

Scilab Textbook Companion for
Electronic Devices and Circuits
by S. L. Kakani and K. C. Bhandari¹

Created by
S Baitlangki Rapsang
Engineering
Others
Sri Jayachamarajendra College of Engineering
College Teacher
None
Cross-Checked by
K. V. P. Pradeep

January 30, 2015

¹Funded 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: Electronic Devices and Circuits

Author: S. L. Kakani and K. C. Bhandari

Publisher: Viva Books, New Delhi

Edition: 1

Year: 2012

ISBN: 9788130917726

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	5
1 Electronic Materials and Components	10
2 Semiconductor Physics	15
3 Semiconductor Diodes	24
4 Diode Circuits	39
5 Transistors	51
6 Thermal Stability	54
7 Amplifiers	58
8 Feedback Amplifiers	65
9 Negative Feedback Amplifier using Opamp	69
10 Operational Amplifier	74
11 Power Amplifiers	79
12 Oscillators	84
14 Optical Fibres and Communication	88
15 Communication System	98

18 Optoelectronic Devices	103
19 Digital Electronics	108
21 Integrated Circuits	123

List of Scilab Codes

Exa 1.1	Color Coding Resistance	10
Exa 1.2	Color Coding Resistance	11
Exa 1.3	Color Coding Resistance	11
Exa 1.4	Current Source Representation	12
Exa 1.5	Percentage Variation in Load Voltage and Current	13
Exa 2.1	Conductivity and Resistivity	15
Exa 2.2	Resistivity	15
Exa 2.3	Resistance	16
Exa 2.4	Density Ratio	16
Exa 2.5	Number Density of Donor Atoms	17
Exa 2.6	Silicon and Germanium Semiconductors	17
Exa 2.7	Average Relaxation Time	18
Exa 2.8	Conduction Current Density	18
Exa 2.9	Number of Mobile Carriers Derivation	20
Exa 2.10	Hall Voltage	21
Exa 2.11	Hall Effect Experiment	22
Exa 2.12	Wavelength of Electromagnetic Radiation	22
Exa 3.1	Ohms Law	24
Exa 3.2	Voltage across Junction	24
Exa 3.3	PN Junction Forward Voltage	25
Exa 3.4	Incremental Resistance	25
Exa 3.5	Forward and Reverse Biased	26
Exa 3.6	AC Resistance	26
Exa 3.7	Change in Temperature	27
Exa 3.8	Forward Current	28
Exa 3.9	Static and Dynamic Resistance	29
Exa 3.10	Voltage across Junction	29
Exa 3.11	Value of Resistance and Power Dissipation	30

Exa 3.12	DC Resistance	31
Exa 3.13	Current through Diode	31
Exa 3.14	Current through Resistor	33
Exa 3.15	Current Drawn from Battery	33
Exa 3.16	Breakdown Voltage Ranges	34
Exa 3.17	Capacitance	35
Exa 3.18	Voltage across each Diode	35
Exa 3.19	LED Current	36
Exa 3.20	AC Resistance	37
Exa 3.21	AC Resistance	37
Exa 4.1	Half Wave Rectifier	39
Exa 4.2	No Load Voltage	40
Exa 4.3	AC Input Power	40
Exa 4.4	Half Wave Rectifier	41
Exa 4.5	Bridge Rectifier	42
Exa 4.6	Half Wave Rectifier	42
Exa 4.7	Full Wave Rectifier	43
Exa 4.8	Full Wave Rectifier	44
Exa 4.9	Full Wave Rectifier	45
Exa 4.10	Full Wave Rectifier	46
Exa 4.11	Full Wave Rectifier	46
Exa 4.12	Zener Current	47
Exa 4.13	Current and Wattage of Elements	48
Exa 4.14	Zener Diode	49
Exa 4.15	Voltage Regulator Circuit	49
Exa 5.1	Current component of a Transistor	51
Exa 5.2	Transistor Components	51
Exa 5.3	Transistor Components	52
Exa 6.1	Collector Base Junction Temperature	54
Exa 6.2	Dissipation Capacity of a Transistor	55
Exa 6.3	Fixed Bias Circuit	56
Exa 6.4	Collector to Base Bias Circuit	56
Exa 7.1	Coupling	58
Exa 7.2	Bypassing	58
Exa 7.3	Small Signal Amplifiers	59
Exa 7.4	Amplifier Circuit	60
Exa 7.5	RC Coupled CE Amplifier	61
Exa 7.6	Audio Amplifier	62

Exa 7.7	RC Coupled CE Amplifier	62
Exa 7.8	RC Coupled Amplifier	63
Exa 7.9	FET Amplifier	63
Exa 8.1	Voltage Gain of Amplifier	65
Exa 8.2	Bandwidth of Amplifier	66
Exa 8.3	Negative Feedback Factor	66
Exa 8.4	Feedback Voltage Gain	66
Exa 8.5	Feedback Voltage Gain	67
Exa 8.6	Feedback Amplifier	67
Exa 9.1	VCVS Amplifier	69
Exa 9.2	ICVS Amplifier	69
Exa 9.3	ICIS Amplifier	70
Exa 9.4	VCVS Amplifier	70
Exa 9.5	VCVS Amplifier	71
Exa 9.6	VCVS Amplifier	71
Exa 9.7	ICIS Amplifier	72
Exa 9.8	VCIS Amplifier	72
Exa 10.1	Common Mode Voltage Gain	74
Exa 10.2	Differential Amplifier	74
Exa 10.3	Opamp Operating Circuit	75
Exa 10.4	Opamp Operating in Inverting Mode	76
Exa 10.5	Closed Loop Voltage Gain	76
Exa 10.6	Opamp Operating in Inverting Mode	77
Exa 10.7	Power Bandwidth of Opamp	78
Exa 10.8	Resistances of Opamp	78
Exa 11.1	RC Coupled Amplifier	79
Exa 11.2	Class B Transformer Coupled Amplifier	80
Exa 11.3	Tuned Amplifier	81
Exa 11.4	RC Coupled Amplifier	81
Exa 11.5	Class C Tuned Amplifier	83
Exa 12.1	Colpitts Oscillator	84
Exa 12.2	Phase Shift Oscillator	85
Exa 12.3	Oscillator	85
Exa 12.4	Pierce Oscillator	86
Exa 12.5	Oscillator	87
Exa 14.1	Optical fibre	88
Exa 14.2	Multimode Index Fibre	88
Exa 14.3	Multimode Step Index Fibre	89

Exa 14.4	Optical Fibre	90
Exa 14.5	Step Index Fibre	90
Exa 14.6	Optical Fibre	91
Exa 14.7	Optical Fibre	91
Exa 14.8	Optical Fibre	92
Exa 14.9	Step Index Fibre	92
Exa 14.10	Step Index Fibre Single Mode	93
Exa 14.11	Loss in Optical Fibre	93
Exa 14.12	Communication System Output Power	94
Exa 14.13	GRIN Optical Fibre	94
Exa 14.14	Multimode Step Index Optical Fibre	95
Exa 14.15	Optical Fibre	96
Exa 14.16	Single Mode Step Index Fibre	96
Exa 15.1	Transmitter Modulation	98
Exa 15.2	Transmitter Modulation	98
Exa 15.3	AM Wave Modulation	99
Exa 15.4	AM Wave Modulation	100
Exa 15.5	AM Wave Modulation	100
Exa 15.6	AM Wave Modulation	101
Exa 15.7	AM Wave Modulation	102
Exa 18.1	DC Circuit of LED	103
Exa 18.2	Spontaneous and Stimulated Emission	103
Exa 18.3	Relative Populations	104
Exa 18.4	Silicon Crystal Problem	105
Exa 18.5	Electrons Diffusion	106
Exa 18.6	Frequency Bandwidth Limitations	106
Exa 18.7	Semiconductor Laser	107
Exa 18.8	Photoconductor Device	107
Exa 19.1	Binary to Decimal	108
Exa 19.2	Binary to Decimal	108
Exa 19.3	Binary to Decimal	109
Exa 19.4	Binary to Decimal	109
Exa 19.5	Binary to Decimal	109
Exa 19.6	Decimal to Binary	110
Exa 19.7	Decimal to Binary	110
Exa 19.8	Decimal to Binary	111
Exa 19.9	Decimal to Binary	111
Exa 19.10	Binary to Decimal	112

Exa 19.11	Octal to Decimal	112
Exa 19.12	Octal to Decimal	113
Exa 19.13	Decimal to Octal	113
Exa 19.14	Decimal to Octal	113
Exa 19.15	Octal to Binary	114
Exa 19.16	Octal to Binary	114
Exa 19.17	Binary to Octal	115
Exa 19.18	Hexadecimal to Binary	115
Exa 19.19	Hexadecimal to Binary	116
Exa 19.20	Binary to Hexadecimal	116
Exa 19.21	Hexadecimal to Decimal	116
Exa 19.22	Decimal to Hexadecimal	117
Exa 19.23	Decimal to Hexadecimal	117
Exa 19.24	Binary Addition	118
Exa 19.25	Binary Subtraction	118
Exa 19.26	Binary Subtraction	118
Exa 19.27	Boolean Expression	119
Exa 19.28	Boolean Expression	120
Exa 19.30	Boolean Expression	120
Exa 19.31	Boolean Expression	121
Exa 21.1	AC Coupled Trigger Pulse	123
Exa 21.2	Astable Mode Time Output	123
Exa 21.3	Astable Mode Time Output	124
Exa 21.4	Astable Mode Output Frequency	124
Exa 21.5	Astable Mode Duty Cycle	125
Exa 21.6	Resistor Values in Astable Mode	125
Exa 21.7	Time Delay Mode	125
Exa 21.8	Decimal to TTL Voltage Terms	126

Chapter 1

Electronic Materials and Components

Scilab code Exa 1.1 Color Coding Resistance

```
1 //Ex1_1 Pg-43
2 clc
3 disp("Refer to the figure 1.52")
4 disp("Hold the resistor as shown in the figure such
   that tolerance is on your extreme right.")
5 disp("Now the value of the resistor is equal to")
6 disp("  Red      Black      Blue      Gold")
7 disp("    2          0          6          (+/-)5%")
8 red=2 //red value
9 blk=0 //black value
10 blu=6 //blue value
11 gld=5 //gold value
12 value_res=(red*10+blk)*10^blu //value of resistor
13 printf("\n The value of resistor is %.0f ohm (+/-)%
   .0f%%",value_res,gld)
14 per_val=0.05*value_res
15 pos_value_res=value_res+per_val //positive range of
   resistor
16 neg_value_res=value_res-per_val //negative range of
```

```

    resistor
17 printf("\n The value of resistor is %.0f Mohm and %
    .0f Mohm",neg_value_res*1e-6,pos_value_res*1e-6)

```

Scilab code Exa 1.2 Color Coding Resistance

```

1 //Ex1.2 Pg-43
2 clc
3 disp("With the help of colour coding table , one
    finds")
4 disp(" 1st_Band    2nd_Band    3rd_Band    4th_Band
    ")
5 printf("      Yellow      Violet      Orange      Gold
    ")
6 disp("      4          7          10^3      (+/-)5%")
7 yel=4 //yellow value
8 vio=7 //violet value
9 org=1e3 //orange value
10 gld=5 //gold value in %
11 val_res=(yel*10+vio)*org
12 printf("\n The value of resistor is %.2f kohm (+/-)%
    .0f%%",val_res*1e-3,gld)
13 gld_ab=0.05 //absolute gold value
14 per_val=gld_ab*val_res
15 printf("\n Now, 5%% of 47k_ohm = %.0f ohm",per_val)
16 range_high=val_res+per_val //higher range
17 range_low=val_res-per_val //lower range
18 printf("\n\n Thus resistance should be within the
    range %.2f kohm(+/-)%.2f Kohm,\n or between %.2f
    kohm and %.2f kohm.",val_res*1e-3,per_val*1e-3,
    range_low*1e-3,range_high*1e-3)

```

Scilab code Exa 1.3 Color Coding Resistance

```

1 //Ex1_3 Pg-44
2 clc
3 disp("The specification of the resistor from the
      color coding table is as follows")
4 disp(" 1st_Band    2nd_Band    3rd_Band    4th_Band
      ")
5 printf("      Gray      Blue      Gold
      Silver")
6 disp("      8          6          10(-1)      (+/-)10%
      ")
7 gray=8 //gray value
8 blu=6 //blue value
9 gld=10-1 //gold value
10 sil=10 //silver value in %
11 val_res=(gray*10+blu)*gld
12 printf("\n The value of resistor is %.1f ohm (+/-)%
      .0f%%",val_res,sil)
13 sil_ab=0.1 //absolute gold value
14 per_val=sil_ab*val_res
15 printf("\n Now, 10%% of 8.6 ohm = %.2f ohm",per_val)
16 range_high=val_res+per_val //higher range
17 range_low=val_res-per_val //lower range
18 printf("\n\n Obviously resistance should lie within
      the range %.2f ohm(+/-)%.2f ohm,\n or between %.2
      f ohm and %.2f ohm.",val_res,per_val,range_high,
      range_low)

```

Scilab code Exa 1.4 Current Source Representation

```

1 //Ex1_4 Pg-44
2 clc
3 disp("Refer to the figure 1.53")
4 Vs=2 //supply voltage in V
5 Rs=1 //resistance in ohm
6 Is=Vs/Rs

```

```

7 printf("\n Current Is = %.0f A \n",Is)
8 disp("      Internal resistance remains the same but
      is now connected in parralel with the current
      sourceIS ,as shown in Figure 1.51(a)")
9 disp("      Now,we connect a load resistance R_L=1 ohm
      across the terminals of two representations ,and
      find I_L and V_L. From Figure 1.54(b) and using
      the current-divider concept,one obtains")
10 RL=1 //load resistance in ohm
11 IL=Is*(Rs/(Rs+RL)) //load current using current-
      divider
12 VL=IL*RL //load voltage
13 printf("\n Load voltage = %.0f V",VL)
14 printf("\n Load current = %.0f A \n",IL)
15 disp("From equation 53(b),using the voltage-divider
      concept,one obtains")
16 VD_v1=Vs*(RL/(RL+Rs)) //load voltage using voltage
      divider
17 VD_il=VL/RL //load current
18 printf("\n Load voltage = %.0f V",VD_v1)
19 printf("\n Load current = %.0f A \n",VD_il)

```

Scilab code Exa 1.5 Percentage Variation in Load Voltage and Current

```

1 //Ex1_5 Pg-45
2 clc
3 disp("Refer to the figure 1.55")
4 disp("(a) R_L varies from 1 ohm to 10 ohm.")
5 disp("Currents for two extreme values of R_L are")
6 Vs=10 //supply voltage
7 RL1=1 //resistance RL1
8 Rs=100 //source resistance
9 IL1=(Vs/(RL1+Rs))
10 RL2=10
11 IL2=(Vs/(RL2+Rs))

```

```

12 per_var_cur=((IL1-IL2)/IL1)*100
13 printf("\n Percentage variation in current = %.2f %%
      \n",per_var_cur)//answer in the text book took a
      .3 decimal round off value
14 disp(" Now,load voltage for the two extreme values
      of R_L are")
15 VL1=IL1*RL
16 VL2=IL2*RL2
17 per_var_vol=((VL2-VL1)/VL2)*100
18 printf("\n Percentage variation in current = %.2f %%
      \n",per_var_vol)
19
20 disp("(b) R_L varies from 1 k-ohm to 10 k-ohm
      (Figure 1.55(b))")
21 disp(" Currents for the two extreme values R_L are")
22 RL11=1000
23 IL11=(Vs/(RL11+Rs))
24 RL22=10000
25 IL22=(Vs/(RL22+Rs)) //mistake in book value
26 per_var_cur11=((IL11-IL22)/IL11)*100
27 printf("\n Percentage variation in current = %.2f %%
      \n",per_var_cur11) //mistake in book value

```

Chapter 2

Semiconductor Physics

Scilab code Exa 2.1 Conductivity and Resistivity

```
1 //Ex2_1 Pg-87
2 clc
3 disp("Conductivity of pure silicon crystal is given
      by")
4 disp("    sigma = ni*e*(ue + uh)")
5 uh=480 //mobility in cm^2/Volt-sec
6 ue=1350 //mobility of electrons in cm^2/Volt-sec
7 e=1.6*10^(-19) //electron charge
8 ni=1.072*10^10 //density of electron hole
9 sigma_i=ni*e*(uh+ue) //conductivity of silicon
10 printf("\n Conductivity of pure silicon crystal = %
        .8f ohm^(-1)/cm \n",sigma_i)
11 disp("Resistivity of silicon is given by")
12 rho=1/sigma_i //resistivity of silicon
13 printf("\n Resistivity of pure silicon crystal = %.0
        f ohm-cm \n",rho)
```

Scilab code Exa 2.2 Resistivity


```

1 //Ex2_2 Pg-87
2 clc
3 disp("sigma = u*e*n")
4 u=1200 //mobility
5 e=1.6*10^(-19) //electron charge
6 n=10^13 //phosphorous concentration
7 sigma=u*e*n //conductivity
8 printf("\n Conductivity of pure silicon crystal = %
      .5f ohm^(-1)/cm \n",sigma)
9 rho=1/sigma //resistivity
10 printf("\n Resistivity of pure phosphorous = %.0f
      ohm-cm \n",rho)

