Testing Open Defects in Memristor-Based Memories

Said Hamdioui, *Senior Member*, *IEEE*, Mottaqiallah Taouil, *Student Member*, *IEEE*, and Nor Zaidi Haron, *Student Member*, *IEEE*

Abstract—Memristor-based memory technology, also referred to as resistive RAM (RRAM), is one of the emerging memory technologies potentially to replace conventional semiconductor memories such as SRAM, DRAM, and flash. Existing research on such novel circuits focuses mainly on the integration between CMOS and non-CMOS, fabrication techniques, and reliability improvement. However, research on (manufacturing) test for yield and quality improvement is still in its infancy stage. This paper presents fault analysis and modeling for open defects based on electrical simulation, introduces fault models, and proposes test approaches for RRAMs. The fault analysis reveals that unique faults occur in addition to some conventional memory faults, and the detection of such unique faults cannot be guaranteed with just the application of traditional march tests. The paper also presents a new Design-for-Testability (DfT) concept to facilitate the detection of the unique faults. Two DfT schemes are developed by exploiting the access time duration and supply voltage level of the RRAM cells, and their simulation results show that the fault coverage can be increased with minor circuit modification. As the fault behavior may vary due to process variations, the DfT schemes are extended to be programmable to track the changes and further improve the fault/defect coverage.

Index Terms—Memory defects, fault models, defect-oriented testing, design-for-testability (DfT), memristor, RRAMs

1 Introduction

Memristor-*based random access memory*, also referred to as resistive random access memory (RRAM), is one of the promising emerging memory technologies potentially to replace conventional semiconductor memories such as SRAM, DRAM and flash [1]–[4]. RRAM uses the resistive effect of memristive devices (memristor) to store data permanently, even in the absence of the power supply. Moreover, memristor can be used as a storage element without requiring access transistors; this advantage enables a crossbar memory cell array structure. Furthermore, memory cell array can be stacked on the top of CMOS peripheral circuits creating three-dimensional ICs; the interconnects between the top and bottom layer are realized using CMOS-to-Nano Vias (CNVs) [5]. These novel devices and advanced circuit architecture offer attractive potentials such as enormous storage capacity, low power consumption, simple fabrication for memory cell array, etc [1]–[7]. Companies such as Hewlett-Packard and Hynix are expecting to market RRAM products in the near future [8].

Because of the above-mentioned potentials and commercialization efforts, research on such a memory technology is growing. To this end, most of the published work has focused mainly on the integration between CMOS and non–CMOS, fabrication techniques and reliability improvement using fault tolerance schemes [3], [4], [6], [7], [10], [12]–[15]. However, to the best knowledge of the authors, there is very limited work published on fault analysis, fault modeling and test development for RRAM. In [16] an electrical model for Metal-Insulator-Metal resistive memory element was proposed. In [17] a fault model (bridging faults) for memristor based memory and its test was introduced. In [18] and [19] faults models based on resistive opens injection and circuit simulation together with their tests were developed. Clearly research on fault modeling, test development and DFT for resistive memories (and memristor-based in particular) is still in its infancy stage. Understanding the faulty behavior of the memory devices in the presence of (manufacturing) defects will enable the development of appropriate fault models and efficient test schemes; thus, improve the outgoing product quality and reliability.

This paper presents fault analysis and modeling for open defects based on electrical simulation, introduces fault models and proposes test approaches for RRAMs. The fault analysis focuses on the RRAM cell array and CNVs as these parts distinguish RRAM from conventional memories such as SRAM and DRAM. The analysis reveals that unique faults occur besides the conventional memory faults; it also shows that march tests cannot guarantee the detection of such faults. Detecting these faults requires new test approaches. Therefore, design-for-testability (DfT) schemes are proposed to improve the defect coverage. Moreover, as the fault behavior may vary subject to process variations, the DfT schemes are made programmable to track the changes and improve the defect coverage. In short, the main contributions of the paper are:

- *S. Hamdioui and M. Taouil are with the Department of Computer Engineering, Delft University of Technology, 2628 CD Delft, The Netherlands. E-mail: {M.Taouil, S.Hamdioui}@Tudelft.nl.*
- *N.Z. Haron is with the Universiti Teknikal Malaysia Melaka, 76100 Melaka, Malaysia. E-mail: zaidi@utem.edu.my.*

Manuscript received 12 Jan. 2013; revised 14 Sep. 2013; accepted 07 Oct. 2013. Date of publication 17 Oct. 2013; date of current version 12 Dec. 2014. Recommended for acceptance by K. Chakrabarty.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2013.206

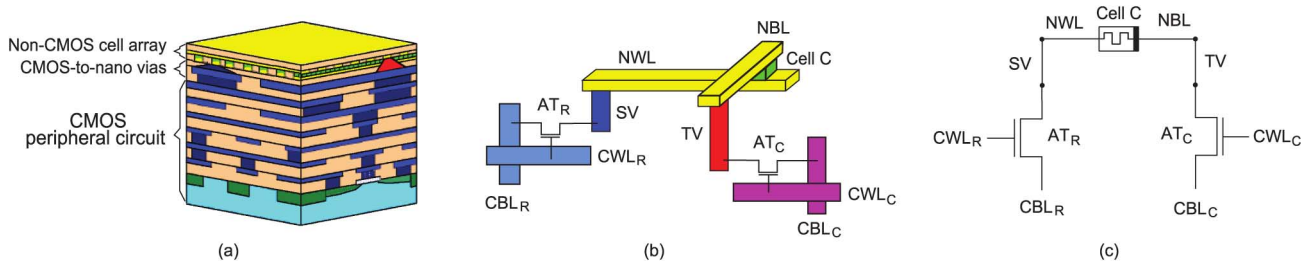

Fig. 1. (a) RRAM structure (b) a single RRAM cell connection (c) electrical equivalent circuit.

- A classification and definition of possible defects in RRAMs.
- An electrical RRAM model for defect injection and circuit simulation.
- An evaluation, analysis and fault modeling of open defects in RRAM cell array and CNVs.
- Two DfT schemes to detect resistive open defects in the RRAM cell array
- Two extended programmable versions of the two DfT schemes which can be tuned during pre-silicon testing to meet pre-determined targets and deal with the unexpected uncertainties.

The rest of the paper is organized as follows. Section 2 discusses the RRAM architecture and its operations. Section 3 classifies and defines possible defects in RRAM. Section 4 describes the simulation methodology and presents the experimental results of the performed defect injection and circuit simulation. Section 5 proposes two DfT schemes to target the observed faults. Section 6 extends the DfT schemes by making them programmable to improve the defect/fault coverage and deal with possible different defect strengths; a comparison of all proposed DFTs is also included. Section 7 concludes this paper.

2 RRAM Architecture and Operations

Today's dominant memory types such as DRAM, flash, and static RAM store data as charge. However, researchers strongly believe these charge-based memory cells have gotten nearly at their end as researchers are turning to storing bits as resistance instead; hence providing nonvolatile memories. There are mainly two competing categories of resistive RAM. The first one is phase-change memory PCM, which is based on heating up a material to change it from a polycrystalline to an amorphous state; thereby creating a measurable but reversible difference in the material's resistance. The second category is RRAM; it uses a voltage rather than heat to reversibly change the resistance. Compared with phase change, RRAM is still at a very early maturity level, but it seems to be more promising [1]. Not all RRAMs are alike. Each type uses a different underlying material with different properties, including access times, endurance, retention, and power consumption. Some types show very good data retention properties (which are essential for any nonvolatile memory), while other types exhibit very fast read/write times required for a DRAM-like main memory.

This section reviews one particular RRAM architecture; i.e., the memristor based RAM as it is the focus of this paper. The review includes RRAM structure, functional model, electrical model, and write and read operations. Similar to Flash memories, memristor based memory cell can be used to store a single bit information or more than a single bit of information (i.e., multi-level memory element), which is a promising application for the memristor device. The focus of this work is the use of a memristor as a "bistable" element; the results found can be easily extended to multi-level memristor device.

2.1 RRAM Structure

Fig. 1(a) shows the generic structure of an RRAM [5]. The memory consists of three main parts: (i) non-CMOS cell array, (ii) CMOS-to-Nano Vias (CNVs), and (iii) CMOS peripheral circuits. The top layer is the memory cell array formed by two sets of parallel nanowires crossing in perpendicular with bistable two-terminal devices (e.g., memristor) sandwiched at each crosspoint; the middle layer is the CNVs made of metal (e.g., copper, tungsten); the bottom layer consists of the peripheral circuits (e.g., decoders, sense amplifiers) structured from CMOS.

Fig. 1(b) illustrates how a single memristor memory cell (C) is connected to the external world; there are two connections groups: (a) the row group, and (b) the column group. The row group creates the connection to the lower-side terminal of the memory cell; it includes the nanowire word line NWL, the short CNV SV, the CMOS access transistor AT_R, the bit line CBL_R and the word line CWL_R. Conversely, the column group realizes the connection to the upper-side terminal of the memory cell; it includes the nanowire bit line NBL, the tall CNV (TV), the CMOS access transistor AT_C, the bit line CBL_C and the word line CWL_C. The electrical equivalent circuit of Fig. 1(b) is shown in Fig. 1(c).

