Delft University of Technology

Data Background-Based Test Development for All Interconnect and Contact Defects in RRAMs

Xun, Hanzhi; Fieback, Moritz; Yuan, Sicong; Zhang, Ziwei ; Taouil, Mottaqiallah; Hamdioui, Said

DOI
10.1109/ETS56758.2023.10174106

Publication date
2023

Document Version
Final published version

Published in
Proceedings of the 2023 IEEE European Test Symposium (ETS)

Citation (APA)
Xun, H., Fieback, M., Yuan, S., Zhang, Z., Taouil, M., & Hamdioui, S. (2023). Data Background-Based Test Development for All Interconnect and Contact Defects in RRAMs. In *Proceedings of the 2023 IEEE European Test Symposium (ETS)* (Proceedings of the European Test Workshop; Vol. 2023-May). IEEE. https://doi.org/10.1109/ETS56758.2023.10174106

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

https://www.openaccess.nl/en/you-share-we-take-care

Otherwise as indicated in the copyright section: the publisher
is the copyright holder of this work and the author uses the
Dutch legislation to make this work public.

Data Background-Based Test Development for All Interconnect and Contact Defects in RRAMs

Hanzhi Xun[1], Moritz Fieback[1], Sicong Yuan[1], Ziwei Zhang[1], Mottaqiallah Taouil[1,2], Said Hamdioui[1,2]
[1]Computer Engineering Laboratory, Delft University of Technology, Mekelweg 4, 2628CD, Delft, The Netherlands
[2]CognitiveIC, Van der Burghweg 1, 2628CS, Delft, The Netherlands
Email: {h.xun, m.c.r.fieback, s.yuan-4, m.taouil, s.hamdioui}@tudelft.nl ziweizhang0808@gmail.com

2023 IEEE European Test Symposium (ETS) | 979-8-3503-3634-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ETS56758.2023.10174106

***Abstract*—Resistive Random Access Memory (RRAM) is a potential technology to replace conventional memories by providing low power consumption and high-density storage. As various manufacturing vendors make significant efforts to push it to high-volume production and commercialization, high-quality and efficient test solutions are of great importance. This paper analyzes interconnect and contact defects in RRAMs, while considering the impact of the memory Data Background (DB), and proposes test solutions. The complete interconnect and contact defect space in a layout-independent RRAM design is defined. Exhaustive defect injection and circuit simulation are performed in a systematic manner to derive appropriate fault models, not only for single-cell and two-cell coupling faults, but also for multi-cell coupling faults where the DBs are important. The results show the existence of unique 3-cell and 4-cell coupling faults due to e.g., the sneak path in the array induced by defects. These unique faults cannot be detected with traditional RRAM test solutions. Therefore, the paper introduces a test generation method that takes into account the DB, which is able to efficiently detect all these faults; hence, further improving the fault/defect coverage in RRAMs.**

***Index Terms*—RRAM, interconnect and contact defects, data background, fault models, test development.**

I. INTRODUCTION

Resistive Random Access Memory (RRAM), as the next-generation memory, is promising to replace conventional memories including Dynamic RAM (DRAM) and Flash [1]. The technology has a variety of benefits such as high integration density, 3D stack-ability, Complementary Metal Oxide Semiconductor (CMOS) compatibility, high cycle endurance, and fast access time [1, 2]. However, both the Front-End-Of-Line (FEOL) and Back-End-Of-Line (BEOL) processes in RRAM manufacturing could introduce production defects [3–5]; these defects can be unique to RRAMs such as over forming [6] or traditional such as interconnect and contact defects [7]. Such defects can cause different kinds of faults leading to the wrong functionality of the memory chips [5, 8]. Hence, understanding the behavior of the RRAM in the presence of such defects is extremely important to develop appropriate test solutions ensuring high outgoing product quality.

Many researchers have addressed the issue of fault modeling and test development for RRAMs. In 2009, Ginez *et al.* modeled bridge defects as linear resistors to study coupling faults in RRAM array [7]. In 2013, sneak-path testing for RRAMs was presented to reduce the test time [9]. In 2015, a dynamic write disturbance fault was identified for the first time by simulating resistive defects in the netlist, and a March test was proposed to cover this fault [10]. In the following year, Lin *et al.* offered a test method for finding the boundary currents of RRAMs in the production test phase [11]. In 2019, Fieback *et al.* put forward the method of 'device-aware test' to model and test unique defects inside the RRAM, but only single-cell faults are considered in this work [6]. In 2021, Liu *et al.* developed a Design-for-Testability (DfT) scheme for 3D hybrid RRAM array [8]. Although all of these works contributed to a better understanding of the memory faulty behavior in the presence of the defects, they restricted the analysis to only faults involving one cell (i.e., victim-cell) or at most two cells (two-cell coupling faults). The potential impact of neighboring cells on the victim cell (i.e., Data Background (DB)) was ignored. This is a worthy aspect to investigate given the fact that extra paths can take place in such memories during read operations; this current is strongly DB dependent, and if high enough may lead to incorrect operations.

This paper advances the state-of-the-art by providing a systematic defect analysis and fault modeling for all possible interconnect and contact defects in RRAMs, while incorporating the impact of the DBs. It demonstrates that such DBs have an impact and could cause unique 3-cell and 4-cell faults; these have to be taken into consideration when developing test solutions, otherwise they lead to tests with low coverage resulting in escapes. The main contributions of the paper are:

- Define and analyze the complete space of interconnect and contact defects in layout-independent RRAM design, and derive all sets of fault models in a systematic manner.
- Demonstrate the existence of unique 3-cell and 4-cell faults using a DB-integrated fault analysis methodology in the presence of the defined defects.
- Use a systematic approach to develop an optimal test algorithm, which detects all sensitized faults by interconnect/contact defects, including the 3-cell and 4-cell faults.
- Validate state-of-the-art RRAM tests in simulation and demonstrate the superiority of our solution.

