



## March iC-: An Improved Version of March C- for ADOFs Detection

Luigi Dilillo, Patrick Girard, Serge Pravossoudovitch, Arnaud Virazel, Simone Borri

### ► To cite this version:

Luigi Dilillo, Patrick Girard, Serge Pravossoudovitch, Arnaud Virazel, Simone Borri. March iC-: An Improved Version of March C- for ADOFs Detection. VTS: VLSI Test Symposium, Apr 2004, Napa Valley, CA, United States. pp.129-134, 10.1109/VTEST.2004.1299236 . lirmm-00108772

**HAL Id: lirmm-00108772**

**<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00108772>**

Submitted on 23 Oct 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# March iC-: An Improved Version of March C- for ADOFs Detection<sup>\*</sup>

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

S. Borri

*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>*

*Infineon Technologies France  
2600, Route des Crêtes  
06560 Sophia-Antipolis, France  
Email: [simone.borri@infineon.com](mailto:simone.borri@infineon.com)  
URL: <http://www.infineon.com>*

## Abstract

*This paper presents a new March test solution for detection of ADOFs, Address Decoder Open Faults, and resistive-ADOFs that are the consequence of resistive-open defects in address decoders of SRAM memories. In this study, we briefly analyze the test conditions and the March test requirements for these particular faults and we introduce some modifications to the well known March C- making it able to detect ADOFs and resistive-ADOFs, without increasing its complexity and its ability to detect the former target faults. The reformulation of March C-, called March iC-, is essentially based on introducing a particular address sequence and a particular read/write data sequence. The proposed March iC- extends the ability of March-based test solutions in detecting dynamic faults in SRAM memories.*

## 1. Introduction

In new design paradigms such as Systems-on-Chip (SoC), memory blocks hold the major part of the silicon area. This is confirmed by the SIA Roadmap which forecasts a memory density nearby 80 % of the chip in the next few years [1]. Moreover, in VDSM (Very Deep SubMicron) technologies, a new class of faults appears to be more and more problematic from a test point of view. These faults, called dynamic faults [2, 3], require more than one operation to be sensitized. For recent memory designs, it therefore becomes evident that efficient test solutions need to be developed in order to reach a high test quality, *i.e.* a high coverage of the new fault class.

Among the known dynamic faults, we focus our study on those generated by intra-gate open and resistive-open defects which may occur in address decoders. When an

open defect appears in an address decoder, particularly in the parallel plane of NAND/NOR gates, two bit lines or word lines may be erroneously selected at the same time. This fault, also called ADOF (Address Decoder Open Fault), have been considered in [4, 5], where an algorithmic solution is proposed, allowing the complete sensitization and observation. If the defect is a resistive one, the fault is called resistive-ADOF and it is considered as a generalization of ADOFs [6]. In this case, similar test approaches can be used with additional timing constraints.

Several test solutions can be used for ADOFs and resistive-ADOFs detection but March tests remain the most attractive solution due to their linear complexity and effectiveness for detection of a large number of other faults. However, March tests are constructed essentially for the detection of static faults as stuck-at and transition faults. Thus, ADOFs and resistive-ADOFs are not targeted by such test algorithms due to their dynamic nature.

March tests need therefore to be modified in order to detect ADOFs and resistive-ADOFs. Some studies have dealt with this solution [7, 8, 9, 10]. They exploit the Degrees of Freedom (DOF) inherent to March tests and especially the DOF that allows an address sequence modification [13]. The previous March test solutions allow the sensitization of all the faults in the address decoders. However, the main drawback of these techniques concerns the fault observation. In presence of an ADOF or a resistive-ADOF, an undefined value is read on the memory output and hence the fault effect cannot be observed. This observation problem will be illustrated in Section 3.

