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

July 31, 2019

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

Book Description

Title: 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

Chapter 2

Direct Current Machines

Scilab code Exa 2.4 `Calculating average induced emf`

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

Scilab code Exa 2.5 Calculating useful flux per pole

```
//Calculating the current and power factor of the primary circuit
//Chapter 3
//Example 3.5
//page 206
clear;
clc;
disp("Example 3.5")
I2=300;........................//Secondary current in amperes
N1=1200;                        //number of primary turns
N2=300;                         //number of secondary turns
I0=2.5;                         //load current in amperes
I1=(I2*N2)/N1;
phi0=acosd(0.2);
phi2=acosd(0.8);
I1c=(I1*cosd(phi2))+(I0*cosd(phi0));
I1s=(I1*sind(phi2))+(I0*sind(phi0));
I=sqrt(I1c^2+I1s^2);
phi=atand(I1s/I1c)
printf("primary power factor=%fdegrees",cosd(phi));
```

Scilab code Exa 2.6 Calculating emf generated on open circuit condition

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

Scilab code Exa 2.7 calculate induced emf

```
//calculate induced emf
//Chapter 2
//Example 2.7
//page 97
clear;
clc;
disp("example 2.7")
disp("flux is constant")

E=180;...............//induced emf at 500rpm
N=500;.................//speed in rpm
K1=(E/N)
printf("K1=%f",K1)
```

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

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

```
//calculating the speed and percentage increase in
   flux
//Chapter 2
//Example 2.8
//page 97
clear;
clc;
disp("example 2.8")
disp("assuming constant flux")
E1=220;            //induced emf at N1 speed in volts
N1=750;           // speed
K1=(E1/N1)
E2=250;           //induced emf at speed N2
N2=E2/K1
printf("speed at induced emf of 250V =%d rpm",N2)
disp("when induced emf is 250V and speed 700 rpm")
E3=250;             //induced emf at N3 speed
N3=700;             //speed
ratio=(E3*N1)/(E1*N3)
Pi=(ratio-1)*100
printf("percentage increase in flux is %f percent",
   Pi)
```

Scilab code Exa 2.9 Calculating electromagnetic torque

```
//Calculating electromagnetic torque
//Chapter 2
//Example 2.9
//page 98
clear;
clc;
disp("example 2.9")
E=200          //emf induced
I=15           //armature current
n=1200         //speed in rpm
omega=(2*3.14*n)/60;
printf("omega=%f \n",omega)
T=(E*I)/omega;
printf("electromagnetic torque=%f Nm",T)
```

Scilab code Exa 2.10 calculating the torque developed

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

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

Scilab code Exa 2.11 calculating various parameters of dc motor

```
//calculating various parameters of dc motor
//Chapter 2
//Example 2.11
//page 99
clear;
clc;
disp("Example 2.11")
V=230;                  //armature voltage supply in volts
Ia=12;                  //armature current in amperes
Ra=0.8;                  //armature resistance in ohms
N=100;                     //speed in radian per second
E=(V-(Ia*Ra))
printf("induced emf,E=%fV",E)
Te=(E*Ia)/N
printf("\nthe electromagnetic torque=%fNm",Te)
Pi=V*Ia
printf("\nelectrical input to the armature,Pinput=%dW",Pi)
Pd=Te*N
printf("\nmechanical developed=%fW",Pd)
loss=(Ia^2*Ra)
printf("\narmature copper loss=%fW",loss)
```

Scilab code Exa 2.12 calculating various parameters of dc machine

```
//calculating various parameters of dc motor
//Chapter 2
//Example 2.11
//page 99
clear;
clc;
disp("Example 2.11")
V=230;                //armature voltage supply in volts
Ia=12;                //armature current in amperes
Ra=0.8;                //armature resistance in ohms
N=100;                  //speed in radian per second
E=(V-(Ia*Ra))
printf("induced emf,E=%fV",E)
Te=(E*Ia)/N
printf("\nthe electromagnetic torque=%fNm",Te)
Pi=V*Ia
printf("\nelectrical input to the armature,Pinput=
    %dW",Pi)
Pd=Te*N
printf("\nmechanical developed=%fW",Pd)
loss=(Ia^2*Ra)
printf("\narmature copper loss=%fW",loss)
```

Scilab code Exa 2.13 calculating speed of machine

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

```
//Chapter 2
//Example 2.13
//page 101
clear;
clc;
disp("Example 2.13")
disp("At generator condition")
P=50000;            //power delivered in watts
V=250;              //voltage in volts
Ra=0.02;            //armature resistance in ohms
Rf=50;              //field resistance in ohms
If=V/Rf             //field current in amperes
Ng=400;             //speed in generating condition in rpm
printf("field current,If=%dA",If)
Il=P/V               //load current in amperes
printf("\nLoad current,If=%dA",Il)
Ia=If+Il             //armature current in amperes
printf("\nAramture current,If=%dA\n",Ia)
Eg=(V+(Ia*Ra))
disp("At motor condition")
Ia=(Il-If)
printf("Aramture current,If=%dA",Ia)
Em=(V-(Ia*Ra))
printf("\nEm=%fV",Em)
Nm=(Ng*Em)/Eg
printf("\nSpeed of the motor=%drpm",Nm)
```

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

```
//calculating speed ratio of generator and motor working conditios
//Chapter 2
```

```
//Example 2.14
//page 101
clear;
clc;
disp("Example 2.14")
V=250;                      //voltage supply in volts
Ra=0.12;                    //armature resistance in ohms
Rf=100;                      //field resistance in ohms
Il=80;                      //load current in amperes
If=V/Rf
printf("Field current,If=%f",If)
disp("When machine is generating")
Ia=Il+If
Eg=(V+(Ia*Ra))
printf("\nIa=%fA",Ia)
printf("\nEg=%fV",Eg)
disp("When machine is motoring")
Ia=Il-If
Em=(V-(Ia*Ra))
printf("\nIa=%fA",Ia)
printf("\nEg=%fV",Em)
ratio=Eg/Em
printf("\nRatio of speeds=%f",ratio)
```

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

```
//calculating flux, area of pole shoe and no-load terminal voltage
//Chapter 2
//Example 2.15
//page 102
clear;
clc;
```

```
disp("Example 2.15")
V=550;                //voltage supply in volts
P=16;               //number of poles
N=150;                //speed in rpm
Z=2500;               //number of armature conductors
A=16;
Power=1500000;          //power in watt
Cl=25000;               //full-load copper loss
B=0.9;                  //flux density in the pole
Ia=Power/V
printf("Full load current=%fA",Ia)
Ra=Cl/(Ia^2)
printf("\nRa=%fohms",Ra)
E=V+(Ia*Ra)
printf("\nInduced emf=%fvolts",E)
phi=(E*60*A)/(Z*N*P)
printf("\nflux density=%fWb/m^2",B)
printf("\nflux=%fWb",phi)
area=(phi/B)
printf("\n Area of pole shoe=%fcm^2",(area*10000))
```

Scilab code Exa 2.16 calculate approximate time of commmutation

```
//calculate approximate time of commmutation
//Chapter 2
//Example 2.16
//page 103
clear;
clc;
disp("Example 2.16")
Cd=0.76;                //commutator diameter in metres
Cr=.38;                 //commutator radius in metres
bw=1.5*10^(-2);           //brush width in metres
```

```
N=600;                    //speed in rpm
n=10;                    //speed in rps
V=Cr*(2*3.14*n);
printf("peripheral speed of commutator,V=%fm/sec",V)
    ;
 Tc=bw/V;
 printf("\nTime of commutation=%fseconds",Tc)
```

Scilab code Exa 2.17 calculate resistance

```
//calculate resistance
//Chapter
//Example 2.17
//page 123
clear;
clc;
disp("Example 2.17")
V=240;                //supply voltage in volts
N=800;                //speed in rpm
Ia=2;                  //armeture current in amperes
Ra=0.4;                //armature resistance in ohms
Rf=160;                //field resistance in ohms
Il1=30;                //line current in amperes
E=V-(Ia*Ra);           //induced emf in volts
disp("At no-load")
printf("E=%fV",E)
If=V/Rf;                  //field current in amperes
printf("\nIf=%fA",If)
K1=E/(If*N);
printf("\nK1=%f",K1)
disp("At a load of 30A")
Ia1=(Il1-If);
E1=V-(Ia1*Ra);
```

```
N1=950;                    //speed in rpm
If1=E1/(K1*N1);
printf("If1=%fA\n",If1);
Rr=V/If1;
R=(Rr-Rf);
printf("\nExtra resistance required in the field
    circuit,R=%fohms",R)
```

Scilab code Exa 2.18 calculating resistance required in series

```
//calculating resistance required in series
//Chapter 2
//Example 2.18
//page 124
clear;
clc;
disp("Example 2.18")
V=230;                          //voltage supply in volts
Ia=20;                          //armature current in
    amperes
Ra=0.5;                          //armature resistance in
    ohms
E=V-(Ia*Ra);
printf("E=%dV",E)
disp("when extra resistance is added in the armature
     circuit,the speed is halved")
E2=E/2;
R=((V-E2)/Ia)-Ra;
disp("The load torque is conatant")
printf("extra resistance in the armature circui,R=
    %fohms",R)
disp("The load torque directly proportional to
    square of speed")
```

```
disp("if N is halfed, Iais one-fourthed")
Ia2=Ia/4;
R=((V-E2)/Ia2)-Ra;
printf("extra resistance in the armature circui,R=%fohms",R)
```

