A Mixed-Signal Approach to Memristive Neuromorphic System Design

Gangotree Chakma, Sagarvarma Sayyaparaju, Ryan Weiss and Garrett S. Rose
Department of Electrical Engineering and Computer Science
University of Tennessee, Knoxville
Knoxville, Tennessee 37996 USA
Email: {gchakma, ssayyapa, rweiss, garose}@utk.edu

***Abstract*—In this paper we present a memristive neuromorphic system for higher power and area efficiency. The system is based on a mixed signal approach considering the digital nature of the peripheral and control logics and the integration being analog. So, the system is connected digitally outside but the core is purely analog. This mixed signal approach provides the advantage of implementing neural networks with spiking events in a synchronous way. Moreover, the use of nano-sclae memristive device saves the area and power of the system and some considerations about the the device have also been proposed in the paper to make the system more energy efficient.**

I. Introduction

The human brain is comprised of a complex interconnection of neurons that process and transmit data via electro-chemical signals. These neurons are interconnected at junctures known as synapses. The "strength" of the signal transmitted from one neuron to another is proportional to the strength of their interconnection, known as the *synaptic weight*. Each neuron performs the weighted summation of the signals it receives from its preceding neurons. When this summation exceeds a threshold, it transmits a *fire* signal to the succeeding neurons. When this condition occurs, the neuron is said to have fired.

The striking feature of biological neural networks is their ability to adapt their architecture to produce the expected outputs when performing tasks such as image and speech recognition. This adaptation is performed by a process known as *learning* wherein the synaptic weights are updated, thereby affecting the information flow in the neural network.

Artificial Neural Networks (ANNs) are a network paradigm that mimic biological neural networks. They consist of a mathematical model that defines how neurons are interconnected, the strengths of their connections (synaptic weights), how weights are updated, and the behavior of neuron firing events. While ANNs have been shown to be effective in representative applications such as pattern, image and text recognition, they are still reliant on conventional von Neumann machines for implementation, which yield the expected results, but the throughput is incomparable to their biological counterparts. This is because the machines that run these ANN algorithms process information in a sequential manner unlike biological neural networks that are truly parallel.

This need for parallel processing motivated the research on dedicated hardware for ANNs. Such hardware for neural networks is known as neuromorphic circuit. Numerous approaches to neuromorphic computing have been proposed, many of which use digital [1] or analog CMOS approaches [2]–[4]. While the digital implementations have precision, robustness, noise resilience and scalability, they are area intensive [2]. Their analog counterparts are quite efficient in terms of silicon area and processing speed. However, they rely on representing synaptic weights as volatile voltages on capacitors [4] or in resistors [5], which do not lend themselves to energy and area efficient learning. A review of several existing implementations of neural networks can be found in [6].

Lately, the semiconductor industry has begun to experience a significant slowdown in performance improvements gained from technology scaling. While this is due in part to the impending end of Moore's Law scaling, power consumption and architectural limitations have also become critical limiting factors for the level of performance achievable. The research proposed here aims to overcome this roadblock by (1) leveraging an emerging nano-scale device (i.e. the memristor [7]) and (2) the Spiking Neural Network (SNN) architecture to realize neuromorphic computing [8]. A memristor-based Dynamic Adaptive Neural Network array (mrDANNA) is described here that addresses contemporary application challenges while also enabling continued performance scaling.

The mrDANNA architecture is based on the Neuroscience-Inspired Dynamic Architecture (NIDA) presented earlier in [9], [10]as an approach to applying neuromorphic computing principles to a wide variety of applications. Key features of the NIDA architecture include: 1) a spiky representation of data, 2) adaptability of the system during run-time, and 3) a synaptic representation including delay distance as well as weight information. The inclusion of delay distance (i.e. a programmable delay between pre- and post-synaptic neurons) is expected to be of particular benefit in the processing of spatio-temporal data. The structure and simplicity of the NIDA architectural model have been leveraged in the development of a Dynamic Adaptive Neural Network Array (DANNA) [11], an efficient digital system constructed from a basic element that can be configured to represent either a neuron or a synapse. Unique characteristics of the NIDA/DANNA approach over other neuroscience-inspired systems include: a simplified neuron model, a higher functionality synapse model, real-time dynamic adaptability, configurability of the overall

978-1-5090-6389-5/17/$31.00 ©2017 IEEE

neuromorphic structure (e.g. number of neurons, number of synapses and connections), and scalability for element performance and system capacity.

