Detecting Random Read Faults to Reduce Test Escapes in FinFET SRAMs

Cardoso Medeiros, G.; Fieback, M.; Gebregiorgis, A.; Taouil, M.; Bolzani Poehls, L.; Hamdioui, S.

DOI
10.1109/ETS50041.2021.9465441

Publication date
2021

Document Version
Accepted author manuscript

Published in
2021 IEEE European Test Symposium (ETS)

Detecting Random Read Faults to Reduce Test Escapes in FinFET SRAMs

G. Cardoso Medeiros[1], M. Fieback[1], A. Gebregiorgis[1], M. Taouil[1], L. Bolzani Poehls[2,3], S. Hamdioui[1]
[1]Computer Engineering Laboratory, Delft University of Technology, The Netherlands
[2]EASE, PUCRS, Brazil, [3]IDS, RWTH Aachen University, Germany
Email: {G.CardosoMedeiros, S.Hamdioui}@tudelft.nl

***Abstract*—Manufacturing defects in FinFET SRAMs can cause hard-to-detect faults such as Random Read Faults (RRFs). Detection of RRFs is not trivial, as they may not lead to incorrect outputs. Undetected RRFs become test escapes, which might lead to no-trouble-found devices and early in-field failures. Therefore, the detection of RRFs is of utmost importance. This paper proposes test solutions to detect RRFs and reduce test escapes. To achieve this, we first statistically analyze the failure rate due to RRFs, followed by an experimental study of stress conditions' (SCs) impact on detecting RRFs, such as test algorithms, supply voltage, and temperature. Based on the results, we propose a new Design-For-Testability (DFT) scheme for FinFET SRAMs to detect such faults using SCs that improve the detection rate of RRFs. This scheme introduces a negligible area and test time overhead while significantly enhancing RRF detection. Hence, using the proposed DFT leads to reduced test escapes and, consequently, higher-quality FinFET SRAMs.**

***Index Terms*—Memory Testing, Test Escapes, SRAM, FinFET, DFT**

I. Introduction

FinFET technology has been used to continue the scaling down of technological nodes according to Moore's Law. The complex structure of FinFETs provides improved short-channel behaviors while overcoming the planar CMOS technology's sub-threshold leakage [1]. Nevertheless, the process to manufacture such an intricate structure may also introduce defects in the transistor's features, such as opens in fins [2], pinholes in the oxide [3], and opens in the 3D gate [4]. These defects may affect FinFET SRAMs and cause Hard-to-Detect faults [5] such as *Random Read Faults* (RRFs), i.e., a significantly reduced bit line swing [6]. In the presence of an RRF, it is impossible to guarantee that the *Sense Amplifier* (SA) will output the cell's content – some RRFs will lead to incorrect outputs, while others will not. Tests that rely on fault observation (e.g., March tests) can only detect RRFs that cause incorrect functional behavior. However, it is still essential to detect RRFs that do not lead to incorrect read outputs as they become test escapes, which are a known cause of no-trouble-found devices [7] and in-field reliability issues [8]. Therefore, new high-quality methodologies to detect RRFs are essential to reduce test escapes and improve the quality and reliability of FinFET memories [9].

Unlike deterministic faults, i.e., faults that always lead to incorrect functionality, detecting RRFs is challenging. March algorithms such as SS, AB, and FFDD [10–12] can easily detect deterministic faults; however, they can detect only the portion of RRFs that will lead to incorrect behavior. Therefore, more complex test solutions (e.g., more extreme stressing conditions and *Design-for-Testability* (DFT) circuits) must be used when targeting RRFs. Examples of such test solutions are schemes that change how read operations are executed [13–15], and schemes that monitor memory parameters [16–18]. However, both approaches have limitations; the former may over-test the memory due to inappropriate stresses, thus leading to yield loss, while the latter may be negatively impacted by *process variation* (PV) effects, thus leading to test escapes. Hence, a DFT that reduces test escapes due to RRFs while minimizing yield loss is still missing.

