In-Field Monitoring and Preventing Read Disturb Faults in RRAMs

Xun, Hanzhi; Fieback, Moritz; Yuan, Sicong; Wang, Changhao; Hua, Erbing; Aziza, Hassen; Bishnoi, Rajendra; Taouil, Mottaqiallah; Hamdioui, Said; More Authors

DOI
10.1109/ETS63895.2025.11049639

Publication date
2025

Document Version
Final published version

Published in
Proceedings of the 2025 IEEE European Test Symposium (ETS)

In-Field Monitoring and Preventing Read Disturb Faults in RRAMs

Hanzhi Xun* Moritz Fieback* Sicong Yuan* Changhao Wang† Erbing Hua* Hassen Aziza‡
Leticia Bolzani Poehls§ Riccardo Cantoro† Rajendra Bishnoi* Mottaqiallah Taouil*¶ Said Hamdioui*¶
*Delft University of Technology, The Netherlands †Politecnico di Torino, Italy ‡Aix-Marseille University, France
§Leibniz Institute for High Performance Microelectronics, Germany ¶CognitiveIC, The Netherlands
Email: {h.xun, s.hamdioui}@tudelft.nl

2025 IEEE European Test Symposium (ETS) | 979-8-3315-9450-3/25/$31.00 ©2025 IEEE | DOI: 10.1109/ETS63895.2025.11049639

***Abstract*—Addressing non-idealities in Resistive Random Access Memories (RRAMs) is crucial for their successful commercialization. For example, the inherent resistance drift that occurs during consecutive read operations can induce Read Disturb Faults (RDF), leading to functional errors. This paper analyzes and characterizes the resistance drift and the RDF based on data measurements and presents a physics-based RRAM compact model that incorporates these non-idealities. Additionally, an in-field mitigation scheme is proposed, leveraging bidirectional read operations to balance the resistance. The scheme is implemented and validated through circuit simulations, both for RRAM used as memory and for RRAM-based computation-in-memory microarchitectures for deep neural networks. The results demonstrate that RRAM without any mitigation scheme can start failing after 8,000 consecutive reads, while our mitigation scheme ensures that the memory remains functional even after 10^6 consecutive reads. Furthermore, the results indicate that using the MNIST dataset as a case study, the accuracy can drop significantly from 86% to as low as 12.5% without any mitigation scheme. In contrast, the proposed mitigation scheme improves this accuracy up to 84.2%.**

***Index Terms*—RRAM reliability, Read disturb faults, AI, Neuromorphic computing**

I. Introduction

Resistive Random Access Memory (RRAM) is an attractive technique for achieving high-density non-volatile storage and facilitating a novel computer paradigm such as Computation in Memory (CIM) for edge AI [1–3]. However, RRAM devices are prone to non-idealities that can lead to functional errors during deployment [4–7]. Such non-idealities include both time-zero and time-dependent ones [8]. Time-zero non-idealities consist of device variation and wire parasitics, whereas time-dependent ones consist of endurance issues, device degradation, and resistance drift. Even if the RRAM chip passes the manufacturing test, it may suffer from in-field functional errors due to these non-idealities. For example, the resistance drift over time during consecutive read operations leads to bit flips referred to as the Read Disturb Faults (RDF) [9, 10]. Hence, it is essential to develop effective migration and recovery solutions to ensure the quality and reliability of RRAMs used as storage or as CIM devices.

This work has been partially funded by the Dutch Organization for Scientific Research (NWO) under grant number KICH1.ST04.22.021 for the project Self-Healing Neuromorphic Systems.

Research on reliability improvements and mitigation of the impact of non-idealities in RRAMs used as 'memory' or as 'computing device (CIM)' is still in its infancy stage. For RRAM as memories, Error Correction Codes (ECC) were proposed [11, 12] for the detection or recovery of hard and soft errors. In addition, an incremental RESET and verify technique was proposed to enhance both variations and reliability in RRAM memories [13]. However, these works only focus on non-idealities caused by write operations and have a high area or time overhead. The work in [14] applied a Monte Carlo-based methodology to investigate the read disturb mechanism, but it fails to develop a solution to mitigate it. The impact of non-idealities in RRAM-based CIM has been widely studied [8, 9], with mitigation approaches proposed at both the software (mapping) [15, 16] and hardware [8, 10, 17] levels. However, all these studies individually analyze non-idealities from a single perspective, lacking a comprehensive treatment that addresses all related issues. Moreover, many of them are highly vulnerable to process variations, limiting their robustness in practical scenarios. Clearly, there is still a need for robust and efficient approaches to deal with non-idealities in RRAM-based systems.