2.2 RRAM Functional Model

Fig. 2 depicts the functional block diagram of the RRAM model proposed in [11] and [12]. The memory comprises: (a) non-CMOS cell array, (b) CMOS-to-Nano Vias (CNVs), and (c) CMOS peripheral circuits. The peripheral circuits consist of the functional units that are similar to those used in existing semiconductor memories. Both row and column decoders operate together to access the selected cells in the memory cell array. The write/read circuits supply appropriate voltage levels for write and read operations. The row multiplexers connect the write/read circuits to the memory cell array during write and read operations. The sense amplifiers sense read current, convert it into voltage and amplify it prior to sending the read value to data register. The column multiplexers connect the memory cell array to the sense amplifiers during read operations; they also connect the write

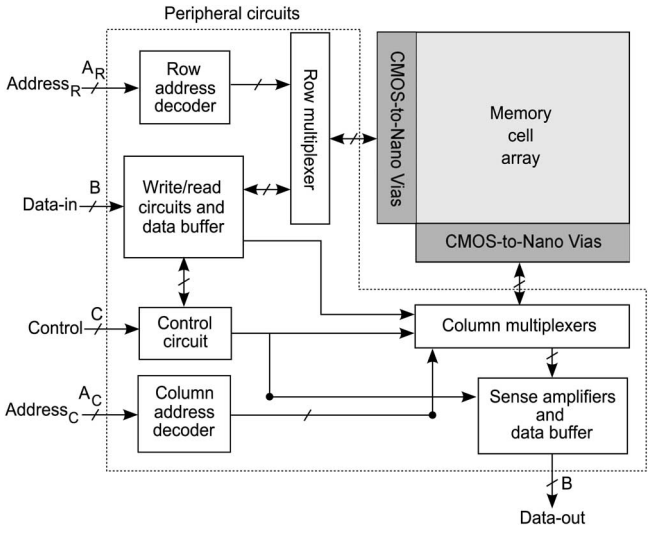

Fig. 2. Block diagram of RRAM functional model.

circuits and memory cell array during write 0 operations. All these operations are controlled by the control circuit.

2.3 RRAM Cell Electrical Model

RRAM cells are based on memristor fabricated using transition-metal oxide materials such as titanium oxide, tantalum oxide, nickel oxide, etc [10], [12]. A memristor is a two-terminal non-linear nanodevice, which retains its internal resistance referred to as memristance even when power is switched off. This capability enables memristors to be used for non-volatile memory applications.

2.3.1 Memristor Electrical Model

Based on the theoretical formulation by L. Chua in 1971 [9], a group of Hewlett-Packard (HP) researchers introduced the memristor prototype and electrical model in 2008 [10]. The HP's memristor is fabricated using a thin film of titanium oxide injected with different oxygen atom doping rates, resulting in *doped* and *undoped* layers, as shown in Fig. 3(a). The doped layer (with thickness w) corresponds to a low memristance R_{ON}, while the undoped layer (with thickness D-w) corresponds to a high memristance R_{OFF}, where D is the total thickness of the memristor.

When a positive voltage is applied to the positive terminal of the memristor (the top terminal in Fig. 3) while grounding the negative terminal (the bottom terminal), the dopants drift toward the undoped layer; thereby reducing the memristance. When the polarity of the supply voltage is reversed, the dopants drift toward the doped layer; thereby increasing the memristance. If now the voltage supply is removed, then

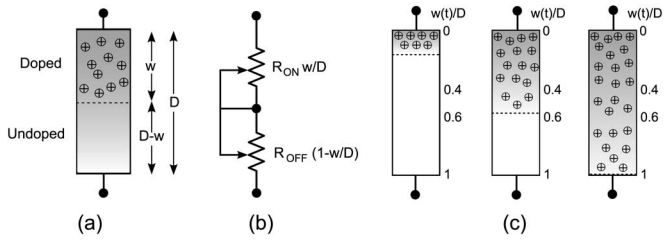

Fig. 3. Memristor (a) model (b) equivalent electrical circuit (c) logic 0, undefined, logic 1 definition.

the dopants remain at their last position; hence, setting the memristance at a specific value. Fig. 3(b) shows the equivalent electrical circuit of a memristor modeled as a coupled variable resistor where each resistor is coupled with its respective doped and undoped layer thickness (normalized to total memristor thickness) [10].

The current moving through a memristor is proportional to the flux of the magnetic field flowing across the device. Depending on the biasing sources, either current or voltage source, memristor can be controlled by charge or flux [12]. Since memory typically uses voltage at the source, flux controlled memristor is considered and modeled. The voltage-current relation of HP's memristor is given as follows [10].

$$\begin{aligned} v(t) &= i(t) \cdot M(t), \\ v(t) &= i(t) \cdot \left\{ R_{ON} \cdot \frac{w(t)}{D} + R_{OFF} \cdot \left(1 - \frac{w(t)}{D}\right) \right\}, \end{aligned} \tag{1}$$

where $M(t)$ is the total memristance and $w(t)$ is the thickness of the doped layer of a memristor at time t.

To satisfy the flux controlled model, $w(t)$ is normalized to D [13]; it is referred to as the memristor internal state and expressed as follows.

$$\frac{w(t)}{D} = 1 - \sqrt{1 - \frac{2 \cdot \mu_v \cdot \Phi(t)}{\beta \cdot D^2}}, \tag{2}$$

where $\Phi(t) = \int_0^t v(\tau) d\tau$ is the injected flux across the memristor for a time t, μ_v is the average dopant mobility and $\beta = \frac{R_{OFF}}{R_{ON}}$.

2.3.2 Logic State Definition

The memristor internal state $\frac{w(t)}{D}$ is the metric used to define the logic value hold by a memristor [10]. The value of $\frac{w(t)}{D}$ represents the change in the physical properties (dopants drift) of the memristor and is inversely proportional to its memristance. Because the targted RRAM in this work operates in binary numbers, yet memristors used as RRAM cells are analog devices, the logic states of the memristor have to be defined. For simplicity and similar to [12], we define a memristor is at *logic 0* when $0 \leq \frac{w(t)}{D} \leq 0.4$ and *logic 1* when $0.6 \leq \frac{w(t)}{D} \leq 1$; see Fig. 3(c). The corresponding ideal low and high levels are $\frac{w(t)}{D} = 0$ and $\frac{w(t)}{D} = 1$, respectively. In reality, one has to account for possible noise injections by adding safety margins for each logic value. The region in between (in this case $0.4 \leq \frac{w(t)}{D} \leq 0.6$) is an unsafe region that should be avoided for read and write data integrity.

2.4 RRAM Write and Read Operations

The selected cell is written and read by applying appropriate voltages across the cell. Fig. 4(a) shows a 2×2 cell array with four access transistors; the figure lists the required voltages for the bit lines and word lines when a write 1 operation is applied to the selected cell C_{11}; see Fig. 4(b) for the timing diagram.

For a write 1 operation, the $Write/\overline{read}$ *enable* signal is set to high to initiate the write operation. During the write operation, the nanowire word line NWL_{sel} is set to V_{dd} (by activating the access transistor AT_{R1}) and the nanowire bit line

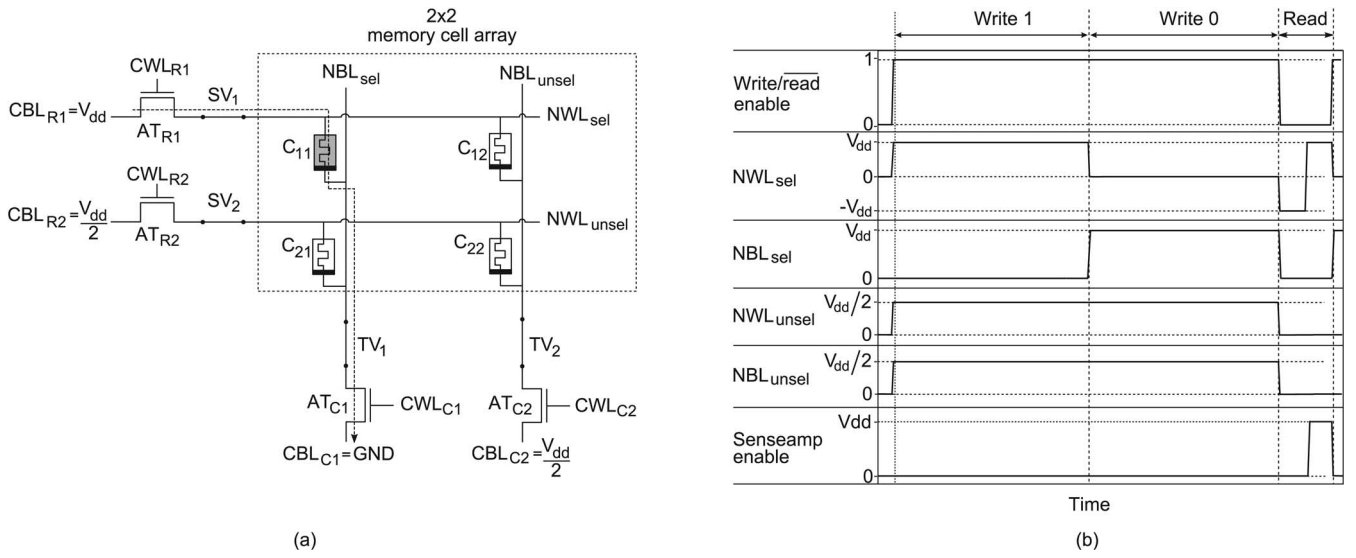

Fig. 4. (a) Write 1 operation to a single RRAM cell (b) timing diagram for RRAM write and read operations.

NBL_{sel} to GND (by activating the access transistor AT_{C1}) [11]. With this biasing condition, the write voltage is equal or larger than the threshold voltage of C_{11} (i.e., $V_{dd} > V_{th}$), which enables the write operation. At the same time, the unselected nanowire word line NWL_{unsel} and nanowire bit line NBL_{unsel} are biased each with $\frac{V_{dd}}{2}$, preventing any voltage drop across unselected cells.

Write 0 operation is performed by biasing the selected cell with $NBLsel$ set to V_{dd} and NWL_{sel} grounded, while NWL_{unsel} and NBL_{unsel} are both set to $\frac{V_{dd}}{2}$; see Fig. 4(b).

For a read operation, the $Write/\overline{read}$ *enable* signal is first set to low to initiate the read operation. During the first half of the read operation, the selected cell is biased with -V_{dd}, and in the second half with +V_{dd}; see the waveform of NWL_{sel} in Fig. 4(b). At the same time, NBL_{sel} is connected to the sense amplifier, and both NWL_{unsel} and NBL_{unsel} are left floating. During the second half of the read operation, the *Senseamp enable* signal is activated to sense the read-current, convert it to voltage, amplify it and send it to output data register. It is worth noting that the first phase of the read operation is a destructive process (i.e., changes the memristance of the device); therefore, the second phase is needed to get the cell to its original state (i.e., $\frac{w(t)}{D} = 1$ in case of read 1); see also Fig. 3.