This paper addresses these problems by proposing test solutions to reduce RRF test escapes in FinFET SRAMs. First, we statistically assess the expected incorrect output rate due to RRFs. We show that the failure rate can be modeled by the distribution of random variations (e.g., PV, noise) during a read operation. We then conduct simulation experiments with a FinFET SRAM under various *stressing conditions* (SC) to identify which lead to the highest RRF detection. Finally, we propose a new DFT scheme that uses the identified SCs to maximize RRF detection. Furthermore, calibration schemes can be easily integrated into this scheme to overcome PV effects and avoid over-testing. Compared to the state of the art, this DFT efficiently improves the detection of RRFs with minimal area overhead and yield loss by using appropriate SCs only. The main contributions of this paper are as follows:

- A model to estimate RRF's failure rate.
- An extensive experimental analysis of the SCs' impact on RRF detection.
- A new DFT scheme for RRFs that uses ideal SCs to maximize fault detection and minimize yield loss.
- Validation and evaluation of the proposed DFT scheme.

This paper is organized as follows. Section II proposes a model to estimate RRF failure rate. Section III explores the RRF's detection dependency on SCs. Section IV introduces a DFT to improve the detection of RRFs. Section V presents a brief discussion on the DFT and its limitations. Finally, Section VI concludes the paper.

This work is supported by European Union's Horizon 2020 Research and Innovation through RESCUE project under grant 722325.

II. RRFs and their Impact

A. Causes & Definition

SRAMs are volatile memories consisting of a cell array and peripheral circuitry. The cell array consists of rows and columns of cells connected horizontally by *word lines* (WLs) and vertically by pairs of *bitlines* (BL and $\overline{BL}$). The peripheral components (decoders, *sense amplifiers* (SAs), output latches, write drivers, prechargers) provide write and read capabilities to the SRAM. SRAMs can be designed using FinFET devices, which are three-dimensional, multi-gate transistors [1]. During their manufacturing process, small particles and lithography inconsistencies can result in defects in their structure [19], e.g., partial opens, misplaced connections, and damaged fin structure [2–4]. These defects may impact a cell's ability to discharge its BLs during a read operation and hence impact the cell's BL swing, which is the voltage difference between a BL pair when the SA is enabled, i.e., BL swing $= |BL - \overline{BL}|$. A *Random Read Fault* (RRF) occurs when this swing is too small for the SA to guarantee a correct output, resulting in a random output, i.e., either '0' or '1' [6].

The Fault Primitive notation [20] describes RRFs as faults in which the read output element (*R*) is expressed as **?**. For example, $\langle 1r1/1/? \rangle$ denotes a read '1' operation that does not impact the cell's content but returns a random value. When the output matches the cell's content, the RRF is not detected. Because RRF detection is not guaranteed by performing a sequence of write and read operations, they are classified as Hard-to-Detect faults [15]. If not detected, RRFs become test escapes, a known cause for no-trouble-found components reliability issues [7, 8]. Therefore, detection of RRFs is critical to assure high-quality FinFET SRAMs.

B. Expected Failure Rate Analysis

It is statistically expected that some RRFs will lead to correct outputs and thus test escapes. The remaining will lead to a failure, i.e., the SA will not output the expected logic value, thereby enabling RRF detection. Two primary metrics determine the read operation outcome: the SA offset and BL swing. The SA offset is the voltage shift that creates a mismatch in the SA's cross-coupled inverter pairs' strengths. In a PV-free SA, this offset is 0 V. However, PV will cause one pair to be stronger than the other, offsetting the SA towards the logic value '1' or '0'. The SA outputs the correct value only if its offset is smaller than the BL swing; we name this voltage difference as ΔV, where ΔV$=$ BL swing $-$ SA offset voltage. Consequently, if $\Delta V < 0$, the SA outputs the incorrect value. For example, an SA with 3 mV offset will correctly output the cell's content if the BL swing is greater than 3 mV. We use the variation of these two parameters that form ΔV(i.e., SA offset and BL swing) to express the failure rate.

We first consider SA's offset variation. In this work, we use a traditional 6T SA design; an in-depth discussion of the SA's implementation will be given in Section IV-B. We performed *Monte Carlo* (MC) analysis (20,000 simulations) on the SA, and obtained a (rounded) mean (μ) offset of 0 V and a standard deviation (σ) of 6 mV. The probability that the SA's offset variation is within or outside a given range can be calculated by the cumulative distribution function [21]; the probability that this variation is outside an $n\sigma$ range (i.e., the SA's offset is greater than $|n\sigma|$, where n = 1, 2, 3...) is given by $1 - erf(n\sigma/\sqrt{2})$, where *erf* is the error function

$$erf\,(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt.$$

We then relate $n\sigma$ to the BL swing, i.e., SA's input. The probability that the SA offset is greater than σ, 2σ, etc, is by default the probability that the SA offset is greater than 6 mV, 12 mV, etc. This failure rate function is plotted in Fig. 1; it represents the probability of an incorrect output value (i.e., failure) for a given BL swing input. Likewise, it also shows the likelihood of no failure, and thus test escapes.

