In-memory computing with resistive switching devices

Daniele Ielmini[1] and H.-S. Philip Wong[2]

[1]Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano and IU.NET, Piazza L. da Vinci 32 – 20133 Milano, Italy. Email: daniele.ielmini@polimi.it

[2]Department of Electrical Engineering and Stanford SystemX Alliance, Paul G. Allen Building 312X, 420 Via Palou, Stanford University, Stanford, CA 94305. Email: hspwong@stanford.edu

Modern computers are based on the von Neumann architecture in which computation and storage are physically separated: data are fetched from the memory unit, shuttled to the processing unit (where computation takes place) and then shuttled back to the memory unit to be stored. The rate at which data can be transferred between the processing unit and the memory unit represents a fundamental limitation of modern computers, known as the memory wall. In-memory computing is an approach that attempts to address this issue by designing systems that compute within the memory, thus eliminating the energy-intensive and time-consuming data movement that plagues current designs. Here we review the development of in-memory computing using resistive switching devices, where the two-terminal structure of the devices and the direct data processing in the memory can enable area- and energy-efficient computation. We examine the different digital, analogue, and stochastic computing schemes that have been proposed, and explore the microscopic physical mechanisms involved. Finally, we discuss the challenges in-memory computing faces, including the required scaling characteristics, in delivering next-generation computing.

Over the past 50 years, progress in computing and information technology was based on the downscaling of the metal-oxide-semiconductor field-effect transistor (MOSFET), which served as the workhorse of the semiconductor industry for analogue and digital circuits. This downscaling enabled digital CMOS systems to sustain an exponential increase of the operating frequency and the number of devices per area at each technology generation [1]. Today, however, the operation frequency and device density have reached a plateau, which stems from at least two barriers: the dissipated power is so large that the temperature increase on the chip cannot be sustained without significant performance degradation [2]; and there exists an increasing performance gap between the central processing unit (where the data is processed) and the computer memory (where data is stored), which is known as the memory wall [3]. It has been evaluated that, for many computing tasks, most of the energy and time are consumed in data movement, rather than computation [4]. These problems are expected to be exacerbated as applications become more data centric, where computing tasks consists of machine-learning operations such as object, image, and speech recognition.

Modern technologies are tackling these barriers from many angles, from the component level to the systems architecture design. Measures include the extensive use of parallelism, such as the graphics processing unit (GPU), which enhances the parallelism by using many cores (even more than 100), each with a dedicated or shared high-throughput connection with the memory. Also, application-specific processors known as accelerators are designed to match the exact computing algorithms and data flow [5]. For instance, the tensor processing unit (TPU) has been recently developed for accelerating the multiply-accumulate (MAC) operation, which constitutes the major workload in the inference phase of neural networks in data centers for image and speech recognition [6]. Another solution is the introduction of memory chips with enhanced bandwidth, such as the hybrid memory cube (HMC) [7], and high bandwidth memory (HBM) [8], where high data-transfer rate and high memory density are achieved by stacking multiple memory chips with through-silicon via (TSV) interconnect.

New and emerging nonvolatile memory concepts have also been introduced into the traditional memory hierarchy to reduce the ‘distance’ between computing and the data [9]. These new memories, which are grouped under the name ‘resistive switching devices’ in this work, have unique storage principles which are not based on charge, as in conventional Flash memory and random access memory (RAM), for example, static RAM (SRAM) and dynamic RAM (DRAM). The storage concept relies instead on the physics of the active materials and the device where they are integrated. These memories include resistance switching RAM (RRAM) [10], phase change memory (PCM) [11], magneto-resistive RAM (MRAM) [12], and ferroelectric RAM (FeRAM) [13]. Although some of these memories have led to commercial technologies which are available on the market [14], they are still too slow, have limited data bandwidth, or are too expensive, to significantly contribute to solve the memory bottleneck.

Instead of re-engineering conventional systems by individual improvements in parallelism, memory bandwidth, or memory concept- in-memory computing aims to radically subvert the von Neumann architecture by carrying out calculations *in situ*, exactly where the data are located [15]. This approach is

similar to the computing scheme in the human brain, where information is processed in sparse networks of neurons and synapses, without any physical separation between computation and memory [16]. In-memory computing offers a clear advantage by totally removing the latency and energy burdens of the memory wall. However, this new architecture requires computational memory devices that can both store data and compute at the same time, usually by device physics or other physical laws, such as the Ohm's law and the Kirchhoff's law in electrical circuits.

Here we review the in-memory computing schemes that have been proposed in both digital and analogue spaces, covering the device physics, the processing algorithms, and the circuit architectures that perform computing tasks within memory.

Computational memory technologies

In-memory computing generally requires fast, high density, low power, scalable memory devices, such as RRAM, PCM, MRAM, and FeRAM sketched in Fig. 1. All these devices are 2-terminal elements, where the application of a voltage results in a change of the materials property. For instance, RRAM (Fig. 1a) consists of a metal-insulator-metal (MIM) stack, where a filamentary path is initially created by soft electrical breakdown, or forming, induced by the application of a voltage. The large concentration of defects, e.g., oxygen vacancies in metal oxides [17] or metallic ions injected from the electrodes [18], are then driven by field-induced migration and diffusion in this conductive filament (CF) [19]. Application of a positive voltage to the top electrode, where the defects are concentrated with higher density, induces defect migration toward the bottom electrode, thus causing the transition to the low resistance state (LRS), due to enhanced conduction at defect sites. Application of a negative voltage induces defect migration back to the top electrode, thus causing the transition to the high resistance state (HRS) due to the disconnection of the CF. These transitions can be seen in the idealized current-voltage (I-V) characteristic in Fig. 1b, where the set transition to the LRS and the reset transition to the HRS occur at opposite voltages. Similar to the bipolar RRAM concept in Fig. 1b, unipolar RRAMs have also been presented, where the set and reset processes both occur under the same voltage polarity because of the dominant role of Joule heating in creating and dissolving the CF [20,21]. Also, non-filamentary switching has been demonstrated in RRAM by interface switching, where the voltage-induced defect migration result in a uniform change of a Schottky or tunneling barrier across the whole device area [22]. All these devices rely on the diffusion and migration of defects and will be referred to as RRAM throughout this Review Article.

In PCM (Fig. 1c), the active material is a chalcogenide phase change material, such as $Ge_2Sb_2Te_5$ [23], which can remain in either crystalline or amorphous states for long periods of time, e.g., 10 years at moderately high temperature. Starting from the amorphous state, the application of voltage pulses with relatively low amplitude causes the crystallization induced by Joule heating (Fig. 1d), whereas the application of pulses at higher amplitudes can lead to local melting and consequent amorphization. The crystalline phase has a low resistance because of the large concentration of carriers, while the amorphous

phase has a high resistance due to its intrinsic semiconductor nature originating from Fermi level pinning at mid gap [24]. A typical PCM cell has a mushroom shape shown in Fig. 1c, where the pillar-like bottom electrode confines heat and current, thus resulting in a hemispherical shape of the molten material. Pore-like PCM cells have been shown to be more energy efficient and more scalable than mushroom cells, thanks to a better confinement of heat and current [25].

RRAM and PCM share many similar features, which make them quite promising technologies for in-memory computing. First, they are both nonvolatile, and the resistance ratio is generally above a factor 10, which allows clear discrimination between digital 0 and 1, as well as making multilevel operation possible. Both devices operate at moderate-high switching speed (typically below 100 ns and even in the sub-ns regime [26,27]). Finally, they both display a better endurance as compared to the conventional Flash storage devices [28]. On the other hand, the morphology of materials change is different in the two cases, namely filamentary switching in the case of RRAM versus thermally-induced volume change in PCM. Also, the switching phenomena are different in that RRAM states are chemically distinct because of redox reactions and migration, while the PCM phases are only physically different (i.e., no change of materials composition).

Fig. 1e shows a magneto-tunnel junction (MTJ), which is the building block for most MRAM devices. The MTJ consists of an MIM structure with two ferromagnetic metal layers, e.g. the CoFeB alloy, and a thin tunnel oxide, e.g., MgO. The two ferromagnetic layers are referred to as either the pinned layer (where the magnetic polarization is structurally fixed to act as a reference) or the free layer (where the magnetic polarization is free to change upon programming). The two ferromagnetic polarizations can thus be either parallel (same direction) or antiparallel (opposite direction), which results in a low and a high resistance of the MTJ, respectively, due to the tunnel magneto-resistive (TMR) effect [29]. To flip the state of the MTJ, the spin transfer torque (STT) has emerged in the recent years as a scalable, low energy mechanism [30]. In STT-MRAM, the transition to the parallel state takes place directly by conduction electrons, which are first spin polarized by the pinned layer, then rotate the free layer magnetic polarization by magnetic momentum conservation [31]. To rotate the free layer magnetization in the antiparallel state, an opposite voltage, hence current direction, is needed, as shown in Fig. 1f. The relative change of resistance, also called magneto-resistance ratio, is typically around 200%, or a factor of three [32]. STT-MRAM is characterized by a high switching speed, which can lower than 1 ns, and by a high endurance above 10^{14} [33].

RRAM, PCM, and MRAM are all resistive switching memories, as the physical switching is reflected in a change of resistance. On the other hand, FeRAM in Fig. 1g relies on the polarization switching in a ferroelectric material, such as a perovskite material [13], or doped-HfO_2 [34]. Here, the individual ferroelectric dipoles change their orientation in response to the voltage applied to the MIM stack (Fig. 1h). Instead of impacting the MIM stack resistance, ferroelectric switching changes the charge induced on the metallic electrodes of the MIM capacitance, which can be sensed by integrating the current over a voltage sweep. To gain a resistance change by ferroelectric switching, one should adopt a ferroelectric field-effect transistor (FeFET) structure with three terminals, where the change in dielectric polarization causes a

variation in the resistance of the FeFET channel [35]. The resistance change in the FeFET thus enables in-memory computing elements such as neuromorphic synapses [36].

