

Creating Composite Wafer Maps for the Semiconductor Industry with PROC GMAP

by

Michael K. Nash and Robert Barrantes
Fujitsu Microelectronics Inc.

ABSTRACT

This paper describes a useful way to construct composite wafer maps using the PROC GMAP procedure and the ANNOTATE facility within SAS/GRAPH. Fujitsu Microelectronics in Gresham, Oregon produces DRAM memory chips, each chip (die) is built on a silicon wafer. Each wafer contains several hundred die, and each production lot usually contains 50 wafers. A wafer map is a graphical representation of a silicon wafer and all the die contained on that wafer. We will provide detailed instructions on how to create a sample choropleth wafer map with 30 die per wafer and how to annotate statistical information on to the map. We have also included an example of a full sized composite wafer map with 260 die per wafer to be used as an example of what can be achieved with our SAS program. We also hope that readers will be able to use this sample program as a simple guide on how to use PROC GMAP and/or the ANNOTATE facility in order to create choropleth maps for fields unrelated to the semiconductor industry.

INTRODUCTION

As stated above Fujitsu Microelectronics in Gresham, Oregon produces DRAM memory chips which are used primarily in personal computers. Each production lot of 50 wafers begins the same way; the lot is tracked into a production stage, some type of processing is done on each wafer in the lot, a data collection step is performed to gather processing results, the lot is tracked out of the stage, and then the lot is ready to be tracked into the next processing stage. There are about 90 of these steps or stages that each lot must undergo before it reaches the end of the production process.

When a lot has completed the production process, it enters the "testing" or Probe Test area. Each die on every wafer is tested for its functionality, each die can either pass (perfect), be repaired, or fail for any number of reasons. These failures are sometimes referred to as "bin failures". There are several categories of bin failure, but for this paper we will discuss only two types of failures: Row, and column defects. In order to analyze each failure category, a separate composite wafer map is needed for each type of bin failure.

As stated earlier each lot usually contains 50 wafers, so for each wafer, a map could be constructed highlighting each die and its test result (i.e. perfect, repair, row defects, column defects...etc.). A composite wafer map is simply the sum of all the individual wafer maps compressed into one wafer map. A composite wafer map is a very useful item because an engineer can quickly identify the most prevalent reasons for failures and also identify any patterns within a specific lot.

Using PROC GMAP we have constructed an exact copy of the die pattern for one of the devices we build at Fujitsu. The wafer maps we use at Fujitsu contains several hundred die but for obvious reasons we won't construct those maps here. The composite map we will build is a very simple map with only 30 die per wafer and five wafers per lot.

THE MAP

In order for PROC GMAP to be able to draw the choropleth wafer map, each die must be a box with X and Y coordinates to describe each corner. So, to draw a wafer map for 30 die, SAS needs 120 observations to describe the location of all the die. Determining the exact coordinates for each die on the wafer is probably the most difficult and time consuming task when creating composite wafer maps. To ascertain the exact coordinates for the die on the wafer map we drew the die pattern on a piece of graph paper. As you can imagine it is fairly easy to determine the coordinates for a wafer map with 30 die, but a map with 300 die is much more tedious and time consuming. Each die is a 1 x 1 square with its own unique number from 1 to 30. We numbered the die from top to bottom and left to right. As you can see in the SAS code below we have used a simple input and cards statement to list the die coordinates on the wafer:

```
DATA WORK.MAP;  
INPUT DIE X Y;  
CARDS;
```

```
1 0 3  
1 1 3  
1 1 4  
1 0 4  
2 0 2  
2 1 2  
2 1 3  
2 0 3  
3 1 4  
3 2 4  
3 2 5  
3 1 5  
4 1 3  
4 2 3  
4 2 4  
4 1 4  
5 1 2  
5 2 2  
5 2 3  
5 1 3  
6 1 1  
6 2 1  
6 2 2  
6 1 2  
7 2 5  
7 3 5  
7 3 6  
7 2 6
```

824
834
835
825
923
933
934
924
1022
1032
1033
1023
1121
1131
1132
1122
1220
1230
1231
1221
1335
1345
1346
1336
1434
1444
1445
1435
1533
1543
1544
1534
1632
1642
1643
1633
1731
1741
1742
1732
1830
1840
1841
1831
1945
1955
1956
1946
2044
2054
2055
2045
2143
2153

