Simulation and analysis of a neuromorphic architecture for in-memory computing

Author: Mar Puigibert Pérez
Facultat de Física, Universitat de Barcelona, Diagonal 645, 08028 Barcelona, Spain.

Advisor: Albert Cirera Hernandez

Abstract: Memristor based systems are proposed to break away from the classical von Neumann architecture and emulate neuromorphic behavior, aiming to address the energy consumption challenges posed by current computing systems. Modeling a memristor based on experimental measurements enables the simulation and analysis of NOT and NOR gates based on NMOS-like RRAM architecture. This study delves into said device's properties, possible systems implemented with it including in-memory computation, and analyses of energy efficiency.

I. INTRODUCTION

The escalating energy consumption of contemporary computing systems, particularly in the realm of artificial intelligence (AI) applications, presents a significant sustainability challenge. Despite our focus on functionality metrics such as speed, accuracy, and parallel processing capabilities, the environmental repercussions of these energy-intensive systems are often overlooked. Notably, data centers, integral to cloud-based systems, currently consume around 200 terawatt hours annually, with a projected tenfold increase by 2030. The burgeoning demand for computing power, outpacing improvements through Moore's law scaling, underscores the urgency of addressing this energy predicament [1].

A critical facet contributing to this energy challenge is the classical von Neumann architecture, wherein computing and memory units operate separately. The transfer of instruction codes and data is facilitated through buses connecting the different units. This configuration necessitates constant data movement, resulting in increased energy consumption and time delays, commonly termed the 'von Neumann bottleneck' [2]. To address the challenges posed by the von Neumann bottleneck more effectively, a promising solution is brain-inspired neuromorphic computing. Research aims to replicate the structure and functioning of biological neural networks, potentially involving the co-location of storage and computing, mirroring the configuration observed in the brain with synapses and neurons [1].

In the pursuit of replicating such behavior, a key electronic component known as the memristor or Resistive Random Access Memory (RRAM), recognized as the fourth elementary passive element by Leo Chua [3], is utilized [2]. The typical RRAM device consists of a resistive switching memory cell having a metal-oxide-metal structure. The RRAM cell undergoes a transition from a high-resistance state (HRS), representing logic value '0' or the OFF state, to a low-resistance state (LRS), representing logic value '1' or the ON state, and vice versa, through the application of an external voltage pulse. This transition is attributed to the resistive switching (RS) phenomenon within the RRAM cell. Initially, the RRAM is in its HRS, and to switch it to its LRS, a high voltage pulse is applied, leading to the formation of conductive paths in the switching layer, a process known as electroforming. This occurs at the forming voltage (V$_F$), dependent on cell area and oxide thickness. To switch the RRAM cell from LRS to HRS, a reset voltage (V$_{RESET}$) is applied in the 'reset' process. The transition from HRS to LRS is initiated by applying a set voltage (V$_{SET}$) in the 'set' process. So that it exhibits hysteresis in the current-voltage curve [4].

Compared to other Random Access Memory (RAM) technologies such as Dynamic Random-Access Memory (DRAM) and Static Random-Access Memory (SRAM), the key distinction lies in their memory volatility, meaning data loss upon power supply removal. DRAM boasts high capacity and density but requires frequent refresh cycles, leading to increased energy consumption. SRAM offers speed but shares volatility concerns with DRAM, compounded by larger cell sizes hindering large-scale implementation. These technologies rely on charge storage to store data. Instead RRAM relies on resistivity changes of its cells, which is conserved even after power is removed. Besides RRAM have demonstrated notable performance features in published experimental results for research device prototypes. These features encompass rapid switching speed, high endurance and data retention (non-volatile memory), high integration density, compatibility with complementary metal-oxide-semiconductor (CMOS) technology and low power consumption [2].

RRAM devices are interesting for excelling in these characteristics, allowing it to be not only a device that stores information but also used as a storage and computing element, meaning there can be in-memory computation (CiM) [5]. Various architectures of RRAM-based logic gates are suggested for applications involving CiM in section III.

This study performs simulations and analyses of NOT and NOR logic gates based on NMOS-like RRAM architecture, using the modeling of an RRAM device based on experimental data. The aim is to verify that the device performs computation in-memory and that the energy required for in-memory computation is sufficiently low to address the energy problem.