Digital computing by binary resistive switching

In the last 20 years, in-memory digital computing has focused on identifying novel logic gate concepts with lower energy and area consumption. Digital computing with nanomagnets [37], quantum dots [38], and even single atoms [39] have been demonstrated experimentally, although the control of the individual cells via voltage and current signals appears challenging. Resistive switching devices, such as RRAM, offer several advantages in digital computing, such as the direct access by interconnect lines, the capability to electrically reconfigure the device, and nanoscale miniaturization [40]. Fig. 2 shows various options to carry out digital Boolean operations with RRAM, differing by the type of input, the type of output, and the physical operation to describe the logic function. In the logic gate of Fig. 2a, the 2 input states X_1 and X_2 are represented by the voltage values applied to the top and bottom electrodes, respectively, while the output of the logic operation is stored as the resistance of the physical element, therefore the scheme will be referred to as the V-R logic gate [41]. The computing element is a bipolar-switching RRAM device, where the application of a positive voltage to the top electrode leads to a transition to the HRS, and the application of a negative voltage to the top electrode leads to a transition to the LRS. (To represent the switching polarity, the RRAM device is drawn as an arrow pointing to the electrode being biased to negative during set transition to the LRS). The output of the computation is the resistive state, namely a logic value 0 for HRS, and 1 for LRS, where the RRAM device is initially prepared in state 1. The logic gate behaves as follows: if the input logic voltages are equal, namely $X_1 = X_2 = 0$, or $X_1 = X_2 = 1$, then the overall voltage drop across the RRAM device is zero, thus the RRAM state remains unchanged (Y = 1, see the truth table in Fig. 2b). On the other hand, the configuration $X_1 = 1$ and $X_2 = 0$ causes a transition to the HRS, hence Y = 0. Finally, the configuration $X_1 = 0$ and $X_2 = 1$ is ineffective as the device is already in state 1. The resulting logic operation is the material implication (IMP), where the output is always 1, except for the condition $X_1 = 1$ and $X_2 = 0$, where the logic implication is not satisfied. Since IMP is functionally complete, all 16 Boolean functions can be realized by suitable combinations of more IMP operations [41]. The V-R gate can also be generalized as a majority gate if the initial state of the RRAM device is also considered as a possible input value [42]. The V-R concept can also be generalized to serial/parallel arrangements of more resistive switches to perform conventional logic operations (e.g., AND) in just one step [43].

The V-R logic gate in Fig. 2a is a nonvolatile concept, in that the output state remains stored as the resistive state without any voltage bias, thus allowing a considerable saving of static power. On the other hand, the sequential cascade of 2 operations, where the first gate's output is directly used as the second gate's input, is impossible, as input and output signals are physically different [44]. Converting the output resistance into an input voltage can be achieved by additional circuits, typically located out of the memory area, which however increase the size, complexity and power consumption of the computing system. As a

result, the V-R gate cannot be ascribed to fully in-memory logic schemes, as the computing flow must 'exit' the memory circuit to convert resistive states into input voltage values.

Fig. 2c shows the V-V logic, also referred to as threshold logic unit. In the V-V logic gates, both input and output values are described by digital voltages, being either low or high to represent a 0 or a 1, respectively [45]. The V-V logic gate can be viewed as a one-layer neural network, where any input voltage V_j stimulates a current I_j given by Ohm's law $I_j = G_j(V_j-V_{com})$, where G_j is the conductance of the j-th resistive switch and V_{com} is the potential of the common node in Fig. 2c. Currents are then summed by Kirchhoff's law at the common node, thus leading to $V_{com} = R_L\Sigma I_j = R_L\Sigma G_j(V_j-V_{com})$, where R_L is the load resistance connecting the common node to ground. The common voltage is thus given by $V_{com} = \Sigma G_jV_j//(1/R_L+\Sigma G_j)$, which describes a weighted sum of input voltages. The common node is usually connected to a rectifying stage, such as a comparator, which restores a digital value for the output V_{out}, given by $V_{out} = f(V_{com}-V_T)$, where f is a highly-nonlinear function, and V_T is an internal threshold voltage, hence the name 'threshold logic'. The Boolean function is thus described by the input/output characteristic in Fig. 2d, where all input values are linearly separated between configurations yielding output 0 or 1. The position of the separating line is dictated by the weights G_j and by V_T, which are carefully tuned to obtain any generic linearly-separable Boolean function (AND in Fig. 2d) [46]. Similar to the V-R logic scheme, the comparator is a relatively massive circuit that must be located out of the memory area. Also, each input voltage must be obtained by a conversion from a stored value (typically a resistance state R), while the resistive switch simply stores the weight for the logic operation, i.e., part of the information required to execute a program code, rather than the input/output values themselves. On the other hand, cascading is possible in V-V logic, as input and output voltages share the same physical nature and amplitude range.

Fig. 2e shows the R-R logic, where both input and output values are the resistive states of the memory elements, and the logic operation is carried out within the memory [47]. This is a true, cascadable in-memory operation, which is also referred to as stateful logic [48,49], as it relies on the nonvolatile states of the resistive elements. Similar to the V-V logic, logic computation is carried out based on physical laws, such as Ohm's and Kirchhoff's laws, except that the comparator function is taken by one of the resistive units in the logic gate, thus enabling true in-memory computation. For instance, in the IMP gate of Fig. 2e, two resistive switches in parallel configuration with input states X_1 and X_2 are biased with voltages $V_{set}-\Delta$ and $V_{set}+\Delta$, respectively, where V_{set} is the nominal voltage to induce a set transition, and Δ is a relatively small fraction, *e.g.*, 10%, of V_{set}. In case X_1 is in the HRS (input logic value 0), the voltage $V_{set}+\Delta$ will drop entirely across X_2, thus leading to an unconditional set operation. As a result, the output state, which is X_2 at the end of the computation, or X_2', is unconditionally equal to 1 (see the truth table in Fig. 2f). On the other hand, if X_1 is in LRS (input logic value 1), the voltage across X_1 and X_2 will be only 2Δ, thus insufficient for the switching of either X_1 or X_2, i.e., the input states will remain unchanged, thus resulting in the IMP operation. More Boolean functions can be obtained by sequentially repeating IMP on more devices [47].

Similar stateful logic gates were proposed by changing the circuit architecture, e.g., adopting a serial switch arrangement [50], or more resistive switches in parallel [51, 52]. For instance, Fig. 2g shows an OR logic gate consisting of two serially-connected resistive switches, where the intermediate node is left floating, i.e., free to change its potential according to the voltage divider made by the two switches. If the two input states are equal, *e.g.*, $X_1 = X_2 = 0$, then the voltage divides equally across the two devices, thus remaining below the threshold V_{set} for set transition. On the other hand, if only one of the two input devices is high, the other input with low conductance will have a large voltage drop across it, thus inducing set transition. This operation yields an OR function with either switch serving as the output. Other functions, such as IMP and logic inversion (NOT), can be realized with the same architecture but different applied voltages [50]. The R-R logic concept has been extended to other memory devices, such as PCM [53] and STT-MRAM [54], thus confirming the universal application of digital in-memory computing.

Stateful R-R logic gates have several advantages over V-V and V-R schemes, including the possibility of sequentially cascading multiple operations, the reconfiguration of the Boolean function by the applied voltage pulses, and the true in-memory processing capability. Both the data and the code, containing the type of operation and the data address, can be stored in the same memory circuit, e.g., a crosspoint RRAM array. The code can be read and executed on data within the memory, thus overcoming the typical memory bottleneck of today's computing architecture [55]. A key limitation of in-memory digital computing is the time and energy burden due to the physical switching process within the device. Resistive switching in RRAMs today requires at least a voltage of about 1 V, with a current consumption in the range of 10 μA and a time of about 10 ns, which yields an energy of approximately 0.1 pJ per operation. For comparison, this is the same energy that is consumed in the 45 nm CMOS technology for a 32-bit integer addition, which consists of many individual Boolean logic steps [4]. Even lower energy consumption is estimated for advanced technology nodes, *e.g.*, only few fJ for an 8-bit addition [56] in a 7 nm CMOS generation. Such a large gap stems from the physics of the device itself, which involves electron drift and capacitive charging in CMOS logic gates, while RRAM relies on the motion of ionic species, which require a relatively large local electric field and temperature by Joule heating for their hopping migration [19]. The high electric field and local temperature in RRAM switching also results in significant degradation, eventually inducing an irreversible breakdown of the MIM interfaces [57]. This is a major limitation compared to charge-based CMOS logic circuits, where device degradation is almost negligible within the expected lifetime, *e.g.*, 10^{16} cycles in the case of a SRAM embedded in the same chip as the CPU. The relatively large operating current can cause unwanted ohmic voltage drops along the signal line, which can be avoided by increasing the width of the metallic interconnect, thus losing some of the high-density advantages of the crosspoint memory architecture. An additional area overhead is taken by the periphery control logic, including latches for synchronous propagation of the signal and row/column multiplexers. Either issues have not been adequately addressed in the literature so far.

Computing by cumulative resistive changes

Digital computing with RRAM generally takes advantage of binary transition, e.g., a set transition from HRS to LRS. More degrees of freedom can be gained by extending to the multilevel domain, where the application of repeated pulses induces a controllable, fractional variation of the device resistance. This is shown in Fig. 3 for a PCM device, where the set transition consists of a gradual crystallization of an amorphous region, and an increasing crystallization fraction results in a decreasing resistance. Fig. 3a shows the simulated temperature profile during programming and the distribution of amorphous/crystalline phases within a PCM device for an increasing crystallization time [53]. As the time increases, the thickness of the amorphous material decreases because of crystallization, thus causing a decrease of the threshold voltage V_T and the device resistance.

Gradual resistive switching is a key concept for analog computing. It can, for example, enable arithmetic summation, Fig. 3b [57]. Here, addition 3 + 4 is carried out in a single PCM element, used as a nanoscale abacus [58]. First, the two limits of the conductance scale are defined, namely, the HRS corresponding to the fully amorphized PCM after the reset operation, and the LRS corresponding to the partial crystalline state obtained after the repetition of N pulses, e.g., N = 8 in Fig. 3b. The device is thus initialized in the HRS with resistance R_0, and a number of pulses corresponding to the first addend and the second addend are applied, i.e., 3 pulses in the first stage, and 4 pulses in the second stage. Finally, the number of pulses to reach the LRS resistance R_8 is evaluated by a program/verify loop, which yields the N-complement of the correct solution, e.g., 1 = 8-(4+3) in Fig. 3b [57].

The concept of the accumulating PCM counter can be extended to several applications in the realm of analog computing, such as the decomposition in prime factors [59], the gradual potentiation of a PCM artificial synapse [60,61], and the logic summation, namely digital OR within a single PCM [53]. In the latter case, the digital operation is executed by sequential pulses, rather than a single step as in Fig. 2, where each pulse is counted by the PCM for digital addition in a cascadable R-R logic gate. Summation can also be extended to the concept of integration of analog pulses, or spikes, which is an essential feature of integrate-and-fire neurons in spiking neural networks [57,62]. This is illustrated by Fig. 3c, showing the concept of a McCulloch-Pitts neuron receiving weighted spiking signals from several pre-synaptic sources. The spikes are summed, *e.g.*, by Kirchhoff's law summation of currents at the input of the neuron circuit, and integrated within a PCM, as shown in Fig. 3d [62]. The PCM accumulates the incoming spikes, eventually hitting the LRS threshold which triggers the generation of a fire event, namely an output spiking signal. A key advantage of the PCM neuron is its nanoscale miniaturization, as opposed to the conventional capacitor-based charge integration, where the capacitor typically occupies a relatively large area of the circuit, e.g., about 60 $\mu m^2/pF$ in a 28 nm CMOS technology [63], thus limiting the maximum number of neurons in a neuromorphic chip. Similar concepts of integrating neurons were shown by adopting threshold switching in a Mott insulator [64] or volatile RRAM with unstable Ag filaments [65]. In these works, the spike-accumulating device spontaneously returns to the off-state after fire, in contrast to the PCM neuron which

must instead be reset to return to the off-state [62]. Pulse accumulation in nanoscale elements has been demonstrated in RRAM devices, where repeated pulses cause incremental reset transition because of the gradual increase of the depleted gap disconnecting the conductive path [66]. Cumulative change of resistance has been adopted for analog synapses in artificial neural networks using both filamentary [67-69] and interface-switching RRAM devices [70], as well as ferroelectric devices [71] and domain-wall STT-MRAM devices [72]. Weight update in floating gate synapses has also been demonstrated by charge integration [72], although this implementation suffers from typically large voltage and expensive double-poly integration process.

