# Scilab Textbook Companion for Applied Physics-i by I. A. Shaikh[1]

Created by
Sanket Bapu Sakharkar
B.E
Others
Mumbai University
College Teacher
Mrugendra Vasmatkar
Cross-Checked by
K. V. P. Pradeep

May 25, 2016

[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:** Applied Physics-i

**Author:** I. A. Shaikh

**Publisher:** Tech-max Publication, Pune

**Edition:** 7

**Year:** 2013

**ISBN:** 9789350770641

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

# Crystallography

**Scilab code Exa 1.3.1** calculate Unit cell dimension

```
1 //Chapter-1,Example1_3_1,pg 1-14
2
3 A=26.98                                   //atomic
      weight of Al
4
5 N=6.023*10^26                             //Avogadro's
      number
6
7 p=2700                                    //Density
8
9 n=4                                       //FCC
      structure
10
11 a=(n*A/(N*p))^(1/3)
12
13 printf("Unit cell dimension of Al=")
14
15 disp(a)
16
17 printf("m")
```

**Scilab code Exa 1.3.2** calculate density

```
1 //Chapter-1,Example1_3_2,pg 1-15
2
3 As=28.1                                   //atomic
     weight of Si
4
5 Ag=69.7                                   //atomic
     weight of Ga
6
7 Aa=74.9                                   //atomic
     weight of As
8
9 as=5.43*10^-8                             //lattice
     constant of Si
10
11 aga=5.65*10^-8                           //lattice
     constant of GaAs
12
13 ns=8                                     //no of
     atoms/unit cell in Si
14
15 nga=4                                    //no of
     atoms/unit cell in GaAs
16
17 N=6.023*10^23                            //Avogadro's
     number
18
19 //p=(n*A)/(N*a^3) this is formula for density
20
21 //for Si
22
23 ps=(ns*As)/(N*as^3)
24
```

```
25 printf("            1) Density of Si=")
26
27 disp(ps)
28
29 printf("gm/cm^3")
30
31 //for GaAs
32
33 Aga=Ag+Aa                                    //molecular
      wt of GaAs
34
35 pga=(nga*Aga)/(N*aga^3)
36
37 printf("     2) Density of GaAs=")
38
39 disp(pga)
40
41 printf("gm/cm^3")
```

**Scilab code Exa 1.3.3** calculate density

```
1 //Chapter-1,Example1_3_3,pg 1-16
2
3 A=63.5                                       //atomic
      weight of Cu
4
5 N=6.023*10^23                                //Avogadro's
      number
6
7 n=4                                          //FCC
      structure
8
9 r=1.28*10^-8                                 //atomic
      radius of Cu
10
```

```
11 //for FCC
12
13 a=4*r/(sqrt(2))                                    //lattice
      constant
14
15 p=(n*A)/(N*a^3)
16
17 printf("Density of Cu=")
18
19 disp(p)
20
21 printf("gm/cm^3")
```

**Scilab code Exa 1.3.4** calculate APF

```
1 //Chapter-1,Example1_3_4,pg 1-17
2
3 A=50                                               //atomic
      weight of chromium
4
5 N=6.023*10^23                                     //Avogadro's
       number
6
7 p=5.96                                            //Density
8
9 n=2                                               //BCC
      structure
10
11 //step 1 : claculation for lattice constant (a)
12
13 a=(n*A/(N*p))^(1/3)
14
15 //step 2 : radius of an atom in BCC
16
17 r=sqrt(3)*a/4
```

```
18
19 //step 3 : Atomic packing factor (APF)
20
21 APF=n*((4/3)*%pi*r^3)/a^3
22
23 printf("Atomic packing factor (APF)=")
24
25 disp(APF)
```

**Scilab code Exa 1.3.5** calculate no of unit cell

```
1  //Chapter-1,Example1_3_5,pg 1-17
2
3  A=120                                    //atomic
       weight of chromium
4
5  N=6.023*10^23                            //Avogadro's
       number
6
7  p=5.2                                    //Density
8
9  n=2                                      //BCC
       structure
10
11 m=20                                     //mass
12
13 //step 1 : claculation for volume of unit cell(a^3)
14
15 a=(n*A/(N*p))
16
17 //step 2 : volume of 20 gm of the element
18
19 v=m/p
20
21 //step 3 :no of unit cell
```

```
22
23 x=v/a
24
25 printf("no of unit cell=")
26
27 disp(x)
```

**Scilab code Exa 1.3.6** calculate no of atoms per meter cube

```
1  //Chapter-1,Example1_3_6,pg 1-18
2
3  A=132.91                                    //atomic
       weight of chromium
4
5  N=6.023*10^26                               //Avogadro's
       number
6
7  p=1900                                      //Density
8
9  a=6.14*10^-10                               //lattice
       constant
10
11 //step 1 : type of structure
12
13 n=(p*N*a^3)/A
14
15 printf("n =")
16
17 disp(round(n))
18
19 printf("BCC structure")
20
21 //step 2: no of atoms/m^3
22
23 x=n/a^3
```

```
24
25 printf("          no of atoms/m^3=")
26
27 disp(x)
```

**Scilab code Exa 1.3.7** calculate no of unit cell

```
1  //Chapter-1,Example1_3_6,pg 1-18
2
3  a=0.4049*10^-9                          //lattice
       constant
4
5  t=0.006*10^-2                           //thickness
       of Al foil
6
7  A=50*10^-4                              //Area of
       foil
8
9  V1=a^3                                  //volume of
       unit cell
10
11 V=A*t                                   //volume of
       the foil
12
13 N=V/V1                                  //no of unit
       cell in the foil
14
15 printf("no of unit cell in the foil=")
16
17 disp(N)
```

**Scilab code Exa 1.5.1** calcukate critical radius ratio of ligancy three

```
1  //Chapter-1,Example1_5_1,pg 1-29
2
3  //refer diagram from textbook
4
5  //on joining centre of 3 anions,an equilateral
       triangle is formed and on joining centres of any
       anion and cation  a right angle triangle ABC os
       formed
6
7  //where AC=rc+ra
8
9  //and BC=ra
10
11 //m(angle (ACB))=30 degree
12
13 //therefore cos (30)=ra/(rc+ra)
14
15 //assume rc/ra=r
16
17 r=(1-cosd(30))/cosd(30)                    //by
       arrangimg terms we get value of r
18
19 printf("critical radius ratio of ligancy 3=")
20
21 disp(r)
```

**Scilab code Exa 1.5.2** calculate critical radius ratio for ligancy six

```
1  //Chapter-1,Example1_5_2,pg 1-30
2
3  //refer diagram from textbook
4
5  //in the said arrangement a cation is squeezed into
       4 anions in a plane and 5th anion is in upper
       layer and 6th in bottom layer
```

```
6
7 //join cation anion centres E and B and complete the
      triangle EBF
8
9 //in triangle EBF m(angle F)=90 and EF=BF
10
11 //m(angle B)=m(angle E)=45
12
13 //and EB=rc+ra and BF=ra
14
15 //cos(45)=ra/(rc+ra)
16
17 //assume rc/ra=r
18
19 r=(1-cosd(45))/cosd(45)                    //by
     arrangimg terms we get value of r
20
21 printf("critical radius ratio for ligancy 6 =")
22
23 disp(r)
```

**Scilab code Exa 1.5.3** calculate critical radius ratio for octohedral

```
1 //Chapter-1,Example1_5_3,pg 1-30
2
3 //refer diagram from textbook
4
5 //since plane is square hence it is same as ligancy
     6
6
7 //in the said arrangement a cation is squeezed into
     4 anions in a plane and 5th anion is in upper
     layer and 6th in bottom layer
8
9 //join cation anion centres E and B and complete the
```

```
    triangle EBF
10
11 //in triangle EBF m(angle F)=90 and EF=BF
12
13 //m(angle B)=m(angle E)=45
14
15 //and EB=rc+ra and BF=ra
16
17 //cos(45)=ra/(rc+ra)
18
19 //assume rc/ra=r
20
21 r=(1-cosd(45))/cosd(45)                    //by
    arrangimg terms we get value of r
22
23 printf("critical radius ratio for ligancy 8 =")
24
25 disp(r)
```

**Scilab code Exa 1.5.4** calculate critical radius ratio for ligancy 4

```
1 //Chapter-1,Example1_5_4,pg 1-31
2
3 //a tetrahedron CAEH can be considered with C as the
    apex of the tetrahedron.
4
5 //the edges AE,AH and EH of the tetrahedron will
    then be the face of the cube faces ABEF,ADHF,EFHG
    resp.
6
7 //from fig
8
9 //AO=ra+rc and AJ=ra
10
11 //AE=root(2)*a and AG=root(3)*a
```

```
12
13 //AO/AJ=AG/AE=(ra+rc)/ra=root(3)*a/root(2)*a
14
15 //assume rc/ra=r
16
17 r=(sqrt(3)-sqrt(2))/sqrt(2)
18
19 printf("critical radius ratio for ligancy 4 =")
20
21 disp(r)
```

**Scilab code Exa 1.5.5** calculate critical radius ratio for ligancy 8

```
1 //Chapter-1,Example1_5_5,pg 1-32
2
3 //ligancy 8 represents cubic arrangment .8 anions
    are at the corners and touch along cube edgs.
    Along the body diagonal the central cation and
    the corner anion are in contact.
4
5 //cube edge=2*ra
6
7 //refer diagram from textbook
8
9 //and body diagonal=root(3)*cube edge=root(3)[2*(rc+
    ra)]
10
11 //assume rc/ra=r
12
13 r=sqrt(3)-1
14
15 printf("critical radius ratio of ligancy 8=")
16
17 disp(r)
```

**Scilab code Exa 1.5.6** calculate critical radius ratio for ligancy

```
1 //Chapter-1,Example1_5_6,pg 1-32
2
3 //for an ionic crystal exibiting HCP structure the
    arrangment of ions refere from textbook
4
5 //at centre we have a cation with radius rc=OA
6
7 //it is an touch with 6 anions with radius ra=AB
8
9 //OB=OC=ra+rc
10
11 //intrangle ODB ,m(angle (OBC))=60 degree ,m(angle (
    ODB))=90 degree
12
13 //therefore cos(60)=BD/OB=AB/(OA+OB)=ra/(rc+ra)
14
15 //assume rc/ra=r
16
17 r=(1-cosd(60))/cosd(60)                    //by
    arrangimg terms we get value of r
18
19 printf("critical radius ratio 0f HCP structure=")
20
21 disp(r)
```

---

**Scilab code Exa 1.6.2** calculate miller indices of plane

```
1 //Chapter-1,Example1_6_2,pg 1-35
2
3 //intercept of planeare in proportion a,b/3,2*c
```

```
4
5 //as a,b and c are basic vectors the proportin of
    intercepts 1:1/3:2
6
7 //therefore reciprocal
8
9 r1=1
10
11 r2=3
12
13 r3=1/2
14
15 //taking LCM
16
17 v=int32([2,1])
18
19 l=double(lcm(v))
20
21 m1=(l*r1)
22
23 m2=(l*r2)
24
25 m3=(l*r3)
26
27 printf("miler indices=")
28
29 disp(m3,m2,m1)
```

**Scilab code Exa 1.6.4** calculate interplanar spacing

```
1 //Chapter-1,Example1_6_4,pg 1-38
2
3 r=1.414                                    //atomic
    radius in amstrong unit
4
```

```
5  //for FCC structure
6
7  a=4*r/sqrt(2)
8
9  //part 1: plane(2,0,0)
10
11 //the interplanar spacing of plane
12
13 h1=2
14
15 k1=0
16
17 l1=0
18
19 //we know that d=a/sqrt(h^2+k^2+l^2)
20
21 d1=a/sqrt(h1^2+k1^2+l1^2)
22
23 printf("              1)interplanar spacing for
      (2,0,0) plane=")
24
25 disp(d1)
26
27 printf("amstrong")
28
29 //part 2: plane(1,1,1)
30
31 //the interplanar spacing of plane
32
33 h2=1
34
35 k2=1
36
37 l2=1
38
39 //we know that d=a/sqrt(h^2+k^2+l^2)
40
41 d2=a/sqrt(h2^2+k2^2+l2^2)
```

```
42
43 printf("     2)interplanar spacing for(1,1,1) plane="
   )
44
45 disp(d2)
46
47 printf("amstrong")
```