II. RRAM MODELING

To model the RRAM device, we rely on a model proposed by HP [3]. It involves a threshold-type switching model of a two-terminal voltage-controlled electrical device. The overall behavior follows the equations [6]:

$$I(t) = G(L,t)V_M \tag{1}$$

Where G is RRAM conductance (memductance), I represents the flowing current, V$_M$ the applied voltage and L is a parameter denoting the tunnel barrier width. In the circuit model, there are two resistors in series: one representing the doped layer (R) which is conductive and the other representing tunneling through the undoped layer (R$_t$) which is an insulator. Because of the $R_t \gg R$, the model focuses mainly on R$_t$, expected to be proportional to a variation of L with no significant error implication: $L_{V_M,t}$. The RRAM resistance (memristance) for a restricted range of L, is described by equation [6]:

$$R_t(L_{V_M,t}) = f_0 \frac{e^{2L_{V_M,t}}}{L_{V_M,t}} \tag{2}$$

Where f$_0$ is a model-fitting constant parameter which includes material and geometrical unknown issues. So the switching effect is mainly caused by fitting the effective distance for tunneling. The expected response of L depending on the time and V$_M$, follows the equation [6]:

$$L(V_M,t) = L_0 \cdot \left(1 - \frac{m}{r(V_M,t)}\right) \tag{3}$$

The voltage-dependent parameter r(V$_M$,t) and the fitting constant parameter m, determine the L boundaries. There are two boundary values: $r = r_{MIN}$, corresponding to L_{MIN} in which memristance is set to the lowest resistive state (R_{ON}), and $r = r_{MAX}$, corresponding to $L_{MAX} \approx L_0$ in which memristance is set to the lowest conductive state (R_{OFF}). A larger L_0 enlarges exponentially the memristance range. Because L can not be zero, the relation between m and r_{MIN} must be $\frac{m}{r_{MIN}} < 1$.

This model is based on the assumption that the switching rate between L_{MIN} and L_{MAX} is fast above the thresholds voltages V_{SET} and V_{RESET}. This condition is integrated into the r(V$_M$,t) derivative equation [6]:

$$\dot{r}(V_M,t) = \begin{cases} \frac{\alpha\cdot(V_M+V_{RESET})}{\gamma+|V_M+V_{RESET}|} & , V_M \in [-V_0, V_{RESET}) \\ \beta \cdot V_M & , V_M \in [V_{RESET}, V_{SET}] \\ \frac{\alpha\cdot(V_M-V_{SET})}{\gamma+|V_M-V_{SET}|} & , V_M \in (V_{SET}, V_0] \end{cases} \tag{4}$$

The parameters α, β and γ are fitting constants for modeling non-linear threshold-based behaviour to shape the rate of memristance change, with $\alpha \gg \beta$ and $\gamma \in (0,1]$ [6]. Therefore the set of parameters α, β, γ and m defines the boundaries for the L. All these parameters are fitted later in this section.

To model the described RRAM device using a SPICE circuit, the memristive system is implemented as a sub-circuit. The sub-circuit combines a current source G_r, a capacitor C_r, an auxiliary resistor R_{aux} and an additional current source G_{pm}. G_r generates a current based on equation (4). C_r models the memory effect and is specified with an initial condition. Choosing the initial voltage value across the capacitor (r_{init}) as one of the boundary values, or any valid value in between, signifies the starting state of the device. In this study, we choose HRS to be the starting state, so $r_{init} = r_{MAX}$. R_{aux} models the memory retention capability. G_{pm} represents top and bottom electrodes of the device and models a behavioral resistor following the equation (2). The voltage-controlled memristor SPICE model netlist is complete in [6].

To calibrate a memristor based on this model, we start with the collection of experimental data from a commercial memristor from Knowm [7]. We seek a measure of the hysteresis cycle by applying a current source with an amplitude of $2\cdot10^{-6}$ A and a frequency of 10 Hz to determine the voltage range of the device and its behaviour. This way, we determine the V$_{RESET}$ and V$_{SET}$ values.