<-- Box 1:

Stochastic random bit generation

A significant drawback of resistive memory devices for both memory and computing is their stochastic variation, which originates from the microscopic switching mechanisms. For instance, set transition in a RRAM consists of the field- and temperature-activated migration of defects, each of them moving along different paths and correspondingly different energy barriers. As a result, the set voltage changes from cycle to cycle even for the same device, as a result of the multiple variables (migration barriers, local configurations, concentration gradient, etc.) affecting the set process. In addition, the conductive filament formed by the set transition also changes from cycle to cycle, thus causing a variable resistance of the LRS [74] from cycle to cycle. Similarly, the reset voltage and the HRS resistance experience stochastic variations which cause errors in the digital operations in Figs. 2 and 3. For instance, the IMP logic gate in Fig. 2c relies on the repeatability of V_{set}, in terms of both the cycle-to-cycle variability, and a suitable degree of matching of device characteristics between the two cells of the logic gate. Similarly, the accuracy of the PCM arithmetic adder in Fig. 3b depends on the repeatability of pulse-induced crystallization, which is inherently stochastic due to the random atomic configurations within the amorphous phases [75].

Variability can be turned into a resource in a random number generator (RNG). Although RNG is not strictly an in-memory computing tool, it has an important function in cryptography and data security. The generation of random keys is also instrumental in the physical unclonable function (PUF)- a one-way function for the authentication of hardware chips [76]. The PUF provides a response to an external challenge, the function generating the response cannot be captured or cloned, thus preventing chip counterfeiting and hacking [77]. RNG is also an enabling tool in probabilistic spiking neural networks, where noise is used as a resource to mimic the stochastic release of synaptic neurotransmitter, or the stochastic opening and closing of membrane channels [78]. The conventional schemes for generating random numbers usually rely on software and hardware techniques, which create a seed-dependent stream of deterministic pseudo-random numbers [79]. To develop a true RNG (TRNG), a physical entropy source is needed, *e.g.*, noise or variability of switching phenomena in memory devices.

Fig. 4 shows examples of stochastic phenomena in RRAM and their exploitation for RNG. Random telegraph noise (RTN) in Fig. 4a is a typical phenomenon taking place in either HRS or LRS; RTN results

from metastable defect fluctuations near the conductive path [80]. RTN appears as a random change of the current from a low value I_0 to a high value I_1, thus sampling the current at random time yields a bimodal probabilistic distribution as shown in Fig. 4b. The random bit can thus be generated by reading the current value, and attributing I_0 to bit 0, and I_1 to bit 1 [81]. The RTN site is generally difficult to control in terms of both amplitude and uniformity, i.e., the sub-distributions in Fig. 4b should be equal to ensure a 50% probability of generating either 0 or 1. RTN is also affected by temperature and applied bias, which also leads to drift and instability of the RTN entropy source.

The quality of the generated random numbers can be improved by exploiting switching variability, such as the stochastic delay time in Fig. 4c. When a constant voltage close to V_{set} is applied to a RRAM in the HRS, set transition takes place after a delay time t_D, which varies from cycle to cycle due to statistical changes in the electrical and ionic conductive paths within the device [74]. A random number can be generated by dividing the time in equally spaced intervals Δt as in Fig. 4d, and attributing bit 0 or 1 to the switching event taking place in even or odd windows, respectively. This scheme improves bit randomness, as the probability of generating 0 or 1 is close to 50% provided that t_D is sufficiently larger than Δt [82]. Instead of measuring the delay time for switching, the state of the device can be conveniently measured after a fixed amount of time, as shown in Fig. 4e. Here, a voltage equal to the median value of the stochastic V_{set} is applied to a RRAM device in the HRS, thus statistically resulting in a set transition for 50% of the times. The resistance distribution of the final states after stochastic switching thus shows a bimodal distribution of HRS and LRS (Fig. 4f), which can be attributed to bit 0 and 1, respectively [83]. This technique was also applied to stochastic computing, where a single device can represent an analogue value corresponding, e.g., to the fraction of HRS in Fig. 4f [84]. For the generated random numbers to be uniform, the exact value of $<V_{set}>$ should be known, which requires a preliminary probability tracking procedure to initialize the RNG [85]. Similar voltage-based RNG schemes were developed adopting STT-MRAM devices, which benefit from a higher cycling lifetime and higher switching speed [85,86]. The need of a probability tracking can be overcome by differential TRNGs (Fig. 4e), where the competition within two switching devices randomly yields HRS-LRS or LRS-HRS pairs, which can be attributed to bit 0 and 1, respectively, with 50% probability [87].

Overall, in-memory TRNG provide physical random numbers with high randomness quality, as assessed by standard tests [82,85,87], and simple circuit layout, consisting of just few switching devices and some external control for stochastic programming and read. Due to its higher stability and better endurance, STT-MRAM devices appear the best device option to implement TRNGs. On the other hand, PUF circuits, which require only random initialization and an optional runtime reconfiguration, may be developed with PCM and RRAM circuits, thus benefitting from a lower cost and easier integration in the CMOS process flow.

end Box 1-->

Analog computing with crosspoint arrays

In-memory computing can adopt not only microscopic physical phenomena, but also universal circuit laws such as the Kirchhoff's and Ohm's law in resistive memory arrays. A typical example is the crosspoint array, consisting of multiple intersections between row and column orthogonal electrodes, each intersection containing a resistive memory element, such as a RRAM [88] or a PCM [89]. The crosspoint memories are extremely attractive to reduce the bit cell size, as the individual device area is just $4F^2$, where F is the lithographic feature size in the process technology. From the viewpoint of in-memory computing, the crosspoint array naturally provides a hardware accelerator for analogue, approximated matrix vector multiplication (MVM). Fig. 5a illustrates the concept of MVM in a crosspoint array, where a voltage V_j is applied to the *j*-th column, with $j = 1, 2, \ldots, N$, where N is the number of rows and columns. The voltage-induced currents of each resistive element are collected at the grounded rows, yielding a total current:

$$I_i = \Sigma_j G_{ij} V_j \qquad (1)$$

at the *i*-th row, where G_{ij} is the conductance of the resistive memory at row *i* and column *j*. Eq. (1) is the analogue product of the conductance matrix G_{ij} and the voltage vector V_j , which implements a hardware-based MVM via Ohm's and Kirchhoff's laws [90]. The analogue MVM in the crosspoint can be carried out in just one step, as opposed to the digital MAC operation, which is a time- and energy-consuming step in classical computers. Note that a significant amount of energy for crosspoint-based MVM is spent in operating analog-digital converters (ADCs) that transform the digital input vector into analog voltages V_j, in cases where the input of the calculations does not come directly from analog sensors, or where further digital processing of the output is needed [91]. A fair comparison of energy and area efficiency should therefore consider both direct (crosspoint) and indirect (periphery) contributions [92].

Crosspoint MVM can be adopted for a broad range of problems, including image compression [91], sparse coding [93], and implementation of artificial neural networks (ANNs), where G_{ij} has the meaning of a synaptic weight, V_j is a pre-synaptic spike amplitude, and I_i is the input signal to the *i*-th neuron [69,70]. For instance, Fig. 5a represents a 3x3 ANN with 3 input neurons and 3 output neurons, where synaptic weights can be trained directly in hardware by gradient-descent algorithm and backpropagation, taking advantage of pulse accumulation in PCM and RRAM for updating the weights. The MVM scheme can be used to implement a content addressable memory (CAM), that is an associative memory which provides the location of the memory where the digital content is the best match to an input set of digital data. Fig. 5b shows the CAM concept, namely a crosspoint array with stored digital data G_{ij}, and a set of input data V_j [52]. The row current in Eq. (1) provides an analogue match function, which is maximum for the input data being closest to the stored data G_{ij}.

While MVM relies on the precise voltage control of columns and rows for MVM operations, alternative biasing techniques can be used, *e.g.*, to generate stochastic challenge-response PUFs. Fig. 5c shows a crosspoint PUF, where random conductance values G_{ij} are stored, and all lines are left floating except for column j* and row i*, which are biased to V and 0, respectively. The current sensed at row i* includes not only the current of element (i*,j*), but also a number of sneak-path currents, flowing across the

indirectly biased resistive elements at floating rows/columns in the array [94]. Sneak-path currents are generally unwanted in memory arrays where the individual memory cell must be sensed to read the stored data. As a result of sneak-path currents in the crosspoint PUF, the output current is a complicated (hence hardly clonable) function of i*, j*, V, and G_{ij}. The same concept can be generalized by biasing an arbitrary number of columns, serving as the input challenge [94]. Thanks to the good scalability and stochastic variation of resistance, the crosspoint PUF appears a promising solution for hardware security in the IoT.

Outlook

In-memory computing provides a promising approach to overcome the limitations of existing von Neumann based computing approaches. However, there are many technical issues that must be addressed for in-memory computing to become a viable solution in information technology. It has been mentioned that switching variability is a major concern for deterministic computing (for example, the IMP logic gate of Fig. 2c can result in errors due to the cell-to-cell and cycle-to-cycle variations of V_{set} in RRAM). On the other hand, inherently stochastic functions, such as stochastic integration in artificial neurons and RNGs, benefit from switching variations in memory elements. It should be noted that statistical variability also affects memory operation for storage [74], although in such case it can be managed at the system level by algorithms that verify and correct the memory state soon after programming and correct for errors after a read operation. Similar verify techniques that are difficult to implement in computing and may affect the benefits of a pure in-memory computing system.

Besides variations, memory instability can also limit the accuracy of in-memory computing, particularly for analogue operations in crosspoint computing. For instance, even assuming a precise tuning of array elements G_{ij} by verify techniques in Fig. 5a, the conductance G_{ij}, might be affected by spontaneous fluctuations, thus causing MVM inaccuracies. Instability particularly affects RRAM devices, as the localization in the LRS and HRS makes the resistance extremely sensitive to individual atomic transitions close to the conductive path [80]. PCM devices are less affected by instability due to the bulk-type conduction mechanism present in these devices. However, the resistance can drift in time as a result of the metastable nature of the amorphous state [95]. Resistance drift originates from structural relaxation of the amorphous phase after quenching from the liquid phase and consists of a decrease of the defect concentration and an increase of band gap and resistivity. Drift can be alleviated by increasing the read current [96], and adopting a core-shell structure of the memory cell, where the conduction path flows away from the core amorphous phase via a metallic shell layer [97]. In general, drift and noise increase their effects at high memory resistance, which also suffers from a higher nonlinearity of conductance, due to field- and temperature-induced enhancement of transport [91]. On the other hand, programming states at low resistance requires relatively high currents, thus impacting the energy efficiency of the computing system. Higher operating currents also raise the parasitic voltage drop across the metallic lines constituting the rows and columns of the memory array. As a result of resistance variation and nonlinearity, crosspoint arrays only compute approximate results, which should be restricted to a limited set of error-tolerant tasks, *e.g.*, pattern recognition, page ranking, and data inference. Given this intimate device-system interaction, the design and optimization of the memory device should rely on a detailed consideration of the system-level performance metrics, including accuracy, energy efficiency, and switching speed. The memory device, as a result, will most likely be different from those targeted for digital data storage.

