Int. Journ. of Unconventional Computing, Vol. 18, pp. 249–267
Reprints available directly from the publisher
Photocopying permitted by license only

©2023 Old City Publishing, Inc.
Published by license under the OCP Science imprint,
a member of the Old City Publishing Group

Design Exploration of Threshold Logic in Memory and Experimental Implementation Using Knowm Memristors

CARLOS FERNANDEZ[1], ALBERT CIRERA[2] AND IOANNIS VOURKAS[*]

[1]*Department of Electronic Engineering, Universidad Técnica Federico Santa María, 2390123, Valparaíso, Chile*
[2]*MIND-IN2UB, Department of Electronic and Biomedical Engineering, Universitat de Barcelona, E-08028 Barcelona, Spain*

Received: April 24, 2023. Accepted: April 28, 2023.

In the context of computing in resistive memories (ReRAM), the implementation of threshold logic gates (TLGs) with memristors has been especially investigated for programable in-memory logic implementations. In this work we focus on two circuit design concepts for threshold logic with memristors: the voltage divider and the voltage adder. We provide an analytical exploration of the impact of the HRS-to-LRS ratio of memristors on the performance of every circuit and highlight the advantages of the voltage adder approach. Through experimental results from a proof-of-concept circuit, implemented using self-directed channel (SDC) memristors by *Knowm Inc*., we validated the conclusions derived from the theoretical analysis for computational memory structures supporting threshold logic gates. Moreover, we present the design and simulation of a bitline driver and sensing circuitry for a 1T1R array, where the proposed threshold logic scheme can seamlessly fit, thus providing solutions towards the development of robust and reliable computational ReRAM modules.

Keywords: Memristor, resistive switching, resistive RAM, ReRAM, in-memory computing, NOR logic, Knowm, SDC memristors

*Corresponding author: ioannis.vourkas@usm.cl

1 INTRODUCTION

Memory-centric computing in resistive memories (ReRAM) has drawn a lot of attention during the last decade [1], [2]. Resistive switching devices (RS devices or "memristors" [3]) are considered among the key emerging technologies for future computational memories that are expected to significantly improve the performance of computing platforms by lowering the requirement for data movement outside the memory modules [4]. In ReRAM arrays, the data are represented by resistance; high resistive state (HRS) for logic '0' and low resistive state (LRS) for logic '1', or vice versa. During memory WRITE operations, a transition from a HRS to a LRS is commonly called as SET, and the opposite as RESET. Resistance switching is achieved via the application of a voltage that exceeds the corresponding switching threshold ($V_{SET/RESET}$). Unlike conventional logic circuits based on transistors, computations inside a ReRAM module are performed by properly driving the word-lines and bit-lines which connect to the memristors that participate in the logic operations [5]. Many different logic design styles have been proposed for memristor devices, allowing computations to be performed either only inside the memory core (i.e., the crossbar array) or also at the periphery [6].

In this context, most recent approaches to such logic circuits rely on implementations that are based on memory READ operations; i.e., they use low-amplitude voltage pulses that do not modify the state of the memristors. Compared to other design styles of the literature [7], [8], when computations are based on READ operations, the conditional switching of any memristor during information processing is avoided. Such principle of computing with memristors is compatible with threshold logic gates (TLGs), which have been also investigated for programable logic implementations in ReRAM circuits [9]. There are many ways to incorporate memristors in TLG implementations [10], [11]. However, reliability of logic operations can be affected by the presence of variability in the switching performance of memristors, so it is important that variability (both device-to-device and cycle-to-cycle) is properly considered in the design of TLG circuits based on memristors.

In this work we focus on two circuit design concepts for threshold logic: i) a *ratioed threshold logic* circuit, based on a voltage divider, and ii) a *linear threshold logic* circuit, based on a voltage adder. We provide an analytical exploration of the impact of the HRS-to-LRS ratio (R_{HRS}/R_{LRS}) of memristors on the performance of every circuit, along with the identification of the most important design parameters to consider for computational ReRAM modules. Focusing on two-input logic operations, our analysis highlights the difficulties in the application of the voltage divider approach given the strong dependence of its operation on the memristors´ switching features, thus promoting the voltage adder as a better alternative to pursue for reliable threshold logic operations in ReRAM. Through experimental results from a linear threshold logic proof of concept circuit for a two-input NOR logic gate, implemented

using *off-the-shelf* components and self-directed channel (SDC) memristors [12], commercialized by *Knowm Inc*. [13], we validate the feasibility and all the key conclusions derived from the theoretical analysis for computational ReRAM structures supporting threshold logic gates. Finally, we present the design and simulation in LTSpice of a bitline driver and sensing circuitry for a 1T1R ReRAM topology where the *linear threshold logic* scheme can seamlessly fit using the voltage adder, thus providing solutions towards the development of robust and reliable computational ReRAM modules.

2 VOLTAGE DIVIDER & VOLTAGE ADDER FOR THRESHOLD LOGIC OPERATIONS

Following the classification of memristor-based logic design styles, presented in [14], a ratioed NOR gate can be implemented as shown in Fig. 1a. Such circuit consists of a voltage divider and a voltage comparator. The voltage divider is formed by a resistive element R_L and the equivalent resistance of the selected memristors which are connected in parallel (selector devices in series with the memristors are omitted from Fig. 1 for simplicity). The number of selected memristors equals the number of input operands and such logic design is compatible with the crossbar array topology [15]. The circuit

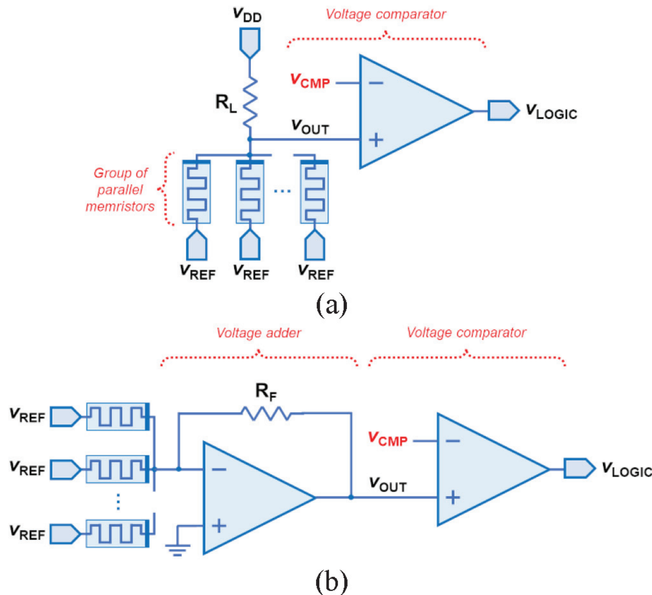

FIGURE 1
Generic implementation scheme of (a) *ratioed* threshold logic, and (b) *linear* threshold logic.

was first proposed in [16] and lately has been reported in other works related to in-memory computing targeting ReRAM architectures [17], [18].

