Scilab Textbook Companion for Solid State Devices by B. S. Nair and S. R. Deepa[1]

Created by
Sakshi Sharma
B.tech
Electronics Engineering
Dr. A.P.J. Abdul Kalam Technical University
College Teacher
None
Cross-Checked by
Bhavani Jalkrish

July 30, 2019

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

Book Description

Title: Solid State Devices

Author: B. S. Nair and S. R. Deepa

Publisher: Phi Learning, New Delhi

Edition: 1

Year: 2010

ISBN: 978-81-203-4106-7

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes

Chapter 1

CRYSTAL STRUCTURES

Scilab code Exa 1.4 PACKING FRACTION OF SIMPLE CUBIC

```
1  clc
2  a=1
3  disp("a= "+string(a)) //initializing value of
       lattice constant(a)=1.
4  r=a/2
5  disp("r=a/2 = "+string(r)) //initializing value of
       radius of atom for simple cubic.
6  v=((4*%pi*(r^3))/3)
7  disp("Volume of one atom ,v=((4*%pi*(r^3))/3) = "+
       string(v)) //calcuation.
8  V=a^3
9  disp("Total Volume of the cube,V=a^3 = "+string(V))
       //calcuation.
10 Fp=(v*100/V)
11 disp("Fp(S.C)=(v*100/V) = "+string(Fp)+"%")//
       calculation
```

Scilab code Exa 1.5 PACKING FRACTION OF BCC

```
clc
a=1
disp("a= "+string(a)) //initializing value of
    lattice constant(a)=1.
r=(sqrt(3)*(a^2/4))
disp("Radius of the atoms,r=(sqrt(3)*(a^2/4)) = "+
    string(r)) //initializing value of radius of atom
     for BCC.
v=((4*%pi*(r^3))/3)*2
disp("Volume of two atom,v=((4*%pi*(r^3))/3)*2 = "+
    string(v)) //calcuation.
V=a^3
disp("Total Volume of the cube,V=a^3 = "+string(V))
    //calcuation.
Fp=(v*100/V)
disp("Fp(B.C.C)=(v*100/V) = "+string(Fp)+"%")//
    calculation
```

Scilab code Exa 1.6 PACKING FRACTION OF FCC

```
clc
a=1
disp("a= "+string(a)) //initializing value of
    lattice constant(a)=1.
r=(a/(2*sqrt(2)))
disp("Radius of the atom,r=(a/(2*sqrt(2)))= "+string
    (r)) //initializing value of radius of atom for
    FCC.
v=(((4*%pi*(r^3))/3)*4)
disp("Volume of the four atom,v=(((4*%pi*(r^3))/3)
    *4) = "+string(v)) //calcuation.
V=a^3
disp("Total volume of the cube,V=a^3 = "+string(V))
    //calcuation.
Fp=(v*100/V)
```

```
11 disp("Fp(F.C.C)=(v*100/V) = "+string(Fp)+"%")//
   calculation
```

Scilab code Exa 1.8 PACKING FRACTION OF DIAMOND CRYSTAL STRUCTURE

```
1  clc
2  a=1
3  disp("a= "+string(a)) //initializing value of
   lattice constant(a)=1.
4  r=((sqrt(3)*a/8))
5  disp("Radius of the atom,r=(sqrt(3)*a/8) )= "+string
   (r)) //initializing value of radius of atom for
   diamond.
6  v=(((4*%pi*(r^3))/3)*8)
7  disp("v=(((4*%pi*(r^3))/3)*8) = "+string(v)) //
   calcuation.
8  V=a^3
9  disp("V=a^3 = "+string(V)) //calcuation.
10 Fp=(v*100/V)
11 disp("Fp(Diamond)=(v*100/V) = "+string(Fp)+"%")//
   calculation
```

Scilab code Exa 1.9 PACKING FRACTION OF BCC

```
1  clc
2  a=5*10^-8
3  disp("a = "+string(a)+" cm") //initializing value of
    lattice constant.
4  r=(sqrt(3)*(a/4))
5  disp("Radius of the atom,r=(sqrt(3)*(a/4)) = "+
   string(r)) //initializing value of radius of atom
    for BCC.
6  v=((4*%pi*(r^3))/3)*2
```

```
7 disp("Volume of the two atoms,v=((4*%pi*(r^3))/3)*2
    = "+string(v)) //calcuation.
8 V=a^3
9 disp("Total Volume of the cube,V=a^3 = "+string(V))
    //calcuation.
10 Fp=(v*100/V)
11 disp("Fp(B.C.C)=(v*100/V) = "+string(Fp)+"%")//
    calculation
```

Scilab code Exa 1.10 MILLER INDICES OF LATTICE PLANE

```
1 clc
2 x=1
3 disp("x intercept = "+string(x)) //initializing
    value of x intercept..
4 y=%inf
5 disp("y intercept = "+string(y)) //initializing
    value of y intercept.
6 z=%inf
7 disp("z intercept = "+string(z)) //initializing
    value of z intercept.
8 h=[1/x]
9 disp("miller indices,h=(1/x) = "+string(h))//
    calculation
10 k=[1/y]
11 disp("k=(1/y) = "+string(k))//calculation
12 l=[1/z]
13 disp("l=(1/z) = "+string(l))//calculation
```

Scilab code Exa 1.11 MILLER INDICES OF LATTICE PLANE

```
1 clc
2 x=%inf
```

```
3 disp("x intercept = "+string(x)) //initializing
    value of x intercept.
4 y=%inf
5 disp("y intercept = "+string(y)) //initializing
    value of Y intercept.
6 z=1
7 disp("z intercept = "+string(z)) //initializing
    value of Z intercept.
8 h=[1/x]
9 disp("miller indices,h=[1/x] = "+string(h))//
    calculation
10 k=[1/y]
11 disp("k=[1/y] = "+string(k))//calculation
12 l=[1/z]
13 disp("l=[1/z] = "+string(l))//calculation
```

Scilab code Exa 1.12 MILLER INDICES OF LATTICE PLANE

```
1 clc
2 x=%inf
3 disp("x intercept = "+string(x)) //initializing
    value of X intercept.
4 y=1
5 disp("y intercept = "+string(y)) //initializing
    value of Y intercept.
6 z=%inf
7 disp("z intercept = "+string(z)) //initializing
    value of Z intercept.
8 h=[1/x]
9 disp("miller indices,h=[1/x] = "+string(h))//
    calculation
10 k=[1/y]
11 disp("k=[1/y] = "+string(k))//calculation
12 l=[1/z]
13 disp("l=[1/z] = "+string(l))//calculation
```

Scilab code Exa 1.13 MILLER INDICES OF LATTICE PLANE

```
clc
x=1
disp("x intercept = "+string(x)) //initializing
    value of X intercept.
y=1
disp("y intercept = "+string(y)) //initializing
    value of Y intercept.
z=%inf
disp("z intercept = "+string(z)) //initializing
    value of Z intercept.
h=[1/x]
disp("miller indices,h=[1/x] = "+string(h))//
    calculation
k=[1/y]
disp("k=[1/y] = "+string(k))//calculation
l=[1/z]
disp("l=[1/z] = "+string(l))//calculation
```

Scilab code Exa 1.14 MILLER INDICES OF LATTICE PLANE

```
clc
x=%inf
disp("x intercept = "+string(x)) //initializing
    value of X intercept.
y=1
disp("y intercept = "+string(y)) //initializing
    value of Y intercept.
z=1
disp("z intercept = "+string(z)) //initializing
    value of Z intercept.
```

```
h=[1/x]
disp("miller indices,h=[1/x] = "+string(h))//
    calculation
k=[1/y]
disp("k=[1/y] = "+string(k))//calculation
l=[1/z]
disp("l=[1/z] = "+string(l))//calculation
```

Scilab code Exa 1.15 MILLER INDICES OF LATTICE PLANE

```
clc
x=2
disp("x intercept = "+string(x)) //initializing
    value of X intercept.
y=2
disp("y intercept = "+string(y)) //initializing
    value of Y intercept.
z=2
disp("z intercept = "+string(z)) //initializing
    value of Z intercept.
c=2
disp("common factor of all the intercept= "+string(c
    )) //initializing value of common factor of all
    the intercepts.
h=[c/x]
disp("miller indices,h=[c/x] = "+string(h))//
    calculation
k=[c/y]
disp("k=[c/y] = "+string(k))//calculation
l=[c/z]
disp("l=[c/z] = "+string(l))//calculation
```

Scilab code Exa 1.16 NO OF SILICON ATOMS

```
clc
Wa=28.1
disp("Wa = "+string(Wa)) //initializing value of
   atomic weight.
D=2.33
disp("D = "+string(D)+"gram/cm^3") //initializing
   value of density.
Na=6.02*10^23
disp("Na = "+string(Na)+"atoms/mole") //initializing
    value of avagadro number.
na =(Na*D)/(Wa)
disp("na =(Na*D)/(Wa)= "+string(na)+" atoms/cm^3")//
   calculation

//the value of na (number of atoms in 1 cm^3 of
   silicon), provided after calculation in the book
   is wrong.
```

Scilab code Exa 1.17 VOLUME DENSITY OF A BCC LATTICE

```
clc
a=5*10^-8
disp("a= "+string(a)+"cm") //initializing value of
   lattice constant.
N=2
disp("N= "+string(N)) //initializing value of no.of
   atoms in unit cell.
V=a^3
disp("V=a^3 = "+string(V)+"cm^3") //initializing
   value of total Volume of the unit cell.
na =(N/(V))
disp("na=(no.of atoms in unit cell/Volume of the
   unit cell) =(N/(V))= "+string(na))//calculation
```

Scilab code Exa 1.18 VOLUME DENSITY OF SILICON

```
clc
a=5.43*10^-8
disp("a = "+string(a)+"cm") //initializing value of
    lattice constant.
N=8
disp("N = "+string(N)) //initializing value of no.
    of atoms in a unit cell.
ns =(N/(a^3))
disp("Number of atom in the cm^3,ns =(N/(a^3))= "+
    string(ns))//calculation
```

Scilab code Exa 1.19 DENSITY OF SILICON

```
clc
a=5.43*10^-8
disp(" a = "+string(a)+"cm") //initializing value of
     lattice constant.
Wa=28.1
disp(" Wa = "+string(Wa)) //initializing value of
    atomic weight.
Na=6.02*10^23
disp(" Na = "+string(Na)) //initializing value of
    avagdro number.
ns=5*10^22
disp(" ns = "+string(ns)+"atoms/cm^3") //
    initializing value of atoms/cm^3.
D =(ns*Wa)/(Na)
disp(" Density of silicon ,D =(ns*Wa)/(Na) = "+string
    (D)+" gm/cm^2")//calculation
```

Scilab code Exa 1.20 VOLUME DENSITY OF A FCC LATTICE

```
1 clc
2 a=4.75*10^-8
3 disp(" a = "+string(a)+"cm") //initializing value of
      lattice constant.
4 N=4
5 disp(" N = "+string(N)) //initializing value of
     number of atoms in the unit cell.
6 na =(N/(a^3))
7 disp(" na =(N/(a^3)) = "+string(na))//calculation
```

Chapter 2

ENERGY BAND THEORY OF SOLIDS

Scilab code Exa 2.1 MINORITY CARRIER CONCENTRATION

```
1 clc
2 n=10^14
3 disp("n = "+string(n)+"/cm^3") //initializing value
    of electrons/cm^3.
4 no=1.5*10^10
5 disp("no. of EHPs/cm^3 = "+string(no)+"/cm^3") //
    initializing value of electron hole pairs/cm^3.
6 p=(no^2/n)
7 disp("minority carriers concentration,p=(no^2/n))= "
    +string(p)+" /cm^3")//calculation
```

Scilab code Exa 2.2 CURRENT GENERATED BY MINORITY CARRIER MOVEMENT

```
1 clc
2 e=1.6*10^-19
3 disp(" electron charge = "+string(e)+"columns") //
    initializing the value of electron charge.
```

```
4 no=1.5*10^10
5 disp("no. of EHPs/cm^3 = "+string(no)+"/cm^3") //
    initializing value of electron hole pairs/cm^3.
6 n=(1/e)
7 disp("Number of free electrons in 1 columns ,n=(1/e)
    )= "+string(n))//calculation
8 i=(1/n)
9 disp("Current by movement of one electrons ,i=(1/n))
    = "+string(i)+" Amphere ")//calculation
10 I=(no*i)
11 disp("Current by movement of (1.5*10^10) electrons ,
    I=(no*i))= "+string(I)+" Amphere ")//calculation
```

Scilab code Exa 2.3 CURRENT GENERATED BY MINORITY CARRIER MOVEMENT

```
1 clc
2 e=1.6*10^-19
3 disp(" Electron charge = "+string(e)+"columns") //
    initializing the value of electron charge.
4 no=2.5*10^13
5 disp("Number of free electrons/cm^3 in Ge ,n
    =2.5*10^13)= "+string(no)+"electrons/cm^3")//
    calculation
6 n=(1/e)
7 disp("Number of free electrons in 1 columns ,n=(1/e)
    )= "+string(n))//calculation
8 i=(1/n)
9 disp("Current by movement of one electrons ,i=(1/n))
    = "+string(i)+" amphere ")//calculation
10 I=(no*i)
11 disp("Current by movement of (2.5*10^13) electrons
    in Ge,I=(no*i))= "+string(I)+" amphere ")//
    calculation
```

Scilab code Exa 2.4 FREQUENCY OF RADIATION EMITTED BY GaAs

```
clc
Eg=1.43*1.6*10^-19
disp(" energy gap = "+string(Eg)+"Volt") //
   initializing the value of energy gap.
h=6.624*10^-34
disp(" plank constant = "+string(h)+"joule")//
   initializing the value of plank constant.
c=3*10^8
disp(" light speed = "+string(c)+"m/s") //
   initializing the value of speed of light.
f=(Eg/h)
disp("frequency of radiation emitted ,f=(Eg/h))= "+
   string(f)+" Hz ")//calculation
lamda=(c/f)
disp("wavelength of radiation emitted,lamda=(c/f))=
   "+string(lamda)+" metre ")//calculation
```

Scilab code Exa 2.5 FREQUENCY OF RADIATION EMITTED

```
clc
Eg1=1.43
disp(" Energy gap of GaAs = "+string(Eg1)+"eV") //
   initializing the value of energy gap of GaAs.
Eg2=2.43
disp(" Energy gap of GaP = "+string(Eg2)+"eV")//
   initializing the value of energy gap of Gap.
h=6.624*10^-34
disp(" plank constant = "+string(h)+"joule")//
   initializing the value of plank constant.
c=3*10^8
```

```
9  disp(" light speed = "+string(c)+"m/s") //
     initializing the value of speed of light.
10 x=(Eg2-Eg1)
11 disp("Difference between the energy gap of GaAs and
     GaP ,x=(Eg2-Eg1))= "+string(x)+" eV")//
     calculation
12 g=(0.5*x)
13 disp("Excess energy gap added to GaAs to form GaAsP
     =(0.5*x))= "+string(g)+" eV ")//calculation
14 Eg=(Eg1+g)
15 disp("Band gap energy GaAs(0.5)P(0.5),Eg=(Eg1+g))= "
     +string(Eg)+" eV ")//calculation
16 lamda=(c*h/(Eg*1.6*10^-19))
17 disp("Wavelength of radiation emitted,w=(c*h/Eg))= "
     +string(lamda)+" metre ")//calculation
```

Scilab code Exa 2.6 FERMI DIREC DISTRIBUTION FUNCTION

```
1  clc
2  E=1.1
3  disp(" E = "+string(E)+"eV") //initializing the
     value of energy level E in the crystal.
4  Ef=0.6
5  disp(" Ef = "+string(Ef)+"eV")//initializing the
     value of energy of fermi level of material.
6  T=300
7  disp(" temp = "+string(T)+"K")//initializing the
     value of temperature.
8  e=2.718
9  disp(" e = "+string(e)) //initializing the value of
     exponential.
10 k=1.38*10^-23
11 disp(" k = "+string(k)+"J/k") //initializing the
     value of boltzmann constant.
12 a=(((E-Ef)*1.6*10^-19)/(k*T))
```

```
13 disp("alpha ,a=(((E-Ef)*1.6*10^-19)/(k*T)))= "+
     string(a))//calculation
14 fE=(1/(1+(e^a)))
15 disp("fE(Fermi Direc Distribution Function),fE
     =(1/(1+(e^a))))= "+string(fE))//calculation
16
17 //the value of Ef is different in the question than
     used in the solution.
18 //I have used the value ,used in the solution(i.e Ef
     =0.6)
```

Scilab code Exa 2.7 FERMI DIREC DISTRIBUTION FUNCTION

```
1 clc
2 E=0.6
3 disp(" E = "+string(E)+"eV") //initializing the
     value of energy level E in the crystal.
4 Ef=1.1
5 disp(" Ef = "+string(Ef)+"eV")//initializing the
     value of fermi level of material.
6 T=300
7 disp(" temp = "+string(T)+"kelvin")//initializing
     the value of temperature.
8 k=1.38*10^-23
9 disp(" k = "+string(k)+"J/k") //initializing the
     value of boltzmann constant.
10 e=2.718
11 disp(" e = "+string(e)) //initializing the value of
     exponential.
12 a=(((-(Ef-E))*1.6*10^-19)/(k*T))
13 disp("alpha ,a=(((-(Ef-E))*1.6*10^-19)/(k*T)))= "+
     string(a))//calculation
14 fE=(1/(1+(e^a)))
15 disp("fE(Fermi Direc Distribution Function),fE
     =(1/(1+(e^a))))= "+string(fE))//calculation
```

```
16
17 //the value of E is different in the question than
     used in the solution.
18 //I have used the value ,used in the solution(i.e E
     =0.6)
```

Scilab code Exa 2.8 FERMI DIREC DISTRIBUTION FUNCTION

```
1  clc
2  E=1.1
3  disp(" E = "+string(E)+"eV") //initializing the
     value of energy level E in the crystal.
4  Ef=0.6
5  disp(" Ef = "+string(Ef)+"eV")//initializing the
     value of fermi level of material.
6  T=1000
7  disp(" Temp = "+string(T)+"K")//initializing the
     value of temperature.
8  k=1.38*10^-23
9  disp(" k = "+string(k)+"J/k") //initializing the
     value of boltzmann constant.
10 e=2.718
11 disp(" e = "+string(e)) //initializing the value of
     exponential.
12 a=(((-Ef+E)*1.6*10^-19)/(k*T))
13 disp("alpha ,a=(((-E+Ef)*1.6*10^-19)/(k*T)))= "+
     string(a))//calculation
14 fE=(1/(1+(e^a)))
15 disp("fE(Fermi Direc Distribution Function),fE
     =(1/(1+(e^a)))= "+string(fE))//calculation
16
17 //The value of Ef is different in the question than
     used in the solution.
18 //I have used the value ,used in the solution(i.e Ef
     =0.6)
```

Scilab code Exa 2.9 FERMI DIREC DISTRIBUTION FUNCTION

```
1  clc
2  Ef=0.6
3  disp(" Ef = "+string(Ef)+"eV") //initializing the
      value of fermi level of material.
4  E=1.1
5  disp(" E = "+string(E)+"eV")//initializing the value
       of level E in a crystal.
6  T=1000
7  disp(" Temp = "+string(T)+"kelvin")//initializing
      the value of temperature.
8  k=1.38*10^-23
9  disp(" k = "+string(k)+"J/k") //initializing the
      value of boltzmann constant.
10 e=2.718
11 disp(" e = "+string(e)) //initializing the value of
      exponential.
12 a=(((E-Ef)*1.6*10^-19)/(k*T))
13 disp("alpha ,a=(((E-Ef)*1.6*10^-19)/(k*T)))= "+
      string(a))//calculation
14 fE=(1/(1+(e^a)))
15 disp("fE(Fermi Direc Distribution Function),fE
      =(1/(1+(e^a))))= "+string(fE))//calculation
16
17 //The value of Ef ,temperature is different in the
      question than used in the solution.
18 //I have used the value of Ef,used in the solution(i
      .e Ef=0.6)and value of temperature used is T=1000
     K
19 //The value of a(alpha),provided in the solution
      after calculation is wrong.As book has used
      different value of the temperature in solution
      than provided in the question.
```

Scilab code Exa 2.10 FERMI DIREC DISTRIBUTION FUNCTION

```
clc
E=1.1
disp(" E = "+string(E)+"eV") //initializing the
    value of level E in a crystal.
Ef=0.6
disp(" Ef = "+string(Ef)+"eV")//initializing the
    value of fermi level of material.
T=2500
disp(" temp = "+string(T)+"K")//initializing the
    value of temperature.
k=1.38*10^-23
disp(" k = "+string(k)+"J/k") //initializing the
    value of boltzmann constant.
e=2.718
disp(" e = "+string(e)) //initializing the value of
    exponential.
a=(((E-Ef)*1.6*10^-19)/(k*T))
disp("alpha ,a=(((E-Ef)*1.6*10^-19)/(k*T)))= "+
    string(a))//calculation
fE=(1/(1+(e^a)))
disp("fE(Fermi Direc Distribution Function),fE
    =(1/(1+(e^a))))= "+string(fE))//calculation

//The value of Ef is different in the question than
    used in the solution.
//I have used the value ,used in the solution(i.e Ef
    =0.6)
```