A key requirement for in-memory computing to become a mainstream technology is scaling. The growth of internet data has been driving the scaling of Flash memory density by 40% increase per year, to sustain the storage capacity in mobile computers and data centres [98]. To enable a similar growth rate for

in-memory computing, crosspoint memory arrays should scale down. The density increase can be obtained by a decrease of the individual cell size (for example, by reducing the diameter of the RRAM device in Fig. 6a). Reducing the size of the computing element, however, raises concerns about the control of switching parameters and increased cell-to-cell variability. With the device downscaling, the interconnect line and the periphery circuit area should correspondingly decrease. However, interconnect downscaling causes an increase of series resistance, due to both the geometry scaling and the enhanced surface scattering [99]. The increased line resistance complicates the operation of the crosspoint circuit due to parasitic voltage drops, especially at high operating currents. Methods to reduce the interconnect resistivity include the use of novel materials, such as carbon nanotube and graphene [100], and alternative scaling paths.

To overcome the difficulties of in-plane scaling, novel 3D array architectures have been proposed, such as the horizontal stacked 3D structure in Fig. 6b [89], and the vertical 3D structure in Fig. 6c [101]. The vertical 3D structure offers a better processing yield and cost efficiency with respect to horizontal 3D structures, as the critical lithography steps are limited to the creation of the pillar across the multiple electrode/spacing layers. In fact, in a vertical 3D array, memory cells are formed at the crossing between a horizontal plane and a vertical pillar, consisting of a core-shell structure with a metallic core and an insulating shell serving as the switching layer. Within a 3D array, high density can be achieved by increasing the number of stacked layers, instead of reducing the cell size and line width. The horizontal cell-cell pitch can be reduced by decreasing the thickness of the switching layer, which strongly favours RRAM and FeRAM memories thanks to the ultrathin switching layer, as opposed to relatively thick PCM and STT-MRAM elements. Vertical 3D arrays have been recently demonstrated for in-memory computing applications, where 3D RRAM devices were used as multiplication-addition-permutation (MAP) kernels to classify and associate data for an integrated hyper-dimensional computing system [102]. Addressing the multiple technology challenges of 3D co-integration of memory devices, CMOS periphery, and low-resistivity interconnect, can serve as a future highway for high density, energy efficient in-memory computing.

To assess the full potential of in-memory computing, one should consider the individual computing blocks, such as the logic gates for Boolean operations or the crosspoint array that are discussed earlier in this article, and also system-level aspects such as the periphery circuit, the area efficiency, and the time and energy efficiency of the system. For instance, operating the logic gates in Fig. 3 requires a control logic in the periphery, biasing the lines of selected data while optimizing the crosspoint area use for maximum parallelism, and minimizing the interaction between independent operations, and the disturb to unselected bits. Other important considerations include power and clock delivery, especially for circuits in which the signal lines also need to supply the power and for circuits that need multi-phase clocks or precisely timed clocks. Without a critical assessment of the control system and algorithms, a comparison with conventional von Neumann computers is not possible. Similarly, comparing digital MAC and crosspoint-based MVM requires a detailed evaluation of the system complexity, periphery circuit area, error tolerance, memory array utilization, and overall energy efficiency. In this scenario, choosing a suitable application plays a significant

role, where in-memory computing might be better suited to data intensive, error tolerant tasks. The development of improved devices, with higher endurance, lower cycle-to-cycle variation, lower energy consumption, and lower instability, might considerably advance in-memory computing concepts and accelerate its adoption in the information communication technology world.

In-memory computing can subvert the conventional architecture of the computer and eliminate the memory wall of today's computing systems. Various schemes have been proposed to compute within resistive switching devices by exploiting the device physics to perform digital, analogue and stochastic computation. Although highly promising, significant efforts are still needed to address the interdisciplinary challenges of device optimization, circuit design, and system management. The development of resistive switching devices for storage is likely to strongly accelerate in-memory computing as a feasible alternative technology in post-Moore microelectronic industry.

Acknowledgements

DI acknowledges funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 648635). HSPW is supported in part by DARPA, the National Science Foundation (E2CDA, Expeditions in Computing), in addition to member companies of: Stanford Non-Volatile Memory Technology Research Initiative (NMTRI) and Stanford SystemX Alliance.

Author contributions

D.I. and H.S.P.W. conceived the project, carried out the discussions and wrote the manuscript.

Competing interests

The authors declare no competing financial interests.

Additional information

Reprints and permissions information is available at www.nature.com/reprints.

Correspondence and requests for materials should be addressed to D.I. or H.S.P.W.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

[1] G. E. Moore, "Cramming More Components onto Integrated Circuits," Electronics, pp. 114–117 (1965).

[2] M. M. Waldrop, "The chips are down for Moore's law," Nature 530, 144–147, (2016). DOI:10.1038/530144a

[3] W. A. Wulf, and S. A. McKee, "Hitting the memory wall: implications of the obvious," ACM SIGARCH Computer Architecture 23, 20-24 (1995). Doi:10.1145/216585.216588

[4] M. Horowitz, "Computing's energy problem (and what we can do about it)," ISSCC Tech. Dig. 10-14 (2014). DOI: 10.1109/ISSCC.2014.6757323

[5] Y.-H. Chen, T. Krishna, J. S. Emer, and V. Sze, "Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks," J. Solid-State Circ. 52, 127-138 (2017). DOI: Identifier 10.1109/JSSC.2016.2616357

[6] N. P. Jouppi, et al., "In-Datacenter Performance Analysis of a Tensor Processing Unit," 44th International Symposium on Computer Architecture (ISCA), Toronto, Canada, June 26, 2017. DOI:10.1145/3079856.3080246

[7] J. T. Pawlowski, "Hybrid memory cube (HMC)," IEEE Hot Chips 23 Symposium (2011). DOI: 10.1109/HOTCHIPS.2011.7477494

[8] D.U. Lee, K. W. Kim, K. W. Kim, et al. "A 1.2 V 8Gb 8-channel 128GB/s high-bandwidth memory (HBM) stacked DRAM with effective microbump I/O test methods using 29 nm process and TSV," 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 432–433 (2014). DOI: 10.1109/ISSCC.2014.6757501

[9] H.-S. P. Wong and S. Salahuddin, "Memory leads the way to better computing," Nat. Nanotechnol. 10(3), pp. 191-194 (2015). DOI:10.1038/nnano.2015.29

[10] R. Waser and M. Aono, "Nanoionics-Based Resistive Switching Memories," Nat. Mater., 6, 833 (2007). doi:10.1038/nmat2023

[11] S. Raoux, W. Welnic and D. Ielmini, "Phase change materials and their application to non-volatile memories," Chem. Rev. 110, 240-267 (2010). DOI: 10.1021/cr900040x

[12] A. D. Kent and D. C. Worledge, "A new spin on magnetic memories," Nat. Nanotechnol. 10, 187-191 (2015). doi:10.1038/nnano.2015.24.

[13] T. Mikolajick, C. Dehm, W. Hartner, I. Kasko, M.J. Kastner, N. Nagel, M. Moert, and C. Mazure, "FeRAM technology for high density applications," Microelectronics Reliability 41 (7), 947-950 (2001). DOI: 10.1016/S0026-2714(01)00049-X

[14] https://www.intel.com/content/www/us/en/architecture-and-technology/intel-optane-technology.html

[15] M. Di Ventra and Y. V. Pershin, “The parallel approach,” Nature Physics 9, 200–202 (2013). DOIdoi:10.1038/nphys2566

[16] G. Indiveri and S.-C. Liu, “Memory and Information Processing in Neuromorphic Systems,” Proc. IEEE 103, 1379-1397 (2015). DOI: 10.1109/JPROC.2015.2444094

[17] A. Beck, J. G. Bednorz, Ch. Gerber, C. Rossel, and D. Widmer, “Reproducible switching effect in thin oxide films for memory applications,” Appl. Phys. Lett. 77, 139 (2000). Doi:10.1063/1.126902

[18] Q. Liu, J. Sun, H. Lv, S. Long, K. Yin, N. Wan, Y. Li, L. Sun, and M. Liu, “Real-Time Observation on Dynamic Growth/Dissolution of Conductive Filaments in Oxide-Electrolyte-Based ReRAM,” Adv. Mater. 24, 1844-1849 (2012). DOI: 10.1002/adma.201104104

[19] D. Ielmini, “Modeling the universal set/reset characteristics of bipolar RRAM by field- and temperature-driven filament growth,” IEEE Trans. Electron Devices 58(12), pp. 4309-4317 (2011). DOI: 10.1109/TED.2011.2167513

[20] J. J. Yang, D. B. Strukov, and D. R. Stewart, “Memristive devices for computing,” Nat. Nanotechnol. 8, 13–24 (2013). DOI: 10.1038/NNANO.2012.240

[21] K. M. Kim, D. S. Jeong and C. S. Hwang, “Nanofilamentary resistive switching in binary oxide system; a review on the present status and outlook,” Nanotechnology 22, 254002 (2011). doi:10.1088/0957-4484/22/25/254002

[22] A. Sawa, “Resistive switching in transition metal oxides,” Materials Today 11, 28-36 (2008). Doi: 10.1016/S1369-7021(08)70119-6

[23] N. Yamada, E. Ohno, K. Nishiuchi, and N. Akahira, “Rapid-phase transitions of GeTe-Sb_2Te_3 pseudobinary amorphous thin films for an optical disk memory,” J. Appl. Phys. 69, 2849 (1991). DOI: 10.1063/1.348620

[24] D. Ielmini and Y. Zhang, “Analytical model for subthreshold conduction and threshold switching in chalcogenide-based memory devices,” J. Appl. Phys. 102, 054517 (2007).