During circuit operation, a voltage is applied across the voltage divider such that the voltage on the selected memristor(s) $|V_{OUT}-V_{REF}|$ stays below their switching thresholds ($V_{SET/RESET}$) to not disturb the stored information in the memory cells (thus equivalent to a READ operation). The output voltage V_{OUT} is described by Eq. (1), where R_{eq} is the equivalent resistance of the selected memristors.

$$V_{OUT} = V_{REF} + (V_{DD} - V_{REF}) \cdot \frac{R_{eq}}{R_L + R_{eq}} \tag{1}$$

Here we assume that high resistive state (HRS) represents logic '0' and low resistive state (LRS) represents logic '1'. If the resistive element exhibits a resistance $R_L \approx R_{LRS}$, when only one memristor is selected, this topology can be used for memory READ operations[1]; the output voltage V_{OUT} is compared with V_{CMP} to decode the information stored in the device as logic '1' or logic '0' (data I/O correspond to resistance/voltage)[2]. However, all logic operations are typically equivalent to memory READ operations too. For two-input logic operations: i) when inputs are "00" (both devices in R_{HRS}), V_{OUT} is roughly close to V_{DD}, which is interpreted as logic '1'; ii) if at least one input is logic '1' (R_{LRS}) then V_{OUT} is interpreted as logic '0'. By comparing V_{OUT} with V_{CMP} we produce the logic '0' or 'logic 1' digital output (V_{LOGIC}).

In such *ratioed threshold logic* design, the reliability of logic operations depends on the selection of V_{CMP}. We define as V_{OUT_1} (V_{OUT_0}) the output voltage given by Eq. (1) corresponding to logic '1' (logic '0'). Therefore, the greater the difference $|V_{OUT_1}-V_{OUT_0}|$, the easier it becomes to define the V_{CMP} value. The smallest margin is observed between the "00" and the "10"/"01" input combinations (for higher *fan-in* the smallest margin is observed between the "all zeros" input case and the case when only one of the memristors is in LRS). Moreover, such voltage margin improves when the devices exhibit a resistance ratio with $R_{HRS} >> R_{LRS}$.

An alternative implementation concept is the *linear threshold logic* which was previously discussed in [11] and more recently seen in [19], [20]. It requires that the memory READ and the logic operations are both based on a *voltage adder* instead of a *voltage divider*, as shown in Fig. 1b. Here, a READ voltage V_{REF} is applied to the input terminals of the selected memristors,

1 For HRS as logic '0' and LRS as logic '1', note that the logic output of such READ operations is complemented.

2 Two thresholds are required if an undefined region is assumed in-between the HRS and LRS states, thus a 2-bit A/D would serve in such case, where one of the two output bits would act as a flag to denote resistance value in the undefined region.

which are connected to the inverting input of an operational amplifier (opamp) in a *summing amplifier* configuration, which is placed before the comparator stage. Owing to the virtual ground of the opamp, the voltage drop on the terminals of memristors is V_{REF}, which is lower than their switching thresholds. V_{OUT} corresponds to the weighted sum of the applied input voltages, and is given by Eq. (2). R_{eq} is the equivalent resistance of the selected memristors and R_F is the value of the feedback resistor of the voltage adder circuit.

$$V_{OUT} = -V_{REF} \cdot \frac{R_F}{R_{eq}} \tag{2}$$

As in the case of the voltage divider, the same topology serves for typical memory READ operations in the ReRAM module. Thus, V_{OUT} reflects the result of both the memory READ and the logic operations. The V_{OUT} voltage increases with the number of memristors that are in LRS. For instance, if HRS represents logic ‘0’ and LRS represents logic ‘1’, then for two-input logic operations the “00” combination will produce a very small output voltage, whereas “01” (or equivalently “10”) will give a higher output voltage, and “11” will result in the highest V_{OUT} voltage. This is next compared with V_{CMP} to decode the resulting V_{OUT} voltage as logic ‘0’ or logic ‘1’ (V_{LOGIC}). Note that for memory READ operations we select only the target memristor to be read, and the circuit is converted into a transimpedance amplifier. Consequently, the level of the output voltage can be very low when the memristor is in HRS and thus susceptible to noise. In such case, two cells could instead be selected, the second being a “dummy” memory cell corresponding to a memristor in HRS, to increase V_{OUT} by a certain offset. It should be noted that in Fig. 1b it is $V_{OUT_1} < V_{OUT_0}$, thus the interpretation of logic ‘1’ and ‘0’ at the comparison stage will be reversed, compared to what was mentioned previously in the case of the voltage divider.

In both threshold logic implementations shown in Fig. 1, varying dynamically the comparison threshold V_{CMP} and selectively inverting the comparator´s output (V_{LOGIC}), provides flexibility and allows implementing different logic gates without modifying the rest of the circuit. This can be very beneficial for cascaded in-memory logic operations and the realization of complex logic operations, which were not explored further in this work.

3 PERFORMANCE COMPARISON OF THRESHOLD LOGIC DESIGNS FROM A RELIABILITY POINT OF VIEW

According to Eq. (1), the output voltage V_{OUT} for NOR *ratioed threshold logic* operations based on the voltage divider depends almost exclusively on R_{eq}, and thus on the $k = R_{HRS}/R_{LRS}$ ratio. Therefore, next we identify the influ-

ence that k has on the minimum output voltage window $|V_{OUT_1} - V_{OUT_0}|$ which determines the robustness and reliability of such operations. We assume two cases: (a) when $R_{HRS} >> R_{LRS}$, and (b) when such condition does not hold. From Eq. (1), assuming $R_L = R_{LRS}$, the equivalent resistance of the two parallel memristors for the "00" input combination is $R_{eq,00} = R_{HRS}/2$. Thus, the voltage at the logic output is given by Eq. (3):

$$V_{OUT_1} = \begin{cases} V_{DD},\ R_{HRS} >> R_{LRS} \\ V_{REF} + (V_{DD} - V_{REF}) \cdot \dfrac{1}{\frac{2}{k}+1},\ otherwise \end{cases} \quad (3)$$

