

Scilab Textbook Companion for
Feedback Circuits And Operational Amplifiers
by D. H. Horrocks¹

Created by
Naveen Varma Champathi
B.Tech (pursuing)
Electronics Engineering
National Institute of Technology, Nagpur
College Teacher
Eswar Rao
Cross-Checked by
Aditya Sengupta, IIT Bombay

July 31, 2019

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: Feedback Circuits And Operational Amplifiers

Author: D. H. Horrocks

Publisher: Chapman And Hall

Edition: 2

Year: 1990

ISBN: 0-412-34270-7

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

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

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

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

Contents

List of Scilab Codes	4
2 GENERAL PROPERTIES OF FEEDBACK AMPLIFIERS	5
3 AMPLIFIERS WITHOUT FEEDBACK	8
4 FEEDBACK AMPLIFIER CIRCUITS	12
5 MORE ABOUT FEEDBACK AMPLIFIERS	16
6 THE OP AMPS BASIC IDEAS AND CIRCUITS	23
7 OP AMP NON IDEALITIES	25
8 SELECTED OP AMP APPLICATIONS	28
9 FURTHER OP AMP APPLICATIONS	31

List of Scilab Codes

Exa 2.1	feedback fraction	5
Exa 2.2	Expected variations in closed loop gain . . .	6
Exa 3.1	Output voltage and gain of two stage amplifier	8
Exa 3.2	single stage amplifier parameters	9
Exa 3.3	Common mode rejection ratio and differential mode	10
Exa 4.1	input resistance of a feedback amplifier . . .	12
Exa 4.2	input resistance of a feedback amplifier . . .	12
Exa 4.3	parameters of a feedback amplifier	13
Exa 4.4	series voltage feedback circuit	14
Exa 5.1	Voltage shunt feedback circuit	16
Exa 5.2	amplifier parameters	17
Exa 5.4	closed loop current gain input output resis- tance	18
Exa 5.5	Voltage shunt feedback circuit	20
Exa 5.6	feed back fraction	22
Exa 6.1	design values of an inverting amplifier	23
Exa 6.2	gain of a negative feedback amplifier	23
Exa 6.4	non inverting amplifier	24
Exa 7.1	Output voltage	25
Exa 7.2	output offset voltage	25
Exa 7.4	inverting voltage amplifier	26
Exa 7.5	Full power bandwidth frequency	27
Exa 8.1	basic differential amplifier	28
Exa 8.2	Instrumental amplifier	29
Exa 8.3	wein bridge oscillator	29
Exa 9.1	Design of phase shifter	31
Exa 9.2	design of non inverting amplifier	32

Exa 9.3	low pass second order filter	33
---------	--	----

Chapter 2

GENERAL PROPERTIES OF FEEDBACK AMPLIFIERS

Scilab code Exa 2.1 feedback fraction

```
1 //example1:
2
3 printf(" Given:")
4 disp(" value of A=1000")//To display given values
5 A=1000
6 disp(" Magnitude of closed loop gain with feedback ,Af
   =10")
7 Af=10
8 disp(" Af=(A)/(1+A*b)")//standard formula for closed
   loop gain
9 disp(" b=((A/Af)-1)*(1/A)")
10 b=((A/Af)-1)*(1/A)
11 printf(' The value of b is %f',b)
12 //if ,A becomes 900 angle(-30 degrees)
13 A=900*cos(-%pi/6)+%i*900*sin(-%pi/6)
14 disp(" If ,A becomes 900 angle(-30 degrees)")
15 disp(" Af=(A)/((1+A*b))")//standard formula for
   closed loop gain
16 Af=(A)/((1+A*b))
```

```

17 r=real(Af)//To get real part of Af
18 i=imag(Af)//To get imaginary part of Af
19 printf(' The value of Af is (%f)+i(%f)',r,i)//to
    display value of Af
20 t=tan(i/r)//to get angle in radians of Af
21 t=t*180/%pi//to get angle in degrees
22 m=abs(Af)//to get magnitude of Af
23 printf('\n Af=%fangle(%f degrees)',m,t)
24 printf('\n Af=10.004angle(-0.32 degrees)')//rounding
    to 3 digits

```

Scilab code Exa 2.2 Expected variations in closed loop gain

```

1 //To calculate the expected variations in closed
  loop gain
2 disp(" Given:")
3 disp(" Af=10+0.2% or 10-0.2%")//To display given
  values
4 disp("-->Af=10 and dAf/Af=0.2%")
5 disp("The available amplifiers are with gains
  50,500,5000")
6 disp("dA/A=20%")
7 disp("We have from formula")
8 disp(" dAf/Af=(1/(1+A*b))*(dA/A)")//standard formula
  for closed loop gain
9 disp(" let f=0.2%=(dAf/Af), a=20%=(dA/A), k=(1+A*b)")
10 disp(" Therefore we have:")
11 Af=10;
12 printf(" Af=%d\n", Af)
13 f=0.2/100;
14 printf(" f=%f\n", f)
15 a=20/100;
16 printf(" a=%f", a)
17 disp(" Therefore, the above formula becomes f=(1/k)*a"
  )

