Scilab Textbook Companion for Power Semiconductor Drives by S. Sivanagaraju[1]

Created by
Akshay Arun Raut
BE
Electrical Engineering
Government College Of Engineering,Karad
College Teacher
None
Cross-Checked by
None

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: Power Semiconductor Drives

Author: S. Sivanagaraju

Publisher: Prentice Hall India Learning Pvt Ltd

Edition: 1

Year: 2009

ISBN: 8120336585

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

Control of DC Motor by Single Phase Converters

Scilab code Exa 1.1 Ex1

```
clc
// Variable Initialization
Vm=230 //Supply Voltage in Volts
af=0 // Firing Angle of Converters in Field
Rf=200 //Field Resistance in ohm
T=50 //Load Torque in N-m
Kt=0.8 //Torque Constant in N-m/A^2
N=900 //Motor Speed in rpm
Ra=0.3 // Armature Resistance in ohm

//Solution
Vf=Vm*(1+cos(af))*(1/%pi)*1.414  //Voltage across Field in Volts
If=Vf/Rf //Field Current in Amp
Ia=T/(Kt*If) //Armature Current in Amp
w=(2*%pi*N)/60 //Angular Speed in rad/sec
Eb=Kt*If*w //Back Emf in Volts
Va=Eb+Ia*Ra // Voltage across Armature
A=%pi*Va*(1/Vm)*(1/1.414)
```

```
aa=acosd(A-1) //Firing Angle of Semi Converter in
    Armature Circuit
VaIa=Va*Ia //Power output of Converters in Armature
    Circuit in Watts
I=sqrt((180-aa)*(1/180))*Ia //Input Current
VA=Vm*I //Input VA
pf=VaIa/VA //Input Power Factor
//Result
printf('\n\n The Field Current=%0.1f Amp\n\n',If)
printf('\n\n The Firing Angle of Semi Converter in
    Armature Circuit=%0.1f degree\n\n',aa)
printf('\n\n The Power Factor of Semi Converter in
    Armature Circuit=%0.1f\n\n',pf)
```

Scilab code Exa 1.2 Ex2

```
clc
// variable Initialization
Vm=208 //Supply Voltage In Volts
af=0 //Firing Angle Of Converters In Field
Rf=147 //Field Resistance In Ohm
Ra=0.25 //Armature Resistance In Ohm
T=45 //Load Torque In N-m
Kv=0.7032 //Motor Voltage Constant
N=1000 //Motor Speed In Rpm

//Solution
Vf=Vm*(1+cos(af))*(1/%pi)*1.414 //Voltage Across
    Field In Volts
If=Vf/Rf //Field Current In Amp
Ia=T/(Kv*If) //Armature Current In Amp
w=(2*%pi*N)/60 //Angular Speed In Rad/Sec
Eb=Kv*If*w //Back Emf In Volts
Va=Eb+Ia*Ra //Voltage Across Armature In Volts
A=%pi*Va*(1/Vm)*(1/1.414)
```

```
aa=acosd(A-1) //Delay Angle Of Semi Converter In Armature Circuit
VaIa=Va*Ia //Power Output Of Converters In Armature Circuit In Watts
I=sqrt((180-aa)*(1/180))*Ia //Input Current
VA=Vm*I //Input VA
Pf=VaIa/VA //Input Power Factor

//Result
printf('\n\n The Field Current=%0.1f Amp\n\n',If)
printf('\n\n The Delay Angle Of Semi Converter In Armature Circuit=%0.1f degree\n\n',aa)
printf('\n\n The Power Factor Of Semi Converter In Armature Circuit=%0.1f\n\n',Pf)
//The answers vary due to round off error(2nd and 3rd)
```

Scilab code Exa 1.3 Ex3

```
clc
//Variable Initialization
V=120 //Supply Voltage In Volts
Vm=120*1.414 //Max. Voltage In Volts
Ra=10 //Armature Resistance In Ohm
af=0 //Firing Angle Of Converter
Eb=60 //Back Emf In Volts

//Solution
Va=Vm*(1+cos(af))*(1/%pi)//Voltage Across armature In Volts
Ia=(Va-Eb)/Ra //Average Value Of Armature Current In Amp

//Result
printf('\n\n The Average Value Of Armature Current=
```

```
%0.1f Amp\n\n',Ia)
```

Scilab code Exa 1.4 Ex4

```
clc
//Variable Initialization
Vm=320 //Input Voltage In Volts
Eb=100 //Back Emf In Volts
Ra=5 //Armature Resistance In Ohm
af=45 // Firing Angle Of SCR In Degree
N=1200 //Speed Of Motor In RPM

//Solution
Va=Vm*(1/%pi)*(1+cosd(af)) //Voltage Across
    Armature In volts
Ia=(Va-Eb)/Ra //Armature Current Amp
W=(2*%pi*N)/60 //Angular Speed In rad/Sec.
K=Eb/W //Voltage Constant In V-rad/Sec
T=K*Ia //Torque Of Motor In N-m

//Result
printf('\n\n The Armature Current=%0.1f Amp\n\n',Ia)
printf('\n\n The Motor Torque=%0.1f N-m\n\n',T)
//The answers vary due to round off error
```

Scilab code Exa 1.5 Ex5

```
clc
//variable Initialization
Vm=230//Input Voltage In Volts
Ra=0.5 //Armature Resistance In Ohm
Rf=190 //Field resistace in Ohm
N=1400 //Speed Of Motor In RPM
```

```
Ka=0.8 //Motor voltage constant in V/A-rad/sec
T=50 //Load Torque In N-m

//Solution
W=(2*%pi*N)/60 //Angular Speed In rad/Sec.
//Since the maximum field voltage and current is obtained at a Firing Angle of '0'degree
af=0 //Firing Angle Of SCR In Degree
Vf=(Vm*1.414)*(1+cosd(af))*(1/%pi) //field Voltage In volts
If=Vf/Rf //Field Current In Amp.
Ia=T/(Ka*If) //Armature current in Amp.
Eb=(Ka*If*W)//this value is wrongly calculated in book
Vdc=Eb+Ia*Ra //Voltage across armature in volts
//for the semi converter fed dc motor the armature voltage is given by...
A=%pi*Vdc*(1/Vm)*(1/1.414)
aa=acosd(A-1) //slight change occurs in ans as Eb is wrongly calculated in book

//Results
printf('\n\n The field Current =%0.1f Amp\n\n',If)
printf('\n\n The firing Angle Of Armature=%0.1f Degree\n\n',aa)
//The answer provided in the textbook is wrong(2nd answer)
```

Scilab code Exa 1.6 Ex6

```
clc
// Variable Initialization
Vm=200 //Supply Voltage in Volts
N=1000 //speed of motor in RPM
I=13 //Motor current in Ampere
```

```
Ra=3 //Armature circuit resistance in Ohm
L=100e-3 //Armature circuit Inductance in mH
V=230 //Ac Source voltage in Volts
f=50 //source Frequency in Hz
a=30
aa=(30*%pi)/180 //in rad/sec
N2=400
//Solution
P=atand((2*%pi*f*L)/Ra) // In Degree
Cot_P=1/(tand(P))
A=exp(-%pi*Cot_P)
B=exp((-aa)*Cot_P)
Z=sqrt((Ra^(2))+((2*%pi*f*L)^2)) //Impedance In Ohm
Eb=Vm-(I*Ra) //back emf in Volts
w=(2*%pi*N)/60 //Angular Speed in rad/sec
K=Eb/w
AA=(Ra*V*1.41)/(K*Z)
Wmc=AA*((sind(P)*B)-(sind(a-P)*A))*(1/(1-A)) //Critical Speed in rad/Sec
Wrpm=(Wmc*60)/(2*%pi) //speed in rpm
//As the motor speed of 400 rpm is less than the critical speed ,the drive is operating under continuous conduction mode
af=30 //firing angle in Degree
Va=(V*1.414)*(1+cosd(af))*(1/%pi) //Armature voltage in volts
//At 400 RPM
Eb1=Eb*(N2/N) //This Value Is Wrongly Calculated in Textbook
T=K*(Va-Eb1)*(1/Ra) //Torque in N-m
//Motor back emf for critical speed equal to 1149.67 rpm
Ec=(Wrpm*Eb)/N //critical emf in volts
Tc=K*(Va-Ec)*(1/Ra) //Critical Torque in N-m
//Since the motor torque of 70 N-m is greater than the critical torque Tc ,the drive is operating in continuous conduction
Ia=T/K //Armature current in Amp
```

```
Eb2=Va-(Ia*Ra) // Back emf in Volts
Nm=(Eb2*N)/161 //Answer changed due wrong value is taken in book of Eb1

//Results
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The motor Speed =%0.1f RPM \n\n',Nm)
//The answer provided in the textbook is wrong
```

Scilab code Exa 1.7 Ex7

```
clc
// Variable Initialization
Vm=230 //Supply Voltage in Volts
N=650//speed of motor in RPM
I=100//Motor curent in Ampere
Ra=0.08 //Armature circuit resistance in Ohm
L=8e-3 //Armature circuit Inductance in mH
V=230 //Ac Source voltage in Volts
f=50 //source Frequency in Hz
a=60
aa=(60*%pi)/180 //in rad/sec
a1=120
aa1=(120*%pi)/180 //in rad/sec
N2=400
T=1000

//Solution
P=atand((2*%pi*f*L)/Ra) // In Degree
Cot_P=1/(tand(P))
A=exp(-%pi*Cot_P)
B=exp((-aa)*Cot_P)
Z=sqrt((Ra^2)+((2*%pi*f*L)^2)) //Impedance In Ohm
Eb=Vm-(I*Ra) //back emf in Volts
w=(2*%pi*N)/60 //Angular Speed in rad/sec
```

```
K=Eb/w
AA=(Ra*Vm*1.414)/(K*Z)
Wmc=AA*((sind(P)*B)-(sind(a-P)*A))*(1/(1-A)) //Critical Speed in rad/Sec
Wrpm=(Wmc*60)/(2*%pi) //speed in rpm
//motor back emf for critical speed of 148 RPM
Ec=(Wrpm*Eb)/N //Critical emf in Volts
Va=(V*1.414)*(1+cosd(a))*(1/%pi) //Armature voltage in volts
Tc=K*(Va-Ec)*(1/Ra) //Critical Torque in N-m
//The check condition is
Ia=T/K //Armature current in Amp
Eb2=Va-(Ia*Ra) // Back emf in Volts
Nm=(Eb2*N)/Eb//Motor speed in Rpm

B1=exp((-aa1)*Cot_P)
Wmc1=AA*((sind(P)*B1)-(sind(a1-P)*A))*(1/(1-A)) //Critical Speed in rad/Sec
Wrpm1=(Wmc1*60)/(2*%pi) //speed in rpm
//motor back emf for critical speed of 154 RPM
Ec1=(Wrpm1*Eb)/N //Critical emf in Volts
Va1=(V*1.414)*(1+cosd(a1))*(1/%pi) //Armature voltage in volts
Tc1=K*(Va1-Ec1)*(1/Ra) //Critical Torque in N-m
//The check condition is
Ia1=-T/K //Armature current in Amp
Eb3=Va1-(Ia1*Ra) // Back emf in Volts
Nm1=(Eb3*N)/Eb//Motor speed in Rpm

//Results
printf('\n\n The motor Speed =%0.1f RPM \n\n',Nm)
printf('\n\n The motor Speed =%0.1f RPM \n\n',Nm1)
//The answers vary due to round off error(1st answer)
```

Scilab code Exa 1.8 Ex8

```
clc
// Variable Initialization
Vm=230 //Supply Voltage in Volts
N=650//speed of motor in RPM
I=100//Motor curent in Ampere
Ra=0.08 //Armature circuit resistance in Ohm
L=8e-3 //Armature circuit Inductance in mH
V=230 //Ac Source voltage in Volts
f=50 //source Frequency in Hz
a=60
aa=(60*%pi)/180 //in rad/sec
N2=200

//Solution
P=atand((2*%pi*f*L)/Ra) // In Degree
Cot_P=1/(tand(P))
A=exp(-%pi*Cot_P)
B=exp((-aa)*Cot_P)
Z=sqrt((Ra^2)+((2*%pi*f*L)^2)) //Impedance In Ohm
Eb=Vm-(I*Ra) //back emf in Volts
w=(2*%pi*N)/60 //Angular Speed in rad/sec
K=Eb/w
AA=(Ra*Vm*1.414)/(K*Z)
Wmc=AA*((sind(P)*B)-(sind(a-P)*A))*(1/(1-A)) //
    Critical Speed in rad/Sec
Wrpm=(Wmc*60)/(2*%pi) //speed in rpm
//since the speed of 200 rpm is less than the
    critical speed ,the drive is operating under
    continuous conduction .Hence
Va=(V*1.414)*(1+cosd(a))*(1/%pi) //Armature voltage
    in volts
Ec=(N2*Eb)/N //Critical emf in Volts
T=K*(Va-Ec)*(1/Ra) //Critical Torque in N-m

//For
a1=120
```

```
aa1=(120*%pi)/180 //in rad/sec
B1=exp((-aa1)*Cot_P)
Wmc1=AA*((sind(P)*B1)-(sind(a1-P)*A))*(1/(1-A)) //Critical Speed in rad/Sec
Wrpm1=(Wmc*60)/(2*%pi) //speed in rpm
//since the motor speed of 200 RPM is greater than the critical speed the drive is operating under discontinuous condition for which
AA1=(Ra*Vm*1.414)/(Ec*Z)
e1=AA1*((sind(P)*A)-(sind(a1-P)*B1))+B1
b=(log(e1))/Cot_P
b=117.38
Va1=((Vm*1.41*(1+cosd(a1))+(%pi+((a1-b)*%pi/180))*Ec))/%pi//square root of 2 is rounded off as 1.4
T1=K*(Va1-Ec)/Ra //Critical Torque in N-m

//Results
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The motor Torque=%0.1f N-m \n\n',T1)
//The answers vary due to round off error
```

Scilab code Exa 1.9 Ex9

```
clc
// Variable Initialization
Vm=230 //Supply Voltage in Volts
Ia=50//Motor Armature current in Ampere
Ra=0.25//Armature circuit resistance in Ohm
Rf=200 // Field circuit resistance in Ohm
f=50 //source Frequency in Hz
a=45 //Firing angle  in the armature  circuit in degree
Vd=1 //Brush contact drop V/brush
af=0 //Firing angle in the field circuit in degree
Kf=1.1 //Torque and voltage constant
```

```
//Solution
Vf=(2*Vm*1.414)*cosd(af)*(1/%pi) //Voltage in field circuit in Volts
//When a=45 degree
Va=(2*Vm*1.414)*cosd(a)*(1/%pi) //Voltage in Armature circuit in Volts
If=Vf/Rf //Field current in Amp.
T=Kf*Ia*If //Torque in N-m
Eb=Va-(Ia*Ra)-Vd*2 //Back emf in Volts
W=Eb/(Kf*If) //Angular speed in Rad/sec
N=W*60/(2*%pi) //Motor speed in RPM

//Results
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The motor Speed=%0.1f RPM \n\n',N)
```

Scilab code Exa 1.10 Ex10

```
clc
// Variable Initialization
Vm=230 //Supply Voltage in Volts
R=0.25//Combined Field and Armature circuit resistance in Ohm
N=1000 //Motor speed in Rpm
Kaf=0.03 //Constant in N-m/A^2
Krcs=0.075 //Constant in V-S/Rad
a=30 //firing angle in Degree

//Solution
//For a full converter controlled dc motor drive
W=(2*%pi*N)/60 //Angular speed in Rad/sec
Ia=((((2*Vm*1.414)/%pi)*cosd(a))-(Krcs*W))/(R+(Kaf*W)) //Armature current in Amp
T=Kaf*(Ia^2) //Torque in N-m
```

```
Va=(2*Vm*1.414)*cosd(a)*(1/%pi)//Average voltage in volts
Po=Va*Ia //Output power in Watt
Irms=Ia*sqrt((180-a)/180) //Rms Current in Amp
Pi=Vm*Irms
pf=Po/Pi //Power factor

//Results
printf('\n\n The motor Current =%0.1f Amp \n\n',Ia)
printf('\n\n The motor Torque=%0.1f N-m \n\n',T)
printf('\n\n The Supply Power Factor=%0.1f Lag\n\n',pf)
//The answers vary due to round off error
```

Scilab code Exa 1.11 Ex11

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=0.5//Armature circuit resistance in Ohm
Irms=25 //Armature current in Amp
Nr=800 //Motor speed in Rpm
Kaf=0.172 //Motor Voltage Constant in V/rpm
a=60//firing angle in Degree

//Solution
//CASE:A
//For motoring action
Ka=(Kaf*60)/(2*%pi)//Constant in V-s/rad
T=Ka*Irms //Torque of motor in N-m
Va=(2*Vm*1.414)*cosd(a)*(1/%pi)//Average voltage in Volts
Eb=Va-(Irms*Ra)//Back Emf in Volts
N=Eb/Kaf//Speed of motor in Rpm
//The supply current is square wave if motor current
```

```
is constant and ripple-free with Amplitude 25A
P=Vm*Irms //Supply VA in Watt
//Power from supply is real power if losses in converter are neglected
Ps=Va*Irms //Power in Watt
pf=Ps/P //Power factor lag

//CASE:B
//For polarity reversal (regeneration action)
Eb1=-Eb //Back emf in Volts
Va1=Eb1+(Irms*Ra)
af=acosd((Va1*%pi)/(2*Vm*1.414))//Firing angle in Degree
//Power fed from DC Machine
Pdc=Eb*Irms //Power in watt
//Power  lost in armature resistance
PL=((Irms)^2)*Ra //Power in Watt
//Power fed back to ac supply is
PF=Pdc-PL //Power in watt

//Results
printf('\n\n The motor Torque=%0.1f N-m \n\n',T)
printf('\n\n The motor Speed  =%0.1f RPM \n\n',N)
printf('\n\n The Supply Power Factor=%0.1f Lag\n\n',pf)
printf('\n\n The Firing Angle=%0.1f Degree\n\n',af)
printf('\n\n The Power fed back to Supply=%0.1f Watt\n\n',PF)
//The answers vary due to round off error
```

Scilab code Exa 1.12 Ex12

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
```

```
Ra=0.3//Armature circuit resistance in Ohm
I=18 //Rated current in Amp
Nr=1500 //Motor speed in Rpm
a=45//firing angle in Degree
Vs=220 //input in volts

//Solution
Va=(2*Vm*1.414)*cosd(a)*(1/%pi)//Average voltage in Volts
Eb=Va-(I*Ra)//Back emf in volts
//When the polarity of the back emf is reversed ,the machine would act as generator and hence the governing equation
Va=-Eb+(I*Ra) //Armature voltage in volts
af=acosd((Va*%pi)/(2*Vs*sqrt(2)))//Firing angle in Degree(This value is Wrongly calculated in Textbook)
Pg=Eb*I //Power generated in Watt
Pl=((I)^2)*Ra //Power loss in armature in Watt
P=Pg-Pl //Power fed to the supply in Watt

//Results
printf('\n\n The Firing Angle=%0.1f Degree\n\n',af)
printf('\n\n The Power fed back to Supply=%0.1f Watt\n\n',P)
//The answer provided in the textbook is wrong(1st answer)
```

Scilab code Exa 1.13 Ex13

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=1//Armature circuit resistance in Ohm
I=12 //Rated current in Amp
```

```
Nr=1500 //Motor speed in Rpm
Va=210 //Motor voltage in volts
N1=1200 //Speed in RPM

//Solution
Eb=Va-(I*Ra) //Back emf in volts
N=(Nr*2*%pi)/60 //Speed in Rad/sec
Ka=Eb/N //Constant
//Current at rated torque
 //At 1200 Rpm
 Eb1=(N1*Eb)/Nr
 af=acosd((((I*Ra)+Eb1)*%pi)/(2*Vm*1.414))//Firing angle in Degree
 //At Eb=-198 V at speed 1500 RPM
 af1=acosd((((I*Ra)-Eb)*%pi)/(2*Vm*1.414))//Firing angle in Degree
 //But reversal of field of armature forward regeneration is obtained
 Ka1=-Ka
 W=-Eb/Ka1 //Angular speed in Rad/sec(Wrongly calculated in book,wrong value of Eb is taken)
 N1=(W*60)/(2*%pi) //Speed in Rpm

