Contents

ENM 056 : Machine Design Assignment

V 1.1 Responsible: Group_8 Contact: smitd@student.chalmers.se, marjog@student.chalmers.se and rahulp@student.chalmers.se

```
% ************************************************************************
% This is a template file to help the studnets start up machine design
% assignment in ENM 056. Please fill in the relevant sections one by one.
% In order to run the code section by section, use "Run Section" option in
% MATLAB or press Ctrl + Enter.
% Note: Please be careful when changing variable names in the template
% file. Also, the students may still need to write some sections by
% themselves.
% ************************************************************************

clear
close all
clc
```

Parameters of the reference machine

Please fill the parameters of the machine below in SI units

```
mm = 1e-3; % mm to SI unit
OD_stator = 176 * mm; % Outer diameter of stator
ID_stator = 124 * mm; % Inner diameter of stator
OD_rotor = 122 * mm ; % Outer diameter of rotor
ID_rotor = 60 * mm ; % Inner diameter of rotor
L_stack = 100 * mm  ; % Stack length
Hs0 = 0.5 * mm     ; % Slot opening height
Hs1 =  0.5 * mm     ; % Slot wedge height
Hs2 = 14 * mm      ; % Slot body height
w_tooth = 4.4 * mm  ; % Tooth width
Rs =   0.5 * mm     ; % Slot bottom radius fillet
Bs0 =  2 * mm     ; % Slot opening
N_pole =  8  ; % Number of poles
```

```
N_slot =  48  ; % Number of slots
w_layer = 2  ; % Number of winding layers
k_p =   5    ; % Coil pitch
D_wire = 1.02 * mm   ; % Wire diameter
f_Cu_max = 0.45 ; % Maximum Cu-fill factor
V_DC =  400    ; % DC link voltage [V]
mu_0 = 4 * pi * 1e-7; % Magnetic permeability of vacuum [H/m]
Omega_base = 4000;    % Base speed [rpm]
Omega__mech_max =12000 ; % Maximum speed [rpm]
N_parallel =  4      ; % Number of parallel branch
N_strand =   1    ; % Number of strands
rho_Cu = 1.72e-8; % Resistivity of Cu [ohm/m]
w_PM = 20 * mm; % width of the permanent magnet can vary from 10 - 30 mm.
t_PM = 5.5 * mm; % the thickness of the Permanent magnet can vary from 3 - 8 mm.
```

