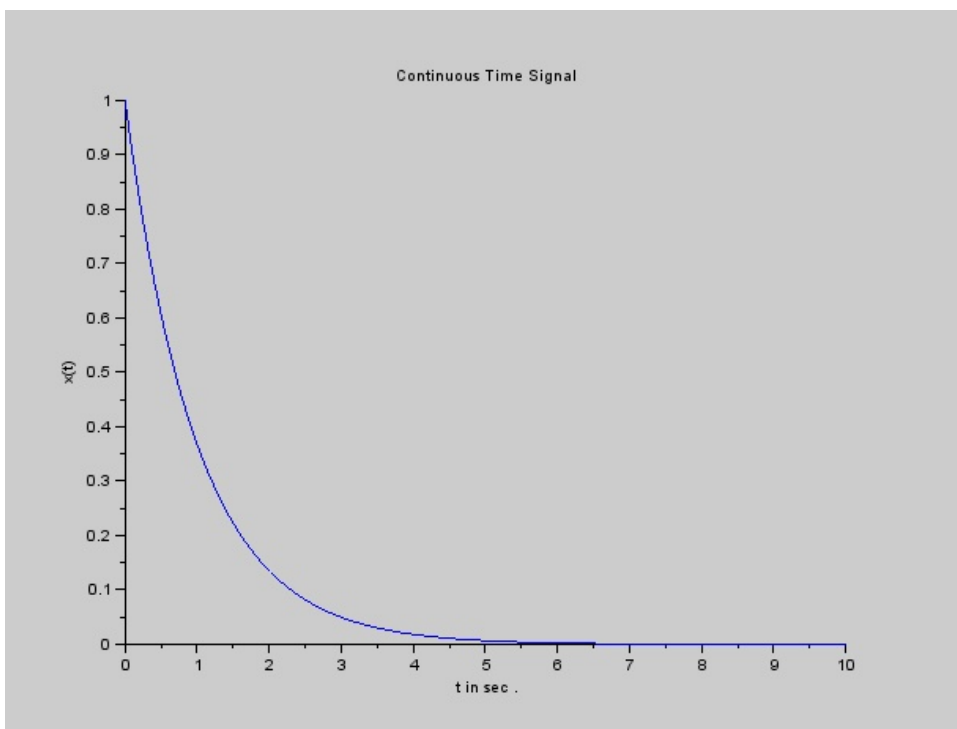


Figure 1.9: Signals Ex 1 9

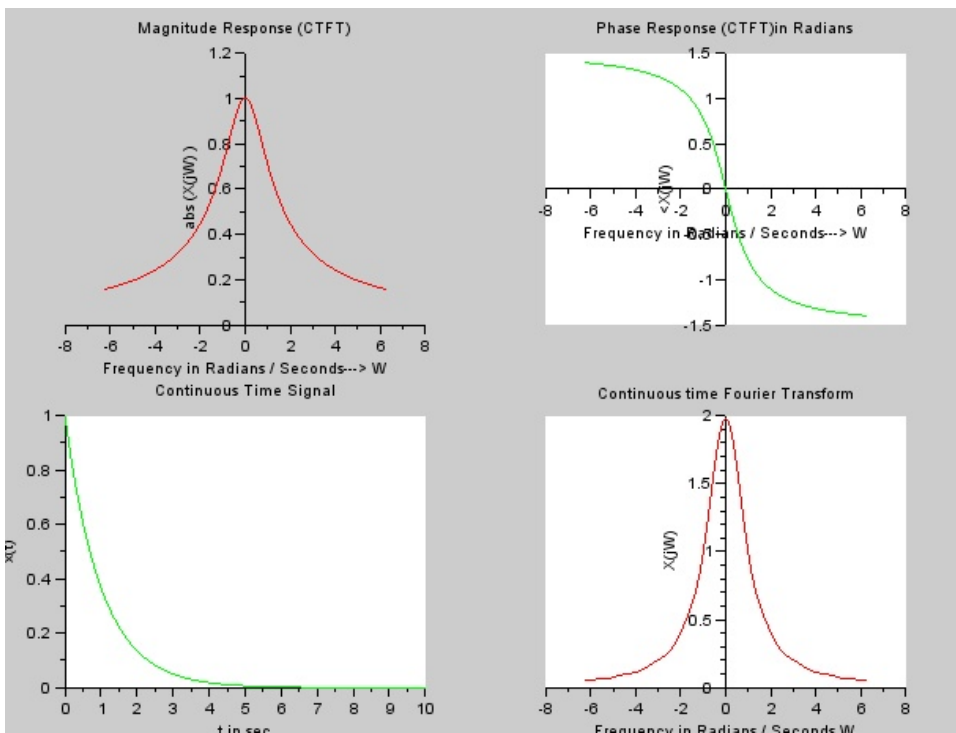


Figure 1.10: Signals Ex 1 9

Scilab code Exa 1.9 Signals Ex 1 9

```
1  clc; clear; close;
2  //Chapter1: Signals
3  //Example 1.9a, page no12
4  //Given:
5  // Analog Signal
6  A =1; // Amplitude
7  Dt = 0.005;
8  t = 0: Dt :10;
9  xt = exp(-A*t);
10 // Continuous time Fourier Transform
11 Wmax =2*%pi*1; // Analog Frequency = 1Hz
12 K = 4;
13 k = 0:(K/1000):K;
14 W = k* Wmax /K;
15 XW = xt*exp(-sqrt(-1)*t'*W)*Dt
16 XW_Mag =abs(XW);
17 W = [-mtlbfliplr(W),W(2:1001)]; // Omega from -Wmax
    to Wmax
18 XW_Mag=[mtlbfliplr(XW_Mag ),XW_Mag(2:1001)];
19 [XW_Phase ,db] = phasemag (XW);
20 XW_Phase = [-mtlbfliplr( XW_Phase ),XW_Phase
    (2:1001)];
21 // Plotting Continuous Time Signal
22 figure
23 a=gca();
24 a.y_location = "origin";
25 plot(t,xt);
26 xlabel( 't in sec .');
27 ylabel( ' x(t) ')
28 title( ' Continuous Time Signal ' )
29 figure
30 // Plotting Magnitude Response of CTS
```

```

31 subplot (2 ,2 ,1);
32 a = gca ();
33 a.y_location = "origin";
34 plot2d(W, XW_Mag,style=5 );
35 xlabel ( ' Frequency in Radians / Seconds——> W' );
36 ylabel ( ' abs (X(jW) ) ' )
37 title ( 'Magnitude Response (CTFT) ' )
38 // Plotting Phase Reponse of CTS
39 subplot (2 ,2 ,2);
40 a =gca();
41 a.y_location="origin";
42 a.x_location="origin";
43 plot2d(W, XW_Phase *%pi/180,style=3);
44 xlabel(' Frequency in Radians / Seconds——> W');
45 ylabel('∠X(jW) ')
46 title(' Phase Response (CTFT)in Radians' )
47 mprintf(' |F(w)|= 1/sqrt(a^2+w^2) and\n Theta(w)=-
         atan(w/a) ')
48
49 //Chapter1: Signals
50 //Example 1.9b,page no12
51 //Given:
52 // Analog Signal
53
54
55 A=1;// Amplitude
56 Dt=0.005;
57 t1=-4.5:Dt:4.5;
58 xt1=exp(-A*abs(t1));
59 // Continuous time Fourier Transform
60 Wmax1 =2*%pi*1;// Analog Frequency = 1Hz
61 K=4;
62 k=0:(K/1000):K;
63 W1=k*Wmax1/K;
64 XW1=xt1*exp(-sqrt(-1)*t1 '*W1)*Dt;
65 XW1=real(XW1);
66 W1=[-mtlb_fliplr(W1), W1(2:1001) ]; // Omega from -
         Wmax to Wmax

```

```

67 XW1=[ mtlbfliplr(XW1), XW1(2:1001) ];
68 //subplot(1 ,1 ,1)
69 subplot(2 ,2 ,3);
70 b=gca();
71 b.y_location = "origin";
72 plot2d(t,xt,3);
73 xlabel('t in sec. ');
74 ylabel('x(t)')
75 title(' Continuous Time Signal')
76 subplot(2 ,2 ,4);
77 b =gca();
78 b.y_location = "origin";
79 plot2d(W1,XW1,5);
80 xlabel('Frequency in Radians / Seconds W');
81 ylabel('X(jW)')
82 title('Continuous time Fourier Transform ')
83
84 mprintf(' |F(w)|= 2*a/sqrt(a^2+w^2) and\n Theta(w)=0'
)

```

Scilab code Exa 1.10 Signals Ex 1 10

```

1  clc
2  //Chapter1: Signals
3  //Example1.10, page no 38
4  //Given
5  //a
6  A=1,delta=1e-3,T=10e-3
7  w0=2*%pi/T,n=0
8  for i=0:10
9      if n==0 then
10         Sa=1
11     else

```

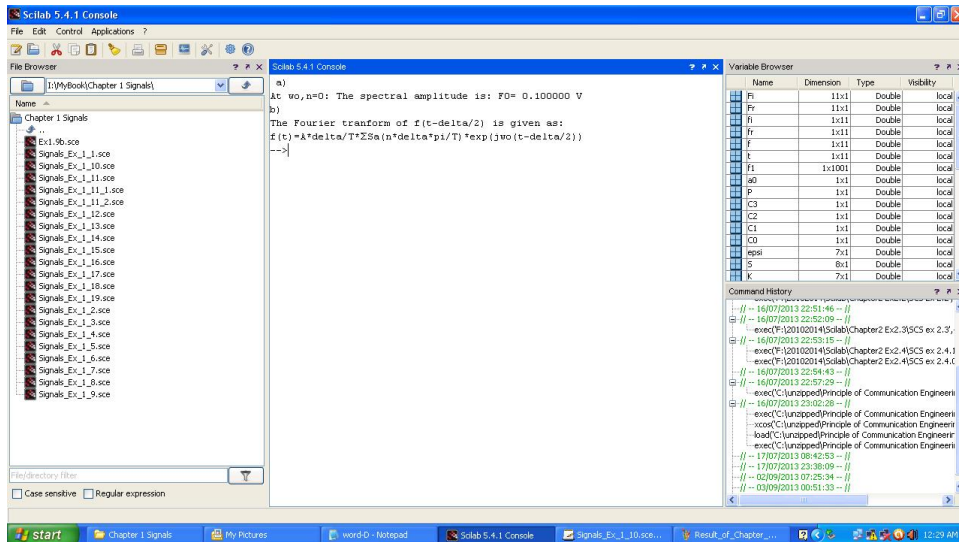


Figure 1.11: Signals Ex 1 10

```

12         Sa=sin(n*%pi*delta/T)/(n*%pi*delta/T)
13     end
14
15 end
16
17 F=(A*delta/T)*Sa//spectral Amplitude
18 mprintf('a)\nAt wo,n=0: The spectral amplitude is:
    F0= %f V\n',F)
19 //b
20 // displaying the fourier Transform of the given
    function
21 mprintf('b)\nThe Fourier tranform of f(t-delta/2) is
    given as: '),
22 mprintf('\nf(t)=A*delta/T* Sa(n*delta*pi/T)*exp(
    j*omega*(t-delta/2))')

```

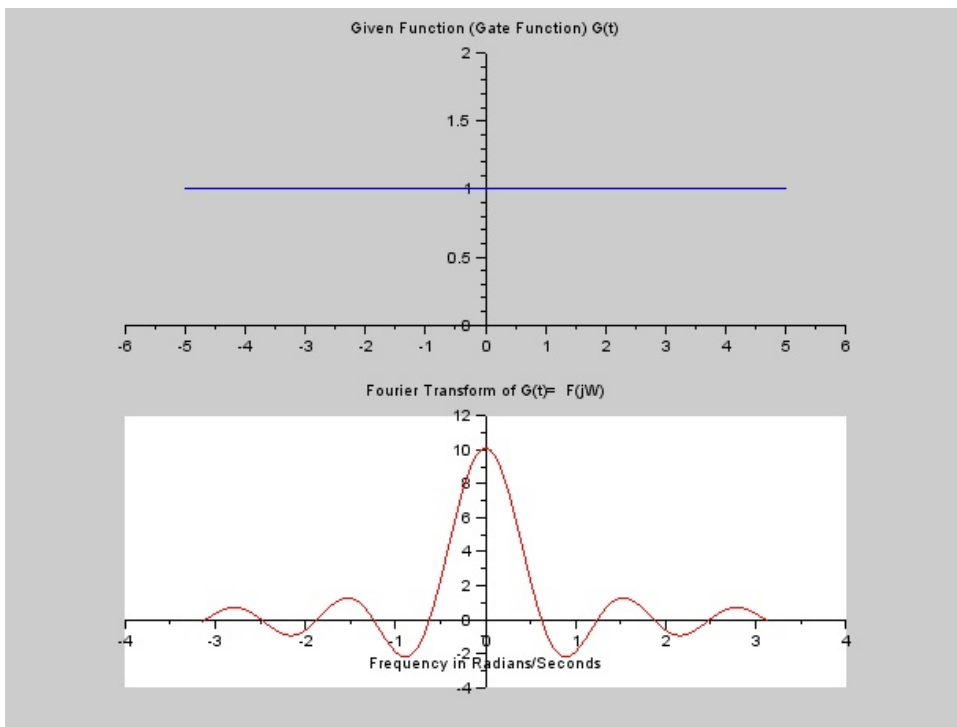


Figure 1.12: Signals Ex 1 11 Pg 39

Scilab code Exa 1.11.1 Signals Ex 1 11 Pg 39

```
1 clear ;
2 clc ;
3 close ;
4 //Chapter1
5 //Example1.11(1), page no 39
6 //Given
7 T = 10; //time Tau
8 Tg = -T/2 :0.1: T/2; // time period for given Gate
    Function  $-\tau/2$  to  $\tau/2$ 
9 G_t0 = 1; //Magnitude of Gate Function (A)
10 G_t = G_t0* ones (1, length (Tg)); // Gate function G
    (t)
11 f = -%pi: %pi / length (Tg): %pi ;
12 Dw = 0.1;
13 F_jW =G_t*exp(sqrt(-1)*Tg'*f)*Dw; // fourier
    Transform of the gate function
14 F_jW = real(F_jW);
15 // Plotting the Fourier Transform of G(t)
16 figure
17 subplot (2 ,1 ,1)
18 a=gca();
19 a.y_location = "origin";
20 a.x_location = "origin";
21 plot2d(Tg,G_t,2);
22 xtitle( ' Given Function (Gate Function) G(t) ' )
23 subplot(2 ,1 ,2)
24 a= gca();
25 a.y_location = "origin";
26 a.x_location = "origin";
27 plot2d(f,F_jW,5);
28 xlabel('Frequency in Radians/Seconds ');
29 title('Continuous time Fourier Transform X(jW)')
30 xtitle ( ' Fourier Transform of G(t)= F(jW) ' )
31 mprintf('F(w)= A*t*Sa(w*t/2)')
```

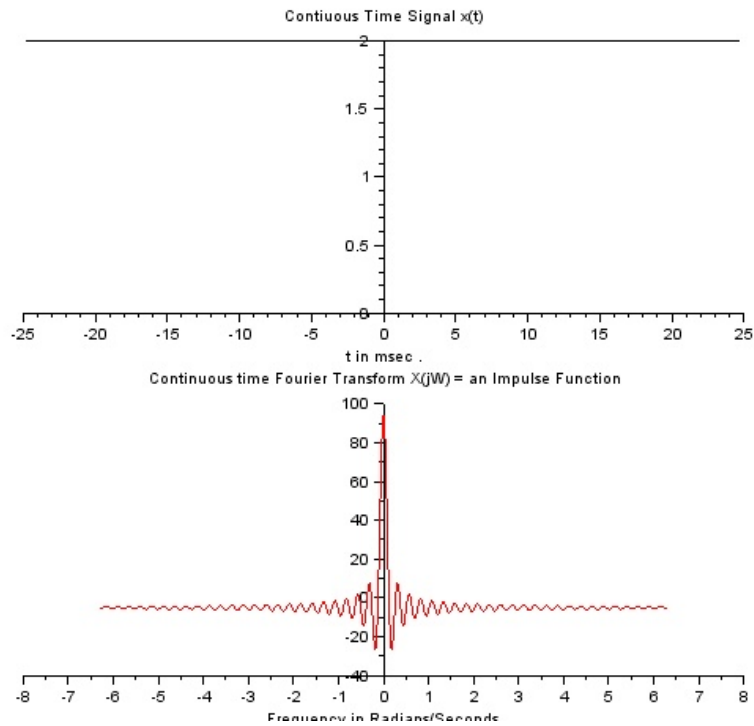


Figure 1.13: Signals Ex 1 11 Pg43

Scilab code Exa 1.11.2 Signals Ex 1 11 Pg43

```

1  clc
2  //Chapter1
3  //Example1.11(2) , page no 43
4  //Given
5  clear ;
6  close ;
7  // CTS Signal
8  A=2; // Amplitude

```

```

 9 Dt=0.01;
10 T1=49.5; //Time in seconds
11 t=-T1/2: Dt:T1 /2;
12 for i=1:length(t)
13 xt(i)= A;
14 end
15 // Continuous time Fourier Transform
16 Wmax=2*%pi*1;// Analog Frequency = 1Hz
17 K =4;
18 k=0:(K/1000):K;
19 W=k*Wmax/K;
20 xt=xt';
21 XW =(xt*exp(-sqrt(-1)*t'*W)*Dt)-5;
22 XW_Mag =real(XW);
23 W =[-mtlbfliplr(W), W(2:1001)]; // Omega from -Wmax
    to Wmax
24 XW_Mag =[mtlbfliplr(XW_Mag ), XW_Mag(2:1001)];
25 subplot(2 ,1 ,1);
26 a =gca();
27 a.data_bounds =[ -4 ,0;4 ,2];
28 a.y_location ="origin";
29 plot2d(t,xt);
30 xlabel('t in msec .');
31 title('Contiuous Time Signal x(t) ')
32 subplot(2 ,1 ,2);
33 a=gca();
34 a.y_location ="origin";
35 plot2d(W,XW_Mag,5);
36 xlabel('Frequency in Radians/Seconds ');
37 title('Continuous time Fourier Transform X(jW) = an
    Impulse Function ')
38 mprintf(' |F(w)|= 2*pi*A*delta(w), Hence the Fourier
    Transform of constant is an Impulse Function ')

```

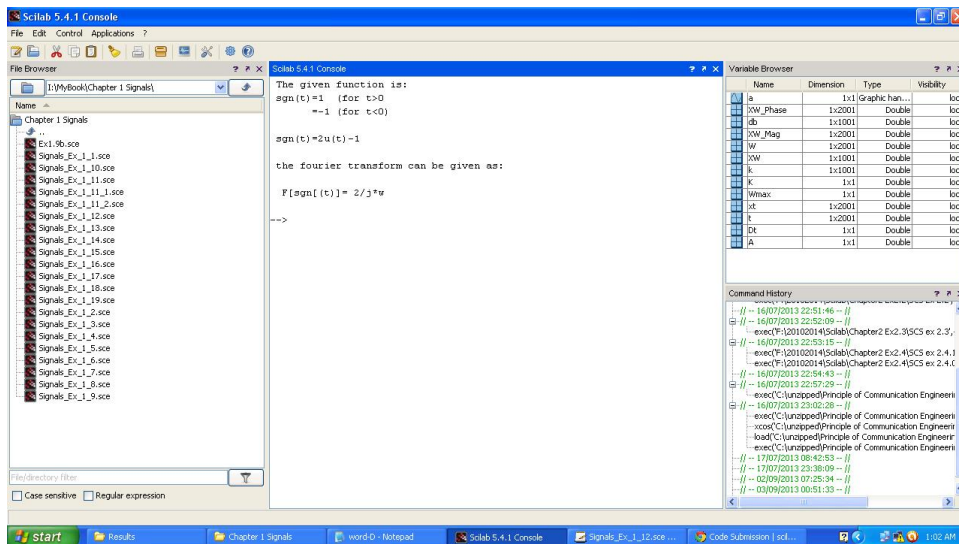


Figure 1.14: Signals Ex 1 12

Scilab code Exa 1.12 Signals Ex 1 12

```

1
2 clc
3 //Chapter 1
4 //Ex1.12, page no 43
5 //Given
6 mprintf('The given function is:\n sgn(t)=1 (for t
   >0\n
   =-1 (for t<0)\n')// displaying the
   given function
7 disp('sgn(t)=2u(t)-1')
8 disp('the fourier transform can be given as:'),disp(
   ' F[sgn[(t)]= 2/j*w')// displaying the fourier
   Transform of the given function

```

Scilab code Exa 1.13 Signals Ex 1 13

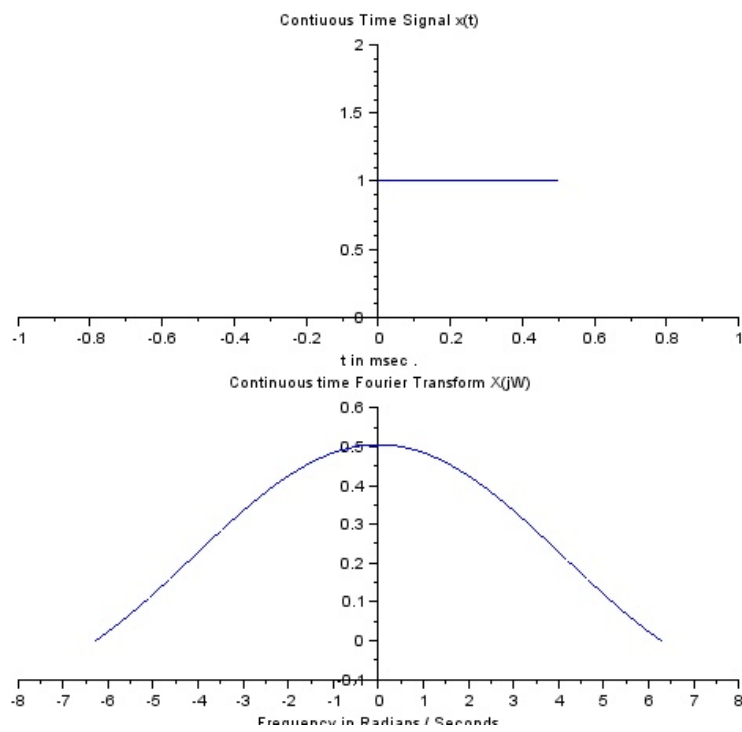


Figure 1.15: Signals Ex 1 13

```

1 clear ;
2 clc ;
3 close ;
4 //Chapter 1
5 //Ex1.13, page no 44
6 // CTS Signal
7 A =1; // Amplitude
8 Dt = 0.005;
9 T1 =0.5; //Time in seconds
10 t=0:Dt:T1;
11 for i = 1: length(t)
12 xt(i)=A;
13 end
14 // Continuous time Fourier Transform
15 Wmax= 2*%pi*1; // Analog Frequency = 1Hz
16 K =4;
17 k=0:(K/1000):K;
18 W =k*Wmax/K;
19 xt=xt';
20 XW =xt*exp(-sqrt(-1)*t'*W)*Dt;
21 XW_Mag =real(XW);
22 W =[-mtlbfliplr(W), W(2:1001)]; // Omega from -Wmax
    to Wmax
23 XW_Mag =[mtlbfliplr( XW_Mag ), XW_Mag(2:1001)];
24 // displaying the given function
25 subplot(2 ,1 ,1);
26 a =gca();
27 a.data_bounds =[-1,0;1,2];
28 a.y_location ="origin";
29 plot(t,xt);
30 xlabel('t in msec .');
31 title(' Contiuous Time Signal x(t) ')
32 // displaying the fourier Transform of the given
    function
33 subplot(2 ,1 ,2);
34 a=gca();
35 mprintf('F(w)= 1/(j*w)+ pi*delta(w)')
36 a.y_location ="origin";

```

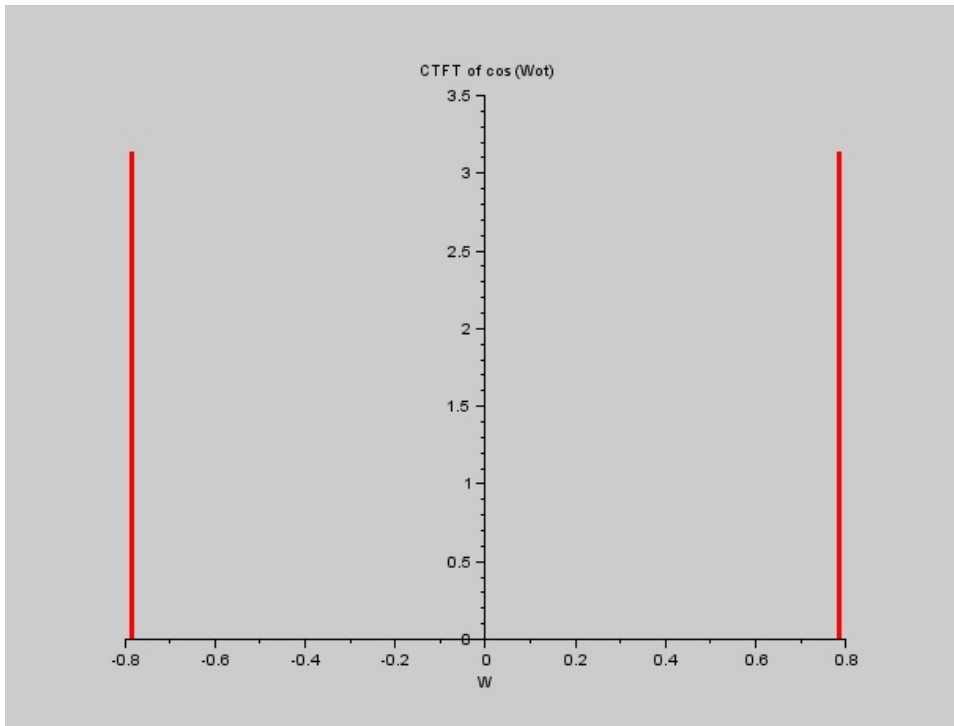


Figure 1.16: Signals Ex 1 14

```

37 plot(W, XW_Mag);
38 xlabel('Frequency in Radians / Seconds ');
39 title('Continuous time Fourier Transform X(jW)')

```

Scilab code Exa 1.14 Signals Ex 1 14

```

1 //Chapter 1
2 //Ex1.14, page no 44
3 // CTS Signal

```

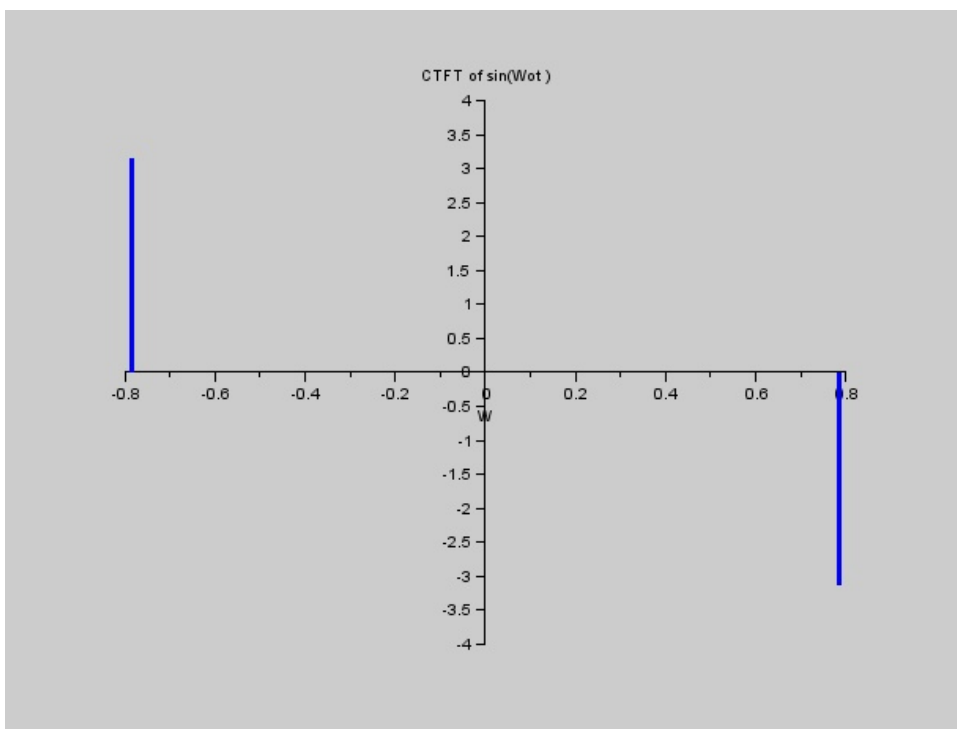


Figure 1.17: Signals Ex 1 14


```

4 // Continuous Time Fourier Transforms of
5 // Sinusoidal waveforms (a) sin(Wot) (b) cos(Wot)
6 clear ;
7 clc ;
8 close ;
9 // CTFT
10 T1 = 2;
11 T = 4* T1;
12 Wo = 2* %pi /T;
13 W = [-Wo ,0, Wo ];
14 ak = (2* %pi *Wo*T1/ %pi )/ sqrt ( -1);
15 XW = [-ak ,0, ak ];
16 ak1 = (2* %pi*Wo*T1/%pi);
17 XW1 =[ ak1 ,0, ak1 ];
18 //displaying the given function
19 figure
20 a =gca();
21 a.y_location=" origin";
22 a.x_location=" origin";
23 plot2d3('gnn',W,imag(XW),2);
24 poly1=a.children(1).children(1) ;
25 poly1.thickness =3;
26 xlabel('W' );
27 title( 'CTFT of sin(Wot) ')
28 //displaying the fourier Transform of the given
    function
29 figure
30 a =gca();
31 a.y_location=" origin";
32 a.x_location=" origin";
33 plot2d3('gnn',W,XW1,5);
34 poly1 =a.children(1).children(1) ;
35 poly1.thickness = 3;
36 xlabel('W' );
37 title( 'CTFT of cos (Wot) ')

```

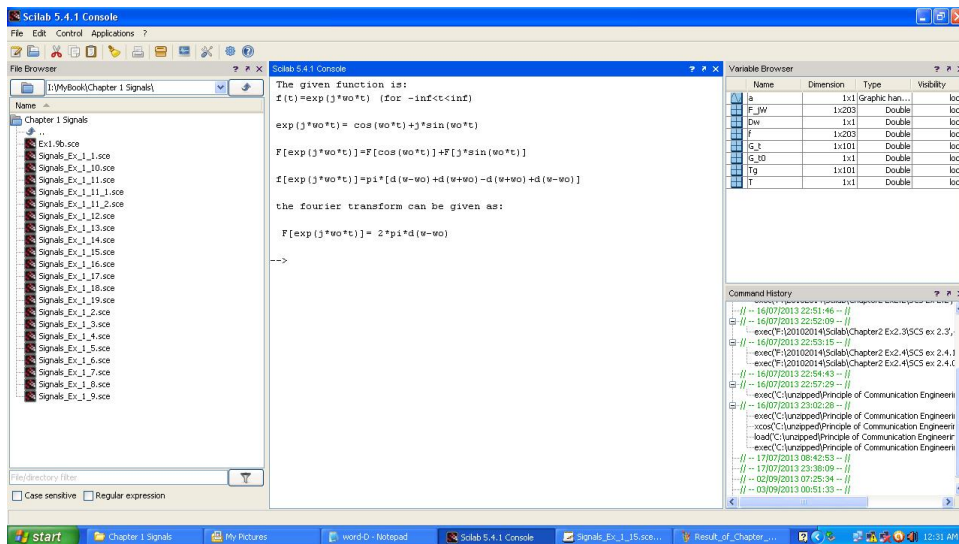


Figure 1.18: Signals Ex 1 15

Scilab code Exa 1.15 Signals Ex 1 15

```

1 clc
2 //Chapter 1
3 //Ex1.12, page no 43
4 //Given
5 mprintf('The given function is:\n f(t)=exp(j*wo*t) (
   for -inf<t<inf)\n')// Displaying the given
   function
6 disp('exp(j*wo*t)= cos(wo*t)+j*sin(wo*t)')
7 disp('F[exp(j*wo*t)]=F[cos(wo*t)]+F[j*sin(wo*t)]')
8 disp('f[exp(j*wo*t)]=pi*[d(w-wo)+d(w+wo)-d(w+wo)+d(w
   -wo)]')
9 disp('the fourier transform can be given as:'),disp(
   'F[exp(j*wo*t)]= 2*pi*d(w-wo)')// displaying the