Scilab code Exa 2.19 `calculating resistance required in series and also the speedw`

```
//calculating resistance required in series and also the speedwhen torque is halfed
//Chapter 2
//Example 2.19
//page 125
clear;
clc;
disp("Example 2.19")
V=250;                      //voltage supply in volts
Ia=50;                      //armature current in amperes
Ra=0.3;                      //armature resistance in ohms
N=1000;
E=V-(Ia*Ra);
printf("E=%dV",E)
disp("when extra resistance is added in the armature circuit when the speed is 800rpm")
N2=800;
E2=(E*N2)/N;
printf("\nE at 800rpm=%dV",E2)
R=((V-E2)/Ia)-Ra;
printf("\nextra resistance in the armature circui,R=%fohms",R)
disp("if load is halfed,Ia will be halfed")
```

```
Ia2=Ia/2;
E1=V-(Ia2*(Ra+R));
printf("E1=%dV",E1)
N1=(N2*E1)/E2;
printf("\nN1=%frpm",N1)
```

Scilab code Exa 2.20 calculating the speed of the motor

```
//calculating the speed of the motor
//Chapter 2
//Example 2.20
//page 125
clear;
clc;
disp("Example 2.20")
Il=5;                //current in amperes al no-load
V=250;                //voltage in volts
Rf=250;                //field resistance in ohms
If1=V/Rf;                //field current in amperes
Ia1=Il-If1;                //armature current
Ra=0.2;                        //armature resistance in
    ohms
disp("at a load current of 50A")
Il2=50;                        //load current in amperes
//armature reaction weakens by 3percent
If2=0.97;                        //current in amperes
Ia2=Il2-If2;
N1=1000;
E1=(V-(Ia1*Ra));
E2=(V-(Ia2*Ra));
N2=(N1*E2)/(0.97*E1);
printf("N2=%frpm",N2)
```

Scilab code Exa 2.21 Calculate the fullyload speed of the motor

```
//Calculate the fully-load speed of the motor
//Chapter 2
//Example 2.21
//page 126
clear;
clc;
disp("Example 2.21")
P=4;..................//pole
V=500;................//shunt motor in volts
Ia=60;......................//armature current in
    amperes
Ra=0.2;.........................//armature
    resistance in ohms
E=V-(Ia*Ra)-2;
printf("voltage drop across each brush=%fV",E)
phi=0.03;...............................//flux per
     pole in Wb
Z=720;....................................//total
    armature current in volts
A=2;
N=(E*60*A)/(phi*Z*P)
printf("\nfull load speed of the motor=%frpm",N)
```

Scilab code Exa 2.22 Calculate the value of resistance

```
//Calculate the value of resistance
```

```
//Chapter 2
//Example 2.22
//page 126
clear;
clc;
disp("Example 2.22")
V=440;                    //primary voltage in volts
Ia=50;                    //armature current in amperes
Ra=0.2;                    //armature resistance in ohms
N=600;                      //speed in rpm
E=V-(Ia*Ra);               //emf induced in volts
    before adding extra resistance
//E=K*phi*N=K1*Ia*N
K1=E/(Ia*N);
//we have the relation T=Kt1*Ia^2, T1=Kt1*Ia1^2
//when torque is half, say torque be T1
//T1=T/2. r=T/T1
r=2;
Ia1=sqrt(Ia^2/r);
printf("Ia1=%fA",Ia1);
//extra resistance R is introduced in the circuit
N1=400;
E1=(K1*Ia1*N1);
R=((V-E1)/Ia1)-Ra;
printf("\nvalue of extra resistance added=%fohms",R)
```

Scilab code Exa 2.23 Calculate the speed

```
//Calculate the speed
//Chapter 2
//Example 2.23
//page 127
clear;
```

```
clc;
disp("Example 2.23")
V=200;                              //voltage in volts
Ia=20;                              //armature current in amperes
Ra=0.5;                              //armature resistance in ohms
Rse=0.2;                              //field winding resistance in ohms
E=V-(Ia*(Ra+Rse));
printf("In first case,E=%fV",E)
//E=k*phi*N
N=1000;                              //speed in rpm
Kphi=E/N;
//a resistance R is connected in parallel with the series field which is called diverter
disp("when resistace R is added and new conditions")
I=20;                              //total current flowing
//current is equally devided between series field and diverter
Ise2=I/2;
//flux at 10A current is 20percent of flux at 20A current
p=0.70;                    //percentage of flux
Kpih1=p*Kphi;
E1=(V-((Ia*Ra)+(Ise2*Rse)));
printf("Induced emf=%fV",E1)
//new speed is N1
N1=E1/(p*Kphi)
printf("\nN1=%frpm",N1)
```

Scilab code Exa 2.24 Calculate the fullyload speed of the motor

```
//Calculate the fully-load speed of the motor
//Chapter 2
//Example 2.24
//page 128
clear;
clc;
disp("Example 2.24")
V=200;.............................//motor runs in volts
Ia=15;.............................//current taken in amperes
Ra=1;...............................//motor resistance in ohms
E1=V-(Ia*Ra);
printf("resistance when 1ohm=%fV",E1)
R=5;...................................//resistance
E2=V-(Ia*(Ra+R))
printf("\nResistance when 5ohms connected in series=%fV",E2)
N1=800;............................//speed of motor in rpm
N2=N1*(E2/E1);
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`

```
//Calculate the ampere turns for each commutating pole
//Chapter 2
//Example 2.25
//page 135
clear;
```

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

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

```
//Estimating the number of turns needed on each commutating pole
//Chapter 2
//Example 2.26
//page 135
clear;
clc;
disp("Example 2.26")
Bag=0.3;..................................//flux density in the interpole air gap in Wb/m^2
Ia=200000/200;.........................//armature current in amperes
printf("Armature current=%f",Ia)
Z=540;.........................//Number of armature conductors
```

```
Zt=540/2;............................//Number armature winding turns
printf("\nNumber armature winding turns=%f",Zt)
A=6;...............//the winding lap
Ap=Zt/A;......................//Number of armature turns per parallel path
printf("\nNumber of armature turns per parallel path=%f",Ap)
P=6;.............................//pole
Np=((Ia*Ap)/P);
printf("\nNumber of armature ampere turns per pole=%f",Np)
lg=0.01;.............................//inter pole air gap in meters
pi=3.14;
Mu=(4*pi*10^-7)
Nipg=((Bag*lg)/Mu);.........................//Air gap
printf("\nampere turns for the air gap=%f",Nipg)
NipI=(Np+Nipg);...............................//total interpole ampere
printf("\nTotal interpole ampere turns=%f",NipI)
Nip=(NipI/Ia);
printf("\nNumber of turns needed on each commutating pole=%f",Nip)
```

Scilab code Exa 2.27 Calculating the efficiency of motor

```
//Calculating the efficiency of motor
//Chapter 2
//Example 2.27
//page 128
clear;
```

```
clc;
disp("Example 2.27")
N=960;............................//speed in rpm
F=23;.............................//effictive load in kgf
r=45/2;...............................//radius of the drum
printf("radius of the drum=%fcm",r)
pi=3.14;
OP=(2*pi*N*F*r*9.81)/(60*100);
printf("\noutput power=%fW",OP)
Vi=230;..................//motor input in volts
Ci=28;......................//input current in amperes
IP=(Vi*Ci);
printf("\ninput power =%fW",IP)
Effi=(OP/IP)*100;
printf("\nEfficiency of the motor=%fpercent",Effi)
```

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

```
//Calculate the efficiency of machine when running as generator and motor
//Chapter 2
//Example 2.29
//page 145
clear;
clc;
disp("Example 2.29")
I=440;......................//input at no-load in watt
V=220;........................//voltage in volts
Ic=I/V;......................//input current at no-
```

```
load in amperes
i=1;....................//input current in amperes
A=2;.......................//current in amperes
C=A-i;...................//armature current at no-load in amperes
L=I-((((C)^2)*0.5)+(V*C));................//iron , friction and windage losses in watt
a=40;...................//motor current in amperes
OP=(V*a);
Ra=0.5;
Effi=(OP*100)/(OP+(((a+i)^2)*Ra)+(V*i)+L)
printf("Efficiency as a generator when delivering 40 A at 220V=%fpercent",Effi)
Eff=((OP-(((a-i)^2)*Ra)-(V*C)-L)/OP)*100;
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

```
//Calculating the efficiency of the generator at full load and at half load
//Chapter 2
//Example 2.30
//page 147
clear;
clc;
disp("Example 2.30")
V=400;.............................//motor in volts
Rf=200;............................//field resistance in ohms
If=V/Rf;...........................//current in amperes
i=5;......... .....................//current at no
```

```
load in amperes
IP=V*i;.... ........................//motor input at no load
Ia=3;..... ........................//aramture current in amperes
Ra=0.5;.... .......................//armature resistance in ohms
L=IP-(((Ia)^2)*Ra)-(V*If);.....................//iron,friction and windage in losses in watt
printf("iron,friction and windage in losses=%fW",L)
At=50;......................                  ..//armature total current in amperes
A=At-2;..........                          ...//armature current in amperes
Ls=(((A)^2)*Ra)+(V*If)+L;..............        //Losses
Eff=(((V*At)-Ls)/(V*At))*100;
printf("\nEfficiency of full load=%fpercent",Eff)
//flux is constant
E1=V-(Ia*Ra);...................          //induced emf in the armature at no load
E2=V-(A*Ra);............................ //induced emf in the armature at full load
// since N1/N2=E1/E2
percentload=(1-(E2/E1))*100;
printf("\nPercentage change in speed from no load to full load=%fpercent",percentload)
```

Scilab code Exa 2.31 Calculate the efficiency of machine

```
//Calculate the efficiency of machine
//Chapter 2
//Example 2.31
```

```
//page 148
clear;
clc;
disp("Example 2.31")
Ra=0.5;.................//armature resistance in
    ohms
Rf=750;...............//field circuit resistance in
    ohms
V=500;.......................//voltage in volts
If=V/Rf;...........................//current in
    amperes
l=3;...........................//line current in
    amperes
i=2.33;.........................//current in motor
    in amperes
I=0.67;.........................//current i amperes
L=(V*l)-(((i)^2)*Ra)-(V*I);........................
    //Iron, friction and windage losses
O=20;..............................//generator
OP=(O*1000)/V;................//output current of
    the generator under loaded condition in amperes
Ia=I+OP;............//output in amperes
Effi=(O*1000*100)/((O*1000)+(((Ia)^2)*Ra)+(V*I)+L);
printf("efficiency of the machine=%fpercent",Effi)
```