To guarantee that a logic value is appropriately written in a cell (i.e., changing the memristance value of the device to an appropriate value), a write operation needs to take an ample write time T_{write}. For a given write voltage value V_{write} (put on the terminal of the device), the required time for an appropriate write operation is [12]:

$$T_{write} = \left| \frac{\phi(t)}{V_{write} \cdot R_{OFF}^2} \cdot (R_{OFF}^2 - R_{ON}^2) \right|, \tag{3}$$

where $\phi(t)$ is the *effective* flux for $0 < D < 1$ expressed as follows.

$$\phi(t) = \frac{(\beta \cdot D)^2}{2 \cdot \mu_v \cdot (\beta - 1)}. \tag{4}$$

In contrast to a write operation, a read operation requires a shorter time duration T_{read} than the write operation T_{write}; this short time is used to alleviate a serious change in the memristor internal state that might lead to soft errors [12].

It is worth noting that the used model here, which is based on [12], does not accurately mimic the real behavior of a memristor device, especially for extreme cases. For instance, at negligible voltage the memrirsor should keep its state unchangeable irrespective of the amount of writing time; however, the model is not able to correctly incorporate this.

3 Defect Classification and Definition

Defects in memory circuits are the physical structures that deviate from the intended layout design and caused by imperfection in the fabrication process. They introduce unintended disconnections or connections in the memory. Defects such as broken or missing metal lines, extra metal lines, etc. can be modeled at the electrical level using a resistor as follows [20]:

- **Open:** This is an unintended series resistance R_{op} within a connection in the range of $0 < R_{op} \leq \infty\Omega$.
- **Bridge:** This is an unintended parallel resistance R_{br} between two connections in the range of $0 < R_{br} \leq \infty\Omega$.
- **Short:** This is an unintended resistive path R_{sh} between a node and supplied voltage V_{dd}, or ground GND. The short resistance can be in the range of $0 < R_{sh} \leq \infty\Omega$.

Table 1 gives the classification of the defects in the three main parts of RRAM. For the targeted memory architecture, only opens and bridges may occur in the memory cell array and CNVs; shorts are not possible as these memory parts are neither directly connected to the supply voltage nor to the ground (see Fig. 1). In the peripheral circuits, opens, bridges as well as shorts may occur. In this paper, only defects in the memory cell array and CNVs are discussed because these two parts are those that distinguish RRAM from the existing semiconductor memories. Defects in the peripheral circuits are similar to that in conventional RAMs [23]–[25]. In the rest of this section, defects within the memory cell array and in CNVs will be discussed.

TABLE 1
List of Defects

Part	Classification	Location	Notation
Memory cell array	Open	Within cells	OC
		At nanowire bit lines (NBLs)	OB
		At nanowire word lines (NWLs)	OW
	Bridge	Between NBLs	BB
		Between NWLs	BW
		Between NBLs and NWLs	BBW
CMOS-to-Nano Vias	Open	Within short CNVs	OSV
		Within tall CNVs	OTV
	Bridges	Between short CNVs	BSV
		Between short and tall CNVs	$BSTV$
Peripheral Circuits	Open	At gate, source and drain of transistor, interconnects	OPC
	Bridge	Among gate, source, drain and substrate of transistors, interconnects	BPC
	Short	At gate, source and drain of transistor, interconnects	SPC

3.1 Defects in Memory Cell Array

Defects in the memory cell array can be either opens or bridges and are summarized in Table 1 and briefly explained next. Opens can exist within the memory cells (i.e., memristor), at nanowire bit lines (NBLs) and nanowire word lines (NWLs). Conversely, bridges can be present between NBLs, between NWLs as well as between NBLs and NWLs. Although bridges can impact an arbitrary number of adjacent nanowire lines regardless of their distance, only those involving physically adjacent nanowires are considered here as they have the highest occurrence probability.

3.1.1 Opens within Cells, at NBLs and at NWLs

Opens create an increased connection resistivity of the affected locations. For example, the defective cell C_{11} in Fig. 5 may suffer from insufficient dopant [13], [31] that will cause its resistance to be extremely high (modeled as OC). An open in NBLs and NWLs may occur due to missing material, broken nanowire, etc. This may be a result of an excessive force of the nanoimprint lithography process, doping variation in the nanowire, or device thickness fluctuation due to variations in nanoimprint lithography and patterning process [6], [22]. For example, an open defect at NBL_2 (OB) in Fig. 5 may affect operations to cells along the nanowire bit line such as C_{12}. Consequently, depending on the severity of the induced resistance, write and read operations to one or more cells might be affected.

3.1.2 Bridges between NBLs, between NWLs, and between NBLs and NWLs

Bridges create unintended connections between NBLs, between NWLs, or between NBLs and NWLs. These defect types may occur due to, e.g., an unintended wider nanowire connected to its adjacent nanowire [6]. For example, a bridge defect between NBL_1 and NBL_2 (BB) in Fig. 5 may cause the content of cell C_{21} to flip when an operation is performed to C_{22}. Likewise, a bridge defect between NWL_1 and NWL_2 (BW) may cause a fault in C_{22} when an on operation is performed to C_{12}, etc.

3.2 Defects in CMOS-to-Nano Vias

Defects in CNVs can be either opens or bridges [7], [21], [22].

3.2.1 Opens in CNVs

Opens in short CNVs and in tall CNVs may occur due to, e.g., broken or crack as the result of excessive polishing during fabrication, void after filling, misalignment, poor contact between the topedge of tall (short) CNVs and NBL, etc. [7], [21], [22]. For example, the open defect at SV1 (OSV) in Fig. 6, may cause a fault when operations are performed to C_{11} and/or C_{12}. Likewise, an open defect at TV1 (OTV) may cause a fault when operations are performed to C_{11} and/or C_{21}.

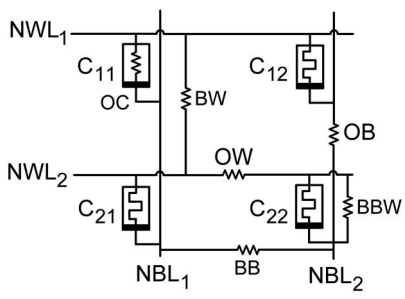

Fig. 5. Possible defects within the memory cell array.

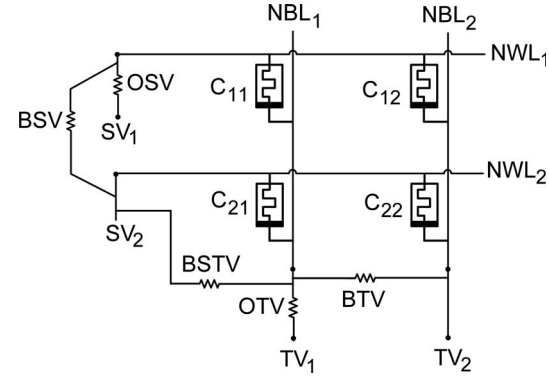

Fig. 6. Possible opens and bridges in CNVs.

3.2.2 Bridges between CNVs

A bridge between short CNVs, between tall CNVs and between short and tall CNVs may occur due to excessive misalignment as a result of, e.g., imprecise mask [7], [21]. For example in Fig. 6, a bridge defect between SV1 and SV2 (BSV) may cause a fault in C_{21} when operations are performed to C_{11}. Likewise, a bridge defect between TV1 and TV2 (BTV) may cause a fault in C_{22} when operations are performed to C_{21}.

4 Defect Injection/Circuit Simulation

This section focuses on the analysis of open defects as their occurrence probability is steadily increasing compared with that of bridges [26]. Estimating the occurrence probability of each open defect targeted in this work is impossible as there is no data published data. Nevertheless, one can argue that some defects of Table 1 have large impact and/or are more important than others. For instance, open defects on bit lines (OB) have more effect than opens on word line (OW), as the sensing circuitry (which is very sensitive) is placed at the end of the bit lines. In the rest of this section, first the simulation model and methodology is explained; then, the simulation results for defect-free memory are presented. Thereafter, the simulation results of open defects and their analysis are provided.

4.1 Simulation Model and Methodology

HSPICE simulation is used to perform defect injection and circuit simulation. Since this requires too much simulation time for a complete memory, an appropriate simulation model is built, which both accurately describes the behavior of the memory and only requiring a reasonable simulation time. A simplified version of Fig. 2 is build with 2x2 memory array matrix. The required peripheral circuits (address decoders, write drivers, sense amplifiers, etc,) are included in the model; they are synthesized using 45 nm PTM transistor models. In addition, the build model makes use of the memristor electrical model proposed by HP, which is already explained in Section 2.3. The specifications of the RRAM cell model used in the simulation are as follows: $R_{ON} = 100$, $R_{OFF} = 100\text{k}\Omega, D = 3\text{nm}, \mu_v = 3 \cdot 10^{-8} m^2 v^{-1} s^{-1}$ and $V_{write} =$ 1.5 V; hence a supply voltage Vdd = 1.5 V [12], [13]. Using these specification values, write time $T_{write} = 100$ ns is obtained; see Eq. 3. The read operation T_{read} is set to 20 ns as it is sufficient for sense amplifiers to sense the read current and to avoid soft errors [12]; see Fig. 4 for the overall timing of the the control signals used for simulation.

To ensure that the electrical RRAM model works correctly for defect injection and circuit simulation, it must be verified. Therefore, a defect-free simulation has been performed using the sequence $S = w1, r1, w0, r0$; note that $w1(w0)$ denotes a write 1(0) to the cell under consideration, while $r1(r0)$ denotes a read operation with the expected value 1(0). The sequence S is applied and verified after initializing the cell to 0.