```

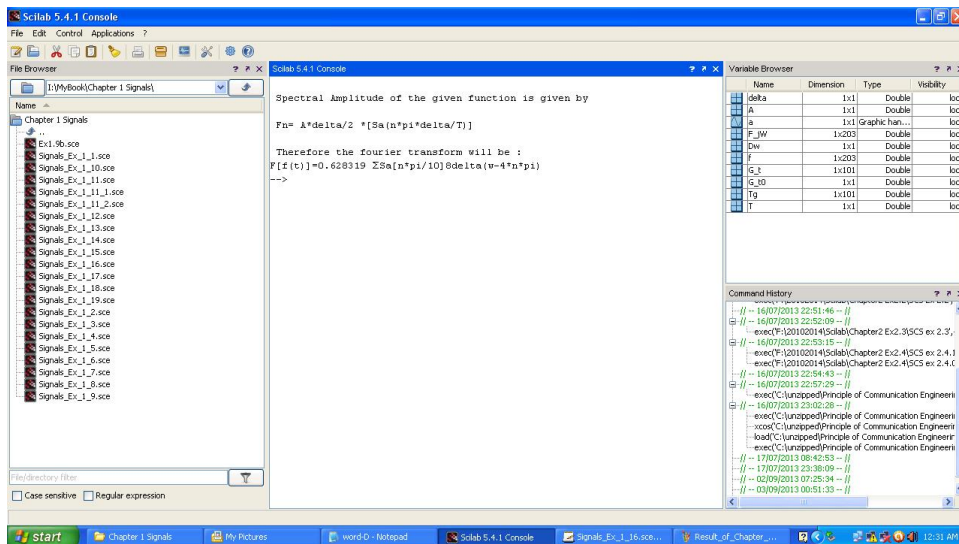


Figure 1.19: Signals Ex 1 16

fourier Transform of the given function

Scilab code Exa 1.16 Signals Ex 1 16

```
1 clc
2
3 //Chapter 1
4 //Ex1.16, page no 47
5 A=1,delta=50e-3,T=500e-3
6 disp('Spectral Amplitude of the given function is
   given by ')// Displaying the expression for
   Spectral Amplitude
7 disp('Fn= A*delta/2 *[Sa(n*pi*delta/T)] ')
8 disp('Therefore the fourier transform will be :')
9 mprintf('F[f(t)]=%f Sa [n*pi/10]8delta(w-4*n*pi)
   ,2*%pi*A*delta/T)// Displaying the Fourier
```

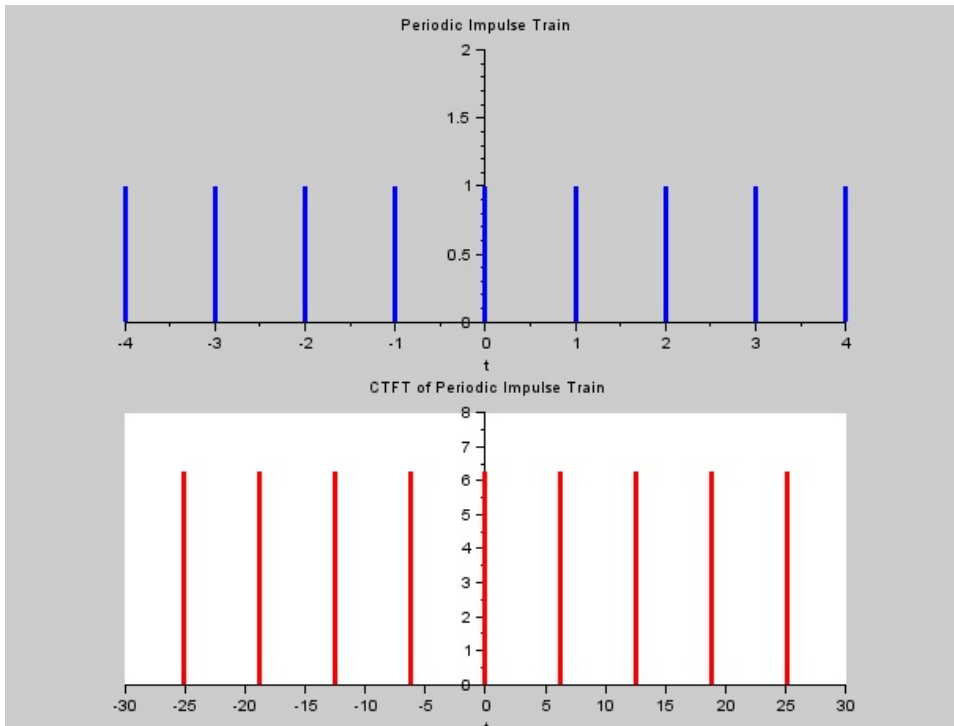


Figure 1.20: Signals Ex 1 17

transform

Scilab code Exa 1.17 Signals Ex 1 17

```

1  clc
2  //Chapter1: Signals
3  //Example1, page no12
4  //Given:
5  // CTFT
6  T = -4:4;;
7  T1 = 1; // Sampling Interval

```

```

8 xt = ones (1, length (T));
9 ak = 1/ T1;
10 XW = 2* %pi *ak* ones (1, length (T));
11 Wo = 2*%pi/T1;
12 W = Wo*T;
13 // Displaying the given function
14 figure
15 subplot(2 ,1 ,1)
16 a=gca();
17 a.y_location=" origin";
18 a.x_location=" origin";
19 plot2d3('gnn',T,xt,2) ;
20 poly1 =a.children(1).children (1) ;
21 poly1.thickness = 3;
22 xlabel ( 't ' );
23 title('Periodic Impulse Train ')
24 // displaying the fourier Transform of the given
    function
25 subplot(2 ,1 ,2)
26 a=gca();
27 a.y_location=" origin";
28 a.x_location=" origin";
29 plot2d3('gnn',W,XW,5) ;
30 poly1=a.children(1).children(1) ;
31 poly1.thickness =3 ;
32 xlabel('t');
33 title ( 'CTFT of Periodic Impulse Train ')
34
35 mprintf('F[ t (t)]= 2*pi/T* (w-wo) ')

```

Scilab code Exa 1.18 Signals Ex 1 18

```
1 clear ;
```

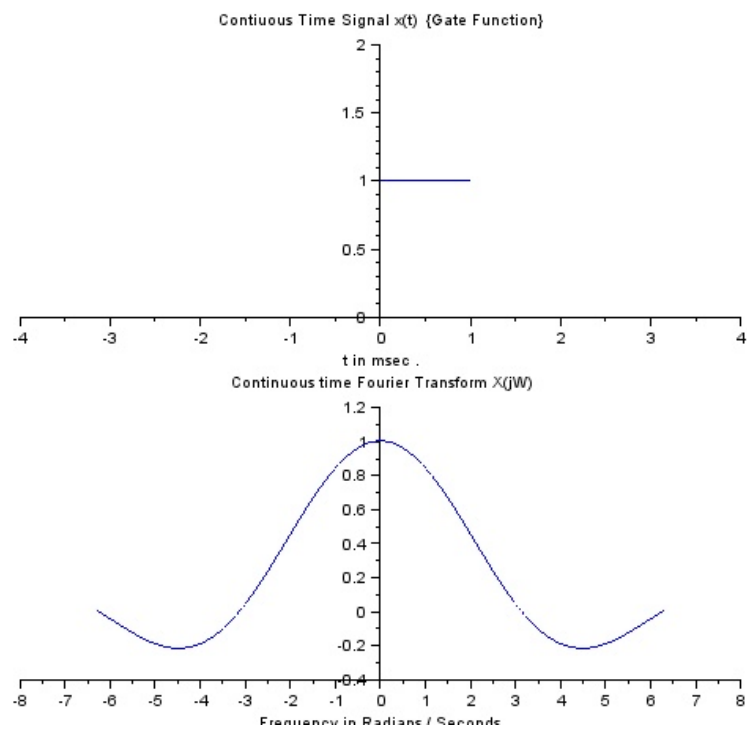


Figure 1.21: Signals Ex 1 18

```

2  clc ;
3  close ;
4  // CTS Signal
5  A =1; // Amplitude
6  Dt = 0.005;
7  T1 = 2; //Time in seconds
8  t = 0: Dt:T1 /2;
9  for i = 1: length (t)
10 xt(i) = A;
11 end
12 // Continuous time Fourier Transform
13 Wmax= 2*%pi*1; // Analog Frequency = 1Hz
14 K =4;
15 k=0:(K/1000):K;
16 W =k*Wmax/K;
17 xt=xt';
18 XW =xt*exp(-sqrt(-1)*t'*W)*Dt;
19 XW_Mag =real(XW);
20 W =[-mtlb_fliplr(W), W(2:1001)]; // Omega from Wmax
    to Wmax
21 XW_Mag =[mtlb_fliplr( XW_Mag ), XW_Mag(2:1001)];
22 // displaying the given function
23 subplot(2 ,1 ,1);
24 a =gca();
25 a.data_bounds =[ -4 ,0;4 ,2];
26 a.y_location ="origin";
27 plot(t,xt);
28 xlabel('t in msec .');
29 title(' Contiuous Time Signal x(t) {Gate Function}
    ')
30 // displaying the fourier Transform of the given
    function
31 subplot(2 ,1 ,2);
32 a=gca();
33 a.y_location ="origin";
34 plot(W, XW_Mag);
35 xlabel('Frequency in Radians / Seconds ');
36 title('Continuous time Fourier Transform X(jW)' )

```

```
37 mprintf('Hence Fourier transform of given Gate  
function is:\n A*delta*Sa[w*delta/2]/ exp(-j*w*  
delta/2)')
```

Chapter 2

Switched Communication Systems

Scilab code Exa 2.2 Chapter 2 Example 2 2

```
1  clc;
2  // Chapter 2 Switched communication systems
3  //Example 2.2, page no 125
4  //given
5  Io=4*10^-3 //rqueired operating current
6  N1=10000 //no of turns in the main winding
7  R1=645 //resistence of the main winding in ohms
8  N2=200 //no of turns in auxillary winding
9  B=2 //spacing bias
10 Iaux=B/N2 //maximum auxillary current
11 mprintf('maximum auxillary current is:%f mA\n',Iaux
    *1e3)
12 MMFaux=N2*Iaux //MMF in the auxillary winding
13 mprintf('MMF in the auxillary winding is:%fAT \n',
    MMFaux)
14 MMFop=Io*N1 //operating MFF in main winding
15 mprintf( ' MMF in main winding is:%f AT \n',MMFop)
```

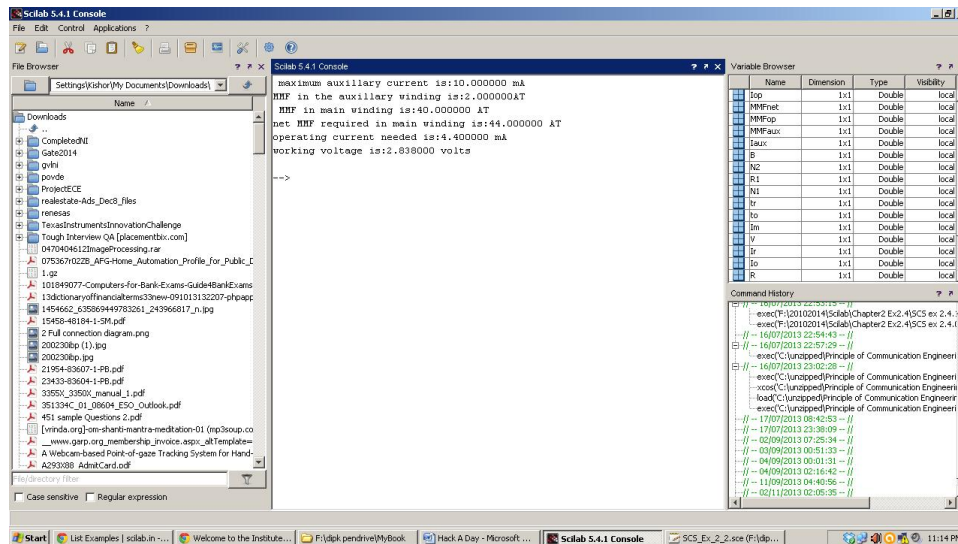


Figure 2.1: Chapter 2 Example 2 2

```

16 MMFnet=MMFop+(0.1*MMFop) //net MMF required in main
    winding
17 mprintf('net MMF required in main winding is:%f AT \
    n',MMFnet)
18 Iop=MMFnet/N1 //operating current needed
19 mprintf('operating current needed is:%f mA \n',Iop*1
    e3)
20 V=Iop*R1 //working voltage in volts
21 mprintf('working voltage is:%f volts \n',V)

```

Scilab code Exa 2.3 Chapter 2 Example 2 3

```

1 clc;
2 // Chapter 2 Switched communication systems
3 //Example 2.3, page no 125
4 //given

```

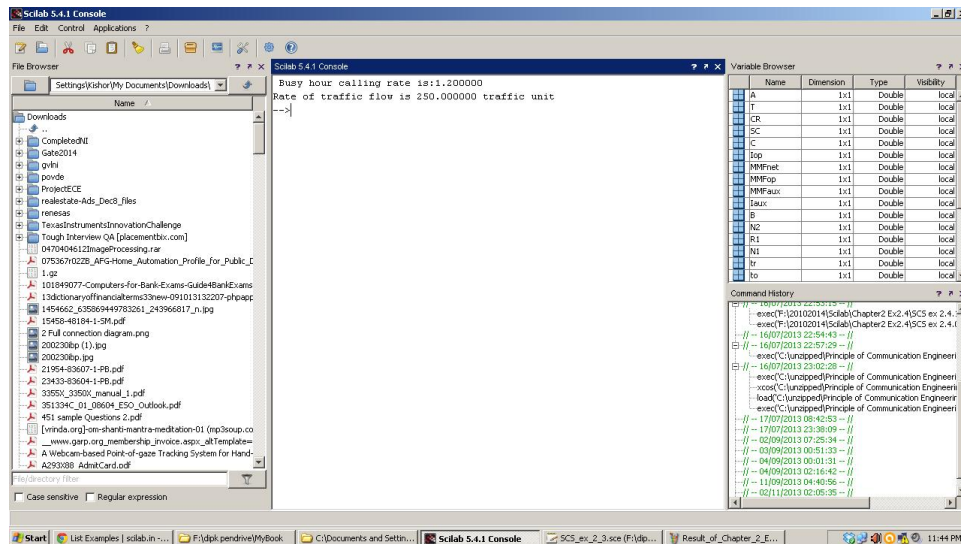


Figure 2.2: Chapter 2 Example 2 3

```

5 C=6000//Tatol no of call in busy hour
6 SC=5000//no of subscribers
7 CR=C/SC//busy hour calling rate
8 mprintf('Busy hour calling rate is:%f \n',CR)
9 T=2.5/60//avaraage duration of calls in hours
10
11 A=C*T//rate of traffic flow
12 mprintf('Rate of traffic flow is %f traffic unit ',A
    )

```

Scilab code Exa 2.4.0 Chapter 2 Example 2 4 Pg126 Top

```

1 clc;
2 // Chapter 2 Switched communication systems
3 //Example 2.4 ,page no 126
4 //given

```

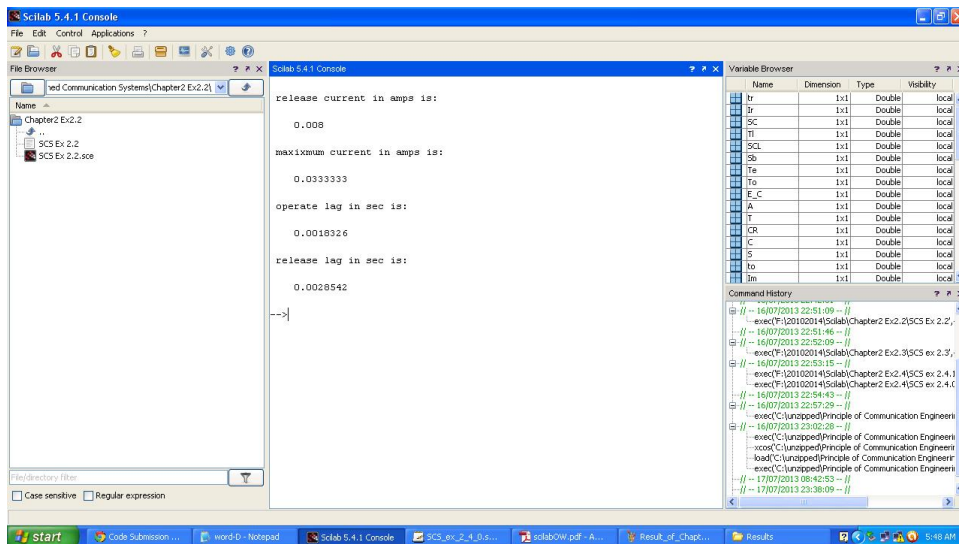


Figure 2.3: Chapter 2 Example 2 4 Pg126 Top

```

5 L=3//relay inductance in henry
6 R=1500//relay resistance in ohm
7 Io=20e-3//operating current in amps
8 Ir=8e-3//release current in amps
9
10 V=50//supply volatage in volts
11 Im=V/R//maxixmum current in amps
12 mprintf('maxixmum current is %f mamps \n',Im*1e3)
13 to=(L/R)*log(1/(1-(Io/Im)))//operate lag in sec
14 mprintf('operate lag is %f msec \n',to*1000)
15 tr=(L/R)*log(Im/Ir)//release lag in sec
16 mprintf('release lag is %f msec \n',tr*1000)

```

Scilab code Exa 2.4.1 Chapter 2 Example 2 4 Pg126 Bottom

```
1 clc;
```

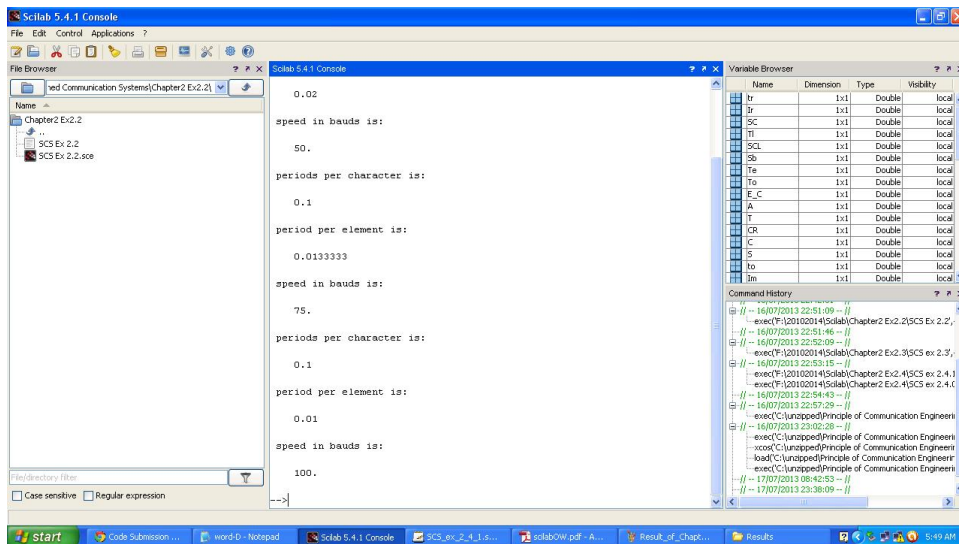


Figure 2.4: Chapter 2 Example 2 4 Pg126 Bottom

```

2 // Chapter 2 Switched communication systems
3 //Example 2.4.1 ,page no 126
4 //given
5 //a
6 C_S1=20/3//speed in characters per second
7 P_C1=1/C_S1//periods per character
8 mprintf('(a)\nperiods per character is:%f msec\n',
          P_C1*1e3)
9 E_C1=7.5//elements per character
10 P_E1=P_C1/E_C1//period per element
11 mprintf('period per element is:%f msec\n',P_E1*1e3)
12 Sb1=1/P_E1//speed in bauds
13 mprintf('speed is:%f bauds\n\n',Sb1)
14 //b
15 C_S2=10//speed in characters per second
16 P_C2=1/C_S2//periods per character
17 mprintf('(b)\nperiods per character is:%f msec\n',
          P_C2*1e3)
18 E_C2=7.5//elements per character
19 P_E2=P_C2/E_C2//period per element

```

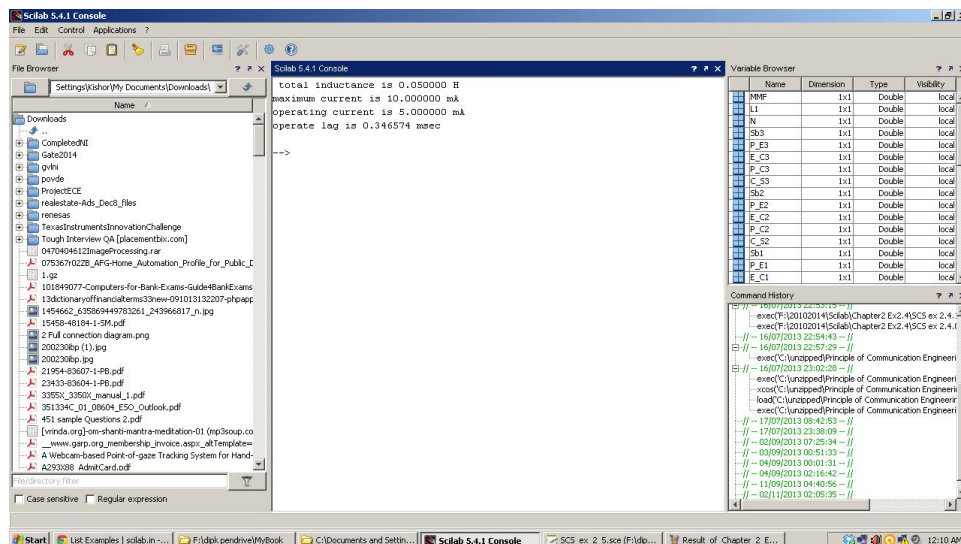


Figure 2.5: Chapter 2 Example 2 5

```

20 mprintf('period per element is:%f msec\n',P_E2*1e3)
21 Sb2=1/P_E2//speed in bauds
22 mprintf('speed is %f bauds\n\n', Sb2)
23 //c
24 C_S3=10//speed in characters per second
25 P_C3=1/C_S3//periods per character
26 mprintf('(c)\nperiods per character is:%f msec\n',
    P_C3*1e3)
27 E_C3=10//elements per character
28 P_E3=P_C3/E_C3//period per element
29 mprintf('period per element is:%f msec\n',P_E3*1e3)
30 Sb3=1/P_E3//speed in bauds
31 mprintf('speed is %f bauds\n',Sb3)

```

Scilab code Exa 2.5 Chapter 2 Example 2 5

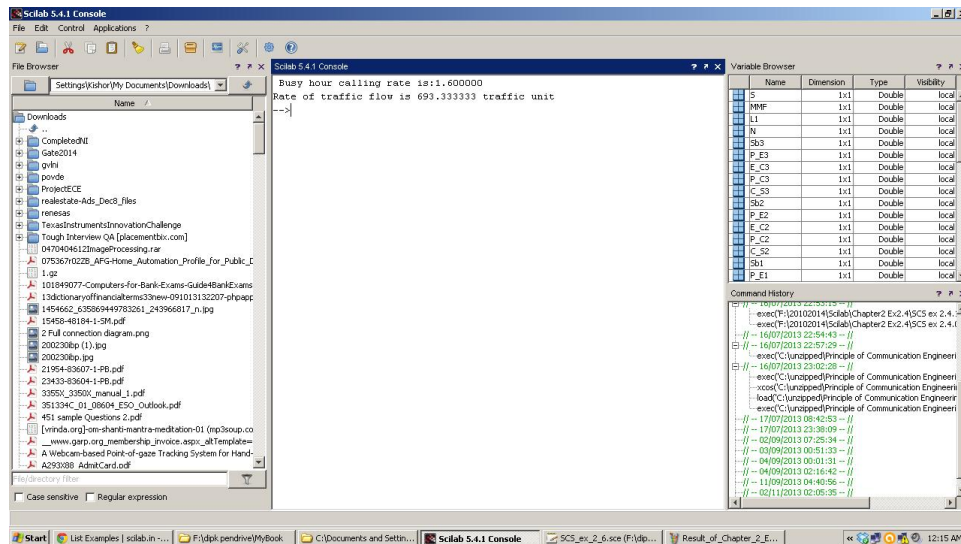


Figure 2.6: Chapter 2 Example 2 6

```

1  clc ;
2  // Chapter 2 Switched communication systems
3  //Example 2.5 ,page no 127
4  //given
5  N=1000//no of turns
6  L1=5e-8//inductance per turn
7  L=N^2*L1//total inductance
8  mprintf('total inductance is %f H \n',L)
9  R=100//resistance of winding in ohm
10 MMF=5//operating MMF in amp. turn
11 V=1//voltage of received signal in volts
12 Im=V/R//maximum current
13 mprintf('maximum current is %f mA \n',Im*1e3)
14 Io=MMF/N//operating current
15 mprintf('operating current is %f mA \n',Io*1e3)
16 to=(L/R)*log(1/(1-(Io/Im)))//operate lag
17 mprintf('operate lag is %f msec \n',to*1e3)

```

Scilab code Exa 2.6 Chapter 2 Example 2 6

```
1  clc;
2  // Chapter 2 Switched communication systems
3  //Example 2.6 ,page no 128
4  //given
5  S=10000//no of subscribers
6  C=16000//Tatol no of call in busy hour
7  CR=C/S//busy hour calling rate
8  mprintf('Busy hour calling rate is:%f \n',CR)
9  T=2.6//avarage duration of calls in min
10
11 A=C*(T/60)//rate of traffic flow
12 mprintf('Rate of traffic flow is %f traffic unit ',A
    )
```

Scilab code Exa 2.7 Chapter 2 Example 2 7

```
1  clc;
2  // Chapter 2 Switched communication systems
3  //Example 2.7 ,page no 135
4  //given
5  N=7//no of character elements
6  E_C=10//elements per character (1+7+1+1)
7  To=100e-3//duration of one character
8  Te=To/E_C//duration of each element
9  mprintf('duration of each element is:%f msec\n',Te*1
    e3)
10 Sb=1/Te//speed in bauds
11 mprintf('speed is %f bauds\n',Sb)
```

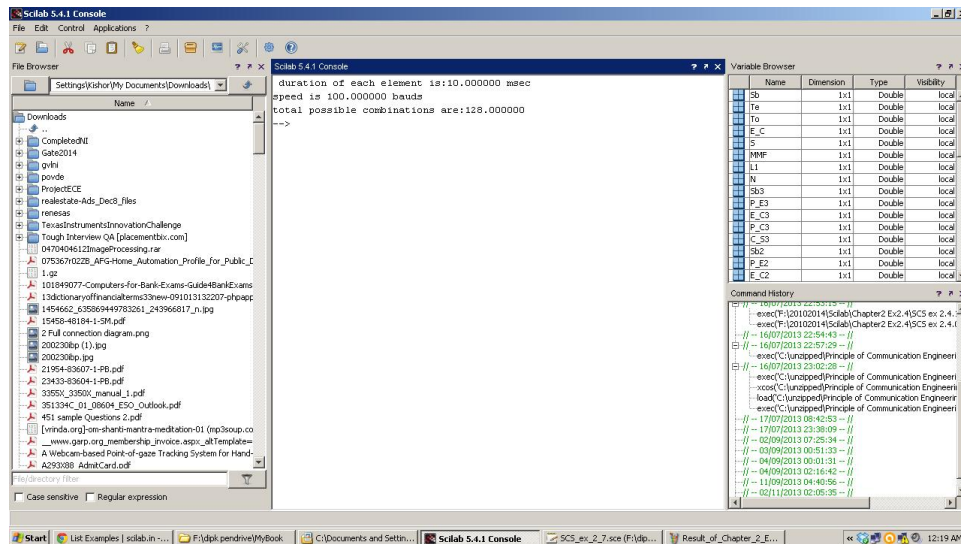



Figure 2.7: Chapter 2 Example 2 7

```

12 C=2^N//total possible combinations
13 mprintf('total possible combinations are:%f',C)

```

Scilab code Exa 2.8 Chapter 2 Example 2 8

```

1 clc;
2 // Chapter 2 Switched communication systems
3 //Example 2.8,page no 129
4 //given
5 S=1000//no of subscribers
6 T=2.4/60//avarage duration of calls in hours
7 A=60//rate of traffic flow
8 C=A/T//Tatol no of call in busy hour
9 mprintf('Total no of call in busy hour is:%f calls
    per Hour\n',C)
10 CR=C/S//busy hour calling rate

```

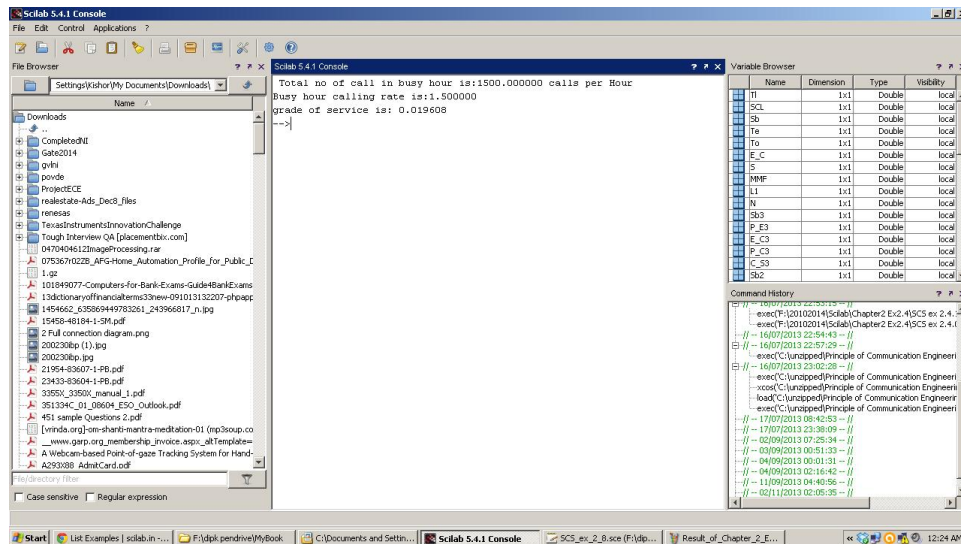


Figure 2.8: Chapter 2 Example 2 8

```

11 mprintf('Busy hour calling rate is:%f \n',CR)
12 SCL=30//no of call lost per hour
13
14 B=SCL/(C+SCL)//grade of service
15 mprintf('grade of service is: %f',B)

