A Survey of Memristive Threshold Logic Circuits

Akshay Kumar Maan, Deepthi Anirudhan Jayadevi, *Member, IEEE*,
and Alex Pappachen James, *Senior Member, IEEE*

***Abstract*— In this paper, we review different memristive threshold logic (MTL) circuits that are inspired from the synaptic action of the flow of neurotransmitters in the biological brain. The brainlike generalization ability and the area minimization of these threshold logic circuits aim toward crossing Moore's law boundaries at device, circuits, and systems levels. Fast switching memory, signal processing, control systems, programmable logic, image processing, reconfigurable computing, and pattern recognition are identified as some of the potential applications of MTL systems. The physical realization of nanoscale devices with memristive behavior from materials, such as TiO_2, ferroelectrics, silicon, and polymers, has accelerated research effort in these application areas, inspiring the scientific community to pursue the design of high-speed, low-cost, low-power, and high-density neuromorphic architectures.**

***Index Terms*— Memristors, neural circuits, neurocomputing, threshold logic (TL).**

I. Introduction

SEARCH for scalable hardware architectures for emulating neuron circuits in the biological brain is one of the major research areas to realizing silicon-based artificial-intelligence devices. The implementation of such circuits in standard devices, such as CMOS, has limitations of design and scalability. Memristive devices provide an interesting alternative not only by the way of their high packing density but also by the way they can enable a profoundly different approach to the large-scale computing inspired by the principle of firing of neurons in the biological brain. In biological brains, a neuron fires only if the total weight of the synapses that receive impulses in a short period (called the period of latent summation) exceeds a threshold. A threshold logic (TL) implemented in the memristive devices thus offers: 1) a synaptic action, in which the weight (memristance of the device) can be incrementally modified by controlling the charge or flux through it and 2) a thresholding system that governs the firing of the output.

The memristor, for example, is one such device [1]–[6] in the more-than-Moore (MtM) era of device integration [7]–[11], which has a high packing density with features such as low operational voltage, nonvolatility, and high switching speed. The memristor (a portmanteau of MEMory ResISTOR) is currently being described as the nanoscale device capable of emulating synaptic behavior in the brain [12]–[14], because it can remember the charge flown through it by changing its resistance. A wide range of device models and applications have sprung up in the recent years since Hewlett-Packard (HP) laboratories in 2008 [15], [16] described the behavior of physical TiO_2–TiO_{2-x} devices by a memristor model.

The TL itself was first introduced by McCulloch and Pitts [17] in 1943 in the early model of an artificial neuron based on the basic property of firing of the biological neuron. The logic computed the sign of the weighted sum of its inputs. Modeling a neuron as a TL gate (TLG) that fires when the input reaches a threshold has been the basis of the research on neural networks (NNs) and their hardware implementation in the standard CMOS logic [18]. However, to continue the geometrical scaling of semiconductor components as per Moore's law (and beyond), semiconductor industry needs to investigate a future beyond CMOS and design beyond the fetch–decode–execute paradigm of the von Neumann architecture. The memristive devices lend themselves nicely to this neuromorphic computing paradigm, making use of very-large-scale integration (VLSI) systems to mimic neurobiological architectures present in the nervous system that need no initializing software, run on negligibly low power, and are able to perform many cognitive tasks rapidly, effortlessly, and in real time. The memristive TL (MTL) is thus a step in the direction of building neuromorphic architectures while overcoming the design and scalability issues presented by CMOS NNs.

In Section II, we give a brief background of memristive systems and the TL. Section III provides a review of various implementations of the MTL, detailing the different architectures employed to achieve the thresholding behavior with memristive circuits and their applications, followed by discussion and open problems in Section IV. Section V concludes this paper.

II. Background

A. Memristive Devices and Systems

Memristors are two-terminal circuit elements, exhibiting passivity property characterized by a relationship between the charge $q(t)$ and the flux-linkage $\phi(t)$. The memristance (M) of the memristor has been defined by the relation $M = d\phi_m/dq$, and is expressed in Wb/C or Ω. The characteristic behavior

Manuscript received February 12, 2015; accepted March 18, 2016. Date of publication May 3, 2016; date of current version July 18, 2017. *(Corresponding author: Alex Pappachen James.)*

A. K. Maan is with Griffith University, Brisbane, QLD 4111, Australia (e-mail: akshay.maan@gmail.com).

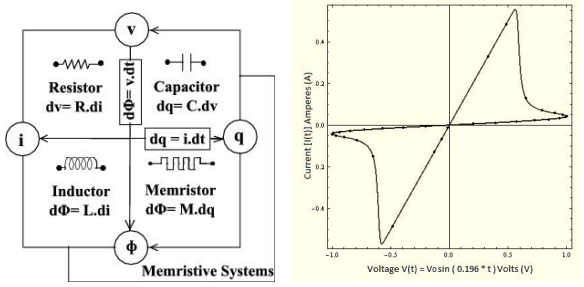
D. A. Jayadevi is with Enview Research and Development Laboratories, Thiruvananthapuram 695024, India (e-mail: deepthi_aj2002@yahoo.com).

A. P. James is with the Department of Electrical and Electronic Engineering, Nazarbayev University, Astana 010000, Kazakhstan (e-mail: apj@ieee.org).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org (File size: 1 MB).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2016.2547842



(a) (b)

Fig. 1. (a) Four fundamental two-terminal circuit elements: resistor (R), capacitor (C), inductor (I), and memristor (M). R, C, L, and M can be the functions of the independent variable in their defining equations, yielding nonlinear elements. For example, a charge-controlled memristor is defined by a single-valued function $M(q)$. (b) Voltage–current behavior in a TiO_2–TiO_{2-x} memristor model [16].

of a memristor was proposed in [19]. Chua [19] developed a circuit model built with at least 15 transistors and other passive elements [16] to emulate the behavior of a single memristor. The memristor model satisfied the passivity criterion and was characterized by a monotonically increasing ϕ–q curve [20], [21]. Fig. 1(a) shows the relationship between the following four fundamental circuit elements [resistor (R), capacitor (C), inductor (L), and memristor (M)] and the typical voltage–current behavior of a TiO_2–TiO_{2-x} memristive device. In 1976, Chua and Kang [22] defined any nonlinear dynamical systems with memristors as memristive systems. The memristive system was found to possess memory and behaved like the resistive devices endowed with a variety of dynamic characteristics. The memristive systems were incapable of energy discharge. Yet, they were found to exhibit small-signal inductive or capacitive effects without introducing a phase shift between the input and output waveforms.

A physical solid-state device using nanoscale titanium-dioxide films exhibiting the memristive properties was invented in 2008 by a team from HP laboratories [15]. The memristive device was found to be equipped with an ability to function like synapses in a biological brain [12]. The research team proposed a crossbar architecture, which was a fully connected mesh of perpendicular platinum wires with memristive switches made of TiO_2–TiO_{2-x} to connect any two crossing wires. These nanoscale switches were found to exhibit Lissajoux voltage–current behavior as shown in Fig. 1(b). Until then, similar dynamics were only common in relatively larger devices [23], [24]. The memristive dynamics mapped by the pinched-hysteresis loop [25] explains the switching behavior [26] of the device beginning with a high resistance. As the applied voltage increases, the charge flow inside the device slowly increases at first owing to the drop in the resistance value. This behavior is followed by a rapid increase in the device current up to the maximum increase in the applied voltage. When the voltage was decreased, the current decreased more slowly resulting in an ON-switching loop. The OFF-switching loop was observed when the voltage turned negative, leading to the increase in resistance of the device. It was observed that the resistance of the film, as a whole, was dependent on how much charge had been passed through it in a particular direction. In addition, the resistance value of the film was found to be reversible on changing the direction of the current [19], [27], [28]. The HP device was considered as a nonionic device owing to its property of displaying fast ion conduction at nanoscale.

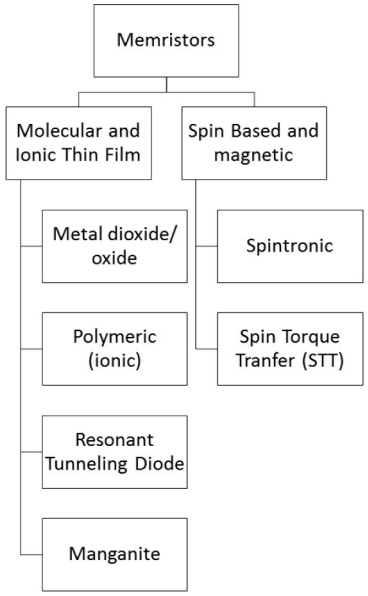


Fig. 2. Taxonomy of memrisors based on memristive behavior observed in fabrication materials [60].

In 2008, Strukov *et al.* [16] presented analytical results, which showed that the memristance naturally arises in nanoscale systems in which solid-state electronics and ionic transport are coupled under an external bias voltage. This paved a way to set the foundation for understanding a wide range of hysteretic current–voltage behavior observed in many nanoscale electronic devices that involved the motion of charged atomic or molecular species; in particular, certain titanium-dioxide cross-point switches [16], [19]–[21], [29]–[55]. In 2008, Chen [56] recounted the impact of invention of memristive devices in technology as recognition to the promising discovery of memristors by Chua. The modeling aspect of engineering that involved memristors and memristive systems was discussed in-depth by [98]. In 2010, the HP laboratories introduced practical memristors of size 3 nm $\times$ 3 nm found operable at a switching time of 1 ns ($\sim$1 GHz) [57]. Memristors have thus paved a way for the further miniaturization of integrated electronic circuits [58], promising a future in technology beyond Moore's law [59].