After verifying the correctness of the model in the absence of defects, a defect injection and simulation is performed. Opens within the memory cell array and within CMOS-to-Nano vias are considered for the simulation; see Table 1. These consist of: (a) an open within memory cell OC, (b) an open at nanowire bit line OB, (c) an open at nanowire word line OW, (d) an open in short via OSV, and (e) an open in tall via OTV. These opens are modeled using a series resistor injected -one at a time- at the considered locations; the resistor values are swept between $0 < R_{op} \leq \infty\Omega$. Note that for fault modeling, it is not required to perform defect simulation for all opens [20] due to the symmetrical structure of the memory cell (see Fig. 1); e.g., the effect of an open at *NWL* and an open at *NBL* have similarities, although their strengths are different. For instance, resistive open defects have more effect on bit lines rather than word lines as the sensing circuitry is placed at the end of bit lines. The memory operations used during the simulation consist of:

- $0w1$: write 1 to a cell initialized to 0.
- $1w0$: write 0 to a cell initialized to 1.
- $1r1$: read an expected value 1 from a cell.
- $0r0$: read an expected value 0 from a cell.

For each simulation the value of injected defect is swept and the operation is inspected to check if it fails or not. The boundaries are values at which an operations start to fail.

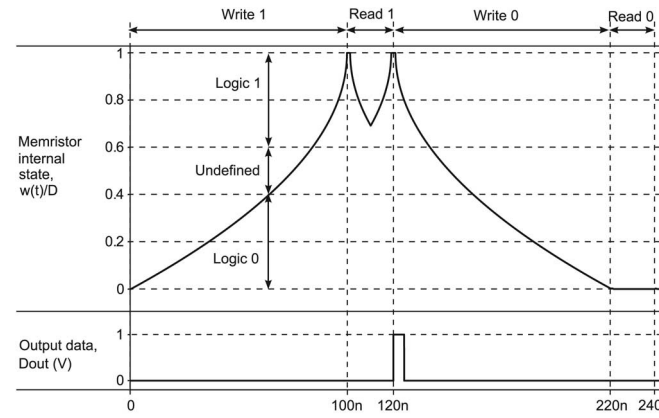

Fig. 7. Simulation results of defect-free case.

4.2 Defect-Free Simulation Result

Fig. 7 shows the defect-free simulation results for the sequence $S = w1, r1, w0, r0$. The figure consists of two graphs: (a) the top graph describing the internal state of the cell (memristor) $\frac{w(t)}{D}$, which corresponds to the logic value being written or read, and (b) the bottom graph showing the output data value sent to the data register.

- For write 1 operation, $\frac{w(t)}{D}$ increases slowly (almost linearly) starting from 0. However, once $\frac{w(t)}{D}$ approaches 0.6, it increases quickly (exponentially) to reach 1. This is due to a slower ionic drift at smaller $\frac{w(t)}{D}$ values [10].
- For read 1 operation, $\frac{w(t)}{D}$ decreases from 1 to 0.64 before increasing back to 1. During second read period, the read current is sensed, converted to voltage and sent to output data register at $t = 120ns$. Note that the first phase of the read operation is a destructive process; therefore, the second phase is needed to restore the initial state $\frac{w(t)}{D} = 1$ of the cell [see also the timing of the read operation in Fig. 4(b)].
- For write 0 operation, $\frac{w(t)}{D}$ decreases from 1 to 0 showing a symmetrical shape to that of write 1 operation.
- For read 0 operation, $\frac{w(t)}{D}$ remains at 0. The read value is sent to output data buffer at $t = 240ns$.

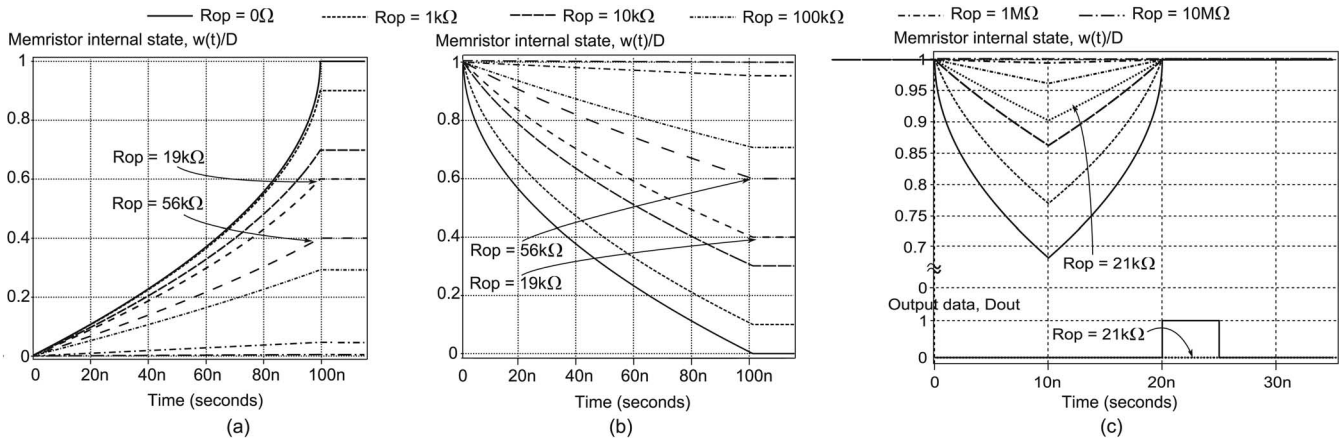

Fig. 8. Resistive open cell behaviors under (a) $0w1$ operation (b) $1w0$ operation (c) $1r1$ operation.

4.3 Open Defect Simulation Results

Due to space limitation, only simulation results for the defect OC (within the cell) are provided in details. Thereafter, simulation results for other open defects will be briefly reported; more results can be found in [41].

Fig. 8(a) shows the simulation results of the $0w1$ sequence for OC for different R_{op} values injected into the cell. It reveals that the maximum value of the internal state at the end of write operation decreases as R_{op} increases, and it even enters an undefined state for $19k\Omega \leq R_{op} \leq 56k\Omega$ (since $0.64 < \frac{w(t)}{D} < 0.4$). For $R_{op} > 56k\Omega, \frac{w(t)}{D} \leq 0.4$ meaning that the write operation fails to set the defective cell to logic 1. We will refer to resistance defect values at which the memory cell enters the undefined or the incorrect state with *critical values*; e.g., $19k\Omega$ and $56k\Omega$ are two critical values for the performed $0w1$ sequence. Fig. 8(b) presents the simulation results of $1w0$ sequence for different R_{op} values; it clearly shows that depending on the defect value, the cell may fail to undergo a down write transition ($1w0$) or enter the undefined state. Fig. 8(c) reports the simulation results of $1r1$ sequence for different R_{op}; it indicates that the defective cell returns an incorrect logic value while keeping the correct stored data when $R_{op} > 21k\Omega$. The sequence $0r0$ is also performed on a defective cell but the results did not show any faulty behavior.

The simulation results for an open defect in NWL (OW) exhibit similar faulty behaviors at slightly different critical values. When performing $0w1$ sequence, the cell enters the undefined state for defects with values $14k\Omega \leq R_{op} \leq 49k\Omega$ and fails to undergo an up/down transition write operation for $R_{op} > 49k\Omega$. In addition, when performing the $1r1$ sequence, the cell returns an incorrect logic value while keeping the correct stored data for $R_{op} > 17k\Omega$. For the sequence $0r0$, no faulty behavior is observed.

4.4 Simulation Analysis

Table 2 summarizes the observed faulty behaviors of the defective RRAM cell including their corresponding fault models. The analysis reveals that depending on the defect values, two types of faults can occur:

- **Conventional faults**: these consists of, e.g., *Transition Faults* (TF_0 and TF_1), *Stuck at Faults* (SAF_0 and SAF_1) and *Incorrect Read Fault* (IRF_1) [27]–[29]. These faults *can* be detected using existing march tests.
- **Unique faults**: these consist of e.g., Undefined State Faults (USF_0 and USF_1). Detecting such faults cannot be guaranteed with existing march tests because the faults cause a random logic value to be read from defective RRAM cells, while march tests deal only with fixed, predetermined patterns of logic values. Therefore, a special design-for-testability (DfT) scheme is required; it will be described next.

It is worth noting that other architectures of memristor-based memories exist, which are somehow different than the one one analyzed in this work. For example, memristor crossbar based on 1T1 m (1 Transistor – 1 Memristor) architecture [40]. Nevertheless, the unique fault models found here are also applicable for other architectures as these faults are caused also by defects in the memristor itself (defect OC in Table 1).

5 Design-for-Testability Schemes

Classical DFT methods for memories use mainly Built-In-Self Test (BIST) to detect memory faults [34]–[38]. They all are based on implementing deterministic patterns such as march tests at different levels; e.g. algorithm level, march element level, etc [39]. However, none of them is able to guarantee the detection of Undefined State faults as these faults cause

TABLE 2
Observed Functional Fault Models Due to Open Defect within the Memory Cell OC

Sequence	Defect value Ω	Faulty behaviors	Fault models
$0w1$,$1w0$	$19k \leq R_{op} \leq 56k$	The cell is set to an undefined state by a write 0 (1) operation	USF_0 (USF_1)
	$R_{op} > 56k$	The cell fails to undergo a down-transition (up-transition) when write 0 (write 1) is performed	TF_0 (TF_1)
	$R_{op} > 10M$	The cell is always stuck at logic 0 (logic 1)	SAF_0 (SAF_1)
$1r1$	$R_{op} > 21k$	The cell returns an incorrect read logic value 0 while it keeps its correct logic 1	IRF_1

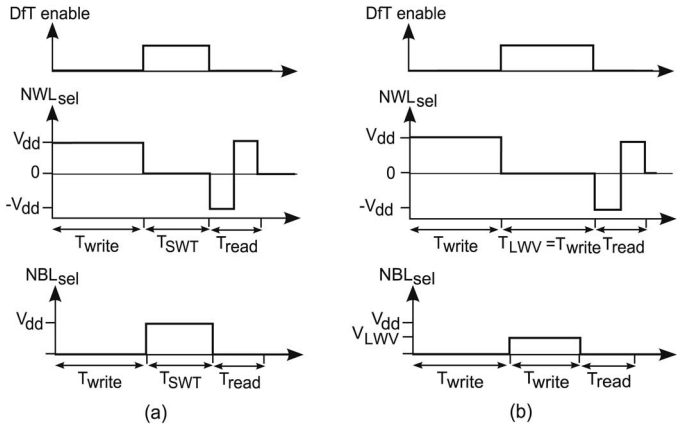

Fig. 9. Control signals for (a) SWT (b) LWV schemes.

