Scilab Textbook Companion for Microwaves and Radar Principles and Applications
by A. K. Maini[1]

Created by
Pasupulati Guruarun
B.Tech
Electrical Engineering
Sastra University
College Teacher
Prof. K. Narasimhan
Cross-Checked by
Chaitanya Potti

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: Microwaves and Radar Principles and Applications

Author: A. K. Maini

Publisher: Khanna Publishers, New Delhi

Edition: 3

Year: 2004

ISBN: 81-7409-129-7

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

Chapter 1

Introduction To Microwaves

Scilab code Exa 1.1 Finding dielectric constant of medium

```
// Chapter 1 example 1
// ------------------------------------------------------------------------------
clc;
clear;

// Given data
R   = 1.2;      // ratio of free space wavelength of
    a microwave signal to its wavelength when prop.
   through a dielectric medium

// Calculations
// lamda = lamda0/sqrt(er);
// er    = (lamda0/lamda)^2;
// let lamda0/lamda  = R

er  = (R)^2;         // Dielectric constant of medium

// Output
mprintf('The Dielectric constant of medium = %3.2f',
```

```
   er );
//
   ______________________________________________________________________________
```

Scilab code Exa 1.2 Finding height of antenna

```
// Chapter 1 example 2
//
   ______________________________________________________________________________

clc;
clear;
// Given data
Rmax        = 112;      // Max permissable range in
   Kms
H1          = 256;      // Ht of the antenna in m
// Calculations
// Rmax  = 4(sqrt(H1) + sqrt(H2));
// H2    = ((Rmax/4)-sqrt(H1))^2;
H2    = ((Rmax/4)-sqrt(H1))^2;      // Ht of other
   antenna
// Output
mprintf('Height of other antenna = %d m',H2);
//
   ______________________________________________________________________________
```

Chapter 2

Maxwells Equations

Scilab code Exa 2.1 `Finding magnetic field intensity`

```
// chapter 2 example 1
//  ____________________________________________________________________________

clc;
clear;
//   r1   = 3;      // relative permeability of region 1
//   r2   = 5;      // relative permeability of region 2
//   H1 = (4ax + 3ay -6az)A/m;    Magnetic field intensity
// Therefore B1 =    or1H1
//                  =  o (12ax + 9ay -18az)A/m
// since normal component of (B) is continuous across the interface
// Therefore, B2 =  o [12ax + 9( r2 / r1 )ay -18( r2 / r1 )az]
//                   =  o [12ax + 15ay - 30az]
//                H2 = [12/5ax + 15/5ay - 30/5az]A/m
//                H2 = (2.4ax + 3ay - 6az)
```

```
H2       = sqrt(2.4^2 + 3^2 + 6^2);

// output
mprintf('Magnetic field intensity in region- 2 = %3
    .2f A/m',H2);
//
    ____________________________________________________________________________
```

Scilab code Exa 2.2 `finding expressions of B and H`

```
// chapter 2 example 2
//
    ____________________________________________________________________________
clc;
clear;
// ur1 = 3
// ur2 = 5
// B1 = 2ax + ay
// choosing the unit normal an = (ay + az)/  2
// |Bn1| = ((2ax + ay)*(ay + az))/  2   = 1/  2
//Therefore Bn1 = 1/  2an   = (1/  2  )*(ay + az)/  2
// Also, Bn2 = Bn1 = 0.5ay + 0.5az
// the tangential component of B1 is given by
// Bt1 = B1 - Bn1 = (2ax + ay)-(0.5ay + 0.5az)
//      = 2ax + 0.5ay - 0.5az
// this gives Ht1 = (1/  o  )((2/3)ax + (0.5/3)ay -
    (0.5/3)az)
// Ht1  = (1/  o  )*(0.66ax + 0.16ay  -0.16az) = Ht2
// Bt2  =    or2Ht2    = 3.3ax + 0.8ay - 0.8az
// now B2 = Bn2 + Bt2 = (0.5ay + 0.5az)+(3.3ax + 0.8
    ay - 0.8az)
//         = (3.3ax +1.3ay - 0.3az)
// H2   = (1/  o  )*((3.3/5)ax + (1.3/5)ay - (0.3/5)az
```

```
)
// H2    = (1/  o  )*(0.66ax  +0.26ay −  0.06az)
mprintf('B2 = (3.3ax  +1.3ay −  0.3az)\n H2 = (1/  o  )*(0.66ax  +0.26ay −  0.06az)' );
//____________________________________________________________________________
```

Scilab code Exa 2.7 `Finding Amplitude of Displacement current density`

```
// chapter 2 example 7
//____________________________________________________________________________

clc;
clear;

//          | ax              ay                  az   |
// v*H =    |     /  x                 /  y                 /  z    |
//          | 0   10^6  *cos(377t + 1.2566*10^-6z)  0|
//      =      /  z  (10^6  *cos(377t + 1.2566*10^-6z))ax
//      = -1.2566*10^-6  *10^6sin(377t  +1.2566*10^-6  z )
//      = -1.2566sin(377t + 1.2566*10^-6z)ax
mprintf('Amplitude of displacement current density = 1.2566 A/m^2');
//____________________________________________________________________________
```

Scilab code Exa 2.8 `Finding amplitude of displacement current density`

```
// chapter 2 example 8
// ------------------------------------------------------------------------
clc;
clear;

//          | ax        ay                    az                 |
// v*E =    |     /  x          /  y                        /  z      |
//          | 0         0   80*cos(6.277*10^8t - 2.092y)     |
// Electric flux density D =    oE
//        = 8.85*10^-12  *80cos(6.277*10^8t - 2.092y)ax
//        = 708*10^-12  *cos(6.277*10^8 t  -2.092y)ax
// Displacement current density =   D  /  t
//   D  /  t    = -708*10^-12*6.277*10^8*sin(6.277*10^8 t - 2.092y)ax
//          = -0.444sin(6.277*10^8t - 2.092y)ax
mprintf('Amplitude of displacement current density = 0.0444 A/m^2');
// ------------------------------------------------------------------------
```

Scilab code Exa 2.9 Finding electric and magnetic field intensity

```
// chapter 2 example 9
// ------------------------------------------------------------------------
clc;
clear;

```

```
// A    = (10^-3 y cos(3*10^8 t)cosz)az
// V    = 3*10^5 y sin(3*10^8 t)sinz volts
uo      = 4*%pi*10^-7
ur      = 1;
er      = 1;

//           | ax      ay                  az       |
// v*A =     |    / x      / y                   / z    |
//           | 0       0    (10^-3 y cos(3*10^8 t)cosz)   |
//      =     / y (10^-3 y cos(3*10^8 t)cosz)ax
//      = 10^-3 ax cos(3*10^8t)cosz
// H    = B/(  or  )
// H    = (10^-3 ax cos(3*10^8t)cosz)/( 4*%pi*10^-7)
// H    = 796axcos(3*10^8 t)cosz
// Electric intensity can be computed from
// E    = - V V -  A / t
// Now V V =  V / x  ax +  V / y  ay +  V / z  az
//  = 3*10^5 sin 3*10^8 t sinz + 3*10^5 y sin3*10^8 t cosz
//  A / t  = -10^-3 * 3*10^8 y sin 3*10^8t cosz
// E     = 3*10^5 sin 3*10^8 t sinz + 3*10^5 y sin3*10^8 t cosz + 3*10^5 y sin 3*10^8t cosz
// E     = -3*10^5 sin 3*10^8tsinz
mprintf('magnetic field intensity = 796axcos(3*10^8 t)cosz\n Electric field intensity = -3*10^5 sin 3*10^8tsinz')
```

Scilab code Exa 2.10 Finding amplitude of displacement current density

```
// chapter 2 example 10
//
```

```
---------------------------------------------------------------------------------
clc;
clear;
// given data
// D  = 3*10^-7 sin(6*10^7 - 0.35x)az
er = 100;        // relative permitivity

// Calculations
//  D / t  =  3*10^-7 * 6*10^7* cos(6*10^7 - 0.35x)az
A =  3*10^-7 * 6*10^7

// output
mprintf('Amplitude of displacement current density = %d A/m^2',A);
// ---------------------------------------------------------------------------------
```

Scilab code Exa 2.12 Finding beta and Hm

```
// chapter 2 example 12
// ---------------------------------------------------------------------------------
clc;
clear;
// given data
// E = 40  e^j(10^9t +  z )ax
// H = Hm e^j(10^9t +  z )ay
// w/  = 1/sqrt(e*uo) = 3*10^8
w   = 10^9;       // from given expression
b   = w/3*10^8
Em  = 40*%pi      // from given expression
```

```
// E/H = 120;          // for free space

Hm   = Em/(120*%pi);
//V * E = -  B  /  t
//        =|      ax              ay               az|
// V*E   =|        /  x                  /  y               /  t  |
//        =|40    e^j(10^9t +  z ) 0           0   |
// V*E  = j  40    e^j(10^9t +  z ) ay
                                         ----- 1
//  - B  /  t   = uo*  H  /  t   = -j*10^9  *uo*Hm e^j (10^9t +  z )ay
                                   ----- 2
// Comparing 1 and 2 shoes that Hm must be negative Hence Hm = -1/3 A/m
mprintf('Hm = - %3.2f A/m',Hm);
// ------------------------------------------------------------------------------------------
```

Scilab code Exa 2.13 Finding w and Hm

```
// chapter 2 example 13
// ------------------------------------------------------------------------------------------

clc;
clear;
// given data
// E = 20    e^j(wt -  z )ax
// H = Hm e^j(wt +  z )ay
lamda = 1.8;          // wavelength in m
c     = 3*10^8;       // vel. in m/s
er    = 49;           // relative permitivity
ur    = 1;            // relative permeability
Em    = 20*%pi        // from the given expression
```

```
// Calculations
v      = c/sqrt(er);  // velocity of propagation of wave in medium with er rel.permitivity
w      = (2*%pi*v)/lamda;
// let k = E/H
k = (120*%pi)*sqrt(ur/er);
Hm = Em/k
// sign of Hm can be determined by evaluating the maxwells eqn
// V*E =   B  /  x
// V*E = -j20   e^j(wt -   z  )ay ------------------ 1
// - B  /  x   = -juow Hm e^j(wt +   z  )ay ------------------ 2
// comparing 1 and 2 singn of Hm must be positive
mprintf('w = %3.1e rad/s\n Hm = %3.2f A/m',w,Hm);
// ------------------------------------------------------------------------------
```

Scilab code Exa 2.14 Finding amplitude of displacement current density

```
// chapter 2 example 14
// ------------------------------------------------------------------------------
clc;
clear;
// given data
f         = 1000;       // frequency in Hz
sigma     = 5*10^7;     // conductivity in mho/m
er        = 1;          // relative permitivity
eo        = 8.85*10^-12;   // permitivity
//J       = 10^8sin(wt-444z)ax A/m^2

```

```
// Calculations
w        = 2*%pi*f
// J  =  E
// E  = 10^8 sin(wt-444z)ax/sigma
// E  = 0.2 sin(6280t-444z)ax
// D  = eoerE
// D  = 8.85*10^-12*0.2 sin(6280t-444z)ax
//   D / t   = 1.77*10^-12*6280 cos(6280t - 444z)ax
A      =  1.77*10^-12*6280
mprintf('Amplitude of displacement current density = 
    %3.2e A/m^2',A);
mprintf('\n Note: calculation mistake in textbook');
//
   ------------------------------------------------------------------------------------
```

Chapter 3

Transmission Media Transmission lines and Waveguides

Scilab code Exa 3.2 Finding reflection coefficient and SWR

```
// Chapter 3 example 2
//
   ------------------------------------------------------------------------------------------------------------
clc;
clear;
// Given data
Lr      = 18;        // return loss in db
// Calculations
// Lr   = 20*log(1/p);
p       = 1/10^(Lr/20);      // reflection co-
   efficient
swr     = (1 + p)/(1 - p);       // standing wave
   ratio
// Output
mprintf('Reflection co-efficient is %3.3f\n SWR = %3
   .2f',p,swr);
```

```
//
    ________________________________________________________________________
```

Scilab code Exa 3.3 Finding min length of cable

```
// Chapter 3 example 3
//
    ________________________________________________________________________

clc;
clear;
// Given data
PW  = 30*10^-6;      // pulse width in sec
ips = 20*10^-6;      // inter pulse separation
v   = 3*10^8;        // propagation speed in m/s

// Calculations
T   = PW+ips+PW+ips+PW       // time duration of the
    pulse train for having 3 pulses on the line at a
    time
l   = v*T;                   // minimum length of
    cable required

// Output
mprintf('Minimum length of cable required = %d km',l
    /1000);
//
    ________________________________________________________________________
```

Scilab code Exa 3.4 Finding reflection coefficient and characteristic impedance

```
// Chapter 3 example 4
// ------------------------------------------------------------------------------
clc;
clear;
// Given data
RmsVmax     = 100;       // max value of RMS vtg
RmsVmin     = 25;        // min value of RMS vtg
Zl          = 300;       // load impedance in ohm

// Calculations
VSWR        = RmsVmax/RmsVmin;
// wkt VSWR  = Zl/Zo; assuming Zl > Zo
Zo          = Zl/VSWR;    // charecteristic impedance in ohm
p           = (Zl - Zo)/(Zl + Zo);    // reflection co-efficient

// Output
mprintf('Reflection Co-efficient = %3.1f\n Charecteristic impedance = %d ohm',p,Zo);
// ------------------------------------------------------------------------------
```

Scilab code Exa 3.5 Finding load resistance reflection coefficient and power

```
// Chapter 3 example 5
// ------------------------------------------------------------------------------
clc;
clear;
// Given data
```

```
Zo      = 75;        // charecteristic impedance in ohm
Vref    = 100;       // reflected voltage
Pref    = 100;       // reflected power in watts

// Calculations
Zl      = (Vref)^2 /Pref            // load impedance
p       = (Zl - Zo)/(Zl + Zo);    // reflection co-efficient
Pinc    = Pref/p                    // incident power
Pobs    = Pinc - Pref               // power obsorbed

// Output
mprintf('Load Resistance = %d ohm\n Reflection Co-efficient = %3.3f\n incident power = %d watts\n power obsorbed = %d watts',Zl,p,Pinc,Pobs);
// ---------------------------------------------------------------------------
```

Scilab code Exa 3.6 Finding length of line and characteristic impedance

```
// Chapter 3 example 6
// ---------------------------------------------------------------------------
clc;
clear;
// Given data
c      = 3*10^8;        // velocity in m/s
f      = 100*10^6       // operating frequency in hz
Zin    = 100;
Zl     = 25;

// Calculations
```

```

lamda   = c/f              // wavelength in m
Lreq    = lamda/4;         // required length in m
Zo      = sqrt(Zin*Zl);    // charecteristic impedance in ohm

// Output
mprintf('Length of line required = %d cm\n Charecteristic impedance = %d ohm',Lreq*10^2,Zo);
// ________________________________________________________________
```

Scilab code Exa 3.7 Finding input impedance

```
// Chapter 3 example 7
// ________________________________________________________________

clc;
clear;
// in the first case when the line is lamda/2 long, the i/p impedance is same as the load resistance
Zl      = 300;      // load resistance in ohm
Zo      = 75;       // charecteristic impedance in ohm

// calculations
//Zi      = Zo*((Zl + iZotan l)/(Zo + iZltan l))
//        = Zo*(((Zl/tan l) + iZo))/((Zl/tan l) + iZo)))
// for l = lamda/2  l  = (2*  /lamda)*(lamda/2) =

// therefore tan l = 0 which gives Zi = Zl
// in the second case when the operating frequency
```

```
   is halved , the wavelength is dou led which means
    the same line is now lamda/4 long
// for l = lamda/4 , l  = (2*   /lamda)*(lamda/4) =
   /2
// therefore tan l =
Zi          = (Zo^2)/Zl;  // input impedance

mprintf('Input impedance = %3.2f ohm',Zi);
//
  ------------------------------------------------------------------------------
```

