Exploration of the viability of TiN/TiO_X ReRAM in Computational Random-Access Memory (CRAM)

Brandon R. Zink
Alternative Computing Group
National Institute of Standards and Technology
Gaithersburg, MD
Brandon.zink@nist.gov

Advait Madhavan
Alternative Computing Group
National Institute of Standards and Technology
Gaithersburg, MD
Advait.madhavan@nist.gov

William A. Borders
Alternative Computing Group
National Institute of Standards and Technology
Gaithersburg, MD
William.borders@nist.gov

Jabez McClelland
Alternative Computing Group
National Institute of Standards and Technology
Gaithersburg, MD
Jabez.mcclelland@nist.gov

***Abstract*—In-memory computing is a promising solution for solving the von-Neumann bottleneck. In particular, computational random-access memory (CRAM) is a promising form of in-memory computing where cascading logic operations can be performed directly within the memory array. However, a recent experiment utilizing magnetoresistve devices as the memory element in CRAM only gave the correct answer in 63 % of trials. One way to improve the accuracy is to build CRAM cells using resistive devices with larger ON/OFF ratios. In this study, we explore the performance of CRAM using resistive random-access memory (ReRAM) cells. Using experimental data obtained from various TiN/TiOx-based ReRAM devices in Monte Carlo simulations, we determine that the performance of the full adder operation using ReRAM based CRAM is still subject to the same inaccuracies as CRAM that utilizes magnetoresistive devices. However, our analysis reveals that by reducing the write voltages and removing the effects of complementary resistive switching in the ReRAM devices, 100 % accuracy over 100,000 trials can be achieved.**

Keywords—In-Memory Computing, Computational Random-Access Memory, Resistive Random-Access Memory

I. Introduction

The separation between memory and computing units in modern computer architectures increases the energy consumption due to high stand-by power [1-3] and introduces large computation delays due to the Von-Neumann bottleneck. Several studies have addressed this issue by proposing designs that reduce the physical distance between the memory and computing units [4-5]. These have been successful in alleviating some of the shortcomings, however, the only way to truly eliminate the von-Neumann bottleneck and the energy consumption from data transfer is to perform logic operations directly within the memory array using a type of nonvolatile memory technology.

In recent years, several studies have proposed various designs and methods for achieving in-memory computing. In a recent review [6], Z. Sun et al. categorized in-memory computing designs into six possible configurations, one of which is labelled XYZ-CIM, which is a promising solution for performing cascading logic operations [7-11]. This configuration is considered to be the most 'ideal' form of in-memory computing since both the input cells (X and Y) and the output cell (Z) are stored in the memory array and used for Boolean logic operations. One example of an XYZ-CIM configuration is called computational random-access memory (CRAM). This configuration consists of an array of cells comprised of one non-volatile memory unit and two transistors, allowing separation of current paths for memory and logic operations [11-14]. Previous theoretical studies found CRAM to be more energy efficient than modern CMOS-based near-memory processors in certain applications involving large-scale data analytics. For example, when comparing CRAM to CMOS-based near memory processors, M. Zabihi et al. showed that CRAM has a 23x and 40x reduction in total energy consumption for 2D image filtering and digit recognition, respectively [11]. Z. Chowdhury et al showed a 1.8x reduction in energy consumption for CRAM compared to CMOS based processing near memory in bioinformatics applications [12]. In each study, the reduction in energy consumption was attributed to the elimination of large data movement between memory and processor units. Y. Lv et al recently performed the first hardware demonstration of a full adder operation in CRAM using magnetic tunnel junctions (MTJs) as the non-volatile memory units [15]. While this experiment is a very important milestone towards the realization of true in-memory computing, the results revealed that CRAM had accuracies as low as 63 % under certain input configurations. This low accuracy was attributed to the low ON/OFF ratio of MTJs, which typically have an ON/OFF ratio of 2.

The purpose of this study is to attempt to address the low output accuracy in CRAM by evaluating the performance of CRAM based on resistive random-access memory (ReRAM) instead of MTJ-based CRAM. ReRAM devices were chosen because they have an ON/OFF ratio typically exceeding 100, and can even reach 10^5 in some cases [16]. The high ON/OFF ratio combined with their nonvolatility makes them ideal candidates for improving CRAM performance. Our analysis combines experimental data acquired from TiN/TiO_X ReRAM devices fabricated at the NanoFab in NIST (results of which can be seen in [17]) with Monte Carlo simulations that take into account several factors such as device switching probability with write voltage, resistance dependence on voltage, and device-to-device and cycle-to-cycle variations in resistance. Our simulations provide realistic results for the performance of ReRAM-based CRAM and show potential for a significant improvement in the full adder operation when compared to the previous experimental results.