Scilab code Exa 2.11 FERMI DIREC DISTRIBUTION FUNCTION

```
clc
E=0.6
disp(" E = "+string(E)+"eV") //initializing the
   value of level E in a crystal.
Ef=1.1
disp(" Ef = "+string(Ef)+"eV")//initializing the
   value of fermi level of the material.
T=2500
disp(" Temp = "+string(T)+"kelvin")//initializing
   the value of temperature.
k=1.38*10^-23
disp(" k = "+string(k)+"J/k") //initializing the
   value of boltzmann constant.
e=2.718
disp(" e = "+string(e)) //initializing the value of
   exponential.
a=(((-(Ef-E))*1.6*10^-19)/(k*T))
disp("a ,a=(((E-Ef)*1.6*10^-19)/(k*T)))= "+string(a)
   )//calculation
fE=(1/(1+(e^a)))
disp("fE(Fermi Direc Distribution Function),fE
   =(1/(1+(e^a)))= "+string(fE))//calculation


//The value of E is different in the question than
   used in the solution.
//I have used the value ,used in the solution(i.e E
   =0.6)
```

Scilab code Exa 2.12 NUMBER OF HOLES IN A MEDIMUM DOPED N SEMICONDUCTOR

```
clc
n=10^16
disp("n = "+string(n)+" /cm^3") //initializing value
    of number of electrons per cm^3.
```

```
4 no=1.5*10^10
5 disp("no = "+string(no)+" /cm^3") //initializing
    value of electron hole pairs/cm^3.
6 p=(no^2/n)
7 disp("Number of hole ,p=(no^2/n))= "+string(p)+" /cm
    ^3")//calculation
```

Scilab code Exa 2.13 VALUE OF FERMI LEVEL

```
1 clc
2 e=1.6*10^-19
3 disp("e = "+string(e)+"columb") //initializing the
    value of electronic charge.
4 n=1*10^16
5 disp("n = "+string(n)) //initializing the value of
    number of electrons per cm^3.
6 no=1.5*10^10
7 disp("no = "+string(no)+" /cm^3") //initializing
    value of electron hole pairs/cm^3..
8 T=300
9 disp("T = "+string(no)+" K") //initializing value of
     temperature.
10 k=1.38*10^-23
11 disp("k = "+string(k)+" J/K") //initializing value
    of boltzmann constant.
12 Ef=((k*T/e)*log(n/no))
13 disp("fermi level ,Ef-Efi=((k*T/e)*ln(n/no)) )= "+
    string(Ef)+" eV")//calculation
```

Scilab code Exa 2.14 NUMBER OF HOLES IN A HEAVILY DOPED N SEMICONDUCTOR

```
1 clc
2 no=1.5*10^10
```

```
3 disp("no = "+string(no)+"/cm^3") //initializing
    value of electrons and hole per cm^3.
4 n=1*10^18
5 disp("n = "+string(n)+"/cm^3") //initializing value
    of number of electrons per cm^3.
6 p=(no^2/n)
7 disp("number of holes ,p=(no^2/n))= "+string(p)+" /
    cm^3")//calculation
8
9
10
11 //this is solved problem 2.1 of chapter 2.
```

Scilab code Exa 2.15 VALUE OF INTRINSIC CONCENTRATION

```
1 clc
2 n=1*10^5
3 disp("n = "+string(n)+" /cm^3") //initializing value
    of electrons and hole per cm^3.
4 p=1*10^19
5 disp("p = "+string(p)+" /cm^3") //initializing value
    of number of hole per cm^3
6 no=sqrt(n*p)
7 disp("Value of intrinsic concentration ,no=sqrt(n*p))
    = "+string(no)+" /cm^3")//calculation
8
9
10
11 //this is solved problem 2.2 of chapter 2.
```

Scilab code Exa 2.16 VALUE OF n

```
1 clc
```

```
2 e=1.6*10^-19
3 disp("e = "+string(e)+"columb") //initializing the
    value of electronic charge.
4 Ef_Efi=0.309
5 disp("Ef-Efi = "+string(Ef_Efi)+" eV") //
    initializing the value of difference in the
    energy levels.
6 no=2.5*10^13
7 disp("no = "+string(no)+" /cm^3") //initializing
    value of number of electrons per cm^3
8 T=300
9 disp("T = "+string(T)+" K") //initializing value of
    temperature.
10 ex=2.718
11 disp("exp = "+string(ex)) //initializing the value
    of exponential.
12 k=1.38*10^-23
13 disp("k = "+string(k)+" J/K") //initializing value
    of boltzmann constant.
14 n=no*(ex^((Ef_Efi*e)/(k*T)))
15 disp("number of electrons per cm^3, n=no*(ex^((Ef-
    Efi)/kT)))= "+string(n)+" /cm^3")//calculation
16
17
18
19 //This is solved problem 2.3 of chapter 2.
20 //The value used for "Ef-Efi" in the solution is
    different than provided in the question.
21 //I have used the value provided in the solution (i.
    e Ef_Efi=0.309)
```

Scilab code Exa 2.17 VALUE OF no

```
1 clc
2 e=1.6*10^-19
```

```
3 disp("e = "+string(e)+" columb") //initializing the
    value of electronic charge.
4 Ef=0.4065
5 disp("Ef = "+string(Ef)+" eV") //initializing the
    value of fermi level.
6 n=10^17
7 disp("n = "+string(n)+" /cm^3") //initializing value
     of number of electrons per cm^3.
8 T=300
9 disp("T = "+string(T)+" K") //initializing value of
    temperature.
10 ex=2.718
11 disp("exp = "+string(ex)) //initializing the value
    of exponential.
12 k=1.38*10^-23
13 disp("k = "+string(k)+" J/K") //initializing value
    of boltzmann constant.
14 no=n/(ex^((Ef*e)/(k*T)))
15 disp("Number of electrons per cm^3, no=n/(ex^((Ef)/
    kT)))= "+string(no)+" electrons/cm^3")//
    calculation
16
17
18 //this is solved problem 2.4 of chapter 2.
19 //the value used for "n" in the solution is
    different than provided in the question.
20 //I have used the value provided in the solution (i.
    e n=10^17)
```

Scilab code Exa 2.18 RESTIVITY AND RESISTANCE OF SEMICONDUCTOR

```
1 clc
2 e=1.6*10^-19
3 disp("e = "+string(e)+"columb") //initializing the
    value of electronic charge.
```

```
4 n=1*10^22
5 disp("n = "+string(n)+" /m^3") //initializing value
    of number of electrons per cm^3
6 u=1200*10^-4
7 disp("u = "+string(u)+" m^2/Vs") //initializing the
    value of mobility.
8 L=0.1*10^-2
9 disp("L = "+string(L)+" m") //initializing the value
     of length.
10 A=100*10^-12
11 disp("A = "+string(A)+" m^2") //initializing the
    value of area of cross section.
12 sigma=n*e*u
13 disp("conductivity ,sigma=n*e*u)= "+string(sigma)+"
    siemen/m")//calculation.
14 p=(1/sigma)
15 disp("Resistivity ,p=(1/sigma))= "+string(p)+" ohm
    metre")//calculation.
16 R=(p*L/A)
17 disp("resistance ,R=(p*L/A))= "+string(R)+" ohm")//
    calculation.
18
19
20 //this is solved problem 2.5 of chapter 2.
21 //the value used for "A" in the solution is
    different than provided in the question.
22 //I have used the value provided in the solution (i.
    e A=100*10^-12)
```

Scilab code Exa 2.19 VALUE OF DRIFT CURRENT

```
1 clc
2 R=52.08 *10^3
3 disp("R = "+string(R)+"ohm") //initializing value of
     Resistance.
```

```
4 V=5
5 disp("V = "+string(V)+" volt") //initializing value
    of voltage.
6 I=(V/R)
7 disp("Drift current,I=(V/R))= "+string(I)+" amphere"
    )//calculation
8
9
10 //this is solved problem 2.6 of chapter 2.
```

Scilab code Exa 2.20 FREQUENCY OF RADIATION EMITTED

```
1 clc
2 Eg1=1.43
3 disp(" Energy gap of GaAs = "+string(Eg1)+"eV") //
    initializing the value of energy gap of GaAs.
4 Eg2=2.43
5 disp(" Energy gap of GaP = "+string(Eg2)+"eV")//
    initializing the value of energy gap of Gap.
6 h=6.624*10^-34
7 disp(" Plank constant = "+string(h)+"joule")//
    initializing the value of plank constant.
8 c=3*10^8
9 disp(" Light speed = "+string(c)+"m/s") //
    initializing the value of speed of light.
10 x=(Eg2-Eg1)
11 disp("Difference between the energy gap of GaAs and
    GaP ,x=(Eg2-Eg1))= "+string(x)+" eV")//
    calculation
12 g=(0.4*x)
13 disp("Excess energy gap added to GaAs to form GaAsP
    ,(0.4*x))= "+string(g)+" eV ")//calculation
14 Eg=(Eg1+g)
15 disp("Band gap energy GaAsP,Eg=(Eg1+g))= "+string(Eg
    )+" eV ")//calculation
```

```
16 lamda=(c*h/(Eg*1.6*10^-19))
17 disp("wavelength of radiation emitted,lamda=(c*h/Eg)
     )= "+string(lamda)+" metre ")//calculation
18
19 //this is solved problem 2.7 of chapter 2.
```

Scilab code Exa 2.21 BAND GAP ENERGY AND VALUE OF X

```
1  clc
2  Eg1=1.43
3  disp(" Energy gap of GaAs = "+string(Eg1)+" eV") //
     initializing the value of energy gap of GaAs.
4  Eg2=2.43
5  disp(" Energy gap of GaP = "+string(Eg2)+" eV")//
     initializing the value of energy gap of Gap.
6  h=6.624*10^-34
7  disp(" Plank constant = "+string(h)+" joule")//
     initializing the value of plank constant.
8  c=3*10^8
9  disp(" Light speed = "+string(c)+" m/s") //
     initializing the value of speed of light.
10 lamda=540*10^6
11 disp(" lamda = "+string(lamda)+" m") //initializing
     the value of wavelength.
12 x=(Eg2-Eg1)
13 disp("Difference between the energy gap of GaAs and
     GaP ,x=(Eg2-Eg1))= "+string(x)+" eV")//
     calculation
14 Eg=((c*h/(lamda*(1.6*10^-19))))
15 disp("Band gap energy,Eg=(c*h/lamda*(1.6*10^-19)))=
     "+string(Eg)+" eV")//calculation
16 X=Eg-(Eg1)
17 disp("X=Eg-(Eg1)= "+string(X))//calculation
18
19
```

```
20
21
22 //this is solved problem 2.8 of chapter 2.
23 //the value of Eg(band gap energy )is provided wrong
       in the book after calculation.Due to this value
      of X ,also differ.
```

Scilab code Exa 2.22 VALUE OF THERMAL EQUILIBRIUM HOLE CONCENTRATION IN SILICON

```
1 clc
2 T1=500
3 disp(" Temperature 1 = "+string(T1)+"K") //
      initializing the value of temperature 1.
4 Nv=2*10^19
5 disp(" Nv = "+string(Nv)+"cm^-3")//initializing the
      value of effective density of state for valence
      band .
6 T2=300
7 disp(" Temperature 2 = "+string(T2)+"K")//
      initializing the value of temperature 2.
8 NV=(Nv*((500/300)^(3/2)))
9 disp("NV at 500K=(Nv((500/300)^(3/2)))= "+string(NV
      )+" cm^-3 ")//calculation
10
11
12 //this is solved problem 2.9 of chapter 2.
```

Scilab code Exa 2.23 FRACTION OF ELECTRONS THAT ARE STILL IN THE DONOR STATE

```
1 clc
2 Nd=1*10^17
3 disp("Nd = "+string(Nd)+"cm^-3") //initializing the
      value of effective energy density of state.
```

```
4  Ec_Ed=0.045
5  disp(" Ec_Ed = "+string(Ec_Ed))//initializing the
       value of donor ionisation level.
6  Vt=0.0259
7  disp("Vt = "+string(Vt)+" eV ")//initializing the
       value of thermal voltage.
8  Nc=2.8*10^19
9  disp(" Nc = "+string(Nc)+"cm^-3")//initializing the
       value of effective density of state of conduction
        band.
10 e=2.718
11 disp("exp = "+string(e))//initializing the value of
       exponential.
12 N=(((Nc/Nd)*e^((-(Ec_Ed))/Vt))+1)^-1
13 disp("Fraction of electron still in the donor state
       ,(nd/(nd+n)=(((Nc/Nd)*e^((-Ec_Ed)/Vt))+1)^-1)= "+
       string(N))//calculation
14
15
16 //this is solved problem 2.10 of chapter 2.
```

Scilab code Exa 2.24 FRACTION OF HOLES THAT ARE STILL IN THE ACCEPTOR STATE

```
1  clc
2  Na=1*10^16
3  disp("Na = "+string(Na)+" cm^-3")//initializing the
       value of acceptor concentration.
4  Ea_Ev=0.045
5  disp("Ea_Ev = "+string(Ea_Ev))//initializing the
       value of boron acceptor ionization energy.
6  Nv=(1.04*10^19)
7  disp("Nv = "+string(Nv)+" cm^-3")//initializing the
       value of effective density of state for valence
       band.
8  Vt=(0.0259)
```

```
9  disp("Vt = "+string(Vt)+" eV")//initializing the
       value of thermal voltage.
10 p=(1+((Nv/(4*Na))*exp(-(Ea_Ev)/Vt)))^(-1)
11 disp("Fraction of holes that are still in the
       acceptor state,(pa/(pa+p))=(1+((Nv/4*Na)*exp(-(Ea
       -Ev)/Vt)))^(-1)= "+string(p))//calculation
12
13 //this is solved problem 2.11 of chapter 2.
```

Scilab code Exa 2.25 ELECTRON AND HOLE CONCENTRATION

```
1  clc
2  Nd=1*10^17
3  disp("Nd = "+string(Nd)+" cm^-3") //initializing the
       value of donor concentration.
4  Na=0
5  disp("Na = "+string(Na)+" cm^-3")//initializing the
       value of acceptor concentration.
6  no=1.5*10^10
7  disp("ni = "+string(no)+" cm^-3")//initializing the
       value of electron hole per cm^3.
8  n=(-(Na-Nd)+sqrt((Na-Nd)^2+4*no))/2
9  disp("Electron concentration,n=(-(Na-Nd)+sqrt((Na-Nd
       )^2+4*no))/2)= "+string(n)+" cm^-3")//calculation
10 p=(no^2/n)
11 disp("Hole concentration,p)= "+string(p)+" cm^-3")//
       calculation
12
13
14
15 //this is solved problem 2.13 of chapter 2.
```

Scilab code Exa 2.26 ELECTRON AND HOLE CONCENTRATION

```
clc
Nd=6*10^16
disp("Nd = "+string(Nd)+" cm^-3") //initializing the value of donor concentration.
Na=10^17
disp("Na = "+string(Na)+" cm^-3")//initializing the value of acceptor concentration.
no=1.5*10^10
disp("no = "+string(no)+" cm^-3")//initializing the value of electron and hole per cm^3.
p=((Na-Nd)+sqrt((Na-Nd)^2+4*no))/2
disp("Hole concentration ,n=(-(Na-Nd)+sqrt((Na-Nd)^2+4*no))/2)= "+string(p)+" cm^-3")//calculation
n=(no^2/p)
disp("Electron concentration ,n=(no^2/p))= "+string(n))//calculation


//this is solved problem 2.14 of chapter 2.
//the value of Na,Nd in the solution is different than provided in the question
//I have used the value used in the solution(i.e Na=10^17,Nd=6*10^16)
```

Scilab code Exa 2.27 ELECTRON AND HOLE CONCENTRATION

```
clc
Nd=6*10^16
disp("Nd = "+string(Nd)+" cm^-3") //initializing the value of donor concentration.
Na=10^17
disp("Na = "+string(Na)+" cm^-3")//initializing the value of acceptor concentration.
no=1.5*10^10
```

```
7 disp("no = "+string(no)+" cm^-3")//initializing the
    value of electron and hole per cm^3.
8 p=((Na-Nd)+sqrt((Na-Nd)^2+4*no))/2
9 disp("Hole concentration ,n=(-(Na-Nd)+sqrt((Na-Nd)
    ^2+4*no))/2)= "+string(p)+"cm^-3")//calculation
10 n=(no^2/p)
11 disp("Electron concentration ,n=(no^2/p))= "+string(n
    ))//calculation
12
13
14 //this is solved problem 2.15 of chapter 2.
15 //the value of Na,Nd in the solution is different
    than provided in the question
16 //I have used the value used in the solution(i.e Na
    =10^17,Nd=6*10^16)
```

Scilab code Exa 2.28 VALUE OF THERMAL EQUILIBRIUM HOLE CONCENTRATION IN SILICON

```
1 clc
2 Nv1=1.04*10^19
3 disp("Nv = "+string(Nv1)+" cm^-3")//initializing the
    value of valence band concentration at 300K.
4 Ef_Ev=0.3
5 disp("Ef_Ev = "+string(Ef_Ev)+" eV")//initializing
    the value of boron acceptor ionization energy.
6 T1=300
7 disp("T = "+string(T1)+"K")//initializing the value
    of temperature 1.
8 T2=500
9 disp("T = "+string(T2)+"K")//initializing the value
    of temperature 2.
10 Vt1=0.0259
11 disp("Vt1 = "+string(Vt1)+"eV")//initializing the
    value of thermal voltage at 300K.
12 k=1.38*10^-23
```

```
13 disp("k = "+string(k)+"J/K") //initializing value of
     boltzmann constant.
14 e=1.6*10^-19
15 disp("e = "+string(e)+"columb") //initializing the
     value of electronic charge.
16 K1=(Nv1/((T1)^(3/2)))
17 disp("Value of constant,K1=(Nv/((T)^(3/2)))= "+
     string(K1)+" cm^-3 K(-2/3)")//calculation
18 Nv2=K1*T2^(3/2)
19 disp("Value of valence band concentration at 500K,Nv
     =K1*T(3/2)= "+string(Nv2)+" cm^-3")//calculation
20 VT=(k*T2/e)
21 disp("Value of parameter VT at 500K,VT=(K*T/e)= "+
     string(VT)+" cm^-3")//calculation
22 p=(Nv2*(exp(-(Ef_Ev)/(VT))))
23 disp("Hole concentration,p=(Nv*(exp(Ef_Ev)/(VT)))= "
     +string(p)+" cm^-3")//calculation
24
25 //this is solved problem 2.16 of chapter 2.
```

Scilab code Exa 2.29 INTRINSIC CARRIER CONCENTRATION

```
1 clc
2 Nv=7*10^18
3 disp("Nv = "+string(Nv)+"cm^-3")//initializing the
     value of valence band concentration at 300K.
4 Nc=4.7*10^17
5 disp("Nc = "+string(Nc)+"cm^-3")//initializing the
     value of conduction band concentration at 300K.
6 T1=300
7 disp("T = "+string(T1)+"K")//initializing the value
     of temperature 1.
8 T2=450
9 disp("T = "+string(T2)+"K")//initializing the value
     of temperature 2.
```

```
10 Vt1=0.0259
11 disp("Vt1 = "+string(Vt1)+"eV")//initializing the
     value of thermal voltage at 300K.
12 Vt2=0.03881
13 disp("Vt2 = "+string(Vt2)+"eV")//initializing the
     value of thermal voltage at 450K.
14 Eg=1.42
15 disp("Eg = "+string(Eg)+"eV")//initializing the
     value of thermal voltage.
16 no=(sqrt(Nc*Nv*(exp(-Eg/Vt1))))
17 disp("intrinsic concentration at 300K,no=(sqrt(Nc*Nv
     *(exp(-Eg/Vt1))))= "+string(no))//calculation
18 K1=(Nc/((T1)^(3/2)))
19 disp("Value of constant,K1=(Nc/((T)^(3/2)))= "+
     string(K1))//calculation
20 k1=(K1*T2^(3/2))
21 disp("Value of constant k1 at 450K ,k1=(K1*T2^(3/2))
     = "+string(k1))//calculation
22 K2=(Nv/((T1)^(3/2)))
23 disp("Value of constant,K2=(Nv/((T1)^(3/2)))= "+
     string(K2))//calculation
24 k2=(K2*T2^(3/2))
25 disp("Value of constant k2 at 450K ,k2=(K2*T2^(3/2))
     = "+string(k2))//calculation
26 K=k1*k2
27 disp("Value of constant K,= "+string(K))//
     calculation
28 no1=(sqrt(K*(exp(-Eg/Vt2))))
29 disp("intrinsic concentration at 450K,no=(sqrt(K*(
     exp(-Eg/Vt2))))= "+string(no1)+" cm^3")//
     calculation
30 //this is solved problem 2.17 of chapter 2.
```

