A Signature-based Approach for Diagnosis of Dynamic Faults in SRAMs

Alexandre Ney, Alberto Bosio, Luigi Dilillo, Patrick Girard, Serge Pravossoudovitch, Arnaud Virazel

A Signature-based Approach for Diagnosis of Dynamic Faults in SRAMs*

A. Ney A. Bosio L. Dilillo P. Girard S. Pravossoudovitch A. Virazel

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier – LIRMM
Université de Montpellier II / CNRS
161, rue Ada – 34392 Montpellier Cedex 5, France
Email: <lastname>@lirmm.fr URL: http://www.lirmm.fr/~w3mic

Abstract

This paper focuses on diagnosis of dynamic faults in SRAMs. The current techniques for fault diagnosis are mainly based on the signature method. Here, we introduce an extension of the signature scheme by taking in account additional information related to the addressing order during March test execution. A first advantage of the proposed approach is its capability to distinguish between static and dynamic faults. Another main feature is the correct identification of the location of the failure in a given memory component: the core-cell array, write drivers, sense amplifiers, address decoders and pre-charge circuits. Moreover, since this approach does not modify the March test, there is no increase of test complexity, conversely to other existing diagnosis techniques.

1 Introduction

Fault detection, diagnosis and location are widely used to repair manufacturing defects in Static Random Access Memories (SRAMs) and improve the product quality, reliability and yield [1]. Traditional techniques for memory repair consider identification of fault type and fault location as two separate phases. For repair purposes, the identification of the exact location of the detected fault is more important than the information on the type of fault. In particular, mapping the faults on the core-cell array allows the correct use of the spare columns and rows. Conversely, diagnosis approaches targeting yield ramp up require the identification of the cause of the failure as well as its location.

As soon as the application of March algorithms has revealed logic errors in a given memory, diagnosis can be performed. Generally, diagnosis approaches are based on a signature (syndrome) method as for logic design diagnosis [2]. The diagnosability ratio (DR) is used to measure the quality of a diagnosis algorithm and represents the ratio between the number of distinguishable fault types and the number of total detectable fault types. Diagnosis methods generally use a fault dictionary and try to achieve the highest diagnosability ratio for a given test algorithm [3, 4, 5, 6].

Most of the existing solutions target only the diagnosis of static faults. In this paper we introduce an extension of the signature technique, by adding new fields in the syndrome, to make it able to deal with dynamic faults as well. This extension is made possible by using information on the addressing sequence during the March test execution. The addressing order information has been demonstrated to be important in the detection of dynamic faults in SRAMs as well as the data background [7, 8] that we intend to consider in the development of this study. The proposed approach allows to diagnose dynamic faults, to distinguish between static and dynamic faults, and to localize the related failure in the memory. The additional information introduced in the signature is taken from the algorithm itself, thus it does not increase its complexity. In the paper, we illustrate the proposed signature-based diagnosis approach by considering as case study the dynamic fault Un-Restored Write fault (URWF), affecting write driver and pre-charge circuit of SRAMs.

The rest of the paper is organized as follows. In Section 2, we describe the classical signature-based diagnosis. In

* This work has been partially funded by the French government under the framework of the MEDEA+ 2A702 "NanoTEST" European program .

978-1-4244-1577-9/08/$25.00 ©2008 IEEE

Section 3, we present the proposed extension of signature-based diagnosis to deal with dynamic faults. Analysis and developments of the proposed study are given in Section 4. Concluding remarks are in Section 5.

2 Signature-based diagnosis principle

A classical diagnosis technique for memories is based on signature. The signature, also called syndrome, is composed of a set of read operations included in the considered March test (MT). Each signature represents a set of possible fault models affecting the memory. The quality of a diagnosis, named Diagnosability Ratio (DR), is defined as the ratio between the number of distinguishable fault types and the number of total detectable fault types. Since the fault dictionary is based on a given MT, the DR is strictly related to:

- the set of fault models covered by the MT itself
- the number of read operations included in the MT

To illustrate the signature-based diagnosis principle, let us consider the well-known March C-, whose structure is shown in Figure 1.

$$\{\Updownarrow(w0)\ \Uparrow(r0,w1)\ \Uparrow(r1,w0)\ \Downarrow(r0,w1)\ \Downarrow(r1,w0)\ \Updownarrow(r0)\}$$
$$M_0 \quad M_1 \quad M_2 \quad M_3 \quad M_4 \quad M_5$$

Figure 1: March C- structure

For a given test algorithm, the fault dictionary can be generated by listing the fault models and their corresponding syndromes. For example, the fault dictionary, concerning only stuck-at (SAF) and transition (TF) faults, for March C- is given in Table 1 [6].