The rest of this paper is structured as follows. Section II establishes the background on RRAMs and tests. Section III defines the complete interconnect and contact defect space. Section IV presents the simulation methodology. Section V analyzes the resulting faults. Section VI proposes a test solution. Finally, Section VII discusses and concludes the paper.

979-8-3503-3634-4/23/$31.00 ©2023 IEEE

Authorized licensed use limited to: TU Delft Library. Downloaded on July 20,2023 at 12:26:44 UTC from IEEE Xplore. Restrictions apply.

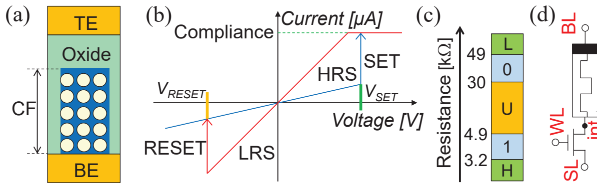


Fig. 1. RRAM device technology. (a) RRAM structural, (b) Simplified switching I-V curve, (c) RRAM resistance range, (d) 1T-1R cell.

II. Background

A. RRAM principles and cell designs

The RRAM device is a Metal-Insulator-Metal (MIM) construction, as shown in Fig. 1a [1, 3]; a middle metal oxide is sandwiched between two metal electrodes: the Top Electrode (TE), and the Bottom Electrode (BE) [1, 2]. By applying a high positive voltage (i.e., forming voltage) between TE and BE, localized deficiency leads to the formation of Conducting Filament (CF) [1, 2]. Fig. 1b describes the typical current-voltage (I-V) curve during the switching process [1]. By applying specific programming voltages to an RRAM device, its resistance can be switched between different states [2]. The CF length will increase when applying a positive voltage larger than the specified threshold $V_{TE} \geq V_{SET}$ (as this generates more oxygen vacancies [3]), and will be shortened when applying a negative voltage lower than the reset threshold $V_{TE} \leq V_{RESET}$ (as oxygen ions migrate back from the electrode and fill the vacancies) As shown in Fig. 1c, (binary) RRAMs can have up to five states [4, 12, 13]: 1) the extremely high conductance faulty state 'H', 2) the low resistive correct state '1', 3) the undefined faulty state 'U', 4) the high resistive correct state '0', and 5) the extremely low conductance faulty state 'L'.

The most popular RRAM array design is based on One-Transistor-One-Resistor (1T-1R) cell structure, shown in Fig. 1d [1]. The cell has a Word Line (WL) to control the turn-on of the transistor to make the data stored in desired cells accessible. In addition, a Bit Line (BL) and Select Line (SL) are set to appropriate voltages for performing write/read operations. Fig. 2 shows the 2×2 1T-1R circuit architecture used in this paper. It consists of the core memory cell array and peripheral circuits; cells in the same row share the same WL and SL, whereas those in the same column share the same BL. The peripheral circuits include BL and SL drivers, a WL decoder, a column address decoder, and Sense Amplifiers (SAs) to read out the cells. During a read operation, the SA senses the current through the RRAM cell and compares it to a reference [14].

B. RRAM test development approach

Fig. 3 shows the RRAM test development approach applied in this work [6]; it is called Device-Aware-Test and consists of three steps: 1) *defect modeling* where defects are modeled in an appropriate way, 2) *fault modeling* where the defective device is replaced with a representative compact model during circuit simulation to analyze the memory behavior, and 3) *test development* targeting validated faults in step 2.

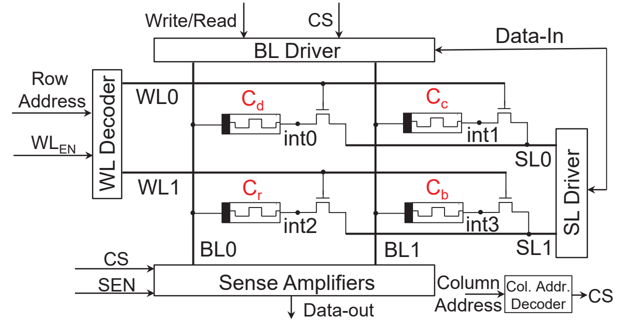


Fig. 2. A 2×2 RRAM circuit architecture.

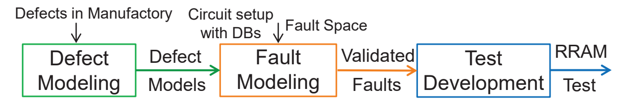


Fig. 3. RRAM test development approach [6, 15].

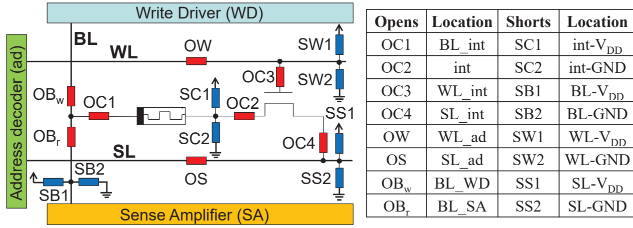
III. Defect modeling

As the targeted defects in this work consist of interconnect and contact defects, it is appropriate to use *linear resistors* to model them [7, 16]; note that this is not the case for other unique defects in RRAMs; e.g., over-forming [6, 17]. Interconnect and contact defects can cause opens, shorts, and bridges [7, 16]. An *open* is defined as increased resistance in an existing connection, a *short* as an undesired resistive path between a node and a power node (V_{DD} or GND), and a *bridge* as a resistor between a pair of nodes different from the power nodes.