The best (highest) and the worst (lowest) value of V_{OUT_1} is provided by Eq. (3). We note the dependence of the voltage on the resistance ratio k; the smaller it gets, the farther the V_{OUT_1} gets from V_{DD}. Likewise, we repeat the same analysis for the case of having one memristor in R_{LRS}, thus a "01"/ "10" input combination. The corresponding equivalent resistance $R_{eq,01}$ is calculated in Eq. (4).

$$R_{eq_01} = \begin{cases} R_{LRS},\ R_{HRS} >> R_{LRS} \\ \dfrac{R_{HRS}}{1+k},\ otherwise \end{cases} \quad (4)$$

Using Eq. (4), through Eq. (1) we compute the corresponding values for V_{OUT_0}, as shown in Eq. (5).

$$V_{OUT_0} = \begin{cases} \dfrac{1}{2}\left(V_{REF} + V_{DD}\right),\ R_{HRS} >> R_{LRS} \\ V_{REF} + (V_{DD} - V_{REF}) \cdot \dfrac{1}{\frac{k+1}{k}+1},\ otherwise \end{cases} \quad (5)$$

By observing Eq. (5), we notice that the resistance ratio k does not have as a significant impact on the difference between the highest and lowest possible value of V_{OUT_0}, as it happens in Eq. (3). We are mostly concerned about the minimum difference $|V_{OUT_1} - V_{OUT_0}|$, which occurs in case (b) when the value of k is small. As an example, applying $V_{DD} = 1V$ and $V_{REF} = 0.6V$, while assuming a ratio of just $k = 10$ we calculate a voltage window $\Delta V_{OUT} = V_{OUT_1} - V_{OUT_0} = 0.933V – 0.79V = 143mV$. Moreover, ΔV_{OUT} could get as low as 40mV for $k = 2$.

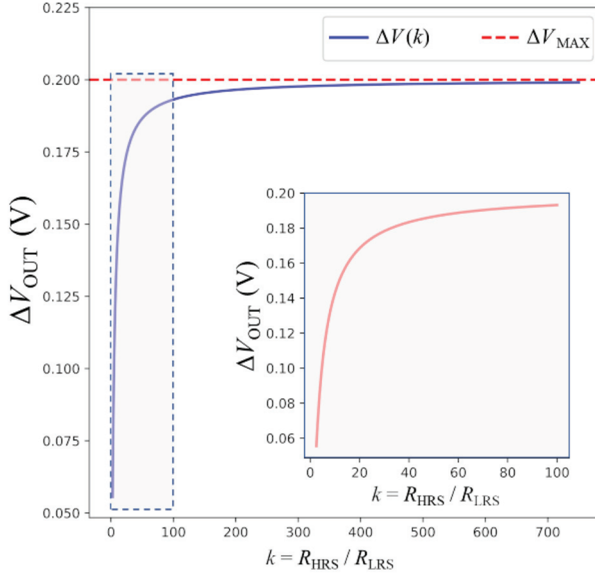

FIGURE 2
Evolution of $\Delta V_{OUT} = V_{OUT_1} - V_{OUT_0}$ for a wide range of values of the resistance ratio $k = R_{HRS}/R_{LRS}$, according to Eq. (1) (V_{DD} = 1V, V_{REF} = 0.6V). The horizontal (red) dashed line shows the maximum value of ΔV_{OUT}. The inset shows enlarged the part of the plot where it is $2 \leq k \leq 100$.

In order to better understand the trends and the dependence of ΔV_{OUT} on k, we present graphically in Fig. 2 the evolution of ΔV_{OUT} w.r.t. the value of k. We observe that ΔV_{OUT} approximates its maximum value $(V_{DD} - V_{REF})/2$ (equal to 200mV in our example) when k exceeds two orders of magnitude. On the other hand, for lower values of k, the voltage window rapidly gets narrow. Note that the absolute values of V_{OUT_1} and V_{OUT_0} could be modified using a different V_{REF}. However, such modification is constrained by the values of the switching thresholds of the memristors ($V_{SET/RESET}$). These observations highlight the difficulties in the design and implementation of the *ratioed* version of *threshold logic* circuits. The problem originates from the strong dependance of the output voltage on the memristor´s switching features, such as the resistance ratio k, which is affected by variability. Moreover, the value of V_{REF} is the only design parameter to consider in this case. On the other hand, the *linear threshold logic* circuit based on the voltage adder has R_F as an additional design parameter, whose value can be selected depending both on k and on the applied voltage V_{REF}.

Next, we compute the minimum voltage window $|\Delta V_{OUT}|$ for two-input NOR *linear threshold logic* operations to highlight the reliability gains offered by this alternative topology. From Eq. (2), the equivalent resistance of the two memristors in “00” input combination is $R_{eq,00} = R_{HRS}/2$ and at the logic gate output it is

$V_{OUT_1} = -V_{REF} \cdot (2 \cdot R_F / R_{HRS})$. Based on Eq. (4), from Eq. (2) we compute the value of the voltage V_{OUT_0} at the logic gate output, as shown in Eq. (6).

$$V_{OUT_0} = \begin{cases} -V_{REF} \cdot \frac{R_F}{R_{HRS}} \cdot k, \ R_{HRS} >> R_{LRS} \\ -V_{REF} \cdot \frac{R_F}{R_{HRS}} \cdot (k+1), \ otherwise \end{cases} \quad (6)$$

By observing Eq. (6) we figure out that the corresponding output voltage V_{OUT_0} is always affected by k. Likewise the voltage divider circuit, here small k values lead to a small ΔV_{OUT} voltage window. The narrower voltage window is given by $\Delta V_{OUT} = V_{OUT_0} - V_{OUT_1} = -V_{REF} \cdot (R_F/R_{HRS}) \cdot (k-2)$. The latter is a value proportional to both k and R_F, and thus it can be engineered appropriately. For example, assuming R_F = 50KOhm, R_{HRS} = 100KOhm, and V_{REF} = 100mV, then a resistance ratio of just $k = 10$ produces a ΔV_{OUT} of 400mV, whereas $k = 100$ results in a much higher ΔV_{OUT} of 4.9V. Such numbers are significantly larger than the maximum ΔV_{OUT} observed in the voltage divider configuration. For $k >> 1$, it can be $\Delta V_{OUT} \approx V_{OUT_0}$. On the other hand, for the same parameters, a resistance ratio as low as $k \approx 2$ leaves little space for voltage comparison, but this can be compensated using higher R_F values.