The memristive systems based on device properties can be broadly grouped into those based on molecular and ionic thin films, and into those based on spin and magnetic effects. An overall taxonomy of the memristors based on their memristive material properties [60], [61] has been presented in Fig. 2. The ionic thin-film and molecular memristors mostly rely on different material properties of the thin-film atomic lattices that display hysteresis below the application of charge. These memristors can be classified into four distinct groups viz., metal-dioxide memristors, ionic or polymeric memristors, resonant tunneling diode memristors, and manganite memristors. The metal-dioxide memristors, titanium oxide in particular,

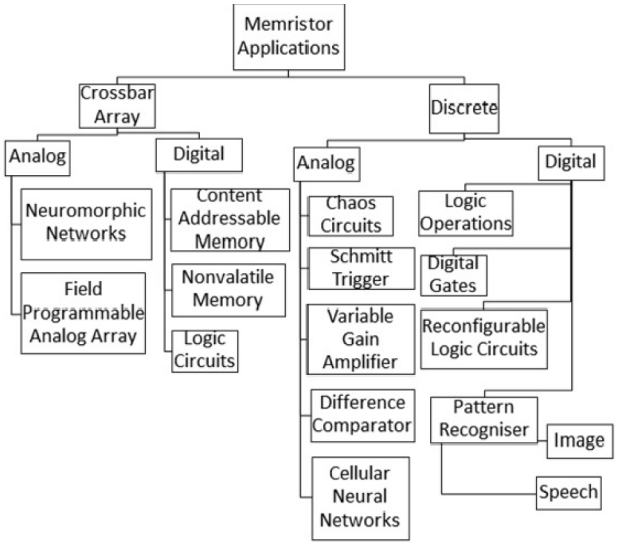


Fig. 3. Modified taxonomy of memristor circuit applications from that reported in [62] to the incorporating emerging applications such as in imaging and speech.

are broadly explored for designing and modeling. The ionic or polymeric memristors utilize dynamic doping of inorganic dielectric type or polymer materials. In this type of memristors, the ionic charge carriers move all over the solid-state structure. The resonant tunneling diode memristors use specially doped quantum-well diodes of the space layers between the source and drain regions. The manganite memristors use a substrate of bilayer oxide films based on manganite as opposed to titanium-dioxide memristors.

The magnetic and spin-based memristors are opposite to ionic nanostructure and molecule-based systems. This category of memristive devices solely rely on the degree of electronic spin and its polarization. This memristor type can be categorized further into two types viz., spintronic memristors and spin torque transfer (STT) memristors. In spintronic memristors, the route of the spin of electrons changes the magnetization state of the device, which consequently changes its resistance. In STT memristors, the comparative magnetization position of the two electrodes affects the magnetic state of a tunnel junction, which in turn changes its resistance. Since HP's announcement, interest in memristive electronics and their applications has rapidly grown with several research groups, demonstrating memristive behavior in different devices and systems. Fig. 3 shows the taxonomy of memristor applications proposed by Mazumder *et al.* [62] in 2012. The devices reported in the literature all have different underlying physics governing their memristive behavior.

B. Threshold Logic

The first simplified mathematical model of the biological neuron was introduced in McCulloch and Pitts [17] in 1943 in the form of the TLG. It computed the sign of the weighted sum of inputs

$$f(x_1, x_2, \ldots x_n) = \text{sgn}(w_1x_1 + w_2x_2 + \cdots + w_nx_n) = \text{sgn}(\Sigma_{i=1}^{n} w_i x_i - T) \quad (1)$$

where the values of w_i are the synaptic weights associated with the inputs x_i, and T is the threshold that the gate needs to meet to fire, and n is the fan-in of the TLG [63]. Since then, a large number of hardware implementations have been reported in the literature, a comprehensive survey of which can be found in [63]. The implementations range from the early electromechanical (tubes, motors, and clutches) neurocomputer in 1951 [64] to the potentiometer *perceptron* in 1957 [65], to the electrically adjustable memristor-based adaptive linear element [46], and to the various VLSI implementations in the last two decades. These implementations are too many to be included in this survey, but they can be broadly divided into CMOS, capacitive, output-wired inverters, floating-gate, and pseudo-nMOS solutions, each catering to a specific need or one or more performance parameters, such as power dissipation, noise margins, sensitivity to process variations, and fan-in.

The survey in [63] concludes with an interesting observation. The authors provided a list of potential applications of TL apart from hardware neurons, such as "... general microprocessors, DSPs, and cores where addition, multiplication, and multiply-accumulate, ...encryption/decryption, ...convolution/deconvolution, ...and compression/decompression [66]." The fundamental reason, the authors point out, is that the TLGs need full custom design and have been in direct competition with Boolean gates and a lot of research effort has been spent on improving the Boolean logic gates since the 1970s. In this survey, we will identify approaches and techniques on how to use TL based on the memristive systems to build different logic functionalities, which will, we hope, direct some of the future research efforts in this direction. Beiu *et al.* [63] offer another insight: "Lastly, because nano (and reconfigurable) computing will probably get center-stage positions in the (near) future, TL will surely benefit from that." This observation is another motivation of the current survey, because the MTL, as we shall see, is capable not only of emulating synaptic action in hardware, but it does so in a highly dense, low power, and scalable architecture.

III. Threshold Logic Gates and Application Circuits Using Memristors

The resistive switching nature exhibited by memristors [67]–[69] has been utilized in realizing brain-inspired TL computing circuits. Here, we present a review of the different implementations of the MTL, which realizes specific function pertaining to the application under consideration.

An early implementation of the TL by using memristors was proposed by Rajendran *et al.* [70] in 2009. The proposed programmable TL array (PTLA) by using memristors exhibited multiple levels of resistance to provide weighted inputs to each threshold gate. The distinguishing feature of the PTLA was the combination of a negative differential resistance-based molecular switch and a multilevel resistance memristor leading to the programmable threshold gate. The design was extended to the implementation of an image classifier, which classifies a 3×5 image into a rectangle or a triangle by using three 5-input TLGs in the first level and a goto pair-based majority voter, as shown in Fig. 4. A goto pair or twin [71] is a

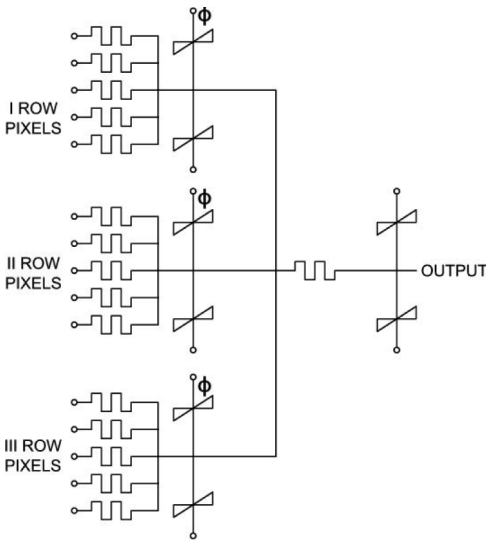


Fig. 4. TLG implemented using memristors and goto pair [70].

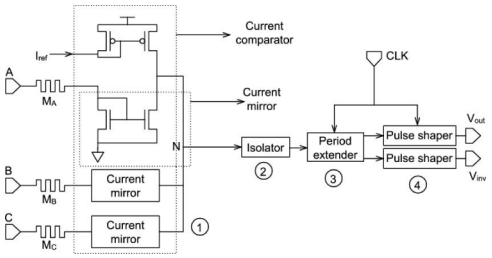


Fig. 5. Circuit diagram of a three-input threshold gate using memristors as weights. 1: current mirror to prevent reverse flow of current. 2: isolator to prevent loading. 3: period extender to retain input pulse period. 4: pulse shaper to retain memristance [72].

series connection of two tunnel negative resistance diodes that has been used as a majority voting circuit. When the operating voltages are just enough to bring the pair to either of the stable states (0 or 1), a majority circuit can be built as in [71] to output the majority value among the inputs.

In 2010, Rajendran *et al.* [72] implemented different Boolean functions by programming the weights at the input of gates in the proposed threshold gate-array architecture. The key idea was to use the memristors as weights of the inputs to a threshold gate, as shown in Fig. 5. Additional circuits were employed to avoid the problems arising from the straightforward implementation of the memristor-based threshold gate, such as reverse current from the output to the input of the circuit, loading of the current comparator by the next stage, partial restoration of the input, and temporal change in memristance. To avoid these problems, the authors added circuitry, namely: 1) a two-transistor current mirror on each input line; 2) a two-transistor isolation circuit; 3) a period extender circuit to restore the negative pulse immediately following the positive pulse to represent the logic 1; and 4) a six transistor pulse shaper circuit to generate a positive pulse followed by a negative pulse to represent a logic 1. In general, an n-input threshold gate required $(2 \times n) + 18$ transistors. The proposed 3×3 crossbar-based island

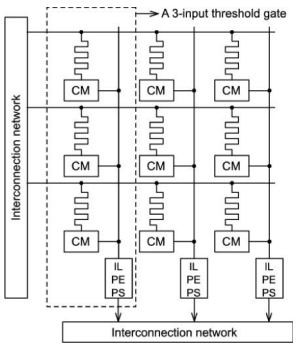


Fig. 6. Island of threshold gates [72]. Each column corresponds to a single threshold gate and the number of rows determines the fan-in of the threshold gate. Dotted box: single threshold gate. IL: isolator. CM: current mirror. PE: period extender. PS: pulse shaper.

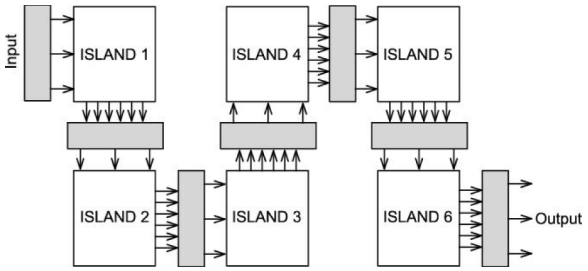


Fig. 7. Cascaded architecture using two rows of islands [72]. Unshaded box: island of threshold gates. Lightly shaded box: interconnection network.

