Pulse-Width Modulation based Dot-Product Engine for Neuromorphic Computing System using Memristor Crossbar Array

[†]Hao Jiang, [†]Kevin Yamada, [†]Zizhe Ren, [†]Thomas Kwok, [†]Fu Luo, [‡]Qing Yang, [†]Xiaorong Zhang, [§]J. Joshua Yang, [§]Qiangfei Xia, [‡]Yiran Chen, [‡]Hai Li, [¶]Qing Wu, and [¶]Mark Barnell
[†]School of Engineering, San Francisco State University, San Francisco, California, USA
[‡]Department of Electrical and Computer Engineering, Duke University, Durham, North Carolina, USA
[§]Department of Electrical Engineering, University of Massachusetts Amherst, Massachusetts, USA
[¶]Information Directorate, Air Force Research Laboratory, New York, USA

***Abstract*—The Dot-Product Engine (DPE) is a critical circuit for implementing neural networks in hardware. The recent-developed memristor crossbar array technology, which is able to efficiently carry out dot-product multiplication and update its weights in real time, has been considered as one of the viable technologies to build a high-efficient neural network computing system. In this paper, the Pulse-Width-Modulation (PWM) based DPE has been presented and analyzed. Here, the PWM based signal, instead of the traditional amplitude modulated (AM) signal, is used as the computation variable. Comparing to the existing AM based system, this PWM counterpart provides an alternative approach to reduce the power consumption and chip area of its peripheral circuits. Power and area saving becomes more prominent when the size and/or the number of arrays increase. This new approach also provides the critically needed scalability to accommodate the computation variable with higher precision. In this paper, a 4-bit (can be easily expanded to 8-bit) feed forward neural network with 3-bit weights (memristor's conductance) is constructed using the proposed PWM DPE to identify digits from the MNIST data set. The circuit system is implemented in 130 nm standard CMOS technology. The entire circuit system consumes about 53mW with more than 86% recognition accuracy in average.**

***Keywords*—Neuromorphic computing, Dot-product engine, memristor crossbar array.**

I. Introduction

Machine learning has been widely used to boost the computation efficiency in many data-intensive applications. Neural networks, such as CNNs and DNNs (Convolutional and Deep Neural Networks), are widely used in machine learning [1][2]. In CNNs or DNNs, a large number of dot-product (multiply-accumulate) operations with matrices in various sizes have been frequently carried out [1][2]. DPE (Dot-Product Engine) becomes one of the most critical components in these popular machine learning implementations. To avoid the challenge of the memory wall [3], the MCA (Memristor-Crossbar-Array) has been proposed to seamlessly combine the memory and the dot-product operations in the hardware based DPE to significantly boost the computation efficiency in CNNs and DNNs [4].

The circuitry implementation of this promising MCA based DPE has been extensively discussed in [5] and [6]. In [5] [6], a DAC (Digital-to-Analog Convertor) is used to convert n-bit digital input into an AM (Amplitude Modulated) voltage signal as the computation variable, as depicted in Fig 1(b). The dot-product operation is carried out by the MCA. The amplitude of the MCA output current, which represents the dot-product operation result, is sensed and amplified by a TIA (Trans-Impedance Amplifier), and then digitized by an ADC (Analog-to-Digital Convertors) [5][6]. In [5] [6], a DAC and TIA are required for every row and every column of the MCA, respectively. An ADC is shared by two or more columns depending on the desired computation speed and precision [5]. Both [5][6] indicate that the power consumption and the required chip areas of these peripheral I/O circuits significantly hinder the MCA based DPE's computation efficiency. The efficiency could get worse when the size (number of the rows and columns) of MCA and the number of matrices involved

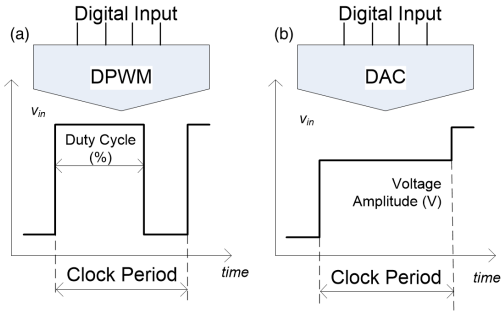

Fig. 1. Illustration of the (a) PWM (Pulse-Width Modulation) based and (b) AM (Amplitude Modulation) based computation variable