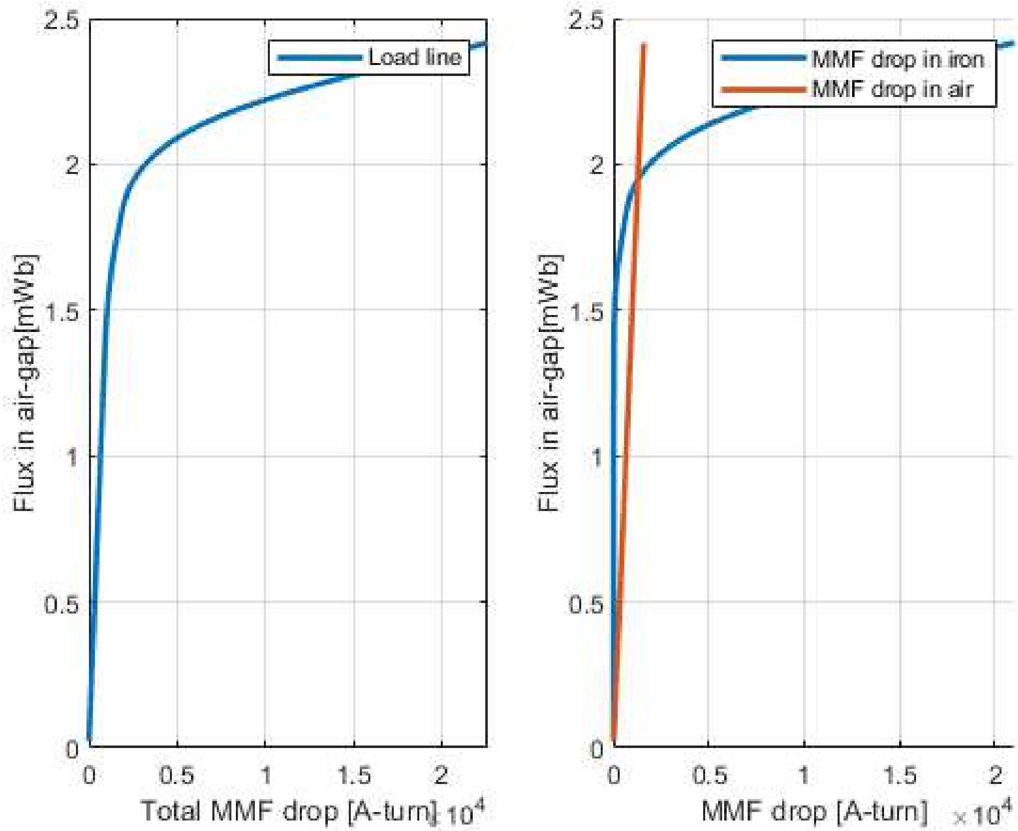
Load line

```
% Start by assuming an air-gap flux density [T]
B_gap = 0.01: 0.001: 1;

% The effective air-gap cross section orthogonal to flux crossing the [sq.m]
% air-gap
A_gap = (pi/16)*(( ID_stator + OD_rotor)/2) *L_stack ;

% Flux in the air-gap [Wb]
Phi_gap = A_gap * B_gap;

% Cross-section of tooth perpendicular to flux [sq.m]
A_tooth = 2.5 * w_tooth * L_stack ;

t_yoke = [((OD_stator - ID_stator)/2 )- (Hs0 + Hs1 + Hs2 + Rs )] ; % yoke thickness [m]

% Cross-section of yoke perpendicul+ar to flux [sq.m]
A_yoke =  L_stack * t_yoke ;

% Rotor Cross-sectional area perpendicular to the flux[sq.m]
A_rotor = w_PM * L_stack;

% Output values
disp('Cross-section of different parts')
fprintf('Air-gap cross-section = % .2f [mm^2] \n',A_gap * 1e6)
fprintf('Stator tooth cross-section = % .2f [mm^2] \n',A_tooth * 1e6)
fprintf('Stator yoke cross-section = % .2f [mm^2] \n \n',A_yoke * 1e6)

% Flux densities in different parts of the machine [T]
B_tooth = Phi_gap /A_tooth; % Flux density in stator tooth
B_yoke = Phi_gap /A_yoke ; % Flux density in stator yoke
B_rotor = Phi_gap /A_rotor ;

index = B_gap == 0.7;

% Output corresponding flux densities for air-gap flux density of 0.7 [T] in
% air-gap
fprintf('Flux densities in the different part of the machine when air-gap flux density is % .
2f [T]\n', B_gap(index))
fprintf('Stator tooth flux density = % .2f [T] \n',B_tooth(index))
```

```
fprintf('Stator yoke flux density = % .2f [T] \n',B_yoke(index))
fprintf('Rotor yoke flux density = % .2f [T] \n \n',B_rotor(index))

% Import the B-H Curve of the M250-35A Steel from a TAB file
BH_data = importdata('SURA M250-35A - BH Curve @ 50 Hz .tab'); % import data
H_data = BH_data(:,1); % copy (Row All , Column One) as H
B_data = BH_data(:,2); % copy (Row All , Column Two) as B

% Calculated magnetic field intensity [A/m]
% Interpolation
method   = 'spline'; % 'linear' or 'spline' can be selected as interpolation method
H_rotor = interp1(B_data,H_data,B_rotor,method); % Interpolat rotor flux density, B_rotor to
calculate corresponding H_rotor
H_stator_tooth =interp1(B_data,H_data,B_tooth,method) ; % Interpolat tooth flux density, B_to
oth to calculate corresponding H_tooth
H_stator_yoke =interp1(B_data,H_data,B_yoke,method) ; % Interpolat tooth flux density, B_yoke
 to calculate corresponding H_yoke
H_gap = B_gap/mu_0 ; % Calculate magnetic field intensity in the air-gap. Note: Air-gap does
not contain iron.

% Output magnetic field intensities
fprintf('Magnetic field intensities in the different part of the machine when air-gap flux de
nsity is % .2f [T] \n', B_gap(index))
fprintf('Air-gap Magnetic field intensity = % .2f [A/m] \n',H_gap(index))
fprintf('Stator tooth Magnetic field intensity = % .2f [A/m] \n',H_stator_tooth(index))
fprintf('Stator yoke Magnetic field intensity = % .2f [A/m] \n',H_stator_yoke(index))
fprintf('Rotor yoke Magnetic field intensity = % .2f [A/m] \n \n',H_rotor(index))

% Length of flux path [m]
l_rotor =[(pi/2)* (4*pi*OD_rotor/48) - (2*t_PM) ]; % Length of flux path in rotor yoke
dsy_1 = ID_stator + 2*(Hs0 + Hs1 + Hs2 + Rs);
dsy_2 = OD_stator;
l_stator_yoke = [(pi*(dsy_1 + dsy_2)/24) + (1/2)*(dsy_2 - dsy_1)]; % Length of flux path in s