Before defining the total number of defects to be simulated, we need to define the simulation platform. To reduce the simulation time, we use the symmetrical nature of the memory array to derive a representative simulation platform while reducing the number of defects to be simulated. Each cell in the memory array (say base cell C_b) has at most 4 adjacent diagonal cells (C_d), two adjacent cells in the same column (C_d), and two adjacent cells in the same row (C_r). Given the symmetry, the simulation platform can be reduced to a 2×2 cell array as shown in Fig. 2; it presents a base cell C_b with a representative of each neighbor.

The symmetry can further help in reducing the number of defects to simulate within 2×2 arrays. For example, in Fig. 2, a bridge between int0 of C_d and int3 of C_b exhibits symmetry to a bridge between int1 of C_c and int2 of C_r; hence, only one of these needs to be simulated. Applying the symmetry to the simulation platform of Fig. 2 results in 8 opens, 8 shorts, and 23 bridges; these give the complete defect space that needs to be simulated in order to fully analyze all possible interconnect and contact defects within an RRAM array. Fig. 4 and Table I give the complete list of opens, shorts, and bridges to be simulated in this work. OX is used to denote the opens $(\mathrm{OX}, \mathrm{X} \in \{\mathrm{C}\,(\text{inside the cell}),\ \mathrm{W}\,(\text{in the WL}),$ $\mathrm{S}\,(\text{in the SL}),\ \mathrm{B_{w/r}}\,(\text{on the write/read side of BL})\})$, SX is used to denote the shorts $(\mathrm{SX}, \mathrm{X} \in \{\mathrm{C}, \mathrm{W}, \mathrm{S}, \mathrm{B}\})$, and xB is used to denote the bridges $(\mathrm{xB}, \mathrm{x} \in \{\mathrm{i}, \mathrm{c}, \mathrm{r}, \mathrm{d}\})$.

Authorized licensed use limited to: TU Delft Library. Downloaded on July 20,2023 at 12:26:44 UTC from IEEE Xplore. Restrictions apply.



Opens	Location	Shorts	Location
OC1	BL_int	SC1	int-V_{DD}
OC2	int	SC2	int-GND
OC3	WL_int	SB1	BL-V_{DD}
OC4	SL_int	SB2	BL-GND
OW	WL_ad	SW1	WL-V_{DD}
OS	SL_ad	SW2	WL-GND
OB_w	BL_WD	SS1	SL-V_{DD}
OB_r	BL_SA	SS2	SL-GND

Fig. 4. Open and short defect locations.

TABLE I
BRIDGE DEFECT LOCATIONS.

Bridges	Location	Bridges	Location	Bridges	Location
iB1	BL1-int3	cB3	int1-SL1	rB2	BL0-int3
iB2	BL1-WL1	cB4	WL0-int3	rB3	int3-BL1
iB3	BL1-SL1	cB5	WL0-WL1	rB4	int2-int3
iB4	int3-WL1	cB6	WL0-SL1	dB1	int0-BL1
iB5	int3-SL1	cB7	SL0-int3	dB2	int0-int3
iB6	WL1-SL1	cB8	SL0-WL1	dB3	int0-WL1
cB1	int1-int3	cB9	SL0-SL1	dB4	int0-SL1
cB2	int1-WL1	rB1	BL0-BL1		

TABLE II
FAULT PRIMITIVE NOTATION [18].

	Explanation	Values
S	Sensitizing sequence	$x_0 O_1 x_1 \ldots O_q x_q$, $j \in \{0, 1, ..., q\}$ $x_j \in \{0, 1\}$, $O_j \in \{\mathrm{w}, \mathrm{r}\}$
F	Faulty behavior	L, 0, U, 1, H
R	Readout value	0, 1, ?, -

IV. FAULT MODELING METHODOLOGY

In this section, we define the fault space and propose the simulation setup methodology.

A. Fault space and classification

A fault primitive (FP), denoted by $\langle \mathrm{S/F/R} \rangle$, is a systematic method to describe all faults that lead to incorrect logical behavior, as illustrated in Table II [18]. By using the FP notation, the fault space for single-cell faults can be defined and described [18]. Such notation can be extended to describe multi-cell faults involving p cells (p $\geq$ 2) to: $\langle \mathrm{S_{a_1}; \ldots; S_{a_{p-1}}; S_v/F/R} \rangle$, where $\mathrm{S_{ai}}$ ($\mathrm{i} \in [1, \mathrm{p}-1]$) represents the *sensitizing sequence* of the aggressor cell and $\mathrm{S_v}$ indicates the state/sequence performed on the victim cell [18].

Depending on the number of operations involved in sensitizing operation(s), faults can be classified into *static* and *dynamic* faults. Static faults are sensitized by applying up to one operation, while dynamic faults are sensitized by applying multiple consecutive operations. Moreover, each fault can be either *strong* or *weak* [15]; a strong fault is always sensitized by a certain sequence of operations and it can be described by an FP, while a weak fault does not cause any functional errors but parametric deviations instead and cannot be described by an FP (e.g., a voltage drop in the BL during a writing operation). Strong faults that are *guaranteed* to be sensitized and detected by regular memory operations are called *Easy-to-Detect (EtD)* faults. Strong faults which have no deterministic behavior (e.g., random read) and weak faults are called *strong/weak Hard-to-Detect (sHtD/wHtD) faults*.