 //Results
printf('\n\n The Firing Angle=%0.1f Degree\n\n',af)
printf('\n\n The Firing Angle=%0.1f Degree\n\n',af1)
printf('\n\n The motor Speed =%0.1f RPM \n\n',N1)
//The answer provided in the textbook is wrong(3rd answer omly)
```

Scilab code Exa 1.14 Ex14

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
```

```
Ra=0.75//Armature circuit resistance in Ohm
I=8 //Armature current in Amp
Nr=1200 //Motor speed in Rpm
Va=220//Rated voltage of motor in Volts
a=45//firing angle in Degree
T=8 //Motor torque in N-m
Nr1=800 //Speed in Rpm


//Solution
N=(Nr*2*%pi)/60 //Speed in Rad/sec
Kaf=(Va-I*Ra)/N //Motor Constant
//(A) For torque of 8 N-m
Ia=T/Kaf //Armature Current in Amp
V=(2*Vm*1.414)*cosd(a)*(1/%pi)//Average voltage in Volts
W=(V-Ia*Ra)/Kaf //Angular speed in Rad/sec
N=(W*60)/(2*%pi) //Speed in Rpm
//(B) a=45 Degree
N1=(Nr1*2*%pi)/60 //Speed in Rad/sec
Ia1=(V-Kaf*N1)/Ra //armature current in amp
T1=Kaf*Ia1 //Torque in N-m

//Results
printf('\n\n The motor Speed  =%0.1f RPM \n\n',N)
printf('\n\n The motor Torque=%0.1f N-m \n\n',T1)
//The answers vary due to round off error
```

Scilab code Exa 1.15 Ex15

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=0.2//Armature circuit resistance in Ohm
I=80 //Rated current in Amp
```

```
Nr=1200 //Motor speed in Rpm
V=220//Rated voltage of motor in Volts
a=150//firing angle in Degree
Nr1=-700 //Speed in Rpm

//Solution
N=(Nr*2*%pi)/60 //Speed in Rad/sec
Kaf=(V-I*Ra)/N //Motor Constant inN-m/A
//(A)For rated motor torque
Va=(2*Vm*1.414)*cosd(a)*(1/%pi)//Average voltage in Volts
N1=(Nr1*2*%pi)/60 //Speed in Rad/sec
af=acosd(((Kaf*N1)+(I*Ra))*%pi/(2*Vm*1.414))//Firing angle in Degree
Eb1=V-(I*Ra)//Back emf in Volts
Eb2=(Nr1/Nr)*Eb1 //Back emf in Volts
Va1=Eb2+(I*Ra)//Armature voltage in volts
af1=acosd((Va1*%pi)/(2*Vm*1.414))//Firing angle in Degree
//(B) Half rated Torque
Ia=(1/2)*I //Armature current in Amp
W=(Va-(Ia*Ra))/Kaf //Angular speed in Rad/sec
N=(W*60)/(2*%pi) //Speed in Rpm

 //Results
printf('\n\n The Firing Angle=%0.1f Degree\n\n',af1)
printf('\n\n The motor Speed=%0.1f RPM \n\n',N)
//The answers vary due to round off error(2nd only)
```

Scilab code Exa 1.16 Ex16

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=0.05//Armature circuit resistance in Ohm
```

```
L=0.85e-3//Inductance in mH
N=750//Motor speed in Rpm
V=220//Rated voltage of motor in Volts
a=60//firing angle in Degree
La=0.75e-3//External Inductance in mH
I=175 //motor current in Amp
f=50 //source Frequency in Hz

//Solution
P=atand((2*%pi*f*L)/Ra) // In Degree
Z=sqrt((Ra^2)+((2*%pi*f*L)^2)) //Impedance In Ohm
Eb=V-(I*Ra) //back emf in Volts
w=(2*%pi*N)/60 //Angular Speed in rad/sec
K=Eb/w
Cot_P=1/(tand(P))
A=exp(-%pi*Cot_P)
AA=(Ra*Vm*1.414)/(K*Z)
Wmc=AA*sind(a-P)*((1+A)/(A-1))//Critical Speed in
    rad/Sec
Wrpm=(Wmc*60)/(2*%pi) //speed in rpm
//Since the motor Speed is greater than critical
    Value Wmc ,The drive is operating under
    discontinuous conduction
//At N=600 Rpm
N1=600 //Speed in Rpm
Eb1=(N1/N)*Eb//Back emf in Volts
//By trial and error method
b=201.45//Bita in Degree
Va=((Vm*1.414)*(cosd(a)-cosd(b))+(%pi+(a-b)*(%pi
    /180))*(Eb1))*(1/%pi) //Armature voltage in Volts
Ia=(Va-Eb1)/Ra //Armature current in Amp
T=K*Ia//Torque in N-m

//La=2.85 mH
N2=-400
a1=120
P1=atand((2*%pi*f*La)/Ra) // In Degree
Cot_P1=1/(tand(P1))
```

```
Eb2=(N2/N)*Eb//Back emf in Volts
Z1=sqrt((Ra^2)+((2*%pi*f*La)^2)) //Impedance In Ohm
AA1=(Ra*Vm*1.414)/(K*Z1)
A1=exp(-%pi*Cot_P1)
Wmc1=AA1*sind(a1-P1)*((1+A1)/(A1-1))//Critical Speed in rad/Sec
Wrpm1=(Wmc1*60)/(2*%pi) //speed in rpm
//By trial and error method
b1=297.5//Bita in Degree
Va1=((Vm*1.414)*(cosd(a1)-cosd(b1))+(%pi+(a1-b1)*(%pi/180))*(Eb2))*(1/%pi) //Armature voltage in Volts
Ia1=(Va1-Eb2)/Ra //Armature current in Amp
T1=K*Ia1//Torque in N-m

//Since the motor speed (-600 rpm) is less than critical speed (-409.17 Rpm) the drive's opertion is continuous condition
N3=-600
Va2=(2*1.414*Vm)*(cosd(a1))*(1/%pi)
Eb3=(N3/N)*Eb//Back emf in Volts
Ia2=(Va2-Eb3)/Ra //Armature current in Amp
T2=K*Ia2//Torque in N-m

//Results
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The motor Torque =%0.1f N-m \n\n',T1)
printf('\n\n The motor Torque =%0.1f N-m \n\n',T2)
//The answers vary due to round off error
```

Scilab code Exa 1.17 Ex17

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
```

```
Ra=2//Armature circuit resistance in Ohm
L=50e-3//Inductance in mH
N=1500//Motor speed in Rpm
V=220//Rated voltage of motor in Volts
a=60//firing angle in Degree
a1=120//firing angle in Degree
I=10//motor current in Amp
f=50 //source Frequency in Hz

//Solution
P=atand((2*%pi*f*L)/Ra) // In Degree
Z=sqrt((Ra^2)+((2*%pi*f*L)^2)) //Impedance In Ohm
Eb=V-(I*Ra) //back emf in Volts
w=(2*%pi*N)/60 //Angular Speed in rad/sec
K=Eb/w//value obtain is 1.2732395 approximating to 1.27
K=1.27//Approximation of K as per book and our obtained value
//At No Load
Wo=(Vm*1.414)/K//Angular Speed in rad/sec(For 0<a<%pi/2)
No=Wo*(60/(2*%pi))//Speed in Rpm
Cot_P=1/(tand(P))
A=exp(-%pi*Cot_P)
AA=(Ra*Vm*1.414)/(K*Z)
Wmc=AA*sind(a-P)*((1+A)/(A-1))//Critical Speed in rad/Sec
Wrpm=(Wmc*60)/(2*%pi) //speed in rpm
Eb1=(Wrpm*Eb)/N
//By trial and error method
b=240.45//Beta in Degree
bx=249.45//Beta in Degree
Va=((Vm*1.414)*(cosd(a)-cosd(bx))+(%pi+(a-b)*(%pi/180))*(Eb1))*(1/%pi) //Armature voltage in Volts
Ia=(Va-Eb1)/Ra //Armature current in Amp
T=K*Ia//Torque in N-m

//At a1=120
```

```
Wmc1=AA*sind(a1-P)*((1+A)/(A-1))//Critical Speed in rad/Sec
Wrpm1=(Wmc1*60)/(2*%pi) //speed in rpm
Wo1=(Vm*1.414*sind(a1))/1.273//Angular Speed in rad/sec(For 0<a<%pi/2)&K=1.273
No1=Wo1*(60/(2*%pi))//Speed in Rpm
Eb2=(Wrpm1*Eb)/N
//By trial and error method
b1=217.2//Beta in Degree
Va1=((Vm*1.414)*(cosd(a1)-cosd(b1))-((%pi+((a1-b1)*(1/180))*%pi)*Eb2))*(1/%pi) //Armature voltage in Volts
Ia1=(Va1-Eb2)/Ra //Armature current in Amp
T1=K*Ia1//Torque in N-m

//Results
printf('\n\n The motor No load Speed =%0.1f RPM \n\n',No)
printf('\n\n The motor Critical Speed =%0.1f RPM \n\n',Wrpm)
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The motor No load Speed =%0.1f RPM \n\n',No1)
printf('\n\n The motor Critical Speed =%0.1f RPM \n\n',Wrpm1)
printf('\n\n The motor Torque =%0.1f N-m \n\n',T1)
//The answers vary due to round off error
```

Scilab code Exa 1.18 Ex18

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=1.8//Armature circuit resistance in Ohm
L=32e-3//Inductance in mH
```

```
N=1200//Motor speed in Rpm
V=220//Rated voltage of motor in Volts
a=60//firing angle in Degree
I=15 //motor current in Amp
f=50 //source Frequency in Hz

//Solution
P=atand((2*%pi*f*L)/Ra) // In Degree
Z=sqrt((Ra^2)+((2*%pi*f*L)^2)) //Impedance In Ohm
Eb=V-(I*Ra) //back emf in Volts
w=(2*%pi*N)/60 //Angular Speed in rad/sec
K=Eb/w
Cot_P=1/(tand(P))
A=exp(-%pi*Cot_P)
AA=(Ra*Vm*1.414)/(K*Z)
Wmc=AA*sind(a-P)*((1+A)/(A-1))//Critical Speed in rad/Sec
Wrpm=(Wmc*60)/(2*%pi) //speed in rpm
//Since the motor Speed is greater than critical Value Wmc ,The drive is operating under discontinuous conduction
//At N=600 Rpm
N1=450 //Speed in Rpm
Eb1=(N1/N)*Eb//Back emf in Volts
//By trial and error method
b=239.4//Bita in Degree
Va=((Vm*1.414)*(cosd(a)-cosd(b))*(%pi+(a-b)*(%pi/180))*(Eb1))*(1/%pi) //Armature voltage in Volts
Ia=(Va-Eb1)/Ra //Armature current in Amp
T=K*Ia//Torque in N-m
//Since the motor Speed is greater than critical Value Wmc ,The drive is operating in discontinuous mode at
N2=1500
Eb2=(N2/N)*Eb//Back emf in Volts
//By trial and error method
b1=172.2//Bita in Degree
Va1=((Vm*1.414)*(cosd(a)-cosd(b))*(%pi+(a-b)*(%pi
```

```
/180))*(Eb2))*(1/%pi) //Wromg calculation in book
   they have taken value of beta as 17.2 instead of
   172.2
Ia1=(Va1-Eb2)/Ra //Armature current in Amp
T1=K*Ia1//Torque in N-m

//Results
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)//
   The answers vary due to round off error
printf('\n\n The motor Torque =%0.1f N-m \n\n',T1)
//The answer provided in the textbook is wrong
```

Scilab code Exa 1.19 Ex19

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=0.05//Armature circuit resistance in Ohm
La=3e-3//Inductance in mH
N=750//Motor speed in Rpm
V=220//Rated voltage of motor in Volts
a=60//firing angle in Degree
I=175 //motor current in Amp
f=50 //source Frequency in Hz

//La=2.85 mH
N2=-400
a1=120
P1=atand((2*%pi*f*La)/Ra) // In Degree
Eb=V-(I*Ra) //back emf in Volts
w=(2*%pi*N)/60 //Angular Speed in rad/sec
K=Eb/w
Cot_P1=1/(tand(P1))
Eb1=(N2/N)*Eb//Back emf in Volts
Z1=sqrt((Ra^2)+((2*%pi*f*La)^2)) //Impedance In Ohm
```

```
AA1=(Ra*Vm*1.414)/(K*Z1)
A1=exp(-%pi*Cot_P1)
Wmc1=AA1*sind(a1-P1)*((1+A1)/(A1-1))//Critical Speed in rad/Sec
Wrpm1=(Wmc1*60)/(2*%pi) //speed in rpm
Eb2=(Wrpm1*Eb)/N
a2=150
Va=(2*1.414*Vm)*(cosd(a2))*(1/%pi)
Ia=(Va-Eb2)/Ra
T=K*Ia//Torque in N-m
//As the torque of 400 N-m is greater than T ,hence the opertion is in Continuous conduction mode
T1=400
Ia1=T1/K
Eb3=Va-(Ia1*Ra)
Ns=(Eb3*N)/Eb
//As the torque of 400 N-m is less than T ,hence the opertion is in Discontinuous conduction mode
Z=0.2716
q=V*1.414/Z
//nothing is solved in textbook using numericals
//By Trial and error method beta is calculated
b=233.240

//Results
printf('\n\n The motor Speed =%0.1f rpm \n\n',Ns)
```

Scilab code Exa 1.20 Ex20

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=0.25//Combined Field and Armature circuit resistance in Ohm
N=1000 //Motor speed in Rpm
```

```
V=210//Rated voltage of motor in Volts
a=30//firing angle in Degree
Kaf=0.03 //Constant in N-m/A^2
Kres=0.075 //Constant in V-s/Rad

//Solution
//For semi-converter controlled Dc Drive
W=(2*%pi*N)/60 //angular speed in Rad/sec
Ia=(((Vm*1.414)/%pi)*(1+cosd(a))-(Kres*W))*(1/(Ra+
    Kaf*W))//Armature current in Amp
T=Kaf*(Ia)^2//Torque in N-m
Va=(Vm*1.414)*(1+cosd(a))*(1/%pi)//Average voltage
    in VoltsI
Irms=Ia*((180-a)/180)^(1/2)//RMS Current in Amp
Pa=Va*Ia //Power in Watt
Pi=Vm*Irms //Input power in Watt
Pf=(Pa/Pi)//Power Factor in Lag

//Results
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The motor Current =%0.1f Amp \n\n',Irms
    )
printf('\n\n The Supply Power Factor =%0.1f Lag \n\n
    ',Pf)
```

Scilab code Exa 1.21 Ex21

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=0.75//Combined Field and Armature circuit
    resistance in Ohm
N=1300 //Motor speed in Rpm
V=210//Rated voltage of motor in Volts
a=45//firing angle in Degree
```

```
Kaf=0.03 //Constant in N-m/A^2
Kres=0.075 //Constant in V-s/Rad

//Solution
//For semi-converter controlled Dc Drive
W=(2*%pi*N)/60 //angular speed in Rad/sec
Ia=(((Vm*1.414)/%pi)*(1+cosd(a))-(Kres*W))*(1/(Ra+
    Kaf*W))//Armature current in Amp
T=Kaf*(Ia)^2//Torque in N-m
I=(T/Kaf)^(1/2)//motor current in Amp
Va=(Vm*1.414)*(1+cosd(a))*(1/%pi)//Average voltage
    in Volts
//Input Power if losses are neglected
Ps=Va*I //Power loss in Watt
Pi=Vm*I*(5/6)^0.5 //power input in watt
Pf=(Ps/Pi)//Power Factor in Lag

//Results
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The motor Current =%0.1f Amp \n\n',I)
printf('\n\n The Supply Power Factor =%0.1f Lag \n\n
    ',Pf)
```

Scilab code Exa 1.22 Ex22

```
clc
// Variable Initialization
Vm=240//Supply Voltage in Volts
Ra=0.9//Combined Field and Armature circuit
    resistance in Ohm
N=900 //Motor speed in Rpm
V=220//Rated voltage of motor in Volts
a=45//firing angle in Degree
Kaf=0.035 //Constant in N-m/A^2

```

```
//Solution
//For semi-converter controlled Dc Drive
Va=(Vm*1.414)*(1+cosd(a))*(1/%pi)//Average voltage in Volts
W=(2*%pi*N)/60 //angular speed in Rad/sec
Ia=Va/(Ra+W*Kaf)//Current in Amp
T=Kaf*(Ia)^2//Torque in N-m

//Results
printf('\n\n The motor Current =%0.1f Amp \n\n',Ia)
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
```

Scilab code Exa 1.23 Ex23

```
clc
// Variable Initialization
Vm=250//Supply Voltage in Volts
Ra=0.3//Combined Field and Armature circuit resistance in Ohm
N=900 //Motor speed in Rpm
V=220//Rated voltage of motor in Volts
a=30//firing angle in Degree
Kaf=0.03 //Constant in N-m/A^2
Kres=0.075 //Constant in V-s/Rad

//Solution
W=(2*%pi*N)/60 //angular speed in Rad/sec
Ia=(((Vm*1.414)/%pi)*(1+cosd(a))-(Kres*W))*(1/(Ra+Kaf*W))//Armature current in Amp
T=Kaf*(Ia)^2//Torque in N-m
I=(T/Kaf)^(1/2)//motor current in Amp
//The motor Terminal voltage would be given by the output voltage of the converter
Va=(Vm*1.414)*(1+cosd(a))*(1/%pi)//Output voltage in Volts
```

```
//Results
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The motor Current =%0.1f Amp \n\n',I)
printf('\n\n The motor output voltage =%0.1f Volts\n\n',Va)
```

Scilab code Exa 1.24 Ex24

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=0.25//Combined Field and Armature circuit resistance in Ohm
N=1000//Motor speed in Rpm
V=210//Rated voltage of motor in Volts
a=30//firing angle in Degree
Kaf=0.03 //Constant in N-m/A^2
Kres=0.075 //Constant in V-s/Rad

//Solution
//For Full-converter controlled Dc Drive
W=(2*%pi*N)/60 //angular speed in Rad/sec
Ia=(((2*Vm*1.414)/%pi)*(cosd(a))-(Kres*W))*(1/(Ra+Kaf*W))//Armature current in Amp
T=Kaf*(Ia)^2//Torque in N-m
Va=(2*Vm*1.414)*(cosd(a))*(1/%pi)//Average voltage in Volts
//Input Power if losses are neglected
Ps=Va*Ia //Power loss in Watt
Pi=Vm*Ia//power input in watt
Pf=(Ps/Pi)//Power Factor in Lag

//Results
printf('\n\n The motor Current =%0.1f Amp \n\n',Ia)
```

```
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The Supply Power Factor =%0.1f Lag \n\n',Pf)//The answers vary due to round off error
```

Scilab code Exa 1.25 Ex25

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
Ra=2//Combined Field and Armature circuit resistance in Ohm
Kaf=0.23 //Constant in H
TL=20 //Load Torque in N-m
a=65//firing angle in Degree

//Solution
Ia=(TL/Kaf)^(1/2)//Average Armature current in Amp
V=(2*Vm*1.414)*cosd(a)*(1/%pi)//Average voltage in Volts
W=(V-Ia*Ra)/(Kaf*Ia) //Angular speed in Rad/sec
N=(W*60)/(2*%pi) //Speed in Rpm

//Results
printf('\n\n The motor Average Armature Current =%0.1f Amp \n\n',Ia)
printf('\n\n The motor Average Speed =%0.1f RPM \n\n',N)//The answers vary due to round off error
```

Scilab code Exa 1.26 Ex26

```
clc
// Variable Initialization
Vm=230//Supply Voltage in Volts
```

```
Ra=0.75//Combined Field and Armature circuit resistance in Ohm
N=1300//Motor speed in Rpm
V=210//Rated voltage of motor in Volts
a=45//firing angle in Degree
Kaf=0.03 //Constant in N-m/A^2
Kres=0.075 //Constant in V-s/Rad

//Solution
//For Full-converter controlled Dc Drive
W=(2*%pi*N)/60//angular speed in Rad/sec
Ia=(((2*Vm*1.414)/%pi)*(cosd(a))-(Kres*W))*(1/(Ra+Kaf*W))//Armature current in Amp
T=Kaf*(Ia)^2//Torque in N-m
I=(T/Kaf)^(1/2)//motor current in Amp
Va=(2*Vm*1.414)*(cosd(a))*(1/%pi)//Average voltage in Volts
//Input Power if losses are neglected
Ps=Va*I//Power loss in Watt
Pi=Vm*I//power input in watt
Pf=(Ps/Pi)//Power Factor in Lag

//Result
printf('\n\n The motor Torque =%0.1f N-m \n\n',T)
printf('\n\n The Supply Power Factor =%0.1f Lag \n\n',Pf)
```

Chapter 2

Control of DC Motor by Three Phase Converters

Scilab code Exa 2.1 Ex1

```
clc
//variable initialisation
a1=0//Initial Firing Angle of Converter
Vl=460//Line to Line Voltage in Volts
Ia=170//armature Current in Ampere
Ra=0.0999//Armature Resistance in ohm
K=0.33
a2=30
N1=1750//Rotor Speed in rpm

