Mixed-Signal Computing with Non-Volatile Memories

Zahra Fahimi, M. Reza Mahmoodi, Mohammad Bavandpour, and Dmitri Strukov
ECE Department, UC Santa Barbara
Santa Barbara, CA 93106-5630 USA
{zfahimi, mrmahmoodi, mbavandpour, strukov}@ucsb.edu

ABSTRACT

In this paper, we review current-mode and time-domain mixed-signal implementations of vector-by-matrix multipliers (VMMs), based on floating-gate transistor and resistive random-access memories, which are two prominent classes of analog nonvolatile memories.

1. Introduction

VMM is a very common operation in scientific computing, signal processing, and machine learning algorithms. Low-to-medium precision VMM is by far the most critical operation in machine learning inference, which heavily dominates todays applications of neuromorphic computing, and will be crucial for other neuromorphic computing tasks, e.g. for feedforward propagation during training, and emerging neural models, such as spiking neural networks [1]. The need for fast and energy-efficient VMMs circuits will become much more acute with advent of internet of things and edge computing.

The best results for low-to-medium precision VMM circuits were reported for analog and mixed-signal implementations, see, e.g. original results in Refs. [2-18] and also reviews in Refs. [19-22]. The use of analog computing is in part motivated by extreme energy efficiency of biological neural networks, which perform similar low-precision operations for information processing. The rapidly maturing analog nonvolatile memories (NVMs) [23], which are used to store matrix weights, have greatly renewed interest in analog-domain VMM circuits. The purpose of this paper is to provide a brief review for some of the most promising approaches for analog and mixed-signal VMM implementations.

2. Mixed-Signal VMMs

The general architecture of mixed-signal VMMs based on NVMs is depicted in Fig. 1. Here, vector-by-matrix multiplication is defined as $y = Wx$, where $x \in \mathbb{R}^N$ is the input vector, $y \in \mathbb{R}^M$ is the output vector, and W is the weight matrix. N and M are the number of inputs and outputs, respectively.

The input vector is presented to the digital-to-analog converters (DACs), which convert the digital input to analog signals. The predetermined weight vector is encoded to the conductance of the NVM cells. NVM cell at each crosspoint generates current proportional to the amplitude of the input signal times its conductance. As a result, the multiplication operations are performed in parallel using Ohm's law. The current in all columns (bit lines) are summed up based on Kirchhoff's law and sensed by the peripheral circuits (PC), which is then followed by analog-to-digital converters (ADCs).

3. Analog Nonvolatile Memories for Computing

The most important memory device characteristics in the context of analog VMM circuits are cell size and its scalability, analog properties (linearity, noise, etc.), analog-grade retention, compatibility with conventional semiconductor technology, and large scale integration. The switching endurance, on the other hand, maybe rather low, because of matrix weights are changed infrequently in many applications, such as inference task in machine learning.

Resistive random access memories (RRAM), which are also called memristors, NOR flash and phase-change memories have been typically considered for analog computing [23]. The first two classes of memories have shown arguably the most promising results, and are discussed next.

3.1 Flash Memory

The idea of using floating-gate transistors to implement programmable analog VMMs was proposed more than a two decades ago. The most common are "synaptic transistors", which were fabricated in common CMOS foundries [2-4]. Their main problem is large cell footprint [19].

Commercial nonvolatile floating-gate memory cells, on the other hand, have been highly optimized and scaled down and may be embedded into CMOS integrated circuits. These cells are quite suitable to serve as adjustable synapse in neuromorphic network, provided their memory array wiring is modified to allow tuning of individual cells. Such modification was performed for the 180-nm ESF1 [24] and the 55-nm ESF3 [13] embedded commercial NOR flash memory technology of SST Inc., with good prospect for its scaling down to at least F = 28 nm. Redesigned memory features large programming dynamic range. Typically, >7-bit programming accuracy is achievable with enough number of pulses [25].