```

Scilab code Exa 2.3 Resistance

```

1 //Ex2_4 Pg-88
2 clc
3 disp("(n_i)^2 = n*p = n_p*N_a")
4 ni=2.5*10^19 //density of electron hole
5 Na=1.1*10^20 //acceptor density
6 np=(ni^2)/Na
7 N=np/ni
8 printf("\n The ratio of n_p/n_i = %.4f",N)

```

Scilab code Exa 2.4 Density Ratio

```

1 //Ex2_4 Pg-88
2 clc
3 clear
4
5 disp("(n_i)^2 = n*p = n_p*N_a")
6 ni=2.5*10^19 //density of electron hole per m^3
7 Na=1.1*10^20 //acceptor density in atoms/m^3

```

```

8 np=(ni^2)/Na //number of holes
9 N=np/ni //The ratio of np/ni
10 printf("\n The ratio of n_p/n_i = %.2f",N)

```

Scilab code Exa 2.5 Number Density of Donor Atoms

```

1 //Ex2_5 Pg-88
2 clc
3 disp("sigma_n = u_n*e*N_d")
4 e=1.6*1e-19 //electron charge
5 sigma=5 //conductivity in mho/cm
6 un=3850 //mobility of electrons
7 Nd=sigma/(e*un) //concentration
8 printf("Number density of donor atoms = %.1f*1e16
        per cm^3",Nd*1e-16)
9 //the answer in the book is wrong the correct answer
    is 0.8*1e16 per cm^3

```

Scilab code Exa 2.6 Silicon and Germanium Semiconductors

```

1 //Ex2_6 Pg-88
2 clc
3 printf("We know forbidden energy gap between
        conduction and valence \n bands for a
        semiconductor is nearly 1 eV. For Ge and Si ,
        energy \n gap is 0.785 eV and 1.21 eV
        respectively at 0K. Energy gap \n decreases with
        increase in temperature which is represented \n
        by the expression . Obviously, Si and Ge will
        remain \n semiconductors at 1000K ambient
        temperature. \n\n")
4
5 disp("Eg(T) = 1.21 - 3.6*10^(-4)*T (For Si)")

```

```

6 T=1000 //temperature
7 Eg(T) = 1.21 - 3.6*10^(-4)*T
8 printf("\n Eg(1000) = 1.21 - 3.6*10^(-4)*1000 = %.2 f
    eV" ,Eg(T))

```

Scilab code Exa 2.7 Average Relaxation Time

```

1 //Ex2_7 Pg-88
2 clc
3 disp("Relaxation time in terms of mobility is given
    by")
4 disp("          t=m*u/e")
5 printf("\n\n Taking effective mass of electron an
    holes in consideration ,\n relaxation time is
    given by \n")
6 disp("          t=m_star*u/e")
7 disp("(a) foe electrons ,m_star = 0.259*m_0")
8 m0=9.1*10^(-31)
9 ue=0.135 //mobility of electrons
10 e=1.6*10^(-19) //electron charge
11 t_e=(0.259*m0*ue)/e
12 printf("\n Average relaxation time of eletrons = %.2
    f*1e-13 secs\n" ,t_e*1e13)
13
14 uh=0.048 //mobility of holes
15 disp("(b) For holes in the valance band ,m=0.537*m_0"
    )
16 t_h=(0.537*m0*uh)/e
17 printf("\n Average relaxation time of eletrons = %.2
    f*1e-13 secs\n" ,t_h*1e13)

```

Scilab code Exa 2.8 Conduction Current Density

```

1 //Ex2_8 Pg-89
2 clc
3
4 disp(" Conductivity of an intrinsic material is given
   by ")
5 disp("      sigma = e*ni*un*(1+up/un)")
6 sigma_i=100/60
7 ni=2.5*10^19 //concentration of intrinsic carrier in
   m^3
8 up_un=0.5
9 e=1.6*10^(-19) //electron charge
10 un=(sigma_i/(e*ni*(1+(up_un)))) //concentration of
   electrons
11 up=un/2 //holes concentratin
12
13 disp("      Let n be eletron concentration and p be
   hole con-centration for doped sample. Sincethe
   sample is electrically neutral ,we have")
14 disp("      Nd + p = Na + n")
15 disp("      Where Nd is donor concentration and Na is
   acceptor concentration ,assumed to be fully
   ionized. From mass action law ,we have np =ni^2*S0
   ")
16 disp("      Nd + (ni^2)/n = Na + n")
17 disp(" or      n^2 + (Na-Nd)*n-ni^2 = 0")
18 disp(" or      n=0.5*([Nd-N]) (+/-) sqrt (Nd-Na)^2 +
   4*ni^2)")
19 disp("      n is positive and hence we can drop
   negative sign before the radical")
20
21 Nd=10^20 //number density of donor atoms /m^3
22 Na=5*10^19 //number of acceptor atoms in /m^3
23 n=0.5*((Nd-Na)+ sqrt((Nd-Na)^2 + 4*ni^2)) //electron
   concentration for doped sample
24 p=ni^2/n //hole concentration for doped sample
25 conduct_doped=e*(n*un+p*up) //conductivity of doped
   sample(value in textbook is wrong)
26 printf("\n Conductivity of doped sample = %.2f S/m \

```

```

    n",conduct_doped)
27
28 disp("      We have assumed that carrier mobilities
      in the doped signal sample are the same as those
      in the intrinsic material.This assumption is
      justified by the low doping level , and Coulomb
      scattering Applied by the ionized impurities is
      weak at 300K.")
29 disp(" Applied electric field")
30
31 F_cm=2 //applied electric field in V/cm
32 F_m=2*100 //applied electric field in V/m
33 J=conduct_doped*F_m //total conduction current
      density
34 J_cm=J/1000 //cm^2 to m^2s
35
36 printf("Total conduction current density = %.0 f A/m
      ^2", J)
37 printf("\n                                = %.2 f A/cm
      ^2", J_cm)

```

Scilab code Exa 2.9 Number of Mobile Carriers Derivation

```

1 //Ex2_9 Pg-91
2 clc
3
4 disp("For P-type ,Nd=0. By charge neutrality and mass
      action law ,")
5
6 disp("      p + Nd = p = +na = (ni)^2/n")
7
8 disp("or      n^2 + Na*n - (ni)^2 = 0")
9
10 disp("Solving the quadratic equation for n and
      discharging the extraneous negative root ,one

```

```

    obtains")
11
12 disp("      n = 0.5*(-Na + sqrt(Na^2 + 4*ni^2))")
13
14 disp("Knowing n, one obtains from mass action law p =
      ni^2/n")
15
16 disp("For N-type doping ,Na=0. By analogous procedure
      ,")
17
18 disp("      p = 0.5*(-Nd +sqrt(Nd^2 + 4*Ni^2));  n=
      ni^2/p")

```

Scilab code Exa 2.10 Hall Voltage

```

1 //Ex2_10 Pg-91
2 clc
3
4 e=1.6*10^(-19) //electron charge
5 un=700 //mobility of silicon
6 n=10^17 //concentration of phosphorous atoms
7 sigma=e*un*n //conductivity
8 printf("Conductivity = %.1f (ohm-cm)^-1",sigma)
9 res=sigma^(-1) //resistivity
10 printf("\n Resistivity = %.4f ohm-cm",res)
11 Rh=-(e*n)^(-1) //hall coefficient
12 printf("\n Hall coefficient = %.1f cm^3/C",Rh)
13 Ix=10^-3
14 Bz=10^(-5)
15 x=10^(-2)
16 Vh=(Ix*Bz*Rh)/x
17 printf("\n Hall Volage = %.1f uV",Vh*10^6)

```

Scilab code Exa 2.11 Hall Effect Experiment

```
1 //Ex2_11 Pg-91
2 clc
3
4 disp("(a) Hall coefficient is")
5 disp("      Rh = Eh/(Jx*B)")
6 Vh=21.4*10^(-3) //hall voltage
7 b=1.7*10^(-2) //breadth
8 Eh=Vh/b //electric field
9 t=0.052*10^(-3) //thickness
10 I=200*10^(-6) //current
11 Jx=I/(b*t) //current density
12 B=0.5 //magnetic field
13 Rh=Eh/(Jx*B)
14 printf("\n Hall coefficient = %.6f m^3/C \n\n",Rh)
15
16 disp("(b) Electrons per unit volume,")
17 e=1.6*10^(-19) //electron charge
18 n=1/(Rh*e) //electrons per unit volume
19 printf("\n\n Electrons per unit volume = %.0f
      electrons/m^3 \n\n",n)
20 V=195*10^(-3) //voltage
21 I=200*10^(-6) //current
22 R=V/I //resistance
23 disp(" Since R=(1/(A*sigma) = (1/(A*e*n*R)))")
24 l=2.65*10^(-2) //length
25 w=1.7*10^(-2) //width
26 t=0.052*10^(-3) //thicknes
27 A=t*w //area
28 sigma=1/(A*e*n*R) //conductivity
29 printf("\n\n Conductivity = %.3f m^3/Vs",sigma)
```

Scilab code Exa 2.12 Wavelength of Electromagnetic Radiation

```
1 //Ex2_12 Pg-92
2 clc
3
4 C=3*10^8 //speed of light in m/s
5 h=6.6*10^(-34) //plank's constant in J
6 Eg=1.98*1.6*10^(-19) //band gap
7 lamda=(C*h)/Eg //wavelength
8 printf("Wavelength = %.0f nm \n",lamda*10^9)
9 disp("      Since lamda is in the red region of the
      visible light and hence the colour of emitted
      radiation is RED")
```

Chapter 3

Semiconductor Diodes

Scilab code Exa 3.1 Ohms Law

```
1 //Ex3_1 Pg-127
2 clc
3
4 disp("Refer to the diagram 3.11(b)")
5 disp("Using ohm' 's law")
6 disp("Vt = Vd1 + Vr1")
7 Vd1=0.7 //voltage drop in V
8 Vt=12 //supply voltage in V
9 R1=1.2*10^3 //resistor R1 in ohm
10 Vr1=Vt-Vd1 //voltage across R1 in V
11 It=Vr1/R1 //current in A
12 disp("Ohm' 's law")
13 printf("\n Current I_t = %.2f mA",It*10^3)
```

Scilab code Exa 3.2 Voltage across Junction

```
1 //Ex3_2 Pg-182
2 clc
```

```

3 J=10^5 //forward current density
4 Js=250*10^(-3) //saturation current density
5 e=1.6*10^(-19) //electron charge
6 T=300 //temperature
7 k=1.38*10^(-23) //Boltzmann constant
8 V=(log(J/Js)*k*T)/e //voltage applied across
    junction
9 printf("The voltage applied across the junction =%.2
    f V",V)

```

Scilab code Exa 3.3 PN Junction Forward Voltage

```

1 //Ex3_3 Pg-182
2 clc
3
4 I=2*10^6 //forward current density
5 Is=30 //saturation current density
6 ekt=40
7 V=(1/40)*log(I/Is) //Applied forward voltage
8 printf("Applied forward voltage = %.3f V",V)

```

Scilab code Exa 3.4 Incremental Resistance

```

1 //Ex3_4 Pg-182
2 clc
3
4 disp("          I = Is*exp(eV/kT) = Is*exp(40V)")
5 disp("          Re = del_V/del_I = 1/40I")
6 disp("          Dividing throughtout by volume, one
          obtains the equation in the form of current
          density as")
7 disp("          J = Js*(exp(eV/kT)-1)")
8 J=10^5 //forward current density

```

```

9 Js=250*10^(-3) //saturation current density
10 e=1.6*10^(-19) //electron charge
11 T=300 //temperature
12 k=1.38*10^(-23) //Boltzmann constant
13 V=(log(J/Js)*k*T)/e //voltage applied across
    junction
14 printf("\n The voltage applied across the junction =
    %.2 f V",V)

```

Scilab code Exa 3.5 Forward and Reverse Biased

```

1 //Ex3_5 Pg-183
2 clc
3
4 disp("(a) Forward-bias")
5 Av=50 //applied voltage
6 Jr=5000 //junction resistance
7 Er=50 //external resistance
8 cur=Av/(Er+Jr) //current
9 printf("\n Current = %.1 f mA \n",cur*10^3)
10
11 disp("(b) Reverse-bias")
12 cur_rev=Av/(Jr+10^6) //book expression is wrong
13 printf("\n Current = %.3 f*1e-2 mA \n",cur_rev*10^5)

```

Scilab code Exa 3.6 AC Resistance

```

1 //Ex3_6 Pg-183
2 clc
3
4 disp("We know that")
5 disp(" r_ac = dV/dI - 1/(dI/dV) = 1/((I0/KT) exp(V/
    KT))")

```

```

6 k=8.62*10^(-5)
7 T=300 //temperaturein K
8 kT=k*T
9 I0=10^(-6) //saturation current
10 V=150*10^(-3) //voltage forward biased
11 r_ac = 1/((I0/kT)*exp(V/kT)) //value of exp
      (0.15/0.02586)=330.45 and not the textbook value
      of 332.66
12 printf("\n The AC resistance = %.2f ohm",r_ac) //
      text book value wrong

```

Scilab code Exa 3.7 Change in Temperature

```

1 //Ex3_7 Pg-184
2 clc
3
4 disp("We know that  $(I_0) \cdot T_2 = (I_0) \cdot T_1 \cdot (2)^{((T_2-T_1)/10)}$ 
      ")
5
6 disp("Substituting the given values ,we have ")
7
8 disp("(40*10^(-6)) = (25*10^(-6))*(2)^x where x=(T2-
      T1)/10")
9
10 disp("(2)^x = 1.6")
11
12 disp("Taking log on both sides ,one obtains")
13
14 disp("      x*log(2) = log(1.6)")
15
16 disp("or      x = log(1.6)/log(2)")
17
18 x=log(1.6)/log(2)
19
20 disp(" Now x = (T2-T1)/10 or 0.678 = (T2-25)/10")

```

```

21 T1=25 //temperature T1
22 T2=x*10+T1 //temperature T2
23 diff_temp=T2-T1 //change in temperature
24 printf("\n So the change in temperature = %.2f
        degree celsius",diff_temp)

```

Scilab code Exa 3.8 Forward Current

```

1 //Ex3_8 Pg-185
2 clc
3 disp("Forward current I is given by ")
4 disp(" I=I0*exp(V/(n*Vt))-1")
5
6 I_22=poly(0," I_22")
7 V=0.3 //voltage
8 n=1 //constant
9 T1=22+273 //temperature T1 in Kelvin
10 Vt1=T1/11600
11 I=I_22*(exp(V/0.025)-1)
12 disp("When temperature rises to 72 degree celcius ,
        then")
13
14 T2=72+273 //temperature T2 in Kelvin
15 Vt2=T2/11600
16 TR=T2-T1 //temperature rise
17 I_72=poly(0," I_72")
18 I_72=I_22*(2)^(TR/10)
19
20 I_hash=I_72*(exp(V/(Vt2))-1)
21 for_cur_rises=I_hash/I
22 disp("Thus, at 72 degree celcius Forward current
        rises by ")
23 disp(for_cur_rises)
24 cur_I=768849.72
25 cur_I_hash=162753.79

```

```

26 FCR=cur_I/cur_I_hash
27 printf("\n      = %.2 f",FCR)
28 //answer in the book is wrong

```

Scilab code Exa 3.9 Static and Dynamic Resistance

```

1 //Ex3_9 Pg-185
2 clc
3
4 n=1 //constant
5 T=27+273 //temperature in K
6 Vt=T/11600
7 V=0.2 //voltage
8 I0=10^(-6) //saturation current
9 I=I0*(exp(V/Vt)-1)
10 stat_res=V/I //static resistance
11 printf("Static resistance = %.2 f ohm \n",stat_res)
12
13 dyn_res=Vt/(I+I0) //dynamic resistance
14 printf("Dynamic resistance = %.2 f ohm",dyn_res)

```

Scilab code Exa 3.10 Voltage across Junction

```

1 //Ex3_10 Pg-186
2 clc
3
4 disp("We know that          I = I0*(exp(V/n*Vt)-1)")
5 disp("Dividing on both sides by area A, one obtains
   ")
6 disp("          I/A = I0/A*(exp(V/n*Vt)-1)")
7 disp("or          J = J0*(exp(V/n*Vt)-1)")
8 n=1 //constant
9 T=300 //temperature in K

```

```

10 Vt=T/11600
11 J=10^5 //forward current density
12 J0=250*10^(-3) //saturation current density
13 V=Vt*log(J/J0)
14 printf("\n The voltage applied across the junction =
    %.4f V",V)

```

Scilab code Exa 3.11 Value of Resistance and Power Dissipation

```

1 //Ex3_11 Pg-186
2 clc
3
4 Vmin=0.7 //minimum voltage across diode in V
5 V=5 //supply voltage in V
6 V_R1=V-Vmin //voltage across resistor R in V
7 Imin=10^(-3) //minimum current
8 R1=V_R1/Imin
9
10 printf("Maximum value of R =%.1f kohm \n ",R1*1e-3)
11
12 I=5*10^(-3) //current through resistance in A
13 V_R2=V-Vmin //voltage across resistor R in V
14 R2=V_R2/I
15 printf("\n\n Minimum value of R =%.0f ohm ",R2)
16
17 Vb=6 //supply voltage
18 Vb_res=Vb-Vmin //voltage across resistor
19 P=I*Vb_res //power dissipated across resistor
20 printf("\n\n Power dissipated across R =%.1f W",P
    *10^3)
21
22 P_diode=I*Vmin //power dissipated across diode
23 printf("\n power dissipated across diode =%.1f mW",
    P_diode*1e3)
24 R=10^3 //resistor in ohm