Scilab code Exa 2.32 Calculate the appox efficiency of each machine

```
//Calculate the appox. efficiency of each machine
//Chapter 2
//Example 2.32
//page 149
clear;
clc;
```

```
disp("Example 2.32")
Ig=25;................//current of generator in amperes
I=30;..................//current in motor in amperes
Il=I-Ig;..............//current in amperes
Ra=0.25;................//resistance in ohms
Gl=((Ig)^2)*Ra;................//loss in generator in watt
M=((I)^2)*Ra;....................//loss in motor in watt
T=Gl+M;..................//total loss in watt
V=100;.............//voltage in volts
P=V*Il;...............//power supplied from mains in watt
L=P-T;.................//iron,friction and windages losses in the two machines in ohms
l=L/2;..................//iron,friction and windages losses in each machines in ohms
IP=I*V;..................//input
Eff=((IP-M-l)/IP)*100;
printf("Efficiency of the motor=%fpercent",Eff)
OP=Ig*V;................//output
Effi=((OP)/(OP+Gl+l))*100;
printf("\nEfficiency of the generator=%fpercent",Effi)
```

Scilab code Exa 2.33 Calculate the appox efficiency of each machine

```
//Calculate the appox. efficiency of each machine
//Chapter 2
//Example 2.33
//page 150
```

```
clear;
clc;
disp("Example 2.33")
V=440;.....................//voltage in volts
P=200*1000;...............//power in watt
Ig=P/V;..............//rated current of each machine
    in amperes
//assume losses to be equal
I=90;..............//addition currnet supply
Effi=sqrt(Ig/(Ig+I))*100;
printf("approximate efficiency=%fpercent",Effi)
```

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

```
//Calculate the efficiences of the generator at full
    load
//Chapter 2
//Example 2.34
//page 150
clear;
clc;
disp("Example 2.34")
Ig=2000;.............................//output
   current of generator in amperes
I=380;..............................//Input current
    from supply mains in amperes
Effi=sqrt(Ig/(Ig+I))*100;..................//
   Efficiency of generator assuming equal
   efficiencies of the two machines
printf("Efficiences of the generator at full load
   assuming equal efficiencies=%fpercent",Effi)
S=22;..........................//Shunt field
   current of generator
```

```
G=Ig+S;.........................//Armature current of
    generator in amperes
R=0.01;..............................//Resistance
   of the armature circuit of each machine in ohms
Gc=((G)^2)*R;.........................//copper loss
    in arrmature circuit of generator in W
V=500;...............................//Voltage in
   volts
L=V*S;..............................//loss in the
   field circuit of the generator in W
T=Ig+I;.............................//total current
   suuply in amperes
Sf=17;........................................//
   shunt field current of motor in amperes
A=T-Sf;..............................//armature
   current in motor in amperes
Lc=((A)^2)*R;.......................//loss in
   armature circuit of motor in amperes
Lf=V*Sf;..............................//loss in
   the shunt field circuit of motor in W
Tin=V*I;......................//total input to motor
    and generator in W
Ml=Tin-(Gc+L+Lc+Lf);....................//iron ,
   friction and windage loss in both machines in W
Me=Ml/2;...................................//iron ,
   friction and windage loss in each machine in W
p=1000;.....................//power in kW
OP=(Ig*V)/p;........................//full load
   output of the generator
Eff=(p*100)/(p+((Gc+L+Me)/1000));
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

```
//calculating number of turns,primary and secondary currents and value of flux
//Chapter 3
//Example 3.1
//page 196
clear;
clc;
disp("Example 3.1")
kVA=500;                    //rating
V1=11000;                   //primary voltage in volts
V2=400;                     //secondary voltage in volts
N2=100;                     //number of turns in secondary winding
f=50;                       //frequency in hertz
N1=(V1*N2)/V2;              //number of turns in primary winding
printf("number of turns in primary  winding,N1=%dturns",N1)
I1=(kVA*1000)/V1;
I2=(kVA*1000)/V2
```

```
printf("\nprimary current,I1=%fA",I1)
printf("\nsecondary current,I2=%fA",I2)
E1=V1;
phi=E1/(4.44*f*N1)
printf("\nmaximium flux in the core=%fWb",phi)
```

Scilab code Exa 3.2 calculating number of primary and secondary turns

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

Scilab code Exa 3.3 `calculating induced emf and maximium flux density`

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

Scilab code Exa 3.4 `calculating induced emf and maximium flux density`

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

```
//page 197
clear;
clc;
disp("Example 3.3")
kVA=40;                    //rating of the transformer
V1=2000;                    //primary side voltage in
   volts
V2=250;                     //secondary side voltage in
   volts
R1=1.15;                    //primary resistance in ohms
R2=0.0155;                  //secondary resistance in
   ohms
R=R2+(((V2/V1)^2)*R1)
printf("Total resistance of the transformer in terms
    of the secondary winding=%fohms",R)
I2=(kVA*1000)/V2;
printf("\nFull load secondary current=%dA",I2)
printf("\nTotal resistance load on full load=%fVolts
   ",(I2*R))
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 circui`

```
//Calculating the current and power factor of the
   primary circuit
//Chapter 3
//Example 3.5
//page 206
clear;
clc;
disp("Example 3.5")
I2=300;........................//Secondary current
```

```
   in amperes
N1=1200;                          //number of primary
   turns
N2=300;                           //number of
   secondary turns
I0=2.5;                           //load current in
   amperes
I1=(I2*N2)/N1;
phi0=acosd(0.2);
phi2=acosd(0.8);
I1c=(I1*cosd(phi2))+(I0*cosd(phi0));
I1s=(I1*sind(phi2))+(I0*sind(phi0));
I=sqrt(I1c^2+I1s^2);
phi=atand(I1s/I1c)
printf("primary power factor=%fdegrees",cosd(phi));
```

Scilab code Exa 3.6 Calculating the value of primary current

```
//Calculating the value of primary current
//Chapter 3
//Example 3.6
//page 207
clear;
clc;
disp("Example 3.6")
I0=1.5;                   //no-load current
phi0=acosd(0.2)
I2=40;                    //secondary current in
   amperes
phi2=acosd(0.8)
r=3;                       //ratio of primary and
   secondary turns
I1=I2/r;
```

```
I1c=(I1*cosd(phi2))+(I0*cosd(phi0));
I1s=(I1*sind(phi2))+(I0*sind(phi0));
I=sqrt(I1c^2+I1s^2);
printf("I1=%fA",I)
```

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

```
//Calculating the magnetising current,core loss and
    flux
//Chapter 3
//Example 3.7
//page 208
clear;
clc;
disp("Example 3.7")
V1=230;                    //voltage in volts
f=50;                      //frequency of supply in
    hertz
N1=250;                    //number of primary turns
I0=4.5;                    //no-load current in amperes
phi0=acosd(0.25);
Im=I0*sind(phi0)
printf("magnetising current,Im=%fA",Im);
Pc=V1*I0*cosd(phi0);
printf("\nCore loss=%dW",Pc)
disp("neglecting I^2R loss in primary winding at no-
    load")
E1=V1;
phi=E1/(4.44*f*N1);
printf("\nMaximium value of flux in the core=%fWb",
    phi)
```

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

```
//Calculating the current and power factor of the primary circuit
//Chapter 3
//Example 3.8
//page 209
clear;
clc;
disp("Example 3.8")
I2=30;........................//Secondary current in amperes
I0=2;                         //load current in amperes
V1=660;                        //primary voltage in volts
V2=220;                        //secondary voltage in volts
I1=(I2*V2)/V1;
phi0=acosd(0.225);
phi2=acosd(0.9);
I1c=(I1*cosd(phi2))+(I0*cosd(phi0));
I1s=(I1*sind(phi2))+(I0*sind(phi0));
I=sqrt(I1c^2+I1s^2);
phi=atand(I1s/I1c)
printf("I1=%fA",I)
printf("\nprimary power factor=%fdegrees",cosd(phi));
```

Scilab code Exa 3.9 Calculating magnetising current and primary current and primar

```
//Calculating magnetising current,primary current and primary power factor
//Chapter 3
//Example 3.9
//page 210
clear;
clc;
disp("Example 3.9")
phi_m=7.5*10^(-3);                    //maximium flux
f=50;                                //frequecy in hertz
N1=144;                             //number of primary turns
N2=432;                             //number of secondary turns
kVA=0.24;                            //rating of transformer
E1=(4.44*phi_m*f*N1)
V1=E1;
printf("V1=%dV",V1)
I0=(kVA*1000)/V1;
phi0=acosd(0.26);
Im=I0*sind(phi0);
printf("\nIm=%fA",Im);
V2=(E1*N2)/N1
printf("\nV2=%fV",V2)
disp("At a load of 1.2kVA and power factor of 0.8 lagging")
kVA=1.2;
phi2=acosd(0.8);
```

```
I2=(kVA*1000)/V2;
I=(I2*N2)/N1;
I1c=(I*cosd(phi2))+(I0*cosd(phi0));
I1s=(I*sind(phi2))+(I0*sind(phi0));
I=sqrt(I1c^2+I1s^2);
printf("\nI1=%fA",I);
phi=acosd(((I*cosd(phi2))+(I0*cosd(phi0)))/I);
printf("\nprimary power factor=%flagging",cosd(phi))
```