Scilab code Exa 3.8 `Finding expressions for Vin and Vl`

```
// Chapter 3 example 8
//
  ------------------------------------------------------------------------------

clc;
clear;
// Given data
f        = 100*10^6;             // operating
   frequency in Hz
v        = 2*10^8 ;              // propagation
   velocity in m/s
Zo       = 300;                  // charecteristic
   impedance in ohm
Zin      = 300;                  // input impedance
   in ohm
l        = 1;                    // length in m
V        = 100;

// Calculations
lamda    = v/f                   // wavelength in m
if lamda/2 ==l then
```

```
    Zl   = Zin;
end
k        = (V*Zin)/(Zin+Zl)
//Vin    = k*cos(2*%pi*f*t)
// since the line is lamda/2 long ,the signal undergoes a phase delay of  l  = (2*  )/lamda *(lamda/2) =
// Output
mprintf('Vin = %dcos(2  *%3.0et)\n Vl = %dcos(2  *%3.0et-  )',k,f,k,f );
// ----------------------------------------------------------------------------
```

Scilab code Exa 3.9 Finding magnitude of reflection coefficient and frequency of o

```
// Chapter 3 example 9
// ----------------------------------------------------------------------------
clc;
clear;
// Given data
VSWR     = 3;          // voltage standing wave ratio
d        = 20*10^-2    // separation b/w 2 successive minimas
er       = 2.25;       // dielectric constant
v        = 3*10^8;     // velocity in m/s

// Calculations
// VSWR = (1 + p)/(1 - p)
p        = (VSWR -1)/(VSWR + 1);      // reflection co-efficient
lamda    = 2*d;                       // wavelength of tx line
```

```
lamda_fr= lamda*sqrt(er);              // free space wavelength
f        = v/lamda_fr;                 // operating frequency in Hz

// output
mprintf('Magnitude of Reflection Co-efficient = %3.1f\n Frequency of Operation = %3.0f Mhz',p,f/10^6);
// ____________________________________________________________________________
```

Scilab code Exa 3.10 Finding per unit inductance Zo phase shift constant and refle

```
// Chapter 3 example 10
// ____________________________________________________________________________

clc;
clear;
// Given data
C    = 30;             // per unit capacitance in pF/m
Vp   = 260;            // velocity of propagation in m/us
f    = 500*10^6        // freq in Hz
Zl   = 50;             // terminating load impedance in ohm

// calculations
v    = Vp/10^-6        // conversion from m/us to m/s
C1   = C*10^-12        // conversion from pF/m to F/m
// 1/sqrt(LC) = Vp
L    = 1/(v^2 * C1);    // per unit inductance
Zo   = sqrt(L/C1);       // charecteristic impedance in
```

```
ohm
lamda = v/f             // wavelength
b     = (2*%pi)/lamda   // phase shift constant
p     = (Zl - Zo)/(Zl + Zo);       // Reflection coefficient

// Output
mprintf('Per Unit inductance = %d nH/m\n Charecteristic Impedance = %d ohm\n Phase shift Constant = %d rad/m\n Reflection co-efficient = %3.3f',L*10^9,Zo,b,abs(p));
// ----------------------------------------------------------------------------------------------------------
```

Scilab code Exa 3.11 `showing certain freq passing through waveguide`

```
// Chapter 3 example 11
// ----------------------------------------------------------------------------------------------------------
clc;
clear;
// Given data
a    = 1.5*10^-2;         // width of waveguide
b    = 1*10^-2            // narrow dimension of waveguide
er   = 4;                 // dielectric constant
f    = 8*10^9;            // frequency in Hz
c    = 3*10^8             // velocity in m/s

// calculations
lamda_c      = 2*a;       // cut-off wavelength for TE10 mode
lamda        = c/f        // wavelength corresponding
```

```
    to given freq.
lamda_d      = lamda/sqrt(er);    // wavelength when waveguide filled with dielectric
if lamda_d < lamda_c then
    mprintf('8 Ghz frequency will pass through the guide');
end
```

Scilab code Exa 3.12 Finding min frequency

```
// Chapter 3 example 12
//________________________________________________________________________________

clc;
clear;
// Given data
a    = 4*10^-2;            // width of waveguide
b    = 2*10^-2             // narrow dimension of waveguide
er   = 4;                  // dielectric constant
c    = 3*10^8              // velocity in m/s

// Calculations
lamda_c      = 2*a;        // max cut-off wavelength
fcmin        = c/lamda_c // min freq
lamda_d      = lamda_c/sqrt(er);      // wavelength if we insert dielectric
fc           = c/lamda_d              // min frequency in presence of dielectric

// Output
mprintf('Minimum Frequency that can be passed with dielectric in waveguide is %3.1f Ghz',fc/10^9);
//
```

```
________________________________________________________________________________
```

Scilab code Exa 3.13 `Showing certain frequency does not pass through waveguide`

```
// Chapter 3 example 13
//
________________________________________________________________________________
clc;
clear;
// Given data
f    = 1*10^9;          // frequency in Hz
a    = 5*10^-2;         // wall separation
c    = 3*10^8;          // velocity of EM wave in m/s
m    = 1;               // for TE10
n    = 0;               // for TE10

// Calculations
// lamda0    = 2/sqrt((m/a)^2 + (n/b)^2)
lamda0       = (2*a)/m
lamda_frspc = c/f;
if lamda_frspc > lamda0 then
    mprintf('1 Ghz signal cannot propagate in TE10
        mode')
else
    mprintf('1 Ghz signal can propagate in TE10 mode
        ');
end
```

Scilab code Exa 3.14 `Finding longest cutoff wavelength`

```
// Chapter 3 example 14
```

```
//
    ------------------------------------------------------------------------------------

clc;
clear;
// Given data
a   = 30;              // width of waveguide
b   = 20;             // narrow dimension of waveguide
c   = 3*10^8;         // velocity of EM wave in m/s
m   = 1;              // for TE10
n   = 0;              // for TE10

// Calculations
// lamda0   = 2/sqrt((m/a)^2 + (n/b)^2)
lamda0      = (2*a)/m;       // longest cut-off
   wavelength in dominant mode TE10

// Output
mprintf('longest cut-off wavelength = %d mm',lamda0
   );
//
    ------------------------------------------------------------------------------------
```

Scilab code Exa 3.15 `Finding frequency range`

```
// Chapter 3 example 15
//
    ------------------------------------------------------------------------------------

clc;
clear;
// Given data
a   = 4*10^-2;       // width of waveguide
b   = 2*10^-2;       // narrow dimension of waveguide
```

```
c    = 3*10^8;          // velocity of EM wave in m/s
m1   = 1;               // for TE10
m2   = 2;               // for TE20
n    = 0;               // for TE10
// Calculations
lamda_c     = 2*a        // cutoff wavelength for TE10 mode
f1          = c/lamda_c // frequency in Hz
// the frequency range for single mode operation is the range of frequencies corresponding to the dominant mode and the second highest cutoff wavelength
lamda_c_2   = 2/sqrt((m2/a)^2 + (n/b)^2)
f2          = c/lamda_c_2;    // freq at second largest cutoff wavelength

// Output
mprintf('Therefore,single mode operating range = %3.2f Ghz to %3.1f Ghz\n',f1/10^9,f2/10^9 );
mprintf(' Note: instead of 3.75,3.5 is printed in textbook');
// ____________________________________________________________________________
```

Scilab code Exa 3.16 Finding group and phase velocity

```
// Chapter 3 example 16
// ____________________________________________________________________________
clc;
clear;
// Given data
a    = 7.2         ;      // width of waveguide in cm
```

```
b   = 3.4;          // narrow dimension of waveguide in cm
c   = 3*10^10;        // free space velocity of EM wave in cm/s
f   = 2.4*10^9;      // frequency in Hz

// Calculation
lamda   = c/f        // free space wavelength in cm
lamda_c = 2*a        // cutoff wavelength in cm
lamda_g = lamda/sqrt(1 - (lamda/lamda_c)^2); // guide wavelength in cm
vp      = (lamda_g * c)/lamda                // phase velocity in cm/s
vg      = c^2/vp;                            // group velocity in cm/s

// Output
mprintf('Group velocity = %3.1e cm/s\n Phase Velocity = %3.1e cm/s',vg,vp);
// -----------------------------------------------------------------------------
```

Scilab code Exa 3.18 proof

```
// Chapter 3 example 18
// -----------------------------------------------------------------------------
clc;
clear;
// let 'a' and 'b' be the broad and narrow dimensions of the rectangular guide and 'r' be internal radius of circular guide
// Dominant mode in rectangular guide =TE10
```

```
// cutoff wavelength = 2a
// dominant mode in circular guide = TE11
// cut-off wavelength = 2*pi*r/1.841  = 3.41r
// for the two cut-off wavelengths to equal
// 2a  = 3.41r
// a = 1.705r
// now area of cross section of rectangular guide = a*b
//assuming a= 2b,which is very reasonable assumption ,we get
// area of cross section of rectangular waveguide = a*a/2 = ((1.705^2)*r*r)/2 = 1.453r^2
// area of cross-section of circular guide = pi*r*r = 3.14r^2
// ratio of two cross sectional areas = (3.14r^2)/(1.453r^2) = 2.16
mprintf('Circular guide is 2.16 times larger in cross section as compared to rectangular guide');
// ------------------------------------------------------------------------
```

Scilab code Exa 3.19 Finding all the possible modes that will propagate in a waveg

```
// Chapter 3 example 19
// ------------------------------------------------------------------------
clc;
clear;
// Given data
a   = 4*10^-2;      // width of waveguide
b   = 2*10^-2;      // narrow dimension of waveguide
c   = 3*10^8;       // velocity of EM wave in m/s
f   = 5*10^9        // operating frequency in Hz
```

```
m0   = 0;              // for TE01
m1   = 1;              // for TE10 / TE11 /TM11
n0   = 0;             // for TE10
n1   = 1;              // for TE11 or TM11
// Calculations
lamda          = c/f;                           // operating wavelength
lamda_TE01     = 2/sqrt((m0/a)^2 + (n1/b)^2)   // cutoff wavelength for TE01
lamda_TE10     = 2/sqrt((m1/a)^2 + (n0/b)^2)   // cutoff wavelength for TE10
lamda_TE11     = 2/sqrt((m1/a)^2 + (n1/b)^2)   // cutoff wavelength for TE11 or TM11
if lamda_TE01 >lamda then
    mprintf('TE01 propagates in the given guide at the given operating frequency');
    elseif lamda_TE10 >lamda then
    mprintf('TE10 propagates in the given guide at the given operating frequency');
    elseif lamda_TE11 >lamda then
    mprintf('TE11 propagates in the given guide at the given operating frequency');
end
```

Scilab code Exa 3.20 Finding frequency of wave

```
// Chapter 3 example 20
//------------------------------------------------------------------------------
clc;
clear;
// Given data
a    = 4*10^-2;        // width of waveguide
b    = 2*10^-2;        // narrow dimension of waveguide
```

```
c    = 3*10^8;          // velocity of EM wave in m/s
d    = 4*10^-2;         // distance b/w field maxima and minima
// Calculations
lamda_c      = 2*a;        // cut-off wavelength in dominant mode
lamda_g      = 4*d;        // guide wavelength
// lamda_g  = lamda0/(sqrt(1 - (lamda0/lamda_c)^2))
lamda0       = sqrt((lamda_c * lamda_g)^2 / (lamda_c^2 + lamda_g^2));
f0           = c/lamda0;       // frequency of the wave

// Output
mprintf('Frequency of the wave = %3.3f Ghz',f0/10^9);
// ______________________________________________________________________
```

Scilab code Exa 3.21 computing guide wavelength phase shift constant and phase vel

```
// Chapter 3 example 21
// ______________________________________________________________________

clc;
clear;
// Given data
a    = 6;        // width of waveguide in cm
b    = 3;        // narrow dimension of waveguide in cm
lamda = 4;      // operating wavelength in cm
c    = 3*10^8;          // velocity of EM wave in cm/s

// Calculations
lamda_c   = 2*a;        // cut-off wavelength in
```

```
dominant mode
lamda_g  = lamda/(sqrt(1 - (lamda/lamda_c)^2)) // guide wavelength
Vp       = (lamda_g/lamda)*c
b        = (2*%pi)/lamda_g;      // phase shift constant

// Output
mprintf('Guide wavelength = %3.2f cm\n Phase velocity = %3.2e m/s\n Phase shift constant = %3.2f radians/cm',lamda_g,Vp,b)
//
```

Scilab code Exa 3.22 computing cutoff freq phase velocity and guided wavelength

```
// Chapter 3 example 22
//

clc;
clear;
// Given data
er       = 9;          // relative permittivity
c        = 3*10^10;    // velocity of EM wave in free space
f        = 2*10^9;     // operating frequency in Ghz
a        = 7;        // width of waveguide in cm
b        = 3.5;      // narrow dimension of waveguide in cm

// calculations
lamda_c  = 2*a;                    // cut-off wavelength in dominant mode
```

```
fc        = c/lamda_c             // cut-off frequency
    in Hz
lamda     = c/(sqrt(er)*f);       // operating
   wavelength
lamda_g   = lamda/(sqrt(1 - (lamda/lamda_c)^2)) //
   guide wavelength
Vp        = (lamda_g/lamda)*c

// Output
mprintf('Cut-off frequency = %3.3f Ghz\n Phase
   velocity = %3.2e m/s\n Guide wavelength = %3.2f
   cm',fc/10^9,Vp/10^2,lamda_g);
//
   ------------------------------------------------------------------------
```

Chapter 4

Microwave Components

Scilab code Exa 4.1 Finding power at coupled port

```
// chapter 4 example 1
//
   ____________________________________________________________________________

clc;
clear;
// given data
Pi      = 10;              // Input power in mW
CF      = 20;              // coupling factor in dB

// calculations
// CF(db)    = 10log(Pi/Pc)
Pc      = Pi/(10^(CF/10))  // antilog conversion and
    coupling power

// Output
mprintf('Coupled Power = %d uW',Pc*10^3);
//
   ____________________________________________________________________________
```

Scilab code Exa 4.2 `Finding power available at the straight through port output`

```
// chapter 4 example 2
//------------------------------------------------------------------------------
clc;
clear;
// given data
Pi      = 10;            // Input power in mW
IL      = 0.4;           // insertion loss in dB
// calculations
// ILdb)   = 10log(Pi/Po)
Po      = Pi/(10^(IL/10))  // antilog conversion and coupling power

// Output
mprintf('Power available at the straight through port output = %3.3f mW',Po);
//------------------------------------------------------------------------------
```

Scilab code Exa 4.3 `Finding directivity power at isolated port`

```
// chapter 4 example 3
//------------------------------------------------------------------------------
clc;
clear;
// given data
```

```
CF      = 20;        // Coupling factor in dB
I       = 50;        // Isolation in dB
Pc      = 100*10^-6; // coupling power in W

// calculations
// D    = 10log(Pc/Piso)
D       = I - CF;              // Directivity in dB
Piso    = Pc/(10^(D/10))  // antilog conversion and coupling power

// Output
mprintf('Directivity = %d dB\n Power at isolated port = %d nW',D,Piso*10^9);
// ____________________________________________________________________
```

