

Scilab Textbook Companion for
Op-Amps And Linear Integrated Circuits
by R. A. Gayakwad¹

Created by

Amar Ds

BE

Instrumentation Engineering

Sri Jayachamarajendra College Of Engineering

College Teacher

Rathnnakara S

Cross-Checked by

TechPassion

May 19, 2016

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: Op-Amps And Linear Integrated Circuits

Author: R. A. Gayakwad

Publisher: PHI Learning Pvt. Ltd., New Delhi

Edition: 4

Year: 2004

ISBN: 978-81-203-2058-1

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	4
1 Introduction to Operational Amplifier	5
2 Interpretation of Data Sheets and Characteristics of an Opamp	9
3 An Opamp with Negative Feedback	12
4 The Practical Opamp	19
5 Frequency Response of an Opamp	35
6 General Linear Applications	38
7 Active Filters and Oscillators	51
8 Comparators and Converters	72
9 Specialized IC Applications	78

List of Scilab Codes

Exa 1.1.a	Collector current and dc voltage	5
Exa 1.1.b	Voltage gain of the opamp	7
Exa 1.1.c	Input resistance of the opamp	7
Exa 2.1.a	Output voltage for Openloop differential amplifier	9
Exa 2.1.b	Output voltage for openloop differential amplifier	10
Exa 2.2.a	Output voltage for inverting amplifier	10
Exa 2.2.b	Output voltage for inverting amplifier	11
Exa 3.1	Parameters Of voltageseries feedback amplifier	12
Exa 3.2	Parameters Of voltageseries feedback amplifier	13
Exa 3.3	Parameters of Voltageshunt feedback amplifier	14
Exa 3.4	Output voltage of voltage shunt feedback amplifier	15
Exa 3.5.a	Gain input resistance of the amplifier	16
Exa 3.5.b	Output voltage of an Opamp	16
Exa 3.6.a	Voltage gain and input resistance of the Opamp	17
Exa 3.6.b	Output voltage of the Opamp	18
Exa 4.1	Design of Compensating Network	19
Exa 4.2	Max Output offset voltage	20
Exa 4.3	Design of input offset voltage compensating network	20
Exa 4.4.a	Max Output offset voltage	21
Exa 4.4.b	Effect of input bias current	22
Exa 4.5.a	Max Output offset voltage	22
Exa 4.5.b	Effect of input bias current	23
Exa 4.6	Max Output offset voltage	23
Exa 4.7	Total Output offset voltage	24
Exa 4.8.a	Error voltage and output voltage	25
Exa 4.8.b	Error voltage and output voltage	26
Exa 4.9.a	Error voltage and output voltage	27
Exa 4.9.b	Output waveform	28

Exa 4.10.a	Error voltage and output voltage	28
Exa 4.10.b	Error voltage and output voltage	29
Exa 4.11.a	Output offset voltage	30
Exa 4.11.b	Output offset voltage	31
Exa 4.12	Output ripple voltage	32
Exa 4.13	Change in output offset voltage	32
Exa 4.14.a	Output voltage	33
Exa 4.14.b	Output common mode voltage	33
Exa 5.1	Maximum gain	35
Exa 5.2	Gain equation and break frequencies	36
Exa 5.3	Stability of voltage follower	36
Exa 6.1	Bandwidth of the amplifier	38
Exa 6.2.a	Bandwidth of the amplifier	39
Exa 6.2.b	Max output voltage swing	40
Exa 6.3	Components of peak amplifier	40
Exa 6.4	Output voltage	41
Exa 6.5	Output voltage	42
Exa 6.6	Output voltage	42
Exa 6.7	Output voltage	43
Exa 6.8	Change in resistance in straingage	44
Exa 6.9	Gain of the amplifier	44
Exa 6.10	Range of input voltage	45
Exa 6.11	Current and voltage drop	45
Exa 6.12.a	Load current	46
Exa 6.12.b	Output voltage	46
Exa 6.13	Range of output voltage	47
Exa 6.14	Change in output voltage	47
Exa 6.15	Output voltage of an integrator	48
Exa 6.16.a	Design of differentiator	49
Exa 6.16.b	Output waveform of differentiator	50
Exa 7.1	Design of low pass filter	51
Exa 7.2	Design of low pass filter	52
Exa 7.3	Frequency response of low pass filter	52
Exa 7.4.a	Design of second order low pass filter	54
Exa 7.4.b	Frequency response of second order highpass filter	55
Exa 7.5.a	Design of highpass filter	56
Exa 7.5.b	Frequency response of highpass filter	57
Exa 7.6.a	Determination of low cutoff frequency	59

Exa 7.6.b	Frequency response of second order highpass filter . . .	59
Exa 7.7.a	Design of wide bandpass filter	61
Exa 7.7.b	Frequency response of bandpass filter	62
Exa 7.7.c	Calculation of quality factor	63
Exa 7.8.a	Design of narrow bandpass filter	64
Exa 7.8.b	Design of narrow bandpass filter	65
Exa 7.9	Design of wide bandreject filter	65
Exa 7.10	Design of notch filter	66
Exa 7.11	Phase angle	66
Exa 7.12	Design of phase shift oscillator	67
Exa 7.13	Design of wein bridge oscillator	68
Exa 7.14	Design of quadrature oscillator	68
Exa 7.15	Design of squarewave oscillator	69
Exa 7.16	Design of triangular wave generator	69
Exa 7.17.a	Nominal frequency	70
Exa 7.17.b	Modulation in output frequency	71
Exa 8.1	Threshold voltage	72
Exa 8.2	Output voltage swing	73
Exa 8.3	Output frequencies	74
Exa 8.4	Output voltage	76
Exa 8.5	Output voltage	76
Exa 9.1	Second order inverting butterworth lowpass filter . . .	78
Exa 9.2	Second order inverting butterworth bandpass filter . .	79
Exa 9.3	Design of notch filter	80
Exa 9.4	Second order butterworth lowpass filter	81
Exa 9.5	Value of capacitor	81
Exa 9.6	Value of resistor	82
Exa 9.7	Value of t_c t_d and f_0	82
Exa 9.8	Freq of free running ramp generator	83
Exa 9.9	Value of f_{out} f_l f_c	84
Exa 9.10	Design of current source	84
Exa 9.11	Design of voltage regulator	85
Exa 9.12	Design of stepdown switching regulator	86
Exa 9.13	Design of stepdown switching regulator	88

Chapter 1

Introduction to Operational Amplifier

Scilab code Exa 1.1.a Collector current and dc voltage

```
1 //Chapter1
2 //Page.No-4,
3 //Example_1_1_a , Figure.No-1.2
4 //Collector current and dc voltage
5 //Given:
6 clear;clc;
7 Vcc=6;Vbe5=0.7;Vee=6;Vbe3=6.7;Vbe6=0.7;Vbe7=0.7; //
   Voltage in volts
8 Rc1=6.7*10^3; // Resistance in ohms
9 Ic1=rand();
10 Vc1=Vcc-Rc1*Ic1;
11 Ve4=Vc1-Vbe5;
12 I4=(Ve4+Vee)/(9.1*10^3+5.5*10^3);
13 Vb3=5.5*10^3*I4-Vee;
14 Ve3=Vb3-Vbe3;
15 Ie3=(Ve3+Vbe3)/3.3*10^3;
16 Ic1=1.08*10^-3/2.765; // Since Ie3=2*Ic1 ,
   substituting in above equation and simplifying
17 printf("\n Collector current Ic1 is = %.5f A \n",Ic1
```



```

    ) // Result
18 Vc1=Vcc-Rc1*Ic1;
19 printf("\n Voltage Vc1 is = %.2f V \n",Vc1) //
    Result
20 Ve4=Vc1-Vbe5;
21 printf("\n Voltage Ve4 is = %.2f V \n",Ve4) //
    Result
22 Ie4=(Ve4+Vee)/(29.2*10^3);
23 printf("\n Current Ie4 is = %.6f A \n",Ie4) //
    Result
24 Ic5=Ie4;
25 printf("\n Current Ic5 is = %.6f A \n",Ic5) //
    Result
26 Vc5=Vcc-3.8*10^3*Ic5;
27 printf("\n Voltage Vc5 is = %.2f V \n",Vc5) //
    Result
28 Ve6=Vc5-Vbe6;
29 printf("\n Voltage Ve6 is = %.2f V \n",Ve6) //
    Result
30 Ie6=(Ve6+Vee)/(15*10^3);
31 printf("\n Current Ie6 is = %.6f A \n",Ie6) //
    Result
32 Ve7=Ve6+Vbe7;
33 printf("\n Voltage Ve7 is = %.2f V \n",Ve7) //
    Result
34 I1=(Vcc-Ve7)/400;
35 printf("\n Current I1 is = %.6f A \n",I1) // Result
36 Ie8=I1;
37 printf("\n Current Ie8 is = %.6f A \n",Ie8) //
    Result
38 Ve8=-Vee+2*10^3*Ie8;
39 printf("\n Voltage Ve8 at the output terminal is = %
    .2f V \n",Ve8) // Result

```

Scilab code Exa 1.1.b Voltage gain of the opamp

```
1 //Chapter1
2 //Page.No-4,
3 //Example_1_1_b , Figure.No-1.2
4 //Voltage gain of the opamp
5 //Given:
6 clear;clc;
7 Ie1=0.39*10^-3;Ie4=0.298*10^-3;Ie6=0.678*10^-3; //
   Current in amps
8 Rc1=6.7*10^3;Rc5=3.8*10^3; // Resistance in ohms
9 beta_ac=150;
10 re1=(25*10^-3)/Ie1;
11 re2=re1;
12 re4=(25*10^-3)/Ie4;
13 re5=re4;
14 re6=(25*10^-3)/Ie6;
15 k=(Rc1*2*beta_ac*re4)/(Rc1+2*beta_ac*re4);
16 Ad1=k/re1;
17 printf("\n Voltage gain of the dual-input ,balanced
   output-differential amplifier is = %.2f \n",Ad1)
   // Result
18 k1=(Rc5*beta_ac*(re6+15*10^3))/(Rc5+beta_ac*(re6
   +15*10^3));
19 Ad2=k1/(2*re5);
20 printf("\n Voltage gain of the dual-input ,unbalanced
   output-differential amplifier is = %.1f \n",Ad2
   ) // Result
21 Ad=82.55*22.6; // Using Ad=Ad1*Ad2
22 printf("\n Overall gain of the op-amp is = %.2f \n"
   ,Ad) // Result
```

Scilab code Exa 1.1.c Input resistance of the opamp

```
1 //Chapter1
2 //Page.No-4,
3 //Example_1_1_c , Figure.No-1.2
4 //Input resistance of the opamp
5 //Given:
6 clear;clc;
7 beta_ac=150;
8 re1=64.1; // Resistance in ohms
9 Ri=2*beta_ac*re1;
10 printf("\n Input resistance Ri is = %.1f ohm \n",Ri)
    // Result
```

Chapter 2

Interpretation of Data Sheets and Characteristics of an Opamp

Scilab code Exa 2.1.a Output voltage for Openloop differential amplifier

```
1 //Chapter2
2 //Page.No-45, Figure.No-2.9
3 //Example_2_1_a
4 //Output voltage for open-loop differential
  amplifier
5 //Given:
6 clear;clc;
7 vin1=5*10^-6;vin2=-7*10^-6; // Both input voltages
  are in volts
8 A=200000; // Voltage gain
9 vo=A*(vin1-vin2); //Output voltage in volts
10 printf("\n Output voltage is vo = %.1f V dc \n",vo)
  // Result
```

Scilab code Exa 2.1.b Output voltage for openloop differential amplifier

```
1 //Chapter2
2 //Page.No-45, Figure.No-2.9
3 //EXAMPLE_2_1_b
4 //Output voltage for open-loop differential
  amplifier
5 //Given:
6 clear;clc;
7 vin1=10*10^-3;vin2=20*10^-3; // Both input voltages
  are in volts
8 A=200000; // Voltage gain
9 vo=A*(vin1-vin2); // Output voltage in volts
10 printf("\n Output voltage is vo = %.f V rms \n",vo)
  // Result
```

Scilab code Exa 2.2.a Output voltage for inverting amplifier

```
1 //Chapter2
2 //Page.No-46, Figure.No-2.10
3 //Example_2_2_a
4 //Output voltage for inverting amplifier
5 //Given:
6 clear;clc;
7 vin=20*10^-3; // Input voltage in volts
8 A=200000; // Voltage gain
9 vo=-(A*vin); // Output voltage in volts
```

```
10 printf("\n Output voltage is vo = %.f V \n",vo) //  
    Result
```

Scilab code Exa 2.2.b Output voltage for inverting amplifier

```
1 //Chapter2  
2 //Page.No-46, Figure.No-2.10  
3 //Example_2_2_b  
4 //Output voltage for inverting amplifier  
5 //Given:  
6 clear;clc;  
7 vin=-50*10^-6; // Input voltage in volts  
8 A=200000; // Voltage gain  
9 vo=-(A*vin); // Output voltage in volts  
10 printf("\n Output voltage is vo = %.f V \n",vo) //  
    Result
```