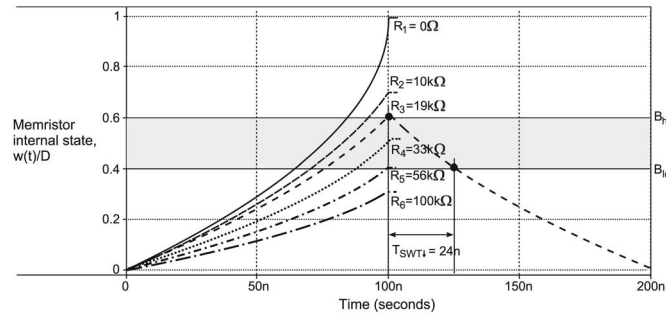

Fig. 10. Simulation results to determine $T_{SWT\downarrow}$.

random logic values to be read, while march tests deal only with predetermined patterns of logic values. Therefore new DFTs/ test approaches are needed.

A possible way to improve the fault coverage of march tests and detect the Undefined State Faults is to stress the cell in such a way that:

- If the cell is faulty (i.e., undefined state), then stressing has to shift the cell's state from an undefined state to an incorrect state. Performing a read operation after stressing the cell will detect the fault.
- If the cell is fault-free, then it has to remain in its correct state. Otherwise, the stress may lead to overkill and yield loss.

The state of an RRAM cell is determined by the *duration* and the *strength* of the injected flux across the memristor device. Hence, the RRAM operations rely mainly on the *duration of access time* and the *voltage value* applied to the terminals of the cells [10], [12], [13]. Changing any of these parameters results in a test stress. We will exploit these two parameters to develop two DfT schemes: Short Write Time based DfT and Low Write Voltage based Dft; they are discussed next.

5.1 Short Write Time-Based DfT

In this section first the DfT concept is presented. Then, the DfT circuitry is given. Finally, the experimental results are presented.

5.1.1 DfT Concept

The concept of this DfT is based on changing the *duration of access time*. Every normal write operation requires a specific write access time; e.g., $100ns$ in Fig. 7. If the access time is reduced, the cell will not have enough time to change its state from logic 0 (i.e., $\frac{w(t)}{D} = 0$) to logic 1 (i.e., $\frac{w(t)}{D} = 1$) or vice versa; see Fig. 7. However, if the cell was already in the undefined state (i.e., $0.4 < \frac{w(t)}{D} < 0.6$), then applying a write operation with a reduced write time will push the cell to shift from an undefined state to a defined state. This scheme, referred to as *Short Write Time* (SWT), is illustrated in Fig. 9(a). During a normal mode, a write operation (say $w1$) is applied by putting the NWL_{sel} to V_{dd} for the normal access time T_{write}; see also Fig. 4(b). Thereafter, the DfT mode is enabled and a write operation, referred to as *weak write operation* (say $\widehat{w}0$), is applied by setting the NBL_{sel} to V_{dd} for a shorter access time T_{SWT}. The DfT mode is then set-off and a normal read operation is applied (say $r1$). If the cell suffers from an open defect, then the $w1$ operation will put the cell in an undefined state and the $\widehat{w}0$ operation will shift the cell's state to 0. The $r1$ operation will return an incorrect value 0 instead of 1 and the fault will be detected. If the cell is fault-free, then the weak write will not change the cell's state and the read operation will return a correct value.

To detect open defects within RRAM cells one can apply an appropriate test algorithm (such as march test), but by incorporating the above DfT schemes, as it is done in [30] for SRAM. Remind that using a march test without DFT will be not able to guaranteed the detection of undefined state faults. The following test procedure can be applied. Note that the test length is $8n$ where n is the memory size.

1. Initialize memory cells to 1.
2. Write 0 to memory cells.
3. Activate SWT based DfT and apply weak write 1; only defective cells will flip to 1.
4. Deactivate the DfT and read 0 from the cells. If the read value is 1, then faults are detected.
5. Repeat Step 1 to 4 with complementary data values, e.g., initialize the cells to 0, etc.

Note that the above algorithm can be written as a combination of two sequences: $1w0\widehat{w}1r0$ (steps 1 to 4) and $0w1\widehat{w}0r1$ (step 5).

5.1.2 DfT Circuit

The DfT scheme is based on reducing the write access time while maintaining the level of write voltage at V_{dd}. Identifying the duration SWT of weak write is crucial; it has to detect faulty cells but at the same time prevent overkill. To accomplish this, a two-step process is used.

- First, perform defect injection and circuit simulation for different defect values in order to identify the critical value of the open R_{cr} that puts the cell in a boundary state, i.e., between the defined correct state and the undefined state. For example, as shown in Fig. 10, when performing $0w1$ in the presence of a defect, the R_{cr} found to be $R_{cr\downarrow} = R_3 = 19k\Omega$ corresponding to $\frac{w(t)}{D} = 0.6 = B_{high}$. Similarly we found when simulating $1w0$ operation that $R_{cr\uparrow} = R_2 = 19k\Omega$ corresponding to $\frac{w(t)}{D} = 0.4 = B_{low}$.
- Second, use R_{cr} to determine the weak write duration that will cause the cell to shift from B_{high} to B_{low} and vice versa. For example, for $R_{cr\downarrow} = R_3$ in Fig. 10, the required duration $T_{SWT\downarrow}$ of $\widehat{w}0$ to shift the state cell from $\frac{w(t)}{D} = 0.6$ to $\frac{w(t)}{D} = 0.4$ is $24ns$. Similarly, the required duration $T_{SWT\uparrow}$

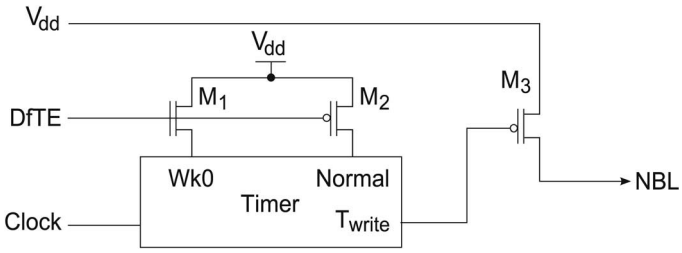

Fig. 11. Schematic of SWT circuit.

for the $\widehat{w}1$ operation to shift $\frac{w(t)}{D}$ from B_{low} to B_{high} found to be $43ns$.

Two SWT circuits are required to perform $\widehat{w}0$ and $\widehat{w}1$ operations. The SWT circuits are embedded inside the write circuit shown in Fig. 2. Fig. 11 depicts the concept of the SWT circuit that provides the required control signals for normal and $\widehat{w}0$ operations. The NMOS transistor $M1$ and the PMOS transistor $M2$ are used to switch between the normal mode and DfT mode. When the $DfTE$ signal is set to low, $Timer$ will set up the T_{write} signal to high for a normal write operation. However, when the $DfTE$ signal is set to high, $Timer$ will provide the required duration of $\widehat{w}0$. Subsequently, the T_{write} signal switches transistor $M3$ on allowing $\widehat{w}0$ to be supplied.

5.1.3 Simulation Results

To evaluate the proposed scheme, the sequence $1w0\widehat{w}1r0$ is performed for three open resistance values: (a) $R_{op} = 0\Omega$, (b) $R_{op} = 19k\Omega$, and (c) $R_{op} = 33k\Omega$; the sequence presents the first part of the test algorithm mentioned in the previous section. Fig. 12 reports the simulation results.

- $R_{op} = 33k\Omega$: the $\widehat{w}1$ operation will shift the cell's state $\frac{w(t)}{D}$ from the undefined state (at $t = 100ns$) to state 1 (at $t = 143ns$). Hence, a read 0 operation will easily detect the fault.
- $R_{op} = 19k\Omega$: this is the critical value. The $\widehat{w}1$ operation will shift $\frac{w(t)}{D}$ from B_{low} to B_{high}. Although the read operation returns an incorrect value in this experiment, typically the read value can be considered random as the cell's state is at the boundary. It is worth noting that defects with values less but close to the the critical value will end in the undefined state after $\widehat{w}1$ operation; the simulation results showed that this is the case for defects between $\sim 10k\Omega$ and $19k\Omega$. Operations on cells with such defects may pass correctly. Nevertheless, they cause weak faults; a weak fault is a fault that escapes the test program (because it does not cause an error/system failure). Cells with weak faults are potentially the first ones which will cause faults in field as any small degradation could strength the weak faults. Hence, the DFT scheme can even used to detect cells suffering from reliability problems. Section 6 will show how multiple stress strength settings can be generated to tune the stress to suit the targets.
- $R_{op} = 0\Omega$: the healthy cell keeps its state 0; hence the cell passes the test.

One can conclude that due to the DfT scheme, the detection of any open defect with $R_{op} > 19k\Omega$ will be guaranteed.

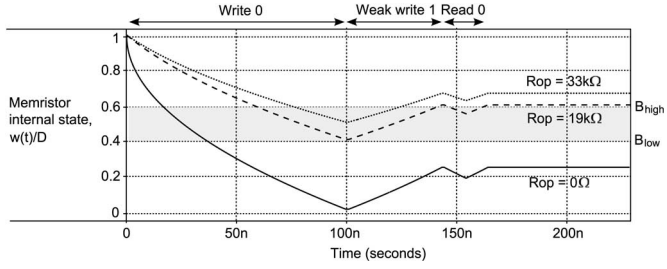

Fig. 12. Simulation results of $1w0\widehat{w}1r0$ using SWT.

