Scilab Textbook Companion for Electrical Machines by R. K. Srivastava[1]

Created by
Balachandra P
Bachelor of Engineering
Electrical Engineering
Sri Jayachamarajendra College of Engineering
College Teacher
None
Cross-Checked by
Bhavani Jalkrish

August 13, 2014

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

Book Description

Title: Electrical Machines

Author: R. K. Srivastava

Publisher: Cengage Learning, New Delhi

Edition: 1

Year: 2011

ISBN: 9788131511701

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

List of Figures

Chapter 2

Forces in a Electromagnetic System

Scilab code Exa 2.1 To find flux flux density and magnetic field intensity in the core

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 2 : FORCES IN AN ELECTROMAGNETIC
   SYSTEMS

// EXAMPLE : 2.1

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

A = 0.0001;                    // The Cross-sectional
```

```
    area of core in metre-square
Mo = 4*%pi*(10)^(-7);        // Permeability of air in Henre/metre
Mr = 1000;                   // Relative permeability of core
N1 = 10;N2=20;N3=10;         // Number of turns
I1 = 1.0;I2=0.5;I3=1.5;      // Currents in Amphere
d = 2.5;                     // Dimension of inner window in centimetre
w = 1.0;                     // Each limb wide in centimeter


// CALCULATIONS

F = (N1*I1)+(N2*I2)-(N3*I3);            // MMF in Amphere-turns (minus because third coil produces the flux in opposite direction to that of other to coils)
L = ((d*4)+(I2*2*4))*10^-2;             // Length of the Magnetic path in metre (4-is sides of the windows)(2-Going and returning of current I2)
R = L/(Mr*Mo*A);                        // Reluctance of the Magnetic path in MKS unit of Reluctance
phi = (F*10^3)/R;                       // Flux in milli-Weber
B = phi/A;                              // Flux Density in Weber/metre Square
H = F/L;                                // Magnetic Field Intensity in Amphere-turns/Metre


// DISPLAY RESULTS

disp("EXAMPLE : 2.1 : SOLUTION :-") ;
printf("\n (a) Flux in the core, phi = %.6f mWb ,\n",phi);
printf("\n (b) Flux Density in the core, B = %.2f Wb
```

```
/metre square \n",B);
printf("\n (c) Magnetic Field Intensity in the core, H = %.2f At/m \n",H);
```

Scilab code Exa 2.2 To find Total MMF coil current relative permeability of each ferromagnetic material and inductance of the coil

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 2 : FORCES IN AN ELECTROMAGNETIC SYSTEMS

// EXAMPLE : 2.2

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

N = 100;                 // Number of turns
La = 0.3;                // Mean arc length of material "a" is a Nickel-iron alloy in Metre
Lb = 0.2;                // Mean arc length of material "b" is a Steel in Metre
Lc = 0.1;                // Mean arc length of material "c" is a Cast Steel in Metre
a = 0.001;               // Area of the all Materials "a,b,c" in Metre-Square
phi = 6*10^-4;           // Magnetic Flux in Weber
mue_0 = 4*%pi*10^ -7;    // Permeability of the air in
```

```
Henry/Meter


// CALCULATIONS

B = phi/a;                          // Flux Density in Telsa (Here Flux Density same for all the Materials "a,b,c" because Area of Cross Section is Same)
Ha = 10;                            // Fileld Intensity in Amphere−Turn/Meter Corresponding to Flux density (B) of material "a" obtained from the Standard B−H curve
Hb = 77;                            // Fileld Intensity in Amphere−Turn/Meter Corresponding to Flux density (B) of material "b" obtained from the Standard B−H curve
Hc = 270;                           // Fileld Intensity in Amphere−Turn/Meter Corresponding to Flux density (B) of material "c" obtained from the Standard B−H curve
F = (Ha*La)+(Hb*Lb)+(Hc*Lc);        // The Total MMF Required in Amphere−Turns
I = F/N;                            // Current flowing through the Coil in Amphere
mue_r_a = B/(Ha*mue_0);            // Relatative permeability of the Material "a"
mue_r_b = B/(Hb*mue_0);            // Relatative permeability of the Material "a"
mue_r_c = B/(Hc*mue_0);            // Relatative permeability of the Material "a"
Ra = (Ha*La)/phi;                  // Relucatnce of the Material "a" in MKS unit
Rb = (Hb*Lb)/phi;                  // Relucatnce of the Material "b" in MKS unit
Rc = (Hc*Lc)/phi;                  // Relucatnce of the Material "c" in MKS unit
L = (N*phi)/I;                     // Inductance of the
```

```
Coil in Henry


// DISPLAY RESULTS

disp("EXAMPLE : 2.2 : SOLUTION :-") ;
printf("\n (a)  The Total MMF , F = %.1f At \n ",F);
printf("\n (b)  Current flowing through the Coil , I = %.3f A \n",I);
printf("\n (c.1) Relatative permeability of the Material a, mue_r_a = %.f \n ",mue_r_a);
printf("\n (c.2) Relatative permeability of the Material b, mue_r_b = %.f \n ",mue_r_b);
printf("\n (c.3) Relatative permeability of the Material c, mue_r_c = %.f \n ",mue_r_c);
printf("\n (c.4) Relucatnce of the Material a, Ra= %.f MKS unit \n",Ra);
printf("\n (c.5) Relucatnce of the Material b, Rb= %.1f MKS unit \n",Rb);
printf("\n (c.6) Relucatnce of the Material c, Rc= %.f MKS unit \n",Rc);
printf("\n (d)  Inductance of the Coil , L = %.4f H \n",L);
```

Scilab code Exa 2.3 To find magnetic field produced by an appiled MMf of 35At

```

// ELECTRICAL MACHINES
// R.K.Srivartava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 2 : FORCES IN AN ELECTROMAGNETIC
```

```
SYSTEMS

// EXAMPLE : 2.3

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

F = 35;                       // Total MMF in Amphere-Turns
Lc = 0.1;                     // Inductance of The Material "c" in Henry
a = 0.001;                    // Area of the all Materials "a,b,c" in Metre-Square


// CALCULATIONS

Hc = F/Lc;                    // Field Intensity in Amphere-Turns/Meter (Given that entire MMf apperas on Material "c" Because of the highest reluctance about 45000 MKS unit From Example 2.2)
Bc = 0.65;                    // Flux density of material "c" in in Telsa obtained from the Standard B-H curve
phi = Bc*a;                   // Flux in the core in Weber
Ba = Bc;                      // Flux density of material "a" in in Telsa Same because Area of Cross Section is Same
Bb = Bc;                      // Flux density of material "b" in in Telsabecause Area of Cross Section is Same


// DISPLAY RESULTS

disp("EXAMPLE : 2.3 : SOLUTION :-") ;
```

```
printf("\n (a) Flux in the core , phi = %.5f Wb \n ",phi);
printf("\n (b) Flux density of material a,b,c , Ba = Bb = Bc %.2f T \n",Ba);
```

Scilab code Exa 2.4 To find flux density in air gap output of the sensor if hall effect generates voltage of 50 millivolt per 1 telsa and ratio of flux density without and with air gap

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 2 : FORCES IN AN ELECTROMAGNETIC SYSTEMS

// EXAMPLE : 2.4

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA
// Refer figure 2.7:- Page no. 41

a = 0.0001;                    // Cross Sectional Area of the Core in Meter-Square
Li = 0.158;                    // Total length of the Path abcdef in Meter (4.0*4.0 - 0.2 = 15.8cm = 0.158m)
Lg = 0.002;                    // Length of the air gap in Meter
mue_0 = 4*%pi*10^-7;           // Permeability of the
```

```
air in Henry/Meter
mue_r = 10000;                  // Permeability of the core
N = 10;                         // Number of Turns
I = 1.0;                        // Current in the Coil in Amphere
v = 50;                         // hall effect sensor generates volatge produces in milli volt per 1 Telsa
Li_new = 0.16;                   // Length of the Flux path in Absence of the Air gap in Meter


// CALCUALTIONS

F = N*I;                              // MMF of the Coil in Amphere-turn
Ri = Li/(mue_0*mue_r*a);              // Relucatnce of the Iron Coil in MKS unit
Rg = Lg/(mue_0*a);                    // Relucatnce of air gap in MKS unit
R = Ri+Rg;                            // Total Reluctance in MKS unit
phi = F/R;                            // Flux in the Core in Weber
B = phi/a;                            // FLux density in the core(Presence of the Air gap) in Weber/Meter-Square
HEV = B*50;                           // Output of the Hall effect Sensor device in Milli-Volt
R_new = Li_new/(mue_0*mue_r*a)    // Relucatance of the Magnetic Circuit in Absence of the Air gap
phi_new = F/R_new;                    // New Flux in the Core in Weber
B_new = phi_new/a;                    // New FLux density in the core in Weber/Meter-Square
Ratio = B_new/B;                      // Ratio of the Flux Density in Absence of the Air gap and in the
```

```
presence of the Air gap


// DISPLAY RESULTS

disp("EXAMPLE : 2.4 : SOLUTION :-") ;
printf("\n (a) Flux density in the core(Presence of the Air gap) , B = %.8f Wb/Meter-Square \n ",B);
printf("\n (b) Output of the Hall effect Sensor device , HEV = %.7f mV \n",HEV);
printf("\n (c) Ratio of the Flux Density in Absence of the Air gap and in the presence of the Air gap , Ratio = %.2f \n ",Ratio);
```

Scilab code Exa 2.5 To find how many turns should the exciting coil have in order to establish a flux density of 1 Wb per meter square if core made up of silicon steel having 1 telsa and 200 At per meter and ferrite core having relative permeability of 20000

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 2 : FORCES IN AN ELECTROMAGNETIC SYSTEMS

// EXAMPLE : 2.5

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA
```

```
// Refer figure 2.3(a):- Page no. 36

B = 1.0;                          // Flux Density in the Core in Weber/Meter-Square
Liron = 0.55;                     // Mean length of the flux path of Iron in Meter
Lair = 0.002;                     // Mean length of the flux path of Air Gap in Meter
I = 20;                           // Coil Current in Amphere
H = 200;                          // Field Intensity in Amphere-Turns/Meter
mue_r = 20000;                    // Relative permeability of Ferrite core
mue_0 = 4*%pi*10^-7;              // Permeability of the air in Henry/Meter
a = 0.0025;                       // Area of the Cross sectional of the core oin Metre-Square


// CALCULATIONS

phi = B*a;                                  // Toatl Flux in the core in Weber
Rair = Lair/(mue_0*a);                      // Relucatnce in the Air gap
Fair = Rair*phi;                            // MMf in the Air gap in Amphere-Turns
Firon = H*Liron;                            // MMf in the Iron core in Amphere-Turns
F = Firon+Fair;                             // Total MMF in Amphere-Turns
N = F/I;                                    // Number of turns in the Coil
F_new = B/(mue_0*mue_r);                    // Field Intensity in Amphere-Turns/Meter
F_new_total = (Fair+F_new);                 // Total MMF in Amphere-Turns
```

```
N_new = F_new_total/I;                          // Number of
   turns in the Coil


// DISPLAY RESULTS

disp("EXAMPLE : 2.5 : SOLUTION :-") ;
printf("\n (a) Number of turns in the Coil in air
   gap made of Silicon Steel having an field
   intensity 200At/m corresounds to 1.0 T Flux
   Density , N = %.2f appoximately 85 \n ",N);
printf("\n (b) Number of turns in the Coil for a
   ferrite core of having Relative premeability of
   20000 and magnetic Field Density corresponnds to
   1.0 T , N_new = %.2f appoximately 82  \n",N_new);
```

Chapter 3

Transformers

Scilab code Exa 3.1 To find voltage drops line losses and generating voltage

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.1

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

Z = (0.05 + 0.05 * %i) * 100;          //
    Transmission line parameters (impedance) in Ohms
    (multiplied by 100 because distance of the
    Transmission line is 100km)
```

```
R = 0.05 * 100;                             // Transmission line Resistance in Ohms (multiplied by 100 because distance of the Transmission line is 100km)
V1 = 220;                                   // Terminal voltage in Volts
V2 = 1 * 10 ^ 3;                            // Terminal volatge from Generator side in Volts
P = 20 * 10 ^ 3;                            // Power in Watts


// CACULATIONS

I1 = P/V1;                                  // Line current for 220V in Amphere
I2 = P/V2;                                  // Line current for 1kV in Amphere
I1Z = Z*I1;                                 // Voltage drop due to I1 in Volts
I2Z = Z*I2;                                 // Voltage drop due to I2 in Volts
Loss1 = (I1 ^ 2) * R * 10 ^ -3;             // Line loss for I1 in kW
Loss2 = (I2 ^ 2) * R * 10 ^ -3;             // Line loss for I2 in kW
Vg1 = V1 + I1Z;                             // Input Voltages on Generator Terminal in Volts
Vg2 = V2 + I2Z;                             // Input Voltages on Generator Terminal in Volts


// DISPLAY RESULTS

disp("EXAMPLE : 3.1 : SOLUTION :-") ;
printf("\n (a.1) Voltage drop due to I1 , I1Z = %.2f+j%.2f V \n ",real(I1Z),imag(I1Z));
printf("\n (a.2) Voltage drop due to I2 , I2Z = %.f
```

```
    +j%.f  V \n",real(I2Z),imag(I2Z));
printf("\n (b.1) Line loss for I1 , Loss1 = %.2f kW
    \n ",Loss1);
printf("\n (b.2) Line loss for I2 , Loss2 = % .2f kW
     \n",Loss2);
printf("\n (c.1) Input Voltages on Generator
    Terminal from a load terminal , Vg1 = %.2f+j%.2f
    = %.2f V \n ",real(Vg1),imag(Vg1),abs(Vg1));
printf("\n (c.2) Input Voltages on Generator
    Terminal from a Generating Station , Vg2 = % .f+
    j%.f = %.2f V \n",real(Vg2),imag(Vg2),abs(Vg2));
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED
    WRONGLY ( I verified by manual calculation )]\n"
    );
printf("\n       WRONGLY PRINTED ANSWERS ARE :- (a)
    I1Z = (450.45)+j(450.45)V instead of (454.55)+j
    (454.55) V\n" );
printf("\n                                    (b)
    Vg1 = (670.45)+j(450.45) = 807.72 V  instead of %
     .2f+j%.2f = %.2f V \n",real(Vg1),imag(Vg1),abs(
    Vg1) );
```

Scilab code Exa 3.2 To find numbers of turns in each winding and voltage per turn

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.2

```

```
clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

E1 = 6.6 * 10 ^ 3;                 // Primary voltage in Volts
E2 = 220;                          // Secondary Voltage in volts
f = 50;                            // Frequency in Hertz
phi_m = 0.06;                      // Flux in Weber
S = 50 * 10^6;                     // Rating of the single-phase transformer in VA

// CALCULATIONS

N1 = E1/(4.44*f*phi_m);            // Number of turns in Primary
vpn1 = E1/N1;                       // Voltage per turns in Primary in Volts/turn
N2 = E2/(4.44*f*phi_m);            // Number of turns in Secondary
vpn2 = E2/N2;                       // Voltage per turns in Secondary in Volts/turn


// DISPLAY RESULTS

disp("EXAMPLE : 3.2 : SOLUTION :-") ;
printf("\n (a.1) Number of turns in Primary , N1 = %.1f Turns nearly 496 Turns \n ",N1);
printf("\n (a.2) Number of turns in Secondary , N2 = %.1f Turns nearly 16 Turns \n",N2);
printf("\n (b.1) Voltage per turns in Primary , vpn1 = %.1f Volts/turns \n ",vpn1);
printf("\n (b.2) Voltage per turns in Secondary ,
```

```
vpn2 = %.2f Volts/turns \n ",vpn2);
```

Scilab code Exa 3.3 To find maximum value of the core flux

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.3

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

f = 50;                    // Frequency in Hertz
N = 50;                    // Number of turns in Secondary
E =220;                    // Induced voltage in Volts


// CALCULATIONs

phi_m = E/(4.44*f*N);          // Maximum value of the Flux in Weber


// DISPLAY RESULTS

disp("EXAMPLE : 3.3 : SOLUTION :-") ;
```

```
printf("\n (a) Maximum value of the Flux , phi_m = %
    .7f Wb \n ",phi_m);
```

Scilab code Exa 3.4 To find rated currents of the two windings

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.4

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

S = 1.5;                  // Transformer Rating in KVA
E1 = 220;                 // HV side voltage in volts
E2 = 40;                  // LV side voltage in volts


// CALCULATION

Ihv = (S * 10 ^ 3)/E1;              // Rated HV side
    Curent in Amphere
Ilv = (S * 10 ^ 3)/E2;              // Rated lV side
    Curent in Amphere


// DISPLAY RESULTS
```

```
disp("EXAMPLE : 3.4 : SOLUTION :-") ;
printf("\n (a) Rated HV side Curent , Ihv = %.2f A \n ",Ihv);
printf("\n (b) Rated LV side Curent , Ilv = %.1f A \n",Ilv);
```

Scilab code Exa 3.5 To find maximum flux no load current no load power factor of the transformer

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.5

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

Ai = 2.3 * 10 ^ -3;                    // Cross-Sectional area of the core in Meter-Square
mue_0 = 4*%pi*10^ -7;                  // Permeability of the air in Henry/Meter
Fe_loss = 2.6;                         // Iron loss at the working Flux density Watts/kg
Fe_den = 7.8 * 10 ^ 3;                 // Density of the Iron in kg/Meter-Cube
N1 = 800;                              // Number of
```

```
Turns of the Primary winding
L = 2.5;                              // Length of the Flux path in Meter
mue_r = 1000;                         // Relative Permeability
E = 400;                              // Primary Volatge of the Transformer in Volts
f = 50;                               // Frequency in Hertz


// CALCULATIONS

Bm = E/(4.44*f*Ai*800);               // Flux Density in Weber/Meter-Square
phi_m = (Bm*Ai)*10^3;                 // Maximum Flux in the core in milli-Weber
F = (L*Bm)/(mue_r*mue_0);             // Magnetizing MMF in Amphere-turns
Im = F/(N1*sqrt(2));                  // Magnetizing Current in Amphere
Vol = L*Ai;                           // Volume of the Core in Meter-Cube
W = Vol * Fe_den;                     // Weight of the Core in kg
Total_Fe_loss = Fe_loss * W;          // Total Iron loss in Watt
Ic = Total_Fe_loss/E;                 // Loss component of Current in Amphere
Io= sqrt((Ic ^ 2)+(Im ^ 2));          // No load Current in Amphere
pf_angle = atand(Io/Ic);              // No load Power factor angle in degree
pf = cosd(pf_angle);                  // No load Power factor


// DISPLAY RESULTS
```

```

disp("EXAMPLE : 3.5 : SOLUTION :-") ;
printf("\n (a) Maximum Flux in the core , phi_m = %
    .6f mWb \n ",phi_m);
printf("\n (b) No load Current , I0 = %.3f A \n",Io)
    ;
printf("\n (c) No load Power factor angle = %.3f
    degree \n ",pf_angle);
printf("\n (d) No load Power factor = %.4f \n ",pf)
    ;
```

Scilab code Exa 3.6 To find turn ratio primary and secondary currents at full load and at 4kW load at pf lagging 80 percent

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.6

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

S = 5;                    // Transformer Rating in kVA
V1 = 220;                 // HV side voltage in volts
V2 = 110;                 // LV side voltage in Volts
P = 4 * 10 ^ 2;           // Load of the Transformer
pf = 0.8;                 // Power Factor (lagging)
```

```
f = 50;                    // Frequency in Hertz


// CALCULATIONS

a = V1/V2;                 // Turn Ratio of the Transformer

// case (a) At full load
I1 = (S * 10 ^ 3)/V1;        // Primary current at full load in Amphere
I2 = (S * 10 ^ 3)/V2;        // Secondary Current at full Load in Amphere

// Case (b) At 4kW, 0.8 lagging pf load
I11 = (4 * 10 ^ 3 * 0.8)/V1;         // Primary current At 4kW, 0.8 lagging pf load in Amphere
I22 = (4 * 10 ^ 3 * 0.8)/V2;         // Secondary Current At 4kW, 0.8 lagging pf load in Amphere


// DISPLAY RESULTS

disp("EXAMPLE : 3.6 : SOLUTION :-") ;
printf("\n   (a)   Turn Ratio of the Transformer , a = %.f \n ",a);
printf("\n (b.1.1) Primary current at full load , I1 = %.2f A \n",I1);
printf("\n (b.1.2) Secondary current at full load , I2 = %.2f A \n ",I2);
printf("\n (b.2.1) Primary current at 4kW, 0.8 lagging pf load , I1 = %.3f A \n",I11);
printf("\n (b.2.1) Secondary current at 4kW, 0.8 lagging pf load , I2 = %.3f A \n",I22);
```

Scilab code Exa 3.7 To find referred value of resistance from primary side

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.7

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

a = 100/200;                              // Turn ratio of the Ideal transformer
R = 1.0;                                  // Resistance across the secondary side having 200 turns in Ohms


// CALCULATIONS

R1 = (a ^ 2)*R;                           // Referred value of the resistance from Primary side having 100 turns in Ohms


// DISPLAY RESULTS

disp("EXAMPLE : 3.7 : SOLUTION :-") ;
printf("\n (a) Referred value of the %.f Ohm resistance from Primary side having 100 turns , R1 = %.2f ohms \n ",R,R1);
```

```
printf("\n\n      [ TEXT BOOK SOLUTION IS PRINTED
   WRONGLY ( I verified by manual calculation )]\n"
   );
printf("\n       WRONGLY PRINTED ANSWERS ARE :- (a)
    Referred value of the resistance from Primary
    side having 100 turns = 0.025 Ohms instead of %.2
    f Ohms \n ",R1);
```

Scilab code Exa 3.8 To find equivalent resistance as referred to primary and secondary side and full load copper loss

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.8

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

S = 60;                    // Tranformer Rating in kVA
V1 = 6600;                 // HV Side Voltage Rating of
    the Transformer in Volts
V2 = 220;                  // LV Side Voltage Rating of
    the Transformer in Volts
R1 = 7.8;                  // primary Resistances of the
     Transformer in Ohms
R2 = 0.0085;               // Secondary Resistances of
```

```
the Transformer in Ohms


// CALCULATIONS

a = V1/V2;                // Transformation Ratio
Rp = R1+(a^2)*R2;         // Resistance referred to Primary side in Ohms
Rs = (R1/(a^2))+R2;       // Resistance referred to Secondary side in Ohms
Ip = (S*1000)/V1          //  Current in Primary Side in Ampheres
Cu_loss = Rp*(Ip^2);      // Copper loss in Transformer in Watts


// DISPLAY RESULTS

disp("EXAMPLE : 3.8 : SOLUTION :-") ;
printf("\n (a) Equlivalent Resistance as Referred to Primary Side, Rp = %.2f ohms \n",Rp)
printf("\n (b) Equlivalent Resistance as Referred to Secondary Side, Rs = %.5f ohms \n",Rs)
printf("\n (c) Total Copper Loss, Cu_loss = %.2f W \n",Cu_loss)
```

Scilab code Exa 3.9 To find equivalent resistance referred from primary side

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

```

```
// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.9

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

V1 = 11000;              // HV Side Voltage Rating of
    the Transformer in Volts
V2 = 440;                // LV Side Voltage Rating of
    the Transformer in Volts
R = 1.0;                 // Resistance across the
    secondary side having 11kV in Ohms


// CALCULATIONS

a = V1/V2;                              // Turns ratio
    of the ideal transformers
R2 = (a ^ 2)*R;                         // Referred
    value of the resistance from Primary side having
    440V in Ohms


// DISPLAY RESULTS

disp("EXAMPLE : 3.9 : SOLUTION :-") ;
printf("\n (a) Referred value of the resistance from
     Primary side having 440V , R2 = %.f Ohms \n ",R2
    );
```

Scilab code Exa 3.10 To find current taken by primary side

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.10

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

V1 = 440;                                       // HV Side Voltage Rating of the Transformer in Volts
V2 = 220;                                       // LV Side Voltage Rating of the Transformer in Volts
pf_o = 0.2;                                     // No-load Power factor lagging
pf_l = 0.8;                                     // Load Power factor lagging
I_o = 5;                                        // No-load current in Amphere
I_2 = 120;                                      // Load current in Amphere

// CALCULATIONS

a = V1/V2;                                      // Turns ratio of the two winding Transformers
theta_o = acosd(pf_o);                          // No load power factor of the two winding Transformers in Degrees
```

```
Io = I_o * exp(-(%i*theta_o*%pi/180));                    // No load current of the two winding Transformers (minus because lagging) in Amphere
theta = acosd(pf_l);                                       // load power factor of the two winding Transformers in Degrees
I2 = I_2 * exp(-(%i*theta*%pi/180));                       // secondary load current of the two winding Transformers (minus because lagging) in Amphere
I21 = I2/a;                                                // Secondary referred to the primaryin Amphere
I1 = Io + I21;                                             // Primary current in Amphere
I1_mag = abs(I1);                                          // Primary current magnitude inj Amphere
theta_1 = atand( imag(I1),real(I1));                       // Primary current angle in Degree


// DISPLAY RESULTS

disp("EXAMPLE : 3.10 : SOLUTION :-") ;
printf("\n (a) Primary current , I1 = %.2f < %.1f A \n ",I1_mag,theta_1);
```

Scilab code Exa 3.11 To find core loss no load pf angle pf current through core loss component and core loss resistance current through Xm percentage of exciting current

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

```

```
// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.11

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

S = 50;                 // kVA Rating of the Transformer
f = 50;                 // Frequency in Hertz
Wo = 190;               // Meter Readings when HV Winding kept open in Watt
Vo = 230;               // Meter Readings when HV Winding kept open in Volts
Io = 6.5;               // Meter Readings when HV Winding kept open in Amphere
R2 = 0.06;              // Resistance of the LV Winding in Ohms
V1 = 2300;              // Voltage across the HV Side in Volts
V2 = 230;               // Voltage across the LV Side in Volts
AC = 230;               // Tranformer connected to AC mains in Volts


// CALCULATIONS

a = V1/V2;                                    // Trasformation ratio of the Transformer
Wc = Wo - ((Io ^ 2) * R2);                    // Core loss in Watts
Po = Wc;                                      // Core loss in Watts
Pc = Wc;                                      // Core loss in Watts
```

```
cos_theta_o = Po/(Vo*Io);                              // No load power factor
theta_o = acosd(cos_theta_o);                          // No load power factor angle in Degrees
Ic = Io * cosd(theta_o);
E = V1 - Io * exp(%i*(theta_o)*%pi/180);
Rc = Pc/(Ic ^ 2 );                                      // Core loss Resistance in Ohms
Im = Io * sind(theta_o);                                // Current through the Magnetizing branch in Amphre
Xm = E/Im;                                              // Magnetizing  Reactance in Ohms
Ift = (S * 10 ^ 3)/V2;                                  // Full Load current in Amphere
Ie = (Io/Ift)*100;                                      // Percentage of the Exicting Current in Amphere


// DISPLAY RESULTS

disp("EXAMPLE : 3.11 : SOLUTION :-") ;
printf("\n  (a)  Core loss , Wc = %.2f W \n ",Wc);
printf("\n (b.1) No load power factor angle , theta_o = % .2f Degree \n",theta_o);
printf("\n (b.2) No load power factor , cos(theta_o) = % .6f \n",cos_theta_o );
printf("\n (c.1) Curent through Core loss Component , Ic = %.4f A \n ",Ic);
printf("\n (c.2) Core loss Resistance , Rc = %.2f Ohms \n ",Rc);
printf("\n  (d)  Current through the Magnetizing Component Xm , Im = % .2f A \n",Im);
printf("\n  (e)  Percentage of the Exicting Current = % .2f Percent \n",Ie);
```

Scilab code Exa 3.12 To find hysteresis and eddy current losses st 50Hz

```

// ELECTRICAL MACHINES
// R.K. Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.12

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

N1 = 1000;                    // 1st Test at No-load
    condition f1 Frequency, Speed in RPM
Vo1 = 250;                    // 1st Test at No-load
    condition f1 Frequency, Voltage in Volts
Io1 = 0.5;                    // 1st Test at No-load
    condition f1 Frequency, Current in Amphere
Wo1= 230;                     // 1st Test at No-load
    condition f1 Frequency, Power in Watts

N2 = 900;                     // 2nd Test at No-load
    condition f2 Frequency, Speed in RPM
Vo2 = 225;                    // 2nd Test at No-load
    condition f2 Frequency, Voltage in Volts
Io2= 0.5;                     // 2nd Test at No-load
    condition f2 Frequency, Current in Amphere
Wo2 = 200;                    // 2nd Test at No-load
    condition f2 Frequency, Power in Watts
p = 6;                       // Number of poles of single
     phase alternator
N = 220;                     // Number of the turns of
```

```
single phase alternator
R = 0.66;                  // Resistance of the single phase alternator in Ohms


// CALCULATIONS

f1 = (N1*p)/120;                // 1st case Supply Frequency in Hertz
Ratio1 = Vo1/f1;                // 1st case Ratio of the Volatge and Frequency in Volts/Hertz
f2 = (N2*p)/120;                // 2nd case Supply Frequency in Hertz
Ratio2 = Vo2/f2;                // 2nd case Ratio of the Volatge and Frequency in Volts/Hertz

c = (Wo1-(Io1^2)*R)/f1;           // No-load corrected losses Eq 1 in Watts
d = (Wo2-(Io2^2)*R)/f2;           // No-load corrected losses Eq 2 in watts

x = [ 1 f1 ; 1 f2 ];              // No-load corrected losses Eq 1 in watts
y = [ c ; d ];                    // No-load corrected losses Eq 2 in watts

E = x\y;                          // Solution of constants A in Watts/Hertz and B in watts/Hertz-Sqare in matrix form
A = E(1,1);                       // Solution of constant A in Watts/Hertz
B = E(2,1);                       // Solution of constant B in watts/Hertz-Sqare
Ph = f1*A;                        // Hysteresis loss at 50 Hertz in Watts
Pe = (f1^2)*B;                    // Eddy current loss at 50 Hertz in Watts

```

```

// DISPLAY RESULTS

disp("EXAMPLE : 3.12 : SOLUTION :-") ;
printf("\n (a) Hysteresis loss at %.f Hertz , Ph = %.3f W \n ",f1,Ph);
printf("\n (b) Eddy current loss at %.f Hertz , Pe = % .2f W \n",f1,Pe);
```

Scilab code Exa 3.13 To find hysteresis and eddy currents

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.13

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

N1 = 1500;                 // 1st Test on Transformer at f1 Frequency and Vo1 voltage , Speed in RPM
Vo1 = 250;                 // 1st Test on Transformer at f1 Frequency and Vo1 voltage , Voltage in Volts
Wo1= 55;                   // 1st Test on Transformer at f1 Frequency and Vo1 voltage , Power in Watts
N2 = 1200;                 // 2nd Test on Transformer at f2 Frequency and Vo2 voltage , Speed in RPM
```

```
Vo2 = 200;                    // 2nd Test on Transformer at f2 Frequency and Vo2 voltage, Voltage in Volts
Wo2 = 40;                     // 2nd Test on Transformer at f2 Frequency and Vo2 voltage, Power in Watts
p = 4;                        // Number of poles of single phase alternator
// CALCULATIONS
f1 = (N1*p)/120;                    // 1st case Supply Frequency in Hertz
Ratio1 = Vo1/f1;                    // 1st case Ratio of the Volatge and Frequency in Volts/Hertz
f2 = (N2*p)/120;                    // 2nd case Supply Frequency in Hertz
Ratio2 = Vo2/f2;                    // 2nd case Ratio of the Volatge and Frequency in Volts/Hertz
c = Wo1/f1;            // No-load corrected losses Eq 1 in Watts
d = Wo2/f2;            // No-load corrected losses Eq 2 in watts
x = [ 1 f1 ; 1 f2 ];                     // No-load corrected losses Eq 1 in watts
y = [ c ; d ];                           // No-load corrected losses Eq 2 in watts
E = x\y;                                // Solution of constants A in Watts/Hertz and B in watts/Hertz-Sqare in matrix form
A = E(1,1);                              // Solution of constant A in Watts/Hertz
B = E(2,1);                              // Solution of constant B in watts/Hertz-Sqare
Ph1 = f1*A;                              // Hysteresis loss at 50 Hertz in Watts
```

```
Pe1 = (f1^2)*B;                    // Eddy current loss at 50 Hertz in Watts
Ph2 = f2*A;                        // Hysteresis loss at 40 Hertz in Watts
Pe2 = (f2^2)*B;                    // Eddy current loss at 40 Hertz in Watts


// DISPLAY RESULTS

disp("EXAMPLE : 3.13 : SOLUTION :-") ;
printf("\n (a.1) Hysteresis loss at %.f Hertz , Ph = %.f W \n ",f1,Ph1);
printf("\n (a.2) Eddy current loss at %.f Hertz , Pe = %.f W \n",f1,Pe1);
printf("\n (b.1) Hysteresis loss at %.f Hertz , Ph = %.f W \n ",f2,Ph2);
printf("\n (b.2) Eddy current loss at %.f Hertz , Pe = %.f W \n",f2,Pe2);
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n       WRONGLY PRINTED ANSWERS ARE :- (a) Hysteresis loss at %.f Hertz , Ph = 25 W instead of %.f W \n ",f2,Ph2);
```

Scilab code Exa 3.14 To find efficiency at full load and half load at power factor of unity 80 percentage lagging and leading and also maximum efficiency ant output at which maximum efficiency occurs

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
```

```
// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.14

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

S = 10 * 10 ^ 3;          // Rating of the Single Transformer in VA
f = 50;                   // Frequency in Hertz
Pc = 110;                 // Required input no-load at normal voltage in Watts (Core loss)
Psc = 120;                // Required input Short-circuit at full-load current in Watts (copper loss or short circuit loss)


// CALCUATIONS
// case (a) for Unity power factor

cos_theta1 = 1;                                          // Unity Power factor
K1 = 1.0;                                                // Full load
K2 = 0.5;                                                // Half load
eta_11 = 100 * (K1*S*cos_theta1)/((K1*S*cos_theta1)+Pc+( K1 ^ 2 )*Psc);          // Efficiency at unity factor and full load ( beacuse taken k1 = 1 ) in percentage
eta_12 = 100 * (K2*S*cos_theta1)/((K2*S*cos_theta1)+
```

```
Pc+( K2 ^ 2 )*Psc);            // Efficiency at unity factor and half load ( beacuse taken k2 = 0.5 ) in percentage

// case (b) for 0.8 power factor lagging

cos_theta2 = 0.8;                                          // 0.8 power factor lagging
eta_21 = 100 * (K1*S*cos_theta2)/((K1*S*cos_theta2)+Pc+( K1 ^ 2 )*Psc);            // Efficiency at 0.8 power factor lagging and full load ( beacuse taken k1 = 1 ) in percentage
eta_22 = 100 * (K2*S*cos_theta2)/((K2*S*cos_theta2)+Pc+( K2 ^ 2 )*Psc);            // Efficiency at 0.8 power factor lagging and half load ( beacuse taken k2 = 0.5 ) in percentage

// Case (c) for 0.8 poer factor leading

eta_31 = eta_21;                                   // Efficiency at 0.8 power factor leading and full load will be same as the Efficiency at 0.8 power factor lagging and full load in percentage
eta_32 = eta_22;                                   // Efficiency at 0.8 power factor leading and half load will be same as the Efficiency at 0.8 power factor lagging and half load in percentage

// Case (d) Maximum Efficiency assumed that unity power factor
// Psc = Pc At Maximum Efficiency

eta_41 = 100 * (K1*S*cos_theta1)/((K1*S*cos_theta1)+Pc+Pc);            // Maximum Efficiency at unity factor and full load ( beacuse taken k1 = 1 ) in percentage

```

```
// Case (e) Maximum Efficiency assumed that 0.8 power factor lagging
// Psc = Pc At Maximum Efficiency

eta_51 = 100 * (K1*S*cos_theta2)/((K1*S*cos_theta2)+Pc+Pc);           // Maximum Efficiency at unity factor and full load ( beacuse taken k1 = 1 ) in percentage

// Case (f) Maximum Efficiency assumed that 0.8 power factor leading
// Psc = Pc At Maximum Efficiency

eta_61 = eta_51;

    // Maximum Efficiency at 0.8 power factor leading and full load will be same as the Maximum Efficiency at 0.8 power factor lagging and full load in percentage
out1 = K1*S*cos_theta1;                                         // Output at which maximum efficiency occurs at unity power factor at full load in Watts
out2 = K1*S*cos_theta2;                                         // Output at which maximum efficiency occurs at 0.8 power factor lagging at full load in Watts
out3 = K1*S*cos_theta2;                                         // Output at which maximum efficiency occurs at unity power factor leading at full load in Watts

// DISPLAY RESULTS

disp("EXAMPLE : 3.14 : SOLUTION :-") ;
printf("\n (a.1) Efficiency at unity power factor and full load , eta = %.2f Percent \n ",eta_11);
printf("\n (a.2) Efficiency at unity power factor
```

```
and half load , eta= % .2f Percent \n",eta_12);
printf("\n (b.1) Efficiency at 0.8 power factor lagging and full load , eta = %.2f Percent \n ",eta_21);
printf("\n (b.2) Efficiency at 0.8 power factor lagging and half load , eta= % .2f Percent \n",eta_22);
printf("\n (c.1) Efficiency at 0.8 power factor leading and full load , eta = %.2f Percent \n ",eta_31);
printf("\n (c.2) Efficiency at 0.8 power factor leading and half load , eta= % .2f Percent \n",eta_32);
printf("\n (d) Maximum Efficiency at unity power factor and full load , eta = %.2f Percent \n ",eta_41);
printf("\n (e) Maximum Efficiency at 0.8 power factor lagging and full load , eta = %.2f Percent \n ",eta_51);
printf("\n (f) Maximum Efficiency at 0.8 power factor leading and full load , eta = %.2f Percent \n ",eta_61);
printf("\n (g) Output at which maximum efficiency occurs at unity power factor at full load = %.f W \n ",out1);
printf("\n (h) Output at which maximum efficiency occurs at 0.8 power factor lagging at full load = %.f W \n ",out2);
printf("\n (i) Output at which maximum efficiency occurs at 0.8 power factor leading at full load = %.f W \n ",out3);
printf("\n IN THE ABOVE PROBLEM MAXIMUM EFFICIENCY AND THE OUTPUT AT WHICH THE MAXIMUM EFFICIENCY OCCURS IS NOT CALCULATED IN THE TEXT BOOK \n")
```

Scilab code Exa 3.15 To find all day efficiency

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.15

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA
// Refer figure 3.17 page no. 101

S = 500 * 10 ^ 6;                        // Rating of
    power transformer in VA
V1 = 400 * 10^3;                         // HV side
    rating of the power transformer in Volts
V2 = 131 * 10^3;                         // LV side
    rating of the power transformer in Volts
pcu = 5;                                 // Rated Copper
    loss in Percentage
pi = 1;                                  // Rated Core
    loss in Percentage


// CALCULATIONS

Pcu = S*(pcu/100);                                   //
    Rated Copper loss in Watts
Pi = S*(pi/100);                                     //
    Rated Core loss in Watts
kt = 0.25*3 + 0.75*3 + 1*3 + 0.5*3 + 1.0*3 + 0.25*6
```

```
+ 1.0*3;                // From graph figure 3.17 page no. 101
out = S*kt;                                        // Output energy in kilo-watt-hour
kt2 = 0.54375;                                     // From graph figure 3.17 page no. 101
eloss = 24*Pi + S*kt2;                             // Energy required in losses in kilo-watt-hour {Energy required in losses = 24*Pi + sigma(copper loss * duration)}
eta = 100*(out/(out+eloss));                       // All day efficiency


// DISPLAY RESULTS

disp("EXAMPLE : 3.15: SOLUTION :-");
printf("\n (a) All day efficiency = %.2f percent \n",eta)
```