Scilab code Exa 4.4 Finding power available at output port

```
// chapter 4 example 4
// ____________________________________________________________________
clc;
clear;
// given data
CF      = 20;              // coupling factor in dB
D       = 30;              // Directivity in dB
Pin     = 10;              // input power in dBm

// Calculations
// 10logPi = Pin
Pi      = 10^(Pin/10);    // power in mW
I       = D + CF          // isolation in dB
Pc      = Pin - CF;
```

```
Pcwatts = 10^(Pc/10)       // power at coupled port in mW
Piso    = Pin - I
Pisowatts = 10^(Piso/10) // Power at isolated port in mW
Po      = Pi -(Pcwatts + Pisowatts);     // power at o/p port in mW

// Output
mprintf('Power Available at the output port = %3.5f mW',Po);
// ______________________________________________________________________________
```

Scilab code Exa 4.5 Finding directivity

```
// chapter 4 example 5
// ______________________________________________________________________________

clc;
clear;
// given data
Pi      = 5*10^-3;               // Input power in W
CF      = 10;                   // coupling factor
Piso    = 10*10^-6              // power at isolated port in w
// calculations
// CF   = 10log(Pi/Pc)
Pc      = Pi/(10^(CF/10))  // antilog conversion and coupling power
// D    = 10log(Pc/Piso)     // Directivity
D       = 10*log10(Pc/Piso)
// Output
```

```
mprintf('Directivity = %3.0f dB\n',D);
//
    ------------------------------------------------------------------------
```

Scilab code Exa 4.6 Finding lowest resonant frequency

```
// chapter 4 example 6
//
    ------------------------------------------------------------------------

clc;
clear;
// given data
a       = 2;        // width in cm
b       = 1;        // Height in cm
d       = 3;        // length in cm
c       = 3*10^10;    // vel in free space in cm/s
// For TE101 mode
m       = 1
n       = 0;
p       = 1;

// Calculations
fo      = (c/2)*sqrt((m/a)^2 + (n/b)^2 + (p/d)^2);

// Output
mprintf('Resonant Frequency = %d Ghz',fo/10^9);
//
    ------------------------------------------------------------------------
```

Scilab code Exa 4.7 Finding resonant frequency

```
// chapter 4 example 7
//------------------------------------------------------------------------------
clc;
clear;
// given data
fo      = 10;        // resonant freq in Ghz
mprintf('The Resonant frequency for a TM mode in a
   rectangular cavity resonator for a given integral
   \n');
mprintf(' values of m,n and p is same as that of a
   TE mode for same values of m,n and p\n');
mprintf(' Therefore,TM111 mode resonant frequency =
   %d Ghz',fo);
//------------------------------------------------------------------------------
```

Scilab code Exa 4.8 Finding length of cavity resonator

```
// chapter 4 example 8
//------------------------------------------------------------------------------
clc;
clear;
// given data
a       = 4;         // width in cm
b       = 2;         // Height in cm
c       = 3*10^10;    // vel in free space in cm/s
fo      = 6*10^9;     // resonator frequency in Ghz
// For TE101 mode
m       = 1
n       = 0;
```

```
p       = 1;

// Calculations
//fo      = (c/2)*sqrt((m/a)^2 + (n/b)^2 + (p/d)^2);
d       = sqrt((p^2)/((2*fo/c)^2 - (m/a)^2 - (n/b)
   ^2));
// Output
mprintf('Length of cavity resonator = %3.1f cm',d);
//
   ------------------------------------------------------------------------------------------------
```

Scilab code Exa 4.9 Finding length of cavity resonator

```
// chapter 4 example 9
//
   ------------------------------------------------------------------------------------------------
// Note : some data from is problem is taken from
   Ex4.8
clc;
clear;
// given data
a       = 4;        // width in cm
b       = 2;        // Height in cm
c       = 3*10^10;   // vel in free space in cm/s
fo      = 6*10^9;    // resonator frequency in Ghz
d       = 3.2;       // length of cavity resonator
   in cm
// For TE101 mode
m       = 1
n       = 0;

// Calculations
lamda_c = 2/sqrt((m/a)^2 + (n/b)^2);        // cut-
```

```
off wavelength in m
lamda   = c/fo;                              // operating wavelength in m
lamda_g = lamda/sqrt(1 - (lamda/lamda_c)^2) // guide wavelength in m

mprintf('Length of resonator is %3.1f cm and  guide wavelength is %3.1f cm',d,lamda_g);
mprintf('\n length of resonator is half of guide wavelength');
//____________________________________________________________________________
```

Scilab code Exa 4.10 `Finding length of resonator`

```
// chapter 4 example 10
//____________________________________________________________________________

clc;
clear;
// given data
di       = 8;        // internal diameter in cms
a        = 4;        // internal radius in cms
fo      = 10*10^9;  // operating frequency in Ghz
ha01    = 2.405;    // Eigen value of bessel function
c       = 3*10^10   // velocity of EM wave in cm/sec
// For TM011 mode
m       = 0
n       = 1
p       = 1

// Calcultions
```

```
//f0   = (c/2*pi)*sqrt((ha/a)^2 + (p*pi/d)^2)
   operating frequency
d      = (p*%pi)/(sqrt((fo*2*%pi/c)^2 - (ha01/a)^2))
     //length of resonator

// Output
mprintf('Length of resonator = %3.3f cm',d);
//
   ----------------------------------------------------------------------------
```

Scilab code Exa 4.11 Finding resonant frequency

```
// chapter 4 example 11
//
   ----------------------------------------------------------------------------

clc;
clear;
// given data
di        = 6;        // internal diameter in cms
d         = 5;        // length in cm
a         = 4;        // internal radius in cms
fo       = 10*10^9;  // operating frequency in Ghz
ha01     = 2.405;    // Eigen value of bessel
   function
ha11     = 1.841;    // Eigen value of bessel
   function
c        = 3*10^10   // velocity of EM wave in cm/sec
// For TM011 mode and TE111 mode
m0        = 0
m1        = 1
n1        = 1
p1        = 1
p2        = 2
```

```
// Calcultions
f0  = (c/(2*%pi))*sqrt((ha01/a)^2 + (p2*%pi/d)^2)   //resonant frequency for TM012 mode
f01  = (c/(2*%pi))*sqrt((ha11/a)^2 + (p1*%pi/d)^2)   //resonant frequency for TE111 mode

// Output
mprintf('Resonant frequency for TM012 mode = %3.3f Ghz\n Resonant frequency for TM111 mode = %3.3f Ghz\n',f0/10^9,f01/10^9 );
//-------------------------------------------------------------------------
```

Chapter 5

Microwave Tubes

Scilab code Exa 5.1 Finding transit time of electron in repeller space

```
//chapter 5 example 1 pg no-226
//==================================================================

clc;
clear;
//Given Data
F      =  100*10^9;//reflex klystron operating frequency
n      =  3;//integer corresponding to mode

//Calculations
T_c    =  (n+(3/4))//transit time in cycles
T      =  T_c/F//transit time in seconds

//Output
mprintf('Transit Time of the electron in the repeller space is %3.1f ps',T/10^-12);

//==================================================================
```

Scilab code Exa 5.2 Finding change in frequency

```
//chapter 5 example 1 pg no-227
//==================================================================

clc;
clear;
//Given Data
F      =  2*10^9;//reflex klystron operating
    frequency
Vr     =  2000;//Repeller voltage
Va     =  500;//Accelarating voltage
n      =  1;//integer corresponding to mode
e      =  1.6*10^-19;//charge of electron
m      =  9.1*10^-31;//mass of electron in kg
s      =  2*10^-2;//space b/w exit of gap and
    repeller electrode
dVr1   = 2;//(change in Vr in percentage
//Calculations
dVr    = dVr1*Vr/100;//conversion from percentage to
    decimal
//dVr/df = ((2*pi*s)/((2*pi*n)-pi/2))*sqrt(8*m*Va/e)
    );
//let df = dVr/((2*pi*s)/((2*pi*n)-pi/2))*sqrt(8*m*
    Va/e));

df        =  (dVr)/((2*%pi*s)/((2*%pi*n)-(%pi/2))*
    sqrt(8*m*Va/e));//change in freq as a fun of
    repeller voltage


//Output
```

```
mprintf('Change in frequency is %3.0f MHz',df/10^6);

//
```

Scilab code Exa 5.3 Finding percentage change in frequency

```
//chapter 5 example 3
//

clc;
clear;
//Given Data
//let l  = dVr/Vr ; f  = df/f ; Vr/f  = R
l      = 5;//percentage change in repeller voltage
f      = 1;//percentage change in operating frequency
R      = 1;//ratio of repeller voltage to operating frequency
NR     = 1.5;//new ratio of repeller voltage to operating frequency in volts/MHz
e      =  1.6*10^-19;//charge of electron
m      =  9.1*10^-31;//mass of electron in kg

//Calculations

//dVr/df = ((2*pi*s)/((2*pi*n)-pi/2))*sqrt(8*m*Va/e));
//((df/f)/(dVr/Vr)) = (Vr/f)*((2*pi*n)-pi/2)/(2*pi*s)*sqrt(e/(8*m*Va));
//((df/f)/(dVr/Vr)) = K*(Vr/f);
//where K = (((2*pi*n)-pi/2)/(2*pi*s))*sqrt(e/(8*m*Va))
K     = (f/l)*(1/R)
```

```
PCF   = NR*K*1//percentage change in frequency when new ratio (Vr/f)=1.5

//Output
mprintf('Percentage Change in frequency is %3.2f percent',PCF);

//
```

Scilab code Exa 5.4 Finding electronic efficiency and output power

```
//chapter 5 example 4
//

clc;
clear;
//Given Data
Va    = 40*10^3;//Anode voltage of cross field amplifier
Ia    = 15;//Anode current in Amp
Pin   = 40*10^3;//input power in watts
G     = 10;//gain in dB
n     = 40/100;//overall efficiency converted from percentage to decimal
//Calculations
//Gain  = (1+(Pgen/Pin))
Pgen  = (G-1)*Pin//Generated power
ne    = (Pgen/(Va*Ia))//electronic efficiency
nc    = n/(ne)//circuit efficiency
Pout  = Pin+(Pgen*nc)//output power
//Output
mprintf('Electronic Efficiency is %3.2f\n Output
```

```
power is %g KW',ne,Pout/1000);

//
```

Scilab code Exa 5.5 Finding no of cycles

```
//chapter 5 example 5
//
clc;
clear;
//Given Data
F     =  1*10^9;//two cavity klystron operating
   frequency
Va    =  2500;//Accelarating voltage in volts
e     =  1.6*10^-19;//charge of electron
m     =  9.1*10^-31;//mass of electron in kg
s     =  0.1*10^-2;//input cavity space
//Calculations

u     = sqrt((2*e*Va)/m);//velocity at which
   electron beam enters the gap
T     = s/u ;//Time spent in the gap
f     = T*F;//number of cycles

//Output
mprintf('Number of cycles that elase during transit
   of beam through input gap is %3.3f cycle',f);

//
```

Scilab code Exa 5.6 `Finding phase difference and number possible useful modes of r`

```
//chapter 5 example 6
//
clc;
clear;
//Given Data
N      = 8;//no. of resonators

//Calculations
mprintf('    = (2*   *n)/N \n');//phase difference
mprintf('       = (n*   )/4\n');//phase difference
K    = N/2;//useful no. of nodes
//Most dominant mode is the one for which phase differnce b/w adjacent resonators is     radians
//Therefore (n*   )/4 =
n = 4


//Output
mprintf('Number of possible modes of Resonance is %d\n',N);
mprintf('Number of useful modes of Resonance is %d\n',K);
mprintf('value of integer n for the most dominant mode is %d',n);

//
```

Scilab code Exa 5.7 `Finding peak amplitude`

```
//chapter 5 example 7
//
clc;
clear;
//Given Data
Va     =  1200;//Anode potential
F      =  10*10^9;//Operating frequency in Hz
S      =  5*10^-2;//spacing b/w 2 cavities
GS     = 1*10^-3;//gap spacing in either cavity
e      =  1.6*10^-19;//charge of electron
m      =  9.1*10^-31;//mass of electron in kg
//Calculations
//Condition of maximum output is (V1/Vo)max =
    (3.68)/((2*pi*n)-(pi/2);
//(2*pi*n)-(pi/2) = Transit angle b/w two cavities
//V1  = Peak amplitude of RF i/p
//Vo  = accelarating potential

Vo     = sqrt(2*e*Va/m);//velocity of the electrons
T      = S/Vo;//Transit time b/w the cavities
TA     = 2*%pi*F*T;//transit angle in radians
V1     = (3.68*Va)/TA;
//Output
mprintf('Required Peak Amplitude of i/p RF signal is
     %3.2f volts',V1);
//
```

Scilab code Exa 5.8 Finding anode voltage of TWT

```
//chapter 5 example 8
//==================================================================
clc;
clear;
// Given Data
R        = 10;             // circumference to pitch ratio
e        = 1.6*10^-19;     // charge of electron
m        = 9.1*10^-31;     // mass of electron in Kg
c        = 3*10^8;         // vel. of EM waves in m/s

// Calculations
Vp       = c/R;             // axial phase velocity = free space vel*(pitch/circumference)
Va       = (Vp^2 * m)/(2*e);

// Output
mprintf('Anode Voltage = %3.2f kV',Va/1000);
disp('In practice,the electron beam velocity is kept slightly greater than the axial phase velocity of RF signal')
//------------------------------------------------------------------
```

Chapter 6

Semiconductor Microwave Devices and Integrated Circuits

Scilab code Exa 6.1 proof

```
// Chapter 6 example 1
// ------------------------------------------------------------------------------
clc;
clear;
// Given data
gs      = 0.0025;        // output conductance in mho
gl      = 0.0025;        // load conductance
r       = -250;          // negative resistance of
   microwave device

// calculations

//  P1  = Vl^2 *gl            // power that is
   transferred to load
//  P   = Vl^2 *gs            // source is matched to
   load
//  P   = [Is/(gl+gs)]^2 *gs
```

```
//        = ((Is^2)/(4*gs^2))*gs
//        = (Is^2)/(4*gl)
//   P2   = Vl^2 *gl             // Load power
//        = [Is/(gs+gl-g)]^2 *gl
//        = (Is^2 *gl)/(2gl - g)^2
// P2/P1 = ((Is^2 *gl)/(2gl - g)^2)*(4*gl)/(Is^2)
//         = (4*gl^2)/(2gl - g)^2;
//         = (4*gl^2)/(4gl^2 + g(g-4gl))
// For P2/P1 > 1 , 4gl > g so that denominator is less than numerator
g         = 1/r
// let k = P2/P1
k         = (4*gl*gl)/((2*gs)+ g)^2

// output
    mprintf('Power gain = %d',k);
```

Scilab code Exa 6.2 `Finding max negative differential conductance`

```
// Chapter 6 example 2
// ------------------------------------------------------------------------------
clc;
clear;
// Given data
Rl        = 500;        // load resistance

// Calculations
gl        = 1/Rl;       // load conductance
gmax      = 4*gl;       // max negative diff. conductance

// Output
mprintf('gmax = %3.3f mho',gmax);
```

```
// -----------------------------------------------------------------------------------------------
```