R_0	R_1	R_2	R_3	R_4	Fault model
0	1	0	1	0	SAF0
1	0	1	0	1	SAF1
0	1	0	1	0	TF1
0	0	1	0	1	TF0

Table 1: Fault dictionary for the March C- algorithm

In Table 1, $R_i = 0$ (1) means that the i^{th} read operation of the test algorithm has returned a correct (faulty) value. For example, a stuck-at-0 fault (SAF0) is detected by all r1 operations. Consequently, the March syndrome for SAF0 is (01010), see Table 1. Note that this fault dictionary can be extended to the whole set of fault models detected by March C-. Based on these principles, a lot of works have been proposed in the literature and target the classical case of static faults such as stuck-at, transition and coupling faults.

Static faults require maximum one operation (read or write) to be sensitized [1, 9], while dynamic faults require more than one operation to be sensitized [9, 10 and 11]. In order to target dynamic faults, new March tests have been proposed, such as March RAW [12], March AB [13], as well as modifications of existing March tests, such as March iC- [7]. To the best of our knowledge, dynamic faults have been considered for diagnosis purpose in the literature only in [14], where the authors focus on dynamic coupling faults and extend the syndrome using the written data as field. Static faults have been more widely addressed [15, 16, 17]. These works propose the extension of the considered MT, by the addition of extra read operations, in order to increase the signature fields and therefore improve the DR. The first drawback of such techniques is that the increased complexity of MT, *e.g.* 46n in [16], can be excessive to be used for industrial purpose. In addition, these solutions are most of the time unable to distinguish between all faults (or all fault models) and hence do not allow to determine which element of the memory is defective. In fact, a fault model may be related to more than one defect in the various elements of the memory. For example, it has been shown in [18, 19] that a transition fault can be due to defects locating in a core-cell or in a write driver. In this case, these diagnosis solutions will indicate that the memory is affected by a transition fault without any information on which actually is the faulty element of the memory.

Consequently, there is a clear need of new diagnosis solutions that consider both static and dynamic faults, and that are able to provide accurate location of the fault (in the core-cell array, write drivers, sense amplifier, address decoders, pre-charge circuits, etc). Such diagnostic information may save a significant amount of time during the ramp up phase.

3 Signature-based dynamic fault diagnosis

This paper addresses the problem of diagnosis of both static and dynamic faults. Moreover, the presented approach allows the identification of the defective memory elements. This approach is still based on the classic signature methodology, but it reaches a high DR without raising the MT complexity, *i.e.* without adding additional read operations in the MT. For this purpose, we expand the number of the signature fields by adding information related to the address sequence used during the MT execution. Considering all the fields in the signature, the information required for the diagnosis is the following:

- The tester report: the addresses of the core-cells where the read operation has returned a faulty logic value;
- The executed MT;
- The list of fault models covered by the executed MT;
- The addressing sequence adopted during the MT execution;
- The SRAM architecture providing information about the core-cell array structure from the logic point of view.

In order to achieve a more efficient diagnosis, we now

improve the signature-based diagnosis by introducing information extracted from the addressing order adopted during the MT execution and the SRAM architecture. For an easy understanding of our proposition, we consider the dynamic fault called Un-Restored Write Fault (URWF) as case study. The definition of such fault is given below. The URWF can be due to different electric causes such as resistive-open defects in the write driver [20] or in the pre-charge circuit [21] of SRAMs. In the first case, the write driver is turned off too late (or even not turned off in case of a high resistive-open defect), inducing a wrong level on bit lines at the end of the pre-charge phase. In the second case, the pre-charge circuit itself is not strong enough to fully charge bit lines. The common effect in both cases is that the final voltage level of bit lines is erroneous at the end of the pre-charge phase, and the following read operation fails.

