Speed-up of Memory Failure Shape Analysis

†JYH-MING JANG, HSING-CHUNG LIANG, and HWANG-CHERNG CHOW

†Nanya Tech. Corporation
336 Sec 1, Nankan Rd., Luchu
Tao Yuan 338, Taiwan, R.O.C.

Dept. of Electronic Eng., Chang Gung Univ.
No. 259, Wen Hua 1st Road, Kuei Shan
Tao Yuan 333, Taiwan, R.O.C.

Abstract: A failure shape analysis system is developed to quickly analyze the defect distribution of memories. Compared to the original system, the newer one consists of several programs feasible both on workstations and PCs. These programs transform and grasp the required data for further failure analysis. The complete system is highly reliable and fault tolerant for the entire manufacture and test procedure. It can achieve the similar defect distribution as the original one but with only 11% and 6% of run time on workstations and PCs, respectively.

Key-Words: defect cell, failure shape, fail pattern, fail bit map, wafer fail bit map, failure shape analysis

1 Introduction

For the advancement of semiconductor technology in recent years, it becomes rather difficult to achieve higher production rate with stable yield at the same time. The distribution of defect cells on a die or a wafer can be analyzed to diagnosis the possible causes of failures. There are about thirty kinds of failure shapes or fail patterns for DRAMs[1], which demand judging methods for later repair decision. It seems that utilizing many conditions may positively judge each type of failure shape, nevertheless it will take lengthy time and be difficult to determine an exact failure shape. This is because one failure may include some failure shapes and one failure shape may belong to several failures. The used order of failure shapes therefore determines the obtained distribution graph of those failures.

1.1 Fail Bit Map (FBM)

It's a physical planar graph for demonstrating the failure locations on a die[2,3] and accordingly being helpful for analyzing the failure shapes. The similar graph is also feasible for the failure distribution on a wafer, called the wafer fail bit map (WFB).

The growing volume of memories costs it lengthy time to extract the FBM data. Compaction methods with ratios ranging from several to hundreds were therefore proposed to speed up diagnosing the wafer failures[2-5].

1.2 Fail Shape Analysis (FSA)

Present FSA or fail pattern recognition[4,5] defines appropriate failure shapes and their order during the decision procedure. The memory data of a wafer is compacted and then analyzed by multiple CPUs or computers. The results are then utilized to determine the repair solutions and improve the wafer yields.

This study targets on developing a complete FSA system that can analyze the defect cells in shorter time to achieve the distribution graphs. The revised programs can obtain the similar distribution graphs as the original ones; however, they can be run on PCs with 17 times faster speed than the original ones performable on workstations. Section 2 describes the FSA system and its analyzing programs, including the revised ones. Section 3 summarizes the considered failure shapes with their order and the revised FSA system. The experimental results are given in section 4 for proving the effectiveness. The last section concludes the study and our future work.

2 WFB and FSA System

In our revised WFB and FSA system, the FBM data for each wafer will be transferred to the Utopia PCs for FSA analysis. The resultant data of FSA (W-file) and that of WFB (KLA-file) will then be processed by Utopia server to generate the D-file and M-file. The contents of these files are as follows:

FBM-file : This file is generated from the testing machine, recording the data of logic failures. In order to reduce the run time of data extraction and FSA analysis, we compact the corresponding data with ratio of the sixteenth.

W-file : This file counts up the number of each type of failure shape occurred on a chip. As the counted numbers of failure shape are not appropriate for describing the faulty behavior of a lot of wafers, they are recalculated into another forms described in the M-file and D-file.

KLA-file : The location addresses of the defect cells are provided in this file for yield analysis system to generate the WFB graph.

M-file : This file calculates the percentage of die number of each failure shape with respect to the total number of dies on a wafer.

D-file : For each type of failure shape, this file counts up the occurring numbers in all dies on a wafer and is then divided by the number of these dies to get the average value.

2.1 Revised FSA

Fig. 1 shows the original flow of FSA programs, including a program of transforming the FBM file into another one with physical fail addresses for those defect cells. The transformed file can then be feasible for FSA analysis. Because of large data volume for the FBM file, the process needs lengthy time to read/write the hard disks. Consequently, in Fig. 2, the revised FSA flow combines the procedures of address transformation and FSA analysis, which specifically speed up the system run time.