This paper focuses on resistance drift and read disturb fault, and proposes a mitigation scheme to address them. We utilize the bi-directional read scheme to alleviate the resistance drift. The approach can be used for manufacturing tests, yield learning, and even for in-field detection and recovery because it can monitor the fault behavior in each read cycle. The main contributions of this paper are:

- Analyze the resistance drift and RDF, providing measurements and simulations at both device and circuit levels.
- Propose test and mitigation schemes based on bi-directional read.
- Implement and validate the effectiveness of the different implementations both for RRAM as a memory as well as for an RRAM-based CIM architecture.

The rest of this paper is structured as follows. Section II establishes the RRAM basics. Section III analyzes the RDF and provides measurements and simulations. Section IV presents proposed approaches of bi-directional read schemes. Section V validates the approach and the impact on storage memory and CIM inference. Section VI discusses and concludes the paper.

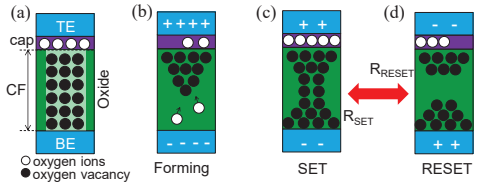

Fig. 1. (a) RRAM stack, (b) Forming, (c) SET, (d) RESET.

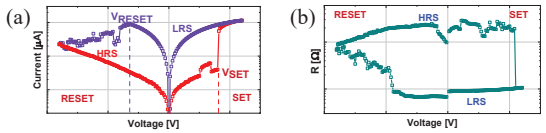

Fig. 2. (a) RRAM I-V curve in log scale, (b) RRAM R-V curve in log scale.

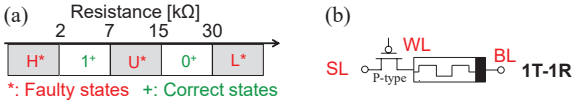

Fig. 3. (a) The five RRAM resistance states, (b) 1T-1R cell.

II. RRAM Basics and Background

An RRAM device consists of a Metal-Insulator-Metal (MIM) stack, as illustrated in Fig. 1 (a) [18]. In its organization, the middle metallic oxide is situated between the top and bottom metal electrodes (TE and BE), and is constructed with an additional capping layer (cap), which functions primarily as an oxygen reservoir, hence enhancing switching performance. Typically, an RRAM device necessitates a forming process, which is a post-manufacturing procedure involving the application of a high voltage between two electrodes to form a *CF* consisting of oxygen vacancies (OV), as shown in Fig. 1 (b).

Fig. 2 shows the switching I-V and R-V curves for a bipolar RRAM. The shape of the CF decides the different resistances of the device; the generation of more OV (Fig. 1 (c)), which is referred to as a SET operation, causes the CF length to rise when applying a positive voltage V_{TE} (across TE and BE) greater than a threshold V_{SET}. Oppositely, the dissolution of the CF (see Fig. 1 (d)) is referred to as a RESET operation and takes place when $V_{\mathrm{TE}} \leq V_{\mathrm{RESET}}$.

The formation and dissolution of the CF is a result of the stochastic O^{2-} movement, which causes variations in the resistance [18]. Hence, the RRAM (usually used as a binary device) can be divided into 5 states, as shown in Fig. 3 (a) [19]: 1) the faulty extremely high conductance state 'H', 2) the correct low resistive state '1', 3) the faulty undefined state 'U', 4) the correct high resistive state '0', and 5) the faulty extremely low conductance state 'L'.

Fig. 3 (b) shows a typical RRAM 1-Transistor-1-Resistor (1T-1R) cell with three terminals connecting to the Bit Line (BL), Source Line (SL), and Word Line (WL). The 1T-1R cell design effectively prevents sneak-path currents but introduces an additional voltage drop. Note that resistance switching under the SET voltage is easier than under the RESET voltage due to the thermal runaway [20–22]. Hence, a P-type transistor is employed as the selector in this work to facilitate the RESET operation. WL makes the cell accessible, while BL and SL are set to appropriate voltages for write (SET and RESET) and read operations.

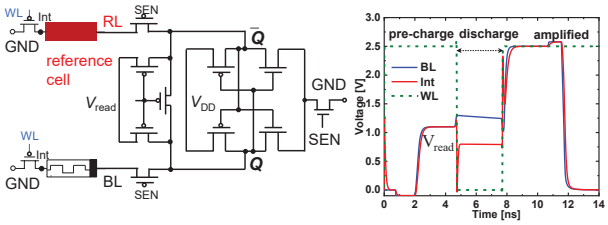

Fig. 4. Schematic of the pre-charged SA and voltages between the RRAM.

