Scilab Textbook Companion for Electrical Machines 3rd Edition by S. K. Bhattacharya[1]

Created by
Devavarapu Hemanth Kumar
B TECH
Electrical Engineering
NIT Durgapur
College Teacher
Dr. Sankar Narayan Mahato
Cross-Checked by

May 23, 2016

Book Description

Title: Electrical Machines 3rd Edition

Author: S. K. Bhattacharya

Publisher: Tata McGraw - Hill Education, New Delhi

Edition: 3

Year: 2009

ISBN: 9780070669215

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

5

7

Chapter 2

Direct Current Machines

Scilab code Exa 2.4 Calculating average induced emf

```
 1  //Calculating average induced emf
 2  //Chapter 2
 3  //Example 2.4
 4  //page 92
 5  clear;
 6  clc;
 7  disp("example 2.4")
 8  P=2           //number of poles
 9  Z=400         //number of conducters
10  n=300         //speed in rpm
11  E=200          //voltage of generator
12  A=2            //number of parallel paths
13  N=1200         //number of turns in each field coil
14  phi=(E*60*A)/(Z*n*P)    //flux at the end of 0.15sec
15  t=0.15                        //time
16  printf("magnitude of flux at the end of 15sec is %f
        wb",phi)
17  e=N*(phi/t)
18  printf("\ninduced emf in the field coil= %d volts",e
        )
```

Scilab code Exa 2.5 Calculating useful flux per pole

```
1  //Calculating the current and power factor of the
       primary circuit
2  //Chapter 3
3  //Example 3.5
4  //page 206
5  clear;
6  clc;
7  disp("Example 3.5")
8  I2=300;..........................//Secondary current
       in amperes
9  N1=1200;                         //number of primary
       turns
10 N2=300;                          //number of
       secondary turns
11 I0=2.5;                          //load current in
       amperes
12 I1=(I2*N2)/N1;
13 phi0=acosd(0.2);
14 phi2=acosd(0.8);
15 I1c=(I1*cosd(phi2))+(I0*cosd(phi0));
16 I1s=(I1*sind(phi2))+(I0*sind(phi0));
17 I=sqrt(I1c^2+I1s^2);
18 phi=atand(I1s/I1c)
19 printf("primary power factor=%fdegrees",cosd(phi));
```

Scilab code Exa 2.6 Calculating emf generated on open circuit condition

```
1  // Calculating  emf  generated  onopen  circuit  condition
2  // Chapter  2
3  // Example  2.6
4  // page  93
5  clear ;
6  clc ;
7  disp ( " example  2.5 " )
8  P =8            // number  of  poles
9  A =8            // number  of  parallel  paths  in  the
        armature
10  Z =960          // number  of  conductors
11  N =400          // speed  in  rpm
12  phi =0.04       // flux  per  pole
13  E =( phi * Z * N * P ) /(60* A )        // emf  generated  onopen
        circuit  condition
14  printf ( " emf  generated  on  open  circuit  condition , E=
        %d  volts " ,E )
```

Scilab code Exa 2.7 calculate induced emf

```
1  // calculate  induced  emf
2  // Chapter  2
3  // Example  2.7
4  // page  97
5  clear ;
6  clc ;
7  disp ( " example  2.7 " )
8  disp ( " flux  is  constant " )
9
10  E =180; . . . . . . . . . . . . . . // induced  emf  at  500rpm
11  N =500; . . . . . . . . . . . . . . . . // speed  in  rpm
12  K1 =( E / N )
13  printf ( " K1=%f " , K1 )
```

```
14  E1 =( K1 *600)                  // induced  emf  at  600rpm
15  printf("\n induced  emf  at  600rpm  is=%d V",E1)
```

Scilab code Exa 2.8 calculating the speed and percentage increase in flux

```
 1  // calculating  the  speed  and  percentage  increase  in
        flux
 2  // Chapter  2
 3  // Example  2.8
 4  // page  97
 5  clear ;
 6  clc ;
 7  disp ("example  2.8")
 8  disp ("assuming  constant  flux")
 9  E1 =220;           // induced  emf  at  N1  speed  in  volts
10  N1 =750;           //  speed
11  K1 =( E1/N1 )
12  E2 =250;           // induced  emf  at  speed  N2
13  N2 = E2 / K1
14  printf ("speed  at  induced  emf  of  250V  =%d rpm",N2)
15  disp ("when  induced  emf  is  250V  and  speed  700  rpm")
16  E3 =250;           // induced  emf  at  N3  speed
17  N3 =700;           // speed
18  ratio =( E3*N1 )/( E1*N3 )
19  Pi =( ratio -1) *100
20  printf ("percentage  increase  in  flux  is  %f  percent",
        Pi )
```

Scilab code Exa 2.9 Calculating electromagnetic torque

```
1  // Calculating electromagnetic torque
2  // Chapter 2
3  // Example 2.9
4  // page 98
5  clear;
6  clc;
7  disp("example 2.9")
8  E=200        // emf induced
9  I=15          // armature current
10 n=1200            // speed in rpm
11 omega=(2*3.14*n)/60;
12 printf("omega=%f \n",omega)
13 T=(E*I)/omega;
14 printf("electromagnetic torque=%f Nm",T)
```

Scilab code Exa 2.10 calculating the torque developed

```
1  // calculating the torque developed
2  // Chapter 2
3  // Example 2.10
4  // page 98
5  clear;
6  clc;
7  disp("Example 2.10")
8  n=10;              // number of turns in 1 coil
9  l=0.2;
10 d=0.2;          // diameter in metres
11 B=1;              // uniform magnetic field density
       in weber per m^2
12 N=1500;            // speed in rpm
13 r=(d/2);          // radius in metres
14 E=(B*l*((2*3.14*N)/60)*r*2*n);
15 printf("total induced emf=%f V",E)
```

```
16  R=4;                          //total resistance in ohms
17  I=E/R;
18  printf("\nThe current through the armature coil when
        connected to the load,I=%f A",I)
19  T=(E*I)/((2*3.14*N)/60)
20  printf("\ntorque=%f Nm",T)
```

Scilab code Exa 2.11 calculating various parameters of dc motor

```
 1  //calculating various parameters of dc motor
 2  //Chapter 2
 3  //Example 2.11
 4  //page 99
 5  clear;
 6  clc;
 7  disp("Example 2.11")
 8  V=230;               //armature voltage supply in volts
 9  Ia=12;               //armature current in amperes
10  Ra=0.8;              //armature resistance in ohms
11  N=100;               //speed in radian per second
12  E=(V-(Ia*Ra))
13  printf("induced emf,E=%fV",E)
14  Te=(E*Ia)/N
15  printf("\nthe electromagnetic torque=%fNm",Te)
16  Pi=V*Ia
17  printf("\nelectrical input to the armature,Pinput=
        %dW",Pi)
18  Pd=Te*N
19  printf("\nmechanical developed=%fW",Pd)
20  loss=(Ia^2*Ra)
21  printf("\narmature copper loss=%fW",loss)
```

Scilab code Exa 2.12 calculating various parameters of dc machine

```
1  //calculating various parameters of dc motor
2  //Chapter 2
3  //Example 2.11
4  //page 99
5  clear;
6  clc;
7  disp("Example 2.11")
8  V=230;              //armature voltage supply in volts
9  Ia=12;             //armature current in amperes
10 Ra=0.8;             //armature resistance in ohms
11 N=100;                //speed in radian per second
12 E=(V-(Ia*Ra))
13 printf("induced emf,E=%fV",E)
14 Te=(E*Ia)/N
15 printf("\nthe electromagnetic torque=%fNm",Te)
16 Pi=V*Ia
17 printf("\nelectrical input to the armature,Pinput=
       %dW",Pi)
18 Pd=Te*N
19 printf("\nmechanical developed=%fW",Pd)
20 loss=(Ia^2*Ra)
21 printf("\narmature copper loss=%fW",loss)
```

Scilab code Exa 2.13 calculating speed of machine

```
1  //calculating speed of machine
```

```
2  //Chapter 2
3  //Example 2.13
4  //page 101
5  clear;
6  clc;
7  disp("Example 2.13")
8  disp("At generator condition")
9  P=50000;              //power delivered in watts
10 V=250;               //voltage in volts
11 Ra=0.02;             //armature resistance in ohms
12 Rf=50;               //field resistance in ohms
13 If=V/Rf              //field current in amperes
14 Ng=400;              //speed in generating condition
       in rpm
15 printf("field current ,If=%dA",If)
16 Il=P/V               //load current in amperes
17 printf("\nLoad current ,If=%dA",Il)
18 Ia=If+Il             //armature current in amperes
19 printf("\nAramture current ,If=%dA\n",Ia)
20 Eg=(V+(Ia*Ra))
21 disp("At motor condition")
22 Ia=(Il-If)
23 printf("Aramture current ,If=%dA",Ia)
24 Em=(V-(Ia*Ra))
25 printf("\nEm=%fV",Em)
26 Nm=(Ng*Em)/Eg
27 printf("\nSpeed of the motor=%drpm",Nm)
```

Scilab code Exa 2.14 calculating speed ratio of generator and motor working conditios

```
1  //calculating speed ratio of generator and motor
       working conditios
```

```
2  //Chapter 2
3  //Example 2.14
4  //page 101
5  clear;
6  clc;
7  disp("Example 2.14")
8  V=250;                      //voltage supply in volts
9  Ra=0.12;                    //armature resistance in ohms
10 Rf=100;                      //field resistance in ohms
11 Il=80;                      //load current in amperes
12 If=V/Rf
13 printf("Field current,If=%f",If)
14 disp("When machine is generating")
15 Ia=Il+If
16 Eg=(V+(Ia*Ra))
17 printf("\nIa=%fA",Ia)
18 printf("\nEg=%fV",Eg)
19 disp("When machine is motoring")
20 Ia=Il-If
21 Em=(V-(Ia*Ra))
22 printf("\nIa=%fA",Ia)
23 printf("\nEg=%fV",Em)
24 ratio=Eg/Em
25 printf("\nRatio of speeds=%f",ratio)
```

Scilab code Exa 2.15 calculating flux and area of pole shoe and no load terminal voltage

```
1  //calculating flux, area of pole shoe and no-load
       terminal voltage
2  //Chapter 2
3  //Example 2.15
4  //page 102
```

```
 5  clear ;
 6  clc ;
 7  disp (" Example  2.15")
 8  V =550;                  // voltage  supply  in  volts
 9  P =16;               // number  of  poles
10  N =150;                 // speed  in  rpm
11  Z =2500;                  // number  of  armature  conductors
12  A =16;
13  Power =1500000;            // power  in  watt
14  Cl =25000;                  // full −load  copper  loss
15  B =0.9;                  // flux  density  in  the  pole
16  Ia = Power / V
17  printf (" Full  load  current=%fA" ,Ia )
18  Ra = Cl /( Ia ^2)
19  printf (" \ nRa=%fohms" ,Ra )
20  E = V +( Ia * Ra )
21  printf (" \ nInduced  emf=%fvolts" ,E )
22  phi =( E *60* A )/( Z * N * P )
23  printf (" \ nflux  density=%fWb/m^2" ,B )
24  printf (" \ nflux=%fWb" ,phi )
25  area =( phi / B )
26  printf (" \ n  Area  of  pole  shoe=%fcm^2" ,( area *10000))
```

Scilab code Exa 2.16 calculate approximate time of commmutation

```
 1  // calculate  approximate  time  of  commmutation
 2  // Chapter  2
 3  // Example  2.16
 4  // page  103
 5  clear ;
 6  clc ;
 7  disp (" Example  2.16")
 8  Cd =0.76;                  // commutator  diameter  in  metres
```

```
 9  Cr =.38;                    //commutator  radius  in  metres
10  bw =1.5*10^( -2);             //brush  width  in  metres
11  N =600;                      //speed  in  rpm
12  n =10;                       //speed  in  rps
13  V=Cr*(2*3.14*n);
14  printf("peripheral  speed  of  commutator ,V=%fm/ sec" ,V)
       ;
15   Tc=bw/V;
16   printf("\nTime  of  commutation=%fseconds" ,Tc)
```

Scilab code Exa 2.17 calculate resistance

```
 1  // calculate  resistance
 2  // Chapter
 3  // Example  2.17
 4  // page  123
 5  clear;
 6  clc;
 7  disp(" Example  2.17")
 8  V =240;             //supply  voltage  in  volts
 9  N =800;             //speed  in  rpm
10  Ia =2;              //armeture  current  in  amperes
11  Ra =0.4;            //armature  resistance  in  ohms
12  Rf =160;            //field  resistance  in  ohms
13  Il1 =30;            //line  current  in  amperes
14  E=V-(Ia*Ra);        //induced  emf  in  volts
15  disp(" At  no−load")
16  printf(" E=%fV" ,E)
17  If =V/Rf;           //field  current  in  amperes
18  printf("\nIf=%fA" ,If)
19  K1=E/(If*N);
20  printf("\nK1=%f" ,K1)
21  disp(" At  a  load  of  30A")
```

```
22  Ia1 =(Il1 -If ) ;
23  E1=V-(Ia1*Ra ) ;
24  N1 =950;                    // speed  in  rpm
25  If1=E1/(K1*N1 ) ;
26  printf (" If1=%fA\n", If1 ) ;
27  Rr=V/ If1 ;
28  R=(Rr-Rf ) ;
29  printf ("\nExtra  resistance  required  in  the  field
        circuit ,R=%fohms",R)
```

Scilab code Exa 2.18 calculating resistance required in series

```
1  // calculating  resistance  required  in  series
2  //Chapter  2
3  //Example  2.18
4  //page  124
5  clear ;
6  clc ;
7  disp ("Example  2.18")
8  V=230;                    // voltage  supply  in  volts
9  Ia=20;                    // armature  current  in
        amperes
10 Ra=0.5;                   // armature  resistance  in
        ohms
11 E=V-(Ia*Ra ) ;
12 printf ("E=%dV",E)
13 disp ("when  extra  resistance  is  added  in  the  armature
        circuit ,the  speed  is  halved")
14 E2=E/2 ;
15 R=((V-E2)/ Ia)-Ra ;
16 disp ("The  load  torque  is  conatant")
17 printf ("extra  resistance  in  the  armature  circui ,R=
        %fohms",R)
```

```
18  disp("The load torque directly proportional to
        square of speed")
19  disp("if N is halfed , Iais one−fourthed")
20  Ia2=Ia/4;
21  R=((V-E2)/Ia2)-Ra;
22  printf("extra resistance in the armature circui ,R=
        %fohms",R)
```

Scilab code Exa 2.19 calculating resistance required in series and also the speedwhen torque is halfed

```
1  //calculating resistance required in series and also
        the speedwhen torque is halfed
2  //Chapter 2
3  //Example 2.19
4  //page 125
5  clear;
6  clc;
7  disp("Example 2.19")
8  V=250;                    //voltage supply in volts
9  Ia=50;                    //armature current in
        amperes
10 Ra=0.3;                   //armature resistance in
        ohms
11 N=1000;
12 E=V-(Ia*Ra);
13 printf("E=%dV",E)
14 disp("when extra resistance is added in the armature
        circuit when the speed is 800rpm")
15 N2=800;
16 E2=(E*N2)/N;
17 printf("\nE at 800rpm=%dV",E2)
18 R=((V-E2)/Ia)-Ra;
```

```
19  printf("\nextra resistance in the armature circui,R=
        %fohms",R)
20  disp("if load is halfed,Ia will be halfed")
21  Ia2=Ia/2;
22  E1=V-(Ia2*(Ra+R));
23  printf("E1=%dV",E1)
24  N1=(N2*E1)/E2;
25  printf("\nN1=%frpm",N1)
```