21 5 4
21 4 4
22 4 2
22 5 2
22 5 3
22 4 3
23 4 1
23 5 1
23 5 2
23 4 2
24 4 0
24 5 0
24 5 1
24 4 1
25 5 4
25 6 4
25 6 5
25 5 5
26 5 3
26 6 3
26 6 4
26 5 4
27 5 2
27 6 2
27 6 3
27 5 3
28 5 1
28 6 1
28 6 2
28 5 2
29 6 3
29 7 3
29 7 4
29 6 4
30 6 2
30 7 2
30 7 3
30 6 3
;
RUN;

DATA WORK.MAP;
SET WORK.MAP;
FORMAT SEGMENT 2.;
SEGMENT = 1;
RUN;

THE TEST RESULT DATA

At Fujitsu, the test result data comes from a flat (ASCII) file which contains information about each die within the wafer and for each wafer in the lot. This flat file comes directly off the tester equipment in the fab.

For the sake of simplicity, we have provided the sample test result data here in the paper as opposed to referring to an established SAS data set. We have also listed the bin results for five wafers as opposed to all fifty. Each die for all five wafers within the lot (4530) has a specific test result, and these results are stored in a temporary SAS data set named "WORK.FAILURE". Once again we have used the input and cards statements in a data step to input the test result data into SAS. In the SAS code listed below, "COL" refers to a "column" failure, "REPAIR" to a "repairable" die, "ROW" to a "row" defect, and "PERF" refers to a "perfect" die.

```
DATA WORK.FAILURE;
INPUT LOTID WFID DIE_1 $ DIE_2 $ DIE_3 $ DIE_4 $ DIE_5 $ DIE_6 $ DIE_7 $ DIE_8 $ DIE_9 $
DIE_10 $ DIE_11 $ DIE_12 $ DIE_13 $ DIE_14 $ DIE_15 $ DIE_16 $ DIE_17 $ DIE_18 $ DIE_19 $ DIE_20 $
DIE_21 $ DIE_22 $ DIE_23 $ DIE_24 $ DIE_25 $ DIE_26 $ DIE_27 $ DIE_28 $ DIE_29 $ DIE_30 $;
CARDS;
4530 1 COL PERF REPAIR REPAIR PERF REPAIR COL ROW PERF COL ROW COL ROW
      COL REPAIR PERF REPAIR ROW REPAIR PERF ROW PERF ROW PERF COL ROW
      PERF REPAIR REPAIR COL
4530 2 ROW COL PERF REPAIR COL REPAIR REPAIR PERF REPAIR PERF PERF COL ROW
      PERF REPAIR PERF REPAIR PERF REPAIR COL PERF PERF REPAIR ROW COL ROW
      ROW PERF REPAIR COL
4530 3 PERF PERF COL COL ROW REPAIR PERF ROW COL COL PERF COL ROW
      COL REPAIR PERF REPAIR PERF REPAIR PERF ROW COL REPAIR PERF ROW PERF
      ROW REPAIR REPAIR ROW
4530 4 REPAIR REPAIR PERF PERF COL REPAIR ROW PERF COL COL ROW COL ROW
      COL REPAIR PERF REPAIR REPAIR REPAIR COL ROW COL ROW PERF REPAIR ROW
      ROW REPAIR REPAIR COL
4530 5 COL REPAIR ROW REPAIR COL REPAIR COL ROW PERF COL PERF COL ROW
      COL REPAIR PERF REPAIR ROW REPAIR ROW ROW COL REPAIR PERF ROW PERF
      ROW PERF REPAIR PERF
;
RUN;

PROC SORT DATA = WORK.FAILURE NODUP;
BY LOTID WFID;
RUN;
```

Listed below is a printout of the data set WORK.FAILURE. The data set has been truncated to fit the page, however, in reality the data set contains observations for all the variables up to DIE_30.

