Scilab Textbook Companion for An Introduction to Electrical Machines and Transformers by G. Mcpherson[1]

Created by
Rohan Singla
Electrical Technology
Electrical Engineering
Thapar University
College Teacher
None
Cross-Checked by
Spandana

July 30, 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: An Introduction to Electrical Machines and Transformers

Author: G. Mcphersion

Publisher: John Wiley, NY

Edition: 2

Year: 1990

ISBN: 978-0-471-63529-1

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

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

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

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

Contents

List of Scilab Codes

Chapter 1

What machines and transformers have in common

Scilab code Exa 1.1 `Find efficiency of the 5 horesepower three indcution motor ope`

```
clc

horsepower=2.5  //rating of induction motor in horsepower at half load
Vl=230 //terminal voltage of motor in volts
Il=7   //load current of motor in amperes
pf=0.8 //power factor of the machine
Pin=sqrt(3)*Vl*Il*pf //input power in watts
mprintf("Pin=%f W\n",Pin)//The answer may vary due to roundoff error
Whp=746 //watts per hp
Pout=horsepower*Whp //output power in watts
mprintf("Pout=% f W\n",Pout)
mprintf("  =%f\n",Pout/Pin)//The answer may vary due to roundoff error //efficiency of the machine
mprintf("Losses=Pin-Pout=%f W\n",Pin-Pout)//The answer may vary due to roundoff error //losses in the machine in watts
```

Scilab code Exa 1.2 Calculate efficiency and output power when losses are 365 Watt

```
clc

//the below exmaple is an extension of Ex1_1.sce
Vl=230 //terminal voltage of machine in volts
Il=7 //current drawn by machine in amperes
pf=0.8 //power factor of machine
Pin=sqrt(3)*Vl*Il*pf //from Ex1_1 //input power in watts
Losses=365 //in watts
Pout=Pin-Losses //output power in watts
Whp=746 //watts per hp
mprintf("  =1-(Losses/Input)=%f\n",1-(Losses/Pin)) //The answer may vary due to roundoff error //efficiency of the machine
mprintf("Pout=%f W\n",Pout)//The answer may vary due to roundoff error
mprintf("Pout=%fhp",Pout/Whp)//The asnwer may vary due to roundoff error //output power in horsepower
```

Scilab code Exa 1.3 Find hysteresis loss in core A and B

```
clc
f=60 //frequency of voltage source in Hz
x=1.9 //Steinmetz coefficient
V=80 //applied sinusoidal voltage in volts
t=100 //no of turns wound on a coil
hc=500 //hysteresis coefficient
w=2*%pi*f //angular frequency in rads/sec
```

```
phimax=(sqrt(2)*V)/(t*w)//maximum value of flux in the core in webers
mprintf("phimax=%fWb\n",phimax)//the answer may vary due to roundoff error
A1=0.0025 //cross-sectional area of core in metre square
Bmax1=phimax/A1 //flux density in core A in tesla
mprintf("Bmax=%fT\n",Bmax1)//the answer may vary due to roundoff error
lfe1=0.5 //mean flux path length of core A in meters
VolA=A1*lfe1 //volume of core A in metre cube
mprintf("VolA=%f metre cube\n",VolA)
//for core A
Ph1=VolA*f*hc*(Bmax1^x) //hysteresis loss in core A in watts
mprintf("Ph=%f W\n",Ph1)//the answer may vary due to roundoff error
//for core B
A2=A1*3 //cross sectional area of core B in metre square
lfe2=0.866 //mean flux path length of core B in metres
Bmax2=phimax/A2 //flux density in core B in tesla
VolB=A2*lfe2 //volume of core B in metre cubes
Ph2=VolB*f*hc*(Bmax2^x) //hysteresis loss of core B in watts
mprintf("Ph=%f W\n",Ph2)//the answer may vary due to roundoff error
```

Scilab code Exa 1.4 Find voltage induced at 30Hz eddycurrent loss at 60Hz hysteres

```
clc
V1=240 //voltage applied to a winding of transformer (three phase) in volts
f1=60 //initial applied frequency in Hz
```

```
f2=30 //reduced frequency in Hz
Phe1=400 //core loss in watts at f1 frequency
Phe2=169 //core losses in watts at f2 frequency
mprintf("V2=%dV\n",(f2*V1)/f1)//voltage at 30 Hz
   frequency
mprintf("Ph+e/f=Ch+Ce*f\n")//equation for
   claculating  hysteresis and eddy current loss
   coefficients
a=[1 f1;1 f2] //left hand side matix for the
   equation above
b=[Phe1/f1;Phe2/f2] //right hand side matrix for the
    equation above
c=inv(a)*b
Ch=c(1,:)//hysteresis loss coefficient in W/Hz
Ce=c(2,:)//eddy current loss coefficient in W/(Hz*Hz
   )
mprintf("Ph=%fW\n",Ch*f1)//ans may vary due to
   roundoff error //hysteresis loss in watts at 60
   Hz
mprintf("Pe=%fW\n",Ce*f1*f1)//ans may vary due to
   roundoff error //eddy current loss at 60 Hz in
   watts
```

Scilab code Exa 1.5 Find the kVA load for the transformer to operate at maximum ef

```
clc
Pk=75 //core loss of transfomer in watts
R=0.048 //internal resistance in ohms
V2=240// secondary voltage in volts
I2=sqrt(Pk/R)//secondary current in amperes
mprintf("I2=%f A\n",I2)//ans may vary due to
   roundoff error
mprintf("|S|=V2*I2=%d VA",V2*I2)//The answer in the
   textbook is wrong //output volt ampere of
   transformer
```

Scilab code Exa 1.6 Find full load speed regulation of the motor

```
clc
sfl=1746 //speed at full load in rev/min
snl=1799.5 //speed at no load in rev/min
mprintf("Voltage Regulation=%f",(snl-sfl)/sfl) //the
    ans may vary due to round of error
```

Scilab code Exa 1.7 Find voltage regulation for unity power factor and zero point

```
clc
Vnl=27.3 //no load voltage in volts
Vfl1=24   //full load voltage at power factor 1 in
    volts
mprintf("(Vnl-Vfl/Vfl)=%f\n",(Vnl-Vfl1)/Vfl1) //ans
    may vary due to roundoff error
Vfl2=22.1 //full load voltage at power factor 0.7 in
    volts
mprintf("Voltage Regulation=%f",(Vnl-Vfl2)/Vfl1)
```

Chapter 2

Synchronous Machines

Scilab code Exa 2.1 Find flux per pole off the synchronous machine

```
clc
L=0.25 //length of stator stack in metre
r=0.15 //radius of stator stack in metres
BImax=0.96 //peak value of air gap flux density in tesla
P=6 //no of machine poles
phi=(4*L*r*BImax)/P //flux per pole in webers
mprintf("  =%fWb",phi)
```

Scilab code Exa 2.2 Determine the coil pitch and pitch factor

```
clc
//the example below is an extension of Ex2_1
L=0.25 //length of stator stack in metres
r=0.15 //radius of stator stack in metres
BImax=0.96 //peak value of air gap flux density in tesla
P=6 //no of machine poles
```

```
phi=(4*L*r*BImax)/P //flux per pole in webers
//above comes from Ex2_1
span=5 //span of each coil given by  no of slots
edps=30 //electrical degrees per slot in degrees
p=span*edps//coil pitch in degrees
mprintf("p=%d \n",span*edps)
Nc=2//turns of coil
Kp=sin(((p/2)*%pi)/180) //pitch factor //degree being converted to radians before calculation
mprintf("Kp=sin(p/2)=%f\n",Kp) //the ans may vary due to roundoff error
mprintf(" cmax =Nc*Kp*  =%fWb turns\n",Nc*Kp*phi)//max flux linkage //ans may vary due to roundoff error
ns=1000 //machine speed in rev/min
p=6 //no of poles
f=(p*ns)/120 //frequency at given speed in Hertz
mprintf("f=%dHz\n",f)
mprintf("Ec=sqrt(2)*  *f*Nc*kp*  =%fV\n",sqrt(2)*%pi*f*Nc*Kp*phi)//ans may vary due to roundoff error //voltage induced at above frequency
```

Scilab code Exa 2.3 If the armature winding is star connected find the phase volta

```
clc
//the example below is an extension of Ex2_1 and Ex2_2
S1=36 //no of slots
q1=3 //no of phases
p=6 //no of poles
Nc=2 //no of turns per coil
L=0.25 //length of stator stack in metres
r=0.15 //radius of stator stack in metres
BImax=0.96 //peak value of air gap flux density in tesla
```

```
P=6 //no of machine poles
phi=(4*L*r*BImax)/P //flux per pole in webers
span=5 //span of each coil given by  no of slots
edps=30 //electrical degrees per slot in degrees
p=span*edps//coil pitch in degrees
Nc=2//turns of coil
kp=sin(((p/2)*%pi)/180) //pitch factor //degree being converted to radians before calculation
ns=1000 //machine speed in rev/min
p=6 //no of poles
f=(p*ns)/120 //frequency at given speed in Hertz
Ec=sqrt(2)*%pi*f*Nc*kp*phi//voltage induced at above frequency
n=S1/(q1*p)
mprintf("n=S1/(q1*p)=%f\n",n) //coils per group
edps=30 //electrical degrees per slot //equal to  as per textbook
kd=(sin((n*edps*%pi)/(180*2)))/(n*sin((edps/2)*%pi/180)) //distribution factor of the machine //degree converted to radian for calculation
mprintf("kd=sin(n*   /2)/n*sin(   /2)=%f\n",kd)//ans may vary due to roundoff error
mprintf("|Egroup|=n*Ec*kd=%fV\n",n*Ec*kd)//ans may vary due to roundoff error
mprintf("|E  |=p*|Egroup|=%fV\n",p*n*Ec*kd)//ans may vary due to roundoff error
mprintf("sqrt(3)*E  =%dV\n",sqrt(3)*n*Ec*kd*p)//ans may vary due to roundoff error
stp=n*Nc*p //series turns per phase //equal to N  in textbook
mprintf("N  =n*Nc*p=%dturns\n",stp)
mprintf("|E  |=sqrt(2)*   *N  *f*   *kp*kd=%fV",sqrt(2)*%pi*stp*f*kp*kd*phi) //ans may vary due to round off error //induced phase winding
```