```

Scilab code Exa 2.9 Chapter 2 Example 2 9

```

1 clc;
2 // Chapter 2 Switched communication systems
3 //Example 2.9 ,page no 129
4 //given
5 N=5//no of switches
6 A=0.9//traffic offered
7 //grade of service  $B = (A^N/N!) / (1 + A + A^2/2! + A^3/3! + \dots + A^N/N!)$ 

```

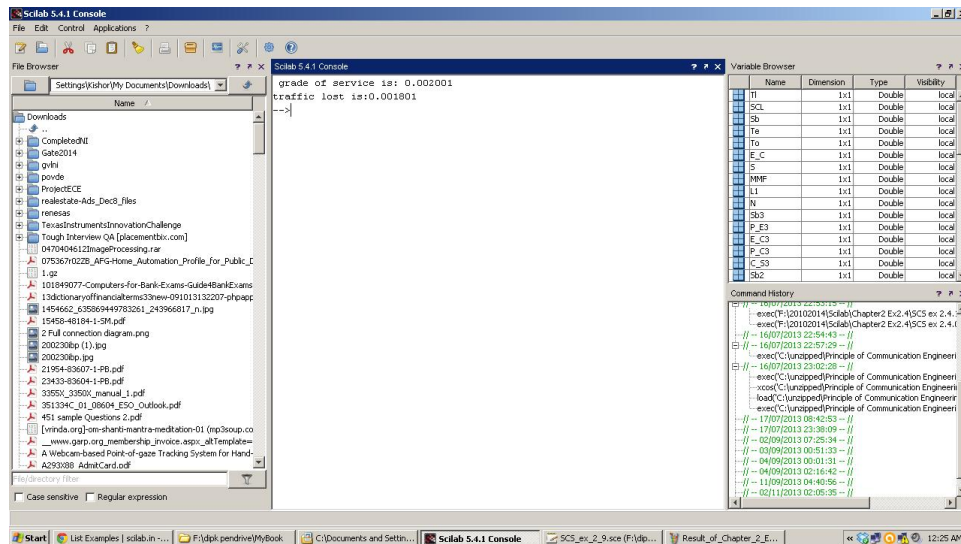


Figure 2.9: Chapter 2 Example 2 9

```

8 // here
9 B=(A^N/factorial(N))/(1+A+(A^2/factorial(2))+(A^3/
    factorial(3))+(A^4/factorial(4))+(A^5/factorial
    (5)))
10 mprintf('grade of service is: %f\n',B)
11 Tl=A*B// traffic lost
12 mprintf('traffic lost is:%f',Tl)

```

Chapter 3

Modulation

Scilab code Exa 3.1 Modulation Ex 3 1

```
1  clc
2  //Chapter3: Modulation
3  //Example3.1, page no 135
4  //Given
5  Ic=10 //carrier current in Amps
6  Imod=11.6// Current after modulation
7  Rl=1//Assumed load in ohm
8  Pmod=Rl*Imod^2//power before modulation
9  Ma= sqrt(2*((Pmod/Ic^2)-1))//percentage modulation
10 Pc=10
11 Pmod=Pc*(1+(Ma^2/2))//power after modulation
12 mprintf('percentage modulation is:%f%c\n Power after
    modulation is:%f watts ',Ma*100, '%',Pmod)
```

Scilab code Exa 3.2 Modulation Ex 3 2

```
1  clc
```

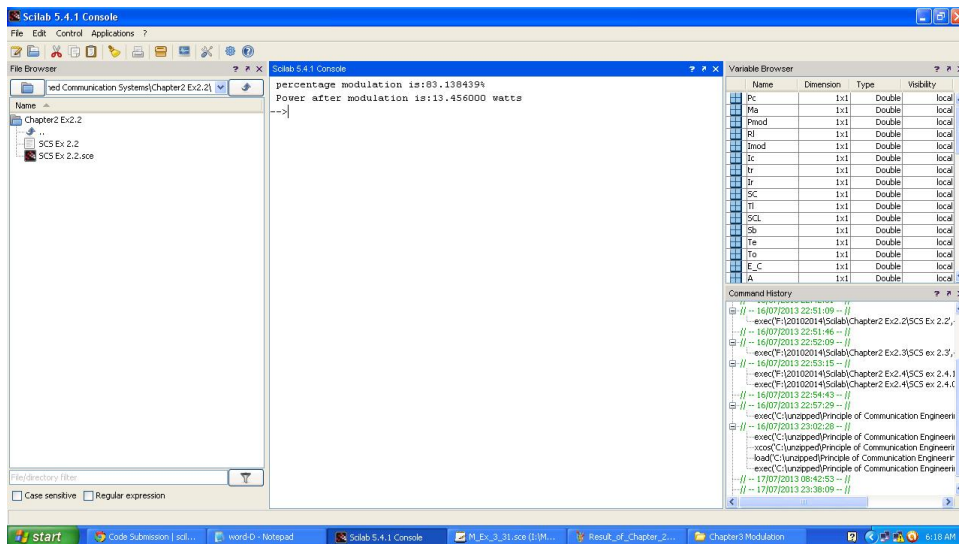


Figure 3.1: Modulation Ex 3 1

```

2 //Chapter3: Modulation
3 //Example3.2, page no 135
4 //Given
5 Pc=9e3// Tx Power without modulation
6 Pmod=10.125e3//Tx Power after modulation
7 Ma= sqrt(2*((Pmod/Pc)-1))//depth of (percentage)
  modulation
8 mprintf('Depth of modulation is:%f',Ma)

```

Scilab code Exa 3.3 Modulation Ex 3 3

```

1 clc
2 //Chapter3: Modulation
3 //Example3.3
4 //Given
5 M1=0.2//depth of modulation for first tone

```

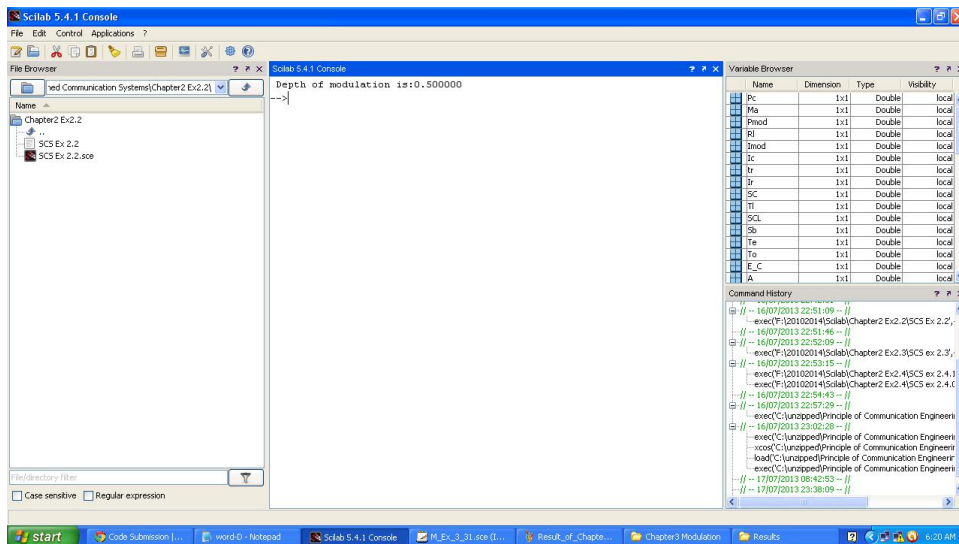


Figure 3.2: Modulation Ex 3 2

```

6 M2=0.4//depth of modulation for second tone
7 Pc=1200//Tx Power
8 Pmod=Pc*(1+M1^2/2+M2^2/2)//total power radiated
   after modulation by both the tones
9 mprintf('The total power radiated is %d watts',Pmod)

```

Scilab code Exa 3.4 Modulation Ex 3 4

```

1 clc
2 //Chapter3: Modulation
3 //Example3.4, page no 138
4 //Given
5 Ebb=2e3//DC plate supply
6 Ecc=-500//DC grid bias
7 Ib=67e-3//DC plate current
8 Ic=30e-3//DC grid current

```

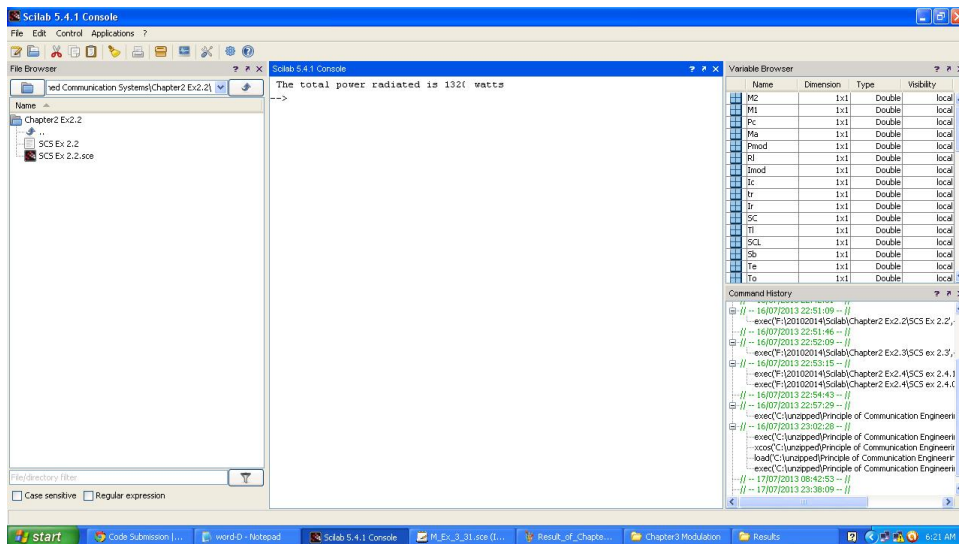


Figure 3.3: Modulation Ex 3 3

```

9 Egm=750//RF peak grid voltage
10 Pout=75//RF Power output
11 Ma=0.75//Depth of modulation
12 Paf=(Ma^2*Ebb*Ib)/(2*1)//modulating power required
    from the audio source
13 Pdc=Ebb*Ib//Power supplied by DC source
14 Zm=Ebb^2/Pdc//Modulator Impedance
15
16 Pd=Pdc+Paf-Pout//Plate dissipation
17 mprintf('modulating power required from the audio
    source\n is:%f watts\n Modulator Impedance is:%f
    ohm\n Plate dissipation is:%f watts ',Paf,Zm,Pd)

```

Scilab code Exa 3.5 Modulation Ex 3 5

```
1 clc
```

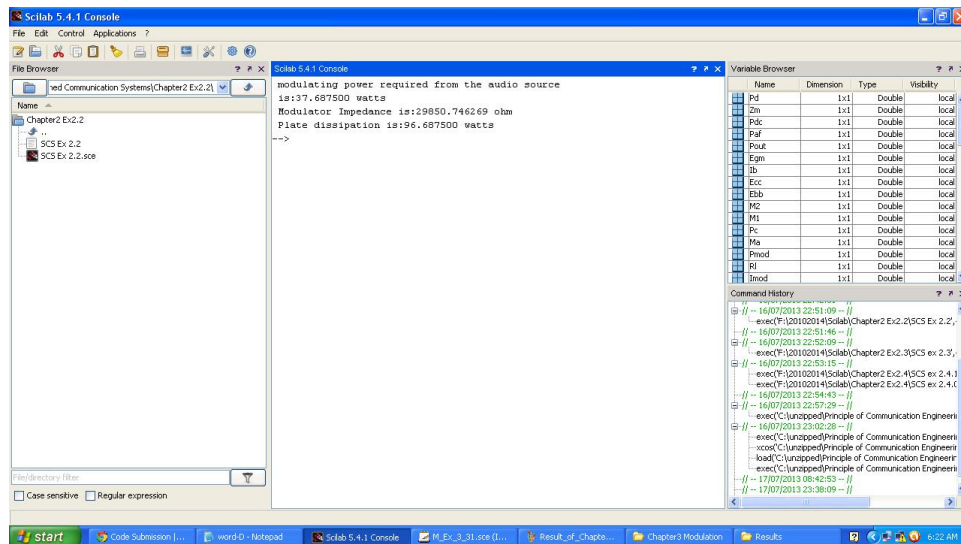


Figure 3.4: Modulation Ex 3 4

```

2 //Chapter3: Modulation
3 //Example3.5b, page no 139
4 //Given
5 Pd=944//Anode dissipation of the class C amplifier
   in watts
6 Ma=0.6//modulation depth,
7 Etta=0.6//efficiency
8 Pout=(Etta*Pd/(1-Etta))//power dissipation at 60%
   modulation
9 Pc=Pout/(1+(Ma^2/2))//Tx power
10 Psb=Pout-Pc
11 Pdc1=Pc/Etta//DC power inputto PA
12 Paf=Psb/Etta// modulation power input to PA
13 Eff=0.25// efficiency of the modulator
14 Pdc2=Paf/Eff//DC power input to modulator
15 Pdct=Pdc1+Pdc2//Total DC power to the system
16 Effo=Pout/Pdct//Overall Efficiency
17 Ma=1 // 100% modulation
18 Pt=Pc*(1+(Ma^2)/2)
19 Psb=(Pc*Ma^2)/2

```

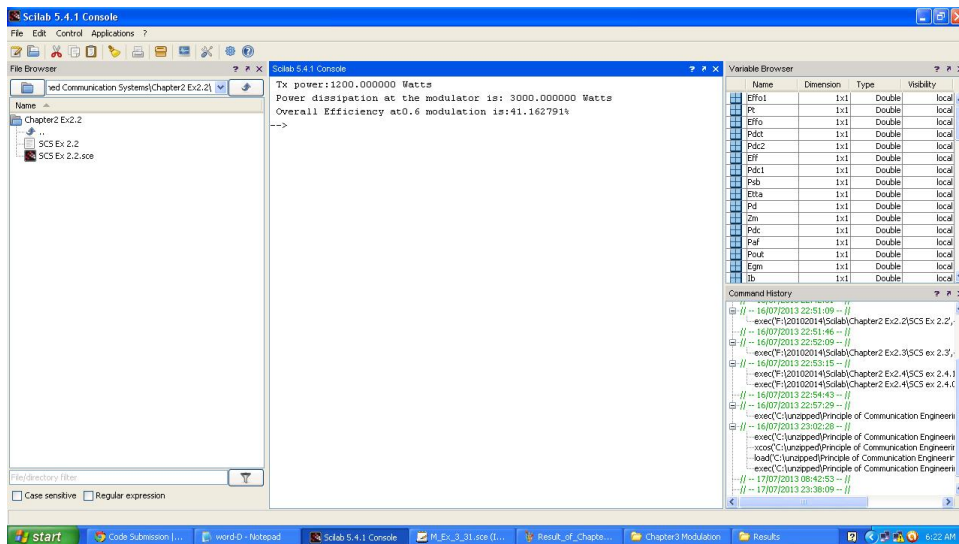



Figure 3.5: Modulation Ex 3 5

```

20 Paf=Psb/Etta//modulating input power to PA
21 Pdc2=Paf/Eff// DC power input to modulator
22 Pd=Pdc2-Paf//Power dissipation at the modulator
23 Effo1=Pout/(Pdc1+Pdc2)//Overall Efficiency
24 mprintf('Tx power:%f Watts\n Power dissipation at
the modulator is: %f Watts\n Overall Efficiency
at0.6 modulation is:%f%c ',Pc,Pd,100*Effo,'%')

```

Scilab code Exa 3.6 Modulation Ex 3 6

```

1 clc
2 //Chapter3: Modulation
3 //Example3.6, page no 141
4 //Given
5 Pdc=1400//DC power i/p to PA under 100% modulation
6 Ptdc=400//Plate dissipation

```

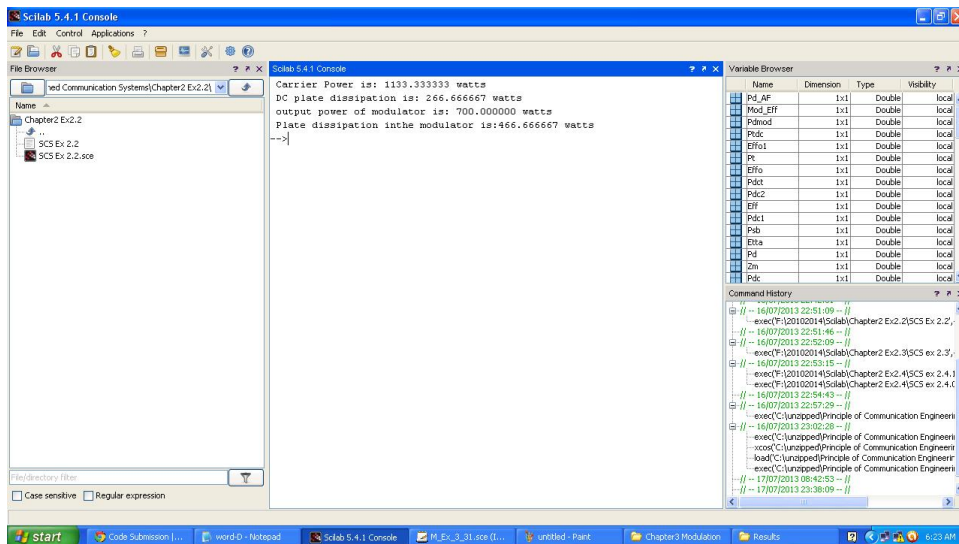


Figure 3.6: Modulation Ex 3 6

```

7 Pd=Ptdc*(2/3)//DC plate dissipation
8
9 Pdmmod=Ptdc*(1/3)//
10 Pc=Pdc-Pd// Carrier Power
11
12 Psb=Pc/2//side band power
13 Paf=Psb+Pdmmod//output power of modulator
14
15 Mod_Eff=0.6
16 Pdc2=Paf/Mod_Eff//DC i/p power to the modulator
17 Pd_AF=Pdc2-Paf//Plate dissipation inthe modulator
18 mprintf('Carrier Power is: %f watts \n DC plate
    dissipation is: %f watts\n output power of
    modulator is: %f watts\n Plate dissipation inthe
    modulator is: %f watts',Pc,Pd,Paf,Pd_AF)

```

Scilab code Exa 3.7 Modulation Ex 3 7

```
1 clc
2 //Chapter3: Modulation
3 //Example3.7, page no 141
4 //Given
5 Paf=500//Modulator output power
6 Eff=0.75//Efficiency of the amplifier
7 P_lost=Paf*(1-Eff)//modulating power lost in the
   amplifier
8 Psb=Paf*Eff//side band power
9
10 m=1
11 Pc=2*Psb
12
13 Pt=Pc+Psb//Total RF power
14 mprintf('Maximum carrier power is: %d watts\n Total
   RF power is: %d watts',Pc,Pt)
```

Scilab code Exa 3.8 Modulation Ex 3 8

```
1 clc
2 //Chapter3: Modulation
3 //Example3.8, page no 143
4 //Given
5 Po=3000// Rating of Power Amplifier
6 Pr=750//Push-Pull amplifier rated as
7 Paf=2*Pr//Rated power output from Push-Pull
   modulator
8 Eff=0.6
9 P_lost=Paf-(Eff*Paf)//Modulation power lost
10 Psb=Paf-P_lost//side band power
11
```

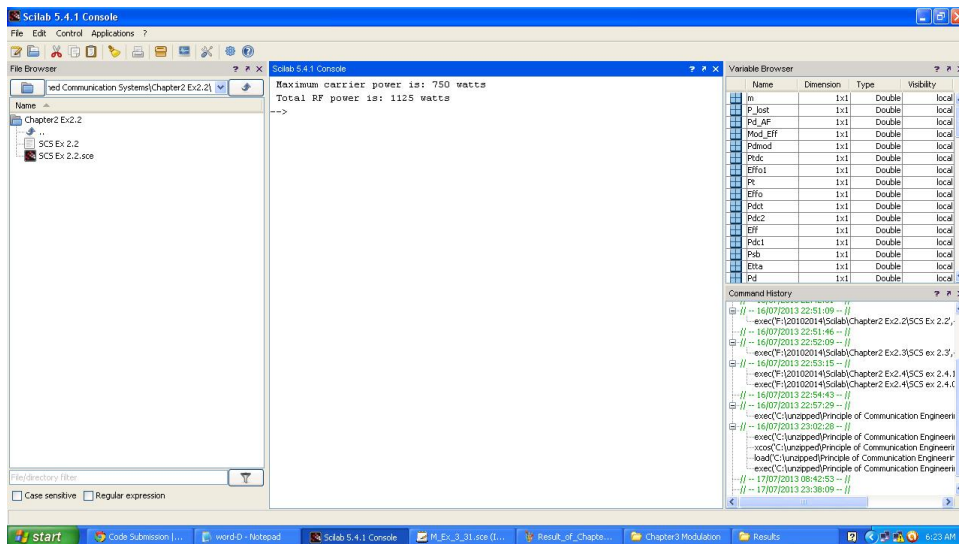


Figure 3.7: Modulation Ex 3 7

```

12 Pc=Po-Psb // Carrier power
13 Ma=sqrt(2*Psb/Pc)*100 //Maximum depth of modulation
14 mprintf('Carrier power is: %d watts\n Maximum depth
    of modulation is: %f',Pc, Ma)

```

Scilab code Exa 3.9 Modulation Ex 3 9

```

1 clc
2 //Chapter3: Modulation , page no 142
3 //Example3.9
4 //Given
5 t=0:0.001:10
6 //e=500*(1+(0.4*sin(3140*t)))*sin(6.28e7*t)
7 //a
8 wc=6.28e7 // Carrier angular frequency
9 fc=wc/(2*pi) // Carrier freq

```

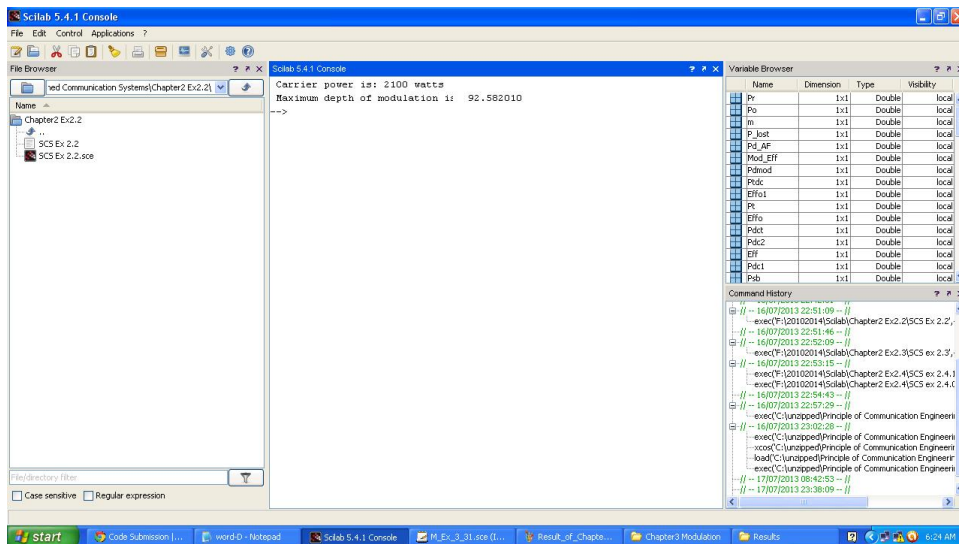


Figure 3.8: Modulation Ex 3 8

```

10 //b
11 wm=3140//Modulating angular freq
12 fm=wm/(2*%pi)//Modulating freq
13 //c
14 Ec=500///peak carrier voltage
15 Pc=(Ec^2)/(2*600)//Carrier power
16 //d
17 Ma=0.4
18 Pt=Pc*(1+(Ma^2 / 2))//Mean output power
19 //e
20 Rl=600//load resistance
21 Ecp=Ec+(Ma*Ec)//Peak output voltage
22 Ptm=Ecp^2/(2*Rl)//Peak power
23 mprintf('Carrier freq is: %d MHz\nModulating freq is
: %d Hz\nCarrier power is: %f watts\nMean output
power is: %f watts\nPeak output power is: %f
watts ',round(fc*1e-6),round(fm),Pc,Pt,Ptm)

```

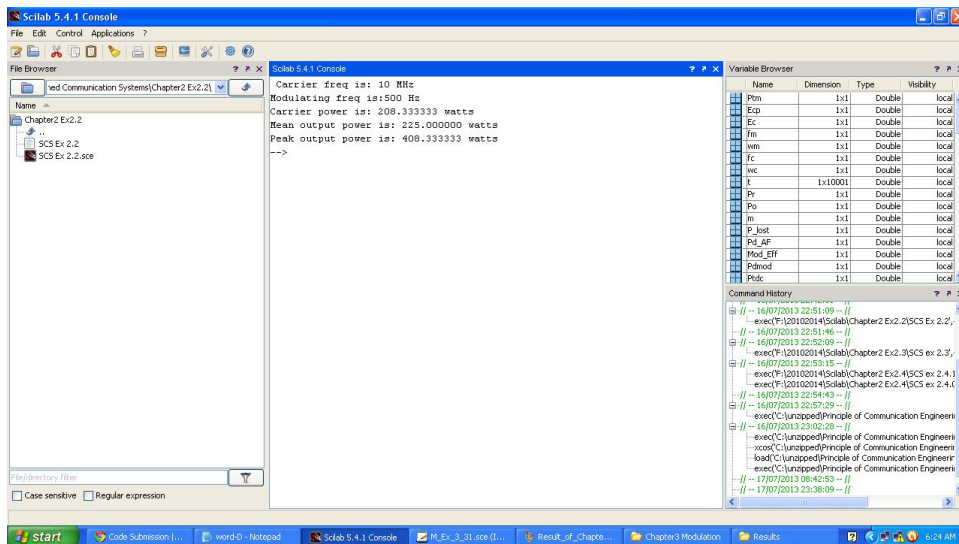


Figure 3.9: Modulation Ex 3 9

Scilab code Exa 3.10 Modulation Ex 3 10

```

1
2 clc
3 //Answers from the book are little deviated but the
   evaluated values in the scilab are correct
   results
4 //Chapter3: Modulation
5 //Example3.10, page no 143
6 //Given
7 //b
8 Pc=50e3//Carrier power
9 Z=36 + %i*40//base impedance of the antenna
10 Ma=1//modulation depth
11 Pmod=Pc*(1+((Ma^2)/2))//power delivered to the
   antenna under 100% modulation
12 //i

```

```

13 R=36//resistance of the antenna
14 Irms=sqrt(Pmod/R)//Antenna Current
15
16 // ii
17 Ic=sqrt(Pc/R)//RMS carrier current
18
19 Icm=Ic*sqrt(2)// Peak carrier current
20 Imod=2*Icm//Modulated current
21
22 Theta=atan(40/36)*180/%pi// from real and imaginary
    components of Z
23 Vbm100=Imod*Z//Peak base output voltage for 100%
    modulation
24 [Re_Vb , Im_Vb]=polar(Vbm100)
25
26 // iii
27 Ma=0.5
28 Imod=Icm*(1+0.5)
29
30 Vbm50=Imod*Z
31 [Re_Vb1 , Im_Vb1]=polar(Vbm50)
32 mprintf('Antenna current for full modulation is: %f
    amp\nPeak base voltage is: %f/_%d volts\nPeak
    base voltage is: %f/_%d volts',Irms,Re_Vb,Theta,
    Re_Vb1,Theta)
33 // The Ans is little deviated from that of book as
    the decimal places considered while calculating
    at different stages might be different

```

Scilab code Exa 3.11 Modulation Ex 3 11

```

1 clf();
2 clc

```

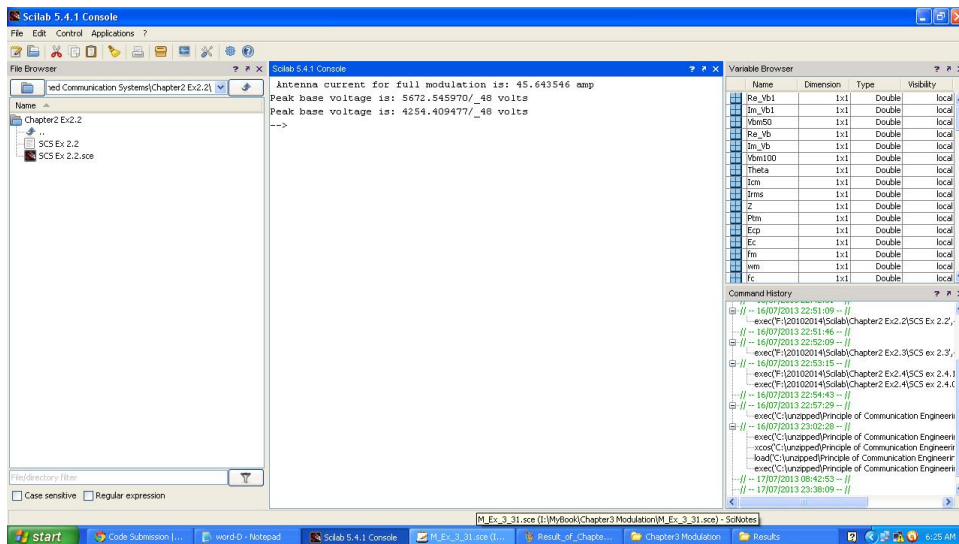


Figure 3.10: Modulation Ex 3 10

```

3 //Chapter3: Modulation
4 //Example3.11, page no 144
5 //Given
6 deff (" [y]=f(x)", "y=Ec*(1+ma*(sin(wm*x)))*sin(wc*x)")
7 Ec=10, ma=0.5, wm=10000*%pi, wc=2*%pi*1e7
8 x=[0:0.01:20]*%pi/10;
9 subplot(2,1,1)
10 fplot2d(x,f)
11 xlabel("t", "fontsize", 3);
12 ylabel("Modulated Wave", "fontsize", 3, "color", "
    red");
13 Fc=wc/(2*%pi)
14 Fm=wm/(2*%pi)
15 Fusb=(wm+wc)/(2*%pi)
16 Flsb=(wm-wc)/(2*%pi)
17 mprintf('USB freq=%d k5Hz\nUSB amplitude=%f V\nLSB
    freq=%d kHz\nLSB amplitude=%f V\nCarrier
    amplitude=%d V', Fusb*1e-3, 2.5, Flsb*-1e-3, 2.5, 10)
18 F=[0, 2.5, 10, 2.5, 0]
19 T=[-2, -1, 0, 1, 2]

```

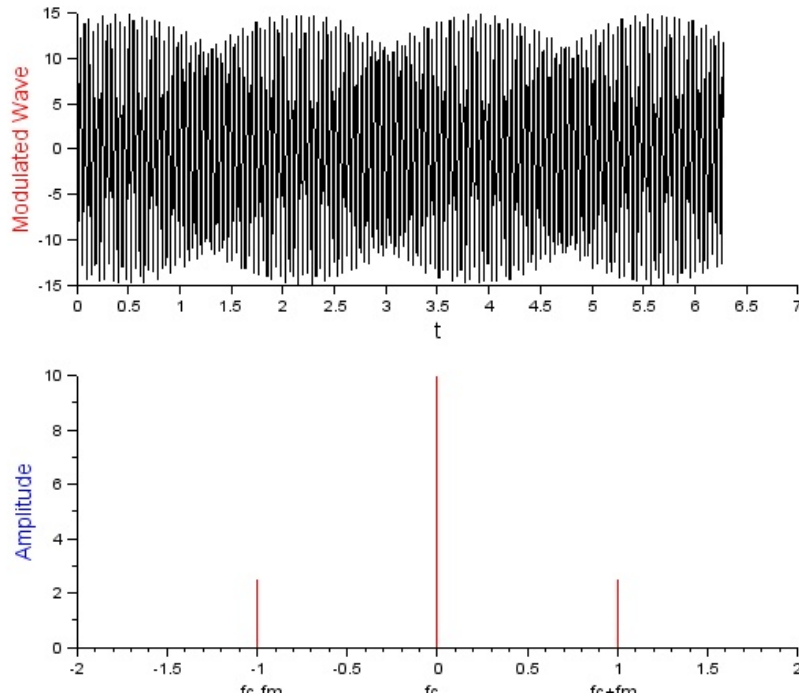



Figure 3.11: Modulation Ex 3 11

```

20 subplot(2,1,2)
21 plot2d3(T,F,5)
22 xlabel("Freq", "fontsize", 3);
23 ylabel("Amplitude", "fontsize", 3, "color", "blue");
24 xlabel("fc - fm"           "fc"
           "fc+fm", "fontsize"
           , 2);