Chapter 3

An Opamp with Negative Feedback

Scilab code Exa 3.1 Parameters Of voltageseries feedback amplifier

```
1 // Chapter3
2 // Page.No-75, Figure.No-3.2
3 // Example_3_1
4 // Parameters of voltage-series feedback amplifier
5 // Given
6 clear;clc;
7 R1=1000;Rf=10000;
8 A=200000; // Open-loop voltage gain
9 Ri=2*10^6; // Input resistance without feedback
10 Ro=75; // Output resistance without feedback
11 fo=5; // Break frequency of an Op-amp
12 Vsat=13; // Saturation voltage
13 B=R1/(R1+Rf); // Gain of the feedback circuit
14 Af=A/(1+A*B); // Closed-loop voltage gain
15 printf("\n Closed-loop voltage gain is Af = %.2f \n"
        ,Af) // Result
16 RiF=Ri*(1+A*B); // Input resistance with feedback
17 printf("\n Input resistance with feedback is RiF = %
        .2f ohms \n",RiF) // Result
```

```

18 RoF=Ro/(1+A*B); // Output resistance with feedback
19 printf("\n Output resistance with feedback is RoF =
    %f ohms \n",RoF) // Result
20 fF=fo*(1+A*B); // Bandwidth with feedback
21 printf("\n Bandwidth with feedback is vo = %.1f Hz \
    n",fF) // Result
22 VooT=Vsat/(1+A*B); // Total output offset voltage
    with feedback
23 printf("\n Total output offset voltage with feedback
    is VooT = %f V \n",VooT) // Result

```

Scilab code Exa 3.2 Parameters Of voltageseries feedback amplifier

```

1 // Chapter3
2 // Page.No-83, Figure.No-3.7
3 // Example_3_2
4 // Parameters of voltage-series feedback amplifier
5 // Given
6 clear;clc;
7 R1=1000;Rf=10000;
8 A=200000; // Open-loop voltage gain
9 Ri=2*10^6; // Input resistance without feedback
10 Ro=75; // Output resistance without feedback
11 fo=5; // Break frequency of an Op-amp
12 Vsat=13; // Saturation voltage
13 B=1; // Gain of the feedback circuit and it is equal
    to 1 for voltage follower
14 Af=A/(1+A*B); // Closed-loop voltage gain
15 printf("\n Closed-loop voltage gain is Af = %.f \n",
    Af) // Result
16 RiF=Ri*(1+A*B); // Input resistance with feedback
17 printf("\n Input resistance with feedback is RiF = %
    .1f ohms \n",RiF) // Result

```



```

18 RoF=Ro/(1+A*B); // Output resistance with feedback
19 printf("\n Output resistance with feedback is RoF =
    %f ohms \n",RoF) // Result
20 fF=fo*(1+A*B); // Bandwidth with feedback
21 printf("\n Bandwidth with feedback is vo = %.1 f Hz \
    n",fF) // Result
22 VooT=Vsat/(1+A*B); // Total output offset voltage
    with feedback
23 printf("\n Total output offset voltage with feedback
    is VooT = %f V \n",VooT) // Result

```

Scilab code Exa 3.3 Parameters of Voltageshunt feedback amplifier

```

1 // Chapter3
2 // Page.No-86, Figure.No-3.8
3 // Example_3_3
4 // Parameters of voltage-shunt feedback amplifier
5 // Given
6 clear;clc;
7 R1=470;Rf=4.7*10^3;
8 A=200000; // Open-loop voltage gain
9 Ri=2*10^6; // Input resistance without feedback
10 Ro=75; // Output resistance without feedback
11 fo=5; // Break frequency of an Op-amp
12 Vsat=13; // Saturation voltage
13 K=Rf/(R1+Rf); // Voltage attenuation factor
14 B=R1/(R1+Rf); // Gain of the feedback circuit
15 Af=-(A*K)/(1+A*B); // Closed-loop voltage gain
16 printf("\n Closed-loop voltage gain is Af = %.f \n",
    Af) // Result
17 X=Rf/(1+A);
18 RiF=R1+(X*Ri)/(X+Ri); // Input resistance with
    feedback

```

```

19 printf("\n Input resistance with feedback is RiF = %
    .1 f ohms \n",RiF) // Result
20 RoF=Ro/(1+A*B); // Output resistance with feedback
21 printf("\n Output resistance with feedback is RoF =
    %f ohms \n",RoF) // Result
22 fF=fo*(1+A*B)/K; // Bandwidth with feedback
23 printf("\n Bandwidth with feedback is vo = %.2 f Hz \
    n",fF) // Result
24 VooT=Vsat/(1+A*B); // Total output offset voltage
    with feedback
25 printf("\n Total output offset voltage with feedback
    is VooT = %f V \n",VooT) // Result

```

Scilab code Exa 3.4 Output voltage of voltage shunt feedback amplifier

```

1 // Chapter3
2 // Page.No-86, Figure.No-3.8
3 // Example_3_4
4 // Output voltage of voltage-shunt feedback
    amplifier
5 // Given
6 clear;clc;
7 R1=470;Rf=4.7*10^3;
8 A=200000; // Open-loop voltage gain
9 vin=1; // Input voltage in volts
10 K=Rf/(R1+Rf); // Voltage attenuation factor
11 B=R1/(R1+Rf); // Gain of the feedback circuit
12 Af=-(A*K)/(1+A*B); // Closed-loop voltage gain
13 vo=Af*vin; // Output voltage
14 printf("\n Output voltage is vo = %.f V \n",vo) //
    Result
15 t=0:0.1:2*pi;
16 vo=-10*sin(t);

```

```

17 plot(t,vo);
18 title('Output Voltage');
19 xlabel('t');
20 ylabel('Vo');

```

Scilab code Exa 3.5.a Gain input resistance of the amplifier

```

1 // Chapter3
2 // Page.No-96, Figure.No-3.14
3 // Example_3_5_a
4 // Gain Input resistance of the amplifier
5 // Given
6 clear;clc;
7 R1=1000;R2=1000;
8 Rf=10*10^3;R3=10*10^3;
9 AD=-Rf/R1; // Voltage gain
10 printf("\n Voltage gain is AD = %.f \n",AD) //
    Result
11 RiFx=R1; // Input resistance of inverting amplifier
12 printf("\n Input resistance of inverting amplifier
    is RiFx = %.f ohms \n",RiFx) // Result
13 RiFy=R2+R3; // Input resistance of non-inverting
    amplifier
14 printf("\n Input resistance of non-inverting
    amplifier is RiFy = %.f ohms \n",RiFy) // Result

```

Scilab code Exa 3.5.b Output voltage of an Opamp

```

1 // Chapter3

```

```

2 // Page.No-96, Figure.No-3.14
3 // Example_3_5_b
4 // Output voltage of an Op-amp
5 // Given
6 clear;clc;
7 vx=2.7;vy=3; //Both input voltages are in volts
8 Rf=10*10^3;R1=1000; // Both are in ohms
9 AD=-Rf/R1; // Voltage gain
10 vxy=vx-vy;
11 vo=AD*vxy; // Output voltage
12 printf("\n Output voltage is vo = %.f V \n",vo) //
    Result

```

Scilab code Exa 3.6.a Voltage gain and input resistance of the Opamp

```

1 // Chapter3
2 // Page.No-99, Figure.No-3.16
3 // Example_3_6_a
4 // Voltage gain and input resistance of Op-amp
5 // Given
6 clear;clc;
7 R1=680;R3=680; // Both are in ohms
8 RF=6800;R2=6800; // Both are in ohms
9 Ri=2*10^6; // Open-loop input resistance of the op-
    amp
10 vx=-1.5;vy=-2; // Both are in volts
11 A=200000; // Open-loop Gain
12 AD=1+RF/R1; // Voltage gain
13 printf("\n Voltage gain is AD = %.f \n",AD) //
    Result
14 B=R2/(R2+R3);
15 RiFy=Ri*(1+A*B); // Input resistance of first stage
    amplifier

```

```

16 printf("\n Input resistance of first stage amplifier
    is RiFy = %.1f ohms \n",RiFy) // Result
17 B=R1/(R1+RF);
18 RiFx=Ri*(1+A*B); // Input resistance of second stage
    amplifier
19 printf("\n Input resistance of second stage
    amplifier is RiFx = %.1f ohms \n",RiFx) // Result

```

Scilab code Exa 3.6.b Output voltage of the Opamp

```

1 // Chapter3
2 // Page.No-99, Figure.No-3.16
3 // Example_3_6_b
4 // Output voltage of the Op-amp
5 // Given
6 clear;clc;
7 R1=680;RF=6800 // Both are in ohms
8 vx=-1.5;vy=-2; // Both input voltages are in volts
9 AD=1+RF/R1; // Voltage gain
10 vxy=vx-vy;
11 vo=AD*vxy; // Output voltage
12 printf("\n Output voltage is vo = %.1f V \n",vo) //
    Result

```

Chapter 4

The Practical Opamp

Scilab code Exa 4.1 Design of Compensating Network

```
1 // Chapter4
2 // Page.No-114
3 // Example_4.1
4 // Design of Compensating Network
5 // Given
6 clear;clc;
7 V=10 // Supply voltage
8 Vio=10*10^-3; // Input offset voltage
9 Rc=10; // Assumption
10 Rb=(V/Vio)*Rc;
11 printf("\n Resistance Rb is = %.f ohms \n",Rb) //
    Result
12 Ra=Rb/2.5; // Since Rb>Rmax,let us choose Rb=10*Rmax
    where Rmax=Ra/4
13 printf("\n Resistance Ra is = %.f ohms \n",Ra) //
    Result
```

This code can be downloaded from the website www.scilab.in

Scilab code Exa 4.2 Max Output offset voltage

```
1 // Chapter4
2 // Page.No-121, Figure.No-4.13
3 // Example_4_2
4 // Max Output offset voltage
5 // Given
6 clear;clc;
7 R1=1*10^3;Rf=10*10^3;
8 Vio=10*10^-3; // Input offset voltage
9 Aoo=1+Rf/R1; // To find max value of Voo,we reduce
   input voltage vin to zero.
10 Voo=Aoo*Vio; // Max output offset voltage
11 printf("\n Max output offset voltage is = %.3f V dc
   \n",Voo) // Result
```

Scilab code Exa 4.3 Design of input offset voltage compensating network

```
1 // Chapter4
2 // Page.No-121, Figure.No-4.14
3 // Example_4_3
4 // Design of input offset voltage-compensating
   network
5 // Given
6 clear;clc;
```

```

7 R1=1*10^3;Rf=10*10^3;Rc=10;
8 Af=1+Rf/(R1+Rc); // Closed loop gain of non-
    inverting amplifier
9 printf("\n Closed loop gain of non-inverting
    amplifier is = %.1f \n",Af) // Result

```

This code can be downloaded from the website www.scilab.in

Scilab code Exa 4.4.a Max Output offset voltage

```

1 // Chapter4
2 // Page.No-127, Figure.No-4.19
3 // Example_4_4_a
4 // Max Output offset voltage
5 // Given
6 clear;clc;
7 R1=470;Rf=47*10^3;
8 Vio=6*10^-3;
9 Ib=500*10^-9;
10 Vs=15;
11 // Max output offset voltage due to input offset
    voltage ,Vio is :
12 Voo=(1+Rf/R1)*Vio; // Max output offset voltage
13 printf("\n Max output offset voltage is = %.3f V dc
    \n",Voo) // Result
14 // Max output offset voltage due to input offset
    voltage ,Ib is :
15 VoIb=Rf*Ib; // Max output offset voltage
16 printf("\n Max output offset voltage due to input
    offset current ,Ib is = %.6f V dc \n",VoIb) //
    Result

```

Scilab code Exa 4.4.b Effect of input bias current

```
1 // Chapter4
2 // Page.No-127, Figure.No-4.19
3 // Example_4_4_b
4 // Effect of input bias current
5 // Given
6 clear;clc;
7 R1=470;Rf=47*10^3;
8 ROM=R1*Rf/(R1+Rf); // Parallel combination of R1 and
   Rf
9 printf("\n Parallel combination of R1 and Rf,i.e ROM
   is = %.1f ohm \n",ROM) // Approximately the
   value is 47 ohm
```

Scilab code Exa 4.5.a Max Output offset voltage

```
1 // Chapter4
2 // Page.No-127, Figure.No-4.19
3 // Example_4_5_a
4 // Max Output offset voltage
5 // Given
6 clear;clc;
7 R1=1*10^3;Rf=100*10^3;
8 Vio=6*10^-3;
9 Ib=500*10^-9;
10 Vs=15;
```

```

11 // Max output offset voltage due to input offset
    voltage ,Vio is :
12 Voo=(1+Rf/R1)*Vio; // Max output offset voltage
13 printf("\n Max output offset voltage due to input
    offset voltage ,Vio is = %.4f V dc \n",Voo) //
    Result
14 // Max output offset voltage due to input offset
    voltage ,Ib is :
15 VoIb=Rf*Ib; // Max output offset voltage
16 printf("\n Max output offset voltage due to input
    offset current ,Ib is = %.4f V dc \n",VoIb) //
    Result