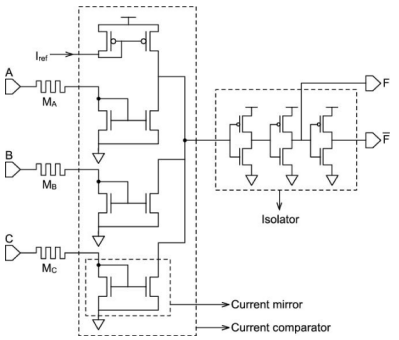


Fig. 8. Three-input MTL gate with memristors as weights and Iref as the threshold [73].

architecture as shown in Fig. 6 consisted of three 3-input threshold gates. A cascaded architecture formed using these threshold gate islands connected to each other through an interconnection network shown in Fig. 7 prevented signal degradation in successive stages with the help of the period extender and pulse shaper, which provided signal and memristance restoration. The programming circuitry consisted of a pulse generator to program the memristor. The memristance property utilized in the architecture was able to reduce the power consumption and effective area footprint to $\sim$75% in comparison with the CMOS-based lookup tables (LUTs). The delay penalty of the programmable threshold gates was found as almost $12\times$ the delay of the four-input LUTs. The memristors were utilized as weights in the realization of

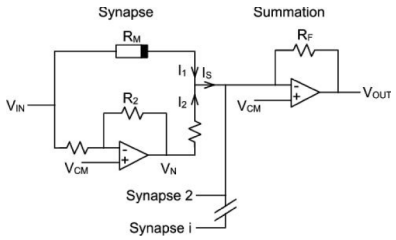


Fig. 9. Circuit implementation with memristive devices with circuitry for negative weights [74].

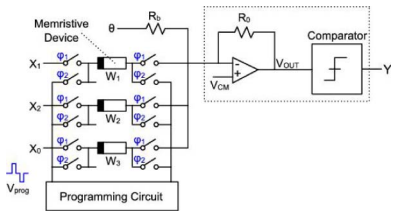


Fig. 10. Circuit implementation for programming memristive devices [74].

low-power field programmable gate arrays (FPGAs) using the TL in [73]. The proposed MTL gate shown in Fig. 8 utilized the multiresistance property of memristors to implement the Boolean functions, which are the subsets of threshold functions. The programmable threshold gates consume less power and area when compared with their implementations using CMOS, LUTs, and Capacitive Transistor Logic gates. The energy performance and area overhead of the TL implementation were evaluated using the CAD tools from the Cadence and Berkeley SIS logic synthesis tool. In addition, the essential counter measures to combat the issues of using memristors in logic circuits in the presence of memristance driftlike memristor refresh were proposed.

Another circuit design capable of realizing four different logic operations by changing the resistance of the memristive devices was proposed by Tran *et al.* [74] in 2012. The design exploration of reconfigurable TLG implemented using silver chalcogenide memristive devices [75] combined with CMOS circuits was presented in this paper. The proposed re-programmable TLG shown in Fig. 9 was realized in discrete hardware using a summing op-amp circuit with the memristive devices implementing the weights (w_i). A feedback-based adaptive programming circuit, shown in Fig. 10, was developed to program the individual memristive devices to the predetermined resistance values to create each logic operation. According to the analysis carried out in MATLAB-Simulink/Cadence, the TLGs with Ag–Ch memristive devices had a fan-in of >10, switched at >1 GHz speed, and dissipated lower static power than a corresponding CMOS implementation. In 2012, Manem *et al.* [76] added another contribution to the field of neuromorphic computing using memristor-based threshold gates by introducing a variation-tolerant training methodology to efficiently reconfigure memristive synapses in a trainable threshold gate array (TTGA) system. The TTGA consisted of the arrays of trainable threshold gates [73] interleaved with switch blocks to enable the realization of the reconfigurable

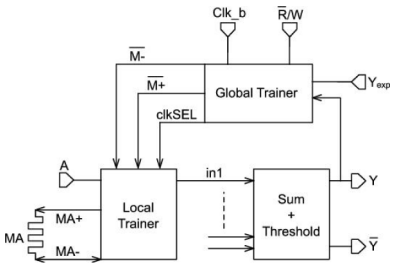


Fig. 11. Fully connected perceptron with training circuitry [76].

logic fabric as shown in Fig. 11. A single layer four-input perceptron (trainable threshold gate) capable of implementing all linearly separable one-, two-, three-, and four-input Boolean functions [77] was considered to be the unit-sized configurable logic block. The proposed TTGA system was designed and implemented from trainable perceptron-based threshold gates [77] in Cadence Spectre with 45-nm Berkeley predictive technology models (PTMs) for the CMOS circuitry. The proposed training methodology based on the stochastic gradient descent training technique was capable of efficiently reconfiguring the memristive synapses in a TTGA. The technique overcame many circuit level issues, such as parasitics and device variations that configured memristive devices. The training and performance results for the TTGA and the one transistor and one memristor (1T1M) multilevel memristive memory [42], [78], [79] showed that the TTGA (minimum memristance values, i.e., OR pretrain) to be the most energy-delay and area-effective solution. The methodology was observed as robust to the unpredictability of CMOS and nanocircuits with decreasing technology sizes.

In 2013, Ligang *et al.* [80] proposed that a programmable TLG can be implemented using a hybrid CMOS/memristor logic. The proposed linear threshold gate (LTG) shown in Fig. 12 was comprised of memristive devices in a universal gate, which is much more powerful than similar fan-in single NAND or NOR gates. The memristive devices implemented a ratioed diode–resistor logic, wherein several (N) memristive devices connected in parallel to a single pull-down resistor R_L, so that a dynamic range (i.e., the ratio $R^H_{\rm ON}/R^L_{\rm ON}$) dictated the number of different Boolean functions the LTG could implement. This configuration made LTG in-field configurable and potentially very compact.

The concept was experimentally verified by implementing a four-input symmetric LTG with an integrated circuit CMOS flip-flop, silicon diodes, and Ag/a-Si/Pt memristive devices. For their effectiveness in high-throughput pipelined circuits, the CMOS flip-flop was preferred over a CMOS gate (and possibly an inverter) to restore the output voltage to a clear binary. The proposed implementation claimed to be more robust as compared with the approaches suggested in [70] and [81], since it does not rely on changing the state of memristive devices during the logic operation. In addition, each memristive device in the suggested TL in [70], [72], and [82] was served by a CMOS-based current mirror circuit, which leads to considerable overhead. However, the memristive devices in the proposed hybrid architecture could be integrated into crossbar circuits, which are pat-

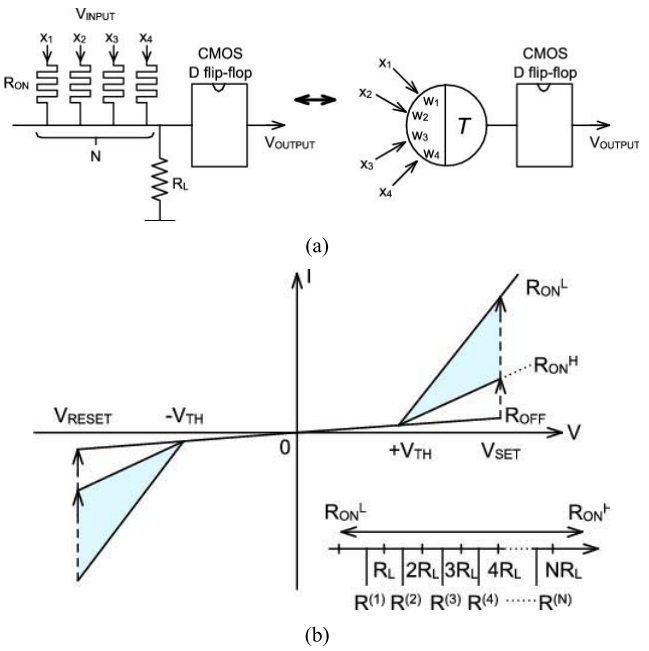


Fig. 12. (a) Main idea of an LTG implemented with memristors and CMOS D flip-flop. (b) I–V characteristics of memristive devices (schematically represented). Shaded area: range of possible intermediate states utilized for the implementation of the threshold gate [80].

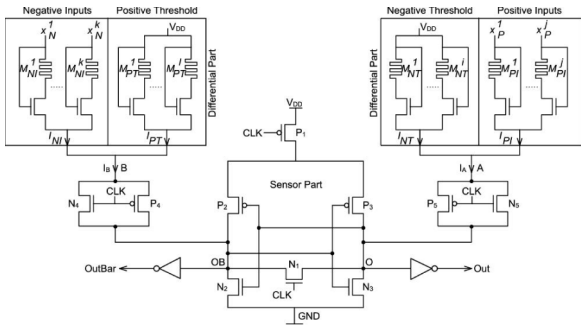


Fig. 13. CMMTL [84].

terned above the CMOS layer, e.g., similar to the proposed (3-D) hybrid CMOS/molecular circuits concepts [43], [83], such that the threshold gate area would be determined mostly by the CMOS flip-flop. The method required only some minor additional circuitry for programming the memristive devices.