Scilab code Exa 2.30 VALUE OF DONOR CONCENTRATION

```
clc
Nv=1.04*10^19
disp("Nv = "+string(Nv)+"cm^-3")//initializing the value of valence band concentration at 300K.
Nc=2.8*10^19
disp("Nc = "+string(Nc)+"cm^-3")//initializing the value of conduction band concentration at 300K.
T1=300
disp("T = "+string(T1)+"K")//initializing the value of temperature 1.
T2=550
disp("T = "+string(T2)+"K")//initializing the value of temperature 2.
Vt1=0.0259
disp("Vt1 = "+string(Vt1)+"eV")//initializing the value of thermal voltage at 300K.
Vt2=0.0474
disp("Vt2 = "+string(Vt2)+"eV")//initializing the value of thermal voltage at 550K.
Eg1=1.12
disp("Eg1 = "+string(Eg1)+"eV")//initializing the value of thermal voltage.
no=(sqrt(Nc*Nv*(exp(-Eg1/Vt1))))
disp("intrinsic concentration at 300K,no=(sqrt(Nc*Nv*(exp(-Eg1/Vt1))))= "+string(no))//calculation
K1=(Nc/((T1)^(3/2)))
disp("Value of constant,K1=(Nc/((T)^(3/2)))= "+string(K1))//calculation
k1=(K1*T2^(3/2))
disp("Value of constant k1 at 550K ,k1=(K1*T2^(3/2)) = "+string(k1))//calculation
K2=(Nv/((T1)^(3/2)))
disp("Value of constant,K2=(Nv/((T1)^(3/2)))= "+string(K2))//calculation
k2=(K2*T2^(3/2))
disp("Value of constant k2 at 550K ,k2=(K2*T2^(3/2)) = "+string(k2))//calculation
K=k1*k2
```

```
27 disp("Value of constant K,= "+string(K))//
     calculation
28 no1=(sqrt(K*(exp(-Eg1/Vt2))))
29 disp("Intrinsic concentration at 550K,no=(sqrt(K*(
     exp(-Eg1/Vt2))))= "+string(no1)+" cm^3")//
     calculation
30 Nd=(4*(no1^2)/(1.2))
31 disp("Donor concentration at which intrinsic
     concentration is 10% of the total electron
     concentration ,Nd=(4*(no1^2)/(1.2))= "+string(Nd)+
     " cm^3")//calculation
32 //this is solved problem 2.18 of chapter 2.
33 //the value of temperature and % of the intrinsic
     carrier concentration given in the question is
     different than used in the solution.
34 //I have used the value provided in the solution (i.
     e T2=550 and % of the intrinsic carrier
     concentration =10%)
35 //the value of Donor concentration at which
     intrinsic concentration is 10% of the total
     electron concentration(Nd),is provided wrong in
     the book after calculation.
```

Scilab code Exa 2.31 DONOR CONCENTRATION IN SILICON

```
1 clc
2 Ec_Ef=0.2
3 disp("Ec_Ef = "+string(Ec_Ef)+" eV") //initializing
     the value of difference in the energy levels.
4 Nc=2.8*10^19
5 disp("Nc = "+string(Nc)+" cm^-3")//initializing the
     value of conduction band concentration.
6 Na=3*10^16
7 disp("Na = "+string(Na)+" cm^-3")//initializing the
     value of acceptor concentration.
```

```
Vt=0.0259
disp("Vt = "+string(Vt)+" eV")//initializing the
    value of thermal voltage at 300K.
Nd=(Nc*(exp(-(Ec_Ef)/(Vt))))+(Na)
disp("Donor concentration ,Nd=(Nc*(exp(-(Ec_Ef)/(Vt))
    ))+(Na)= "+string(Nd)+" cm^-3")//calculation


//this is solved problem 2.19 of chapter 2.
```

Scilab code Exa 2.32 INTRINSIC CARRIER CONCENTRATION

```
clc
Nv=6*10^18
disp("Nv = "+string(Nv)+"cm^-3")//initializing the
    value of valence band concentration at 300K.
Nc=1.04*10^19
disp("Nc = "+string(Nc)+"cm^-3")//initializing the
    value of conduction band concentration at 300K.
T1=300
disp("T1 = "+string(T1)+"K")//initializing the value
     of temperature 1.
T2=200
disp("T2 = "+string(T2)+"K")//initializing the value
     of temperature 2.
Vt1=0.0259
disp("Vt1 = "+string(Vt1)+"eV")//initializing the
    value of thermal voltage at 300K.
Vt2=0.0173
disp("Vt2 = "+string(Vt2)+"eV")//initializing the
    value of thermal voltage at 200K.
Eg1=0.60
disp("Eg1 = "+string(Eg1)+"eV")//initializing the
    value of thermal voltage used for 300K .
no=(sqrt(Nc*Nv*(exp(-Eg1/Vt1))))
```

```
17 disp("intrinsic concentration at 300K,no=(sqrt(Nc*Nv
   *(exp(-Eg1/Vt1))))= "+string(no))//calculation
18 Eg2=0.66
19 disp("Eg2 = "+string(Eg2)+"eV")//initializing the
   value of thermal voltage used for 200K.
20 K1=(Nc/((T1)^(3/2)))
21 disp("Value of constant,K1=(Nc/((T)^(3/2)))= "+
   string(K1))//calculation
22 k1=(K1*T2^(3/2))
23 disp("Value of constant k1 at 200K ,k1=(K1*T2^(3/2))
   = "+string(k1))//calculation
24 K2=(Nv/((T1)^(3/2)))
25 disp("Value of constant,K2=(Nv/((T1)^(3/2)))= "+
   string(K2))//calculation
26 k2=(K2*T2^(3/2))
27 disp("Value of constant k2 at 200K ,k2=(K2*T2^(3/2))
   = "+string(k2))//calculation
28 K=k1*k2
29 disp("Value of constant K,= "+string(K))//
   calculation
30 no1=(sqrt(K*(exp(-Eg2/Vt2))))
31 disp("intrinsic concentration at 200K,no=(sqrt(K*(
   exp(-Eg2/Vt2))))= "+string(no1)+" cm^3")//
   calculation
32
33 //this is solved problem 2.20 of chapter 2.
34 //The answer of intrinsic concentration at 300K,(no)
    is provided wrong in the book.
```

Scilab code Exa 2.33 RATIO OF INTRINSIC CONCENTRATION

```
1 clc
2 Eg1=2
3 disp("Eg1 = "+string(Eg1)+" eV") //initializing the
   value of band energy gap for semiconductor1.
```

```
4 Eg2=2.2
5 disp("Eg2 = "+string(Eg2)+" eV")//initializing the
    value of band energy gap for semiconductor2.
6 Vt=0.0259
7 disp("Vt = "+string(Vt)+" eV")//initializing the
    value of thermal voltage at 300K.
8 No=sqrt(exp((-Eg1/Vt)-(-Eg2/Vt)))
9 disp("Ratio of their intrinsic concentration at 300K
    ,(no1/no2)=sqrt(exp((-Eg1/Vt)-(-Eg2/Vt)))= "+
    string(No))//calculation
```

Chapter 3

CARRIER TRANSPORT IN SEMICONDUCTOR

Scilab code Exa 3.1 VALUE OF HALL ELECTRIC FIELD

```
1  clc
2  I=5*10^-3
3  disp("I = "+string(I)+" amphere") //initializing
      value of current flowing through the sample.
4  B=1*10^-6
5  disp("B= "+string(B)+" Tesla") //initializing value
      of magnetic field.
6  w=0.01*10^-2
7  disp("w = "+string(w)+" m") //initializing value of
      width of germanium sample.
8  l=0.1*10^-2
9  disp("l = "+string(l)+" m") ////initializing value
      of length of germanium sample.
10 t=0.001*10^-2
11 disp("t = "+string(t)+" m") ////initializing value
      of thickness of germanium sample.
12 p=10^17
13 disp("p = "+string(p)+" atoms/m^3") //initializing
      value of doped acceptor atoms.
```

```
e=1.6*10^-19
disp("e = "+string(e)+" columb") //initializing
    value of charge of electron.
EH=(I*B)/(w*t*p*e)
disp("Hall electric field ,EH=(I*B)/(w*t*p*e)= "+
    string(EH)+" V/m")//calculation
E=EH*10^-2
disp("Hall electric field in centimeter ,EH=(I*B)/(w*
    t*p*e)= "+string(E)+" V/cm")//calculation
```

Scilab code Exa 3.2 VALUE OF HALL COFFICIENT

```
clc
I=5*10^-3
disp("I = "+string(I)+" amphere") //initializing
    value of current flowing through the sample.
B=1*10^-6
disp("B= "+string(B)+" Tesla") //initializing value
    of magnetic field.
w=0.01*10^-2
disp("w = "+string(w)+" m") //initializing value of
    width of germanium sample.
l=0.1*10^-2
disp("l = "+string(l)+" m") //initializing value of
    length of germanium sample.
t=0.001*10^-2
disp("t = "+string(t)+" m") //initializing value of
    thickness of germanium sample.
p=10^17
disp("p = "+string(p)+" atoms/cm^3") //initializing
    value of doped acceptor atoms.
e=1.6*10^-19
disp("e = "+string(e)+" columb") //initializing
    value of charge of electron.
Rh=(1/(p*e))
```

```
17 disp("hall cofficient ,Rh=(1/(p*e))= "+string(Rh)+"
   cm^3/C")//calculation
```

Scilab code Exa 3.3 VALUE OF HALL VOLTAGE

```
1  clc
2  I=10*10^-3
3  disp("I = "+string(I)+" amphere") //initializing
   value of current flowing through the sample.
4  B=10*10^-6
5  disp("B= "+string(B)+" Tesla") //initializing value
   of magnetic field.
6  w=0.01*10^-2
7  disp("w = "+string(w)+" m") //initializing value of
   width of germanium sample.
8  l=0.1*10^-2
9  disp("l = "+string(l)+" m") //initializing value of
   length of germanium sample.
10 t=0.001*10^-2
11 disp("t = "+string(t)+" m") //initializing value of
   thickness of germanium sample.
12 n=10^16
13 disp("n = "+string(n)+" atoms/cm^3") //initializing
   value of doped donor atoms.
14 e=1.6*10^-19
15 disp("e = "+string(e)+" columb") //initializing
   value of charge of electron.
16 Vh=((I*B)/(n*e*t))
17 disp("Hall voltage ,Vh=((I*B)/(n*e*t)))= "+string(Vh)
   +" V")//calculation
```

Scilab code Exa 3.4 VALUE OF HALL RESISTANCE

```
clc
I=10*10^-3
disp("I = "+string(I)+" amphere") //initializing
   value of current flowing through the sample.
B=10*10^-6
disp("B= "+string(B)+" Tesla") //initializing value
   of magnetic field.
w=0.01*10^-2
disp("w = "+string(w)+" m") //initializing value of
   width of germanium sample.
l=0.1*10^-2
disp("l = "+string(l)+" m") ////initializing value
   of length of germanium sample.
t=0.001*10^-2
disp("t = "+string(t)+" m") ////initializing value
   of thickness of germanium sample.
p=10^18
disp("p = "+string(p)+" atoms/cm^3") //initializing
   value of doped donor atoms.
e=1.6*10^-19
disp("e = "+string(e)+" columb") //initializing
   value of charge of electron.
Yh=((B)/(p*e*t))
disp("Hall voltage ,Yh=((B)/(p*e*t)))= "+string(Yh)+"
    ohm")//calculation
```

Scilab code Exa 3.8 MINIMUM CONDUCTIVITY AND MAXIMUM RESISTIVITY

```
clc
no=1.5*10^10
disp("no = "+string(no)) //initializing value of
   electron hole per cm^3.
n=2*10^16
disp("n= "+string(n)) //initializing value of number
    of electrons per cm^3.
```

```
un=1200
disp("un = "+string(un)) //initializing value of
    mobility of n-type carrier.
up=500
disp("up = "+string(up)) ////initializing value of
    mobility of p-type carrier.
e=1.6*10^-19
disp("e = "+string(e)+" columb") //initializing
    value of charge of electron.
p=(1/(2*e*no*(sqrt(un*up))))
disp("resistivity ,p=(1/(2*e*no*(sqrt(un/up))))= "+
    string(p)+" ohm")//calculation
sigmamin=(1/p)
disp("conductivity ,s=(1/p))= "+string(sigmamin)+" S
    /cm")//calculation
sigma=e*no*(un+up)
disp("intrinsic conductivity ,sigma=e*no*(un+up))= "
    +string(sigma)+" S/cm")//calculation
```

Scilab code Exa 3.10 VALUE OF DIFFERENCE IN FERMI LEVELS

```
clc
po=10^18
disp("po = "+string(po)+" cm^-3") //initializing
    value of N type doping level.
no=1.5*10^10
disp("no = "+string(no)+" /cm^-3") //initializing
    value of electron and hole concentration per cm
    ^3.
Po=10^17
disp("P(o)= "+string(Po)+" cm^-3") //initializing
    value of excess hole concentration.
A=0.1
disp("A = "+string(A)+" cm^-2") //initializing the
    value of area.
```

```
10 up=300
11 disp("up = "+string(up)+" cm^2/Vs") //initializing
     value of mobility of p-type carrier.
12 t=7*10^-9
13 disp("t = "+string(t)+" sec") //initializing value
     of transit time.
14 T=300
15 disp("T = "+string(T)+" K") //initializing value of
     temperature.
16 Vt=0.0259
17 disp("Vt = "+string(Vt)+" eV") //initializing value
     of thermal voltage at 300K.
18 x=500*10^-8
19 disp("x = "+string(x)+" cm") //initializing value of
      distance at which difference in fermi level is
     to calculated.
20 Dp=(Vt*up)
21 disp("Diffusion cofficient ,Dp=(Vt*up))= "+string(Dp)
     +" cm^2/s")//calculation
22 Lp=(sqrt(Dp*t))
23 disp("Diffusion length ,Lp=(sqrt(Dp*t)))= "+string(Lp
     )+" cm")//calculation
24 px=(po+(Po*exp(-x/Lp)))
25 disp("Excess charge generated ,p(x)=(po+(P(o)*exp(-x/
     Lp)))= "+string(px)+" cm^-3")//calculation
26 Efi_Efp=(Vt*log(px/no))
27 disp("Fermi level ,Efi_Efp=(Vt*log(p(x)/no)))= "+
     string(Efi_Efp)+" eV")//calculation
```

Scilab code Exa 3.11 HOLE CURRENT AND STORED EXCESS HOLE CHARGES

```
1 clc
2 A=0.1*10^-4
3 disp("A = "+string(A)+" cm^2") //initializing value
     of area.
```

```
4 Dp=7.77*10^-4
5 disp("Dp= "+string(Dp)+" cm^2/s") //initializing
    value of diffusion cofficient.
6 Lp=0.233*10^-5
7 disp("Lp = "+string(Lp)+" cm") //initializing value
    of diffusion length.
8 x=500*10^-8
9 disp("x = "+string(x)+" cm") //initializing value of
     distance
10 P=10^17*10^6
11 disp("P(O)-po = "+string(P)) //initializing value of
     P(O)-po
12 e=1.6*10^-19
13 disp("e = "+string(e)+"column")//initializing value
    of charge of electron.
14 I=(((e*A*Dp*P)/Lp)*exp(-x/Lp))
15 disp("Hole current ,I=(((e*A*Dp*[P(O)-po])/Lp)*exp(-x
    /Lp))= "+string(I)+"amphere")//calculation
16 Q=(e*A*Dp*Lp*P)
17 disp("stored excess hole ,Q=(e*A*Dp*Lp*P))= "+string
    (Q)+"C")//calculation
18
19 //the value of current(I) given after calculation in
     the book is wrong, (as the value of Lp used in
    the formula while finding value of hole current (
    I)at two places //is used different).
20 //I have used the value Lp=0.233*10^-5 cm
```

Scilab code Exa 3.12 ELECTRONIC CONCENTRATION AND MOBILITY

```
1 clc
2 I=2*10^-3
3 disp("I = "+string(I)+" amphere") //initializing
    value of current flowing through the sample.
4 B=1000*10^-4
```

```
5  disp("B= "+string(B)+" Tesla") //initializing value
     of magnetic field.
6  w=0.2*10^-3
7  disp("w = "+string(w)+" mm") //initializing value of
     width of sample.
8  l=2*10^-3
9  disp("l = "+string(l)+" m") //initializing value of
     length of sample.
10 t=0.02*10^-3
11 disp("t = "+string(t)+" m") //initializing value of
     thickness of sample.
12 Vaa=10
13 disp("Vaa = "+string(Vaa)+" V") //initializing value
     of applied voltage.
14 Vh=-10*10^-3
15 disp("Vh = "+string(Vh)+" V") //initializing value
     of hall voltage.
16 e=1.6*10^-19
17 disp("e = "+string(e)+" columb") //initializing
     value of charge of electron.
18 n=((I*B)/(e*t*Vh))
19 disp("electron concentration,n=((I*B)/(e*t*Vh))= "+
     string(n)+" m^-3")//calculation
20 un=(I*l/(e*abs(n)*Vaa*w*t))
21 disp("mobility,un=(I*L/(e*n*Vaa*w*t))= "+string(un)+
     " m^2/Vs")//calculation
```

Scilab code Exa 3.14 INDUCED ELECTRIC FIELD

```
1  clc
2  disp("ND_x = ((10^17)-(10^18*x))") //donor
     concentration in an N type semiconductor
3  disp("differentiating above equation with resprct to
     x")
4  disp("d[ND_x]/dx = (-10^18) cm^-4")
```

```
disp("now, electric field is given by ")
disp("E_x = -(VT/ND_x)*(d[ND_x]/dx) = (0.0259*10^18)/((10^15)-(10^18*x))")// equation for electric field
disp("for x = 0")
x = 0
E_x = (0.0259*10^18)/((10^15)-(10^18*x))
disp("E_x = "+string(E_x)+"V/cm")
disp("for x = 1*10^-4 cm")
x = 1*10^-4
E_x = (0.0259*10^18)/((10^15)-(10^18*x))
disp("E_x = "+string(E_x)+"V/cm")
```

Scilab code Exa 3.17 ELECTRON AND HOLE CONCENTRATION AND DRIFT CURRENT DENSITY

```
clc
Nd=10^17
disp("Nd = "+string(Nd)+" /cm^3") //initializing value of donor concentration.
Na=0
disp("Na= "+string(Na)+" /cm^3") //initializing value of acceptor concentration.
no=1.8*10^6
disp("no = "+string(no)+" /cm^3") //initializing value of electron and hole concentration per cm^3.
E=5
disp("E = "+string(E)+" V/cm") ////initializing value of electric field.
un=7500
disp("un = "+string(un)+" cm^2/s") ////initializing value of mobility.
n1=10^17
disp("n1= "+string(n1)+" cm^-3") //initializing value of impurity concentration.
```

```
14 e=1.6*10^-19
15 disp("e = "+string(e)+" columb") //initializing
      value of charge of electron.
16 n=(-(Na-Nd)+sqrt((Na-Nd)^2+4*no))/2
17 disp("Electron concentration,n=(-(Na-Nd)+sqrt((Na-Nd
      )^2+4*no))/2)= "+string(n)+" cm^-3")//calculation
18 p=(no^2/n)
19 disp("Hole concentration,p=(no^2/n))= "+string(p)+"
      cm^-3")//calculation
20 Jdrift=n1*un*e*E
21 disp("Drift current density,Jdrift=n1*un*e*E)= "+
      string(Jdrift)+" A/cm^2")//calculation
```

Scilab code Exa 3.18 ELECTRON AND HOLE CONCENTRATION AND DRIFT CURRENT DENSITY

```
1 clc
2 Nd=0
3 disp("Nd = "+string(Nd)+" /cm^3") //initializing
      value of donor concentration.
4 Na=10^17
5 disp("Na= "+string(Na)+" /cm^3") //initializing
      value of acceptor concentration.
6 no=1.8*10^6
7 disp("no = "+string(no)+" /cm^3") //initializing
      value of electron and hole concentration per cm
      ^3.
8 E=10
9 disp("E = "+string(E)+" V/cm") ////initializing
      value of electric field.
10 un=200
11 disp("un = "+string(un)+" cm^2/s") ////initializing
      value of mobility.
12 p1=10^17
13 disp("p1= "+string(p1)+" cm^-3") //initializing
      value of impurity concentration.
```

```
14 e=1.6*10^-19
15 disp("e = "+string(e)+" columb") //initializing
      value of charge of electron.
16 p=-(-(Na-Nd)-sqrt((Na-Nd)^2+4*(no^2)))/2
17 disp("Electron concentration ,p=-(-(Na-Nd)-sqrt((Na-
      Nd)^2+4*(no^2)))/2= "+string(p)+" cm^-3")//
      calculation
18 n=(no^2/p)
19 disp("Hole concentration ,n=(no^2/p))= "+string(n)+"
      cm^-3")//calculation
20 Jdrift=p1*un*e*E
21 disp("Drift current density ,Jdrift=n1*un*e*E)= "+
      string(Jdrift)+" A/cm^2")//calculation
```

Scilab code Exa 3.19 VALUE OF THERMAL EQUILIBRIUM HOLE CONCENTRATION

```
1 clc
2 D=120
3 disp("D = "+string(D)+" A/cm^2") //initializing
      value of drift current density.
4 E=5
5 disp("E = "+string(E)+" V/cm") //initializing value
      of electric field.
6 e=1.6*10^-19
7 disp("e = "+string(e)+" columb") //initializing
      value of charge of electron.
8 p=(D/(450*e*E))
9 disp("thermal equilibrium value of hole
      concentration ,p=(D/(450*e*E)))= "+string(p)+" /
      cm^3")//calculation
```