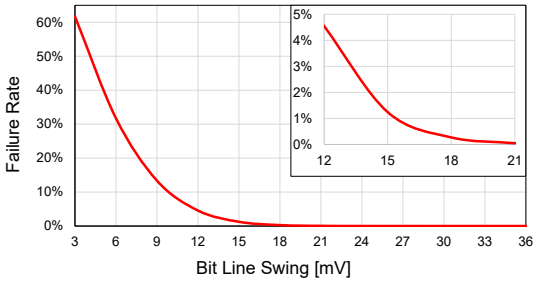


Figure 1. Expected failure rate for a given BL swing.

A second failure rate can be estimated based on BL swing variation. Consider a defect that bridges a column's BL pair; every read operation on this column will be impaired. If the impact is significant, this defect will cause RRFs. Nonetheless, the BL swing among this column's cells will slightly vary from one to another due to PV; some cells will lead to correct outputs, while others will not. Considering that the SA voltage offset is fixed after manufacturing, the probability p of incorrect output becomes a function based on the BL swing variation from one cell to another. In details, p is the probability that the BL swing variation will counter the voltage difference between mean BL swing in the column and the SA's offset, i.e., $p = Pr(\text{BL swing variation} < -\Delta V)$. Furthermore, we can use p to estimate the failure rate after a sequence of read operations in the same column but different cells. As each cell has a different BL swing, there is a possibility that after n read operations in the column, one of these variations will annul ΔV, leading to an incorrect output.

We illustrate this failure rate with the following example. Consider that a defect bridging both BLs reduces the column's mean BL swing from its nominal value to 15 mV. Furthermore, consider that, after manufacturing, the SA is offsetted 5 mV due to PV. Accordingly, the ΔVin this column is $\Delta V = 15 - 5 = 10$ mV. Once ΔV is defined, we measure the standard deviation of the BL swing variation among cells. After MC analysis of 20,000 read operations, a BL swing variation of $\sigma = 4.913$ mV was observed. We use this variation's cumulative distribution function to find the probability of BL swing variation annulling ΔV. For

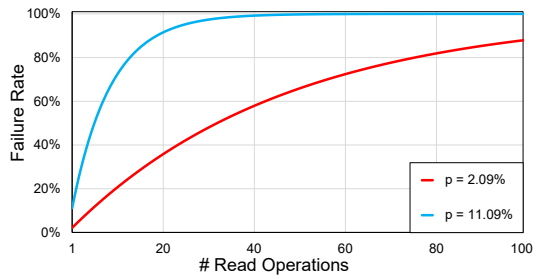


Figure 2. Expected failure rate after n read operations for two scenarios.

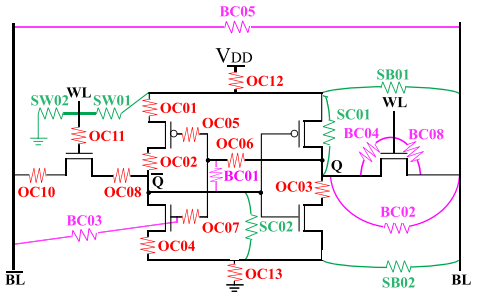


Figure 3. A 6T cell and the injected resistive defects: opens, shorts, bridges.

ΔV= 10 mV, the probability of incorrect output is $p = Pr(\text{BL swing variation} < -10\text{ mV}) = 2.09\%$. Thus, there is a 2.09% chance that the BL swing will be smaller than the SA's offset, and consequently, trigger an incorrect output. We can then use equation $1–(1-p)^n$ [21] to estimate the failure rate after performing n read operations based on the probability p of a single read operation; after 25 operations, the failure rate is around 40%, and 87% after 100 operations, as shown in Fig. 2. Additionally, a second failure rate for an exemplary case where $\Delta V = 6$ mV is also plotted. As ΔV is smaller than the first case, the probability that the BL swing will annul ΔV, i.e., $Pr(\text{BL swing variation} < -6\text{ mV})$, is higher than before (namely 11.09%). Accordingly, the number of read operations, and therefore effort, required to guarantee an incorrect output (i.e., 100% failure rate) and detect the RRF is much smaller.

