Open access • Journal Article • DOI:10.1109/TNANO.2014.2299558

Memristor Multiport Readout: A Closed-Form Solution for Sneak Paths

Mohammed A. Zidan, Ahmed M. Eltawil, Fadi J. Kurdahi, Hossam A. H. Fahmy ...+1 more authors

Institutions: King Abdullah University of Science and Technology, University of California, Irvine, Cairo University

Published on: 01 Mar 2014 - IEEE Transactions on Nanotechnology (IEEE)

Topics: Memristor, Memistor and Reading (computer)

Memristor Multiport Readout: A Closed-Form Solution for Sneak Paths

Mohammed Affan Zidan, *Student Member, IEEE*, Ahmed M. Eltawil, *Member, IEEE*, Fadi Kurdahi, *Fellow, IEEE*, Hossam A. H. Fahmy, *Senior Member, IEEE*, and Khaled N. Salama, *Senior Member, IEEE*

***Abstract*—In this paper, we introduce for the first time, a closed-form solution for the memristor-based memory sneak paths without using any gating elements. The introduced technique fully eliminates the effect of sneak paths by reading the stored data using multiple access points and evaluating a simple addition/subtraction on the different readings. The new method requires fewer reading steps compared to previously reported techniques, and has a very small impact on the memory density. To verify the underlying theory, the proposed system is simulated using Synopsys HSPICE showing the ability to achieve a 100% sneak-path error-free memory. In addition, the effect of quantization bits on the system performance is studied.**

***Index Terms*—Memory, memristor, sneak paths.**

I. Introduction

THE memristor (memory resistor) is a nonlinear passive device which changes its state according to the net charge passing through its two terminals, and maintains its state after the electrical bias is removed [1]. The memristor concept was initially introduced in the 1970s [2], [3], and it was not until 2008 when a fabricated device was related to the theory [4]. Several implementations have been reported recently [5]–[8]. Memristors find applications in memory arrays [9]–[12], programmable analog circuits [13], [14], logic and arithmetic circuits [15]–[22], modeling and emulation of synaptic activity [23]–[27], and electronic oscillators [28]–[31]. In addition, several models for memristor devices have been introduced [32]–[39].

Recently, the memristor has been explored as a future replacement for the current CMOS-based memories and solid state drives [40]–[42]. A memristor memory array comprises a crossbar structure, in which the memory cell is located at the intersection of two bars. Data are stored in the form of a high or low resistance by setting the memristor device to ON or OFF. One of the main challenges hindering the realization of memristor-based memories, from a circuit and architecture point of view, is the sneak-paths phenomenon [43]. A sneak path is an undesired current that flows through the memory cells parallel to the selected one, thus significantly impacting the read operation. These paths act as an unknown parallel resistance to the desired cell. Further complications arise because the sneak-current value depends on the memory contents.

Gating memristor cells using a transistor can be considered as a solution for sneak paths, however it has its own problems. The array density will be limited by the size of the gating transistors rather than the small memristors. Therefore, in order not to reduce the memristor array density considerably, the smallest possible transistor should be used. However, recently introduced small transistors are very leaky. Hence, the gating technique reduces the magnitude of the sneak-paths problem, but does not eliminate it completely. On the other hand, using larger transistors with lower leakage will reduce the memory density significantly. Moreover, fabricating high-density memristor–transistor arrays does not look very promising yet [44]. Another well-known cell gating strategy is using diodes, however such method is not suitable for the bipolar memristor devices [43].

Several other techniques were proposed in an attempt to address the sneak-paths problem without the use of gating elements [44], [45]. The viable gate-less strategies try to estimate the stored data in the memory in the presence of the sneak paths rather than eliminating it. The main advantages of such methods are keeping the memristor high density untouched and retaining a simple fabrication process for the memristor array. However, these strategies reduce the severity of the problem to various degrees, but they do not provide a fully curative solution [43], or the proposed technique is composed of many reading and writing stages [44].

In this study, we introduce the first closed-form solution for the sneak paths based on a multipoint reading method. The proposed method eliminates the effect of all the sneak paths by evaluating a simple addition/subtraction operation on the different readings of the system. The multipoint technique is simulated using Synopsys HSPICE to verify the underlying theory. The simulations show the ability to achieve a 100% sneak-paths error free memristor-based memory using the proposed method. The simulations are made for the complete memristor array which captures the parasitic effects of the array crossbars. Furthermore, the required processing circuit area is estimated by synthesizing the digital part using Cadence RTL compiler and TSMC technology kit, while the analog area is reported from a state-of-the-art

Manuscript received February 22, 2013; revised November 20, 2013; accepted January 1, 2014. Date of publication January 10, 2014; date of current version March 6, 2014. The review of this paper was arranged by Associate Editor D. Strukov.

M. A. Zidan and K. N. Salama are with the Electrical Engineering Program, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia (e-mail: mohammed.zidan@kaust.edu.sa; khaled.salama@kaust.edu.sa).

A. M. Eltawil and F. Kurdahi are with the Department of Electrical Engineering and Computer Science, University of California, Irvine, Irvine, CA 92617 USA (e-mail: aeltawil@uci.edu; kurdahi@uci.edu).

H. A. H. Fahmy is with the Department of Electronics and Communication, Faculty of Engineering, Cairo University, Giza 12316, Egypt (e-mail: hfahmy@stanfordalumni.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNANO.2014.2299558

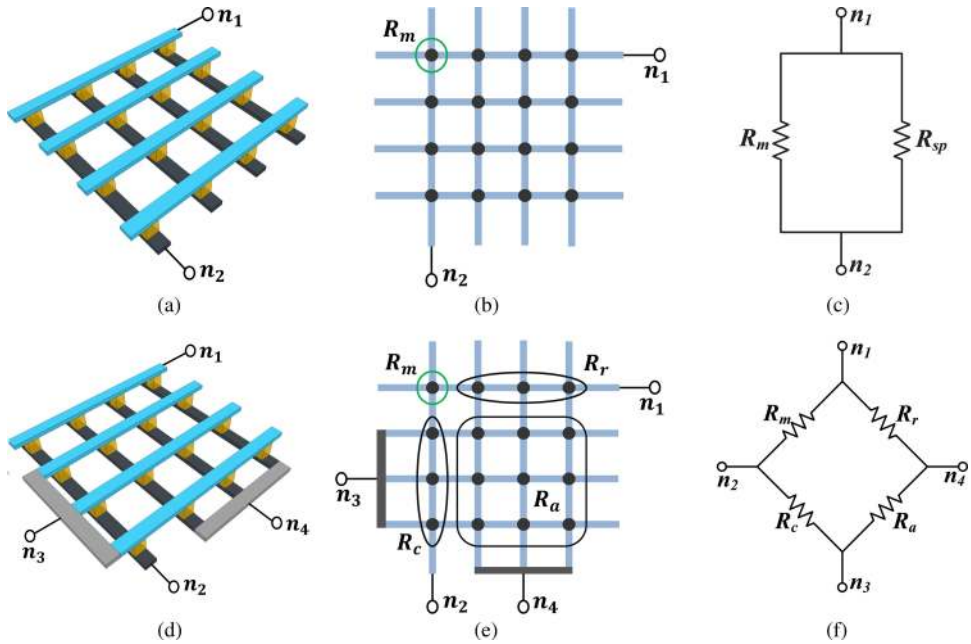

Fig. 1. Memristor arrays (regular and multipoint) and their equivalent circuits. (a) and (d) three-dimensional (3-D) illustration for the arrays where the memristor devices are represented as yellow boxes. (b) and (e) 2-D schematic for the arrays showing the lumped resistances where the blue lines represent the crossbars, the black dots represent memristor memory cells, and the green circle represents the cell of interest. (c) and (f) The equivalent circuits. It should be noted that the switching circuitry are not included in the figure for the sake of simplicity. (a) Regular Array (3D). (b) Regular Array (2D). (c) Regular Array Eqv. Cir. (d) Shorted Array (3D). (e) Shorted Array (2D). (f) Shorted Array Eqv. Cir.

work at the same technology node. The calculations show that the needed circuitry has a very small impact on memory.