Our mrDANNA network utilizes a pair of memristors to realize a synapse. This synapse acts as an electrical interlink between a pair of neurons that are analog CMOS circuits consisting of capacitors and operational amplifiers. However, a digital control circuit drives the memristors in the synapse. Spiking events of the neuron are sampled (using a clock), digitized and then used to drive the synapse. Henceforth, our synchronous mixed-signal approach to implement the memristive synapse-neuron system blends together the advantages associated with both digital and analog CMOS design in addition to the merits of using a nano-device. Additionally, we utilize the mrDANNA fabric for pattern recognition as a proof of concept of the proposed system. Results presented show high accuracy of the proposed fabric in recognizing basic shapes.

The remainder of the paper is as follows. Section 2 details the background for memristive devices (fabrication, characterization and modeling) and the DANNA system. The circuit specifications and design for the hardware implementation of the synapses and neurons with the algorithm of using it in a particular neuromorphic design are described in section 3. Results illustrated in section 4 show the operation and benefits of the proposed system, with Section 5 detailing future work and eventual fabrication plans for the mrDANNA system.

II. Background

Memristors are two terminal nanoscale non-volatile devices first theorized by Leon O. Chua [7] in 1971. Memristors are resistors whose resistance can be modulated by the amount of voltage flux or charge injected into the device. A memristor can attain multiple resistance levels between the two bounds known as their low resistance state (LRS) and high resistance state (HRS). The LRS and HRS of any memristor is dependent on the switching material, process conditions, noise and environmental conditions. Materials used to build memristors include TaO_x [12], TiO_2 [13], and HfO_x [14]. All of these memristors are differentiated by their LRS values, LRS to HRS ratios, threshold voltage, and switching time. For this design a suitable ranges of LRS and HRS have been considered based on the values in literature.

Owing to their programmability and non-volatility, artificial synapses can be implemented using memristors to represent weight values and transmit analog weighted results to post-synaptic neurons. The neuron uses the analog output of the synapse to produce a firing event (or spike) that is synchronized with the system. Further, the system considered here leverages the unsupervised Long Term Plasticity for on-chip learning. Based on the temporal relation of the pre- and the post-neuron's fires, the synaptic weight is modified by the synaptic control block and the feedback from the post-neuron.

III. Memristive DANNA System

The memristive DANNA system Fig. 1 consists of several mrDANNA cores. Each core contains n number of synapses and an analog Integrate and Fire (IAF) neuron. The construction and function of synapse and neuron are described in the below subsections with elaboration.

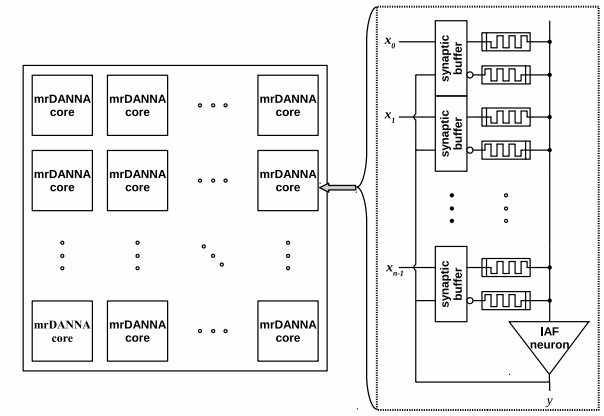

Fig. 1: Connection of mrDANNA system.

A. Synapse

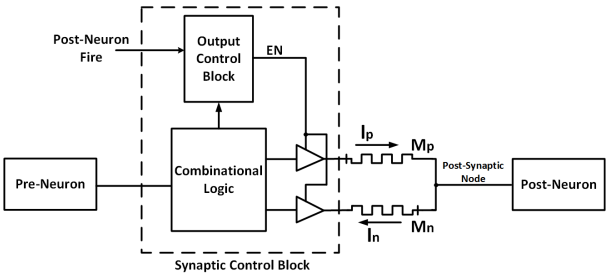

Fig. 2: Twin memristor synapse along with its control block providing the interlink between the pre- and post-neuron.

The synapse design considered here (shown in Fig. 2) uses a twin memristor configuration to store the weight value. The synaptic weights are represented using memristors, where the current flowing out of it (into the post-synaptic node) is dependent on its weight, which is in turn dependent on the memristances of the two memristors. This approach follows that of several other memristor-based neural network designs where voltage inputs across memristive weights yields a weighted sum in the form of a current [15]–[18].

