Efficient Mixed-Signal Neurocomputing via Successive Integration and Re-scaling

Mohammad Bavandpour*, Shubham Sahay*, *Member, IEEE,* Mohammad R. Mahmoodi, and Dmitri Strukov, *Senior Member, IEEE*

***Abstract*— The widespread and ever-increasing demand for performing in-situ inference, signal processing and other computationally intensive applications in mobile IoT devices requires fast, compact and energy-efficient Vector-by-Matrix Multipliers (VMM). The time-domain VMMs based on emerging non-volatile memory devices exhibit significantly higher circuit density and energy efficiency than their current-mode counterparts. However, load capacitors used to accumulate the weighted summation of the inputs in the time-domain-based circuits, dominate their energy dissipation and footprint area. The true potential of the time-domain-based VMMs may be realized only when this overhead is minimized. To this end, in this brief, we propose a novel Successive Integration and Re-scaling (SIR) approach for implementing a highly efficient mixed-signal time-domain VMM for low-to-medium-precision computing. For a proof of the concept, we quantitatively evaluated performance of the proposed SIR VMM, and compared it with the results for conventional time-domain VMM, using a similar 1T-1R array. Preliminary simulation results for the 4-bit 200×200 VMM, implemented using 55-nm technology node, show area and energy efficiencies of 1.33 bits/µm^2 and ~1.3 POp/J – the numbers, respectively, ~2.5× and ~2.65× higher than those for the prior-work time-domain VMM. Furthermore, we analyze the system-level performance of the proposed SIR VMM engine in the neuromorphic accelerator architectures and provide preliminary estimates for various Deep/Reccurent Neural Network (DNN/RNN) applications.**

***Index Terms*— Mixed-Signal Computing, Time-Domain Computing, Vector-by-Matrix Multiplier, Successive Integration and Re-scaling Technique, 1T1R Memory, ReRAM, Memristor**

I. Introduction

Weighted summation of the inputs tends to be the dominant operation in the widely used bio-inspired feedforward/recurrent [1]-[3] and more advanced, spiking [3]-[4] neural networks and many other signal processing algorithms. The demand for implementing these resource-intensive applications on mobile devices in our age of big data and internet-of-things (IoT) calls for efficient hardware realization of vector-by-matrix multipliers (VMM) [5]-[22]. Owing to the sparse design (off-chip synaptic weight storage) and frequent memory access, even the most advanced digital implementations of VMM engines are extremely power hungry [6]. The most promising solutions for implementation of energy-efficient VMMs are arguably based on emerging non-volatile memory devices [6], [8], [10]-[12], and [22]. Such mixed-signal VMMs demonstrate remarkable efficiency due to their natural ability to perform in-memory multiply-and-add operation using Ohm's and Kirchoff's laws, as well as their inherent massive parallelism, performing all such operations over the entire weight array in a single cycle.

Unfortunately, the circuit density and energy efficiency of the most prospective current-mode VMMs based on resistive-switching devices are so far limited because of relatively high cell currents, which results in higher overhead of the Input/Output (I/O) peripheral circuitry [10]-[11]. In principle, the lower currents and much higher input/output array impedances offered by the modified NOR flash array [10] allows a drastic reduction in the peripheral overhead of the current-mode VMM circuits based on embedded NOR flash [23]. However, the limited feature-size and scaling prospects of the floating-gate memory devices restricts the circuit density of these VMM implementations [10]. Similarly, inferior density is one of the major disadvantages of the mixed-signal VMMs based on switch-capacitor approach [9]-[10].

In order to address this issue, time-domain VMMs with passive and digital peripheral I/O circuits were proposed and explored [13]-[22]. However, the linear mapping of input signals on the input pulse duration, used in this architecture, necessitates the use of a large load capacitor to integrate the weighted sum of partial components of the signal at the output. Such large capacitor dominates the area and energy landscape in the existing time-domain VMMs, with its domination only growing as the VMM size is increased. Additionally, the VMM latency grows exponentially with input/computing precision in the originally proposed architecture. Therefore, to realize the potential of the time-domain VMM approach, the load capacitor should be minimized and different encoding scheme must be utilized.

In this brief, we address this challenge by proposing a novel Successive Integration and Re-scaling (SIR) approach, which does that while preserving the inherent advantages of the time-domain architecture, such as a low I/O peripheral circuitry overhead. In the proposed approach, the individual bits in the digital input are encoded by binary pulses of fixed amplitude and duration - unlike their encoding by the duration of fixed-amplitude pulses in the conventional time-domain VMMs. After each binary pulse, the electric charge accumulated on the load capacitor is divided via the charge-sharing mechanism, yielding an area- and power-efficient VMM.