SONOS structure devices have been also employed for this analog computing, though the main issue with this technology is inferior analog-grade retention [11].

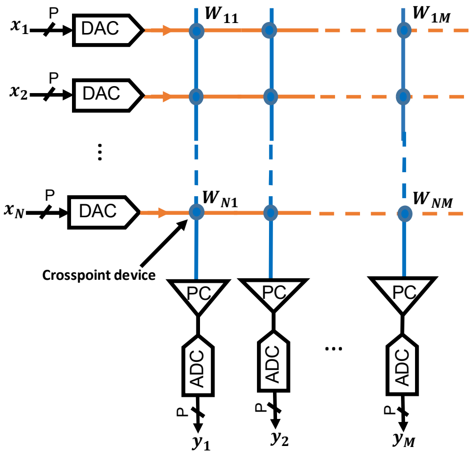

Fig.1 General architecture of a NVM-based VMM, with nonvolatile memory at each crosspoint.

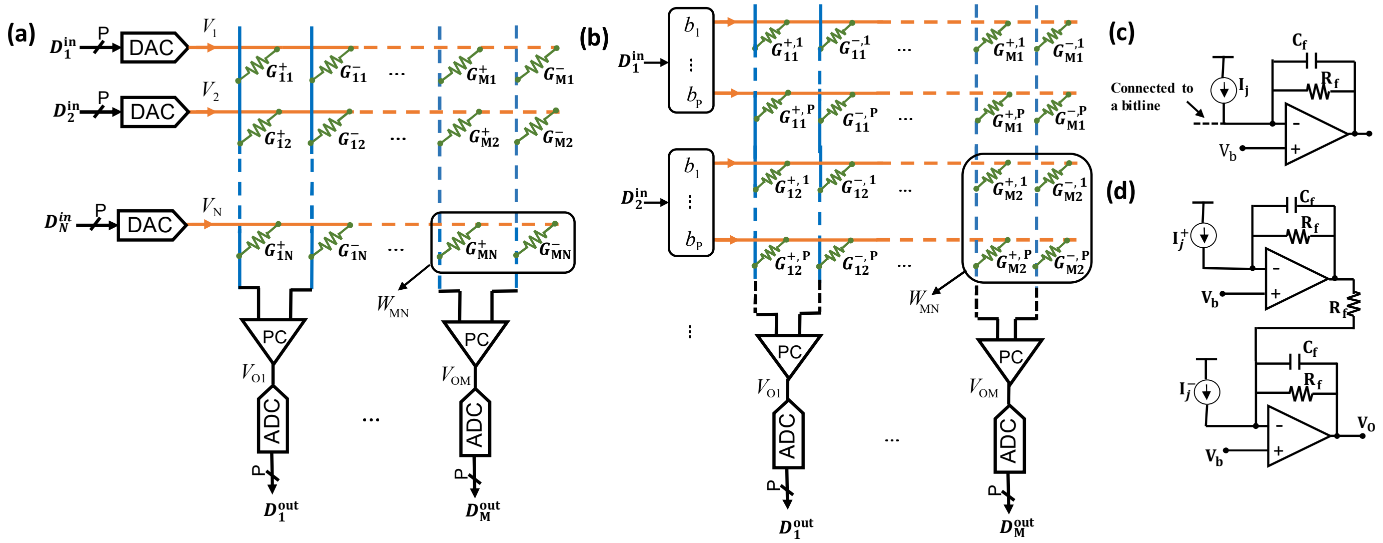

Fig.2 Current-mode RRAM-based VMMs: (a) Common differential structure with external DACs, (b) merged-DAC VMM, (c) TIA for current sensing, (d) differential current-mirror peripheral circuit.

3.2 RRAM

RRAM is rapidly maturing device technology, which has been already integrated in some CMOS foundries [1]. Such devices can be engineered to have lower programing energy, higher endurance and faster switching, as compared to those of floating gate memories. The most important feature of RRAM technology is, however, extremely compact device footprint, which is $4F^2$ for passive devices, which could be further reduced by integrating multiple passive crossbars vertically [26, 27]. RRAM's cell footprint is least 25 times denser compared to that of NOR flash memory cells, fabricated in the same technology.