In this work, we use the 1T-1R circuit to build the memory cell array and the peripheral circuits needed to perform write and read operations. Since this paper mainly focuses on read disturbance, special attention is given to the SA design. During the read operation, a *pre-charged* SA applies specific voltages across the RRAM cell (see Fig. 4) and compares the current flowing through the selected memory cell with that of a *reference cell*. It consists of three phases: 1) Q and $\bar{Q}$ are pre-charged to V_{read}, 2) Q and $\bar{Q}$ discharge simultaneously along two paths (when the WL is active), and 3) after closing the WL, Q and $\bar{Q}$ are amplified to full voltage low and high as logic '0' and '1', respectively.

III. Physics and Characterization of RDFs

This section provides device physical properties behind the RDF, and gives the characterization of the RDF both at the device as well as the circuit level.

A. Physical phenomenon behind resistance drift

The RDF is a time-dependent phenomenon where reading a memory cell unintentionally alters its stored data, potentially causing a bit flip. The *resistance drift* of the device is behind this phenomenon [9, 10, 21–25]. Read operations utilize a small read current to avoid affecting the programmed state. However, extensive repeated read operations still stress the device, especially for applications that require multiple read operations, switching the state. The likelihood of unintended switching during read is affected by several aspects, including device variability, polarity of the read current, and device state. The RDF should be evaluated for both voltage polarities (SET and RESET). Specifically, reading in the SET direction poses greater risks due to broader voltage distributions and increased variability, potentially leading to unintended state changes. Conversely, applying read voltages in the RESET direction to devices initially in LRS has demonstrated comparatively limited drift, which can be explained by the hard RESET nature of RRAM devices [21, 22]. Hence, we target the resistance drift of HRS due to the read disturbance in this paper. Furthermore, resistance drift itself is non-linear and state-dependent (low resistance state is hard to SET further), influenced significantly by applied voltage amplitude and pulse duration (even at ns level) accumulated to the device.

B. Device-level RDF characterization

Next, we will provide the characterization of the RDF at a device level. 32 RRAM devices with a (BE/oxide/cap/TE) = (30 nm Pt/5 nm ZrO_2/20 nm Ta/30 nm Pt) stack are fabricated and characterized by applying different voltages in SET and

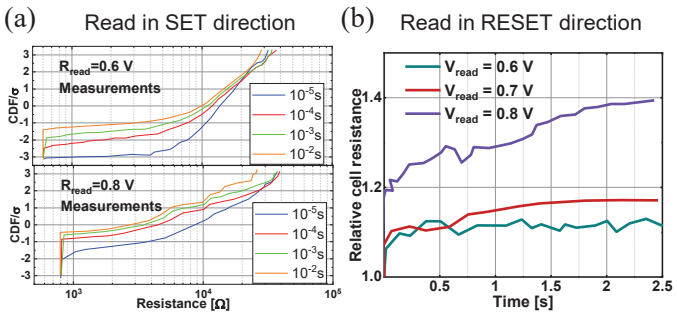

Fig. 5. Experiments [9]. (a) In SET direction, (b) In RESET direction.

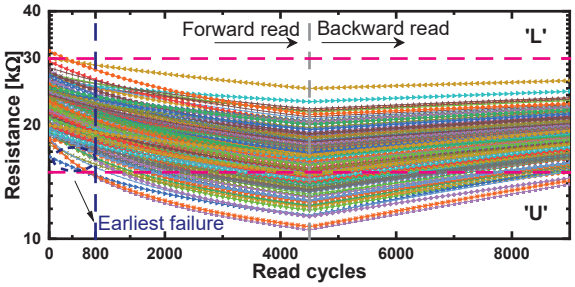

Fig. 6. The resistance changes when performing forward and backward read operations.

RESET polarity [9]. From the measurements, a device exhibits a nominally defect-free switching behavior, where LRS is defined with $[2\,\mathrm{k\Omega}, 7\,\mathrm{k\Omega}]$, and HRS with $[15\,\mathrm{k\Omega}, 30\,\mathrm{k\Omega}]$.

Fig. 5 (a) shows the resulting resistance distributions of measurements, as a Cumulative Distribution Function (CDF) under the standard normal distribution σ, for different pulse width ranges from 10^{-5} to 10^{-2}s applied in a SET polarity. Cells are in an initial HRS from $15\,\mathrm{k\Omega}$ to $25\,\mathrm{k\Omega}$. The resistance decreases as the applied voltage pulse becomes higher (e.g., from 0.6 to $0.8\,\mathrm{V}$) and wider (e.g., from 10^{-5} to 10^{-2} s), indicating a gradual shift from HRS towards LRS. Clearly, read in SET direction can cause unintended switching, e.g., more than 50% cells switch to LRS by using read pulse of $0.8\,\mathrm{V}$ height and 10^{-4}s width.