Scilab code Exa 3.20 CURRENT THROUGH THE BAR

```
clc
Nd=5*10^16
disp("Nd = "+string(Nd)+" /cm^3") //initializing
   value of donor concentration.
A=50*10^-8
disp("A= "+string(A)+" cm^2") //initializing value
   of area.
l=0.2
disp("l = "+string(l)+" /cm") //initializing value
   of length.
E=10
disp("E = "+string(E)+" V") //initializing value of
   electric field.
up=1100
disp("un = "+string(up)+" cm^2/s") //initializing
   value of mobility.
p=5*10^16
disp("p= "+string(p)+" /cm^-3") //initializing value
    of impurity concentration.
e=1.6*10^-19
disp("e = "+string(e)+" columb") //initializing
   value of charge of electron.
I=(p*up*e*E*A)/l
disp("Current through the bar,I=(p*up*e*E*A)/l)= "+
   string(I)+"A")//calculation
```

Chapter 4

EXCESS CARRIER IN SEMICONDUCTOR

Scilab code Exa 4.2 DIELECTRIC RELAXATION TIME

```
1  clc
2  Nd=2*10^17
3  disp("Nd = "+string(Nd)+"cm^-3") //initializing
      value of donor concentration.
4  Er=11.9
5  disp("Er = "+string(Er)) //initializing value of
      relative dielectric constant.
6  e=1.6*10^-19
7  disp("e = "+string(e)+"columns") //initializing
      value of charge of electrons.
8  Eo=8.854*10^-14
9  disp("eo = "+string(Eo)) //initializing value of
      permittivity of free space.
10 un=1350
11 disp("un = "+string(un)+"cm2/Vs") //initializing
      value of mobility.
12 sigma=e*un*Nd
13 disp("conducitivity ,sigma=e*un*Nd)="+string(sigma)+"
      S/cm")//calculation
```

```
14 td=((Er*Eo)/sigma)
15 disp("Dielectric releaxation time,td=((Er*Eo)/sigma)
   )="+string(td)+"s")//calculation
```

Scilab code Exa 4.3 ELECTRON HOLE RECOMBINATION

```
1 clc
2 n=10^15
3 disp("n = "+string(n)+"cm^-3") //initializing value
   of concentration of electrons/cm^3.
4 no=10^10
5 disp("no = "+string(no)+"cm^-3") //initializing
   value of intrinsic concentration of electron.
6 t=10^-6
7 disp("t = "+string(t)+"s") //initializing value of
   carrier lifetime.
8 c=1*10^14
9 disp("Excess electron concentration = "+string(c)+"
   cm^-3") //initializing value of excess electrons
   concentration.
10 R=(c/t)
11 disp("electron hole recombination,R=(c/t))="+string(
   R)+" /cm^3s")//calculation
```

Scilab code Exa 4.4 GENERATION AND RECOMBINATION RATE OF EHPS AND MAJORITY CARRIER

```
1 clc
2 Nd=10^15
3 disp("Nd = "+string(Nd)+"cm^-3") //initializing
   value of donor concentration..
4 tn=10*10^-6
```

```
5 disp("minority carrier lifetime = "+string(tn)+"s")
    //initializing value of minority carrier lifetime
    .
6 no=1.5*10^10
7 disp("no = "+string(no)+"cm^-3") //initializing
    value of electron and hole concentration per cm
    ^3.
8 p=(no^2/Nd)
9 disp("excess carrier concentration,p=(no^2/Nd))="+
    string(p)+"/cm^3")//calculation
10 R=(p/tn)
11 disp("electron hole generation and recombination
    rate,R=(p/t))="+string(R)+"/cm^3s")//calculation
12 t=Nd/R
13 disp("majority carrier concentration,t=Nd/R)="+
    string(t)+"s")//calculation.
14
15
16 //the value of majority carrier concentration,t=Nd/R
    (after calculation),is provided wrong in the
    solution.
```

Scilab code Exa 4.5 QUASI FERMI LEVELS

```
1 clc
2 Nd=10^16
3 disp("Nd = "+string(Nd)+" cm^-3") //initializing
    value of donor concentration.
4 p=10^6
5 disp("p = "+string(p)+" cm^-3") //initializing value
    of minority hole concentration.
6 no=10^10
7 disp("no = "+string(no)+" cm^-3") //initializing
    value of electron and hole concentration per cm
    ^3..
```

```
n1=10^15
disp("n* = "+string(n1)+" cm^-3") //initializing value of excess electron carrier concentration( denoted by n*).
p1=10^15
disp("p* = "+string(p1)+" cm^-3") //initializing value of excess hole carrier concentration( denoted by p*).
KT=0.0259
disp("KT = "+string(KT)+" eV") //initializing value of multipication of temperature and bolzmann constant.
T=300
disp("T = "+string(T)+" K") //initializing value of temperature.
Ef_Efi=(log(Nd/no)*KT)
disp("Thermal equilibirium fermi level,(Ef_Efi)=(KT*log(n/no)))="+string(Ef_Efi)+" eV")//calculation.
Efn_Efi=log((Nd+n1)/no)*KT
disp("Quasi-fermi levels for n-type dopant,(Efn_Efi)=(KT*log((n+n*)/no))="+string(Efn_Efi)+" eV")//calculation.
Efi_Efp=log((Nd+p1)/no)*KT
disp("Quasi-fermi levels for p-type dopant,(Efi_Efp)=(KT*log((p+p*)/no))="+string(Efi_Efp)+" eV")//calculation.

//the answer for Efn_Efi,Efi_Efp is provided wrong in the book.
//In this question,Nd=(n(used in the formula)).
```

Scilab code Exa 4.6 ENERGY DIFFERENCE BETWEEN EFn AND EFi

```
clc
Nd=5*10^16
```

```
disp("Nd = "+string(Nd)+"cm^-3") //initializing value of donor ion concentration.
Na=0
disp("Na = "+string(Na)+"cm^-3") //initializing value of acceptor ion concentration.
no=1.5*10^10
disp("no = "+string(no)+"cm^-3") //initializing value of electron and hole concentration per cm ^3.
n1=5*10^14
disp("n* = "+string(n1)+"cm^-3") //initializing value of excess electron carrier concentration.
p1=5*10^14
disp("p* = "+string(p1)+"cm^-3") //initializing value of excess hole carrier concentration.
KT=0.0259
disp("KT = "+string(KT)) //initializing value of thermal voltage.
Ef_Efi=(KT*log(Nd/no))
disp("thermal equilibrium fermi level ,(Ef_Efi)=(KT*log(n/no)))="+string(Ef_Efi)+"eV")//calculation.
Efn_Efi=log((Nd+n1)/no)*KT
disp("Excess carrier concentration ,(Efn_Efi)=(KT*log((n+n*)/no))="+string(Efn_Efi)+"eV")//calculation.
Efi_Efp=log((Na+p1)/no)*KT
disp("(Ef_Efi)=(KT*log((p+p*)/no))="+string(Efi_Efp)+"eV")//calculation.
```

Chapter 5

PN JUNCTION DIODE

Scilab code Exa 5.5 FERMI LEVEL IN P AND N JUNCTION AND CONTACT POTENTIAL

```
1  clc
2  Na=10^17
3  disp("Na = "+string(Na)+"/cm^3") //initializing
      value of medium p doping concentration.
4  Nd=10^15
5  disp("Nd = "+string(Nd)+"/cm^3") //initializing
      value of light n doping .
6  no=1.5*10^10
7  disp("no = "+string(no)+"cm^-3") //initializing
      value of intrinsic carrier concentration.
8  e=1.6*10^-19
9  disp("e = "+string(e)+"columns") //initializing
      value of charge of electrons.
10 K=1.38*10^-23
11 disp("K = "+string(K)+"J/k") //initializing value of
      boltzmann constant.
12 T=300
13 disp("T = "+string(T)+"K") //initializing value of
      temperature.
14 Efi_Efp=((K*T/e)*log(Na/no))
15 disp("(a)fermi level in the P region,Efi_Efp=((KT/e)
```

```
    *log(Na/no)))="+string(Efi_Efp)+"eV")//
    calculation.
16 Efn_Efi=((K*T/e)*log(Nd/no))
17 disp("fermi level in the n region,Efn_Efi=((KT/e)*
    log(Nd/no)))="+string(Efn_Efi)+"eV")//calculation
    .
18 Efn_Efp=(Efi_Efp)+(Efn_Efi)
19 disp("(b)junction potential at room temperature,
    Efn_Efp=(Efi_Efp)+(Efn_Efi))="+string(Efn_Efp)+"
    eV")//calculation.
```

Scilab code Exa 5.6 VALUE OF JUNCTION POTENTIAL

```
1  clc
2  Na=10^17
3  disp("Na = "+string(Na)+"/cm^3") //initializing
    value of medium p doping concentration.
4  Nd=10^15
5  disp("Nd = "+string(Nd)+"/cm^3") //initializing
    value of light n doping .
6  no=1.5*10^10
7  disp("no = "+string(no)+"cm^-3") //initializing
    value of intrinsic carrier concentration.
8  e=1.6*10^-19
9  disp("e = "+string(e)+"columbs") //initializing
    value of charge of electrons.
10 K=1.38*10^-23
11 disp("K = "+string(K)+"J/k") //initializing value of
    boltzmann constant.
12 T=300
13 disp("T = "+string(T)+"K") //initializing value of
    temperature.
14 Vj=((K*T/e)*log((Na*Nd)/(no)^2))
15 disp("Junction potential,Vj=((K*T/e)*log((Na*Nd)/(no
    )^2))="+string(Vj)+"eV")//calculation.
```

Scilab code Exa 5.7 VALUE OF REVERSE AND FORWARD CURRENT

```
clc
Pp=10^18
disp("Pp = "+string(Pp)+"/cm^3") //initializing
   value of doping concentration in p region.
Nn=10^15
disp("Nn = "+string(Nn)+"/cm^3") //initializing
   value of doping concentration in n region.
tp=7*10^-6
disp("tp = "+string(tp)+"s") //initializing value of
    hole lifetime.
tn=0.2*10^-6
disp("tn = "+string(tn)+"s") //initializing value of
    electron lifetime.
up=800
disp("up = "+string(up)+"cm2/Vs") //initializing
   value of P side mobility.
un=300
disp("un = "+string(un)+"cm2/Vs") //initializing
   value of n side mobility.
no=1.5*10^10
disp("no = "+string(no)+"cm^-3") //initializing
   value of intrinsic concentration.
Vf=0.6
disp("Vf = "+string(Vf)+"V") //initializing value of
    forward bias voltage.
A=100*10^-6
disp("A = "+string(A)+"m^2") //initializing value of
    diode cross-sectional area.
e=1.6*10^-19
disp("e = "+string(e)+"columns") //initializing
   value of charge of electrons.
K=1.38*10^-23
```

```
23 disp("K = "+string(K)+"J/k") //initializing value of
     boltzmann constant.
24 T=300
25 disp("T = "+string(T)+"K") //initializing value of
     temperature.
26 Vt=(K*T/e)
27 disp("Vt=(K*T/e))="+string(Vt)+"eV")//calculation.
28 Dp=Vt*un
29 disp("Dp=Vt*un="+string(Dp)+"cm^-3")//calculation.
30 Dn=Vt*up
31 disp("Dn=Vt*up="+string(Dn)+"cm^-3")//calculation.
32 Lp=sqrt(Dp*tp)
33 disp("Lp=(sqrt(Dp*tp))="+string(Lp)+"cm")//
     calculation.
34 Ln=(sqrt(Dn*tn))
35 disp("Ln=(sqrt(Dn*tn))="+string(Ln)+"cm")//
     calculation.
36 npo=(no^2/Pp)
37 disp("npo=(no^2/Pp)="+string(npo)+"cm^-3")//
     calculation.
38 Ppo=(no^2/Nn)
39 disp("Ppo=(no^2/Nn)="+string(Ppo)+"cm^-3")//
     calculation.
40 Io=(((Dp*Ppo)/(Lp))+((Dn*npo)/(Ln)))*e*A
41 disp("Reverse saturation current ,Io=(((Dp*Ppo)/(Lp)
     )+((Dn*npo)/(Ln)))*e*A="+string(Io)+"A")//
     calculation.
42 If=Io*((exp(Vf/Vt))-1)
43 disp("Diode forward current ,If=Io*((exp(Vf/Vt))-1)="
     +string(If)+"A")//calculation.
44 //the value of Io(reverse saturation current ),after
     calculation is provided wrong in the book.Due to
     which If (diode forward current )also differ.
```

Scilab code Exa 5.8 VALUE OF BUILT IN POTENTIAL

```
clc
Na=4*10^16
disp("Na = "+string(Na)+"cm^-3") //initializing value of acceptor concentration.
Nd=2*10^19
disp("Nd = "+string(Nd)+"cm^-3") //initializing value of donor concentration.
no=1.5*10^10
disp("no = "+string(no)+"cm^-3") //initializing value of intrinsic carrier concentration.
e=1.6*10^-19
disp("e = "+string(e)+"columns") //initializing value of charge of electrons.
K=1.38*10^-23
disp("K = "+string(K)+"J/k") //initializing value of boltzmann constant.
T=300
disp("T = "+string(T)+"K") //initializing value of temperature.
Vbi=((K*T/e)*log((Na*Nd)/(no)^2))
disp("Built in potential potential,Vbi=((K*T/e)*log((Na*Nd)/(no)^2))="+string(Vbi)+"V")//calculation.

//The value used for Nd in the book for solution is different than provided in the question.
//I have used the value provided in the solution(i.e Nd=2*10^19)
```

Scilab code Exa 5.9 VALUE OF BUILT IN POTENTIAL

```
clc
Na=4*10^16
disp("Na = "+string(Na)+"cm^-3") //initializing value of acceptor concentration.
```

```
4 Nd=2*10^19
5 disp("Nd = "+string(Nd)+"cm^-3") //initializing
    value of donor concentration.
6 no=1.8*10^6
7 disp("no = "+string(no)+"cm^-3") //initializing
    value of intrinsic carrier concentration.
8 e=1.6*10^-19
9 disp("e = "+string(e)+"columns") //initializing
    value of charge of electrons.
10 K=1.38*10^-23
11 disp("K = "+string(K)+"J/k") //initializing value of
    boltzmann constant.
12 T=300
13 disp("T = "+string(T)+"K") //initializing value of
    temperature.
14 Vbi=((K*T/e)*log((Na*Nd)/(no)^2))
15 disp("Built in potential potential,Vbi=((K*T/e)*log
    ((Na*Nd)/(no)^2))="+string(Vbi)+"V")//calculation
    .
```

Scilab code Exa 5.10 VALUE OF DEPLETION WIDTH AND ACCEPTOR CONCENTRATION

```
1 clc
2 Na=10^16
3 disp("Na = "+string(Na)+"/cm^3") //initializing
    value of medium p doping concentration.
4 Nd=10^18
5 disp("Nd = "+string(Nd)+"/cm^3") //initializing
    value of light n doping .
6 Vbi=0.64
7 disp("Vbi = "+string(Vbi)+"V") //initializing value
    of built in voltage.
8 e=1.6*10^-19
9 disp("e = "+string(e)+"columns") //initializing
    value of charge of electrons.
```

```
10 Er=11.9
11 disp("Er = "+string(Er)) //initializing value of
      relative dielectric permittivity constant .
12 Eo=8.854*10^-14
13 disp("Eo = "+string(Eo)+" F/cm") //initializing
      value of permittivity of free space.
14 E=Eo*Er
15 disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
      )//calculation.
16 W=sqrt((2*E*Vbi/e)*((Nd+Na)/(Na*Nd)))
17 disp("W=sqrt((2*E*Vbi/e)*((Nd+Na)/(Na*Nd))))="+
      string(W)+" cm")//calculation.
18 xn=((W*Na)/(Nd+Na))
19 disp("xn=((W*Na)/(Nd+Na)))="+string(xn)+"cm")//
      calculation.
20 xp=((W*Nd)/(Nd+Na))
21 disp("xp=((W*Nd)/(Nd+Na)))="+string(xp)+"cm")//
      calculation.
22 Emax=(-e*Nd*xn)/E
23 disp("Emax=(-e*Nd*xn)/E)="+string(Emax)+"V/cm")//
      calculation.
24
25 //the value and unit of W(depletion width),provided
      after calculation in the book is wrong.Due to
      this xn,xp ,Emax also differ.
```

Scilab code Exa 5.12 VALUE OF MAXIMUM ELECTRIAL FIELD

```
1 clc
2 Na=10^16
3 disp("Na = "+string(Na)+"/cm^3") //initializing
      value of medium p doping concentration.
4 Nd=10^18
5 disp("Nd = "+string(Nd)+"/cm^3") //initializing
      value of light n doping .
```

```
Vbi=0.64
disp("Vbi = "+string(Vbi)+"V") //initializing value
    of built in voltage.
Vr=20
disp("Vr = "+string(Vr)+"V") //initializing value of
     applied reverse voltage.
e=1.6*10^-19
disp("e = "+string(e)+"columns") //initializing
    value of charge of electrons.
Er=11.9
disp("Er = "+string(Er)) //initializing value of
    relative dielectric permittivity constant .
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
    value of permittivity of free space.
E=Eo*Er
disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
    )//calculation.
Emax=-(sqrt(((2*e*(Vbi+Vr))/(E))*((Nd*Na)/(Na+Nd))))
disp("Emax=-(sqrt(((2*e*(Vbi+Vr))/(E))*((Nd*Na)/(Na+
    Nd)))))="+string(Emax)+"V/cm")//calculation.
```

Scilab code Exa 5.13 VALUE OF N TYPE DOPING CONCENTRATION

```
clc
Emax=2*10^5
disp("Emax = "+string(Emax)+"V/cm") //initializing
    value of maximum electric field.
Nd=10^18
disp("Nd = "+string(Nd)+"/cm^3") //initializing
    value of donor concentration .
Vbi=0.54
disp("Vbi = "+string(Vbi)+"V") //initializing value
    of built in voltage.
Vr=20
```

```
9 disp("Vr = "+string(Vr)+"V") //initializing value of
      applied reverse voltage.
10 e=1.6*10^-19
11 disp("e = "+string(e)+"columns") //initializing
     value of charge of electrons.
12 Er=11.9
13 disp("Er = "+string(Er)) //initializing value of
     relative dielectric permittivity constant .
14 Eo=8.854*10^-14
15 disp("Eo = "+string(Eo)+" F/cm") //initializing
     value of permittivity of free space.
16 E=Eo*Er
17 disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
     )//calculation.
18 Na=(Emax^2*E*Nd)/((2*e*(Vbi+Vr)*Nd)-(Emax^2*E))
19 disp("Na=(Emax^2*E*Nd)/((2*e*(Vbi+Vr)*Nd)-(Emax^2*E)
     ))="+string(Na)+"cm^-3")//calculation.
```

Scilab code Exa 5.14 VALUE OF JUNCTION CAPACITANCE

```
1 clc
2 Na=10^16
3 disp("Na = "+string(Na)+"/cm^3") //initializing
     value of acceptor concentration.
4 Nd=10^18
5 disp("Nd = "+string(Nd)+"/cm^3") //initializing
     value of donor concentration .
6 A=1
7 disp("A = "+string(A)+"cm^2") //initializing value
     of area for finding junction capacitance per unit
      area .
8 Vj=0.54
9 disp("Vj = "+string(Vj)+"V") //initializing value of
      built in voltage.
10 Va=10
```

```
11 disp("Va = "+string(Va)+"V") //initializing value of
       applied reverse voltage.
12 e=1.6*10^-19
13 disp("e = "+string(e)+" columns") //initializing
      value of charge of electrons.
14 Er=11.9
15 disp("Er = "+string(Er)) //initializing value of
      relative dielectric permittivity constant .
16 Eo=8.854*10^-14
17 disp("Eo = "+string(Eo)+" F/cm") //initializing
      value of permittivity of free space.
18 E=Eo*Er
19 disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
      )//calculation.
20 Cj=sqrt((e*E*A^2/(2*(Va+Vj)))*((Na*Nd)/(Na+Nd)))
21 disp("Cj=sqrt((e*E*A^2/(2*(Va+Vj))*((Na*Nd)/(Na+Nd))
      ))="+string(Cj)+"f/cm^2")//calculation.
```

Scilab code Exa 5.15 VALUE OF JUNCTION CAPACITANCE

```
1 clc
2 Na=10^15
3 disp("Na = "+string(Na)+" cm^-3") //initializing
      value of acceptor concentration.
4 Nd=10^18
5 disp("Nd = "+string(Nd)+" cm^-3") //initializing
      value of donor concentration.
6 no=1.8*10^6
7 disp("no = "+string(no)+" cm^-3") //initializing
      value of intrinsic carrier concentration.
8 e=1.6*10^-19
9 disp("e = "+string(e)+" columbs") //initializing
      value of charge of electrons.
10 K=1.38*10^-23
11 disp("K = "+string(K)+" J/k") //initializing value
```

```
    of boltzmann constant.
12 T=300
13 disp("T = "+string(T)+" K") //initializing value of
    temperature.
14 Vbi=((K*T/e)*log((Na*Nd)/(no)^2))
15 disp("Built in potential potential,Vbi=((K*T/e)*log
    ((Na*Nd)/(no)^2))="+string(Vbi)+" V")//
    calculation.
```