```

Scilab code Exa 3.12 Modulation Ex 3 12

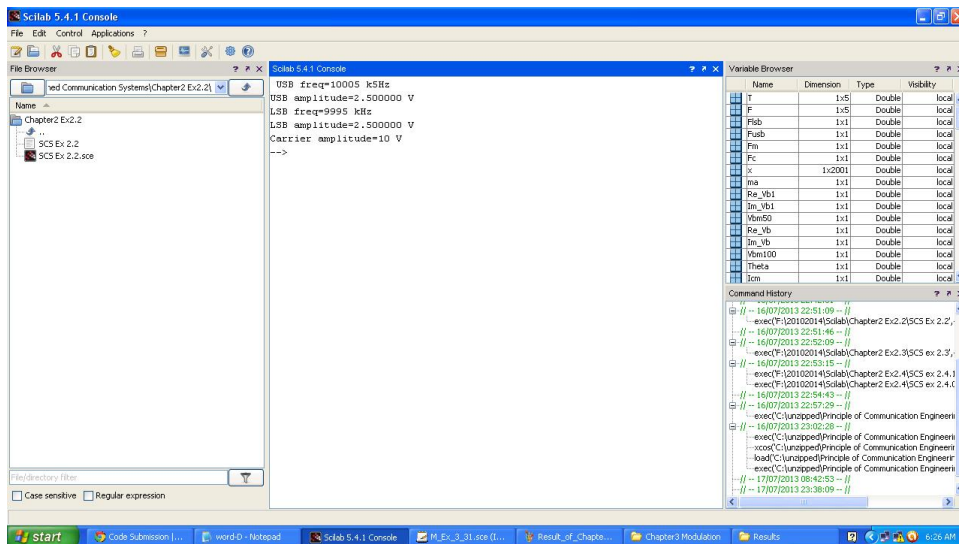


Figure 3.12: Modulation Ex 3 11

```

1  clc
2  //Chapter3: Modulation
3  //Example3.12 page no 145
4  //Given
5  Pc=9e3//unmodulated carrier power
6  Pt=10.125e3//Modulated carrier power
7  Ma=sqrt(2*((Pt/Pc)-1))//depth of modulation
8  mprintf('The depth of modulation is: %d%%',Ma*100, '%
    ')

```

Scilab code Exa 3.13 Modulation Ex 3 13

```

1  clc
2  //Chapter3: Modulation
3  //Example3.13 page no 148
4  //Given

```

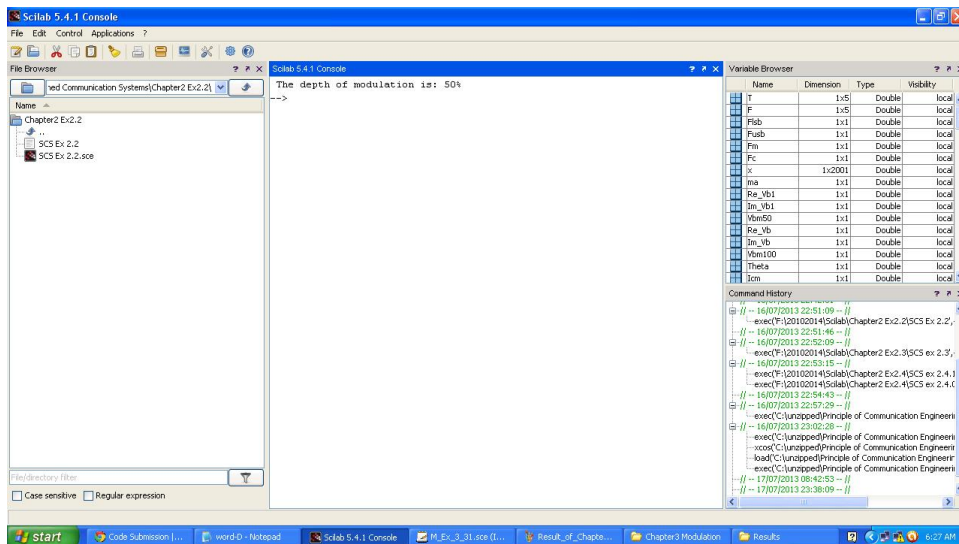


Figure 3.13: Modulation Ex 3 12

```

5 Pt=5e3//carrier power for 95% modulation
6 Ma=0.95
7 Pc=Pt/(1+((Ma^2)/2))//carrier power
8 Ma=0.2//average modulation by speech signal
9 Psb=(Ma^2)*Pc/2//the power n the sideband
10 Pout=Psb/2// because one of the side band is
    suppressed
11 mprintf('The power output is: %f W',Pout)

```

Scilab code Exa 3.14 Modulation Ex 3 14

```

1 clc
2 //Chapter3: Modulation
3 //Example3.14 page no 152
4 //Given

```

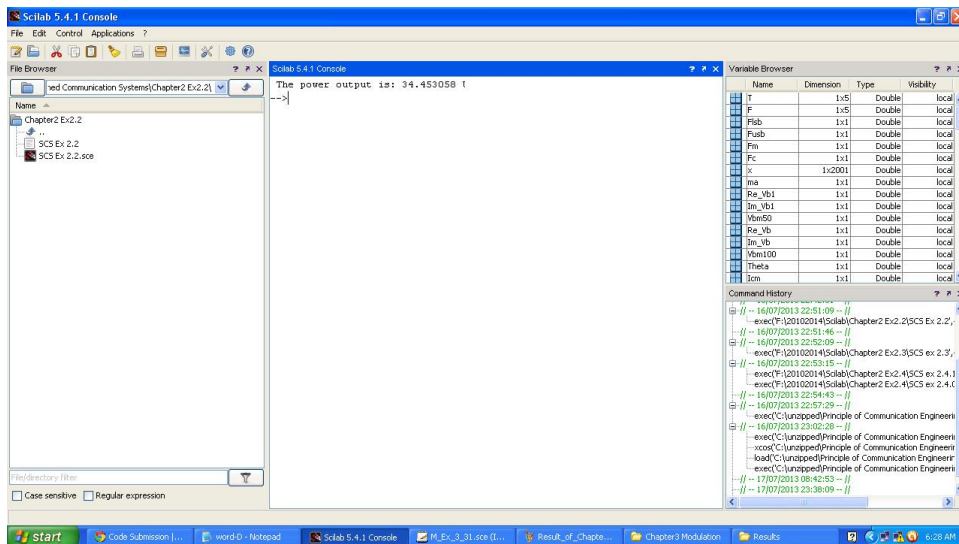


Figure 3.14: Modulation Ex 3 13

```

5 //Phi=(wc*t+Mf*sin(wmt)) ... instantaneous phase of
  FM
6 fm=5000//modulating freq
7 deltaf=50e3//freq deviation
8 deltaPhi1=deltaf/fm// Advance or retard in phase
9
10 fm=100//modulating freq in second signal
11 deltaPhi2=deltaf/fm
12 mprintf('DeltaPhi1= %d rad\nDeltaPhi2=%d rad\n',
  deltaPhi1,deltaPhi2)

```

Scilab code Exa 3.15 Modulation Ex 3 15

```

1 clc
2 //Chapter3: Modulation
3 //Example3.14 page no 157

```

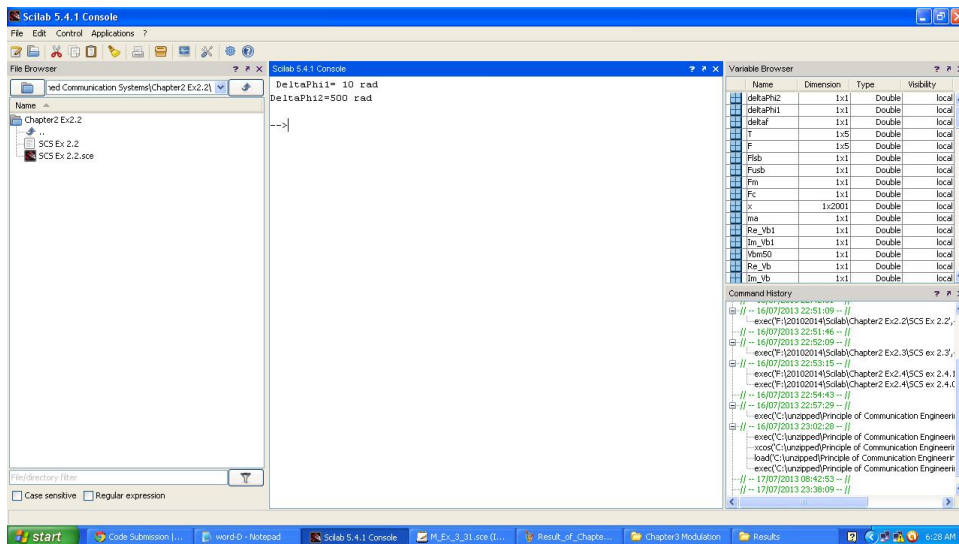


Figure 3.15: Modulation Ex 3 14

```

4 // Given
5 // e=Ec(1+0.4 cos (2 pie3*t))*sin (2 pie7*t)
6 fm=1000 //modulating s/g freq
7 deltaTheta=2*atan (0.4) //peak phase deviation
8
9 deltaF=deltaTheta*fm //Peak freq deviation
10
11 Ec=1
12 Er=sqrt ((Ec^2)*(1+(0.4^2)))
13 m=(Er-Ec)/Ec //depth of residual AM
14
15 AMFr=2*fm // freq of residual AM
16 mprintf ('Peak Phase Deviation: %f rad\nPeak Freq
    Deviation: %d Hz\nDepth of residual AM: %f\n
    Residual AM freq: %d kHz', deltaTheta, deltaF, (
    round (m*100)/100), AMFr*1e-3)

```

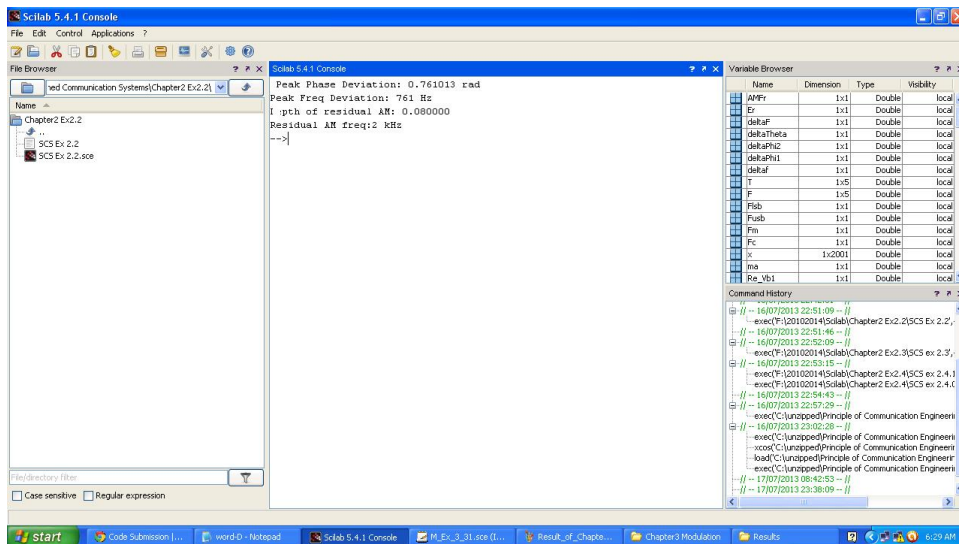


Figure 3.16: Modulation Ex 3 15

Scilab code Exa 3.16 Modulation Ex 3 16

```

1  clc
2  //Chapter3: Modulation
3  //Example3.16 page no 170
4  //Given
5  deltaF=25e3//freq deviation
6  //a
7  fm=100//modulation signal freq
8  mf=deltaF/fm// Max phase deviation
9  disp('a')
10 mprintf('Max phase deviation is:%d rad',mf)
11 //b
12 fm=10e3//modulation signal freq
13 mf=deltaF/fm//Max phase deviation
14
15 disp('b')

```

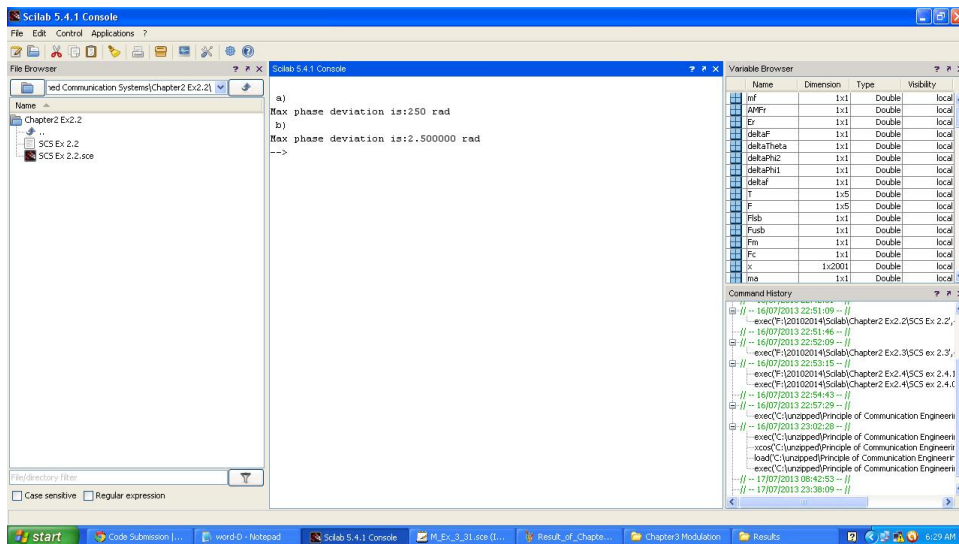


Figure 3.17: Modulation Ex 3 16

16 `mprintf('Max phase deviation is:%f rad',mf)`

Scilab code Exa 3.17 Modulation Ex 3 17

```

1  clc
2  //Chapter3: Modulation
3  //Example3.17, page no 171
4  //Given
5  gm=0.1e-3// trans-conductance variation A/V
6  C=0.5e-12// capacitance between anode and grid
7  R=1e3// resistance
8  fo=10e6// oscillator freq
9  Vrms=1.414//AF RMS voltage
10 Vp=sqrt(2)*Vrms//Peak voltage
11 Ct=100e-12//tank capacitance
12 deltaC=gm*C*R*Vp

```

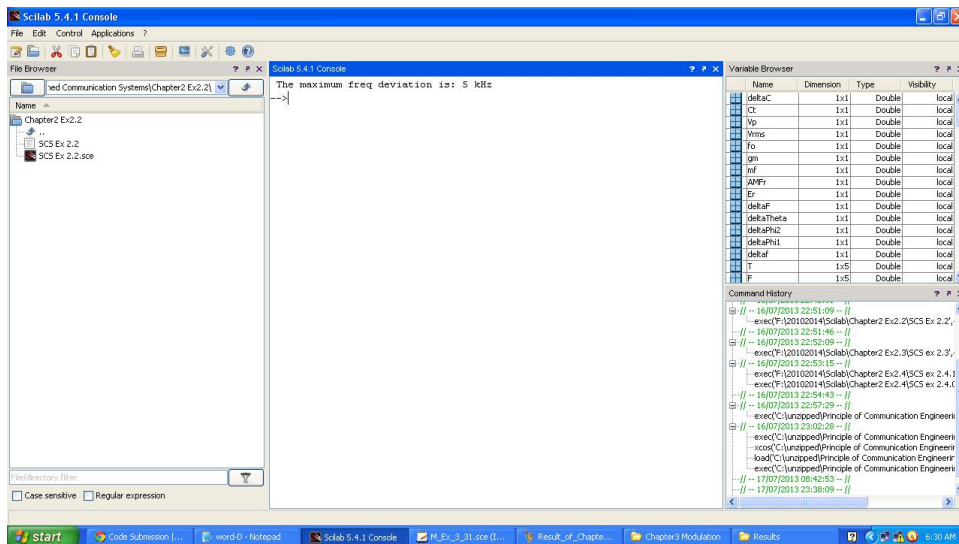


Figure 3.18: Modulation Ex 3 17

```

13
14 deltaF=fo*(deltaC/(2*Ct))// maximum freq deviation
15 mprintf('The maximum freq deviation is: %d kHz',
        round(deltaF/1000))

```

Scilab code Exa 3.18 Modulation Ex 3 18

```

1 clc
2 //Chapter3: Modulation
3 //Example3.18, page no 172
4 //Given
5 deltaF=1e6// max freq deviation
6 fm=10e3//modulating freq
7 mf=(2*deltaF)/fm// modulation coefficient
8 BW=mf*fm// bandwidth

```

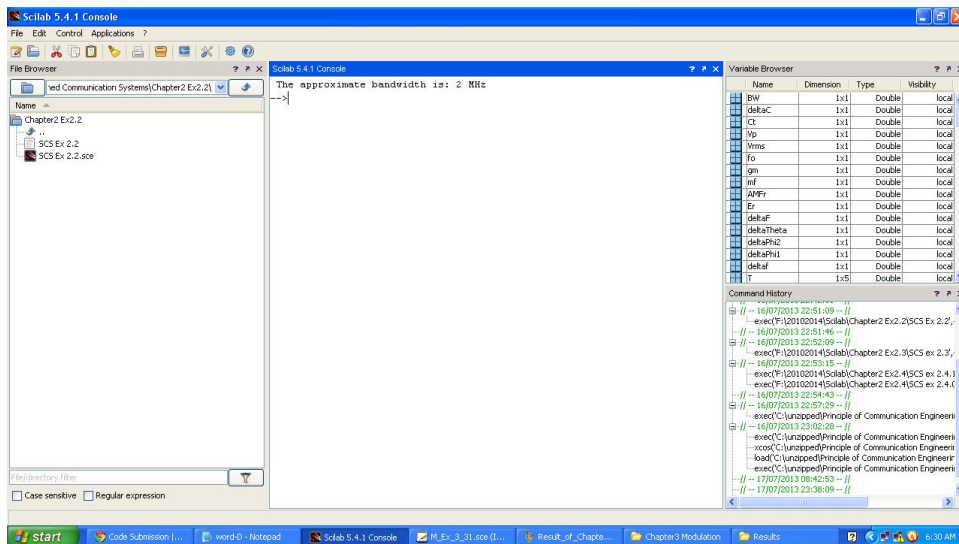



Figure 3.19: Modulation Ex 3 18

```
9 mprintf('The approximate bandwidth is: %d MHz',BW/1e6)
```

Scilab code Exa 3.19 Modulation Ex 3 19

```
1 clc
2 //Chapter3: Modulation
3 //Example3.19, page no 172
4 //Given
5 deltaF=75e3// max freq deviation
6 fm=15e3//modulation freq
7 mf=(2*deltaF)/fm// freq modulation depth
8 BW=mf*fm// Bandwidth
9 mprintf('The approximate bandwidth is: %d kHz',BW/1e3)
```

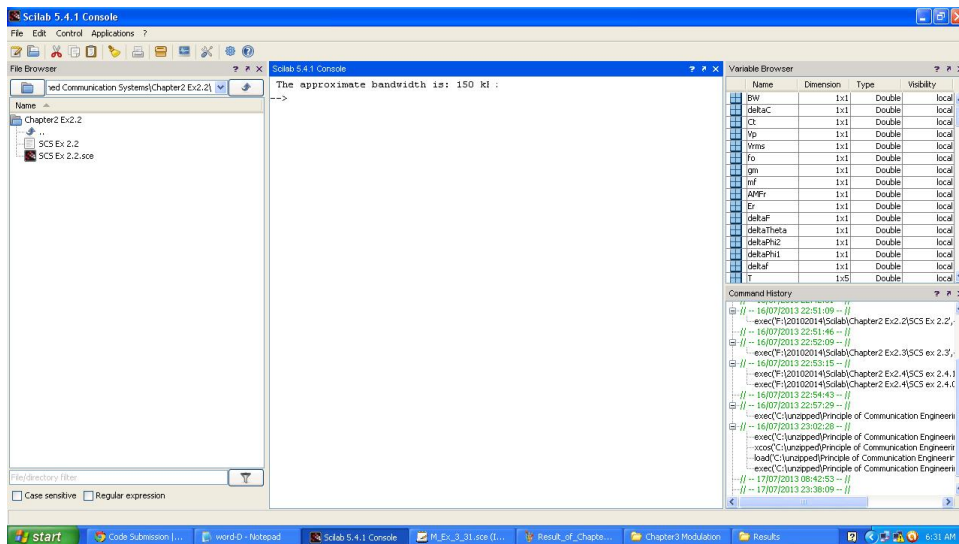


Figure 3.20: Modulation Ex 3 19

Scilab code Exa 3.21 Modulation Ex 3 21

```

1  clc
2  //Chapter3: Modulation
3  //Example3.21, page no 173
4  //Given
5  deltaF=75e3//freq deviation
6  fm=15e3// modulating freq
7  mf=deltaF/fm
8  BW=2*mf*fm// Bandwidth
9  GB=25e3//Guard Band
10 BWo=BW+(2*GB) // Overall bandwidth
11 mprintf('Overall bandwidth including guard band is
           %d kHz ',BWo/1e3)

```

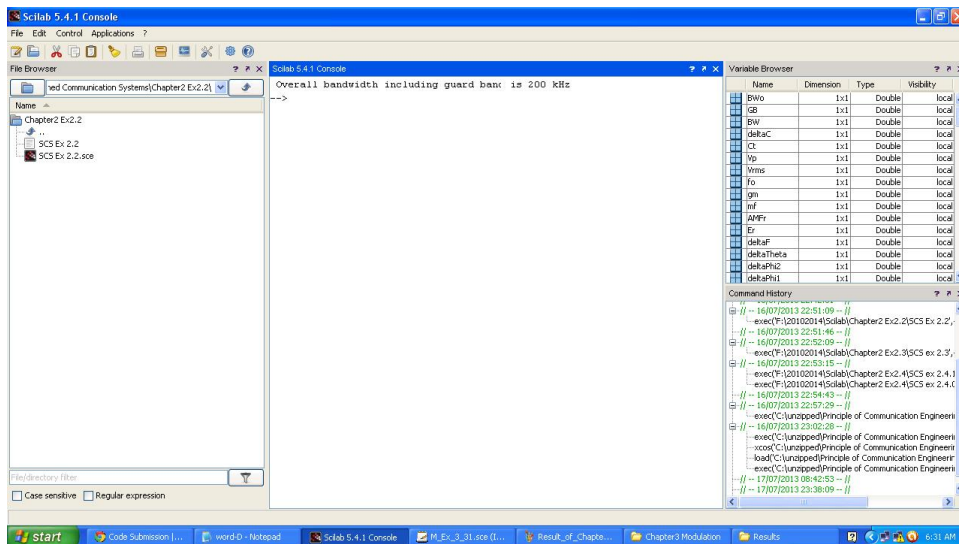


Figure 3.21: Modulation Ex 3 21

Scilab code Exa 3.25 Modulation Ex 3 25

```

1  clc
2  //Chapter3: Modulation
3  //Example3.25, pageno 175
4  //Given
5  //em=3sin(2*pi*1000t)+5cos(2*pi*3000t)
6  //ec=50sin(2*pi*500e3*t)
7  m1=0.06//(sine wave amplitude/ peak carrier voltage)
8  m2=0.1//(cosine wave amplitude/ peak carrier voltage
9  )
10 Vc=50//Carrier voltage
11 R=50//load resistance
12 Pc=(Vc^2)/(2*R)//Peak carrier power
13 Pt=Pc*(1+((m1^2+m2^2)/2))//Total power after

```

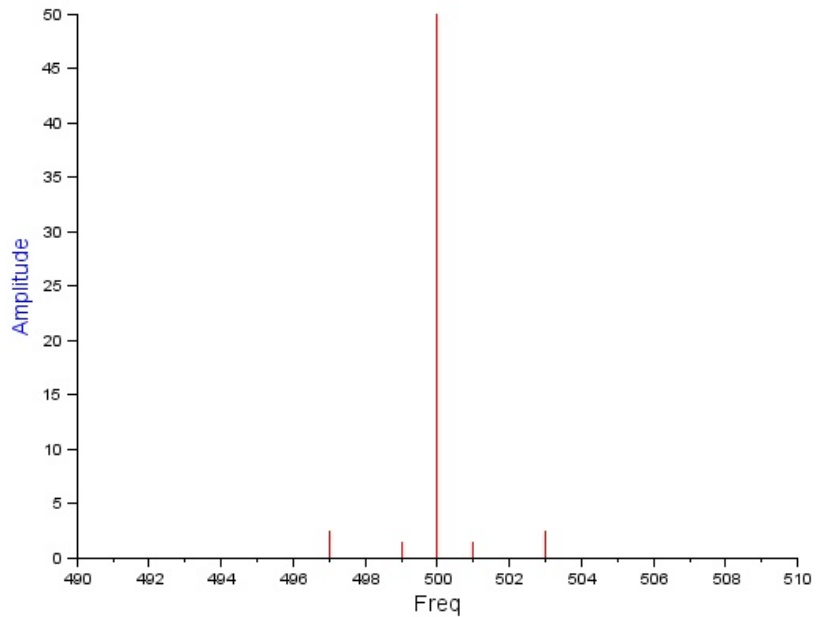


Figure 3.22: Modulation Ex 3 25

```

modulation
13 mprintf('Average power is: %f watts',Pt)
14 F=[0,2.5,1.5,50,1.5,2.5,0]
15 T=[490,497,499,500,501,503,510]
16 plot2d3(T,F,5)
17 xlabel("Freq", "fontsize",3);
18 ylabel("Amplitude", "fontsize",3, "color", "blue");

```

Scilab code Exa 3.26 Modulation Ex 3 26

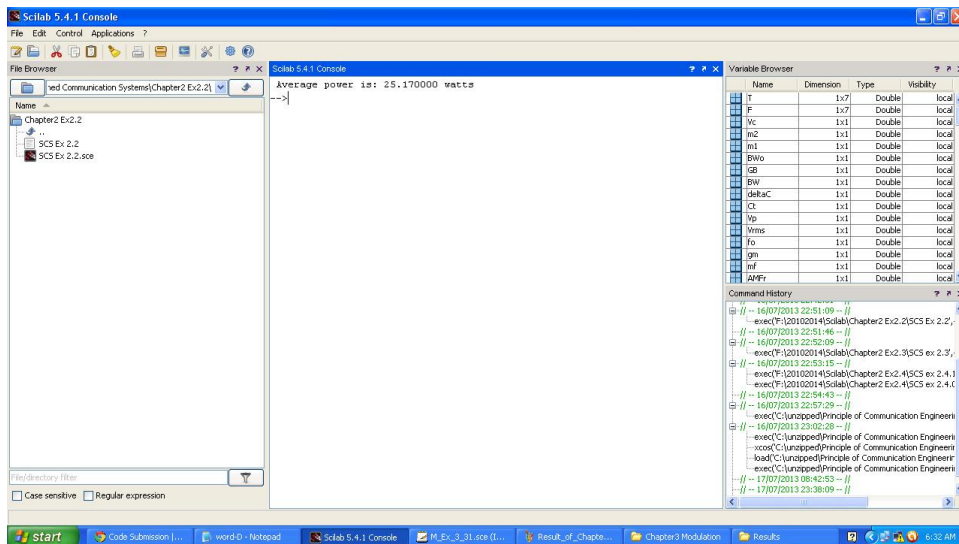


Figure 3.23: Modulation Ex 3 25

```

1  clc
2  //Chapter3: Modulation
3  //Example3.26 , page no 176
4  //Given
5  mp=0.1//Modulating index
6  fm=400//Modulating signal freq
7  deltaF=mp*fm//Max freq deviation
8  //disp(deltaF)
9  ReqDev=50e3// Required deviation
10 MF=ReqDev/deltaF// multiplication factor
11 mprintf('Required Deviation is: %d kHz\n',ReqDev/1e3
12         )
12 mprintf('Required Multiplication Factor is: 5*5*5*5*2
13         ')

```

Scilab code Exa 3.27 Modulation Ex 3 27

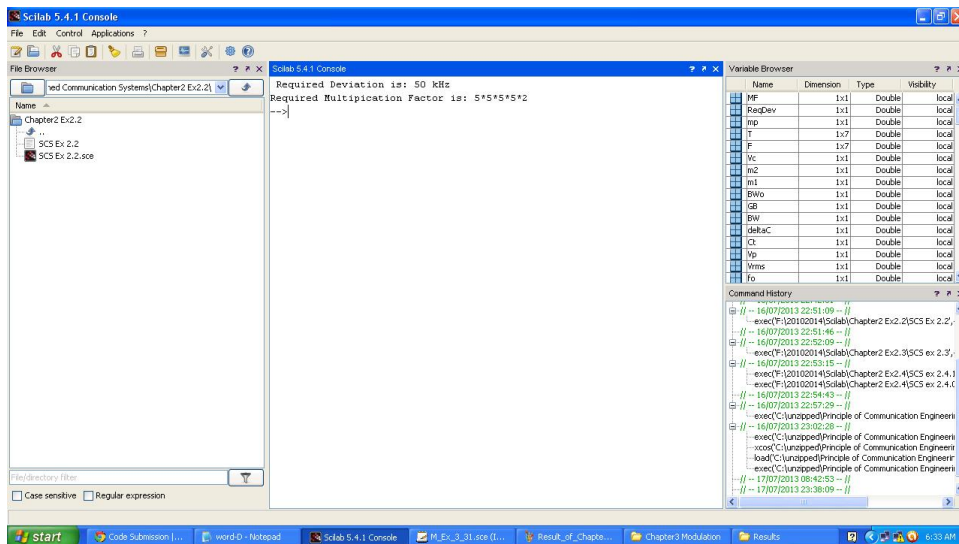


Figure 3.24: Modulation Ex 3 26

```

1  clc
2  //Chapter3: Modulation
3  //Example3.27, page no 176
4  //Given
5  Q=100 //Q factor
6  fc=1000e3// Carrier freq
7  fsb1=999e3//lower Side band freq
8  fsb2=1001e3//Upper side Band freq
9  ma=0.5//Modulation depth of signal current
10 Ma=ma/1.019// Expression for Ma after simplification
11 mprintf('The Depth of modulation across the \n
           circuit is : Ma= %f%c',Ma*100,'%')
12
13 // Note : There are some calculation errors in the
           solution presented in the book