In [6], we have proposed a complete comparison between ADOFs and resistive-ADOFs test conditions. We have shown that additional timing constraints are required for resistive-ADOFs observation. The March test problematic for detecting such faults has been studied in [12] in which we propose two new March elements allowing the sensitization of ADOFs and resistive-ADOFs and ensuring complete fault observation. As for the previous techniques, the fault sensitization is achieved by an address sequence

---

<sup>\*</sup> This work has been partially funded by the French government under the framework of the MEDEA+ A503 "ASSOCIATE" European program.

modification, but this time the observation is ensured by a new read/write data sequence.

In this paper, we consider the well-known and efficient March C- and we propose to embed in it the properties of the March elements proposed in [12]. The reformulation of March C-, called March iC-, is essentially based on introducing a particular address sequence and a particular read/write data sequence making it able to detect ADOFs and resistive-ADOFs. We also show that these modifications do not change i) its complexity and, in particular, ii) its ability to detect the former target faults of March C-.

The rest of the paper is organized as follows. In Section 2 there is the description of a normal address decoder, the behavior and test conditions of ADOFs and resistive-ADOFs. Section 3 provides a detailed analysis of the March test coverage for such faults models. In this section we introduce some elements of our work proposed in [12]. In Section 4, we present the new March iC-, improved version of March C- that allows the detection of ADOFs and resistive-ADOFs. Concluding remarks are discussed in Section 5.

## 2. ADOFs and resistive-ADOFs

Figure 1 depicts the scheme of a 2-bit word line decoder. It is based on NOR-gates. The NAND and inverter gates are used for synchronization and buffering respectively. A similar address decoder is employed for bit line selection. Such a structure can be found in the Infineon 0.13 $\mu$ m synchronous embedded-SRAM architecture.

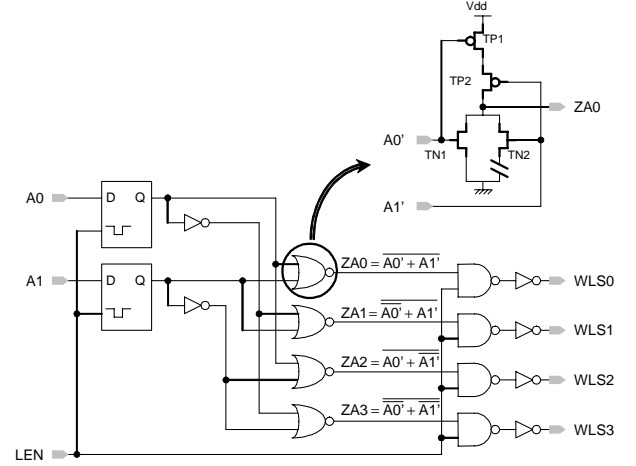
When it is fault free, the address decoder given in Figure 1, driven by signals A0 and A1, activates only one word line for each clock cycle. For example:

1.  $\langle A0, A1 \rangle = \langle 0, 0 \rangle \Rightarrow$  WLS0 is activated,
2. rising transition on A1:  $\langle A0, A1 \rangle = \langle 0, 1 \rangle \Rightarrow$  WLS2 is activated and WLS0 is deactivated.

Now let us describe the sequential behavior of an ADOF. Consider the address decoder of Figure 1 with the open defect in TN2. The corresponding waveforms are shown in Figure 2. As already mentioned, the presence of an ADOF induces a concurrent selection of two word lines:

1.  $\langle A0, A1 \rangle = \langle 0, 0 \rangle \Rightarrow$  WLS0 is activated,
2. rising transition on A1:  $\langle A0, A1 \rangle = \langle 0, 1 \rangle \Rightarrow$  WLS2 is activated and WLS0 remains activated.

In this case, the open defect in TN2 prevents the pull-down of the ZA0 node, which remains at logic level high because of the memory effect (node and NAND input capacitances).