B. Simulation setup and methodology

The circuit in Fig. 2 is implemented in Cadence's Spectre simulator by using the Predictive Technology Model (PTM) 130-nm transistor library [19] and the RRAM compact model from [20]. The nominal supply voltage for the memory is 3 V. In order to accurately evaluate the circuit, capacitive loads are applied to BLs, SLs, and WLs in the simulation. The defect-free circuit is verified for correct operations [15].

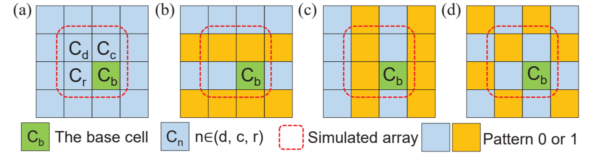


Fig. 5. RRAM simulation setup with common DBs. (a) Solid, (b) Row stripe, (c) Column stripe, (d) Checkerboard.

For defect injection and circuit simulation of a 2 × 2 array, we consider three parameters: 1) defect strengths, 2) sensitizing sequences (S), and 3) Data-backgrounds (DBs). To perform simulation for different strengths of the defect, each resistive defect is swept from 1 Ω to 100 MΩ with 81 different defect strengths, distributed on a logarithmic scale. The applied sensitizing sequences consist of *up to three* consecutive read/write operations (in total 80). Finally, a DB is established and defined as the pattern of ones and zeros as seen in an array of memory cells. Here, we use the most commonly known 4 DBs [18]: solid (all 0s and all 1s), row stripe (0000.../1111.../0000.../1111...), column stripe (0101.../0101.../0101.../0101...) and checkerboard (0101.../1010.../0101.../1010...); these are illustrated in colors in Fig. 5. We also represent the DB as states of '$\mathrm{C_d, C_c, C_r}$'. For example, '1, 1, 0' refers to the row stripe.

For each defect with a set strength injected in the 2 × 2 array, the sensitizing operations are applied, then the DB and its complement are established. After each DB and its complement, the state of each of the 2 × 2 cells is extracted before establishing the next DB. Once all four DBs are simulated, the Ss are changed and the process is repeated. Once all Ss are performed, the strength of the defect is changed and the process of applying Ss is repeated. Once all strengths are simulated, the next defect will be simulated using a similar process. This approach enables investigating the sensitization of single-cell faults and multi-cell faults at the same time. For example, if a fault in a cell occurs regardless of the states of other cells, then it is a single-cell fault; otherwise it is a multi-cell coupling fault. The number of involved cells determines whether the fault is a 2-cell, 3-cell, or 4-cell coupling fault.

V. FAULT MODELING RESULTS

We present the detailed validated fault results for a single defect, followed by the combined results for all defects.

A. Results for a single defect (dB1)

Table III shows faults that are sensitized with sensitizing sequences, varying DBs, and defect strengths for the bridge defect dB1 (see Table I). In the table, we selectively list sequences due to limited space: 1) static sequences that sensitize faults, 2) dynamic sequences that sensitize faults in additional defect ranges. The DB consists of two parts: 1) the state of the $\mathrm{C_d}$, 2) states of the two other neighboring cells $\mathrm{C_c}$, $\mathrm{C_r}$,

Authorized licensed use limited to: TU Delft Library. Downloaded on July 20,2023 at 12:26:44 UTC from IEEE Xplore. Restrictions apply.

TABLE III
FAULT MAP FOR dB1 DEFECT. ETD SHTD FAULT FREE

Defect: dB1													
Ss	DBs		Defect strength region										
	Cd	Cc Cr	1	20k	25k	40k	50k	63k	79k	159k	316k	398k	501k-100M
0w0	0	x x	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	sHtD	sHtD	Fault free
	1	x 0	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free
		x 1	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free
0w1	0	x x	sHtD	sHtD	sHtD	EtD	EtD	EtD	EtD	EtD	sHtD	sHtD	Fault free
	1	x 0	EtD	EtD	EtD	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free
		x 1	EtD	EtD	EtD	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free
1w0	0	x x	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	sHtD	sHtD	Fault free
	1	x 0	sHtD	sHtD	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free
		x 1	sHtD	sHtD	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free
1w1	0	x x	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	sHtD	sHtD	Fault free
	1	x 0	EtD	EtD	EtD	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free
		x 1	EtD	EtD	EtD	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free
0r0	0	x x	EtD	EtD	EtD	EtD	EtD	Fault free	Fault free	Fault free	Fault free	Fault free	Fault free
	1	x 0	EtD	EtD	EtD	EtD	EtD	EtD	EtD	Fault free	Fault free	Fault free	Fault free
		x 1	EtD	EtD	EtD	EtD	EtD	EtD	Fault free	Fault free	Fault free	Fault free	Fault free
0w0r0	0	x x	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	sHtD	Fault free
	1	x 0	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	Fault free	Fault free
		x 1	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	Fault free	Fault free
1w0r0	0	x x	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	sHtD	Fault free
	1	x 0	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	Fault free
		x 1	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	sHtD	Fault free
0r0r0	0	x x	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	Fault free	Fault free
	1	x 0	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	sHtD	Fault free
		x 1	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	EtD	Fault free	Fault free

where $x \in \{0, 1\}$. The grey shape indicates fault-free behavior, the green presents sensitized EtD faults and the orange sHtD faults. Note that several types of faults can be sensitized by even one defect strength and sequence. For example, both $\langle 1w0; 0/1/- \rangle$ and $\langle 1w0/U/- \rangle$ are sensitized by 1w0 under the DB of 'C_d, C_c, C_r'='0, x, x' for the defect strength up to $159\,k\Omega$; these details are not included in the table. If one of these is EtD, the box is green. The corresponding defect can be detected by at least sensitizing one EtD fault.