Scilab code Exa 3.10 Calculating primary current and primary power factor

```
//Calculating primary current and primary power
   factor
//Chapter 3
//Example 3.10
//page 211
clear;
clc;
disp("Example 3.10")
V1=6600;                        //primary voltage in
   volts
V2=240;                         //secondary voltage in
   volts
kW1=10;                         //power
phi1=acosd(0.8);
I2=50;                           //current in amperes
kW3=5;                           //power
phi2=acosd(0.7)
kVA=8;                           //rating
phi4=acosd(0.6)
I1=(kW1*1000)/(cosd(phi1)*V2);
I3=(kW3*1000)/(1*V2);
I4=(kVA*1000)/V2;
```

```
Ih=((I1*cosd(phi1))+(I2*cosd(phi2))+I3+(I4*cosd(phi4)));
Iv=((I1*sind(phi1))+(I2*sind(phi2))-(I4*sind(phi4)));
I5=sqrt((Ih^2)+(Iv^2))
printf("I5=%dA",I5)
Ip=(I5*V2)/V1;
printf("\nThe current drawn by the primary from 6600 Vmains is equal to,Ip=%fA",Ip);
phi=atand(Iv/Ih);
printf("\npower factor=%flagging",cosd(phi))
```

Scilab code Exa 3.11 Calculating equivalent impedence referred to primary

```
//Calculating equivalent impedence referred to primary
//Chapter 3
//Example 3.11
//page 212
clear;
clc;
disp("Example 3.11")
kVA=100;                    //rating of the tronsfromer
N1=400;                       //number of primary turns
N2=80;                        //number of secondary turns
R1=0.3;                        //primary resistance in ohms
R2=0.01;                        //secondary resistance in ohms
X1=1.1;                            //primary leakage reactance in ohs
X2=0.035;                         //secondary leakage
```

```
    reactance in ohms
Rr2=(((N1/N2)^2)*R2)
printf("R2=%f ohms",Rr2);
Xx2=(((N1/N2)^2)*X2);
printf("\nX2=%f ohms",Xx2);
Ze=sqrt((R1+Rr2)^2+(X1+Xx2)^2);
printf("\nEquivqlent impedence=%f",Ze);
```

Scilab code Exa 3.12 Calculating equivalent impedence referred to primary

```
//Calculating equivalent impedence referred to
    primary
//Chapter 3
//Example 3.12
//page 216
clear;
clc;
disp("Example 3.11")
f=50;                    //frequency in hertz
r=6;                    //turns ratio
R1=0.90;                 //primary resistance in ohms
R2=0.03;                  //secondary resistance in ohms
X1=5;                     //primary reactance in ohms
X2=0.13;                  //secondary reactance in ohms
I2=200;                    //full-load current
Re=(R1+(R2*r^2));
printf("equivalent resistance reffered to primary,Re
    =%fohms",Re);
Xe=(X1+(X2*r^2));
printf("\nequivalent reactance reffered to primary,
    Xe=%fohms",Xe);
Ze=sqrt(Re^2+Xe^2);
printf("\nequivalent impedance reffered to primary,
```

```
Ze=%fohms",Ze);
Ii2=r*I2;
printf("\nsecondary current reffered to primary side
=%fA",Ii2);
printf("\n(a)Voltage to be applied to the high
voltage side=%dvolts",(Ii2*Ze));
printf("\n(b)Power factor=%f",(Re/Ze));
```

Scilab code Exa 3.13 Calculate current and power input

```
//Calculate current and power input
//Chapter 3
//Example 3.13
//page 216
clear;
clc;
disp("Example 3.13")
R1=0.21;                    //primary resistance in
    ohms
X1=1;                       //primary reactance in
    ohms
R2=2.72*10^(-4);            //secondary resistance
    in ohms
X2=1.3*10^(-3);             //secondary reactanced
    in ohms
V1=6600;                    //primary voltage in
    volts
V2=250;                     //secondary voltage
    in volts
r=V1/V2;                    //turns ratio
Re=R1+(r^2*R2);
printf("Equivalent resistance referred to primary
    side=%fohms",Re);
```

```
Xe=X1+(r^2*X2);
printf("\nEquivalent reactance referred to primary
    side=%fohms",Xe);
Ze=sqrt(Re^2+Xe^2);
printf("\nequivalent impedance reffered to primary,
    Ze=%fohms",Ze);
V=400;                          //voltage in volts
I1=V/Ze;
printf("\nI1=%f",I1);
printf("\nPower input=%fW",(I1^2*Re));
```

Scilab code Exa 3.14 `Calculate current and power input`

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

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

Scilab code Exa 3.16 Calculating secondary voltage and voltage regulation

```
//Calculating secondary voltage and voltage
   regulation
//Chapter 3
//Example 3.16
//page 218
clear;
clc;
disp("Example 3.16")
kVA=10;                    //rating of the transformer
V1=2000;                    //primary voltage in volts
V2=400;                    //secondary voltage in volts
R1=5.5;                    //primary voltage in ohms
R2=0.2;                    //secondary voltage in ohms
X1=12;                      //primary reactance in ohms
X2=0.45;                    //secondary reactance in
   ohms
//assuming  (V1/V2)=(N1/N2)
Re=R2+(R1*(V2/V1)^2);
printf("equivalent resistance referred to the
   secondary=%fohms",Re);
Xe=X2+(X1*(V2/V1)^2);
printf("equivalent reactance referred to the
   secondary=%fohms",Xe);
Ze=sqrt(Re^2+Xe^2);
printf("equivalent impedance referred to the
   secondary=%fohms",Ze);
phi=acosd(0.8);
Vl=374.5;
printf("\nVoltage across the full load and 0.8 p.f
   lagging=%fV",Vl);
reg=((V2-Vl)*100)/Vl;
printf("\npercentage voltage regulation=%f percent",
   reg);
```

Scilab code Exa 3.17 Calculating regulation

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

Scilab code Exa 3.19 Calculating the efficiency and voltage regulation

```
//Calculating the efficiency and voltage regulation
   //Chapter 3
//Example 3.19
//page 225
clear;
clc;
```

```
disp("Example 3.19")
kVA=50;                              //rating of the transformer
V1=3300;                              //open circuit primary voltage
Culoss=540;                          //copper loss from short circuit test
coreloss=460;                        //core loss from open circuit test
V1sc=124;                            //short circuit primary voltage in volts
I1sc=15.4;                            //short circuit primary current in amperes
Psc=540                               //short circuit primary power in watts
phi=acosd(0.8)
effi=(kVA*1000*cosd(phi)*100)/((kVA*1000*cosd(phi))+Culoss+coreloss)
printf("From the open-circuit test, core-loss=%dW",coreloss);
printf("\nFrom short circuit test, copper loss=%dW",Culoss);
printf("\nThe efficiency at full-load and 0.8 lagging power factor=%f",effi);
Ze=V1sc/I1sc;
Re=Psc/I1sc^2;
Xe=sqrt(Ze^2-Re^2);
V2=3203;
phi2=acosd(0.8);
phie=acosd(Culoss/(V1sc*I1sc));
reg=(V1sc*cosd(phie-phi2)*100)/V1;
printf("\nVoltage regulation=%dpercent",reg)
```

Scilab code Exa 3.20 Calculate voltsge to be applied

```
//Calculate voltsge to be applied//Chapter 3
//Example 3.20
//page 226
clear;
clc;
disp("Example 3.20")
kVA=100;
V1=6600;                    //primary voltage in volts
V2=330;                     //secondary voltage in
   volts
f=50;                       //frequency in hertz
V1sc=100;                         //short circuit
   primary voltage in volts
I1sc=10;                         //short circuit
   primary current in amperes
Psc=436;                           //short circuit
   primary power in watts
Ze=V1sc/I1sc;
Re=Psc/I1sc^2;
phi=acosd(0.8);
Xe=sqrt(Ze^2-Re^2);
printf("\nTotal resistance=%fohms",Re);
printf("\nTotal impedence=%fohms",Ze)
Il=(kVA*1000)/V1;
V1dash=(sqrt(((V1*cosd(phi))+(Il*Re))^2+((V1*sind(
   phi))+(Il*Xe))^2));
printf("\nfull voltage current,V1=%dV",V1dash)
```

Scilab code Exa 3.21 Calculate circuit constants and efficiency

```
//Calculate circuit constants and efficiency //
   Chapter 3
//Example 3.21
//page 227
clear;
clc;
disp("Example 3.21")
V2=500;                    //secondary voltage in volts
V1=250;                  //primary voltage in short
   circuit test in volts
I0=1;                    //current in short circuit test
    in amperes
P=80;                    //core loss in watt
Psc=100;                    //power in short circuit
   test in watts
Vsc=20;                     //short circuit voltage in
   volts
Isc=12;                     //short circuit current in
   amperes
phi0=acosd(P/(V1*I0));
printf("From open circuit test , cos(phi0)=%f",cos(
   phi0));
Ic=I0*cosd(phi0);
printf("\nLoss component of no-load current,Ic=%fA",
   Ic)
Im=sqrt(I0^2-Ic^2);
printf("\nMagnetising current,Im=%fA",Im);
Rm=V1/Ic;
Xm=V1/Im;
Re=Psc/(Isc^2);
Ze=Vsc/Isc;
Xe=sqrt(Ze^2-Re^2);
printf("\n\nEquvalent resistance referred to
   secondary=%fohms",Re);
printf("\nEquvalent reactance referred to secondary=
   %fohms",Xe);
printf("\nEquvalent impedance referred to secondary=
   %fohms",Ze);
```

```
K=V2/V1;                              //turns ratio
printf("\n\nEquvalent resistance referred to primary
   =%fohms",(Re/K^2));
printf("\nEquvalent reactance referred to primary=
   %fohms",(Xe/K^2));
printf("\nEquvalent impedance referred to primary=
   %fohms",(Ze/K^2));
V=500;                         //output in volts
I=10;                          //output current in
   amperes
phi=acosd(0.80);
effi=(V*I*cosd(phi)*100)/((V*I*cosd(phi))+P+((I)^2*
   Re));
printf("\nEffiency=%fpercent",effi);
```