<u>LOTID</u>	<u>WFID</u>	<u>DIE_1</u>	<u>DIE_2</u>	<u>DIE_3</u>	<u>DIE_4</u>	etc...
4530	1	COL	PERF	REPAIR	REPAIR	
4530	2	ROW	COL	PERF	REPAIR	
4530	3	PERF	PERF	COL	COL	
4530	4	REPAIR	REPAIR	PERF	PERF	
4530	5	COL	REPAIR	ROW	REPAIR	

PROC TRANSPOSE

We need to count the number of bin failures at each die position so we can annotate that number directly on top of the die position in the choropleth composite wafer map. In order to count the number of bin failures at each die position, it is easiest to use PROC TRANSPOSE to transpose the failure data and store it as one variable called “_NAME_”. Once this is done, a PROC FREQ is needed to count the number of observations in each category (i.e. perfect, repair, column, row) for each die position.. Once we have the correct failure count for each die we must use the ANNOTATE facility when we run the PROC GMAP to correctly place the number at each die location.

```
PROC TRANSPOSE DATA = WORK.FAILURE OUT = WORK.INFO;  
  BY LOTID WFID;  
  VAR DIE_1-DIE_30;  
RUN;
```

```
PROC SORT DATA = WORK.INFO;  
  BY LOTID _NAME_;  
RUN;
```

Below is the SAS data set WORK.INFO. The entire data set contains 150 observations. (Five wafers x 30 die).

OBS	LOTID	WFID	_NAME_	COL1
1	4530	1	DIE_1	COL
2	4530	2	DIE_1	ROW
3	4530	3	DIE_1	PERF
4	4530	4	DIE_1	REPAIR
5	4530	5	DIE_1	COL
6	4530	1	DIE_10	COL
7	4530	2	DIE_10	PERF
8	4530	3	DIE_10	COL
9	4530	4	DIE_10	COL
10	4530	5	DIE_10	COL
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

```
PROC FREQ DATA=WORK.INFO;  
  BY LOTID _NAME_;  
  TABLES COL1/OUT = WORK.INFO2;  
RUN;
```

At the top of the next page is the SAS data set WORK.INFO2. This data set contains only 66 observations because we are counting the total number of bin failures at each die position. So, for example, four out of five wafers have a “column failure” for DIE_10 and one wafer has a ‘perfect’ die.

<u>OBS</u>	<u>LOTID</u>	<u>NAME</u>	<u>COL1</u>	<u>COUNT</u>	<u>PERCENT</u>
1	4530	DIE_1	COL	2	40
2	4530	DIE_1	PERF	1	20
3	4530	DIE_1	REPAIR	1	20
4	4530	DIE_1	ROW	1	20
5	4530	DIE_10	COL	4	80
6	4530	DIE_10	PERF	1	20
7	4530	DIE_11	PERF	3	60
8	4530	DIE_11	ROW	2	40
9	4530	DIE_12	COL	5	100
10	4530	DIE_13	ROW	5	100
.					
.					
.					

DIE POSITION

Each die will have a number (from 1 to 5) annotated directly on top of its position on the wafer. This value (variable 'count' in data set WORK.INFO2) is the number of observations for each particular category. In order to correctly position the values on top of the die, each die must have the specific X and Y coordinates for the center of the die. What we have found is that when annotating the values on a choropleth map, it is best to use the midpoint of the X coordinates for the X value. However for the Y-value it is best to use 3/5 of the distance from the bottom Y-coordinate to top Y-coordinate. These new coordinates for each die will position the annotated value directly over the center of the die. If we were creating a wafer map with several hundred die, we could use a SAS data step to calculate the X and Y coordinates for the annotated values. But since we are creating a map with only 30 die per wafer we will use a cards statement to input the data into a temporary SAS data set named WORK.DIE_POS.

```
DATA WORK.DIE_POS;
  INPUT DIE X Y;
  CARDS;
1 0.5 3.6
2 0.5 2.6
3 1.5 4.6
4 1.5 3.6
5 1.5 2.6
6 1.5 1.6
7 2.5 5.6
8 2.5 4.6
9 2.5 3.6
10 2.5 2.6
11 2.5 1.6
12 2.5 0.6
13 3.5 5.6
14 3.5 4.6
15 3.5 3.6
16 3.5 2.6
17 3.5 1.6
```

```

18 3.5 0.6
19 4.5 5.6
20 4.5 4.6
21 4.5 3.6
22 4.5 2.6
23 4.5 1.6
24 4.5 0.6
25 5.5 4.6
26 5.5 3.6
27 5.5 2.6
28 5.5 1.6
29 6.5 3.6
30 6.5 2.6
;
RUN;