Scilab code Exa 3.16 To find percentage regulation at full load UPF and at 60 percentage lagging and leading pf

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.16

clear ; clc ; close ; // Clear the work space and console
```

```


// GIVEN DATA

S = 20 * 10 ^ 3;                        // Rating of the Step−down Transformer in VA
f = 50;                                 // Frequency in Hertz
V = 200;                                // Normally supplied Voltage of Step−down Transformer in Volts
Vsc = 100;                              // Potential difference when Secondary being Short− Circuited in Volts
Isc = 10;                               // Primary Current when Secondary being Short− Circuited in Amphere
Cos_theta_sc = 0.28;                    // Power factor when Secondary being Short− Circuited


// CALCULATIONS

I = S/V;                                        // Rated primary current in Amphere
Wsc = Vsc * Isc * Cos_theta_sc;                 // Power loss when Secondary being Short− Circuited in Watts
R = Wsc/(Isc ^ 2);                              // Resistance of Transformer referred to primary side in Ohms
Z = Vsc/Isc;                                    // Referred Impedence in Ohms
X = sqrt((Z^2)-(R^2));                          // Leakage Reactance referred to primary side in Ohms
Er = (I*R)/V;                                   // Per unit Resistance in Ohms
```

```
Ex = (I*X)/V;                                   // Per unit Reactance in Ohms
Cos_theta1 = 1.0;                               // Unity Power factor
Cos_theta2 = 0.6;                               // 0.6 Power factor Lagging
Cos_theta3 = 0.6;                               // 0.6 Power factor Leading
Sin_theta1 = 0.0;                               // Unity Power factor
Sin_theta2 = 0.8;                               // 0.6 Power factor Lagging
Sin_theta3 = 0.8;                               // 0.6 Power factor Leading
E1 = (Er*Cos_theta1)+(Ex*Sin_theta1);           // pu Regulation at Unity Power factor
E2 = (Er*Cos_theta2)+(Ex*Sin_theta2);           // pu Regulation at 0.6 Power factor Lagging
E3 = (Er*Cos_theta3)-(Ex*Sin_theta3);           // pu Regulation at 0.6 Power factor Leading


// DISPLAY RESULTS

disp("EXAMPLE : 3.16 : SOLUTION :-") ;
printf("\n (a) pu Regulation at Unity Power factor , E = %.1f \n ",E1);
printf("\n (b) pu Regulation at 0.6 Power factor Lagging , E= % .2f \n",E2);
printf("\n (c) pu Regulation at 0.6 Power factor Leading , E= % .2f \n",E3);
```

Scilab code Exa 3.17 To find load pf at which secondary terminal voltage will be minimum

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.17

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

S = 500;                                  // Rating of the
     3-Phase transformer in kVA
V1 = 11 * 10 ^ 3;                         // Votage rating
     of the 3-Phase transformer on HV side in Volts
V2 = 400;                                 // Votage rating
     of the 3-Phase transformer on LV side in Volts
f = 60;                                   // Frequencyin
    Hertz
eta = 98;                                 // Maximum
    Efficency of the Transformer in Percentage
    Operating at 80% full load and Unity Power factor
K = 0.8;                                  // Beacuse 80%
    Full load
x = 1.0;                                  // Unity Power
    factor
Ex = 4.5;                                 // Percentage
    impedance


// CALCULATIONS

Out = S * K * x;                            // Output in
```

```
KiloWatts at 80% full load and Unity Power factor
Inp = Out/(eta/100);                    // Input in KiloWatts at full load and Unity Power factor
Total_loss = Inp - Out;                 // Total loss at full load  in KiloWatts
Cu_loss = Total_loss/2;                 // Copper loss  in KiloWatts at 80% full load and Unity Power factor
Pcu = Cu_loss/(K ^2 );                  // Full load Copper loss in KiloWatts
Er = Pcu/S;                             // Per unit Resistance
theta = atand((Ex/100)/Er);             // Power factor angle at secondary terminal voltage is minimum in Degrees
Pf = cosd(theta);                       // Load power factor for minimum volatge of the secondary terminal


// DISPLAY RESULTS

disp("EXAMPLE : 3.17 : SOLUTION :-") ;
printf("\n (a) Load power factor for minimum volatge  of the secondary terminal , cos(theta) = %.4f lagging \n ",Pf);
```

Scilab code Exa 3.19 To find how will sharing of the two transformers if the total load of 400kW at UPF

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
```

```
// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.19

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

Sa = 200;                    // Rating of the TWO 1-Phase
     Transformer in kVA
Z1 = 0.005 + 0.08 * %i   // Equivalent Impedance of
    the Transformer-1 in Per-Unit
Z2 = 0.0075 + 0.04 * %i // Equivalent Impedance of
    the Transformer-2 in Per-Unit
P = 400;                     // Total load in kiloWatts
Cos_theta = 1.0;             // Unity power factor


// CALCULATIONS

kVA = P/Cos_theta;            // kVA rating of the
    Transformer
S = kVA;                      // kVA rating of the
    Transformer
S1 = ( Z2/(Z1+Z2) )*S;      // Load shared by
    Transformer-1 in kVA
S2 = S - S1;                  // Load shared by
    Transformer-2 in kVA


// DISPLAY RESULTS

disp("EXAMPLE : 3.19 : SOLUTION :-") ;
printf("\n (a) Load shared by Transformer-1 , S1 = %
    .2f+j(%.2f) kVA \n ",real(S1),imag(S1));
```

```
printf("\n (b )Load shared by Transformer-2 , S2 = %
   .2 f+j% .2 f kVA \n ",real(S2),imag(S2));
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED
   WRONGLY ( I verified by manual calculation )]\n"
   );
printf("\n      WRONGLY PRINTED ANSWERS ARE :-  (a)
   S1 = (-131.90)+j(38.47)kVA instead of %.2 f+j(%.2 f
   ) kVA \n ",real(S1),imag(S1));
printf("\n                                          (b)
   S2 = (268.1)+j(.38047)kVA  instead of %.2 f+j%.2 f
   kVA \n ",real(S2),imag(S2));
```

Scilab code Exa 3.20 To find current flowing in the primary winding

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.20

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

V1 = 110;                   // Primary voltage of the
   Two Transformers the two primaries are connected
   in parallel in Volts
I1 = 2.0;                   // Primary Current in
   Amphere
```

```
P1 = 40;                      // Primary power intake in Watts
V2 = 28;                      // secondary voltage of the Two Transformers the two secondary are connected in phase opposition in Volts
I2 = 6.8;                     // secondary Current in Amphere
P2 = 180;                     // secondary power intake in Watts
a = 110/220;                  // Turn ratio of the Transformer


// CALCULATIONS

theta_o = acosd((a*P1)/(a*I1*V1));                          // Primary Power factor angle in Degrees
Io = 1.0 * (cosd(theta_o)-sind(theta_o)* %i);                       // No-load current in individual transformer in Amphere
theta_sc = acosd((a*P2)/(a*I2*V2));                        // Secondary Power factor angle in Degrees
i_sc = I2 * ( cosd(theta_sc)-sind(theta_sc)* %i);          // Secondary current in Amphere
I_sc = (1/a)*i_sc;                                                   // referred Secondary current in each of the primary side in Amphere
It1 = Io + I_sc;                                                          // LT winding current in the 1st Transformer in Amphere
It2 = Io - I_sc;                                                          // LT winding current in the 2nd Transformer in Amphere
In1 = It1 + It2;                                                          // The
```

```
    current flowing on paralel connected LT winding (
    This is same as total no-load current in the two
    Transformer) in Amphere


// DISPLAY RESULTS

disp("EXAMPLE : 3.20 : SOLUTION :-") ;
printf("\n (a) LT ( Primary ) winding current in the
     1st Transformer , It1 = %.3f+j(%.4f) A \n ",real
    (It1),imag(It1));
printf("\n (b) LT ( Primary ) winding current in the
     2nd Transformer , It2= %.3f+j%.5f A \n",real(It2
    ),imag(It2));
printf("\n (c) LT winding are connected in parallel ,
     the current flowing on paralel connected LT
    winding , In1 = %.3f+j(%.5f) A \n",real(In1),imag(
    In1));
```

Scilab code Exa 3.21 To find calculate the kVA rating and currents of autotransformers as shown in figure 3 31b and 3 31c

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.21

clear ; clc ; close ; // Clear the work space and
    console

```

```

// GIVEN DATA
// Refer figures 3.31(a), 3.31(b) and 3.31(c):- Page no. 121

VaH = 220;                    // HV side Voltage of the two winding Transformer in volts for case(a)
VaL = 110;                    // LV side Voltage of the two winding Transformer in volts for case(a)
VbH = 330;                    // HV side Voltage of the two winding Transformer in volts for case(b)
VbL = 220;                     // LV side Voltage of the two winding Transformer in volts for case(b)
VcH = 330;                     // HV side Voltage of the two winding Transformer in volts for case(c)
VcL = 110;                     // LV side Voltage of the two winding Transformer in volts for case(c)
S = 1.5;                       // Ratings of the the two winding Transformer in kVA
I1 = 6.8;                      // Rated current in HV side in Amphere
I2 = 13.6;                     // Rated current in LV side in Amphere


// CALCULATIONS
// for case(a):- figure 3.31(b) page no. 121

IbH = I2;                         // Current of Auto-Transformer in HV side in Amphere
IbL = I1 + I2;                    // Current of Auto-Transformer in LV side in Amphere
KVA_b_L = (VbL*IbL)/1000;         // LV side kVA rating of the Auto-Transformer in kVA
KVA_b_H = (VbH*IbH)/1000;         // HV side kVA rating of the Auto-Transformer in kVA

// for case(b):- figure 3.31(c) page no. 121
```

```

IcH = I1;                          // Current of Auto-Transformer in HV side in Amphere
IcL = I1 + I2;                     // Current of Auto-Transformer in LV side in Amphere
KVA_c_L = (VcL*IcL)/1000;          // LV side kVA rating of the Auto-Transformer in kVA
KVA_c_H = (VcH*IcH)/1000;          // HV side kVA rating of the Auto-Transformer in kVA


// DISPLAY RESULTS

disp("EXAMPLE : 3.21 : SOLUTION :-") ;
printf("\n (a.1) Current of Auto-Transformer in HV side for case (b) , IH = %.1f A \n ",IbH);
printf("\n       Current of Auto-Transformer in LV side for case (b), IL= % .1f A \n",IbL);
printf("\n (a.2) LV side kVA rating of the Auto-Transformer for case (b), KVAL = % .3f kVA \n ",KVA_b_L);
printf("\n       HV side kVA rating of the Auto-Transformer for case (b), KVAH= % .3f kVA \n",KVA_b_H);
printf("\n (b.1) Current of Auto-Transformer in HV side for case (c) , IH = %.1f A \n ",IcH);
printf("\n       Current of Auto-Transformer in LV side for case (c) , IL= % .1f A \n",IcL);
printf("\n (b.2) LV side kVA rating of the Auto-Transformer for case (c), KVAL = % .3f kVA \n ",KVA_c_L);
printf("\n       HV side kVA rating of the Auto-Transformer for case (c) , KVAH= % .3f kVA \n",KVA_c_H);
```

Scilab code Exa 3.22 To find current supplied by the common winding kVA rating of the auto transformer and voltage on secondary if common winding are open

```
// ELECTRICAL MACHINES
// R.K. Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.22

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

S = 10 * 10 ^ 3;                         // Rating of the
    Two-winding Transformer in VA
V1 = 2000;                               // HV side voltage
     of the Two-winding Transformer in Volts
V2 = 200;                                // LV side voltage
     of the Two-winding Transformer in Volts
V_A_H = 2200;                            // Two-winding
    Transformer is connected to auto transformer HV
    side in Volts
V_A_L = 200;                             // Two-winding
    Transformer is connected to auto transformer LV
    side in Volts
f = 50;                                  // Frequency in
    Hertz


// CALCULATIONS
// for finding (a)
```

```
I2 = S/V2;                              // Rated LV side current of winding for Step-up Auto transformer in Amphere
I1 = S/V1;                              // Rated HV side current of winding for Step-up Auto transformer in Amphere
IaH = I2;                               // The HV side current in the Auto-Transformer for Full-load in Amphere
IaL = I2 + I1 ;                         // The LV side current in the Auto-Transformer for Full-load in Amphere
VL = V1;                                // LV side voltage in Volts
VH = V1 + V2;                           // HV side voltage in Volts
KVA_a_L = (VL*IaL)/1000;                // kVA rating of LV SIDE
KVA_a_H = (VH*IaH)/1000;                // kVA rating of HV SIDE

// For finding (b)

IbH = I1;                               // HV side Rated current through the Auto-Transformer in Amphere
IbL = I1 + I2;                          // LV side Rated current through the Auto-Transformer in Amphere
KVA_b_L = (V_A_L*IbL)/1000;             // kVA rating of LV SIDE as output Auto-Transformer
KVA_b_H = (V_A_H*IbH)/1000;             // kVA rating of HV SIDE as output Auto-Transformer

// case (c)

V = V1;                                 // Voltage on the Secondary, if the Commom windings are open

```

```

// DISPLAY RESULTS

disp("EXAMPLE : 3.22 : SOLUTION :-") ;
printf("\n (a.1) HV side Curent supplied by the common windings , IH = %.f A \n ",IaH);
printf("\n (a.2) LV side Curent supplied by the common windings , IL= %.f A \n",IaL);
printf("\n (b.1) KVA rating of LV SIDE as output Auto-Transformer , KVAL = %.f kVA \n ",KVA_b_L);
printf("\n (b.2) KVA rating of HV SIDE as output Auto-Transformer , KVAH= %.f kVA \n",KVA_b_H);
printf("\n (c) Voltage on the Secondary , if the Commom windings are open , V = %.f V \n ",V);
```

Scilab code Exa 3.23 To find current and phase voltages and the voltage across the neutral assuming ideal transformer

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.23

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

S = 100;                                  // Rating
```

```
of the 3-Phase Transformer in kVA
VH = 11;                                            // HV side voltage in kilo-Volts
VL = 440;                                           // LV side voltage in Volts
Vl = 400;                                           // Line voltage in Volts
ZA = 0.6;                                           // Line impedance in line A in Ohms
ZB = 0.6*(0.8 + 0.6 * %i);                          // Line impedance in line B in Ohms
ZC = 0.6*(0.5 - 0.866 * %i);                        // Line impedance in line C in Ohms


// CALCULATIONS

Vp = Vl/sqrt(3);
                                                    // Phase voltage in Volts
VAB = Vl * exp( %i * 0 * %pi/180);                  // Line Voltage across line A and B in Volts
VBC = Vl * exp( %i * (-120) * %pi/180);             // Line Voltage across line B and C in Volts
VCA = Vl * exp( %i * 120 * %pi/180);                // Line Voltage across line C and A in Volts
VAN = (Vl/sqrt(3)) * exp( %i * (-30) * %pi/180);    // Phase Voltage across line A and Neutral in Volts
VBN = (Vl/sqrt(3)) * exp( %i * (-150) * %pi/180);   // Phase Voltage across line B and Neutral in Volts
VCN = (Vl/sqrt(3)) * exp( %i * (90) * %pi/180);     // Phase Voltage across line C and Neutral in Volts
```

```
IA = VAN/ZA;                                     // Line current in line A in Amphere
IB = VBN/ZB;                                     // Line current in line B in Amphere
IC = VCN/ZC;                                     // Line current in line C in Amphere
IN = IA + IB + IC ;                              // Current in the Neutral in Amphere
Y = (1/ZA)+(1/ZB)+(1/ZC);                        // Net Admittance in mho
VN = IN/Y;                                       // Neutral Potential in Volts
VDA = VAN - VN;                                  // Voltage drops across the ZA in Volts
VDB = VBN - VN;                                  // Voltage drops across the ZB in Volts
VDC = VCN - VN;                                  // Voltage drops across the ZC in Volts


// DISPLAY RESULTS

disp("EXAMPLE : 3.23 : SOLUTION :-") ;
printf("\n (a.1) Line current in line A , IA = %.f<%.f A \n ",abs(IA),atand(imag(IA),real(IA)));
printf("\n (a.2) Line current in line B , IB = %.f<%.2f A \n",abs(IB),atand(imag(IB),real(IB)));
printf("\n (a.3) Line current in line C , IC = %.f<%.f A \n ",abs(IC),atand(imag(IC),real(IC)));
```

```
printf("\n (b.1) Phase Voltage across line A and Neutral , VAN = %.f<%.f V \n",abs(VAN),atand(imag(VAN),real(VAN)));
printf("\n (b.2) Phase Voltage across line B and Neutral , VBN = %.f<%.f V \n ",abs(VBN),atand(imag(VBN),real(VBN)));
printf("\n (b.3) Phase Voltage across line C and Neutral , VCN = %.f<%.f V \n",abs(VCN),atand(imag(VCN),real(VCN)));
printf("\n  (c)   Neutral Potential , VN = %.1f<%.2f V \n ",abs(VN),atand(imag(VN),real(VN)));
printf("\n\n     [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n       WRONGLY PRINTED ANSWERS ARE :- (a) IC = 385<-90.1 V instead of %.f<%.f A \n ",abs(IC),atand(imag(IC),real(IC)));
printf("\n                                    (b) VN = 230.5<78.17 V  instead of %.1f<%.2f V \n ",abs(VN),atand(imag(VN),real(VN)) );
printf("\n From Calculation of the IC, rest all the Calculated values in the TEXT BOOK is WRONG because of the IC value is WRONGLY calculated and the same used for the further Calculation part \n")
```

Scilab code Exa 3.24 To find secondary line voltage and current primary and secondary phase current and the output for star star connection star delta connection delta delta connection and delta star connection

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
```

```
// CHAPTER : 3 : TRANSFORMERS

// EXAMPLE : 3.24

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

VL = 11000;                          // Line-line voltage of the 3 identical 1-phase Transformer in Volts
IL = 10;                             // Line current of the 3 identical 1-phase Transformer in Amphere
a = 10;                              // Ratio of trun per phase of the 3 identical 1-phase Transformer


// CALCULATIONS
// For case (a) STAR-STAR

VPp_a = VL/sqrt(3);                  // Primary phase volatge in Volts
IPp_a = IL;                          // Primary phase current in Amphere
VSp_a = VPp_a/a;                     // Secondary phase voltage in Volts
ISp_a = a*IPp_a;                     // Secondary phase current in Amphere
ISl_a = ISp_a;                       // Secondary line current in Amphere
VSl_a = VSp_a*sqrt(3);               // Secondary line voltage in Volts
Out_a = sqrt(3)*VSl_a*ISl_a/1000;    // Output in kVA

// For case (b) STAR-DELTA

```

```
VPp_b = VL/sqrt(3);                // Primary phase volatge in Volts
IPp_b = IL;                        // Primary phase current in Amphere
VSp_b = VPp_a/a;                   // Secondary phase voltage in Volts
ISp_b = a*IPp_b;                   // Secondary phase current in Amphere
ISl_b = sqrt(3)*ISp_b;             // Secondary line current in Amphere
VSl_b = VSp_b;                     // Secondary line voltage in Volts
Out_b = sqrt(3)*VSl_b*ISl_b/1000;  // Output in kVA

// For case (c) DELTA-DELTA

VPp_c = VL;                        // Primary phase volatge in Volts
IPp_c = IL/sqrt(3);                // Primary phase current in Amphere
VSp_c = VPp_c/a;                   // Secondary phase voltage in Volts
ISp_c = a*IPp_c;                   // Secondary phase current in Amphere
ISl_c = sqrt(3)*ISp_c;             // Secondary line current in Amphere
VSl_c = VSp_c;                     // Secondary line voltage in Volts
Out_c = sqrt(3)*VSl_c*ISl_c/1000;  // Output in kVA

// For case (d) DALTA-STAR

VPp_d = VL;                        // Primary phase volatge in Volts
IPp_d = IL/sqrt(3);                // Primary phase current in Amphere
VSp_d = VPp_d/a;                   // Secondary phase voltage in Volts
```

```
ISp_d = a*IPp_d;                     // Secondary phase current in Amphere
ISl_d = ISp_d;                       // Secondary line current in Amphere
VSl_d = sqrt(3)*VSp_d;               // Secondary line voltage in Volts
Out_d = sqrt(3)*VSl_d*ISl_d/1000;  //Output in kVA


// DISPLAY RESULTS

disp("EXAMPLE : 3.24 : SOLUTION :-") ;
printf("\n For STAR-STAR Connection \n\n (a.1) Secondary line voltage = %.f V \n ",VSl_a);
printf("\n (a.2) Secondary line current = % .f A \n",ISl_a);
printf("\n (a.3) Primary phase current = %.f A \n ",IPp_a);
printf("\n (a.4) Secondary phase current = %.f A \n",ISp_a);
printf("\n (a.5) Output = %.2f kVA \n ",Out_a);
printf("\n For STAR-DELTA Connection \n\n  (b.1) Secondary line voltage = % .f V \n",VSl_b);
printf("\n (b.2) Secondary line current = %.f A \n ",ISl_b);
printf("\n (b.3) Primary phase current = %.f A \n",IPp_b);
printf("\n (b.4) Secondary phase current = %.f A \n ",ISp_b);
printf("\n (b.5) Output = % .2f kVA \n",Out_b);
printf("\n For DELTA-DELTA Connection \n\n (c.1) Secondary line voltage = %.f V \n ",VSl_c);
printf("\n (c.2) Secondary line current = %.2f A \n",ISl_c);
printf("\n (c.3) Primary phase current = %.2f A \n ",IPp_c);
printf("\n (c.4) Secondary phase current = %.1f A \n ",ISp_c);
```

```
printf("\n (c.5) Output = %.1f kVA \n ",Out_c);
printf("\n For DELTA-STAR Connection \n\n (d.1) Secondary line voltage = % .2f V \n",VSl_d);
printf("\n (d.2) Secondary line current = %.1f A \n ",ISl_d);
printf("\n (d.3) Primary phase current = %.2f A \n",IPp_d);
printf("\n (d.4) Secondary phase current  = %.1f A \n ",ISp_d);
printf("\n (d.5) Output = % .1f kVA \n",Out_d);
```

Chapter 4

Direct Current Machines

Scilab code Exa 4.1 To find maximum induced EMF in the armature conductor

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.1

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

N = 600;                      // Speed of the driven
   Machine in RPM
D = 2;                        // Diameter of the
   Machine in Meter
```

```
L = 0.3;                     // Length of the Machine in Meter
Bm = 1.0;                    // Flux Density in Weber per Meter-Square


// CALCULATIONS

n = N/60;                     // Revolution per second
v = %pi * D * n;              // Peripheral velocity in Meter per second
E = Bm * v * L;               // Maximum EMF induced in the Conducter in Volts


// DISPLAY RESULTS

disp("EXAMPLE : 4.1 : SOLUTION :-") ;
printf("\n (a) Maximum EMF induced in the Conducter , E = %.3f V \n ",E);
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n      WRONGLY PRINTED ANSWERS ARE :- (a) Induced EMF, E = 2.826 V instead of %.3f A \n ",E);
printf("\n From Calculation of the peripheral velocity(v), rest all the Calculated values in the TEXT BOOK is WRONG because of the peripheral velocity(v) value is WRONGLY calculated and the same used for the further Calculation part \n")
```

Scilab code Exa 4.2 To find EMF induced

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.2

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

p = 8;                              // Number of the poles in Dc machine
a = 8;                              // Number of the Parallel path
N = 500;                            // Rotation per minute in RPM
phi = 0.095;                        // Average flux in air gap in Weber per meter
Za = 1000;                          // Total number of the Conductor in Armature


// CALCUALTIONS

n = N/60;                           // Rotation (Revolution) per Second
E = (p/a)*n*phi*Za;                 // EMF induced in Volts


// DISPLAY RESULTS

disp("EXAMPLE : 4.2 : SOLUTION :-") ;
```

```
printf("\n (a) EMF induced , E = %.1f A \n ",E);
```

Scilab code Exa 4.3 To find the number of conductors in parallel path

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.3

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

E = 420;                    // EMF induced in Volts
N = 900;                    // Rotation speed in RPM
phi = 0.06;                 // Flux per pole in Weber
    per pole
Two_p = 4;                  // Total number of poles


// CALCULATIONS

n = N/60;                               // Revolution Per
    second
Zc = E/(Two_p*phi*n);                   // Number of the
    Conductor in Parallel Path


```

```
// DISPLAY RESULTS

disp("EXAMPLE : 4.3 : SOLUTION :-") ;
printf("\n (a) Number of the Conductor in Parallel
   Path , Zc = %.2f Conductors nearly 117 conductors
    \n ",Zc);
```

Scilab code Exa 4.4 To find torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.4

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

L = 0.3;                // Length of the Machine in
   Meter
Ia = 10;                // Current through The
   Conductors in Amperes
N = 10;                 // Number of the Conductors in
   each Slot
Za = 24;                // Number of the Slots
Bav = 0.6;              // Average Flux Density in Telsa
D = 0.1;                // Machine Daimeter in Meter

```

```

// CALCULATIONS

F = N*Ia*Bav*L;                    // Force due to the Single Slot in Newton
T = (Bav*L*Ia*N*D*Za)/2            // Torque produced in the Machine in Newton-Meter


// DISPLAY RESULTS

disp("EXAMPLE : 4.4 : SOLUTION :-") ;
printf("\n (a) Torque produced in the Machine , T = %.1f N-m \n",T);
```

Scilab code Exa 4.5 To find useful flux per pole when the armature is lap and wave connected

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.5

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

p = 4;                  // Number of the Poles in the DC
```

```
machine
Nt = 100;             // Number of the turns in the Dc machine
N = 600;              // Rotation speed of the DC machine in RPM
E = 220;              // EMF generated in open circuit in Volts
Z = 200;              // Total number of the Conductor in armature


// CALCUALTIONS
// For case (a) Lap Connected

a = 4;                              // Number of the Poles in the DC machine
n = N/60;                           // Revolution per second
phi_a = (E*a)/(p*Z*n);              // Useful flux per pole when Armature is Lap connected in Weber

// For case (b) Wave Connected

a = 2;                              // Number of the Poles in the DC machine
phi_b = (E*a)/(p*Z*n);              // Useful flux per pole when Armature is Wave connected in Weber


// DISPLAY RESULTS

disp("EXAMPLE : 4.5 : SOLUTION :-") ;
printf("\n (a) Useful flux per pole when Armature is Lap connected , phi = %.1f Wb \n ",phi_a);
printf("\n (B) Useful flux per pole when Armature is Lap connected , phi = %.3f Wb \n ",phi_b);
```

Scilab code Exa 4.6 To find Torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.6

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

p = 6;                          // Number of the pole in DC Motor
Ia = 20;                        // Armature Current in Amphere
Z = 1000;                       // Number of the Conductors
a = 6;                          // Number of the Parallel paths
phi = 25 * 10 ^ -3;             // Flux per pole in Weber


// CALCULATIONS

T = (p/a)*((Z*Ia*phi)/(2*%pi));         // Deleloped Torque in Newton-Meter
```

```
// DISPLAY RESULTS

disp("EXAMPLE : 4.6 : SOLUTION :-") ;
printf("\n (a) Developed Torque in an Six-pole DC
    Motor , T = %.1f N-m \n ",T);
```

Scilab code Exa 4.7 To find reactance voltage

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.7

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

p = 2;                  // Number of the Pole
N = 1000;               // Rotation speed of the
    Armature in RPM
Ia = 20;                // Armature Current in
    Amphere
CS = 36;                // Commutator Segments
BW = 1.4;               // Brush width is 1.4 times
    of the Commutator Segments
L = 0.09 * 10 ^ -3;     // Inducatnce of the each
```

```
Armature Coil


// CALULATIONS

a = p;                      // Number of the Parallel paths (Equal to number of poles because Lap Connected Armature)
n = N/60;                   // Revoultion per second
I = Ia/2;                   // Current Through the each Conductor in Amphere
v = n * CS;                 // Peripheral Velocity of Commutator in Commutator segments per Seconds
Tc = BW/v;                  // Time of the Commutation in Seconds
Er = (L*2*I)/Tc;            // Reactance voltage in Volts


// DISPLAY RESULTS

disp("EXAMPLE : 4.7 : SOLUTION :-") ;
printf("\n (a) Reactance voltage assuming Linear Commutation , Er = %.4f V \n",Er);
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n      WRONGLY PRINTED ANSWERS ARE :- (a) Tc = 0.014 s instead of %.4f s \n",Tc);
printf("\n                                      (b) Er = 1.2857 V instead of %.4f V\n",Er);
printf("\n From Calculation of the Time of commutation (Tc), rest all the Calculated values in the TEXT BOOK is WRONG because of the Time of commutation (Tc) value is WRONGLY calculated and the same used for the further Calculation part \n")
```

Scilab code Exa 4.8 To find approximate time of commutation

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.8

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

N = 800;                    // Rotation speed of the
    Commutator in RPM
D = 50;                     // Diameter in Centimeter
BW = 1.5;                   // Brush Width in Centimeter


// CALCULATIONS

r = D/2;                        // Radius in Centimeter
n = N/60;                       // Revoultion per second
w = (2 * %pi)*n;                // Angular velocity
v = w*r;                        // Peripheral Speed in
    centimeter per second
Tc = (BW/v)*1000;                      // Time of the
    Commutation in Second

```

```
// DISPLAY RESULTS

disp("EXAMPLE : 4.8 : SOLUTION :-") ;
printf("\n (a) Time of the Commutation , Tc = %.4f
   ms \n",Tc);
```

Scilab code Exa 4.9 To find Ampere turn per pole the demagnetizing armature turn per pole cross magnetizing armature turn per pole and ampere turn of compensating winding when lap and wave conncted

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.9

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

p = 4;                          // Number of the pole in
     the Generator
Ia = 100;                       //  supplying Current by
     the Generator in Amphere
Za = 500;                       // Armature conductor
beta = 8;                       // Brush shift in
   degrees
If = 5;                         // Current in the
```

```
Seperately exicted field winding
ratio = 0.7;                    // Ratio of Pole arc to Pole pitch


// CALCULATIONS
// For case (a) Lap winding

a_a = p;                                                    // Number of the Parallel Paths
AT_a = (Za*Ia)/(2*a_a*p);                                   // Amphere turns per pole
ATd_a = (beta*Za*Ia)/(360*a_a);                             // Demagnetizing Armature Amphere turns per pole
ATc_a = ((1/p)-(beta/180))*((Za*Ia)/(2*a_a));     // CrossMagnetizing Armature Amphere turns per pole
ATw_a = ratio*AT_a;                                       // Amphere turns of Compensating winding

// For case (b) Wave winding

a_b = p/2;                                                  // Number of the Parallel Paths
AT_b = (Za*Ia)/(2*a_b*p);                                   // Amphere turns per pole
ATd_b = (beta*Za*Ia)/(360*a_b);                             // Demagnetizing Armature Amphere turns per pole
ATc_b = ((1/p)-(beta/180))*((Za*Ia)/(2*a_b));     // CrossMagnetizing Armature Amphere turns per pole
ATw_b = ratio*AT_b;                                         // Amphere turns of Compensating winding


// DISPLAY RESULTS

disp("EXAMPLE : 4.9 : SOLUTION :-") ;
printf("\n For LAP winding \n\n (a.1) Amphere turns per pole , AT = %.1f AT \n",AT_a);
```

```
printf("\n (a.2) Demagnetizing Armature Amphere
    turns per pole, ATd = %.1f AT \n",ATd_a);
printf("\n (a.3) Cross-Magnetizing Armature Amphere
    turns per pole, ATc = %.1f AT \n",ATc_a);
printf("\n (a.4) Amphere turns of Compensating
    winding, ATw = %.1f AT \n",ATw_a);
printf("\n For WAVE winding \n\n (b.1) Amphere turns
     per pole, AT = %.f AT \n",AT_b);
printf("\n (b.2) Demagnetizing Armature Amphere
    turns per pole, ATd = %.2f AT \n",ATd_b);
printf("\n (b.3) Cross-Magnetizing Armature Amphere
    turns per pole, ATc = %.f AT \n",ATc_b);
printf("\n (b.4) Amphere turns of Compensating
    winding, ATw = %.1f AT \n",ATw_b);
```

Scilab code Exa 4.10 To find the number of turns on each interpole

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.10

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

p = 6;                          // Number of the Poles
P = 100 * 10 ^ 3;               // Power rating of the
```

```
    DC machine in KiloWatts
V = 440;                          // Voltage rating of the
     DC machine in Volts
Z = 500;                          // Total number of the
    Armature Conductor
Ig = 1.0 * 10 ^ -2;               // Interpolar Air gap in
     Meter
Bi = 0.28;                        // Interpolar Flux
    Densist in Weber per Meter-Square
mue_0 = 4*%pi*10^ -7;             // Permeability of the
    air in Henry/Meter


// CALCULATIONS

Ia = P/V;                                       // Full
    load current in Amphere
a = p;                                          // Number
    of the Parallel path (Equal to p because LAP
    WINDING)
ATi = (Z*Ia)/(2*a*p)+((Bi*Ig)/mue_0);          // Amphere
     turns for each Interpole
Nc = ATi/Ia;                                    // Number
    of turns per pole of interpole


// DISPLAY RESULTS

disp("EXAMPLE : 4.10 : SOLUTION :-") ;
printf("\n (a) Amphere turns for each Interpole , ATi
     = %.2f AT \n",ATi);
printf("\n (b) Number of turns per pole of interpole
    , Nc = %.2f turns per pole nearly %.f turns per
    pole \n",Nc,Nc);
```

Scilab code Exa 4.11 To find voltage at the terminal of the machine and critical resistance

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.11

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

printf("\n EXAMPLE : 4.11 : \n\n            Given Data between the Field current and Open-Circuit EMF generated by DC shunt wound Generator \n");
printf("\n If(A)     0         1         2         3         4         5         6   \n");
printf("\n Voc(v)   10        90        170       217.5     251       272.5     281  \n");
N = 1000;                    // Speed of an DC Shunt wound generator on open circuit in RPM
Rf = 50;                     // Shunt field resistance in Ohms


// CALCULATIONS
// Refer Figure 4.20:- Page no. 180

Vt = 277.17;                             // Terminal Voltage in Volts from Figure 4.20 (The slope of the Resistance line Rf cuts the OCC at this Voltage [
```

```
   point])
Voc_r = 90;                          // Critical Open
   circuit voltage in Volts from Figure 4.20 page no
   . 180
If_r = 1.0;                          // Critical Field
   current in Amphere from Figure 4.20 page no. 180
Rc = Voc_r/If_r;                     // Crictical field
   Resistance in Ohms


// DISPLAY RESULTS

printf(" \n\n          SOLUTION :-\n") ;
printf("\n (a) Crictical field Resistance , Rc = %.f
   Ohms \n",Rc);
```

Scilab code Exa 4.12 To find critical resistance and terminal voltage

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.12

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

printf("\n EXAMPLE : 4.12 : \n\n             Given Data
```

```
 between the Field current and Open-Circuit EMF generated by DC Machine \n");
printf("\n If(A)    0       0.25      0.5       1.0       1.5       2.0       2.5      3.0 \n");
printf("\n Voc(v)   8        43        77       151       198       229       253      269\n");
N = 600;                        // Speed of an DC Shunt wound generator on open circuit in RPM
Rf1 = 100;                      // Shunt field resistance in Ohms
Rf2 = 125;                      // Shunt field resistance in Ohms


// CALCULATIONS
// Refer Figure 4.21:- Page no. 181

Vt1 = 253.33;                          // Terminal Voltage in Volts correspounding to field resistance of 100 Ohms from Figure4.21 Page no. 181 (The slope of the Resistance line Rf cuts the OCC at this Voltage [point])
Vt2 = 213.33;                          // Terminal Voltage in Volts correspounding to field resistance of 125 Ohms from Figure 4.21 Page no. 181 (The slope of the Resistance line Rf cuts the OCC at this Voltage [point])
Voc_r = 151;                           // Critical Open circuit voltage in Volts from Figure 4.20
If_r = 1.0;                            // Critical Field current in Amphere from Figure 4.20
Rc = Voc_r/If_r;                       // Crictical field Resistance in Ohms


// DISPLAY RESULTS

printf(" \n\n          SOLUTION :-\n") ;
```

```
printf("\n   (a)  Crictical field Resistance, Rc = %.
    f Ohms \n",Rc);
printf("\n (b.1) Terminal Voltage correspounding to
    field resistance of 100 Ohms is %.2f V \n ", Vt1)
    ;
printf("\n (b.1) Terminal Voltage correspounding to
    field resistance of 125 Ohms is %.2f V \n ", Vt2)
    ;
```

Scilab code Exa 4.13 To find load current

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.13

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

N1 = 1200;                // Rotation speed of the
    Separately excited Generator in RPM at case (1)
Ia1 = 100;                // Current supplied by the
    Generator in Amphere
V1 = 220;                 // Opearting Volatge of the
    Generator in Volts
Ra = 0.08;                // Armature Resistance in
    Ohms
```

```
N2 = 1000;                    // Rotation speed of the Separately excited Generator in RPM at case (2)
Vb = 2.0;                     // Total Brush drop in Volts


// CALCULATIONS

RL = V1/Ia1;                                    // Load resistance in Ohms
E1 = V1 + Vb + (Ra * Ia1);                      // Back EMF at case (1) in Volts
E2 = (N2/N1)*E1;                                //  Back EMF at case (2) in Volts (Excitation is Constant)
Ia2 = (E2 - Vb)/(RL + Ra);                      // New load current in Amphere for case (2)


// DISPLAY RESULTS

disp("EXAMPLE : 4.13 : SOLUTION :-") ;
printf("\n (a) New load current at %.f RPM , Ia2 = %.2f A \n",N2,Ia2);
```

Scilab code Exa 4.14 To find current and percentage change in speed of the machine

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.14
```

```
clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

I = 50;                              // Curent supplied by the Separately Excitated Generator in Amphere
V = 250;                             // Dc bus bar in Volts
phi_1 = 0.03;                        // Useful Flux in Weber
Ra = 0.5;                            // Armature Resistance in Ohms
phi_2 = 0.029;                       // New(Changed) Flux in Weber


