Static and Dynamic Behavior of Memory Cell Array Spot Defects in Embedded DRAMs

Zaid Al-Ars and Ad J. van de Goor, *Fellow*, *IEEE*

Abstract—Spot defects in memory devices are caused by imperfections in the fabrication process of these devices. In order to analyze the faulty effect of spot defects on the memory behavior, simulations have been performed on an electrical model of the memory in which the defects are injected, causing opens, shorts, or bridges. In this paper, simulation is used to analyze the faulty behavior of embedded DRAM (*e*DRAM) devices produced by Infineon Technologies. The paper applies the new approach of fault primitives to perform this analysis. The analysis shows the existence of most traditional memory fault models and establishes new ones. The paper also investigates the concept of dynamic faulty behavior and establishes its importance for *e*DRAMs. Conditions to test the newly established fault models, together with a test, are also given.

Index Terms—Embedded DRAM, functional fault models, fault primitives, spot defects, defect simulation, dynamic faulty behavior.

1 Introduction

Embedded DRAMs (eDRAMs) are dynamic RAM cores used on-chip along with other electrical components. Using on-chip memory components has many advantages over external memory chips, such as an increased bandwidth, reduced power consumption, suitable memory organization, and low electromagnetic interference [1]. Although *e*DRAMs have been extensively used in application specific integrated circuits (ASICs), little has been published on their fault analysis and testing.

For quite a while now, researchers have been studying the faulty behavior of memory devices and defining *functional fault models (FFMs)* to describe the detected faulty behavior, and develop tests which target these FFMs [2], [3]. On the other hand, papers have been published that study the faulty behavior of memory devices by performing a large number of tests and statistically analyzing the detected FFMs [4], [5]. The results of the theoretical and practical analysis show that our ability to understand, and thus predict, the faulty behavior of memories is still limited to relatively simple cases of defective devices.

Much of the work on functional fault modeling has been concerned with modeling faults sensitized by a single performed operation; these fault models are referred to as *static FFMs*. In this paper, it is shown that a large number of FFMs exist that have to be sensitized by a sequence of two or more operations; these are referred to as *dynamic FFMs*. The analysis is performed on the memory cell array of an *e*DRAM, by injecting electrical models of the spot defects into a model of the *e*DRAM. Naik et al. have used this approach for static FFMs in SRAMs [6].

• *The authors are with the Section of Computer Engineering, Faculty of Information Technology and Systems, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands.*
E-mail: z.e.al-ars@its.tudelft.nl, vdGoor@cardit.et.tudelft.nl.

Manuscript received 13 Mar. 2001; revised 21 Jan. 2002; accepted 11 Apr. 2002.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number 113785.

This paper is organized as follows: Section 2 describes the used *e*DRAM simulation model, then Section 3 defines the static and dynamic FFMs targeted in this paper. In Section 4, the spot defects to be injected into the simulation model are defined and classified. Section 5 gives the methodology to be used for performing the simulations and extracting the FFMs, while Section 6 discusses the simulation results. Section 7 uses these results to derive detection conditions and to extend current functional tests to detect the dynamic faulty behavior. Finally, Section 8 ends with the conclusions.

2 *e*DRAM Simulation Model

This section introduces the *e*DRAM simulation model used for defect injection and fault analysis. The analysis focuses on the memory cell array part of the *e*DRAM since it has the largest chip area and is the most fault sensitive.

The simulation model is based on a design-validation model of an actual *e*DRAM produced by Infineon Technologies. Since the time needed for simulating a complete memory device is excessively long, the simulation model used is simplified, taking two factors into consideration in order to preserve the accuracy of the model. First, removed components should be electrically compensated, and, second, the resulting simplified circuit should describe enough of the memory to enable injecting the defects of interest.

Fig. 1 shows a block diagram of the cell array column of the simulated *e*DRAM. The simplified simulation model contains a 2×2 cell array, in addition to two reference cells, precharge circuits, and a sense amplifier. The removed memory cells are compensated by resistances and capacitances along the bit line. In addition to the shown cell array column, the simulation model contains one data output buffer needed to examine data on output lines and a write driver needed to perform write operations.

All simulations have been done using the simulator "Pstar" (a commercial Spice-based simulator) and using a

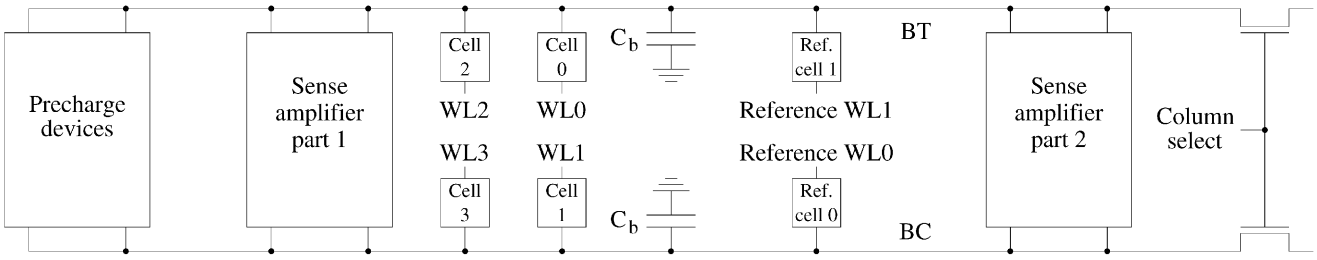


Fig. 1. Cell array column of the *e*DRAM, complete with reference cells, sense amplifier, precharge, and access circuits.

transistor model compatible with the Spice Level 3 model. Fig. 2 shows the simulation results of the properly functional memory while performing a write 0 operation followed by a read operation performed on Cell 0. The figure is divided into three panels, each with time as the horizontal axis and the voltage as the vertical axis. The first panel shows voltages on the bit lines (BT and BC), named `VN(BT)` and `VN(BC)`, respectively. They show the shared effect of any defect in the cell array column on other parts of the column. The second panel shows the voltage stored across the storage capacitor of Cell 0 (referred to in the figure as `V(C_S0)`), which reveals the short and long term effects of a defect on the stored logic value. Finally, the third panel shows the voltage on the T and F nodes of the data output buffer (referred to as `VN(DATA_T)` and `VN(DATA_F)`, respectively), which indicate whether the defect in the array column causes a fault to be detected on the output.

Despite the fact that the general structure of the *e*DRAM model shown in Fig. 1 is similar to the structure of other types of DRAM, the device parameters used for this model are derived for an *e*DRAM fabrication process. Whether the results given in this paper are also applicable for other DRAM products is a question open for investigation.

3 Definition of FFMs

In this section, the FFMs used in this paper are defined. First, a classification of the FFMs is presented, which divides the total space of faults into a number of classes. Then, four of these classes are discussed and used to define the targeted FFMs.

3.1 Classification of Fault Models

Two basic ingredients are needed to define any fault model: a sequence of performed memory operations and the corresponding deviations in the observed behavior from the expected one. The only functional deviations considered relevant to the faulty behavior are the stored logic value in the cell and the output value of a read operation.

Any difference between the observed and expected memory behavior can be denoted by the following notation $< S/F/R >$, referred to as a *fault primitive (FP)*. S describes the *sensitizing operation sequence (SOS)* that sensitizes the fault, F describes the value of the faulty cell, $F \in \{0, 1\}$, and R describes the logic output level of a read operation, $R \in \{0, 1, -\}$. The "–" is used in case a write, and not a read, is the operation that sensitizes the fault.

The set of all possible FPs spans a two-dimensional space with axis $\#C$ and $\#O$ (see Fig. 3). $\#C$ represents the number of different cells accessed by an SOS, while $\#O$ represents the number of operations performed in that SOS [7].

The notion of FPs makes it possible to give a precise definition of an FFM as understood for memory devices. This definition is presented next.

A **functional fault model** (FFM) is a nonempty set of fault primitives (FPs).

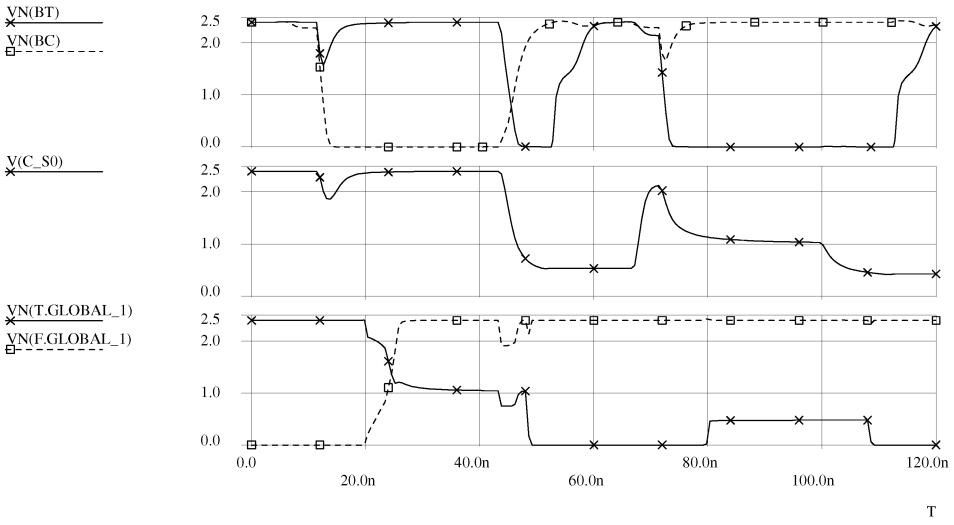


Fig. 2. Simulation of $1w0r0$ performed on Cell 0.

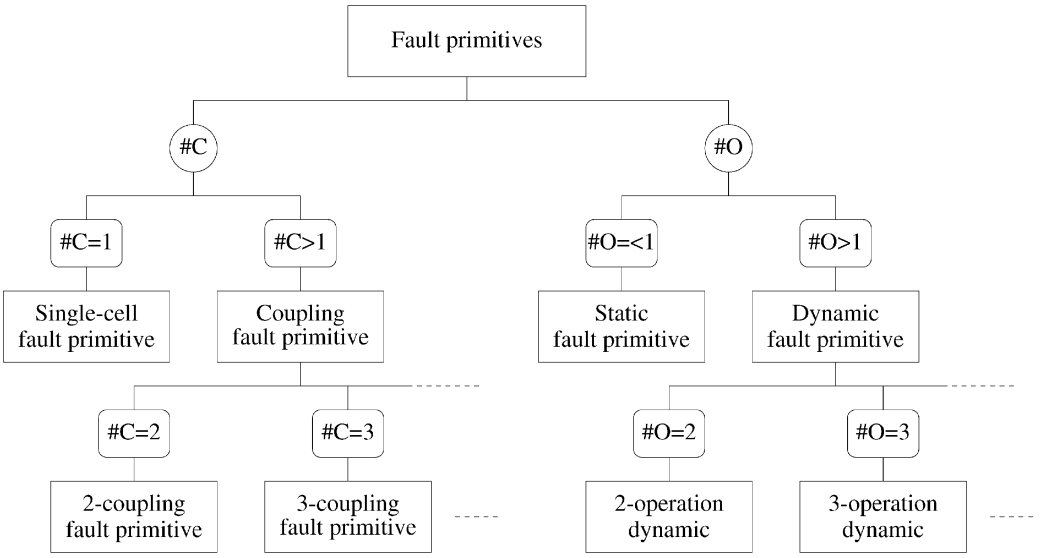


Fig. 3. Taxonomy of fault primitives.

