

---

```

function [R_pu,X_pu,B_pu]=e230481_IPEK(text_path,library_path)
fid = fopen(text_path);
while true
    if fgetl(fid) == 'Sbase (MVA)'
        S_base=str2double(fgetl(fid))*1000000;
    end
    if fgetl(fid) == 'Vbase (kV)'
        V_base=str2double(fgetl(fid))*1000;
    end
    if fgetl(fid) == 'Number of circuits'
        N_circuit=str2double(fgetl(fid));
    end
    if fgetl(fid) == 'Number of bundle conductors per phase'
        N_bundle=str2double(fgetl(fid));
    end
    if fgetl(fid) == 'Bundle distance (m)'
        D_bundle=str2double(fgetl(fid));
    end
    if fgetl(fid) == 'Length of the line (km)'
        length=str2double(fgetl(fid))*1000;
    end
    if fgetl(fid) == 'ACSR conductor name'
        conductor_name=fgetl(fid);
    end
    if fgetl(fid) == 'C1 Phase A (centre)'
        x1=str2double(fgetl(fid));
        y1=str2double(fgetl(fid));
    end
    if fgetl(fid) == 'C1 Phase B (centre)'
        x2=str2double(fgetl(fid));
        y2=str2double(fgetl(fid));
    end
    if fgetl(fid) == 'C1 Phase C (centre)'
        x3=str2double(fgetl(fid));
        y3=str2double(fgetl(fid));
    end
    if N_circuit == 2
        if fgetl(fid) == 'C2 Phase A (centre)'
            x4=str2double(fgetl(fid));
            y4=str2double(fgetl(fid));
        end
        if fgetl(fid) == 'C2 Phase B (centre)'
            x5=str2double(fgetl(fid));
            y5=str2double(fgetl(fid));
        end
        if fgetl(fid) == 'C2 Phase C (centre)'
            x6=str2double(fgetl(fid));
            y6=str2double(fgetl(fid));
        end
    end
    if fgetl(fid) == '-999'
        break
    end
end

```

---

---

```

    end
end
fclose(fid);
T = readtable(library_path, 'VariableNamingRule', 'preserve'); %
    converting csv to table T
names=table2array(T(:,1)); % Extracting conductor names from table T
index = find(strcmp(names, conductor_name)); % Finding index where our
    conductor is
outside_diameters=table2array(T(:,5)); % Extracting outside diameter
    values from table T
outside_diameter=outside_diameters(index); % Finding desired outside
    diameter in in
outside_diameter= outside_diameter * 0.0254; % Converting in to m ( 1
    in == 0.0254 m)
RACs=table2array(T(:,7)); % Extracting AC resistance values from table
    T
R_AC=RACs(index); % Finding desired outside diameter in ohm/mil
R_AC= R_AC / 1609.344; % Converting ohm/mil to ohm/m ( 1 mil ==
    1609.344 m)
GMRconductors=table2array(T(:,8)); % Extracting AC resistance values
    from table T
GMR_conductor=GMRconductors(index); % Finding desired GMR in ft
GMR_conductor=GMR_conductor * 0.3048; % Converting ft to m ( 1 ft ==
    0.3048 m)

%Phase-2
%Depending on N_bundle, it is finding out GMR_BUNDLE
switch N_bundle
    case 1
        GMR_bundle=GMR_conductor;
    case 2
        GMR_bundle=sqrt(GMR_conductor*D_bundle);
    case 3
        GMR_bundle=(GMR_conductor*D_bundle*D_bundle)^(1/3);
    case 4
        GMR_bundle=((GMR_conductor*(D_bundle^3)*sqrt(2))^(1/4));
    case 5

GMR_bundle=((GMR_conductor*(D_bundle^4)*((1+sqrt(5))/2)^2)^(1/5));
    case 6
        GMR_bundle=((GMR_conductor*(D_bundle^5)*(2*3))^(1/6));
    case 7
        D_short= (D_bundle^2 + D_bundle^2
-2*D_bundle*D_bundle*cos(128.57*pi/180))^(1/2);
        D_long = (D_short^2 + D_bundle^2
-2*D_short*D_bundle*cos(102.85*pi/180))^(1/2);

GMR_bundle=((GMR_conductor*(D_bundle^2)*(D_short^2)*(D_long^2))^(1/7));
    case 8
        D_short = (D_bundle^2 + D_bundle^2-
2*D_bundle*D_bundle*cos(135*pi/180))^(1/2);
        D_medium = (D_short^2 + D_bundle^2 -
2*D_short*D_bundle*cos(112.5*pi/180))^(1/2);