Scilab code Exa 2.4 1Find horsepower being delivered to the pump2Rheostat setting

```
clc
Vl=2300 //terminal voltage of synchronous motor in
   volts
Il=8.8 //minimum line current in ampere
P=sqrt(3)*Vl*Il
mprintf("P=%fKW\n",P/1000)//power drawn from the
   line //ans may vary due to round off error
pf=0.8 //operating power factor
mprintf("HP=P/746=%fhp\n",P/746)//ans may vary due
   to round off error //conversion of power to hp
   requires division by 746
S=P/(pf*1000) //total volt amperes of motor in kVA
mprintf("Q=|S|sin  m=|S|sin cos-1(pf)=%fkVAR",S*
   sin(acos(pf))) //kVAR supplied by motor to the
   system //ans may vary due to roundoff error
```

Scilab code Exa 2.5 Find new plant power factor and percent decrease in line curre

```
clc
//the following code contains userdefined fucntion
   complexstring
function s=complexstring(a)

    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
Load=5000 //load of the plant in kW
pf1=0.8 //power factor of load(lagging)
```

```
pf2=0.9 //power factor of induction motor
pf3=0.8 //power factor of synchronous motor
Hp=500 //rating of induction motor to be replaced in hp
Pout=0.746*Hp //output power of induction motor in kW
Eta=0.96 //efficiency of the induction motor equal to    in textbook
Sp=Load+(Load*tan(acos(pf1)))*%i //original complex power of load in kVA
disp('Sp=' + complexstring(Sp)+'kVA')
Pin=Pout/Eta //input power in kW
mprintf("Pin=%fkW\n",Pin)//complex power of induction motor //the ans may vary due to round off error
Sm=Pin+(Pin*tan(acos(pf2)))*%i
disp('Sm='+complexstring(Sm)+'kVA')//the ans may vary due to round off error //complex power of induction motor
mprintf("\n")
Ss=Pin-(Pin*tan(acos(pf3)))*%i
disp('Ss='+complexstring(Ss)+'kVA')//complex power of synchronous machine //the ans may vary due to round off error
mprintf("\n")
Qm=(Pin*tan(acos(pf2)))*%i//reactive power of induction motor in kVAR
Qs=(-1*(Pin*tan(acos(pf3)))*%i)//reactive power of synchronous motor in kVAR
Sp1=Sp-Qm+Qs
disp('Sp1='+complexstring(Sp1)+'kVA')//new plant requirement,equal to Sp' in textbook
mprintf("\n")
pha=acos(real(Sp1)/abs(Sp1)) //phase angle of Sp1 in radians
mprintf("New power factor=%f\n",cos(pha))//new power factor //the ans may vary due to round off error
invl=abs(Sp)//initial value of complex power in kVA
```

```
fnvl=abs(Sp1) //final value of complex power in kVA
mprintf("Percent reduction=%f%c\n",(((invl-fnvl)/invl)*100),'%')//the ans may vary due to round off error
```

Scilab code Exa 2.6 What should be the kVAR rating of a synchronous condenser to c

```
clc
//the example below is an extension of Ex2_5
//the following code contains userdefined fucntion complexstring
function s=complexstring(a)

    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
Load=5000 //load of the plant in kW
pf1=0.8 //power factor of load(lagging)
Sp=Load+(Load*tan(acos(pf1)))*%i //original complex power of load in kVA
disp('Sp='+complexstring(Sp)+'kVA')
pf2=0.9 //new power factor
Qp1=real(Sp)*tan(acos(0.9)) //reactive power,equal to Qp` in textbook
mprintf("Qp`=%fkVAR\n",Qp1)//the ans vary due to roundoff error
Qp=imag(Sp)
mprintf("Qs=%fkVAR",Qp1-Qp)//KVAR to be supplied by synchronous condenser
```

Scilab code Exa 2.7 Calculate the excitation requirement for the alternator

```
clc

//the code below uses userdefined complexstring
   function
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
VLB=2400 //line to base voltage in volts
V1=VLB/sqrt(3)  //reference phasor in volts //ans
   may vary due to roundoff error
mprintf("V1=%fV\n",V1)
kVAB=9375 //rated kVA
I1B=(kVAB*1000)/(sqrt(3)*VLB)
pf=0.8 //power factor
mprintf("I1B=%fA\n",I1B) //ans may vary due to
   roundoff error
I1=I1B*exp((-1)*%i*(acos(pf)))
disp('I1='+complexstring(I1)+'A')//ans may vary due
   to roundoff error
mprintf("\n")
x1=0.1//in ohms
disp('E =V1+jI1x1='+complexstring(V1+%i*I1*x1)+'V')
   //ans may vary due to roundoff error
mprintf("\n")
```

```
disp('sqrt3*|E  |='+complexstring((abs(V1+%i*I1*x1))
    *sqrt(3))+'V')
Ifu=110 //value in ampere,dc
Ifs=149 //value in ampere,dc
ks=Ifs/Ifu
mprintf("ks=%f\n",ks) //ans may vary due to roundoff
     error
m1=(abs((V1+%i*I1*x1)))/Ifs //equal to m‘ in
    textbook
mprintf(" m    =|E  |/Ifs=%f \n",m1)//ans may vary
    due to roundoff error
xdu=0.8 //in ohms
xd=x1+((xdu-x1)/ks)
mprintf("xd=x1+(xdu-x1)/ks=%f \n",xd)//ans may vary
     due to roundoff error
Ef=V1+(%i*I1*xd)
disp('Ef='+complexstring(Ef)+'V')//ans may vary due
    to roundoff error
mprintf("\n")
mprintf("If=%fA\n",abs(Ef)/m1)//ans may vary due to
    roundoff error
```

Scilab code Exa 2.8 Find filed current and open circuit voltage and maximumkVAR

```
clc
VLB=2400 //line to base voltage in volts
Ix=2005 //current in amperes
xda=VLB/(sqrt(3)*Ix)
mprintf("xda=%f \n",xda)//ans may vary due to
    roundoff error
Ifv=116 //current in amperes
ma1=VLB/(sqrt(3)*Ifv)//equal to ma‘ in textbook
mprintf(" ma =V1B/Ifv=%f \n",ma1)//ans may vary due
     to roundoff error
//from ex 2_7
```

```
V1=VLB/sqrt(3)  //reference phasor in volts
kVAB=9375 //rated kVA
I1B=(kVAB*1000)/(sqrt(3)*VLB)//current in amperes
pf=0.8 //power factor
I1=I1B*exp((-1)*%i*(acos(pf)))//current in amperes
Ef=V1+%i*I1*xda
disp('Ef='+string(Ef)+'V')//ans may vary due to
   roundoff error
mprintf("If=|Ef|/ma =%fA\n",abs(Ef)/ma1)//ans may
   vary due to roundoff error
Voc=2960 //line to line volatge in Volts
mprintf("V1oc=%fV\n",Voc/sqrt(3))//ans may vary due
   to roundoff error
If=240 //current in amperes
Efmax=ma1*If
mprintf("Efmax=%dV\n",Efmax)//ans in textbook is
   wrong
I1max=(Efmax-V1)/xda //ans in textbook is wrong
mprintf("I1max=%fA\n",I1max)//ans may vary due to
   roundoff error
mprintf("Qmax=%fMVAR",sqrt(3)*VLB*I1max*(10^-6))//
   ans may vary due to roundoff error
```

Scilab code Exa 2.10 Find initial current and current at the end of two cycles

```
clc
xd=1 //in ohms per unit
xd1=0.3 //in ohms per unit
xd2=0.2 //in ohms per unit
Td2=0.03 //time in seconds
Td1=1 //time in seconds
MVA=100 //rating in mega volt ampere
V=16000 //voltage in volts
I2pu=1/xd2
mprintf("Ipu =%dper unit\n",I2pu)
```

```
Ib=(MVA*(10^6))/(sqrt(3)*V)
mprintf("Ib=%fA\n",Ib)//ans may vary due to roundoff error
mprintf(" I    =%fA\n",I2pu*Ib)//ans in textbook is wrong
I1=1/xd1 //current in per unit
mprintf(" I  =Efo/xd =%fper unit\n",I1)//ans may vary due to roundoff error
Iss=1/xd//current in per unit
mprintf("Iss=Efo/xd=1 per unit\n")
t=2/60 //time in seconds
mprintf("I=%fper unit\n",(I2pu-I1)*exp(-t/Td2)+(I1-Iss)*exp(-t/Td1)+1)//ans may vary due to roundoff error
t=10 //time in seconds
mprintf("I=%fper unit\n",(I2pu-I1)*exp(-t/Td2)+(I1-Iss)*exp(-t/Td1)+1)//ans may vary due to roundoff error
```