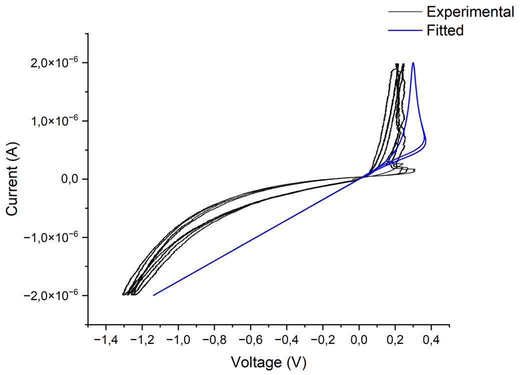

FIG. 1: Experimental data of the current as a function of voltage for the Knowm memristor compared to the fitted data.

A second measure is an Idle-SET-Idle-READ-Idle-RESET-Idle-READ sequence, applying a current pulse with I$_{SET}$ = $2\cdot10^{-6}$ A, I$_{RESET}$ = $-2\cdot10^{-6}$ A, and I$_{READ}$ = $0.1\cdot10^{-6}$ A. This sequence provides a more detailed understanding of the voltage across the memristor when in the LRS and HRS states. Applying a read current I$_{READ}$ to the device helps differentiate between the states. This current should be very small to avoid forcing a state change, as we are interested in verifying that the device retains the programmed state until there is no forced change by I$_{SET}$ or I$_{RESET}$.

Fitting the model parameters to these experimental data, the parameters take the following values: $r_{MIN} = 130$, $r_{MAX} = 175$, $r_{INIT} = 175$, $\alpha = 2 \cdot 10^5$, $\beta = 10^3$, $\gamma = 1$, $V_{SET} = 0.3$ V, $V_{RESET} = -1.3$ V, $y_0 = 0.1$, m= 85, $f_0 = 7500$ and $L_0 = 5$.

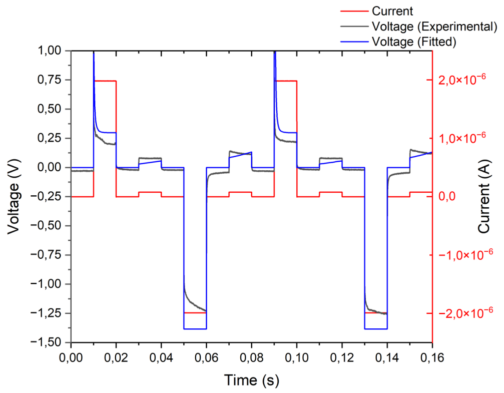

FIG. 2: Experimental data of the Idle-SET-Idle-READ-Idle-RESET-Idle-READ sequence for the memristor Knowm compared to fitted data.

III. SIMULATION AND ANALYSIS OF NOT AND NOR LOGICAL GATES

Researchers discovered that the RRAM device can serve as a valuable logic device. Various families of RRAM-based logic gates, which can be array-implementable, were suggested for applications involving CiM. Some examples are: IMPLY (material implication), CRS (complementary resistive switches), MAGIC (memristor aided logic) and MPLA (memristive programmable logic array). The IMPLY logic circuits require a high count of cycles and a high count of RRAM devices. Because of the serialization of its operations, these logic circuits are the slowest. The CRS gates, while being logically complete and suitable for array implementation, rely on specialized complementary RRAM cells for their realization. It is essential to acknowledge that CRS circuits are susceptible to destructive reads. The MAGIC gates are made by using the RRAM devices in series and/or in parallel, but only the NOR and NOT gates can be implemented in the RRAM array. And the MPLA logic is incomplete [4].

In contrast to other RRAM-based logic gate, the CMOS-like RRAM gates are similar to CMOS logic gates. The CMOS-like RRAM gates require only two cycles per computation, highlighting their high performance. RRAM devices are typically arranged in arrays, however CMOS-like RRAM gates can not be implemented in arrays. This limitation implies that they are not suitable for CiM. This section aims to overcome the mentioned limitation while maintaining the high performance features of CMOS-like RRAM gates. Our strategy involves the introduction of NMOS-like RRAM-based gates. NMOS-like RRAM gates are faster than IMPLY gates, consume less RRAM cells than the other RRAM gate families and they are logic complete [4]. In this section we simulate and analyze the NMOS-like RRAM NOT and NOR gates. Notice that to achieve complete logic functionality, an additional AND logic gate would be required.

