Chapter

Development of Compute-in-Memory Memristive Crossbar Architecture with Composite Memory Cells

Mehri Teimoory, Amirali Amirsoleimani, Arash Ahmadi and Majid Ahmadi

Abstract

In this chapter, we discuss the compute-in-memory memristive architectures and develop a 2M1M crossbar array which can be applied for both memory and logic applications. In the first section of this chapter, we briefly discuss compute-in-memory memristive architectural concepts and specifically investigate the current state off the art composite memristor-based switch cells. Also, we define their applications e.g. digital/analog logic, memory, etc. along with their drawbacks and implementation limitations. These composite cells can be designed to be adapted into different design needs can enhance the performance of the memristor crossbar array while preserving their advantages in terms of area and/or energy efficiency. In the second section of the chapter, we discuss a 2M1M memristor switch and its functionality which can be applied into memory crossbars and enables both memory and logic functions. In the next section of the chapter, we define logic implementation by using 2M1M cells and describe variety of in-memory digital logic 2M1M gates. In the next section of the chapter, 2M1M crossbar array performance to be utilized as memory platform is described and we conceived pure memristive 2M1M crossbar array maintains high density, energy efficiency and low read and write time in comparison with other state of art memory architectures. This chapter concluded that utilizing a composite memory cell based on non-volatile memristor devices allow a more efficient combination of processing and storage architectures (compute-in-memory) to overcome the memory wall problem and enhance the computational efficiency for beyond Von-Neumann computing platforms.

Keywords: compute-in-memory, crossbar, logic design, memory, memristor

1. Introduction

In general, memory devices are considered as one of the most important primitives in every computing system. Although, they play an undeniable role in conventional computers, which the processing units and memory are separate, it is generally believed that future computers, unlike von Neumann architectures, will have a compute-in-memory (CIM) structure. According to Moore's Law and fundamental VLSI limitations, CMOS technology is expected to face constraints and serious challenges at each technology node [1]. These challenges require solutions both short-term and long-term solving technical and strategic difficulties on Moore's Law way. Accordingly, researchers both in academia and industries are working hard on different available options and solutions from device up to architecture levels proposing incremental as well as revolutionary approaches. Regarding to this requirements, many efforts and initiatives have done by researches in order to keep on progress in the emerging memory technologies such as Ferroelectric Random Access Memory (FeRAM) [2], Magnetic Random Access Memory (MRAM) [3], and Resistive Random Access Memory (RRAM) [4], etc. Among all these technologies RRAM (generally referred as memristor) has received a lot of attention not only because of its favorable characteristics of low operating voltage, high speed, simple structure, and nano-scale but also with its logic implementation capabilities of the memristor devices. Memristor first in 1971, was proposed by Chua as a non-linear passive element [5], and then almost 37 years later, in 2008, was physically realized by the HP company, which was fabricated utilizing a Pt–TiO_2–Pt structure [6]. These nano-devices are based on a resistor with6variable resistance which can maintain resistance value upon bias removal that can be used as non-volatile memory cells. In addition to the conventional usage as memory cells, this device has also found variety of interesting applications such as machine learning platforms [7, 8], logic circuits design [9, 10], and neuromorphic systems [11]. Considering memristor as a non-volatile memory device makes it an interesting building block for large scale non-volatile memory systems. The memristor, or memory resistor, has been used in crossbar array architectures [12].

Due to the structural limitations (e.g. sneak path problems, interconnect resistance and etc.) of fully passive arrays (0T1M) various resistive switching memory based structures for the memory cells has been offered in literature such as 1T1M [13, 14], 4M1M [15], 1S1M [16–22], 1D1M [23], and 2T1M [24]. One challenging issue in crossbar array performance is sneak path current which can lead to negative effects on power consumption and limit the array size and other negative effects. Despite of amazing footprint size ($4F^2$) in fully passive crossbars, 1T1M arrays has been developed to reduce the impact of alternate currents with the cost of adding an access CMOS transistor in a single memory cell which significantly reduces the area efficiency of the array. These pseudo-crossbar structures mostly developed for digital memory arrays and they enable making large crossbars by adding more accessibility to each memory cell and avoiding the problem of voltage degradation over memory crossbar interconnects. 2T1M structure [24] is also presented and the auxiliary CMOS device is added to help for self-learning mechanism and these structure are used for spiking neural networks (SNNs). Also, a modified version of these cells are designed in 1T1M [13, 14] manner by getting benefit from the new type of transistor which has a smaller size and has the ability to change the sign of the charge carriers. Two terminal selectors such as non-linear switching elements and diodes are attracting a lot of attentions due to the scalability and small footprint sizes. Symmetric voltage–current characteristics for 1S1R structure in [17–22] avoid using these type of cells in logic applications. Also, for composite memory cells with diodes, Zener diode is utilized due to the low break down voltage which makes possible the rewriting over the memrisor device in each cell. Complementary resistive switch (CRS) with back to back memristor devices provide resiliency toward the sneakpath current by keeping one of the series device in high resistance which reduce the alternate current path in non-selected cells. Pure memristive composite memory array with 4M1M structure is proposed in [15], this structure provides a memristor switch to avoid sneakpath. In this method, at least one of the input devices is in low-resistance state (R_{on}) all the time which connects target cell to other cells in the row through a low resistive network. However, this structure suffers from parallel branches detour currents which considerably impact the power consumption and writing current.

Other attractive domain for composite memory structure is utilizing them for logic applications by collocating the computing within the memory in the same place. Several number of logic design and implementation research works have recently been proposed using memristor devices. Memristor Ratio Logic (MRL) is a CMOS-Memristor structure approach for combinational logic design [25]. In this method logical values are presented as node voltages, but it is a hybrid approach consisting both memristor and MOS transistors in the crossbar fabric. There are also other methods, such as MAGIC [26] and IMPLY [27] in which unlike MRL, memristance of memristors represent logical values. Each approach has positive and negative points regarding required number of memristor or MOS transistors or required time steps.

This chapter discuss 2M1M composite memory array and its application in both memory and logic. The proposed switch provides three modes namely, ON, OFF and No-Change, designed with three memristor. This structure not only can be used as AND, OR, NAND, and NOR logic gates with less computational steps compared to [27], but also the IMPLY logic can be implemented in crossbar array by this memory cell. The proposed cell is a pure memristor memory cell as 2M1M. The read and write operations are done by the same memristor circuits without need for additional circuitry within memory fabric. Thus, significantly reducing the number of required elements and simplifies the crossbar structure. The technique presented in the reading circuit does not need an isolated access to the memristor node which in turn reduces circuit wiring, and leads to a very simple structure with less complexity. Proposed structure provides an effective gating mechanism by which memory elements can be partially isolated from the access line during reading cycle which considerable reduces the sneak path currents. The remainder of this chapter is organized as follows: Section 2 introduces memristor-based switch circuit and its application and performance in the proposed memory cell. In Section 3 the proposed crossbar structure is discussed. Logic implementation and computational operations by 2M1M memory cell are presented in Section 4 and some explanation about sneak path are discussed.

2. A 2M1M memristor cell and its functionality

2.1 2M1M switch circuit

The 2M1M three state switch [9] which functions in ON, OFF, and NC are shown in **Figure 1**. As it can be seen, the proposed memory cell comprises of three memristor devices X_A, X_B and X_C. X_A and X_B devices are the access devices and they isolate the target device X_C which stores the information. There are three terminals A, B, and C in this structure in which A and B are considered as input terminals and V_a and V_b (as input voltages of '$-V$' or '$+V$') should be applied to terminals A and B respectively. Operation of the circuit, regarding V_a and V_b as input voltages and V_M as its output can be explained as follows. The input voltage ($+V$) for logic '1' and the input voltage ($-V$) for the logic '0' are applied to the X_A and X_B memristors ($|\pm V| > |V_{th}|$). The voltage V_M on common node of memristors represents output of the circuit while memristor X_C maintains this value in form of memristance. The truth table of this circuit is shown in **Table 1**.

