DOI: 10.1002/((please add manuscript number))
Article type: Communication

Logic computing with stateful neural networks of resistive switches

*Zhong Sun, Elia Ambrosi, Alessandro Bricalli and Daniele Ielmini**

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano and IU.NET, Piazza L. da Vinci 32 – 20133 Milano, Italy
E-mail: daniele.ielmini@polimi.it

Keywords: stateful logic, neural network, resistive switching memory, in-memory computing, neuromorphic

Brain-inspired neural networks can process information with high efficiency, thus providing the solution of choice for pattern recognition and other artificial intelligent tasks. By adopting binary inputs/outputs, neural networks can be used to perform Boolean logic operations, thus potentially surpassing the complementary-metal-oxide-semiconductor (CMOS) logic in terms of area efficiency, execution time and computing parallelism. Here we introduce the concept of a neural-network-based logic circuit consisting of resistive switches, which can perform all logic functions with the same network topology. The neural network relies on physical computing according to Ohm's law, Kirchhoff's law, and the ionic migration within an output switch serving as the highly nonlinear activation function in the McCulloch-Pitts neuron model. The input and output are both nonvolatile resistance states of devices, thus enabling stateful and cascadable logic operations. Applied voltages provide the synaptic weights, which enables the convenient reconfiguration of the same circuit to serve various logic functions. The neural network can solve all 2-input logic operations with just one step, except for the exclusive OR (XOR) needing 2 sequential steps. 1-bit full adder operation is shown to take place with just 2 steps and 5 resistive switches, thus highlighting the high efficiencies of space, time, and energy of logic computing with the stateful neural network.

After more than 50 years of evolution, Moore's law is approaching its end due to the inevitable technological and physical scaling limits of the CMOS transistors.[1] A key concern of the conventional von Neumann computer architecture is the round data transfer between physically-separated memory and computing units, which causes high energy-inefficiency and latency burdens.[2,3] In the era of big data, these major obstacles must be overcome to develop computing systems capable of processing a huge amount of data with high efficiency. A key concept to address these issues is in-memory computing, where logic operations are performed *in situ* within a memory unit, usually consisting of a crossbar array of resistive switching devices.[4-12] A resistive switching device, also known as memristor, is a 2-terminal electronic device whose resistance can be electrically switched, and the behavior is nonvolatile.[4] In a typical resistive switch, known as the bipolar resistive random access memory (RRAM), application of a pulse with positive voltage above the set voltage (V_{set}) induces the transition to the low resistance state (LRS), which is due to the field-induced migration of defects that forms a conductive filament across a dielectric layer. Application of a pulse with negative voltage below the reset voltage (V_{reset}) causes the retraction of the conductive filament, thus inducing a transition to the high resistance state (HRS, see Figure S1 in Supporting Information).[13] Resistive switching devices have been used to implement Boolean logic operations,[5-9] where the LRS and HRS can be used as binary input/output states. In a typical RRAM logic gate, one or more RRAM act as input devices, while their resistance states conditionally imply ionic migration in an output device, causing a change of the resistance according to the truth table of a logic function, such as AND[8] or material implication (IMP).[5] However, more advanced functions, such as XOR, addition, and multiplication, require a relatively large number of RRAM and multiple operation steps, *e.g.*, 7 steps with 11 devices for a 1-bit full adder (FA).[8] Also, no universal theoretical framework has been developed so far to synthesize logic gates, in terms of circuit architecture and applied voltages to execute the computation. A sound theoretical basis is however essential to

developing electronic design automation (EDA) tools for future in-memory logic computing systems.

In this scenario, the neural network concept can provide a universal architecture to implement various logic functions efficiently.[14] In a brain-inspired neural network, the neuron is the basic computing unit, providing addition of many input stimuli after proper weighting by synaptic junctions (Figure S2, Supporting Information). The possibility to perform logic computation within an artificial neuron has been recognized since the seminal model by McCulloch and Pitts,[15] later extended to the concept of threshold logic.[16,17] In a typical threshold logic circuit, input currents are summed to yield an internal state variable, which in turn triggers the output firing according to a highly nonlinear response, *e.g.*, a step-like function. Here we introduce a neural network with resistive switches playing the roles of input states and an output artificial neuron, able to perform linearly-separable logic functions (all 2-input Boolean functions except XOR and its complement), in just one step. Thanks to the nonvolatile nature of the logic variables, the circuit can be extended to the multi-layer neural network to realize linearly non-separable functions in multiple steps, such as XOR operation and 1-bit FA, in just 2 steps, thus paving the way to compact, fast, and energy-efficient stateful logic computing circuits based on resistive switches.

Figure 1a illustrates the RRAM circuit, consisting of 3 devices A, B, and C, connected to an internal node, and to 3 independent voltage supplies of voltage V_A, V_B and V_C, respectively. A load resistor with conductance G_L connects the internal node to ground. The current flowing through any RRAM device (or the load resistor) can be generally written as $(V_i-V_{int})G_i$, where i is an index spanning A, B, C or L, with V_L being generally connected to ground (V_L = 0 V), and G_i being the conductance of the i-th RRAM (or the load resistor). By equating all currents at the internal node to zero according to Kirchhoff's law, we get $\Sigma_i(V_i-V_{int})G_i = 0$, which allows to write the internal potential V_{int} as:

$$V_{int} = \frac{\sum_i V_i G_i}{\sum_i G_i}. \quad (1)$$

To operate the circuit as a logic gate, A and B serve as input data, with LRS and HRS corresponding to logic 1 and 0, respectively. In RRAM devices with high HRS/LRS resistance ratio, the LRS conductance and HRS conductance can be normalized approximately as the real numbers 1 and 0, respectively, which allows the logical value and device conductance to be used interchangeably. The element C is assigned the role of output RRAM, whose initial state is always off, *i.e.*, $G_C = 0$. For C to switch to the on state, the voltage V_C-V_{int} must exceed the threshold voltage V_{set}, namely $V_C - V_{int} \geq V_{set}$. After substituting Eq. (1) in the previous expression, we get:

$$\frac{\sum_i G_i (V_C - V_i - V_{set})}{\sum_i G_i} = \frac{Y}{\sum_i G_i} \geq 0, \quad (2)$$

where the internal state variable $Y = \Sigma_i G_i (V_C - V_i - V_{set})$ is an equivalent current whose sign controls the condition for C to switch to the on state ($Y \geq 0$) or not ($Y < 0$). The state variable Y can be viewed as the weighted sum in a neural network, namely $Y = \Sigma_i G_i w_i$ where G_i values act as input variables, while the synaptic weights are given by:

$$w_i = V_C - V_i - V_{set}. \quad (3)$$

Therefore, the circuit can be viewed as a perceptron neural network[14] (Figure 1b), where input neurons A and B contribute their signals to the internal node potential V_{int}, while C serves as the output neuron. With input from A and B, the internal state variable Y controls the conditional transition of C to a new state C', *e.g.*, C' = 1 for $Y \geq 0$. The abrupt set transition in C plays the role of step-like activation function in the McCulloch-Pitts neuron model, thus realizing a multiple-addend integrate&fire operation in a nanoscale element via physical computing.[18,19] Note that the synaptic weights are determined by the applied voltages, therefore the neural network can be reconfigured directly by the analog voltages applied to the circuit. Although C appears among the input data in Eq. (2), it is always initialized in the HRS

($G_C = 0$), thus plays no role in the input signals. Similarly, the load resistance conductance G_L can be considered as a constant analog bias in the network.

By tuning the synaptic weights, *i.e.*, applying different voltages V_i to the perceptron circuit in Figure 1a, all the linearly-separable logic functions such as NAND and NOR can be implemented in one step.[14] In fact, Boolean logic can be regarded as pattern classification problem, which is straightforwardly addressed by the perceptron. Figure 1c considers the case of a NAND function, showing the input/output characteristics, namely the final state of C as a function of A and B on axis x and y, respectively, and the corresponding truth table. In Figure 1c, the final C is always 1 (C' = 1, labeled by full symbols) except for A = B = 1, where C' = 0 (open symbol), *i.e.*, unchanged with respect to the initial state. The decision boundary that separates the outputs of 1 and 0 gives the relationships between the weights and bias, from which the required voltages can be deduced for a specified load resistance, as shown at the bottom of the truth table. The set condition in Figure 1c can be described by the linear inequality:

$$A+B \leq 3/2, \quad (4)$$

where the input states A and B are mapped by the RRAM conductance values, *i.e.*, G_A and G_B, respectively, and the real number 3/2 is mapped to $3G_{LRS}/2$, where G_{LRS} is the nominal LRS conductance, corresponding to the logical value 1. The decision boundary was chosen to maximize the tolerance with respect to the conductance variation of devices (Figure S3, Supporting Information), while it could be differently optimized in practical cases. By comparing Eq. (4) with the condition $Y \geq 0$ in Eq. (2), we identify two conditions for parameters V_A, V_B, V_C and G_L, which help to dictate the applied voltages. For instance, assuming $G_L = 1.4G_{LRS}$ and $V_A = 0.7V_{set}$, with the identified conditions, we obtained $V_B = 0.7V_{set}$ and $V_C = 1.35V_{set}$, which configure a NAND logic gate in the circuit of Figure 1a. Details are explained in Figure S4 (Supporting Information), where the choice of the value of G_L is also discussed. Similarly, from the boundary condition A+B ≤ 1/2 in the NOR

characteristics of Figure 1d, one can derive $V_A = V_B = 0.5V_{set}$ and $V_C = 1.1V_{set}$ for the NOR logic gate (Figure S4, Supporting Information). The value of V_A should be chosen within a certain range, to support the logic operations while preventing changes of the input states during the logic operation (Table S1, Supporting Information). To overcome the V_{set} variation of output device, V_A should be chosen around the middle of the corresponding range. The same concept is applied to all other 12 linearly-separable 2-input Boolean logic functions (Table S1, Supporting Information). Therefore, the neural-network nature of the circuit in Figure 1a enables a versatile logic tile for universal design of generic linearly-separable 2-input logic gates.