[25] M. Boniardi, A. Redaelli, C. Cupeta, F. Pellizzer, L. Crespi, G. D’Arrigo, A. L. Lacaita and G. Servalli, “Optimization Metrics for Phase Change Memory (PCM) Cell Architectures,” IEDM Tech. Dig. 681-684 (2014). DOI: 10.1109/IEDM.2014.7047131

[26] B. J. Choi, A. C. Torrezan, J. P. Strachan, P. G. Kotula, A. J. Lohn, M. J. Marinella, Z. Li, R. S. Williams, J. J. Yang, “High-Speed and Low-Energy Nitride Memristors,” Adv. Funct. Mater. 26, 5290-5296 (2016). doi: 10.1002/adfm.201600680

[27] D. Loke, T. H. Lee, W. J. Wang, L. P. Shi, R. Zhao, Y. C. Yeo, T. C. Chong, and S. R. Elliott, “Breaking the Speed Limits of Phase-Change Memory,” Science 336, 1566 (2012). DOI: 10.1126/science.1221561

[28] M.-J. Lee, C. B. Lee, D. Lee, S. R. Lee, M. Chang, J. H. Hur, Y.-B. Kim, C.-J. Kim, D. H. Seo, S. Seo, U-I. Chung, I.-K. Yoo and K. Kim, "A fast, high-endurance and scalable non-volatile memory device made from asymmetric Ta_2O_{5-x}/TaO_{2-x} bilayer structures," Nat. Mater. 10, 625–630 (2011). doi:10.1038/nmat3070

[29] C. Chappert, A. Fert and F. Nguyen Van Dau, "The emergence of spin electronics in data storage," Nature Materials volume 6, pages 813–823 (2007). doi:10.1038/nmat2024

[30] N. Locatelli, V. Cros and J. Grollier, "Spin-torque building blocks," Nature Materials 13, 11–20 (2014). doi:10.1038/nmat3823

[31] J. Slonczewski, "Current-driven excitation of magnetic multilayers," J. Magn. Magn. Mater. 159, L1–L7 (1996). Doi:10.1016/0304-8853(96)00062-5

[32] S. Yuasa, T. Nagahama, A. Fukushima, Y. Suzuki and K. Ando, "Giant room-temperature magnetoresistance in single-crystal Fe/MgO/Fe magnetic tunnel junctions," Nat. Mater. 3, 868–871 (2004). doi:10.1038/nmat1257

[33] R. Carboni, S. Ambrogio, W. Chen, M. Siddik, J. Harms, A. Lyle, W. Kula, G. Sandhu, and D. Ielmini, "Understanding cycling endurance in perpendicular spin-transfer torque (p-STT) magnetic memory," IEDM Tech. Dig. 572 (2016). DOI: 10.1109/IEDM.2016.7838468

[34] T. S. Boescke, J. Mueller, D. Brauhaus, U. Schroeder, and U. Boettger, "Ferroelectricity in hafnium oxide thin films," Appl. Phys. Lett. 99, 102903 (2011). doi:10.1063/1.3634052

[35] M. Trentzsch, S. Flachowsky, R. Richter, J. Paul, B. Reimer, D. Utess, S. Jansen, H. Mulaosmanovic, S. Müller, S. Slesazeck, J. Ocker, M. Noack, J. Müller, P. Polakowski, J. Schreiter, S. Beyer, T. Mikolajick, and B. Rice, "A 28nm HKMG super low power embedded NVM technology based on ferroelectric FETs," IEDM The. Dig. 294 (2016). DOI: 10.1109/IEDM.2016.7838397

[36] S. Oh, T. Kim, M. Kwak, J. Song, J. Woo, S. Jeon, I. K. Yoo, and H. Hwang, "$HfZrO_x$ -Based Ferroelectric Synapse Device With 32 Levels of Conductance States for Neuromorphic Applications," IEEE Electron Device Lett. 38, 732 (2017). DOI: 10.1109/LED.2017.2698083

[37] M. T. Niemier, G. H. Bernstein, G. Csaba, A. Dingler, X. S. Hu, S. Kurtz, S. Liu, J. Nahas, W. Porod, M. Siddiq and E. Varga, "Nanomagnet logic: progress toward system-level integration," J. Phys.: Condens. Matter 23, 493202 (2011). DOI:10.1088/0953-8984/23/49/493202

[38] I. Amlani, A. O. Orlov, G. Toth, G. H. Bernstein, C. S. Lent, G. L. Snider, "Digital Logic Gate Using Quantum-Dot Cellular Automata," Science 284, 289-291 (1999). DOI: 10.1126/science.284.5412.289

[39] A. A. Khajetoorians, J. Wiebe, B. Chilian, R. Wiesendanger, "Realizing All-Spin–Based Logic Operations Atom by Atom," Science 27, 332, 1062-1064 (2011). DOI: 10.1126/science.1201725

[40] B. Govoreanu, G. S. Kar, Y.-Y. Chen, V. Paraschiv, S. Kubicek, A. Fantini, I. P. Radu, L. Goux, S. Clima, R. Degraeve, N. Jossart, O. Richard, T. Vandeweyer, K. Seo, P. Hendrickx, G. Pourtois, H. Bender, L. Altimime, D. J. Wouters, J. A. Kittl, and M. Jurczak, "$10x10nm^2$ Hf/HfO_x Crossbar Resistive RAM with

Excellent Performance, Reliability and Low-Energy Operation," IEDM Tech. Dig. 729 (2011). DOI:10.1109/IEDM.2011.6131652

[41] E. Linn, R. Rosezin, S. Tappertzhofen, U. Böttger and R. Waser, "Beyond von Neumann—logic operations in passive crossbar arrays alongside memory operations," Nanotechnology 23, 305205 (2012). doi:10.1088/0957-4484/23/30/305205

[42] P.-E. Gaillardon, L. Amarù, A. Siemon, E. Linn, R. Waser, A. Chattopadhyay, G. De Micheli, "The Programmable Logic-in-Memory (PLiM) Computer," IEEE Design, Automation & Test in Europe Conference & Exhibition (DATE) 427-432 (2016).

[43] G. Papandroulidakis, I. Vourkas, N. Vasileiadis, and G. Ch. Sirakoulis, "Boolean Logic Operations and Computing Circuits Based on Memristors," IEEE Trans. Circuits Syst. II, Exp. Briefs 61, 972-976 (2014). DOI: 10.1109/TCSII.2014.2357351

[44] D. E. Nikonov, and I. A. Young, "Overview of Beyond-CMOS Devices and a Uniform Methodology for Their Benchmarking," Proc. IEEE 101, 2498-2533 (2013). DOI:10.1109/JPROC.2013.2252317

[45] L. Gao, F. Alibart, and D. B. Strukov, "Programmable CMOS/memristor threshold logic," IEEE Trans. Nanotechnology 12, 115-119 (2013). DOI: 10.1109/TNANO.2013.2241075

[46] A. P. James, L. R. V. J. Francis, and D. S. Kumar, "Resistive Threshold Logic," IEEE Trans. Very Large Scale Integr. (VLSI) 22, 190-195 (2014). DOI: 10.1109/TVLSI.2012.2232946

[47] J. Borghetti, G. S. Snider, P. J. Kuekes, J. J. Yang, D. R. Stewart, and S. S. Williams, "'Memristive' switches enable 'stateful' logic operations via material implication," Nature, 464, 873 (2010). doi:10.1038/nature08940

[48] J. Reuben, R. Ben-Hur, N. Wald, N. Talati, A. Haj Ali, P.-E. Gaillardon, S. Kvatinsky, "Memristive Logic: A Framework for Evaluation and Comparison," 27th International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS), 2017. DOI: 10.1109/PATMOS.2017.8106959

[49] D. S. Jeong, K. M. Kim, S. Kim, B. J. Choi, and C. S. Hwang, "Memristors for Energy-Efficient New Computing Paradigms," Adv. Electron. Mater. 2, 1600090 (2016). DOI: 10.1002/aelm.201600090

[50] S. Balatti, S. Ambrogio, and D. Ielmini, "Normally-off logic based on resistive switches – Part I: Logic gates," IEEE Trans. Electron Devices 62, 1831-1838 (2015). DOI: 10.1109/TED.2015.2422999

[51] P. Huang, J. Kang, Y. Zhao, S. Chen, R. Han, Z. Zhou, Z. Chen, W. Ma, M. Li, L. Liu, and X. Liu, "Reconfigurable nonvolatile logic operations in resistance switching crossbar array for large-scale circuits," Adv. Mater. 28(44), pp. 9758-9764 (2016). DOI: 10.1002/adma.201602418

[52] B. Chen, F. Cai, J. Zhou, W. Ma, P. Sheridan, and W. D. Lu, "Efficient in-memory computing architecture based on crossbar arrays," in IEDM Tech. Dig., pp. 17.5.1-17.5.4 (2015). DOI: 10.1109/IEDM.2015.7409720

[53] M. Cassinerio, N. Ciocchini and D. Ielmini, “Logic computation in phase change materials by threshold and memory switching,” Adv. Mater. 25, pp. 5975-5980 (2013). DOI: 10.1002/adma.201301940

[54] H. Mahmoudi, T. Windbacher, V. Sverdlov, S. Selberherr, “Implication Logic Gates Using Spin-Transfer-Torque-Operated Magnetic Tunnel Junctions for Intrinsic Logic-In-Memory,” Solid-State Electronics 84, 191-197 (2013). doi:10.1016/j.sse.2013.02.017.

[55] S. Balatti, S. Ambrogio, Z.-Q. Wang, S. Sills, A. Calderoni, N. Ramaswamy, and D. Ielmini, “Voltage-controlled cycling endurance of HfO_x-based resistive-switching memory (RRAM),” IEEE Trans. Electron Devices 62, 3365 (2015).

[56] L.T. Clark, V. Vashishtha, L. Shifren, A. Gujja, S. Sinha, B. Cline, C. Ramamurthya, and G. Yeric, “ASAP7: A 7-nm FinFET Predictive Process Design Kit,” Microelectronics Journal, vol. 53, pp. 105-115, July 2016. doi: 10.1016/j.mejo.2016.04.006

[57] C. D. Wright, P. Hosseini, J. A. Vazquez Diosdado, “Beyond von-Neumann Computing with Nanoscale Phase-Change Memory Devices,” Adv. Functional Mater. 23, 2248–2254 (2013). DOI: 10.1002/adfm.201202383

[58] J. Feldmann, M. Stegmaier, N. Gruhler, C. Ríos, H. Bhaskaran, C.D. Wright and W.H.P. Pernice, “Calculating with light using a chip-scale all-optical abacus,” Nature Communications 8:1256 (2017). DOI: 10.1038/s41467-017-01506-3

[59] P. Hosseini, A. Sebastian, N. Papandreou, C. D. Wright, and H. Bhaskaran, “Accumulation-based computing using phase-change memories with FET access devices,” IEEE Electron Device Lett. 36(9), 975-977 (2015). DOI: 10.1109/LED.2015.2457243

[60] O. Bichler, M. Suri, D. Querlioz, D. Vuillaume, B. De Salvo, and C. Gamrat, “Visual pattern extraction using energy-efficient 2-PCM synapse neuromorphic architecture,” IEEE Trans. Electron Devices 59, 2206 (2012).

