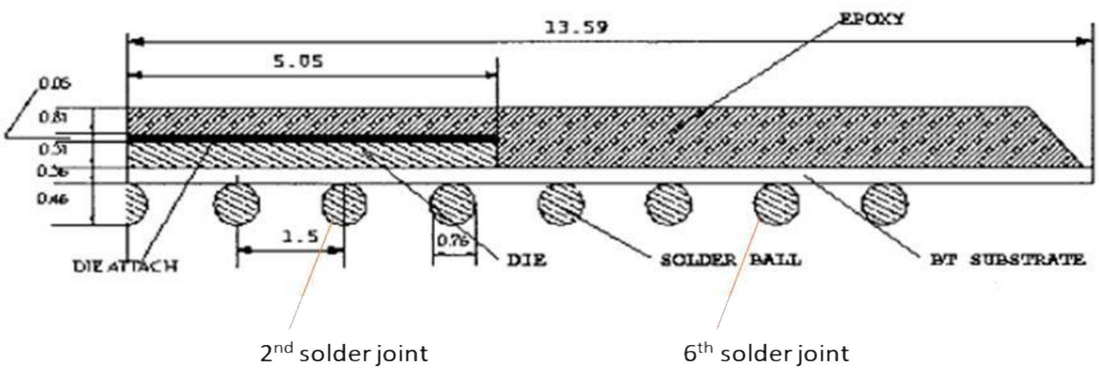
ME 571 Project 2 Report

Umair Sarwar

Date: 1st May 2018

Summary

The purpose of the project was to make use of the analytical model provided, as developed by Heinrich and supplemented by data from Zhang, to understand the causes of uncertainty in the fatigue life of soldered joints in the PBGA. Using the derived analytical expression, *First Order Reliability Methods* were applied to the equation to understand the main variable leading to uncertainty in the measure fatigue life. Owing to the large amount of equations involved with various variables, Matlab's symbolic toolbox was used to assist with the calculations.



Question 1

To achieve the goal of finding the main variable causing the uncertainty in fatigue life, First Order Reliability Method(FORM) was applied to the expression derived by Heinrich and provided in the *Project 2 Manual*. The relation between shear displacement(Δu) between the package, taking into account the solder joint stiffness was derived for the various variables. There are nine independent variables($r_d, r_m, E_d, E_m, \alpha_d, \alpha_m, \alpha_p. \beta_1, \beta_2$) in the equations which, each of which was normally distributed $\sim N(\mu, \sigma)$. Alongside these, there are four deterministic quantities in the expressions which are: $r, \Delta T, v_d, v_m$. To obtain the expression for mean of shear displacement($\mu_{\Delta u}$), the expressions from the manual were changed to input the mean value of the normal variables while the actual value of the deterministic quantities. These relations for $\mu_{\Delta u}$ are shown in eqn.1-3.

$$\overline{\Delta u} = \begin{cases} \overline{\beta_1}[(\overline{\alpha_d} - \overline{\alpha_p}) + \overline{A_1}(\overline{\alpha_m} - \overline{\alpha_d})r\Delta T & 0 \le r \le r_d \\ \overline{\beta_2}[(\overline{\alpha_m} - \overline{\alpha_p}) + \overline{A_2}(\overline{\alpha_d} - \overline{\alpha_m})r\Delta T & r_d \le r \le r_m \end{cases}, 0 \le \beta_1, \beta_2 \le 1 \quad \ldots eqn.1$$

$$\overline{A_1} = \frac{(1 - v_d)[1 - \left(\frac{\overline{r_d}}{\overline{r_m}}\right)^2]}{(1 - v_d)\left[1 - \left(\frac{\overline{r_d}}{\overline{r_m}}\right)^2\right] + \frac{\overline{E_d}}{\overline{E_m}}[(1 + v_m) + (1 - v_m)\left(\frac{\overline{r_d}}{\overline{r_m}}\right)^2]} \quad \ldots. eqn.2$$

$$A_2 = \frac{\frac{\overline{E_d}}{\overline{E_m}}[(1 + v_m)\left(\frac{\overline{r_d}}{r}\right)^2 + (1 - v_m)\left(\frac{\overline{r_d}}{\overline{r_m}}\right)^2]}{(1 - v_d)\left[1 - \left(\frac{\overline{r_d}}{\overline{r_m}}\right)^2\right] + \frac{\overline{E_d}}{\overline{E_m}}[(1 + v_m) + (1 - v_m)\left(\frac{\overline{r_d}}{\overline{r_m}}\right)^2]} \quad \ldots. eqn.3$$

For the second part of this process, the FORM method was applied onto the equations above. The general equation for first reliability methods to obtain the variance is shown below in eqn. 4. For this equation, each of the variable replacing x is one of the nine independent variables talked about above. The *Symbolic* toolbox in Matlab was used to help find the derivatives and the expressions. (The FORM expressions $\sigma^2(\Delta u)$ for can be seen in the Appendix below.)

$$var(\Delta u) = \sigma^2(\Delta u) = \sum_{i=1}^{n}\left(\frac{\partial \Delta u}{\partial x_i}\right)^2 * var(x_i) \quad \ldots eqn.4$$