```

```

25 V_R=R*Imin //voltage drop across resistor R in V
26 Vb=V_R+Vmin
27 printf("\n\n The minimum voltage across diode = %.1f
      V",Vb)

```

Scilab code Exa 3.12 DC Resistance

```

1 //Ex3_12 Pg-188
2 clc
3
4 disp("Refer to the figure 3.51")
5 Id1=2*10^(-3) //diode current in I
6 Vd1=0.5 //diode voltage in V
7 Rf1=Vd1/Id1 //Dc resistance
8 disp("At Id=2mA and Vd=0.5V")
9 printf("\n Rf = %.0f ohm \n\n",Rf1)
10
11 Id2=20*10^(-3) //diode current in I
12 Vd2=0.75 //diode voltage in V
13 Rf2=Vd2/Id2 //Dc resistance
14 disp("At Id=20mA and Vd=0.75V")
15 printf("\n Rf = %.1f ohm \n\n",Rf2)
16
17 Id3=2*10^(-6) //diode current in I
18 Vd3=10 //diode voltage in V
19 Rf3=Vd3/Id3 //Dc resistance
20 disp("At Id=2*10^(-6)A and Vd=10V")
21 printf("\n Rf = %.0f Mohm \n\n",Rf3*10^-6)

```

Scilab code Exa 3.13 Current through Diode

```

1 //Ex3_13 Pg-188
2 clc

```



```

3
4 disp(" Refer to the figure 3.52")
5 disp("(a) Assuming the diode D to be ideal ")
6 disp("      Ignoring diode D,voltage across R2 is
      given as (By applying potential divider concept)"
      )
7 R1=45 //resistor R1
8 R2=5 //resistor R2
9 Vaa=10 //supply voltage
10 Vab=Vaa*(R2/(R1+R2))
11 printf("\n Vab= %.0f V \n",Vab)
12
13 disp(" Thus,diode D is forward biased.It conducts ,
      offering zero resistance Hence no current would
      flow through the parallel bransh R2.The circuit
      equivalent to that shown in figure 3.53(a)")
14 ID=Vaa/R1 //diode current
15 printf("\n Current through diode = %.0f mA \n\n",ID
      *10^3)
16
17 disp("(b)Assuming the diode to be real")
18 disp("      Voltage Vab is much larger than Vt
      hencethe diode conducts.It is replaced by its
      equivalent as shown in figure 3.53(b).To
      determine current Id through the diode we first
      find the Thevenin''s equivalent of the circuit on
      the left of AB.Vth=open circuit voltage across
      AB")
19 Vth=Vaa*(R2/(R1+R2))
20 printf("\n Vth=%.0f V \n",Vth)
21 Rth=R1*R2/(R1+R2)
22 printf("\n Rth=%.0f ohm \n",Rth) //textbook value is
      wrong
23 disp("Thus,the equivalent circuit becomes as shown
      in figure 3.53(c)")
24 Vt=0.3 //load voltage
25 tf=25 //load resistance
26 Id=(Vth-Vt)/(Rth+tf)

```

```
27 printf("\n Current through diode =%.1f mA \n\n", Id
    *10^3)
```

Scilab code Exa 3.14 Current through Resistor

```
1 //Ex3_14 Pg-190
2 clc
3
4 disp("      Diodes D2 and D3 are reverse-biased.
      Therefore, these are like open-switches. Diodes
      D1 and D2 are forward biased. These are replaced
      by their equivalent circuits, as shown in figure
      3.54. Since the diodes are silicon V=0.7V. ")
5 Vt=0.7 //voltage drop
6 Vaa=20 //supply voltage in V
7 net_emf=Vaa-Vt-Vt //net emf
8 R1=5
9 R2=90
10 R3=5 //R1,R2,R3 are resistances
11 tot_res=R1+R2+R3 //total resistance
12 disp("Therefore, current through 90 ohm resistor is"
    )
13 I=net_emf/tot_res
14 printf("\n Current I =%.0f mA", I*10^3)
```

Scilab code Exa 3.15 Current Drawn from Battery

```
1 //Ex3_15 Pg-190
2 clc
3
4 disp("(a) When the diode is forward biased figure
      3.55(b), it offers zero resistance. It is like
      shorted switch. This shorted switch across AB also
```

```

        short-circuits the resistance R2. Obviously, a
        parallel combination of the diode and R2 is
        equivalent to a resistance of zero ohms.")
5 R1=100 //resistor R1 in ohm
6 R=R1
7 Vaa=10 //supply voltage in V
8 I=Vaa/R
9 printf("\n Current drawn from battery =%.1f A \n\n",
    I)
10
11 disp("(a) When the diode is reverse biased figure
    3.55(b). It is like open switch. Obviously, it then
    does not make any difference whether the diode is
    connected or not.")
12 R2=100 //resistor R2 in ohm
13 tot_R=R1+R2
14 I1=Vaa/tot_R
15
16 printf("\n Current drawn from battery =%.2f A \n\n",
    I1)

```

Scilab code Exa 3.16 Breakdown Voltage Ranges

```

1 //Ex3_16 Pg-191
2 clc
3
4 Vz=9 //breakdown voltage in V
5 per=0.1 //10% tolerance
6 Tol=Vz*per
7 printf("Tolerance =%.1f V",Tol)
8 tol_high=Vz+Tol
9 tol_low=Vz-Tol //ranges in tolerance
10 printf("\n Range of breakdown voltage= %.1f to %.1f
    V",tol_low,tol_high)
11 // in the textbook the value 8.2 is wrong the

```

```

    correct value is 8.1
12 T1=25 //temperature T1 in degree celcius
13 T2=75 //temperature T2 in degree celcius
14 diff_temp=T2-T1 //chnage in temperature
15 Vzener=2*10^(-3) //zener voltage
16 fall_break_vol=Vzener*diff_temp //fall in breakdown
    voltage
17 new_break_vol=Vz-fall_break_vol //new break don
    voltage
18 printf("\n New break don voltage =%.1f V",
    new_break_vol)
19
20 range_high=tol_low-fall_break_vol
21 range_low=tol_high-fall_break_vol
22 printf("\n Range of breakdown voltage= %.1f to %.1f
    V",range_high,range_low)

```

Scilab code Exa 3.17 Capacitance

```

1 //Ex3_17 Pg-192
2 clc
3 C=20 //capacitance in pF
4 V=5 //supply voltage in V
5 K=C*sqrt(V)
6 C_V1=K/sqrt(V+1)
7 printf(" Capacitance for 1V increase =%.1f pF",C_V1)

```

Scilab code Exa 3.18 Voltage across each Diode

```

1 //Ex3_18 Pg-192
2 clc
3

```

```

4 printf("(a) The two diodes are connected in series
and hence the \n current I flows in D1 and D2.
Obviously, it is in forward \n direction through
D2 and in reverse direction through D1.\n Since
D2 diode is forward biased, V2 will be very small
and \n hence V1=(5-V2) will be very much larger
than Vt=0.026V.\n This means the current will be
equal to reverse saturation \n current I0. Now,
we consider diode D2. We have \n\n")
5 disp("      I = I0*(exp(V/Vt)-1)")
6 disp(" Putting I=I0 and V=V2, we have")
7 disp("      I0 = I0*(exp(V2/Vt)-1)")
8 disp("      exp(V2/Vt)-1 = 1")
9 Vt=0.026 //threshold voltage
10 V2=Vt*log(2)
11 V=5 //supply voltage in V
12 V1=V-V2 //value in textbook incorrect
13 printf("\n V2 = %.3f V",V2)
14 printf("\n V1 = %.3f V\n ",V1)
15 disp(" Effect of temperature : V2=Vt*ln(2) = kT*ln(2)
")
16 disp(" So V2 will increase with temperature ")
17 disp("(b) If Vz is 4.9V then D1 will breakdown. This
means V1=4.9V")
18 Vz=4.9 //breakdown voltage
19 V2=V-Vz
20 disp(" Now using I0=5*10^(-6)A and V2=0.1V, one
obtains")
21 I0=5*10^(-6) //current in ampere
22 I=I0*(exp(V2/Vt)-1)
23 printf("\n Current I=%.0f microA",I*10^6)

```

Scilab code Exa 3.19 LED Current

```
1 //Ex3_19 Pg-193
```

```

2  clc
3
4  disp("We have Iv=40*iD")
5  Iv=1 //luminous intensity
6  iD=Iv/(40*10^(-3)) //LED current
7  printf("LED current = %.0f mA",iD)

```

Scilab code Exa 3.20 AC Resistance

```

1  //Ex3_20 Pg-193
2  clc
3
4  disp("(a) At Id=10mA,")
5  V=25 //voltage in mV
6  Id=10 //current in mA
7  Rac=V/Id
8  //AC resistance (value in textbook is wrong)
9  printf("\n AC resistance Rac=%.1f ohm",Rac)
10
11 Id=20 //current in mA
12 Rac=V/Id
13 //AC resistance (value in textbook is wrong)
14 printf("\n AC resistance Rac=%.2f ohm",Rac)

```

Scilab code Exa 3.21 AC Resistance

```

1  //Ex3_21 Pg-194
2  clc
3
4  disp("We know that")
5  disp("  r_ac = dV/dI - 1/(dI/dV) = 1/((I0/KT)exp(V/
      KT))")
6  k=8.62*10^(-5)

```

```
7 T=300 //temperaturein K
8 kT=k*T
9 I0=10^(-6) //saturation current
10 V=150*10^(-3) //voltage forward biased
11 r_ac = 1/((I0/kT)*exp(V/kT))
12 //value of exp(0.15/0.02586)=330.45 and not the
    textbook value of 332.66
13
14 printf("\n The AC resistance = %.2f ohm",r_ac) //
    text book value wrong
```

Chapter 4

Diode Circuits

Scilab code Exa 4.1 Half Wave Rectifier

```
1 //Ex4_1 Pg-213
2 clc
3
4 Vrms=110 //rms volatage in V
5 Vm=Vrms/0.707 //peak source voltage
6 printf("Peak source voltage=%0.1f V",Vm) //textbook
   answer wrong
7
8 disp("(a) With an ideal diode ")
9 Vpout=Vm //peak output voltage
10 printf("\n Peak output voltage=%0.1f V",Vpout)
11 Vdc=Vm/%pi //Dc load voltage
12 printf("\n DC load voltage=%0.2f V \n",Vdc) //
   textbook answer wrong
13
14 disp("(b) With second approximation")
15 Vpin=Vm //peak input voltage
16 Vpout=Vpin-0.7
17 printf("\n Peak output voltage=%0.1f V",Vpout)
18 Vdc=Vpout/%pi //Dc load voltage
19 printf("\n DC load voltage=%0.1f V \n",Vdc) //
```


Scilab code Exa 4.2 No Load Voltage

```
1 //Ex4_2 Pg-214
2 clc
3
4 disp("      VR = (V_NoLoad - V_FullLoad)/V_FullLoad
      *100%")
5 disp("(a) VR = 0%")
6 V_FullLoad=20 //full load voltage
7 V_NoLoad=V_FullLoad//no load voltage
8 printf("\n V_FullLoad = V_NoLoad= %.0f V \n",
      V_NoLoad)
9
10 disp("(b) VR = 100%")
11 VR=100 //voltage regulation in %
12 V_NoLoad=(VR*V_FullLoad)/(100)+V_FullLoad
13 printf("\n V_NoLoad= %.0f V \n",V_NoLoad)
```

Scilab code Exa 4.3 AC Input Power

```
1 //Ex4_3 Pg-214
2 clc
3
4 disp("      Ratio of rectification or efficiency of
      halfwave rectifier ,")
5 disp("      n = 0.406 = DC power delivered to the load/
      AC input powerfrom transformer secondary ")
6 DC_power=500 //dcdc power delivered to the load
7 n=0.406 //efficiency
8 AC_in_power=DC_power/n //AC input powerfrom
      transformer secondary
```

```
9 printf("\n AC input powerfrom transformer secondary
   =%.0 f Watt", AC_in_power)
```

Scilab code Exa 4.4 Half Wave Rectifier

```
1 //Ex4_4 Pg-220
2 clc
3
4 Rl=3.5*10^(3) //resistance in k-ohm
5 rF=800 //secondary resistance in k-ohm
6 Vm=240 // input voltage
7 disp("(1)(a) Peak value of current flowing")
8 Im=Vm/(rF+Rl) //peak current
9 printf("      Im = %.2 f mA\n ", Im*10^3)
10
11 disp("(b) Average or DC current flowing")
12 Idc=Im/%pi //DC current
13 printf("      Idc = %.2 f mA\n ", Idc*10^3)
14
15 disp("(c) R.M.S value of current flowing")
16 Irms=Im/2 //rms current
17 printf("      Irms = %.2 f mA\n ", Irms*10^3)
18
19 disp("(2) DC output power")
20 Pdc=(Idc)^2*Rl //dc output power
21 printf("      Pdc = %.1 f Watt\n ", Pdc)
22
23 disp("(3) AC input power")
24 Pac=(Irms)^2*(rF+Rl)
25 printf("      Pac = %.2 f Watt\n ", Pac)
26
27 disp("(4) Efficiency of rectifier")
28 n=(Pdc/Pac)*100 //efficiency
29 printf("      n = %.2 f %%\n ", n)
```

Scilab code Exa 4.5 Bridge Rectifier

```
1 //Ex4_5 Pg-221
2 clc
3
4 Vr=0.7 //diodes voltage drop
5 Rl=820 //load resistor in ohm
6 Vin=40 //input voltage in V
7
8 disp("(1) Peak output volatge: Current flows
      through load only when two diodes conduct. While
      conducting, there is voltage drop across the
      diode.")
9 V_drop_2=2*Vr //voltage drop across 2 diodes
10 Vm=Vin-V_drop_2 //peak voltage
11 printf("\n      Vm = %.2 f V\n ",Vm)
12
13 disp("(2) Average output current")
14 Idc=(2*Vm/%pi)/Rl //average output current
15 printf("      Idc = %.0 f mA\n ",Idc*10^3)
16
17 disp("(3) Diode dissipation")
18 DD=Idc*Vr //Diode dissipation
19 printf("      = %.0 f mW\n ",DD*10^3)
```

Scilab code Exa 4.6 Half Wave Rectifier

```
1 //Ex4_6 Pg-222
2 clc
3
4 Vr=0.7 //voltage drop
5 Vi=120 //input voltage
```

```

6 disp("RMS value of secondary voltage")
7 V_sec=Vi/5 //RMS value of secondary voltage
8 printf("          = %.0 f V\n ",V_sec)
9
10 disp("Peak secondary voltage")
11 Vm=V_sec*sqrt(2) //Peak secondary voltage
12 printf("          = %.0 f V\n ",Vm)
13
14 disp("Peak inverse voltage of diode")
15 Vinv=-(Vm) //Peak inverse voltage of diode
16 printf("          = %.0 f V\n ",Vinv)
17
18 printf("\n Peak load voltage =%.0 f V\n ",Vm)
19
20 disp("DC load voltage")
21 Vdc=Vm/%pi //DC load voltage
22 printf("          = %.1 f V\n ",Vdc)
23
24 disp("Assuming second approximation")
25 disp("Vm' ' = Vm - Vr ")
26 disp("Peak load voltage")
27 Vm_dash=Vm-Vr //Peak load voltage
28 printf("          = %.1 f V\n ",Vm_dash)
29
30 disp("DC load voltage")
31 Vdc=Vm_dash/%pi //DC load voltage
32 printf("          = %.1 f V\n ",Vdc)

```

Scilab code Exa 4.7 Full Wave Rectifier

```

1 //Ex4_7 Pg-222
2 clc
3
4 Vi=120 //supply voltage n V
5 Rl=5*10^3 //load resistance

```

```

6
7 disp("Secondary RMS voltage")
8 Vrms=Vi/5 //Secondary RMS voltage
9 printf("          = %.0f V\n ",Vrms)
10
11 disp("Secondary pek voltage")
12 Vm=Vrms*sqrt(2) //Secondary pek voltage
13 printf("          = %.0f V\n ",Vm)
14
15 disp("  Half of the secondary voltage is input to
        the half section.")
16 disp("So input to the half section")
17 in=Vm/2 //input to the half section
18 printf("          = %.0f V\n ",in)
19
20 disp("Peak voltage across load")
21 printf("          = %.0f V\n ",in)
22
23 disp("    DC voltage across load = 17V. Since the
        capacitor gets charged up to peak value,")
24 disp("DC load current")
25 Vdc=in
26 Idc=Vdc/Rl //DC load current
27 printf("          = %.1f mA\n ",Idc*10^3)