4. VMM Architectures

Two-quadrant (2Q) VMMs, often utilized in DNNs, are multipliers with unipolar input and bipolar weights. Since conductance is inherently positive, differential-weight topologies are typically employed, in which a single weight is implemented with two memory cells. Hence, an $N\times M$ VMM performs $2(N\times M)$ operations per input. We assume the resolution of data converters is P and the weight precision is P_W. The input vector is $[D_1^{in}, D_j^{in} \ldots D_N^{in}]$, and its j^{th} element can be represented by a P-bit binary number, whose digits are denoted by $b_{k,j}$, where $1\leq k\leq P$.

4.1 RRAM-Based VMMs

Metal-oxide RRAM-based NVMs have been already commercialized for digital memory applications. Hence, binary-VMM designs based on digital memory macros are quite common (e.g., see Ref. [18]). However, the most prominent implementation of low-to-medium resolution 2Q VMMs are based on truly analog memory devices.

4.1.1 Current-Mode RRAM-Based VMM

The current-mode implementation (Fig. 2a) is particularly suitable for high-speed and higher resolution VMMs. In this approach, using a voltage-mode DAC, each multi-bit input signal is encoded as voltage $V_j = \sum_{k=0}^{p-1} b_{k,j}\left(\frac{V_{FS}}{2^P}\right)2^k$, where V_{FS} is the full-scale voltage and $b_{k,j}$ is the k^{th} bit of j^{th} input. The analog voltages are then applied to the word lines (horizontal lines in Fig. 2a). The normalized weight value, W_{ij}, is mapped to its corresponding analog conductance values, $G_{ij}^{\pm}$. The current flowing in each device is $I_{ij}^{\pm} = G_{ij}^{\pm} V_j$, and, hence, assuming PCs with gain K, the corresponding output voltage after subtraction is

$$V_{Oi} = K\sum_{j=1}^{N}\sum_{k=0}^{p-1} b_{k,j}\left(\frac{V_{FS}}{2^P}\right)2^k\left(G_{ij}^{+}-G_{ij}^{-}\right). \qquad (1)$$

Input conversion could be implemented by voltage-mode DACs, e.g., implemented by buffered R-2R ladder, binary-weighted design etc. [28]. However, such external DACs are often large and power hungry. In light of these deficiencies, merged-DAC structure is a viable solution (Fig. 2b). In fact, rewriting Eq. 1 yields:

$$V_{Oi} = K\sum_{j=1}^{N}\sum_{k=0}^{p-1} V_{FS}\, b_{k,j}\left[\left(\frac{G_{ij}^{+}2^k}{2^P}\right)-\left(\frac{G_{ij}^{-}2^k}{2^P}\right)\right].$$

Here, the conductance of memory cell is set to $G_{ij}^{\pm}2^{k-P}$.

In the simplest case, a trans-impedance amplifier (TIA) could be employed for PC or sensing circuit to ensure virtual ground on a bit line [6-10, 14, 27] (Fig. 2c). For a single-supply design (Fig. 2c), the virtual bias is set to V_b. For differential sensing, the current-mirror topology may be deployed (Fig. 2d).