Question 2 The second question of the project involves the use of the derived expressions in part 1 to obtain certain characteristics. The following data was provided in the project manual: 1. Mean and CV (Coefficient of Variation) Data for 7 independent variables, 2. Shear displacement for the 2nd and 6th joints To make use of the data, the COV was changed into variance by multiplying the CV to mean and squaring the value. After that, the experimental data provided in Table 2 of the *Project Manual* was used. It was seen that the 2nd solder joint fails in the first part of the piecewise range while the 6th joint fails in the second range of the shear displacement equations. Using the Δu data provided as another input to the equation, 4 equations were obtained, each with one unknown. As there is β_1 in the first range, the two equations from the first range ($0 \le r \le r_d$) was used to obtain the μ and σ value for β_1.Same procedure was used to obtain β_2 using the equations ranging from $r_d \le r \le r_m$. *Table 1* below shows the results of the statistics obtained on the two parameters. Fig. 1 and Fig. 2 also show the normal distribution of the statistics of the two beta parameters.

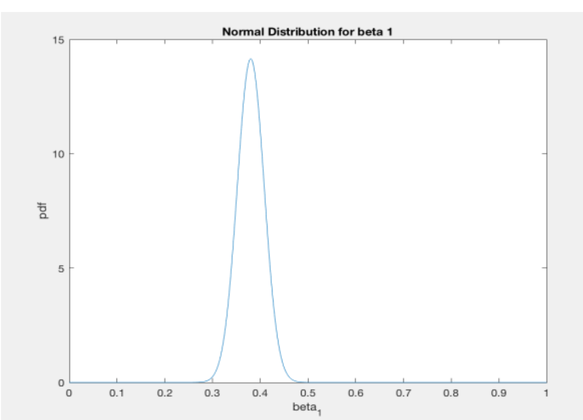


FIGURE 1. BETA 1 DISTRIBUTION

Table 1. Parameter Statistics

Variable	$\boldsymbol{\mu}$	$\boldsymbol{\sigma}$
$\boldsymbol{\beta_1}$	0.3708	0.02818
$\boldsymbol{\beta_2}$	0.2712	0.05744

Question 3

For the third question, the derived equations from question 1 were evaluated using the known values of all the variables. The value of $\mu_{\Delta u}$ and $\sigma^2(\Delta u)$ was evaluated for all the joints, at different values of r. The function of fatigue life was provided to us in the manual. To calculate the mean, the equation was written in the following form:

$$\overline{\emptyset(\Delta u)} = \frac{1}{\Delta u^2} \ldots .\ eqn. 5$$

For the purpose of evaluating the variance of phi, FORM was again applied on the fatigue life function provided. The final equation of the form put in Matlab for evaluation is:

$$var(\emptyset) = \sigma^2(\emptyset) = \frac{\partial \emptyset}{\partial \Delta u}^2 * var(\Delta u) \quad \ldots .\ eqn. 6$$

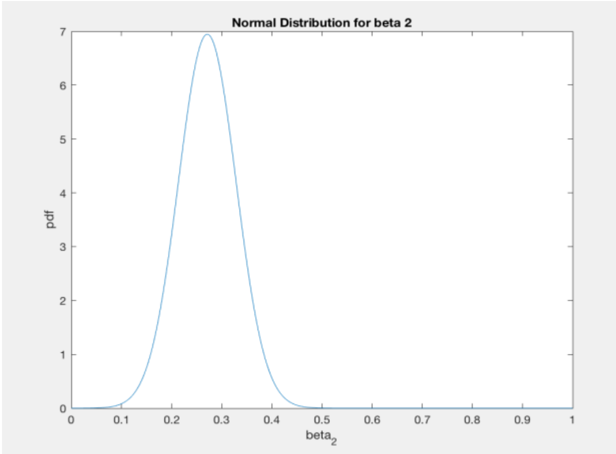


FIGURE 2. BETA 2 DISTRIBUTION

Evaluating each of the expressions for each joint, the value of the mean and standard deviation for each of the joint value was obtained and recorded in the table. The highlighted value in red is fr the critical join. It is join number 3 as it has the lowest μ_ϕ value amongst all the 7 joints. An image of the distribution for joint 7 is also attatched below in Figure 3.

Table 2. Statistics for each joint

Joint Number	$\boldsymbol{\mu_\phi}$	$\boldsymbol{\sigma_\phi}$
1	$2.002 * 10^{12}$	$3.9713 * 10^{11}$
2	$0.5004 * 10^{12}$	$9.9275 * 10^{10}$
3	$0.2224 * 10^{12}$	$4.4159 * 10^{10}$
4	$0.4117 * 10^{12}$	$2.3381 * 10^{11}$
5	$0.5208 * 10^{12}$	$3.0871 * 10^{11}$
6	$0.5951 * 10^{12}$	$3.7567 * 10^{11}$
7	$0.6354 * 10^{12}$	$4.3096 * 10^{11}$