Finally, the same model can also estimate the failure after n read operations on the same cell. In this case, variations in the BL swing must originate from dynamic effects that will change the cell's BL swing from one read operation to another, such as white, flicker, and temperature noise. Based on these analyses, we conclude the following:

- The probability of an incorrect output increases as more read operations are performed, which justifies the industry's hammering techniques to test memories.
- Smaller BL swings lead to higher failure rates. Hence, an efficient way to improve RRF detection is by reducing the BL swing, which can be achieved by applying specific stresses or using dedicated *Design-for-Testability* (DFT) circuits.
- Reducing the BL swing reduces test effort and time (blue curve in Fig. 2). A smaller BL swing results in a smaller ΔV and a bigger p, leading to fewer read operations necessary to achieve a higher failure rate.

III. RRF Detection Dependency on Stress Conditions

This section estimates the impact of stress conditions *SCs* on the detection of RRFs. We first introduce a classification of SCs, followed by a description of the simulation setup, and finally, the detection results.

A. Classification of Stress Conditions

SCs are applied during testing to boost fault sensitization and detection. They can be categorized into two classes: algorithm-related and environment-related [22]. Algorithm-related stresses specify the algorithm applied to the memory cells; they include all SCs derived from writing and reading operations. Examples of algorithm-related SCs are as follows:

- **Base Test (BT)**: A BT is a sequence of operations (reads and writes) applied to a memory cell.
- **Address Order (AO)**: The AO is the address sequence generated by an addressing method (e.g., increasing (⇑), decreasing (⇓), binary, hamming distance, H2/H3/HN1 [23]) that defines how the algorithm accesses addresses.
- **Address Direction (AD)**: The AD indicates how the AO is applied considering rows and columns. The most common ADs are fast-row and fast-column [22].
- **Data Background (DB)**: DB is the pattern of ones and zeros as seen in the memory array. The most known DBs are Solid, Checkerboard, and Row/Column Stripe [22].

Environment-related SCs use additional stress sources to change the operating conditions of the memory. These include [22], but are not limited to:

- **Voltage Stress** in the entire circuit (e.g., changing the supply voltage) or a peripheral (e.g., write driver, SA).
- **Timing Stress** in the entire circuit (e.g., changing frequency) or specific components (e.g., write driver, SA) to change memory operations' timing.
- **Temperature Stress** by either increasing or reducing temperature from its nominal value.

B. Simulation Setup

Memory model: the netlist is described using the predictive technology model (PTM) 14 nm FinFET SPICE library [24]. The array comprises 128 rows and 64 columns; each column has a write driver, SA, and prechargers. Capacitive loads are applied to BLs and word lines to emulate a 1 kB memory. The memory operates on a nominal clock frequency of 2 GHz and contains a timing circuit to generate control signals.

Injected defects: Twenty-eight single-cell resistive defects have been injected in the cell, as shown in Fig. 3. They are either *Resistive-Open* (RO), *Resistive-Short* (RS), or *Resistive-Bridge* (RB) defects [25].

Experiments: Each defect was swept with increasing sizes (i.e., resistances) to identify the size range in which RRFs are triggered. Each simulation scenario (i.e., using stress conditions X and injecting defect Y of size Z) was simulated 100 times using MC simulations; from these, a mean BL

swing and a detection rate are calculated. For example, the scenario of using fast-row and checkerboard SC and injecting OC03 of size 20 kΩ was simulated 100 times applying PV effects. These effects are modeled using Pelgrom's model [26] and simulated using a voltage source on the transistor's gate contact of transistors. Measure commands are used to estimate the BL swing and check the cell's content. An RRF occurs if the BL swing is reduced and the cell's content has not flipped during the read operation. Contrarily, a Read Destructive fault occurs when the cell's content is destroyed; even if the read operation outputs a random value, the fault is detected by performing a second read on this same cell.

C. Detection Results for Algorithm-Related SCs

We focus on RRF detection only; the SCs discussed below may impact other types of faults differently. Experiments have shown that three algorithm-related SCs had little impact on RRF detection: base test, address order, and address direction. BT comprises only read operations; write operations are only used to initialize the cells. AO stress did not impact RRF detection; linear addressing methods ⇑ or ⇓ had the same outcome. Finally, fast row and column AD led to a marginal detection rate gain – less than 1% improvement. Nonetheless, DBs proved to impact RRF detection significantly. Hence, we explore two DB stresses: solid and checkerboard. In the solid DB, all cells in the array store the same logic value, e.g., 0000.../0000.../. In the checkerboard DB, adjacent cells have opposite logic values, e.g., 0101.../1010../.