```

THE ANNOTATE FACILITY

For the annotated data set, several variables are required by the ANNOTATE facility to describe the data. Some of these variables are self explanatory like 'COLOR', 'STYLE', and 'SIZE', but others like 'HYSYS', 'XSYS', 'YSYS', 'WHEN', and 'FUNCTION' may require some reference. All of these variables are described in *SAS/GRAPH Reference Volume 1*.

```

DATA WORK.DIE_POS;
LENGTH FUNCTION COLOR STYLE $8. SIZE 2.;
RETAIN XSYS YSYS '2' HSYS '3' WHEN 'A';
SET WORK.DIE_POS;
FUNCTION='LABEL'; COLOR='BLACK'; STYLE='SWISS'; SIZE=5;
RUN;

```

```

PROC SORT DATA = WORK.DIE_POS;
BY DIE;
RUN;

```

SAS Data set name WORK.DIE_POS

FUNCTION	COLOR	STYLE	SIZE	XSYS	YSYS	HSYS	WHEN	DIE	X	Y
LABEL	BLACK	SWISS	5	2	2	3	A	1	0.5	3.6
LABEL	BLACK	SWISS	5	2	2	3	A	2	0.5	2.6
LABEL	BLACK	SWISS	5	2	2	3	A	3	1.5	4.6
LABEL	BLACK	SWISS	5	2	2	3	A	4	1.5	3.6
LABEL	BLACK	SWISS	5	2	2	3	A	5	1.5	2.6
LABEL	BLACK	SWISS	5	2	2	3	A	6	1.5	1.6
LABEL	BLACK	SWISS	5	2	2	3	A	7	2.5	5.6
LABEL	BLACK	SWISS	5	2	2	3	A	8	2.5	4.6
LABEL	BLACK	SWISS	5	2	2	3	A	9	2.5	3.6
LABEL	BLACK	SWISS	5	2	2	3	A	10	2.5	2.6
LABEL	BLACK	SWISS	5	2	2	3	A	11	2.5	1.6
LABEL	BLACK	SWISS	5	2	2	3	A	12	2.5	0.6
LABEL	BLACK	SWISS	5	2	2	3	A	13	3.5	5.6
LABEL	BLACK	SWISS	5	2	2	3	A	14	3.5	4.6

LABEL	BLACK SWISS	5	2	2	3	A	15	3.5	3.6
LABEL	BLACK SWISS	5	2	2	3	A	16	3.5	2.6
LABEL	BLACK SWISS	5	2	2	3	A	17	3.5	1.6
LABEL	BLACK SWISS	5	2	2	3	A	18	3.5	0.6
LABEL	BLACK SWISS	5	2	2	3	A	19	4.5	5.6
LABEL	BLACK SWISS	5	2	2	3	A	20	4.5	4.6
LABEL	BLACK SWISS	5	2	2	3	A	21	4.5	3.6
LABEL	BLACK SWISS	5	2	2	3	A	22	4.5	2.6
LABEL	BLACK SWISS	5	2	2	3	A	23	4.5	1.6
LABEL	BLACK SWISS	5	2	2	3	A	24	4.5	0.6
LABEL	BLACK SWISS	5	2	2	3	A	25	5.5	4.6
LABEL	BLACK SWISS	5	2	2	3	A	26	5.5	3.6
LABEL	BLACK SWISS	5	2	2	3	A	27	5.5	2.6
LABEL	BLACK SWISS	5	2	2	3	A	28	5.5	1.6
LABEL	BLACK SWISS	5	2	2	3	A	29	6.5	3.6
LABEL	BLACK SWISS	5	2	2	3	A	30	6.5	2.6