```

Scilab code Exa 4.5.b Effect of input bias current

```

1 // Chapter4
2 // Page.No-127, Figure.No-4.19
3 // Example_4_4_b
4 // Effect of input bias current
5 // Given
6 clear;clc;
7 R1=1*10^3;Rf=100*10^3;
8 ROM=R1*Rf/(R1+Rf); // Parallel combination of R1 and
    Rf
9 printf("\n Parallel combination of R1 and Rf,i.e ROM
    is = %.f ohm \n",ROM)

```

Scilab code Exa 4.6 Max Output offset voltage

```

1 // Chapter4
2 // Page.No-130, Figure.No-4.21
3 // Example_4_6
4 // Max Output offset voltage
5 // Given
6 clear;clc;
7 Iio=200*10^-9; // Input offset current
8 Rf=100*10^3;
9 VoIio=Rf*Iio; // Max output offset voltage
10 printf("\n Max output offset voltage due to input
    offset current ,Ib is = %.4f V dc \n",VoIio) //
    Result

```

Scilab code Exa 4.7 Total Output offset voltage

```

1 // Chapter4
2 // Page.No-132, Figure.No-4.22(a)
3 // Example_4_7
4 // Total Output offset voltage
5 // Given
6 clear;clc;
7 R1=1*10^3;Rf=10*10^3;
8 Vio=7.5*10^-3; // Max input offset voltage
9 Iio=50*10^-9; // Max input offset current
10 Ib=250*10^-9; // Max input bias current
11 // For figure 4.22(a)
12 VooT=(1+Rf/R1)*Vio+(Rf*Ib); // Since the current
    generated output offset voltage is due to input
    bias current Ib
13 printf("\n Max total output offset voltage due to
    input offset current ,Ib is = %.4f V \n",VooT) //
    Result
14

```

```

15 // For figure 4.22(b)
16 VooT=(1+Rf/R1)*Vio+(Rf*Iio); // Since the current
    generated output offset voltage is due to input
    offset current Ib
17 printf("\n Max total output offset voltage due to
    input offset current ,Ib is = %.4f V \n",VooT) //
    Result

```

Scilab code Exa 4.8.a Error voltage and output voltage

```

1 // Chapter4
2 // Page.No-136, Figure.No-4.24
3 // Example_4.8_a
4 // Error voltage and output voltage
5 // Given
6 clear;clc;
7 delta_Vio=(30*10^-6); // Change in input offset
    voltage
8 delta_T=1; // Unit change in temperature
9 delta_Iio=(300*10^-12); // Change in input offset
    current
10 Vs=15;
11 R1=1*10^3;Rf=100*10^3;Rl=10*10^3;
12 Vin=1*10^-3; // Input voltage
13 k=25; // Amplifier is nulled at 25 deg
14 T=35-k; // Change in temperature
15 Ev=(1+Rf/R1)*(delta_Vio/delta_T)*T + Rf*(delta_Iio/
    delta_T)*T; // Error voltage
16 printf("\n Error voltage is = %.4f V \n",Ev) //
    Result
17 Vo=- (Rf/R1)*Vin+Ev; // Output voltage
18 printf("\n Output voltage is = %.4f V \n",Vo) //
    Result

```

```

19 // (OR)
20 Vo=-(Rf/R1)*Vin-Ev; // Output voltage
21 printf("\n Output voltage is = %.4f V \n",Vo) //
    Result

```

Scilab code Exa 4.8.b Error voltage and output voltage

```

1 // Chapter4
2 // Page.No-136, Figure.No-4.24
3 // Example_4_8_b
4 // Error voltage and output voltage
5 // Given
6 clear;clc;
7 delta_Vio=(30*10^-6); // Change in input offset
    voltage
8 delta_T=1; // Unit change in temperature
9 delta_Iio=(300*10^-12); // Change in input offset
    current
10 Vs=15;
11 R1=1*10^3;Rf=100*10^3;Rl=10*10^3;
12 Vin=10*10^-3; // Input voltage
13 k=25; // Amplifier is nulled at 25 deg
14 T=35-k; // Change in temperature
15 Ev=(1+Rf/R1)*(delta_Vio/delta_T)*T + Rf*(delta_Iio/
    delta_T)*T; // Error voltage
16 printf("\n Error voltage is = %.4f V \n",Ev) //
    Result
17 Vo=-(Rf/R1)*Vin+Ev; // Output voltage
18 printf("\n Output voltage is = %.4f V \n",Vo) //
    Result
19 // (OR)
20 Vo=-(Rf/R1)*Vin-Ev; // Output voltage
21 printf("\n Output voltage is = %.4f V \n",Vo) //

```

Result

Scilab code Exa 4.9.a Error voltage and output voltage

```
1 // Chapter4
2 // Page.No-136, Figure.No-4.24
3 // Example_4_9_a
4 // Error voltage and output voltage
5 // Given
6 clear;clc;
7 delta_Vio=(30*10^-6); // Change in input offset
   voltage
8 delta_T=1; // Unit change in temperature
9 delta_Iio=(300*10^-12); // Change in input offset
   current
10 Vs=15;
11 R1=1*10^3;Rf=100*10^3;Rl=10*10^3;
12 Vin=10*10^-3; // Input voltage
13 k=25; // Amplifier is nulled at 25 deg
14 T=55-k; // Change in temperature
15 Ev=(1+Rf/R1)*(delta_Vio/delta_T)*T + Rf*(delta_Iio/
   delta_T)*T; // Error voltage
16 printf("\n Error voltage is = %.4f V dc \n",Ev) //
   Result
17 Vo=-(Rf/R1)*Vin+Ev; // Output voltage
18 printf("\n Output voltage is = %.4f V \n",Vo) //
   Result
19 // (OR)
20 Vo=-(Rf/R1)*Vin-Ev; // Output voltage
21 printf("\n Output voltage is = %.4f V \n",Vo) //
   Result
```

Scilab code Exa 4.9.b Output waveform

```
1 // Chapter4
2 // Page.No-136, Figure.No-4.24
3 // Example_4_9_b
4 // Output waveform
5 // Given
6 clear;clc;
7 t =0:0.1:2*%pi;
8 y = -1000*sin(t)+91.8;
9 a = gca();
10 a.x_label.text = 'Time';
11 a.y_label.text = 'Voltage';
12 a.title.text = 'Output waveform';
13 plot2d(t,y);
14 t1=0:0.1:2*%pi;
15 y1=91.8*(t1>=0);
16 b=gca();
17 b.line_style=3;
18 plot2d(t1,y1);
```

Scilab code Exa 4.10.a Error voltage and output voltage

```
1 // Chapter4
2 // Page.No-141, Figure.No-4.26
3 // Example_4_10_a
4 // Error voltage and output voltage
5 // Given
6 clear;clc;
```

```

7 delta_Vio=(30*10^-6); // Change in input offset
  voltage
8 delta_T=1; // Unit change in temperature
9 delta_Iio=(300*10^-12); // Change in input offset
  current
10 Vs=15;
11 R1=1*10^3;Rf=100*10^3;
12 Vin=1*10^-3; // Input voltage
13 k=25; // Amplifier is nulled at 25 deg
14 T=35-k; // Change in temperature
15 Ev=(1+Rf/R1)*(delta_Vio/delta_T)*T + Rf*(delta_Iio/
  delta_T)*T; // Error voltage
16 printf("\n Error voltage is = %.4f V dc \n",Ev) //
  Result
17 Vo=(1+Rf/R1)*Vin+Ev; // Output voltage
18 printf("\n Output voltage is = %.4f V \n",Vo) //
  Result
19 // (OR)
20 Vo=(1+Rf/R1)*Vin-Ev; // Output voltage
21 printf("\n Output voltage is = %.4f V \n",Vo) //
  Result

```

Scilab code Exa 4.10.b Error voltage and output voltage

```

1 // Chapter4
2 // Page.No-141, Figure.No-4.26
3 // Example_4_10_b
4 // Error voltage and output voltage
5 // Given
6 clear;clc;
7 delta_Vio=(30*10^-6); // Change in input offset
  voltage
8 delta_T=1; // Unit change in temperature

```



```

9 delta_Iio=(300*10^-12); // Change in input offset
  current
10 Vs=15;
11 R1=1*10^3;Rf=100*10^3;Rl=10*10^3;
12 Vin=10*10^-3; // Input voltage
13 k=25; // Amplifier is nulled at 25 deg
14 T=35-k; // Change in temperature
15 Ev=(1+Rf/R1)*(delta_Vio/delta_T)*T + Rf*(delta_Iio/
  delta_T)*T; // Error voltage
16 printf("\n Error voltage is = %.4f V dc \n",Ev) //
  Result
17 Vo=(1+Rf/R1)*Vin+Ev; // Output voltage
18 printf("\n Output voltage is = %.4f V dc \n",Vo) //
  Result
19 // (OR)
20 Vo=(1+Rf/R1)*Vin-Ev; // Output voltage
21 printf("\n Output voltage is = %.4f V dc \n",Vo) //
  Result

```

Scilab code Exa 4.11.a Output offset voltage

```

1 // Chapter4
2 // Page.No-141, Figure.No-4.28
3 // Example_4_11_a
4 // Output offset voltage
5 // Given
6 clear;clc;
7 delta_Vio=15.85*10^-6; // Change in input offset
  voltage
8 delta_V=1; // Unit change in supply voltage
9 V=2; // Change in supply voltage
10 R1=1*10^3;Rf=100*10^3;
11 delta_Voo=(1+Rf/R1)*(delta_Vio/delta_V)*V; // Change

```

```

    in output offset voltage
12 printf("\n Change in output offset voltage is = %.4f
    V \n",delta_Voo) // Result

```

Scilab code Exa 4.11.b Output offset voltage

```

1 // Chapter4
2 // Page.No-141, Figure.No-4.28
3 // Example_4_11_b
4 // Output offset voltage
5 // Given
6 clear;clc;
7 delta_Vio=15.85*10^-6; // Change in input offset
    voltage
8 delta_V=1; // Unit change in supply voltage
9 V=2; // Change in supply voltage
10 Vin=10*10^-3;
11 R1=1*10^3;Rf=100*10^3;
12 delta_Voo=(1+Rf/R1)*(delta_Vio/delta_V)*V; // Output
    offset voltage
13 Vo=(-Rf/R1)*Vin+delta_Voo; // Total output offset
    voltage
14 printf("\n Total output offset voltage is = %.4f V \
    n",Vo) // Result
15 // (OR)
16 Vo=(-Rf/R1)*Vin-delta_Voo; // Total output offset
    voltage
17 printf("\n Total output offset voltage is = %.4f V \
    n",Vo) // Result

```

Scilab code Exa 4.12 Output ripple voltage

```
1 // Chapter4
2 // Page.No-141, Figure.No-4.28(b)
3 // Example_4_12
4 // Output ripple voltage
5 // Given
6 clear;clc;
7 delta_Vio=15.85*10^-6; // Change in input offset
    voltage
8 delta_V=1; // Unit change in supply voltage
9 V=10*10^-3; // Change in supply voltage
10 R1=1*10^3;Rf=100*10^3;
11 delta_Voo=(1+Rf/R1)*(delta_Vio/delta_V)*V; // Change
    in output offset voltage
12 printf("\n Change in output offset voltage is = %.6 f
    V \n",delta_Voo) // Result
```

Scilab code Exa 4.13 Change in output offset voltage

```
1 // Chapter4
2 // Page.No-136, Figure.No-4.24
3 // Example_4_13
4 // Change in output offset voltage
5 // Given
6 clear;clc;
7 delta_Vio=5*10^-6; // Change in input offset voltage
8 delta_t=1; // Unit change in time
```

```

9 delta_Iio=2*10^-9; // Change in input offset current
10 t=4; // Time elapsed(weeks)
11 R1=1*10^3;Rf=100*10^3;Rl=10*10^3;
12 delta_Voot=(1+Rf/R1)*(delta_Vio/delta_t)*t+Rf*(
    delta_Iio/delta_t)*t; // Change in output offset
    voltage
13 printf("\n Change in output offset voltage is = %.4f
    V \n",delta_Voot) // Result

```

Scilab code Exa 4.14.a Output voltage

```

1 // Chapter4
2 // Page.No-153, Figure.No-4.32
3 // Example_4_14_a
4 // Output voltage
5 // Given
6 clear;clc;
7 R1=1*10^3;R2=1*10^3;Rf=10*10^3;R3=10*10^3;
8 vd=5*10^-3; // Differential voltage
9 vcm=2*10^-3; // Common-mode voltage
10 Ad=Rf/R1; // Closed-loop differential gain
11 vo=Ad*vd; // Output voltage
12 printf("\n Output voltage is = %.3f V \n",vo) //
    Result

```

Scilab code Exa 4.14.b Output common mode voltage

```

1 // Chapter4
2 // Page.No-153, Figure.No-4.32