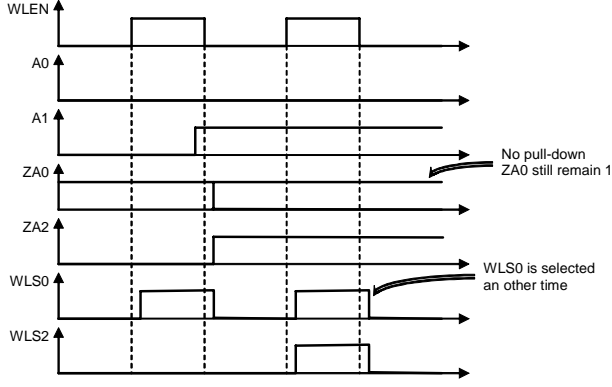
**Figure 1: A NOR-based word line address decoder**

In this example, we have first activated WLS0, (address  $\langle 0, 0 \rangle$ ) and then WLS2 (address  $\langle 0, 1 \rangle$ ). Between the two addresses only one-bit changes. This is required to sensitize the fault because a two-bit transition from  $\langle 0, 0 \rangle$  to  $\langle 1, 1 \rangle$  would activate both TN1 and TN2, thus discharging the node ZA0. So, WLS0 would be correctly deactivated in presence of the open defect because transistor TN1 would also be active, thus masking the faulty behavior of TN2. This shows that a necessary test condition for this fault is to provide an address sequence with a Hamming distance of 1 ( $Hd = 1$ ), i.e. each address must have only a single-bit transition compared to the previous one. An example of detection sequence is the following:

- a. Write ZERO at address  $\langle 0, 0 \rangle \Rightarrow$  WL0 is active,
- b. Write ONE at address  $\langle 0, 1 \rangle \Rightarrow$  WL2 is active.  
WLS0 remains active and there is a write operation in two memory cells,
- c. Read address  $\langle 0, 0 \rangle \Rightarrow$  ZERO is expected.

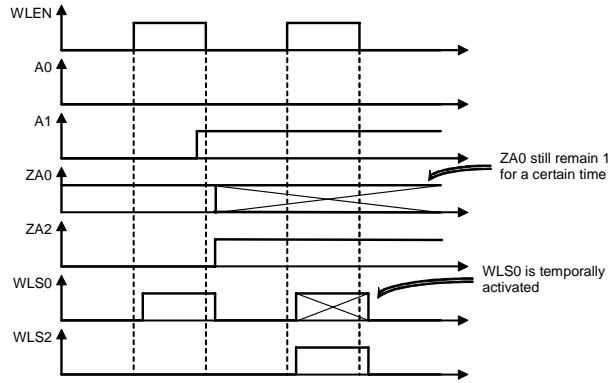
The first two phases, a and b, are useful for sensitization while phase c operates the observation. This strategy has been suggested in [4, 5], where the author proposes an algorithm (hereafter referred as Sachdev's algorithm) that produces the same sequence which sensitizes and detects the ADOFs through the whole memory. Sachdev's algorithm is effective for the sensitization and observation of ADOFs because the double addressing occurs during a write operation.

When the defect is a resistive-open, we are in presence of a resistive-ADOF, which has the behavior shown in Figure 3. After the input transition  $\langle A0, A1 \rangle = \langle 0, 0 \rangle \rightarrow \langle A0, A1 \rangle = \langle 0, 1 \rangle$ , WLS2 is correctly activated, while WLS0 remains activated erroneously for a certain time due to the delay of the pull-down operation.



**Figure 2: Waveforms of NOR-based address decoder with an ADOF**

For resistive-ADOFs, three cases are possible. A large resistive open defect involves the same behavior as an ADOF, thus two word lines are selected during the whole read or write phases. An intermediate defect size may induce the activation of two word lines for a certain time. In this case, there is a high probability that a dynamic fault occurs, with the same effects than in presence of an ADOF. Finally, if the defect is very small, the delay perturbation introduced in the circuit is irrelevant and not pathological.



**Figure 3: Waveforms of NOR-based address decoder with a resistive-ADOF**

Due to the partial or complete activation of two word lines, resistive-ADOFs can be detected by the same algorithms used for ADOFs testing. However, resistive-ADOF detection requires an additional timing constraint linked to the occurrence of the address transition. In [6] it is shown that the best test condition is obtained for an address transition close to the WLEN signal deactivation. Resistive-ADOFs can be considered as a generalization of ADOFs. In the following we will refer to these faults under the unique term ADOFs.