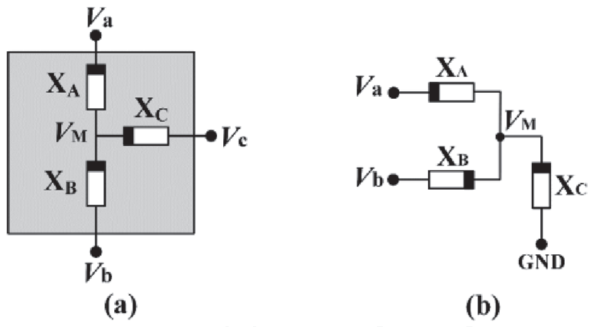

Figure 1.
Schematic of the proposed memory cell. (a) General circuit of memory cell. (b) Configuration for switch circuit mode. Figure reprinted by [9].

Case	V_a	V_b	V_M	R_c	Switch State
1	$-V$	$-V$	$-V$	R_{off}	OFF
2	$-V$	$+V$	0	R_{ini}	NC
3	$+V$	$-V$	0	R_{ini}	NC
4	$+V$	$+V$	$+V$	R_{on}	ON

Table 1.
Truth table for the proposed switch circuit.

$$\begin{gathered}\frac{(V_M - V_a)}{R_a} + \frac{(V_M - V_b)}{R_b} + \frac{V_M}{R_c} = 0 \\ \rightarrow V_M\left(\frac{1}{R_a} + \frac{1}{R_b} + \frac{1}{R_c}\right) = \frac{V_a}{R_a} + \frac{V_b}{R_b}\end{gathered} \tag{1}$$

and providing R_a, $R_b << R_c$ we can approximate V_M as:

$$V_M = \frac{R_b}{R_a + R_b} V_a + \frac{R_a}{R_a + R_b} V_b \tag{2}$$

When both inputs are '0' ($-V$) according to the polarity of memristors, since a negative voltage is applied across memristor X_A and a positive voltage across memristor X_B, so their memristance, regardless of their initial states, will change to Roff and Ron, respectively. Therefore, according to Kirchhoff's law and also considering the initial state of the memristor X_C, as $R_C >> R_A$, R_B (memristance of X_A and X_B), the voltage in common node of memristors is: $V_M = -V$. This will set memristance of X_C to R_{off}, which is logical zero:

$$\begin{aligned} V_a &= V_b = -V \\ V_M &= \frac{R_a + R_b}{R_a + R_b} \cdot (-V) = (-V) \approx Logical\ 0 \end{aligned} \tag{3}$$

For logic 1, according to the fourth row of the **Table 1**, when both inputs are in the same value of $+V$, similarly, based on the polarity and direction of memristors, the memristor X_A is set to R_{on} and memristor X_B becomes R_{off}. Therefore, the voltage on the output node (M) is approximately $+V$ which will change the memristance of the output memristor, X_C, to R_{on} representing logical one:

$$\begin{aligned} V_a &= V_b = V \\ V_M &= \frac{R_a + R_b}{R_a + R_b} \cdot V = V \approx Logical1 \end{aligned} \tag{4}$$

Otherwise, if the value of input voltages is different as $(+V)$ and $(-V)$, both input memristors have the same value of either R_{on} or R_{off} according to the applied voltage and their polarity. This results in a zero voltage on common node. Since $V_{\text{c}} = 0$, in this case memristance of X_C does not change:

$$\begin{aligned} V_a &= -V_b = V \\ V_M &= \frac{R_{on}}{R_{on} + R_{on}} \cdot V + \frac{R_{on}}{R_{on} + R_{on}} \cdot (-V) = 0 \end{aligned} \tag{5}$$

or similarly:

$$\begin{aligned} V_a &= -V_b = (-V) \\ V_M &= \frac{R_{off}}{R_{off} + R_{off}} \cdot (-V) + \frac{R_{off}}{R_{off} + R_{off}} \cdot V = 0 \end{aligned} \tag{6}$$

This state is called a NO-Change state. To have a timing analysis of switches operation, according to [15]:

$$\frac{dR(t)}{dt} = -k \cdot i(t) = -k \cdot \frac{V}{R(t)} \tag{7}$$

$$k = \mu.\Delta R \cdot R_{on}/D^2 \tag{8}$$

$$\Delta R = R_{off} - R_{on} \tag{9}$$

Because in the fourth combination of the truth table of memristor based switch (**Table 1**), memristor X_A is parallel with memristor X_B then $V_a = V_b$. Therefore:

$$V_a = \frac{R(t) \cdot dR(t)}{-k \cdot d(t)} \tag{10}$$

By integrating (8) and also assuming that $\phi_0 = 0$, $\phi(t)$ is given by

$$\int Va = \int \frac{R(t) \cdot dR(t)}{-k \cdot d(t)} = \int \frac{d\phi}{dt} \tag{11}$$

$$R_a{}^2(t) - R_{ai}{}^2 = -2k_a\phi(t) \tag{12}$$

$$\phi(t) = \frac{\left(R_a{}^2(t) - R_{ai}{}^2\right)}{-2k_a} \tag{13}$$

and also by supposing the initial state of memristor X_A is R_{off} and its final state is R_{on}, the required flux across the memristor X_A is

$$\phi(t) = \frac{\left(R_{on}{}^2 - R_{off}{}^2\right)}{-2k_a} \tag{14}$$

Thus, the required time for change state of memristor X_A and X_B is given by:

$$\Delta\phi_a = V_a T_1 \tag{15}$$

$$\begin{aligned} T_1 &= \frac{\left(R_{on} + R_{off}\right) \cdot D^2}{-2V_a \mu_v R_{on}} \\ T_1 &= T_2 \end{aligned} \tag{16}$$

In memristor X_C the required time for the change of state is:

$$T_3 = \frac{\left(R'_{on} + R'_{off}\right) \cdot D^2}{-2V_C \mu'_v R'_{on}} \tag{17}$$

Therefore, the total time to change the cell state, T_t, is given by:

$$\begin{aligned} T_t &= T_1 + T_3 \\ T_t &= \frac{\left({R_{off}}^2 - {R_{on}}^2\right)}{-2k_a V_a} + \frac{\left(R'_{on} + R'_{off}\right) \cdot D^2}{-2V_C \mu'_v R'_{on}} \end{aligned} \tag{18}$$

2.1.1 Simulation result for 2M1M switch circuit

This is in general agreement with the simulation results as presented in follow,

Despite several memristor SPICE models which are presented in [28, 29], the simulation results are performed using Biolek model presented in [30]. This model is selected due to the fact that, it can be utilized in mathematical analysis for power and delay estimation besides its validity to characterize the memristor switching behavior. PSPICE software has been utilized to perform the simulations. The simulations are carried out by using the parameters in **Table 2**, and for a fair comparison, these parameters are similar with [15] to evaluate functionality of the design.

Different combinations of inputs which are applied to the switch are shown in **Figure 2**. As it can be seen the simulations results are in agreement with the truth table of **Table 1**. Here the voltage is applied and output logics is represented by memristance of the X_C. Delay or settling times for this switch is defined by the time which X_C memristance reaches to its final value. According to the simulation results, this time is 1.11 ns which is also in agreement with theoretical calculations.

2.2 Write and read operations

For the write operation, the memory cell should work based on the first and forth rows of **Table 1**, respectively for writing '0' and '1', as descried in details in subsection 2.1. For read operation, unlike previous works, these cells do not need any additional wiring or complicated sense circuitry. This is because in this circuit

Parameters	Value	Parameters	Value
D(nm)	1	V_{th}(V)	0.11
V_w(V)	0.9	V_R(V)	0.1
R_{on}(Ω)	100	R'_{on}(kΩ)	10
R_{off}(Ω)	900	R'_{off}(Ω)	1900
μ(m^{2}V^{-1} s^{-1})	1×10^{-6}	μ'(m^{2}V^{-1} s^{-1})	1×10^{-7}

Table 2.
Simulation parameters for 2M1M memory architecture.