3.2 Single-Cell Static FFMs

Single-cell static FFMs describe faults sensitized by performing at most one operation on the faulty cell. As mentioned earlier, a particular FP is denoted by $< S/F/R >$. S describes the value or operation that sensitizes the fault, $S \in \{0, 1, 0w0, 0w1, 1w0, 1w1, 0r0, 1r1\}$ for static FPs. A 0 means that the simulation starts with a memory cell initialized to a logic 0. A $0w0$ means that the simulation starts with a cell initialized to 0 and thereafter it performs a $w0$. A $0r0$ means that the simulation starts with a cell initialized to 0 and performs a read operation thereafter, with expected value 0. F and R have already been defined in Section 3.1.

Now that the possible values for S, F, and R are known for single-cell static FPs, it is possible to list all FPs using this notation. Table 1 lists all 12 possible combinations of the values in the $< S/F/R >$ notation that result in FPs. The column "Fault model" states the FFM defined by the corresponding FP.

All FPs listed in Table 1 are targeted in this paper. Below, they are used to define six different FFMs described in terms of nonempty sets of FPs.

1. **State faults** ($\mathbf{SF}_x$)—A cell is said to have an SF if the logic value of the cell flips before it is accessed, even if no operation is performed on it.[1] Two types of SF exist: $\mathbf{SF}_0 = \{< 0/1/- >\}$, with FP #1, and $\mathbf{SF}_1 = \{< 1/0/- >\}$, with FP #2. The notation $< 0/1/- >$ denotes that $S = 0$ (i.e., the simulator attempts to initialize the cell to logic 0), $F = 1$ (i.e., the cell contains a 1), and $R = -$ (i.e., no value is read since the SOS does not contain a read operation applied to the faulty cell).
2. **Transition faults** ($\mathbf{TF}x$)—A cell is said to have a TF if it fails to undergo a transition ($0 \rightarrow 1$ or $1 \rightarrow 0$) when it is written. Two types of TF exist: $\mathrm{TF} \uparrow = \{< 0w1/0/- >\}$, with FP #4, and $\mathrm{TF} \downarrow = \{< 1w0/1/- >\}$, with FP #5.
3. **Read disturb faults** ($\mathbf{RDF}_x$) [3]—A cell is said to have an RDF if a read operation performed on the cell changes the data in the cell and returns an incorrect value on the output. Two types of RDF exist: $\mathrm{RDF}_0 = \{< 0r0/1/1 >\}$, with FP #9, and $\mathrm{RDF}_1 = \{< 1r1/0/0 >\}$, with FP #10.
4. **Write disturb faults** ($\mathbf{WDF}_x$)—A cell is said to have a WDF if a nontransition write operation ($0w0$ or $1w1$) causes a transition in the cell. Two types of WDF exist: $\mathrm{WDF}_0 = \{< 0w0/1/- >\}$, with FP #3, and $\mathrm{WDF}_1 = \{< 1w1/0/- >\}$, with FP #6.
5. **Incorrect read faults** ($\mathbf{IRF}_x$)—A cell is said to have an IRF if a read operation performed on the cell returns the incorrect logic value while keeping the correct stored value in the cell. Two types of IRF exist: $\mathrm{IRF}_0 = \{< 0r0/0/1 >\}$, with FP #7, and $\mathrm{IRF}_1 = \{< 1r1/1/0 >\}$, with FP #12.
6. **Deceptive read disturb faults** ($\mathbf{DRDF}_x$) [3]—A cell is said to have a DRDF if a read operation performed on the cell returns the correct logic value, while it results in changing the contents of the cell. Two types of DRDF exist: $\mathrm{DRDF}_0 = \{< 0r0/1/0 >\}$, with FP #8, and $\mathrm{DRDF}_1 = \{< 1r1/0/1 >\}$, with FP #11.

The six FFMs defined above cover the space of all 12 single-cell static FPs of Table 1. Any single-cell static FFM can be represented as the union set of two or more of these 12 FPs. For example, if a particular defect results in a faulty behavior represented by an incorrect read-1 fault (IRF_1) as well as a read-0 disturb fault (RDF_0), then the corresponding behavior is described as

$$\{< 1r1/1/0 >\} \cup \{< 0r0/1/1 >\} = \mathrm{IRF}_1 \cup \mathrm{RDF}_0.$$

3.3 Single-Cell Dynamic FFMs

FFMs sensitized by performing more than one operation on the faulty memory cell are called *dynamic FFMs*. There are 2-operation, 3-operation, ..., dynamic FFMs; depending on

1. It should be noted that the state fault should be understood in the static sense. That is, the cell should flip in the short time period after initialization and before accessing the cell.

TABLE 1
All Possible Single-Cell Static FPs

#	S	F	R	FP	Fault model
1	0	1	–	$<0/1/->$	SF_0
2	1	0	–	$<1/0/->$	SF_1
3	$0w0$	1	–	$<0w0/1/->$	WDF_0
4	$0w1$	0	–	$<0w1/0/->$	TF↑
5	$1w0$	1	–	$<1w0/1/->$	TF↓
6	$1w1$	0	–	$<1w1/0/->$	WDF_1
7	$0r0$	0	1	$<0r0/0/1>$	IRF_0
8	$0r0$	1	0	$<0r0/1/0>$	$DRDF_0$
9	$0r0$	1	1	$<0r0/1/1>$	RDF_0
10	$1r1$	0	0	$<1r1/0/0>$	RDF_1
11	$1r1$	0	1	$<1r1/0/1>$	$DRDF_1$
12	$1r1$	1	0	$<1r1/1/0>$	IRF_1

$\#O$. Here, we restrict ourselves to the analysis of 2-operation dynamic FFMs.

There are 30 different single-cell 2-operation dynamic FPs possible [7], all of which are compiled in Table 2. In order to reduce simulation time, not all 30 FPs are considered. We choose only to target the four dynamic SOSs $0w0r0$, $0w1r1$, $1w0r0$, and $1w1r1$ (in short, $xwyry$), resulting in 12 possible FPs (listed in bold in Table 2): 3, 4, 5, 8, 9, 10, 13, 14, 15, 18, 19, and 20. The motivation for this choice is the fact that, in memory devices, an isolated write operation may not be sufficient to detect a fault since, externally, a cell needs to be read to detect the stored value set during the write.

The four targeted SOS's are capable of sensitizing 12 single-cell 2-operation FPs, which are used to define the following three FFMs. The names of these FFMs are chosen in such a way that they represent an extension of the single-cell static FFMs defined in Section 3.2.

1. **Dynamic read disturb fault** ($\mathbf{RDF}_{xy}$) is a fault whereby an $xwyry$ SOS changes the stored logic value to $\overline{y}$ and gives an incorrect output. Four types of dynamic RDF exist: $\mathrm{RDF}_{00}=\{<0w0r0/1/1>\}$ with FP #5, $\mathrm{RDF}_{11}=\{<1w1r1/0/0>\}$ with FP #18, $\mathrm{RDF}_{01}=\{<0w1r1/0/0>\}$ with FP #8, and $\mathrm{RDF}_{10}=\{<1w0r0/1/1>\}$ with FP #15.
2. **Dynamic incorrect read fault** ($\mathbf{IRF}_{xy}$) is a fault whereby an $xwyry$ SOS returns the logic value $\overline{y}$ while keeping the correct state of the cell. Four types of dynamic IRF exist: $\mathrm{IRF}_{00}=\{<0w0r0/0/1>\}$ with FP #3, $\mathrm{IRF}_{11}=\{<1w1r1/1/0>\}$ with FP #20, $\mathrm{IRF}_{01}=\{<0w1r1/1/0>\}$ with FP #10, and $\mathrm{IRF}_{10}=\{<1w0r0/0/1>\}$ with FP #13.
3. **Dynamic deceptive read disturb fault** ($\mathbf{DRDF}_{xy}$) is a fault whereby an $xwyry$ SOS returns the correct logic value y while destroying the state of the cell. Four types of dynamic DRDF exist: $\mathrm{DRDF}_{00}=\{<0w0r0/1/0>\}$ with FP #4, $\mathrm{DRDF}_{11}=\{<1w1r1/0/1>\}$ with FP #19, $\mathrm{DRDF}_{01}=\{<0w1r1/0/1>\}$ with FP #9, and $\mathrm{DRDF}_{10}=\{<1w0r0/1/0>\}$ with FP #14.

3.4 Two-Cell Static FFMs

Two-cell static FFMs describe faults sensitized by performing at most one operation while considering the effect two different cells have on each other. A two-cell static FP can be represented as follows: $<S/F/R>=<S_a;S_v/F/R>_{a,v}$, where S_a and S_v are the sequences performed on the aggressor and victim, respectively. S_a and $S_v \in \{0,1,0w0,0w1,1w0,1w1,0r0,1r1\}$. Table 3 lists all 36 possible two-cell static FPs this notation can distinguish [7].

Below, a list of FFMs is constructed from the 36 FPs in such a way that all FPs are covered by at least one FFM.

1. **State coupling fault (CFst)** [8] is a fault whereby the victim is forced into a given logic state only if the aggressor is in a given state, without performing any operation on the victim. This fault is special in the sense that no operation is needed to sensitize it and, therefore, it only depends on the initial stored values in the cells. Four types of CFst exist: $\mathrm{CFst}_{0;0}=\{<0;0/1/->\}$ with FP #1, $\mathrm{CFst}_{0;1}=\{<0;1/0/->\}$ with FP #2, $\mathrm{CFst}_{1;0}=\{<1;0/1/->\}$ with FP #3, and $\mathrm{CFst}_{1;1}=\{<1;1/0/->\}$ with FP #4.
2. **Disturb coupling fault (CFds)** is a fault whereby an operation (write or read) performed on the aggressor forces the victim into a given logic state [9] (i.e., $S_a=\{r0,r1,w0,w1\}$). Here, a read as well as a write performed on the aggressor is a sensitizing operation for the fault, whereby, in case of a write operation, the value of the to be written data, and not the fact whether it is a transition write or a nontransition write operation, is relevant for the fault model. Twelve types of CFds exist: $\mathrm{CFds}_{0w0;0}=\{<0w0;0/1/->\}$ with FP #5, $\mathrm{CFds}_{0w0;1}=\{<0w0;1/0/->\}$ with FP #6, $\mathrm{CFds}_{1w1;0}=\{<1w1;0/1/->\}$ with FP #11, $\mathrm{CFds}_{1w1;1}=\{<1w1;1/0/->\}$ with FP #12, $\mathrm{CFds}_{0w1;0}=\{<0w1;0/1/->\}$ with FP #7, $\mathrm{CFds}_{0w1;1}=\{<0w1;1/0/->\}$ with FP #8, $\mathrm{CFds}_{1w0;0}=\{<1w0;0/1/->\}$ with FP #9, $\mathrm{CFds}_{1w0;1}=\{<1w0;1/0/->\}$ with FP #10, $\mathrm{CFds}_{0r0;0}=\{<0r0;0/1/->\}$ with FP #13, $\mathrm{CFds}_{0r0;1}=\{<0r0;1/0/->\}$ with FP #14, $\mathrm{CFds}_{1r1;0}=\{<1r1;0/1/->\}$ with FP #15, and $\mathrm{CFds}_{1r1;1}=\{<1r1;1/0/->\}$ with FP #16.
3. **Transition coupling fault (CFtr)** is a fault whereby a given logic value in the aggressor results in the failure of a transition write operation performed on the victim. This fault is sensitized by a write operation on the victim and setting the aggressor into a given state. Four types of CFtr exist: $\mathrm{CFtr}_{0;\uparrow}=\{<0;0w1/0/->\}$ with FP #19, $\mathrm{CFtr}_{0;\downarrow}=\{<0;1w0/1/->\}$ with