Fig. 5 (b) shows the result for reading devices configured in parallel with both LRS and HRS cells in the RESET direction. As observed, applying a read voltage in the RESET direction up to $0.8\,\mathrm{V}$ can increase the resistance slightly. Under the same applied voltage, the impact of RESET-direction reading on cell resistance is smaller compared to SET-direction reading, as rupturing CFs is inherently more difficult than forming them.

C. Circuit-level RDF characterization

We set up a simulation model consisting of a 1T-1R cell and a pre-charge based SA, see Fig. 4. We use the JART V1b var model for the RRAM device [26] and calibrate the model with the measurement presented in the previous section.

We applied 200 Monte Carlo (MC) iterations, while performing 4500 forward read operations in SET direction (when the device is in HRS with a mean value of $22\,\mathrm{k\Omega}$), as well as 4500 backward read operations in RESET direction.

Fig. 6 shows that cells in HRS exhibit the resistance decrease by applying forward read operations. This resistance drift is state-dependent and variable CF shapes of RRAMs

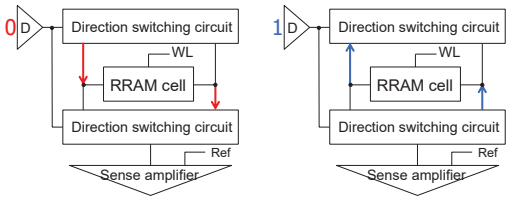

Fig. 7. The bi-directional read circuit with alternate current paths shown in red and blue.

induce the observed variation [26]. Hence, the same initialized resistance states exhibit different degrees of resistance increase or decrease with repeated read operations. For each iteration, the resistance decreases at a nonlinear rate with forward reading; it goes down faster at first and then slower.

Fig. 6 also shows that the resistance of cells is able to increase during backward read operations. Note that the 1T-1R cell has an asymmetry between SET and RESET directions (reading in SET direction induces a voltage drop). Consequently, all cells consistently exhibit a gradual increase in HRS when repeatedly read in the RESET direction.

Therefore, applying backward read operations can help mitigate resistance reduction, which provides a promising opportunity for bi-directional read approaches.

IV. The Bi-Directional Read Approach

This section proposes a mitigation scheme for the RDF; it is based on a bi-directional read of the RRAM devices. Different implementations are provided, together with a trade-off analysis of such implementations.

A. Concept and potential implementations

The resistance of bipolar RRAM either decreases (causing the RDF) or increases (recovering) by applying positive or negative voltages. Thus, a reverse read direction can boost device resistance and recover it from the RDF. Fig. 7 illustrates the concept of bi-directional read for RRAMs. It allows changing the reading current directions through the cell by selecting the control signal D. When D=0, a normal read (i.e., forward) is performed, causing a shift or decrease in the RRAM resistance. When D=1, the reverse read is performed, enabling the recovery of the RRAM resistance (see also Fig. 6). There are three schemes for such an approach: 1) a *static* method based on applying bi-directional reads with a fixed ratio, 2) an *enhanced static* method that can reduce costs, and 3) a *dynamic* method based on online monitoring of the RRAM state (using dedicated sensors to switch the read direction at the right time).

1) Static approach: The direct approach involves applying static implementations, i.e., bi-directional read operations with a *fixed* forward-to-backward ratio, typically determined through measurements to balance RRAM resistance change. These measurements are essential for characterization at both the device and circuit levels. However, it has some limitations; e.g., it cannot accommodate the wide variability RRAM is inherently suffering from in HRS, and it is relatively hard to implement when considering the management of the access of each address.

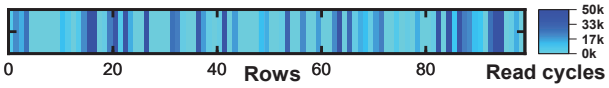

Fig. 8. Read cycle distribution across DNN array.

2) Enhanced static approach: This approach is designed to improve the aforementioned approach by accounting for the unequal distribution of workloads and data access frequency across the array. It has been reported that memory-cell accesses follow a power-law distribution, with approximately 84% of total accesses concentrated in the top 30% most frequently accessed cells [27]. For instance, Fig. 8 shows the results of a high-level simulation (C++) we performed using RRAM array model of a Deep Neural Network (DNN) for the MNIST classification task. The figure shows a heatmap depicting the distribution of read cycles across individual memory cells. The color bar represents the number of read accesses for each cell, with the gradient increasing from light blue to dark blue. Due to the inherent sparsity of neural networks, read operations for rows that correspond to zero-valued inputs are skipped, leading to different access frequencies per row. The heatmap illustrates read access patterns, indicating that certain parts of the array are accessed more frequently due to inputs of the DNN model. To efficiently track frequently accessed cells, we propose to keep a dynamically updated list of recently accessed addresses with the number of accesses per address. If the address is not on the list, it will be added on the top with the address count set to one. If an address is already on the list, it gets put on the top with its increased address count. If the list is full, and another address needs to be added, the bottom address will be removed from the list. In this way, the recovery scheme will only be applied to frequently accessed cells.