A new clocked design that combines memristors and CMOS transistors to implement current mode logic TL gates was presented by Dara *et al.* [84] in 2013. The novel current-mode memristor-based TL (CMMTL) shown in Fig. 13 consisted of two parts, a differential part and a sensor part, the differential being a series combination of a memristor and n-MOS, and having two further divisions comprising a negative threshold and positive inputs, and a positive threshold and negative inputs. The inputs are applied to the positive/negative parts of the differential part. When, for a given input configuration, the current I_A through Node A is greater than the current I_B through Node B, the voltage at output Node O rises faster than the voltage at output Node O_B resulting in a high voltage at the Node O and low voltage

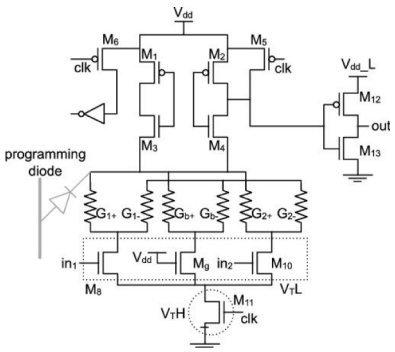


Fig. 14. Circuit for a 2-fan-in DRTL gate [85].

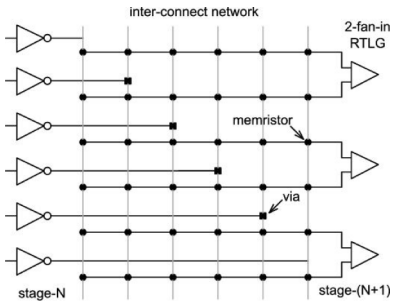


Fig. 15. Schematic for interconnect design for DRTL using resistive crossbar memory [85].

at Node O_B. The inverse voltage allocation at Node O_B and Node O happens when I_B is greater than I_A. The approach was found to outperform the combinational design [73] proposed earlier in terms of performance and energy consumption on three-, four-, and five-input benchmark threshold functions. The percentage improvements were summarized as 77% for the delay, 50% for the energy, and 88% for the energy-delay product (EDP). The delay and energy consumption, using the proposed implementation, using Berkeley PTMs for 45-nm CMOS transistors were 0.44 ns and 3.410, respectively. The proposed CMMTL is capable of implementing all possible weight configurations, i.e., positive-weighted gates, negative-weighted gates, and positive- and negative-weighted gates. The proposed method scales well over [73] as indicated by the increase in average EDP with the increase in number of inputs to the threshold function. Both the current mode logic [84] and the combinatorial design of [73] the used sense amplifiers to restore output voltage swing as opposed to CMOS D-flip-flops of [80].

A dynamic resistive TL (DRTL) design based on nonvolatile programmable resistive memory elements for reconfigurable computing was proposed by Sharad *et al.* [85] in 2013. In DRTL shown in Fig. 14, the resistive memory elements are used to implement the weights and the thresholds, while a compact dynamic CMOS latch is used for the comparison operation. The multiple stages in a DRTL design could be connected using energy-efficient low-swing programmable interconnect networks based on resistive switches. The dynamic operation of the CMOS latches that minimizes static-power dissipation [43], [72], [74], [86] along with the memory-based compact logic and interconnect design shown in Fig. 15

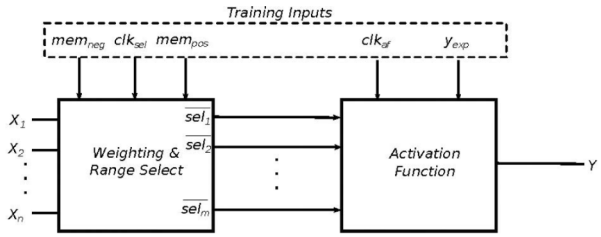


Fig. 16. NLB design from [87].

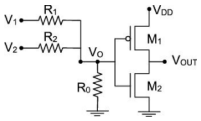


Fig. 17. Circuit diagram of the resistive divider Boolean logic cell that consists of a two-input resistive divider and a variable threshold CMOS inverter [88].

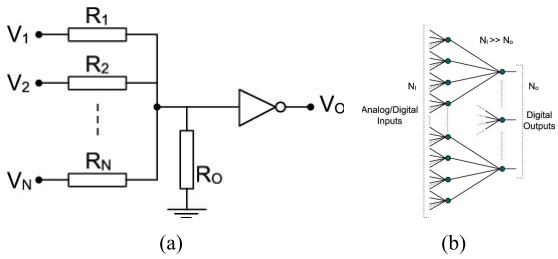


Fig. 18. Memory network cell and architecture. (a) Cell structure. (b) Network architecture (each node represents a cell) with no crossover wiring. Cells are arranged in hierarchical manner with $N1 \gg N0$ [88].

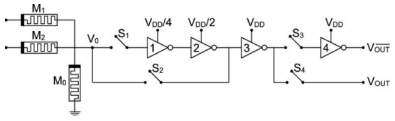


Fig. 19. Circuit diagram to implement NAND, NOR, AND, OR, and NOT logic functions consisting of memristive resistance divider and CMOS inverters with three different power supply values [88].

makes DRTL a dynamic, pipelined logic scheme with low power consumption and high performance. The performance analysis of the DRTL gate and the interconnect design was evaluated by comparing the performance of DRTL with that of four-input LUT-based CMOS FPGA [33], for some ISCAS-85 benchmarks. The performance results show the possibility of 96% higher energy efficiency and more than two orders of magnitude lower EDP for DRTL.

In 2013, Soltiz *et al.* [87] presented a robust and area efficient hardware implementation of a neural logic block (NLB) with an adaptive activation function, containing a weighting and range select and an activation function, as shown in Fig. 16. The main motivation was to make threshold function adaptive, so that nonlinearly separable functions, such as XOR, could be implemented within a single layer. Inspired by the principle of neuromodulators in the brain, the adaptive activation function comprised of m points can model any function with $(m-1)$ boundaries. The weighting and range select component applies an adjustable weight to each input, calculates the weighted summation, and determines which of the m ranges the summation falls in. The activation function associates a digital output with each range. The inputs are passed through the memristors that are trained to a memristance M, ranging from R_{ON} to R_{OFF}. To change the functionality, a different value of the memristance might be selected. The authors note that while the proposed design adds significant complexity when compared with a threshold activation function, the adaptive activation function provides benefits of fast training convergence times when compared with a NLB that only adjusts input weights, by training the shape of the activation function. On an optical character recognition application the work shows a 90% improvement in the EDP over LUT-based implementations.

A universal Boolean logic cell based on an analog resistive divider and TL circuit useful for mimicing brain-like large variable logic functions in VLSI was proposed by James *et al.* [88] in 2014. The logic cell shown in Fig. 17 employed a CMOS—resistance TL codesign, which successfully optimized the circuit design of conventional CMOS-based large variable Boolean logic problems. In the proposed resistance-based TL family, the resistive divider was implemented using memristors. The output of resistive divider was then converted into a binary value by a threshold operation implemented by CMOS inverter and/or op-amp. For a two-input resistance divider circuit, if the threshold voltage of the inverter was set between 0 and 1/3 V, the cell would work as NOR logic, and if it was between 2/3 and 1/3 V the cell would work as NAND logic. To operate the cell with a large number of inputs (>20), the threshold voltage of the inverter, for example in the case of NOR, needed to be lowered to a very small range. To accommodate this effect, the authors introduced three inverters (Fig. 18) with three different V_{DD} values to form a universal gate structure to implement AND, NAND, OR, NOR, and NOT logics. For the cell to work as a NAND logic, the switches S_2 and S_4 were closed, and the output was taken from V_{out}, so that three inverters would be enabled. To implement AND logic, the switches S_1 and S_3 were closed, and the output was taken from $\overline{V_{\mathrm{out}}}$. If the switches S_1 and S_4 were closed, a NOR logic from V_{out} was achieved: here only one inverter had to be enabled. If both S_2 and S_3 were closed, OR logic could be implemented. The proposed universal logic cell was based on the cognitive-memory network [89], a resistive memory network that has no crossover wiring that overcame the hardware limitations to size and functional complexity associated with conventional analog NNs. The universal logic cell shown in Fig. 19 was employed to realize in an application to implement conventional digital logic gates. The simulation was performed in SPICE using feature size of 0.25-m TSMC process The BSIM models and the HP memristor model for comparison with the CMOS implementation using a 16 b adder and a 16×1 MUX. The analysis shows that the proposed cell offered the advantages of smaller area and design simplicity in comparison with CMOS-based logic circuits when the number of input variables became very high.

In 2014, Fan *et al.* [90] proposed a spintronic threshold device (STD), which can be combined with CMOS compatible Ag–Si memristors for designing ultralow-energy spin-MTL (SMTL). The SMTL gates shown in Fig. 20

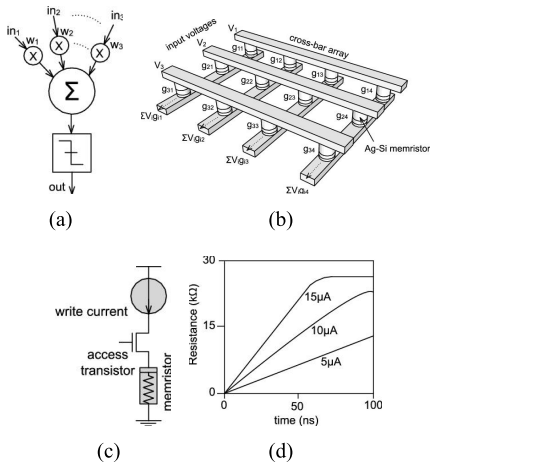


Fig. 20. (a) Schematic of a TLG, (b) MCA, (c) resistive memory cell with access transistors, and (d) transient change in resistance for different magnitudes of programming current [90].

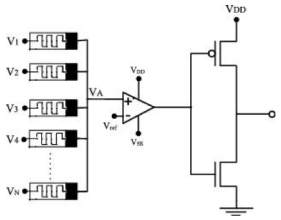


Fig. 21. MTL cell [93].