### 3. Problematic of March tests

Standard March tests are not efficient for ADOFs testing. Some test conditions have been proposed in [10] in order to detect all ADOFs by March tests. The fault sensitization requires an address sequence including all the pattern pairs with  $H_d = 1$ . Moreover, the considered March test must be effective for address decoder fault (AFs) testing.

The condition on the address sequence ( $H_d = 1$ ) can be operated by exploiting the first of the six Degrees of Freedom (DOF) of March tests:

**DOF I:** Any arbitrary address sequence can be defined as an  $\uparrow\uparrow$  sequence, as long as all addresses occur exactly once ( $\downarrow\downarrow$  is the reverse of  $\uparrow\uparrow$ ). The fault detection properties are independent of the utilized address sequence [11, 13].

The second condition, AFs detection, is achieved by many March tests and among these, we consider March C- [14] as a case study. It has a 10N complexity with the six March elements presented in Figure 4. Independently of the address sequence, March C- is effective for detecting many other fault types.

$$\{ \uparrow(w0) \uparrow(r0, w1) \uparrow(r1, w0) \downarrow(r0, w1) \downarrow(r1, w0) \uparrow(r0) \}$$

$$M_0 \quad M_1 \quad M_2 \quad M_3 \quad M_4 \quad M_5$$

**Figure 4: March C-**

Now, we propose an analysis of the previous statement with the NOR-based address decoder of Figure 1 and we show that during the observation phase some problems appear. The observation is exploited during the second March element ( $M_1$ ), when the ADOF involves a double addressing during the read operation as shown in Figure 5.

	Ad0	Ad1	Ad3	...
$M_0$	w0	w0	w0	...
$M_1$	r0, w1	r0, w1	r0, w1	...
...				

Two opposite values are read at the same time

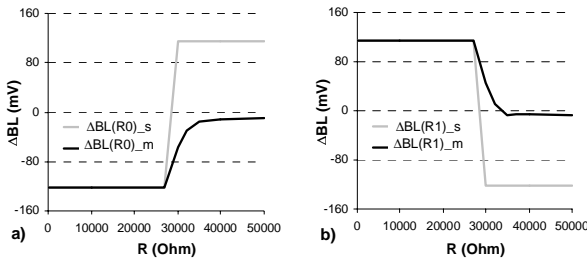
**Figure 5: ADOFs detection with March C- (Address sequence with  $H_d = 1$ )**

During  $M_1$ , for the first address,  $Ad0 \langle A0, A1 \rangle = \langle 0, 0 \rangle$ , we have a correct behavior. A "0" is read and a '1' is written. For the following address,  $Ad1 \langle A0, A1 \rangle = \langle 0, 1 \rangle$ , with only one bit transition, the r0 is performed and in presence of an ADOF the previous cell remains selected. Consequently, two different logic values, the '1' stored at  $Ad0$  and the '0' stored at  $Ad1$ , are read on the same bit line (BL) at the same time.

Electrical simulations have been performed for different size of the resistive-open defect (R) on the 0.13  $\mu m$  Infineon synchronous single-port SRAMs in order to

evaluate this particular condition [15]. Note that for this technology the data detection limit is  $\pm 80\text{mV}$  for  $\Delta\text{BL}$  ( $\Delta\text{BL} = \text{BL} - \text{BLB}$ ), *i.e.* this is the minimal voltage difference allowing to perform a correct read operation. BL and BLB signals are the core cell outputs.

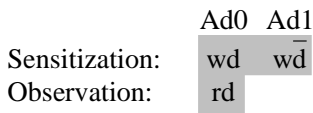
Figure 6.a gives the comparison during the r0 operation between Sachdev's algorithm and March C- test with an  $\text{Hd} = 1$  address sequence. It is shown that Sachdev's algorithm, referred as  $\Delta\text{BL}(\text{R0})_s$  in the waveform, is effective for ADOFs detection because a '1' is read instead of a '0' for a certain defect size  $R \geq 27\text{k}\Omega$ . On the other hand, the March C-, referred as  $\Delta\text{BL}(\text{R0})_m$ , does not ensure the fault detection because an undefined value is read for the same defect size. In Figure 6.b similar results are shown for a r1 operation.