tator yoke
l_stator_tooth = (Hs0 + Hs1 + Hs2 + Rs); % Length of flux path in stator tooth
l_gap = ( ID_stator - OD_rotor ) / 2; % Length of flux path in air-gap

% Output values
disp('Length of flux path in different parts of the machine')
fprintf('Length of flux path in rotor = % .2f [mm] \n',l_rotor * 1e3)
fprintf('Length of flux path in stator yoke = % .2f [mm] \n',l_stator_yoke * 1e3)
fprintf('Length of flux path in stator tooth = % .2f [mm] \n',l_stator_tooth * 1e3)
fprintf('Length of flux path in air-gap = % .2f [mm] \n \n',l_gap * 1e3)

% MMF drop in different flux path [Aturn]
MMF_rotor = H_rotor * l_rotor; % MMF drop in rotor yoke
MMF_stator_yoke =H_stator_yoke * l_stator_yoke ; % MMF drop in stator yoke
MMF_stator_tooth = H_stator_tooth * l_stator_tooth; % MMF drop in stator tooth
MMF_gap = H_gap * l_gap ; % MMF drop in air-gap

% Output values
fprintf('MMF drop in different parts of the machine when air-gap flux density is % .2f [T] \n
', B_gap(index))
fprintf('MMF drop in rotor yoke = % .2f [A-turn] \n',MMF_rotor(index))
fprintf('MMF drop in stator tooth = % .2f [A-turn] \n',MMF_stator_tooth(index))
fprintf('MMF drop in stator yoke = % .2f [A-turn] \n',MMF_stator_yoke(index))
fprintf('MMF drop in air-gap = % .2f [A-turn] \n\n',MMF_gap(index))
```

```
% Total MMF drop
MMF_total =MMF_rotor + MMF_stator_yoke + 2 * MMF_stator_tooth + 2 * MMF_gap;

MMF_iron = MMF_rotor + MMF_stator_yoke + 2 * MMF_stator_tooth; % Total MMF drop in the iron p
arts
MMF_air = 2* MMF_gap; % Total MMF drop in air-gap

% Plot load line
figure(1)                % create Figure 1
clf                      % clear figure
subplot(1,2,1)
plot(MMF_total,Phi_gap*1e3, 'LineWidth', 2)
xlabel('Total MMF drop [A-turn]')
ylabel('Flux in air-gap[mWb]')
legend('Load line')
grid on

subplot(1,2,2)
hold on
plot(MMF_iron,Phi_gap*1e3, 'LineWidth', 2)
plot(MMF_air,Phi_gap*1e3, 'LineWidth', 2)
xlabel('MMF drop [A-turn]')
ylabel('Flux in air-gap[mWb]')
legend('MMF drop in iron','MMF drop in air')
grid on
```

```
Cross-section of different parts
Air-gap cross-section =  2415.10 [mm^2]
Stator tooth cross-section =  1100.00 [mm^2]
Stator yoke cross-section =  1050.00 [mm^2]

Flux densities in the different part of the machine when air-gap flux density is  0.70 [T]
Stator tooth flux density =  1.54 [T]
Stator yoke flux density =  1.61 [T]
Rotor yoke flux density =  0.85 [T]

Magnetic field intensities in the different part of the machine when air-gap flux density is
 0.70 [T]
Air-gap Magnetic field intensity =  557042.30 [A/m]
Stator tooth Magnetic field intensity =  2475.80 [A/m]
Stator yoke Magnetic field intensity =  4338.37 [A/m]
Rotor yoke Magnetic field intensity =  82.74 [A/m]

Length of flux path in different parts of the machine
Length of flux path in rotor =  39.17 [mm]
Length of flux path in stator yoke =  53.83 [mm]
Length of flux path in stator tooth =  15.50 [mm]
Length of flux path in air-gap =  1.00 [mm]

MMF drop in different parts of the machine when air-gap flux density is  0.70 [T]
MMF drop in rotor yoke =  3.24 [A-turn]
MMF drop in stator tooth =  38.37 [A-turn]
MMF drop in stator yoke =  233.52 [A-turn]
MMF drop in air-gap =  557.04 [A-turn]
```