// CALCULATIONS

Vd = I * Ra;                         // Voltage drop in the Armature in Volts
E1 = V + Vd;                         // EMF Generated in Volts
E2 = (phi_2/phi_1)*E1;               // EMF Generated in Volts immediately after flux changes but speed will remains same
Ia = (E2 - V)/Ra;                    // Armature Current in Amphere immediately after flux changes
perct = 100 * (( phi_1 - phi_2)/phi_2);   // Percenatge change in the speed of the machine that is required to restore the original Armature current but EMF raised to the original value and its Proportional to the speed and flux


// DISPLAY RESULTS
```

```
disp("EXAMPLE : 4.14 : SOLUTION :-") ;
printf("\n (a) Armature Current immediately after flux changes, Ia = %.1f A \n",Ia);
printf("\n (b) Percenatge change in the speed of the machine (that is required to restore the original Armature current) is %.2f Percenatge \n",perct);
```

Scilab code Exa 4.15 To find sharing of the two transformers when the load is 800kVA

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.15

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

S = 500 * 10 ^ 3;                    // Rating of the Generator-1 and Generator-2
VI = 800 * 10 ^ 3'                   // Actual load


// CALCULATIONS
//For Case (a)
```

```
Voc_a = 500;                                        // Open-circuit EMF Generator-1 and Generator-2 in Volts
I = 1000;                                           // Full load current in Amphere
perct_1a = 2/100;                                   // Percenatge fall of the Voltage in Generator-1
perct_2a = 3/100;                                   // Percenatge fall of the Voltage in Generator-2
V1a = Voc_a - (perct_1a * Voc_a);                   // Voltage in the Generator-1 in Volts when it falls to 2% at fully loaded
V2a = Voc_a - (perct_2a * Voc_a);                   // Voltage in the Generator-2 in Volts when it falls to 3% at fully loaded
// From Chacteristics can be assumed linear as, for Generator 1 is V = 500 + ((500-490)*I1)/(0-1000), V = -0.01*I1+500 and for Generator 2 is V = 500 + ((500-485)*I2)/(0-1000), V = 0.015*I2+500
// When sharing load of 800KVA at voltage, the load current will be I = I1+I2 = (800*1000)/V
// From above equations we get I1 = 1.5*I2 thus, 2.5*I2 = (800*1000)/V
// Putting the above equations in the Generator 2 equation we get V = -0.015*((800*1000)/(2.5*V))+500 solving we get, 25*V^2 - 12500V + 120000 = 0
V_a = poly ([120000 -12500 25],'x','coeff');
                                        // Expression for the load Voltage in Quadratic form
r_a = roots (V_a);

    // Value of the load Voltage in Volts (neglecting lower value)
I_a = VI/r_a(1,1);
I2_a = I_a/2.5;
I1_a = 1.5*I2_a;

// For Case (b)
```

```
perct = 2/100;                                  // Percenatge fall of the Voltage in Generator-1and Generator-2
Voc_1b = 500;                                   // Open-circuit EMF Generator-1 in Volts
Voc_2b = 505;                                   // Open-circuit EMF Generator-2 in Volts
I = 1000;                                       // Full load current in Amphere
V1 = Voc_1b - (perct * Voc_1b);                 // Voltage in the Generator-1 in Volts when it falls to 2% at fully loaded
V2 = Voc_2b - (perct * Voc_2b);                 // Voltage in the Generator-2 in Volts when it falls to 2% at fully loaded
// From Chacteristics can be assumed linear as, for Generator 1 is V = 500 + ((500-490)*I1)/(0-1000), V = -0.01*I1+500 and for Generator 2 is V = 505 + ((505-494.5)*I2)/(0-1000), V = -0.0101*I2+505
// When sharing load of 800KVA at voltage, the load current will be I = I1+I2 = (800*1000)/V
// From above equations we get V = -0.01*I1 + 500, I1 = -V/0.01 + 500/0.01 = 50000 - 100*V, V = -0.0101*I2 + 505 and I2 = 505/0.0101 - V/.0101 = 50000-99.0099*V
// Putting the above equations in the Current I equation we get I = I1+I2 = (800*1000)/V = 2*50000-199.0099*V solving we get, 199.0099*V^2 - 100000V + 800000 = 0
V_b = poly ([800000 -100000 199.0099],'x','coeff');                 // Expression for the load Voltage in Quadratic form
r_b = roots (V_b);                              // Value of the load Voltage in Volts (neglecting lower value)
I_b = VI/r_b(1,1);
```

```
I1_b = 50000-100*r_b(1,1)
I2_b = 50000-99.0099*r_b(1,1)


// DISPLAY RESULTS

disp("EXAMPLE : 4.15: SOLUTION :-");
printf("\n For case (a) Having open-circuit EMfs of
   500V but their voltage falls to 2 percent and 3
   percent when fully loaded Load Voltage,\n\n
   Load Voltage = %.2f V \n\n       Load current = %
   .2f A \n\n       Individual currents are %.2f A
   and %.2f A \n",r_a(1,1),I_a,I1_a,I2_a)
printf("\n For case (b) Having open-circuit EMfs of
   500V and 505V but their governors have identical
   speed regulation of 2 percent when fully loaded
   Load Voltage,\n\n       Load Voltage = %.2f V \n\n
         Load current = %.2f A \n\n       Individual
   currents are %.2f A and %.2f A \n",r_b(1,1),I_b,
   I1_b,I2_b)
printf("\n\n     [ TEXT BOOK SOLUTION IS PRINTED
   WRONGLY ( I verified by manual calculation )]\n"
   );
printf("\n       WRONGLY PRINTED ANSWERS ARE :-  For
   case(b) Load voltage = 493.35 V A instead of %.2f
    V \n ",r_b(1,1));
printf("\n

   Load current = 1634.73 A instead of %.2f A \n ",
   I_b)
printf("\n

   Individual currents 665 A and 1153.5 A instead of
    %.2f A and %.2f \n ",I1_b,I2_b)
printf("\n For Case (b):- From Calculation of the
   Load Voltage (V), rest of all the Calculated
   values in the TEXT BOOK is WRONG because of the
   value Load Voltage (V) is WRONGLY calculated and
```

```
the same used for the further Calculation part \n
")
```

Scilab code Exa 4.16 To find the graded resistance

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.16

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

Out_hp = 20;                        // Output of the
    Motor in HP
eta = 90/100;                       // Full load
    efficiency of the Motor
V = 220;                            // Motor voltage in
    Volts
ns = 5;                             // Number of the step
     of Starter
Rf = 220;                           // Field Resistance
    in Ohms
cr = 1.8;                           // Lowest Current
    rating is 1.8 times of the Full load current
Cu = 5/100;                         // Total Copper loss
    is 5% of the Input
```

```


// CALCULATIONS

Out = 20 * 746;                    ..      // Output of the Motor in Watt
Inp = (Out/eta);                           // Input of the Motor in KiloWatt
I = Inp/Rf;                                // Full-Load Current in Amphere
Cu_l = Inp*Cu;                             // Total Copper loss in Watts
olf = (V ^ 2)/Rf;                          // Ohmic loss in the Fiels in the Watts
Acu = Cu_l - olf;                          // Armature Copper loss in Watts
Ra = Acu/(I * I);                          // Armature Resistance in Ohms
I2 = I * cr;                               // Lower Current in Amphere
n = ns - 1;                                // Number of the Resistance
gama = ( (I2 * Ra)/Rf ) ^ (1/(n + 1));     // Current Ratio
I1 = I2/gama;                              // Initial Current in amphere
R1 = V/I1;                                 // Initial Resistance in Ohms
R2 = gama * R1;                            // Initial Resistance in Ohms
r1 = R1 - R2;                              // Graded Resistance in Ohms
R3 = gama * R2;                            // Initial Resistance in Ohms
r2 = gama * r1;                            // Graded Resistance in Ohms
r3 = gama ^ 2 * r1;                        // Graded Resistance in Ohms
```

```
r4 = gama ^ 3 * r1;                              //
    Graded Resistance in Ohms


// DISPLAY RESULTS

disp("EXAMPLE : 4.16 : SOLUTION :-") ;
printf("\n (a) Graded Resistances are %.4f Ohms, %.4
    f Ohms, %.4f Ohms and %.4f Ohms \n",r1,r2,r3,r4);
```

Scilab code Exa 4.17 To find lower current limit and resistance each section

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.17

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

I = 30;                       // Initial starting
    Current in Amphere
ns = 5;                       // Number of Steps of
    the starter
V = 500;                      // Operating Voltage of
    the DC Shunt Motor in Volts
```

```
I1 = 50;                          // Peak(Upper) Current
   limit in Amphere
Ra = 1.0;                         // Armature Circuit
   resistance in Ohms


// CALCULATIONS

R1 = V / I;                               // Initial
   Resistance in Ohms
gama = ( Ra/R1) ^ (1/(ns-1));             // Current Ratio
I2 = gama * I1;                           // Lower Current
    limit in Amphere
r1 = R1 * (1-gama);                       // Graded
   Resistances in Ohms
r2 = gama * r1;                           // Graded
   Resistances in Ohms
r3 = gama * r2;                           // Graded
   Resistances in Ohms
r4 = gama * r3;                           // Graded
   Resistances in Ohms


// DISPLAY RESULTS

disp("EXAMPLE : 4.17 : SOLUTION :-") ;
printf("\n (a)Graded Resistances are %.2f Ohms, %.4f
    Ohms, %.4f Ohms and %.4f Ohms \n",r1,r2,r3,r4);
```

Scilab code Exa 4.18 To find resistance steps in series motor starter

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
```

```
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.18

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

V = 500;                                    // Operating voltage of the DC series motor in Volts
P_hp = 10;                                  // Operating Power in HP
Il = 40;                                    // Lower currents limit in Amphere
Ih = 60;                                    // Higher currents limit in Amphere
f = 0.5/100;                                // Motor flux rises by 0.5% per amphere
Rt = 0.8;                                   // Motor terminal resistance in Ohms
eta = 90/100;                               // Motor efficiency

// CALCULATIONS

E1 = V-Il*Rt;                               // Induced EMF E1 in Volts
// Induced EMF, E2 = 500-60(0.8+r4) = 500 - 60*R4 where r4 is the fourth-step resistance, and R4 = 0.8+r4 and E1 = 1.1*E2 , 500 - 40*0.8 = 1.1*(500-60(0.8+r4)), 500-32 = 550-66*R4 thus we get, R4 = (550-500+32)/66 refer page no. 197
R4 = (V-(E1/1.1))/Ih;
```

```
r4 = R4 - Rt;                                                    // Fourth-step resistance in ohms
R3 = (V-((V-Il*R4)/1.1))/Ih;
r3 = R3 - R4;                                                    // Third-step resistance in ohms
R2 = (V-((V-Il*R3)/1.1))/Ih;
r2 = R2 - R3;                                                    // Second-step resistance in ohms
R1 = (V-((V-Il*R2)/1.1))/Ih;
r1 = R1 - R2;                                                    // First-step resistance in ohms


// DISPLAY RESULTS

disp("EXAMPLE : 4.18: SOLUTION :-");
printf("\n (a) The resistance steps in series motor stater are %.3f Ohms,%.4f Ohms, %.3f Ohms and %.2f Ohms \n",r1,r2,r3,r4)
```

Scilab code Exa 4.19 To find speed power torque of the motor

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.19
```

```
clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

Vac = 250;                    // Operating AC Voltage
    in Volts
V = 220;                      // Operating Voltage of
    the separately excited DC motor in Volts
fa = 30;                      // Firing Angle in
    Degree
Out_hp = 20;                  // DC Motor Output in HP
La = 20 * 10 ^ -3;            // Armature Inducatnce
    in Henry
Ra = 0.15;                    // Armature Resistance
    in Ohms
E_cons = 0.2;                 // EMF Constant in Volts
    /RPM
eta = 90/100;                 // Motor Operating
    Efficiency
N = 1000;                     // Rotational Speed of
    the Motor in RPM


// CALCULATIONS

out = 20 * 746;                              // DC Motor
    Output in Watt
Vt = ((Vac*2*sqrt(2))/%pi)*cosd(fa);         // Average
    Terminal volatge in Volts
Ia = out/(V*eta);                            // Rated
    Current in Amphere
E = Vt - ( Ia * Ra );                        // Back EMF
    in Volts
n = E/E_cons;                                // Speed of
    the Motor in RPM
```

```
e_cons = (E_cons*60)/ ( 2 * %pi);          // EMF Constant in Volts-Second per radians
T = e_cons * Ia;                           // Devolped Torque in Newton-Meter
pi = (E*Ia)+(Ia^2*Ra);                     // Power intake in Watts
pi_v = Vt * Ia;                            // Power intake in Watts (Verification)


// DISPLAY RESULTS

disp("EXAMPLE : 4.19 : SOLUTION :-") ;
printf("\n (a) Speed of the Motor, N = %.2f RPM \n",n);
printf("\n (b) Devolped Torque, T = %.2f N-m \n",T);
printf("\n (b) Power intake at Rated current and Firing angle of %.f deg, VI = %.1f W \n",fa,pi);
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n      WRONGLY PRINTED ANSWERS IS :- (a) T = 114.07 N-m instead of 143.91 N-m \n" );
```

Scilab code Exa 4.20 To find the firing angle and no load speed at 0 and 30 deg firing angle

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

```

```
// EXAMPLE : 4.20

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

N = 1500;                          // Speed of the separately excited DC Motor in RPM
Out_hp = 100;                      // Output of the DC Motor in HP
V = 500;                           // Motor operating Volatge in Volts
VL = 440;                          // 3-phase Line-line Voltage in Volts
f = 50;                            // Frequency in Hertz
Ra = 0.0835;                       // Armature Resistance in Ohms
La = 5.7 * 10 ^ -3;                // Armature Inductance in Henry
eta = 89/100;                      // Operating Efficiency of the Motor
E_cons = 0.35;                     // EMF constant in Volts per RPM


// CALCULATIONS
// For case (a)

Out = Out_hp * 746;                        // Output of the DC Motor in Watts
Ia = Out/(V*eta);                          // Rated Current in Amphere
Vph = VL/sqrt(3);                          // Phase Voltage in Volts
a = (3*Vph*sqrt(6)) / %pi;                 // Constant
E = N * E_cons;                            // Back EMF at
```

```
Rated speed
V = E + (Ia * Ra);                          // Terminal Volatge in Volts
alpa = acosd(V/a);                          // Firing Angle

// For case (b)
// Assumed that No load current is about 12% of full load current

Io = ( 0.12 * Ia );                              // No load current in Amphere
V_b1 = a * cosd(0);                              // Terminal Voltage at Firing Angle 0 deg
E_b1 = V_b1 + (Io * Ra);                         // Back EMF at Firing Angle 0 deg
N_b1 = E_b1/E_cons;                              // No load speed at Firing Angle 0 deg
V_b2 = a * cosd(30);                             // Terminal Voltage at Firing Angle 30 deg
E_b2 = V_b2 + (Io * Ra);                         // Back EMF at Firing Angle 30 deg
N_b2 = E_b2/E_cons;                              // No load speed at Firing Angle 30 deg


// DISPLAY RESULTS

disp("EXAMPLE : 4.20 : SOLUTION :-") ;
printf("\n  (a)  Firing Angle at Rated speed and Rated Motor Current , alpa = %.2f deg \n",alpa);
printf("\n (b.1) No load speed at Firing Angle 0 deg , N = %.1f RPM \n",N_b1);
printf("\n (b.2) No load speed at Firing Angle 30 deg, N = %.1f RPM \n",N_b2);
```

Scilab code Exa 4.21 To find speed control range of the duty cycle

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.21

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA
// Given that Back EMF is Zero Because Motor is at
    Standstill

V = 250;                              // DC supply Voltage
     to separately excited DC Motor in Volts
Ra = 1.0;                             // Armature
    Resistance in Ohms
La = 30 * 10 ^ -3;                    // Armature
    Inductance in Henry
E_cons = 0.19;                        // Motor (EMF)
    Constant in Volts per RPM
Ia = 25;                              // Average Armature
    Current in Amphere


// CALCULATIONS

V1 = Ia * Ra;                         // Minimum Terminal
    Volatge in Volts
alpa_mini = Ia/V;                     // Minimum Duty
    Cycle
```

```
alpa_max = 1.0;                    // Maximum Duty
    Cycle
V2 = V;                            // Maximum Terminal
    Volatge in Volts when Duty cycle (alpa) is 1.0
E2 = V2 - (V1 * alpa_max);         // Back EMF at
    Maximum Duty cycle ( i.e alpa = 1.0) in Volts
N = E2/E_cons;                     // Speed of the
    Motor


// DISPLAY RESULTS

disp("EXAMPLE : 4.21 : SOLUTION :-") ;
printf("\n (a) Range of the Speed is from 0 RPM to %
    .2f RPM and Range of the Duty Cycle is %.1f to %
    .1f \n",N,alpa_mini,alpa_max);
```

Scilab code Exa 4.22 To find new speed of the motor

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.22

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

```

```
N1 = 1000;                    // Speed of the DC shunt Motor in RPM
Out_hp = 20;                  // Output of the DC shunt Motor in HP
V = 220;                      // Motor operating Volatge in Volts
Ra = 0.9;                     // Armature Resistance in Ohms
Rf = 200;                     // Field Resistance in Ohms
eta = 89/100;                 // Operating Efficiency of the Motor
Ra_a = 0.2;                   // Resistance inserted to the armature circuit


// CALCULATIONS

out = Out_hp * 746;                     // Output of the DC Motor in watts
I =  out/(V * eta);                     // Rated current in Amphere
If = V/Rf;                              // Field current in Amphere
Ia1 = I - If;                           // Armature current in Amphere
E1 = V - (Ia1 * Ra);                    // Back EMF in Volts
// Assuming that Torque and Armature current is constant
E2 = V - ( Ra + Ra_a ) * Ia1;           // New Back EMF in Volts
N2 = N1*(E2/E1);                        // New speed in RPM


// DISPLAY RESULTS

```

```
disp("EXAMPLE : 4.22: SOLUTION :-") ;
printf("\n (a) New Speed of the Motor , N2 = %.2f
    RPM \n",N2);
```

Scilab code Exa 4.23 To find new speed of the motor

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.23

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

N1 = 600;                       // Speed of the DC shunt
     Motor in RPM
Out_hp = 10;                    // Output of the DC
    shunt Motor in HP
V = 220;                        // Motor operating
    Volatge in Volts
Ra = 1.5;                       // Armature Resistance
    in Ohms
Rf = 250;                       // Field Resistance in
    Ohms
eta = 88/100;                   // Operating Efficiency
    of the Motor
Rf_a = 50;                      // Resistance inserted
```

```
    to the field circuit


// CALCULATIONS

out = Out_hp * 746                      // Output of the DC Motor in watts
I =  out/(V * eta);                     // Rated current in Amphere
If1 = V/Rf;                             // Field current in Amphere
Ia1 = I - If1;                          // Aramature current in Amphere
E1 = V - Ra*Ia1;                        // Back EMF in Volts
If2 = V/(Rf+Rf_a);                      // New Field current in Amphere after 50 Ohms Resistance inserted to the field circuit

// Refer page no. 217 we have T1 = K*If1*Ia1 proportional to 1/W1^2 and T1 = K*If2*Ia2 proportional to 1/W2^2 thus T1/T2 = (If1*Ia1)/(If2*Ia2) = (W2^2)/(W1^2) = (N2^2)/(N1^2), Ia2 = (If1*Ia1*W1^2)/(If1*W1^2) = (0.88*37.65*N2^2)/(0.733*600*600)
// Now New EMF E2 is E2 = V - Ia2*Ra, E1/E2 = (k*If1*N1)/(k*If2*N2), E2 = (0.733*N2)/(0.88*600) = 220 - (0.88*37.65*1.5*N2^2)/(0.733*600*600) Thus we have 0.001388*N2^2 = 220 - 1.833*10^-4*N2
N2 = poly ([-220 0.001388 1.833*10^-4],'x','coeff');          // Expression for the new speed of the motor in Quadratic form
r = roots (N2);
    // Value of the New speed of the motor in RPM


// DISPLAY RESULTS
```

```
disp("EXAMPLE : 4.23 : SOLUTION :-") ;
printf("\n (a) New speed of the motor , N2 = %.2f RPM
    nearly %.f RPM \n",r(2,1),r(2,1));
```

Scilab code Exa 4.24 To find resistances for both dynamic and counter current braking

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.24

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

N1 = 1000;                   // Speed of the DC shunt
    Motor in RPM
Out_hp = 10;                 // Output of the DC
    shunt Motor in HP
V = 220;                     // Motor operating
    Volatge in Volts
Ra = 0.5;                    // Armature Resistance
    in Ohms
Rf = 100;                    // Field Resistance in
    Ohms
eta = 90/100;                // Operating Efficiency
    of the Motor
```

```


// CALCULATIONS

out = Out_hp * 746;                    // Output of the DC Motor in watts
I =  out/(V * eta);                    // Rated current in Amphere
If = V/Rf;                             // Field current in Amphere
Ia = I-If;                             // Armature current in Amphere
E = V - (Ia*Ra);                       // Back EMF of the Motor in Volts
Rd = E/I;                              // Resistance at Dynamic Braking in Ohms
Rc = (V+E)/I;                          // Resistance at Counter Current Braking in Ohms

// DISPLAY RESULTS

disp("EXAMPLE : 4.24 : SOLUTION :-") ;
printf("\n (a) Resistance at Dynamic Braking, Rd = %.2f Ohms \n",Rd);
printf("\n (b) Resistance at Counter Current Braking, Rc = %.1f Ohms \n",Rc);
```

Scilab code Exa 4.25 To find resistance must be added in the field circuit

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

```

```
// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.25

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

V = 220;                    // Motor operating Volatge in Volts
Ra = 1.0;                   // Armature Resistance in Ohms
Rf = 220;                   // Field Resistance in Ohms
Ia1 = 20;                   // Armature Current in Amphere
N1 = 800;                   // Motor drving speed in RPM
N2 = 1000;                  // To be obtained speed in RPM


// CALCULATIONS

If = V/Rf;                                  // Field Current in Amphere
E1 = V - ( Ia1 - If ) * Ra;                 // Back EMF E1 at N1 Speed in Volts
// Now we have Back EMF E2 at N2 Speed, E2 = 220-Ia2*1.0 = 220-Ia2 and the field flux be proportional to the field current, since torque is constant we get, If2*Ia2 = If1*Ia1 = 20
// Thus (220-Ia2)/201 = (If2*N2)/(If1*N1) = If2*(1000/(800*1.0)), 220-Ia2 = 201*(10/8)*(20/Ia2) = 5000/Ia2 solving this we get Ia2^2 - 220Ia2 + 2000 = 0
```

```
Ia2 = poly ([5000 -220 1],'x','coeff');                    // Expression for the new Armature current in Quadratic form
r = roots (Ia2);                                            // Value of the New Armature current in Amphere
If2 = If*(Ia1/r(2,1));                                      // New field current in Amphere when New Armature current is 39.29A
Rfn = V/If2;                                                // New field resistance in ohms
ERf = Rfn - Rf;                                             // Extra resistance in Ohms


// DISPLAY RESULTS

disp("EXAMPLE : 4.25 : SOLUTION :-") ;
printf("\n (a) Extra resistance should be added in the field circuit for raising the speed to %.f RPM is = %.2f Ohms \n",N2,ERf);
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n      WRONGLY PRINTED ANSWERS ARE :- (a) Ia2 = 39.29 A and 180.71 A instead of %.2f A and %.2f A \n ",r(1,1),r(2,1));
printf("\n                                  (b) Extra resistance required is 212.22 Ohms  instead of %.2f Ohms \n ",ERf);
printf("\n From Calculation of the New armature current (Ia2), rest all the Calculated values in the TEXT BOOK is WRONG because of the New armature current (Ia2) value is WRONGLY calculated and the same used for the further
```

```
Calculation part \n")
```

Scilab code Exa 4.26 To find voltage and current when magnetic circuit unsaturated and saturated

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.26

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

R = 2.0;                              // DC series Motor
   Resistance between the terminals in Ohms
V1 = 220;                             // Motor Operating
   voltage in Volts
N1 = 500;                             // Rotation Sped of
   the DC series Motor in RPM
I1 = 22;                              // Current in Motor
   in Amphere
N2 = 600;                             // New Rotation Sped
    of the DC series Motor in RPM


// CALCULATIONS
// For case (a) when magnetic circuit is Unsaturated
```

```
E1 = V1 - (I1 * R);                      // Back EmF at N1 Speed in Volts
I2_a = (N2/N1)*I1;                       // Current in Motor at N2 speed in Amphere
E2_a = (E1*I2_a*N2)/(I1*N1);            // Back EmF at N2 Speed in Volts
V2_a = E2_a + (I2_a * R);               // Applied Voltage at N2 Speed in Volts

// For case (b) when magnetic circuit is Saturated

I2_b = ((N2/N1)^2)*I1;                  // Current in Motor at N2 speed in Amphere
E2_b = (N2/N1)*E1;                      // Back EmF at N2 Speed in Volts
V2_b = E2_b + (I2_b * R);                  // Applied Voltage at N2 Speed in Volts


// DISPLAY RESULTS

disp("EXAMPLE : 4.26 : SOLUTION :-") ;
printf("\n (a.1) Applied Voltage when magnetic circuit is Unsaturated,V2 = %.2f V \n",V2_a);
printf("\n (a.2) Current in Motor when magnetic circuit is Unsaturated, I2 = %.1f A \n",I2_a);
printf("\n (b.1) Applied Voltage when magnetic circuit is Saturated,V2 = %.2f V \n",V2_b);
printf("\n (b.2) Current in Motor when magnetic circuit is Saturated, I2 = %.2f A \n",I2_b);
```

Scilab code Exa 4.27 To find new speed and current

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.27

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

V = 220;                        // DC series Motor operating Volatge in Volts
Ra = 1.0;                       // Armature Resistance in Ohms
Rf = 1.0;                       // Field Resistance in Ohms
I1 = 20;                        // Armature Current in Amphere
N1 = 1800;                      // Motor drving speed in RPM
If = 20;                        // Armature Current in Amphere
Rd = 0.5;                       // Diverter Resistance in Ohms


// CALCULATIONS

E1 = V - ( Ra + Rf ) * I1;                     // Back EMF in Volts
I2 = sqrt(3)*I1;                               // New Armature current in Amphere
If2 = ( Rd * I2 )/(Ra + Rd);                   // New
```

```
field Current in Amphere
E2 = V - ( Ra + (1/3))*I1;                    // New BAck EMF in Volts
N2 = (N1*E2*If)/(E1*If2);                     // New Rotation speed of the Motor in RPM


// DISPLAY RESULTS

disp("EXAMPLE : 4.27 : SOLUTION :-") ;
printf("\n (a) New Rotation speed of the Motor at torque remains constant , N2 = %.f RPM \n",N2);
printf("\n (b.1) New Armature Current at torque remains constant , I2 = %.2f A \n",I2);
printf("\n (b.2) New Field Current at torque remains constant , If2 = %.2f A \n",If2);
```

Scilab code Exa 4.28 To find speed torque efficiency of the motor

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.28

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

V = 220;                         // DC shunt Motor
```

```
operating Volatge in Volts
Ra = 1.0;                      // Armature Resistance in Ohms
Rf = 220;                      // Field Resistance in Ohms
In1 = 5;                       // No-Load Current in Amphere
N1 = 1000;                     // Motor drving speed in RPM
inp = 10 * 10 ^ 3;             // Motor input in Watts


// CALCULATIONS

If = V/Rf;                                  // Field Current in Amphere
Ian1 = In1 - If;                            // No load Armature Current in Amphere
E1 = V - (Ian1 * Ra);                       // Back EMF in Volts
Iin = inp/V;                                // Motor Input Current in Amphere
Ia = Iin - If;                              // Armature current in Amphere
E2 = V - (Ia * Ra);                         // New Back EMF in Volts
N2 = (N1*E2)/E1;                            // New Rotation speed of the Motor in RPM
Pa = E2 * Ia;                               // Developed Armature Power in Watts
T = Pa/((2*%pi*N2)/60);                     // Developed Torque in Newton-Meter
Pi = V * In1;                               // No-Load input Power in Watts
Pa_cu = Ian1 ^ 2 * Ra;                      // No-Load Armature Copper loss in Watts
F_loss = Pi - Pa_cu;                        // Fixed losses in Watts
```

```
Pa_cu_load = Ia ^ 2 * Ra;                    // Loaded Armature Copper loss in Watts
Total_loss = F_loss + Pa_cu_load;            // Total losses in loaded conditions in Watts
out = inp - Total_loss;                      // Shaft output in Watts
Ts = out/((2*%pi*N2)/60);                    // Shaft torque in Newton-Meter
eta = (out/inp)*100;
        // Efficiency in Percentage


// DISPLAY RESULTS

disp("EXAMPLE : 4.28 : SOLUTION :-") ;
printf("\n  (a)  New Rotation speed of the Motor , N2 = %.f RPM \n",N2);
printf("\n (b.1) Developed Torque, T = %.1f N-m A \n",T);
printf("\n (b.2) Shaft torque, Ts = %.2f N-m \n",Ts);
printf("\n  (c)  Efficiency in Percentage, eta = %.2f percent  \n",eta);
```

Scilab code Exa 4.29 To find efficiency of each machine

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.29
```

```
clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

V = 220;                        // Shunt Motor operating Line Volatge in Volts
Ra = 0.2;                       // Armature Resistance in Ohms
Iam = 72;                       // Motor Armature current in Amphere
I = 12;                         // Line Current in Amphere
Ifm = 1;                        // Motor field Current in Amphere
Ifg = 1.5;                      // Generator field Current in Amphere


// CALCULATIONS

Iag = Iam - I;                                  // Geneartor Armature current in Amphere
Pfm = V * Ifm;                                  // Loss in Motor Field winding in Watts
Pfg = V * Ifg;                                  // Loss in Geneartor Field winding in Watts
loss_ma = Iam ^ 2 * Ra;                         // Loss in Motor Armature circuit in Watts
loss_ga = Iag ^ 2 * Ra;                         // Loss in Generator Armature circuit in Watts
Em = V - Iam * Ra;                              // Motor EMF in Volts
Eg = V + Iag * Ra;                              // Generator EMF in Volts
T_loss = (V*I) - (Ra*Iam^2 + Ra*Iag^2);  // Total
```

```
    Iron and Rotational Loss in Watts
Pim = (V*Iam)+(V*Ifm);                     // Motor input in Watts
Wc = 0.5 * T_loss;                         // Total Iron and Rotational Loss in each Machine in Watts
Wm = Wc+(Ra*Iam^2)+V*Ifm;                  // Motor losses in Watts
Pom = Pim - Wm;                            // Motor output in Watts
eta_m = (1-(Wm/Pom))*100;                  // Motor Efficiency in Percentage
Pog = V*Iag;                               // Generator output in Watts
Wg = Wc+(Ra*Iag^2)+V*Ifg;                  // Generator losses in Watts
Pin = Pog + Wg;                            // Generator input power in Watts
eta_g = (1-(Wg/Pin))*100;                  // Generator Efficiency in Percentage


// DISPLAY RESULTS

disp("EXAMPLE : 4.29 : SOLUTION :-") ;
printf("\n (a) Motor Efficiency , eta = %.2f Percentage \n ",eta_m);
printf("\n (b) Generator Efficiency , eta = %.2f Percantage \n ",eta_g);
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n       WRONGLY PRINTED ANSWERS ARE :- (a) Total Iron and Rotational Loss = 720 W instead of %.1f W \n ",T_loss);
printf("\n                                      (b) Pim = 15912 W instead of %.f W \n ",Pim);
printf("\n                                      (c) Wm = 1371.4 Winstead of %.1f W \n ",Wm);
```

```
printf("\n                                           (d)
   Pom = 14540.6 W instead of %.1f W \n ",Pom);
printf("\n                                           (e)
   eta_m = 90.54 Percentage  instead of %.2f
   Percentage \n ",eta_m);
printf("\n                                           (f)
   eta_g = 93.22 Percentage  instead of %.2f
   Percentage \n ",eta_g);
printf("\n From Calculation of the Total Iron and
   Rotational Loss in each Machine (Wc), rest all
   the Calculated values in the TEXT BOOK is WRONG
   because of the Total Iron and Rotational Loss in
   each Machine (Wc) value is WRONGLY calculated and
    the same used for the further Calculation part \
   n")
```

Scilab code Exa 4.30 To find efficiency of motor and generator and also torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 4 : DIRECT CURRENT MACHINES

// EXAMPLE : 4.30

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

```

```
Vg = 110;                           // Generator operating Volatge in Volts
Vm = 102;                           // Motor operating Volatge in Volts
Vs = 274;                           // Supply Volatge in Volts
Ra = 1.0;                           // Armature Resistance in Ohms for both the Machines
Rf = 0.82;                          // Field Resistance in Ohms for both the Machines
N = 1440;                           // Speed of the Set in RPM
Ig = 17.5;                          // Generator current in Amphere
Im = 9.5;                           // Motor current in Amphere


// CALCULATIONS

Pi = Vs * Im;                                               // Input power in Watts
Pg = Vg * Ig;                                               // Output power in Watts
Pim = Vm * Im;                                              // Power Input to the Motor in Watts
Pl = Pi - Pg;                                               // Losses in the entire set in Watts
Pcu = Im^2*(Ra+2*Rf) + Ig^2*Ra;                             // Total Copper loss for both the Machines in Watts
P_l = Pi - Pg - Pcu;                                        // Frictional , Windage and core losses of the both Machines in Watts
Po = P_l/2;                                                 // Frictional , Windage and core loss of each Machines in Watts
eta_m = (1 - ((Po + Im^2*(Ra+Rf))/Pim))*100;                // Motor Effiicency in Percentage
```

```
Pig = Pg + Po + Ig^2*Ra + Im^2*Rf;                    // Generator input in Watts
eta_g = (Pg / Pig)*100;                               // Generator Effiicency in Percentage
T = (Vg*Ig *60)/(2*%pi*N);                            // Torque in Newton-Meter


// DISPLAY RESULTS

disp("EXAMPLE : 4.30 : SOLUTION :-") ;
printf("\n (a) Motor Efficiency , eta_m = %.2f percentage \n ",eta_m);
printf("\n (b) Generator Efficiency , eta_g = %.2f Percentage \n ",eta_g);
printf("\n (c) Torque , T = %.2f N-m \n ",T);
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n      WRONGLY PRINTED ANSWERS ARE :- (a) Generator input = 2307.5 W instead of %.f W \n ",Pig);
printf("\n                                    (b) eta_g = 83.42 Percenatge  instead of %.2f Percentage \n ",eta_g);
printf("\n From Calculation of the Generator input, rest all the Calculated values in the TEXT BOOK is WRONG because of the Generator input value is WRONGLY calculated and the same used for the further Calculation part \n")
```

Chapter 5

Induction Machines

Scilab code Exa 5.1 To find slot per pole per phase in each case

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.1

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA
// For Case (a)

S_a = 30;                  // Total number of Slots
m_a = 3;                   // Total number of Poles
p_a = 2;                   // Total number of Phases

```

```
// For Case (b)

S_b = 60;                    // Total number of Slots
m_b = 3;                     // Total number of Poles
p_b = 4;                     // Total number of Phases

// For Case (c)

S_c = 24;                   // Total number of Slots
m_c = 3;                    // Total number of Poles
p_c = 4;                    // Total number of Phases

// For Case (d)

S_d = 12;                   // Total number of Slots
m_d = 3;                    // Total number of Poles
p_d = 2;                    // Total number of Phases


// CALCULATIONS
// For Case (a)

spp_a = S_a/(p_a*m_a);                    // Slot per poles per phase

// For Case (b)

spp_b = S_b/(p_b*m_b);                    // Slot per poles per phase

// For Case (c)

spp_c = S_c/(p_c*m_c);                    // Slot per poles per phase

// For Case (d)

spp_d = S_d/(p_d*m_d);                    // Slot per
```

```
poles per phase


// DISPLAY RESULTS

disp("EXAMPLE : 5.1 : SOLUTION :-") ;
printf("\n For case (a) Slot per poles per phase, spp = %.f \n ",spp_a);
printf("\n For case (b) Slot per poles per phase, spp = %.f \n ",spp_b);
printf("\n For case (c) Slot per poles per phase, spp = %.f \n ",spp_c);
printf("\n For case (d) Slot per poles per phase, spp = %.f \n ",spp_d);
```

Scilab code Exa 5.2 To find slot per pole per phase phase allocation series and state whether balanced winding is possible

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.2

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA
// For Case (a)

```

```
S_a = 54;                      // Total number of Slots
m_a = 3;                       // Total number of Poles
p_a = 8;                       // Total number of Phases

// For Case (b)

S_b = 32;                      // Total number of Slots
m_b = 3;                       // Total number of Poles
p_b = 4;                       // Total number of Phases

// For Case (c)

S_c = 30;                     // Total number of Slots
m_c = 3;                      // Total number of Poles
p_c = 4;                      // Total number of Phases


// CALCULATIONS
// For Case (a)

spp_a = S_a/(p_a*m_a);                    // Slot per poles per phase
l_a = 0 * spp_a;                          // Phase allociation Series
m_a = 1 * spp_a;                          // Phase allociation Series
n_a = 2 * spp_a;                          // Phase allociation Series
o_a = 3 * spp_a;                          // Phase allociation Series
p_a = 4 * spp_a;                          // Phase allociation Series
d_a = 0;                                  // d_a = l_a (Rounding off)
e_a = 2;                                  // e_a = m_a (Rounding off)
f_a = 4;                                  // f_a = n_a (Rounding off)
```

```
g_a = 6;                              // g_a = o_a ( Rounding off)
h_a = 9;                              // h_a = p_a ( Rounding off)
R_a = e_a - d_a;                      // Phase allociation
Y_a = f_a - e_a;                      // Phase allociation
B_a = g_a - f_a;                      // Phase allociation
R1_a = h_a - g_a;                     // Phase allociation

// For Case (b)

spp_b = S_b/(p_b*m_b);                // Slot per poles per phase
l_b = 0 * spp_b;                      // Phase allociation Series
m_b = 1 * spp_b;                      // Phase allociation Series
n_b = 2 * spp_b;                      // Phase allociation Series
o_b = 3 * spp_b;                      // Phase allociation Series
d_b = 0;                              // d_b = l_b ( Rounding off)
e_b = 2;                              // e_b = m_b ( Rounding off)
f_b = 5;                              // f_b = n_b ( Rounding off)
g_b = 8;                              // g_b = o_b ( Rounding off)
R_b = e_b - d_b;                      // Phase allociation
Y_b = f_b - e_b;                      // Phase allociation
B_b = g_b - f_b;                      // Phase
```

```
                                        allociation

// For Case (c)

spp_c = S_c/(p_c*m_c);                   // Slot per poles per phase
l_c = 0 * spp_c;                         // Phase allociation Series
m_c = 1 * spp_c;                         // Phase allociation Series
n_c = 2 * spp_c;                         // Phase allociation Series
d_c = 0;                                 // d_b = l_b (Rounding off)
e_c = 2;                                 // e_b = m_b (Rounding off)
f_c = 5;                                 // f_b = n_b (Rounding off)
R_c = e_c - d_c;                         // Phase allociation
Y_c = f_c - e_c;                         // Phase allociation

