Scilab Textbook Companion for Principle of Communication Engineering by A. Singh and A. K. Chhabra[1]

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

July 31, 2019

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Textbook Companion and Scilab codes written in it can be downloaded from the "Textbook Companion Project" section at the website http://scilab.in

Book Description

Title: 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`

```
clc
//Chapter1: Signals
//Example1,page no12
//Given:
n=round(rand())//any integers
m=round(rand())//any integers
wo=2*(n+m)*%pi//Angular Freq
t=0:0.01:2*%pi/wo
to=0,t1=2*%pi/wo
C= integrate('sin(n*wo*t)*cos(m*wo*t)','t',to,t1)//
   integrating sin(n*wo*t)*cos(m*wo*t) function
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)
```

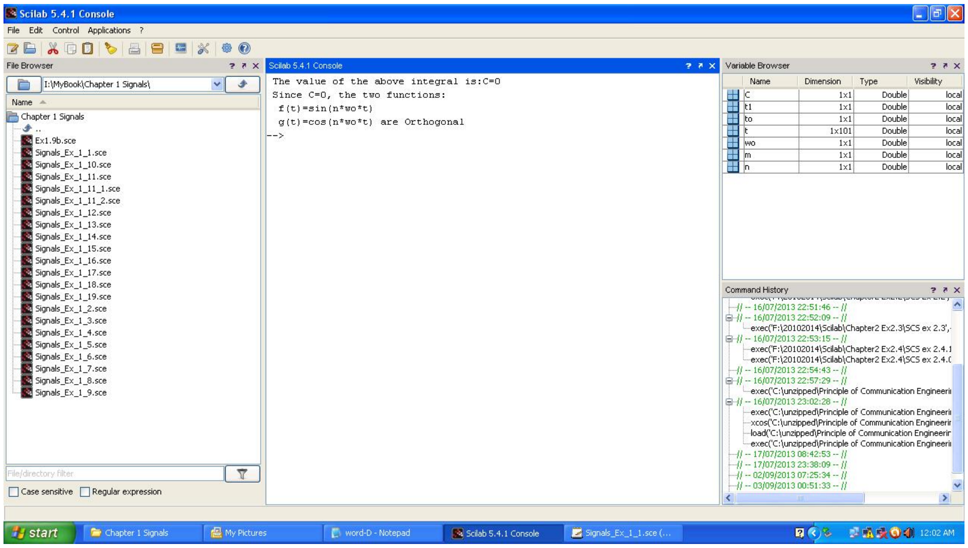

Figure 1.1: Signals Ex 1 1

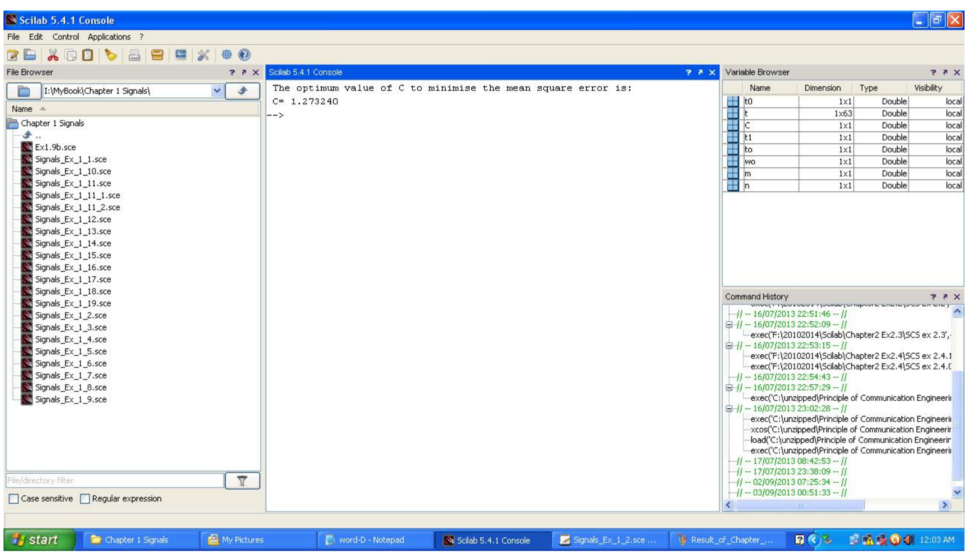

Figure 1.2: Signals Ex 1 2

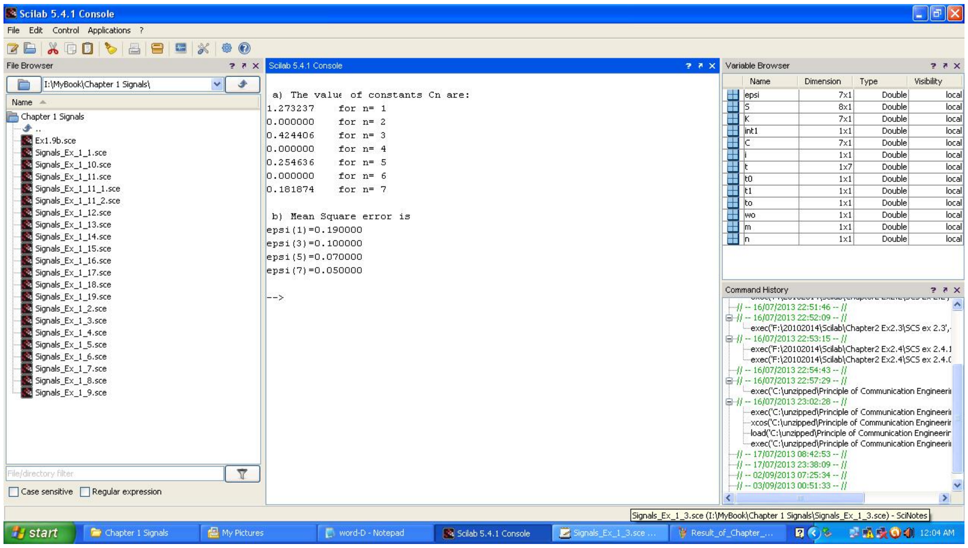

Figure 1.3: Signals Ex 1 3

Scilab code Exa 1.2 `Signals Ex 1 2`

```
clc
//Chapter1: Signals
//Example1,page no 12
//Given:
// Curve on page no 9....fig 1.6
t=0:0.1:2*%pi,t0=0,t1=2*%pi
C=((integrate('sin(t)','t',t0,t1/2)-integrate('sin(t
    )','t',t1/2,t1))/integrate('(sin(t))^2','t',t0,t1
    ))

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`

```
clc
//Chapter1: Signals
//Example2,page no12
//Given:
//a  // Referance Figure on page no 9.. (1.6d)
t=0:2*3.14,t0=0,t1=2*3.14
disp('a) The value of constants Cn are:')
for i=1:7
    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))
    if C(i)<=0.01 then C(i)=0
    end
    mprintf('%f    for n= %d\n',C(i),i)
end
//b Mean Square error

int1=integrate('(1)^2','t',t0,t1)
for n=1:7
    if modulo(n,2)==0 then
        C(n)=0
    else
    C(n)=4/(n*%pi)
end
end
for n=1:7

    K(n)=integrate('(sin(n*t))^2','t',t0,t1)

end
K(n)=%pi

for n=1:7
S(1)=0
    S(n+1)=S(n)+(((C(n))^2)*K(n))

end
//Mean Square error
```

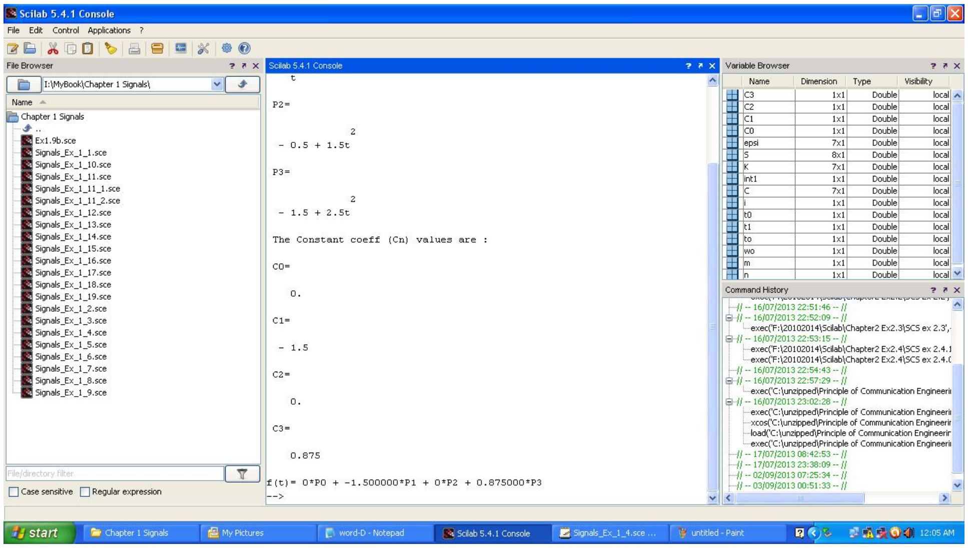

Figure 1.4: Signals Ex 1 4

```
for n=1:7
    epsi(n)=(1/(t1-t0)*(int1-S(n+1)))
end
disp('b) Mean Square error is')
for n=1:2:7
mprintf('epsi(%d)=%f\n',n,round(100*epsi(n))/100)

end
```

Scilab code Exa 1.4 Signals Ex 1 4

```
clc
//Chapter1: Signals
//Example1,page no12
//Given:
t=-1:0.01:1,t0=-1,t1=1
```

```
// Legendre Polynomial
t=poly(0,"t")
P0=poly(1,"t","c")
P1=poly([0,1],"t","c")
P2=poly([-0.5,0,1.5],"t","c")
P3=poly([-1.5,0,2.5,0],"t","c")
disp(P3,'P3=',P2,'P2=',P1,'P1=',P0,'P0=')
//The Constant coeff (Cn)
C0=0.5*(integrate('1','t',-1,0)+integrate('-1','t',0,1))
C1=1.5*(integrate('t','t',-1,0)+integrate('-t','t',0,1))
C2=2.5*(integrate('(1.5*t^2)-0.5','t',-1,0)+integrate('-(1.5*t^2)+0.5','t',0,1))
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))
disp('The Constant coeff (Cn) values are :')
disp(C3,'C3=',C2,'C2=',C1,'C1=',C0,'C0=')
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

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

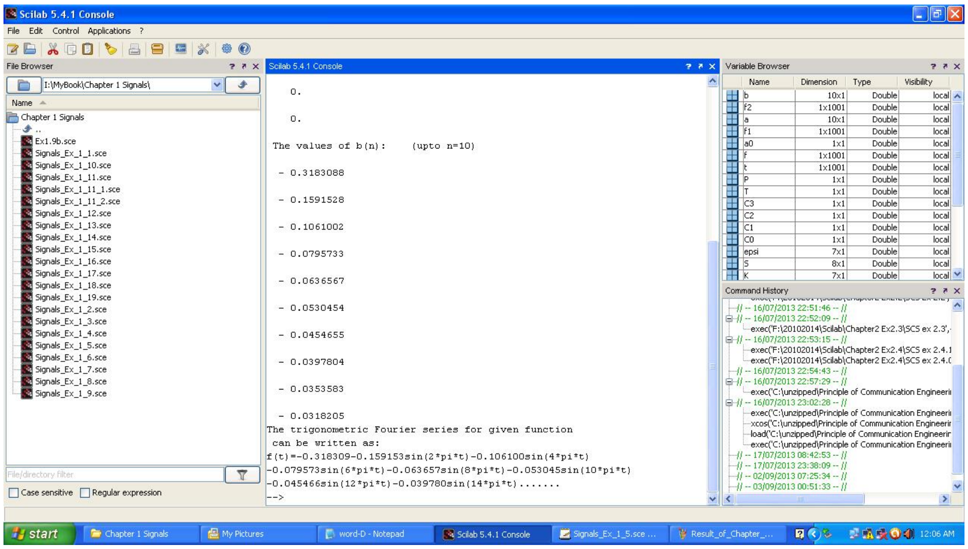

Figure 1.5: Signals Ex 1 5

```
a0=(1/T)*inttrap(t,f)

for n=1:10
    f1=(P*t).*cos(wo*n*t)
    a(n)=(2/T)*inttrap(t,f1)
    if a(n)<0.01 then
        a(n)=0
    end
end
for n=1:10
    f2=(P*t).*sin(2*%pi*(1/T)*n*t)
    b(n)=(2/T)*inttrap(t,f2)
end
// Displaying trigonometric Fourier series coeff
mprintf('The value of a0 is: %f\n',a0)
disp('The values of a(n):   (upto n=10)')
for n=1:10
    disp(a(n))
end
disp('The values of b(n):    (upto n=10)')
```

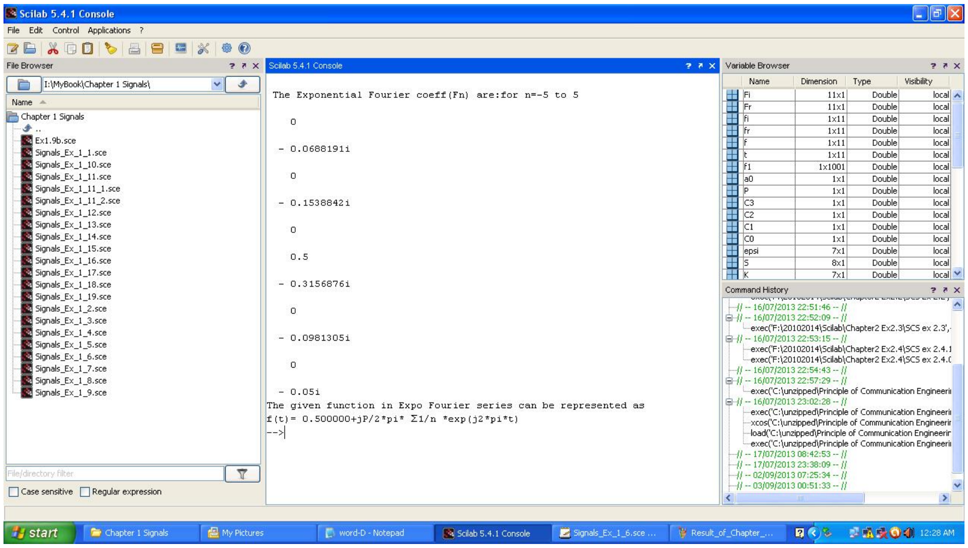

Figure 1.6: Signals Ex 1 6

```
for n=1:10
    disp(b(n))
end
mprintf('The trigonometric Fourier series for given
    function\n can be written as:\n')
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`

```
clc
//Chapter 1 Signals
//Example 1.6, page no 21
//given
```

```

t0=1,T=1,w0=2*3.14/T,P=1
t=0:0.1:1
f=P*t// function f(t)=P*t, 0<t<1
a=1
disp('The Exponential Fourier coeff(Fn) are:for n=-5
    to 5')
for n=-5:5// Calculating the fourier coeff
    fr=f.*cos(%pi*n*t/T)
    Fr(a)=inttrap(t,fr)
    fi=f.*sin(%pi*n*t/T)
    Fi(a)=inttrap(t,fi)
    if Fr(a)<0.01 then Fr(a)=0
    end
    if Fi(a)<0.01 then
        Fi(a)=0
    end
    disp(Fr(a)-%i*Fi(a))
    a=a+1
end
mprintf('The given function in Expo Fourier series
   can be represented as \n')
mprintf('f(t)= %f+jP/2*pi*   1  /n *exp(j2*pi*t)',P
   /2)
```

Scilab code Exa 1.7 Signals Ex 1 7

```
clc
close
clear
//Chapter 1 Signals
//Example 1.7, page no 22
//given
```

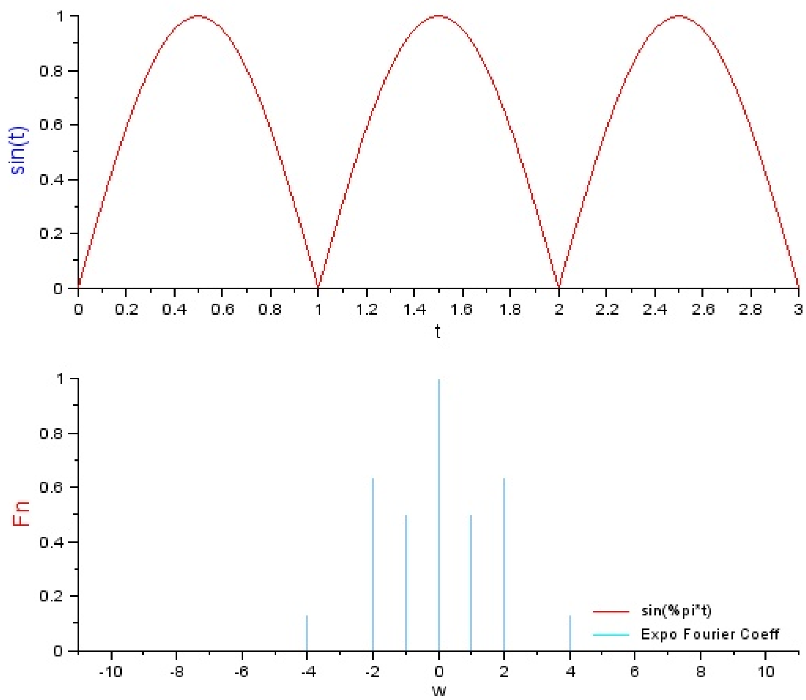

Figure 1.7: Signals Ex 1 7

```
V=1
t0=1,T=1,w0=2*3.14/T,P=1
t=0:0.01:3
f=V*abs(sin(%pi*t))
//The Expo fourier series coeff
disp('The Expo fourier series coeff are: for n=-5 to 5')
a=1
for n=-5:5
    fr=f.*cos(%pi*n*t/T)
    Fr(a)=inttrap(t,fr)
    fi=f.*sin(%pi*n*t/T)
    Fi(a)=inttrap(t,fi)

    mag(a)=abs(Fr(a)+%i*Fi(a))

    disp(Fr(a)-(%i*Fi(a)))
    x(1 ,size(t,2))=0
    x=x+((Fr(a))-%i*Fi(a)).*(cos(%pi*n*t/T)+%i*sin(%pi*n*t/T))
    a=a+1
end
mprintf('The given function in Expo Fourier series can be represented as \n')
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      -2V*exp(-j2*pi*t)/3*pi -2V*exp(-j2*pi*t)/15*pi...')
n=-5:5
subplot(2,1,1),plot2d(t,f,style=5)// Rectified sine function Plot
xlabel("t", "fontsize", 2);
ylabel("sin(t)", "fontsize", 3, "color", "blue");
subplot(2,1,2),plot2d3(n,mag,12,rect=[-11,0,11,1],style=4)//Plot of the magnitude of the Fourier coeff
xlabel("w", "fontsize", 2);
ylabel("Fn", "fontsize", 3, "color", "red");
```

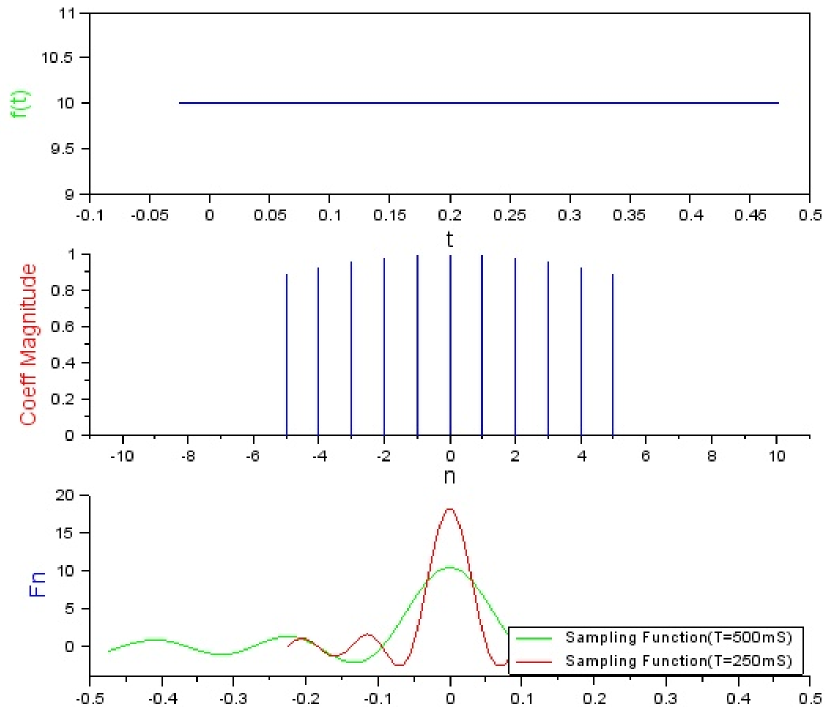

Figure 1.8: Signals Ex 1 8

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

Scilab code Exa 1.8 Signals Ex 1 8

```
clc
clear
close
//Chapter 1 Signals
//Example 1.8, page no 24
//given
```

```

T=500e-3,w0=2*%pi/T,d=50e-3,A=10
t=-d/2:0.01:T-d/2
t1=-d/2:0.01:d/2
f1=A
t2=d/2:0.01:T-(d/2)
f2=0
a=1
disp('The fourier series coeff Fn are:')
for n=-5:5
    if n==0 then
       Fr(a)=1,Fi(a)=0
    else
    fa1=f1.*cos(%pi*n*t1/T)
    fa2=f2.*cos(%pi*n*t2/T)
     fb1=f1.*sin(%pi*n*t1/T)
    fb2=f2.*sin(%pi*n*t2/T)
    end
    Fr(a)=1/T*(inttrap(t1,fa1)+inttrap(t2,fa2))
    Fi(a)=1/T*(inttrap(t1,fb1)+inttrap(t2,fb2))
    mag(a)=abs(Fr(a)+%i*Fi(a))
    //  disp(mag(a))
    disp(Fr(a)-%i*Fi(a))
     x(1,size(t,2))=0
    x=x+((Fr(a))-%i*Fi(a)).*(cos(%pi*n*t/T)+%i*sin(%pi*n*t/T))
    a=a+1

end
n=-5:5
subplot(3,1,1),plot(t,f1)
xlabel("t", "fontsize", 3);
ylabel("f(t)", "fontsize", 3, "color", "green");
subplot(3,1,2),plot2d3(n,mag,2,rect=[-11,0,11,1])//expo fourier series coeff
xlabel("n", "fontsize", 3);
ylabel("Coeff Magnitude", "fontsize", 3, "color", "red");
```

```
subplot(3,1,3),plot2d(t,x,style=3),plot2d(-t,x,style=3) // one sided spectrum with T=500ms
xlabel("w", "fontsize", 3);
ylabel("Fn", "fontsize", 3, "color", "blue");
legends(['Sampling Function(T=500mS)';'Sampling Function(T=250mS)'],[3,5],opt="lr")
T1=T/2
t=-d/2:0.01:T1-d/2
t1=-d/2:0.01:d/2
f1=A
t2=d/2:0.01:T1-(d/2)
f2=0
//The Expo fourier series coeff
for n=-5:5
    if n==0 then
       Fr1(a)=1,Fi1(a)=0
    else
    fr1=f1.*cos(%pi*n*t1/T1)
    fr2=f2.*cos(%pi*n*t2/T1)
     fi1=f1.*sin(%pi*n*t1/T1)
    fi2=f2.*sin(%pi*n*t2/T1)
    end
    Fr1(a)=1/T1*(inttrap(t1,fr1)+inttrap(t2,fr2))
    Fi1(a)=1/T1*(inttrap(t1,fi1)+inttrap(t2,fi2))
    mag(a)=abs(Fr1(a)+%i*Fi1(a))
    disp(Fr1(a)-%i*Fi1(a))
     y(1,size(t,2))=0
    y=y+((Fr1(a))-%i*Fi1(a)).*(cos(%pi*n*t/T1)+%i*sin(%pi*n*t/T1))
     a=a+1
end
plot2d(t,y, style=5),plot2d(-t,y,style=5)// double sided spectrum with T=250ms
```

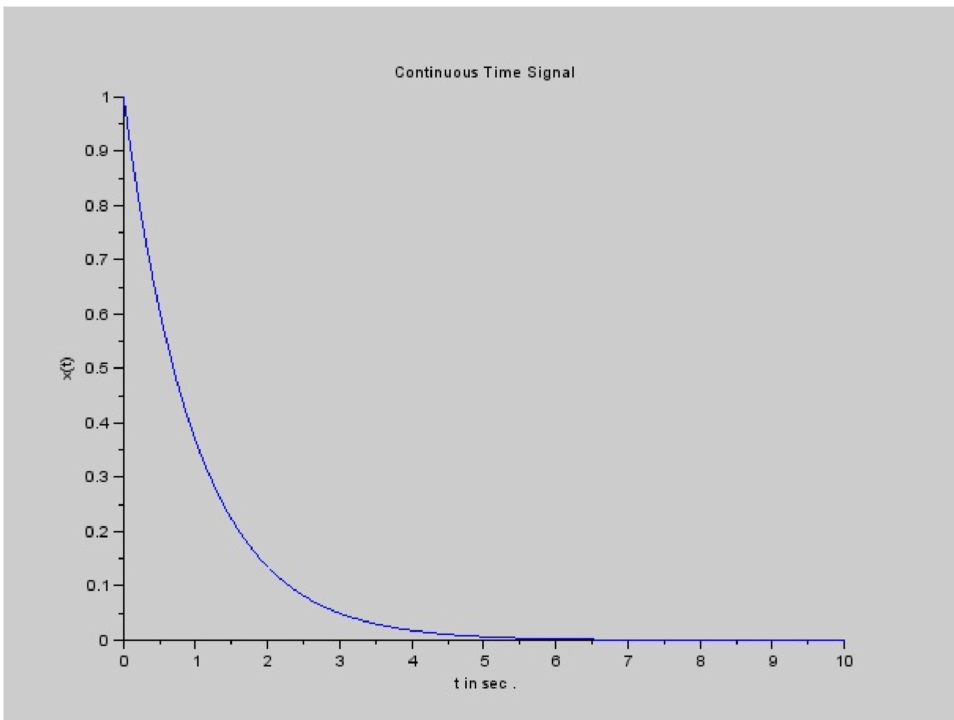

Figure 1.9: Signals Ex 1 9

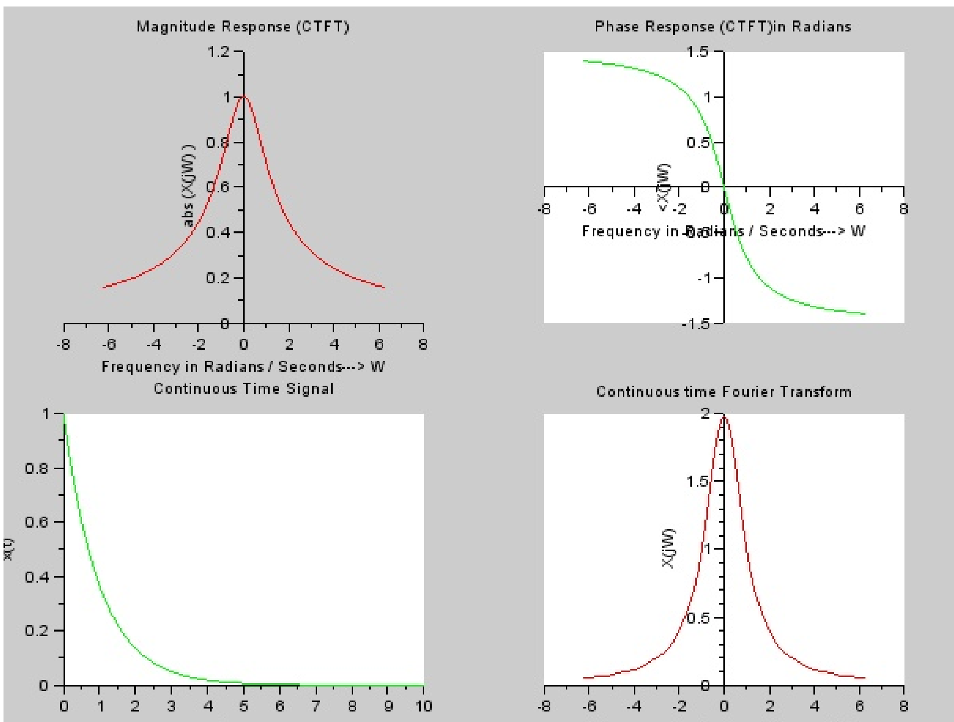

Figure 1.10: Signals Ex 1 9

Scilab code Exa 1.9 Signals Ex 1 9

```
clc; clear;close;
//Chapter1: Signals
//Example 1.9a,page no12
//Given:
// Analog Signal
A =1; // Amplitude
Dt = 0.005;
t = 0: Dt :10;
xt = exp(-A*t);
// Continuous time Fourier Transform
Wmax =2*%pi*1; // Analog Frequency = 1Hz
K = 4;
k = 0:(K/1000):K;
W = k* Wmax /K;
XW = xt*exp(-sqrt(-1)*t'*W)*Dt
XW_Mag =abs(XW);
W = [-mtlb_fliplr(W),W(2:1001)]; // Omega from -Wmax
    to Wmax
XW_Mag=[mtlb_fliplr(XW_Mag ),XW_Mag(2:1001)];
[XW_Phase ,db] = phasemag (XW);
XW_Phase = [-mtlb_fliplr( XW_Phase ),XW_Phase
    (2:1001)];
// Plotting Continuous Time Signal
figure
a=gca();
a.y_location = "origin";
plot(t,xt);
xlabel( 't in sec .');
ylabel(' x(t) ')
title(' Continuous Time Signal ' )
figure
// Plotting Magnitude Response of CTS
```