To directly demonstrate the aforementioned concept, RRAM devices with HfO_2 dielectric layer were used as resistive switches in the circuit. Each RRAM device had a transistor connected in a one-transistor-one-resistor (1T1R) structure, to enable current limitation during the set transition thus preventing destructive breakdown, and to controllably tune the LRS conductance.[20] **Figure 2**a shows the experimental current-voltage (*I*-*V*) characteristic of the RRAM device under quasi-stationary conditions, indicating set transition for positive voltage above V_{set} (1.7 V), and reset transition for negative voltage. Figure 2b shows the set/reset transitions under pulsed conditions used in the logic operations. A load resistor with conductance $G_L = 1.4G_{LRS} = 56$ μS was adopted for both NAND and NOR logic gates. Before the logic operations, each RRAM was initialized to its desired state, then read with a low voltage pulse to avoid any disturbance. Voltages V_A, V_B and V_C were applied simultaneously, although with different pulse-widths, namely 100 μs, 200 μs and 300 μs for V_C, V_B and V_A, respectively, to better visualize the switching process. After the application of the pulses, conductance of each RRAM was read again to check the computing results. Figure 2c shows the applied voltages for read and NAND logic operation (top panel), the measured currents across each RRAM during the read phases, and the measured currents across the load resistor during the logic computing phase (bottom panel). Device C shows set transition according to

the NAND function of inputs A and B, *i.e.*, for either A or B (or both) being initially in their LRS. In all cases, the input states are left unchanged after the logic operations. The same is shown in Figure 2d for the NOR logic operation, where C undergoes set transition only for A = B = 0, thus satisfying the input/output characteristics of Figure 1d. From a circuit viewpoint, the logic gates rely on the comparison between voltage across the output RRAM V_C-V_{int} and the threshold V_{set}, as shown in Figure 2e and f for NAND and NOR, respectively. The NAND and NOR logic operations were also simulated by circuit simulations adopting an analytical model of RRAM,[21] and the results are reported in Figures S5 and S6 (Supporting Information), respectively. The agreement between the simulated and experimental characteristics confirms the solid understanding and prediction of the logic operations.

Among the 2-input Boolean logic functions, the input/output characteristics of XOR operation shown in **Figure 3**a is not linearly separable,[14,22] thus cannot be represented by a single-layer perceptron as in Figure 1c and d. As linearly non-separable functions can be realized by a 2-layer perceptron, the XOR gate is synthesized by sequentially-cascading 2 stateful logic operations in the same network as shown in Figure 3a. First, the material non-implication (NIMP) is executed to induce set transition for input A = 1 and B = 0, followed by the converse non-implication (C-NIMP) to induce set transition for A = 0 and B = 1. The opposite sequence might also be performed, as intermediate states are always stored *in situ*, and input states are never affected by the logic operations. Note that this approach benefits from the nonvolatile state of C, to cumulate the set transition in the first operation with output C', and the second operation with output C'', to yield a non-separable XOR output. Figure 3b depicts the 2-layer perceptron for the XOR function, where each layer has different synaptic weights corresponding to NIMP and C-NIMP operations, respectively. Figure 3c shows the corresponding truth table and the voltage values V_A, V_B and V_C, to yield the correct weights for the two sequential operations. G_L is equal to $0.5G_{LRS}$ for both operations. As a reference, previous stateful XOR logic gates required at least 4 operations with more devices,[6,8,9] which

highlights the time/space efficiency of the present neural-network-based computing approach. Figure 3d shows the experimental results of the XOR operation, where the resistance states were read with low voltage pulses before and after each logic operation. Figure 3d also shows the voltage V_C-V_{int} compared to V_{set}, showing that the set transition of C is possible only according to the truth tables of NIMP and C-NIMP. The XOR logic operation was simulated by using an analytical model of RRAM,[21] showing good accuracy of predicting the circuit operation for all configurations of input states (Figure S7, Supporting Information).

As a neural network can have a large number of inputs, it also allows to implement 3-input logic operations, such as majority and FA functions. Generally, implementing these functions with conventional CMOS technology leads to a high transistor count and a correspondingly large area, *e.g.*, 28 transistors for a CMOS 1-bit FA[23] and a corresponding area of few thousands of F^2, where F is the minimum feature size in the microelectronic manufacturing process.[23,24] On the other hand, implementation of 3-input logic functions is a straightforward extension of the 2-input case with a neural network. For instance, **Figure 4**a shows the truth table of the 1-bit FA operation, including input data A, B and carry-in (C_{in}), and the outputs carry-out (C_{out}) and sum (S), with the latter corresponding respectively to the most significant bit and the least significant bit of the 1-bit summation of A, B, and C_{in}. Note that C_{out} and S can be viewed as the majority and parity functions of the input data, respectively,[25] which are shown in the input/output characteristics of Figure 4b. Since the majority function is linearly separable, C_{out} can be immediately calculated by single-layer perceptron in a single-step operation, as in Figure 1. On the other hand, Figure 4b shows that the 3-input parity function is not linearly separable in the (A, B, C_{in}) space, thus requiring a multi-step operation as in the XOR case in Figure 3. However, it can be shown that S becomes linearly separable in the (A, B, C_{in}, C_{out}) space,[26] *i.e.*, C_{out} is also considered among the input variables (Figure S8, Supporting Information). In this approach, the 1-bit FA can be

implemented with only 5 resistive switches, one for each input/output variable, and 2 steps, to sequentially calculate C_{out} and S.

Figure 4c shows the 1-bit FA circuit and the corresponding 2-layer perceptron, while Figure 4d shows the applied voltages and measured currents in the logic gate during read and compute phases. First, the majority operation C_{out} = majority(A, B, C_{in}) is completed by tuning the synaptic weights via the applied voltages, namely $V_A = V_B = V_{Cin} = -V_{set}$, and $V_{Cout} = 0.4V_{set}$, while V_S is left unbiased. The weights are designed such that the voltage difference V_C-V_{int} increases linearly with the number of input RRAM in LRS, thus inducing a set transition across the output device when there are at least 2 input RRAM in LRS. In the second step, S is calculated by applying the same voltages as the preceding operation, with the additional $V_S = 0.52V_{set}$. The load resistor is $G_L = 0.83G_{LRS}$ during both steps. The experimental results in Figure 4d support the feasibility of the 1-bit FA operation in just 2 computing steps. The FA circuit behavior can be accurately predicted by circuit simulations with the analytical model of RRAM (Figure S9, Supporting Information), thus supporting the controllability of the RRAM neural network for advanced logic operations.

The proposed scheme of stateful neural network allows flexible logic computing, namely, all logic gates share the same basic structure, which can be viewed as a crossbar array with several resistive switches connected to separate column lines and the same row line. This allows carrying out logic computing tasks directly on a crossbar memory circuit,[27,28] without the need of data transfer between the memory and the computing units. Different logic functions can be implemented in the same circuit by adjusting the weights, which conveniently takes place by analog tuning of the applied voltages. The same load resistance can be employed for different logic operations, as demonstrated for NAND and NOR, and placing a transistor as a tunable load resistor would allow full flexibility for the logic gate design. More operations can be cascaded thanks to the coherence of input and output states being coded as the nonvolatile RRAM conductance.[5-10] As a comparison, some other in-

memory computing strategies involve different logic variables for input and output,[29,30] such as input voltage and output resistance, which is not compatible with logic cascading, as it requires an additional transformation, *e.g.*, from resistance to voltage, to be completed out of the memory circuit.[31] The voltage-based reconfigurability of the neural-network circuit and the nonvolatile devices allow minimizing the number of steps for completing logic operations with respect to other stateful logic approaches,[5-9,32-34] as summarized in **Table 1** for XOR and 1-bit FA functions.

Physical computing in the neural-network-based logic circuit takes place by 3 physical principles, namely (1) the Kirchhoff's law of electrical current conservation, (2) the Ohm's law relating potential and electrical current, and (3) the voltage-controlled drift of ionized defects within a dielectric material,[35-37] *e.g.*, the HfO_2 layer in the RRAM devices used in this work. Combined together, these 3 phenomena control the switching of the output switch subject to the condition of the V_C-V_{int} overcoming the threshold V_{set}. The parameter V_{set} is indeed critical to determining the error probability in the logic operations. In particular, any stochastic variation of V_{set} from device to device, or even from cycle to cycle in the same device,[38] may cause unpredictable deviations and consequent failure of computing. The maximum variation of V_{set} which still ensure safe operations for all logic function can be estimated with respect to a maximum tolerated error rate, *e.g.*, 10^{-6} in Figure S10 (Supporting Information). The maximum error depends on the required accuracy of the calculation, and the logic computing architecture.[39] The results indicate that different logic operations are differently tolerant to V_{set} variations, thus providing a guideline about how to select some functions instead of others to minimize computing errors. Improvement of V_{set} variability might also be achieved, *e.g.*, via dispersion of metal nanodots within the dielectric switching layer,[40] or via engineering dislocations in a single-crystalline device.[41] On the other hand, the resistance variations represent a negligible concern, since the reported LRS variations, *e.g.*,

generally a few percent when programmed with a relatively high compliance current,[38] are well within the allowed range (Figure S11, Supporting Information).