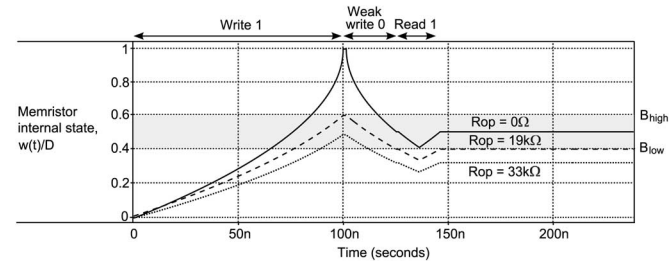

Fig. 13. Simulation results of $1w0\widehat{w}1r0$ using SWT.

Fig. 13 shows the result of performing the sequence $0w1\widehat{w}0r1$, which describes the second part of the test algorithm described in the previous section.

- $R_{op} = 33k\Omega$: the $\widehat{w}0$ operation will shift $\frac{w(t)}{D}$ from the undefined state to state 0. Hence, a read 1 operation will easily detect the fault.
- $R_{op} = 19k\Omega$: this is the critical value. The $\widehat{w}0$ operation will shift $\frac{w(t)}{D}$ from B_{high} to B_{low}. Also here even the read operation returns an incorrect value, the read value can be considered random as the cell's state is at the boundary.
- $R_{op} = 0\Omega$: the cell is supposed to keep its state 1. However, the simulation results reveal that the state of the healthy cell is also shifted to the undefined state, which may cause *overkill* and *yield loss*. Because the cell (memristor) is initially fully doped, $\widehat{w}0$ causes a faster ionic drift leading to a quick drop of the cell's state; see also Section 4.2.

Overall the simulation results clearly show that the proposed DFT suffer from limitations. The DFT provides a fixed stress strength (i.e., fixed short write time); this is determined based on the best available pre-silicon design information. Therefore it may overstress the memristor test for the targeted defects. To prevent this drawback, the DfT should be able to perform testing at multiple stress settings (short write times). Hence, DFT should be designed to support such a flexibility and during a post-silicon testing, a suitable setting stress is selected based on pre-determined targets. This will be be discussed in Section 6.

5.2 Low Write Voltage-Based DfT

In this section first the concept of Low Write Voltage based DfT is presented. Then, the DfT circuitry is given. Finally, the experimental results are reported.

5.2.1 DfT Concept

The concept of this DfT is based on changing the *applied voltage value*. Every normal write operation requires a specific write voltage value; this is set to $1.5V$ in our case. If the write voltage is reduced, the induced electric field will not be sufficient to

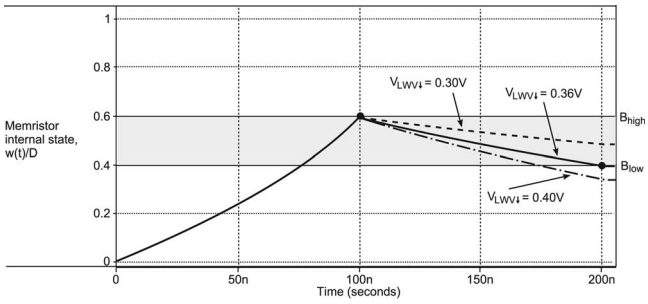

Fig. 14. Simulation results to determine $V_{LWV\downarrow}$.

change the cell state from logic 0 to logic 1 or vice versa. However, if the cell was already in the undefined state, then applying a write operation with a reduced write voltage will push the cell to shift from the undefined to the defined state. This scheme, referred to as *Low Write Voltage* (LWV), is illustrated in Fig. 9(b). During a normal mode, a write operation (say $w1$) is applied by setting NWL_{sel} to V_{dd} for the normal access time duration T_{write}. Thereafter, the DfT mode is enabled and a write operation, referred to as *weak write operation* (say $\widehat{w}0$), is applied by putting the NBL_{sel} to a reduced supply voltage V_{LWV}. The DfT mode is then set-off and a normal read operation is applied (say $r1$). If the cell suffers from an open defect, then the $w1$ operation will put the cell in an undefined state and the $\widehat{w}0$ operation will shift the cell's state to 0. The $r1$ operation will return an incorrect value 0 instead of 1 and the fault will be detected. If the cell is fault-free, then the weak write will not change the state of the cell and the read operation will return a correct value.

To detect opens using LWV based DFT, a similar test algorithm –as that used with SWT based DFT– can be applied; it combines the two sequences: $1w0\widehat{w}1r0$ and $0w1\widehat{w}0r1$.

5.2.2 DfT Circuit

Similar steps as those used for the SWT scheme are applied to determine the appropriate write voltages. Each time an open defect is injected, a different V_{LWV} value is supplied and the sequences $0w1\widehat{w}0$ and $1w0\widehat{w}1$ are performed. When $\frac{w(t)}{D}$ is shifted to the desired boundary (B_{high} or B_{low}), then the required write voltage is found. For example, Fig. 14 shows the the results for $0w1\widehat{w}0$ for $R_{op} = 19k\Omega$; note that in order to shift $\frac{w(t)}{D}$ from B_{high} to B_{low} after $100ns$ (nominal time), the required write voltage is $V_{LWV\downarrow} = 0.36V$. Similarly, we found that to shift $\frac{w(t)}{D}$ from B_{low} to B_{high}, the write voltage of $V_{LWV\uparrow} = 0.64V$ is needed.

Two LWV circuits are required to perform $\widehat{w}0$ and $\widehat{w}1$ operations. Fig. 15(a) shows the concept of such circuits. The value of $DfTE$ signal together with the two MOS transistors

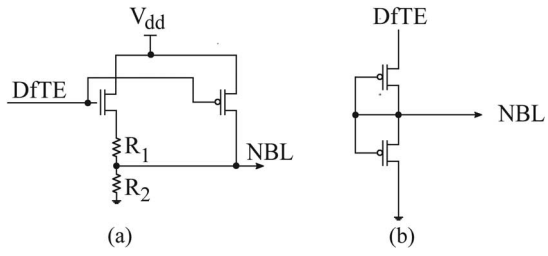

Fig. 15. (a) LWV circuit concept (b) implementation.

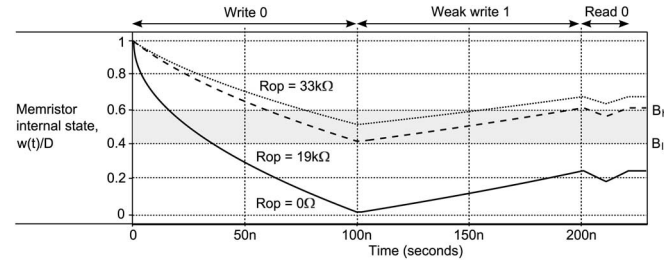

Fig. 16. Simulation results of $1w0\widehat{w}1r0$ with LWV.

determine either V_{dd} or $\frac{V_{dd} \cdot R_2}{R_1 + R_2}$ will provided to NBL as write voltage. Fig. 15(b) shows a possible implementation scheme [32]. The desired voltage can be used to determine the transconductance ratio of the NMOS and PMOS transistors [32]:

$$\frac{\beta_n}{\beta_p} = \left(\frac{DfTE - V_{LWV\downarrow} - |V_{Tp}|}{V_{LWV\downarrow} - |V_{Tn}|} \right)^2. \quad (5)$$

Here, $|V_{Tp}|$ and $|V_{Tn}|$ are the PMOS and NMOS threshold voltages respectively. The transconductance ratio is determined by the geometrical sizes of the transistors.

5.2.3 Simulation Results

Fig. 16 depicts the simulation results of $1w0\widehat{w}1r0$ sequence and Fig. 17 those of the $0w1\widehat{w}0r1$ sequence for our case-study. The conclusions than can be drawn from the two figures are similar to those drawn from Figs. 12 and 13, including the limitations of the DFTs.

6 Programmable DFT Schemes

The proposed DFT schemes in the previous section have major limitations. They define a single stress strength, which is determined at the design stage using the best available pre-fabrication data. Therefore they cannot be tuned during post-silicon test to deal with the unexpected uncertainties (e.g. process variations). As a consequence, they may cause undesirable under-stress or over-stress tests resulting in more test escapes and/or yield loss. To deal with these limitations, the DFTs have to be designed in such a way that they can be calibrated during post-silicon testing; they have to be able to provide multiple stress strength settings. During the post-silicon testing, the appropriate setting is selected based on acceptable or pre-determined targets (quality and reliability versus yield tradeoffs). The programmable versions of the two proposed DFT schemes are introduced next.

It is worth noting that the selection of appropriate programmable DFT settings can be done by characterizing the

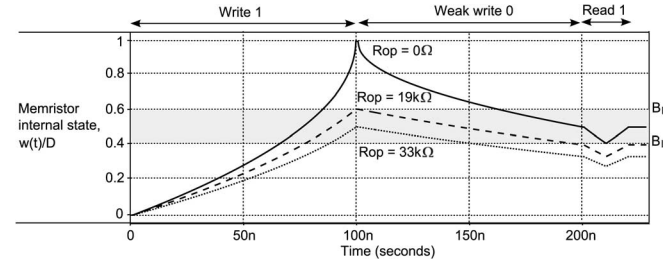

Fig. 17. Simulation results of $0w1\widehat{w}0r1$ with LWV.