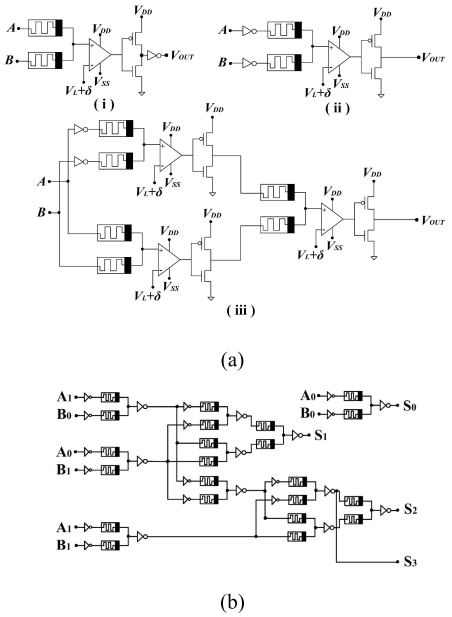


Fig. 22. (a) Circuit diagrams of the logic gates using proposed cell. (a.i) OR gate. (a.ii) AND gate. (a.iii) XOR gate. (b) 2-b memristive threshold vedic multiplier [93].

employ a memristive crossbar array (MCA) to perform current-mode summation of binary inputs. The low-voltage fast-switching STD shown in Fig. 20, based on magnetic domain wall (DW), was found suitable for the design of energy-efficient SMTL. The spin-torque switches based on magnetometallic DW motion [91], [92] allows ultralow voltage operation of memristive TLGs leading to low energy dissipation at the gate level. Field programmable SMTL gate arrays were found to operate successfully at a small terminal voltage of $\tilde{5}0$ mV, resulting in ultralow power consumption in gates as well as programmable interconnect networks. The performance analysis done on common benchmarks shows that the proposed hardware can achieve more than 100× improvement in energy and 1000× improvement in EDP, as compared with the state-of-art CMOS FPGA-based TLG. TL computing using hybrid memristive-CMOS cell architecture designed for fast Fourier transform (FFT) and vedic multiplication has been proposed by James *et al.* [93] in 2014. The proposed architecture involved a memristive-threshold circuit configuration, which consisted of the memristive averaging circuit in combination with operational amplifier and/or CMOS inverters as shown in Fig. 21 in application to realizing complex computing circuits. The developed TL claims to outperform the previous memristive-CMOS logic cells by providing lower chip area, lower Total Harmonic Distortion, and controllable leakage power; except for a higher power dissipation with respect to CMOS logic.

The proposed MTL cell was designed specifically for implementing FFT and multiplication circuits, as shown in Fig. 22, in modern microprocessors with the desired lower power dissipation and smaller on-chip area footprint using nanoscale fabrication techniques. Some additional desirable features of the proposed cell include the generalization ability of the cell with a single cell structure with multiple functionality; and robustness to process variability in temperature, memristances, and technology lengths indicating the fault tolerance ability of brainlike logic circuits. The paper reports the successful application of the MTL cells in the examples of FFT and vedic multiplication computing circuits.

In 2015, Soudry *et al.* [94] proposed memristor-based grid to perform multiplication operation for learning backpropogation algorithms in multilayer NNs. Synapses comprising one memristor and two CMOS transistors were set in a grid formation each receiving two complimentary read/write pulses and an enable signal, and outputting on a current line. The column inputs fed the training data and row inputs the classification label. The two computational bottlenecks, the authors addressed in gradient descent machine learning algorithms, were matrix × vector and vector × vector multiplications. The matrix × vector product was implemented through the memristive grid by multiplication through the Ohms law and the analog summation of currents. The vector × vector product was done by using time × voltage paradigm under the approximation that given a voltage pulse, the conductivity of a memristor would increment proportionally to the pulse duration multiplied by the pulse magnitude. The authors proposed that the area and power consumption were expected to be reduced by a factor of 13–50 in comparison with standard CMOS technology.

Yang *et al.* [95] in 2014 used oxide-based resistive RAM devices to implement TL in order to compute logic functions with low power, robust circuits at low supply voltages (0.6 V). They used a differential TLG (DTG) circuit that consisted of five main components (Fig. 23): 1) a differential sense amplifier, which consists of two cross

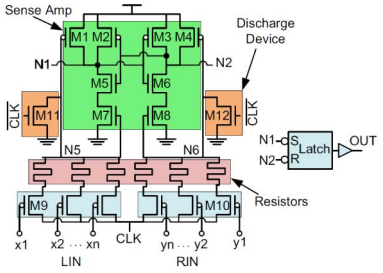


Fig. 23. Schematic of a DTG circuit [95].

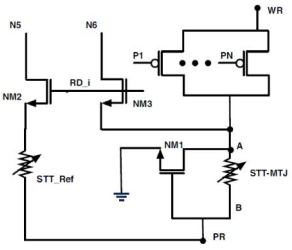


Fig. 24. STL-Cell [96].

coupled NAND gates; 2) an SR latch; 3) two discharge devices; 4) left input network (LIN) and right input network (RIN); and 5) a network of resistors. The resistive network was implemented using a oxide-based random access memory to behave as a CMOS-compatible nanoscale resistor. The circuit outputs a logic 1 based on the inequality of number of active p-FETs in the LIN and RIN networks. The benefits of robustness, area, and energy delay were demonstrated on a 16-b full adder and a 128-b comparator.

Earlier in 2012, Nukala *et al.* [96] from the same group had used an STT-magnetic tunneling junction (STT-MTJ) device with conventional MOSFETs to build a TL architecture. The resulting cell was used to program a large number of TL functions, many of which would require a multilevel network of conventional CMOS logic gates. Based on an array architecture of these cells, they demonstrated the advantages of nonvolatility and zero standby power on a 16-b carry look-ahead adder and compared with two conventional FPGA implementations. The array had 12× lower transistor count (compared with CLA-FPGA) and 10× reduction (compared with ripple carry-ahead FPGA) with comparable energy. The cell is shown in Fig. 24. In the write phase of the cell, Write (WR) is asserted, so that a certain amount of current (I) flows through the STT-MTJ device depending on the number of ON p-MOS transistors. If $I \geq (Ic)$, the switching current, then the STT-MTJ device switches to the low-resistance state otherwise it remains in high-resistance state. When the WR pulse goes low, no current flows through the STT-MTJ device, and since the device is nonvolatile, the state is maintained.

In 2015, Maan *et al.* [97] presented a programmable MTL circuit design for real-time detection of moving objects. The proposed TL shown in Fig. 25, which was targeted at high-speed imaging that could lead to near-continuous real-time object tracking and even surpass human object tracking ability, outperformed CMOS equivalent implementation in terms of area, leakage power, power dissipation, and delay.

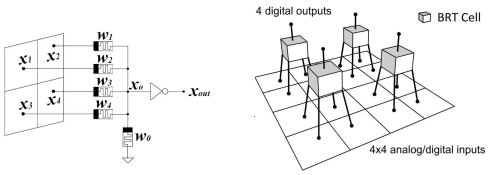


Fig. 25. (a) Modular bilevel programmable resistance MTL cell that used four-input pixel values. (b) Architectural representation of the cell arrangement for a 4 × 4 pixels image; in this configuration, each cell uses four analog/digital inputs, has bivalued weights, and implements image dimensionality reduction [97].

TABLE I
IMPLEMENTATIONS OF AN MTL—A COMPARISON

Methodology	Area	Power (mW)
J. Rajendran et al. (2009) [70]	77.52 pm^2	44.12
J. Rajendran et al. (2010) [72]	234.46 pm^2	32.56
J. Rajendran et al. (2012) [73]	193.77 pm^2	31.3
T. Tran et al. (2012) [74]	4.0 nm^2	30.8
H. Manem et al. (2012) [76]	193.87 pm^2	45.7
L.Gao et al. (2013) [80]	0.4992 μm^2	28.81
C.B. Dara et al. (2013) [84]	270.55 pm^2	118.22
James A.P. et al. (2014) [88]	155.52 pm^2	45.34
James A.P. et al. (2015) [93]	2.08 nm^2	30.72
A. K. Maan et al. (2015) [97]	38.91 pm^2	43.66

The proposed resistive switching-based TL cell comprised of a voltage divider circuit and a CMOS inverter gate, which encoded the pixels of a template image. The presented logic provided a framework to implement brainlike logic in a memory and learning driven detection of multiple objects.

In Table I, the MTL implementations that were discussed have been compared on the basis of their area and static power dissipation. The analyses have been carried out in circuits by realizing a two-input logic function using the proposed MTL techniques using simulations under the same technology parameters (i.e., HP memristor model with 0.25u TSMC technology). According to Table II, [97] requires the lowest implementation area, which is 38.91 pm^2, and [80] has the lowest static power consumption, which is 28.81 mW. Among the two-input cells implemented to process the image pixels [70], [97], the MTL design proposed by Mann *et al.* [97], in 2015, has the lowest area and static power consumption. From the memory crossbar array implementations [85], [90] compared in Table I, the MTL design proposed by Fan *et al.* [90], in 2014, has the lowest EDP.