4 EXPERIMENTAL IMPLEMENTATION OF TWO-INPUT NOR GATE WITH A LINEAR THRESHOLD LOGIC CIRCUIT

Building upon the analyses presented in previous sections, here we present our results from the experimental validation of the voltage adder concept for two-input NOR *linear threshold logic* operations. In the experimental setup we used the digital oscilloscope, the function generator and digital output signals of the Digilent Analog Discovery 2 (AD2) instrumentation tool [21]. The memristors were discrete self-directed-channel (SDC) bipolar devices with tungsten (W) as dopant on a chalcogenide material [12], developed and commercialized in 16-pin ceramic DIP packages by *Knowm Inc*. [13]. The SDC devices are distinct from Conductive Bridge RAM (CBRAM) [22] devices and represent their own sub-class of electrochemical metallization memory (ECM) which, in response to an applied voltage, use a metal-catalyzed reaction within the active layer to generate Ag ion transport routes that contain Ag agglomeration sites, permanent under similar operating conditions.

For the forming of the pristine memristors and the posterior evaluation of cycling (SET-RESET) performance, every memristor was first individu-

ally connected to the inverting input of an OP97 opamp in a *transimpedance amplifier* topology with a 47KOhm feedback resistor [23]. A positive voltage ramp was applied, rising from 0V to 1V in 100us, while limiting the maximum current at 90uA, as described in [24]. If forming was unsuccessful in the first attempt, the process was repeated and the ramp duration was gradually increased. Once the devices were formed successfully, triangular wave voltage was applied to normalize the cycling performance of every device. In Fig. 3 we observe the cycling behavior of one of the memristors during 50 cycles under symmetric triangular voltage of |1V| amplitude and 10ms period. Next, the devices were submitted to a series of memory WRITE and READ operations to evaluate the switching performance under pulsed input voltage, required for our target digital application. When single-pulse WRITE operations are concerned, as in our case, careful selection of the WRITE pulse amplitude is necessary. Higher WRITE pulse amplitudes have a high probability to trigger a successful SET or RESET. Based on observations on the device behavior under manual pulsing, we selected 500us-wide pulses of 0.9V for SET and -1.2V for RESET to guarantee a complete transition, whereas READ-out was performed at -0.15V using 3ms-wide READ pulses.

Similar switching characteristics are desired for all the memristors to be used in a practical digital application. However, circuit design and evaluation processes should take into consideration not only the cycle-to-cycle (C2C) variability in discrete devices, as evidenced in the *i*-*v* characteristic of Fig. 3,

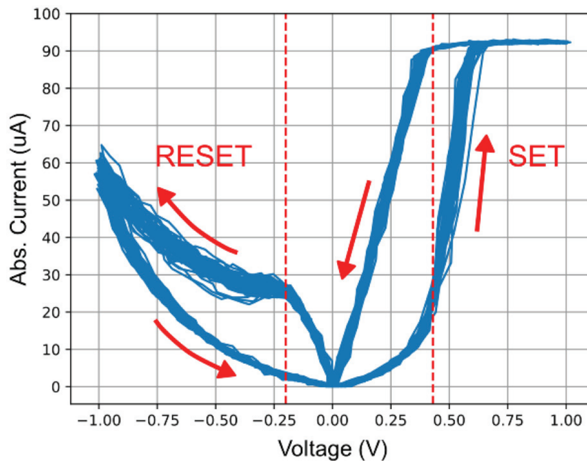

FIGURE 3
Absolute current-voltage (abs(*i*)-*v*) curve for a memristor during 50 cycles under symmetric triangular wave voltage of |1V| amplitude and 10ms period. The vertical dashed lines are a guide to the eye showing the minimum voltage required, on average, to initiate a change in the resistance for SET (R_{HRS} to R_{LRS}) and RESET (R_{LRS} to R_{HRS}).

but also the device-to-device (D2D) variability. The two memristors used in our case, namely M_A and M_B, demonstrated a qualitatively similar behavior under the same applied voltage signal, with very similar voltage thresholds for SET & RESET on average, as well as similar time-response, but with different resistance ratio k. In fact, the average values for the HRS and LRS states were measured as $\{R_{LRS}, R_{HRS}\} = \{15, 30\}$kOhm for M_A and $\{R_{LRS}, R_{HRS}\} = \{5, 45\}$kOhm for M_B. The histograms in Fig. 4 present the corresponding distributions. Such values demonstrate a low HRS-to-LRS resistance ratio of nearly $2 \leq k \leq 9$ which, according to the theoretical analysis presented previously, it could result challenging for the correct operation of the threshold logic circuit.

The schematic of the circuit in our extended experimental setup, used to implement the two-input NOR *linear threshold logic* operations, is shown in Fig. 5a. Each memristor was connected in series to an analog switch (MAX320) used to emulate the cross-point selector devices and to allow individual access to every memristor. The control signals used with the switches were the digital signals generated by the AD2 instrument, which was controlled through the *Waveforms* software interface. Both branches of memristors were connected to the inverting input of the same OP97 opamp, in a *summing amplifier* topology with a 47KOhm feedback resistor R_F. The output of the voltage adder went through another OP97 opamp used as *unity gain buffer*, whose output was driven to an AD8561 comparator to produce the digital output of the logic operations. Hysteresis was considered in the comparator´s circuit to avoid fluctuations of the digital output signal; the ratio of the resistors $R_1 / (R_1 + R_2)$ in Fig. 5a establishes the width of the hysteresis window with V_{SUP} setting the average switching voltage V_{CMP}. In Fig. 5b we show the details of the pinout interface of AD2 and an illustration of the

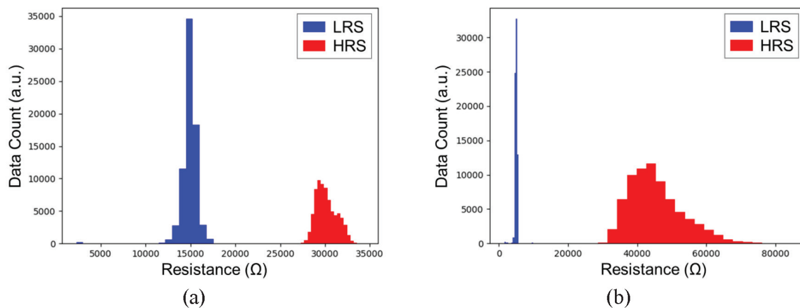

FIGURE 4
Histograms of the resistance values observed in memristors (a) M_A and (b) M_B, during 50 repetitions of the SET – READ – RESET – READ pulse sequence. The results concern the resistance values calculated from data collected during the applied READ pulses. Data count values correspond to the amount of data measured by the AD2 instrument.

