```
%*********************************************************************
%*********************************************************************
%
%                    MATLAB-Script zur Ermittlung
%   der Drehzahl einer VEB KMR 160 M6 Drehstromasynchronmaschine
%                    zur Verwendung als Generator im
%           Kleinwasserkraftwerk Niedermuehle / Nussbaumer
%
%*********************************************************************
%*********************************************************************

% -------------------------------------------------------------------
% >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
% Eingabe und Definition der Konstanten und Arrays
% >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
% -------------------------------------------------------------------

% -------------------------------------------------------------------
% Dateneingabe Betriebspunkt
% -------------------------------------------------------------------

    % Turbinenleistung um die Riemenverluste vermindert in kW
        P_turbine = -15;

% -------------------------------------------------------------------
% Eingabe der Konstanten
% -------------------------------------------------------------------

    % Effektive Betriebsspannung in Volt
        U = 400;
        u = complex(U,0);

    % Synchrondrehzahl in Umdrehungen pro Minute
        ns = 1000;

    % Nenndrehzahl in Umdrehungen pro Minute
        nn = 965;

    % rein mechanischer Wirkungsgrad bei Nenndrehzahl (Annahme)
        eta_mech = 0.96;

    % Betriebstemperatur der Wicklungen in °C
        T = 120;

    % Temperaturkoeffizienten von Kupfer in 1/K bzw. 1/K^2 bei 20°C
        alpha_20 = 3.93*10^(-3);
        beta_20  = 0.60*10^(-6);

    % Ersatzwiderstaende des Schaltbildes bei 120°C Betriebstemperatur
        R1_120 = 1.2920;
        R2_120 = 1.3780;
        Rfe = 1119;
        X1 = 2.782;
```

```
        X2 = 3.495;
        Xh = 88.061;

        R1 = R1_120*(1 + alpha_20*(T-120) + beta_20*(T-120)^2);
        R2 = R2_120*(1 + alpha_20*(T-120) + beta_20*(T-120)^2);

        Z1 = complex(0,X1);
        Z2 = complex(0,X2);
        Zh = complex(0,Xh);

    % Drehzahlarray
        N = 930:0.05:1100;
        dim = length(N);

    % Wirkstromarray
        Ir = zeros(dim);

    % Blindstromarray
        Ii = zeros(dim);

    % Leistungsfaktorarray
        COSPHI = zeros(dim);

    % Wirkleistungsarray
        P = zeros(dim);

    % mechanisches Leistungsarray
        P_mech = zeros(dim);

    % Wirkungsgradarray
        ETA = zeros(dim);

    % Drehmomentenarray
        T = zeros(dim);

% ---------------------------------------------------------------------
% >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
% Berechnungsschritte
% >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
% ---------------------------------------------------------------------

% ---------------------------------------------------------------------
% Stromortskurve
% ---------------------------------------------------------------------

    % Berechnung:
    % Z> = R1 + Z1 + 1/(1/(Z2+(R2/s)) + 1/Zh + 1/Rfe)
    % I> = U>/Z> ...ohmsches Gesetz in Vektorschreibweise

for i=1:dim,
    n = N(i);
    s = (ns-n)/ns;
    Ir(i) = real(u/(R1 + Z1 + 1/(1/(Z2+(R2*(1-s)/s)+R2) + 1/Zh + 1/Rfe)));
```

```
    Ii(i) = -imag(u/(R1 + Z1 + 1/(1/(Z2+(R2*(1-s)/s)+R2) + 1/Zh + 1/Rfe)));
end

% ---------------------------------------------------------------------
% Leerlaufstrom, s = 0 (Real-, Imaginaerteil und Absolutbetrag)
% ---------------------------------------------------------------------

    Ir_l = real(u/(R1 + Z1 + 1/(1/Zh + 1/Rfe)));
    Ii_l = -imag(u/(R1 + Z1 + 1/(1/Zh + 1/Rfe)));

% ---------------------------------------------------------------------
% Kurzschlussstrom, s = 1 (Real-, Imaginaerteil und Absolutbetrag)
% ---------------------------------------------------------------------

    Ir_k = real(u/(R1 + Z1 + 1/(1/(Z2+R2) + 1/Zh + 1/Rfe)));
    Ii_k = -imag(u/(R1 + Z1 + 1/(1/(Z2+R2) + 1/Zh + 1/Rfe)));

% ---------------------------------------------------------------------
% Strom bei s = unendlich (Real-, Imaginaerteil und Absolutbetrag)
% ---------------------------------------------------------------------

    Ir_u = real(u/(R1 + Z1 + 1/(1/(Z2) + 1/Zh + 1/Rfe)));
    Ii_u = -imag(u/(R1 + Z1 + 1/(1/(Z2) + 1/Zh + 1/Rfe)));

% ---------------------------------------------------------------------
% Drehmomentengerade und Leistungsgerade
% ---------------------------------------------------------------------

    % Steigung und Offset der Leistungsgeraden:
    k_p = (Ir_k - Ir_l)/(Ii_k - Ii_l);
    d_p = Ir_l - k_p*Ii_l;

    % Steigung und Offset der Drehmomentengeraden:
    k_t = (Ir_u - Ir_l)/(Ii_u - Ii_l);
    d_t = Ir_l - k_t*Ii_l;

% ---------------------------------------------------------------------
% Kennlinien mit Verlustbetrachtung
% ---------------------------------------------------------------------

    pos = find(N==nn);
    T_v = (1-eta_mech)*3*U*(Ir(pos) - k_p*Ii(pos) - d_p)/((nn*pi)/30);

for i=1:dim,
    COSPHI(i) = cos(atan(Ii(i)/Ir(i)));
    P(i) = 3*U*Ir(i);
    P_mech(i) = 3*U*(Ir(i) - k_p*Ii(i) - d_p) - T_v*(N(i)*pi/30);
    ETA(i) = P_mech(i)/P(i);
    T(i) = 30*P_mech(i)/(N(i)*pi);
end

% ---------------------------------------------------------------------
% >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