// DISPLAY RESULTS

disp("EXAMPLE : 5.2 : SOLUTION :-") ;
printf("\n For Case (a) Slot per poles per phase , spp = %.3f \n ",spp_a);
printf("\n                    Phase allociation series is %.f, %.f, %.f, %.f, %.f, %.f, %.f, %.f, %.f, slots are allocated respectively to R, Y, B, R, Y, B, R, Y, B....... phase in Sequence\n ",R_a,Y_a,B_a,R1_a,R_a,Y_a,B_a,R1_a,R_a);
printf("\n                    By seeing Sequence its Slot per pole per phase is an Integer and such, balanced winding may be possible \n");
printf("\n For Case (b) Slot per poles per phase , spp = %.3f \n ",spp_b);
```

```
printf("\n                    Phase allociation series is
    %.f, %.f, %.f \n",R_b,Y_b,B_b);
printf("\n                    By seeing Sequence its Slot
    per pole per phase are not Integer therefore R-
   phase will have 8 slots whereas Y-phase and B-
   phase will have 12 slots \n");
printf("\n  For Case (c) Slot per poles per phase   ,
    spp = %.1f \n ",spp_c);
printf("\n                    Phase allociation series is
    %.f, %.f, %.f, %.f, %.f, %.f, %.f, %.f, %.f, %.f
   , %.f, %.f slots are allocated respectively to R,
    Y, B, R, Y, B, R, Y, B, R, Y, B....... phase in
   Sequence\n ",R_c,Y_c,R_c,Y_c,R_c,Y_c,R_c,Y_c,R_c,
   Y_c,R_c,Y_c);
printf("\n                    By seeing Sequence its Slot
    per pole per phase is an Integer and such,
   balanced winding may be possible \n");
```

Scilab code Exa 5.3 To find pitch factor

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.3

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA
```

```
s = 24;                        // Total number of the pole
p = 4;                         // Total number of the poles in the Alternator


// CALCULATIONS
// For Case (a) Short pitching by one Slots

spp = s/p;                          // Slot per pole
E_a = ((180*2)/24)*(4/2);           // Slot angle in Electrical
kp_a = cosd(E_a/2);                 // Pitch Factor
kp5_a = cosd((5*E_a)/2);            // Pitch Factor
kp7_a = cosd((7*E_a)/2);            // Pitch Factor

// For Case(b) Short pitching by two Slots

E_b = 2*((180*2)/24)*(4/2);         // Slot angle in Electrical
kp_b = cosd(E_b/2);                 // Pitch Factor
kp5_b = cosd((5*E_b)/2)             // Pitch Factor
kp7_b = cosd((7*E_b)/2);            // Pitch Factor


// DISPLAY RESULTS

disp("EXAMPLE : 5.3 : SOLUTION :-") ;
printf("\n For Case (a) Short pitching by one Slots :- Pitch Facor , kp = %.4f \n ",kp_a);
printf("\n                                                kp5 = %.4f \n ",kp5_a);
printf("\n                                                kp7 = %.4f \n ",kp7_a);
printf("\n For Case (a) Short pitching by Two Slots
```

```
    :- Pitch Facor , kp = %.4f \n ",kp_b);
printf("\n

    kp5 = %.4f \n ",kp5_b);
printf("\n

    kp7 = %.4f \n ",kp7_b);
```

Scilab code Exa 5.4 To find distribution factor

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.4

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

s = 60;                        // Total number of Slot
m = 3;                         // Total number of Phase
p = 4;                         // Total number of Pole


// CALCULATIONS

M = s/(m*p);                                    // Slot
     per pole per Phase
sigma = 180/m;                                  //
```

```
Phase Spread in angle (deg)
Ka = sind((M*sigma)/2)/(M*sind(sigma/2));     // Distribution Factor


// DISPLAY RESULTS

disp("EXAMPLE : 5.4 : SOLUTION :-");
printf("\n (a) Distribution Factor, Ka = %.1f \n",Ka)
```

Scilab code Exa 5.7 To find the synchronous speed

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.7

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

f = 50;                        // Frequency of the 2-pole Induction Motor
p = 2;                         // Total Number of Poles


// CALCULATIONS

```

```
Ns = (120*f)/p;                     // Synchronous Speed
    in RPM
Ns5 = -(120*f)/(5*p);               // Synchronous Speed
    of 5th order space harmonic in RPM
N5 = -(120*5*f)/p;                  // Synchronous Speed
    of 5th order time harmonic in RPM
Ns7 = (120*f)/(7*p);               // Synchronous Speed
   of 7th order space harmonic in RPM
N7 = (120*7*f)/p;                  // Synchronous Speed
   of 7th order time harmonic in RPM


// DISPLAY RESULTS

disp("EXAMPLE : 5.7 : SOLUTION :-");
printf("\n (a.1) Synchronous Speed of 5th order
   space harmonic, Ns5 = %.f RPM \n",Ns5)
printf("\n (a.2) Synchronous Speed of 5th order time
    harmonic, N5 = %.f RPM \n",N5)
printf("\n (b.1) Synchronous Speed of 7th order
   space harmonic, Ns7 = %.2f RPM \n",Ns7)
printf("\n (b.2) Synchronous Speed of 7th order time
    harmonic, N7 = %.f RPM \n",N7)
```

Scilab code Exa 5.8 To find the frequency of the rotor current

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.8

```

```
clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

p_a = 6;                              // Total number of Poles in the Alternator
p_m = 4;                              // Total number of Poles of Induction Motor
N_a = 900;                            // Running Speed of the Alternator in RPM
N_m = 1250;                           // Running Speed of the Induction Motor in RPM
m = 3;                                // Total Number of phase in Induction Motor


// CALCULATIONS

f = (N_a*p_a)/120;                        // Frequency of the 6-pole Alternator running at 900 RPM in Hertz
Ns = (120*f)/p_m;                         // Synchronous Speed of 4-pole Induction Motor in RPM
s = (Ns-N_m)/Ns;                          // Slip
fr = s*f;                                 // Frequency of the Rotor Current in Hertz


// DISPLAY RESULTS

disp("EXAMPLE : 5.8 : SOLUTION :-");
printf("\n (a) Frequency of the Rotor Current, fr = %.2f Hz \n",fr)
```

Scilab code Exa 5.9 To find the slip and frequency of rotor current

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.9

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

p = 2;                              // Total number of Poles of Induction Motor
f = 50;                             // Frequency in Hertz
Nr = 2800;                          // Running Speed of the Induction Motor in RPM
m = 3;                              // Total Number of phase in Induction Motor
V = 400;                            // Operating Voltage of Induction Motor in Volts


// CALCULATIONS

Ns = (120*f)/p;                         // Synchronous Speed in RPM
s = 100*((Ns-Nr)/Ns);                   // Slip in Percentage
fr = (s/100)*f;                         // Frequency of the Rotor Current in Hertz


```

```
// DISPLAY RESULTS

disp("EXAMPLE : 5.9 : SOLUTION :-");
printf("\n (a) Slip , s = %.2f percent \n",s);
printf("\n (b) Frequency of the Rotor Current , fr =
   %.2f Hz \n",fr)
```

Scilab code Exa 5.10 To find the rotor speed

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.10

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA


m = 3;                          // Total Number of
   phase in Induction Motor
p = 4;                          // Total number of
   Poles in Induction Motor
f = 50;                         // Frequency in
   Hertz
s = 0.03;                       // Slip


```

```
// CALCULATIONS

Ns = (120*f)/p;                         // Synchronous
   Speed in RPM
Nr = (1-s)*Ns;                          // Rotor Speed
   in RPM


// DISPLAY RESULTS

disp("EXAMPLE : 5.10 : SOLUTION :-");
printf("\n (a) Rotor Speed , Nr = %.f RPM \n",Nr)
```

Scilab code Exa 5.11 To find the speed of forward and backward rotating magnetic field with respect to stator and rotor

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.11

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA


m = 3;                               // Total Number of
   phase in Induction Motor
```

```
p = 6;                              // Total number of Poles of Induction Motor
f = 50;                             // Frequency in Hertz
s = 0.03;                           // Slip


// CALCULATIONS

Ns = (120*f)/p;                         // Synchronous Speed in RPM
Nr = (1-s)*Ns;                          // Rotor Speed in RPM
Nf = Ns - Nr;                           // Speed of Forward Rotating magnetic fields with respect to stator and rotor in RPM
Nb = Ns + Nr;                           // Speed of Backward Rotating magnetic fields with respect to stator and rotor in RPM


// DISPLAY RESULTS

disp("EXAMPLE : 5.11 : SOLUTION :-");
printf("\n (a) Speed of Forward Rotating magnetic fields with respect to stator and rotor is equal to + %.f RPM \n",Nf)
printf("\n (b) Speed of Backward Rotating magnetic fields with respect to stator and rotor is equal to + %.f RPM \n",Nb)
```

Scilab code Exa 5.12 To find the speed of forward and backward rotating magnetic field with respect to stator and rotor

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.12

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

m = 3;                              // Total Number of phase in Induction Motor
p = 2;                              // Total number of Poles of Induction Motor
f = 50;                             // Frequency in Hertz
s = 0.05;                           // Slip


// CALCULATIONS

Ns = (120*f)/p;                         // Synchronous Speed in RPM
Nr = (1-s)*Ns;                          // Rotor Speed in RPM
Nf = s*Ns;                              // Speed of Forward Rotating magnetic fields with respect to stator and rotor in RPM
Nb = (p-s)*Ns;                          // Speed of Backward Rotating magnetic fields with respect to stator and rotor in RPM
fr = (p-s)*f;                           // Backward rotating magnetic field induces a current of
```

```
frequency in Hertz


// DISPLAY RESULTS

disp("EXAMPLE : 5.12 : SOLUTION :-");
printf("\n (a) Speed of Forward Rotating magnetic fields with respect to stator and rotor is equal to + %.f RPM \n",Nf)
printf("\n (b) Speed of Backward Rotating magnetic fields with respect to stator and rotor is equal to + %.f RPM \n",Nb)
```

Scilab code Exa 5.13 To find the speed of forward and backward rotating magnetic field with respect field to rotor to stator

```

// ELECTRICAL MACHINES
// R.K. Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.13

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA


m = 3;                              // Total Number of phase in Induction Motor
```

```
p = 4;                              // Total number of Poles of Induction Motor
f = 50;                             // Frequency in Hertz
s = 0.05;                           // Slip


// CALCULATIONS

Ns = (120*f)/p;                         // Synchronous Speed in RPM
fr = s*f;                               // Rotor-induced Frequency of forward field in Hertz
Nfr = s*Ns;                             // Speed of Forward Rotating magnetic fields with respect to rotor surface in RPM
f2r = s*f;                               // Rotor-induced Frequency of Backward field in Hertz
Nbr = -(s*Ns);                           // Speed of Backward Rotating magnetic fields with respect to rotor surface in RPM
Nr = (1-s)*Ns;                           // Rotor running in Forward direction in RPM
Nfs = Nr+(s*Ns);                         // Speed of Forward Rotating magnetic fields with respect to stator surface in RPM
Nbs = Nr-(s*Ns);                         // Speed of Backward Rotating magnetic fields with respect to stator surface in RPM
Nbs_new = -(0.5*Ns)+(1-0.5)*Nr;          // Speed of Backward Rotating magnetic fields with respect to stator for 50% of slip in RPM


// DISPLAY RESULTS

disp("EXAMPLE : 5.13 : SOLUTION :-");
printf("\n (a.1) Speed of Forward Rotating magnetic
```

```
fields with respect to rotor surface is equal to
 + %.f RPM \n",Nfr)
printf("\n (a.2) Speed of Backward Rotating magnetic
 fields with respect to rotor surface is equal to
 + %.f RPM \n",Nbr)
printf("\n (b.1) Speed of Forward Rotating magnetic
 fields with respect to stator surface is equal to
 + %.f RPM \n",Nfs)
printf("\n (b.2) Speed of Backward Rotating magnetic
 fields with respect to stator surface is equal
 to + %.f RPM \n",Nbs)
printf("\n  (c)  Speed of Backward Rotating magnetic
 fields with respect to stator for 50 percenatge
 slip is equal to  %.1f RPM \n",Nbs_new)
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED
 WRONGLY ( I verified by manual calculation )]\n"
 );
printf("\n      WRONGLY PRINTED ANSWERS ARE :- (a)
 Speed of Backward Rotating magnetic fields with
 respect to stator for 50 percenatge slip is equal
 to 0 RPM instead of %.1f RPM  \n ",Nbs_new);
```

Scilab code Exa 5.14 To find the speed of rotor

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.14

clear ; clc ; close ; // Clear the work space and
```

```
console


// GIVEN DATA

f = 50;                // Stator Frequency of Inductor Motor in Hertz
fr = 10;               // Rotor Frequency of Inductor Motor in Hertz
p = 2;                 // Number of poles


// CALCULATIONS

Ns = (120*f)/p;         // Synchronous Speed of Induction Motor in RPM
s = fr/f;               // Slip of the Induction Motor
Nr = (1-s)*Ns;          // Rotor Speed of the Induction Motor


// DISPLAY RESULTS

disp("EXAMPLE : 5.14 : SOLUTION :-");
printf("\n (a) Rotor Speed of Induction Motor, Nr = %.f RPM \n",Nr)
```

Scilab code Exa 5.15 To find equivalent circuit parameters current pf torque output power

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
```

```
// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.15

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

printf("\n EXAMPLE : 5.15 : \n\n          Given Data
    No-load test : 440V, 30A, 4.5KW \n");
printf("\n          Blocked rotor test :      90V,50
   Hz, 120A, 16KW \n");
m = 3;                              // Total Number of
   phase in Induction Motor
p = 6;                              // Total number of
   Poles of Induction Motor
V = 440;                            // Operating voltage
    of the Induction motor in Volts
out_hp = 100;                       // Output of the
   Induction motor in Horse-Power
R = 0.15;                           // Average dc
   Resistance in Ohms
Wsc = 16000;                        // Power at Blocked
   Rotor test in Watts
Vsc = 90;                           // Voltage at
   Blocked Rotor test in Volts
Isc = 120;                          // Current at
   Blocked Rotor test in Amphere
W0 = 4500;                          // Power at No-load
   test in Watts
V0 = 440;                           // Voltage at No-
   load test in Volts
I0 = 30;                            // Current at No-
   load test in Amphere
s = 0.05;                           // Slip
```

```
f = 50;                            // Frequency in Hertz


// CALCULATIONS

R1 = R/2;                                          // DC winding Resistance per phase in Ohms
Rac = Wsc/(3*Isc^2);                               // AC Resistance referred to stator from locked rotor test at supply frequency in Ohms
R_2 = Rac - R1;                                    // Per phase Rotor Resistance to Stator in Ohms
Zsc = Vsc/(sqrt(3)*Isc);                           // Per phase Impedance from locked rotor test in Ohms
Xs = sqrt((Zsc^2)-(Rac^2));                        // Per phase leakage Reactance referred to stator in Ohms
theta_0 = acosd(W0/(V0*I0*sqrt(3)));               // No-load power factor angle in degree
Im = I0*sind(theta_0);                             // Reactive component of no-load current in Amphere
Xm = V0/(Im*sqrt(3));                              // Magnetizing Reactance in Ohms
Pc = W0 - 3*I0^2*R1;                               // Total Core loss in Watts
Rc = (V0/sqrt(3))^2*(3/Pc);                        // Per phase core loss Resistance in Watts
Vph = V0/sqrt(3);                                  // Per phase Voltage in Volts
Ic = Vph/Rc;                                       // Core loss current in Amphere
I_m = Vph/(%i * Xm);                               // Magnetizing Current in Amphere
I_o = Ic + I_m;                                    // No-load current in Amphere
I_2 = Vph/(R1+(R_2/s)+(%i*Xs));                    //
```

```
Current in Amphere
I1 = I_o + I_2;                                    // Input Current in Amphere
Pf = cosd(atand(imag(I1)/real(I1)));               // Power factor
P1 = (3*(abs(I_2)^2*R_2)/s)/1000;                  // 3-phase air gap power or Rotor intake Power in Kilo-Watts
Po = P1*(1-s);                                     // Output Power in Kilo-Watts
Ws = 2*%pi*((120*f/p)*(1/60));                     // Angular Roatation in Radians per Seconds
T = P1*1000/Ws;                                    // Torque in Newton-Meter


// DISPLAY RESULTS

disp(" SOLUTION :-");
printf("\n (a.1)  DC winding Resistance per phase, R1 = %.3f Ohms \n",R1)
printf("\n (a.2)  AC Resistance referred to stator from locked rotor test at supply frequency  = %.4f Ohms \n",Rac)
printf("\n (a.3)  Per phase Rotor Resistance to Stator, R2 = %.4f Ohms \n",R_2)
printf("\n (a.4)  Per phase Impedance from locked rotor test,Zsc = %.3f Ohms \n",Zsc)
printf("\n (a.5)  Per phase leakage Reactance referred to stator, Xs = %.4f Ohms \n",Xs)
printf("\n (a.6)  No-load power factor angle, theta_O = %.2f Degree \n",theta_0)
printf("\n (a.7)  Reactive component of no-load current, Im = %.1f A \n",Im)
printf("\n (a.8)  Magnetizing Reactance, Xm = %.2f Ohms \n",Xm)
printf("\n (a.9)  Total Core loss, Pc = %.1f W \n",Pc)
```

```
printf("\n (a.10) Per phase core loss Resistance, Pc = %.f Ohms \n",Rc)
printf("\n (a.11) Per phase Voltage, Vph = %.f V \n",Vph)
printf("\n (a.12) Core loss current, Ic = %.2f < %.f A \n",abs(Ic),atand(imag(Ic),real(Ic)))
printf("\n (a.13) Magnetizing Current, Im = %.1f < %.f A \n",abs(I_m),atand(imag(I_m),real(I_m)))
printf("\n (a.14) No-load current, I0 = %.2f < %.2f A \n",abs(I_o),atand(imag(I_o),real(I_o)))
printf("\n (a.15) Current, I2 = %.2f < %.2f A \n",abs(I_2),atand(imag(I_2),real(I_2)))
printf("\n   (b)  Input current, I1 = %.2f < %.2f A \n",abs(I1),atand(imag(I1),real(I1)))
printf("\n   (c)  Power Factor, Pf = %.4f Lagging \n",Pf)
printf("\n   (d)  Output Power, P0 = %.1f kW \n",Po)
printf("\n   (e)  Torque, T = %.2f NM \n",T)
```

Scilab code Exa 5.17 To find per phase core loss resistance and magnetizing reactance of equivalent circuit

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.17

clear ; clc ; close ; // Clear the work space and console

```

```

// GIVEN DATA

printf("\n EXAMPLE : 5.17 : \n\n            Given Data No-load test : 440V, 3.0A, 500KW, 50Hz \n");
printf("\n            Blocked rotor test at rated frequency : 110V, 18A, 2500W, 50Hz \n");
printf("\n            DC test on Stator per phase : 10 V, 15A \n");
m = 3;                          // Total Number of phase in Induction Motor
p = 4;                          // Total number of Poles of Induction Motor
f = 50;                         // Frequency in Hertz
V = 440;                        // Operating Voltage of the Inductuon Motor
out_hp = 20;                    // Motor Power Rating in Horse-Power
Vdc = 10;                       // DC Voltage in Volts
Idc = 15;                       // DC Current in Amphere
Wsc = 2500;                     // Power at Blocked Rotor test rated frequency in Watts
Wsc_red = 2050;                 // Power at Blocked Rotor test reduced frequency in Watts
Vsc = 110;                      // Voltage at Blocked Rotor test rated frequency in Volts
Isc = 18;                       // Current at Blocked Rotor test rated frequency in Amphere
Wo = 500;                       // Power at No-load test in Watts
Vo = 440;                       // Voltage at No-load test in Volts
Io = 4.0;                       // Current at No-load test in Amphere
fsc = 50;                       // Rated Frequency
```

```
at blocked rotor test in Hertz
fo = 50;                              // Rated Frequency at no-load test in Hertz
fsc1 = 15;                            // Reduced Frequency at blocked rotor in Hertz
Pfw = 200;                            // Friction and Windage loss in Watts


// CALCULATIONS

R1dc = Vdc/Idc;                                // DC winding Resistance per phase in Ohms
Rac = Wsc/(3*Isc^2);                           // AC Resistance from Locked rotor test at supply frequency
Rac_red = Wsc_red/(3*Isc^2);                   // AC Resistance from Locked rotor test at reduced frequency
R1ac = (Rac/Rac_red)*R1dc;                     // Corrected Value of AC stator winding Resistance in Ohms
R2dc = Rac_red - R1dc;                         // Second rotor parameter, rotor resistance referred to stator is at low frequency in Ohms
Zsc = Vsc/(sqrt(3)*Isc);                               // Per phase Impedance from locked rotor test at power frequency in Ohms
Xs = sqrt((Zsc^2)-(Rac^2));                            // Per phase leakage Reactance referred to stator in Ohms
theta_0 = acosd(Wo/(Vo*Io*sqrt(3)));                   // No-load power factor angle in degree
Im = Io*sind(theta_0);                                 // Reactive component of no-load current in Amphere
Xm = Vo/(Im*sqrt(3));                                  // Magnetizing Reactance in Ohms
Pc = Wo - 3*Io^2*R1ac-Pfw;                             // Total Core loss in Watts
```

```
Rc = (Vo/sqrt(3))^2*(3/Pc);                    // Per phase core loss Resistance in Watts


// DISPLAY RESULTS

disp(" SOLUTION :-");
printf("\n (a)  Magnetizing Reactance of Equivalent circuit , Xm = %.1f Ohms \n",Xm)
printf("\n (b) Per phase core loss Resistance , Pc = %.f Ohms \n",Rc)
```

Scilab code Exa 5.18 To find current pf torque output power

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.18

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA
// From Previous problem data (Example 5.17)

R1ac = 0.8127;                          // Corrected Value of AC stator winding Resistance in Ohms
R2dc = 1.4433;                          // Second rotor parameter , rotor resistance referred to stator is
```

```
    at low frequency in Ohms
Xs = 2.42;                          // Per phase leakage Reactance referred to stator in Ohms
Xm = 64.4;                          // Magnetizing Reactance in Ohms
Rc = 742;                           // Per phase core loss Resistance in Watts
s = 0.035;                          // Slip
m = 3;                              // Total Number of phase in Induction Motor
p = 4;                              // Total number of Poles of Induction Motor
f = 50;                             // Frequency in Hertz
V = 440;                            // Operating Voltage of the Inductuon Motor
out_hp = 20;                        // Motor Power Rating in Horse-Power


// CALCULATIONS

Vph = V/sqrt(3);                                 // Per phase Voltage in Volts
Ic = Vph/Rc;                                     // Core loss current in Amphere
I_m = Vph/(%i * Xm);                             // Magnetizing Current in Amphere
I_o = Ic + I_m;                                  // No-load current in Amphere
I_2 = Vph/(R1ac+(R2dc/s)+(%i*Xs));               // Current in Amphere
I1 = I_o + I_2;                                  // Input Current in Amphere
Pf = cosd(atand(imag(I1)/real(I1)));             // Power factor
P1 = 3*(abs(I_2)^2*R2dc)/s;                      // 3-phase air gap power or Rotor intake Power in Watts
```

```
Po = P1*(1-s);                                          // Output Power in Watts
Ws = 2*%pi*((120*f/p)*(1/60));                          // Angular Roatation in Radians per Seconds
T = P1/Ws;                                              // Torque in Newton-Meter


// DISPLAY RESULTS

disp("EXAMPLE : 5.18 : SOLUTION :-");
printf("\n  (a)  Input current , I1 = %.2f < %.2f A \n",abs(I1),atand(imag(I1),real(I1)))
printf("\n  (b)  Power Factor , Pf = %.3f Lagging \n ",Pf)
printf("\n  (c)  Output Power , P0 = %.2f W \n",Po)
printf("\n  (d)  Torque , T = %.2f NM \n",T)
printf("\n\n   [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n      WRONGLY PRINTED ANSWERS ARE :- (a) T = 4340.82 Nm instead of %.2f Nm \n ",T);
printf("\n\n        IN TEXT BOOK, CALCULATION OF TORQUE IS NOT DONE \n ");
```

Scilab code Exa 5.19 To find input line current pf torque Hp output and efficiency

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES
```

```

// EXAMPLE : 5.19

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

s = 0.05;                              // Slip
m = 3;                                 // Total Number of
    phase in Induction Motor
p = 4;                                 // Total number of
    Poles of Induction Motor
f = 50;                                // Frequency in
    Hertz
V = 440;                               // Operrating
    Voltage of the Inductuon Motor
R1 = 0.10;                             // Circuit Parameter
     in Ohms
R2 = 0.11;                             // Circuit Parameter
     in Ohms
X1 = 0.35;                             // Circuit Parameter
     in Ohms
X2 = 0.40;                             // Circuit Parameter
     in Ohms
pf = 0.2;                              // Power factor (
    Lagging)
Pr = 900;                              // Rotational Loss
    in Watts
Psc = 1000;                            // Stator core loss
    in Watts
I = 15;                                // Line current
    draws by the motor in Amphere


// CALCULATIONS

```

```
Vph = V/sqrt(3);                                  // Per phase Voltage in Volts
I_2 = Vph/(R1+(R2/s)+(%i*(X1+X2)));               // Current in Amphere
Io = I * exp(-( %i * acosd(pf) * %pi/180));       // No-load current in Amphere
I1 = Io + I_2;                                    // Input line Current in Amphere
PF = cosd(atand(imag(I1)/real(I1)));              // Power factor
Ws = 2*%pi*((120*f/p)*(1/60));                    // Angular Roatation in Radians per Seconds
Pg = (3*(abs(I1)^2*R2))/s;                        // 3-phase air gap power or Rotor intake Power in Watts
T = Pg/Ws;                                        // Torque in Newton-Meter
Po = Pg*(1-s)-Pr;                                 // Output Power in Watts
Po_HP = Po/746;                                   // Output Power in Horse-Power
eta = (Po/(Po+Psc+Pr))*100;                       // Efficiency in Percentage


// DISPLAY RESULTS

disp("EXAMPLE : 5.19 : SOLUTION :-");
printf("\n    (a)  Input line current , I1 = %.1f < %.2f A \n",abs(I1),atand(imag(I1),real(I1)))
printf("\n    (b)  Power Factor , Pf = %.4f Lagging \n",PF)
printf("\n    (c)  Output Power , P0 = %.1f HP \n",Po_HP)
printf("\n    (d)  Torque , T = %.2f Nm \n",T)
printf("\n    (e)  Efficiency , eta = %.1f Percenatge \n",eta)
printf("\n\n     [ TEXT BOOK SOLUTION IS PRINTED
```

```
WRONGLY ( I verified by manual calculation )]\n");
printf("\n        WRONGLY PRINTED ANSWERS ARE :- (a) I1 = 114.2<-24.68 A instead of %.1f<%.2f A \n ",abs(I1),atand(imag(I1),real(I1)));
printf("\n                                        (b) T = 548.24 Nm instead of %.2f Nm \n ",T);
printf("\n                                        (c) Po = 108.4 HP instead of %.1f HP \n ",Po_HP);
```

Scilab code Exa 5.20 To find input line current pf torque Hp output efficiency

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.20

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

m = 3;                              // Total Number of phase in Induction Motor
p = 6;                              // Total number of Poles of Induction Motor
f = 50;                             // Frequency in Hertz
V = 440;                            // Operating Voltage
```

```
of the Inductuon Motor
R1 = 0.25;                        // Circuit Parameter in Ohms
R2 = 0.25;                        // Circuit Parameter in Ohms
X1 = 0.75;                        // Circuit Parameter in Ohms
X2 = 0.75;                        // Circuit Parameter in Ohms
Xm = 1000;                        // Circuit Parameters in Ohms
Rc = 100;                         // Circuit Parameters in Watts
s = 0.025;                        // Slip
Pr = 450;                         // Rotational Loss in Watts
Psc = 800;                        // Stator core loss in Watts


// CALCULATIONS

Vph = V/sqrt(3);    // Per phase Voltage in Volts
I_2 = Vph/(R1+(R2/s)+(%i*(X1+X2)));    // Current in Amphere
Ic = Vph/Rc;                                  // Core loss current in Amphere
I_m = Vph/(%i * Xm);                          // Magnetizing Current in Amphere
I_o = Ic + I_m;                               // No-load current in Amphere
I1 = I_o + I_2;                               // Input Current in Amphere
Pf = cosd(atand(imag(I1)/real(I1)));          // Power factor
Ns = (120*f)/p;                               // Synronous Speed in RPM
```

```
Pg = 3*(abs(I_2)^2*R2)/s;                       // 3-phase air gap power or Rotor intake Power in Watts
Pm = Pg*(1-s);                                  // Output Power in Watts
Ws = 2*%pi*Ns*(1/60);                           // Angular Roatation in Radians per Seconds
T = Pg/Ws;                                      // Torque in Newton-Meter
Po = Pm-Pr;                                     // Output Power in Watts
Po_HP = Po/746;                                 // Output Power in Horse-Power
eta = (Po/(Po+Psc+Pr))*100;                     // Efficiency in Percentage


// DISPLAY RESULTS

disp("EXAMPLE : 5.20 : SOLUTION :-");
printf("\n   (a)  Input line current , I1 = %.f < %.2f A \n",abs(I1),atand(imag(I1),real(I1)))
printf("\n   (b)  Power Factor , Pf = %.4f Lagging \n",Pf)
printf("\n   (c)  Output Power , P0 = %.2f HP \n",Po_HP)
printf("\n   (d)  Torque , T = %.1f Nm \n",T)
printf("\n   (e)  Efficiency , eta = %.1f Percenatge \n",eta)
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n      WRONGLY PRINTED ANSWERS ARE :-  (a) I1 = 26.8-j3.584 {27<-7.62} A in instead of (%.1f)+(j%.3f) {%.f<%.2f} A \n ",real(I1),imag(I1),abs(I1),atand(imag(I1),real(I1)));
printf("\n                                      (b) pf = 0.9885 Lagging instead of %.4f Lagging \n ",
```

```
    Pf);
```

Scilab code Exa 5.21 To find torque and output power

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.21

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

m = 3;                              // Total Number of
    phase in Induction Motor
p = 4;                              // Total number of
    Poles of Induction Motor
s = 0.05;                           // Slip
f = 50;                             // Frequency in
    Hertz
Tm = 500;                           // Maximum Torque in
     Newton-Meter
Tst = 200;                          // Starting Torque
    in Newton-Meter
sst = 1.0;                          // Starting Slip


// CALCULATONS
```

```

p1 = poly([1 -5 1],'sm','c');    // Slip at Maximum Torque (obtained from Equation  Tst = (2*Tm)/((sst/sm)+(sm+sst))
a = roots(p1);                   // Value of slip at Maximum Torque (obtained from Equation  Tst = (2*Tm)/((sst/sm)+(sm+sst))
sm = a(2,1);                     // Slip at Maximum Torque (obtained from Equation  Tst = (2*Tm)/((sst/sm)+(sm+sst)) { 1st root is 4.8 so its out of range because slip value is lies between 0-1 so its neglected and second root will be slip }
T = (2*Tm)/((s/sm)+(sm/s));      // Torque at 0.05 slip
Ns = (120*f)/p;                  // Synchronous Speed in RPM
Wr = (2*%pi)*(1-s)*(Ns/60);      // Angular Velocity in Radians-per-Second
P = T * Wr;                      // Power Output in Watts
P_HP = P/746;                    // Power Output in Horse-Power


// DISPLAY RESULTS

disp("EXAMPLE : 5.21 : SOLUTION :-");
printf("\n (a) Torque at 0.05 slip , T = %.2f Nm \n",T)
printf("\n (b) Power Output at 0.05 slip , P = %.1f W = %.2f HP \n",P,P_HP)
```

Scilab code Exa 5.22 To find current pf and torque

```

```

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.22

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

Wsc = 1000;                          // Power at Blocked
    Rotor test in Watts
Vsc = 56;                            // Voltage at Blocked
    Rotor test in Volts
Isc = 18;                            // Current at Blocked
    Rotor test in Amphere
Woc = 52;                            // Power at No-load
    test in Watts
Voc = 220;                           // Voltage at No-load
    test in Volts
Ioc = 2.6;                           // Current at No-load
    test in Amphere
m = 3;                               // Total Number of
    phase in Induction Motor
p = 4;                               // Total number of
    Poles of Induction Motor
V = 220;                             // Operating voltage
    of the Induction motor in Volts
f = 50;                              // Frequency in Hertz
s = 0.05;                            // Slip
R = 0.65;                            // Per phase stator
    Resistance in Ohms

```

```

// CALCULATIONS

Vph = Voc/sqrt(3);                                  // Per phase Voltage in Volts
Wo = Woc/m;                                         // Per phase No-load loss in Watts
theta_0 = acosd(Wo/(Voc*Ioc*sqrt(3)));              // No-load power factor angle in degree
VSC = Vsc/sqrt(3);                                  // Per phase locked rotor Voltage in Volts
WSC = Wsc/m;                                        // Per phase locked rotor loss in Watts
theta_sc = acosd(WSC/(VSC*Isc));                    // No-load power factor angle in degree
ISC = Isc*(Voc/Vsc);                                // locked rotor current at full Voltage in Amphere
Re = WSC/Isc^2;                                     // Resistance in Ohms
R1 = R*1.1;                                         // Per phase AC stator Resistance in Ohms
R_2 = Re - R1;                                      // Per phase rotor Resistance in Ohms
Zsc = VSC/Isc;                                      // Per phase impedance in Ohms
Xs = sqrt((Zsc^2)-(Re^2));                          // Leakage Reactance in Ohms
I_2 = (Voc/sqrt(3))/sqrt((R1+(R_2/s))^2+(Xs^2)); // Current in Amphere
pf = cosd(atand(Xs/(R1+(R_2/s))));                 // Power Factor
Ws = 2*%pi*((120*f/p)*(1/60));                      // Rotational Speed in Radians per Seconds
Pg = (3*(abs(I_2)^2*R_2))/s;                        // 3-phase air gap power or Rotor intake Power in Watts
T = Pg/Ws;                                          // Torque in Newton-Meter
```

```
// CALCULATIONS OR DATA OBTAINED FROM CIRCLE DIAGRAM FIGURE 5.35 and PAGE NO:-303
OA = 2.60;                                              // Correspounding Current in Amphere at 87' from Y-axis (from Circle diagram)
OE = 70.70;                                             // Correspounding Current in Amphere at 55' from Y-axis (from Circle diagram)
OP = 17.77;                                             // Current in Amphere (from Circle diagram)
OV = Voc/sqrt(3);                                       // Phase Voltage in No-load test or value obatined from circle diagram in Volts
PK = 11.6;                                              // Correspounding Value from Circle diagram
JK = 0.8;                                               // Correspounding Value from Circle diagram
PJ = 10.8;                                              // Correspounding Value from Circle diagram
PM = 11.6;                                              // Correspounding Value from Circle diagram
Pir = 3*OV*PK;                                          // Total Rotor intake in Watts
Plr = 3*OV*JK;                                          // Total Rotor loss in Watts
Po = 3*OV*PJ;                                           // Total Mechanical power output in Watts
T_c = (3*OV*PK)/Ws;                                     // Total Torque in Newton-Meter
s_c = JK/PK;                                            // Slip obtained from Circle diagram
s_pc = 100*s_c;                                         // Slip in percentage
eta = 100*(PJ/PM);                                      // Eifficiency in Percentage


// DISPLAY RESULTS
```

```

disp("EXAMPLE : 5.22 : SOLUTION :-");
printf("\n    (a)  Input line current , I2 = %.2f A \n
   ",I_2)
printf("\n    (b)  Power Factor , Pf = %.3f \n",pf)
printf("\n    (c)  Torque , T = %.2f Nm \n",T)
printf(" \n Verification Results from Circle Diagram
    :-\n");
printf("\n    (a)  Efficency , eta = %.2f Percent \n",
   eta)
printf("\n    (b)  slip , s = %.3f = %.f percent \n",
   s_c,s_pc)
printf("\n    (c)  Torque , T = %.2f Nm \n",T_c)
```

Scilab code Exa 5.23 To find initial starting current and torque in each cases

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.23

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

R1 = 0.2;                       // Circuit Parameter
   in Ohms
```

```
R2 = 0.4;                              // Circuit Parameter
    in Ohms
X1 = 1.0;                              // Circuit Parameter
    in Ohms
X2 = 1.5;                              // Circuit Parameter
    in Ohms
m = 3;                                  // Total Number of
    phase in Induction Motor
p = 2;                                  // Total number of
    Poles of Induction Motor
f = 50;                                 // Frequency in
    Hertz
V = 440;                                // Operating Voltage
     of the Inductuon Motor


// CALCULATIONS

Ws = 2*%pi*f;                                        //
    Synchronous angular speed in Radians per second
Z = (R1+R2)+((%i)*(X1+X2));                          // At
    slip s=1, the impedance seen from the terminals
    in Ohms
s = 1;                                               // Slip

// For Case(a) Winding is connected in star

Isy_a = V/(abs(Z)*sqrt(3));                           //
    Current in Amphere
Tsy_a = (3*Isy_a^2*R2)/(s*Ws);                        //
    Torque in Newton-Meter

// Winding is connected in delta

Isd_a = (V*sqrt(3))/abs(Z);                           //
    Current in Amphere
Tsd_a = (3*(Isd_a/sqrt(3))^2*R2)/(s*Ws);              //
    Torque in Newton-Meter
```

```
I_R = Isd_a/Isy_a;                              // Ratio of the line current
T_R = Tsd_a/Tsy_a;                              // Ratio of the Torque

// For Case(b) Machine is started using auto−transfromer and voltage is 50% reduced

Isy_b = (0.5*V)/(abs(Z)*sqrt(3));                   // Current in Amphere when Winding is connected star
Tsy_b = (3*Isy_b^2*R2)/(s*Ws);                      // Torque in Newton−Meter when Winding is connected star
Isd_b = (0.5*V*sqrt(3))/abs(Z);                     // Current in Amphere when Winding is connected delta
Tsd_b = (3*(Isd_b/sqrt(3))^2*R2)/(s*Ws);            // Torque in Newton−Meter when Winding is connected delta

// For Case(c) Both Voltage and Frequency are reduced to 50%

f_new = (10/100)*f;                                 // New Frequency
Ws_c = 2*%pi*f_new;                                 // Synchronous angular speed in Radians per second
Z_c = ((R1+R2)+((%i)*(X1+X2))*(f_new/f));           // At slip s=1, the impedance seen from the terminals in Ohms
Isy_c = (0.1*V)/(abs(Z_c)*sqrt(3));                 // Current in Amphere when Winding is connected star
Tsy_c = (3*Isy_c^2*R2)/(s*Ws_c);                    // Torque in Newton−Meter when Winding is connected star
Isd_c = (0.1*V*sqrt(3))/abs(Z_c);                   //
```

```
Current in Amphere when Winding is connected delta
Tsd_c = (3*(Isd_c/sqrt(3))^2*R2)/(s*Ws_c);          // Torque in Newton-Meter when Winding is connected delta


// DISPLAY RESULTS

disp("EXAMPLE : 5.23 : SOLUTION :-");
printf("\n For Case (a.1) Winding is connected in star \n");
printf("\n (a.1.1) Per phase impedance seen from the terminals in Ohms, Z = %.3f < %.1f Ohms \n",abs(Z),atand(imag(Z),real(Z)));
printf("\n (a.1.2) Initial Starting Current , Isy = %.2f A \n",Isy_a)
printf("\n (a.1.3) Starting Torque , Tsy = %.1f Nm \n",Tsy_a)
printf("\n For Case (a.2) Winding is connected in delta \n" );
printf("\n (a.2.1) Initial Starting Current , Isd = %.2f A \n",Isd_a)
printf("\n (a.2.2) Starting Torque , Tsd = %.2f Nm \n",Tsd_a)
printf("\n For Case (b) Machine is started using auto-transfromer and voltage is 50 percentage reduced :- (b.1) Winding is connected in star \n")
printf("\n (b.1.1) Per phase impedance seen from the terminals in Ohms, Z = %.3f<%.1f Ohms \n",abs(Z),atand(imag(Z),real(Z)));
printf("\n (b.1.2) Initial Starting Current , Isy = %.1f A \n",Isy_b)
printf("\n (b.1.3) Starting Torque , Tsy = %.2f Nm \n",Tsy_b)
printf("\n For Case (b.2) Winding is connected in delta \n" );
```

```
printf("\n (b.2.1) Initial Starting Current , Isd = %.2f A \n",Isd_b)
printf("\n (b.2.2) Starting Torque , Tsd = %.f Nm \n",Tsd_b)
printf("\n For Case (c) Both Voltage and Frequency are reduced to 50 percentage :- (c.1) Winding is connected in star \n ");
printf("\n (c.1.1) Per phase impedance seen from the  terminals in Ohms, Z =  %.2f<%.2f Ohms \n",abs(Z_c),atand(imag(Z_c),real(Z_c)));
printf("\n (c.1.2) Initial Starting Current , Isy = %.2f A \n",Isy_c)
printf("\n (c.1.3) Starting Torque , Tsy = %.2f Nm \n",Tsy_c)
printf("\n For Case (c.2) Winding is connected in delta \n" );
printf("\n (c.2.1) Initial Starting Current , Isd = %.2f A \n",Isd_c)
printf("\n (c.2.2) Starting Torque , Tsd = %.2f Nm \n",Tsd_c)
printf('\nComparing the Calculated values of starting current and torque eid rated frequency and rated voltage\n")
printf("\n                                    star                    delta\n")
printf("\n                             440V,50Hz    44V,5Hz         440V,50Hz    44V,5Hz \n")
printf("\n starting current      %.2f A      %.f A          %.f A         %.2f A \n",Isy_a,Isy_c,Isd_a,Isd_c)
printf("\n starting Torque       %.1f Nm    %.2f Nm          %.f Nm        %.2f Nm \n",Tsy_a,Tsy_c,Tsd_a,Tsd_c)
printf("\n\n     [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n       WRONGLY PRINTED ANSWERS ARE :- For Case (a.2) Winding is connected in delta :- (a)
```

```
    Initial Starting Current Isy = 254.01 A instead
    of %.2f A \n\n ",Isd_a);
```

