

LETTER

Automatic Defect Classification System in Semiconductors EDS Test Based on System Entity Structure Methodology

Young-Shin HAN^{†a)}, SoYoung KIM^{††b)}, Members, TaeKyu KIM^{†††c)}, and Jason J. JUNG^{††††d)}, Nonmembers

SUMMARY We exploit a structural knowledge representation scheme called System Entity Structure (SES) methodology to represent and manage camera wafer failure patterns which can make a significant influence to FABs in the semiconductor industry. It is important for the engineers to simulate various system verification processes by using predefined system entities (e.g., decomposition, taxonomy, and coupling relationships of a system) contained in the SES. For better computational performance, given a certain failure pattern, a Pruned SES (PES) can be extracted by selecting the only relevant system entities from the SES. Therefore, the SES-based simulation system allows the engineers to efficiently evaluate and monitor semiconductor data by *i)* analyzing failures to find out the corresponding causes and *ii)* managing historical data related to such failures.

key words: semiconductor, system entity structure, electrical die sorting, fail bit map data, pruning

1. Introduction

The yield of Electrical Die Sorting (EDS) wafer test is the most important standard in measuring the productivity of a FAB. The main goal of this fabrication process is the earliest achievement of adequate FAB yield for new products and maintenance of high yield by minimizing defects. The defects can be generated in a certain wafer bin, which is a basic element to determine whether the wafer fails or not. Thus, to secure and improve stable yield, most engineers have studied how to efficiently detect the defects, as well as how to analyze the causes of the defects. In addition, a team of experts can be set up to detect more accurately defects and verify sophisticated defect causes.

However, one of the main problems with verifying defect types is labor consumption, because the verification process is required as a prerequisite for analyzing the defects. Not only many engineers classify and summarize types of defects manually after verifying wafer maps that occur every day. Many yield related professionals are charged with verifying defect types of every device. Even though defect verification is a manual process, it is still difficult to

efficiently decide the types and causes of defects. Dramatic increase in fabricated memory density tightens the design rule and complicates the cell structure, which in turn gives rise to design margins causing a wide range of failures [1].

The failure cannot be used as an efficient resource in analyzing the yield as characters of each failed device are defined and arranged whenever they are needed [2]. If engineers automate a series of processes that enables detecting the cause of defect occurring in FABs by defining each character of failed device and classifying wafers with each type of failure, they will be able to analyze the defects more effectively as well as improve the yield and the quality of the production process.

To solve these problems, we propose a novel defect classification scheme based on the System Entity Structure (SES) method. The SES can be applied to minimize the error rate by defining defects in each type of device, after we have gathered the data from the semiconductor fabrication system.

Figure 1 shows the system architecture based on the SES method. Context-aware modeling and simulation for network behavior analysis can be automated. We note that the overall process consists of the following four steps:

1. Capture fail bit map data [3] for generating a pruned SES (PES),
2. Create new ontology according to user requests,
3. Map from PESs of captured data to newly generated SES for resulting PESs out, and
4. Model and simulate using PESs generated in the previous step.

For this paper, we have built several models and run simulations. Creating new ontology in SES format is invoked first from the Selector model. The Selector model

Manuscript received October 30, 2009.

Manuscript revised March 2, 2010.

[†]The author was with Yeungnam Univ. to conduct this study, and currently with Sungkyunkwan University, Korea. Corresponding author.

^{††}The author is with Sungkyunkwan Univ., Korea.

^{†††}The author is with Security Management Institute, Korea.

^{††††}The author is with Yeungnam University, Gyeongsan, Korea.

a) E-mail: yshan95@ewhain.net

b) E-mail: ksyoun@skku.edu

c) E-mail: taekyu@gmail.com

d) E-mail: j2jung@intelligent.pe.kr

DOI: 10.1587/transinf.E93.D.2001

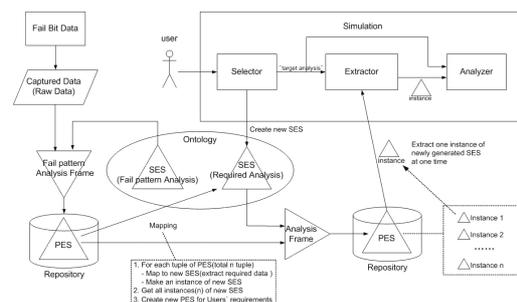


Fig. 1 SES-based system for automating modeling and simulation.

gets users inquiries. The Selector model first starts creating new SES and then allows the other two models, the Extractor model and the Analyzer model prepare for target analyses. The Extractor model retrieves source data from PESs produced by mapping operations in step three. The data are instance objects. Each instance depicts one tuple of user defined SES and every instance includes data values. The extracted instance objects are transmitted to the Analyzer model. Finally, the Analyzer model obtains necessary attributes from the received object instances and exhibits evaluation results.

This paper is organized as follows. Section 2 describes the computational representation of SES and the pruned entity structure (PES). The analysis process of defect types is described in Sect. 3. In Sect. 4, we explain the simulation results. Finally, Sect. 5 draws the conclusion of this work.