4.1.2 Time-Mode RRAM-Based VMM

A time-mode VMM based on RRAM devices is shown in Fig. 3a. In this approach, input data are encoded into fixed-amplitude pulse widths and the entire computation is performed in time domain [29]. For compatibility with digital circuits, the time-domain circuits may also require digital-to-time converter (DTC) and analog-to-time converter (ATC). (Some complex computations may be performed efficiently completely in time domain [30].) The computation for each input is performed in a predetermined fixed period of time, T_{FS}. In a DTC, a multi-bit input signal is encoded in pulse width $T_j =$

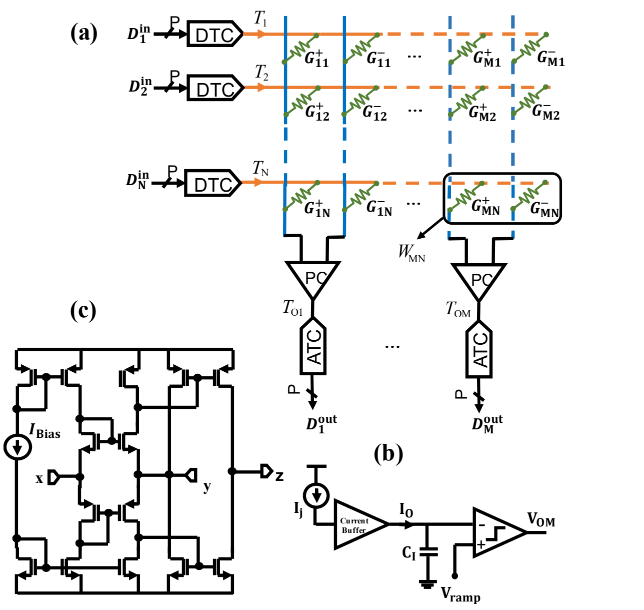

Fig. 3. Time-mode VMM design based on RRAM devices: (a) VMM architecture, (b) time-mode peripheral circuit, (c) current buffer, which was proposed in Ref. [5] for driving the capacitor.

$\sum_{k=0}^{p-1} b_{k,j}\left(\frac{T_{FS}}{2^P}\right)2^k$, where $b_{k,j}$ is the kth bit of jth input. Conductances are determined similar to the current-mode design. Assuming a current buffer with the current gain of K, the buffer's output current is $I_{Oi}^{\pm}(t) = K\sum_{j=1}^{N} G_{ij}^{\pm} V_j(t)$. This current is integrated on bit-line capacitor (Fig. 3b), so that the capacitor voltage is

$$V_{Oi}^{\pm}(t) = \frac{KV_{FS}}{C_I}\sum_{j=1}^{N}\left(G_{ij}^{\pm}\,T_j\right).$$

The result of the computation is encoded in time

$$T_{Oi}^{\pm} = \frac{KV_{FS}}{K'C_I}\sum_{j=1}^{N}\left(G_{ij}^{\pm}\,T_j\right),$$

which is a time at which $V_{Oi}(t)$ becomes equal to the ramp voltage $V_{ramp} = K't$, that is simultaneously applied to another input of the comparator (Fig. 3b). At time T_{FS}, the remaining charge on the capacitor is removed, and a new computation can be started within time period [T_{FS}, $2T_{FS}$].

A current buffer is generally required to suppress bit line voltage variations. In practice, designing a low-cost current buffer is challenging. In Ref. [15], a current conveyor was proposed. Conveyors are faster and much more energy efficient than TIAs. Compensation for offset in conveyors can be done efficiently with two additional word lines in the main array [17].

Additionally, a low-offset high-speed comparator is needed in the last stage of each channel to avoid loss of precision. It should be also noted that the output pulse can be easily converted back to digital using (a shared between all channels) binary counter and a digital accumulator [16].

4.2 Floating Gate Memory Based VMMs

Much of the earlier floating gate memory VMM implementations were based on synaptic transistors [2-4]. Ref. [12] reports on a two-layer neural network comprising of a 784×64 merged-DAC and 64×10 gate-coupled current-mode VMMs, which was the first mixed-signal experimentally tested network with such complexity. Fabricated in 180 nm eFlash CMOS process, the system achieved >10^3× better energy efficiency than 28 nm IBM TrueNorth digital chip for the same task at a similar fidelity.