The use of two memristors enables the realization of negative weights. A similar approach has been presented in [17], wherein a pair of memristive crossbar arrays are used to represent the negative and positive components of the weights. However, we do not presume the implementation here to be a crossbar though that is an alternative design option for potentially increased density. In the memristor pair for each synapse, one memristor is used to drive positive current while the other drives negative current (pulls current from the

integrator). The effective current flowing into the post-neuron thus depends on the relative values of the two memristances. The weight of this synapse is proportional to the effective conductivity of the pair of memristors, given by:

$$G_{eff} = \frac{1}{M_p} - \frac{1}{M_n} \tag{1}$$

If the memristance of both memristors in the pair are equal, their currents will cancel each other for any given input spike and the effective weight is zero. On similar lines, if M_p is lesser (greater) than M_n, the weight is positive (negative).

The synapse uses a digital logic block to provide driving voltages to each memristor in the pair. The synapse here operates in two phases, namely *accumulation* and *learning*. The *accumulation* phase occurs when the pre-neuron fires. This fire triggers the synaptic control block to drive a positive current through M_p and a negative current through M_n. The effective current flowing into the post-neuron either accumulates charge on it or discharges from it during this phase. *Learning* phase occurs when the post-neuron fire. If the pre-neuron fires before the post-neuron, the synapse weight is increased (potentiation). On the contrary, if the pre-neuron fires after the post-neuron, the synapse's weight is decreased (depression). This is in accordance with the STDP rule, which is believed to be the cause for learning in biological neural networks.

B. Analog Neuron

For the neuron, we implement an integrate-and-fire circuit (Fig. 3) similar to that described by Wu *et al.* [19]. Here the neurons are designed to produce spikes based on the incoming synaptic signals. The design allows the neurons to operate in two different phases, integration phase and the firing phase. When the neuron operates in its integration phase, the op amp acts as an integrator such that the capacitor, C_{fb} accumulates charge (from the current coming from the synapse) resulting in the change in membrane potential V_{mem}. A comparator circuit compares the membrane potential V_{mem} with the threshold voltage V_{th} and generates firing spikes.

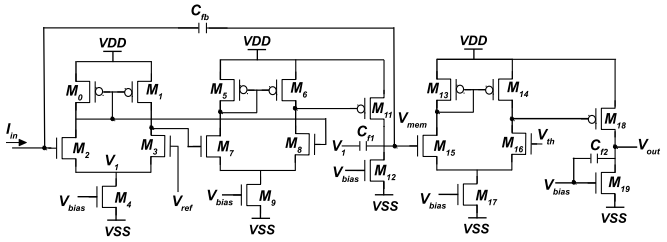

Fig. 3: Analog integrate and fire neuron.

IV. Results

To showcase the usefulness of this type of network with memristive synapses and integrate and fire neurons, a circuit for recognizing some basic shapes such as triangle, square, diamond and plus has been constructed. The proposed circuit has been simulated in Cadence Spectre by implementing the Verilog-A code of the memristor model. Besides recognizing the perfect triangle, some imperfect noisy image of triangles, squares, diamonds and also plus signs have been considered to determine the accuracy level of recognition of this network. A python script was used to generate the noisy signals. The script was written in such a way to generate random noise bits and any number of bits within twenty five can be used in the script and then those images were used in Cadence Spectre simulation. Fig. 4 illustrates the perfect and examples of the noisy patterns input to the circuit.

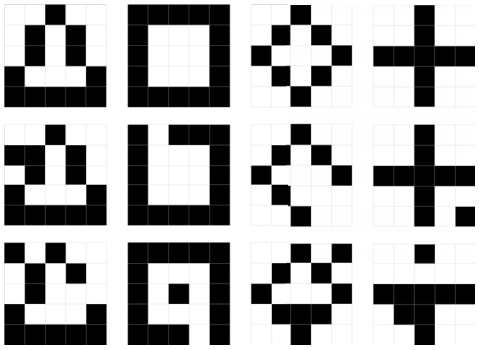

Fig. 4: Example 5×5 test images for noiseless shapes (top), 1 error pixel (middle) and 2 error pixels (bottom). The four basic shapes considered for the classification network presented are illustrated: triangle, square, diamond and plus.