II. Successive Integration and Re-scaling Approach

A. General Idea

An N×M VMM operation may be represented in a compact matrix form of:

$$\boldsymbol{Y}^{N\times 1} = \boldsymbol{W}^{N\times M}\boldsymbol{X}^{M\times 1}, \qquad (1)$$

where the individual elements $y_j \in \boldsymbol{Y}$, $x_i \in \boldsymbol{X}$ and $w_{ij} \in \boldsymbol{W}$ are related as:

$$y_j = \sum_{i=1}^{M} x_i w_{ij} \qquad (2)$$

Using the binary representation of P-bit inputs x_i as: $x_i(P\text{-}1)\ldots\ldots x_i(1)x_i(0)$, we can express the generalized P-bit dot-

This paper was submitted for review on July 7, 2019.

All authors are with the department of electrical and computer engineering, University of California at Santa Barbara, CA, USA, 93106 (e-mails: {mbavandpour, shubhamsahay, mrmahmoodi, strukov}@ ucsb.edu).

*These authors contributed equally.

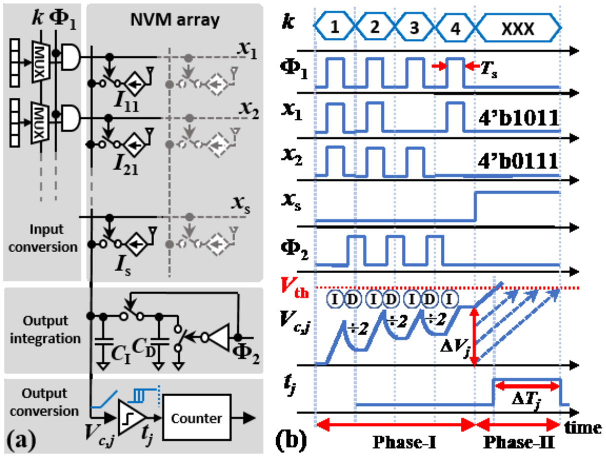

Fig. 1. (a) SIR-VMM structure, and (b) an example of the time diagram of VMM operation (on the example of the 4-bit precision).

product operation with respect to the input bits as:

$$y_j = \sum_{k=0}^{P-1} 2^k \sum_{i=1}^{M} x_i(k) w_{ij}, \tag{3}$$

where $x_i(k)$ is the kth bit (with $k = 0$ corresponding to the LSB) of the ith digital input. A re-scaled version of this dot-product, $y_j' = 2^{-P} y_j$ can be easily obtained in an iterative manner by initializing $y_{j,-1} = 0$ and then successively computing the weighted partial sum ($y_{j,k}$) for the kth input bit $x_i(k)$ (starting from the least significant bit), and adding the result to half the value of the previous sum:

$$y_{j,k} = \sum_{i=1}^{M} x_i(k) w_{ij} + \frac{1}{2} y_{j,k-1}. \tag{4}$$

After P-1 successive bit-wise dot-product calculation ("integration") and re-scaling ("division") operations, we get $y_{j,P\text{-}1} = y_j'$.

This representation is the basis for the proposed SIR-VMM. The overall structure of the VMM and its operation scheme are shown in Figs. 1(a) and 1(b), respectively. In the proposed VMM architecture, programmable embedded nonvolatile memory devices serve as the cross-point current sources. The VMM operation is performed in two phases. During Phase I, P successive integration (I) and re-scaling (D) operations are performed. At the kth step of this phase, the kth bits of all inputs are selected, using the signal k (see Fig. 1a), passed via the select line to the array of P×1 input Multiplexers (MUXs). If the kth bit of the ith input, $x_i(k)$, equals 1, a digital pulse with duration of T_s is generated by a simple AND operation performed on the selected bit-location (k), and a shared input pulse Φ_1. This input pulse enables the current source with a current proportional to the magnitude of the weight, $I_{ij} \propto w_{ij}$ (with i =1:N) on that row for the fixed duration of T_s. The integrating capacitor (C_I) remains disconnected from the dividing capacitor (C_D) during this period, and accumulates the charge carried by currents from the programmable nonvolatile memories. As a result, during this phase the voltage on C_I changes by

$$dv_{j,k} = \frac{T_S}{C_I} \sum_{i=1}^{M} x_i(k) I_{ij}, \tag{5}$$

While C_I integrates the charge, C_D is kept grounded through a pass-gate. Once the integration of each single-bit component (besides the Pth bit) has been completed, this capacitor, with $C_D = C_I$, is connected to C_I via a pass gate. The resulting charge sharing re-scales the voltage by the required factor of 2. After P-1 iterations of such integration and re-scaling operations, the change in voltage across C_I becomes proportional to the dot-product y_j:

$$\Delta V_j = 2^{-(P-1)} \frac{T_S}{C_I} \sum_{k=0}^{P-1} 2^k \sum_{i=1}^{M} x_i(k) I_{ij} \propto y_j \tag{6}$$