```

---

---

```

        D_long = (D_medium^2 + D_bundle^2)^(1/2);

        GMR_bundle=(GMR_conductor*(D_bundle^2)*(D_short^2)*(D_medium^2)*D_long)^(1/8);
    end

    %Depending on N_bundle, it is finding out r_eq
    r_eqcond=outside_diameter/2;

    switch N_bundle
        case 1
            r_eq_b4_number_of_circuit=r_eqcond;
        case 2
            r_eq_b4_number_of_circuit=sqrt(r_eqcond*D_bundle);
        case 3
            r_eq_b4_number_of_circuit=(r_eqcond*D_bundle*D_bundle)^(1/3);
        case 4

            r_eq_b4_number_of_circuit=((r_eqcond*(D_bundle^3)*sqrt(2))^(1/4));
        case 5

            r_eq_b4_number_of_circuit=((r_eqcond*(D_bundle^4)*((1+sqrt(5))/2)^2)^(1/5));
        case 6

            r_eq_b4_number_of_circuit=((r_eqcond*(D_bundle^5)*(2*3))^(1/6));
        case 7
            D_short= (D_bundle^2 + D_bundle^2
            -2*D_bundle*D_bundle*cos(128.57*pi/180))^(1/2);
            D_long = (D_short^2 + D_bundle^2
            -2*D_short*D_bundle*cos(102.85*pi/180))^(1/2);

            r_eq_b4_number_of_circuit=((r_eqcond*(D_bundle^2)*(D_short^2)*(D_long^2))^(1/7));
        case 8
            D_short = (D_bundle^2 + D_bundle^2-
            2*D_bundle*D_bundle*cos(135*pi/180))^(1/2);
            D_medium = (D_short^2 + D_bundle^2 -
            2*D_short*D_bundle*cos(112.5*pi/180))^(1/2);
            D_long = (D_medium^2 + D_bundle^2)^(1/2);

            r_eq_b4_number_of_circuit=(r_eqcond*(D_bundle^2)*(D_short^2)*(D_medium^2)*D_long)
    end

    %Depending on N_circuit, it is finding out GMR , r_eq and GMD
    switch N_circuit
        case 1
            %GMR is again same
            GMR= GMR_bundle;
            %Calculating GMD
            GMD_ab=sqrt((x2-x1)^2 + (y2-y1)^2);
            GMD_ac=sqrt((x3-x1)^2 + (y3-y1)^2);
            GMD_bc=sqrt((x3-x2)^2 + (y3-y2)^2);
            GMD=(GMD_ab*GMD_bc*GMD_ac)^(1/3);
            %r_eq will be same
            r_eq=r_eq_b4_number_of_circuit;
        case 2