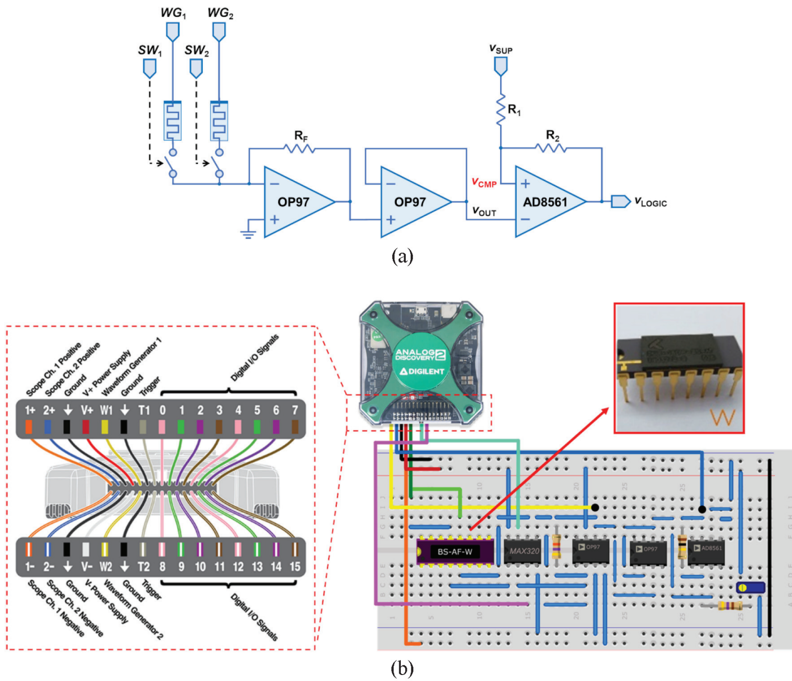

FIGURE 5
(a) Schematic of the circuit used to implement the two-input NOR *linear threshold logic* operations. (b) Illustration of the circuit implementation in a protoboard using *Knowm* memristors in a ceramic DIP package. The inset shows the interconnection interface of the AD2 instrument in detail.

actual implementation in a protoboard with the necessary connections to AD2. A 50KOhm potentiometer was used in series with R_1 to properly configure the desired V_{CMP}.

We performed a sequence of operations which included memory WRITE operations both for logic ‘0’ (RESET) and for logic ‘1’ (SET), as well as logic NOR operations, by selectively applying the required voltage pulses to the memristors via the wave generator channels WG_1 and WG_2. The characteristics of the applied pulses were as mentioned previously. More specifically, the process sequence is as follows:

i) a logic ‘1’ is written to both memristors simultaneously through two SET operations to store the “11” combination;
ii) a NOR operation takes place with the simultaneous application of V_{REF} voltage to both memristors;
iii) the memristor M_A is RESET to create the “10” combination;
iv) a NOR operation takes place as in step (ii);
v) the memristor M_B is RESET to create the “00” combination;

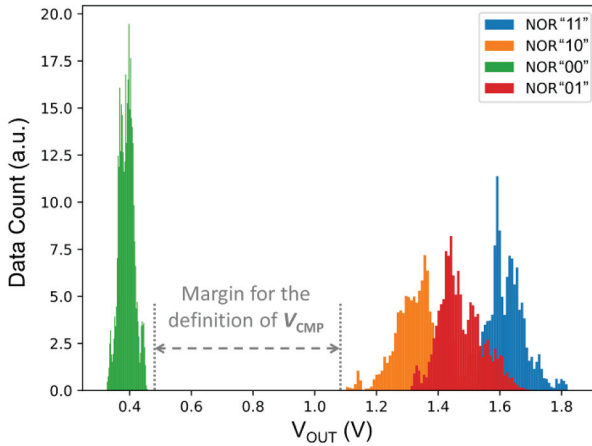

FIGURE 6
Histogram presenting the distribution of the measured voltages at the output of the summing amplifier during two-input NOR logic operations, depending on the input logic combination encoded in the resistive state of memristors. A total of 50 measurements were carried out for every input logic combination. Data count values correspond to the amount of data measured by the AD2 instrument.

vi) a NOR operation takes place as in step (ii);
vii) the memristor M_A is SET to create the "01" combination;
viii) a NOR operation takes place as in step (ii).

The abovementioned event sequence from step (i) to (viii) was repeated 50 times, aiming to observe the obtained voltage levels at the output of the summing amplifier (V_{OUT}) for every binary input combination, encoded in the resistive state of the devices. We focus only on the results corresponding to the logic NOR computations. We computed the average voltage at the output of the summing amplifier while the V_{REF} voltage was applied for logic operations. We present in Fig. 6 the histogram of the measured voltage levels, corresponding to every input logic combination.

By observing Fig. 6, we confirm the expected operation, which agrees quantitatively with theoretical calculations we performed using V_{REF} = 0.15V, R_F = 47KOhm, and the average $\{R_{LRS}, R_{HRS}\}$ values of the two memristors, mentioned previously. Based on this performance, we can define an adequate V_{CMP} voltage in the comparator stage to correctly identify logic '0' and logic '1' output. We note a wide voltage margin between 0.6V and 1.0V, which is above the largest voltage value measured for "00" input combination, and below the minimum voltage value measured for "01"/"10" input combination. So, the circuit is robust enough to respond correctly even using memristors with a much worse switching performance. In our case, with V_{CMP} = 0.7V, the comparator´s output was always correct in all the performed NOR logic operations.

Therefore, this experimental work validated the use of the voltage adder topology for threshold logic operations using memristors and demonstrated its practicality in adverse operating conditions using memristors with considerably low HRS-to-LRS ratio (below one order of magnitude) as well as C2C variability and D2D variability, evidenced through the different HRS-to-LRS resistive windows.

5 FITTING THE VOLTAGE ADDER IN COMPUTATIONAL RERAM MODULES SUPPORTING THRESHOLD LOGIC

The difficulties in the use of the voltage divider practically pose certain conditions for the suitability of any memristor device type for *ratioed threshold logic* circuits, depending on its switching features. However, as shown in the previous sections, the voltage adder constitutes an alternative to pursue for more reliable logic operations following a *linear threshold logic* scheme. In this context, we show next how the voltage adder can fit in computational ReRAM modules like that introduced in [15].