Chapter 3

Transformers

Scilab code Exa 3.1 1Voltage to be applied to the transformer to result in rated v

```
clc
//code uses userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
r2=0.02;
X11=20 //in ohm
x1=0.05 //in ohm
X22=2000 //in ohm
x2=5 //in ohm
Xm1=X11-x1
Xm2=X22-x2
mprintf("Xm1=X11-x1=%f \n",Xm1)
```

```
mprintf("Xm2=X22—x2=%f \n",Xm2)
X12=sqrt(Xm1*Xm2)
mprintf("X12=sqrt(Xm1*Xm2)=%f \n",X12)//ans may vary due to roundoff error
kVA=10 //rated kVA
V2=1000 //secondary voltage in volts
I2=(kVA*(10^3))/V2 //rated current
mprintf("I2=ratedkVA*1000/raated V2=%dA\n",I2)
Zl=V2/I2 //load impedence
I1=((Zl+r2+(%i*X22))*I2)/(%i*X12)//ans may vary due to roundoff error
disp('I1=(Zl+r2+jwL22)*I2/wL12*I1='+complexstring(I1)+'A')
r1=0.01 //in ohm
V1=((r1+(%i*X11))*I1)-(%i*X12*I2)
disp('V1=(r1+jwL11)I1-jwL12I2='+complexstring(V1)+'V')//ans may vary due to roundoff error
k1=Xm1/X11
k2=Xm2/X22
mprintf("k1=%f\n",k1)
mprintf("k2=%f\n",k2)
k=sqrt(k1*k2)
mprintf("k=sqrt(k1*k2)=%f\n",k)
```

Scilab code Exa 3.2 1Find the turns ratio a 2primary and secondary induced voltage

```
clc
//code contains user defined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))
```

```

    end
    funcprot(0)
endfunction
i2=141.4 //load current max val in amperes
r2=0.02 //secondary resistance in ohms
V2=707/sqrt(2)
pha=-30 //phase angle of load current with reference with reference voltage in degrees
I2=(i2/sqrt(2))*exp(%i*pha*3.14/180)//ans may vary due to roundoff error,conversion of degrees in radian for calculation
mprintf("V2=%dV\n",V2)
disp('I2='+complexstring(I2)+'A')
disp('Secondary drop I2r2 is='+complexstring(I2*r2)+'V')//ans may vary due to roundoff error
L12=3*(10^(-4))//secondary leakage inductance in henry
w=377 //angular frequency of the supply in rad/sec
x2=w*L12 //secondary leakage reactance
mprintf("x2=%f \n",x2)
E12=(I2*%i*x2)//ans may vary due to roundoff error
disp('-E12=I2jx2='+complexstring(E12)+'V')
E2=V2+(r2+(%i*x2))*I2//ans may vary due to roundoff error
disp('E2='+complexstring(E2)+'V')
N1=300//primary winding turns
N2=30 //secondary turns
a=N1/N2
mprintf("a=N1/N2=%d\n",a)
E1=a*E2//ans may vary due to roundoff error
disp('E1=aE2='+complexstring(E1)+'V')
Iex1=0.707 //magnitude of exciting current of transformer in amperes
paex=-80 //phase angle of exciting current in degrees with reference voltage
Iex=(Iex1/sqrt(2))*exp(%i*paex*3.14/180)//ans may vary due to roundoff error,conversion of degrees
```

```
to radians for calculation
I1=(I2/a)+Iex//ans may vary due to roundoff error
disp('I1='+complexstring(I1)+'A')
mprintf("Actual ratio=I2/I1=%f\n",abs(I2)/abs(I1))//ans may vary due to roundoff error
L11=0.03 //leakage inductance of primary in henry
E11=%i*w*L11*I1//ans may vary due to roundoff error
disp('E11=jwL11I1='+complexstring(E11)+'V')
r1=2 //primary winding resistance in ohms
I1r1=I1*r1//ans may vary due to roundoff error
disp('I1r2='+complexstring(I1r1)+'V')
V1=E1+I1r1+E11//ans may vary due to roundoff error
disp('V1=E1+I1r2+E11='+complexstring(V1)+'V')
mprintf("Actual voltage ratio is V1/V2=%f\n",abs(V1)/abs(V2))//ans may vary due to roundoff error
```

Scilab code Exa 3.3 1Find the secondary current 2primary current and impedance see

```
clc
//the code uses userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
E1=2400 //primary voltage rating in volts
E2=240 //secondary voltage rating in volts
z=2 //magnitude of impedance connected to secondary terminals in ohms
```

```
pha1=36.9 //phase angle of impedance connected with reference in degrees
a=E1/E2
mprintf("a=%d\n",a)
V1=2200 // applied primary voltage to transformer in volts
V2=V1/a
mprintf("|V2|=|V1|/a=%dV\n",V2)
I2=V2/(z*exp(pha1*%i*3.14/180))//ans in textbook is wrong,conversion of degree to radian for calculation
disp('I2='+complexstring(I2)+'A')
I1=I2/a //ans may vary due to roundoff error
disp('I1=I2/a='+complexstring(I1)+'A')
Zin=V1/I1
disp('Zin=V1/I1='+complexstring(Zin)+'  ')
S2=V2*I2
pf=0.8 //power factor of load
mprintf("|S2|=|V2||I2|=%fkVA\n",(abs(V2)*abs(I2))/1000)
mprintf("P2=|S2|* cos  2=%fkW\n",(abs(S2)*pf)/1000)
mprintf("|S1|=|V2||I1|=%fkVA\n",(abs(V1)*abs(I1))/1000)
mprintf("P1=|S1| cos  1=%fkW\n",((abs(V1)*abs(I1))*cos(pha1*3.14/180))/1000)//ans may vary due to roundoff error,conversion of degree to radian for calculation
```

Scilab code Exa 3.4 Find turns ratio of transformer and voltage levels on applicat

```
clc
Z=4 //impedance of loudspeaker in ohms
Zin=500 //impedance of audio line in ohms
a=sqrt(Zin/Z)//ans may vary due to roundoff error
mprintf("a=sqrt(Zin/Z)=%f\n",a)//ans may vary due to
```

```
    roundoff error
P2=10 //audio power in watts
V2=sqrt(40)//ans may vary due to roundoff error
mprintf("V2=4*P2=%fV\n",V2) //ans may vary due to
   roundoff error
V1=a*V2
mprintf("V1=aV2=%fV\n",V1)
```

Scilab code Exa 3.5 Find core and copper losses and efficiency

```
clc
//code uses a userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
V2=120 //reference voltage in volts
kVA=16.67*(10^3) //kVA rating of transformer
I2=kVA/V2 //secondaary current aat unity pf
mprintf("I2=kVA/V2=%fA\n",I2)//ans may be wrong due
   to roundoff error
r2=0.00519 //secondary winding resistance in ohms
x2=0.0216  //secondary winding reactance in ohms
a=7200/120
E2=V2+(I2*(r2+(%i*x2)))//secondary induced voltage
   //ans may be wrong due to roundoff error
disp('E2=V2+I2(r2+jx2)='+complexstring(E2)+'V')
E1=a*E2//ans may be wrong due to roundoff error
```

```
disp('E1='+complexstring(E1)+'V')
Rc=311000
Ihe=E1/Rc
disp('core loss current='+complexstring(Ihe)+'A')
Phe=((abs(Ihe))^2)*Rc//ans may be wrong due to roundoff error
mprintf("Core loss Ph+e=|Ih+e|^2*Rc=%fW\n",Phe)
Xm=54800
disp('I  =E1/jXm='+complexstring(E1/(%i*Xm))+'A')//ans may be wrong due to roundoff error
Iex=Ihe+(E1/(%i*Xm))
disp('Iex=Ih+e+I  ='+complexstring(Iex)+'A')//ans may be wrong due to roundoff error
I1=Iex+(I2/a)
disp('I1=Iex+I2/a='+complexstring(I1)+'A')//ans may be wrong due to roundoff error
r1=18.7 //primary side resistaance
x1=77.8
V1=E1+(I1*(r1+(%i*x1)))
disp('V1=E1+I1(r1+jx1)='+complexstring(V1)+'V')//ans in the textbook is wrong
Pcu=(((abs(I1))^2)*r1)+(((abs(I2))^2)*r2)//copper loss
mprintf("Pcu=%fW\n",Pcu)//ans may be wrong due to roundoff error
mprintf("Efficiency =output watts/output+losses=%f\n",16670/(16670+Pcu+Phe))//ans may be wrong due to roundoff error
```

Scilab code Exa 3.6 Find the primary terminal voltage

```
clc
//extension of Ex3_1
//uses a userdefined function complexstring
function s=complexstring(a)
```

```
    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
I2=10
V2=1000
r2=1
X11=20 //in ohm
x1=0.05 //in ohm
X22=2000 //in ohm
x2=5 //in ohm
Xm1=X11-x1
Xm2=X22-x2
X12=sqrt(Xm1*Xm2)
V12=V2+I2*(r2+(%i*(X22-X12)))//ans may vary due to
    roundof error
disp('V12='+complexstring(V12)+'V')
I1=I2+(V12/(%i*X12))//ans may vary due to roundof
    error
disp('I1='+complexstring(I1)+'A')
r1=0.01
V1=V12+(I1*(r1+(%i*(X11-X12))))//ans may vary due to
     roundof error
disp('V1='+complexstring(V1)+'V')
a=0.1
Zeq1=r1+(a*a*r2)+(%i*(x1+(a*a*x2)))//ans may vary
    due to roundof error
disp('Zeq1='+complexstring(Zeq1)+'  ')
V1=(a*V2)+(I2^Zeq1/a)//ans may vary due to roundof
    error
disp('V1='+complexstring(V1)+'V')
```