```

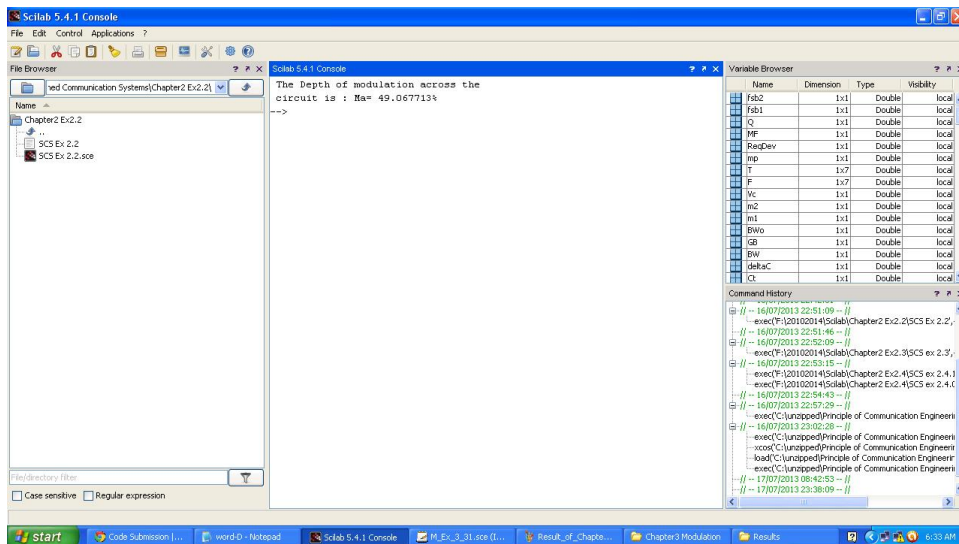


Figure 3.25: Modulation Ex 3 27

Scilab code Exa 3.28 Modulation Ex 3 28

```

1  clc
2  //Chapter3: Modulation
3  //Example3.28, page no 177
4  //Given
5  R=1//Antenna Resistance assumed to be 1 ohm for ease
      of calculation
6  Ic=10.8// current with no modulation
7  Pc=Ic^2*R//power with no modulation
8  It=12.15//modulated current
9  Pt=It^2*R// modulated power
10 ma=(sqrt(2*((It/Ic)^2)-1))//modulation depth)
11
12 mprintf('Depth of modulation:%f %c',round(1000*ma)
      /10, '%');

```

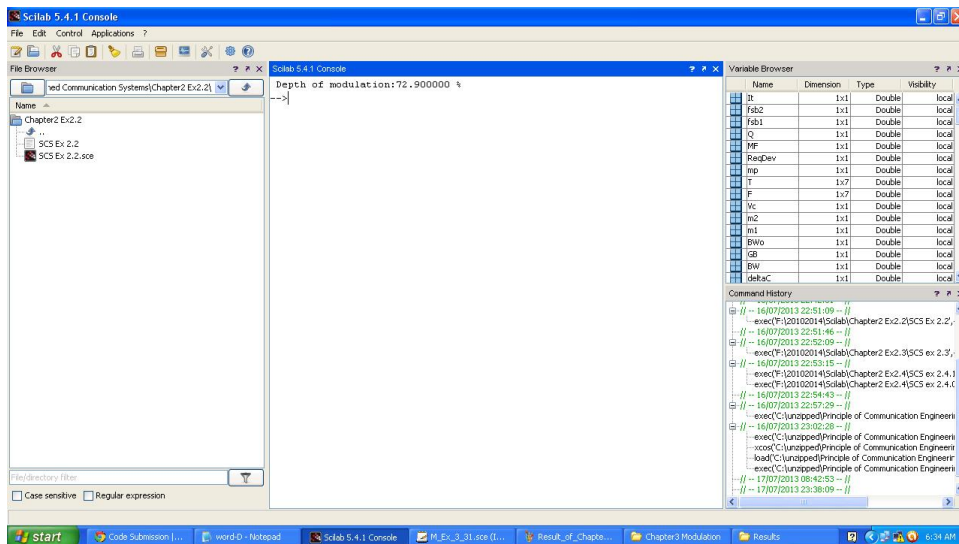


Figure 3.26: Modulation Ex 3 28

Scilab code Exa 3.29 Modulation Ex 3 29

```

1  clc
2  //Chapter3: Modulation
3  //Example3.29, page no 177
4  //Given
5  Pc=100e3//Carrier power
6  ma=0.5//Depth of modulation
7  Pt=Pc*(1+((ma^2)/2))//total RF power
8  mprintf('Total RF power delivered is :Pt= %f kW',Pt/1
          e3)

```

Scilab code Exa 3.30 Modulation Ex 3 30

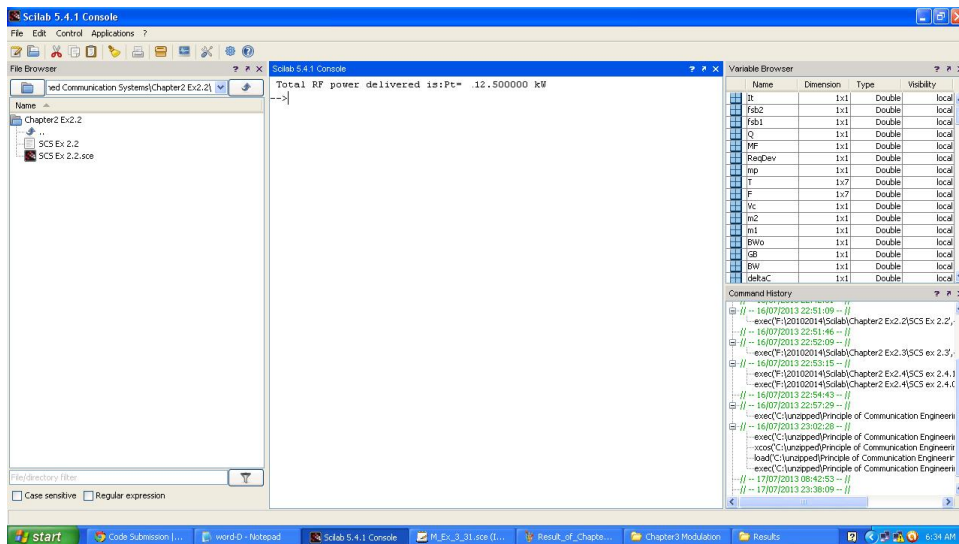


Figure 3.27: Modulation Ex 3 29

```

1  clc
2  //Chapter3: Modulation
3  //Example3.30, page no 178
4  //Given
5  Pt=100e3// Total power
6  ma=0.9//Depth of modulation
7  Pc=Pt/(1+((ma^2)/2))//Carrier power
8  Psb=Pt-Pc// Intelligence power i.e sideband power
9  mprintf('Carrier power:%f kW\nThe Intelligence power
           : %f kW',Pc/1000,Psb/1000)

```

Scilab code Exa 3.31 Modulation Ex 3 31

```

1  clc
2  //Chapter3: Modulation
3  //Example3.19, page no 178

```

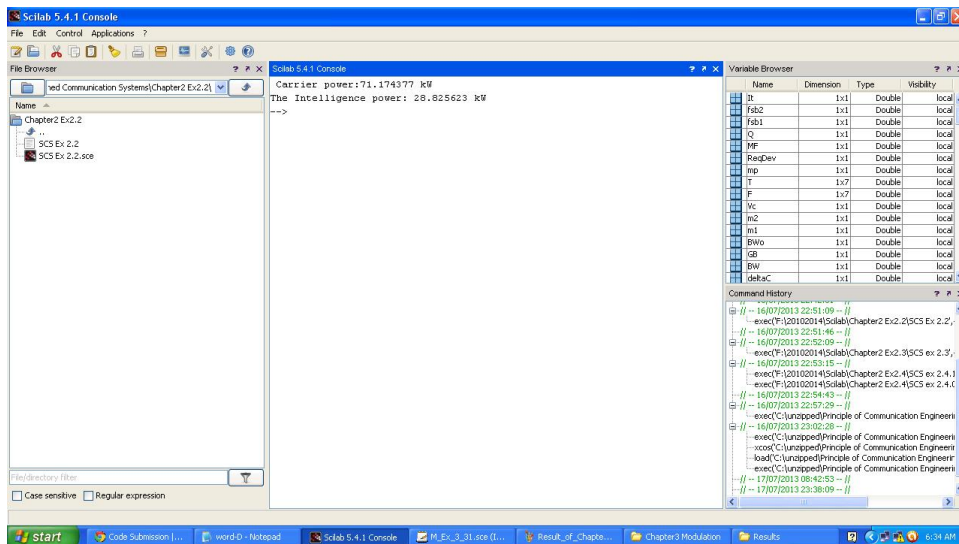


Figure 3.28: Modulation Ex 3 30

```

4 //Given
5 R=1// load resistance
6 Eo=100//RF voltage
7 Po=Eo^2/R// Carrier power
8 E=110//Modulated RMS voltage
9 Pt=E^2/R//Total modulated power
10 ma=sqrt(2*((Pt/Po)-1))// Depth of modulation
11 mprintf('Modulation Index is: %f %c',ma*100,'%')

```

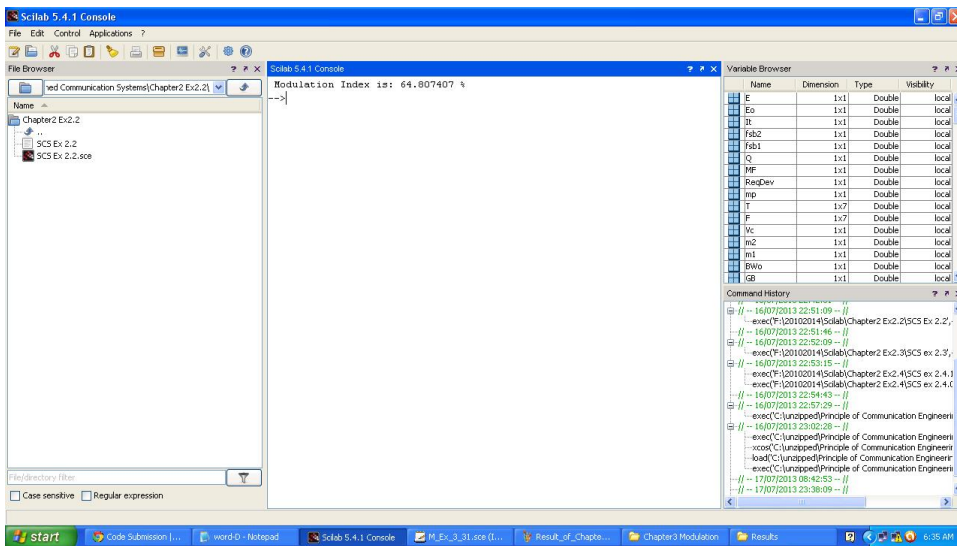


Figure 3.29: Modulation Ex 3 31

Chapter 5

Radio Transmission System

Scilab code Exa 5.1 RTS Ex 5 1

```
1 clc
2 //Chapter8
3 //Example8.15, page no 230
4 //Given
5 //b
6 fm=1e2//modulation freq
7 Phimax=10*%pi/180// Max Phase deviation
8 //i
9 Freq_dev=Phimax*fm// Freq deviation
10 //ii
11 Mul_fact=30e3/Freq_dev// Multification factor
12 mprintf('Freq deviation is %f Hz\n Multification
    factor is %d\n corresponding modified max freq
    deviation is 30114kHz',Freq_dev,Mul_fact)
```

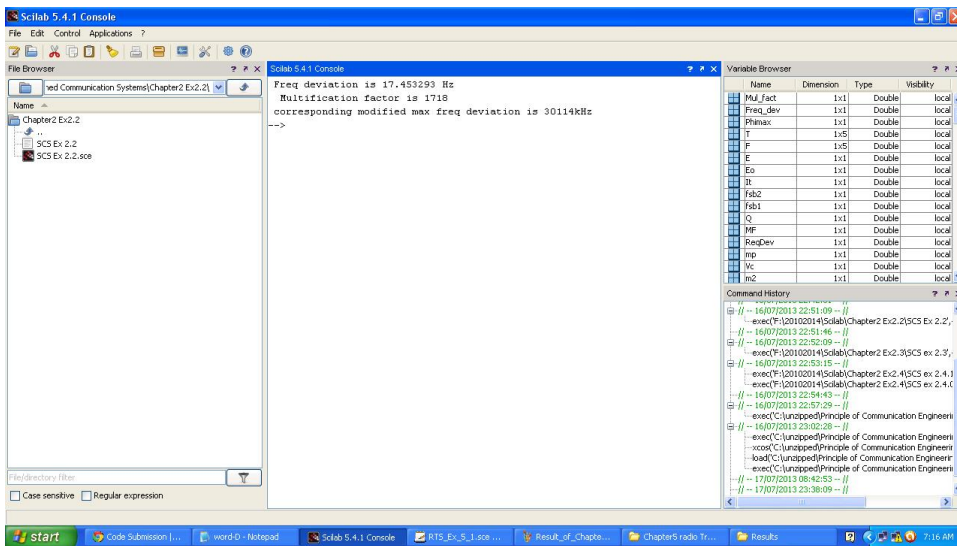


Figure 5.1: RTS Ex 5 1

Chapter 6

Radio Receivers

Scilab code Exa 6.1 Radio Receivers Ex 6 1

```
1 clc
2 //Chapter8
3 //Example8.15, page no 262
4 //Given
5 //Vm(t),Vc(t),Vmod(t)
6 fm=10e3//modulating freq
7 BW=2*fm// Bandwidth
8 fc=100*BW// Carrier freq
9 mprintf('Carrier freq for the BW to be 1%c of fc is:
    %d kHz ', '%',fc/1000)
```

Scilab code Exa 6.2 Radio Receivers Ex 6 2

```
1 clc
2 //Chapter8
3 //Example6.2, page no 262
4 //Given
```

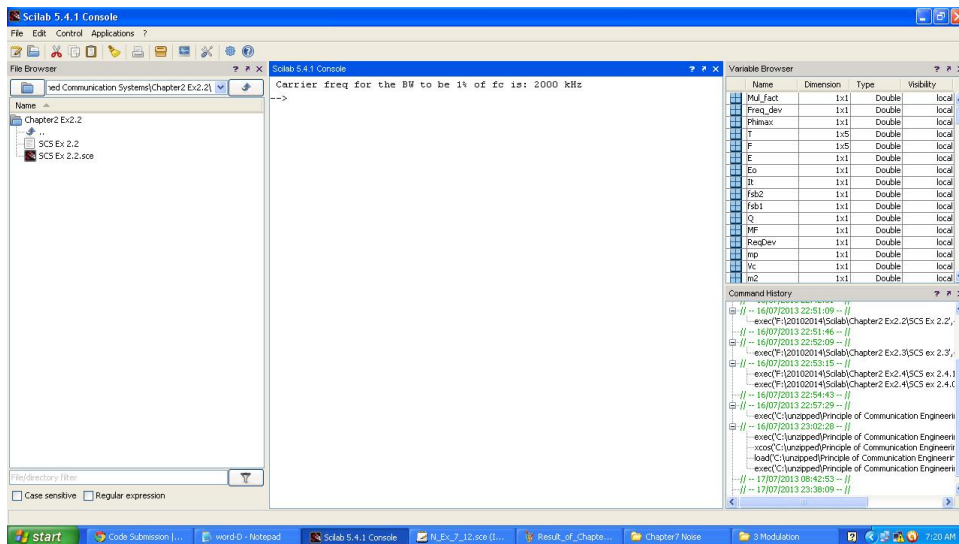


Figure 6.1: Radio Receivers Ex 6 1

```

5 fmax=1600e3 , fmin=500e3 , IF=465e3
6 // i
7 fo1max=fmax+IF , fo1min=fmin+IF
8 C1max_C1min=(fo1max/fo1min)^2
9 // ii
10 fo2max=fmax-IF , fo2min=fmin-IF
11 C2max_C2min=(fo2max/fo2min)^2
12 mprintf('a)\nTuning capacitor range is: %f\nb)\n
    Tuning capacitor range is: %d',C1max_C1min,
    C2max_C2min)

```

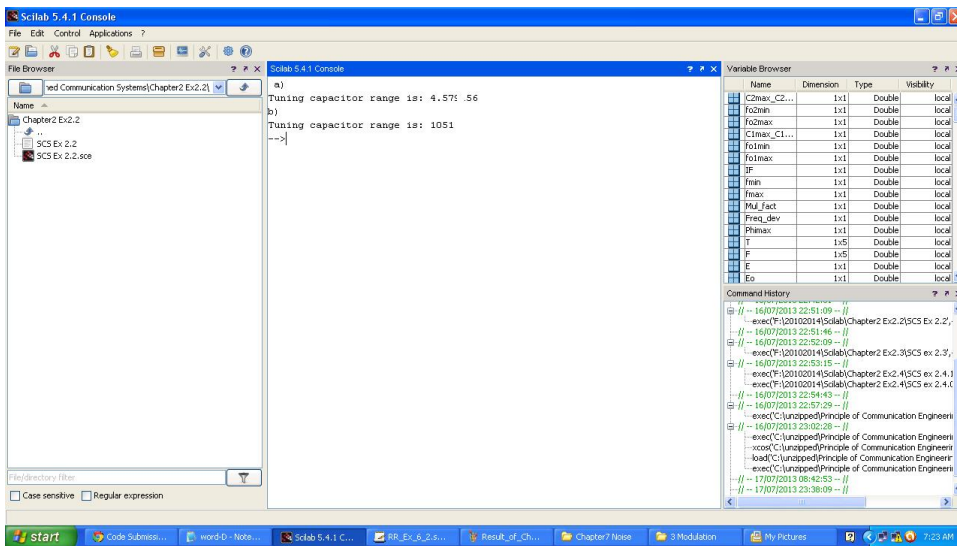


Figure 6.2: Radio Receivers Ex 6 2

Chapter 7

Noise

Scilab code Exa 7.2 Noise Ex 7 2

```
1  clc
2  //Chapter7
3  //Example7.2, page no 276
4  //Given
5  mue=25//
6  rp=5e3
7  Rl=10e3
8  C=1e-9
9  gm=mue/rp
10 Req=2.5/gm
11 //disp(Req , gm)
12 k=1.381e-23
13 T=293
14 R1=1e5
15 // Power density spectrum for respective res
16 d1=2*k*T*R1
17 d2=2*k*T*Req
18 d3=2*k*T*Rl
19 xo=0
20 x1=1e14
21 w=0:%inf
```

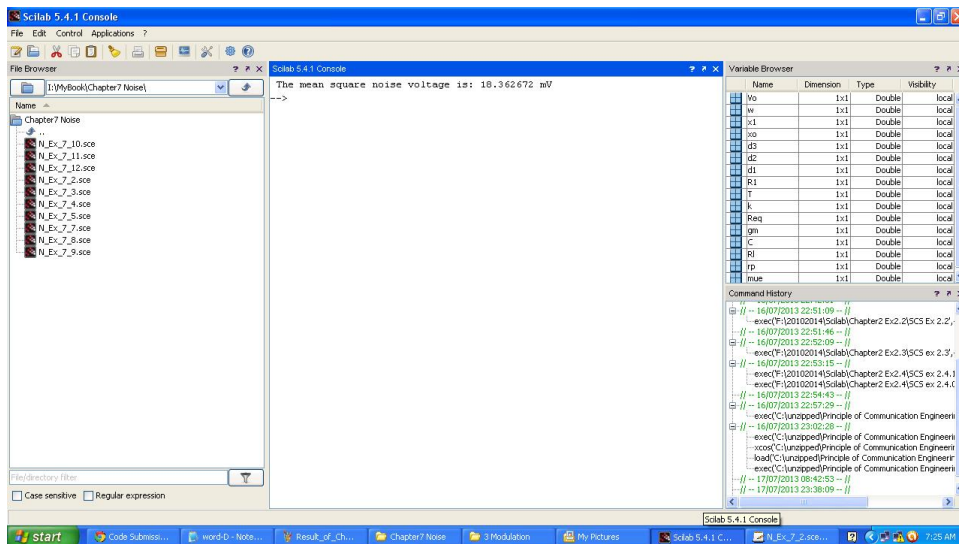


Figure 7.1: Noise Ex 7 2

```

22 //H1(w)=(-gm*rp*Rl)/(rp+Rl+(%i*w*rp*Rl*C))
23 Vo=sqrt((20231.65e2/%pi)*integrate('1/(((3e9)^2)+(w
    ^2))', 'w', xo, x1))
24 mprintf('The mean square noise voltage is: %f mV',Vo
    *1e3)

```

Scilab code Exa 7.3 Noise Ex 7 3

```

1 clc
2 //Chapter7
3 //Example7.3, page no 279
4 //Given
5 mue=25
6 rp=5e3
7 Rs=1e3//input resistance
8 //Coupling Capacitors are assumed as short circuit

```

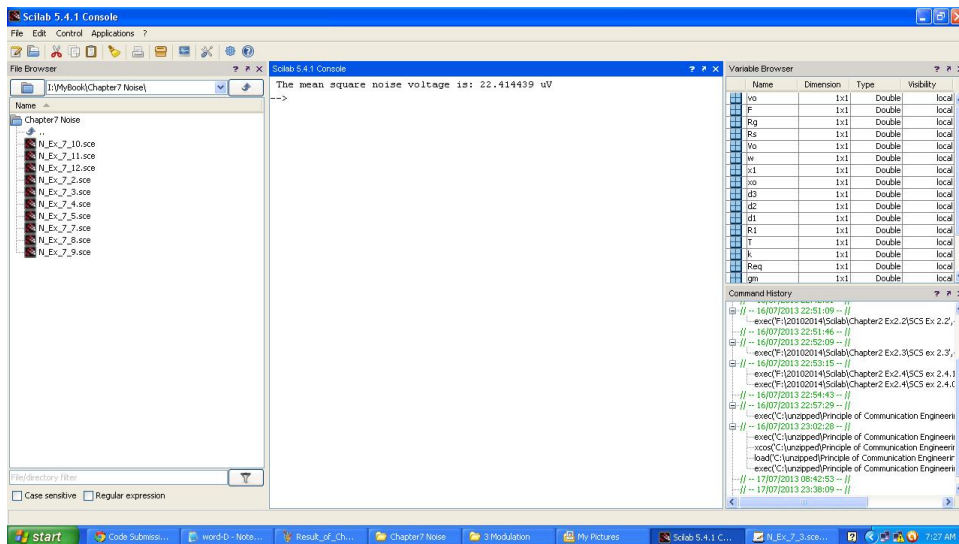


Figure 7.2: Noise Ex 7 3

```

9 Rg=1e5
10 gm=25/5e3
11 Req=2.5/gm
12 F=1+(((Req*(Rs+Rg)^2)+(Rg*Rs^2))/(Rs*Rg^2))
13 xo=0
14 x1=1e10
15 w=0:%inf
16
17 vo=sqrt((30145e-8/%pi)*integrate('1/(((3e5)^2)+(w^2)
    )','w',xo,x1))
18 mprintf('The mean square noise voltage is: %f uV',vo
    *1e6)

```

Scilab code Exa 7.4 Noise Ex 7 4

```
1 clc
```

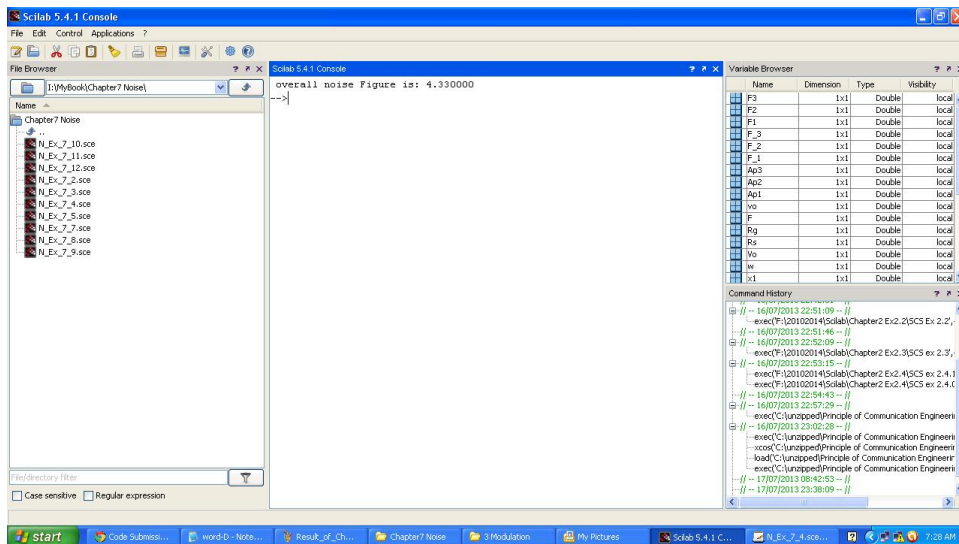


Figure 7.3: Noise Ex 7 4

```

2 //Chapter8
3 //Example7.4, page no 283
4 //Given
5 Ap1=10, Ap2=10, Ap3=10; // Gain of each states
6 F_1=6, F_2=6, F_3=6; //Noise figure of each state
7 F1= round(10^(F_1/10)), F2= round(10^(F_2/10)), F3=
   round(10^(F_3/10)); // approximating the values
8
9 F=F1+((F2-1)/Ap1)+((F3-1)/(Ap1*Ap2))
10 mprintf('overall noise Figure is: %f',F)

```

Scilab code Exa 7.5 Noise Ex 7 5

```

1 clc
2 //Chapter7
3 //Example7.5, page no 283

```

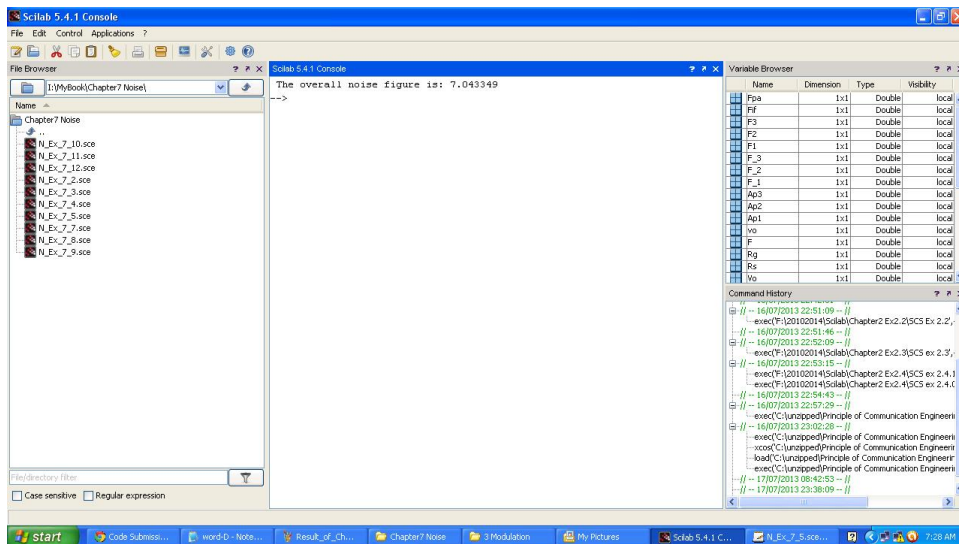


Figure 7.4: Noise Ex 7 5

```

4 //Given
5 Fif=15// Noise figure of IF amplifier
6 Ap1=10// Gain of Preamplifier
7 Fpa=6//Noise figure of preamplifier
8 F2=10^(Fif/10)
9 F1=10^(Fpa/10)
10
11 F=F1+((F2-1)/Ap1)//overall noise figure
12 mprintf('The overall noise figure is: %f',F)

```

Scilab code Exa 7.7 Noise Ex 7 7

```

1 clc
2 //Chapter7
3 //Example7.6
4 //Given

```

```

5  mue=25// tube parameters
6  rp=10e3// tube parameters
7  gm=2.5e-3// transconductance
8  Req=2.5/gm// equivalent resistance
9  Rs=1000
10 Rg=1e5
11 F1=1+(((Req*((Rs+Rg)^2))+Rg*Rs^2)/(Rs*(Rg^2)))//
    noise figure of the first stage
12 Rg2=9.1e3
13 Rs2=10e3
14 Es=1// assuming Es=1 for ease of calculation
15 Pi=((Es/2e3)^2)*1e3
16 Po=1.532e-2*Es^2
17 Ap1=Po/Pi
18 F2=1+(((Req*((Rs2+Rg2)^2))+Rg2*Rs2^2)/(Rs2*(Rg2^2)))
    // noise figure of the second stage
19 F=(F1)+((F2-1)/Ap1)
20 mprintf('Overall Noise figure is:%f',F)

```

Scilab code Exa 7.8 Noise Ex 7 8

```

1  clc
2  //Chapter7
3  //Example7.8
4  //Given
5  g01=30// gain of 1st stage
6  g02=20//gain of 2nd stage
7  g03=40//gain of 3rd stage
8  F2=6// Noise factor of stage 2
9  F3=12// Noise factor of stage 3
10 Te1=4// Eq noise temp of stage 1
11 T=290// Room
12 G01=round(10^(g01/10))

```

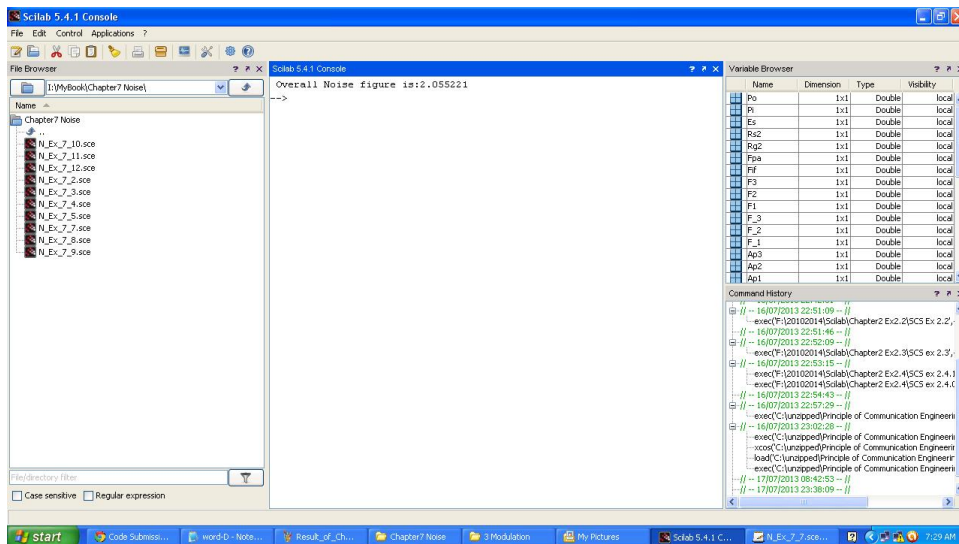


Figure 7.5: Noise Ex 7 7

```

13 G02=round(10^(g02/10))
14 G03=round(10^(g03/10))
15 F_2=round(10^(F2/10))
16 F_3=round(10^(F3/10))
17 Te2=round((F_2-1))*T
18 Te3=round((F_3-1))*T
19 Te=Te1+(Te2/G01)+(Te3/(G01*G02))// Eq overall noise
    temp
20 mprintf('The equivalent noise temp is: %f K',Te)

```

Scilab code Exa 7.9 Noise Ex 7 9

```

1 clc
2 // Chapter7
3 // Example7.9
4 // Given