In Phase II, this dot-product output is converted to digital domain by a counter-based converter similar to the conventional time-domain VMM approach [15]: C_I is charged with a constant current source (I_s) until the capacitor voltage reaches the threshold voltage V_{th}. At this instance, a pulse with a rising edge is generated at the output of the threshold circuit, enabling a counter to count the time from the threshold-crossing time to a reference time point (ΔT_j).

For the total swing of voltage across the capacitor C_I to have a desired value ΔV_0, at the given bit precision P, minimum pulse width T_s and maximum cell current I_{max}, its capacitance should be equal to

$$C_I = \frac{2 M I_{max} T_S}{\Delta V_0} \left[1 - \left(\frac{1}{2}\right)^P\right]. \tag{7}$$

This value is proportional to $T_{Phase\text{-}I}$, which in turn scales as P, while in the conventional time-domain approach [15] and [22], $T_{phase\text{-}I}$ ($\propto 2^P$) should be large enough to accommodate 2^P quantized discrete intervals representing P-bit input information. This difference leads to a significant reduction of the required load capacitance in this SIR scheme.

B. SIR Multiplier Based on 1T-1R Memory Array

As a proof-of-concept, we have quantitatively evaluated the performance of the proposed SIR VMM and of the conventional time-domain VMM, based on a modified memory array of 1T-1R memory cells, each consisting of a memristor device connected at the source of the select transistor (MOSFET), rather than on the drain (as is the case for the conventional 1T-1R memory array) and acting as a programmable current sink to encode a particular analog weight [22] – see Fig. 1(a). Application of a pulse corresponding to each input bit activates the MOSFET, which acts like a switch. The MOSFET is operated in the sub-threshold mode and its source voltage is dictated by the resistance-state of the memristor. Therefore, by tuning the resistance of the memristor, the 1T-1R block can sink a current proportional to the resistance-state of the memristor.

Similar to [22], we have used a simple compact model for the memristor, $I = \frac{\beta}{R_0}\sinh(\beta V)$, where R_0 is the initial resistance-state, and β is the non-linearity factor. An initial ON-state resistance (R_0 = R_{ON}) of 1 MΩ and OFF-state resistance (R_0 = R_{OFF}) of 9 MΩ has been considered in this work. The MOSFET models and the line parasitics and process variations have been adapted from GlobalFoundries 55-nm technology node Process Design Kit (PDK). may be noted that since 1T-1R blocks act as programmable current sink, in the simulated circuit, the integrating capacitor is initially charged to a high voltage (V_{reset}=V_{th}+ΔV_0) and then discharged through the memory block (programmed to store the matrix of weights w_{ij}) during the VMM operation. The difference ΔV_0 between the initial voltage and the threshold voltage has been set to 0.2 V to minimize the voltage drop across the memristor and eliminate the possibility of accidental cell re-programming (state disturbance).

III. Results and Discussion

A. VMM-Level Implementation

We performed a rigorous analysis to explore the design space for optimizing the performance of the proposed SIR VMM. The device parameters and circuit operating points such as the gate length, input gate voltage, etc. were chosen to minimize the non-idealities such as Channel Length Modulation (CLM), Drain-Induced Barrier Lowering (DIBL), etc. which tend to deviate the behavior of the modified 1T-1R array from the ideal constant current sink. For instance, the input gate voltage was selected to bias the MOSFET within the 1T-1R block in sub-threshold mode which is relatively immune to the

