Scilab Manual for
Satellite Communication
by Dr Lochan Jolly
Electrical Engineering
Thakur College of Engineering & Technology[1]

Solutions provided by
Dr Lochan Jolly
Electrical Engineering
Mumbai university/ Thakur college of engg and tech

May 24, 2026

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

Contents

List of Experiments

List of Figures

Experiment: 1

To calculate coverage angle.

Scilab code Solution 1.1 Coverage angle

```
1 //find coverage angle for given elevation
2 //OS=Windows XP sp3
3 //Scilab version 5.4.0
4 //Sample Values
5 //rsh=42164;
6 //reh=6378;
7 //E=5;
8 clear;
9 close;
10 clc;
11 rsh=input("enter the height of satellite rsh");
12 reh=input("enter the radius of earth reh");
13 E=input("enter the value of angle of Elevation");
14
15 alpha=reh/rsh;
16 ca=2*asind(alpha*cosd(E));
17 disp ("degree", ca,"coverage angle=");
```

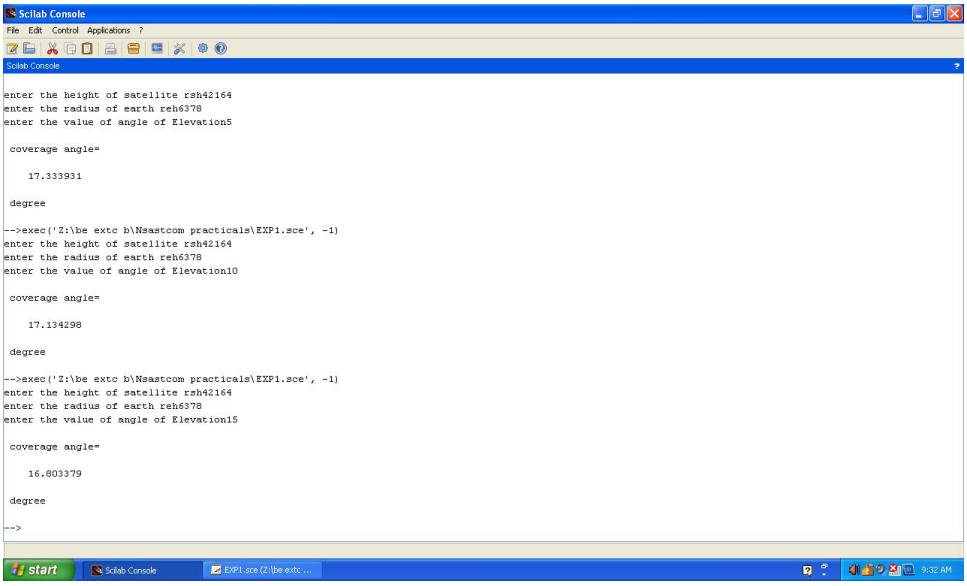

Figure 1.1: Coverage angle

Experiment: 2

To calculate slant range .

Scilab code Solution 2.2 slant range

```
1 //find  slant range  for given elevation
2 //OS=Windows XP sp3
3 //Scilab version 5.4.0
4 //Sample Values
5 //rsh=42164;
6 //reh=6378;
7 //E=5;
8 clear;
9 close;
10 //clc;
11 rsh=input("enter the height of satellite rsh");
12 reh=input("enter the radius of earth reh");
13 E=input("enter the value of angle of Elevation");
14 //rsh=42164;
15 //reh=6378;
16 //E=5;
17 v=3*10^8;
18 alpha=reh/rsh;
19 S=cosd(E+alpha)*reh/sind(alpha);
20 disp ("Km",S, "Slant Range=");
```