Thirteen defects sensitized RRFs: OC03, OC11, OC08, OC11, OC13, SC01, SC02, SB02, SW02, BC01, BC03, BC05, and BC08. We focus on OC03, OC11, and BC05 as they have defect size ranges in which only RRFs are sensitized, i.e., a severely reduced BL swing and no impact on the cell's content. Thus, the detection of these defects is only achieved through RRF detection. Fig. 4 depicts these defects' detection rates. BC05's detection rates after 2 and 10 read operations (ops.) are also plotted; it is expected that its detection rate increases as more read operations are performed in different cells from the same column. Furthermore, the figure also shows the expected failure rate presented in Section II-B. We can see that solid DB does not detect RRFs, i.e., the read output always matched what was expected; this is due to the output latch and its own DB. Because the BL swing is too small to influence the SA, the output latch influences the SA to remain in its current logic state. As the expected read output never changes, RRFs will not lead to incorrect read outputs, becoming test escapes. Therefore, to improve test algorithms' (e.g., March tests) RRF detection, they must include checkerboard DB stress.

As expected, the detection rate of BC05 improved after consecutive read operations on the defective column. Furthermore, we can notice that the detection rate when using checkerboard DB matches the expected failure rate behavior, but with a shifted starting point depending on the defect. Thus, additional parameters, variations, and conditions must still be included in the expected failure rate to estimate incorrect outputs due to RRF accurately.

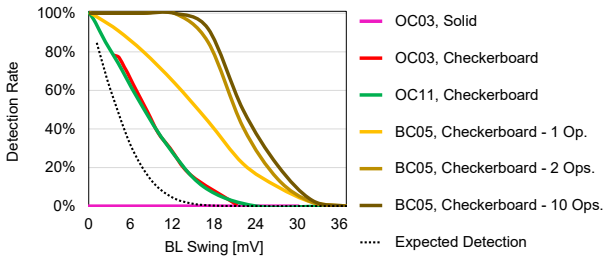


Figure 4. Detection rate of defects OC03, OC11, and BC05.

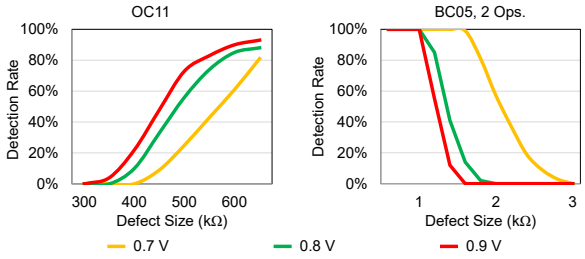


Figure 5. Detection rate of OC11 and BC05 for different supply voltages.

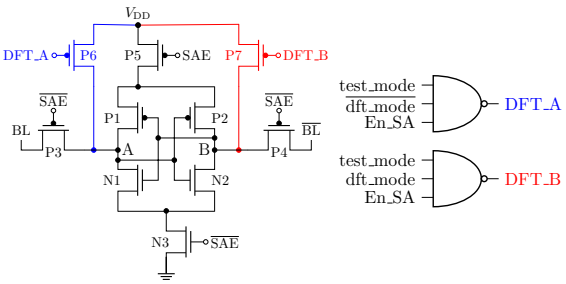
D. Detection Results for Environment-Related SCs

Environment-related SCs set the memory's operating conditions. Reducing frequency can boost RRF detection; a higher frequency leads to a smaller WL enable period, reducing the BL swing and increasing incorrect outputs. However, our memory uses self-timing circuits to limit WL enable period. Therefore, in our experiments, altering frequency did not affect RRF detection. Nevertheless, temperature and voltage improved RRF detection. While both scenarios had similar results, the detection gain was more significant for supply voltage than temperature; we focus on the former's results.

We have observed that supply voltage does not alter the relation between BL swing and detection rate, i.e., a given BL swing led to a similar detection rate. However, altering supply voltage changes the BL swing for a given defect size. Thus, there were significant changes in the relation between defect size and detection rate. Fig. 5 shows OC11's and BC05's (2 ops.) detection rates for varied defect sizes and supply voltages; defect OC03 is omitted as it experiences the same impact as BC05. For OC11, increasing the supply voltage reduces the BL swing. Since reducing the BL swing is one of the best approaches to improve RRF detection, increasing the supply voltage boosts detection of RRFs caused by OC11. On the other hand, a reduced supply voltage leads to a smaller BL swing when considering defect BC05. Hence, the supply's voltage impact on the detection ultimately depends if changing the supply voltage will reduce or increase the BL swing.