Magnet dimension

```
t_mag = 5.5 * mm ; % Thickness of magnet segment
w_mag =  20* mm; % Width of magnet segment
A_mag = L_stack *  w_mag; % area of the magnet
% Magnet data
B_mag = [0 0.5912 1.1824];
H_mag = [-902285 -451142 0];

MMF_mag = 2 * H_mag * t_mag; % MMF produced by magnet
Phi_mag = B_mag * A_mag ; % Flux produced by the magnet

% Magnet characteristic
figure(2)                % create Figure 1
clf                      % clear figure
plot(MMF_mag,Phi_mag * 1e3, 'LineWidth', 2)
xlim([min(MMF_mag) 0])
xlabel('MMF produced by magnet [A-turn]')
ylabel('Flux due to magnet [mWb]')
legend('Demagnetization characteristic')
grid on
```

Finding the intersection

To avoid extrapolation of data, we can limit the x-axis of the load line to maximum MMF that can be produced by magnet

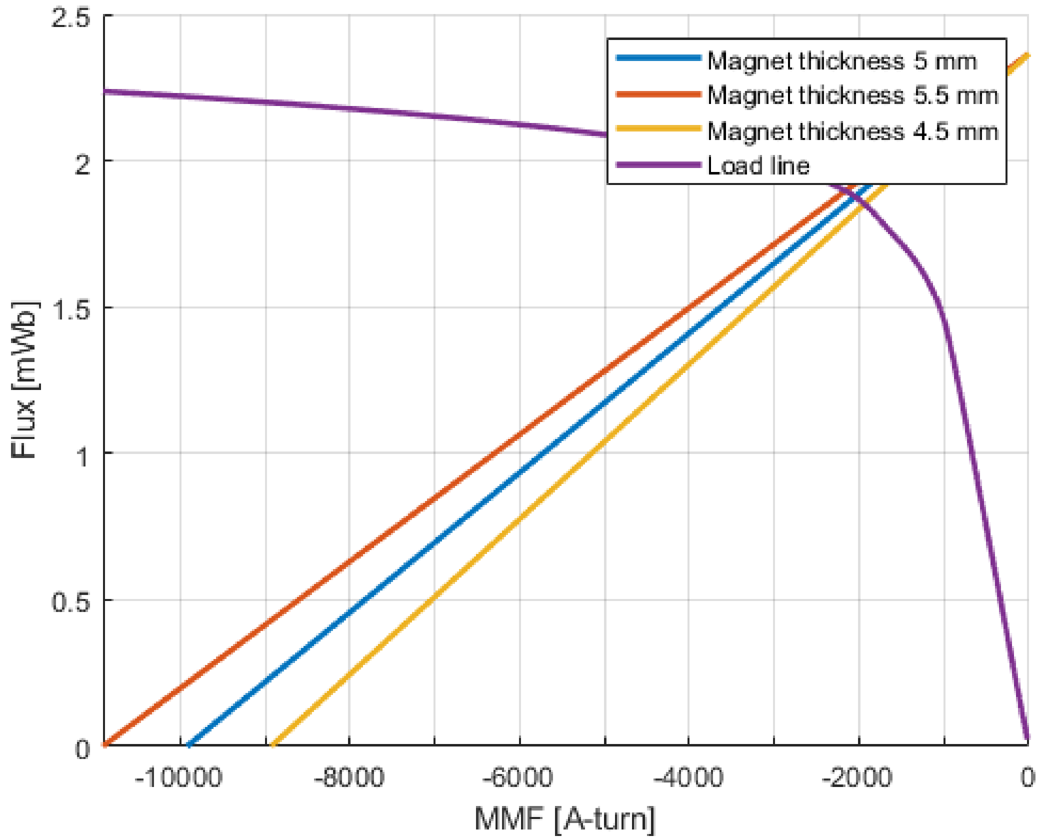
```
index = MMF_total < max(abs(MMF_mag));
MMF_total_con = MMF_total(index);
Phi_gap_con = Phi_gap(index);


% Interpolate magnet characteristic corresponding to total MMF drop to have
% same x-axis. Don't forget the negative sine to move it to second quadrant
Phi_mag_interp = interp1( abs(MMF_mag), Phi_mag , MMF_total_con , method);
[value, index] = min(abs(Phi_mag_interp - Phi_gap_con));

Phi_gap_no_load = Phi_gap_con(index);
B_gap_no_load = Phi_gap_no_load / A_gap;
B_tooth_no_load = Phi_gap_no_load / A_tooth; % Flux density in stator tooth
B_yoke_no_load = Phi_gap_no_load / A_yoke; % Flux density in stator yoke

fprintf('Magnet thickness = % .2f [mm] \n',t_mag * 1e3)
fprintf('Magnet width = % .2f [mm] \n',w_mag * 1e3)
fprintf('No load flux in the air-gap = % .5f [Wb] \n',Phi_gap_no_load)
fprintf('No load flux density in the air-gap = % .2f [T] \n',B_gap_no_load)
fprintf('No load flux density in stator tooth = % .2f [T] \n',B_tooth_no_load)
fprintf('No load flux density in stator yoke = % .2f [T] \n\n',B_yoke_no_load)
```

