Memristor-Based In-Memory Computing Model For Energy Efficient Data Processing

Alexander Kotovis
Electrical Engineering,
Penn State Altoona,
3000 Ivyside Park, Altoona,
PA 16601

Dr. Sohail Anwar
Department of Engineering
Penn State Altoona,
3000 Ivyside Park, Altoona,
PA 16601

***Abstract*—Memristors are nanoscale resistive devices that can store and process information in the same place. They help address the von Neumann bottleneck. Because they perform analog computation with low energy, they are promising for in-memory and neuromorphic systems. This paper reviews recent progress in HfOx, TaOx, and OxRAM devices, as well as large-scale crossbar demonstrations for neural network inference. Results show that memristor arrays can efficiently perform matrix–vector multiplication, but face challenges like device variability, nonlinear switching, and conductance drift. Improving materials, calibration, and CMOS-compatible fabrication is important for making memristor-based computing reliable and scalable.**

***Keywords*— Memristor, von Neumann architecture, Resistive Switching, Neuromorphic Computing, In-Memory Computing**

1. Introduction

Traditional computers follow the von Neumann architecture. This is where memory and processors are separated, leading to bottlenecking. Energy and time are wasted moving data back and forth to compute, creating the need for new technology to solve this bottleneck. There are two main types of memory: volatile and non-volatile. Non-volatile memory (NVM) is useful because it retains data even when power is turned off, unlike volatile memory (like RAM), which loses all information when power is lost. This is commonly found in devices like SSDs, flash drives, and ROM. However, these types of devices can only read and/or write. The second component to computers is the processor, most commonly known as the CPU. The CPU strictly does computations; it takes input data, processes it according to a program's commands, and produces output. Memristors have gained notability for their ability to not just read/write but also compute directly on stored data (which is non-volatile), eliminating the "von Neumann bottleneck" [1], [2]. Because memristors operate at the nanoscale, billions can be integrated into a dense network that combines storage and computing, offering exceptional energy efficiency and speed [2], [7].

1.1 **Memristors**

A memristor is an electronic device whose resistance depends on the past electrical signals applied to it. The idea was first introduced by Leon Chua in 1971, who proposed a missing circuit element linking charge and magnetic flux [1]. This theoretical model showed that resistance could depend on the history of charge flow, allowing a device to "remember" its previous state.

In practical devices, this memory behavior comes from internal physical changes. When a voltage is applied, ions inside the material move, or conductive filaments grow and shrink. These changes modify the device's resistance, and the new state remains even after power is removed [3], [7]. Because the internal configuration is preserved, memristors are non-volatile and can store information as stable resistance levels.

Memristors are useful in systems where storage and computation needs to happen in the same location. Their ability to hold multiple resistance states and update those states with electrical pulses makes them suitable for applications such as non-volatile memory, in-memory computing, neuromorphic circuits, and artificial synapses in brain-inspired hardware [1], [6], [8].

1.2 **Current Progress in Memristors**

In the United States, a team at the University of Michigan demonstrated a HfOx-based 64×64 RRAM crossbar array that was capable of performing analog matrix–vector multiplications for MNIST image classification [4]. Their system achieved 94–97% accuracy after device-aware training and it operated at significantly lower energy than a digital CPU baseline [4]. Similarly, U.S. national laboratories reported TaOx and HfOx devices with stable analog conductance tuning and multi-level weights suitable for IMC accelerators [6].

In Europe, IMEC and research groups in the Netherlands and Germany have developed foundry-compatible RRAM arrays using materials and processes compatible with commercial CMOS fabrication [2], [3]. One demonstration used a 256×256 1T1R OxRAM array that reached 8–10-bit effective precision in analog MAC operations through write–verify programming and calibration, enabling deeper neural network inference on analog hardware [2]. European teams have also reported sub-picojoule programming energy and high device endurance that support scalable neuromorphic implementations [6].

In Asia, researchers at Tsinghua University were able to build a 128×64 TaOx memristor array for keyword spotting operating under 200 µW, making it suitable for battery-powered edge-AI devices [5]. Additionally, Japanese and South Korean groups have demonstrated memristor-based convolution operations and spike-timing-dependent plasticity (STDP), enabling compact analog neural networks for low-power pattern recognition tasks [7].