TABLE 2
All Possible Single-Cell 2-Operation FPs

#	S	F	R	FP	#	S	F	R	FP
1	$0w0w0$	1	–	$<0w0w0/1/->$	2	$0w0w1$	0	–	$<0w0w1/0/->$
3	$0w0r0$	0	1	$<\mathbf{0w0r0/0/1}>$	4	$0w0r0$	1	0	$<\mathbf{0w0r0/1/0}>$
5	$0w0r0$	1	1	$<\mathbf{0w0r0/1/1}>$					
6	$0w1w0$	1	–	$<0w1w0/1/->$	7	$0w1w1$	0	–	$<0w1w1/0/->$
8	$0w1r1$	0	0	$<\mathbf{0w1r1/0/0}>$	9	$0w1r1$	0	1	$<\mathbf{0w1r1/0/1}>$
10	$0w1r1$	1	0	$<\mathbf{0w1r1/1/0}>$					
11	$1w0w0$	1	–	$<1w0w0/1/->$	12	$1w0w1$	0	–	$<1w0w1/0/->$
13	$1w0r0$	0	1	$<\mathbf{1w0r0/0/1}>$	14	$1w0r0$	1	0	$<\mathbf{1w0r0/1/0}>$
15	$1w0r0$	1	1	$<\mathbf{1w0r0/1/1}>$					
16	$1w1w0$	1	–	$<1w1w0/1/->$	17	$1w1w1$	0	–	$<1w1w1/0/->$
18	$1w1r1$	0	0	$<\mathbf{1w1r1/0/0}>$	19	$1w1r1$	0	1	$<\mathbf{1w1r1/0/1}>$
20	$1w1r1$	1	0	$<\mathbf{1w1r1/1/0}>$					
21	$0r0w0$	1	–	$<0r0w0/1/->$	22	$0r0w1$	0	–	$<0r0w1/0/->$
23	$0r0r0$	0	1	$<0r0r0/0/1>$	24	$0r0r0$	1	0	$<0r0r0/1/0>$
25	$0r0r0$	1	1	$<0r0r0/1/1>$					
26	$1r1w0$	1	–	$<1r1w0/1/->$	27	$1r1w1$	0	–	$<1r1w1/0/->$
28	$1r1r1$	0	0	$<1r1r1/0/0>$	29	$1r1r1$	0	1	$<1r1r1/0/1>$
30	$1r1r1$	1	0	$<1r1r1/1/0>$					

FP #21, $\mathrm{CFtr}_{1;\uparrow} = \{<1;0w1/0/->\}$ with FP #20, and $\mathrm{CFtr}_{1;\downarrow} = \{<1;1w0/1/->\}$ with FP #22.

4. **Write disturb coupling fault (CFwd)** is a fault whereby a nontransition write operation performed on the victim results in a transition when the aggressor is set into a given logic state. Four types of CFwd exist: $\mathrm{CFwd}_{0;0} = \{<0;0w0/1/->\}$ with FP #17, $\mathrm{CFwd}_{0;1} = \{<0;1w1/0/->\}$ with FP #23, $\mathrm{CFwd}_{1;0} = \{<1;0w0/1/->\}$ with FP #18, and $\mathrm{CFwd}_{1;1} = \{<1;1w1/0/->\}$ with FP #24.
5. **Read disturb coupling fault (CFrd)** is a fault whereby a read operation performed on the victim destroys the data stored in the victim if a given state is present in the aggressor. Four types of CFrd exist: $\mathrm{CFrd}_{0;0} = \{<0;0r0/1/1>\}$ with FP #29, $\mathrm{CFrd}_{0;1} = \{<0;1r1/0/0>\}$ with FP #31, $\mathrm{CFrd}_{1;0} = \{<1;0r0/1/1>\}$ with FP #30, and $\mathrm{CFrd}_{1;1} = \{<1;1r1/0/0>\}$ with FP #32.
6. **Incorrect read coupling fault (CFir)** is a fault whereby a read operation performed on the victim returns the incorrect logic value when the aggressor is set into a given state. Four types of CFir exist: $\mathrm{CFir}_{0;0} = \{<0;0r0/0/1>\}$ with FP #25, $\mathrm{CFir}_{0;1} = \{<0;1r1/1/0>\}$ with FP #35, $\mathrm{CFir}_{1;0} = \{<1;0r0/0/1>\}$ with FP #26, and $\mathrm{CFir}_{1;1} = \{<1;1r1/1/0>\}$ with FP #36.
7. **Deceptive read disturb coupling fault (CFdr)** is a fault whereby a read operation performed on the victim returns the correct logic value and changes the contents of the victim, when the aggressor is set into a given logic state. Four types of CFdr exist: $\mathrm{CFdr}_{0;0} = \{<0;0r0/1/0>\}$ with FP #27, $\mathrm{CFdr}_{0;1} = \{<0;1r1/0/1>\}$ with FP #33, $\mathrm{CFdr}_{1;0} = \{<1;0r0/1/0>\}$ with FP #28, and $\mathrm{CFdr}_{1;1} = \{<1;1r1/0/1>\}$ with FP #34.

3.5 Dynamic Two-Cell FFMs

Just like the case of single-cell dynamic FFMs, we restrict ourselves here to the analysis of 2-operation dynamic fault models. Any particular FP is denoted by $<S/F/R>$, where S has the form given in Section 3.1. For example, the two-cell 2-operation FP $<v(0r0)\ a(1r1)/1/->$ stands for an FP sensitized by performing a $0r0$ first on the victim, then performing a $1r1$ on the aggressor. After performing the sensitizing sequence, a 1 is detected in the victim cell instead of the expected 0. Based on the values of S, F, and R, 192 detectable two-cell 2-operation dynamic FPs can be compiled [7].

TABLE 3
All Possible Two-Cell Static FPs

#	S_a	S_v	F	R	$< S_a; S_v/F/R >$	#	S_a	S_v	F	R	$< S_a; S_v/F/R >$
1	0	0	1	–	$< 0; 0/1/- >$	2	0	1	0	–	$< 0; 1/0/- >$
3	1	0	1	–	$< 1; 0/1/- >$	4	1	1	0	–	$< 1; 1/0/- >$
5	$0w0$	0	1	–	$< 0w0; 0/1/- >$	6	$0w0$	1	0	–	$< 0w0; 1/0/- >$
7	$0w1$	0	1	–	$< 0w1; 0/1/- >$	8	$0w1$	1	0	–	$< 0w1; 1/0/- >$
9	$1w0$	0	1	–	$< 1w0; 0/1/- >$	10	$1w0$	1	0	–	$< 1w0; 1/0/- >$
11	$1w1$	0	1	–	$< 1w1; 0/1/- >$	12	$1w1$	1	0	–	$< 1w1; 1/0/- >$
13	$0r0$	0	1	–	$< 0r0; 0/1/- >$	14	$0r0$	1	0	–	$< 0r0; 1/0/- >$
15	$1r1$	0	1	–	$< 1r1; 0/1/- >$	16	$1r1$	1	0	–	$< 1r1; 1/0/- >$
17	0	$0w0$	1	–	$< 0; 0w0/1/- >$	18	1	$0w0$	1	–	$< 1; 0w0/1/- >$
19	0	$0w1$	0	–	$< 0; 0w1/0/- >$	20	1	$0w1$	0	–	$< 1; 0w1/0/- >$
21	0	$1w0$	1	–	$< 0; 1w0/1/- >$	22	1	$1w0$	1	–	$< 1; 1w0/1/- >$
23	0	$1w1$	0	–	$< 0; 1w1/0/- >$	24	1	$1w1$	0	–	$< 1; 1w1/0/- >$
25	0	$0r0$	0	1	$< 0; 0r0/0/1 >$	26	1	$0r0$	0	1	$< 1; 0r0/0/1 >$
27	0	$0r0$	1	0	$< 0; 0r0/1/0 >$	28	1	$0r0$	1	0	$< 1; 0r0/1/0 >$
29	0	$0r0$	1	1	$< 0; 0r0/1/1 >$	30	1	$0r0$	1	1	$< 1; 0r0/1/1 >$
31	0	$1r1$	0	0	$< 0; 1r1/0/0 >$	32	1	$1r1$	0	0	$< 1; 1r1/0/0 >$
33	0	$1r1$	0	1	$< 0; 1r1/0/1 >$	34	1	$1r1$	0	1	$< 1; 1r1/0/1 >$
35	0	$1r1$	1	0	$< 0; 1r1/1/0 >$	36	1	$1r1$	1	0	$< 1; 1r1/1/0 >$

Since we will only attempt to verify a limited number of dynamic FFMs for the reasons mentioned in Section 3.3, only those FPs with an SOS of the form $S = a(x)\ v(ywzrz)$ are targeted, where x, y, and $z \in \{0, 1\}$. This choice of S results in 24 different FPs targeted by the performed simulation. Since these FPs have not been observed, they are not used to define corresponding FFMs here. Such FFM definitions can be found in a previously published paper [7].

4 Simulated Spot Defects

In this section, we discuss the spot defects to be considered for injection and simulation in the *e*DRAM model. The defects are first classified, then the location of each of them is shown on the simulated *e*DRAM model.

4.1 Specifying the Defects

The spot defects to be considered for injection and analysis are modeled at the electrical level by parasitic components with a given impedance. The impedance (Z) consists of a resistance (R) and a capacitance (C) connected in parallel between two defective nodes. Depending on the defective nodes the injected defects are connected to, the defects may be classified into opens, shorts, and bridges. The list of considered spot defects is meant to be comprehensive and it is not related to a specific memory layout. If only defects realistic to a given memory layout are needed, inductive fault analysis techniques can be used to extract the most probable defects [10].

Opens represent unwanted impedances on a signal line that is supposed to conduct perfectly. For an open defect, the impedance value is given by Z_{op} and is predominantly resistive (i.e., $C_{op} \approx 0$, making $Z_{op} \approx R_{op}$). The open resistance may take any value in the resistance domain, which gives $0 \leq Z_{op} \leq \infty\ \Omega$. The fact that opens result in negligible capacitive coupling between the broken nodes has been substantiated by Henderson et al. [11].

Shorts represent unwanted impedances between a signal line and V_{DD} or GND. For a short, the impedance value is denoted by Z_{sh} and may have resistive and capacitive components. The value of R_{sh} for a short may again have any value ($0 \leq R_{sh} \leq \infty\ \Omega$), while C_{sh} is bounded by some given realistic limits ($C_{min} < C_{sh} < C_{max}$). The lower bound of the short capacitance is taken to be 0 F ($C_{min} = 0$ F), while the maximum bound is considered to be equal to the bit line capacitance in the memory ($C_{max} = C_b$). The reason behind this choice is that the bit line has the highest capacitance along the data path of a single cell array column. Therefore, it is highly unexpected for a parasitic capacitance to have yet a higher value.

Bridges represent unwanted impedances between two signal lines. Bridges between signal lines and V_{DD} or GND are not considered as bridges since these are covered by

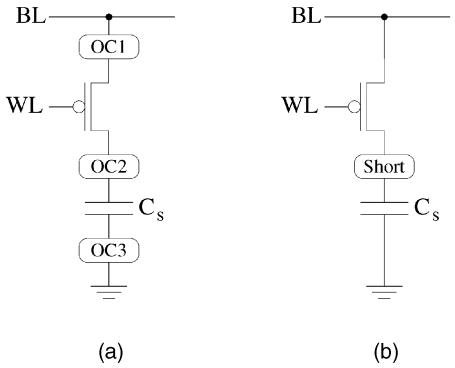


Fig. 4. Memory cell with possible locations of (a) opens and (b) shorts.

shorts. The electrical specifications of bridges are the same as those for shorts.