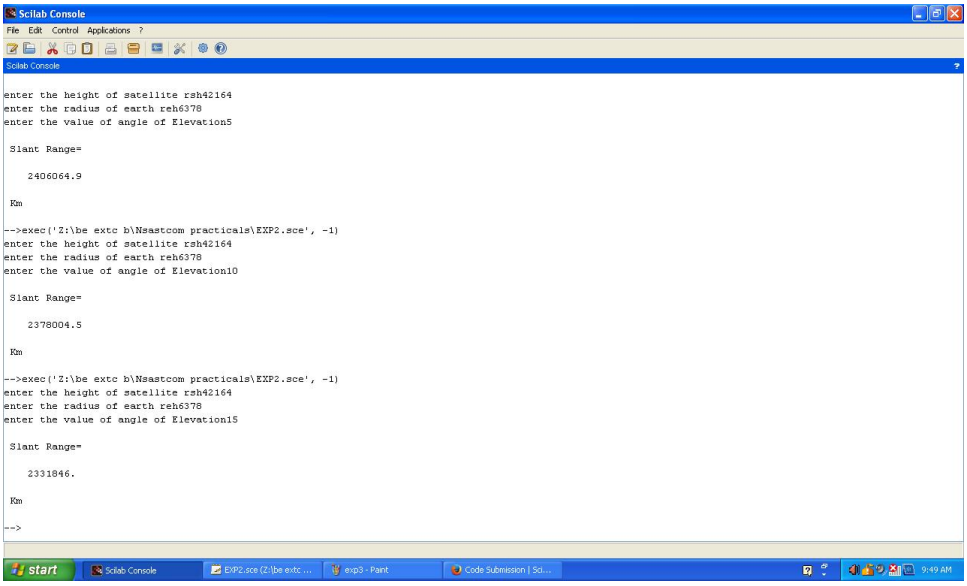

Figure 2.1: slant range

Experiment: 3

To calculate propagation delay

Scilab code Solution 3.3 Propagation Delay

```
1  //find  propagation delay for given elevation
2  //OS=Windows XP sp3
3  //Scilab version 5.4.0
4  //Sample Values
5  //rsh=42164;
6  //reh=6378;
7  //E=5;
8  clear;
9  close;
10 //clc;
11 rsh=input("enter the height of satellite rsh");
12 reh=input("enter the radius of earth reh");
13 E=input("enter the value of angle of Elevation");
14 //rsh=42164;
15 //reh=6378;
16 //E=5;
17 v=3*10^8;
18 alpha=reh/rsh;
19 S=cosd(E+alpha)*reh/sind(alpha);
20 P=2*S/v;
21 disp ("sec",P, "Propagation delay=");
```

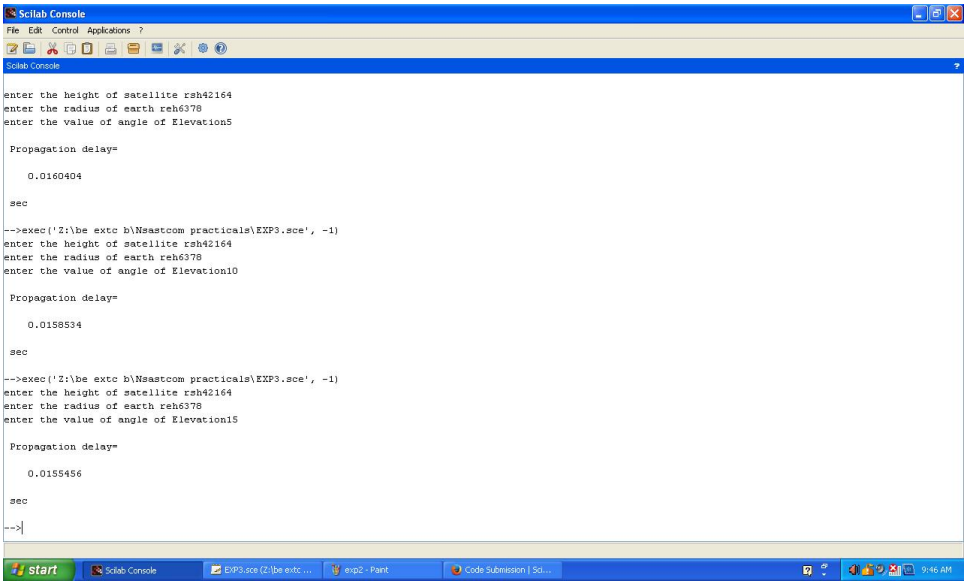

Figure 3.1: Propagation Delay

Experiment: 4

To calculate Antenna Look Angels.