Scilab code Exa 5.24 To find initial starting current during DOL starting

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.24

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

m = 3;                              // Total Number of
    phase in Induction Motor
f = 50;                             // Frequency in
    Hertz
V = 440;                            // Operating voltage
     of the Induction Motor in Volts
R1 = 0.2;                          // Circuit Parameter
    in Ohms
R2 = 0.4;                          // Circuit Parameter
    in Ohms
X1 = 1.0;                          // Circuit Parameter
    in Ohms
X2 = 1.5;                          // Circuit Parameter
    in Ohms
```

```
Rc = 150;                           // Circuit Parameter in Ohms
Xm = 30;                            // Circuit Parameter in Ohms


// CALCULATIONS

V1 = V/sqrt(3);                     // Rated Voltage in Volts
Zdol = (R1+%i*X1)+(Rc*%i*Xm*(R2+%i*X2))/(Rc*%i*Xm+Rc*(R2+%i*X2)+(%i*Xm)*(R2+%i*X2));    // Equivalent impedance per phase in DOL starter in Ohms
I = V1/Zdol;                        // Starting Current in DOL starter in Amphere

// For Case(a) A per Phase resistance of 0.5 Ohms is added in Series with the stator circuit

Zsr =  (0.5+R1+%i*X1)+((Rc*%i*Xm*(R2+%i*X2))/((Rc*%i*Xm+Rc*(R2+%i*X2)+(%i*Xm)*(R2+%i*X2))));    // Total impedance seen from the terminals in Ohms
Isr = V1/Zsr;                       // Starting Current in DOL starter in Amphere

// For Case(b) A per Phase resistance of 0.5 Ohms is added in Series with the rotor circuit here assumed that stator to rotor turn ratio is 1.0

Zrr =  (R1+%i*X1)+((Rc*%i*Xm*(0.5+R2+%i*X2))/(Rc*%i*Xm+Rc*(0.5+R2+%i*X2)+(%i*Xm)*(0.5+R2+%i*X2)));    // Total impedance seen from the terminals in Ohms
Irr = V1/Zrr;
```

```
    // Starting Current in DOL starter in Amphere

// For Case(c) When applied Voltage reduced to 50%

I_c = (0.5*V1)/Zdol;
    // Starting Current in DOL starter in Amphere

// For Case(d) When Motor is supplied by reduced Voltage of 44V ( Voltage is reduced by 10%) and the reduced frequency is 5Hz

f_n = 5;                                    // Reduced Frequency in Hertz
X1_n = (f_n/f)*X1;                          // Changed Circuit Parameter in Ohms
X2_n = (f_n/f)*X2;                          // Changed Circuit Parameter in Ohms
Xm_n = (f_n/f)*Xm;                          // Changed Circuit Parameter in Ohms
Zdol_n = (R1+%i*X1_n)+((Rc*%i*Xm_n*(R2+%i*X2_n))/(Rc*%i*Xm_n+Rc*(R2+%i*X2_n)+(%i*Xm_n)*(R2+%i*X2_n)));                 // Equivalent impedance per phase in DOL starter in Ohms
I_n = (V1*0.1)/Zdol_n;                      // Starting Current in DOL starter in Amphere
Ratio = abs(I_n)/abs(I);                    // Ratio of the Starting Current witha the rated Voltage and frequency to the reduced Voltage and frequency


// DISPLAY RESULTS

disp("EXAMPLE : 5.24 : SOLUTION :-");
printf("\n Normal Initial Starting Current in DOL starter , I = %.1f <%.1f A \n",abs(I),atand(imag(I),real(I)))
printf("\n For Case(a) A per Phase resistance of 0.5
```

```
 Ohms is added in Series with the stator circuit \n")
printf("\n                Initial Starting Current in DOL starter , I = %.1f <%.2f A \n",abs(Isr),atand(imag(Isr),real(Isr)))
printf("\n  For Case(b) A per Phase resistance of 0.5 Ohms is added in Series with the rotor circuit \n")
printf("\n                Initial Starting Current in DOL starter , I = %.2f <%.1f A \n",abs(Irr),atand(imag(Irr),real(Irr)))
printf("\n  For Case(c) When applied Voltage reduced to 50 percentage \n")
printf("\n                Initial Starting Current in DOL starter , I = %.2f <%.1f A \n",abs(I_c),atand(imag(I_c),real(I_c)))
printf("\n  For Case(d) When Motor is supplied by reduced Voltage of 44V ( Voltage is reduced by 10 percenatge ) and the reduced frequency is 5Hz \n")
printf("\n                Initial Starting Current in DOL starter , I = %.1f <%.1f A \n",abs(I_n),atand(imag(I_n),real(I_n)))
printf("\n By reducing volatge as well as the frequency , the peak starting current at the instant os starting is reduced by a fector of %.4f of the starting current with the reted volatge and frequency \n",Ratio)
printf("\n\n     [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n       WRONGLY PRINTED ANSWERS ARE :-  For Case(d) When Motor is supplied by reduced Voltage of 44V ( Voltage is reduced by 10 percenatge ) and the reduced frequency is 5Hz,  I = 24.1 < 25.6 A instead of %.1f < (%.2f) A \n ",abs(I_n),atand(imag(I_n),real(I_n)));
printf("\n  Ratio of the Starting Current with the
```

```
    rated Voltage and frequency to the reduced
    Voltage and frequency , Ratio = 0.2518 instead of
    %.4f \n ",Ratio);
```

Scilab code Exa 5.25 To find range of speed

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.25

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

m1 = 3;                                  // Total Number of
    phase in 1st Induction Motor
p1 = 6;                                  // Total number of
    Poles of 1st Induction Motor
f = 50;                                  // Frequency in
    Hertz
m2 = 3;                                  // Total Number of
    phase in 2nd Induction Motor
p2 = 10;                                 // Total number of
    Poles of 2nd Induction Motor


// CALCULATIONS
```

```
Ns1 = (120*f)/p1;                              // Synchronous speed of 1st Induction Motor in RPM
Ns2 = (120*f)/p2;                              // Synchronous speed of 2nd Induction Motor in RPM
Nscu = (120*f)/(p1+p2);                        // Speed during cumalative casade in RPM
Ndiff = (120*f)/(p2-p1);                       // Speed during cumalative casade in RPM


// DISPLAY RESULTS

disp("EXAMPLE : 5.25 : SOLUTION :-");
printf("\n (a)  Range of speed is %.f - %.f - %.f - %.f RPM \n",Nscu,Ns2,Ns1,Ndiff)
```

Scilab code Exa 5.26 To find current and torque in each cases

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.26

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

```

```
m = 3;                              // Total Number of phase in Induction Motor
p = 4;                              // Total number of Poles of Induction Motor
f = 50;                             // Frequency in Hertz
V = 440;                            // Operating Voltage of the Inductuon Motor
R1 = 0.25;                          // Circuit Parameter in Ohms
R2 = 0.5;                           // Circuit Parameter in Ohms
X1 = 1.5;                           // Circuit Parameter in Ohms
X2 = 1.5;                           // Circuit Parameter in Ohms


// CALCULATIONS

Vph = V/sqrt(3);                    // Per phase Voltage in Volts
Ns = (120*f)/p;                     // Synchoronous Speed in RPM
Ws = (2*%pi*Ns)/60;                 // Roatation Speed in Radians per Seconds

// For Case (a) Machine running at, N = 1400 RPM

N_a = 1400;                         // Machine running in RPM
s_a = (Ns-N_a)/Ns;                  // Slip
I_2_a = Vph/(R1+(R2/s_a)+(%i*(X1+X2)));   // Rotor per phase Current referred to the stator side in Amphere
Pg_a = 3*(abs(I_2_a)^2*R2)/s_a;     // 3-phase air gap power or Rotor intake Power in
```

```
Watts
T_a = Pg_a/Ws;  // Torque in Newton-Meter

// For Case (b) Machine running at, N = 1600 RPM

N_b = 1600;  // Machine running in RPM
s_b = (Ns-N_b)/Ns;  // Slip
I_2_b = Vph/(R1+(R2/s_b)+(%i*(X1+X2)));  // Rotor per phase Current referred to the stator side in Amphere
Pg_b = 3*(abs(I_2_b)^2*R2)/s_b;  // 3-phase air gap power or Rotor intake Power in Watts
T_b = Pg_b/Ws;  // Torque in Newton-Meter

// For Case (b) Machine running at, N = -100 RPM

N_c = -100;  // Machine running in RPM
s_c = (Ns-N_c)/Ns;  // Slip
I_2_c = Vph/(R1+(R2/s_c)+(%i*(X1+X2)));  // Rotor per phase Current referred to the stator side in Amphere
Pg_c = 3*(abs(I_2_c)^2*R2)/s_c;  // 3-phase air gap power or Rotor intake Power in Watts
T_c = -Pg_c/Ws;  // Torque in Newton-Meter (minus sign because its counter opposing torque)

// DISPLAY RESULTS

```

```
disp("EXAMPLE : 5.26 : SOLUTION :-");
printf("\n For Case (a) Machine running at , N = 1400 RPM \n ")
printf("\n (a.1) Rotor per phase Current referred to the stator side , I2 = %.2f < %.2f A \n",abs(I_2_a),atand(imag(I_2_a),real(I_2_a)))
printf("\n (a.2) Developed Torque , T = %.2f Nm \n",T_a)
printf("\n For Case (b) Machine running at , N = 1600 RPM \n ")
printf("\n (a.1) Rotor per phase Current referred to the stator side , I2 = %.2f < %.2f A \n",abs(I_2_b),atand(imag(I_2_b),real(I_2_b)))
disp("   ( angle -157.52 + 180 = 22.48 ) ")
 printf("\n (a.2) Developed Torque , T = %.2f Nm \n",T_b)
printf("\n For Case (c) Machine running at , N = -100 RPM \n ")
printf("\n (c.1) Rotor per phase Current referred to the stator side , I2 = %.2f < %.2f A \n",abs(I_2_c),atand(imag(I_2_c),real(I_2_c)))
printf("\n (c.2) Developed Torque , T = %.2f Nm \n",T_c)
```

Scilab code Exa 5.27 To find slip at maximum torque and find torque at 5 percent and minus 5 percent of slip

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

```

```
// EXAMPLE : 5.27

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

m = 3;                                // Total Number of phase in Induction Motor
p = 2;                                // Total number of Poles of Induction Motor
f = 50;                               // Frequency in Hertz
V = 440;                              // Operating Voltage of the Inductuon Motor
R1 = 0.25;                            // Circuit Parameter in Ohms
R2 = 0.25;                            // Circuit Parameter in Ohms
X1 = 0.75;                            // Circuit Parameter in Ohms
X2 = 0.75;                            // Circuit Parameter in Ohms
out_hp = 50;                          // Output of the induction motor in HP


// CALCULATIONS

V1 = V/sqrt(3);                                                    // Phase Voltage in Volts
I = (out_hp*746)/(V*sqrt(3));                                      // Rated Current in Amphere
sm = R2/(sqrt(R1^2+(X1+X2)^2));                                    // Slip at Maximum
```

```
    torque both its in Positive and negative sign
Ws = 2*%pi*((120*f/p)*(1/60));                          // Angular Roatation in Radians per Seconds
Tm = (3*V1^2)/((2*Ws)*(R1+sqrt((R1^2)+(X1+X2)^2)));          // Maximum torque during motoring in Newton-Meter
Tg = -(3*V1^2)/((2*Ws)*(-R1+sqrt((R1^2)+(X1+X2)^2)));          // Maximum torque during generating in Newton-Meter

// For Case (a) slip = 0.05

s_a = 0.05;                                             // Slip
T_a = (2*Tm)/((s_a/sm)+(sm/s_a));                       // Torque in Newton-Meter

// For Case (b) slip = -0.05

s_b = -0.05;                                            // Slip
T_b = (2*Tg)/((s_b/sm)+(sm/s_b));                       // Torque in Newton-Meter


// DISPLAY RESULTS

disp("EXAMPLE : 5.27 : SOLUTION :-");
printf("\n Maximim Torque during Motoring , Tm = %.f N-m \n",Tm)
printf("\n Maximim Torque during Generating , Tm = %.2f N-m \n",Tg)
printf("\n For Case (a) slip = 0.05 \n ")
```

```
printf("\n (a.1) Torque, T = %.2f Nm \n",T_a)
printf("\n For Case (b) slip = -0.05 \n ")
printf("\n (b.1) Torque, T = %.2f Nm \n",T_b)
```

Scilab code Exa 5.28 To find starting torque and running torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.28

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

m = 3;                            // Total Number of
    phase in Induction Motor
p = 2;                            // Total number of
    Poles of Induction Motor
f = 50;                           // Frequency in
    Hertz
V = 440;                          // Operating Voltage
     of the Inductuon Motor in Volts
R0 = 0.5;                         // Circuit Parameter
     in Ohms
Ri = 0.05;                        // Circuit Parameter
     in Ohms
X0 = 0.2;                         // Circuit Parameter
```

```
in Ohms
Xi = 0.9;                       // Circuit Parameter in Ohms
s = 0.07;                       // Slip


// CALCULATIONS

Ws = 2*%pi*f;                   // Synchronous speed in Radins per second
v = V/sqrt(3);                  // Phase Voltage in Volts
Io = v/(R0+%i*X0);              // Starting Current in the outer cage in Amphere
Ii = v/(Ri+%i*Xi);              // Starting Current in the inner cage in Amphere
Tst = ((3*abs(Io)^2*R0)/Ws)+((3*abs(Ii)^2*Ri)/Ws);          // Starting torque i.e at standstill, s=1
Ios = v/((R0/s)+(%i*X0));       // Current in the outer cage at slip = 0.07
Iis = v/((Ri/s)+(%i*Xi));       // Current in the outer cage at slip = 0.07
T = ((3*abs(Ios)^2*R0)/(s*Ws))+((3*abs(Iis)^2*Ri)/(s*Ws));  // Starting torque at s=0.07 in Newton-Meter


// DISPLAY RESULTS

disp("EXAMPLE : 5.28 : SOLUTION :-");
printf("\n (a) Starting torque, Tst = %.2f Nm \n",
```

```
    Tst)
printf("\n (b) Running torque at slip = 0.07, T = %
    .2f Nm \n",T)
```

Scilab code Exa 5.29 To find running torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.29

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

p = 4;                              // Total number of
    Poles of Induction Motor
f = 50;                             // Frequency in
    Hertz
V = 440;                            // Operating Voltage
     of the Inductuon Motor in Volts
out = 25*1000;                      // Power rating of
    the Induction motor in Watts
R0 = 2.5;                           // Circuit Parameter
     in Ohms
Ri = 0.5;                           // Circuit Parameter
     in Ohms
X0 = 1.0;                           // Circuit Parameter
```

```
    in Ohms
Xi = 5.0;                          // Circuit Parameter in Ohms
Rc = 500;                          // Circuit Parameter in Ohms
R1 = 0.2;                          // Circuit Parameter in Ohms
Xm = 50;                           // Circuit Parameter in Ohms
X123 = 2.0;                        // Circuit Parameter in Ohms
s = 0.05;                          // Slip


// CALCULATIONS

Ws = (2*%pi*120*f)/(p*60);                    // Synchronous speed in Radins per second
Zo = (R0/s)+(%i*X0);                          // Outer cage impedance at slip = 0.05 in Ohms
Zi = (Ri/s)+(%i*Xi);                          // Inner cage impedance at slip = 0.05 in Ohms
Z = (R1+%i*X123)+((Zo*Zi)/(Zo+Zi));           // Total impdance in Ohms
I = V/Z;                                      // Current in the Cage winding in Amphere
Io = (I*((Zo*Zi)/(Zo+Zi)))/Zo;                // Current in the outer cage in Amphere
Ii = (I*((Zo*Zi)/(Zo+Zi)))/Zi;                // Current in the inner cage in Amphere
T = ((3*abs(Io)^2*R0)/(s*Ws))+((3*abs(Ii)^2*Ri)/(s*Ws));  // Starting torque in Newton-Meter
```

```


// DISPLAY RESULTS

disp("EXAMPLE : 5.29 : SOLUTION :-");
printf("\n (a) Torque at slip %.2f , T = %.2f Nm \n",s,T)
```

Scilab code Exa 5.30 To find input current pf and running torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.30

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

m = 1;                              // Total Number of phase in Induction Motor
p = 2;                              // Total number of Poles of Induction Motor
f = 50;                             // Frequency in Hertz
V = 220;                            // Operating Voltage of the Inductuon Motor in Volts
R1 = 10;                            // Circuit Parameter
```

```
in Ohms
R2 = 11;                          // Circuit Parameter in Ohms
X1 = 12;                          // Circuit Parameter in Ohms
X2 = 12;                          // Circuit Parameter in Ohms
Xm = 125;                         // Circuit Parameter in Ohms
s = 0.03;                         // Slip


// CALCULATIONS

Zf = ((%i*Xm/2)*((R2/(2*s))+(%i*X2/2)))/((%i*Xm/2)+(R2/(2*s))+(%i*X2/2));           // Impedance offered by the forward field in Ohms
Zb = ((%i*Xm/2)*((R2/(2*(2-s)))+(%i*X2/2)))/((%i*Xm/2)+(R2/(2*(2-s)))+(%i*X2/2));       // Impedance offered by the backward field in Ohms
Z = (R1+%i*X1)+Zf+Zb;                                   // Total Impedance in Ohms
I = V/Z;                                                // Total input current in Amphere
pf = cosd(atand(imag(I),real(I)));                      // Power Factor (lagging)
Vf = I*Zf;                                              // Forward Volatge at slip 0.03 in Volts
Vb = I*Zb;                                              // Backward Volatge at slip 0.03 in Volts
If = Vf/(0.5*R2/s);                                     // Forward Current in Amphere
Ib = Vb/(0.5*R2/(2-s));                                 // Forward Current in Amphere
Ws = 2*%pi*f;                                           // Synchronous Speed in radians per second
T =  ((0.5*If^2*R2)/(s*Ws))-((0.5*Ib^2*R2)/((2-s)*Ws));            // Starting torque
```

```
// DISPLAY RESULTS

disp("EXAMPLE : 5.30 : SOLUTION :-");
printf("\n (a) Input Current , I = %.2f < %.f A \n",abs(I),atand(imag(I),real(I)))
printf("\n (b) Power factor , pf = %.2f Lagging \n",pf)
printf("\n (c) Developed Torque , T = %.3f Nm \n",T)
```

Scilab code Exa 5.31 To find equivalent circuit parameters of the induction motor current and pf

```

// ELECTRICAL MACHINES
// R.K. Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.31

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

Wsc = 900;                          // Power at Blocked Rotor test in Watts
Vsc = 200;                          // Voltage at Blocked Rotor test in Volts
Isc = 5.0;                          // Current at Blocked
```

```
Rotor test in Amphere
Wo = 60;                          // Power at No-load test in Watts
Vo = 220;                         // Voltage at No-load test in Volts
Io = 1.5;                         // Current at No-load test in Amphere
m = 1;                            // Total Number of phase in Induction Motor
p = 4;                            // Total number of Poles of Induction Motor
V = 220;                          // Operating voltage of the Induction motor in Volts
f = 50;                           // Frequency in Hertz
s = 0.07;                         // Slip
R1 = 12;                          // Resistance of the main primary winding in Ohms


// CALCULATIONS

Zsc = Vsc/Isc;                         // Impedance in Blocked Rotor test in Ohms
Rsc = Wsc/(Isc^2);                     // Resistance in Blocked Rotor test in Ohms
Xsc = sqrt((Zsc^2)-(Rsc^2));           // Reactance in Blocked Rotor test in Ohms
Xl1 = Xsc/2;                           // Leakage reactance of stator and rotor to be equal in Ohms
Xl2 = Xsc/2;                           // Leakage reactance of stator and rotor to be equal in Ohms
R2 = Rsc-R1;                       // Equivalent resistance of rotor referred to stator in Ohms
Z0 = Vo/Io;                            // Impedance in Blocked Rotor test in Ohms
R0 = Wo/(Io^2);                        // Resistance in Blocked Rotor test in Ohms
X0 = sqrt((Z0^2)-(R0^2));              // Reactance in
```

```
Blocked Rotor test in Ohms
Wloss = Wo - ((Io^2)*(R1+R2));        // Loss in Watts
Xm_half = X0-Xl1-Xl2/2;
R2f = (R2/s)+((%i*Xl2)/2);                              // Forward resiatance in Ohms
Zf = ((%i*Xm_half)*R2f)/(%i*Xm_half+R2f);               // Total Forward impedance in Ohms
R2b = (R2/(2-s))+((%i*Xl2)/2);                          // Backward resiatance in Ohms
Zb = ((%i*Xm_half)*R2b)/(%i*Xm_half+R2b);               // Total Backward impedance in Ohms
Z = Zf+Zb+(R1+%i*Xl1);                                  // Total impedance in Ohms
I = V/Z;                                                // Motor Current in Amphere
pf = cosd(atand(imag(I),real(I)));                      // Power Factor (lagging)


// DISPLAY RESULTS

disp("EXAMPLE : 5.31 : SOLUTION :-");
printf("\n Circuit Parameters are \n\n (a) Leakage reactance of stator and rotor to be equal, Xl1 = Xl2 = %.2f Ohms \n",Xl1)
printf("\n (b) Equivalent resistance of rotor referred to stator, R2 = %.f Ohms \n",R2)
printf("\n (c) Total Forward impedance, Zf = %.1f < %.2f Ohms \n",abs(Zf),atand(imag(Zf),real(Zf)))
printf("\n (c) Total Backward impedance, Zb = %.2f < %.2f Ohms \n",abs(Zb),atand(imag(Zb),real(Zb)))
printf("\n (d) Total impedance, Z = %.2f < %.2f Ohms \n",abs(Z),atand(imag(Z),real(Z)))
printf("\n (e) Input Current, I = %.2f < %.2f A \n",
```

```
abs(I),atand(imag(I),real(I)))
printf("\n (f) Power factor , pf = %.2f Lagging \n",pf)
```

Scilab code Exa 5.32 To find To find equivalent circuit parameters of the induction motor current pf forwrad torque backward torque net torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 5 : INDUCTION MACHINES

// EXAMPLE : 5.32

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

Wsc = 600;                      // Power at Blocked Rotor test in Watts
Vsc = 125;                      // Voltage at Blocked Rotor test in Volts
Isc = 15.0;                     // Current at Blocked Rotor test in Amphere
Wo = 360;                       // Power at No-load test in Watts
Vo = 220;                       // Voltage at No-load test in Volts
Io = 6.5;                       // Current at No-load test in Amphere
```

```
m = 1;                               // Total Number of phase in Induction Motor
p = 4;                               // Total number of Poles of Induction Motor
V = 220;                             // Operating voltage of the Induction motor in Volts
f = 50;                              // Frequency in Hertz
s = 0.05;                            // Slip
R1 = 1.2;                            // Resistance of the main primary winding in Ohms


// CALCULATIONS

Zlr = Vsc/Isc;                            // Impedance in Blocked Rotor test in Ohms
Rlr = Wsc/(Isc^2);                        // Resistance in  Blocked Rotor test in Ohms
Xlr = sqrt((Zlr^2)-(Rlr^2));              // Reactance in Blocked Rotor test in Ohms
Xl1 = Xlr/2;                              // Leakage reactance of stator and rotor to be equal in Ohms
Xl2 = Xlr/2;                              // Leakage reactance of stator and rotor to be equal in Ohms
R2 = (Rlr-R1);                            // Equivalent resistance of rotor referred to stator in Ohms
R2_half = R2/2;                           // Equivalent resistance of rotor referred to stator in Ohms
Z0 = Vo/Io;                               // Impedance in Blocked Rotor test in Ohms
R0 = Wo/(Io^2);                           // Resistance in  Blocked Rotor test in Ohms
X0 = sqrt((Z0^2)-(R0^2));                 // Reactance in Blocked Rotor test in Ohms
Wloss = Wo - ((Io^2)*(R1+R2));            // Loss in Watts
Xm_half = X0-Xl1-Xl2/2;
R2f = (R2/(2*s))+((%i*Xl2)/2);
                                     // Forward resiatance in
```

```
Ohms
Zf = ((%i*Xm_half)*R2f)/(%i*Xm_half+R2f);            // Total Forward impedance in Ohms
R2b = (R2/(2*(2-s)))+((%i*Xl2)/2);                    // Backward resiatance in Ohms
Zb = ((%i*Xm_half)*R2b)/(%i*Xm_half+R2b);            // Total Backward impedance in Ohms
Z = Zf+Zb+(R1+%i*Xl1);                                // Total impedance in Ohms
I = V/Z;                                              // Motor Current in Amphere
pf = cosd(atand(imag(I),real(I)));                    // Power Factor (lagging)
Vf = I*Zf;                                            // Voltage across forward impedance in Volts
If = Vf/R2f;                                          // Forward current producing torque in Amphere
Tf = ((abs(If)^2)*R2)/(2*s);                          // Forward torque in synchronous Watts
Vb = I*Zb;                                            // Voltage across Backward impedance in Volts
Ib = Vb/R2b;                                          // Backward current producing torque in Amphere
Tb = ((abs(Ib)^2)*R2)/(2*(2-s));                      // Backward torque in synchronous Watts
T = Tf-Tb;                                            // Net torque in Synchronous Watts

```

```
// DISPLAY RESULTS

disp("EXAMPLE : 5.32 : SOLUTION :-");
printf("\n Circuit Parameters are \n\n (a) Leakage reactance of stator and rotor to be equal, Xl1 = Xl2 = %.2f Ohms \n",Xl1)
printf("\n (b) Equivalent resistance of rotor referred to stator, R2 = %.2f Ohms \n",R2)
printf("\n (c) Total Forward impedance, Zf = %.1f < %.2f Ohms \n",abs(Zf),atand(imag(Zf),real(Zf)))
printf("\n (c) Total Backward impedance, Zb = %.2f < %.2f Ohms \n",abs(Zb),atand(imag(Zb),real(Zb)))
printf("\n (d) Total impedance, Z = %.2f < %.2f Ohms \n",abs(Z),atand(imag(Z),real(Z)))
printf("\n (e) Input Current, I = %.2f < %.f A \n",abs(I),atand(imag(I),real(I)))
printf("\n (f) Power factor, pf = %.4f Lagging \n",pf)
printf("\n (g) Forward torque, Tf = %.2f Synchronous Watts \n",Tf)
printf("\n (h) Backward torque, Tb = %.2f Synchronous Watts \n",Tb)
printf("\n (i) Net torque, T = %.2f Synchronous Watts \n",T)
```

Chapter 6

Synchronous Machines

Scilab code Exa 6.1 To find number of poles in each cases

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.1

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

f = 50;                              // Generating
    Frequency in Hertz


// CALCULATIONS
```

```
// For Case(a)

Ns_a = 3000;                    // Synchronous speed in RPM
p_a = (120*f)/Ns_a;             // Number of poles

// For Case(b)

Ns_b = 1000;                    // Synchronous speed in RPM
p_b = (120*f)/Ns_b;             // Number of poles

// For Case(c)

Ns_c = 300;                     // Synchronous speed in RPM
p_c = (120*f)/Ns_c;             // Number of poles

// For Case(d)

Ns_d = 40;                      // Synchronous speed in RPM
p_d = (120*f)/Ns_d;             // Number of poles


// DISPLAY RESULTS

disp("EXAMPLE : 6.1 : SOLUTION :-") ;
printf("\n For Case(a) Ns = %.f , p = %.f \n ",Ns_a, p_a);
printf("\n For Case(b) Ns = %.f , p = %.f \n ",Ns_b, p_b);
printf("\n For Case(c) Ns = %.f , p = %.f \n ",Ns_c, p_c);
```

```
printf("\n For Case(d) Ns = %.f , p = %.f \n ",Ns_d,
    p_d);
```

Scilab code Exa 6.2 To find frequency

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.2

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

f = 60;                                 // Generating
    Frequency in Hertz
Ns = 1200;                              // Synchronous
    speed in RPM
Ns_r = 1000;                            // Alternator
    running speed in RPM


// CALCULATIONS

p = (120*f)/Ns;                         // Total number
    of poles
f_r = (p*Ns_r)/120;                     // Alternator
    running frequency in Hertz
```

```
// DISPLAY RESULTS

disp("EXAMPLE : 6.2 : SOLUTION :-") ;
printf("\n Alternator running frequency , f = %.f Hz
    \n ",f_r);
```

Scilab code Exa 6.3 To find Synchronous speed in each cases

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.3

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

p = 2;                                          // Total
     number of poles


// CALCULATIONS
// For Case(a)

f_a = 10;                                       //
    Frequency in Hertz
```

```
Ns_a = (120*f_a)/p;                              // Synchronous speed in RPM

// For Case(b)

f_b = 50;                                        // Frequency in Hertz
Ns_b = (120*f_b)/p;                              // Synchronous speed in RPM

// For Case(c)

f_c = 60;                                        // Frequency in Hertz
Ns_c = (120*f_c)/p;                              // Synchronous speed in RPM

// For Case(d)

f_d = 100;                                       // Frequency in Hertz
Ns_d = (120*f_d)/p;                              // Synchronous speed in RPM

// For Case(e)

f_e = 400;                                       // Frequency in Hertz
Ns_e = (120*f_e)/p;                              // Synchronous speed in RPM


// DISPLAY RESULTS

disp("EXAMPLE : 6.3 : SOLUTION :-");
printf("\n For Case (a) When f = %.f, Synchronous speed, Ns = %.f RPM \n",f_a,Ns_a)
printf("\n For Case (b) When f = %.f, Synchronous
```

```
speed, Ns = %.f RPM \n",f_b,Ns_b)
printf("\n For Case (c) When f = %.f, Synchronous speed, Ns = %.f RPM \n",f_c,Ns_c)
printf("\n For Case (d) When f = %.f, Synchronous speed, Ns = %.f RPM \n",f_d,Ns_d)
printf("\n For Case (e) When f = %.f, Synchronous speed, Ns = %.f RPM \n",f_e,Ns_e)
```

Scilab code Exa 6.4 To find leakage reactance and open circuit EMF in each cases

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.4

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

printf("\n EXAMPLE : 6.4 : \n\n            Given Data \n");
printf("\n Voc(v)   215        284        320        380        400        422        452        472        488        508        520        532        540        552        560 \n");
printf("\n If(A)     6.5        8          9          10         11         12         14         15         16
```

```
          17         18            19          20          22
          24 \n\n");
m = 3;                              // Total Number of phase
p = 6;                              // Total number of Poles
V = 400;                            // Operating voltage in Volts
I = 13.5;                           // Operating current in Amphere
N = 1000;                           // Speed in RPM
Ia_scc = 13.5;                      // SCC test Armature current in Amphere at If = 9.5 A
If_scc = 9.5;                       // SCC test field Rated current in Amphere
Ia_zpf = 13.5;                      // ZPF test Armature current in Amphere at If = 24 A
If_zpf = 24;                        // ZPF test field Rated current in Amphere


// CALCULATIONS
// Some of the data obtained from OCC and SCC test Graph or Pottier triangle in Figure6.15 & Page no:-386
Ra = 1.0;                                         // Armature resistance in Ohms

// For case (a)

BC = 120;                                         // Open circuit Voltage in Volts obtained from OCC and SCC test Graph or Pottier triangle Figure6.15 & Page no:-386
Xl = BC/(sqrt(3)*Ia_scc);                         // Per phase leakage reactance in Ohms


```

```
// For Case (b.1) 0.8 pf Lagging

pfa_b1 = acosd(0.8);                          // Power factor angle in degree
Er_b1 = (V/sqrt(3))+(Ia_scc*(cosd(pfa_b1)-%i*sind(pfa_b1))*(Ra+%i*Xl));       // Induced Voltage in Volts
R_b1 = 10; A_b1 = 9.5;                        //From OCC the field current required for Er_b1 (Should be in Line-line Voltage) Er_b1 = 379.12V will get R_b1 & A_b1 value Respectively from SCC (Figure6.15 & Page no:-386)
angle_b1 = 136.35;                            // Angle between R_b1 & A_b1 (figure 6.16(a) & Page no:-388) = 90'+9.48'+36.87' = 136.35'
F_b1 = sqrt((R_b1^2)+(A_b1^2)-(2*R_b1*A_b1*cosd(angle_b1)));                 // From phasor diagram in figure 6.16(a) & Page no:-388 the neccessary field excitation in Amphere
Eo_b1 = 525;                                  // Corresponding to field current F_b1 = 18.12 A the open circuit EMF from OCC is 525 V (Figure6.15 & Page no:-386)
r_b1 = 100*((Eo_b1-V)/V);                     // Percentage regulation


// For Case (b.2) 0.8 pf Leading

pfa_b2 = acosd(0.8);                          // Power factor angle in degree
Er_b2 = (V/sqrt(3))+(Ia_scc*(cosd(pfa_b2)+%i*sind(
```

```
   pfa_b2))*(Ra+%i*Xl));       // Induced Voltage in
   Volts
R_b2 = 8.3; A_b2 = 9.5;
   //From OCC the field current required for Er_b2 (
   Should be in Line−line Voltage) Er_b1 = 363.71 V
   will get R_b2 & A_b2 value Respectively from SCC
   (Figure6.15 & Page no:−386)
angle_b2 = 70.61;
   // Angle between R_b2 & A_b2 (figure 6.16(b) &
   Page no:−388) = 90'+17.48'−36.87' = 70.61'
F_b2 = sqrt((R_b2^2)+(A_b2^2)-(2*R_b2*A_b2*cosd(
   angle_b2)));                  // From phasor diagram
   in figure 6.16(b) & Page no:−388 the neccessary
   field excitation in Amphere
Eo_b2 = 338;
   // Corresponding to field current F_b2 = 10.36 A
   the open circuit EMF from OCC is 338 V (Figure6
   .15 & Page no:−386)
r_b2 = 100*((Eo_b2-V)/V);
   // Percentage regulation


// For Case (b.3) Unity pf Leading

pfa_b3 = acosd(1.0);
   // Power factor angle in degree
Er_b3 = (V/sqrt(3))+(Ia_scc*(cosd(pfa_b3)-%i*sind(
   pfa_b3))*(Ra+%i*Xl));       // Induced Voltage in
   Volts
R_b3 = 13; A_b3 = 9.5;
   //From OCC the field current required for Er_b3 (
   Should be in Line−line Voltage) Er_b1 = 440.30 V
```

```
    will get R_b3 & A_b3 value Respectively from SCC
    (Figure6.15 & Page no:-386)
angle_b3 = 105.81;
    // Angle between R_b3 & A_b3 (figure 6.16(c) &
    Page no:-388) = 90'+15.81' = 105.81'
F_b3 = sqrt((R_b3^2)+(A_b3^2)-(2*R_b3*A_b3*cosd(
    angle_b3)));                    // From phasor diagram
    in figure 6.16(c) & Page no:-388 the neccessary
    field excitation in Amphere
Eo_b3 = 520;
    // Corresponding to field current F_b2 = 18.10 A
    the open circuit EMF from OCC is 520 V (Figure6
    .15 & Page no:-386)
r_b3 = 100*((Eo_b3-V)/V);
    // Percentage regulation


// For Case (b.4) ZPF Lagging

pfa_b4 = acosd(0);
    // Power factor angle in degree
Er_b4 = (V/sqrt(3))+(Ia_scc*(cosd(pfa_b4)-%i*sind(
    pfa_b4))*(Ra+%i*Xl));        // Induced Voltage in
    Volts
R_b4 = 18; A_b4 = 9.5;
    //From OCC the field current required for Er_b4 (
    Should be in Line-line Voltage) Er_b4 = 521 V
    will get R_b4 & A_b4 value Respectively from SCC
    (Figure6.15 & Page no:-386)
angle_b4 = 177.57;
    // Angle between R_b4 & A_b4 = 90'-2.43'+90' =
    177.57'
```

```
F_b4 = sqrt((R_b4^2)+(A_b4^2)-(2*R_b4*A_b4*cosd(angle_b4)));                // The neccessary field excitation in Amphere
Eo_b4 = 570;                                                 // Corresponding to field current F_b4 = 27.50 A the open circuit EMF from OCC is 525 V (Figure6.15 & Page no:-386)
r_b4 = 100*((Eo_b4-V)/V);                                    // Percentage regulation


// For Case (b.4) ZPF Lagging

pfa_b5 = acosd(0);                                           // Power factor angle in degree
Er_b5 = (V/sqrt(3))+(Ia_scc*(cosd(pfa_b5)+%i*sind(pfa_b5))*(Ra+%i*Xl));      // Induced Voltage in Volts
R_b5 = 6.0; A_b5 = 9.50;                                     //From OCC the field current required for Er_b5 (Should be in Line-line Voltage) Er_b5 = 280.70 V will get R_b5 & A_b5 value Respectively from SCC (Figure6.15 & Page no:-386)
angle_b5 = 4.77;                                             // Angle between R_b5 & A_b5 = 90'-4.77'-90' = 4.77'
F_b5 = sqrt((R_b5^2)+(A_b5^2)-(2*R_b5*A_b5*cosd(angle_b5)));                // The neccessary field excitation in Amphere
Eo_b5 = 135;                                                 // Corresponding to field current F_b4 = 27.50 A the open circuit EMF from OCC is 135 V (Figure6
```

```
.15 & Page no:-386)
r_b5 = 100*((Eo_b5-V)/V);                     // Percentage regulation


// DISPLAY RESULTS

disp(" SOLUTION :-");
printf("\n (a)  Per phase leakage reactance , Xl = %.2f Ohms \n",Xl)
printf("\n For Case (b.1) 0.8 pf Lagging \n  Open circuit EMF, EMF = %.f V \n",Eo_b1)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",r_b1)
printf("\n For Case (b.2) 0.8 pf Leading \n  Open circuit EMF, EMF = %.f V \n",Eo_b2)
printf("\n Percenatge Regulation , R = %.1f Percenatge \n",r_b2)
printf("\n For Case (b.3) Unity pf Lagging \n  Open circuit EMF, EMF = %.f V \n",Eo_b3)
printf("\n Percenatge Regulation , R = %.f Percenatge \n",r_b3)
printf("\n For Case (b.4) ZPF Lagging \n  Open circuit EMF, EMF = %.f V\n",Eo_b4)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",r_b4)
printf("\n For Case (b.5) ZPF Leading \n  Open circuit EMF, EMF = %.f V \n",Eo_b5)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n\n",r_b5)
disp(" Calculated Answer in Tabular Column")
printf("\n   Power Factor                    0.8 Lag        0.8 Lead          1.0          ZPF Lag        ZPF Lead \n")
printf("\n Open circuit EMF (V)              %.f              %.f                %.f          %.f              %.f \n",Eo_b1,Eo_b2,Eo_b3,Eo_b4,Eo_b5)
```

```
printf("\n Percenatge Regulation          %.2f          %.1f          %.f.          %.2f          %.2f \n",r_b1,r_b2,r_b3,r_b4,r_b5)
```