979-8-3315-4127-9/24/$31.00 ©2024 IEEE

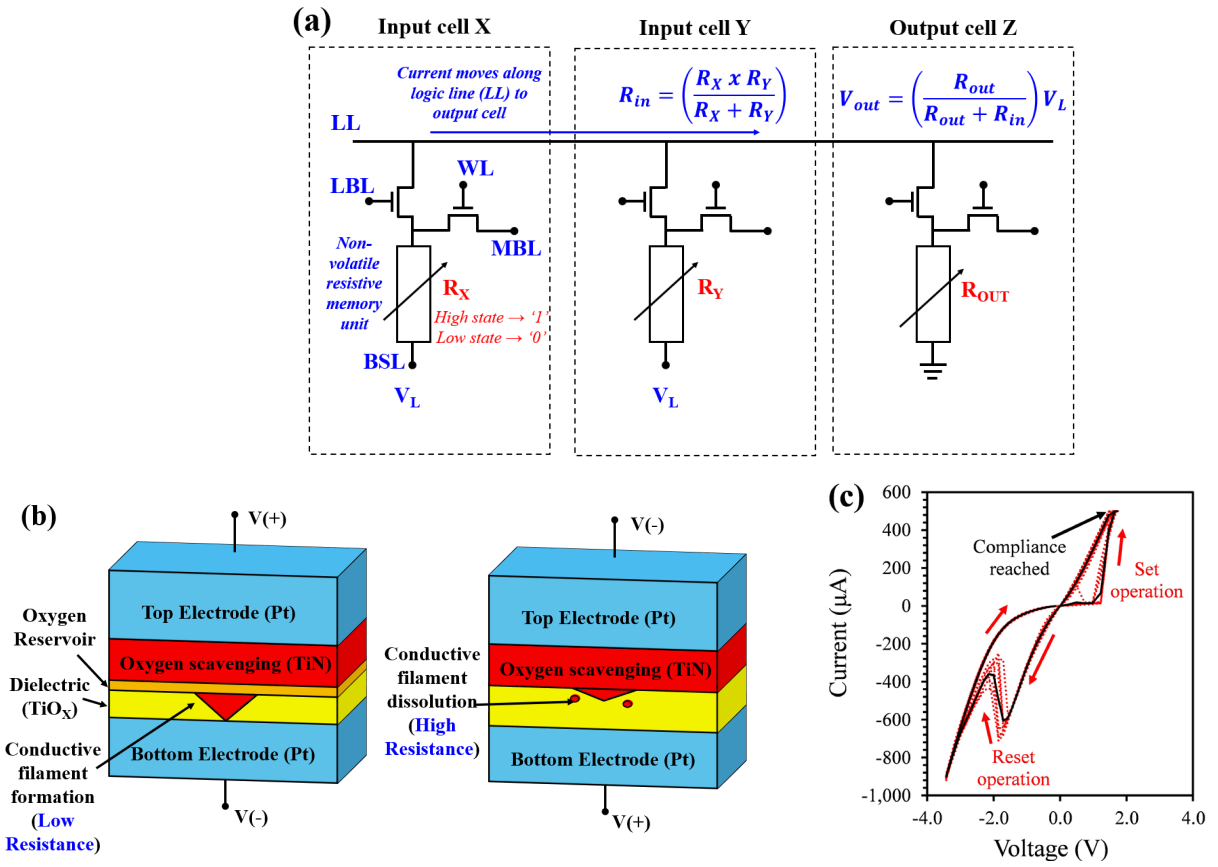

Fig. 1. (a) CRAM circuit diagram, **(b)** illustration of ReRAM switching mechanism, and **(c)** example of current vs voltage (I-V) switching characteristics in ReRAM for sample set 1 at a reset voltage of -3.5 V and current compliance of500 μA. The solid black line shows the I-V curve for the current values averaged over 10 consecutive I-V sweeps, the dotted red lines show all 10 individual I-V sweeps, and the red arrows show how the voltage is ramped during the I-V sweep, thus illustrating the polarity of the voltages during the set and reset operations.

II. Methods and Benchmarking

A. Background

A schematic for CRAM is shown in Fig. 1(a). This figure shows three CRAM cells used to perform Boolean logic operations, where two cells are used as inputs (X and Y) and one is used as an output (Z). In CRAM, memory-write and memory-read operations for each cell are done via the memory bit lines (MBL) and the logic operations are performed by activating the word lines (WL). This provides an electrical connection between each cell via the logic line (LL). During the logic operation, a voltage (V_L) is applied to the input cells and the voltage at the output (V_{OUT}) is dependent on the combined resistance of the input cells (R_{IN}), where $R_{IN} = (R_X R_Y)/(R_X+R_Y)$. Note that the resistance of each cell can be in one of two states, the low resistance state (corresponds to binary '0') or the high resistance state (corresponds to binary '1'). Various logic operations can be performed in CRAM depending on the magnitude of V_{OUT} and the initial resistance state of the output cell. For example, AND logic can be performed by setting the output cell, Z, to state '1' and applying V_L so that Z switches to '0' when the input states (X,Y) are (0,0), (0,1), or (1,0), but not when the input states are (1,1). NAND operation follows a very similar principle except that the output is initialized to state '0' and the opposite polarity of V_L is used to facilitate '0' to '1' switching. Table 1 shows the conditions for the initial state of the output and the conditions for V_L for AND, NAND, OR, and NOR logic.

TABLE I. Logic operations in CRAM.

Logic Operation	Output Preset state	Conditions for voltage at the output cell (V_{OUT})
AND	1 ($V_L > 0$)	$V_{OUT} < V_{THRESH}$ when inputs are (1,1) $V_{OUT} > V_{THRESH}$ otherwise
NAND	0 ($V_L < 0$)	
OR	1 ($V_L > 0$)	$V_{OUT} < V_{THRESH}$ when inputs are (0,0) $V_{OUT} > V_{THRESH}$ otherwise
NOR	0 ($V_L < 0$)	

V_{THRESH} = minimum voltage required to switch state of Z.

The switching mechanism for ReRAM devices is illustrated in Fig. 1(b). A ReRAM device consists of a bottom electrode, a dielectric layer, an oxygen scavenging layer, and a top electrode. For the devices we tested, the materials for these layers were Pt, TiO_X, TiN, and Pt, respectively. Upon application of a positive voltage across the stack, the oxygen scavenging layer introduces defects into the dielectric layer, causing oxygen ions to migrate to the interface of the dielectric and oxygen vacancies to accumulate within the dielectric. The result is a conductive filament, which reduces the resistance of the device and sets it into the low-resistance state (binary '0'). This process is referred to as the set operation. A current compliance is typically used during the set operation to avoid dielectric breakdown caused by a sudden increase in current flowing through the dielectric layer the moment the conductive filament forms. When a negative

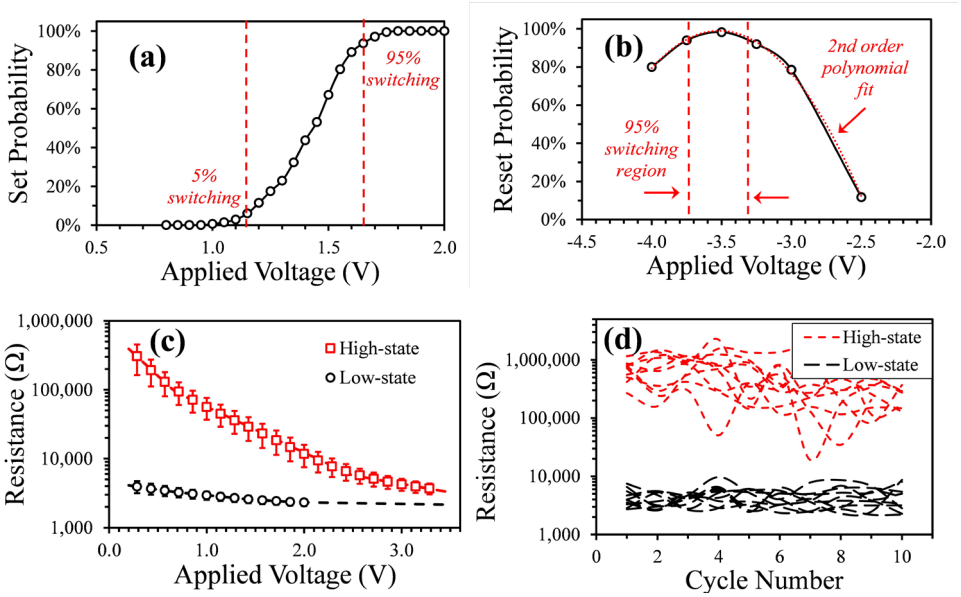

Fig. 2. Experimental data used for simulations in CRAM. Data shown was obtained from sample set 1 at a reset voltage of -3.5 V and a current compliance of 1 mA. **(a)** Switching probability distribution in the set operation, **(b)** switching probability distribution in the reset operation, **(c)** high state and low state resistance dependence on voltage and their exponential fit (represented as dotted lines), and **(d)** high state and low state resistance versus cycle number for 10 devices in sample set 1 illustrating the effects of cycle-to-cycle variations. The error bars in **(c)** represent the standard deviation among all the resistance measurements for a given voltage/cycle.