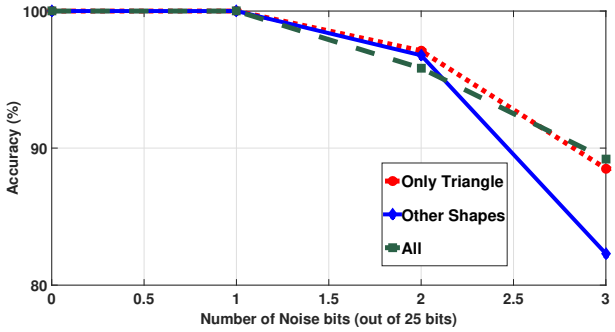

Fig. 5: Accuracy of triangle recognition versus other shapes.

Zero, one, two and three noise bits were considered for simulation and results for percentage of accuracy are shown in Fig.5. The results show that the network recognizes most of the cases with noisy bits up-to 3bits among the 25 bits of the image. The accuracy of recognizing only triangles are a bit higher than recognizing all shapes. The network recognizes images with one noisy bit with a hundred percent accuracy for all cases but with the increase in number of noise bits, the accuracy level slightly goes down but the percentage of accuracy is higher than 80 percent at 3 noise bits, which makes the circuit worthy enough in recognizing a particular shape.

The efficiency of the network is reflected by it's design metrics such as the average power. Initially the circuit was

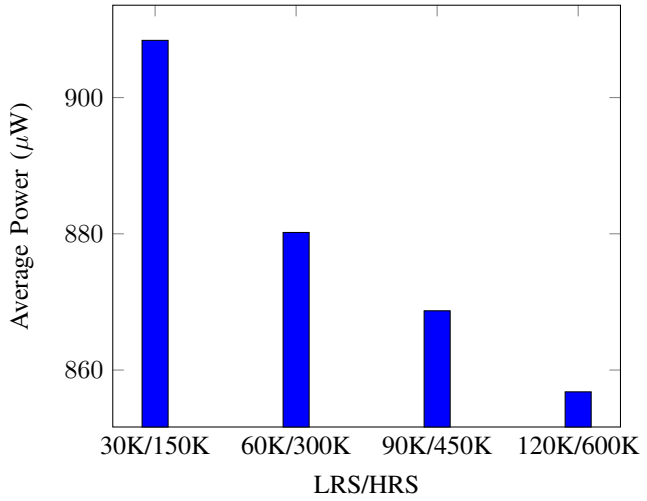

Fig. 6: Average application power.

simulated for LRS and HRS values of 30KΩ and 150KΩ respectively. Average power was calculated for those values and the result was observed to be too high. This high power consumption is an artifact of using the analog neurons and the relatively low resistance levels that drive higher currents resulting in high power dissipation. The power consumed by each analog neuron during the total simulation is $196.5\mu W$ and the power of per digital block is $1.31\mu W$. To ensure lower power consumption, the LRS and HRS levels have been altered to different values such as 60KΩ-300KΩ, 90KΩ-450KΩ, 120KΩ-600KΩ etc. The results show that increasing the resistance levels causes the power to decrease.

V. Conclusion

In this paper, memristive device is used for the development of a memristive dynamic adaptive neural network array (mrDANNA) fabric. We have described a basic pattern recognition network as an application example for the re-configurable mrDANNA system. This work demonstrates the efficiency of recognizing different patterns and also presents the power consumption for different LRS and HRS levels. In future work, more complex networks for spatio-temporal data applications and large pattern recognition will be solved with this efficient computing architecture.

Acknowledgment

The authors thank Dr. Mark Dean, Md. Musabbir Adnan, and Sherif Amer from the University of Tennessee, Knoxville for interesting and useful discussions on this topic.

This material is based in part upon research sponsored by Air Force Research Laboratory under agreement number FA8750-16-0065 and the National Science Foundation under Grant No. NCS-FO-1631472. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon.

References

[1] J.-s. Seo, B. Brezzo, Y. Liu, B. D. Parker, S. K. Esser, R. K. Montoye, B. Rajendran, J. A. Tierno, L. Chang, D. S. Modha *et al.*, "A 45nm cmos neuromorphic chip with a scalable architecture for learning in networks of spiking neurons," in *Custom Integrated Circuits Conference (CICC), 2011 IEEE*. IEEE, 2011, pp. 1–4.

[2] B. Linares-Barranco, E. Sanchez-Sinencio, A. Rodriguez-Vazquez, and J. L. Huertas, "A cmos analog adaptive bam with on-chip learning and weight refreshing," *IEEE Transactions on Neural networks*, vol. 4, no. 3, pp. 445–455, 1993.