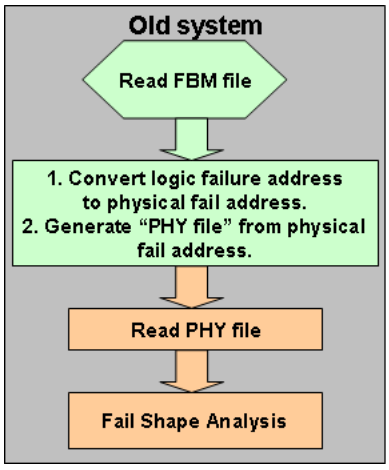

Fig. 1: Original FSA flow

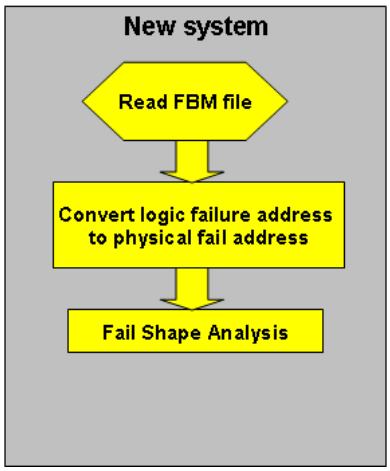

Fig. 2: Revised FSA flow

2.2 Order of failure shapes

The considered order of failure shapes is as follows: Bank, Block, Cross, Y-dec(M2), SA, C-row(M1), C-col(M0), 2-row, 1-row, P-row, 2-col, 1-col, 1-BL, P-col, M-bit, C-bit, 2-bit, 1-bit. Every failure shape has its respective counting range for determination. The order is important because of the following two reasons:

1. correctly decide the failure shape : For the Cross failure as an example, its shape is a cross of one row and one column. The Y-dec and SA failures are both on columns, being lengthier than that of Cross. If a Cross failure is coupled with other failures on the same column, it will be forgot on the corresponding column because that column will be considered as part of a Y-dec or SA failures. Therefore, Cross failures must be considered in prior order.
2. speed up the analysis procedure : The failures occupying larger area are therefore considered in prior order.

3 Experimental Results

The revised system and the original one are compared as follows, including the speed and distribution graph. To speed up the process, only one failure shape will be identified for each defect cell.

The original system can only be run on workstations. Sun Blade 1500, with 1GHz IIIi CPU and 4GB RAM, and Utopia PC, with XEON 1.2GHz double CPUs and 2GB RAM, were used for comparison. Fig. 3 compares the normalized run time on a wafer for the revised and original systems. It's obvious that the revised one is 9 times faster than the original if running on the same workstation. It can have 17 times speed-up if running on the Utopia PCs.

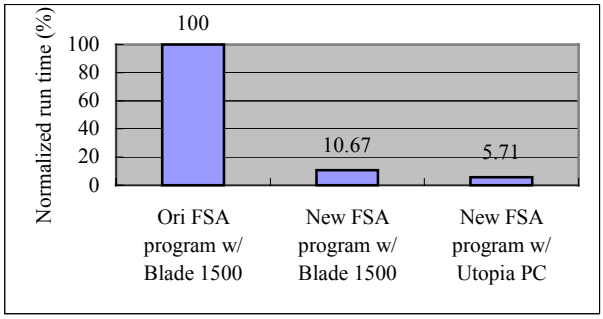

Fig. 3: Comparison of normalized run time

At first the difference of the two systems on identifying the failure shapes on a die is shown in Fig. 4. Obviously they obtained almost the same counting results, most of which are P-row and 1-bit failures. As given by M-file, the counting die percentages of each type of failure shapes on a wafer are compared in Fig. 5. There is still very little difference between the two systems. Fig. 6 provides the resultant average numbers of failures occurred in those corresponding dies, which are obtained by the D-file. There're considerable differences between the

two systems, e.g. for the failures of P-col, Cross, SA, and P-row. The results seem to be useless, but the distribution graph can initially help judging the main wafer failures. The faithful analysis can still be obtained by further referring to the exact numbers given in W-file.

4 Conclusion and Future Work

To speed up the process of failure shape analysis for memory repair, this study develops new programs to analyze the FBM data to achieve similar defect distribution graph as that of the original system. The obtained distribution results are also feasible for the yield enhancement system to diagnose the defect causes, but only need about tenth of the original waiting time. The developed programs count up the numbers of failure shapes in specific order, which help attain similar results as the original system except for the average failure shapes for those occurring dies on a wafer. This will be further revised by not only analyzing the counting numbers of the failure shapes, but also their respective locations in our future work.