voltage is applied, dissolution of the conductive filament occurs and oxygen ions migrate back into the dielectric layer, thus setting the device back to the high-resistance state (binary '1'). This process is referred to as the reset operation. Fig. 1(c) shows the I-V hysteresis characteristics obtained from one of our ReRAM devices tested. This plot illustrates both the set operation at positive voltages, as seen in the sudden increase in the measured current, and the reset operation at negative voltages. Note that the data for this plot was acquired through 10 consecutive I-V sweeps on the same device. The solid black line represents the average current measurements at each voltage and the dotted red lines shows each individual I-V sweep for 10 consecutive measurements. Fig 1(c) illustrates that the ReRAM devices had some variation between cycles, which will be addressed in the following sections.

B. ReRAM device properties

For our simulations, we used data from three different ReRAM sample sets, the properties of which are listed in Table 2, where the uncertainty represents the standard deviation in the measurements over 35 devices. The main difference in these three sample sets is the layer thicknesses. Sample set 1 has a TiO_X thickness of 10 nm whereas sample sets 2 and 3 have TiO_X thicknesses of 5 nm. This means that sample set 1 has a higher resistance in the high-resistance state and a higher ON-OFF ratio; however, the results presented in [17] also show that this sample set has larger variation in resistances between devices. Sample set 2 has a TiN thickness of 10 nm whereas sample set 3 has a TiN thickness of 20 nm. In theory, sample set 3 should have a higher success rate for the reset operation; however, both

TABLE II. PROPERTIES OF SAMPLE SETS TESTED

	Sample set 1	**Sample set 2**	**Sample set 3**
TiN/TiO_X thickness	15 nm/10 nm	10 nm/5 nm	20 nm/5 nm
ON/OFF ratio	125 ± 30.0	25.3 ± 3.21	27.9 ± 2.79
High state resistance	(520 ± 48.5) kΩ	(68.6 ± 4.59) kΩ	(78.4 ± 5.37) kΩ
Low state resistance	(4.20 ± 0.157) kΩ	(2.78 ± 0.056) kΩ	(3.05 ± 0.117) kΩ

samples achieved 100 % reset probability under certain testing voltages.

In [17], we test 6 ReRAM wafers with 100 mm diameters containing at least 1,100 ReRAM devices at reset voltages ranging between -2 V to -4 V and at compliance currents ranging between 100 µA to 1 mA. We also studied the influence of background oxygen during etching. However, for the analysis in the current study, we only consider the three most promising sets.

C. Parameters for simulation

There are four key factors considered for our simulations: success rates for the set and reset operations, resistance and its dependence on voltage, variations in resistance among devices and variation in resistance between I-V cycles[1]. The success rate was simply defined as the percentage of devices that switched to the appropriate resistance state at a given voltage during the set and reset operations. Examples of success rates of the set and reset operations vs voltage are shown in Figs. 2(a) and 2(b),

[1] In each data set, there were a few outliers that caused particularly large standard deviations. As a result, the standard deviations actually included nearly 90 % of the measurements in all cases.