```
subplot (2 ,2 ,1);
a = gca ();
a.y_location = "origin";
plot2d(W, XW_Mag,style=5 );
xlabel ( ' Frequency in Radians / Seconds---> W' );
ylabel ( ' abs (X(jW) ) ' )
title ( 'Magnitude Response (CTFT) ' )
// Plotting Phase Reponse of CTS
subplot (2 ,2 ,2);
a =gca();
a.y_location="origin";
a.x_location="origin";
plot2d(W, XW_Phase *%pi/180,style=3);
xlabel(' Frequency in Radians / Seconds---> W');
ylabel('<X(jW) ')
title(' Phase Response (CTFT)in Radians' )
mprintf('|F(w)|= 1/sqrt(a^2+w^2) and\n Theta(w)=-
    atan(w/a)')

//Chapter1: Signals
//Example 1.9b,page no12
//Given:
// Analog Signal


A=1;// Amplitude
Dt=0.005;
t1=-4.5:Dt:4.5;
xt1=exp(-A*abs(t1));
// Continuous time Fourier Transform
Wmax1 =2*%pi*1;// Analog Frequency = 1Hz
K=4;
k=0:(K/1000):K;
W1=k*Wmax1/K;
XW1=xt1*exp(-sqrt(-1)*t1'*W1)*Dt;
XW1=real(XW1);
W1=[-mtlb_fliplr(W1), W1(2:1001) ]; // Omega from -
    Wmax to Wmax
```

```
XW1=[ mtlb_fliplr(XW1), XW1(2:1001) ];
//subplot(1 ,1 ,1)
subplot(2 ,2 ,3);
b=gca();
b.y_location = "origin";
plot2d(t,xt,3);
xlabel('t in sec.');
ylabel('x(t)')
title(' Continuous Time Signal')
subplot(2 ,2 ,4);
b =gca();
b.y_location = "origin";
plot2d(W1,XW1,5);
xlabel('Frequency in Radians / Seconds W');
ylabel('X(jW)')
title('Continuous time Fourier Transform ')

mprintf('|F(w)|= 2*a/sqrt(a^2+w^2) and\n Theta(w)=0')
```

Scilab code Exa 1.10 Signals Ex 1 10

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

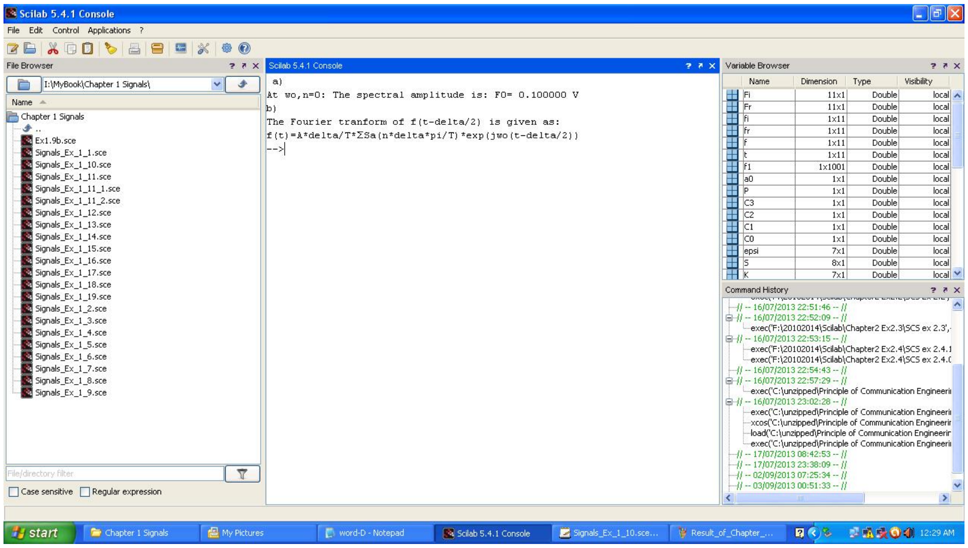

Figure 1.11: Signals Ex 1 10

```
            Sa=sin(n*%pi*delta/T)/(n*%pi*delta/T)
        end

end

F=(A*delta/T)*Sa//spectral Amplitude
mprintf('a)\nAt wo,n=0: The spectral amplitude is: F0= %f V\n',F)
//b
// displaying the fourier Transform of the given function
mprintf('b)\nThe Fourier tranform of f(t-delta/2) is given as: '),
mprintf('\nf(t)=A*delta/T*  Sa  (n*delta*pi/T)*exp(jwo(t-delta/2))')
```

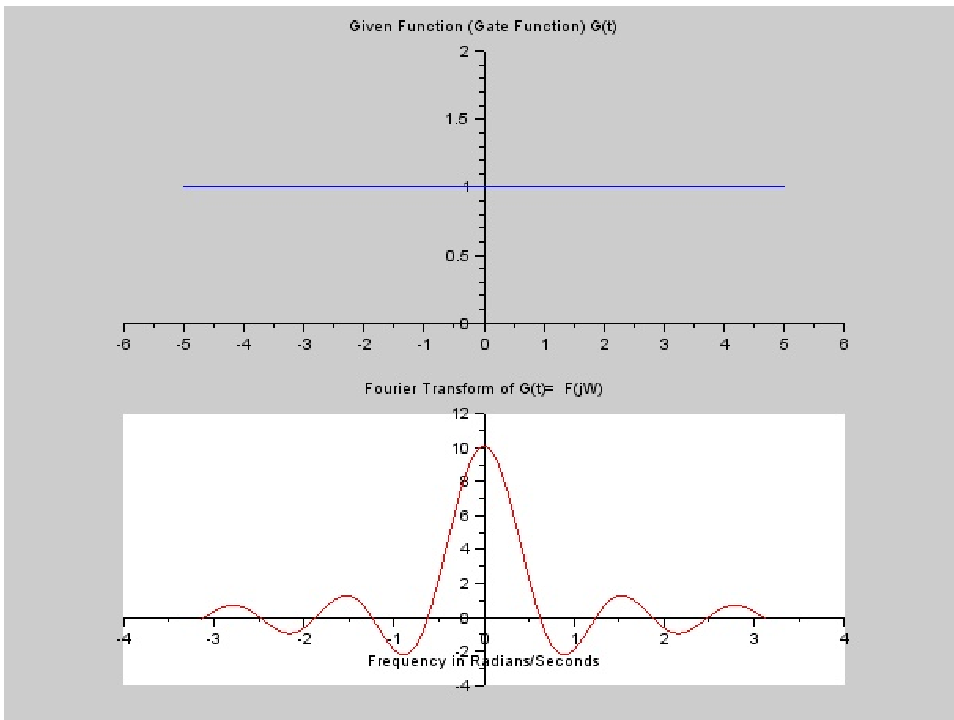

Figure 1.12: Signals Ex 1 11 Pg 39

Scilab code Exa 1.11.1 Signals Ex 1 11 Pg 39

```
clear ;
clc ;
close ;
//Chapter1
//Example1.11(1), page no 39
//Given
T = 10; //time Tau
Tg = -T/2 :0.1: T/2; // time period for given Gate Function -tau/2 to tau/2
G_t0 = 1; //Magnitude of Gate Function (A)
G_t = G_t0* ones (1, length (Tg));// Gate function G(t)
f = -%pi: %pi / length (Tg): %pi ;
Dw = 0.1;
F_jW =G_t*exp(sqrt(-1)*Tg'*f)*Dw;// fourier Transform of the gate function
F_jW = real(F_jW);
// Plotting the Fourier Transform of G(t)
figure
subplot (2 ,1 ,1)
a=gca();
a.y_location ="origin";
a.x_location ="origin";
plot2d(Tg,G_t,2);
xtitle( ' Given Function (Gate Function) G(t) ' )
subplot(2 ,1 ,2)
a= gca();
a.y_location ="origin";
a.x_location ="origin";
plot2d(f,F_jW,5);
xlabel('Frequency in Radians/Seconds ');
title('Continuous time Fourier Transform X(jW)' )
xtitle ( 'Fourier Transform of G(t)= F(jW) ' )
mprintf('F(w)= A*t*Sa(w*t/2) ')
```

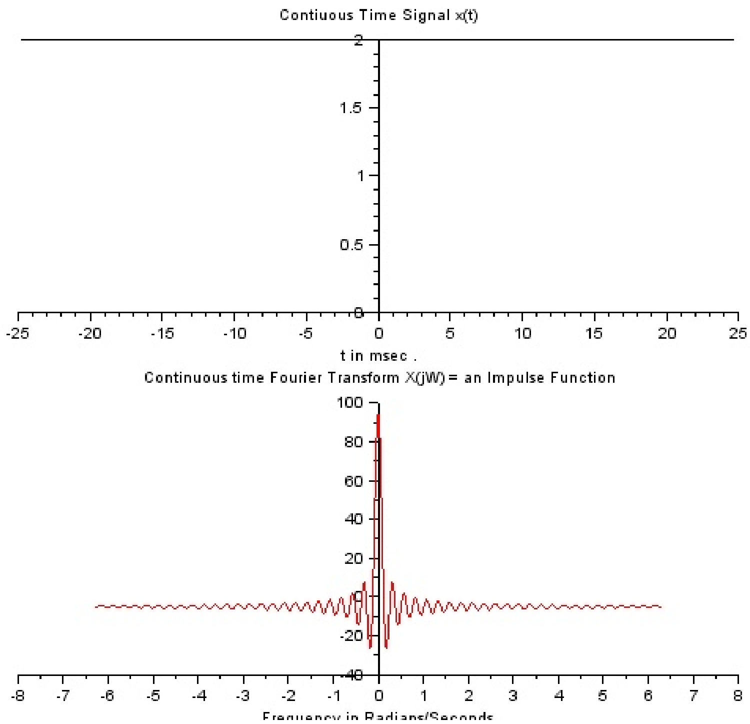

Figure 1.13: Signals Ex 1 11 Pg43

Scilab code Exa 1.11.2 `Signals Ex 1 11 Pg43`

```
clc
//Chapter1
//Example1.11(2), page no 43
//Given
clear ;
close ;
// CTS Signal
A=2;// Amplitude
```

```
Dt=0.01;
T1=49.5; //Time in seconds
t=-T1/2: Dt:T1 /2;
for i=1:length(t)
xt(i)= A;
end
// Continuous time Fourier Transform
Wmax=2*%pi*1;// Analog Frequency = 1Hz
K =4;
k=0:(K/1000):K;
W=k*Wmax/K;
xt=xt';
XW =(xt*exp(-sqrt(-1)*t'*W)*Dt)-5;
XW_Mag =real(XW);
W =[-mtlb_fliplr(W), W(2:1001)]; // Omega from -Wmax to Wmax
XW_Mag =[mtlb_fliplr( XW_Mag ), XW_Mag(2:1001)];
subplot(2 ,1 ,1);
a =gca();
a.data_bounds =[ -4 ,0;4 ,2];
a.y_location ="origin";
plot2d(t,xt);
xlabel('t in msec .');
title(' Contiuous Time Signal x(t) ')
subplot(2 ,1 ,2);
a=gca();
a.y_location ="origin";
plot2d(W,XW_Mag,5);
xlabel('Frequency in Radians/Seconds ');
title('Continuous time Fourier Transform X(jW) = an Impulse Function' )
mprintf('|F(w)|= 2*pi*A*delta(w), Hence the Fourier Transform of constant is an Impulse Function')
```

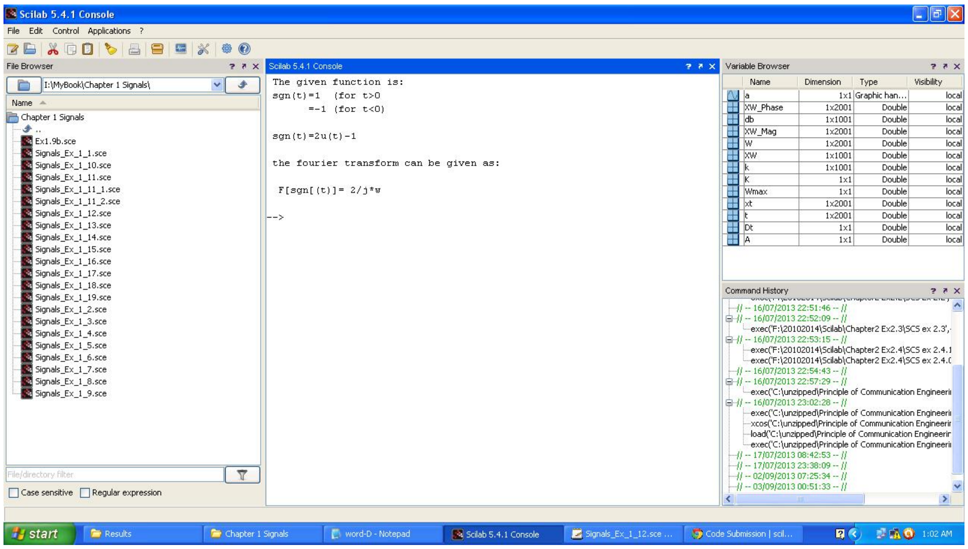

Figure 1.14: Signals Ex 1 12

Scilab code Exa 1.12 `Signals Ex 1 12`

```

clc
//Chapter 1
//Ex1.12, page no 43
//Given
mprintf('The given function is:\n sgn(t)=1  (for t
    >0\n         =-1 (for t<0)\n')// displaying the
    given function
disp('sgn(t)=2u(t)-1')
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`

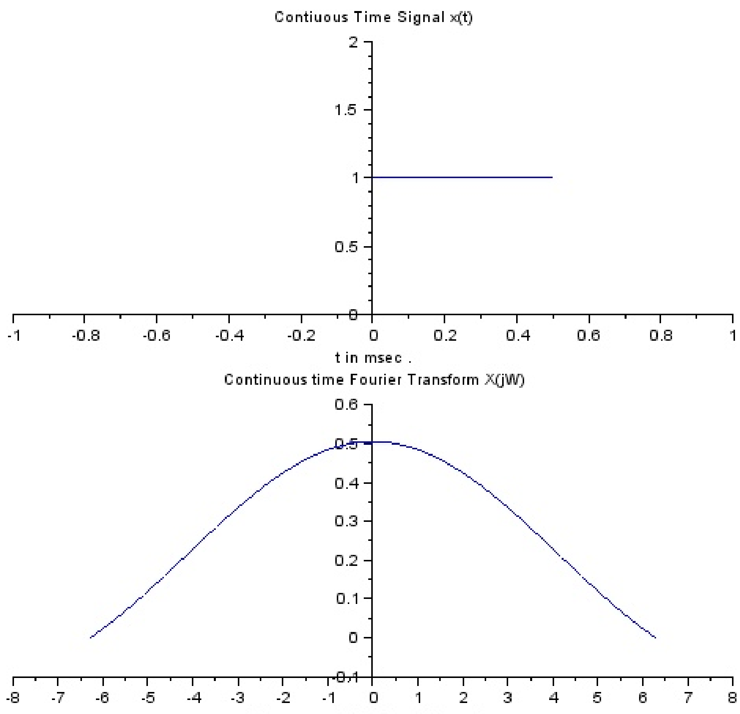

Figure 1.15: Signals Ex 1 13

```
clear ;
clc ;
close ;
//Chapter 1
//Ex1.13 , page no 44
// CTS Signal
A =1; // Amplitude
Dt = 0.005;
T1 =0.5; //Time in seconds
t=0:Dt:T1;
for i = 1: length(t)
xt(i)=A;
end
// Continuous time Fourier Transform
Wmax= 2*%pi*1; // Analog Frequency = 1Hz
K =4;
k=0:(K/1000):K;
W =k*Wmax/K;
xt=xt';
XW =xt*exp(-sqrt(-1)*t'*W)*Dt;
XW_Mag =real(XW);
W =[-mtlb_fliplr(W), W(2:1001)]; // Omega from -Wmax
    to Wmax
XW_Mag =[mtlb_fliplr( XW_Mag ), XW_Mag(2:1001)];
// displaying the given function
subplot(2 ,1 ,1);
a =gca();
a.data_bounds =[ -1,0;1,2];
a.y_location ="origin";
plot(t,xt);
xlabel('t in msec .');
title(' Contiuous Time Signal x(t) ')
// displaying the fourier Transform of the given
    function
subplot(2 ,1 ,2);
a=gca();
mprintf('F(w)= 1/(j*w)+ pi*delta(w)')
a.y_location ="origin";
```

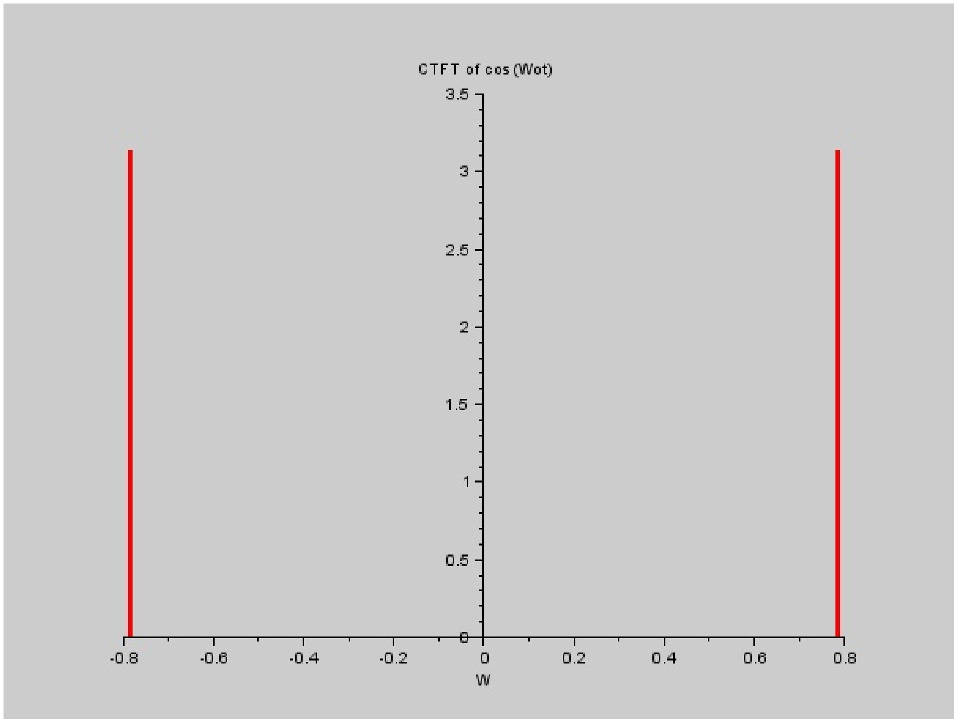

Figure 1.16: Signals Ex 1 14

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

Scilab code Exa 1.14 `Signals Ex 1 14`

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

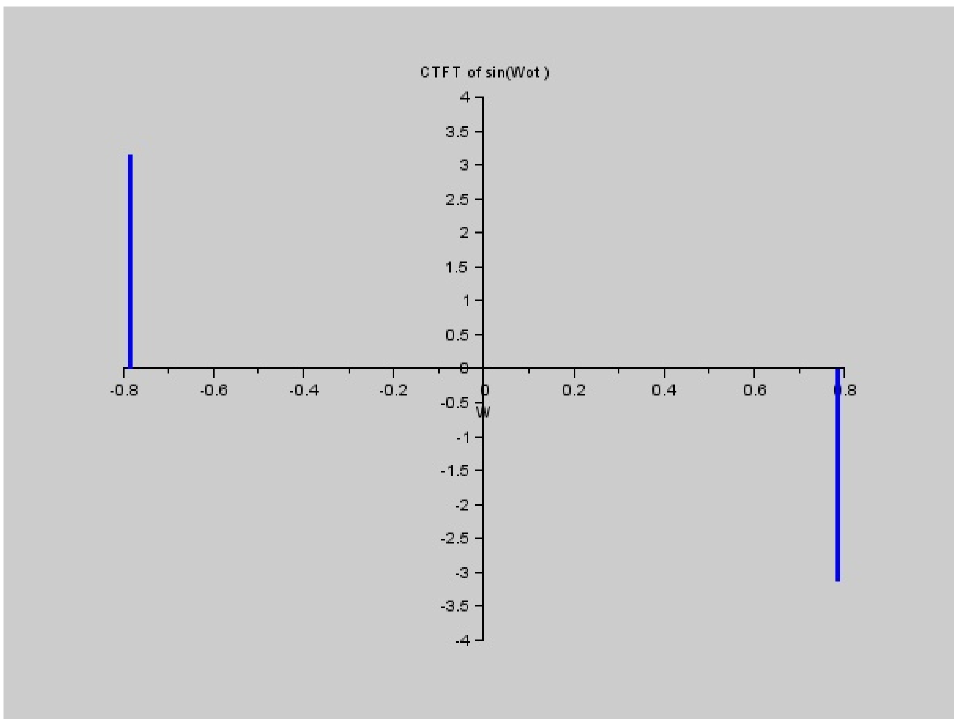

Figure 1.17: Signals Ex 1 14

```
// Continuous Time Fourier Transforms of
// Sinusoidal waveforms(a)sin(Wot)(b)cos(Wot)
clear ;
clc ;
close ;
// CTFT
T1 = 2;
T = 4* T1;
Wo = 2* %pi /T;
W = [-Wo ,0, Wo ];
ak = (2* %pi *Wo*T1/ %pi )/ sqrt ( -1);
XW = [-ak ,0, ak ];
ak1 = (2* %pi*Wo*T1/%pi);
XW1 =[ ak1 ,0, ak1 ];
//displaying the given function
figure
a =gca();
a.y_location="origin";
a.x_location="origin";
plot2d3('gnn',W,imag(XW),2);
poly1=a.children(1).children(1) ;
poly1.thickness =3;
xlabel('W' );
title( 'CTFT of sin(Wot ) ')
//displaying the fourier Transform of the given function
figure
a =gca();
a.y_location="origin";
a.x_location="origin";
plot2d3('gnn',W,XW1,5);
poly1 =a.children(1).children(1) ;
poly1.thickness = 3;
xlabel('W' );
title( 'CTFT of cos (Wot)')
```

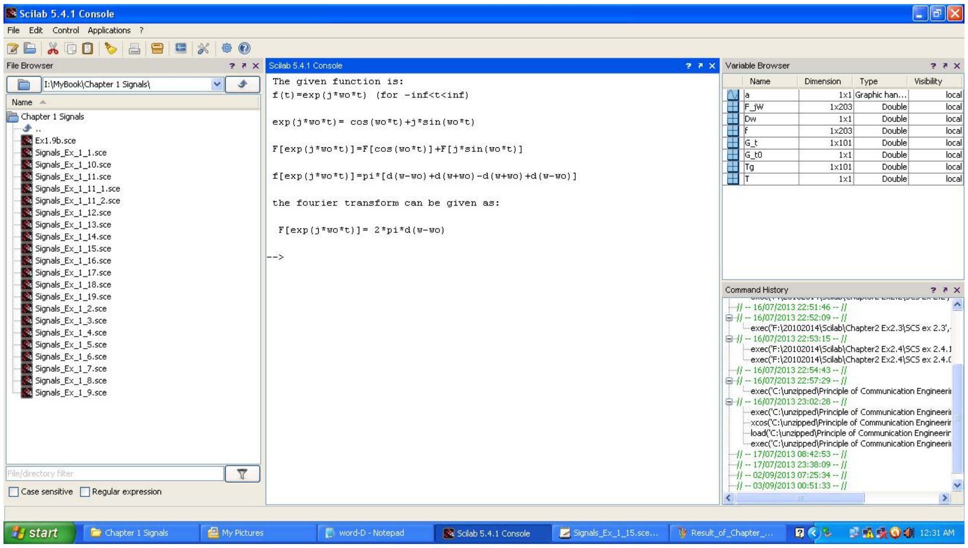

Figure 1.18: Signals Ex 1 15

Scilab code Exa 1.15 `Signals Ex 1 15`

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

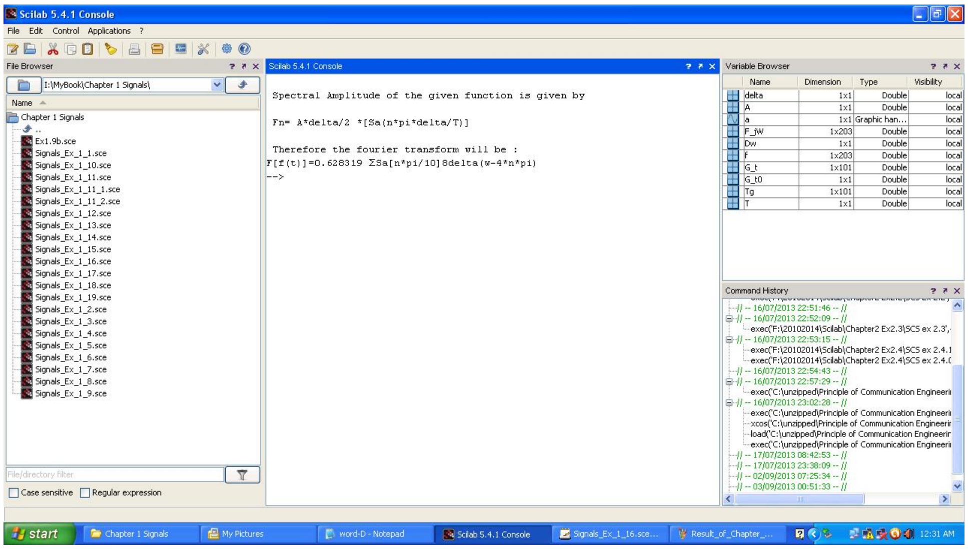

Figure 1.19: Signals Ex 1 16

```
    fourier Transform of the given function
```

Scilab code Exa 1.16 `Signals Ex 1 16`

```
clc

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

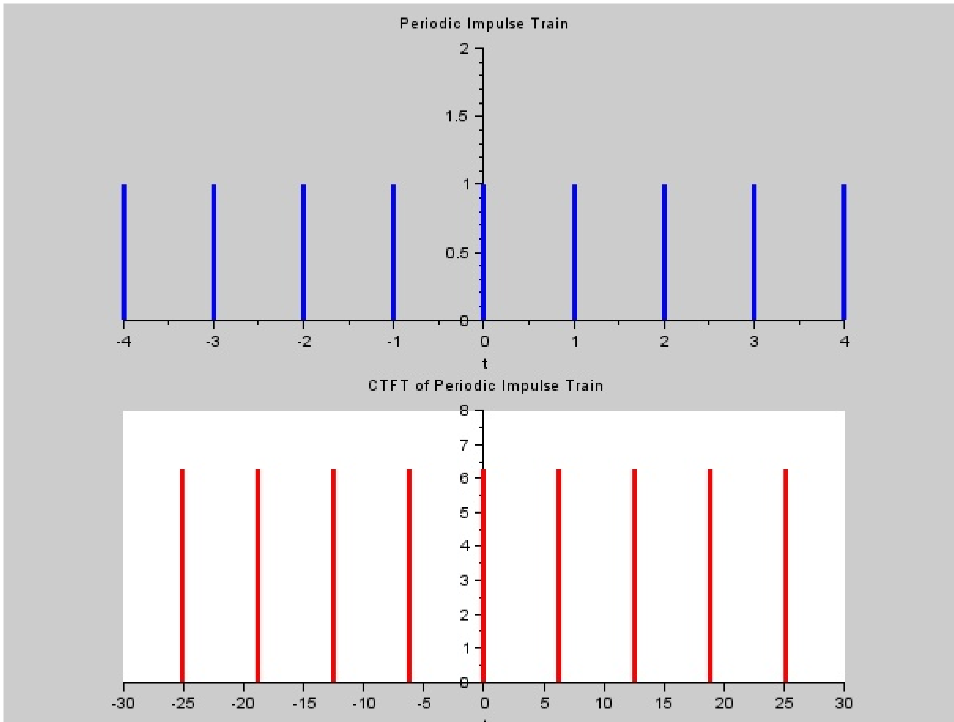

Figure 1.20: Signals Ex 1 17

```
transform
```

Scilab code Exa 1.17 `Signals Ex 1 17`

```
clc
//Chapter1: Signals
//Example1,page no12
//Given:
// CTFT
T = -4:4;;
T1 = 1; // Sampling Interval
```

```
xt = ones (1, length (T));
ak = 1/ T1;
XW = 2* %pi *ak* ones (1, length (T));
Wo = 2*%pi/T1;
W = Wo*T;
// Displaying the given function
figure
subplot(2 ,1 ,1)
a=gca();
a.y_location="origin";
a.x_location="origin";
plot2d3('gnn',T,xt,2) ;
poly1 =a.children(1).children (1) ;
poly1.thickness = 3;
xlabel ( 't ' );
title('Periodic Impulse Train ')
// displaying the fourier Transform of the given function
subplot(2 ,1 ,2)
a=gca();
a.y_location="origin";
a.x_location="origin";
plot2d3('gnn',W,XW,5) ;
poly1=a.children(1).children(1) ;
poly1.thickness =3 ;
xlabel('t');
title ( 'CTFT of Periodic Impulse Train')

mprintf('F[ t (t)]= 2*pi/T* (w-wo)')
```

Scilab code Exa 1.18 `Signals Ex 1 18`

```
clear ;
```

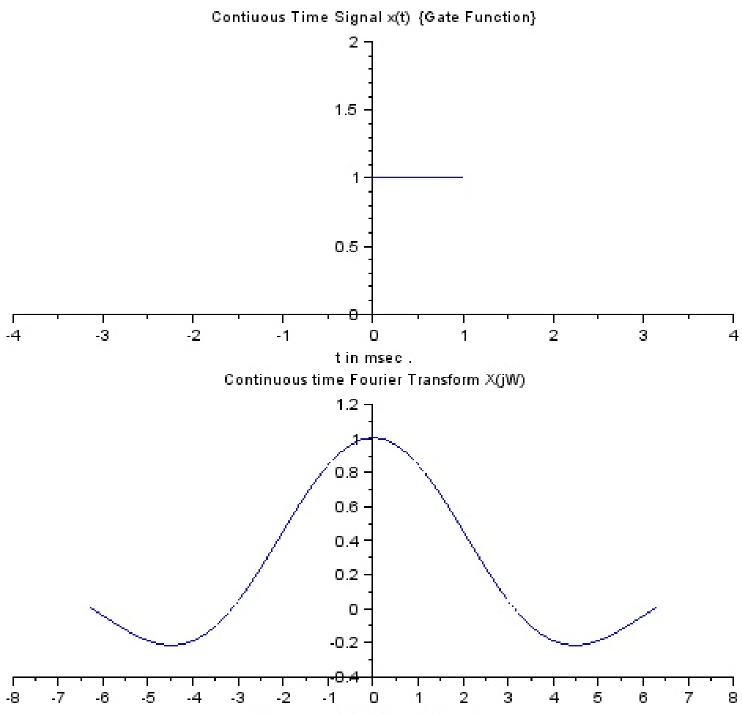

Figure 1.21: Signals Ex 1 18