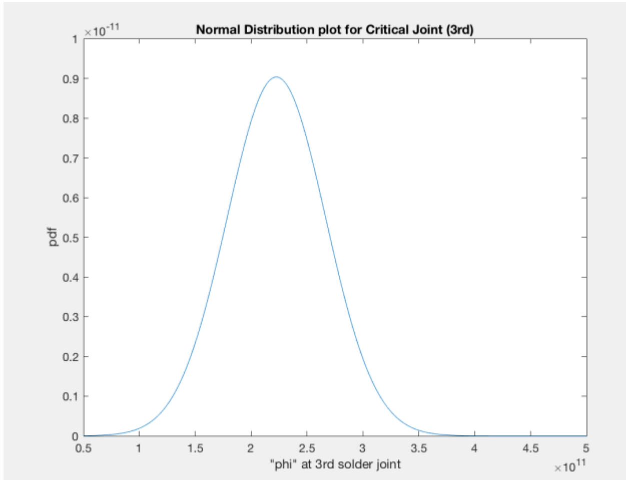


FIGURE 3. CRITICAL JOINT DISTRIBUTION

Question 4

The goal of the last question was to figure out which variable has the highest effect on the fatigue life. For this purpose, FORM was used again. Same general formula used to calculate variance in question 1(eqn. 4) was used to find the variance of phi in terms of each of the independent variables. The difference between using eqn .4 in each of the questions is that in question 4, instead of summing the terms, each term was considered separately using the method: $(Differential\ (\emptyset_{variable}) * variance(variable)$. This was done for each of the 9 variables separately. With all the variables known and r being constant in all the equations, the values were compared to see which variance value was the greatest.

After performing the calculations in Matlab, it was found that β_1 has the biggest effect on the fatigue life. To interpret the results, the fatigue life is most affected by the way $\beta_1$changes in the equations. Changing the value of β_1 would affect the fatigue life the joint the most.

Lessons Learnt

- FORM method on multiple variables and seeing how it effects the parameters
- Understanding how to find variables that effect the uncertainty the most
- Symbolic toolbox on Matlab

Appendices:

<u>Command Script Window for Question 1 as example</u>

```
var_del_1 =

beta1^2*deltat^2*r^2*var_alphap + deltat^2*r^2*var_beta1*(alphap - alphad + ((rd^2/rm^2 -
1)*(alphad - alpham)*(phid - 1))/((rd^2/rm^2 - 1)*(phid - 1) - (Ed*(phim + (rd^2*(phim - 1))/rm^2 -
1))/Em))^2 + beta1^2*deltat^2*r^2*var_rd*((2*rd*(alphad - alpham)*(phid - 1))/(rm^2*((rd^2/rm^2 -
1)*(phid - 1) - (Ed*(phim + (rd^2*(phim - 1))/rm^2 - 1))/Em)) - (((2*rd*(phid - 1))/rm^2 - (2*Ed*rd*(phim
- 1))/(Em*rm^2))*(rd^2/rm^2 - 1)*(alphad - alpham)*(phid - 1))/((rd^2/rm^2 - 1)*(phid - 1) - (Ed*(phim +
(rd^2*(phim - 1))/rm^2 - 1))/Em)^2)^2 + beta1^2*deltat^2*r^2*var_alphad*(((rd^2/rm^2 - 1)*(phid -
1))/((rd^2/rm^2 - 1)*(phid - 1) - (Ed*(phim + (rd^2*(phim - 1))/rm^2 - 1))/Em) - 1)^2 +
beta1^2*deltat^2*r^2*var_rm*(((rd^2/rm^2 - 1)*(alphad - alpham)*(phid - 1)*((2*rd^2*(phid - 1))/rm^3
- (2*Ed*rd^2*(phim - 1))/(Em*rm^3)))/((rd^2/rm^2 - 1)*(phid - 1) - (Ed*(phim + (rd^2*(phim - 1))/rm^2 -
1))/Em)^2 - (2*rd^2*(alphad - alpham)*(phid - 1))/(rm^3*((rd^2/rm^2 - 1)*(phid - 1) - (Ed*(phim +
(rd^2*(phim\\n - 1))/rm^2 - 1))/Em)))^2 + (beta1^2*deltat^2*r^2*var_alpham*(rd^2/rm^2 - 1)^2*(phid
- 1)^2)/((rd^2/rm^2 - 1)*(phid - 1) - (Ed*(phim + (rd^2*(phim - 1))/rm^2 - 1))/Em)^2 +
(beta1^2*deltat^2*r^2*var_Ed*(rd^2/rm^2 - 1)^2*(alphad - alpham)^2*(phid - 1)^2*(phim +
(rd^2*(phim - 1))/rm^2 - 1)^2)/(Em^2*((rd^2/rm^2 - 1)*(phid - 1) - (Ed*(phim + (rd^2*(phim - 1))/rm^2 -
1))/Em)^4) + (Ed^2*beta1^2*deltat^2*r^2*var_Em*(rd^2/rm^2 - 1)^2*(alphad - alpham)^2*(phid -
1)^2*(phim + (rd^2*(phim - 1))/rm^2 - 1)^2)/(Em^4*((rd^2/rm^2 - 1)*(phid - 1) - (Ed*(phim +
(rd^2*(phim - 1))/rm^2 - 1))/Em)^4)

var_del_2 =

beta2^2*deltat^2*r^2*var_alphap + deltat^2*r^2*var_beta2*(alpham - alphap + (Ed*((rd^2*(phim +
1))/r^2 - (rd^2*(phim - 1))/rm^2)*(alphad - alpham))/(Em*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim -
(rd^2*(phim - 1))/rm^2 + 1))/Em)))^2 + beta2^2*deltat^2*r^2*var_Em*((Ed*((rd^2*(phim + 1))/r^2 -
(rd^2*(phim - 1))/rm^2)*(alphad - alpham))/(Em^2*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim - (rd^2*(phim
- 1))/rm^2 + 1))/Em)) - (Ed^2*((rd^2*(phim + 1))/r^2 - (rd^2*(phim - 1))/rm^2)*(alphad - alpham)*(phim -
(rd^2*(phim - 1))/rm^2 + 1))/(Em^3*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim - (rd^2*(phim - 1))/rm^2 +
1))/Em)^2))^2 + beta2^2*deltat^2*r^2*var_Ed*((((rd^2*(phim + 1))/r^2 - (rd^2*(phim -
1))/rm^2)*(alphad - alpham))/(Em*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim - (rd^2*(phim - 1))/rm^2 +
1))/Em)) - (Ed*((rd^2*(phim + 1))/r^2 - (rd^2*(phim - 1))/rm^2)*(alphad - alpham)*(phim - (rd^2*(phim -
1))/rm^2 + 1))/(Em^2*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim - (rd^2*(phim - 1))/rm^2 + 1))/Em)^2))^2 +
beta2\\n^2*deltat^2*r^2*var_rm*((Ed*((rd^2*(phim + 1))/r^2 - (rd^2*(phim - 1))/rm^2)*(alphad -
alpham)*((2*rd^2*(phid - 1))/rm^3 - (2*Ed*rd^2*(phim - 1))/(Em*rm^3)))/(Em*((rd^2/rm^2 - 1)*(phid -
1) + (Ed*(phim - (rd^2*(phim - 1))/rm^2 + 1))/Em)^2) + (2*Ed*rd^2*(alphad - alpham)*(phim -
1))/(Em*rm^3*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim - (rd^2*(phim - 1))/rm^2 + 1))/Em)))^2 +
beta2^2*deltat^2*r^2*var_alpham*((Ed*((rd^2*(phim + 1))/r^2 - (rd^2*(phim -
1))/rm^2))/(Em*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim - (rd^2*(phim - 1))/rm^2 + 1))/Em)) - 1)^2 +
beta2^2*deltat^2*r^2*var_rd*((Ed*((2*rd*(phim + 1))/r^2 - (2*rd*(phim - 1))/rm^2)*(alphad -
alpham))/(Em*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim - (rd^2*(phim - 1))/rm^2 + 1))/Em)) -
(Ed*((rd^2*(phim + 1))/r^2 - (rd^2*(phim - 1))/rm^2)*((2*rd*(phid - 1))/rm^2 - (2*Ed*rd*(phim -
1))/(Em*rm^2))*(alphad - alpham))/(Em*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim - (rd^2*(phim - 1))/rm^2
```