Although the neural-network-based logic circuit is extremely attractive in terms of numbers of computing steps and switches, it should be recalled that operating a logic circuit in the memory requires specific periphery circuits.[42] These circuits include registers to store the addresses of data and the codes to be executed, the pulsed voltage generators and the multiplexers to distribute the voltage pulses to the rows/columns in the array. In addition, the circuit should be capable of programming/reading the memory elements as in a conventional memory array. In particular, with respect to previous approaches,[9] our concept makes extensive use of analog voltages needed to tune the synaptic weights in the stateful neural network. Generating and distributing these analog voltages impose an additional complexity to the periphery circuit. On the other hand, analog voltages are becoming increasingly used by in-memory computing architectures in recent works. Matrix-vector multiplication (MVM) in crosspoint arrays relies on the generation and conversion of analog voltages, thus enabling image processing,[43] sparse coding,[28] and deep learning accelerators.[44] In this scenario, the development of mixed analog/digital circuits for computing in memory arrays will further boost analog solutions like the one proposed in this work.

Another key concern for all in-memory computing systems is the energy efficiency of computation. Since in-memory logic computing relies on the conditional resistance switching in the output RRAM device, the stateful logic requires a larger energy compared to conventional CMOS logic gates.[42] The typical energy consumption for RRAM switching is about 0.1 pJ per operation,[45] which far exceeds the energy for dynamic switching in CMOS devices.[42] However, the evaluation of energy efficiency should not limit itself to the individual logic operation, rather it should cover the entire computing process, which in the case of CMOS logic involves also significant energy for data transfer between the memory and the processing units, and the energy for sustaining and refreshing the volatile bits in

CMOS memory and logic units. All these energy consumptions are suppressed in the stateful logic paradigm, thanks to the non-volatile nature of RRAM, and to the *in situ* computing with non-von Neumann architecture. The neural-network circuit proposed in this work reduces intensively the numbers of computing steps and devices, thus contributing to higher efficiency of stateful logic.

In summary, we introduce a neural-network-based logic computing concept for a resistive switch circuit, which acts as a universal architecture to compute Boolean functions, for both linearly separable and non-separable operations. The universality and flexibility of neural network endow the circuit with significant advantages for logic design, *e.g.*, one step for NAND, NOR and majority operations, and 2 steps for XOR operation and 1-bit FA, using the same circuit topology. Since the logic input and output variables are encoded in nonvolatile conductance states of resistive switches, *i.e.*, HRS and LRS, the circuit enables stateful logic computing, thus eliminating the von Neumann bottleneck the plagues classic computers. The homogeneity of logic variables allows the operations to be directly cascaded, while the voltage-controlled synaptic weights enable function reconfiguration within the same circuit. The stateful neural network is thus promising as a flexible logic scheme for high-performance, high-density in-memory computing.

Experimental Section

Experimental Devices: The RRAM devices in this work consist of a stack of a thin layer (5 nm) of HfO_2 deposited by e-beam evaporation on a confined bottom electrode of graphitic carbon. A thin film of Ti was deposited as top electrode on the HfO_2 dielectric layer by e-beam evaporation without breaking the vacuum. The deposited Ti has been reported to act as oxygen scavenger,[46] leading to an oxygen exchange layer of TiO_x between Ti and HfO_2. The oxygen exchange layer is instrumental in creating a local enhancement of oxygen vacancy concentration in HfO_2, thus improving the leakage current in the pristine state, reducing the

forming voltage, and forcing a unidirectional switching behavior, where set and reset transitions take place under positive and negative voltages applied to the top electrode, respectively. Forming was operated in DC regime by applying a voltage sweep to the top electrode from 0 to 5 V, to induce a soft breakdown process of HfO_2 which initiated the CF creation and the related resistive switching process. The RRAM bottom electrode was connected to an integrated transistor by a W plug. Integration of transistor and the RRAM devices on the same chip were obtained by conventional CMOS process. The DC conduction and switching characteristics of the RRAM were collected by an HP4155B Semiconductor Parameter Analyzer connected to the experimental device in a conventional probe station for electrical characterization.

Experimental Measurements: The experiments were carried out in a probe station using a 4-channels arbitrary waveform generator (Aim-TTi TGA12104) and an oscilloscope (LeCroy Waverunner 64Xi). During the logic operations, the source node of the transistors in the 1T1R structures were connected to an external load resistor, whose other node was connected to the oscilloscope to probe the total current. The gates of the integrated transistors were short-circuited and connected to the waveform generator through pad-to-pad wire bonding. Before the logic operations, the load resistance was short-circuited and the input/output devices were prepared in their initial states and read individually at a low voltage (0.5 V). The devices were read again after the logic operation to detect any change of state in either input or output devices. For 2-input/1-output logic operations, such as NAND and NOR, the voltages were supplied by the arbitrary waveform generator to the top electrodes and gates. The voltage pulse widths were 100 μs, 200 μs and 300 μs for the output, input B and input A, respectively. For the majority operation (3-input/1-output), two input devices having the same voltage value shared the same channel of the waveform generator. A Keithley 707 Switching Matrix was adopted to independently access each device during the read phase. For the second

operation in the FA to obtain S (4-input/1-output), three input devices having the same voltage value shared the same channel of the waveform generator.

Analytical Simulations: The simulations of logic operations were based on the analytical model for HfO_2-based RRAM proposed in Ref. 21. In this model, the resistances in LRS and HRS are controlled by the filament diameter and gap length, respectively, and the set and reset transitions are described by filament growth and gap opening, respectively, which are activated by the local field and temperature. In the logic circuit, all the RRAM devices were simulated by the same model. For the logic operations, the initial RRAM state '0', *i.e.*, the HRS, is described by a ruptured filament with a gap of length 2.4 nm, while the initial state '1', *i.e.*, the LRS, is described by a complete filament with a diameter 2.1 nm. When the external voltage pulses are applied, the voltage and current across each RRAM device are calculated based on Kirchhoff's law and Ohm's law, and the set and reset transitions arise as a result of the migration of ionized defects as described in the analytical model.[21] The employed parameters, including the applied voltages, pulse widths, load resistances and also the gate voltages of transistors, are all the same as the experimental ones.

Supporting Information
Supporting Information is available from the Wiley Online Library or from the author.

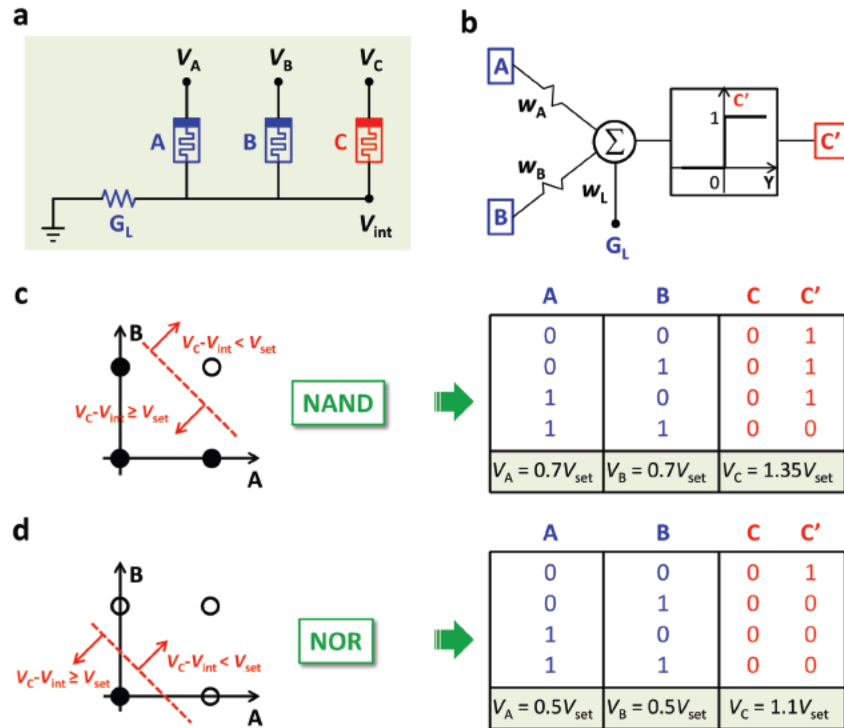
Acknowledgements
This article has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 648635).

Received: ((will be filled in by the editorial staff))
Revised: ((will be filled in by the editorial staff))
Published online: ((will be filled in by the editorial staff))