```
clc ;
close ;
// CTS Signal
A =1; // Amplitude
Dt = 0.005;
T1 = 2; //Time in seconds
t = 0: Dt:T1 /2;
for i = 1: length (t)
xt(i) = A;
end
// Continuous time Fourier Transform
Wmax= 2*%pi*1; // Analog Frequency = 1Hz
K =4;
k=0:(K/1000):K;
W =k*Wmax/K;
xt=xt';
XW =xt*exp(-sqrt(-1)*t'*W)*Dt;
XW_Mag =real(XW);
W =[-mtlb_fliplr(W), W(2:1001)]; // Omega from  Wmax to Wmax
XW_Mag =[mtlb_fliplr( XW_Mag ), XW_Mag(2:1001)];
// displaying the given function
subplot(2 ,1 ,1);
a =gca();
a.data_bounds =[ -4 ,0;4 ,2];
a.y_location ="origin";
plot(t,xt);
xlabel('t in msec .');
title(' Contiuous Time Signal x(t)  {Gate Function}')
// displaying the fourier Transform of the given function
subplot(2 ,1 ,2);
a=gca();
a.y_location ="origin";
plot(W, XW_Mag);
xlabel('Frequency in Radians / Seconds ');
title('Continuous time Fourier Transform X(jW)' )
```

```
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

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

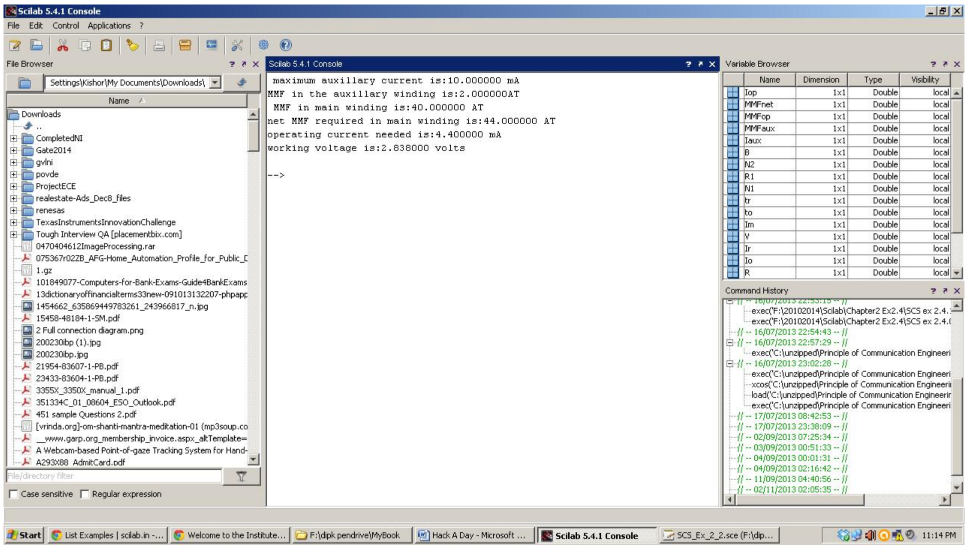

Figure 2.1: Chapter 2 Example 2 2

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

Scilab code Exa 2.3 Chapter 2 Example 2 3

```
clc;
// Chapter 2 Switched communication systems
//Example 2.3,page no 125
//given
```

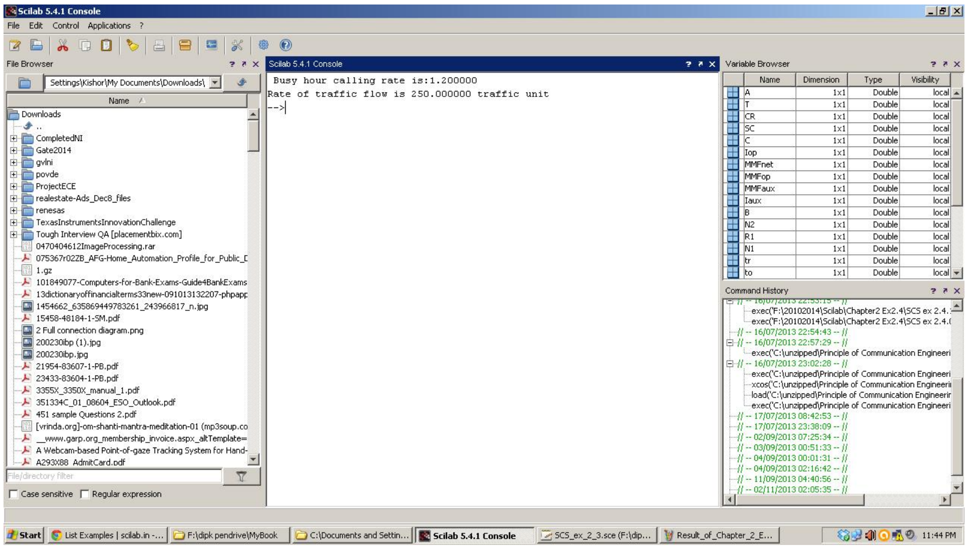

Figure 2.2: Chapter 2 Example 2 3

```
C=6000//Tatol no of call in busy hour
SC=5000//no of subscribers
CR=C/SC//busy hour calling rate
mprintf('Busy hour calling rate is:%f \n',CR)
T=2.5/60//avarage duration of calls in hours

A=C*T//rate of traffic flow
mprintf('Rate of traffic flow is %f traffic unit ',A
    )
```

Scilab code Exa 2.4.0 Chapter 2 Example 2 4 Pg126 Top

```
clc;
// Chapter 2 Switched communication systems
//Example 2.4,page no 126
//given
```

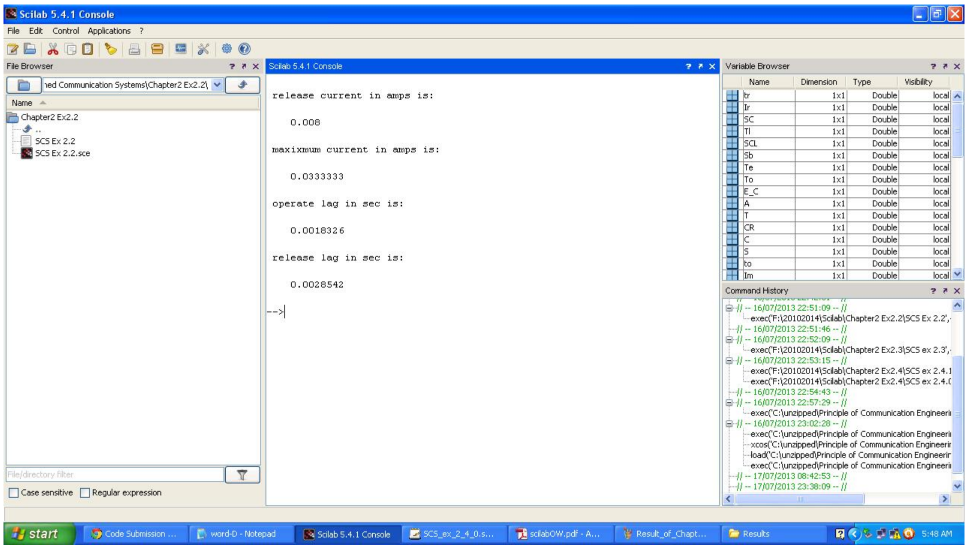

Figure 2.3: Chapter 2 Example 2 4 Pg126 Top

```
L=3//relay inductance in henry
R=1500//relay resistance in ohm
Io=20e-3//oparating current in amps
Ir=8e-3//release current in amps

V=50//supply volatage in volts
Im=V/R//maxixmum current in amps
mprintf('maxixmum current is %f mamps \n',Im*1e3)
to=(L/R)*log(1/(1-(Io/Im)))//operate lag in sec
mprintf('operate lag is %f msec \n',to*1000)
tr=(L/R)*log(Im/Ir)//release lag in sec
mprintf('release lag is %f msec \n',tr*1000)
```

Scilab code Exa 2.4.1 Chapter 2 Example 2 4 Pg126 Bottom

```
clc;
```

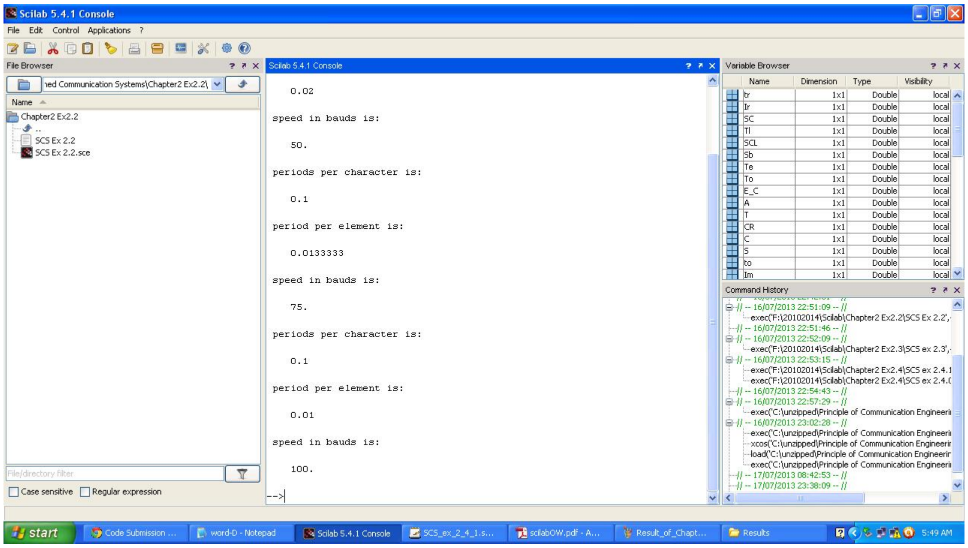

Figure 2.4: Chapter 2 Example 2 4 Pg126 Bottom

```
// Chapter 2 Switched communication systems
//Example 2.4.1,page no 126
//given
//a
C_S1=20/3//speed in characters per second
P_C1=1/C_S1//periods per character
mprintf('(a)\nperiods per character is:%f msec\n',
    P_C1*1e3)
E_C1=7.5//elements per character
P_E1=P_C1/E_C1//period per element
mprintf('period per element is:%f msec\n',P_E1*1e3)
Sb1=1/P_E1//speed in bauds
mprintf('speed is:%f bauds\n\n',Sb1)
//b
C_S2=10//speed in characters per second
P_C2=1/C_S2//periods per character
mprintf('(b)\nperiods per character is:%f msec\n',
    P_C2*1e3)
E_C2=7.5//elements per character
P_E2=P_C2/E_C2//period per element
```

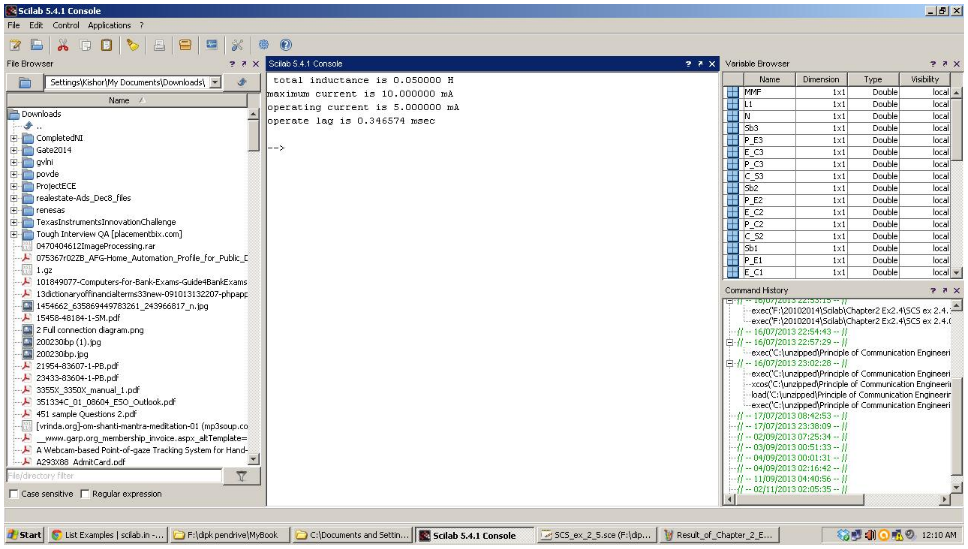

Figure 2.5: Chapter 2 Example 2 5

```
mprintf('period per element is:%f msec\n',P_E2*1e3)
Sb2=1/P_E2//speed in bauds
mprintf('speed is %f bauds\n\n', Sb2)
//c
C_S3=10//speed in characters per second
P_C3=1/C_S3//periods per character
mprintf('(c)\nperiods per character is:%f msec\n',
    P_C3*1e3)
E_C3=10//elements per character
P_E3=P_C3/E_C3//period per element
mprintf('period per element is:%f msec\n',P_E3*1e3)
Sb3=1/P_E3//speed in bauds
mprintf('speed is %f bauds\n',Sb3)
```

Scilab code Exa 2.5 Chapter 2 Example 2 5

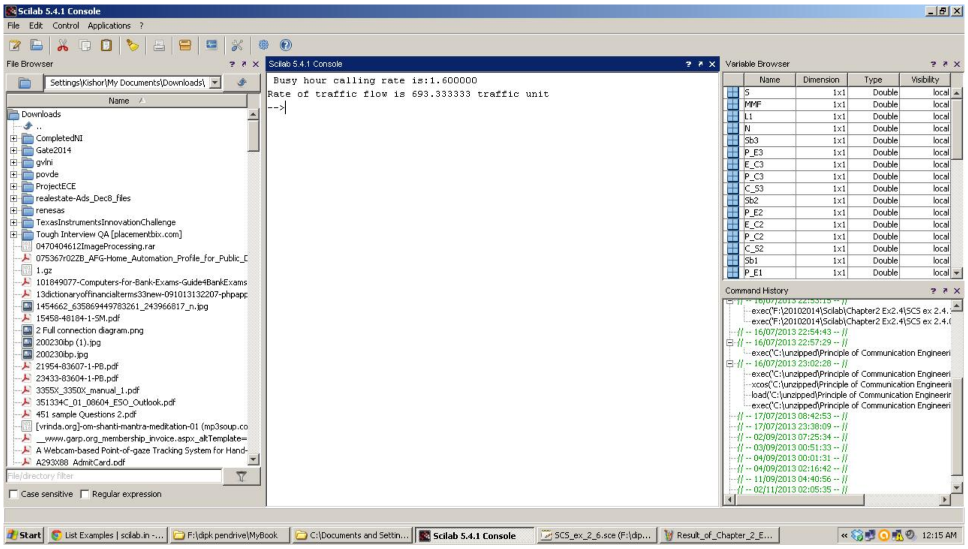

Figure 2.6: Chapter 2 Example 2 6

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

Scilab code Exa 2.6 Chapter 2 Example 2 6

```
clc;
// Chapter 2 Switched communication systems
//Example 2.6,page no 128
//given
S=10000//no of subscribers
C=16000//Tatol no of call in busy hour
CR=C/S//busy hour calling rate
mprintf('Busy hour calling rate is:%f \n',CR)
T=2.6//avarage duration of calls in min

A=C*(T/60)//rate of traffic flow
mprintf('Rate of traffic flow is %f traffic unit ',A
   )
```

Scilab code Exa 2.7 Chapter 2 Example 2 7

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

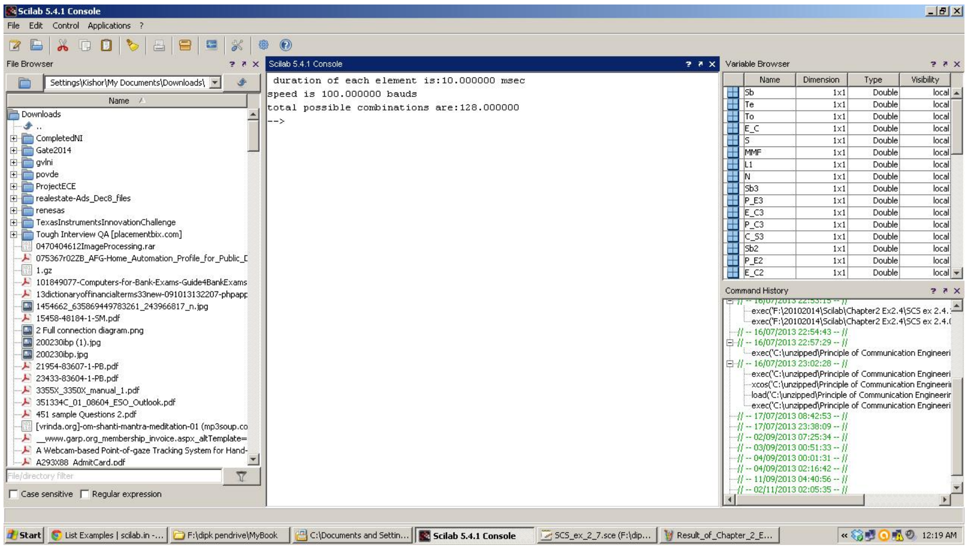

Figure 2.7: Chapter 2 Example 2 7

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

Scilab code Exa 2.8 Chapter 2 Example 2 8

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

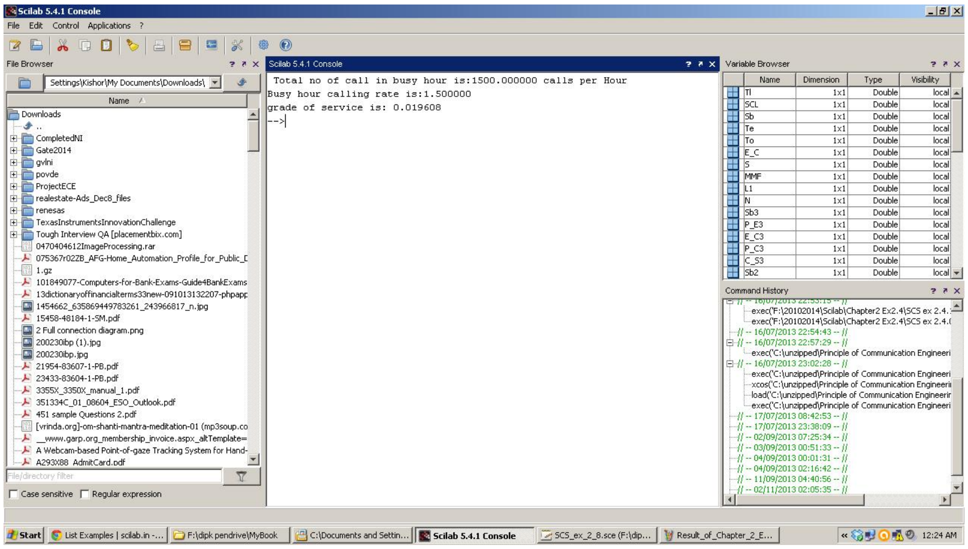

Figure 2.8: Chapter 2 Example 2 8

```
mprintf('Busy hour calling rate is:%f \n',CR)
SCL=30//no of call lost per hour

B=SCL/(C+SCL)//grade of service
mprintf('grade of service is: %f',B)
```

Scilab code Exa 2.9 Chapter 2 Example 2 9

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

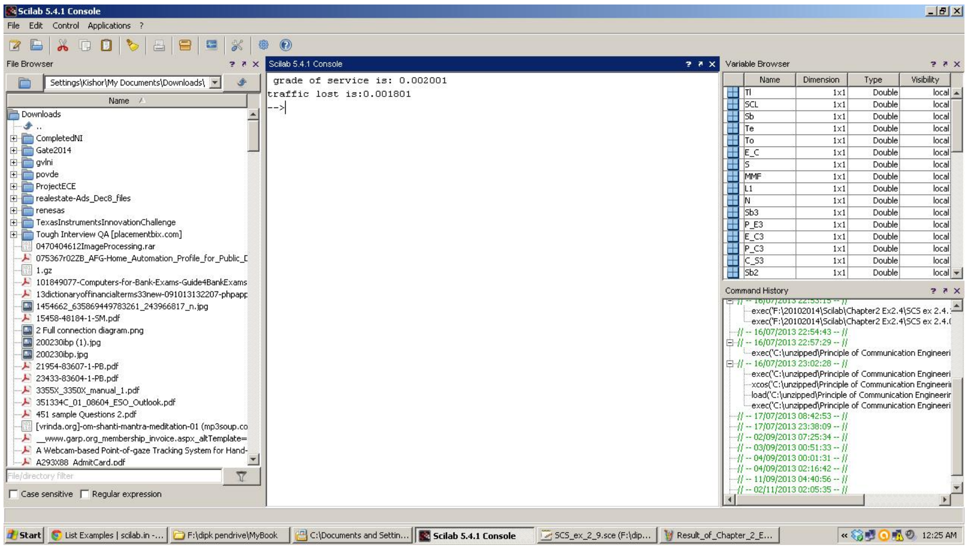

Figure 2.9: Chapter 2 Example 2 9

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

Chapter 3

Modulation

Scilab code Exa 3.1 Modulation Ex 3 1

```
clc
//Chapter3: Modulation
//Example3.1, page no 135
//Given
Ic=10 //carrier current in Amps
Imod=11.6// Current after modulation
Rl=1//Assumed load in ohm
Pmod=Rl*Imod^2//power before modulation
Ma= sqrt(2*((Pmod/Ic^2)-1))//percentage modulation
Pc=10
Pmod=Pc*(1+(Ma^2/2))//power after modulation
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

```
clc
```

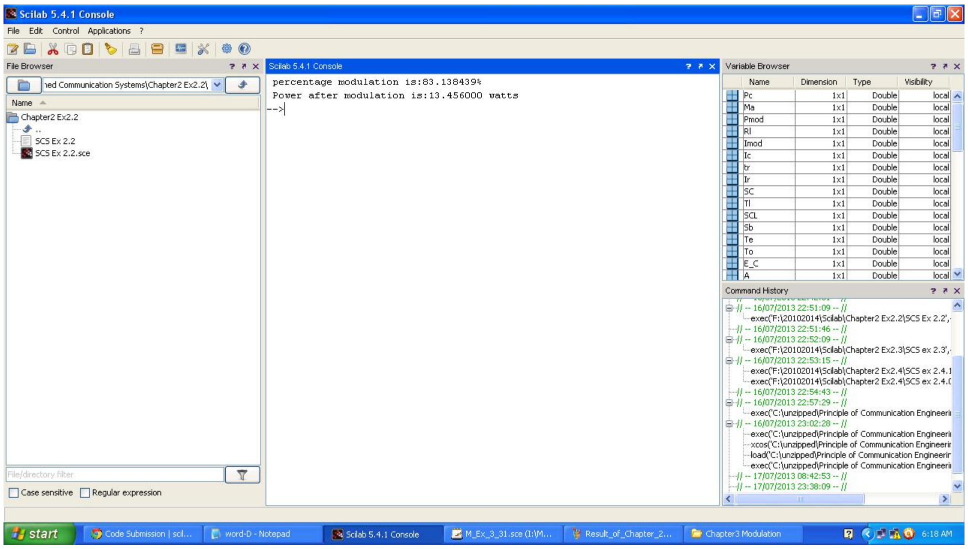

Figure 3.1: Modulation Ex 3 1

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

Scilab code Exa 3.3 Modulation Ex 3 3

```
clc
//Chapter3: Modulation
//Example3.3
//Given
M1=0.2//depth of modulation for first tone
```

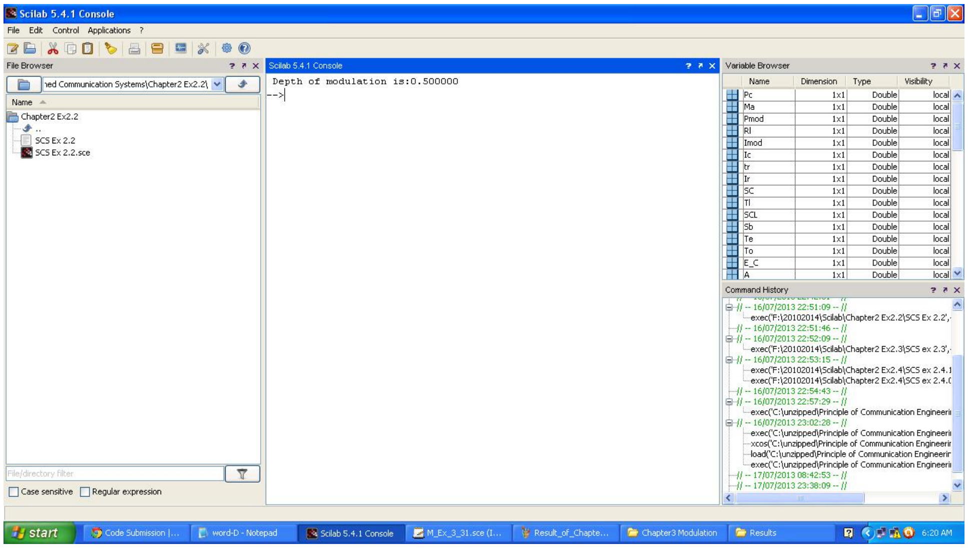

Figure 3.2: Modulation Ex 3 2

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

Scilab code Exa 3.4 Modulation Ex 3 4

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

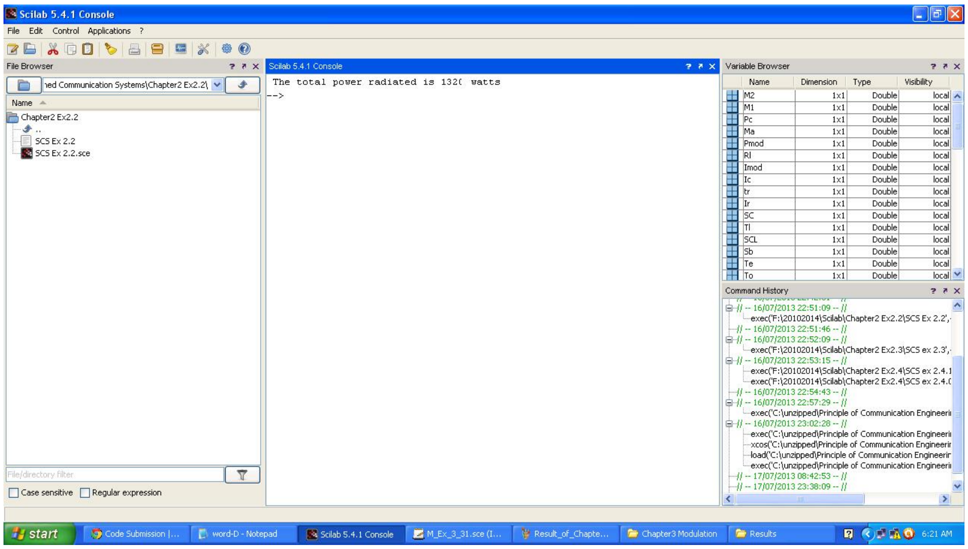

Figure 3.3: Modulation Ex 3 3

```
Egm=750//RF peak grid voltage
Pout=75//RF Power output
Ma=0.75//Depth of modulation
Paf=(Ma^2*Ebb*Ib)/(2*1)//modulating power required
    from the audio source
Pdc=Ebb*Ib//Power supplied by DC source
Zm=Ebb^2/Pdc//Modulator Impedance

Pd=Pdc+Paf-Pout//Plate dissipation
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

```
clc
```

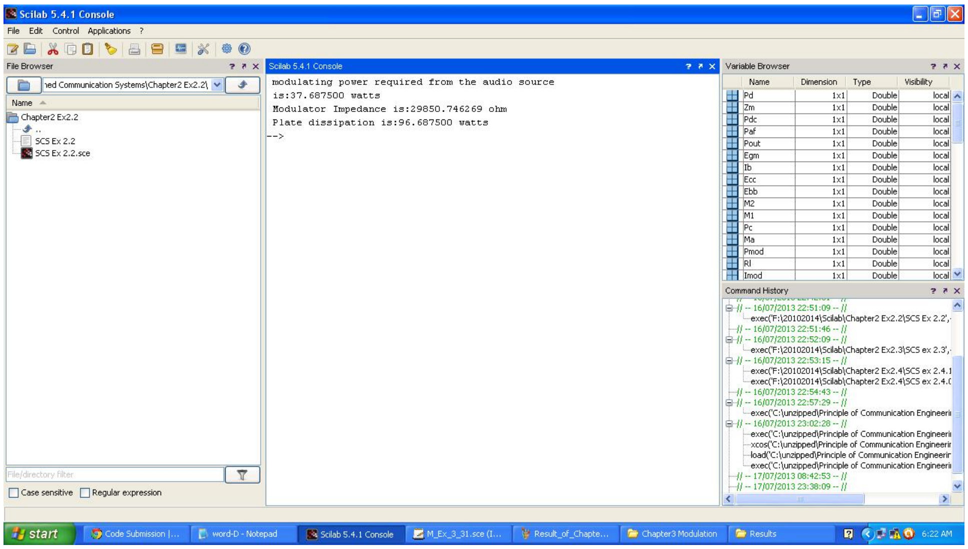

Figure 3.4: Modulation Ex 3 4

```
//Chapter3: Modulation
//Example3.5b, page no 139
//Given
Pd=944//Anode dissipation of the class C amplifier in watts
Ma=0.6//modulation depth,
Etta=0.6//efficiency
Pout=(Etta*Pd/(1-Etta))//power dissipation at 60% modulation
Pc=Pout/(1+(Ma^2/2))//Tx power
Psb=Pout-Pc
Pdc1=Pc/Etta//DC power inputto PA
Paf=Psb/Etta// modulation power input to PA
Eff=0.25// efficiency of the modulator
Pdc2=Paf/Eff//DC power input to modulator
Pdct=Pdc1+Pdc2//Total DC power to the system
Effo=Pout/Pdct//Overall Efficiency
Ma=1// 100% modulation
Pt=Pc*(1+(Ma^2)/2)
Psb=(Pc*Ma^2)/2
```