Scilab code Exa 2.20 calculating the speed of the motor

```
1  //calculating the speed of the motor
2  //Chapter 2
3  //Example 2.20
4  //page 125
5  clear;
6  clc;
7  disp("Example 2.20")
8  Il=5;                //current in amperes al no-load
9  V=250;               //voltage in volts
10 Rf=250;              //field resistance in ohms
11 If1=V/Rf;            //field current in amperes
12 Ia1=Il-If1;          //armature current
13 Ra=0.2;              //armature resistance in
        ohms
14 disp("at a load current of 50A")
15 Il2=50;              //load current in amperes
16 //armature reaction weakens by 3percent
17 If2=0.97;            //current in amperes
18 Ia2=Il2-If2;
19 N1=1000;
20 E1=(V-(Ia1*Ra));
21 E2=(V-(Ia2*Ra));
```

```
22  N2 =( N1 * E2 ) /( 0 . 9 7 * E1 ) ;
23  printf ( " N2=%frpm " , N2 )
```

Scilab code Exa 2.21 Calculate the fullyload speed of the motor

```
1  // Calculate  the  fully −load  speed  of  the  motor
2  // Chapter  2
3  // Example  2.21
4  // page  126
5  clear ;
6  clc ;
7  disp ( " Example  2.21 " )
8  P =4 ; . . . . . . . . . . . . . . . . . . . // pole
9  V =500 ; . . . . . . . . . . . . . . . . . // shunt  motor   in  volts
10 Ia =60 ; . . . . . . . . . . . . . . . . . . . . . . . // armature  current  in
       amperes
11 Ra =0 . 2 ; . . . . . . . . . . . . . . . . . . . . . . . . . . . // armature
       resistance  in  ohms
12 E =V -( Ia * Ra ) -2 ;
13 printf ( " voltage  drop  across  each  brush=%fV " , E )
14 phi =0 . 0 3 ; . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . // flux  per
        pole  in  Wb
15 Z =720 ; . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . // total
       armature  current  in  volts
16 A =2 ;
17 N =( E * 60 * A ) /( phi * Z * P )
18 printf ( " \ nfull  load  speed  of  the  motor=%frpm " , N )
```

Scilab code Exa 2.22 Calculate the value of resistance

```
1  //Calculate  the  value  of  resistance
2  //Chapter  2
3  //Example  2.22
4  //page  126
5  clear;
6  clc;
7  disp("Example  2.22")
8  V=440;                    //primary  voltage  in  volts
9  Ia=50;                    //armature  current  in  amperes
10  Ra=0.2;                   //armature  resistance  in  ohms
11  N=600;                     //speed  in  rpm
12  E=V-(Ia*Ra);              //emf induced   in  volts
       before  adding  extra  resistance
13  //E=K*phi*N=K1*Ia*N
14  K1=E/(Ia*N);
15  //we  have  the  relation  T=Kt1*Ia^2,  T1=Kt1*Ia1^2
16  //when  torque  is  half ,  say  torque  be  T1
17  //T1=T/2.  r=T/T1
18  r=2;
19  Ia1=sqrt(Ia^2/r);
20  printf("Ia1=%fA",Ia1);
21  //extra  resistance  R is  introduced  in  the  circuit
22  N1=400;
23  E1=(K1*Ia1*N1);
24  R=((V-E1)/Ia1)-Ra;
25  printf("\nvalue  of  extra  resistance  added=%fohms",R)
```

Scilab code Exa 2.23 Calculate the speed

```
1  //Calculate  the  speed
2  //Chapter  2
3  //Example  2.23
4  //page  127
```

```scilab
5  clear;
6  clc;
7  disp("Example 2.23")
8  V=200;                          //voltage in volts
9  Ia=20;                          //armature current in
       amperes
10 Ra=0.5;                          //armature resistance
        in ohms
11 Rse=0.2;                          //field winding
       resistance in ohms
12 E=V-(Ia*(Ra+Rse));
13 printf("In first case,E=%fV",E)
14 //E=k*phi*N
15 N=1000;                          //speed in rpm
16 Kphi=E/N;
17 //a resistance R is connected in parallel with the
       series field which is called diverter
18 disp("when resistace R is added and new conditions")
19 I=20;                          //total current flowing
20 //current is equally devided between series field
       and diverter
21 Ise2=I/2;
22 //flux at 10A current is 20percent of flux at 20A
       current
23 p=0.70;                          //percentage of flux
24 Kpih1=p*Kphi;
25 E1=(V-((Ia*Ra)+(Ise2*Rse)));
26 printf("Induced emf=%fV",E1)
27 //new speed is N1
28 N1=E1/(p*Kphi)
29 printf("\nN1=%frpm",N1)
```

Scilab code Exa 2.24 Calculate the fullyload speed of the motor

```scilab
1  //Calculate  the  fully−load  speed  of  the  motor
2  //Chapter  2
3  //Example  2.24
4  //page  128
5  clear;
6  clc;
7  disp(”Example  2.24”)
8  V=200;...............................//motor  runs  in
       volts
9  Ia=15;............................//current  taken
       in  amperes
10  Ra=1;.................................//motor
       resistance  in  ohms
11  E1=V-(Ia*Ra);
12  printf(”resistance  when  1ohm=%fV”,E1)
13  R=5;.......................................//resistance
14  E2=V-(Ia*(Ra+R))
15  printf(”\nResistance  when  5ohms  connected  in  series=
       %fV”,E2)
16  N1=800;............................//speed  of  motor
       in  rpm
17  N2=N1*(E2/E1);
18  printf(”\nspeed  at  which  motor  will  run  when
       resistance  is  5ohms=%frpm”,N2)
```

Scilab code Exa 2.25 Calculate the ampere turns for each commutating pole

```scilab
1  //Calculate  the  ampere  turns  for  each  commutating
       pole
2  //Chapter  2
3  //Example  2.25
4  //page  135
```

```
5  clear;
6  clc;
7  disp("Example 2.25")
8  P=8;...........................//pole
9  Z=107;..........................//generator with
       slots
10 Ia=1000;.....................//current containing in
       amperes
11 Bag=0.32;.....................//gap flux density in
       Wb/m^2
12 lg=0.012;..........................//interpole air
       gap in meters
13 pi=3.14;
14 Mu=(4*pi*10^-7)
15 AT=(((Ia*Z)/(2*P))+((Bag*lg)/Mu));
16 printf("current for each commutating pole=%f",AT)
```
___

Scilab code Exa 2.26 Estimating the number of turns needed on each commutating pole

```
1  //Estimating the number of turns needed on each
       commutating pole
2  //Chapter 2
3  //Example 2.26
4  //page 135
5  clear;
6  clc;
7  disp("Example 2.26")
8  Bag=0.3;..............................//flux
       density in the interpole air gap in Wb/m^2
9  Ia=200000/200;..........................//armature
       current in amperes
10 printf("Armature current=%f",Ia)
```

```
11  Z=540;............................//Number of armature
        conductors
12  Zt=540/2;...........................//Number
        armature winding turns
13  printf("\nNumber armature winding turns=%f",Zt)
14  A=6;...............//the winding lap
15  Ap=Zt/A;.........................//Number of armature
        turns per parallel path
16  printf("\nNumber of armature turns per parallel path
        =%f",Ap)
17  P=6;...............................//pole
18  Np=((Ia*Ap)/P);
19  printf("\nNumber of armature ampere turns per pole=
        %f",Np)
20  lg=0.01;............................//inter pole
        air gap in meters
21  pi=3.14;
22  Mu=(4*pi*10^-7)
23  Nipg=((Bag*lg)/Mu);........................//Air
        gap
24  printf("\nampere turns for the air gap=%f",Nipg)
25  NipI=(Np+Nipg);...............................//
        total interpole ampere
26  printf("\nTotal interpole ampere turns=%f",NipI)
27  Nip=(NipI/Ia);
28  printf("\nNumber of turns needed on each commutating
        pole=%f",Nip)
```

Scilab code Exa 2.27 Calculating the efficiency of motor

```
1  //Calculating the efficiency of motor
2  //Chapter 2
3  //Example 2.27
```

```
4  //page 128
5  clear;
6  clc;
7  disp("Example 2.27")
8  N=960;...........................//speed in rpm
9  F=23;............................//effictive load in
       kgf
10 r=45/2;..........................//radius of
       the drum
11 printf("radius of the drum=%fcm",r)
12 pi=3.14;
13 OP=(2*pi*N*F*r*9.81)/(60*100);
14 printf("\noutput power=%fW",OP)
15 Vi=230;..................//motor input in volts
16 Ci=28;.......................//input current in
       amperes
17 IP=(Vi*Ci);
18 printf("\ninput power =%fW",IP)
19 Effi=(OP/IP)*100;
20 printf("\nEfficiency of the motor=%fpercent",Effi)
```

Scilab code Exa 2.29 Calculate the efficiency of machine when running as generator and motor

```
1  //Calculate the efficiency of machine when running
       as generator and motor
2  //Chapter 2
3  //Example 2.29
4  //page 145
5  clear;
6  clc;
7  disp("Example 2.29")
8  I=440;......................//input at no-load in
```

```
        watt
 9  V =220;........................//voltage in volts
10  Ic=I/V;.....................//input current at no−
        load in amperes
11  i=1;...................//input current in amperes
12  A=2;......................//current in amperes
13  C=A-i;....................//armature current at no−
        load in amperes
14  L=I-((((C)^2)*0.5)+(V*C));................//iron,
        friction and windage losses in watt
15  a=40;...................//motor current in amperes
16  OP=(V*a);
17  Ra=0.5;
18  Effi=(OP*100)/(OP+(((a+i)^2)*Ra)+(V*i)+L)
19  printf("Efficiency as a generator when delivering 40
        A at 220V=%fpercent",Effi)
20  Eff=((OP-(((a-i)^2)*Ra)-(V*C)-L)/OP)*100;
21  printf("\nEfficiency as a motor when taking 40A from
        at 220V=%fpercent",Eff)
```

Scilab code Exa 2.30 Calculating the efficiency of the generator at full load and at half load

```
1  //Calculating the efficiency of the generator at
        full load and at half load
2  //Chapter 2
3  //Example 2.30
4  //page 147
5  clear;
6  clc;
7  disp("Example 2.30")
8  V=400;............................//motor in volts
9  Rf=200;...........................//field
```

```
          resistance in ohms
10  If=V/Rf;...........................//current in
          amperes
11  i=5;......... ...................//current at no
          load in amperes
12  IP=V*i;....   ...................//motor input at
          no load
13  Ia=3;..... .......................//aramture
          current in amperes
14  Ra=0.5;.... .....................//armature
          resistance in ohms
15  L=IP-(((Ia)^2)*Ra)-(V*If);....................//
          iron,friction and windage in losses in watt
16  printf("iron,friction and windage in losses=%fW",L)
17  At=50;.......................                 ..//
          armature total current in amperes
18  A=At-2;.........                           ...//
          armature current in amperes
19  Ls=(((A)^2)*Ra)+(V*If)+L;.............         //
          Losses
20  Eff=(((V*At)-Ls)/(V*At))*100;
21  printf("\nEfficiency of full load=%fpercent",Eff)
22  //flux is constant
23  E1=V-(Ia*Ra);...................         //induced
          emf in the armature at no load
24  E2=V-(A*Ra);.......................... //induced
          emf in the armature at full load
25  // since N1/N2=E1/E2
26  percentload=(1-(E2/E1))*100;
27  printf("\nPercentage change in speed from no load to
          full load=%fpercent",percentload)
```

Scilab code Exa 2.31 Calculate the efficiency of machine

```
1  //Calculate the efficiency of machine
2  //Chapter 2
3  //Example 2.31
4  //page 148
5  clear;
6  clc;
7  disp("Example 2.31")
8  Ra=0.5;.................//armature resistance in
       ohms
9  Rf=750;..............//field circuit resistance in
       ohms
10 V=500;......................//voltage in volts
11 If=V/Rf;..........................//current in
       amperes
12 l=3;..........................//line current in
       amperes
13 i=2.33;........................//current in motor
       in amperes
14 I=0.67;.........................//current i amperes
15 L=(V*l)-(((i)^2)*Ra)-(V*I);.......................
       //Iron,friction and windage losses
16 O=20;...............................//generator
17 OP=(O*1000)/V;................//output current of
       the generator under loaded condition in amperes
18 Ia=I+OP;............//output in amperes
19 Effi=(O*1000*100)/((O*1000)+(((Ia)^2)*Ra)+(V*I)+L);
20 printf("efficiency of the machine=%fpercent",Effi)
```

Scilab code Exa 2.32 Calculate the appox efficiency of each machine

```
1  //Calculate the appox. efficiency of each machine
2  //Chapter 2
3  //Example 2.32
```

```
4  //page 149
5  clear ;
6  clc ;
7  disp ("Example 2.32")
8  Ig =25; . . . . . . . . . . . . . . .// current of generator in
       amperes
9  I =30; . . . . . . . . . . . . . . . . . .// current in motor in
       amperes
10  I1 =I - Ig ; . . . . . . . . . . . . .// current in amperes
11  Ra =0.25; . . . . . . . . . . . . . . .// resistance in ohms
12  Gl =(( Ig )^2)* Ra ; . . . . . . . . . . . . . . . .// loss in generator
       in watt
13  M =(( I )^2)* Ra ; . . . . . . . . . . . . . . . . . . .// loss in motor in
       watt
14  T =Gl +M ; . . . . . . . . . . . . . . . . . .// total loss in watt
15  V =100; . . . . . . . . . . . . .// voltage in volts
16  P =V * I1 ; . . . . . . . . . . . . . . .// power supplied from mains in
         watt
17  L =P -T ; . . . . . . . . . . . . . . . . . .// iron , friction and windages
         losses in the two machines in ohms
18  l =L /2; . . . . . . . . . . . . . . . . . . .// iron , friction and
       windages losses in each machines in ohms
19  IP =I * V ; . . . . . . . . . . . . . . . . . . . .// input
20  Eff =(( IP -M - l ) / IP ) *100;
21  printf ("Efficiency of the motor=%fpercent",Eff)
22  OP =Ig * V ; . . . . . . . . . . . . . . . . .// output
23  Effi =(( OP ) / ( OP + Gl + l ) ) *100;
24  printf ("\ nEfficiency of the generator=%fpercent",
       Effi )
```

Scilab code Exa 2.33 Calculate the appox efficiency of each machine

```
1  // Calculate the appox . efficiency of each machine
```

```
2  //Chapter 2
3  //Example 2.33
4  //page 150
5  clear;
6  clc;
7  disp("Example 2.33")
8  V=440;.....................//voltage in volts
9  P=200*1000;................//power in watt
10 Ig=P/V;..............//rated current of each machine
       in amperes
11 //assume losses to be equal
12 I=90;..............//addition currnet supply
13 Effi=sqrt(Ig/(Ig+I))*100;
14 printf("approximate efficiency=%fpercent",Effi)
```