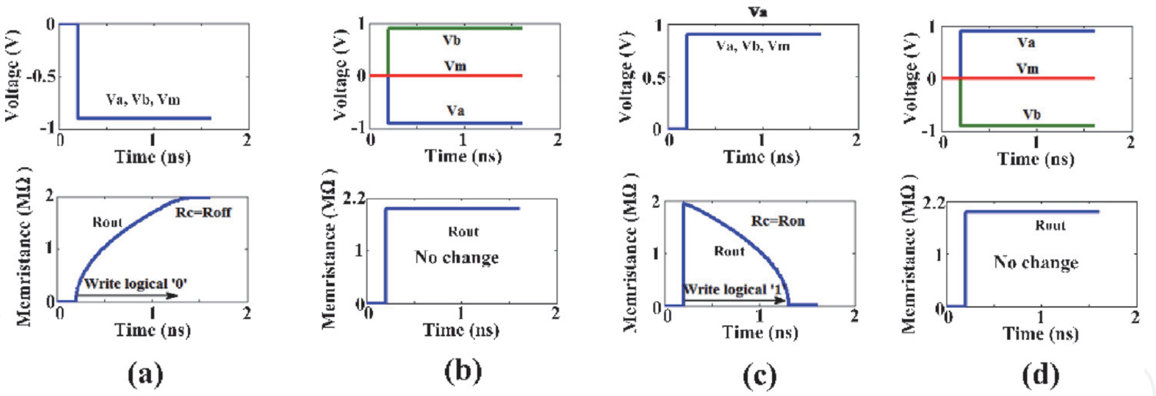

Figure 2.
Memristor switch circuit simulation results for different cases. In each subfigure, upper figure displays the inputs and common node voltages while the below figure displays the output device resistance state. (a) a = 0, B = 0. *(b)* a = 0, B = 1. *(c)* a = 1, B = 0. *(d)* a = 1, B = 1. *Figure reprinted by [9].*

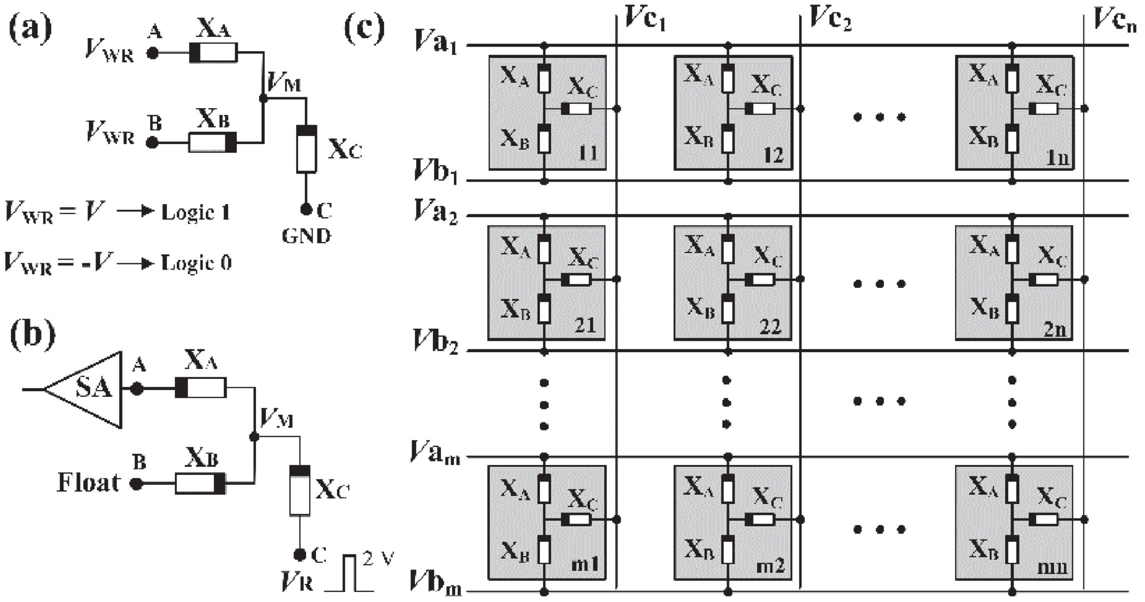

Figure 3.
(a) Write and (b) read circuit configurations of the proposed memory cell. (c) 2M1M Memristor-based crossbar architecture. Figure reprinted by [9].

memristor change inertia, as shown in **Figure 3**, is exploited. In general, the READ operation is done by floating node B and applying READ signal V_R to node C. Then, a sense amplifier (SA) is sensed the current from node A of the memory cell. In other words, in this technique a read signal is applied to the memristor which does not change the memristance, either because of its high frequency ($f > f_{th}$) or/and because of its low voltage ($V < V_{th}$).

In read process method, a pulse *V*R with appropriate amplitude and small duration is applied to the circuit as "read signal". A sense amplifier then measures value of the propagated signal on port A of the switch. Width and amplitude of this pulse (or spike) should be chosen in a way to do not affect memristors' state during read process. As mentioned before in high frequencies memristor operates like a pure resistive element. If we connect A and B ports of the proposed circuit to the ground, and apply to the other end of the memristor X_C (port C), a read spike with amplitude voltage of 2 V, as shown in **Figure 3b**, the output voltage can be read at node VM as:

$$V_M = \frac{(R_{on}\|R_{off})}{(R_{on}\|R_{off}) + R_c} \cdot 2V \cong \frac{R_{on}}{R_{on} + R_c} \cdot 2V \tag{19}$$

which in terms of logic vales can be described as:

$$\begin{aligned} if: R_c = R_{on} \rightarrow V_M = V = \text{logical} \quad 1 \\ if: R_c = R_{off} \rightarrow V_M \cong 0 = \text{logical} \quad 0 \end{aligned} \tag{20}$$

With this technique we can increase reading speed and reduce power consumption. In addition, if we read the output value from port A instead of node M, this read method does not require any additional wiring to access node M. This considerably reduces fabrication and wiring complexity of the proposed crossbar structure.

3. 2M1M Memristor crossbar architecture

In crossbar architecture the 2M1M memory cell can be used effectively as shown in **Figure 3c**. While, there is no need CMOS transistors for each cell within cross bar fabric in this architecture. As it can be seen in **Figure 3c**, the similar nodes of memory cells in the crossbar structure are connected to each other in the horizontal rows nodes A_i and B_i are of the cells are connected to each other separately while in vertical columns modes C_i are connected to each other. The desired reading or writing operation are performed by applying appropriate voltages in suitable rows and columns to activate a cell and disable others.

3.1 Write operation in the crossbar

For write operation in the crossbar architecture, like a single memory cell, appropriate input values need to be applied to the memory cell based on **Table 1**. This means that, both applied voltages V_a and V_b should be similar, either amount of '+V' or '$-V$' to write logical '1' or to write logical '0' respectively. Otherwise, the other states in truth table, the switch is in the No-Change state. This scheme is easily applicable to the crossbar structure in the same way as a single cell using connected cell port (a_i, b_i and c_i). It should be considered that when read or write signal are applied, cells should be completely isolated from the target cell. When writing in a cell (or a number of associated cells as a word), the other cells should have maintained their saved vales. For more explanation for write operation, as an example. Considering cell 22 as a target cell to write '0' ('1') in **Figure 4**. In this case, the same voltages $-V$ (similarly +V for '1') should be applied to both memristors X_A and X_B and node C is connected to GND. In this situation, in terms of applied voltages

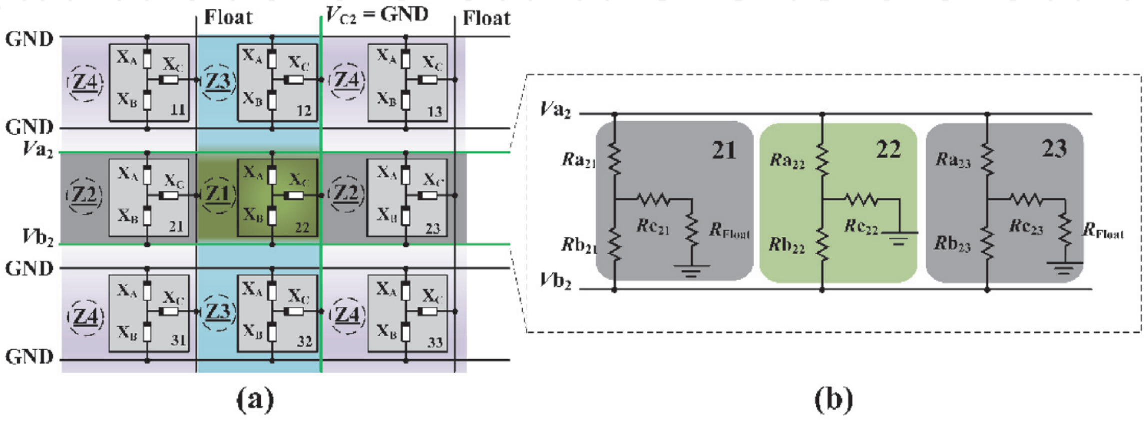

Figure 4.
(a) Write configuration in 3 × 3 2M1M crossbar memory and the hazardous zones are displayed. Z1 zone contains target cell 22 and highlighted by green. (b) Resistive equivalent circuit of zone Z_2 in the proposed 3 × 3 2M1M crossbar memory. The equivalent resistor circuit of the target cell 22 is highlighted by green. Figure reprinted by [9].