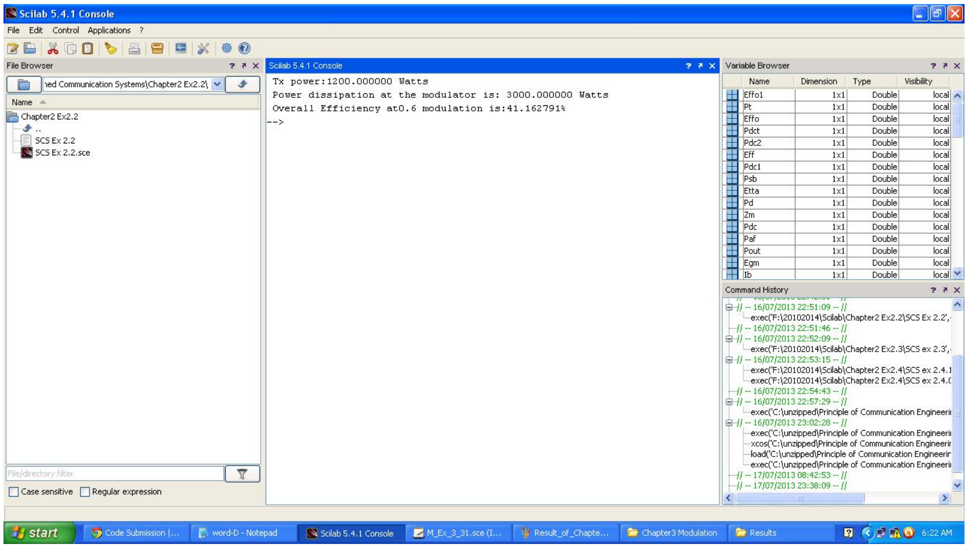

Figure 3.5: Modulation Ex 3 5

```
Paf=Psb/Etta//modulating input power to PA
Pdc2=Paf/Eff// DC power input to modulator
Pd=Pdc2-Paf//Power dissipation at the modulator
Effo1=Pout/(Pdc1+Pdc2)//Overall Efficiency
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

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

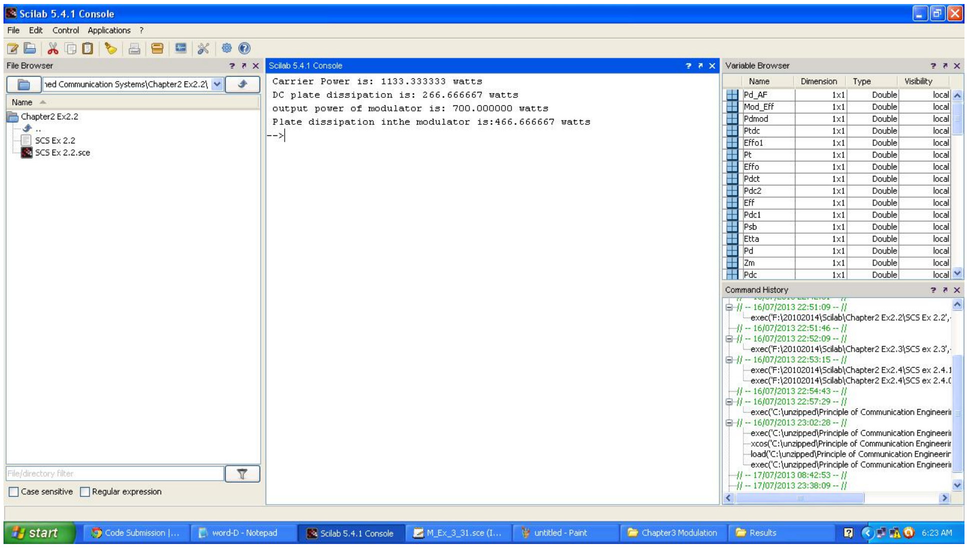

Figure 3.6: Modulation Ex 3 6

```
Pd=Ptdc*(2/3)//DC plate dissipation

Pdmod=Ptdc*(1/3)//
Pc=Pdc-Pd//Carrier Power

Psb=Pc/2//side band power
Paf=Psb+Pdmod//output power of modulator

Mod_Eff=0.6
Pdc2=Paf/Mod_Eff//DC i/p power to the modulator
Pd_AF=Pdc2-Paf//Plate dissipation inthe modulator
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

```
clc
//Chapter3: Modulation
//Example3.7, page no 141
//Given
Paf=500//Modulator output power
Eff=0.75//Efficiency of the amplifier
P_lost=Paf*(1-Eff)//modulating power lost in the
    amplifier
Psb=Paf*Eff//side band power

m=1
Pc=2*Psb

Pt=Pc+Psb//Total RF power
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

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

```

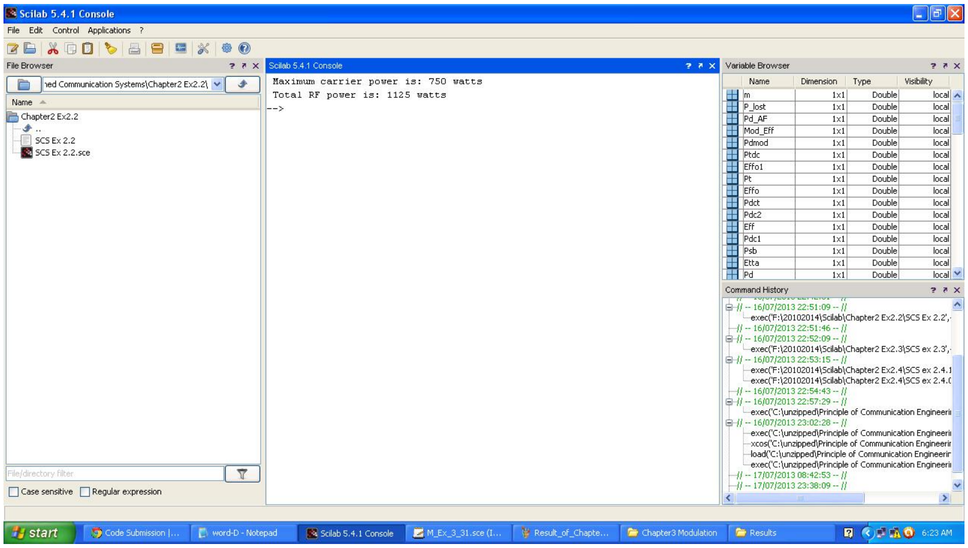

Figure 3.7: Modulation Ex 3 7

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

Scilab code Exa 3.9 `Modulation Ex 3 9`

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

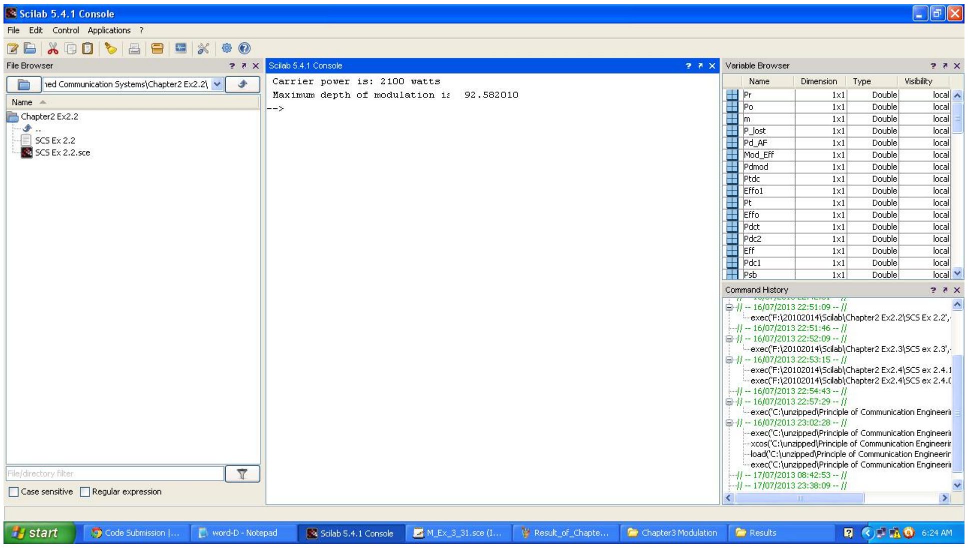

Figure 3.8: Modulation Ex 3 8

```
//b
wm=3140//Modulating angular freq
fm=wm/(2*%pi)//Modulating freq
//c
Ec=500///peak carrier voltage
Pc=(Ec^2)/(2*600)//Carrier power
//d
Ma=0.4
Pt=Pc*(1+(Ma^2 / 2))//Mean output power
//e
Rl=600//load resistance
Ecp=Ec+(Ma*Ec)//Peak output voltage
Ptm=Ecp^2/(2*Rl)//Peak power
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)
```

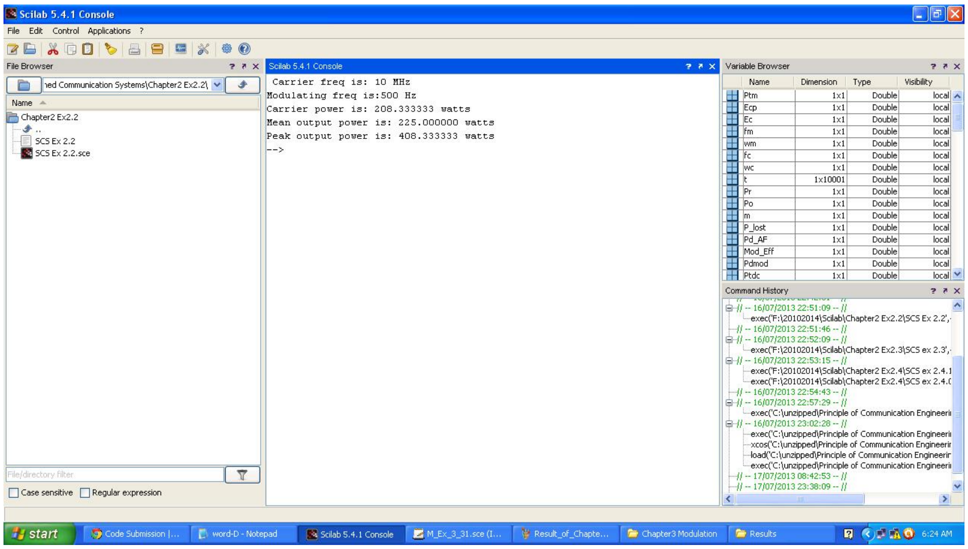

Figure 3.9: Modulation Ex 3 9

Scilab code Exa 3.10 `Modulation Ex 3 10`

```

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

```
R=36//resistance of the antenna
Irms=sqrt(Pmod/R)//Antenna Current

//ii
Ic=sqrt(Pc/R)//RMS carrier current

Icm=Ic*sqrt(2)// Peak carrier current
Imod=2*Icm//Modulated current

Theta=atan(40/36)*180/%pi// from real and imaginary
    components of Z
Vbm100=Imod*Z//Peak base output voltage for 100%
    modulation
[Re_Vb,Im_Vb]=polar(Vbm100)

//iii
Ma=0.5
Imod=Icm*(1+0.5)

Vbm50=Imod*Z
[Re_Vb1,Im_Vb1]=polar(Vbm50)
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)
// 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

```
clf();
clc
```

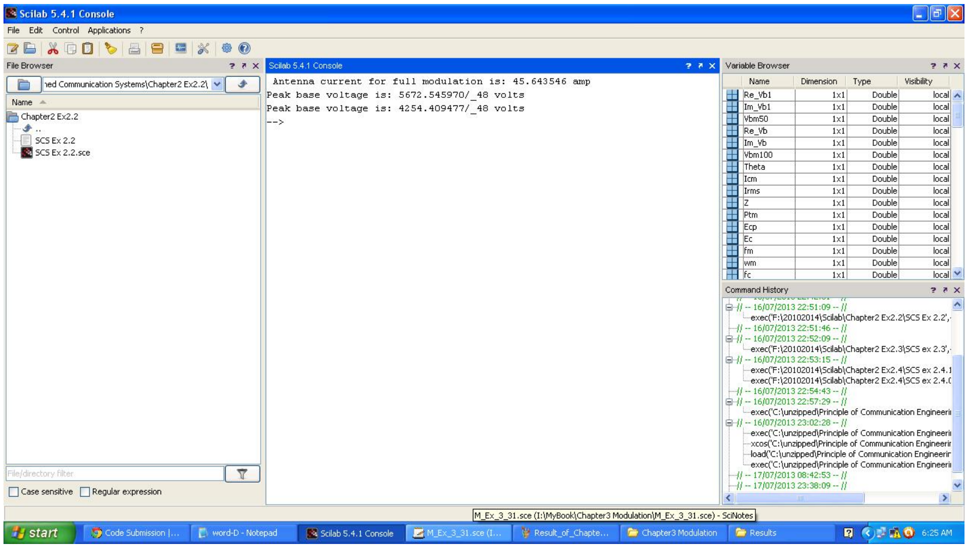

Figure 3.10: Modulation Ex 3 10

```
//Chapter3: Modulation
//Example3.11, page no 144
//Given
deff("[y]=f(x)","y=Ec*(1+ma*(sin(wm*x)))*sin(wc*x)")
Ec=10,ma=0.5,wm=10000*%pi,wc=2*%pi*1e7
x=[0:0.01:20]*%pi/10;
subplot(2,1,1)
fplot2d(x,f)
xlabel("t", "fontsize", 3);
ylabel("Modulated Wave", "fontsize", 3, "color", "red");
Fc=wc/(2*%pi)
Fm=wm/(2*%pi)
Fusb=(wm+wc)/(2*%pi)
Flsb=(wm-wc)/(2*%pi)
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)
F=[0,2.5,10,2.5,0]
T=[-2,-1,0,1,2]
```

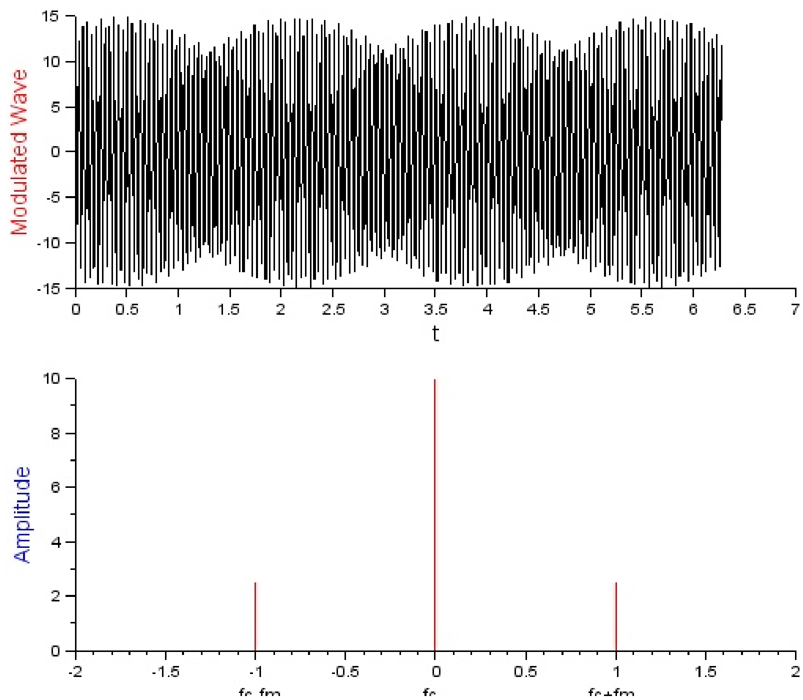

Figure 3.11: Modulation Ex 3 11

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

Scilab code Exa 3.12 `Modulation Ex 3 12`

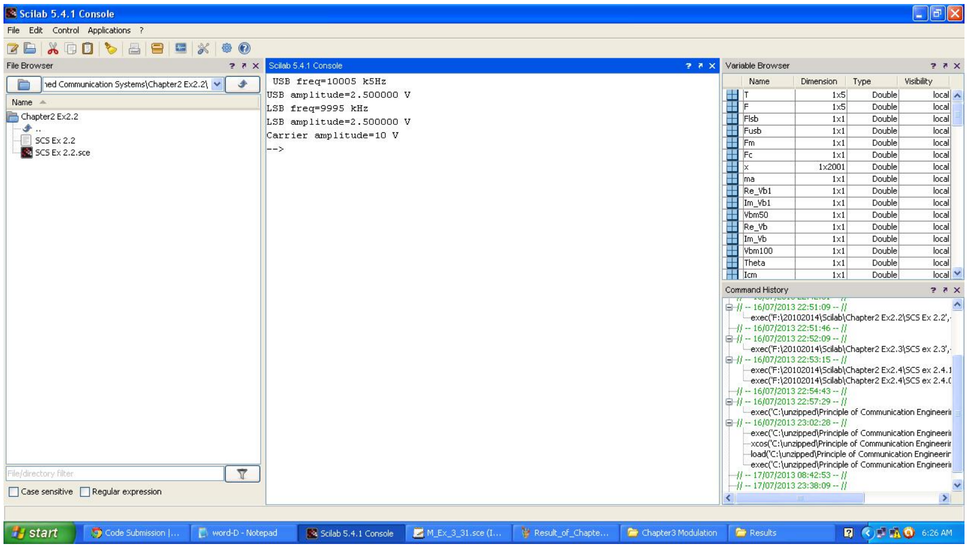

Figure 3.12: Modulation Ex 3 11

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

Scilab code Exa 3.13 `Modulation Ex 3 13`

```
clc
//Chapter3: Modulation
//Example3.13 page no 148
//Given
```

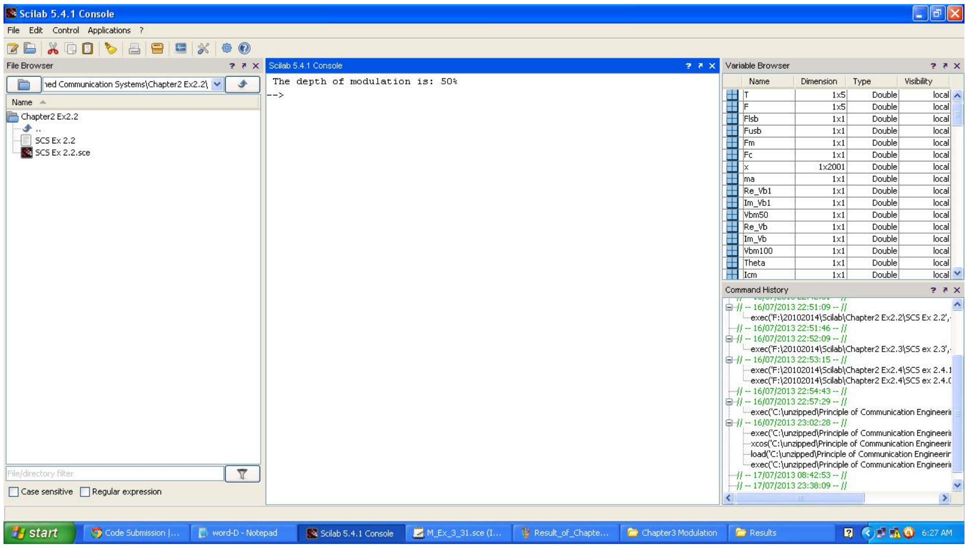

Figure 3.13: Modulation Ex 3 12

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

Scilab code Exa 3.14 Modulation Ex 3 14

```
clc
//Chapter3: Modulation
//Example3.14 page no 152
//Given
```

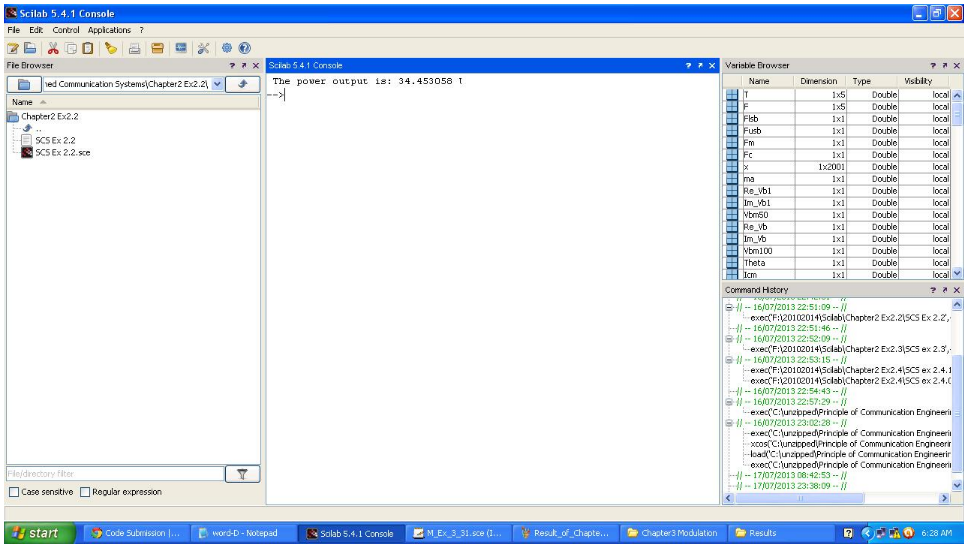

Figure 3.14: Modulation Ex 3 13

```
//Phi=(wc*t+Mf*sin(wmt))....instantaneous phase of
   FM
fm=5000//modulating freq
deltaf=50e3//freq deviation
deltaPhi1=deltaf/fm// Advance or retard in phase

fm=100//modulating freq in second signal
deltaPhi2=deltaf/fm
mprintf('DeltaPhi1= %d rad\nDeltaPhi2=%d rad\n',
    deltaPhi1,deltaPhi2)
```

Scilab code Exa 3.15 Modulation Ex 3 15

```
clc
//Chapter3: Modulation
//Example3.14 page no 157
```

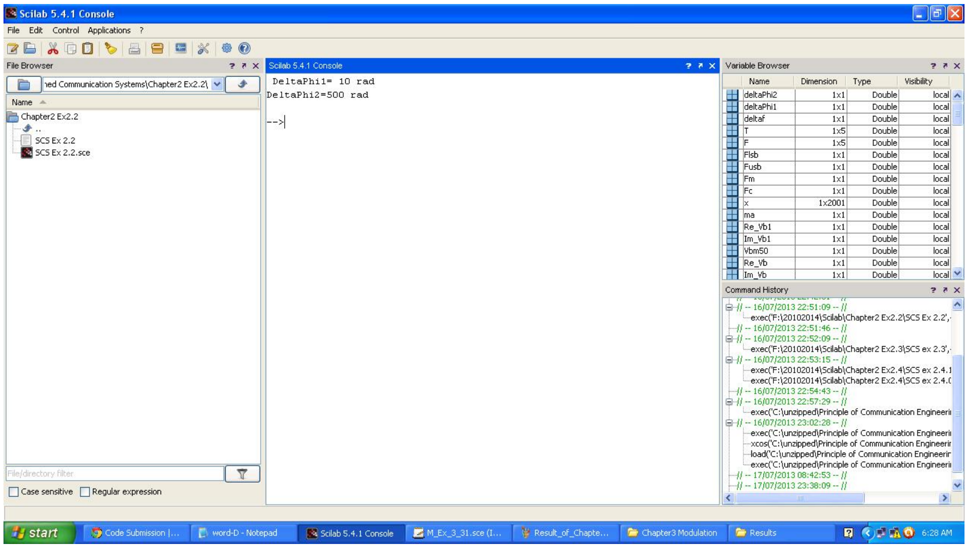

Figure 3.15: Modulation Ex 3 14

```
//Given
//e=Ec(1+0.4cos(2pie3*t))*sin(2pie7*t)
fm=1000//modulating s/g freq
deltaTheta=2*atan(0.4)//peak phase deviation

deltaF=deltaTheta*fm//Peak freq deviation

Ec=1
Er=sqrt((Ec^2)*(1+(0.4^2)))
m=(Er-Ec)/Ec//depth of residual AM

AMFr=2*fm// freq ofresidual AM
mprintf('Peak Phase Deviation: %f rad\nPeak Freq
    Deviation: %d Hz\nDepth of residual AM: %f\
    nResidual AM freq:%d kHz',deltaTheta,deltaF,(
    round(m*100)/100),AMFr*1e-3)
```

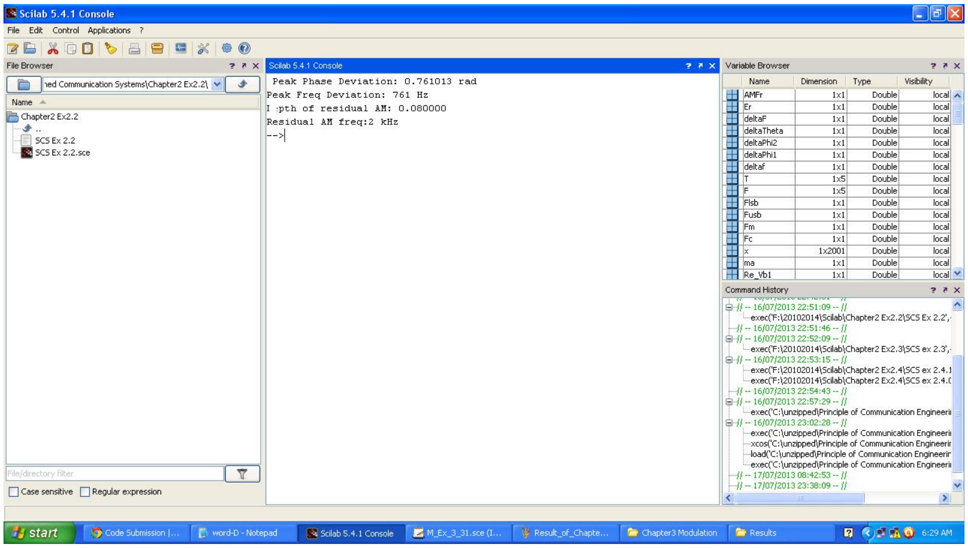

Figure 3.16: Modulation Ex 3 15

Scilab code Exa 3.16 `Modulation Ex 3 16`

```
clc
//Chapter3: Modulation
//Example3.16 page no 170
//Given
deltaF=25e3//freq deviation
//a
fm=100//modulation signal freq
mf=deltaF/fm// Max phase deviation
disp('a)')
mprintf('Max phase deviation is:%d rad',mf)
//b
fm=10e3//modulation signal freq
mf=deltaF/fm//Max phase deviation

disp('b)')
```