The remainder of the paper is organized such that the proposed multipoint structure is introduced in the next section. In Section III, the mathematical closed-form solution for the sneak paths is presented. Then in Section IV, the crossbar resistance impact on the system is studied. Finally, in Section V, the memory simulations are presented and discussed.

II. Multipoint Structure

Reading the data stored in a memristor array is conventionally done in the form of resistive sensing between the selected row and column, as shown in Fig. 1(a) and (b). This resistive sensing can be achieved using various voltage- or current-based techniques. However, the equivalent circuit for such array consists of two parallel resistances, one for the desired cell (R_m) and another for the unknown sneak-paths resistance ($R_{\rm sp}$), as shown in Fig. 1(c). Therefore, trying to estimate the value of (R_m) is equivalent to solving for two independent unknowns in one equation, where a unique solution is not available. Moreover, the estimated value of the desired resistance based on a single reading is useless, since as the array size increases, the value of $R_{\rm sp}$ dominates the total resistance, thus significantly impacting the reliability of any estimation.

Better estimation of the stored data (R_m) can be achieved using multiple observations. The goal is to have more information for better estimations, or even to create a set of equations with independent unknowns ($R_{\rm sp}$ and R_m) that can be exactly solved. However, not all multiobservation strategies lead to an exact solution, since it is difficult to create an independent set of equations. Multiple observations for the array content can be made either in the time domain or the spatial domain. The time domain multiple observations is not practical for memory applications, because multiple memory writings have to occur between every two successive observations. However, intentional editing of the memory data may help. In [44], a multistage reading technique is introduced, where three readings, three writings, and one comparison operation are used for estimating the stored value in a memristor cell. Their proposed reading procedure is given as follows:

1) read from the target cell;
2) write "Zero" to the cell;
3) read again the target cell;
4) write "One" to the cell;
5) read the target cell for the third time;
6) compare the measured values to determine the state of the cell;
7) write back the memory cell to its (assumed) original state.

These successive readings and writings of the desired cell enables better estimation of the sneak current. It can also be interpreted as selecting an adaptive threshold for each reading. Currently, the multistage reading can be considered as the most successful gate-less reading technique in the literature.

On the other hand, spatial-based observations can be realized using a multipoint architecture. One of the simplest techniques to create a multipoint architecture is by shorting all the unselected columns together and all the unselected rows together. Hence, the memory array will have now four access points, as shown in Fig. 1(d) and (e). Those multiple points enable having multiple observations, and allow having more information to solve for the value of R_m. The main advantage of having shorted rows and columns is that the array can be mathematically

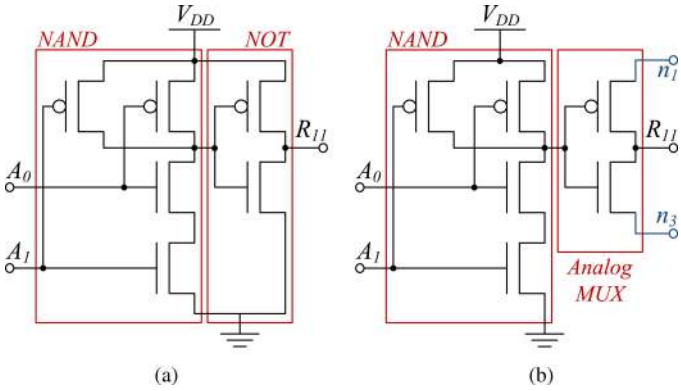

Fig. 2. 2:4 row decoder cell. Typically $V_{n1} = V_{\rm DD}$ and $V_{n3} \rightarrow {\rm GND}$. (a) Original. (b) Modified.

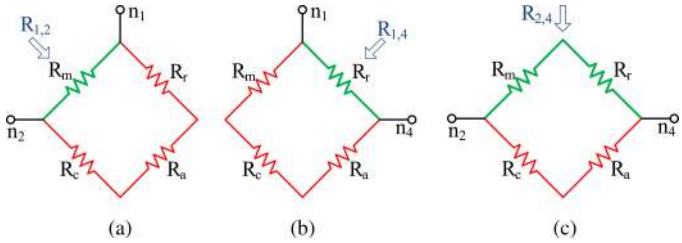

Fig. 3. Three-reading combination for evaluating the desired cell resistance. The green and the red colors represent the two parallel current paths. (a) Reading "1,2." (b) Reading "1,4." and (c) Reading "2,4."

modeled. Hence, an exact solution can be found. Fig. 1(f) shows the equivalent circuit of a shorted-terminals array. The circuit is made of four resistances, the desired cell resistance (R_m) and three components forming the sneak-path resistance,

$$R_{\rm sp} = R_r + R_a + R_c \tag{1}$$

where R_r is the lumped "*row*" resistance, R_c is the lumped "*column*" resistance, and R_a is the lumped "*array*" resistance, as shown in Fig. 1(e) and (f). Splitting $R_{\rm sp}$ into three lumped components enables having independent set of equations for solving R_m. Each of the three sneak-path components is composed of parallel resistances that are shorted together from both directions. For a square memory array of length (L) the total sneak-path resistance is given as

$$R_{\rm sp} = \left[\frac{O_r}{R_{\rm on}} + \frac{L - O_r - 1}{R_{\rm off}}\right]^{-1} + \left[\frac{O_c}{R_{\rm on}} + \frac{L - O_c - 1}{R_{\rm off}}\right]^{-1} + \left[\frac{O_a}{R_{\rm on}} + \frac{(L-1)^2 - O_a}{R_{\rm off}}\right]^{-1} \tag{2}$$

where O_r, O_c, and O_a are the number of ON resistances in the parallel arrays forming R_r, R_c, and R_a respectively. The ONE is represented by an ON resistance and the ZERO is represented by an OFF resistance. For a memory containing values other than $R_{\rm on}$ and $R_{\rm off}$, the formula given in (2) can be simply generalized using summation. The effect of the crossbar resistance on the proposed model is discussed in Section IV.

The shorting row and column can be fabricated as an extra row or column, or at a higher metal layer. For the first option, the decrease in density as a result of having extra row and column per array is given as

$$\triangle D = \frac{2L - 1}{L^2} \approx \frac{2}{L} \tag{3}$$

where $\triangle D$ is the decrease in a square array density and L is length of the array. $\triangle D$ equals to 0.195% for array size of 256 kb. For shorting the array terminals, the available row and column select circuits can be adopted. The unselected rows will be switched to a common bar, and so will the unselected columns, instead of being left floating. Fig. 2 shows the original and the modified 2:4 row decoder cell, where the inverter is transformed into an analog MUX. The same concept can be used for column selection.