**Scilab code Exa 1.14.1** calculate lattice constant

```
1 //Chapter-1,Example1_14_1,pg 1-58
2
3 n=4                                          //FCC
   structure
4
5 ro=2180                                      //density of
    NaCl
6
7 M=23+35.5                                    //molecular
   weight of NaCl
8
9 N=6.023*10^26                                //Avogadro's
    number
10
11 a=((n*M)/(N*ro))^(1/3)
12
13 printf("Lattice constant=")
14
15 disp(a)
16
17 printf("m")
```

**Scilab code Exa 1.14.2** calculate Lattice constant and diameter

```
1  //Chapter-1,Example1_14_2,pg 1-58
2
3  n=4                                          //FCC
       structure
4
5  ro=8.9                                       //density of
       Cu atom
6
7  A=63.55                                      //atomic
       weight of Cu atom
8
9  N=6.023*10^23                                //Avogadro's
       number
10
11 a=((n*A)/(N*ro))^(1/3)
12
13 printf("          1) Lattice constant=")
14
15 disp(a)
16
17 printf("cm")
18
19 r=sqrt(2)*a/4                                //radius of
       Cu atom
20
21 d=2*r                                        //diameter
       of Cu atom
22
23 printf("        2) Diameter of Cu atom=")
24
25 disp(d)
26
27 printf("cm")
```

**Scilab code Exa 1.14.3** calculate Density of diamond

```
1 //Chapter-1,Example1_14_3,pg 1-59
2
3 n=8                                          //diamond
     structure
4
5 A=12.01                                     //atomic wt
6
7 N=6.023*10^23                               //Avogadro's
     number
8
9 a=3.75*10^-8                                //lattice
     constant of diamond
10
11 ro=(n*A)/(N*(a^3))
12
13 printf("Density of diamond=")
14
15 disp(ro)
16
17 printf("gm/cc")
```

**Scilab code Exa 1.14.4** calculate miller indices

```
1 //Chapter-1,Example1_14_4,pg 1-59
2
3 //intercept of planeare in proportion 3a:4b:infinity
     (plane parallel to z axis)
4
5 //as a,b and c are basic vectors the proportin of
     intercepts 3:4:infinity
6
7 //therefore reciprocal
8
9 r1=1/3
10
```

```
r2=1/4

r3=0

//taking LCM

v=int32([3,4])

l=double(lcm(v))

m1=(l*r1)

m2=(l*r2)

m3=(l*r3)

printf("miler indices=")

disp(m3,m2,m1)
```

**Scilab code Exa 1.14.5** calculate miller indices

```
//Chapter-1,Example1_14_5,pg 1-59

//intercept of planeare in proportion 3a:-2b:3/2c

//as a,b and c are basic vectors the proportin of
   intercepts 3:-2:3/2

//therefore reciprocal

r1=1/3

r2=-1/2

```

```
13 r3=2/3
14
15 //taking LCM
16
17 v=int32([3,2,3/2])
18
19 l=double(lcm(v))
20
21 m1=(l*r1)
22
23 m2=(l*r2)
24
25 m3=(l*r3)
26
27 printf("miler indices=")
28
29 disp(m3,m2,m1)
```

**Scilab code Exa 1.14.6** calculate ratio of intercepts

```
1 //Chapter-1,Example1_14_6,pg 1-59
2
3 //if a plane cut at length m,n,p on the three
    crystal axes,then
4
5 //m:n:p=xa:yb:zc
6
7 //when primitive vectors of unit cell and numbers x,
    y,z,are related to miller indices (h,k,l)of the
    plane by relation
8
9 //1/x:1/y:1/z=h:k:l
10
11 //since a=b=c (crystal is simple cubic)
12
```

```
13 //and (h,k,l)=(1,2,3)
14
15 //therefore reciprocal
16
17 r1=1/1
18
19 r2=1/2
20
21 r3=1/3
22
23 //taking LCM
24
25 v=int32([1,2,3])
26
27 l=double(lcm(v))
28
29 m=(l*r1)
30
31 n=(l*r2)
32
33 p=(l*r3)
34
35 printf("ratio of intercepts=")
36
37 disp(m)
38
39 disp(n)
40
41 disp(p)
```

**Scilab code Exa 1.14.7** calculate y and z intercepts

```
1 //Chapter-1,Example1_14_7,pg 1-60
2
3 //primitive vectors
```

```
4
5 a=1.2                                          //in
     amstrong unit
6
7 b=1.8                                          //in
     amstrong unit
8
9 c=2                                           //in amstrong
      unit
10
11 //miller indices of the plane
12
13 h=2
14
15 k=3
16
17 l=1
18
19 //therefore intercepts are a/h,b/k,c/l
20
21 x=a/h
22
23 y=b/k
24
25 z=c/l
26
27 //this gives intercepts along x axis as x amstrong
     but it is given tthat plane cut x axis at 1.2
     amstrong .
28
29 t=1.2/x
30
31 //this shows that the plane under consideration is
     another plane which is parallel to it(to keep
     miller indices same)
32
33 n=t*y                                           //Y
     intercept
```

```
34
35 p=t*z                                          //Z
       intercept
36
37 printf("            1) Y intercept=")
38
39 disp(n)
40
41 printf("amstrong")
42
43 printf("    2)Z intercept=")
44
45 disp(p)
46
47 printf("amstrong")
```

**Scilab code Exa 1.14.8** calculate radius

```
1  //Chapter-1,Example1_14_8,pg 1-61
2
3  //the interplanar spacing of plane
4
5  h=1
6
7  k=1
8
9  l=0
10
11 d=2                                          //interpanar
       spacing in amstrong unit
12
13 //we know that d=a/sqrt(h^2+k^2+l^2) therefore
14
15 a=d*sqrt(h^2+k^2+l^2)
16
```

```
17 //for FCC structure
18
19 r=sqrt(2)*a/4
20
21 printf("radius r=")
22
23 disp(r)
24
25 printf("amstrong")
```

**Scilab code Exa 1.14.9** calculate density and diameter

```
1  //Chapter-1,Example1_14_9,pg 1-61
2
3  n=4                                          //for FCC
       structure
4
5  //the interplanar spacing of plane
6
7  h=1
8
9  k=1
10
11 l=1
12
13 d=2.08*10^-10                                //distance
14
15 A=63.54                                      //atomic
       weight of Cu
16
17 N=6.023*10^26                                //amstrong
       no
18
19 //we know that d=a/sqrt(h^2+k^2+l^2) therefore
20
```

```
a=d*sqrt(h^2+k^2+l^2)

//also (a^3*q)=n*A/N

q=n*A/(N*a^3)

printf("                1)density=")

disp(q)

printf("kg/m^3")

//for FCC structure

r=sqrt(2)*a/4

d=r*2

printf("          2)radius r=")

disp(r)

printf("m")

printf("              3)diameter d=")

disp(d)

printf("m")
```

**Scilab code Exa 1.14.10** calculate free electron concentration

```
//Chapter-1,Example1_14_10,pg 1-62

A=63.546                                    //atomic
```

```
    weight of Cu
4
5 N=6.023*10^26                                    //Avogadro's
     number
6
7 p=8930                                           //Density
8
9 n=1.23                                           //no.of
    electron per atom
10
11 //density=mass/volume
12
13 //therfore 1/volume=density/mass
14
15 //since electron concentration is needed, let us
    find out no of atoms/volume(x)
16
17 x=N*p/A
18
19 //now one atom contribute n=1.23 electron
20
21 //therefore x atoms contribute  y no of free
    electron
22
23 y=x*n
24
25 printf("free electron concentration=")
26
27 disp(y)
28
29 printf("electron/m^3")
```

**Scilab code Exa 1.14.11** calculate Y and Z intercept

```
1 //Chapter-1,Example1_14_11,pg 1-62
```

```
2
3 //primitive vectors
4
5 a=1.5                                          //in
     amstrong unit
6
7 b=2                                            //in
     amstrong unit
8
9 c=4                                           //in amstrong
      unit
10
11 //miller indices of the plane
12
13 h=3
14
15 k=2
16
17 l=6
18
19 //therefore intercepts are a/h,b/k,c/l
20
21 x=a/h
22
23 y=b/k
24
25 z=c/l
26
27 //this gives intercepts along x axis as x amstrong
     but it is given that plane cut x axis at 1.2
     amstrong .
28
29 t=1.5/x
30
31 //this shows that the plane under consideration is
     another plane which is parallel to it(to keep
     miller indices same)
32
```

```
33 n=t*y                                          //Y
      intercept
34
35 p=t*z                                          //Z
      intercept
36
37 printf("           1) Y intercept=")
38
39 disp(n)
40
41 printf("amstrong")
42
43 printf("  2)Z intercept=")
44
45 disp(p)
46
47 printf("amstrong")
```

**Scilab code Exa 1.14.12** calculate Number of atom per unit cell

```
1 //Chapter-1,Example1_14_12,pg 1-63
2
3 ro=7.87                                         //density of
      metal
4
5 A=55.85                                         //atomic wt
      of metal
6
7 N=6.023*10^23                                   //Avogadro's
      number
8
9 a=2.9*10^-8                                     //lattice
      constant of metal
10
11 n=(N*(a^3)*ro)/A
```

```
12
13 printf("Number of atom per unit cell of a metal=")
14
15 disp(int32(n))
```

**Scilab code Exa 1.14.13** calculate Lattice constant

```
1 //Chapter-1,Example1_14_13,pg 1-63
2
3 n=2                                          //BCC
    structure
4
5 ro=9.6*10^2                                 //density of
    sodium crystal
6
7 A=23                                        //atomic
    weight of sodium crystal
8
9 N=6.023*10^26                               //Avogadro's
    number
10
11 a=((n*A)/(N*ro))^(1/3)
12
13 printf("Lattice constant=")
14
15 disp(a)
16
17 printf("m")
```

**Scilab code Exa 1.14.15** calculate Number of atom per unit cell and atomic radius

```
1 //Chapter-1,Example1_14_15,pg 1-64
```

```
2
3 ro=2.7*10^3                                   //density of
      metal
4
5 A=27                                          //atomic wt
     of metal
6
7 N=6.023*10^26                                 //Avogadro's
      number
8
9 a=4.05*10^-10                                 //lattice
     constant of metal
10
11 n=(N*(a^3)*ro)/A
12
13 printf("1) Number of atom per unit cell of a metal="
     )
14
15 disp(int32(n))
16
17 r=sqrt(2)*a/4                                //radius of
     metal
18
19 printf("2) atomic radius of a metal=")
20
21 disp(r)
22
23 printf("m")
```

**Scilab code Exa 1.14.16** calculate Lattice constant and APF

```
1 //Chapter-1,Example1_14_16,pg 1-64
2
3 n=2                                           //BCC
     structure
```

```
4
5 ro=5.98*10^3                                  //density of
      chromium
6
7 A=50                                          //atomic wt
     of chromium
8
9 N=6.023*10^26                                 //Avogadro's
      number
10
11 a=((n*A)/(N*ro))^(1/3)
12
13 printf("      1) Lattice constant=")
14
15 disp(a)
16
17 printf("m")
18
19 //for BCC
20
21 r=sqrt(3)*a/4                                //radius of
     chromium
22
23 APF=(n*(4/3)*%pi*(r^3))/(a^3)
24
25 printf("     2) A.P.F. for chromium=")
26
27 disp(APF)
```

**Scilab code Exa 1.14.17** calculate Lattice constant

```
1 //Chapter-1,Example1_14_17,pg 1-65
2
3 n=4                                           //FCC
     structure
```

```
4
5 ro=6250                                          //density
6
7 M=60.2                                           //molecular
     weight
8
9 N=6.023*10^26                                    //Avogadro's
     number
10
11 a=((n*M)/(N*ro))^(1/3)
12
13 printf("Lattice constant=")
14
15 disp(a)
16
17 printf("m")
```