All aforementioned designs are based on analog input/output VMMs which lack data converters. In Ref. [15], a complete VMM with digital interface is proposed. Simulation results showed up to 1.68 POps/J energy efficiency for a 5-bit 400×400 VMM designed in 55 nm eFlash CMOS process. In another recent work [16], the proposed time-based VMM with digital interface consumes 7 fJ to perform an operation for a 6-bit 200×200 VMM. In the following, we discuss these current-mode and time-domain architectures in more details.

4.2.1 Current-Mode Flash-Memory-Based VMM

Floating-gate transistors are typically biased in weak inversion to implement analog memory functionality. In this regime, the drain-source current is $I_{ds} \approx I_o e^{(V_g - V_{th})/nV_T} \equiv wI_o e^{V_g/nV_T}$, while the corresponding weight of single memory cell is defined as $w \equiv e^{-V_{th}/nV_T}$. (Here all the parameters have their usual meanings.)

In case of merged-DAC structure, i.e. digital-input topology, each device either conducts zero current (V_{WL} =0) or current $wI_o e^{V_{FS}/nV_T}$ (Fig. 4a). Due to the area overhead of additional devices, merged-DAC topology is a suitable candidate for only relatively small-scale VMMs.

The analog-input approach (Fig. 4b) is based on current mirror circuit in which a peripheral column of devices is utilized to sink the currents from an external DAC. Assuming the states of a peripheral device and a gate-coupled device in the main array are w_P and w_a, the gate voltage is given by $V_{cg} = nV_T\ln(I_{in}/w_P I_o)$. The mirrored device sinks the current $w_a I_o\, e^{V_{cg}/nV_T} = (w_a/w_P)I_{in}$. Thus, the effective weight is $W = (w_a/w_P)$, which can be adjusted by tuning the amount of charge on the transistor's floating gate. The total current in each channel is given by $I_{Oi} = \sum_{j=1}^{N}\sum_{k=0}^{p-1} b_{k,j}\left(\frac{I_{FS}}{2^P}\right)2^k\left(W_{ij}^{+} - W_{ij}^{-}\right)$ where I_{FS} is the maximum full-scale current of the external DAC.

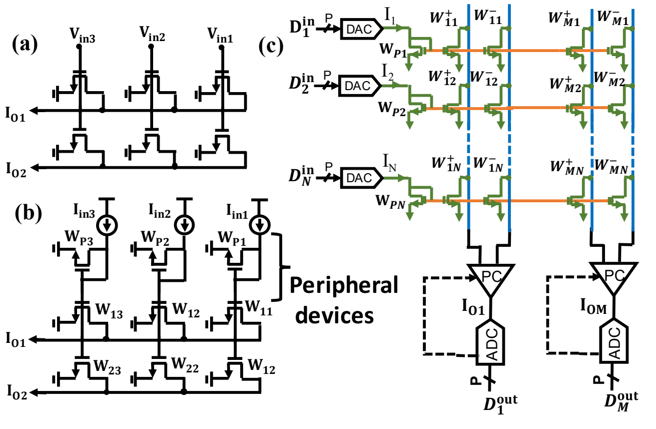

Fig. 4 Flash-memory based VMM design: (a) digital-input VMM circuit, (b) gate-coupled VMM circuit, (c) fully current-mode VMM circuit with external DACs and algorithmic ADCs.

All-current-mode VMM is shown in Fig. 4c. Current steering DAC [15] is a viable choice for input conversion. I_{FS} should be designed in such a way that the pole associated with the capacitive load in a shared word line gate is well-above the desired operational frequency. For VMMs with large gate-terminal parasitics, the multi-peripheral design technique introduced in Ref. [17] could be also useful.

Similar to the RRAM case, a TIA and a flash ADC can be used as PC and back-end data converter. By providing a virtual bias on a bit line, a resistive feedback amplifier followed by a flash ADC may be utilized to read current, generate proportional voltage and perform conversion (Fig. 2d). However, in comparison with RRAM, floating-gate memory devices operating in weak inversion show much higher output conductance.