III. Reading Technique

The proposed detection concept is based on using multiple readings to evaluate the desired cell resistance (R_m). By using the introduced circuit model, the different readings are represented as functions in the four circuit unknowns (R_m, R_r, R_a, and R_c). For solving the four unknowns, four different readings are required, where each reading is a "resistive sensing" between two of the four array nodes. The total number of possible readouts are six ($R_{1,2}, \ldots$), where the number donates the node number. However, we are interested only in calculating R_m, which we will call the main variable, and refer to the other three variables as auxiliary variables. Hence, solving for the main variable only requires fewer readings and equations. This can be achieved by lumping two of the auxiliary variables in all the equations as shown next, where three readings are sufficient. The three readings have to be selected such that they are represented in three independent equations, after lumping two of the auxiliary variables. Fig. 3 shows one of the possible three readings combinations for calculating R_m, where their equivalent resistances are as follows:

$$R_{1,2} = \frac{R_m R_c + R_m R_a + R_m R_r}{R_m + Rc + R_a + R_r} \tag{4a}$$

$$R_{1,4} = \frac{R_r R_m + R_r R_c + R_r R_a}{R_m + Rc + R_a + R_r} \tag{4b}$$

$$R_{2,4} = \frac{R_m R_c + R_m R_a + R_r R_c + R_r R_a}{R_m + Rc + R_a + R_r} \tag{4c}$$

where $R_{1,2}$, $R_{1,4}$, and $R_{2,4}$ are the sensed resistances of readings "1, 2," "1, 4," and "2, 4," respectively. The number of variables can be reduced into three only by lumping the two auxiliary variables, R_a and R_c into R_x.

A. Solving for General R_m

Without applying any constraints on the possible values of the unknowns, the three nonlinear equations (4a)–(4c) can be solved for R_m as

$$R_m = \frac{1}{2}\left(R_{1,2} - R_{1,4} - R_{2,4}\right) - \frac{2R_{1,4}R_{2,4}}{R_{1,2} - R_{1,4} - R_{2,4}}. \tag{5}$$

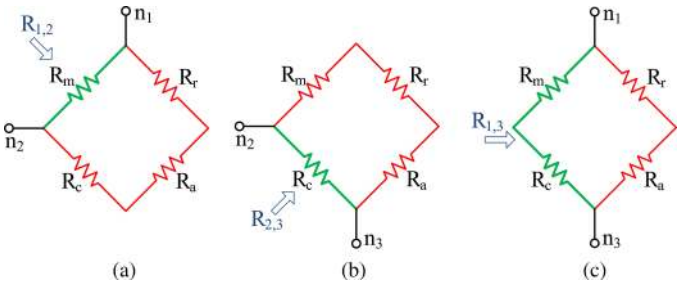

Fig. 4. Second three-reading combination for evaluating the desired cell. The green and the red colors represent the two parallel current paths. (a) Reading "1,2." (b) Reading "2,3." (c) Reading "1,3."

Alternatively, the desired cell resistance can be calculated using a different combination of readings, as shown in Fig. 4. Using the readings "1,2," "2,3," and "1,3," R_m can be solved as

$$R_m = \frac{1}{2}\left(R_{1,2} - R_{2,3} - R_{1,3}\right) - \frac{2R_{2,3}R_{1,3}}{R_{1,2} - R_{2,3} - R_{1,3}} \quad (6)$$

where $R_{1,2}$, $R_{2,3}$, and $R_{1,3}$ are resistances of readings "1,2," "2,3," and "1,3," respectively.

Equations (5) and (6) are general solutions for R_m without considering any extra information or boundary conditions that exist due to the actual circuit implementation and/or operation. In the following sections, we illustrate that a considerable reduction in complexity can be achieved by constraining the possible values of the unknowns using information regarding the operation of the system.

B. Solving for Binary R_m

For a binary memory system, each memristor cell is written as one of two saturated values representing zeroes and ones,

$$R_m = \{R_{\rm on}, R_{\rm off}\} \quad (7)$$

where $R_{\rm on}$ is the minimum device resistance and $R_{\rm off}$ is the maximum one. Therefore, the calculated value of (5) or (6) will be compared with a threshold to estimate the desired cell value. A more efficient technique is to do a first stage off-line thresholding based on the equation before calculating the value of R_m. For instance, solving (5) for $R_{1,2}$, such that,

$$R_{1,2} = R_{1,4} + R_{2,4} + R_m \pm \sqrt{4R_{1,4}R_{2,4} + R_m^2}. \quad (8)$$

By considering the binary nature of R_m, (8) has four possible solutions. However, not all the four solutions are valid for a resistive array system, since,

$$\{R_{1,2}, R_{1,4}, R_{2,4}, R_m\} \geq 0. \quad (9)$$

In addition, the first reading is always smaller than R_m,

$$R_{1,2} = R_m \parallel R_{\rm sp} < R_m. \quad (10)$$

Using the constraints given by (9) and (10), (8) will have two possible solutions only such that,

$$R_{1,2}\left(R_{\rm on}\right) = R_{1,4} + R_{2,4} + R_{\rm on} + \sqrt{4R_{1,4}R_{2,4} + R_{\rm on}^2}. \quad (11a)$$

$$R_{1,2}\left(R_{\rm off}\right) = R_{1,4} + R_{2,4} + R_{\rm off} - \sqrt{4R_{1,4}R_{2,4} + R_{\rm off}^2}. \quad (11b)$$

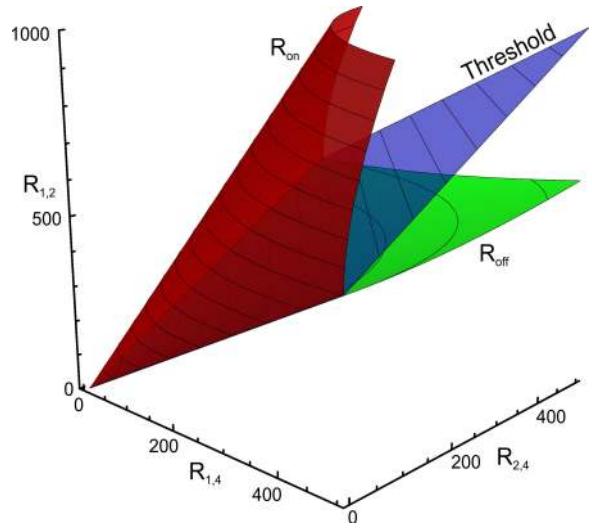

Fig. 5. Relation between the three readings $R_{1,2}$, $R_{1,4}$, and $R_{2,4}$ representing the cases of $R_{\rm on}$, $R_{\rm off}$, and the threshold plane between them described by the (11a), (11b), and (13).