Figure 2 depicts a simplified scheme of an SRAM composed by two 4x4 blocks. Each block presents its own I/O circuitry that is shared by four columns. As presented in [20], the URWF resulting from a resistive-open defect in the write driver requires the following sequence of operations to be detected:

$$wx_a\ r\overline{x}_b \qquad \text{(Eq. 1)}$$

where ' wx_a ' means write the value x in cell *a* and ' $r\overline{x}_b$ ' means read the opposite value in cell *b*. Both operations have to be performed on two distinct core-cells that belong to the same I/O circuitry (see Figure 1).

When an URWF is due to a resistive-open defect in the pre-charge circuit, the sensitization sequence is the same than that of Eq. 1, but in this case, both operations (wx_a and $r\overline{x}_b$) have to be performed on two distinct core-cells belonging to the same column [21].

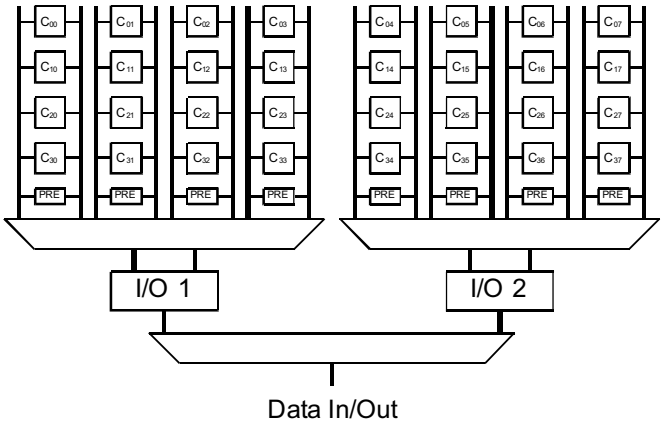

Figure 2: A two blocks SRAM architecture

Among March test algorithms, March C- is able to detect both types of URWF [20, 21] if it is executed with the specific addressing order "column after column". The degrees of freedom of March tests allow this particular addressing order without modifying the coverage of the other targeted faults of March C- (coupling faults, transition faults, stuck at faults) [23].

Considering the memory architecture shown in Figure 2, we can determine the whole set of possible addressing situations:

- Ad_i is the address of the currently accessed cell and during a read at this address, a fault is detected.
- Ad_{i-1} is the address of the cell previously accessed w.r.t. Ad_i
- Ad_{i+1} is the address of the next cell to be accessed w.r.t. Ad_i .

Considering three consecutive address locations (Ad_{i-1}, Ad_i and Ad_{i+1}) during test execution, the possible combinations are the following ones:

- Ad_{i-1} belongs (or not) to the same I/O circuitry w.r.t. Ad_i.
- Ad_{i-1} belongs (or not) to the same column w.r.t. Ad_i.
- Ad_{i+1} belongs (or not) to the same I/O circuitry w.r.t. Ad_i.
- Ad_{i+1} belongs (or not) to the same column w.r.t. Ad_i.

Combinations described above are summarized in Figure 3. On the left side, the list of the different addressing configurations concerning the previous accessed cell Ad_{i-1} is presented. The next accessed core-cell Ad_{i+1} is considered on the right part of the scheme.

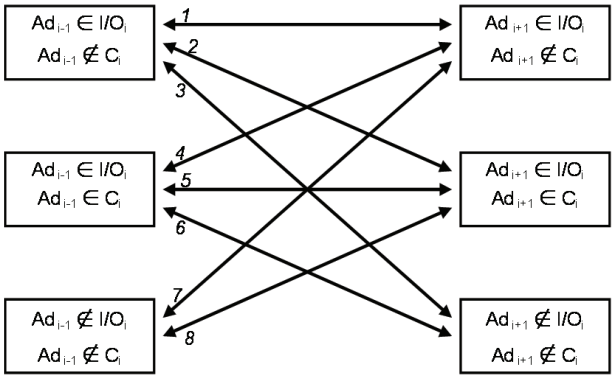

Figure 3: Possible address sequence during test execution for URWF detection, considering the memory architecture

For URWF detection, the sensitization sequence has to be applied at least on two distinct core-cells belonging to the same I/O circuitry (in case of write driver failure) or to the same column (in case of a faulty pre-charge circuit). Moreover, we consider not only the previous accessed core-cell but also the next accessed core-cell because most of March algorithms have up (↑) and down (↓) addressing order. Consequently, the next accessed core-cell in the up (↑) addressing order is the previous accessed core-cell during the down (↓) addressing order, and vice versa.