Scilab code Exa 6.3 `Finding operational frequency`

```
// Chapter 6 example 3
// -----------------------------------------------------------------------------------------------
clc;
clear;
// Given data
L       = 10*10^-6;      // width of N-region
Vs      = 10^5;          // saturated vel. of carriers

// Calculations
fo      = (3*Vs)/(4*L);      // oscillation frequency

// output
mprintf('Operational frequency = %3.1f Ghz\n',fo/10^9);
mprintf(' Note: In textbook it is wrongly printed as 6.5 Ghz')
// -----------------------------------------------------------------------------------------------
```

Scilab code Exa 6.4 `finding unity gain cutoff frequency`

```
// Chapter 6 example 4
```

```
//------------------------------------------------------------------------------
clc;
clear;
// Given data
L        = 10^-6;         // gate length
Vs       = 10^5;          // saturation velocity in m/s

// calculations
fT       = Vs/(2*%pi*L); // cut-off freq.

// Output
mprintf('Unity gain cut-off frequency = %3.0f Ghz',fT/10^9);
//------------------------------------------------------------------------------
```

Scilab code Exa 6.5 Finding length of active layer

```
// Chapter 6 example 5
//------------------------------------------------------------------------------
clc;
clear;
// Given data
f        = 10*10^9;       // oscillating freq. of Gunn diode
Vs       = 10^5;          // saturation carrier velocity in m/s

// calculations
```

```
L       = Vs/f;          // length of active layer

// Output
mprintf('Length of active layer = %3.0f  m ',L/10^-6);
// ---------------------------------------------------------------------------------
```

Scilab code Exa 6.6 Finding doping concentration

```
// Chapter 6 example 5
// ---------------------------------------------------------------------------------
clc;
clear;
// Given data
f       = 10*10^9;       // oscillating freq. of Gunn diode
Vs      = 10^5;          // saturation carrier velocity in m/s
er      = 13;            // relative permitivity
u       = 100*10^-4;     // mobility in m^2/V-s
eo      = 8.85*10^-12;   // permitivity in F/m
e       = 1.6*10^-19;    // charge of electron

// Calculations
L       = Vs/f;          // length of active layer
no      = (eo*er*Vs)/(L*e*u);   // doping concentration

// Output
mprintf('Doping Concentration no >> %3.2g /m^3',no);
//
```

```
------------------------------------------------------------------------
```

Scilab code Exa 6.7 Proof

```
// Chapter 6 example 7
//------------------------------------------------------------------------------
clc;
clear;
// Given data
fo       = 40*10^9;        // oscillating freq. of Gunn diode
no       = 10^15;          // doping concentration
up       = 8000;           // mobility in positive conductance region
er       = 13;             // relative permitivity
um       = 100;             // mobility in m^2/V-s
eo       = 8.85*10^-14;    // permitivity in F/cm
e        = 1.6*10^-19;     // charge of electron

// Calculations
// (eo*er)/(e*up) << no/fo < (eo*er)/(e*um)  // condition to be satisfied
// let k = (eo*er)/(e*up) , l = (eo*er)/(e*um) , p = no/fo
p        = no/fo
k        = (eo*er)/(e*up)
l        = (eo*er)/(e*um)
if (k<p) then
    if p<l then
        mprintf('Necessary Condition satisfied')
    end
    end
```

Scilab code Exa 6.8 Finding dielectric relaxation time

```
// Chapter 6 example 8
//
------------------------------------------------------------------------------
clc;
clear;
// Given data
n       = 10^15;          // doping concentration in /
   cm^3
u       = 8500;           // mobility in m^2/V-s
er      = 13;             // relative permitivity
eo      = 8.85*10^-14;    // permitivity in F/cm
e       = 1.6*10^-19;     // charge of electron

// Calculations
Td      = (eo*er)/(n*u*e)        // Dielectric
   relaxation time

// Output
mprintf('Dielectric relaxation time = %3.3f ps',Td
   *10^12);
//
------------------------------------------------------------------------------
```

Scilab code Exa 6.9 Finding length ogf GUN device

```
// Chapter 6 example 9
```

```
//
   ____________________________________________________________________________
clc;
clear;
// Given data
f        = 20*10^9;       // oscillating freq. of Gunn
    device
Vs       = 10^5;          // saturation carrier
   velocity in m/s

// Calculations
L        = Vs/f           // length of device

// output
mprintf('length of device = %d  m ',L*10^6);
//
   ____________________________________________________________________________
```

Scilab code Exa 6.10 Finding mobility values

```
// Chapter 6 example 10
//
   ____________________________________________________________________________
clc;
clear;
// Given data from graph
up  = (2*10^7)/3000;                  // mobility of
   diode in positive conductance region
un  = (2*10^7 - 10^7)/((10-3)*10^3); // mobility of
   diode in negative conductance region

// Output
```

```
mprintf('mobility of diode in positive conductance region = %d cm^2/V-s\n mobility of diode in negative conductance region = %3.0f cm^2/V-s',up,un);
//____________________________________________________________________________
```

Scilab code Exa 6.11 Finding electric field and punch through voltage

```
// Chapter 6 example 11
//____________________________________________________________________________
clc;
clear;
// Given data
e       = 1.6*10^-19;    // charge of electron
Nd      = 10^15*10^6;    // mobility
L       = 10*10^-6;      // active layer of Barritt diode
er      = 12.5           // relative permitivity
eo      = 8.85*10^-12;   // permitivity in F/cm

// calculations
Ex      = (e*Nd*L)/(2*eo*er) // electric field for Va = Vpt and x = L/2
E       = Ex/10^2;           // electric field in v/cm
Vpt     = 10*10^-4*E

// Output
mprintf('Electric field E(x) = %3.0f KV/cm\n Punch through voltage = %3.0f Volts',E/1000,Vpt);
//
```

```
---------------------------------------------------------------------------
```

Scilab code Exa 6.12 `Finding Hfe`

```
// Chapter 6 example 12
// ---------------------------------------------------------------------------
clc;
clear;
// Given data
fT      = 10;        // ft specification of BJT
f_a     = 2;         // operating freq in Ghz case a
f_b     = 10;        // operating freq in Ghz case b

// calculations
hFE_a   = fT/f_a;
hFE_b   = fT/f_b;

// Output
mprintf('case a:\n  hFE = %d\n case b:\n  hFE = %d\n
   ',hFE_a,hFE_b);
// ---------------------------------------------------------------------------
```

Scilab code Exa 6.13 `Finding dielectric relaxation time`

```
// Chapter 6 example 13
// ---------------------------------------------------------------------------
```

```
clc;
clear;
// Given data
n        = 10^15;          // doping concentration in /
   cm^3
er       = 15;             // relative permitivity
eo       = 8.85*10^-14;    // permitivity in F/cm
e        = 1.6*10^-19;     // charge of electron
sigma    = 133*10^-2;      // conductivity in ohm/cm

// calculations
Td       = (eo*er)/sigma      // dielectric relaxation
   time constant
u        = sigma/(n*e)        // mobility

// Output
mprintf('Dielectric relaxation time constant = %3.0f
    ps\n Carrier Mobility = %d cm^2/V-s\n',Td*10^12,
   u );
//
   ------------------------------------------------------------------------
```

Scilab code Exa 6.14 Finding frequency

```
// Chapter 6 example 14
//
   ------------------------------------------------------------------------

clc;
clear;
// Given data
gm       = 50*10^-3;          // conductance in mho
cgs      = 0.6*10^-12;      // gate to source
   capacitance
```

```
cgd       = 0.015*10^-12;   // gate to drain
   capacitance
Rg        = 3;              // gate resistance in ohm
Rs        = 2;              // source resistance in ohm
Ri        = 2.5;            // intrinsic channel
   resistance
Rds       = 400;            // drain to source
   resistance

// Calculations
fT        = gm/(2*%pi*cgs);    // device's fT
t3        = 2*%pi*Rg*cgd;
r1        = (Rg+Rs+Ri)/Rds;
fmax      = fT/(2*sqrt(r1 + (fT*t3)));     // max
   usable frequency
if fmax>40*10^9 then
    mprintf('Operation at 40 GHz is Theoretically
       possible\n');
end

// Output
mprintf(' fT = %3.1f Ghz\n fmax = %3.1f',fT/10^9,
   fmax/10^9 )
```

Scilab code Exa 6.15 Finding power gain

```
// Chapter 6 example 15
//
   ------------------------------------------------------------------------------------

clc;
clear;
// Given data
f2        = 20;         // pump frequency in GHz
f1        = 2;          // signal frequency in GHz
```

```

// Calculations
Gp       = (f1+f2)/f1;        // power gain if parametric amp. operated as USB up-converter
Gp_dB    = 10*log10(Gp);      // power gain in dB
Gp_lsb   = (f2-f1)/f1;        // power gain if parametric amp. operated as LSB up-converter
Gp_db_lsb = 10*log10(Gp_lsb )// power gain in dB

// output
mprintf('Power gain of parametric amplifier when operated as USB up-converter = %3.1f dB\n Power gain of parametric amplifier when operated as LSB up-converter = %3.1f dB',Gp_dB,Gp_db_lsb)
// ------------------------------------------------------------------------------
```

Scilab code Exa 6.16 Finding output laser wavelength

```
// Chapter 6 example 16
// ------------------------------------------------------------------------------
clc;
clear;
// Given data
h        = 6.63*10^-34;       // planck's constant in Joule-sec
el       = 0.25;              // lower energy level in eV from energy level diag.
eh       = 1.5;               // higher energy level in eV from energy level diag.
e        = 1.6*10^-19;       // charge of electron
c        = 3*10^8;           // vel. of light in m/s
```

```
// calculations
hf      = (eh - el)*e                  // energy diff b/w two levels in J
f       = hf/h;                        // frequency
lamda   = c/f                          // o/p laser wavelength in m

// Output
mprintf('Output laser wavelength = %3.0e m or%3.0f m  ',lamda,lamda*10^6)
```

Scilab code Exa 6.17 Finding resistance

```
// Chapter 6 example 17
//------------------------------------------------------------------------------
clc;
clear;
// Given data
p   = 0.1*10^-2;           // resistivity in ohm-m
t   = 100*10^-6;           // thickness in m
AR  = 10/1;                // aspect ratio

// Calculations
ps  = p/t
R   = ps*AR;               // Resistance in ohm

// Output
mprintf('Resistance = %d   ',R);
//------------------------------------------------------------------------------
```

Scilab code Exa 6.18 Finding sheet resistivity and Resistance

```
// Chapter 6 example 18
//------------------------------------------------------------------------------
clc;
clear;
// Given data from fig
R_a = 1000;      // resistance shown in fig a
W1  = 0.15*10^-3 // width of geometry fig 6.72a
L1  = 3*10^-3    // Length of geometry fig 6.72a
W2  = 75*10^-6   // width of geometry fig 6.72b
L2  = 1500*10^-6 // Length of geometry fig 6.72b
t1  = 10*10^-6   // thickness of geometry fig 6.72a
t2  = 20*10^-6   // thickness of geometry fig 6.72b

//R1  =  s1 *(L1/W1)  // resistor geometry of fig 6.72a
// s1  = (R1*W1)/L1
ps1   = (R_a*W1)/L1  // sheet resistivity of geometry of fig 6.72a
p     = ps1*t1;      // resistivity
ps2   = p/t2;        // sheet resistivity of geometry of fig 6.72b
R2    = ps2*(L2/W2); // resistance of geometry of fig 6.72b

// Output
mprintf('For Geometry in Fig 6.72b\n sheet resistivity = %3.0f  /  \n Resistance = %d  ',ps2,R2)
//------------------------------------------------------------------------------
```

Scilab code Exa 6.19 Finding capacitance

```
// Chapter 6 example 19
//------------------------------------------------------------------------------
clc;
clear;
// Given data
A       = 100*100*10^-12;         // Area of electrode
er      = 9.6;                    // relative permitivity
t       = 500*10^-6;              // substrate thickness
eo      = 8.85*10^-12;            // permitivity
// Calculations
C       = (eo*er*A)/t;              // capacitance in farad

// Output
mprintf('Capacitance = %3.2e pF',C*10^12);
//------------------------------------------------------------------------------
```

Scilab code Exa 6.20 Semiconductor Microwave Devices and Integrated Circuits

```
// Chapter 6 example 20
//------------------------------------------------------------------------------
```

```
clc;
clear;
// Given data
ps       = 100;       // sheet resistivity
L        = 1.04       // length
W        = 0.02       // width

// Calculations
NOS      = L/W          // number of squares
R        = ps * NOS    // resistance

// Output
mprintf('Resistance = %3.1f  K  ',R/1000);
//
    -----------------------------------------------------------------------
```

Chapter 7

Antennas

Scilab code Exa 7.1 `Calculating Q`

```
// chapter 7 example 1
// ______________________________________________________________________________

clc;
clear;
// given data
Ldipole     = 50;        // Length of dipole in cm
c           = 3*10^10;   // velocity of EM wave in cm/s
BW          = 10*10^6;   // bandwidth in Hz

// Calculations
lamda       = 2*Ldipole;     // wavelength in cm
fo          = c/lamda;       // operating frequency in Hz
Q           = fo/BW          // quality factor

// Output
mprintf('Q = %d',Q);
//
```

```
//
   ________________________________________________________________________
```

Scilab code Exa 7.2 Finding Directivity

```
// chapter 7 example 2
//
   ________________________________________________________________________

clc;
clear;
// given data
Rr      = 72;        // Radiation resistance in ohms
Rl      = 8;         // Loss resistance in ohms
Ap      = 27;        // power gain

// Calculations
n       = Rr/(Rr + Rl);      // radiation efficiency
D       = Ap/n;              // Directivity
D_dB    = 10*log10(D);       // directivity in dB

// Output
mprintf('Directivity = %3.2f dB',D_dB );
//
   ________________________________________________________________________
```

Scilab code Exa 7.3 Finding Aperture and gain of antenna

```
// chapter 7 example 3
//
   ________________________________________________________________________
```

```
clc;
clear;
// given data
AZ_BW       = 0.5;       // beamwidth in degrees
E_BW        = 0.5;       // beamwidth in degrees
lamda       = 3*10^-2;   // radar emission wavelength

// Calculations

AZ_BW_r     = AZ_BW*%pi/180;      // azimuth beamwidth in radians
E_BW_r      = E_BW*%pi/180;      // elevation beamwidth in radians
G           = (4*%pi)/(AZ_BW_r *E_BW_r )      // antenna gain
G_db        = 10*log10(G)        // gain in dB
A           = (G*lamda*lamda)/(4*%pi);   // antenna aperture

// Output
mprintf('Gain of Antenna = %3.2f dB\n Antenna Aperture = %3.3f m',G_db,A);
// ------------------------------------------------------------------------------
```

Scilab code Exa 7.4 Finding effective aperture of antenna

```
// chapter 7 example 4
// ------------------------------------------------------------------------------
clc;
clear;
// given data
```

```
n_az        = 0.5;          //length efficiency in azimuth direction
n_el        = 0.7;          //length efficiency in elevation direction
A           = 10;           // area in square mts

// Calculations
n           = n_az * n_el;      // aperture efficiency
Ae          = n*A;              // Effective aperture

// Output
mprintf('Effective aperture of the antenna = %3.1f sq.m',Ae);
//____________________________________________________________
```

Scilab code Exa 7.5 finding Directivity

```
// chapter 7 example 5
//____________________________________________________________
clc;
clear;
// given data
Ptot          = 100;        // certain antenna radiating power
Ptot_iso      = 10*10^3;    // isotropic antenna radiating power

// Calculations
D             = 10*log10(Ptot_iso/Ptot);   // Directivity of antenna

```

```
// Output
mprintf('Directivity of antenna = %d dB',D);
//
   ________________________________________________________________________
```