Scilab code Exa 6.5a To find Synchronous reactance and open circuit EMF in each cases

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.5a

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

printf("\n EXAMPLE : 6.5a( Data is same as Example 6.4): \n\n          Given Data \n");
printf("\n Voc(v)   215     284     320     380     400     422     452     472     488     508     520     532     540     552     560 \n");
printf("\n If(A)     6.5     8     9     10     11     12     14     15     16     17     18     19     20     22     24 \n\n");
m = 3;                          // Total Number of phase in Induction Motor
```

```
p = 6;                                // Total number of Poles of Induction Motor
V = 400;                              // Operating voltage of the Induction motor in Volts
I = 13.5;                             // Operating current of the Induction motor in Amphere
N = 1000;                             // speed of the Induction motor in RPM
Ia_scc = 13.5;                        // SCC test Armature current in Amphere at If = 9.5 A
If_scc = 9.5;                         // SCC test field Rated current in Amphere
Ia_zpf = 13.5;                        // ZPF test Armature current in Amphere at If = 24 A
If_zpf = 24;                          // ZPF test field Rated current in Amphere


// CALCULATIONS
// Some of the data obtained from OCC and SCC test Graph or Pottier triangle in Figure6.15 & Page no:-386
Ra = 1.0;                                             // Armature resistance in Ohms
v = V/sqrt(3);                                        // Rated phase Voltage in Volts

// For case (a)

EMF_a1 = 345;                                         // From OCC and SCC test Graph or Pottier triangle in Figure6.15 & Page no:-386 open-circuit line-line voltage per phase is 345vVfor If = 9.50A in Volts
Zs_a1 = EMF_a1/(Ia_zpf*sqrt(3));                      // Unsaturated synchronous impedance at If=9.50A in Ohms
Xs_a1 = sqrt((Zs_a1^2)-(Ra^2));                       // Synchronous reactance at If =9.50A in Ohms
```

```
Ia_a2 = 15.75;                                   // Current from SCC in Figure6.15 & Page no:-386 is 15.75A for correspounding to the rated Voltage in Volts
Zs_a2 = V/(Ia_a2*sqrt(3));                       // Unsaturated synchronous impedance at If=9.50A in Ohms
Xs_a2 = sqrt((Zs_a2^2)-(Ra^2));                  // Synchronous reactance at If =9.50A in Ohms

// For Case (b.1) 0.8 pf Lagging

pfa_b1 = acosd(0.8);                             // Power factor angle in degree
real_b1 = (v+Ia_zpf*Ra*cosd(pfa_b1)+Ia_zpf*Xs_a1*sind(pfa_b1));
imag_b1 = (Ia_zpf*Xs_a1*cosd(pfa_b1)-Ia_zpf*Ra*sind(pfa_b1));
E_b1 = sqrt(real_b1^2+imag_b1^2);                // Induced Voltage pr phase in Volts from Figure6.19 (a) & Page no:-394 shows the phasor diagram for lagging pf
del_b1 = atand(imag_b1/real_b1);                 // Power angle in degree
r_b1 = 100*(E_b1-v)/v;                           // Percantage regulation


// For Case (b.2) 0.8 pf Leading

pfa_b2 = acosd(0.8);                             // Power factor angle in degree
real_b2 = (v+Ia_zpf*Ra*cosd(pfa_b2)-Ia_zpf*Xs_a1*
```

```
sind(pfa_b2));
imag_b2 = (Ia_zpf*Xs_a1*cosd(pfa_b2)+Ia_zpf*Ra*sind(pfa_b2));
E_b2 = sqrt(real_b2^2+imag_b2^2);                         // Induced Voltage pr phase in Volts from Figure6.19 (b) & Page no:-394 shows the phasor diagram for leading pf
del_b2 = atand(imag_b2/real_b2);                          // Power angle in degree
r_b2 = 100*(E_b2-v)/v;                                    // Percantage regulation


// For Case (b.3) Unity pf

pfa_b3 = acosd(1.0);                                      // Power factor angle in degree
real_b3 = (v+Ia_zpf*Ra);
imag_b3 = (Ia_zpf*Xs_a1);
E_b3 = sqrt(real_b3^2+imag_b3^2);                         // Induced Voltage pr phase in Volts from Figure6.19 (a) & Page no:-394 shows the phasor diagram for unity pf
del_b3 = atand(imag_b3/real_b3);                          // Power angle in degree
r_b3 = 100*(E_b3-v)/v;                                    // Percantage regulation

// For Case (b.4) ZPF pf Lagging

pfa_b4 = acosd(0);
```

```
    // Power factor angle in degree
real_b4 = (v+Ia_zpf*Xs_a1);
imag_b4 = (-Ia_zpf*Ra);
E_b4 = sqrt(real_b4^2+imag_b4^2);                    // Induced Voltage pr phase in Volts ZPF for lagging pf
del_b4 = atand(imag_b4/real_b4);                     // Power angle in degree
r_b4 = 100*(E_b4-v)/v;                               // Percantage regulation

// For Case (b.5) ZPF pf Leading

pfa_b5 = acosd(0);                                   // Power factor angle in degree
real_b5 = (v-Ia_zpf*Xs_a1);
imag_b5 = (Ia_zpf*Ra);
E_b5 = sqrt(real_b5^2+imag_b5^2);                    // Induced Voltage pr phase in Volts ZPF for lagging pf
del_b5 = atand(imag_b5/real_b5);                     // Power angle in degree
r_b5 = 100*(E_b5-v)/v;                               // Percantage regulation




// DISPLAY RESULTS

disp(" SOLUTION :-");
printf("\n (a.1)  Synchronous reactance for rated
```

```
current at If = %.2f, Xs = %.2f Ohms \n",If_scc,Xs_a1)
printf("\n (a.2)  Synchronous reactance for rated per phase Voltage  at v  = %.f, Xs = %.2f Ohms \n",v,Xs_a2)
printf("\n For Case (b.1) 0.8 pf Lagging \n  Induced EMF per phase , EMF = %.2f V \n",E_b1)
printf("\n Percenatge Regulation, R = %.2f Percenatge \n",r_b1)
printf("\n Power angle = %.2f degree \n",del_b1)
printf("\n For Case (b.2) 0.8 pf Leading \n  Induced EMF per phase, EMF = %.2f V \n",E_b2)
printf("\n Percenatge Regulation, R = %.2f Percenatge \n",r_b2)
printf("\n Power angle = %.2f degree \n",del_b2)
printf("\n For Case (b.3) Unity pf Lagging \n Induced EMF per phase, EMF = %.2f V \n",E_b3)
printf("\n Percenatge Regulation, R = %.2f Percenatge \n",r_b3)
printf("\n Power angle = %.2f degree \n",del_b3)
printf("\n For Case (b.4) ZPF Lagging \n  Induced EMF per phase, EMF = %.2f V\n",E_b4)
printf("\n Percenatge Regulation, R = %.2f Percenatge \n",r_b4)
printf("\n Power angle = %.1f degree \n",del_b4)
printf("\n For Case (b.5) ZPF Leading \n  Induced EMF per phase, EMF = %.2f V \n",E_b5)
printf("\n Percenatge Regulation, R = %.2f Percenatge \n",r_b5)
printf("\n Power angle = %.2f degree \n\n",del_b5)
disp(" Calculated Answer in Tabular Column")
printf("\n   Power Factor                    0.8 Lag         0.8 Lead          1.0          ZPF Lag        ZPF Lead \n")
printf("\n Open circuit EMF (V)             %.2f            %.2f             %.2f          %.2f              %.2f  \n",E_b1,E_b2,E_b3,E_b4,E_b5)
printf("\n Percenatge Regulation            %.2f
```

```
                %.2 f                %.2 f .          %.2 f              %
    .2 f  \n" ,r_b1 ,r_b2 ,r_b3 ,r_b4 ,r_b5)
printf("\n  Power  angle                          %.2 f
                %.2 f                %.2 f .          %.1 f
   %.2 f  \n" ,del_b1 ,del_b2 ,del_b3 ,del_b4 ,del_b5)
```

Scilab code Exa 6.6 To find open circuit EMF in each cases

```

// ELECTRICAL MACHINES
// R.K. Srivastava
//  First  Impression  2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.6

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

printf("\n EXAMPLE : 6.4 ( Data is same as Exaple
    6.4 ): \n\n              Given Data \n");
printf("\n Voc(v)  215        284        320        380
          400        422        452        472        488
          508        520        532        540        552
          560 \n");
printf("\n If(A)     6.5         8           9          10
           11         12          14          15          16
           17         18          19          20          22
           24 \n\n");
m = 3;                              // Total Number of
```

```
    phase in Induction Motor
p = 6;                              // Total number of Poles of Induction Motor
V = 400;                            // Operating voltage of the Induction motor in Volts
I = 13.5;                           // Operating current of the Induction motor in Amphere
N = 1000;                           // speed of the Induction motor in RPM
Ia_scc = 13.5;                      // SCC test Armature current in Amphere at If = 9.5 A
If_scc = 9.5;                       // SCC test field Rated current in Amphere
Ia_zpf = 13.5;                      // ZPF test Armature current in Amphere at If = 24 A
If_zpf = 24;                        // ZPF test field Rated current in Amphere


// CALCULATIONS
// Some of the data obtained from OCC and SCC test Graph or Pottier triangle in Figure6.15 & Page no :-386
Ra = 1.0;                                           // Armature resistance in Ohms
v = V/sqrt(3);                                      // Rated phase voltage in Volts


// For Case (a) 0.8 pf Lagging

pfa_a = acosd(0.8);
    // Power factor angle in degree
E_a = v+(Ia_scc*(cosd(pfa_a)-%i*sind(pfa_a))*Ra);
                                  // Induced Voltage in Volts
R1_a = 11.8; A_a = 9.50;
```

```
    //From OCC the field current required for E_a (
    Should be in Line-line Voltage) E_a = 419.05V
    will get R1_a & A_a value Respectively from SCC (
    Figure6.15 & Page no:-386)
angle_a = 124.95;

    // Angle between R1_a & A_a (Figure6.21a & Page
    no:-400) = 90'-1.92'+36.87' = 124.95'
F_a = sqrt((R1_a^2)+(A_a^2)-(2*R1_a*A_a*cosd(angle_a
    )));                    // From phasor diagram in
    figure 6.21(a) & Page no:-400 the neccessary
    field excitation in Amphere
Eo_a = 538;

    // Corresponding to field current F_a = 18.94 A
    the open circuit EMF from OCC is 538 V (Figure6
    .15 & Page no:-386)
r_a = 100*((Eo_a-V)/V);

    // Percentage regulation


// For Case (b) 0.8 pf Leading

pfa_b = acosd(0.8);

    // Power factor angle in degree
E_b = v+(Ia_scc*(cosd(pfa_b)+%i*sind(pfa_b))*Ra);
                                   // Induced Voltage in
    Volts
R1_b = 11.80; A_b = 9.50;

    //From OCC the field current required for E_b (
    Should be in Line-line Voltage) E_b = 419.10V
    will get R1_b & A_b value Respectively from SCC (
    Figure6.15 & Page no:-386)
angle_b = 55.07;
```

```
// Angle between R1_b & A_b (Figure6.21b & Page no:-400) = 90'-1.92'-36.87' = 55.07'
F_b = sqrt((R1_b^2)+(A_b^2)-(2*R1_b*A_b*cosd(angle_b)));            // From phasor diagram in figure 6.21(b) & Page no:-400 the neccessary field excitation in Amphere
Eo_b = 382;

// Corresponding to field current F_b = 10.10 A the open circuit EMF from OCC is 382 V (Figure6.15 & Page no:-386)
r_b = 100*((Eo_b-V)/V);

// Percentage regulation

// For Case (c) Unity pf

pfa_c = acosd(1);

// Power factor angle in degree
E_c = v+(Ia_scc*(cosd(pfa_c)+%i*sind(pfa_c))*Ra)                    // Induced Voltage in Volts
R1_c = 12.10; A_c = 9.50;
                                                                    // From OCC the field current required for E_c (Should be in Line-line Voltage) E_c = 423.50V will get R1_c & A_c value Respectively from SCC (Figure6.15 & Page no:-386)
angle_c = 90;

// Angle between R1_a & A_a (Figure6.21a & Page no:-400) = 90'
F_c = sqrt((R1_c^2)+(A_c^2)-(2*R1_c*A_c*cosd(angle_c)));            // From phasor diagram in figure 6.21(c) & Page no:-400 the neccessary field excitation in Amphere
```

```
Eo_c = 480;    // Corresponding to field current F_c = 15.38 A the open circuit EMF from OCC is 538 V (Figure6.15 & Page no:-386)
r_c = 100*((Eo_c-V)/V);    // Percentage regulation


// For Case (d) ZPF Lagging

pfa_d = acosd(0.0);    // Power factor angle in degree
E_d = v+(Ia_scc*(cosd(pfa_d)-%i*sind(pfa_d))*Ra)    // Induced Voltage in Volts
R1_d = 11.20; A_d = 9.50;    // From OCC the field current required for E_d (Should be in Line-line Voltage) E_d = 400.80V will get R1_d & A_d value Respectively from SCC (Figure6.15 & Page no:-386)
angle_d = 179.40;    // Angle between R1_d & A_d = 90'-0.6'+90' = 179.40'
F_d = sqrt((R1_d^2)+(A_d^2)-(2*R1_d*A_d*cosd(angle_d)));    // From phasor diagram the neccessary field excitation in Amphere
Eo_d = 545;    // Corresponding to field current F_d = 18.12 A the open circuit EMF from OCC is 545 V (Figure6.15 & Page no:-386)
r_d = 100*((Eo_d-V)/V);
```

```
// Percentage regulation

// For Case (d) ZPF Lagging

pfa_e = acosd(0.0);                              // Power factor angle in degree
E_e = v+(Ia_scc*(cosd(pfa_e)+%i*sind(pfa_e))*Ra)                           // Induced Voltage in Volts
R1_e = 11.20; A_e = 9.50;                                                  // From OCC the field current required for E_e (Should be in Line-line Voltage) E_d = 400.80V will get R1_e & A_e value Respectively from SCC (Figure6.15 & Page no:-386)
angle_e = 0.60;                                  // Angle between R1_e & A_e = 90'+0.6'-90' = 0.60'
F_e = sqrt((R1_e^2)+(A_e^2)-(2*R1_e*A_e*cosd(angle_e)));                   // From phasor diagram the neccessary field excitation in Amphere
Eo_e = 63;                                       // Corresponding to field current F_e = 1.70 A the open circuit EMF from OCC is 545 V (Figure6.15 & Page no:-386)
r_e = 100*((Eo_e-V)/V);                          // Percentage regulation



// DISPLAY RESULTS

disp(" SOLUTION :-");
printf("\n For Case (a) 0.8 pf Lagging \n  Open
```

```
circuit EMF, EMF = %.f V \n",Eo_a)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",r_a)
printf("\n For Case (b) 0.8 pf Leading \n  Open circuit EMF, EMF = %.f V \n",Eo_b)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",r_b)
printf("\n For Case (c) Unity pf Lagging \n  Open circuit EMF, EMF = %.f V \n",Eo_c)
printf("\n Percenatge Regulation , R = %.f Percenatge \n",r_c)
printf("\n For Case (d) ZPF Lagging \n  Open circuit EMF, EMF = %.f V\n",Eo_d)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",r_d)
printf("\n For Case (e) ZPF Leading \n  Open circuit EMF, EMF = %.f V \n",Eo_e)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n\n",r_e)
disp(" Calculated Answer in Tabular Column")
printf("\n   Power Factor                    0.8 Lag          0.8 Lead          1.0        ZPF Lag        ZPF Lead \n")
printf("\n Open circuit EMF (V)              %.f              %.f              %.f            %.f             %.f \n",Eo_a,Eo_b,Eo_c,Eo_d,Eo_e)
printf("\n Percenatge Regulation             %.2f             %.2f              %.f         %.2f            %.2f \n",r_a,r_b,r_c,r_d,r_e)
```

Scilab code Exa 6.7 To find open circuit EMF in each cases

```

// ELECTRICAL MACHINES
// R.K.Srivastava
```

```
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.7

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

printf("\n EXAMPLE : 6.7 ( Data as same as Example 6.4 ) : \n\n            Given Data \n");
printf("\n Voc(v)  215      284      320      380      400      422      452      472      488      508      520      532      540      552      560 \n");
printf("\n If(A)     6.5       8        9        10       11       12       14       15       16       17       18       19       20       22       24 \n\n");
m = 3;                          // Total Number of phase in Induction Motor
p = 6;                          // Total number of Poles of Induction Motor
V = 400;                        // Operating voltage of the Induction motor in Volts
I = 13.5;                       // Operating current of the Induction motor in Amphere
N = 1000;                       // speed of the Induction motor in RPM
Ia_scc = 13.5;                  // SCC test Armature current in Amphere at If = 9.5 A
If_scc = 9.5;                   // SCC test field Rated current in Amphere
Ia_zpf = 13.5;                  // ZPF test Armature
```

```
current in Amphere at If = 24 A
If_zpf = 24;                          // ZPF test field Rated current in Amphere


// CALCULATIONS
// Some of the data obtained from OCC and SCC test Graph or Pottier triangle in Figure6.15 & Page no:-386
Ra = 1.0;                                           // Armature resistance in Ohms
BC = 120;                                           // Open circuit Voltage in Volts obtained from OCC and SCC test Graph or Pottier triangle Figure6.15 & Page no:-386
Xl = BC/(sqrt(3)*Ia_scc);                           // Per phase leakage reactance in Ohms for this referring to example6.4 & page no:- 386

// For Case (a) 0.8 pf Lagging

pfa_a = acosd(0.8);                                 // Power factor angle in degree
Er_a = (V/sqrt(3))+(Ia_scc*(cosd(pfa_a)-%i*sind(pfa_a))*(Ra+%i*Xl));       // Induced Voltage in Volts
R_a = 9.8; A_a = 9.5;                               //From OCC the field current required for Er_a (Should be in Line-line Voltage) Er_a = 479.60V will get R_a & A_a value Respectively from SCC (Figure6.15 & Page no:-386)
angle_a = 126.87;                                   // Angle between R_a & A_a (Figure6.22(a) & Page no:-403) = 90'+36.87' = 126.87'
F_a = sqrt((R_a^2)+(A_a^2)-(2*R_a*A_a*cosd(angle_a))
```

```
);                    // From phasor diagram in
   figure 6.22(a) & Page no:-403 the neccessary
   field excitation in Amphere
Eo_a = 560;

   // Corresponding to field current ( OF'=OF+FF') ,
   F_a = 17.28 + 6.2 = 23.46 A the open circuit EMF
   from OCC is 560 V (Figure6.15 & Page no:-386)
r_a = 100*((Eo_a-V)/V);

   // Percentage regulation


// For Case (b) 0.8 pf Leading

pfa_b = acosd(0.8);

   // Power factor angle in degree
Er_b = (V/sqrt(3))+(Ia_scc*(cosd(pfa_b)+%i*sind(
   pfa_b))*(Ra+%i*Xl));      // Induced Voltage in
   Volts
R_b = 9.8; A_b = 9.5;

   //From OCC the field current required for Er_b (
   Should be in Line-line Voltage) Er_b = 363.90 V
   will get R_b & A_b value Respectively from SCC (
   Figure6.15 & Page no:-386)
angle_b = 53.13;

   // Angle between R_b2 & A_b2 (Figure6.22b & Page
   no:-403) = 90'-36.87' = 53.13'
F_b = sqrt((R_b^2)+(A_b^2)-(2*R_b*A_b*cosd(angle_b))
   );                    // From phasor diagram in
   figure 6.22(b) & Page no:-403 the neccessary
   field excitation in Amphere
Eo_b = 380;

   // Corresponding to field current ( OF'=OF+FF')
```

```
    F_b = 8.62+1.5=10.12 A the open circuit EMF from
    OCC is 380 V (Figure6.15 & Page no:-386)
r_b = 100*((Eo_b-V)/V);
    // Percentage regulation


// For Case (c) Unity pf Leading

pfa_c = acosd(1.0);
    // Power factor angle in degree
Er_c = (V/sqrt(3))+(Ia_scc*(cosd(pfa_c)-%i*sind(
    pfa_c))*(Ra+%i*Xl));        // Induced Voltage in
    Volts
R_c = 9.8; A_c = 9.5;
    //From OCC the field current required for Er_c (
    Should be in Line-line Voltage) Er_c = 440.11 V
    will get R_c & A_c value Respectively from SCC (
    Figure6.15 & Page no:-386)
angle_c = 90;
    // Angle between R_c & A_c (Figure6.22c & Page no
    :-403) = 90' = 90'
F_c = sqrt((R_c^2)+(A_c^2)-(2*R_c*A_c*cosd(angle_c))
    );                      // From phasor diagram in
    figure 6.22(c) & Page no:-403 the neccessary
    field excitation in Amphere
Eo_c = 510;
    // Corresponding to field current ( OF'=OF+FF')
    F_c = 13.65+3.0=16.65A the open circuit EMF from
    OCC is 510 V (Figure6.15 & Page no:-386)
r_c = 100*((Eo_c-V)/V);
    // Percentage regulation

```

```
// For Case (d) ZPF Lagging

pfa_d = acosd(0);                // Power factor angle in degree
Er_d = (V/sqrt(3))+(Ia_scc*(cosd(pfa_d)-%i*sind(pfa_d))*(Ra+%i*Xl));       // Induced Voltage in Volts
R_d = 9.8; A_d = 9.5;            //From OCC the field current required for Er_d (Should be in Line-line Voltage) Er_d = 570.20 V will get R_d & A_d value Respectively from SCC (Figure6.15 & Page no:-386)
angle_d = 180.0;                 // Angle between R_d & A_d = 90'+90' = 180'
F_d = sqrt((R_d^2)+(A_d^2)-(2*R_d*A_d*cosd(angle_d)));                 // The neccessary field excitation in Amphere
Eo_d = 600;                      // Corresponding to field current ( OF'=OF+FF') F_d = 19.3+16=35.30 A the open circuit EMF from OCC is 525 V (Figure6.15 & Page no:-386)
r_d = 100*((Eo_d-V)/V);          // Percentage regulation


// For Case (e) ZPF Lagging

pfa_e = acosd(0);                // Power factor angle in degree
Er_e = (V/sqrt(3))+(Ia_scc*(cosd(pfa_e)+%i*sind(pfa_e))*(Ra+%i*Xl));      // Induced Voltage in
```

```
    Volts
R_e = 9.8; A_e = 9.50;
    //From OCC the field current required for Er_e (
    Should be in Line-line Voltage) Er_e = 281.10 V
    will get R_e & A_e value Respectively from SCC (
    Figure6.15 & Page no:-386)
angle_e = 0.0;
    // Angle between R_e & A_e = 90'-90' = 0.0'
F_e = sqrt((R_e^2)+(A_e^2)-(2*R_e*A_e*cosd(angle_e))
    );                          // The neccessary field
    excitation in Amphere
Eo_e = 5;
    // Corresponding to field current ( OF'=OF+FF')
    F_e = 0.0+0.30=0.30 A the open circuit EMF from
    OCC is 5 V (Figure6.15 & Page no:-386)
r_e = 100*((Eo_e-V)/V);
    // Percentage regulation


// DISPLAY RESULTS

disp(" SOLUTION :-");
printf("\n  Per phase leakage reactance , Xl = %.2f
    Ohms \n",Xl)
printf("\n For Case (a) 0.8 pf Lagging \n  Open
    circuit EMF, EMF = %.f V \n",Eo_a)
printf("\n Percenatge Regulation , R = %.f Percenatge
     \n",r_a)
printf("\n For Case (b) 0.8 pf Leading \n  Open
    circuit EMF, EMF = %.f V \n",Eo_b)
printf("\n Percenatge Regulation , R = %.f Percenatge
     \n",r_b)
printf("\n For Case (c) Unity pf Lagging \n  Open
    circuit EMF, EMF = %.f V \n",Eo_c)
```

```
printf("\n Percenatge Regulation , R = %.f Percenatge \n",r_c)
printf("\n For Case (d) ZPF Lagging \n  Open circuit EMF, EMF = %.f V\n",Eo_d)
printf("\n Percenatge Regulation , R = %.f Percenatge \n",r_d)
printf("\n For Case (e) ZPF Leading \n  Open circuit EMF, EMF = %.f V \n",Eo_e)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n\n",r_e)
disp(" Calculated Answer in Tabular Column")
printf("\n   Power Factor                  0.8 Lag          0.8 Lead         1.0        ZPF Lag        ZPF Lead \n")
printf("\n Open circuit EMF (V)            %.f             %.f              %.f         %.f            %.f \n",Eo_a,Eo_b,Eo_c,Eo_d,Eo_e)
printf("\n Percenatge Regulation           %.f               %.f             %.1f         %.f           %.2f \n",r_a,r_b,r_c,r_d,r_e)
```

Scilab code Exa 6.9 To find induced EMF power angle and percent regulation

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.9

clear ; clc ; close ; // Clear the work space and
```

```
console


// GIVEN DATA

printf("\n EXAMPLE : 6.9 : \n\n              Given Data \n");
printf("\n Voc(kV)    10       10.80       11.50          12.10          12.60          13          14          14.50          14.80 \n");
printf("\n If(A)     175        200          225          250           275          300          400          450          500 \n\n");
p = 6;                              // Total number of Poles of Alternator
V = 11*10^3;                        // Operating voltage of the Alternator in Volts
N = 1500;                           // speed of the Alternator in RPM
Ia_scc = 2099;                      // SCC test Armature current in Amphere at If = 200 A
If_scc = 200;                       // SCC test field Rated current in Amphere
Ia_pt = 2099;                       // Pottier test Armature current in Amphere at If = 450 A
If_pt = 450;                        // Pottier test field  Rated current in Amphere
VA = 40*10^6;                       // VA rating of the Alternator in Volts-Amphere
f = 50;                             // Operating Frequency of the Alternator in Hertz
pf = 0.8;                           // Power factor (lagging)

// CALCULATIONS
// Some of the data obtained from OCC and SCC test Graph or Pottier triangle in Figure6.24 & Page no :-407
```

```
v = V/sqrt(3);                                      // Rated phase Voltage in Volts
I = VA/(sqrt(3)*V);                                 // Full-load phase current in Amphere
Xl = 0.4481;                                        // Leakage reactance in Ohms From OCC and SCC test Graph or Pottier triangle in Figure6.24 & Page no:-407


// For Case(a) General Method

pfa_a = acosd(pf);                                  // Power factor angle in degree
Er_a = (V/sqrt(3))+(Ia_scc*(cosd(pfa_a)-%i*sind(pfa_a))*Xl);   // Induced Voltage in Volts
R_a = 208.4; A_a = 200;                             //From OCC the field current required for Er_a (Should be in Line-line Voltage) Er_a = 11043.66 V will get R_a & A_a value Respectively from SCC (Figure6.24 & Page no:-407)
angle_a = 131.93;                                   // Angle between R_a & A_a (Figure6.25(a) & Page no:-408) = 90'+5.06'+36.87' = 131.93'
F_a = sqrt((R_a^2)+(A_a^2)-(2*R_a*A_a*cosd(angle_a)));   // From phasor diagram in figure 6.25(a) & Page no:-408 the neccessary field excitation in Amphere
Eo_a = 13720;
```

```
    // Corresponding to field current, F_a = 373 A
    the open circuit EMF from OCC is 560 V (Figure6
    .15 & Page no:-386)
r_a = 100*((Eo_a-V)/V);
    // Percentage regulation


// For Case (b) ASA Method

pfa_b = acosd(pf);
    // Power factor angle in degree
Er_b = (V/sqrt(3))+Ia_scc*(cosd(pfa_b)-%i*sind(pfa_b
    ))*Xl;                  // Induced Voltage in Volts
R_b = 160; A_b = 200;
    //From OCC the field current required for Er_b (
    Should be in Line-line Voltage) Er_b = 11043.66 V
     will get R_b & A_b value Respectively from SCC (
    Figure6.24 & Page no:-407)
angle_b = 126.87;
    // Angle between R_b2 & A_b2 (Figure6.22b & Page
    no:-403) = 90'+36.87' = 126.87'
F_b = sqrt((R_b^2)+(A_b^2)-(2*R_b*A_b*cosd(angle_b))
    );                      // From phasor diagram in
    figure 6.25(b) & Page no:-408 the neccessary
    field excitation in Amphere
Eo_b = 13660;
    // Corresponding to field current ( OF'=OF+FF')
    F_b = 337.88+15.38=337.88  A the open circuit EMF
     from OCC is 13660 V (Figure6.15 & Page no:-386)
r_b = 100*((Eo_b-V)/V);
    // Percentage regulation

```

```

// DISPLAY RESULTS

disp(" SOLUTION :-");
printf("\n For Case (a) General(ZPF) Method \n Induced EMF, EMF = %.f < %.2f V \n",abs(Er_a),atand(imag(Er_a),real(Er_a)))
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",r_a)
printf("\n For Case (b) ASA Method \n  Induced EMF, EMF = %.f < %.2f V \n",abs(Er_b),atand(imag(Er_b),real(Er_b)))
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",r_b)
printf("\n\n     [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n        WRONGLY PRINTED ANSWERS ARE :- For Case (a) General(ZPF) Method (a) Induced EMF = 6376<-5.07 degree instead of %.f < %.2f \n ",abs(Er_a),atand(imag(Er_a),real(Er_a)))
printf("\n                                           For Case (b) ASA Method            (a) Induced EMF = 6376<-5.07 degree instead of %.f < %.2f \n\n ",abs(Er_b),atand(imag(Er_b),real(Er_b)))
printf(" CALCULATION OF THE POWER ANGLE IS NOT CALCULATED IN THE TEXT BOOK FOR THIS PROBLEM\n ")
printf("\n INDUCED EMF AND PERCENTAGE REGULATION IS APPROXIMATED VALUE BECACUSE IN THE TEXT BOOK, CALCULATED INDUCED EMF IS WRONGLY PRINTED")
```

Scilab code Exa 6.10 To find field current

```

// ELECTRICAL MACHINES
```

```
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.10

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

m = 3;                              // Total Number of
    Phase in Alternator
p = 2;                              // Total number of
    Poles of Alternator
V = 11*10^3;                        // Operating voltage
    of the Alternator in Volts
VA = 10*10^6;                       // VA rating of the
    Alternator in Volts-Amphere
f = 50;                             // Operating
    Frequency of the alternator in Hertz
pf = 0.8;                           // Power factor (
    lagging)
Vf = 12*10^3;                       // Operating field
    voltage of the Alternator in Volts
If = 160;                           // Field Current in
    Amphere
Ra = 0.05;                          // Armature
    Resistance per phase in Ohms
Xs = 1.5;                           // Winding leakage
    reactance per phase in Ohms
A = 150;                          // The armature MMF at
    rated current is equivalent to Field Current in
    Amphere

```

```

// CALCULATIONS

Vt = V/sqrt(3);
    // Rated per phase Voltage in Volts
Ia = VA/(sqrt(3)*V);
    // Rated Armature Current in Amphere
pfa = acosd(pf);
    // Power factor angle in degree
Er = Vt+Ia*(cosd(pfa)-%i*sind(pfa))*(Ra+%i*Xs);
    // Induced EMF in Volts
R_a = 90 + atand(imag(Er),real(Er));
    // Angle of R in Degree
R = 160 * exp( %i * (R_a) * %pi/180);
    // (Line-line Voltage) Er = 11902.40V will get R
    from Air gap Characteristics
A_n = A * exp( %i * (-pfa) * %pi/180);
F = R - A_n;
    // Field Current required to produce the
    excitation EMF in Amphere


// DISPLAY RESULTS

disp("EXAMPLE : 6.10: SOLUTION :-");
printf("\n (a) Field Current required to produce the
     excitation EMF, F = %.2f A \n",abs(F))
```

Scilab code Exa 6.11 To find leakage and synchronous reactance and field current

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
```

```
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.11

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

printf("\n EXAMPLE : 6.11 : \n\n           Given Data
     \n");
printf("\n Voc( V)    12      13        13.8       14.5
          15.1   \n");
printf("\n If(A)     175       200        225        250
           275    \n\n");
V = 11*10^3;                   // Operating voltage
    of the Synchronous generator in Volts
VA = 50*10^6;                  // VA rating of the
    Synchronous generator in Volts-Amphere
f = 50;                        // Operating
    Frequency of the Synchronous generator in Hertz
N = 1500;                      // Speed of the
    Synchronous generator in RPM
If_scc = 200;                  // SCC test field
    Rated current in Amphere at rated Short circuit
    current
If_zpf = 400;                  // ZPF test field
    Rated current in Amphere at rated voltage and
    rated current
pf = 0.8;                      // Power factor (
    lagging)


// CALCULATIONS
// Some of the data obtained from OCC and SCC test
```

```
Graph or Pottier triangle in Figure6.30 & Page no:-413

Vt = V/sqrt(3);                                    // Rated per phase Voltage in Volts
Ia = VA/(sqrt(3)*V);                               // Rated Armature Current in Amphere
pfa = acosd(pf);                                   // Power factor angle in degree
O = 13000;                                         // Open circuit Voltage in Volts obtained from OCC and SCC test Graph or Pottier triangle Figure6.30 & Page no:-413
Xs = O/(sqrt(3)*Ia);                               // Synchronous reactance per phase in Ohms
BC = 4000;                                         // Open circuit Voltage in Volts obtained from OCC and SCC test Graph or Pottier triangle Figure6.30 & Page no:-413
Xl = BC/(sqrt(3)*Ia );                             // Per phase leakage reactance in Ohms

// For Case (a) General (ZPF) Method

Er_a = Vt+Ia*(cosd(pfa)-%i*sind(pfa))*(%i*Xl);                            // Induced EMF in Volts
R_a = 220; A_a = 200;                              //From OCC the field current required for Er_a (Should be in Line-line Voltage) Er_a = 13776V will get R_a & A_a value Respectively from SCC (Figure6.30 & Page no:-403)
angle_a = 140.3;                                   // Angle between R_a & A_a = 90'+13.43'+36.87' = 140.3'
F_a = sqrt((R_a^2)+(A_a^2)-(2*R_a*A_a*cosd(angle_a))
```

```
);                    // From phasor diagram in figure 6.16(a) & Page no:-388 the neccessary field excitation in Amphere
Eo_a = 20000;                    // Corresponding to field current F_a = 470.90 A the open circuit EMF from OCC is 20000 V (Figure6.30 & Page no:-413)
r_a = 100*((Eo_a-V)/V);                    // Percentage regulation


// For Case(b) EMF Method

Er_b = Vt+Ia*(cosd(pfa)-%i*sind(pfa))*(%i*Xs);                    // Induced Voltage in Volts
F_b = 500;                    //From OCC the field current required for Er_b (Should be in Line-line Voltage) Er_b = 21404 V will get 500A from SCC (Figure6.15 & Page no:-386)

// For Case (c) MMF Method

Er_c = Vt+Ia*(cosd(pfa)-%i*sind(pfa))*0;                    // Induced Voltage in Volts ( Zero is multipied because Armature reistance is zero (not mentioned))
R_c = 160; A_c = 200;                    //From OCC the field current required for Er_c (Should be in Line-line Voltage) Er_c = 11000 V will get R_c & A_c value Respectively from SCC (Figure6.30 & Page no:-413)
angle_c = 126.27;
```

```
// Angle between R_c & A_c = 90'-0'+36.87' = 126.27' {can refer figure 6.21a at page no:-400}
F_c = sqrt((R_c^2)+(A_c^2)-(2*R_c*A_c*cosd(angle_c)));                    // From phasor diagram {can refer figure 6.21a at page no:-400} the neccessary field excitation in Amphere