//solution
Vm=(sqrt(2)/sqrt(3))*Vl
Va=(3*sqrt(3)/%pi)*Vm*cosd(a1)
Ia1=17
Eb1=Va-(Ia1*Ra)
N0=Eb1/K//no load Speed in rpm
Va2=Va*cosd(a2)//For alpha=30
Eb2=Va2-(Ia1*Ra)
N01=Eb2/K//No load speed at alpha=30
```

```
Eb3=K*N1//For Full load Condition
Va3=Eb3+(Ia*Ra)
a3=acosd(Va3/Va)
P=Va3*Ia
Irms=Ia*sqrt(((180-a3)/180))
Vph=Vl/sqrt(3)
pf=P/(3*Vph*Irms)//Power Factor
Eb4=Va3-(Ia1*Ra)
N4=Eb4/K
R=(N4-N1)*100/N1//Speed Regulation
printf('\n\n No load speed at alpha 0=%0.1f rpm\n\n'
    ,N0)
printf('\n\n No load speed at alpha 30=%0.1f rpm\n\n
    ',N01)
printf('\n\n The Firing Angle for rated speed=%0.1f\
    n\n',a3)
printf('\n\n Power Factor at rated speed=%0.1f\n\n',
    pf)
printf('\n\n Speed Regulation=%0.1f\n\n',R)
```

Scilab code Exa 2.2 Ex2

```
clc
//variable initialisation
a1=0//Initial Firing Angle of Converter
Vl=460//Line to Line Voltage in Volts
Ia=150//armature Current in Ampere
Ra=0.0999//Armature Resistance in ohm
K=0.33
N1=1650//Rotor Speed in rpm
Ia1=15//armature Current for 2nd case in Ampere
//solution
Vm=(sqrt(2)/sqrt(3))*Vl
Va=(3*sqrt(3)/%pi)*Vm*cosd(a1)
Eb1=Va-(Ia*Ra)
```

```
N0=Eb1/K//no load Speed in rpm
a2=45
Va2=Va*cosd(a2)
Eb2=Va2-(Ia*Ra)
N01=Eb2/K//No load speed at alpha=30
Eb3=K*N1
Va3=Eb3+(Ia*Ra)
a3=acosd(Va3/Va)
Irms=Ia*sqrt((180-a3)/180)
P1=3*(Vl/sqrt(3))*Irms//Supply VA
P=Va3*Ia//Power input to motor
Pa=Va3*Ia
pf=Pa/P//Power Factor
Eb4=Va3-(Ia1*Ra)
N4=Eb4/K
R=(N4-N1)*100/N1//Speed Regulation
printf('\n\n No load speed at alpha 0=%0.1f rpm\n\n',N0)
printf('\n\n No load speed at alpha 45=%0.1f rpm\n\n',N01)
printf('\n\n The Firing Angle for rated speed=%0.1f\n\n',a3)
printf('\n\n Supply Power at rated speed=%0.1f watts\n\n',P)
printf('\n\n Power Factor at rated speed=%0.1f\n\n',pf)
printf('\n\n Speed Regulation=%0.1f\n\n',R)
//The answers vary due to round off error
```

Scilab code Exa 2.3 Ex3

```
clc
//variable initialisation
Va=220 //supply voltage in volts
N1=1500 //speed in rpm
```

```
I=50 // current in ampere
Ra=0.5 //armature resistance in ohm
Vl=440 //line voltage in volts
f=50 //frequency in Hz
N2=1200 //speed in rpm

//solution
Vm=(Va*%pi)/(3*sqrt(3))
Vph=(Vl*(sqrt(2)))/(sqrt(3))
Xmer_ratio=Vph/Vm
Eb1=Va-(Ra*I)
Eb2=(N2/N1)*Eb1
Va=Eb2+Ra*I
a=acosd((Va*%pi)/(3*sqrt(3)*Vm))
N3=800
Eb3=(-N3/N1)*Eb1
Va1=Eb3+(2*I*Ra)
a1=acosd((Va1*%pi)/(3*sqrt(3)*Vm))
printf('\n\n Transformer Turns Ratio=%0.1f\n\n',
    Xmer_ratio)
printf('\n\n The Firing Angle=%0.1f\n\n',a)
printf('\n\n The Firing Angle=%0.1f\n\n',a1)
```

Scilab code Exa 2.4 Ex4

```
clc
//variable initialisation
Va=220 //supply voltage in volts
N1=1500 //speed in rpm
Ra=2 //armature resistance in ohm
La=0.02836 //armature inductance in mH
f=50 //frequency in Hz

//solution
Vl=(Va*%pi)/(3*sqrt(2))
```

```
Vm=sqrt(2)*V1
printf('\n\n The Source Voltage Required=%0.1f Volts
   \n\n',Vm)
```

Scilab code Exa 2.5 Ex5

```
clc
//variable initialisation
V1=450 //terminal voltage in volts
Vd=30 //voltage drop in volts
V2=420 //input supply in volts
f=50 //frequency in Hz
a1=0//Firing Angle of Converter
//solution
Vt=V1+Vd
V0_0=(3*sqrt(2))/(%pi)
V0_a=480

a2=acosd(V0_a/(V0_0*V2))
printf('\n\n The Firing Angle=%0.1f\n\n',a2)
```

Scilab code Exa 2.6 Ex6

```
clc
//variable initialization
N=800 //speed in rpm
P=80000 //power in kw
V=440 //Supply voltage in volts
f=50 //Supply frequency in Hz
Ra=0.8 //armature resistance in ohm
k=0.2 //machine constant in V/rpm
Ia=160 //rated current in ampere

```

```
//solution
Vp=V/(sqrt(3))//Phase Voltage
Eb=k*N//Back emf in Volts
V2=Eb+(Ia*Ra)
a=acosd((V2*%pi)/(3*sqrt(6)*Vp))
printf('\n\n The Firing Angle=%0.1f\n\n',a)
```

Scilab code Exa 2.7 Ex7

```
clc
//variable initialisation
N=900 //speed in rpm
V=430 //Supply voltage in volts
Ia=20 //current in ampere
N1=900 //speed in rpm
N2=450 //speed in rpm
Ra=0.2 //armature resistance in ohm

//solution
V1=V/1.35
V2=((V-(Ia*Ra))/2)+Ia*Ra
a=acosd(V2/V)
printf('\n\n The RMS Voltage per phase=%0.1f Volts\n\n',V2)
printf('\n\n The Firing Angle=%0.1f\n\n',a)
```

Scilab code Exa 2.8 Ex8

```
clc
//Variable Initialisation
Vs=220//Supply Voltage in Volts
N=600//Rotor Speed in rpm
Ra=0.4//Armature Resistance in ohm
```

```
Rf=150//Field Resistance in ohm
af=0//Firing Angle for Maximum field current
aa=0//Firing Angle for maximum Armature current
N2=1200//Speed in rpm
T=120//Torque in N-m
K=1.4//Motor voltage Constant
//Solution
Vm=Vs/sqrt(3)
W=2*%pi*N/60
Vf1=3*sqrt(3)*sqrt(2)*Vm*cosd(af)/%pi
If1=Vf1/Rf//Field Current in Amp
Ia1=T/(K*If1)
Eb1=K*If1*W
Va1=Eb1+(Ia1*Ra)
aa1=acosd(Va1*%pi/(3*sqrt(3)*sqrt(2)*Vm))
Va2=3*sqrt(3)*sqrt(2)*Vm*cosd(aa)/%pi
Eb2=Va2-(Ia1*Ra)
N3=Eb2/(K*If1)//Speed in rad/s
N3rpm=N3*60/(2*%pi)//Speed in Rpm
W2=N2*2*%pi/60
If2=Eb2/(K*W2)
Vf2=If2*Rf
af1=acosd(Vf2*%pi/(3*sqrt(3)*sqrt(2)*Vm))
printf('\n\n The Firing Angle=%0.1f\n\n',aa1)
printf('\n\n The Field Current=%0.1f Amp\n\n',N3rpm)
printf('\n\n The Field Current=%0.1f Amp\n\n',af1)
```

Scilab code Exa 2.9 Ex9

```
clc
//variable initialisation
V=440 //voltage in volts
P=100e+3 //power in Watts
N=900 //speed in rpm
V1=415 //supply voltage in volts
```

```

//solution
k=(3*sqrt(2))/%pi
a=acosd(V/(k*V1))
V2=0.5*V//At 50% of rated speed
a1=acosd(V2/(k*V1))
printf('\n\n The Firing Angle=%0.1f\n\n',a1)
```

Scilab code Exa 2.10 Ex10

```
clc
//variable initialization
P=80e+3 //power in Watt
Va1=440 //voltage in volts
N1=800 //speed in rpm
N2=600 //speed in rpm
Eb1=410 //Given back emf in volts
Vrms=415 //voltage in volts

//solution
Eb2=Eb1*(N2/N1)
Ia1=P/Va1
Ra=(Va1-Eb1)/Ia1
Ia2=0.75*Ia1//As motor is operating at 75% of rated
    torque
Va2=Eb2+(Ia2*Ra)
a=acosd(Va2*%pi/(3*sqrt(2)*Vrms))
printf('\n\n The Triggering Angle=%0.1f\n\n',a)
```

Scilab code Exa 2.11 Ex11

```
clc
//variable initialization
```

```
P=100e+3 //power in W
Va1=440 //Supply voltage in volts
N1=1000 //speed in rpm
N2=800 //speed in rpm
Eb1=410 //given Back EMF in volts
Vrms=415 //RMS voltage in volts
f=50 //Supply frequency in Hz

//solution
Eb2=Eb1*(N2/N1)
Ia1=P/Va1
Ra=(Va1-Eb1)/Ia1
Ia2=0.75*Ia1//At 75% of rated torque
Va2=Eb2+(Ia2*Ra)
a=acosd((Va2*%pi)/(3*sqrt(2)*Vrms))
printf('\n\n The Firing Angle=%0.1f\n\n',a)
```

Scilab code Exa 2.12 Ex12

```
clc
//variable initialization
Php=100 //power in hp
P=Php*735.5 //power in Watts
Va1=440 //voltage in volts
N1=1000 //speed in rpm
N2=500 //speed in rpm
Eb1=430//Given back EMF in volts
Vrms=415 //RMS voltage in volts

//solution
Eb2=Eb1*(N2/N1)
Ia1=P/Va1
Ra=10/Ia1
Ia2=0.85*Ia1
Va2=Eb2+Ia2*Ra//At 85% load and 500 rpm
```

```
a=acosd(Va2/(1.35*Vrms))
printf('\n\n The Firing Angle=%0.1f\n\n',a)
```

Scilab code Exa 2.13 Ex13

```
clc
//variable initialisation
Va=230 //Supply voltage in volts
N1=1400 //speed in rpm
N2=600 //speed in rpm
N3=1400 //speed in rpm
Vdrop=15//Voltage drop in Volts
//solution
Eb1=Va-15
Eb2=Eb1*(N2/N1)
Va1=Eb2+Vdrop
a2=acosd(Va1/Va)
printf('\n\n The Firing Angle=%0.1f\n\n',a2)
```

Chapter 3

Four Quadrant Operations of DC Drives

Scilab code Exa 3.1 Ex1

```
clc
//variable initialization
Vm= 220 //armature voltage in volts
N= 1000 //speed in rpm
N1= 900 // speed in rpm
Ia= 60 //armature current in ampere
Ra= 0.6 //armature resistance in ohm
a= 0
V= 165 //line voltage in volts

//solution
Eb1= Vm-Ia*Ra //back emf in volts
Eb2= (N1/N)*Eb1 //back emf in volts
Ea=Eb2+(Ia*Ra) //armature voltage in volts
Em= V*sqrt(2)
A=(((Ea*%pi)/(3*Em)))
a1=acosd(A)
a2=180-a1
Ea1=V-(Ia*Ra) //armature voltage in volts
```

```
cosa1=((Ea/Em)*(%pi/3))
a11=acosd((Ea1*%pi)/(3*Em))
a22=180-a1
printf('\n\n Firing Angle for motoring operations at rated motor torqu and 900 or -900 rpm=%0.1f \n\n',a1)
printf('\n\n Firing Angle for motoring operations at rated motor torqu and 900 or -900 rpm=%0.1f \n\n',a2)
printf('\n\n Firing Angle for braking operations at rated motor torqu and 900 or -900 rpm=%0.1f \n\n',a11)
printf('\n\n Firing Angle for braking operations at rated motor torqu and 900 or -900 rpm=%0.1f \n\n',a22)
//The answers vary due to round off error
```

Scilab code Exa 3.2 Ex2

```
clc
// Variable Initiallization
Vm=220 //armature voltage in volts
f=50 //frequency in Hz
Ra=10 //armature resistance in ohm
Lr=50e-3 //circulating inductance in mH
a1=30
a2=150

//solution
w=2*%pi*f
Em=sqrt(2)*220 //voltage in volts
cosa1=cosd(a1)
cosa2=cosd(a2)
Irmax1=((2*Em)/(w*Lr))*(1-cosa1)
Irmax2=((2*Em)/(w*Lr))*(1-cosa2)
```

```
Ip=(Em/Ra)
I1=Ip+Irmax1
I2=Ip+Irmax2
printf('\n\n Peak Current of Converter 1=%0.1f Amp\n
    \n',I1)
printf('\n\n Peak Current of Converter 2=%0.1f Amp\n
    \n',I2)
```

Scilab code Exa 3.3 Ex3

```

clc
// Variable initialization
F=50 //Supply Frequency In Hz
Vm=400 //Supply Voltage In Volts
Ip=20 //Peak Circulating Current In Ampere
A=60 //firing angle

// solution
Ea=Vm/(sqrt(3))
W=2*%pi*F
Lr=[(3*sqrt(2)*Ea)/(W*Ip)]*(1-sind(A))
Lr1=Lr*1000//Inductance in mH
printf('\n\n Inductance Value Needed=%0.1f mH\n\n',
    Lr1)
//The answer vary due to round off error
```

Scilab code Exa 3.4 Ex4

```
clc
// Variable Initiallization
F=50 //Supply Frequency In Hz
Erms=230 //RMS Voltage Per Phase In Volts
```

```
L=0.015 //Inductance In Henry
A1=60 //Firing Angle
A2=120 //Firing Angle




//solution
W=2*%pi*F
Icp=((3*sqrt(2)*Erms)/(W*L))*(1-sind(A1))
printf('\n\n The Peak value of Circulating Current=%0.1f Amp\n\n',Icp)
```

Scilab code Exa 3.5 Ex5

```
clc
// Variable Initiallization
Vm=400 //Supply Voltage In Volt
Ea1= 220// Voltage Of Motor In Volt
Ia=200 //Line Current In Ampere
Ra=0.05 //Armature Resistance In Ohm
N1=750 //Speed Of Motor In rpm
N2=600 //Speed Of Motor In rpm

//solution
Eb1=Ea1-(Ia*Ra)
Eb2=(N2/N1)*Eb1
Ea2=Eb2+(Ia*Ra)
A=acosd((Ea2*%pi)/(Vm*3*sqrt(2)))
printf('\n\n The Firing Angle of Rectifier=%0.1f\n\n',A)
```

Scilab code Exa 3.6 Ex6

```
clc
// Variable Initiallization
Ea=400 // Voltage Of MOtor In Volt
Ia1=70 //Line Current In Ampere
Ia2=90 //Line Current In Ampere
Ra=0.3 //Armature Resistance In Ohm
N1=750 //Speed Of Motor In rpm
N2=300 //Speed Of Motor In rpm

//Solution
Eb1=Ea-(Ia1*Ra)
Eb2=(N2/N1)*Eb1
Rb=-((Eb2-Ea-Ia2*Ra)/Ia2)//Wrongly calculated in book,wrong value of Eb2 is taken
W1=(2*%pi*N1)/60
Kt1=Eb1/W1
T1=Kt1*Ia1
W2=(2*%pi*N2)/60
Kt2=Eb2/W2//Wrongly computed in textbook
T2=Kt2*Ia2//The answer provided in the textbook is wrong
printf('\n\n External resistance to be added=%0.1f ohm\n\n',Rb)
printf('\n\n Initial braking torque=%0.1f N-m\n\n',T1)
printf('\n\n braking torque at 300 rpm=%0.1f N-m\n\n',T2)
//The answer provided in the textbook is wrong(both)
```

Chapter 4

Chopper Controlled DC Drives

Scilab code Exa 4.1 Ex1

```
clc
//Variable Initialisation
V=230//Input Voltage of motor in volts
Vdc=240//Dc equivalent input to motor in Volts
Po=746//Power rating of DC motor in Watt
N=500//Rated Speed of Motor in rpm
Ia=4.1//Armature Current in Ampere
Ra=7.56//Armature resistance in ohm
La=55e-3//Armature inductance in Henry
f=500//Chopper Frequency
Tmin=5//minimum load Torque in N-m
//Solution
T=(1/f)//Period of Chopper
w=2*%pi*N*(1/60)
Eb=V-(Ia*Ra)//Back emf in Volts
k=Eb*(1/w)
Pin=V*Ia//Armature Power Input
L=Pin-Po-(Ia*Ia*Ra)//Rotational Loss
Ta=L/w
Ta1=Tmin+Ta//Average internal Torque in N-m
Ia1=Ta1/k
```

```
E0=Eb+(Ia1*Ra)
ton=(E0/Vdc)*T
ton11=ton*1000//ton in microseconds
ta=(La/Ra)
A=log(((Eb/Vdc)*(%e^(T/ta)-1))+1)
ton1=A*ta
ton12=ton1*1000//ton in microseconds
E01=(ton1/T)*Vdc
Ia2=(E01-Eb)/Ra
Ta2=k*Ia2
Tc=Ta2-Ta
printf('\n\n ton for minimum load torque of 500rpm=
    %0.1f 10^(-3)sec\n\n',ton11)
printf('\n\n ton for current is continuous at 500rpm
    =%0.1f 10^(-3)sec\n\n',ton12)
if ton>ton1 then
    disp("Current (Ia) is continuous")
else
    disp("Current (Ia) is not continuous")
end
printf('\n\n The coupling Torque for minimum value
    of ton obtain=%0.1f N-m\n\n',Tc)
```

Scilab code Exa 4.2 Ex2

```
clc
//Variable Initialisation
V=230//Input Voltage of motor in volts
N=1750//Rated Speed of Motor in rpm
Ia=74//Armature Current in Ampere
Ra=0.180//Armature resistance in ohm
Vdc=240//Dc equivalent input to motor in Volts
f=500//Chopper Frequency
W0=2*f*%pi
la=2.93*10^(-3)//Armature inductance in Henry
```

```
//Solution
T=1/f//Period of Chopper
I0=Ia
W=2*%pi*N/60
Eb=V-(Ia*Ra)//Back EMF in Volts
k=Eb/W
Ea=Vdc/2//Average Voltage
Eb1=Ea-(Ia*Ra)
W1=Eb1/k
N1=W1*(60/(2*%pi))
ton=T/2
Irms=((sqrt(2)*Vdc)/(%pi*W0*la))*sin(W0*ton/2)
Irms1=sqrt((I0^2)+(Irms^2))
k1=Irms/I0
I01=Ia/2//Average Value of Source Current
Irms2=sqrt(2)*Ia/%pi
k2=Irms2/I01//Source Current Ripple Factor
printf('\n\n The Motor Speed=%0.1f rpm\n\n',N1)
printf('\n\n The RMS Armature Current=%0.1f Amp\n\n',Irms1)
printf('\n\n The RMS and line current ripple factor=%0.1f\n\n',k2)
```

Scilab code Exa 4.3 Ex3

```
clc
//Variable Initialisation
Ia=50//Armature Current in Ampere
Ea=440//Input Voltage to armature in volts
N=1000//Rated Speed of Motor in rpm
Ra=0.5//Armature resistance in ohm
Ra1=10.5//Armature resistance in ohm
Rf=100//Field resistance in ohm
N1=400//Speed of Motor in rpm
N2=800//Speed of Motor in rpm
```

```
ton=2*10^(-3)
//Solution
If=Ea/Rf
Eb=Ea-(Ia*Ra)
Eb1=(N1/N)*Eb
E01=Eb1+(Ia*Ra1)
t1=(Ea/E01)*2*10^3
f1=1/t1
Eb2=(N2/N)*Eb
E02=Eb2+(Ia*Ra)
t2=(Ea/E02)*ton
f2=1/t2
printf('\n\n The Chopping Frequency 1=%0.1f\n\n',f1)
printf('\n\n The Chopping Frequency 2=%0.1f\n\n',f2)
//The answers vary due to round off error
```

Scilab code Exa 4.4 Ex4