Scilab code Exa 3.22 Calculate efficiency

```
//Calculate efficiency //Chapter 3
//Example 3.22
//page 231
clear;
clc;
disp("Example 3.22")
kVA=200;                 //Rating of the transformer
Pin=3.4;                 //power input to two
   transformer in watt
Pin2=5.2;
coreloss=Pin;            //core loss of two
   transformers
phi=acosd(0.8);
printf("\nCore loss of two transformer=%fkW",Pin)
printf("\nCore loss of each transformer=%fkW",(Pin
   /2))
```

```
printf("\nFull load copper loss of the two
    transformer=%fkW",Pin2)
printf("Therefore, full load copper loss of each
    transformer=%fkW",(Pin2/2));
effi=(kVA*cosd(phi)*100)/((kVA*cosd(phi))+(Pin/2)+(
    Pin2/2))
printf("\nFull load efficiency at 0.8 p.f. lagging=
    %fpercent",effi);
```

Scilab code Exa 3.24 Calculate efficiency of transformer

```
//Calculate efficiency of transformer //Chapter 3
//Example 3.24
//page 233
clear;
clc;
disp("Example 3.24")
kVA=50;                                  //rating of the
    transformer
V1=6360;                                  //primary voltage
    rating
V2=240;                                    //secondary
    voltage rating
pf=0.8
coreloss=2;                              //core loss in kilo
     watt from open circuit test
Culoss=2;                                //copper loss at
    secondary current of 175A
I=175;                                    //current in
    amperes
I2=(kVA*1000)/V2;
printf("Full load secondary current,I2=%fA",I2);
effi=(kVA*pf*100)/((kVA*pf)+coreloss+(Culoss*(I2/I)
```

```
^2))
printf("\nEfficiency=%fpercent",effi)
```

Scilab code Exa 3.25 Calculate efficiency of transformer

```
//Calculate efficiency of transformer //Chapter 3
//Example 3.25
//page 234
clear;
clc;
disp("Example 3.25")
kVA=500;                    //rating of the transformer
R1=0.4;                    //resistance in primary winding inohms
R2=0.001;                  //resistance in secondary winding in ohms
V1=6600;                    //primary voltahe in volts
V2=400;                     //secondary voltage in volts
ironloss=3;                  //iron loss in kilowatt
pf=0.8;                      //power factor lagging
I1=(kVA*1000)/V1;
printf("\nPrimary winding current=%fA",I1);
I2=(I1*V1)/V2;
printf("\nSecondary winding current=%fA",I2);
Culoss=((I1^2*R1)+(I2^2*R2));
printf("\nCopper losses in the two winding=%fWatts",Culoss);
effi=(kVA*pf*100)/((kVA*pf)+ironloss+(Culoss/1000));
printf("\nEfficiency at 0.8 p.f=%fpercent",effi);
```

Scilab code Exa 3.26 `Calculate efficiency of transformer`

```
//Calculate efficiency of transformer //Chapter 3
//Example 3.26
//page 234
clear;
clc;
disp("Example 3.26")
kVA=400;                              //rating of the transformer
ironloss=2;                          //iron loss in kilowatt
pf=0.8;                               //power factor
kW=240;                              //load in kilowatt
kVA1=kW/pf;
disp("Efficiency is maximium when,core-loss=copper-loss")
coreloss=ironloss;
disp("Maximium efficiency occurs at 240kw,0.8 power factor,i.e., at 300kVA load")
Cl300=coreloss;
Cl400=(Cl300*(kVA/kVA1)^2);
pf1=0.71;                   //power factor for full load
effi=(kVA*pf1*100)/((kVA*pf1)+coreloss+Cl400);
printf("\nEfficiency at full-load and 071 power factor=%dpercent",effi);
pf2=1                            //maximium efficiency occurs at unity power factor
MAXeffi=(kVA1*pf2*100)/((kVA1*pf2)+coreloss+Cl300)
printf("\nMaximium efficiency at 300kVA and unity power factor=%fpercent",MAXeffi);
```

Scilab code Exa 3.27 Calculate efficiency of transformer

```
//Calculate efficiency of transformer //Chapter 3
//Example 3.27
//page 235
clear;
clc;
disp("Example 3.27")
kVA=40;                        //rating of the transformer
coreloss=450;                  //core-loss in watts
Culoss=800;                    //copper loss in watt
pf=0.8;                        //power factor of the load
FLeffi=(kVA*pf*100)/((kVA*pf)+((coreloss+Culoss)/1000));
printf("Full-load efficiency=%fpercent",FLeffi);
disp("For maximium efficiency , Core loss=copper loss")
Culoss2=coreloss;              //for maximium efficiency
n=sqrt(Culoss2/Culoss);
kVA2=n*kVA;                    //load for maximium efficiency
MAXeffi=(kVA2*pf*100)/((kVA2*pf)+((coreloss+Culoss2)/1000));
printf("\nValue of maximium efficiency=%fpercent",MAXeffi);
```

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

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

Scilab code Exa 3.29 Calculate efficiency of transformer

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

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

Scilab code Exa 3.30 Calculate efficiency of transformer

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

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

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

```
//Calculate current in different parts of winding of autotransformer//Chapter 3
//Example 3.32
//page 240
clear;
clc;
disp("Example 3.32")
V1=230;                        //primary voltage of auto-transformer
V2=75;                         //secondary voltage of auto-transformer
r=(V1/V2);                      //ratio of primary to secondary turns
I2=200;                        //load current in amperes
I1=I2/r;
printf("Primary current,I1=%fA",I1);
printf("\nLoad current,I1=%fA",I2);
printf("\ncirrent flowing through the common portion of winding=%fA",(I2-I1));
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 con

```
//Calculating synchronous speed and speed of a rotor
//Chapter 4
//Example 4.1
//page 288
clear;
clc;
disp("example 4.1");
f=50;    //frequency
p=6; // number of poles
V=400; //voltage supply
S=4;   //percentage slip
Ns=(120*f)/p;  //synchronous speed
printf("Syhchronous speed,Ns=%d \n",Ns);
Nr=(1-(S/100))*Ns;
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`

```
//determining rotor running at high slip
//Chapter 4
//Example 4.2
//page 288
clear;
clc;
disp("example 4.2");
f=50;   //frequency
V=400; //voltage supply

p=2;
printf("when P=2, Syhchronous speed,Ns=%d \n",((120*f)/p));
p=4;
printf("when P=2, Syhchronous speed,Ns=%d \n",((120*f)/p));
p=6;
printf("when P=2, Syhchronous speed,Ns=%d \n",((120*f)/p));
p=8;
printf("when P=2, Syhchronous speed,Ns=%d \n",((120*f)/p));
disp("for Nr to be 1440 , Ns will be 1500, thus p=4")
Ns=1500;Nr1=1440;
S1=((Ns-Nr1)/Ns)*100;
printf("slip=%d\n",S1);
disp("for Nr to be 940 , Ns will be 1000, thus p=6")
Ns=1000;Nr2=940;
S2=((Ns-Nr2)/Ns)*100;
```

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

Scilab code Exa 4.3 calculating slip and number of poles

```
//Calculating synchronous speed and speed of a rotor
//Chapter 4
//Example 4.3
//page 289
clear;
clc;
disp("example 4.3");
disp("induction motor is to be run at 1440 rpm")
P=10;    //poles of alternator
N=600;    //speed of alternator
f=(P*N)/120    //frequency
printf("frequency=%d",f);
disp("when P=2");p=2
Ns=(120*f)/p;  //synchronous speed
printf("Syhchronous speed,Ns=%d \n",Ns);
disp("when P=4");p=4;
Ns=(120*f)/p;  //synchronous speed
printf("Syhchronous speed,Ns=%d \n",Ns);
//speed of rotor(1440) is less than synchronous
   speed 1500, therefore P=4
disp("speed of rotor(1440) is less than synchronous
   speed 1500, therefore P=4\n")
Ns=1500;
```

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

Scilab code Exa 4.4 `Calculate frequency of rotor induced emf`

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

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

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

```
//Example 4.5
//page 294
clear;
clc;
disp("Example 4.5")
f=50;...................//induction motor frequency in hertz
fr=1.5;.................//rotor frequency in hertz
S=fr/f;................//slip
P=8;...................//pole
Ns=(120*f)/P;
printf("synchronous speed=%frpm",Ns)
Nr=Ns-(S*Ns);
printf("\nmotor running speed=%frpm",Nr)
S1=S*100;
printf("\nslip percent=%fpercent",S1)
```

Scilab code Exa 4.6 `Calculating the speed of rotating magnetic field`

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

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

Scilab code Exa 4.7 Calculate rotor current and phase difference

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

```
S=1;
I2=(S*E20p)/sqrt(R2^2+(S*X20)^2)
printf("I2=%dA",I2);
phi2=acosd(R2/(sqrt(R2^2+(S*X20)^2)));
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 magne

```
//Calculating the running speed and frequency of the
    rotor magnet current
//Chapter 4
//Example 4.8
//page 298
clear;
clc;
disp("Example 4.8")
f=50;..................//frequency of induction motor
P=4;..................//pole
Ns=(120*f)/P;
S=3;..................//slip percent
Nr=Ns-((Ns*S)/100)
fr=(S*f)/100;
printf("synchronous speed=%frpm",Ns)
printf("\nspeed of running motor=%frpm",Nr)
printf("\nrotor frequency=%fHz",fr)
```

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

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

Scilab code Exa 4.10 Calculating the frequency of the rotor current

```
//Calculating the frequency of the rotor current
//Chapter 4
//Example 4.10
//page 299
clear;
clc;
disp("Example 4.10")
P=12;.......................//pole
f=50;.......................//frequency of induction
    motor in hertz
Nr=485;.....................//induction motor
   speed in rpm
```

```
Ns=(120*f)/P;
S=(Ns-Nr)/Nr;
fr=S*f;
printf("frequency of rotor current=%fHz",fr)
```

Scilab code Exa 4.11 `Calculating the rotor current`