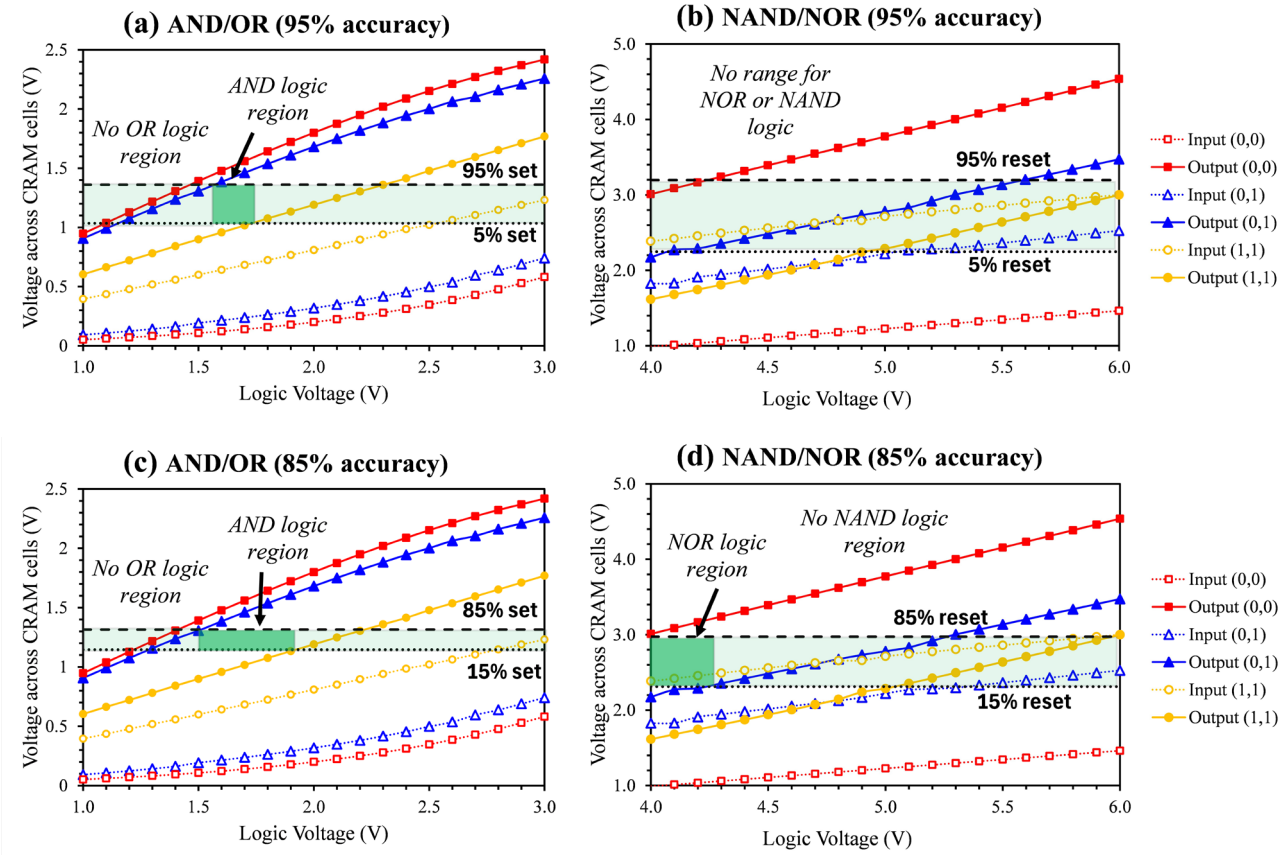

Fig. 3. Calculations of voltage across the input cells and output cells with respect to logic voltage under ideal conditions (no resistance variations) for sample set 2 for **(a,c)** AND/OR logic with 95 % and 85 % accuracy, respectively, and **(b,d)** NAND/NOR logic with 95 % and 85 % accuracy, respectively. **(a)** and **(c)** reveal a range of logic voltages for AND logic at both 95 % and 85 % accuracy, but no range of logic voltages exist for OR logic. **(b)** and **(d)** show that no region exists for NAND or NOR logic at 95 % accuracy, but a region for NOR logic arises when accuracy is 85 %.

respectively. These data sets was collected for sample set 1 at a current compliance of 1 mA. For the set operation, the cumulative probability distribution curves, such as the one seen in Fig. 2(a), were obtained by determining the percentage of devices that switched to the low resistance state at a given voltage. The data seen in Fig. 2(b) were obtained in a slightly different manner. Sets of 35 ReRAM devices were tested at fixed reset voltages ranging from -2 V to -4 V. Given that 10 consecutive I-V sweeps were applied to each device, there were a total of 350 data series used to calculate each point in Fig. 2(b). A trial was considered a successful reset operation if the maximum ON/OFF ratio exceeded 2. Cumulative probability distribution plots like these were obtained for all three sample sets and were then used to determine the set or reset probability in our simulations for a given logic voltage. For the set operation, the set probabilities in our simulations were calculated through linear interpolation between adjacent data points on the cumulative probability distribution data for the set operation for a given voltage. However, for the reset operation, the reset probabilities in our simulation were calculated by fitting a second-order polynomial function to the reset probability data. This was done because, for all sample sets, the reset probability peaked at around -3.5 V, then decreased as the reset voltage increased due to the effects of complimentary resistive switching [18-19].

The next factor that was considered was the dependence of the high and low state resistance on voltage, and this is shown in Fig. 2(c). The example shown in Fig. 2(c) was collected from sample set 1 at a current compliance of 1 mA, but the same measurements were also acquired for all sample sets at current compliances of 100 µA, 250 µA, 500 µA, and 1 mA, and at a reset voltage of -3.5 V, since this voltage had the highest success rate for the reset operation for all sample sets. These data were obtained by measuring the low and high state resistances at each voltage for 35 ReRAM devices per data set and 10 I-V sweeps per device (350 data sets total). The high and low state resistances at each point along the voltage axis between 0 and -2 V were calculated using I-V data to obtain the resistance versus voltage data shown in Fig. 2(c). Note that positive voltages were not used for the resistance calculations since a current compliance was used during the set operation (recall Fig. 1(c)), which would result in inaccurate resistance calculations at high voltages. Therefore, we assumed that the resistance vs voltage characteristics were symmetric between positive and negative voltages. These data were used to determine the voltages across the input and output CRAM cells during logic operations. Details of how these voltages were determined are provided in section III-A.

The last two factors that were accounted for in our simulations were device-to-device and cycle-to-cycle variations in the high and low state resistances. Device-to-device variations were determined by averaging the resistance measurements over 10 I-V sweeps and calculating the standard deviation among those measurements. The error bars in Fig. 2(c) show the standard deviations from device-to-device in the high and low state resistances for sample set 1. Fig. 2(d) shows high and low state resistance measurements for 10 devices in sample set 1, measured at 245 mV in the I-V sweep). These data illustrate that the resistance of the ReRAM devices fluctuates randomly between cycles as well as between devices. Because of this, the standard deviation among cycles was also considered in our simulation. The process of implementing these deviations in our simulation is described in further detail in section III-B.

III. Results

Our simulations were divided into two steps. The first was to calculate the resistances of each cell and voltages across the input and output cells (V_{IN} and V_{OUT}, respectively) under a wide range of logic voltages. These calculations are described in further detail in section III-A. The second step was to determine the accuracy of AND/NAND logic operations in CRAM via Monte Carlo simulations using the calculations made in the first step as an initial approximation for the input and output resistances. These simulations are described in further detail in section III-B.

A. Logic Voltage

Prior to determining the accuracy of logic in CRAM using the ReRAM results, we needed to calculate the voltage across the input and output cells over a range of logic voltages. These calculations were carried out at three input combinations (0,0), (0,1), and (1,1) and for AND/OR and NAND/NOR logic. Note that, since line resistance is not considered, the input combination (1,0) was not calculated separately since it provides the same results as the (0,1) combination. Also note that AND (NAND) logic provides the same results as OR (NOR) logic since both have the output cell initialized to '1' ('0') for both operations. The nonlinear I-V characteristics of the ReRAM cells require that the resistances of the devices and the voltages across them be found self-consistently. We used an iterative process to find these solutions.

Results of the V_{IN} and V_{OUT} calculations for AND/OR and NAND/NOR logic are shown in Figs. 3(a-d). These figures show V_{IN} and V_{OUT} calculations for all three input combinations. Figs. 3(a-b) illustrate cases where the ReRAM devices achieve 95 % and 5 % set/reset probability, which was determined using the cumulative probability distribution data (recall Figs. 2(a) and 2(b)) and Figs. 3(c-d) illustrate cases where the ReRAM devices achieve 85 % and 15 % set/reset probability. The examples shown in Fig. 3(a-d) present the data from sample set 2, but the same calculations were obtained for each sample set. Note that these calculations were also repeated over all current compliances; however, current compliances below 1 mA did not show any logic voltage where any Boolean logic operation could be carried out. Therefore, for our analysis, we focused on data from 1 mA compliances.