```
clc
//Variable Initialisation
V=230//Input Voltage of motor in volts
Ra=1.5//Armature resistance in ohm
La=1e-3//Armature inductance in ohm
Ia=15//Armature Current in Ampere
k=0.05//Voltage constant
//Solution
Eb=0 //when d=0
Ea=Eb+(Ia*Ra)
d=Ea/V
Eb1=V-(Ia*Ra)
//when d1=1
N=Eb1/k
printf('\n\n Range of speed control is from 0 to %0
    .1f\n\n',N)
printf('\n\n The Duty Cycle is from%0.1f to 1\n\n',d
```

```
)
//The answers vary due to round off error
```

Scilab code Exa 4.5 Ex5

```
clc
//Variable Initialisation
Ea=220//Input Voltage to armature in volts
N1=1000//Rated Speed of Motor in rpm
N2=500//Speed of Motor in rpm
Ia=24//Armature Current in Ampere
Ra=2//Armature resistance in ohm
Es=230//Source voltage in Volts
//Solution
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
E0=Eb2+(1.2*Ia*Ra)
d=E0/Es
printf('\n\n The Duty Ratio=%0.1f\n\n',d)
//The answers vary due to round off error
```

Scilab code Exa 4.6 Ex6

```
clc
//Variable Initialisation
Ea=500//Input Voltage to armature in volts
Ra=0.09//Armature resistance in ohm
If=3//Field Current in Ampere
K=1.527//Voltage constant
T=560//Torque Developed in N-m
N1=0//Speed of Motor in rpm
d2=1//duty ratio
//Solution
```

```
Ia=T/(K*If)
Eb=K*N1
d1=(Eb+(Ia*Ra))/Ea
N2=((d2*Ea)-(Ia*Ra))/(K*If)
N2r=N2*60/(2*%pi)
d3=[0.2,0.4,0.6,0.8,1.0]
N3r=[556.56 ,1181.92,1807.28 ,2432.6,3058.0]
plot(d3,N3r)
xlabel ('Duty Interval')
ylabel ('Speed in RPM')
```

Scilab code Exa 4.7 Ex7

```
clc
//Variable Initialisation
Ra=0.08//Armature resistance in ohm
Ea=450//Input Voltage to armature in volts
Ia=275//Armature Current in Ampere
If=3//Field Current in Ampere
K=1.527//Voltage constant
d=0.65//Duty ratio
//Solution
Pin=d*Ea*Ia
E0=d*Ea
Eb=E0-(Ia*Ra)
W=Eb/(K*If)
N=W*60/(2*%pi)
T=K*If*Ia
printf('\n\n The Input power from Generator Source=
    %0.1f Watt\n\n',Pin)
printf('\n\n The Speed of Motor=%0.1f rpm\n\n',N)
printf('\n\n The Torque developed=%0.1f N-m\n\n',T)
```

Scilab code Exa 4.8 Ex8

```
clc
//Variable Initialisation
ton=15
toff=10
Ea=220//Input Voltage to armature in volts
Km=0.4//Voltage constant
N=1400//Rated Speed of Motor in rpm
Ra=2//Armature resistance in ohm
//Solution
d=ton/(ton+toff)
E0=d*Ea
W=2*%pi*N/60
Ia=(E0-(Km*W))/Ra
printf('\n\n The Average load Current=%0.1f Amp\n\n'
    ,Ia)
```

Scilab code Exa 4.9 Ex9

```
clc
//Variable Initialisation
Ea=200//Input Voltage to armature in volts
Ia=20//Armature Current in Ampere
Ra=0.4//Armature resistance in ohm
k=0.1
N1=0//Speed of Motor in rpm
//Solution
Eb1=k*N1
d1=(Eb1+(Ia*Ra))/Ea
d2=1
Eb2=d2*Ea-(Ia*Ra)
N2=Eb2/k
printf('\n\n Range of speed control from 0 to %0.1f\
    n\n',N2)
```

```
printf('\n\n Range of duty cycle from %0.1f to 1\n\n
    ',d1)
//The answers vary due to round off error
```

Scilab code Exa 4.10 Ex10

```
clc
//Variable Initialisation
V=230//Input Voltage of motor in volts
f=300//Chopper Frequency
Tl=40//Load Torque in N-m
N1=900//Rated Speed of Motor in rpm
Ra=0//Armature resistance in ohm
La=12e-3//Inductance in Henry
k=2//Motor Constant
//Solution
Ia=Tl/k
W=2*%pi*N1/60
Eb=k*W
d=(Eb+(Ia*Ra))/V
t1=1/f
ton=d*t1
toff=(1-d)*t1
Z1=(V-Eb)/La
Z2=-Eb/La
A=Z1*ton //A=Imax-Imin
B=2*Ia //B=Imax+Imin
Imax=(A+B)/2
Imin=(B-A)/2

t=poly(0,'t');
x=Imin+Z1*t
y=Imax+Z2*t

disp (Imax ,"Maximum Armature Current in Amp is")
```

```
disp (Imin ,"Minimum Armature Current in Amp is")
disp (A ,"Armature Current Excursion in Amp is")
disp (x ,"Armature Current During Ton in Amp is")
disp (y ,"Armature Current During Toff in Amp is")
```

Scilab code Exa 4.11 Ex11

```
clc
//Variable Initialisation
Ea=200//Input Voltage of motor in volts
Ra=0.12//Armature resistance in ohm
La=12e-3//Armature Inductance in ohm
K=2//Motor constant in V-s/rad
Eb=150//Motor back EMF
Ia=30//Armature Current in Ampere
f=300//Chopper Frequency
//Solution
T=1/f
d=(Eb+(Ia*Ra))/Ea
ton=d*T
toff=(1-d)*T
t=Ra/La
Ea1=Ea
Imin=poly(0,'Imin');
Ia1=((Ea1-Eb)/Ra)*(1-%e^(-ton*t))+(Imin*%e^(-ton*t))
disp (Ia1 ,"Imax is")
Ea2=0
Imax=poly(0,'Imax');
Ia2=((Ea2-Eb)/Ra)*(1-%e^(-toff*t))+(Imax*%e^(-toff*t))
disp (Ia2 ,"Imin is")
a=poly(0,'a');
b=poly(0,'b');
Imax1=(10.409+(0.975*(-9.96)))/(1-(0.975*0.992))//From above displayed values and rounding off
```

```
Imin1=(-9.960)+(0.992*Imax1)
Im=Imax1-Imin1//Armature Current Excursion
printf('\n\n Maximum Armature Current=%0.1f Amp\n\n',Imax1)
printf('\n\n Minimum Armature Current=%0.1f Amp\n\n',Imin1)
printf('\n\n Armature Current Excursion=%0.1f Amp\n\n',Im)
```

Scilab code Exa 4.12 Ex12

```
clc
//Variable Initialisation
V=440//Input Voltage of motor in volts
Rf=100//Field resistance in ohm
Il=50//Load Current in Ampere
N1=900//Rated Speed of Motor in rpm
N2=300//Rated Speed of Motor in rpm
N3=400//Rated Speed of Motor in rpm
N4=600//Rated Speed of Motor in rpm
Ra=0.3//Armature resistance in ohm
ton=4e-3//On period of Chopper in sec
//Solution
If=V/Rf//Motor Field Current in Amp
Ia=Il-If//Armature Current in Amp
Eb1=V-(Ia*Ra)//Back EMF of Motor
Eb2=(N2/N3)*Eb1
V2=Eb2+(Ia*Ra)//Required Terminal Voltage in volts
T1=(V/V2)*ton//Chopping Period
f1=1/T1///Chopping Period
Eb3=(N4/N1)*Eb1//Back Emf at 600 rpm
V3=Eb3+(Ia*Ra)//Required Terminal Voltage in volts
T2=(V/V3)*ton//Chopping Period
f2=1/T2//Chopping Period
printf('\n\n Frequency of chopper=%0.1f Hz\n\n',f2)
```

Scilab code Exa 4.13 Ex13

```
clc
//Variable Initialisation
ton=10//On time of Chopper
toff=12//Off time of Chopper
Ea=220//Input Voltage of motor in volts
k=0.495//Motor Voltage constant
W=146.60//Rated Speed of Motor in rad/sec
Ra=2.87//Armature resistance in ohm
//Solution
d=ton/(ton+toff)//Duty cycle ratio
Ia=((d*Ea)-(k*W))/Ra
printf('\n\n Average load Current=%0.1f Amp\n\n',Ia)
//The answers vary due to round off error
```

Scilab code Exa 4.14 Ex14

```
clc
//Variable Initialisation
Ea=450//Input Voltage of motor in volts
Ra=0.06//Armature resistance in ohm
Kt=1.4//Motor Voltage Constant
Ia=300//Armature Current in Ampere
If=3.3//Motor Field Current in Amp
d=0.7//Duty cycle of Converter
//Solution
Pin1=Kt*Ea*Ia//Input Power
Re1=Ea/(Kt*Ia)//Equivalent Resistance
E01=Kt*Ea
Eb1=E01-(Ia*Ra)
```

```

Pin2=d*Ea*Ia
Re2=Ea/(d*Ia)
E02=d*Ea
Eb2=E02-(Ia*Ra)
N1=Eb2/(Kt*If)
N=N1*60/(2*%pi)
T=Kt*Ia*If
printf('\n\n Input Power=%0.1f KW\n\n',Pin1*10^-3)
printf('\n\n Equivalent Resistance developed=%0.1f ohm\n\n',Re1)
printf('\n\n Motor Speed=%0.1f rpm\n\n',N)
printf('\n\n Motor Torque=%0.1f N-m\n\n',T)
```

Scilab code Exa 4.15 Ex15

```
clc
//Variable Initialisation
Ea=210//Input Voltage of motor in volts
Ia=25//Armature Current in Ampere
Es=230
N1=1500//Rated Speed of Motor in rpm
Ra=3//Armature resistance in ohm
N2=800//Rated Speed of Motor in rpm
//Solution
Ia2=1.5*Ia
Eb=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb
E0=Eb2+(Ia2*Ra)
d=E0/Es
printf('\n\n Duty Ratio=%0.1f\n\n',d)
```

Scilab code Exa 4.16 Ex16

```
clc
//Variable Initialisation
Ea=200//Input Voltage of motor in volts
Ia=20//Armature Current in Ampere
Ra=0.33//Armature resistance in ohm
La=11e-3//Armature Inductance in ohm
N1=1200//Rated Speed of Motor in rpm
N2=800//Rated Speed of Motor in rpm
f=500//Chopper Frequency in Hz
//Solution
T=1/f
t=Ra/La
t1=1/t
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
E0=Eb2+(Ia*Ra)
d=E0/Ea
ton1=d*T
A=log(1+((Eb2/Ea)*((%e^(T/t1))-1)))//Ia2=0 & A=ton2/t
ton2=A*t1
printf('\n\n Duty Cycle=%0.1f\n\n',ton2)
//The answer provided in the textbook is wrong(answer given in textbook is in invalid range)
if ton2<ton1 then disp('Current is Continuous')
end
```

Scilab code Exa 4.17 Ex17

```

clc
//Variable Initialisation
Ea=220//Input Voltage of motor in volts
Ia=100//Armature Current in Ampere
Ra=0.01//Armature resistance in ohm
```

```
N1=1000//Rated Speed of Motor in rpm
N2=500//Rated Speed of Motor in rpm
//Solution
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
Ea2=Eb2+(Ia*Ra)
d1=Ea2/Ea
Ea3=Eb2-(Ia*Ra)
d2=Ea3/Ea
printf('\n\n Duty Ratio of Chopper in motoring operation=%0.1f\n\n',d1)
printf('\n\n Duty Ratio of Chopper in breaking operation=%0.1f\n\n',d2)
//The answers vary due to round off error
```

Scilab code Exa 4.18 Ex18

```
clc
//Variable Initialisation
Ea=230//Input Voltage of motor in volts
Ia=50//Armature Current in Ampere
N1=800//Rated Speed of Motor in rpm
Ra=0.4//Armature resistance in ohm
d1=0.3//Duty ratio for Motoring Operation
d2=0.6//Duty ratio for Motoring Operation
d3=0.7//Duty ratio for Braking Operation
d4=0.4//Duty ratio for Braking Operation
//Solution
E01=d1*Ea
Eb1=Ea-(Ia*Ra)
Eb2=E01-(Ia*Ra)
N2=(Eb2/Eb1)*N1
E02=d2*Ea
Eb3=E02-(Ia*Ra)
N3=(Eb3/Eb1)*N1
```

```
E03=d3*Ea
Eb4=E03+(Ia*Ra)
N4=(Eb4/Eb1)*N1
E04=d4*Ea
Eb5=E04+(Ia*Ra)
N5=(Eb5/Eb1)*N1
printf('\n\n Motor speed for Motoring Operation 1 =
    %0.1f rpm\n\n',N2)
printf('\n\n Motor speed for Motoring Operation 2=%0
    .1f rpm\n\n',N3)
printf('\n\n Motor speed for Braking Operation 1=%0
    .1f rpm\n\n',N4)
printf('\n\n Motor speed for Braking Operation 2=%0
    .1f rpm\n\n',N5)
```

Scilab code Exa 4.19 Ex19

```
clc
//Variable Initialisation
Ea=220//Input Voltage of motor in volts
d=0.95//Maximum Duty Ratio
Ia1=100//Armature Current in Ampere
Ia2=150//Armature Current in Ampere
Ra=0.01//Armature resistance in ohm
N1=1000//Rated Speed of Motor in rpm
//Solution
Eb1=Ea-(Ia1*Ra)
E0=d*Ea
Eb2=E0+(Ia2*Ra)
N2=(Eb2/Eb1)*N1
Pin=E0*Ia2
printf('\n\n Maximum Permissible MOtor Speed=%0.1f
    rpm\n\n',N2)
//The answers vary due to round off error
```

Scilab code Exa 4.20 Ex20

```
clc
//Variable Initialisation
Ea=230//Input Voltage of motor in volts
Ia=30//Armature Current in Ampere
Ia2=60//Armature Current in Ampere
N1=1000//Rated Speed of Motor in rpm
N2=800//Rated Speed of Motor in rpm
Ra=0.7//Armature resistance in ohm
d2=0.6//Duty Ratio
d3=0.9//Duty Ratio
d4=0.9//Duty Ratio
//Solution
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
E01=Eb2-(Ia*Ra)
d1=E01/Ea
E02=d2*Ea
Eb3=E02+(Ia*Ra)
N3=(Eb3/Eb1)*N1
E03=d3*Ea
Eb4=E03+(Ia2*Ra)
N4=(Eb4/Eb1)*N1
E04=d4*Ea
Pin=E04*Ia2
printf('\n\n Duty Ratio Of Chopper=%0.1f\n\n',d1)
printf('\n\n Motor Speed for duty ratio 0.6=%0.1f
    rpm\n\n',N3)
printf('\n\n Maximum Aloowable Speed=%0.1f rpm\n\n',
    N4)
printf('\n\n Power Fed to Source=%0.1f KW\n\n',Pin
    *10^-3)
//The answers vary due to round off error
```

Scilab code Exa 4.21 Ex21

```
clc
//Variable Initialisation
Ea=220//Input Voltage of motor in volts
Ia=150//Armature Current in Ampere
Ra=0.06//Armature resistance in ohm
N1=1000//Rated Speed of Motor in rpm
N2=500//Rated Speed of Motor in rpm
N4=1350//Rated Speed of Motor in rpm
d3=0.91//Duty Ratio
Ia2=2*Ia
//Solution
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
E01=Eb2+(Ia*Ra)
d1=E01/Ea
E02=Eb2-(Ia*Ra)
d2=E02/Ea
E03=d3*Ea
Eb3=E03+(Ia2*Ra)
N3=(Eb3/Eb1)*N1
Pin=E03*Ia2
R=N1/N4 //Ratio of If1 and If2
printf('\n\n Duty ratio for motoring operation=%0.1f
    \n\n',d1)
printf('\n\n Duty ratio for braking operation=%0.1f\
    n\n',d2)
printf('\n\n Maximum permissible motor Speed=%0.1f
    rpm\n\n',N3)
printf('\n\n Ratio of If1 and If2=%0.1f\n\n',R)
```

Scilab code Exa 4.22 Ex22

```
clc
//Variable Initialisation
Ea=220//Input Voltage of motor in volts
Ia=100//Armature Current in Ampere
Ia2=1.5*Ia
Ra=0.01//Armature resistance in ohm
Rb=2
N1=1000//Rated Speed of Motor in rpm
N2=500//Rated Speed of Motor in rpm
d2=0.5
//Solution
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
d=(1-(((Eb2/Ia2)-Ra)/Rb))//Wrongly solved in textbook
Eb3=Ia2*(((1-d2)*Rb)+Ra)
N3=(Eb3/Eb1)*N1
printf('\n\n Duty Ratio of chopper=%0.1f\n\n',d)//The answer provided in the textbook is wrong
printf('\n\n Motor Speed=%0.1f rpm\n\n',N3)
```

Scilab code Exa 4.23 Ex23

```
clc
//Variable Initialisation
Ea=220//Input Voltage of motor in volts
Ia=150//Armature Current in Ampere
Ia2=300 //torque is doubled
Ra=0.06//Armature resistance in ohm
Rb=2.2
N1=1000//Rated Speed of Motor in rpm
N2=700//Rated Speed of Motor in rpm
d2=0.55
```

```
//Solution
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
d=(1-(((Eb2/Ia2)-Ra)/Rb))
Eb3=Ia2*(((1-d2)*Rb)+Ra)
N3=(Eb3/Eb1)*N1
printf('\n\n Duty Ratio Of Chopper=%0.1f\n\n',d)
printf('\n\n Motor Speed=%0.1f rpm\n\n',N3)
```

Scilab code Exa 4.24 Ex24

```
clc
//Variable Initialisation
Ea=230//Input Voltage of motor in volts
N1=1200//Rated Speed of Motor in rpm
N2=1000//Rated Speed of Motor in rpm
Ia=15//Armature Current in Ampere
Ia2=1.5*Ia
Ra=1.2//Armature resistance in ohm
Rb=20
d2=0.5
//Solution
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
d1=(1-(((Eb2/Ia2)-Ra)/Rb))
Eb3=Ia*(((1-d2)*Rb)+Ra)
N3=(Eb3/Eb1)*N1
printf('\n\n Duty Ratio Of Chopper=%0.1f\n\n',d1)
printf('\n\n Motor Speed=%0.1f rpm\n\n',N3)
```

Scilab code Exa 4.25 Ex25

```
clc
```

```
//Variable Initialisation
Ia=180//Armature Current in Ampere
Ra=0.06//Armature resistance in ohm
Rb=8
If=2//Field Current in Ampere
d=0.5
K=1.527
//Solution
E0=Ia*Rb*(1-d)
Req=Rb*(1-d)+Ra
Pb=(Ia^2)*(Rb*(1-d))
Eb=E0+(Ia*Ra)
W=Eb/(K*If)
W1=(W*60)/(2*%pi)
Ep=Ia*Rb
printf('\n\n The Average Voltage across chopper=%0.1
    f Volts\n\n',E0)
printf('\n\n Equivalent Resistance of motor=%0.1f
    ohm\n\n',Req)
printf('\n\n Power dissipated in braking resistor=%0
    .1f KW\n\n',Pb*10^-3)
printf('\n\n The Motor Speed=%0.1f rpm\n\n',W1)
printf('\n\n Peak to Peak Voltage=%0.1f Volts\n\n',
    Ep)
```

Scilab code Exa 4.26 Ex26

```
clc
//Variable Initialisation
Ea=440//Input Voltage of motor in volts
d=0.5//Duty Ratio
Ia=200//Armature Current in Ampere
Ra=0.15//Armature resistance in ohm
K=1//Motor Constant
//Solution
```

```
E0=(1-d)*Ea
Pr=E0*Ia
Wmin1=(Ia*Ra)/K
Wmin=Wmin1*60/(2*%pi)
Wmax1=(Ea+(Ia*Ra))/K
Wmax=Wmax1*60/(2*%pi)
Eb=E0+(Ia*Ra)
Wm1=Eb/K
Wm=Wm1*60/(2*%pi)
printf('\n\n The Power Returned=%0.1f KW\n\n',Pr*10^-3)
printf('\n\n Minimum braking Speed=%0.1f rpm\n\n',Wmin)
printf('\n\n Maximum braking Speed=%0.1f rpm\n\n',Wmax)//The answers vary due to round off error
printf('\n\n Speed during Regenerative Braking=%0.1f rpm\n\n',Wm)
```

Scilab code Exa 4.27 Ex27