4.2 Classification of Defects

By analyzing the electrical circuits of the cell array column, we notice some symmetry in the topology of these circuits. This results in a symmetry in the faulty behavior, which can be used to reduce the number of defects to be simulated and analyzed. The faulty behavior of one defect can help deduce the faulty behavior of another symmetrically related defect. For this purpose, we provide the following definitions:

- A defect D1 at a given position shows the **complementary faulty behavior** of a defect D2 at another position, if the faulty behavior of D1 is the same as that of D2, with the only difference that all 1s are replaced by 0s and vice versa. For example, if D1 sensitizes $< 0r0/1/1 >$, then D2 sensitizes $< 1r1/0/0 >$.
- A defect D1 shows the **exchanged faulty behavior** of a defect D2, if the faulty behavior of D1 and D2 contain two-cell faults and if these two-cell faults are the same with the exception that the aggressor and victim are exchanged. In general, if a two-cell fault has the following notation $< S/F/R >_{x,y}$, then the exchanged fault is given by the notation $< S/F/R >_{y,x}$.
- A defect D1 shows a **single-sided complementary behavior** of a defect D2, if the faulty behavior of D1 and D2 contain two-cell faults and if these two-cell faults are the same with the exception that all 1s are replaced by 0s and vice versa, in either the aggressor or the victim cells (not both). If the victim sides of two faults are the complement of each other, then these two faults are called *victim-sided complementary*. If the aggressor sides of two faults are the complement of each other, then these two faults are called *aggressor-sided complementary*. For example, suppose that defects D1, D2, and D3 affect cells x and y and that D1 forces a $0w0$ operation to cause an up transition in y if cell x is in state 1, then this faulty behavior of D1 is denoted by $< 1; 0w0/1/- >_{x,y}$. The aggressor-sided complementary defect D2 should force a $0w0$ operation to cause an up transition in y if cell x is in state 0, which is the fault denoted by $< 0; 0w0/1/- >_{x,y}$. On the other hand, the victim-sided complementary defect D3 should force a $1w1$ operation to cause a down transition in y if cell x is in state 1, which is the fault denoted by $< 1; 1w1/0/- >_{x,y}$.

It is important to note that the exchanged behavior classification scheme is independent from the complementary and the single-sided complementary classification schemes. This means that it is possible to have a defect classified to be exchanged only, exchanged complementary, exchanged single-sided, etc. On the other hand, the complementary and the single-sided complementary schemes *are*, in fact, related to each other. The complementary behavior of a defect is the same as the combination of the aggressor-sided and the victim-sided complementary behavior of a defect.

4.3 Locations of Opens

The possible locations of opens within memory cells (OC), along bit lines (OB), and on word lines (OW) are enumerated and provided with a label for future reference.

Opens within a memory cell (OC) can occur at any node within the storage cell. Fig. 4a shows one memory cell where the three possible defect locations are indicated. The choice has been made to simulate the opens within a cell on the true bit line (BT) and these defects are therefore labeled as OCxs (where $1 \leq x \leq 3$ and "s" stands for *simulated*, see Table 4). Consequently, the faulty behavior of an open in a cell on the complement bit line (BC), which is labeled as OCxc ("c" for complementary), may be derived from the corresponding simulated one because it shows the complementary faulty behavior.

Opens along a bit line (OB) can occur anywhere on the bit line. Fig. 5 shows a complete cell array column with BT and BC together with the bit line opens. The bit lines are divided into 10 regions, each of which may contain an open. Every open on BT has its complementary open on BC and

TABLE 4
Simulated and Complementary Opens within a Cell

OC on BT	OC on BC	Description
OC1s	OC1c	Pass transistor connection to bit line broken
OC2s	OC2c	Pass transistor connection to storage capacitor broken
OC3s	OC3c	Cell connection to ground broken

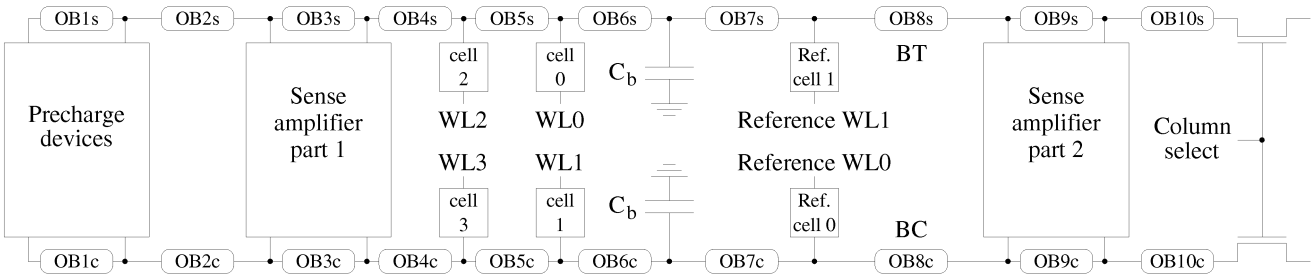


Fig. 5. Cell array column with BT and BC on which possible locations of opens are indicated.

vice versa. Thus, only opens present on BT are simulated. Every defect on BT is given the name OBxs, while its counterpart on BC is given the name OBxc.

Opens on a word line (OW) can be at only one position, between the row decoder and the gate of the pass transistor of a memory cell. The behavior of the cell with an open on its word line is the same for every cell on BT and complementary to that on BC. Therefore, only one open is simulated, namely that on WL0, which is called OW1s. The open located on WL1 is called OW1c.

4.4 Locations of Shorts

The possible locations of shorts within memory cells (SC) and along bit lines (SB) are enumerated and provided with a label for future reference. Shorts on word lines are not simulated since the reduced model drives word lines directly by voltage sources. Shorts between V_{DD} and ground are not included since they are power shorts, which do not belong to the class of memory cell array shorts.

Shorts within a memory cell (SC) can be injected within the storage cell at only one node between the pass transistor and the storage capacitor, as shown in Fig. 4b. This gives two possible shorts: SC1, which is a connection between the cell and V_{DD}, and SC2, which is a connection between the cell and GND. Every short in a cell on BT has its complementary short in a cell on BC and vice versa. Thus, only shorts in cells on BT are simulated. Shorts in cells on BT are called SCxs, while their counterparts on BC are called SCxc.

Shorts along a bit line (SB) can connect BT or BC to either V_{DD} or GND. A bit line short to V_{DD} is called SB1, while a bit line short to GND is called SB2. Every short on BT has its complementary short on BC and vice versa. Thus, only shorts along BT are simulated. Shorts on BT are called SBxs, while their counterparts on BC are called SBxc.

4.5 Location of Bridges

A bridge in the memory cell array can connect any arbitrary pair of nodes. However, not all possible bridges in the cell array have been simulated, but only those that take place within a single cell or between different cells. This choice is motivated by the fact that memory cells take the largest part of the surface area in a dynamic RAM. Bridges connecting circuit nodes to V_{DD} or GND are excluded; these are considered shorts (see Section 4.4).

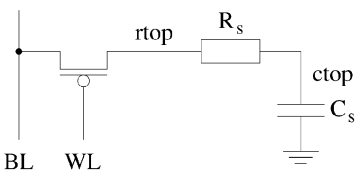


Fig. 6. Memory cell in which the nodes are given names.

Bridge within a memory cell (BWC) can connect any node of the cell to any other node. In Fig. 6, the different nodes in the cell are given names. There are four nodes in the cell with the names BL,[2] WL, rtop, and ctop. Here, bridges between word and bit lines are excluded. The choice is made to simulate the bridges of a cell on BT. As a result, the behavior of a BWC defect in a cell on the complement bit line may be derived from the corresponding simulated one since these two defects show a complementary behavior. A list of the simulated and complementary BWC defects is given in Table 5.

Bridge between two memory cells (BBC) can connect any node in one cell (on BT or BC) to any other node in any other cell. In Fig. 6, the different nodes in the cell are given names. There are four nodes in the cell with the names BL, WL, rtop, and ctop. A number between 1 and 4 is added to every node name to indicate the cell each node belongs to (see Fig. 5). Here, bridges among word lines, among bit lines, and between word lines and bit lines are excluded. Bridges between bit lines and cells are considered as bridges within cells. A list of the BBC defects is given in Table 6. The table classifies BBCs into six classes: simulated, complementary, exchanged, exchanged complementary, aggressor-sided complementary, and exchanged aggressor-sided complementary defects. It has been chosen to take cell 0 to be the victim for simulated BBC defects and cell 1 to be the victim for complementary BBC defects. BBC1, for example, has six bridge defect instances listed in the table. BBC1s is the only defect simulated and connects WL in cell 2 (aggressor) and rtop in cell 0 (victim), where both cells are on BT. BBC1c is the complementary of BBC1s since it connects the equivalent nodes in cells on BC. BBC1e is the exchanged of BBC1s since it swaps the nodes of BBC1s. BBC1ec is the exchanged complementary of BBC1s since it connects the equivalent nodes of BBC1s in cells on BC and swaps them. BBC1a is the aggressor-sided complementary of BBC1s since it replaces the node of the aggressor (WL2) with an equivalent node on BC (WL1). Finally, BBC1ea is the exchanged aggressor-sided complementary of BBC1s

2. BL means BT if the cell is connected to the true bit line, while it means BC if the cell is connected to the complementary bit line.

TABLE 5
Simulated and Complementary Bridges within a Cell (BWC)

BWC on BT	BWC on BC	Description
BWC1s	BWC1c	Bridge between bit line and rtop
BWC2s	BWC2c	Bridge between bit line and ctop
BWC3s	BWC3c	Bridge between word line and rtop
BWC4s	BWC4c	Bridge between word line and ctop
BWC5s	BWC5c	Bridge between ctop and rtop

since it swaps the nodes of the aggressor-sided complementary of BBC1s.

5 Simulation Methodology

This section discusses the electrical level simulation employed to establish FFMs caused by opens, shorts, and bridges.

5.1 Simulation of Opens

The behavior of the *e*DRAM is studied after injecting and simulating each of the opens defined in Section 4.3. The analysis considers open resistances within the range ($10\ \Omega \leq R_{op} \leq 10\ \mathrm{M}\Omega$) on a logarithmic scale using five points per decade, in addition to $R_{op} = \infty\ \Omega$. Each injected open in the memory model creates floating nodes, the voltage of which is varied between V_{DD} and GND on a linear scale using 10 points. When an interesting faulty behavior is observed, more detailed simulations are performed. Determining the floating node resulting from each injected open depends on the type of the open. For opens along bit lines, the floating node is always taken to be the one connected to column access devices, not the one connected to the precharge devices (see Fig. 5) since this node is precharged to a known voltage at the beginning of each operation. The floating node for opens within memory cells is taken to be the node connected to the cell capacitor. For opens on word lines, the floating node is the node connected to the memory cell.

Although, during normal memory operation, some floating nodes do not assume all considered U_{init} values in the range from V_{dd} to GND, it is still important to take the whole range into consideration. On the one hand, this gives the most restrictive conditions for performing the fault analysis which ensures generating memory tests capable of detecting any possible faulty behavior. On the other hand, signal line voltages in memories may vary in unexpected ways (at memory start-up or because of supply voltage fluctuations, for example), which may set floating memory nodes to some unexpected values.