[61] G. W. Burr, R. M. Shelby, S. Sidler, C. di Nolfo, J. Jang, I. Boybat, R. S. Shenoy, P. Narayanan, K. Virwani, E. U. Giacometti, B. N. Kurdi, and H. Hwang, “Experimental demonstration and tolerancing of a large-scale neural network (165 000 synapses) using phase-change memory as the synaptic weight element,” IEEE Trans. Electron Devices 62, pp. 3498-3507 (2015). DOI: 10.1109/TED.2015.2439635

[62] T. Tuma, A. Pantazi, M. Le Gallo, A. Sebastian, and E. Eleftheriou, “Stochastic phase-change neurons,” Nat. Nanotechnol. 11, pp. 693-699 (2016). DOI: 10.1038/NNANO.2016.70

[63] N. Qiao and G. Indiveri, “Scaling mixed-signal neuromorphic processors to 28 nm FD-SOI technologies,” IEEE Biomedical Circuits & Systems Conference (BioCAS), 552-555 (2016). DOI: 10.1109/BioCAS.2016.7833854

[64] P. Stoliar, J. Tranchant, B. Corraze, E. Janod, M.-P. Besland, F. Tesler, M. Rozenberg, L. Cario, "A Leaky-Integrate-and-Fire Neuron Analog Realized with a Mott Insulator," Adv. Funct. Mater. 27, 1604740 (2017). DOI: 10.1002/adfm.201604740

[65] Z. Wang, S. Joshi, S. Savel'ev, W. Song, R. Midya, Y. Li, M. Rao, P. Yan, S. Asapu, Y. Zhuo, H. Jiang, P. Lin, C. Li, J. H. Yoon, N. K. Upadhyay, J. Zhang, M. Hu, J. P. Strachan, M. Barnell, Q. Wu, H. Wu, R. S. Williams, Q. Xia, and J. J. Yang, "Fully memristive neural networks for pattern classification with unsupervised learning," Nature Electronics 1, 137–145 (2018). doi:10.1038/s41928-018-0023-2

[66] S. Larentis, F. Nardi, S. Balatti, D. C. Gilmer and D. Ielmini, "Resistive switching by voltage-driven ion migration in bipolar RRAM – Part II: Modeling," IEEE Trans. Electron Devices 59(9), pp. 2468–2475 (2012). DOI: 10.1109/TED.2012.2202320

[67] S. Yu, Y. Wu, R. Jeyasingh, D. Kuzum, and H.-S. P. Wong, "An electronic synapse device based on metal oxide resistive switching memory for neuromorphic computation," IEEE Trans. Electron Devices 58(8), 2729-2737 (2011). DOI: 10.1109/TED.2011.2147791

[68] Yu, S. et al. A low energy oxide-based electronic synaptic device for neuromorphic visual systems with tolerance to device variation. Adv. Mater. 25, 1774–1779 (2013). Doi: 10.1002/adma.201203680

[69] M. Prezioso, F. Merrikh-Bayat, B. D. Hoskins, G. C. Adam, K. K. Likharev, and D. B. Strukov, "Training and operation of an integrated neuromorphic network based on metal-oxide memristors." Nature 521, 61–64 (2015). doi:10.1038/nature14441

[70] J.-W. Jang, S. Park, G. W. Burr, H. Hwang, and Y.-H. Jeong "Optimization of conductance change in $Pr_{1-x}Ca_xMnO_3$-based synaptic devices for neuromorphic systems," IEEE Electron Device Lett. 36(5), 457-459 (2015). DOI: 10.1109/LED.2015.2418342

[72] A. Chanthbouala, V. Garcia, R. O. Cherifi, K. Bouzehouane, S. Fusil, X. Moya, S. Xavier, H. Yamada, C. Deranlot, N. D. Mathur, M. Bibes, A. Barthélémy, J. Grollier "A ferroelectric memristor," Nature Materials 11 (10), 860-864 (2012). doi:10.1038/nmat3415

[72] S. Lequeux, J. Sampaio, V. Cros, K. Yakushiji, A. Fukushima, R. Matsumoto, H. Kubota, S. Yuasa and J. Grollier, "A magnetic synapse: multilevel spin-torque memristor with perpendicular anisotropy," Sci. Rep. 6:31510 (2016). doi:10.1038/srep31510

[73] C. Diorio, P. Hasler, B. A. Minch, and C. Mead, "A single-transistor silicon synapse," IEEE Transactions on Electron Devices 43, 1972–1980 (1996). DOI: 10.1109/16.543035

[74] S. Ambrogio, S. Balatti, A. Cubeta, A. Calderoni, N. Ramaswamy, and D. Ielmini, "Statistical fluctuations in HfOx resistive-switching memory (RRAM): Part I – Set/Reset variability," IEEE Trans. Electron Devices 61, 2912-2919 (2014). DOI: 10.1109/TED.2014.2330200

[75] M. Rizzi, N. Ciocchini, A. Montefiori, M. Ferro, P. Fantini, A. L. Lacaita, and D. Ielmini, "Cell-to-Cell and Cycle-to-Cycle Retention Statistics in Phase-Change Memory Arrays," IEEE Trans. Electron Devices 62, 2205–2211 (2015). DOI: 10.1109/TED.2015.2434278

[76] A. Chen, "Utilizing the Variability of Resistive Random Access Memory to Implement Reconfigurable Physical Unclonable Functions," IEEE Electron Device Lett 36, 138-140 (2015). DOI: 10.1109/LED.2014.2385870

[77] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, "Physical Unclonable Functions and Applications: A Tutorial," Proc. IEEE 102, 1126 – 1141 (2014). DOI: 10.1109/JPROC.2014.2320516

[78] W. Maass, "Noise as a Resource for Computation and Learning in Networks of Spiking Neurons," Proc. IEEE 102, 860 - 880 (2014). DOI:10.1109/JPROC.2014.2310593

[79] B. Jun and P. Kocher, "The Intel RNG," White Paper, 1999 [Online]. Available: https://www.rambus.com/intel-random-number-generator/

[80] S. Ambrogio, S. Balatti, A. Cubeta, A. Calderoni, N. Ramaswamy, and D. Ielmini, "Statistical fluctuations in HfO_x resistive-switching memory (RRAM): Part II – Random telegraph noise," IEEE Trans. Electron Devices 61, 2920-2927 (2014). DOI: 10.1109/TED.2014.2330202

[81] C.-Y. Huang, W. C. Shen, Y.-H. Tseng, Y.-C. King, and C.-J. Lin, "A Contact-Resistive Random-Access-Memory-Based True Random Number Generator," IEEE Electron Device Lett. 33, 1108-1110 (2012). DOI: 10.1109/LED.2012.2199734

[82] H. Jiang, D. Belkin, S. E. Savel'ev, S. Lin, Z. Wang, Y. Li, S. Joshi, R. Midya, C. Li, M. Rao, M. Barnell, Q. Wu, J. J. Yang and Q. Xia, "A novel true random number generator based on a stochastic diffusive memristor," Nature Communications 8: 882 (2017). doi:10.1038/s41467-017-00869-x

[83] S. Balatti, S. Ambrogio, Z.-Q. Wang, and D. Ielmini, "True random number generation by variability of resistive switching in oxide-based devices," IEEE J. Emerging Topics in Circuits and Systems (JETCAS) 5, 214-221 (2015). DOI: 10.1109/JETCAS.2015.2426492

[84] S. Gaba, P. Sheridan, J. Zhou, S. Choi and W. Lu, "Stochastic memristive devices for computing and neuromorphic applications," Nanoscale 5, 5872 (2013). DOI: 10.1039/C3NR01176C

[85] W. H. Choi, Y. Lv, J. Kim, A. Deshpande, G. Kang, J.-P. Wang, and C. H. Kim, "A Magnetic Tunnel Junction Based True Random Number Generator with Conditional Perturb and Real-Time Output Probability Tracking," IEDM Tech. Dig. 12.5.1-12.5.4 (2014). DOI: 10.1109/IEDM.2014.7047039

[86] A. Fukushima, T. Seki, K. Yakushiji, H. Kubota, H. Imamura, S. Yuasa and K. Ando, "Spin dice: A scalable truly random number generator based on spintronics," Appl. Phys. Express 7, 083001 (2014). DOI: 10.7567/APEX.7.083001

[87] S. Balatti, S. Ambrogio, R. Carboni, V. Milo, Z.-Q. Wang, A. Calderoni, N. Ramaswamy, and D. Ielmini, "Physical unbiased generation of random numbers with coupled resistive switching devices," IEEE Trans. Electron Devices 63, 2029-2035 (2016). DOI: 10.1109/TED.2016.2537792

[88] S. H. Jo, K.-H. Kim and W. Lu, "High-Density Crossbar Arrays Based on a Si Memristive System," Nano Lett. 9, 870–874 (2009). DOI: 10.1021/nl8037689

[89] D. Kau, S. Tang, I. V. Karpov, R. Dodge, B. Klehn, J. A. Kalb, J. Strand, A. Diaz, N. Leung, J. Wu, S. Lee, T. Langtry, K.-W. Chang, C. Papagianni, J. Lee, J. Hirst, S. Erra, E. Flores, N. Righos, H. Castro and G. Spadini, "A stackable cross point Phase Change Memory," IEDM Tech. Dig. 617 (2009). DOI:10.1109/IEDM.2009.5424263

[90] S. N. Truong and K.-S. Min, "New Memristor-Based Crossbar Array Architecture with 50-% Area Reduction and 48-% Power Saving for Matrix-Vector Multiplication of Analog Neuromorphic Computing," J. Semicond. Technol. Sci. 14, 356-363 (2014). DOI: 10.5573/JSTS.2014.14.3.356

[91] 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 and Q. Xia, "Analogue signal and image processing with large memristor crossbars," Nat. Electronics 1, 52–59 (2018). doi:10.1038/s41928-017-0002-z

[92] B. Eryilmaz, "Brain-inspired and non-conventional computing with emerging memory devices", PhD Thesis, Stanford University. https://searchworks.stanford.edu/view/12137356

[93] P. M. Sheridan, F. Cai, C. Du, W. Ma, Z. Zhang, and W. D. Lu, "Sparse coding with memristor networks," Nat. Nanotechnol. 12, pp. 784-789 (2017). DOI: 10.1038/nnano.2017.83

[94] L. Gao, P.-Y. Chen, R. Liu and S. Yu, "Physical Unclonable Function Exploiting Sneak Paths in Resistive Cross-point Array," IEEE Trans. Electron Devices 63, 3109-3115 (2016). DOI: 10.1109/TED.2016.2578720

[95] D. Ielmini, A. L. Lacaita and D. Mantegazza, "Recovery and drift dynamics of resistance and threshold voltages in phase change memories," IEEE Trans. Electron Devices 54, 308-315 (2007). DOI: 10.1109/TED.2006.888752

[96] D. Ielmini, D. Sharma, S. Lavizzari and A. L. Lacaita, "Reliability impact of chalcogenide-structure relaxation in phase change memory (PCM) cells – Part I: Experimental study," IEEE Trans. Electron Devices 56, 1070-1077 (2009).