```
% Plots
% >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
% ---------------------------------------------------------------------

% Nachfolgende Punkte beschreiben die Ausgabefenster

% ---------------------------------------------------------------------
% Ausgabefenster 1: Die Stromortskurve
% ---------------------------------------------------------------------

figure(1)

for i=1:dim
    plot(Ii(i),Ir(i),'color','k')                   % Stromortskurve
    %axis([0,20,-10,10])
    grid on
    hold on
end

for i=1:dim
    plot(Ii(i),k_t*Ii(i) + d_t,'color','b')         % Drehmomentengerade
    hold on
end

for i=1:dim
    plot(Ii(i),k_p*Ii(i) + d_p,'color','r')         % Leistungsgerade
    hold on
end

title('Stromortskurve','Fontsize',12);
xlabel('Blindstrom');
ylabel('Wirkstrom');

% ---------------------------------------------------------------------
% Ausgabefenster 2: Cos(phi) über Drehzahl
% ---------------------------------------------------------------------

figure(2)

for i=1:dim,
    plot(N(i),COSPHI(i),'color','b')
    grid on
    hold on
end
title('Leistungsfaktor','Fontsize',12);
xlabel('Drehzahl in U/min');
ylabel('cos(phi)');

% ---------------------------------------------------------------------
% Ausgabefenster 3: Drehmoment über Drehzahl
% ---------------------------------------------------------------------

figure (3)
```

```
for i=1:dim,
    plot(N(i),T(i),'color','b')
    grid on
    hold on
end
title('Drehmoment','Fontsize',12);
xlabel('Drehzahl in U/min');
ylabel('Drehmoment in Nm');

% ---------------------------------------------------------------------
% Ausgabefenster 4: Mechanische Leistung über Drehzahl
% ---------------------------------------------------------------------

figure (4)

for i=1:dim,
    plot(N(i),P_mech(i)/1000,'color','b')       % /1000 wegen Angabe in kW
    hold on
end
title('mechanische Leistung','Fontsize',12);
grid on
xlabel('Drehzahl in U/min');
ylabel('mechanische Leistung in kW');

% ---------------------------------------------------------------------
% Ausgabefenster 5: Zusammenfassung der Daten für die vorgeg. Leistung
% ---------------------------------------------------------------------

figure (5)

title('Daten am Auslegungspunkt','Fontsize',12);

for i=1:dim,
    if ((P_mech(i)-(-P_turbine*1000)) >= 0);
    betr_pkt = i;
    else
        break;
    end
end

s1 = -P_mech(betr_pkt)/1000;
string1=[texlabel('Aufgenommene mechanische Leistung: ') num2str(s1) ' kW'];
text(0.1,0.9,string1)

s2 = -P(betr_pkt)/1000;
string2=[texlabel('Abgegebene elektrische Leistung: ') num2str(s2) ' kW'];
text(0.1,0.8,string2)

s3 = -(P_mech(betr_pkt)/1000 - P(betr_pkt)/1000);
string3=[texlabel('Abgegebene Wärmeleistung: ') num2str(s3) ' kW'];
text(0.1,0.7,string3)

```