```

```

3 // Example_4_14_b
4 // Output common-mode voltage
5 // Given
6 clear;clc;
7 R1=1*10^3;R2=1*10^3;Rf=10*10^3;R3=10*10^3;
8 vd=5*10^-3; // Differential voltage
9 vcm=2*10^-3; // Common-mode voltage
10 Ad=Rf/R1; // Closed-loop differential gain
11 CMRRdb=90
12 CMRR=10^(90/20); // Using CMRRdb=20*log10(CMRR), to
    convert the CMRR(dB) value into its equivalent
    numerical value
13 printf("\n CMRR is = %.2f \n",CMRR) // Result
14 vocm=(Ad*vcm)/CMRR; // Output common-mode voltage
15 printf("\n Output common-mode voltage is = %.8f V \n
    ",vocm) // Result

```

Chapter 5

Frequency Response of an Opamp

Scilab code Exa 5.1 Maximum gain

```
1 // Chapter5
2 // Page.No-171, Figure.No-5.4
3 // Example_5_1
4 // Maximum gain
5 // Given
6 clear;clc;
7 fo=5; // Break freq of the op-amp in Hz
8 s=%s;
9 A=200000; // Gain of the op-amp at 0 Hz
10 H=syslin('c',(A*fo*2*%pi)/((fo*2*%pi)+s));
11 fmin=1;
12 fmax=100000;
13 bode(H,fmin,fmax);
14 Aol=40;
15 printf("\n Maximum gain is = %.f dB \n ",Aol); //
    From the graph
```

Scilab code Exa 5.2 Gain equation and break frequencies

```
1 //Chapter5
2 //Page.No-172, Figure.No-5.5
3 //Example_5_2
4 //Gain equation and break frequencies
5 //Given:
6 clear;clc;
7 phase=-157.5; // Phase shift at about 3 MHz
8 f=3*10^6;
9 disp("Gain equation is  $A_{ol}(f)=A/((1+(f/fo1)*j)*(1+(f/fo2)*j))$ ,where fo1-first break frequency and fo2
    -second break frequency") // From the figure
10 fo1=6;
11 printf("\n First break frequency fo1 is = %.f Hz \n"
    ,fo1) // From the graph
12 k=-atand(f/fo1)-phase;
13 fo2=f/tand(k);
14 printf("\n Second break frequency fo2 is = %.1f Hz \n"
    ,fo2) // Result
```

Scilab code Exa 5.3 Stability of voltage follower

```
1 // Chapter5
2 // Page.No-83, Figure.No-3.7
3 // Example_5_3
4 // Stability of voltage follower
5 // Given
```

```
6 clear;clc;
7 fo=5; // Break freq of the op-amp in Hz
8 s=%s;
9 A=200000; // Gain of the op-amp at 0 Hz
10 H=syslin('c',(A*fo*2*pi)/((fo*2*pi)+s));
11 fmin=10;
12 fmax=1000000;
13 bode(H,fmin,fmax);
14 Aol=0;
15 printf("\n Magnitude at which voltage follower is
    stable is = %.f dB \n ",Aol); // From the graph
```

Chapter 6

General Linear Applications

Scilab code Exa 6.1 Bandwidth of the amplifier

```
1 // Chapter6
2 // Page.No-193, Figure.No-6.3(a)
3 // Example_6_1
4 // Bandwidth of the amplifier
5 // Given
6 clear;clc;
7 R1=100;Rf=1*10^3;Rin=50;Rl=10*10^3;
8 Ci=0.1*10^-6; // Capacitance b/w 2 stages being
   coupled
9 RiF=R1; // ac input resistance of the second stage
10 Ro=Rin; // ac output resistance of the 1st stage
11 UGB=10^6; // Unity gain bandwidth
12 fl=1/(2*%pi*Ci*(RiF+Ro)); // Low-freq cutoff
13 printf("\n Low-freq cutoff is = %.1f Hz \n",fl) //
   Result
14 K=Rf/(R1+Rf);
15 Af=-Rf/R1; // closed loop voltage gain
16 fh=UGB*K/abs(Af); // High-freq cutoff
17 printf("\n High-freq cutoff is = %.1f Hz \n",fh) //
   Result
18 BW=fh-fl; // Bandwidth
```

```
19 printf("\n Bandwidth is = %.1f Hz \n",BW) // Result
```

Scilab code Exa 6.2.a Bandwidth of the amplifier

```
1 // Chapter6
2 // Page.No-193, Figure.No-6.4(c)
3 // Example_6_2_a
4 // Bandwidth of the amplifier
5 // Given
6 clear;clc;
7 R1=100*10^3;R2=100*10^3;R3=100*10^3;Rf=1*10^6;Rin
   =50;
8 Ci=0.1*10^-6; // Capacitance b/w 2 stages being
   coupled
9 Ro=Rin; // ac output resistance of the 1st stage
10 Vcc=15;
11 UGB=10^6; // Unity gain bandwidth
12 Rif=R2*R3/(R2+R3); // since Ri*(1+A*B)>>R2 or R3
13 f1=1/(2*%pi*Ci*(Rif+Ro)); // low-freq cutoff
14 printf("\n Low-freq cutoff is = %.1f Hz \n",f1) //
   Result
15 K=Rf/(R1+Rf);
16 Af=-Rf/R1; // closed loop voltage gain
17 fh=UGB*K/abs(Af); // High-freq cutoff
18 printf("\n High-freq cutoff is = %.1f Hz \n",fh) //
   Result
19 BW=fh-f1; // Bandwidth
20 printf("\n Bandwidth is = %.1f Hz \n",BW) // Result
```

Scilab code Exa 6.2.b Max output voltage swing

```
1 // Chapter6
2 // Page.No-193, Figure.No-6.4(c)
3 // Example_6_2_b
4 // Max output voltage swing
5 // Given
6 clear;clc;
7 R1=100*10^3;R2=100*10^3;R3=100*10^3;Rf=1*10^6;Rin
  =50;
8 Ci=0.1*10^-6; // Capacitance b/w 2 stages being
  coupled
9 Ro=Rin; // ac output resistance of the 1st stage
10 UGB=10^6; // Unity gain bandwidth
11 Vcc=15;
12 printf("\n The ideal maximum output voltage swing is
  = %.f V pp \n",Vcc)
```

Scilab code Exa 6.3 Components of peak amplifier

```
1 // Chapter6
2 // Page.No-193, Figure.No-6.5(a)
3 // Example_6_3
4 // Components of peak amplifier
5 // Given
6 clear;clc;
7 fp=16*10^3; // Peak frequency
8 Af=10; // Gain at peak frequency
9 C=0.01*10^-6; // Assume
10 L=1/(((2*pi*fp)^2)*10^-8); // Simplifying fp=1/(2*
  pi*sqrt(L*C))
11 printf("\n Inductance is = %.4f H \n",L)
12 L=10*10^-3; // Approximate
```

```

13 R=30; // Assume the value of internal resistance of
    the inductor
14 Xl=2*%pi*fp*L; // Inductive reactance
15 Qcoil=Xl/R; // Figure of merit of the coil
16 printf("\n Figure of merit of the coil is = %.1f \n"
    ,Qcoil)
17 Rp=(Qcoil)^2*R; // Parallel resistance of the tank
    circuit
18 printf("\n Parallel resistance of the tank circuit
    is = %.1f ohm \n",Rp)
19 R1=100; // Assume the value of internal resistance
    of the coil
20 Rf=-Rp/(1-(Rp/(Af*R1))); // Simplifying Af=(Rf||Rp)/
    R1
21 printf("\n Feedback resistance is = %.1f ohm \n",Rf)

```

Scilab code Exa 6.4 Output voltage

```

1 // Chapter6
2 // Page.No-200, Figure.No-6.6
3 // Example_6_4
4 // Output voltage
5 // Given
6 clear;clc;
7 Va=1;Vb=2;Vc=3; // Input voltages in volts
8 Ra=3*10^3;Rb=3*10^3;Rc=3*10^3;Rf=1*10^3;
9 Vo=-((Rf/Ra)*Va+(Rf/Rb)*Vb+(Rf/Rc)*Vc); // Output
    voltage
10 printf("\n Output voltage is = %.f V \n",Vo)

```

Scilab code Exa 6.5 Output voltage

```
1 // Chapter6
2 // Page.No-203, Figure.No-6.7
3 // Example_6_5
4 // Output voltage
5 // Given
6 clear;clc;
7 Va=2;Vb=-3;Vc=4; // Input voltages in volts
8 R1=1*10^3;Rf=2*10^3;
9 V1=(Va+Vb+Vc)/3; // Voltage at non-inverting
   terminal
10 printf("\n Voltage at non-inverting terminal is = %.
   f V \n",V1)
11 Vo=(1+Rf/R1)*V1; // Output voltage
12 printf("\n Output voltage is = %.f V \n",Vo)
```

Scilab code Exa 6.6 Output voltage

```
1 // Chapter6
2 // Page.No-205, Figure.No-6.9
3 // Example_6_6
4 // Output voltage
5 // Given
6 clear;clc;
7 Va=2;Vb=3;Vc=4;Vd=5; // Input voltages in volts
8 R=1*10^3;
9 Vo=-Va-Vb+Vc+Vd; // Output voltage
```

```
10 printf("\n Output voltage is = %.f V \n",Vo)
```

Scilab code Exa 6.7 Output voltage

```
1 // Chapter6
2 // Page.No-209, Figure.No-6.12
3 // Example_6_7
4 // Output voltage
5 // Given
6 clear;clc;
7 R1=1*10^3;Rf=4.7*10^3;Ra=100*10^3;Rb=100*10^3;Rc
   =100*10^3;
8 Vdc=5;
9 Rt=100*10^3; // Resistance of a thermistor
10 temp_coeff=1*10^3;
11
12 // Output voltage at 0 degree
13 delta_R=-temp_coeff*(0-25); // Change in resistance
14 R=Ra; // Ra=Rb=Rc=R
15 Vo=((Rf*delta_R)/(R1*4*R))*Vdc;
16 printf("\n Output voltage at 0 degree is = %.2f V \n
   ",Vo)
17
18 // Output voltage at 100 degree
19 delta_R=-temp_coeff*(100-25); // Change in
   resistance
20 Vo=((Rf*delta_R)/(R1*4*R))*Vdc;
21 printf("\n Output voltage at 100 degree is = %.2f V
   \n",Vo)
```

Scilab code Exa 6.8 Change in resistance in straingage

```
1 // Chapter6
2 // Page.No-209, Figure.No-6.12
3 // Example_6_7
4 // Change in resistance in straingage
5 // Given
6 clear;clc;
7 A=-100; // Gain of the differential instrumentation
      amplifier
8 Ra=100;Rb=100;Rc=100;
9 Vdc=10;Vo=1;
10 R=Ra; // Ra=Rb=Rc=R
11 delta_R=(Vo*R)/(Vdc*abs(A)); // Change in resistance
12 printf("\n Change in resistance is = %.1f ohm \n",
      delta_R)
```

Scilab code Exa 6.9 Gain of the amplifier

```
1 // Chapter6
2 // Page.No-216, Figure.No-6.14(a)
3 // Example_6_9
4 // Gain of the amplifier
5 // Given
6 clear;clc;
7 Vo=3.7;Vin=100*10^-3;
8 R1=100; // Assume
9 Rf=0.5*((Vo*R1)/Vin-1); // Feedback resisrance
```

```

10 printf("\n Feedback resisrance is = %.1f ohm \n",Rf)
11 A=(1+2*Rf/R1); // Gain of the differential amplifier
12 printf("\n Gain of the differential amplifier is = %
    .1f \n",A)

```

Scilab code Exa 6.10 Range of input voltage

```

1 // Chapter6
2 // Page.No-220, Figure.No-6.17
3 // Example_6_10
4 // Range of input voltage
5 // Given
6 clear;clc;
7 R1min=1*10^3;R1max=6.8*10^3;
8 io=1*10^-3; // Meter current for full-wave
    rectification
9 vin_min=1.1*R1min*io; // Minimum input voltage
10 printf("\n Minimum input voltage is = %.1f V \n",
    vin_min)
11 vin_max=1.1*R1max*io; // Maximum input voltage
12 printf("\n Maximum input voltage is = %.2f V \n",
    vin_max)

```

Scilab code Exa 6.11 Current and voltage drop

```

1 // Chapter6
2 // Page.No-222, Figure.No-6.18
3 // Example_6_11
4 // Current and voltage drop

```



```

5 // Given
6 clear;clc;
7 Vin=0.5;Vo=1.2;
8 R1=100;
9 Io=Vin/R1; // Current through diode
10 printf("\n Current through diode is = %.4f A \n",Io)
11 Vd=Vo-Vin; // Voltage drop across diode
12 printf("\n Voltage drop across diode is = %.1f V \n"
    ,Vd)

```

Scilab code Exa 6.12.a Load current

```

1 // Chapter6
2 // Page.No-222, Figure.No-6.19
3 // Example_6_12_a
4 // Load current
5 // Given
6 clear;clc;
7 Vin=5;V1=1;
8 R=10*10^3;
9 I1=Vin/R; // Load current
10 printf("\n Load current is = %.5f A \n",I1)