2. System Entity Structure

2.1 Data Engineering

Data engineering becomes increasingly important with the popularity of Service Oriented Architecture (SOA) and web services. Such data engineering requires a design methodology within an ontology framework.

In this paper, the SES serves as an abstract ontology framework for world state descriptions particularly involving dynamics in space and time. It is implemented within a software tool, the SESBuilder, employing the extended markup language XML. PES represents the logically possible set of world state descriptions consistent with those of SES. At the implementation level, an SES is represented by a schema or Document Type Definition (DTD) whose instance documents represent the possible prunings. The SESBuilder, through natural language and graphical interfaces, supports convenient specification of SESs, pruning to create PESs, and transformation of the created PESs to XML representations. Thus, the software can provide a data engineering work space.

2.2 System Entity Structure (SES)

The basic idea of SES is that a system entity represents the real system enclosed within a certain choice of system boundary. Figure 2 shows a simple view of entities and their relationship in a SES. Key components comprise the SES [4]:

- Entity: An entity is intended to represent a real world object which either can be independently identified or can be postulated as a component in some decomposition or a real world object.
- Aspect: An aspect represents the decomposition of many possible of entities. The children of an aspect are entities representing components in a decomposition of its parents.
- Specialization: A specialization is a mode of classifying entities and is used to express alternative choices

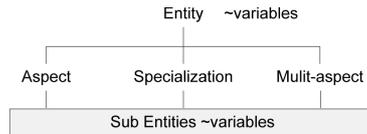


Fig. 2 Overview of SES items and relationships.

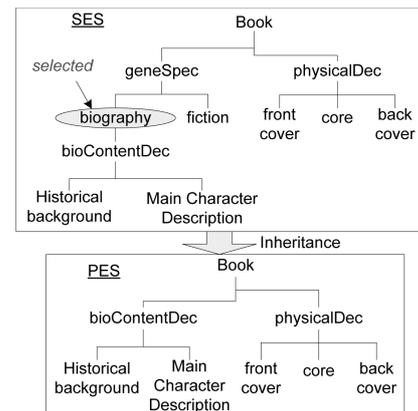


Fig. 3 SES and PES inheritance.

for components of the system being modeled. The children of a specialization are entities representing variants of its parents.

- Multi-Aspect: A multi-aspect is an aspect of one kind of components.
- Variable: A variable is a slot attached to an entity that can be assigned a value from a given range set. It denotes a property, quality, or attribute of an entity to which it is attached.

2.3 Pruned Entity Structure (PES) and Its Inheritance Representation

The process of pruning the SES is to construct a desired entity structure to meet particular application objectives. More specifically, a specialization may have several entities to select from that represent different ways to specialize an entity. Ultimately, in a completely pruned entity structure, every specialization has exactly one entity. In other words, the process of pruning is to reduce the SES by making selections in all its specializations, allowing multi-aspect expansions.

PES inheritance is defined so that the parent and any child of a specialization combine their individual variables, aspects and remaining specializations when pruning is activated. Figure 3 illustrates how the inheritance is processed in the case of multiple decompositions.

The SES framework is especially applicable to simulation modeling of dynamical systems. Comparison with other ontology frameworks is presented in [3].

3. Analyzing Process of Defect Types

Defect types can be classified into chip level and wafer level.

Table 1 Chip level defect type classification.

Chip Category	Chip Type	Example of chip type	Fail Type
Random	Random		A
	Cluster		B, C
	Col/Row		D
Block	Partial Block		E
	Block		F
Spatial	NWLE (row)		G
	CSL (col)		H

Engineers can define various defects in the chip level by classifying them according to the specific defect type of the cell contained in each chip. Chips can be categorized into two groups of wafer bins, according to their test results by ICC failure and function failure. In a function test, engineers carry out a wide range of tests on chips. Although engineers can detect failure in chips through these tests, they need another classification according to each failure type, as the cause of defect can change from the influence of the failed block. Defects in wafers can be classified according to its distribution type of defective chips on a wafer, and the cause of defect can be changed by area character [5], [6].

3.1 Definition of Chip Level Defect Types

In type classification, chip levels should have a priority over that of the wafer level. To categorize the types of defects according to the unit block of a chip, this paper uses failed characteristics of the device to define the defects with the similar cause. They are defined based on the type and shape of a failed test within a unit block of a chip. Table 1 shows three types of defects which are random, block, and spatial. According to the distribution density, the random type is divided into three types; random, cluster type, and column/row. Block type can be categorized into partial block and block. Finally, according to the distribution characteristics of vertical and horizontal failures, the spatial type can be divided into Column Select Line (CSL) and New Word Line Enable (NWLE). Thus, this classification has been employed to represent the hierarchical SES tree[†] of semiconductor.