**Figure 6: Sachdev's algorithm vs. March C-**

In conclusion, Sachdev's algorithm is effective for the sensitization and observation of ADOFs because the double addressing occurs during a write operation. The same does not occur for the March C-. This algorithm is effective for the sensitization, but the observation phase is not done. The double cell access occurs during the read operation causing two opposite values to be driven on the same bit line.

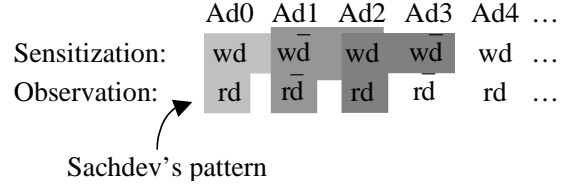
For a proper detection of ADOFs, it is necessary to sensitize the fault during the write operation and observe it during a separate read as done by Sachdev's algorithm. Its three phases can be graphically illustrated as in Figure 7, where between  $\text{Ad0}$  and  $\text{Ad1}$ ,  $\text{Hd} = 1$ ,  $d = \text{data}$  ('0' or '1') and  $\bar{d}$  is its opposite value.



**Figure 7: Sachdev's pattern**

Now, we show how this pattern can be implemented by a March test. For this purpose, we can reiterate the Sachdev's pattern for all addressable cells, as shown in Figure 8. Instead of performing sensitization and observation for each cell, we can execute a sensitization

phase at the same time for all the cells by a serial write operation with alternating data  $d$  and  $\bar{d}$  followed by a global observation phase by reading the written data.



**Figure 8: Sachdev's adaptation to March elements**

This can be translated in the two March elements of Figure 9, where  $A$  is a logic value which starts with '0' or '1' and takes the opposite value for each new address having  $\text{Hd} = 1$ .

$$\left\{ \uparrow(wA) \uparrow(rA) \right\} \quad A \text{ alternating logic value}$$

$$M_A \quad M_B$$

**Figure 9: New March elements for ADOFs detection**

The possibility to change the data values during the execution of a March element is justified by the fourth DOF of March tests:

**DOF IV:** *The data within a read/write operation does not necessarily have to be equivalent for all memory addresses as long as the detection probabilities of basic faults are not affected* [11].

In order to ensure that the proposed March elements cover all the ADOFs and resistive-ADOFs, it is necessary that the sequence of  $2^m$  produced addresses (where  $m$  is the total number of address bits) contains all the  $n \times 2^n$  single-bit transitions (where  $n$  is the bit-width of the considered decoder) [7]. In other words, the necessary condition for complete detection is the following one:

$$2^m \geq n \times 2^n + 1 \quad (\text{Eq. 1})$$

As detailed in [12] this condition (Eq. 1) is most of the time satisfied. This is the case of the considered Infineon SRAM memory architecture where the address decoders are composed by pre-decoders and post-decoders. In particular the largest sub-address decoder has a 3-bit width; consequently there are all the needed conditions for the generation of the necessary addresses with  $\text{Hd}=1$  by a structure like an LFSR, linear feedback shift register.

The validation of the proposed solution is observable by the waveforms of Figure 6. During the test operation of the new March elements, the electrical behavior of the memory circuit is similar to Sachdev's algorithm simulation. The double addressing, due to the ADOF, occurs during the write operation (sensitization phase). The observation phase is done during the read operation

without uncertainty because there is not double addressing with opposite data as before.

#### 4. The improved March C-: March iC-

In the previous section we have introduced two new March elements in order to detect ADOFs (Figure 9). Now we propose an evolution of this solution. In other words, we intend to embed the particular characteristics of these elements in the March C- to make it able to test ADOFs, without degradation of its target fault detection capability and without complexity increase. This new March test is called March iC-. In the following subsections, we present this new test and we validate its capability to detect ADOFs and the former target faults.