combination, four different zones are recognizable in the crossbar structure, as shown in **Figure 4**. As it can be seen only the target cell is located in the first area, Z_1. The second area is Z_2, where all cells have the same voltages as target cell on their a_i and b_i nodes. In the third area, Z_3, cells have the voltage on their node C, which is the same as the target cell 22. In the fourth zone, Z_4, there is no input in common with cell 22. In the Z_1 to write '0' ('1') into the cell 22 voltage $-V$ $(+V)$ is applied to rows V_{b2} and V_{a2} where the column V_{c2} is connected to GND.

As can be seen in the **Figure 4**, Z_2 is the hazardous zone because in this area same voltages as the target cell ($-V$ or $+V$) are applied to the V_{a2} and V_{b2} ports of the cells. It can cause an unwanted writing and changing the state of the memristors that are not supposed to change. To deal with this issue, C nodes of the neighbor cells in zone Z_2 are floated or in practice connected to a high impedance open circuit (the columns V_{c1} and V_{c3} in **Figure 4**). Since the columns c_1 and c_2 are floated, we consider a resistance R_{Float} for each of these columns and this resistance is connected to the non-bar side of the X_C device in each of 21 and 23 2M1M memory cells. The bar side of X_C device is connected to the common node between X_A and X_B devices. Then, the equivalent resistor-based circuit for cell 21 has two serially connected resistors R_{a21} and R_{b21} which are connected to rows a_2 and b_2 and they are memristances of X_A and X_B devices in cell 21, respectively. The common node of X_A and X_B is connected to a R_{c21} resistor which is a memristance of Xc device in cell 21. Therefore, resistance of the float column R_{Float} will be in series with R_{c21} resistor (this is true for cell 23). If the float resistance terminals were connected to both terminals of X_C device, then we could consider R_{Float} was parallel with R_{c21} while here only the non-bar side of X_C device is connected to the floated column c_1.

Resistor equivalent circuit of this zone is depicted in **Figure 4b**. This floated port connection reduces current through X_C memristor of the unselected cells (≈ 0) which keeps the stored values of the cells untouched. The rest of the rows and columns in the cross-bar structure are connected to ground. Thus, points A, B and C of cells that are in zones Z_3 and Z_4 are either floating or connected to GND. the logical state of these cells therefore do not change during write operation. Although, maybe one of the two memristor X_A and X_B is sufficient to perform write operations and by help one of them could to done correctly write operation but as mentioned, this structure is designed to be based on a three-state switch ON, OFF and NO Change. The second case is used to high impedance operation for memristors without changing of output memristor in practice reading that through this can reduce the sneak paths current. In addition, one of the applications of these cells has been mentioned is the implementation of logic circuits which is explained in the Section 4. Please note that in this structure cells in the same column are almost independent and can be written or read simultaneously. This makes it possible to have a parallel read/write process on these cells for higher rate memory access operations or combine a number of them forming data "word" rather than collections of single bits.

In the write operation as can be seen in **Figure 4a**, the memory cells 21 and 23 are in zone 2 and they are the neighboring cells of the target cell for write operation. The equivalent resistor-based circuit of these cells are displayed in **Figure 4b**. Write operation of the memory cell in 3×3 2M1M crossbar array is simulated in **Figure 5a** and **b**. This memory cell is functioning even by having a time difference between the applied voltage input V_a and V_b. To test the proposed memory cell for this special case, two asynchronous input voltages are applied to the memory cell and the simulation results prove its functionality (**Figure 5c**).

3.2 Read operation in the crossbar

Regarding read operation, there are four zones in the crossbar as described in previous section. During read operation, as shown in **Figure 6a**, by applying voltage

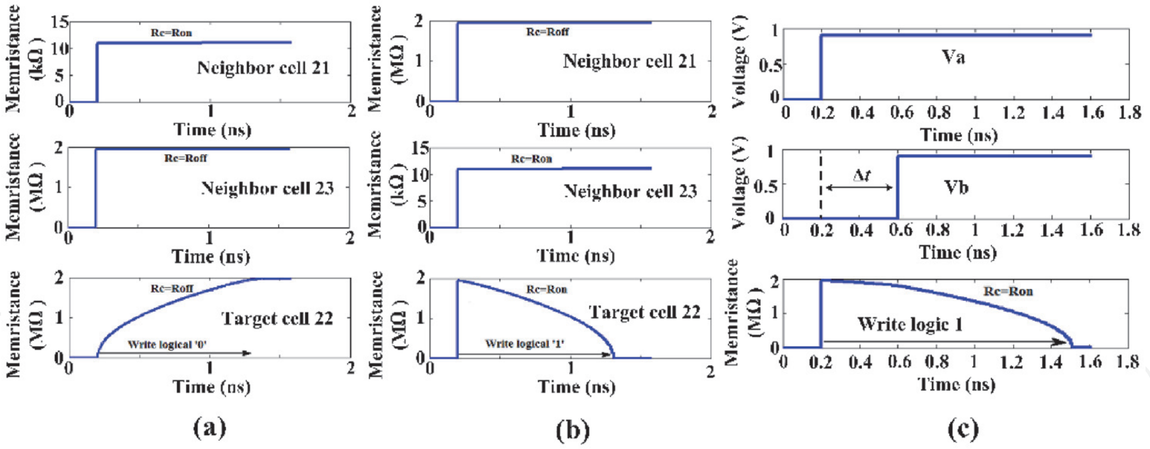

Figure 5.
Writing in target cell with different neighbor cell's stored value. (a) Write 1 in target cell. (b) Write 0 in target cell. (c) Write operation when two asynchronous input voltages are applied to the target cell with the time difference. Figure reprinted by [9].

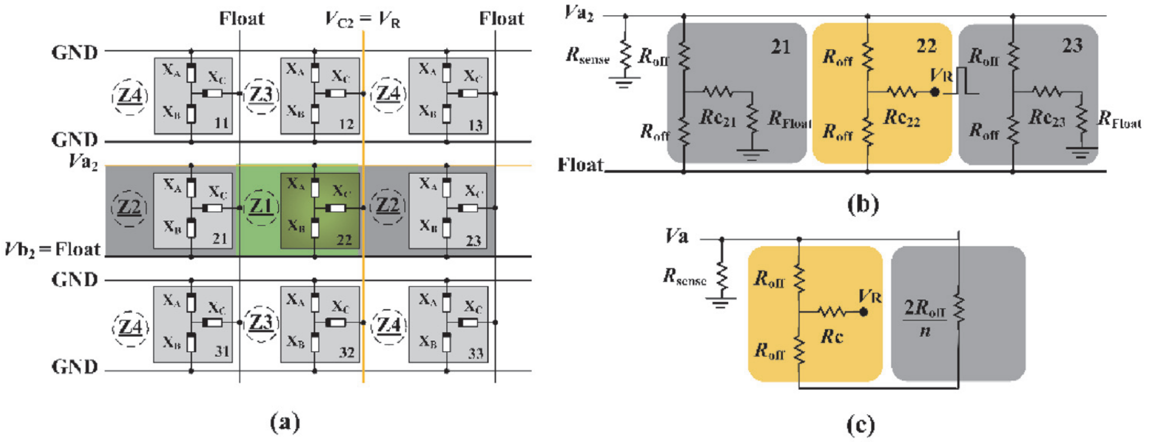

Figure 6.
Read configuration in 3×3 2M1M crossbar memory and the hazardous zones are displayed. (a) Z1 which contains target cell 22 is highlighted by green. Equivalent resistive circuit for hazardous zone for read operation. Target cell for read operation is highlighted by orange. (b) Resistive circuit for hazardous zone in 3×3 2M1M crossbar memory. (c) Equivalent resistive circuit by considering n neighbor cells. Figure reprinted by [9].