These data were used to verify whether Boolean logic operations can be performed by determining a logic voltage window for a desired output accuracy. For AND/NAND logic, this window was defined as the difference between logic voltages where the (1,1) output line crosses the 5 % set/reset lines and where the (0,1) output line crosses the 95 % set/reset lines (assuming 95% accuracy is desired). Fig. 3(a) shows that sample set 2 has a window that is approximately 180 mV wide (between 1.78 V and 1.55 V) for AND logic at 95% accuracy and Fig. 3(c) shows that this window increases to approximately 400 mV when the accuracy is reduced to 85%. This illustrates that as the desired accuracy is increased, the difference between maximum and minimum switching voltages increases, thus decreasing the window of logic voltages. Figs. 3(b) and 3(d) show that no window for NAND logic exists for either 95 % or 85 % accuracy. For OR/NOR logic, the logic voltage window is defined as the difference in logic voltages where the (0,1) output line crosses the 5 % set/reset lines and where the (0,0) output line crosses the 95 % set/reset lines (assuming 95 % accuracy). Figs. 3(a) and 3(c) show that this region does not exist for OR logic. This is due to the fact that the high ON/OFF ratio of the ReRAM devices meant that the differences in R_{IN} between the (0,0) and (0,1) cases were small, meaning that the V_{IN} and V_{OUT} curves between these cases were very similar, thus resulting in no logic voltage windows for OR logic. Fig. 3(b) shows that no window exists for NOR logic at 95%, however, a small window appears at 85%.

It should be noted that there is a small risk of the logic voltage switching the states of the input cells. For AND logic, the V_{IN} values for the (0,0) and (0,1) combinations are very small compared to the 5 % set/reset line. As a result, the probability that the logic voltage will cause the input cells to switch states is small. The (1,1) combination does cross the 5 % set/reset lines, however the (1,1) input line crosses the 5 % set line at logic voltages significantly larger than the logic voltages in the operating window for AND logic. For NAND logic, the (0,1) and (1,1) input lines cross the 5 % reset line, but for the (0,1) case, this occurs at logic voltages much higher than the logic voltages for NOR and NAND logic. For the (1,1) case, the polarity of the logic voltage is set to facilitate '0' to '1' switching, so there is no risk for either of the inputs switching back to '0'. In the next section, the probability that the logic voltages cause the inputs to switch will be quantified.

B. Logic accuracy

Accuracy of AND, NAND, and NOR logic operations were calculated for all sample sets using Monte Carlo simulations. These calculations were carried out over a range of logic voltages and repeated under ideal conditions (no resistance variations) and with resistance variations. The accuracy was determined using the voltage and resistance calculations carried out in section III-A, which were mapped onto the cumulative probability distributions in Figs. 2(a) and (b), where the set probabilities were used for AND logic and reset probabilities were used for NAND/NOR logic. The cumulative probability distribution data and random number generators were used in our simulation to determine if the output cell switched under a given logic voltage and input combination. This process was repeated 10,000 times to determine the probability of switching the output cell, which was used to determine the success

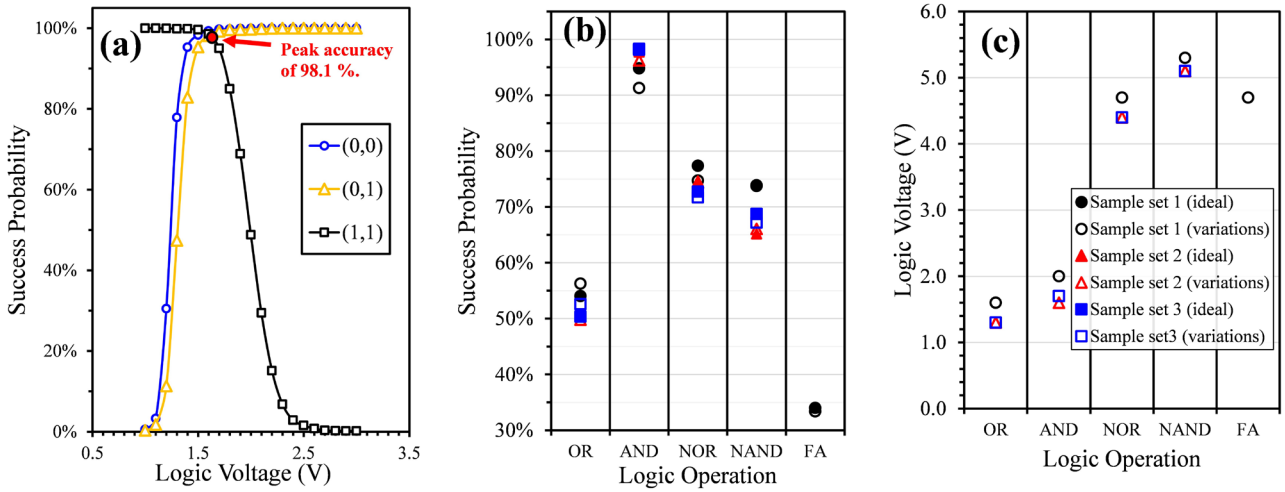

Fig. 4. (a) Output accuracy of AND logic for each input combination for sample set 3 with respect to logic voltage showing (with resistance variations) showing a peak accuracy of 98.1 % over 10,000 trials (based on minimum accuracy of all input combinations) at a logic voltage of 1.7 V. **(b)** Peak accuracy for AND, OR, NAND, and NOR logic for all sample sets with and without resistance variations plus output accuracy of NOR based full adder (FA) using results from sample set 1. Success probabilities were measured over 10,000 trials. **(c)** Logic voltages for each sample set and for each logic operation. Note that for each sample set, the logic voltages were the same with and without resistance variations.

probability. The success probability depends on the logic operations and the input combination. For example, using AND logic, the output should switch states for input combinations (0,0) and (0,1) but not for input (1,1). Therefore, the success probability increases with logic voltage for the (0,0) and (0,1) cases but decreases with logic voltage for the (1,1) cases. The overall success probability is maximized at logic voltages where the success probability curves for all three input combinations are maximized.

For calculations using 'ideal' conditions, only the cumulative probability distributions are considered, so no additional steps are required. It should be noted that the 'ideal' conditions in our calculations still account for device and cycle variations in set and reset voltages, since these variations are built into the cumulative probability distributions. When considering resistance variations, the same process is used, however, the resistance of each cell varies randomly for every iteration. At the beginning of each iteration, the resistance of a cell is estimated using a normally distributed random variate where the mean and standard deviation are drawn from the measured statistics [i.e. Fig. 2(c)]. The standard deviation is the sum of the quadrature of the standard deviations for device and cycle variations. This process is repeated 10,000 times to determine the accuracy of logic operations with resistance variations. Based on the calculations for the voltages across the input cells, the probability that the logic voltage causes the input cells to switch should be relatively small (<1 %). In our simulations, we determined the switching probability of the input cells as well as the output cell. If the logic voltage causes an input cell to switch, then the trial is considered a failure, regardless of the state of the output cell.