In Table II, a comparison between the various performance parameters of the TL cells implemented with memristors that have been reported in the paper have been summarized. These example applications of TL cells does not directly imply high degree of generality but indicate that the TL circuits are scalable and can find its use in a wide range of tasks. The circuits are also arranged with respect to the applications from the more general logic gate applications to very recent pattern recognition applications. Recent works indicate the power dissipation to be a concerning factor in the design of these circuits and require changes in the circuit configurations as well as memristor device design. The low area footprint for

TABLE II
COMPARISON OF TL IMPLEMENTATION USING MEMRISTORS

Methodology	Area	Power	Leakage Power	Speed	Fan-in	Fan-out	Target application	Energy	Delay	Energy Delay Product (EDP)
J. Rajendran et al. (2009) [70]	NR	0.57- 1.56μW	NR	NR	NR	NR	Image Classification	NR	40 ps	NR
J. Rajendran et al. (2010) [72]	28*	42 μW	NR	NR	NR	NR	NR	NR	6.1 ns	NR
J. Rajendran et al. (2012) [73]	182**	NR	NR	NR	NR	NR	Boolean Logic Gates	6.3fJ', 3.15fJ"	2.99ns', 5.84ns"	NR
T. Tran et al. (2012) [74]	NR	NR	NR	$>$1 GHz	$>$10	NR	Digital Logic Gates	NR	NR	NR
H. Manem et al. (2012) [76]	NR	NR	NR	NR	NR	NR	Memory	3.82nJ	189ps', 180ns"	721.98zsJ', 687.6zsJ"
L.Gao et al. (2013) [80]	NR	NR	NR	NR	NR	NR	Digital Logic Gates	NR	NR	NR
C.B. Dara et al. (2013) [84]	NR	NR	NR	NR	NR	NR	Boolean Logic Gates	3.41fJ"'	0.42ns"'	1.43ysJ"'
M. Sharad et al. (2013) [85]	NR	NR	NR	2 GHz	NR	NR	Programmable Logic Hardware, Crossbar memory	480fJ***	0.5ns***	240ysJ
James A.P. et al. (2014) [88]	$70pm^2$	9.2 μW	0.014 nW	NR	$14.498M$	NR	Boolean Logic gates, Arithmetic modules	NR	0.45 μs	NR
Deliang Fan et al. (2014) [90]	NR	NR	NR	NR	4	NR	Programmable Logic Hardware	$\approx$300fJ***	$\approx$2ns***	$\approx$0.6zsJ
James A.P. et al. (2015) [93]	$4.55mm^2$	3.00 μW	14.30 pW	1 GHz	NR	NR	FFT and Vedic Multiplication	0.30pJ	NR	NR
A. K. Maan et al. (2015) [97]	$9.66mm^2$	12.30 μW	12.25 pW	100 MHz	NR	NR	Object Detection	NR	NR	NR

NR - Not Reported in the reference work. *Transistor count **Transistor count for C17-ISCAS-85 benchmark *** C432-ISCAS-85 benchmark; 'ISCAS-85 Benchmark C17-Before refresh " ISCAS-85 Benchmark C17-After refresh
'Writing to 1T1M memory circuit of physical array size 4 x 4; "Reading from 1T1M memory circuit of physical array size 4 x 4 "'Implementation of 3-input function "A+BC" using CMMTL

on-chip implementation has been the key attraction of these logic family of gates and circuits. The possibility that the TLG with memristors can lead to MtM's law outweighs the limitations of the power dissipation and lower speeds.

IV. DISCUSSION AND OPEN PROBLEMS

The important properties of low leakage currents, ability of switching into memorized resistor levels, and smaller on-chip area make memristor an attractive element to mimic the principle of firing of neurons in silicon. Being a resistive device, the biggest challenges in designing memristor-based circuit configurations are with reduction in power dissipation in comparison with CMOS-only circuits. The resistive path introduced by memristive devices within the designed TL circuits often drives larger currents through the circuits resulting in higher power dissipation. Recently, the CMOS-memristor hybrid circuits have gained popularity as CMOS is a matured process technology, however, is gain-limited by higher power dissipation compared with CMOS-only counterparts. In addition, in MOS-gated memristive arrays the leakage currents become an important issue, which requires specialized compensated read-out techniques. Programmability of the memristive states in hybrid circuits become a challenging problem often requiring complex add-on circuits. If not designed carefully, the area of programming circuits can outweigh the benefits of lower area offered by the memristors.

The progress in the VLSI circuits development depends also on the availability of accurate models for memristors, which can be incorporated into widely accepted simulation tools. Although there are several simulation models available in SPICE and VerilogA, they still do not cover the wide range of memristive devices available for use today. In addition, the physical level design of memristors and memristor-CMOS hybrid circuits is challenged by the lack of process standards and requires further adoption of these devices in semiconductor industry. In this sense, memristor devices and circuits are quite early in the stages of commercial implementation. We note that while there are aforementioned issues, the ability of MTL cells to be general and programmable makes it an interesting alternative to CMOS logic family of gates. In addition, the ability of the TL cells to be programmed to behave like a pattern recognizer mimics the principle of neurons that can provide multiple functionalities with the same cell structure.

The hardware programmability opens up a wide range of applications for circuit designers and artificial intelligence researchers. Some of these are: 1) development of memories that can learn, store, and forget like human memory; 2) development of humanlike self-learning classifiers in silicon; 3) building higher level intelligent modules on programmable and self-learning chips; 4) building array processing circuits and reinventing analog sensory processing, such as in computer vision; 5) developing the CAD tools and systems for memristive hybrid circuits; and 6) the development very large scale simulation and implementation of such TL systems.

Memristive behavior is observed in devices dating back to electric arc experiments performed by Sir Humphry Davy in 1808. Although lost in the history, the resurgence of memristive device as a possible front-runner in the implementation of NN is made possible mainly due to the advances in nanoscale technologies in last decade. In addition, the limitations imposed by scaling issues of MOSFET devices, and the need to create large scale NNs for imitating silicon brain continue to inspire research in this field.

V. CONCLUSION

This paper discussed the impact of the MTL by sketching the number of applications that have sprung up utilizing the inherent nature of several devices that show memristive behavior. This paper reviewed the different circuit implementations of the TL that employed the resistive-switching nature of memristors potentially leading to highly efficient, high-density neurocomputing devices in the future. Even with several new advances, this field of research is relatively very new, and offers several challenges and opportunities for NNs, learning systems, circuits and systems communities to work together. The practical applications of the MTL design extend to real-time processing and recognition of natural signals, its differentiation, and efficient architectures for silicon memories. While the possibilities are many there exist several challenges as well, including the need for better energy and power dissipation ratings, newer circuits for efficient programming of memristor arrays, accurate simulation models for memristors, and long-term development of foundry support.

ACKNOWLEDGMENT

The authors would like to thank all the reviewers for the constructive feedback and comments, which has helped to improve the overall quality of this paper.