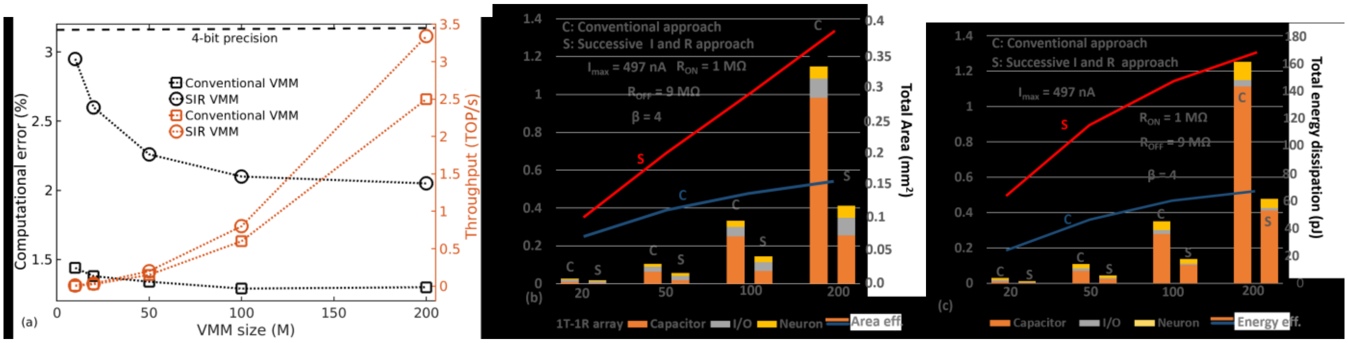

Fig. 2. (a) The computational error (99.9 percentile of the error values for multiple runs of VMM operation with randomly chosen weights and inputs) and throughput, (b) area, and (c) energy efficiency, and their breakdown for the proposed and conventional time-domain VMMs [21] targeting 4-bit precision, for a 55-nm 1T-1R technology, as functions of the linear size of the square weight matrix.

voltage swing on the drain. Moreover, similar to [22], MOSFETs with a larger length (L_g = 240 nm) which exhibit a lower DIBL/CLM error were used instead of the minimum-sized MOSFETs (L_g = 60 nm). Extensive VMM-level simulations of both circuits were performed in HSPICE for different VMM sizes. To calculate the output error, multiple runs of HSPICE simulations were performed using different randomized inputs and weights in each run to traverse the entire sample space of inputs and weights, following the methodology discussed in detail in ref. [22]. The output computational error was then obtained as the maximum difference between the theoretically calculated output time period considering ideal current sinks and the output time period obtained via transient simulation of the entire VMM circuit in HSPICE. Moreover, we also considered 10% fabrication process variations in the line parasitics while characterizing the computed outputs' errors. The single-bit pulse width T_s was chosen as 1 ns (operating frequency of 1 GHz) considering the response time (RC delay including parasitics) of the 1T-1R array as well as the digital I/O circuitry which was designed to operate up to a frequency of 1 GHz. Furthermore, to take into account the worst case due to fabrication errors, a 3-sigma variation of 10% in the line parasitics was also considered in our simulations.

Fig. 2(a) shows the worst-case errors, calculated as the maximum error over the entire parameter space of possible inputs and weights, as functions of the VMM size. Through rigorous analysis, we have found that output errors in the proposed SIR VMM are, just as in the conventional VMM [22], caused mainly by CLM, DIBL, and interconnect parasitics. However, the inclusion of the pass transistors connecting C_I and C_D introduces an additional resistance to the flow of charges during the charge sharing (D) process. A somewhat slower charging and discharging of C_D and C_I, respectively, due to the channel resistance of the pass-gate transistors leads to an increased computational error in the proposed SIR scheme as observed in Fig. 2(a). To mitigate this effect, therefore, the width of the pass transistors was also increased by 120 nm per 10 inputs of the array. Such a sub-optimal design point was chosen after performing estimations based on the actual layout and PDK models from the 55-nm technology node considering all the aforementioned factors. Despite this additional source of computational error, the SIR VMM supports a precision of 4-bits which is suitable for applications that can be performed with high accuracy utilizing even low precision (~4 bits) VMM operations such as inference, classification, recognition etc. [22].

Other possible sources of errors in the proposed SIR VMM are the noise contribution of the memory array (especially on the weights associated with the more significant input bits), unintentional differences between the capacitances of C_I and C_D, and variations in MOSFETs' and memristors' characteristics. Owing to the lack of an experimentally validated noise and variation model for the memristor, their analysis is left for future work, but we believe that they would not affect the difference between the proposed and conventional time-domain VMMs and hence, the conclusions made in this brief would remain the same. Moreover, input-independent errors can be compensated by properly adjusting memory cell currents.

Fig. 2(a) also shows the results for the signal throughput of the proposed SIR VMM and the conventional time-domain VMM. It clearly indicates that the 4-bit SIR VMM provides a significantly higher throughput (for our design point, 3.3 TOp/s) due to the binary-weighted encoding of inputs instead of their direct mapping. For a given VMM size and operating frequency, the throughput of the proposed SIR scheme depends only on the precision (P) and is higher than the conventional time-domain VMM by a factor of $\frac{2^P}{P+2^{P-1}}$ (~1.33 times for bit-precision of 4).