References

[1] M. M. Waldrop, *Nature* **2016**, *530*, 144.
[2] M. Horowitz, in *Int. Solid-State Circuits Conf. Dig. Tech. Papers*, **2014**, p. 10.
[3] H. S. P. Wong, S. Salahuddin, *Nat. Nanotechnol.* **2015**, *10*, 191.
[4] D. B. Strukov, G. S. Snider, D. R. Stewart, R. S. Williams, *Nature* **2008**, *453*, 80.
[5] J. Borghetti, G. S. Snider, P. J. Kuekes, J. J. Yang, D. R. Stewart, R. S. Williams, *Nature* **2010**, *464*, 873.
[6] S. Kvatinsky, G. Satat, N. Wald, E. G. Friedman, A. Kolodny, U. C. Weiser, *IEEE Trans. Very Large Scale Integr. Syst.* **2014**, *22*, 2054.
[7] B. Chen, F. Cai, J. Zhou, W. Ma, P. Sheridan, W. D. Lu, in *IEDM Tech. Dig.* **2015**, p. 17.5.1.
[8] S. Balatti, S. Ambrogio, D. Ielmini, *IEEE Trans. Electron Dev.* **2015***, 62*, 1831.
[9] P. Huang, J. Kang, Y. Zhao, S. Chen, R. Han, Z. Zhou, Z. Chen, W. Ma, M. Li, L. Liu, X. Liu, *Adv. Mater.* **2016**, *28*, 9758.
[10] M. Cassinerio, N. Ciocchini, D. Ielmini, *Adv. Mater.* **2013**, *25*, 5975.
[11] Y. Li, Y. P. Zhong, Y. F. Deng, Y. X. Zhou, L. Xu, X. S. Miao, *J. Appl. Phys.* **2013***, 114*, 234503.
[12] D. Loke, J. M. Skelton, W. -J. Wang, T. -H. Lee, R. Zhao, T. -C. Chong, S. R. Elliott, *Proc. Natl. Acad. Sci. USA* **2014**, *111*, 13272.
[13] D. Ielmini, *Semicond. Sci. Technol.* **2016,** *31*, 063002.
[14] R. Rojas, *Neural Networks: A Systematic Introduction*, Springer-Verlag, Berlin, Germany **1996**.
[15] W. S. McCulloch, W. Pitts, *Bull. Math. Biophys.* **1943**, *5*, 115.
[16] V. Beiu, J. M. Quintana, M. J. Avedillo, *IEEE Trans. Neural Netw.* **2003***, 14*, 1217.
[17] A. K. Maan, D. A. Jayadevi, A. P. James, *IEEE Trans. Neural Netw. Learn. Syst.* **2016***, 28*, 1734.
[18] M. D. Pickett, G. Medeiros-Ribeiro, R. S. Williams, *Nat. Mater.* **2013**, *12*, 114.
[19] T. Tuma, A. Pantazi, M. Le Gallo, A. Sebastian, E. Eleftheriou, *Nat. Nanotechnol.* **2016**, *11*, 693.
[20] D. Ielmini, *IEEE Trans. Electron Devices* **2011**, *58*, 4309.
[21] S. Ambrogio, S. Balatti, D. C. Gilmer, D. Ielmini, *IEEE Trans. Electron Devices* **2014**, *61*, 2378.
[22] M. L. Minsky, S. A. Papert, *Perceptrons*, MIT Press, Cambridge, MA, USA **1969**.
[23] M. Alioto, G. Palumbo, *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2002**, *10*, 806.
[24] D. M. Nikonov, I. A. Young, *Proc. IEEE* **2013***, 101*, 2498.
[25] D. Hampel, *IEEE Trans. Comput.* **1973**, *C-22*, 197.
[26] R. Waser, *Nanoelectronics and Information Technology* 3rd ed., John Wiley & Sons, Weinheim, Germany **2012**.
[27] S. Balatti, S. Ambrogio, D. Ielmini, *IEEE Trans. Electron Devices* **2015**, *62*, 1839.
[28] P. M. Sheridan, F. Cai, C. Du, W. Ma, Z. Zhang, W. D. Lu, *Nat. Nanotechnol.* **2017**, *12*, 784.
[29] A. Siemon, T. Breuer, N. Aslam, S. Ferch, W. Kim, J. van den Hurk, V. Rana, S. Hoffmann-Eifert, R. Waser, S. Menzel, E. Linn, *Adv. Funct. Mater.* **2015**, *25*, 6414.
[30] S. Gao, F. Zeng, M. Wang, G. Wang, C. Song, F. Pan, *Sci. Rep.* **2015**, *5*, 15467.
[31] J. Reuben, R. Ben-Hur, N. Wald, N. Talati, A. H. Ali, P. -E. Gaillardon, S. Kvatinsky, in *27th International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, **2017**, p. 1.
[32] E. Lehtonen, M. Laiho, in *Proc. IEEE/ACM Int. Symp. Nanosc. Archit.*, **2009**, p. 33.
[33] G. C. Adam, B. D. Hoskins, M. Prezioso, D. B. Strukov, *Nano Res.* **2016**, *9*, 3914.

[34] L. Cheng, M. -Y. Zhang, Y. Li, Y. -X. Zhou, Z. -R. Wang, S. -Y. Hu, S. -B. Long, M. Liu, X. -S. Miao, *J. Phys. D: Appl. Phys.* **2017**, *50*, 505102.
[35] S. Larentis, F. Nardi, S. Balatti, D. C. Gilmer, D. Ielmini, *IEEE Trans. Electron Devices* **2012**, *59*, 2468.
[36] S. Kim, S. -J. Kim, K. M. Kim, S. R. Lee, M. Chang, E. Cho, Y. -B. Kim, C. J. Kim, U. -I. Chung, I. -K. Yoo1, *Sci. Rep.* **2013**, *3*, 1680.
[37] S. Kim, S. Choi, W. Lu, *ACS Nano* **2014**, *8*, 2369.
[38] S. Ambrogio, S. Balatti, A. Cubeta, A. Calderoni, N. Ramaswamy, D. Ielmini, *IEEE Trans. Electron Devices* **2014**, *61*, 2912.
[39] D. Pradhan, *Fault-Tolerant Computer System Design*, Prentice-Hall, NJ, USA **1996**.
[40] J. H. Yoon, J. H. Han, J. S. Jung, W. Jeon, G. H. Kim, S. J. Song, J. Y. Seok, K. J. Yoon, M. H. Lee, C. S. Hwang, *Adv. Mater.* **2013**, *25*, 1987.
[41] S. Choi, S. H. Tan, Z. Li, Y. Kim, C. Choi, P. -Y. Chen, H. Yeon, S. Yu, J. Kim, *Nat. Mater.* **2018**, *17*, 335.
[42] D. Ielmini, H.-S. P. Wong, *Nat. Electron.* **2018**, *1*, 333.
[43] C. Li, M. Hu, Y. Li, H. Jiang, N. Ge, E. Montgomery, J. Zhang, W. Song, N. Dávila, C. E. Graves, Z. Li, J. P. Strachan, P. Lin, Z. Wang, M. Barnell, Q. Wu, R. S. Williams, J. J. Yang, Q. Xia, *Nat. Electron.* **2018**, *1*, 52.
[44] M. Prezioso, F. Merrikh-Bayat, B. D. Hoskins, G. C. Adam, K. K. Likharev, D. B. Strukov, *Nature* **2015**, *521*, 61.
[45] S. Yu, B. Gao, Z. Fang, H. Yu, J. Kang, H.-S. P. Wong, *Adv. Mater.* **2013**, *25*, 1774.
[46] H. Y. Lee, P. S. Chen, T. Y. Wu, Y. S. Chen, C. C. Wang, P. J. Tzeng, C. H. Lin, F. Chen, C. H. Lien, M. -J. Tsai, in *IEDM Tech. Dig.* **2008**, p. 1.



A	B	C	C'
0	0	0	1
0	1	0	1
1	0	0	1
1	1	0	0
$V_A = 0.7V_{set}$	$V_B = 0.7V_{set}$	$V_C = 1.35V_{set}$	

A	B	C	C'
0	0	0	1
0	1	0	0
1	0	0	0
1	1	0	0
$V_A = 0.5V_{set}$	$V_B = 0.5V_{set}$	$V_C = 1.1V_{set}$	

Figure 1. Fundamental concept of the neural-network-based stateful logic circuit. a) RRAM circuit with 2 input devices A, B, one output device C, and the load resistor of conductance G_L. b) Equivalent neural network model for the RRAM circuit, where conductance states of A and B serve as input variables, the conductance state of C serves as output variable, and the load resistor plays the role of bias in the neural network. Synaptic weights w_A, w_B, and w_L are dictated by the applied voltages according to Eq. (3). If the internal state function Y is larger than 0, the output C switches to the final state C' = 1 due to set transition. If Y is smaller than 0, the output C remains in its initial state, thus C' = 0. c,d) Input/output characteristics and truth table for NAND operation (c) and NOR operation (d). Full symbols and open symbols correspond to the output state being 1 and 0, respectively. The boundary lines for linear separation of outputs, and the voltage values applied to A, B, and C, to tune the synaptic weights, are also shown. G_L was chosen equal to $1.4G_{LRS}$ for both logic operations.

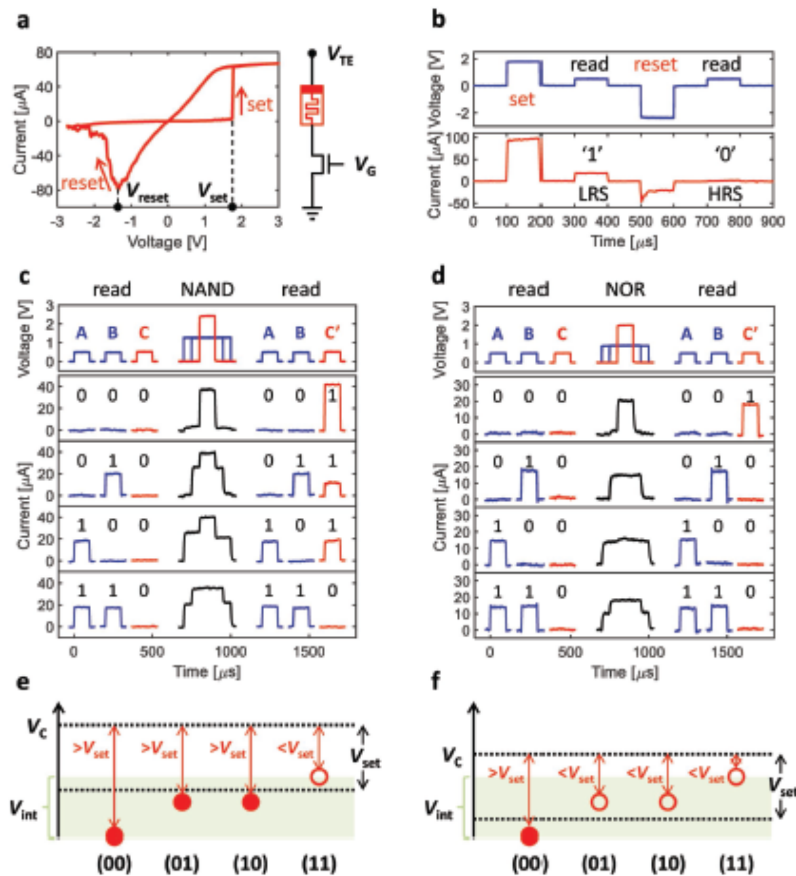


Figure 2. Experimental demonstration of NAND and NOR operations. a) *I-V* characteristics of the HfO_2 RRAM under quasi-stationary conditions and corresponding 1T1R structure of the device. The device show set and reset transitions at positive and negative voltages, respectively. b) Applied voltage (top) and current response of the RRAM device during set, reset and read operated by pulses of 100 µS width. c,d) Experimental demonstration of the NAND logic function (c) and NOR logic function (d), including applied voltages (top) and current response (bottom) for all configurations of input states. The sequence includes initial read, logic operation, and final read, for both input and output RRAM devices, the latter showing switching according to the truth table. During logic operations, the current through load resistor is collected. e,f) The internal potential V_{int} for NAND (e) and NOR (f) operations. The voltage across the output V_C-V_{int} is also shown, with the condition V_C-$V_{int} > V_{set}$ controlling the conditional set (C' = 1) in the output device.