```
//Calculating the rotor current
//Chapter 4
//Example 4.11
//page 299
clear;
clc;
disp("Example 4.11")
E20=100;.................................//induced emf of induction motor at standstill in volts
E20p=E20/sqrt(3);........................//induced emf per phase in volts
S=0.40;..................................//slip
E2=S*E20p;...............................//rotor induced emf at slip S in volts
printf("Rotor induced emf at a slip E2=%fV",E2);
R2=0.4;..................................//resistance per phase in ohms
X20=2.25;...............................//standstill resistance per phase i ohms
Z2=sqrt((R2)^2+(S*X20)^2);....................//rotor impedence at slip S in ohms
printf("\nRotor impedence at a slip S, Z2=%fohms",Z2)
I=E2/Z2;
printf("\nrotor current=%fA",I)
```

Scilab code Exa 4.12 Calculate power developed and efficiency

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

Scilab code Exa 4.13 Calculating the rotor loss and rotor speed

```
//Calculating the rotor loss and rotor speed
```

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

Scilab code Exa 4.14 Calculating standstill rotor reactance

```
//Calculating standstill rotor reactance
//Chapter 4
//Example 4.14
//page 310
clear;
clc;
```

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

Scilab code Exa 4.15 `Calculating new full load speed`

```
//Calculating new full load speed
//Chapter 4
//Example 4.15
//page 310
clear;
clc;
disp("Example 4.15")
f=50;......................//frequency of induction
   motor in hertz
P=4;                            //number of poles
Nr=1440;                        //rotor speed in rpm
R2=0.1;                         //rotor resistance in
   ohms
X20=0.6;                        //rotor standstill
   resistance in ohms
Ns=(120*f)/P;
```

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

Scilab code Exa 4.16 Calculating starting torque

```
//Calculating starting torque
//Chapter 4
//Example 4.16
//page 311
clear;
clc;
disp("Example 4.16")
f=50;                          //frequency in hertz
P=4;                          //number of poles
R2=0.04;                          //rotor resistance in
    ohms
Ns=(120*f)/P;
printf("Syncronous speed=%drpm",Ns);
Nr=1200;                      //rotor speed at maximium
    torque in rpm
S=(Ns-Nr)/Ns;
printf("\nSlip at maximium torque=%f",S);
X20=R2/S;
//starting torque is developed when S=1
//r=(Tst/Tm)
r=(R2/(R2^2+X20^2))*(2*X20);
```

```
printf("\nTherefore, starting torque is %fpercent of
    the maximium torque",(r*100));
```

Scilab code Exa 4.18 `Calculating external resistance`

```
//Calculating external resistance
//Chapter 4
//Example 4.18
//page 313
clear;
clc;
disp("Example 4.18")
P=4;                        //number of poles
f=50;                       //frequency in hertz
ph=3;                        //three phase supply
R2=0.25;                      //rotor resistance in
   ohms
Nr=1440;                        //rotor speed in rpm
Ns=(120*f)/P;
S1=(Ns-Nr)/Ns;
printf("S1=%f",S1);
Nr2=1200;     //rotor speed when external is added
S2=(Ns-Nr2)/Ns;
//torque remains constant,we get the relation R2'=R2
   *(S2/S1)
R2dash=R2*(S2/S1)
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 t

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

Scilab code Exa 4.21 Calculating the slip and rotor copper loss and the output hor

```
//Calculating the slip ,rotor copper loss ,the output
    horse power and efficiency
//Chapter 4
//Example 4.21
//page 316
clear;
clc;
disp("Example 4.21")
f=50;...................//frequency of induction
    motor in hertz
P=6;....................//pole
Ns=(120*f)/P;
Nr=975;.........................//induction motor
    running speed in rpm
S=(Ns-Nr)/Ns;
printf("the slip=%f",S)
Pin=40;....................//power input to stator
    in kW
Sl=1;....................//stator losses in kW
Rin=Pin-Sl;.................//output from stator in
    kW
Rc=S*Rin;
printf("\nrotor copper losses=%fkW",Rc)
l=2;....................//total losses in kW
p=Rin-Rc-l;.................//output power in kw
HP=(p*1000)/735.5;
printf("\noutput horse output=%fHP",HP)
in=40;...........................//input in kW
effi=(p/in)*100;
printf("\nefficiency=%fpercent",effi)
```

Scilab code Exa 2.22 Calculate the value of resistance

```
//Calculate the value of resistance
//Chapter 2
//Example 2.22
//page 126
clear;
clc;
disp("Example 2.22")
V=440;                      //primary voltage in volts
Ia=50;                      //armature current in amperes
Ra=0.2;                      //armature resistance in ohms
N=600;                        //speed in rpm
E=V-(Ia*Ra);                 //emf induced  in volts
    before adding extra resistance
//E=K*phi*N=K1*Ia*N
K1=E/(Ia*N);
//we have the relation T=Kt1*Ia^2, T1=Kt1*Ia1^2
//when torque is half, say torque be T1
//T1=T/2. r=T/T1
r=2;
Ia1=sqrt(Ia^2/r);
printf("Ia1=%fA",Ia1);
//extra resistance R is introduced in the circuit
N1=400;
E1=(K1*Ia1*N1);
R=((V-E1)/Ia1)-Ra;
printf("\nvalue of extra resistance added=%fohms",R)
```

Scilab code Exa 4.22 Calculating the slip and rotor speed and mechanical power dev

```
//Calculating the slip,rotor speed,mechanical power
    developed,rotor copper loss per phase and
    resistance per phase
//Chapter 4
```

```
//Example 4.22
//page 316
clear;
clc;
disp("Example 4.22")
f=50;...........................//frequency of induction motor in hertz
P=6;............................//pole
Ns=(120*f)/P;
printf("synchronous speed=%frpm",Ns)
fr=120/60;.........................//rotor frequency
S=fr/f;
printf("\nthe slip=%f",S)
Nr=Ns-(Ns*S);
printf("\nrotor speed=%frpm",Nr)
Rin=80;........................//rotor input in kW
Rc=S*Rin;......................//Rotor copper loss in kW
Ph=3;...............................//number of phases
Rcp=(Rc/Ph)*1000;.........................//loss per phase in watt
p=((Rin-Rc)*1000)/735.5;
printf("\nmechanical power developed=%fhp",p)
Ir=60;.........................//rotor current in amperes
R2=Rcp/(Ir)^2;
printf("\nrotor resistance per phase at rotor current 60A=%fohms",R2)
```

Scilab code Exa 4.23 `Calculating additional resistance required`

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

Scilab code Exa 4.24 Calculate speed of motor and maximium torque

```
//Calculate speed of motor and maximium torque
//Chapter 4
//Example 4.24
//page 321
clear;
clc;
disp("Example 4.24")
V=400;                 //supply voltage in volts
f=50;                    //frequency in hertz
P=6;                   //number of poles
ph=3;                   //three phase supply
```

```
R2=0.03;                    //rotor resistance in ohms
X20=0.4;                     //rptor reactance in ohms
Nr=960;                    //full load speed in rpm
Ns=(120*f)/P;
printf("synchronous speed=%drpm",Ns)
S=(Ns-Nr)/Ns;                //corresponding slip
//maximium torque Tm occurs at S=(R2/X20)
//we get Tm=k/(2*X20)
a=R2/X20;
//r=Tm/T
r=(a^2+S^2)/(2*a*S);
Sm=(R2/X20);
printf("\nSlip at maximium torque,Sm=%f",Sm);
//corresponding speed
Nr2=Ns*(1-Sm);
printf("\nRotor speed at maximium torque=%drpm",Nr2)
```

Scilab code Exa 4.25 Calculate starting current

```
//Calculate starting current
//Chapter 4
//Example 4.25
//page 321
clear;
clc;
disp("Example 4.25")
V=400;                     //supply voltage in volts
f=50;                        //frequency in hertz
P=4;                       //number of poles
ph=3;                       //three phase supply
S=0.04;
If=30;                      //Full load current in
   amperes
```

```
Isc=6*If;
//let r be the ratio of starting torque nd full load torque, r=Ts/Tf
r=(Isc/If)^2*S;
//Tf=Tm is produced when voltage is Vm
Vm=sqrt(V^2/r);
printf("\nvoltage at maximium torque=%fvolts",Vm);
Is=6*If*(Vm/V);
printf("\nFull-load current at 333.3 volts is=%fA",Is)
```

Scilab code Exa 4.26 Calculate starting line current and starting torque

```
//Calculate starting line current and starting torque
//Chapter 4
//Example 4.26
//page 330
clear;
clc;
disp("Example 4.26")
V=400;                      //supply voltage in volts
f=50;                          //frequency in hertz
Id=75;                          //current taken when delta-connected in amperes
printf("current taken when delta-connected=%dA",Id);
Is=Id/3;                          //current taken when star-connected in amperes
printf("\ncurrent taken when star-connected=%dA",Is);
//Tfl be the full load torque
//r=Ts/Tfl
r=1.5;
```

```
//since voltage becomes (1/sqrt(3)) when star connected
//torque is directly proportional to square of voltage
printf("\nStarting torque with winding star connected=%f times of Tfl",(r/3));
```

Scilab code Exa 4.28 Calculate starting torque

```
//Calculate starting torque
//Chapter 4
//Example 4.28
//page 333
clear;
clc;
disp("Example 4.28")
ph=3;
//rotor copper loss=slip*rotor input
//Tst= starting torque
//Tfl=torque at full load
//Ist/Ifl=r
r=6;
S=0.04
printf(" At slip=0.04")
printf("\nFor direct-on-line starting, (Tst/Tfl)=%f",((r^2*S)));
//phase current in start is (1/sqrt(3)) times the phase current in delta

printf("\nFor direct-on-line starting, (Tst/Tfl)=%f",((r/sqrt(3))^2*S));
```

Scilab code Exa 4.29 Calculate full load speed

```
//Calculate full load speed
//Chapter 4
//Example 4.29
//page 334
clear;
clc;
disp("Example 4.29")
V=400;                          //voltage in volts
f=50;                          //frequency in hertz
P=4;                          //number of poles
//r1=(Ts/Tfl)
r1=1.6;
//r2=(Tm/Tfl)
r2=2;
//r3=(Ts/Tm)=(2*a)/(1+a^2)
r3=0.8;
//on solving , we get a=0.04   ,
a=0.04;
Sm=0.04;    //slip at maximium torque
printf("Slip at maximium torque,Sm=%f",Sm)
Ns=(120*f)/P;                 //synchronous speed in rpm
Nr=Ns*(1-Sm)                  //rotor speed in rpm
//r2=(a^2+Sfl^2)/(2*a*Sfl)
Sfl=0.01;
Nr2=Ns*(1-Sfl);
printf("\nfull load speed,Nr=%drpm",Nr2)
```