```
clc
//Variable Initialisation
Ia1=190//Armature Current in Ampere
Ia2=0.9*Ia1
Ra=0.08//Armature resistance in ohm
Ri=0.05
Ea=210//Input Voltage of motor in volts
N1=950//Rated Speed of Motor in rpm
N2=750//Rated Speed of Motor in rpm
//Solution
Eb1=Ea-(Ia1*Ra)
Eb2=(N2/N1)*Eb1
Vi=Eb2-(Ia2*(Ra+Ri))
printf('\n\n Internal Voltage of Source=%0.1f Volts\n\n',Vi)
```

Scilab code Exa 4.28 Ex28

```
clc
//Variable Initialisation
Ea=210//Input Voltage of motor in volts
Ia1=140//Armature Current in Ampere
Ia2=2*Ia1
Ra=0.08//Armature resistance in ohm
N1=1100//Rated Speed of Motor in rpm
N2=1200//Rated Speed of Motor in rpm
//Solution
Eb1=Ea-(Ia1*Ra)
Eb2=(N2/N1)*Eb1
Rb=((Eb2+Ea)/Ia2)-Ra
W=(2*%pi*N2)/60
T1=(Eb2*Ia2)/W
Ia3=Ea/(Ra+Rb)
T2=T1*(Ia3/Ia2)
printf('\n\n Resistance to be placed=%0.1f ohm\n\n',
    Rb)
printf('\n\n Braking torque=%0.1f N-m\n\n',T1)
printf('\n\n torque=%0.1f N-m\n\n',T2)//The answers
    vary due to round off error
```

Scilab code Exa 4.29 Ex29

```
clc
//Variable Initialisation
Ra=0.08//Armature resistance in ohm
Rb=1.5
Rf=12
```

```
N=500//Rated Speed of Motor in rpm
//Solution
If0=poly(0,'If0')
Eb0=poly(0,'Eb0')

If=[4.16,6.2,8.33,10.5,12.5,14.6,16.6,18.8,20]
Eb=[41.6,61.2,75,85,92,96.6,101,105,125]
W=2*%pi*N/60
Ia=((Rb+Rf)/Rb)*If0
K=Eb0*(1/W)
If1=12.6
Eb1=102.2
for If0=If1
    disp(Ia)
end
for Eb0=Eb1
    disp(K)
end
If2=12.6
K2=1.75
Eb2=102.2
K1=Eb2*(1/W)
Ia2=9*If2
Eb3=(If2*Rf)+(Ia2*Ra)//Wrongly calculated in book
N2=Eb3*60/(K1*2*%pi)
printf('\n\n Motor Speed at which load is hold by motor=%0.1f rpm\n\n',N2)//The answer provided in the textbook is wrong
```

Scilab code Exa 4.30 Ex30

```
clc
//Variable Initialisation
Ra=0.08//Armature resistance in ohm
T=300//Torque in N-m
```

```
N=1000//Rated Speed of Motor in rpm
Rf=12//Field Winding Resistor in ohm
//Solution
Eb=poly(0,'Eb')
W=2*%pi*N/60
Pd=T*W
Ea=Eb-((Pd*Ra)/Eb)
If=20//From previous Example
Ea1=Rf*If
Eb1=250//From previous Example
Ia=(Eb1-Ea1)/Ra
If1=Ea1/Rf
Ir=Ia-If
Rb=Ea1/Ir
printf('\n\n Braking Resistance=%0.1f ohm\n\n',Rb)
```

Scilab code Exa 4.31 Ex31

```
clc
//Variable Initialisation
Ea=230//Input Voltage of motor in volts
Ia=200//Armature Current in Ampere
Ra=0.02//Armature resistance in ohm
N1=960//Rated Speed of Motor in rpm
N2=350//Rated Speed of Motor in rpm
N4=1200//Rated Speed of Motor in rpm
d3=0.95
//Solution
Ia2=2*Ia
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
E01=Eb2+(Ia*Ra)
d1=E01/Ea
Ea2=Eb2-(Ia*Ra)
d2=Ea2/Ea
```

```
Eam=d3*Ea
P=Eam*Ia2
Eb3=Eam+(Ia2*Ra)
N3=(Eb3/Eb1)*N1
Ifr=(N1/N4)
printf('\n\n Duty ratio for motoring operation=%0.1f\n\n',d1)
printf('\n\n Duty ratio for braking operation=%0.1f\n\n',d2)
printf('\n\n Maximum permissible motor Speed=%0.1f rpm\n\n',N3)
printf('\n\n Field Current as Ratio of its rated value=%0.1f\n\n',Ifr)
```

Scilab code Exa 4.32 Ex32

```
clc
//Variable Initialisation
Ea=230//Input Voltage of motor in volts
Ia=10//Armature Current in Ampere
Ia2=2*Ia//Given condition for armature Current
Ia3=2*Ia//Given condition for armature Current
Ra=1.5//Armature resistance in ohm
Rb=15//Braking Resistance in ohm
N1=1500//Rated Speed of Motor in rpm
N2=1200//Rated Speed of Motor in rpm
//Solution
Eb1=Ea-(Ia*Ra)
Eb2=(N2/N1)*Eb1
d1=1-(((Eb2/Ia2)-Ra)/Rb)
d2=0.6//Duty ratio
Eb3=Ia3*(((1-d2)*Rb)+Ra)
N3=(Eb3/Eb1)*N1
printf('\n\n Duty ratio =%0.1f\n\n',d1)
printf('\n\n motor Speed=%0.1f rpm\n\n',N3)
```

Scilab code Exa 4.33 Ex33

```
clc
//Variable Initialisation
Ea=230//Input Voltage of motor in volts
Ia=50//Armature Current in Ampere
Ra=0.4//Armature resistance in ohm
N1=800//Rated Speed of Motor in rpm
//Solution
T=poly(0,'T')
W=2*%pi*N1/60
Eb=Ea-(Ia*Ra)
K=Eb/W
d1=0.3
W1=((d1*Ea)/K)-(Ra/(K^2))*T
d2=0.6
W2=((d2*Ea)/K)-(Ra/(K^2))*T
d3=0.7
W3=((d3*Ea)/K)+(Ra/(K^2))*T
d4=0.4
W4=((d4*Ea)/K)+(Ra/(K^2))*T
disp(W1,'Speed in terms of torque for motoring
    operation for duty ratio 0.3')
disp(W2,'Speed in terms of torque for motoring
    operation for duty ratio 0.6')
disp(W3,'Speed in terms of torque for Braking
    operation for duty ratio 0.7')
disp(W4,'Speed in terms of torque for Braking
    operation for duty ratio 0.4')
```

Scilab code Exa 4.34 Ex34

```
clc
//Variable Initialisation
Ea=230//Input Voltage of motor in volts
Ia=15//Armature Current in Ampere
Ra=1.2//Armature resistance in ohm
Rb=20//Braking Resistance in ohm
Ia2=15*Ia
N1=1200//Motor Speed in rpm
//Solution
Eb=Ea-(Ia*Ra)
Eb1=Ea-(Ia2*Ra)
d1=1-(((Eb1/Ia2)-Ra)/Rb)
d2=0.5//Duty ratio
Eb2=Ia*(((1-d2)*Rb)+Ra)
N2=(Eb2/Eb)*N1
printf('\n\n Duty ratio Of Chopper=%0.1f\n\n',d1)
printf('\n\n Motor Speed=%0.1frpm\n\n',N2)
```

Scilab code Exa 4.35 Ex35

```
clc
//Variable Initialisation
Ea=400//Input Voltage of motor in volts
Ia=200//Armature Current in Ampere
d=0.5//Duty Ratio
Ra=0.03//Armature resistance in ohm
Rs=0.05
K=3e-3//Motor Constant
//Solution
E0=d*Ea
Pin=E0*Ia//Input power in watt
R=Ra+Rs
Eb=E0-(Ia*R)
Wm=Eb/(K*Ia)
Wmrpm=Wm*30/%pi
```

```
T=K*(Ia^2)
printf('\n\n Input Power From Source=%0.1f KW\n\n',Pin*10^-3)
printf('\n\n Motor Speed=%0.1f rpm\n\n',Wmrpm)
printf('\n\n Motor Torque=%0.1f N-m\n\n',T)
```

Scilab code Exa 4.36 Ex36

```
clc
//Variable Initialisation
Ea=400//Input Voltage of motor in volts
Ia=200//Armature Current in Ampere
Ra=0.05//Armature resistance in ohm
Rs=0.07
d=0.5//Duty Ratio
K=5e-3//Motor Constant
//Solution
E0=d*Ea
Pin=E0*Ia
Wm=((E0-Ia*(Ra+Rs))/(K*Ia))*(30/%pi)
T=K*(Ia^2)
printf('\n\n Input Power From Source=%0.1f KW\n\n',Pin*10^-3)
printf('\n\n Motor Speed=%0.1f rpm\n\n',Wm)
printf('\n\n Motor Torque=%0.1f N-m\n\n',T)
```

Scilab code Exa 4.37 Ex37

```
clc
//Variable Initialisation
Ea=210//Input Voltage of motor in volts
Ia1=80//Armature Current in Ampere
N1=1200//Rated Speed of Motor in rpm
```

```
Ra=0.08//Armature resistance in ohm
Rf=0.08
N2=1000
//Solution
T1=poly(0,'T1')
T2=2*T1
A=T2/T1
Ia2=Ia1*(sqrt(2))//A=2
Eb1=Ea-(Ia1*Ra)
Eb2=Ia2*N2*Eb1/(Ia1*N1)
Rb=(Eb2/Ia2)-Ra
printf('\n\n Braking=%0.1f Amp\n\n',Ia2)
printf('\n\n Braking Resistor=%0.1f ohm\n\n',Rb)
```

Scilab code Exa 4.38 Ex38

```
clc
//Variable Initialisation
Ea=500//Input Voltage of motor in volts
Ra=0.06//Armature resistance in ohm
Rf=0.09//Field Resistance in ohm
K=35e-3//Motor Constant
T=560//Rated Torque in N-m
N1=0//Rated Speed of Motor in rpm
d2=1//Duty ratio
//Solution
Ia=sqrt(T/K)//Armature Current in Ampere
d1=(Ia*(Ra+Rf)+K*Ia*N1)/Ea
N2=(d2*Ea-Ia*(Ra+Rf))/(K*Ia)
N2rpm=N2*30/%pi
d3=0.6
N3=((d3*Ea-Ia*(Ra+Rf))/(K*Ia))*30/%pi
d4=0.8
N4=((d4*Ea-Ia*(Ra+Rf))/(K*Ia))*30/%pi
d=[d1,d3,d4,d2]
```

```
N=[N1,N3,N4,N2rpm]
plot(d,N)
ylabel("Speed in rpm", "fontsize", 2)
xlabel("Duty Ratio", "fontsize", 2)
```

Scilab code Exa 4.39 Ex39

```
clc
//Variable Initialisation
Ea=600//Input Voltage of motor in volts
Ia=500//Armature Current in Ampere
d1=0.6//Duty Ratio
Ra=0.05//Armature resistance in ohm
Rf=0.07//Field Resistance in ohm
K=15.27e-3//Motor Constant
//Solution
E0=d1*Ea
Pin=E0*Ia
Re=Ea/(Ia*d1)
Eb=E0-(Ia*(Ra+Rf))
W=Eb/(Ia*K)
N=W*60/(2*%pi)
T=K*(Ia^2)
printf('\n\n Input Power From Source=%0.1f KW\n\n',
   Pin*10^-3)
printf('\n\n Equivalent Output Resistor=%0.1f ohm\n\
   n',Re)
printf('\n\n Motor Speed=%0.1f rpm\n\n',N)
printf('\n\n Motor Torque=%0.1f N-m\n\n',T)
```

Scilab code Exa 4.40 Ex40

```
clc
```

```
//Variable Initialisation
Ea=220//Input Voltage of motor in volts
N1=700//Rated Speed of Motor in rpm
T1=247
Ra=0.06//Armature resistance in ohm
Rf=0.04
d1=0.7
T2=1.5*T1
Ia1=133//Armature Current in Ampere
//Solution
K=T2/Ia1
R=Ra+Rf
Eb=(d1*Ea)-(Ia1*R)//Wrong value taken in book for
    Armature current
W=Eb/K
N2=W*60/(2*%pi)
printf('\n\n Motor Speed=%0.1f rpm\n\n',N2)//The
    answer provided in the textbook is wrong
```

Scilab code Exa 4.41 Ex41

```
clc
//Variable Initialisation
Ea=220//Input Voltage of motor in volts
Ia=125//Armature Current in Ampere
Eb=60//Average Value of Back EMF
f=200//Chopper Frequency
Ra=0.025//Armature resistance in ohm
Rf=0.015//Field resistance in ohm
//Solution
d1=(Eb+(Ia*(Ra+Rf)))/Ea
T=(1/f)
ton=d1*T
printf('\n\n The Pulse Width=%0.1f msec\n\n',ton
    *10^3)
```

Scilab code Exa 4.42 Ex42

```
clc
//Variable Initialisation
Ea=220//Input Voltage of motor in volts
d1=0.8//Duty Ratio
Ia1=300//Armature Current in Ampere
Ra=0.04//Armature resistance in ohm
N1=600//Rated Speed of Motor in rpm
//Solution
E0=d1*Ea
Eb1=E0-(Ia1*Ra)
Eb2=210
N2=(Eb1/Eb2)*N1
Ia3=310
N3=500
Eb3=142
T=Eb3*Ia3/(2*%pi*N3/60)//Wrong calculated in book
   used N=520 instead of 500
printf('\n\n The Motor Speed=%0.1f rpm\n\n',N2)
printf('\n\n The Motor Torque=%0.1f N-m\n\n',T)//The
    answer provided in the textbook is wrong)
```

Scilab code Exa 4.43 Ex43

```
clc
//Variable Initialisation
Ea=220//Input Voltage of motor in volts
d1=0.8//Duty Ratio
Ia1=310//Armature Current in Ampere
Ra=0.04//Armature resistance in ohm
```

```
N4=1500//Rated Speed of Motor in rpm
//Solution
E0=d1*Ea
Eb1=E0+(Ia1*Ra)
N1=610//Motor speed in rpm
Eb2=215
N2=(Eb1/Eb2)*N1
d2=0.95
E02=d2*Ea
Eb3=E02+(Ia1*Ra)
N3=(Eb3/Eb2)*N1
Eb4=(N4/N1)*Eb2
Ra1=((Eb4-E02)/Ia1)
printf('\n\n Motor Speed=%0.1f rpm\n\n',N2)
printf('\n\n Maximum Allowable Speed=%0.1f rpm\n\n',
    N3)//The answers vary due to round off error
printf('\n\n Resistance to be Inserted=%0.1f ohm\n\n
    ',Ra1)
```

Scilab code Exa 4.44 Ex44

```
clc
//Variable Initialisation
Eb2=215//Average Value of Back EMF
Ia=300//Armature Current in Ampere
Ia1=310//Armature Current in Ampere
Ra=0.04//Armature resistance in ohm
N1=610//Rated Speed of Motor in rpm
N2=750//Rated Speed of Motor in rpm
dmin=0.05//Minimum Duty Ratio
//Solution
Eb1=(N2/N1)*Eb2
Rbe=(Eb1/Ia)-Ra
Rb=Rbe/(1-dmin)
R=Rb*(1-dmin)+Ra
```

```
K=Eb2/(2*%pi*N1/60)//The answer provided in the textbook is wrong
T=K*Ia
printf('\n\n Value of Braking Resistor=%0.1f ohm\n\n',Rb)
printf('\n\n Maximum Available Motor Torque=%0.1f N-m\n\n',T)
```

Scilab code Exa 4.45 Ex45

```
clc
//Variable Initialisation
Ra=0.4//Armature resistance in ohm
Rf=0.4//Field resistance in ohm
N1=400//Rated Speed of Motor in rpm
N2=500//Rated Speed of Motor in rpm
//Solution
W1=2*%pi*N1/60
W2=2*%pi*N2/60
Ia=97.5
K=15.8
Eb=K*W2
Rb=(Eb/Ia)-Ra
printf('\n\n Resistance across motor terminal=%0.1f ohm\n\n',Rb)
```

Scilab code Exa 4.46 Ex46

```
clc
//Variable Initialisation
Ea=230//Input Voltage of motor in volts
d1=0.8//Duty Ratio
d2=0.75//Duty Ratio
```

```
Ia1=80//Armature Current in Ampere
Ra=0.25//Armature resistance in ohm
N2=750//Rated Speed of Motor in rpm
N3=600//Rated Speed of Motor in rpm
Ia2=70
Eb2=210//Average Value of Back EMF
//Solution
E01=d1*Ea
Eb1=E01-(Ia1*Ra)
N1=(Eb1/Eb2)*N2
Ia2=86
E02=d2*Ea
Eb3=E02-(Ia2*Ra)
Wm=2*%pi*N3/60
T=Eb3*Ia2/Wm
printf('\n\n Motor Speed=%0.1f rpm\n\n',N1)
printf('\n\n Torque produced=%0.1f N-m\n\n',T)
```

Scilab code Exa 4.47 Ex47

```
clc
//Variable Initialisation
Ea=230//Input Voltage of motor in volts
d1=0.4//Duty Ratio
Ia1=86//Armature Current in Ampere
Ra=0.25//Armature resistance in ohm
N1=850//Rated Speed of Motor in rpm
N4=1300//Rated Speed of Motor in rpm
Eb1=220//Average Value of Back EMF
//Solution
E01=d1*Ea
Eb2=E01+(Ia1*Ra)
N2=(Eb2/Eb1)*N1
dmax=0.98//Maximum aloowable duty ratio
E02=dmax*Ea
```

```
Eb3=E02+(Ia1*Ra)
N3=(Eb3/Eb1)*N1//Wrong value of N1 is taken in
    textbook
Eb4=(N4/N1)*Eb1
R=((Eb4-E02)/Ia1)-Ra
E1=(N1/N4)*Eb3
n=E1/Eb1
printf('\n\n Motor speed=%0.1f rpm\n\n',N2)
printf('\n\n Maximum allowable motor Speed=%0.1f rpm
    \n\n',N3)//The answer provided in the textbook is
     wrong
printf('\n\n Resistance to be inserted=%0.1f ohm\n\n
    ',R)
printf('\n\n number of turns reduced to fraction=%0
    .1f\n\n',n)
```

Scilab code Exa 4.48 Ex48

```
clc
//Variable Initialisation
Ia1=86//Armature Current in Ampere
Ra=0.25//Armature resistance in ohm
N1=1000//Rated Speed of Motor in rpm
N2=850//Rated Speed of Motor in rpm
Eb1=220//Average Value of Back EMF
dmin=0.5//Minimum Duty Ratio
dmax=0.95//Maximum Duty Ratio
//Solution
Eb2=(N1/N2)*Eb1
Rbe=(Eb2/Ia1)-Ra
Rb=Rbe/(1-dmin)
R=Rb*(1-dmax)+Ra
Eb3=Ia1*R
Eb4=190
Ia2=55
```

```
K=Eb4/(2*%pi*N2/60)
T=K*Ia2
printf('\n\n Braking Resistor=%0.1f ohm\n\n',Rb)
printf('\n\n Maximum Available Motor Torque=%0.1f N-
    m\n\n',T)
```

Scilab code Exa 4.49 Ex49

```
clc
//Variable Initialisation
Ea=500//Input Voltage of motor in volts
d1=0.65//Duty Ratio
Ra=0.06//Armature resistance in ohm
Ia=300//Armature Current in Ampere
Rf=0.08//Field resistance in ohm
K=15.27e-3//Motor Constant
//Solution
E0=(1-d1)*Ea
R=Ra+Rf
Req=(1-d1)*(Ea/Ia)+R
Pgen=E0*Ia
Wmin=R/K
Wminr=Wmin*(30/%pi)
Wmax=(Ea/(K*Ia))+(R*Ia/(K*Ia))
Wmaxr=Wmax*(30/%pi)
Eb=E0+(Ia*Ra)
W=Eb/(K*Ia)
Wr=W*(30/%pi)
printf('\n\n Voltage across Chopper=%0.1f Volts\n\n'
    ,E0)
printf('\n\n Equivalent Resistance of Motor=%0.1f
    ohm\n\n',Req)
printf('\n\n Power Generated=%0.1f KW\n\n',Pgen
    *10^-3)
printf('\n\n Maximum Permissible Braking Speed=%0.1f
```

```
rpm\n\n',Wmaxr)
printf('\n\n Minimum Permissible Braking Speed=%0.1f rpm\n\n',Wminr)
printf('\n\n Motor Speed=%0.1f rpm\n\n',Wr)
```

Scilab code Exa 4.50 Ex50