Scilab code Exa 5.16 VALUE OF INTRINSIC CONCENTRATION

```
1  clc
2  Na=10^18
3  disp("Na = "+string(Na)+" cm^-3") //initializing
    value of acceptor concentration.
4  Nd=10^18
5  disp("Nd = "+string(Nd)+" cm^-3") //initializing
    value of donor concentration.
6  Vbi=1.4
7  disp("Vbi = "+string(Vbi)+" V") //initializing value
    of built in voltage.
8  e=1.6*10^-19
9  disp("e = "+string(e)+" columbs") //initializing
    value of charge of electrons.
10 K=1.38*10^-23
11 disp("K = "+string(K)+" J/k") //initializing value
    of boltzmann constant.
12 T=300
13 disp("T = "+string(T)+" K") //initializing value of
    temperature.
14 Vt=0.0259
15 disp("Vt = "+string(Vt)+" eV") //initializing value
    of thermal voltage.
16 no=sqrt((Na*Nd)/(exp(Vbi/Vt)))
17 disp("no=sqrt((Na*Nd)/(exp(Vbi/Vt)))="+string(no)+"
```

```
cm^-3")//calculation.
```

Scilab code Exa 5.18 VALUE OF Vbi AND FERMI LEVEL AND Vbi FROM FERMI LEVEL

```
clc
Na=10^17
disp("Na = "+string(Na)+" cm^-3") //initializing
    value of acceptor concentration.
Nd=5*10^16
disp("Nd = "+string(Nd)+" cm^-3") //initializing
    value of donor concentration.
e=1.6*10^-19
disp("e = "+string(e)+" columbs") //initializing
    value of charge of electrons.
no=1.5*10^10
disp("no = "+string(no)+" cm^3") //initializing
    value of intrinsic carrier concentration.
T=300
disp("T = "+string(T)+" K") //initializing value of
    temperature.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value
    of thermal voltage.
Vbi=(Vt*(log(Na*Nd/(no^2))))
disp("(a)Vbi=(Vt*(log(Na*Nd/(no^2))))="+string(Vbi)+
    " V")//calculation.
Efi_Efp=(Vt*log(Na/(no)))
disp("(b)value of fermi level on each side of
    junction,Efi_Efp=(Vt*log(Na/(no)))="+string(
    Efi_Efp)+" V")//calculation.
Efn_Efi=(Vt*log(Nd/(no)))
disp("Efn_Efi=(Vt*log(Nd/(no)))="+string(Efn_Efi)+"
    V")//calculation.
disp("(c)The energy band digram is similar to Fig P5
    .3")
```

```
21 Vbi=((Efi_Efp)+(Efn_Efi))
22 disp("(d)Vbi=((Efi_Efp)+(Efn_Efi))/(e)=Vj="+string(
   Vbi)+" V")//calculation.
```

Scilab code Exa 5.19 VALUE OF Vbi AND FERMI LEVEL AND Vbi FROM FERMI LEVEL

```
1  clc
2  Na=5*10^17
3  disp("Na = "+string(Na)+"/cm^3") //initializing
      value of medium p doping concentration.
4  Nd=5*10^17
5  disp("Nd = "+string(Nd)+"/cm^3") //initializing
      value of light n doping .
6  no=1.5*10^10
7  disp("no = "+string(no)+"cm^-3") //initializing
      value of intrinsic concentration.
8  e=1.6*10^-19
9  disp("e = "+string(e)+"columbs") //initializing
      value of charge of electrons.
10 K=1.38*10^-23
11 disp("K = "+string(K)+"J/k") //initializing value of
     boltzmann constant.
12 T=300
13 disp("T = "+string(T)+"K") //initializing value of
      temperature.
14 Vbi=((K*T/e)*log((Na*Nd)/(no)^2))
15 disp("(a)Built in potential potential,Vbi=((K*T/e)*
      log((Na*Nd)/(no)^2))="+string(Vbi)+"eV")//
      calculation.
16 Efi_Efp=((K*T/e)*log(Na/no))
17 disp("(b)fermi level in the P region and N region,
      Efi_Efp=((KT/e)*log(Na/no)))="+string(Efi_Efp)+"
      eV")//calculation.
18 VBI=2*(Efi_Efp)
19 disp("(c)VBI from the fermi level,VBI=2*(Efi_Efp))="
```

```
+string(VBI)+"V")//calculation.
```

Scilab code Exa 5.20 VALUE OF IMPURITY CONCENTRATION AND VbI

```
clc
Nc=2.8*10^19
disp("Nc = "+string(Nc)+" /cm^3") //initializing
   value of number of electron in the conduction
   band.
Nv=1.04*10^19
disp("Nv = "+string(Nv)+" /cm^3") //initializing
   value of number of electron in the valence band..
no=1.5*10^10
disp("no = "+string(no)+" cm^-3") //initializing
   value of intrinsic carrier concentration.
e=1.6*10^-19
disp("e = "+string(e)+" columbs") //initializing
   value of charge of electrons.
K=8.62*10^-5
disp("K = "+string(K)+" J/k") //initializing value
   of boltzmann constant.
T=300
disp("T = "+string(T)+" K") //initializing value of
   temperature.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value
   of thermal voltage.
Ec_Ef=0.21
disp("Ec_Ef = "+string(Ec_Ef)+" eV") //initializing
   value of energy difference between conduction
   band and fermi level.
Ef_Ev=0.18
disp("Ef_Ev = "+string(Ef_Ev)+" eV") //initializing
   value of energy difference between fermi level
   and valence band.
```

```
Nd=(Nc/exp((Ec_Ef)/(K*T)))
disp("Nd=(Nc/exp((Ec-Ef)/(K*T)))="+string(Nd)+" cm^-3")//calculation.
Na=(Nv/exp((Ef_Ev)/(K*T)))
disp("Na=(Nv/exp((Ef-Ev)/(K*T)))="+string(Na)+" cm^-3")//calculation.
Vbi=(Vt*(log(Na*Nd/(no^2))))
disp("Built in potential potential,Vbi=(Vt*(log(Na*Nd/(no^2))))="+string(Vbi)+" V")//calculation.
```

Scilab code Exa 5.21 VALUE OF Na Xn Xp AND Emax

```
clc
Vbi=1.2
disp("Vbi = "+string(Vbi)+"/cm^3") //initializing value of built in voltage.
no=1.8*10^6
disp("no = "+string(no)+"cm^-3") //initializing value of intrinsic concentration.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value of thermal voltage.
Er=13.1
disp("Er = "+string(Er)) //initializing value of relative dielectric permittivity constant .
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing value of permittivity of free space.
e=1.6*10^-19
disp("e = "+string(e)+" columbs") //initializing value of charge of electrons.
E=Eo*Er
disp("total permittivity,E=Eo*Er="+string(E)+" F/cm")//calculation.
NaNd=((no^2)*(exp(Vbi/Vt)))
```

```
17 disp("(a)NaNd=((no^2)*(exp(Vbi/Vt)))="+string(NaNd)+
     " /cm^6")//calculation.
18 Na=(sqrt(NaNd/(4)))
19 disp("Na=(sqrt(NaNd/(4)))="+string(Na)+" /cm^3")//
     calculation.
20 Nd=4*Na
21 disp("(b)Nd=4*Na="+string(Nd)+" /cm^3")//calculation
     .
22 W=sqrt((2*E*Vbi/e)*((Nd+Na)/(Na*Nd)))
23 disp("(c)W=sqrt((2*E*Vbi/e)*((Nd+Na)/(Na*Nd))))="+
     string(W)+" cm")//calculation.
24 xn=0.2*W
25 disp("(d)xn=0.2*W="+string(xn)+" cm")//calculation.
26 xp=0.8*W
27 disp("xp=0.8*W="+string(xp)+" cm")//calculation.
28 Emax=(-e*Nd*xn)/E
29 disp("(e)Emax=(-e*Nd*xn)/E)="+string(Emax)+"V/cm")//
     calculation.
30 //The value of Na after calculation is provided
     wrong in the book.Due to which value of W,xn,xp
     and Emax differ ,than the answer provided in the
     book.
```

Scilab code Exa 5.22 VALUE OF TEMPERATURE

```
1 clc
2 Na=10^16
3 disp("Na = "+string(Na)+"cm^-3") //initializing
     value of acceptor concentration.
4 Nd=5*10^15
5 disp("Nd = "+string(Nd)+"cm^-3") //initializing
     value of donor concentration.
6 no=1.5*10^10
7 disp("no = "+string(no)+"cm^-3") //initializing
     value of intrinsic concentration.
```

```
8  Vbi=0.676
9  disp("Vbi = "+string(Vbi)+"V") //initializing value
       of built in voltage.
10 e=1.6*10^-19
11 disp("e = "+string(e)+"columns") //initializing
       value of charge of electrons.
12 K=1.38*10^-23
13 disp("K = "+string(K)+"J/k") //initializing value of
        boltzmann constant.
14 T=(Vbi*(e/K)*(1/(log((Na*Nd)/(no^2)))))
15 disp("T=(Vbi*(e/K)*(1/(log((Na*Nd)/(no^2)))))=" +
       string(T)+"K")//calculation.
```

Scilab code Exa 5.23 VALUE OF Vbi AND TEMPERATURE

```
1  clc
2  Na=5*10^17
3  disp("Na = "+string(Na)+"cm^-3") //initializing
       value of acceptor concentration.
4  Nd=10^17
5  disp("Nd = "+string(Nd)+"cm^-3") //initializing
       value of donor concentration.
6  no=1.5*10^10
7  disp("no = "+string(no)+"cm^-3") //initializing
       value of intrinsic concentration.
8  e=1.6*10^-19
9  disp("e = "+string(e)+"columns") //initializing
       value of charge of electrons.
10 K=1.38*10^-23
11 disp("K = "+string(K)+"J/k") //initializing value of
        boltzmann constant.
12 T=300
13 disp("T = "+string(T)+"K") //initializing value of
       temperature.
14 VBI=0.847
```

```
15 disp("VBI = "+string(VBI)+"V") //initializing value
   of VBI when VBI is reduced by 1%.
16 Vbi=((K*T/e)*log((Na*Nd)/(no)^2))
17 disp("(a)Built in potential potential,Vbi=((K*T/e)*
   log((Na*Nd)/(no)^2))="+string(Vbi)+" V")//
   calculation.
18 T=(e*VBI/K)*((log(Na*Nd/(no^2)))^-1)
19 disp("(b)T=(VBI*(e/K)*(1/(log((Na*Nd)/(no^2))))))="+
   string(T)+"K")//calculation.
20 //the answer for part (b) is not provided in the
   book.
```

Scilab code Exa 5.24 VALUE OF Vbi AND W AND Xn AND Xp AND Emax

```
1  clc
2  Na=4*10^12
3  disp("Na = "+string(Na)+" /cm^3") //initializing
   value of medium p doping concentration.
4  Nd=4*10^16
5  disp("Nd = "+string(Nd)+" /cm^3") //initializing
   value of light n doping.
6  no=1.5*10^10
7  disp("no = "+string(no)+" /cm^3") //initializing
   value of intrinsic carrier concentration.
8  K=1.38*10^-23
9  disp("K = "+string(K)+" J/k") //initializing value
   of boltzmann constant.
10 T=300
11 disp("T = "+string(T)+" K") //initializing value of
   temperature.
12 e=1.6*10^-19
13 disp("e = "+string(e)+" columbs") //initializing
   value of charge of electrons.
14 Er=11.9
15 disp("Er = "+string(Er)) //initializing value of
```

```
   relative dielectric permittivity constant .
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
   value of permittivity of free space.
E=Eo*Er
disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
   )//calculation.
Vbi=((K*T/e)*log((Na*Nd)/(no)^2))
disp("Built in potential potential ,Vbi=((K*T/e)*log
   ((Na*Nd)/(no)^2))="+string(Vbi)+" eV")//
   calculation.
W=sqrt((2*E*Vbi/e)*((Nd+Na)/(Na*Nd)))
disp("W=sqrt((2*E*Vbi/e)*((Nd+Na)/(Na*Nd))))="+
   string(W)+" cm")//calculation.
xn=((W*Na)/(Nd+Na))
disp("xn=((W*Na)/(Nd+Na)))="+string(xn)+" cm")//
   calculation.
xp=((W*Nd)/(Nd+Na))
disp("xp=((W*Nd)/(Nd+Na)))="+string(xp)+" cm")//
   calculation.
Emax=(e*Nd*xn)/E
disp("Emax=(e*Nd*xn)/E)="+string(Emax)+" V/cm")//
   calculation.
//the value of W(depletion width) ,after calculation
    is provided wrong in the book,due to this xn,xp
   ,Emax also differ.(also,the value of Nd+Na
   substitute in the
//formula for for xn,xp is wrong )
```

Scilab code Exa 5.25 MAGNITUDE OF APPLIED REVERSE BIAS

```
clc
Na=4*10^17
disp("Na = "+string(Na)+"/cm^3") //initializing
   value of donor concentration.
```

```
4  Nd=4*10^15
5  disp("Nd = "+string(Nd)+"/cm^3") //initializing
       value of light n doping.
6  no=1.5*10^10
7  disp("no = "+string(no)+"cm^-3") //initializing
       value of intrinsic concentration.
8  Emax=300*10^3
9  disp("Emax = "+string(Emax)+"/cm^3") //initializing
       value of maximum electric field.
10 K=1.38*10^-23
11 disp("K = "+string(K)+"J/k") //initializing value of
       boltzmann constant.
12 T=300
13 disp("T = "+string(T)+"K") //initializing value of
       temperature.
14 e=1.6*10^-19
15 disp("e = "+string(e)+"columns") //initializing
       value of charge of electrons.
16 Er=11.9
17 disp("Er = "+string(Er)) //initializing value of
       relative dielectric permittivity constant .
18 Eo=8.854*10^-14
19 disp("Eo = "+string(Eo)+" F/cm") //initializing
       value of permittivity of free space.
20 E=Eo*Er
21 disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
       )//calculation.
22 Vbi=((K*T/e)*log((Na*Nd)/(no)^2))
23 disp("Built in potential potential ,Vbi=((K*T/e)*log
       ((Na*Nd)/(no)^2))="+string(Vbi)+" V")//
       calculation.
24 xn=(E*Emax/(Nd*e))
25 disp("xn=(E*Emax)/(e*Nd))="+string(xn)+" cm")//
       calculation.
26 W=(xn*(Nd+Na)/Na)
27 disp("W=(xn(Nd+Na)/Na))="+string(W)+" cm")//
       calculation.
28 Vr=((W^2*e/(2*E))*((Na*Nd)/(Na+Nd)))-(Vbi)
```

```
29 disp("Vr=(W^2*e/(2*E))*((Na*Nd)/(Na+Nd))-(Vbi))="+
   string(Vr)+" V")//calculation.
```

Scilab code Exa 5.26 VALUE OF DOPING CONCENTRATION

```
1  clc
2  Na=5*10^15
3  disp("Na = "+string(Na)+"cm^-3") //initializing
      value of acceptor concentration.
4  Nd=10^18
5  disp("Nd = "+string(Nd)+"cm^-3") //initializing
      value of donor concentration.
6  e=1.6*10^-19
7  disp("e = "+string(e)+"columns") //initializing
      value of charge of electrons.
8  Vr=10
9  disp("Vr = "+string(Vr)+"V") //initializing value
      reverse voltage.
10 Er=11.9
11 disp("Er = "+string(Er)) //initializing value of
      relative dielectric permittivity constant .
12 Eo=8.854*10^-14
13 disp("Eo = "+string(Eo)+" F/cm") //initializing
      value of permittivity of free space.
14 E=Eo*Er
15 disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
   )//calculation.
16 Emax=10^6
17 disp("Emax = "+string(Emax)+"V/cm") //initializing
      value of maximum electric field.
18 W=(2*Vr/(Emax))
19 disp("W = "+string(W)+"cm") //calculation.
20 Nd=(Emax*E)/(W*e)
21 disp("Nd=(Emax*e)/(W*e))="+string(Nd)+"cm^-3")//
      calculation.
```

Scilab code Exa 5.27 VALUE OF JUNCTION CAPACITANCE

```
clc
Na=5*10^15
disp("Na = "+string(Na)+"cm^3") //initializing value
    of acceptor concentration.
Nd=10^18
disp("Nd = "+string(Nd)+"cm^3") //initializing value
    of donor concentration .
no=1.5*10^10
disp("no = "+string(no)+"cm^-3") //initializing
   value of intrinsic carrier concentration.
Vr1=0
disp("Vr1 = "+string(Vr1)+"V") //initializing value
   of built in voltage.
Vr2=5
disp("Vr2 = "+string(Vr2)+"V") //initializing value
   of applied reverse voltage.
A=3*10^-5
disp("A = "+string(A)+"cm^2") //initializing value
   of cross sectional area.
e=1.6*10^-19
disp("e = "+string(e)+"columns") //initializing
   value of charge of electrons.
Er=11.9
disp("Er = "+string(Er)) //initializing value of
   relative dielectric permittivity constant .
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
   value of permittivity of free space.
E=Eo*Er
disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
   )//calculation.
Vt=0.0259
```

```
23 disp("Vt="+string(Vt)+" V")//initializing the value
   of thermal voltage .
24 Vbi=((Vt)*log((Na*Nd)/(no)^2))
25 disp("Built in potential ,Vbi=(Vt*log((Na*Nd)/(no)
   ^2))="+string(Vbi)+" V")//calculation.
26 Cj1=sqrt((e*E*(A^2)/(2*(Vr1+Vbi)))*((Na*Nd)/(Na+Nd))
   )
27 disp("Cj1=sqrt((e*E*(A^2)/(2*(Vr1+Vbi))*((Na*Nd)/(Na
   +Nd))))="+string(Cj1)+" F")//calculation.
28 Cj2=sqrt((e*E*(A^2)/(2*(Vr2+Vbi)))*((Na*Nd)/(Na+Nd))
   )
29 disp("Cj2=sqrt((e*E*(A^2)/(2*(Vr2+Vbi))*((Na*Nd)/(Na
   +Nd))))="+string(Cj2)+" F")//calculation.
30 //the value of Vr2 use for calculating answer of Cj2
    is different than provided in question.
31 //I have used the value provided in the solution (i.
   e.Vr2=5)
```

Scilab code Exa 5.28 VALUE OF REVERSE SATURATION CURRENT

```
1 clc
2 Na=1*10^16
3 disp("Na = "+string(Na)+"cm^3") //initializing value
    of acceptor concentration.
4 Nd=5*10^16
5 disp("Nd = "+string(Nd)+"cm^3") //initializing value
    of donor concentration .
6 no=1.5*10^10
7 disp("no = "+string(no)+"cm^-3") //initializing
   value of intrinsic concentration.
8 Dn=25
9 disp("Dn = "+string(Dn)+"cm^2/sec") //initializing
   value of diffusion cofficient on the P side.
10 Dp=10
11 disp("Dp = "+string(Dp)+"cm^2/sec") //initializing
```

```
    value of diffusion cofficient on the N side.
tp=5*10^-7
disp("tn = "+string(tp)+"s") //initializing value of
    hole lifetime.
tn=5*10^-7
disp("tp = "+string(tn)+"s") //initializing value of
    electron lifetime.
e=1.6*10^-19
disp("e = "+string(e)+"columns") //initializing
    value of charge of electrons.
Pno=(no^2/Nd)
disp("Pno=(no^2/Nd))="+string(Pno)+"cm^-3")//
    calculation.
Npo=(no^2/Na)
disp("Npo=(no^2/Na))="+string(Npo)+"cm^-3")//
    calculation.
Lp=(sqrt(Dp*tp))
disp("Lp=(sqrt(Dp*tp)))="+string(Lp)+"cm")//
    calculation.
Ln=(sqrt(Dn*tn))
disp("Ln=(sqrt(Dn*tn)))="+string(Ln)+"cm")//
    calculation.
Jo=((e*((Dp*Pno/(Lp))+(Dn*Npo)/(Ln))))
disp("Jo=((e*((Dp*Pno/(Lp))+(Dn*Npo)/(Ln))))="+
    string(Jo)+" A/cm^2")//calculation.
```

Scilab code Exa 5.29 VALUE OF REVERSE SATURATION CURRENT

```
clc
Na=10^15
disp("Na = "+string(Na)+"cm^3") //initializing value
    of acceptor concentration.
Nd=10^15
disp("Nd = "+string(Nd)+"cm^3") //initializing value
    of donor concentration .
```

```
no=1.5*10^10
disp("no = "+string(no)+"cm^-3") //initializing
   value of intrinsic carrier concentration.
Dn=50
disp("Dn = "+string(Dn)+"cm^2/sec") //initializing
   value of built in voltage.
Dp=20
disp("Dp = "+string(Dp)+"cm^2/sec") //initializing
   value of applied reverse voltage.
tp=5*10^-7
disp("tn = "+string(tp)+"s") //initializing value of
    hole lifetime.
tn=5*10^-7
disp("tp = "+string(tn)+"s") //initializing value of
    electrons lifetime.
e=1.6*10^-19
disp("e = "+string(e)+"columns") //initializing
   value of charge of electrons.
Pno=(no^2/Nd)
disp("Pno=(no^2/Nd))="+string(Pno)+"cm^-3")//
   calculation.
Npo=(no^2/Na)
disp("Npo=(no^2/Na))="+string(Npo)+"cm^-3")//
   calculation.
Lp=(sqrt(Dp*tp))
disp("Lp=(sqrt(Dp*tp)))="+string(Lp)+"cm")//
   calculation.
Ln=(sqrt(Dn*tn))
disp("Ln=(sqrt(Dn*tn)))="+string(Ln)+"cm")//
   calculation.
Jo=((e*((Dp*Pno/(Lp))+(Dn*Npo)/(Ln))))
disp("Jo=((e*((Dp*Pno/(Lp))+(Dn*Npo)/(Ln))))="+
   string(Jo)+"A/cm^2")//calculation.
//the value of tp,tn provided in the question,is
   different than that provided in the solution.
//I have used the value ,provided in the solution(i.
   e.tp=tn=5*10^7)
```