**Scilab code Exa 1.14.19** calculate wavlength

```
1 //Chapter-1,Example1_14_19,pg 1-66
2
3 a=2.82*10^-9                                     //lattice
     constant
4
5 n=2                                              //FCC
     crystal
6
7 t=17.167                                         //glancing
     angle in degree
8
9 q=%pi/180*t                                      //glancing
     angle in radians
10
11 //assuming reflection in (1,0,0) plane
12
```

```
13 h=1
14
15 k=0
16
17 l=0
18
19 d=a/sqrt(h^2+k^2+l^2)
20
21 //using Bragg's law , 2*d*sin(q)=n*la
22
23 la=2*d*sin(q)/n
24
25 printf("wavlength of X-ray=")
26
27 disp(la)
28
29 printf("m")
```

**Scilab code Exa 1.14.20** calculate Lattice constant and atomic radius

```
1 //Chapter-1,Example1_14_20,pg 1-66
2
3 n=8                                    //Diamond
     structure
4
5 ro=2.33*10^3                          //density of
     diamond
6
7 M=28.9                                //atomic
     weight of diamond
8
9 N=6.023*10^26                         //Avogadro's
     number
10
11 a=((n*M)/(N*ro))^(1/3)
```

```
12
13 printf("      1) Lattice constant=")
14
15 disp(a)
16
17 printf("m")
18
19 r=sqrt(3)*a/8                                  //radius of
     diamond structure
20
21 printf("     2) atomic radius of a metal=")
22
23 disp(r)
24
25 printf("m")
```

**Scilab code Exa 1.14.21** calculate mass of one atom

```
1 //Chapter-1,Example1_14_21,pg 1-66
2
3 n=2                                            //BCC
     structure
4
5 ro=8.57*10^3                                   //density of
      chromium
6
7 d=2.86*10^-10                                  //nearest
     atoms distance
8
9 //d=sqrt(3)/2*a
10
11 a=2*d/sqrt(3)
12
13 //now use formulae a^3*ro=n*A/N
14
```

```
15 //therefore a^3*ro/n=mass of unit cell/(no of atoms
       pre unit cell)=mass of one atom
16
17 m=a^3*ro/n
18
19 printf("mass of one atom=")
20
21 disp(m)
22
23 printf("kg")
```

**Scilab code Exa 1.15.1** calculate glancing angle and highest order

```
1 //Chapter-1,Example1_15_1,pg 1-68
2
3 d=4.255*10^-10                              //
      interplaner spacing
4
5 l=1.549*10^-10                              //wavelength
       of x ray
6
7 //part 1: for smallest glancing angle(n=1)
8
9 n1=1
10
11 //using Bragg's law n*l=2*d*sin(q)
12
13 q=asind(n1*l/(2*d))
14
15 printf("          1)glancing angle=")
16
17 disp(q)
18
19 printf("degree")
20
```

```
21 //part 2: for highst order
22
23 //for highest order sin(q) not exceed one i.e
      maximum value is one
24
25 //using Bragg's law n*l=2*d*sin(q)
26
27 n2=2*d/l                                    //since sin(
      q)is one
28
29 printf("  2)highest order possible =")
30
31 disp(floor(n2))
```

**Scilab code Exa 1.15.2** calculate glancing angle

```
1 //Chapter-1,Example1_15_2,pg 1-69
2
3 a=2.125*10^-10                              //lattice
      constant
4
5 d=a/2                                       //
      interplaner spacing
6
7 n=2                                         //second
      order maximum
8
9 l=0.592*10^-10                              //wavelength
       of rock salt crystal
10
11 //using Bragg's law
12
13 m=asin((n*l)/(2*d))                        //glancing
      angle
14
```

```
15 Q=m*180/%pi
16
17 printf("glancing angle=")
18
19 disp(Q)
20
21 printf("degree")
```

**Scilab code Exa 1.15.3** calculate second order reflection angle

```
1  //Chapter-1,Example1_15_3,pg 1-69
2
3  n1=1                                          //for 1st
       order
4
5  n2=2                                          //for 2nd
       order
6
7  t=3.4                                         //angle
       where 1st order reflection done
8
9  t1=t*%pi/180                                  //convert
       degree to radian
10
11 m=sin(t1)
12
13 //but from Bragg's law
14
15 //n*l=2*d*sin(t)
16
17 //for for constant distance(d) and wavelength(l)
18
19 //order(n) is directly proportionl to sine of angle
       i.e (sin(t))
20
```

```
21 //n1/n2=sin(t1)/sin(t2)
22
23 //assume sin(t2)=a
24
25 a=n2/n1*m
26
27 t2=asind(a)                                    //taking
     sin inverese in degree
28
29 printf("second order reflection take place at an
     angle=")
30
31 disp(t2)
32
33 printf("degree")
```

**Scilab code Exa 1.15.4** calculate shortest wavelength and glancing angle

```
1 //Chapter-1,Example1_15_4,pg 1-70
2
3 V=50*10^3                                     //operating
     voltage of x-ray
4
5 M=74.6                                        //molecular
     weight
6
7 p=1.99*10^3                                   //density
8
9 n=4                                           //no of
     atoms per unit cell(for FCC structure)
10
11 h=6.63*10^-34                                //plank's
     constant
12
13 c=3*10^8                                     //velocity
```

```
14
15 e=1.6*10^-19                                  //charge on
      electron
16
17 N=6.023*10^26                                 //Avogadro's
       number
18
19 //step 1:clculating shortest wavelength
20
21 l=h*c/(e*V)
22
23 printf("        1)shortest wavelength=")
24
25 disp(l)
26
27 printf("m")
28
29 //step:2 calculating distance(d)
30
31 //now a^3*p=n*M/N therefore,
32
33 a=(n*M/(N*p))^(1/3)
34
35 //since KCl is ionic crystal herefore,
36
37 d=a/2
38
39 //step 3: calculaing glancing angle
40
41 //using Bragg's law
42
43 //n*l=2*d*sin(t)
44
45 //assume sin(t)=a, wavelength is minimum i.e l and n
      =1
46
47 n=1
48
```

```
49 a=n*l/(2*d)
50
51 t=asind(a)                                    //taking sin
       inverese in degree
52
53 printf("       2) glancing angle=")
54
55 disp(t)
56
57 printf("degree")
```

**Scilab code Exa 1.15.5** find possible solution of planes

```
1 //Chapter-1,Example1_15_5,pg 1-70
2
3 n=1                                              //first
     order maximum
4
5 l=0.82*10^-10                                    //wavelength
      of X ray
6
7 qd=7                                             //glancing
     angle in degree
8
9 qm=51/60                                         //glancing
     angle in minute
10
11 qs=48/3600                                      //glancing
     angle in second
12
13 q=qd+qm+qs                                      //total
     glancin angle in degree
14
15 //using Bragg's law n*l=2*d*sin(q)
16
```

```
17 d=n*l/(2*sind(q))
18
19 a=3*10^-10                                    //lattice
      constant
20
21 //we know that d=a/root(h^2+k^2+l^2)
22
23 //assume root(h^2+k^2+l^2) =m
24
25 //arranging terms we get
26
27 m=a/d
28
29 printf("square root(h^2+k^2+l^2)=")
30
31 disp(int32(m))
32
33 printf("hence possible solutions are (100),(010)
      ,(001)")
```

**Scilab code Exa 1.15.6** calculate cubic lattice structure

```
1 //Chapter-1,Example1_15_6,pg 1-71
2
3 n=1                                           //first
      order maximum
4
5 l=%i                                         //wavelength
       of X ray
6
7 //part 1:for(100)
8
9 //using Bragg's law n*l=2*d*sin(q)
10
11 q1=5.4                                        //glancing
```

```
    angle in degree
12
13 dl1=n*l/(2*sind(q1))
14
15 //part 2:for(110)
16
17 //using Bragg's law n*l=2*d*sin(q)
18
19 q2=7.6                                    //glancing
    angle in degree
20
21 dl2=n*l/(2*sind(q2))
22
23 //part 3:for(111)
24
25 //using Bragg's law n*l=2*d*sin(q)
26
27 q3=9.4                                    //glancing
    angle in degree
28
29 dl3=n*l/(2*sind(q3))
30
31 //for taking ratio divide all dl by dl1
32
33 d1=dl1/dl1
34
35 d2=dl2/dl1
36
37 d3=dl3/dl1
38
39 printf("cubic lattice structure is=")
40
41 disp(d3,d2,d1)
```

**Scilab code Exa 1.15.7** calculate lattice constant

```
1 //Chapter-1,Example1_15_7,pg 1-71
2
3 n=1                                          //first
     order maximum
4
5 l=1.54*10^-10                               //wavelength
     of rock salt crystal
6
7 q=21.7                                      //glancing
     angle in degree
8
9 //using Bragg's law n*l=2*d*sin(q)
10
11 d=n*l/(2*sind(q))
12
13 printf("lattice constant of crystal=")
14
15 disp(d)
16
17 printf("meter")
```

**Scilab code Exa 1.15.8** calculate glancing angle

```
1 //Chapter-1,Example1_15_8,pg 1-72
2
3 a=2.814*10^-10                              //lattice
     constant
4
5 //the interplanar spacing of plane
6
7 h=1
8
9 k=0
10
11 l=0
```

```
12
13 d=a/sqrt(h^2+k^2+l^2)
14
15 n=2                                          //first
       order maximum
16
17 l=0.714*10^-10                              //wavelength
       of X-ray crystal
18
19 //using Bragg's law
20
21 m=asin((n*l)/(2*d))                          //glancing
       angle
22
23 Q=m*180/%pi
24
25 printf("glancing angle=")
26
27 disp(Q)
28
29 printf("degree")
```

**Scilab code Exa 1.15.9** calculate wavelength and glancing angle and highest order

```
1 //Chapter-1,Example1_15_9,pg 1-72
2
3 d=2.82*10^-10                                //
       interplaner spacing
4
5 t=10                                         //glancing
       angle
6
7 //for part 1
8
```

```
9  n=1                                          //first
       order maximum
10
11 //using Bragg's law n*l=2*d*sin(t)
12
13 l=2*d*sind(t)/n
14
15 printf("          1)wavelength=")
16
17 disp(l)
18
19 printf("meter")
20
21 //for part 2
22
23 n1=2
24
25 //using Bragg's law n*l=2*d*sin(q)
26
27 q=asind(n1*l/(2*d))
28
29 printf("    2)glancing angle=")
30
31 disp(q)
32
33 printf("degree")
34
35 //for part 3
36
37 //for highest order sin(q) not exceed one i.e
       maximum value is one
38
39 //using Bragg's law n*l=2*d*sin(q)
40
41 n2=2*d/l                                      //since sin(
       q)is one
42
43 printf("  3)highest order possible =")
```

```
44
45 disp(floor(n2))
```

**Scilab code Exa 1.15.10** calculate wavelength

```
1  //Chapter-1,Example1_15_10,pg 1-73
2
3  //for line -A
4
5  n1=1                                          //1st order
       maximum
6
7  q1=30                                         //glancing
       angle in degree
8
9  //using Bragg's law for line A n1*l1=2*d1*sin(q1)
10
11 //d1=n1*l1/(2*sin(q1))
12
13 //for line B
14
15 l2=0.97                                       //wavelength
       in amstrong unit
16
17 n2=3                                          //1st order
       maximum
18
19 q2=60                                         //glancing
       angle in degree
20
21 //using Bragg's law for line B n2*l2=2*d2*sin(q2)
22
23 //since for both lines A and B we use same plane of
       same crystal,therefore
24
```

```
25 //d1=d2
26
27 //therefore equution became n2*l2=2*n1*l1/(2*sin(q1)
   )*sin(q2)
28
29 //by arranging terms we get
30
31
32 l1=n2*l2*2*sind(q1)/(2*n1*sind(q2))
33
34 printf("wavelength of the line A=")
35
36 disp(l1)
37
38 printf("amstrong")
```