V_R to node C and stored bit in X_C can be read as a voltage from node A. Suppose that we want to read from cell 22. The read operation must be performed in 2 stages; in the first stage, memristors X_A, X_B of the Z_2 memory cells are changed to high impedance (R_{off}) to partially isolate neighbor cells in this zone from applied read spike which can be done by applying voltages $-V$ and $+V$, according to truth table of **Table 1**, to lines V_a and V_b of the cells in the zone respectively. At second stage the read signal is applied to port C of the target cell and voltage of port A of the cell is read. This stage must be performed by floating row V_{b2}, applying voltage V_C to column V_{c2}, reading (measuring) the voltage on node A using a sense amplifier. The important point at this stage is considering appropriate signal as read signal. It is very important that applied read pulse be strong enough to induce a readable voltage at A line of the row. And also this signal should not affect memristance values of the memristors in the target cell or the neighbors. Here a spike shaped narrow pulse is used as read signal (V_C).

Another consideration which is so important in this crossbar architecture is effect of neighbor cells in the output readout value. In this case the circuit can be assumed as a resistive network and areas involved in this operation are Z_1, Z_2 that can be seen in **Figure 6**. Sneak path current is considered as one of the most important issues in memristor crossbar memories. Here, X_A and X_B of the neighbor cells to "gate" effect of the X_C memristance of the neighbors from read signal are used. By changing memristance of X_A and X_B memristors of the neighbors to R_{off}, as shown in **Figure 6b**, the target cell will be in parallel connection with its neighbors which are gated form a_i line by $2R_{off}$ memristance. Since all cells in these areas, except cell 22, have a floating (R_{float}) resistance connected to the node C, each neighbor cell can be considered as a 2Roff resistors in parallel with cell 22. The value of these parallel resistances is equivalent to $(2/n) \times R_{off}$ (n = total number of columns per row). Accordingly, equivalent resistance of the neighbor cells from ai line is almost independent from their X_C memristance, which represents stored value in the cell. This technique considerably reduces sneak path effect and its negative effect on cells' readout process. Using equivalent circuit of **Figure 6c**, the readout voltage and equivalent neighbor cells resistance can be calculated as:

$$V_a = \frac{R_{sense}}{((1+\frac{2}{n})R_{off})||R_{off}) + R_c + R_{sense}} . V_c \tag{21}$$

$$\left(1+\frac{2}{n}\right)R_{off})||R_{off}) \cong \frac{R_{off}}{2} \tag{22}$$

by selecting of R_{sense} = R_{off},

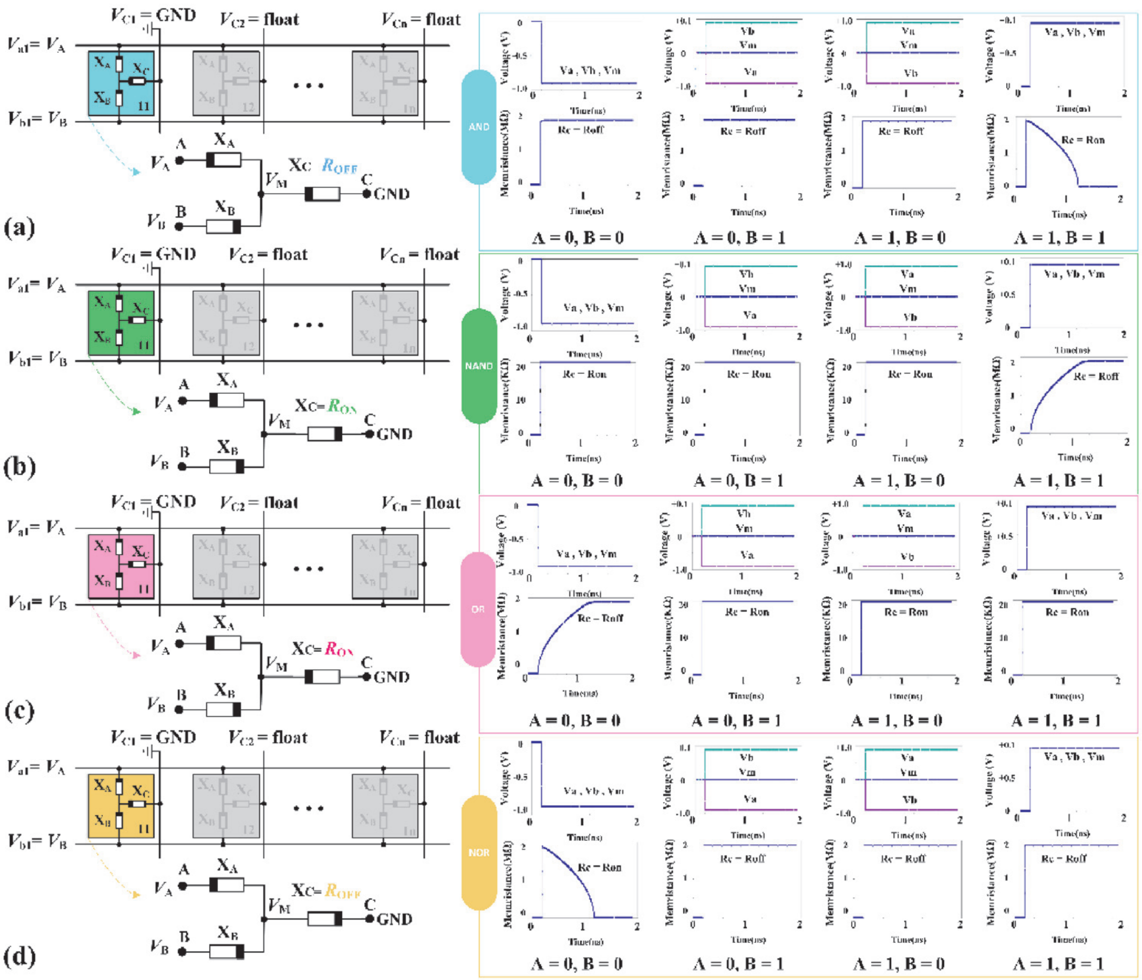

Figure 7.
2M1M array logic schematics and simulation results for AND, NAND, OR, and NOR. (a) AND logic gate for 2M1M switch. (b) NAND logic gate for 2M1M switch. (c) OR logic gate for 2M1M switch. (d) NOR logic gate for 2M1M switch.

$$
\begin{aligned}
&V_a = \frac{R_{off}}{\frac{R_{off}}{2} + R_c + R_{off}} \cdot V_c \\
&if : R_c = R_{on} \Rightarrow V_a = \frac{R_{off}}{\frac{R_{off}}{2} + R_{on} + R_{off}} \cdot V_c \approx \text{Logical } '1' \\
&if : R_c = R_{off} \Rightarrow V_a = \frac{R_{off}}{\frac{R_{off}}{2} + R_{off} + R_{off}} \cdot V_c \approx \text{Logical } '0'
\end{aligned}
\tag{23}
$$

The second voltage V_a from node A is higher or lower voltage according to X_c which is low or high resistance state (R_{on} or R_{off}). **Figure 7**, presents simulation results for a read operation in the crossbar structure. As it is discussed, to read the stored value of a cell we have to apply a spike like pulse to the C node of the cell and read the voltage from line a_i, where bi line of the row and C node of the other cells in the same row are float. When R_c is R_{off} the voltage in node A is a low voltage that is equivalent to logic zero and vice versa, when R_c has the value of R_{on}, the voltage in node A has a higher voltage which represents to a logic one. **Figure 8**, presents two different cases. **Figure 8a**, shows reading '0' from a cell, when the neighbor stored '1' **Figure 8b**, shows reading '1' from a cell, when the neighbor stored '1'. In both cases target cell has been readout correctly. According to the simulation results