```

Scilab code Exa 6.12.b Output voltage

```

1 // Chapter6
2 // Page.No-222, Figure.No-6.19
3 // Example_6_12_b
4 // Output voltage

```

```

5 // Given
6 clear;clc;
7 Vin=5;V1=1;
8 R=10*103;
9 Vo=2*V1; // Output voltage
10 printf("\n Output voltage is = %.f V \n",Vo)

```

Scilab code Exa 6.13 Range of output voltage

```

1 // Chapter6
2 // Page.No-222, Figure.No-6.20
3 // Example_6_13
4 // Range of output voltage
5 // Given
6 clear;clc;
7 R1=1*103;Rf=2.7*103;
8 Vref=2;
9 Io=0; // Since all the binary inputs D0 to D7 are
    logic zero
10 Vo_min=Io*Rf; // Minimum output voltage
11 printf("\n Minimum output voltage is = %.f V \n",
    Vo_min)
12 Io=(Vref/R1)
    *(1/2+1/4+1/8+1/16+1/32+1/64+1/128+1/256);
13 Vo_max=Io*Rf; // Maximum output voltage
14 printf("\n Maximum output voltage is = %.2f V \n",
    Vo_max)

```

Scilab code Exa 6.14 Change in output voltage

```

1 // Chapter6
2 // Page.No-228, Figure.No-6.21
3 // Example_6_14
4 // Change in output voltage
5 // Given
6 clear;clc;
7 Rf=3*10^3;
8 Vdc=5;
9 Rt=100*10^3; // Resistance at darkness
10 Vomn=-(Vdc/Rt)*Rf; // Min output voltage at
    darkness
11 printf("\n Min output voltage at darkness is = %.2f
    V \n",Vomn)
12 Rt=1.5*10^3; // Resistance at Illumination
13 Vomax=-(Vdc/Rt)*Rf; // Max output voltage at
    Illumination
14 printf("\n Max output voltage at Illumination is = %
    .f V \n",Vomax)

```

Scilab code Exa 6.15 Output voltage of an integrator

```

1 // Chapter6
2 // Page.No-230, Figure.No-6.23
3 // Example_6_15
4 // Output voltage of an integrator
5 // Given
6 clear;clc;
7 Vin=2; // Input voltage in volt
8 Vo0=0;
9 Vo1=-integrate('2','t',0,1);
10 disp(Vo1)
11 Vo2=-integrate('2','t',1,2)+Vo1;
12 disp(Vo2)

```

```

13 Vo3=-integrate('2','t',2,3)+Vo2;
14 disp(Vo3)
15 Vo4=-integrate('2','t',3,4)+Vo3;
16 disp(Vo4)
17 Vo=[Vo0 Vo1 Vo2 Vo3 Vo4];
18 t=[0 1 2 3 4];
19 plot(t,Vo);
20 title('Output Voltage');
21 xlabel('t');
22 ylabel('Vo');

```

Scilab code Exa 6.16.a Design of differentiator

```

1 // Chapter6
2 // Page.No-238
3 // Example_6_16_a
4 // Design of differentiator
5 // Given
6 clear;clc;
7 C1=0.1*10^-6; // Assume
8 fa=1*10^3; // Freq at which gain is 0 dB
9 Rf=1/(2*%pi*fa*C1); // Using fa=1/(2*%pi*Rf*C1)
10 printf("\n Feedback resistance is = %.1f ohm \n",Rf)
11 Rf=1.5*10^3; // Approximation
12 fb=20*10^3; // Gain limiting freq
13 R1=1/(2*%pi*fb*C1);
14 printf("\n Resistance ,R1 is = %.1f ohm \n",R1)
15 R1=82; // Approximation
16 Cf=R1*C1/Rf;
17 printf("\n Capacitance ,Cf is = %.10f farad \n",Cf)
18 Cf=0.005*10^-6; // Approximation

```

Scilab code Exa 6.16.b Output waveform of differentiator

```
1 // Chapter6
2 // Page.No-238
3 // Example_6_16_b
4 // Output waveform of differentiator
5 // Given
6 clear;clc;
7 step=0.01;
8 t=0:step:2*%pi;
9 dy=diff(sin(t))/step; //approximate differentiation
    of sine function
10 Vo=-1.5*10^3*0.1*10^-6*2*%pi*10^3*dy;
11 plot(Vo);
12 title('Output Voltage');
13 xlabel('t');
14 ylabel('Vo');
```

Chapter 7

Active Filters and Oscillators

Scilab code Exa 7.1 Design of low pass filter

```
1 // Chapter7
2 // Page.No-256
3 // Example_7_1
4 // Design of low pass filter
5 // Given
6 clear;clc;
7 fh=1*10^3; // Cut-off frequency
8 C=0.01*10^-6; // Assumption
9 R=1/(2*pi*fh*C);
10 printf("\n Resistance R is = %.1f ohm \n",R) //
    Result
11 printf("\n Use 20 kohm POT as R \n")
12 R1=10*10^3; // Assumption
13 printf("\n Resistance R1 is = %.1f ohm \n",R1)
14 Rf=R1; // Sice passband gain is 2,R1 and Rf must be
    equal
15 printf("\n Resistance Rf is = %.1f ohm \n",Rf)
```

This code can be downloaded from the website www.scilab.in

Scilab code Exa 7.2 Design of low pass filter

```
1 // Chapter7
2 // Page.No-256
3 // Example_7_2
4 // Design of low pass filter
5 // Given
6 clear;clc;
7 fc0=1*10^3; // Original cut-off frequency
8 fc1=1.6*10^3; // New cut-off frequency
9 R=15.9*10^3; // Original resistance value
10 k=fc0/fc1;
11 Rnew=R*k;
12 printf("\n New Resistance Rnew is = %.1f ohm \n",
    Rnew) // Result
```

Scilab code Exa 7.3 Frequency response of low pass filter

```
1 // Chapter7
2 // Page.No-257
3 // Example_7_3
4 // Frequency response of low pass filter
5 // Given
6 clear;clc;
7 Af=2; // Passband gain of the filter
```

```

8 fh=1000; // Cut-off frequency
9 f1=10; // Input freq in Hz
10 av1=Af/sqrt(1+(f1/fh)^2);
11 printf("\n Gain magnitude av1 at f1 is = %.2f \n",
    av1) // Result
12 f2=100; // Freq in Hz
13 av2=Af/sqrt(1+(f2/fh)^2);
14 printf("\n Gain magnitude av2 at f2 is = %.2f \n",
    av2) // Result
15 f3=200; // Freq in Hz
16 av3=Af/sqrt(1+(f3/fh)^2);
17 printf("\n Gain magnitude av3 at f3 is = %.2f \n",
    av3) // Result
18 f4=700; // Freq in Hz
19 av4=Af/sqrt(1+(f4/fh)^2);
20 printf("\n Gain magnitude av4 at f4 is = %.2f \n",
    av4) // Result
21 f5=1000; // Freq in Hz
22 av5=Af/sqrt(1+(f5/fh)^2);
23 printf("\n Gain magnitude av5 at f5 is = %.2f \n",
    av5) // Result
24 f6=3000; // Freq in Hz
25 av6=Af/sqrt(1+(f6/fh)^2);
26 printf("\n Gain magnitude av6 at f6 is = %.2f \n",
    av6) // Result
27 f7=7000; // Freq in Hz
28 av7=Af/sqrt(1+(f7/fh)^2);
29 printf("\n Gain magnitude av7 at f7 is = %.2f \n",
    av7) // Result
30 f8=10000; // Freq in Hz
31 av8=Af/sqrt(1+(f8/fh)^2);
32 printf("\n Gain magnitude av8 at f8 is = %.2f \n",
    av8) // Result
33 f9=30000; // Freq in Hz
34 av9=Af/sqrt(1+(f9/fh)^2);
35 printf("\n Gain magnitude av9 at f9 is = %.2f \n",
    av9) // Result
36 f10=100000; // Freq in Hz

```



```

37 av10=Af/sqrt(1+(f10/fh)^2);
38 printf("\n Gain magnitude av10 at f10 is = %.2f \n",
    av10) // Result
39 x=[f1 f2 f3 f4 f5 f6 f7 f8 f9 f10];
40 y=[av1 av2 av3 av4 av5 av6 av7 av8 av9 av10];
41 gainplot(x,y);
42 title('Frequency Response');
43 xlabel('Frequency (Hz)');
44 ylabel('Voltage gain(dB)');

```

Scilab code Exa 7.4.a Design of second order low pass filter

```

1 // Chapter7
2 // Page.No-256
3 // Example_7_4_a
4 // Design of second order low pass filter
5 // Given
6 clear;clc;
7 fh=1*10^3; // Cut-off frequency
8 C2=0.0047*10^-6; // Assumption
9 C3=C2;
10 R2=1/(2*pi*fh*C2);
11 printf("\n Resistance R2 is = %.1f ohm \n",R2) //
    Result
12 R2=33*10^3; // Approximation
13 R3=R2;
14 printf("\n Resistance R3 is = %.1f ohm \n",R3) //
    Result
15 R1=27*10^3; // Assumption
16 Rf=0.586*R1;
17 printf("\n Resistance Rf is = %.1f ohm \n",Rf) //
    Result
18 printf("\n Use 20 kohm POT as Rf \n")

```

Scilab code Exa 7.4.b Frequency response of second order highpass filter

```
1 // Chapter7
2 // Page.No-260
3 // Example_7_4_b
4 // Frequency response of second order highpass
  filter
5 // Given
6 clear;clc;
7 Af=1.586; // Passband gain of the filter
8 fh=1000; // Cut-off frequency
9 f1=10; // Input freq in Hz
10 av1=Af/sqrt(1+(f1/fh)^4);
11 printf("\n Gain magnitude av1 at f1 is = %.2f \n",
  av1) // Result
12 f2=100; // Freq in Hz
13 av2=Af/sqrt(1+(f2/fh)^4);
14 printf("\n Gain magnitude av2 at f2 is = %.2f \n",
  av2) // Result
15 f3=200; // Freq in Hz
16 av3=Af/sqrt(1+(f3/fh)^4);
17 printf("\n Gain magnitude av3 at f3 is = %.2f \n",
  av3) // Result
18 f4=700; // Freq in Hz
19 av4=Af/sqrt(1+(f4/fh)^4);
20 printf("\n Gain magnitude av4 at f4 is = %.2f \n",
  av4) // Result
21 f5=1000; // Freq in Hz
22 av5=Af/sqrt(1+(f5/fh)^4);
23 printf("\n Gain magnitude av5 at f5 is = %.2f \n",
  av5) // Result
24 f6=3000; // Freq in Hz
```

```

25 av6=Af/sqrt(1+(f6/fh)^4);
26 printf("\n Gain magnitude av6 at f6 is = %.2f \n",
    av6) // Result
27 f7=7000; // Freq in Hz
28 av7=Af/sqrt(1+(f7/fh)^4);
29 printf("\n Gain magnitude av7 at f7 is = %.2f \n",
    av7) // Result
30 f8=10000; // Freq in Hz
31 av8=Af/sqrt(1+(f8/fh)^4);
32 printf("\n Gain magnitude av8 at f8 is = %.2f \n",
    av8) // Result
33 f9=30000; // Freq in Hz
34 av9=Af/sqrt(1+(f9/fh)^4);
35 printf("\n Gain magnitude av9 at f9 is = %.5f \n",
    av9) // Result
36 f10=100000; // Freq in Hz
37 av10=Af/sqrt(1+(f10/fh)^4);
38 printf("\n Gain magnitude av10 at f10 is = %.6f \n",
    av10) // Result
39 x=[f1 f2 f3 f4 f5 f6 f7 f8 f9 f10];
40 y=[av1 av2 av3 av4 av5 av6 av7 av8 av9 av10];
41 gainplot(x,y);
42 title('Frequency Response');
43 xlabel('Frequency(Hz)');
44 ylabel('Voltage gain(dB)');

```

Scilab code Exa 7.5.a Design of highpass filter

```

1 // Chapter7
2 // Page.No-263
3 // Example_7_5_a
4 // Design of highpass filter
5 // Given

```

```

6 clear;clc;
7 fh=1*10^3; // Cut-off frequency
8 Af=2; // Passband gain of the filter
9 C=0.01*10^-6; // Assumption
10 R=1/(2*pi*fh*C);
11 printf("\n Resistance R is = %.1f ohm \n",R) //
    Result
12 printf("\n Use 20 kohm POT as R \n")
13 R1=10*10^3; // Assumption
14 printf("\n Resistance R1 is = %.1f ohm \n",R1)
15 Rf=R1; // Sice passband gain is 2,R1 and Rf must be
    equal
16 printf("\n Resistance Rf is = %.1f ohm \n",Rf)