**Scilab code Exa 1.15.11** calculate glancing angle

```
1 //Chapter-1,Example1_15_11,pg 1-74
2
3 n=1                                    //first
   order minimum
4
5 d=5.5*10^-11                          //atomic
   spacing
6
7 e=1.6*10^-19                          //charge on
   one electron
8
9 Ee=10*10^3                            //energy in
   eV
10
11 E=e*Ee                               //energy in
   Joule
12
```

```
13 m=9.1*10^-31                                    //mass of
      elelctron
14
15 h=6.63*10^-34                                   //plank's
      constant
16
17 l=h/sqrt(2*m*E)                                 //wavelength
18
19 //using Bragg's law
20
21 Q=asind((n*l)/(2*d))                             //glancing
      angle
22
23 printf("glancing angle=")
24
25 disp(Q)
26
27 printf("degree")
```

**Scilab code Exa 1.15.12** calculate glancing angle

```
1 //Chapter-1,Example1_15_12,pg 1-74
2
3 a=2.814*10^-10                                   //lattice
      constant
4
5 //for rock salt
6
7 d=a/2                                            //
      interplaner spacing
8
9 n=1                                              //first
      order maximum
10
11 l=1.541*10^-10                                  //wavelength
```

```
    of rock salt crystal
12
13 //using Bragg's law
14
15 m=asin((n*l)/(2*d))                    //glancing
    angle
16
17 Q=m*180/%pi
18
19 printf("glancing angle=")
20
21 disp(Q)
22
23 printf("degree")
```

**Scilab code Exa 1.16.1** calculate ratio of vacancies

```
1 //Chapter-1,Example1_16_1,pg 1-75
2
3 Ev=1.08                                //average
    energy required to creaet a vacancy
4
5 k=1.38*10^-23                          //boltzman
    constant in J/K
6
7 e=1.6*10^-19                           //charge on
    1 electron
8
9 K=k/e                                  //boltzman
    constant in eV/K
10
11 //for a low concentration of vacancies a relation is
12
13 //n=Nexp(-Ev/KT)
14
```

```
15 //since total no atom is 1 hence N=1
16
17 //at 1000k
18
19 T1=1000                                    //
      temperature
20
21 n1=exp(-Ev/(K*T1))
22
23 //at 500k
24
25 T2=500                                     //
      temperature
26
27 n2=exp(-Ev/(K*T2))
28
29 v=(n1)/(n2)                                 //ratio
      of vacancies
30
31 printf("ratio of vacancies=")
32
33 disp(v)
```

**Scilab code Exa 1.16.2** calculate ratio of vacancies to no of atom

```
1 //Chapter-1,Example1_16_2,pg 1-75
2
3 Ev=1.95                                     //average
      energy required to creaet a vacancy
4
5 k=1.38*10^-23                               //boltzman
      constant in J/K
6
7 e=1.6*10^-19                                //charge on
      1 electron
```

```
8
9  K=k/e                                        //boltzman
       constant in eV/K
10
11 T=500                                        //
       temperature
12
13 //for a low concentration of vacancies a relation is
14
15 //n=Nexp(-Ev/KT)
16
17 m=exp(-Ev/(K*T))                                //ratio
       of no of vacancies to no of atoms n/N
18
19 printf("ratio of no of vacancies to no of atoms=")
20
21 disp(m)
```

**Scilab code Exa 1.16.3** calculate ratio of vacancies

```
1  //Chapter-1,Example1_16_3,pg 1-76
2
3  Ev=1.8                                       //average
       energy required to creaet a vacancy
4
5  k=1.38*10^-23                                //boltzman
       constant in J/K
6
7  e=1.6*10^-19                                 //charge on
       1 electron
8
9  K=k/e                                        //boltzman
       constant in eV/K
10
11 //for a low concentration of vacancies a relation is
```

```
12
13 //n=Nexp(-Ev/KT)
14
15 //ratio of vacancy is n/N assume be r=exp(-Ev/KT)
16
17 //since total no atom is 1 hence N=1
18
19 //at 1000k
20
21 t1=-119                                    //
      temperature in degree
22
23 T1=t1+273                                  //
      temperature in kelvine
24
25 r1=exp(-Ev/(K*T1))
26
27 printf("1)ratio of vacancies at -119 degree=")
28
29 disp(r1)
30
31 //at 500k
32
33 t2=80                                      //
      temperature in degree
34
35 T2=t2+273                                  //
      temperature in kelvine
36
37 r2=exp(-Ev/(K*T2))
38
39 v=(r1)/(r2)                                //ratio
      of vacancies
40
41 printf("2)ratio of vacancies at 80 degree=")
42
43 disp(r2)
```

**Scilab code Exa 1.16.4** calculate no of frankel defects

```
1 //Chapter-1,Example1_16_4,pg 1-76
2
3 Ev=1.5                                    //energy of
     formaton of frankel defect
4
5 k=1.38*10^-23                             //boltzman
     constant in J/K
6
7 e=1.6*10^-19                              //charge on
     1 electron
8
9 K=k/e                                     //boltzman
     constant in eV/K
10
11 T=700                                    //
     temperature
12
13 N=6.023*10^26                            //avogadro's
     no
14
15 //for a low concentration of vacancies a relation is
16
17 //n=Nexp(-Ev/KT)
18
19 m=exp(-Ev/(2*K*T))                        //ratio of
     no of vacancies to no of atoms n/N
20
21 qs=5.56                                   //specific
     density
22
23 q=5.56*10^3                               //real
     density ke/m^3
```

```
24
25 M=0.143                                          //molecular
       weight in kg/m^3
26
27 ma=M/N                                           //mass of
       one molecule
28
29 v=ma/q                                           //vol of
       one molecule
30
31 //v volume containe 1 molecule
32
33 //therefore 1 m^3 containe x molecule
34
35 x=1/v
36
37 d=m*x                                            //defect per
       m^3
38
39 dm=d*10^-9                                       //defect per
       mm^3
40
41 printf("number of frankel defects per mm^3=")
42
43 disp(dm)
```

# Chapter 2

# Semiconductor Physics

**Scilab code Exa 2.21.1** calculate mobility of electron

```
1 //Chapter-2,Example2_21_1,pg 2-47
2
3 ro=1.72*10^-8                          //
    resistivity of Cu
4
5 s=1/ro                                 //
    conductivity of Cu
6
7 n=10.41*10^28                          //no of
    electron per unit volume
8
9 e=1.6*10^-19                           //charge on
    electron
10
11 u=s/(n*e)
12
13 printf("mobility of electron in Cu =")
14
15 disp(u)
16
17 printf("m^2/volt-sec")
```

**Scilab code Exa 2.21.2** calculate Resistivity of Cu

```
1 //Chapter-2,Example2_21_2,pg 2-47
2
3 m=63.5                                    //atomic
      weight
4
5 u=43.3                                    //mobility
      of electron
6
7 e=1.6*10^-19                              //charge on
      electron
8
9 N=6.02*10^23                              //Avogadro's
       number
10
11 d=8.96                                   //density
12
13 Ad=N*d/m                                 //Atomic
      density
14
15 n=1*Ad
16
17 ro=1/(n*e*u)
18
19 printf("Resistivity of Cu =")
20
21 disp(ro)
22
23 printf("ohm-cm")
```

**Scilab code Exa 2.21.3** calculate Resistivity of Ge

```
1 //Chapter-2,Example2_21_3,pg 2-47
2
3 e=1.6*10^-19                                    //charge on
     electron
4
5 ne=2.5*10^19                                   //density of
      carriers
6
7 nh=ne                                          //for
     intrinsic semiconductor
8
9 ue=0.39                                        //mobility
     of electron
10
11 uh=0.19                                       //mobility
     of hole
12
13 s=ne*e*ue+nh*e*uh                             //
     conductivity of Ge
14
15 ro=1/s                                        //
     resistivity of Ge
16
17 printf("Resistivity of Ge =")
18
19 disp(ro)
20
21 printf("ohm-m")
```

**Scilab code Exa 2.21.5** calculate Ratio between conductivity

```
1 //Chapter-2,Example2_21_5,pg 2-48
2
3 Eg=1.2                                         //energy gap
4
```

```
5  T1=600                                          //
       temperature
6
7  T2=300                                          //
       temperature
8
9  //since ue>>uh for intrinsic semiconductor
10
11 //s=ni*e*ue
12
13 K=8.62*10^-5                                    //Boltzman
       constant
14
15 s=%s
16
17 s1=s*exp((-Eg)/(2*K*T1))
18
19 s2=s*exp((-Eg)/(2*K*T2))
20
21 m=(s1/s2)
22
23 printf('Ratio between conductivity =')
24
25 disp(m)
```

**Scilab code Exa 2.21.6** calculate conductivity

```
1  //Chapter-2,Example2_21_6,pg 2-49
2
3  c=5*10^28                                       //
       concentration of Si atoms
4
5  e=1.6*10^-19                                    //charge on
       electron
6
```

```
7  u=0.048                                          //mobility
       of hole
8
9  s=4.4*10^-4                                      //
       conductivity of Si
10
11 //since millionth Si atom is replaced by an indium
       atom
12
13 n=c*10^-6
14
15 sp=u*e*n                                         //
       conductivity of resultant
16
17 printf("conductivity =")
18
19 disp(sp)
20
21 printf("mho/m")
```

**Scilab code Exa 2.21.7** calculate hole concentration and mobility

```
1  //Chapter-2,Example2_21_7,pg 2-49
2
3  m=28.1                                           //atomic
       weight of Si
4
5  e=1.6*10^-19                                     //charge on
       electron
6
7  N=6.02*10^26                                     //Avogadro's
       number
8
9  d=2.4*10^3                                       //density of
       Si
```

```
10
11 p=0.25                                        //
      resistivity
12
13 //no. of Si atom/m^3
14
15 Ad=N*d/m                                       //Atomic
      density
16
17 //impurity level is 0.01 ppm i.e. 1 atom in every
      10^8 atoms of Si
18
19 n=Ad/10^8                                      //no of
      impurity atoms
20
21 //since each impurity produce 1 hole
22
23 nh=n
24
25 printf("1) hole concentration =")
26
27 disp(n)
28
29 printf("holes/m^3")
30
31 up=1/(e*p*nh)
32
33 printf("      2) mobility =")
34
35 disp(up)
36
37 printf("m^2/volt.sec")
```

**Scilab code Exa 2.22.1** calculate probability of an electron

```
1  //Chapter-2,Example2_22_1,pg 2-50
2
3  t=27                                          //temp in
       degree
4
5  T=t+273                                       //temp in
       kelvin
6
7  K=8.62*10^-5                                  //Boltzman
       constant in eV
8
9  Eg=1.12                                       //Energy
       band gap
10
11 //For intrensic semiconductor (Ec-Ev)=Eg/2
12
13 //let (Ec-Ev)=m
14
15 m=Eg/2
16
17 a=(m/(K*T))
18
19 //probability f(Ec)=1/(1+exp((Ec-Ev)/(K*T))
20
21 p=1/(1+exp(a))
22
23
24 printf("probability of an electron being thermally
       excited to conduction band=")
25
26 disp(p)
```

**Scilab code Exa 2.22.2** calculate probability of an electron

```
1  //Chapter-2,Example2_22_2,pg 2-50
```

```
2
3 T=300                                          //temp in
     kelvin
4
5 K=8.62*10^-5                                   //Boltzman
     constant in eV
6
7 m=0.012                                        //energy
     level(Ef-E)
8
9 a=(m/(K*T))
10
11 //probability f(Ec)=1/(1+exp((Ec-Ev)/(K*T)))
12
13 p=1/(1+exp(a))
14
15 p1=1-p
16
17 printf("probability of an energy level not being
     occupied by an electron=")
18
19 disp(p1)
```