Scilab code Exa 5.30 VALUE OF THE CHANGE IN THE APPLIED VOLTAGE

```
clc
Eg=-1.1
disp("Eg = "+string(Eg)+"V") //initializing value of
    energy gap.
Vf1=0.6
disp("Vf1 = "+string(Vf1)+"V") //initializing value
   of forward voltage for case 1.
T1=300
disp("T1 = "+string(T1)+"K") //initializing value of
    temperature for case 1.
T2=310
disp("T2 = "+string(T2)+"K") //initializing value of
    temperature for case 2 .
Vf2=(((Eg+Vf1)*T2)/(T1))-Eg
disp("Forward voltage for case 2,Vf2=((Eg+Vf1)*T2)/(
   T1)+Eg)="+string(Vf2)+" V")//calculation.
```

Scilab code Exa 5.31 RELATIVE CHANGE IN Io

```
clc
T=300
disp("T = "+string(T)+" K") //initializing value of
   temperature .
Eg=0.7*1.6*10^-19
disp("Eg = "+string(Eg)+" V") //initializing value
   of Band Gap energy of the material.
K=1.38*10^-23
disp("K = "+string(K)+" J/K") //initializing value
   of boltzmann constant.
X=((Eg/(K*T^2)))+(3/T)
```

```
9  disp("Relative change in Io ,... 1/Io*(dIo/dT)= "+
       string(X))//calculation
10 //in percent form
11 disp("Relative change in Io ,... 1/Io*(dIo/dT)= "+
       string(X*100)+" %")//calculation
12
13 //... taking, (1/Io)*(dIo/dT) as 'X'
14
15 //This is the solved example of chapter 5 (Ex 5.1)On
       page 128
```

Chapter 6

ELECTRICAL BREAKDOWN IN PN JUNCTIONE

Scilab code Exa 6.2 VALUE OF Eg Ev Ec and Vbi

```
1  clc
2  X1=4.13
3  disp("X1 = "+string(X1)+" eV") //initializing value
      of eldelta_Ectron effinity of germanium.
4  X2=4.07
5  disp("X2 = "+string(X2)+" eV") //initializing value
      of electron effinity of gallium arsenide.
6  Eg1=0.7
7  disp("Eg1 = "+string(Eg1)+" eV") //initializing
      value of energy gap of germanium.
8  Eg2=1.43
9  disp("Eg2 = "+string(Eg2)+" F/cm") //initializing
      value of energy gap of gallium arsenide..
10 Nv1=6*10^18
11 disp("Nv1 = "+string(Nv1)+" cm^-3") //initializing
      value of density of states in Valence band,Nv for
       germanium.
12 Nv2=7*10^18
13 disp("Nv2 = "+string(Nv2)+" cm^-3") //initializing
```

```
    value of density of states in Valence band,Nv for
     galliminum arsenide.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value
    of thermal voltage... Vt = K*T/e
e=1.6*10^-19
disp("e = "+string(e)+" columbs") //initializing
    value of electronic charge.
no=2.5*10^13
disp("no = "+string(no)+" cm^-3") //initializing
    value of intrinsic carrier concentration.
Pp=10^17
disp("Pp = "+string(Pp)+" cm^-3") //initializing
    value of hole concentration on the depletion edge
     of the N region.
Nd=10^17
disp("Nd = "+string(Nd)+" cm^-3") //initializing
    value of number of donor ions (which is equal to
    hole concentration on the depletion edge of the N
     region).
np=(no^2)/Pp
disp("np="+string(np)+" cm^-3")//calculation
delta_Eg=(Eg2-Eg1)
disp("delta_Eg=(Eg2-Eg1)="+string(delta_Eg)+" eV")//
    calculation
delta_Ec=(X1-X2)
disp("delta_Ec=(X1-X2)="+string(delta_Ec)+" eV")//
    calculation
delta_Ev=(delta_Eg-delta_Ec)
disp("delta_Ev=(delta_Eg-delta_Ec)="+string(delta_Ev
    )+" eV")//calculation
Vbi=((delta_Ev*1.6*10^-19)/(e))+((Vt*log((Nv1*Nd)/(
    np*Nv2))))
disp("Vbi=((delta_Ev*1.6*10^-19)/(e))+((Vt*log((Nv1*
    Nd)/(np*Nv2))))="+string(Vbi)+" V")//calculation
```

Scilab code Exa 6.4 VALUE OF DONOR CONCENTRATION

```
clc
Nc=2.8*10^19
disp("Nc = "+string(Nc)+" cm^-3") //initializing value of effective density of state in the conduction band.
k=-4*10^15
disp("k = "+string(k)+" cm^4F^-2V^-1") //initializing value of slope of the (1/C^2) versus V curve.
Er=11.9
disp("Er = "+string(Er)) //initializing value of relative dielectric permittivity constant .
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing value of dielectric constant of free space.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing value of charge of electrons.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value of thermal voltage.
VBI=0.3
disp("VBI = "+string(VBI)+" V") //initializing value of built in voltage.
E=Eo*Er
disp("total permittivity ,E=Eo*Er ="+string(E)+" F/cm")//calculation
Nd=((-2)/(e*E)*(1/k))
disp("Nd=((-2)/(e*E)*(1/k)))="+string(Nd)+" cm^-3") //calculation
Vn=(Vt*(log(Nc/Nd)))
disp("Vn=(Vt*(log(Nc/Nd)))="+string(Vn)+" V")//
```

```
    calculation
22 VBn=(VBI+Vn)
23 disp("VBn=(VBI+Vn)="+string(VBn)+" V")//calculation
24
25
26 // taking ,... d(1/C^2)/dV as k,... for
    simlification
```

Scilab code Exa 6.5 VALUE OF THE BARRIER HEIGHT AND BUILT IN POTENTIAL

```
1  clc
2  Nd=2*10^17
3  disp("Nd = "+string(Nd)+"/cm^-3") //initializing
     value of donor concentration.
4  Nc=2.8*10^19
5  disp("Nc = "+string(Nc)+"/cm^-3") //initializing
     value of effective density of state in the
     conduction band.
6  Js=40*10^-6
7  disp("Js = "+string(Js)+"A/cm^2") //initializing
     value of saturation current density.
8  T=300
9  disp("T = "+string(T)+"K") //initializing value of
     absolute temperature.
10 R=110
11 disp("R = "+string(R)+" A/(K-cm^2)") //initializing
     value of richardson's constant.
12 Vt=0.0259
13 disp("Vt = "+string(Vt)+" eV") //initializing value
     of thermal voltage.
14 VBn=(Vt*(log(R*T^2/Js)))
15 disp("VBn = "+string(VBn)+" V") // calculation.
16 Vn=(Vt*(log(Nc/Nd)))
17 disp("Vn = "+string(Vn)+" V") // calculation.
18 VBI=(VBn-Vn)
```

```
19 disp("VBI=(VBn-Vn))="+string(VBI)+" V")//calculation
20
21 //The value of Vn (after calculation ) is provided
     wrong in the book,due to which VBI also differ.
```

Scilab code Exa 6.6 VALUE OF DEPLETION WIDTH DIFFUSION LENGTH AND SATURATION HOLE

```
1  clc
2  Nd=2*10^17
3  disp("Nd = "+string(Nd)+" /cm^-3") //initializing
     value of donor concentration.
4  Dp=30
5  disp("Dp = "+string(Dp)+" cm^2/s") //initializing
     value of diffusion cofficient.
6  Nc=2.8*10^19
7  disp("Nc = "+string(Nc)+" /cm^-3") //initializing
     value of effective density of state in the
     conduction band.
8  Js=40*10^-6
9  disp("Js = "+string(Js)+ "A/cm^2") //initializing
     value of saturation current density.
10 no=1.5*10^10
11 disp("no = "+string(no)+" cm^-3") //initializing
     value of intrinsic concentration of electrons.
12 tp=10^-6
13 disp("tp = "+string(tp)+" s") //initializing value
     of hole life-time.
14 T=300
15 disp("T = "+string(T)+" K") //initializing value of
     absolute temperature.
16 R=110
17 disp("R = "+string(R)+" A/(K-cm^2)") //initializing
     value of richardson's constant.
18 Vt=0.0259
19 disp("Vt = "+string(Vt)+" eV") //initializing value
```

```
    of thermal voltage.
e=1.6*10^-19
disp("e = "+string(e)+" columbs") //initializing
    value of charge of electrons.
Er=11.9
disp("Er = "+string(Er)) //initializing value of
    relative dielectric permittivity constant .
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
    value of dielectric constant of free space.
E=Eo*Er
disp("total permittivity ,E=Eo*Er)="+string(E)+" F/cm
    ")//calculation
VBn=(Vt*(log(R*T^2/Js)))
disp("VBn = "+string(VBn)+" V") // calculation.
Vn=(Vt*(log(Nc/Nd)))
disp("Vn = "+string(Vn)+" V") // calculation.
VBI=(VBn-Vn)
disp("VBI=(VBn-Vn))="+string(VBI)+" V")//calculation
W=(sqrt((E*VBI)/(e*Nd)))
disp("current density in a metal semiconductor
    junction ,W = "+string(W)+" A") // calculation.
Lp=(sqrt(Dp*tp))
disp("Diffusion length ,Lp=(sqrt(Dp*tp)) = "+string(
    Lp)+" cm") // calculation.
Jpo=(e*Dp*no^2)/(Lp*Nd)
disp("saturation hole current density ,Jpo=(e*Dp*no
    ^2)/(Lp*Nd)) = "+string(Jpo)+" A/cm^2") //
    calculation.


//The value of Vn (after calculation ) is provided
    wrong in the book,due to which VBI differ and due
     to VBI ,current density in a metal semiconductor
     junction (W)gets changed.
//The value of Jpo (saturation hole current density)
     ,after calculation is also provided wrong in the
     book.
```

Scilab code Exa 6.8 MAXIMUM N TYPE DOPING CONCENTRATION

```
clc
Er=11.9
disp("Er = "+string(Er)) //initializing value of
   relative dielectric permittivity constant.
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
   value of permittivity of free space.
VBD=20
disp("VBD = "+string(VBD)+" V") //initializing value
    of  break down voltage.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing
   value of charge of electrons.
E=Eo*Er
disp("total permittivity ,E=Eo*Er)="+string(E)+" F/cm
   ")//calculation
Emax=5*10^5
disp("Emax = "+string(Emax)+" V/cm") //initializing
   value of maximum critical electric field.
ND=(Eo*Er*(Emax^2))/(2*e*VBD)
disp("ND=(Eo*Er*(Emax^2))/(2*e*VBD)="+string(ND)+"
   cm^-3")//calculation


//the formula given in the solution for VBD is
   somewhat written wrong.The correct formula is (
   VBD=(E*Emax^2/2*e*ND)).
```

Scilab code Exa 6.9 VALUE OF REVERSE BREAK DOWN VOLTAGE

```
clc
Er=11.9
disp("Er = "+string(Er)) //initializing value of
   relative dielectric permittivity constant.
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
   value of permittivity of free space.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing
   value of charge of electrons.
no=1.5*10^10
disp("no = "+string(no)+"cm^-3") //initializing
   value of intrinsic concentration of electrons.
Nd=1*10^16
disp("Nd="+string(Nd)+" cm^-3")//initializing the
   value of donor concentration.
Emax=2*10^5
disp("Emax = "+string(Emax)+" V/cm") //initializing
   value of maximum critical electric field.
Na=1*10^16
disp("Na="+string(Na)+" cm^-3")//initializing the
   value of acceptor concentration.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value
   of thermal voltage.
E=Eo*Er
disp("total permittivity ,E=Eo*Er)="+string(E)+" F/cm
   ")//calculation
VBI=(Vt*(log(Na*Nd/no^2)))
disp("VBI=(Vt*(log(Na*Nd/no^2))) = "+string(VBI)+" V
   ") // calculation.
V=(E*Emax^2)/(e*Nd)
disp("breakdown voltage for symetrical abrupt
   junction ,VBD+VBI=(E*Emax^2)/(e*Nd))="+string(V)+"
    V")//calculation
VBD=V-VBI
disp("VBD=V-VBI)="+string(VBD)+" V")//calculation
```

Scilab code Exa 6.10 VALUE OF REVERSE BREAK DOWN VOLTAGE

```
clc
Er=11.9
disp("Er = "+string(Er)) //initializing value of
    relative dielectric permittivity constant.
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
    value of permittivity of free space.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing
    value of charge of electrons.
no=1.5*10^10
disp("no = "+string(no)+"cm^-3") //initializing
    value of intrinsic concentration of electrons.
Emax=10^6
disp("Emax = "+string(Emax)+" V/cm") //initializing
    value of maximum critical electric field..
Nd=1*10^18
disp("Nd="+string(Nd)+" cm^-3")//initializing the
    value of donor concentration.
Na=1*10^18
disp("Na="+string(Na)+" cm^-3")//initializing the
    value of acceptor concentration.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value
    of thermal voltage.
VBI=(Vt*(log(Na*Nd/no^2)))
disp("VBI=(Vt*(log(Na*Nd/no^2))) = "+string(VBI)+" V
    ") // calculation.
E=Eo*Er
disp("total permittivity ,E=Eo*Er)="+string(E)+" F/cm
    ")//calculation
V=(E*Emax^2)/(e*Nd)
```

```
23 disp("breakdown voltage for symetrical abrupt
     junction ,VBD+VBI=(E*Emax^2)/(e*Nd))="+string(V)+"
     V")//calculation
24 VBD=V-VBI
25 disp("VBD=V-VBI)="+string(VBD)+" V")//calculation
```

Scilab code Exa 6.11 VALUE OF REVERSE BREAK DOWN VOLTAGE

```
1  clc
2  Nd=1*10^18
3  disp("Nd = "+string(Nd)+" cm^-3") //initializing
     value of donor concentration.
4  Na=-1*10^18
5  disp("Na = "+string(Na)+" cm^3") //initializing
     value of acceptor concentration.
6  Er=11.9
7  disp("Er = "+string(Er)) //initializing value of
     relative dielectric permittivity constant .
8  Eo=8.854*10^-14
9  disp("Eo = "+string(Eo)+" F/cm") //initializing
     value of dielectric constant of free space.
10 e=1.6*10^-19
11 disp("e = "+string(e)+" columns") //initializing
     value of charge of electrons.
12 Vt=0.0259
13 disp("Vt = "+string(Vt)+" eV") //initializing value
     of thermal voltage.
14 Vbd=15
15 disp("Vbd = "+string(Vbd)+" eV") //initializing
     value of break down voltage.
16 W=2*10^-4
17 disp("W = "+string(W)+" cm") //initializing value of
     the distance over which doping profile varies.
18 E=Eo*Er
19 disp("total permittivity ,E=Eo*Er)="+string(E)+" F/cm
```

```
    ")//calculation
20 a=((Nd-Na)/(W))
21 disp("slope of doping profile curve,a=((Nd-Na)/(W))=
    "+string(a)+" cm^-4")//calculation
22 Emax=(((Vbd)^2)*9*e*a/(32*E))^(1/3)
23 disp("Emax=(((Vbd)^2)*9*e*a/(32*E))^(1/3)="+string(
    Emax)+" V/cm")//calculation
```

Scilab code Exa 6.12 VALUE OF THE BARRIER HEIGHT AND BUILT IN POTENTIAL AND DEPLET

```
1  clc
2  Ew=4.55
3  disp("Ew = "+string(Ew)+" V") //initializing value
    of work function of tungusten.
4  X=4.01
5  disp("X = "+string(X)+"V") //initializing value of
    electron effinity of silicon.
6  Er=11.9
7  disp("Er = "+string(Er)) //initializing value of
    relative dielectric permittivity constant.
8  Eo=8.854*10^-14
9  disp("Eo = "+string(Eo)+" F/cm") //initializing
    value of permittivity of free space.
10 e=1.6*10^-19
11 disp("e = "+string(e)+" columns") //initializing
    value of charge of electrons.
12 Nc=2.8*10^19
13 disp("Nc = "+string(Nc)+"/cm^-3") //initializing
    value of effective density of state in the
    conduction band.
14 Nd=10^17
15 disp("Nd = "+string(Nd)+"/cm^-3") //initializing
    value of donor concentration.
16 Vt=0.0259
17 disp("Vt = "+string(Vt)+" eV") //initializing value
```

```
    of thermal voltage.
18 VB=(Ew-X)
19 disp("Barrier height,VB=(Ew-X) = "+string(VB)+" V")
    // calculation.
20 Ec_Ef=(Vt*log(Nc/Nd))
21 disp("Ec_Ef=(Vt*log(Nc/Nd))="+string(Ec_Ef)+" V")//
    calculation
22 VBI=(VB-(Ec_Ef))
23 disp("VBI=(VB-(Ec_Ef))="+string(VBI)+" V")//
    calculation
24 xn=sqrt(2*Eo*Er*VBI/(e*Nd))
25 disp("Depletion width,xn=sqrt(2*Eo*Er*VBI/(e*Nd))="+
    string(xn)+" cm")//calculation
26 Emax=(e*Nd*xn/(Eo*Er))
27 disp("maximum electric field,Emax=(e*Nd*xn/(Eo*Er))=
    "+string(Emax)+" V/cm")//calculation
```

Scilab code Exa 6.13 VALUE OF MAXIMUM ELECTRIAL FIELD AND JUNCTION CAPACITANCE PER

```
1 clc
2 Ew=4.5
3 disp("Ew = "+string(Ew)+" V") //initializing value
    of work function of tungusten.
4 X=4.01
5 disp("X = "+string(X)+"V") //initializing value of
    electron effinity of silicon.
6 Er=12
7 disp("Er = "+string(Er)) //initializing value of
    relative dielectric permittivity constant.
8 Eo=8.854*10^-14
9 disp("Eo = "+string(Eo)+" F/cm") //initializing
    value of permittivity of free space.
10 Vr=3
11 disp("Vr = "+string(Vr)+" V") //initializing value
    of  reverse voltage.
```

```
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing value of charge of electrons.
Nc=2.8*10^19
disp("Nc = "+string(Nc)+"/cm^-3") //initializing value of effective density of state in the conduction band.
Nd=10^17
disp("Nd = "+string(Nd)+"/cm^-3") //initializing value of donor concentration.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value of thermal voltage.
VB=(Ew-X)
disp("barrier height,VB=(Ew-X) = "+string(VB)+" V") // calculation.
Ec_Ef=(Vt*log(Nc/Nd))
disp("Ec_Ef=(Vt*log(Nc/Nd))="+string(Ec_Ef)+" V")//calculation
VBI=(VB-(Ec_Ef))
disp("VBI=(VB-(Ec_Ef))="+string(VBI)+" V")//calculation
xn=sqrt((2*Eo*Er*(VBI+Vr))/(e*Nd))
disp("Depletion width,xn=sqrt(2*Eo*Er*(VBI+Vr)/(e*Nd))="+string(xn)+" cm")//calculation
Emax=(e*Nd*xn/(Eo*Er))
disp("maximum electric field,Emax=(e*Nd*xn/(Eo*Er))="+string(Emax)+" V/cm")//calculation
C=sqrt((e*Eo*Er*Nd)/(2*(VBI+Vr)))
disp("Capitance per unit area,C=sqrt((e*Eo*Er*Nd)/(2*(VBI+Vr)))="+string(C)+" F/cm^2")//calculation
//the Value of reverse voltage(Vr) provided in the question is different than used in the solution.I have used the value provided in the solution(i.e Vr=3).
//the value of C (Capitance per unit area) after calculation is provided wrong in the book.
```

Scilab code Exa 6.14 VALUE OF VB Xn AND Emax

```
clc
Ew=4.28
disp("Ew = "+string(Ew)+" V") //initializing value
    of work function of tungusten.
X=4.01
disp("X = "+string(X)+"V") //initializing value of
    electron effinity of silicon.
Er=11.9
disp("Er = "+string(Er)) //initializing value of
    relative dielectric permittivity constant.
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
    value of permittivity of free space.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing
    value of charge of electrons.
Nc=2.8*10^19
disp("Nc = "+string(Nc)+"/cm^-3") //initializing
    value of effective density of state in the
    conduction band.
Nd=10^15
disp("Nd = "+string(Nd)+"/cm^-3") //initializing
    value of donor concentration.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value
    of thermal voltage.
VB=(Ew-X)
disp("barrier height ,VB=(Ew-X) = "+string(VB)+" V")
    // calculation.
Ec_Ef=(Vt*log(Nc/Nd))
disp("Ec_Ef=(Vt*log(Nc/Nd))="+string(Ec_Ef)+" V")//
    calculation
```

```
22 VBI=(VB-(Ec_Ef))
23 disp("VBI=(VB-(Ec_Ef))="+string(VBI)+" V")//
     calculation
24 xn=sqrt(2*Eo*Er*VBI/(e*Nd))
25 disp("Depletion width,xn=sqrt(2*Eo*Er*VBI/(e*Nd))="+
     string(xn)+" cm")//calculation
26 Emax=(e*Nd*xn/(Eo*Er))
27 disp("maximum electric field,Emax=(e*Nd*xn/(Eo*Er))=
     "+string(Emax)+" V/cm")//calculation
28
29 //the Value of donor concentration (Nd) provided in
     the question is different than used in the
     solution.I have used the value provided in the
     question(i.e Nd=10^15). ,i.e answer differs than
     provided in the book.
```