For each value of the open resistance (R_{op}) and of the initial floating node voltage (U_{init}), all the SOSs associated with the targeted FPs defined in Section 3 are performed and inspected for proper functionality. As a result, the faulty behavior resulting from the analysis of opens is represented as regions in the (U_{init}, R_{op}) plane. Each region

TABLE 6
List of Possible Bridge Defects between Memory Cells (BBC)

Simulated BBC	Complementary BBC	Exchanged BBC	Exchanged complementary BBC	Aggressor-sided complementary BBC	Exchanged aggressor-sided BBC
BBC1s WL2-rtop0	BBC1c WL3-rtop1	BBC1e WL0-rtop2	BBC1ec WL1-rtop3	BBC1a WL1-rtop0	BBC1ea WL0-rtop1
BBC2s WL2-ctop0	BBC2c WL3-ctop1	BBC2e WL0-ctop2	BBC2ec WL1-ctop3	BBC2a WL1-ctop0	BBC2ea WL0-ctop1
BBC3s rtop0-rtop2	BBC3c rtop1-rtop3			BBC3a rtop0-rtop1	
BBC4s rtop0-ctop2	BBC4c rtop1-ctop3	BBC4e rtop2-ctop0	BBC4ec rtop3-ctop1	BBC4a rtop0-ctop1	BBC4ea rtop1-ctop0
BBC5s ctop0-rtop2	BBC5c ctop1-rtop3	BBC5e ctop2-rtop0	BBC5ec ctop3-rtop1	BBC5a ctop0-rtop1	BBC5ea ctop1-rtop0
BBC6s ctop0-ctop2	BBC6c ctop1-ctop3			BBC6a ctop0-ctop1	
BBC7s WL0-rtop2	BBC7c WL1-rtop3	BBC7e WL2-rtop0	BBC7ec WL3-rtop1	BBC7a WL0-rtop1	BBC7ea WL1-rtop0
BBC8s WL0-ctop2	BBC8c WL1-ctop3	BBC8e WL2-ctop0	BBC8ec WL3-ctop1	BBC8a WL0-ctop1	BBC8ea WL1-ctop0

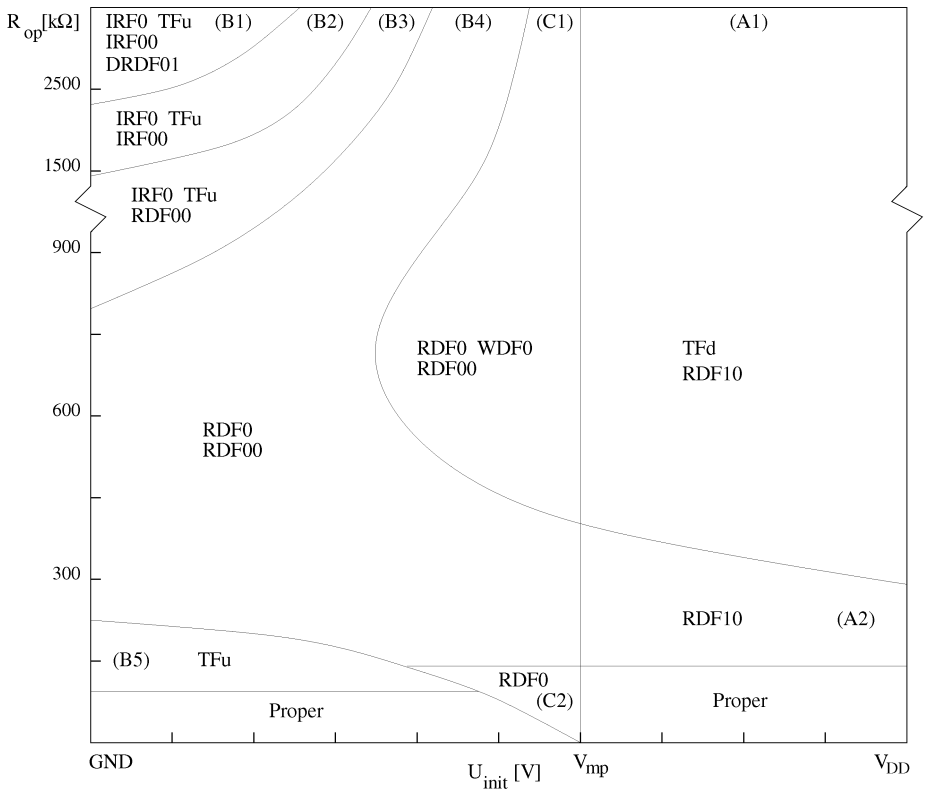


Fig. 7. Summary of the fault analysis results of the defect OC1s in the (U_{init}, R_{op}) plane under $T = 27^{\circ}$ C.

contains a number of sensitized FPs that describe the FFM of the memory in this region.

As an example, the results of the fault analysis performed on OC1s (see Fig. 4a) are given in Fig. 7, which shows the observed faulty behavior in the (U_{init}, R_{op}) plane. In the figure, TFd stands for TF $\downarrow$, TFu stands for TF $\uparrow$, while V_{mp} is the mid-point voltage (threshold voltage between logic 0 and logic 1). The figure shows a number of different fault regions for different combinations of U_{init} and R_{op}. The fault regions may be classified according to the initial floating node voltage under which they can be detected as follows:

1. Faults detectable with $U_{init} = V_{DD}$:
   - 1a. Fault region TF $\downarrow \cup$RDF$_{10}$.
   - 1b. Fault region RDF$_{10}$.
2. Faults detectable with $U_{init} =$ GND:
   - 2a. Fault region IRF$_0 \cup$ TF $\uparrow \cup$IRF$_{00} \cup$ DRDF$_{01}$.
   - 2b. Fault region IRF$_0 \cup$ TF $\uparrow \cup$IRF$_{00}$.
   - 2c. Fault region IRF$_0 \cup$ TF $\uparrow \cup$RDF$_{00}$.
   - 2d. Fault region RDF$_0 \cup$ RDF$_{00}$.
   - 2e. Fault region TF $\uparrow$.
3. Faults only detectable with GND $< U_{init} < V_{DD}$:
   - 3a. Fault region RDF$_0 \cup$ WDF$_0 \cup$ RDF$_{00}$.
   - 3b. Fault region RDF$_0$.

Each fault region in the figure contains a number of FPs, each of which describes a failing SOS with the associated faulty behavior. If a region contains more that one FP, it means that more than one SOS is failing at the same time. As a result, if a test performs *any* of the failing SOSs in a given fault region, then the test sensitizes a fault in that region. In Region A1, for example, two SOSs fail ($1w0$ and $1w0r0$) resulting in the FFMs TF $\downarrow = \{< 1w0/1/- >\}$ and RDF$_{10} = \{< 1w0r0/1/1 >\}$. The $1w0$ operation fails since the open within the memory cell partially disconnects the cell from the bit line, which prevents discharging the stored 1 to a 0. Subsequently, the sequence $1w0r0$, which concatenates a read to the failing $1w0$ operation, would also fail due to detecting a 1 instead of the desired 0.

If we fix U_{init} at V_{DD}, then, as R_{op} decreases below about 300 kΩ, Region A2 begins where only one FFM is present (RDF$_{10} = \{< 1w0r0/1/1 >\}$). This means that, in this region, the operation $1w0$ functions correctly, leaving only $1w0r0$ failing. The reason for this behavior is that, as the $1w0$ operation succeeds in setting a 0 into the cell, this 0 starts out very weak. With the presence of the open, trying to read this weak 0 detects a faulty 1 since the cell is not able to sufficiently discharge the bit line for the sense amplifier to detect a 0. If R_{op} falls further below 150 kΩ, all SOSs become proper since the written 0 becomes strong enough for the read in $1w0r0$ to detect.

Inspecting the faulty behavior shown in the figure reveals that Region C1 and Region C2 have FFMs that are not detectable at a U_{init} equal to either V_{DD} or GND. Moreover, Region C1 contains the FFM WDF$_0$, which cannot be detected in any other fault region. This indicates that performing the fault analysis with all possible U_{init} values is important such that all sensitized FFMs resulting from a given defect are to be established.

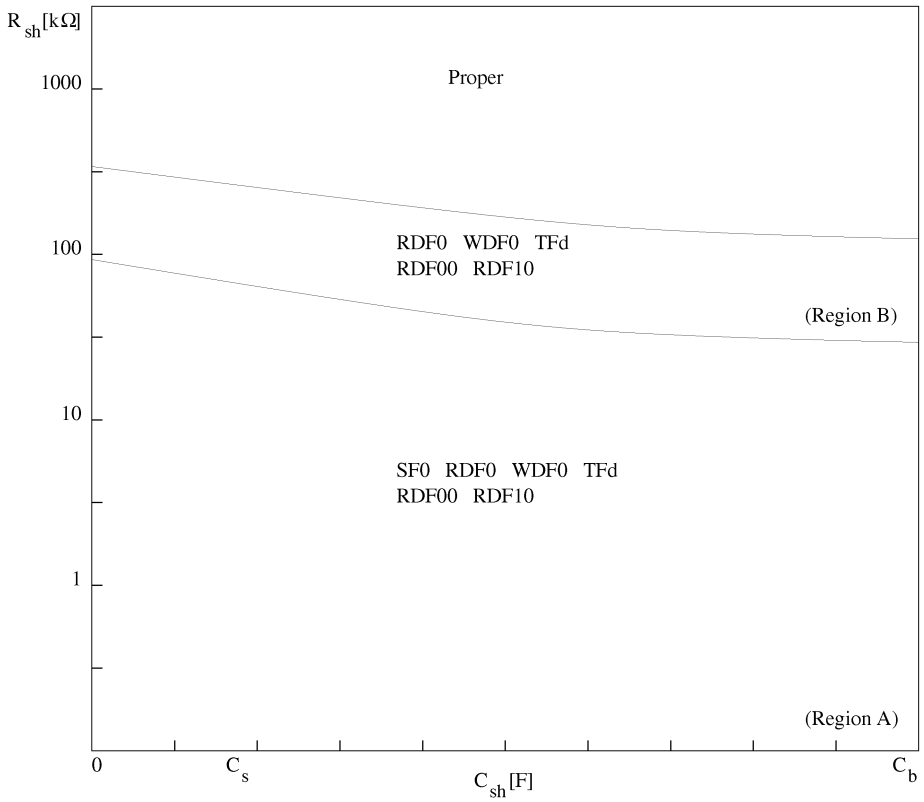


Fig. 8. Summary of the fault analysis results for the SCIs short to V_{DD} in the (C_{sh}, R_{sh}) plane under $T = 27^{\circ}$ C.

Region A2 only contains the FFM RDF_{10}, which means that $1w0r0$ is the only failing SOS in this region. This, in turn, means that performing the traditional *static* analysis on this fault region reveals no improper memory behavior. Only by applying *dynamic* SOSs is it possible to detect this improper behavior. This shows the significance of performing the dynamic analysis on memory devices. Strictly dynamic faults are used in Section 7 to generate new detection conditions and tests since no tests have yet been proposed for them. To show that strictly dynamic regions are realistic and take place for a large range of R_{op} values, a list of all strictly dynamic fault regions observed in the analysis of opens is given in Section 6.1, along with their corresponding R_{op} range.

5.2 Simulation of Shorts

The behavior of the *e*DRAM is studied after injecting and simulating each of the shorts defined in Section 4.4. The short impedance can have a resistive and a capacitive component. The short resistor and short capacitor are connected in parallel between the defective node and the power supply. The short resistance (R_{sh}) is varied in the same way as R_{op}, while the short capacitance (C_{sh}) is varied between 0 F and C_b on a linear scale using 10 points.

For each value of R_{sh} and C_{sh}, all the SOSs associated with the *targeted* FPs defined in Section 3 are performed and inspected for proper functionality. As a result, the faulty behavior resulting from the analysis of shorts is represented as regions in the (C_{sh}, R_{sh}) plain.

As an example, the results of the fault analysis performed on SC1s (short between memory cell and V_{DD}, see Fig. 4b) are shown in Fig. 8, where TFd stands for $\mathrm{TF}\downarrow$. According to the figure, the faulty behavior of SC1s depends more on R_{sh} than on C_{sh}. There are two fault regions shown in the figure, listed next with increasing R_{sh} value.