// For Case (d) ASA Method

Er_d = Vt+Ia*(cosd(pfa)-%i*sind(pfa))*(%i*Xl);                    // Induced Voltage in Volts
R_d = 220; A_d = 200;
    //From OCC the field current required for Er_d (Should be in Line-line Voltage) Er_d = 13800 V will get R_d & A_d value Respectively from SCC (Figure6.30 & Page no:-413)
angle_d = 126.87;
    // Angle between R_d & A_d = 90'+36.87' = 126.87'{can refer figure 6.22a at page no:-40}
F_d1 = sqrt((R_d^2)+(A_d^2)-(2*R_d*A_d*cosd(angle_d)));                    // from Phasor diagram {can refer figure 6.2a at page no:-400 The neccessary field excitation in Amphere
F_d = F_d1 + 30;
    // from Phasor diagram {can refer figure 6.2a at page no:-400 The Total neccessary field excitation in Amphere


// DISPLAY RESULTS

disp(" SOLUTION :-");
```

```
printf("\n (a) Leakage Reactance, Xl = %.2f Ohms \n"
    ,Xl)
printf("\n (b) Synchronous Reactance, Xs = %.2f Ohms
     \n",Xs)
printf("\n For Case (a) General (ZPF) Method \n
    Field Current required for maintaing the rated
    terminal voltage for rated kVA rating at %.2f
    Lagging Power factor , F = %.2f A \n",pf,F_a)
printf("\n For Case (a) EMF Method \n Field Current
    required for maintaing the rated terminal voltage
     for rated kVA rating at %.2f Lagging Power
    factor , F = %.f A \n",pf,F_b)
printf("\n For Case (a) MMF Method \n Field Current
    required for maintaing the rated terminal voltage
     for rated kVA rating at %.2f Lagging Power
    factor , F = %.2f A \n",pf,F_c)
printf("\n For Case (a) ASA Method \n Field Current
    required for maintaing the rated terminal voltage
     for rated kVA rating at %.2f Lagging Power
    factor , F = %.f A \n",pf,F_d)
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED
   WRONGLY ( I verified by manual calculation )]\n"
    );
printf("\n      WRONGLY PRINTED ANSWERS ARE :- For
    Case (a) General (ZPF) Method \n (a) Field
    Current required for maintaining the rated
    terminal voltage for rated kVA rating at %.2f
    Lagging Power factor , F = 470.90 A instead of %
    .2f A \n",pf,F_a);
```

Scilab code Exa 6.12 To find leakage and synchronous reactance and field current

```

// ELECTRICAL MACHINES
```

```
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.12

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA


printf("\n EXAMPLE : 6.12 : \n\n            Given Data \n");
printf("\n Voc(V)    175      250       280       300       330       350       370        380 \n");
printf("\n If(A)      10        17        20        23        30        38        50         60 \n");
printf("\n Vzpf(V)       -          -         -         0       130       210       265       280 \n\n");
V = 433;                              // Operating voltage of the Alternator in Volts
N = 3000;                             // speed of the Alternator in RPM
VA = 20*10^3;                         // VA rating of the Alternator in Volts-Amphere
f = 50;                               // Operating Frequency of the Alternator in Hertz
pf = 0.8;                             // Power factor (lagging)


// CALCULATIONS
// Some of the data obtained from OCC and SCC test Graph or Pottier triangle in Figure6.35 & Page no
```

```
:-420

Vt = V/sqrt(3); // Rated per phase Voltage in Volts
Ia = VA/(sqrt(3)*V); // Rated Armature Current in Amphere
pfa = acosd(pf); // Power factor angle in degree
O = 298; // Open circuit Voltage in Volts obtained from OCC and SCC test Graph or Pottier triangle Figure6.30 & Page no:-413
Xs = O/(sqrt(3)*Ia); // Synchronous reactance per phase in Ohms
BC = 70; // Open circuit Voltage in Volts obtained from OCC and SCC test Graph or Pottier triangle Figure6.30 & Page no:-413
Xl = BC/(sqrt(3)*Ia ); // Per phase leakage reactance in Ohms
E = Vt+Ia*(cosd(pfa)-%i*sind(pfa))*(%i*Xs); // Induced EMF in Volts using EMF Method
c = 380-60; // The open Voltage voltage is 694.50V (line-line) its Obatained by extrapolation
y = 694.50; // The open Voltage voltage is 694.50V (line-line) its Obatained by extrapolation
// Extrapolation Equation is y = (x*(380-370)/(60-50))*c
x = y - c; // The required field current in Amphere


// DISPLAY RESULTS

disp(" SOLUTION :-");
printf("\n (a) Leakage Reactance , Xl = %.2f Ohms \n"
```

```
,Xl)
printf("\n (b) Synchronous Reactance, Xs = %.2f Ohms \n",Xs)
printf("\n (c) Field Current required for maintaing the rated terminal voltage for rated kVA rating at %.2f Lagging Power factor , F = %.2f A \n",pf,x)
```

Scilab code Exa 6.13 To find induced EMF power angle and percent regulation

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.13

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA


V = 400;                              // Operating voltage of the Synchronous generator in Volts
VA = 60*10^3;                         // VA rating of the Synchronous generator in Volts-Amphere
f = 50;                               // Operating Frequency of the Synchronous generator in Hertz
xd = 1.5;                             // Direct axis
```

```
    reactances in Ohms
xq = 0.6;                          // Quadrature axis
    reactances in Ohms


// CALCULATIONS

I = VA/(sqrt(3)*V);                              //
    Rated current in Amphere
v = V/sqrt(3);                                   //
    Rated Phase Votage in Volts

// For Case (a) 0.80 lagging Power factor (Refer
    figure 6.36 page no. 421)

pf_a = 0.8;

    // Power factor
pfa_a = acosd(pf_a);
                                               //
    Power factor angle in deg
pa_a = atand((I*xq*cosd(pfa_a))/(v+I*xq*sind(pfa_a))
    );           // Power angle in deg
Iq_a = I*cosd(pfa_a+pa_a);
                                         // Current
    in Amphere
Id_a = I*sind(pfa_a+pa_a);
                                         // Current
    in Amphere
Eo_a = sqrt((v+Id_a*xd*cosd(pa_a)-Iq_a*xq*sind(pa_a)
    )^2 + (Id_a*xd*sind(pa_a)+Iq_a*xq*cosd(pa_a))^2);
                                                  //
    Induced EMF in Volts
pr_a = ((Eo_a-v)/v)*100;
                                            //
    Percentage regulation

// For Case (b) Unity Power factor (Refer figure
```

```
6.37 page no. 422)

pf_b = 1.0;                                     // Power factor
pfa_b= acosd(pf_b);                                              // Power factor angle in deg
pa_b = atand((I*xq*cosd(pfa_b))/(v+I*xq*sind(pfa_b)));           // Power angle in deg
Iq_b = I*cosd(pfa_b+pa_b);
Id_b = I*sind(pfa_b+pa_b);
Eo_b = sqrt((v+Id_b*xd*cosd(pa_b)-Iq_b*xq*sind(pa_b))^2 + (Id_b*xd*sind(pa_b)+Iq_b*xq*cosd(pa_b))^2);    // Induced EMF in Volts
pr_b = ((Eo_b-v)/v)*100;                                          // Percentage regulation

// For Case (c) 0.80 lagging Power factor (Refer figure 6.36 page no. 421)

pf_c = 0.8;                                     // Power factor
pfa_c = acosd(pf_c);                                             // Power factor angle in deg
pa_c = atand((I*xq*cosd(pfa_c))/(v-I*xq*sind(pfa_c)));           // Power angle in deg
Iq_c = I*cosd(pfa_c-pa_c);
Id_c = I*sind(pfa_c-pa_c);
Eo_c = sqrt((v-Id_c*xd*cosd(pa_c)-Iq_c*xq*sind(pa_c))^2 + (-Id_c*xd*sind(pa_c)+Iq_c*xq*cosd(pa_c))^2);    // Induced EMF in Volts
pr_c = ((Eo_c-v)/v)*100;
```

```
                                                    //
   Percentage regulation


// DISPLAY RESULTS

disp("EXAMPLE : 6.13: SOLUTION :-");
printf("\n For Case (a) 0.80 lagging Power factor \n
     Induced EMF, EMF = %.2f V \n",Eo_a)
printf("\n Power angle = %.3f degree \n",pa_a)
printf("\n Percenatge Regulation , R = %.1f
   Percenatge \n",pr_a)
printf("\n For Case (b) Unity Power factor \n
   Induced EMF, EMF = %.2f V \n",Eo_b)
printf("\n Power angle = %.2f degree \n",pa_b)
printf("\n Percenatge Regulation , R = %.2f
   Percenatge \n",pr_b)
printf("\n For Case (c) 0.80 leading Power factor \n
     Induced EMF, EMF = %.2f V \n",Eo_c)
printf("\n Power angle = %.2f degree \n",pa_c)
printf("\n Percenatge Regulation , R = %.2f
   Percenatge \n",pr_c)
```

Scilab code Exa 6.14 To find induced EMF power angle and percent regulation

```

// ELECTRICAL MACHINES
// R.K. Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.14
```

```
clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA


v = 1.0;                          // Operating voltage
    of the Synchronous generator in pu
xd = 1.0;                         // Direct axis
    reactances in pu
xq = 0.5;                         // Quadrature axis
    reactances in pu
I = 1.0;                          // Rated current in
    pu


// CALCULATIONS

// For Case (a) 0.80 lagging Power factor (Refer
    figure 6.36 page no. 421)

pf_a = 0.8;

    // Power factor
pfa_a = acosd(pf_a);
                                                      //
    Power factor angle in deg
pa_a = atand((I*xq*cosd(pfa_a))/(v+I*xq*sind(pfa_a))
    );          // Power angle in deg
Iq_a = I*cosd(pfa_a+pa_a);
Id_a = I*sind(pfa_a+pa_a);
Eo_a = sqrt((v+Id_a*xd*cosd(pa_a)-Iq_a*xq*sind(pa_a)
    )^2 + (Id_a*xd*sind(pa_a)+Iq_a*xq*cosd(pa_a))^2);
                                                         //
    Induced EMF in Volts
pr_a = ((Eo_a-v)/v)*100;
```

```
                                        // Percentage regulation

// For Case (b) Unity Power factor (Refer figure 6.37 page no. 422)

pf_b = 1.0;
    // Power factor
pfa_b= acosd(pf_b);
                                        // Power factor angle in deg
pa_b = atand((I*xq*cosd(pfa_b))/(v+I*xq*sind(pfa_b))
    );          // Power angle in deg
Iq_b = I*cosd(pfa_b+pa_b);
Id_b = I*sind(pfa_b+pa_b);
Eo_b = sqrt((v+Id_b*xd*cosd(pa_b)-Iq_b*xq*sind(pa_b)
    )^2 + (Id_b*xd*sind(pa_b)+Iq_b*xq*cosd(pa_b))^2);
                                        // Induced EMF in Volts
pr_b = ((Eo_b-v)/v)*100;
                                        // Percentage regulation

// For Case (c) 0.80 lagging Power factor (Refer figure 6.36 page no. 421)

pf_c = 0.8;
    // Power factor
pfa_c = acosd(pf_c);
                                        // Power factor angle in deg
pa_c = atand((I*xq*cosd(pfa_c))/(v-I*xq*sind(pfa_c))
    );           // Power angle in deg
Iq_c = I*cosd(pfa_c-pa_c);
Id_c = I*sind(pfa_c-pa_c);
Eo_c = sqrt((v-Id_c*xd*cosd(pa_c)-Iq_c*xq*sind(pa_c)
```

```
)^2 + (-Id_c*xd*sind(pa_c)+Iq_c*xq*cosd(pa_c))^2);    // Induced EMF in Volts
pr_c = ((Eo_c-v)/v)*100;    // Percentage regulation


// DISPLAY RESULTS

disp("EXAMPLE : 6.14: SOLUTION :-");
printf("\n For Case (a) 0.80 lagging Power factor \n Induced EMF, EMF = %.4f V \n",Eo_a)
printf("\n Power angle = %.1f degree \n",pa_a)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",pr_a)
printf("\n For Case (b) Unity Power factor \n Induced EMF, EMF = %.2f V \n",Eo_b)
printf("\n Power angle = %.2f degree \n",pa_b)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",pr_b)
printf("\n For Case (c) 0.80 leading Power factor \n Induced EMF, EMF = %.4f V \n",Eo_c)
printf("\n Power angle = %.2f degree \n",pa_c)
printf("\n Percenatge Regulation , R = %.2f Percenatge \n",pr_c)
```

Scilab code Exa 6.15 To find field current

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

```

```
// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.15

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA


If = 1.25;                    // Given that rated voltage at air gap line for this field current in pu
IF = 0.75;                    // Rated current in SC test for this field current in pu
Ia = 1.0;                     // Rated current in Per unit
pf = 0.8;                     // Power factor
V = 1.0;                      // Rated Volatge in pu


// CALCULATIONS

pfa = acosd(pf);                                        // Power factor angle in deg
Voc = (V*IF)/If;                                        // Open circuit volatge in pu
xs = Voc/Ia;                                            // Syncronous reactance in pu
E = V + Ia*(cosd(pfa)-(%i)*sind(pfa))*(%i*xs);          // Induced EMF in pu
a = abs(E)*If;

```

```
// DISPLAY RESULTS

disp("EXAMPLE : 6.15: SOLUTION :-");
printf("\n Induced EMF, E = %.2f < %.2f pu \n ",abs(E),atand(imag(E),real(E)))
printf("\n The field current required for %.2f pu voltage on air gap line %.1f pu \n",abs(E),a)
```

Scilab code Exa 6.16 To find induced EMF power angle power and counter torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.16

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA


V = 440;                              // Operating voltage of the alternator in Volts
VA = 20*10^3;                         // VA rating of the alternator in Volts-Amphere
f = 50;                               // Operating Frequency of the alternator in Hertz
N = 3000;                             // Rotation of the
```

```
alternator in RPM
Ra = 0.0;                        // Armature reistance in Ohms
xl = 0.6;                        // Armature reactances in Ohms
pf = 0.8;                        // Power factor lagging
pfa = acosd(pf);                  // ower factor angle in deg
p = (120*f)/N;                   // Number of poles
w = (2*%pi*f);                   //  Rotation speed in Radians per second
v = V/sqrt(3);                   // Rated phase voltage in Volts
I = VA/(sqrt(3)*V);              // Rated curent in Amphere
If = I;                          // Given field current = armature current from SCC test in Amphere
E = 16*If;                       // Open-circuit EMF at field current in Volts given from Equation E = 16If refer page no. 431
xs = E/(If*sqrt(3));             // Synchronous reactance in Ohms
Eo = sqrt((v+I*xs*sind(pfa))^2 + (I*xs*cosd(pfa))^2);             // Induced EMF in Volts
pa = atand(193.98/399.49);
                                                  // From above equation Eo
pr = ((Eo-v)/v)*100;
                                                        // Percent regulation
P = (3*v*Eo*sind(pa))/(xs*1000);
                                             // Power inKilo-Watts
T = (P*1000)/w;

// Torque devolped in Newton-meter
```

```
// DISPLAY RESULTS

disp("EXAMPLE : 6.16: SOLUTION :-");
printf("\n Induced EMF, EMF = %.f V \n",Eo)
printf("\n Power angle = %.2f degree \n",pa)
printf("\n Power, P = %.3f kW \n",P)
printf("\n Counter Torque, T = %.2f N-m \n",T)
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED
   WRONGLY ( I verified by manual calculation )]\n"
   );
printf("\n       WRONGLY PRINTED ANSWERS ARE :- \n (a
   ) Induced EMF, EMF = 471 V instead of %.f V \n",
   Eo)
printf("\n (b) Power angle = 18.05 degree instead of
    %.2f degree \n",pa)
printf("\n (c) Power, P = 12.003 kV instead of %.3f
   kW \n",P)
printf("\n (d) Counter Torque, T = 38.23 N-m instead
    of %.2f N-m \n",T)
printf("\n From Calculation of the Induced EMF(E),
   rest all the Calculated values in the TEXT BOOK
   is WRONG because of the Induced EMF(E) value is
   WRONGLY calculated and the same used for the
   further Calculation part \n")
```

Scilab code Exa 6.17 To find line current pf power torque and excitation EMF and power angle at UPF

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
```

```
// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.17

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA


V = 440;                              // Operating voltage of the Synchronous Motor in pu
E = 200;                              // Induced voltage in Volts
xs = 8.0;                             // Synchronous reactance in Ohms
f = 50;                               // Frequency in Hertz
pa = 36;                              // Power angle in degree


// CALCULATIONS

v = V/sqrt(3);                        // Rated phase voltage in Volts
ws = 2*%pi*f;                         // Synchronous speed in Radians per second
// To calculate the power factor angle refer page no 438 n figure 6.50
// Since E*cos(delta) < v so Power factor is lagging, let power factor angle be theta from ohasor diagram figure 6.50:- page no. 438
// v = E*cos(delta) + I*xs*sin(theta), I*sin(theta) = (254-0.809*200)/8 = 11.525
// Similarly, E*sin(delta) = I*xs*cos(theta), I*cos(theta) = (200*0.59)8 = 14.70
```

```
// From above two equations, tan(theta) = 0.784
theta = -38.1;                                          // Power factor angle in degree (minus sign because of lagging)
pf = cosd(theta);                                       // Power factor lagging
I = 14.7/cosd(theta);                                   // Line current in Amphere (I*cos(theta) = 14.7)
p = 3 * v * 14.7;                                       // Input to motor in watts ( p = 3*V*I*cos(theta), I*cos(theta) = 14.7)
P = (3*E*v*sind(pa))/(xs*1000);                         // Power in Kilo-watts
T = (P*1000)/ws;                                        // Torque in Newton-meter
// For Power factor unity
// let the current will be I2, thus 3*v*I2 = 3*v*I*cos(theta) , I2 = I*cos(theta) = 14.10 A
// let ecitation will be E2, thus v = E2*cos(delta2) and E2*sin(delta2) = I2*xs, E2*cos(delta2) = 254 and E2*sin(delta2) = 117.60, by solving these two equations we get E2 = sqrt(254^2+117.6^2) = 279.90 V and delta2 = atand(117.6/254) = 24.84 degree
E2 = 279.90;
delta2 = 24.84;
P_1 = (3*v*E2*sind(delta2))/(xs*1000);                  // Power in kilo-watts
T_1 = (P_1*1000)/ws;                                    // Torque in Newton-meter


// DISPLAY RESULTS

disp("EXAMPLE : 6.17: SOLUTION :-");
printf("\n (a) Line current, I = %.2f A \n",I)
printf("\n (b) Power factor angle = %.1f degree \n",
```

```
theta)
printf("\n (c) Power , P = %.3f kW \n",P)
printf("\n (d) Torque , T = %.2f N-m \n",T)
printf("\n (e) Power factor = %.2f lagging \n",pf)
printf("\n To make the Power factor to UNITY requirements are:- \n (a) Excitation EMF, E = %.2f V \n",E2)
printf("\n (b) Power angle = %.2f degree \n",delta2)
printf("\n (c) Power , P = %.3f kW \n",P_1)
printf("\n (d) Torque , T = %.2f N-m \n",T_1)
```

Scilab code Exa 6.18 To find excitation EMF per phase line current pf power torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.18

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA


v = 1100;                              // Operating voltage of the Synchronous Motor in Volts
p = 4;                                 // Total number of Poles
m = 3;                                 // number of phase
```

```
xs = 5.0;                          // Synchrouons reactances in Ohms
f = 50;                            // Frequency in Hertz
delta = 9;                         // Power angle in degree
p_hp = 150;                        // Motor delivering power in HP
eta = 89/100;                      // Efficiency of motor


// CALCULATIONS

V = v/sqrt(3);                                       // Phase voltage in Volts
ws = (4*%pi*f)/p;                                    // Synchronous speed in Radians per second
// We have (746*150)/0.89) = 125730.34 W = sqrt(3)*1100*I*cos(theta) refer page no. 440, thus we get I*cos(theta) = 12530.34/(1100*sqrt(3)) = 65.99 and E*sin(delta) = I*xs*cos(theta)
E = (xs*65.99)/sind(delta);                          // Exitation EMF per phase in Volts
// since E*cos(delta) > V, therefore the machine is over excited and power factor is leading, thus we get V = E*cos(delta) + I*xs*sin(theta), I*sin(theta) = (635.1-2109.2*cos(9)/5 = -289.586 and we have I*cos(theta) = 65.99 thus by solving these two equations we get theta = atand(-286.586/65.99) = 77.16 degre
theta = 77.16;                                      // Power factor angle in degree
I = 65.99/cosd(theta);                              // Current in Amphere
pf = cosd(theta);                                   // Power factor leading
P = (3*V*E*sind(delta))/(xs*1000);                  // Power in kilo-Watts
```

```
T = (P*1000)/ws;                                    // torque in Newton-meter


// DISPLAY RESULTS

disp("EXAMPLE : 6.18: SOLUTION :-");
printf("\n (a) Excitation EMF, E = %.1f V \n",E)
printf("\n (b) Line current , I = %.2f A \n",I)
printf("\n (c) Power factor = %.3f leading \n",pf)
printf("\n (d) Power , P = %.4f kW \n",P)
printf("\n (e) Torque , T = %.2f N-m \n",T)
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n      WRONGLY PRINTED ANSWERS ARE :- \n (a) Power , P = 13.0667 kW instaed of %.4f kW \n",P)
printf("\n (b) Torque , T = 83.22 N-m instaed of %.2f N-m \n",T)
printf("\n From Calculation of the Power(P), rest all the Calculated values in the TEXT BOOK is WRONG because of the Power(P) value is WRONGLY calculated and the same used for the further Calculation part \n")
```

Scilab code Exa 6.19 To find efficiency induced voltage torque angle power torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
// CHAPTER : 6 : SYNCHRONOUS MACHINES
```

```

// EXAMPLE : 6.19

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

v = 440;                              // Operating voltage of the Synchronous Motor in Volts
p = 6;                                // Total number of Poles
m = 3;                                // Number of phase
xs = 5;                               // Synchrouons reactances per phase in Ohms
f = 50;                               // Frequency in Hertz
p_hp = 10;                            // Motor delivering power in HP
loss = 1000;                          // Total iron,copper and friction losses in Watts
pf = 0.8;                             // Power factor lagging
I = 10;                               // Motor drawing current in Amphere at 0.8 PF lagging


// CALCULATIONS

V = v/sqrt(3);                                         // Phase voltage in Volts
ws = (4*%pi*f)/p;                                      // Synchronous speed in Radians per second
theta = acosd(pf);                                     // Power factor angle in degree
Po = p_hp*746;                                         // Output power in Watts
Pi = Po+loss;                                          //
```

```
Input power in Watts
eta = (Po/Pi)*100;                                    // Efficiency
// we have V = E*cos(delta) - I*xs*sin(theta), 254 = E*cos(delta) - 5*10*0.6, so E*cos(delta) = 254 + 30 = 284 and E*sin(delta) = I*xs*cos(theta) = 5*10*0.8 = 40 by solving these two equations we get delta = atand(40/284) = 8.01 degree
delta = 8.01;                                         // Power angle in degree
E = 40/sind(delta);                                   // Induced EMF per phase in Volts
P = (3*V*E*sind(delta))/(xs*1000);                    // Power in Kilo-watts
T = (P*1000)/ws;                                      // Torque in Newton-meter


// DISPLAY RESULTS

disp("EXAMPLE : 6.19: SOLUTION :-");
printf("\n (a) Efficiency , eta = %.2f Percent \n",eta)
printf("\n (b) Induced EMF, E = %.f V per phase and \n\n      Power (Torque) angle = %.2f degree \n",E,delta)
printf("\n (c) Power , P = %.4f kW \n",P)
printf("\n (d) Torque , T = %.2f N-m \n",T)
```

Scilab code Exa 6.20 To find induced voltage power and torque in each cases

```

// ELECTRICAL MACHINES
// R.K.Srivastava
```

```
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.20

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA


v = 11*10^3;                    // Operating voltage of the Synchronous Motor in Volts
p = 4;                          // Total number of Poles
m = 3;                          // number of phase
xs = 7;                         // Synchrouons reactances per phase in Ohms
f = 50;                         // Frequency in Hertz
KVA = 1500;                     // KVA rating (whole)
kva = 500;                      // Each case KVA rating


// CALCULATIONS

V = v/sqrt(3);                                  // Phase voltage in Volts
ws = (4*%pi*f)/p;                               // Synchronous speed in Radians per second
I = (sqrt(3)*kva)/v;                            // Phase Current in Amphere

// For Case (a) 0.8 pf lagging

pf_a = 0.8;                                     //
```

```
Power factor lagging
pfa_a = acosd(pf_a);                                    // Power factor angle in degree
// we have E*cos(delta) = V - I*xs*sin(theta) = 6351-78.73*7*0.6 = 6020.334 and E*sin(delta) = I*xs*cos(theta) = 78.73*7*0.8 = 440.888 thus we get by sloving these two equatins E = 6036.46 V and delta = atand(440.888/6020.334) = 4.19 degree
E_a = 6036.46;                                         // Induced Voltage in Volts
delta_a = 4.19;                                        // Power angle in degree
P_a = (3*V*E_a*sind(delta_a))/(xs*10^6);               // Power in Mega-Watts
T_a = (P_a*10^6)/ws;                                   // Torque in Newton-meter

// For Case (b) 0.8 pf leading

pf_b = 0.8;                                             // Power factor lagging
pfa_b = acosd(pf_b);                                    // Power factor angle in degree
// we have E*cos(delta) = V + I*xs*sin(theta) = 6351+78.73*7*0.6 = 6681.666 and E*sin(delta) = I*xs*cos(theta) = 78.73*7*0.8 = 440.888 thus we get by sloving these two equatins E = 6696.2 V and delta = atand(440.888/6681.666) = 3.78 degree
E_b = 6696.2;                                          // Induced Voltage in Volts
delta_b = 3.78;                                        // Power angle in degree
P_b = (3*V*E_b*sind(delta_b))/(xs*10^6);               // Power in Mega-Watts
T_b = (P_b*10^6)/ws;                                   // Torque in Newton-meter

// For Case (c) UPf
```

```

pf_c = 1.0;                                             // Power factor lagging
pfa_c = acosd(pf_c);                                    // Power factor angle in degree
// we have E*cos(delta) = V = 6351 and E*sin(delta) = I*xs = 78.73*7 = 551.11 thus we get by sloving these two equatins E = 6374.9 V and delta = atand(551.11/6351) = 4.96 degree
E_c = 6374.9;                                           // Induced Voltage in Volts
delta_c = 4.96;                                         // Power angle in degree
P_c = (3*V*E_c*sind(delta_c))/(xs*10^6);                // Power in Mega-Watts
T_c = (P_c*10^6)/ws;                                    // Torque in Newton-meter


// DISPLAY RESULTS

disp("EXAMPLE : 6.20: SOLUTION :-");
printf("\n For Case (a) 0.80 pf lagging :- \n (a) Induced EMF, E = %.2f V \n",E_a)
printf("\n (b) Power , P = %.1f MW \n",P_a)
printf("\n (c) Torque , T = %.2f N-m \n",T_a)
printf("\n For Case (b) 0.80 pf leading :- \n (a) Induced EMF, E = %.1f V \n",E_b)
printf("\n (b) Power , P = %.3f MW \n",P_b)
printf("\n (c) Torque , T = %.2f N-m \n",T_b)
printf("\n For Case (a) UPf :- \n (a) Induced EMF, E = %.1f V \n",E_c)
printf("\n (b) Power , P = %.2f MW \n",P_c)
printf("\n (c) Torque , T = %.f N-m \n",T_c)
```

Scilab code Exa 6.21 To find induced EMF torque angle power torque current and pf

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.21

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

v = 440;                              // Operating voltage
    of the Synchronous Motor in Volts
f = 50;                               // Operating
    Frequency of the Synchronous Motor in Hertz
xd = 10;                              // Direct axis
    reactances in Ohms
xq = 7.0;                             // Quadrature axis
    reactances in Ohms
p = 6;                                // Total number of
    Poles
pf = 0.8;                             // Power factor
    lagging
i = 10;                               // Motor drawing
    current in Amphere


// CALCULATIONS

V = v/sqrt(3);                                      //
```

```
Phase voltage in Volts
ws = (4*%pi*f)/p;                                    // Synchronous speed in Radians per second
theta = acosd(pf);                                   // Power factor angle in degree
I = 10*(cosd(theta)+(%i*sind(theta)));               // Motor drawing current in Amphere at 0.8 PF leading
delta = atand((i*xq*cosd(theta))/(V+i*xq*sind(theta)));            // Power angle for motoring mode in degree
Iq = i*cosd(theta+delta);                                          // Current in Amphere
Id = i*sind(theta+delta);                                          // Current in Amphere
Eo = V*cosd(delta) + Id*xd;                                        // Induced EMF in Volts
P = ((3*V*Eo*sind(delta))/xd)+(3*V^2*((1/xq)-(1/xd))*sind(2*delta))/2;        // Power in Watts
T = ((3*V*Eo*sind(delta))/(xd*ws))+(3*V^2*((1/xq)-(1/xd))*sind(2*delta))/(2*ws);        // Torque in Newton-meter

// when the machine is running as alternator, the magnitude of induced EMF = 323.38V. Let the new current will be Inew at lagging power factor thetanew. Now torque angle is 10.71 deg from phasor diagram Figure 6.51 and page no. 444 we get V+Id*xd*cos(delta)-Iq*xq*sin(delta) = Eo*cos(delta), 254+9.825*Id-1.3Iq = 317.75, 9.825*Id-1.3*Iq = 63.75, 7.56*Id-Iq = 49 and we have Id*xd*sin(delta)+Iq*xq*cosdelta) = Eo*sin(delta), 1.85*Id+6.88*Iq = 60.1, 0.27*Id+Iq = 8.74 by solving these two equations we get Idnew =
```

```
    123.85/10.095 = 12.27A and Iqnew = 5.43A
Iqnew = 5.43;                                        // New current in Amphere
Idnew = 12.27;                                       // New current in Amphere
Inew = sqrt(Idnew^2 + Iqnew^2);                      // New total Current in Amphere
// We know that torque angle, tan(delta) = (I*xd*cos(theta))/(V+I*Xq*sin(theta)) so by calutaion for new power factor angle thetanew we get, tan(10.17) = (13.42*7*cos(thetanew))/(254+13.42*7*sin(thetanew)), 0.189(254+13.42*7*sin(thetanew) = 13.42*7*cos(thetanew), 48-93.94cos(thetanew)+17.75*sin(thetanew) = 0 by solving this equatuon we gwt thetanew = 49.5 lagging
thetanew = 49.5;                                     // New power factor angle in degree
pfnew = cosd(thetanew);                              // Power factor lagging


// DISPLAY RESULTS

disp("EXAMPLE : 6.21: SOLUTION :-");
printf("\n (a) Induced EMF, E = %.2f V \n",Eo)
printf("\n (b) Power (Torque) angle = %.2f degree \n",delta)
printf("\n      Power , P = %.2f W \n",P)
printf("\n      Torque , T = %.2f N-m \n",T)
printf("\n (c) when the machine is running as alternator requirements are:- \n\n      New Current = %.2f A\n",Inew)
printf("\n      Power factor = %.3f lagging \n",pfnew)
```

Scilab code Exa 6.22 To find terminal voltages load currents active power output and no load circulating current

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.22

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

E1 = 1100 + (%i*0);                          // EMFs
   of two identicel synchronous Generators in Volts
   per phase
E2 = 1100*(cosd(5)-(%i*sind(5)));            // EMF
   in Volts per phase
Zl = 1.0 + (%i*1.0);                         // Load
   impedance in Ohms per phase
Zs1 = 0.15 + (%i*2.1);                       //
   Synchronous impedance in Ohms per phase
Zs2 = 0.2 + (%i*3.3);                        //
   Synchronous impedance in Ohms per phase
f = 50;                                      //
   Frequency in Hertz


// CALCULATONS

Ys1 = 1/Zs1;
   // Synchronous Admittance in Ohms per phase
Ys2 = 1/Zs2;
```

```
// Synchronous Admittance in Ohms per ohase
Yl = 1/Zl;                       // Load Admittance in Ohms per ohase
V = ((E1*Ys1)+(E2*Ys2))/(Yl+Ys2+Ys1);       // Terminal Voltage in Volts per phase (From Millman's Theorem)
I1 = (E1-V)/Zs1;                 // Individual current in Amphere per phase
I2 = (E2-V)/Zs2;                 // Individual current in Amphere per phase
P1 = abs(V)*abs(I1)*cosd(atand(imag(V),real(V))-atand(imag(I1),real(I1)));                    // Per phase actice power in Watts
P2 = abs(V)*abs(I2)*cosd(atand(imag(V),real(V))-atand(imag(I2),real(I2)));                    // Per phase actice power in Watts
Ic = (E2-E1)/(Zs1+Zs2);          // No-load circulating current in Amphere per phase


// DISPLAY RESULTS

disp("EXAMPLE : 6.22 : SOLUTION :-");
printf("\n (a) Terminal Voltage per phase, V = %.2f < %.1f V \n",abs(V),atand(imag(V),real(V)))
printf("\n (b) Individual currents per phase, I1 = %.f < %.1f A \n\n      I2 = %.1f < %.1f A \n",abs(I1),atand(imag(I1),real(I1)),abs(I2),atand(imag(I2),real(I2)))
printf("\n (c) Per phase Active Power , P1 = %.f W \n\n      P2 = %.1f W \n ",P1,P2)
printf("\n (d) No-load current per phase, Ic = %.2f < %.2f A \n",abs(Ic),atand(imag(Ic),real(Ic)))
```

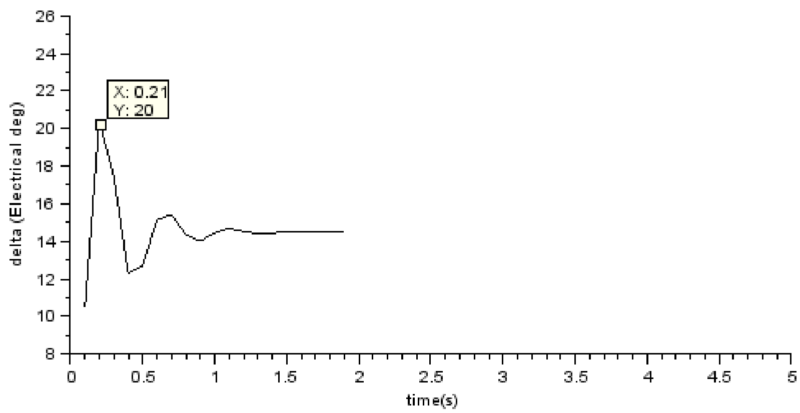

Figure 6.1: To find maximum value of power angle and maximum value of overshoot

Scilab code Exa 6.23 To find maximum value of power angle and maximum value of overshoot

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.23

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA
```

```
p = 4;                                  // Number of the poles in the Alternator
f = 50;                                 // Frequency in Hertz
pkw = 500;                              // Alternator delivering load in kilo-watts
pkwinc = 1000;                          // Generator increases its share of the common elictrical in kilo-watts
Kj = 1.5;                               // Inertia acceleration coefficient for the combined prime mover-alternator in N-m/elec deg/second square
Kd = 12;                                // Damping torque coefficient in N-m/elec deg/second
delta1 = 9;                             // Initial value of the Power angle in degree


// CALCULATIONS

delta2 = (pkwinc/pkw)*delta1;
                                        // Final value (maximum value) of the Power angle in degree (considering Linear variation)
ws = (4*%pi*f)/p;
                                                      // Rotational speed in Radians per second
Ts = (pkw*1000)/ws;
                                                    // Synchornizing torque at 500kW in N-m
Ks = Ts/delta1;
                                                        // Synchornizing torque cofficient at 500kW in N-m/elec-deg
// Laplace transform of the swing Equation can be written as :- s^2 + ((Kd/Kj)*s) + (Ks/Kj) = 0, s^2 + (12/1.5)s + (353.86/1.5) = 0 and compring
```

```
with the standard equation s^2 + s(2*zeta*Wn) + Wn^2 = 0 we get:- mentined below  (refer page no. 454 and 455)
Wn = sqrt(Ks/Kj);                                           // Natural frequency of oscillations in Radians per second
fn = Wn/(2*%pi);                                            // Frequency of natural oscillations in Hertz
zeta = (1*Kd)/(2*Wn*Kj);                                    // Damping ratio
Wd = Wn*(sqrt(1-zeta^2));                                   // Frequency of damped oscillations in radians/s
fd = Wd/(2*%pi);                                            // Frequency of damped oscillations in Hertz
ts = 5/(zeta*Wn);                                           // Settling time in second
deltamax = delta1 + 1.42*(delta2-delta1);                   // The maximum overshoot for damping ratio of 0.2604 is about 42% the maximum appoximate value of the overshoot in terms of 1% tolearance band in Electrical degree


// DISPLAY RESULTS

disp("EXAMPLE : 6.23: SOLUTION :-");
printf("\n (a.1) Final value (maximum value) of the Power angle (considering Linear variation), delta2 = %.f degree \n",delta2)
printf("\n (a.2) Natural frequency of oscillations , Ns = %.2f radians/s \n",Wn)
printf("\n (a.3) Damping ratio , zeta = %.4f \n",zeta
```

```
)
printf("\n (a.4) Frequency of damped oscillations ,
    Wd = %.2f radians/s \n",Wd)
printf("\n (a.5) Settling time , ts = %.2f seconds \n
    ",ts)
printf("\n  (b)  The maximum overshoot for damping
    ratio of 0.2604 is about 42 percent the maximum
    appoximate value of the overshoot in terms of 1
    percent tolearance band is , deltamax = %.2f
    degree \n",deltamax)
printf("\n\n FOR CASE (C) CANNOT BE DO IT IN THIS
    BECAUSE AS IT REQUIRES MATLAB SIMULINK \n")
```

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 6.25 To find reactive kVA of the motor and overall pf of the motor and load and resultant current

```

// ELECTRICAL MACHINES
// R.K.Srivastava
//  First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.25

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

```

```
v = 440;                                // Operating voltage of the Synchronous generator in Volts
f = 50;                                 // Operating Frequency of the Synchronous generator in Hertz
m = 3;                                  // Total number of Phase
pf = 0.8;                               // Power factor lagging
Il = 100;                               // Motor drawing current in Amphere
xs = 2;                                 // Synchronous reactances in Ohms
delta = 20;                             // Power angle in degree
P = 50*10^3;                            // Total Power developed by the motor in Watts
Ppp = (50*10^3)/3;                      // Power developed by the motor per phase in Watts


// CALCULATIONS

V = v/sqrt(3);                                        // Phase voltage in Volts
Eo = (Ppp*xs)/(3*V*sind(delta));                      // Per phase Induced voltage in Volts
// Let us assume thetam is Power factor angle in degree and Im is the Motor current now, from phasor diagram figure 6.67 page no. 465 we get, Eo*cosd(delta) = V+Im*xs*sind(thetam), Im*sind(thetam) = ((383.84*cosd(20))-254.03)/2 = 53.35 and Im*xs*cosd(thetam) = Eo*sin(delta), Im*cosd(theta) = ((383.84*sind(20))/2 = 65.60 by sloving these two equations we get Im = sqrt(65.60^2 + 53.35^2) = 84.56 A and thetam = atand (53.35/65.60) = 39.13 degree
Im = sqrt(65.60^2 + 53.35^2);
                                   // Motor current in
```

```
Amphere
thetam = atand(53.35/65.60);                                    // Power factor angle in degree
kVA = (sqrt(3)*V*Im*sind(thetam))/1000;                        // Rective kVA of the motor in kVAR
thetal = acosd(pf);                                              // Load power factor angle in degree
thetaR = atand((Im*sind(thetam)-Il*sind(thetal))/(Im*cosd(thetam)+Il*cosd(thetal)));   // Resultant Power factor angle in degree
ovpf = cosd(thetaR);                                             // Overall Power factor lagging
IR = sqrt((Im*sind(thetam)-Il*sind(thetal))^2 + (Im*cosd(thetam)+Il*cosd(thetal))^2);   // Resultant (magnitude) current in Amphere refer phasor diagram figure 6.69 page no. 467


// DISPLAY RESULTS

disp("EXAMPLE : 6.25: SOLUTION :-");
printf("\n (a) Rective kVA of the motor = %.3f kVAR \n",kVA)
printf("\n (b) Overall Power factor of the load and motor = %.4f lagging and \n",ovpf)
printf("\n     Resultant (magnitude) current = %.2f A \n",IR)
```