Scilab code Exa 7.6 Finding beamwidth effective aperture and gain

```
// chapter 7 example 6
//
   ________________________________________________________________________

clc;
clear;
// given data
D        = 3;          // diameter of the antenna in m
n_l      = 0.7;        // length efficiency
nr       = 0.9;        // radiation efficiency
f        = 10*10^9;    // antenna operating freq.
c        = 3*10^8;     // vel of EM waves in m/s

// calculations
def      = D*n_l      // Effective diameter
lamda    = c/f        // wavelength in m
Beam_w   = lamda/def  // beamwidth in radian
Beam_w_d= Beam_w*180/%pi;         // beam width in
   degree;
n_a      = n_l * n_l;     // Aperture efficiency
AA       = (%pi*D*D)/4;   // actual area in sq m
Ae       = AA*n_a;        // Effective aperture
G        = (4*%pi*Ae)/(lamda^2); // Gain
G_db     = 10*log10(G);

// Output
mprintf('Beam Width = %3.2f degrees\n Effective
```

```
    Aperture = %3.2fsq m\n Gain = %3.1f dB',Beam_w_d,
    Ae,G_db);
//
    ___________________________________________________________________________
```

Scilab code Exa 7.7 `Finding radiation resistance`

```
// chapter 7 example 7
//
    ___________________________________________________________________________
clc;
clear;
// given data
// given (lamda/10) wire dipole
// Radiation resistance of short dipoles is Rr =
    790*(1/lamda)^2;
// Rr   = 790*(lamda/(10*lamda))^2;
// Rr   = 7.9;
mprintf('Radiation resistance = 7.9 ohms');
//
    ___________________________________________________________________________
```

Scilab code Exa 7.8 `Finding Beamwidth effective aperture and gain`

```
// chapter 7 example 8
//
    ___________________________________________________________________________
clc;
clear;
```

```
// given data
a_l      = 6;          // Azimuth length in m
n_a      = 0.7;        // Azimuth length efficiency
n_e      = 0.5;        // elevation length efficiency
e_l      = 4;          // elevation length in m
w        = 6;          // width of antenna
h        = 4;          // height of antenna
lamda    = 3*10^-2;    // wavelength

// Calculations
Eff_A_l = a_l*n_a;    // effective azimuth length
Eff_E_l = e_l*n_e;    // effective elevation length
A        = w*h         // actual area
n        = n_a*n_e;    // aperture efficiency
Ae       = A*n;        // effective aperture
Az_BW    = lamda/Eff_A_l   // Azimuth beam width
E_BW     = lamda/Eff_E_l   // elevation beam width
Az_BW_d = Az_BW*180/%pi   // rad to deg conv
E_BW_d   = E_BW*180/%pi;   // rad to deg conv
G        = (4*%pi*Ae)/(lamda^2); //Gain
G_dB     = 10*log10(G);   // gain in dB

// Output
mprintf('Azimuth Beamwidth = %3.2f degrees\n
   Elevation Beamwidth = %3.2f degrees\n Gain = %3.1
   f dB',Az_BW_d,E_BW_d,G_dB);
//
   ____________________________________________________________________________
```

Scilab code Exa 7.9 Finding beamwidth

```
// chapter 7 example 9
//
   ____________________________________________________________________________
```

```
clc;
clear;
// given data
Beam_w_3db  = 0.4;

// Calculations
N2N_Beam_w  = 2*Beam_w_3db; // Null to Null
   beamwidth

// output
mprintf('Null to Null Beam width = %3.1f degrees',
   N2N_Beam_w);
//
   ____________________________________________________________
```

Scilab code Exa 7.10 Finding Received signal strength

```
// chapter 7 example 10
//
   ____________________________________________________________

clc;
clear;
// given data
RSSR        = 20;    // Rx signal strength in
   horizontal polarised antenna when rx RHCP

// Calculations
// When incident polarisation is circularly
   polarised and the antenna is linearly polarised,
   there is a ploarisation loss of 3dB
ISS         = RSSR + 3;
// a
```

```
// when the Rx polarisation is same as the antenna polarisation , the polarisation loss is zero
RSS_HP      = ISS;      // rx signal strength for incident wave horizontally polarised
// b
// when the incident wave is vertically polarised , the angle between the incident polarisation and the antenna polarisation is 90
// polarisation loss = 20log(1/cos(  ))
//                   = 20log(1/cos90) =
RSS_VP      = 0;        // rx signal strength for incident wave vertically polarised
// c
// When the incident wave is LHCP and the antenna polarisation is linear ,there will be a 3dB polarisation loss and the
// Rx signal strength therefore will be 20 dB only
RSS_LHCP    = RSSR;     // rx signal strength for incident wave Left hand circularly polarised
// d
// The angle between the incident wave polarisation and the antenna polarisation is 60 degrees
phi         = 60;                          // rx wave polarisation angle with horizontal
PL          = 20*log10(1/cos(60*%pi/180));       // polarisation loss in dB
RSS_Pangle  = ISS - PL;
//output
mprintf('Received signal strength if incident wave horizontally polarised = %d dB\n Received signal strength if incident wave vertically polarised = %d dB\n Received signal strength if incident wave Left hand circularly polarised is %d dB\n Received signal strength if Received wave polarisation making 60deg angle with horizontal is %3.0f dB',RSS_HP,RSS_VP,RSS_LHCP,RSS_Pangle);
//______________________________________________________________________________
```

Scilab code Exa 7.11 Finding length of halfwave dipole

```
// chapter 7 example 11
//
   ------------------------------------------------------------------------
clc;
clear;
// given data
f       = 300*10^6;      // operating frequency in Hz
c       = 3*10^10;        // velocity of EM wave in
   cm/s

// Calculations
lamda   = c/f;            // wavelength in cm
// Physical length of antenna is made 5% shorter
   than desired length as per rule of thumb
l       = lamda/2;        // length of halfwave dipole
lphy    = l-(5/100)*l;   // as per rule of thumb

// Output
mprintf('Length of a half wave dipole to be cut = %3
   .1f cm',lphy);
//
   ------------------------------------------------------------------------
```

Scilab code Exa 7.12 Finding input impedance

```
// chapter 7 example 12
```

```
//
   ________________________________________________________________________

clc;
clear;
// given data
Zi       = 72;        // input impedance in ohms
// A     = 1.5a       // area of cross section in sq.
   cm
// Zif   = Zi*[(sum of areas of cross section of
   various components)/(Area of cross section of the
    driven element )]^2
// Zif   = 72*((a + 1.5a)/a)^2;
// Zif   = 72*(2.5*a/a)^2;
Zif      = 72*(2.5)^2;
mprintf('Input impedance for a folded dipole = %d
   ',Zif);
//
   ________________________________________________________________________
```

Scilab code Exa 7.13 `Designing yagi antenna`

```
// chapter 7 example 13
//
   ________________________________________________________________________

clc;
clear;
// given data
f        = 60*10^6;       // frequency in Hz
c        = 3*10^8         // velocity of EM wave in m/
   s

// Calculations
```

```
lamda    = c/f;             // wavelength in m
l_dipole= lamda/2          // length of diplole
// Physical length of antenna is made 5% shorter than desired length as per rule of thumb
L        = l_dipole - (5/100)*l_dipole;  // actual physical length
L_D      = L - (4/100)*L;                // length of director
L_R      = L + (4/100)*L;                // length of reflector
DDS      = 0.12*lamda;                   // director dipole spacing
RDS      = 0.2*lamda;                    // Reflector dipole spacing

// Output
mprintf('Length of dipole = %3.3f m\n length of Director = %3.2f m\n length of Reflector = %3.2f m\n director dipole spacing = %3.1f m\n Reflector dipole spacing = %3.1f m',L,L_D,L_R,DDS,RDS);
// ______________________________________________________________________
```

Scilab code Exa 7.14 finding beamwidth

```
// chapter 7 example 14
// ______________________________________________________________________
clc;
clear;
// given data
D        = 2;          // Mouth diameter in m
f        = 2;          // focal length in m
```

```
bw3db    = 90/100;    // beamwidth of antenna chosen to be 90% of angle subtended by feed

// Calculations
theta    = 4*atan(1/(4*f/D));     // angle subtended by the focal point feed at edges of reflector
theta_d = theta*180/%pi
Beam_w_3dB = bw3db*theta_d;        // 3 dB beam width
NNBW     = 2*(Beam_w_3dB );

// Output
mprintf('3 dB Beamwidth = %3.1f  \n Null-to-Null beam width = %3.2f  \n',Beam_w_3dB,NNBW);
//------------------------------------------------------------------------------
```

Scilab code Exa 7.15 Finding focal length of antenna

```
// chapter 7 example 15
//------------------------------------------------------------------------------
clc;
clear;
f        = 3;          // focal length in m
fpos     = 1.5;        // feed is placed 1.5m from pt of intersection os sec.reflector and antenna axis

// Calculation
f_hyp    = f-fpos;    // focal length of hyperboloid from figure;

// Output
mprintf('focal length of hyperboloid = %3.1f m',
```

```
    f_hyp);
//
    ____________________________________________________________
```

Scilab code Exa 7.16 Finding distance of the feed

```
// chapter 7 example 16
//
    ____________________________________________________________

clc;
clear;
// given data
D       = 3;        // Mouth diameter in m
//f       = 2;        // focal length in m
bw3db   = 63;      // 3dB beam width
k       = 0.9;      // beam width is k times
    subtended angle

// Calculations
theta   = bw3db/k;  // subtended angle
theta_r = theta
//theta   = 4*atan(1/(4*f/D));
f       = D/(4*tan((theta_r/4)*(%pi/180)));

// Output
mprintf('Distance of feed from the point of
    intersection of antenna axis and the reflector
    surface = %3.2f m',f);
//
    ____________________________________________________________
```

Scilab code Exa 7.17 Finding desired phases of all elements

```
// chapter 7 example 17
// ------------------------------------------------------------------------------------
clc;
clear;
// given data
c        = 3*10^8;         // velocity of EM waves in m/s
f        = 2.5*10^9;       // operating frequency in Ghz
S        = 10*10^-2;       // inter element spacing
theta    = 10;             // steering angle

// Calculations
lamda    = c/f            // Wavelength in m
phi      = (360*(S/lamda))*sin(theta*(%pi/180))
phi1     = 0*phi          // phase angle for element 1
phi2     = 1*phi          // phase angle for element 2
phi3     = 2*phi          // phase angle for element 3
phi4     = 3*phi          // phase angle for element 4
phi5     = 4*phi          // phase angle for element 5

// Output
mprintf('Phase angles for elements 1,2,3,4,5 are %d , %d , %d , %d , %d ',phi1,phi2,phi3,phi4,phi5);
// ------------------------------------------------------------------------------------
```

Scilab code Exa 7.18 `Finding Phase angles`

```
// chapter 7 example 17
//------------------------------------------------------------------------------
clc;
clear;
// Data is taken from Example 17. The beam steers towards left of the axis with all parameters remaining in Ex 17 are same
c       = 3*10^8;        // velocity of EM waves in m/s
f       = 2.5*10^9;      // operating frequency in Ghz
S       = 10*10^-2;      // inter element spacing
theta   = -10;            // steering angle

// Calculations
lamda   = c/f           // Wavelength in m
phi     = (360*S/lamda)*sin(theta*%pi/180)
phi1    = 0*phi         // phase angle for element 1
phi2    = 1*phi         // phase angle for element 2
phi3    = 2*phi         // phase angle for element 3
phi4    = 3*phi         // phase angle for element 4
phi5    = 4*phi         // phase angle for element 5

// Output
mprintf('Phase angles for elements 1,2,3,4,5 are %d , %d , %d , %d , %d ',phi1,phi2,phi3,phi4,phi5);
//------------------------------------------------------------------------------
```

Scilab code Exa 7.19 Finding beam position

```
// chapter 7 example 8
//
   ------------------------------------------------------------------------
clc;
clear;
// given data
S         = 5*10^-2;        // inter spacing distance
lamda     = 6*10^-2;        // operating wavelength in
   cms
phi_Az     = 25              // angle in azimuth
   direction
phi_E      = 35              // angle in Elevation
   direction

// Calculations
theta_Az   = asin((lamda*phi_Az)/(360*S))
theta_E    = asin((lamda*phi_E)/(360*S))
Theta_Az   = theta_Az*(180/%pi)
Theta_E    = theta_E*(180/%pi)

// Output
mprintf('Steering angle in Azimuth = %3.1f  \n
   Steering angle in Elevation = %3.1f  ',Theta_Az,
   Theta_E);
//
   ------------------------------------------------------------------------
```

Chapter 9

Radar Fundamentals

Scilab code Exa 9.1 Finding max unambiguous range of radar

```
// Chapter 9 example 1
// ________________________________________________________________________________
clc;
clear;
// Given Data
PRF      = 1000;          // Pulse repetitive
   frequency in Hz
t        = 0.15*10^-3;    // Round propagation time in
    s
c        = 3*10^8;        // velocity of EM waves in m
   /s
// calculations
R        = (c*t)/2;       // Range
Runamb   = c/(2*PRF)      // Max unambiguous range

// Output
mprintf('Target Range = %3.1f Km\n Maximum
   Unambiguous range = %d Km',R/1000,Runamb/1000);
//
```

Scilab code Exa 9.2 Finding Rx signal frequency

```
// Chapter 9 example 2
//------------------------------------------------------------------------------
clc;
clear;
// Given Data
f        = 10*10^9;        // radar Tx frequency
c        = 3*10^8;         // velocity of EM waves in m/s
V        = 108;            // vel of car in kmph

// Calculations
lamda    = c/f;            // wavelength in m
Vr       = V*(5/18);       // vel of car in m/s
fd       = (2*Vr)/lamda    // Doppler shift in Hz
fr       = f + fd          // received freq
fr_away = f-fd             // Rx frequency if the car is moving away from radar

// Output
mprintf('Doppler Shift = %d Khz\n Frequency of Received signal = %3.6f Ghz\n Received Frequency if car is moving away from radar = %3.6f Ghz',fd/1000,fr/10^9,fr_away/10^9);
//------------------------------------------------------------------------------
```

Scilab code Exa 9.3 Determining whether radar is capable of measuring certain radi

```
// Chapter 9 example 3
//    ________________________________________________________________________
clc;
clear;
// Given Data
f        = 10*10^9;        // radar Tx frequency
PRF      = 2000;           // Pulse repetitive frequency in Hz
Vr       = 0.5;            // radial vel in Mach
c        = 3*10^8;         // velocity of EM waves in m/s
vs       = 330;            // velocity of sound in m/s

// Calculations
lamda            = c/f;            // wavelength in m
max_unamb_fd     = PRF/2;          // maximum unambiguous doppler shift
Vrunamb          = (lamda*max_unamb_fd)/2; // doppler shift
Vaircraft        = 0.5*vs;         // Converting from Mach to m/s
fd_desired       = (2*Vaircraft)/lamda;
PRF_desired      = 2*fd_desired; // desired PRF

// Output
if  Vrunamb < Vaircraft then
    mprintf('The radar is not capable of determining unambiguously the velocity of the approaching aircraft\n');
end
```

```
mprintf(' Desired Pulse Repetition Rate = %d Khz',
   PRF_desired/1000);
//
   ______________________________________________________________________
```