**Scilab code Exa 2.22.3** calculate probability of an electron

```
1 //Chapter-2,Example2_22_3,pg 2-51
2
3 t=20                                           //temp in
     degree
4
5 T=t+273                                        //temp in
     kelvin
6
7 K=8.62*10^-5                                   //Boltzman
     constant in eV
```

```
8
9  Eg=1.12                                          //Energy
       band gap
10
11 //For intrensic semiconductor (Ec-Ev)=Eg/2
12
13 //let (Ec-Ev)=m
14
15 m=Eg/2
16
17 a=(m/(K*T))
18
19 //probability f(Ec)=1/(1+exp((Ec-Ev)/(K*T))
20
21 p=1/(1+exp(a))
22
23
24 printf("probability of an electron being thermally
       excited to conduction band=")
25
26 disp(p)
```

**Scilab code Exa 2.22.4** calculate energy for different probability

```
1  //Chapter-2,Example2_22_4,pg 2-51
2
3  T=300                                            //temp in
       kelvin
4
5  K=8.62*10^-5                                     //Boltzman
       constant in eV
6
7  Eg=2.1                                           //Energy
       band gap
8
```

```
9  //probability  f(Ec)=1/(1+exp((Ec-Ev)/(K*T))
10
11 m=K*T
12
13 //for  f(E)=0.99
14
15 p1=0.99
16
17 b=1-1/p1
18
19 a=log(b)                                    //a=(E-2.1)/m
20
21 E=2.1+m*a
22
23 printf("1) Energy for which probability  is 0.99=")
24
25 disp(real(E))
26
27 printf("eV")
28
29 //for  f(E)=0.01
30
31 p2=0.01
32
33 b2=1-1/p2
34
35 a1=log(b2)                                   //a=(E-2.1)/m
36
37 E1=2.1+m*a1
38
39 printf("2)Energy for which probability  is 0.01=")
40
41 disp(real(E1))
42
43 printf("eV")
```

**Scilab code Exa 2.23.1** calculate Potential barrier for Ge

```
1 //Chapter-2,Example2_23_1,pg 2-52
2
3 ni=2.4*10^19                          //density of
      intrensic semiconductor
4
5 n=4.4*10^28                           //no atom in
      Ge crystal
6
7 Nd=n/10^6                             //density
8
9 Na=Nd
10
11 e=1.6*10^-19                         //charge on
     electron
12
13 T=300                                //temerature
      at N.T.P.
14
15 K=1.38*10^-23                        //Boltzman
     constant
16
17 Vo=(K*T/e)*log(Na*Nd/(ni^2))
18
19 printf("Potential barrier for Ge =")
20
21 disp(Vo)
22
23 printf("Volts")
```

**Scilab code Exa 2.23.2** calculate Hall voltage

```
1  //Chapter-2,Example2_23_2,pg 2-52
2
3  B=0.6                                    //magnetic
       field
4
5  d=5*10^-3                                //
       distancebetween surface
6
7  J=500                                    //current
       density
8
9  Nd=10^21                                 //density
10
11 e=1.6*10^-19                             //charge on
       electron
12
13 Vh=(B*J*d)/(Nd*e)                        //due to
       Hall effect
14
15 printf("Hall voltage =")
16
17 disp(Vh)
18
19 printf("Volts")
```

**Scilab code Exa 2.23.3** calculate Hall voltage

```
1  //Chapter-2,Example2_23_3,pg 2-53
2
3  Rh=6*10^-7                               //Hall
       coefficient
4
5  B=1.5                                    //magnetic
       field
6
```

```
I=200                                          //current in
    strip

W=1*10^-3                                      //thickness
   of strip

Vh=Rh*(B*I)/W                                  //due to
   Hall effect

printf("Hall voltage =")

disp(Vh)

printf("Volt")
```

**Scilab code Exa 2.23.4** calculate Resistivity of P type silicon

```
//Chapter-2,Example2_23_4,pg 2-53

Rh=2.25*10^-5                                  //Hall
   coefficient

u=0.025                                        //mobility
   of hole

r=Rh/u

printf("Resistivity of P type silicon =")

disp(r)

printf("ohm-m")
```

**Scilab code Exa 2.23.5** calculate hall voltage hall coefficient and hall angle

```
1 //Chapter-2,Example2_23_5,pg 2-53
2
3 B=0.55                                    //magnetic
    field
4
5 d=4.5*10^-3                               //
    distancebetween surface
6
7 J=500                                     //current
    density
8
9 n=10^20                                   //density
10
11 e=1.6*10^-19                             //charge on
    electron
12
13 Rh=1/(n*e)                               //Hall
    coefficient
14
15 Vh=Rh*B*J*d                              //Hall
    voltage
16
17 printf("          1) Hall voltage =")
18
19 disp(Vh)
20
21 printf("Volts")
22
23 printf("      2) Hall coefficient =")
24
25 disp(Rh)
26
27 printf("m^3/C")
28
29 u=0.17                                   //mobility
```

```
    of electrom
30
31 m=atan(u*B)
32
33 a=m*180/%pi                                    //conversion
       randian into degree
34
35 printf("      3) Hall angle =")
36
37 disp(a)
38
39 printf("degree")
```

**Scilab code Exa 2.23.6** calculate density and mobility

```
1 //Chapter-2,Example2_23_6,pg 2-54
2
3 Rh=3.66*10^-4                                  //Hall
    coefficient
4
5 r=8.93*10^-3                                   //
    resistivity
6
7 e=1.6*10^-19                                   //charge on
    electron
8
9 //Hall coefficient Rh=1/(n*e)
10
11 n=1/(Rh*e)                                    //density
12
13 printf("          1) density(n) =")
14
15 disp(n)
16
17 printf("/m^3")
```

```
18
19 u=Rh/r                                          //mobility
       of electron
20
21 printf("      2) mobility (u) =")
22
23 disp(u)
24
25 printf("m^2/v-s")
```

**Scilab code Exa 2.23.7** calculate Hall voltage

```
1  //Chapter-2,Example2_23_7,pg 2-55
2
3  B=0.2                                          //magnetic
       field
4
5  e=1.6*10^-19                                   //charge on
       electron
6
7  ue=0.39                                        //mobility
       of electron
8
9  l=0.01                                         //length
10
11 A=0.001*0.001                                  //cross
       section area of bar
12
13 V=1*10^-3                                      //Applied
       voltage
14
15 d=0.001                                        //sample of
       width
16
17 r=1/(ue*e)                                     //
```

```
    resistivity
18
19 R=r*l/A                                    //resistance
       of Ge bar
20
21 //using ohm's law
22
23 I=V/R
24
25 Rh=r*ue                                    //hall
    coefficient
26
27 //using formulae for hall effect
28
29 J=I/A                                      //current
    density
30
31 Vh=Rh*B*J*d
32
33 printf("Hall voltage =")
34
35 disp(Vh)
```

**Scilab code Exa 2.24.1** calculate fermi level

```
1 //Chapter-2,Example2_24_1,pg 2-55
2
3 x1=0.4                                     //difference
       between fermi level and conduction band(Ec-Ef)
4
5 T=300                                      //temp in
    kelvin
6
7 K=8.62*10^-5                               //Boltzman
    constant in eV
```

```
8
9  //ne=N*e^(-(Ec-Ef)/(K*T))
10
11 //ne is no of electron in conduction band
12
13 //since concentration of donor electron is doubled
14
15 a=2                                          //ratio of
      no of electron
16
17 //let x2 be the difference between new fermi level
      and conduction band(Ec-Ef')
18
19 x2=-log(a)*(K*T)+x1                          //arranging
      equation ne=N*e^(-(Ec-Ef)/(K*T))
20
21 printf("Fermi level will be shifted towards
      conduction band by")
22
23 disp(x2)
24
25 printf("eV")
```

# Chapter 3

# Dielectric And Magnetic Materials

**Scilab code Exa 3.17.1** calculate resultant voltage

```
1 //Chapter-3,Example3_17_1,pg 3-35
2
3 A=650*10^-6                                   //area
4
5 d=4*10^-3                                     //seperation
      of plate
6
7 Q=2*10^-10                                    //charge
8
9 er=3.5                                        //relative
     permitivity
10
11 e0=8.85*10^-12                               //absolute
     permitivity
12
13 V=(Q*d)/(e0*er*A)
14
15 printf("voltage across capacitor =")
16
```

```
17 disp(V)
18
19 printf("Volt")
```

**Scilab code Exa 3.17.2** find capacitance of capacitor

```
1  //Chapter-3,Example3_17_2,pg 3-36
2
3  A=2000*10^-6                                    //area
4
5  d=0.5*10^-6                                     //seperation
       of plate
6
7  er=8                                            //relative
     permitivity
8
9  e0=8.85*10^-12                                  //absolute
     permitivity
10
11 C=(e0*er*A)/d
12
13 printf("capacitance for capacitor =")
14
15 disp(C)
16
17 printf("Faraday")
```

**Scilab code Exa 3.17.3** calculate relative permittivity

```
1  //Chapter-3,Example3_17_3,pg 3-36
2
3  E=1000                                          //electric
     field
```

```
4
5 P=4.3*10^-8                                    //
    polarization
6
7 e0=8.854*10^-12                                //absolute
    permitivity
8
9 er=(P/(e0*E))+1                                //as P/E=e0(
    er-1)
10
11 printf("relative permittivity =")
12
13 disp(er)
```

**Scilab code Exa 3.17.4** ratio of two capacitor

```
1 //Chapter-3,Example3_17_4,pg 3-36
2
3 //As C=e0*er*A/d
4
5 e0=%e                                          //absolute
    permitivity
6
7 Ag=%s
8
9 Ap=Ag                                          //Assuming
    Area of glass plate and plastic film is same
10
11 //for glass
12
13 erg=6                                         //relative
    permitivity
14
15 dg=0.25                                       //thickness
16
```

```
17 Cg=e0*erg*Ag/dg
18
19 //for plastic film
20
21 erp=3                                          //relative
      permitivity
22
23 dp=0.1                                         //thickness
24
25 Cp=e0*erp*Ap/dp
26
27 m=Cg/Cp
28
29 printf("since Cg/Cp=")
30
31 disp(m)
32
33 printf("plastic film holds more charge")
```

**Scilab code Exa 3.17.5** calculate electronic polarizability and radius of He atom

```
1 //Chapter-3,Example3_17_5,pg 3-37
2
3 N=2.7*10^25                                    //no of
      atoms per m^3
4
5 er=1.0000684                                   //dielectric
      constant of He atom at NTP
6
7 e0=8.854*10^-12                                //absolute
      permitivity
8
9 a=e0*(er-1)/N                                  //electronic
      polarizability
```

```
10
11 printf("1) electronic polarizability=")
12
13 disp(a)
14
15 R=(a/(4*%pi*e0))^(1/3)                    //radius of
      helium atom
16
17 printf("2) radius of He atoms =")
18
19 disp(R)
20
21 printf("meter")
```

**Scilab code Exa 3.17.6** calculate electric susceptibility

```
1 //Chapter-3,Example3_17_6,pg 3-37
2
3 er=1.000014                                //dielectric
     constant of He atom at NTP
4
5 Xe=er-1                                    //electric
     susceptibility
6
7 printf("electric susceptibility =")
8
9 disp(Xe)
```

**Scilab code Exa 3.17.7** calculate relative permeability

```
1 //Chapter-3,Example3_17_7,pg 3-37
2
```

```
3 T=300                                          //
    temperature of paramagnetic material
4
5 X=3.7*10^-3                                    //
    susceptibility of material
6
7 C=X*T                                          //using
    Curie's law
8
9 T1=250                                         //
    temperature
10
11 T2=600                                        //
    temperature
12
13 u1=C/T1                                       //relative
    permeability of material at 250k
14
15 u2=C/T2                                       //relative
    permeability of material at 350k
16
17 printf("relative permeability at temp 250K=")
18
19 disp(u1)
20
21 printf("relative permeability at temp 600K =")
22
23 disp(u2)
```