1. Fault region
$$\mathrm{SF}_0 \cup \mathrm{RDF}_0 \cup \mathrm{WDF}_0 \cup \mathrm{TF}\downarrow \cup \mathrm{RDF}_{00} \cup \mathrm{RDF}_{10}.$$
2. Fault region $\mathrm{RDF}_0 \cup \mathrm{WDF}_0 \cup \mathrm{TF}\downarrow \cup \mathrm{RDF}_{00} \cup \mathrm{RDF}_{10}$.

Fig. 8 shows a relatively simple faulty behavior of the short defect SC1s with only two fault regions. Region A has six FPs that describe faults in either writing or reading a 0 from the cell as a result of the short to V_{DD}. In this region, it is not possible to initialize the cell 0 (SF_0), to read a 0 (RDF_0), nor to write a 0 (WDF_0 & $\mathrm{TF}\downarrow$). The combination of these four FPs mean that a cell has a stuck-at 1 fault in this region. As R_{sh} increases, it gradually takes more time to charge up the cell and destroy a stored 0. When $R_{sh} > 90$ kΩ (with $C_{sh} = 0$ F), it becomes possible to initialize the cell to 0; however, it is still not possible to read or write a 0 into the cell. As the short resistance increases further, the number of failing SOSs decreases until the memory functions properly with $R_{sh} > 400$ kΩ when $C_{sh} = 0$ F.

The faulty behavior is, to a large extent, independent of the value of the short capacitance. Nevertheless, as the defect capacitance increases, the regions of faulty behavior decrease slightly in size, while the size of the region of proper operation increases. This can be explained by the fact that, for this defect, the defect capacitance supports the

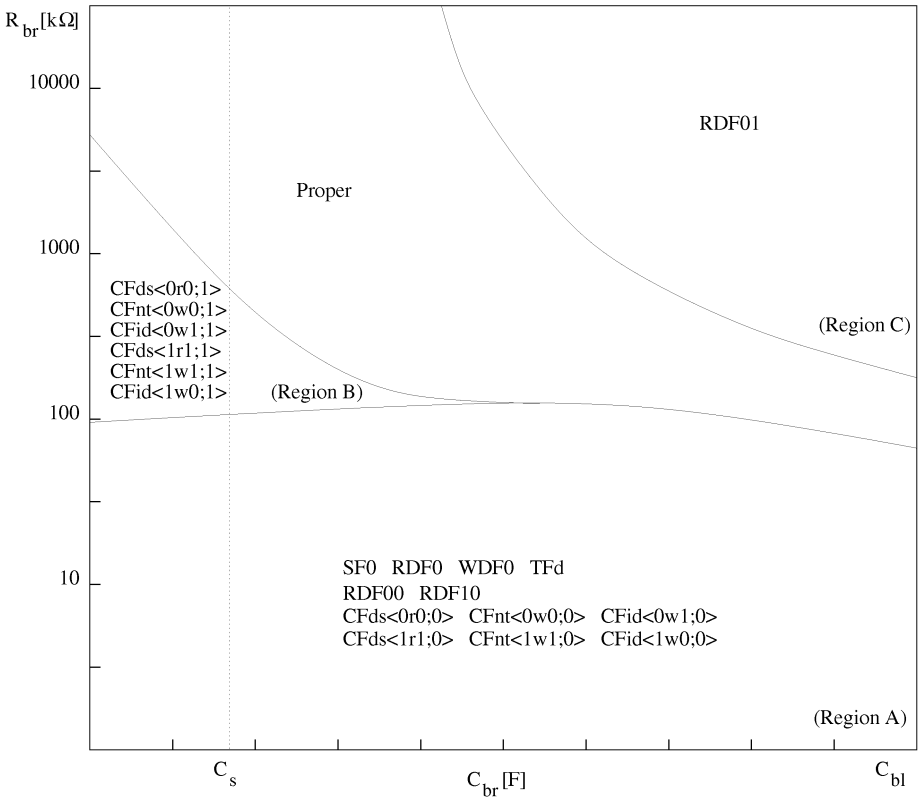


Fig. 9. Summary of the fault analysis results for the BBC1s bridge on the (C_{br}, R_{br}) plane under $T = 27°C$.

cell capacitance in storing the cell voltage, which means that an increasing defect capacitance stabilizes the stored voltage and reduces the ability of the resistive short to modify the stored voltage.

5.3 Simulation of Bridges

The behavior of the *e*DRAM is studied after injecting and simulating each of the bridges defined in Section 4.5. The bridge impedance has a resistive and a capacitive component that are connected in parallel between the two defective nodes. The bridge resistance (R_{br}) and bridge capacitance (C_{br}) are varied in the same way as R_{sh} and C_{sh}, respectively. The bridge capacitor is initialized to a voltage consistent with the initial voltages of the defective nodes.

For each value of R_{br} and C_{br}, all the SOSs associated with the *targeted* FPs defined in Section 3 are performed and inspected for proper functionality. As a result, the faulty behavior resulting from the analysis of bridges is represented as regions on the (C_{br}, R_{br}) plain. In the following, an example of the simulation results of a bridge between cells is shown first, followed by the results of a bridge within a cell. The simulation results given next are a bit simplified so that insight into the shown figures is not lost.

5.3.1 Bridge between Cells

The results of the fault analysis performed on BBC1s (bridge between word line and rtop nodes of two memory cells) are shown in Fig. 9. According to the figure, the faulty behavior of BBC1s depends on both R_{br} and C_{br}. There are three fault regions shown in the figure, listed next with increasing R_{br} value.

1. Fault region

$$\mathrm{SF}_0 \cup \mathrm{RDF}_0 \cup \mathrm{WDF}_0 \cup \mathrm{TF}\downarrow \cup \mathrm{RDF}_{00} \cup \mathrm{RDF}_{10} \cup \mathrm{CFds}_{0r0;0} \cup \mathrm{CFnt}_{0w0;0} \cup \mathrm{CFid}_{0w1;0} \cup \mathrm{CFds}_{1r1;0} \cup \mathrm{CFnt}_{1w1;0} \cup \mathrm{CFid}_{1w0;0}.$$

2. Fault region

$$\mathrm{CFds}_{0r0;1} \cup \mathrm{CFnt}_{0w0;1} \cup \mathrm{CFid}_{0w1;1} \cup \mathrm{CFds}_{1r1;1} \cup \mathrm{CFnt}_{1w1;1} \cup \mathrm{CFid}_{1w0;1}.$$

3. Fault region RDF_{01}.

Fig. 9 shows a relatively simple faulty behavior of the bridge defect BBC1s as compared with other analyzed bridge defects [12]. Region A in the figure shows not only single-cell but also two-cell FFMs. The bridge connects an internal cell node to the word line of another cell. Since that word line has a high voltage during precharge and a low voltage during cell access, all FFMs in this region indicate the failure of the victim to retain a logic 0. As R_{br} increases above 100 kΩ (while $C_{br} = 0$ F), Region B starts where only two-cell FFMs are present. The faulty behavior in this region is, in a sense, the opposite to that in Region A since the FFMs indicate the failure of the victim to retain a logic 1. This can be explained by noting that increasing the bridge resistance increases the phase shift between the voltage on the aggressor WL and the stored victim charge. Therefore, when WL is driven low as an operation is performed on the

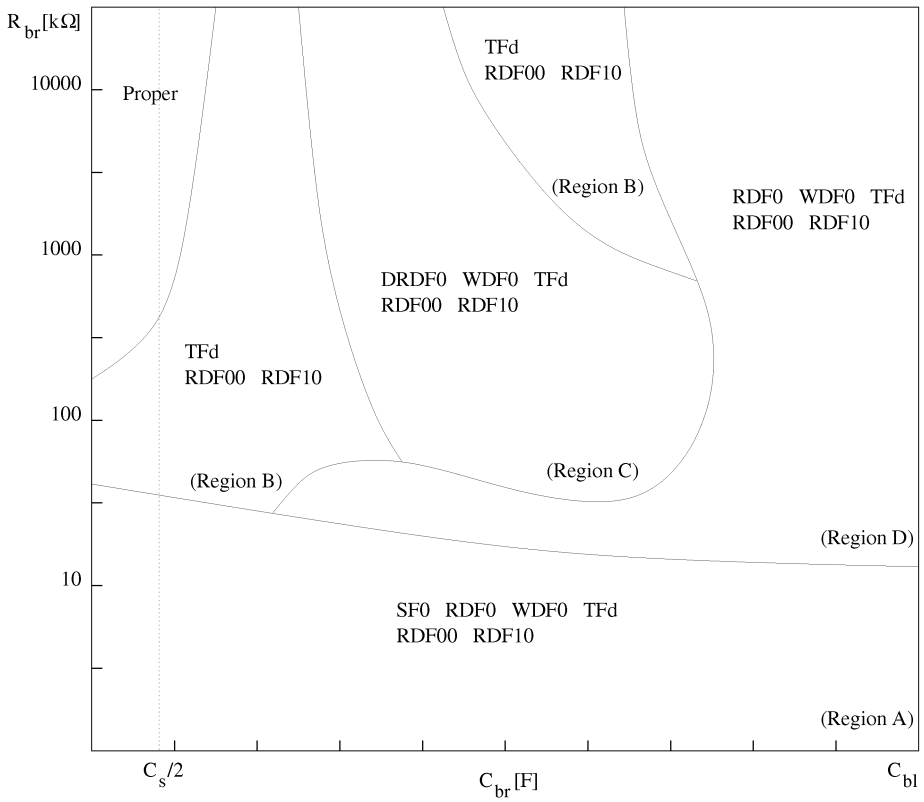


Fig. 10. Summary of the fault analysis results for BWC1s in the (C_{br}, R_{br}) plane under $T = 27^{\circ}$ C.

aggressor, the victim is discharged to 0 and keeps this logic 0 for a while after the aggressor operation ends. For $C_{br} = 0$ F and an increasing R_{br}, the number of failing SOSs decreases until the memory starts to function properly with $R_{br} > 5$ MΩ. On the other hand, the faulty behavior is also dependent on the value of C_{br} such that, for increasing C_{br}, the region of proper operation decreases gradually in size. Note that the faulty behavior changes gradually with no new fault regions appearing as long as $C_{br} < 2C_s$.

It is interesting to note that, for any C_{br} value, there is a R_{br} for which the memory behaves properly. In other words, despite the presence of a defect, given combinations of R_{br} and C_{br}, values can neutralize the faulty effect of the bridge and result in a properly operational memory (at least for the used SOSs).

Region C with RDF_{01} is an interesting region because it only shows dynamic faulty behavior. The SOS $0w1r1$ succeeds at first in writing a 1, but the subsequent read results in a faulty 0 on the output and leaves a stored 0 within the cell. This fault is mainly caused by the high value of C_{br}, which shares the voltage of a write 1 operation with the storage capacitor, resulting in storing a weak 1 into the cell. Note that this fault cannot be detected with the static SOSs $0w1$ and $1r1$ because, between the two SOSs, the BBC1s defect would charge the cell to 1. The fact that only a dynamic fault is sensitized means that, in order to detect the faulty behavior of this region, tests should be used that specifically target dynamic faults.

5.3.2 Bridge within a Cell

The results of the fault analysis performed on BWC1s (bridge between bit line and rtop node) are shown in Fig. 10 (TFd in the figure stands for TF $\downarrow$). According to the figure, the faulty behavior of BWC1s is a rather complex function of both C_{br} as well as R_{br}. There are four fault regions shown in the figure that are listed next with increasing R_{br} value.