The table provides two insights for test development. First, the longest EtD range needs to be chosen for high test coverage; this is the case for Ss with a higher number of operations. E.g., a sequence 1w0r0 sensitizes more EtD faults and covers a wider defect range than 1w0 only, irrespective of the DBs. Hence, when designing a test, 1w0r0 should be selected over 1w0. Secondly, the DB does have an effect on the faulty behavior of RRAMs. For instance, the sequence 1w0r0 uniquely sensitizes EtD faults from $316\,k\Omega$ up to $398\,k\Omega$ under the DB of '1, x, 0'; i.e., the states of two neighboring cells are required. This case implies that we must incorporate the DB into the RRAM test development or risk missing EtD faults.

B. Overall result overview

1) Static faults: Fig. 6 gives the relative number of sensitized static faults for all simulated defects; *the number of faults* for each sequence is defined as *the number of defect strengths that sensitize faults*. For example, the number of (single static) faults sensitized by 1w0 for defect dB1 (see Table III) is 2; note that there are faults, which are DB independent taking place for defect sizes of up to $20\,k\Omega$. The total number of faults (single-cell) sensitized by 1w0 (when considering all defects) counts for 83% as shown in the orange bar of Fig. 6. Note that Fig. 6 is normalized to 1w0 because this sequence is the one sensitizing the maximum number of faults. This is because the reset process in an RRAM device is a negative feedback loop that is susceptible to defect-induced variations [2, 20]. Fig. 6 shows that both single-cell and multi-cell faults are sensitized. The number of single-cell and two-cell faults accounts for the majority (99%) of the total. However, a small

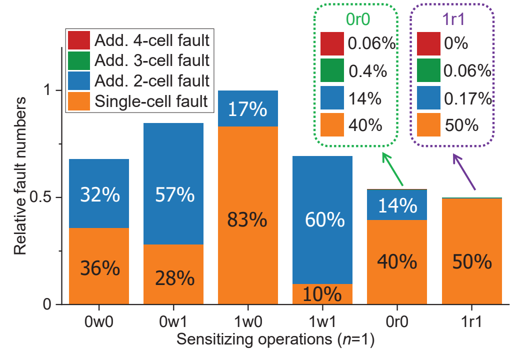


Fig. 6. Relative number of faults sensitized by static sequences.

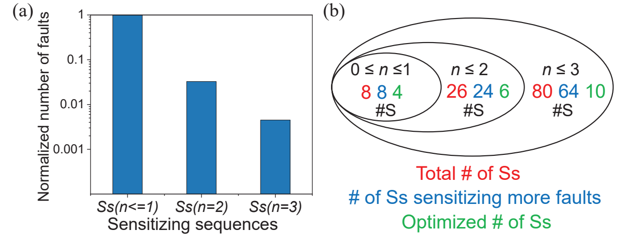


Fig. 7. Dynamic EtD faults sensitized as n_{max} increases. (a) The normalized number of faults, (b) The number of sequences.

number of 3-cell and 4-cell faults are observed when applying read operations, showing the importance of sensitizing multi-cell static faults if high product quality is targeted. The detailed analysis will be provided in Section V-B3.

2) Dynamic faults: Fig. 7a shows the number of 2-operation and 3-operation sensitized dynamic EtD faults as compared to static faults. The total number of Ss is: $\#S = \sum_{n=0}^{3} 2 \times 3^n = 80$, consisting of 8 1-operation Ss, 18 2-operation Ss, and 54 3-operation Ss [15]. The results of Fig. 7a are normalized to the total number of faults sensitized by the static analysis. As already mentioned, the number of faults is defined as the *number of defect strengths* that sensitize faults. The figure shows that although the total number of Ss increases exponentially, the expansion of defect strengths sensitizing new faults slows down. Clearly, increasing the number of operations (n_{max}) per S contributes to the sensitization of more faults/defect strengths.

In Fig. 7b, three ellipses present faults sensitized by sequences with different lengths given by n. In each ellipse, there are three numbers from left to right: 1) the total number of sequences for the corresponding n (red), 2) the number of sequences for this n that can sensitize faults which were not sensitized for the case $\leq n-1$ (blue), and 3) the minimum numbers of sequence that need to be applied in order to cover the maximum defect coverage (green). For example, for $n \leq 2$, there are 26 possible Ss (from which 8 are applied also for $n \leq 1$), only 16 (24-8) Ss can sensitize additional faults which were not observed for $0 \leq n \leq 1$, and only 6 Ss are needed to maximum defect coverage; this optimization will be explained for the test development in Section VI. Fig. 7b indicates that not all dynamic sequences sensitize additional faults than static sequences, hence only a subset is needed to detect all defects. E.g., for $n \leq 3$, only 4 (10-6) additional Ss are needed as compared with $n \leq 2$ to sensitize all validated faults.

Authorized licensed use limited to: TU Delft Library. Downloaded on July 20,2023 at 12:26:44 UTC from IEEE Xplore. Restrictions apply.