Scilab code Exa 3.7 Find the primary voltage required to induce 240 volts at secon

```
clc
//the code uses a userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
r1=3
r2=0.03
x1=15
x2=0.15
V1B=2400 //primary side voltage
V2B=240 //secondary side voltage
a=V1B/V2B
Zeq2=(r1/(a^2))+r2+(%i*((x1/(a^2))+x2))//ans may vary due to roundoff error
disp('Zeq2='+complexstring(Zeq2)+'  ')
SB=10000// rated kva of the trnsformer
V2B=240
I2B=SB/V2B
mprintf("I2B=%fA\n",I2B)//ans may vary due to roundoff error
//with V2 reference
//0.8 pf lagging
I2=I2B*exp(%i*(-1)*acos(0.8))//ans may vary due to roundoff error
```

```
disp('I2='+complexstring(I2)+'A')
V2=240
V1=a*(V2+I2*Zeq2)//ans may vary due to roundoff error
disp('V1/a='+complexstring(V1/a)+'V')
mprintf("|V1|=%fV\n",abs(V1))
//0.8 pf leading
I2B=SB/V2B
I2=I2B*exp(%i*acos(0.8))//ans may vary due to roundoff error
V1=a*(V2+(I2*Zeq2))//ans may vary due to roundoff error
disp('V1='+complexstring(V1/a)+'V')
mprintf("|V1|=%fV\n",abs(V1))//ans may vary due to roundoff error
```

Scilab code Exa 3.8 Find the full voltage regulation at zero point eight pf laggin

```
clc
//example below is an extension of Ex3_7
//values below from Ex3_7
V2B=240 //secndary side voltage
a=10
//0.8 pf lagging
V1=2496.44
V=V1/a //secondary voltage at full load
mprintf("|V1/a|=%fV\n",V)
Regulation=(V-V2B)/V2B //ans may vary due to roundoff error
mprintf("Regulation=(|V1/a|-V2B)/V2B=%f\n",Regulation)
//0.8 pf leading
V1=2347.8
V=V1/a
mprintf("V at 0.8 pf leading=%fV\n",V)
```

```
mprintf("Regulation=%f\n",(V-V2B)/V2B)
```

Scilab code Exa 3.9 Find voltage regulation at full load and zero point eight pf l

```
clc
//code uses usedefined function
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
pf=0.8 //power factor of full load
I2=1 //magnitude of load current in amperes in per unit system
I2pu=I2*exp(%i*(-1)*acos(pf))//−1 comes due to lagging power factor
disp('I2pu='+complexstring(I2pu)+'A')
pres=2 //percent resistance in ohms
preact=5 //percent reactance in ohms
Zeqpu=(pres/100)+(%i*(preact/100))
disp('Zeqpu='+complexstring(Zeqpu)+'  ')
V1pu=1+(I2pu*Zeqpu)
disp('V1pu='+complexstring(V1pu)+'V')
Regulation=abs(V1pu)-1
mprintf("|V1pu|−1=%f\n",Regulation)
```

Scilab code Exa 3.10 Find the ratings and turns ratio of a three phase transformer

```
clc
SB=10000000 //rating of transformer
VL1B=230000 //voltage rating
IL1B=SB/(sqrt(3)*VL1B)
mprintf("ILIB=%fA\n",IL1B)
VL2B=4160
IL2B=SB/(sqrt(3)*VL2B)
mprintf("IL2B=%fA\n",IL2B)
//star delta connected
mprintf("Rated kVA=SB/1000=%fkVA\n",SB/1000)
mprintf("Rated 11=I1B=ILIB=%fA\n",IL1B)
mprintf("Rated I2=I2B=IL2B/sqrt(3)=%fA\n",IL2B/sqrt
   (3))
VL1=230 //rating in kV
VL2=4160//rating in kV
mprintf("Rated V1=V1B=VL1/sqrt(3)=%fkV\n",VL1/sqrt
   (3))
mprintf("V2=V2B=%fV\n",VL2)
mprintf("turns ratio=V1B/V2B=%f\n",(VL1*1000)/(VL2*
   sqrt(3)))
mprintf("kVA per phase=%dkVA\n",3333)
//delta star connected
mprintf("Rated kVA=%fkVA\n",SB/1000)
mprintf("kVa per phase=%dkVA\n",3333)
mprintf("V1B=VL1B=%fkV\n",VL1)
mprintf("V2B=VL2B/sqrt(3)=%fV\n",VL2/sqrt(3))
mprintf("I1B=IL1B/sqrt(3)=%fA\n",IL1B/sqrt(3))
mprintf("I2B=IL2B=%fA\n",IL2B)
mprintf("a=V1B/V2B=%f\n",(VL1B*sqrt(3))/VL2B)

//delta delta connected
mprintf("Rated kVA=%fkVA\n",SB/1000)
mprintf("kVA per phase=%dkVA\n",3333)
mprintf("V1B=%fKV\n",VL1B/1000)
mprintf("V2B=%fV\n",VL2B)
mprintf("I1B=%fA\n",IL1B/sqrt(3))
mprintf("IL2B=%fA\n",IL2B/sqrt(3))
mprintf("a=%f\n",VL1B/VL2B)
```

Scilab code Exa 3.11 Find the voltage regulation at full load and zero point eight

```
clc
//the code uses userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
//delta connected
// sol 1
V1B=7200 //primary voltage in volts
VL1B=7200 //primary voltage in volts
kVA=50 //kva rating
IL1B=(kVA*1000)/((sqrt(3))*VL1B)//ans may vary due
   to roundoff error
mprintf("IL1B=%fA\n",IL1B)
I1B=IL1B/sqrt(3)//ans may vary due to roundoff error
mprintf("I1B=%fA\n",I1B)
//star connected
VL2B=208 //seconadry voltage in volts
V2B=VL2B/sqrt(3)//ans may vary due to roundoff error
mprintf("V2B=%fV\n",VL2B/sqrt(3))
IL2B=(kVA*1000)/(sqrt(3)*VL2B)//ans may vary due to
   roundoff error
mprintf("IL2B=%fA\n",IL2B)
I2B=IL2B
a=V1B/V2B//ans may vary due to roundoff error
```

```
mprintf("a=%f\n",a)
Z2B=V2B/I2B//ans may vary due to roundoff error
mprintf("Z2B=V2B/I2B=%f \n",Z2B)
Reqpu=0.012 //percent resistance in ohms
Xeqpu=0.05 //percent reactance in ohms
Zeqpu=Reqpu+(%i*Xeqpu)
mprintf("Zeqpu=%f     with phase angle of %f degrees \n",abs(Zeqpu),(acos(Reqpu/(abs(Zeqpu))))*180/%pi)//ans may vary due to roundoff error,conversion of radians to degree
Zeq2=Z2B*Zeqpu//ans may vary due to roundoff error
mprintf("Zeq2=%f  with a phase angle of %f degrees\n",abs(Zeq2),(acos(real(Zeq2)/abs(Zeq2)))*180/%pi)//ans may vary due to roundoff error,conversion of radians to degree
pf=0.8 //power factor of load
I2=IL2B*exp(%i*(-1)*acos(pf))//ans may vary due to roundoff error,-1 comes due to the lagging power factor
mprintf("I2=%fA with a phase angle of %f degress\n",abs(I2),(-1)*(acos(real(I2)/abs(I2)))*180/%pi)//ans may vary due to roundoff error,conversion of radians to degree
V2=120 //seconadry voltage in volts
V1=a*(V2+(I2*Zeq2))//ans may vary due to roundoff error
mprintf("V1=%fV with a phase angle of %f degrees\n",abs(V1/a),(acos(real(V1)/abs(V1)))*180/%pi)//ans may vary due to roundoff error,conversion of radians to degree
Regulation=(abs(V1/a)-V2)/V2//ans may vary due to roundoff error
mprintf("Regulation=%f\n",Regulation)
//sol 2(per unit method)
I2pu=exp(%i*(-1)*acos(pf)) //seconadry current in per unit in amperes
V2pu=1 //seconadry voltage in per unit in volts
V1pu=V2pu+(I2pu*Zeqpu)
```

```
mprintf("V1pu=%fV with a phase angle of %f degrees\n
    ",abs(V1pu),(acos(real(V1pu)/abs(V1pu))*180/%pi)
    //ans may vary due to roundoff error
Regulation=(abs(V1/(a*V2B))-(V2B/V2B))/(V2B/V2B)
mprintf("Regulation=%f\n",Regulation)//ans may vary
    due to roundoff error
```

Scilab code Exa 3.12 Find Magnitude of current in secondary of each transfomer

```
clc
//the code uses a userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
kVAL=100 //kva required for supply
kVAM=20 //kVA rating of motor of the air
    conditioning compressor
V=220 //supply voltage in volts
IL3=(kVAM*1000)/(sqrt(3)*V)
mprintf("IL3=%fA=|I2| of small transformer\n",IL3)
//abc sequence
ph1=36.9 //phase angle of motor current
IL3=IL3*exp(%i*(-1)*ph1*%pi/180)//-1 comes due to
    the lagging power factor,conversion of degree to
    radian for calculation
mprintf("Il3=%fA with a phase angle of %f degrees\n"
    ,abs(IL3),(-1)*ph1)//-1 comes due to the lagging
```

```
    power factor
disp('IL3='+complexstring(IL3)+'A')
ph2=30-25.8 //phase angle of Il1
IL1=((kVAL*1000)/V)*exp(%i*(ph2)*%pi/180)
disp('IL1='+complexstring(IL1)+'A')
mprintf("IL1=%f with a phase angle of %f degrees\n",abs(IL1),ph2)
I2=IL3+IL1
disp('I2='+complexstring(I2)+'A')
mprintf("I2=%fA with a phase angle of %f degrees\n",abs(I2),(acos(real(I2)/abs(I2)))*180/%pi)
//acb sequence
ph3=30+25.8 //phase angle of Il1 in degrees
IL1=abs(IL1)*exp(%i*(-1)*(ph3)*%pi/180) //-1 comes due to lagging power factor
disp('IL1='+complexstring(IL1)+'A')
mprintf("IL1=%f with a phase angle of %f degrees\n",abs(IL1),(-1)*ph3)//-1 comes due to the lagging power factor
I2=IL3+IL1
disp('I2='+complexstring(I2)+'A')
mprintf("I2=%fA with a phase angle of %f degrees\n",abs(I2),(acos(real(I2)/abs(I2)))*180/%pi)
```