An URWF can be sensitized when two core-cells are

accessed with any addressing order described in Figure 3. The knowledge of the addressing sequence allows to deduce the following important information:

i. It allows to determine the faulty memory element, *i.e.* pre-charge circuit or write driver.
ii. It allows to determine the nature of the observed fault, *i.e.* static or dynamic.

Let us consider the first point (i). In accordance with the possible addressing configurations presented in Figure 3, four cases are possible:

1. The configuration allows detecting URWFs caused by malfunction in a write driver.
2. The configuration allows detecting URWFs caused by malfunction in a pre-charge circuit.
3. The configuration allows detecting URWFs caused by malfunction of both pre-charge circuit and write driver but it cannot provide the failure location.
4. The configuration allows detecting URWFs caused by malfunction in both pre-charge circuit and write driver but also provides the failure location.

In Table 2, we list all possible *extended signatures* obtained with the application of the March C- algorithm (c.f. in Figure 1) and leading to URWFs detection.

Fault model	Faulty element	CONF	R_0	R_1	R_2	R_3	R_4	Ad_{i-1}	Ad_{i+1}
URWF	WD	1	1	0	1	0	0	10	10
	WD	2a	1	0	1	0	0	10	11
	Pre	2b	0	0	1	0	0	10	11
	WD	3	1	0	0	0	0	10	00
	WD	4a	1	0	1	0	0	11	10
	Pre	4b	1	0	0	0	0	11	10
	WD, Pre	5	1	0	1	0	0	11	11
	WD, Pre	6	1	0	0	0	0	11	00
	WD	7	0	0	1	0	0	00	10
	WD, Pre	8	0	0	1	0	0	00	11

Table 2: List of *extended signatures* for URWF detection during March C- execution

The two first columns provide the fault model (URWF) and the memory element(s) whose failure involves the URWF: *WD* for write driver and *Pre* for pre-charge circuit. The third column gives the configuration, with respect to the scheme in Figure 3 (labels on arrows). Columns from four to eight provide the classical March C- signatures for URWF detection. Finally, the two last columns of Table 2 are fields we have added to represent the addressing order, Ad_{i-1} for the previous accessed core-cell and Ad_{i+1} for the next one. These additional fields require two bits to represent all address configurations presented in Figure 3:

- The first bit indicates if address Ad_{i-1} (or Ad_{i+1}) shares the same I/O element than the current accessed memory core-cell ('1' if yes, '0' if no, x if don't care).
- The second bit indicates if address Ad_{i-1} (or Ad_{i+1}) shares the same column than the current accessed memory core-cell ('1' if yes, '0' if no, x if don't care).

For example, Ad_{i-1} = 10 means that the previous accessed core-cell shares only the same I/O circuitry with the current accessed core-cell. For the diagnosis of other dynamic fault models, it is necessary to use additional bits to describe other specific addressing sequence. For example, Ad_{i-1} and Ad_i belonging to the same word line is the addressing sequence useful to detect dynamic Read Destructive Faults (dRDF)[24], unitary hamming distance addresses are necessary to detect Address Decoder Open Faults (ADOF) [7].

For a better understanding of Table 2, we propose the reading of the first line. The last two columns indicate that Ad_{i-1} and Ad_{i+1} belong to the same I/O of the current accessed core-cell *I*, but not to the same column (Ad_{i-1} = Ad_{i+1} = 10). With such addressing sequence, the only detectable failing element is the write driver (WD), as previously explained; this information is shown in column two. Finally, the March C- application with this addressing configuration between Ad_{i-1}, Ad_i and Ad_{i+1} provide the basic signature based on read operations (10100), exposed in columns three to eight. Thus, the extended signature is "101001010".