**Scilab code Exa 3.17.8** calculate Temperature

```
1 //Chapter-3,Example3_17_8,pg 3-38
2
3 u=0.8*10^-23                                   //magnetic
    dipole moment of an atom
```

```
4
5 B=0.8                                          //magnetic
    field
6
7 K=1.38*10^-23                                  //boltzmann
    constant
8
9 T=(2*u*B)/(3*K)                                //
    temperature
10
11 printf("Temperature at which average thermal energy
    of an atom is equal to magntic energy=")
12
13 disp(T)
14
15 printf("K")
```

**Scilab code Exa 3.17.9** calculate magnetization of paramagnetic material

```
1 //Chapter-3,Example3_17_9,pg 3-38
2
3 B=0.5                                          //magnetic
    field
4
5 t=27                                           //
    temperature in degree celcius
6
7 T=273+t                                        //
    temperature in kelvin
8
9 u0=4*%pi*10^-7                                 //
    permeability of free space
10
11 C=2*10^-3                                     //Curie's
    constant
```

```
12
13 M=(C*B)/(u0*T)                                    //
       magnetization of material
14
15 printf("magnetization of paramagnetic material =")
16
17 disp(M)
18
19 printf("A/m")
```

**Scilab code Exa 3.17.10** calculate Horizontal component of magnetic field

```
1 //Chapter-3,Example3_17_10,pg 3-38
2
3 u0=4*%pi*10^-7                                    //
       permeability of free space
4
5 B=10.9*10^-5                                      //flux
       density
6
7 H=B/u0                                            //magnetic
       field
8
9 printf("Horizontal component of magnetic field =")
10
11 disp(H)
12
13 printf("A-m")
```

**Scilab code Exa 3.17.11** calculate current required

```
1 //Chapter-3,Example3_17_11,pg 3-39
2
```

```
3  phi=5.9*10^-3                                    //magnetic
       flux
4
5  ur=900                                           //relative
       permeability of material
6
7  n=700                                            //number of
       turns
8
9  u0=4*%pi*10^-7                                   //
       permeability of free space
10
11 A=60*10^-4                                       //cross
       section area of ring
12
13 l=2                                              //mean
       circumference of ring
14
15 B=phi/A                                          //flux
       density
16
17 H=B/(u0*ur)                                      //magnetic
       field
18
19 At=H*l                                           //Amp-turns
       required
20
21 I=At/n                                           //current
       required
22
23 printf("Current required to produce a flux=")
24
25 disp(I)
26
27 printf("Amp")
```

**Scilab code Exa 3.17.12** calculate Current required

```
1 //Chapter-3,Example3_17_12,pg 3-39
2
3 phi=2.7*10^-3                          //magnetic
    flux
4
5 A=25*10^-4                             //cross
    section area of ring
6
7 r=25*10^-2                             //mean
    circumference of ring
8
9 la=10^-3                               //air gap
10
11 ur=900                                //relative
    permeability of material
12
13 n=400                                 //number of
    turns
14
15 u0=4*%pi*10^-7                        //
    permeability of free space
16
17 d=40*10^-2                            //mean
    diameter of ring
18
19 li=2*%pi*r                            //mean
    circumference of ring
20
21 B=phi/A                               //flux
    density
22
23 //for air gap
```

```
24
25 Ha=B/(u0)                                    //magnetic
      field for air gap
26
27 //for iron ring
28
29 Hi=B/(u0*ur)                                  //magnetic
      field for iron ring
30
31 //therefore, Amp turn in air gap
32
33 Ata=Ha*la                                    //Amp-turns
      required
34
35 //therefore, Amp-turn in ring
36
37 Ati=Hi*li                                     //Amp-turns
      required
38
39 //therrfore total mmf required
40
41 mmf=Ata+Ati
42
43 //Current required
44
45 I=mmf/n                                       //current
      required
46
47 printf("Current required =")
48
49 disp(I)
50
51 printf("Amp")
```

**Scilab code Exa 3.17.13** calculate 1 magnetic intensity 2 magnetization 3 Relative Permeability

```
1 //Chapter-3,Example3_17_13,pg 3-40
2
3 n1=10                                          //no of
    turns per cm
4
5 i=2                                            //current
6
7 B=1                                            //flux
    density
8
9 u0=4*%pi*10^-7                                 //
    permeability of free space
10
11 n=n1*100                                      //no turns
    per m
12
13 H=n*i
14
15 printf("                    1) magnetic intensity =")
16
17 disp(H)
18
19 printf("Amp-turn/meter")
20
21 //calculation for magnetization
22
23 I=B/u0-H
24
25 printf("      2) magnetization =")
26
27 disp(I)
28
29 printf("Amp-turn/meter")
30
31 //relative permeability
```

```
32
33 ur=B/(u0*H)
34
35 printf("      3) Relative Permeability of the ring =")
36
37 disp(int(ur))
```

**Scilab code Exa 3.17.14** calculate Loss of energy

```
1  //Chapter-3,Example3_17_14,pg 3-40
2
3  m=40                                    //wt of the
       core
4
5  d=7.5*10^3                              //density of
       iron
6
7  n=100                                   //frequency
8
9  V=m/d                                   //volume of
       the iron core
10
11 E1=3800*10^-1                           //loss of
       energy in core per cycles/cc
12
13 E2=E1*V                                 //loss of
       energy in core per cycles
14
15 N=60*n                                  //no of
       cycles per minute
16
17 E=E2*N                                  //loss of
       energy per minute
18
```

```
19 printf("Loss of energy per minute =")
20
21 disp(E)
22
23 printf("Joule")
```

---

**Scilab code Exa 3.17.15** calculate various parameter of magnetic field

```
1 //Chapter-3,Example3_17_15,pg 3-40
2
3 l=30*10^-2                                    //length of
     ring
4
5 A=1*10^-4                                     //cross
     section area of ring
6
7 i=0.032                                       //current
8
9 phi=2*10^-6                                   //magnetic
     flux
10
11 u0=4*%pi*10^-7                               //
     permeability of free space
12
13 N=300                                        //no of
     turns in the coil
14
15 //1) flux density
16
17 B=phi/A                                      //flux
     density
18
19 printf("1) Flux density in the ring =")
20
21 disp(B)
```

```
22
23 printf("Wb/m^2")
24
25 //2) magnetic intensity of ring
26
27 n=N/l                                          //no of
      turns per unit length
28
29 H=n*i                                          //magnetic
      intensity
30
31 printf("                2) magnetic intensity =")
32
33 disp(H)
34
35 printf("Amp-turn/meter")
36
37 //3) permeability and relative permeability of the
      ring
38
39 u=B/H
40
41 printf("       3) Permeability of the ring =")
42
43 disp(u)
44
45 printf("Wb/A-m")
46
47 ur=u/u0
48
49 printf("                4) Relative Permeability of
      the ring =")
50
51 disp(ur)
52
53 //4)Susceptibility
54
55 Xm=ur-1
```

```
56
57 printf("5) magnetic Susceptibility of the ring =")
58
59 disp(Xm)
```

**Scilab code Exa 3.17.16** calculate loss of energy per hour

```
1  //Chapter-3,Example3_17_16,pg 3-41
2
3  E=3000                                    //loss of
       energy per cycle per cm^3
4
5  m=12*10^3                                 //wt of the
       core
6
7  d=7.5                                     //density of
       iron
8
9  n=50                                      //frequency
10
11 V=m/d                                     //volume of
       the core
12
13 El=E*V*n*60*60                            //loss of
       energy per hour
14
15 printf("Loss of energy per hour =")
16
17 disp(El)
18
19 printf("Erg")
```

**Scilab code Exa 3.17.17** calculate Hysteresis power loss

```
1 //Chapter-3,Example3_17_17,pg 3-41
2
3 n=50                                          //frequency
4
5 V=10^-3                                      //volume of
     the specimen
6
7 //Area of B-H loop
8
9 A=0.5*10^3*1
10
11 P=n*V*A
12
13 printf("Hysteresis power loss =")
14
15 disp(P)
16
17 printf("Watt")
```

**Scilab code Exa 3.17.18** calculate current required

```
1 //Chapter-3,Example3_17_18,pg 3-42
2
3 phi=1.5*10^-4                                //magnetic
     flux
4
5 ur=900                                       //relative
     permeability of material
6
7 n=600                                        //number of
     turns
8
9 u0=4*%pi*10^-7                               //
     permeability of free space
10
```

```
11 A=5.8*10^-4                                  //cross
      section area of ring
12
13 d=40*10^-2                                   //mean
      diameter of ring
14
15 li=%pi*d                                     //mean
      circumference of ring
16
17 la=5*10^-3                                   //air gap
18
19 B=phi/A                                      //flux
      density
20
21 //for air gap
22
23 Ha=B/(u0)                                    //magnetic
      field for air gap
24
25 //for iron ring
26
27 Hi=B/(u0*ur)                                 //magnetic
      field for iron ring
28
29 //therefore, Amp turn in air gap
30
31 Ata=Ha*la                                    //Amp-turns
      required
32
33 //therefore, Amp-turn in ring
34
35 Ati=Hi*li                                    //Amp-turns
      required
36
37 //therrfore total mmf required
38
39 mmf=Ata+Ati
40
```

```
41 //Current required
42
43 I=mmf/n                                          //current
      required
44
45 printf("Current required =")
46
47 disp(I)
48
49 printf("Amp")
```

**Scilab code Exa 3.17.19** calculate reluctance and mmf

```
1 //Chapter-3,Example3_17_19,pg 3-42
2
3 la=1*10^-2                                       //air gap
4
5 r=0.5                                            //radius of
      ring
6
7 A=5*10^-4                                        //cross
      section area of ring
8
9 i=5                                              //current
10
11 u=6*10^-3                                       //
      permeability of iron
12
13 u0=4*%pi*10^-7                                  //
      permeability of free space
14
15 N=900                                           //no of
      turns in the coil
16
17 //let reluctance of iron ring with air gap be S
```

```
18
19 S=la/(u0*A)+(2*%pi*r-la)/(u*A)
20
21 printf("          1) Reluctance =")
22
23 disp(S)
24
25 printf("A-T/Wb")
26
27 mmf=N*i
28
29 printf("   2) m.m.f =")
30
31 disp(mmf)
32
33 printf("Amp-turn")
```

**Scilab code Exa 3.17.20** calculate current

```
1 //Chapter-3,Example3_17_20,pg 3-43
2
3 //the magnetization force is given by,
4
5 //H=NI/l
6
7 H=5*10^3                                    //coercivity
      of bar magnet
8
9 l=10*10^-2                                 //length of
     solenoid
10
11 N=50                                      //number of
     turns
12
13 I=l*H/N
```

```
14
15 printf("current =")
16
17 disp(I)
18
19 printf("Ampere")
```

**Scilab code Exa 3.17.21** calculate Reluctance and current

```
1 //Chapter-3,Example3_17_21,pg 3-43
2
3 ur=380                                    //relative
     permeability of air
4
5 u0=4*%pi*10^-7                           //
     permeability of free space
6
7 A=5*10^-4                                //cross
     section area of ring
8
9 n=200                                    //number of
     turns
10
11 d=20*10^-2                              //mean
     diameter of ring
12
13 l=%pi*d                                 //mean
     circumference of ring
14
15 phi=2*10^-3                             //magnetic
     flux
16
17 S=l/(u0*ur*A)                           //reluctance
18
19 //using ohm's law for magnetic circuit
```

```
20
21 //phi=N*I/S
22
23 I=S*phi/n
24
25 printf("          1) Reluctance =")
26
27 disp(S)
28
29 printf("A-T/Wb")
30
31
32 printf("    2) current =")
33
34 disp(I)
35
36 printf("Ampere")
```