[97] S. Kim, N. Sosa, M. BrightSky, D. Mori, W. Kim, Y. Zhu, K. Suu, and C. Lam, "A phase change memory cell with metallic surfactant layer as a resistance drift stabilizer," IEDM Tech. Dig. 762-765 (2013), DOI: 10.1109/IEDM.2013.6724727

[98] D. C. Daly, L. C. Fujino, and K. C. Smith “Through the Looking Glass - The 2017 Edition: Trends in Solid-State Circuits from ISSCC,” IEEE Solid-State Circuits Magazine 9, 12–22 (2017). DOI: 10.1109/MSSC.2016.262296

[99] P. Kapur, J. P. McVittie, and K. C. Saraswat, “Technology and Reliability Constrained Future Copper Interconnects—Part I: Resistance Modeling,” IEEE Trans. Electron Devices 49, 590-597 (2002). DOI: 10.1109/16.992867

[100] A. K. Geim and K. S. Novoselov, “The rise of graphene,” Nature Materials 6, 183–191 (2007). doi:10.1038/nmat1849

[101] S. Yu, H.-Y. Chen, B. Gao, J. Kang, and H.-S. P. Wong, “HfO_x Based Vertical Resistive Switching Random Access Memory Suitable for Bit-Cost-Effective Three-Dimensional Cross-Point Architecture,” ACS Nano 7, 2320 (2013). DOI: 10.1021/nn305510u

[102] H. Li, T. F. Wu, S. Mitra, and H.-S. P. Wong, “Resistive RAM-Centric Computing: Design and Modeling Methodology,” IEEE Trans. Circuits and Systems I: Regular Papers, Vol. 64, No. 9, pp. 2263 – 2273 (2017). DOI: 10.1109/TCSI.2017.2709812

Figure Captions

Fig. 1 | Computational memory devices. (a,b) Resistive switching random access memory (RRAM) structure and current-voltage (I-V) characteristic of a bipolar switching device. The set transition from the high resistance state (HRS) to the low resistance state (LRS) occurs at positive voltage due to the formation of a filament shunting the top and bottom electrodes, while the reset transition from LRS to HRS under negative voltage indicates the voltage-induced filament disconnection.(c,d) Phase change memory (PCM) structure and resistance change characteristic, showing the resistance measured after a voltage pulse is applied to a PCM device in the amorphous state. The decrease of resistance indicates increasing crystalized volume in the active material, while the increase of resistance above the melting point indicates increasing amorphous volume. (e,f) Magnetic tunnel junction (MTJ) structure and resistance-voltage (R-V) characteristic of a spin transfer torque magnetic random access memory (STT-MRAM) device. The parallel (P) and antiparallel (AP) states have low and high resistance, respectively, which can be attained at positive and negative voltage, respectively. (g,h) FeRAM structure and polarization-voltage hysteretic characteristic. The orientation of electrical dipoles causes permanent positive polarization at positive voltage, and negative polarization at negative voltage.

Fig. 2 | RRAM-based digital logic gates. (a,b) V-R logic gate and corresponding truth table for material implication (IMP). The V-R logic gate consists of a single resistive switch, where the input/output signals are the applied voltages at the 2 ends of the device, and the switch conductance state, respectively. (c,d) V-V logic, also known as the threshold logic gate, and the input/output characteristic. Input and output signals are the applied voltages at the input nodes, and the output of the comparator stage. The four configurations of input values can be linearly separated according to the weights G_j and the comparator threshold V_T, thus yielding a reconfigurable Boolean function. The input/output characteristic indicates an AND function, where low and high values of Y are indicated as filled and open symbols, respectively. (e,f) Parallel R-R stateful logic and corresponding truth table. Unconditional set transition occurs for $X_1 = 0$, while no switching takes place for $X_1 = 1$, thus resulting in an IMP operation. (g,h) Serial R-R stateful logic for OR operation and corresponding truth table. Conditional set transition from 0 to 1 takes place for odd input states, thus resulting in an OR operation. R-R logic is the only true in-memory option, as it is fully resident in the memory circuit.

Fig. 3 | Analogue computing in a PCM device. (a) Numerical simulations of the temperature profile during programming and the phase distribution within a mushroom-type phase change memory (PCM) device after an increasing number of 50 ns pulses. More applied pulses lead to an increasing crystalline phase, causing a decrease of threshold voltage and resistance. (b) Arithmetic summation of addends 4 and 3 by pulse accumulation in a PCM. (c) Integrate-and-fire neuron, where integration is carried

out by accumulating incoming spikes in a PCM element. (d) Synaptic potentiation by cumulative crystallization in a PCM synapse.

Fig. 4 | Stochastic computing with resistive switching devices. (a,b) Random telegraph noise (RTN) current fluctuations and corresponding probabilistic distribution function (PDF), attributing random bit values 0 and 1 to current sub-distributions I_0 and I_1, respectively. (c,d) Applied voltage pulse, its current response evidencing the random delay time t_D, and PDF of t_D with equally spaced time window to uniformly attribute bit values 0 and 1. (e,f) Measured I-V curves evidencing cycle-to-cycle variation of V_{set}, and PDF of the resistance measured after stochastic set, where sub-distributions of the high resistance state (HRS) and the low resistance state (LRS) are attributed to bits 0 and 1, respectively. (g) Differential pair for generating uniform sequences of random bits without probability tracking.

Fig. 5 | Analogue computing in crosspoint arrays. (a) Matrix-vector multiplication (MVM) within an artificial neural network (ANN), where input voltages V_j serve as pre-synaptic (input) neuron signals, and the array conductance G_{ij} describes the synaptic weight. The output row current I_i provides the sum of weighted currents feeding the post-synaptic neuron. (b) Content-addressable memory (CAM) concept adopting MVM of input data V_j and stored data G_{ij}. The MVM provides the best match to data, where the maximum response yields the address of input data. (c) Crosspoint physical unclonable function (PUF) for generating a response I* to a challenge, namely, the configuration of biased columns. The PUF relies on multiple sneak paths to yield a random unclonable function.

Fig. 6 | Crosspoint memory architecture and scaling. (a) Resistance switching random access memory (RRAM) crosspoint structure with a single memory layer. To increase the device density, the RRAM diameter should be reduced. (b) Horizontal stacked 3D array, where device stacking enables density multiplication, roughly by the number of layers. (c) Vertical 3D array, combining high density and cost-effective processing technology.

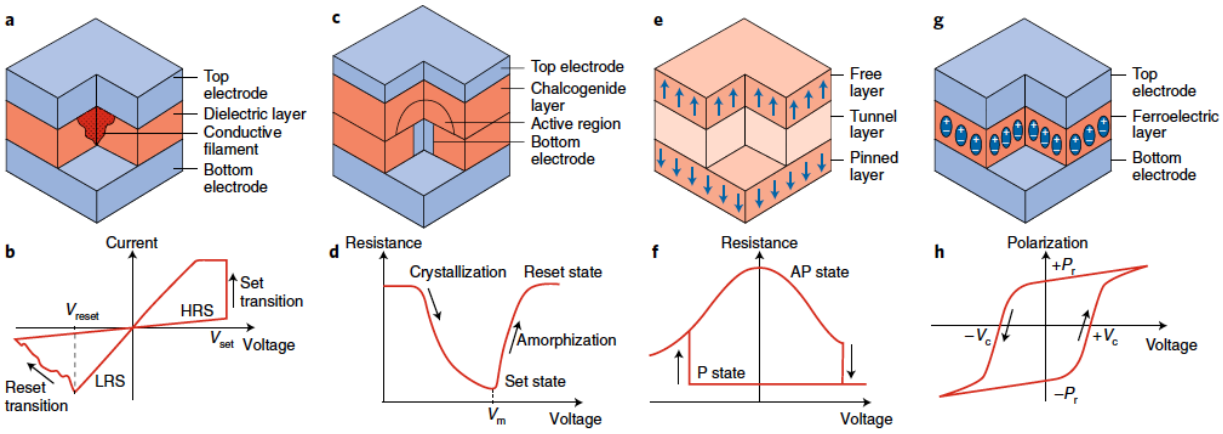
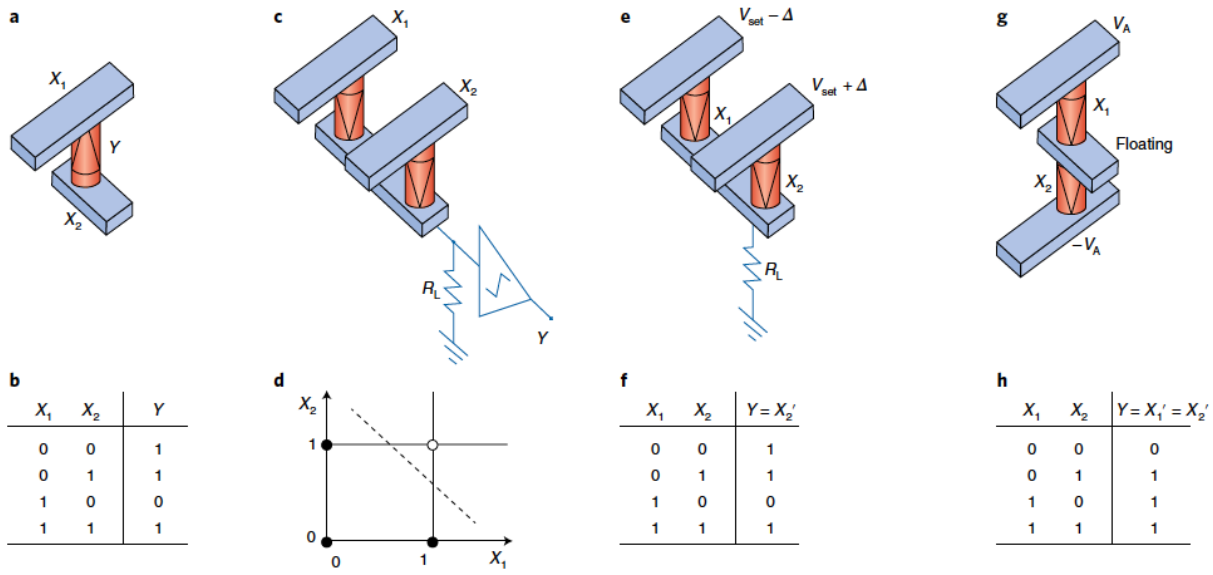

a
Top electrode
Dielectric layer
Conductive filament
Bottom electrode
b
Current
Set transition
HRS
V_{reset}
V_{set}
Voltage
Reset transition
LRS
c
Top electrode
Chalcogenide layer
Active region
Bottom electrode
d
Resistance
Crystallization
Reset state
Amorphization
Set state
V_m
Voltage
e
Free layer
Tunnel layer
Pinned layer
f
Resistance
AP state
P state
Voltage
g
Top electrode
Ferroelectric layer
Bottom electrode
h
Polarization
$+P_r$
$-V_c$
$+V_c$
Voltage
$-P_r$