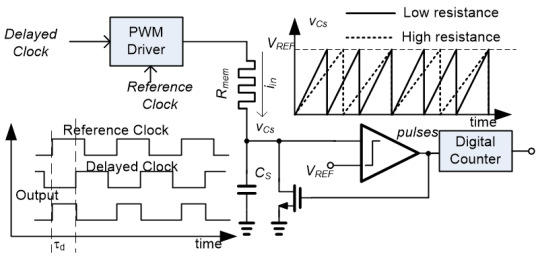

Fig. 2. The simplified schematic of the PWM write and read circuit

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

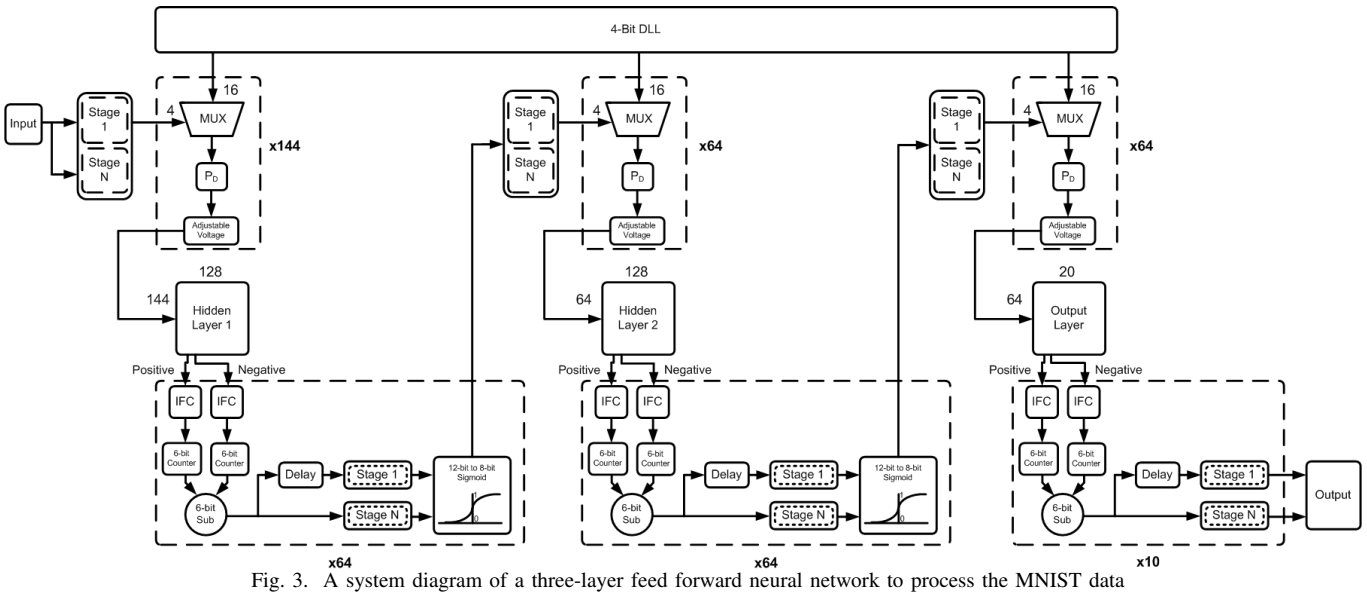

Fig. 3. A system diagram of a three-layer feed forward neural network to process the MNIST data

in the DPE grows.

In this paper, the PWM (Pulse-Width Modulation) based signal, instead of AM based signal, has been adopted as the computational variable. As depicted in Fig 1 (a), the duty cycle of a constant amplitude voltage signal is designated as the computation variable. The most conspicuous advantage of the PWM system is its potential to significant reduce power consumption and chip area. By replacing DACs, TIAs and ADCs with "digital-like" circuits, $i.e.$, D-flipflops, IFCs (Integrated-and-Fire Circuit), and counters, the peripheral circuits that are associated with each row or column of the MCA could consume less power compared to the existing AM based circuit system [5][6], specially with a multi-layer neural network with large array size used in a CNN/DNN. With the elimination of amplifiers associated with the DAC, TIA and ADCs, the reduction of the chip area can also be anticipated.

In this paper, the system and the circuit implementation of the PWM DPE are described in Section II, followed by the conclusions in Section III.

