Generating Pipeline Chart by Using SAS/GRAPH

Jeff Y. Yuan

ABSTRACT

Pipeline Chart is a graphical representation of the capability and the centering of parameters that are critical to a process or a production line. This paper illustrates the use of SAS/GRAPH to generate the informative Pipeline Chart. In the chart, the horizontal axis has the name of the parameter and the vertical axis is the normalized measurement of all parameters. Two horizontal and parallel lines represent the specification limits of the parameters and another horizontal line, in the middle of the specification lines, represents the target value. For every selected parameter, statistics such as mean, standard deviation, Cp, Cpk, and the failure rate of all data collected during the reported period are calculated and displayed in the chart.

INTRODUCTION

As companies are confronted with the challenge of handling and analyzing an ever-increasing amount of data, it has become more difficult to present the desired information derived from the data in a simple and clear format. During the manufacturing and development of a single high-technology semiconductor product, thousands of electrical tests are performed to understand and model the variations caused by process and design conditions and to make statistical control on the process. The Pipeline Chart is a very useful tool to report in one chart the status of the critical and key parameters of a process and to have a synthetic overview of the capability of critical process parameters. However, there is no SAS procedure or macro available to generate the Pipeline Chart. In this paper, Pipeline Chart is generated to present information derived from a large amount of data in a simple graphical report by using PROC GPLOT and the Annotate facility in SAS/GRAPH. We have applied our SAS codes to several large data sets extracted from databases and conclude that it is faster and more convenient to get the Pipeline Chart by using SAS software than by using other software packages.

DATA MANIPULATION

The data can be from a manufacturing or a business process and can be extracted from any database or data file. The underlying distribution of each parameter in the chart is a Normal distribution and the data set should include information on the upper specification limit (USL), the lower specification limit (LSL) and the target (usually, it is the average of the upper limit and the lower limit) for each parameter. In addition, a new variable, called value_a, should be created to normalize the observation value of each parameter so that all parameters are transformed to be in [-1, 1].

value_a = 2 * (value - target) / (USL - LSL) (1)

The steps on the data manipulation vary based on the sources of the data. After necessary manipulation by using PROC SQL, PROC TRANSPOSE or Data Steps, the data table should include the following columns: Para (the name of the parameter), Value (the value of the parameter before normalization), Value_a (the value of the parameter after normalization), USL, LSL and Target. Then, PROC MEANS, PROC SQL and DATA MERGE Step can be used to get the summary data.

Table 1. mean - the summary data set

Para	maen_b	sigma_b	mean_a	sigma_a	USL	LSL	target
bv1a1n_	-8.95	0.12	-0.30	0.08	-7	-10	-8
bv2a1n_	-8.97	0.08	-0.31	0.05	-7	-10	-8
…	…	…	…	…	…	…	
rr35a1n	6.68	0.85	-0.19	0.12	15	1	8

where mean_b and sigma_b are the mean and the standard deviation of variable Value which are used to calculate the Cp and Cpk of the parameter, mean_a and sigma_a are the mean and the standard deviation of value_a which are used to create a new data set for the Annotate facility in generating the chart. By transformation (1), it can be mathematically proved that

mean_a = 2 * (mean_b - target) / (USL - LSL)
sigma_a = 2 * sigma_b / (USL - LSL).

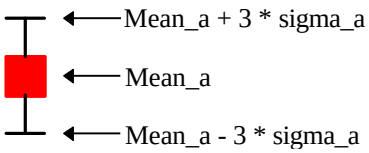
Mean_b and sigma_b are used to calculate the two capability indexes Cp and Cpk. The formulas for the calculation are

Cp = (USL - LSL) / (6 * sigma_b)
Cpl = (mean_b - LSL) / (3 * sigma_b)
Cpu = (USL - mean_b) / (3 * sigma_b)
Cpk = min (Cpl, Cpu).

THE ANNOTATE DATA SET

After normalization, the upper specification limits for all parameters are equal to 1 and all the lower specification limits are equal to -1. They are represented in the chart by two straight lines.

The mean of a parameter is represented by a square. The variation ranges, represented on the graph by a segment, are reduced proportionally in the report in order to represent them with the same normalized limits.