```

Scilab code Exa 4.8 Full Wave Rectifier

```

1 //Ex4_8 Pg-227
2 clc
3
4 f=50 //frequency in Hz
5 C=100*10^(-6) //capacitance in F
6 Rl=2*10^3 //load resistance
7 Vrms=40 //rms secondary voltage
8

```

```

9 disp(" (a) Ripple factor for a full wave rectifier")
10 r=1/(4*sqrt(3)*f*C*Rl) //Ripple factor for a full
    wave rectifier
11 printf("          = %.3 f \n ",r)
12
13 disp(" (b) DC output voltage")
14 Vm=Vrms*sqrt(2)
15 Vdc=Vm/(1+(1/(4*f*C*Rl))) //DC output voltage
16 printf("          = %.1 f V \n ",Vdc)
17
18 disp(" (c) Percentage voltage regulation")
19 per=100/(4*f*C*Rl) //Percentage voltage regulation
20 printf("          = %.1 f %%\n ",per)

```

Scilab code Exa 4.9 Full Wave Rectifier

```

1 //Ex4_9 Pg-237
2 clc
3
4 Vrms=300 //rms voltage in V
5 f=60 //frequency
6 Idc=0.2 //load current
7 C=10 //shunt capacitor in microFarad
8
9 Vm=Vrms*sqrt(2) //peak voltage
10 Vdc=(2*Vm)/%pi //Dc voltage
11
12 disp(" Connected load")
13 Rl=Vdc/Idc //Connected load
14 printf("          Rl = %.0 f ohm = (955.6)*sqrt(2) ohm\n
    ",Rl)
15
16 disp(" Ripple factor in case of shunt capacitor
    filter ")
17 disp("          =2410/C*Rl")

```

```

18 r=2410/(C*R1) //ripple factor
19 printf("\n                = %.2f \n ",r)

```

Scilab code Exa 4.10 Full Wave Rectifier

```

1 //Ex4_10 Pg-238
2 clc
3
4 f=60 //frequency in Hz
5 C=100*10^(-6) //capacitance in F
6 R1=1*10^3 //load resistance
7
8 disp("Since the transformer is center tapped ,the
      rms value of voltage across half the secondary
      coil")
9 Vct=12.6 //voltage of center tapped transformer
10 Vrms=Vct/2 //rms voltage
11
12 disp("Peak voltage")
13 Vm=Vrms*sqrt(2) //peak voltage
14 printf("                = %.2f V\n ",Vm)
15
16 disp("(b) DC output voltage")
17 Vdc=Vm/(1+(1/(4*f*C*R1))) //DC output voltage
18 printf("                = %.2f V \n ",Vdc)
19
20 disp("Ripple factor in case of capacitor filter ")
21 disp("                =2410/C*R1")
22 r=2410/(100*R1)*100 //ripple factor
23 printf("\n                = %.1f %%\n ",r)

```

Scilab code Exa 4.11 Full Wave Rectifier

```

1 //Ex4_11 Pg-238
2 clc
3 Vdc=9 //dc voltage
4 Idc=100*10(-3) //dc load current
5 disp("Ripple factor with an L-C filter ,r=(0.83/LC)")
6 disp("          where L-> Henry ,C->microFarad")
7 gamm=0.02 //maximum ripple
8 LC=0.83/gamm
9 printf("          LC = %.1f \n ",LC) //let LC=42
10
11 disp("LOad connected to the filter ,")
12 RL=Vdc/Idc //load resistance in ohm
13 printf("          RL = %.0f ohm\n ",RL)
14
15 disp("Critical value of inductor ,")
16 Lk=RL/900 //Critical value of inductor
17 printf("          Lk = %.1f \n ",Lk)
18
19 disp("Capacitance")
20 LC=42 //rounding of 41.5 to 42
21 C=LC/Lk //capacitance in microFarad
22 printf("          C = %.0f uF\n ",C)

```

Scilab code Exa 4.12 Zener Current

```

1 //Ex4_12 Pg-245
2 clc
3
4 V=20 //source voltage
5 Vz=12 //zener voltage
6 Vr=V-Vz //voltage across resistor
7 Rs=330 //series resistance
8 disp("Voltage across resistor ")
9 printf("          = %.0f V \n ",Vr)
10

```



```

11 disp("Current through series resistor")
12 Iser=Vr/Rs //Current through series resistor
13 printf("          = %.1f mA \n ",Iser*10^3)
14
15 disp("Since Zener diode is in series with resistor ,
      current through it is equal to current flowing
      through resistor ,i.e 24.2mA ")

```

Scilab code Exa 4.13 Current and Wattage of Elements

```

1 //Ex4_13 Pg-245
2 clc
3
4 V=20 //source voltage in V
5 Vz=12 //zener voltage in V
6 Vs=V-Vz //voltage across resistor in V
7 Rs=330 //series resistance in ohm
8 RL=1.5*10^3 //load resistance in ohm
9 disp("Voltage across resistor ")
10 printf("          = %.0f V \n ",Vr)
11
12 disp("(1) Current through series resistor Is")
13 Is=Vr/Rs //Current through series resistor
14 printf("          Is = %.1f mA \n ",Is*10^3)
15
16 disp("(2) Current through series load Il")
17 VL=Vz //voltage across load
18 IL=VL/RL //Current through series load
19 printf("          IL = %.0f mA \n ",IL*10^3)
20
21 disp("(3) Current through zener diode")
22 Iz=Is-IL //Current through zener diode
23 printf("          IL = %.1f mA \n ",Iz*10^3)
24
25 disp("(4) Respective wattage of elements used")

```

```

26 disp("(a) Series resistor -> W=Is*Vs")
27 W=Vs*Is //wattage of series resistor
28 printf("          = %.1f mW \n ",W*10^3)
29
30 disp("(b) Zener diode -> W=Iz*Vz")
31 W=Vz*Iz //wattage of zener diode
32 printf("          = %.1f mW \n ",W*10^3)
33
34
35 disp("(b) Load resistor -> W=IL*VL")
36 W=VL*IL //wattage of zener diode
37 printf("          = %.0f mW \n ",W*10^3)

```

Scilab code Exa 4.14 Zener Diode

```

1 //Ex4_14 Pg-246
2 clc
3
4 RL=1*10^3 //load resistance in ohm
5 Rs=270 //series resistor in ohm
6 Vs=18 //supply voltage in V
7 vz=10 //xener voltage
8
9 disp("Applying Thevenin''s theorem, Thevenin voltage
      across the zener diode")
10 Vth=(RL/(RL+Rs))*Vs //Thevenin voltage
11 printf("\n          Vth = %.1f V \n ",Vth)
12
13 disp("Thus Vth is greater than Vz(zener voltage),i.e
      14.2 >10. So Zener diode is operating in the
      breakdown voltage.")

```

Scilab code Exa 4.15 Voltage Regulator Circuit

```

1 //Ex4_15 Pg-246
2 clc
3
4 IL1=10*10(-3)
5 IL2=20*10(-3) //IL1,IL2 range of load current in A
6 Vin=20 //supply voltage in V
7 Iz=6*10(-3) //zener current in A
8 Vz=15 //zener voltage in V
9
10 disp("Average load current")
11 IL=(IL1+IL2)/2 // Average load current
12 printf("\n      IL = %.0f mA \n ",IL*103)
13
14 disp("Total current entering the circuit")
15 Is=IL+Iz //current entering the circuit
16 printf("\n      Is = %.0f mA \n ",Is*103)
17
18 disp("Series resistor")
19 Rs=(Vin-Vz)/Is //Series resistor in ohm
20 printf("\n      Rs = %.0f ohm \n ",Rs)
21
22 disp("Power rating of resistor")
23 Vs=Vin-Vz
24 P=(Vs2)/Rs //Power rating of resistor
25 printf("\n      P = %.1f W \n ",P)

```

Chapter 5

Transistors

Scilab code Exa 5.1 Current component of a Transistor

```
1 //Ex5_1 Pg-278
2 clc
3
4 alpha_dc=0.97 //transistor current gain
5 ICBO=10*10^(-6) //collector to base leakage current
   in A
6 Ib=50*10^(-6) //base current in A
7
8 Ic=((alpha_dc*Ib)/(1-alpha_dc))+(ICBO/(1-alpha_dc))
   //collector current
9 printf(" Collector current = %.2f mA \n",Ic*10^3)
10
11 Ie=Ic+Ib //emitter current
12 printf(" Emitter current = %.0f mA",Ie*10^3)
```

Scilab code Exa 5.2 Transistor Components

```
1 //Ex5_2 Pg-279
```

```

2  clc
3
4  Ic=5.255*10^(-3) //collector current in A
5  Ib=100*10^(-6) //base current in A
6  ICB0=5*10^(-6) //collector to base leakage current
   in A
7
8  alpha_dc=(Ic-ICB0)/(Ib+Ic) //common current gain
   factor
9  printf("Common current gain factor alpha_dc = %.2f",
   alpha_dc)
10
11 Beta=alpha_dc/(1-alpha_dc) //Dc emitter current gain
   factor value in text book is wrong
12 printf("\n Dc emitter current gain factor beta = %.2
   f",Beta)
13
14 Ie=Ic+Ib //emitter current value in text book wrong
15 printf("\n Emitter current = %.3f mA",Ie*10^3)

```

Scilab code Exa 5.3 Transistor Components

```

1  //Ex5_3 Pg-279
2  clc
3
4  Ic=12.427*10^(-3) //collector current in A
5  Ib=200*10^(-6) //base current in A
6  ICB0=7*10^(-6) //collector to base leakage current
   in A
7
8  Beta=(Ic-ICB0)/(Ib+ICB0) //Dc emitter current gain
   factor (value in texbook is wrong)
9  printf("\n Dc emitter current gain factor beta = %.0
   f",Beta)
10

```

```
11 Ie=Ic+Ib //emitter current
12 printf("\n Emitter current = %.1f mA",Ie*10^3)
13
14 alpha_dc=(Ic-ICB0)/(Ib+Ic) //common current gain
    factor
15 printf("\n Common current gain factor alpha_dc = %.2
    f",alpha_dc)
16
17 Ib=150*10^(-6) //new base current
18 Ic=Beta*Ib+(Beta+1)*ICB0 //collector current (value
    in textbook is wrong)
19 printf("\n Collector current = %.3f mA \n",Ic*10^3)
```

Chapter 6

Thermal Stability

Scilab code Exa 6.1 Collector Base Junction Temperature

```
1 //Ex6_1 Pg-333
2 clc
3
4 Ta=25 //ambient temperature in degree celsius
5 tetha=10 //thermal resistance
6 Pd=2 //power dessoriated in transistor
7
8 Tj=Ta+tetha*Pd //collector base junction transistor
   temperature
9 printf(" \n Collector base junction transistor
   temperature = %.0f degree celcius \n",Tj)
10
11 disp("    tetha=10 degree celcius/watt means for
   every watt consumed its temperature will rise by
   10 degree celcius")
12
13 printf("\n While using a transistor ,we must keep in
   mind that it must \n not reach a condition called
   thermal runaway. The heat \n released at
   collector base junction must not exceed the rate
   \n at which heat can dessoriated under steady
```

```

    state. For this ,\n ")
14
15 disp("(del_Pd/del_Tj) < (1/tetha)")
16
17 disp("This is the relation for thermal stability.")

```

Scilab code Exa 6.2 Dissipation Capacity of a Transistor

```

1 //Ex6_2 Pg-335
2 clc
3
4 printf("Draw a vertical line from temperature axis
    at 50 degree \n celcius to cut the 71 degree
    celcius line. Join the point of \n intersection P
    through a horizontal line at Y-axis. The point \
    n where it intersects Y-axis gives the value of
    permissible \n dissipation equal to 45%% of
    maximum rating. \n")
5
6 per=.45 //permissible dissipation in percentage
7 max_diss=165 //maximum dissipation
8 diss_cap=per*max_diss //dissipation capability
9 disp("The dissipation capability at 50 degree
    celcius")
10
11 printf("                = %.0f mW \n ",
    diss_cap)
12
13 disp("    Its value ranges from (0.2) degree
    celcius/watt to (1000) degree celcius/watt for a
    transistor that has an efficient heat sink")
14
15 disp("                Tj = Ta + tetha*Pd")
16
17 disp("    The above equation reflects that collector-

```


junction temperature T_j of the transistor will be higher than ambient temperature T_a by an amount equal to the product of θ and P_d .”)

Scilab code Exa 6.3 Fixed Bias Circuit

```
1 //Ex6_3 Pg-336
2 clc
3
4 Rb=200*10^(3) //base resistance in ohm
5 Vcc=10 //supply voltage in V
6 Vbe=0.7 //voltage drop in V
7 Rl=2*10^(3) //load resistor in ohm
8 Beta=50 //transistor gain
9
10 disp(" If Beta=50")
11 Ib=(Vcc-Vbe)/Rb //base current in A
12 Ic=Beta*Ib //collector current
13 Vce=Vcc-Ic*Rl //collector emitter voltage
14 printf("\n The operating point coordinates are [%.2 f
      V, %.2 f mA]\n ",Vce,Ic*10^3)
15
16 disp(" If Beta=60")
17 Beta2=60 //second transistor gain
18 Ic2=Beta2*Ib //collector current
19 Vce2=Vcc-Ic2*Rl //collector emitter voltage
20 printf("\n The operating point coordinates are [%.2 f
      V, %.2 f mA]\n ",Vce2,Ic2*10^3)
```

Scilab code Exa 6.4 Collector to Base Bias Circuit

```
1 //Ex6_4 Pg-335
2 clc
```

```
3
4 Rb=330*10^(3) //base resistance in ohm
5 Vcc=15 //supply voltage in V
6 Vbe=0.7 //voltage drop in V
7 Rl=3.3*10^(3) //load resistor in ohm
8 Beta=60 //transistor gain
9
10 Ib=(Vcc-Vbe)/Rb //base current in A
11 Ic=Beta*Ib //collector current (value in textbook is
    wrong)
12 Vce=Vcc-(Ic+Ib)*Rl
13 printf("\n The collector emitter voltage = %.2f V\n
    ",Vce)
14 //collector emitter voltage (value in textbook is
    wrong)
```

Chapter 7

Amplifiers

Scilab code Exa 7.1 Coupling

```
1 //Ex7_1 Pg-369
2 clc
3
4 disp("Refer to the figure 7.19")
5 disp("For good coupling")
6 disp("          Xc <= 0.1*R")
7 R=10*10^(3) //resistor R in ohm
8 C=68*10^(-6) //capacitance in Farad
9 f=1/(2*%pi*C*0.1*R)
10 disp("Lowest frequency ,f")
11 printf("          = %.2 f Hz" ,f)
12 disp("or lowest frequency is approximately 3 Hz")
```

Scilab code Exa 7.2 Bypassing

```
1 //Ex7_2 Pg-369
2 clc
3
```

```

4 disp(" Refer to the figure 7.20")
5 disp(" For good coupling")
6 disp("          Xc <= 0.1*R")
7 R=10*10^(3) //resistor R in ohm
8 C=220*10^(-6) //capacitance in Farad
9 f=1/(2*%pi*C*0.1*R)
10 disp(" Lowest frequency ,f")
11 printf("          = %.2 f Hz",f)
12 disp(" or lowest frequency is approximately 1 Hz")

```

Scilab code Exa 7.3 Small Signal Amplifiers

```

1 //Ex7_3 Pg-369
2 clc
3
4 Beta=250 //transistor gain
5 Vcc=15 //supply voltage
6 R1=2000 //resistor R1 in ohm
7 R2=470 //resistor R2 in ohm
8 Vce=0.7 //voltage drop in V
9 Re=220 //emitter resistor in ohm
10
11 Vb=(Vcc*R2)/(R1+R2) //base voltage in V
12 disp("(1) Base voltage Vb")
13 printf("          = %.2 f V",Vb)
14 Ve=Vb-Vce //emitter voltage in V
15 disp(" Emitter voltage Vb")
16 printf("          = %.2 f V",Ve)
17 Ie=Ve/Re //emitter current
18 disp(" Emitter current ")
19 printf("          Ie = %.3 f*10^-2 A",Ie*10^2)
20 disp(" For small signal operation , ie <= 0.1*Ie")
21 ie=0.1*Ie
22 printf("          =%.3 f mA \n",ie*10^3)
23

```

```

24 disp("(2) AC emitter diode resistance =25mV/ie")
25 Re_ac=25*10^(-3)/ie //AC emitter diode resistance
26 printf("                =%.0 f ohm \n",
        Re_ac)
27
28 disp("(3) Z'vm = Beta*r'e")
29 Re_ac=26 //AC emitter diode resistance assumed 26
        ohm not 25.53 ohm
30 Zvm=Beta*Re_ac
31 printf("                =%.0 f ohm",Zvm)

```

Scilab code Exa 7.4 Amplifier Circuit

```

1 //Ex7_4 Pg-370
2 clc
3
4 Beta=100 //transistor gain
5 Vcc=10 //supply voltage
6 R1=10*10^(3) //resistor R1 in ohm
7 R2=2200 //resistor R2 in ohm
8 Vce=0.7 //voltage drop in V
9 Re=2000 //emitter resistor in ohm
10 Rs=600 //source resistor in ohm
11
12 Vb=(Vcc*R2)/(R1+R2) //base voltage in V
13 disp("Base voltage Vb")
14 printf("                = %.1 f V",Vb)
15 Ve=Vb-Vce //emitter voltage in V
16 disp("Emitter voltage Vb")
17 printf("                = %.1 f V",Ve)
18 Ie=Ve/Re //emitter current
19 disp("Emitter current")
20 printf("                = %.2 f mA",Ie*10^3)
21 disp("AC emitter diode resistance =25mV/ie")
22 re=25*10^(-3)/Ie //AC emitter diode resistance

