Framework for Fault Analysis and Test Generation in DRAMs

Zaid Al-Ars[1,2,3] Said Hamdioui[2] Georg Mueller[3] Ad J. van de Goor[2]

[1]CatRam Solutions
Reinier de Graafweg 188
2625 DE Delft, The Netherlands

[2]Delft University of Technology
Laboratory of Computer Engineering
Mekelweg 4, 2628 CD Delft, The Netherlands

[3]Infineon Technologies AG
Department of Product Engineering
Balanstr. 73, 81541 Munich, Germany

E-mail: zaid.al-ars@catram.com

Abstract: *With the increasing complexity of memory behavior, attempts are being made to come up with a methodical approach that employs electrical simulation to tackle the memory test problem. This paper describes a framework of algorithms and tools developed jointly by the Delft University of Technology and Infineon Technologies to systematically generate DRAM tests using Spice simulation. The proposed Spice-based test approach enjoys the advantage of being relatively inexpensive, yet highly accurate in describing the desired memory faulty behavior.*

Keywords: *tool framework, DRAM testing, faulty behavior, defect simulation, test generation.*

1 Introduction

The process of DRAM testing has developed into a rather complex combination of historically proven test sets, highly specialized expert knowledge, and cutting-edge *failure analysis (FA)* methods to evaluate any newly encountered failure mechanisms [Nakamae03]. However, many of these test development methods follow two main paths, 1. test development based on memory specifications, and 2. test development based on post-manufacture failure analysis. This paper presents a third industrial alternative for test development, which is based on the application of Spice simulation to efficiently perform fault analysis and test generation. Simulation-based test generation serves as a tradeoff between the qualitative and inexpensive specifications-based testing, and the quantitative yet expensive manufacturing-based testing.

Section 2 starts with a global discussion of the current approach to DRAM test generation. Section 3 shows the way electrical simulation can support test generation activities. Section 4 discusses the suggested framework to implement the simulation-based fault analysis approach.

2 Traditional DRAM test flow

The memory *test flow* is a description of the stages and activities needed to test memory devices. Figure 1 shows a block diagram of a typical manufacturing test flow for a memory manufacturer [Nakamae03]. The figure also shows a three-stage representation of the design flow, starting with the specification of a new memory technology, followed by the design process and ending with chip manufacturing. Within the test flow, the actual testing process takes place in only two blocks in the figure

- *frontend* (or wafer level) testing, and
- *backend* (or component level) testing,

drawn as two large rectangles in the figure. Frontend testing is performed before chip packaging, to prevent packaging defective devices. Backend testing ensures that packaged chips function properly and are ready to be delivered to the customer.

Traditionally, memory tests performed in the test flow are generated in two steps: 1. based on information from the specification stage of the design flow and, 2. based on information from tests performed on manufactured chips in the manufacturing stage of the design flow [see the two highlighted stages of the design flow in Figure 1]. These two stages are described next.

1. Specification stage—Every test flow for a new memory design starts with an analysis of the specifications of the new memory design and an analysis of the tests used for the previous memory design [see block "Tests from old design"]. The specifications are used to generate a new set of tests by adapting the old tests to comply with the specifications of the new memory [see block "Test generation"], an approach referred to as *specifications-based* test generation. This approach is relatively fast and inexpensive, since it assumes that the faulty behavior of the memory does not change with changing memory technology, which is generally not true. Therefore, this approach is not memory-specific enough to derive accurate tests for the memory.

2. Manufacturing stage—After chip manufacturing, specifications-based tests, generated in the previous step, are applied to the memory in the frontend and backend stages of the test flow. For every new memory, the "Yield analysis" stage identifies new failure mechanisms for which additional tests are needed. Furthermore, memories that fail in the field are sent back to the manufacturer as *application fails*, where they get analyzed in the "Failure analysis" stage of the test flow. Feedback from these two analysis stages is used to adapt the tests in the "Test adaptation" stage, in order to detect any previously undetected faults. This test generation and adaptation approach is