Cp and Cpk of the parameter are written below the pattern, in alignment with the plotted lines. Additional information can be added to the chart and will be written in the same vertical alignment.

Cp and Cpk are written in two colors: the ones with Cpk value less then 1.33 are written in red, the others are written in green. (This can not see in the white/black color printout). This feature will help management or the customer easily identify problems in the process or help production by viewing the pipeline chart and then defining an action plan.

The Annotate facility in SAS is very important to integrate all the above features into one chart. Building an Annotate data set is the key to using the Annotate facility. The SAS code listed below provides the detail to build the Annotate data set for a pipeline chart. Figure 1 was generated by using the following code:

```
/************************************************/
/* The below SAS codes do not include the part for */
/* data manipulation. It should be easy to get the */
/* data set (called mean) that includes variables  */
/* para, mean_b, sigma_b, mean_a, sigma_a, usl,    */
/* lsl and target.                                 */
/* The observations that are not in the validation */
/* range should be deleted, and additional         */
/* information such as number of observations      */
/* (&freq in the below codes) can be also          */
/* calculated during data manipulation.            */
/* See Table 1 for the contents of dataset mean    */
/************************************************/


%let key=Critical;

data final;
  set mean;
  format cp 4.1;

  /*********************************************/
  /*  Use 99 to stand for the very big Cp and Cpk */
  /*  so that there is enough room to write them  */
  /*  on the chart.                              */
  /*********************************************/
  if sigma_a <= 0.00001 then
     do;
     cp=99; cpk=99;
     end;
  else do;
     /*****************************************/
     /* Calculate Cp and Cpk by using the data  */
     /* before being normalized.                */
     /*****************************************/
     cp=(usl-lsl)/(6*sigma_b);
     cpl=(mean_b-lsl)/(3*sigma_b);
     cpu=(usl-mean_b)/(3*sigma_b);
     cpk=min(cpl,cpu);
  end;
run;
```

```
data draw; set mean end=last;
   retain xsys ysys '2' hsys '3';
   length text $ 4 color cc $ 8;
        /*****************************************************/
        /* Define a variable for the colors for Cp and Cpk. */
        /*****************************************************/
        if cpk <=1.33 then cc='red';
        else cc='green';
        /*****************************************************/
        /* Limit the plotting points within the defined range  */
        /* (in this example the range is (-2.5, 2.5)) on the   */
        /* vertical axis.                                      */
        /*****************************************************/
        if (mean_a+3*sigma_a)>2.5 then ub=(2.49-mean_a)/3;
        else ub=sigma_a;
        if (mean_a-3*sigma_a)<-2.5 then lb=(mean_a+2.51)/3;
        else lb=sigma_a;
     y=mean_a - 3*lb;
     xc=para;
     function ='label'; text='-'; size=1; output;
     FUNCTION = 'MOVE'; output;
     y=mean_a + 3*ub;
     function = 'LABEL'; text='-'; size=1; output;
     function = 'DRAW'; size=0.01; output;
     function = 'LABEL'; y=-2.7; text=left(put(cp,4.1)); size=1.5; style='simplex'; color=cc; output;
     function = 'LABEL'; y=-3.2; text=left(put(cpk,4.1)); size=1.5; style='simplex'; color=cc; output;
   if LAST then do;
      xsys=1; ysys=2;
```

```
      FUNCTION = 'MOVE'; x=0;   y=0;  output;
      FUNCTION = 'DRAW'; x=100; y=0;  line=3; size=0.01; color='green'; output;
      FUNCTION = 'MOVE'; x=0;   y=-1; output;
      FUNCTION = 'DRAW'; x=100; y=-1; line=1; size=0.01; color='blue'; output;
      FUNCTION = 'MOVE'; x=0;   y=1;  output;
      FUNCTION = 'DRAW'; x=100; y=1;  line=1; size=0.01; color='blue'; output;
      FUNCTION = 'LABEL'; x=-3; y=1.1; text='Spec';size=2; style='swiss'; output;
      FUNCTION = 'LABEL'; x=-3; y=0.1; text='tgt'; size=2; style='swiss'; output;
      FUNCTION = 'LABEL'; x=-3; y=-0.9; text='Spec';size=2; style='swiss';output;
      FUNCTION = 'Label'; x=-3; y=-2.7; text='Cp: '; size=2.5; style='swiss';output;
      FUNCTION = 'Label'; x=-3; y=-3.2; text='Cpk:'; size=2.5; style='swiss';output;
    end;

/*****************************************************************************/
/* The goptions work with SAS on SUN/OS and printers HP LaserJet and HP DeskJet. */
/*****************************************************************************/
goptions RESET =all dev=xcolor targetdevice=dj1600c cback=white hsize=12 vsize=6
         border gaccess=sasgastd gsfname=gsasfile gsfmode=replace vpos=90;
filename gsasfile pipe 'lp -d$PRINTER -oraw';

title1 h=5.0 'PIPELINE CHART';
title2 h=3.0 font=swiss "&key Parameters";
title3 h=2.0 font=swiss j=c "&freq Data Points";
footnote1 h=2.5 font=swiss j=l 'Weekly Review';
footnote2 h=1.5 font=swiss j=r "Generated on &sysdate";
proc gplot data=mean;
    plot mean_a * para
        /vaxis=axis1 haxis=axis2 vminor=0
         annotate=draw;
    symbol1 font=marker v=U h=2 c=red;
    axis1   order=(-3.5 to 2.7 by 0.5) label=none value=none major=none;
    axis2   label=(height=3 font=swiss "Parameter")
            value=(j=c font=simplex h=1.2) origin=(6 pct,) offset=(2,);
run;
quit;
```