Scilab code Exa 9.4 Determining Range Resolution

```
// Chapter 9 example 4
//
   ______________________________________________________________________

clc;
clear;
// Given Data
PW_tx   = 10^-6;            // Transmitted pulse
   width
Rx_PW   = 10^-6;            // Received pulse width
c       = 3*10^8;           // velocity of EM waves
   in m/s

// Calculations
RR      = (c*Rx_PW)/2;      // Range Resolution in m

// output
mprintf('This Radar can resolve upto an inter target
    separation in range of %d m\n Therefore,given
   radar will be able to resolve the targets',RR);
//
   ______________________________________________________________________
```

Scilab code Exa 9.5 Determining max beamwidth

```
// Chapter 9 example 5
// ------------------------------------------------------------------------------
clc;
clear;
// Given Data
CRR      = 100;       // Cross range resolution in m
R        = 3000;      // radial range

// Calculations
// CRR  = (R*theta3)*(%pi/180);
theta3  = (180*CRR)/(%pi*R)          // 3 dB beamwidth

// Output
mprintf('3 dB beamwidth = %3.2f  ',theta3);
// ------------------------------------------------------------------------------
```

Scilab code Exa 9.6 Finding min look time

```
// Chapter 9 example 6
// ------------------------------------------------------------------------------
clc;
clear;
// Given Data
Vs       = 330;           // velocity of sound in m/s
NM       = 1.85*(5/18)    // 1NM equivalent in m/s
V1       = 0.5;           // velocity of first aircraft in mach
V2       = 400;           // velocity of second
```

```
aircraft in NM/hr
theta    = 30;            // angle with radial axis in degrees
lamda    = 3*10^-2;        // wavelength in m

// Calculations
v1       = V1*Vs          // velocity of first aircraft in m/s
fd1      = (2*v1)/lamda  // doppler freq.
v2       = V2*NM*cos(30*(%pi/180))  // velocity of second aircraft in m/s
fd2      = (2*v2)/lamda  // doppler freq
dd       = fd2 - fd1     // doppler difference
Tl       = 1/dd          // look time in s

// Output
mprintf('Required minimum look time = %3.2f ms',Tl/10^-3);
mprintf('\n Note: Cos(30) value is taken as 0.5 in textbook');
// ____________________________________________________________________
```

Scilab code Exa 9.7 `Significance of denominator`

```
// Chapter 9 example 7
// ____________________________________________________________________
clc;
clear;
// Given Data
// Rmax       = [1000000/(12.4*PRF)]NM
//            = [1000000*t/12.4]NM
```

```
mprintf('The Numerator represents round trip
   propagation time in us\n');
mprintf(' Therefore, number 12.4 represents the
   units microseconds per nautical miles\n');
mprintf(' In other words, this means that the round
   propagation time for one nautical mile is 12.4 us
    which is equivalent to 6.66us for 1km range')
```

Scilab code Exa 9.8 Finding center frequency

```
// Chapter 9 example 8
//
   ------------------------------------------------------------------------------

clc;
clear;
// Given Data
PW        = 10*10^-6;           // pulse width in sec
f         = 10*10^9;            // frequency in Hz
fm        = 1000;               // modulating frequency

// calculations
BW_M      = 1/PW                // matched bandwidth
cf1       = f+fm;               // closest freq.
cf2       = f-fm;               // closest freq.
fo        = f;                  // centre freq.

// Output
mprintf('Centre of frequency spectrum = %d Khz\n The
    two closet frequencies to the center of the
   spectrum are %d Khz and %d Khz',fo/10^3,cf1/10^3,
   cf2/10^3);
//
   ------------------------------------------------------------------------------
```

Scilab code Exa 9.9 Finding centre of spectrum bandwidth and compressed pulse widt

```
// Chapter 9 example 9
// ------------------------------------------------------------------
clc;
clear;
// Given Data
fc1      = 495;       // freq in Mhz
fc2      = 505;       // freq in Mhz

// Calculations
fo       = (fc1 + fc2)/2;         // Center of spectrum in Mhz
BW       = fc2 - fc1;             // Bandwidth in Mhz
PW       = 1/BW;                  // compressed pulse width in us

// Output
mprintf('Center of spectrum = %d Mhz\n Matched Bandwidth = %d Mhz\n Compressed Pulse width = %3.1fus',fo,BW,PW);
// ------------------------------------------------------------------
```

Scilab code Exa 9.10 Finding Bandwidth and range resolution

```
// Chapter 9 example 10
```

```
//
   ----------------------------------------------------------------------------

clc;
clear;
// Given Data
f        = 10^9;          // CW radar waveform freq.
fm       = 100;           // modulation freq. in Hz
MaxfD    = 500;           // max freq deviation in Hz
c        = 3*10^8;        // vel. of EM waves in m/s

// Calculations
Mf       = MaxfD/fm       // Modulation index
BW       = 2*(Mf + 1)*fm  // Bandwidth
RR       = c/(2*BW);      // Range Resolution in m

// Output
mprintf('Bandwidth = %d Hz\n Range Resolution = %d
   Km',BW,RR/1000);
//
   ----------------------------------------------------------------------------
```

Scilab code Exa 9.11 Finding matched bandwidth and center frequency of spectrum

```
// Chapter 9 example 11
//
   ----------------------------------------------------------------------------

clc;
clear;
// Given Data
f        = 10^9;          // Centre freq. of spectrum
t        = 13             // pulse width in us
N        = 13;            // N-bit Barker code
```

```

// Calculations
Sub_PW   = t/N;            // sub pulsewidth
match_BW= 1/Sub_PW;        // Matched bandwidth in Mhz

// Output
mprintf('Matched Bandwidth = %d Mhz\n Center
   Frequency of the spectrum = %d Ghz',match_BW,f
   /10^9 );
//
   ____________________________________________________________________________
```

Scilab code Exa 9.12 Finding average power and look energy

```
// Chapter 9 example 12
//
   ____________________________________________________________________________

clc;
clear;
// Given Data
PW       = 10^-6;              // Pulse width in sec
Pp       = 100*10^3;           // Peak power in watts
PRF      = 1000;               // pulse rep.rate
N_target= 20;                  // no of target hits in
   1 dwell period

// Calculations
PE       = Pp*PW;              // Pulse energy in Joule
LE       = N_target *PE;       // look energy
DC       = PW*PRF              // Duty cycle
Pav      = Pp*DC;              // Average power
Pavg     = 10*log10(Pav);      // Avg power in dB

```

```
// Output
mprintf('Average power = %d dB\n Look Energy = %3.0f Joules',Pavg,LE);
//
```

Scilab code Exa 9.13 `finding duty cycle correction factor`

```
// Chapter 9 example 13
// Data taken from Ex 12
//

clc;
clear;
// Given Data
PW      = 10^-6;             // Pulse width in sec
Pp      = 100*10^3;          // Peak power in watts
PRF     = 1000;              // pulse rep.rate
N_target= 20;                // no of target hits in 1 dwell period

// Calculations
PE      = Pp*PW;             // Pulse energy in Joule
LE      = N_target *PE;      // look energy
DC      = PW*PRF             // Duty cycle
Pav     = Pp*DC;             // Average power
Pavg    = 10*log10(Pav);     // Avg power in dB
Pp_dB   = 10*log10(Pp);      // Peak power in dB
DCCF    = Pp_dB - Pavg       // Duty cycle correction factor
// Output
mprintf('Duty cycle correction factor = %d dB',DCCF);
```

```
//  ____________________________________________________________________________
```

Scilab code Exa 9.14 `Finding Equivalent noise temperature`

```
// Chapter 9 example 14
//  ____________________________________________________________________________

clc;
clear;
// Given Data
G_rx          = 97;         // Rx gain in dB
Bn            = 5*10^6;     // Bandwidth in Hz
To            = 300;        // temperature in kelvin
K             = 1.38*10^-23;   // Boltzmann constant in J/k
n             = -3          // o/p noise power in dBm

// calculations
Pn_dB        = n-G_rx        // input noise power
Pn           = 10^(Pn_dB/10)*10^-3       // converting from dBm to watts
// Pn        = KToBnF;
F            = Pn/(K*To*Bn)   // Noise Factor
T            = To*[F - 1]     // Equivalent Noise Temperature

// Output
mprintf('Equivalent Noise Temperature = %d K',T );
//  ____________________________________________________________________________
```

Scilab code Exa 9.15 Determining ratio of noise powers

```
// Chapter 9 example 15
//
   ------------------------------------------------------------------------
clc;
clear;
// Given Data
Gx      = 60;        // gain of Rx 'X' in dB
Gy      = 70;        // gain of Rx 'Y' in dB
Fx      = 3;         // Noise factor of 'X'
Fy      = 2;         // Noise factor of 'Y'

// calculations
Gx_W    = 10^(Gx/10)    // gain in watts
Gy_W    = 10^(Gy/10)    // gain in watts
// k    = Pnx/Pny;      // Ratio of noise power
   levels produced at the o/p's of Rx 'X' and 'Y'
k       = (Fx*Gx_W)/(Fy*Gy_W);    // Ratio of noise
   power levels produced at the o/p's of Rx 'X' and
   'Y'

// output
mprintf('Ratio of noise power levels produced at the
    outputs of Rx X and Y = %3.2f',k);
//
   ------------------------------------------------------------------------
```

Scilab code Exa 9.16 Finding noise power

```
// Chapter 9 example 16
//------------------------------------------------------------------------------
clc;
clear;
// Given Data
Pn      = -70;          // Noise power in dBm
fl      = 10^6;         // lower cut-off freq in Hz
fh      = 11*10^6;      // upper cut-off freq in Hz
BP_fl   = 13*10^6;      // Bandpass filter lower cutoff freq
BP_fh   = 14*10^6;      // Bandpass filter lower cutoff freq

// Calculations
Pn_W    = 10^(Pn/10)*10^-3;     // coversion from dBm to Watts
BW      = fh - fl
PSD     = Pn_W/BW               // Noise power spectral density
// Since white noise has the same spectral power density through the frequency spectrum,
// therefore Noise power in second case
B       = BP_fh - BP_fl
Pn_2    = PSD*B;                // Noise power in second case

// Output
mprintf('Noise power for BandPass filter having Cutoff frequencies 13Mhz and 14Mhz = %3.0e W',Pn_2);
//------------------------------------------------------------------------------
```

Scilab code Exa 9.17 Finding azimuth coordinates

```
// Chapter 9 example 16
//
    ------------------------------------------------------------------------------
clc;
clear;
// Given Data from Figure triagle OAB
OA      = 100       // in Km
OB      = OA*cos(60*%pi/180);       // Range of
    Target 2

// Output
mprintf('Range of Target-2 = %d Km\n Azimuth angle
    of target-1 = 60  \n Azimuth angle of Target-2 =
    120  ',OB);
//
    ------------------------------------------------------------------------------
```

Chapter 10

Radar Systems

Scilab code Exa 10.1 Finding Target range

```
// Chapter 10 example 1
//______________________________________________________________________________

clc;
clear;
c       = 3*10^8;           // velocity of EM waves in m/s
f       = 10*10^9;          // carrier freq in Hz
fm      = 100;              // freq of modlating signal
dphi    = 10;               // separation b/w tx FM signal and demod echo signal in degrees

// Calculations
Tp      = dphi/(360*fm);    // round trip propagation time
R       = (c*Tp)/2;         // target range

// output
mprintf('Target Range = %3.2f Km',R/1000);
```

```
//
______________________________________________________________________
```

Scilab code Exa 10.2 Finding Target Range and Radial velocity

```
// Chapter 10 example 1
//
______________________________________________________________________

clc;
clear;
f        = 10*10^9;              // center freq. in Hz
f_us     = 60*10^3;              // upsweep freq. in Hz
f_ds     = 80*10^3;              // down sweep freq. in Hz
fm       = 100;                  // modulation freq. in Hz
B        = 2*10^6;               // sweep bandwidth in Hz
c        = 3*10^8;               // vel. of EM waves in m/s
T        = 5*10^-3;

// Calculations
fd       = (f_ds - f_us)/2;
df       = (f_ds + f_us)/2;
R        = (c*T*df)/(2*B);          // range in m
// fd    = (2*Vr*f)/c
Vr       = (c*fd)/(2*f);            // target radial velocity
Vr_kmph  = Vr*(18/5);               // target radial velocity in kmph
Vr_nmph  = Vr_kmph/1.85;            // target radial velocity in Nautical miles per hour

```

```
// Output
mprintf('Target Range = %3.2f Km\n Radial velocity =
    %3.1f Nmi/hr',R/1000,Vr_nmph);
//
    ------------------------------------------------------------------------
```

Scilab code Exa 10.3 `Finding error in doppler shift measurement`

```
// Chapter 10 example 3
//
    ------------------------------------------------------------------------

clc;
clear;
Vr = 150
c  = 3*10^8
df1= 10^6;
// Given data
// fd   = (2*Vr)/lamda  = (2*Vr*f)/c
// for 'Vr' and 'c' as constant(for a given radial
    velocity,Vr is constant)
// fd   = K.f where 'f' is the operating frequency
    and K = (2*Vr)/c
// Therefore df =    1 Mhz around the center
    frequency
k = (2*Vr)/c
df_d = df1*k

// Output
mprintf('Doppler shift due to carrier frequency
    sweep =  %d  Hz',df_d);
//
    ------------------------------------------------------------------------
```

Scilab code Exa 10.4 Finding Range and radial velocity

```
// Chapter 10 example 4
//
    ------------------------------------------------------------------------------
clc;
clear;

// Given data
f        = 10*10^9;            // operating frequency in Hz
f_us     = 100*10^3;           // upsweep freq
f_ds     = 100*10^3;           // downsweep freq
Tus      = 5*10^-3;            // up-sweep period
Tds      = 5*10^-3;            // down-sweep period
T        = 10*10^-3
B        = 10*10^6;            // sweep bandwidth
c        = 3*10^8;             // vel of EM waves in m/s
f_us_b   = 80*10^3;            // upsweep freq in fig b
f_ds_b   = 50*10^3;            // downsweep freq in fig b
f_us_c   = 50*10^3;            // upsweep freq in fig b
f_ds_c   = 80*10^3;            // downsweep freq in fig b

// Calculations
// a
fd       = (f_us - f_ds)/2;        // doppler shift
df       = (f_us + f_ds)/2;        // freq diff
Vr_a     = (c*fd)/(2*f);           // radial velocity
R        = (c*Tus*df)/(2*B);         // Range
if Vr_a == 0 then
```

```
    mprintf('Case a:\n Radial velocity = %d \n Range = %3.3f Km\n',Vr_a,R/1000);
end
// b
fd      = (f_us_b - f_ds_b)/2;      // doppler shift
df_b    = (f_us_b + f_ds_b)/2;      // freq difference due to range
R_b     = (c*T*df_b)/(2*B);        // Range
Vr_b    = (c*fd)/(2*f);           // radial velocity
mprintf(' Case b:\n Radial velocity = %3.2fm/s \n Range = %3.3f Km\n',Vr_b,R_b/1000');
mprintf(' As the up-sweep frequency difference is less than downspeed freq diff, this implies that doppler shift is\n contributing towards an increase in the echo signal freq. so, target is moving towards radar\n')
// c
fd      = (f_us_c - f_ds_c)/2;      // doppler shift
df_c    = (f_us_c + f_ds_c)/2;      // freq difference due to range
R_c     = (c*T*df_c)/(2*B);        // Range
Vr_c    = (c*fd)/(2*f);           // radial velocity
mprintf(' Case c:\n Radial velocity = %3.2f m/s \n Range = %3.3f Km\n',abs(Vr_c),R_c/1000');
mprintf(' As the up-sweep frequency difference is greater than downspeed freq diff, this implies that doppler shift is\n contributing towards an decrease in the echo signal freq. so, target is moving away from radar')
```