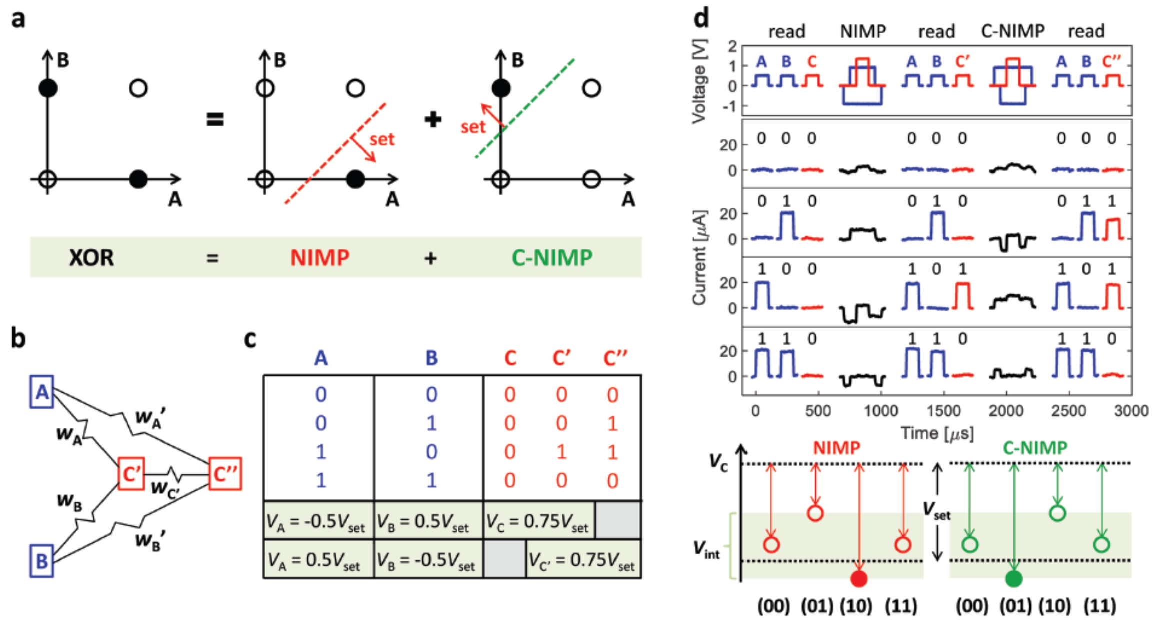


Figure 3. XOR operation with the stateful neural network. a) Input/output characteristics of the linearly non-separable XOR operation, which is disassembled into the sequence of the linearly-separable NIMP and C-NIMP operations. b) Schematic of the 2-layer perceptron network. c) Truth table for the XOR operation, with sequential output states C' and C'' corresponding to the result of NIMP and C-NIMP operations. The truth table also includes the applied voltages to tune the synaptic weights in the network. A load resistor with conductance $G_L = 0.5G_{LRS}$ is adopted in the network. d) Experimental demonstration of the XOR operation, showing the applied voltage and the current response for the 4 input configurations. e,f) The internal potential V_{int} for NIMP (e) and C-NIMP (f) operations, indicating set transition for V_C-$V_{int} > V_{set}$.

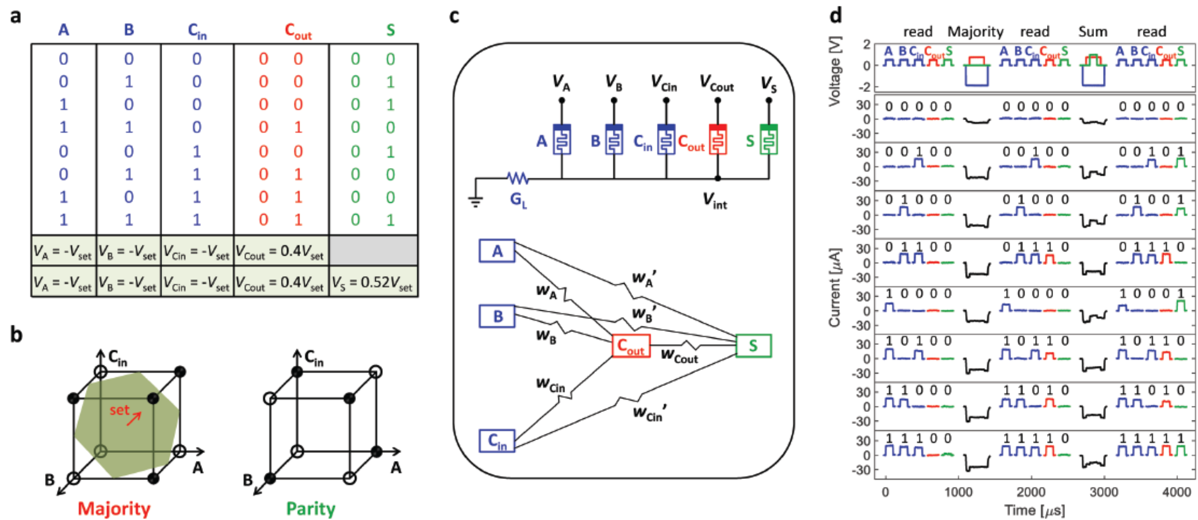


Figure 4. 1-bit FA with the stateful neural network. a) Truth table of the 1-bit FA, where the carry-out (C_{out}) and sum (S) are majority and parity functions of three inputs (A, B, C_{in}), respectively. The applied voltages for computing C_{out} and S are also included. A load resistor of conductance $G_L = 0.83G_{LRS}$ is used in the circuit. b) Input/output characteristics of linearly-separable majority function (left) and linearly non-separable parity function (right). c) RRAM circuit and the corresponding equivalent 2-layer perceptron network for the 1-bit FA. d) Applied voltages (top) and current response (bottom) for all 8 configurations of input A, B, and C_{in}.

Table 1. Comparison of stateful XOR and 1-bit FA schemes. The table reports various options for XOR and 1-bit FA according to the literature, comparing the number of RRAM devices and computing steps.

Reference	XOR		1-bit FA		Logic basis
	Devices	Steps	Devices	Steps	
[32]	-	-	8	89	IMPLY & FALSE
[33]	-	-	6	35	
[5],[34]	4	6	8	27	
[6]	5	13	6	29	
[7]	-	-	16	10	NOR & NOT
[8]	4	5	11	7	IMPLY & AND
[9]	4	4	9	10	NAND & AND
This work	3	2	5	2	Neural network

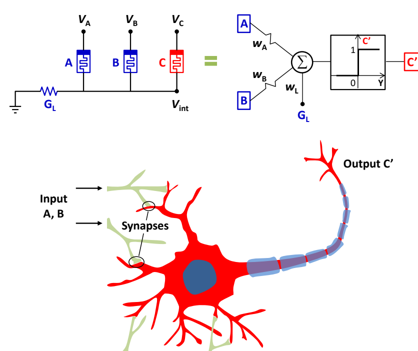
The concept of stateful neural network is introduced based on a resistive memory circuit. Thanks to the universality and flexibility of neural network, the circuit enables one-step operation for all linearly-separable logic functions, thus extremely reducing the numbers of computing steps and devices for stateful logic computing, for instance, 2 steps and 5 devices for the 1-bit full adder.

Keyword: stateful logic, neural network, resistive switching memory, in-memory computing, neuromorphic

Zhong Sun, Elia Ambrosi, Alessandro Bricalli and Daniele Ielmini*

Title: Logic computing with stateful neural networks of resistive switches

ToC figure



Copyright WILEY-VCH Verlag GmbH & Co. KGaA, 69469 Weinheim, Germany, 2016.