Scilab code Exa 3.13 Find efficiency of transformer for a power factor of zero poi

```
clc
SB=300 //rating in kVA at full load
S=150 //kVA at half load
Phe=2.7 //core loss in kW
Phepu=Phe/SB //ans may vary due to roundoff error
mprintf("Phepu=%f\n",Phepu)
Reqpu=0.0140 //per unit resistance in ohms=per unit copper loss at full load in watts
pf=0.9 //power factor at full load
```

```
//efficiency at full load
mprintf(" fl =%f\n",pf/(pf+Phepu+Reqpu))//ans may vary due to roundoff error
//efficiency at half load
a=S/SB //ratio of kVA at half and full load
mprintf(" fl =%f\n",(a*pf)/((a*pf)+Phepu+(a*a*Reqpu)))//ans may vary due to roundoff error
//for max efficiency
mprintf("|S|/SB=sqrt(Phepu/Reqpu)=%fA\n",sqrt(Phepu/Reqpu))//ans may vary due to roundoff error
```

Scilab code Exa 3.14 Open circuit and short circuit test on transformer

```
clc
//open ckt short ckt test
//code uses userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
kVA=50 //kVA rating
Poc=500 //core loss in watts
Voc=208 //open ckt voltage in volts
Vphioc=Voc/sqrt(3)
mprintf("V oc=Voc/sqrt(3)=%fV\n",Vphioc)//ans may vary due to roundoff error
Pphioc=Poc/3
mprintf("P oc=Poc/3=%fW\n",Pphioc)//ans may vary
```

```
due to roundoff error
Ioc=8 //open ckt current in amperes
mprintf("RcLV=V  *V  / P  =%f \n",(Vphioc*Vphioc)/Pphioc)//ans may vary due to roundoff error
mprintf("Voc^2/Poc=%f \n",(Voc^2)/Poc)//ans may vary due to roundoff error
mprintf("sin oc=%f\n",sin(acos(Poc/(sqrt(3)*Ioc*Voc))))//ans may vary due to roundoff error
mprintf("I  =I oc *sin oc=%fA\n",Ioc*sin(acos(Poc/(sqrt(3)*Ioc*Voc))))//ans may vary due to roundoff error
mprintf("XmLV=V oc/I  =%f \n",(Voc/sqrt(3))/(Ioc*sin(acos(Poc/(sqrt(3)*Ioc*Voc)))))//ans may vary due to roundoff error
//short ckt
Psc=600 //copper loss in watts
Isc=4.011 //short circuit current in amperes
Vsc=370 //short circuit voltage in volts
ReqHV=(Psc/3)/((Isc/sqrt(3))^2)
mprintf("ReqHV=P sc/I sc^2=%f \n",ReqHV)//ans may vary due to roundoff error
ZeqHV=Vsc/(Isc/sqrt(3))
mprintf("|ZeqHV|=V sc/I sc=%f \n",ZeqHV)//ans may vary due to roundoff error
XeqHV=sqrt((ZeqHV^2)-(ReqHV^2))
mprintf("XeqHV=%f \n",XeqHV)//ans may vary due to roundoff error
VHVB=7200//secondary side voltage in volts
VLVB=208/sqrt(3)//primary side voltage in volts
aV=VHVB/VLVB
mprintf("NHV/NLV=VHVB/VLVB=%f\n",aV)//ans may vary due to roundoff error
mprintf("RcHV=RcLV*aV*aV=%f \n",((Vphioc*Vphioc)/Pphioc)*aV*aV)//ans in the textbook is wrong
mprintf("XmHV=XmLV*aV*aV=%f \n",(Voc/sqrt(3))/(Ioc*sin(acos(Poc/(sqrt(3)*Ioc*Voc))))*aV*aV)//ans in the textbook is wrong
ZeqLV=(ReqHV+(%i*XeqHV))/(aV*aV)
```

```
disp('ZeqLV='+complexstring(ZeqLV)+'  ')//ans may vary due to roundoff error
mprintf("ZeqLV=%f ohms with a phase angle of %f degrees\n",abs(ZeqLV),(acos(real(ZeqLV)/abs(ZeqLV)))*180/%pi)
SB=50000 //rating of transformer
ZLVB=(Voc*Voc)/SB
mprintf("ZLVB=%f \n",ZLVB)//ans may vary due to roundoff error
Reqpu=(ReqHV/(aV*aV))/ZLVB
mprintf("Reqpu=%f \n",Reqpu)//ans may vary due to roundoff error
Xeqpu=(XeqHV/(aV*aV))/ZLVB
mprintf("Xeqpu=%f \n",Xeqpu)//ans may vary due to roundoff error
Zeqpu=Reqpu+(%i*Xeqpu)
disp('Zeqpu='+complexstring(Zeqpu)+'  ')//ans may vary due to roundoff error
mprintf("Zeqpu=%fohms with a pgase angle of %f degrees\n",abs(Zeqpu),(acos(real(Zeqpu)/abs(Zeqpu)))*180/%pi)
V1pu=1+((exp(%i*(-1)*acos(0.8)))*Zeqpu)
disp('V1pu='+complexstring(V1pu))//ans may vary due to roundoff error
mprintf("V1pu=%fV with a phase angle of %f degrees\n",abs(V1pu),(acos(real(V1pu)/abs(V1pu)))*180/%pi)
mprintf("Regulation=%f\n",(abs(V1pu)-1))//ans may vary due to roundoff error
//full load efficiency
pf=0.8 //power factor of load
Phepu=Poc/SB
mprintf("  =cos  / cos  +Reqpu+Phepu=%f\n",pf/(pf+Reqpu+Phepu))//ans may vary due to roundoff error
//second method
mprintf("  =%f\n",(SB*pf)/((SB*pf)+Poc+Psc))
//ans may vary due to roundoff error
```

Chapter 4

Induction or Asynchronous Machines

Scilab code Exa 4.1 Calculate air gap power and developed mechanical power and out

```
clc
SCL=1000 //stator copper loss in watts
V=460 //line voltage of induction motor in volts
I=25 //line current of motor in amperes
pf=0.85 //power factor of motor
Pin=sqrt(3)*V*I*pf //ans  may vary due to roundoff
   error
mprintf("Pin=%fW\n",Pin)
Pg=Pin-SCL //air gap power
mprintf("Pg=%fW\n",Pg)//ans  may vary due to
   roundoff error
RCL=500 //rotor copper loss in watts
Phe=800 //core loss in watts
Pfw=250 //winding and friction loss in Watts
PLL=200 //stray load loss in watts
DMP=Pg-RCL ///developed mechanical power in watts
mprintf("DMP=%fW\n",DMP)//ans  may vary due to
   roundoff error
Prot=Phe+Pfw+PLL //power loss in rotor in watts
```

```
Pout=DMP-Prot
mprintf("Pout=DMP-Prot=%fW\n",Pout)//ans  may vary due to roundoff error
mprintf("Horsepower=Pout/746=%fhp\n",Pout/746)//ans may vary due to roundoff error,conversion of watts to hp needs division by 746
mprintf("  =Pout/Pin=%f\n",Pout/Pin)//ans  may vary due to roundoff error
```

Scilab code Exa 4.2 Find slip and operating speed and developed torque and output

```
clc
//this is an extension of Ex4_1
//following comes from Ex4_1
SCL=1000 //stator copper loss in watts
V=460 //line voltage of induction motor in volts
I=25 //line current of motor in amperes
pf=0.85 //power factor of motor
Pin=sqrt(3)*V*I*pf //ans  may vary due to roundoff error
Pg=Pin-SCL //air gap power
RCL=500 //rotor copper loss in watts
Phe=800 //core loss in watts
Pfw=250 //winding and friction loss in Watts
PLL=200 //stray load loss in watts
DMP=Pg-RCL ///developed mechanical power in watts
Prot=Phe+Pfw+PLL //power loss in rotor in watts
Pout=DMP-Prot
//above is from Ex4_1
s=RCL/Pg
p=4 //no of poles
mprintf("s=RCL/Pg=%f\n",s)//ans may vary due to roundoff error
ws=(4*%pi*60)/p //synchronous angular frequency
mprintf("ws=%frad/s\n",ws)//ans may vary due to
```

```
   roundoff error
ns=(120*60)/p
mprintf("ns=%drev/min\n",ns)//ans may vary due to roundoff error
w=ws*(1-s)
n=ns*(1-s)
mprintf("w=ws(1-s)=%frad/s\n",w)//ans may vary due to roundoff error
mprintf("n=ns(1-s)=%frev/min\n",n)//ans may vary due to roundoff error
mprintf(" d =DMP/w=%fN-m\n",DMP/w)//ans may vary due to roundoff error
mprintf("  =Pout/w=%fN-m\n",Pout/w)//ans may vary due to roundoff error
```