3) Root and analysis of 3-cell and 4-cell faults: Table III shows the 3-cell fault takes place in the presence of dB1 for $S=$ 1w0r0 and DB of ‘1, x, 0’. For the defect-free circuit, when performing a 1w0r0 operation, read current flows from BL1, into int3, and from SL1 to ground. For the defective circuit, extra read current also flows from BL1, through the defect, to int0, BL0, int2 (WL1 is active), and SL1 to ground. The current induced by the defect flows through neighboring cells ($\mathrm{C_d}$ and $\mathrm{C_r}$) of accessed cell $\mathrm{C_b}$, which affects the discharge speed of the sensed node, leading to an incorrect read fault. The magnitude of the current depends on the resistance of the defect and the states of C_d and C_r. When the defect resistance is low, a read fault will occur that does not depend on the DB. However, with increasing defect resistance, the resistance of the neighboring cells becomes essential, and thus the DB starts to play a role in fault sensitizing. Hence, the 3-cell fault is sensitized. Furthermore, 4-cell faults can also be sensitized. The gate-drain capacitance of the transistor consumes a slight current. Hence, even though a particular cell (e.g., $\mathrm{C_c}$) is neither in the path induced by defect nor is its WL conducting, the cell state has an impact on the magnitude of the charging current. Besides, the current induced by the defect still exists. In these special cases, the read current flows through all 4 cells and is affected by their states, leading to an incorrect 4-cell read fault.

VI. Test development

This section uses a specific test generation method to develop a test solution for targeted faults in this paper. The test is validated and thereafter compared with prior work.

A. DB-aware test approach

1) Test generation: To detect both single and multi-cell faults, a test solution needs to take the DB into account. There are two requirements: 1) to fully cover defect strengths that sensitize EtD faults, and 2) to minimize the test length. In [15], the test development is formulated as an Integer Linear Programming (ILP) problem that minimizes the number of applied sequences while maximizing the defect coverage. We apply a similar method. In addition, we aim at minimizing the number of the DBs during the test envelopment as this makes the test implementation easier and even results in shorter test length. Hence, a DB-ILP method is proposed.

We illustrate the use of DB-ILP approach in Table IV. Assume a four-dimensional binary matrix $a_{d,r,z,b} \in \mathbf{A}$, where ‘$d$’ denotes the defects ($\mathrm{D}_d$) from 1 to D, ‘r’ denotes the defect strengths (DS_r) for each defect from 1 to R, ‘z’ denotes the sequence ($\mathrm{S}_{z,b}$) from 1 to Z, and ‘b’ denotes DBs (DB_b) from 1 to B. The total number of elements in $\mathbf{A}$ is $D \times R \times Z \times B$. Then, if at least one EtD fault is sensitized for defect strength r of the defect d with the corresponding sequence $\mathrm{S}_{z,b}$ of the DB ‘b’, this element ($a_{d,r,z,b}$) in the matrix is set as ‘1’, otherwise it is set to ‘0’. For example, for ‘green’ entities of Table III (for defect dB1) will correspond to 1’s when mapped onto a single row of Table IV, while the remaining entities will be mapped into 0’s on the same row. The last column lists the sum

TABLE IV
EXAMPLE TO SOLVE THE DB-ILP.

			DB_b, for $b: 1 \to B$							$\sum_{b=1}^{B} \sum_{z=1}^{Z} a_{d,r,z,b}$
			DB_1			$\cdots$	DB_B			
			$\mathrm{S}_{z,b}$ for $z: 1 \to Z$			$\cdots$	$\mathrm{S}_{z,b}$ for $z: 1 \to Z$			
			$\mathrm{S}_{1,1}$	$\cdots$	$\mathrm{S}_{Z,1}$	$\cdots$	$\mathrm{S}_{1,B}$	$\cdots$	$\mathrm{S}_{Z,B}$	
D_1	DS_1	1	1	$\cdots$	1	$\cdots$	0	$\cdots$	0	2
	DS_2	10	1	$\cdots$	0	$\cdots$	0	$\cdots$	1	2
	$\cdots$									
	$\cdots$	10M	1	$\cdots$	0	$\cdots$	0	$\cdots$	1	2
	DS_R	100M	0	$\cdots$	0	$\cdots$	0	$\cdots$	0	0
$\cdots$										
D_D	DS_1	1	0	$\cdots$	0	$\cdots$	0	$\cdots$	1	1
	$\cdots$	10	1	$\cdots$	0	$\cdots$	0	$\cdots$	0	1
	$\cdots$									
	$\cdots$	10M	1	$\cdots$	0	$\cdots$	0	$\cdots$	0	1
	DS_R	100M	0	$\cdots$	0	$\cdots$	0	$\cdots$	0	0

of elements in every row, indicating the number of sequences that can sensitize a fault in the presence of a defect ‘d’ with a defect strength ‘r’. This number is always greater than or equal to 1 as at least one $S_{z,b}$ should sensitize an EtD fault in the presence of D_d. For example, the sum of the elements of row D_D-DS_R is ‘1’, hence we have to choose sequence $\mathrm{S}_{Z,B}$ with DB_B for the test development since D_D-DS_R can be only exited by single sensitizing operation. Now, the DB-ILP optimization can be mathematically denoted as:

$$\min \sum\nolimits_{b=1}^{B} \left(\beta \cdot \mathrm{DB(sel)}_b \cdot \sum\nolimits_{z=1}^{Z} \mathrm{S(sel)}_{z,b} \right)$$

$$\text{s.t.} \begin{cases} \text{For } (d: 1\, to\, D, r: 1\, to\, R): \\ \quad \text{if } \sum_{b=1}^{B} \sum_{z=1}^{Z} a_{d,r,z,b} \geq 1: \\ \qquad \sum_{b=1}^{B} \sum_{z=1}^{Z} a_{d,r,z,b} \cdot \mathrm{DB(sel)}_b \cdot \mathrm{S(sel)}_{z,b} \geq 1. \end{cases}$$