+ 1))/Em)^2))^2 + (Ed^2*beta2^2*deltat^2*r^2*var_alphad*((rd^2*(phim\\\n + 1))/r^2 - (rd^2*(phim - 1))/rm^2)^2)/(Em^2*((rd^2/rm^2 - 1)*(phid - 1) + (Ed*(phim - (rd^2*(phim - 1))/rm^2 + 1))/Em)^2)

mu_delt1 =

-deltat*mu_beta1*r*(mu_alphap - mu_alphad + ((mu_rd^2/mu_rm^2 - 1)*(mu_alphad - mu_alpham)*(phid - 1))/((mu_rd^2/mu_rm^2 - 1)*(phid - 1) - (mu_Ed*(phim + (mu_rd^2*(phim - 1))/mu_rm^2 - 1))/mu_Em))

mu_delt2 =

-deltat*mu_beta2*r*(mu_alphap - mu_alpham + (mu_Ed*(mu_alphad - mu_alpham)*((mu_rd^2*(phim - 1))/mu_rm^2 - (mu_rd^2*(phim + 1))/r^2))/(mu_Em*((mu_rd^2/mu_rm^2 - 1)*(phid - 1) + (mu_Ed*(phim - (mu_rd^2*(phim - 1))/mu_rm^2 + 1))/mu_Em)))