Scilab code Exa 4.3 Calculate performance at 1746 rev per min and starting current

```
clc
//code uses userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction

//induction machine parameters in ohms
r1=0.39 //primary resistance
r2=0.14 //secondary resistance
x1=0.35//primary reactance
```

```
x2=0.35//secondary reactance
Xm=16//manetizing reactance
VL=220 //supply volatge in volts
f=60 //frequency in Hz
//part a
p=4 //no of poles
ns=(120*f)/p
mprintf("ns=%drev/min\n",ns)
n=1746 //runnimg speed of motor in rev/min
s=(ns-n)/ns
mprintf("s=%f\n",s)
z2=(r2/s)+(%i*x2)//ans may vary due to roundoff error
disp('Z2='+complexstring(z2)+'  ')
mprintf("Z2=%fohm having a phase angle of %f degrees\n",abs(z2),(acos(real(z2)/abs(z2)))*180/%pi)
Zf=(%i*Xm*z2)/(z2+(%i*Xm))//ans may vary due to roundoff error
disp('Zf='+complexstring(Zf)+'  ')
mprintf("Zf=%fohms having a phase angle of %f degrees\n",abs(Zf),(acos(real(Zf)/abs(Zf)))*180/%pi)
Rf=real(Zf)//ans may vary due to roundoff error
mprintf("Rf=%f \n",Rf)
Zin=r1+(%i*x1)+Zf//ans may vary due to roundoff error
disp('Zin=r1+jx1+Zf='+complexstring(Zin)+'  ')
mprintf("Zin=%fohms having a phase angle of %f degrees\n",abs(Zin),(acos(real(Zin)/abs(Zin)))*180/%pi)
Powerfctor=real(Zin)/abs(Zin)//ans may vary due to roundoff error
mprintf("Power facto=%f\n",Powerfctor)
I1=VL/(sqrt(3)*abs(Zin))
mprintf("|I1|=%fA\n",I1)//ans may vary due to roundoff error
Pin=sqrt(3)*I1*VL*Powerfctor
mprintf("Pin=%fW\n",Pin)//ans in the textbook is
```

```
wrong
Pg=3*I1*I1*Rf
mprintf("Pg=%fW\n",Pg)//ans in the textbook is wrong
DMP=(1-s)*Pg
mprintf("Developed power=(1-s)Pg=%fW\n",DMP)//ans in the textbook is wrong
Prot=s*Pg //rotor copper losses
Pout=DMP-Prot//ans in the textbook is wrong
mprintf("Output power=%fW\n",Pout)
mprintf("Output horsepower=%f\n",Pout/746)//ans may vary due to roundoff error,1 hp=746 watts
mprintf("Developed torque=%flb-ft\n",7.04*(Pg/ns))//ans may vary due to roundoff error,1 N-m=7.04 lb-ft ot torque
n=(1-s)*ns//ans may vary due to roundoff error
mprintf("Output torque=%flb-ft\n",7.04*(Pout/n))
mprintf("Efficiency=%f\n",Pout/Pin)
//part b
s=1 //machine at stanstill
z2=r2+(%i*x2)//ans may vary due to roundoff error
disp('Z2='+complexstring(z2)+'  ')
mprintf("Z2=%fohm having a phase angle of %f degrees\n",abs(z2),(acos(real(z2)/abs(z2)))*180/%pi)
Zf=(%i*Xm*z2)/(z2+(%i*Xm))//ans may vary due to roundoff error
disp('Zf='+complexstring(Zf)+'  ')
mprintf("Zf=%fohms having a phase angle of %f degrees\n",abs(Zf),(acos(real(Zf)/abs(Zf)))*180/%pi)
Zin=r1+(%i*x1)+Zf//ans may vary due to roundoff error
disp('Zin='+complexstring(Zin)+'  ')
mprintf("Zin=%fohms having a phase angle of %f degrees\n",abs(Zin),(acos(real(Zin)/abs(Zin)))*180/%pi)
I1=VL/(sqrt(3)*abs(Zin))//ans may vary due to roundoff error
Rf=real(Zf)
```

```
mprintf("Starting current=%fA\n",I1)
Pg=3*I1*I1*Rf
mprintf("Pg=%fW\n",Pg)//ans in the textbook is wrong
mprintf(" d =7.04*(Pg/ns)=%flb-ft\n",7.04*(Pg/ns))//
    ans may vary due to roundoff error,1 N-M=7.04 lb-
    ft of torque
```

Scilab code Exa 4.4 Find the pull out torque and slip at which it occurs

```
clc
//below is an extension of Ex4_3
//code uses userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
x1=0.35 //primary reactance in ohms
r1=0.39 //primary resistance in ohms
Xm=16 //magnetizing reactance
r2=0.14 //secondary resistance in ohms
x2=0.35 //secondary reactance in ohms
ws=188.5 //angular frequency in rad/sec
V=220 //rated voltage in volts
//part a
V1m=V/sqrt(3)//ans may vary due to roundoff error
VTH=V1m*(Xm/(Xm+x2))
mprintf("VTH=V1m=%fV\n",VTH)//ans may vary due to
    roundoff error
```

```
X1=x1
mprintf("X1=%f \n",X1)
R1=r1*(Xm/(x1+Xm))//ans may vary due to roundoff error
mprintf("R1=%f \n",R1)
mprintf(" max =%fN−m\n",((3/ws)*(VTH^2))/(2*(R1+sqrt((R1^2)+((2*X1)^2)))))//ans may vary due to roundoff error
//part b
sM=r2/sqrt((R1^2)+((X1+x1)^2))//ans may vary due to roundoff error
mprintf("sM=%f\n",sM)
mprintf("r2/sM=%f \n",r2/sM)//ans may vary due to roundoff error
Zf=((%i*Xm)*((r2/sM)+(%i*x2)))/((r2/sM)+(%i*(x2+Xm)))//ans may vary due to roundoff error
disp('Zf='+complexstring(Zf)+'  ')
mprintf("Zf=%fohm having a phase angle of %f degrees\n",abs(Zf),(acos(real(Zf)/abs(Zf)))*180/%pi)
z1=r1+(%i*x1)
Zin=z1+Zf
disp('Zin='+complexstring(Zin)+'  ')//ans may vary due to roundoff error
mprintf("Zin=%fohm having a phase angle of %f degrees\n",abs(Zin),(acos(real(Zin)/abs(Zin)))*180/%pi)
I1=V1m/abs(Zin)
mprintf("I1=%fA\n",I1)//ans may vary due to roundoff error
Rf=real(Zf) //resistance in ohms
Pg=3*I1*I1*Rf//ans  in the textbook is wrong
mprintf("Pg=%fW\n",Pg)
mprintf(" max =Pg/ws=%fN−m\n",Pg/ws)//ans may vary due to roundoff error
```

Scilab code Exa 4.5 Find speed at half of the load and the corresponding outpt in

```
clc
ns=1800 //synchronous speed in rev/min
n=1745 //initial speed in rev/min
hp=10 //hp rating of the motor horsepower(1 hp=746 Watts)
s=(ns-n)/ns
mprintf("s=%f\n",s)//ans may vary due to roundoff error
s=s/2 //slip at half torque
n1=ns*(1-s)//ans may vary due to roundoff error
mprintf("n=ns(1-s)=%frev/min\n",n1)
//output at half torque
mprintf("New horsepower output=%fhp\n",(0.5*hp*n1)/n)//ans may vary due to roundoff error,0.5 factor comes due to half torque
```

Scilab code Exa 4.6 Will the mootor run hotter or cooler if voltage drops to 90 pe

```
clc
V1m(1)=1 //reference voltage in volts
V1m(2)=0.9//reduced voltage in volts
ratio=(V1m(1)/V1m(2))^2 //ratio of s2/s1
mprintf("s2/s1=%f\n",ratio)//ans may vary due to roundoff error
mprintf("I2(2)/I2(1)=s2*V1m(2)/s1*V1m(1)=%f\n",(V1m(2)/V1m(1))*ratio)//ans may vary due to roundoff error
mprintf("(copperloss)2/(copperloss)1=(I2(2)/I2(1))^2=%f\n",(V1m(1)/V1m(2))^2)//ans may vary due to roundoff error
s=0.03 //at 60Hz slip
ns=1800 //synchronous speed in rev/min
mprintf("Speed at 90 percent voltage=%frev/min\n",ns
```

```
*(1-(ratio*s)))//ans may vary due to roundoff
error
```