3) Dynamic approach: This approach can in-field monitor the RDF and adjust the read direction, at the right time. Fig. 9 shows the high-level implementation, including the core array (C0-C3), dummy cells (D0 and D1), and peripheral circuits. The dummy cells are placed in a single column, and share WL's and SL's drivers, as well as registers, per row. They act as sensors (with references R_{ref1}, R_{ref2}), tracking maximum read cycles and triggering read direction switching. Whenever a row cell is accessed, the corresponding dummy cell is simultaneously read, and its resistance is recorded; hence, it presents the worst-case scenario. Upon switching the dummy cell resistance state, the control signal D is triggered to execute the read operation in the reverse manner.

B. Trade-off for approach selection

Applying a static read ratio requires detailed measurements. Cell-based counters need to be used to implement the static approach. Depending on workloads, the scrolling list can help further apply the approach only to frequently accessed cells. However, the blind approaches are limited by inadequate consideration of variations. Thus, including the real-time sensor to switch read direction properly when measurements are insufficient is more reliable and robust to variations, as opposed to using a dummy cell that typically assumes worst-case conditions, leading to possible over-repair.

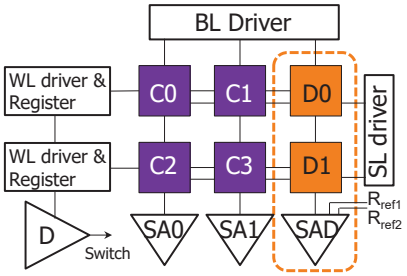

Fig. 9. The high-level implementation of the in-field test approach.

V. Validation

This section shows the validation of the bi-directional read approaches. First, we briefly present the simulation setup. Then, through simulation results, we demonstrate that the proposed approaches are effective in addressing the RDF. Finally, we analyze the feasibility of the bipolar read approach for improving the accuracy of CIM implementation.

A. Simulation set up

To estimate the read disturb behavior, the RRAM resistance drift under periodic read was simulated for HRS. The RRAM array circuit in Fig. 9 is built in Cadence's Spectre simulator by using the TSMC $40\,\text{nm}$ transistor library and the physics-based JART VCM v1b [26, 28] RRAM compact model, which is calibrated with measurements in [9]. The pre-charged SA performs the bi-directional read operations with the control signal D. The positive read is performed by the current discharge from the pre-charged SA to the SL, while the negative read is performed by the current discharge from the pre-charged SA to the BL. The supply voltage $V_{\text{DD}} = 2.5\,\text{V}$ and read voltage $V_{\text{read}} = 1.1\,\text{V}$. The V_{read} is selected to sufficiently surpass the transistor voltage drop (V_{th} is around $0.8\,\text{V}$) for current sensing. The read period is $12\,\text{ns}$ for both positive and negative read operations.

B. Validations for storage memory array

1) Validation of the static approach: To validate the approach, a $m:n$ switching ratio is implemented by performing m forward reads, followed by n backward reads. We set the ratio to $r = \frac{n}{m} = 1$. Fig. 10 shows the final resistance states after 40k read cycles for three cases: a) by performing regular read operations (i.e., only forward read operations), b) by performing bi-directional read with ratio 1 assuming the ideal scenario (i.e., without variations), and c) the same as in (b), but taking the variability into consideration. The figure shows that applying regular reads will cause HRS to drift to $10\,\text{k}\Omega$ after 40k cycles, and become undefined already after 8k reads. On the other hand, bi-directional read is quite effective to mitigate RDF. However, in the extreme variety cases, the scheme may fail; the bold gray line in the figure for bi-directional read (with variations) shows that the cell enters an undefined state.

2) Validation of the enhanced static approach: For our experiment, we assume 1M read accesses to a 256×256 RRAM array; $80\,\%$ of these reads are randomly applied to cells inside a core subarray of size $s \times s$ (which are the ones to likely cause RDF), while $20\,\%$ are applied to other cells

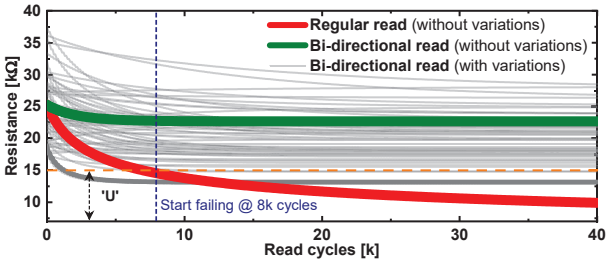

Fig. 10. Resistance drifts for bi-directional read compared to regular read.