Matlab Code

```
clc
clear all
%%UMAIR SARWAR
%%ME 571-Reliability Based Design

%INPUT DATA PROVIDED
del_2_joint = 10^-6 * [1.275 1.589 1.37 1.558 1.427 1.248 1.656 1.326 1.347 1.338];
del_6_joint = 10^-6 * [1.536 .46 1.356 1.714 1.508 1.499 1.231 1.127 .791 1.743];
mu_del_2_joint = mean(del_2_joint);
mu_del_6_joint = mean(del_6_joint);
var_del_2_joint = var(del_2_joint);
var_del_6_joint = var(del_6_joint);

%QUESTION 1
syms rd mu_rd var_rd rm mu_rm var_rm Em mu_Em var_Em Ed mu_Ed var_Ed alphad mu_alphad var_alphad alpham
mu_alpham var_alpham alphap mu_alphap var_alphap beta1 mu_beta1 var_beta1 beta2 mu_beta2 var_beta2 r
deltat phid phim

A_1 = ((1 - phid)*(1 - (rd/rm)^2)) / ((1 - phid)*(1 - (rd/rm)^2) + (Ed/Em)*((1 - phim) + (1 -
phim)*(rd/rm)^2));
A_2 = ((Ed/Em)*((1 + phim)*(rd/r)^2 + (1 - phim)*(rd/rm)^2)) / ((1 - phid)*(1 - (rd/rm)^2) +
(Ed/Em)*((1 + phim) + (1 - phim)*(rd/rm)^2));
del_1 = beta1*((alphad - alphap) + A_1*(alpham - alphad))*r*deltat;
del_2 = beta2*((alpham - alphap) + A_2*(alphad - alpham))*r*deltat;
var_del_1 = ((diff(del_1,rd)^2) * var_rd) + ((diff(del_1,rm)^2) * var_rm) + ((diff(del_1,Ed)^2) *
var_Ed) + ((diff(del_1,Em)^2) * var_Em) + ((diff(del_1,alphad)^2) * var_alphad) +
((diff(del_1,alpham)^2) * var_alpham) + ((diff(del_1,alphap)^2) * var_alphap) + ((diff(del_1,beta1)^2)
* var_beta1) + ((diff(del_1,beta2)^2) * var_beta2)
var_del_2 = ((diff(del_2,rd)^2) * var_rd) + ((diff(del_2,rm)^2) * var_rm) + ((diff(del_2,Ed)^2) *
var_Ed) + ((diff(del_2,Em)^2) * var_Em) + ((diff(del_2,alphad)^2) * var_alphad) +
((diff(del_2,alpham)^2) * var_alpham) + ((diff(del_2,alphap)^2) * var_alphap) + ((diff(del_2,beta1)^2)
* var_beta1) + ((diff(del_2,beta2)^2) * var_beta2)
mu_A1 = ((1 - phid)*(1 - (mu_rd/mu_rm)^2)) / ((1 - phid)*(1 - (mu_rd/mu_rm)^2) + (mu_Ed/mu_Em)*((1 -
phim) + (1 - phim)*(mu_rd/mu_rm)^2));
mu_A2 = ((mu_Ed/mu_Em)*((1 + phim)*(mu_rd/r)^2 + (1 - phim)*(mu_rd/mu_rm)^2)) / ((1 - phid)*(1 -
(mu_rd/mu_rm)^2) + (mu_Ed/mu_Em)*((1 + phim) + (1 - phim)*(mu_rd/mu_rm)^2));
mu_delt1 = mu_beta1*((mu_alphad - mu_alphap) + mu_A1*(mu_alpham - mu_alphad))*r*deltat
mu_delt2 = mu_beta2*((mu_alpham - mu_alphap) + mu_A2*(mu_alphad - mu_alpham))*r*deltat

%QUESTION 2
%INPUT SETUP FOR QUESTION 2
deltat = -100;
mu_rd = 5.05/1000; %Conversion of mm to m
var_rd = ((10/100)*mu_rd)^2;
mu_rm = 13.59/1000; %Conversion of mm to m
var_rm = ((10/100)*mu_rm)^2;
mu_Ed = 130*(10^9);%GPa to Pa Conversion
var_Ed = ((5/100)*mu_Ed)^2;
mu_Em = 15.5*(10^9); %GPa to Pa Conversion
var_Em = ((5/100)*mu_Em)^2;
```

```
phid = 0.28;
phim = 0.25;
mu_alphad = 2.62*(10^-6);
var_alphad = (.05*mu_alphad)^2;
mu_alphap = 16*(10^-6);
var_alphap = (.05*mu_alphap)^2;
mu_alpham = 15*(10^-6);
var_alpham = (.05*mu_alpham)^2;
r_0 = 0; r_1 = .0015; r_2 = r_1 + .0015; r_3 = r_2 + .0015; r_4 = r_3 + .0015; r_5 = r_4 + .0015; r_6 =
r_5 + .0015; r_7 = r_6 + .0015;

%Calculating mu_beta1
r = .003;
mu_delt1 = subs(mu_delt1)
mean_beta1 = (mu_del_2_joint*684684565331771543098291650560000000) / 2545472288561474981500803282063;

%Calculating mu_beta2
r = r_6;
mu_delt2 = subs(mu_delt2)
mean_beta2 = (mu_del_6_joint*337204723530369031171566940979200000000) /
16118383318339683907712809836942 3;

%Calculating var_beta1
r = .003;
rd = mu_rd;
rm = mu_rm;
Ed = mu_Ed;
Em = mu_Em;
alphad = mu_alphad;
alpham = mu_alpham;
alphap = mu_alphap;
beta1 = .3801;
beta2 = .2712;
var_del_1 = subs(var_del_1)
variance_beta1 = ((var_del_2_joint -
1323511586406733850273497586424868815877854650771572394269378639089482193650048944031384800731 3/1524938
742804162247811778803037710223233248099090360943835241553472614486117368081503200462168391680000000000)