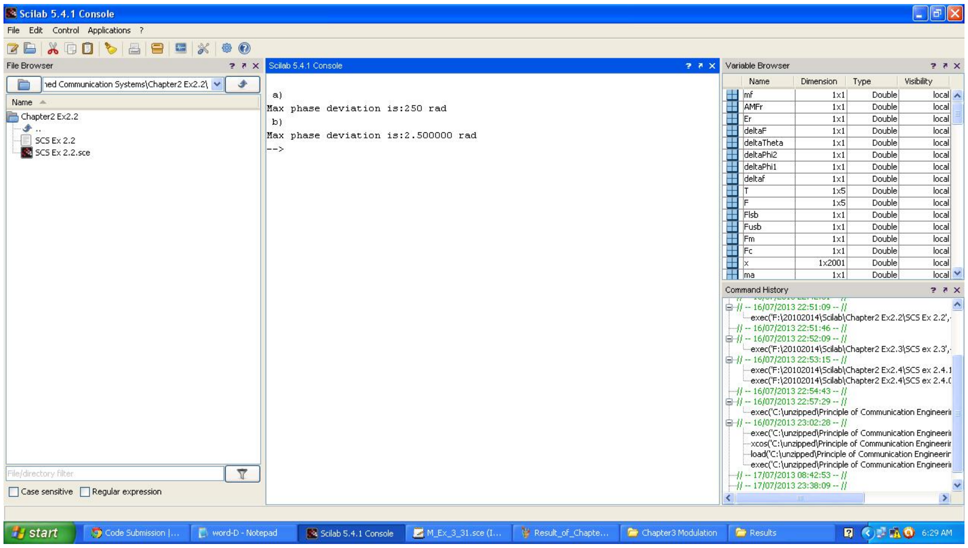

Figure 3.17: Modulation Ex 3 16

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

Scilab code Exa 3.17 `Modulation Ex 3 17`

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

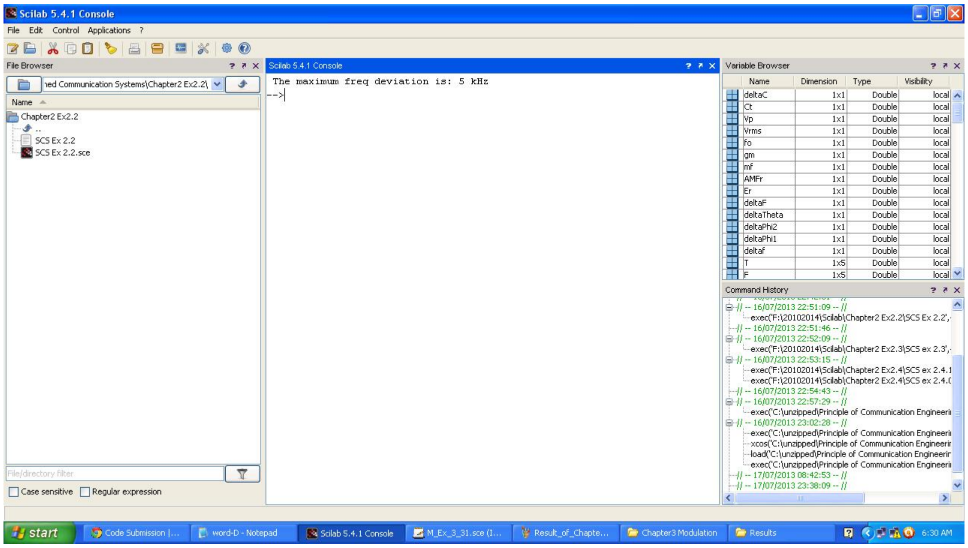

Figure 3.18: Modulation Ex 3 17

```

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

Scilab code Exa 3.18 Modulation Ex 3 18

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

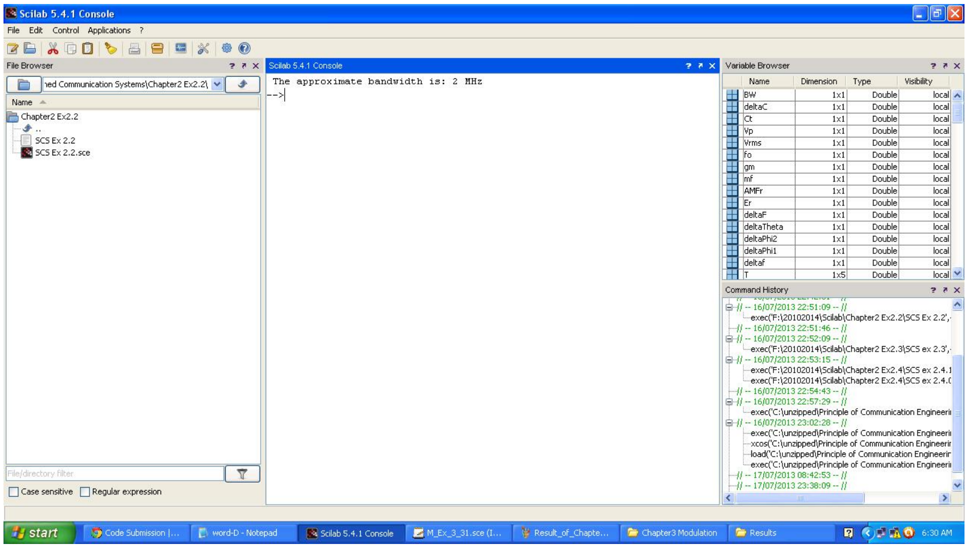

Figure 3.19: Modulation Ex 3 18

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

Scilab code Exa 3.19 `Modulation Ex 3 19`

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

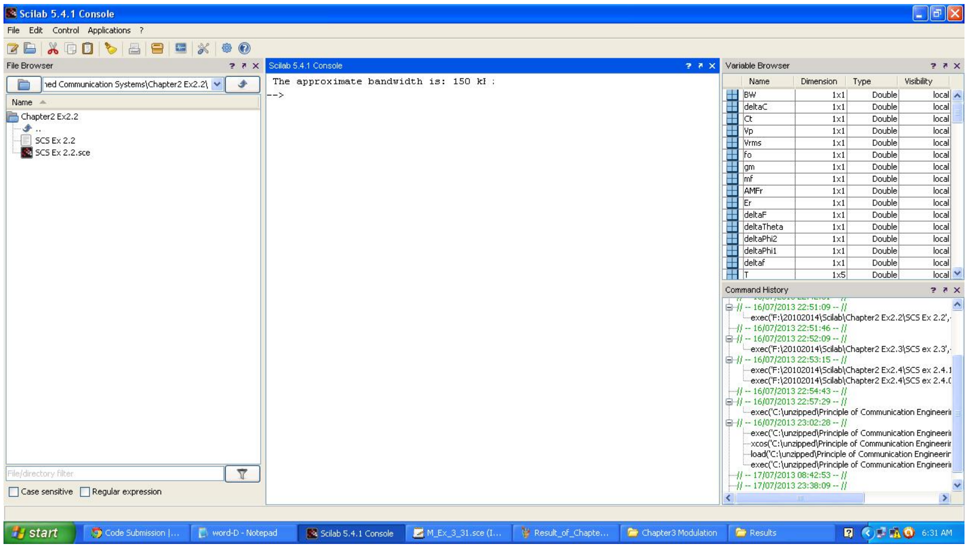

Figure 3.20: Modulation Ex 3 19

Scilab code Exa 3.21 Modulation Ex 3 21

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

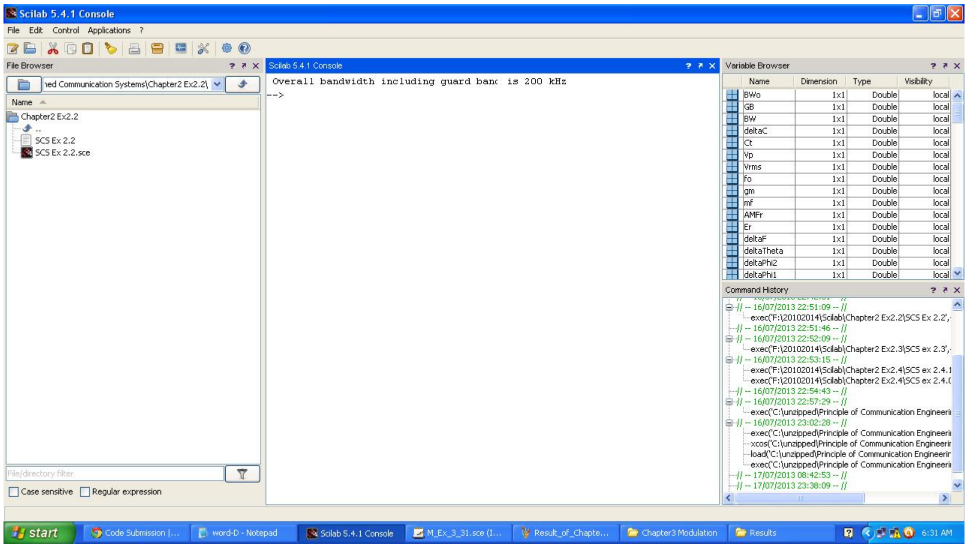

Figure 3.21: Modulation Ex 3 21

Scilab code Exa 3.25 Modulation Ex 3 25

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

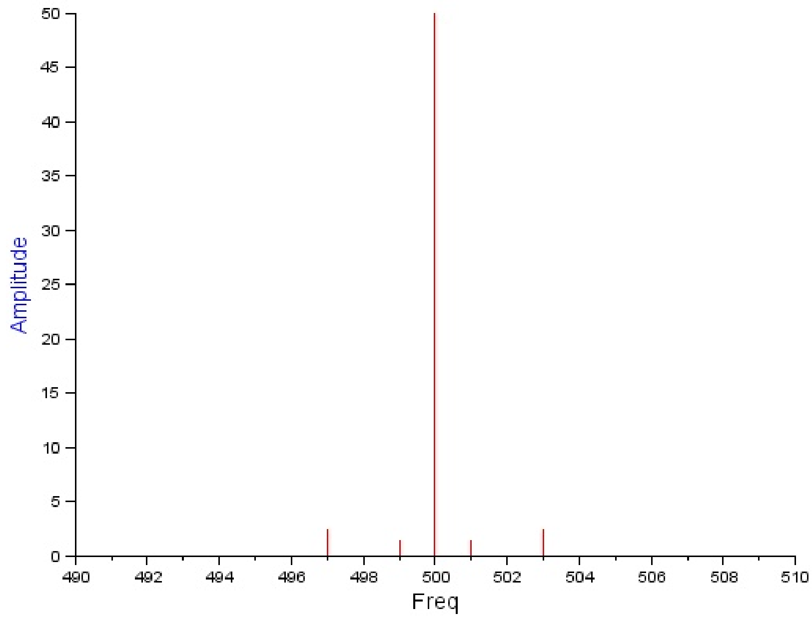

Figure 3.22: Modulation Ex 3 25

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

Scilab code Exa 3.26 `Modulation Ex 3 26`

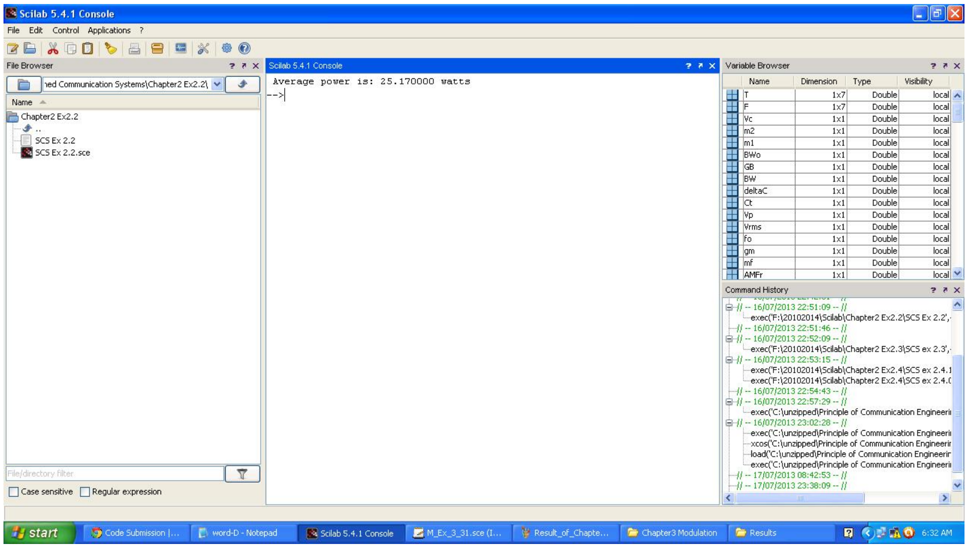

Figure 3.23: Modulation Ex 3 25

```
clc
//Chapter3: Modulation
//Example3.26, page no 176
//Given
mp=0.1//Modulating index
fm=400//Modulating signal freq
deltaF=mp*fm//Max freq deviation
//disp(deltaF)
ReqDev=50e3// Required deviation
MF=ReqDev/deltaF// multiplication factor
mprintf('Required Deviation is: %d kHz\n',ReqDev/1e3
    )
mprintf('Required Multipication Factor is: 5*5*5*5*2
    ')
```

Scilab code Exa 3.27 `Modulation Ex 3 27`

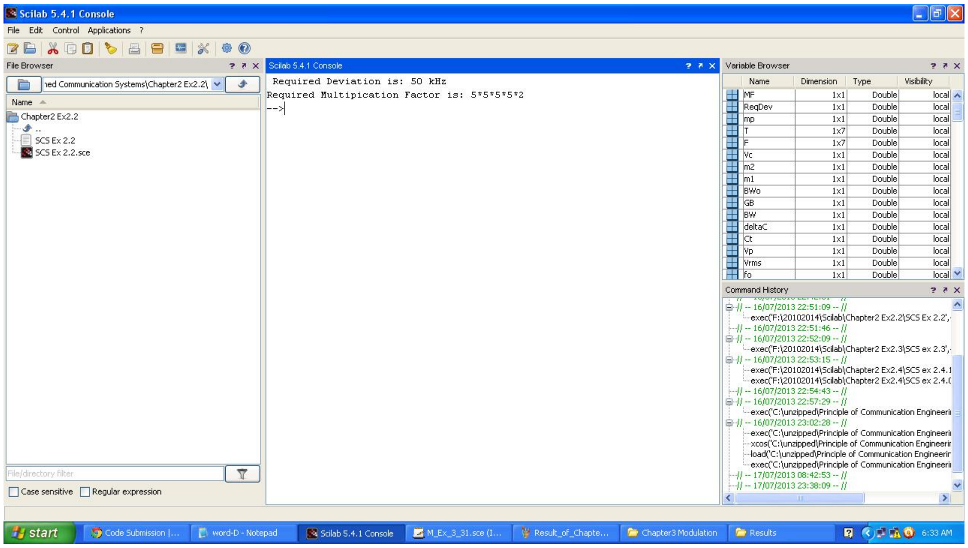

Figure 3.24: Modulation Ex 3 26

```
clc
//Chapter3: Modulation
//Example3.27, page no 176
//Given
Q=100 //Q factor
 fc=1000e3// Carrier freq
fsb1=999e3//lower Side band freq
fsb2=1001e3//Upper side Band freq
ma=0.5//Modulation depth of signal current
Ma=ma/1.019// Expression for Ma after simplification
mprintf('The Depth of modulation across the \n
   circuit is : Ma= %f%c',Ma*100,'%')

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

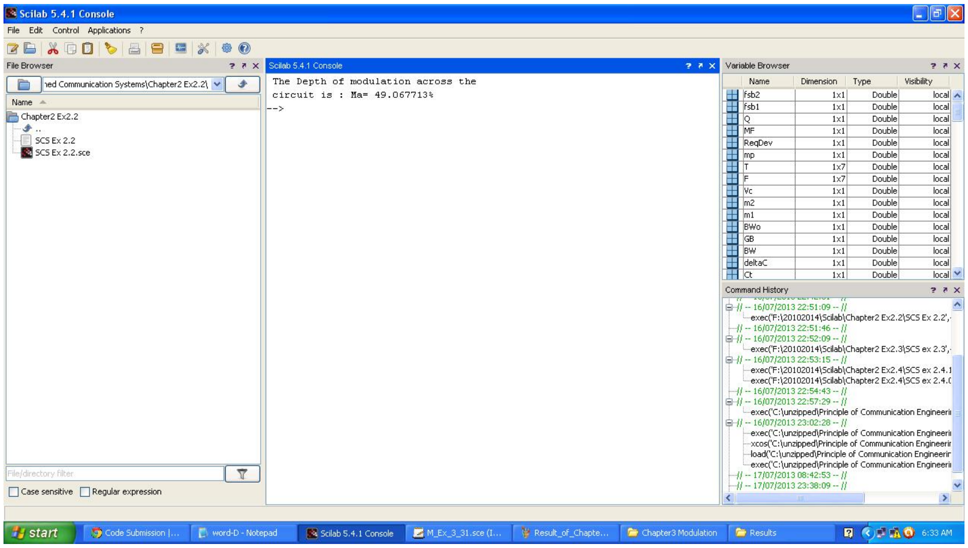

Figure 3.25: Modulation Ex 3 27

Scilab code Exa 3.28 `Modulation Ex 3 28`

```
clc
//Chapter3: Modulation
//Example3.28, page no 177
//Given
R=1//Antenna Resistance assumed to be 1 ohm for ease of calculation
Ic=10.8// current with no modulation
Pc=Ic^2*R//power with no modulation
It=12.15//modulated current
Pt=It^2*R// modulated power
ma=(sqrt(2*(((It/Ic)^2)-1)))//modulation depth)

mprintf('Depth of modulation:%f %c',round(1000*ma)/10,'%');
```

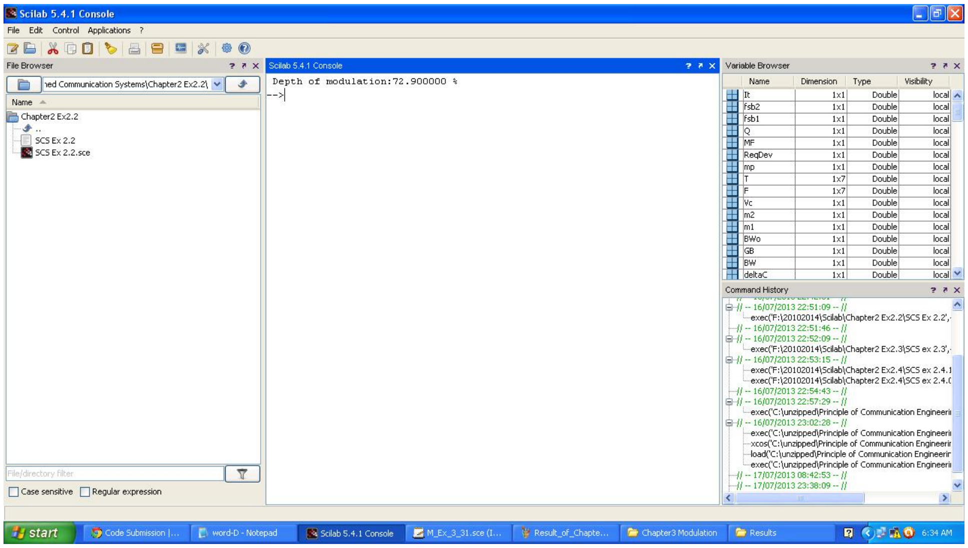

Figure 3.26: Modulation Ex 3 28

Scilab code Exa 3.29 Modulation Ex 3 29

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

Scilab code Exa 3.30 Modulation Ex 3 30

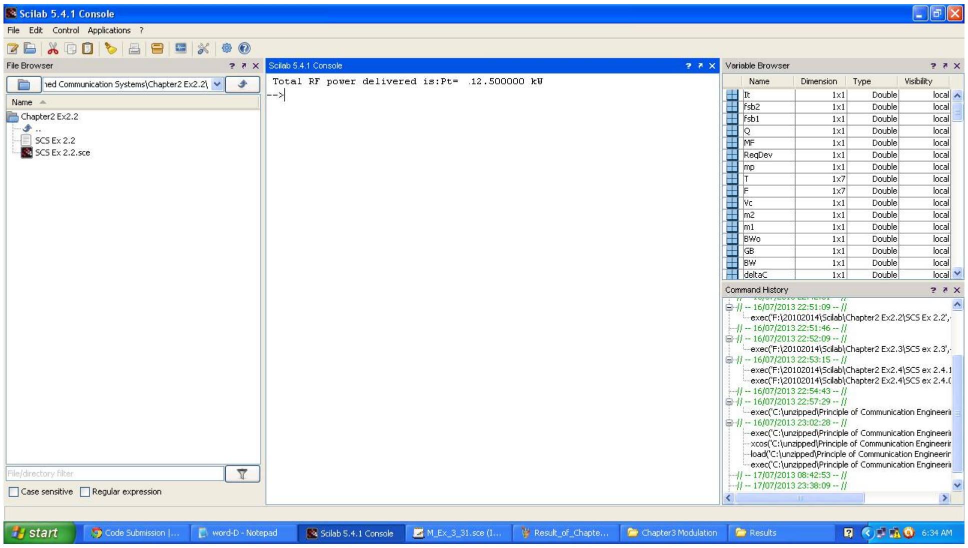

Figure 3.27: Modulation Ex 3 29

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

Scilab code Exa 3.31 `Modulation Ex 3 31`

```
clc
//Chapter3: Modulation
//Example3.19, page no 178
```

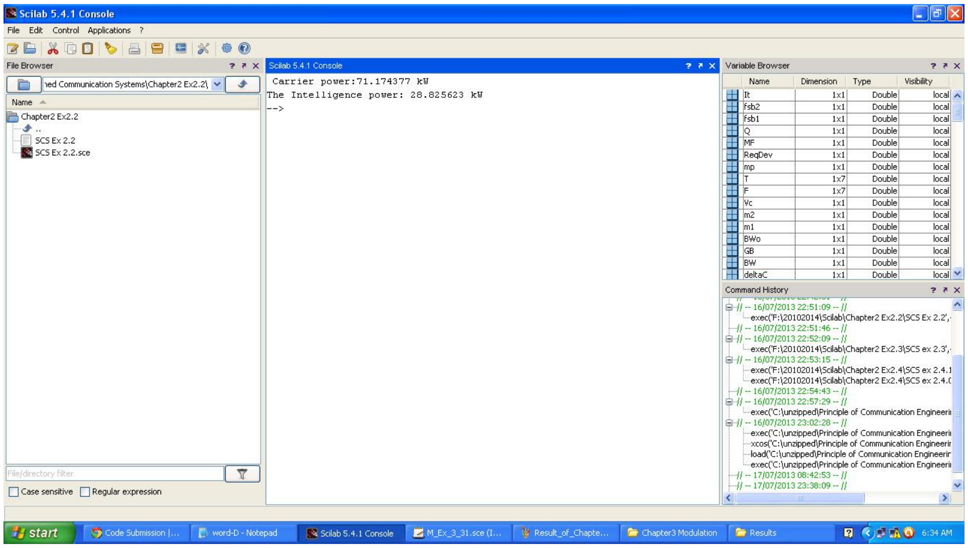

Figure 3.28: Modulation Ex 3 30

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

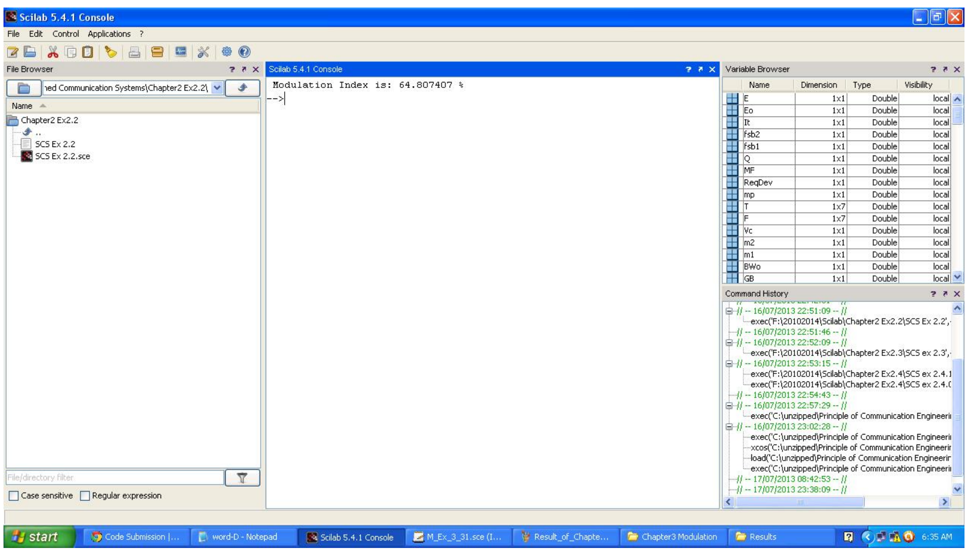

Figure 3.29: Modulation Ex 3 31

Chapter 5

Radio Transmission System

Scilab code Exa 5.1 RTS Ex 5 1

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

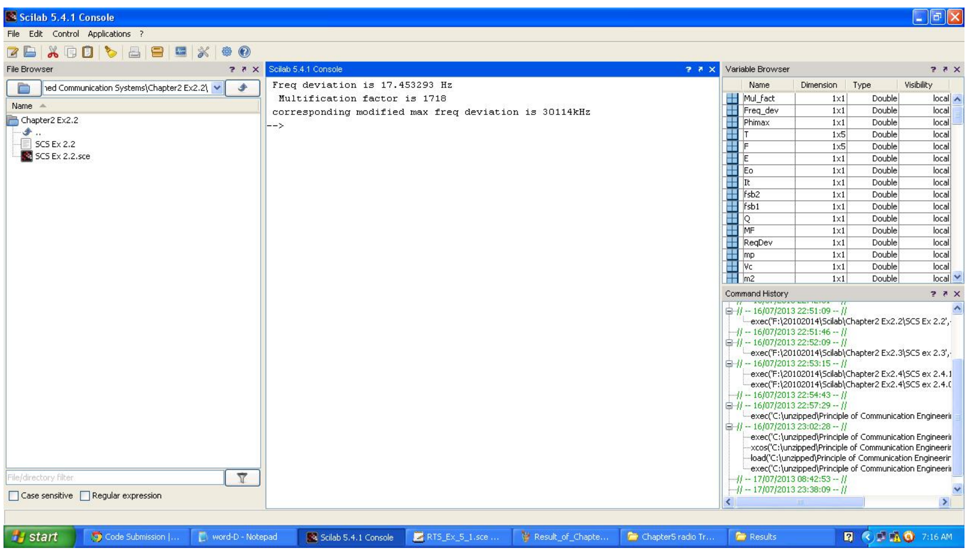

Figure 5.1: RTS Ex 5 1

Chapter 6

Radio Receivers

Scilab code Exa 6.1 Radio Receivers Ex 6 1

```
clc
//Chapter8
//Example8.15, page no 262
//Given
//Vm(t),Vc(t),Vmod(t)
fm=10e3//modulating freq
BW=2*fm// Bandwidth
fc=100*BW//Carrier freq
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

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

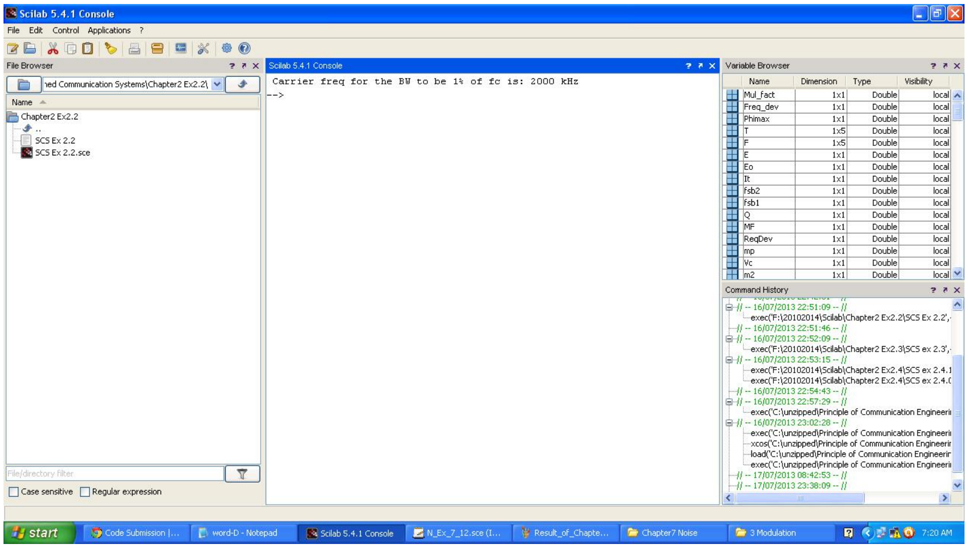

Figure 6.1: Radio Receivers Ex 6 1

```
fmax=1600e3,fmin=500e3,IF=465e3
//i
fo1max=fmax+IF,fo1min=fmin+IF
C1max_C1min=(fo1max/fo1min)^2
//ii
fo2max=fmax-IF,fo2min=fmin-IF
C2max_C2min=(fo2max/fo2min)^2
mprintf('a)\nTuning capacitor range is: %f\nb)\
    nTuning capacitor range is: %d',C1max_C1min,
    C2max_C2min)
```

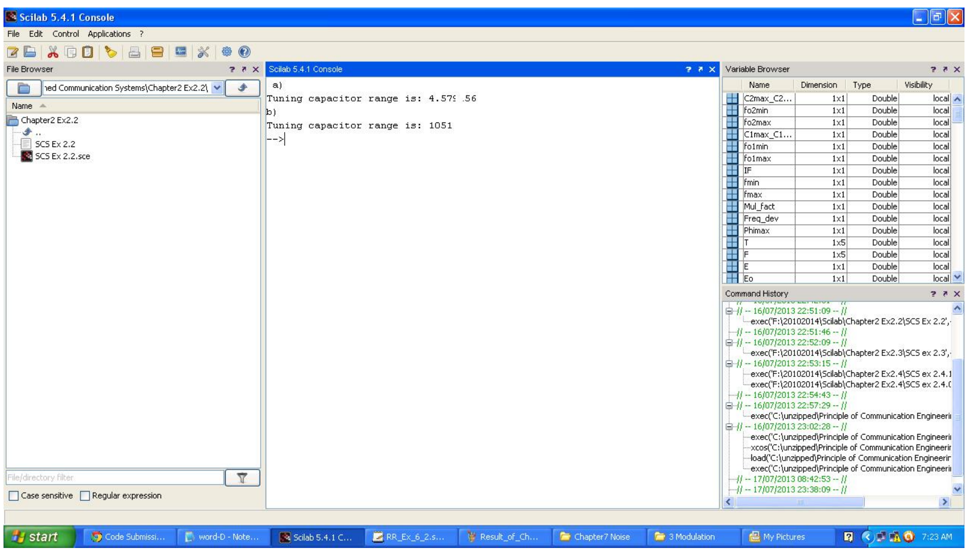

Figure 6.2: Radio Receivers Ex 6 2

Chapter 7

Noise

Scilab code Exa 7.2 Noise Ex 7 2

```
clc
//Chapter7
//Example7.2, page no 276
//Given
mue=25//
rp=5e3
Rl=10e3
C=1e-9
gm=mue/rp
Req=2.5/gm
//disp(Req , gm)
k=1.381e-23
T=293
R1=1e5
// Power density spectrum for respective res
d1=2*k*T*R1
d2=2*k*T*Req
d3=2*k*T*Rl
xo=0
x1=1e14
w=0:%inf
```

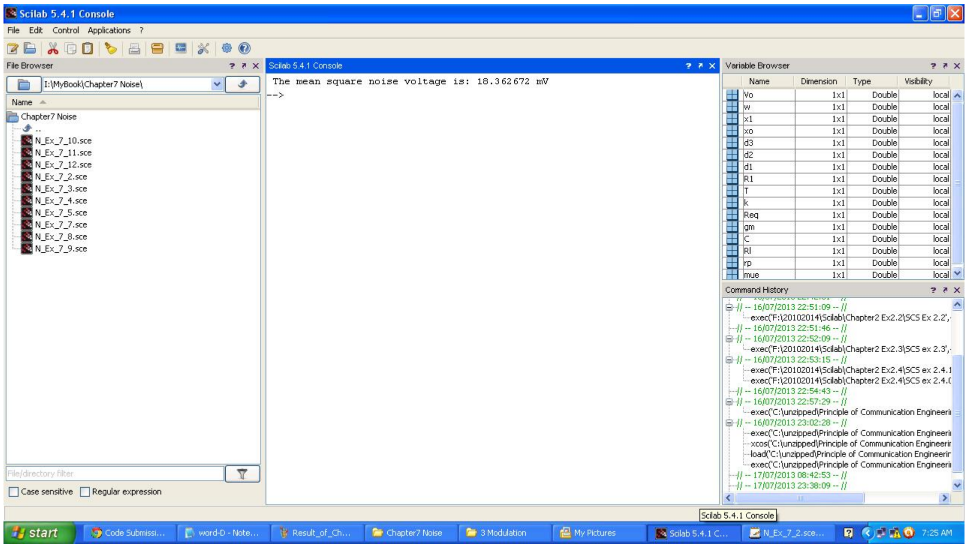

Figure 7.1: Noise Ex 7 2

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

Scilab code Exa 7.3 Noise Ex 7 3

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

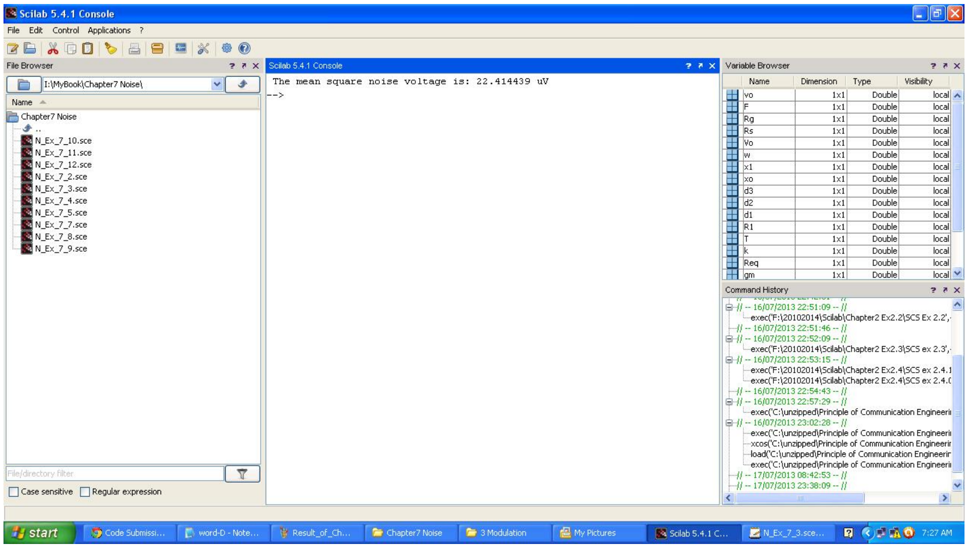

Figure 7.2: Noise Ex 7 3