Supporting Information

Logic computing with stateful neural networks of resistive switches

*Zhong Sun, Elia Ambrosi, Alessandro Bricalli and Daniele Ielmini**

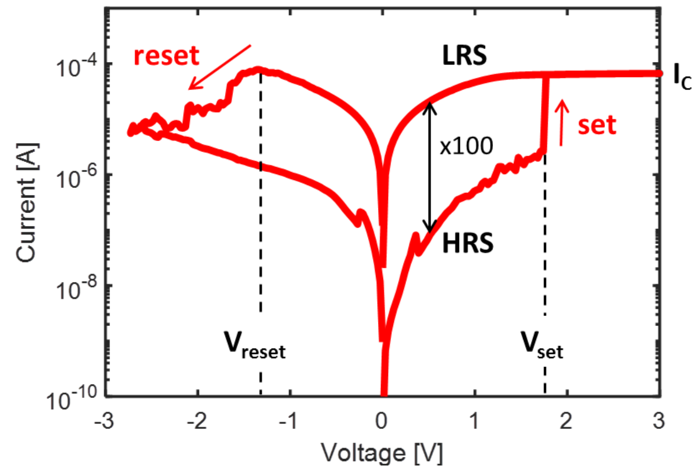


Figure S1. Current-voltage characteristics showing bipolar resistive switching of RRAM device in quasi-stationary (DC) mode. The positive voltage sweep triggers set transition from the high resistance state (HRS) to the low resistance state (LRS), when the voltage reaches a time-dependent threshold value (V_{set}). During the set transition, a compliance current (I_C = 70 μA in the figure) is provided by the integrated select transistor to protect the device from destructive breakdown. The negative voltage sweep triggers the reset transition from LRS to HRS when the voltage exceeds V_{reset}. The reset transition of resistance is more gradual compared to the set transition, in line with other results in the literature.[1-6] The resistance ratio between HRS and LRS is around 100 for the device shown in the figure.

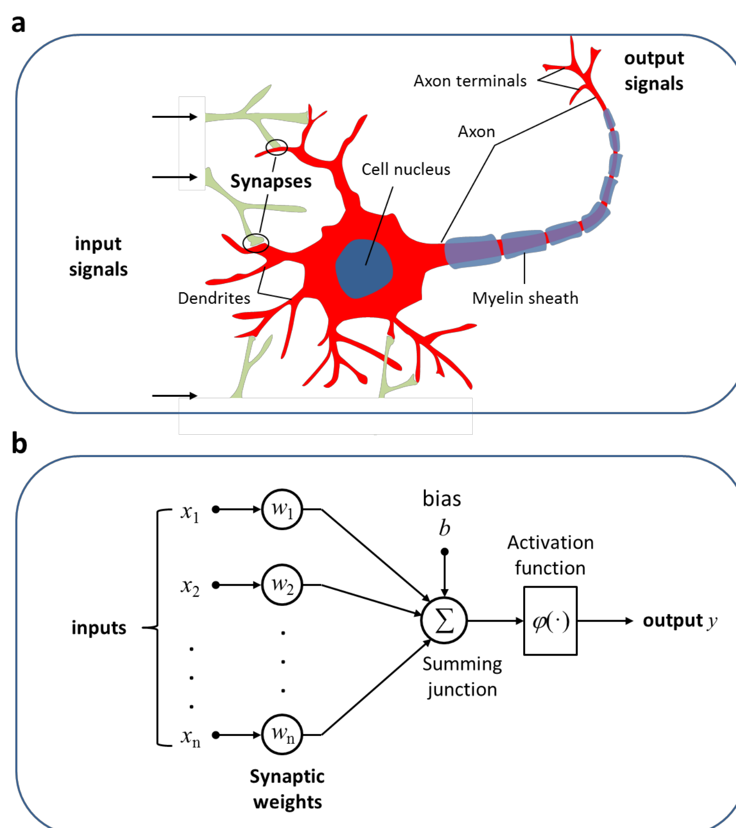


Figure S2. Sketch of a biological neuron (a) and of a neural network (b) within the McCulloch-Pitts model.[7] In the biological neuron, input signals reach the neuron soma from the synapses and dendrites. When the neuron fires, it sends an electrical spike through the axon and its terminals, thus stimulating other neurons. In the McCulloch-Pitts neuron, inputs are multiplied by synaptic weights w_i, then summed in the summing junctions before being subject to the activation function. A spike y is originated by the nonlinear activation function when the sum reaches a threshold.

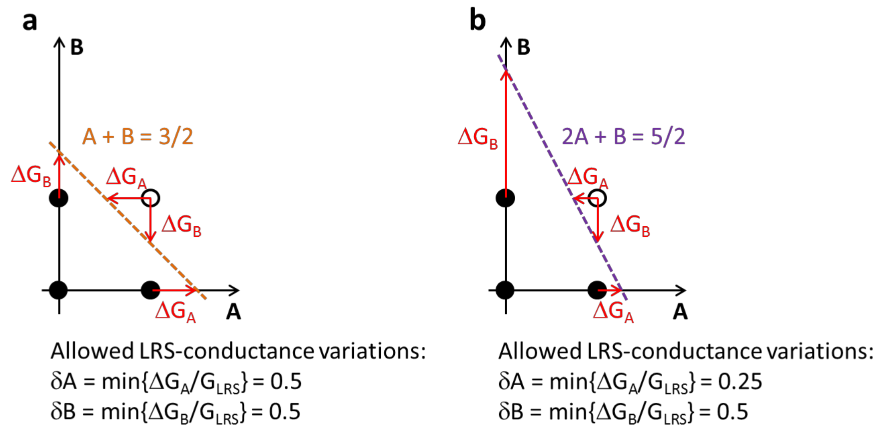


Figure S3. Input/output characteristics for various decision boundaries in NAND operation, comparing boundaries A + B = 3/2 (a) and 2A + B = 5/2 (b). The full symbols and open symbols represent output 1 and 0, respectively. In (a), the maximum conductance variation for an input device is allowed to be 50%, while in (b), the maximum variation is reduced to be 25% for input device A. The boundary line in (a) thus appears as the best choice to maximize the immunity to device variation.

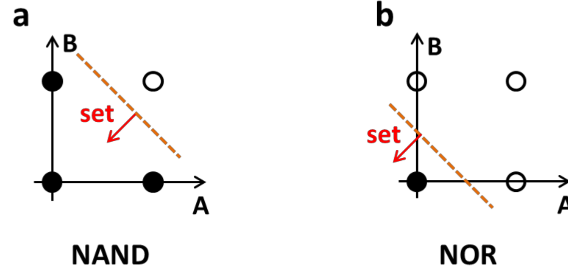


Figure S4. Input/output characteristics and decision boundaries for NAND logic function (a) and NOR logic function (b). The full symbols and open symbols represent output 1 and 0, respectively.

The domain of set transition for NAND function is A + B - 3/2 < 0, where the input states A and B are mapped by the conductance values G_A and G_B, respectively, and the real number 3/2 is mapped to $3G_{LRS}/2$, where G_{LRS} is the nominal LRS conductance, corresponding to a logical value of 1, thus leading to

$$G_A + G_B - 3G_{LRS}/2 < 0, \qquad \text{(S1)}$$

Following Eq. (2) in the main text, the internal state variable is given by the weighted sum:

$$Y = \Sigma_i G_i(V_C - V_i - V_{set}) = w_A G_A + w_B G_B + w_L G_L > 0 \qquad \text{(S2)}$$

where the inequality gives the condition for the set transition in the output RRAM. Note that $w_C G_C$ is neglected in Eq. (S2) since the output RRAM C is always prepared in HRS. Comparing Eqs. (S1) and (S2), we get:

$$w_A = w_B < 0 \qquad \text{(S3)}$$

$$w_L = -3w_A/(2G_L/G_{LRS}) \qquad \text{(S4)}$$

Following the weight definition in Eq. (3), we obtain:

$$V_A = V_B \qquad \text{(S5)}$$

$$V_C = V_A/(1+2G_L/(3G_{LRS})) + V_{set} \qquad \text{(S6)}$$

A possible solution is $G_L = 1.4G_{LRS}$, $V_A = V_B = 0.7V_{set}$ and $V_C = 1.35V_{set}$, which yields synaptic weights $w_A = w_B = -0.35V_{set}$, and $w_L = 0.35V_{set}$. Note that w_A and w_B are negative, and w_L is positive, consistent with the signs of inequalities in Eqs. (S1) and (S2) being opposite.

The values of the internal node potential V_{int} and the state variable $Y = \Sigma_i G_i w_i$ for various input configurations of A and B are the following:

Input (A,B)	V_{int}/V_{set}	$Y/(G_{LRS}V_{set})$
(0,0)	0	0.49
(0,1), (1,0)	0.29	0.14
(1,1)	0.41	-0.21

Therefore, only for the case (1,1), the state variable Y is negative, while for all other cases,

there is $Y > 0$, thus inducing set transition with C' = 1, which is consistent with the NAND operation.

The domain of set transition for NOR function is A + B - 1/2 < 0, so

$G_A + G_B - G_{LRS}/2 < 0,$ (S7)

Comparing Eqs. (S7) and (S2), we get:

$$w_A = w_B < 0 \quad \text{(S8)}$$

$$w_L = -w_A/(2G_L/G_{LRS}) \quad \text{(S9)}$$

Following the weight definition in Eq. (3), we obtain:

$$V_A = V_B \quad \text{(S10)}$$

$$V_C = V_A/(1+2G_L/G_{LRS}) + V_{set} \quad \text{(S11)}$$

A possible solution is $G_L = 1.4G_{LRS}$, $V_A = V_B = 0.5V_{set}$ and $V_C = 1.1V_{set}$, which yields synaptic weights $w_A = w_B = -0.4V_{set}$, and $w_L = 0.1V_{set}$. The load resistor is set as the same with NAND function for simplicity. Note that w_A and w_B are negative, and w_L is positive, which are consistent with the signs of inequalities in Eqs. (S7) and (S2) being opposite.

The values of the internal node potential V_{int} and the state variable $Y = \Sigma_i G_i w_i$ for various input configurations of A and B are the following:

Input (A,B)	V_{int}/V_{set}	$Y/(G_{LRS}V_{set})$
(0,0)	0	0.14
(0,1), (1,0)	0.21	-0.26
(1,1)	0.29	-0.66

Therefore, only for the case (0,0), the state variable Y is positive, thus inducing set transition with C' = 1, which is consistent with the NOR operation.