* 468792954003556934441337498823432806950277276729809148313600000000000000) /
6479429171834392955946468963312955328100925058931072737535969;
var_beta1 = .00079426;
cov_beta1_percent = ((variance_beta1^.5) / mean_beta1) * 100

%Calculating var_beta2
r = .009;
rd = mu_rd;
rm = mu_rm;
Ed = mu_Ed;
Em = mu_Em;
alphad = mu_alphad;
alpham = mu_alpham;
alphap = mu_alphap;,
beta1 = .3801;
beta2 = .2712;
var_del_2 = subs(var_del_2)
variance_beta2 = ((var_del_6_joint -
(504886523597677804263906926215947841871626249477520968538405700448860353736076913747106455174618 97/547
576631807326298429377880509328951596991584245950850756740410584609991210340496961970689071927787520000 0
00000)) * 113707025571192613769264500843563515030180205547360296254832640000000000000000) /
25980228079693099998705779637851318160733208465877390831381352929;
var_beta2 = .0033;
cov_beta2_percent = ((variance_beta2^.5) / mean_beta2) * 100

%eta 1 and Beta 2 statistics
xB1 = 0:.000001:1;
normB1 = normpdf(xB1,.3802,.02818262);
xB2 = 0:.000001:1;
normB2 = normpdf(xB2,.2712,.05744562647);
figure
plot(xB1,normB1)
title('Normal Distribution for beta 1')
xlabel('beta_1')
ylabel('pdf')
figure
plot(xB2,normB2)
title('Normal Distribution for beta 2')
```

```
xlabel('beta_2')
ylabel('pdf')

%QUESTION 3
clear
syms rd mu_rd var_rd rm mu_rm var_rm Ed mu_Ed var_Ed Em mu_Em var_Em alphad mu_alphad var_alphad alpham
mu_alpham var_alpham alphap mu_alphap var_alphap beta1 mu_beta1 var_beta1 beta2 mu_beta2 var_beta2 r
deltat phid phim

%aFirst Order Reliability method
A1 = ((1 - phid)*(1 - (rd/rm)^2)) / ((1 - phid)*(1 - (rd/rm)^2) + (Ed/Em)*((1 - phim) + (1 -
phim)*(rd/rm)^2));
A2 = ((Ed/Em)*((1 + phim)*(rd/r)^2 + (1 - phim)*(rd/rm)^2)) / ((1 - phid)*(1 - (rd/rm)^2) + (Ed/Em)*((1
+ phim) + (1 - phim)*(rd/rm)^2));
delt_1 = beta1*((alphad - alphap) + A1*(alpham - alphad))*r*deltat;
phi1 = 1 / (delt_1^2);
delt_2 = beta2*((alpham - alphap) + A2*(alphad - alpham))*r*deltat;
phi2 = 1 / (delt_2^2);
var_phi1 = ((diff(phi1,rd)^2) * var_rd) + ((diff(phi1,rm)^2) * var_rm) + ((diff(phi1,Ed)^2) * var_Ed) +
((diff(phi1,Em)^2) * var_Em) + ((diff(phi1,alphad)^2) * var_alphad) + ((diff(phi1,alpham)^2) *
var_alpham) + ((diff(phi1,alphap)^2) * var_alphap) + ((diff(phi1,beta1)^2) * var_beta1) +
((diff(phi1,beta2)^2) * var_beta2)
var_phi2 = ((diff(phi2,rd)^2) * var_rd) + ((diff(phi2,rm)^2) * var_rm) + ((diff(phi2,Ed)^2) * var_Ed) +
((diff(phi2,Em)^2) * var_Em) + ((diff(phi2,alphad)^2) * var_alphad) + ((diff(phi2,alpham)^2) *
var_alpham) + ((diff(phi2,alphap)^2) * var_alphap) + ((diff(phi2,beta1)^2) * var_beta1) +
((diff(phi2,beta2)^2) * var_beta2)
var_delt1 = ((diff(delt_1,rd)^2) * var_rd) + ((diff(delt_1,rm)^2) * var_rm) + ((diff(delt_1,Ed)^2) *
var_Ed) + ((diff(delt_1,Em)^2) * var_Em) + ((diff(delt_1,alphad)^2) * var_alphad) +
((diff(delt_1,alpham)^2) * var_alpham) + ((diff(delt_1,alphap)^2) * var_alphap) +
((diff(delt_1,beta1)^2) * var_beta1) + ((diff(delt_1,beta2)^2) * var_beta2);
var_delt2 = ((diff(delt_2,rd)^2) * var_rd) + ((diff(delt_2,rm)^2) * var_rm) + ((diff(delt_2,Ed)^2) *
var_Ed) + ((diff(delt_2,Em)^2) * var_Em) + ((diff(delt_2,alphad)^2) * var_alphad) +
((diff(delt_2,alpham)^2) * var_alpham) + ((diff(delt_2,alphap)^2) * var_alphap) +
((diff(delt_2,beta1)^2) * var_beta1) + ((diff(delt_2,beta2)^2) * var_beta2);
mu_A1 = ((1 - phid)*(1 - (mu_rd/mu_rm)^2)) / ((1 - phid)*(1 - (mu_rd/mu_rm)^2) + (mu_Ed/mu_Em)*((1 -
phim) + (1 - phim)*(mu_rd/mu_rm)^2));
mu_A2 = ((mu_Ed/mu_Em)*((1 + phim)*(mu_rd/r)^2 + (1 - phim)*(mu_rd/mu_rm)^2)) / ((1 - phid)*(1 -
(mu_rd/mu_rm)^2) + (mu_Ed/mu_Em)*((1 + phim) + (1 - phim)*(mu_rd/mu_rm)^2));
mu_delt1 = mu_beta1*((mu_alphad - mu_alphap) + mu_A1*(mu_alpham - mu_alphad))*r*deltat;