SUMMARY

The Pipeline Chart is a useful tool to visualize the information from a large amount of data. By using it, the management can easily evaluate the capability of a process or a production line.

The Annotate facility in SAS provides the flexibility to generate custom graphics. It is not difficult to add additional information to the Pipeline Chart. For example, Figure 2 provides information on Cp and Cpk, as well as the failure rate for every parameter.

The SAS codes listed above can be easily change to a macro so that it can be used repeatedly.

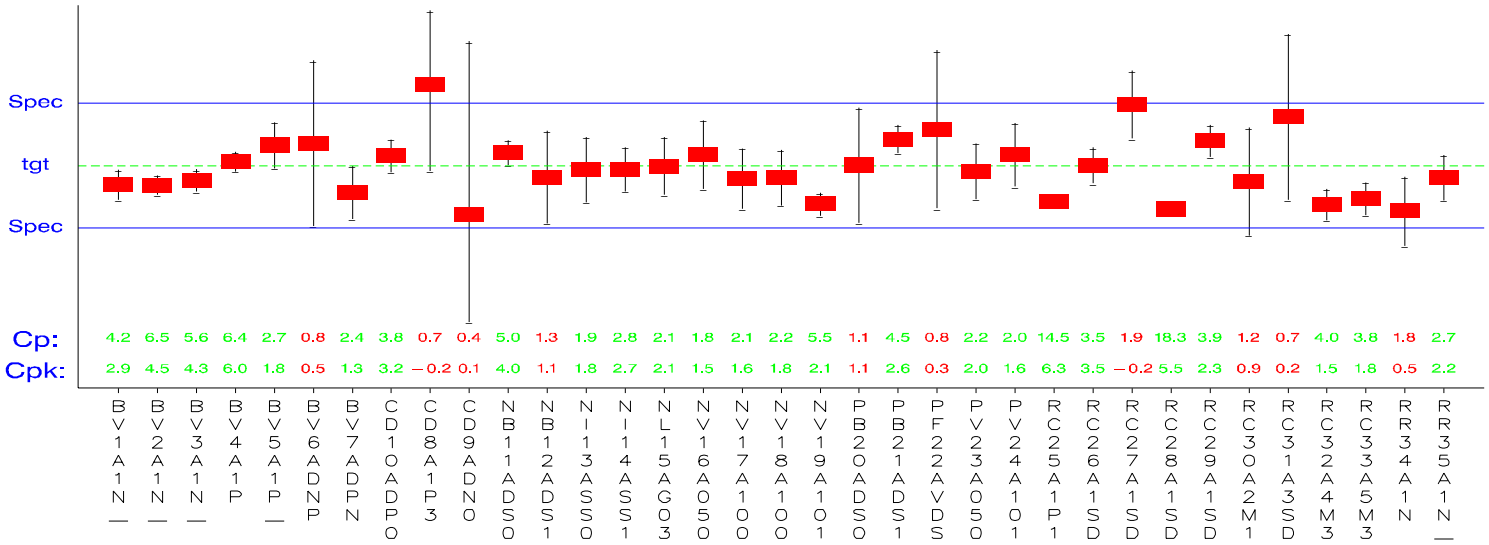
ACKNOWLEDGMENT

I would like to thank Robert Hodges and other colleagues for bringing the topic to me and for the valuable suggestions during my testing on the SAS program.

PIPELINE CHART

Critical Parameters

489 Data Points



Parameter

Weekly Review

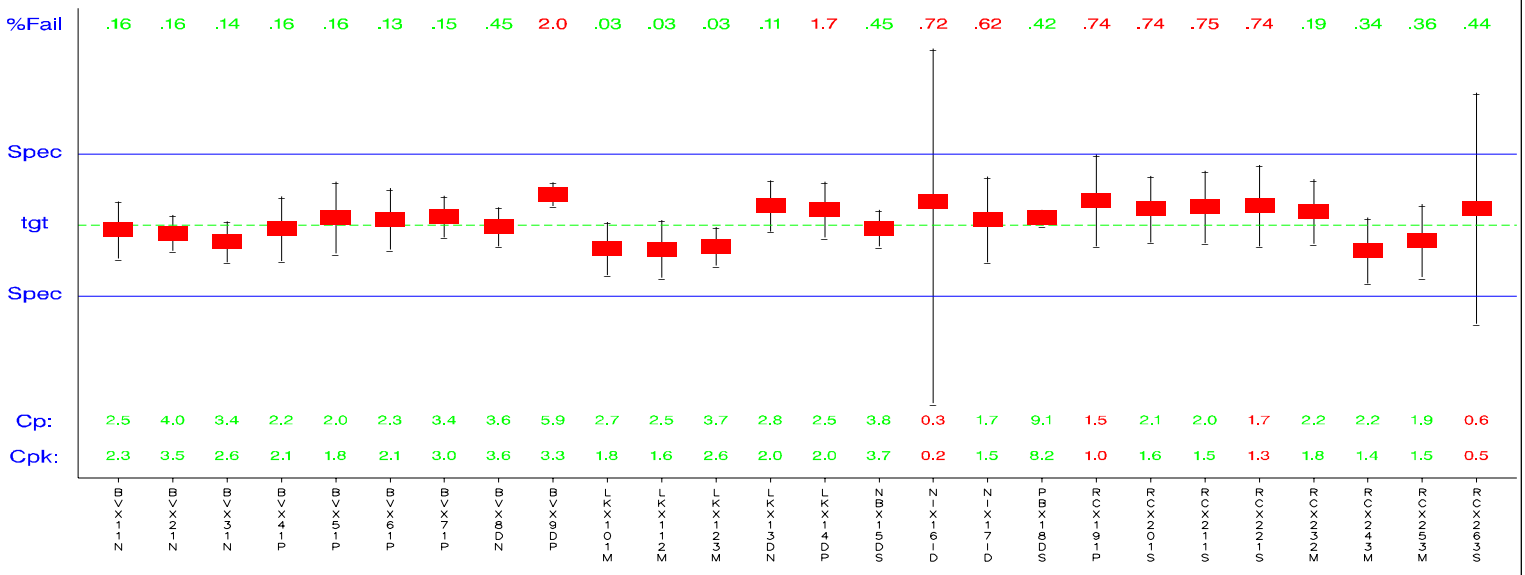
Generated on 24JUL98

Figure 1

Pipeline Chart

Date: 31/MAY/98 to 04/JUL/98

Total 125 Lots



Parameter

Generated on 24JUL98

Figure 2