Scilab code Exa 6.15 CAPACITANCE OF A GOLD SILICON JUNCTION

```
1 clc
2 Ew=5.1
3 disp("Ew = "+string(Ew)+" V") //initializing value
     of work function of tungusten.
4 X=4.01
5 disp("X = "+string(X)+"V") //initializing value of
     electron effinity of silicon.
6 Er=11.9
7 disp("Er = "+string(Er)) //initializing value of
     relative dielectric permittivity constant.
8 Eo=8.854*10^-14
9 disp("Eo = "+string(Eo)+" F/cm") //initializing
     value of permittivity of free space.
10 e=1.6*10^-19
11 disp("e = "+string(e)+" columns") //initializing
     value of charge of electrons.
12 Nc=2.8*10^19
```

```
13 disp("Nc = "+string(Nc)+"/cm^-3") //initializing
     value of effective density of state in the
     conduction band.
14 Nd=5*10^15
15 disp("Nd = "+string(Nd)+"/cm^-3") //initializing
     value of donor concentration.
16 Vt=0.0259
17 disp("Vt = "+string(Vt)+" eV") //initializing value
     of thermal voltage.
18 Vr=5
19 disp("Vr = "+string(Vr)+" V") //initializing value
     of reverse voltage.
20 A=1*10^-4
21 disp("A = "+string(A)+" cm^2") //initializing value
     of area of the gold silicon junction diode..
22 VB=(Ew-X)
23 disp("barrier height,VB=(Ew-X) = "+string(VB)+" V")
     // calculation.
24 Ec_Ef=(Vt*log(Nc/Nd))
25 disp("Ec_Ef=(Vt*log(Nc/Nd))="+string(Ec_Ef)+" V")//
     calculation
26 VBI=(VB-(Ec_Ef))
27 disp("VBI=(VB-(Ec_Ef))="+string(VBI)+" V")//
     calculation
28 C1=sqrt((e*Eo*Er*Nd)/(2*(VBI+Vr)))
29 disp("Capitance per unit area,C1=sqrt((e*Eo*Er*Nd)
     /(2*(VBI+Vr)))="+string(C1)+" F/cm^2")//
     calculation
30 C=C1*A
31 disp("total junction capatiance,C=C1*A="+string(C)+"
     F")//calculation
```

Scilab code Exa 6.17 VALUE OF LOWERING OF BARRIER HEIGHT

```
1 clc
```

```
Er=13.1
disp("Er = "+string(Er)) //initializing value of
   relative dielectric permittivity constant.
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
   value of permittivity of free space.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing
   value of charge of electrons.
Emax=30*10^3
disp("Emax = "+string(Emax)+" V/cm") //initializing
   value of maximum critical electric field..
E=Eo*Er
disp("total permittivity ,E=Eo*Er)="+string(E)+" F/cm
   ")//calculation
V=sqrt(e*Emax/(4*%pi*E))
disp("lowering of the barrier height ,V=sqrt(e*Emax
   /(4*%pi*E))="+string(V)+" V")//calculation
Xmax=sqrt(e/(16*%pi*E*Emax))
disp("position of the maximum barrier height ,Xmax=
   sqrt(e/(16*%pi*E*Emax))="+string(Xmax)+" cm")//
   calculation
```

Scilab code Exa 6.18 VALUE OF IDEAL REVERSE SATURATION CURRENT AND THE DIODE CURRE

```
clc
A=10^-4
disp("A = "+string(A)+" cm^-2") //initializing value
    of cross sectional area.
VBn=0.55
disp("VBn = "+string(VBn)+"V") //initializing value
   of barrier height.
T=300
disp("T = "+string(T)+"K") //initializing value of
   absolute temperature.
```

```
R=110
disp("R = "+string(R)+" A/(K-cm^2)") //initializing value of richardson's constant.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value of thermal voltage.
V=0.25
disp("V = "+string(V)+" V") //initializing value of forward bias voltage.
Io=A*R*T^2*exp(-VBn/Vt)
disp("reverse saturation current,Io=A*R*T^2*exp(-VBn/Vt) = "+string(Io)+" A") // calculation.
I=Io*((exp(V/Vt))-1)
disp("diode current,I=Io(exp(V/Vt)-1)="+string(I)+" A")//calculation
```

Scilab code Exa 6.19 VALUE OF FORWARD VOLTAGE

```
clc
Io1=10^-9
disp("Io1 = "+string(Io1)+" A") //initializing value of reverse saturation current of silicon SBD.
Io2=10^-14
disp("Io2 = "+string(Io2)+"A") //initializing value of reverse saturation current of a PN junction.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value of thermal voltage.
I=100*10^-6
disp("I = "+string(I)+" A") //initializing value of required current.
VfSBD=Vt*((log(I/Io1+1)))
disp("forward Voltage for silicon SBD,VfSBD=Vt*((log(I/Io1+1)))= "+string(VfSBD)+" V") // calculation.
```

```
12 VfPN=Vt*((log(I/Io2+1)))
13 disp("forward Voltage for silicon SBD,VfPN=Vt*((log(
   I/Io2+1)))="+string(VfPN)+" V")//calculation
```

Scilab code Exa 6.20 VALUE OF REVERSE SATURATION CURRENT

```
1  clc
2  Io1=10*10^-7
3  disp("Io1 = "+string(Io1)+" A") //initializing value
       of reverse saturation current of silicon SBD.
4  Io2=10*10^-7
5  disp("Io2 = "+string(Io2)+"A") //initializing value
   of reverse saturation current of a PN junction.
6  Vt=0.0259
7  disp("Vt = "+string(Vt)+" eV") //initializing value
   of thermal voltage.
8  I=1*10^-3
9  disp("I = "+string(I)+" A") //initializing value of
   forward current.
10 V=0.25
11 disp("V = "+string(V)+" V") //initializing value of
   difference in the forward voltage of the two
   diode.
12 VfSBD=Vt*((log(I/Io1+1)))
13 disp("forward Voltage for silicon SBD,VfSBD=Vt*((log
   (I/Io1+1)))= "+string(VfSBD)+" V") // calculation
   .
14 VfPN=(V+VfSBD)
15 disp("forward volage applied across the PN Diode,
   VfPN=(V+VfSBD)="+string(VfPN)+" V")//calculation
16 Io=(I/((exp(VfPN/Vt))-1))
17 disp("reverse saturation current of the PN junction
   Diode,Io=(I/((exp(VfPN/Vt))-1))="+string(Io)+" A"
   )//calculation
```

Chapter 7

BIPOLAR JUNCTION TRANSISTORB

Scilab code Exa 7.1 MAGNITUDE OF Io AND COLLECTOR CURRENT

```
1  clc
2  Dnb=20
3  disp("Dnb = "+string(Dnb)+" cm^2/s") //
       initializiation the value of one of base
       parametre of NPN transistor.
4  nB=10^4
5  disp("nB = "+string(nB)+" /cm^3") //initializiation
       the value of one of base parametre of NPN
       transistor.
6  xB=1*10^-6
7  disp("xB = "+string(xB)+" m") //initializiation the
       value of one of base parametre of NPN transistor.
8  AB=10^-4
9  disp("AB = "+string(AB)+" cm^2") //initializiation
       the value of one of base parametre of NPN
       transistor.
10 e=1.6*10^-19
11 disp("e = "+string(e)+" columns") //initializiation
       the value of electronic charge .
```

```
12 Vbe=0.5
13 disp("Vbe = "+string(Vbe)+" V") //initializiation
      the value of base emitter voltage of NPN
      transistor..
14 VT=0.0259
15 disp("VT = "+string(VT)+" V") //initializiation the
      value of threshold voltage.
16 WB=10^-4
17 disp("WB = "+string(WB)+" cm") //initializiation the
       value of base width of NPN transistor.
18 Io=((e*AB*Dnb*nB)/(WB))
19 disp("Magnitude of Io, Io=((e*AB*Dnb*nB)/(WB)))="+
      string(Io)+" A")//calculation
20 Ic=Io*(exp(Vbe/VT)-1)
21 disp("Collector current, Ic=Io((exp(Vbe/VT))-1))="+
      string(Ic)+" A")//calculation
```

Scilab code Exa 7.2 CONCENTRATION OF nEO pBO AND nCO

```
1 clc
2 NE=5*10^17
3 disp("NE = "+string(NE)+" /cm^3") //initializiation
      the value of doping concentration in the emitter
      .
4 NB=10^16
5 disp("NB = "+string(NB)+" /cm^3") //initializiation
      the value of doping concentration in the base.
6 NC=10^15
7 disp("NC = "+string(NC)+" /cm^3") //initializiation
      the value of doping concentration in the
      collector.
8 WB=0.8*10^-4
9 disp("WB = "+string(WB)+" cm") //initializiation the
       value of base width of NPN transistor.
10 no=1.5*10^10
```

```
11 disp("no = "+string(no)+"cm^-3") //initializing the
     intrinsic carrier concentration.
12 pEO=(no^2/NE)
13 disp("Number of Majority holes in the emitter,pEO=(
     no^2/NE) )="+string(pEO)+" /cm^3")//calculation
14 nBO=(no^2/NB)
15 disp("Number of Majority holes in the base,nBO=(no
     ^2/NB) )="+string(nBO)+" /cm^3")//calculation
16 pCO=(no^2/NC)
17 disp("Number of Majority holes in the collector,pCO
     =(no^2/NC) )="+string(pCO)+" /cm^3")//calculation
```

Scilab code Exa 7.3 VALUE OF pBO AND nB

```
1  clc
2  NE=5*10^17
3  disp("NE = "+string(NE)+" /cm^3") //initializiation
     of doping concentration in the emitter.
4  NB=10^16
5  disp("NB = "+string(NB)+" /cm^3") //initializiation
     of doping concentration in the base.
6  NC=10^15
7  disp("NC = "+string(NC)+" /cm^3") //initializiation
     of doping concentration in the collector.
8  WB=0.8*10^-4
9  disp("WB = "+string(WB)+" cm") //initializiation
     the value of base width of NPN transistor.
10 no=1.5*10^10
11 disp("no = "+string(no)+"cm^-3") //initializing the
     value of  intrinsic carrier concentration.
12 VT=0.0259
13 disp("VT = "+string(VT)+" V") //initializiation the
     value of threshold voltage.
14 VJ=0.6258
15 disp("VJ=Vbe = "+string(VJ)+" V") //initializiation
```

```
    the value of base emitter voltage.
16 pEO=(no^2/NE)
17 disp("Number of Majority holes in the emitter,pEO=(
    no^2/NE) )="+string(pEO)+" /cm^3")//calculation
18 nBO=(no^2/NB)
19 disp("Number of Majority holes in the base,nBO=(no
    ^2/NB))="+string(nBO)+" /cm^3")//calculation
20 pCO=(no^2/NC)
21 disp("Number of Majority holes in the collector,pCO
    =(no^2/NC))="+string(pCO)+" /cm^3")//calculation
22 pE=pEO*(exp(VJ/VT))
23 disp("pE(O)=pEO*(exp(VJ/VT)))="+string(pE)+" /cm^3")
    //calculation
24 nB=nBO*(exp(VJ/VT))
25 disp("nB=(nBO*(exp(VJ/VT))))="+string(nB)+" /cm^3")
    //calculation
26
27
28 //the answer provided in the book for pE,nB is some
    what different than actual calculated.
```

Scilab code Exa 7.5 VALUE OF BASE WIDTH

```
1 clc
2 Db=10
3 disp("Db = "+string(Db)+" cm^2/s") //initializiation
    the value of one of parametere of the transistor
    .
4 Bt=0.95
5 disp("Bt = "+string(Bt)) //initializiation the value
    of base transport factor of the transistor.
6 tb=10^-7
7 disp("tb = "+string(tb)+" s") //initializiation the
    value of one of parametere of the transistor.
8 Lp=(sqrt(Db*tb))
```

```
9  disp("Lp=(sqrt(Db*tb)))="+string(Lp)+" cm")//
       calculation
10 WB=(Lp*(acosh(1/Bt)))
11 disp("WB=(Lp*(acosh(1/Bt)))="+string(WB)+" cm")//
       calculation
```

Scilab code Exa 7.7 VALUE OF DELTA

```
1  clc
2  Jro=10^-9
3  disp("Jro = "+string(Jro)+" A/cm^2") //
       initializiation the value of recombination
       current density.
4  Jo=10^-12
5  disp("Jo = "+string(Jo)+" A/cm^2") //initializiation
       the value of reverse saturation current density.
6  Vbe=0.5
7  disp("Vbe = "+string(Vbe)+" V") //initializiation
       the value of base emitter voltage.
8  VT=0.0259
9  disp("VT = "+string(VT)+" V") //initializiation the
       value of threshold voltage.
10 delta=(1+((Jro/Jo)*(exp((-Vbe)/(2*VT)))))^-1
11 disp("delta (recombination factor)=(1+((Jro/Jo)*(exp
       ((-Vbe)/(2*VT)))))^-1)="+string(delta))//
       calculation.
```

Scilab code Exa 7.8 COMMON BASE AND COMMON EMITTER CURRENT GAIN

```
1  clc
2  NE=1*10^17
```

```
disp("NE = "+string(NE)+" /cm^3") //initializiation
   the value of doping concentration of emitter in
   the NPN transistor.
NB=10^15
disp("NB = "+string(NB)+" /cm^3") //initializiation
   the value of doping concentration of base in the
   NPN transistor.
WE=0.6*10^-4
disp("WE = "+string(WE)+" cm") //initializiation the
    value of one of parametre of the transistor.
WB=0.8*10^-4
disp("WB = "+string(WB)+" cm") //initializiation the
    value of one of parametre of the transistor.
no=1.5*10^10
disp("no = "+string(no)+"cm^-3") //initializing the
   value of intrinsic carrier concentration.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializiation
   the value of electronic charge
DE=15
disp("DE = "+string(DE)+" cm^2/s") //initializiation
    the value of one of parametere of the transistor
   .
DB=20
disp("DB = "+string(DB)+" cm^2/s") //initializiation
    the value of one of parametere of the transistor
   .
tE=0.2*10^-6
disp("tE = "+string(tE)+" s") //initializiation the
   value of one of parametere of the transistor.
tB=0.1*10^-6
disp("tB = "+string(tB)+" s") //initializiation the
   value of one of parametere of the transistor.
Vbe=0.60
disp("Vbe = "+string(Vbe)+" V") //initializiation
   the value of base emitter voltage .
VT=0.0259
disp("VT = "+string(VT)+" V") //initializiation the
```

```
    value of threshold voltage.
Jro=2*10^-8
disp("Jro = "+string(Jro)+" A/cm^2") //
    initializiation the value of recombination
    current density.
LE=(sqrt(DE*tE))
disp("LE=(sqrt(DE*tE)))="+string(LE)+" cm")//
    calculation
LB=(sqrt(DB*tB))
disp("LB=(sqrt(DB*tB)))="+string(LB)+" cm")//
    calculation
pEO=(no^2/NE)
disp("Number of Majority holes in the emitter,pEO=(
    no^2/NE) )="+string(pEO)+" /cm^3")//calculation
nBO=(no^2/NB)
disp("Number of Majority holes in the base,nBO=(no
    ^2/NB))="+string(nBO)+" /cm^3")//calculation
Y=(1+(((NB*DE*LB)/(NE*DB*LE))*((tanh(WB/LB)/tanh(WE/
    LE)))))^(-1)
disp("Emitter injection efficiency,Y=(1+((NB*DE*LB)
    /(NE*DB*LE)*(tanh(WB/LB)/tanh(WE/LE)))) )="+
    string(Y))//calculation
Bt=(cosh(WB/LB))^-1
disp("Base transport factor,Bt=(cosh(WB/LB))^-1)="+
    string(Bt))//calculation
Jo=((e*DB*nBO)/(LB*tanh(WB/LB)))
disp("Reverse saturation current Density,Jro=((e*DB*
    nBO)/(LB*tanh(WB/LB))))="+string(Jo)+"A/cm^2")//
    calculation
delta=(1+((Jro/Jo)*(exp((-Vbe)/(2*VT)))))^-1
disp("delta(recombination factor)=(1+((Jro/Jo)*(exp
    ((-Vbe)/(2*VT)))))^-1)="+string(delta)+" A")//
    calculation
a=Bt*delta*Y
disp("common base current amplification factor,(
    alpha=Bt*delta*Y)="+string(a))//calculation
B=(a/(1-a))
disp("common emitter current amplification factor,
```

```
    Beta=(a/(1-a)))="+string(B))//calculation
48 //the value of NE provided in the question is
    different than used in the solution .
49 //I have used the value (while solving) provided in
    the question (i.e NE=10^17/cm^3).
```

Scilab code Exa 7.9 CHANGE IN THE NEUTRAL BASE WIDTH

```
1  clc
2  NB=5*10^16
3  disp("NB = "+string(NB)+" /cm^3") //initializiation
    the doping concentration in the base.
4  NC=2*10^15
5  disp("NC = "+string(NC)+" /cm^3") //initializiation
    the doping concentration in the collector.
6  WBm=0.6*10^-4
7  disp("WBm = "+string(WBm)+" cm") //initializiation
    the value of actual base width.
8  e=1.6*10^-19
9  disp("e = "+string(e)+" columns") //initializiation
    the value of electronic charge.
10 VCB1=1
11 disp("VCB1 = "+string(VCB1)+" V") //initializiation
    the initial value of collector base voltage .
12 VCB2=4
13 disp("VCB2 = "+string(VCB2)+" V") //initializiation
    the final value of collector base voltage.
14 Er=11.9
15 disp("Er = "+string(Er)) //initializing value of
    relative dielectric permittivity constant .
16 Eo=8.854*10^-14
17 disp("Eo = "+string(Eo)+" F/cm") //initializing
    value of permittivity of free space.
18 no=1.5*10^10
19 disp("no = "+string(no)+"cm^-3") //initializing  the
```

```
    value of intrinsic charge carriers
VT=0.0259
disp("VT = "+string(VT)+" V") //initializiation the
    value of threshold voltage.
VBI=VT*(log((NB*NC)/no^2))
disp(" VBI=VT*(log((NB*NC)/no^2))="+string(VBI)+" V"
    )//calculation
WBS1=((2*Eo*Er*(VBI+VCB1)/e)*(NC/NB)*(1/(NC+NB)))
    ^(1/2)
disp("WBS=((2*Eo*Er*(VBI+VCB1)/e)*(NC/NB)*(1/(NC+NB)
    ))^(1/2))="+string(WBS1)+" cm")//calculation
Wb1=WBm-WBS1
disp("Neutral base width for VCB1,WB(neutral)=WBm-
    WBS1="+string(Wb1)+" cm")//calculation
WBS2=((2*Eo*Er*(VBI+VCB2)/e)*(NC/NB)*(1/(NC+NB)))
    ^(1/2)
disp("WBS=((2*Eo*Er*(VBI+VCB2)/e)*(NC/NB)*(1/(NC+NB)
    ))^(1/2))="+string(WBS2)+" cm")//calculation
Wb2=WBm-WBS2
disp("Neutral base width for VCB2,WB(neutral)=WBm-
    WBS2="+string(Wb2)+" cm")//calculation
deltaWbneutral=Wb1-Wb2
disp("change in the neutal base width ,deltaWb(
    neutral)=Wb1-Wb2="+string(deltaWbneutral)+" cm")
    //calculation
```

Scilab code Exa 7.10 CHANGE IN THE COLLECTOR CURRENT

```
clc
ro=500*10^3
disp("ro = "+string(ro)+" ohm") //initializiation
    the value of output resistance .
Vce1=7
disp("Vce1 = "+string(Vce1)+" V") //initializiation
    the initial value of collector emitter voltage .
```

```
Vce2=1
disp("Vce2 = "+string(Vce2)+" V") //initializiation
   the final value of collector emitter voltage .
Vce=6
disp("change in the collector-emitter voltage,Vce1-
   Vce2 = "+string(Vce)+" V") //calculation.
Ic=(Vce/ro)
disp("change in the collector current,Ic=(Vce/ro))="
   +string(Ic)+" A")//calculation
```