```

---

---

```

    %Calculating GMR
    GMR_aa= sqrt(GMR_bundle*sqrt((x1-x4)^2 + (y1-y4)^2));
    GMR_bb= sqrt(GMR_bundle*sqrt((x2-x5)^2 + (y2-y5)^2));
    GMR_cc= sqrt(GMR_bundle*sqrt((x3-x6)^2 + (y3-y6)^2));
    GMR=(GMR_aa*GMR_bb*GMR_cc)^(1/3);
    %Calculating GMD
    GMD_ab=(sqrt((x2-x1)^2 + (y2-y1)^2)*sqrt((x5-x1)^2 + (y5-
y1)^2)*sqrt((x5-x4)^2 + (y5-y4)^2)*sqrt((x2-x4)^2 + (y2-y4)^2))^(1/4);
    GMD_ac=(sqrt((x3-x1)^2 + (y3-y1)^2)*sqrt((x6-x1)^2 + (y6-
y1)^2)*sqrt((x3-x4)^2 + (y3-y4)^2)*sqrt((x6-x4)^2 + (y6-y4)^2))^(1/4);
    GMD_bc=(sqrt((x2-x6)^2 + (y2-y6)^2)*sqrt((x2-x3)^2 + (y2-
y3)^2)*sqrt((x5-x6)^2 + (y5-y6)^2)*sqrt((x5-x3)^2 + (y5-y3)^2))^(1/4);
    GMD=(GMD_ab*GMD_bc*GMD_ac)^(1/3);
    %Calculating R_eq
    Req_aa= sqrt(r_eq_b4_number_of_circuit*sqrt((x1-x4)^2 + (y1-
y4)^2));
    Req_bb= sqrt(r_eq_b4_number_of_circuit*sqrt((x2-x5)^2 + (y2-
y5)^2));
    Req_cc= sqrt(r_eq_b4_number_of_circuit*sqrt((x3-x6)^2 + (y3-
y6)^2));
    r_eq=(Req_aa*Req_bb*Req_cc)^(1/3);

end

%Calculating R,L,C in per m
L= (2e-7)*log(GMD/GMR);
R=R_AC/(N_bundle*N_circuit);
epsilon_zero=8.854187817e-12;
switch N_circuit
    case 1
        H12=sqrt((x1-x2)^2 + (y1-(-y2))^2);
        H13=sqrt((x1-x3)^2 + (y1-(-y3))^2);
        H23=sqrt((x2-x3)^2 + (y2-(-y3))^2);
        H1=2*abs(y1);
        H2=2*abs(y2);
        H3=2*abs(y3);
        Earth_effect=(1/3)*log((H12*H13*H23)/(H1*H2*H3));
        C=(2*pi*(epsilon_zero))/(log(GMD/r_eq)-Earth_effect);
    case 2
        H12=(sqrt((x1-x2)^2 + (y1-(-y2))^2)*sqrt((x1-x5)^2 + (y1-
(-y5))^2)*sqrt((x4-x2)^2 + (y4-(-y2))^2)*sqrt((x4-x5)^2 + (y4-(-
y5))^2))^(1/4);
        H13=(sqrt((x1-x3)^2 + (y1-(-y3))^2)*sqrt((x1-x6)^2 + (y1-
(-y6))^2)*sqrt((x4-x3)^2 + (y4-(-y3))^2)*sqrt((x4-x6)^2 + (y4-(-
y6))^2))^(1/4);
        H23=(sqrt((x2-x3)^2 + (y2-(-y3))^2)*sqrt((x2-x6)^2 + (y2-
(-y6))^2)*sqrt((x5-x3)^2 + (y5-(-y3))^2)*sqrt((x5-x6)^2 + (y5-(-
y6))^2))^(1/4);
        H1=(2*abs(y1)*2*abs(y4)*sqrt((x4-x1)^2 + (y1-(-
y4))^2)*sqrt((x1-x4)^2 + (y4-(-y1))^2))^(1/4);
        H2=(2*abs(y2)*2*abs(y5)*sqrt((x5-x2)^2 + (y5-(-
y2))^2)*sqrt((x2-x5)^2 + (y2-(-y5))^2))^(1/4);
        H3=(2*abs(y3)*2*abs(y6)*sqrt((x3-x6)^2 + (y3-(-
y6))^2)*sqrt((x6-x3)^2 + (y6-(-y3))^2))^(1/4);

```

---

---

```

        Earth_effect=(1/3)*log((H12*H13*H23)/(H1*H2*H3));
        C=2*pi*(epsilon_zero)/(log(GMD/r_eq)-Earth_effect);
end
%Calculating R,X,B
R=R*length;
omega=2*pi*50;
X=omega*L*length;
B=omega*C*length;
%Converting them into per unit system

Z_base= (V_base^2)/(S_base);
R_pu = R/ Z_base;
X_pu = X/ Z_base;
B_pu = B/ (1/Z_base);
end

```

*Published with MATLAB® R2021a*