4.2.2 Time-Mode Flash-Based VMM

The idea of pulsed-width encoding discussed in previous section can be employed in flash memory arrays as well. In fact, designing time-mode VMMs based on flash technology is more promising since the current buffer (Fig. 3c), which had a significant overhead, could be removed. The voltage swing on bit line (connected to output capacitors) defines the multiplication precision. Ref. [16] proposes very compact readout circuitry, essentially just an SR latch, which is used to generate the final pulse-modulated output. The remaining circuitry (TDC and DTC) could be similar to that discussed in previous section.

5. Summary

We reviewed mixed-signal implementations of vector-by-matrix multipliers based on nonvolatile memories, specifically focusing on current and time mode approaches using floating gate and resistive switching memories. The previous results show that energy efficiency, speed, and density of such VMM circuits could greatly exceed those of their digital counterpart at low-to-medium precision [31, 32]. As a result, we expect that mixed-signal implementations would be increasingly more adopted in various applications that heavily rely on VMM operation, including scientific computing, signal processing, and, most importantly, machine learning.

6. References

[1] H. Y. Tsai *et al.*, "Recent progress in analog memory-based accelerators for deep learning", *Journal of Physics D: Applied Physics*, vol. 51, art. 283001, 2018.

[2] C. R. Schlottmann and P. E. Hasler, "A highly dense, low power, programmable analog vector-matrix multiplier: The FPAA implementation", *IEEE JETCAS*, vol. 1 (3), pp. 403-411, 2011.

[3] S. Ramakrishnan and J. Hasler, "Vector-matrix multiply and winner-take-all as an analog classifier", *IEEE TVLSI Systems*, vol. 22 (2), pp. 353-361, 2014.

[4] R. Chawla *et al.*, "A 531 nW/MHz, 128×32 current-mode programmable analog vector-matrix multiplier with over two decades of linearity", in: *Proc. CICC'04*, Orlando, FL, Oct. 2004, pp. 651-654.

[5] M. J. Marinella *et al.*, "Multiscale co-design analysis of energy, latency, area, and accuracy of a ReRAM analog neural training accelerator", *IEEE JETCAS*, vol. 8 (1), pp. 86-101, 2018.

[6] M. Hu *et al.*, "Dot-product engine for neuromorphic computing: programming 1T1M crossbar to accelerate matrix-vector multiplication", in: *Proc. DAC'16,* Austin, TX, Jun. 2016, art. 19.

[7] C. Li *et al.*, "Large memristor crossbars for analog computing", in: *Proc. ISCAS'18*, Florence, Italy, May 2018, pp. 1-4.

[8] M. Hu *et al.*, "Dot-product engine as computing memory to accelerate machine learning algorithms", in: *Proc. ISQED'16*, Santa Clara, CA, Mar. 2016, pp. 374-379.

[9] C. Liu *et al.*, "Rescuing memristor-based neuromorphic design with high defects", in: *Proc. DAC'17*, Austin, TX, Jun. 2017, art. 87.

[10] C. Li *et al.*, "In-memory computing with memristor arrays", in: Proc. *IMW'18*, Kioto, Japan, May 2018, pp. 1-4.

[11] L. Fick *et al.*, "Analog in-memory subthreshold deep neural network accelerator", in: *Proc. CICC'17,* Austin, TX, Apr. 2017*,* pp. 1-4.

[12] X. Guo *et al.*, "Fast, energy-efficient, robust, and reproducible mixed-signal neuromorphic classifier based on embedded NOR flash memory technology', in: *Proc. IEDM'17,* San Francisco, CA, Dec. 2017, pp. 6.5.1– 6.5.4.

[13] X. Guo *et al.*, "Temperature-insensitive analog vector-by-matrix multiplier based on 55 nm NOR flash memory cells", in: *Proc. CICC'17*, Austin, TX, Apr.-May 2017, pp. 1-4.