mu_delt2 = mu_beta2*((mu_alpham - mu_alphap) + mu_A2*(mu_alphad - mu_alpham))*r*deltat;

mu_phi1 = 1 / (mu_delt1^2)
mu_phi2 = 1 / (mu_delt2^2)

%Setting all inputs again
rd = .00505;mu_rd = .00505;var_rd = 2.5503 * 10^-7;rm = .01359;mu_rm = .01359;var_rm = 1.8469 * 10^-6;
Ed = 1.3 * 10^11;mu_Ed = 1.3 * 10^11;var_Ed = 4.225 * 10^19;Em = 1.55 * 10^10;mu_Em = 1.55 *
10^10;var_Em = 6.0063 * 10^17;alphad = 2.62 * 10^-6;mu_alphad = 2.62 * 10^-6;var_alphad = 1.7161 * 10^-
14;
alpham = 1.5 * 10^-5;mu_alpham = 1.5 * 10^-5;var_alpham = 5.625 * 10^-13;alphap = 1.6 * 10^-5;mu_alphap
= 1.6 * 10^-5;var_alphap = 6.4 * 10^-13;beta1 = .3801;mu_beta1 = .3801;var_beta1 = 7.9426 * 10^-4;beta2
= .2712;mu_beta2 = .2712;var_beta2 = .0033;deltat = -100;phid = .28;phim = .25;

%Changing equations to make them in terms of r
mu_delt1 = subs(mu_delt1)
mu_delt2 = subs(mu_delt2)
var_delt1 = subs(var_delt1)
var_delt2 = subs(var_delt2)

%Calculating mean and variance of phi at each joint

i = 1; %Loop counter
for r = 0.0015:0.0015:0.0105
    if r < 0.005
        mean_delta_u(i,:) =
(3225113389607388836730889821373821*r)/6846845653317715430982916505600000000
        mean_phi_joints(i,:) = 1 / (mean_delta_u(i,:) ^ 2);
        variance_delta_u(i,:) =
(208164637460033652969024970646783125263028519866600223932478741627157456913219497597790544038528 9*r^2)
/95308671425260140488236175189856888952078006193147558989702597092038405382335505093950028885524480000
000
        variance_phi_joints(i,:) = ((-2 / (mean_delta_u(i,:) ^ 3)) ^ 2) * variance_delta_u(i,:);
    else
```

```
        mean_delta_u(i,:) =
(678*r*(2386459648261925641029491451/(8632440922377447197992113689067520000*r^2) +
4000337180156305167798654879l7/2107529522064806444822293381l2000000))/25
        mean_phi_joints(i,:) = 1 / (mean_delta_u(i,:) ^ 2);
        variance_delta_u(i,:) = subs(var_delt2);
        variance_phi_joints(i,:) = ((-2 / (mean_delta_u(i,:) ^ 3)) ^ 2) * variance_delta_u(i,:);
    end
    i = 1+i;
end
mean_delta_u_joints = mean_delta_u;
variance_delta_u_joints = variance_delta_u;
mean_phi_joints = mean_phi_joints
variance_phi_joints = variance_phi_joints
std_dev_phi_joints = variance_phi_joints .^ .5

%Plotting for critical point(3rd Joint selected from analysis)
xphi_3rd = 0.05*10^12:1000000:.5*10^12;
normphi_3rd = normpdf(xphi_3rd,10^12 * .2226,10^11 * .4417);

figure
plot(xphi_3rd,normphi_3rd)
title('Normal Distribution plot for Critical Joint (3rd)')
xlabel('"phi" at 3rd solder joint')
ylabel('pdf')

%QUESTION 4
clear

syms rd mu_rd var_rd rm mu_rm var_rm Ed mu_Ed var_Ed Em mu_Em var_Em alphad mu_alphad var_alphad alpham
mu_alpham var_alpham alphap mu_alphap var_alphap beta1 mu_beta1 var_beta1 beta2 mu_beta2 var_beta2 r
deltat phid phim

A1 = ((1 - phid)*(1 - (rd/rm)^2)) / ((1 - phid)*(1 - (rd/rm)^2) + (Ed/Em)*((1 - phim) + (1 -
phim)*(rd/rm)^2));
A2 = ((Ed/Em)*((1 + phim)*(rd/r)^2 + (1 - phim)*(rd/rm)^2)) / ((1 - phid)*(1 - (rd/rm)^2) + (Ed/Em)*((1
+ phim) + (1 - phim)*(rd/rm)^2));
delt_1 = beta1*((alphad - alphap) + A1*(alpham - alphad))*r*deltat;
phi1 = 1 / (delt_1^2);
delt_2 = beta2*((alpham - alphap) + A2*(alphad - alpham))*r*deltat;
phi2 = 1 / (delt_2^2);

%Partial differentials

var_phi1_rd = ((diff(phi1,rd)^2) * var_rd);var_phi1_rm = ((diff(phi1,rm)^2) * var_rm);var_phi1_Ed =
((diff(phi1,Ed)^2) * var_Ed);
var_phi1_Em = ((diff(phi1,Em)^2) * var_Em);var_phi1_alphad = ((diff(phi1,alphad)^2) * var_alphad)
;var_phi1_alpham = ((diff(phi1,alpham)^2) * var_alpham) ;
var_phi1_alphap = ((diff(phi1,alphap)^2) * var_alphap);var_phi1_beta1 = ((diff(phi1,beta1)^2) *
var_beta1);var_phi1_beta2 = ((diff(phi1,beta2)^2) * var_beta2);