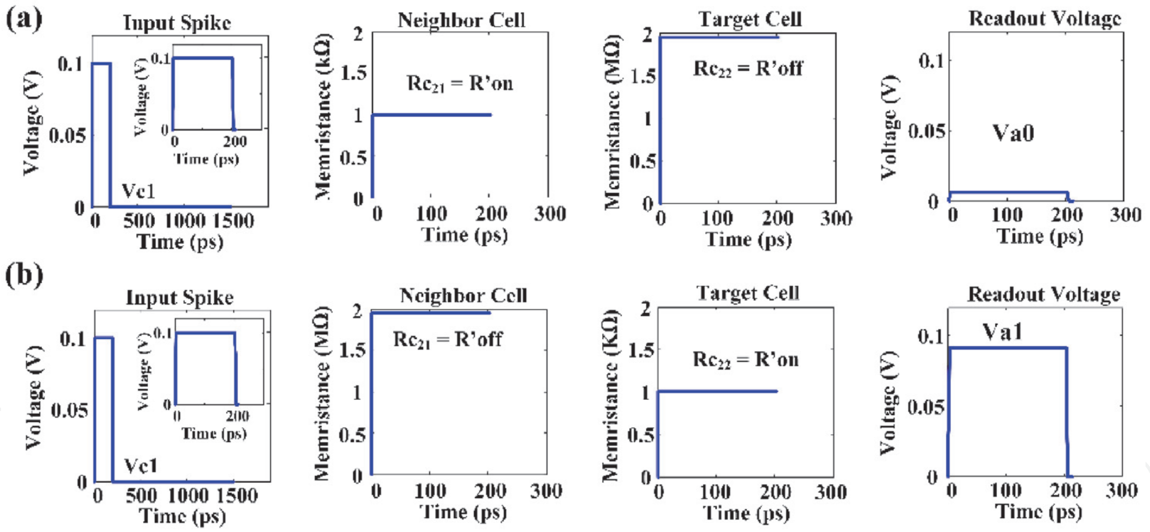

Figure 8.
The simulations of memory cell in crossbar array for; (a) read of logic 0 from target cell and other cell, (b) read logic 1 from target cell and other. Figure reprinted by [9].

	Memory [23]	Memory [24]	4M1M [15]	2M1M
Read time	1.2 ns	1.095 ns	0.25 ns	20 ps
Write voltage	1.0 V	0.9 V	0.9 V	0.9 V
Read voltage	±1.0 V	0.9 V	0.1 V	0.1 V
Number of consecutive read	—	130	$\gg 10^5$	$\gg 10^5$
Number of consecutive read by 10% noise	20	—	$\gg 10^5$	$\gg 10^5$

Table 3.
Comparison of read operation with previous works.

and the formula presented in the [16], reading margin in this work is equal to the amount 0.7 V which can be a reasonable amount.

$$\mathrm{RM} = \frac{\Delta Vout}{\Delta Vread} = \frac{Vout(LRS) - Vout(HRS)}{V_{WS}} \tag{24}$$

where V_{WS} is the read voltage applied. Simulation results are compared with previous works in **Table 3**. As it can be seen in this table, the proposed method is considerably better than [31, 32], and is similar to [15].

4. Logic implementation and computational operations by 2M1M memory cell

Composite memory cells can be applied to implement digital logics. In addition to its memory application, the proposed memory cell is capable of implementing logic which makes it capable for in-memory computing applications. Here, in this section we are assessing the logic implementation of the proposed architecture with 2M1M cells.

4.1 Logic gates with 2M1M switch

From switching point of view, this circuit is a three state switch as 'ON', 'OFF' and 'No-Change'. Interestingly, this switch can also be used as logic gates. By setting the initial memristance value of the output memristor to R_{on} or R_{off}, final memristance state of memristor $\mathrm{X_C}$, respectively, AND or OR logic gate operations are developed. Further, by changing the polarity of the output memristor ($\mathrm{X_C}$) one can make NAND and NOR gates in a similar way. Therefore, the 2M1M array can develop two different logic schemes based on the polarity of memristor $\mathrm{X_C}$. First, include AND and OR gates and by changing the polarity of $\mathrm{X_C}$ the array can develop NAND and NOR gates. The logic is based on the resistance of device and not the voltage. This will make this logic to enable in-memory compute logic family as the data will store within the memory array after finishing the operation.

The input voltage pulses with amplitude $+V$ and $-V$ are applied as logic 1 and 0 into the rows $\mathrm{a_1}$ and $\mathrm{b_1}$. Other unselected rows will be floated and the column $\mathrm{c_1}$ is grounded to shape a 2M1M cell 11 as a logic gate. Other unselected columns need to be floated to inactive the rest of the 2M1M cells in the corresponding row. As an example, the AND gate can be implemented by a 2M1M switch over the 2M1M array by applying the appropriate voltages. This gate is comprised of two access devices $\mathrm{X_A}$ and $\mathrm{X_B}$ which are connected in parallel with different polarities to node M. The output device $\mathrm{X_C}$ is connected between node M and bit-line of the array by a positive polarity. The input voltages should be applied to $\mathrm{a_1}$ and $\mathrm{b_1}$ lines as V_{A} and

A	B	AND		OR		NAND		NOR	
		Vo	Ro	Vo	Ro	Vo	Ro	Vo	Ro
$-V$	$-V$	0	R_{OFF}	0	R_{OFF}	1	R_{ON}	1	R_{ON}
$-V$	$+V$	0	$R_{\mathrm{IN}} = R_{\mathrm{OFF}}$	0	$R_{\mathrm{IN}} = R_{\mathrm{ON}}$	0	$R_{\mathrm{IN}} = R_{\mathrm{ON}}$	0	$R_{\mathrm{IN}} = R_{\mathrm{OFF}}$
$+V$	$-V$	0	$R_{\mathrm{IN}} = R_{\mathrm{OFF}}$	0	$R_{\mathrm{IN}} = R_{\mathrm{ON}}$	0	$R_{\mathrm{IN}} = R_{\mathrm{ON}}$	0	$R_{\mathrm{IN}} = R_{\mathrm{OFF}}$
$+V$	$+V$	1	R_{ON}	1	R_{ON}	0	R_{OFF}	0	R_{OFF}

Table 4.
Truth table of the proposed memristor logic gates.

		IMPLY [27]	4M1M [15]	2M1M
Operation step	AND	4	1	1
	OR	3	1	1
	NAND	3	2	1
	NOR	4	2	1
Required memristors	AND	4	3	3
	OR	3	3	3
	NAND	3	4	3
	NOR	3	4	3

Table 5.
Comparison of proposed 2M1M logic gates with [15, 27].

V_B. Also, $R_C >> R_A$, R_B and the resistance of R_C will specify the output of the logic. The logic can be described for different input combinations by considering the Eqs. (1)–(5).

The truth table of different 2M1M logic gates has been in presented in **Table 4**, by showing different input combination voltages, output voltage and resistance state of the output device. Different 2M1M logic cells, their implementations on memristor crossbar array and the simulation results corresponding to each AND, NAND, OR, and NOR logic gates by using 2M1M cells for different input combinations have been displayed in **Figure 7**. In **Table 5**, the proposed 2M1M logic gates have been compared in terms of number of with IMPLY logic [27] and 4M1M [15]. It has been shown that the proposed logic requires only one computational step to implement in-memory logic for AND, NAND, OR, and NOR gates. Also, the number of required devices to implement all of these logic gates are 3 devices included in a 2M1M cell structure.