The following explains how we choose the conductance of load resistor for NAND operation. For NAND logic gate, the two input RRAM are applied with the same voltage, $V_A = V_B$. For input of (01) or (10), $V_{int} = V_A G_{LRS}/(G_L + G_{LRS})$, while for input of (11), $V_{int} = 2V_A G_{LRS}/(G_L + 2G_{LRS})$. For a specified V_A, to maximize the difference between the two V_{int} is to find out the maximum for the function below

$$f(x) = \frac{2}{x+2} - \frac{1}{x+1} \quad \text{(S12)}$$

Differentiating the equation, the maximum is found to locate at

$x_{\max} = \sqrt{2} \approx 1.4$

Therefore, the conductance of load resistor for NAND function is set as $G_L = 1.4G_{LRS}$.

Logic function	Schematic	Devices	Operation voltages
True		1	$V_C \geq V_{set}$
False		1	$V_C \leq V_{reset}$
Proposition A		2	$V_C = \frac{1}{2g_L+1}V_A + V_{set}$ $-\frac{(2g_L+1)(g_L+2)}{2g_L^2+3g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+2}\right) < V_A < 0$
Proposition B		2	$V_C = \frac{1}{2g_L+1}V_B + V_{set}$ $-\frac{(2g_L+1)(g_L+2)}{2g_L^2+3g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+2}\right) < V_B < 0$
Negation of A (NOT)		2	$V_C = \frac{1}{2g_L+1}V_A + V_{set}$ $\max\left\{0, -\frac{(2g_L+1)(g_L+1)}{2g_L^2+3g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+1}\right)\right\} < V_A < V_{set}$
Negation of B (NOT)		2	$V_C = \frac{1}{2g_L+1}V_B + V_{set}$ $\max\left\{0, -\frac{(2g_L+1)(g_L+1)}{2g_L^2+3g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+1}\right)\right\} < V_B < V_{set}$
Conjunction (AND)		3	$V_B = V_A, V_C = \frac{3}{2g_L+3}V_A + V_{set}$ $\max\left\{-\frac{g_L+1}{g_L}\lvert V_{reset}\rvert, -\frac{(g_L+3)(2g_L+3)}{2g_L^2+5g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+3}\right)\right\} < V_A < 0$
Disjunction (OR)		3	$V_B = V_A, V_C = \frac{1}{2g_L+1}V_A + V_{set}$ $\max\left\{-\frac{g_L+1}{g_L}\lvert V_{reset}\rvert, -\frac{(g_L+2)(2g_L+1)}{2g_L^2+3g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+2}\right)\right\} < V_A < 0$
Alternative denial (NAND)		3	$V_B = V_A, V_C = \frac{3}{2g_L+3}V_A + V_{set}$ $\max\left\{0, -\frac{(g_L+2)(2g_L+3)}{2g_L^2+5g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+2}\right)\right\} < V_A < V_{set}$
Joint denial (NOR)		3	$V_B = V_A, V_C = \frac{1}{2g_L+1}V_A + V_{set}$ $\max\left\{0, -\frac{(g_L+1)(2g_L+1)}{2g_L^2+3g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+1}\right)\right\} < V_A < V_{set}$

Logic function	Schematic	Devices	Operation voltages
Material implication (IMP)		3	$V_B = -\frac{2g_L-1}{2g_L+1}V_A, V_C = \frac{1}{2g_L+1}V_A + V_{set}$ $0 < V_A < \min\left\{ \frac{(g_L+1)(2g_L+1)}{2g_L^2+5g_L}V_{set}, \frac{(g_L+2)(2g_L+1)}{2g_L^2+3g_L}\lvert V_{reset}\rvert, \frac{g_L+2}{g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+2}\right), \frac{(g_L+3)(2g_L+1)}{2g_L^2+5g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+3}\right) \right\}$
Converse implication (C-IMP)		3	$V_B = -\frac{2g_L+1}{2g_L-1}V_A, V_C = -\frac{1}{2g_L-1}V_A + V_{set}$ $\max\left\{ -\frac{(g_L+1)(2g_L-1)}{2g_L^2+5g_L}V_{set}, -\frac{(g_L+2)(2g_L-1)}{2g_L^2+3g_L}\lvert V_{reset}\rvert, -\frac{(g_L+2)(2g_L-1)}{2g_L^2+g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+2}\right), -\frac{(g_L+3)(2g_L-1)}{2g_L^2+5g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+3}\right) \right\} < V_A < 0$
Material nonimplication (NIMP)		3	$V_B = -\frac{2g_L-1}{2g_L+1}V_A, V_C = \frac{1}{2g_L+1}V_A + V_{set}$ $\max\left\{ -\frac{(g_L+3)(2g_L+1)}{2g_L^2+5g_L}V_{set}, -\frac{(g_L+2)(2g_L+1)}{2g_L^2+5g_L}\lvert V_{reset}\rvert, -\frac{(g_L+2)(2g_L+1)}{2g_L^2+3g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+2}\right) \right\} < V_A < 0$
Converse nonimplication (C-NIMP)		3	$V_B = -\frac{2g_L+1}{2g_L-1}V_A, V_C = -\frac{1}{2g_L-1}V_A + V_{set}$ $0 < V_A < \min\left\{ \frac{(g_L+1)(2g_L-1)}{2g_L^2+5g_L}V_{set}, \frac{(g_L+2)(2g_L-1)}{2g_L^2+5g_L}\lvert V_{reset}\rvert, \frac{(g_L+2)(2g_L-1)}{2g_L^2+3g_L}\left(\lvert V_{reset}\rvert - \frac{V_{set}}{g_L+2}\right) \right\}$

Table S1. Summary of the 14 linearly separable Boolean logic functions which can be realized in one step with the neural network concept, and also (from left to right) their input/output characteristics, the overall number of input/output resistive switching devices, the voltage values to tune the synaptic weight according to the decision boundaries, and the V_A range which prevents unwanted changes of the input state. The true and false logic functions correspond to the set and reset operations, respectively, thus only one device is needed. The 2 propositions and the 2 negations each involve 2 devices, one for the input, and the other for the output. The other 8 logic functions involve 3 devices, namely 2 for the input and 1 for the output. Specific voltages are provided based on the decision boundary for each function. It is defined that $g_L = G_L/G_{LRS}$, where G_{LRS} is the conductance of LRS. The range of V_A is calculated to enable correct logic operations while preventing changes in the input states during logic operations. Approximately, the V_A range is (0, V_{set}) for logic operations of NOT, NAND, NOR, IMP and C-NIMP, and (V_{reset}, 0) for proposition, AND, OR, C-IMP and NIMP. To comply with the V_{set} variation of the output device, V_A should be chosen around the middle of the corresponding range.

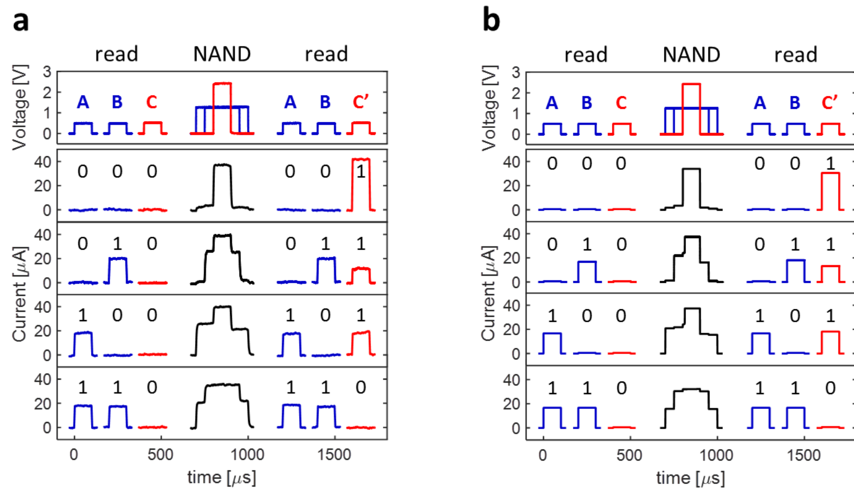


Figure S5. a) Experimental demonstration of the NAND operation. b) Simulation results obtained with an analytical model for RRAM devices.[3] The simulation parameters, including the values of the applied voltage during read and during the logic operation, the pulse widths, the load resistor conductance, are consistent with the experimental values. Very similar output states and currents are seen in the simulations and the experiments.

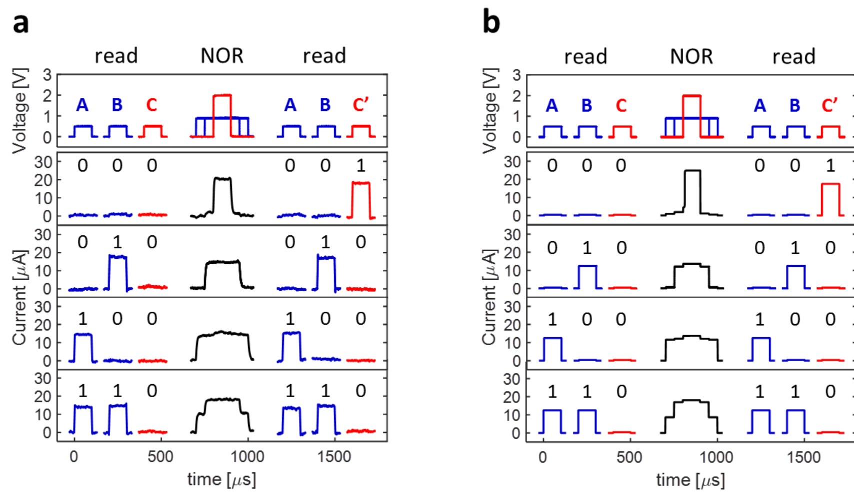


Figure S6. Same as Figure S5, but for the NOR operation.