rd = .0051;mu_rd = .0051;var_rd = 2.5503 * 10^-7;rm = .0136;mu_rm = .0136;var_rm = 1.8469 * 10^-6;Ed =
1.3 * 10^11;mu_Ed = 1.3 * 10^11;
var_Ed = 4.225 * 10^19;Em = 1.55 * 10^10;mu_Em = 1.55 * 10^10;var_Em = 6.0063 * 10^17;alphad = 2.62 *
10^-6;mu_alphad = 2.62 * 10^-6;
var_alphad = 1.7161 * 10^-14;alpham = 1.5 * 10^-5;mu_alpham = 1.5 * 10^-5;var_alpham = 5.625 * 10^-
13;alphap = 1.6 * 10^-5;
mu_alphap = 1.6 * 10^-5;var_alphap = 6.4 * 10^-13;beta1 = .3801;mu_beta1 = .3801;var_beta1 = 7.9426 *
10^-4;beta2 = .2712;
mu_beta2 = .2712;var_beta2 = .0033;deltat = -100;phid = .28;phim = .25;
r = 1;

%Highest value for partial differential multiplied by variance calculated with r = 1 for all

subs(var_phi1_rd);subs(var_phi1_rm);subs(var_phi1_Ed);subs(var_phi1_Em);subs(var_phi1_alphad);subs(var_
phi1_alpham);subs(var_phi1_alphap);subs(var_phi1_beta1);subs(var_phi1_beta2)

rd_eff =
3981487763415650052983834139043493464326095258065041739160741146544977784132137790526754356802734935058
7372823518799462400000000000000000000000000000000000000000000000000/28531173913278513785340158169900
9986216419944435976576780799055289962820164369380406626941371752071811604653450384165376440883814757381
0015624884528429548794500054663
rm_eff =
2838298522881492761609351163273874448154162888702282346263909422554805430723518995718483466858000150783
3032010967516446720000000000000000000000000000000000000000000000000/19971821739294959649738110718 9
3699035149396110518360374655933870297397411505856628463885896022645026812325741526891576350861867033016
67010937419169900684156150038264l
```

```
Ed_eff =
6720217538360268066112970567638497644588488188594918773378177248710691340647029984569711964991645081553
7878404000352807881277440000000000000000000000000000000000000000000000/62195984655243818978423182169699
9699527493598431323300065861758916144522614280610940310486650650700391381718814602913050441411878794118
71918227423152462193956759321
Em_eff =
2240091160632856223960837190171743789257958327140210093848982088598322602806979185366413373587475149682
22079255191019948651996774000000000000000000000000000000000000000000/20731994885081272992807727389899
9899842497866143774433355287252972048174204760203646770162216883566797127239604867637683480470626264706
23972742474384154064652253107
alpahd_eff =
1432574323300953099727179484016261956234424837986973131968204029856655264652515472569778720331754815349
239190821367300885577728000000000000000000000000000000000000000000000000/186587953965731456935269546509
0999098582480795293969900197585276748433567842841832820931459951952101174145156443808739151324235636382
35615754682269457386581870277963
alpham_eff =
1164058107903764089534712374145882724760065642061289384632698999274929331765577800091181142348939328137
952693161173935632198664192000000000000000000000000000000000000000000/6219598465524381897842318216969
9969952749359843132330006586175891614452261428061094031048665065070039138171881460291305044141187879411
871918227423152462193956759321
alphap_eff =
1891149663979759649962610006857868233347674738174233174074601608319250290369091008829661870803123531803
620102763965148427638961864704000000000000000000000000000000000000000000/5597638618971943708058086395
2729972957474423858819097005927558302453007035285254984627943798558563035224354693314262174539727069091
4706847264046808372159745610833889
beta1_eff=
2884723083233431638623237367425760753434455632330273780445580906426455943727866021257758638080000000000
000000000000000000000000000000/6467755997545135243367537863673586789807352960690406615532248459543185 11
13942304439863977563479680198022917533518779526 49
beta2_eff = 0
```