[14] F. Merikh Bayat *et al.*, "Memristor-based perceptron classifier: Increasing complexity and coping with imperfect hardware", in: *Proc. ICCAD'17*, Irvine, CA, Nov. 2017, pp. 549-554.

[15] M. R. Mahmoodi and D. Strukov, "An ultra-low energy internally analog, externally digital vector-matrix multiplier based on NOR flash memory technology", in: *Proc. DAC'18*, Austin, TX, Jun. 2018, art. 22.

[16] M. Bavandpour *et al.*, "Energy-efficient time-domain vector-by-matrix multiplier for neurocomputing and beyond", *arXiv:1711.10673*, 2017.

[17] M. R. Mahmoodi and D. Strukov, "Breaking POp/J barrier with analog multiplier circuits based on nonvolatile memories", in: *Proc.ISLPED'18,* Bellevue, WA*,* Jul. 2018 (accepted).

[18] S. Yu *et al.*, "Binary neural network with 16 Mb RRAM macro chip for classification and online training", in: *Proc. IEDM'16,* San Francisco, CA, Dec. 2016, pp. 16.2.1-16.2.4.

[19] J. Hasler and H. B. Marr, "Finding a roadmap to achieve large neuromorphic hardware systems", *Frontiers in Neuroscience*, vol. 7, art. 118, 2013.

[20] F. Merrikh Bayat *et al.*, "Memory technologies for neural networks", in: *Proc. IMW'15*, Monterey, CA, May 2015, pp. 1-4.

[21] L. Ceze *et al.*, "Nanoelectronic neurocomputing: Status and prospects", in: *Proc. DRC'16*, Newark, DE, Jun. 2016, pp. 1-2.

[22] M. R. Mahmoodi and D. B. Strukov, "Mixed-signal POp/J computing with nonvolatile memories", in: *Proc. GLSVLSI'18*, Chicago, IL, May 2018, pp. 513-514.

[23] A. Chen, "A review of emerging non-volatile memory (NVM) technologies and applications", *Solid-State Electronics*, vol. 125, pp. 25-38, 2016.

[24] F. Merikh Bayat *et al.*, "Redesigning commercial floating-gate memory for analog computing applications", in: *Proc. ISCAS'15,* Lisbon, Portugal, May *2015*, pp. 1921-1924.

[25] F. Merikh Bayat *et al.*, "Model-based high-precision tuning of NOR flash memory cells for analog computing applications", in: *Proc. DRC'16*, Newark, DE, June 2016, pp. 1-2.

[26] G. Adam *et al.*, "Highly-uniform multi-layer ReRAM crossbar circuits", in: *Proc. ESSDERC'16*, Lausanne, Switzerland, Sept. 2016, pp. 436-439.

[27] B. Chakrabarti *et al.*, "A multiply-add engine with monolithically integrated 3D memristor crossbar/CMOS hybrid circuit", *Scientific Reports*, vol. 7, art. 42429, 2017.

[28] B. Razavi, *Principles of Data Conversion System Design*, vol. 126, New York: IEEE press, 1995.

[29] R. D'Angelo *et al.*, "A time-mode translinear principle for nonlinear analog computation", *IEEE TCAS-II*, vol. 62 (9), pp. 2187-2195, 2015.

[30] A. Madhavan *et al.*, "A 4-mm^2 180-nm-CMOS 15-Giga-cell-updates-per-second DNA sequence alignment engine based on asynchronous race conditions", in: *Proc. CICC'17*, Austin, TX, May 2017, pp. 1-4.

[31] Y. H. Chen *et al.*, "Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks", *IEEE Journal of Solid-State Circuits*, vol. 52 (1), pp. 127-138, 2017.

[32] B. Moons *et al.*, "Envision: A 0.26-to-10 TOps/W subword-parallel dynamic-voltage-accuracy-frequency-scalable convolutional neural network processor in 28 nm FDSOI", in: *Proc. ISSCC'17*, San Francisco, CA, Feb. 2017, pp. 246-247.