By applying the different constraints, the possible relation between the three readings is restricted by (11a) and (11a), which can be represented by two surfaces in the 3-D space as shown in Fig. 5. Now, a threshold equation can be defined to be located between the two surfaces representing $R_{\rm on}$ and $R_{\rm off}$ cases. The threshold equation has to satisfy the boundary conditions between the two surfaces, which are defined as

$$R_{1,2} = R_{1,4}, \text{ at } R_{2,4} \to 0 \quad (12a)$$

$$R_{1,2} = R_{2,4}, \text{ at } R_{1,4} \to 0. \quad (12b)$$

It should be noted that the readings are distributed around the symmetry plane $R_{2,4} = R_{1,4}$, and the boundary conditions are only open interval limits. A threshold plane which satisfies (12a) and (12b) is defined as

$$R_{1,2} = R_{1,4} + R_{2,4}. \quad (13)$$

Fig. 5 shows how the selected threshold plane is located between the $R_{\rm on}$ and the $R_{\rm off}$ surfaces and satisfies their boundary conditions. The threshold plane can be realized using a very simple circuit as shown in Fig. 6. The figure shows that such circuit can be implemented using a three-operand adder/subtracter, where

$$R_t = R_{1,4} + R_{2,4} - R_{1,2} \quad (14)$$

where the output is detected as $R_{\rm on}$ for R_t greater than a constant threshold, and $R_{\rm off}$ otherwise. The output of the addition

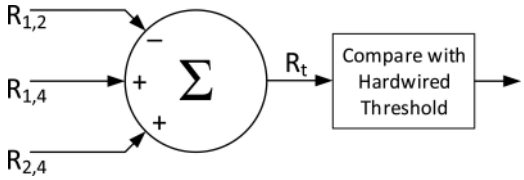

Fig. 6. Circuit realizing the threshold plan equation.

operation is then compared with a fixed threshold, which can be realized using a simple circuit. The circuit also can be implemented using sequential two-operand adder/subtracter, since the different readings are captured in series. The proposed multipoint reading procedure, shown in Fig. 3, is summarized as follows:

1) read the desired cell using the ports "1" and "2;"
2) read the desired cell using the ports "1" and "4;"
3) read the desired cell using the ports "2" and "4;"
4) take a decision based on calculating R_t.

Compared to the multistage reading technique introduced by HP Labs in [44], our proposed system requires three readings only, while the multistage reading requires three writings and three readings. Moreover, the proposed technique significantly reduces the write cycles of the memory and so increases its endurance lifetime compared to the multistage reading. This is an important property, knowing that device endurance is one of the main challenges facing memristor fabrication, where the reported high endurance devices have order of magnitude less endurance than the DRAM [7], [46]. Another advantage for the proposed technique is that its decision stage can be pipelined, while this is not true for the multistage reading; since in the multistage technique the final writing to the memory occurs after the decision stage. Furthermore, in [44], a wrong reading will corrupt the cell for all the next readings, since the wrong estimated value is written-back to the cell, while this is not true for the multipoint technique. It is worth noting that the fourth step in the multipoint readout scheme is equivalent to sixth step of the multistage reading [44] requiring similar complexity circuit. However, our technique uses a fixed threshold comparison as opposed to the variable threshold used in the multistage reading.

It should be noted here that the memristor devices fabricated for memory applications are engineered to have a writing threshold [44], and reading with a voltage below this threshold will not affect the stored data. Yet, for other threshold-less devices, a zero net-flux reading strategy can be used [47].

IV. Crossbar Effect

The resistance of the crossbar is unwanted parasitic component in the memory array, and in turn for the proposed model. However, it has a useful effect of damping the sneak-paths current. In this section, we study the effect of the crossbar resistance on the sneak current. Moreover, we include it in all our simulations, by simulating the complete memory array rather than its equivalent circuit. For calculating the crossbar resistance component between two adjacent cells ($R_{\rm cb}$), we assume a square memristor cell of dimensions of $u \times u$, the separation between two cells is u/2, and the metal bar is of thickness t. The resistance can be simply defined as

$$R_{\rm cb} = \rho \frac{3u}{2u \times t} = \frac{3\rho}{2t} \tag{15}$$

where ρ is the resistivity of the crossbar metal. Most of the memristor arrays uses platinum ($\rho = 105$ n·Ωm) as the crossbar metal with a thickness ranges from 10 to 20 nm [5], [48]. This leads to a relatively high crossbar resistance per cell ($R_{\rm cb} \approx 10\ \Omega$). However, arrays with similar dimensions can be built in the regular CMOS process using copper ($\rho = 16.78$ n·Ωm) with metal thickness of around 100 nm. This leads to a much smaller crossbar resistance per cell ($R_{\rm cb} < 0.3\ \Omega$). However, we considered the worst case of $R_{\rm cb} = 10\ \Omega$ in all our memory simulations. For studying the crossbar effect, we simulate memristor arrays versus array size using Synopsys HSPICE. To have a trend describing the $R_{\rm cb}$ effect, we used a checkered data pattern to represent equiprobable zeros and ones memory. A wider set of random data patterns is used in the system simulations in the next section. Since memristors fabricated for memory applications are characterized by having high ON and OFF resistances [5], [44], [46], [48], we used the values reported in [5] and [44], where $R_{\rm on} = 1$ MΩ and $R_{\rm off} = 1$GΩ.

The first undesirable effect of the crossbar resistance is reducing the effective OFF/ON ratio of the memristor device, since the parasitic resistance will act as dc value added to both $R_{\rm on}$ and $R_{\rm off}$. Fig. 7(a) shows the decrease in OFF/ON ratio versus the memristor array size, where $\Delta_{\rm OFF/ON} = (\rm Apparent_{OFF/ON} - Device_{OFF/ON})/Device_{OFF/ON}$. The maximum parasitic series resistance is considered in the case of $\rm Apparent_{OFF/ON}$, where we assume that two series resistances of two full crossbar lengths are added. Another effect of the crossbar resistance is damping the total current consumed by memristor array. Fig. 7(b) shows the total current consumed versus the array area for different $R_{\rm cb}$ values. $R_{\rm cb}$ adds a damping effect to the sneak current causing it to have a saturation behavior. For $R_{\rm cb} = 10\ \Omega$, the total current saturates below 110 μA. This damping effect is directly proportional to the crossbar resistance as shown in the figure. Such effect has positive and negative sides. The main advantage of the current saturation is reducing the power consumption by a memristor array; however it narrows the gap between the ON and OFF current. This can increase the complexity of the sensing circuit. The average steady sate power consumption of a 256-K array is around 91.3 μW for a voltage source of 1 V, during the reading and writing processes, and this value saturates below 110 μW for larger arrays. Finally, the effect of the ON resistance of the memristor device on the saturation current is presented in Fig. 7(c). The figure shows that the saturation current increases with smaller device resistance and that saturation starts to appear at smaller array sizes. In general, the different simulations presented in Fig. 7 show that the effect crossbar parasitic resistance, with its advantages and disadvantages, have to be considered in all the memristor memory simulations.