A significant reduction in the load capacitor due to the modified encoding scheme (as explained in section II.A) leads to a considerably improved area and energy efficiency in the SIR VMM as shown in Figs. 2(b) and (c). Moreover, the contribution of the load capacitors into the circuit area and the energy dissipation reduces by more than a factor of 3 in the proposed SIR scheme as compared to the conventional time-domain VMM. Furthermore, the input decoder's contribution to the area and energy also reduces, due to the modified binary weight encoding scheme. Also, Phase II in SIR VMM does not require a dedicated 1T-1R block of the same size for sweeping constant currents as in [22], resulting in smaller overheads of output conversion circuitry. Moreover, the throughput, area- and energy-efficiency of the proposed SIR architecture improves with increasing VMM size similar to the conventional time-domain approach [22] owing to the larger sharing factor of the I/O circuitry with increasing VMM size.

In particular, our simulations indicate that the area and energy efficiencies of the SIR VMM may be as high as, respectively, 1.33 bits/μm^2 and ~1.3 POp/J (1 Op = 1 P-bit arithmetic operation such as addition or multiplication), for a >4-bit 200×200 VMM – the numbers ~2.5× and ~2.65× higher than those for the prior-work time-

TABLE I
VMM-PERFORMANCE BENCHMARKING

Reference	[7]	[8]	[10]	[11]	[15]	[17]	[22]	This work
Approach	CM	CM	CM	TD	TD	TD	TD	TD
Process(nm)	180	22	180	14	55	250	55	55
Precision (bits)	3	~4	~5	<8	~6	~7	~5	~4
EE(TOP/J)	6.4	60	5.7	18	85	<290	498	1305
I/O included	Yes	No	Yes	No	Yes	No	Yes	Yes
Results	Sim	Sim	Exp	Sim	Sim	Sim	Sim	Sim

CM: current-mode TD: time-domain

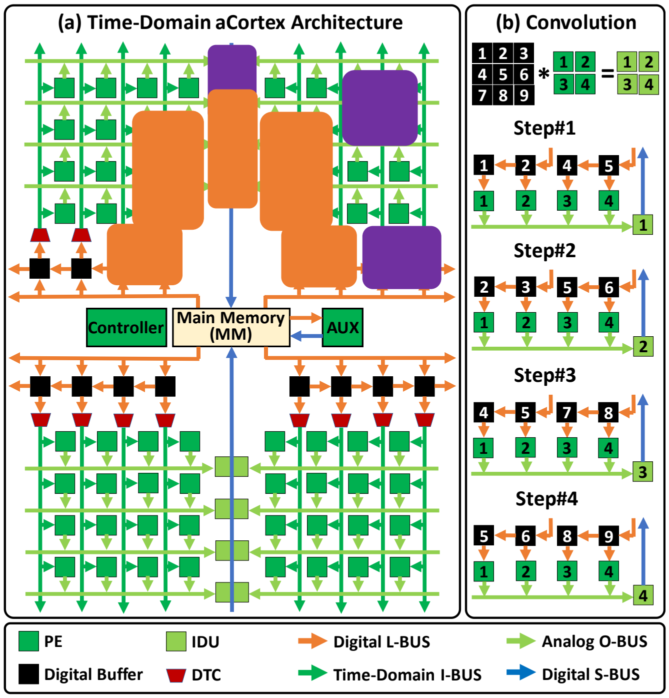

Fig. 3. (a) aCortex overall architecture, main components, and interconnections for time-domain implementation. This figure also shows the packing of weight matrices and active components during each computation step. (b) aCortex row-first operation scheme for convolution tasks.

domain VMM [22] and significantly higher than the other current-mode implementations (shown in table I). Interestingly, the new energy efficiency results for time-domain VMM approach are close to the estimates obtained for current-mode circuits based on (less-scalable) embedded NOR flash memories [23].

B. *System-Level Implementation*

To analyze the system-level performance of the proposed SIR technique, we target a recently-proposed energy/area efficient architecture called aCortex [10] (shown in Fig. 3(a)). The main components of aCortex architecture includes a centralized eDRAM-based Main Memory (MM), a configurable chain of digital buffers, an array of input DTC convertors, 2D arrays of analog/time-domain VMM blocks (PE), an array of output Integrate and Digitalize Units (IDU), and finally a digital auxiliary unit (AUX) used for infrequent pooling/addition/vector-vector multiplication operations. This architecture aims to maximize one-shot analog/time domain computing and minimize the peripheral-circuitry overhead by sharing them on a very large 2D mesh structure of PEs connected via shared analog Input/Output Buses (I-BUS/O-BUS). For each VMM step, inputs are loaded from MM to digital buffers, converted to time-domain signals, and propagated through target columns of PEs. Meanwhile, target PEs/IDUs are activated, and the outputs are integrated and converted back to digital domain at IDUs. At the last step, output results are stored back into the MM. Note that bus splitters are used between PEs/IDUs/Buffers in order to minimize the energy overhead of unutilized potion of different buses.