Fig. 1



b

X_1	X_2	Y
0	0	1
0	1	1
1	0	0
1	1	1

d

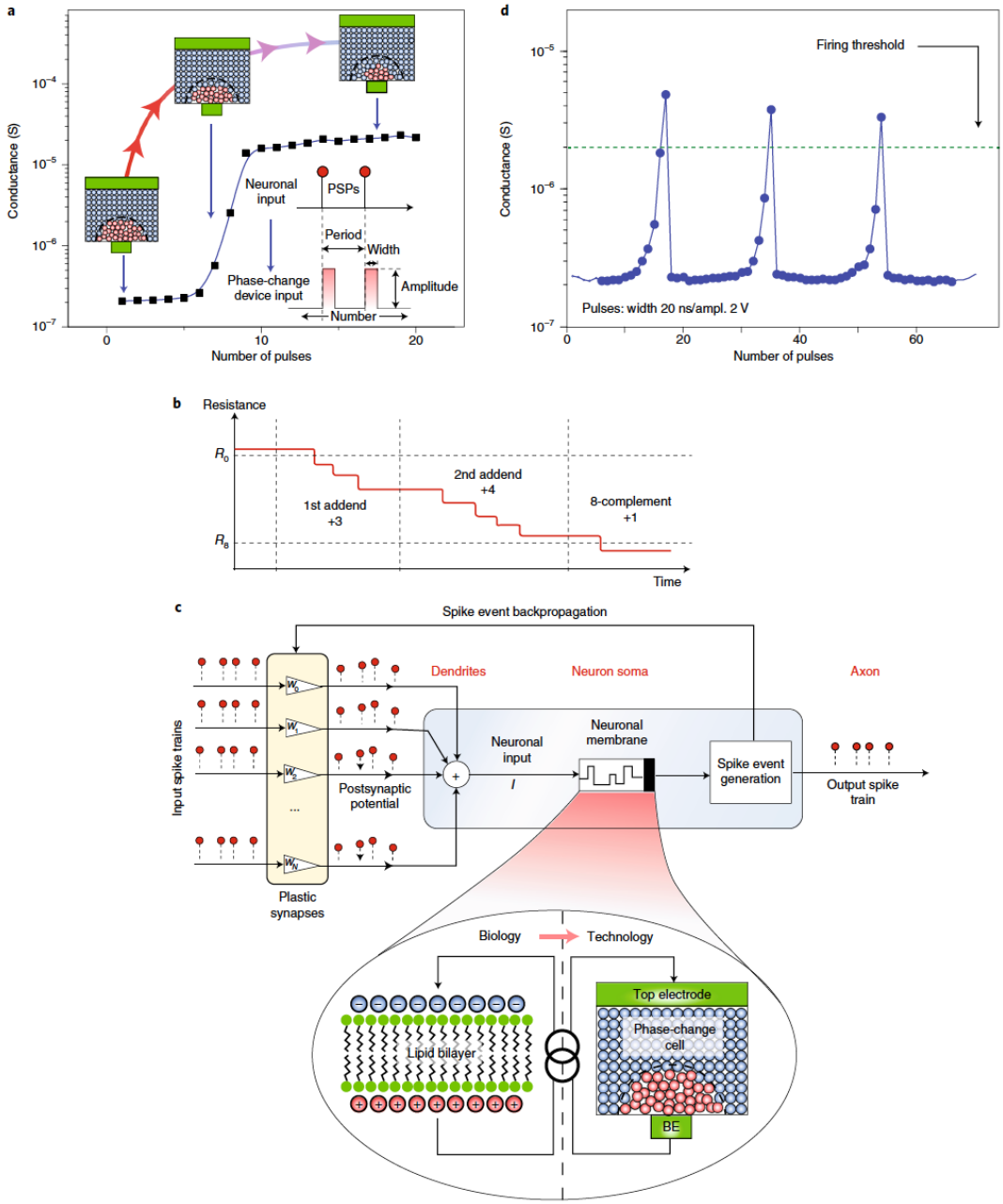
f

X_1	X_2	$Y = X_2'$
0	0	1
0	1	1
1	0	0
1	1	1

h

X_1	X_2	$Y = X_1' = X_2'$
0	0	0
0	1	1
1	0	1
1	1	1

Fig. 2


a
Conductance (S)
Neuronal input
PSPs
Period
Width
Phase-change device input
Amplitude
Number
Number of pulses
d
Firing threshold
Pulses: width 20 ns/ampl. 2 V
Conductance (S)
Number of pulses
b
Resistance
1st addend +3
2nd addend +4
8-complement +1
Time
c
Spike event backpropagation
Dendrites
Neuron soma
Axon
Input spike trains
Postsynaptic potential
Neuronal input
Neuronal membrane
Spike event generation
Output spike train
Plastic synapses
Biology
Technology
Lipid bilayer
Top electrode
Phase-change cell
BE

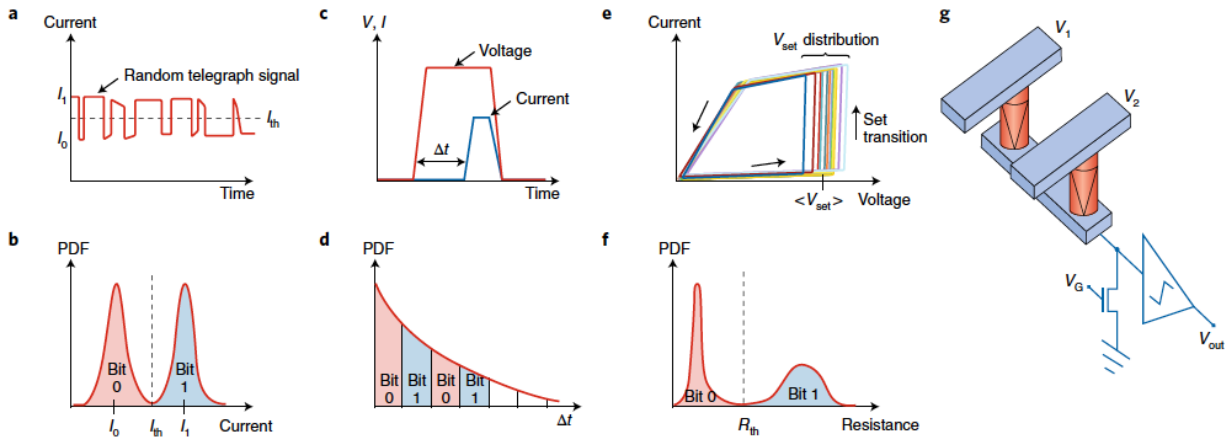

a
Current
Random telegraph signal
I_1
I_0
I_{th}
Time
b
PDF
Bit 0
Bit 1
I_0
I_{th}
I_1
Current
c
V, I
Voltage
Current
Δt
Time
d
PDF
Bit 0
Bit 1
Bit 0
Bit 1
Δt
e
Current
V_{set} distribution
Set transition
$<V_{set}>$
Voltage
f
PDF
Bit 0
Bit 1
R_{th}
Resistance
g
V_1
V_2
V_G
V_{out}

Fig. 4

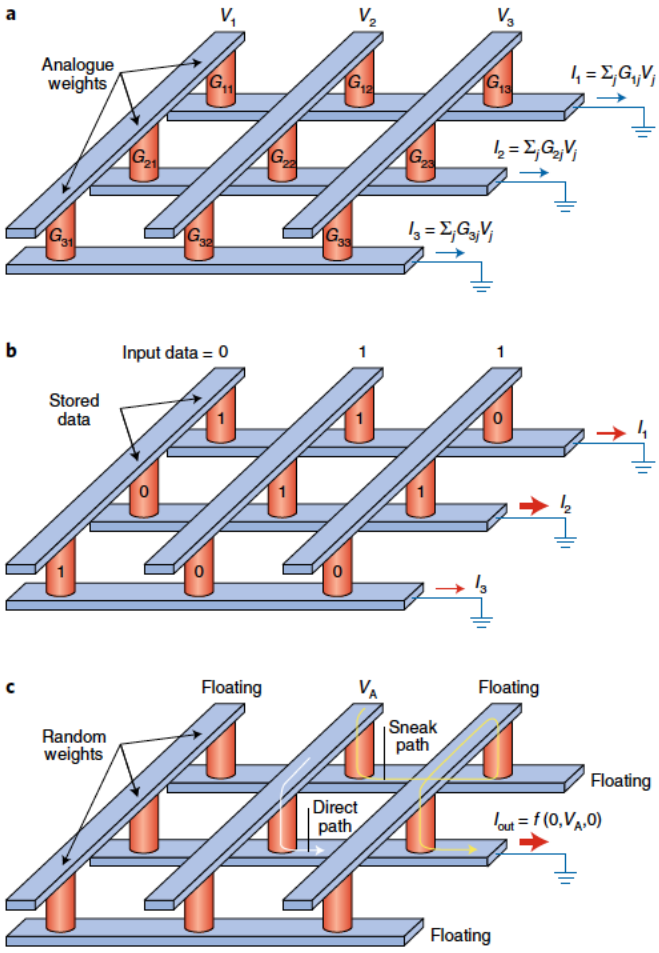

a
V_1
V_2
V_3
Analogue weights
G_{11}
G_{12}
G_{13}
G_{21}
G_{22}
G_{23}
G_{31}
G_{32}
G_{33}
$I_1 = \Sigma_j G_{1j} V_j$
$I_2 = \Sigma_j G_{2j} V_j$
$I_3 = \Sigma_j G_{3j} V_j$
b
Input data = 0
1
1
Stored data
1
1
0
0
1
1
1
0
0
I_1
I_2
I_3
c
Floating
V_A
Floating
Random weights
Sneak path
Floating
Direct path
$I_{out} = f(0, V_A, 0)$
Floating

Fig. 5

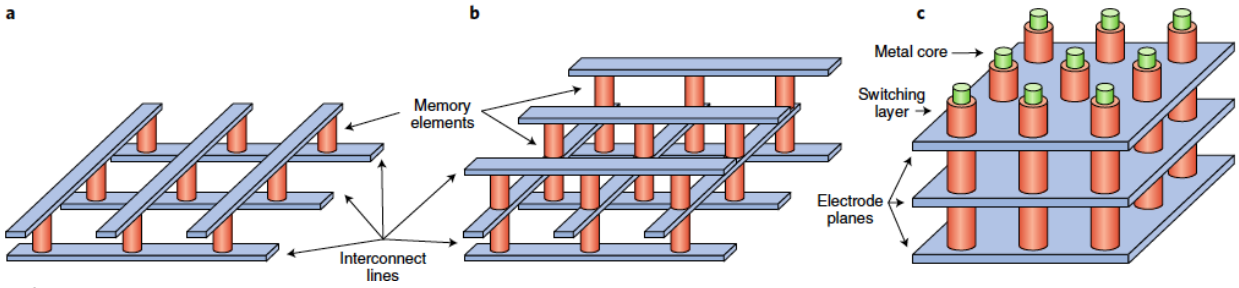

a
b
c
Memory elements
Interconnect lines
Metal core
Switching layer
Electrode planes

Fig. 6