```
Rg=1e5
gm=25/5e3
Req=2.5/gm
F=1+((((Req*(Rs+Rg)^2)+(Rg*Rs^2))/(Rs*Rg^2)))
xo=0
x1=1e10
w=0:%inf

vo=sqrt((30145e-8/%pi)*integrate('1/(((3e5)^2)+(w^2))','w',xo,x1))
mprintf('The mean square noise voltage is: %f uV',vo*1e6)
```

Scilab code Exa 7.4 Noise Ex 7 4

```
clc
```

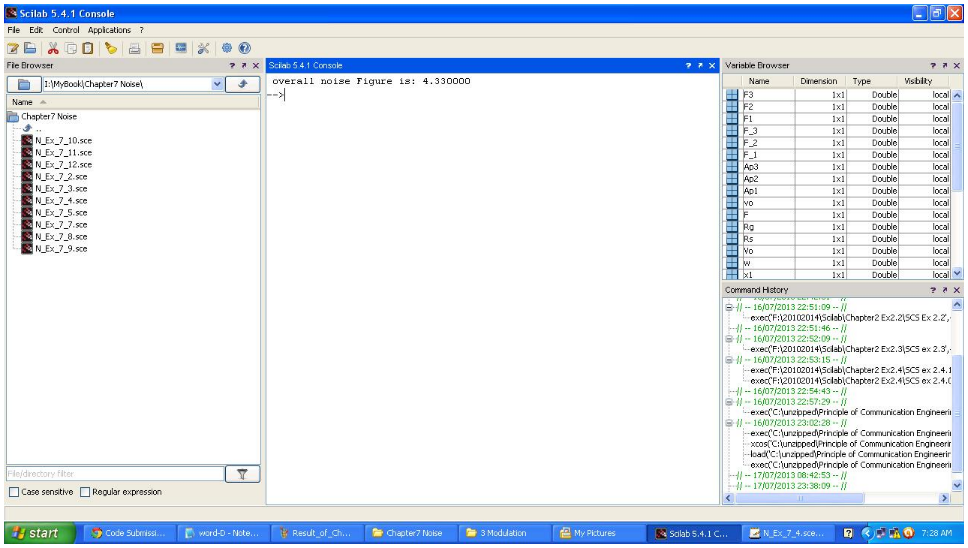

Figure 7.3: Noise Ex 7 4

```
//Chapter8
//Example7.4, page no 283
//Given
Ap1=10, Ap2=10, Ap3=10;  // Gain of each states
F_1=6, F_2=6, F_3=6;  //Noise figure of each state
F1= round(10^(F_1/10)), F2= round(10^(F_2/10)), F3=
    round(10^(F_3/10));  // approximating the values

F=F1+((F2-1)/Ap1)+((F3-1)/(Ap1*Ap2))
mprintf('overall noise Figure is: %f',F)
```

Scilab code Exa 7.5 Noise Ex 7 5

```
clc
//Chapter7
//Example7.5, page no 283
```

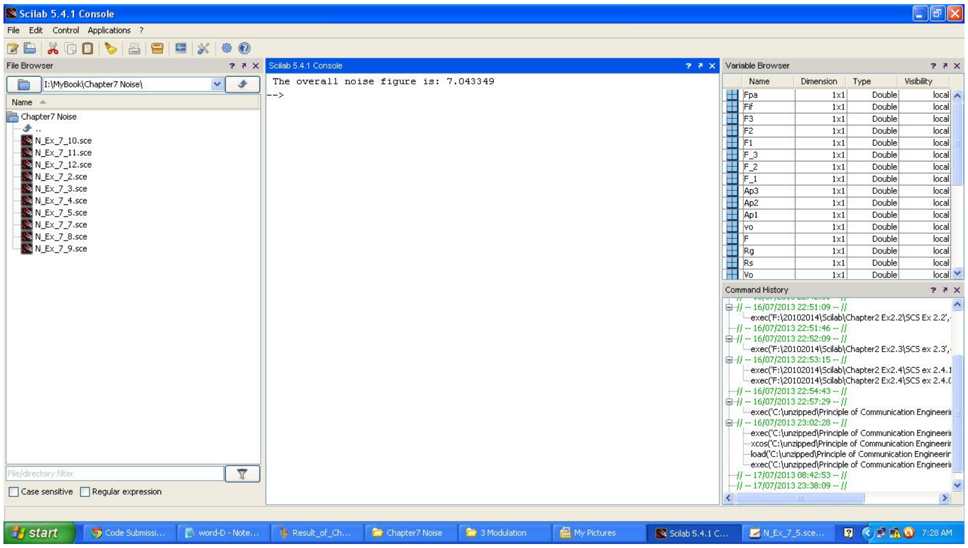

Figure 7.4: Noise Ex 7 5

```
//Given
Fif=15// Noise figure of IF amplifier
Ap1=10// Gain of Preamplifier
Fpa=6//Noise figure of preamplifier
F2=10^(Fif/10)
F1=10^(Fpa/10)

F=F1+((F2-1)/Ap1)//overall noise figure
mprintf('The overall noise figure is: %f',F)
```

Scilab code Exa 7.7 Noise Ex 7 7

```
clc
//Chapter7
//Example7.6
//Given
```

```
mue=25// tube parameters
rp=10e3// tube parameters
gm=2.5e-3// transconductance
Req=2.5/gm// equivalent resistance
Rs=1000
Rg=1e5
F1=1+(((Req*((Rs+Rg)^2))+Rg*Rs^2)/(Rs*(Rg^2)))//
    noise figure of the first stage
Rg2=9.1e3
Rs2=10e3
Es=1// assuming Es=1 for ease of calculation
Pi=((Es/2e3)^2)*1e3
Po=1.532e-2*Es^2
Ap1=Po/Pi
F2=1+(((Req*((Rs2+Rg2)^2))+Rg2*Rs2^2)/(Rs2*(Rg2^2)))
    // noise figure of the second stage
F=(F1)+((F2-1)/Ap1)
mprintf('Overall Noise figure is:%f',F)
```

Scilab code Exa 7.8 Noise Ex 7 8

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

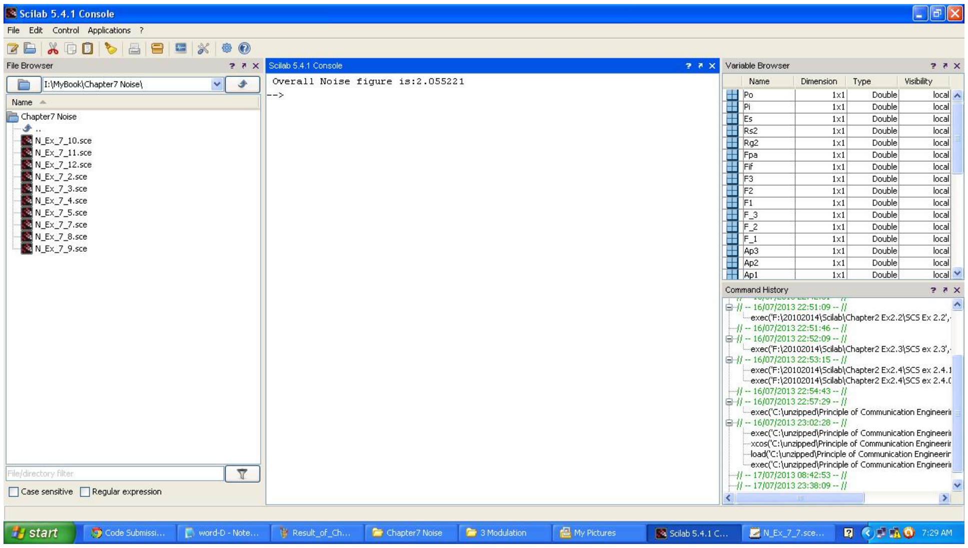

Figure 7.5: Noise Ex 7 7

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

Scilab code Exa 7.9 Noise Ex 7 9

```
clc
//Chapter7
//Example7.9
//Given
```

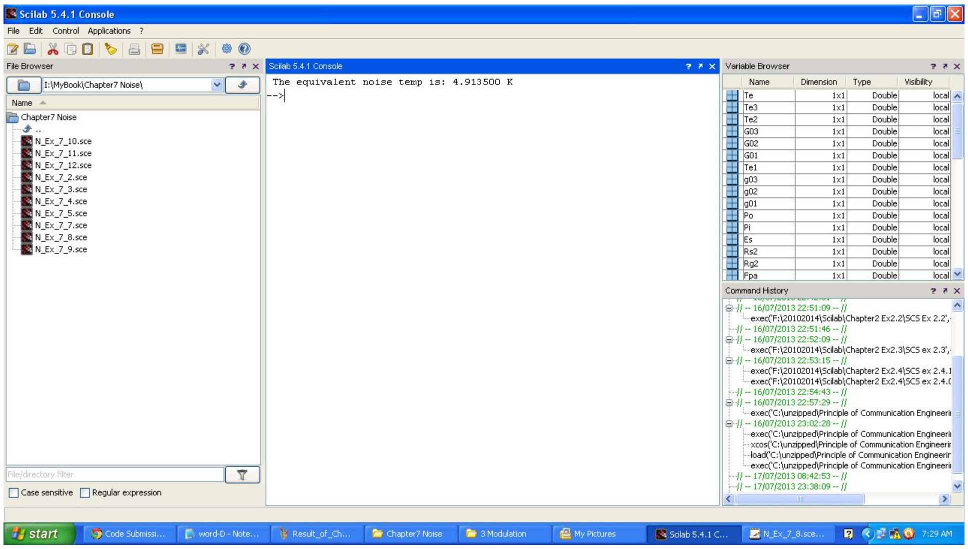

Figure 7.6: Noise Ex 7 8

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

Scilab code Exa 7.10 Noise Ex 7 10

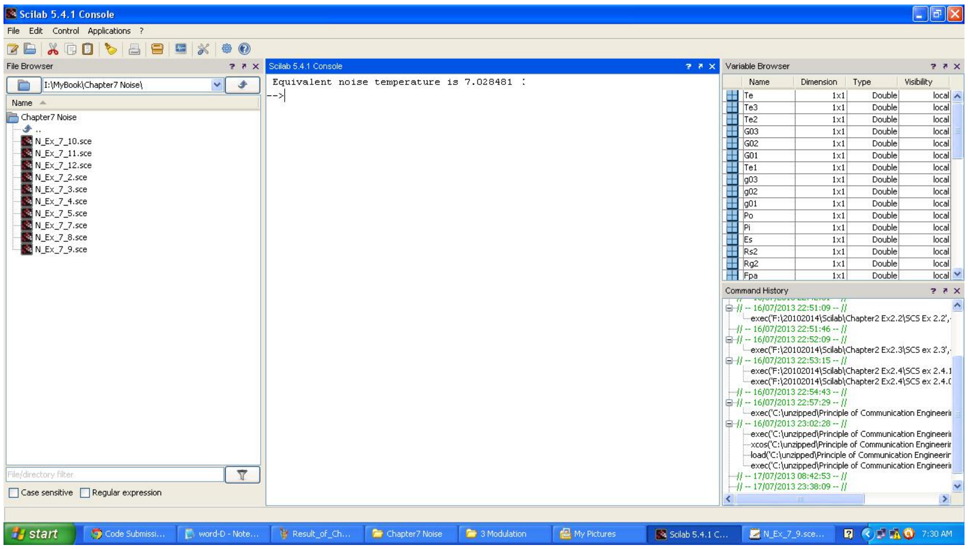

Figure 7.7: Noise Ex 7 9

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

Scilab code Exa 7.11 `Noise Ex 7 11`

```
clc
//Chapter7
```

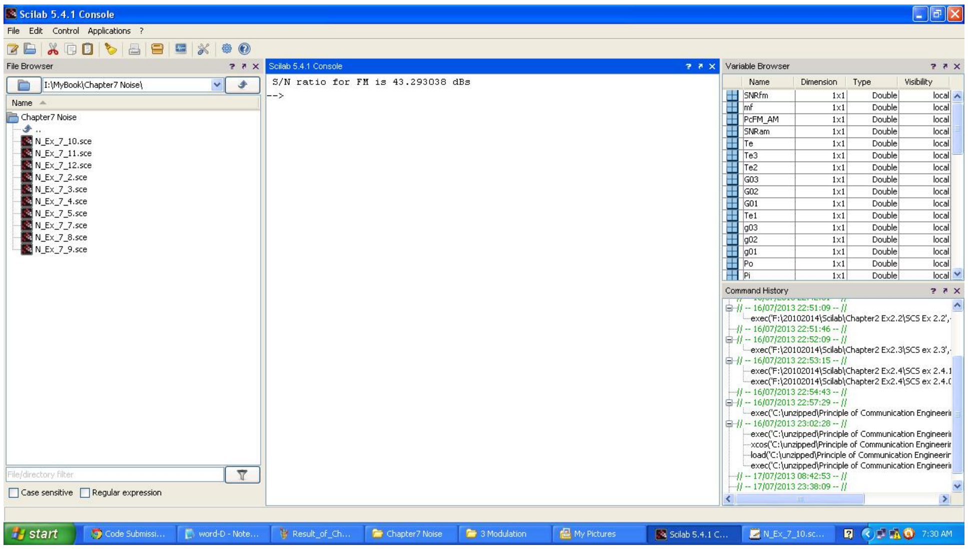

Figure 7.8: Noise Ex 7 10

```
//Example7.11
//Given
ma=0.3
SNR=20// s/n ratio
SNR1=10^(0.1*SNR)
SNR_new=SNR+3
ma2=0.6// increased new depth of modulation
Pt_Ni=SNR1*((1+(ma^2))/(ma^2))
SNR2=10*log10(Pt_Ni*((ma2^2)/(1+((ma2^2)/2))))

mprintf('a)\n  New SNR for 3dB increase in input s/g
    is %d dBs\nb)  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

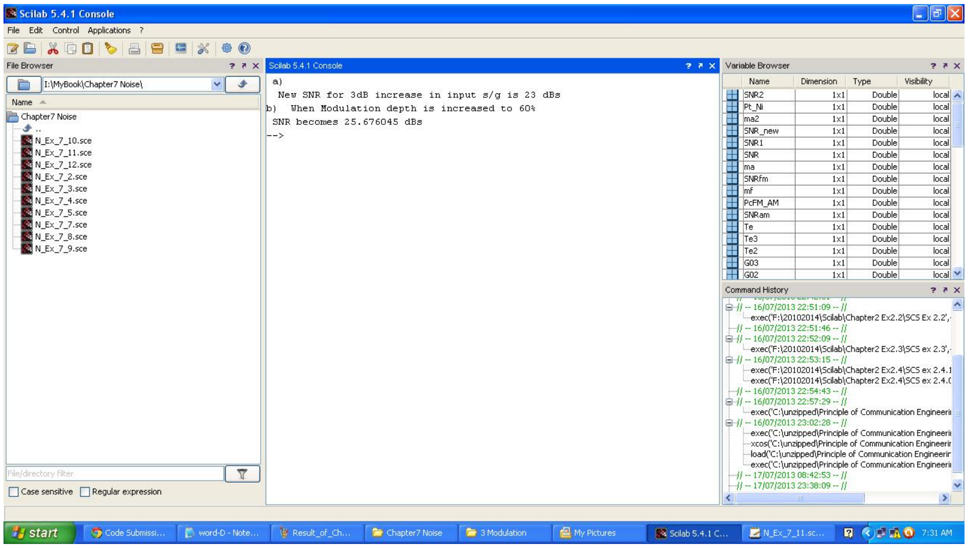

Figure 7.9: Noise Ex 7 11

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

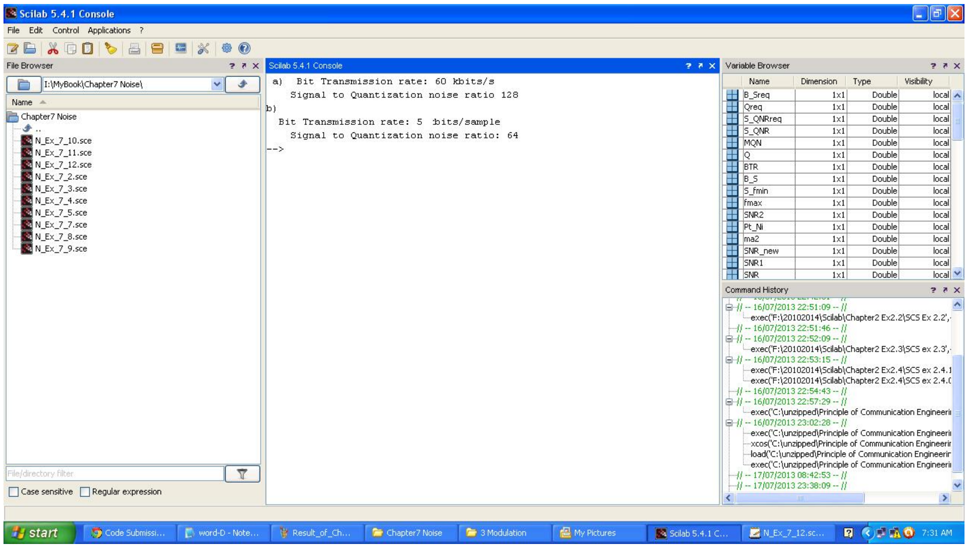

Figure 7.10: Noise Ex 7 12

```
    B_Sreq,S_QNRreq)
```

Chapter 8

Transmission Line

Scilab code Exa 8.1 Transmission Line Ex 8 1

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

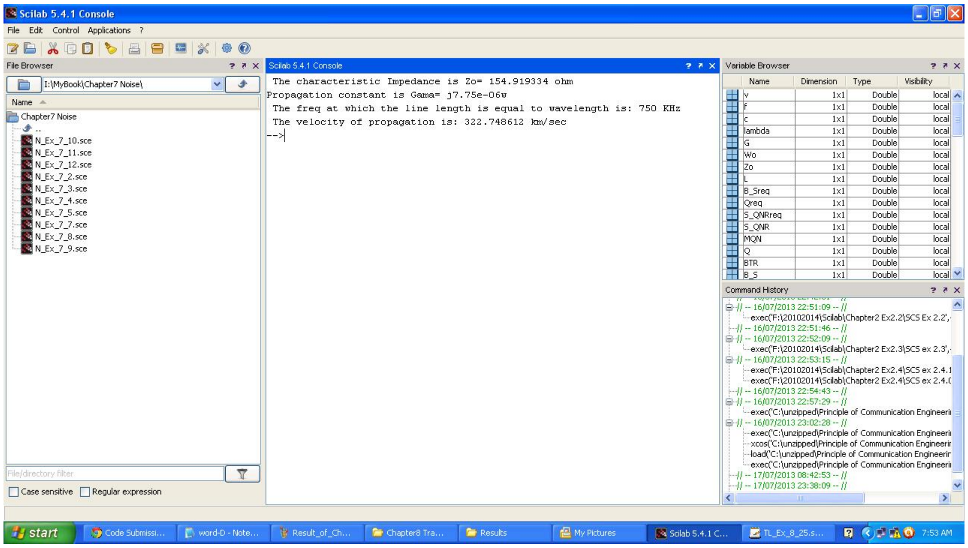

Figure 8.1: Transmission Line Ex 8 1

```
//ii
L=L'*0.4
C=C'*0.4
v=1/(sqrt(L*C))
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

```
clc
//Chapter8
//Example8.2, page no 314
//Given
v=3e8// velocty of light
f=1.2e6// Operating Freq
```

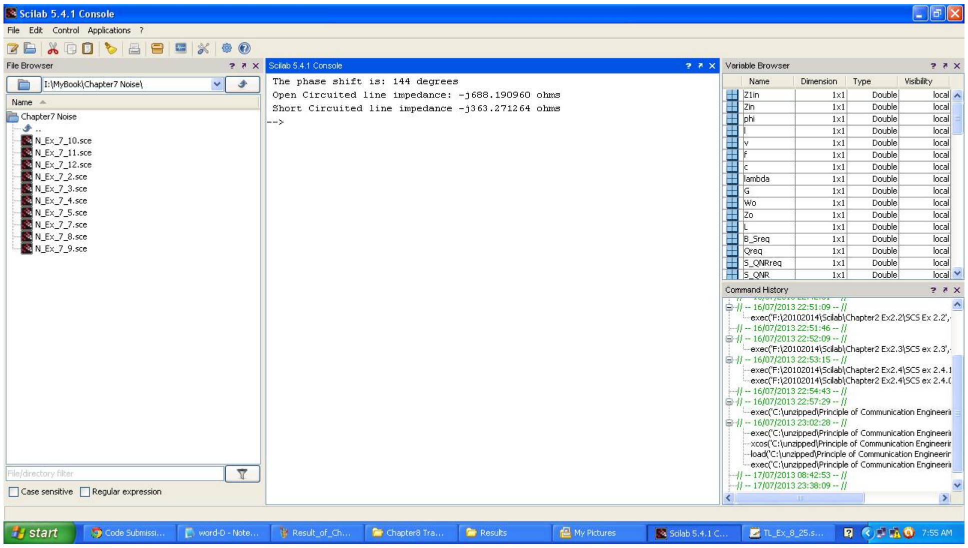

Figure 8.2: Transmission Line Ex 8 2

```
lambda=v/f
//disp(lambda)
l=100// length of the Tx-Line
phi=2*(%pi*l)/(lambda)// Phase shift in degrees
Zo=500// Characteristic impedance
//a Open circuited Line

Zin=-%i*Zo*(cos(phi)/sin(phi))

//b Short circuited Line
Z1in=%i*Zo*tan(phi)
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)
```

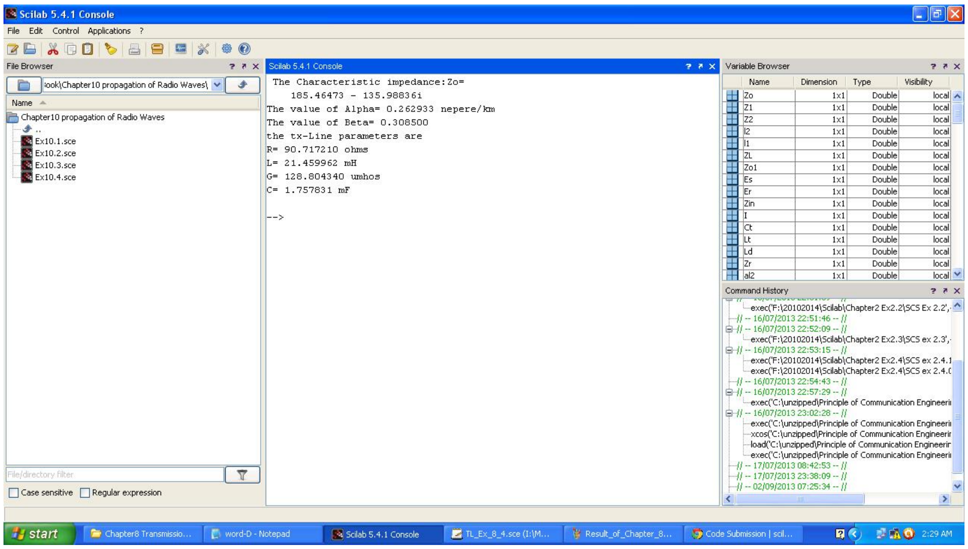

Figure 8.3: Transmission Line Ex 8 3

Scilab code Exa 8.3 `Transmission Line Ex 8 3`

```

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

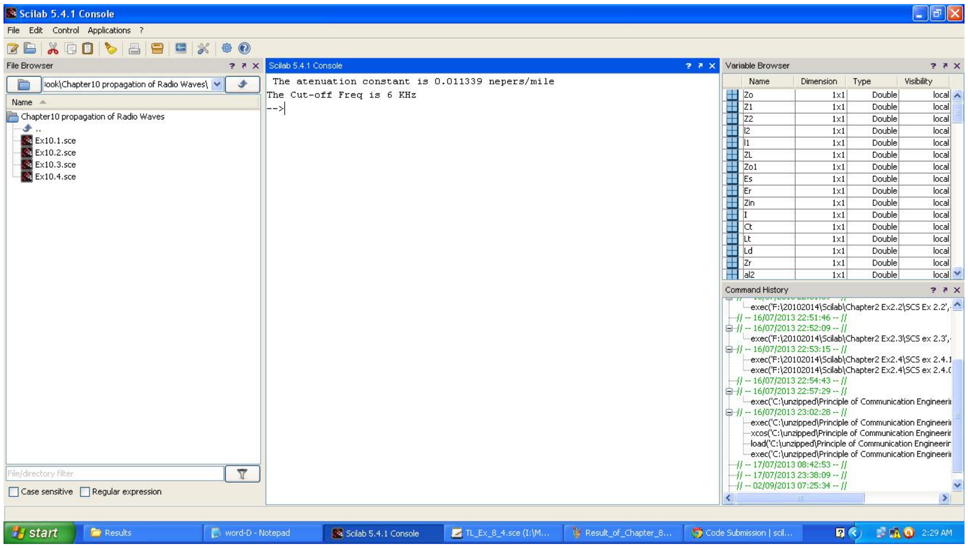

Figure 8.4: Transmission Line Ex 8 4

```
    constant

//the tx-Line parameters
R=real(Zo*complex(alpha,Beta))
L=imag(Zo*complex(alpha,Beta))
G=real(complex(alpha,Beta)/Zo)
C=imag(complex(alpha,Beta)/Zo)
mprintf('The Characteristic impedance:Zo= ') ,disp(
    Zo)
mprintf('The value of Alpha= %f nepere/km\n',alpha)
mprintf('The value of Beta= %f \n',Beta)
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)

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

Scilab code Exa 8.4 `Transmission Line Ex 8 4`

```
clc
//Chapter8
//Example8.4, page no 316
//Given
d=0.7// distance between two insertions
Ld_m= (80e-3)*(10/7)//Loading coil inductance
//disp(Ld_m)
Rd_m=100/7//Loading coil resistance
//disp(Rd_m)
R=20+Rd_m//Line resistance
L=Ld_m// Line inductance
C=0.05e-6// Line Capacitance
alfa=0.5*R*sqrt(C/L)//Attenuation Constant
//
fc=(%pi*d*sqrt(L*C))^-1//cut off freq
mprintf('The atenuation constant is %f nepers/mile\
   nThe Cut-off Freq is %d KHz',alfa,fc*1e-3)