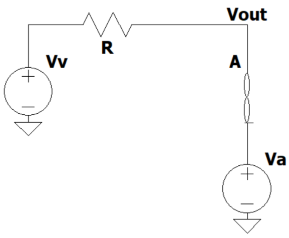

FIG. 3: NMOS-like NOT gate setup in SPICE.

In order to implement the NOT logic gate, the following setup in SPICE is required (Figure 3). In this circuit we dispose of one RRAM cell (memristor A) connected in series with a resistor (R) whose value lies in between the memristance's HRS and LRS values [4]. Because the memristance's HRS and LRS values of the fitted model are respectively $R_{OFF} = 505 \cdot 10^3\,\Omega$ and $R_{ON} = 135 \cdot 10^3\,\Omega$, we choose an average resistor value $R = (R_{ON} + R_{OFF})/2 = 320 \cdot 10^3\,\Omega$. The output node of this circuit would be the output of the resulting voltage divider. The ends of this circuit are connected to two voltage sources. The first one, which is the one connected to the resistor (V$_v$), would operate in the manner of a clock. The other one (V$_a$) would represent the logic input to the gate. The memristor's set end, marked with the black line, needs to be directly connected to this second source. The gate circuit works in two cycles:

Input or writing cycle. It writes the logic input into the RRAM cells using the input voltage (V$_a$). When we want to write the logic input 1, the top end of the circuit is connected to ground (V$_v$), and the voltage V$_a$ is applied to the set end, where V$_a$ is defined by the condition $V_a > \max(|V_{HRS}|, V_{LRS})$ [4]. Applied to our fitted model V_a >max$(|-1.3|, 0.3)$ V, we finally chose $V_a = 4$ V so that the voltage drop across the memristor is large enough to trigger a change of state to LRS (logic 1). In case we want to write the logic input 0, the top end of the corresponding cell is also connected to ground, while the voltage -V$_a$ is applied to the bottom end.

Computation or execution cycle. Once we've written onto the device, the voltage V$_v$ is applied following the condition V_v <min($|V_{HRS}|, V_{LRS}$) [4]. Applied to our fitted model V_v <min($|-1.3|, 0.3$) V, choosing $V_v = 0.2$ V. Meanwhile V$_a$ is connected to ground. It is in this phase when the output of the gate is generated by the voltage divisor between the RRAM device and the resistor. Considering $R \gg R_{ON}$, when the logic input is 1, it results in a logic output of 0, demonstrated using the voltage divider equation:

$$V_{out,A} = V_v \cdot \frac{R_{ON}}{R_{ON} + R} \to 0 \tag{5}$$

Considering $R_{OFF} \gg R$, when the logic input is 0, it results in a logic output of 1. In terms of V_{out}:

$$V_{out,A} = V_v \cdot \frac{R_{OFF}}{R_{OFF} + R} \to V_v \tag{6}$$

The fitted model does not exhibit a significant difference in the memristances as they are of the same order of magnitude. Due to this small difference, the output voltage is low when the logical input of memristor A is 1 but does not approach to zero as in the limit case. Using the equation (5), $V_{out,A=1} = 0.06$ V. Likewise, for the opposite case the output voltage is higher but it is not equal to the limit case. Using the equation (6), $V_{out,A=0} = 0.12$ V (Figure 4).

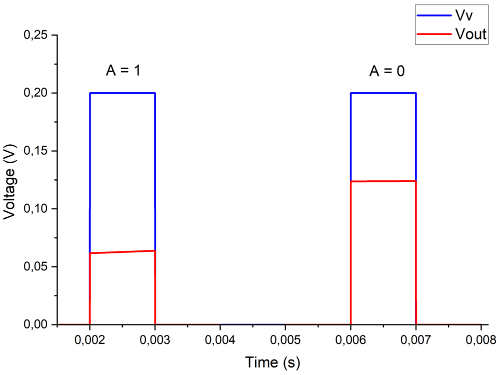

FIG. 4: NMOS-like NOT gate SPICE simulation of V_v and V_{out}, corresponding to the execution cycle after writing a logic input of 1 and a logic input of 0 in memristor A.