Scilab code Exa 6.27 To find Overall and individual currents of the machines and pf

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.27

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

v = 440;                          // Operating voltage
    of the Synchronous generator in Volts
f = 50;                           // Operating
    Frequency of the Synchronous generator in Hertz
m = 3;                            // Total number of
    Phase
xs = 5;                           // Synchronous
    reactances in Ohms
Eo = 500;                         // Indduced Voltage
    in Volts per phase
R1 = 0.1;                         // Circuit Parameter
    in Ohms
R2 = 0.1;                         // Circuit Parameter
    in Ohms
X1 = 1.55;                        // Circuit Parameter
    in Ohms
X2 = 1.55;                        // Circuit Parameter
    in Ohms
s = 0.03;                         // Slip
P = 30*10^3;                      // Total Power
```

```
developed by the motor in Watts


// CALCULATIONS

V = v/sqrt(3);                                        // Phase voltage in Volts
Ii = V/sqrt((R1+R2/s)^2 + (X1+X2)^2);                 // Per phase induction motor current in Amphere
thetal = atand((X1+X2)/(R1+R2/s));                    // Power factor angle of the induction motor in degree
pf = cosd(thetal);                                    // Power factor of the induction motor lagging
// Let us assume thetam is leading Power factor angle in degree and Im is the synchronous Motor current now, from phasor diagram figure 6.70 page no. 469
delta = asind((xs*P)/(3*V*Eo));                       // Power angle in degree
// From phasor diagram figure 6.70 page no. 469 we have, Im*xs*cos(thetam) = Eo*sin(delta), Im*cos(delta) = ((500*sind(23.18))/5 = 39.37 and Eo*cosd(delta) = V+Im*xs*sind(thetam), Im*sind(thetam) = ((500*cosd(23.18))-254.03)/5 = 41.12 by sloving these two equations we get Im = sqrt(39.37^2 + 41.12^2) = 56.93 A and thetam = atand(41.12/39.37) = 46.25 degree
Im = sqrt(39.37^2 + 41.12^2);                         // Motor current in Amphere
thetam = atand(41.12/39.37);                          // Power factor angle in degree
kVA = (sqrt(3)*V*Im*sind(thetam))/1000;               // Rective kVA of the motor in kVAR
II = Ii * exp( %i * (-thetal) * %pi/180);
```

```
                    // Induction Motor current in
   Amphere
Im = Im * exp( %i * thetam * %pi/180);
                    // Synchronous Motor current in
   Amphere
It = II + Im;
                                              //
   Total per phase current in Amphere
ovpf = cosd(atand(imag(It),real(It)));
                    // Overall Power factor leading


// DISPLAY RESULTS

disp("EXAMPLE : 6.27: SOLUTION :-");
printf("\n (a) Reactive kVA of the motor = %.3f kVAR
    \n",kVA)
printf("\n (b) Individual currents:- \n
   Induction Motor current , II = %.2f + i(%.2f) A \n
   \n      Synchronous Motor current , Im = %.2f + i(%
   .2f) A \n",real(II),imag(II),real(Im),imag(Im))
printf("\n (c) Resultant (overall) current = %.2f <
   %.2f A \n",abs(It),atand(imag(It),real(It)))
printf("\n (d) Overall Power factor = %.4f leading \
   n",ovpf)
```

Scilab code Exa 6.28 To find torque and operating speed

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES
```

```
// EXAMPLE : 6.28

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA


V = 400;                        // Operating voltage
    of the Synchronous generator in Volts
f = 50;                         // Operating
    Frequency of the Synchronous generator in Hertz
xd = 12;                       // Direct axis
    reactances in Ohms
xq = 5;                        // Quadrature axis
    reactances in Ohms
delta = 15;                    // Power(Torque) angle
     in degree
p = 2;                         // Number of the poles
m = 3;                         // Number of the phase


// CALCULATIONS

v = V/sqrt(3);
                                                // Rated
     Phase Votage in Volts
Ns = (120*f)/p;
                                               //
    Operating speed in RPM
Ws = (2*%pi*f)/(p/2);
                                         // Synchronous
    speed in radians/s
T = (3*v^2*sind(2*delta)/(2*Ws))*((1/xq)-(1/xd));
          // Developed Torque in Newton-meter

```

```
// DISPLAY RESULTS

disp("EXAMPLE : 6.28: SOLUTION :-");
printf("\n (a) Operating speed, Ns = %.f RPM \n",Ns)
printf("\n (b) Developed Torque , T = %.2f N-m \n",T)
```

Scilab code Exa 6.29 To find torque and operating speed

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.29

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

f = 400;                                    // Operating Frequency of the Synchronous generator in Hertz
Ld = 50*10^-3;                              // Direct axis reactances in Henry
Lq = 15*10^-3;                              // Quadrature axis reactances in Henry
delta = 15;                                 // Power(Torque) angle in degree
p = 2;                                      // Number of
```

```
the poles
m = 3;                                   // Number of the phase
I = 10;                                  // Operating current in Amphere


// CALCULATIONS

Ns = (120*f)/p;                                                          // Operating speed in RPM
Ws = (2*%pi*f)/(p/2);                                              // Synchronous speed in radians/s
xd = 2*%pi*f*Ld;                                                      // Direct axis reactances in reactance
xq = 2*%pi*f*Lq;                                                      // Quadrature axis reactances in reactance
E1 = 0;                                              // Induced EMF in Volts (Its ZERO beacuse when field winding current is zero)
v = xq*I;                                            // Applied voltage in Volts
T = (3*v^2*sind(2*delta)/(2*Ws))*((1/xq)-(1/xd));            // Developed Torque in Newton-meter


// DISPLAY RESULTS

disp("EXAMPLE : 6.29: SOLUTION :-");
printf("\n (a) Operating speed , Ns = %.f RPM \n",Ns)
printf("\n (b) Developed Torque , T = %.5f N-m \n",T)
```

```
printf("\n\n      [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n        WRONGLY PRINTED ANSWERS ARE :-  (a) xd = 12.56 instead of %.2f Ohms \n ",xd);
printf("\n                                      (b) xq = 3.768 instead of %.2f Ohms \n ",xq);
printf("\n                                      (c) v = 36.68 instead of %.2f V \n ",v);
printf("\n                                      (d) T = 0.07875 instead of %.4f N-m \n ",T);
printf("\n From Calculation of the d-axis and q-axis reactance (xd and xq respectively), rest all the Calculated values in the TEXT BOOK is WRONG because of the d-axis and q-axis reactance (xd and xq respectively) value is WRONGLY calculated and the same used for the further Calculation part \n")
```

Scilab code Exa 6.30 To find power and torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.30

clear ; clc ; close ; // Clear the work space and console


```

```
// GIVEN DATA

f = 50;                         // Operating Frequency of the Synchronous generator in Hertz
p = 2;                          // Number of the poles
Pt = 800;                       // Total loss in Watts
Pr = 10;                        // Rotational loss in Watts


// CALCULATIONS

Ws = (4*%pi*f)/p;                                    // Synchronous speed in radians/s
Ph = Pt-Pr;                                          // Hysteresis loss refered to stator in Watts
Th = Ph/Ws;                                          // Torque at the shaft in Newton-meter


// DISPLAY RESULTS

disp("EXAMPLE : 6.30: SOLUTION :-");
printf("\n (a) Power at the shaft , Ph = %.f W \n",Ph)
printf("\n )b) Torque at the shaft , Th = %.2f N-m \n",Th)
```

Scilab code Exa 6.31 To find induced EMF per phase torque angle and power in each cases

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD
// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.31

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

Pi = 2*10^6;                                        // Power
    input in Volt-Amphere
v = 6.6*10^3;                                       //
   Operating voltage in Volts


// CALCULATIONS

I = Pi/(v*sqrt(3));                                 // Rated
    current in Amphere
V = v/sqrt(3);                                      // Phase
    voltage in Volts
xs = v/(I*sqrt(3));                                 //
   Synchronous reactance in Ohms

// For case (a) 0.8 pf lagging

pf_a = 0.8;                                         // Power
    factor
pfa_a = acosd(pf_a);                                // Power
    factor angle in degree
a_a = (V + (I*xs*sind(pfa_a)));
b_a = (I*xs*cosd(pfa_a));
E_a = sqrt(a_a^2 + b_a^2);                          //
```

```
    Induced EMF in Volts
delta_a = atand(b_a/a_a);                      // Torque (power) angle in degree
P_a = (3*V*E_a*sind(delta_a))/(xs*10^6);      // Power developed in MVA

// For case (b) 0.8 pf leading

pf_b = 0.8;                                    // Power factor
pfa_b = acosd(pf_b);                           // Power factor angle in degree
a_b = (V - (I*xs*sind(pfa_a)));
b_b = (I*xs*cosd(pfa_b));
E_b = sqrt(a_b^2 + b_b^2);                     // Induced EMF in Volts
delta_b = atand(b_b/a_b);                      // Torque (power) angle in degree
P_b = (3*V*E_b*sind(delta_b))/(xs*10^6);      // Power developed in MVA

// For case (c) UPF

pf_c = 1.0;                                    // Power factor
pfa_c = acosd(pf_c);                           // Power factor angle in degree
a_c = V;
b_c = I*xs;
E_c = sqrt(a_c^2 + b_c^2);                     // Induced EMF in Volts
delta_c = atand(b_c/a_c);                      // Torque (power) angle in degree
P_c = (3*V*E_c*sind(delta_c))/(xs*10^6);      // Power developed in MVA


disp("EXAMPLE : 6.31: SOLUTION :-");
```

```
printf("\n For Case (a) 0.80 lagging Power factor \n   Induced EMF, EMF = %.2f V \n",E_a)
printf("\n Power (Torque) angle = %.2f degree \n",delta_a)
printf("\n Power developed , P = %.1f MVA \n",P_a)
printf("\n For Case (b) 0.80 leading Power factor \n   Induced EMF, EMF = %.f V \n",E_b)
printf("\n Power (Torque) angle = %.2f degree \n",delta_b)
printf("\n Power developed , P = %.3f MVA \n",P_b)
printf("\n For Case (c) Unity Power Factor \n Induced EMF, EMF = %.1f V \n",E_c)
printf("\n Power (Torque) angle = %.2f degree \n",delta_c)
printf("\n Power developed , P = %.1f MVA \n",P_c)
printf("\n\n     [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n       WRONGLY PRINTED ANSWERS ARE :-   xs =  20.14 instead of %.2f Ohms \n ",xs);
printf("\n  For Case (a) 0.80 lagging Pf (a.1)  E = 6561.42 instead of %.2f V \n ",E_a);
printf("\n                                (a.2) delta = 25.45 instead of %.2f degree \n ",delta_a);
printf("\n  For Case (b) 0.80 leading Pf (b.1)  E = 3290 instead of %.1f V \n ",E_b);
printf("\n                                (b.2) delta = 58.98 instead of %.2f degree \n ",delta_b);
printf("\n                                (b.3) Power developed = 1.617 instead of %.3f MVA \n ",P_b);
printf("\n  For Case (c) UPF  (c.1)  E = 5190.2 instead of %.2f V \n ",E_c);
printf("\n                    (c.2)  delta = 42.77 instead of %.2f degree \n ",delta_c);
printf("\n In all the three cases from Calculation
```

```
of the Synchronous reactance (xs), rest all the Calculated values in the TEXT BOOK is WRONG because of the Synchronous reactance (xs) value is WRONGLY calculated and the same used for the further Calculation part \n")
```

Scilab code Exa 6.32 To find induced EMF per phase power angle and percent regulation

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.32

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA
// Refer phasor diagram figure 6.76 and page no. 476

pf = 0.8;                                                    // Power factor lagging
pa = acosd(pf);                                              // Power factor angle in degree
v = 1.0 * exp( %i * pa * %pi/180);                           // Operating voltage of the alternator in pu
```

```
xd = 0.8;                                                // Direct axis reactances in pu
xq = 0.4;                                                // Quadrature axis reactances in pu
I = 1.0;                                                 // Current in pu taking this as reference


// CALCULATIONS

A = v + (%i*xq*I);
delta = atand(imag(A),real(A))-pa;                       // Power angle in degree
Iq = I * cosd(atand(imag(A),real(A)));                   // d-axis currents in Amphere

Id = I * sind(atand(imag(A),real(A)));                   // q-axis currents in Amphere
E = abs(v)*cosd(delta) + Id*xd;                          // Induced EMF per phase in Per unit
pr = ((abs(E)-abs(v))/abs(v))*100;                       // Percentage regulation

// DISPLAY RESULTS

disp("EXAMPLE : 6.32: SOLUTION :-");
printf("\n (a) Induced EMF per phase, E = %.4f < %.2f pu \n",E,delta)
printf("\n (b) Power angle = %.2f degree \n",delta)
printf("\n (C) Percenatge Regulation, R = %.2f Percent \n",pr)
printf("\n\n IN THIS PROBLEM PERCENTAGE REGULATION IS NOT CALCULATED IN THE TEXT BOOK\n")
```

Scilab code Exa 6.33 To find the ratio of the maximum electromagnetic torque to the actual electromagnetic torque

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 6 : SYNCHRONOUS MACHINES

// EXAMPLE : 6.33

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

v = 6.6*10^3;                      // Operating voltage of the Synchronous motor in Volts
P = 5*10^6;                        // Operating power of the Synchronous motor in Watts
pf = 1.0;                          // Power factor
xd = 3.0;                          // Direct axis reactances in Ohms
xq = 1.0;                          // Quadrature axis reactances in Ohms
eta = 0.98;                        // OPerating efficiency


// CALCULATIONS

```

```
V = v/sqrt(3);                                  // Per phase voltage in Volts
I = P/(eta*v*sqrt(3));                          // Line current in Amphere
delta = atand((xq*I)/v);                        // power angle in degree
E = v*cosd(delta) + xd*I*sind(delta);           // Induced EMF in Volts
Tmax = ((3*E*V*sind(90))/xd) + ((3*V^2*sind(180))/2)*((1/xq)-(1/xd));
    // Maximum electromagnetic torque in N-m
T = ((3*E*V*sind(delta))/xd) + ((3*V^2*sind(2*delta))/2)*((1/xq)-(1/xd));
                                // Actual electromagnetic torque in N-m
Ratio = Tmax/T;                                 // Ratio of the Maximum electromagnetic torque to the actual electromagnetic torque


// DISPLAY RESULTS

disp("EXAMPLE : 6.33: SOLUTION :-");
printf("\n (a) Ratio of the Maximum electromagnetic torque to the actual electromagnetic torque is %.2f \n",Ratio)
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED WRONGLY ( I verified by manual calculation )]\n");
printf("\n      WRONGLY PRINTED ANSWERS ARE :- (a) delta = 2.41 instead of %.2f degree \n ",delta);
printf("\n                                      (b) E = 6379 instead of %.2f V \n ",E);
printf("\n                                      (c) Ratio = 10.84 instead of %.2f \n ",Ratio);
printf("\n From Calculation of the Power angle (delta), rest all the Calculated values in the TEXT BOOK is WRONG because of the Power angle (
```

```
delta) value is WRONGLY calculated and the same
used for the further Calculation part \n")
```

Chapter 7

Special Motors and Introduction to Generalized Machines Theory

Scilab code Exa 7.1 To find induced EMF

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO
   GENERALIZED MACHINE THEORY

// EXAMPLE : 7.1

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

```

```
D = 35*10^-2;                                         // Outer diameter of the conducting disk in Meter
d = 10*10^-2;                                         // Inner diameter of the conducting disk in Meter
B = 1.0;                                              // Axial magnetic field in Telsa
N = 900;                                              // Rotating shaft running in RPM


// CALCULATIONS

Wr = (2*%pi*N)/60;                                    // Rotational angular speed in radians/s
Er = ((D^2-d^2)*B*Wr)/8;                              // EMF induced in Volts


// DISPLAY RESULTS

disp("EXAMPLE : 7.1: SOLUTION :-");
printf("\n (a) Induced EMF in the outer and inner rims of the disk , Er = %.4f V \n",Er)
```

Scilab code Exa 7.2 To find torque and number of the conductors required

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO GENERALIZED MACHINE THEORY

```

```
// EXAMPLE : 7.2

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

D = 0.120;                  // Outer Radius of the
    Printed Circuit Motor in meter
d = 0.060;                  // Inner Radius of the
    Printed Circuit Motor in meter
  B = 0.7;                  // Axial Flux Density in
       Telsa
V = 12;                     // Volage Supplied to
    the Motor in Volts
R = 2700;                   // Motor Speed in RPM
n = 0.65;                   // Efficiency of Motor
p = 94;                     // Output Power of Motor
     in Watts
I = 12;                     // Motor current in
    Ampheres


// CALCULATIONS

w = ((2*(%pi))*R)/60;        // The Angular Velocity
     in Radians/second
T = p/w;                     // Torque in Newton-
    Meter
N = (8*T)/((D^2-d^2)*B*I)    // Total Number of
    Conductors


// DISPLAY RESULTS

disp("EXAMPLE : 7.2 : SOLUTION :-") ;
printf("\n (a) Torque, T = %.2f N-m \n ",T);
```

```
printf("\n (b) Total Number of Conductors , N = %.2f nearly 30 \n",N);
```

Scilab code Exa 7.3 To find resultant torque and stator phase currents

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO GENERALIZED MACHINE THEORY

// EXAMPLE : 7.3

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

m = 2;                                        // Total number of phase in servo Motor
f = 50;                                       // Frequency in Hertz
V = 220;                                      // Operating Voltage of the servo Motor in Volts
R1 = 250;                                     // Circuit Parameter in Ohms
R2 = 750;                                     // Circuit Parameter in Ohms
X1 = 50;                                      // Circuit Parameter in Ohms
X2 = 50;                                      //
```

```
Circuit Parameter in Ohms
Xm = 1000;                                              // Circuit Parameter in Ohms
s = 0.6;                                                // Slip
Va = 220;                                               // Unbalanced Voltage in Volts
Vb = 150 * exp(%i * (-60) * %pi/180);                   // Unbalanced Voltage in Volts


// CALCULATIONS

Va1 = (Va + %i*Vb)/2;                                   // Positive sequence voltage in Volts
Va2 = (Va - %i*Vb)/2;                                   // Negative sequence voltage in Volts
Z11 = (R1+%i*X1);
Z12 =  (((%i*Xm)*(R2/s+%i*X2))/(%i*Xm+R2/s+%i*X2));
Z1 = Z11 + Z12 ;                                        // Positive sequence impedance in Ohms
Z2 = (R1+%i*X1) + (((%i*Xm)*(R2/(2-s)+%i*X2))/(%i*Xm+R2/(2-s)+%i*X2));    // Negative sequence impedance in Ohms
Ia1 = Va1/Z1;                                           // Positive sequence current in Amphere
I12 = (Ia1*Z12)/(R2/s);                                 // Positive sequence current in Amphere
Ia2 = Va2/Z2;                                           // Negative sequence current in Amphere
I22 = (Ia2*Z2)/(R2/(2-s));                              // Negative
```

```
sequence current in Amphere
T1 = 2*(abs(I12)^2)*R2/s;                                    // Positive sequence torque in Newton-meter
T2 = 2*(abs(I22)^2)*R2/(2-s);                                // Negative sequence torque in Newton-meter
T = T1 - T2;                                                 // Resultant torque in Newton-meter
Ia = Ia1 + Ia2;                                              // Line current in Amphere
Ib = (-%i*Ia1) + (%i*Ia2);                                   // Line current in Amphere


// DISPLAY RESULTS

disp("EXAMPLE : 7.3: SOLUTION :-");
printf("\n (a) Resultant torque , T = %.2f N-m \n",T)
printf("\n (b) Phase currents (line currents), Ia = %.2f < %.2f A \n\n                                        Ib = %.2f < %.2f \n",abs(Ia),atand(imag(Ia),real(Ia)),abs(Ib),atand(imag(Ib),real(Ib)))
printf("\n\n IN THE ABOVE PROBLEM ALL THE VALUES PRINTED IN THE TEXT BOOK ARE NOT ACCURATE, SO VALUE OF THE TORQUE AND LINE CURRENTS ARE DIFFERING, WHEN WE COMPARED TO THE TEXT BOOK ANSWERS FOR THE SAME. \n\n")
printf("\n IN EVERY CALCULATED PARAMETER IN THE TEXT BOOK SLIGHT VARIATION IS THERE AS WE COMPARED TO MANUAL CALCULATION ITS FROM POSITIVE SEQUENCE VOLTAGE (Va1) \n")
```

Scilab code Exa 7.4 To find the resultant torque

```

// ELECTRICAL MACHINES
// R.K. Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO GENERALIZED MACHINE THEORY

// EXAMPLE : 7.4

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

m = 2;                                          // Total number of phase in AC drag-cup servo Motor
p = 2;                                          // Number of poles
Va = 220;                                       // Operating Voltage of the servo Motor in Volts
R1 = 350;                                       // Circuit Parameter in Ohms
R2 = 250;                                       // Circuit Parameter in Ohms
X1 = 60;                                        // Circuit Parameter in Ohms
X2 = 50;                                        // Circuit Parameter in Ohms
Xm = 900;                                       //
```

```
Circuit Parameter in Ohms
s = 0.3;                                         // Slip
p = 0.8;                                         // Ratio of the control winding and main winding voltage


// CALCULATIONS

Va1 = (Va*(1+p))/2;                                   // Positive sequence voltage in Volts
Va2 = (Va*(1-p))/2;                                   // Negative sequence voltage in Volts
Z11 = (R1+%i*X1);
Z12 =  (((%i*Xm)*(R2/s+%i*X2))/(%i*Xm+R2/s+%i*X2));
Z1 = Z11 + Z12 ;                                      // Positive sequence impedance in Ohms
Z2 = (R1+%i*X1) + (((%i*Xm)*(R2/(2-s)+%i*X2))/(%i*Xm+R2/(2-s)+%i*X2));    // Negative sequence impedance in Ohms
Ia1 = Va1/Z1;                                         // Positive sequence current in Amphere
I12 = (Ia1*Z12)/(R2/s);                               // Positive sequence current in Amphere
Ia2 = Va2/Z2;                                         // Negative sequence current in Amphere
I22 = (Ia2*Z2)/(R2/(2-s));                            // Negative sequence current in Amphere
T1 = 2*(abs(I12)^2)*R2/s;                             // Positive
```

```
    sequence torque in Newton−meter
T2 = 2*(abs(I22)^2)*R2/(2-s);                                    // Negative sequence torque in Newton−meter
T = T1 - T2;                                                      // Resultant torque in Newton−meter


// DISPLAY RESULTS

disp("EXAMPLE : 7.4: SOLUTION :−");
printf("\n (a) Resultant torque , T = %.2f N−m \n",T)
printf("\n\n IN THE ABOVE PROBLEM ALL THE VALUES PRINTED IN THE TEXT BOOK ARE NOT ACCURATE, SO VALUE OF THE TORQUE AND LINE CURRENTS ARE DIFFERING WHEN WE COMPARED TO THE TEXT BOOK ANSWERS FOR THE SAME. \n\n")
printf("\n IN EVERY CALCULATED PARAMETER IN THE TEXT BOOK SLIGHT VARIATION IS THERE AS WE COMPARED TO MANUAL CALCULATION ITS FROM POSITIVE SEQUENCE IMPEDANCE (Z1) \n")
```

Scilab code Exa 7.5 To find speed and pf

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO GENERALIZED MACHINE THEORY

// EXAMPLE : 7.5
```

```
clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

R = 15;                // Resistance of the fractional horse power AC series motor in Ohms
V = 230;               // AC supply voltage in Volts
f = 50;                // Frequency in Hertz
I = 1.2;               // Motor current in Amphere
NDC = 2500;            // Rotating speed of the Motor during DC Operation in RPM
L = 0.5;               // Total inductance in Henry


// CALCUALTIONS

X = 2*%pi*f*L;                                              // Reactance in Ohms
NAC = NDC * (sqrt(V^2-(I*X)^2)-(I*R)) / (V-(I*R));          // Rotating speed of the Motor during AC Operation in RPM
pf = sqrt(1-((I*X)/V)^2);                                   // Power factor lagging


// DISPLAY RESULTS

```

```
disp("EXAMPLE : 7.5: SOLUTION :-");
printf("\n When the Motor operting at AC 230V, 50 Hz
    \n\n (a) NAC = %.f RPM \n",NAC)
printf("\n (b) Power factor = %.4f lagging \n",pf)
```

Scilab code Exa 7.6 To find current pf torque at rated condition and speed pf torque and efficiency at load current is halved

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO
    GENERALIZED MACHINE THEORY

// EXAMPLE : 7.6

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

R = 1.4;
    // Total Resistance of the AC series motor in
    Ohms
V = 115;
    // supply voltage in Volts
f = 50;
    // Frequency in Hertz
N = 5000;
    // Rotating speed in RPM
X = 12;
```

```
// Total reactance in Ohms
P = 250;                // Electrical power output in Watts
loss = 18;              // Rotational losses in Watts


// CALCULATIONS

Pd = P + loss;          // Mechanical power developed in Watts
// We know that Er = Pd/I and from phasor diagram in figure 7.11 page no. 501 V^2 = (Er+I*R)^2+(I*X)^2, 115^2 = (268/I-1.4*I)^2+(12*I)^2, 13225*I^2 = 71824+2.036*I^4-750.4*I^2+144*I^2, solving this we get 2.036*I^4-13831.4*I^2+71824 = 0, I^4-6793.42*I^2+3577 = 0 this gives I = 2.28A or 82.38A (The above calculation part is wrong )
i = poly ([3577 0 -6793.42 0 1],'x','coeff');          // Expression for the Current in Quadratic form
a = roots (i);          // 4-Value of the current in Amphere
I = a(4,1);             // Curent in Amphere neglecting higher value and negative value

pf_a = sqrt(1-((I*X)/V)^2);          // Power factor lagging
Er_a = sqrt(V^2-(I*X)^2)-(I*R);      // Rotational Voltage in Volts
T_a = (Er_a*I)/(2*%pi*N/60);         // Developed
```

```
torque in Newton-meter
Ih = I/2;                                   // Current halved in Amphere
pf_b = sqrt(1-((Ih*X)/V)^2);                // Power factor lagging when load current halved
Er_b = sqrt(V^2-(Ih*X)^2)-(Ih*R);           // Rotational Voltage in Volts when load current halved
N2 = (N*Er_b*I)/(Er_a*Ih);                  // New speed in RPM when load current halved
T_b = (Er_b*Ih)/(2*%pi*N2/60);              // Developed torque in Newton-meter when load current halved
eta = 100*(Er_b*Ih)/(V*Ih*pf_b);            // Efficiency when load current halved


// DISPLAY RESULTS

disp("EXAMPLE : 7.6: SOLUTION :-");
printf("\n At rated condition, \n\n (a.1) Current, I = %.2f A \n",I)
printf("\n (a.2) Power factor = %.3f lagging \n",pf_a)
printf("\n (a.3) Developed torque = %.2f N-m \n",T_a)
printf("\n When load current halved (reduced to half), \n\n (b.1) Speed, N2 = %.f RPM \n",N2)
printf("\n (b.2) Power factor = %.4f lagging \n",pf_b)
printf("\n (b.3) Developed torque = %.2f N-m \n",T_b)
printf("\n (b.4) Efficiency = %.1f percenatge \n",eta)
```

```
printf("\n From Calculation of the Current(I), rest all the Calculated values in the TEXT BOOK is WRONG because of the Current equation and its value both are WRONGLY calculated and the same used for the further Calculation part, so all the values are in the TEXT BOOK IS WRONG \n")
```

Scilab code Exa 7.7 To find line current pf and efficiency of motor

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO GENERALIZED MACHINE THEORY

// EXAMPLE : 7.7

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

V = 220;                // supply voltage in Volts
f = 50;                 // Frequency in Hertz
p = 4;                  // Number of poles
Xm = 50;                // Mutual reactance in Ohms
Rs = 0.4;
```

```
// Resistance of stator windings in Ohms
Xs = 2.5; // Leakage reactance of stator windings in Ohms
Ra = 2.2; // Resistance of Armature windings in Ohms
Xa = 3.1; // Leakage reactance of armature windings in Ohms
loss = 30; // Rotational losses in Watts
N = 2000; // Motor running speed in RPM


// CALCULATIONS

Ns = (120*f)/p; // Synchronous speed in RPM
s = N/Ns; // Speed ratio
I1 = V/(2*Rs + 2*%i*Xs + 2*%i*Xm + (%i*Xm^2)*((s-%i)/(Ra+%i*Xa+%i*Xm)));// line current in Amphere
pf = cosd(atand(imag(I1),real(I1))); // Power factor lagging
I2 = (s-%i)*(%i*Xm*I1)/(Ra+%i*Xa+%i*Xm); // line current in Amphere
P1 = V*abs(I1)*cosd(atand(imag(I1),real(I1))); // Input power in Watts
Pm = P1 - 2*(abs(I1)^2)*Rs - (abs(I2)^2)*Ra; // Mechanical power developed in Watts
Po = Pm - loss; // output power in Watts
eta = 100*(Po/P1);
```

```
    // Efficiency


// DISPLAY RESULTS

disp("EXAMPLE : 7.7: SOLUTION :-");
printf("\n (a) Line currents , I1 = %.2f < %.2f A and
     I2 = %.2f < %.2f A \n",abs(I1),atand(imag(I1),
    real(I1)),abs(I2),atand(imag(I2),real(I2)))
printf("\n (b) Power factor = %.2f lagging \n",pf)
printf("\n (c) Efficiency = %.2f percentage \n",eta)
printf("\n\n    [ TEXT BOOK SOLUTION IS PRINTED
    WRONGLY ( I verified by manual calculation )]\n"
    );
printf("\n       WRONGLY PRINTED ANSWERS ARE :- (a)
    I1 = 3.37 < -42.78 A instead of %.2f < j(%.2f) A
    \n ",abs(I1),atand(imag(I1),real(I1)));
printf("\n                                         (b)
    I2 = 5.26 < -77.34 A instead of %.2f < j(%.2f) A
    \n ",abs(I2),atand(imag(I2),real(I2)));
printf("\n                                         (b)
    eta = 81.53 percent instead of %.2f percent  \n "
    ,eta)
printf("\n From Calculation of the I1, rest all the
    Calculated values in the TEXT BOOK is WRONG
    because of the I1 value is WRONGLY calculated and
     the same used for the further Calculation part \
    n")
```

Scilab code Exa 7.8 To find current and pf of the motor

```

// ELECTRICAL MACHINES
// R.K.Srivastava
```

```
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO GENERALIZED MACHINE THEORY

// EXAMPLE : 7.8

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

V = 220;                // supply voltage in Volts
f = 50;                 // Frequency in Hertz
p = 4;                  // Number of poles
Xm = 60;                // Mutual reactance in Ohms
Rs = 1.0;               // Resistance of stator windings in Ohms
Xs = 6.0;               // Leakage reactance of stator windings in Ohms
Ra = 2.5;               // Resistance of Armature windings in Ohms
Xa = 6.0;               // Leakage reactance of armature windings in Ohms
P_hp = 1;               // Output power in HP
N = 1400;               // Motor running speed in RPM
alpha = 15;             // Brush displacement from the low-impedance position in degree

```

```
// CALCULATIONS

Ns = (120*f)/p;                                    // Synchronous speed in RPM
s = N/Ns;                                          // Speed ratio
I = V / (Rs + %i*(Xs+Xm) + (%i*Xm^2*cosd(alpha))*(s*
    sind(alpha)-(%i*cosd(alpha)))/(Ra+%i*(Xa+Xm)));        // Curent in Amphere
pf = cosd(atand(imag(I),real(I)));                           // Power factor lagging


// DISPLAY RESULTS

disp("EXAMPLE : 7.8: SOLUTION :-");
printf("\n (a) Currents , I = %.2f < %.2f A \n",abs(I
    ),atand(imag(I),real(I)))
printf("\n (b) Power factor = %.4f lagging \n",pf)
```

Scilab code Exa 7.9 To find no load slips

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO
    GENERALIZED MACHINE THEORY

```

```
// EXAMPLE : 7.9

clear ; clc ; close ; // Clear the work space and console


// GIVEN DATA

E2 = 100;                                              // Per phase standstill EMF in Volts
Z2s = 0.025 + %i*0.08;                                 // Rotor circuit impedance at standstill
E = 50;                                                // Injected EMF in Volts


// CALCULATIONS

I2 = 0;                                                // Assuming Current is zero
s1 = (E/E2)+(I2*Z2s)/E2;                               // Slip when injected EMF is opposite to the E2
s2 = (-E/E2)+(I2*Z2s)/E2;                              // Slip when injected EMF is phase with E2


// DISPLAY RESULTS

disp("EXAMPLE : 7.9: SOLUTION :-");
printf("\n (a) Slip when injected EMF is opposite to the E2, s = %.1f \n",s1)
printf("\n (b) Slip when injected EMF is phase with E2, s = %.1f \n",s2)
```

Scilab code Exa 7.10 To find synchronous linear speed per phase current and pf

```
// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO
   GENERALIZED MACHINE THEORY

// EXAMPLE : 7.10

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

L = 1.0;
   // Length in Meter
S = 60;
   // Number of slots
f = 50;
   // Frequency in Hertz
v = 440;
   // Operating Volage of the Motor in Volts
V = 11.5;
   // Running speed of the motor in Meter/second
m = 3;
   // Number of phases
p = 8;
   // Total number of Poles


// CALCULATIONS

```

```
Vs = (2*L*f)/p;                                     // Synchronous linear speed in Meter/second
s = (Vs-V)/Vs;                                      // Linear slip
Vph = v/sqrt(3);                                    // Phase Voltage in Volts
Z1 = 6.0 + %i*5;                                    // Impedance in Ohms refer figure and page no. 526
Z2 = ((100*%i)*(5*%i+8.2/s))/(100*%i+5*%i+8.2/s);   // Impedance in Ohms refer figure and page no. 526
Z = Z1 + Z2;                                        // Total Impedance in Ohms
I = Vph/Z;                                          // Per phase Current when Machine is running at 11.5 m/s in Amphere
pf = cosd(atand(imag(I),real(I)));                  // Power factor lagging


// DISPLAY RESULTS

disp("EXAMPLE : 7.10 : SOLUTION :-") ;
printf("\n (a) Synchronous linear speed , Vs = %.1f m/s \n ",Vs);
printf("\n (b) Per phase current when Machine is running at 11.5 m/s , I = %.2f < %.2f A \n",abs(I),atand(imag(I),real(I)))
```

Scilab code Exa 7.11 To find input pulse rate in pulses per second

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO
   GENERALIZED MACHINE THEORY

// EXAMPLE : 7.11

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

s = 9;
   // Degree per step of the stepper motor
N = 200;
   // Rotation Speed of the Stepper motor in RPM


// CALCULATIONS

spr = 360/s;
   // Steps Per Revolution (360 is full revolution)
pps = (N*spr)/60;                                    // Input
   pulse rate in pulses per second


// DISPLAY RESULTS

disp("EXAMPLE : 7.11: SOLUTION :-");
printf("\n (a) Input pulse rate is %.2f pulses per
```

```
second \n",pps)
```

Scilab code Exa 7.13 To find torque and state whether force is positive or negative

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO
    GENERALIZED MACHINE THEORY

// EXAMPLE : 7.13

clear ; clc ; close ; // Clear the work space and
    console


// GIVEN DATA

L1 = 1.1;                                                //
    Inductance in Henry
L2 = 1.07;                                               //
    Inductance in Henry
dtheta = 1;                                              //
    Rotor rotation in Mechanical degree
r = 0.10;                                                //
    Radius of the rotor in Meter
I = 20;
```

```
// Coil Current in Amphere


// CALCULATIONS

dl = L1-L2;                                  // Change in Inductance of one of its stator coils in Henry
F = (I^2*dl)/(2*dtheta);                     // Force on the single rotor pole in Newton
T = F*r;                                     // Instantaneous Torque in Newton-meter


// DISPLAY RESULTS

disp("EXAMPLE : 7.13: SOLUTION :-");
printf("\n (a) Instantaneous Torque, T = %.1f N-m \n \n",T)
printf("The force is a motoring force since inductance of the coil is rising")
```

Scilab code Exa 7.14 To find positive sequence negative sequence zero sequence voltages

```

// ELECTRICAL MACHINES
// R.K.Srivastava
// First Impression 2011
// CENGAGE LEARNING INDIA PVT. LTD

```

```
// CHAPTER : 7 : SPECIAL MOTORS AND INTRODUCTION TO
   GENERALIZED MACHINE THEORY

// EXAMPLE : 7.14

clear ; clc ; close ; // Clear the work space and
   console


// GIVEN DATA

Va = 220 * exp( %i * 0 * %pi/180);
                                        // Three phase in
   Volts
Vb = 230 * exp( %i * (-115) * %pi/180);
                                   //  Three phase in
   Volts
Vc = 250 * exp( %i * (-245) * %pi/180);
                                   //  Three phase in
   Volts


// CALCUALTIONS
// We know that operator :-

alpha = 1 * exp( %i * 120 * %pi/180);
alpha2 = 1 * exp( %i * (-120) * %pi/180);
Va0 = (Va+Vb+Vc)/3
                                                        //
   Zero sequence Voltage in Volts
Va1 = (Va+alpha*Vb+alpha2*Vc)/3
                                         // Positive
   sequence Voltage in Volts
Va2 = (Va+alpha2*Vb+alpha*Vc)/3
                                         // Negative
   sequence Voltage in Volts


```

```
// DISPLAY RESULTS

disp("EXAMPLE : 7.14 : SOLUTION :-") ;
printf("\n (a) Zero sequence Voltage, Va0 = %.2f < %
    .2f V \n",abs(Va0),atand(imag(Va0),real(Va0)))
printf("\n (b) Positive sequence Voltage, Va1 = %.3f
     < %.2f V \n",abs(Va1),atand(imag(Va1),real(Va1))
    )
printf("\n (c) Negative sequence Voltage, Va1 = %.2f
     < %.1f V \n",abs(Va2),atand(imag(Va2),real(Va2))
    )
```