```



```

18 k=a/f
19 printf(" Therefore (1=A*b)>=%d, but ,A=(1+A*b)*Af" ,k)
20 l=k*Af//store partial value
21 disp("A>=k*Af")//given condition
22 printf(" Therefore we got the value of A to be >=%d"
    ,l)//display value of A
23 printf("\n Therefore A=5000 satisfies the obtained
    condition. Therefore A=5000 is most economical")
24 A=5000
25 b=((A/Af)-1)*(1/A)
26 printf(" The value of b is %f",b)
27 disp(" Pratical values of A are 4000 and 6000, since
    dA/A=(+-20%) and A=5000")
28 disp(" Therefore ")
29 Afmin=(4000/(1+(4000*b)))//to print result
30 Afmax=(6000/(1+(6000*b)))//to print result
31 printf(" Afmin=%1.3 f" ,Afmin)
32 devAfmin=(1-(Afmin/Af))*100;//deviation of Afmin
    from Af
33 printf("( %1.2 f percent low)",devAfmin)//to display
    deviation from original value
34 printf("\n Afmax=%2.3 f" ,Afmax)
35 devAfmax=(1-(Afmax/Af))*100;//deviation of Afmax
    from Af
36 dev=abs(devAfmax);//negative value of devAfmax
    indicates Afmax is greater than Af by abs(
    devAfmax)
37 printf("( %1.2 f percent high) ",dev)//to display
    deviation from original value

```

Chapter 3

AMPLIFIERS WITHOUT FEEDBACK

Scilab code Exa 3.1 Output voltage and gain of two stage amplifier

```
1 //example1:
2
3 printf(" Given:")
4 disp("The value of Rin=Rout=100kohms ,Rload=1kohm ,Kv
   =100")//To display given values
5 Rin=100//assigning given values
6 Rout=100
7 Rload=1
8 Kv=100
9 disp(" All resistances are in kohms")
10 disp("The whole two stage amplifier is fed by a
   generator having a voltage of Eg=1mV when no load
   and having self resistance Rg=20kohms")
11 Eg=(1/1000)//Eg converted to volts
12 Rg=20//all resistances are in kohms
13 disp(" Since amplifiers are identical")//given
14 disp(" Rin1=Rin")
15 disp(" Rin2=Rin")
16 disp(" Rout1=Rout")
```

```

17 disp(" Rout2=Rout")
18 Rin1=Rin//assigning given values
19 Rin2=Rin
20 Rout1=Rout
21 Rout2=Rout
22 disp(" Vout2 is given by equation")
23 disp(" Vout2=((Eg)*(Rin1/(Rg+Rin1)))*(Kv)*(Rin2/(
    Rout1+Rin2))*(Kv)*(Rload/(Rout2+Rload))")
24 Vout2=((Eg)*(Rin1/(Rg+Rin1)))*(Kv)*(Rin2/(Rout1+Rin2
    ))*(Kv)*(Rload/(Rout2+Rload))//equation for Vout2
25 printf(" Therefore Vout2=%f",Vout2)
26 disp(" Vin is given by equation ")
27 disp(" Vin=(Eg*(Rin1/Rg+Rin1))")
28 Vin=(Eg*(Rin1/(Rg+Rin1))//Equation for Vin
29 disp("Now overall gain ")
30 disp(" Av=(Vout2/Vin)")
31 Av=(Vout2/Vin)//Equation for Av
32 printf(" Therefore Vout2=%f and overall voltage gain
    Av=%f",Vout2,Av)//To print the required values.

```

Scilab code Exa 3.2 single stage amplifier parameters

```

1
2 rin=10000;//in ohms
3 Kv=100;//constant for voltage-controlled voltage-
    source
4 rout=1000;//in ohms
5 //load resistance Rload
6 Rload=4000;//in ohms
7 C2=50*(10**-12);//in Farads
8 disp('Vout=Vin*(input voltage coupling)*Kv*(output
    voltage coupling)');
9 disp('(i)Mid band gain');
10 disp('Av=(Vout/Vin)=Kv*(Rload/(rout+Rload))');
11 Av=Kv*(Rload/(rout+Rload));

```

```

12 disp('(ii)For the upper half-power frequency');
13 disp('fu=(1/(2*pi))*(1/(C2*(rout||Rload)))');
14 fu=(1/(2*pi))*(1/(C2*((rout*Rload)/(rout+Rload))));
15 //Let coupling capacitance be C
16 disp('(iii)For coupling capacitance C ');
17 disp('fl=(1/(2*pi))*(1/(C*rin))');
18 fl=10; //given lower half-power frequency fl
19 disp('C=1/(2*pi*fl*rin)');
20 C=1/(2*pi*fl*rin);
21 disp('RESULTS:\n');
22 printf('(i)Mid-band gain=%d\n',Av);
23 printf('(ii)Upper half-power frequency=%1.2f MHz\n',
    fu/(10**6)); //fu divided by 10^6 to convert into
    MHz
24 printf('(iii)Coupling capacitance C for fl to be 10
    Hz is %1.2f uF\n',C*(10**6)); // C multiplied by
    10^6 by convert to microFarads