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

```
//Calculate full load rotor loss and rotor input and
    output torque
//Chapter 4
//Example 4.30
//page 345
clear;
clc;
disp("Example 4.30")
hp=20;                      //power in horsepower
f=50;                      //frequency in hertz
P=4;                      //number of poles
Ns=(120*f)/P;                      //synchronous speed
printf("Synchronous speed,Ns=%drpm",Ns);
S=0.04;                                  //slip
Nr=Ns*(1-S);
OP=hp*735.5;
printf("\nOutput power=%fW",OP);
OT=OP/(2*3.14*(Nr/60));
printf("\nOutput torque=%fNm",OT);
FL=0.02*OP;              //Friction and windage loss
PD=OP+FL;
printf("\nPower developed by the rotor=%fW",PD);
//from relation, (rotor I^2R-loss=S*Rotor input) we
   get following relation
RL=(S*PD)/(1-S);
printf("\nRotor I^2R-loss=%fW",RL);
RI=RL/S;
printf("\nRotor input=%dW",RI)
```

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

```
//Calculate full load rotor loss and rotor input and output torque
//Chapter 4
//Example 4.31
//page 347
clear;
clc;
disp("Example 4.31")
P=4;                    //number of poles
f=50;                   //frequency in hertz
V=230;                  //voltage in volts
hp=5;                   //power in horsepower
Ib=15;                  //current in block rotor test in amperes
output=hp*735.5;        //output in watts
//in block rotor test: power input=Full=load I^2R losses=735W
FLl=735;                                  //Full-load I^2R losses
printf("Full-load I^2R losses=%fW",FLl);
Re=FLl/(3*Ib^2);
Io=6.3;                   //current in no load condition in amperes
lossNL=(3*(Io)^2*Re);     //I^2R loss at no-load condition
printf("\nI^2R loss at no-load=%fW",lossNL);
PiNL=275;                   //power input at no-load
printf("\nCore loss plus friction and windage loss=%dW",(PiNL-lossNL));
TL=FLl+(PiNL-lossNL);
```

```
effi=(output*100)/(output+TL);
printf("\nEfficiency=%fpercent",effi)
```

Scilab code Exa 4.32 Calculate full load efficiency

```
//Calculate full load efficiency
//Chapter 4
//Example 4.32
//page 347
clear;
clc;
disp("Example 4.32")
Vl=415;              //voltage in volts
Il=50;                  //line current in amperes
R1=0.5;                  //resistrance of stator
    winding per phase in ohms
pf=0.85;                //power factor
S=0.04;
IFL=(sqrt(3)*Vl*Il*pf)          //input to the motor
    on full load
printf("Input to the motor on full load=%dW",IFL);
I1=Il/sqrt(3);
SLFL=(3*I1^2*R1)              //Stator I^2R loss on
    full load
printf("\nStator I^2R loss on full load=%dW",SLFL);
//given ratio of stator core loss friction and
    windahe loss be r=(r1:r2)
r1=3;
r2=2;
TL=1500;              //total loss
SCL=(r1*TL)/(r1+r2);               //stator core loss
FWL=(r2*TL)/(r1+r2);               //Friction and
    windage loss
```

```
SL=SLFL+SCL;                                //total stator loss
SI=IFL;                                     //Stator input
Pa=SI-SL;                                   //power transferred through the air-gap=input to the rotor
RI=Pa
RL=S*RI;                                    //rotor losses
TRL=FWL+RL;                                 //total rotor losses
OP=RI-TRL;                                  //Output power at the shaft
effi=(OP*100)/SI;
printf("\nEfficiency=%f percent",effi)
```

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

```
//Calculating the rotor current at slip 3 precent and when the rotor develops maximum torque
//Chapter 4
//Example 4.33
//page 351
clear;
clc;
disp("Example 4.33")
E20=100;...............................//induced emf between slip terminals in volts
E20p=E20/sqrt(3);........................//induced emf per phase in volts
printf("induced emf per phase=%fV",E20p)
S=3/100;...........................//slip
R2=0.2;..............................//resistance in ohms
```

```
X20=1;................................//standstill
   resistance in ohms
I2=(S*E20p)/sqrt((R2)^2+(S*X20)^2)
printf("\nrotor current at slip 0.03 =%fA per phase"
   ,I2)
Sm=R2/X20;
I2m=(Sm*E20p)/sqrt((R2)^2+(Sm*X20)^2)
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

```
//Calculating the rotor current at slip 3 precent
   and when the rotor develops maximum torque
//Chapter 4
//Example 4.34
//page 352
clear;
clc;
disp("Example 4.34")
E20=120;......................//induced emf of motor
    at standstill in volts
E20p=120/sqrt(3);.....................//induced emf
   per phase
f=50;.............................//frequency of
   the motor in hertz
R2=0.2;...............................//Rotor
   Resistance  per phase
X20=1;...................................//
   Standstill resistance in ohms
P=4;..............................//pole
I=16;.......................//
S=(I*R2)/sqrt((E20)^2-(I*X20)^2);
```

```
Ns=(120*f)/P;
printf("Synchronous speed=%frpm",Ns)
Nr=Ns-(Ns*S)
Sm=R2/X20;
Nr=Ns-(Ns*Sm)
I2=(Sm*E20p)/sqrt((R2)^2+(Sm*X20)^2)
printf("\nrotor current at maximum torque=%fAper
    Phase",I2)
Pi=(3*((I2)^2)*R2)/Sm;
printf("\nRotor input for the three phase=%fW",Pi)
```

Scilab code Exa 4.35 Calculate the circuit elements

```
//Calculate the circuit elements
//Chapter 4
//Example 4.35
//page 356
clear;
clc;
disp("Example 4.35")
R1dc=0.01;                         //DC resistance in
    ohms
V=400;                             //voltage in volts
r=1.5;                         //ratio of ac to dc
    resistance
R1=r*R1dc;                         //AC resistance in ohms
//at no-load
Io=20;                         //no-load current in
    amperes
SL=(3*Io^2*R1);             //I^2R loss in the stator
    phases in watts
FWL=300;                       //Friction and windage loss
     in watts
```

```
TL=1200;                    //total losses=no-load power input in watts
CL=TL-(SL+FWL);             //core loss in watt
CLp=CL/sqrt(3);               //core loss per phase
Vp=V/sqrt(3);                 //voltage per phase
Rm=(Vp^3)/CL;                   //motor resistance
pf=CL/(Vp*Io);
phi0=acosd(pf);
Xm=Vp/(Io*sind(phi0));                    //motor reactance
//Under blocked rotor test
Vb=100;                     //voltage in volts
Isc=45;                       //current in amperes
Vbp=100/sqrt(3);            //voltage per phase in volts
P=2750;                     //power supplied in watts
Ze=Vbp/Isc;                         //Motor impedance reffered to stator side in ohms
Re=P/(3*Isc^2);
R2=Re-R1;                 //rotor resistance referred to stator side
Xe=sqrt(Ze^2-Re^2);
//assuming X1=X2
X2=Xe/2
X1=X2;
printf("Thus the elements of the equivalent circuit are:");
printf("\nRm=%fohms",Rm);
printf("\nXm=%fohms",Xm);
printf("\n\nR1=%fohms",R1);
printf("\nrotor resistance referred to stator side, R2=%fohms",R2);
printf("\nequivalent resistance referred to stator side,Re=%fohms",Re);

printf("\n\nX1=%fohms",X1);
printf("\nrotor reactance referred to stator side,X2=%fohms",X2);
```

```
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

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

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

Scilab code Exa 5.2 `To calculate distribution factor`

```
//chapter 5
//example 5.2
//page 425
clear;
clc;
disp("example 5.2")
printf("\n");
slots=36;     //number of slots
poles=4;      //number of poles
ph=3;         //single layer three phase winding
SP=slots/ph; //number of slots per phase
printf("number of slots per phase= %d\n",SP);
m=SP/poles;  //munber of slots per pole per phase
printf("number of slots per pole per phase,m=%d\n",m)
SA_m=360/slots;          //slot angle mechanical
SA_e=(poles/2)*SA_m      //slot angle electrical
printf("slot angle= %d degree electrical\n",SA_e)
k_d=sind((m*SA_e)/2)/(m*sind(SA_e/2));
printf("distribution factor= %f",k_d)
```

Scilab code Exa 5.3 `To calculate pitch factor`

```
//chapter 5
```

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

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

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

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

Scilab code Exa 5.5 Calculating useful flux per pole

```
//chapter 5
//example 5.5
//page 427
clear;
clc;
disp("example 5.5");
disp("E(line to line)=  440V");
E_l=440;     //line-to-line voltage
E_p=E_l/(sqrt(3));
N=750;       //speed in rpm
fre=50;      //frequency
P=(120*fre)/N;
printf("P=  %d\n",P);
printf("E(per phase)=  %dV\n",E_p);
ph=3;        //3-phase machine
```

```
m=2;          //number of slots per pole per phase
slots=m*P*ph;         //total number of stator slots
SA_m=360/slots;       //slot angle mechanical
SA_e=(P/2)*SA_m;      //slot angle electrical
k_p=1;                //assuming full pitch
printf("slot angle= %d degree electrical\n",SA_e);
printf("pitch factor=%f\n",k_p);
k_d=sind((m*SA_e)/2)/(m*sind(SA_e/2));
printf("distribution factor= %f\n\n",k_d);
//2 slots per pole per phase
NSp=2*P;              //number of slots per phase
NTc=4;                //number of turns per coil
T=8*NTc;              //number of turns per phase
Fp=E_p/(4.44*fre*T*k_d*k_p);
printf("flux per pole= %fWb\n",Fp);
```