[3] C. Schneider and H. Card, "Analog cmos synaptic learning circuits adapted from invertebrate biology," *IEEE transactions on circuits and systems*, vol. 38, no. 12, pp. 1430–1438, 1991.

[4] ——, "Cmos implementation of analog hebbian synaptic learning circuits," in *IJCNN-91-Seattle International Joint Conference on Neural Networks*, vol. i, Jul 1991, pp. 437–442 vol.1.

[5] H. Graf, L. Jackel, R. Howard, B. Straughn, J. Denker, W. Hubbard, D. Tennant, D. Schwartz, and J. S. Denker, "Vlsi implementation of a neural network memory with several hundreds of neurons," in *AIP conference proceedings*, vol. 151, no. 1. AIP, 1986, pp. 182–187.

[6] J. Misra and I. Saha, "Artificial neural networks in hardware: A survey of two decades of progress," *Neurocomputing*, vol. 74, no. 1, pp. 239–255, 2010.

[7] L. O. Chua, "Memristor-the missing circuit element," *IEEE Transactions on Circuit Theory*, vol. 18, no. 5, pp. 507–519, September 1971.

[8] W. Maass, "Networks of spiking neurons: the third generation of neural network models," *Neural networks*, vol. 10, no. 9, pp. 1659–1671, 1997.

[9] C. D. Schuman and J. D. Birdwell, "Dynamic artificial neural networks with affective systems," *PLoS ONE*, vol. 8, no. 11, p. e80455, November 2013. [Online]. Available: http://dx.doi.org/10.1371

[10] C. Schuman, J. Birdwell, and M. Dean, "Neuroscience-inspired inspired dynamic architectures," in *Biomedical Science and Engineering Center Conference (BSEC), 2014 Annual Oak Ridge National Laboratory*, May 2014, pp. 1–4.

[11] M. E. Dean, C. D. Schuman, and J. D. Birdwell, "Dynamic adaptive neural network array," in *International Conference on Unconventional Computation and Natural Computation*. Springer, 2014, pp. 129–141.

[12] J. J. Yang, M. Zhang, J. P. Strachan, F. Miao, M. D. Pickett, R. D. Kelley, G. Medeiros-Ribeiro, and R. S. Williams, "High switching endurance in taox memristive devices," *Applied Physics Letters*, vol. 97, no. 23, p. 232102, 2010.

[13] G. Medeiros-Ribeiro, F. Perner, R. Carter, H. Abdalla, M. D. Pickett, and R. S. Williams, "Lognormal switching times for titanium dioxide bipolar memristors: origin and resolution," *Nanotechnology*, vol. 22, no. 9, p. 095702, 2011.

[14] H. Lee, Y. Chen, P. Chen, T. Wu, F. Chen, C. Wang, P. Tzeng, M.-J. Tsai, and C. Lien, "Low-power and nanosecond switching in robust hafnium oxide resistive memory with a thin ti cap," *IEEE Electron Device Letters*, vol. 31, no. 1, pp. 44–46, 2010.

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

[16] C. E. Merkel and D. Kudithipudi, "A current-mode cmos/memristor hybrid implementation of an extreme learning machine," in *Great Lakes Symposium on VLSI 2014, GLSVLSI '14, Houston, TX, USA - May 21 - 23, 2014*, 2014, pp. 241–242. [Online]. Available: http://doi.acm.org/10.1145/2591513.2591572

[17] M. Hu, H. Li, Y. Chen, Q. Wu, G. S. Rose, and R. W. Linderman, "Memristor crossbar-based neuromorphic computing system: A case study," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 25, no. 10, pp. 1864–1878, October 2014. [Online]. Available: http://dx.doi.org/10.1109/TNNLS.2013.2296777

[18] I. Kataeva, F. Merrikh-Bayat, E. Zamanidoost, and D. Strukov, "Efficient training algorithms for neural networks based on memristive crossbar circuits," in *2015 International Joint Conference on Neural Networks, IJCNN*, July 2015, pp. 1–8. [Online]. Available: http://dx.doi.org/10.1109/IJCNN.2015.7280785

[19] X. Wu, V. Saxena, and K. A. Campbell, "Energy-efficient STDP-based learning circuits with memristor synapses," in *Proceedings of SPIE*, vol. 9119, 2014, pp. 911 906–1–911 906–7. [Online]. Available: http://dx.doi.org/10.1117/12.2053359