```

Scilab code Exa 7.5.b Frequency response of highpass filter

```

1 // Chapter7
2 // Page.No-263
3 // Example_7_5_b
4 // Frequency response of highpass filter
5 // Given
6 clear;clc;
7 Af=2; // Passband gain of the filter
8 f1=1000; // Cut-off frequency
9 f1=100; // Input freq in Hz
10 av1=(Af*(f1/f1))/sqrt(1+(f1/f1)^2);
11 printf("\n Gain magnitude av1 at f1 is = %.2f \n",
    av1) // Result
12 f2=200; // Freq in Hz
13 av2=(Af*(f2/f1))/sqrt(1+(f2/f1)^2);
14 printf("\n Gain magnitude av2 at f2 is = %.2f \n",
    av2) // Result
15 f3=400; // Freq in Hz

```

```

16 av3=(Af*(f3/f1))/sqrt(1+(f3/f1)^2);
17 printf("\n Gain magnitude av3 at f3 is = %.2f \n",
    av3) // Result
18 f4=700; // Freq in Hz
19 av4=(Af*(f4/f1))/sqrt(1+(f4/f1)^2);
20 printf("\n Gain magnitude av4 at f4 is = %.2f \n",
    av4) // Result
21 f5=1000; // Freq in Hz
22 av5=(Af*(f5/f1))/sqrt(1+(f5/f1)^2);
23 printf("\n Gain magnitude av5 at f5 is = %.2f \n",
    av5) // Result
24 f6=3000; // Freq in Hz
25 av6=(Af*(f6/f1))/sqrt(1+(f6/f1)^2);
26 printf("\n Gain magnitude av6 at f6 is = %.2f \n",
    av6) // Result
27 f7=7000; // Freq in Hz
28 av7=(Af*(f7/f1))/sqrt(1+(f7/f1)^2);
29 printf("\n Gain magnitude av7 at f7 is = %.2f \n",
    av7) // Result
30 f8=10000; // Freq in Hz
31 av8=(Af*(f8/f1))/sqrt(1+(f8/f1)^2);
32 printf("\n Gain magnitude av8 at f8 is = %.2f \n",
    av8) // Result
33 f9=30000; // Freq in Hz
34 av9=(Af*(f9/f1))/sqrt(1+(f9/f1)^2);
35 printf("\n Gain magnitude av9 at f9 is = %.2f \n",
    av9) // Result
36 f10=100000; // Freq in Hz
37 av10=(Af*(f10/f1))/sqrt(1+(f10/f1)^2);
38 printf("\n Gain magnitude av10 at f10 is = %.2f \n",
    av10) // Result
39 x=[f1 f2 f3 f4 f5 f6 f7 f8 f9 f10];
40 y=[av1 av2 av3 av4 av5 av6 av7 av8 av9 av10];
41 gainplot(x,y);
42 title('Frequency Response');
43 xlabel('Frequency (Hz)');
44 ylabel('Voltage gain (dB)');

```

Scilab code Exa 7.6.a Determination of low cutoff frequency

```
1 // Chapter7
2 // Page.No-264
3 // Example_7_6
4 // Determination of low cutoff frequency
5 // Given
6 clear;clc;
7 R2=33*10^3;
8 R3=R2;
9 C2=0.0047*10^-6;
10 C3=C2;
11 f1=1/(2*pi*sqrt(R2*R3*C2*C3));
12 printf("\n Low cutoff freq f1 is = %.1f Hz \n",f1)
    // Result
```

Scilab code Exa 7.6.b Frequency response of second order highpass filter

```
1 // Chapter7
2 // Page.No-264
3 // Example_7_6_b
4 // Frequency response of second order highpass
  filter
5 // Given
6 clear;clc;
7 Af=1.586; // Passband gain of the filter
8 f1=1000; // Cut-off frequency
9 f1=100; // Input freq in Hz
```

```

10 av1=Af/sqrt(1+(f1/f1)^4);
11 printf("\n Gain magnitude av1 at f1 is = %.5f \n",
    av1) // Result
12 f2=200; // Freq in Hz
13 av2=Af/sqrt(1+(f1/f2)^4);
14 printf("\n Gain magnitude av2 at f2 is = %.4f \n",
    av2) // Result
15 f3=700; // Freq in Hz
16 av3=Af/sqrt(1+(f1/f3)^4);
17 printf("\n Gain magnitude av3 at f3 is = %.4f \n",
    av3) // Result
18 f4=1000; // Freq in Hz
19 av4=Af/sqrt(1+(f1/f4)^4);
20 printf("\n Gain magnitude av4 at f4 is = %.4f \n",
    av4) // Result
21 f5=3000; // Freq in Hz
22 av5=Af/sqrt(1+(f1/f5)^4);
23 printf("\n Gain magnitude av5 at f5 is = %.4f \n",
    av5) // Result
24 f6=7000; // Freq in Hz
25 av6=Af/sqrt(1+(f1/f6)^4);
26 printf("\n Gain magnitude av6 at f6 is = %.4f \n",
    av6) // Result
27 f7=10000; // Freq in Hz
28 av7=Af/sqrt(1+(f1/f7)^4);
29 printf("\n Gain magnitude av7 at f7 is = %.4f \n",
    av7) // Result
30 f8=30000; // Freq in Hz
31 av8=Af/sqrt(1+(f1/f8)^4);
32 printf("\n Gain magnitude av8 at f8 is = %.4f \n",
    av8) // Result
33 f9=100000; // Freq in Hz
34 av9=Af/sqrt(1+(f1/f9)^4);
35 printf("\n Gain magnitude av9 at f9 is = %.4f \n",
    av9) // Result
36 x=[f1 f2 f3 f4 f5 f6 f7 f8 f9];
37 y=[av1 av2 av3 av4 av5 av6 av7 av8 av9];
38 gainplot(x,y);

```

```
39 title('Frequency Response');
40 xlabel('Frequency (Hz)');
41 ylabel('Voltage gain (dB)');
```

Scilab code Exa 7.7.a Design of wide bandpass filter

```
1 // Chapter7
2 // Page.No-270
3 // Example_7_7_a
4 // Design of wide bandpass filter
5 // Given
6 clear;clc;
7 fl=200; // Low cutoff freq in Hz
8 fh=1*10^3; // High cutoff freq in Hz
9 C1=0.01*10^-6; // Assumption
10 R1=1/(2*pi*fh*C1);
11 printf("\n Resistance R1 is = %.1f ohm \n", R1) //
    Result
12 C=0.05*10^-6;
13 R=1/(2*pi*fl*C);
14 printf("\n Resistance R is = %.1f ohm \n", R) //
    Result
15 printf("\n Bandpass Gain Af is = 4 \n") // Since
    gain of high pass and lowpass is set to 2
16 R1=10*10^3; // Assumption
17 printf("\n Resistance R1 is = %.1f ohm \n",R1)
18 Rf=R1; // Since passband gain is 2,R1 and Rf must be
    equal
19 printf("\n Resistance Rf is = %.1f ohm \n",Rf)
```

Scilab code Exa 7.7.b Frequency response of bandpass filter

```
1 // Chapter7
2 // Page.No-270
3 // Example_7_7_b
4 // Frequency response of bandpass filter
5 // Given
6 clear;clc;
7 Aft=4; // Passband gain of the filter
8 fl=200; // Lower Cut-off frequency
9 fh=1000; // Higher Cut-off frequency
10 f1=10; // Input freq in Hz
11 av1=(Aft*(f1/fl))/sqrt((1+(f1/fl)^2)*(1+(f1/fh)^2));
12 printf("\n Gain magnitude av1 at f1 is = %.4f \n",
    av1) // Result
13 f2=30; // Freq in Hz
14 av2=(Aft*(f2/fl))/sqrt((1+(f2/fl)^2)*(1+(f2/fh)^2));
15 printf("\n Gain magnitude av2 at f2 is = %.4f \n",
    av2) // Result
16 f3=100; // Freq in Hz
17 av3=(Aft*(f3/fl))/sqrt((1+(f3/fl)^2)*(1+(f3/fh)^2));
18 printf("\n Gain magnitude av3 at f3 is = %.4f \n",
    av3) // Result
19 f4=200; // Freq in Hz
20 av4=(Aft*(f4/fl))/sqrt((1+(f4/fl)^2)*(1+(f4/fh)^2));
21 printf("\n Gain magnitude av4 at f4 is = %.4f \n",
    av4) // Result
22 f5=447.2; // Freq in Hz
23 av5=(Aft*(f5/fl))/sqrt((1+(f5/fl)^2)*(1+(f5/fh)^2));
24 printf("\n Gain magnitude av5 at f5 is = %.4f \n",
    av5) // Result
25 f6=700; // Freq in Hz
```

```

26 av6=(Aft*(f6/f1))/sqrt((1+(f6/f1)^2)*(1+(f6/fh)^2));
27 printf("\n Gain magnitude av6 at f6 is = %.4f \n",
    av6) // Result
28 f7=1000; // Freq in Hz
29 av7=(Aft*(f7/f1))/sqrt((1+(f7/f1)^2)*(1+(f7/fh)^2));
30 printf("\n Gain magnitude av7 at f7 is = %.4f \n",
    av7) // Result
31 f8=2000; // Freq in Hz
32 av8=(Aft*(f8/f1))/sqrt((1+(f8/f1)^2)*(1+(f8/fh)^2));
33 printf("\n Gain magnitude av8 at f8 is = %.4f \n",
    av8) // Result
34 f9=7000; // Freq in Hz
35 av9=(Aft*(f9/f1))/sqrt((1+(f9/f1)^2)*(1+(f9/fh)^2));
36 printf("\n Gain magnitude av9 at f9 is = %.4f \n",
    av9) // Result
37 f10=10000; // Freq in Hz
38 av10=(Aft*(f10/f1))/sqrt((1+(f10/f1)^2)*(1+(f10/fh)
    ^2));
39 printf("\n Gain magnitude av10 at f10 is = %.4f \n",
    av10) // Result
40 x=[f1 f2 f3 f4 f5 f6 f7 f8 f9 f10];
41 y=[av1 av2 av3 av4 av5 av6 av7 av8 av9 av10];
42 gainplot(x,y);
43 title('Frequency Response');
44 xlabel('Frequency (Hz)');
45 ylabel('Voltage gain(dB)');

```

Scilab code Exa 7.7.c Calculation of quality factor

```

1 // Chapter7
2 // Page.No-270
3 // Example_7_7_c
4 // Calculation of quality factor

```

```

5 // Given
6 clear;clc;
7 fh=1*10^3; // Higher cut-off frequency
8 fl=200; // Lower cut-off frequency
9 fc=sqrt(fl*fh); // Center frequency
10 printf("\n Center frequency fc is = %.1f Hz \n",fc)
    // Result
11 Q=fc/(fh-fl); // Quality factor
12 printf("\n Quality factor Q is = %.2f \n",Q) //
    Result

```

Scilab code Exa 7.8.a Design of narrow bandpass filter

```

1 // Chapter7
2 // Page.No-272
3 // Example_7_8_a
4 // Design of narrow bandpass filter
5 // Given
6 clear;clc;
7 fc=1*10^3; // Center frequency
8 Q=3; // Quality factor
9 Af=10; // Passband gain
10 C1=0.01*10^-6; // Assumption
11 C2=C1;
12 R1=Q/(2*%pi*fc*C1*Af);
13 R2=Q/(2*%pi*fc*C1*(2*Q^2-Af));
14 R3=Q/(%pi*fc*C1);
15 printf("\n Resistance R1 is = %.1f ohm \n", R1) //
    Result
16 printf("\n Resistance R2 is = %.1f ohm \n", R2) //
    Result
17 printf("\n Resistance R3 is = %.1f ohm \n", R3) //
    Result

```

Scilab code Exa 7.8.b Design of narrow bandpass filter

```
1 // Chapter7
2 // Page.No-272
3 // Example_7_8_a
4 // Design of narrow bandpass filter
5 // Given
6 clear;clc;
7 fc0=1*10^3; // Original center frequency
8 fc1=1.5*10^3; // New center frequency
9 R2=5.97*10^3; // Original resistance
10 R2new=R2*(fc0/fc1)^2;
11 printf("\n Resistance R1 is = %.1f ohm \n", R2new)
    // Result
```

Scilab code Exa 7.9 Design of wide bandreject filter

```
1 // Chapter7
2 // Page.No-274
3 // Example_7_9
4 // Design of wide bandreject filter
5 // Given
6 clear;clc;
7 fh=200; // Low cutoff freq in Hz
8 fl=1*10^3; // High cutoff freq in Hz
9 C2=0.01*10^-6; // Assumption
10 R2=1/(2*pi*fl*C2);
```

```

11 printf("\n Resistance R2 of highpass section is = %
    .1f ohm \n", R2) // Result
12 C=0.05*10^-6;
13 R=1/(2*pi*fh*C);
14 printf("\n Resistance R of lowpass section is = %.1f
    ohm \n", R) // Result
15 printf("\n Bandpass Gain Af is = 4 \n") // Since
    gain of high pass and lowpass is set to 2
16 R1=10*10^3; // Assumption
17 printf("\n Resistance R1 is = %.1f ohm \n",R1)
18 Rf=R1; // Since passband gain is 2,R1 and Rf must be
    equal
19 printf("\n Resistance Rf is = %.1f ohm \n",Rf)