1. Fault region
$$\mathrm{SF}_0 \cup \mathrm{RDF}_0 \cup \mathrm{WDF}_0 \cup \mathrm{TF}\downarrow \cup \mathrm{RDF}_{00} \cup \mathrm{RDF}_{10}.$$
2. Fault region $\mathrm{TF}\downarrow \cup \mathrm{RDF}_{00} \cup \mathrm{RDF}_{10}$.
3. Fault region
$$\mathrm{DRDF}_0 \cup \mathrm{WDF}_0 \cup \mathrm{TF}\downarrow \cup \mathrm{RDF}_{00} \cup \mathrm{RDF}_{10}.$$
4. Fault region
$$\mathrm{RDF}_0 \cup \mathrm{WDF}_0 \cup \mathrm{TF}\downarrow \cup \mathrm{RDF}_{00} \cup \mathrm{RDF}_{10}.$$

Fig. 10 shows that if $C_{br} = 0$ F and, for increasing R_{br}, the number of faulty SOSs decreases until the memory starts to function properly with $R_{br} > 200$ kΩ. As C_{br} increases, the region of proper operation decreases rapidly until it disappears for $C_{br} > C_s$. According to the figure, the faulty behavior of this bridge changes gradually as long as $C_{br} < \frac{C_s}{2}$; this result can be stated for all simulated bridges.

In Region A (approximately when $R_{br} < 10$ kΩ), there are six FFMs, all of which indicate that the cell is unable to

TABLE 7
Summary of the Observed FFMs for Each Open

Open	Simulated	Complementary
OC1–3	TF↑, TF↓, WDF_0, RDF_0, IRF_0, RDF_{00}, RDF_{01}, RDF_{10}, IRF_{00}, $DRDF_{01}$	TF↓, TF↑, WDF_1, RDF_1, IRF_1, RDF_{11}, RDF_{10}, RDF_{01}, IRF_{11}, $DRDF_{10}$
OB1–3	RDF_1	RDF_0
OB4–5	TF↑, WDF_1, RDF_1, $DRDF_1$, RDF_{01}, RDF_{11}	TF↓, WDF_0, RDF_0, $DRDF_0$, RDF_{10}, RDF_{00}
OB6	TF↓, WDF_0, RDF_0, RDF_1, IRF_1, RDF_{01}, RDF_{11}, IRF_{01}, IRF_{11}, $DRDF_{00}$, $DRDF_{10}$	TF↑, WDF_1, RDF_1, RDF_0, IRF_0, RDF_{10}, RDF_{00}, IRF_{10}, IRF_{00}, $DRDF_{11}$, $DRDF_{01}$
OB7–8	TF↓, WDF_0, RDF_1, IRF_1, $DRDF_0$, RDF_{01}, RDF_{11}, IRF_{01}, IRF_{11}, $DRDF_{00}$, $DRDF_{10}$	TF↑, WDF_1, RDF_0, IRF_0, $DRDF_1$, RDF_{10}, RDF_{00}, IRF_{10}, IRF_{00}, $DRDF_{11}$, $DRDF_{01}$
OB9	RDF_0, IRF_0, $DRDF_{00}$, $DRDF_{10}$	RDF_1, IRF_1, $DRDF_{11}$, $DRDF_{01}$
OB10	TF↓, IRF_0, RDF_{10}, $DRDF_{10}$	TF↑, IRF_1, RDF_{01}, $DRDF_{01}$
OW1	SF_0, TF↑, TF↓, RDF_0, IRF_0, RDF_{00}, RDF_{10}, IRF_{00}, IRF_{10}	SF_1, TF↑, TF↓, RDF_1, IRF_1, RDF_{11}, RDF_{01}, IRF_{11}, IRF_{01}

retain a stored 0. This is caused by the high precharge voltage of the bit lines, which also charges the cell up as a result of the bridge. As the bridge resistance increases, it takes more time for the defect to charge the cell up. Still, the presence of the bridge capacitance prevents the cells from proper functionality. For example, Regions B, C, and D all have problems with performing $1w0$. This can be explained by the presence of the high bridge capacitance since the write operation needs to discharge the cell as well as the bridge capacitance to store a proper 0 into the cell.

It is interesting to note that Region B appears twice in the figure, interrupted by the Region C. By holding R_{br} at 1 MΩ, for example, and increasing C_{br} gradually from 0 F, we leave the region of proper operation into Region B, then C, then back to B, and, finally, end with Region D. Region C has the same FFMs as Region B in addition to $DRDF_0$ and WDF_0. This means that *increasing* the defect capacitance can sometimes *reduce* the number of failing SOSs. A similar observation has been made for the bridge BBC1s (see Fig. 9).

6 Discussing Simulation Results

All opens, shorts, and bridges defined above have been injected, simulated, and analyzed. The analysis results of opens are organized in figures depicting parts of the (U_{init}, R_{op}) plane, while the analysis results of shorts and bridges are organized in figures depicting a part of the (C_{sh}, R_{sh}) plane [12]. In the following, the results of opens are discussed first, then the results of shorts and bridges.

6.1 Results of Opens

Table 7 gives a summary of the observed FFMs for each open defect within memory cells, along bit lines and on word lines [13]. The first column in the table specifies the analyzed defects (in case a number of defects sensitize the same FFMs, they are listed together), while the second and third columns list the FFMs detected for the simulated and complementary instances of these defects, respectively. Inspecting Table 7 reveals that all FFMs defined in Section 3 are present and result from at least one defect.

The table shows that all opens within cells (OC1–3) cause the same faulty behavior, while the behavior caused by opens on bit lines (OB1–10) changes significantly by changing the position of the open. Moreover, some opens can cause a faulty behavior more easily than others (i.e., with less open resistance). The most sensitive position for an open is that at OB7 (see Fig. 5), where an open with an open resistance as low as 200 Ω can result in a faulty memory behavior. The table shows that all defects cause static FFMs and that most opens result in 2-operation dynamic FFMs (with the exception of OB1, OB2, and OB3).

It is important, from a testing point of view, to state the fault regions that only show dynamic faulty behavior (and, therefore, have to be detected by tests for dynamic FPs) since testing these regions for static faulty behavior is not effective. Table 8 lists all strictly dynamic fault regions detected in the analysis. The first column states the open resulting in the fault region, the second column describes the faulty behavior of the dynamic fault region, and the third column gives information about the fault region in the (U_{init}, R_{op}) plane. According to the table, there are four 2-operation FFMs that appear in fault regions with strictly dynamic behavior; they are the four types of the dynamic read disturb fault (RDF_{xy}).

6.2 Results of shorts

Table 9 gives a summary of the observed FFMs for each short defect within memory cells and along bit lines [13]. The first column in the table lists the analyzed shorts, while the second and third columns list the FFMs detected for the simulated and complementary instances of these defects, respectively. Inspecting the table reveals that, contrary to opens, not all the FFMs defined in Section 3 result from short defects. The table shows that all shorts cause both static and 2-operation dynamic FFMs.

The only strictly dynamic fault region detected in the analysis of shorts is caused by SC2 [12]. The simulated

TABLE 8
Strictly Dynamic Fault Regions Resulting from Opens

Open	Simulated	Complementary	Region
OC1–3	RDF_{10}	RDF_{01}	$150\ k\Omega < R_{op} < 450\ k\Omega$
OB4–5	RDF_{01}	RDF_{10}	$10\ k\Omega < R_{op} < 20\ k\Omega$
OB6	RDF_{01}	RDF_{10}	$25\ k\Omega < R_{op} < 75\ k\Omega$
	$RDF_{01} \cup RDF_{11}$	$RDF_{10} \cup RDF_{00}$	$75\ k\Omega < R_{op} < 125\ k\Omega$

TABLE 9
Summary of the Observed FFMs for Each Short

Short	Simulated	Complementary
SC1	SF_0, TF↓, WDF_0, RDF_0, RDF_{00}, RDF_{10}	SF_1, TF↑, WDF_1, RDF_1, RDF_{11}, RDF_{01}
SC2	SF_1, TF↑, WDF_1, RDF_1, RDF_{01}, RDF_{11}	SF_0, TF↓, WDF_0, RDF_0, RDF_{10}, RDF_{11}
SB1	TF↓, WDF_0, RDF_0, RDF_{00}, RDF_{10}	TF↑, WDF_1, RDF_1, RDF_{11}, RDF_{01}
SB2	TF↑, WDF_1, RDF_0, RDF_1, RDF_{00}, RDF_{01}, RDF_{10}, RDF_{11}	TF↓, WDF_0, RDF_1, RDF_0, RDF_{11}, RDF_{10}, RDF_{01}, RDF_{00}

TABLE 10
Summary of the Observed Single-Cell FFMs for Each One of the Analyzed Bridge Defects

Bridge	Simulated	Complementary
BWC1–2	SF_0, TF↓, WDF_0, RDF_0, $DRDF_0$, RDF_{00}, RDF_{10}	SF_1, TF↑, WDF_1, RDF_1, $DRDF_1$, RDF_{11}, RDF_{01}
BWC3–4	SF_0, TF↓, WDF_0, RDF_1, IRF_1, $DRDF_0$, IRF_{01}, IRF_{11}, $DRDF_{00}$, $DRDF_{10}$	SF_1, TF↑, WDF_1, RDF_0, IRF_0, $DRDF_1$, IRF_{10}, IRF_{00}, $DRDF_{11}$, $DRDF_{01}$
BWC5	—	—
BBC1–2	SF_0, TF↓, WDF_0, RDF_0, RDF_{00}, RDF_{01}, RDF_{10}	SF_1, TF↑, WDF_1, RDF_1, RDF_{11}, RDF_{10}, RDF_{01}

faulty behavior of this region is $RDF_{01} \cup RDF_{11}$, while the complementary faulty behavior is $RDF_{10} \cup RDF_{00}$. This strictly dynamic region spreads across $300\ k\Omega < R_{sh} < 1.2\ M\Omega$ with a short capacitance of 0 F. Just like the case with opens, the FFMs present in this fault region are the four types of the dynamic read disturb fault (RDF_{xy}).

6.3 Results of Bridges

Table 10 gives a summary of the observed *single-cell* FFMs for all analyzed bridge defects [14]. The first column in the table specifies the names of the bridges (in case more than one defect sensitize the same FFMs, they are listed together), while the second and third columns list the FFMs observed for the simulated and complementary instances of these defects, respectively. Inspecting the table reveals that all single-cell FFMs defined in Section 3 are present.

The table shows that bridges between a node and rtop or between a node and ctop (see Fig. 6) cause the same faulty behavior. The table also shows that defects not only cause static FFMs, but also result in 2-operation dynamic FFMs, which indicates the significance of dynamic fault analysis. Note that BWC5 does not result in any faulty behavior since it connects the two sides of the parasitic resistance R_s within the cell (see Fig. 6), which is supposed to improve the functionality of the cell. It is important from a testing point of view to state the observed fault regions that only show dynamic faulty behavior since, for these regions, testing for static faulty behavior cannot detect the defect. The only regions with strict dynamic single-cell FFMs belong to BBC1 and BBC2 [12], where, for approximately $R_{br} > 400\ k\Omega$ and $C_{br} > 3C_s$, only RDF_{01} and RDF_{10} can be detected (see Fig. 9).