Scilab code Exa 2.34 Calculate the efficiences of the generator at full load

```
1  //Calculate the efficiences of the generator at full
       load
2  //Chapter 2
3  //Example 2.34
4  //page 150
5  clear;
6  clc;
7  disp("Example 2.34")
8  Ig=2000;...........................//output
       current of generator in amperes
9  I=380;.............................//Input current
       from supply mains in amperes
10 Effi=sqrt(Ig/(Ig+I))*100;.................//
       Efficiency of generator assuming equal
       efficiencies of the two machines
11 printf("Efficiences of the generator at full load
```

```
        assuming equal efficiencies=%fpercent",Effi)
12  S=22;.............................//Shunt field
        current of generator
13  G=Ig+S;.........................//Armature current of
        generator in amperes
14  R=0.01;..............................//Resistance
        of the armature circuit of each machine in ohms
15  Gc=((G)^2)*R;...........................//copper loss
        in arrmature circuit of generator in W
16  V=500;...............................//Voltage in
        volts
17  L=V*S;...............................//loss in the
        field circuit of the generator in W
18  T=Ig+I;.............................//total current
        suuply in amperes
19  Sf=17;.................................//
        shunt field current of motor in amperes
20  A=T-Sf;..............................//armature
        current in motor in amperes
21  Lc=((A)^2)*R;.........................//loss in
        armature circuit of motor in amperes
22  Lf=V*Sf;...............................//loss in
        the shunt field circuit of motor in W
23  Tin=V*I;.......................//total input to motor
        and generator in W
24  Ml=Tin-(Gc+L+Lc+Lf);.....................//iron,
        friction and windage loss in both machines in W
25  Me=Ml/2;.....................................//iron,
        friction and windage loss in each machine in W
26  p=1000;.....................//power in kW
27  OP=(Ig*V)/p;........................//full load
        output of the generator
28  Eff=(p*100)/(p+((Gc+L+Me)/1000));
29  printf("\nEfficiency of the generator at full load=
        %fpercent",Eff)
```

Chapter 3

Transformers

Scilab code Exa 3.1 calculating number of turns and primary and secondary currents and value of flux

```scilab
1  //calculating number of turns,primary and secondary
       currents and value of flux
2  //Chapter 3
3  //Example 3.1
4  //page 196
5  clear ;
6  clc ;
7  disp("Example 3.1")
8  kVA=500;                  //rating
9  V1=11000;                  //primary voltage in volts
10 V2=400;                    //secondary voltage in
       volts
11 N2=100;                    //number of turns in
       secondary winding
12 f=50;                      //frequency in hertz
13 N1=(V1*N2)/V2;             //number of turns in
       primary winding
14 printf("number of turns in primary  winding ,N1=
       %dturns",N1)
15 I1=(kVA*1000)/V1;
```

```
16  I2=(kVA*1000)/V2
17  printf("\nprimary current ,I1=%fA",I1)
18  printf("\nsecondary current ,I2=%fA",I2)
19  E1=V1;
20  phi=E1/(4.44*f*N1)
21  printf("\nmaximium flux in the core=%fWb",phi)
```

Scilab code Exa 3.2 calculating number of primary and secondary turns

```
1  //calculating number of primary and secondary turns
2  //Chapter 3
3  //Example 3.2
4  //page 196
5  clear;
6  clc;
7  disp("Example 3.2")
8  V1=6600;                    //primary voltage in volts
9  V2=230;                     //secondary voltage in
       volts
10 f=50;                       //frequency in hertz
11 Bm=1.1;                     //flux density in Wb/m^2
12 A=(25*25*10^(-4));          //area of the core in m^2
13 phi=Bm*A
14 printf("flux=%fWb",phi)
15 E1=V1;
16 E2=V2;
17 N1=E1/(4.44*f*phi);
18 N2=E2/(4.44*f*phi);
19 printf("\nnumber of turns in primary winding ,N1=
       %dturns",N1)
20 printf("\nnumber of turns in secondary winding ,N2=
       %dturns",N2)
```

Scilab code Exa 3.3 calculating induced emf and maximium flux density

```
1  // calculating induced emf and maximium flux density
2  // Chapter 3
3  // Example 3.3
4  // page 197
5  clear;
6  clc;
7  disp("Example 3.3")
8  V1=230;                  // primary voltage in volts
9  f=50;                    // frequency in hertz
10 N1=100;                  // number of primary turns
11 N2=400;                  // number of secondary turns
12 A=250*10^(-4);           // cross section area of
       core in m^2
13 disp("since at no-load E2=V2")
14 E2=(V1*N2)/N1;
15 printf("induced secondary winding,E2=%dV",E2);
16 phi=E2/(4.44*f*N2);
17 Bm=phi/A;
18 printf("\nMaximium flux density in the core=%fWb/m^2
       ",Bm)
```

Scilab code Exa 3.4 calculating induced emf and maximium flux density

```
1  // calculating induced emf and maximium flux density
2  // Chapter 3
3  // Example 3.3
```

```
4  // page 197
5  clear ;
6  clc ;
7  disp (" Example 3.3")
8  kVA =40;                   // rating of the transformer
9  V1 =2000;                  // primary side voltage in
       volts
10 V2 =250;                   // secondary side voltage in
       volts
11 R1 =1.15;                  // primary resistance in ohms
12 R2 =0.0155;                // secondary resistance in
       ohms
13 R = R2 +((( V2 / V1 )^2)* R1 )
14 printf (" Total resistance of the transformer in terms
        of the secondary winding=%fohms" , R )
15 I2 =( kVA *1000)/ V2 ;
16 printf ("\ nFull load secondary current=%dA" , I2 )
17 printf ("\ nTotal resistance load on full load=%fVolts
      " ,( I2 * R ))
18 printf ("\ nTotal copper loss on full load=%fWatts" ,((
      I2 )^2* R ))
```

Scilab code Exa 3.5 Calculating the current and power factor of the primary circuit

```
1 // Calculating the current and power factor of the
      primary circuit
2 // Chapter 3
3 // Example 3.5
4 // page 206
5 clear ;
6 clc ;
7 disp (" Example 3.5")
```

```
 8  I2 =300;.........................//Secondary  current
       in  amperes
 9  N1 =1200;                         //number  of  primary
       turns
10  N2 =300;                          //number  of
       secondary  turns
11  I0 =2.5;                          //load  current  in
       amperes
12  I1 =( I2*N2 )/N1;
13  phi0 = acosd (0.2);
14  phi2 = acosd (0.8);
15  I1c =( I1*cosd ( phi2 ))+( I0*cosd ( phi0 ));
16  I1s =( I1*sind ( phi2 ))+( I0*sind ( phi0 ));
17  I= sqrt ( I1c ^2+ I1s ^2);
18  phi = atand ( I1s/I1c )
19  printf (" primary  power  factor=%fdegrees ",cosd ( phi ));
```

Scilab code Exa 3.6 Calculating the value of primary current

```
 1  // Calculating  the  value  of  primary  current
 2  // Chapter  3
 3  // Example  3.6
 4  // page  207
 5  clear ;
 6  clc ;
 7  disp (" Example  3.6")
 8  I0 =1.5;                  //no−load  current
 9  phi0 = acosd (0.2)
10  I2 =40;                   // secondary  current  in
       amperes
11  phi2 = acosd (0.8)
12  r=3;                      // ratio  of  primary  and
       secondary  turns
```

```
13  I1=I2/r;
14  I1c=(I1*cosd(phi2))+(I0*cosd(phi0));
15  I1s=(I1*sind(phi2))+(I0*sind(phi0));
16  I=sqrt(I1c^2+I1s^2);
17  printf("I1=%fA",I)
```

Scilab code Exa 3.7 Calculating the magnetising current and core loss and flux

```
1  //Calculating  the  magnetising  current ,core  loss  and
       flux
2  //Chapter  3
3  //Example  3.7
4  //page  208
5  clear;
6  clc;
7  disp("Example  3.7")
8  V1=230;                     //voltage  in  volts
9  f=50;                       //frequency  of  supply  in
       hertz
10  N1=250;                    //number  of  primary  turns
11  I0=4.5;                    //no−load  current  in  amperes
12  phi0=acosd(0.25);
13  Im=I0*sind(phi0)
14  printf("magnetising  current ,Im=%fA",Im);
15  Pc=V1*I0*cosd(phi0);
16  printf("\nCore  loss=%dW",Pc)
17  disp("neglecting  I^2R  loss  in  primary  winding  at  no−
       load")
18  E1=V1;
19  phi=E1/(4.44*f*N1);
20  printf("\nMaximium  value  of  flux  in  the  core=%fWb",
       phi)
```

Scilab code Exa 3.8 Calculating the current and power factor of the primary circuit

```
1  //Calculating the current and power factor of the
       primary circuit
2  //Chapter 3
3  //Example 3.8
4  //page 209
5  clear;
6  clc;
7  disp("Example 3.8")
8  I2=30;.........................//Secondary current in
        amperes
9  I0=2;                              //load current in
       amperes
10 V1=660;                                //primary voltage in
       volts
11 V2=220;                               //secondary voltage
       in volts
12 I1=(I2*V2)/V1;
13 phi0=acosd(0.225);
14 phi2=acosd(0.9);
15 I1c=(I1*cosd(phi2))+(I0*cosd(phi0));
16 I1s=(I1*sind(phi2))+(I0*sind(phi0));
17 I=sqrt(I1c^2+I1s^2);
18 phi=atand(I1s/I1c)
19 printf("I1=%fA",I)
20 printf("\nprimary power factor=%fdegrees",cosd(phi))
       ;
```

Scilab code Exa 3.9 Calculating magnetising current and primary current and primary power factor

```
1  //Calculating magnetising current, primary current
       and primary power factor
2  //Chapter 3
3  //Example 3.9
4  //page 210
5  clear;
6  clc;
7  disp("Example 3.9")
8  phi_m=7.5*10^(-3);                      //maximium
       flux
9  f=50;                                 //frequecy in hertz
10 N1=144;                               //number of primary
       turns
11 N2=432;                               //number of
       secondary turns
12 kVA=0.24;                             //rating of
       transformer
13 E1=(4.44*phi_m*f*N1)
14 V1=E1;
15 printf("V1=%dV",V1)
16 I0=(kVA*1000)/V1;
17 phi0=acosd(0.26);
18 Im=I0*sind(phi0);
19 printf("\nIm=%fA",Im);
20 V2=(E1*N2)/N1;
21 printf("\nV2=%fV",V2)
22 disp("At a load of 1.2kVA and power factor of 0.8
       lagging")
23 kVA=1.2;
```

```
24  phi2=acosd(0.8);
25  I2=(kVA*1000)/V2;
26  I=(I2*N2)/N1;
27  I1c=(I*cosd(phi2))+(I0*cosd(phi0));
28  I1s=(I*sind(phi2))+(I0*sind(phi0));
29  I=sqrt(I1c^2+I1s^2);
30  printf("\nI1=%fA",I);
31  phi=acosd(((I*cosd(phi2))+(I0*cosd(phi0)))/I);
32  printf("\nprimary power factor=%flagging",cosd(phi))
```

Scilab code Exa 3.10 Calculating primary current and primary power factor

```
1   //Calculating primary current and primary power
        factor
2   //Chapter 3
3   //Example 3.10
4   //page 211
5   clear;
6   clc;
7   disp("Example 3.10")
8   V1=6600;                      //primary voltage in
        volts
9   V2=240;                       //secondary voltage in
        volts
10  kW1=10;                       //power
11  phi1=acosd(0.8);
12  I2=50;                        //current in amperes
13  kW3=5;                        //power
14  phi2=acosd(0.7)
15  kVA=8;                        //rating
16  phi4=acosd(0.6)
17  I1=(kW1*1000)/(cosd(phi1)*V2);
```

```
18  I3=(kW3*1000)/(1*V2);
19  I4=(kVA*1000)/V2;
20  Ih=((I1*cosd(phi1))+(I2*cosd(phi2))+I3+(I4*cosd(phi4
        )));
21  Iv=((I1*sind(phi1))+(I2*sind(phi2))-(I4*sind(phi4)))
        ;
22  I5=sqrt((Ih^2)+(Iv^2))
23  printf("I5=%dA",I5)
24  Ip=(I5*V2)/V1;
25  printf("\nThe current drawn by the primary from 6600
         Vmains is equal to,Ip=%fA",Ip);
26  phi=atand(Iv/Ih);
27  printf("\n
28  power factor=%flagging",cosd(phi))
```

Scilab code Exa 3.11 Calculating equivalent impedence referred to primary

```
1  //Calculating equivalent impedence referred to
        primary
2  //Chapter 3
3  //Example 3.11
4  //page 212
5  clear;
6  clc;
7  disp("Example 3.11")
8  kVA=100;                    //rating of the tronsfromer
9  N1=400;                      //number of primary turns
10 N2=80;                       //number of secondary
        turns
11 R1=0.3;                       //primary resistance in
        ohms
12 R2=0.01;                       //secondary resistance
```

```
        in  ohms
13  X1 =1.1;                              // primary  leakage
        reactance  in  ohs
14  X2 =0.035;                            // secondary  leakage
        reactance  in  ohms
15  Rr2 =(((N1/N2)^2)*R2)
16  printf ("R2=%f  ohms",Rr2);
17  Xx2 =(((N1/N2)^2)*X2);
18  printf ("\nX2=%f  ohms",Xx2);
19  Ze =sqrt ((R1+Rr2)^2+(X1+Xx2)^2);
20  printf ("\nEquivqlent  impedence=%f",Ze);
```

Scilab code Exa 3.12 Calculating equivalent impedence referred to primary

```
1  // Calculating  equivalent  impedence  referred  to
       primary
2  // Chapter  3
3  // Example  3.12
4  // page  216
5  clear ;
6  clc ;
7  disp ("Example  3.11")
8  f =50;                    // frequency  in  hertz
9  r =6;                  // turns  ratio
10  R1 =0.90;                // primary  resistance  in  ohms
11  R2 =0.03;                // secondary  resistance  in  ohms
12  X1 =5;                   // primary  reactance  in  ohms
13  X2 =0.13;                // secondary  reactance  in  ohms
14  I2 =200;                 // full −load  current
15  Re =(R1+(R2*r^2));
16  printf ("equivalent  resistance  reffered  to  primary ,Re
       =%fohms",Re);
```

```
17  Xe=(X1+(X2*r^2));
18  printf("\nequivalent  reactance  reffered  to  primary ,
        Xe=%fohms",Xe);
19  Ze=sqrt(Re^2+Xe^2);
20  printf("\nequivalent  impedance  reffered  to  primary ,
        Ze=%fohms",Ze);
21  Ii2=r*I2;
22  printf("\nsecondary  current  reffered  to  primary  side
        =%fA",Ii2);
23  printf("\n(a)Voltage  to  be  applied  to  the  high
        voltage  side=%dvolts",(Ii2*Ze));
24  printf("\n(b)Power  factor=%f",(Re/Ze));
```

Scilab code Exa 3.13 Calculate current and power input

```
1  //Calculate  current  and  power  input
2  //Chapter  3
3  //Example  3.13
4  //page  216
5  clear;
6  clc;
7  disp("Example  3.13")
8  R1=0.21;                     //primary  resistance  in
        ohms
9  X1=1;                        //primary  reactance  in
        ohms
10  R2=2.72*10^(-4);            //secondary  resistance
        in  ohms
11  X2=1.3*10^(-3);             //secondary  reactanced
        in  ohms
12  V1=6600;                    //primary  voltage  in
        volts
13  V2=250;                     //secondary  voltage
```

```
       in volts
14 r=V1/V2;                          //turns ratio
15 Re=R1+(r^2*R2);
16 printf("Equivalent resistance referred to primary
       side=%fohms",Re);
17 Xe=X1+(r^2*X2);
18 printf("\nEquivalent reactance referred to primary
       side=%fohms",Xe);
19 Ze=sqrt(Re^2+Xe^2);
20 printf("\nequivalent impedance reffered to primary,
       Ze=%fohms",Ze);
21 V=400;                           //voltage in volts
22 I1=V/Ze;
23 printf("\nI1=%f",I1);
24 printf("\nPower input=%fW",(I1^2*Re));
```