Here, $\mathrm{DB(sel)}_b$ and $\mathrm{S(sel)}_{z,b}$ are binary values, indicating whether the bth DB (DB_b) and the zth sensitizing sequence ($\mathrm{S}_{z,b}$) are selected (i.e., ‘1’ is selected), meaning that $\mathrm{S}_{z,b}$ sensitizes an EtD fault. β is parameters used to give higher weight/cost for changing DBs as compared with changing $\mathrm{S}_{z,b}$. We aim at having fewer DBs selected and set β=80, which is the maximum number of sequences for n=3 for a single DB. The minimization statement guarantees that we get the minimal number of DBs and the associated minimum number of $\mathrm{S}_{z,b}$. The constraints ensure that all sensitized defect strengths are covered.

We apply Python3’s PuLP optimization package to solve the above DB-ILP problem for EtD faults [21]. The output provides the required minimum number of DBs and their associated sensitizing sequences $\mathrm{S}_{z,b}$ needed to sensitize all targeted EtD faults. Note that multiple solutions may exist; they all have the same cost (length). In our case, a minimum of three DBs, each with associated sensitizing sequences, are needed; they are:

$$\begin{aligned} &\mathrm{DB1} = 111 \text{ for } 1\mathrm{w}0, 0\mathrm{r}0\mathrm{r}0, 1\mathrm{w}0\mathrm{r}0, 0\mathrm{w}0\mathrm{w}0\mathrm{w}0, 1\mathrm{r}1\mathrm{w}0\mathrm{r}0 \\ &\mathrm{DB2} = 000 \text{ for } 1\mathrm{w}1, 1\mathrm{r}1, 1\mathrm{w}1\mathrm{r}1\mathrm{r}1, 1\mathrm{w}1\mathrm{w}1\mathrm{w}1 \\ &\mathrm{DB3} = 110 \text{ for } 0\mathrm{r}0, 1\mathrm{w}0\mathrm{r}0 \end{aligned}$$

Note that DB1=(111)=(C_d, C_c, C_r) and DB2 represent solid DBs, while DB3 represents the row stripe DB (see Fig. 5). Sequences with the same DB can be further combined to optimize the test cost. An additional read operation should be added after each sensitizing sequence to ensure the

Authorized licensed use limited to: TU Delft Library. Downloaded on July 20,2023 at 12:26:44 UTC from IEEE Xplore. Restrictions apply.

TABLE V
TEST VALIDATION OF EXISTING RRAM TESTS.

Year	Name	Test escapes of EtD Defects	Test time	
			Write	Read
2013	March-MOM [9]	4.600%	5N	4N
2015	March-1T1R [10]	0.366%	(1+2a+2b)N	5N
2015	March C* [13]	0.653%	4N	6N
2016	March C*-1T1R [22]	0.340%	6N	6N
2017	March W-1T1R [23]	0.392%	9N	8N
2022	**Proposed (March-EtD)**	**0%**	**16N**	**10N**

detection of the fault. Furthermore, $1\mathrm{w}1$ and $1\mathrm{w}1\mathrm{r}1\mathrm{r}1$ can be combined into $1\mathrm{w}1\mathrm{r}1\mathrm{r}1$. In this way, the March test algorithm (refereed to as March-EtD) to detect all EtD faults is generated as follows:

$$\begin{aligned}&\{\Updownarrow(\mathrm{w}\mathbf{1});\Updownarrow(\mathrm{r1,w0,r0,r0,r0,w0,w0,w0,r0,w1});\\&\Updownarrow(\mathrm{w}\mathbf{0});\Updownarrow(\mathrm{w1,w1,w1,w1,r1,r1,r1,w0});\\&\Updownarrow(\mathrm{w}\mathbf{B});\Updownarrow\left(\mathrm{w}\overline{\mathrm{B}}_{\mathrm{e}},\mathrm{wB_e},\mathrm{rB_e},\mathrm{rB_e}\right);\\&\Updownarrow(\mathrm{w}\mathbf{B});\Updownarrow\left(\mathrm{w}\overline{\mathrm{B}}_{\mathrm{u}},\mathrm{wB_u},\mathrm{rB_u},\mathrm{rB_u}\right)\}.\end{aligned}$$

The test used march notation [18]; the first march element M_1 uses '$\Updownarrow$' addressing (indicates that addressing direction is irrelevant) to initialize the memory to $\mathrm{DB1}$ = Solid 1. M_2 ensures the application of all sensitizing sequences associated to $\mathrm{DB1}$, and adds a read operation after sequences to guarantee the detection of faults, irrespective of whether they are destructive or deceptive [16]. M_3 and M_4 do the same but then for $\mathrm{DB2}$ = Solid 0. M_5 applies $\mathrm{DB3}$ = B = row stripe to the memory. M_6 applies the associated sensitising sequences to *even* row consisting of 0's (e.g., w$\mathbf{B}_e$); a read operation is also applied after each sequence. M_7 writes the complementary of B to the memory, and M_8 does the same as M_6, but then for *uneven* rows containing 0's (e.g., w$\mathbf{B}_u$). The test length of March-EtD is $16N_w$+$10N_r$, where N_w/N_r indicate the number of writes and reads, respectively.

2) Test validation: The proposed test is applied to the defect-free and all defect-injected circuits to validate the test coverage. We apply the same simulation setup and circuits as in Section IV. The defect detection is defined as at least one incorrect readout of the March test. Our validated result shows that every defect that sensitizes EtD faults can be detected.