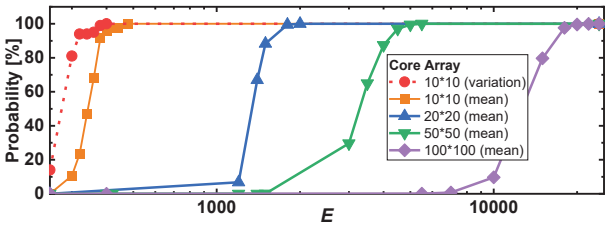

Fig. 11. The probability of counting cells with RDF.

[27]. We examined the dependency of the list size, in terms of entries E, on the percentage of the recorded cells exhibiting the RDF. The results are shown in Fig. 11 for five cases; note that for s=10, two options are reported (with and without considering variability). In the figure, each solid line shows the probability of recording cells starting to fail; as shown in the previous section, the RRAM cells start failing after 8k read cycles (see also Fig. 10). As expected, the minimum E to achieve 100% recording of RDF cells increases gradually as the core subarray size grows; however, E increases much slower than the increase in the core subarray size. For example, the minimum E needed in case $s = 100$ is about $50\times$ larger than that needed when $s = 10$. It is also worth noting that when considering the variability, the minimum required E is relatively smaller than in the mean case. Limitations of this approach include: a) power and area overhead for scrolling the update list, and b) test escapes due to insufficient E when cell workload is evenly distributed.

3) Validation of the dynamic approach: This approach is validated in the circuit by switching to perform backward reads and thus repairing memory cells when the sensor detects the RDFs of dummy cells. Fig. 12 shows how the resistance of the RRAM device degrades over time when performing regular reads and how bi-directional read enables the recovery after 1000 cycles; note that the reference value is set to $R_{\text{ref1}} = 15\,\text{k}\Omega$ (see section IV). The resistance decreases from $15.5\,\text{k}\Omega$ to $14.5\,\text{k}\Omega$ after 2000 consecutive regular read operations. With the dynamic approach, the output of the SA changes from GND (read output for '0') to V_{DD} (read output for '1') after 1000 regular reads, thus triggering the backward read direction to repair the cell. Hence, it can be repaired to $15.5\,\text{k}\Omega$ by performing 1000 backward read operations after 1000 forward read operations. The approach provides correct in-field testing and repair with the application of a bi-directional read scheme. Furthermore, the robustness is validated by assuming the worst case with variations: the dummy cell resistance decreases very fast with forward reads until the boundary (700 cycles, see Fig. 6) and triggers cells to

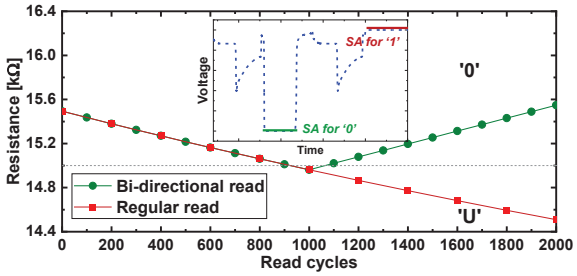

Fig. 12. The effectiveness of the in-field test and repair approach.

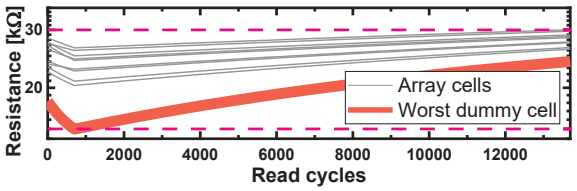

Fig. 13. The worst case of the dummy cell setting with variations.

perform backward reads. As shown in Fig. 13, by performing 700 forward reads and 13k backward reads on other cells that decrease the least, they are still within the HRS specification.

In conclusion, the dynamic approach can easily ensure the robustness of RRAM devices against RDFs; the exact resistance value of the RRAM does not matter as long as we guarantee that it is within the specification of the '0' state.

C. Validations for system-level CIM inference

As the impact of RDF intensifies with frequent read tasks, we validate the proposed approaches using an RRAM-based CIM inference engine [29]. The applied platform is developed to emulate the online learning classification task with an MNIST handwritten dataset in a 2-layer ($400\times100\times10$) DNN-based RRAM array. Each weight is mapped to one RRAM cell. We simulate the RDF as resistance drift according to read cycles only for hardware inference, as the hardware training is assumed to be fault-free. The accuracy is determined as the maximal value from 30 epochs in each inference. As shown in Fig. 14, regular reads up to 27k cycles lead to a significant accuracy drop, from 86% to 12.5% without variations, and from 48.32% to 10.27% under worst-case variation conditions.