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

Scilab code Exa 8.5 `Transmission Line Ex 8 5`

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

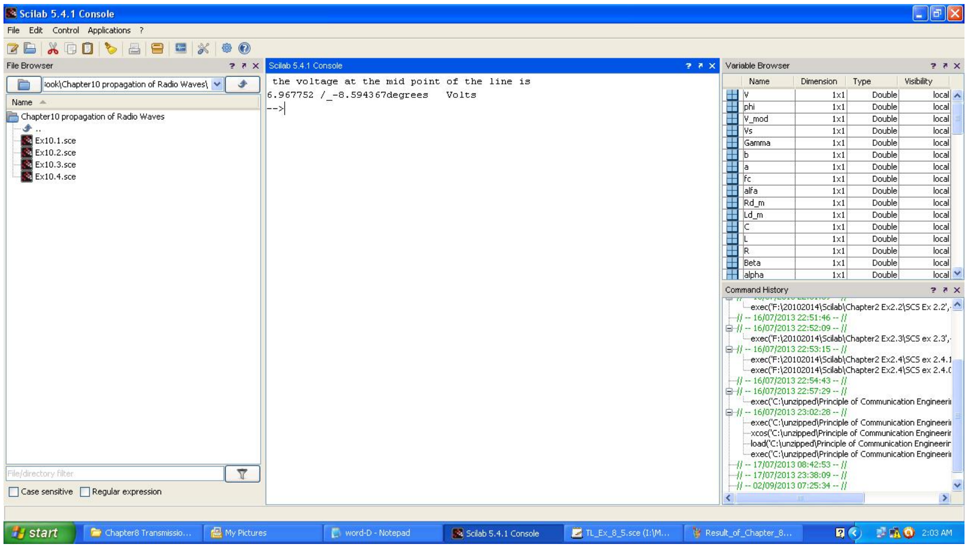

Figure 8.5: Transmission Line Ex 8 5

```
b=0.3//phase constant
Gamma=a+(%i*b)//propagation constant
l=0.5// half length of line( for midpoint)
Vs=10// Excitation voltage
V_mod=Vs*(%e^(-a*l))//Magnitude of the Vs

phi=b*l*180/%pi//phase shift
V=V_mod*(%e^(-%i*(phi*%pi/180)))//voltage at the mid point
mprintf('the voltage at the mid point of the line is \n%f /_-%fdegrees    Volts',V,phi)

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

Scilab code Exa 8.6 `Transmission Line Ex 8 6`

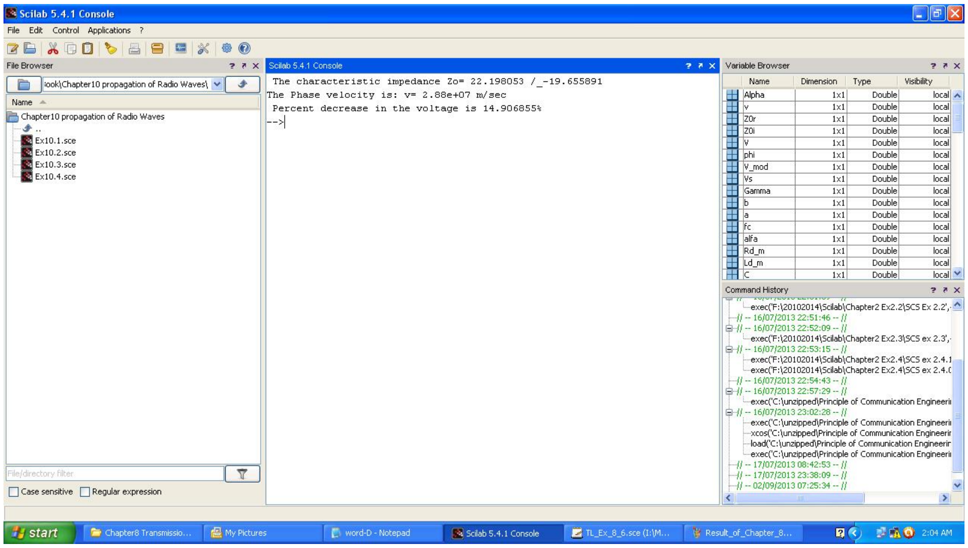

Figure 8.6: Transmission Line Ex 8 6

```
clc
//Chapter8
//Example8.6, page no 317
//GivenR=0.01
R=0.01,l=1e3
L=1e-6
G=1e-6
C=0.001e-6
f=1.59e3// operating freq
w=2*%pi*f// angular freq
//a
Zo=sqrt((R+(%i*w*L))*0.35/(G+(%i*w*C)))//characteristic impedance
[Z0r,Z0i]=polar(Zo)
//b

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

```

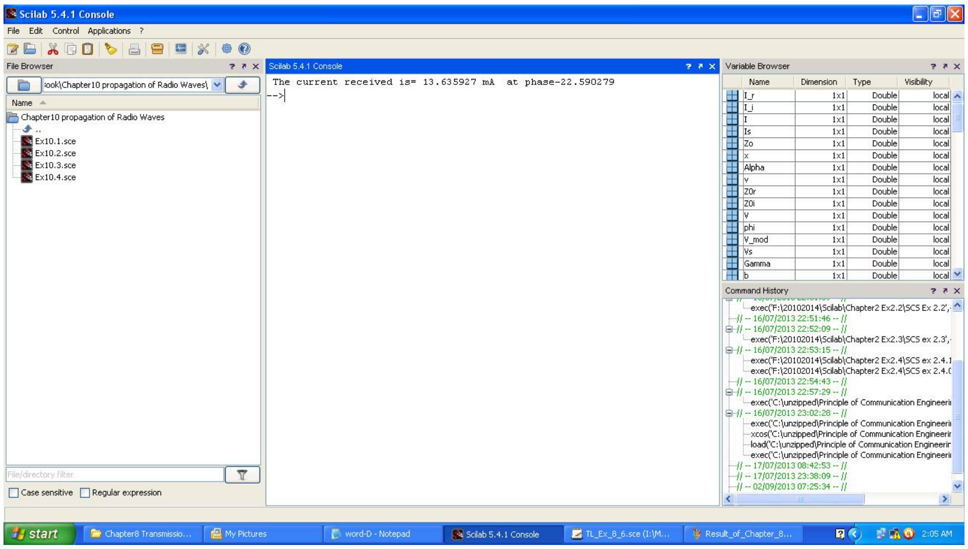

Figure 8.7: Transmission Line Ex 8 15 Pg334

```
v=w/Beta//phase velocity

//c
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
Vs=1//Assumed for easeof calculation
A=(Vs-(Vs*exp(-Alpha*l)))*100
mprintf('The characteristic impedance Zo= %f /_%f \n
    ',Z0r,Z0i*180/%pi)
mprintf('The Phase velocity is: v= %3.2e m/sec\n
    Percent decrease in the voltage is %f%c',v,A,'%')

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

Scilab code Exa 8.15 Transmission Line Ex 8 15 Pg334

```
clc
//Chapter8
//Example8.15, page no 334
//GivenR=0.01
x=10//line length
Zo=100// characteristic impedance
a=0.1// attenuation constant
Beta=0.05// phase constant
Is=20e-3// source current
Gamma=a+ %i*Beta// propagation constant

I=Is/cosh(Gamma*x)// received current

[I_r,I_i]=polar(I)

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

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

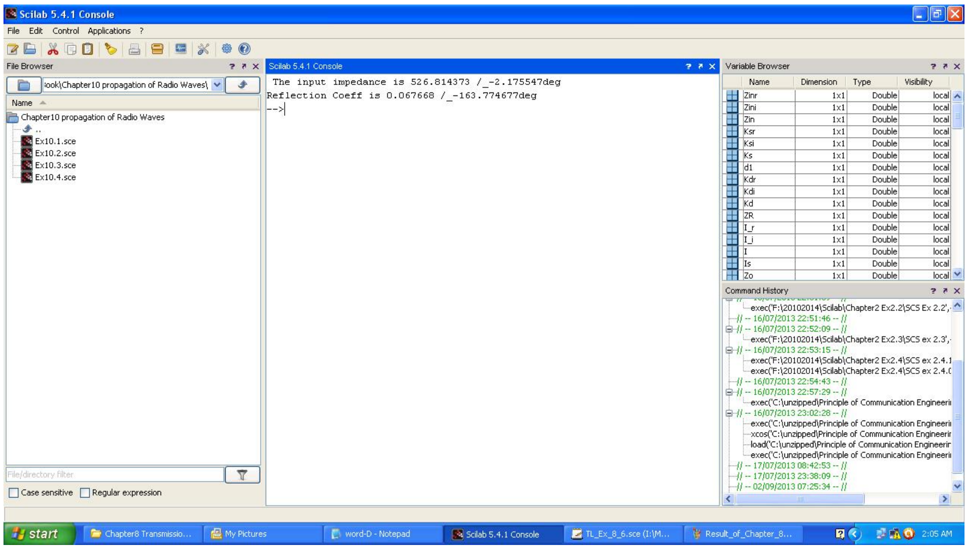

Figure 8.8: Transmission Line Ex 8 15 Pg348

```
Kd=((ZR-Zo)/(ZR+Zo))*%e^(-2*Gamma*d)//reflection
    coeff at point D d km from load
[Kdr,Kdi]=polar(Kd)
d1=100// distance
Ks=((ZR-Zo)/(ZR+Zo))*%e^(-2*Gamma*d1)//reflection
    coeff at the sending end
[Ksr,Ksi]=polar(Ks)
Zin=Zo*(((ZR*cosh(Gamma*l))+(Zo*sinh(Gamma*l)))/((Zo
    *cosh(Gamma*l))+(ZR*sinh(Gamma*l))))//Input
    impedance
[Zinr,Zini]=polar(Zin)
mprintf('The input impedance is %f /_%fdeg\
    nReflection Coeff is %f /_%fdeg',Zinr,Zini*180/
    %pi,Ksr,Ksi*180/%pi)

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

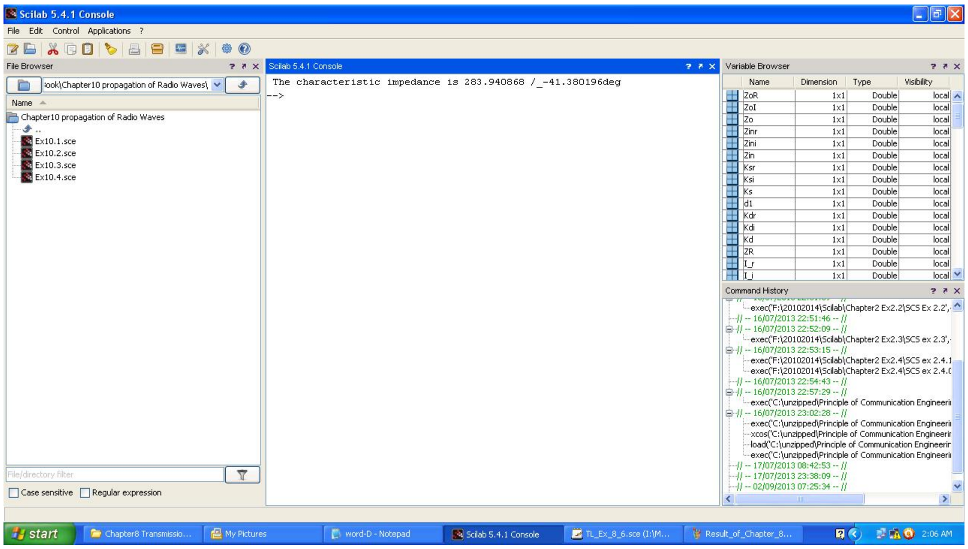

Figure 8.9: Transmission Line Ex 8 16

Scilab code Exa 8.16 `Transmission Line Ex 8 16`

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

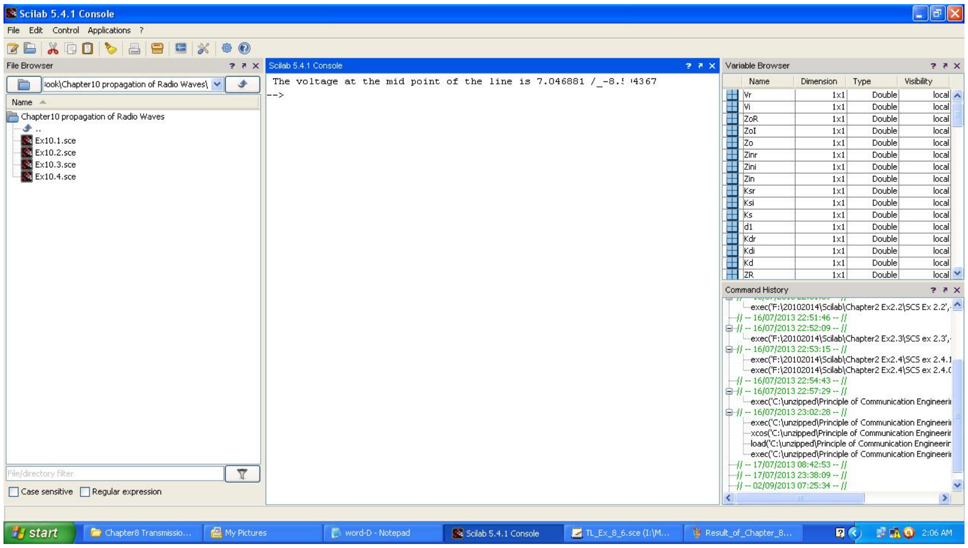

Figure 8.10: Transmission Line Ex 8 17

```

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

Scilab code Exa 8.17 Transmission Line Ex 8 17

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

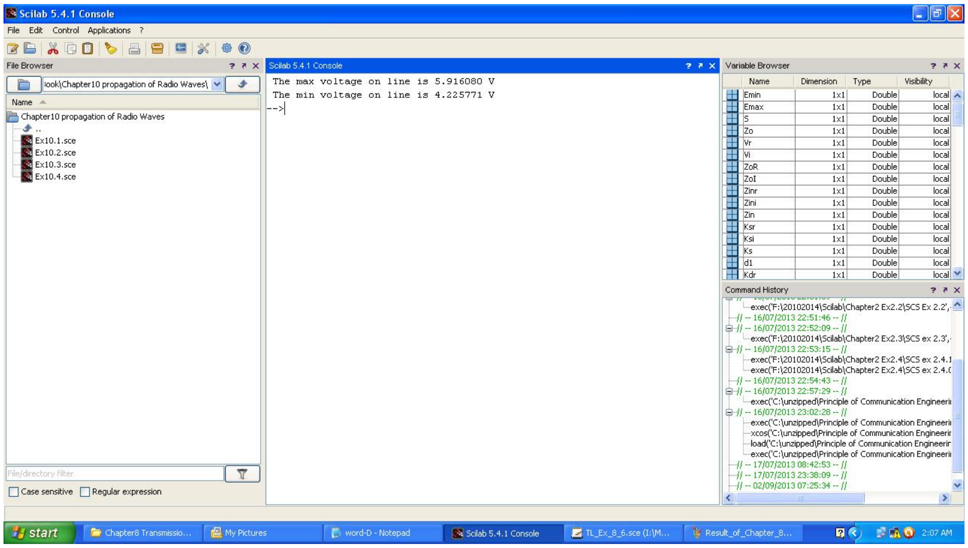

Figure 8.11: Transmission Line Ex 8 18

Scilab code Exa 8.18 `Transmission Line Ex 8 18`

```
clc
//Chapter8
//Example8.18, page no350
//Given
Zo=50// characteristic impedance
P=500e-3//Supplied power
S=1.4//VSWR on the line
Emax=sqrt(Zo*S*P)//Max vtg

Emin=sqrt(Zo*P/S)// Min vtg
mprintf('The max voltage on line is %f V\n The min voltage on line is %f V',Emax,Emin)
```

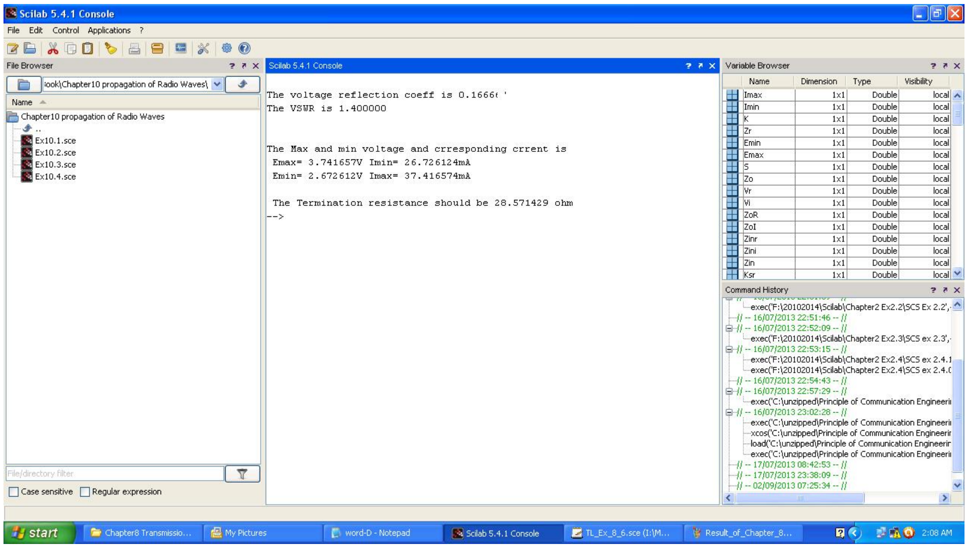

Figure 8.12: Transmission Line Ex 8 19

Scilab code Exa 8.19 `Transmission Line Ex 8 19`

```
clc
//Chapter8
//Example8.19, page no 350
//Given
Zo=100// Characteristic Impedance
P=100e-3//Load power
Zr=140//Load Resistance
f=100e3// Operating freq
//a
K=(Zr-Zo)/(Zo+Zr)//Vtg Reflection coeff

//b
S=(1+K)/(1-K)//VSWR
```

```
//c+d
Emax=sqrt(Zr*P)//Max line vltg
Imin=Emax/Zr//Min line current

Emin=Emax/S// Min line vltg
Imax=S*Imin//Max line current

//e
R=14000/40

Zr=(Zo^2)/R//
mprintf('\nThe voltage reflection coeff is %f\nThe VSWR is %f\n\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

```
clc
//Chapter8
//Example8.20, page no 352
//Given
V=0.5//receiving vtg
Vs=2//Source vtg
al=-log(V/Vs)//attenuation

al2=al*1.5
V=Vs*%e^-al2//receiving voltage
mprintf('the receiving voltage will be %f V',V)
```

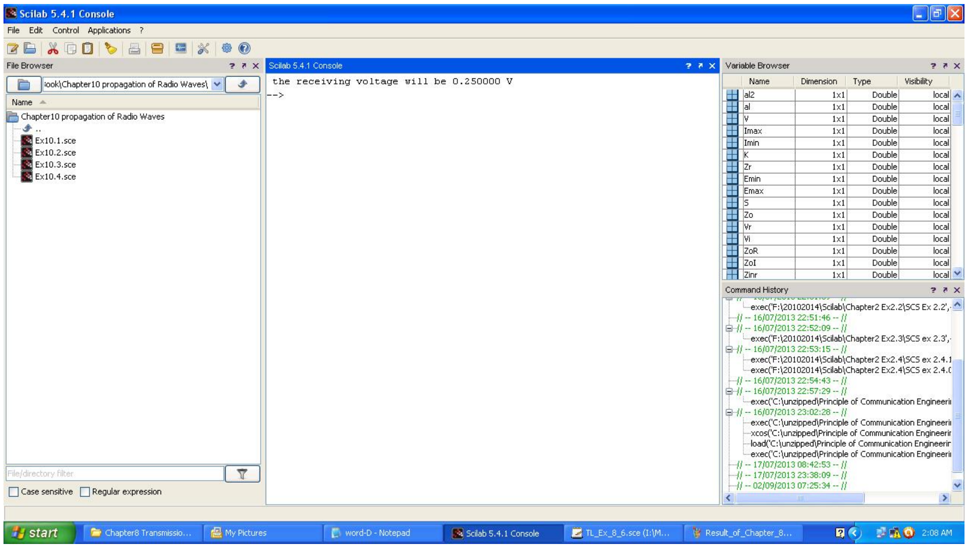

Figure 8.13: Transmission Line Ex 8 20

Scilab code Exa 8.22 `Transmission Line Ex 8 22`

```
clc
//Chapter8
//Example8.22, page no352
//Given
Zin=25+%i*15// Internal Impedance
Zr=70-%i*42//load
f=3e6//operating freq
v=3e8//light velocity
L=v/(4*f)//length of the line

Zo=sqrt(Zin*Zr)//Characteristic Impedance
mprintf('The lrngth should be %d metres\nThe
    Characteristic Impedance should be %f ohms',L,Zo)
```

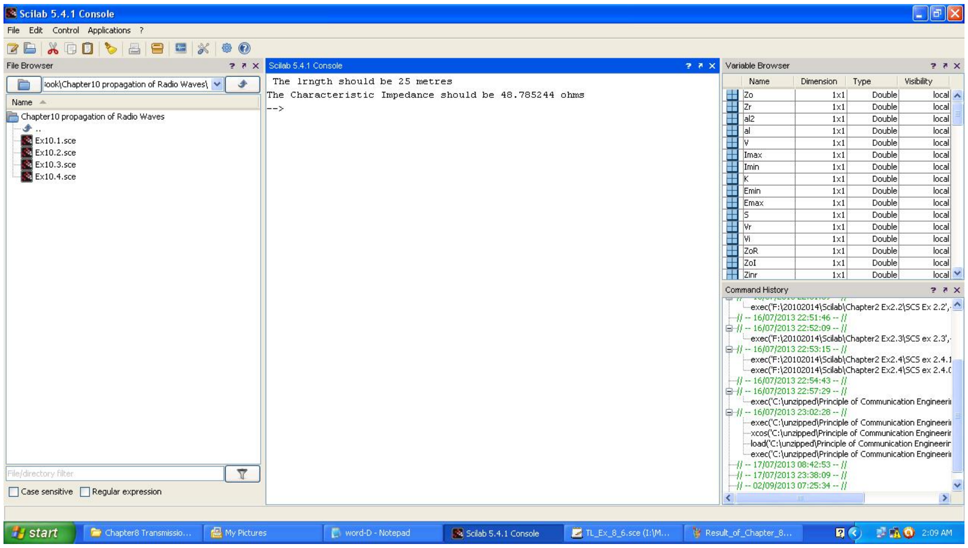

Figure 8.14: Transmission Line Ex 8 22

Scilab code Exa 8.23 `Transmission Line Ex 8 23`

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

```

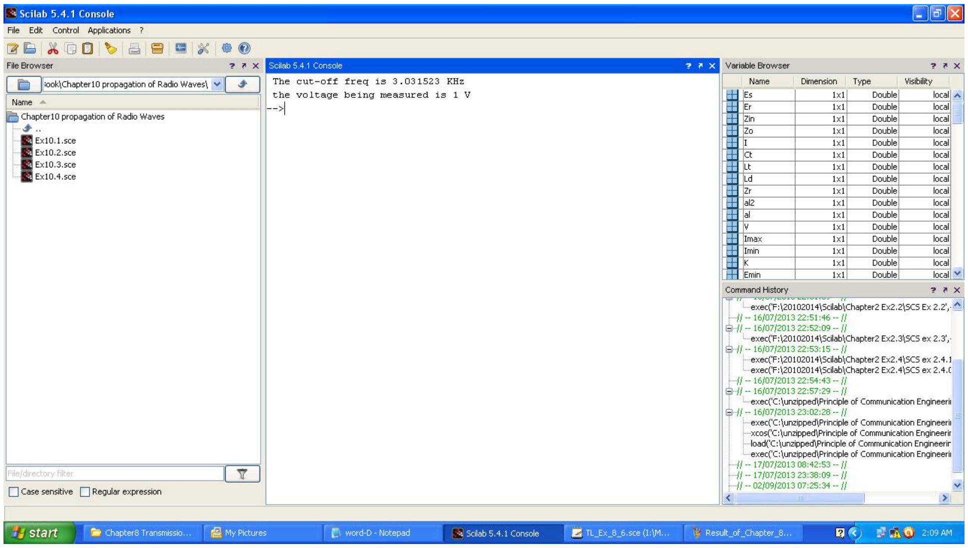

Figure 8.15: Transmission Line Ex 8 23

```
//b
I=100e-3//milliameter range
R=1//milliameter resistance
Zo=100//characteristic impedance
Zin=(Zo^2)/R//input impedance

Er=I*R//
Es=Er*sqrt(Zin/Zo)
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

```
clc
//Chapter8
//Example8.24, page no 354
```

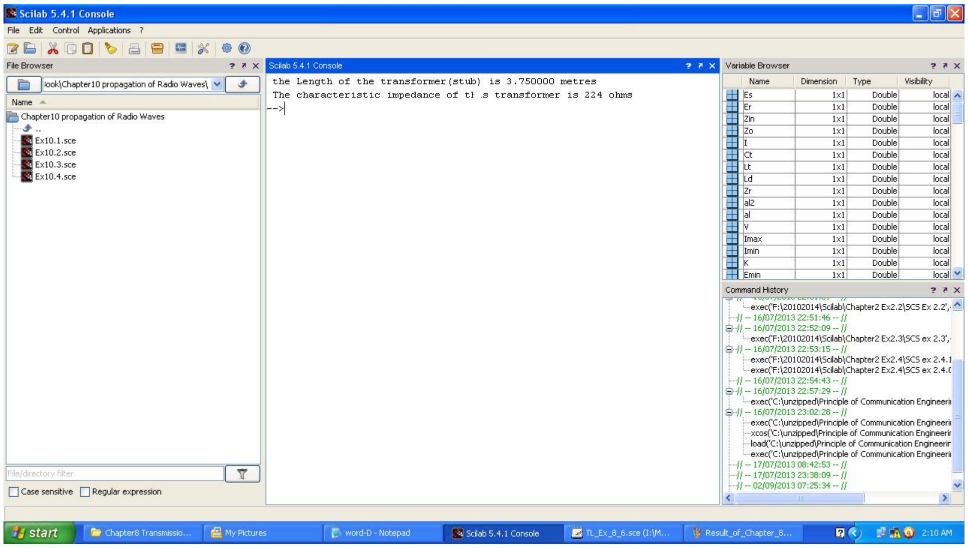

Figure 8.16: Transmission Line Ex 8 24

```
//Given
f=20e6//tuned freq
ZR=100//Equivalent aerial Resistance
Zin=500//input impedance
c=3e8
lambda=c/f
l=lambda/4//lambda/4 Transformer

Zo=sqrt(Zin*ZR)//Characteristic impedance
mprintf('the Length of the transformer(stub) is %f
    metres\n The characteristic impedance of this
    transformer is %d ohms',l,round(Zo))
```

Scilab code Exa 8.25 Transmission Line Ex 8 25

```
clc
```

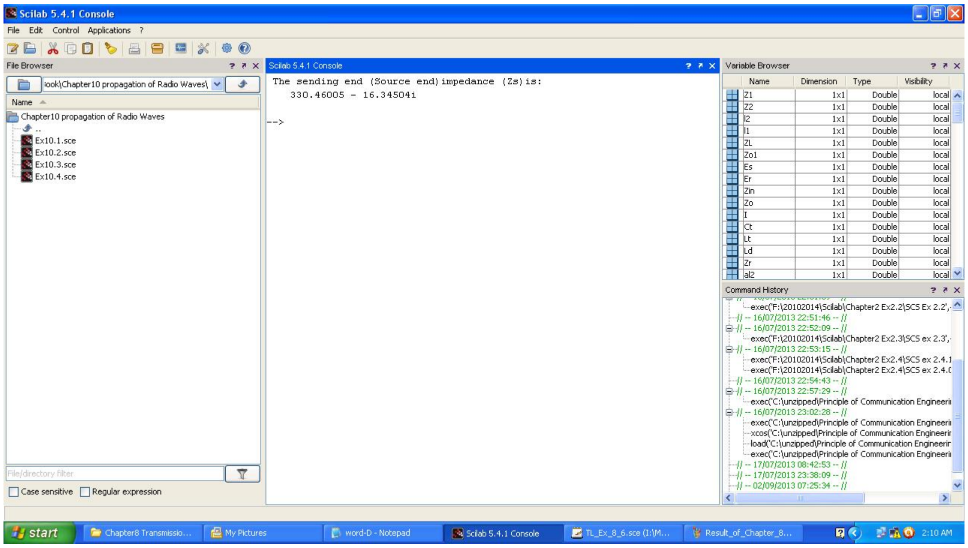

Figure 8.17: Transmission Line Ex 8 25

```
//Chapter8
//Example8.25, page no 354
//Given
lambda=5//wavelength
Zo=200//Characteristic impedance
Zo1=100//Zo'
ZL=50+(%i*50)// load impedance
l1=0.4*lambda
l2=0.2*lambda
Beta=(2*%pi/lambda)// phase difference
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
Z1=Zo*(((Z2*cos(Beta*l1))+(%i*Zo*sin(Beta*l1)))/((Zo
    *cos(Beta*l1))+(%i*Z2*sin(Beta*l1))))//I/p
    impedance
mprintf('The sending end (Source end)impedance (Zl)
    is: '),disp(Z1)
```