Scilab code Solution 4.4 Look angels

```
1  //find Azimuth and elevation
2  //OS=Windows XP sp3
3  //Scilab version 5.4.0
4  //Sample Values
5  //La=39;
6  //dirae='N';
7  //Loe=77;
8  //diroe='W'
9  //Los=99;
10 //diros='W';
11 //re=6318;
12 //rs=42164;
13 clear;
14 close;
15 clc;
16 La=input("enter the value of Latitude of ES Lae");
17 dirae=input("enter the direction of Lattitude s (N/S
       )");
18 Loe=input("enter the value of Longitude of earth
```

```
    station Loe");
19 diroe=input("enter the direction of Longitude of ES
    (E/W)");
20 Los=input("enter the value of Longitude of satellite
     Los");
21 diros=input("enter the direction of Longitude of
    Satellite (E/W)");
22 re=input("enter the value of earth radius re");
23 rs=input("enter the height of satellite rs");
24 Adash=atand(tand(Los-Loe)/sind(La));
25 El=atand((rs-re*cosd(La)*cosd(abs(Los-Loe)))/(re*
    sind(acosd(cosd(La)*cosd(abs(Los-Loe))))))-acosd(
    cosd(La)*cosd(abs(Los-Loe)))
26
27 if dirae=='N' then
28     if diroe=='E' & diros=='E' then
29     if Loe>Los then A=180+Adash; else A=180-Adash;
          end; end;
30     if diroe=='E' & diros=='W' then A=180+Adash;end;
31     if diroe=='W' & diros=='E' then A=180-Adash; end
          ;
32     if diroe=='W' & diros=='W' then
33     if Loe>Los then A=180-Adash; else A=180+Adash;
34    end; end;
35 end;
36   if dirae=='S' then
37           if diroe=='E' & diros=='E' then
38      if Loe>Los then A=360-Adash; else A=Adash; end;
           end;
39     if diroe=='E' & diros=='W' then A=360-Adash; end
          ;
40     if diroe=='W' & diros=='E' then A=Adash; end;
41     if diroe=='W' & diros=='W' then
42     if Loe>Los then A=Adash; else A=360-Adash;
43    end;   end;
44 end;
45 disp ("degree", A, "Azimuth=");
46 disp ("degree", El, "Angle of elevation=");
```

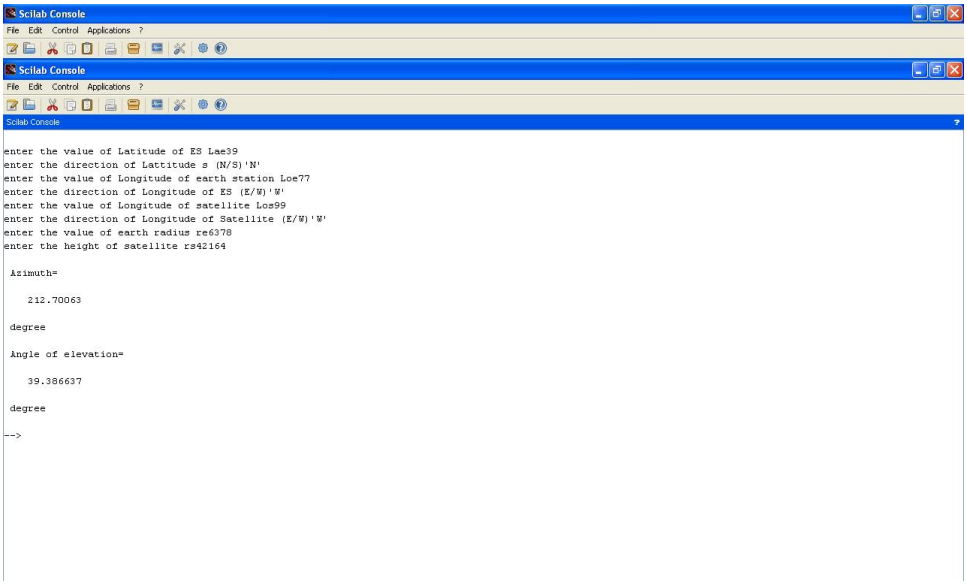

Figure 4.1: Look angels

Experiment: 5

To calculate Limits of visibility.