II. System and Circuit Implementation

A. Operating Principle

The described PWM based DPE has a global DLL (Delay Locked Loop) for every DPE in the neural networks, and a set of write and read circuits for each row (word line) and column (bit line) of the MCA, as depicted in Fig. 2. The global DLL produces a set of clock signals whose delay is evenly distributed within a clock period [7]. In this design, the DLL is designed to generate 2^n-1 delayed clocks, where n represents the number of bits that each PWM signal is going to represent.

At the input of each row, one of the 2^n-1 delayed clocks, the mth delayed clock, is selected by a n-to-1 multiplexer according to the n-bit digital input. By comparing the selected m-th clock to the reference clock, a PWM driver, which is made of two flipflops (similar to the phase-frequency detector in a regular PLL [8]), is able to produce a voltage signal whose duty cycle is proportional to $m/2^n$ with a constant amplitude (0 to constant V_{in}).

When the constant amplitude PWM voltage signal is applied to each row of the MCA, the averaged current over one clock period at the output of each column is proportional to the overall conductance of the specific column (i.e., weights of the DPE) and the duty-cycle of all input voltage signals (i.e., input of the DPE). Based on the input voltage and the output current that are averaged over one clock period, the PWM based system is equivalent to the AM based counterpart.

The output current can be sensed by the IFC with a digital counter [9], since the input voltage signal has constant amplitude. When the output current from the MCA charges the sensing capacitor, C_S, in the IFC to the pre-defined voltage V_{REF}, C_S will be reset, as depicted in Fig. 2. The reset rate is proportional to the amount of current that charges C_S[9]. By counting the number of resets over one clock period, the averaged output current from the MCA is measured and digitized.

B. Circuit System

A three-layer feed forward neural network using the proposed PWM DPE is constructed using 130 nm standard CMOS technology to process the MNIST data set, as depicted in Fig. 3 [10]. Similar to [10], the size of the three layers is 144×64 (hidden), 64×64 (hidden), and 64×10 (output), respectively. Each layer has two MCAs to represent positive and negative weights, as depicted in Fig. 3. The weights are pre-trained and have 3-bit precision. 12×12 4-bit inputs from MNIST are applied to the first layer MCA with the constant amplitude PWM voltage signals. The outputs from each pair of positive and negative weights first subtract each other and then feed into the sigmoid encoders. The 4-bit output from each signmoid encoder is delivered to the second layer. The digit is identified at the output of the third layer.

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

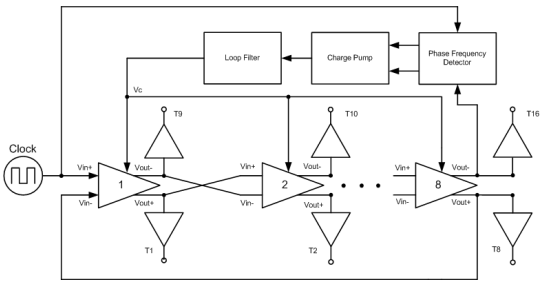

Fig. 4. The circuit schematic of the DLL (Delay Locked Loop)

Using the time-division method, the precision of the computation variable can be expanded to 8-bit or higher easily. To accommodate the computation variable with 8-bit precision, the PWM based DPE can process the first 4 bits and the last 4 bits consecutively using the same 4-bit PWM DPE with two clocks. The output of the first 4 bits passes an one-clock delay cell while the output of the last 4 bits doesn't. Thus, the output can be restored to 8-bit by combining these two outputs in parallel, as depicted in the dashed boxes in Fig. 3. By adding the clocked $4k$-to-4 and 4-to-$4k$ delay cell to the input and the output of each DPE, a 4-bit PWM based DPE is able to operate with $4k$-bit precision by slowing down the computation speed at k-folds. Without the time-division method, the PWM DPE could lose half of its computation speed whenever the precision of the computation variable increases by 1 bit.

C. Circuit Blocks

1) DLL: A traditional DLL is used to produce 2^4-1 evenly delayed clocks to support the 4-bit operation [7]. As depicted in Fig. 4. the DLL consists of 8 differential CML (Current Mode Logic) delay stages whose delay time can be controlled by the control voltage (v_C), a PFD (Phase Frequency Detector) that detects the phase-frequency difference between the last delayed clock and the reference clock, a CP (Charge Pump) and a loop filter to provide v_C to control the delay time of every delay stage. The PFD, CP and loop filter are widely used in PLLs [8]. The DLL is implemented in 130 nm standard CMOS technology. Including the buffers at each tap, the DLL consumes about 850 μW when it operates at 50 MHz.