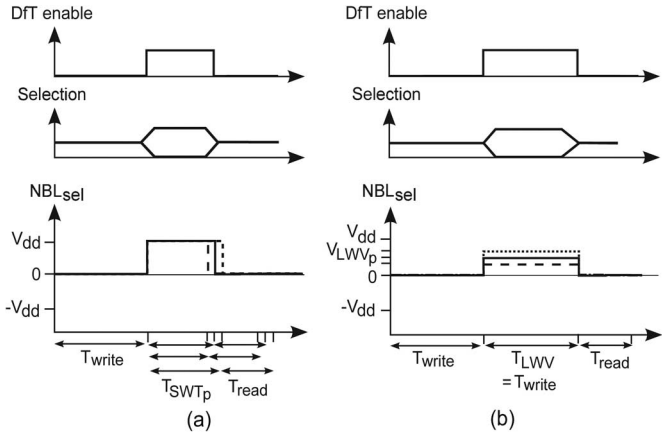

Fig. 18. Control signals for (a) PSWT (b) PLWV schemes.

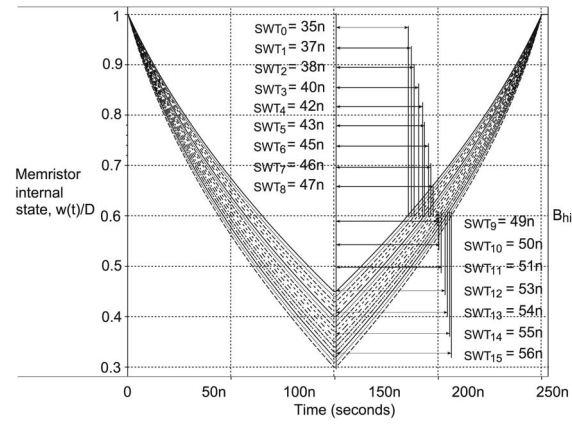

Fig. 19. Determination of required write time periods.

behavior of a population of devices. First, testing is performed on this population using several stress settings. Thereafter, the results are analyzed to select an appropriate stress setting that satisfies a predetermined quality (and reliability) versus yield.

6.1 Programmable Short Write Time PSWT

If the SWT scheme is extended to have multiple programmable access time durations, one (or more) weak write operations can be used to shift the cell to a desired defined state. Fig. 18(a) shows the control signals of the PSWT scheme; a weak write operation that uses the supply voltage V_{dd} (signal NBL_{sel} in the figure) can be applied in the test mode using different time durations T_{SWT_p}; p indicates the selected number of programmable write times and is a power of 2.

To find these p different write times, the design steps for the SWT scheme mentioned in Section 5.1 are used with a slight modification; they consist of the following three steps.

1) Set up the number of programmable write times p. This corresponds to the number of possible faulty (boundary) states that the DFT scheme will shift to defined states. As case study, we consider the memory operations $1w0\widehat{w}1r0$, and define $p = 2^4 = 16$ to have a sufficient precision at an acceptable area overhead cost. In addition, we assume that in the presence of a defect, the w 0 operation of the sequence will put the cell in a state between $B_0 = 0.45$ and $B_{15} = 0.3$ (i.e., around the lower bound of the undefined region). Note that this is just an example and any margin around the bound can be assumed.
2) Search for the appropriate R_{op} values that produce the p different considered states. For our case study with $p = 16$, open defects with various resistance values ($1k\Omega \leq R_{op} \leq 100k\Omega$) are injected within the RRAM cell and $1w0$ sequences are performed.
3) Determine the required 16 write time durations by performing $1w0w1$ sequence in the presence of each of the p (16 in our case) R_{op} values obtained in the second step. Fig. 19 shows the simulation results for our case study. Each $\frac{w(t)}{D}$ curve corresponds to the respective R_{op} value injected (as found in the previous step). Observing the time at which the curves reach $B_{high} = 0.6$ from their lowest point shows that the required T_{SWT_p} durations are between $T_{SWT_0} = 35ns$ and $T_{SWT_{15}} = 56ns$.

PSWT circuit: Fig. 20 shows the block diagram of PSWT circuit; it is an extension of that of Fig. 11 with a decoder (in that case 4 to 16) being able to generate the required number of programmable write times. Depending on the selection signals S_1 to S_4, the decoder will set *Timer* to activate one of the T_{SWT_p} durations. For example, when $S_1S_2S_3S_4 = 0000$, the decoder sets *Timer* to activate the T_{write} for $T_{SWT_0} = 35ns$.

6.2 Programmable Low Write Voltage PLWV

If the LWV scheme is extended to have multiple programmable voltage values, one (or more) weak write operations (resulting in different electrical fields) can be used to shift the cell to a desired defined state. Fig. 18(b) shows the control signals of the PLWV scheme; a weak write operation that uses the normal write time (signal NBL_{sel} in the figure) can be applied in the test mode but using different reduced write voltages V_{LVW_p}; p indicates the selected number of programmable write voltages and is a power of 2.

To find the p different write voltages, a similar methodology as that of PSWT (consisting of three steps) can be applied. However, in the second step and for each R_{op} injected, different V_{LWV_p} were supplied during the simulation. Again we assume for our case study that $p = 16$; the results of step 3 (where the required 16 write voltages should be found by performing $1w0w1$ sequence in the presence of each of the 16 R_{op} values) are shown in Fig. 21; note that depending on the simulated defect, T_{LWV_p} values range from $T_{LWV_0} = 0.53V$ to $T_{LWV_{15}} = 0.84V$.

PLWV circuit: Fig. 22 shows a possible implementation of PLWV circuit; a better and more optimized implementation of a voltage divider circuit can be found in [33]. The implementation of Fig. 22 is based on the circuit of Fig. 15(b). In normal

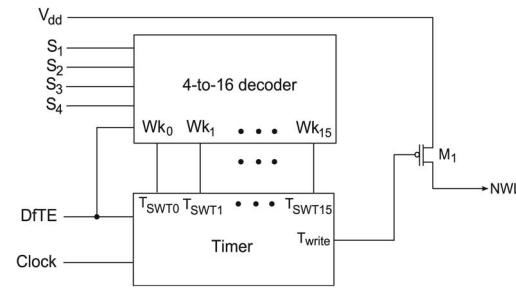

Fig. 20. PSWT circuit.

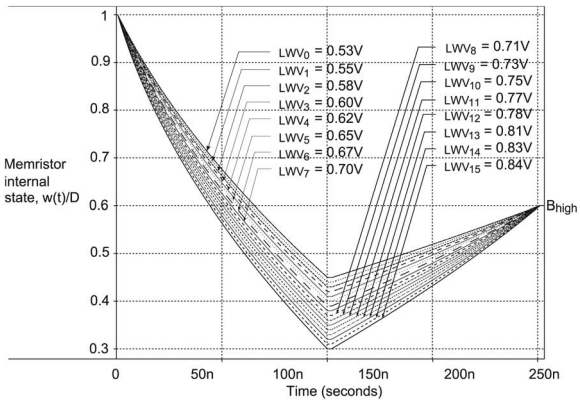

Fig. 21. Determination of required write voltages.

mode (i.e., *DfTE* is low), the circuit generates the normal V_{dd} voltage on the *NBL* output. In test mode (i.e., *DfTE* is high), a voltage divider circuit is formed by one of the PMOS transistors $P0$ to $P15$ and the NMOS transistor $N1$. The selection transistors $T0$ to $T15$ activate a particular voltage circuit divider when switched on. For example, the combination $S_1S_2S_3S_4 = 0000$ selects Wk_0 to be high; this will turn transistor $T0$ on and activate the voltage divider formed by transistors $P0$ and $N1$. The desired voltage transfer is achieved by the following equation [32]:

$$\frac{\beta_{N1}}{\beta_{Pi}} = \left(\frac{V_{dd} - V_{write,i} - |V_{T,Pi}|}{V_{write,i} - |V_{T,N1}|}\right)^2. \quad (6)$$

Here, Pi denotes the PMOS transistor with index i, $V_{T,Pi}$ ($V_{T,N1}$) the threshold voltage of transistor Pi ($N1$), $V_{write,i}$ is the desired write voltage when Wk_i is selected. The values of $V_{write,i}$ are visible in Fig. 21.

6.3 Comparison of DfT Schemes

Table 3 provides a comparison (in a relative manner) of the proposed DFT schemes in this work; the considered aspects are fault detection capability, test time and implementation. Obviously the PSWT and PLWV schemes have a high fault detection capability due to their programmability attributes.

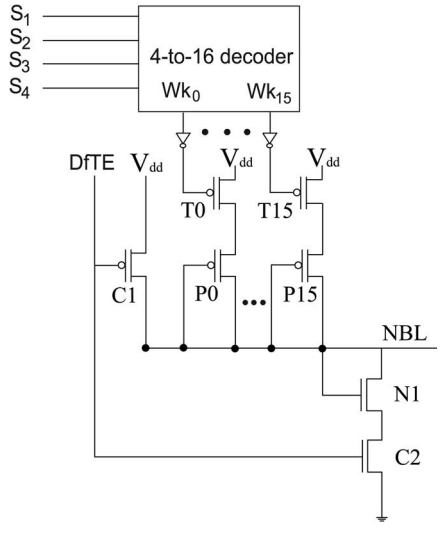

Fig. 22. PLWV circuit.

TABLE 3
Comparison between the Proposed DfT Schemes

DfT scheme	Fault detection capability	Test time	Implementation
SWT	Low	Short	Easy
LWV	Low	Medium	Hard
PSWT	High	Long	Easy
PLWV	High	Very long	Hard

However, they require longer test time as compared with non-programmable solutions SWT and LWV. Moreover, it can be easy seen that the implementation of SWT/PSWT is much easier than that of LWV/PLWV; the first has mainly a digital behavior that can be realized with a counter, while the second has an analog nature and realizing accurate writing voltages could be difficult due to e.g. process variations.

It is worth noting that all the DFTs are scalable; the larger the memory, the smaller the area overhead. All these DFTs require assigning a similar circuit to each bit-line. Hence, the area overhead increases linearly with the number of bit lines, however it is independent of the number of word-lines. Given the fact

7 Conclusion

In this paper, a fault analysis for resistive open defects based on electrical simulation and test approach for RRAMs have been presented. It is shown that in addition to traditional memory fault models, other unique faults may occur; the detection of the latter cannot be easily guaranteed with the popular march tests. Therefore, (programmable) DFT schemes are proposed. Simulation results shows that these DFTs improve the overall defect/fault coverage.