Scilab code Exa 3.14 Calculate current and power input

```
 1 //Calculate current and power input
 2 //Chapter 3
 3 //Example 3.14
 4 //page 217
 5 clear;
 6 clc;
 7 disp("Example 3.14")
 8 N1=90;                  //number of primary turns
 9 N2=180;                 //number of secondary turns
10 R1=0.067;               //primary resistance in ohms
11 R2=0.233;               //secondary resistance in
       ohms
12 printf("Primary winding resistance referred to
       secondary side=%fohms",(R1*(N2/N1)^2))
13 printf("\nsecondary winding resistance referred to
```

48

```
        primary  side=%fohms" ,(R2*(N1/N2)^2))
14  printf("\nTotal  resistance  of  the  transformer
        refferred  to  primary  side=%fohms" ,((R1*(N2/N1)^2)
        +(R2*(N2/N1)^2)))
```

Scilab code Exa 3.15 Calculate percentage regulation

```
 1  //Calculate  percentage  regulation
 2  //Chapter  3
 3  //Example  3.15
 4  //page  217
 5  clear;
 6  clc;
 7  disp("Example  3.15")
 8  kVA=30;                    //rating  of  the  transformer
 9  V1=6000;                    //primary  voltage  in  volts
10  V2=230;                    //secondary  voltage  in  volts
11  R1=10;                    //primary  resistance  in  ohms
12  R2=0.016;                    //secondary  resistance  in
        ohms
13  Xe=23;                        //total  reactance  reffered
         to  the  primary
14  phi=acosd(0.8);                    //lagging
15  Re=(R1+((V1/V2)^2*R2))
16  printf("equivalent  resistance ,Re=%fohms" ,Re)
17  I2dash=(kVA*1000)/V1;
18  V2dash=5847;
19  Reg=((I2dash*((Re*cosd(phi))+(Xe*sind(phi))))*100)/
        V2dash;
20  printf("\npercentage  regulation=%fpercent" ,Reg)
```

Scilab code Exa 3.16 Calculating secondary voltage and voltage regulation

```
1  // Calculating secondary voltage and voltage
       regulation
2  // Chapter 3
3  // Example 3.16
4  // page 218
5  clear ;
6  clc ;
7  disp ("Example 3.16")
8  kVA =10;                    // rating of the transformer
9  V1 =2000;                    // primary voltage in volts
10 V2 =400;                    // secondary voltage in volts
11 R1 =5.5;                    // primary voltage in ohms
12 R2 =0.2;                    // secondary voltage in ohms
13 X1 =12;                     // primary reactance in ohms
14 X2 =0.45;                    // secondary reactance in
       ohms
15 // assuming  (V1/V2)=(N1/N2)
16 Re=R2+(R1*(V2/V1)^2);
17 printf (" equivalent resistance referred to the
       secondary=%fohms", Re );
18 Xe=X2+(X1*(V2/V1)^2);
19 printf (" equivalent reactance referred to the
       secondary=%fohms", Xe );
20 Ze=sqrt (Re^2+Xe^2);
21 printf (" equivalent impedance referred to the
       secondary=%fohms", Ze );
22 phi = acosd (0.8);
23 Vl =374.5;
24 printf ("\ nVoltage across the full load and 0.8 p.f
```

50

```
       lagging=%fV",V1);
25  reg=((V2-V1)*100)/V1;
26  printf("\npercentage  voltage  regulation=%f percent",
        reg);
```

Scilab code Exa 3.17 Calculating regulation

```
 1  //Calculating  regulation
 2  //Chapter  3
 3  //Example  3.17
 4  //page  219
 5  clear;
 6  clc;
 7  disp("Example  3.17")
 8  kVA=80;                 //rating  of  the  transformer
 9  V1=2000;                 //primary  voltage  in  volts
10  V2=200;                  //secondary  voltage  in  volts
11  f=50;                     //frequency  in  hertz
12  Id=8;                     //impedence  drop
13  Rd=4;                     //resistance  drop
14  phi=acosd(0.8)
15  I2Ze=(V2*Id)/100;
16  I2Re=(V2*Rd)/100;
17  I2Xe=sqrt(I2Ze^2-I2Re^2)
18  reg=((I2Re*cosd(phi))+(I2Xe*sind(phi)))*(100/V2)
19  printf("percentage  regulation=%fpercent",reg)
20  pf=I2Xe/sqrt(I2Re^2+I2Xe^2)
21  printf("\nPower  factor  for  zero  regulation=%f(
        leading)",pf)
```

Scilab code Exa 3.19 Calculating the efficiency and voltage regulation

```scilab
1  //Calculating the efficiency and voltage regulation
       //Chapter 3
2  //Example 3.19
3  //page 225
4  clear;
5  clc;
6  disp("Example 3.19")
7  kVA=50;                          //rating of the
       transformer
8  V1=3300;                          //open circuit
       primary voltage
9  Culoss=540;                      //copper loss from
       short circuit test
10 coreloss=460;                    //core loss from open
        circuit test
11 V1sc=124;                        //short circuit
       primary voltage in volts
12 I1sc=15.4;                       //short circuit
       primary current in amperes
13 Psc=540                          //short circuit
       primary power in watts
14 phi=acosd(0.8)
15 effi=(kVA*1000*cosd(phi)*100)/((kVA*1000*cosd(phi))+
       Culoss+coreloss)
16 printf("From the open-circuit test, core-loss=%dW",
       coreloss);
17 printf("\nFrom short circuit test, copper loss=%dW",
       Culoss);
18 printf("\nThe efficiency at full-load and 0.8
       lagging power factor=%f",effi);
```

```
19  Ze=V1sc/I1sc;
20  Re=Psc/I1sc^2;
21  Xe=sqrt(Ze^2-Re^2);
22  V2=3203;
23  phi2=acosd(0.8);
24  phie=acosd(Culoss/(V1sc*I1sc));
25  reg=(V1sc*cosd(phie-phi2)*100)/V1;
26  printf("\nVoltage regulation=%dpercent",reg)
```

Scilab code Exa 3.20 Calculate voltsge to be applied

```
1  //Calculate voltsge to be applied//Chapter 3
2  //Example 3.20
3  //page 226
4  clear;
5  clc;
6  disp("Example 3.20")
7  kVA=100;
8  V1=6600;                    //primary voltage in volts
9  V2=330;                     //secondary voltage in
       volts
10  f=50;                      //frequency in hertz
11  V1sc=100;                  //short circuit
       primary voltage in volts
12  I1sc=10;                   //short circuit
       primary current in amperes
13  Psc=436;                   //short circuit
       primary power in watts
14  Ze=V1sc/I1sc;
15  Re=Psc/I1sc^2;
16  phi=acosd(0.8);
17  Xe=sqrt(Ze^2-Re^2);
18  printf("\nTotal resistance=%fohms",Re);
```

53

```
19  printf("\nTotal impedence=%fohms",Ze)
20  Il=(kVA*1000)/V1;
21  V1dash=(sqrt(((V1*cosd(phi))+(Il*Re))^2+((V1*sind(
        phi))+(Il*Xe))^2));
22  printf("\nfull voltage current,V1=%dV",V1dash)
```

Scilab code Exa 3.21 Calculate circuit constants and efficiency

```
 1  //Calculate circuit constants and efficiency //
        Chapter 3
 2  //Example 3.21
 3  //page 227
 4  clear;
 5  clc;
 6  disp("Example 3.21")
 7  V2=500;                    //secondary voltage in volts
 8  V1=250;                    //primary voltage in short
        circuit test in volts
 9  I0=1;                      //current in short circuit test
         in amperes
10  P=80;                      //core loss in watt
11  Psc=100;                   //power in short circuit
        test in watts
12  Vsc=20;                    //short circuit voltage in
        volts
13  Isc=12;                    //short circuit current in
        amperes
14  phi0=acosd(P/(V1*I0));
15  printf("From open circuit test , cos(phi0)=%f",cos(
        phi0));
16  Ic=I0*cosd(phi0);
17  printf("\nLoss component of no−load current,Ic=%fA",
        Ic)
```

```
18  Im=sqrt(I0^2-Ic^2);
19  printf("\nMagnetising current ,Im=%fA",Im);
20  Rm=V1/Ic;
21  Xm=V1/Im;
22  Re=Psc/(Isc^2);
23  Ze=Vsc/Isc;
24  Xe=sqrt(Ze^2-Re^2);
25  printf("\n\nEquvalent resistance referred to
        secondary=%fohms",Re);
26  printf("\nEquvalent reactance referred to secondary=
        %fohms",Xe);
27  printf("\nEquvalent impedance referred to secondary=
        %fohms",Ze);
28  K=V2/V1;                                //turns ratio
29  printf("\n\nEquvalent resistance referred to primary
        =%fohms",(Re/K^2));
30  printf("\nEquvalent reactance referred to primary=
        %fohms",(Xe/K^2));
31  printf("\nEquvalent impedance referred to primary=
        %fohms",(Ze/K^2));
32  V=500;                                  //output in volts
33  I=10;                                   //output current in
        amperes
34  phi=acosd(0.80);
35  effi=(V*I*cosd(phi)*100)/((V*I*cosd(phi))+P+((I)^2*
        Re));
36  printf("\nEffiency=%fpercent",effi);
```

Scilab code Exa 3.22 Calculate efficiency

```
1  //Calculate efficiency  //Chapter 3
2  //Example 3.22
3  //page 231
```

```
 4 clear;
 5 clc;
 6 disp("Example 3.22")
 7 kVA=200;                    //Rating of the transformer
 8 Pin=3.4;                    //power input to two
       transformer in watt
 9 Pin2=5.2;
10 coreloss=Pin;              //core loss of two
       transformers
11 phi=acosd(0.8);
12 printf("\nCore loss of two transformer=%fkW",Pin)
13 printf("\nCore loss of each transformer=%fkW",(Pin
       /2))
14 printf("\nFull load copper loss of the two
       transformer=%fkW",Pin2)
15 printf("Therefore,full load copper loss of each
       transformer=%fkW",(Pin2/2));
16 effi=(kVA*cosd(phi)*100)/((kVA*cosd(phi))+(Pin/2)+(
       Pin2/2))
17 printf("\nFull load efficiency at 0.8 p.f. lagging=
       %fpercent",effi);
```

Scilab code Exa 3.24 Calculate efficiency of transformer

```
 1 //Calculate efficiency of transformer //Chapter 3
 2 //Example 3.24
 3 //page 233
 4 clear;
 5 clc;
 6 disp("Example 3.24")
 7 kVA=50;                        //rating of the
       transformer
 8 V1=6360;                         //primary voltage
```

```
        rating
 9  V2 =240;                              // secondary
        voltage  rating
10  pf =0.8
11  coreloss =2;                          // core  loss  in  kilo
         watt  from  open  circuit  test
12  Culoss =2;                            // copper  loss  at
        secondary  current  of  175A
13  I =175;                               // current  in
        amperes
14  I2 =( kVA *1000)/V2;
15  printf (" Full  load  secondary  current ,I2=%fA",I2);
16  effi =( kVA *pf *100)/(( kVA *pf )+ coreloss +( Culoss *(I2/I)
        ^2))
17  printf ("\nEfficiency=%fpercent",effi)
```

Scilab code Exa 3.25 Calculate efficiency of transformer

```
 1  // Calculate  efficiency  of  transformer  // Chapter  3
 2  // Example  3.25
 3  // page  234
 4  clear;
 5  clc;
 6  disp (" Example  3.25")
 7  kVA =500;                // rating  of  the  transformer
 8  R1 =0.4;                 // resistance  in  primary
        winding  inohms
 9  R2 =0.001;               // resistance  in  secondary
        winding  in  ohms
10  V1 =6600;                // primary  voltahe  in  volts
11  V2 =400;                 // secondary  voltage  in  volts
12  ironloss =3;             // iron  loss  in  kilowatt
13  pf =0.8;                 // power  factor  lagging
```

```
14  I1=(kVA*1000)/V1;
15  printf(" \nPrimary  winding  current=%fA",I1);
16  I2=(I1*V1)/V2;
17  printf(" \nSecondary  winding  current=%fA",I2);
18  Culoss=((I1^2*R1)+(I2^2*R2));
19  printf(" \nCopper  losses  in  the  two  winding=%fWatts",
        Culoss);
20  effi=(kVA*pf*100)/((kVA*pf)+ironloss+(Culoss/1000));
21  printf(" \nEfficiency  at  0.8  p.f=%fpercent",effi);
```

Scilab code Exa 3.26 Calculate efficiency of transformer

```
1  //Calculate  efficiency  of  transformer  //Chapter  3
2  //Example  3.26
3  //page  234
4  clear;
5  clc;
6  disp("Example  3.26")
7  kVA=400;                        //rating  of  the
        transformer
8  ironloss=2;                    //iron  loss  in  kilowatt
9  pf=0.8;                         //power  factor
10  kW=240;                        //load  in  kilowatt
11  kVA1=kW/pf;
12  disp("Efficiency  is  maximium  when,core-loss=copper-
        loss")
13  coreloss=ironloss;
14  disp("Maximium  efficiency  occurs  at  240kw,0.8  power
        factor,i.e.,  at  300kVA  load")
15  Cl300=coreloss;
16  Cl400=(Cl300*(kVA/kVA1)^2);
17  pf1=0.71;                      //power  factor  for  full  load
18  effi=(kVA*pf1*100)/((kVA*pf1)+coreloss+Cl400);
```

```
19  printf("\nEfficiency  at  full−load  and  071  power
        factor=%dpercent",effi);
20  pf2=1                          //maximium  efficiency
        occurs  at  unity  power  factor
21  MAXeffi=(kVA1*pf2*100)/((kVA1*pf2)+coreloss+Cl300)
22  printf("\nMaximium  efficiency  at  300kVA  and  unity
        power  factor=%fpercent",MAXeffi);
```

Scilab code Exa 3.27 Calculate efficiency of transformer

```
1  //Calculate  efficiency  of  transformer  //Chapter  3
2  //Example  3.27
3  //page  235
4  clear;
5  clc;
6  disp("Example  3.27")
7  kVA=40;                          //rating  of  the
        transformer
8  coreloss=450;                    //core−loss  in  watts
9  Culoss=800;                      //copper  loss  in  watt
10  pf=0.8;                         //power  factor  of  the
        load
11  FLeffi=(kVA*pf*100)/((kVA*pf)+((coreloss+Culoss)
        /1000));
12  printf("Full−load  efficiency=%fpercent",FLeffi);
13  disp("For  maximium  efficiency ,  Core  loss=copper  loss
        ")
14  Culoss2=coreloss;               //for  maximium
        efficiency
15  n=sqrt(Culoss2/Culoss);
16  kVA2=n*kVA;                     //load  for  maximium
        efficiency
17  MAXeffi=(kVA2*pf*100)/((kVA2*pf)+((coreloss+Culoss2)
```

```
    /1000));
18  printf("\nValue of maximium efficiency=%fpercent",
       MAXeffi);
```