```

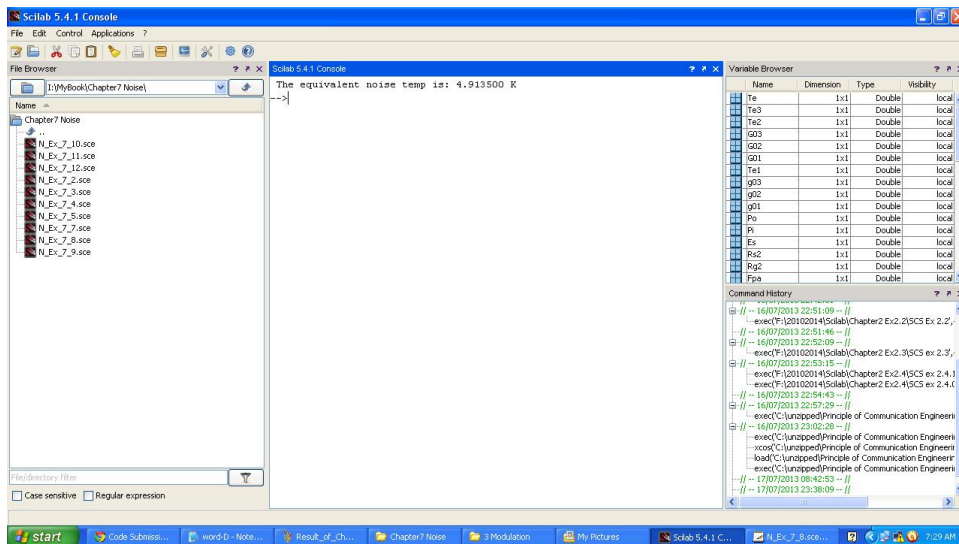


Figure 7.6: Noise Ex 7 8

```

5 g01=round(10^(25/10))//low noise amplifier gain
6 Te1=4//low noise amplifier noise temp
7 g02=round(10^(1.7))//preamplifier gain
8 F2=round(10^0.6)//preamplifier noise figure
9 F3=round(10^1.2)//preamplifier noise figure
10 T=290// room temp
11 Te2=round((F2-1)*T)
12 Te3=round((F3-1)*T)
13 Te=Te1+(Te2/g01)+(Te3/(g01*g02))//Overall noise
    Temperature
14 mprintf('Equivalent noise temperature is %f K',Te)

```

Scilab code Exa 7.10 Noise Ex 7 10

1
2

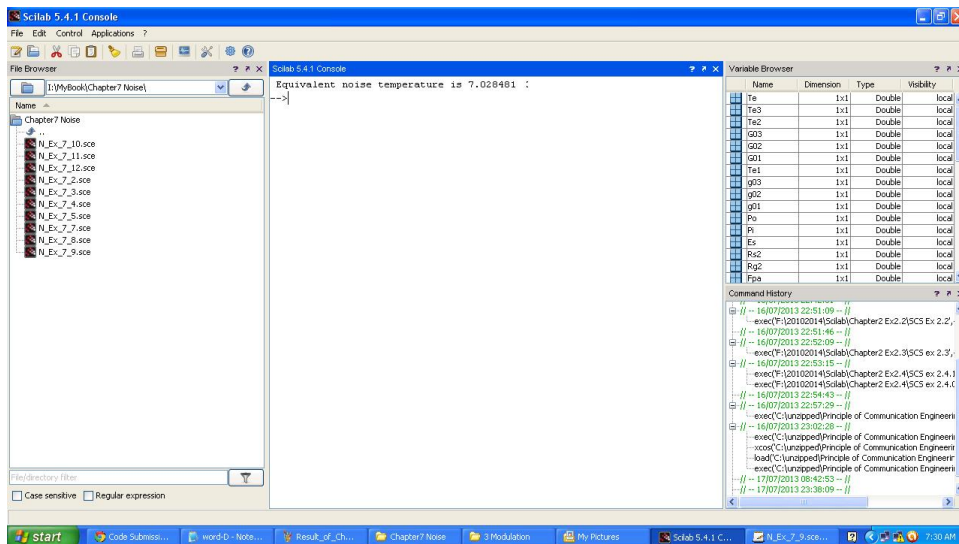


Figure 7.7: Noise Ex 7 9

```

3  clc
4  //Chapter7
5  //Example7.10
6  //Given
7  SNRfm=25// Signal to noise ratio of AM
8  PcFM_AM=0.9//
9  mf=5
10 SNRfm=(10*log10(3*(mf^2)*(PcFM_AM)))+SNRfm
11 mprintf('S/N ratio for FM is %f dBs',SNRfm)
12 // Note : There are some calculation errors in the
    solution presented in the book

```

Scilab code Exa 7.11 Noise Ex 7 11

```

1  clc
2  //Chapter7

```

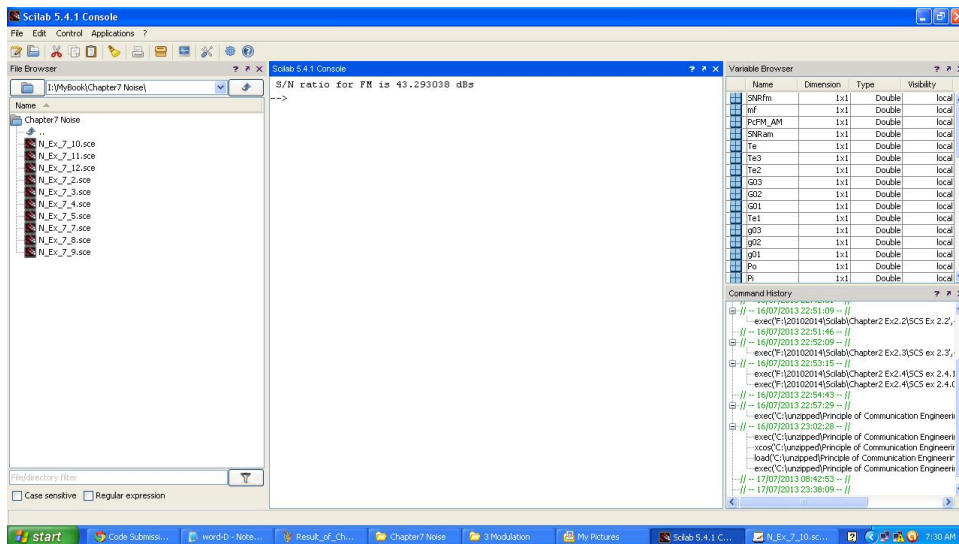


Figure 7.8: Noise Ex 7 10

```

3 //Example7.11
4 //Given
5 ma=0.3
6 SNR=20// s/n ratio
7 SNR1=10^(0.1*SNR)
8 SNR_new=SNR+3
9 ma2=0.6// increased new depth of modulation
10 Pt_Ni=SNR1*((1+(ma^2))/(ma^2))
11 SNR2=10*log10(Pt_Ni*((ma2^2)/(1+((ma2^2)/2))))
12
13 mprintf('a)\n New SNR for 3dB increase in input s/g
         is %d dBs\n\b) When Modulation depth is
         increased to 60%c\n SNR becomes %f dBs',SNR_new,'
         %',SNR2)

```

Scilab code Exa 7.12 Noise Ex 7 12

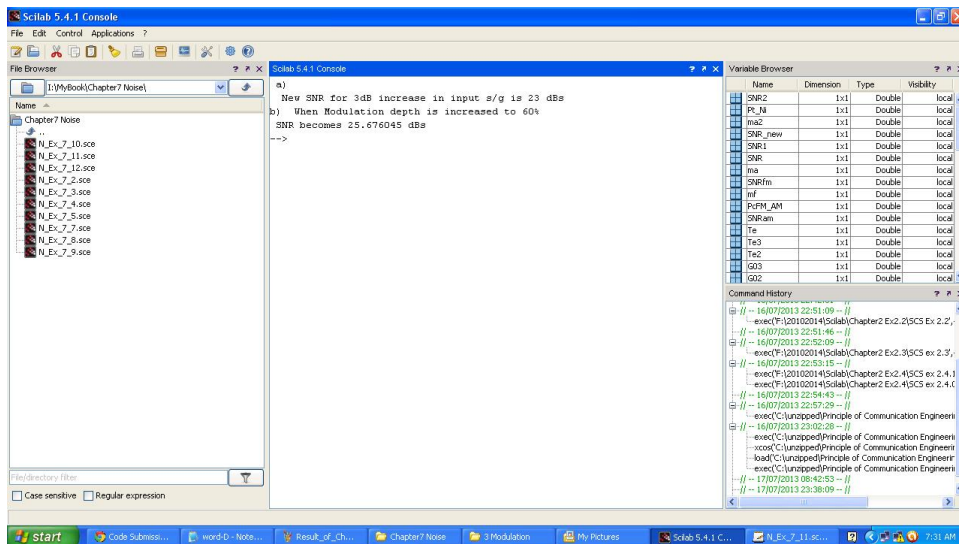


Figure 7.9: Noise Ex 7 11

```

1  clc
2  //Chapter7
3  //Example7.12
4  //Given
5  fmax=5e3//max s/g freq
6  S_fmin=2*fmax// Min sampling freq
7  B_S=6//Binary bits sent per sample
8  BTR=B_S*S_fmin//Bit Transmission rate
9  Q=2^B_S//No of Quantizable levels
10 MQN=0.5/Q//Max Quantization noise
11 S_QNR=MQN^-1// Signal to Quantization noise ratio
12 //b
13 S_QNRreq=0.5*S_QNR// Signal to Quantization noise
    ratio
14 Qreq=0.5*S_QNRreq//No of Quantizable levels
15 B_Sreq=log2(Qreq)//Binary bits sent per sample
16 mprintf('a) Bit Transmission rate: %d kbits/s\n
    Signal to Quantization noise ratio %d \n)\n Bit
    Transmission rate: %d kbits/sample\n    Signal
    to Quantization noise ratio: %d',BTR/1000,S_QNR,

```

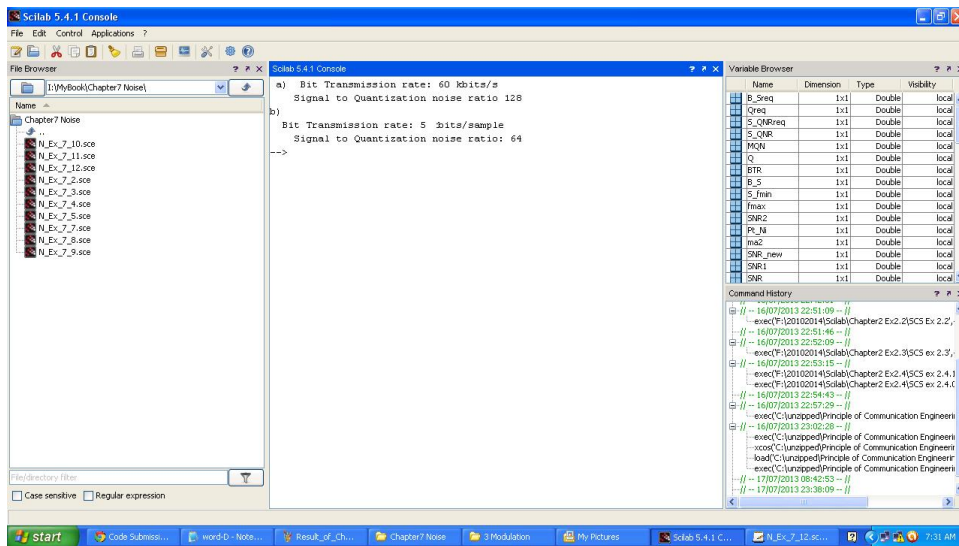


Figure 7.10: Noise Ex 7 12

B_Sreq, S_QNRreq)

Chapter 8

Transmission Line

Scilab code Exa 8.1 Transmission Line Ex 8 1

```
1  clc
2  //Chapter8
3  //Example8.1, page no 313
4  //Given
5  //a
6  L=1.2*10^-3//distributed inductance
7  C=0.05*10^-6//distributed capacitance
8  Zo=sqrt(L/C)//Characteristic Impedance
9  mprintf('The characteristic Impedance is Zo= %f ohm',
           ,Zo)
10 Wo=1// Assumedfor ease of calculation
11 G=%i*sqrt(L*C)*Wo
12 mprintf('\nPropagation constant is Gama= j%3.2ew',G
           *-%i)
13 //b
14 //i
15 lambda=0.4e3//wavelength=Line length
16 c=3e8
17 f=c/lambda
```

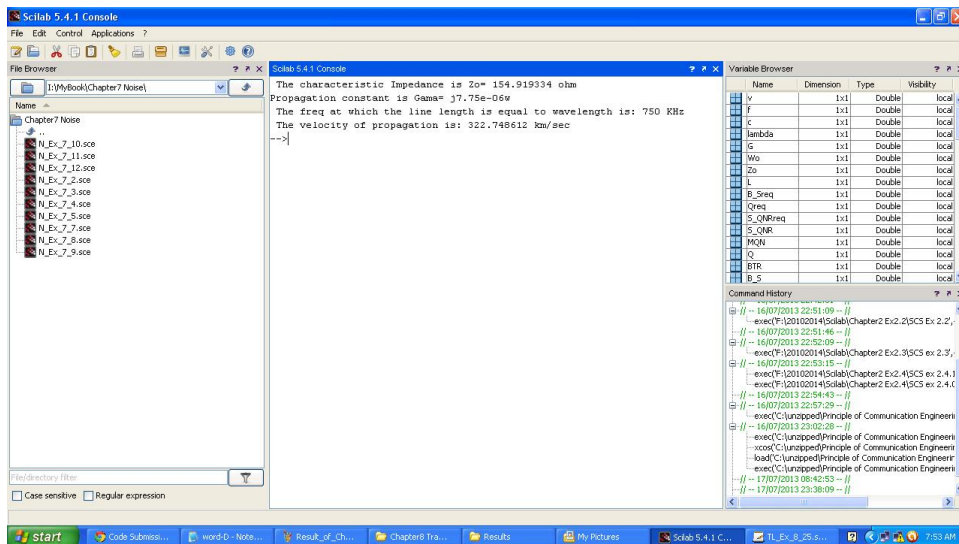


Figure 8.1: Transmission Line Ex 8 1

```

18 // ii
19 L=L '*0.4
20 C=C '*0.4
21 v=1/(sqrt(L*C))
22 mprintf('\n The freq at which the line length is
    equal to wavelength is: %d KHz\n The velocity of
    propagation is: %f km/sec ',f*1e-3,v*1e-3)

```

Scilab code Exa 8.2 Transmission Line Ex 8 2

```

1 clc
2 //Chapter8
3 //Example8.2, page no 314
4 //Given
5 v=3e8// velocity of light
6 f=1.2e6// Operating Freq

```

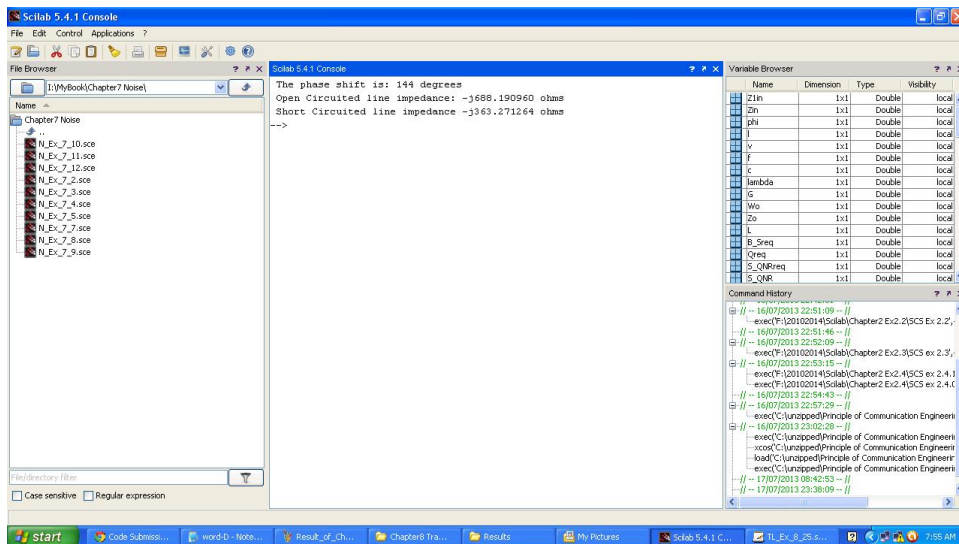


Figure 8.2: Transmission Line Ex 8 2

```

7 lambda=v/f
8 //disp(lambda)
9 l=100// length of the Tx-Line
10 phi=2*(%pi*l)/(lambda)// Phase shift in degrees
11 Zo=500// Characteristic impedance
12 //a Open circuited Line
13
14 Zin=-%i*Zo*(cos(phi)/sin(phi))
15
16 //b Short circuited Line
17 Z1in=%i*Zo*tan(phi)
18 mprintf('The phase shift is: %d degrees\n Open
    Circuited line impedance: -j%f ohms\n Short
    Circuited line impedance -j%f ohms ',phi*180/%pi,-
    Zin*%i,Z1in*%i)

```

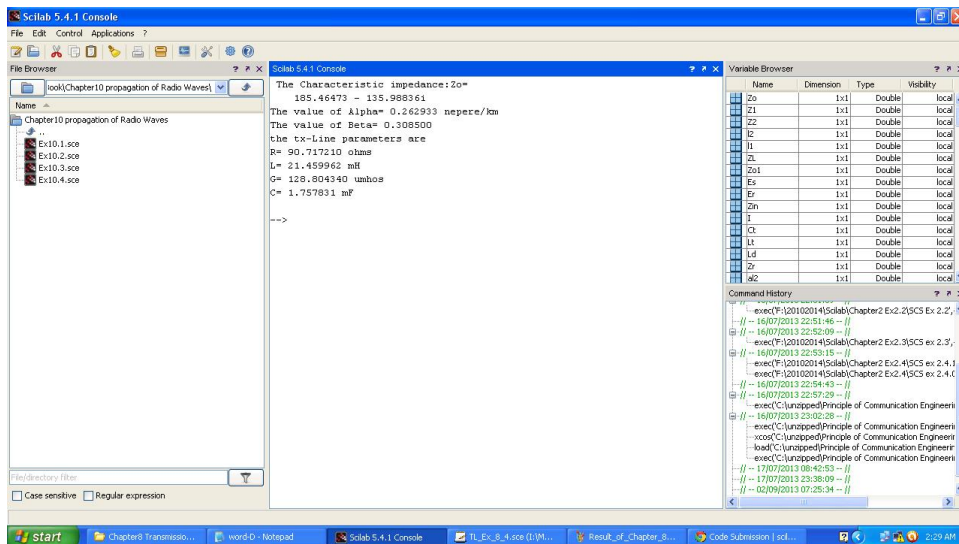


Figure 8.3: Transmission Line Ex 8 3

Scilab code Exa 8.3 Transmission Line Ex 8 3

```

1
2 clc
3 //Chapter8
4 //Example8.3, page no 315
5 //Given
6 f=1600
7 w=1000
8 Zoc=2460*exp(%i*-86.5*%pi/180)// Open circuited Line
   impedance
9 Zsc=21.5*exp(%i*14*%pi/180)// Short circuited Line
   impedance
10 Zo=sqrt(Zoc*Zsc)// Characteristic impedance
11 A=real(sqrt(Zsc/Zoc))// tan(a+ jBeta) = A + jB
12 B=imag(sqrt(Zsc/Zoc))
13 l=1/4
14 alpha=(1/(4*1))*log((1+A^2+B)^2)/((1-A)^2+B^2))
   //Attenuation Constant
15 Beta=(1/(2*1))*atan((2*B)/(1-A^2-B)) //phase

```

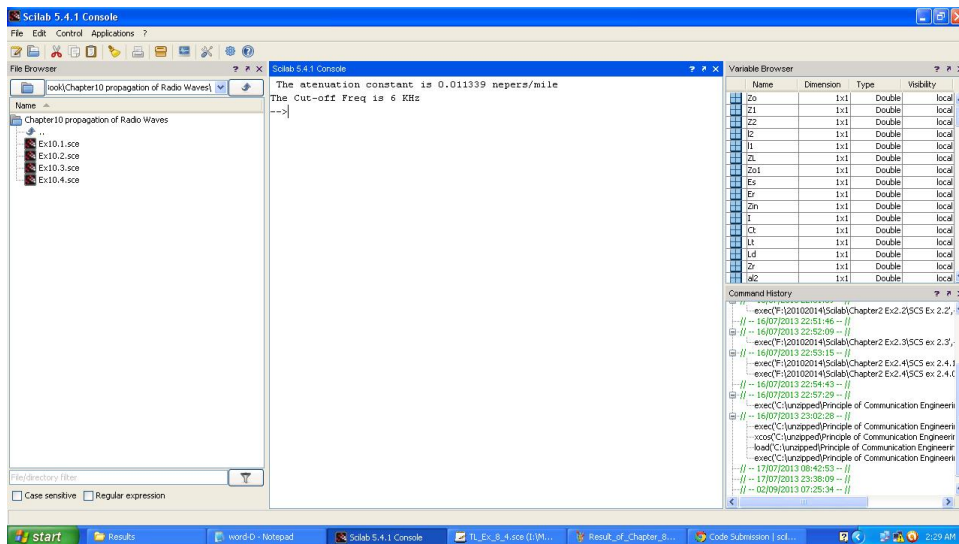



Figure 8.4: Transmission Line Ex 8 4

```

constant
16
17 //the tx-Line parameters
18 R=real(Zo*complex(alpha,Beta))
19 L=imag(Zo*complex(alpha,Beta))
20 G=real(complex(alpha,Beta)/Zo)
21 C=imag(complex(alpha,Beta)/Zo)
22 mprintf('The Characteristic impedance:Zo= '),disp(
    Zo)
23 mprintf('The value of Alpha= %f nepere/km\n',alpha)
24 mprintf('The value of Beta= %f \n',Beta)
25 mprintf('the tx-Line parameters are\nR= %f ohms\nL=
    %f mH\nG= %f umhos\nC= %f mF\n',R,L,G*1e6,C*1e3)
26
27 // Note : There are some calculation errors in the
    solution presented in the book

```

Scilab code Exa 8.4 Transmission Line Ex 8 4

```
1 clc
2 //Chapter8
3 //Example8.4, page no 316
4 //Given
5 d=0.7// distance between two insertions
6 Ld_m= (80e-3)*(10/7)//Loading coil inductance
7 //disp(Ld_m)
8 Rd_m=100/7//Loading coil resistance
9 //disp(Rd_m)
10 R=20+Rd_m//Line resistance
11 L=Ld_m// Line inductance
12 C=0.05e-6// Line Capacitance
13 alfa=0.5*R*sqrt(C/L)//Attenuation Constant
14 //
15 fc=(%pi*d*sqrt(L*C))^-1//cut off freq
16 mprintf('The atenuation constant is %f nepers/mile\
nThe Cut-off Freq is %d KHz',alfa,fc*1e-3)
17
18 // Note : There are some calculation errors in the
    solution presented in the book
```

Scilab code Exa 8.5 Transmission Line Ex 8 5

```
1 clc
2 //Chapter8
3 //Example8.5, page no 317
4 //Given
5 a=0.7//attenuation constant
```

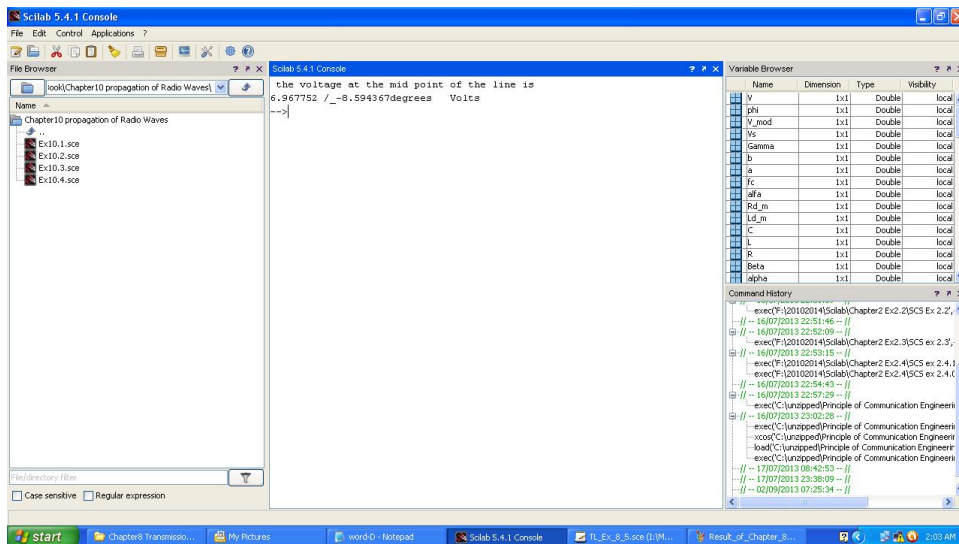


Figure 8.5: Transmission Line Ex 8 5

```

6 b=0.3 //phase constant
7 Gamma=a+(%i*b) //propagation constant
8 l=0.5 // half length of line( for midpoint)
9 Vs=10 // Excitation voltage
10 V_mod=Vs*(%e^(-a*l)) //Magnitude of the Vs
11
12 phi=b*l*180/%pi //phase shift
13 V=V_mod*(%e^(-%i*(phi*%pi/180))) //voltage at the mid
    point
14 mprintf('the voltage at the mid point of the line is
    \n%f /_-%fdegrees Volts ',V,phi)
15
16 // Note : There are some calculation errors in the
    solution presented in the book

```

Scilab code Exa 8.6 Transmission Line Ex 8 6

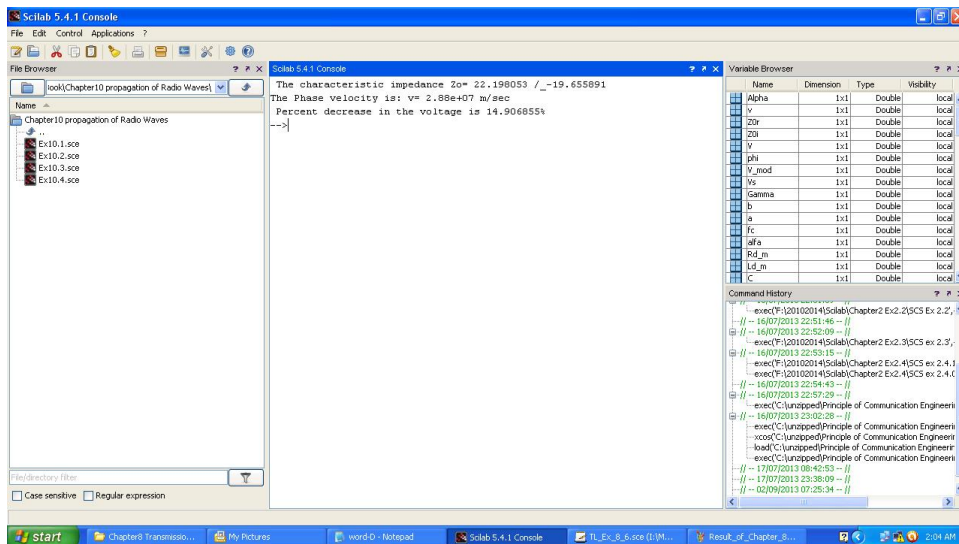


Figure 8.6: Transmission Line Ex 8 6

```

1  clc
2  //Chapter8
3  //Example8.6 , page no 317
4  //GivenR=0.01
5  R=0.01 , l=1e3
6  L=1e-6
7  G=1e-6
8  C=0.001e-6
9  f=1.59e3 // operating freq
10 w=2*%pi*f // angular freq
11 //a
12 Zo=sqrt((R+(%i*w*L))*0.35/(G+(%i*w*C))) //
    characteristic impedance
13 [Z0r,Z0i]=polar(Zo)
14 //b
15
16 Beta=sqrt(0.5*(sqrt(((R^2)+(round(w^2)*(L^2)))*
    round(G^2)+(round(w^2)*(C^2))))-(round(R*G)-((w
    ^2)*L*C)))) //Phase constant
17

```

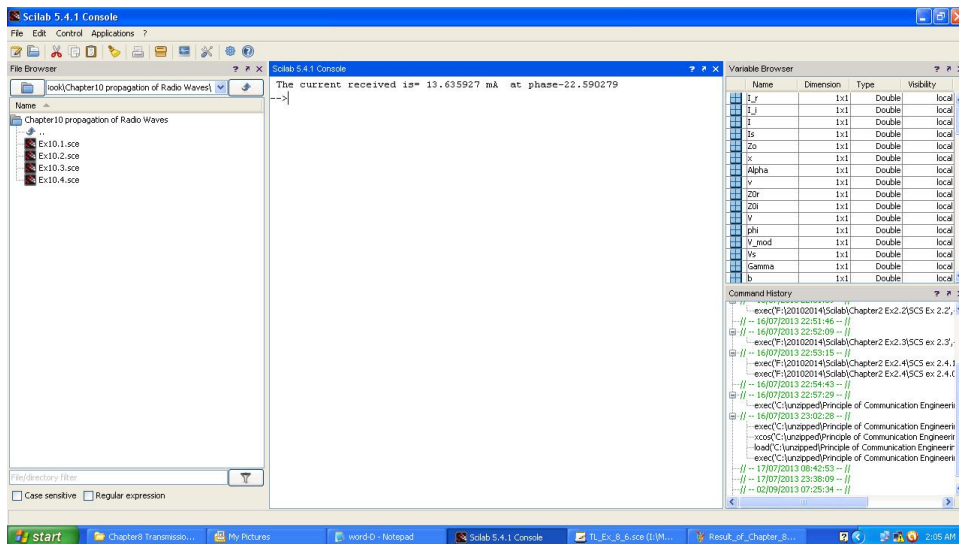


Figure 8.7: Transmission Line Ex 8 15 Pg334

```

18 v=w/Beta//phase velocity
19
20 //c
21 Alpha=sqrt(0.5*(sqrt(((R^2)+((w^2)*(L^2)))*((G^2)
    +((w^2)*(C^2)))))+(R*G)-((w^2)*L*C)))//
    attenuation constant
22 Vs=1//Assumed for easeof calculation
23 A=(Vs-(Vs*exp(-Alpha*l)))*100
24 mprintf('The characteristic impedance Zo= %f /-%f \n
    ',Z0r,Z0i*180/%pi)
25 mprintf('The Phase velocity is: v= %3.2e m/sec\n
    Percent decrease in the voltage is %f%c',v,A,'%')
26
27 // Note : There are some calculation errors in the
    solution presented in the book

```

Scilab code Exa 8.15 Transmission Line Ex 8 15 Pg334

```
1 clc
2 //Chapter8
3 //Example8.15, page no 334
4 //GivenR=0.01
5 x=10//line length
6 Zo=100// characteristic impedance
7 a=0.1// attenuation constant
8 Beta=0.05// phase constant
9 Is=20e-3// source current
10 Gamma=a+ %i*Beta// propagation constant
11
12 I=Is/cosh(Gamma*x)// received current
13
14 [I_r, I_i]=polar(I)
15
16 mprintf('The current received is= %f mA at phase%f'
,1000*I_r, I_i*180/%pi)
```

Scilab code Exa 8.15.1 Transmission Line Ex 8 15 Pg348

```
1 clc
2 //Chapter8
3 //Example8.15, page no 348
4 //Given
5 l=100// Tx-line length
6 ZR=200//Terminal resistance
7 Zo=600// Characteristic impedance
8 a=0.01//attenuation constant
9 Beta=0.03//phase constant
10 d=0//reflection coeff at load is Zero
11 Gamma=a+%i*Beta//propagation constant
```