```
clc
//Variable Initialisation
Ea=500//Input Voltage of motor in volts
Ra=0.06//Armature resistance in ohm
Rf=0.09//Field resistance in ohm
K=12e-3//Motor Constant
Ia=400//Armature Current in Ampere
d1=0.6//Duty Ratio
//Solution
E0=(1-d1)*Ea
Pin=E0*Ia
R=Ra+Rf
Req=(E0/Ia)+R
Wmin=R/K
Wminr=Wmin*30/%pi
Wmax=(R/K)+(Ea/(K*Ia))
Wmaxr=Wmax*30/%pi
Eb=E0+(Ia*R)
W=Eb/(K*Ia)
Wr=W*30/%pi
printf('\n\n Voltage across Converter=%0.1f Volts\n\n',E0)
printf('\n\n Power Generated=%0.1f KW\n\n',Pin*10^-3)
printf('\n\n Equivalent Resistance of Motor=%0.1f ohm\n\n',Req)
printf('\n\n Maximum Permissible Braking Speed=%0.1f rpm\n\n',Wmaxr)
```

```
printf('\n\n Minimum Permissible Braking Speed=%0.1f
    rpm\n\n',Wminr)
printf('\n\n Motor Speed=%0.1f rpm\n\n',Wr)
```

Scilab code Exa 4.51 Ex51

```
clc
//Variable Initialisation
Ia=300//Armature Current in Ampere
Rb=8//Braking resistance in ohm
Ra=0.05//Armature resistance in ohm
Rf=0.08//Field resistance in ohm
d=0.5//Duty Ratio
K=14e-3//Motor Constant
//Solution
E0=(1-d)*Ia*Rb
Pin=(Ia^2)*Rb*(1-d)
R=Ra+Rf
Req=Rb*(1-d)+R
Eb=E0+(Ia*R)
W=Eb/(K*Ia)
Wr=W*30/%pi
Ep=Ia*Rb
printf('\n\n Voltage across Converter=%0.1f Volts\n\
   n',E0)
printf('\n\n Power dissipated=%0.1f KW\n\n',Pin
   *10^-3)
printf('\n\n Equivalent Resistance of Motor=%0.1f
   ohm\n\n',Req)
printf('\n\n Motor Speed=%0.1f rpm\n\n',Wr)
printf('\n\n Peak to Peak Voltage of Converter=%0.1f
    Volts\n\n',Ep)
```

Chapter 5

Stator Voltage Control of Induction Motor Drives

Scilab code Exa 5.1 Ex1

```
clc
//Variable Initilisation
Ns=1500 //Speed of Squirrel Cage Induction Motor in RPM
N1=1460 //Speed of Squirrel Cage Induction Motor in RPM
N2=1350 //Speed of Squirrel Cage Induction Motor in RPM

// At 1460 rpm the speed slip is given by
S1=(Ns-N1)/Ns //Slip
I=(sqrt(1/3)*(2/3))/(sqrt(S1)*(1-S1))
// At 1350 rpm the speed slip is given by
S2=(Ns-N2)/Ns //Slip
I1=(sqrt(1/3)*(2/3))/(sqrt(S2)*(1-S2))


//Results
printf('\n\n The motor maximum Current in terms of
```

```
rated current at the above speed =%0.1f \n\n',I)
printf('\n\n The motor maximum Current in terms of rated current at the above speed =%0.1f \n\n',I1)
```

Scilab code Exa 5.2 Ex2

```
clc
//variable Initialisation
V=415//Voltage Input in Volts
f=50//supply frequency in Hz
P=4//No of Poles
N1=1450//Rotor Speed in rpm
N2=1290//Rotor Speed in rpm for case II
R1=1.01
R2=0.69
X1=1.08
X2=1.60
Xm=36
Tl=42//Rated torque in N-m
//Solution
Vph=V/sqrt(3)
Ns=120*f/P
Ws=2*%pi*Ns/60
Wm=2*%pi*N1/60
K=Tl/(Wm^2)
s=(Ns-N2)/Ns//Slip
Wm2=Ws*(1-s)
Tl=K*(Wm2^2)//Load Torque in N-m
Tl2=Tl*Wm2//Torque in Synchronous Watts
I2=sqrt((Tl2*s)/(3*R2*(1-s)))
Z=R1+(R2/s)+(%i*(X1+X2))//Impedance at slip s
V2=I2*abs(Z)//Voltage applied in Volts/Phase
Im=V2/(%i*Xm)
Im1=abs(Im)
Ir=V2/Z//Rotor Current
```

```
Is=Ir+Im//Stator Current
a=atand(imag(Is)/real(Is))
Pin=3*V2*abs(Is)*cosd(a)//Input Power
Smax=1/3//Smax is obtain theoretically
I2max=Ws*sqrt(Smax)*(1-Smax)*sqrt(K*Ws/(3*R2))
Nr=Ns*(1-Smax)//Speed at maximum Current
Wmax=2*%pi*Nr/60
T=3*(I2max^2)*R2*(1-Smax)/(Smax*Wmax)//Torque at maximum Current
printf('\n\n The Load torque=%0.1f N-m\n\n',Tl)
printf('\n\n The Rotor Current=%0.1f Amp\n\n',Ir)
printf('\n\n The Stator Supply Voltage=%0.1f Volts\n\n',V2)
printf('\n\n The Motor input current=%0.1f Amp\n\n',Is)
printf('\n\n The Motor input power=%0.1f Watt\n\n',Pin)//The answer provided in the textbook is wrong
printf('\n\n Maximum rotor Current=%0.1f Amp\n\n',I2max)
printf('\n\n The speed at maximum current=%0.1f rad/sec\n\n',Wmax)
printf('\n\n The torque at maximum current=%0.1f N-m\n\n',T)
```

Scilab code Exa 5.3 Ex3

```
clc
//variable Initialisation
V1=440//Voltage Input in Volts
f1=50//supply frequency in Hz
P=25e+3//power rating in Watts
N1=950//Rotor Speed in rpm
Z=0.1+(%i*3)//Rotor Impedance
pole=6//No of poles
```

```
f2=80//Supply 2 frequency in Hz
//Solution
V=Vl/sqrt(3)//Phase Voltage in Volts
Wm=2*%pi*N1/60
Tfl=P/Wm//Full load Torque in N-m
Ns=120*f2/pole
Ws=2*%pi*Ns/60
Z2=Z*(f2/f1)//Rotor Impedance at 80 Hz
S=3*(V^2)*0.5/(Ws*((abs(Z2))^2)*Tfl)
Nr=Ns*(1-S)
Rl=real(Z)
Xl=imag(Z2)
Smax=(Rl/Xl)
Tmax=3*(V^2)/(Ws*2*Xl)
printf('\n\n The Motor speed=%0.1f rpm\n\n',Nr)//The
    answers vary due to round off error
printf('\n\n The Slip at which maximum torque occurs
  =%0.1f\n\n',Smax)
printf('\n\n The maximum Torque=%0.1f\n\n',Tmax)
```

Scilab code Exa 5.4 Ex4

```
clc
//variable initialisation
Vm=400 //Input Voltage in volt
F=50 //supply frequency in Hz
P1=4 //number of poles
R1=0.15 //resistance of stator in ohm
R2=0.12 //resistance of rotor in ohm
X1=0.45 //reactance of Motor in ohm
X2=0.45 //reactance of Motor in ohm
Xm=28.5 //reactance of Motor in ohm
S=0.04 //Slip Of Motor

//Solution
```

```
Rl=R2*((1/S)-1)
Vph=Vm/sqrt(3)
I2=Vph/((R1+R2+Rl)+%i*(X1+X2))
I0=Vph/(%i*Xm)
I1=I0+I2
y=imag(I1)
x=real(I1)
phi=atand(y/x)
pf=cosd(phi)
printf('\n\n The Stator Current=%0.1f Amp\n\n',I1)
printf('\n\n The Power Factor=%0.1f lag\n\n',pf)
```

Scilab code Exa 5.5 Ex5

```
clc
//variable initialisation
Pout=37.3 //Motor Output In KW
Vm=440 //Motor Input in volt
F=50 //supply frequency in Hz
I0=20 //NO Load Line Current Of Motor
R1=0.1 //resistance of stator in ohm
R2=0.15 //resistance of rotor in ohm
X1=0.4 //reactance of Motor in ohm
X2=0.44 //reactance of Motor in ohm
S=0.03 //Slip Of Motor
Ls=1250 //Stator Core Loses In Watt
Lr=1000 //Rotational Losses In KW
Ns=1500 // Synchronous Speed Of Motor

//Solution
Vph=Vm/sqrt(3)
I2=Vph/((R1+(R2/S))+%i*(X1+X2))
I21=abs(I2)
I21=49.1//rounding off to avoid computational error
I0=1.78-(%i*19.9)//Taken in book for No load motor
```

```
   current
I1=I0+I2
y=imag(I1)
x=real(I1)
phi=atand(y/x)
pf=cosd(phi)
P2=3*((I21)^2)*(R2/S)
Tg=(9.55*P2)/Ns
Pm=(1-S)*P2
Pout1=Pm+Lr
Lcs=3*((I21)^2)*R1//Wrong value of I2 is taken in
   textbook
Lcr=S*P2
Lt=Ls+Lr+Lcs+Lcr
Pin=Lt+Pout1
n=Pout1/Pin
printf('\n\n The input line Current=%0.1f Amp\n\n',
   I1)
printf('\n\n The power factor=%0.1f lag\n\n',pf)
printf('\n\n The Electromagnetic Torque Developed=%0
   .1f N-m\n\n',Tg)
printf('\n\n The output=%0.1f Watts\n\n',Pm)
printf('\n\n The efficiency of Motor=%0.1f\n\n',n)
//The answers vary due to round off error
```

Scilab code Exa 5.6 Ex6

```
clc
//variable initialisation
Vm=400 //Supply Voltage in volt
F=50 //supply frequency in hrtz
P=6 //Number Of Poles
R1=0.15 //resistance of stator in ohm
R2=0.15 //resistance of rotor in ohm
X1=0.8 //reactance of Motor in ohm
```

```
X2=0.8 //reactance of Motor in ohm
S=0.04 //Slip Of Motor

//Solution
Ns=(120*F)/P
Ws=((2*%pi)/60)*1000
Sr=2-S
Vph=Vm/(sqrt(3))
I2=Vph/(sqrt(((R1+(R2/((2-S))))^2)+((X1+X2)^2)))
Tsb=(3*((I2)^2)*(R2/(2-S)))/(Ws)
printf('\n\n The Initial Braking Torque=%0.1f N-m\n\n',Tsb)
```

Scilab code Exa 5.7 Ex7

```
clc
//variable initialisation
Pout=7.5 // Output Of Motor In KW
Vm=230 //Supply Voltage in volt
F=50 //supply frequency in hrtz
R1=0.36 //resistance of stator in ohm
R2=0.222 //resistance of rotor in ohm
X1=0.47 //reactance of Motor in ohm
X2=0.47 //reactance of Motor in
Xm=15.5 //reactance of Motor in ohm
S=0.4723 //Slip Of Motor
P=4 //Number Of Poles

//Solution
Vph=Vm/sqrt(3)
Z=((R1+(R2/S))+(%i*(X1+X2)))
I2=Vph/Z
I2r=abs(I2)
Lcr=3*((I2r)^2)*R2
P2=Lcr/S
```

```
Ns=(120*F)/P
Tst=(9.55*P2)/Ns
Sm=R2/X2
Z1=(R1+R2)+%i*(X1+X2)
Z2=abs(Z1)
I3=Vph/Z2
I4=abs(I3)
P3=3*((I4)^2)*R2
Tst1=(9.55*P3)/(Ns)
printf('\n\n The Maximum Internal Torque=%0.1f N-m\n\n',Tst)
printf('\n\n Slip at Maximum Torque=%0.1f\n\n',Sm)
printf('\n\n The Starting Torque=%0.1f N-m\n\n',Tst1)
```

Scilab code Exa 5.8 Ex8

```
clc
//variable initialisation
Vm=400 //input of motor in volt
F=50 //supply frequency in hrtz
P=4 //Number Of Poles
R1=1 //resistance of stator in ohm
R2=0.4 //resistance of rotor in ohm
X1=1 //reactance of Motor in ohm
X2=1 //reactance of Motor in ohm
Xm=50 //reactance of Motor in ohm
Vc=231 //Constant Voltage Source In Volt
I1=28//Current from Constant Current Source in Amp
//Solution
Xe=(X1*Xm)/(X2+Xm)
Sm=R2/(Xe+X2)
Sm1=R2/(X2+Xm)
Ve=(Vc*Xm)/(X1+Xm)
Ws=(4*%pi*F)/(P)
```

```
Test=(3/Ws)*(((Ve)^2)/(R2^2+(X2+Xe)^2))*R2
Tem=(3/Ws)*(((Ve)^2)/(2*(X2+Xe)))
Test1=(3/Ws)*(((I1*Xm)^2)/(R2^2+(X2+Xm)^2))*R2
Tem1=(3/Ws)*(((I1*Xm)^2)/(2*(X2+Xm)))
Im=I1*((R2/Sm1)+(%i*X2))/((R2/Sm1)+%i*(X2+Xm))
V1=sqrt(3)*abs(Im)*Xm
printf('\n\n The Slip for maximum torque for Voltage source=%0.1f\n\n',Sm)
printf('\n\n The Slip for maximum torque for current source=%0.1f\n\n',Sm1)
printf('\n\n The Starting Torque for Voltage source=%0.1f N-m\n\n',Test)
printf('\n\n The Maximum Torque for Voltage Source=%0.1f N-m\n\n',Tem)
printf('\n\n The Starting Torque for Current Source=%0.1f N-m\n\n',Test1)
printf('\n\n The Maximum Torque for Current Source=%0.1f N-m\n\n',Tem1)
printf('\n\n The Supply voltage required=%0.1f Volt\n\n',V1)
//The answers vary due to round off error
```

Scilab code Exa 5.9 Ex9

```
clc
//variable initialisation
Vph=2200 //Supply Voltage in volt
F=50 //supply frequency in Hz
Pout=2600 // Output Of Motor In KW
P=8 //Number Of Poles
N1=735 //Speed OF Motor In rpm
Rs=0.075 //Resistance of stator in ohm
Rr=0.1 //Resistance of rotor in ohm
Xs=0.45 //Reactance of Motor in ohm
Xr=0.55 //Reactance of Motor in ohm
```

```
//Solution
Ns=(120*F)/P
S=(Ns-N1)/Ns
Iph=Vph/sqrt((Rs+((Rr/S)^2))+((Xs+Xr)^2))
Il=sqrt(3)*Iph
Wms=(2*%pi*Ns)/60
Tl=(3*((Iph)^2))/(S*(Wms))
Ilm=Vph/(sqrt(3)*sqrt(((Rs+Rr)^2)+((Xs+Xr)^2)))
S2=1
Tst=(3*(((Ilm)^2)*0.1))/(S2*Wms)
r1=Tst/Tl//ratio of Tst and Tl
Tmax=(3/(2*Wms))*(((Vph/sqrt(3))^2)/((Rs+sqrt((Rs^2)+((Xs+Xr)^2)))))
r2=Tmax/Tl//ratio of Tmax and Tl
Rr2=0.15
Xr2=0.9
Il1=(sqrt(3)*Vph)/(sqrt(((Rs+Rr)^2)+((Xs+Xr2)^2)))
Iph1=Il1/(sqrt(3))
Tst1=(3*((Iph1)^2)*(Rr))/(Wms)
Rs1=Rs/3
Rr1=Rr/3
Xs1=Xs/3
Xr1=Xr/3
Inew=2*Il
X=sqrt(((Vph/(sqrt(3)*Inew))^2)-((Rs1+Rr1)^2))
Xe=X-Xs1-Xr1
printf('\n\n Ratio of starting torque and load torque=%0.1f \n\n',r1)//The answer provided in the textbook is wrong
printf('\n\n Ratio of maximum torque and load torque=%0.1f \n\n',r2)
printf('\n\n The Maximum line current during starting=%0.1f Amp\n\n',Il1)//The answer provided in the textbook is wrong
printf('\n\n The maximum torque at starting=%0.1f N-m\n\n',Tst1)//The answer provided in the textbook is wrong
```

```
printf('\n\n The required value of reactor=%0.1f ohm
   \n\n',Xe)
```

Scilab code Exa 5.10 Ex10

```
clc
//variable initialisation
Vm=400 //Rated Voltage of motor in volt
Vs=440 //Supply Voltage of motor in volt
F=50 //Supply frequency in hrtz
P=4 //Number Of Poles
N1=1475 //Speed OF Motor In rpm
R1=0.35 //Resistance of stator in ohm
R2=0.18 //Resistance of rotor in ohm
X1=0.9 //Reactance of Motor in ohm
X2=0.7 //Reactance of Motor in ohm
Xm=25 //Reactance of Motor in ohm

//Solution
Vph1=Vs/(sqrt(3))
Vph2=Vm/(sqrt(3))
Ns=(120*F)/(P)
S=(Ns-N1)/Ns
I2=(Vph2)/sqrt(((R1+(R2/S))^2)+((X1+X2)^2))
Pg=3*(I2^2)*(R2/S)
Sm=R2/sqrt((R1)^2+((X1+X2)^2))
Wms=(2*%pi*Ns)/60
Tm=3*(Vph1^2)/((2*Wms)*(R1+sqrt((R1)^2+((X1+X2)^2)))
   )
Zi=%i*(Xm*((R1+(R2/S))+%i*(X1+X2)))/(R1+(R2/S)+%i*(
   X1+X2+Xm))
Z=abs(Zi)
printf('\n\n The Slip=%0.1f\n\n',S)
printf('\n\n The Air gap Power Angle=%0.1f Watts\n\n
   ',Pg)
```

```
printf('\n\n The Slip for maximum torque=%0.1f\n\n',
   Sm)
printf('\n\n The Maximum Torque=%0.1f N-m\n\n',Tm)
printf('\n\n The Input Impedance=%0.1f\n\n',Z)
//The answers vary due to round off error
```

Scilab code Exa 5.11 Ex11

```
clc
//variable Initialisation
Vm=240 //Terminal Voltage Of Motor In Volt
F=50 //Supply Frequency Of Motor
P=4 //Number Of Pole
R1=0.25 //Resistance Of Motor in Ohm
R2=0.60 //Resistance Of Motor in Ohm
X1=0.36//Reactance in Ohm
X2=0.36//Reactance in Ohm
Xm=17.3//Reactance In Ohm
Nr1=1400 //Speed Of Rotor In RPM
Nr2=600 //Speed Of Rotor In RPM

//Solution
#Case=1
W=((2*%pi)/60)*(Nr1)
Ns=(120*F)/(P)
S1=(Ns-Nr1)/Ns
S2=(Ns-Nr2)/Ns
Zr=(R2/S1)+%i*(X2)
Zs=R1+%i*(X1)
Zt=Zr+Zs
Zin=(%i*(Xm)*(Zt))/(%i*(Xm)+(Zt))
Tl=1.4*((10)^-3)*(W)^2
n=Nr1/60
I2=sqrt((S1*Tl*2*%pi*n)/(3*R2*(1-S1)))
#Case=2
```

```
Zr1=(R2/S2)+%i*(X2)
Zs1=R1+%i*(X1)
Zt1=Zr1+Zs1
Zin1=(%i*(Xm)*(Zt1))/(%i*(Xm)+(Zt1))
W1=((2*%pi)/60)*(Nr2)
Tl1=1.4*((10)^-3)*(W1)^2
n1=Nr2/60
I3=sqrt((S2*Tl1*2*%pi*n1)/(3*R2*(1-S2)))
//Given base currents in Amp
Ib1=17.55
Ib2=100.27
Ip1=I2/Ib1
Ip2=I3/Ib2
printf('\n\n The per unit rotor Current for case 1=
    %0.1f\n\n',Ip1)
printf('\n\n The per unit rotor Current for case 2=
    %0.1f\n\n',Ip2)
```

Scilab code Exa 5.12 Ex12

```
clc
//variable Initialisation
Vm=400 //Terminal Voltage Of Motor In Volt
F=50 //Supply Frequency Of Motor
P=6 //Number Of Pole
R1=0.2 //Resistance Of Motor in Ohm
R2=0.2 //Resistance Of Motor in Ohm
X1=0.5 //Reactance in Ohm
X2=0.5 //Reactance in Ohm
Xm=15 //Reactance In Ohm
S=0.05 //Slip Of Motor

//Solution
Ns=(120*F)/(P)
Ws=((2*%pi)/60)*(Ns)
```

```
Vph=Vm/sqrt(3)
S1=2-S
I2=(Vph)/sqrt(((R1+(R2/S1))^2)+((X1+X2)^2))
Im=Vph/Xm
I1=Im+I2
Tb=((3*((I2)^2))/(Ws))*(R2/S1)
printf('\n\n The Primary Current=%0.1f Amp\n\n',I1)
printf('\n\n The Braking Torque=%0.1f N-m\n\n',Tb)
```

Scilab code Exa 5.13 Ex13