MERGE IT

In order to make four separate composite wafer maps (one for each category), we decided to use a SAS macro routine. A temporary data set named WORK.COMP is created consisting of the values in WORK.INFO2 where COL1 is equal to the appropriate category (PERF, REPAIR, COL, or ROW). This data set is then merged with WORK.DIE_POS in order to give each die a category value (variable 'text' in data set WORK.COMP) and X - Y coordinates to position this value over the appropriate die.

```
%MACRO MAP (PARAM,NAME);
```

```
DATA WORK.COMP;
SET WORK.INFO2;
IF COL1 = "&PARAM";
FORMAT DIE 2.;
DIE = SUBSTR(_NAME_,5,3);
TEXT = LEFT(PUT(COUNT,5.));
RUN;
```

```
PROC SORT DATA = WORK.COMP;
BY DIE;
RUN;
```

```
DATA WORK.COMP;
MERGE WORK.COMP WORK.DIE_POS;
BY DIE;
IF LOTID = . THEN DELETE;
RUN;
```

At the top of the next page is the complete data set WORK.COMP for the "PERFECT" category. This data contains an observation for each die on the composite wafer map.

LOTID	NAME	COLI	COUNT	PERCENT	DIE	TEXT	FUNCTION	COLOR	STYLE	SIZE	XSYS	YSYS	HSYS	WHEN	Y
4530	DIE_1	PERF	1	20	1	1	LABEL	BLACK	SWISS	5	2	2	3	A	0.5 3.6
4530	DIE_2	PERF	2	40	2	2	LABEL	BLACK	SWISS	5	2	2	3	A	0.5 2.6
4530	DIE_3	PERF	2	40	3	2	LABEL	BLACK	SWISS	5	2	2	3	A	1.5 4.6
4530	DIE_4	PERF	1	20	4	1	LABEL	BLACK	SWISS	5	2	2	3	A	1.5 3.6
4530	DIE_5	PERF	1	20	5	1	LABEL	BLACK	SWISS	5	2	2	3	A	1.5 2.6
4530	DIE_7	PERF	1	20	7	1	LABEL	BLACK	SWISS	5	2	2	3	A	2.5 5.6
4530	DIE_8	PERF	2	40	8	2	LABEL	BLACK	SWISS	5	2	2	3	A	2.5 4.6
4530	DIE_9	PERF	2	40	9	2	LABEL	BLACK	SWISS	5	2	2	3	A	2.5 3.6
4530	DIE_10	PERF	1	20	10	1	LABEL	BLACK	SWISS	5	2	2	3	A	2.5 2.6
4530	DIE_11	PERF	3	60	11	3	LABEL	BLACK	SWISS	5	2	2	3	A	2.5 1.6
4530	DIE_14	PERF	1	20	14	1	LABEL	BLACK	SWISS	5	2	2	3	A	3.5 4.6
4530	DIE_16	PERF	5	100	16	5	LABEL	BLACK	SWISS	5	2	2	3	A	3.5 2.6
4530	DIE_18	PERF	2	40	18	2	LABEL	BLACK	SWISS	5	2	2	3	A	3.5 0.6
4530	DIE_20	PERF	2	40	20	2	LABEL	BLACK	SWISS	5	2	2	3	A	4.5 4.6
4530	DIE_21	PERF	1	20	21	1	LABEL	BLACK	SWISS	5	2	2	3	A	4.5 3.6
4530	DIE_22	PERF	2	40	22	2	LABEL	BLACK	SWISS	5	2	2	3	A	4.5 2.6
4530	DIE_24	PERF	4	80	24	4	LABEL	BLACK	SWISS	5	2	2	3	A	4.5 0.6
4530	DIE_26	PERF	2	40	26	2	LABEL	BLACK	SWISS	5	2	2	3	A	5.5 3.6
4530	DIE_27	PERF	1	20	27	1	LABEL	BLACK	SWISS	5	2	2	3	A	5.5 2.6
4530	DIE_28	PERF	2	40	28	2	LABEL	BLACK	SWISS	5	2	2	3	A	5.5 1.6
4530	DIE_30	PERF	1	20	30	1	LABEL	BLACK	SWISS	5	2	2	3	A	6.5 2.6