The list of extended signatures presented in Table 2 shows that:

1. The addressing configurations 1, 3 and 7 allow the detection and the location of URWF in the write driver.
2. The addressing configurations 5, 6 and 8 allow the detection of URWF but do not provide any information on the failure location. That means the URWF can be due to a failure in the pre-charge circuit or in the write driver as well.
3. The addressing configurations 2 and 4 allow the detection of URWF but also are able to exactly determine the failure location. In fact, with the same addressing sequence, different syndromes are generated according to the faulty elements. The "a" suffix is attached to faulty write drivers, whereas the "b" suffix is attached to faulty pre-charge circuits.

In the signature, the additional fields concerning the addressing sequence have been helpful to determine the occurrence of an URWF as well as the failure location.

Now, we analyze the second point mentioned above (ii), *i.e.* how to determine the static or dynamic nature of the fault. In Table 3, we give the set of signatures related to URWF and CFst (taken from [6]), considering March C- execution.

Fault model	R0	R1	R2	R3	R4	Ad_{i-1}	Ad_{i+1}
URWF	1	0	0	0	0	10	00
	1	0	0	0	0	11	10
	1	0	0	0	0	11	00
CFst(L,1,1)	1	0	0	0	0	xx	xx

Table 3: signatures -URWF v.s. CFst -

In this table, the fields marked with 'x' associated to the addressing configuration CFst signature, mean that the

addressing order has no impact on the fault detection. Table 3 shows that URWF and CFst are not distinguishable as they have the same syndrome. However, it also shows that depending on the sequence of accessed core-cells during test application, we can state if there is no URWF occurrence. In other words, with a single test sequence application, it is possible to determine if the memory is affected by static faults only. However, further test applications with different addressing sequences should be useful to completely differentiate static and dynamic faults.

4 Analysis and future development

The use of the *extended signature* is crucial to distinguish between static and dynamic detected faults. Although necessary, additional information concerning the address sequence used during test is sometimes not sufficient to achieve clear fault model discrimination. For this purpose, we intend to extend the current study and improve the presented diagnosis approach. This development will be mainly based on two axes:

- A further extension of the signature by adding information about the data background used during test application. In other words, we intend to take in account the data stored in the cells, as a field in the signature. In March tests, accurate data backgrounds are necessary to target specific dynamic fault models, such as Address Decoder Open Faults (ADOFs) [7] and Leakage Read Faults [8].
- The use of diagnosis responses given by intercrossing information coming from multiple March test signatures. This intercrossed information may allow reducing the number of ambiguous diagnosis and then improve the Diagnosability Ratio.

The development that we propose in this Section will involve the manipulation of a large amount of data. In order to make this approach really usable, we also plan to develop a software platform able to automate the data processing.

5 Conclusion

In this paper, we have proposed an approach for dynamic fault diagnosis in SRAMs. This approach is based on the extension of existing signature-based diagnosis methods. We show that the information concerning the addressing configuration of the executed March test can be crucial for the diagnosis of dynamic fault models. We have demonstrated the effectiveness of the proposed solution in identifying the failure location in the memory on a case study, the URWF. At the end of the paper, we have listed some future developments of this approach that take in account the test data background for further signature extension and the use of intercrossed signature for the solution of ambiguous diagnosis. In the paper we refer to one case study of dynamic fault, but the proposed approach is applicable to the whole set of dynamic fault models.

6 References

[1] A.J. van de Goor, Testing Semiconductor Memories, Theory and Practice, COMTEX Publishing, Gouda, The Netherlands, 1998

[2] M. Abramovich et al., “Digital System Testing and Testable Design”, IEEE Press, 1990

[3] M.F. Chang, W.K. Fuchs and J.H. Patel, “Diagnosis and Repair of Memory with Coupling Faults”, IEEE Transactions on Computers, vol. 38, no. 4, pp. 493-500, April 1989

[4] V.N. Yarmolik, Y.V. Klimets, A.J. van de Goor, S.N. Demidenko, “RAM diagnostic tests”, Memory Technology, Design and Testing, pp. 100-102, 1996

[5] D. Niggemeyer, M. Redeker, E.M. Rudnick, “Diagnostic testing of embedded memories based on output tracing”, Memory Technology, Design and Testing, pp. 113-118, 2000

[6] J.-F. Li, K.-L. Cheng, C.-T. Huang and C.-W. Wu, “March-Based RAM Diagnosis Algorithms for Stuck-At and Coupling Faults”, International Test Conference, pp. 758-767, 2001