```

Scilab code Exa 7.10 Design of notch filter

```

1 // Chapter7
2 // Page.No-277
3 // Example_7_10
4 // Design of notch filter
5 // Given
6 clear;clc;
7 fn=60; // Notch-out frequency in Hz
8 C=0.068*10^-6; // Assumption
9 R=1/(2*pi*fn*C);
10 printf("\n Resistance R is = %.1f ohm \n",R) //
    Result

```

Scilab code Exa 7.11 Phase angle

```

1 // Chapter7
2 // Page.No-279
3 // Example_7_11
4 // Phase angle
5 // Given
6 clear;clc;
7 f=1*10^3; // Input frequency in Hz
8 C=0.01*10^-6;
9 R=15.9*10^3;
10 phi=-2*atand(2*pi*f*C*R); // Phase angle
11 printf("\n Phase angle phi is = %.f deg \n",phi) //
    Result

```

Scilab code Exa 7.12 Design of phase shift oscillator

```

1 // Chapter7
2 // Page.No-282
3 // Example_7_1
4 // Design of phase shift oscillator
5 // Given
6 clear;clc;
7 fo=200; // Frequency of oscillation
8 C=0.1*10^-6; // Assumption
9 R=0.065/(fo*C);
10 printf("\n Resistance R is = %.1f ohm \n",R) //
    Result
11 printf("\n Use Resistance R as 3.3 kohm \n")
12 R=3.3*10^3;
13 R1=10*R; // To prevent loading of amplifier
14 Rf=29*R1;
15 printf("\n Resistance Rf is = %.1f ohm \n",Rf) //
    Result

```

Scilab code Exa 7.13 Design of wein bridge oscillator

```
1 // Chapter7
2 // Page.No-282
3 // Example_7_13
4 // Design of wein bridge oscillator
5 // Given
6 clear;clc;
7 fo=965; // Frequency of oscillation
8 C=0.05*10^-6; // Assumption
9 R=0.159/(fo*C);
10 printf("\n Resistance R is = %.1f ohm \n",R) //
    Result
11 R1=12*10^3; // Assumption
12 Rf=2*R1;
13 printf("\n Resistance Rf is = %.1f ohm \n",Rf) //
    Result
```

Scilab code Exa 7.14 Design of quadrature oscillator

```
1 // Chapter7
2 // Page.No-285
3 // Example_7_14
4 // Design of quadrature oscillator
5 // Given
6 clear;clc;
7 fo=159; // Frequency of oscillation
8 C=0.01*10^-6; // Assumption
```

```

9 R=0.159/(fo*C);
10 printf("\n Resistance values R1,R2,R3 is = %.1f ohm
    \n",R) // R1=R2=R3=R
11 printf("\n Capacitance values C1,C2,C3 is = %.8f
    farad \n",C) // C1=C2=C3=C

```

Scilab code Exa 7.15 Design of squarewave oscillator

```

1 // Chapter7
2 // Page.No-287
3 // Example_7_15
4 // Design of squarewave oscillator
5 // Given
6 clear;clc;
7 fo=1*10^3;; // Frequency of oscillation
8 R1=10*10^3; // Assumption
9 C=0.05*10^-6; // Assumption
10 R2=1.16*R1;
11 printf("\n Resistance R2 is = %.1f ohm \n",R2) //
    Result
12 R=1/(2*fo*C);
13 printf("\n Resistance R is = %.1f ohm \n",R) //
    Result

```

Scilab code Exa 7.16 Design of triangular wave generator

```

1 // Chapter7
2 // Page.No-291
3 // Example_7_16

```



```

4 // Design of triangular wave generator
5 // Given
6 clear;clc;
7 fo=2*10^3; // Frequency of oscillation
8 vo=7; // Output voltage
9 Vsat=14; // Saturation voltage for opamp 1458
10 R3=40*10^3; // Assumption
11 R2=(vo*R3)/(2*Vsat);
12 printf("\n Resistance R2 is = %.1f ohm \n",R2) //
    Result
13 k=R3/(4*fo*R2); // Using fo=R3/(4*R1*C1*R2),k=R1*C1;
14 C1=0.05*10^-6; // Assumption
15 R1=k/C1;
16 printf("\n Resistance R1 is = %.1f ohm \n",R1) //
    Result

```

Scilab code Exa 7.17.a Nominal frequency

```

1 // Chapter7
2 // Page.No-296
3 // Example_7_17_a
4 // Nominal frequency
5 // Given
6 clear;clc;
7 R2=1.5*10^3;
8 R1=10*10^3;
9 R3=10*10^3;
10 C1=0.001*10^-6;
11 V=12; // Supply voltage
12 Vc=R3*V/(R2+R3); // Using voltage divider rule
13 printf("\n Terminal voltage Vc is = %.2f V \n",Vc)
    // Result
14 fo=2*(V-Vc)/(V*R1*C1);

```

```
15 printf("\n Approximate Nominal freq fo is = %.1f Hz
    \n",fo) // Result
```

Scilab code Exa 7.17.b Modulation in output frequency

```
1 // Chapter7
2 // Page.No-296
3 // Example_7_17_b
4 // Modulation in output frequency
5 // Given
6 clear;clc;
7 R2=1.5*10^3;
8 R1=10*10^3;
9 R3=10*10^3;
10 C1=0.001*10^-6;
11 V=12; // Supply voltage
12 Vc1=9.5;
13 Vc2=11.5;
14 fo1=2*(V-Vc1)/(V*R1*C1);
15 printf("\n Approximate Nominal freq fo1 is = %.1f Hz
    \n",fo1) // Result
16 fo2=2*(V-Vc2)/(V*R1*C1);
17 printf("\n Approximate Nominal freq fo2 is = %.1f Hz
    \n",fo2) // Result
18 delta_fo=fo1-fo2; // Change in output freq
19 printf("\n Change in output freq delta_fo is = %.1f
    Hz \n",delta_fo) // Result
```

Chapter 8

Comparators and Converters

Scilab code Exa 8.1 Threshold voltage

```
1 // Chapter8
2 // Page.No-320, Figure.No-8.4(a)
3 // Example_8_1
4 // Threshold voltage
5 // Given
6 clear;clc;
7 R1=100;R2=56*10^3;
8 vin=1; // Input voltage in volt
9 pos_Vsat=14; // Positive saturation voltage in volt
10 neg_Vsat=-14; // Negative saturation voltage in volt
11 Vut=(R1/(R1+R2))*(pos_Vsat); // Upper threshold
    voltage
12 printf("\n Upper threshold voltage is = %.4f V \n",
    Vut) // Result
13 Vlt=(R1/(R1+R2))*(neg_Vsat); // Lower threshold
    voltage
14 printf("\n Lower threshold voltage is = %.4f V \n",
    Vlt) // Result
15 t=0:0.1:2*pi;
16 vut=0.5*sin(t);
17 subplot(2,1,1);
```

```

18 plot(t,vut);
19 title('Input Voltage');
20 xlabel('t');
21 ylabel('Vin');
22 c=0;
23 for i=0:0.1:2*pi
24     c=c+1;
25 end
26 for i=1:c;
27     if vut(i)>0.025
28         v(i)=-14;
29     else if vut(i)<-0.025
30         v(i)=14;
31     end
32 end
33 end
34 subplot(2,1,2);
35 plot(t,v);
36 title('Output Waveform');
37 xlabel('t');
38 ylabel('Vo');

```

Scilab code Exa 8.2 Output voltage swing

```

1 // Chapter8
2 // Page.No-326, Figure.No-8.7(a)
3 // Example_8_2
4 // Output voltage swing
5 // Given
6 clear;clc;
7 vin=5*10^-3;
8 R=100;
9 Vd1=-0.7; // Output voltage during positive half-

```

```

        cycle of the input
10 Vd2=5.1; // Output voltage during negative half-
        cycle of the input
11 printf("\n Output voltage during positive half-cycle
        of the input is = %.1f V \n",Vd1) // Since zener
        diode is forward biased
12 printf("\n Output voltage during negative half-cycle
        of the input is = %.1f V \n",Vd2) // Since zener
        diode is reverse-biased
13 t=0:0.1:2*pi;
14 vut=0.5*sin(t);
15 subplot(2,1,1);
16 plot(t,vut);
17 title('Input Voltage');
18 xlabel('t');
19 ylabel('Vin');
20 c=1;
21 for t=0:0.1:2*pi
22 if t<%pi
23     v(c)=-0.7;
24     else
25     v(c)=5.1;
26 end
27 c=c+1;
28 subplot(2,1,2);
29 plot(v);
30 end
31 title('Output Waveform');
32 xlabel('t');
33 ylabel('Vo');

```

Scilab code Exa 8.3 Output frequencies

```

1 // Chapter8
2 // Page.No-320, Figure.No-8.12
3 // Example_8_3
4 // Output frequencies
5 // Given
6 clear;clc;
7 Vin=2; // Input voltage
8 Fo1=2*10^3; // Output freq Fo when Vin=2V
9 Fo2=1*10^3; // Output freq Fo/2 when Vin=2V
10 printf("\n Output freq Fo is = %.f Hz \n",Fo1) //
    Result
11 printf("\n Output freq Fo/2 is = %.f Hz \n",Fo2) //
    Result
12 count=1;
13 for i=1:50; //for 5 cycles
14 if count<4;
15     v(i)=5;
16 else
17     v(i)=0;
18 end
19 if count<10
20     count=count+1;
21 else
22     count=1;
23 end
24 end
25 subplot(2,1,1);
26 plot(v);
27 title('Output Waveform');
28 xlabel('t(microsec)');
29 ylabel('Pulse freq output ,Fo(V)');
30 for i=1:50; //for 5 cycles
31 if count<10;
32     v(i)=5;
33 else
34     v(i)=0;
35 end
36 if count<20

```

```

37     count=count+1;
38 else
39     count=1;
40 end
41 end
42 subplot(2,1,2);
43 plot(v);
44 title('Output Waveform');
45 xlabel('t(microsec)');
46 ylabel('Pulse freq output ,Fo/2(V)');

```

Scilab code Exa 8.4 Output voltage

```

1 // Chapter8
2 // Page.No-335, Figure.No-8.14(a)
3 // Example_8_4
4 // Output voltage
5 // Given
6 clear;clc;
7 Vo=2.8; // At Finmax of 10kHz
8 Vo1=Vo/10; // Output voltage at Fin=1kHz
9 printf("\n Output voltage is = %.2f V \n",Vo1) //
   Result

```

Scilab code Exa 8.5 Output voltage

```

1 // Chapter8
2 // Page.No-335, Figure.No-8.25(a)
3 // Example_8_5

```

```
4 // Output voltage
5 // Given
6 clear;clc;
7 vin=100*10^-3;
8 t=0:0.1:2*%pi;
9 i=1;
10 for t=0:0.1:2*%pi;
11     if t<=%pi
12         v(i)=vin*sin(t);
13     else
14         v(i)=0;
15     end
16     i=i+1;
17 end
18 t=0:0.1:2*%pi;
19 plot(t,v)
```

Chapter 9

Specialized IC Applications

Scilab code Exa 9.1 Second order inverting butterworth lowpass filter

```
1 // Chapter9
2 // Page.No-387
3 // Example_9_1
4 // Second order inverting butterworth lowpass filter
5 // Given
6 clear;clc;
7 dc_gain=5;
8 f1=2*10^3; // Cutoff freq in Hz
9 Q=10; // Figure of merit
10 R2=(316*10^3)/10; // Resistance R2
11 printf("\n Resistance R2 is = %.1f ohm \n",R2) //
    Result
12 R3=(100*10^3)/((3.16*Q)-1);
13 printf("\n Resistance R3 is = %.1f ohm \n",R3) //
    Result
14 printf("\n Resistance R1 is Open \n") // Result
15 R4=(5.03*10^7)/f1;
16 printf("\n Resistance R4 is = %.1f ohm \n",R4) //
    Result
17 R5=R4;
18 printf("\n Resistance R5 is = %.1f ohm \n",R5) //
```

```

    Result
19 R6=1.8*10^3; // Assumption
20 R7=dc_gain*R6;
21 printf("\n Resistance R7 is = %.1f ohm \n",R7) //
    Result and its a potentiometer
22 R8=(R6*R7)/(R6+R7);
23 printf("\n Resistance R8 is = %.3f ohm \n",R8) //
    Result

```

Scilab code Exa 9.2 Second order inverting butterworth bandpass filter

```

1 // Chapter9
2 // Page.No-388
3 // Example_9_2
4 // Second order inverting butterworth bandpass
    filter
5 // Given
6 clear;clc;
7 f1=5*10^3; // Center freq in Hz
8 Q=10; // Figure of merit
9 R2=100*10^3; // Constant for band-pass filter
10 printf("\n Resistance R2 is = %.1f ohm \n",R2) //
    Result
11 R3=(100*10^3)/((3.48*Q)-1);
12 printf("\n Resistance R3 is = %.1f ohm \n",R3) //
    Result
13 printf("\n Resistance R1 is Open \n") // Result
14 R4=(5.03*10^7)/f1;
15 printf("\n Resistance R4 is = %.1f ohm \n",R4) //
    Approximately 10kohm
16 R5=R4;
17 printf("\n Resistance R5 is = %.1f ohm \n",R5) //
    Approximately 10kohm and its a potentiometer