Based on the experiments using different types of SCs, we conclude the following:

- SCs impacts BL swing and SA's amplification differently and must be considered when estimating RRF failure rate.
- Checkerboard data background is a must to detect RRFs.
- Performing more read operations boosts RRF detection.
- Testing in different operating conditions is a must to obtain the highest RRF detection rate possible.



(a) SA Design with the DFT. (b) DFT Control.

Figure 6. SA & DFT Organization.

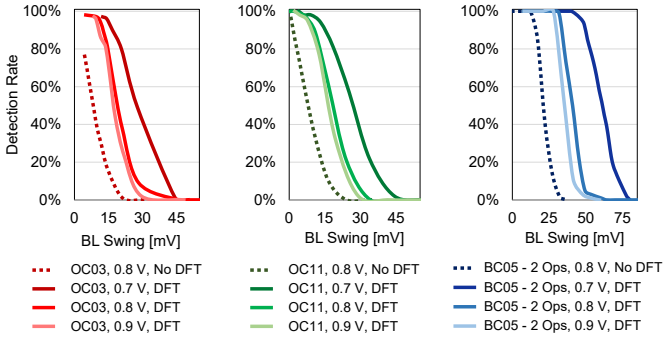


Figure 7. DFT Detection rate considering BL swing.

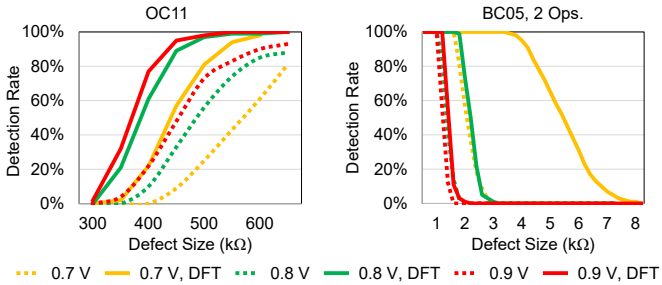


Figure 8. DFT Detection rate considering defect size.

IV. Proposed DFT to Detect RRFs

A. Concept

The DFT technique proposed in this work targets RRFs by creating a mismatch in the SA. Additional PMOS transistors are inserted into the SA and activated during test-mode read operations to skew the SA's amplification towards the opposite value that is being read. Fault-free read operations will easily overcome the DFT's mismatch, and the SA will output the correct cell content. On the other hand, when RRFs occur, the SA will fail to overcome the DFT's mismatch and output the opposite expected logic value.

B. Implementation

The DFT is implemented in two parts, as shown in Fig. 6. The first part consists of 1-fin PMOS transistors in the SA's pull-up network. Only one transistor is enabled at a time to create a mismatch towards the opposite logic value expected from the read operation. For example, P6 is activated during a read '0' operation and charges node *A* since it is expected that BL will discharge this same node. During amplification, the SA becomes less likely to amplify based on the BL swing and more likely based on the DFT configuration, i.e., opposite of the expected value. The RRF is then detected if the SA indeed amplifies to and outputs the opposite expected value. The second part is two 3-input NAND gates that generate the controlling signals based on the enable SA signal (*En_SA*) that generates the *SAE* signal; hence, both DFT and SA have the same timing scheme. Furthermore, two signals configure the DFT. The *test_mode* signal enables the DFT, while *dft_mode* indicates the current read operation, i.e. *dft_mode* = '0' during a read '0' operation and *dft_mode* = '1' during a read '1'.

C. Detection Results

Fig. 7 shows the DFT's RRF detection for a given BL swing under different supply voltages; algorithm-related SCs have been set to fast-row AD and checkerboard DB. Without the DFT, supply voltage does not change the relationship between the detection rate and BL swing; hence, only the detection rate for 0.8 V is shown. However, a considerable increase in this relation is observed when using our DFT, leading to higher RRFs detection for a given BL swing; the most significant gain is obtained with a supply voltage of 0.7 V. Nevertheless, we also analyze the DFT's detection based on defect size, as shown in Fig. 8; this analysis proves necessary as supply voltage's impact on BL swing changes from defect to defect. While we see that BC05's RRF detection rate is indeed the highest when reducing supply voltage, the same is not valid for OC11. Although there is a significant detection gain when using the DFT at 0.7 V supply voltage, the highest coverage is still achieved at 0.9 V. Therefore, using the DFT with reduced and increased supply voltage is the best approach.