Scilab code Solution 5.5 Visibility Limit

```
1 //find limit of visibility of Earth station at
     certain lattitude and longitude and with certain
     elevation
2 //OS=Windows XP sp3
3 //Scilab version 5.4.0
4 //Sample Values
5 //rsh=42164;
6 //reh=6378;
7 //E=5;u can also keeo E=0,10
8 //La=48.429
9 //Lo=89.23
10 //dir='W'
11 clear;
12 close;
13 clc;
14 rsh=input("enter the height of satellite rsh");
15 reh=input("enter the radius of earth reh");
16 E=input("enter the value of angle of Elevation");
17 La=input("enter the value of Latitude La");
18 Lo=input("enter the value of Longitude Lo");
19 dir1=input("enter the direction of Longitude s (E/W)
```

```
    ");
x=reh*cosd(E)/rsh;
alpha=asind(x);
theta=90-E-alpha;
beta1=acosd(cosd(theta)/cosd(La));
LV1=Lo+beta1;//limit of visibility
LV2=Lo-beta1;//limit of visibility
if LV1>0 s1=dir1
elseif (dir1=='E') s1='W'
elseif (dir1=='W')  s1='E'
end
if LV2>0 s2=dir1
elseif (dir1=='E') s2='W'
elseif (dir1=='W')  s2='E'
end
disp ("degree", s1, abs(LV1),"Limit of visibilty 1=")
    );
disp ("degree", s2, abs(LV2),"Limit of visibilty 2=")
    );
```