```

Scilab code Exa 3.3 Common mode rejection ratio and differential mode

```

1
2 disp(" given:")
3 disp("The voltage generated by an transducer which
    is connected to differential amplifier is Vdm=50
    mV ")
4 Vdm=50*(1/1000) //Vdm in volts
5 disp("Vcm=5V")
6 Vcm=5
7 disp("Vout should be equal to 10V")
8 Vout=10
9 disp("The unwanted output component owing to the
    common-mode input is to be less than 1% of the
    wanted component i.e Voutcm=1% of Voutdm")
10 Voutcm=(1/100)*10
11 printf(" Voutcm=%f" ,Voutcm)

```

```

12 disp("Let amplifier differential-mode gain be Adm")
13 disp("Vout,Admand Vdm are related as")
14 disp("Vout=Adm*Vdm-----(1)")
15 disp("Adm=Vout/Vdm")
16 Adm=Vout/Vdm
17 printf(" Therefore amplifier differential-mode gain(
    Adm)=%d" ,Adm)
18 disp("Equation (1)can also be used for calculating
    unwanted output component")
19 disp("Unwanted components arise when operating in
    common-mode")
20 disp("Acm=Voutcm/Vcm")
21 Acm=Voutcm/Vcm
22 printf(" Common-mode gain Acm=%f" ,Acm)
23 disp("Now CMRR can be calculated using relation")
24 disp("CMRR=Adm/Acm")
25 CMRR=Adm/Acm
26 printf(" There fore CMRR of amplifier should be
    greater than or equal to %f" ,CMRR)
27 printf(" CMRR>=%d" ,CMRR)

```

Chapter 4

FEEDBACK AMPLIFIER CIRCUITS

Scilab code Exa 4.1 input resistance of a feedback amplifier

```
1 //input resistance in Kohm
2 Rin=1;
3 //voltage gain
4 Av=1000;
5 //feedback fraction
6 Bv=0.1;
7 //Let input resistance after feedback is applied be
  Zif
8 AB=Av*Bv;
9 Zif=Rin*(1+AB); //Zif in Kohms
10 printf("RESULTS:\n");
11 printf("Input resistance after feedback is applied
  Zif=%dKohms",Zif);
```

Scilab code Exa 4.2 input resistance of a feedback amplifier

```

1 //input impedance in ohms
2 Rin=10000;
3 //Trans-resistance in ohms
4 Rt=(10)^(5);
5 //feedback fraction Bg
6 Bg=(10)^(-3);
7 AB=Rt*Bg;
8 //input impedance after feedback is applied Zif in
  ohms
9 Zif=Rin/(1+AB);
10 printf("RESULTS:\n");
11 printf("Input impedance after feedback applied is
  Zif=%d Ohms",Zif);

```

Scilab code Exa 4.3 parameters of a feedback amplifier

```

1 //Trans-resistance Rtf in Kohms
2 Rtf=(-10);
3 //given
4 R1=20;//in Kohms
5 R2=5;
6 Rc=1;
7 //transistor parameters in kohms
8 hie=1;
9 hfe=0.1;
10 //feedback fraction Bg
11 Bg=1/Rtf;
12 R=(-Rtf);
13 //input resistance ri in ohms
14 a=(R1*R2*hie)/((R1*R2)+(R1*hie)+(R2*hie));
15 ri=(a);
16 //Output resistance in ohms
17 ro=1000*Rc;
18 b=(R1*R2)/(R1+R2);
19 //let

```

```

20 i1=1;
21 ib=i1*b/(b+hie);
22 ic=hfe*ib;
23 //output voltage Vo
24 Vo=(-Rc*ic);
25 Rt=Vo/i1*1000;
26 //Feedback factor F
27 F=1+(Rt*Bg);
28 //closed loop gain Rtf
29 Rtf=Rt/F;
30 //closed loop input resistance rif
31 rif=ri/F;
32 //closed loop output resistance rof
33 rof=ro/F;
34 printf("RESULTS\n");
35 printf("closed-loop gain ,Rtf=%1.2fKohms\n",Rtf);
36 printf("closed-loop input resistance rif=%2.0f Ohms\n
n",1000*rif);//in ohms
37 printf("closed-loop output resistance rof=%d Ohms",
rof);