```

Scilab code Exa 9.3 Design of notch filter

```
1 // Chapter9
2 // Page.No-390
3 // Example_9_3
4 // Design of notch filter
5 // Given
6 clear;clc;
7 f1=5*10^3; // notch freq in Hz
8 Q=10; // Figure of merit
9 R2=100*10^3; // Constant for band-pass filter
10 printf("\n Resistance R2 is = %.1f ohm \n",R2) //
    Result
11 R3=(100*10^3)/((3.48*Q)-1);
12 printf("\n Resistance R3 is = %.1f ohm \n",R3) //
    Result
13 printf("\n Resistance R1 is Open \n") // Result
14 R4=(5.03*10^7)/f1;
15 printf("\n Resistance R4 is = %.1f ohm \n",R4) //
    Approximately 10kohm
16 R5=R4;
17 printf("\n Resistance R5 is = %.1f ohm \n",R5) //
    Approximately 10kohm and its a potentiometer
18 R6=10*10^3; // Assumption
19 printf("\n Resistance R6 is = %.1f ohm \n",R6) //
    Result
20 R7=R6;
21 printf("\n Resistance R7 is = %.1f ohm \n",R7) //
    Result
22 R8=R6;
23 printf("\n Resistance R8 is = %.1f ohm \n",R8) //
    Result
```

```

24 R9=(R6*R7*R8)/(R6*R7+R6*R8+R7*R8); // Since R6||R7||
    R8
25 printf("\n Resistance R9 is = %.1f ohm \n",R9) //
    Result

```

Scilab code Exa 9.4 Second order butterworth lowpass filter

```

1 // Chapter9
2 // Page.No-398
3 // Example_9_4
4 // Second order butterworth lowpass filter
5 // Given
6 clear;clc;
7 f1=500; // Cut-off freq in Hz
8 Holp=-2; // Passband gain
9 R1=10*10^3; // Assumption
10 R2=-R1*Holp; // Using Holp=-R2/R1;
11 printf("\n Resistance R2 is = %.1f ohm \n",R2) //
    Result
12 Q=0.707; // Figure of merit Q is fixed for second
    order butterworth low-pass filter
13 R3=Q*R2; // Using Q=R3/R2
14 printf("\n Resistance R3 is = %.1f ohm \n",R3) //
    Approximately 15kohm

```

Scilab code Exa 9.5 Value of capacitor

```

1 // Chapter9
2 // Page.No-402, Figure.No-9.16(a)

```

```

3 // Example_9_5
4 // Value of capacitor
5 // Given
6 clear;clc;
7 Ra=10*10^3; // Resistance in ohm
8 tp=10*10^-3; // Output pulse width
9 C=tp/(1.1*Ra);
10 printf("\n Capacitance C is = %.9f farad \n",C) //
    Approximately 1uF

```

Scilab code Exa 9.6 Value of resistor

```

1 // Chapter9
2 // Page.No-402, Figure.No-9.16(a)
3 // Example_9_6
4 // Value of resistor
5 // Given
6 clear;clc;
7 f=2*10^3; // Freq of input trigger signal in Hz
8 C=0.01*10^-6;
9 tp=1.2/f;
10 Ra=tp/(1.1*C);
11 printf("\n Resistance Ra is = %.1f ohm \n",Ra) //
    Result

```

Scilab code Exa 9.7 Value of t_c t_d and f_0

```

1 // Chapter9
2 // Page.No-402, Figure.No-9.21(a)

```

```

3 // Example_9_7
4 // Value of tc,td and f0
5 // Given
6 clear;clc;
7 Ra=2.2*10^3; // Resistance in ohm
8 Rb=3.9*10^3; // Resistance in ohm
9 C=0.1*10^-6; // capacitance in farad
10 tc=0.69*(Ra+Rb)*C; // Charging time of the capacitor
11 printf("\n Charging time of the capacitor is = %.6f
    sec \n",tc) // Result
12 td=0.69*Rb*C; // Discharging time of the capacitor
13 printf("\n Discharging time of the capacitor is = %
    .6f sec \n",td) // Result
14 T=tc+td;
15 fo=1/T // Freq of oscillation
16 printf("\n Freq of oscillation is = %.1f Hz \n",fo)
    // Result

```

Scilab code Exa 9.8 Freq of free running ramp generator

```

1 // Chapter9
2 // Page.No-412, Figure.No-9.24(a)
3 // Example_9_8
4 // Freq of free running ramp generator
5 // Given
6 clear;clc;
7 R=10*10^3; // Resistance in ohm
8 Vcc=5 // Supply voltage in volt
9 Vbe=0.7 // Base to emitter voltage in volt
10 C=0.05*10^-6; // Capacitance in farad
11 Ic=(Vcc-Vbe)/R; // Collector current in ampere
12 fo=(3*Ic)/(Vcc*C);
13 printf("\n Freq of free running ramp generator is =

```

```
%.1 f Hz \n",fo) // Result
```

Scilab code Exa 9.9 Value of fout fl fc

```
1 // Chapter9
2 // Page.No-423, Figure.No-9.33(a)
3 // Example_9_9
4 // Value of fout , fl , fc
5 // Given
6 clear;clc;
7 R1=12*10^3; // Resistance in ohm
8 V_plus=10 // Supply voltage in volt
9 V_minus=-10 // Supply voltage in volt
10 C1=0.01*10^-6; // Capacitance in farad
11 C2=10*10^-6; // Capacitance in farad
12 fout=1.2/(4*R1*C1);
13 printf("\n Free running frequency of VCO is = %.1 f
      Hz \n",fout) // Result
14 V=V_plus-V_minus;
15 fl=(8*fout)/V;
16 printf("\n Lock range frequency of VCO is = %.1 f Hz
      \n",fl) // Result
17 fc=sqrt(fl/(2*pi*3.6*10^3*C2));
18 printf("\n Capture range frequency of VCO is = %.2 f
      Hz \n",fc) // Result
```

Scilab code Exa 9.10 Design of current source

```
1 // Chapter9
```

```

2 // Page.No-440
3 // Example_9_10
4 // Design of current source
5 // Given
6 clear;clc;
7 Vr=5; // Voltage in volt
8 I1=0.25; // Load current in ampere
9 R1=48; // Load resistance in ohm
10 dropout_volt=2; // Constant for IC7805C
11 R=Vr/I1; // Approximate result since Iq is negligible
    in the eq. I1=(Vr/I1)+Iq where Iq is quiescent
    current
12 printf("\n Resistance R is = %.f ohm \n",R) //
    Result
13 V1=R1*I1;
14 Vo=Vr+V1;
15 printf("\n Output voltage Vo is = %.f V \n",Vo) //
    Result
16 Vin=Vo+dropout_volt;
17 printf("\n Min input voltage Vin is = %.f V \n",Vin)
    // Result

```

Scilab code Exa 9.11 Design of voltage regulator

```

1 // Chapter9
2 // Page.No-444
3 // Example_9_11
4 // Design of voltage regulator
5 // Given
6 clear;clc;
7 Vo_min=5; // Min output voltage in volt
8 Vo_max=12; // Max output voltage in volt
9 Vref=1.25; // Reference voltage in volt

```



```

10 Iadj=100*10^-6; // Adjustment pin current in ampere
11 R1=240; // Assumption
12 R2_min=R1*(Vo_min-Vref)/(Vref+Iadj*R1); // Using
    Vo_min=Vref*(1+R2/R1)+Iadj*R2
13 printf("\n Resistance R2_min is = %.1f ohm \n",
    R2_min) // Result
14 R2_max=R1*(Vo_max-Vref)/(Vref+Iadj*R1); // Using
    Vo_max=Vref*(1+R2/R1)+Iadj*R2
15 printf("\n Resistance R2_max is = %.1f ohm \n",
    R2_max) // Result
16 printf("\n Therefore resistance should be varied
    from R2_min to R2_max values.To do this we take
    R2 as 3kohm potentiometer \n")
17 C2=1*10^-6; // Added to the circuit to improve
    transient response
18 C3=1*10^-6; // Added to the circuit to obtain high
    ripple rejection ratios

```

Scilab code Exa 9.12 Design of stepdown switching regulator

```

1 // Chapter9
2 // Page.No-453
3 // Example_9_12
4 // Design of stepdown switching regulator
5 // Given
6 clear;clc;
7 Iomax=500*10^-3; // Max output current in ampere
8 Vo=5; // Output voltage in volt
9 Vd=1.25; // Voltage drop across the power diode in
    volt
10 Vin=12; // Input voltage in volt
11 Vs=1.1; // Output saturation voltage in volt
12 Vripple=50*10^-3; // Output ripple voltage in volt

```

```

13 Vref=1.245; // Reference voltage in volt
14 Vr2=1.2; // Voltage across resistance R2 in volt
15 Ipk=2*Iomax; // Sense current in ampere
16 printf("\n Sense current ,Ipk is = %.f A \n",Ipk) //
    Result
17 Rsc=0.33/Ipk; // Sense resistance in ohm
18 printf("\n Sense resistance ,Rsc is = %.2f ohm \n",
    Rsc) // Result
19 K=(Vo+Vd)/(Vin-Vs-Vo); // K= ton/toff
20 printf("\n Constant K = %.2f \n",K) // Result
21 printf("\n i.e, ton is K times of toff \n")
22 f=20*10^3; // Assuming operating freq in Hz
23 T=1/f;
24 toff=T/2.06; // Using ton+toff=T and substituting
    for ton
25 printf("\n OFF time period ,toff is = %.8f sec \n",
    toff) // Result
26 ton=1.06*toff;
27 printf("\n ON time period ,ton is = %.8f sec \n",ton)
    // Result
28 Ct=45*10^-5*toff; // Oscillator timing capacitance
    in farad
29 printf("\n Oscillator timing capacitance ,Ct is = %
    .10f F \n",Ct) // Result
30 L=((Vo+Vd)/Ipk)*toff; // Inductance in henry
31 printf("\n Inductance ,L is = %.8f H \n",L) // Result
32 Co=Ipk*((ton+toff)/(8*Vripple)); // Output
    capacitance in farad
33 printf("\n Output capacitance ,Co is = %.7f F \n",Co)
    // Result
34 I2=0.1*10^-3; // Assuming the current through R2
35 R2=Vref/I2; // Resistance R2 in ohm
36 printf("\n Resistance R2 is = %.1f ohm \n",R2) //
    Result
37 R2=12*10^3; // Taking approximate value
38 R1=(R2*(Vo-Vr2))/Vr2; // Using Vr2=(R1*Vo)/R1+R2,
    voltage divider rule
39 printf("\n Resistance R1 is = %.1f ohm \n",R1) //

```

```

    Result
40 efficiency=((Vin-Vs+Vd)/Vin)*(Vo/(Vo+Vd))*100;
41 printf("\n efficiency is = %.1f \n",efficiency) //
    Result

```

Scilab code Exa 9.13 Design of stepdown switching regulator

```

1 // Chapter9
2 // Page.No-458
3 // Example_9.13
4 // Design of stepdown switching regulator
5 // Given
6 clear;clc;
7 Iomax=3; // Max output current in ampere
8 Vo=5; // Output voltage in volt
9 Vd=1.25; // Voltage drop across the power diode in
    volt
10 Vin=12; // Input voltage in volt
11 Vs=1.1; // Output saturation voltage in volt
12 Vripple=50*10^-3; // Output ripple voltage in volt
13 Vref=1.245; // Reference voltage in volt
14 Vr2=1.2; // Voltage across resistance R2 in volt
15 Ipk=2*Iomax; // Sense current in ampere
16 printf("\n Sense current ,Ipk is = %.f A \n",Ipk) //
    Result
17 Rsc=0.33/Ipk; // Sense resistance in ohm
18 printf("\n Sense resistance ,Rsc is = %.3f ohm \n",
    Rsc) // Result
19 K=(Vo+Vd)/(Vin-Vs-Vo); // K= ton/toff
20 printf("\n Constant K = %.2f \n",K) // Result
21 printf("\n i.e, ton is K times of toff \n")
22 f=20*10^3; // Assuming operating freq in Hz
23 T=1/f;

```

```

24 toff=T/2.06; // Using ton+toff=T and substituting
    for ton
25 printf("\n OFF time period ,toff is = %.8f sec \n",
    toff) // Result
26 ton=1.06*toff;
27 printf("\n ON time period ,ton is = %.8f sec \n",ton)
    // Result
28 Ct=45*10^-5*toff; // Oscillator timing capacitance
    in farad
29 printf("\n Oscillator timing capacitance ,Ct is = %
    .10f F \n",Ct) // Result
30 L=((Vo+Vd)/Ipk)*toff; // Inductance in henry
31 printf("\n Inductance ,L is = %.8f H \n",L) // Result
32 Co=Ipk*((ton+toff)/(8*Vripple)); // Output
    capacitance in farad
33 printf("\n Output capacitance ,Co is = %.7f F \n",Co)
    // Result

```