```

```

23 printf("                =%.0 f ohm \n",
        re)
24 rc=((3.6*10)/(3.6+10))*10^(3) //Collector dived
        resistance
25 A=rc/re //voltage gain(value in text book is wrong)
26 disp("Voltage gain A")
27 printf("                = %.0 f",A)
28 zin_1=((10*2.2)/(10+2.2))
29 zin=((zin_1*Beta*A)/(zin_1+(Beta*A)))*1000
30 disp("Zin stage")
31 printf("                = %.3 f kohm",zin*10^-3)
32 Vin=(zin/(Rs+zin))*10^(-3) //input voltage (value in
        text book is wrong)
33 disp("Input voltage")
34 printf("                = %.2 f mV",Vin*10^3)
35 Vout=A*Vin //output voltage (value in textbook is
        wrong)
36 disp("Output voltage")
37 printf("                = %.2 f mV",Vout*10^3)

```

Scilab code Exa 7.5 RC Coupled CE Amplifier

```

1 //Ex7_5 Pg-371
2 clc
3
4 hfe=50 //current gain
5 Rl=10*10^(3) //load resistor in ohm
6 Rs=500 //source resistor in ohm
7 hie=10^(3) //input resitance in ohm
8
9 A=hfe*Rl/(Rs+hie) //volatge gain
10 printf("Voltage gain = %.1 f \n",A)
11 Vs=0.02 //source voltage in V
12 Vout=A*Vs //output voltage
13 printf("Output voltage = %.2 f V",Vout)

```

Scilab code Exa 7.6 Audio Amplifier

```
1 //Ex7_6 Pg-372
2 clc
3
4 Vout=5 //output voltage
5 Vin=0.5 //input voltage
6 Ri=10*10^(3) //input resistance in ohm
7 Ro=10 //output resistance
8
9 A=Vout/Vin //voltage gain
10 printf(" Voltage gain =%.0f \n",A)
11 Pi=Vin^2/Ri //input power
12 Po=Vout^2/Ro //output power
13 Pow_gain=10*(log10(Po)-log10(Pi)) //power gain
14 printf(" Power gain(in decibel) = %.0f dB \n\n",
    Pow_gain)
15
16 disp("When Ri=Ro")
17 Ri=Ro
18 A=Vout/Vin //voltage gain
19 Pi=Vin^2/Ri //input power
20 Po=Vout^2/Ro //output power
21 Pow_gain=10*(log10(Po)-log10(Pi)) //power gain
22 printf(" Power gain(in decibel) = %.0f dB",Pow_gain)
```

Scilab code Exa 7.7 RC Coupled CE Amplifier

```
1 //Ex7_7 Pg-372
2 clc
3
```

```

4 Rl=2*10^(3) //load resistance in ohm
5 Ri=500 //input resistance in ohm
6 C=2*10^(-6) //capacitor in microFarad
7
8 f=(1/(2*%pi*C*(Rl+Ri))) //textbook answer is wrong
9 printf("Lowest cut-off frequency = %.0f Hz",f)

```

Scilab code Exa 7.8 RC Coupled Amplifier

```

1 //Ex7_8 Pg-372
2 clc
3
4 Rl=20*10^(3) //load resistance in ohm
5 Ri=5000 //input resistance in ohm
6 f=33 //lower cut-off frequency in Hz
7 f2=33*10^(3) //higher cut-off frequency
8
9 C=(1/(2*%pi*f*(Rl+Ri))) //coupling capacitance (
    value in textbook is wrong)
10 printf("Coupling Capacitor = %.1f uF \n",C*10^6)
11
12 Req=(Rl*Ri)/(Rl+Ri) //equivalent resistance
13 Cs=1/(2*%pi*f2*Req) //shunting capacitance (value in
    textbook is wrong)
14 printf("Coupling Capacitor = %.0f uF \n",Cs*10^12)

```

Scilab code Exa 7.9 FET Amplifier

```

1 //Ex7_9 Pg-373
2 clc
3
4 Rd=3000 //source resistance in ohm
5 Rl=5000 //load resistance in ohm

```



```
6 Req=Rd*Rl/(Rd+Rl) //equivlent resistance
7
8 gm=4500*10^(-6) //voltage gain in microMhos
9 Av=(-1)*gm*Req //voltage amplification
10 printf(" Voltage Amplification = %.2f \n",Av)
11 Vi=100*10^(-3) //input voltage
12 Vout=abs(Av)*Vi //output voltage (value in textbook
    is wrong)
13 printf(" Output voltage = %.1f V",Vout)
```

Chapter 8

Feedback Amplifiers

Scilab code Exa 8.1 Voltage Gain of Amplifier

```
1 //Ex8_1 pg-434
2 clc
3
4 A=120 //amplification gain
5 Vi=50*10^(-3) //input voltage
6 Beta=0.1 //feedback factor
7
8 Vo= A*Vi //output voltage without feedback
9 printf("Input signal = %.2f V \n",Vo)
10
11 Vs=Vi-Beta*Vo
12 //input signal +ve output because of -ve feedback (
    value in textbook is wrong)
13 printf(" Input signal = %.2f V \n",abs(Vs))
14
15 vg=A/(1+Beta*A) //voltage gain
16 printf(" Gain after feedback = %.1f \n",vg)
17
18 fb=(-1)*20*log10(1+(Beta*A))
19 printf(" Feedback (db)= %.3f \n",fb)
```

Scilab code Exa 8.2 Bandwidth of Amplifier

```
1 //Ex8_2 pg-435
2 clc
3
4 ff=4 //feedback factor
5 BW=6*10^(6) //bandwidth in Hz
6
7 BW_fb=BW*(1+ff) //Bandwidth with feedback factor(
  since Beta is +ve)
8 printf("Bandwidth with feedback factor = %.0f MHz",
  BW_fb*10^-6)
```

Scilab code Exa 8.3 Negative Feedback Factor

```
1 //Ex8_3 pg-435
2 clc
3
4 openA=60000 //open loop gain
5 closeA=10000 //closed loop gain
6 Beta=((openA/closeA)-1)/closeA
7 printf("Negative Feedback factor = %.4f \n",Beta)
8 BA=Beta*openA //value of Beta*A
9 printf(" Beta*A = %.0f",BA)
```

Scilab code Exa 8.4 Feedback Voltage Gain

```
1 //Ex8_4 pg-435
2 clc
```

```

3
4 Df=0.5/100 //distortion after negative feedback
5 D=8/100 //harmonic distortion
6
7 BA=D/Df-1 //value of Beta*A
8 A=200
9 Beta=BA/A //feedback factor
10 printf("Feedback factor = %.3f \n",Beta)
11 Af=A/(1+BA) //Gain after -ve feedback
12 printf("Gain after negative feedback = %.1f",Af)

```

Scilab code Exa 8.5 Feedback Voltage Gain

```

1 //Ex8_5 pg-436
2 clc
3
4 A=100 //voltage gain
5 per=10/100 //percentage of negative feedback applied
6 BA=A*per //value of Beta*A
7 Af=A/(1+BA) //gain after negative feedback
8 printf("Decrement in distortion ,D-Df = D-(D/(1+Beta*
    A)) \n")
9 printf("                = %.1f\n ",Af)
10 per_dis=(BA/(1+BA))*100 //percentage change in
    distortion
11 printf("Percentage change in distortion = %.0f %% \n
    ",per_dis)
12 red=100-per_dis //reduction
13 printf(" Therefore reduction is = %.0f %%",red)

```

Scilab code Exa 8.6 Feedback Amplifier

```

1 //Ex8_6 pg-436

```

```

2  clc
3
4  A=50 //voltage gain
5  per=10/100 //percentage of negative feedback applied
6  BA=per*A //value of Beta*A
7  Af=A/(1+BA) //gain after negative feedback
8  printf("(1) Voltage gain after negative feedback \n"
   )
9  printf("                = %.2 f\n ",Af)
10
11 A=50 //voltage gain
12 per=5/100 //percentage of negative feedback applied
13 BA=per*A //value of Beta*A
14 Af1=A/(1+BA) //gain after negative feedback
15 printf("(2) Voltage gain after negative feedback \n"
   )
16 printf("                = %.1 f\n ",Af1)
17 disp("So the new gain is not double the previous
   case")
18 disp("    The difference between expected value and
   actual value of gain obtained is")
19 diff_value=2*Af-Af1
20 printf("                = %.2 f",
   diff_value)
21
22 printf("\n\n(3) To have the gain double of case(1) i
   .e 16.66,let the \n feedback introduced be Beta(
   assuming negative feedback)")
23 Af=16.66 //voltage gain with negative feedback
24 A=50 //voltage gain
25 Beta=((A/Af)-1)/A //feedback in percentage
26 printf("\n                Beta = %.2 f",Beta)

```

Chapter 9

Negative Feedback Amplifier using Opamp

Scilab code Exa 9.1 VCVS Amplifier

```
1 //Ex9_1 Pg-475
2 clc
3
4 Ao1= 250000 //open loop gain
5 fo1=160 //open loop frequency in HZ
6 Ac1=50 //close loop gain
7
8 fc1=Ao1/Ac1*fo1 //close loop frequency in Hz
9 printf("Close loop Bandwidth = %.0f kHz",fc1*10^-3)
```

Scilab code Exa 9.2 ICVS Amplifier

```
1 //Ex9_2 Pg-475
2 clc
3
4 Ao1= 50000 //open loop gain
```

```

5 fol=14 //open loop frequency in HZ
6 fcl=(Aol+1)*fol //close loop frequency in Hz
7 printf("Close loop Bandwidth = %.0f kHz",fcl*10^-3)

```

Scilab code Exa 9.3 ICIS Amplifier

```

1 //Ex9_3 Pg-475
2 clc
3
4 Aol_Beta_1= 2500 //open loop gain
5 fol=20 //open loop frequency in HZ
6 fcl=Aol_Beta_1*fol //close loop frequency in Hz
7 printf("Close loop Bandwidth = %.0f kHz",fcl*10^-3)

```

Scilab code Exa 9.4 VCVS Amplifier

```

1 //Ex9_4 Pg-475
2 clc
3
4 funi=1*10^(6) //unity frequency in Hz
5 Sr=0.5/10^(-6) //slew rate in V/sec
6 Acl=10 //close loop gain
7
8 fcl=funi/Acl //close loop frequency in Hz
9 printf("(1) Close loop Bandwidth = %.0f kHz \n",fcl
10         *10^-3)
11
12 Vp_max=Sr/(2*%pi*fcl) //output peak value
13 printf("(2) Peak value of output = %.3f V \n",
14         Vp_max)

```

Scilab code Exa 9.5 VCVS Amplifier

```
1 //Ex9_5 Pg-475
2 clc
3
4 Aol= 88 //open loop gain in db
5 R1=2.7*10^(3) //resistor R1 in ohm
6 R2=68*10^(3) //resistor R2 in ohm
7
8 Beta=R1/(R1+R2) //Feedback fraction
9 printf("Feedback fraction = %.3f \n",Beta)
10
11 Acl=1/Beta //ideal closed loop gain
12 printf(" Ideal closed loop gain = %.2f \n",Acl)
13
14 Aol=10^(88/20) //open loop gain
15 Acl=Aol/(1+Beta*Aol) //exact closed loop voltage
    gain
16 printf(" Exact closed loop voltage gain = %.2f",Acl)
```

Scilab code Exa 9.6 VCVS Amplifier

```
1 //Ex9_6 Pg-476
2 clc
3
4 Aol=20000 //open loop gain
5 R1=100 //resistor R1 in ohm
6 R2=7.5*10^(3) //resistor R2 in ohm
7 Rin=3*10^(6) //input resistor in ohm
8 Rcm=500*10^(6)
9
10 Beta=R1/(R1+R2) //Feedback fraction
11 printf("Feedback fraction = 1\\76 = %.5f \n",Beta)
12
13 Req=(Rin*Rcm)/(Rin+Rcm) //equivalent resistance of
```


Rin and Rcm

```
14 Zc1=(1+Beta*Ao1)*Req //closed loop input impedance (  
    textbook answer is wrong)  
15 printf(" Closed loop input impedance = %.0f Mohm",  
    Zc1*10^-6)
```

Scilab code Exa 9.7 ICIS Amplifier

```
1 //Ex9_7 Pg-477  
2 clc  
3  
4 R1=1.8 //resistor R1 in ohm  
5 R2=1.5*10^(3) //resistor R2 in ohm  
6 Iin=1*10^(-3) //input current in A  
7  
8 Ai=1+(R2/R1) //Current gain  
9 printf(" Current gain = %.0f \n",Ai)  
10  
11 I1=Ai*Iin //Output current  
12 printf(" Output current = %.0f mA pp",I1*10^3)
```

Scilab code Exa 9.8 VCIS Amplifier

```
1 //Ex9_8 Pg-477  
2 clc  
3  
4 R1=2.7 //resistor R1 in ohm  
5 R2=1//resistor R2 in ohm  
6 Vin=0.5 //input voltage in V  
7  
8 Io=Vin/R1 //output current  
9 printf("(1) Output current = %.0f mA \n",Io*10^3)  
10
```

```
11 P=Io^2*R2 //load power
12 printf(" (2) Load power = %.1 f mW \n",P*10^3)
13
14 R2=2 // new load resistor R2 in ohm
15 P=Io^2*R2 //load power
16 printf(" (2) Load power = %.1 f mW",P*10^3)
```

Chapter 10

Operational Amplifier

Scilab code Exa 10.1 Common Mode Voltage Gain

```
1 //Ex10_1 Pg-490
2 clc
3
4 Rc=1*10^(6) //collector resisstor in ohm
5 Re=2*10^(6) //emitter resistor in ohm
6 Vin=1*10^(-3) //input voltage in V
7
8 Acm=Rc/Re //Common moce voltage gain
9 printf("Common moce voltage gain = %.1f \n",Acm)
10
11 Vo=Acm*Vin //output voltage
12 printf(" Output voltage = %.1f mV",Vo*1e3)
13
14 disp("Thus a differential amplifier in common mode
      attenuates the input signal rather than
      amplifying it")
```

Scilab code Exa 10.2 Differential Amplifier

```

1 //Ex10_2 Pg-491
2 clc
3
4 A=150 //voltage gain
5 Acm=0.5 //common mode voltage gain
6 Vin=1*10^(-3) //input voltage in V
7
8 Vo=A*Vin //output voltage
9 printf(" Amplified output voltage = %.2 f V \n",Vo)
10
11 Vo=Acm*Vin //output voltage
12 printf(" Attenuated output voltage = %.1 f mV",Vo*1e3
    )

```

Scilab code Exa 10.3 Opamp Operating Circuit

```

1 //Ex10_3 Pg-517
2 clc
3
4 R1=10*10^(3) //resistor R1 in ohm
5 Rf=50*10^(3) //feedback resistor in ohm
6 Vin=10*10^(-3) //input voltage in V
7 Ro=5000 //load resistor in ohm
8
9 disp("A'' = Vo/Vi = (-1)*Rf/R1*(1+1/A*(1+Rf/R1))^-1
    ")
10 A=5000
11 Vo=Vin*(Rf/R1)/(1+1/A*(1+Rf/R1)) //output voltage
12 printf("\n When gain A=%.0 f",A)
13 printf("\n Amplified output voltage = %.1 f mV \n",
    Vo*1e3)
14
15 A=10000
16 Vo=Vin*(Rf/R1)/(1+1/A*(1+Rf/R1))
17 printf("\n When gain A=%.0 f",A)

```

```

18 printf(" \n Amplified output voltage = %.2 f mV \n",
        Vo*1e3)
19
20 A=5000
21 Rout=Ro/(1+A*R1/Rf) //load resistance
22 printf(" \n Ro'' = %.3 f ohm \n",Rout)
23
24 A=10000
25 Rout=Ro/(1+A*R1/Rf) //load resistance
26 printf(" \n Ro'' = %.3 f ohm \n",Rout)

```

Scilab code Exa 10.4 Opamp Operating in Inverting Mode

```

1 //Ex10_4 Pg-518
2 clc
3
4 R1=1.5*10^(3) //resistor R1 in ohm
5 Rf=75*10^(3) //feedback resistor in ohm
6 Vin=10*10^(-3) //input voltage in V
7 funi=1*10^(6) //unity frequency in Hz
8
9 Acl=(-1)*Rf/R1 //closed loop gain
10 printf("Magnitude of Closed loop gain = %.0 f \n",abs
        (Acl))
11
12 fcl=funi/abs(Acl) //closed loop frequency
13 printf(" Closed loop frequency = %.0 f kHz \n",fcl*1e
        -3)
14
15 Vout=abs(Acl)*Vin //output voltage
16 printf(" Output voltage = %.1 f mV pp",Vout*1e3)