Scilab code Exa 4.7 Calculate performance of motor at given slip

```
clc
//code uses userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
//dc test
Vdc=13.8 //dc voltage in volts
Idc=13 //direct current in amperes
//no load test
Vnl=220 //applied no voltage in volts
f=60 //applied frequency in Hz
//blocked rotor test
VBR=23.5 //blocked rotor voltage in volts
f1=15 //frequency in Hz
Ia=12.8 //current of phase A
Ib=13.1 //current of phase B
Ic=12.9 //current of phase C
//from blocked rotor
IBR=(Ia+Ib+Ic)/3 //ans may vary due to roundoff
    error
mprintf("IBR=%fA\n",IBR)
ZBR=VBR/(sqrt(3)*IBR)
```

```
mprintf("|ZBR|= %f \n",ZBR)//ans may vary due to roundoff error
P1=179 //power in watts
P2=290 //power in watts
PBR=P1+P2
mprintf("PBR=%fW\n",PBR)
RBR=PBR/(3*(IBR^2))//ans may vary due to roundoff error
mprintf("RBR= %f \n",RBR)
mprintf(" BR =%f\n",(acos(PBR/(sqrt(3)*VBR*IBR)))*(180/%pi))//ans may vary due to roundoff error
mprintf("X'BR=|ZBR|*sin BR=%f \n",ZBR*sin(acos(PBR/(sqrt(3)*VBR*IBR))))//ans may vary due to roundoff error
XBR=(f/f1)*(ZBR*sin(acos(PBR/(sqrt(3)*VBR*IBR))))
mprintf("XBR=(fB/f1)*X'BR= %f \n",XBR)//ans may vary due to roundoff error
x1=0.4*XBR //designed reactance
x2=0.6*XBR //designed reactance
mprintf("x1= %f \n",x1)//ans may vary due to roundoff error
mprintf("x2= %f \n",x2)//ans may vary due to roundoff error
//from dc test
r1=0.5*(Vdc/Idc)
mprintf("r1= %f \n",r1)//ans may vary due to roundoff error
r2=RBR-r1
mprintf("r2= %f \n",r2)//ans may vary due to roundoff error
//from no load test
Ia=3.86 //current of phase A in amperes
Ib=3.86 //current of phase B in amperes
Ic=3.89 //current of phase C in amperes
Inl=(Ia+Ib+Ic)/3
mprintf("Inl=%fA\n",Inl)//ans may vary due to roundoff error
Znl=Vnl/(sqrt(3)*Inl)
```

```
mprintf("Znl=x1+Xm=%f \n",Znl)//ans may vary due to roundoff error
Xm=Znl-x1
mprintf("Xm=Znl-x1=%f \n",Xm)//ans may vary due to roundoff error
P1=550 //power in watts
P2=-350 //power in watts
Pnl=P1+P2
mprintf("Pnl=%fW\n",Pnl)//ans may vary due to roundoff error
Pfwc=Pnl-(3*Inl*Inl*r1)
mprintf("Pfwc=%fW\n",Pfwc)//ans may vary due to roundoff error
Prot=Pfwc
s=0.03
z2=(r2/s)+(%i*x2)
disp('z2='+complexstring(z2)+'  ')//ans may vary due to roundoff error
mprintf("Z2=%fohms with a phase angle of %fdegrees\n",abs(z2),(acos(real(z2)/abs(z2)))*180/%pi)
Zf=(z2*(%i*Xm))/(z2+(%i*Xm))
disp('Zf='+complexstring(Zf)+'  ')//ans may vary due to roundoff error
mprintf("Zf=%fohms with a phase angle of %fdegrees\n",abs(Zf),(acos(real(Zf)/abs(Zf)))*180/%pi)
Rf=real(Zf)
Zin=r1+Zf+(%i*x1)
disp('Zin='+complexstring(Zin)+'  ')//ans may vary due to roundoff error
mprintf("Zin=%fohms with a phase angle of %fdegrees\n",abs(Zin),(acos(real(Zin)/abs(Zin)))*180/%pi)
mprintf("power factor=%f\n",(real(Zin)/abs(Zin)))//ans may vary due to roundoff error
I1=Vnl/(sqrt(3)*abs(Zin))
mprintf("|I1|=%fA\n",I1)//ans may vary due to roundoff error
Pin=(sqrt(3)*(real(Zin)/abs(Zin))*I1*Vnl)//ans is wrong in textbook
```

```
mprintf("power drawn from line=sqrt(3)*VL*|I|*
    cos   =%fW\n",Pin)
Rf=real(Zf)
Pg=3*I1*I1*Rf
mprintf("Pg=%fW\n",Pg)//ans is wrong in textbook
DMP=Pg*(1-s)
mprintf("DMP=%fW\n",DMP)//ans is wrong in textbook
Pout=DMP-Prot
mprintf("output horsepower=%fhp\n",Pout/746)//ans
    may vary due to roundoff error ,1 hp=746 watts
mprintf("  =Pout/Pin=%f\n",Pout/Pin)//ans may vary
    due to roundoff error
```

Scilab code Exa 4.8 Upper limit of the starting current at 230v

```
clc
kVA=6.3 //upper limit for kVA per horsepower
hp=10 //rating of induction motor in hp.(1 hp=746
    watts)
V=230 //voltage rating of the motor
I=(kVA*hp*1000)/(sqrt(3)*V)
mprintf("I=%fA\n",I)//ans may vary due to roundoff
    error
```

Scilab code Exa 4.9 Find the starting line current and torque with provided compen

```
clc
vtap=0.8 //percantage voltage tap of compensator
hp=100 //rating of motor in horsepower,I hp=746
    watts
n=1750 //rated speed of motor in rev/min
a=1/vtap //compensator turns ratio
V=2300 //voltage rating of induction motor in volts
```

```
I1=150 //current rating in amperes
mprintf("a=%f\n",a)
mprintf("Voltage applied at starting of motor=%fV\n",V/a)
I1start=I1/a
mprintf("I1start=(1840/2300)*150A=150/a=%fA\n",I1start)
IL=I1start/a
mprintf("IL=I1start/a=%fA\n",IL)
tfl=hp*5252/n
mprintf(" fl =(hp*5252)/(rev/min)=%flb-ft\n",tfl)//ans may vary due to roundoff error
t=1.2*tfl //120 percent of the full load torque in lb-ft
mprintf(" st =360/a*a=%flb-ft\n",t/(a*a))//ans may vary due to roundoff error
```

Chapter 5

Direct Current Machines

Scilab code Exa 5.1 Find out the power developed and the torque and determination

```
clc
B=0.78 //flux density in tesla
A=200*(10^(-4))//cross sectional area in centimetre square
mprintf("Flux per pole   =B.A=%fWb\n",B*A)
C=95 //no of coils
Nc=2 //no of turns in each coil
Z=2*C*Nc
mprintf("Z=2*C*Nc=%fconductors\n",Z)
n=1200 //rotating speed in rev/min
w=(n/60)*(2*%pi)
mprintf("w=%frad/s\n",w)//ans may vary due to rounof error
a=2 //no of paths
p=4 //no of poles
Ka=(Z*p)/(2*%pi*a)
mprintf("Ka=%fV-s/Wb\n",Ka)//ans may vary due to rounof error
Eg=Ka*B*A*w
mprintf("Eg=Ka*  *w=%fV\n",Eg)//ans may vary due to rounof error
```

```
VT=250 //terminal voltage in volts
ra=0.2 //armture resistance in ohms
Ia=(VT-Eg)/ra
mprintf("Ia=%fA\n",Ia)//ans may vary due to rounof error
Pin=VT*Ia
mprintf("Pin=%fW\n",Pin)//ans in textbook is wrong
mprintf("Armature copper loss=%fW\n",((Ia*Ia)*ra))//ans in textbook is wrong
Pd=Pin-((Ia*Ia)*ra)//ans in textbook is wrong
mprintf("Pd=Pin-coper loss=%fW\n",Pd)
mprintf(" d =Pd/w=%fN-m ",Pd/w)
cf=0.7376 //conversion factor for conversion from N-m to lb-ft
mprintf("or %flb-ft",(Pd/w)*cf)//ans may vary due to roundoff error
```

Scilab code Exa 5.2 Find torque developed by the motor and speed and armature cure

```
clc
I=100 //current drawn in amperes
ra=0.07 //armature resistance in ohms
Vt=230 //terminal voltage of motor in volts
mprintf("Eg*=VT*-Iara*=%fV\n",Vt-(I*ra))
n=1200 //speed of rotation in rev/min
mprintf("w*=%d rad/sec\n",(n/60)*2)
mprintf(" Ka =Eg*/w*=%fV-s/rad\n",(Vt-(I*ra))/((n/60)*2*%pi))//ans may vary due to roundoff error
Ia=100 //armature current in ampere
mprintf(" d =Ka Ia=%fN-m\n",(Ia*(Vt-(I*ra))/((n/60)*2*%pi)))//ans may vary due to roundoff error
Td=300 //torque in N-m
Ia=Td/((Vt-(I*ra))/((n/60)*2*%pi))//ans may vary due to roundoff error
mprintf("Ia= d /Ka =%fA\n",Ia)
```

```
ra=0.07 //resistance in ohms
VT=230 //voltage in volts
w=(VT-Ia*ra)/((Vt-(I*ra))/((n/60)*2*%pi))
mprintf("w=(VT−Iara)/Ka =%frad/sec\n",w)//ans may vary due to roundoff error
```

Scilab code Exa 5.3 Find speed of the motor when it develops a torque of 300 Newto

```
clc
//Ex5_3 uses a magnetization curve given in textbook
mprintf("At 1200 rev/min and shunt field current of 0.7A Eg*=90V \n") //from magnetization curve
n=1200 //speed of rotation in rev/min
Eg1=90 //voltage in volts
wB=(n/60)*2
mprintf("wB=%d rad/sec\n",wB)
mprintf("Ka *=Eg*/wB=%fV−s/rad\n",Eg1/(wB*%pi))//ans may vary due to roundoff error
Td=30 //torque in N−m
Ia=Td/(Eg1/(wB*%pi))
mprintf("Ia= d /Ka *=%fA\n",Ia)//ans may vary due to roundoff error
VT=125 //voltage in volts
ra=0.2 //resistance in ohms
Eg=VT-(Ia*ra)
mprintf("Eg=%fV\n",Eg)//ans may vary due to roundoff error
w=Eg/((Eg1/(wB*%pi)))
mprintf("w=Eg/Ka *=%frad/s\n",w)//ans may vary due to roundoff error
n=(w*60)/(2*%pi)
mprintf("n=%frev/min\n",n)//ans may vary due to roundoff error
//other two techniques
//first technique
```

```
nB=1200 //speed in rev/min
n=nB*(Eg/Eg1)//ans may vary due to roundoff error
mprintf("n=%frev/min\n",n)
//second technique
mprintf(" d =%flb−ft\n",Td*0.738)//ans may vary due to roundoff error
mprintf("Ka‘  =Eg*/nB=%fV−min/rev\n",Eg/nB)
Ia=(Td*0.738)/(7.04*(Eg1/nB))//ans may vary due to roundoff error
mprintf("Ia= d /(7.04*Ka‘*  )=%fA\n",Ia)
n=Eg/(Eg1/nB)
mprintf("n=Eg/K‘ a  =%frev/min\n",n)//ans may vary due to roundoff error
```