Scilab code Exa 5.6 To calculate the frequency and induced EMF

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

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

Scilab code Exa 5.7 Finding the number of armature conductors

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

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

Scilab code Exa 5.8 To calculate induced EMF per phase

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

```
printf("total number of conductors per phase= %d\n",
    ((TS*Cs)/3));
T=((TS*Cs)/3)/2;
printf("total number of turns per phase=%d\n",T)
m=(TS/(P*3));
printf("number of slots per pole per phase,m= %d\n",
   m);
SA_m=360/TS;              //slot angle mechanical
SA_e=(P/2)*SA_m;
k_d=sind((m*SA_e)/2)/(m*sind(SA_e/2));
printf("distribution factor=%f\n\n",k_d);
disp("coil pitch is 5/6 of full-pitch");
printf("\n");
bheta=180-(5/6)*180;      //short pitch angle
printf("short pitch angle= %d degrees\n",bheta)
k_p=cosd(bheta/2);
printf("pitch factor= %f \n",k_p);
E=4.44*Fp*f*T*k_d*k_p;
printf("induced per phase=  %fV\n",E)
```

Scilab code Exa 5.9 To find the voltage regulation

```
//chapter 5
//example 5.9
//page 439
clear;
clc;
disp("example 5.9");
printf("\n");
OP=500000;        //output power
V_l=3300;         //line voltage
I_l=OP/(sqrt(3)*V_l);       //line current
printf("line current=  %fA\n",I_l);
```

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

Scilab code Exa 5.10 To calculate voltage regulation

```
//chapter 5
//example 5.10
//page 440
disp("example 5.10")
clear;
clc;
V=2000;
V_oc=500;           //open circuit voltage
I_sc=100;           //short circuit current
I_a=100;
R_s=0.8;            //armature resistance
Z_s=V_oc/I_sc;      //synchronous impedence
printf("Z_s= %d ohm\n",Z_s);
X_s=sqrt(Z_s^2-R_s^2);
printf("X_s= %f ohm\n",X_s);
pf=1;
```

```
phi=acosd(pf);
disp("At unity power factor");
printf("\n");
E=sqrt((V*cosd(phi)+I_a*R_s)^2+(V*sind(phi)+I_a*X_s)
    ^2);
printf("induced emf= %fV\n",E);
R=((E-V)*100)/V;
printf("regulation= %f percent\n",R);
clear pf;
pf=0.71;
phi=acosd(pf);
disp("At 0.71 lagging power factor");
printf("\n");
E=sqrt((V*cosd(phi)+I_a*R_s)^2+(V*sind(phi)+I_a*X_s)
    ^2);
printf("induced emf= %fV\n",E);
R=((E-V)*100)/V;
printf("regulation= %fpercent\n",R);
clear pf;
pf=0.8;
phi=acosd(pf);
disp("At 0.8 leading power factor");
printf("\n");
E=sqrt((V*cosd(phi)+I_a*R_s)^2+(V*sind(phi)-I_a*X_s)
    ^2);
printf("induced emf= %fV\n",E);
R=((E-V)*100)/V;
printf("regulation=  %fpercent\n",R);
```

Scilab code Exa 5.11 To calculate internal voltage drop

```
//chapter 5
//example 5.11
```

```
//page 441
clear;
clc;
disp("example 5.11");
printf("\n");
disp("field exitation current=10A");
V_oc=900;      //induced emf on open circuit
I_sc=150;      //short circuit current
Z_s=V_oc/I_sc; //synchronous impedence
printf("synchronous impedence,Z_s= %d ohm\n",Z_s);
I_a=60;
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

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

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

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

```
//chapter 5
//example 5.13
//page 442
clear;
clc;
disp("example 5.13");
printf("\n");
KVA=1200;         //output power
printf("output power=%d\n",KVA)
V_l=3300;         //line voltage
R_a=0.25;         //armature resistance
I_l=(KVA*1000)/(sqrt(3)*V_l);     //line current
//for star connected I_l=I_a
I_a=I_l;
V_p=V_l/sqrt(3);
printf("V per phase= %dV\n",V_p)
//field current of 40A produces short circuit
   current of 200A and open circuit emf 1100
v_l=1100;
i_s=200;
Z_s= v_l/(sqrt(3)*i_s);      //synchronous impedence
printf("Synchronous impedance,Zs=%f ohm\n",Z_s)
X_s=sqrt(Z_s^2-R_a^2);       //synchronous reactance
disp("(a)for 0.8 lagging power facor");
pf=0.8;
phi=acosd(pf);
```

```
E=sqrt((V_p*cosd(phi)+I_a*R_a)^2+(V_p*sind(phi)+I_a*X_s)^2)
printf("induced emf,E=%f V\n",E);
R=((E-V_p)*100)/V_p;
printf("regulation=%f percent\n\n",R);
clear pf;
pf=0.8;
phi=acosd(pf);
disp("(b)For leading power factor load")
E=sqrt((V_p*cosd(phi)+I_a*R_a)^2+(V_p*sind(phi)-I_a*X_s)^2)
printf("induced emf,E= %f V\n",E);
R=((E-V_p)*100)/V_p;
printf("regulation=%f percent",R);
```

Scilab code Exa 5.14 To calculate terminal voltage for same excitation and load cu

```
//chapter 5
//example 5.14
//page 443
clear;
clc;
disp("example 5.14");
disp("star connected alternator")
printf("\n");
KVA=1500;            //rating
ph=3;                //3-phase
V_l=6600;            //voltage
Ra=0.4               //armature resistance
Xs=6;                //reactance
Ia=(KVA*1000)/(sqrt(3)*V_l);
printf("Full-load current= %d A\n",Ia);
V=V_l/sqrt(3);
```

```
printf("Voltage per phase=%d V\n",V);
disp("for 0.8 lagging power facor");
pf=0.8;            //power factor
phi=acosd(pf);
E=sqrt((V*cosd(phi)+Ia*Ra)^2+(V*sind(phi)+Ia*Xs)^2)
printf("induced emf=%f V\n\n",E);
disp("then at 0.8 leading power factor");
Vt=4743; //solved manually
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

```
//chapter 5
//example 5.15
//page 450
clear;
clc;
disp("example 5.15");
L=8000;      //load
La=5000;
pf=0.8;
phi=acosd(pf);
printf("\ntan phi=  %f\n",tand(phi));
disp("FOR ALTERNATOR A");
pf_a=0.9;
phi_a=acosd(pf_a);
printf("\ntan phi_a=  %f\n",tand(phi_a));
disp("reactive load=active load*tan phi");
disp("Active load=8000kW");
printf("reactive load=  %d KVAr\n",(8000*tand(phi_a)
    ));
disp("Active Load A=5000kW\n");
```

```
printf("Reactive load A= %dkVAr\n",(5000*tand(phi_a
    )));
printf("Active load of B= %dkW\n",L-La);
a=((8000*tand(phi))-(5000*tand(phi_a)))
printf("Reactive load of B= %dkVAr\n",a);
B=a/(L-La);
phi_b=atand(B);
printf("phi_b= %f\n",phi_b)
printf("Power Factor of B= %f",cosd(phi_b));
```

Scilab code Exa 5.16 To calculate armature curren and power factor

```
//chapter 5
//example 5.16
//page 451
clear;
clc;
disp("example 5.16")
V=6600;
ph=3;  //3-phase alternators
power=10000; //total load
disp("Two alternators in parallel connection");
pf=0.8;
Ia=438;         //armature current
Il=(power*1000)/(sqrt(3)*V*pf);   //load current
printf("load current= %fA\n\n",Il);
phi=acosd(pf);
Ac=(Il*cosd(phi));
Rc=(Il*sind(phi));
printf("Active component of current= %fA\n",Ac);
printf("Reactive component of current= %fA\n",Rc);
printf("Current supplied by each alternator=%fA\n",(
    Il/2));
```

```
printf("Active component of current supplied by each alternator= %fA\n",(Ac/2));
printf("Reactive component of current supplied by each alternator= %fA\n\n",(Rc/2));
disp("Since steam supply is same,the active component remain the same ");
RI1=sqrt(Ia^2-(Ac/2)^2);
printf("Reactive component of I1= %dA\n",RI1);
RI2=(Rc-RI1);
printf("reactive component of I2= %fA\n",RI2);
I2=sqrt((Ac/2)^2+(RI2)^2);
printf(" I2= %fA\n",I2);
phi_2=atand(RI2/(Ac/2));
printf("phi 2= %f degrees\n",phi_2);
printf("cos phi 2= %f",cosd(phi_2));
```

Scilab code Exa 5.17 To determine KVA rating and power facor

```
//chapter 5
//example 5.17
//page 455
clear;
clc;
disp("example 5.17");
disp("power factor of existing load is 0.8 lagging");
pf=0.8; //power factor
phi=acosd(pf);
printf("phi= %d degree\n",phi);
L=800;    //load
kVAr1=(L*tand(phi));
printf("kVAr1= %d \n",kVAr1);
disp("output for the synchronous motor is 200kW");
```

```
output=200;
efficiency=0.9;
kW=(output/efficiency);
printf("Input to the synchronous motor= %fkW\n",kW);
TL=(L+kW); // total load
printf("Total load on the system= %fkW\n",TL);
disp("overall power factor of the load is to be raised to 0.92 lagging");
pf=0.92;
phi=acosd(pf);
kVAr2=(TL*tand(phi))
printf("kVAr2=%f\n",kVAr2);
kVAr=kVAr1-kVAr2;
printf("lagging kVAr of synchronous codenser= %f\n",kVAr);
printf("leading kVAr supplied by the motor= %f\n",kVAr);
phi=atand(kVAr/kW);
printf("phi= %d degree\n\n",phi);
printf("Power factor of the synchronos motor= %f leading \n",cosd(phi));
printf("KVA rating of the synchronous motor= %f",(kW/cosd(phi)));
```