TABLE II
SYSTEM_LEVEL RESULTS FOR CONVENTIONAL AND SIR ACORTEX

VMM Technique	Conventional			SIR		
Network	**Inc-V1**	**ResNet**	**GNMT**	**Inc-V1**	**ResNet**	**GNMT**
SE (MB/mm^2)	0.135	0.157	0.132	0.26	0.37	0.38
Area (mm^2)	46.2	240.3	602	24	102	209
NVM (%)	38.97	45.2	46.2	37.5	50.9	53.3
Periphery (%)	42.93	49.16	50.25	27.6	36.1	37.2
Other (digital) (%)	18.1	5.64	3.55	34.8	13.0	9.5
EE (TOp/J)	106.7	106.4	421.8	103	107	548
Computation (%)	11.57	9.17	23.5	19.8	25.8	55.8
Communication (%)	41.27	51.64	36.11	41.5	50.8	26.5
Memory access (%)	47.16	39.19	40.39	38.7	23.3	17.7
Throughput (TOp/s)	1.41	1.99	10.88	1.4	1.98	15.6
Inference time (ms)	3.38	9.48	0.22	3.4	9.5	0.16

TABLE III
SYSTEM-LEVEL PERFORMANCE BENCHMARKING

Architecture	DaDian Nao [24]	TPU [25]	ISAAC [26]	PUMA [27]	aCortex [10]	This Work
Tech. node	28 nm	28 nm	32 nm	32 nm	55 nm	**55 nm**
Approach	digital	digital	ReRAM	ReRAM	2D-NOR	**1T1R**
Precision(bits)	16	8	16	16	4	**4**
Area (mm^2)	88	330	85.4	90.6	292.9	**209**
Power (W)	20.1	40	65.8	62.5	0.039	**0.025**
Thr.put(TOp/s)	5.54	92	39.9	52.31	14.97	**15.6**
CE (TOp/s-mm^2)	0.063	0.28	0.46 (0.62*)	0.58 (0.78*)	0.051	**0.074**
SE(MB/mm^2)	0.2	off-chip**	0.74 (0.25*)	0.76 (0.257*)	0.273	**0.38**
EE(TOp/J)	0.286	0.43	0.35 (5.14*)	0.84 (12.09*)	380.25	**548**

*Highly optimistic mapping of the performance results to our design spec (55nm, 4-bit), ** Memory access overhead not included

Table III shows the system-level performance results and the energy/area breakdown for two architectures based on conventional and SIR VMM techniques targeting two DNNs: Inception-V1 and ResNet-152, as well as Google's RNN for language translation (GNMT). As shown in Table II, the SIR scheme achieves ~40% improvement over conventional scheme in terms of Storage Efficiency (SE) while maintaining comparable Energy Efficiency (EE), throughput and Computational Efficiency (CE). Since the SIR scheme requires multiple pulses on the input bus while the conventional scheme requires only one pulse, a higher activity on the I-BUS results in larger communication energy for the SIR scheme. This increase in the communication energy degrades the EE, and ultimately results in comparable system-level EE with the conventional scheme despite the lower contribution from the small load capacitor in the SIR scheme. Moreover, the delay overhead of multiple control signals per operation from Controller to the PEs negates the throughput gain of the SIR scheme over its conventional counterparts, and results in relatively low throughput gain. However, it may be noted that for heavily granular multi-core architectures in which the energy overhead of time-domain signal bus is negligible compared to the digital buses and the control signals are highly localized and fast, VMM-level EE and throughput gain of the SIR scheme results in a significant boost in system-level EE and speed.

Finally, Table III compares the system-level performance metrics of the SIR with 1T-1R current sink-based aCortex architecture and the state-of-the-art digital and mixed-signal accelerators. The SIR scheme based aCortex achieves record-breaking EE and SE while meeting the throughput/inference time for a wide variety of applications such as near sensor inference (IoT/mobile devices).

IV. CONCLUSION

In this brief, we propose a novel successive integration and re-scaling scheme to perform extremely energy- and area-efficient VMM operation with embedded nonvolatile memory devices in the time domain. The proposed approach alleviates the need for utilizing bulky load capacitor for integrating the dot-product in the conventional memory-based time domain approaches and allows for linear scaling of VMM latency with input/computing precision. We believe that this work is an important step in the quest for ultra-efficient VMM engines, providing new optimal design options for neuromorphic processors [28]-[29].