```

Scilab code Exa 4.4 series voltage feedback circuit

```

1 //caption:series voltage feedback circuit
2 //given in ohms
3 R1=10000;
4 R2=2000;
5 Rc=2000;
6 //transistor parameters in Ohms
7 hie=1000;//in ohms
8 hfe=100;
9 //loop gain Avf
10 Avf1=(R1+R2)/R2;//there is corrected equation.in
text book,it needs a correction
11 Bv=1/Avf1;

```



```

12 //forward voltage gain Av
13 Av=(0.5)*hfe*Rc/hie;
14 //feedback factor F
15 F=(1+Av*Bv);
16 //closed loop gain Avf
17 Avf=Av/F;
18 //input resistance ri
19 ri=2*hie;
20 //input resistance after feed back rif
21 rif=ri*F;
22 //output resistance ro
23 ro=Rc;
24 //output resistance after feedback rof
25 rof=ro/F;
26 printf("RESULTS\n");
27 printf("(i)voltage amplification Avf=%1.2f, This
    accords with the previous value of Avf=%d\n",Avf,
    Avf1);
28 printf("(ii)input resistance after feed back rif=%2
    .1fKohms\n",rif/1000);//to convert ohms to Kohms
29 printf("(iii)input resistance after feed back rof=
    %dKohms\n",rof);//to convert ohms to Kohms

```

Chapter 5

MORE ABOUT FEEDBACK AMPLIFIERS

Scilab code Exa 5.1 Voltage shunt feedback circuit

```
1 //caption:shunt voltage feedback circuit
2 //example5.1
3 printf(" Given :");
4 printf("R=10 Kohms ,\nR1=20Kohms ,\nR2=5Kohms ,\n hfe
   =100Kohms , hie=1Kohm ,\n Rc=1Kohm");
5 R=10000;
6 R1=20000; //in ohms
7 R2=5000;
8 Rc=1000;
9 hfe=100;
10 Bg=-1/R;
11 hie=1000;
12 printf(" Therefore Bg=%1.5 f\n",Bg);
13 printf("RT=input current coupling factor*- hfe*
   effective ccollector resistance\n");
14 printf("RT=(R1 || R2 || R) / ((R1 || R2 || R)+hfe) * hfe *(Rc || R)
   \n");
15 a=(R1*R2*R) / ((R1*R) + (R*R2) + (R1*R2));
16 b=(Rc*R) / (Rc+R);
```

```

17 RT=(-hfe*a*b)/(a+hie)/1000;
18 printf("RT=%2.1fKohms\n",RT);
19 c=1+(RT*1000*Bg);
20 d=(1/R1)+(1/R2)+(1/R)+(1/hie);
21 Zi=1/d;
22 Zo=b;
23 RTf=RT/c;
24 rif=Zi/c;
25 rof=Zo/c;
26 printf("RESULTS:\n");
27 printf("RT=%2.1fKohms\n",RT);
28 printf("Zi=%3.1fohms\n",Zi);
29 printf("Zo=%dohms\n",Zo);
30 printf("Closed-loop gain ,RTf=%1.2fKohms\n",RTf);
31 printf("Closed-loop input resistance ,rif=%dohms\n",
    rif);
32 printf("Closed-loop output resistance ,rof=%dohms\n",
    rof);

```

Scilab code Exa 5.2 amplifier parameters

```

1 //caption:amplifier parameters
2 //given values of resistances
3 R1=10000;//in ohms
4 R2=2000;//in ohms
5 Rc=2000;//in ohms
6 hie=1000;//in ohms
7 hfe=100;
8 ri1=2*hie;
9 //unloaded feed-backfraction Bv
10 Bv=R2/(R1+R2);
11 disp("Av=ri1/(ri1+(R1||R2))*(Vo/Via)");
12 disp('Av=(1*hfe/2)*(ri1/(ri1+(R1||R2)))*(Rc||(R1+R2)
    )/hie');
13 Av=(1*hfe/2)*(ri1/(ri1+(R1*R2/(R1+R2))))*(Rc*(R1+R2)

```

```

        /(Rc+R1+R2))/hie;
14 C=1+(Av*Bv);
15 //open-loop input resistance ri
16 ri=ri1+(R1*R2/(R1+R2));
17 //open-loop output resistance ro
18 ro=Rc*(R1+R2)/(Rc+R1+R2);
19 //closed-loop gain ,Avf
20 disp('closed-loop gain , Avf=Av/C, where C=1+Av*Bv');
21 Avf=Av/C;
22 //closed-loop input resistance ,rif
23 disp('closed-loop input resistance , rif=ri*C');
24 rif=ri*C;
25 //closed-loop output resistance , rof
26 disp('closed-loop output resistance , rof=ro/C, where
        C=1+Av*Bv');
27 rof=ro/C;
28 printf("RESULTS:\n");
29 printf("Closed-loop gain ,Avf=%1.2f( previously 5.66)\n",Avf);
30 printf("Closed-loop input resistance ,rif=%2.1f Kohms
        (previously 35.3Kohms)\n",rif/1000);//divided by
        1000 to convert to Kohms
31 printf("Closed-loop output resistance ,rof=%3.0f Ohms
        (previously 113 ohms)\n",rof);