B. Comparison with existing tests

Many RRAM test solutions have been provided in literature [9, 10, 13, 22, 23]. We compare these March tests with our test based on the validated test escape rate and test time. We use the same simulation setup as in Section IV. For each defect, the missing numbers of defect strengths are counted and divided by the total number of EtD defect strengths. The comparison result between March tests is summarized in Table V. It can be concluded that all other March tests have test escapes. For example, defect dB1 ranging from $316\,\mathrm{k}\Omega$ to $398\,\mathrm{k}\Omega$ cannot be detected, although EtD faults are sensitized (see Table III). Note that even a small number of test escapes is problematic towards meeting Defective Part Per Billion (DPPB)-level requirements for RRAMs. Besides, these existing test solutions may sensitize unrealistic faults as overkill, and lead to yield loss. The validation of DfT schemes for HtD faults is not included in this paper.

VII. DISCUSSION AND CONCLUSION

The paper has demonstrated the existence of unique 3-cell and 4-cell coupling faults in RRAMs (in addition to traditional single-cell and two-cell coupling faults) in the presence of interconnect and contact defects. These unique faults are DB-dependent and require special attention otherwise they will lead to escapes. It is worth noting the following:

- **Prevention versus defects:** From the validated fault space, we find that inter-cell bridges have serious impacts on faults among all defects. Especially, bridges between C_b and C_d cause unique 3-cell and 4-cell faults. Hence, we recommend preventing bridge defects by optimizing the layout design.
- **DfT designs:** HtD faults are validated in our simulation and still not guaranteed to be detected by March test. The combination of our DB-ILP March tests with DfTs has the potential to satisfy the detection of those faults.

REFERENCES

[1] H.-S. P. Wong *et al.*, "Metal–Oxide RRAM," *Proc. IEEE*, vol. 100, no. 6, 2012.

[2] F. Nardi *et al.*, "Resistive Switching by Voltage-Driven Ion Migration in Bipolar RRAM—Part I: Experimental Study," *IEEE TED*, vol. 59, no. 9, 2012.

[3] A. Grossi *et al.*, "Fundamental Variability Limits of Filament-Based RRAM," in *IEDM*, 2016.

[4] M. Fieback *et al.*, "Testing Resistive Memories: Where Are We and What Is Missing?" In *ITC*, 2018.

[5] E. I. Vatajelu *et al.*, "Challenges and Solutions in Emerging Memory Testing," *IEEE TETC*, vol. 7, no. 3, 2019.

[6] M. Fieback *et al.*, "Device-Aware Test: A New Test Approach Towards DPPB Level," in *ITC*, 2019.

[7] O. Ginez *et al.*, "Design and Test Challenges in Resistive Switching RAM (ReRAM): An Electrical Model for Defect Injections," in *ETS*, 2009.

[8] P. Liu *et al.*, "Defect Analysis and Parallel Testing for 3D Hybrid CMOS-Memristor Memory," *IEEE TETC*, vol. 9, no. 2, 2021.

[9] S. Kannan *et al.*, "Sneak-Path Testing of Crossbar-Based Nonvolatile Random Access Memories," *IEEE TN*, vol. 12, no. 3, 2013.

[10] Y.-X. Chen *et al.*, "Fault Modeling and Testing of 1T1R Memristor Memories," in *VTS*, 2015.

[11] T.-Y. Lin *et al.*, "A Test Method for Finding Boundary Currents of 1T1R Memristor Memories," in *ATS*, 2016.

[12] S. Hamdioui *et al.*, "Testing Open Defects in Memristor-Based Memories," *IEEE TC*, vol. 64, no. 1, 2015.

[13] C. Y. Chen *et al.*, "RRAM Defect Modeling and Failure Analysis Based on March Test and a Novel Squeeze-Search Scheme," *IEEE TC*, vol. 64, no. 1, 2015.

[14] W. Zhao *et al.*, "Synchronous Non-Volatile Logic Gate Design Based on Resistive Switching Memories," *IEEE TCAS-I*, vol. 61, no. 2, 2014.

[15] M. Fieback *et al.*, "Defects, Fault Modeling, and Test Development Framework for RRAMs," *ACM JETC*, vol. 18, no. 3, 2022.

[16] S. Hamdioui *et al.*, "An Experimental Analysis of Spot Defects in SRAMs: Realistic Fault Models and Tests," in *ATS*, 2000.

[17] M. Fieback *et al.*, "Intermittent Undefined State Fault in RRAMs," in *ETS*, 2021.

[18] S. Hamdioui *et al.*, "Memory Fault Modeling Trends: A Case Study," *JETTA*, vol. 20, no. 3, 2004.

[19] ASU, *Predictive Technology Model (PTM)*, 2012.

[20] H. Li *et al.*, "A SPICE Model of Resistive Random Access Memory for Large-Scale Memory Array Simulation," *IEEE EDL*, vol. 35, no. 2, 2014.

[21] S. Mitchell *et al.*, "PuLP: A Linear Programming Toolkit for Python. 2011," *URL https://code. google. com/p/pulp-or*, 2011.

[22] P. Liu *et al.*, "Efficient March Test Algorithm for 1T1R Cross-Bar with Complete Fault Coverage," *EL*, vol. 52, 18 2016.

[23] Y. Luo *et al.*, "A High Fault Coverage March Test for 1T1R Memristor Array," in *EDSSC*, 2017.

Authorized licensed use limited to: TU Delft Library. Downloaded on July 20,2023 at 12:26:44 UTC from IEEE Xplore. Restrictions apply.