V. Memory Simulations and Discussions

In order to verify the proposed concept, the memristor memory system is simulated using Synopsys HSPICE, where the

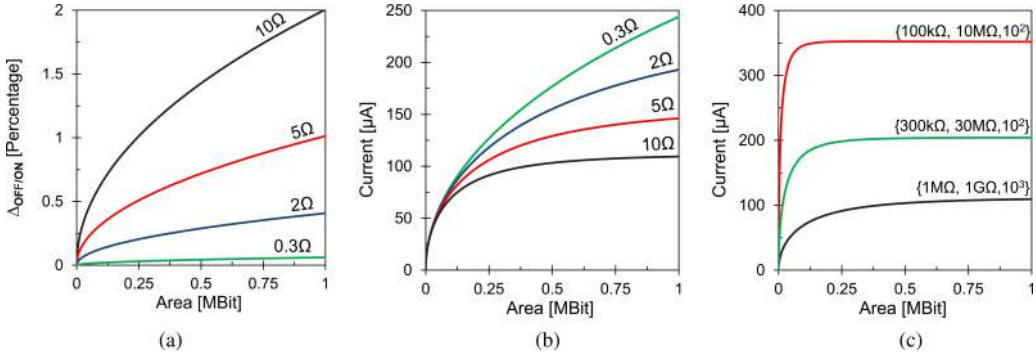

Fig. 7. (a) Decrease in OFF/ON ratio versus the memristor array size for various $R_{\rm cb}$, where $\Delta_{\rm OFF/ON} = ({\rm Apparent}_{\rm OFF/ON} - {\rm Device}_{\rm OFF/ON})/{\rm Device}_{\rm OFF/ON}$. (b) HSPICE simulation for the current consumed by multipoint array first reading versus the memristor array size for various $R_{\rm cb}$ at $R_{\rm ON} = 1$ MΩ, $R_{\rm OFF} = 1$ GΩ, and checkered data patterns, and (c) HSPICE simulation for the current consumed by multipoint array first reading versus the memristor array size for various $\{R_{\rm ON}, R_{\rm OFF}, {\rm OFF} - {\rm ON\ ratio}\}$ resistances at $R_{\rm cb} = 10\ \Omega$ and checkered data pattern. For all the simulations, a voltage source of 1 V is used.

whole array is simulated rather than its equivalent circuit. This enables including all the unmodeled or overlooked parasitic effects in our simulations. In addition, this allows capturing the crossbar resistance effect precisely, where the worst case of $R_{\rm cb} = 10\ \Omega$ was used in our simulations. In addition, we include the effect of the switches used to connect the array terminals to the shorting bars in our simulations. For estimating the switch resistance, we used 32-nm N-channel MOSFET (NMOS) devices provided by the predictive technology model [49]. According to HSPICE simulations using minimum size devices ($W = L = 32$ nm), the ON resistance of each device is less than 5 kΩ. However, we used a more conservative value of 10 kΩ in our memory simulations.

Including the different nonlinearities leads to a very long simulation time. A Python script was written to create the SPICE net-list and to do sweeps over area and different data patterns by calling HSPICE iteratively. All the simulations used 64 data patterns, one checkered, and 63 pseudorandom data. This is a total of more than 8 K simulation runs of the whole set of arrays. In general, binary coding used within the computer systems is characterized by having equal probability for each of its two symbols. Therefore, the random patterns used are set to have equiprobable chance of zeros and ones, to mimic realistic data distributions.

As with other emerging memory technologies (e.g., PCRAM [50]), the memristor hierarchy was assumed to be similar to DRAM. This hierarchy includes banks, blocks, and subblocks to increase the memory bandwidth and reduce the parasitic effects, such as the capacitive loading of long crossbars [51], [52]. Other techniques used to reduce the effective size of array, e.g., folding arrays and open digiline [52], can be borrowed from DRAM architectures. For DRAM memory, the largest continuous array is about 256 Kb and circuitry is used to connect arrays building larger memory blocks [51]. Coping with memory splitting strategies, we believe that simulating continuous memristor array up to the size of 1 Mb is sufficient, which we used in all the simulation presented in this study.

The HSPICE simulations are used to verify the proposed concept in this study. The main goal of the multipoint system is to eliminate the effect of the sneak paths. Those sneak paths cause

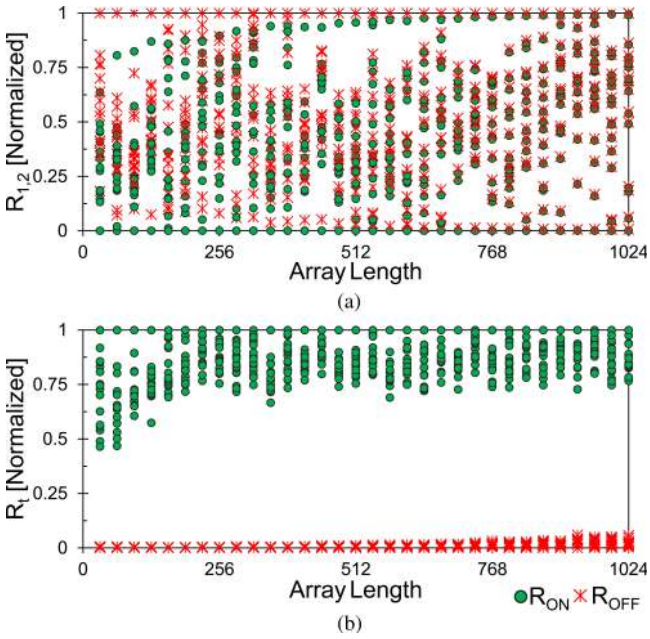

Fig. 8. HSPICE simulations showing the normalized memory readings in case of either "One" or "Zero" stored in the desired cell for different array lengths (L) for, (a) Reading between the nodes "1" and "2" ($R_{1,2}$), and (b) calculated R_t. The array size equals L^2. The normalization factors for $R_{1,2}$ are $\{18.1, 11.8, 9.9, 9.4\} \times 10^3$, for the array lengths of 256, 512, 768, and 1024, respectively.

zeros and ones reading to be mixed together, where no single threshold can be defined at the design stage. This effect is represented using HSPICE simulations in Fig. 8(a), where no single threshold can be distinguished between the different memory readings. The figure shows normalized values for memristor memory readings in case of either "One" or "Zero" stored in the desired cell for different array sizes. On the other hand, applying (14) changes the picture completely, where this equation is a single addition/subtraction for three memory readings. Fig. 8(b) shows the normalized value of R_t versus the array size, and how (14) correctly ordered all the readings into two distinguishable sets. A single fixed threshold can then be defined for the memory system. It should be noted that the reduced dataset is used in Fig. 8 for the sake of visualization.

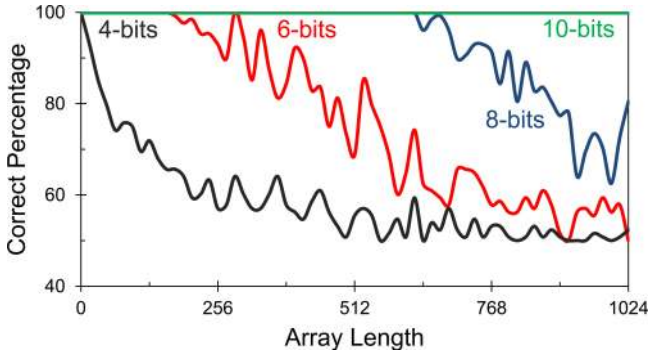

Fig. 9. Percentage of correctly read data versus the array size for different number of quantization bits.

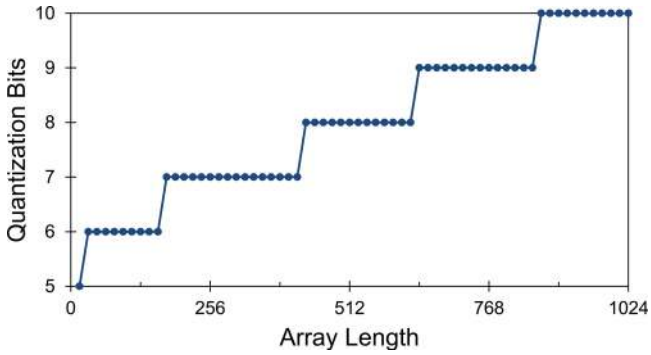

Fig. 10. Required number of bits for error-free output versus the array length (L), where the array size equals L^2.