Figure 4(a) illustrates how the peak accuracy is determined in our calculations. For each sample set, logic operation, and variation condition (ideal or with resistance variations), the success probability for each input combination is plotted with logic voltage, as illustrated in Fig. 4(a). Ideally, there should be a range of logic voltages where all three input combinations are near 100 %, which is the operating range for that logic operation. The example shown in Fig. 4(a) shows the calculations for the AND operation in sample set 3 with resistance variations. These data show that AND logic for this sample set has a peak accuracy of 98.1 % over 10,000 trials at a logic voltage of 1.7 V. Note that the accuracy at each logic voltage is defined using the input combination with the lowest accuracy. The peak accuracy for each logic operation and each sample set is shown in Fig. 4(b), which shows calculations for both ideal devices and devices with resistance variations. For AND logic, all sample sets had peak accuracies above 95 % over 10,000 trials under ideal conditions, but the accuracy for sample set 1 dropped to 91.3 % when resistance variations were introduced. OR logic showed the lowest accuracy of all the logic operations, which is expected given that no range of logic voltages for OR logic was found in Figs. 3(a) and 3(c). Unfortunately, NAND and NOR logic operations showed much worse accuracy than that of AND logic. NAND logic showed the lowest peak accuracies, where all sample sets were below 80 % over 10,000 trials and even below 70 % for sample sets 1 and 2.. Accuracy of NOR logic was slightly improved over NAND logic, but accuracies for all sample sets were still below 80 % over 10,000 trials.

Note that unintentional switching of the input cells is not the reason for the low accuracies observed for OR/NOR logic since the inputs did not switch in any of the 10,000 trials for these logic operations under the logic voltages that produced peak accuracy (see Fig. 4(c)). This was also true for the AND logic operation. This is not true for NAND logic, where the switching probabilities of the input cells over 10,000 trials were 0 %, 11.5 %, and 6.5 % under ideal conditions, and to 0.5 %, 12.6 %, and 8.8 % with resistance variations, for samples sets 1, 2, and 3, respectively. Sample set 1 had lower input-cell switching probability than sample sets 2 and 3 because the cumulative

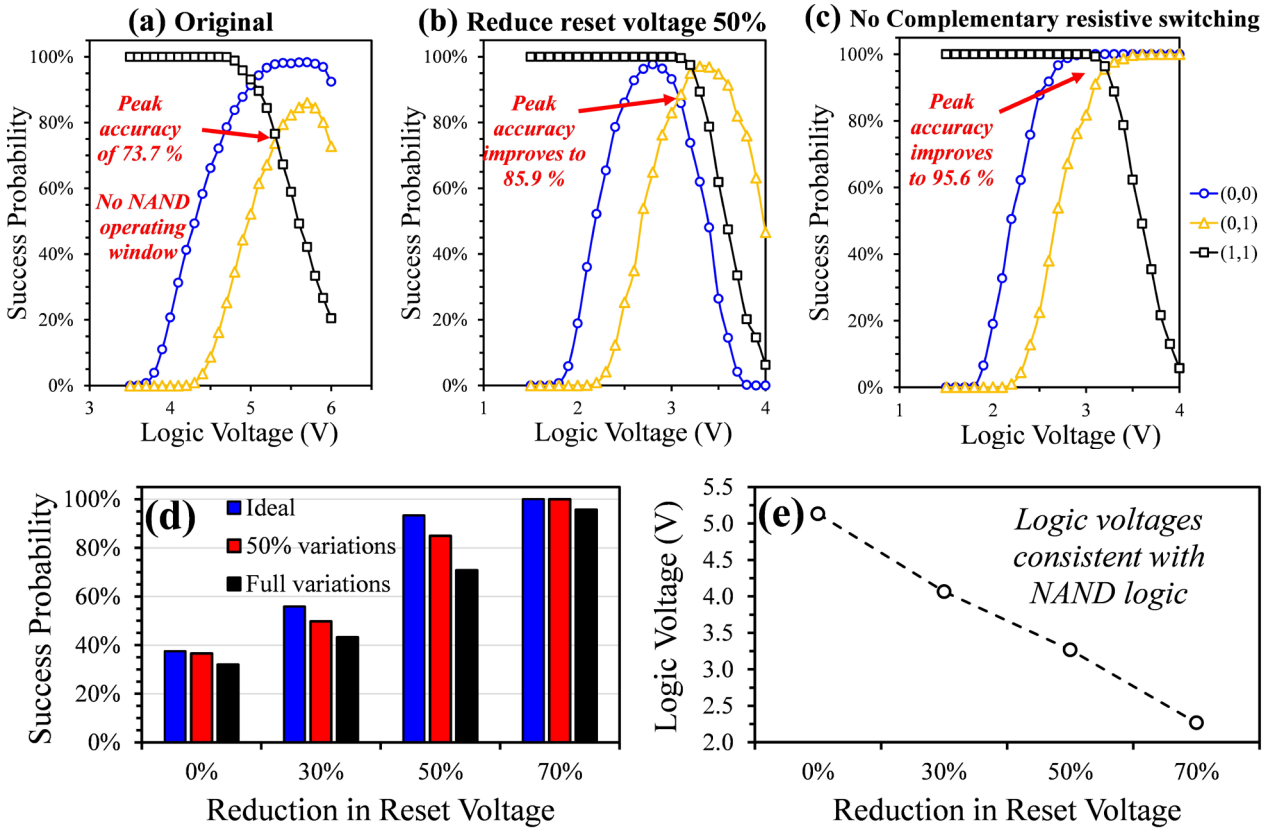

Fig. 5. (a-c) Output accuracy for NAND logic for each input combination for sample set 1 with respect to logic voltage showing (with resistance variations) for **(a)** original data, **(b)** reducing the reset voltages by 50 %, and **(c)** reducing reset voltages by 50 % and eliminating the influence of complementary resistive switching. **(d)** Peak accuracy for a NAND/NOR based full adder (using original data from sample set 1) and **(e)** logic voltages that produce peak accuracy for the NAND/NOR based full adder. Results in **(d-e)** are shown with respect to percent decrease in the reset voltage. The logic voltages shown in **(e)** are consistent with NAND logic.

probability distribution curve for the reset operation is sharper for sample set 1 than for samples sets 2 and 3. Since sample set 1 had the lowest probability of inadvertent switching of the input cells, we decided to use this data set to test the accuracy of a NOR/NAND based full-adder, which in CRAM requires 9 NAND/NOR operations and 12 cells. Note that the circuit for a NAND-based full adder and a NOR-based full adder has the same configuration, so in CRAM, the only difference between a NAND-based and a NOR-based full adder is the logic voltage applied. Based on the logic voltage that produced peak accuracy shown in Fig. 4(c), the NOR-based full adder produced better results than the NAND-based full adder. Fig. 4(b) shows that the accuracy of the full adder is only 34.0 % and 33.4 % for ideal devices and devices with resistance variations, respectively.