Acknowledgment

The authors would like to thank Prof. K. Likharev for useful discussions. This work was supported by gifts from Google and Applied Materials Inc.

References

[1] C. Mead, *Analog VLSI and Neural Systems*, Addison Wesley, 1989.

[2] J. Hasler and H. Marr, "Finding a roadmap to achieve large neuromorphic hardware systems", *Front. Neurosci.*, vol. 7, art. 118, 2013. DOI: 10.3389/fnins.2013.00118

[3] K. Likharev, "CrossNets: Neuromorphic hybrid CMOS/nanoelectronic networks", *Sci. Adv. Mat.*, vol. 3, pp. 322 - 331, 2011. DOI: 10.1166/sam.2011.1177

[4] G. Indiveri *et al.*, "Neuromorphic silicon neuron circuits", *Front. Neurosci.*, vol. 5, pp. 1–23, 2011. DOI: 10.3389/fnins.2011.00073

[5] C. R. Schlottmann and P. E. Hasler, "A highly dense, low power, programmable analog vector-matrix multiplier: The FPAA implementation", *IEEE J. Emerging and Selected Topics in Circuits and Systems*, vol. 1, pp. 403-411, 2011. DOI: 10.1109/JETCAS.2011.2165755

[6] G. W. Burr, *et. al*, "Experimental demonstration and tolerancing of a large-scale neural network (165,000 Synapses), using phase-change memory as the synaptic weight element," in: *Proc. Int. Electron Device Meeting*, CA, Dec. 2014. DOI: 10.1109/IEDM.2014.7047135

[7] J. Binas, D. Neil, G. Indiveri, S. C. Liu, and M. Pfeiffer, "Precise deep neural network computation on imprecise low-power analog hardware," *arXiv preprint* arXiv:1606.07786, 2016.

[8] M. Hu, J. P. Strachan, Z. Li, E. M. Grafals, N. Davila, C. Graves, S. Lam, N. Ge, J. J. Yang, and R. S. Williams, "Dot-product engine for neuromorphic computing: programming 1T1M crossbar to accelerate matrix-vector multiplication," in: *Proc. Design Automation Conference*, pp.1-6, Austin TX, June 2016. DOI: 10.1145/2897937.2898010

[9] E. H. Lee, and S. S. Wong, "Analysis and design of a passive switched-capacitor matrix multiplier for approximate computing*," IEEE J. Solid-State Circuits*, vol. 52, no. 1, pp.261-271, 2017. DOI:10.1109/JSSC.2016.2599536

[10] M. Bavandpour, M. R. Mahmoodi, H. Nili, F. M. Bayat, M. Prezioso, A. Vincent, D. B. Strukov, and K. K. Likharev, "Mixed-signal neuromorphic inference accelerators: Recent results and future prospects," in: *Proc. Int. Electron Device Meeting*, San Francisco, CA, Dec. 2018. DOI: 10.1109/IEDM.2018.8614659

[11] M. J. Marinella, S. Agarwal, A. Hsia, I. Richter, R. Jacobs-Gedrim, J. Niroula, S. J. Plimpton, E. Ipek, and C. D. James, "Multiscale co-design analysis of energy, latency, area, and accuracy of a ReRAM analog neural training accelerator," *IEEE J. Emerging and Selected Topics in Circuits and Systems*, vol. 8, no. 1, pp. 86-101, 2018. DOI: 10.1109/JETCAS.2018.2796379

[12] W. Woods, and C. Teuscher, "Approximate vector matrix multiplication implementations for neuromorphic applications using memristive crossbars," in *IEEE/ACM NANOARCH,* pp. 103-108, July 2017. DOI: 10.1109/NANOARCH.2017.8053729

[13] T. Tohara, H. Liang, H. Tanaka, M. Igarashi, S. Samukawa, K. Endo, Y. Takahashi and T. Morie, "Silicon nanodisk array with a fin field-effect transistor for time-domain weighted sum calculation toward massively parallel spiking neural networks," *Appl. Phys. Expr.*, vol. 9, no. 3, p.034201, 2016. DOI: 10.7567/APEX.9.034201

[14] T. Morie, H. Liang, T. Tohara, H. Tanaka, M. Igarashi, S. Samukawa, K. Endo, and Y. Takahashi, "Spike-based time-domain weighted-sum calculation using nanodevices for low power operation," in: *Proc. IEEE Int. Nanotechnology Conference*, pp. 390-392, Sendai, Japan, Aug. 2016. DOI: 10.1109/NANO.2016.7751490