##### 4.1. March iC- and notations

The modifications that we introduce in March C- are the following ones:

- Address sequence with  $Hd = 1$ ,
- Alternating data value  $A_v$ , for the read or write operations; with  $v$  is the initial value.

With these modifications we produce the improved March C-, that we call March iC-. It has the structure shown in Figure 10.

$$\left\{ \begin{array}{cccccc} \uparrow(A_v) & \uparrow(A_v, wA_v) & \uparrow(A_v, wA_v) & \downarrow(A_v, wA_v) & \downarrow(A_v, wA_v) & \uparrow(A_v) \\ M_0 & M_1 & M_2 & M_3 & M_4 & M_5 \end{array} \right\}$$

Figure 10: March iC-, general version

In order to clarify the structure and the use of the March iC-, we introduce some elements of notations. Every memory structure is composed by an even number of cells, thus there is an even number of addresses. We denote as  $A_v$  the alternating data with starting value  $v \in \{0,1\}$ ;  $A_{\bar{v}}$  is its opposite. If we choose  $v = '0'$  as starting value for a write operation the value written in the last cell is '1'. If after this we do a read operation with inverse addressing order, the first value that is expected is  $v = '1'$ . For this reason, in  $M_2 \uparrow(A_{\bar{v}}, wA_v)$  the starting value for the alternative data to write is  $v$  and the starting read value in  $M_3 \downarrow(A_v, wA_v)$  is  $\bar{v}$ . This is clearly shown in Figure 11 where  $M_2$  and  $M_3$  are exploited in the case of a four cell memory.

After the application of element  $M_2$ , the stored data is shown in the column in the middle of Figure 11. The following read operation, element  $M_3$ , starts with the expected data  $\bar{v}$  at the address '3', i.e. the last cell where  $M_2$  operates.

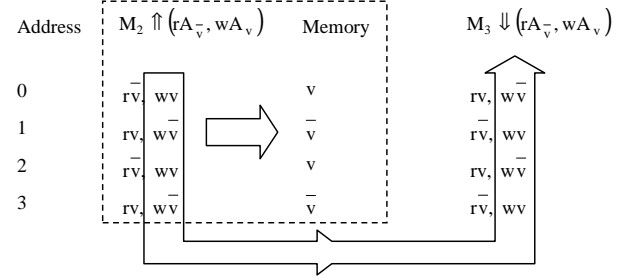


Figure 11: Elements M2 and M3 applied in a four cells memory

##### 4.2. Detection capability of March iC-

Now we demonstrate the capability of March iC- to detect ADOFs and the former target faults of March C-.

We can easily find in the March iC- the main characteristics, which make the elements proposed in Figure 9 able to detect ADOFs. In fact March iC- is exploited with an addressing order with  $Hd = 1$ , and the data stored and expected are alternate. Consequently the element  $M_0$  followed by  $M_1$ , and  $M_3$  followed by  $M_4$ , satisfy the conditions to detect ADOFs, as described in section 3. We analyze the couple of elements  $M_0, M_1$ . The writing only element  $M_0$  operates the sensitization of eventual ADOFs by writing alternating data  $A_v$ . In fact, in presence of an ADOF one of the write operations is abnormally performed on two cells at same time, overwriting the stored value in a cell with its opposite. The read operations of  $M_1$  perform the observation phase allowing to detect the fault as an unexpected data. The same remarks can be done for other sequences of elements like  $M_3$  and  $M_4$ .

The classic March C- is able to detect the following fault types: SAFs: stuck at faults; TFs: transition faults; CFids: unlinked idempotent 2-coupling faults; CFinvns: unlinked inversion 2-coupling faults; CFdyns: dynamic coupling faults; SCFs: static coupling faults; AFs: static address decoder faults.