Perform sensitivity analysis

analysis 0: Do nothing analysis 1: Change in magnet thickness

```
analysis = 1;

switch analysis

    case 0
    % DO NOTHING

    case 1       % Change in magnet thickness

        t_mag_1 = t_mag * 1.1; % 10% increase in magnet thickness
        t_mag_2 = t_mag * 0.9; % 10% decrease in magnet thickness

        MMF_mag_1 = 2 * H_mag * t_mag_1;
        MMF_mag_2 = 2 * H_mag * t_mag_2;

        % Magnet characteristic
        figure(4)               % create Figure 1
        clf                     % clear figure
        hold on
        plot(MMF_mag,Phi_mag * 1e3, 'LineWidth', 2)
        plot(MMF_mag_1,Phi_mag * 1e3, 'LineWidth', 2)
        plot(MMF_mag_2,Phi_mag * 1e3, 'LineWidth', 2)
        plot(-MMF_total,Phi_gap * 1e3, 'LineWidth', 2)
        hold off
        xlim([min(MMF_mag_1) 0])
        xlabel('MMF [A-turn]')
        ylabel('Flux [mWb]')
        legend('Magnet thickness 5 mm' , 'Magnet thickness 5.5 mm','Magnet thickness 4.5 mm',
 'Load line')
```

```
        grid on

end
```



Slot area [sq.m]

```
% Tooth base [m]
Bs1=pi*(ID_stator+(2*(Hs0+Hs1)))/N_slot-w_tooth;
Bs2=pi*(ID_stator+(2*(Hs0+Hs1+Hs2)))/N_slot-w_tooth;


% Tooth and slot area [sq.m]
A_tooth_slot = (Bs2-2*Rs)*Rs+pi/(2*Rs^2);

% Slot area [sq.m]
A_slot = (Bs1+Bs2)/2*Hs2+(Bs2-2*Rs)*(Rs+pi)/2*Rs^2;

% Tooth area [sq.m]
A_tooth =(Hs2+Rs^2)*(w_tooth);
```