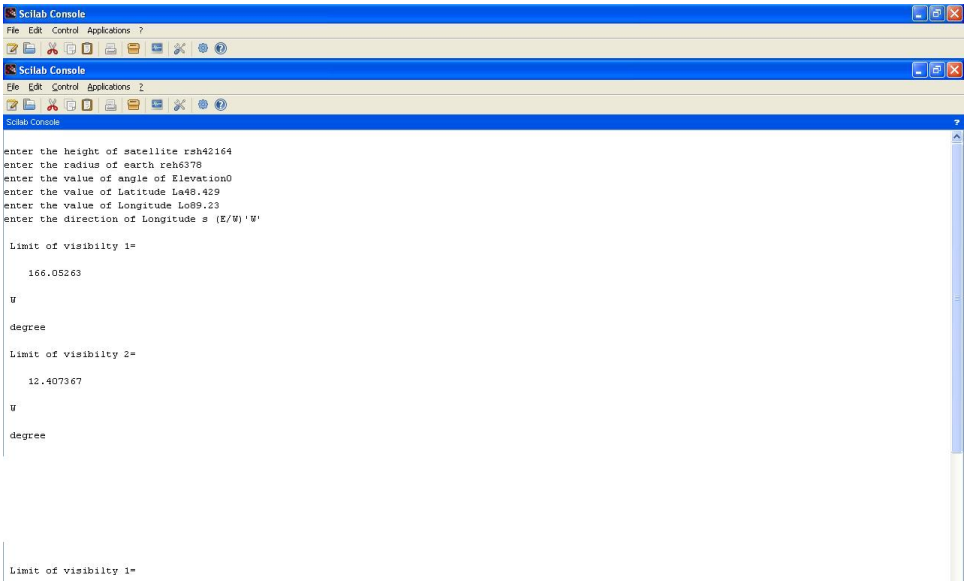

Figure 5.1: Visibility Limit

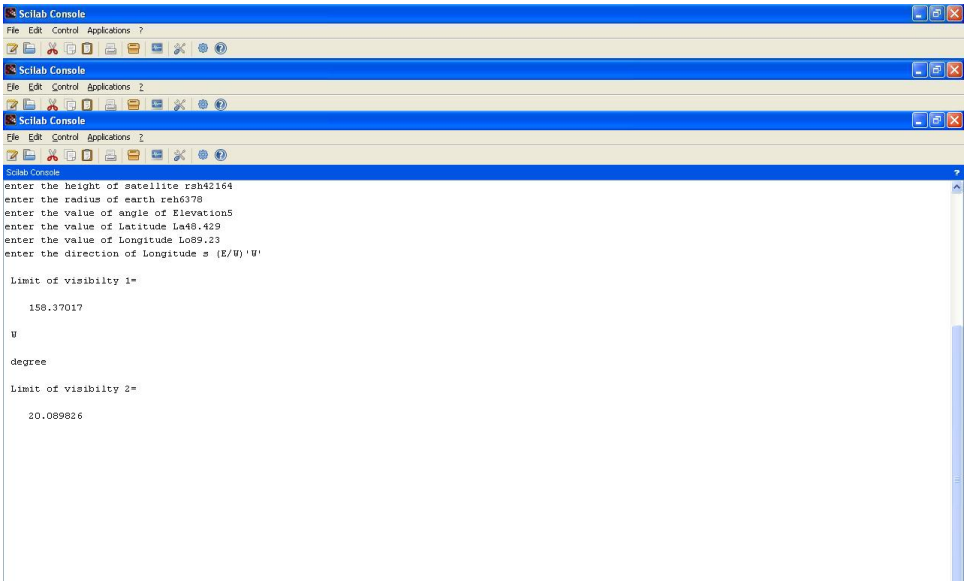

Figure 5.2: Visibility Limit

Experiment: 6

To calculate Antenna Power and antenna gain for any given satellite communication antenna.

Scilab code Solution 6.6 Antenna

```
1 //find the gain of an antenna
2 //OS=Windows XP sp3
3 //Scilab version 5.4.0
4 //Sample Values
5 //F=4
6 //Pt=100
7 //Dt=3
8 clear;
9 close;
10 clc;
11 F=input("enter the value of frequency in Ghz F")
12 c=3e8;
13 lambda= c/(F*1e9);
14 //find the gain of parabolic reflector at given
   frequency and daimeter
```

Figure 6.1: Antenna

```
15 eta=0.55;//efficiency
16 Pt=input("Power at the input of the antenna")
17 Dt=input("enter the value of daimeter of
      transmitting antenna=");
18 Gt=eta*(%pi*Dt/lambda)^2;
19 Gtdb=10*log(Gt);
20 po=Gt*Pt;
21 disp (Gtdb,"Antenna gain in dB=");
22 disp ('Watts',po,"Power output from the antenna=");
```

Experiment: 7

To calculate EIPR for any given satelite communication link.

Scilab code Solution 7.7 Link1

```
1  //Find EIRP
2  //OS=Windows XP sp3
3  //Scilab version 5.4.0
4  //Sample Values
5  //F=4
6  //Pt=100
7  //Dt=3
8  clear;
9  close;
10 clc;
11 F=input("enter the value of frequency in Ghz F")
12 c=3e8;
13 lambda= c/(F*1e9);
14 eta=0.55;//efficiency
15 Pt=input("Power at the input of the antenna")
16 Dt=input("enter the value of daimeter of
      transmitting antenna=");
17 Gt=eta*(%pi*Dt/lambda)^2;
18 Gtdb=10*log(Gt);
```

Figure 7.1: Link1

```
19 po=Gt*Pt;
20 EIRP=10*log(po);
21 disp (Gtdb,"Antenna gain in dB=");
22 disp ('dB',EIRP,"EIRP=");
```

Experiment: 8

To calculate (C/N) for Communication Satellite link considering various losses when input transmitted power is given.

Scilab code Solution 8.8 Link2

```
1 // To find the signal to noise ratio for satelite
     communication link for given input power
2 //OS=Windows XP sp3
3 //Scilab version 5.4.0
4 //Sample Values
5 //F=6.1
6 //Gtdb=54
7 //R=37500
8 //Losses=6
9 //Pt=100
10 //Grdb=26
11 //Ta=35
12 //Tr=500
```

```
13 //B=36
14 clear;
15 close;
16 clc;
17 F=input("enter the value of frequency in Ghz F");
18 c=3e8;
19 lambda= c/(F*1e9);
20 //find the gain of parabolic reflector at given
      frequency  and daimeter
21 Gtdb=input("enter the value of gain of transmitting
      antenna in dB=");
22 Gt=10^(Gtdb/10);
23 //calculate Free space Loss for given frequency and
      distance between earth and satellite log10
24 R=input("enter the value of distance between earth
      and satellite=");
25 FSL=(4*%pi*R*1e3/lambda);
26 FSLdb=20*log10(FSL);
27 losses=input("enter the value of additional losses
      in dB")
28 Totallosses=FSLdb+losses
29 TL=10^(Totallosses/10);
30 //determine received power
31 Pt=input("enter the value of power transmitted in W=
      ");
32 Grdb=input("enter the value receiver antenna gain in
       dB=");
33 Gr=10^(Grdb/10);;
34 Pr=Pt*Gt*Gr/TL;
35 Prdb=10*log10(Pr);
36 //Calculate noise power density and noise power
37 Ta=input("enter the value of antenna noise
      temperaturem in K=");
38 Tr=input("enter the value receiver noise temperature
       in K=");
39 B=input("enter the value receiver bandwidth MHz=");
40 En=Ta+Tr;//equivalent noise temperature
41 K=1.36e-23;//Boltzman constant
```