V. Discussion

Applicability: there are situations in which the costs of increasing RRF detection are not justified. In non-critical applications in which some defects are allowed, it is cheaper to adapt SCs to increase RRF detection rather than modifying sensitive areas such as the SA. Notwithstanding, for critical applications such as automotive and aerospace, 0 defective parts per million becomes a must; in these scenarios, the costs of modifying circuits to include DFTs are acceptable. Therefore, the DFT's applicability is a trade-off between fault coverage and test cost. Furthermore, the DFT can also be used during the memory's prototyping, characterization, and validation to identify the occurrence of RRFs and obtain knowledge of the memory's yield before its mass production.

Applicability to scaled FinFET memories: to the best of our knowledge, there are no public works investigating RRFs in further-scaled FinFET memories (e.g., 7 and 5 nm). However, RRFs will likely be (at least) as relevant in further-scaled memories as in 14 nm memories. With the scaling down of supply voltage and increased parasitics, even slight environmental noise will likely lead to random faulty behaviors. Thus, our DFT is applicable to improve the detection of RRFs in further-scaled FinFET memories.

Applicability to emerging memories: RRFs also impact emerging memory technologies such as RRAM and STT-MRAM. However, a DFT for these memories may be less critical as algorithms can already provide sufficient RRF detection due to higher cycle-to-cycle variability [27]. Nevertheless, such DFT can still improve the detection of RRFs regardless of memory type; the only requirement for this DFT is a memory that uses an SA with an amplification phase.

Calibration Capabilities: the DFT offers many ways to calibrate its detection rate. Run-time calibration can be achieved using different SCs. Post-silicon calibration is possible by changing the DFT timing to adjust the mismatch and hence detection; this can be implemented with a dynamic delay selector on the control gates, like the calibration in [15].

Overhead: the overhead introduced by the proposed DFT scheme are negligible. The DFT's control comprises only two 3-input NAND gates, which is negligible compared to other peripheral circuitry. Furthermore, the two 1-fin PMOS transistors introduced into each SA represent an area overhead of only 2.5% of the SA; this overhead is even more negligible considering the cell area covered by a single SA. The DFT does not introduce any test time overhead as it can be enabled throughout all read operations. Hence, it is possible to combine the algorithm's fault coverage with the DFT's RRF coverage with the same effort and zero time overhead.

Comparison with state of the art: compared to traditional March algorithms, our DFT significantly improves the detection rate of RRFs as it introduces additional stress into the SA. Compared to monitoring hardware schemes [16–18], our scheme significantly reduces hardware complexity, area overhead, and over-testing due to PV. Finally, schemes that change read operations' behavior (such as the DFT proposed in this work) have proven to be a suitable solution to detect RRFs; nevertheless, the way they are implemented may cause drawbacks. The scheme presented in [14] introduces unrealistic stress in the circuit to change the WL enable time, which may increase over-testing by up to 30%, according to the authors. Our solution aims at reducing over-testing by introducing minimal stress possible into the SA, i.e., only one 1-fin PMOS transistor. Finally, the DFT in [15] focus on the pre-charge of SAs. However, many additional transistors in the SA are necessary to counter the pre-charge, leading to additional noise and increased hardware complexity. We overcome this problem by focusing on the amplification phase instead, enabling the detection of RRF with less effort.

Drawbacks & Limitations: the main drawback is the increased SA design complexity. Due to additional transistors in the SA, further routing, coupling, and leakage issues may occur. A more negligible drawback is the impact on BL swing due to additional capacitive load in the SA: fault-free read operations have shown a mean BL swing decrease of $0.2\ \mathrm{mV}$ (from $160.2\ \mathrm{mV}$ to $160\ \mathrm{mV}$), an impact of only 0.12%. Finally, the DFT's main limitation is its lack of parametric control. As it is a functional test solution, it still requires an incorrect functional behavior to detect RRFs. To fully detect all parametric deviations on the BL swing, a parametric test solution that identifies reduced BL swings is required.

VI. Conclusion

In this work, we have presented an analysis of the occurrence and detection of *Random Read Faults* (RRFs) in FinFET SRAMs. We have shown that the failure rate of RRFs is directly related to the *bit line swing* (BLS), the number of read operations performed, and *stressing conditions* (SCs). Furthermore, we have proposed a new DFT scheme to detect RRFs by mismatching the SA during the amplification phase. The DFT significantly improves RRF coverage, thus leading to reduced test escapes and higher quality FinFET SRAMs.