Stator winding design

```
q = 2 ; % Number of slots per pole per phase
r=2;
% Omega_mech = (800 * pi) /6;
omega_elec=((2*pi)/60)*Omega_base;
alpha = 360/(2 * q * 3);
```

```
k_w1 = 0.933; % Electrical angular frequency [rad/s] from Calculations
Omega_max=(2*pi*Omega__mech_max)/60;

f_elec = omega_elec/(2*pi) ; % Electrical frequency [Hz]

% % Total number of turns per phase
 N_ph_total =  (N_pole/2) * q * r * N_parallel/(sqrt(2)*pi*omega_elec*Phi_gap_no_load*k_w1*q*
r*N_pole/2)*N_slot ;
%
% % Number of series branch
N_series =N_pole/N_parallel;

% % Number of coils in series per phase
N_coil_ph = q*r/2*N_series;
% N_turn_ph = N_turn_coil*N_coil_ph;

%Number of turns per coil
N_turn_coil = round (N_ph_total / N_coil_ph);
N_ph_total = N_coil_ph * N_turn_coil;


N_turn = floor( 230.95 * N_parallel/(omega_elec* Phi_gap_no_load* q * k_w1 * r * 16)) ;
N_turn_ph = ((N_pole/2)*q*r/N_parallel)*N_turn; % Total turns of the machine

% % Number of conductors per slot
N_cond_slot =N_turn_coil*r*N_strand;
%
% % Cu area
A_Cu =(D_wire/2)^2*pi*N_strand*N_turn_coil*r ;
%
% % Fill factor
f_Cu = A_Cu / A_slot;
%
% % Maximum induced voltage at no-load
E_no_load =N_turn_coil*sqrt(2)*pi*f_elec*Phi_gap_no_load*k_w1*q*r*(N_pole/2)/N_parallel ;

% % Maximum line to line induced voltage at no -load
E_LL_no_load =E_no_load*sqrt(3);

% % Maximum line to line induced voltage at max speed
E_LL_max_speed =(sqrt(3)*(N_pole/2)*q*Omega_max*Phi_gap_no_load*N_turn*k_w1*r*(N_pole/2))/N_p
arallel;

 fprintf('N_turn = % .2f \n',N_turn)
 fprintf('Total number of turns per phase = % .0f \n',N_ph_total )
 fprintf('Total number of coils in series per phase = % .0f \n',N_coil_ph )
 fprintf('Total number of turns per coil = % .0f \n',N_turn_coil )
 fprintf('Total number of conductors per slot = % .0f \n',N_cond_slot )
 fprintf('Slot area = % .1f [mm^2] \n',A_slot * 1e6)
 fprintf('Cu area = % .1f [mm^2] \n',A_Cu * 1e6)
 fprintf('Fill factor = % .2f \n\n',f_Cu)

 fprintf('Maximum per phase induced voltage at no-load = % .1f [V] \n',E_no_load)
 fprintf('Maximum line to line induced voltage at no-load = % .1f [V] \n',E_LL_no_load)
 fprintf('Maximum line to line induced voltage at %.0f [rpm] = % .1f [V] \n',Omega_max, E_LL_
max_speed)
 fprintf('Ratio of maximum line to line induced and DC link voltage at max speed of %.0f [rpm
```

```
] = % .1f \n\n',Omega_max, E_LL_max_speed / V_DC)
```

```
N_turn =  19.00
Total number of turns per phase =  60
Total number of coils in series per phase =  4
Total number of turns per coil =  15
Total number of conductors per slot =  30
Slot area =  66.7 [mm^2]
Cu area =  24.5 [mm^2]
Fill factor =  0.37

Maximum per phase induced voltage at no-load =  31.1 [V]
Maximum line to line induced voltage at no-load =  53.9 [V]
Maximum line to line induced voltage at 1257 [rpm] =  1158.5 [V]
Ratio of maximum line to line induced and DC link voltage at max speed of 1257 [rpm] =  2.9
```