PROC FORMAT

A PROC FORMAT procedure is used to correctly classify the response variable 'COUNT' into numerical ranges. The DISCRETE option is invoked in the PROC GMAP procedure in order to make each formatted value into a response level.. Pattern statements are also included so that each value in the legend has a different color.

```
PROC FORMAT;
  VALUE COUNTFMT 1 - 2 = '1-2'
                3 - 4 = '3-4'
                5 - 6 = '5';
```

```
RUN;
```

```
PATTERN1 VALUE=MSOLID COLOR = RED;
PATTERN2 VALUE=MSOLID COLOR = YELLOW;
PATTERN3 VALUE=MSOLID COLOR = BLUE;
PATTERN4 VALUE=MSOLID COLOR = GREEN;
```

PROC GMAP

Finally, the PROC GMAP procedure draws the choropleth map, giving each die a color based on the formatting scheme. Once this is done the ANNOTATE facility appends the values for the variable 'text' directly above each die. Each map is given a name so that the graphs can be replayed using a PROC GREPLAY statement. A legend is added to quickly identify numerical values with a distinct color. We have also included two title statements; the first identifies the lot

number, while the second picks up the macro variable "&NAME" which is the test result category (bin failure).

```
GOPTIONS RESET=ALL NODISPLAY BORDER ROTATE=PORTRAIT DEVICE=PS300
HTITLE = 1.0 CM HTEXT = 0.5 CM FTEXT = CENTB;
```

```
TITLE1 "Composite Map for Lot 4530";
TITLE2 H=3 "Category = &NAME";
```

```
LEGEND1 LABEL=(FONT=ZAPF "&NAME" POSITION=(LEFT MIDDLE)) FRAME;
```

```
PROC GMAP MAP=WORK.MAP DATA=COMP ALL GOUT = GRAPH;
FORMAT COUNT COUNTFMT.;
ID DIE;
CHORO COUNT/ COUTLINE = GRAY ANNOTATE = COMP DISCRETE LEGEND=LEGEND1
NAME = "&NAME";
RUN;
QUIT;
```

```
%MEND;
```

```
%MAP (PERF,Perfect);
%MAP (REPAIR,Repair);
%MAP (COL,Column);
%MAP (ROW,Row);
```

CONCLUSION

Composite wafer maps created using SAS/GRAPH are a helpful yet easy way to analyze bin failure data. We have provided an example of how to create a sample wafer map and how to annotate the fail data on the choropleth composite wafer map. The principles introduced here in this paper are the same whether you intend to build a wafer map with 30 die or 300 die per wafer.

REFERENCES

SAS Institute Inc. (1990) *SAS/GRAPH Software Volume 1, Reference, Version 6, First Edition*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1990) *SAS/GRAPH Software Volume 2, Reference, Version 6, First Edition*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1990) *SAS/GRAPH Software, Usage, Version 6, First Edition*, Cary, NC: SAS Institute Inc.

AUTHORS BIOGRAPHIES

Michael K. Nash - has worked at Fujitsu Microelectronics in Gresham, Oregon for two and a half years as an Engineering Support Specialist. Prior to coming to Fujitsu, he worked in Eastern Europe in the import/export industry. He graduated from Oregon State University in 1992 with a B.S. degree in Business Administration.

Robert Barrantes - Database Analyst at EDS/Fujitsu since 1990. SAS user for 14 years. Fujitsu SAS System Manager. B.S. degree in Physiology from Louisiana State University. Completed graduate coursework in Statistics. Responsible for data analysis, production reporting for all departments, and user support at Fujitsu.

CONTACT INFORMATION

Michael K. Nash
Fujitsu Microelectronics, Inc.
Gresham Manufacturing Division
21015 S.E. Stark Street
Gresham, Oregon 97030
(503) 669-6120
e-mail: mike.nash@gmd.fujitsu.com

Robert Barrantes
EDS/Fujitsu Microelectronics, Inc.
Gresham Manufacturing Division
21015 S.E. Stark Street
Gresham, Oregon 97030
(503) 669-6159
e-mail: robert.barrantes@gmd.fujitsu.com

GRAPHICS OUTPUT

Below is the graphics output for the sample wafer maps that we have constructed in this paper. Each choropleth composite map (one for each test result category) is replayed on a four window template using PROC GREPLAY.