Sneak path current is considered as one of the important challenges against practical application of memristor crossbars. During reading operation the sneak path currents through neighbor cells can affect readout value of the target cell. To eliminate or reduce the sneak path in the crossbar array several methods have been proposed by researchers. In general, proposed methods can be divided into two categories. In the first approach [33–36], researchers focus on device level structure of the memristor or read process in the crossbar to make it more resilient against this effect. Among these methods is the way provided in [33] in which read operation is done by an algorithm in several stages. This method improves the sneak path problem but increases read time and require additional circuit to realize the read algorithm stages. Another approach relies on memristive devices with inherent nonlinear structure such as [34, 35] in which a three-terminal memristor device is proposed to solve this problem. In another approach as presented in [36] to eliminate sneak path currents separate columns are considered for each element in the crossbar architecture. That increases cell area and therefore reduces the memory density. In the second approach, to solve the problem of sneak path currents, it is suggested to add additional switches to each memory cell in the crossbar architecture to separate reading path of the target cell from the other unwanted paths. There are several suggestions in this approach, but the most popular structure is 1T1M (one transistor for one memristor) [14]. This structure uses a transistor to separate each cell from other cells during read operation. In this way, added transistor is the gating element of the cell. This method has problems due to the scalability considerations of the CMOS-memristor structure [13]. In [23] diodes, instead of transistors, have been suggested to reduce sneak path. There are difficulties with this approach as well due to diode behavior. In another approach [28], back-to-back memristors are proposed to overcome the problem of sneak path current in which always one of the memristors is in R_{off} state and the other one is R_{on}. In this way, equivalent memristance is always greater than R_{off} which can reduce the sneak path effect. In [15] a memristor based switch is suggested to solve the problem of sneak path. In this method at least one of the input memristors is always in state R_{on}, which connects target cell to other cells in the row through a low resistive network. In this structure, as shown in **Figure 9**, during write process, there is a detour current path through M_S-M_P and M_T-M_P, in all parallel branches in the crossbar structure; witch can considerably increase the writing current and power consumption.

In this study, effect of sneak path can be easily reduced using proposed gating mechanism created by X_A and X_B memristors in the cell. By changing state of these memristors to R_{off}, memristor X_C, which keeps the saved value ('0' or '1') of the memory cell, can be isolated from rest of the network. As discussed before, in the second and third rows of the truth table switch goes to No-Change state and X_C keeps its sate untouched, where both X_A and X_B memristors become either R_{off} or R_{on}. Therefore, if in the crossbar array structure, we apply $-V$ and $+V$ to a_i and b_i lines of the row respectively, memristors X_A and X_B of all the cells in the row will to R_{off} state, which is a high impedance, without any change in their X_C memristance. So unlike cells provided in [15] there is no resistance of R_{on} between the selected node and the other nodes of the circuit. In fact, high impedance of the X_A and X_B memristors isolate X_C of all the cells in the crossbar from each other.

With this approach equivalent circuit of the neighboring cells in a row is as shown in **Figure 9**. Interestingly, target cell (first cell from left) sees an equivalent resistor of the network which is almost independent from stored values (in terms of R_{on} or R_{off}) in other cells. This means if $R_{\text{float}} >> R_{\text{off}}$ then effect of R_c state is negligible on I_{read} current. Simulation results are presented in **Table 6**. As it can be seen this method is far better than [15, 24]. In comparison with [31] sneak path current in this work is higher but please note that in [31] there are two transistors for each memory cell but our cell is transistor-less.

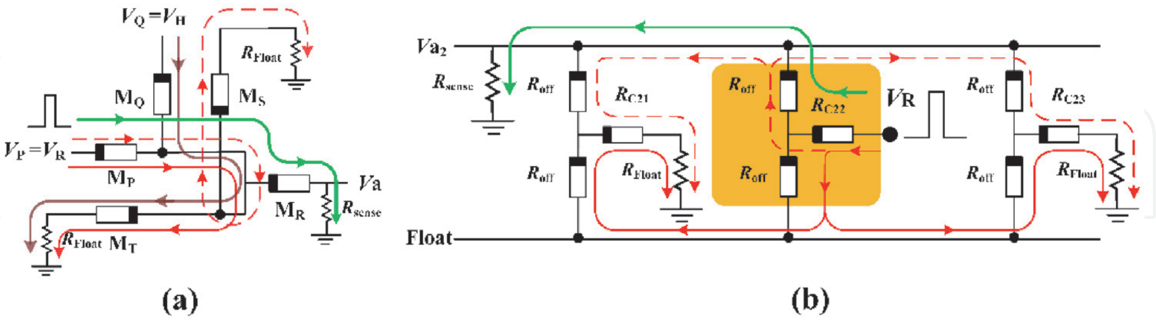

Figure 9.
Sneak path current in 4M1M cell [15] and the proposed 2M1M crossbar memory in a read operation. (a) 4M1M [15] sneak path currents in read operation. (b) Sneak path current in 2M1M crossbar during read operation. Figure reprinted by [9].

	1T1M [13]	2T1M [24]	4M1M [15]	2M1M
Sneak current	0.25–33.29 μA	5.0 pA	0.24–1.77 μA	9 nA
State change	0.261	0	0	0

Table 6.
Comparison of sneak current effect of the proposed architecture with other architectures.

	SRAM [37]	Memory [38]	Memory [16]	4M1M [15]	2M1M
Density (Gbt/cm^2)	0.338	1.6	—	50	80
Energy (fj/bit)	28.4	—	0.011	2.5×10^{-4}	23.2×10^{-9}

Table 7.
Comparisons of density and energy consumption with previous works.

By providing the structure and strategies for array-based 1S1R [16–22], many of the structures have offered while having high density $4F^2$, very small sneak paths current, very low power consumption and high read margin that is very promising. Compared with 2M1M structure, can be said that 1S1R based structures has been created in series connection memory element and selector in terms of manufacturing technology because are of the two different types perhaps compared with 2M1M structure which is a memristor uniform structure be more complexity. And in the 2M1M structure used of memristor, that is a memory and a computing element. The aim is to implementation the logic and computing capabilities for future applications of this structure in memory which can help to achieve a beyond classical von Neumann architecture. It hopes that by development and progression of 2M1M, the valuable feature in 1S1R structure is achieved for a higher density and removes sneak paths. Approximated device density and power consumption of the proposed architecture is compared with previous works in **Table 7**. As it is attainable form this table, due to lower number of memristors per memory cell, proposed architecture offers higher density compared with previous works. In terms of power consumption, since authors did not find a clear explanation regarding details of previous studies for their power calculations, power consumption of the cells in various operations are presented and compared in details.

5. Conclusions

In summary, this chapter discusses the resistive switching based composite memory cells and offers a solution toward the limitations within the current state-of-art 0T1R fully passive arrays and 1T1R active arrays to implement more efficient compute-in-memory structure for future beyond von-Neumann computing architectures. The first section of this chapter briefly review different resistive switching based composite memory arrays and discusses their advantages and limitations toward compute-in-memory applications and implementations. The next section, define a 2M1M memory array cell and analyzes its switching characteristics and the write and read operation principles within the crossbar structure. The final section of the chapter discusses the logic application with 2M1M switch and its capability to implement AND, NAND, OR, and NOR logic gates within 2M1M memory array structure and its compute-in-memory feature. Also, this section discusses the problem of sneakpath within the composite memory arrays and 2M1M array structure. We hope this chapter provide a good basis toward development of resistive switching based composite memory array platforms and providing a good insight over 2M1M structural benefits for compute-in-memory applications.

Author details

Mehri Teimoory[1], Amirali Amirsoleimani[2], Arash Ahmadi[3] and Majid Ahmadi[4]*

1 Department of Electrical Engineering, Razi University, Kermanshah, Iran

2 Department of Electrical Engineering and Computer Science, Lassonde School of Engineering, York University, Toronto, ON, Canada

3 Department of Electronics, Carleton University, Ottawa, ON, Canada

4 Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON, Canada

*Address all correspondence to: ahmadi@uwindsor.ca

References

[1] International Technology Roadmap for Semiconductors. Available online at: http://www.itrs.net (Nov. 2015).

[2] Ducharme S, Reece TJ, Othon CM, Rannow RK. Ferroelectric polymer Langmuir-Blodgett films for nonvolatile memory applications. IEEE Transactions on Device and Materials Reliability. 2005 Dec;**5**(4):720-735

[3] Li H, Chen Y. An overview of non-volatile memory technology and the implication for tools and architectures. In2009 design, Automation & Test in Europe Conference & Exhibition 2009 Apr 20 (pp. 731-736). IEEE.