2) Input Circuit: An input circuit is needed for each row (word-line) of the MCA to translate the digital input to the PWM signal. In the described 4-bit PWM DPE, a 4-to-1 multiplexer is used to select one of 2^4-1 delayed clocks according to the 4-bit digital input, as depicted in Fig. 2 and Fig. 3. In the circuit implementation, a PWM driver that is made of two D-type flip-flops is used to generate a signal whose duty cycle is proportional to the digital input. A digital buffer is added to the output of the PD so that it is able to drive a large-size MCA with many parallel-connected memristors. In this implementation, the input circuit consists of a 4-to-1 MUX, two D-flipflops and a digital buffer.

3) Output Circuit: An output circuit is needed for each column (bit-line) of the MCA to measure current and convert it to a digital word. In the PWM based DPE, the input voltage

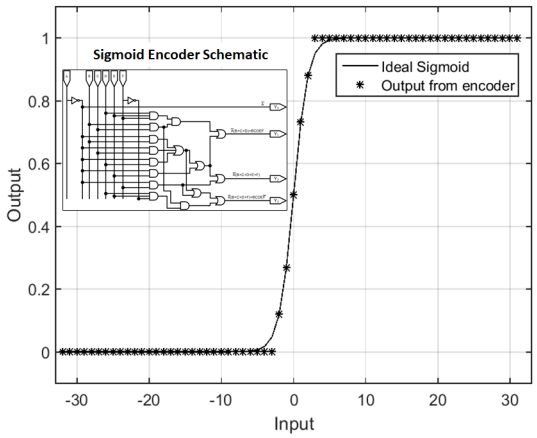

Fig. 5. The output vs. input of the sigmoid encoder and its schematic (inserted)

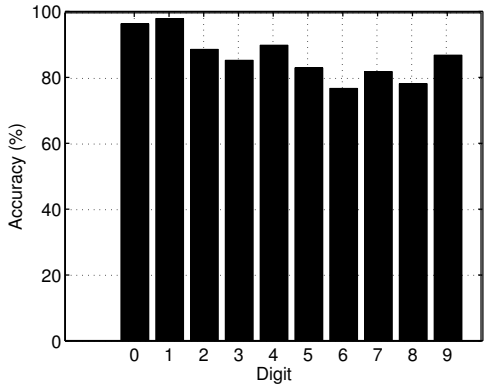

Fig. 6. The accuracy of the network for each digit

of the MCA has constant amplitude. Thus, a high-speed IFC, instead of a large and power-hungry ADC [11], is used to measure the output current from each column of the MCA, and convert the output current into a set of time rated pulses [9][12]. In this implementation, a 6-bit counter is used to count the number of pulses from the output of each IFC within a clock period. The output number represents the averaged output current from each column of the MCA. By subtracting the output of the negative MCA from that of the positive MCA, as depicted in Fig. 3, the digitized output from the dot-product operation is acquired.

4) Activation Function: The sigmoid is widely used as a classic activation function used in feed forward neural networks to introduce non-linearity [13]. Its schematic is inserted in Fig. 5. The output of the sigmoid encoder, is consistent with the ideal sigmoid function, as depicted in Fig. 5. The 6-bit input has sufficient dynamic range for a large number of pulses generated by the IFC. The 4-bit output, which has the same precision as that of the input, differentiates between not firing from fully saturated.

D. System Performance

A three-layer feed-forward multi-layer perceptron that is built upon the PWM based DPEs to classify digits off of a