2. **Implementation**

Memristors are built using a metal–insulator–metal (MIM) structure. This usually includes a top electrode, an oxide switching layer, and a bottom electrode (Fig. 1. a-b). A voltage is then applied, and ions or oxygen vacancies move inside the oxide. This movement forms or breaks a conductive path, which causes a change in the resistance. The low-resistance state is called SET, and the high-resistance state is called RESET (Fig. 1.c). After switching, the resistance stays in its new state even with the power off; this is what makes memristors non-volatile [2], [3].

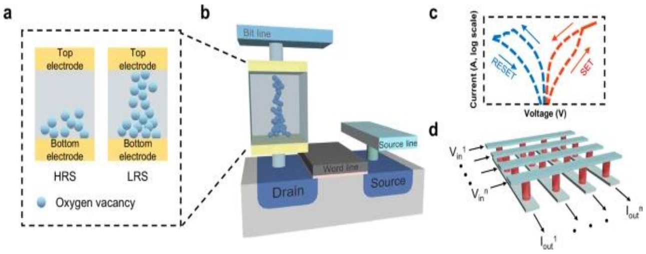

Figure 1. Memristor device structure, switching mechanism, I–V behavior, and crossbar operation [2].

(a) Oxygen-vacancy distribution in high-resistance (HRS) and low-resistance (LRS) states. (b) A memristor integrated with a transistor (1T1R structure). (c) Typical memristor I–V curve showing SET and RESET transitions. (d) Crossbar array)

The hysteresis loop in Fig. 1c shows the typical I–V response during switching. When many memristors are placed in a crossbar array (Fig. 1d), each one acts as a programmable conductance for matrix–vector multiplication [2].

Most practical systems use either a 1T1R or 1S1R structure. In a 1T1R, each memristor is paired with a transistor, making it common in in-memory computing (IMC) because it provides better control over programming and reduces sneak currents [4], [6]. A 1S1R structure uses a selector instead of a transistor; it's used in more compact designs where density is prioritized [3]. In both cases, devices are arranged into crossbar arrays so that each cell acts like a programmable conductance (Fig. 1d).

Implementations follow a similar design across different research groups. The University of Michigan used a 64×64 HfOx 1T1R array to perform analog matrix–vector multiplication for MNIST classification [4]. Their system relied on applying small voltage pulses to gradually tune each device to a target conductance. This closed-loop "write–verify" style programming is common because individual devices vary from one another [2], [3], [6]. IMEC's 256×256 OxRAM array used a similar approach, but with additional calibration steps to reach 8–10-bit effective precision for larger networks [2]. These steps adjust for nonlinear switching and drift so the array can maintain stable weights.

Other implementations focus on low-power edge computing. Tsinghua University was able to run a TaOx array running keyword spotting at under 200 µW by simplifying the readout circuits and reducing ADC resolution [5]. Groups in Japan and South Korea have used related structures for neuromorphic learning [7].

3. **Review and Discussion**

There are several shared patterns across previously discussed memristor-based systems. Most arrays use incremental programming, where small pulses gradually shift the conductance until it reaches a target value. This is usually paired with a write–verify step because individual devices vary and can over- or undershoot the intended state if programmed in a single pulse [2], [4], [6]. 1T1R structures remain the most reliable option for IMC applications since the transistor helps block unwanted currents and improve control during programming [3], [6].

Another common trend is the use of per-column ADCs to read analog currents from crossbars. Even though ADCs add overhead, they are necessary to convert the accumulated current into a digital value for further processing [2], [5]. Larger arrays and analog tasks often require calibration cycles to manage drift, nonlinearity, and device aging [3], [6]. These techniques appear consistent across different materials (HfOx, TaOx, OxRAM) and array sizes.

3.1 **Observed Strengths**

One of the main strengths of memristor arrays is that they naturally perform matrix–vector multiplication in one step; it is the core operation in neural networks [1], [2]. By computing inside the array, the need to shuttle data back and forth between memory and processor is greatly reduced.

Another strength is compatibility with CMOS build processes, especially in OxRAM arrays demonstrated by IMEC and other fabrication-focused groups [2], [3], [6]. This improves the likelihood of future commercial adoption because the devices can be integrated using established manufacturing steps.