```

Scilab code Exa 5.4 closed loop current gain input output resistance

```

1
2 //example5.4
3 //Shunt circuit feedback configuration//given
4 R1=10000;//in ohms
5 R2=2000;//in ohms
6 Rc1=5000;//in ohms
7 hie=1000;//in ohms
8 hfe=100;

```

```

 9 //unloaded feedback fraction B1
10 B1=R2/(R1+R2);
11 disp('open loop forward current amplification A1=io/
      i ');
12 iia=1;//let
13 disp('ib=input-current coupling factor*iia ');
14 disp('ib=(R1+R2)*iia/(R1+R2+hie) ');
15 ib=(R1+R2)*iia/(R1+R2+hie);
16 disp('V2=(-hfe)*(Rc1||ri2)*ib ');
17 disp('ri2=hie+(1+hfe)*(R1||R2) ');
18 ri2=hie+(1+hfe)*(R1*R2/(R1+R2));
19 V2=(-hfe)*(Rc1*ri2/(Rc1+ri2))*ib;
20 disp('io/V2=C=-hfe/(hie+(1+hfe)*(R1||R2)) ');
21 C=-hfe/(hie+(1+hfe)*(R1*R2/(R1+R2)));
22 disp('Open loop current gain=A1=io/i=(ib/i)*(V2/ib)
      *(io/V2) ');
23 A1=(ib/iia)*(V2/ib)*(C);
24 disp('Open-loop input resistance , ri=(R1+R2)||hie ');
25 ri=(R1+R2)*hie/(R1+R2+hie);
26 disp('Open-loop output resistance ro as seen by the
      load resistance Rc2 is infinitely large since the
      load is in series with the infinitely large
      collector resistance of the transistor. ');
27 disp('Closed-loop current gain, A1f=A1/(1+A1*B1) ');
28 A1f=A1/(1+A1*B1);
29 disp(A1f);
30 disp('Closed-loop input resistance , rif=ri/(1+A1*B1) '
      );
31 rif=ri/(1+A1*B1);
32 disp(rif);
33 disp('Closed-loop output resistance , rof=ro(1+A1*B1) ,
      since ro is infinitely large ,the rof is also
      large infinitely .. ');
34 printf(" RESULTS:\n\n");
35 printf(" A1f=%1.2f ,\n\n",A1f);
36 printf(" rif=%2.1f ohms,\n\n",rif);//approximately
37 printf(" rof=infinite\n\n");

```

Scilab code Exa 5.5 Voltage shunt feedback circuit

```
1 //important:In this example ro1 value is
   contradicting in text book,sometimes they used 1
   KOhm and sometimes 4Kohms,the code below used ro1
   =2KOhms as specified in question
2 //example5.5
3 //caption:Shunt voltage feedback circuit
4 //input resistance ri1
5 ri1=1000;//in ohms
6 //output resistance ro1
7 ro1=2000;//in ohms
8 //trans resistance Kr
9 Kr=-10^6;//in ohms
10 //Feedback resistor R
11 R=10000;//in ohms
12 //current source Jg
13 Jg=0.001;//in Amps
14 //source internal resistance rg
15 rg=2000;//in ohms
16 //load resistance rl
17 rl=5000;//in ohms
18 //unloaded feedback fraction
19 Bg=-1/R;
20 disp('Unloaded feedback fraction ,Bg=-1/R');
21 disp('open loop gain ,RTs=Vo/Jg=input current
   coupling factor*Kr*output voltage coupling factor
   ');
22 RTs=((rg*R/(rg+R))/(ri1+(rg*R/(rg+R))))*Kr*((rl*R/(
   rl+R))/(ro1+(rg*R/(rg+R))));
23 printf("RTs=%d ohms",RTs/1000);
24 disp('Open-loop input resistance ,ris=rg ||R|| ri1 ');
25 ris=rg*R*ri1/(rg*R+R*ri1+rg*ri1);
26 printf("\nris=%d ohms",ris);
```

```

27 disp('Open-loop output resistance , ros=ro1 || R || rl ');
28 ros=rl*R*ro1/(rl*R+R*ro1+rl*ro1);
29 printf(" ros=%d Ohms", ros);
30 disp(' Closed-loop gain , RTf=RTs/(1+RTs*Bg) ');
31 RTfs=RTs/(1+RTs*Bg);
32 printf("\nRTfs=%1.2 f Kohms", RTfs/1000);
33 disp(' Closed-loop input resistance , rifs=ris/(1+RTs*
      Bg) ');
34 rifs=ris/(1+RTs*Bg);
35 printf(" rifs=%d Ohms", rifs);
36 disp(' Closed-loop output resistance , rof=ros/(1+RTs*
      Bg) ');
37 rofs=ros/(1+RTs*Bg);
38 printf(" rofs=%2.1 f Ohms", rofs);
39 printf(" RESULTS:\n");
40 Vout=RTfs*Jg;
41 printf("\n(i)The output voltage=%1.2 f V", Vout);
42 disp(' rifs=rif*rg/(rif+rg)==>rif=rg*rifs/(rg-rifs) ')
      ;
43 rif=rg*rifs/(rg-rifs);
44 printf("\n(ii)The input resistance seen by the
      actual signal source is %2.1 f Ohms", rif);
45 disp(' rofs=rof*rl/(rof+rl)==>rof=rl*rofs/(rl-rofs) ')
      ;
46 rof=rl*rofs/(rl-rofs)
47 printf("\n(iii)The output resistance seen by the
      load is %2.1 f Ohms", rof);
48 printf("(iV)The closed-loop gain of the amplifier");
49 disp(' RTf=Vo/i1 , where ')
50 disp(' i1=current coupling factor*Jg=rg*Jg/(rg+rif) ')
      ;
51 RTf=(rg+rif)*RTfs/rg;
52 printf("(iV)The closed-loop gain of the amplifier
      circuit is %1.2 f Kohms", RTf/1000); //divided by
      1000 to convert ohms to Kohms.