**Scilab code Exa 3.17.22** calculate various parameter of magnetic field

```
1 //Chapter-3,Example3_17_22,pg 3-43
2
3 ur=1                                    //relative
     permeability of air
4
5 u0=4*%pi*10^-7                         //
     permeability of free space
6
7 A=6*10^-4                              //cross
     section area of torroid
8
9 n=500                                  //number of
     turns
10
11 r=15*10^-2                            //radius of
```

```
    torroid
12
13 I=4                                          //current in
       coil
14
15 l=2*%pi*r                                    //mean
    circumference of torroid
16
17 MMF=n*I
18
19 printf("1) MMF (NI) =")
20
21 disp(MMF)
22
23 printf("AT")
24
25 R=l/(u0*ur*A)                                //Reluctance
26
27 printf("        2) Reluctance (R) =")
28
29 disp(R)
30
31 printf("AT/Wb")
32
33 phi=MMF/R                                    //flux
34
35 printf("     3) Magnetic flux =")
36
37 disp(phi)
38
39 printf("Wb")
40
41 B=phi/A                                       //flux
    density
42
43 printf("        4) Flux density =")
44
45 disp(B)
```

```
46
47 printf("Wb/m^2")
48
49 H=B/(u0*ur)                                    //magnetic
      field intensity
50
51 printf("    5) Magnetic field intensity =")
52
53 disp(H)
54
55 printf("A/m")
```

**Scilab code Exa 3.17.23** calculate Number of AmpereTurns

```
1 //Chapter-3,Example3_17_23,pg 3-44
2
3 phi=10^-3                                     //magnetic
      flux
4
5 ur=1000                                       //relative
      permeability of iron
6
7 u0=4*%pi*10^-7                                //
      permeability of free space
8
9 A=5*10^-4                                     //cross
      section area of ring
10
11 la=2*10^-3                                   //air gap
12
13 d=20*10^-3                                   //mean
      diameter of ring
14
15 li=%pi*d-la                                  //mean
      circumference of ring
```

```
16
17 //using KVL for magnetic circuit
18
19 //AT(total)=AT(iron)+AT(air gap)
20
21 ATt=(phi/(u0*A))*((li/ur)+la)
22
23 printf("Number of Ampere-Turns required =")
24
25 disp(round(ATt))
```

**Scilab code Exa 3.17.24** calculate intensity magnetization and flux density

```
1 //Chapter-3,Example3_17_24,pg 3-44
2
3 X=0.5*10^-5                                    //
    susceptibility of material
4
5 H=10^6                                         //magnetic
    field strength
6
7 I=X*H                                          //intensity
    of magnetization
8
9 u0=4*%pi*10^-7                                 //
    permeability of free space
10
11 B=u0*(H+I)                                    //flux
    density
12
13 printf("            1) intensity magnetization =")
14
15 disp(I)
16
```

```
17 printf("Amp/m")
18
19 printf("        2) flux density in the material =")
20
21 disp(B)
22
23 printf("wb/m^2")
```

# Chapter 4

# Acoustics and Ultrasonics

**Scilab code Exa 4.11.1** calculate length

```
1 //Chapter-4,Example4_11_1,pg 4-17
2
3 d=8900                                    //density
4
5 Y=20.8*10^10                              //Young's
    modulus
6
7 n=40*10^3                                 //frequency
    of wave
8
9 k=1                                       //consider 1
    st harmonic
10
11 l=(k/(2*n))*sqrt(Y/d)                    //arranging
    formula of natural frequency
12
13 printf("length =")
14
15 disp(l)
16
17 printf("meter")
```

**Scilab code Exa 4.12.1** calculate thickness

```
1 //Chapter-4,Example4_12_1,pg 4-20
2
3 d=2.65*10^3                                    //density
4
5 Y=8*10^10                                      //Young's
      modulus
6
7 n=1*10^6                                       //frequency
      of wave
8
9 k=1                                            //consider 1
      st harmonic
10
11 t=(k/(2*n))*sqrt(Y/d)                         //arranging
      formula of natural frequency
12
13 printf("thickness =")
14
15 disp(t)
16
17 printf("meter")
```

**Scilab code Exa 4.15.1** calculate Reverberation time

```
1 //Chapter-4,Example4_15_1,pg 4-25
2
3 l=20                                           //length of
      room
4
```

```
5  b=15                                          //bredth of
       room
6
7  h=10                                          //height of
       room
8
9  V=l*b*h                                       //volume of
       room
10
11 a=0.106                                       //absorption
        coefficient
12
13 S=2*(l*b+b*h+h*l)                             //surface
       area of hall
14
15 T=(0.161*V)/(a*S)                             //
       Reverberation time,using Sabine's formula
16
17 printf("Reverberation time =")
18
19 disp(T)
20
21 printf("sec")
```

**Scilab code Exa 4.15.2** calculate change in intensity level

```
1  //Chapter-4,Example4_15_2,pg 4-26
2
3  m=%i                                          //original
       sound intensity
4
5  n=1000*%i                                     //increased
       intensity value
6
7  l=10*log10(n/m)                               //change in
```

```
    intensity level
8
9  printf("change in intensity level =")
10
11 disp(l)
12
13 printf("dB")
```

**Scilab code Exa 4.15.3** clculate average sound absorption coefficient and reverberation time

```
1  //Chapter-4,Example4_15_3,pg 4-26
2
3  S1=220                                    //wall area
4
5  a1=0.03                                   //absorption
       coefficient for the wall
6
7  S2=120                                    //floor area
8
9  a2=0.8                                    //absorption
       coefficient for the floor
10
11 S3=120                                    //ceiling
     area
12
13 a3=0.06                                   //absorption
       coefficient for the ceiling
14
15 V=600                                     //volume of
     room
16
17 S=S1+S2+S3                                //total
     surface area
18
```

```
19 a=(a1*S1+a2*S2+a3*S3)/S                    //average
     sound absorption coefficient
20
21 printf("1) average sound absorption coefficient =")
22
23 disp(a)
24
25 T=(0.161*V)/(a*S)                           //
     Reverberation time,using Sabine's formula
26
27 printf("2) Reverberation time =")
28
29 disp(T)
30
31 printf("sec")
```

**Scilab code Exa 4.15.4** calculate average absorption coefficient

```
1 //Chapter-4,Example4_15_4,pg 4-27
2
3 V=5500                                      //volume
4
5 T=2.3                                       //
     Reverberation time
6
7 S=750                                       //sound
     absorption coefficient
8
9 a=(0.161*V)/(S*T)                           //using
     Sabine's formula
10
11 printf("average absorption coefficient =")
12
13 disp(a)
```

**Scilab code Exa 4.15.5** claculate average absorption coefficient and area of floor

```
1 //Chapter-4,Example4_15_5,pg 4-27
2
3 l=20                                          //length of
     room
4
5 b=12                                          //bredth of
     room
6
7 h=12                                          //height of
     room
8
9 V=l*b*h                                       //volume of
     room
10
11 S=2*(l*b+b*h+h*l)                            //surface
     area of hall
12
13 T1=2.5                                       //
     Reverberation time
14
15 a=(0.161*V)/(T1*S)                           //using
     Sabine's formula
16
17 printf("1) average absorption coefficient =")
18
19 disp(a)
20
21 a1=0.5                                       //absorption
      coefficient
22
23 T2=2                                         //
```

```
    Reverberation time
24
25 S1=(0.161*V/(a1-a))*(1/T2-1/T1)
26
27 printf("2) carpet area required =")
28
29 disp(S1)
30
31 printf("m^2")
```

**Scilab code Exa 4.15.6** calculate reverberation time for various case

```
1 //Chapter-4,Example4_15_6,pg 4-28
2
3 Ac=10*12                                    //area of
    carpet covering entire floor
4
5 ac=0.06                                     //absorption
     coefficient of carpet
6
7 aS1=Ac*ac                                   //absorption
     due to carpet
8
9 Af=10*12                                    //area of
    false celling
10
11 af=0.03                                    //absorption
     coefficient of celling
12
13 aS2=Af*af                                  //absorption
     due to celling
14
15 As=100*1                                   //area of
    cushioned sets
16
```

```
17 as=1                                          //absorption
       coefficient of cushion sets
18
19 aS3=As*as                                     //absorption
       due to cusion sets
20
21 Aw=346*1                                      //area of
      walls covered with absorbent
22
23 aw=0.2                                        //absorption
       coefficient of walls
24
25 aS4=Aw*aw                                     //absorption
       due to walls
26
27 Ad=346*1                                      //area of
      wooden door
28
29 ad=0.2                                        //absorption
       coefficient of wooden door
30
31 aS5=Ad*ad                                     //absorption
       due to wooden door
32
33 aS=aS1+aS2+aS3+aS4                            //total
      absorption
34
35 ap=0.46                                       //absorption
       coefficient of audience/person
36
37 l=12                                          //assuming
      length of wall
38
39 b=10                                          //assuming
      breadth of wall
40
41 h=8                                           //assuming
      height of wall
```

```
42
43 V=l*b*h                                    //volume of
      hall
44
45 //case 1 :(no one inside/emptey  hall)
46
47 T1=(0.161*V)/aS                            //
      reverberation time
48
49 printf("      1)reverberation time of empty hall =")
50
51 disp(T1)
52
53 printf("sec")
54
55 //case 2 :(50 person inside hall)
56
57 T2=(0.161*V)/(aS+50*0.46)                   //
      reverberation time
58
59 printf("  2)reverberation time of hall with 50
      person =")
60
61 disp(T2)
62
63 printf("sec")
64
65 //case 2 :(100 person inside hall/full capacity of
      hall)
66
67 T3=(0.161*V)/(aS+100*0.46)                  //
      reverberation time
68
69 printf("  3)reverberation time of hall with 100
      person =")
70
71 disp(T3)
72
```

```
73 printf("sec")
```

---

**Scilab code Exa 4.15.7** calculate average absorption coefficient and total absorption

```
1  //Chapter-4,Example4_15_7,pg 4-30
2
3  l=20                                          //length of
       room
4
5  b=15                                          //bredth of
       room
6
7  h=5                                           //height of
       room
8
9  V=l*b*h                                       //volume of
       room
10
11 S=2*(l*b+b*h+h*l)                             //surface
       area of hall
12
13 T=3.5                                         //
       Reverberation time
14
15 a=(0.161*V)/(T*S)                             //using
       Sabine's formula
16
17 printf("1) average absorption coefficient =")
18
19 disp(a)
20
21 avg=a*S                                       //average
       total absorption
22
```

```
23 printf("2) average total absorption =")
24
25 disp(avg)
26
27 printf("m^2.S")
```

---

**Scilab code Exa 4.15.8** calculate change in reverberation time

```
1 //Chapter-4,Example4_15_8,pg 4-30
2
3 l=20                                    //length of
     room
4
5 b=15                                    //bredth of
     room
6
7 h=10                                    //height of
     room
8
9 V=l*b*h                                 //volume of
     room
10
11 a=0.1                                  //absorption
     coefficient
12
13 S=2*(l*b+b*h+h*l)                      //surface
     area of hall
14
15 T1=(0.161*V)/(a*S)                     //
     Reverberation time,using Sabine's formula
16
17 printf("1) Reverberation time =")
18
19 disp(T1)
20
```

```
21 printf("sec")
22
23 a2=0.66                                    //absorption
       coefficient of curtain cloth
24
25 S2=100                                     //surface
       area of a curtain cloth
26
27 T2=(0.161*V)/(a*S+a2*S2*2)                 //
       Reverberation time,using Sabine's formula
28
29 T=T1-T2                                    //change in
       Reverberation time
30
31 printf("2) change in Reverberation time =")
32
33 disp(T)
34
35 printf("sec")
```

**Scilab code Exa 4.15.9** calculate average absorption coefficient and reverberation time

```
1 //Chapter-4,Example4_15_9,pg 4-30
2
3 S1=220                                     //wall area
4
5 a1=0.03                                    //absorption
       coefficient for the wall
6
7 S2=120                                     //floor area
8
9 a2=0.8                                     //absorption
       coefficient for the floor
10
```

```
11 S3=120                                          //ceiling
      area
12
13 a3=0.06                                         //absorption
       coefficient for the ceiling
14
15 V=600                                           //volume of
      room
16
17 S=S1+S2+S3                                      //total
      surface area
18
19 a=(a1*S1+a2*S2+a3*S3)/S                         //average
      sound absorption coefficient
20
21 printf("1) average sound absorption coefficient =")
22
23 disp(a)
24
25 T=(0.161*V)/(a*S)                               //
      Reverberation time,using Sabine's formula
26
27 printf("2) Reverberation time =")
28
29 disp(T)
30
31 printf("sec")
```