3.2 **Limitations and Challenges**

Despite the advantages, challenges remain across all systems. Device variability being one of the most common issues. Conductance changes in different devices can happen due to identical programming pulses, which is why write–verify and calibration steps are needed [4], [6]. However, this adds time and complexity to the overall system.

Another limitation is nonlinear switching behavior; this makes it difficult to achieve high precision without closed-loop tuning [3]. Drift over time can shift conductance away from the desired value [6].

3.3 **Future Systems**

A hybrid approach may solve some challenges. Memristors can handle analog MAC operations while digital logic does control and corrections [2], [8]. This reduced ADC overhead and low-precision inference can also enable more efficient designs.

4. **Conclusion**

Based on current designs and data, memristors show promising results to significantly reduce the von Neumann bottleneck by combining storage and computation in the same location [1], [2]. Recent work shows that memristor arrays can perform core operations like matrix–vector multiplication with lower energy, making them useful for in-memory computing and neuromorphic tasks [4].

At the same time, several challenges remain. Variability, nonlinear switching, drift, and the need for calibration all affect accuracy and reliability [3], [6]. Large arrays also face issues with sneak currents and circuit overhead. These must be addressed before memristor technologies can be used at a commercial scale.

Progress across materials, programming schemes, and CMOS-compatible fabrication is validation for continuation of this research in the future of memristors and their development.

References:

[1]. Zhou H, Li S, Ang KW, Zhang YW. Recent Advances in In-Memory Computing: Exploring Memristor and Memtransistor Arrays with 2D Materials. Nanomicro Lett. 2024 Feb 19;16(1):121. doi: 10.1007/s40820-024-01335-2. PMID: 38372805; PMCID: PMC10876512.

[2]. Ding, C., Ren, Y., Liu, Z. et al. Transforming memristor noises into computational innovations. Commun Mater 6, 149 (2025). https://doi.org/10.1038/s43246-025-00876-2

[3]. Xiao, Y., Jiang, B., Zhang, Z., Ke, S., Jin, Y., Wen, X., & Ye, C. (2023). A review of memristor: material and structure design, device performance, applications and prospects. Science and Technology of Advanced Materials, 24(1). https://doi.org/10.1080/14686996.2022.2162323

[4]. Aguirre, F., Sebastian, A., Le Gallo, M. et al. Hardware implementation of memristor-based artificial neural networks. Nat Commun 15, 1974 (2024). https://doi.org/10.1038/s41467-024-45670-9

[5]. Kaiyun Gou, Yanran Li, Honglin Song, Rong Lu, Jie Jiang, Optimization strategy of the emerging memristors: From material preparation to device applications, iScience, Volume 27, Issue 12, 2024, 111327, ISSN 2589-0042, https://doi.org/10.1016/j.isci.2024.111327.

[6]. Isah, A.; Bilbault, J.-M. Review on the Basic Circuit Elements and Memristor Interpretation: Analysis, Technology and Applications. J. Low Power Electron. Appl. 2022, 12, 44. https://doi.org/10.3390/jlpea12030044

[7]. W. Chen, L. Song, S. Wang, Z. Zhang, G. Wang, G. Hu, S. Gao, Essential Characteristics of Memristors for Neuromorphic Computing. Adv. Electron. Mater. 2023, 9, 2200833. https://doi.org/10.1002/aelm.202200833

[8]. Zarepour A, Khosravi A, Iravani S, Zarrabi A. Biohybrid Micro/Nanorobots: Pioneering the Next Generation of Medical Technology. Adv Healthc Mater. 2024 Dec;13(31):e2402102. doi: 10.1002/adhm.202402102. Epub 2024 Oct 7. PMID: 39373299; PMCID: PMC11650542.

[9]. Hasan, R., Taha, T.M. and Alam, M.S. (2025), Near-Sensor 2D Pattern Classification System Based on 3D Memristor Array. Electron. Lett., 61: e70192. https://doi.org/10.1049/ell2.70192

[10]. Bettayeb, Meriem & Halawani, Yasmin & Khan, Muhammad Umair & Mohammad, Baker & Saleh, Hani. (2023). Memristor-Based In-Memory Computing. 10.1007/978-3-031-34233-2_6.