```
clc
//variable Initialisation
Vm=400 //Terminal Voltage Of Motor In Volt
F=50 //Supply Frequency
P=6 //Number Of Pole
R1=1.5 //Resistance Of Motor in Ohm
R2=1.5 //Resistance Of Motor in Ohm
X1=2.5//Reactance in Ohm
X2=2.5//Reactance in Ohm
Nr1=900 //Speed Of Rotor In RPM
Nr2=400 //Speed Of Rotor In RPM

//Solution
Vph=Vm/sqrt(3)
Ns=(120*F)/(P)
S=(Ns-Nr1)/Ns
I2=(Vph)/sqrt(((R1+(R2/S))^2)+((X1+X2)^2))
Ws=((2*%pi)/60)*(Ns)
T=((3*((I2)^2))/(Ws))*(R2/S)
//At Braking
Sb=2-S
I2b=(Vph)/sqrt(((R1+(R2/Sb))^2)+((X1+X2)^2))
Tb=((3*((I2b)^2))/(Ws))*(R2/Sb)
S1=(Ns+Nr2)/Ns
```

```
I3=(Vph)/sqrt(((R1+(R2/S1))^2)+((X1+X2)^2))
T1=((3*((I3)^2))/(Ws))*(R2/S1)
printf('\n\n The Full load Torque=%0.1f N-m\n\n',T)
printf('\n\n The Initial braking Torque=%0.1f N-m\n\
  n',Tb)
printf('\n\n The braking Torque at 400 rpm=%0.1f N-m
  \n\n',T1)
```

Scilab code Exa 5.14 Ex14

```

clc
//variable Initialisation
Vm=400 //Terminal Voltage Of Motor In Volts
Pout=3 //Output Of Motor In KW
F=50 //Supply Frequency
P=4 //Number Of Pole
R1=2.5 //Resistance Of Motor in Ohm
R2=4.5 //Resistance Of Motor in Ohm
X1=6 //Reactance in Ohm
X2=6 //Reactance in Ohm
Nr1=1400 //Speed Of Rotor In RPM
Nr2=1300 //Speed Of Rotor In RPM

//Solution
Ns=(120*F)/(P)
S=(Ns-Nr1)/Ns
Vph=Vm/sqrt(3)
I2=(Vm)/sqrt(((R1+(R2/S))^2)+((X1+X2)^2))
Ws=((2*%pi)/60)*(Ns)
Tl=((3*((I2)^2))/(Ws))*(R2/S)
K=Tl/((1-S)^2)
S1=(Ns-Nr2)/Ns
Tl1=K*((1-S1)^2)
Vs=sqrt(Tl1*S1*Ws*(((R1+(R2/S1))^2)+((X1+X2)^2))
```

```
   /((3)*(R2)))//Wrongly calculated in textbook
printf('\n\n The Voltage To be Applied=%0.1f Volts\n
   \n',Vs)
//The answer provided in the textbook is wrong
```

Scilab code Exa 5.15 Ex15

```
clc
//variable Initialisation
Vm=400 //Terminal Voltage Of Motor In Volt
F=50 //Supply Frequency
P=6 //Number Of Pole
R1=0.6 //Resistnce Of Motor in Ohm
R2=0.5 //Resistnce Of Motor in Ohm
X1=1.3 //Reactance in Ohm
X2=1.3 //Reactance in Ohm
Xm=50 //Reactance In Ohm
Nr=950 //Speed Of Rotor In RPM

//Solution
Ns=(120*F)/(P)
Wms=((2*%pi)/60)*(Ns)
S=(Ns-Nr)/Ns
Vph=Vm/sqrt(3)
I2=(Vph)/sqrt(((R1+(R2/S))^2)+((X1+X2)^2))
Im=Vph/Xm
I1=Im+I2
Z1=(%i*(Xm)*((R2/S)+%i*(X2)))/((R2/S)+(%i*(X2+Xm)))
Zf=R1+%i*(X1)+(Z1)
Z2=%i*Xm*((R2/(2-S))+(%i*(X2)))/((R2/(2-S))+(%i*(X2+
   Xm)))
Zb=R1+%i*(X1)+(Z2)
Z3=Zf+Zb
Znew=abs(Z3)
I=Vph/Znew
```

```
Tp=(3*((I)^2)*((Xm)^2)*(R2/S))/((Wms)*(((R2/S)^2)+((
    X2+Xm)^2)))
Tn=-((3*((I)^2)*((Xm)^2)*(R2/(2-S)))/((Wms)*(((R2
    /(2-S))^2)+((X2+Xm)^2))))
T=Tp+Tn
Wm=Wms*(1-S)
Tl=(8.4/1000)*(Wm^2)//Given
printf('\n\n The Motor Speed=%0.1f rpm\n\n',Ns)
printf('\n\n The motor Current=%0.1f Amp\n\n',I)
disp("Since T=Tl,Motor will run Safely")
```

Chapter 6

Control of Induction Motor Drives through Stator Frequency

Scilab code Exa 6.1 Ex1

```
clc
//variable initialization
Vm=415 //input of motor in volt
F1=50 //supply frequency in hrtz
F2=35 //supply frequency in hrtz
F3=10 //supply frequency in hrtz
N=1460 //speed of motor in rpm
P=4 //number of poles
R1=0.65 //resistance of stator in ohm
R2=0.35 //resistance of rotor in ohm
X1=0.95 //reactance of Motor in ohm
X2=1.43 //reactance of Motor in ohm
Xm=28 //reactance of Motor in ohm



//Solution
```

```
V1ph=Vm/sqrt(3)
Ns1=(120*F1)/P
Wsm1=(2*%pi/60)*Ns1
Sm1=R2/sqrt((R1^2)+(X1+X2)^2)//Slip for maximum torque
Nr1=Ns1*(1-Sm1)
Tm1=3*((V1ph)^2)/(2*Wsm1*(R1+sqrt((R1)^2+(X1+X2)^2)))

V2ph=Vm/sqrt(3)
X3=X1*(F2/F1)
X4=X2*(F2/F1)
Sm2=R2/sqrt((R1^2)+(X3+X4)^2)//Slip for maximum torque
Ns2=(120*F2)/P
Wsm2=(2*%pi/60)*Ns2
Nr2=Ns2*(1-Sm2)
Tm2=3*((V2ph*F2/F1)^2)/(2*Wsm2*(R1+sqrt((R1)^2+(X3+X4)^2)))

V3ph=Vm/sqrt(3)
X5=X1*(F3/F1)
X6=X2*(F3/F1)
Sm3=R2/(sqrt((R1^2)+((X5+X6)^2)))//Slip for maximum torque
Ns3=(120*F3)/P
Wsm3=(2*%pi/60)*Ns3
Nr3=Ns3*(1-Sm3)
Tm3=3*((V3ph*F3/F1)^2)/(2*Wsm3*(R1+sqrt((R1)^2+(X5+X6)^2)))
printf('\n\n speed at which maximum torque occurs for 50 Hz=%0.1f rpm\n\n',Nr1)
printf('\n\n maximum torque for 50 Hz=%0.1f N-m\n\n',Tm1)
printf('\n\n speed at which maximum torque occurs for 35 Hz=%0.1f rpm\n\n',Nr2)
printf('\n\n maximum torque for 35 Hz=%0.1f N-m\n\n',Tm2)
```

```
printf('\n\n speed at which maximum torque occurs
    for 10 Hz=%0.1f rpm\n\n',Nr3)
printf('\n\n maximum torque for 10 Hz=%0.1f N-m\n\n'
    ,Tm3)
```

Scilab code Exa 6.2 Ex2

```
clc
//variable initialization
Pout=50 //output of induction motor in kilowatt
Vm=400 //input of motor in volt
F0=50 //supply frequency in hrtz
N1=1470 //speed of motor in rpm
P=4 //number of pole
Rs=0.42 //resistance of stator in ohm
Rr=0.23 //resistance of rotor in ohm
Xs=0.95 //reactance of stator in ohm
Xr=0.85 //reactance of rotor in ohm
Xm=28 //reactance of motor in ohm
Sm=0.12 //slip of motor
//Solution
Vs=Vm/sqrt(3)
W0=2*%pi*F0
K=Rr/(Sm*(Xs+Xr))
F=K*F0//Supply Frequency
Tdm=3*P*Vs^2/(2*(K^2)*W0(Xs+Xr))
Ws=(K*W0*2)/(P)
Wm=Ws*(1-Sm)
N2=Wm*60/(2*%pi)
printf('\n\n Supply Frequency=%0.1f Hz\n\n',F)
printf('\n\n The Breakdown Torque=%0.1f N-m\n\n',Tdm
    )
printf('\n\n The Speed at maximum torque=%0.1f rpm\n
    \n',N2)
```

Scilab code Exa 6.3 Ex3

```
clc
//variable initialization
Pout=50 //output of induction motor in kilowatt
Vm=400 //input of motor in volt
F0=50 //supply frequency in hrtz
N1=1475 //speed of motor in rpm
P=4 //number of poles
Rs=0.42 //resistance of stator in ohm
Rr=0.23 //resistance of rotor in ohm
Xs=0.95 //reactance of stator in ohm
Xr=0.85 //reactance of rotor in ohm
Xm=30 //reactance of motor in ohm
Tdm=225 //Breakdown Torque In N-m
K=poly(0,'K')




//Solution
W0=2*%pi*F0
Vp=Vm/sqrt(3)
K=sqrt((3*2*(Vp^2))/(2*Tdm*W0*(Xs+Xr)))
W1=K*W0
F1=W1/(2*%pi)
Sm=Rr/(K*(Xs+Xr))
Ws=2*K*W0/(P)
Wm=Ws*(1-Sm)
N=Wm*60/(2*%pi)
printf('\n\n The Supply Frequency=%0.1f Hz\n\n',F1)
printf('\n\n The slip at maximum torque=%0.1f\n\n',
    Sm)
printf('\n\n The speed at maximum torque=%0.1f rpm\n
```

```
    \n',N)
```

Scilab code Exa 6.4 Ex4

```
clc
//variable initialization
Pout=50 //output of induction motor in kilowatt
Vm=420 //input of motor in volt
F0=50 //supply frequency in hrtz
F1=58 // frequency in hrtz
N1=1475 //speed of motor in rpm
P=4 //number of poles
Rs=0.4 //resistance of stator in ohm
Rr=0.21 //resistance of rotor in ohm
Xs=0.95 //reactance of stator in ohm
Xr=0.85 //reactance of rotor in ohm
Xm=32 //reactance of motor in ohm

//Solution
Vp=Vm/sqrt(3)
K=F1/F0
W0=2*%pi*F0
W=W0*K
Sm=Rr/(K*(Xs+Xr))
Ws=2*K*W0/P
Wm=Ws*(1-Sm)
N=Wm*60/(2*%pi)
Tdm1=(3*2*(Vp^2))/(2*(K^2)*W0*(Xs+Xr))
printf('\n\n The Slip at maximum torque=%0.1f\n\n',
    Sm)
printf('\n\n The Speed at maximum torque=%0.1f rpm\n
    \n',N)//The answers vary due to round off error
printf('\n\n The Breakdown torque=%0.1f N-m\n\n',
    Tdm1)
```

Scilab code Exa 6.5 Ex5

```
clc
//variable initialization
Pout=30 //output of induction motor in kilowatt
Vm=400 //input of motor in volt
F0=50 //supply frequency in hrtz
F1=40 // frequency in hrtz
P=4 //number of poles
Rs=0.33 //resistance of stator in ohm
Rr=0.22 //resistance of rotor in ohm
Xs=0.9 //reactance of stator in ohm
Xr=0.9 //reactance of rotor in ohm

//Solution
Vs=Vm/sqrt(3)
Sm=Rr/(sqrt((Rs^2)+((Xs+Xr)^2)))
Ir=Vs/sqrt(((Rs+(Rr/Sm))^2)+((Xs+Xr)^2))
cos_P=cosd(atand((Xs+Xr)/(Rs+(Rr/Sm))))
Pi=sqrt(3)*Vm*Ir*cos_P
P0=3*(Ir^2)*Rr*((1/Sm)-1)
n=(P0/Pi)*100

K=F1/F0//for frequency of 40 Hz
Xs2=K*Xs
Xr2=K*Xr
Sm2=Rr/(sqrt((Rs^2)+((Xs2+Xr2)^2)))
Vs2=K*Vs
Ir2=Vs2/sqrt(((Rs+(Rr/Sm2))^2)+((Xs2+Xr2)^2))
cos_p2=cosd(atand((Xs2+Xr2)/(Rs+(Rr/Sm2))))
Pi2=3*Vs2*Ir2*cos_p2
P02=3*(Ir2^2)*Rr*((1/Sm2)-1)
n2=(P02/Pi2)*100
printf('\n\n The Efficiency at breakdown torque with
```

```
    50Hz=%0.1f\n\n',n)
printf('\n\n The Efficiency at breakdown torque with
    40Hz=%0.1f\n\n',n2)
//The answers vary due to round off error
```

Scilab code Exa 6.6 Ex6

```
clc
//variable initialization
Vm=400 //input of motor in volt
F=50 //supply frequency in hrtz
N=1500 //speed of motor in rpm
P=6 //number of poles
R1=2 //resistance of stator in ohm
R2=3 //resistance of rotor in ohm
X1=4 //reactance of Motor in ohm
X2=4 //reactance of Motor in ohm
S=1 //Slip Of Motor

//Solution
Ns=(120*F)/P
Ws=(2*%pi/60)*Ns
Vph=Vm/sqrt(3)
Tst=(3/Ws)*((Vph^2)/((R1+(R2/S))^2+(X1+X2)^2))*R2
Ist=Vph/sqrt((R1+R2)^2+(X1+X2)^2)
printf('\n\n The Starting Torque=%0.1f N-m\n\n',Tst)
printf('\n\n The starting Current=%0.1f Amp\n\n',Ist
    )
```

Chapter 7

Control of Induction Motor Drives from Rotor Side

Scilab code Exa 7.1 Ex1

```
clc
//variable initialization
V=440 //voltage in volts
P=6 //number of poles
f=50 //frequency in Hz
R=0.3 //rotor resistance in ohm
X=1 //leakage reactance in ohm
s=0.03 //slip
N=800 //speed in rpm
K=2.2 //stator to rotor ratio

//solution
Ns=(120*f)/6
w=(2*%pi/60)*Ns
Tf=(3/w)*((V^(2))*(R/s))*(1/(((R/s)^(2))+(X^(2))))
k=Tf/((Ns*(1-s))^(2))
Tl=k*(N^(2))
s1=((Ns-N)/Ns)
Re=(X^(2))*(s*Tl*w)*(1/(3*(V^(2))-(Tl*w)))
```

```
X1=14.78
X2=0.07
Ree=(X1*0.2)-0.3
Ree1=(X2*0.2)-0.3
//since negative value of resistance is not feasible
Ree=2.656
//Rotor -referred value of external resistance
Rex=Ree/K^(2)
printf('\n\n The Resistance to be inserted in rotor
    circuit=%0.1f ohm\n\n',Rex)
```

Scilab code Exa 7.2 Ex2

```
clc
//variable initialisation
V=440 //Supply voltage in volts
P=4 //number of poles
f=50 //Supply frequency in Hz
R=0.2 //rotor resistance in ohm
X=0.35 //leakage reactance in ohm
N1=1450 //speed in rpm
N2=1200 //speed in rpm
S2=0.2
//solution
Vph=V/sqrt(3)
Ns=(120*f)/P//Synchronous Speed
Wms=2*%pi*Ns/60
S=(Ns-N1)/Ns
T=(3/Wms)*(Vph^2)*(R/S)/((R/S)^2+(X)^2)//The answer
    provided in the textbook is wrong
K=T/(1-S)
T2=K*(1-S2)
Vph2=sqrt(T2*((R/S)^2+(X)^2)/((3/Wms)*(R/S)))
Vl=Vph2*sqrt(3)
printf('\n\n Torque=%0.1f N-m\n\n',T)//The answer
```

```
    provided in the textbook is wrong
printf('\n\n Line Voltage to be imposed=%0.1f Volts\
    n\n',Vph2)
```

Scilab code Exa 7.3 Ex3

```
clc
//variable initialization
V=440 //voltage in volts
P=6 //number of poles
f=50 //frequency in Hz
R1=2 //resistance in ohm
R2=2 //resistance in ohm
X1=3 //reactance in ohm
X2=4 //reactance in ohm
N1=945 //speed in rpm
N2=800 //speed in rpm

//solution
Ns=(120*f)/P
s=(Ns-N1)/Ns
w=2*%pi*Ns/60
T=(3/w)*((V^(2)*(R2/s))/(((R1+(R2/s))^(2))+(X1+X2)
    ^(2)))
k=T/(1-s)^2//The answer provided in the textbook is
    wrong
s1=(Ns-N2)/Ns
Tl=k*((1-s1)^(2))//The answer provided in the
    textbook is wrong
V2=sqrt((Tl*w*(((R1+(R2/s1))^2)+((X1+X2)^2))/((R2/s1
    )*3)))//The answer provided in the textbook is
    wrong
I=V2/((R1+(R2/s1))+(%i*(X1+X2)))//The answer
    provided in the textbook is wrong
Il=sqrt(3)*I//The answer provided in the textbook is
```

```
    wrong
printf('\n\n The Line Current=%0.1f Amp\n\n',I1)
```

Scilab code Exa 7.4 Ex4

```
clc
//variable initialisation
V=400 //Supply voltage in volts
P=24 //number of poles
f=50 //frequency in Hz
P1=100e+3 //power in Watt
P2=100e+3 //power in Watt
K=1.4 //Turns ratio
R=0.03 //resistance in ohmm
N1=240 //speed in rpm
N2=180 //speed in rpm

//solution
Vp=V/sqrt(3)
Ns=(120*f)/P
s=(Ns-N1)/Ns
w=(2*%pi*N1)/60
T=P1/w
R1=K^(2)*R
X=sqrt((3*Vp^2*R1/(T*2*%pi*Ns*s/60))-((R1/s)^(2)))
s1=(Ns-N2)/Ns
T1=T*(N2/N1)^2
A=(T*2*%pi*Ns*s1/60)/(3*(Vp^2))
R22=poly(0,'R22')
0==(R22^2)*(A/(s1^2))+(A*X^2)-R22//Polynomial
    equation obtained for R22
R22=0.745//After solving equation value of Resistor
R2=(R22-R1)/K^2
printf('\n\n The Resistance to connect in series=%0
    .1f ohm\n\n',R2)
```

Scilab code Exa 7.5 Ex5

```
clc
V= 100 //supply voltage in volts
f=50 //frequency in Hz
p=6 //number of poles
Rs=0.6 //parameters in ohm
Rr=0.45 //parameters in ohm
Xr=1.2 //parameters in ohm
Xs=1.2 //parameters in ohm
Xm=45 //parameters in ohm
Sm=1
R=0.4495

//solution
Re=(((Rs^2)+(Xs+Xr)^2)*Sm-Rr) //external resistance
    in ohm
Ns=1000
N=poly(0,'N')
a=1-(((((Rs^2)+(Xs+Xr)^2)*((Ns-N)/Ns)-Rr))/(4.5*R))
printf('\n\n The Ratio of External Resistance=%0.1f\
    n\n',Re)
disp(a,'Duty Ratio alpha is ')
//The answer provided in the textbook is wrong
```

Scilab code Exa 7.6 Ex6

```
clc
//variable initialization
p=4 //number of poles
f=50 //frequency in Hz
```

```
T1=40 // Torque in N-m
s=0.04 //Average slip
T=35 //Torque in N-m
N0=1250

//solution
Tav=35 //average torque in N-m
Ns=1500 //synchronous speed in rpm
N1=(1-s)*Ns
N2=sqrt(((Tav/T1)*(N1)^2))
T2=T1*(N0^2)/(N1^2)
Tratio=((Tav-T2)/(T1-Tav))
printf('\n\n The Average Torque=%0.1f N-m\n\n',Tav)
printf('\n\n The Speed=%0.1f rpm\n\n',N2)
printf('\n\n The required ratio of torque=%0.1f\n\n'
    ,Tratio)
```

Scilab code Exa 7.7 Ex7