Scilab code Exa 5.4 Find the terminal voltage at full load and no load and voltage

```
clc
//Ex5_4 uses a figure given in textbook
Ia=50 //current in amperes
IB=50 //current in amperes
nB=1200 //speed in rev/min
ratio=0.01 //ratio of Nsc/Nf ,unit less
Isc=0.6*Ia //equation given in textbook
mprintf("Isc=%dA\n",Isc)
If=1.3 //field current in amperes
mprintf("If*=If+(Nsc/Nf)*Isc=%fA\n",If+(ratio*Isc))
Eg1=132.5 //voltage in volts
mprintf("Ka‘  =Eg*/nB=%fV−min/rev\n",Eg1/nB)//ans may vary due to roundoff error
n=1140 //speed in rev/min
Eg=n*(Eg1/nB)
mprintf("Eg=Ka‘n=%fV\n",Eg)//ans may vary due to roundoff error
ra=0.2 //resistance in ohms
Ra=0.03+ra //by kirchodff's law and parallel
```

```
combination or resistances
mprintf("Ra=%f \n",Ra)
VTfl=Eg-(Ia*Ra)
mprintf("VTfl=%fV\n",VTfl)//ans may vary due to roundoff error
mprintf("If*=If+0=%fA\n",If)
Eg2=125 //voltage in volts
VTnl=Eg*(n/nB)
mprintf("Eg=Eg*(n/nB)=%fV\n",VTnl)//ans may vary due to roundoff error
mprintf("Voltage Regulation=(VTnl-VTfl)/VTfl=%f%c",((VTnl-VTfl)/VTfl)*100,'%') //ans may vary due to roundoff error
```

Scilab code Exa 5.5 Find the efficiency and input horsepower requirements under gi

```
clc
V=250 //voltage rating in volts
Pout=125000 //output power in watts
ra=0.025 //armature resistance in ohms
rsc=0.01 //resistance in ohms
rf=30 //field resistance in ohms
If=5 //field current in amperes
mprintf("Shunt field copper loss=%dW\n",If*If*rf)
Iload=Pout/V
Ia=Iload+If
Isc=Iload+If
mprintf("Ia=Isc=Iload+If=%dA\n",Ia)
mprintf("Seires filed copper losses=%dW\n",Isc*Isc*rsc)
mprintf("ACL=%fW\n",Ia*Ia*ra)//ans in textbook is wrong
mprintf("Brush copper loss=2Ia=%dW\n",2*Ia)
mprintf("Stray load loss=1%c of 125Kw=%fW\n",'%',0.01*Pout)
```

```
Prot=5000 //rotational loss in watts
losses=(If*If*rf)+(Isc*Isc*rsc)+(Ia*Ia*ra)+(2*Ia)+(0.01*Pout)+Prot //aadding all losses

mprintf("Efficiency=%f%c\n",(Pout/(Pout+losses))*100,'%')//ans may vary due to roundoff eror
rlosses=500 //rheostat losses in watts
Pin=Pout+losses+rlosses
mprintf("Pin required=%fW\n",Pin) //ans in the textbook is wrong
Ia1=sqrt((Prot+(If*If*rf))/(ra+rsc))
mprintf("Ia1=%fA\n",Ia1)
```

Chapter 6

Single Phase Machines

Scilab code Exa 6.1 Find the performance of a single phase motor under provided co

```
clc
//code uses a userdefined function complexstring
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
r1m=1.9//resistance in ohms
x1m=2.6 //reactance in ohms
r2=3.6 //resistance in ohms
x2=2.6 //reactance in ohms
Xm=56 //magnetizing reactance in ohms
Prot=25 //rotational losses in watts
f=60 //supply frequency in Hz
z1m=r1m+(%i*x1m)
```

```
s=0.05 //slip
disp('Z1m='+complexstring(z1m)+'  ')
Zf=((%i*Xm)*((r2/s)+(%i*x2)))/((%i*Xm)+(r2/s)+(%i*x2))//ans may vary due to roundoff error
disp('Zf/2='+complexstring(Zf/2)+'  ')
Zb=((%i*Xm)*((r2/(2-s))+(%i*x2)))/((%i*Xm)+(r2/(2-s))+(%i*x2))//ans may vary due to roundoff error
disp('Zb/2='+complexstring(Zb/2)+'  ')
Vm=115 //voltage in volts
Im=Vm/((Zf/2)+(Zb/2)+z1m) //ans may vary due to roundoff error
Imf=Im
Imb=Im
disp('Im='+complexstring(Im)+'A')
Pin=Vm*abs(Im)*(real(Im)/abs(Im))//ans may vary due to roundoff error
mprintf("Pin=%fW\n",Pin)
Pg=((abs(Im))^2)*(real(Zf/2)-real(Zb/2))//ans may vary due to roundoff error
mprintf("Pg=Pgf-Pgb=%fW\n",Pg)
mprintf(" d =%fN-m\n",Pg/(2*%pi*(f/2)))
DMP=Pg*(1-s)
mprintf("DMP=%fW\n",DMP)//ans may vary due to roundoff error
Pout=DMP-Prot
mprintf("Pout=%fW\n",Pout)//ans may vary due to roundoff error
mprintf("Efficiency=%f\n",Pout/Pin)//ans may vary due to roundoff error
```

Scilab code Exa 6.2 Calculate the starting torque for the motor under provided par

```
clc
//Ex6_2 is an extension of Ex6_1
//code uses userdefined function complexstring
```

```
function s=complexstring(a)


    if imag(a)>=0 then
        s=sprintf('%g+%gi',real(a),imag(a))
    else
        s=sprintf('%g%gi',real(a),imag(a))

    end
    funcprot(0)
endfunction
r1a=12//resistance in ohms
x1a=6.5//reactance in ohms
Xc=-20 //reactance in ohms
r1m=1.9 //from E6_1
x2=2.6 //from Ex6_1
s=1
a=1.6 //no unit
r2=3.6 //resistance in ohms
x2=2.6 //reactance in ohms
Xm=56 //magnetizing reactance in ohms
Vm=115 //applied voltage in volts
Zf=((%i*Xm)*((r2/s)+(%i*x2)))/((%i*Xm)+(r2/s)+(%i*x2))//from Ex6_1
Zst=Zf
Zb=Zf
z1a=r1a+(%i*x1a)+(%i*Xc)
disp('z1a='+complexstring(z1a)+'  ')//ans may vary due to roundoff error
mprintf("z1a=%fohm havinga phase angle of %f degrees\n",abs(z1a),(acos(real(z1a)/abs(z1a)))*180/%pi)
Z12=((1/2)*(z1a/(a*a)))-(r1m+(%i*x2))//ans in textbook is wrong
disp('Z12='+complexstring(Z12)+'  ')//ans may vary due to roundoff error
mprintf("Z12=%fohm havinga phase angle of %f degrees\n",abs(Z12),(acos(real(Z12)/abs(Z12)))*180/%pi)
Vmf=(Vm/2)*(1-(%i/a))
```

```
disp('Vmf='+complexstring(Vmf)+'V')//ans may vary due to roundoff error
mprintf("Vmf=%fV havinga phase angle of %f degrees\n",abs(Vmf),(-1)*(acos(real(Vmf)/abs(Vmf)))*180/%pi)
Vmb=(Vm/2)*(1+(%i/a))
disp('Vmb='+complexstring(Vmb)+'V')//ans may vary due to roundoff error
mprintf("Vmb=%fV having a phase angle of %f degrees\n",abs(Vmb),(acos(real(Vmb)/abs(Vmb)))*180/%pi)
Imf=11.77*exp(%i*(-1)*54.93*%pi/180)//textbook doesnt provide any formula or hint for this calculation
Imb=4.37*exp(%i*(-1)*19.7*%pi/180)//textbook doesnt provide any formula or hint for this calculation
disp('Imf='+complexstring(Imf)+'A')//ans may vary due to roundoff error
disp('Imb='+complexstring(Imb)+'A')//ans may vary due to roundoff error
mprintf("Imf=%fA having a phase angle of %f degrees\n",11.77,-54.93)
mprintf("Imb=%fA having a phase angle of %f degrees\n",4.37,-19.37)
mprintf(" st =%fN-m\n",(2*real(Zst)*((abs(Imf)^2)-(abs(Imb)^2)))/(60*%pi))//ans may vary due to roundoff error
Im=Imf+Imb
disp('Im='+complexstring(Im)+'A')//ans may vary due to roundoff error
mprintf("Im=%fA having a phase angle of%f degrees\n",abs(Im),(-1)*(acos(real(Im)/abs(Im)))*180/%pi)
Ia=(%i*(Imf-Imb))/a
disp('Ia='+complexstring(Ia)+'A')//ans may vary due to roundoff error
mprintf("Ia=%fA having a phase angle of %f degrees\n",abs(Ia),(acos(real(Ia)/abs(Ia)))*180/%pi)
I=Im+Ia
disp('Line current='+complexstring(I)+'A')//ans may
```

```
    vary due to roundoff error
mprintf("I=%fA having a phase angle of %f degrees\n"
    ,abs(I),(-1)*(acos(real(I)/abs(I)))*180/%pi)
```

Chapter 8

Forces And Torques In Electromagnetic Systems

Scilab code Exa 8.2 `1Find forces on plunger at 1A rms 2Voltage applied for the cur`

```
clc
x=0.01 //length in metres
L=0.03+(270*x*x) //equation provided in the textbook
mprintf("L(0.01)=%fH\n",L)
w=377 //angular  frequency in rad/sec
XL=w*L
mprintf("XL=wL= %f  \n",XL)//ans may vary due to toundoff error
I=1 //current in ampere
V=I*XL
mprintf("V=IXL=%fV\n",V)//ans may vary due to toundoff error
a=540 //comes from an equation in textbook,unit is henry/metre
f=(1/2)*(a*x)
mprintf("f=%fN\n",f)
```