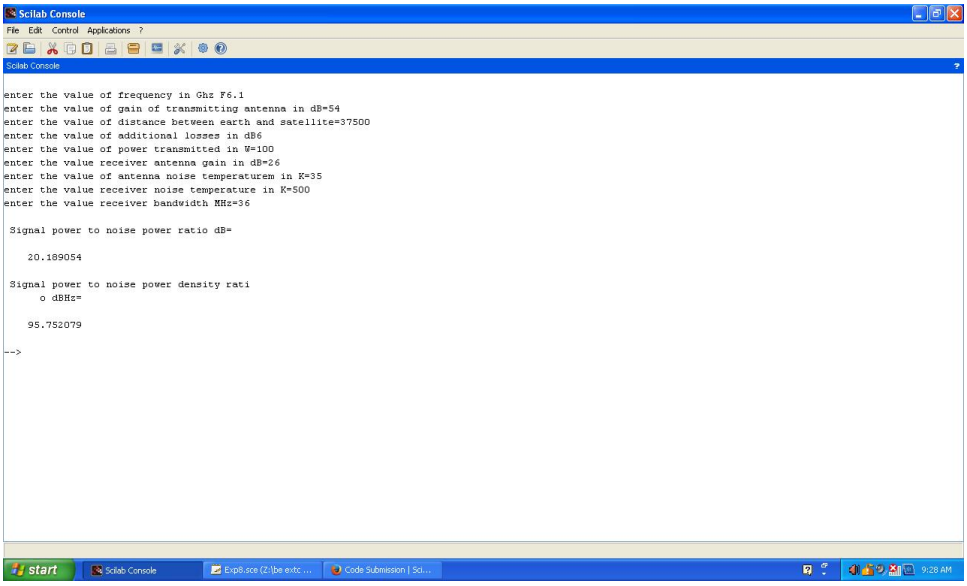

Figure 8.1: Link2

```
42 Pno=K*En*B*1e6;//noise power
43 Pnodb=10*log10(Pno);
44 Pnod=K*En;//noise power density
45 Pnoddb=10*log10(Pnod);
46 // signal to noise ratio
47 CN=Prdb-Pnodb;
48 CN0=Prdb-Pnoddb;
49 disp ('dB',CN,"Signal power to noise power ratio dB=
    ");
50 disp ('dBHz',CN0,"Signal power to noise power
    density ratio dBHz=");
```

Experiment: 9

To calculate (C/N) for Communication Satellite link considering various losses when saturation flux density is given.

Scilab code Solution 9.9 Link3

```
1 //upliknk frequency is given and flux density
    required to saturate the transponder is given
    calculate signal to noise ratio
2 //OS=Windows XP sp3
3 //Scilab version 5.4.0
4 //Sample Values
5 //F=14
6 //R=42164
7 //saidb=-91.4
8 //Bo=11
9 //Pt=100
10 //GT=-6.7
11 clear;
12 close;
13 clc;
```

```
14 F=input("enter the value of frequency in Ghz F");
15 c=3e8;
16 K=-228.6;//dB Boltzman constant
17 lambda= c/(F*1e9);
18 R=input("enter the value of distance between earth
     and satellite=");
19 FSL=(4*%pi*R*1e3/lambda);//free space losses
20 FSLdb=20*log10(FSL);
21 saidb=input("enter the value of saturation flux
     density of transponder in dB=");
22 A0db=-10*log10(4*%pi/(lambda^2));
23 EIRPdb=saidb+FSLdb+A0db;
24 //disp (EIRPdb,"earth station EIRP  required for
     saturation of transponder dB=");
25 Bo=input("enter the value of backoff=");
26 GT=input("enter the value of G/T in dB/K");
27 //B=input("enter the value of bandwidth in Hz");
28 //Bdb=10*log10(B);
29 CNUdb=EIRPdb+GT-Bo-K-FSLdb//-Bdb;
30 disp ('dB',CNUdb,"Sinal to noise density ratio
     uplink=");
```