The data in Fig. 4(b) show that ReRAM-based CRAM has much better performance for AND logic than for NAND/NOR logic. The reason for this discrepancy will be discussed in section III-C, but from an applications perspective, this presents a significant problem for ReRAM-based CRAM. Any logic operation can be constructed using NAND logic gates, whereas AND logic is not universal. Therefore, for ReRAM-based CRAM to become feasible, the performance of NAND logic needs to be improved drastically. In the next section, we will discuss how accuracies for NAND logic in ReRAM-based CRAM can be improved to ≈100 % over 10,000 trials.

C. Solutions for improving accuracy

We have attributed the low accuracies for NAND/NOR logic operations shown in Fig. 4(b) to three main factors. These are: the large voltages required for NAND/NOR logic compared to AND logic, the effect of complementary resistive switching, and the magnitude of resistance variations between devices and between consecutive set/reset cycles.

Accuracies as low as 63 % were also observed in MRAM-based CRAM in [15], and these were attributed to the low ON/OFF ratio of MTJs, which typically have an ON/OFF ratio of ≈2. A higher ON/OFF ratio is desired for CRAM because it increases the difference in V_{OUT} between the (0,1) and (1,1) input cases. If the difference in V_{OUT} between these two input cases were small, then the accuracy of NAND logic would suffer since there would always be a small probability of undesired switching of the output cell for the (1,1) input case as well as a small probability that the output cell fails to switch for the (0,1) case, assuming that the switching probability of the output cell follows the curves in Figs. 2(a) and 2(b). It would seem to follow that the accuracy in CRAM could be improved by replacing MTJs with ReRAM devices, which have a much higher

ON/OFF ratio. However, our results reveal that ReRAM does not necessarily improve the performance of logic operations in CRAM primarily due to the decrease in the ON/OFF ratio with voltage (recall Fig. 2(c)). For ReRAM-based CRAM, logic operations that require a larger voltage generally have worse accuracies. Fig. 4(c) shows the logic voltage at the peak accuracy for each sample set and each logic operation. Note that resistance variations did not change the logic voltage where the accuracy reached a peak value. From this plot, we can observe that AND logic required the lowest voltages (1.2 V to 1.5 V) and NAND logic required voltages between 5 V and 5.5 V. This is because AND logic relies on switching between high-to-low resistance states (set operation) whereas NAND/NOR logic relies on low-to-high resistance switching (reset operation). From Figs. 2(a-b), we can see that, for these sample sets, the set operation requires lower voltages than the reset operation.

Based on these observations, it is reasonable to assume that the accuracy of the NAND operation would improve if the voltage required for the reset operation decreased. Fig. 5(a) shows the accuracy for the NAND operation for sample set 1 with resistance variations. As with the process illustrated in Fig. 4(a), we can determine that a peak accuracy of 73.7 % can be achieved at a logic voltage of 5.3 V. Fig. 5(b) illustrates how these results change when the reset voltages are reduced by 50 %. The logic voltage at the peak accuracy reduces to 3.1 V and the peak accuracy improves to 85.9 % over 10,000 trials. While this is a significant improvement, a key factor that is preventing the accuracy from improving any further is the small overlap between the (0,0) and (0,1) curves. This can be explained by the influence of complementary resistive switching, which occurs when a reset voltage is large enough to contribute to conductive filament formation rather than conductive filament dissolution. This explains why the reset probability peaks at around -3.5 V, then decreases as the reset voltage increases (recall Fig. 2(b)). Experimental work by [19] shows that complementary resistive switching can be avoided in Ta/TaO_X ReRAM devices through manipulation of the top electrode thickness. In our simulations, the effects of complementary resistive switching is removed by changing the cumulative probability distribution for the reset operation (recall Fig. 2(b)) so that the reset probability does not decrease for reset voltages above 3.5 V. Fig. 5(c) shows that the peak accuracy for NAND operation can be improved to 95.6 % over 10,000 trials when the reset voltages are reduced by 50 % and the effects of complementary resistive switching are removed.

The success probabilities of a NAND/NOR-based full adder from sample set 1 at 0 %, 30 %, 50 %, and 70 % decrease in the reset voltage are shown in Fig. 5(d). This bar graph shows measurements for ideal devices, devices with resistance variations, and devices with resistance variations reduced by 50 %. The case where variations are reduced represents a scenario where ReRAM devices are tested prior to their implementation in CRAM so that ReRAM devices with similar resistances can be selected. In this case only cycle-to-cycle variations are considered. The line plot in Fig. 5(e) shows the logic voltages required to achieve peak accuracy at each reduction in reset voltage. It should be noted that prior to reducing the reset voltage and removing the effects of complementary resistive switching, the logic voltage that achieved peak accuracy was consistent with NOR logic. However, once complementary resistive switching is removed, the peak accuracy is achieved using NAND logic. This is because decreasing the reset voltage and removing complementary resistive switching does not improve the accuracy for NOR logic but it does improve the accuracy for NAND logic.

Most importantly, the results in Fig. 5(d) show that 100 % success probability for a NAND-based full adder can be achieved over 10,000 trials for ideal devices and devices with 50 % resistance variations when the reset voltage is reduced by 70 % or higher. It should be noted that a 70 % reduction in reset voltage corresponds to a reset voltage of ≈ 900 mV to 1 V for our sample sets. This can be achieved in Ta/TaO_X-based ReRAM [20]. Our results show that the ReRAM device we fabricated will result in high error rates when used to perform Boolean logic operations in CRAM. However, a deeper analysis of our calculations reveal that ReRAM is a promising hardware solution in CRAM for devices with a sufficiently low reset voltage as well as no complementary resistive switching. Future studies on ReRAM based CRAM should investigate Ta/TaO_X based devices since they possess the desired characteristics and also have endurance of 10^{10} switching cycles [21].