Chapter 9

Aerials

Scilab code Exa 9.1 Aerials Ex 9 1

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

Scilab code Exa 9.2 Aerials Ex 9 2

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

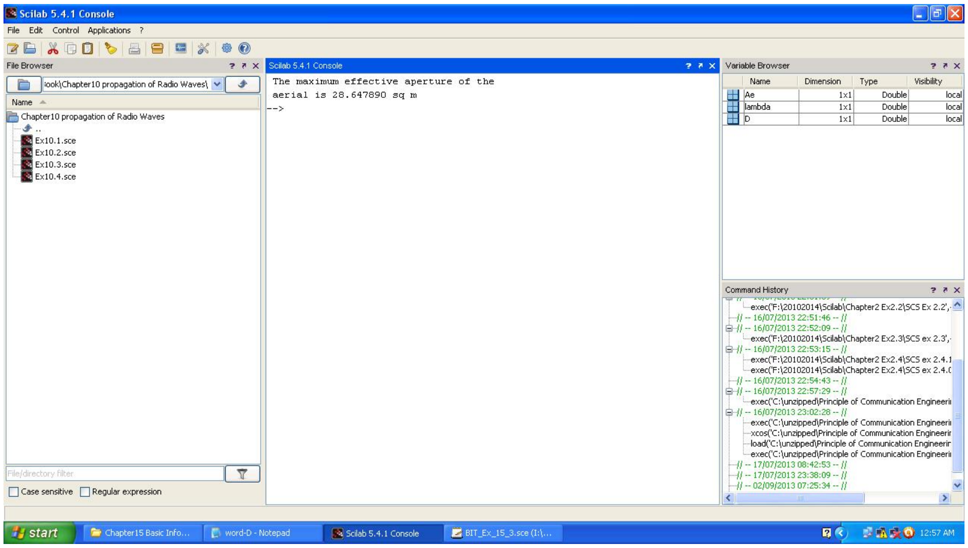

Figure 9.1: Aerials Ex 9 1

```
d=0.5//distance in terms of wavelength
Beam_Width=2/(n*d)//
Beam_Width_degrees=Beam_Width*180/%pi
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

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

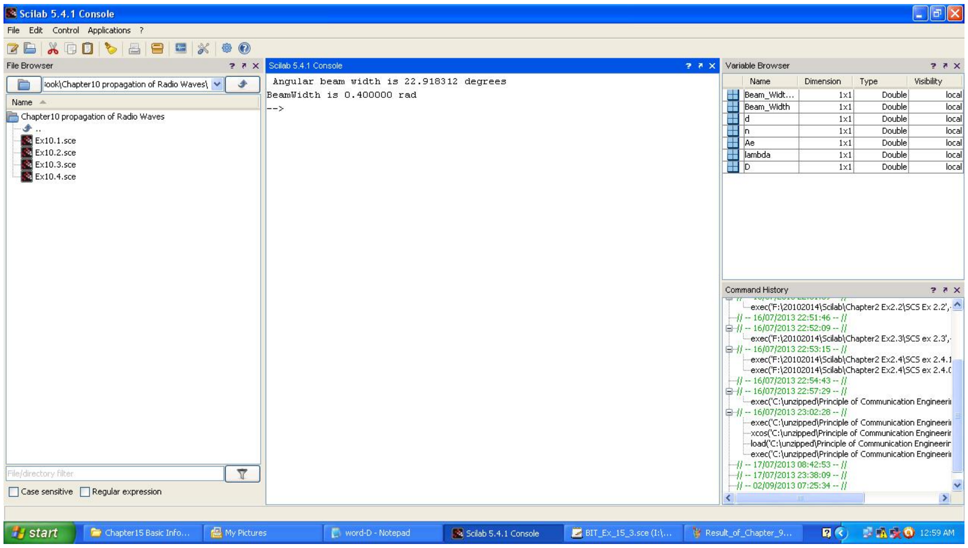

Figure 9.2: Aerials Ex 9 2

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

Scilab code Exa 9.4 Aerials Ex 9 4

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

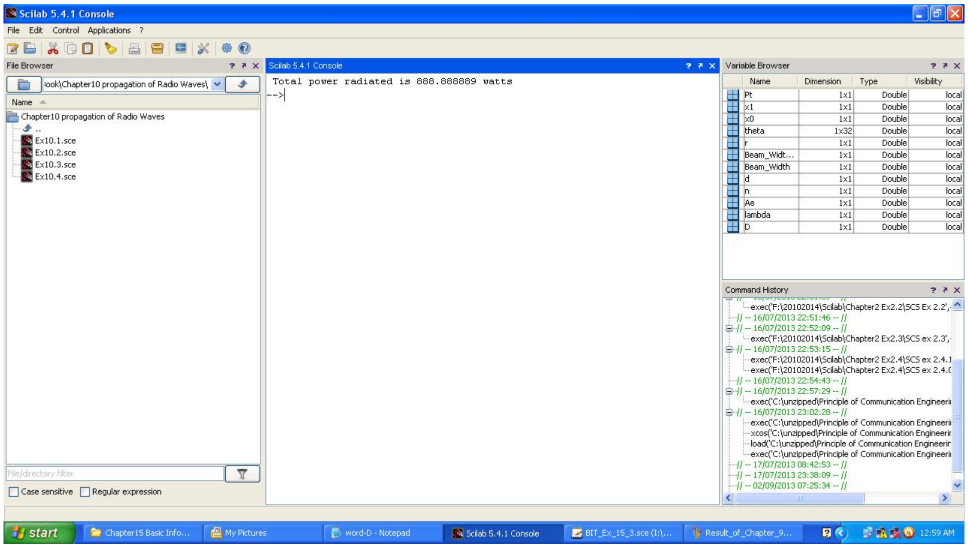

Figure 9.3: Aerials Ex 9 3

```
w=2*%pi*f//angular freq
c=3e8,t=f^-1
Phi=w*(t-(r/c))//Phase shift
Erad=25.13e-5*cos(Phi)//Radiation electric field
    intensity
H=Erad/(120*%pi)//Radiation magnetic field intensity
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

```
clc
//Chapter9
//Example9.5
//Given
```

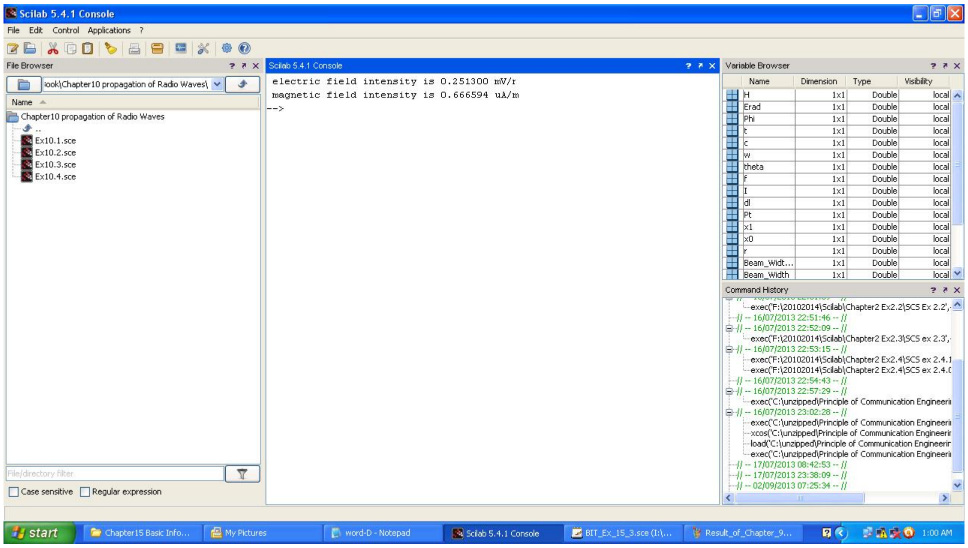

Figure 9.4: Aerials Ex 9 4

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

Scilab code Exa 9.6 Aerials Ex 9 6

```
clc
//Chapter9
//Example9.6
//Given
```

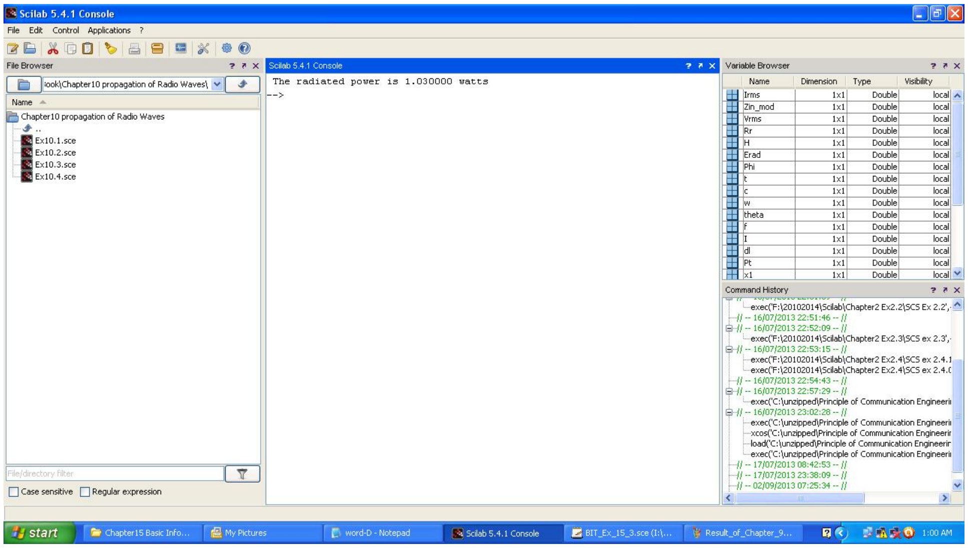

Figure 9.5: Aerials Ex 9 5

```
//b
c=3e8
f=2e9//operating freq
Ae=100//aperture area
lambda=c/f// operating wavwlength
D=((4*3.141*Ae)/(lambda^2))// Directivity
mprintf('Ideal directive gain is %d',D)
```

Scilab code Exa 9.7 Aerials Ex 9 7

```
clc
//Chapter9
//Example9.7, pageno 400
//Given
//b
n=10// no of aerial elements
```

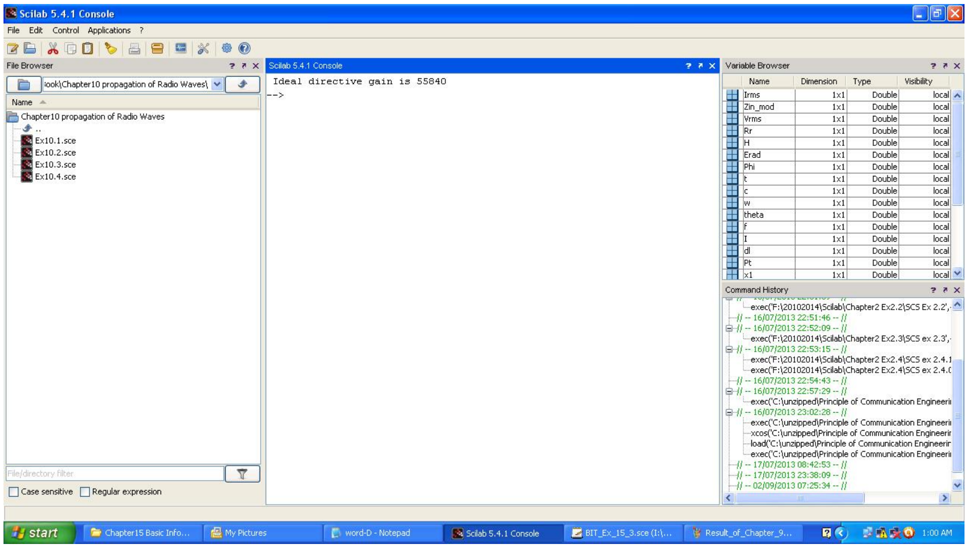

Figure 9.6: Aerials Ex 9 6

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

Scilab code Exa 9.8 Aerials Ex 9 8

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

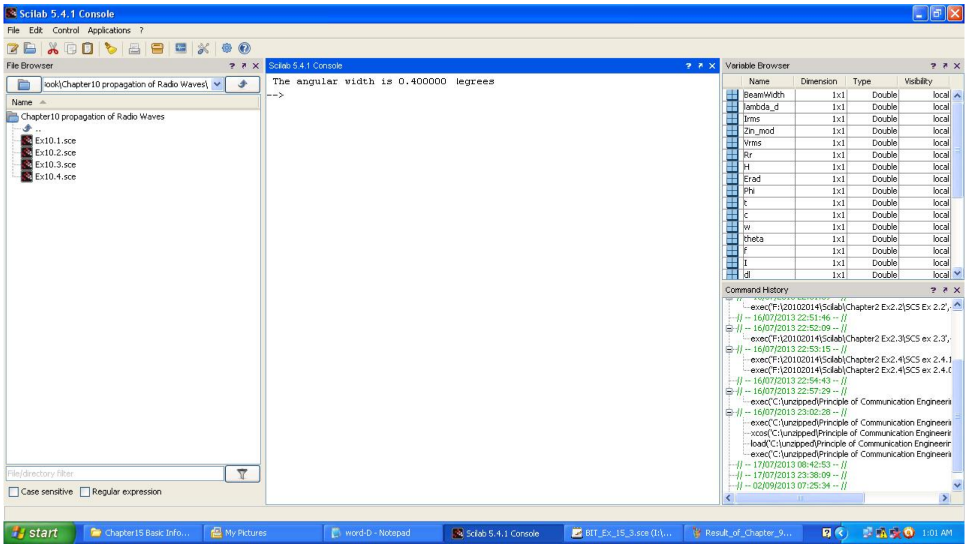

Figure 9.7: Aerials Ex 9 7

Scilab code Exa 9.9 `Aerials Ex 9 9`

```
clc
//Chapter9
//Example9.9
//Given
//b
c=3e8
f=800e3// operating freq
dl=27//effective height
lambda=c/f

Rr=40*(3.142^2)*(dl/lambda)^2//Radiation Resistance
mprintf('Radiation resistance is %f ohm',Rr)
```

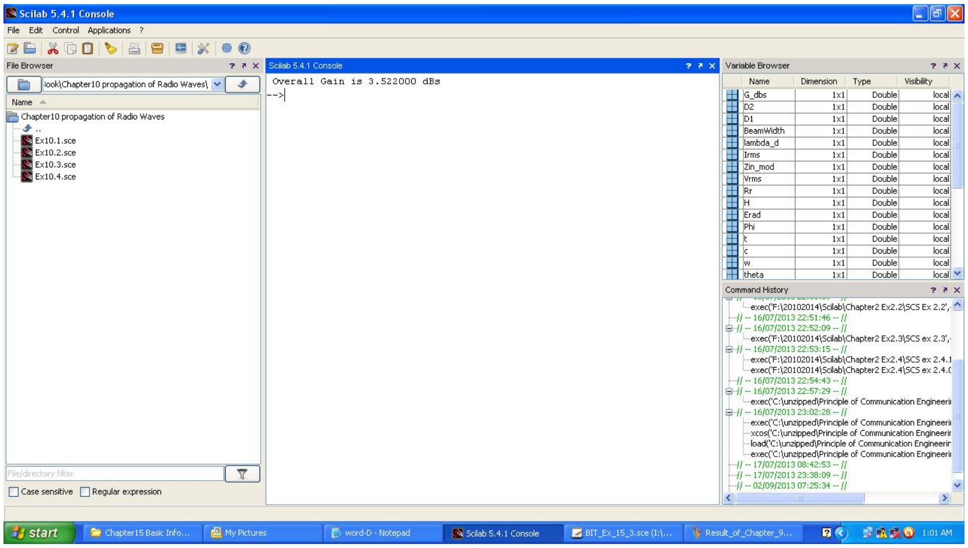

Figure 9.8: Aerials Ex 9 8

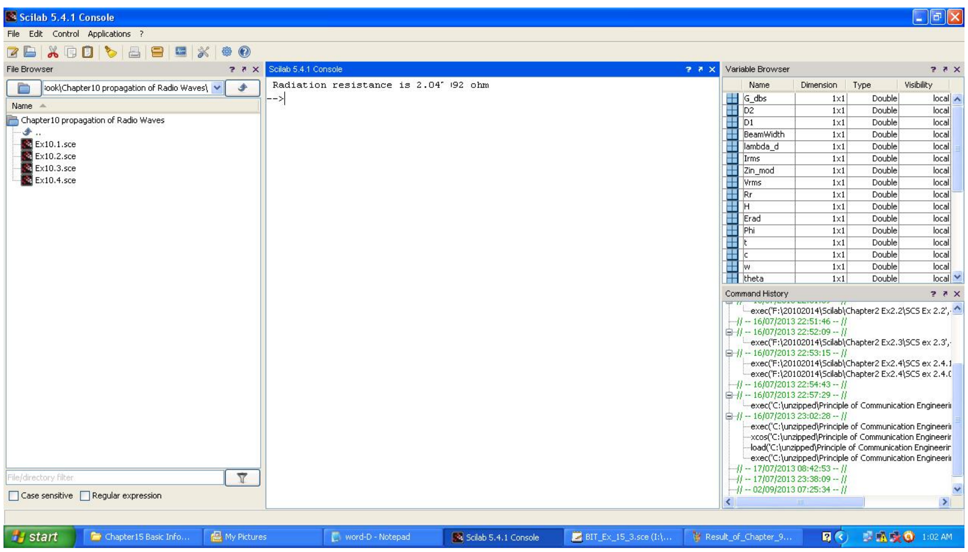

Figure 9.9: Aerials Ex 9 9

Chapter 10

Propagation of Radio Waves

Scilab code Exa 10.1 `PoRW Ex 10 1`

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

Scilab code Exa 10.2 `PoRW Ex 10 2`

```
clc
```

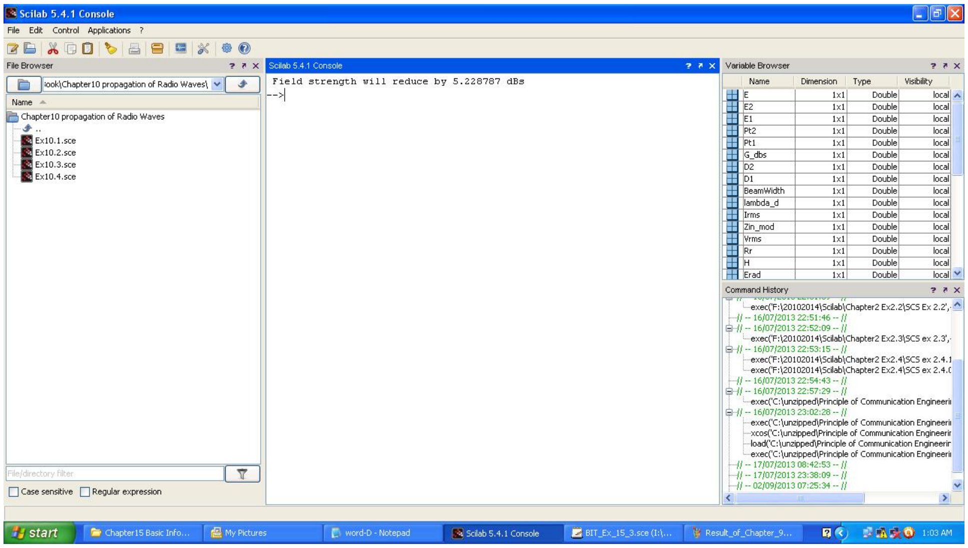

Figure 10.1: PoRW Ex 10 1

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

Scilab code Exa 10.3 PoRW Ex 10 3

```
clc
//Chapter10
```

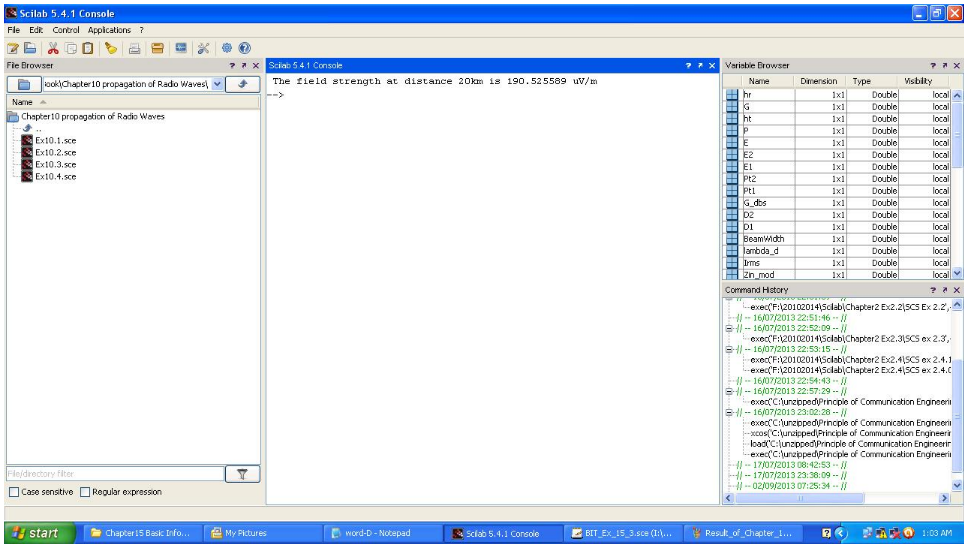

Figure 10.2: PoRW Ex 10 2

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

Scilab code Exa 10.4 PoRW Ex 10 4

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

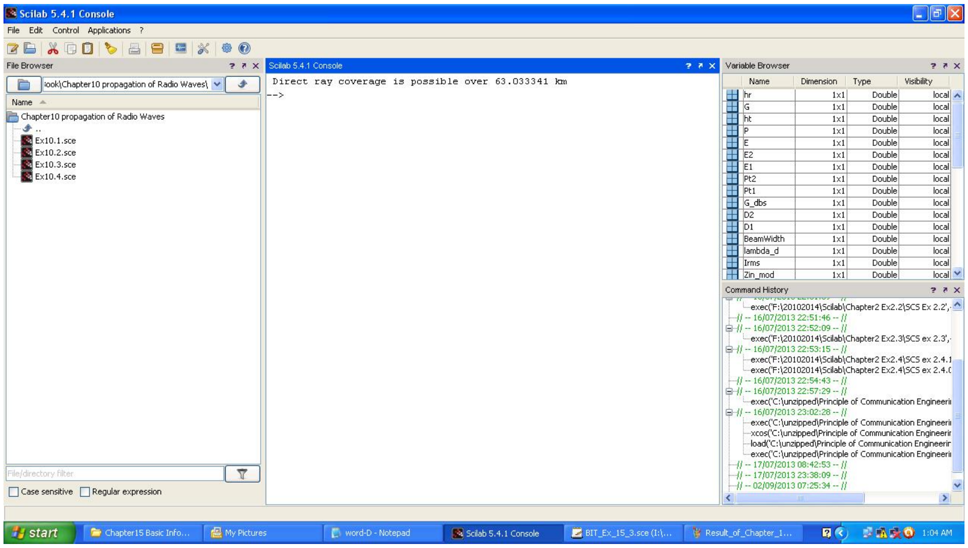

Figure 10.3: PoRW Ex 10 3

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

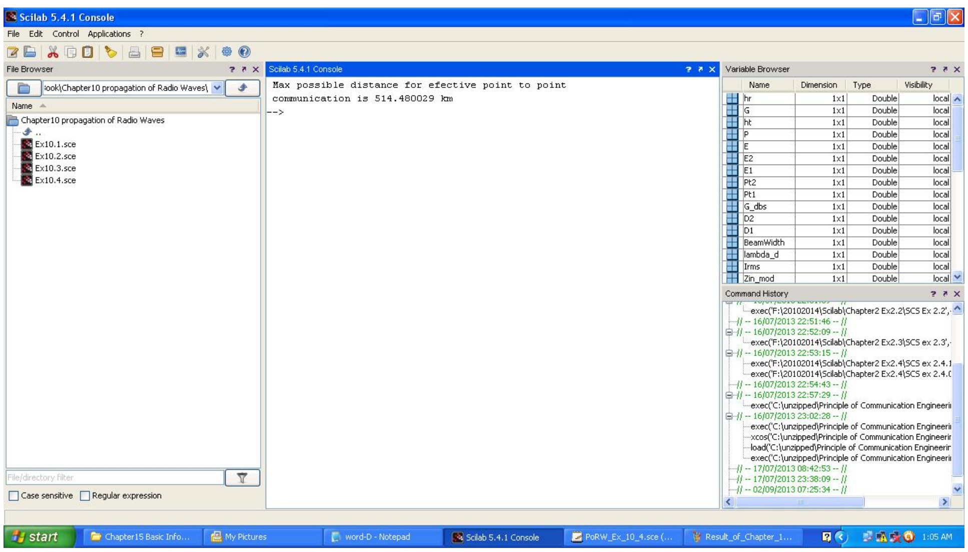

Figure 10.4: PoRW Ex 10 4

Chapter 11

BroadBand Communication

Scilab code Exa 11.1 BBC Ex 11 1

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

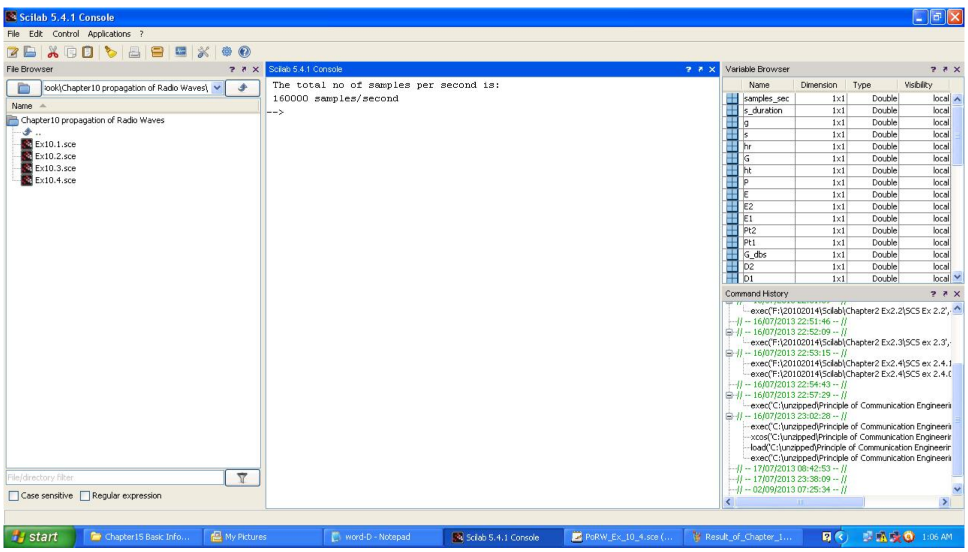

Figure 11.1: BBC Ex 11 1

Chapter 15

Basic Information Theory

Scilab code Exa 15.1 BIT Ex 15 1

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

Scilab code Exa 15.2 BIT Ex 15 2

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

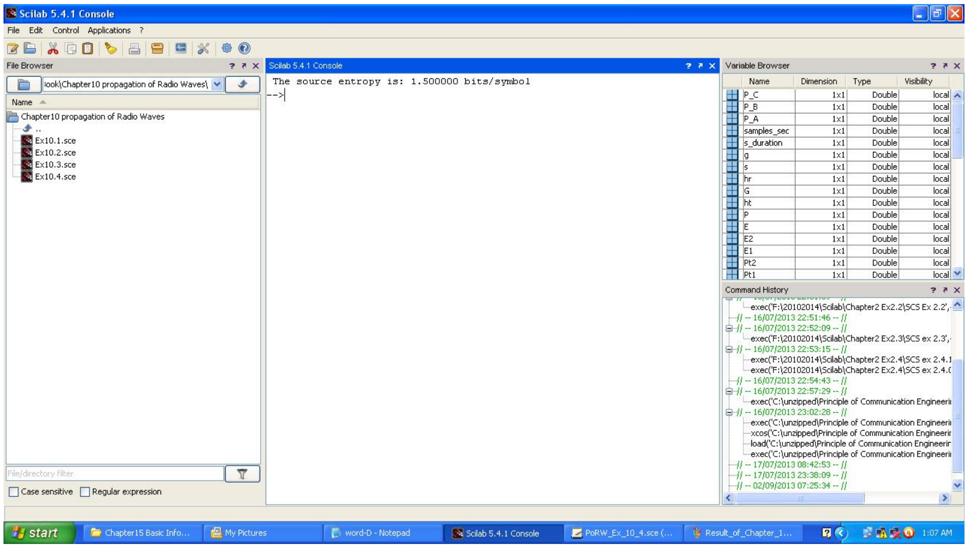

Figure 15.1: BIT Ex 15 1

```
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
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
n=6,T=1
mprintf('The source entropy is: %f bits/symbol',
    round(1000*H)/1000)
```

Scilab code Exa 15.3 BIT Ex 15 3

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

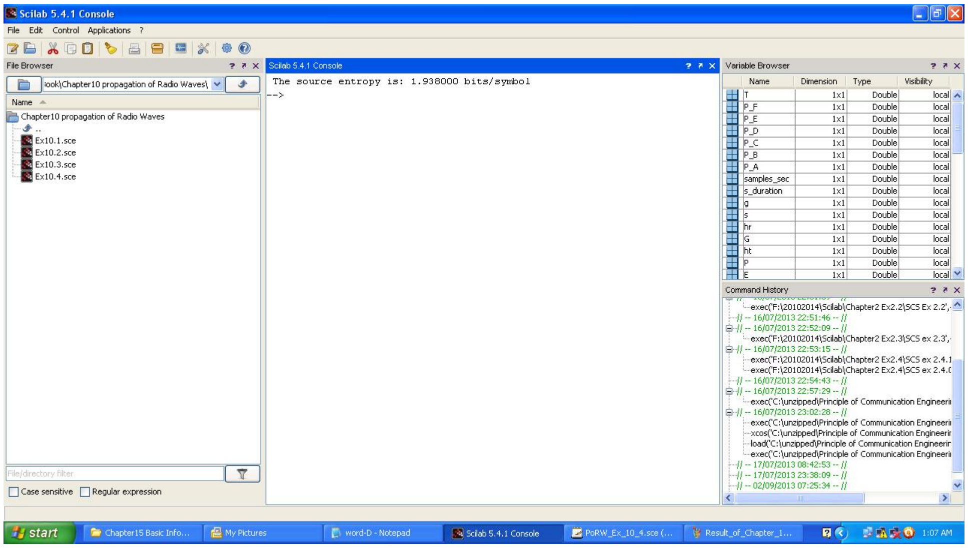

Figure 15.2: BIT Ex 15 2

```
//Given
//a
B1=4e3//Channel Bandwidth
SNR1=31//Channel SNR
C1=B1*log2(1+SNR1)//Channel Capacity
SNR2=14//Reduced SNR
B2=round(C1/log2(1+SNR2))//Bandwidth for reduced SNR with same Channel capacity

//b
B3=3e3//Reduced Bandwidth
SNR3=(2^(C1/B3))-1//Signal Power for reduced bandwidth
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)
```

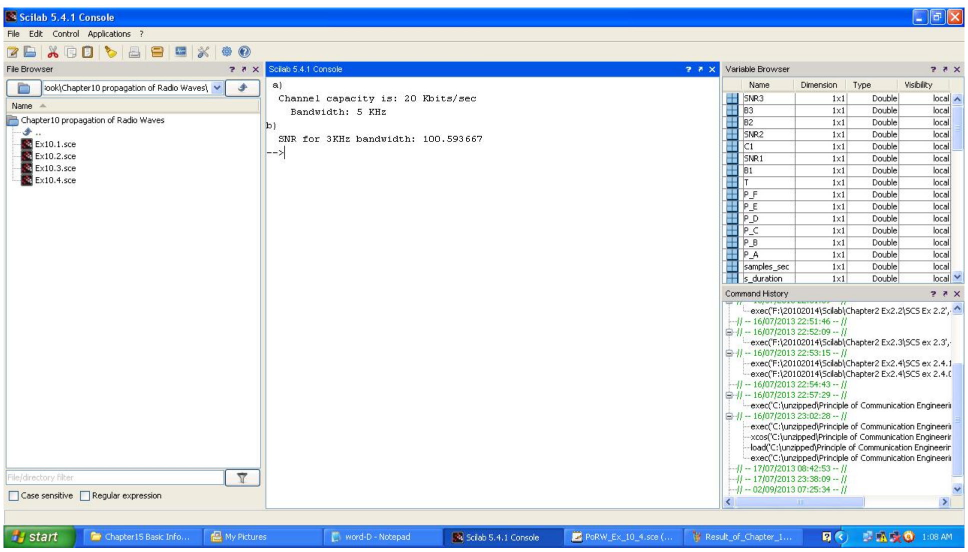

Figure 15.3: BIT Ex 15 3