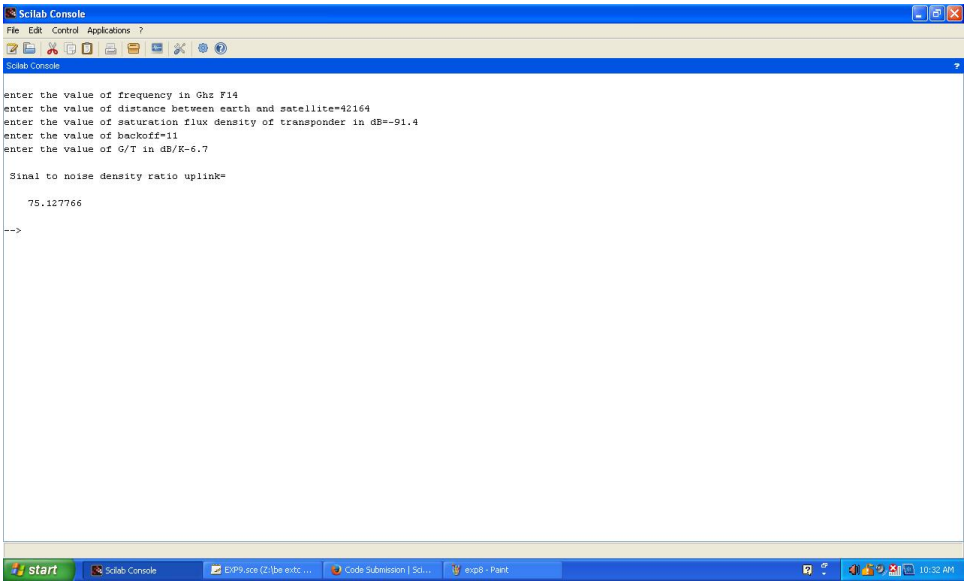

Figure 9.1: Link3

Experiment: 10

To calculate total C/N after calculating (C/N)Downlink and (C/N)uplink.

Scilab code Solution 10.10 Total SNR

```
1 //For given signal to noise ratio of uplink and down
      link  calculate total signal to noise ratio of
     satellite link
2 //OS=Windows XP sp3
3 //Scilab version 5.4.0
4 //Sample Values
5 //CNUdb=26
6 //CNDdb=24
7 clear;
8 close;
9 clc;
10 //for given data of downlink calculate CND and CNT
11 CNUdb=input("enter the value of uplink signal to
     noise density ratio in dB=");
12 CNDdb=input("enter the value ofdownlink signal to
     noise density ratio in dB=");
13 //CNIdb=input("enter the value of inteference signal
```