March iC- is still able to detect these faults because this algorithm differs from March C- only for the addressing order and data written/expected. As mentioned before the use of a particular addressing sequence does not change the capability of the test. Moreover, as demonstrated in [16], when a March test is symmetric (March C- is symmetric), free definition of test data  $d = f(\text{address})$  is allowed. This means that the modifications, that we have done to March C-, have not changed its capability. We have completely verified this statement, i.e. we have verified that March iC- is able to detect all the target faults of March C-. In table 1, as example, we show only a part of this verification, in particular there are the elements of March iC- useful for the test of the 2-coupling faults.

## 5. Conclusions

The presented study has focused on dynamic defects that may occur in address decoders of memories. In particular we have focused our attention on ADOFs and their generalization, resistive-ADOFs.

After electrical analyses we have stated that ADOFs can be detected only when the sensitization phase involves a double addressing during the write operation. With this information we have exploited some Degrees of Freedom of the March tests (DOF I and IV) in order to generate new March elements for ADOFs detection. In a second time we have embedded the properties of these new elements into the well known March C-, producing a new test that we have called March iC-. We have finally showed the capability of this algorithm to detect ADOFs and the former target faults of March C-.

## References

- [1] Semiconductor Industry Association (SIA), "International Technology Roadmap for Semiconductors (ITRS)", 1999 Edition.
- [2] A.J. van de Goor and Z. Al-Ars, "Functional Memory Faults: A Formal Notation and a Taxonomy", Proc. VLSI Test Symposium, May 2000, pp.281-289.
- [3] Z. Al-Ars and A.J. van de Goor, "Static and Dynamic Behavior of Memory Cell Array Opens and Shorts in Embedded DRAMs", Proc. Design, Automation and Test in Europe, 2001, pp. 496-503.
- [4] M. Sachdev, "Test and Testability Techniques for Open Defects in RAM Address Decoders", Proc. European Design & Test Conference, 1996, pp.428-434.
- [5] M. Sachdev, "Open Defects in CMOS RAM Address Decoders", IEEE Design & Test of Computers, vol.14, n.2, Apr-Jun 1997, pp. 26-33.
- [6] L. Dilillo, P. Girard, S. Pravossoudovitch, A. Virazel and S. Borri, "Comparison of open and Resistive-Open Defect Test Conditions in SRAM Address Decoders", Proc. Asian Test Symposium, 2003, pp. 250-255.
- [7] J. Otterstedt, D. Niggemeyer and T.W. Williams, "Detection of CMOS Address Decoder Open Faults with March and Pseudo Random Memory Tests", Proc. Int. Test Conf., 1998, pp.53-62.
- [9] D. Youn, T. Kim and S. Park, "A Microcode-based Memory BIST Implementing Modified March Algorithm", Proc. Asian Test Symposium, 2001, pp. 391-395.
- [8] E. Gizdarski, "Detection of Delay Faults in Memory Address Decoder", Journal of Electronic Testing: Theory and Applications, N°16, 2000, pp. 381-387.
- [10] M. Klaus and Ad J. van de Goor, "Test for resistive and capacitive defects in address decoders", Proc. Asian Test Symposium, 2001, pp. 31-36.
- [11] D. Niggemeyer, M. Redeker and J. Otterstedt, "Integration of Non-classical Faults in Standard March Tests", Records of the Int. Workshop on Memory Technology, Design and Testing, 1998.
- [12] L. Dilillo, P. Girard, S. Pravossoudovitch, A. Virazel, S. Borri and M. Hage-Hassan, "March Tests Improvement for Address Decoder Open and Resistive Open Faults Detection", Latin American Test Workshop, 2004.
- [13] M. Nicolaidis, "An Efficient Built-In Self-Test Scheme for Functional Test of Embedded Memories", Proc. Int. Symposium Fault Tolerant Computing, 1985.
- [14] M. Marinescu, "Simple and Efficient Algorithms for Functional RAM Testing", Proc. Int. Test Conf., 1982, pp.236-239.
- [15] S. Borri, M. Hage-Hassan, P. Girard, S. Pravossoudovitch and A. Virazel, "Defect-Oriented Dynamic Fault Models for Embedded-SRAMs", Proc. European Test Workshop, 2003, pp. 23-28.
- [16] M. Nicolaidis, "Theory of Transparent BIST for RAMs", IEEE Trans. On Computers, vol. 45, N° 10, October 1996, pp. 1141-1155.