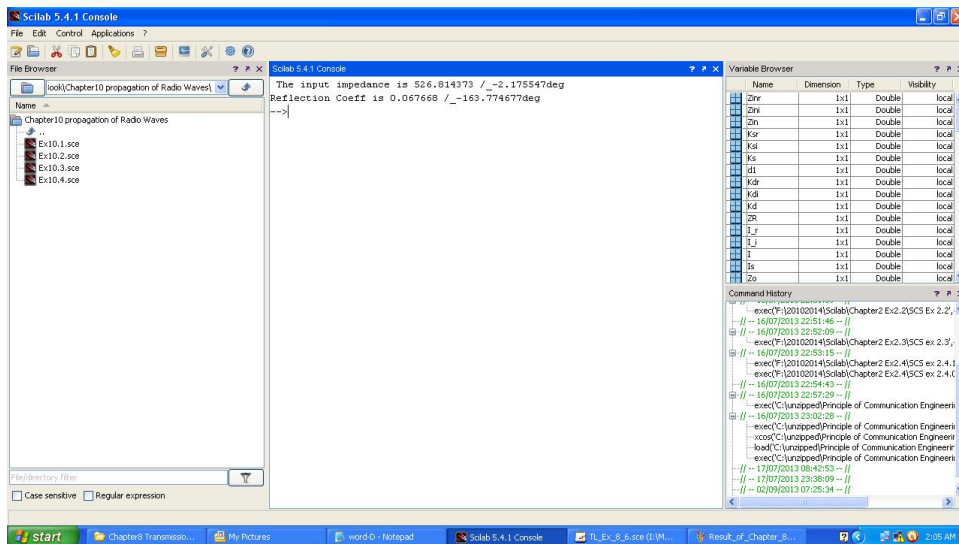


Figure 8.8: Transmission Line Ex 8 15 Pg348

```

12 Kd=((ZR-Zo)/(ZR+Zo))*%e^(-2*Gamma*d) // reflection
    coeff at point D d km from load
13 [Kdr,Kdi]=polar(Kd)
14 d1=100 // distance
15 Ks=((ZR-Zo)/(ZR+Zo))*%e^(-2*Gamma*d1) // reflection
    coeff at the sending end
16 [Ksr,Ksi]=polar(Ks)
17 Zin=Zo*((((ZR*cosh(Gamma*l))+(Zo*sinh(Gamma*l)))/((Zo
    *cosh(Gamma*l))+(ZR*sinh(Gamma*l)))) // Input
    impedance
18 [Zinr,Zini]=polar(Zin)
19 mprintf('The input impedance is %f /-%fdeg\
    nReflection Coeff is %f /-%fdeg',Zinr,Zini*180/
    %pi,Ksr,Ksi*180/%pi)
20
21 // Note : There are some calculation errors in the
    solution presented in the book

```

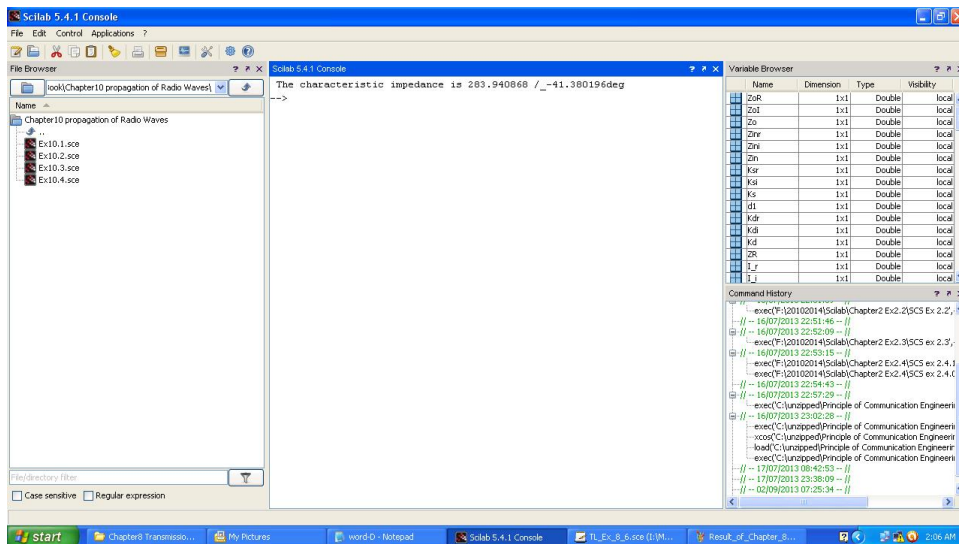


Figure 8.9: Transmission Line Ex 8 16

Scilab code Exa 8.16 Transmission Line Ex 8 16

```

1  clc
2  //Chapter8
3  //Example8.16 , page no 349
4  //Given
5  L=1e-3//inductance
6  R=40// Resistance
7  C=0.1e-6// capacitance
8  G=1e-6//conductance
9  w=5000// angular freq
10 Zo=sqrt(complex(R,(w*L))/complex(G,(w*C)))//
    Characteristic impedance
11 //Zr=sqrt(sqrt(R^2+(w*L)^2)/sqrt(G^2+(w*C)^2))
12 [ZoR,ZoI]=polar(Zo)
13 mprintf('The characteristic impedance is %f /_%fdeg'
    ,ZoR,ZoI*180/%pi)

```

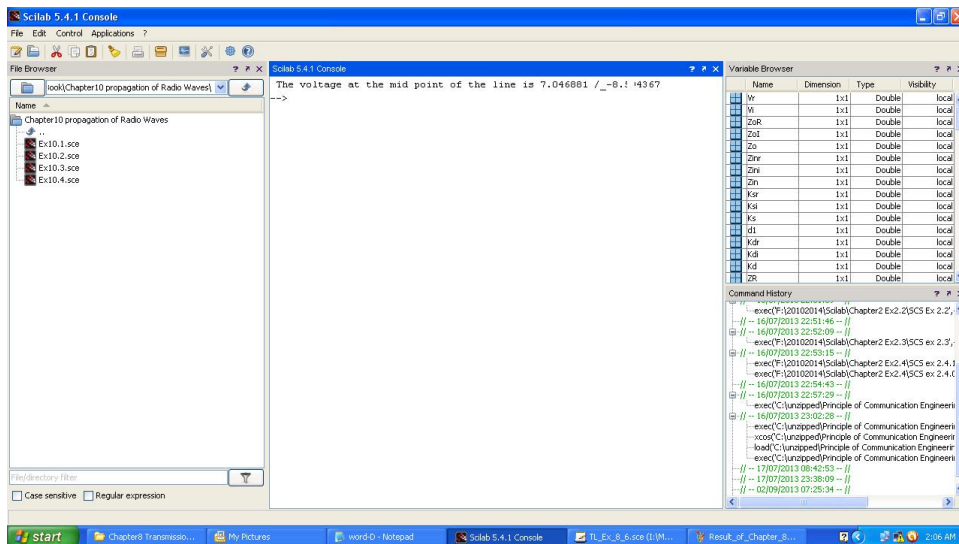



Figure 8.10: Transmission Line Ex 8 17

14

15 // Note : There are some calculation errors in the solution presented in the book

Scilab code Exa 8.17 Transmission Line Ex 8 17

```

1  clc
2  //Chapter8
3  //Example8.17, page no 349
4  //Given
5  l=0.5//half line distance
6  Vs=10//Excitation voltage
7  Gamma=0.7+%i*0.3//propagation constant
8  [Vr,Vi]=polar(Vs*(%e^(-Gamma*l))//vtg at mid point
9  mprintf('The voltage at the mid point of the line is
           %f / -%f ',Vr,Vi*180/%pi)

```

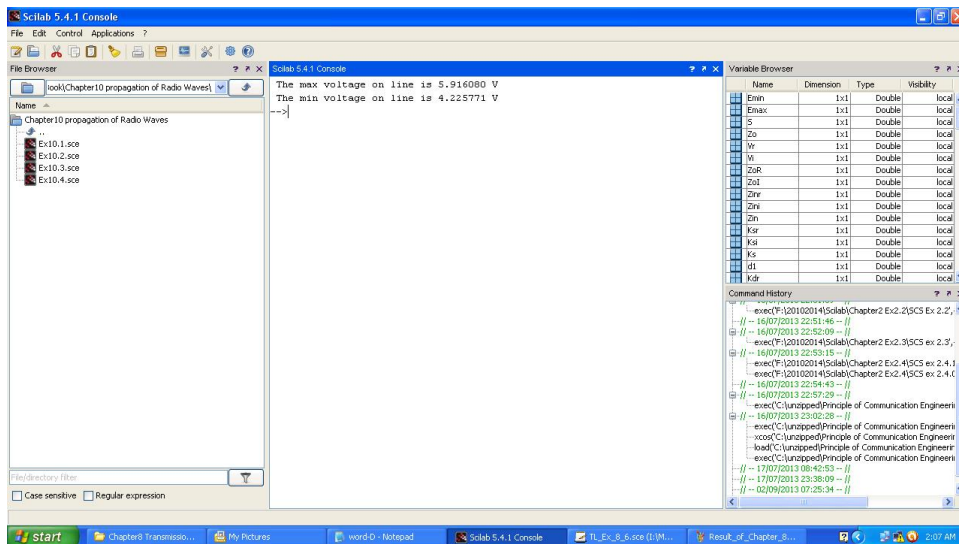


Figure 8.11: Transmission Line Ex 8 18

Scilab code Exa 8.18 Transmission Line Ex 8 18

```

1  clc
2  //Chapter8
3  //Example8.18 , page no350
4  //Given
5  Zo=50// characteristic impedance
6  P=500e-3//Supplied power
7  S=1.4//VSWR on the line
8  Emax=sqrt(Zo*S*P)//Max vtg
9
10 Emin=sqrt(Zo*P/S)// Min vtg
11 mprintf('The max voltage on line is %f V\n The min
           voltage on line is %f V',Emax,Emin)

```

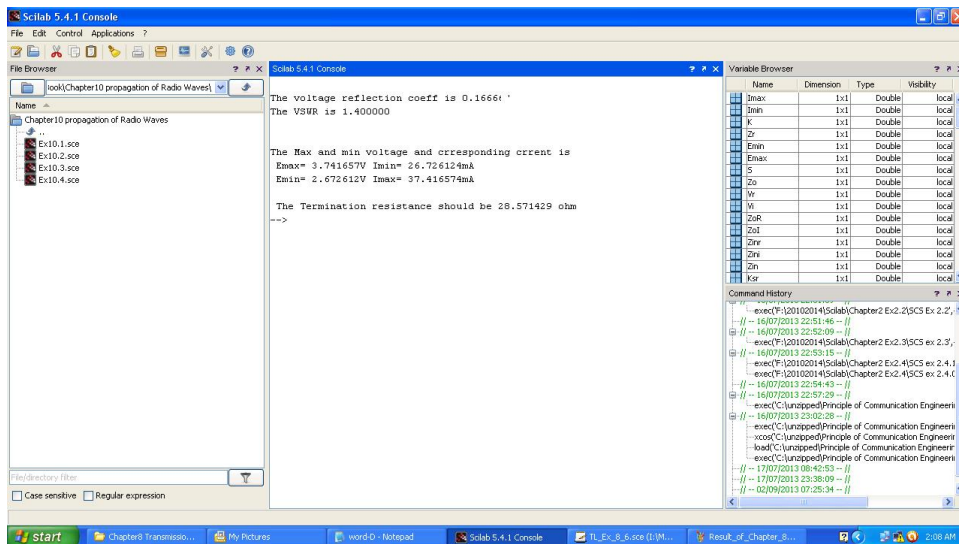


Figure 8.12: Transmission Line Ex 8 19

Scilab code Exa 8.19 Transmission Line Ex 8 19

```

1  clc
2  //Chapter8
3  //Example8.19, page no 350
4  //Given
5  Zo=100// Characteristic Impedance
6  P=100e-3//Load power
7  Zr=140//Load Resistance
8  f=100e3// Operating freq
9  //a
10 K=(Zr-Zo)/(Zo+Zr)//Vtg Reflection coeff
11
12 //b
13 S=(1+K)/(1-K)//VSWR

```

```

14
15 //c+d
16 Emax=sqrt(Zr*P)//Max line vltg
17 Imin=Emax/Zr//Min line current
18
19 Emin=Emax/S// Min line vltg
20 Imax=S*Imin//Max line current
21
22 //e
23 R=14000/40
24
25 Zr=(Zo^2)/R//
26 mprintf('\nThe voltage reflection coeff is %f\nThe
    VSWR is %f\n\nThe Max and min voltage and
    crresponding crrent is\n Emax= %fV Imin= %fmA\n
    Emin= %fV Imax= %fmA\n\n The Termination
    resistance should be %f ohm',K,S,Emax,Imin*1e3,
    Emin,Imax*1e3,Zr)

```

Scilab code Exa 8.20 Transmission Line Ex 8 20

```

1 clc
2 //Chapter8
3 //Example8.20, page no 352
4 //Given
5 V=0.5//receiving vtg
6 Vs=2//Source vtg
7 a1=-log(V/Vs)//attenuation
8
9 a12=a1*1.5
10 V=Vs*e^-a12//receiving voltage
11 mprintf('the receiving voltage will be %f V',V)

```

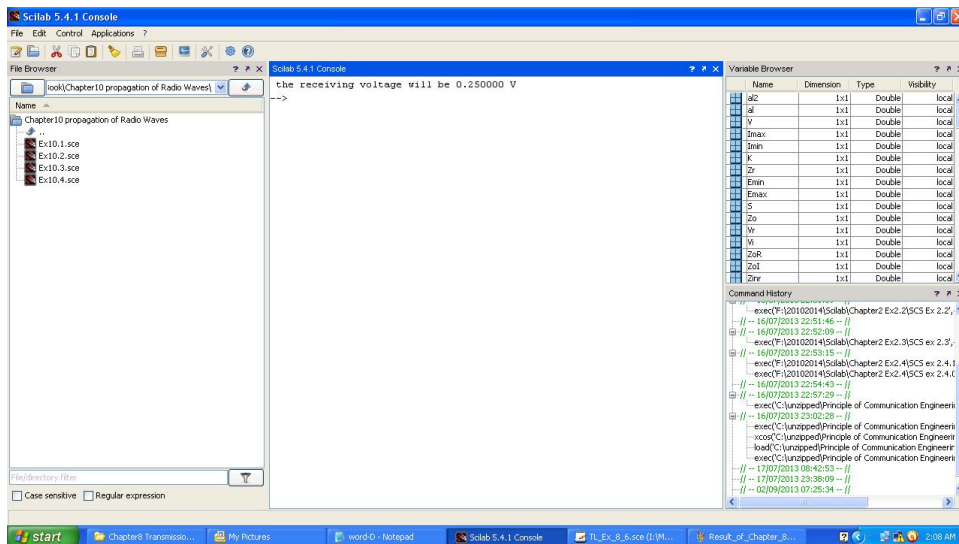


Figure 8.13: Transmission Line Ex 8 20

Scilab code Exa 8.22 Transmission Line Ex 8 22

```

1  clc
2  //Chapter8
3  //Example8.22 , page no352
4  //Given
5  Zin=25+%i*15// Internal Impedance
6  Zr=70-%i*42//load
7  f=3e6//operating freq
8  v=3e8//light velocity
9  L=v/(4*f)//length of the line
10
11 Zo=sqrt(Zin*Zr)//Characteristic Impedance
12 mprintf('The lrngth should be %d metres\nThe
    Characteristic Impedance should be %f ohms ',L,Zo)

```

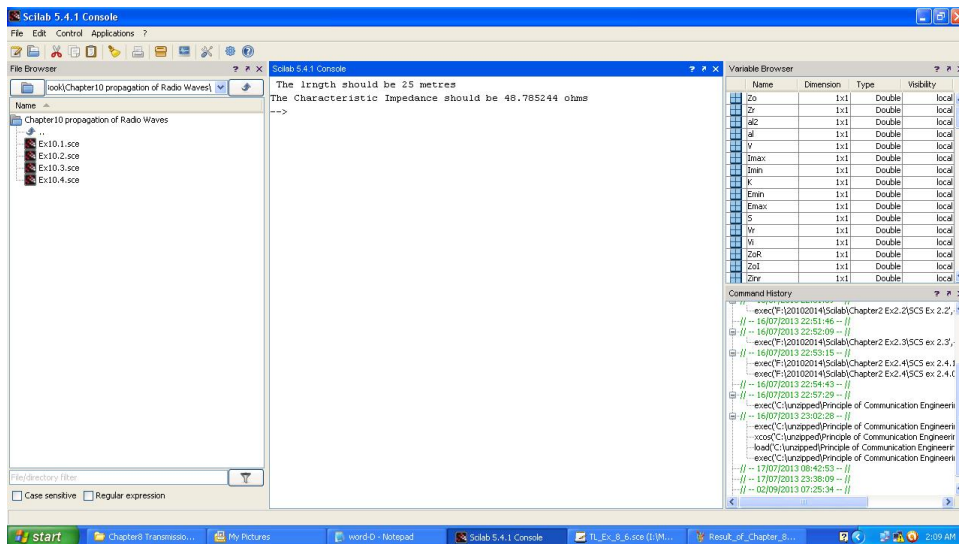


Figure 8.14: Transmission Line Ex 8 22

Scilab code Exa 8.23 Transmission Line Ex 8 23

```

1  clc
2  //Chapter8
3  //Example8.23 , page no353
4  //Given
5  //a
6  L=1e-3// inductance
7  C=61.25e-9// capacitance
8  Ld=44e-3//coil inductance
9  d=2//distance intervals after which coils are added
10 Lt=(L*2)+(Ld*2)//total inductance
11 Ct=C*2//total capacitance
12 fc=(%pi*sqrt(Lt*Ct))^-1//cut off freq
13

```

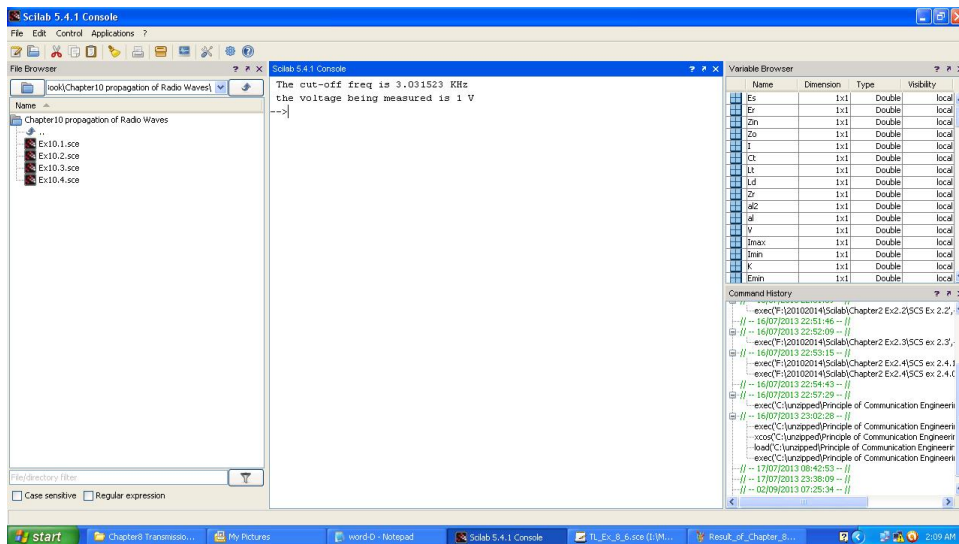


Figure 8.15: Transmission Line Ex 8 23

```

14 //b
15 I=100e-3//milliammeter range
16 R=1//milliammeter resistance
17 Zo=100//characteristic impedance
18 Zin=(Zo^2)/R//input impedance
19
20 Er=I*R//
21 Es=Er*sqrt(Zin/Zo)
22 mprintf('The cut-off freq is %f KHz \n the voltage
    being measured is %d V',fc*1e-3,Es)

```

Scilab code Exa 8.24 Transmission Line Ex 8 24

```

1 clc
2 //Chapter8
3 //Example8.24 , page no 354

```

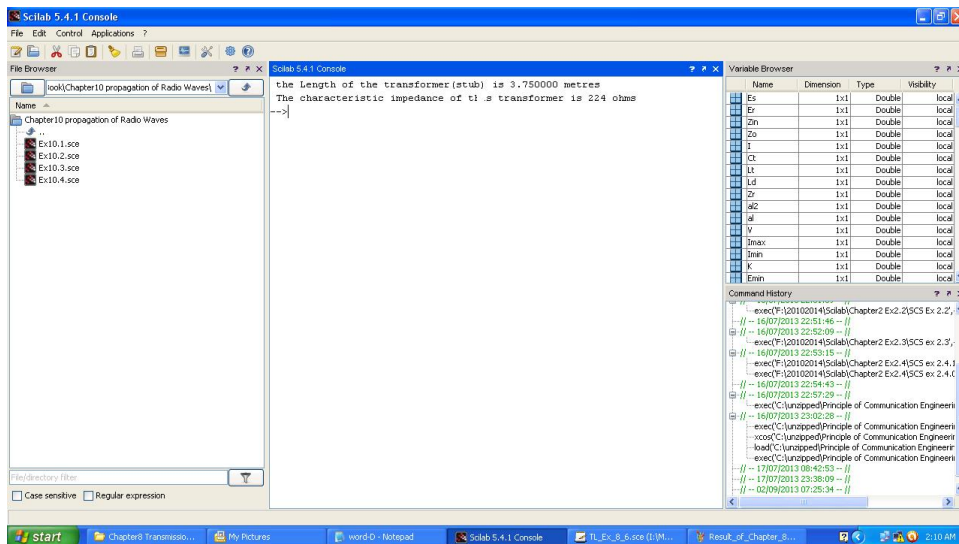


Figure 8.16: Transmission Line Ex 8 24

```

4 //Given
5 f=20e6//tuned freq
6 ZR=100//Equivalent aerial Resistance
7 Zin=500//input impedance
8 c=3e8
9 lambda=c/f
10 l=lambda/4//lambda/4 Transformer
11
12 Zo=sqrt(Zin*ZR)//Characteristic impedance
13 mprintf('the Length of the transformer(stub) is %f
    metres\n The characteristic impedance of this
    transformer is %d ohms ',1,round(Zo))

```

Scilab code Exa 8.25 Transmission Line Ex 8 25

```
1 clc
```

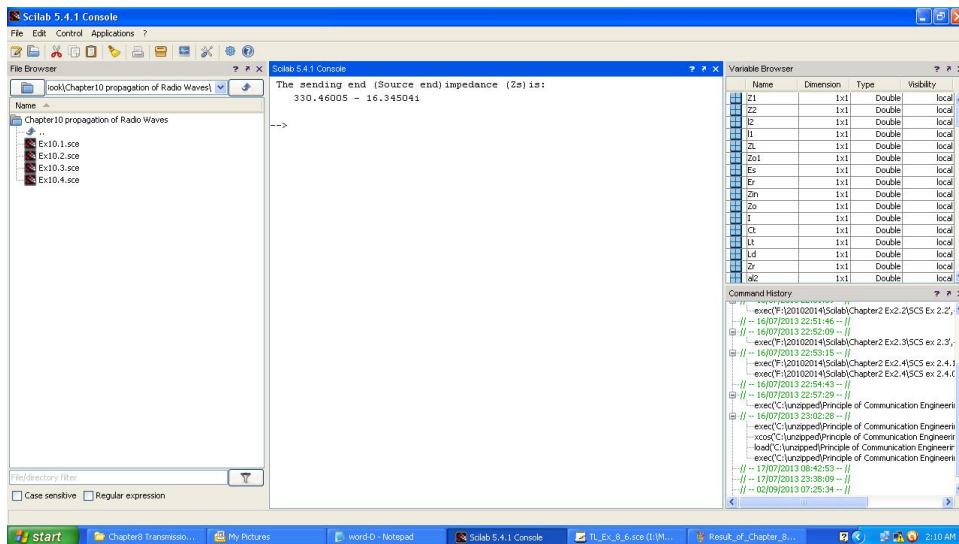



Figure 8.17: Transmission Line Ex 8 25

```

2 //Chapter8
3 //Example8.25 , page no 354
4 //Given
5 lambda=5//wavelength
6 Zo=200//Characteristic impedance
7 Zo1=100//Zo'
8 ZL=50+(%i*50)// load impedance
9 l1=0.4*lambda
10 l2=0.2*lambda
11 Beta=(2*%pi/lambda)// phase difference
12 Z2=Zo1*(((ZL*cos(Beta*l2))+(%i*Zo1*sin(Beta*l2)))/((
    Zo1*cos(Beta*l2))+(%i*ZL*sin(Beta*l2))))//I/p
    Impedance offered by I2toI1
13 Z1=Zo*(((Z2*cos(Beta*l1))+(%i*Zo*sin(Beta*l1)))/((Zo
    *cos(Beta*l1))+(%i*Z2*sin(Beta*l1))))//I/p
    impedance
14 mprintf('The sending end (Source end)impedance (Z1)
    is: '),disp(Z1)

```

Chapter 9

Aerials

Scilab code Exa 9.1 Aerials Ex 9 1

```
1 clc
2 //Chapter9
3 //Example9.1, page no 397
4 //Given
5 D=90// directivity
6 lambda=2// wavelength
7 Ae=(D*(lambda^2))/(4*%pi)//effective aperture
8 mprintf('The maximum effective aperture of the\n
   aerial is %f sq m',Ae)
```

Scilab code Exa 9.2 Aerials Ex 9 2

```
1 clc
2 //Chapter9
3 //Example9.2, page no 397
4 //Given
5 n=10//no of aerial elements
```

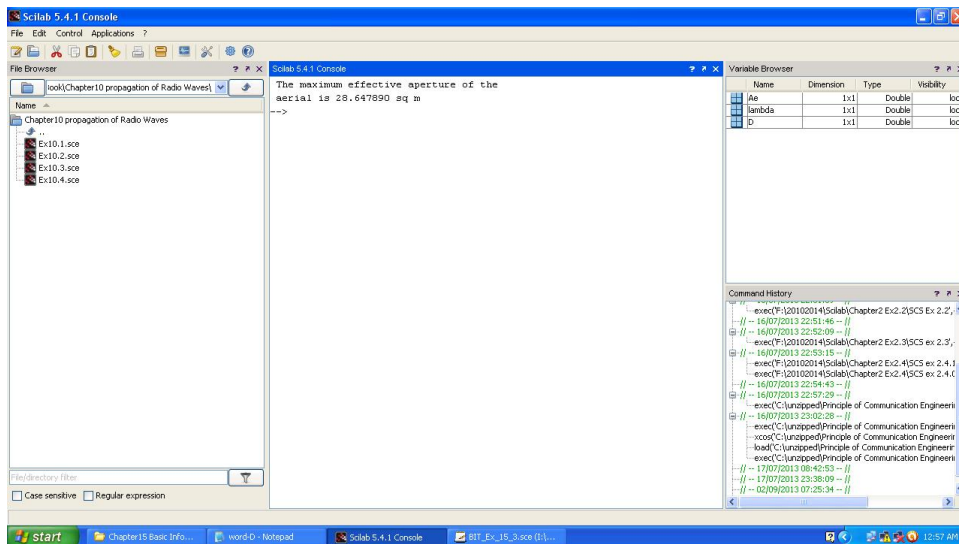


Figure 9.1: Aerials Ex 9 1

```

6 d=0.5//distance in terms of wavelength
7 Beam_Width=2/(n*d)//
8 Beam_Width_degrees=Beam_Width*180/%pi
9 mprintf('Angular beam width is %f degrees\nBeamWidth
    is %f rad',Beam_Width_degrees,Beam_Width)

```

Scilab code Exa 9.3 Aerials Ex 9 3

```

1 clc
2 //Chapter9
3 //Example9.3 , pageno 397
4 //Given
5 r=1//assume distance for ease of calculation
6 //Pav(theta)=(1000/(3*%pi*r^2))*((sin(theta))^2)
7 theta=0:0.1:%pi
8 x0=0, x1=%pi

```

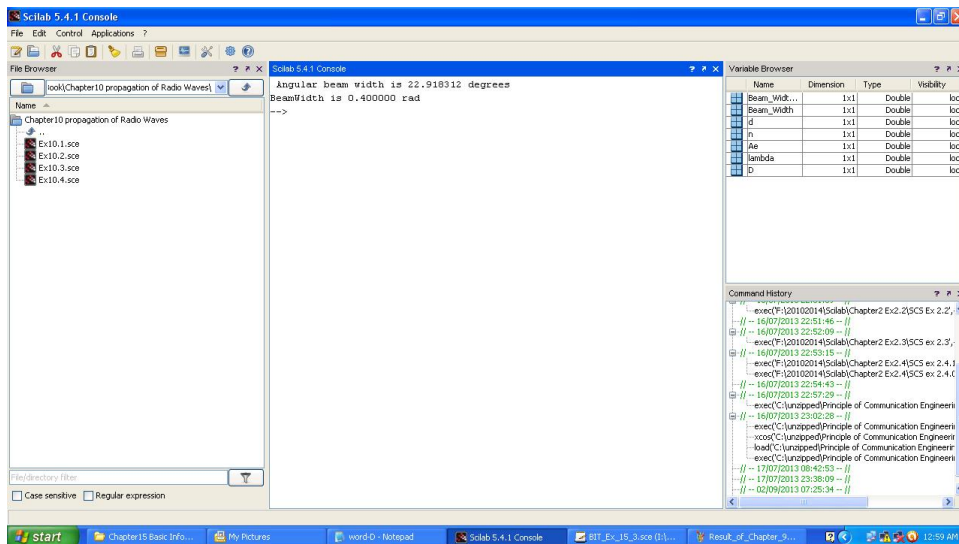


Figure 9.2: Aerials Ex 9 2

```

9 Pt=(2000/(3*r^2))*integrate(' (sin(theta))^3 ', 'theta'
    ,x0,x1)//Total power radiated
10 mprintf('Total power radiated is %f watts',Pt)

```

Scilab code Exa 9.4 Aerials Ex 9 4

```

1 clc
2 //Chapter9
3 //Example9.4, page no 398
4 //Given
5 d1=2// length of wire
6 I=6//current in the wire
7 f=1e6// operating freq
8 r=30e3//distance at which field is to be calculated
9 theta=90//right angles to the wire axis
10 lambda=300// wavelength

```

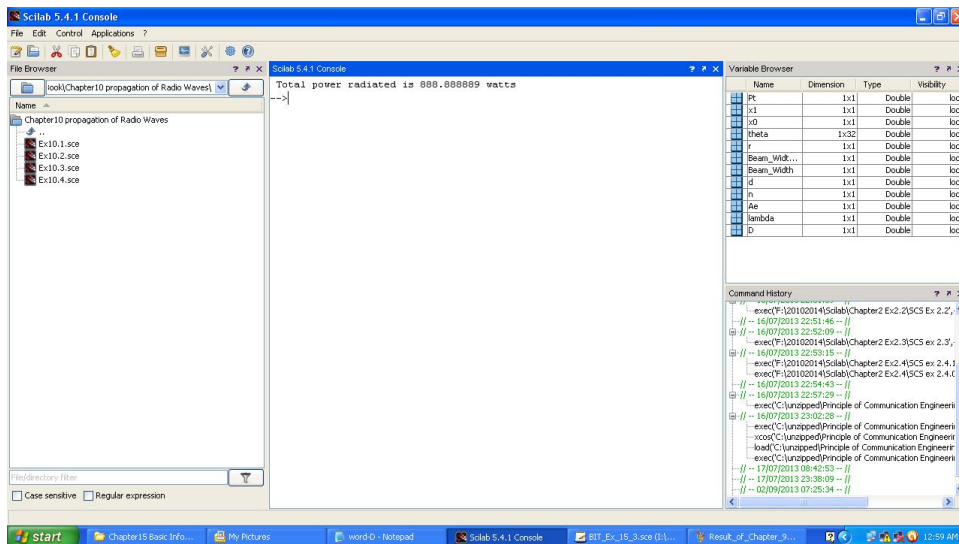


Figure 9.3: Aerials Ex 9 3

```

11 w=2*%pi*f//angular freq
12 c=3e8,t=f^-1
13 Phi=w*(t-(r/c))//Phase shift
14 Erad=25.13e-5*cos(Phi)//Radiation electric field
    intensity
15 H=Erad/(120*%pi)//Radiation magnetic field intensity
16 mprintf('electric field intensity is %f mV/m \n
    magnetic field intensity is %f uA/m',Erad*1e3,H*1
    e6)