A. Digital Decision Circuit

The proposed decision system can be simply implemented by quantizing three readings and applying the digital realization of (14) on the quantized values. This relatively small digital circuit can be simply added to the memory pipeline. The number of bits used for quantizing the sensed analog readings is one of the main design aspects of the multireading array, because it affects the percentage of correctly evaluated readings. The number of required bits for having a 100% correct output is directly proportional to the array size. As this size increases, the impact of the sneak-paths noise becomes more dominant. Therefore, more bits are required to correctly evaluate the stored data in larger arrays. In other words, as sneak-paths noise increases with the increase of the array size, we need to decrease the quantization noise for a correct detection. The simulations show that for a given array size, an error-free output can be achieved using a sufficient number of quantization bits, as shown in Fig. 9. Fig. 10 shows the required number of bits for an error-free output versus the array length, where the array size is the square of its length. It should be noted that analog-to-digital converters (ADCs) normally are designed to quantize the dynamic range of a signal after normalization [53]. The dc value should be selected at the design phase of the ADC. This helped in reducing the required number of bits.

B. Circuit Area Estimation

Fig. 10 shows that for an array size of 256 Kb, eight quantization bits are required, which are used in the circuit area estimation. This array size is selected since it is a common number used for a continuous array of memory as discussed previously [51]. The decision circuit is made of two main blocks, the ADC and the digital circuit. The two blocks have to work at a frequency enabling a fast reading operation. The International Technology Roadmap for Semiconductors (ITRS) report the current REDOX memory (including memristor) reading delay to be less than 50 ns [41]. To comply with such number, a 100 MS/s ADC would be sufficient. However, the current state-of-the-art of the ADC designs is at least four times faster speed than such number. In [54], a 65-nm 8-bit 400 MS/s successive approximation register (SAR) ADC is reported to have a total area of $2.4 \times 10^{-2}\text{mm}^2$. According to the empirical equation scaling equation proposed in [55], the estimated area for 100 MS/s version of the ADC is around $1.65 \times 10^{-2}\text{mm}^2$ in area. This is considered as 0.016% of a memory die area, compared to Micron's 78 nm 1 Gb DDR3 of 102 mm^2 die area [56]. This overhead area should be multiplied by the number of ADCs. The power consumption for the ADC is around 1.34 mW for 100 MS/s operation mode [54], [55]. It should be noted that one ADC is required per each column decoder, and a single ADC is active per each memory bank at a given time. In addition, the required ADC constraints are simpler than the one reported in [54]. And according to the empirical equation proposed in [55], using a more recent technology as 22 nm can scale down a SAR ADC area by a factor of 4.6×, where the new area should be around $3.59 \times 10^{-3}\text{mm}^2$.

The digital circuit was written in Verilog HDL and synthesized using Cadence RTL compiler and TSMC 65 nm standard cell libraries. fThe circuit is designed as a three-operand adder/subtractor combinational circuit, where the inputs are three 8-bit registers. The output of the adder/subtractor is then compared with a hardwired threshold as shown in Fig. 6. The estimated area for the synthesized circuit is $3.2 \times 10^{-4}\text{mm}^2$. This area is considered as 3.14×10^{-4}% of a memory chip area, compared to Micron's 78 nm 1 Gb DDR3 of 102-mm^2 die area [56], which is considered as negligible overhead. Even, smaller area can be reported if a more recent fabrication node is used. The digital circuit power consumption is in order of few microwatts at 100-MHz frequency and 65-nm node [57].

VI. Conclusion

Based on a multipoint reading structure, we introduced the first closed-form solution for the memristor memories sneak paths. The introduced technique is based on calculating the desired cell value using addition/subtraction of three different readings. The new method completely eliminates the effect of sneak paths on the reliability of the read values, without the need of any gating element. The proposed technique implies very small overhead on a memory area for achieving faster reading speed than currently reported by ITRS. In addition, the introduced method requires fewer steps and less impact on the memristor device endurance compared to other techniques reported in the literature. Finally, the proposed system is simulated using Synopsys HSPICE verifying the underlying theory and showing the ability of achieving a 100% sneak-paths error-free output.

ACKNOWLEDGMENT

The authors would like to thank Dr. A. Elshurafa, M. S. Khairy, and H. Omran for their reviews and valuable suggestions.

REFERENCES

[1] M. D. Ventra, Y. V. Pershin, and L. O. Chua, "Circuit elements with memory: Memristors, memcapacitors, and meminductors," *Proc. IEEE*, vol. 97, no. 10, pp. 1717–1724, Oct. 2009.

[2] L. Chua, "Memristor-the missing circuit element," *IEEE Trans. Circuit Theory*, vol. 18, no. 5, pp. 507–519, Sep. 1971.

[3] L. Chua and S. M. Kang, "Memristive devices and systems," *Proc. IEEE*, vol. 64, no. 2, pp. 209–223, Feb. 1976.

[4] D. B. Strukov, G. S. Snider, and D. R. Stewart, "The missing memristor found," *Nature*, vol. 435, pp. 80–83, 2008.

[5] Q. Xia, J. J. Yang, W. Wu, X. Li, and R. S. Williams, "Self-aligned memristor cross-point arrays fabricated with one nanoimprint lithography step," *Nano Lett*., vol. 10, no. 8, pp. 2909–2914, 2010.

[6] N. Gergel-Hackett, J. L. Tedesco, and C. A. Richter, "Memristors with flexible electronic applications," *Proc. IEEE*, vol. 100, no. 6, pp. 1971–1978, Jun. 2012.

[7] J. J. Yang, M.-X. Zhang, J. P. Strachan, F. Miao, M. D. Pickett, R. D. Kelley, G. Medeiros-Ribeiro, and R. S. Williams, "High switching endurance in taox memristive devices," *Appl. Phys. Lett*., vol. 97, no. 23, pp. 232102-1–232102-3, 2010.

[8] O. Kavehei, K. Cho, S. Lee, S.-J. Kim, S. Al-Sarawi, D. Abbott, and K. Eshraghian, "Fabrication and modeling of Ag/TiO2/ITO memristor," in *Proc. IEEE Int. Midwest Symp. Circuits Syst*., 2011, pp. 1–4.

[9] K. H. Jo, C. M. Jung, K. S. Min, and S. M. Kang, "Self-adaptive write circuit for low-power and variation-tolerant memristors," *IEEE Trans. Nanotechnol*., vol. 9, no. 6, pp. 675–678, Nov. 2010.

[10] K. Eshraghian, K. Cho, O. Kavehei, S. Kang, D. Abbott, and S. M. Kang, "Memristor MOS content addressable memory (MCAM): Hybrid architecture for future high performance search engines," *IEEE Trans. Very Large Scale Integr. Syst*., vol. 19, no. 8, pp. 1407–1417, Aug. 2011.

[11] I. E. Ebong and P. Mazumder, "Self-controlled writing and erasing in a memristor crossbar memory," *IEEE Trans. Nanotechnol*., vol. 10, no. 6, pp. 1454–1463, Nov. 2011.

[12] K.-H. Kim, S. Gaba, D. Wheeler, J. M. Cruz-Albrecht, T. Hussain, and N. Srinivasa, W. Lu, "A functional hybrid memristor crossbar-array/CMOS system for data storage and neuromorphic applications," *Nano Lett*., vol. 12, no. 1, pp. 389–395, 2011.