```

Scilab code Exa 10.5 Closed Loop Voltage Gain

```

1 //Ex10_5 Pg-518
2 clc
3
4 R1=2*10^(3) //resistor R1 in ohm
5 Rf=0 //feedback resistor in ohm
6
7 disp("(1) When resistor 100 k-ohm is in zero
      position")
8 A=1+Rf/R1 //gain
9 printf(" Gain = %.0f \n",A)
10
11 Rf=100*10^(3)
12 disp("(1) When resistor 100 k-ohm is in maximum
      position")
13 A=1+Rf/R1 //gain
14 printf(" Gain = %.0f",A)

```

Scilab code Exa 10.6 Opamp Operating in Inverting Mode

```

1 //Ex10_6 Pg-519
2 clc
3
4 R1=50*10^(3) //resistor R1 in ohm
5 Rf=300*10^(3) //feedback resistor in ohm
6 Vin=1 //input voltage in V
7
8 disp("In the inverting mode, voltage gain is ")
9 disp("A' ' = Vo/Vi = (-1)*Rf/R1*(1+1/A*(1+Rf/R1))^-1
      ")
10 A=10000
11 Vo=(-1)*Vin*(Rf/R1)/(1+1/A*(1+Rf/R1)) //output
      voltage
12 printf(" \n Amplified output voltage = %.3f V \n",Vo
      )

```

Scilab code Exa 10.7 Power Bandwidth of Opamp

```
1 //Ex10_7 Pg-519
2 clc
3
4 Sr=15/1e-6 //slew rate in V/sec
5 Vp=10 //peak output voltage
6
7 fmax=Sr/(2*%pi*Vp) //Power Bandwidth
8 printf("Power Bandwidth = %.0f kHz", fmax*1e-3)
```

Scilab code Exa 10.8 Resistances of Opamp

```
1 //Ex10_8 Pg-519
2 clc
3
4 A=5000 //voltage gain
5 Ri=10000 //input resistor in ohm
6 Ro=100 //load resistor in ohm
7 Rf=0 //feedback resistor in ohm
8
9 Rin=A*Ri/(1+Rf/Ri) //input resistance of buffer
   circuit
10 printf("Input resistance of buffer circuit = %.0f*1
   e7 ohm \n", Rin*1e-7)
11
12 Rout=Ro/A*(1+Rf/Ri) //output resistance of buffer
   circuit
13 printf(" Output resistance of buffer circuit = %.2f
   ohm", Rout)
```

Chapter 11

Power Amplifiers

Scilab code Exa 11.1 RC Coupled Amplifier

```
1 //Ex11_1 Pg-536
2 clc
3
4 Vcc=15 //supply voltage in V
5 R1=2*10^(3) //resistor R1 in ohm
6 R2=470 //resistor R2 in ohm
7 Rc=680 //collector resistor in ohm
8 Rl=2.7*10^(3) //load resistor in ohm
9 Re=220 //emitter resistor
10
11 Idc=Vcc/(Rc+Re) //saturation current
12 printf(" (1) Idc_sat = %.1f mA \n", Idc*1e3)
13
14 DCload=Rc //Dc load resistance
15 printf(" (2) DC load = %.0f ohm \n", DCload)
16
17 ACload=Rc*Rl/(Rc+Rl) //Ac load resistance
18 printf(" (3) AC load = %.0f ohm \n", ACload)
19
20 Vb=R2/(R1+R2)*Vcc //base voltage
21 Icq=(Vb-0.7)/Re //collector current
```



```

22 printf("(4) Icq = %.1f mA \n",Icq*1e3)
23 //answer in the book is wrong
24
25 Vc=Vcc-Icq*Rc //collector emitter voltage
26 Vceq=Vc-Icq*Re
27 printf("(5) Vceq = %.1f V \n",Vceq)
28 //answer in the book is wrong
29
30 Pac=Vcc^2/(8*Rl) //ac power
31 Pdc=Vcc*Idc //dc power
32 n=Pac/Pdc*100 //efficiency
33 printf(" Efficiency = %.0f %%",n)

```

Scilab code Exa 11.2 Class B Transformer Coupled Amplifier

```

1 //Ex11_2 Pg-551
2 clc
3
4 Po=4 //power in watts
5 n=80/100 //transformer efficiency in percentage
6 Vcc=30 //supply voltage
7
8 Pout=Po/n //effective power
9 printf(" Effective power to be transfered = %.0f W\n"
    ,Pout)
10
11 disp("Impedance seen when ""looking into"" the whole
    winding of centertapped transformer ")
12 Vp=Vcc //peak voltage
13 Rload=Vp^2/(2*Pout)
14 Rload_4=4*Rload //effective load
15 printf("\n Effective load = %.0f ohm \n",Rload_4)
16
17 disp(" Transformer specification Po=4W,RL=16ohm,RL"
    =360ohm")

```

```

18
19 Vce=2*Vcc //Maximum transistor voltage
20 printf("\n Maximum transistor voltage = %.0 f V\n",
    Vce)
21
22 Ip=2*Pout/Vp //Maximum transistor current
23 Ic=Ip
24 printf("\n Maximum transistor current = %.0 f mA \n",
    Ip*1e3)
25 // answer in the book is different due to
    approximate value
26 printf("\n Transformer specification Vce=60V,Ic=%.0 f
    mA",Ic*1e3)
27 // answer in the book is different due to
    approximate value

```

Scilab code Exa 11.3 Tuned Amplifier

```

1 //Ex11_3 Pg-564
2 clc
3
4 L=2*10^(-6) //inductance in H
5 C=220*10^(-12) //capacitance in F
6
7 f0=1/(2*%pi*sqrt(L*C)) //resonant frequency (
    textbook answer is wrong)
8 printf("Resonant frequency = %.1 f MHz \n",f0*1e-6)
9
10 Q=125 //quality factor
11 BW=f0/Q //Bandwidth (textbook answer is wrong)
12 printf(" Bandwidth = %.0 f kHz",BW*1e-3)

```

Scilab code Exa 11.4 RC Coupled Amplifier

```

1 //Ex11_4 Pg-564
2 clc
3
4 Vcc=10 //supply volage in V
5 Rc=3600 //collector resistor in ohm
6 Re=680 //emitter resistor in ohm
7 Ri=10000 //input resistor in ohm
8 R2=2.2 //resistor R2 in ohm
9 R1=10 //resistor R1 in ohm
10
11 Vb=R2/(R1+R2)*Vcc //bias voltage
12 printf("(1) Bias voltage = %.1f V \n",Vb)
13
14 Ie=(Vb-0.7)/Re //emitter current
15 printf(" Emitter current = %.2f mA\n",Ie*1e3)
16
17 Vc=Vcc-Rc*Ie //Dc collector voltage
18 printf(" DC collector voltage = %.2f V\n",Vc)
19
20 Vceq=Vc-Ie*Re //DC collector to emitter voltage
21 printf(" DC collector to emitter voltage = %.2f V\n"
    ,Vceq)
22
23 Pd=Vceq*Ie //power dissipation
24 printf(" Power dissipation = %.2f mW\n",Pd*1e3)
25
26 printf("\n(2) If collector resistance Rc is replaced
    by tank circuit \n there is no voltage drop
    across it. \n")
27 Vc=Vcc
28 printf(" DC collector voltage = %.0f V\n",Vc)
29
30 Vceq=Vc-Ie*Re //DC collector to emitter voltage
31 printf(" DC collector to emitter voltage = %.2f V\n"
    ,Vceq)
32
33 Pd=Vceq*Ie //power dissipation
34 printf(" Power dissipation = %.2f mW\n",Pd*1e3)

```

Scilab code Exa 11.5 Class C Tuned Amplfier

```
1 //Ex11_5 Pg-565
2 clc
3
4 Vin=5 //input voltage
5 Vp=Vin*sqrt(2) //peak voltage
6 printf("The peak value(maximum amplication) of input
       signal \n")
7 printf("                                = %.2f V",Vp)
8
9 Vin_pp=2*Vp //peak-to-peak value of input voltage
10 printf("\n Peak-to-peak value of input voltage \n")
11 printf("                                = %.2f V",Vin_pp
       )
12
13 Vbg=-1*(Vp-0.7) //base to ground voltage 0.7 is the
       voltage drop
14 printf("\n Base to ground voltage = %.2f",Vbg)
```

Chapter 12

Oscillators

Scilab code Exa 12.1 Colpitts Oscillator

```
1 //Ex12_1 Pg-587
2 clc
3
4 C1=0.001e-6 //capacitor c1 in farad
5 C2=0.01e-6 //capacitor c2 in farad
6 L=5e-6 //inductance in Henry
7
8 disp("To maintain vibrations in a colpitts oscilator
9      ,")
9 disp("                hfe >= C2/C1")
10 hfe=C2/C1 //transistor current gain
11 printf("                = %.f \n",hfe)
12
13 printf("So the value of hfe of transistor used must
14        be greater \n than 10. \n")
14 disp("Frequency of oscillations produced")
15 x=inv(C1)+inv(C2)
16 y=inv(L)
17 f=sqrt(x*y)
18 printf("                = %.1f*1e7 Hz",f*1e-7)
19 // answer in the book is wrong
```

Scilab code Exa 12.2 Phase Shift Oscillator

```
1 //Ex12_2 Pg-588
2 clc
3
4 RL=3.3*10^(3) //load resistor in ohm
5 R=5.6*10^(3) //resistor R in ohm
6 C=0.001*10^(-6) //capacitance in farad
7
8 disp("For oscillations to be maintained in a RC
      oscillator , ")
9 hfe=(23+(29*R/RL)+(4*RL/R)) //transistor current
      gain
10 printf("                = %.1 f \n",hfe)
11
12 disp("Frequency of oscillations ,")
13 f=1/(2*pi*C*sqrt((4*R*RL)+(6*R^2)))
14 //frequency of oscillation (textbook answer is wrong
15 // because of the used of wrong value of C)
16 printf("                = %.1 f Hz",f)
```

Scilab code Exa 12.3 Oscillator

```
1 //Ex12_3 Pg-588
2 clc
3
4 L=0.33 //inductance in henry
5 C=0.065*10^(-12) //capacitance in farad
6 Cm=10^(-12) //capacitance in farad
7 R=0.55*10^(3) //resistor R in ohm
8
```

```

9 disp("Series resonant frequency , fs = 1/2*pi*sqrt(L*
      C)")
10 fs=1/(2*%pi*sqrt(L*C))
11 printf("                = %.2 f MHz \n",fs*1e-6)
12
13 disp("Q of the crystal = 2*pi*fs*L/R")
14 Q=(2*%pi*fs*L)/R //quality factor (textbook answer
      wrong)
15 printf("                = %.0 f \n",Q)

```

Scilab code Exa 12.4 Pierce Oscillator

```

1 //Ex12_4 Pg-602
2 clc
3
4 L=3 //inductance in henry
5 Cs=0.05*10^(-12) //capacitance in farad
6 Cm=10*10^(-12) //capacitance in farad
7 R=2*10^(3) //resistor R in ohm
8
9 disp("Series resonant frequency , fs = 1/2*pi*sqrt(LC
      )")
10 fs=1/(2*%pi*sqrt(L*Cs))
11 printf("                = %.0 f KHz \n",fs*1e-3)
12
13 disp("The equivalent parallel capacitance , Cp = Cm*Cs
      /Cm+Cs")
14 Cp=Cm*Cs/(Cm+Cs) //quality factor
15 printf("                = %.4 f pF \n",Cp*1e12)
16
17 disp("Parallel resonant frequency , fp = 1/2*pi*sqrt(
      L*Cp)")
18 fp=1/(2*%pi*sqrt(L*Cp))
19 printf("                = %.0 f kHz \n",fp*1e-3)

```

Scilab code Exa 12.5 Oscillator

```
1 //Ex12_5 Pg-602
2 clc
3
4 disp("Fundamental frequency of oscillations of
      crystal")
5 disp("          fr = K/t")
6 disp("Let new thickness of the crystal be t''")
7 disp("          fr''/fr = t''/t = 99/100")
8 disp("So, new frequency fr'' = k/t''")
9 disp("or          fr'' = (99/100)*fr")
10 disp("or reduction in frequency,")
11 disp("          fr-fr'' = fr-(99/100)*fr")
12 disp("          = fr(1/100)")
13 disp(" or          fr-fr''/fr = 1/100 ")
14 disp(" Therefore, fr'' reduces by 1%")
```

Chapter 14

Optical Fibres and Communication

Scilab code Exa 14.1 Optical fibre

```
1 //Ex14_1 Pg-695
2 clc
3
4 n1=1.545 //core refractive index
5 n2=1.510 //cladding refractive index
6 d=3*10^(-6) //diameter of optical fiber in m
7
8 a=d/2 //core radius in m
9 del=(n1-n2)/n1 //fractional difference of refractive
    indices
10 lamda_c=(2*%pi*a*n1*sqrt(2*del))/2.405 //cut-off
    wavelength
11 printf("Cut-off wavelength = %.2f um",lamda_c*1e6)
```

Scilab code Exa 14.2 Multimode Index Fibre

```

1 //Ex14_2 Pg-695
2 clc
3
4 n1=1.53 //core refractive index
5 n2=1.5 //cladding refractive index
6 lamda=10^(-6) //cut-off wavelength
7 a=50*10^(-6) //core radius in m
8
9
10 V=(2*pi*a*sqrt(n1^2-n2^2))/lamda //normalised
    frequency
11 printf("Normalised frequency = %.2f \n",V)
12
13 ms=V^2/2 //total number of guided mode
14 printf("Total number of guided mode = %.0f",ms)

```

Scilab code Exa 14.3 Multimode Step Index Fibre

```

1 //Ex14_3 Pg-695
2 clc
3
4 n1=1.5 //core refractive index
5 n2=1.46 //cladding refractive index
6
7 tetha_rad=asin(n2/n1) //critical angle in radians
8 tetha=tetha_rad*180/pi //critical angle in degree
9 printf("Critical angle = %.1f degree \n",tetha)
10
11 tetha_m_rad=asin(sqrt(n1^2-n2^2)) //acceptance angle
    in radians
12 tetha_m=tetha_m_rad*180/pi
13 printf("Acceptance angle = %.1f degree \n",tetha_m)
14
15 NA=sin(tetha_m_rad)
16 printf("Numerical Apperture = %.3f",NA)

```

Scilab code Exa 14.4 Optical Fibre

```
1 //Ex14_4 Pg-696
2 clc
3
4 NA=0.5 //numerical apperture
5 n1=1.54 //core refractive index
6
7 n2=sqrt(n1^2-NA^2) //cladding refractive index
8 printf("(1) Cladding refractive index = %.3f \n",n2)
9
10 RI=(n1-n2)/n1 //change in core cladding refractive
    index
11 printf(" (2) RI of the core = %.4f",RI)
```

Scilab code Exa 14.5 Step Index Fibre

```
1 //Ex14_5 Pg-696
2 clc
3
4 n1=1.5//core refractive index
5 n2=1.48 //cladding refractive index
6 n=1
7
8 NA=sqrt(n1^2-n2^2) //numerical apperture
9 printf("(1) Numerical apperture = %.5f \n",NA)
10
11 AA_rad=asin(NA/n) //maximum Acceptance angle in rad
12 AA=AA_rad*180/%pi //maximum entrance angle in degree
13 printf(" (2) The maximum entrance angle i0 = %.2f
    degree",AA)
```

Scilab code Exa 14.6 Optical Fibre

```
1 //Ex14_6 Pg-697
2 clc
3
4 n2=1.59 //cladding refractive index
5 NA=0.2 //numerical apperture
6 n0=1 //when fiber is in air
7
8 n1=sqrt(n2^2+NA^2) //core refractive index
9 printf("Core refractive index = %.3f \n",n1)
10
11 n0=1.33 //water refractive index
12 NA=sqrt(n1^2-n2^2)/n0 //numerical apperture
13 printf(" Numerical apperture = %.2f \n",NA)
14
15 AA_rad=asin(NA/n) //maximum Acceptance angle in rad
16 AA=AA_rad*180/%pi //maximum entrance angle in degree
17 printf(" The maximum entrance angle i0 = %.2f degree
    ",AA)
```

Scilab code Exa 14.7 Optical Fibre

```
1 //Ex14_7 Pg-697
2 clc
3
4 NA=0.22 //numerical apperture
5 del=0.012 //fractional difference of refractive
    indices
6
7 n1=NA/(sqrt(2*del)) //core refractive index
8 printf("Core refractive index = %.2f \n",n1)
```

```

9
10 n2=n1-delta*n1 //cladding refractive index
11 printf(" Cladding refractive index = %.2f \n",n2)

```

Scilab code Exa 14.8 Optical Fibre

```

1 //Ex14_8 Pg-698
2 clc
3
4 n1=1.52 //core refractive index
5 n2=1.46 //cladding refractive index
6
7 delta=(n1-n2)/n1 //fractional difference of refractive
   indices
8
9 NA=n1*sqrt(2*delta) //numerical aperture
10 printf(" Numerical aperture = %.3f \n",NA)
11
12 AA_rad=asin(NA/n) //maximum Acceptance angle in rad
13 AA=AA_rad*180/pi //maximum entrance angle in degree
14 printf(" Acceptance angle i0 = %.2f degree \n",AA)
15
16 theta_rad=asin(n2/n1) //critical angle in radians
17 theta=theta_rad*180/pi //critical angle in degree
18 printf(" Critical angle = %.1f degree \n",theta)