Scilab code Exa 3.28 Calculate current in different parts of winding of autotransformer

```
 1  //Calculate efficiency of transformer //Chapter 3
 2  //Example 3.29
 3  //page 236
 4  clear;
 5  clc;
 6  disp("Example 3.29")
 7  kVA=50;                         //rating of the
       transformers
 8  I1=250;                         //primary current in
       amperes
 9  Re=0.006;                       //total resistance
       referred to the primary side
10  ironloss=200;                    //iron loss in watt
11  Culoss=(I1^2*Re);                //copper loss in watt
12  pf=0.8;                          //power factor lagging
13  printf("Full-load copper loss=%fW",Culoss);
14  TL1=((Culoss+ironloss)/1000);
15  printf("\nTotal loss on full load=%fkW",TL1);
16  TL2=((((Culoss*(1/2)^2))+ironloss)/1000)
17  printf("\nTotal loss on half load=%fkW",TL2);
18  effi1=(kVA*pf*100)/((kVA*pf)+TL1);
19  printf("\nEfficiency at full load,0.8 power factor
       lagging=%f percent",effi1)
20  effi2=((kVA/2)*pf*100)/(((kVA/2)*pf)+TL2);
21  printf("\nEfficiency at half load,0.8 power factor
       lagging=%f percent",effi2)
```

Scilab code Exa 3.29 Calculate efficiency of transformer

```
 1  //Calculate efficiency of transformer //Chapter 3
 2  //Example 3.30
 3  //page 237
 4  clear;
 5  clc;
 6  disp("Example 3.30")
 7  kVA=10;                         //rating of the
         transformers
 8  V1=400;                         //primary voltage in
         volts
 9  V2=200;                         //secondary voltage in
         volts
10  f=50;                           //frequency in hertz
11  MAXeffi=0.96;                   //maximium efficiency
12  output1=(kVA*0.75);             //output at 75% of full
         load
13  input1=(output1/MAXeffi);
14  printf("\nInput at 75percent of full load=%fkW",
         input1);
15  TL=input1-output1;
16  printf("\nTotal losses=%fkW",TL);
17  Pi=TL/2;
18  Pc=TL/2;
19  disp("Maximiunm efficiency occurs at 3/4th of full
         load")
20  Pc=Pi/(3/4)^2;
21  printf("\nThus,total losses on full load=%fW",((Pc+
         Pi)*1000));
22  pf=0.8;                         //power factor lagging
23  effi=(kVA*pf*100)/((kVA*pf)+(Pc+Pi));
```

```
24  printf("\nEfficiency on full load. 0.8 power factor
        lagging=%fpercent",effi)
```

Scilab code Exa 3.30 Calculate efficiency of transformer

```
1  //Calculate voltage regulation of transformer //
       Chapter 3
2  //Example 3.31
3  //page 237
4  clear;
5  clc;
6  disp("Example 3.31")
7  kVA=500;                        //rating of the
       transformers
8  V1=3300;                        //primary voltage in
       volts
9  V2=500;                         //secondary voltage in
       volts
10 f=50;                           //frequency in hertz
11 MAXeffi=0.97;
12 x=0.75;                         //fraction of full load
       for maximium efficiency
13 pf1=1;
14 output1=(kVA*x*pf1*1000);
15 printf("Output at maximium efficiency=%dwatts",
       output1);
16 losses=((1/MAXeffi)-1)*output1;
17 printf("\nThus, at maximium efficiency,\n  lossses=
       %fW",losses)
18 Culoss=losses/2;
19 printf("\nCopper losses at 75percent of full load=
       %dW",Culoss);
20 CulossFL=Culoss/x^2;
```

```
21  printf("\nCopper losses at full load=%dW",CulossFL);
22  Re=CulossFL/(kVA*1000);
23  Ze=0.1;                        //equivalent
        impedence per unit
24  Xe=sqrt(Ze^2-Re^2);
25  phi=acosd(0.8);
26  reg=((Re*cosd(phi))+(Xe*sind(phi)))*100;
27  printf("\npercentage regulation=%f percent",reg);
```

Scilab code Exa 3.32 Calculate current in different parts of winding of autotransformer

```
1  //Calculate current in different parts of winding of
        autotransformer//Chapter 3
2  //Example 3.32
3  //page 240
4  clear;
5  clc;
6  disp("Example 3.32")
7  V1=230;                        //primary voltage of auto
        -transformer
8  V2=75;                         //secondary voltage of
        auto-transformer
9  r=(V1/V2);                     //ratio of primary to
        secondary turns
10 I2=200;                        //load current in amperes
11 I1=I2/r;
12 printf("Primary current,I1=%fA",I1);
13 printf("\nLoad current,I1=%fA",I2);
14 printf("\ncirrent flowing through the common portion
        of winding=%fA",(I2-I1));
15 printf("\nEconomy in saving in copper in percentage=
        %fpercent",(100/r));
```

Chapter 4

Three Phase Induction Machines

Scilab code Exa 4.1 to calculate synchronous speed and speed of rotro for slip condition

```
1  // Calculating synchronous speed and speed of a rotor
2  // Chapter 4
3  // Example 4.1
4  // page 288
5  clear ;
6  clc ;
7  disp (" example 4.1 ") ;
8  f =50;    // frequency
9  p =6;  //  number of poles
10 V =400;  // voltage supply
11 S =4;    // percentage slip
12 Ns =(120* f )/ p ;   // synchronous speed
13 printf (" Syhchronous speed , Ns=%d \n " , Ns ) ;
14 Nr =(1 -( S /100) )* Ns ;
15 printf (" speed of rotor with slip 4 percent , Nr is %d
       rpm \n " , Nr ) ;
```

Scilab code Exa 4.2 to find out rotor running at higher slip

```
1  // determining  rotor  running  at  high  slip
2  // Chapter  4
3  // Example  4.2
4  // page  288
5  clear;
6  clc;
7  disp("example  4.2");
8  f=50;      // frequency
9  V=400;  // voltage  supply
10
11 p=2;
12 printf("when  P=2,  Syhchronous  speed,Ns=%d  \n",((120*
      f)/p));
13 p=4;
14 printf("when  P=2,  Syhchronous  speed,Ns=%d  \n",((120*
      f)/p));
15 p=6;
16 printf("when  P=2,  Syhchronous  speed,Ns=%d  \n",((120*
      f)/p));
17 p=8;
18 printf("when  P=2,  Syhchronous  speed,Ns=%d  \n",((120*
      f)/p));
19 disp("for  Nr  to  be  1440  ,  Ns  will  be  1500,  thus  p=4"
      )
20 Ns=1500;Nr1=1440;
21 S1=((Ns-Nr1)/Ns)*100;
22 printf("slip=%d\n",S1);
23 disp("for  Nr  to  be  940  ,  Ns  will  be  1000,  thus  p=6")
24 Ns=1000;Nr2=940;
25 S2=((Ns-Nr2)/Ns)*100;
```

```
26  printf(" slip=%d\n",S2);
27  if S1>S2 then
28      disp("motor running at 1440 rpm is running at
            higher slip")
29  elseif S2>S1
30      disp("motor running at 940 rpm is running at
            higher slip")
```

Scilab code Exa 4.3 calculating slip and number of poles

```
1  //Calculating synchronous speed and speed of a rotor
2  //Chapter 4
3  //Example 4.3
4  //page 289
5  clear;
6  clc;
7  disp("example 4.3");
8  disp("induction motor is to be run at 1440 rpm")
9  P=10;    //poles of alternator
10 N=600;    //speed of alternator
11 f=(P*N)/120    //frequency
12 printf("frequency=%d",f);
13 disp("when P=2");p=2
14 Ns=(120*f)/p;   //synchronous speed
15 printf("Syhchronous speed,Ns=%d \n",Ns);
16 disp("when P=4");p=4;
17 Ns=(120*f)/p;   //synchronous speed
18 printf("Syhchronous speed,Ns=%d \n",Ns);
19 //speed of rotor(1440) is less than synchronous
       speed 1500, therefore P=4
20 disp("speed of rotor(1440) is less than synchronous
       speed 1500, therefore P=4\n")
21 Ns=1500;
```

```
22  Nr =1440;
23  S =(( Ns - Nr )/ Ns )*100
24  printf ("\ nslip  is  %d  percent  and  number  of  poles  is
        4" , S )
```

Scilab code Exa 4.4 Calculate frequency of rotor induced emf

```
1  // Calculate  frequency  of  rotor  induced  emf
2  // Chapter  4
3  // Example  4.4
4  // page  293
5  clear ;
6  clc ;
7  disp (" Example  4.4 ")
8  Nr =1440;                      // rotor  speed  in  rpm
9  f =50;                         // frequency  in  hertz
10  // calculating  Ns  for  values  of  P=2,4,6,8  etc
11  // by  checking  P=4
12  P =4;
13  Ns =(120* f )/ P ;              // Synchronous  speed
14  S =( Ns - Nr )/ Ns ;            // slip
15  Fr = S * f ;                    // rotor  frequency
16  printf (" Rotor  frequency=%dHz " , Fr )
```

Scilab code Exa 4.5 Calculating the speed of running motor and its slip

```
1  // Calculating  the  speed  of  running  motor  and  its
        slip
2  // Chapter  4
```

```
3  //Example 4.5
4  //page 294
5  clear;
6  clc;
7  disp("Example 4.5")
8  f=50;..................//induction motor frequency
        in hertz
9  fr=1.5;.................//rotor frequency in hertz
10 S=fr/f;................//slip
11 P=8;...................//pole
12 Ns=(120*f)/P;
13 printf("synchronous speed=%frpm",Ns)
14 Nr=Ns-(S*Ns);
15 printf("\nmotor running speed=%frpm",Nr)
16 S1=S*100;
17 printf("\nslip percent=%fpercent",S1)
```

Scilab code Exa 4.6 Calculating the speed of rotating magnetic field

```
1  //Calculate rotor current and phase difference
2  //Chapter 4
3  //Example 4.7
4  //page 297
5  clear;
6  clc;
7  disp("Example 4.7")
8  E20=100;              //induced emf in volts
9  R2=0.05;             //rotor resistance in ohms
10 X20=0.1;             //rotor reactance in ohms
11 E20p=E20/sqrt(3);
12 disp("When S=0.04")
13 S=0.04;
14 I2=(S*E20p)/sqrt(R2^2+(S*X20)^2)
```

```
15 printf("I2=%dA",I2);
16 phi2=acosd(R2/(sqrt(R2^2+(S*X20)^2)));
17 printf("\nPhase angle between rotor voltage and
       rotor current=%f degrees",phi2);
18 disp("When S=1")
19 S=1;
20 I2=(S*E20p)/sqrt(R2^2+(S*X20)^2)
21 printf("I2=%dA",I2);
22 phi2=acosd(R2/(sqrt(R2^2+(S*X20)^2)));
23 printf("\nPhase angle between rotor voltage and
       rotor current=%f degrees",phi2);
```

Scilab code Exa 4.7 Calculate rotor current and phase difference

```
1 //Calculate rotor current and phase difference
2 //Chapter 4
3 //Example 4.7
4 //page 297
5 clear;
6 clc;
7 disp("Example 4.7")
8 E20=100;              //induced emf in volts
9 R2=0.05;             //rotor resistance in ohms
10 X20=0.1;            //rotor reactance in ohms
11 E20p=E20/sqrt(3);
12 disp("When S=0.04")
13 S=0.04;
14 I2=(S*E20p)/sqrt(R2^2+(S*X20)^2)
15 printf("I2=%dA",I2);
16 phi2=acosd(R2/(sqrt(R2^2+(S*X20)^2)));
17 printf("\nPhase angle between rotor voltage and
       rotor current=%f degrees",phi2);
18 disp("When S=1")
```

```
19  S=1;
20  I2=(S*E20p)/sqrt(R2^2+(S*X20)^2)
21  printf("I2=%dA",I2);
22  phi2=acosd(R2/(sqrt(R2^2+(S*X20)^2)));
23  printf("\nPhase angle between rotor voltage and
        rotor current=%f degrees",phi2);
```

Scilab code Exa 4.8 Calculating the running speed and frequency of the rotor magnet current

```
1  //Calculating the running speed and frequency of the
        rotor magnet current
2  //Chapter 4
3  //Example 4.8
4  //page 298
5  clear;
6  clc;
7  disp("Example 4.8")
8  f=50;..................//frequency of induction motor
9  P=4;.................//pole
10  Ns=(120*f)/P;
11  S=3;.................//slip percent
12  Nr=Ns-((Ns*S)/100)
13  fr=(S*f)/100;
14  printf("synchronous speed=%frpm",Ns)
15  printf("\nspeed of running motor=%frpm",Nr)
16  printf("\nrotor frequency=%fHz",fr)
```

Scilab code Exa 4.9 Calculating the running speed and frequency of the rotor magnet current

```
1  // Calculating  the  running  speed  and  frequency  of  the
       rotor  magnet  current
2  // Chapter  4
3  // Example  4.9
4  // page  299
5  clear ;
6  clc ;
7  disp (" Example  4.9")
8  fr =2;..............................// frequency  of
       motor  induced  emf  in  hertz
9  f =50;.............................// frequency  of
       induction  motor  in  hertz
10 S =( fr/f ) *100;................// slip  percent
11 P =6;.............................// pole
12 Ns =(120* f )/P ;
13 Nr=Ns -(( Ns*S )/100);
14 printf (" percentage  slip=%fpercent",S)
15 printf ("\ nrotor  speed=%frpm",Nr)
```

Scilab code Exa 4.10 Calculating the frequency of the rotor current

```
1  // Calculating  the  frequency  of  the  rotor  current
2  // Chapter  4
3  // Example  4.10
4  // page  299
5  clear ;
6  clc ;
7  disp (" Example  4.10")
8  P =12;.......................// pole
9  f =50;......................// frequency  of  induction
```

```
             motor  in  hertz
10  Nr=485;........................//induction  motor
        speed  in  rpm
11  Ns=(120*f)/P;
12  S=(Ns-Nr)/Nr;
13  fr=S*f;
14  printf("frequency  of  rotor  current=%fHz",fr)
```

Scilab code Exa 4.11 Calculating the rotor current

```
 1  //Calculating  the  rotor  current
 2  //Chapter  4
 3  //Example  4.11
 4  //page  299
 5  clear;
 6  clc;
 7  disp("Example  4.11")
 8  E20=100;...............................//induced
        emf  of  induction  motor  at  standstill  in  volts
 9  E20p=E20/sqrt(3);........................//induced
        emf  per  phase  in  volts
10  S=0.40;...............................//slip
11  E2=S*E20p;................................//rotor
        induced  emf  at  slip  S  in  volts
12  printf("Rotor  induced  emf  at  a  slip  E2=%fV",E2);
13  R2=0.4;..................................//resistance
         per  phase  in  ohms
14  X20=2.25;............................//standstill
        resistance  per  phase  i  ohms
15  Z2=sqrt((R2)^2+(S*X20)^2);....................//
        rotor  impedence  at  slip  S  in  ohms
16  printf("\nRotor  impedence  at  a  slip  S,  Z2=%fohms",Z2
        )
```

```
17  I=E2/Z2;
18  printf("\nrotor current=%fA",I)
```