A block-level abstraction of the topology of such computational ReRAM is shown in Fig. 7. The core of the memory consists of a $m \times n$ transistor-memristor (1T1R) crossbar array with m bitlines (BLs), n wordlines (WLs), and m output lines (OLs) which are connected to the sensing circuits. The peripheral circuits enable threshold logic operations alongside typical memory READ/WRITE

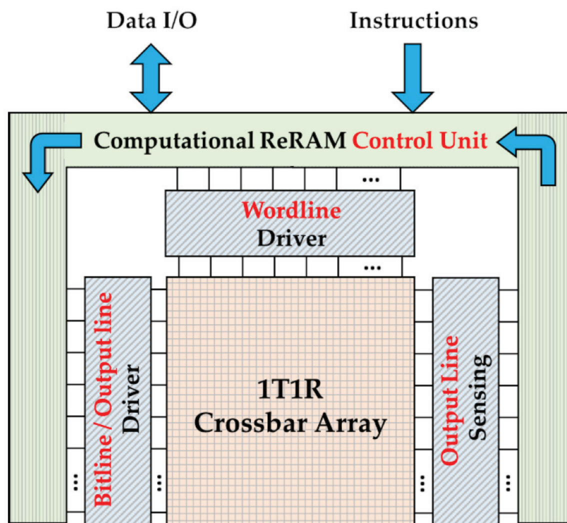

FIGURE 7
High-level block diagram showing the topology of a computational ReRAM system (adapted from [15]).

operations. The memory control unit (MCU) decodes the received instructions to activate the WL and BL/OL involved in every memory/logic operation, in a similar way as presented in [20]. Moreover, the MCU supports internally instructed memory WRITE operations with a logic value that is temporarily stored in the output stage of the sensing circuits and is driven to the BL/OL driver to be next stored in the memory via a SET or RESET process. This enables data movement, as well as *chained logic operations* with an intermediate write step required to store the output of one stage in a target memristor, so that the same data can be used as input in subsequent logic stages.

The authors in [14], [15] proposed a design for the driving/sensing circuitry of every BL/OL in such array. Considering the superior performance of *linear* against *ratioed* threshold logic operations, here we present in Fig. 8 an updated version of that design to make possible fitting the voltage adder in computational ReRAM modules supporting threshold logic. The presented circuit focuses on a single row of the memory and its driving/sensing parts, which can be replicated to account for the *m* BLs/OLs of the array (not shown here). The WLs drive the gate terminals of the select switches which are vertically-grouped to form memory words. Thus, the WLs permit the connection of the selected memristors to a common OL.

The BL/OL driver enables the selective application of WRITE or READ voltages to the terminals of the memristors depending on the value of the signals *write/read_bar* and *data*. The *data* signal defines the logic value to be stored during a WRITE operation through a SET or RESET process. Note that only positive WRITE voltages are assumed; V_{LRS} for SET and V_{HRS} for RESET. The selected memristors, through the top left side of the BL/OL driver see "$V_{BL} = V_{LRS}$" or "V_{BL} = ground" for WRITE operations, or "$V_{BL} = V_{REF}$" for READ/logic operations. Likewise, through the bottom left side of the BL/OL driver they see "$V_{OL} = V_{HRS}$" or "V_{OL} = ground" only during WRITE operations when the switch T_W is conducting. During READ/logic operations, the switch T_W disconnects the BL/OL driver and the switch T_R connects the OL to the sensing circuitry on the far-right side.

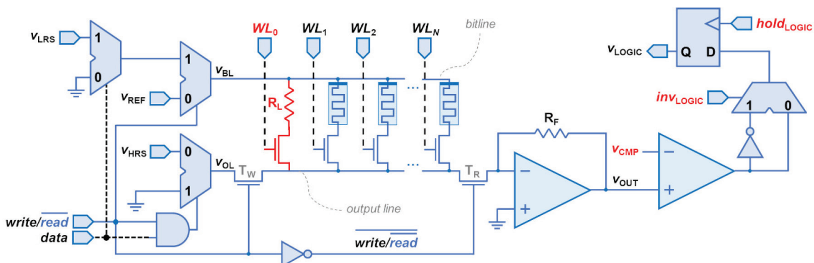

FIGURE 8
Schematic of the proposed circuit corresponding to the bitline driver and the sensing circuitry within a single row of the computational ReRAM module.

More specifically, when *write/read_bar* = ‘1’, a memory WRITE operation takes place. If *data* = ‘1’, the selected memristors see {$V_{BL} = V_{LRS}$ and V_{OL} = ground} at their terminals, whereas if *data* = ‘0’ the selected memristors see {V_{BL} = ground and $V_{OL} = V_{HRS}$}. When *write/read_bar* = ‘0’, a memory READ or a logic operation takes place and a common voltage $V_{BL} = V_{REF}$ is applied to all the selected memristors. For a memory READ operation, only one WL should be enabled. However, the circuit schematic includes in WL_0 a “dummy” memory cell corresponding to a resistive element $R_L = R_{HRS}$, which can be selectively used to increase V_{OUT} by a certain offset. The same configuration applies for a logic NOT operation since only one memristor is selected. In general, when n memristors are selected simultaneously, the topology implements the circuit shown previously in Fig. 1b for n-input linear threshold logic operations. Through the selective inversion at the output stage with the inv_{LOGIC} signal, the circuit enables the OR/NOR and the NOT/COPY operations. Note that, by modifying dynamically the V_{CMP} threshold, more operations could be possible with the same circuit, such as AND/NAND (not covered here). However, in practice the feasibility of this also depends on the distributions of the V_{OUT} voltage observed for all possible input logic combinations, as previously seen in Fig. 6 for experimental results from SDC memristors.

At the right side of the circuit schematic, we observe the output stage of the specific BL. The logic value generated by the comparison of V_{OUT} and V_{CMP}, in the original form or its complement, is asynchronously stored in the output of a local register as V_{LOGIC}, using the $hold_{LOGIC}$ control signal. The stored value can be driven to the external interface of the computational ReRAM or it can be internally driven to the BL/OL drivers to allow the storage of the logic result anywhere inside the memory array, which is equivalent to a COPY operation [25]. Nevertheless, for chained logic operations, the destination cell must be aligned row-wise (i.e., be in the same BL) with the rest of memory cell(s) to be used as input operand(s).

We validated this updated design of the computational ReRAM driver through LTSpice-based circuit simulations for a sequence of memory and logic operations. Our purpose was to demonstrate:

i) the correct transition of memristors between HRS and LRS state during memory WRITE operations;
ii) the correct identification of the stored information during READ operations;
iii) the correct execution of logic NOR operations with two input operands, for all possible input combinations.