Scilab code Exa 10.5 Finding radial velocity

```
// Chapter 10 example 5
//____________________________________________________________________________
```

```
clc;
clear;

// Given data
f       = 10*10^9;          // operating freq in Hz
PRF     = 1000;             // pulse rep. rate
Vr      = 1000;             // radial velocity
c       = 3*10^8;           // vel. of EM waves in m/s

// Calculations
fd      = (2*Vr*f)/c        // true doppler shift
fA1     = modulo( modulo(fd,PRF)-PRF,PRF)
fA2     = modulo( modulo(fd,PRF)+PRF,PRF)
if fA1 < fA2 then
    fd  = fA1;          // apparent doppler shift
else
    fd  = fA2;          // apparent doppler shift
end
Vr      = (c*fd)/(2*f);     // radial velocity in m/s

//output
mprintf('Radial velocity = %3.2f m/s\n The radar measures the target to be moving away from the radial velocity at %3.2f m/s though in reality\n it is moving towards the radar with a velocity of 1000 m/s',Vr,abs(Vr));
//____________________________________________________________________________
```

Scilab code Exa 10.9 `Finding lowest blind speed`

```
// Chapter 10 example 9
//------------------------------------------------------------------------------
clc;
clear;
//Given data
lamda    = 3*10^-2;        // Operating Wavelength in m
PRF      = 2000;           // pulse rep. freq in Hz
n        = 1;              // for lowest blind speed

// Calculations
LBS      = (n*lamda*PRF)/2; // lowest blind speed
Vb_kmph = LBS*(18/5)        // lowest blind speed in kmph

// Output
mprintf('Lowest Blind speed = %d Kmph',Vb_kmph)
//------------------------------------------------------------------------------
```

Scilab code Exa 10.10 `Finding ratio of operating frequencies`

```
// Chapter 10 example 10
//------------------------------------------------------------------------------
clc;
clear;
//Given data
disp('Let the operating frequency of first radar = f1');
disp('Let the operating frequency of second radar = f2');
```

```
disp('Third blind speed of Radar = (3*c/(2*f1)')
disp('fifth blind speed of Radar = (5*c/(2*f1)')
disp('(3*c/(2*f1) = (5*c/(2*f1)');
disp('(f1/f2) = 3/5')
```

Scilab code Exa 10.11 Finding Apparent Range

```
// Chapter 10 example 11
//
   ------------------------------------------------------------------------------
clc;
clear;
//Given data
R        = 100;           // Range in kms
PRR      = 10*10^3;       // pulse rep. rate in Hz
c        = 3*10^5;        // vel. in km/s

// Calculations
PRI      = 1/PRR          // pulse rep. interval
Ra       = modulo(R,(c*PRI/2));       // apparent
   range in km

// Output
mprintf('Apparent Range = %d Km\n',Ra);
//
   ------------------------------------------------------------------------------
```

Scilab code Exa 10.12 Finding true range

```
// Chapter 10 example 12
```

```
//
------------------------------------------------------------------------
clc;
clear;
//Given data
Ra       = 25;        // Apparent Range in km
PRF      = 2000;      // Pulse rep. freq.
c        = 3*10^5;    // vel. of EM waves in km/s
Nr       = 3;         // Range zone

// Calculations
R        = Ra + ((c/2)*((Nr - 1)/PRF))    // true range in km

// Output
mprintf('True Range of the target = %d Km',R);
//
------------------------------------------------------------------------
```

Scilab code Exa 10.13 Estimating true range

```
// Chapter 10 example 13
//
------------------------------------------------------------------------
clc;
clear;
//Given data
PRF_1    = 750;       // pulse rep. freq in Hz
PRF_2    = 1000;      // pulse rep. freq in Hz
PRF_3    = 1250;      // pulse rep. freq in Hz
Ra_1     = 100;       // Apparent range for PRF_1
Ra_2     = 50;        // Apparent range for PRF_2
```

```
Ra_3     = 20;         // Apparent range for PRF_3
c        = 3*10^5;     // Vel of EM waves in Km/s

// Calculations
for Nr = 1:6            // Nr = Radar Zones
    R1(Nr)       = Ra_1 + ((c/2)*((Nr - 1)/PRF_1))       // true range in km
     R2(Nr)       = Ra_2 + ((c/2)*((Nr - 1)/PRF_2))       // true range in km
      R3(Nr)       = Ra_3 + ((c/2)*((Nr - 1)/PRF_3))       // true range in km
end

// Output
mprintf('Possible True Range measurements for 750 PPS\n');
mprintf(' = %dkm \n',R1);
mprintf('Possible True Range measurements for 1000 PPS\n')
mprintf(' = %dkm \n',R2);
mprintf('Possible True Range measurements for 1250 PPS\n')
mprintf(' = %dkm \n',R3);
mprintf('The shortest possible range that has been measured at all PRFs is %d Km True Range = %d km',R1(3),R1(3))
```

Scilab code Exa 10.14 Finding compression ratio and width of compressed pulse

```
// Chapter 10 example 14
// ------------------------------------------------------------------------------
clc;
clear;
```

```
//Given data
Te        = 4       // Expanded pulse width in usec
f1        = 50      // RF freq in Mhz
f2        = 70      // RF freq in Mhz

// Calculations
B         = f2 - f1;        // Signal bandwidth
Tc        = 1/B;            // Compressed pulse width in us
CR        = Te/Tc           // compression ratio

// Output
mprintf('Compression Ratio = %d\n Width of compressed pulse = %3.2f us',CR,Tc);
//----------------------------------------------------------------------
```

Scilab code Exa 10.15 `Finding synthesised aperture and cross range resolution`

```
// Chapter 10 example 15
//----------------------------------------------------------------------
clc;
clear;
//Given data
f         = 10*10^9;        // operating freq in Hz
c         = 3*10^8;         // vel. of EM waves in m/s
Ae        = 2;              // Antenna aperture in m
R         = 10*10^3;        // Target Range in m

// Calculations
lamda     = c/f;            // Wavelength in m
bw3db     = lamda/2;        // 3dB beamwidth in radian
```

```
Leff     = bw3db * R;     // effective length
Xs       = (R*lamda)/(2*Leff);    // Cross range resolution

// Output
mprintf('Effective Length = %d m\n',Leff);
mprintf('Cross range resolution = %d m',Xs);
// ------------------------------------------------------------------------
```

Scilab code Exa 10.16 Finding round trip time

```
// Chapter 10 example 16
// ------------------------------------------------------------------------

clc;
clear;
//Given data
R        = 6000;      // Target Range
c        = 3*10^8;    // speed of light in m/s

// Calculations
t        = (2*R)/c;   // round trip time

// Output
mprintf('Round Trip time = %d us',t/10^-6);
// ------------------------------------------------------------------------
```

Scilab code Exa 10.17 Finding doppler shift

```
// Chapter 10 example 17
//
    ------------------------------------------------------------------------------
clc;
clear;
//Given data
v       = 250;      // velocity in m/s
lamda   = 10.6*10^-6 // operating wavelength
theta   = 60;        // angle of depression

// Calculations
Vr      = v*cos(theta*%pi/180);      // radial
    velocity
fd      = (2*Vr)/lamda;              // doppler shift

// Output
mprintf('Doppler Shift = %3.2f Mhz',fd*10^-6);
//
    ------------------------------------------------------------------------------
```

Scilab code Exa 10.18 Finding Range Resolution

```
// Chapter 10 example 18
//
    ------------------------------------------------------------------------------
clc;
clear;
//Given data
B       = 10^6;      // Bandwidth in Mhz
c       = 3*10^8;    // speed of light in m/s

// Calculations
```

```
RR       = c/(2*B);  // Range Resolution in m

// Output
mprintf('Range Resolution = %d m\n',RR);
// ----------------------------------------------------------------
```

Chapter 11

Satellites and Satellite Communications

Scilab code Exa 11.1 Finding orbital velocity

```
// Chapter 11 example 1
//
   ____________________________________________________________________________

clc;
clear;

// Given data
h      = 150;      // height of satellite from
   earth in km
G      = 6.67*10^-11;      // Gravitational
   constant
M      = 5.98*10^24;       // mass of the earth in
   kg
Re     = 6370;             // radius of earth in km

// Calculations
u      = G*M
V      = sqrt(u/((Re + h)*10^3))  // orbital
```

```
   velocity
V1       = V/1000;                   // orbital
   velocity in km/s

// Output
mprintf('Orbital velocity = %3.3f km/s',V1);
//
   ----------------------------------------------------------------------------
```

Scilab code Exa 11.2 `Finding orbital eccentricity`

```
// Chapter 11 example 2
//
   ----------------------------------------------------------------------------

clc;
clear;

// Given data
Ap_Pe_diff  = 30000;         // difference between
   apogee and perigee in Km
a           = 16000;         // semi major axis of
   orbit

// Calculations
e           =  Ap_Pe_diff/(2*a);    // Eccentricity

// Output
mprintf('Eccentricity = %3.2f',e);
//
   ----------------------------------------------------------------------------
```

Scilab code Exa 11.3 Finding relationship between orbital periods

```
// Chapter 11 example 3
//____________________________________________________________________________
clc;
clear;

// Given data
a1      = 18000;             // semi major axis of the elliptical orbits of satellite 1
a2      = 24000;             // semi major axis of the elliptical orbits of satellite 2

// Calculations
// T     = 2*%pi*sqrt(a^3/u);
//let K = T2/T1;
K       = (a2/a1)^(3/2);     // Ratio of orbital periods

// Output
mprintf('The orbital period of satellite-2 is %3.2f times the orbital period of satellite-1',K);
//____________________________________________________________________________
```

Scilab code Exa 11.4 Finding magnitude of velocity impulse

```
// Chapter 11 example 4
```

```
//------------------------------------------------------------------------------
clc;
clear;

// Given data
h         = 35800;              // height of satellite orbit from earth in km
G         = 6.67*10^-11;        // Gravitational constant
M         = 5.98*10^24;         // mass of the earth in kg
Re        = 6364;               // radius of earth in km
i         = 2;                  // inclination angle

// Calculations
u         = G*M
r         = Re+h
Vi        = sqrt(u/r*10^3)*tan(i*%pi/180);      // magnitude of velocity impulse
V         = Vi/1000;                            // magnitude of velocity impulse in m/s
// Output
mprintf('Magnitude of velocity impulse = %d m/s',V);
//------------------------------------------------------------------------------
```

Scilab code Exa 11.5 Finding maximum shadow angle and max daily eclipse duration

```
// Chapter 11 example 5
//------------------------------------------------------------------------------
```

```
clc;
clear;
// Given data
h        = 13622;         // ht of circular orbit from
    earth's surface
Re       = 6378;          // Radius of earth in km

// Calculations
R        = Re+h;          // Radius of circular orbit
pimax    = 180 - (2*acos(Re/R))*(180/%pi);    //
   Maximum shadow angle
eclipmax_time = (pimax/360)*24;              //
   maximum daily eclipse duration

// output
mprintf('maximum shadow angle = %3.1f  \n Maximum
   daily eclipse duration = %3.2f hours',pimax,
   eclipmax_time);
//
   ----------------------------------------------------------------------------
```

Scilab code Exa 11.6 Finding total time from first day of eclipse to last day of e

```
//Chapter 11 example 6
//
   ----------------------------------------------------------------------------

clc;
clear;
// Given data
h        = 35786;         // ht of geo.stationary
   orbit above earth surface
T        = 365;           // time in days
r        = 6378           // radius of earth in km
```

```
// ie(t) = 23.4*sin(2*%pi*t/T)
// for a circular orbit of 20000 km radius ,phi = 37.4 ,Therefore , the time from first day of eclipse to equinox is given by substituting ie(t) = 37.4/2 = 18.7
phi     = 37.4
ie      = (phi/2)*(%pi/180)
k       = 23.4*(%pi/180)
t       = (365/(2*%pi))*asin((ie/k))
// for geostationary orbit
phimax  = 180 - 2*(acos(r/(r+h)))*(180/%pi)
t_geo   = (365/(2*%pi))*asin((8.7*%pi/180)/k)

// Output
mprintf('Total time from first day of eclipse to last day of eclipse = %3.1f days\n Total time from first day of eclipse to last day of eclipse for geostationary orbit = %3.2f days',t,t_geo)
// ------------------------------------------------------------------------
```

Scilab code Exa 11.7 Finding centrifugal force

```
// Chapter 11 example 7
// ------------------------------------------------------------------------
clc;
clear;
// Given data
m       = 100;       // mass of satellite
V       = 8000;      // orbital velocity in m/s
Re      = 6370;      // radius of earth in Km
```

```
H        = 200;       // satellite height above earth surface

// Calculations
CF       = (m*V^2)/((Re+H)*10^3);       //centrifugal force

// output
mprintf('Centrifugal Force = %d Newtons',CF);
// ________________________________________________________________
```

Scilab code Exa 11.8 Finding semi major axis

```
// Chapter 11 example 8
// ________________________________________________________________

clc;
clear;
// Given data
Apogee   = 30000;        // Apogee pt of satellite elliptical orbit
Perige   = 1000;         // perigee pt of satellite elliptical orbit

// Calculations
a        = (Apogee + Perige)/2;  // semi major axis

// output
mprintf('Semi-major axis = %d Km',a);
// ________________________________________________________________
```

Scilab code Exa 11.9 Finding apogee perigee and orbit eccentricity

```
// Chapter 11 example 9
//
    ------------------------------------------------------------------------------
clc;
clear;
// Given data
farth   = 30000;          // farthest point in
   satellite elliptic eccentric orbit
closest = 200;            // closest point in
   satellite elliptic eccentric orbit
Re      = 6370;           // Radius of earth in km

// Calculations
Apogee  = farth + Re;     // Apogee in km
Perigee = closest + Re; // perigee in km
a       = (Apogee + Perigee)/(2);        // semi-
   major axis
e       = (Apogee - Perigee)/(2*a);      // orbit
   eccentricity

// Output
mprintf('Apogee = %d km\n Perigee = %d km\n orbit
   eccentricity = %3.3f',Apogee,Perigee,e);
//
    ------------------------------------------------------------------------------
```

Scilab code Exa 11.10 Finding apogee and perigee distances

```
// Chapter 11 example 10
//
    ------------------------------------------------------------------------------------------------
clc;
clear;
// Given data
e       = 0.5;       // orbit eccentricity
ae      = 14000;     // from fig. the distance from
    center of ellipse to the centre of earth

// Calculations
a       = ae/e;          // semi major axis
apogee  = a*(1 + e);     // Apogee in km
perige  = a*(1 - e);     // perigee in km

// output
mprintf('Apogee = %d km\n Perigee = %d km',apogee,
    perige);
//
    ------------------------------------------------------------------------------------------------
```

Scilab code Exa 11.11 Finding escape velocity

```
// Chapter 11 example 11
//
    ------------------------------------------------------------------------------------------------
clc;
clear;
// Given data
G       = 6.67*10^-11;      // Gravitational
    constant
M       = 5.98*10^24;       // mass of the earth in
```

```
   kg
Re       = 6370*10^3;         // radius of earth in m

// Calculations
u        = G*M
Vesc     = sqrt(2*u/Re);
Ves      = Vesc/1000;     // escape velocity in km/s

// Output
mprintf('Escape velocity = %3.1f km/s',Ves);
//
   ______________________________________________________________________________
```