Scilab code Exa 4.12 Calculate power developed and efficiency

```
1  //Calculate power developed and efficiency
2  //Chapter 4
3  //Example 4.12
4  //page 308
5  clear;
6  clc;
7  disp("Example 4.12")
8  S=0.03;                   //slip
9  SI=50;                    //stator input in kilowatts
10 SL=2;                     //stator loss in kilowatts
11 RI=SI-SL;                 //rotor input in kilowatts
12 RIL=S*RI;                 //rotor I^2R loss
13 //rotor core loss can be neglected at 3percent slip
14 PDR=RI-RIL;               //power developed by the
       rotor
15 printf("Power developed by the rotor=%fkW",PDR);
16 FWL=1;                    //friction and windage loss in
        kilowatt
17 OP=PDR-FWL;               //output power
18 printf("\nOutput power=%fkW",OP);
19 effi=(OP*100)/SI;
20 printf("\nEfficiency of the motor=%f percent",effi)
```

Scilab code Exa 4.13 Calculating the rotor loss and rotor speed

```
1  // Calculating  the  rotor  loss  and  rotor  speed
2  // Chapter  4
3  // Example  4.13
4  // page  309
5  clear;
6  clc;
7  disp("Example  4.13")
8  f=50;.....................//frequency  of  induction
       motor  in  hertz
9  hp=20;                      //horse  power
10 ph=3;                       //Three  phase  supply
11 P=4;                     //number  of  poles
12 losses=500;               //friction  and  vintage
       losses
13 printf("Output  of  the  motor=%fW",(hp*735.5))
14 Pd=(hp*735.5)+losses;             //power  developed
       in  watt
15 printf("\nPower  developed  by  the  rotor=%dW",Pd);
16 s=0.04;                    //slip
17 rotorloss=(s*Pd)/(1-s);
18 printf("\nRotor  I^2R-loss=%fW",rotorloss);
19 Ns=(120*f)/P;
20 printf("\nNs=%drpm",Ns);
21 Nr=Ns*(1-s);
22 printf("Nr=%drpm",Nr);
```

Scilab code Exa 4.14 Calculating standstill rotor reactance

```
1  // Calculating  standstill  rotor  reactance
2  // Chapter  4
3  // Example  4.14
4  // page  310
5  clear;
```

```
 6 clc;
 7 disp("Example 4.14")
 8 f=50;.....................//frequency of induction
      motor in hertz
 9 P=6;                              //number of poles
10 ph=3;                       //Three phase supply
11 R2=0.1;                     //rotor resistance in
      ohms
12 Ns=(120*f)/P;
13 printf("Syncronous speed,Ns=%drpm",Ns);
14 Nr=940;                      //rotor speed in rpm
15 S=(Ns-Nr)/Ns;
16 printf("\nSlip,S=%f",S);
17 printf("\nstandstill rotor reactance,X20=%fohms",(R2
      /S));
```

Scilab code Exa 4.15 Calculating new full load speed

```
 1 //Calculating new full load speed
 2 //Chapter 4
 3 //Example 4.15
 4 //page 310
 5 clear;
 6 clc;
 7 disp("Example 4.15")
 8 f=50;.....................//frequency of induction
      motor in hertz
 9 P=4;                              //number of poles
10 Nr=1440;                     //rotor speed in rpm
11 R2=0.1;                     //rotor resistance in
      ohms
12 X20=0.6;                    //rotor standstill
      resistance in ohms
```

```
13  Ns=(120*f)/P;
14  printf("Synchronous speed=%drpm",Ns);
15  S1=(Ns-Nr)*(100/Ns);
16  printf("Full-load slip with rotor resistance,R2 i.e.
        S1=%f",S1);
17  disp("on adding extra resistance o.1ohm")
18  //on solving we get S2=0.08
19  S2=0.08;
20  Nr2=Ns*(1-S2);
21  printf("\nNew rotor speed=%drpm",Nr2);
```

Scilab code Exa 4.16 Calculating starting torque

```
1   //Calculating starting torque
2   //Chapter 4
3   //Example 4.16
4   //page 311
5   clear;
6   clc;
7   disp("Example 4.16")
8   f=50;                        //frequency in hertz
9   P=4;                         //number of poles
10  R2=0.04;                        //rotor resistance in
        ohms
11  Ns=(120*f)/P;
12  printf("Syncronous speed=%drpm",Ns);
13  Nr=1200;                      //rotor speed at maximium
        torque in rpm
14  S=(Ns-Nr)/Ns;
15  printf("\nSlip at maximium torque=%f",S);
16  X20=R2/S;
17  //starting torque is developed when S=1
18  //r=(Tst/Tm)
```

```
19  r=(R2/(R2^2+X20^2))*(2*X20);
20  printf("\nTherefore, starting torque is %fpercent of
        the maximium torque",(r*100));
```

Scilab code Exa 4.18 Calculating external resistance

```
1  //Calculating external resistance
2  //Chapter 4
3  //Example 4.18
4  //page 313
5  clear;
6  clc;
7  disp("Example 4.18")
8  P=4;                        //number of poles
9  f=50;                       //frequency in hertz
10 ph=3;                        //three phase supply
11 R2=0.25;                       //rotor resistance in
       ohms
12 Nr=1440;                         //rotor speed in rpm
13 Ns=(120*f)/P;
14 S1=(Ns-Nr)/Ns;
15 printf("S1=%f",S1);
16 Nr2=1200;     //rotor speed when external is added
17 S2=(Ns-Nr2)/Ns;
18 //torque remains constant,we get the relation R2'=R2
       *(S2/S1)
19 R2dash=R2*(S2/S1)
20 printf("\nExtra resistance to be connected in the
        motor circuit=%fohms",(R2dash-R2))
```

Scilab code Exa 4.20 Calculating full load rotor loss and rotor input and output torque

```
1  //Calculating full load rotor loss and rotor input
       and output torque
2  //Chapter 4
3  //Example 4.20
4  //page 311
5  clear;
6  clc;
7  disp("Example 4.20")
8  hp=20;
9  P=4;                          //number of poles
10 f=50;
11 S=0.03;                       //slip
12 MSO=hp*735.5;                 //motor shaft output
13 losses=0.02*MSO              //friction and windage
       loss in watts
14 Pd=MSO+losses;                //power developed
       by the rotor in watts
15 RCL=(S*Pd)/(1-S);            //rotor I^2*R loss
16 printf("rotor copper loss=%fW",RCL);
17 Ri=Pd+RCL                    //rotor iron
       loss is neglected
18 printf("\nRotor input=%fW",Ri);
19 Ns=(120*f)/P;
20 Nr=Ns*(1-S)*(1/60);          //rotor speed
       in rps
21 OT=MSO/(2*3.14*Nr);          //outp[ut
       torque in Nm
22 printf("\noutput torque=%fNm",OT)
```

Scilab code Exa 4.21 Calculating the slip and rotor copper loss and the output horse power and efficiency

```scilab
1  //Calculating the slip ,rotor copper loss ,the output
       horse power and efficiency
2  //Chapter 4
3  //Example 4.21
4  //page 316
5  clear;
6  clc;
7  disp("Example 4.21")
8  f=50;...................//frequency of induction
       motor in hertz
9  P=6;....................//pole
10 Ns=(120*f)/P;
11 Nr=975;.........................//induction motor
       running speed in rpm
12 S=(Ns-Nr)/Ns;
13 printf("the slip=%f",S)
14 Pin=40;...................//power input to stator
       in kW
15 Sl=1;....................//stator losses in kW
16 Rin=Pin-Sl;................//output from stator in
       kW
17 Rc=S*Rin;
18 printf("\nrotor copper losses=%fkW",Rc)
19 l=2;.....................//total losses in kW
20 p=Rin-Rc-l;.................//output power in kw
21 HP=(p*1000)/735.5;
22 printf("\noutput horse output=%fHP",HP)
23 in=40;...........................//input in kW
24 effi=(p/in)*100;
```

```
25 printf("\nefficiency=%fpercent",effi)
```

Scilab code Exa 2.22 Calculate the value of resistance

```
1  //Calculate the value of resistance
2  //Chapter 2
3  //Example 2.22
4  //page 126
5  clear;
6  clc;
7  disp("Example 2.22")
8  V=440;                    //primary voltage in volts
9  Ia=50;                    //armature current in amperes
10 Ra=0.2;                   //armature resistance in ohms
11 N=600;                     //speed in rpm
12 E=V-(Ia*Ra);             //emf induced in volts
      before adding extra resistance
13 //E=K*phi*N=K1*Ia*N
14 K1=E/(Ia*N);
15 //we have the relation T=Kt1*Ia^2, T1=Kt1*Ia1^2
16 //when torque is half, say torque be T1
17 //T1=T/2. r=T/T1
18 r=2;
19 Ia1=sqrt(Ia^2/r);
20 printf("Ia1=%fA",Ia1);
21 //extra resistance R is introduced in the circuit
22 N1=400;
23 E1=(K1*Ia1*N1);
24 R=((V-E1)/Ia1)-Ra;
25 printf("\nvalue of extra resistance added=%fohms",R)
```

81

Scilab code Exa 4.22 Calculating the slip and rotor speed and mechanical power developed and rotor copper loss per phase and resistance per phase

```
1  //Calculating the slip,rotor speed,mechanical power
       developed,rotor copper loss per phase and
       resistance per phase
2  //Chapter 4
3  //Example 4.22
4  //page 316
5  clear;
6  clc;
7  disp("Example 4.22")
8  f=50;............................//frequency of
       induction motor in hertz
9  P=6;...........................//pole
10 Ns=(120*f)/P;
11 printf("synchronous speed=%frpm",Ns)
12 fr=120/60;...........................//rotor
       frequency
13 S=fr/f;
14 printf("\nthe slip=%f",S)
15 Nr=Ns-(Ns*S);
16 printf("\nrotor speed=%frpm",Nr)
17 Rin=80;.......................//rotor input in kW
18 Rc=S*Rin;....................//Rotor copper loss in
       kW
19 Ph=3;..............................//number of
       phases
20 Rcp=(Rc/Ph)*1000;.........................//loss per
       phase in watt
21 p=((Rin-Rc)*1000)/735.5;
22 printf("\nmechanical power developed=%fhp",p)
```

```
23  Ir =60;.........................//rotor  current  in
        amperes
24  R2= Rcp /( Ir )^2;
25  printf ("\nrotor  resistance  per  phase  at  rotor
        current  60A=%fohms" ,R2 )
```

Scilab code Exa 4.23 Calculating additional resistance required

```
1   //Calculating  additional  resistance  required
2   //Chapter  4
3   //Example  4.23
4   //page  320
5   clear ;
6   clc ;
7   disp ("Example  4.23")
8   //  we  know  (Ts/Tm)=((2*a)/(1+a^2))
9   //where   a=(R2/X20)
10  //at  starting  contion  since  Tm=Ts
11  disp ("At  starting  contion  since  Tm=Ts")
12  a=1      //we  obtain  from  the  relations
13  R2 =0.05;                   //circuit  resistance  in
        ohms
14  X2 =0.4;                    //standstill  reactance  in
        ohms
15  r=(a*X2 )-R2;              //r  is  the  extra  that  is
        added  to  the  rotor  circuit
16  printf ("extra  resistance  added ,r=%fohms" ,r )
```

Scilab code Exa 4.24 Calculate speed of motor and maximium torque

```
1  //Calculate speed of motor and maximium torque
2  //Chapter 4
3  //Example 4.24
4  //page 321
5  clear;
6  clc;
7  disp("Example 4.24")
8  V=400;                   //supply voltage in volts
9  f=50;                     //frequency in hertz
10 P=6;                     //number of poles
11 ph=3;                    //three phase supply
12 R2=0.03;                 //rotor resistance in ohms
13 X20=0.4;                 //rptor reactance in ohms
14 Nr=960;                  //full load speed in rpm
15 Ns=(120*f)/P;
16 printf("synchronous speed=%drpm",Ns)
17 S=(Ns-Nr)/Ns;            //corresponding slip
18 //maximium torque Tm occurs at S=(R2/X20)
19 //we get Tm=k/(2*X20)
20 a=R2/X20;
21 //r=Tm/T
22 r=(a^2+S^2)/(2*a*S);
23 Sm=(R2/X20);
24 printf("\nSlip at maximium torque,Sm=%f",Sm);
25 //corresponding speed
26 Nr2=Ns*(1-Sm);
27 printf("\nRotor speed at maximium torque=%drpm",Nr2)
```

Scilab code Exa 4.25 Calculate starting current

```
1  //Calculate starting current
2  //Chapter 4
3  //Example 4.25
```

84

```
4  // page 321
5  clear ;
6  clc ;
7  disp ( " Example 4.25 " )
8  V =400;                    // supply voltage in volts
9  f =50;                      // frequency in hertz
10 P =4;                      // number of poles
11 ph =3;                     // three phase supply
12 S =0.04;
13 If =30;                    // Full load current in
       amperes
14 Isc =6* If ;
15 // let r be the ratio of starting torque nd full load
        torque , r=Ts/Tf
16 r =( Isc / If )^2* S ;
17 // Tf=Tm is produced when voltage is Vm
18 Vm = sqrt ( V ^2/ r );
19 printf ( " \nvoltage at maximium torque=%fvolts " , Vm );
20 Is =6* If *( Vm / V );
21 printf ( " \nFull−load current at 333.3 volts is=%fA " ,
     Is )
```

Scilab code Exa 4.26 Calculate starting line current and starting torque

```
1  // Calculate starting line current and starting
     torque
2  // Chapter 4
3  // Example 4.26
4  // page 330
5  clear ;
6  clc ;
7  disp ( " Example 4.26 " )
8  V =400;                    // supply voltage in volts
```

```
 9   f=50;                              //frequency in hertz
10   Id=75;                             //current taken when delta
        -connected in amperes
11   printf("current taken when delta-connected=%dA",Id);
12   Is=Id/3;                           //current taken when
        star-connected in amperes
13   printf("\ncurrent taken when star-connected=%dA",Is)
        ;
14   //Tfl be the full load torque
15   //r=Ts/Tfl
16   r=1.5;
17   //since voltage becomes (1/sqrt(3)) when star
        connected
18   //torque is directly proportional to square of
        voltage
19   printf("\nStarting torque with winding star
        connected=%f times of Tfl",(r/3));
```

Scilab code Exa 4.28 Calculate starting torque

```
 1   //Calculate starting torque
 2   //Chapter 4
 3   //Example 4.28
 4   //page 333
 5   clear;
 6   clc;
 7   disp("Example 4.28")
 8   ph=3;
 9   //rotor copper loss=slip*rotor input
10   //Tst= starting torque
11   //Tfl=torque at full load
12   //Ist/Ifl=r
13   r=6;
```

```
14  S=0.04
15  printf(" At slip =0.04")
16  printf("\nFor direct-on-line starting, (Tst/Tfl)=%f"
        ,((r^2*S)));
17  //phase current in start is (1/sqrt(3)) times the
        phase current in delta
18
19  printf("\nFor direct-on-line starting, (Tst/Tfl)=%f"
        ,((r/sqrt(3))^2*S));
```

Scilab code Exa 4.29 Calculate full load speed

```
1   //Calculate full load speed
2   //Chapter 4
3   //Example 4.29
4   //page 334
5   clear;
6   clc;
7   disp("Example 4.29")
8   V=400;                    //voltage in volts
9   f=50;                     //frequency in hertz
10  P=4;                      //number of poles
11  //r1=(Ts/Tfl)
12  r1=1.6;
13  //r2=(Tm/Tfl)
14  r2=2;
15  //r3=(Ts/Tm)=(2*a)/(1+a^2)
16  r3=0.8;
17  //on solving , we get a=0.04   ,
18  a=0.04;
19  Sm=0.04;   //slip at maximium torque
20  printf("Slip at maximium torque,Sm=%f",Sm)
21  Ns=(120*f)/P;             //synchronous speed in rpm
```

```
22  Nr=Ns*(1-Sm)                        //rotor  speed  in  rpm
23  //r2=(a^2+Sfl^2)/(2*a*Sfl)
24  Sfl=0.01;
25  Nr2=Ns*(1-Sfl);
26  printf("\nfull  load  speed,Nr=%drpm",Nr2)
```