1) Validation of the static and enhanced static approaches: The green line in Fig. 14 shows that applying the static approach improves the inference accuracy from $12.5\,\%$ to $60.69\,\%$ after 27k read cycles, compared with applying regular reads. Considering the worst case with variations, the accuracy starts at $46.47\,\%$ and drops to $23.25\,\%$ after 27k read cycles, which is still higher than the accuracy using regular reads without variations. The enhanced static approach can be applied to the static approach, achieving the same accuracy in a more cost-effective manner.

2) Validation of the dynamic approach: Fig. 14 shows the validation of the dynamic approach. The reference values are set to $R_{\text{ref1}} = 20\,\text{k}\Omega$, $R_{\text{ref2}} = 30\,\text{k}\Omega$. When the read cycles reach up to 5k, the accuracy first drops and recovers to $65.09\,\%$ since the resistance decreases and triggers the sensor. Then, the accuracy is enhanced to $84.19\,\%$ when the resistance is recovered. After that, the resistance ($> 25\,\text{k}\Omega$)

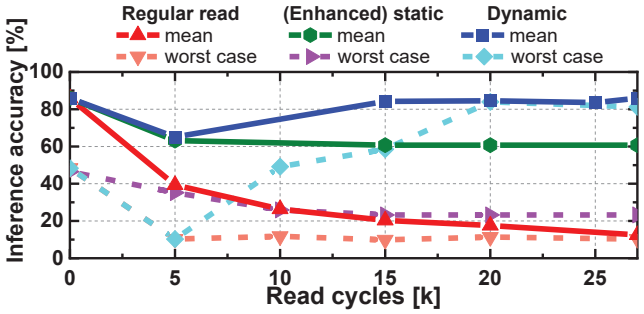

Fig. 14. Inference accuracy enhancement by bi-directional read approaches.

keeps increasing until 30 kΩ, resulting in a slight drop in accuracy but still larger than 80 % up to 25k cycles. Finally, the sensor is triggered again, recovering the accuracy to 86%, after which the process repeats. Furthermore, the worst-case is considered as all array cells drop fast (orange line in Fig. 13) but the dummy cell drops and triggers the read direction slowly. The accuracy first drops until the dummy cell triggers the sensor. Then, the accuracy is enhanced to 80.5 % up to 27k cycles, which shows the robustness against the variation.

In conclusion, the (in-field) dynamic approach is robust to variations and validated to improve the CIM accuracy. Besides, further enhancements of the dynamic approach are possible by fine-tuning its reference range (e.g., changing R_{ref1} and R_{ref2}), enabling early repair of cells with small margins.

VI. Discussion and Conclusion

This paper proposed a bi-directional read system that detects and resolves the RDF in the RRAM array. We investigate the resistance drift referred to as the RDF with device-level analysis and measurements. We propose and implement bi-directional read approaches at a circuit level. The approaches that alleviate the RDF are validated to compare with the regular read operations. Furthermore, the inference accuracy drop due to the RDF in the RRAM-based CIM is investigated and enhanced by the proposed approach, which shows the effectiveness of our designs. It is worth noting the following:

- **Integration with other Design for Tests (DfT):** The proposed bi-directional scheme can and needs to be used with other DFTs, such as multi-reference trimming design, to facilitate the development of testing methods.
- **Actual costs and benefits:** The proposed three approaches all require extra circuits for implementation. The static approach needs cell-based counters to apply the designed ratio between forward and backward reads. The cell-based counter provides precise control over each individual cell in the array; however, this also results in an area overhead comparable to the array itself. In this case, the enhanced static approach can significantly reduce the overhead, as it enables precise control over frequently accessed cells. The overhead of the scrolling list recorder varies with the workload and was found to be less than 1% (480/65536) of the array in one case study (Fig. 11). The dynamic approach employs row-based monitoring with dummy cells, making it particularly efficient as the array's row dimension scales up.

References

[1] H.-S. P. Wong *et al.*, "Metal–Oxide RRAM," *P. IEEE*, vol. 100, no. 6, pp. 1951–1970, 2012.

[2] S. Hamdioui *et al.*, "Applications of Computation-In-Memory Architectures Based on Memristive Devices," in *DATE*, 2019.

[3] A. Singh *et al.*, "Low-Power Memristor-Based Computing for Edge-AI Applications," in *ISCAS*, 2021.

[4] D. Niu *et al.*, "Impact of Process Variations on Emerging Memristor," in *DAC*, 2010.

[5] A. Grossi *et al.*, "Impact of Intercell and Intracell Variability on Forming and Switching Parameters in RRAM Arrays," *TED*, vol. 62, no. 8, pp. 2502–2509, 2015.