Resistance and inductance calculation

```
Bs1 = pi* ((ID_stator + 2*(Hs0 + Hs1))/48) - w_tooth;
Bs2 =   pi* ((ID_stator + 2*(Hs0 + Hs1 + Hs2))/48) - w_tooth;
l_coil_theoretical = 2*L_stack + (pi* ((Bs1 + Bs2)/2 + w_tooth)*5);
r = 2; % Number of winding layers
l_coil_calc =  2*L_stack + 2* 0.8* L_stack;

R_phase = rho_Cu* (( l_coil_calc * N_turn * r * (N_pole/2) * q))/((pi/4)* ((D_wire)^2) * N_st
rand * (N_parallel)^2);

Re_mag = abs(MMF_mag(2))/Phi_mag(2);
Re_gap = MMF_gap(index)/Phi_gap_no_load;
Re_stator_tooth = MMF_stator_tooth/Phi_gap_no_load;
Re_stator_yoke = MMF_stator_yoke/Phi_gap_no_load;
Re_rotor_yoke = MMF_rotor/Phi_gap_no_load;


Re_d = ((MMF_total(index))/Phi_gap_no_load) + Re_mag ;
Re_q = ((MMF_total(index))/Phi_gap_no_load);
alpha = 360/(2 * q * 3);
N = N_turn* k_w1 * q * r ; % Nd = Nq = N = N_turn* Kw * q* r

L_d = (N^2 * (N_pole/2))/(Re_d * N_parallel);
L_q = (N^2 * (N_pole/2))/(Re_q * N_parallel);

saliency = L_q / L_d;

fprintf('Resistance per phase = % .1f [mOhm] \n',R_phase*1e3)
fprintf('Reluctance of d-flux path = % .1f [H^-1] \n',Re_d)
fprintf('Reluctance of q-flux path = % .1f [H^-1] \n',Re_q)
fprintf('D-axis inductance = % .1f [mH] \n',L_d*1e3)
fprintf('Q-axis inductance = % .1f [mH] \n',L_q*1e3)
fprintf('Saliency ratio = % .1f \n',saliency)
fprintf('l_coil_theoretical = % .1f \n',l_coil_theoretical);
fprintf('l_coil_calc = % .1f \n',l_coil_calc);
```

```
Resistance per phase =  144.0 [mOhm]
Reluctance of d-flux path =  5288871.1 [H^-1]
Reluctance of q-flux path =  1091846.4 [H^-1]
D-axis inductance =  1.0 [mH]
Q-axis inductance =  4.6 [mH]
Saliency ratio =  4.8
l_coil_theoretical =  0.3
l_coil_calc =  0.4
```

Load Calculations

```
I = 100; % Current Amplitude = 100 Amps
current_angle = 90; % current angle ( theta_i)
K_h = 156.201; % Hysteresis co-efficient
K_c = 0.0204184; % Eddy current co-efficient
rotor_speed = 3000; % 3000 in rpm Mech

Omega = rotor_speed * 2 * pi * (N_pole/2) * (1/60); %[rad/s]

Omega_mech = Omega /(N_pole/2);

% d and q axis current calculation
I_d = I * cosd(current_angle); % Direct axis current
I_q = I * sind(current_angle); % Quadrature Axis Current

%Volume of different sections
Volume_tooth = A_tooth * L_stack * N_slot;

Volume_statoryoke = (OD_stator - ID_stator)^2 / 4 * pi * L_stack - A_tooth * L_stack * N_slot
;

Volume_rotor = (OD_rotor - ID_rotor)^2 / 4 * pi * L_stack - w_mag * t_mag * L_stack * 16;


% d and q Flux Linkages at Load Condition
Flux_Linkage_PM = (Phi_gap_no_load  * N_turn * q * k_w1 * r * (N_pole/2))/(N_parallel);
psy_d = (L_d * I_d) + Flux_Linkage_PM;
psy_q = L_q * I_q;


Tem = 3/2 * (N_pole/2) * psy_d * I_q;
U_d = R_phase * I_d - Omega * psy_q;
U_q = R_phase * I_q + Omega * psy_d;
U_s = sqrt(U_d^2 + U_q^2);


P_cu = 3 * R_phase * (I / sqrt(2))^2;
P_iron_tooth = (K_h * B_tooth_no_load^2 * Omega / (2*pi) + K_c * (B_tooth_no_load * Omega / (
2*pi))^2) * Volume_tooth;
% DC flux on the rotor yoke
P_iron_statoryoke = (K_h * B_yoke_no_load^2 * f_elec + K_c * B_yoke_no_load^2 * f_elec^2 )* V
olume_statoryoke;

P_shaft = Tem * Omega_mech; % P_friction
P_input = P_shaft  + P_iron_tooth + P_iron_statoryoke + P_cu;
```

```
S = (3/2)*U_s*I;
efficiency = (P_shaft/P_input)* 100;
power_factor=P_input/S;

fprintf('P_shaft = % .1f \n',P_shaft);
fprintf('P_input = % .1f \n',P_input);
fprintf('Apparent Power (S) = % .1f \n',S );
fprintf('efficiency = % .1f \n',efficiency);
fprintf('power_factor = % .1f \n',power_factor);
```

```
P_shaft =  25081.4
P_input =  27265.9
Apparent Power (S) =  90976.2
efficiency =  92.0
power_factor =  0.3
```