Chapter 8

THE FIELD EFFECT TRANSISTOR

Scilab code Exa 8.1 CAPACITANCE Cox AND Co

```
clc
Nd=10^16
disp("Nd = "+string(Nd)+" /cm^3") //initializing
   value of donor ion  concentration.
Er=3.9
disp("Er = "+string(Er)) //initializing value of
   relative dielectric permittivity constant .
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
   value of permittivity of free space.
W=0.5*10^-4
disp("W = "+string(W)+" cm") //initializing value of
    width of p-substrate.
L=10^-4
disp("L = "+string(L)+" cm") //initializing value of
    length of p-substrate.
tox=400*10^-8
disp("tox = "+string(tox)+" cm") //initializing
   value of thickness of p-substrate.
```

```
14 E=Eo*Er
15 disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
   )//calculation
16 Cox=(E*W*L)/tox
17 disp("Oxide capacitance ,Cox=(E*W*L)/tox)="+string(
   Cox)+" F")//calculation
18 Co=(Cox/(W*L))
19 disp("Capacitance per unit area ,Co=(Cox/(W*L)))="+
   string(Co)+" F/cm^2")//calculation
```

Scilab code Exa 8.2 MAXIMUM SPACE CHARGE WIDTH

```
1  clc
2  Na=10^17
3  disp("Na = "+string(Na)+" /cm^3") //initializing
   value of acceptor ion concentration.
4  Vt=0.0259
5  disp("Vt = "+string(Vt)+"V") //initializing value of
    thermal voltage.
6  e=1.6*10^-19
7  disp("e = "+string(e)+" columns") //initializing
   value of charge of electrons.
8  ni=1.5*10^10
9  disp("ni = "+string(ni)+"/cm^3") //initializing
   value of intrinsic carrier concentration.
10 Er=11.9
11 disp("Er = "+string(Er)) //initializing value of
   relative dielectric permittivity constant
12 Eo=8.854*10^-14
13 disp("Eo = "+string(Eo)+" F/cm") //initializing
   value of permittivity of free space.
14 Vs=Vt*log(Na/ni)
15 disp("Vs=Vt*log(Na/ni))="+string(Vs)+" V")//
   calculation
16 E=Eo*Er
```

```
17 disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
   )//calculation
18 Wd=sqrt(4*E*Vs/(e*Na))
19 disp("maximum depletion width ,Wd(max)=Sqrt(4*E*Vs/(e
   *Na)))="+string(Wd)+" cm")//calculation
```

Scilab code Exa 8.3 MAXIMUM SPACE CHARGE WIDTH

```
1  clc
2  Nd=3*10^18
3  disp("Nd = "+string(Nd)+" /cm^3") //initializing
      value of acceptor ion  concentration.
4  Vt=0.0259
5  disp("Vt = "+string(Vt)+"V") //initializing value of
       thermal voltage.
6  e=1.6*10^-19
7  disp("e = "+string(e)+" columns") //initializing
      value of charge of electrons.
8  ni=1.5*10^10
9  disp("ni = "+string(ni)+"/cm^3") //initializing
      value of intrinsic carrier concentration.
10 Er=11.9
11 disp("Er = "+string(Er)) //initializing value of
      relative dielectric permittivity constant
12 Eo=8.854*10^-14
13 disp("Eo = "+string(Eo)+" F/cm") //initializing
      value of permittivity of free space.
14 Vs=Vt*log(Nd/ni)
15 disp("Vs=Vt*log(Nd/ni))="+string(Vs)+" V")//
      calculation
16 E=Eo*Er
17 disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
   )//calculation
18 Wd=sqrt(4*E*Vs/(e*Nd))
19 disp("maximum depletion width ,Wd(max)=Sqrt(4*E*Vs/(e
```

```
*Nd)))="+string(Wd)+" cm")//calculation
```

Scilab code Exa 8.5 METAL SEMICONDUCTOR WORK FUNCTION DIFFERENCE

```
clc
Vm=3.2
disp("Vm = "+string(Vm)+" V") //initializing value
   of modified metal work function.
X=3.25
disp("X = "+string(X)+" V") //initializing value of
   modified electron affinity.
Nd=2*10^16
disp("Nd = "+string(Nd)+" /cm^3") //initializing
   value of donor concentration.
ni=1.5*10^10
disp("ni = "+string(ni)+" V") //initializing value
   of intrinsic carrier concentration.
Vt=0.0259
disp("Vt = "+string(Vt)+"V") //initializing value of
    thermal voltage.
Eg=1.12
disp("Eg = "+string(Eg)+"V") //initializing value of
    energy gap.
Vfp=(Vt*log(Nd/ni))
disp("Vfp=(Vt*log(Nd/ni))="+string(Vfp)+" V")//
   calculation.
Vms=-(Vm+(Eg/2)+Vfp-Vm)
disp("Vms=-(Vm+(Eg/2)+Vfp-Vm)="+string(Vms)+" V")//
   calculation.
```

Scilab code Exa 8.7 CAPACITANCE Co AND FLAT BAND VOLTAGE

```
clc
```

```
2  Nd=10^16
3  disp("Nd = "+string(Nd)+" /cm^3") //initializing
       value of donor ion  concentration.
4  Vms=-1.12
5  disp("Vms = "+string(Vms)+" V") //initializing value
       of metal semiconductor work function difference.
6  Er=3.9
7  disp("Er = "+string(Er)) //initializing value of
       relative dielectric permittivity constant .
8  Eo=8.854*10^-14
9  disp("Eo = "+string(Eo)+" F/cm") //initializing
       value of permittivity of free space.
10 tox=200*10^-8
11 disp("tox = "+string(tox)+" cm") //initializing
       value of thickness of p-type substrate.
12 Qss=2.5*10^-8
13 disp("Qss = "+string(Qss)+" columbs/cm^2") //
       initializing value of charge density on
       semiconductor surface.
14 Eox=Eo*Er
15 disp("Total permittivity ,Eox=Eo*Er="+string(Eox)+" F
       /cm")//calculation
16 Co=(Eox/tox)
17 disp("Capacitance per unit area ,Co=(E/tox))="+string
       (Co)+" F/cm^2")//calculation
18 Vfb=(Vms-(Qss/Co))
19 disp("Flat band voltage ,Vfb=(Vms-(Qss/Co)))="+string
       (Vfb)+" V")//calculation
20
21 //the answer for Co after calculation is provided
       wrong in the book.
```

Scilab code Exa 8.9 THRESHOLD VOLTAGE

```
1  clc
```

```
Na=3*10^16
disp("Na = "+string(Na)+" /cm^3") //initializing
   value of acceptor ion  concentration.
Vms=-1.12
disp("Vms = "+string(Vms)+"V") //initializing value
   of metal semiconductor work function difference.
Er=11.9
disp("Er = "+string(Er)) //initializing value of
   relative dielectric permittivity constant .
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
   value of permittivity of free space.
ni=1.5*10^10
disp("ni = "+string(ni)+"cm^-3") //initializing
   value of intrinsic concentration of electrons.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing
   value of charge of electrons.
tox=300*10^-8
disp("tox = "+string(tox)+" cm") //initializing
   value of thickness of p-type substrate.
Vfb=-1.12
disp("Vfb = "+string(Vfb)+" V") //initializing value
    of flat band voltage.
Qss=10^11
disp("Qss = "+string(Qss)+" electronic charge
   columns/cm^2") //initializing value of charge
   density on semiconductor surface.
Vt=0.0259
disp("Vt = "+string(Vt)+" eV") //initializing value
   of thermal voltage.
er=3.9
disp("er = "+string(er)) //initializing value of
   relative dielectric permittivity constant
Eox=Eo*Er
disp("total permittivity ,Eox=Eo*Er="+string(Eox)+" F
   /cm")//calculation
Vfp=Vt*(log(Na/(ni)))
```

```
27 disp("Potential,Vfp=Vt*(log(Na/(ni))))="+string(Vfp)
   +" V")//calculation
28 Wd=sqrt((4*Eox*Vfp)/(e*Na))
29 disp("Maximum depletion width,Wd=sqrt((4*E*Vs)/(e*Nd
   )))="+string(Wd)+" cm")//calculation
30 QDmax=(e*Na*Wd)
31 disp("Over all maximum depletion width,QDmax=(e*Na*
   Wd))="+string(QDmax)+" columns/cm^2")//
   calculation
32 VT=(((QDmax-1.6*10^-8)*tox)/(er*Eo))+(2*Vfp+Vfb)
33 disp("Thresold Voltage,VT=(((QDmax-1.6*10^-8)*tox)/(
   er*Eo))+(2*Vfp+Vfb)="+string(VT)+" V")//
   calculation
```

Scilab code Exa 8.10 VALUE OF Id

```
1  clc
2  L=1.25*10^-4
3  disp("L = "+string(L)+" cm") //initializing value of
       length of channel.
4  un=600
5  disp("un = "+string(un)+"cm^2/V-s") //initializing
   value of mobility of n-channel MOS transistor.
6  Co=6.9*10^-9
7  disp("Co = "+string(Co)+"F/cm^2") //initializing
   value of capacitance per unit area .
8  VT=0.60
9  disp("VT = "+string(VT)+" V") //initializing value
   of threshold Voltage.
10 Vgs=4
11 disp("Vgs = "+string(Vgs)+" V") //initializing value
       of gate to source voltage.
12 W=12*10^-4
13 disp("W = "+string(W)+"cm") //initializing value of
   width of channel.
```

```
14 Id=((Co*un*W)/(L)*((Vgs-VT)^2/(2)))
15 disp("Drain current,Id=((Co*un*W)/(L)*((Vgs-VT)
   ^2/(2)))="+string(Id)+" A")//calculation.
16
17
18 //The answer provided in the book (for Id) is wrong
   as the value of mobility used for solution is
   different than provided in the question and also
   value of (Vgs-Vt) is put wrong in the solution
   than given in the book.
19 //I have used the value given in the question i.e.
   answer differ.
```

Scilab code Exa 8.13 MINIMUM CAPACITANCE AND FLAT BAND CAPACITANCE

```
1 clc
2 Na=2*10^17
3 disp("Na = "+string(Na)+" /cm^3") //initializing
   value of acceptor ion concentration.
4 Er=11.9
5 disp("Er = "+string(Er)) //initializing value of
   relative dielectric permittivity constant .
6 Eo=8.854*10^-14
7 disp("Eo = "+string(Eo)+" F/cm") //initializing
   value of permittivity of free space.
8 ni=1.5*10^10
9 disp("ni = "+string(ni)+"cm^-3") //initializing
   value of intrinsic concentration of electrons.
10 e=1.6*10^-19
11 disp("e = "+string(e)+" columns") //initializing
   value of charge of electrons.
12 tox=400*10^-8
13 disp("tox = "+string(tox)+" cm") //initializing
   value of thickness of p-type substrate.
14 Vt=0.0259
```

```
15 disp("Vt = "+string(Vt)+" eV") //initializing value
     of thermal voltage.
16 er=3.9
17 disp("er = "+string(er)) //initializing value of
     relative dielectric permittivity constant
18 Vfp=Vt*(log(Na/(ni)))
19 disp("Potential,Vfp=Vt*(log(Na/(ni)))="+string(Vfp)
     +" V")//calculation
20 Wd=sqrt((4*Er*Eo*Vfp)/(e*Na))
21 disp("Depletion width,Wd=sqrt((4*Er*Eo*Vs)/(e*Nd)))=
     "+string(Wd)+" cm")//calculation
22 CTmin=(er*Eo/(((er/Er)*(Wd))+(tox)))
23 disp("Minimum Capacitance,CTmin=(er*Eo/((er/Er)*(Wd)
     +(tox)))="+string(CTmin)+" F/cm^2")//calculation
24 CFB=((er*Eo)/((((er/Er)*sqrt(Vt*Er*Eo/(e*Na))))+(tox
     )))
25 disp("Flat band capacitance,CFB=((er*Eo)/(((er/Er)*
     sqrt(Vt*Er*Eo/(e*Na))))+(tox))="+string(CFB)+" F/
     cm^2")//calculation
26
27 //the value of Na (acceptor ion concentration) and
     tox (thickness of p-type substrate) is provided
     different in the question than used in the
     solution.
28 //I have used the value provided in the solution.(i.
     e Na=2*10^17 and tox=400*10^8cm)
```

Scilab code Exa 8.14 VALUE OF Qss

```
1 clc
2 Vfb=-1.0
3 disp("Vfb = "+string(Vfb)+" V") //initializing value
     of flat band voltage.
4 Vms=-0.9
5 disp("Vms = "+string(Vms)+"V") //initializing value
```

```
    of metal semiconductor work function difference.
tox=200*10^-8
disp("tox = "+string(tox)+" cm") //initializing
    value of gate oxide thickness.
et=3.9
disp("et = "+string(et)) //initializing value of
    relative permittivity.
eo=8.85*10^-14
disp("eo = "+string(eo)+"F/cm") //initializing value
     of free space permittivity.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing
    value of charge of electrons.
eox=(eo*et)
disp("eox=(eo*et))="+string(eox)+" F/cm^2")//
    calculation
Cox=(eox/tox)
disp("Oxide capacitance,Cox=(eox/tox))="+string(Cox)
    +" F/cm^2")//calculation
Qss=((Vms-Vfb)*Cox)
disp("charge density on semiconductor surface,Qss=((
    Vms-Vfb)*Cox))="+string(Qss)+" C/cm^2")//
    calculation
Qss1=Qss/e
disp("charge density on semiconductor surface (in
    terms of number of charges) ,Qss*=Qss/e)="+string
    (Qss1)+" electrons/cm^2")//calculation
```

Scilab code Exa 8.15 VALUE OF Id

```
clc
L=3*10^-6
disp("L = "+string(L)+" meter") //initializing value
     of length of channel.
un=800
```

```
disp("un = "+string(un)+"cm^2/V-s") //initializing
    value of mobility of n-channel MOS transistor.
VT=1
disp("VT = "+string(VT)+" V") //initializing value
    of threshold Voltage.
Vgs=0
disp("Vgs = "+string(Vgs)+" V") //initializing value
    of gate to source voltage.
tox=500*10^-8
disp("tox = "+string(tox)+" cm") //initializing
    value of gate oxide thickness.
et=3.9
disp("et = "+string(et)) //initializing value of
    relative permittivity.
eo=8.85*10^-14
disp("eo = "+string(eo)+"F/cm") //initializing value
    of free space permittivity.
W=30*10^-6
disp("W = "+string(W)+"m") //initializing value of
    width of channel.
eox=(eo*et)
disp("eox=(eo*et)=)"+string(eox)+" F/cm^2")//
    calculation
Id=((eox*un*W)/(tox*L)*((Vgs-VT)^2/(2)))
disp("Drain current,Id=((eox*un*W)/(tox*L)*((Vgs-VT)
    ^2/(2))))="+string(Id)+" A")//calculation
```

Scilab code Exa 8.16 VALUE OF Id

```
clc
L=2.5*10^-6
disp("L = "+string(L)+" meter") //initializing value
    of length of channel.
un=800
disp("un = "+string(un)+"cm^2/V-s") //initializing
```

```
    value of mobility of n-channel MOS transistor.
VT=0.8
disp("VT = "+string(VT)+" V") //initializing value
    of threshold Voltage.
Vgs=1
disp("Vgs = "+string(Vgs)+" V") //initializing value
    of gate to source voltage.
tox=400*10^-8
disp("tox = "+string(tox)+" cm") //initializing
    value of gate oxide thickness.
et=3.9
disp("et = "+string(et)) //initializing value of
    relative permittivity.
eo=8.85*10^-14
disp("eo = "+string(eo)+"F/cm") //initializing value
    of free space permittivity.
eox=(eo*et)
disp("eox=(eo*et))="+string(eox)+" F/cm^2")//
    calculation
W=25*10^-6
disp("W = "+string(W)+"m") //initializing value of
    width of channel..
Id=((eox*un*W)/(tox*L)*((Vgs-VT)^2/(2)))
disp("Drain current,Id=((eox*un*W)/(tox*L)*((Vgs-VT)
    ^2/(2))))="+string(Id)+" A")//calculation
```

Scilab code Exa 8.17 RATIO OF W AND L

```
clc
un=525
disp("un = "+string(un)+"cm^2/V-s") //initializing
    value of mobility of n-channel MOS transistor.
VT=0.75
disp("VT = "+string(VT)+" V") //initializing value
    of threshold Voltage.
```

```
Vgs=2
disp("Vgs = "+string(Vgs)+" V") //initializing value
    of gate to source voltage.
tox=400*10^-8
disp("tox = "+string(tox)+" cm") //initializing
   value of gate oxide thickness.
et=3.9
disp("et = "+string(et)) //initializing value of
   relative permittivity.
eo=8.85*10^-14
disp("eo = "+string(eo)+"F/cm") //initializing value
    of free space permittivity.
eox=(eo*et)
disp("eox=(eo*et))="+string(eox)+" F/cm^2")//
   calculation
Id=6*10^-3
disp("Id = "+string(Id)+"A") //initializing value of
    width of channel..
X=((Id*tox*2)/(eox*un*((Vgs-VT)^2)))
disp("width to length ratio ,W/L=((Id*tox*2)/(eox*un
   *((Vgs-VT)^2))))="+string(X))//calculation
```

Scilab code Exa 8.18 PINCH OFF VOLTAGE

```
clc
Nd=2*10^16
disp("Nd = "+string(Nd)+" /cm^3") //initializing
   value of donor ion  concentration.
a=2*10^-4
disp("a = "+string(a)+" cm") //initializing value of
    height of channel at pinch off.
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing
   value of charge of electrons.
Er=11.9
```

```
9  disp("Er = "+string(Er)) //initializing value of
       relative permittivity.
10 Eo=8.85*10^-14
11 disp("Eo = "+string(Eo)+"F/cm") //initializing value
       of free space permittivity.
12 E=(Eo*Er)
13 disp("E=(Eo*Er))="+string(E)+" F/cm^2")//calculation
14 Vp=((e*Nd*a^2)/(2*E))
15 disp("Pinch off Voltage,Vp=((e*Nd*a^2)/(2*E)))="+
       string(Vp)+" V")//calculation
```

Scilab code Exa 8.20 VALUE OF rDS

```
1  clc
2  a=2*10^-4
3  disp("a = "+string(a)+" cm") //initializing value of
       height of channel at pinch off.
4  Er=11.9
5  disp("Er = "+string(Er)) //initializing value of
       relative dielectric permittivity constant .
6  Eo=8.854*10^-14
7  disp("Eo = "+string(Eo)+" F/cm") //initializing
       value of permittivity of free space.
8  un=1350
9  disp("un = "+string(un)+"cm^2/V-s") //initializing
       value of mobility of n-type silicon Mosfet.
10 W=8*10^-4
11 disp("W = "+string(W)+" cm") //initializing value of
       width of p-substrate.
12 L=10*10^-4
13 disp("L = "+string(L)+" cm") //initializing value of
       length of p-substrate.
14 e=1.6*10^-19
15 disp("e = "+string(e)+" columns") //initializing
       value of charge of electrons.
```

```
16 Vp=4
17 disp("Vp = "+string(Vp)+" V") //initializing value
     of thickness of p-substrate.
18 Vgs=0
19 disp("Vgs = "+string(Vgs)+" V") //initializing value
     of gate to source voltage.
20 E=Eo*Er
21 disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
     )//calculation
22 Nd=((Vp*2*E)/(e*a^2))
23 disp("Donor ion concentration ,Nd=((Vp*2*E)/(e*a^2)))
     ="+string(Nd)+" /cm^3")//calculation
24 rds=(L/(W*a*e*un*Nd))
25 disp("On Drain resistance ,rds=(L/(W*a*e*un*Nd)))="+
     string(rds)+" ohm")//calculation
```

Chapter 10

SILICON CONTROLLED RECTIFIER

Scilab code Exa 10.2 VALUE OF PUNCH THROUGH VOLTAGE

```
clc
Nd=10^14
disp("Nd = "+string(Nd)+" /cm^-3") //initializing
    value of donor ion  concentration.
Er=11.9
disp("Er = "+string(Er)) //initializing value of
    relative dielectric permittivity constant .
e=1.6*10^-19
disp("e = "+string(e)+" columns") //initializing
    value of charge of electrons.
Eo=8.854*10^-14
disp("Eo = "+string(Eo)+" F/cm") //initializing
    value of permittivity of free space.
W=100*10^-4
disp("W = "+string(W)+" cm") //initializing value of
     width of SCR.
E=Eo*Er
disp("total permittivity ,E=Eo*Er="+string(E)+" F/cm"
    )//calculation
```

```
14 Vpt=(e*Nd*W^2)/(2*E)
15 disp("Punch trough voltage,Vpt=(e*Nd*W^2)/(2*E))="+
   string(Vpt)+" V")//calculation
```

Scilab code Exa 10.3 VALUE OF THE DIFFERENTIAL TERM

```
1  clc
2  Ia=2*10^-3
3  disp("Ia = "+string(Ia)+" A") //initializing value
   of forward current of thyrsistor.
4  x=0.9
5  disp("(ap+an) = "+string(x)) //initializing value of
    sum of current gain of n,ptype semiconductor [
   value is get in by variable x,but represented on
   console window through ap +an].
6  a=0.45
7  disp("a = "+string(a)) //initializing value of
   current gain of both n,p type semiconductor (as
   it is assume that ap[current gain of n type
   semiconductor]=an[current gain of ptype
   semiconductor]in the question).
8  Ico=Ia*(1-(2*a))
9  disp("Ico=Ia*(1-(2*an))="+string(Ico)+" A")//
   calculation
10 y=1/2*Ico*((Ia)^-2)
11 disp("(da/dt)=1/2*Ico*((Ia)^-2))="+string(y)+" /A")
   //calculation
12
13 //The answer for (da/dt) after  doing calculation is
    provided wrong in the book.
```