```

Scilab code Exa 14.9 Step Index Fibre

```

1 //Ex14_9 Pg-698
2 clc
3
4 n1=1.45 //core refractive index
5 NA=0.16 //cladding refractive index

```

```

6 lamda=0.9*10(-6) //cut-off wavelength
7 d=60/100 //core radius in m
8
9
10 V=(%pi*d*NA)/lamda //normalised frequency
11 printf("Normalised frequency = %.2f*1e5 \n",V*1e-5)

```

Scilab code Exa 14.10 Step Index Fibre Single Mode

```

1 //Ex14_10 Pg-698
2 clc
3
4 n1=1.48 //core refractive index
5 n2=1.47 //cladding refractive index
6 lamda=850e-6 //cut-off wavelength
7 V=2.405 //normalised frequency
8 //In the book cut off wavelength in the question is
   850 um but in
9 // the calculation part it is taken as 850nm. Here I'
   ve taken 850um
10 d=V*lamda/(%pi*sqrt(n12-n22)) //diamter of core
11 a=d/2 //radius of core
12 printf("Radius of core = %.2f mm \n",a*1e3)//answer
   in the book is wrong
13
14 NA=sqrt(n12-n22) //numerical apperture
15 printf(" Numerical apperture = %.4f \n",NA)
16
17 AA_rad=asin(NA/n) //maximum Acceptance angle in rad
18 AA=AA_rad*180/%pi //maximum entrance angle in degree
19 printf(" Acceptance angle i0 = %.2f degree \n",AA)

```

Scilab code Exa 14.11 Loss in Optical Fibre

```

1 //Ex14_11 Pg-699
2 clc
3
4 L=500/1000 //length of fiber in m
5 Pin=1*10^(-3) //input power in watt
6 Pout=85/100*10^(-3) //output power in watt
7
8 alpha=(10/L)*log10(Pin/Pout) //loss
9 printf("Loss in the fiber = %.2f dB/Km",alpha)

```

Scilab code Exa 14.12 Communication System Output Power

```

1 //Ex14_12 Pg-699
2 clc
3
4 L=10 //length of fiber in km
5 alpha=2.5 //loss in the fiber per km
6 Pin=500*10^(-6) //input power in watt
7
8 tot_alpha=-1*alpha*L //total loss in the fiber
9 Pout=Pin*10^(tot_alpha/10) //output power in watt
10 printf("Output power = %.2f uW",Pout*1e6)

```

Scilab code Exa 14.13 GRIN Optical Fibre

```

1 //Ex14_13 Pg-700
2 clc
3
4 del=1/100 //fractional difference of refractive
   indices
5 lamda=1.3*10^(-6) //cutoff wavelength in m
6 n1=1.5 //refractive index
7 d=6.6*10^(-6) //diameter of the core

```

```

8 alpha=2 //loss in fiber
9
10 disp("      We have for a GRIN , maximum value of
      normalized frequency for single mode operation is
      given by")
11 disp("          V = 2.4*sqrt(1+2/alpha)")
12 V=2.4*sqrt(1+2/alpha) //normalized frequency
13
14 disp("      For maximum core radiation , we have")
15 r=V*lamda/(2*pi*n1*sqrt(2*del)) //radius of the
      core
16 printf("          r = %.1f um \n",r*1e6)
17
18 rr=2*r //diameter of the core
19 printf("\n Maximum core diameter which permit single
      mode operation \n = 2*r = %.1f um",rr*1e6)

```

Scilab code Exa 14.14 Multimode Step Index Optical Fibre

```

1 //Ex14_14 Pg-700
2 clc
3
4 del=1.5/100 //fractional difference of refractive
      indices
5 lamda=0.85*10^(-6) //cutoff wavelength in m
6 n1=1.48 //refractive index
7 d=6.6*10^(-6) //diameter of the core
8 V=2.4 //normalized frequency
9
10
11 disp("      For maximum core radiation , we have")
12 r=V*lamda/(2*pi*n1*sqrt(2*del))
13 printf("          r = %.1f um \n",r*1e6)
14
15 r=1.3*10^(-6) //actual radius=1.266 micrometer and

```



```

    assumed to 1.3 micrometer
16 rr=2*r //diameter of the core
17 printf("\n Maximum core diameter which permit single
    mode operation \n = 2*r = %.1f um",rr*1e6)

```

Scilab code Exa 14.15 Optical Fibre

```

1 //Ex14_15 Pg-701
2 clc
3
4 alpha=3.5 //loss in fiber
5 Pi=0.5//input power in milli watt
6 L=4 //length of fiber in km
7
8 disp("The attenuation of an optical fiber is given
    by")
9 disp("          alpha = (10/L)*log(Pi/Po)")
10
11 Po=Pi/(10^(alpha*L/10))
12 printf("\n Output power = %.2 f mW",Po*1e3)

```

Scilab code Exa 14.16 Single Mode Step Index Fibre

```

1 //Ex14_16 Pg-701
2 clc
3
4 n1=1.46 //core refractive index
5 r=4.5*10^(-6) //radius of the core
6 del=0.25/100 //fractional difference of refractive
    indices
7 Vc=2.405 //normalized frequency
8
9 disp("We have, cut-off wavelength expression")

```

```
10 lamda=(2*%pi*r*n1*sqrt(2*del))/Vc
11 printf("                = %.3f um",lamda*1e6)
```

Chapter 15

Communication System

Scilab code Exa 15.1 Transmitter Modulation

```
1 //Ex15_1 Pg-773
2 clc
3
4 Pc=10000 //carrier input power in watt
5 m=30/100 //modulation of 30%
6
7 disp("Total power = carrier power*(1+m^2/2)")
8 Pt=Pc*(1+m^2/2) //total power
9 printf("                = %.2f kW" ,Pt*1e-3)
```

Scilab code Exa 15.2 Transmitter Modulation

```
1 //Ex15_2 Pg-774
2 clc
3
4 Ic=100 //carrier current in A
5 m=80/100 //modulation of 80%
6
```

```

7 disp(" Total current = carrier current*(1+m^2/2)")
8 It=Ic*sqrt(1+m^2/2) //total power
9 printf("                = %.1f A \n",It)
10
11 change_I=It-Ic //change in current
12 printf(" Therefore, increase in current due to
        modulation = %.1f A",change_I)

```

Scilab code Exa 15.3 AM Wave Modulation

```

1 //Ex15_3 Pg-774
2 clc
3
4 Em=5 //modulated wave amplitude
5 Ec=100 //carrier wave amplitude
6 Fm=50 //frequency of modulated wave
7 Fc=10*10^(3) //frequency of carrier wave
8
9 disp("(1) Modulation Factor")
10 m=Em/Ec //modulation factor
11 per_m=m*100 //modulation factor in percentage
12 printf("                m = %.0f %%" ,per_m)
13
14 disp("(2) Amplitude of each sideband = m*Ec/2")
15 Amp=m*Ec/2 //amplitude of each sideband
16 printf("                = %.1f" ,Amp)
17
18 USB=Fc+Fm //upper side band
19 LSB=Fc-Fm //lower side band
20 disp("(3) Frequenc of sidebands")
21 printf("                USB = %.0f Hz \n",USB)
22 printf("                LSB = %.0f Hz \n",LSB)
23
24 disp("(4) Bandwidth of the wave")
25 BW=2*Fm //Bandwidth

```

```
26 printf("          BW = %.0f", BW)
```

Scilab code Exa 15.4 AM Wave Modulation

```
1 //Ex15_4 Pg-774
2 clc
3
4 Vmax=600 //peak to peak voltage
5 Vmin=100 //valley to valley voltage
6
7 disp("From figure 15.49, we have")
8 m=(Vmax-Vmin)/(Vmax+Vmin) //modulation factor
9 per_m=m*100 //modulation factor in percentage
10 printf(" \n Modulation factor = %.1f %%", per_m )
```

Scilab code Exa 15.5 AM Wave Modulation

```
1 //Ex15_5 Pg-775
2 clc
3
4 disp("The standard equation of AM wave is")
5 disp(" e = Ec*(1+m*sin(omega_m*t))*sin(omega_c*t))
   -->eqn 1")
6 disp(" Given the equation")
7 disp(" e = 20*(1+0.7*sin(6280*t))*sin(628000*t))    --
   eqn 2")
8 disp("Comparing eqn 1 and eqn 2 one obtains")
9 disp("(1) Modulation factor , m = 0.7")
10 m=0.7 //modulation factor
11 disp("(2) Carrier Amplitude , Ec = 20 V")
12 Ec=20 //carrier wave amplitude in V
13 disp("(3) omega_m = 6280")
14 omega_m=6280 //modulating frequency
```

```

15 Fm=omega_m/(2*pi) //signal frequency
16 printf("      Signal frequency = %.0f kHz \n\n",Fm*1e
    -3)
17
18 omega_c=628000 //carrier frequency in Hz
19 Fc=omega_c/(2*pi)
20 printf("(4) Signal frequency = %.0f kHz \n\n",Fc*1e
    -3)
21
22 Emax=Ec+m*Ec //minimum amplitude of wave
23 printf("(5) Emax = %.0f V \n\n",Emax)
24
25 Emin=Ec-m*Ec //minimum amplitude of wave
26 printf("(5) Emin = %.0f V\n\n",Emin)
27
28 BW=2*Fm //Bandwidth
29 printf("(6) BW = %.0f kHz",BW*1e-3)

```

Scilab code Exa 15.6 AM Wave Modulation

```

1 //Ex15_6 Pg-776
2 clc
3
4 Pc=10000 //carrier power in watt
5 m=0.9 //modulation factor
6 disp("We have")
7
8 disp("Total power = carrier power*(1+m^2/2)")
9 Pt=Pc*(1+m^2/2) //total power
10 printf("          = %.0f kW \n\n",Pt*1e-3)
11
12 printf("This will be the maximum power handed by
    the transmitter.\n Now,increased unmodulated
    carrier power can be obtained by \n\n")
13 m=40/100 //modulation in terms of percentage

```

```

14 Pt=14000 //total power
15 Pc=Pt/(1+m^2/2) //new carrier power
16 printf("          Pc = %.2 f kW",Pc*1e-3)

```

Scilab code Exa 15.7 AM Wave Modulation

```

1 //Ex15_7 Pg-776
2 clc
3
4 disp("Given the equation")
5 printf("\n E = 100*sin(628000*t) + 25*sin(621720*t)
        \n   - 25*cos(634280*t)) \n")
6 m=50/100 //modulation factor in percentage
7 Ec=100 //carrier wave amplitude in V
8 Em=10 //modulated wave amplitude in V
9 Fc=100000 //carier frequency in Hz
10 Fm=1000 //modulating frequency in Hz
11 pi=3.14
12
13 omega_c=2*pi*Fc //carier frequency
14 omega_m=2*pi*Em //modulating frequency
15
16 disp("Now,putting these equation in the standard
        equations for modulated voltage wave,")
17 disp(" e = Ec*sin(omega_c*t)+m*Ec/2*cos(omega_c-
        omega_m)*t-m*Ec/2*cos(omega_c-omega_m)*t")
18 USB=omega_c+omega_m //upper sideband
19 LSB=omega_c-omega_m //lower sideband
20 mEc=m*Ec/2
21 printf("\n          = 100*sin(628000*t) + %.0 f*sin(%.0 f*t
        ) \n          - %.0 f*cos(%.0 f*t))",mEc,USB,mEc,LSB)

```

Chapter 18

Optoelectronic Devices

Scilab code Exa 18.1 DC Circuit of LED

```
1 //Ex18_1 Pg-901
2 clc
3
4 Vs=12 //supply voltage in V
5 Vd=2 //forward bias voltage in V
6 Id=20*10(-3) //forward bias current
7
8 Rs=(Vs-Vd)/Id //source resistor
9 printf("Source resistance = %.0f ohm \n",Rs)
10
11 P=Id2*Rs //power
12 printf(" Wattage rating = %.1f mW",P*1e3)
13 disp("Therefore a standard size 0.25 watt = 250mW
      resistor is required")
```

Scilab code Exa 18.2 Spontaneous and Stimulated Emission

```
1 //Ex18_2 Pg-945
```



```

2  clc
3
4  T=2000 //temperature in Kelvin
5  f=5*10^(14) // frequency in Hz
6  h=6.6*10^(-34) //planck constant
7  k=1.38*10^(-23) //Boltzmann constant
8
9  R=exp((h*f)/(k*T)) //ratio of spontaneous and
    stimulated emisson
10 printf("      R = %.2 f*1e5" ,R*1e-5)

```

Scilab code Exa 18.3 Relative Populations

```

1  //Ex18_3 Pg-946
2  clc
3
4  disp("Average wavelength of visible radiation = 550
    nm")
5  disp("      E1 - E2 = hc/lamda")
6  h=6.6*10^(-34) //planck constant
7  c=3*10^(8) //speed of light in sec
8  lamda= 550*10^(-9) //wavelength in m
9  E=h*c/lamda //difference in energy levels in Joules
10 e=1.6*10^(-19) //electron charge in eV
11 E_eV=E/e //difference in energy levels in
    electronVolt
12 printf("      = %.1 f*1e-19 J \n" ,E*1e19)
13 printf("      = %.2 f eV \n" ,E_eV)
14
15 T=300 //temperature in Kelvin
16 k=1.38*10^(-23) //Boltzmann constant
17 disp("Average room temperature=300K and g1=g2 ,we
    have")
18 N=exp((-E)/(k*T))
19 printf(" N2/N1 = %.2 f*1e-37" ,N*1e37)

```

20 //answer in the book is wrong

Scilab code Exa 18.4 Silicon Crystal Problem

```
1 //Ex18_4 Pg-946
2 clc
3
4 w=0.3*10^(-6)*100 //width of silicon in cm
5 alpha=4*10^(4)
6 phi=10^(-2)
7 e=1.6*10^(-19) //electron charge in eV
8
9 disp("(1) Energy absorbed/sec is given by ")
10 E=phi*(1-exp(alpha*w)) //energy absorbed(textbook
    answer is wrong)
11 printf("                = %.1f mW \n",abs(E)*1e3)
12
13 disp("(2) The portion of each photo energy that is
    converted into heat is obtained as hv-Eg/hv")
14 Heat=(3-1.12)/3*100 //photon energy covered to heat
15 printf("                = %.0f %%",Heat)
16 E1=(62/100)*0.0232 //energy dissipated/sec (textbook
    answer is wrong)
17 printf("\n Obviously, the amount of energy
    dissipated/sec to lattice \n is %.1f mW \n",E1*1
    e3)
18
19 disp("(3) Number of photons/sec from recombination
    is ")
20 num_photons=2.4/(e*1.12)
21 printf("                = %.1f*1e19 photon/sec \n",
    num_photons*1e-19)
22 //textbook answer is wrong
23
24 disp("Therefore recombination radiation")
```

```

25 RR=abs(E)-E1 //recombination radiation (textbok
    answer is wrong)
26 printf("                = %.1 f mW" ,RR*1e3)

```

Scilab code Exa 18.5 Electrons Diffusion

```

1 //Ex18_5 Pg-947
2 clc
3
4 d=5*10^(-6) //thickness of silicon in m
5 Dc=3.4*10^(-3) //diffusion coefficient in m^2sec
    ^(-1)
6
7 t=d^2/(2*Dc) //time taken to diffuse
8 printf("Time taken to diffuse = %.1f*1e-9 sec",t*1e9
    )

```

Scilab code Exa 18.6 Frequency Bandwidth Limitations

```

1 //Ex18_6 Pg-947
2 clc
3
4 A=10^(-6) //diode area in m
5 epsilon_r=11.7 //relative permittivity
6 Nd=10^(21) //number of doping carriers
7 V=10 //bias potential in V
8 e=1.6*10^(-19) //electron charge in eV
9 epsilon_0=8.85*10^(-12) //permittivity of free space
10
11 Cj=A/2*sqrt(2*e*epsilon_r*epsilon_0*Nd)/sqrt(V)
12 printf("Diode capacitance = %.f pF" ,Cj*1e12)
13 //textbook answer is wrong

```

Scilab code Exa 18.7 SemiconductorLaser

```
1 //Ex18_7 Pg-947
2 clc
3
4 L=10(-6) //length of cavity in m
5 r2=0.5 //relative coefficient of semiconductor
6 r1=1.5 //relative coefficient of semiconductor
7
8 disp("No internal loss means di=0; we have")
9 g=log10(1/(r1*r2))/(2*L) //gain of the laser (
    textbook answer is wrong)
10 printf(" Gain g = %.2f*1e3 cm(-1)",g*1e-3)
```

Scilab code Exa 18.8 Photoconductor Device

```
1 //Ex18_8 Pg-947
2 clc
3
4 L=100*10(-6) //length of semiconductor in m
5 A=10(-7) //area of semiconductor in cm2
6 V=10 //applied voltage in V
7 mew_n=1350 //mobility of electrons
8 mew_p=480 //mobility of protons
9 tp=10(-6) //lifetime of protons in sec
10
11 tn=L/(mew_n*V) //lifetime of electrons in sec
12
13 Gain=tp/tn*(1+(mew_p/mew_n)) //gain of
    photoconductor
14 printf(" Gain = %.2f*1e2",Gain*1e-2)
```