[7] L. Dilillo, P. Girard, S. Pravossoudovitch, A. Virazel and S. Borri, "March iC-: An Improved Version of March C- for ADOFs Detection", VLSI Test Symposium, pp. 129-134, 2004

[8] L. Dilillo, B, Al-Hashimi, Rosinger P., Girard P. “Leakage Read Fault in Nanoscale SRAM: Analysis, Test and Diagnosis”, Proc. IEEE Int. Design and Test Workshop,Dubai, 2006

[9] Z. Al-Ars and A.J. van de Goor, "Static and Dynamic Behavior of Memory Cell Array Opens and Shorts in Embedded DRAMs", Design Automation and Test in Europe, pp. 496-503, 2001

[10] A.J. van de Goor and Z. Al-Ars, "Functional Memory Faults: A Formal Notation and a Taxonomy", VLSI Test Symposium, pp. 281-289, 2000

[11] S. Hamdioui, R. Wadsworth, J.D. Reyes and A.J. van de Goor, "Importance of Dynamic Faults for New SRAM Technologies", European Test Workshop, pp. 29-34, 2003.

[12] S. Hamdioui, Z Al-Ars and A.J. van de Goor, "Testing Static and Dynamic Faults in Random Access Memories", Proc. IEEE VLSI Test Symposium, pp. 395-400, 2002

[13] A. Benso, et al., “March AB, March AB1: new March tests for unlinked dynamic memory faults”, IEEE International Test Conference, 2005

[14] S. K. Thakur, R. Parekhji, A. N. Chandorkar, “On-chip Test and Repair of Memories for static and Dynamic faults”, IEEE Int. Test Conference, 2006

[15] V.A. Vardanian, et al., “Minimal March-Based Fault Location Algorithm with Partial Diagnosis for All Static Faults in Random Access Memories”, IEEE Design and Diagnostics of Electronic Circuits and systems, pp. 260-265, 2006

[16] S. M. Al-Harbi, et al., “March DSS: A New Diagnostic March Test for All Memory Simple Static Faults”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 26, no. 9, pp. 1713-1720, Sept. 2007

[17] D. Appello, et al., “Embedded Memory Diagnosis: An Industrial Workflow”, IEEE Intern. Test Conference, 2006

[18] S. Borri, M. Hage Hassan, L. Dilillo, P. Girard, S. Pravossoudovitch and A. Virazel, “Analysis of Dynamic Faults in Embedded-SRAMs: Implications for Memory Test”, Journal of Electronic Testing: Theory and Applications, Vol 21 N.2, 169-178, 2005

[19] A. Ney, P. Girard, C. Landrault, S. Pravossoudovitch, A. Virazel and M. Bastian, “Slow Write Driver Faults in 65nm Technology SRAM: Analysis and March Test Solution”, Proc. of IEEE Design Automation and Test in Europe, 2007

[20] A. Ney, P. Girard, C. Landrault, S. Pravossoudovitch, A. Virazel and M. Bastian, "Un-Restored Destructive Write Faults due to Resistive-Open Defects in the Write Driver of SRAMs", Proc. of IEEE VLSI Test symposium, 2007

[21] L. Dilillo, P. Girard, S. Pravossoudovitch, A. Virazel, "Resisitive open defect influence in SRAM pre charge circuit: Analysis and characterization", Proc. of IEEE European Test Symposium, pp 116-121, 2005

[22] R. D. Adams, E. S. Cooley, "False Write Through and Un-Restored Write Electrical Level Fault Models for SRAMs", Proc. IEEE Int. Workshop on Memory Technology, Design and Testing, pp. 27-32, 1997.

[23] D. Niggemeyer, M. Redeker and J. Otterstedt, "Integration of Non-classical Faults in Standard March Tests", Records of the IEEE Int. Workshop on Memory Technology, Design and Testing, pp. 91-96, 1998.

[24] L. Dilillo, P. Girard, S. Pravossoudovitch, A. Virazel, S. Borri, M. Hage-Hassan, “Efficient March Test Procedure for Dynamic Read Destructive Fault Detection in SRAM Memories”, Journal of Electronic Testing: Theory and Applications, Vol 21 N.5, 551-561, 2005