[4] Chua L. Resistance switching memories are memristors. Appl. Phys. 2011; A, 102(4):765–783.

[5] Chua L. Memristor-the missing circuit element. IEEE Transactions on Circuit Theory. 1971 Sep;**18**(5):507-519

[6] Strukov DB, Snider GS, Stewart DR, Williams RS. The missing memristor found. Nature. 2008 May;**453**(7191):80-83

[7] Amirsoleimani A, Alibart F, Yon V, Xu J, Pazhouhandeh MR, Ecoffey S, et al. In-memory vector-matrix multiplication in monolithic complementary metal–oxide–semiconductor-memristor integrated circuits: Design choices, challenges, and perspectives. Advanced Intelligent Systems. 2020 Nov;**2**(11): 2000115

[8] Cai F, Correll JM, Lee SH, Lim Y, Bothra V, Zhang Z, et al. A fully integrated reprogrammable memristor–CMOS system for efficient multiply–accumulate operations. Nature Electronics. 2019 Jul;**2**(7):290-299

[9] Teimoori M, Amirsoleimani A, Ahmadi A, Ahmadi M. A 2M1M crossbar architecture: Memory. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2018 Feb 14;**26**(12):2608-2618

[10] Amirsoleimani A, Ahmadi M, Ahmadi A. Logic design on mirrored memristive crossbars. IEEE Transactions on Circuits and Systems II: Express Briefs. 2017 Jul 19;**65**(11): 1688-1692

[11] Rahimi Azghadi M, Chen YC, Eshraghian JK, Chen J, Lin CY, Amirsoleimani A, et al. Complementary metal-oxide semiconductor and memristive hardware for neuromorphic computing. Advanced Intelligent Systems. 2020 May;**2**(5):1900189

[12] Williams RS. How we found the missing memristor. IEEE Spectrum, vol. 45. No. 2008;12:28-35.

[13] Kim S, Jeong HY, Kim SK, Choi SY, Lee KJ. Flexible memristive memory array on plastic substrates. Nano Letters. 2011 Dec 14;**11**(12):5438-5442

[14] Zangeneh M, Joshi A. Design and optimization of nonvolatile multibit 1T1R resistive RAM. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2013 Sep 10;**22**(8):1815-1828

[15] Zhang Y, Shen Y, Wang X, Cao L. A novel design for memristor-based logic switch and crossbar circuits. IEEE Transactions on Circuits and Systems I: Regular Papers. 2015 May;**62**(5): 1402-1411

[16] Zhou J, Kim KH, Lu W. Crossbar RRAM arrays: Selector device requirements during read operation. IEEE Transactions on Electron Devices. 2014 Mar 25;**61**(5):1369-1376

[17] Huang JJ, Tseng YM, Hsu CW, Hou TH. Bipolar nonlinear Ni/TiO_2/Ni selector for 1S1R crossbar array applications. IEEE Electron

Device Letters. 2011 Aug 1;**32**(10): 1427-1429

[18] Huang JJ, Tseng YM, Luo WC, Hsu CW, Hou TH. One selector-one resistor (1S1R) crossbar array for high-density flexible memory applications. In 2011 international electron devices meeting 2011 Dec 5 (pp. 31-7). IEEE.

[19] Shin J, Kim I, Biju KP, Jo M, Park J, Lee J, Jung S, Lee W, Kim S, Park S, Hwang H. TiO 2-based metal-insulator-metal selection device for bipolar resistive random access memory cross-point application. Journal of Applied Physics. 2011 Feb 1;109(3):033712.

[20] Zhang L, Cosemans S, Wouters DJ, Groeseneken G, Jurczak M, Govoreanu B. Selector design considerations and requirements for 1S1R RRAM crossbar array. In2014 IEEE 6th international memory workshop (IMW) 2014 may 18 (pp. 1-4). IEEE.

[21] Zhang L, Cosemans S, Wouters DJ, Groeseneken G, Jurczak M, Govoreanu B. One-selector one-resistor cross-point array with threshold switching selector. IEEE Transactions on Electron Devices. 2015 Aug 11;**62**(10):3250-3257

[22] Jo SH, Kumar T, Narayanan S, Lu WD, Nazarian H. 3D-stackable crossbar resistive memory based on field assisted superlinear threshold (FAST) selector. In2014 IEEE international electron devices meeting 2014 Dec 15 (pp. 6-7). IEEE.

[23] Srinivasan VS, Chopra S, Karkare P, Bafna P, Lashkare S, Kumbhare P, et al. Punchthrough-diode-based bipolar RRAM selector by Si epitaxy. IEEE Electron Device Letters. 2012 Aug 24; **33**(10):1396-1398

[24] Junsangsri P, Lombardi F. Design of a hybrid memory cell using memristance and ambipolarity. IEEE Transactions on Nanotechnology. 2012 Nov 22;**12**(1):71-80

[25] Kvatinsky S, Wald N, Satat G, Kolodny A, Weiser UC, Friedman EG. MRL—Memristor ratioed logic. In2012 13th international workshop on cellular nanoscale networks and their applications 2012 Aug 29 (pp. 1-6). IEEE.

[26] Kvatinsky S, Belousov D, Liman S, Satat G, Wald N, Friedman EG, et al. MAGIC—Memristor-aided logic. IEEE Transactions on Circuits and Systems II: Express Briefs. 2014 Sep 11;**61**(11): 895-899

[27] Kvatinsky S, Satat G, Wald N, Friedman EG, Kolodny A, Weiser UC. Memristor-based material implication (IMPLY) logic: Design principles and methodologies. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2013 Oct 2;**22**(10):2054-2066

[28] Siemon A, Menzel S, Marchewka A, Nishi Y, Waser R, Linn E. Simulation of TaO x-based complementary resistive switches by a physics-based memristive model. In2014 IEEE international symposium on circuits and systems (ISCAS) 2014 Jun 1 (pp. 1420-1423). IEEE.

[29] Amirsoleimani A, Shamsi J, Ahmadi M, Ahmadi A, Alirezaee S, Mohammadi K, et al. Accurate charge transport model for nanoionic memristive devices. Microelectronics Journal. 2017 Jul 1;**65**:49-57

[30] Biolek Z, Biolek D, Biolkova V. SPICE model of memristor with nonlinear dopant drift. Radioengineering. 2009 Jun;**1**:18(2)

[31] Eshraghian K, Cho KR, Kavehei O, Kang SK, Abbott D, Kang SM. Memristor MOS content addressable memory (MCAM): Hybrid architecture for future high performance search engines. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2010 May 24;**19**(8):1407-1417

[32] Ho Y, Huang GM, Li P. Dynamical properties and design analysis for

nonvolatile memristor memories. IEEE Transactions on Circuits and Systems I: Regular Papers. 2010 Oct 18;**58**(4): 724-736

[33] Vontobel PO, Robinett W, Kuekes PJ, Stewart DR, Straznicky J, Williams RS. Writing to and reading from a nano-scale crossbar memory based on memristors. Nanotechnology. 2009 Sep 25;**20**(42):425204

[34] Gao Y, Kavehei O, Ranasinghe DC, Al-Sarawi SF, Abbott D. Future large-scale memristive device crossbar arrays: Limits imposed by sneak-path currents on read operations. arXiv preprint arXiv:1507.02077. 2015 Jul 8.

[35] Zidan MA, Fahmy HA, Hussain MM, Salama KN. Memristor-based memory: The sneak paths problem and solutions. Microelectronics Journal. 2013 Feb 1;**44**(2):176-183

[36] Fei W, Yu H, Zhang W, Yeo KS. Design exploration of hybrid cmos and memristor circuit by new modified nodal analysis. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2011 May 5;**20**(6):1012-1025

[37] Maeda N, Komatsu S, Morimoto M, Tanaka K, Tsukamoto Y, Nii K, et al. A 0.41 μa standby leakage 32 kb embedded sram with low-voltage resume-standby utilizing all digital current comparator in 28 nm hkmg cmos. IEEE Journal of Solid-State Circuits. 2013 Jan 15;**48**(4): 917-923

[38] Lehtonen E, Poikonen JH, Laiho M, Kanerva P. Large-scale memristive associative memories. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2013 Apr 30;**22**(3):562-574