4. Simulation Results

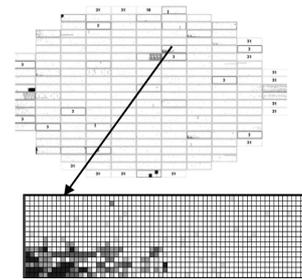
As a result of EDS wafer test, fail bit map and measure data files are generated. Based on those data, we can classify the failure types. Fail bit map data generated from manufactured devices can not be analyzed by storing the bit test results from wafer test. Hence, we store the test results by cell unit blocks.

Each unit block has a grade level determined by the fail bit map, as shown in Table 2. From the grade level distribution over the chip, the defect type can be determined. Data is recorded following the process shown in Fig. 4.

- Failed Bit Map: comprised of many chips
- Chip: comprised of $X \times Y$ unit block

Table 2 Grade of unit block.

Grade	Bit Count	Fail type
0	0	Good
1	1	A
2	10	B
3	20	C
4	40	D
5	160	E
6	25600	F
7	1024000	G
8	90000000	H

**Fig. 4** Example of input fail bit map and chip unit block.**Table 3** Result of chip defect type analysis.

Defect types	Sample number	Accuracy
Chip-level analysis	600 chips among 5200 chips in 4 wafer bins	41 chip error classification ($\frac{600-41}{600} = 0.917$)
Function-level analysis	1200 chips among 5200 chips in 4 wafer bins	95 chip error classification ($\frac{1200-95}{1200} = 0.921$)

- Unit Block: quantized grade

Chip level information, such as wafer ID, total chip number, x-y coordination, Bin value, chip DC, function test results and the grade of each chip's unit block, is used as data for wafer information of a failed bit map.

4.1 Experimental Results

The experiment environment^{††} is as follows.

- Subject Device: ★★MB DRAM
- Subject Wafer: 26 WF of a daily test quantity ★★Lot★★WF
- Application Method: Automated analysis by program generated algorithms
- Testing environment: OS Window XP, Compiler VC++ 6.0, wafer per 2.5 sec

As shown in Table 3, we have measured the accuracy of the defect classification. Given two sampled chips, more than 90% of the subjects (91.7% and 92.1%, respectively) were precisely classified.

[†]The SES tree is available from <http://ke.yu.ac.kr/IEICE.pdf>

^{††}★ indicates confidential information.

In the case of mixed types, error rate was high and improvements in flexibility, extension rate, and false analysis rate were required. This paper has proposed ways to automatically detect and classify types of defective chips used in yield analysis by using failed bit map data occurring in the EDS test of semiconductor fabrication. By applying this method to analyze defects, we automated conventional manual operation and built a system effective in analyzing defects.

5. Conclusion

It has been a difficult and expensive task to manually monitor the wafer quality and detect the defects from the failed wafers for yield control. By using the SES-based method proposed in this paper, automatic analysis and summary of testing results has been designed and implemented in the real semiconductor process. Specifically, the structured management of defect patterns has been a key role of this process, and it is sustainable to the new patterns over time. By constructing the SES-based simulation system from defect type definition and analysis, efficient failure type analysis has improved in terms of yield and quality. As a result, this integrated system helps engineers to easily evaluate and monitor semiconductor data for analyzing reasons of failure and managing additional failures.

Acknowledgments

The authors gratefully acknowledge the anonymous review-

ers for their helpful suggestions and comments. This work is partially supported by post-doctor support program by Yeungnam university, Korea and Samsung Research Fund, Sungkyunkwan University, 2009.

References

- [1] Y. Han and C. Lee, "Automatic classification using decision tree and support vector machine," Proc. 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2005), ed. R. Khosla, R.J. Howlett, and L.C. Jain, Lect. Notes Comput. Sci., vol.3682, pp.1325-1330, Springer, Melbourne, Australia, Sept. 2005.
- [2] F.L. Chen and L.S.-F, "A neural-network approach to recognize defect spatial pattern in semiconductor fabrication," IEEE Trans. Semicond. Manuf., vol.13, no.3, pp.366-373, 2000.
- [3] I. Atsushi, N. Koji, and F. Hiroshi, "Fail pattern classification and analysis system of memory fail bit maps," IEICE Technical Report, ICD99-221, 1999.
- [4] S. Cheon and B.P. Zeigler, "Experimental frame structuring and aggregation of source data: Application to us climate normals," Proc. Spring Simulation Multiconference 2007 (SpringSim'07), Society for Computer Simulation International, pp.243-248, Norfolk, Virginia, 2007.
- [5] T. Fukushima, H. Narazaki, and M. Konishi, "A method of feature extraction from an image for quality analysis," Proc. 1999 IEEE International Conference on Systems, Man, and Cybernetics (SMC '99), pp.952-957, IEEE Computer Society, Tokyo, Japan, 1999.
- [6] C. Ortega, "Human based knowledge for the probe failure pattern classification with the use of a backpropagation neural network. application on submicron linear technologies," Proc. IEEE/SEMI Advanced Semiconductor Manufacturing Conference and Workshop, 1998.