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

TABLE I
POWER CONSUMPTION OF PERIPHERAL CIRCUITS IN THE PWM BASED NEURAL NETWORK

Components	Numbers	Power/Block	Total
Unit:		μW	mW
4-bit DLL	1	854	0.854
MUX	144+64+64	9	2.45
PWM Driver	144+64+64	32	8.67
IFC	2×(64+64+10)	101	27.85
Counter	2×(64+64+10)	31.7	8.73
Subtractor	64+64+10	16.4	2.67
Sigmoid	64+64	10.2	1.30

randomly assorted 10,000 image MNIST test set, as depicted in Fig. 3. The conductance of the memristors with 3-bit precision is pre-trained using the standard stochastic gradient descent back propagation method. In this implementation, the precision of the memristor's resistance is 3-bit. The size of the input image of the MNIST data set is reduced to 12×12. Each input is represented by a 4-bit variable. The positive and negative MCA represent the positive and negative weights, respectively [10]. After processing 10,000 randomly assorted images, the accuracy of the digit recognition is depicted in Fig. 6. "0" and "1" have above 95% accuracy, while "6" and "8" have above 74% accuracy. Overall, the averaged accuracy is 86.5%. The recognition accuracy shows little improvement when the precision of the computation variable is increased to 8-bit.

The power consumption of each circuit block is analyzed in Table I. The total power consumption of the described neural network is about 53 mW.

III. CONCLUSIONS

In this paper, a PWM based DPE has been presented. Using PWM signal as the computation variable could reduce the peripheral circuits' power consumption and chip area comparing to conventional AM based signals, especially when multiple large-size MCAs are used. With the time-division method, this PWM based DPE is able to construct a high-accuracy neuromorphic computing system, $e.g.$, a 32-bit system, without significantly sacrificing computation speed, increasing power consumption or chip area. The PWM based neuromorphic system is implemented in 130 nm standard CMOS technology to recognize digits from the MNIST data set. The novel PWM based DPE described in this paper has the potential to construct a neuromorphic computing system with multiple layer, large-size MCAs.

ACKNOWLEDGMENT AND DISCLAIMER

This work is supported in part by FA8750-15-1-0069 and FA8750-15-2-0048. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of AFRL, or their contractors.

REFERENCES

[1] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *2009 IEEE 12th International Conference on Computer Vision*, Sept 2009, pp. 2146–2153.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[3] T. Luo, S. Liu, L. Li, Y. Wang, S. Zhang, T. Chen, Z. Xu, O. Temam, and Y. Chen, "Dadiannao: A neural network supercomputer," *IEEE Transactions on Computers*, vol. 66, no. 1, pp. 73–88, Jan 2017.

[4] X. Liu, M. Mao, B. Liu, H. Li, Y. Chen, B. Li, Y. Wang, H. Jiang, M. Barnell, Q. Wu, and J. Yang, "Reno: A high-efficient reconfigurable neuromorphic computing accelerator design," in *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, June 2015, pp. 1–6.

[5] 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 *2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA)*, June 2016, pp. 14–26.

[6] 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 *2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, June 2016, pp. 1–6.

[7] J. G. Maneatis, "Low-jitter process-independent dll and pll based on self-biased techniques," *IEEE J. of Solid State Circ.*, vol. 31, no. 11, pp. 1723–1732, Nov. 1996.

[8] F. M. Gardner, *Phaselock Techniques*, 3rd ed. A John Wiley and Sons, 2005.

[9] C. Liu, B. Yan, C. Yang, L. Song, Z. Li, B. Liu, Y. Chen, H. Li, Q. Wu, and H. Jiang, "A spiking neuromorphic design with resistive crossbar," in *52nd Design Automation Conference (DAC)*, San Francisco, CA, June 2015, pp. 1–6.

[10] C. Liu, Q. Yang, B. Yan, J. Yang, X. Du, W. Zhu, H. Jiang, Q. Wu, M. Barnell, and H. Li, "A memristor crossbar based computing engine optimized for high speed and accuracy." 2016 IEEE Computer Society Annual Symposium on VLSI, July 2016.

[11] B. Murmann. ADC performance survey 1997-2017. [Online]. Available: http://web.stanford.edu/~murmann/adcsurvey.html

[12] H. Jiang, W. Zhu, F. Luo, K. Bai, C. Liu, X. Zhang, J. J. Yang, Q. Xia, Y. Chen, and Q. Wu, "Cyclical sensing integrate-and-fire circuit for memristor array based neuromorphic computing," in *2016 IEEE International Symposium on Circuits and Systems*, May 2016.

[13] F. Li, J. Johnson, and S. Yeung. CS231n Convolutional Neural Networks for Visual Recognition. [Online]. Available: http://cs231n.github.io/neural-networks-1/

978-1-5386-4881-0/18/$31.00 ©2018 IEEE