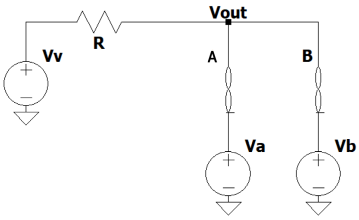

FIG. 5: NMOS-like NOR gate setup in SPICE.

In order to implement the NOR logic gate in SPICE, we must connect two RRAM devices (memristor A and memristor B) in parallel as shown in figure 5, with one additional voltage source (V$_b$) to represent the additional logic input to the gate [4]. Its working principle is the same as the one in the NOT gate, only that its equivalent memristance value ($R_{||}$) would be that of the parallel of the individual memristances values (R_A and R_B):

$$R_{||} = \left(\frac{1}{R_A} + \frac{1}{R_B}\right)^{-1} \tag{7}$$

To determine the value of R we take into account the four possible combinations of the logic input and the equation (7). When the $R_A = R_B = R_{ON}$, the equivalent is $R_{||} = R_{ON}/2 = 68 \cdot 10^3\,\Omega$. When $R_A = R_B = R_{OFF}$, the equivalent is $R_{||} = R_{OFF}/2 = 253 \cdot 10^3\,\Omega$. On the case $R_A = R_{ON}$ and $R_B = R_{OFF}$ and when the $R_B = R_{ON}$ and $R_A = R_{OFF}$, the equivalent memristance is $R_{||} = 107 \cdot 10^3\,\Omega \approx R_{ON}$. So, calculating the average of these $R_{||}$ values, the value of the resistor is R $= 134 \cdot 10^3\,\Omega$. The output voltage on the NOR gate is, in general terms:

$$V_{out,A,B} = V_v \cdot \frac{R_{||}}{R_{||} + R} \tag{8}$$

Applying equation (8) for each combination and considering the NOR resistor value, we have the cases $V_{out,A=B=1} = 0.07$ V and $V_{out,A\neq B} = 0.09$ V approaching to zero as a logical output 0, and the case $V_{out,A=B=0} = 0.13$ V, approaching to a higher value as a logical output 1 (Figure 6).

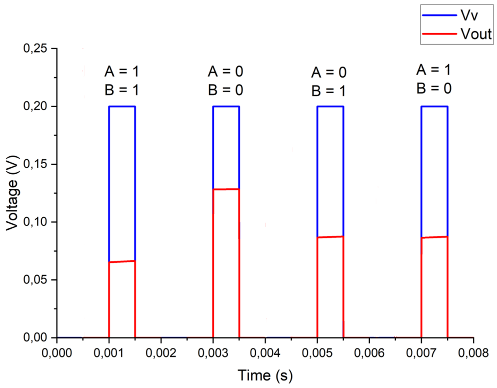

FIG. 6: NMOS-like NOT gate SPICE simulation of V_v and V_{out}, corresponding to the execution cycle after writing the different combinations of logic input in memristors A and B.

In both the NOT and NOR logic gates, it has been verified through a SPICE simulation that there was computation in memory. The verification was done by adding a small read voltage pulse, $V_{\text{read}} = 0.1$ V, not large enough to force a state change, just after each cycle (writing and execution). It was observed that the state of the output voltage after the writing was preserved after execution, so the execution cycle does not affect the written logical value.

In a final simulation and calculation with SPICE, we determine the minimum energy required for an in-memory computation (E$_{CiM}$), the results are in tables I and II. The dissipated Joule energy is $E = P \cdot t$, where P is the dissipated power obtained in the simulation and t the simulation time. For the NOT gate, the minimum reasonable time is t = $1 \cdot 10^{-3}$ s, according to Knowm [7], and P is composed of the dissipated power in the resistor (R) and the memristor (A) for each writing (WC) and executing (EC) cycle. So, the E$_{CiM}$ for the NOT gate is:

$$E_{CiM} = E_{WC}+E_{EC} = E_{WC,R}+E_{WC,A}+E_{EC,R}+E_{EC,A} \quad (9)$$

A	Out	E$_{WC}$ (J)	E$_{EC}$ (J)	E$_{CiM}$ (J)
1	0	$3.51 \cdot 10^{-8}$	$8.55 \cdot 10^{-11}$	$3.52 \cdot 10^{-8}$
0	1	$2.27 \cdot 10^{-8}$	$4.77 \cdot 10^{-11}$	$2.27 \cdot 10^{-8}$

TABLE I: In-memory computing energy (E$_{CiM}$) following the equation (9), for the possible logical inputs of the NOT gate.