[13] S. Shin, K. Kim, and S.-M. Kang, "Memristor applications for programmable analog ICs," *IEEE Trans. Nanotechnol*., vol. 10, no. 2, pp. 266–274, Mar. 2011.

[14] Y. Pershin and M. Di Ventra, "Practical approach to programmable analog circuits with memristors," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 8, pp. 1857–1864, Aug. 2010.

[15] L. Gao, F. Alibart, and D. Strukov, "Programmable CMOS/memristor threshold logic," *IEEE Trans. Nanotechnol*., vol. 12, no. 2, pp. 115–119, Mar. 2013.

[16] W. Robinett, M. Pickett, J. Borghetti, Q. Xia, G. Snider, G. Medeiros-Ribeiro, and R. Williams, "A memristor-based nonvolatile latch circuit," *Nanotechnology*, vol. 21, p. 235203, 2010.

[17] C.-M. Jung, K.-H. Jo, E.-S. Lee, H. M. Vo, and K.-S. Min, "Zero-sleep-leakage flip-flop circuit with conditional-storing memristor retention latch," *IEEE Trans. Nanotechnol*., vol. 11, no. 2, pp. 360–366, Mar. 2012.

[18] F. Merrikh-Bayat and S. Shouraki, "Memristor-based circuits for performing basic arithmetic operations," *Procedia Comput. Sci*., vol. 3, pp. 128–132, 2011.

[19] I. Vourkas and G. Sirakoulis, "A novel design and modeling paradigm for memristor-based crossbar circuits," *IEEE Trans. Nanotechnol*., vol. 11, no. 6, pp. 1151–1159, Nov. 2012.

[20] R. Patel and E. G. Friedman, "Arithmetic encoding for memristive multi-bit storage," in *Proc. IEEE/IFIP Int. Conf. VLSI Syst.-on-Chip*, 2012, pp. 99–104.

[21] O. Kavehei, S. Al-Sarawi, K.-R. Cho, N. Iannella, S.-J. Kim, K. Eshraghian, and D. Abbott, "Memristor-based synaptic networks and logical operations using in-situ computing," in *Proc. Int. Conf. Intell. Sens., Sen. Netw. Inform. Process*., 2011, pp. 137–142.

[22] E. Gale, B. de Lacy Costello, and A. Adamatzky, "Boolean logic gates from a single memristor via low-level sequential logic," in *Unconventional Computation and Natural Computation*. New York, NY, USA: Springer, 2013, pp. 79–89.

[23] A. Subramaniam, K. Cantley, G. Bersuker, D. Gilmer, and E. Vogel, "Spike-timing-dependent plasticity using biologically realistic action potentials and low-temperature materials," *IEEE Trans. Nanotechnol*., vol. 12, no. 3, pp. 450–459, May 2013.

[24] S. Jo, T. Chang, I. Ebong, B. Bhadviya, P. Mazumder, and W. Lu, "Nanoscale memristor device as synapse in neuromorphic systems," *Nano Lett*., vol. 10, no. 4, pp. 1297–1301, 2010.

[25] T. Chang, S.-H. Jo, and W. Lu, "Short-term memory to long-term memory transition in a nanoscale memristor," *ACS Nano*, vol. 5, no. 9, pp. 7669–7676, 2011.

[26] M. Soltiz, D. Kudithipudi, C. Merkel, G. Rose, and R. Pino, "Memristor-based neural logic blocks for non-linearly separable functions," *IEEE Trans. Comput*., vol. 62, no. 8, pp. 1597–1606, Aug. 2013.

[27] G. Indiveri, B. Linares-Barranco, R. Legenstein, G. Deligeorgis, and T. Prodromakis, "Integration of nanoscale memristor synapses in neuromorphic computing architectures," *Nanotechnology*, vol. 24, no. 38, p. 384010, 2013.

[28] M. A. Zidan, H. Omran, C. Smith, A. Syed, A. G. Radwan, and K. N. Salama, "A family of memristor-based reactance-less oscillators," *Int. J. Circuit Theory Appl*., DOI: 10.1002/cta.1908.

[29] A. Talukdar, A. G. Radwan, and K. N. Salama, "Non linear dynamics of memristor based 3rd order oscillatory system," *Microelectron. J*., vol. 43, no. 3, pp. 169–175, 2012.

[30] M. Itoh and L. Chua, "Memristor oscillators," *Int. J. Bifurcation Chaos*, vol. 18, pp. 3183–3206, 2008.

[31] A. Talukdar, A. G. Radwan, and K. N. Salama, "Generalized model for memristor-based wien family oscillators," *Microelectron. J*., vol. 42, no. 9, pp. 1032–1038, 2011.

[32] D. Batas and H. Fiedler, "A memristor SPICE implementation and a new approach for magnetic flux-controlled memristor modeling," *IEEE Trans. Nanotechnol*., vol. 10, no. 2, pp. 250–255, Mar. 2011.

[33] Z. Biolek, D. Biolek, and V. Biolkova, "Spice model of memristor with nonlinear dopant drift," *Radioengineering*, vol. 18, no. 2, pp. 210–214, 2009.

[34] T. Prodromakis, B. Peh, C. Papavassiliou, and C. Toumazou, "A versatile memristor model with nonlinear dopant kinetics," *IEEE Trans. Electron. Devices*, vol. 58, no. 9, pp. 3099–3105, Sep. 2011.

[35] A. G. Radwan, M. A. Zidan, and K. N. Salama, "HP memristor mathematical model for periodic signals and DC," in *Proc. IEEE Int. Midwest Symp. Circuits Syst*., Aug. 2010, pp. 861–864.

[36] S. Kvatinsky, E. Friedman, A. Kolodny, and U. Weiser, "Team: Threshold adaptive memristor model," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 1, pp. 211–221, Jan. 2013.

[37] A. G. Radwan, M. A. Zidan, and K. N. Salama, "On the mathematical modeling of memristors," in *Proc. IEEE Int. Conf. Microelectron*., 2010, pp. 284–287.

[38] K. Eshraghian, O. Kavehei, K.-R. Cho, J. M. Chappell, A. Iqbal, S. F. Al-Sarawi, and D. Abbott, "Memristive device fundamentals and modeling: Applications to circuits and systems simulation," *Proc. IEEE*, vol. 100, no. 6, pp. 1991–2007, Jun. 2012.

[39] A. Ascoli, F. Corinto, V. Senger, and R. Tetzlaff, "Memristor model comparison," *IEEE Circuits Syst. Mag*., vol. 13, no. 2, pp. 89–105, Apr.–Jun. 2013.

[40] "HP to replace flash and SSD in 2013," *Electronics Weekly*, Oct. 2011.

[41] International technology roadmap for semiconductors. (Retrieved 2013). [Online]. Available: http://www.itrs.net/

[42] C. J. Xue, Y. Zhang, Y. Chen, G. Sun, J. J. Yang, and H. Li, "Emerging non-volatile memories: Opportunities and challenges," in *Proc. IEEE 9th Int. Conf. Hardware/Software Codesign Syst. Synthesis*., 2011, pp. 325–334.

[43] M. A. Zidan, H. A. H. Fahmy, M. M. Hussain, and K. N. Salama, "Memristor-based memory: The sneak paths problem and solutions," *Microelectron. J*., vol. 44, no. 2, pp. 176–183, 2013.