Table 11 lists the observed *two-cell* FFMs as a result of simulated bridges [14]. The first column in the table lists the

TABLE 11
Summary of the Observed Two-Cell FFMs for Each One of the Analyzed Bridge Defects

Bridge	Simulated	Complementary	Aggressor-sided complementary
BBC1–2	$CFds_{0r0;0}$, $CFds_{1r1;0}$, $CFds_{0w0;0}$, $CFds_{1w1;0}$, $CFds_{0w1;0}$, $CFds_{1w0;0}$, $CFds_{0r0;1}$, $CFds_{1r1;1}$, $CFds_{0w0;1}$, $CFds_{1w1;1}$, $CFds_{0w1;1}$, $CFds_{1w0;1}$	$CFds_{1r1;1}$, $CFds_{0r0;1}$, $CFds_{1w1;1}$, $CFds_{0w0;1}$, $CFds_{1w0;1}$, $CFds_{0w1;1}$, $CFds_{1r1;0}$, $CFds_{0r0;0}$, $CFds_{1w1;0}$, $CFds_{0w0;0}$, $CFds_{1w0;0}$, $CFds_{0w1;0}$	$CFds_{1r1;0}$, $CFds_{0r0;0}$, $CFds_{1w1;0}$, $CFds_{0w0;0}$, $CFds_{1w0;0}$, $CFds_{0w1;0}$, $CFds_{1r1;1}$, $CFds_{0r0;1}$, $CFds_{1w1;1}$, $CFds_{0w0;1}$, $CFds_{1w0;1}$, $CFds_{0w1;1}$
BBC3–6	$CFst_{0;1}$, $CFrd_{0;1}$, $CFds_{0r0;1}$, $CFds_{1w1;0}$, $CFds_{0w1;0}$, $CFds_{1w0;1}$	$CFst_{1;0}$, $CFrd_{1;0}$, $CFds_{1r1;0}$, $CFds_{0w0;1}$, $CFds_{1w0;1}$, $CFds_{0w1;0}$	$CFst_{1;1}$, $CFrd_{1;1}$, $CFds_{1r1;1}$, $CFds_{0w0;0}$, $CFds_{1w0;0}$, $CFds_{0w1;1}$
BBC7–8	—	—	—

$$\{\Updownarrow(w0); \Uparrow(r0, w1, w0, w1, r1); \Uparrow(r1, w0, w1, w0, r0); \Downarrow(r0, w1, w0, w1, r1); \Downarrow(r1, w0, w1, w0, r0); \Downarrow(r0)\}$$

(a)

$$\{\Updownarrow(w0); \Uparrow(r0, w1, w0, w1, r1, \mathbf{r1}); \Uparrow(r1, w0, w1, w0, r0, \mathbf{r0}); \Downarrow(r0, w1, w0, w1, \mathbf{w1}, \mathbf{r1}, r1); \Downarrow(r1, w0, w1, w0, \mathbf{w0}, \mathbf{r0}, r0); \Downarrow(r0)\}$$

(b)

Fig. 11. (a) Conventional March LA test. (b) March LAd designed to detect the observed 2-operation dynamic FFMs.

defect name, the second lists the simulated faulty behavior, the third lists the complementary behavior, while the fourth lists the aggressor-sided complementary faulty behavior. Note that the table does not include any of the "exchanged defect" classes shown in Table 6 because FFMs caused by an exchanged defect are the same as the FFMs caused by a simulated defect with the only difference that the aggressors and victims are exchanged. Also note that BBC7 and BBC8 do not result in any faulty behavior as they connect the word line of the victim to the cell capacitor of the aggressor. Since the word lines are driven by the strong voltage drivers of the address decoder, they are not much affected by the small charge stored in the cell capacitor of the aggressor. The case when the word line of the aggressor to the cell capacitor of the victim is considered by BBC1 and BBC2, which do result in faults (see Table 11).

The table shows that the following static two-cell FFMs have not been observed: $CFst_{0;0}$, $CFrd_{0;0}$, all CFtrs, all CFwds, all CFirs, and all CFdrs. Moreover, none of the targeted dynamic two-cell FFMs have been observed. This can be explained by the fact that not all possible two-cell dynamic SOSs have been used, but only those that begin with a write operation followed by a read operation on the victim (see Section 3.5). Since no write operation on the victim results in a coupling fault, it is not expected that a subsequent read operation would cause a coupling fault either.

7 Test Implications

The fault analysis performed on the cell array column of the *e*DRAM shows that all defined static and targeted dynamic FPs do occur. Moreover, some defects result in a faulty behavior with only dynamic fault models by performing certain SOSs on a memory cell. In order to ensure that a particular memory cell array is not faulty, tests should be developed to sensitize and detect all static and dynamic FFMs resulting from the sensitized FPs. Many tests have been proposed to detect static FFMs, such as MATS+, March C– [2], and March LA [9].

In order to construct a test that uncovers dynamic FFMs, it is important first to derive detection conditions for these FFMs. As discussed in Section 6, the following targeted dynamic FFMs have been observed in our study: RDF_{xy}, IRF_{xy}, and $DRDF_{xy}$, where x and $y \in \{0, 1\}$. These dynamic FFMs can be detected by a given march test if it contains a march element with the following operation sequences:

1. $\Updownarrow(\ldots 0, w0, r0, r0, \ldots)$ for RDF_{00}, IRF_{00}, and $DRDF_{00}$,
2. $\Updownarrow(\ldots 0, w1, r1, r1, \ldots)$ for RDF_{01}, IRF_{01}, and $DRDF_{01}$,
3. $\Updownarrow(\ldots 1, w0, r0, r0, \ldots)$ for RDF_{10}, IRF_{10}, and $DRDF_{10}$,
4. $\Updownarrow(\ldots 1, w1, r1, r1, \ldots)$ for RDF_{11}, IRF_{11}, and $DRDF_{11}$,

where $(\ldots 0)$ and $(\ldots 1)$ specify the state of the cell before performing the first write operations. The first read operation in the conditions above sensitizes and detects dynamic RDF_{xy} and IRF_{xy}, while $DRDF_{xy}$ is sensitized by the first read operation and detected by the second.

Note that, in order to detect $DRDF_x$ or $DRDF_{yx}$, it should be enough to perform either $\Updownarrow(\ldots, rx, rx, \ldots)$ or $\Updownarrow(\ldots, rx)\Updownarrow(rx, \ldots)$. In other words, performing two consecutive read operations, either within one march element or in two consecutive march elements, should be enough to detect any deceptive read disturb fault. In Conditions 1 to 4, however, the two read operations must be within one march element. This is due to the fact that, in a defective DRAM, operations only partially charge or discharge the memory cell. Since a partially charged cell leaks away its voltage in a shorter period of time than a fully charged cell, the refresh mechanism does not

guarantee restoring the faulty state into the cell anymore. Therefore, the detection condition $\Updownarrow(\ldots, rx)\Updownarrow(rx, \ldots)$ cannot be used since a long period of time passes between the first and the second read operation (usually longer than the refresh time).

The above detection conditions can be used to extend existing tests designed for static FFMs to detect 2-operation dynamic FFMs. As an example, March LA is one of the more complex theoretically derived march tests, designed to detect all then known FFMs [9]. A number of operations can be added to the march elements of March LA, based on the detection conditions, to make it capable of detecting the observed dynamic FFMs. Fig. 11a shows the conventional March LA test, while Fig. 11b shows the extended version called March LAd ("d" for dynamic), which is designed to detect the 2-operation dynamic FFMs [12]. The operations added to March LA are shown in the figure in bold face. Note that march sequences $w1, w1$ in march element 3 and $w0, w0$ in march element 4 are required because of the SOS xwx; the first wx initializes the memory cell.

8 Conclusions

In this paper, the faulty behavior of an *e*DRAM has been analyzed using defect injection and circuit simulation. The fault analysis has not been restricted to the static memory behavior, but the 2-operation dynamic behavior has also been included. Known static FFMs have been observed and related to given defects in the memory. New static FFMs (SF_x and WDF_x) and a number of new dynamic FFMs (RDF_{xy}, IRF_{xy}, and DRDF_{xy}) have been reintroduced and established for injected defects. The analysis showed that dynamic faulty behavior can take place in the absence of static faulty behavior which indicates the importance of dynamic fault analysis. Finally, the results of the analysis have been used to derive detection conditions, together with a test, for the observed dynamic FFMs. The new March LAd test is based on the conventional March LA test and adds a number of write and read operations within existing march elements to be able to detect the dynamic faulty behavior.

References

[1] S.S. Iyer and H.L. Kalter, "Embedded DRAM Technology: Opportunities and Challenges," *IEEE Spectrum*, vol. 36, no. 4, pp. 56-64, 1999.

[2] A.J. van de Goor, *Testing Semiconductor Memories, Theory and Practice.* Gouda, The Netherlands: ComTex Publishing, 1998, http://ce.et.tudelft.nl/~vdgoor/.

[3] R.D. Adams and E.S. Cooley, "Analysis of a Deceptive Destructive Read Memory Fault Model and Recommended Testing," *Proc. IEEE North Atlantic Test Workshop*, 1996.

[4] A.J. van de Goor and J. de Neef, "Industrial Evaluation of DRAM Tests," *Proc. Design, Automation, and Test in Europe*, pp. 623-630, 1999.

[5] I. Schanstra and A.J. van de Goor, "Industrial Evaluation of Stress Combinations for March Tests Applied to SRAMs," *Proc. IEEE Int'l Test Conf.*, pp. 983-992, 1999.

[6] S. Naik, F. Agricola, and W. Maly, "Failure Analysis of High Density CMOS SRAMs," *IEEE Design and Test of Computers*, vol. 10, no. 2, pp. 13-23, 1993.

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

[8] R. Dekker et al., "A Realistic Fault Model and Test Algorithms for Static Random Access Memories," *IEEE Trans. Computers*, vol. 39, no. 6, pp. 567-572, June 1990.

[9] A.J. van de Goor et al., "March LA: A Test for Linked Memory Faults," *Proc. European Design and Test Conf.*, p. 627, 1999.

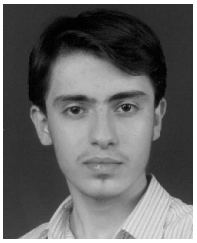
[10] A. Jee and F.J. Ferguson, "Carafe: An Inductive Fault Analysis Tool for CMOS VLSI Circuits," *Proc. IEEE VLSI Test Symp.*, pp. 92-98, 1993.

[11] C.L. Henderson, J.M. Soden, and C.F. Hawkins, "The Behavior and Testing Implications of CMOS IC Logic Gate Open Circuits," *Proc. IEEE Int'l Test Conf.*, pp. 302-310, 1991.

[12] Z. Al-Ars, "Analysis of the Space of Functional Fault Models and Its Application to Embedded DRAMs," Technical Report no. 1-68340-28(1999)-07, CARDIT, Delft Univ. of Technology, Delft, The Netherlands, 1999.

[13] 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*, pp. 496-503, 2001.

[14] Z. Al-Ars and A.J. van de Goor, "Impact of Memory Cell Array Bridges on the Faulty Behavior in Embedded DRAMs," *Proc. Asian Test Symp.*, pp. 282-289, 2000.



Zaid Al-Ars received the MS degree in electrical engineering with honors from the Delft University of Technology, The Netherlands, in 2000. He is working toward the PhD degree in electrical engineering at the same university in cooperation with Infineon Technologies, Munich, Germany, where he is currently based. His research project involves systematic fault analysis and test generation and optimization for commodity as well as embedded DRAM products. He has published more than 10 papers in the field of electrical defect simulation, fault modeling, and test generation in memory devices.

Ad J. van de Goor obtained the MSEE degree from the Delft University of Technology, Delft, The Netherlands, in 1965. He additionally obtained another MSEE degree and the PhD degree from Carnegie-Mellon University, Pittsburgh, Pennsylvania. He worked with Digital Equipment Corporation, in Maynard, Massachusetts, as the chief architect of the PDP-11/45 computer. He also worked for IBM in the Netherlands and in the USA, being responsible for the architecture of embedded systems. Currently, he is a professor of computer engineering at the Delft University of Technology. His main research interests are in testing memories and logic. He has written two books and more than 150 papers in the areas of computer architecture and testing. Dr. van de Goor is a fellow of the IEEE and he is on the editorial board of the *Journal of Electronic Testing: Theory and Applications*.