Scilab code Exa 4.30 Calculate full load rotor loss and rotor input and output torque

```
1  //Calculate  full  load  rotor  loss  and  rotor  input  and
        output  torque
2  //Chapter  4
3  //Example  4.30
4  //page  345
5  clear;
6  clc;
7  disp("Example  4.30")
8  hp=20;                          //power  in  horsepower
9  f=50;                          //frequency  in  hertz
10  P=4;                          //number  of  poles
11  Ns=(120*f)/P;                      //synchronous  speed
12  printf("Synchronous  speed,Ns=%drpm",Ns);
13  S=0.04;                              //slip
14  Nr=Ns*(1-S);
15  OP=hp*735.5;
16  printf("\nOutput  power=%fW",OP);
17  OT=OP/(2*3.14*(Nr/60));
18  printf("\nOutput  torque=%fNm",OT);
19  FL=0.02*OP;                //Friction  and  windage  loss
20  PD=OP+FL;
21  printf("\nPower  developed  by  the  rotor=%fW",PD);
22  //from  relation, (rotor  I^2R-loss=S*Rotor  input)  we
        get  following  relation
```

```
23  RL=(S*PD)/(1-S);
24  printf("\nRotor  I^2R-loss=%fW",RL);
25  RI=RL/S;
26  printf("\nRotor  input=%dW",RI)
```

Scilab code Exa 4.31 Calculate full load rotor loss and rotor input and output torque

```
1  //Calculate  full  load  rotor  loss  and  rotor  input  and
       output  torque
2  //Chapter  4
3  //Example  4.31
4  //page  347
5  clear;
6  clc;
7  disp("Example  4.31")
8  P=4;                    //number  of  poles
9  f=50;                   //frequency  in  hertz
10  V=230;                 //voltage  in  volts
11  hp=5;                  //power  in  horsepower
12  Ib=15;                  //current  in  block  rotor  test
       in  amperes
13  output=hp*735.5;         //output  in  watts
14  //in  block  rotor  test:  power  input=Full=load  I^2R
       losses=735W
15  FL1=735;                          //Full-load
       I^2R  losses
16  printf("Full-load  I^2R  losses=%fW",FL1);
17  Re=FL1/(3*Ib^2);
18  Io=6.3;                  //current  in  no  load
       condition  in  amperes
19  lossNL=(3*(Io)^2*Re);     //I^2R  loss  at  no-load
       condition
```

```
20  printf("\nI^2R loss at no-load=%fW",lossNL);
21  PiNL=275;                        //power input at no-load
22  printf("\nCore loss plus friction and windage loss=
        %dW",(PiNL-lossNL));
23  TL=FLl+(PiNL-lossNL);
24  effi=(output*100)/(output+TL);
25  printf("\nEfficiency=%fpercent",effi)
```

Scilab code Exa 4.32 Calculate full load efficiency

```
1  //Calculate full load efficiency
2  //Chapter 4
3  //Example 4.32
4  //page 347
5  clear;
6  clc;
7  disp("Example 4.32")
8  Vl=415;              //voltage in volts
9  Il=50;                //line current in amperes
10 R1=0.5;                //resistrance of stator
        winding per phase in ohms
11 pf=0.85;              //power factor
12 S=0.04;
13 IFL=(sqrt(3)*Vl*Il*pf)           //input to the motor
        on full load
14 printf("Input to the motor on full load=%dW",IFL);
15 I1=Il/sqrt(3);
16 SLFL=(3*I1^2*R1)               //Stator I^2R loss on
        full load
17 printf("\nStator I^2R loss on full load=%dW",SLFL);
18 //given ratio of stator core loss friction and
        windahe loss be r=(r1:r2)
19 r1=3;
```

90

```
20  r2 =2;
21  TL =1500;                    //total loss
22  SCL =( r1 * TL )/( r1 + r2 );           //stator core loss
23  FWL =( r2 * TL )/( r1 + r2 );           //Friction and
        windage loss
24  SL = SLFL + SCL ;                      //total stator
        loss
25  SI = IFL ;                            //Stator input
26  Pa = SI - SL ;                         //power
        transferred through the air-gap=input to the
        rotor
27  RI = Pa
28  RL = S * RI ;                          //rotor losses
29  TRL = FWL + RL ;                       //total rotor
        losses
30  OP = RI - TRL ;                        //Output power
        at the shaft
31  effi =( OP *100)/ SI ;
32  printf (" \ nEfficiency=%f percent ",effi )
```

Scilab code Exa 4.33 Calculating the rotor current at slip 3 precent and when the rotor develops maximum torque

```
1  //Calculating the rotor current at slip 3 precent
       and when the rotor develops maximum torque
2  //Chapter 4
3  //Example 4.33
4  //page 351
5  clear ;
6  clc ;
7  disp ("Example 4.33")
8  E20 =100;.............................//induced emf
       between slip terminals in volts
```

```
9  E20p=E20/sqrt(3);.......................//induced
      emf per phase in volts
10 printf("induced emf per phase=%fV",E20p)
11 S=3/100;............................//slip
12 R2=0.2;...............................//resistance
      in ohms
13 X20=1;...............................//standstill
      resistance in ohms
14 I2=(S*E20p)/sqrt((R2)^2+(S*X20)^2)
15 printf("\nrotor current at slip 0.03 =%fA per phase"
      ,I2)
16 Sm=R2/X20;
17 I2m=(Sm*E20p)/sqrt((R2)^2+(Sm*X20)^2)
18 printf("\nrotor current when the rotor develops
      maximum torque=%fA per phase",I2m)
```

Scilab code Exa 4.34 Calculating the rotor current at slip 3 precent and when the rotor develops maximum torque

```
1  //Calculating the rotor current at slip 3 precent
      and when the rotor develops maximum torque
2  //Chapter 4
3  //Example 4.34
4  //page 352
5  clear;
6  clc;
7  disp("Example 4.34")
8  E20=120;.....................//induced emf of motor
      at standstill in volts
9  E20p=120/sqrt(3);......................//induced emf
      per phase
10 f=50;...............................//frequency of
      the motor in hertz
```

```
11  R2 =0.2;...............................//Rotor
        Resistance   per  phase
12  X20 =1;................................//
        Standstill  resistance  in  ohms
13  P=4;..................................//pole
14  I =16;........................//
15  S=(I*R2)/sqrt((E20)^2-(I*X20)^2);
16  Ns=(120*f)/P;
17  printf("Synchronous  speed=%frpm",Ns)
18  Nr=Ns-(Ns*S)
19  Sm=R2/X20;
20  Nr=Ns-(Ns*Sm)
21  I2=(Sm*E20p)/sqrt((R2)^2+(Sm*X20)^2)
22  printf("\nrotor  current  at  maximum  torque=%fAper
        Phase",I2)
23  Pi=(3*((I2)^2)*R2)/Sm;
24  printf("\nRotor  input  for  the  three  phase=%fW",Pi)
```

Scilab code Exa 4.35 Calculate the circuit elements

```
1  //Calculate  the  circuit  elements
2  //Chapter  4
3  //Example  4.35
4  //page  356
5  clear;
6  clc;
7  disp("Example  4.35")
8  R1dc=0.01;                    //DC  resistance  in
        ohms
9  V=400;                        //voltage  in  volts
10  r=1.5;                       //ratio  of  ac  to  dc
        resistance
11  R1=r*R1dc;                   //AC  resistance  in  ohms
```

```
12  //at no-load
13  Io=20;                        //no-load current in
        amperes
14  SL=(3*Io^2*R1);          //I^2R loss in the stator
        phases in watts
15  FWL=300;                      //Friction and windage loss
        in watts
16  TL=1200;                       //total losses=no-load
        power input in watts
17  CL=TL-(SL+FWL);          //core loss in watt
18  CLp=CL/sqrt(3);              //core loss per phase
19  Vp=V/sqrt(3);                //voltage per phase
20  Rm=(Vp^3)/CL;                 //motor resistance
21  pf=CL/(Vp*Io);
22  phi0=acosd(pf);
23  Xm=Vp/(Io*sind(phi0));              //motor
        reactance
24  //Under blocked rotor test
25  Vb=100;                       //voltage in volts
26  Isc=45;                        //current in amperes
27  Vbp=100/sqrt(3);            //voltage per phase in
        volts
28  P=2750;                        //power supplied in watts
29  Ze=Vbp/Isc;                          //Motor impedance
        reffered to stator side in ohms
30  Re=P/(3*Isc^2);
31  R2=Re-R1;                   //rotor resistance referred
        to stator side
32  Xe=sqrt(Ze^2-Re^2);
33  //assuming X1=X2
34  X2=Xe/2
35  X1=X2;
36  printf("Thus the elements of the equivalent circuit
        are:");
37  printf("\nRm=%fohms",Rm);
38  printf("\nXm=%fohms",Xm);
39  printf("\n\nR1=%fohms",R1);
40  printf("\nrotor resistance referred to stator side,
```

```
        R2=%fohms",R2);
41  printf("\nequivalent  resistance  referred  to  stator
        side,Re=%fohms",Re);
42
43  printf("\n\nX1=%fohms",X1);
44  printf("\nrotor  reactance  referred  to  stator  side,X2
        =%fohms",X2);
45  printf("\nequivalent  reactance  referred  to  stator
        side,Xe=%fohms",Xe);
```

Chapter 5

Three Phase Synchronous Machines

Scilab code Exa 5.1 To calculate distribution factor

```
1  //caption- for calculating distribution factor
2  //Chapter 5
3  //example 5.1
4  //page 424
5  clear;
6  clc;
7  disp("example 5.1");
8  printf("\n");
9  slots=18;
10 p=2;              //nmber of poles
11 ph=3;             //three phase winding
12 SA=(360/slots);   //slot angle
13 m=slots/(p*ph);   //m=nmber of slots per pole per
       phase
14 printf("number of slots per pole per phase,m=%d\n",m
       );
15 printf("emfs of the oils of each phase will have a
       time-phase difference of  %d degree mechanical \n
       ",SA);
```

```
16  k_d = sind ((m*SA)/2)/(m*sind(SA/2));
17  printf(" distribution  factor=%f",k_d);
```

Scilab code Exa 5.2 To calculate distribution factor

```
1   //chapter 5
2   //example 5.2
3   //page 425
4   clear;
5   clc;
6   disp("example 5.2")
7   printf("\n");
8   slots=36;      //number of slots
9   poles=4;       //number of poles
10  ph=3;          //single layer three phase winding
11  SP=slots/ph; //number of slots per phase
12  printf("number of slots per phase= %d\n",SP);
13  m=SP/poles;   //munber of slots per pole per phase
14  printf("number of slots per pole per phase,m=%d\n",m
       )
15  SA_m=360/slots;           //slot angle mechanical
16  SA_e=(poles/2)*SA_m        //slot angle electrical
17  printf("slot angle= %d degree electrical\n",SA_e)
18  k_d = sind((m*SA_e)/2)/(m*sind(SA_e/2));
19  printf("distribution factor= %f",k_d)
```

Scilab code Exa 5.3 To calculate pitch factor

```
1   //chapter 5
```

```
 2 //example 5.3
 3 //page 426
 4 clear;
 5 clc;
 6 disp("example 5.3");
 7 printf("\n");
 8 slots=48;        //number of slots
 9 poles=4;         //4-pole machine
10 ph=3;            //3-phase machine
11 SA=360/slots;    //slot angle
12 printf("total number of slots= %d\n",slots);
13 printf("slot angle= %f degree mechanical\n",SA);
14 //coil span is 11 slot pitches
15 //12 slots subtend 180degress, short pitched by 1
       slot
16 Bta=1*180/12;
17 k_p=cosd(Bta/2);
18 printf("pitch factor=%f",k_p)
```

Scilab code Exa 5.4 To calculate the rms value of induced EMF

```
 1 //chapter 5
 2 //example 5.4
 3 //page 426
 4 clear;
 5 clc;
 6 disp("example 5.4");
 7 printf("\n");
 8 slots=72;      //number of slots
 9 P=8;           //number of poles
10 ph=3;          //3-phase machine
11 N=750;         //speed of machine in rpm
12 //winding is made with 36 coils having 10 turns
```

```
13  Fp=0.15;         //flux per pole
14  fre=(P*N)/120;
15  NCp=36/ph;       //nmber of coils per phase
16  T=NCp*10;        //number of turns per phase
17  k_p=1;           //since full pitched pitch factor is 1
18  printf("flux per pole=%fWb\n",Fp)
19  printf("number of turns per phase=%d\n",T);
20  printf("pitch factor=%f\n",k_p);
21  m=slots/(P*ph); //slots per pole per phase
22  SA_m=360/slots; //slot angle mechanical
23  SA_e=(P/2)*SA_m;
24  k_d=sind((m*SA_e)/2)/(m*sind(SA_e/2));
25  printf("distribution factor=%f\n",k_d);
26  E=4.44*Fp*fre*T*k_d*k_p;
27  printf("RMS vale of emf induced per phase=%fV\n",E)
```

Scilab code Exa 5.5 Calculating useful flux per pole

```
1  //chapter 5
2  //example 5.5
3  //page 427
4  clear;
5  clc;
6  disp("example 5.5");
7  disp("E(line to line)=  440V");
8  E_l=440;       //line-to-line voltage
9  E_p=E_l/(sqrt(3));
10 N=750;         //speed in rpm
11 fre=50;        //frequency
12 P=(120*fre)/N;
13 printf("P=  %d\n",P);
14 printf("E(per phase)=  %dV\n",E_p);
15 ph=3;          //3-phase machine
```

```
16  m=2;              //number of slots per pole per phase
17  slots=m*P*ph;         //total number of stator slots
18  SA_m=360/slots;       //slot angle mechanical
19  SA_e=(P/2)*SA_m;      //slot angle electrical
20  k_p=1;                //assuming full pitch
21  printf("slot angle= %d degree electrical\n",SA_e);
22  printf("pitch factor=%f\n",k_p);
23  k_d=sind((m*SA_e)/2)/(m*sind(SA_e/2));
24  printf("distribution factor= %f\n\n",k_d);
25  //2 slots per pole per phase
26  NSp=2*P;                //number of slots per phase
27  NTc=4;                  //number of turns per coil
28  T=8*NTc;                //number of turns per phase
29  Fp=E_p/(4.44*fre*T*k_d*k_p);
30  printf("flux per pole= %fWb\n",Fp);
```