References

[1] D. Bhattacharya *et al.*, "FinFETs: From Devices to Architectures," *Advances in Electronics*, sep 2014.

[2] Y. Liu *et al.*, "On modeling faults in FinFET logic circuits," in *Int. Test Conf.* IEEE, nov 2012.

[3] R. Dibaj *et al.*, "Comprehensive investigation of gate oxide short in finfets," in *VLSI Test Symp.* IEEE, apr 2017.

[4] F. Forero *et al.*, "Modeling and detectability of full open gate defects in finfet technology," *IEEE Tran. on VLSI Systems*, 2019.

[5] M. Fieback *et al.*, "Device-aware test: A new test approach towards DPPB," in *Int. Test Conf.*, 2019.

[6] S. Hamdioui *et al.*, "An experimental analysis of spot defects in SRAMs: realistic fault models and tests," in *Asian Test Symp.* IEEE, 2000.

[7] S. Davidson, "Towards an understanding of no trouble found devices," in *IEEE VTS 2005*, may 2005.

[8] H. Villacorta *et al.*, "FinFET SRAM hardening through des. and technol. param. considering process var." in *RADECS 2013*. IEEE, sep 2013.

[9] M. Shah *et al.*, "A Qual. and Rel. Driven DFT and DFR Strategy for Automot. and Ind. Markets," in *VLSI Test Symp.* IEEE, apr 2019.

[10] S. Hamdioui *et al.*, "March SS: a test for all static simple RAM faults," in *Int. Workshop on Memory Technol., Des. and Test*. IEEE, jul 2002.

[11] A. Benso *et al.*, "March AB, march AB1: new march tests for unlinked dynamic memory faults," in *Int. Conf. on Test*, oct 2005.

[12] G. Harutyunyan *et al.*, "Memory Physical Aware Multi-Level Fault Diagnosis Flow," *IEEE Tran. on Emerging Topics in Computing*, 2018.

[13] A. Meixner *et al.*, "Weak write test mode: an SRAM cell stability design for test technique," in *Int. Test Conf.*, nov 1997.

[14] J. Kinseher *et al.*, "Improving SRAM test quality by leveraging self-timed circuits," in *Des., Test, and Automation in Eur.* IEEE, mar 2016.

[15] G. C. Medeiros *et al.*, "A DFT scheme to improve coverage of hard-to-detect faults in FinFET SRAMs," in *Des., Test, and Automation in Eur.* IEEE, mar 2020.

[16] G. Medeiros *et al.*, "A defect-oriented test approach using on-chip current sensors for resistive defects in FinFET SRAMs," *Microelectronics Reliability*, sep 2018.

[17] M.-C. Chen *et al.*, "A BIST Scheme for Detect. Defec. in FinFET-Based SRAM Circuit," in *Asian Test Symposium*. IEEE, oct 2018.

[18] G. C. Medeiros *et al.*, "DFT scheme for hard-to-detect faults in FinFET SRAMs," in *Eur. Test Symp.* IEEE, may 2019.

[19] T. Matsukawa *et al.*, "Comprehensive analysis of variability sources of FinFET characteristics," *Symp. on VLSI Technol.*, jun 2009.

[20] A. J. van de Goor *et al.*, "Functional memory faults: a formal notation and a taxonomy," in *Asian Test Symp.* IEEE, apr 2000.

[21] R. D. Yates *et al.*, *Probability and Stochastic Processes: A Friendly Introduction for Electrical and Computer Engineers*. Wiley, 2005.

[22] A. J. van de Goor *et al.*, "Detecting faults in the peripheral circuits and an evaluation of SRAM tests," in *Int. Conf. on Test*. IEEE, oct 2004.

[23] S. Hamdioui *et al.*, "An investigation on capacitive coupling in RAM address decoders," in *Int. Design and Test Workshop*. IEEE, dec 2007.

[24] Nanoscale Integration and Modeling (NIMO), "Predictive Technology Model (PTM)," 2012. [Online]. Available: http://ptm.asu.edu/

[25] S. Hamdioui *et al.*, "Memory fault modeling trends: A case study," *Journal of Electronic Testing: Theory and Applications*, jun 2004.

[26] M. J. M. Pelgrom *et al.*, "Matching properties of mos transistors," *IEEE Journal of Solid-State Circuits*, 1989.

[27] S. Yu *et al.*, "Emerging memory technologies: Recent trends and prospects," *IEEE Solid-State Circuits Magazine*, 2016.