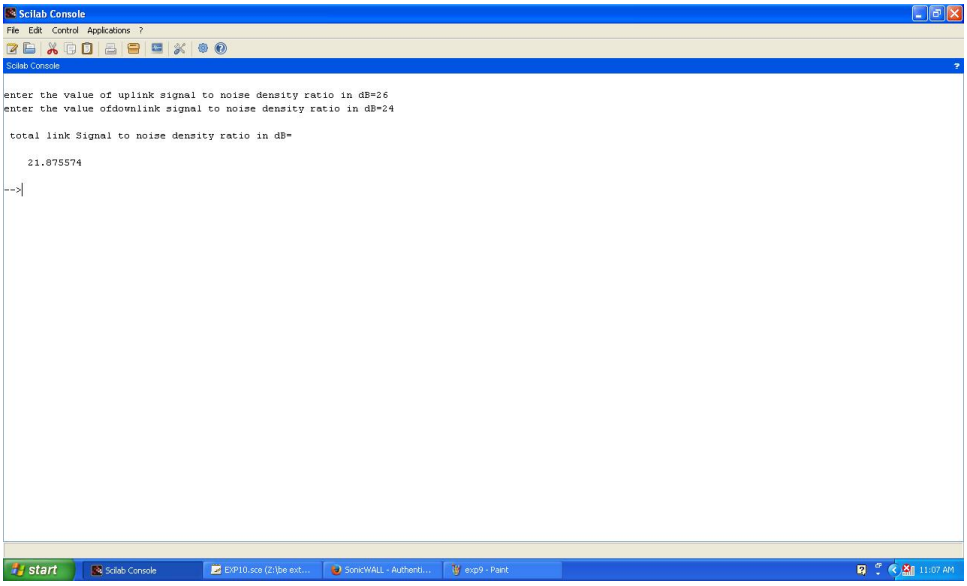

Figure 10.1: Total SNR

```
    to noise density ratio in dB=");
14 CND=10^(CNDdb/10);
15 CNU=10^(CNUdb/10);
16 //CNI=10^(CNIdb/10);
17 CNT=1/((1/CND)+(1/CNU)) // can add this term if
    interfernece is there (1/CNI));
18 CNTdb=10*log10(CNT);
19 disp (CNTdb,"total link Signal to noise density
    ratio in dB=");
```

Experiment: 11

To calculate total noise temperature for any given satelite communication link.

Scilab code Solution 11.11 Noise Temperature

```
1 //find equivalent noise temperature of N stage
    receiver here for lowest equivalent noise
    temperature we r assuming first stage is LNA and
    second stage is waverguide and subsequent stage
    are amplifiers for which noise figure and gain r
    specified
2 //OS=Windows XP sp3
3 //Scilab version 5.4.0
4 //Sample Values
5 //N=3
6 //Tant=35
7 //Te1=150
8 //G(1)=50
9 //L=5
10 //F(3)=12
11 //G(3)=5
12 clear;
```

```
close;
clc;
N=input("enter the No.of stages");
Tant=input("enter the antenna temperature");
Te1=input("enter the equivalent noise temperature of
    LNA");
G(1)=input("enter the gain of LNA in dB");
Ga(1)=10^(G(1)/10);
L=input("enter the attenuation of waveguide in dB");
La=10^(L/10);
To=290;
Te(2)=Tant+Te1+To*(L-1)/G(1)//gain till second stage
Ga(2)=Ga(1)/La;
for i=3:1:N//starting from 3rd stage to N stage
F(i)=input("enter the noise figure of next stage in
   dB");
Fa(i)=10^(F(i)/10)
G(i)=input("enter the gain of next stage amplifierin
    dB ");
Ga(i)=10^(G(i)/10)
Te(i)=Te(i-1)+To*(Fa(i)-1)/(Ga(i-1)*Ga(i));
end
X=Te(i);
disp('K',X,"Equivalent noise temperature");
```

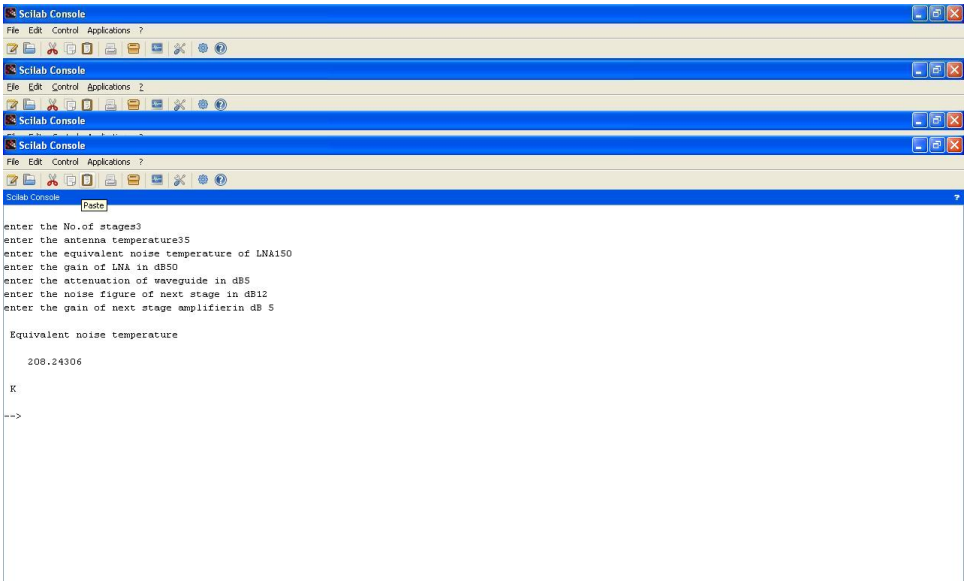

Figure 11.1: Noise Temperature