Scilab code Exa 11.12 Finding orbital period

```
// Chapter 11 example 12
//
   ______________________________________________________________________________

clc;
clear;
// Given data
a        = 25000*10^3;        // semimajor axis in m
   from fig
G        = 6.67*10^-11;        // Gravitational
   constant
M        = 5.98*10^24;         // mass of the earth in
   kg
h = 0

// Calculations
u        = G*M;
T        = 2*%pi*sqrt((a^3)/u)
hr            = T/3600                          // conv.
```

```
    from sec to hrs and min
t           = modulo(T,3600)                // conv.
    from sec to hrs and min
mi          = t/60                          // conv.
    from sec to hrs and min

// Output
mprintf('Orbital time period = %d Hours %d minutes',
   hr,mi)
```

Scilab code Exa 11.13 Finding orbital time period velocity at apogee and perigee p

```
// Chapter 11 example 13
//
   ------------------------------------------------------------------------

clc;
clear;
// Given data
apogee  = 35000;        // farthest point in kms
perigee = 500;          // closest point in kms
r       = 6360;         // radius of earth in kms
G       = 6.67*10^-11   // gravitational constant
M       = 5.98*10^24;   // mass of earth in kgs
// calculations
//funcprot(0)
apogee_dist = apogee + r       // apogee distance in
    kms
perigee_dist= perigee+r ;       // perigee distance
   in kms
a           = (apogee_dist + perigee_dist)/2;   //
   semi-major axis of elliptical orbit
T           = (2*%pi)*sqrt((a*10^3)^3/(G*M));
               // orbital time period
hr          = T/3600                        // conv.
```

```
    from sec to hrs and min
t             = modulo(T,3600)                  // conv. from sec to hrs and min
mi            = t/60                            // conv. from sec to hrs and min
u             = G*M
Vapogee       = sqrt(u*((2/(apogee_dist*10^3)) - (1/(a*10^3))));      // velocity at apogee point
Vperigee      = sqrt((G*M)*((2/(perigee_dist*10^3)-(1/(a*10^3)))))    // velocity at perigee point

// Output
mprintf('Orbital Time Period = %d Hrs %d min \n Velocity at apogee = %3.3f Km/s\n Velocity at perigee = %3.3f Km/s',hr,mi,Vapogee/1000,Vperigee/1000)
mprintf('\n Note: Calculation mistake in textbook in finding velocity at apogee point')
```

Scilab code Exa 11.14 Finding target eccentricity

```
// Chapter 11 example 14
//------------------------------------------------------------------------------
clc;
clear;
// Given data
ra_S_rp      = 50000;        // sum of apogee and perigee distance
ra_D_rp      = 30000;        // difference of apogee and perigee distances

// Calculations
e            = ra_D_rp/ra_S_rp;  // eccentricity
```

```

// Output
mprintf('Target eccentricity = %3.1f',e);
// ------------------------------------------------------------------------------
```

Scilab code Exa 11.15 Finding apogee and perigee distances

```
// Chapter 11 example 15
// ------------------------------------------------------------------------------
clc;
clear;
// Given data
a        = 20000;              // semi major axis of elliptical sate. orbit in kms
b        = 16000;              // semi minor axis of elliptical sate. orbit in kms

// calculations
// a = (ra + rp)/2
// b = sqrt(ra*rp)
// let k = (ra + rp)
// let l = ra*rp
k        = 2*a;                // ra+ rp ----------------1
l        = b^2;                // ra*rp ----------------2
// ra^2 -40000ra + 256000000
p        = [ 1 -k l]
x        = roots(p)
r1       = x(1)
r2       = x(2)
```

```
rp = k - r1;
mprintf('Apogee distance = %d km\n Perigee distance
   = %d km',r1,rp);
//
   ____________________________________________________________________________
```

Scilab code Exa 11.16 `Finding max deviation in latitude and longitude`

```
// Chapter 11 example 16
//
   ____________________________________________________________________________

clc;
clear;
// Given data
H        = 35800;         // height of orbit in kms
re       = 6364;          // radius of earth in kms
i        = 2;             // angle of inclination in
   degrees

// Calculations
r        = H+re;          // radius of orbit in kms
lamdamax = i;             // max latitude deviation
long_dev = (i^2)/228;     // max. longitude deviation
disp_lamda = (r*i*%pi/180)// max disp in km due to
   lamdamax
max_disp1  = disp_lamda*(long_dev/lamdamax) // max
   disp.due to max.longitude deviation

// Output
mprintf('Maximum deviation in latitude = %d \n
   Maximum deviation in longitude = %3.4f  \n
   Maximum displacements due to latitude
   displacement = %d Km\n Maximum displacements due
```

```
    to longitude displacement = %3.1f Km\n',lamdamax,
    long_dev,disp_lamda,max_disp1 );
```

Scilab code Exa 11.17 Finding angle of inclination

```
// Chapter 11 example 17
//
    ------------------------------------------------------------------------------
clc;
clear;
// Given data
r          = 42164;         // orbital radius in kms
Dlamda_max = 500;           // max displacement due to
    latitude deviation

// Calculations
i    = Dlamda_max/r;        // angle of inclination in
    radians
i_deg = i*180/%pi           // rad to deg conv

// Output
mprintf('Angle of inclination = %3.2f  ',i_deg);
//
    ------------------------------------------------------------------------------
```

Scilab code Exa 11.18 Finding maximum coverage angle and max slant range

```
// Chapter 11 example 18
//
    ------------------------------------------------------------------------------
```

```
clc;
clear;
// Given data
H   = 35786;          // ht of orbit from earth surface
Re  = 6378            // radius of earth in kms

// Calculations
// For theoretical max coverage angle,elevation angle E = 0
E = 0
// max coverage angle = 2amax
// 2amax = 2asin(Re/(Re+H)cosE)
amax = 2*asin((Re/(Re+H))*cos(E))
amax_deg = amax*180/%pi      // rad to deg conversion
D   = sqrt( Re^2 + (Re+H)^2 - 2*Re*(Re + H)*asin(E + asin((Re/(Re+H))*cos(E)))) // Max slant range

// Output
mprintf('Maximum Coverage angle = %3.1f  \n Maximum slant Range = %d Km',amax_deg,D);
// ______________________________________________________________________________
```

Chapter 13

Microwave Communication link Basic Design Considerations

Scilab code Exa 13.1 Finding path length

```
//Chapter 13 example 1
//------------------------------------------------------------------------------
clc;
clear;
// Given data
f       = 6;                // microwave terrestrial comm link oper. freq in Ghz
D       = 50;               // single hop path length in miles
// mid way of path length
D1      = 25;
D2      = 25;
N       = 3;                // N value for third fresnal zone

// calculations
F1      = 72.2*((D1*D2)/(D*f))^0.5;      // first
```

```
fresnel zone
F3      = F1*sqrt(N);                    // Third fresnal zone

// Output
mprintf('First Fresnel zone distance = %3.1f feet\n Third Fresnel zone distance = %3.1f feet\n',F1,F3);
// ____________________________________________________________________________
```

Scilab code Exa 13.2 Finding max tolerable obstacle height

```
//Chapter 13 example 2
// ____________________________________________________________________________

clc;
clear;
// Given data
f       = 4.5;           // microwave terrestrial comm link oper. freq in Ghz
D       = 40;            // single hop path length in miles
hant    = 200;           // antenna ht. above surface of earth
// from fig
D1      = 5;
D2      = 35;
K       = 1;             // for normal case

// calculations
F1      = 72.2*((D1*D2)/(D*f))^0.5;      // first fresnel zone
```

```
// computing curvature 'h' of earth at a distance of 10 miles from Transmitter if given by (D1*D2)/(1.5*K)
h       = (D1*D2)/(1.5*K);                // curvature of earth in feet
PLabove = hant - h;                       // path line is PLabove feet above surface of earth
hmaxtol = PLabove - F1;                    // max tolerable height in feet

// Output
mprintf('Maximum tolerable height of obstacle above surface of earth = %3.1f feet',hmaxtol);
// ______________________________________________________________________________
```

Scilab code Exa 13.3 Finding whether first fresnal zone pass without any obstructi

```
//Chapter 13 example 3
// ______________________________________________________________________________

clc;
clear;
// Given data
f       = 4.5;           // microwave terrestrial comm link oper. freq in Ghz
D       = 40;            // single hop path length in miles
hant    = 200;           // antenna ht. above surface of earth
// from fig
D1      = 5;
D2      = 35;
```

```
K       = 2/3;              // K-factor

// calculations
F1      = 72.2*((D1*D2)/(D*f))^0.5;     // first fresnel zone
// computing curvature 'h' of earth at a distance of 10 miles from Transmitter if given by (D1*D2)/(1.5*K)
h       = (D1*D2)/(1.5*K);                // curvature of earth in feet
PLabove = hant - h;                       // path line is PLabove feet above surface of earth
if PLabove < F1 then
   mprintf('Available clearance above the surface of earth = %d feet\n Required first fresnal zone clearance = %3.1f feet,So it would be obstructed',PLabove,F1 )
end
//
```

Scilab code Exa 13.4 Finding outrage time

```
//Chapter 13 example 4
//

clc;
clear;
// Given data
UF      = 2*10^-4;       // unavailability factor

// Calculations
outrage_t = UF*8760;     // outrage time in hours per
```

```
    year

// Output
mprintf('Outrage time = %3.3f hours per year',
    outrage_t);
//
    ________________________________________________________________________
```

Scilab code Exa 13.5 Finding improvement in probability of fade margin

```
//Chapter 13 example 5
//
    ________________________________________________________________________

clc;
clear;
// Given data
PL      = 50;        // path length in miles from fig
FM      = 40;        // fade margin in dB
P_fm_ex = 7*10^-5;   // prob. of fade margin getting
    exceeding
P_fm_ex_50db    = 6*10^-6;  // prob. of fade margin
    getting exceeding for fade margin 50dB
p_fig_30m_40db  = 2*10^-5;  // prob fig for patl
    length of 30miles and fade margin 40dB

// Calculations
impr_prob_a = P_fm_ex/P_fm_ex_50db;    //
    improvement in prob. of fade margin for a
impr_prob_b = P_fm_ex/p_fig_30m_40db   //
    improvement in prob. of fade margin for b

// Output
mprintf('(a):\n Improvement in probability of fade
```

```
margin = %3.1f\n (b):\n Improvement in probability of fade margin = %3.1f\n',impr_prob_a,impr_prob_b);
//
```

Scilab code Exa 13.6 Finding unavailability factor

```
//Chapter 13 example 6
//
clc;
clear;
// Given data
UF_sh    = 0.01;      // unavail. factor for single hop
IF_SD    = 100;       // improvement factor due to space diversity

// Calculations
UF_4hl   = 4* UF_sh/100;      // unavail. factor for 4 hop link and conv from %
UF       = UF_sh/(100*IF_SD);// unavail. factor for single hop link if it employs space diversity

// Output
mprintf('unavail. factor for 4 hop link = %3.4f\n unavail. factor for single hop link if it employs space diversity = %3.0e',UF_4hl,UF);
//
```

Scilab code Exa 13.7 `Finding Outrage Time`

```
//Chapter 13 example 7
//
   ------------------------------------------------------------------
clc;
clear;
// Given data
f        = 3.5;              // operating freq. of
   microwave link in Ghz
D        = 30;               // single hop path
   length in miles
a        = 1;                // roughness
b        = 0.5;              // humid climate
F        = 40;               // fade margin in dB

// Calculations
U        = a*b*2.5*10^-6 *f*D^3 *10^(-F/10);     //
   unavailability factor
U1       = U*525600;                              //
   unavailabilty factor in minutes per year
U4       = U1*4;                                  //
   unavailabilty factor for 4-hop link
// Output
mprintf('Outage Time = %3.1f minutes per year',U4);
//
   ------------------------------------------------------------------
```

Scilab code Exa 13.8 `Finding change in value of unavailability Factor`

```
//Chapter 13 example 8
//
   ________________________________________________________________________________

clc;
clear;
// Given data
// D2 = 2*D1              // path length is doubled
// F2 = F1+10;            // fade margin is increased
   by 10dB
// f2 = 1.25f1            // frequency operation
   increased by 25 %

//(U1/U2)  = (f1* D1^3 * 10^(-F1/10))/ (f1* D1^3 *
   10^(-F1/10))
// sub above values
//(U1/U2)  = (f1* D1^3 * 10^(-F1/10)) / (1.25*f1*8*
   D1^3*10^(-F1/10)*10^-1) = 1
mprintf('Unavailability Factor remains unaltered');
//
   ________________________________________________________________________________
```

Scilab code Exa 13.9 Finding improvement in outrage time

```
//Chapter 13 example 9
//
   ________________________________________________________________________________

clc;
clear;
mprintf('The improvement factor is proportional to
   square of antenna spacing.Therefore,it will
   increase by a factor of 4\n Consequently,the
   unavailability factor and hence the outrage time
```

```
   will also reduce by a factor of 4');
//
   ____________________________________________________________________
```

Scilab code Exa 13.10 Finding composite Fade margin

```
//Chapter 13 example 10
//
   ____________________________________________________________________

clc;
clear;
// Given data
DFM     = 40;       // dispersive fade margin
FFM     = 30;       // flat fade margin

// Calculations
CFM     = -10*log10(10^(-FFM/10) + 10^(-DFM/10));

// Output
mprintf('Composite Fade Margin = %3.2f dB\n',CFM);
mprintf(' minus sign is wrongly printed in Textbook'
   );
//
   ____________________________________________________________________
```

Scilab code Exa 13.11 proof

```
//Chapter 13 example 11
```

```
//
   ________________________________________________________________________
clc;
clear;
// Given data
DFM1    = 50;       // dispersive fade margin
FFM     = 30;       // flat fade margin
DFM2    = 40;       // dispersive fade margin

// Calculations
CFM1    = -10*log10(10^(-FFM/10) + 10^(-DFM1/10));
CFM2    = -10*log10(10^(-FFM/10) + 10^(-DFM2/10));
d_CFM   = CFM1 -CFM2;

// Output
mprintf('CFM increases by %3.2f dB for a 10 dB
   increase in DFM',d_CFM);
//
   ________________________________________________________________________
```

Scilab code Exa 13.12 Finding outrage time

```
//Chapter 13 example 12
//
   ________________________________________________________________________
clc;
clear;
// Given data
f       = 23;               // operating freq. of
   microwave link in Ghz
D       = 10;               // single hop path
   length in miles
```

```
a        = 1;                    // topographic factor
b        = 0.5;                  // climatic factor
DFM      = 40;                   // dispersive fade
    margin
FFM      = 30;                   // flat fade margin

// Calculations
CFM      = -10*log10(10^(-FFM/10) + 10^(-DFM/10));
    // composite fade margin
U        = a*b*2.5*10^-6 *f*D^3 *10^(-CFM/10);      //
     unavailability factor
U1       = U*525600;                                //
     outrage time in min per year

// Output
mprintf('Outrage time = %3.2f minutes per year',U1);
//
   ------------------------------------------------------------------------
```

Scilab code Exa 13.13 Finding improvement in MTBF

```
//Chapter 13 example 13
//
   ------------------------------------------------------------------------

clc;
clear;
// Given data
MTBF2     = 20000;        // microwave Tx output MTBF
    figure
MTBF3     = 60000;        // power amplifier portion
    of MTBF

// Calculations
```

```
MTBF1    = (MTBF2*MTBF3)/(MTBF3-MTBF2);
impr     = MTBF1-MTBF2       // improvement in MTBF
   if power amplifier not used

// output
mprintf('Improvement in MTBF of transmitter if power
    amplifier is not used = %d hours',impr);
//
   ------------------------------------------------------------------------
```