```

Scilab code Exa 5.6 feed back fraction

```
1
2 //given
3 //current gain of transistor A
4 A=100;
5 //upper half-power frequency fh
6 fh=2*(10^6); //in Hz
7 //closed-loop current gain Acl
8 Acl=10;
9 //band width BW
10 BW=10*(10^6); //in Hz
11 //gain bandwidth product ABWp
12 ABWp=Acl*BW; //in Hz
13 //gain bandwidth product of transistor ABWpt
14 ABWpt=A*fh; //in Hz
15 printf("The gain bandwidth product of the transistor
        is %d*(10^8), which is greater the minimum
        required i.e. %d*(10^8) and therefore the
        transistor is suitable\n", ABWpt/(10^8), ABWp
        /(10^8))
16 disp(' Acl=A/(1+A*B) ==> B =(1/A)*(A/Acl-1)\n');
17 B =(1/A)*(A/Acl-1);
18 printf("Required value of B is %1.2f", B);
```

Chapter 6

THE OP AMPS BASIC IDEAS AND CIRCUITS

Scilab code Exa 6.1 design values of an inverting amplifier

```
1
2 R1=1;
3 //voltage gain of an inverting amplifier Av
4 Av=-100;
5 printf("Av=(-R2/R1)\nTherefore R2=-Av*R1");
6 R2=-Av*R1;
7 printf("\nRESULTS:\n");
8 printf("The design values are R1=%d Kohms and R2=%d
    Kohms",R1,R2);
```

Scilab code Exa 6.2 gain of a negative feedback amplifier

```
1
2 //Typical 741 type op-amp Differential gain A
3 A=200000;
4 R1=1; //in Kohms
```

```
5 R2=10;
6 //Circuit gain in negative feedback Av
7 Av=- (R2/R1)*(A*(R1/(R1+R2)))/(1+(A*(R1/(R1+R2)))));
8 printf("Circuit gain Av=%2.4f",Av);
```

Scilab code Exa 6.4 non inverting amplifier

```
1
2 Av=100;
3 printf("Av=1+(R1/R2)\n");
4 //Let R2=1 Kohms
5 R2=1; //in Kohms
6 printf("R1=(Av-1)*R2");
7 R1=(Av-1)*R2;
8 printf("\nRESULTS:\n");
9 printf("The design values are R1=%d Kohms and R2=%d
    Kohm",R1,R2);
```

Chapter 7

OP AMP NON IDEALITIES

Scilab code Exa 7.1 Output voltage

```
1
2 A=200000;
3 //Offset voltage Vos in mV
4 Vos=5;
5 //output voltage Vout=A*Vos in mV
6 printf("\nVout=A*Vos\n");
7 Vout=A*Vos;
8 printf("RESULTS:\n")
9 printf("The maximum output voltage Vout=%d V",Vout
    /1000);//in volts
```

Scilab code Exa 7.2 output offset voltage

```
1
2 printf("A=1+(R1/R2)\n");
3 A=10;
4 //out put voltage of an op-amp owing to voltage
    offset effects Vout
```

```

5 printf("Vout=-A*Vos");
6 Vos=5; //in mV
7 Vout=-A*Vos;
8 printf("\nRESULTS:\n");
9 printf("Output offset voltage is Vout=%d mV",Vout);