For the NOR logic gate, the procedure is the same, but adding the contribution of memristor B in both cycles. The reasonable minimum time for this simulation also changes, and it is t = $0.5 \cdot 10^{-3}$ s.

A	B	Out	E$_{WC}$ (J)	E$_{EC}$ (J)	E$_{CiM}$ (J)
1	1	0	$3.34 \cdot 10^{-8}$	$9.80 \cdot 10^{-11}$	$3.35 \cdot 10^{-8}$
0	0	1	$6.16 \cdot 10^{-8}$	$8.22 \cdot 10^{-11}$	$6.17 \cdot 10^{-8}$
0	1	0	$5.96 \cdot 10^{-8}$	$8.19 \cdot 10^{-11}$	$5.97 \cdot 10^{-8}$
1	0	0	$2.50 \cdot 10^{-8}$	$5.09 \cdot 10^{-11}$	$2.51 \cdot 10^{-8}$

TABLE II: In-memory computing energy (E$_{CiM}$), for the possible logical inputs of the NOR gate.

IV. CONCLUSIONS

In conclusion, the adjustment of the RRAM device model described in section II has been presented based on experimental data collected from a commercial Knowm [7] memristor. The adjustment is considered good as it adequately approximated the experimental values in both the hysteresis cycle and the Idle-SET-READ-Idle-RESET-Idle-READ sequence. Furthermore, the parameters have met the conditions described earlier. The successful model adjustment has enabled the simulation and analysis of the device's behavior in NMOS-like RRAM gates, relying on experimental data to distinguish it from the article [4].

It has been demonstrated that the logic gates operate, although there is one point to note. The difference in memristance between R_{ON} and R_{OFF} is not three orders of magnitude, as in [4], but only one order of magnitude. So, the output voltage values did not approximate as closely to the expected values from the voltage divider equation. This may be because of the model is based on a commercial memristor, not a research one. Therefore, they might not be optimal in terms of the difference between HRS and LRS states.

Apart from the successful simulations, it has been demonstrated that the device CiM, works in very short timescales (on the order of 10^{-3} s) and requires very low energy to perform CiM (on the order of 10^{-8} J). Therefore, despite being a commercial device, it continues to excel in properties such as switching speed, endurance, and data retention. Thus, these devices are promising candidates for addressing the energy challenge, breaking away from the classical von Neumann architecture.

Acknowledgments

I extend my deepest appreciation to my advisor Albert Cirera for their guidance, support and feedback throughout the entire research process. I would like to express my sincere gratitude to all those who have contributed to the completion of this work. Specially to my family, friends and partner for their encouragement and patience.

[1] Adnan Mehonic and Anthony J Kenyon. Brain-inspired computing needs a master plan. *Nature*, 604(7905):255–260, 2022.

[2] Furqan Zahoor, Tun Zainal Azni Zulkifli, and Farooq Ahmad Khanday. Resistive random access memory (rram): an overview of materials, switching mechanism, performance, multilevel cell (mlc) storage, modeling, and applications. *Nanoscale research letters*, 15:1–26, 2020.

[3] Zdeněk Biolek, Dalibor Biolek, and Viera Biolkova. Spice model of memristor with nonlinear dopant drift. *Radioengineering*, 18(2), 2009.

[4] Xiaole Cui, Ye Ma, Feng Wei, and Xiaoxin Cui. The synthesis method of logic circuits based on the nmos-like rram gates. *IEEE Access*, 9:54466–54477, 2020.

[5] Yangyin Chen. Reram: History, status, and future. *IEEE Transactions on Electron Devices*, 67(4):1420–1433, 2020.

[6] Ioannis Vourkas and Georgios Ch Sirakoulis. *Memristor-based nanoelectronic computing circuits and architectures*, volume 19. Springer, 2016.

[7] Albert Cirera, Blas Garrido, Antonio Rubio, and Ioannis Vourkas. Current driven random exploration of resistive switching devices, an opportunity to improve bit error ratio. In *2023 14th Spanish Conference on Electron Devices (CDE)*, pages 1–4, 2023.