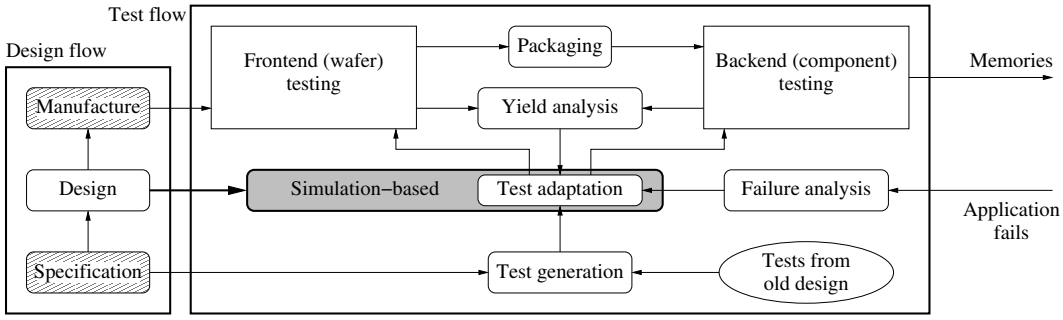

Figure 1. Block diagram of a manufacturing test flow for a memory manufacturer.

called *manufacturing-based* test generation, which is very accurate and memory-specific, since the tests are generated by statistically analyzing test feedback data from real chips in the fab. This style of test generation assumes a large volume of failing memory parts to enable a meaningful statistical analysis of test data which, in turn, requires a rather expensive and time-consuming *test adaptation loop* (such as Frontend → Yield analysis → Test adaptation in Figure 1) until a stable set of tests can be generated.

3 Including electrical simulation

The specifications and manufacturing-based test generation approaches, each has advantages and disadvantages in terms of quality and cost of test generation, as shown in Figure 2.

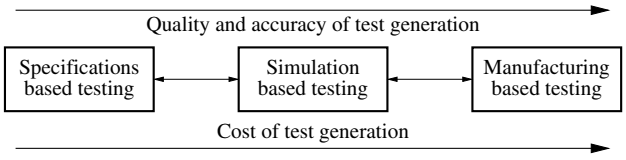

Figure 2. Cost-quality tradeoff of test generation.

This paper proposes a *simulation-based* test generation approach, that strikes a tradeoff between specifications and manufacturing-based test generation, and provides an alternative that is both moderately cheap and device-specific. As shown in the shaded "Simulation-based" block of Figure 1, simulation-based test generation uses electrical Spice models from the design stage of the design flow, to represent the internal memory design and behavior, as well as the characteristics of the fabrication process. This provides a fairly accurate representation of the specific behavior of the memory under analysis. At the same time, simulations can support test adaptation activities based on feedback data from the yield analysis and failure analysis stages. This significantly accelerates the expensive and time-consuming feedback loop required by the manufacturing-based test generation.

4 Simulation-based test generation

Figure 3 suggests a framework to implement the simulation-based test generation approach into the test flow, where a number of steps and tools are identified that would generate the required tests based on an initial Spice simulation model from the designers.

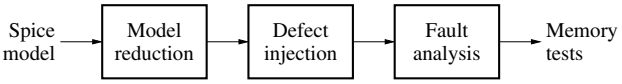

Figure 3. Framework to implement simulation-based test generation.

Model reduction—The framework starts with an electrical Spice model that describes the various circuits of the memory, and makes it possible to simulate their behavior using an electrical simulator. This model is fed into a "Model reduction" tool that generates a reduced simulation model, which helps to reduce the analysis time.

Defect injection—In order to perform the fault analysis on the reduced model, the failure mechanism to be analyzed needs to be modeled and injected into the simulation model. The failure mechanism is modeled using a Spice defect (resistive, parameters change, etc.) and injected using the "Defect injection" tool in Figure 3.

Fault analysis—The fault analysis itself is performed in the "Fault analysis" tool of Figure 3 which generates optimized tests using the reduced, defective simulation model. The algorithms included within this tool depend on the memory to be analyzed and on the type of the required analysis. A number of algorithms have been suggested in the literature to tackle many different problems, such as those for simulation-based test pattern generation in DRAMs [Al-Ars02].

References

[Al-Ars02] Z. Al-Ars and A.J. van de Goor, "Approximating Infinite Dynamic Behavior for DRAM Cell Defects," *in Proc. IEEE VLSI Test Symp.*, 2002, pp. 401–406.

[Nakamae03] K. Nakamae, H. Ikeda and H. Fujioka, "Evaluation of Final Test Process in 64-Mbit DRAM Manufacturing System Through Simulation Analysis," *in Advanced Semiconductor Manufacturing Conf. and Workshop*, 2003, pp 202–207.