[15] M. Bavandpour, M. R. Mahmoodi, and D. B. Strukov, "Energy-efficient time-domain vector-by-matrix multiplier for neurocomputing and beyond," *IEEE Trans. Circuits and Systems II*, 2019. DOI:10.1109/TCSII.2019.2891688M

[16] D. Miyashita, S. Kousai, T. Suzuki, and J. Deguchi, "A neuromorphic chip optimized for deep learning and CMOS technology with time-domain analog and digital mixed-signal processing*," IEEE J. Solid-State Circuits*, vol. 52, no. 10, pp. 2679-2689, 2017. DOI: 10.1109/JSSC.2017.2712626

[17] Q. Wang, H. Tamukoh, and T. Morie, "A time-domain analog weighted-sum calculation model for extremely low power VLSI implementation of multi-layer neural networks," *arXiv preprint* arXiv:1810.06819, 2018.

[18] S. Gopal, P. Agarwal, J. Baylon, L. Renaud, S. N. Ali, P. P. Pande, and D. Heo, "A spatial multi-bit sub-1-V time-domain matrix multiplier interface for approximate computing in 65-nm CMOS," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 8 no. 3, pp.506-518, July 2018. Doi: 10.1109/JETCAS.2018.2852624

[19] M. Yamaguchi, G. Iwamoto, H. Tamukoh, and T. Morie, "An energy-efficient time-domain analog VLSI neural network processor based on a pulse-width modulation approach," *arXiv preprint* arXiv:1902.07707, Feb. 2019.

[20] L. R. Everson, M. Liu, N. Pande, and C. H. Kim, "An energy-efficient one-shot time-based neural network accelerator employing dynamic threshold error correction in 65 nm*," IEEE Journal of Solid-State Circuits*, 2019. Doi: 10.1109/JSSC.2019.2914361

[21] M. Yamaguchi, G. Iwamoto, Y. Abe, Y. Tanaka, Y. Ishida, H. Tamukoh, and T. Morie, "Live demonstration: A VLSI implementation of time-domain analog weighted-sum calculation model for intelligent processing on robots," in: *Proc. IEEE Int. Symposium on Circuits and Systems*, pp. 1-1, Sapporo, Japan, May 2019. Doi: 10.1109/ISCAS.2019.8702222

[22] S. Sahay, M. Bavandpour, M.R. Mahmoodi, and D. B. Strukov, "Time-domain mixed-signal vector-by-matrix multiplier exploiting 1T-1R array," *arXiv preprint* arXiv:1905.09454, June 2019.

[23] 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. in: *Proc. Design Automation Conference*, art. 22, San Francisco CA, June 2018. DOI: 10.1145/3195970.3195989

[24] Y. Chen, T. Luo, S. Liu, S. Zhang, L. He, J. Wang, L. Li, T. Chen, Z. Xu, N. Sun, and O. Temam, "DaDianNao: A Machine-Learning Supercomputer," in 47th Annual IEEE/ACM International Symposium on Microarchitecture, Cambridge, pp. 609-622, 2014.

[25] N.P. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A. Borchers, and R. Boyle, "In-datacenter performance analysis of a tensor processing unit," in ACM/IEEE 44th Annual International Symposium on Computer Architecture (ISCA), Toronto, ON, pp. 1-12, 2017.

[26] A. Shafiee, A. Nag, N. Muralimanohar, R. Balasubramonian, J.P. Strachan, M. Hu, R.S. Williams, and V. Srikumar, "ISAAC: A Convolutional Neural Network Accelerator with In-Situ Analog Arithmetic in Crossbars," in ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA), Seoul, pp. 14-26, 2016.

[27] A. Ankit, I.E. Hajj, S.R. Chalamalasetti, G. Ndu, M. Foltin, R.S. Williams, P. Faraboschi, J.P. Strachan, K. Roy, and D.S. Milojicic, "PUMA: A programmable ultra-efficient memristor-based accelerator for machine learning inference," in arXiv preprint arXiv:1901.10351 (2019).

[28] M. Bavandpour, M.R. Mahmoodi, S. Sahay, and D. B. Strukov, "Mixed signal neuromorphic processors: Quo vadis?", in *Proc. IEEE SOI-3D-Subthreshold Microelectronics Technology Unified Conference*, San Jose, CA, Oct. 2019 (accepted).

[29] M. Bavandpour, S. Sahay, M.R. Mahmoodi and D. Strukov, "3D-aCortex: An ultra-compact energyefficient neurocomputing platform based on commercial 3D-NAND flash memories" arXiv preprint, arXiv:1908.02472, 2019.