To the best knowledge of the authors, this is one of the first papers addressing testing of RRAMs. Many questions are still opens; examples are: (a) the impact of CMOS logic delay (e.g. in address decoders) on memory states, (b) coupling effects, (c) defects in multi-level memory cells, (d) efficient DFTs, etc.

References

[1] International Technology Roadmap for Semiconductors. (2011). *Emerging Research Devices (ERD) 2011 Edition* [Online]. Available: http://www.itrs.net.
[2] G. W. Burr *et al.*, "Overview of candidate device technologies for storage-class memory," *IBM J. Res. Develop.* vol. 52, no. 4/5, pp. 449–464, 2008.
[3] J. J. Yang *et al.*, "Memristive switching mechanism for metal/oxide/metal nanodevices," *J. Nature Nanotechnol.*, vol. 3, pp. 429–433, Jun. 2008.
[4] S. F. Karg *et al.*, "Transition-metal-oxide-based resistance-change memories," *IBM J. Res. Develop.*, vol. 52, no. 4/5, pp. 481–492, 2008.
[5] D. B. Strukov *et al.*, "Prospects for terabit-scale nanoelectronic memories," *Nanotechnology*, vol. 16, no. 1, pp. 137–148, Jan. 2005.
[6] A. Dehon, "Nanowire-based programmable architectures," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 1, no. 2, pp. 109–162, Jul. 2005.
[7] G. S. Snider *et al.*, "Nano/CMOS architectures using a field programmable nanowire interconnect," *Nanotechnology*, vol. 18, no. 3, pp. 1–11, Jan. 2007.
[8] P. Clarke. (2013). *Updated: HP, Hynix Plan to Launch Memristor Memory in 2013*, EE Times [Online]. Available: http://www.eetimes.com/electronics-news/4229171/HP-Hynix-to-launch-memristor-memory-2013.

[9] L. O. Chua, "Memristor—The missing circuit element," *IEEE Trans. Circuit Theory*, vol. 18, no. 5, pp. 507–519, Sep. 1971.
[10] D. B. Strukov *et al.*, "The missing Memristor found," *Nature Lett.*, vol. 453, pp. 80–83, May 2008.
[11] C. Xu *et al.*, "Design implications of Memristor-based RRAM cross-point structures," in *Proc. Design Autom. Test Eur.*, 2011, pp. 1–6.
[12] Y. Ho *et al.*, "Dynamical properties and design analysis for nonvolatile Memristor memories," *IEEE Trans. Circuits Syst.-I: Regular Papers*, vol. 58, no. 4, pp. 724–736, Apr. 2011.
[13] O. Kavehei *et al.*, "The fourth element: Characteristics, modelling and electromagnetic theory of the Memristor," in *Proc. Royal Soc. Math. Phys. Eng. Sci.*, 2010, pp. 1–28.
[14] F. Sun *et al.*, "Defect and transient fault-tolerant system design for hybrid CMOS/Nanodevice digital memories," *IEEE Trans. Nanotechnol.*, vol. 6, no. 3, pp. 341–351, May 2007.
[15] N. Z. Haron *et al.*, "Redundant residue number system code for fault-tolerant hybrid memories," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 7, no. 1, pp. 1–19, 2011, article 4.
[16] O. Ginez, J.-M. Portal, and C. Muller, "Design and test challenges in resistive switching RAM (ReRAM): An electrical model for defect injections," in *Proc. Eur. Test Symp.*, 2009, pp. 61–66.
[17] S. Kannan, J. Rajendran, R. Karri and O. Sinanoglu, "Sneak-path testing of Memristor-based memories," in *Proc. 26th Int. Conf. VLSI Design*, 2013, pp. 386–391.
[18] N. Z. Haron *et al.*, "On defect oriented testing for hybrid CMOS/Memristor memory," in *Proc. Asian Test Symp.*, 2011, pp. 353–358.
[19] N. Z. Haron *et al.*, "DfT schemes for resistive open defects in RRAMs," in *Proc. Design Autom. Test Eur.*, 2012, pp. 799–804.
[20] S. Hamdioui *et al.*, "An experimental analysis of spot defects in SRAMs: Realistic fault models and tests," in *Proc. Asian Test Symp.*, 2000, pp. 131–138.
[21] D. Tu *et al.*, "Three-dimensional CMOL: Three-dimensional integration of CMOS/Nanomaterial hybrid digital circuits," *Micro Nano Lett.*, vol. 2, no. 2, pp. 40–45, 2007.
[22] J. W. Son, "Prediction of defects in nano-imprint lithography using FEM simulation," *J. Key Eng. Mater.*, vol. 345–346, pp. 665–668, 2007.
[23] S. Hamdioui *et al.*, "Opens and delay faults in CMOS RAM address decoder," *IEEE Trans. Comput.*, vol. 55, no. 12, pp. 1630–1639, Dec. 2006.
[24] L. Dilillo *et al.*, "Comparison of open and resistive-open defect test conditions in SRAM address decoders," in *Proc. IEEE Asian Test Symp.*, 2003, pp. 250–255.
[25] M. Klaus and A. J. van de Goor, "Tests for resistive, and capacitive defects in address decoders," in *Proc. 10th Asian Test Symp. (ATS'01)*, 2001, pp. 31–36.
[26] B. Kruseman *et al.*, "Systematic defects in deep sub-micron technologies," in *Proc. Int. Test Conf.*, 2004, pp. 290–299.
[27] S. Hamdioui *et al.*, "Memory test experiment: Industrial results and data," *IEE Proc. Comput. Digit. Techn.*, vol. 153, no. 1, pp. 1–8, 2006.
[28] L. Dilillo *et al.*, "Resistive-open defects in embedded-SRAM core cells: Analysis and march test solution," in *Proc. Asian Test Symp.*, 2004, pp. 266–271.
[29] S. Hamdioui *et al.*, "Memory fault modeling trends: A case study," *J. Electron. Testing*, vol. 20, no. 3, pp. 245–255, 2006.
[30] A. Meixner *et al.*, "Weak write test mode: An SRAM cell stability design for test technique," in *Proc. Int. Test Conf.*, 1996, pp. 309–318.
[31] J. Rajendran *et al.*, "An approach to tolerate process related variations in Memristor-based applications," in *Proc. IEEE Symp. VLSI Design*, 2011, pp. 18–23.
[32] J. P. Uyemura, *Circuit Design for CMOS VLSI*, Dordrecht, the Netherlands: Kluwer. Second Printing 1993. Design, pp. 18–23, 2011.
[33] E. B. Selvin *et al.*, "Programmable weak write test," USA patent US 2003/0210593 A1, 2003.
[34] R. Dekker, F. Beenker, and L. Thijssen, "Realistic built-in self-test for static RAMs," *Design Test Comput.*, vol. 6, no. 1, pp. 26–34, 1989.
[35] T. J. Powell *et al.*, "BIST for deep submicron ASIC memories with high performance application," *Proc. IEEE Int. Test Conf.*, 2003, pp. 386–392.
[36] D. Appello *et al.*, "Exploiting programmable BIST for the diagnosis of embedded memory cores," in *Proc. Int. Test Conf.*, 2003, p. 379.
[37] X. Du, N. Mukherjee, W.-T. Cheng, and S. M. Reddy, "A field-programmable memory BIST architecture supporting algorithms and multiple nested loops," in *Proc. Asian Test Symp.*, 2006, pp. 287–292, paper 45.3.
[38] S. Hamdioui, Z. Al-ars, G. N. Gaydadjiev, A. J. Van De Goor, "Generic march element based memory built-in-self-test," US patent 20,130,086,440, 2013.
[39] S. Hamdioui, "Testing embedded memories: A survey," *Math. Eng. Methods Comput. Sci.*, vol. 7721, pp. 32–42, 2013.
[40] C. J. Lin *et al.*, "45 nm low power CMOS logic compatible embedded STT MRAM utilizing a reverse-connection 1T/1MTJ Cell," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, 2009, pp. 1–4.
[41] N. Z. Haron, "Testability and fault tolerance for emerging nanoelectronic memories," Ph.D. dissertation, Delft Univ. of Technology, Delft, the Netherlands, 2013.

Said Hamdioui received the MSEE and PhD degrees (both with honors) from Delft University of Technology (TUDelft), The Netherlands. He is currently co-leading dependable-nano computing research activities within the Computer Engineering Laboratory of TUDelft. Prior to joining TUDelft, he worked for Microprocessor Products Group at Intel Corporation, USA, for IP and Yield Group at Philips Semiconductors R&D, France, and for DSP design group at Philips/NXP Semiconductors, The Netherlands. His research interests include test technology and design-for-test, 3-D stacked ICs, hardware security, reliability, dependability, nano-computing, memristor techology, etc. He published one book and co-authored over 120 conference and journal papers. He has consulted for many semiconductor companies in the area of memory testing. He delivered dozens of keynote speeches, distinguished lectures, and invited presentations and tutorials at major international forums/conferences and leading semiconductor companies. He serves on the editorial board of the *Journal of Electronic Testing: Theory and Applications (JETTA)*, on that of *IEEE Design and Test*.

Mottaqiallah Taouil received the MSc degree with honors in computer engineering from the Delft University of Technology (TUDelft), The Netherlands. He is currently pursuing a PhD at the same university in the Dependable Nano-Computing Group. His research interests include reconfigurable computing, embedded systems, VLSI design and test, built-in-self-test, 3-D stacked ICs, 3-D architectures, (3-D) design for testability, (3-D) yield analysis, and 3-D memory test structures.

Nor Zaidi Haron received the PhD degree in computer engineering from the Delft University of Technology, The Netherlands. Currently, he is a lecturer at Universiti Teknikal Malaysia Melaka, Malaysia. His research interests include memory fault modeling and testing, design-for-testability, fault tolerance, and nano-architectures. He published several papers in the field of electrical defect simulation, fault modeling and test, as well as fault-tolerant techniques for emerging memories.