```
if P(betr_pkt)>=0,
    s4 = abs(P_mech(betr_pkt)/(3*U*Ir(betr_pkt)));
else
    s4 = abs(3*U*Ir(betr_pkt)/P_mech(betr_pkt));
end

string4=[texlabel('Wirkungsgrad: ') num2str(s4) ' -'];
text(0.1,0.6,string4)

s5 = COSPHI(betr_pkt);
string5=[texlabel('Leistungsfaktor cos(phi): ') num2str(s5) ' -'];
text(0.1,0.5,string5)

s6 = N(betr_pkt);
string6=[texlabel('Drehzahl der Generatorwelle: ') num2str(s6) ' U/min'];
text(0.1,0.4,string6)

s7 = -T(betr_pkt);
string7=[texlabel('Drehmoment an der Generatorwelle: ') num2str(s7) ' Nm'];
text(0.1,0.3,string7)

s8 = sqrt(3)*sqrt((Ir(betr_pkt))^2+(Ii(betr_pkt))^2);
string8=[texlabel('Gesamtstrom: ') num2str(s8) ' A'];
text(0.1,0.2,string8)

text(0.6,0.1,'Copyright by Leibetseder Manuel 2011', 'Fontsize', 6)

% ---------------------------------------------------------------------
% Ausgabefenster 6: Zusammenfassung tabelliert
% ---------------------------------------------------------------------

figure (6)

axis off;                                % keine Achsen darstellen
set(gca,'XTickLabel',[])                 % keine Skalierung an den Achsen
set(gca,'YTickLabel',[])

title('Zusammenfassung tabelliert','Fontsize',12);

text(0.1,0.95,'P_{mech,tats} [kW]')
text(0.2,0.95,'P_{el} [kW]')
text(0.3,0.95,'P_{verl} [kW]')
text(0.4,0.95,'{\eta}')
text(0.5,0.95,'cos(\phi)')
text(0.6,0.95,'n_{welle} [U/min]')
text(0.7,0.95,'T_{welle} [Nm]')
text(0.8,0.95,'I_{ges} [A]')

for j=0:30

    P_temp = j-20;
    betr_pkt = 0;

```

```
    for i=1:dim,
        if ((P_mech(i)-(P_temp*1000)) >= 0);
        betr_pkt = i;
        else
            break;
        end
    end

s1 = round(P_mech(betr_pkt)/1000);
text(0.1,(j+1)/35,num2str(s1))          % mechanische Leistung (0 Kommastellen)

s2 = round(P(betr_pkt)/10)/100;
text(0.2,(j+1)/35,num2str(s2))          % elektrische Leistung (2 Kommastellen)

s3 = round(-(P_mech(betr_pkt)/10 - P(betr_pkt)/10))/100;
text(0.3,(j+1)/35,num2str(s3))          % Verlustleistung (2 Kommastellen)

if P(betr_pkt)>=0,
    s4 = round(abs(P_mech(betr_pkt)/(3*U*Ir(betr_pkt)))*100)/100;
else
    s4 = round(abs(3*U*Ir(betr_pkt)/(P_mech(betr_pkt)))*100)/100;
end

text(0.4,(j+1)/35,num2str(s4))          % Wirkungsgrad (2 Kommastellen)

s5 = round(COSPHI(betr_pkt)*100)/100;
text(0.5,(j+1)/35,num2str(s5))          % Cos(phi) (2 Kommastellen)

s6 = round(N(betr_pkt)*10)/10;
text(0.6,(j+1)/35,num2str(s6))          % Drehzahl (1 Kommastellen)

s7 = round(T(betr_pkt)*10)/10;
text(0.7,(j+1)/35,num2str(s7))          % Drehmoment (1 Kommastellen)

s8 = round((sqrt(3)*sqrt((Ir(betr_pkt))^2+(Ii(betr_pkt))^2))*10)/10;
text(0.8,(j+1)/35,num2str(s8))          % Gesamtstrom (2 Kommastellen)

end
```