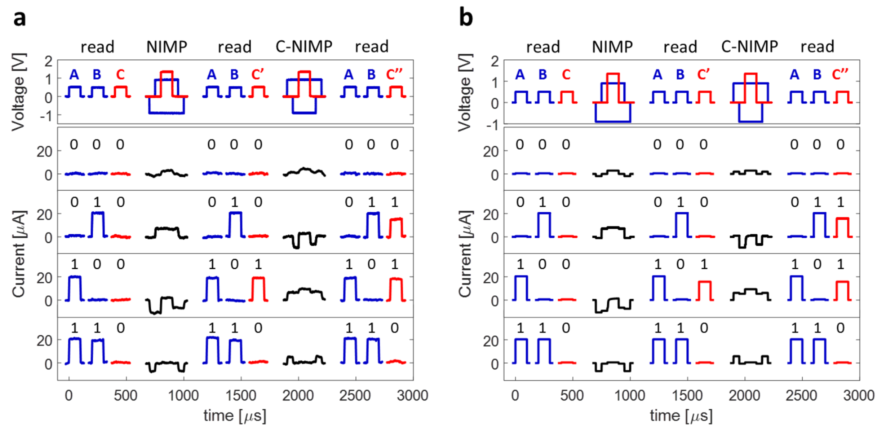


Figure S7. Same as Figures S5 and S6, but for the XOR operation. The linearly non-separable XOR was conducted by NIMP and following C-NIMP on the input RRAM A and B. The simulation results for the current response during the read and logic operations match the experimental data very well.

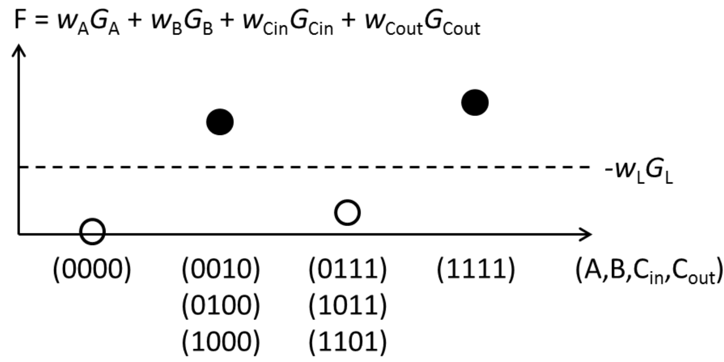


Figure S8. Calculated $Y = \Sigma_i G_i(V_C\text{-}V_i\text{-}V_{int})$ as a function of the configuration of the 4 input values (A, B, C_{in}, C_{out}) in computing sum (S) in the 1-bit FA. The full symbols and open symbols represent the output S being 1 and 0, respectively. Note that Y is larger than 0 for all configurations where the output S should be 1, which is marked as a full symbol in the figure, thus evidencing that S is a linearly-separable function of the inputs. The weights for each input is $w_A = w_B = w_{Cin} = 0.52V_{set}$, $w_{Cout} = -0.88V_{set}$. The load resistor has a conductance $G_L = 0.83G_{LRS}$, leading to a weight $w_L = -0.48V_{set}$. The corresponding voltages are $V_A = V_B = V_{Cin} = -V_{set}$, $V_{Cout} = 0.4V_{set}$, and $V_S = 0.52V_{set}$.

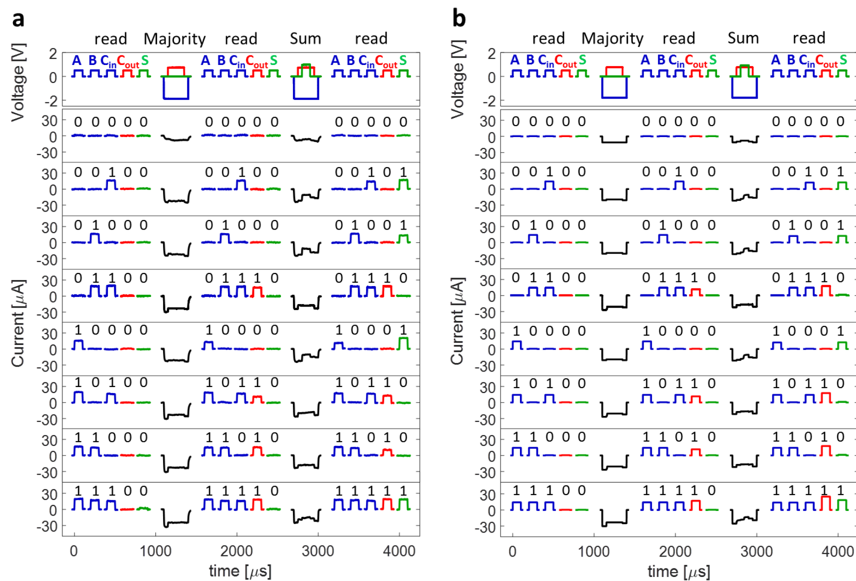


Figure S9. Same as Figure S7, but for the 1-bit FA operation. The 2 output variables are computed in 2 steps, namely C_{out} = majority (A, B, C_{in}) in the first step, followed by S in the second. The linearly-separable function to compute S is reported in Figure S8. The simulation results for the current response during the read and logic operations match the experimental data very well.

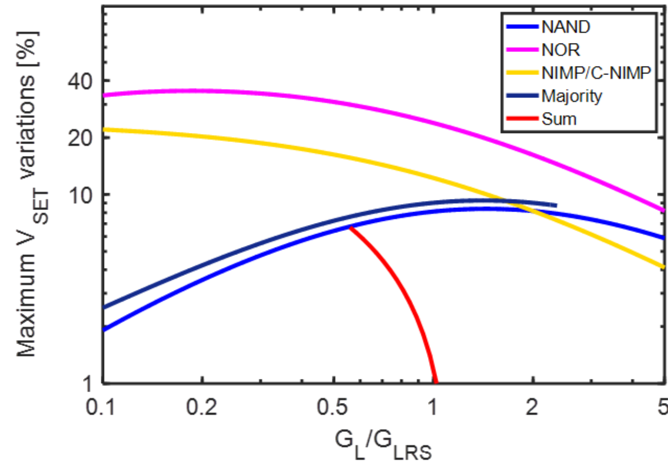


Figure S10. Calculated maximum V_{set} variation as a function of the load resistor conductance G_L normalized to G_{LRS}, for all logic operations demonstrated in this work. To calculate the maximum V_{set} variation, we assumed a resistance ratio between HRS and LRS of 100 (Figure S1). For a set of voltages V_i applied to the input devices and a specified load resistor G_L, the range of voltage on the output device that guarantees the correct operation without switching at the input RRAM was calculated. Half of the voltage range was taken as the maximum V_{set} variation for the given voltages and load resistance. For different load resistances ($0.1<G_L/G_{LRS}<5$), a curve of maximum V_{set} variations as a function of load resistance can be obtained. For different sets of voltages applied on input devices, different curves will be obtained, with the representative results reported in the figure.

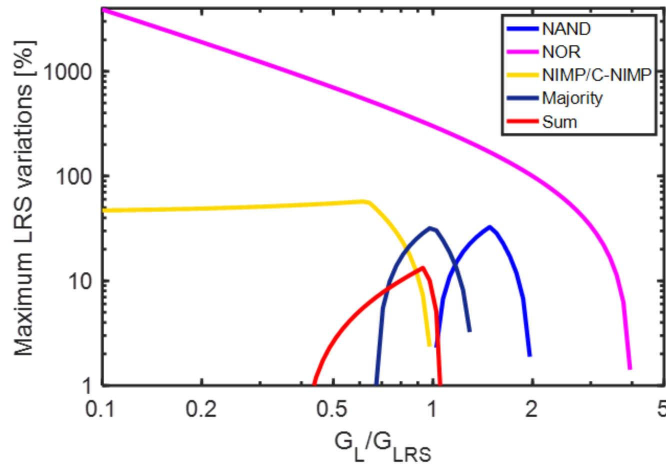


Figure S11. Calculated maximum LRS variation as a function of the load resistor conductance G_L normalized to G_{LRS}, for all logic operations demonstrated in this work. The maximum variation $\delta R_{LRS}/R_{LRS}$ is calculated by determining the LRS resistance range (R_{LRS}-δR_{LRS}, R_{LRS}+δR_{LRS}), in which the logic operations for all input configurations are correct for any LRS resistance value. For a set of voltages V_i applied to the input devices and a specified load resistor G_L, the maximum LRS variation was calculated. For different load resistances ($0.1<G_L/G_{LRS}<5$), a curve of maximum LRS variations as a function of load resistance can be obtained. For different sets of voltages applied on input devices, different curves will be obtained, with the representative results reported in the figure. For NOR operation, the LRS variation can be very large, but only for the positive direction.

References

[1] D. Ielmini, *Semicond. Sci. Technol.* **2016,** *31*, 063002.
[2] D. Ielmini, *IEEE Trans. Electron Devices* **2011**, *58*, 4309.
[3] S. Ambrogio, S. Balatti, D. C. Gilmer, D. Ielmini, *IEEE Trans. Electron Devices* **2014**, *61*, 2378.
[4] S. Larentis, F. Nardi, S. Balatti, D. C. Gilmer, D. Ielmini, *IEEE Trans. Electron Devices* **2012**, *59*, 2468.
[5] S. Kim, S. -J. Kim, K. M. Kim, S. R. Lee, M. Chang, E. Cho, Y. -B. Kim, C. J. Kim, U. -I. Chung, I. -K. Yoo, *Sci. Rep.* **2013**, *3*, 1680.
[6] S. Kim, S. Choi, W. Lu, *ACS Nano* **2014**, *8*, 2369.
[7] W. S. McCulloch, W. Pitts, *Bull. Math. Biophys.* **1943**, *5*, 115.