```

Scilab code Exa 7.4 inverting voltage amplifier

```

1 Av=-10;
2 disp('Av=-(R2/R1)*(A*B/1+A*B), where A-differential
      gain and B=R1/R1+R2');
3 disp('When A*B>>1,Av=-(R2/R1)=Av1');
4 Av1=Av;
5 disp('abs(A)>>1+(R2/R1)=1+abs(Av1)');
6 A=1+abs(Av1);
7 disp('Because the cross over point occurs at a
      significantly higher frequency than the dominant
      frequency at 10Hz');
8 disp('The phase angle of A is very close to -90
      degrees=%pi/2 radians');
9 disp('Using the j-notation the gain A at the
      crossover point is equal to %i*11');
10 A=%i*11;
11 disp('1+abs(Av1)=(R1+R2)/R1');
12 Av=(-10)*(A/(1+abs(Av1)))/(1+(A/(1+abs(Av1))));
13 Av2=abs(Av);
14 M=Av/Av2;
15 printf("The closed-loop gain at the cross over
      point is thus reduced by a factor %f and
      therefore point corresponds to upper half-power
      frequency which is 90KHz",M)
16 disp('Therefore fh=90KHz, fl=0KHz');
17 fh=90; //in KHz
18 fl=0; //in KHz
19 disp('Bandwidth BW=fh-fl');

```

```
20 BW=fh-f1;
21 printf(" RESULTS:\n");
22 printf(" The closed-loop bandwidth is %d KHz",BW);
```

Scilab code Exa 7.5 Full power bandwidth frequency

```
1
2 SR=0.5*(10^6);
3 //given operating power supplies +-12V
4 printf("\nGiven operating power supplies +-12V\n");
5 printf("For 741 op-amp, the output can work typically
   to within 1V of the power supplies.\nSo, Vom=11V\n
   n");
6 Vom=11; //in volts
7 printf(" f=(1/(2*pi))*(SR/Vom)\n");
8 f=(1/(2*pi))*(SR/Vom);
9 printf("RESULTS:\n");
10 printf(" Full-power bandwidth frequency f=%d Hz",f);
```

Chapter 8

SELECTED OP AMP APPLICATIONS

Scilab code Exa 8.1 basic differential amplifier

```
1
2 Adm=100;
3 printf("Gain of an difference amplifier Adm=(1+(R2/
   R1))/(1+(R3/R4))\n");
4 printf("Let (R2/R1)=A and (R3/R4)=B\n");
5 printf("Therefore Adm=(1+A)/(1+B)\n");
6 printf("A=Adm,");
7 printf(" gives B=1/Adm\n");
8 A=Adm;
9 B=1/Adm;
10 printf("By suitable selection of resistors say R1=R2
   =1 Kohm");
11 R1=1; //in Kohm
12 R3=1;
13 R2=Adm*R1;
14 R4=Adm*R3;
15 printf("\nRESULTS:\n");
16 printf("Reasonable set of resistors R1=%d Kohm,R2=%d
   Kohms ,R3=%d Kohm,R4=%d Kohms." ,R1 ,R2 ,R3 ,R4 );
```

Scilab code Exa 8.2 Instrumental amplifier

```
1
2 Adm=100;
3 printf("Adm=100=10*10, therefore it requires (R2/R1)
   =10 Kohms and (R4/R3)=10 Kohms\n");
4 printf("Therefore if R1=R3=10 Kohms, then R2=R4=100
   kohms\n");
5 R1=10; //in Kohms
6 R2=100;
7 R3=10;
8 R4=100;
9 printf("Adm=(1+2(Rb/Ra))*R4/R3\n");
10 printf("(Rb/Ra)=B\n");
11 printf("Adm=(1+2*B)*R4/R3\n");
12 B=((Adm/(R4/R3))-1)/2;
13 printf("Rb/Ra=%1.1f\n", B);
14 printf("If Ra=10 Kohms\n");
15 Ra=10; //in kohms
16 Rb=B*Ra;
17 printf("\nRESULTS:\n");
18 printf("The value of resistors are R1=%d Kohms, R2=%d
   Kohms, R3=%d Kohms, R4=%d Kohms, Ra=%d Kohms, Rb=%d
   Kohms", R1, R2, R3, R4, Ra, Rb);
```

Scilab code Exa 8.3 wein bridge oscillator

```
1 //caption:wein bridge oscillator
2 //To design a wein bridge oscillator of frequency fo
   =10KHz
3 fo=10000; //in Hz
```

```

4 printf("Capacitance C, Resistance R and Frequency fo
   are related as  $C \cdot R = 1 / (2 \cdot \pi \cdot f_o)$ \n");
5 printf("If  $C \cdot R = A$ \n");
6 A = 1 / (2 * %pi * fo);
7 C = 0.01 * (10^-6); //in farads
8 R = A / C; //in ohms
9 B = R / 1000; //to convert to Kohms
10 printf("Taking C=0.01microfarad , we get R=%1.2f
   Kohms\n", B); //in Kohms
11 printf("Gain of a non-inverting amplifier should be
   3 i.e.  $A_v = 1 + (R_1 / R_2) = 3$ \n");
12 printf("This gives  $(R_1 / R_2) = 2$ , by selecting  $R_2 = 10$ 
   Kohms, we get  $R_1 = 20$  Kohms\n");
13 R2 = 10;
14 R1 = 20; //in Kohms
15 printf("The gain must be restricted between 2.8 to
   3.2, by selecting proper resistors Ra and Rb such
   that  $R_1 = R_a || R_b$  this can be achieved.\n");
16 printf("These values of Ra and Rb comes out to be Ra
   =22Kohms and Rb=220Kohms\n");
17 printf("RESULTS:\n");
18 printf("design values are R=%1.2f Kohms, C=0.01
   microFarad ,Ra=22 Kohms,Rb=220 Kohms\n", B);