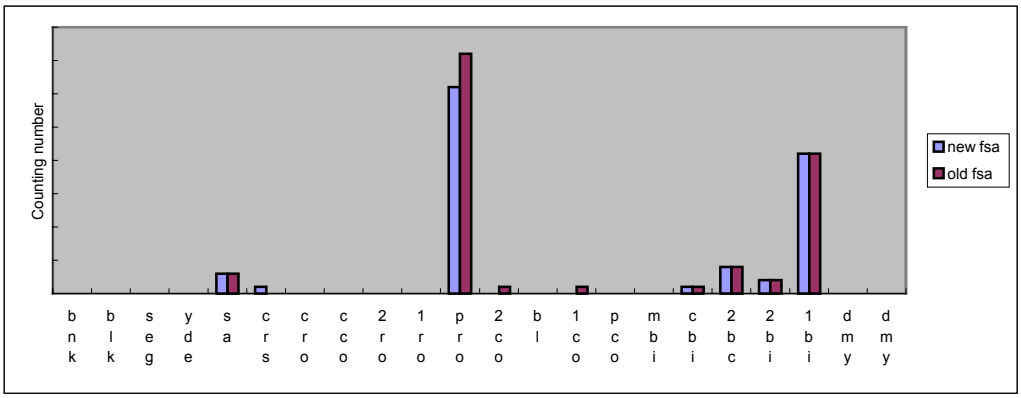

Fig. 4: Comparison of failure shapes counting on a die

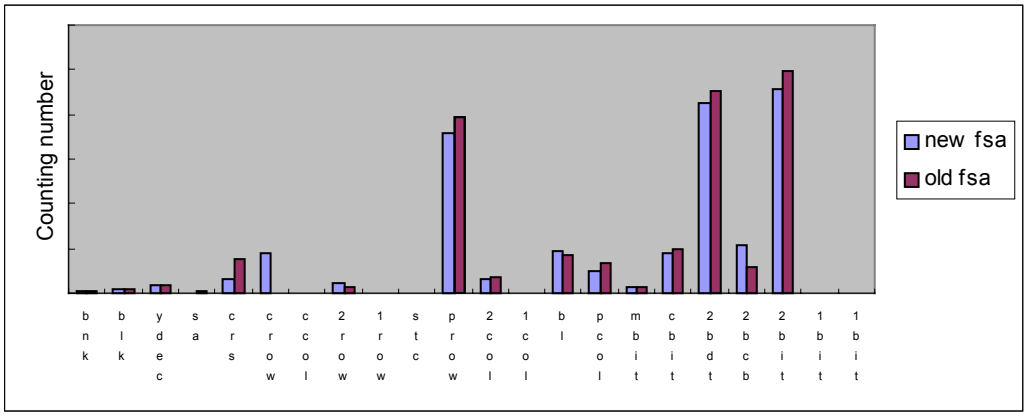

Fig. 5: Comparison of die counting for each failure shape on a wafer

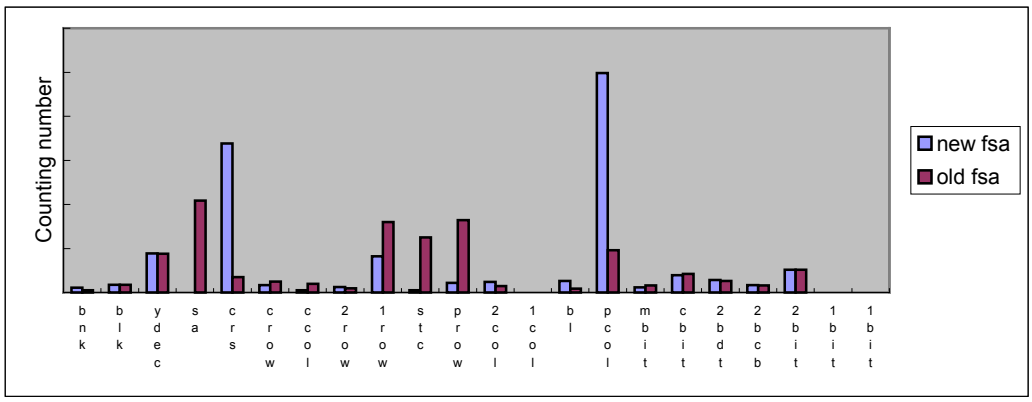

Fig. 6: Average counts of failure shapes for the occurring dies on a wafer

References:

[1] W. Malzfeldt, W. Mohr, H.-D. Oberle, and K. Kodalle, "Fast automatic failbit analysis for DRAMS," *Proc. IEEE Intn'l Test Conference*, paper 20.1, pp.431-438, 1989.

[2] Y. Sakai, J. Sawada, W. Sakamoto, J. Murata, H. Kawamoto, K. Sakai, and K. Nakamuta, "A wafer scale fail bit analysis system for VLSI memory yield improvement," *Proc. IEEE Intn'l Conference on Microelectronic Test Structures*, Vol. 3, pp.175-178, 1990.

[3] S. Ishikawa, K. Ishihara, Y. Miyamoto, I. Miyazaki, T. Ohshima, and M. Sato, "Fail bit analysis system for semiconductor memory wafers," *Proc. Japan Intn'l Electronics Manufacturing Technology Symposium*, pp.287-290, 1993.

[4] T. Hamada and M. Sugimoto, "Application of a bitmap analysis system to the forefront of DRAM devices development," *Proc. IEEE/SEMl Advanced Semiconductor Manufacturing Conference*. pp.222-227, 1997.

[5] J. Vollrath, U. Lederer, and T. Hladschik, "Compressed bit fail maps for memory fail pattern classification," *Proc. IEEE European Test Workshop*, pp.125-130, 2000.