**Scilab code Exa 4.15.10** calculate depth of seabed and wavelength

```
1 //Chapter-4,Example4_15_10,pg 4-31
2
3 f=0.07*10^6                                      //frequency
4
5 t=0.65                                           //time
```

```
6
7 v=1700                                          //velocity
     of sound
8
9 d=v*t/2                                         //depth of
     seabed
10
11 printf("1) depth of seabed =")
12
13 disp(d)
14
15 printf("meter")
16
17 l=v/f                                          //wavelength
18
19 printf("2) wavelength =")
20
21 disp(l)
22
23 printf("meter")
```

**Scilab code Exa 4.15.11** calculate natural frequency

```
1 //Chapter-4,Example4_15_11,pg 4-31
2
3 t=1*10^-3                                       //thicknesss
     of crystal
4
5 d=2.65*10^3                                     //density
6
7 Y=8*10^10                                       //Young's
     modulus
8
9 k=1                                             //consider 1
     st harmonic
```

```
10
11 n=(k/(2*t))*sqrt(Y/d)                    //formula of
       natural frequency
12
13 printf(" natural frequency =")
14
15 disp(n)
16
17 printf("Hz")
```

**Scilab code Exa 4.15.12** calculate thickness

```
1  //Chapter-4,Example4_15_12,pg 4-32
2
3  d=2650                                   //density
4
5  Y=8*10^10                                //Young's
       modulus
6
7  k=1                                      //consider 1
       st harmonic
8
9  //case 1
10
11 n1=3.8*10^6                              //frequency
       of wave
12
13 t1=(k/(2*n1))*sqrt(Y/d)                  //arranging
       formula of natural frequency
14
15 printf("1) thickness =")
16
17 disp(t1)
18
19 printf("meter")
```

```
20
21 //case 2
22
23 n2=300*10^3                                    //frequency
      of wave
24
25 t2=(k/(2*n2))*sqrt(Y/d)                       //arranging
      formula of natural frequency
26
27 printf("2) thickness =")
28
29 disp(t2)
30
31 printf("meter")
```

**Scilab code Exa 4.15.13** calculate thickness

```
1 //Chapter-4,Example4_15_13,pg 4-32
2
3 d=2650                                         //density
4
5 Y=8*10^10                                     //Young's
      modulus
6
7 n=2*10^6                                      //frequency
      of wave
8
9 k=1                                           //consider 1
      st harmonic
10
11 t=(k/(2*n))*sqrt(Y/d)                        //arranging
      formula of natural frequency
12
13 printf("thickness =")
14
```

```
15 disp(t)
16
17 printf("meter")
```

---

**Scilab code Exa 4.15.14** calculate distance between two ships

```
1  //Chapter-4,Example4_15_14,pg 4-33
2
3  f=50*10^3                                    //frequency
4
5  v1=348                                       //velocity
       of ultrasound in atmosphere
6
7  v2=1392                                      //velocity
       of ultrasound in sea water
8
9  t=2                                          //time
       difference
10
11 //distance is constant hence v1*t1=v2*t2
12
13 m=v2/v1                                       //assuming
       constant as m
14
15 //(t1-t2=d) and (t1=m*t2) therefore
16
17 t2=t/(m-1)
18
19 d=v2*t2                                       //distance
       between two ship
20
21 printf("distance between two ships =")
22
23 disp(d)
24
```

```
25 printf("meter")
```

**Scilab code Exa 4.15.15** calculate natural frequency and change in thickness

```
1 //Chapter-4,Example4_15_15,pg 4-34
2
3 //for case1
4 t1=2*10^-3                                    //thicknesss
      of plate
5
6 d=2.65*10^3                                   //density
7
8 Y=8*10^10                                     //Young's
     modulus
9
10 k=1                                          //consider 1
     st harmonic
11
12 n1=(k/(2*t1))*sqrt(Y/d)                      //formula of
      natural frequency
13
14 printf("        1)natural frequency =")
15
16 disp(n1)
17
18 printf("Hz")
19
20 //for case2
21
22 n2=3*10^6                                    //frequency
23
24 t2=(k/(2*n2))*sqrt(Y/d)                      //arranging
     formula of natural frequency
25
```

```
26 t=t1-t2                                          //change in
       thickness
27
28 printf("      2)change in thickness =")
29
30 disp(t)
31
32 printf("meter")
```

**Scilab code Exa 4.15.16** calculate depth of sea bed

```
1  //Chapter-4,Example4_15_16,pg 4-34
2
3  S=10                                             //salinity
4
5  t=2                                              //time
6
7  T=20                                             //
       temperature
8
9  v=1510+1.14*S+4.21*T-0.037*T^2                   //velocity
       of ultrasound in sea
10
11 d=v*t/2                                          //depth of
       sea bed
12
13 printf("depth of sea bed =")
14
15 disp(d)
16
17 printf("meter")
```

**Scilab code Exa 4.15.17** calculate depth of sea bed and frequency

```
1  //Chapter-4,Example4_15_17,pg 4-35
2
3  S=29                                          //salinity
4
5  t=2                                           //time
6
7  l=0.01                                        //wavelength
8
9  T=30                                          //
       temperature
10
11 v=1510+1.14*S+4.21*T-0.037*T^2                //velocity
       of ultrasound in sea
12
13 d=v*t/2                                       //depth of
       sea bed
14
15 printf("1)depth of sea bed =")
16
17 disp(d)
18
19 printf("meter")
20
21 f=v/l                                         //frequency
22
23 printf("2) frequency =")
24
25 disp(f)
26
27 printf("Hz")
```

**Scilab code Exa 4.15.18** calculate real thickness

```
1  //Chapter-4,Example4_15_18,pg 4-35
2
```

```
3 v1=5.9*10^3                                    //velocity
    of UW in mild steel
4
5 v2=4.3*10^3                                    //velocity
    of UW in brass
6
7 t2=15*10^-3                                    //thickness
    of brass plate
8
9 t1=v2*t2/v1                                    //since ve;
    ocity is inversly proportional to thickness
10
11 printf("real thickness =")
12
13 disp(t1)
14
15 printf("meter")
```

**Scilab code Exa 4.15.19** calculate thickness of crystal

```
1 //Chapter-4,Example4_15_19,pg 4-36
2
3 t1=4*10^-3                                     //thickness
    of 1st crystal
4
5 n1=400*10^3                                    //frequency
    of 1st crystal
6
7 n2=500*10^3                                    //frequency
    of 2nd crystal
8
9 t2=n1*t1/n2                                    //since
    frquency is inversly proportional to thickness
10
11 printf("thickness of 2nd crystal =")
```

```
12
13 disp(t2)
14
15 printf("meter")
```

**Scilab code Exa 4.15.20** calculate distance at which defect has occurred

```
1 //Chapter-4,Example4_15_20,pg 4-36
2
3 t2=30*10^-6                                    //pulse
     arrival time of defective steel bar
4
5 t1=80*10^-6                                    //pulse
     arrival time of non defective steel bar
6
7 d=40*10^-2                                     //bar
     thickness
8
9 x=(t2/t1)*d
10
11 printf("distance at which defect has occurred =")
12
13 disp(x)
14
15 printf("meter")
```

**Scilab code Exa 4.15.21** calculate echo time

```
1 //Chapter-4,Example4_15_21,pg 4-37
2
3 d=18*10^-3                                     //thickness
4
5 v=5.9*10^3                                     //velocity
```

```
6
7 t=(2*d)/v                                    //echo time
8
9 printf("echo time =")
10
11 disp(t)
12
13 printf("sec")
```

**Scilab code Exa 4.15.22** calculate frquency of vibration

```
1 //Chapter-4,Example4_15_22,pg 4-37
2
3 t=1*10^-3                                    //thickness
     of quartz crystal
4
5 //given t=l/2
6
7 l=t*2                                        //wavelength
8
9 Y=7.9*10^10                                  //young's
     module of crystal
10
11 p=2650                                      //density of
      crystal
12
13 v=sqrt(Y/p)                                 //velocity
     of vibration
14
15 n=v/l                                       //frequency
     of vibration
16
17 printf("frquency of vibration =")
18
19 disp(n)
```

```
20
21 printf("Hz")
```

**Scilab code Exa 4.15.23** calculate length

```
1 //Chapter-4,Example4_15_23,pg 4-38
2
3 d=7.23*10^3                                    //density
4
5 Y=11.6*10^10                                   //Young's
    modulus
6
7 n=20*10^3                                      //frequency
    of wave
8
9 k=1                                            //consider 1
    st harmonic
10
11 l=(k/(2*n))*sqrt(Y/d)                         //arranging
    formula of natural frequency
12
13 printf("length =")
14
15 disp(l)
16
17 printf("meter")
```

**Scilab code Exa 4.15.24** calculate natural frequency and change in thickness

```
1 //Chapter-4,Example4_15_24,pg 4-38
2
3 //for case1
```

```
4  t1=2*10^-3                                    //thicknesss
       of plate
5
6  d=2.65*10^3                                   //density
7
8  Y=8*10^10                                     //Young's
       modulus
9
10 k=1                                           //consider 1
       st harmonic
11
12 n1=(k/(2*t1))*sqrt(Y/d)                       //formula of
       natural frequency
13
14 printf("        1)natural frequency =")
15
16 disp(n1)
17
18 printf("Hz")
19
20 //for case2
21
22 n2=3*10^6                                     //frequency
23
24 t2=(k/(2*n2))*sqrt(Y/d)                       //arranging
       formula of natural frequency
25
26 t=t1-t2                                       //change in
       thickness
27
28 printf("      2)change in thickness =")
29
30 disp(t)
31
32 printf("meter")
```

**Scilab code Exa 4.15.25** calculate average absorption coefficien and total absorption

```
1 //Chapter-4,Example4_15_25,pg 4-39
2
3 l=20                                          //length of
     room
4
5 b=15                                          //bredth of
     room
6
7 h=10                                          //height of
     room
8
9 V=l*b*h                                       //volume of
     room
10
11 S=2*(l*b+b*h+h*l)                            //surface
     area of hall
12
13 T=3                                          //
     Reverberation time
14
15 a=(0.161*V)/(T*S)                            //using
     Sabine's formula
16
17 printf("1) average absorption coefficient =")
18
19 disp(a)
20
21 m=a*S                                        //total
     absorption
22
23 printf("2) total absorption of surface =")
```

```
24
25 disp(m)
26
27 printf("m^2/sec")
```

**Scilab code Exa 4.15.26** calculate natural frequency and change in thickness

```
1  //Chapter-4,Example4_15_26,pg 4-39
2
3  //for case1
4  t1=1.8*10^-3                          //thicknesss
       of plate
5
6  d=2.65*10^3                           //density
7
8  Y=8*10^10                             //Young's
       modulus
9
10 k=1                                   //consider 1
       st harmonic
11
12 n1=(k/(2*t1))*sqrt(Y/d)               //formula of
       natural frequency
13
14 printf("        1)natural frequency =")
15
16 disp(n1)
17
18 printf("Hz")
19
20 //for case2
21
22 n2=2*10^6                             //frequency
23
```

```
24 t2=(k/(2*n2))*sqrt(Y/d)                    //arranging
     formula of natural frequency
25
26 t=t1-t2                                    //change in
     thickness
27
28 printf("      2)change in thickness =")
29
30 disp(t)
31
32 printf("meter")
```

**Scilab code Exa 4.15.27** calculate Youngs modulus

```
1 //Chapter-4,Example4_15_27,pg 4-39
2
3 n=0.4999*10^6                              //frequency
4
5 t=5.5*10^-3                                //thicknesss
      of plate
6
7 d=2.65*10^3                                //density
8
9 k=1                                        //consider 1
     st harmonic
10
11 Y=4*(t^2)*(n^2)*d/k                       //arranging
     formula of natural frequency
12
13 printf("Youngs modulus =")
14
15 disp(Y)
16
17 printf("N/m^2")
```