[44] P. O. Vontobel, W. Robinett, P. J. Kuekes, D. R. Stewart, J. Straznicky, and R. S. Williams, "Writing to and reading from a nano-scale crossbar memory based on memristors," *Nanotechnology*, vol. 20, no. 42, p. 425204, 2009.

[45] M. Qureshi, W. Yi, G. Medeiros-Ribeiro, and R. Williams, "AC sense technique for memristor crossbar," *Electron. Lett*., vol. 48, no. 13, pp. 757–758, 2012.

[46] K.-H. Kim, S. Hyun Jo, S. Gaba, and W. Lu, "Nanoscale resistive memory with intrinsic diode characteristics and long endurance," *Appl. Phys. Lett.*, vol. 96, no. 5, p. 053106, 2010.
[47] Y. Ho, G. Huang, and P. Li, "Dynamical properties and design analysis for nonvolatile memristor memories," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 4, pp. 724–736, Apr. 2011.
[48] Q. Xia, W. Robinett, M. W. Cumbie, N. Banerjee, T. J. Cardinali, J. J. Yang, W. Wu, X. Li, W. M. Tong, D. B. Strukov, G. S. Snider, G. Medeiros-Ribeiro, and R. S. Williams, "Memristor–CMOS hybrid integrated circuits for reconfigurable logic," *Nano Lett.*, vol. 9, no. 10, pp. 3640–3645, 2009.
[49] Predictive Technology Model (PTM). (Retrieved 2013). [Online]. Available: http://ptm.asu.edu/
[50] M. K. Qureshi, S. Gurumurthi, and B. Rajendran, "Phase change memory: From devices to systems," *Synthesis Lectures Comput. Archit.*, vol. 6, no. 4, pp. 1–134, 2011.
[51] R. J. Baker, *CMOS: Circuit Design, Layout, and Simulation*. Hoboken, NJ, USA: Wiley-IEEE Press, 2011.
[52] B. Keeth, R. J. Baker, B. Johnson, and F. Lin, *DRAM Circuit Design: Fundamental and High-Speed Topics*. Hoboken, NJ, USA Wiley-IEEE Press, 2007.
[53] W. Kester, Ed., *Data Conversion Handbook (Analog Devices)*. Burlington, MA, USA: Newnes, 2004.
[54] H. Wei, C.-H. Chan, U.-F. Chio, S.-W. Sin, U. Seng-Pan, R. Martins, and F. Maloberti, "A 0.024mm2 8b 400MS/s SAR ADC with 2b/cycle and resistive DAC in 65nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2011, pp. 188–190.
[55] M. Verhelst and B. Murmann, "Area scaling analysis of CMOS ADCs," *Electron. Lett.*, vol. 48, no. 6, pp. 314–315, 2012.
[56] "CMP's semiconductor insights recognizes Micron's 78nm 1Gb DDR3 as most innovative DRAM." (Retrieved 2013). [Online]. Available: http://investors.micron.com/releasedetail.cfm?releaseid=440718
[57] A. T. Tran and B. M. Baas, "Design of an energy-efficient 32-bit adder operating at subthreshold voltages in 45-nm CMOS," in *Proc. IEEE Int. Conf. Commun. Electron.*, 2010, pp. 87–91.

Mohammed Affan Zidan (S'11) received the B.Sc. degree (ranked first) in electronics and communications engineering from the Institute of Aviation Engineering and Technology, Giza, Egypt, in 2006, and received the M.Sc. degree (ranked first) in electronics and communications engineering from the Faculty of Engineering, Cairo University, Giza, Egypt, in 2010. He is currently working toward the Ph.D. degree in the Electrical Engineering Program, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia.

Before joining KAUST, he worked as a Teaching Assistant at the German University in Cairo, New Cairo, Egypt and the IAET. His research interests include memristor, memory, chaotic systems, ray-tracing, and computer arithmetic.

Ahmed M. Eltawil (S'97–M'03) received the Doctorate degree from the University of California, Los Angeles, CA, USA, in 2003, and the M.Sc. and B.Sc. degrees (with Hons.) from Cairo University, Giza, Egypt, in 1999 and 1997, respectively.

He is an Associate Professor at the University of California, Irvine, CA, USA. Since 2005, he has been with the Department of Electrical Engineering and Computer Science, University of California, Irvine. He is the Founder and Director of the Wireless Systems and Circuits Laboratory. His current research interests include low power digital circuit and signal processing architectures for wireless communication systems.

Dr. Eltawil has been on the technical program committees and steering committees for numerous workshops, symposia, and conferences in the area of VLSI, and communication system design. He has received several distinguished awards, including the NSF CAREER Award in 2010 supporting his research in low-power systems.

Fadi Kurdahi (S'85–M'87–SM'03–F'05) received the Ph.D. degree from the University of Southern California, Los Angeles, CA, USA, in 1987.

Since then, he has been a faculty at the Department of Electrical & Computer Engineering at UCI, where he conducts research in the areas of system level design, reconfigurable computing and cyber-physical systems, and serves as the Director for the Center for Embedded Computer Systems. His research was disseminated in about 200 publications.

Dr. Kurdahi received numerous accolades including the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION Best Paper Award in 2001 and the ISQED Best Paper Award in 2006. He served on several journal boards and conference committees. He received the Distinguished Alumnus Award from his alma mater, the American University of Beirut in 2008. He is a Fellow of the American Association for the Advancement of Science.

Hossam A. H. Fahmy (S'93–M'04–SM'9) received the B.Sc. and M.Sc. degrees from the Department of Electronics and Communications Engineering, Cairo University, Giza, Egypt, in 1995 and 1997, respectively and received the Ph.D. degree from the Department of Electrical Engineering, Stanford University, Stanford, CA, USA, in 2003.

Currently, he is an Associate Professor with the same department where he leads research on computer arithmetic, computer architecture, and computer fonts. He has taught previously at the German University in Cairo, the American University in Cairo, Nanyang Technological University in Singapore, and King Abdullah University for Science and Technology in Saudi Arabia. His previous research include collaborations with IMEC in Belgium, IBM T.J. Watson Research Center in USA, and AMD in California, USA. He is the Cofounder and Chief Technical Officer of SilMinds the world leader in providing hardware decimal floating point units.

Dr. Fahmy served as a Member of the working group developing the IEEE Standard 754-2008 on Floating Point Arithmetic from 2002 till its publication in 2008. He is currently serving as a Member of the working group formulating the future IEEE Standard P1788 on Interval Arithmetic.

Khaled N. Salama (S'97–M'05–SM'10) received the Bachelor's degree with honors from the Department of Electronics and Communications, Cairo University, Giza, Egypt in 1997, and the Master's and Doctorate degrees from the Department of Electrical Engineering, Stanford University, Stanford, CA, USA, in 2000 and 2005, respectively.

He was an Assistant Professor at RPI between 2005 and 2008. He is an Associate Professor at King Abdullah University of Science and Technology, Thuwal, Saudi Arabia, and was the Electrical Engineering Founding Program Chair from August 2009 till August 2011.

Dr. Salama's work on CMOS sensors for molecular detection has been funded by the National Institutes of Health and the Defense Advanced Research Projects Agency, awarded the Stanford-Berkeley Innovators Challenge Award in biological sciences and was acquired by Lumina Inc., in 2008. He is the Cofounder of ultrawave labs, a biomedical imaging company that was funded by VC. He is the coauthor of 110 papers and 10 patents on low-power mixed-signal circuits for intelligent fully integrated sensors and nonlinear electronics specially memristor devices.