REFERENCES

[1] L. O. Chua, *Introduction to Memristors*. IEEE Expert Now Educational Course, 2009.
[2] R. Tetzlaff, Ed., *Memristors and Memristive Systems*. New York, NY, USA: Springer, 2014.
[3] C. Yakopcic, E. Shin, T. M. Taha, G. Subramanyam, P. T. Murray, and S. Rogers, "Fabrication and testing of memristive devices," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2010, pp. 1–4.
[4] X. Hui, X. Tian, H. Nie, Q. Li, H. Liu, and X. Xu, "On development and expectation of memristor," in *Proc. Int. Conf. Intell. Syst. Design Eng. Appl. (ISDEA)*, vol. 1. Oct. 2010, pp. 1035–1038.
[5] G. S. Rose, "Overview: Memristive devices, circuits and systems," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May/Jun. 2010, pp. 1955–1958.
[6] F. Corinto, P. P. Civalleri, and L. O. Chua, "A theoretical approach to memristor systems," in *Proc. 14th Int. Workshop Cellular Nanosc. Netw. Appl. (CNNA)*, Jul. 2014, pp. 1–2.
[7] G.-Q. Zhang, "'More than Moore'—The changing international landscape, strategy and solutions of micro/nanoelectronics," in *Proc. 8th Int. Conf. Electron. Packag. Technol.*, 2007, p. 1.
[8] W. Arden, M. Brillouët, P. Cogez, M. Graef, B. Huizing, and R. Mahnkopf, Eds., "More-than-Moore," Int. Technol. Roadmap Semicond., White Paper Version 2:14, 2010.
[9] G. Q. Zhang, F. van Roosmalen, and M. Graef, "The paradigm of 'more than Moore,"' in *Proc. 6th Int. Conf. Electron. Packag. Technol.*, 2005, pp. 17–24.
[10] G. Q. Zhang, M. Graef, and F. van Roosmalen, "The rationale and paradigm of 'more than Moore,"' in *Proc. 56th Electron. Compon. Technol. Conf.*, 2006, pp. 151–157.
[11] R. K. Cavin, P. Lugli, and V. V. Zhirnov, "Science and engineering beyond Moore's law," *Proc. IEEE*, vol. 100, pp. 1720–1749, May 2012.
[12] M. P. Sah, H. Kim, and L. O. Chua, "Brains are made of memristors," *IEEE Circuits Syst. Mag.*, vol. 14, no. 1, pp. 12–36, Feb. 2014.
[13] M. Crupi, L. Pradhan, and S. Tozer, "Modelling neural plasticity with memristors," *IEEE Can. Rev.*, vol. 68, pp. 10–14, 2012.
[14] C. Evans-Pughe, "The device with brains [electronics memristors]," *Eng. Technol.*, vol. 5, no. 12, pp. 30–33, Aug. 2010.
[15] R. S. Williams, "How we found the missing memristor," *IEEE Spectr.*, vol. 45, no. 12, pp. 28–35, Dec. 2008.
[16] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, no. 7191, pp. 80–83, May 2008.
[17] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, 1943.
[18] SI Association and others, "International technology roadmap for semiconductors," Semicond. Ind. Assoc., Tech. Rep., 2012.
[19] L. O. Chua, "Memristor—The missing circuit element," *IEEE Trans. Circuit Theory*, vol. CT-18, no. 5, pp. 507–519, Sep. 1971.
[20] L. O. Chua, "Synthesis of new nonlinear network elements," *Proc. IEEE*, vol. 56, no. 8, pp. 1325–1340, Aug. 1968.
[21] L. O. Chua, *Introduction to Nonlinear Network Theory*. New York, NY, USA: McGraw-Hill, 1969.
[22] L. O. Chua and S. M. Kang, "Memristive devices and systems," *Proc. IEEE*, vol. 64, no. 2, pp. 209–223, Feb. 1976.
[23] C. A. Richter, D. R. Stewart, D. A. A. Ohlberg, and R. S. Williams, "Electrical characterization of Al/AlO_x/molecule/Ti/Al devices," *Appl. Phys. A*, vol. 80, no. 6, pp. 1355–1362, 2005.
[24] J. J. Blackstock, W. F. Stickle, C. L. Donley, D. R. Stewart, and R. S. Williams, "Internal structure of a molecular junction device: Chemical reduction of PtO_2 by Ti evaporation onto an interceding organic monolayer," *J. Phys. Chem. C*, vol. 111, no. 1, pp. 16–20, 2007.
[25] P. Penfield and R. P. Rafuse, *Varactor Applications*. Cambridge, MA, USA: MIT Press, 1962.
[26] J. J. Yang, M. D. Pickett, X. Li, D. A. A. Ohlberg, D. R. Stewart, and R. S. Williams, "Memristive switching mechanism for metal/oxide/metal nanodevices," *Nature Nanotechnol.*, vol. 3, no. 7, pp. 429–433, 2008.
[27] T. Prodromakis, K. Michelakis, and C. Toumazou, "Fabrication and electrical characteristics of memristors with TiO_2/TiO_{2+x} active layers," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May/Jun. 2010, pp. 1520–1522.
[28] M. D. Pickett *et al.*, "Switching dynamics in titanium dioxide memristive devices," *J. Appl. Phys.*, vol. 106, no. 7, pp. 074508-1–074508-6, 2009.
[29] L. O. Chua, "Nonlinear circuit foundations for nanodevices. I. The four-element torus," *Proc. IEEE*, vol. 91, no. 11, pp. 1830–1859, Nov. 2003.
[30] Y. N. Joglekar and S. J. Wolf, "The elusive memristor: Properties of basic electrical circuits," *Eur. J. Phys.*, vol. 30, no. 4, p. 661, 2009.
[31] T. W. Hickmott, "Low-frequency negative resistance in thin anodic oxide films," *J. Appl. Phys.*, vol. 33, no. 9, pp. 2669–2682, 1962.
[32] R. Waser and M. Aono, "Nanoionics-based resistive switching memories," *Nature Mater.*, vol. 6, no. 11, pp. 833–840, Nov. 2007.
[33] N. B. Zhitenev, A. Sidorenko, D. M. Tennant, and R. A. Cirelli, "Chemical modification of the electronic conducting states in polymer nanodevices," *Nature Nanotechnol.*, vol. 2, no. 4, pp. 237–242, 2007.
[34] J. H. A. Smits, S. C. J. Meskers, R. A. J. Janssen, A. W. Marsman, and D. M. de Leeuw, "Electrically rewritable memory cells from poly(3-hexylthiophene) Schottky diodes," *Adv. Mater.*, vol. 17, no. 9, pp. 1169–1173, May 2005.
[35] Q. Lai, Z. Zhu, Y. Chen, S. Patil, and F. Wudl, "Organic nonvolatile memory by dopant-configurable polymer," *Appl. Phys. Lett.*, vol. 88, no. 13, p. 133515, 2006.
[36] K. Terabe, T. Hasegawa, T. Nakayama, and M. Aono, "Quantized conductance atomic switch," *Nature*, vol. 433, no. 7021, pp. 47–50, Nov. 2005.
[37] M. N. Kozicki, M. Park, and M. Mitkova, "Nanoscale memory elements based on solid-state electrolytes," *IEEE Trans. Nanotechnol.*, vol. 4, no. 3, pp. 331–338, May 2005.
[38] S. Dietrich *et al.*, "A nonvolatile 2-Mbit CBRAM memory core featuring advanced read and program control," *IEEE J. Solid-State Circuits*, vol. 42, no. 4, pp. 839–845, Apr. 2007.
[39] J. R. Jameson *et al.*, "Field-programmable rectification in rutile TiO_2 crystals," *Appl. Phys. Lett.*, vol. 91, no. 11, p. 112101, 2007.
[40] D. S. Jeong, H. Schroeder, and R. Waser, "Coexistence of bipolar and unipolar resistive switching behaviors in a Pt/TiO_2/Pt stack," *Electrochem. Solid-State Lett.*, vol. 10, no. 8, pp. G51–G53, 2007.
[41] A. Beck, J. G. Bednorz, C. Gerber, C. Rossel, and D. Widmer, "Reproducible switching effect in thin oxide films for memory applications," *Appl. Phys. Lett.*, vol. 77, no. 1, pp. 139–141, 2000.
[42] M. M. Ziegler and M. R. Stan, "Design and analysis of crossbar circuits for molecular nanoelectronics," in *Proc. 2nd IEEE Conf. Nanotechnol. (IEEE-NANO)*, Aug. 2002, pp. 323–327.
[43] D. B. Strukov and R. S. Williams, "Four-dimensional address topology for circuits with stacked multilayer crossbar arrays," *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 48, pp. 20155–20158, 2009.
[44] G. Liu, L. Fang, N. Li, B. C. Sui, and Z. K. Duan, "New behavioral modeling method for crossbar-based memristor," in *Proc. Asia Pacific Conf. Postgraduate Res. Microelectron. Electron. (PrimeAsia)*, Sep. 2010, pp. 356–359.
[45] J. Wu and M. Choi, "Memristor lookup table (MLUT)-based asynchronous nanowire crossbar architecture," in *Proc. 10th IEEE Conf. Nanotechnol. (IEEE-NANO)*, Aug. 2010, pp. 1100–1103.
[46] C. Xu, X. Dong, N. P. Jouppi, and Y. Xie, "Design implications of memristor-based RRAM cross-point structures," in *Proc. Design, Autom. Test Eur. Conf. Exhibit. (DATE)*, Mar. 2011, pp. 1–6.
[47] M. S. Qureshi, M. Pickett, F. Miao, and J. P. Strachan, "CMOS interface circuits for reading and writing memristor crossbar array," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2011, pp. 2954–2957.
[48] M. Bavandpour, S. B. Shouraki, H. Soleimani, A. Ahmadi, and A. A. Makhlooghpour, "Simulation of memristor crossbar structure on GPU platform," in *Proc. 20th Iranian Conf. Elect. Eng. (ICEE)*, May 2012, pp. 178–183.
[49] S. Pi, P. Lin, and Q. Xia, "Memristor crossbar arrays with junction areas towards sub-10×10 nm^2," in *Proc. 13th Int. Workshop Cellular Nanosc. Netw. Appl. (CNNA)*, Aug. 2012, pp. 1–2.
[50] Y. Cassuto, S. Kvatinsky, and E. Yaakobi, "Sneak-path constraints in memristor crossbar arrays," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2013, pp. 156–160.
[51] M. A. Zidan, A. Sultan, H. A. H. Fahmy, and K. N. Salama, "Leakage analysis of crossbar memristor arrays," in *Proc. IEEE 14th Int. Workshop Cellular Nanosc. Netw. Appl. (CNNA)*, Jul. 2014, pp. 1–2.
[52] E. Linn, "Memristive nano-crossbar arrays enabling novel computing paradigms," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Jun. 2014, pp. 2596–2599.
[53] A. Sawa, T. Fujii, M. Kawasaki, and Y. Tokura, "Interface resistance switching at a few nanometer thick perovskite manganite active layers," *Appl. Phys. Lett.*, vol. 88, no. 23, p. 232112, 2006.

[54] M. Hamaguchi, K. Aoyama, S. Asanuma, Y. Uesu, and T. Katsufuji, "Electric-field-induced resistance switching universally observed in transition-metal-oxide thin films," *Appl. Phys. Lett.*, vol. 88, no. 14, p. 142508, 2006.

[55] K. Szot, W. Speier, G. Bihlmayer, and R. Waser, "Switching the electrical resistance of individual dislocations in single-crystalline $SrTiO_3$," *Nature Mater.*, vol. 5, no. 4, pp. 312–320, 2006.

[56] G. Chen, "Leon Chua's memristor [Recognitions]," *IEEE Circuits Syst. Mag.*, vol. 8, no. 2, p. 55, Aug. 2008.

[57] J. Borghetti, G. S. Snider, P. J. Kuekes, J. J. Yang, D. R. Stewart, and R. S. Williams, "'Memristive' switches enable 'stateful' logic operations via material implication," *Nature*, vol. 464, no. 7290, pp. 873–876, 2010.

[58] B. Widrow, W. H. Pierce, and J. B. Angell, "Birth, life, and death in microelectronic systems," *IRE Trans. Military Electron.*, vol. 5, no. 3, pp. 191–201, Jul. 1961.

[59] R. R. Schaller, "Moore's law: Past, present and future," *IEEE Spectr.*, vol. 34, no. 6, pp. 52–59, Jun. 1997.

[60] "Types of memristors," Tech. Rep., 2015. [Online]. Available: http://www.memristor.org/reference/295/types-of-memristors

[61] R. Tetzlaff, Ed., *Memristors and Memristive Systems*. New York, NY, USA: Springer, 2014.

[62] P. Mazumder, S. M. Kang, and R. Waser, "Memristors: Devices, models, and applications [scanning the issue]," *Proc. IEEE*, vol. 100, no. 6, pp. 1911–1919, Jun. 2012.

[63] V. Beiu, J. M. Quintana, and M. J. Avedillo, "VLSI implementations of threshold logic—A comprehensive survey," *IEEE Trans. Neural Netw.*, vol. 14, no. 5, pp. 1217–1243, Sep. 2003.