```
clc
I2=poly(0,'I2')//Defining I2
R2=poly(0,'R2')//Defining R2
R=poly(0,'R')//Defining R
ra=(R2-0.3*R2)/0.3//Equation drawn by neglecting
    stator impedance
Id=I2*sqrt(3/2)//From Copper Losses
R=2*ra

disp(R,'value of resistance = ')
```

Scilab code Exa 7.8 Ex8

```
clc
```

```
// variable initiallitation
T1=50 //torque in N-m
s=0.3 //slip
p=4 //number of poles
f=50 //frequency in Hz
V=400 //supply voltage in volts
Toff=poly(0,'Toff')
Ton=0.4*Toff
//solution
Tratio=0.4
Ns=1500 //synchronous speed in rpm
N1=Ns*(1-s)
T2=40 //torque in N-m
N2=sqrt((T2/T1)*(Ns)^2)
Tav=((T1*Ton+T2*Toff)/(Ton+Toff))
disp(Tav,'The Average Torque Developed')
Tav=60/1.4
printf('\n\n The Average Torque Developed=%0.1f N-m\
    n\n',Tav)
```

Scilab code Exa 7.9 Ex9

```
clc
//Variable Initialisation
V=400//Supply Voltage in Volts
f=50//Supply Frequency in Hz
P=4//No of Poles
N=1460//Rotor Speed in rpm
d1=0.2//Duty Ratio
s1=0.13//Given Slip
d2=0.6//Duty Ratio
s2=0.04//Given Slip
s3=0.0867//Slip of motor
Rs=0.08//Motor Resistance in ohm
Xs=0.95//Motor Reactance in ohm
```

```
Rr1=0.055//Motor Resistance in ohm
X21=0.5//Motor Reactance in ohm
Rd=0.0114//Resistance of link Inductor in ohm
K=2//Stator to Rotor Turns Ratio
//Solution
V1=V/sqrt(3)
Ns=120*f/P
Ws=2*%pi*Ns/60
Sm=Rr1/(sqrt((Rs^2)+((Xs+X21)^2)))//Slip at maximum
    Torque
X2=X21*(K^2)
R2=Rs*(K^2)
Rr=Rr1*(K)//Wrongly Calculated in Textbook
Radd=R2-Rr
Rw=(Radd-Rd)/(1-d1)//The answers vary due to round
    off error
Radd2=Rd+Rw*(1-d2)
R22=Radd2+Rr
Td=3*(V1^2)*R22/(s2*Ws*(((Rs+(R22/s2))^2)+((Xs+X2)
    ^2)))
printf('\n\n External Resistance=%0.1f ohm\n\n',Rw)
printf('\n\n Torque at given condition=%0.1f N-m\n\n
    ',Td)
//The answers vary due to round off error
```

Scilab code Exa 7.10 Ex10

```
clc
//Variable initialisation
V=400//Supply voltage in volts
f=50//Supply Frequency in Hz
P=6//No of poles
Rs=0.2//stator resistance in ohm
Rr=0.07//Rotor resistance in ohm
Xs=0.4//Stator impedance in ohm
```

```
Xr=0.4//Rotor impedance in ohm
Sm1=0.25//Maximum Slip at 25% speed range
N1=750//Speed in rpm
a1=130
am=150//maximum Firing Angle
n=2//Stator to rotor turns ratio
Rd=0.02//Dc link resistance in ohm
N2=950//Speed in rpm
N3=850//Speed in rpm
//Solution
V1=V/sqrt(3)
Ns=120*f/P//Synchronous speed in rpm
Wms=Ns*2*%pi/60
a=-Sm1/cosd(am)//At 25% speed Range
m=2/a//Transformer Turns Ratio
S1=(Ns-N1)/Ns
Vd11=3*sqrt(6)*S1*V1/(%pi*n)
Vd21=3*sqrt(6)*V1*cosd(a1)/(%pi*m)
Rs1=Rs/(n^2)
Rr1=Rr/(n^2)
Id1=(Vd11+Vd21)/(2*((S1*Rs1)+Rr1)+Rd)//The answers vary due to round off error
T1=abs(Vd21)*Id1/(S1*Wms)//The answers vary due to round off error
S2=(Ns-N2)/Ns
Tr=(3/Wms)*V1^2*(Rr/S2)/((Rs+(Rr/S2))^2+(Xs+Xr)^2)//Rated torque in N-m
Thr=Tr/2//Half rated Torque in N-m
S3=(Ns-N3)/Ns
X=poly(0,'X')//let X=cos(a2)
Vd12=3*sqrt(6)*S3*V1/(%pi*n)
Vd22=3*sqrt(6)*V1*X/(%pi*m)
Id2=(Vd12+Vd22)/(2*((S3*Rs1)+Rr1)+Rd)
T2=abs(Vd22)*Id2/(S3*Wms)
//Equating T2 to Thr
0==5547.31*X^2-2878.788*X+349.52//Polynomial Equation in X
X1=(2878.788+sqrt((2878.788^2)-4*5547.31*349.52))
```

```
/(2*5547.31)//Roots of polynomial eqn
X2=(2878.788-sqrt((2878.788^2)-4*5547.31*349.52))/(2*5547.31)//Roots of polynomial eqn
a11=acosd(-X1)
a22=acosd(-X2)
printf('\n\n Transformer Turns Ratio=%0.1f \n\n',m)
printf('\n\n Torque for 750rpm and alpha 130=%0.1f N-m\n\n',T1)
printf('\n\n The Field Current=%0.1f \n\n',a11)
printf('\n\n The Field Current=%0.1f \n\n',a22)
//The answers vary due to round off error
```

Scilab code Exa 7.11 Ex11

```
clc
//variable initialization
V=380 //line voltage in volts
P=8 //number of poles
f=50 //frequency in Hz
n=1.25
N1=600 //speed in rpm
N2=400 //speed in rpm

//solution
Ns=(120*f/P)
s=(Ns-N1)/Ns
Vd1=(3*sqrt(6)*s*(V/sqrt(3)))/(%pi*n)
m=(3*sqrt(6)*(V/sqrt(3)))/(%pi*Vd1)
a=acosd(-(s*(n/m)))
s1=(Ns-N2)/Ns
s1=0.4//TRo avoid further Computational errors
Vdc=(3*sqrt(6)*s1*(V/sqrt(3))/%pi)/n
Vd2=(3*sqrt(6)*s1*(V/sqrt(3)))/(%pi*n)
m1=(((3*sqrt(6))/%pi)*(V/sqrt(3)))/Vd2
a1=acosd(s1/(m1/n))
```

```
w1=(2*%pi*Ns)/60
w2=w1/(1+(m/n))//Speed in rad/sec
w21=w2*60/(2*%pi)
printf('\n\n The Firing Angle for 600rpm=%0.1f\n\n',
    a)
printf('\n\n The Firing Angle for 400rpm=%0.1f\n\n',
    a1)
printf('\n\n Minimum Possible Speed=%0.1f rpm\n\n',
    w21)
//The answers vary due to round off error
```

Scilab code Exa 7.12 Ex12

```
clc
//variable initialisation
V=440 //Supply voltage in volts
p=6 //number of poles
f=50 //Supply frequency in Hz
N1=970 //speed in rpm
N2=750 //speed in rpm
N3=850 //speed in rpm
n=3.5//Turns Ratio
R1=0.2
R2=0.15
X1=0.4
X2=0.4
aa1=170//Firing Angle
aa2=140//Firing Angle
s=0.3

//solution
Ns=(120*f)/p
a=-(s/cosd(aa1))
m=(n/a)
s1=(Ns-N2)/Ns
```

```
Vd1=(3*sqrt(6)*s1*(V/sqrt(3)))/(%pi*n)
Vd2=(3*sqrt(6)*(V/sqrt(3)*cosd(aa2)))/(%pi*m)
Vd2=-39.05//To avoid further computational errors assuming Vd2
Rs1=R1*((1/n)^(2))
R3=(R2*((1/n)^(2)))
Rd=0
Id=(Vd1+Vd2)/(2*((s1*Rs1)+R3)+Rd)
w=Ns*((2*%pi)/60)
Td=(abs(Vd2)*Id/(s1*w))
s2=(Ns-N1)/N1
Tr=(3/w)*((((V/sqrt(3))^(2))*(R2/s2))/(R1+(R2/s2))^(2)+(s2)^(2))
s3=(Ns-N3)/Ns
Vd3=(3*sqrt(6)*s3*(V/sqrt(3)))/(%pi*n)
X=poly(0,'X')//X=-cos alpha
0==1769.4*X^2-884.02*X+51.5//Polynomial Eqn obtained
X1=(884.02+sqrt((884.02^2)-4*1769.4*51.5))/(2*1769.4)//Roots of polynomial eqn
X2=(884.02-sqrt((884.02^2)-4*1769.4*51.5))/(2*1769.4)//Roots of polynomial eqn
a11=acosd(-X1)
a22=acosd(-X2)
printf('\n\n Turns Ratio of Transformer=%0.1f\n\n',m)
printf('\n\n The Torqye for 750rpm=%0.1f N-m\n\n',Td)
printf('\n\n Firing Angle 1=%0.1f\n\n',a11)
printf('\n\n Firing Angle 2=%0.1f\n\n',a22)
//The answers vary due to round off error
```

Scilab code Exa 7.13 Ex13

```
clc
//variable initialization
```

```
Vm=600 //Voltage of motor in volt
Pout=30000 // Output Of Motor In Watt
F=50 //Supply frequency in hrtz
P=4 //Number Of Poles
N1=100 //Speed OF Motor In rpm
N2=1000 //Speed OF Motor In rpm
R1=0.05 //Resistance of stator in ohm
R2=0.07 //Resistance of rotor in ohm
R0=53 //Resistance of rotor in ohm
X=0.153 //Reactance of Motor in ohm
X0=23 //Reactance of Motor in ohm
n=1.3 //Stator To Rotr Ratio
N3=300 //Speed OF Motor In rpm


//Solution
Vph=Vm/(sqrt(3))
a=1/n
Ns=(120*F)/(P)
S=(Ns-N2)/Ns
Wm=(2*%pi)/60
Tl=(Pout)/(Wm*N3)
Id=(Tl*Wm*Ns)/(2.339*a*Vph)
I0=Vph/(X0)
I2=(sqrt(2/3))*(Id*a)
Pi=Pout+(R1*((I2)^2))+(R2*((I2)^2))
e=(Pout/Pi)*100
theta=-(atand(Vph/(0.779*Id*a*X0)))
pf=cosd(theta)
printf('\n\n The Motor Efficiency=%0.1f\n\n',e)
printf('\n\n The Power Factor=%0.1f lag\n\n',pf)
//The answers vary due to round off error
```

Chapter 8

Control of Synchronous Motor Drives

Scilab code Exa 8.1 Ex1

```
clc
P=5e+6// power rating in Watts
Vl=11e+3// line voltage in Volts
f=50// frequency of motor in Hz
p1=0
N=6// no. of poles
Rs=0// resistance of motor in ohm
Xs=10// reactance of motor in ohm
If=60// rated field current in amp

//Solution
Vph=Vl/sqrt(3)//phase voltage in Volts
N1=750// speed of motor at rated motor current Is
p2=36.869898//p2=acosd(0.8)
Is=P/(sqrt(3)*Vl*cos(p1))
E=Vph-(Is*Xs*%i)
Ns=120*f/N// synchronous speed of motor
f1=N1*f/Ns// freaquency of motor while running at N1
Vph1=Vph*f1/(f)//phase voltage for speed N1 in V
```

```
Xs1=(N1/Ns)*Xs//reactance of motor at speed N1 in
   ohm
E2=Vph1-(Is*(cosd(p2)+%i*sind(p2))*(Xs1*%i))
E1=E*N1/Ns//in V
If1=abs(E2)*If/abs(E1)//field current at N1 in amp
P1=3*Vph1*Is*cosd(p2)//output power in kW
wm=N1*2*%pi/60//angular speed in rad/sec
T=P1/wm//torque in Nm
printf('\n\n The Field Current=%0.1f Amp\n\n',If1)
printf('\n\n The Torque for Rated Armature Current=
   %0.1f N-m\n\n',T)
//The answers vary due to round off error
```

Scilab code Exa 8.2 Ex2

```
clc
P=500e+3//power of motor in Watts
V=6.6e+3//rated voltage in Volts
f=60//frequency in Hz
n=6//no. of poles
Rs=0//resistnce of motor in ohm
Xm=78//reactance in ohm
Xsr=3//reactance in ohm
p=0//pf=1
k=5

//solution
Xsr1=3*%i
Vph=V/sqrt(3)
Is=P/(3*Vph*cosd(p))
E=Vph-(Is*%i*Xsr)
E1=abs(E)
d=asind((Is*1*Xsr/E1))
Pm=3*Vph*E*sind(d)/Xsr
Pm1=abs(Pm)//Power in watt
```

```
Pm2=Pm1*10^(-3)//Power in Kw
Ns=120*f/n
N=Ns/k
wm=N*2*%pi/60
T=Pm1/wm
If1=E/(%i*Xsr)
Im=Is+abs(If1)
printf('\n\n Power Pm=%0.1f Kw\n\n',Pm2)
printf('\n\n Torque T=%0.1f N-m\n\n',T)
printf('\n\n The Field Current=%0.1f Amp\n\n',abs(
    If1))
printf('\n\n The motor Current=%0.1f Amp\n\n',Im)
//The answers vary due to round off error
```

Scilab code Exa 8.3 Ex3

```
clc
P=5e+5//rated power output in Watts
P1=25e+4// power at half rated torque
f=50//frequency in Hz
If=10//rated firld current in amp
Xs=10//reactance in ohm
p=4//no.of poles
Vl=33e+2//line voltage in volts

Vph=Vl/sqrt(3)//phase voltage in volts
Is=P/(sqrt(3)*Vl*0.8)//Current in amp
theta1=acosd(0.8)
E=Vph-(-%i*Xs*(Is*(%i*sind(theta1)+cosd(theta1))))
y=imag(E)
x=real(E)
Er=sqrt((y^2)+(x^2))
theta2=atand(y/x)
d1=asind(P1*Xs/(3*Vph*(abs(E))))
Is1=(Vph-Er*(cosd(d1)-%i*sind(d1)))/(%i*Xs)
```

```
y1=imag(Is1)
x1=real(Is1)
Is1r=sqrt((y1^2)+(x1^2))
theta3=atand(y1/x1)
pf=cosd(theta3)
Is2=P/(Vph*3)
E2=Vph-(%i*Xs*(Is2*(%i*sind(0)+cosd(0))))
If1=abs(E2)*If/E
If1r=abs(If1)
If3=15//field current in amp
E3=If3*(Er)/If//in volts
Is3=sqrt(((E3^2)-(Vph^2))/(Xs^2))
P3=3*Vph*Is3*cosd(0)
Ns=120*f/p//synchronous speed
wm=Ns*2*%pi/f//in rad/sec
T=P3/wm//in Nm
printf('\n\n The Armature Current at half rated
    torque and rated field current=%0.1f Amp\n\n',Is1
    )
printf('\n\n The power Factor=%0.1f\n\n',pf)
printf('\n\n The Field Current=%0.1f Amp\n\n',abs(
    If1))
printf('\n\n The Torque for upf operation for 15 amp
     field current=%0.1f N-m\n\n',T)
```

Scilab code Exa 8.4 Ex4

```
clc
P=5e+5//rated power output in Watts
P1=25e+4// power at half rated torque
f=50//frequency in Hz
If=10//rated firld current in amp
Xs=10//reactance in ohm
p=4//no.of poles
Vl=33e+2//line voltage in volts
```

```
N1=1500
//Solution
Vph=Vl/sqrt(3)//phase voltage in volts
Is=P/(sqrt(3)*Vl*0.8)//Current in amp
theta1=acosd(0.8)
E=Vph-(-%i*Xs*(Is*(%i*sind(theta1)+cosd(theta1))))
y=imag(E)
x=real(E)
Er=sqrt((y^2)+(x^2))
theta2=atand(y/x)
Ia=Is
E2=Vph+(%i*Ia*Xs)
y2=imag(E2)
x2=real(E2)
Er2=sqrt((y2^2)+(x2^2))
theta3=atand(y2/x2)
P=3*Vph*Er2*sind(-theta3)/Xs
Wms=2*%pi*N1/f
T=P/Wms
If1=Er2*If/Er
If2=12
Er3=Er*If2/If
P2=-500e+3
d1=asind(P2*Xs/(3*Vph*Er3))
Is=(Vph-Er3*(cosd(d1)+(%i*sind(d1))))/(%i*Xs)
Isr=abs(Is)
u=imag(Is)
v=real(Is)
pf=cosd(atand(u/v))
printf('\n\n The Breaking Torque for machine operation at rated current and upf=%0.1f N-m\n\n',T)
printf('\n\n The Field Current for machine operation at rated current and upf=%0.1f Amp\n\n',If1)
printf('\n\n The Armature Current at 12 A Field Current=%0.1f Amp\n\n',Isr)
printf('\n\n The power factor at 12 A Field Current=%0.1f lead\n\n',pf)
```

```
//The answers vary due to round off error
```

Scilab code Exa 8.5 Ex5

```
clc
//Variable Initialisation
Pm=5e+6//motor rating in Watt
V=11e+3//Input voltage in Volts
f1=50//Supply Frequency
pf=0.9//power factor of motor
N1=1000//rated speed
Rs=0//resistance in ohm
Xs=10//reactance in ohm
N2=750
N3=1500
pf2=0.8
If1=50//rated field current
//Solution
V1=V/sqrt(3)
Is=Pm/(3*V1*pf)
Is1=Is*(cosd(25.84)+(%i*sind(25.84)))
E=V1-(Is1*%i*Xs)
y=imag(E)
x=real(E)
Er=sqrt((y^2)+(x^2))
theta=atand(y/x)
theta1=acosd(0.8)
f2=f1*N2/N1
V2=V1*f2/f1
Xs2=Xs*f2/f1
Is2=Is*(cosd(theta1)+(%i*sind(theta1)))
E2=V2-(Is2*%i*Xs2)
t=imag(E2)
u=real(E2)
Er2=sqrt((t^2)+(u^2))
```

```
theta3=atand(t/u)
E3=Er*N2/N1
If2=If1*Er2/E3
P2=3*V2*Is*pf2
W=2*%pi*N2/f1
T=P2/W
f3=f1*N3/N1
Xs3=f3/f1*Xs
E4=Er*f3/f1
P3=0.75*Pm
k=asind(Xs3*P3/(3*V1*E4))
Is3=(V1-(E4*(cosd(k)+(%i*sind(k)))))/(Xs3*%i)
y2=imag(Is3)
x2=real(Is3)
Is3r=sqrt((y2^2)+(x2^2))
theta4=atand(y2/x2)
pf3=cosd(theta4)
printf('\n\n The torque for Rated armature current
    ,750rpm,0.8pf=%0.1f N-m\n\n',T)
printf('\n\n The Field Current for Rated armature
    current,750rpm,0.8pf=%0.1f Amp\n\n',If2)
printf('\n\n The Armature Current for half the Rated
     torque,1500rpm,rated field current=%0.1f Amp\n\n
    ',Is3r)
printf('\n\n The power factor for half the Rated
    torque,1500rpm,rated field current=%0.1f\n\n',pf3
    )
```

Scilab code Exa 8.6 Ex6

```
clc
//Variable Initialisation
Vs=6.6e+3//Supply voltage in Volts
f1=50//Supply Frequency
Ns=1000//rated motor speed
```

```
Rd=0.2//dc link inductor resistance in ohm
Xs=2.6//Reactance in ohm
P=10e+6//motor rating in Watt
pf1=1
al=150
//solution
V1=Vs/sqrt(3)
Is=P/(3*V1*pf1)
Id=Is*%pi/sqrt(6)
phi=180-al
N2=500
f2=f1*N2/Ns
Vph=V1*N2/Ns
P2=3*Vph*Is*cosd(phi)
Pd=P2*10^(-6)//Power delivered in mega watt
Vdl=3*sqrt(6)*Vph*cosd(al)/%pi
Vds=(Id*Rd)-Vdl
A=Vds*%pi/(3*sqrt(6)*V1)
as=acosd(A)
N3=600
f3=f1*N3/Ns
Vph2=V1*N3/Ns
P3=3*Vph2*Is*pf1
Ps=P3-((Id^2)*Rd)
Ps2=Ps*10^(-6)
Vdl2=3*sqrt(6)*Vph2*pf1/%pi
Vds2=(Id*Rd)-Vdl2
B=Vds2*%pi/(3*sqrt(6)*V1)
as2=acosd(B)
printf('\n\n The Power Delivered by Motor=%0.1f MWatt\n\n',Pd)
printf('\n\n The Firing angle for motoring operation =%0.1f\n\n',as)
printf('\n\n The Power supplied to source =%0.1f MWatt\n\n',Ps2)
printf('\n\n The Firing angle for regenerative braking operation=%0.1f\n\n',as2)
```