IV. Conclusion

CRAM is a promising in-memory computing method as it allows for several cascading Boolean logic operations to be performed directly within the memory array. Previous hardware demonstrations that implement a full adder in MTJ-based CRAM have output accuracies as low as 63 %, which can be attributed to the low ON/OFF ratios of MTJs. To address this issue, we explored a solution where MTJ devices are replaced by ReRAM devices in CRAM, which have a much higher ON/OFF ratio. Our calculations incorporate our experimental results on three samples sets from TiN/TiO_X-based ReRAM to determine their performance in CRAM. Our results reveal that ReRAM-based CRAM is subject to even more inaccuracies than MTJ-based CRAM and that there are factors other than ON/OFF ratio that contribute to these errors. One of these factors is the decrease in the resistance with voltage, which causes the ON/OFF ratio to drop below 2 during the logic operations. Another factor is the effect of complementary resistive switching, which makes it impossible for the peak accuracies across all input combinations to converge to a single range of useful logic voltages. However, further analysis of our results reveals solutions that can attain 100 % accuracy over 10,000 trials for ReRAM-based CRAM by using devices with reduced reset voltages no complementary resistive switching. Based on previous studies, Ta/TaO_X ReRAM satisfies both of these requirements, thus being a viable option for future studies on ReRAM-based CRAM.

Acknowledgment

We thank Brian D. Hoskins for his guidance in ReRAM device fabrication. We also thank Matthew W. Daniels, Mark D. Stiles, and Deyuan Lyu for their careful reading of the manuscript. This work was funded by the National Institute of Standards and Technology. A.M. acknowledges support under the

Cooperative Research Agreement Award No. 70NANB14H209 through the University of Maryland.

REFERENCES

[1] G. Kestor, R. Gioiosa, D. J. Kerbyson, and A. Hoisie, "Quantifying the energy cost of data movement in scientific applications", *IEEE Int. Symp. Workload Characterization (IISWC)*, Portland, OR, pp. 56-65, Sept. 2013.

[2] D. Pandiyan and C.-J. Wu, "Quantifying the energy cost of data movement for emerging smart phone workloads on mobile platforms", *IEEE Int. Symp. Workload Characterization (IIWSCS)*, Raleigh, NC, pp. 171-180, Oct. 2014.

[3] A. Boroumand *et al*, "Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks", *ACM SIGPLAN Notices*, vol. 53, pp. 316 – 331, Mar. 2018.

[4] M. Gao, G. Ayers, and C. Kozyrakis, "Practical near-data processing for in-memory analytics frameworks", *Int. Conf. Parallel Architecture and Compilation (PACT)*, pp. 113 – 124, Oct. 2015.

[5] J. Huang *et al*, "Active-Routing: Compute on the Way for Near-Data Processing", *IEEE Int. Symp. High Performance Computer Architecture (HPCA)*, Washington DC, pp. 674 – 686, Feb. 2019.

[6] Z. Sun, S. Kvatinsky, X. Si, A. Mehonic, Y. Cali, and R. Huang, "A full spectrun of computing-in-memory technologies", *Nat. Electronics*, vol. 6, pp. 823 – 835, Nov. 2023.

[7] J. Borghetti, G. S. Snider, P. J. Kuekes, J. Joshua Yang, D. R. Stewart, and R. Stanley Williams, "'Memristive' switches enable 'stateful' logic operations via material implications", *Nature*, vol. 464, pp. 873 – 876, Apr. 2010.

[8] S. Kvatinsky *et al*, "MAGIC – memristor-aided logic", *IEEE Trans. Circuits Syst. II*, vol. 61, pp. 895 – 899, Sept. 2014.

[9] Z. Sun, E. Ambrosi, A. Bricalli, and D. Ielmini, "Logic Computing with Stateful Neural Networks of Resistive Switches", *Adv. Mater.*, vol. 30, p. 1802554, Aug. 2018.

[10] M. S. Q. Truong *et al*, "Adapting the RACER Architecture to Integrate Improved In-ReRAM Logic Primitives", *IEEE J. Emerging and Selected Topics in Ciruits and Syst.*, vol. 12, pp. 393 – 407, June 2022.

[11] M. Zabihi *et al*, "In-memory processing on the spintronic CRAM: form hardware design to application mapping", *IEEE Trans. Comput.*, vol. 68, pp. 1159 – 1173, Aug. 2019.

[12] Z. Chowdhury *et al*, "Efficient In-Memory Processing Using Spintronics", *IEEE Computer Architecture Lett.*, vol. 17, pp. 42- 46, Jan.-June 2018.

[13] M. Zabihi *et al*, "Using Spin-Hall MTJs to Build an Energy-Efficient In-memory Computation Platform", *20th Int. Symp. Quality Electronic Design (ISQED)*, Santa Clara, CA, pp. 52 – 57, Apr. 2019.

[14] H. Cilasun *et al*, "Spiking Neural Networks in Spintronic Computational RAM", *ACM Trans. Architecture and Code Optimization.*, vol. 18, Article 59, Sept. 2021.

[15] Y. Lv *et al*, "Experimental demonstration of magnetic tunnel junction-based computational random-access memory", *npj Unconventional Computing*, vol. 1, Article 3, July 2024.

[16] Hsu C.-W. *et al*. "Self-rectifying bipolar TaO_X/TiO_2 RRAM with superior endurance over 10^{12} cycles for 3D high-density storage-class memory", *Symp. VLSI Technol.*, Kyoto, Japan, T166 – T167, Aug. 2013.

[17] B. R. Zink, W. A. Borders, A. Madhavan, B. D. Hoskins, and J. McClelland, "Investigation of key performance metrics in TiO_X/TiN based resistive random-access memory cells", *In preparation.*.

[18] Y. Yang, P. Sheridan, and W. Lu "Complementary resistive switching in tantalum oxide-based resistive memory devices", *Appl. Phys. Lett.*, vol. 100, p. 203211, May 2012.

[19] A. Schonhals *et al*, "Critical ReRAM Stack Parameters Controlling Complimentary versus Bipolar Resistive Switching", *IEEE Int. Memory Workshop (IMW)*, Monterery, CA, pp. 1-4, July 2015.

[20] A. Prakash, J. Park, J. Song, J. Woo, E.-J. Cha, and H. Hwang, "Demonstration of Low Power 3-bit Multilevel Cell Characteristics in TaOx-Based RRAM by Stack Engineering", *IEEE Electron Device Lett.*, vol. 36, pp. 32 – 34, Jan. 2015.

[21] J. Joshua Yang *et al*, "High switching endurance in TaOX memristive devices", *Appl. Phys. Lett.*, vol. 97, p. 232102, Dec. 2010.