```

Scilab code Exa 9.5 Aerials Ex 9 5

```

1 clc
2 //Chapter9
3 //Example9.5
4 //Given

```

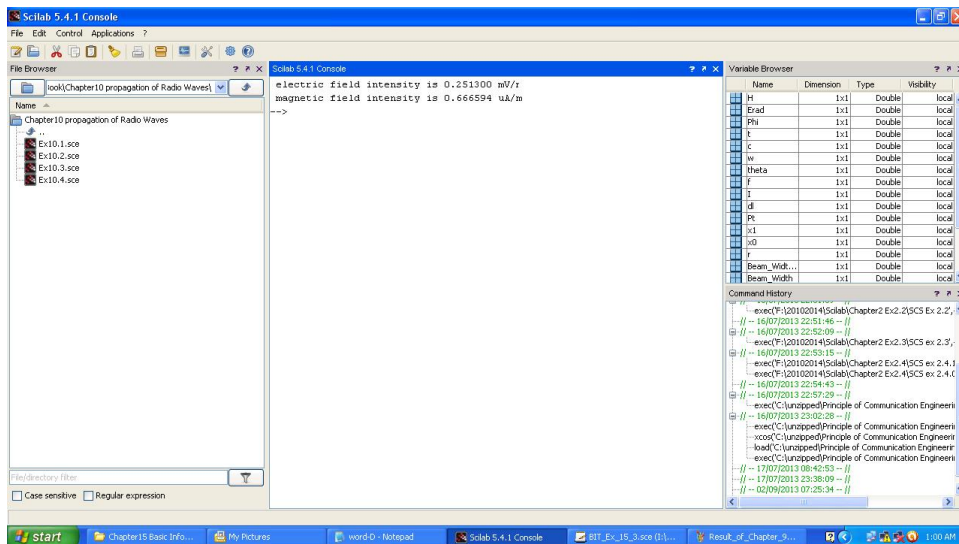


Figure 9.4: Aerials Ex 9 4

```

5 //c
6 Rr=73// radition resistance
7 Vrms=10//RMS voltage of the signal
8 Zin_mod=sqrt((73^2)+(42^2))//absolute input
   impedance
9 Irms=Vrms/Zin_mod//RMS current
10 Pt=(Irms^2)*Rr// Radiated power
11 mprintf('The radiated power is %f watts',round(100*
   Pt)/100)

```

Scilab code Exa 9.6 Aerials Ex 9 6

```

1 clc
2 //Chapter9
3 //Example9.6
4 //Given

```

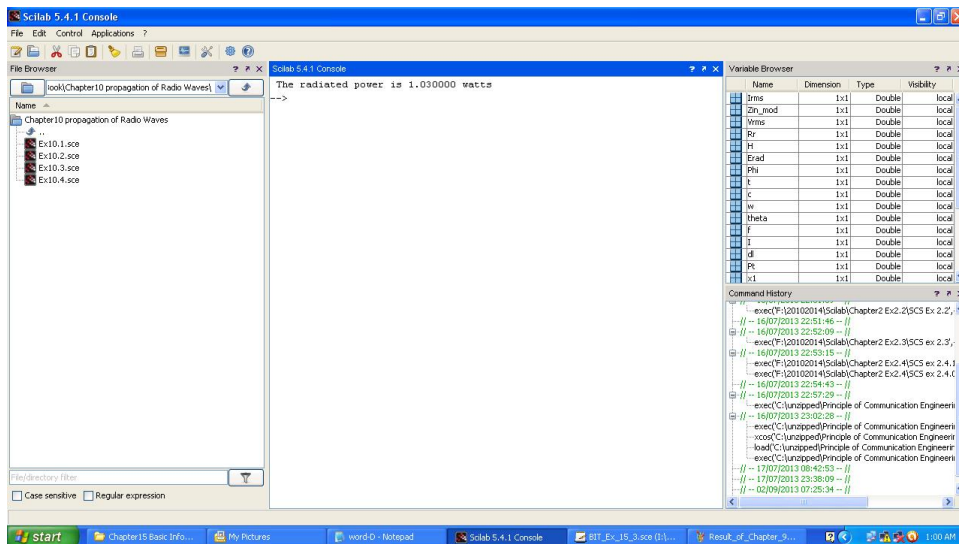


Figure 9.5: Aerials Ex 9 5

```

5 //b
6 c=3e8
7 f=2e9//operating freq
8 Ae=100//aperture area
9 lambda=c/f// operating wavlength
10 D=((4*3.141*Ae)/(lambda^2))// Directivity
11 mprintf('Ideal directive gain is %d',D)

```

Scilab code Exa 9.7 Aerials Ex 9 7

```

1 clc
2 //Chapter9
3 //Example9.7, pageno 400
4 //Given
5 //b
6 n=10// no of aerial elements

```

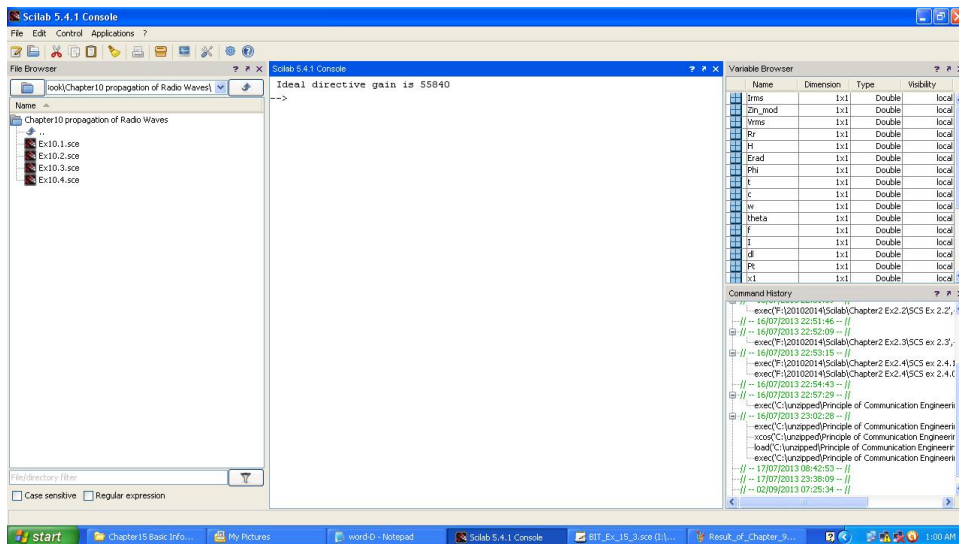


Figure 9.6: Aerials Ex 9 6

```

7 lambda_d=2//
8 BeamWidth=2*lambda_d/n// Beamwidth
9 mprintf('The angular width is %f degrees',BeamWidth)

```

Scilab code Exa 9.8 Aerials Ex 9 8

```

1 clc
2 //Chapter9
3 //Example9.8, page no 400
4 //Given
5 D1=1,D2=1.5*D1 // diameters of the new reflectors
   D1=l assumed for ease of calculation
6 G_dbs=10*log10((D2/D1)^2)//Gain in dBs
7 mprintf('Overall Gain is %f dBs',round(1000*G_dbs)
   /1000)

```

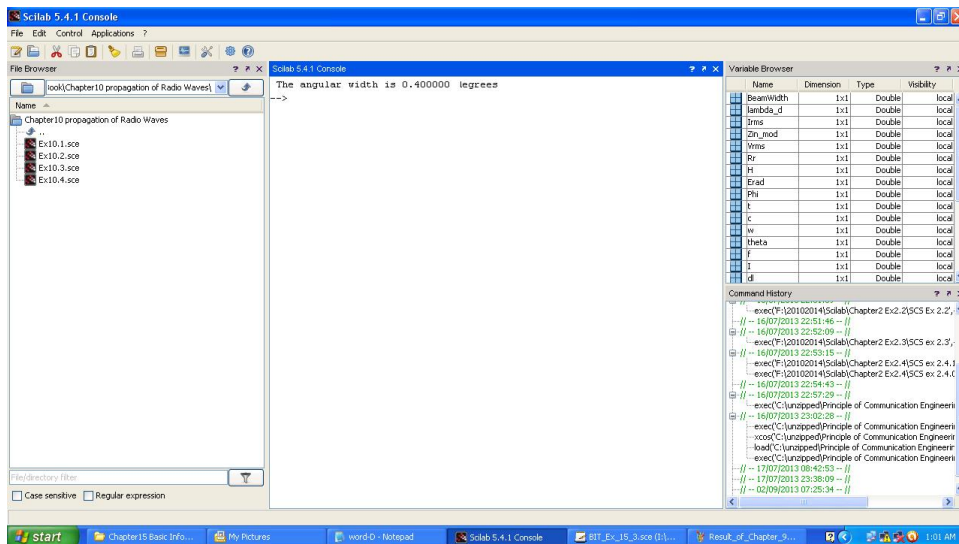


Figure 9.7: Aerials Ex 9 7

Scilab code Exa 9.9 Aerials Ex 9 9

```

1  clc
2  //Chapter9
3  //Example9.9
4  //Given
5  //b
6  c=3e8
7  f=800e3// operating freq
8  d1=27//effective height
9  lambda=c/f
10
11 Rr=40*(3.142^2)*(d1/lambda)^2//Radiation Resistance
12 mprintf('Radiation resistance is %f ohm',Rr)

```

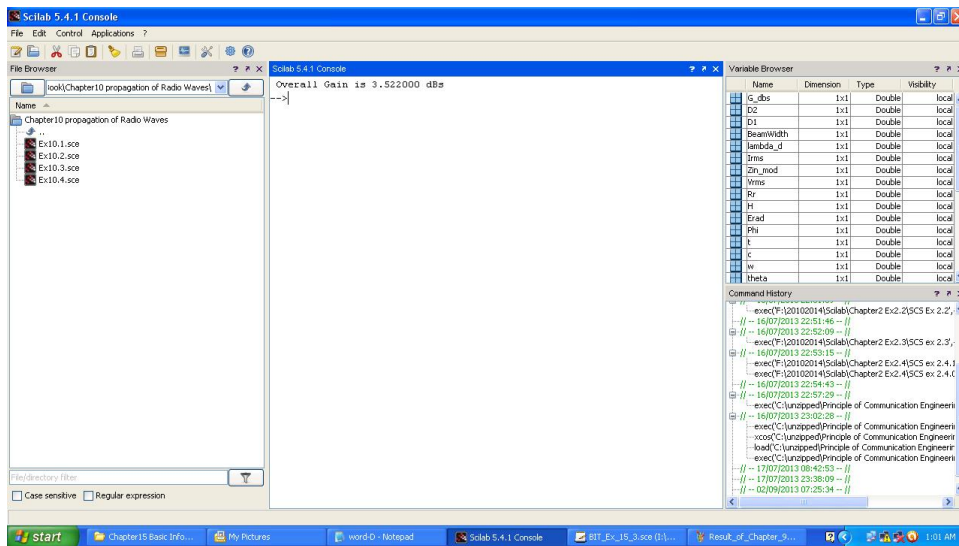


Figure 9.8: Aerials Ex 9 8

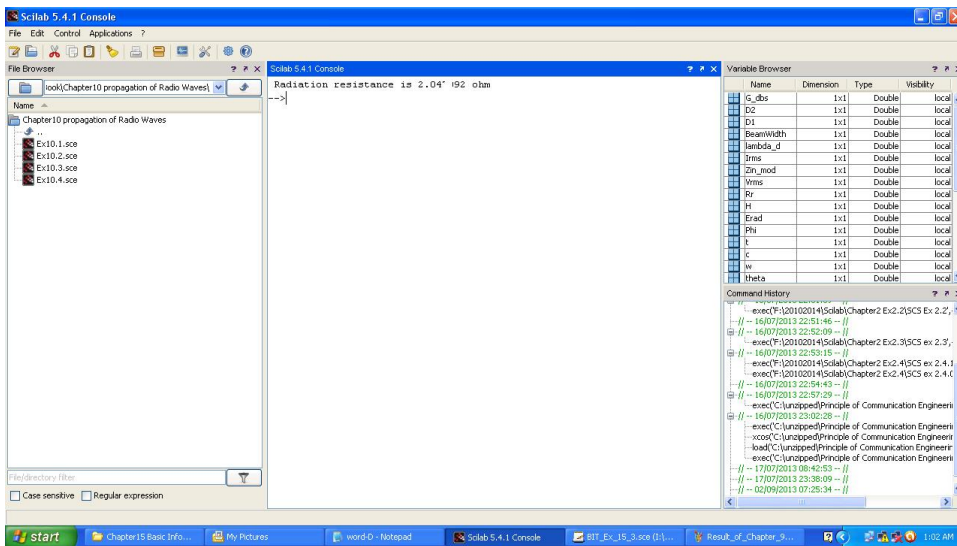


Figure 9.9: Aerials Ex 9 9

Chapter 10

Propagation of Radio Waves

Scilab code Exa 10.1 PoRW Ex 10 1

```
1 clc
2 //Chapter10
3 //Example10.1
4 //Given
5 Pt1=100//Radiated power
6 Pt2=30// Reduced Power
7 r=1//assume distance to be unity for easeof
  calculation
8 E1=300*sqrt(100)/r
9 E2=300*sqrt(30)/r
10 E=20*log10((E2/E1))// Reduction in field strength in
  dBs
11 mprintf('Field strength will reduce by %f dBs',-E)
```

Scilab code Exa 10.2 PoRW Ex 10 2

```
1 clc
```

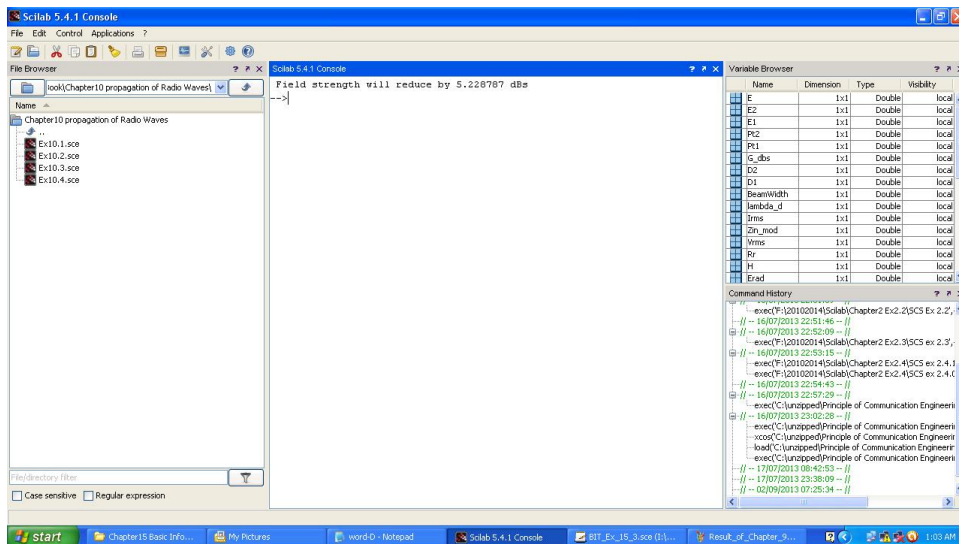


Figure 10.1: PoRW Ex 10 1

```

2 //Chapter10
3 //Example10.2
4 //Given
5 P=3//Transmitter power
6 ht=100// Antenna height
7 G=5//Antenna gain
8 d=20e3//distance
9 lambda=1,hr=1//assumed
10 E=((88*G*ht*hr*P^0.5)/(lambda*d^2))//field strength
11 mprintf('The field strength at distance 20km is %f
           uV/m',E*1e6)

```

Scilab code Exa 10.3 PoRW Ex 10 3

```

1 clc
2 //Chapter10

```

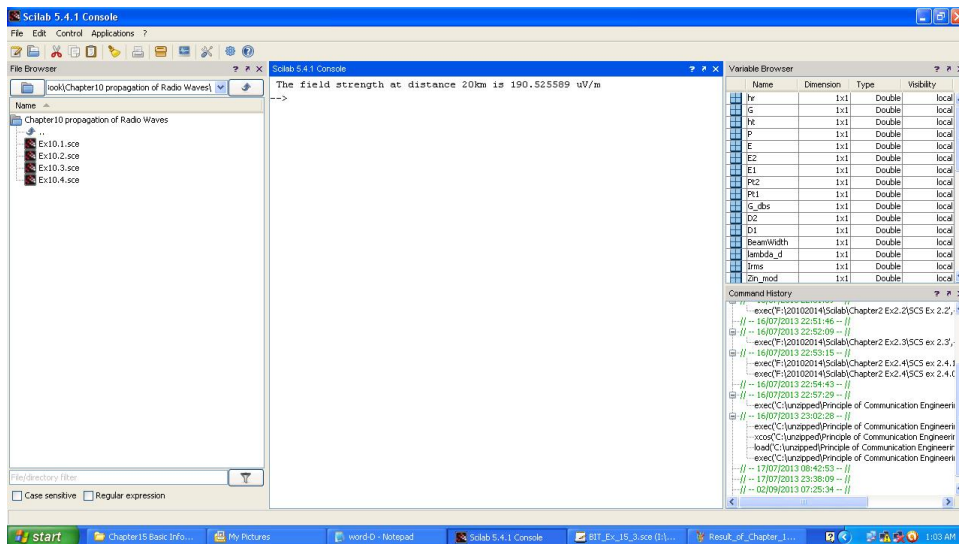


Figure 10.2: PoRW Ex 10 2

```

3 //Example10.3
4 //Given
5 ht=152.5,hr=9.15 // Antenna height
6 d=4100*(sqrt(ht)+sqrt(hr)) //distance
7 mprintf('Direct ray coverage is possible over %f km'
          ,d*1e-3)

```

Scilab code Exa 10.4 PoRW Ex 10 4

```

1 clc
2 //Chapter10
3 //Example10.4
4 //Given
5 //b
6 ht=3e3,hr=5e3 // Antenna height
7 d=4100*(sqrt(ht)+sqrt(hr)) //distance

```

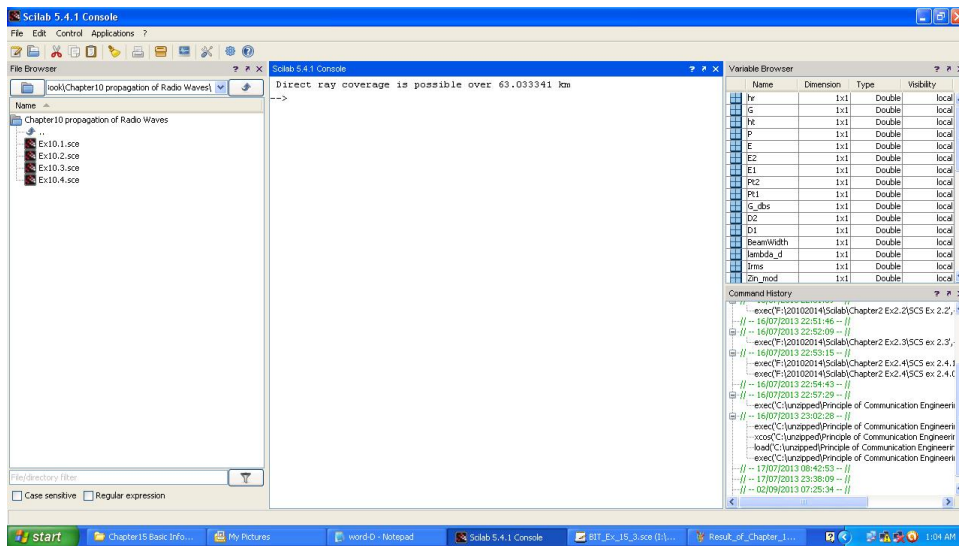


Figure 10.3: PoRW Ex 10 3

8 `mprintf('Max possible distance for efective point to point\n communication is %f km',d*1e-3)`

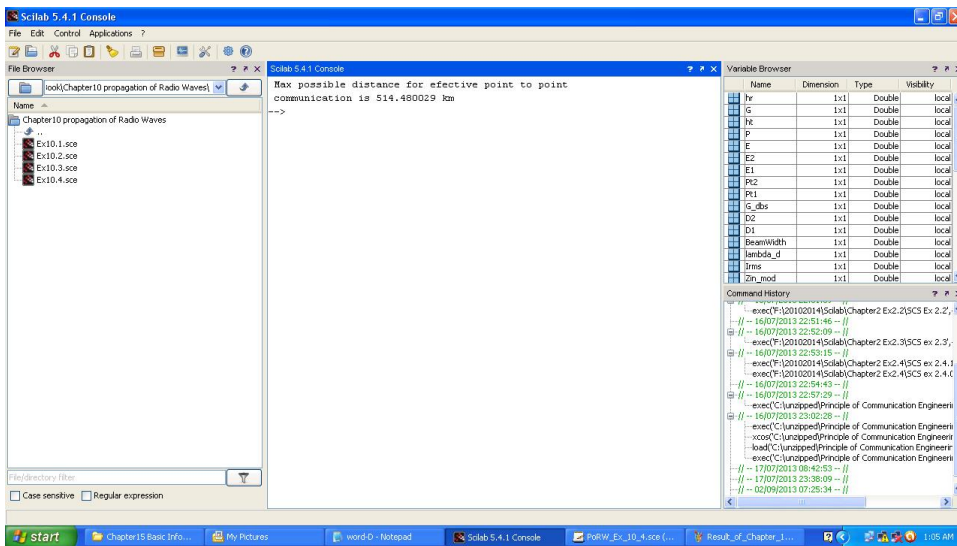


Figure 10.4: PoRW Ex 10 4

Chapter 11

BroadBand Communication

Scilab code Exa 11.1 BBC Ex 11 1

```
1 clc
2 //Chapter11
3 //Example11.1, page no 435
4 //Given
5 //a
6 c=20// no of signal channels
7 s=8e3// Channel sampling rate
8 t=1/s// time interval over which ll channels are
   sampled once
9 //b
10 g=5e-6// guded time for each channel sample
11 s_duration=t-g// duration of each sample
12 //c
13 samples_sec=c*s//
14 mprintf('The total no of samples per second is:\n %d
   samples/second',samples_sec)
```

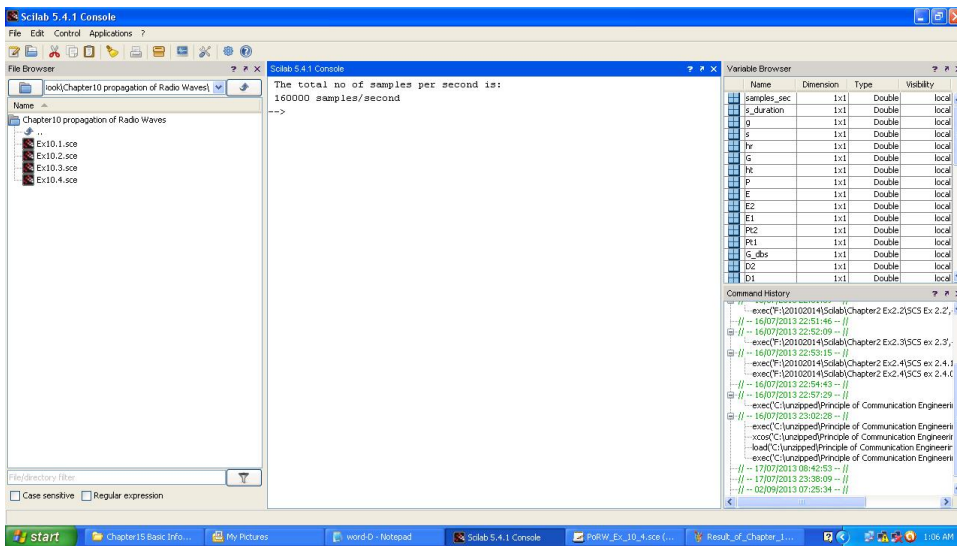


Figure 11.1: BBC Ex 11 1

Chapter 15

Basic Information Theory

Scilab code Exa 15.1 BIT Ex 15 1

```
1 clc
2 //Chapter15
3 //Example15.1, page no 533
4 //Given
5 P_A=0.5// probability of producing symbol 'A'
6 P_B=0.25// probability of producing symbol 'B'
7 P_C=0.25// probability of producing symbol 'C'
8 H=P_A*log2(1/P_A)+P_B*log2(1/P_B)+P_C*log2(1/P_C)//
   the source entropy
9 mprintf('The source entropy is: %f bits/symbol',H)
```

Scilab code Exa 15.2 BIT Ex 15 2

```
1 clc
2 //Chapter15
3 //Example15.2, page no 535
4 //Given
```

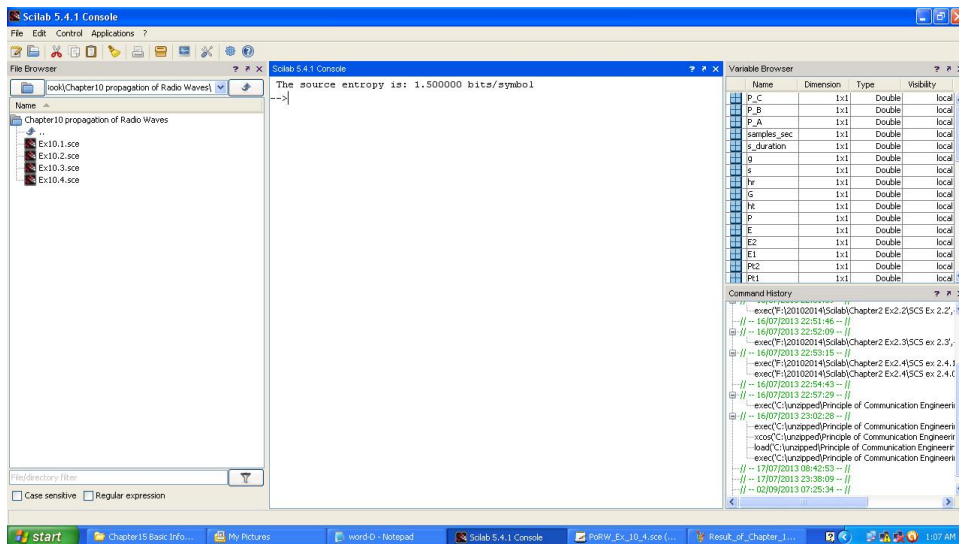


Figure 15.1: BIT Ex 15 1

```

5 P_A=0.5 , P_B=0.25 , P_C=1/32 , P_D=1/8 , P_E=1/16 , P_F=1/32
  // probabilities of producing respective symbol
6 H=(P_A*log2(1/P_A))+(P_B*log2(1/P_B))+(P_C*log2(1/
  P_C))+(P_D*log2(1/P_D))+(P_E*log2(1/P_E))+(P_F*
  log2(1/P_F)) // Source Entropy
7 n=6 , T=1
8 mprintf('The source entropy is: %f bits/symbol',
  round(1000*H)/1000)

```

Scilab code Exa 15.3 BIT Ex 15 3

```

1 // the Answer in the book is wrong.It is printed as
  90.4 for SNR3 but it should be 100.59
2 clc
3 //Chapter15
4 //Example15.3 , page no 536

```

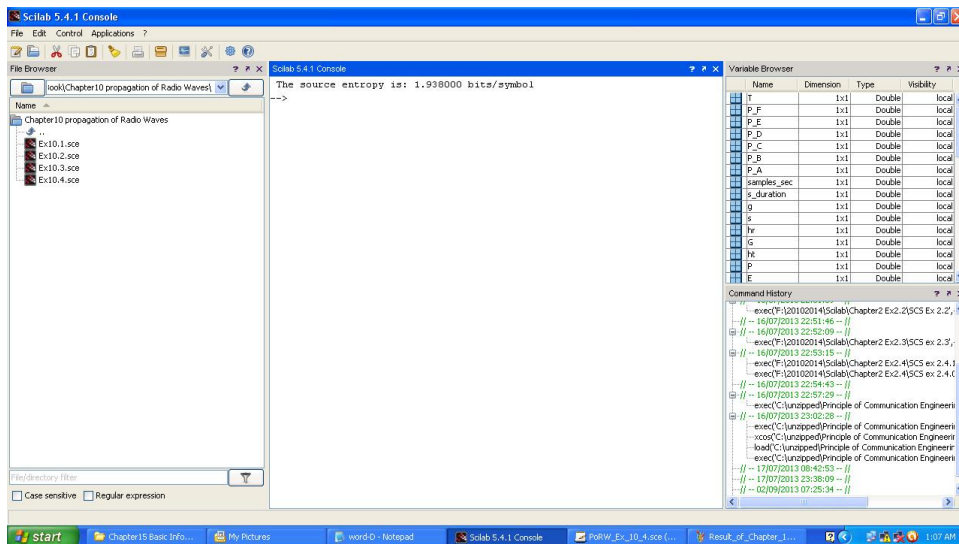


Figure 15.2: BIT Ex 15 2

```

5 //Given
6 //a
7 B1=4e3//Channel Bandwidth
8 SNR1=31//Channel SNR
9 C1=B1*log2(1+SNR1)//Channel Capacity
10 SNR2=14//Reduced SNR
11 B2=round(C1/log2(1+SNR2))//Bandwidth for reduced SNR
    with same Channel capacity
12
13 //b
14 B3=3e3//Reduced Bandwidth
15 SNR3=(2^(C1/B3))-1//Signal Power for reduced
    bandwidth
16 mprintf('a)\n Channel capacity is: %d Kbits/sec\n
    Bandwidth: %d KHz\nb)\n SNR for 3KHz
    bandwidth: %f',C1*1e-3,B2*1e-3,SNR3)

```

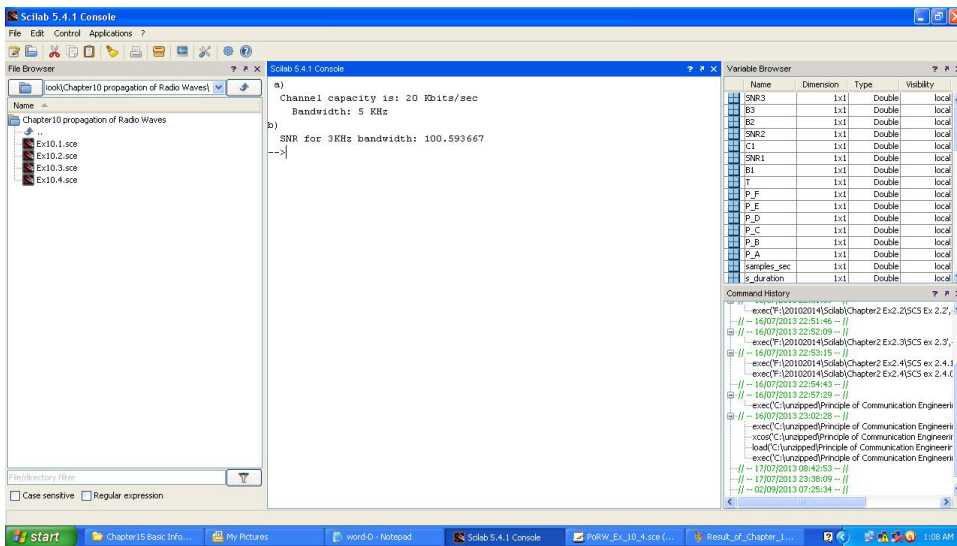


Figure 15.3: BIT Ex 15 3