[64] M. Minsky, "Neural nets and the brain-model problem," Ph.D. dissertation, Dept. Math., Princeton Univ., Princeton, NJ, USA, 1954.

[65] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychol. Rev.*, vol. 65, no. 6, pp. 386–408, 1958.

[66] T. L. E. Dake and U. Cilingiroglu, "Rank-order filter using multiple-winners-take-all," in *Proc. 43rd IEEE Midwest Symp. Circuits Syst.*, vol. 1. Aug. 2000, pp. 482–485.

[67] S. Slesazeck, A. Ascoli, H. Mähne, R. Tetzlaff, and T. Mikolajick, "Unfolding the threshold switching behavior of a memristor," in *Nonlinear Dynamics of Electronic Systems*. Springer, 2014, pp. 156–164.

[68] A. N. Morozovska *et al.*, "Nonlinear space charge dynamics in mixed ionic-electronic conductors: Resistive switching and ferroelectric-like hysteresis of electromechanical response," *J. Appl. Phys.*, vol. 116, no. 6, p. 066808, 2014.

[69] J. P. Strachan, D. B. Strukov, J. Borghetti, J. J. Yang, G. Medeiros-Ribeiro, and R. S. Williams, "The switching location of a bipolar memristor: Chemical, thermal and structural mapping," *Nanotechnology*, vol. 22, no. 25, p. 254015, 2011.

[70] J. Rajendran, H. Manem, and G. S. Rose, "NDR based threshold logic fabric with memristive synapses," in *Proc. 9th IEEE Conf. Nanotechnol. (IEEE-NANO)*, Jul. 2009, pp. 725–728.

[71] E. Goto *et al.*, "Esaki diode high-speed logical circuits," *IRE Trans. Electron. Comput.*, vol. EC-9, no. 1, pp. 25–29, Mar. 1960.

[72] J. Rajendran, H. Manem, R. Karri, and G. S. Rose, "Memristor based programmable threshold logic array," in *Proc. IEEE/ACM Int. Symp. Nanosc. Archit. (NANOARCH)*, Jun. 2010, pp. 5–10.

[73] J. Rajendran, H. Manem, R. Karri, and G. S. Rose, "An energy-efficient memristive threshold logic circuit," *IEEE Trans. Comput.*, vol. 61, no. 4, pp. 474–487, Apr. 2012.

[74] T. Tran, A. Rothenbuhler, E. H. B. Smith, V. Saxena, and K. A. Campbell, "Reconfigurable threshold logic gates using memristive devices," in *Proc. IEEE Subthreshold Microelectron. Conf. (SubVT)*, Oct. 2012, pp. 1–3.

[75] A. S. Oblea, A. Timilsina, D. Moore, and K. A. Campbell, "Silver chalcogenide based memristor devices," in *Proc. IJCNN*, Jul. 2010, pp. 1–3.

[76] H. Manem, J. Rajendran, and G. S. Rose, "Stochastic gradient descent inspired training technique for a CMOS/nano memristive trainable threshold gate array," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 5, pp. 1051–1060, May 2012.

[77] S. Muroga, *Threshold logic and Its Applications*. New York, NY, USA: Wiley, 1971.

[78] H. Manem and G. S. Rose, "A read-monitored write circuit for 1T1M multi-level memristor memories," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2011, pp. 2938–2941.

[79] H. Manem, J. Rajendran, and G. S. Rose, "Design considerations for multilevel CMOS/nano memristive memory," *ACM J. Emerg. Technol. Comput. Syst. (JETC)*, vol. 8, no. 1, 2012, Art. no. 6.

[80] G. Ligang, F. Alibart, and D. B. Strukov, "Programmable CMOS/memristor threshold logic," *IEEE Trans. Nanotechnol.*, vol. 12, no. 2, pp. 115–119, Mar. 2013.

[81] S. Kvatinsky, N. Wald, G. Satat, A. Kolodny, U. C. Weiser, and E. G. Friedman, "MRL—Memristor ratioed logic," in *Proc. 13th Int. Workshop Cellular Nanosc. Netw. Appl. (CNNA)*, Aug. 2012, pp. 1–6.

[82] G. Rose, J. Rajendran, H. Manem, R. Karri, and R. Pino, "Leveraging memristive systems in the construction of digital logic circuits," *Proc. IEEE*, vol. 100, no. 6, pp. 2033–2049, Jun. 2012.

[83] K. K. Likharev, "Hybrid CMOS/nanoelectronic circuits: Opportunities and challenges," *J. Nanoelectron. Optoelectron.*, vol. 3, no. 3, pp. 203–230, 2008.

[84] C. B. Dara, T. Haniotakis, and S. Tragoudas, "Low power and high speed current-mode memristor-based TLGs," in *Proc. IEEE Int. Symp. Defect Fault Tolerance VLSI Nanotechnol. Syst. (DFT)*, Oct. 2013, pp. 89–94.

[85] M. Sharad, D. Fan, and K. Roy. (2013). "Ultra-low energy, high-performance dynamic resistive threshold logic." [Online]. Available: http://arxiv.org/abs/1308.4672

[86] D. Chabi, W. Zhao, D. Querlioz, and J.-O. Klein, "Robust neural logic block (NLB) based on memristor crossbar array," in *Proc. IEEE/ACM Int. Symp. Nanosc. Archit. (NANOARCH)*, Jun. 2011, pp. 137–143.

[87] M. Soltiz, D. Kudithipudi, C. Merkel, G. S. Rose, and R. E. Pino, "Memristor-based neural logic blocks for nonlinearly separable functions," *IEEE Trans. Comput.*, vol. 62, no. 8, pp. 1597–1606, Aug. 2013.

[88] A. P. James, L. R. V. J. Francis, and D. S. Kumar, "Resistive threshold logic," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 1, pp. 190–195, Jan. 2014.

[89] A. P. James and S. Dimitrijev, "Cognitive memory network," *Electron. Lett.*, vol. 46, no. 10, pp. 677–678, May 2010.

[90] D. Fan, M. Sharad, and K. Roy, "Design and synthesis of ultralow energy spin-memristor threshold logic," *IEEE Trans. Nanotechnol.*, vol. 13, no. 3, pp. 574–583, May 2014.

[91] S. Fukami *et al.*, "Low-current perpendicular domain wall motion cell for scalable high-speed MRAM," in *Proc. Symp. VLSI Technol.*, Jun. 2009, pp. 230–231.

[92] C. Augustine *et al.*, "Numerical analysis of domain wall propagation for dense memory arrays," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2011, pp. 17.6.1–17.6.4.

[93] A. P. James, D. S. Kumar, and A. Ajayan, "Threshold logic computing: Memristive-CMOS circuits for fast fourier transform and vedic multiplication," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 11, pp. 2690–2694, Nov. 2015.

[94] D. Soudry, D. Di Castro, A. Gal, A. Kolodny, and S. Kvatinsky, "Memristor-based multilayer neural networks with online gradient descent training," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2408–2421, Oct. 2015.

[95] J. Yang, N. Kulkarni, S. Yu, and S. Vrudhula, "Integration of threshold logic gates with RRAM devices for energy efficient and robust operation," in *Proc. IEEE/ACM Int. Symp. Nanosc. Archit.*, Jul. 2014, pp. 39–44.

[96] N. S. Nukala, N. Kulkarni, and S. Vrudhula, "Spintronic threshold logic array (STLA)—A compact, low leakage, non-volatile gate array architecture," *J. Parallel Distrib. Comput.*, vol. 74, no. 6, pp. 2452–2460, 2014.

[97] A. K. Maan, D. S. Kumar, S. Sugathan, and A. P. James, "Memristive threshold logic circuit design of fast moving object detection," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 10, pp. 2337–2341, Oct. 2015.

[98] L. Goknar, "Models and modeling: Be careful and use your imagination," *IEEE Antennas Propag. Mag.*, vol. 50, no. 5, pp. 215–221, Oct. 2008.

Akshay Kumar Maan received the Ph.D. degree in memristor-based threshold logic cognitive circuits from the Griffith School of Engineering, Griffith University, Brisbane, QLD, Australia, in 2015. The work is inspired by the principle of firing of neurons in the primate brain, and the ability of neuronal structures to perform different cognitive tasks, exploring programmable (mem)resistance based threshold logic memory networks that are shown to perform basic logic functions to highly involved cognitive tasks, such as face recognition and fast moving object detection, in a parallel, scalable hardware architecture.

His current research interests include brain inspired circuits, algorithms and systems, to enable More-than-Moore era of low-power, and high-density VLSI circuits for cognitive applications, such as pattern recognition, face recognition, and speech recognition.

Deepthi Anirudhan Jayadevi (M'13) received the B.Tech. degree in electronics and instrumentation engineering from the Cochin University of Science and Technology, Kochi, India, in 2008, and the M.Tech. degree in very large scale integration (VLSI) and embedded systems from the ER&DCI Institute of Technology, Thiruvananthapuram, India, an Educational Institution under the direct control of the Centre for Development of Advanced Studies, Pune, India, in 2010.

She was a Research Staff Member with Enview Research and Development Laboratories, Thiruvananthapuram. She has over five years of experience in teaching and research in the field of electronics. Her current research interests include VLSI circuits and artificial intelligence.

Alex Pappachen James (SM'13) received the Ph.D. degree from the Queensland Micro and Nanotechnology Centre, Griffith University, Brisbane, QLD, Australia.

He is currently an Associate Professor of Electronics Engineering with Nazarbayev University, Astana, Kazakhstan, and leads the Bioinspired Microelectronic Systems Laboratory. His group is actively engaged in the development of memristive circuits and systems that can be integrated to realistic imaging and pattern recognition systems. His current research interests include brain inspired circuits, algorithms, and systems.