[6] E. I. Vatajelu *et al.*, "Challenges and Solutions in Emerging Memory Testing," *TETC*, vol. 7, no. 3, pp. 493–506, 2019.

[7] H. Xun *et al.*, "Robust Design-for-Testability Scheme for Conventional and Unique Defects in RRAMs," in *ITC*, 2024.

[8] A. Gebregiorgis *et al.*, "Dealing with Non-Idealities in Memristor based Computation-in-Memory Designs," in *VLSI-SoC*, 2022.

[9] C. Bengel *et al.*, "Reliability Aspects of Binary Vector-Matrix-Multiplications Using ReRAM Devices," *Neuromorph. Comput. Eng.*, vol. 2, no. 3, p. 034 001, 2022.

[10] W. Shim *et al.*, "Investigation of Read Disturb and Bipolar Read Scheme on Multilevel RRAM-Based Deep Learning Inference Engine," *TED*, vol. 67, no. 6, pp. 2318–2323, 2020.

[11] C.-L. Su *et al.*, "An Integrated ECC and Redundancy Repair Scheme for Memory Reliability Enhancement," in *DFT*, 2005.

[12] G. Mayuga *et al.*, "An ECC-Based Memory Architecture with Online Self-Repair Capabilities for Reliability Enhancement," in *ETS*, 2015.

[13] A. Baroni *et al.*, "Tackling the Low Conductance State Drift through Incremental Reset and Verify in RRAM arrays," in *IIRW*, 2021.

[14] N. Raghavan *et al.*, "Monte Carlo Model of RESET Stochastics and Failure Rate Estimation of Read Disturb Mechanism in HfO_x RRAM," in *IRPS*, 2015.

[15] D. Gaol *et al.*, "Reliable Memristor-based Neuromorphic Design Using Variation- and Defect-Aware Training," in *ICCAD*, 2021.

[16] S. Diware *et al.*, "Mapping-Aware Biased Training for Accurate Memristor-based Neural Networks," in *AICAS*, 2023.

[17] M. A. Yaldagard *et al.*, "Read-Disturb Detection Methodology for RRAM-Based Computation-In-Memory Architecture," in *AICAS*, 2023.

[18] S. Yu *et al.*, "On the Stochastic Nature of Resistive Switching in Metal Oxide RRAM: Physical Modeling, Monte Carlo Simulation, and Experimental Characterization," in *IEDM*, 2011.

[19] N. Z. Haron *et al.*, "DfT Schemes for Resistive Open Defects in RRAMs," in *DATE*, 2012.

[20] K. Fleck *et al.*, "Uniting Gradual and Abrupt Set Processes in Resistive Switching Oxides," *Phys. Rev. Appl.*, vol. 6, no. 6, p. 064 015, 2016.

[21] H. Li *et al.*, "A SPICE Model of Resistive Random Access Memory for Large-Scale Memory Array Simulation," *EDL*, vol. 35, no. 2, pp. 211–213, 2014.

[22] F. Nardi *et al.*, "Resistive Switching by Voltage-Driven Ion Migration in Bipolar RRAM—Part I: Experimental Study," *TED*, vol. 59, no. 9, pp. 2461–2467, 2012.

[23] V. Mahboubi *et al.*, "On the Fine Tuning of RRAM Resistance Under Variability Using Current Pulses at SET," in *IOLTS*, 2024.

[24] H. Xun *et al.*, "Device-Aware Test for Ion Depletion Defects in RRAMs," in *ITC*, 2023.

[25] J. Park *et al.*, "Improving Endurance and Reliability by Optimizing the Alternating Voltage in Pt/ZnO/TiN RRAM," *Results Phys.*, vol. 39, p. 105 731, 2022.

[26] C. Bengel *et al.*, "Variability-Aware Modeling of Filamentary Oxide-Based Bipolar Resistive Switching Cells Using SPICE Level Compact Models," *Trans. Circuits Syst. I, Regul. Pap.*, vol. 67, no. 12, pp. 4618–4630, 2020.

[27] H. Kim *et al.*, "RecPIM: A PIM-Enabled DRAM-RRAM Hybrid Memory System For Recommendation Models," in *ISLPED*, 2023.

[28] F. Cüppers *et al.*, "Exploiting the Switching Dynamics of HfO_2-based ReRAM Devices for Reliable Analog Memristive Behavior," *APL Mater.*, vol. 7, no. 9, p. 091 105, 2019.

[29] P.-Y. Chen *et al.*, "NeuroSim+: An Integrated Device-to-Algorithm Framework for Benchmarking Synaptic Devices and Array Architectures," in *IEDM*, 2017.