Scilab code Exa 5.6 To calculate the frequency and induced EMF

```
1   //chapter 5
2   //example 5.6
3   //page 428
4   clear;
5   clc;
6   disp("example 5.6");
7   printf("\n");
8   slots=144;    //number of slots
9   ph=3;         //3-phase machine
10  P=16;         //number of poles
11  Cp=10;        //number of conducters per slot
12  Fp=0.03;      //flux per pole
13  Ns=375;       //synchronous speed
14  fre=(Ns*P)/120;       //frequency
15  printf("frequency=%d\n\n",fre);
```

```
16  m=slots/(P*ph);        //number of slots per pole per
       phase
17  printf("number of slots per pole per phase,m=  %d\n"
       ,m);
18  SA_m=360/slots;        //slot angle mechanical
19  SA_e=(P/2)*SA_m;       //slot angle electrical
20  k_p=1                  //no short pitching
21  printf("short pitch=  %d\n",k_p);
22  k_d=sind((m*SA_e)/2)/(m*sind(SA_e/2));
23  printf("distribution factor=  %f\n",k_d);
24  T=(slots*10)/(2*ph);
25  printf("number of turns per phase,T=  %d\n",T);
26  E=4.44*Fp*fre*T*k_d*k_p;
27  printf("RMS value of induced emf per phase,E=  %fV\n
       ",E);
28  printf("induced emf across the linesis %fV \n",(sqrt
       (3)*E));
```

Scilab code Exa 5.7 Finding the number of armature conductors

```
1  //chapter 5
2  //example 5.7
3  //page 428
4  clear;
5  clc;
6  disp("example 5.7");
7  printf("\n");
8  slots=90;    //number of slots
9  P=10;        //number of poles
10 ph=3;        //3-phase machine
11 fre=50;      //frequency
12 Fp=0.16;     //flux per pole
13 E_l=11000;   //line voltage
```

```
14  SA_m=360/slots;   //machanical slot angle
15  SA_e=(P/2)*SA_m;  //electrical slot angle
16  m=slots/(ph*P);
17  printf("slot angle=%d degree elecrical\n",SA_e)
18  printf("number of slots per pole per phase,m=%d\n",m
       );
19  k_p=1;                //assuming full pitch
20  printf("pitch factor=%d\n",k_p);
21  k_d=sind((m*SA_e)/2)/(m*sind(SA_e/2));
22  printf("distribution factor=%f\n\n",k_d);
23  E_p=E_l/sqrt(3);
24  T=E_p/(4.44*Fp*fre*k_p*k_d);
25  printf("total number of armature conductors,Z= %d"
       ,(2*T));
```

Scilab code Exa 5.8 To calculate induced EMF per phase

```
1  //chapter 5
2  //example 5.8
3  //page 429
4  clear;
5  clc;
6  disp("example 5.8");
7  disp("P=6    , f=50");
8  P=6;
9  f=50;
10 Sp=12;          //slots per pole
11 Cs=4;           //conductors per slot
12 Fp=1.5;
13 TS=Sp*P
14 printf("total number of slots=%d\n",TS);
15 printf("total number of slots per phase= %d\n", (TS
       /3));
```

```
16 printf("total number of conductors per phase= %d\n",
       ((TS*Cs)/3));
17 T=((TS*Cs)/3)/2;
18 printf("total number of turns per phase=%d\n",T)
19 m=(TS/(P*3));
20 printf("number of slots per pole per phase ,m= %d\n",
       m);
21 SA_m=360/TS;                  //slot angle mechanical
22 SA_e=(P/2)*SA_m;
23 k_d=sind((m*SA_e)/2)/(m*sind(SA_e/2));
24 printf("distribution factor=%f\n\n",k_d);
25 disp("coil pitch is 5/6 of full-pitch");
26 printf("\n");
27 bheta=180-(5/6)*180;        //short pitch angle
28 printf("short pitch angle= %d degrees\n",bheta)
29 k_p=cosd(bheta/2);
30 printf("pitch factor= %f \n",k_p);
31 E=4.44*Fp*f*T*k_d*k_p;
32 printf("induced per phase=  %fV\n",E)
```

Scilab code Exa 5.9 To find the voltage regulation

```
1 //chapter 5
2 //example 5.9
3 //page 439
4 clear;
5 clc;
6 disp("example 5.9");
7 printf("\n");
8 OP=500000;      //output power
9 V_l=3300;       //line voltage
10 I_l=OP/(sqrt(3)*V_l);       //line current
11 printf("line current=  %fA\n",I_l);
```

```
12  //for star connected alternater, line current is
       equal to phase current
13  I_a=I_l;
14  pf=0.8;            //power factor
15  phi=acosd(pf);
16  R_a=0.3;           //synchronous resistance
17  X_s=4;             //synchronous reactance
18  V_p=V_l/sqrt(3);
19  printf("phase voltage=  %fV\n",V_p)
20  E=sqrt((V_p*cosd(phi)+I_a*R_a)^2+(V_p*sind(phi)+I_a*
       X_s)^2);
21  printf("induced emf=  %f V/Phase\n",E  )
22  PR=((E-V_p)*100)/V_p;
23  printf("percentage regulation=  %f percent\n",PR);
```

Scilab code Exa 5.10 To calculate voltage regulation

```
1  //chapter 5
2  //example 5.10
3  //page 440
4  disp("example 5.10")
5  clear;
6  clc;
7  V=2000;
8  V_oc=500;            //open circuit voltage
9  I_sc=100;            //short circuit current
10 I_a=100;
11 R_s=0.8;             //armature resistance
12 Z_s=V_oc/I_sc;       //synchronous impedence
13 printf("Z_s= %d ohm\n",Z_s);
14 X_s=sqrt(Z_s^2-R_s^2);
15 printf("X_s= %f ohm\n",X_s);
16 pf=1;
```

```
17 phi = acosd ( pf ) ;
18 disp ( ” At unity power factor ” ) ;
19 printf ( ” \n ” ) ;
20 E=sqrt (( V * cosd ( phi ) + I_a * R_s ) ^2+( V * sind ( phi ) + I_a * X_s )
      ^2) ;
21 printf ( ” induced emf= %fV \n ” ,E ) ;
22 R =(( E-V ) *100) / V ;
23 printf ( ” regulation= %f percent \n ” ,R ) ;
24 clear pf ;
25 pf =0.71;
26 phi = acosd ( pf ) ;
27 disp ( ” At 0.71 lagging power factor ” ) ;
28 printf ( ” \n ” ) ;
29 E=sqrt (( V * cosd ( phi ) + I_a * R_s ) ^2+( V * sind ( phi ) + I_a * X_s )
      ^2) ;
30 printf ( ” induced emf= %fV \n ” ,E ) ;
31 R =(( E-V ) *100) / V ;
32 printf ( ” regulation= %fpercent \n ” ,R ) ;
33 clear pf ;
34 pf =0.8;
35 phi = acosd ( pf ) ;
36 disp ( ” At 0.8 leading power factor ” ) ;
37 printf ( ” \n ” ) ;
38 E=sqrt (( V * cosd ( phi ) + I_a * R_s ) ^2+( V * sind ( phi ) - I_a * X_s )
      ^2) ;
39 printf ( ” induced emf= %fV \n ” ,E ) ;
40 R =(( E-V ) *100) / V ;
41 printf ( ” regulation=   %fpercent \n ” ,R ) ;
```

Scilab code Exa 5.11 To calculate internal voltage drop

```
1 // chapter 5
2 // example 5.11
```

```
3  //page 441
4  clear;
5  clc;
6  disp("example 5.11");
7  printf("\n");
8  disp("field exitation current=10A");
9  V_oc=900;        //induced emf on open circuit
10 I_sc=150;        //short circuit current
11 Z_s=V_oc/I_sc;  //synchronous impedence
12 printf("synchronous impedence,Z_s= %d ohm\n",Z_s);
13 I_a=60;
14 printf("internal voltage drop when the load current
       is 60amp=  %d V",(I_a*Z_s));
```

Scilab code Exa 5.12 To calculate percentage change in terminal voltage

```
1  //chapter 5
2  //example 5.12
3  //page 441
4  clear;
5  clc;
6  disp("example 5.12");
7  KVA=2000;
8  V=6600;        //rating
9  V_p=6600/sqrt(3);
10 I_a=(KVA*1000)/(sqrt(3)*V);
11 R_a=0.4;       //armature resistance
12 X_s=4.5        //synchronous reactance
13 pf=0.8;
14 phi=acosd(pf);
15 printf("\nV/phase= %dV \n",V_p)
16 E=sqrt((V_p*cosd(phi)+I_a*R_a)^2+(V_p*sind(phi)+I_a*
       X_s)^2)
```

```
17  printf("E=  %f V per phase\n",E);
18  R=((E-V_p)*100)/V_p;
19  printf("percentage change in terminal voltage= %f
        percent",R);
```

Scilab code Exa 5.13 To calculate regulation on full load power factor loading and lagging condition

```
1  //chapter 5
2  //example 5.13
3  //page 442
4  clear;
5  clc;
6  disp("example 5.13");
7  printf("\n");
8  KVA=1200;         //output power
9  printf("output power=%d\n",KVA)
10  V_l=3300;         //line voltage
11  R_a=0.25;         //armature resistance
12  I_l=(KVA*1000)/(sqrt(3)*V_l);     //line current
13  //for star connected I_l=I_a
14  I_a=I_l;
15  V_p=V_l/sqrt(3);
16  printf("V per phase= %dV\n",V_p)
17  //field current of 40A produces short circuit
        current of 200A and open circuit emf 1100
18  v_l=1100;
19  i_s=200;
20  Z_s= v_l/(sqrt(3)*i_s);      //synchronous impedence
21  printf("Synchronous impedance,Zs=%f ohm\n",Z_s)
22  X_s=sqrt(Z_s^2-R_a^2);         //synchronous reactance
23  disp("(a)for 0.8 lagging power facor");
24  pf=0.8;
```

```
25  phi=acosd(pf);
26  E=sqrt((V_p*cosd(phi)+I_a*R_a)^2+(V_p*sind(phi)+I_a*
        X_s)^2)
27  printf("induced emf,E=%f V\n",E);
28  R=((E-V_p)*100)/V_p;
29  printf("regulation=%f percent\n\n",R);
30  clear pf;
31  pf=0.8;
32  phi=acosd(pf);
33  disp("(b)For leading power factor load")
34  E=sqrt((V_p*cosd(phi)+I_a*R_a)^2+(V_p*sind(phi)-I_a*
        X_s)^2)
35  printf("induced emf,E= %f V\n",E);
36  R=((E-V_p)*100)/V_p;
37  printf("regulation=%f percent",R);
```

Scilab code Exa 5.14 To calculate terminal voltage for same excitation and load current at certain power factor leading

```
1   //chapter 5
2   //example 5.14
3   //page 443
4   clear;
5   clc;
6   disp("example 5.14");
7   disp("star connected alternator")
8   printf("\n");
9   KVA=1500;           //rating
10  ph=3;               //3-phase
11  V_l=6600;           //voltage
12  Ra=0.4              //armature resistance
13  Xs=6;               //reactance
14  Ia=(KVA*1000)/(sqrt(3)*V_l);
```

```
15  printf("Full-load current= %d A\n",Ia);
16  V=V_l/sqrt(3);
17  printf("Voltage per phase=%d V\n",V);
18  disp("for 0.8 lagging power facor");
19  pf=0.8;              //power factor
20  phi=acosd(pf);
21  E=sqrt((V*cosd(phi)+Ia*Ra)^2+(V*sind(phi)+Ia*Xs)^2)
22  printf("induced emf=%f V\n\n",E);
23  disp("then at 0.8 leading power factor");
24  Vt=4743; //solved manually
25  printf("termial Voltage, line-to-line=%d V\n",(sqrt
       (3)*Vt))
```

Scilab code Exa 5.15 to find the power factor of alternator B

```
1  //chapter 5
2  //example 5.15
3  //page 450
4  clear;
5  clc;
6  disp("example 5.15");
7  L=8000;       //load
8  La=5000;
9  pf=0.8;
10 phi=acosd(pf);
11 printf("\ntan phi=  %f\n",tand(phi));
12 disp("FOR ALTERNATOR A");
13 pf_a=0.9;
14 phi_a=acosd(pf_a);
15 printf("\ntan phi_a=  %f\n",tand(phi_a));
16 disp("reactive load=active load*tan phi");
17 disp("Active load=8000kW");
18 printf("reactive load=  %d KVAr\n",(8000*tand(phi_a)
```

```
      ));;
19  disp("Active Load A=5000kW\n");
20  printf("Reactive load A=  %dkVAr\n",(5000*tand(phi_a
        )));
21  printf("Active load of B=  %dkW\n",L-La);
22  a=((8000*tand(phi))-(5000*tand(phi_a)))
23  printf("Reactive load of B=  %dkVAr\n",a);
24  B=a/(L-La);
25  phi_b=atand(B);
26  printf("phi_b=  %f\n",phi_b)
27  printf("Power Factor of B=  %f",cosd(phi_b));
```

Scilab code Exa 5.16 To calculate armature curren and power factor

```
1  //chapter 5
2  //example 5.16
3  //page 451
4  clear;
5  clc;
6  disp("example 5.16")
7  V=6600;
8  ph=3;   //3-phase alternators
9  power=10000; //total load
10 disp("Two alternators in parallel connection");
11 pf=0.8;
12 Ia=438;        //armature current
13 Il=(power*1000)/(sqrt(3)*V*pf);   //load current
14 printf("load current=  %fA\n\n",Il);
15 phi=acosd(pf);
16 Ac=(Il*cosd(phi));
17 Rc=(Il*sind(phi));
18 printf("Active component of current=  %fA\n",Ac);
19 printf("Reactive component of current=  %fA\n",Rc);
```

```
20  printf("Current supplied by each alternator=%fA\n",(
       I1/2));
21  printf("Active component of current supplied by each
          alternator=  %fA\n",(Ac/2));
22  printf("Reactive component of current supplied by
       each alternator=  %fA\n\n",(Rc/2));
23  disp("Since steam supply is same,the active
       component remain the same ");
24  RI1=sqrt(Ia^2-(Ac/2)^2);
25  printf("Reactive component of I1= %dA\n",RI1);
26  RI2=(Rc-RI1);
27  printf("reactive component of I2=  %fA\n",RI2);
28  I2=sqrt((Ac/2)^2+(RI2)^2);
29  printf(" I2=  %fA\n",I2);
30  phi_2=atand(RI2/(Ac/2));
31  printf("phi 2=  %f degrees\n",phi_2);
32  printf("cos phi 2= %f",cosd(phi_2));
```

Scilab code Exa 5.17 To determine KVA rating and power facor

```
1  //chapter 5
2  //example 5.17
3  //page 455
4  clear;
5  clc;
6  disp("example 5.17");
7  disp("power factor of existing load is 0.8 lagging")
       ;
8  pf=0.8; //power factor
9  phi=acosd(pf);
10  printf("phi= %d degree\n",phi);
11  L=800;    //load
12  kVAr1=(L*tand(phi));
```

```
13  printf("kVAr1= %d \n",kVAr1);
14  disp("output for the synchronous motor is 200kW");
15  output=200;
16  efficiency=0.9;
17  kW=(output/efficiency);
18  printf("Input to the synchronous motor= %fkW\n",kW);
19  TL=(L+kW); // total load
20  printf("Total load on the system= %fkW\n",TL);
21  disp("overall power factor of the load is to be
        raised to 0.92 lagging");
22  pf=0.92;
23  phi=acosd(pf);
24  kVAr2=(TL*tand(phi))
25  printf("kVAr2=%f\n",kVAr2);
26  kVAr=kVAr1-kVAr2;
27  printf("lagging kVAr of synchronous codenser= %f\n",
        kVAr);
28  printf("leading kVAr supplied by the motor= %f\n",
        kVAr);
29  phi=atand(kVAr/kW);
30  printf("phi= %d degree\n\n",phi);
31  printf("Power factor of the synchronos motor= %f
        leading \n",cosd(phi));
32  printf("KVA rating of the synchronous motor= %f",(kW
        /cosd(phi)));
```