To this end, we designed and simulated the circuit shown Fig. 8 using four memristors in the same BL and a dummy memory cell whose resistance was fixed in HRS, which was used only in READ operations. For the memristors we used the mean metastable switch memristor model proposed by *Knowm Inc*. [26] with

$\{V_{SET}, V_{RESET}\} = \{450, -450\}$mV. We did not include variability in simulations, but we purposely used a low value for k with $\{R_{LRS}, R_{HRS}\} = \{4, 10\}$KOhm which, according to Fig. 2 creates the smallest possible voltage margin for comparisons at the output stage. For the operational amplifiers, the comparator, and the analog switches, we used the corresponding circuit models as in the experimental implementation with ±5V supply, whereas digital components were treated as ideal elements. Moreover, we used ±1V WRITE pulses, a resistor R_F = 50KOhm, V_{REF} = 100mV, and a voltage threshold V_{CMP} = -1.32V which was adequate for all the READ/logic operations. The simulation results are presented in Fig. 9 which shows the evolution of the resistance of all the memristors, the logic state of the WL signals, the logic state of the control signals *write/read_bar*, inv_{LOGIC}, and $hold_{LOGIC}$, as well as the logic signal of the output of the local register V_{LOGIC}. Memristors M_1 and M_2 were initially found in HRS, whereas M_3 and M_4 were purposely set in LRS. Seven different operations take place, which include memory WRITE (both for SET and for RESET), memory READ (both for logic '1' and for logic '0'), and logic NOR, as summarized in Table I.

In every operation, the *write/read_bar* signal takes the corresponding value, the required WLs are activated and the V_{LOGIC} output is updated on the

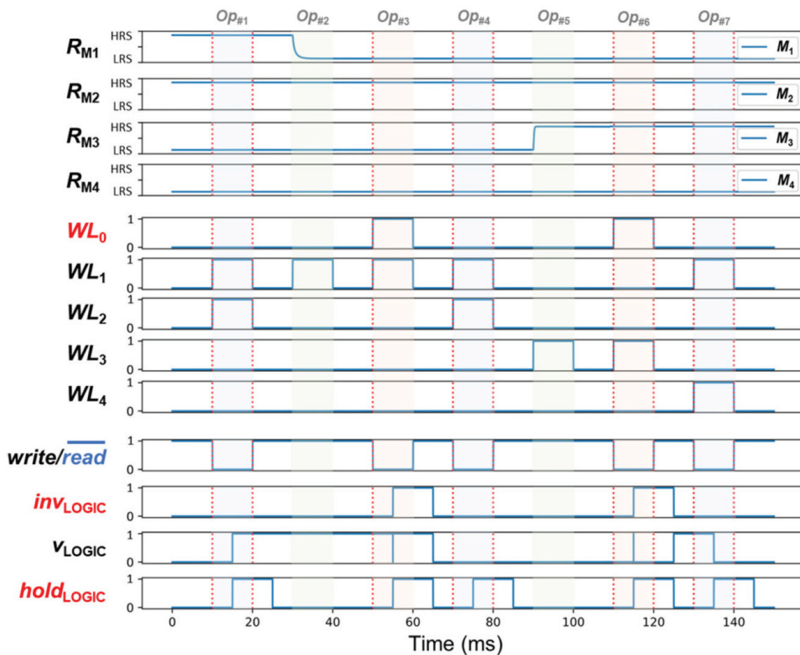

FIGURE 9
LTSpice circuit simulation results for the circuit topology shown in Fig. 8. The vertical dashed lines are a guide to the eye, for the time-frame of different operations which are also identified using different color in the plot background.

Operation #	Type	Description
1	logic NOR(M_1, M_2)	NOR(0, 0) since memristors M_1 and M_2 are in HRS
2	WRITE ‘1’ to M_1	Memristor M_1 undergoes a SET from HRS to LRS
3	READ state of M_1	The stored logic ‘1’(LRS) is interpreted in the output
4	logic NOR(M_1, M_2)	NOR(1, 0) after memristor M_1 was SET in LRS
5	WRITE ‘0’ to M_3	Memristor M_3 undergoes a RESET from LRS to HRS
6	READ state of M_3	The stored logic ‘0’(HRS) is interpreted in the output
7	logic NOR(M_1, M_4)	NOR(1, 1) since memristors M_1 and M_4 are in LRS

TABLE 1
Sequence of Memory and Logic Operations Realized in Simulation

rising edge of the $hold_{LOGIC}$ signal. We observe in Op#1 that for “00” resistive input combination, the V_{LOGIC} signal is set to logic ‘1’, whereas in Op#4 and Op#7 it stays at logic ‘0’, as expected for two-input NOR logic operations. During WRITE operations, we observe the correct transition in the state of the selected memristor, without any change induced to the state of the unselected devices. Finally, during READ operations, along with the WL of the target memristor, the WL_0 activates the dummy cell and the inv_{LOGIC} signal is also high to complement the output. Therefore, for a stored LRS in Op#3 the V_{LOGIC} output takes logic ‘1’ value, whereas for a stored HRS in Op#6 the V_{LOGIC} output takes logic ‘0’ value, which agrees with the convention assumed for the representation of data in form of resistance.

Everything considered, these functional circuit simulation results validate the correct operation of the presented design of the BL/OL drivers and the READ circuitry in the periphery of a ReRAM module supporting threshold logic operations in memory owing to the use of the voltage adder. Through theoretical analyses, experimental validation and simulation results of an extended ReRAM system, this comprehensive study contributes towards the design and development of computational ReRAM modules. Future work will focus on the execution of chained logic operations in memory, the use of unconventional grouping strategies for the select transistors [27], the overhead in peripheral circuit area and the trade-off between latency and parallelism of logic operations in memory, towards the development of dedicated logical and physical synthesis flows for computational ReRAM modules supporting threshold logic.

CONCLUSIONS

This work focused on the design of threshold logic in memory with memristors and assessed two circuit design concepts for their implementation, compatible with the crossbar topology. An analytical exploration of the impact of the HRS-to-LRS ratio of memristors on the performance of every circuit

highlighted the advantages of the voltage adder which distinguished as a practical solution for the design of robust computational resistive memories. Trough experimental results from commercial memristors, we validated the feasibility and the conclusions derived from the theoretical analysis for computational ReRAM modules supporting threshold logic gates. To this end, we presented the design of a bitline driver and sensing circuitry for a 1T1R ReRAM array where the linear threshold logic scheme can seamlessly fit. Therefore, through detailed theoretical analyses, experimental validation and simulation results of an extended ReRAM system, this comprehensive study presents potential solutions towards the development of robust and highly-reliable computational ReRAM modules for future unconventional computing systems.