		$Ad_j < Ad_i$				$Ad_i < Ad_j$			
		$Ad_i \text{ even}$	$Ad_i \text{ odd}$	$Ad_i \text{ even}$	$Ad_i \text{ odd}$	$Ad_i \text{ even}$	$Ad_i \text{ odd}$	$Ad_i \text{ even}$	$Ad_i \text{ odd}$
CFids	$< \uparrow ; 0 >$	M+M	M+M	M	M	M+M	M+M	M	M
	$< \uparrow ; 1 >$	M <sub>1</sub>	M <sub>2</sub>	M <sub>3</sub> +M <sub>4</sub>	M <sub>4</sub> +M <sub>5</sub>	M <sub>4</sub>	M <sub>3</sub>	M <sub>0</sub> +M <sub>1</sub>	M <sub>1</sub> +M <sub>2</sub>
	$< \downarrow ; 0 >$	M <sub>2</sub>	M <sub>1</sub>	M <sub>4</sub> +M <sub>5</sub>	M <sub>3</sub> +M <sub>4</sub>	M <sub>3</sub>	M <sub>4</sub>	M <sub>1</sub> +M <sub>2</sub>	M <sub>0</sub> +M <sub>1</sub>
	$< \downarrow ; 1 >$	M <sub>3</sub> +M <sub>4</sub>	M <sub>4</sub> +M <sub>5</sub>	M <sub>1</sub>	M <sub>2</sub>	M <sub>0</sub> +M <sub>1</sub>	M <sub>1</sub> +M <sub>2</sub>	M <sub>4</sub>	M <sub>3</sub>
CFinvs	$< \uparrow ; \uparrow >$	M	M	M	M	M	M	M	M
	$< \downarrow ; \downarrow >$	M <sub>3</sub>	M <sub>1</sub>	M <sub>1</sub>	M <sub>2</sub>	M <sub>3</sub>	M <sub>4</sub>	M <sub>4</sub>	M <sub>3</sub>
CFdyns	$< r ; 0 >$	M	M	M	M	M	M	M	M
	$< r ; 1 >$	M <sub>1</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>2</sub>	M <sub>4</sub>	M <sub>3</sub>	M <sub>4</sub>	M <sub>3</sub>
	$< w ; 0 >$	M <sub>2</sub>	M <sub>1</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>3</sub>	M <sub>4</sub>	M <sub>3</sub>	M <sub>4</sub>
	$< w ; 1 >$	M <sub>1</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>2</sub>	M <sub>4</sub>	M <sub>3</sub>	M <sub>4</sub>	M <sub>3</sub>
SCFs	$< 0 ; 0 >$	M	M	M+M	M+M	M	M	M+M	M+M
	$< 0 ; 1 >$	M <sub>3</sub> +M <sub>4</sub>	M <sub>4</sub> +M <sub>5</sub>	M <sub>1</sub>	M <sub>2</sub>	M <sub>0</sub> +M <sub>1</sub>	M <sub>1</sub> +M <sub>2</sub>	M <sub>4</sub>	M <sub>3</sub>
	$< 1 ; 0 >$	M <sub>4</sub> +M <sub>5</sub>	M <sub>3</sub> +M <sub>4</sub>	M <sub>2</sub>	M <sub>1</sub>	M <sub>1</sub> +M <sub>2</sub>	M <sub>0</sub> +M <sub>1</sub>	M <sub>3</sub>	M <sub>4</sub>
	$< 1 ; 1 >$	M <sub>1</sub>	M <sub>2</sub>	M <sub>3</sub> +M <sub>4</sub>	M <sub>4</sub> +M <sub>5</sub>	M <sub>4</sub>	M <sub>3</sub>	M <sub>0</sub> +M <sub>1</sub>	M <sub>1</sub> +M <sub>2</sub>

Table 1: Complete faults validation of the improved March C-