```

Chapter 9

FURTHER OP AMP APPLICATIONS

Scilab code Exa 9.1 Design of phase shifter

```
1 //caption:Design of phase shifter
2 //example9.1
3 disp(" Given frequency fo=10 KHz,Vrms=5 V, phi=10
      degrees\n");
4 disp(" Taking A=C3*R4\n");
5 phi=10;//in degrees
6 fo=1000;//in Hz
7 disp(" phi=180-2(atan(2*%pi*f*A))");
8 A=tan((180-10)*%pi/(180*2))/(2*fo*%pi);
9 printf(" Therefore A=C3*R4=%f sec.\n",A);
10 R4=10000;//let (in ohms)
11 printf("C3 and R4 values are selected such that
      their product equals or greater than %f, The
      above values are preferable for low cost and bias
      compensation",A);
12 C3=A/R4;
13 printf("\nC3=%f uF",C3*10^6);
14 disp("To lower the cost of design ,the preferred
      value is C31=0.22uF");
```

```

15 C31=0.22; //let Such that C31>C3
16 disp(" since ,C31*R4>A,C31 can be preferred")
17 printf(" Similarly , R1 and R2 values should be of
    Good matching to obtain accurate unity gain
    modulus ");
18 printf("RESULTS:\n");
19 printf(" (i) Resistors , R1=R2=10Kohms\n");
20 printf(" (iii) R4=%d Kohms\n",R4/1000); //divided by
    1000 to display in Kohms
21 printf(" (iii) Capacitor , C3=%1.2 f uF\n",C31);

```

Scilab code Exa 9.2 design of non inverting amplifier

```

1 //caption:design of non inverting amplifier
2 //to design a non inverting amplifier
3 //mid-band gain Av
4 Av=100;
5 //input impedance Zin
6 Zin=1000000; //in ohms
7 //cut-off frequencies fl1 and fl2
8 fl1=10; //in Hz
9 fl2=10; //in Hz
10 //From problem it follows
11 R3=Zin;
12 disp(' fl2 =1/(2*%pi*C2*R3) ');
13 disp(' C2=1/(2*%pi*fl2*R3) ');
14 C2=1/(2*%pi*fl2*R3);
15 disp(' Av=1+(R2/R1) ');
16 disp(' Bias compensation can also be obtained by
    taking R2=R3=1MHz ');
17 R2=R3;
18 disp(' R1=1/(Av-1)*R2 ');
19 R1=1/(Av-1)*R2;
20 printf(' R1=%1.1 f Kohms, preferred and standard value
    is 10Kohms\n',R1/1000);

```

```

21 R=R1;
22 R1=10000; //in ohms
23 disp('f11=1/(2*%pi*C1*R3)');
24 disp('C1=1/(2*%pi*f11*R3)');
25 C1=1/(2*%pi*f11*R1);
26 printf('RESULTS: Design summary\n');
27 printf('R1=%1.1f Kohms, preferred and standard value
        is 10Kohms\n',R/1000);
28 printf('R2=R3=%d Mohm\n',R3/10**6); //divided by 10^6
        to convert to Mohms
29 printf('C1=%1.2f uF but standard and preferred value
        is 2.2 uF\n',C1*(10**6));
30 printf('C2=%2.1f nF but standard and preferred value
        is 22 uF\n',C2*(10**9));

```

Scilab code Exa 9.3 low pass second order filter

```

1 //caption:low pass second order filter
2 //to design a low pass second order filter
3 //since it is butterworth filter
4 Q=1/sqrt(2);
5 Wo=100; //in rad/sec.
6 H=-2;
7 bo=H;
8 a1=1/(Q*Wo);
9 a2=1/(Wo**2);
10 printf("It is convenient to choose R=100Kohms\n");
11 R=100; //in Kohms
12 R3=R;
13 R4=R;
14 R1=R4/(-bo);
15 a3=R3+R4+((R3*R4)/R1);
16 C5=(a1/a3)*(10**6); //in uF
17 C2=a2/(R3*R4*C5)*(10**9); //in nF
18 printf("RESULTS:\n");

```

```
19 printf("R=%dKohms, \ nR1=%dKohms, \ nR3=%dKohms, \ nR4=
    %dKohms, \ nC2=%1.3 f nF, \ nC5=%2.1 fuF" , R , R1 , R3 , R4 , C2
    , C5);
```