ACKNOWLEDGMENTS

This work was supported in part by the Chilean government under grants FONDECYT Regular No. 1221747 and ANID-Basal FB0008. The authors also acknowledge financial support from MINECO (Spain) grant TED2021-129643B-I00 (FLEXRAM Project).

REFERENCES

[1] D. Ielmini, H.-S. Philip Wong, "In-memory computing with resistive switching devices," *Nat. Electron*., vol. 1, pp. 333–343, 13 June 2018

[2] M. Zidan, A. Chen, G. Indiveri, and W. Lu, "Memristive computing devices and applications," *J. Electroceram.*, vol. 39, pp. 4–20, 2017

[3] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S.Williams, "The missing memristor found," *Nature*, vol. 453, pp. 80–83, 2008

[4] H. Aziza, M. Moreau, M. Fieback, M. Taouil, S. Hamdioui, "An Energy-Efficient Current-Controlled Write and Read Scheme for Resistive RAMs (RRAMs)," *IEEE Access*, vol. 8, pp. 137263 - 137274, 2020

[5] I. Vourkas, and G. Ch. Sirakoulis, "Emerging Memristor-based Logic Circuit Design Approaches: A Review," *IEEE Circuits and Systems Magazine*, vol. 16, no. 3 (3rd quarter), pp. 15-30, 2016

[6] L. Xie *et al*., "Scouting logic: A novel memristor-based logic design for resistive computing," 2017 *Proc. IEEE Comput. Soc. Annu. Symp. VLSI* (ISVLSI), Bochum, Germany, July 3-5

[7] R. Ben Hur, N. Wald, N. Talati, and S. Kvatinsky, "Simple magic: Synthesis and in-memory Mapping of logic execution for memristor-aided logic," 2017 *IEEE/ACM International Conference on Computer-Aided Design* (*ICCAD*), Irvine, CA, USA, Nov. 13-16

[8] M. Maestro-Izquierdo *et. al*., "Experimental time evolution study of the HfO2-based IMPLY gate operation," *IEEE Trans. Electron Devices*, vol. 65, no. 2, pp. 404–410, Feb. 2018

[9] L. Gao, F. Alibart, and D. B. Strukov, "Programmable CMOS/Memristor Threshold Logic," *IEEE Trans. on Nanotechnol.*, vol. 12, no. 2, pp. 115 - 119, 2013

[10] A. P. James, L. R. V. J. Francis, and D. S. Kumar, "Resistive Threshold Logic," *IEEE Tran. on Very Large Scale Integration (VLSI) Systems*, vol. 22, n0. 1, pp. 190 - 195, 2014

[11] T. Tran, A. Rothenbuhler, E. H. Barney Smith, V. Saxena, and K. A. Campbell, "Reconfigurable Threshold Logic Gates using Memristive Devices," 2012 *IEEE Subthreshold Microelectronics Conference* (*SubVT*), Waltham, MA, Oct. 09-10

[12] K. A. Campbell, "Self-directed channel memristor for high temperature operation," *Microelectron. J.*, vol. 59, pp. 10-14, Jan. 2017

[13] Knowm Inc., [Online]. Available: https://knowm.org

[14] C. Fernandez, I. Vourkas, and A. Rubio, "Design and Simulation of Peripheral Driving Circuitry for Computational ReRAM," 2022 *IEEE Conference on Design of Circuits and Integrated Systems* (*DCIS*), Pamplona, Spain, Nov. 16-18

[15] C. Fernandez, and I. Vourkas, "Reliability-Aware Ratioed Logic Operations for Energy-Efficient Computational ReRAM," 2022 *IFIP/IEEE Int. Conference on Very Large Scale Integration* (*VLSI-SoC*), Patras, Greece, Oct. 03-05

[16] M. Escudero, I. Vourkas, A. Rubio, and F. Moll, "Memristive Logic in Crossbar Memory Arrays: Variability-Aware Design for Higher Reliability", *IEEE Trans. Nanotechnol.*, vol. 18, pp. 635 – 646, 2019

[17] C. Fernandez, and I. Vourkas, "ReRAM-based Ratioed Combinational Circuit Design: a Solution for in-Memory Computing," 2020 *Int. Conf. on Modern Circuits and Syst. Technol.* (*MOCAST*), Bremen, Germany, Sept. 07-09

[18] C. Fernandez, I. Vourkas, and A. Rubio, "Circuit Topology and Synthesis Flow Co-Design for the Development of Computational ReRAM," 2022 *IEEE Int. Conf. on Nanotechnology* (*NANO*), Palma de Mallorca, Spain, July 04-08

[19] G. Papandroulidakis, I. Vourkas, A. Abusleme, G. Ch. Sirakoulis, and A. Rubio, "Crossbar-Based Memristive Logic-In-Memory Architecture", *IEEE Trans. Nanotechnol.*, vol. 16, no. 3, pp. 491 - 501, 2017

[20] F. Pinto, and I. Vourkas, "Robust Circuit and System Design for General-Purpose Computational Resistive Memories," *MDPI Electronics*, vol. 10, no. 9, pp. 1074, 2021

[21] Analog Discovery 2, "A high performance, all-in-one portable oscilloscope & instrumentation system," [Online]. Available: https://analogdiscovery.com

[22] I. Valov, "Interfacial interactions and their impact on redox-based resistive switching memories (ReRAMs)," *Semicond. Sci. Technol*. vol. 32, no. 093006, 2017

[23] J. Gomez, I. Vourkas, and A. Abusleme, "Exploring Memristor Multi-Level Tuning Dependencies on the Applied Pulse Properties via a Low Cost Instrumentation Setup", *IEEE Access*, vol. 7, pp. no. 1, pp. 59413 – 59421, 2019

[24] A. Cirera, C. Fernandez, I. Vourkas, and A. Rubio, "Exploring Different Circuit-level Approaches to the Forming of Resistive Random Access Memories," 2022 *Int. Conf. on Modern Circuits and Syst. Technol.* (MOCAST), Bremen, Germany

[25] C. Fernandez, and I. Vourkas, "On the Design and Development of a ReRAM-based Computational Memory Prototype," 2022 *IFIP/IEEE Int. Conference on Very Large Scale Integration* (*VLSI-SoC*), Patras, Greece, Oct. 03-05

[26] The Mean Metastable Switch Memristor Model in LTSpice, [Online]. Available: https://knowm.org/memristor-models-in-ltspice/

[27] I. Vourkas, G. Papandroulidakis, G. Ch. Sirakoulis, and A. Abusleme, "2T1M-Based Double Memristive Crossbar Architecture for In-Memory Computing," *Int. Journ. of Unconventional Computing*, vol. 12, no. 4, pp. 265-280, 2016