Chapter 19

Digital Electronics

Scilab code Exa 19.1 Binary to Decimal

```
1 //Ex19_1 Pg-957
2 clc
3
4 bin='101'; //binary input
5 dec=bin2dec(bin) //decimal output
6 disp("The decimal equivalent of 101 is")
7 disp(dec)
```

Scilab code Exa 19.2 Binary to Decimal

```
1 //Ex19_2 Pg-958
2 clc
3
4 bin='11101'; //binary input
5 dec=bin2dec(bin) //decimal output
6 disp("The decimal equivalent of 11101 is")
7 disp(dec)
```

Scilab code Exa 19.3 Binary to Decimal

```
1 //Ex19_3 Pg-958
2 clc
3
4 a=1
5 b=0
6 c=1
7 dec=a*2^(-1)+b*2^(-2)+c*2^(-3) //decimal output
8 disp("The decimal equivalent of 0.101 is")
9 disp(dec)
```

Scilab code Exa 19.4 Binary to Decimal

```
1 //Ex19_4 Pg-958
2 clc
3
4 a=1
5 b=1
6 c=0
7 d=1
8 dec=a*2^(-1)+b*2^(-2)+c*2^(-3)+d*2^(-4) //decimal
  output
9 disp("The decimal equivalent of 0.1101 is")
10 disp(dec)
```

Scilab code Exa 19.5 Binary to Decimal

```
1 //Ex19_5 Pg-958
```

```

2  clc
3
4  //Integer part
5  bin='10101'; //binary input
6  dec_I=bin2dec(bin) //decimal output
7
8  //Decimal part
9  a=1
10 b=0
11 c=1
12 dec_D=a*2(-1)+b*2(-2)+c*2(-3)
13 dec=dec_I+dec_D //decimal output
14 disp("The decimal equivalent of 10101.101 is")
15 disp(dec)

```

Scilab code Exa 19.6 Decimal to Binary

```

1  //Ex19_6 Pg-959
2  clc
3
4  dec=9 //decimal input
5  bin=dec2bin(dec) //binary output
6  disp("The binary equivalent of 9 is")
7  disp(bin)

```

Scilab code Exa 19.7 Decimal to Binary

```

1  //Ex19_7 Pg-959
2  clc
3
4  dec=31 //decimal input
5  bin=dec2bin(dec) //binary output
6  disp("The binary equivalent of 31 is")

```

```
7 disp(bin)
```

Scilab code Exa 19.8 Decimal to Binary

```
1 //Ex19_8 Pg-959
2 clc
3
4 dec=13 //decimal input
5 bin=dec2bin(dec) //binary output
6 disp("The binary equivalent of 13 is")
7 disp(bin)
```

Scilab code Exa 19.9 Decimal to Binary

```
1 //Ex19_9 Pg-960
2 clc
3 disp("Conversion of decimal number 31.65 base to its
      binary equivalent ")
4 a=31.65;
5 z=modulo(a,1)
6 x=floor(a); //separating the decimal from the integer
      part
7 b=0;
8 c=0;
9 d=0;
10 while(x>0) //taking integer part into a matrix and
      convert to equivalent binary
11 y=modulo(x,2);
12 b=b+(10^c)*y;
13 x=x/2;
14 x=floor(x);
15 c=c+1;
16 end
```



```

17 for i=1:10; //converting the values after the decimal
    point into binary
18     z=z*2;
19     q=floor(z);
20     d=d+q/(10^i);
21     if z>=1 then
22         z=z-1;
23     end
24 end
25 s=b+d;
26 printf("\n                =%.6 f" ,s);

```

Scilab code Exa 19.10 Binary to Decimal

```

1 //Ex19_10 Pg-961
2 clc
3
4 bin='1111'; //binary input
5 dec=bin2dec(bin) //decimal output
6 disp("(1) Binary number -> 1111")
7 disp("Here we have 4 bits")
8 disp("The decimal equivalent is")
9 disp(dec)
10
11
12 bin='1111111'; //binary input
13 dec=bin2dec(bin) //decimal output
14 disp("(2) Binary number -> 1111111")
15 disp("Here we have 7 bits")
16 disp("The decimal equivalent is")
17 disp(dec)

```

Scilab code Exa 19.11 Octal to Decimal

```
1 //Ex19_11 Pg-962
2 clc
3
4 oct='23'; //binary input
5 dec=oct2dec(oct) //decimal output
6 disp("The decimal equivalent of 23 is")
7 disp(dec)
```

Scilab code Exa 19.12 Octal to Decimal

```
1 //Ex19_12 Pg-962
2 clc
3
4 oct='257'; //binary input
5 dec=oct2dec(oct) //decimal output
6 disp("The decimal equivalent of 257 is")
7 disp(dec)
```

Scilab code Exa 19.13 Decimal to Octal

```
1 //Ex19_13 Pg-962
2 clc
3
4 dec=175; //binary input
5 oct=dec2oct(dec) //decimal output
6 disp("The octal equivalent of 175 is")
7 disp(oct)
```

Scilab code Exa 19.14 Decimal to Octal

```

1 //Ex19_14 Pg-963
2 clc
3
4 disp(" Conversion of decimal number 0.85 base to its
      octal equivalent ")
5 a=[0.85] //0.75 value in textbook is wrong
6 z=modulo(a,1)
7 d=0
8 for i=1:10//converting the values after the decimal
      point into octal
9     z=z*8
10    q=floor(z)
11    d=d+q/(10^i)
12    if z>=1 then
13        z=z-q
14    end
15 end
16 s=d
17 printf("\n                = %.6 f" ,s);

```

Scilab code Exa 19.15 Octal to Binary

```

1 //Ex19_15 Pg-963
2 clc
3
4 oct='34' //octal input
5 dec=oct2dec(oct) //decimal output
6 bin=dec2bin(dec) //binary output
7 disp("The binary equivalent of octal 34 is")
8 disp(bin)

```

Scilab code Exa 19.16 Octal to Binary

```

1 //Ex19_16 Pg-963
2 clc
3
4 oct='357' //octal input
5 dec=oct2dec(oct) //decimal output
6 bin=dec2bin(dec) //binary output
7 disp("The binary equivalent of octal 34 is")
8 disp(bin)

```

Scilab code Exa 19.17 Binary to Octal

```

1 //Ex19_17 Pg-963
2 clc
3
4 //Integer part
5 bin='1011'; //binary input
6 dec_I=bin2dec(bin) //decimal output
7 oct_I=dec2oct(dec_I) //octal output
8
9 //Decimal part
10 bin='11010'; //binary input
11 dec_D=bin2dec(bin) //decimal output
12 oct_D=dec2oct(dec_D) //octal output
13 oct=oct_I + oct_D //final octal output
14 b = strcat([ oct_I, oct_D ], '.' ) // combining
    integer and decimal part
15 disp("The octal equivalent of 1011.01101 is")
16 disp(b)

```

Scilab code Exa 19.18 Hexadecimal to Binary

```

1 //Ex19_18 Pg-965
2 clc

```

```
3
4 hex='9AF' //hexadecimal input
5 dec=hex2dec(hex) //decimal output
6 bin=dec2bin(dec) //binary output
7 disp("The binary equivalent of 9AF is")
8 disp(bin)
```

Scilab code Exa 19.19 Hexadecimal to Binary

```
1 //Ex19_19 Pg-965
2 clc
3
4 hex='C5E2' //hexadecimal input
5 dec=hex2dec(hex) //decimal output
6 bin=dec2bin(dec) //binary output
7 disp("The binary equivalent of C5E2 is")
8 disp(bin)
```

Scilab code Exa 19.20 Binary to Hexadecimal

```
1 //Ex19_20 Pg-957
2 clc
3
4 bin='10001100'; //binary input
5 dec=bin2dec(bin) //decimal output
6 hex=dec2hex(dec) //hexadecimal output
7 disp("The hexadecimal equivalent of 10001100 is")
8 disp(hex)
```

Scilab code Exa 19.21 Hexadecimal to Decimal

```

1 //Ex19_21 Pg-965
2 clc
3
4 //Integer part
5 hex='F8E6'; //binary input
6 dec_I=hex2dec(hex) //decimal output
7
8 //Decimal part
9 a=3
10 b=9
11 dec=dec_I+a*16^(-1)+b*16^(-2) //decimal output
12 disp("The decimal equivalent of F8E6.39 is")
13 printf("\n      %.4 f",dec)

```

Scilab code Exa 19.22 Decimal to Hexadecimal

```

1 //Ex19_22 Pg-966
2 clc
3
4 dec=2479 //decimal input
5 hex=dec2hex(dec) //hexadecimal output
6 disp("The Hexadecimal equivalent of 2479 is")
7 disp(hex)

```

Scilab code Exa 19.23 Decimal to Hexadecimal

```

1 //Ex19_23 Pg-966
2 clc
3
4 dec=65535 //decimal input
5 hex=dec2hex(dec) //hexadecimal output
6 disp("The Hexadecimal equivalent of 65535 is")
7 disp(hex)

```

```
8
9 bin=dec2bin(dec) //binary output
10 disp("The Binary equivalent of 65535 is ")
11 disp(bin)
```

Scilab code Exa 19.24 Binary Addition

```
1 //Ex19_24 Pg-969
2 clc
3
4 x=bin2dec('11100') //1st input
5 y=bin2dec('11010') //2nd input
6 z=x+y //binary addition
7 add=dec2bin(z)
8 disp(" 11100 + 11010 = ")
9 disp(add)
```

Scilab code Exa 19.25 Binary Subtraction

```
1 //Ex19_25 Pg-970
2 clc
3
4 x=bin2dec('1101') //1st input
5 y=bin2dec('1010') //2nd input
6 z=x-y //subtraction
7 sub=dec2bin(z,4)
8 disp(" 1101 - 1010 = ")
9 disp(sub)
```

Scilab code Exa 19.26 Binary Subtraction

```

1 //Ex19_26 Pg-970
2 clc
3
4
5 x=200 //1st input
6 y=125 //2nd input
7 z=x-y //subtraction
8 disp("For decimal system 200 - 125 = ")
9 disp(z)
10
11 a=dec2hex(z) //hexadeciaml output of 200-125
12 disp("For hexadecimal system C8 - 7D is")
13 disp(a)
14
15 b=dec2bin(z,8) //binary output of 200-125
16 disp("For binary system 11001000 - 01111101 is")
17 disp(b)

```

Scilab code Exa 19.27 Boolean Expression

```

1 //Ex19_27 Pg-997
2 clc
3
4 disp("(A+B)(A+C) = AA + AC + AB + BC ")
5 //by distributive law
6 disp("          = A + AC + AB +BC")
7 //by theorem(6)
8 disp("          = A(1 + C) + AB + BC")
9 //by distributive law
10 disp("          = A.1 + AB + BC ")
11 //by theorem(3)
12 disp("          = A(B + 1) + BC")
13 //by distributive law
14 disp("          = A + BC")
15 disp("Therefore (A+B)(A+C) = A + BC")

```

Scilab code Exa 19.28 Boolean Expression

```
1 //Ex19_28 Pg-998
2 clc
3
4 disp("AB + A(B + C) + B(B + C) = AB + AB + AC + BB
      + BC")
5 //using distributive law
6 disp("                                = AB + AC + B +BC
      ")
7 //using law 6
8 disp("                                = AB + AC + B(1 + C)
      ")
9 //taking common B from B + BC
10 disp("                                = AB + AC + B")
11 //using law 7
12 disp("                                = B(A + 1) + AC")
13 //taking common B from AB + B
14 disp("                                = B + AC")
15 //using law 7
16 disp("Therefore AB + A(B + C) + B(B + C) = B + AC")
```

Scilab code Exa 19.30 Boolean Expression

```
1 //Ex19_30 Pg-998
2 clc
3
4 // question in the textbook is wrong7
5 disp("LHS : (A + B + C)(A + B + C) ")
6 disp("                                = AA + AB + AC + BA + BB +BC + CA + CB
      + CC")
```

```

7 //using distributive law
8 disp("      = A + AB + AC + BA + B +BC + CA + CB +
      C")
9 //using law 6
10 disp("      = A + AB + AC +BC + CB + C")
11 //using law 5
12 disp("      = A(B + 1) + AC + B + C(B + 1)")
13 //taking A common from A+AB and C from CB+C
14 disp("      = A + AC + B + C")
15 //using law 3
16 disp("      = A + B + C(A + 1)")
17 //taking common C from AC+C
18 disp("      = A + B + C")
19 //using law 3
20 disp("Therefore (A' + B + C)(A' + B' + C) = A' +
      C")

```

Scilab code Exa 19.31 Boolean Expression

```

1 //Ex19_31 Pg-999
2 clc
3
4 disp("LHS : (A+ C)(A' + B) ")
5 disp("      = AA' + AB + CA' + BC") //using
      distributive law
6 disp("      = 0 + AB + CA' + BC") //using law 8
7 disp("      = AB + (A + A')BC + CA'") //using
      law 7
8 disp("      = AB + ABC + A'BC + CA'")
9 //using distributive law
10 disp("      = AB + ABC + A'C(B + 1)")
11 //taking common A'C from A'BC + CA'
12 disp("      = AB + ABC + A'C") //using law 3
13 disp("      = AB(C + 1)+ A'C")
14 //taking common AB from AB + ABC

```

```
15 disp('          = AB + A''C') //using law 3
16 disp('Therefore (A+ C)(A'' + B) = AB + A''C')
```

Chapter 21

Integrated Circuits

Scilab code Exa 21.1 AC Coupled Trigger Pulse

```
1 //Ex21_1 Pg-1067
2 clc
3
4 disp("Refer to figure 21.10")
5
6 R=5000 //resistor R in ohm
7 C=0.1*10^(-6) //capacitance in farad
8 tau=1.1 //time constant
9
10 tON=tau*R*C //pulse width in sec
11 printf(" Pulse width = %.2f msec",tON*1e3)
```

Scilab code Exa 21.2 Astable Mode Time Output

```
1 //Ex21_2 Pg-1068
2 clc
3
4 disp("Refer to figure 21.12")
```

```

5 R1=20000 //timing resistor R1 in ohm
6 R2=R1 //timing resistor R2 in ohm
7 C=0.1*10(-6) //capacitance in farad
8 tau=0.69 //time constant
9
10 tHIGH=tau*(R1+R2)*C //time output that will remain
    high
11 printf(" Time output = %.2f msec",tHIGH*1e3)

```

Scilab code Exa 21.3 Astable Mode Time Output

```

1 //Ex21_3 Pg-1069
2 clc
3
4 disp("Refer to figure 21.12")
5 R2=20000 //timing resistor R2 in ohm
6 C=0.1*10(-6) //capacitance in farad
7 tau=0.69 //time constant
8
9 tLOW=tau*(R2)*C //time output that will remain high
10 printf(" Time output = %.2f msec",tLOW*1e3)

```

Scilab code Exa 21.4 Astable Mode Output Frequency

```

1 //Ex21_4 Pg-1069
2 clc
3
4 R1=20000 //timing resistor R1 in ohm
5 R2=R1 //timing resistor R2 in ohm
6 C=0.1*10(-6) //capacitance in farad
7
8 f0=1.45/((R1+2*R2)*C) //output frequency
9 printf("Output frequency = %.2f Hz",f0)

```

Scilab code Exa 21.5 Astable Mode Duty Cycle

```
1 //Ex21_5 Pg-1069
2 clc
3
4 disp("Refer to figure 21.12")
5 R1=20000 //timing resistor R1 in ohm
6 R2=R1 //timing resistor R2 in ohm
7
8 D=(R1+R2)/(R1+2*R2)*100 //duty cycle
9 printf(" Duty cycle = %.1f %%",D)
```

Scilab code Exa 21.6 Resistor Values in Astable Mode

```
1 //Ex21_6 Pg-1070
2 clc
3
4 C=0.01*10^(-6) //capacitance in farad
5 f0=2000 //frequency in Hz
6
7 Req=1.45/(f0*C) //equivalent resistance or R1+R2
8 disp(" Because a square wave has duty cycle of 50
9 % each resistor must be the same")
9 R1=Req/2
10 R2=R1
11 printf(" R1 = R2 = %.2f kohm",R2*1e-3)
```

Scilab code Exa 21.7 Time Delay Mode

```

1 //Ex21_7 Pg-1073
2 clc
3
4 R=260000 //resistor R in ohm
5 C=25*10(-6) //capacitance in farad
6 tau=1.1 //time constant
7
8 t_delay=tau*R*C //pulse width in sec
9 printf(" Pulse width = %.2f sec",t_delay)

```

Scilab code Exa 21.8 Decimal to TTL Voltage Terms

```

1 //Ex21_8 Pg-1076
2 clc
3
4 dec=253 //decimal number
5 bin=dec2bin(dec) //binary value of 253
6 disp(" Binary value of 253 =")
7 disp(bin)
8 disp("In TTL for 1s it is represented in 5V and 0s
   as 0V")
9 disp(" Therefore 253 in TTL voltage terms is")
10 disp(" 5V,5V,5V,5V,5V,5V,0V,5V")

```
