A DCSRO Based Time Domain MAC Core

Tawfiq Musah[1], Kevin Du[1], and Ahmed Abdelaziz[1]

[1]The Ohio State University

February 22, 2023

Abstract

This article presents a time domain multiply-and-accumulate (MAC) engine used for convolutional neural networks. Time domain is chosen for efficiency as it allows for compact representation of multi-bit inputs on a single wire. This reduces gate count and switching capacitance (Cdyn) compared to traditional all-digital implementation. The inputs are encoded by selecting a pulse of varying width depending on input code. The multiplication operation and accumulation is implemented using a digitally controlled switched-ring oscillator time-to-digital converter functioning as a time accumulator. The digital control allows for accumulation and quantization of two signals simultaneously, halving the required time to quantize a certain value. The proposed MAC is designed in a 28nm CMOS process and can achieve a simulated power efficiency of 0.32pJ/b, which is 1.8X better than what can be achieved by a single input gated ring oscillator (GRO) design.

Received <day> <Month>, <year>; | Revised <day> <Month>, <year>; | Accepted <day> <Month>, <year>

DOI: xxx/xxxx

ORIGINAL ARTICLE

A DCSRO Based Time Domain MAC Core

Kevin Du[1,2] | Ahmed Abdelaziz[1] | Tawfiq Musah*[1]

[1]Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210, USA
[2]Now with, Advanced Micro Devices, Inc, San Jose, CA 95124, USA

Correspondence
*Tawfiq Musah. Email: musah.3@osu.edu

Abstract

This article presents a time domain multiply-and-accumulate (MAC) engine used for convolutional neural networks. Time domain is chosen for efficiency as it allows for compact representation of multi-bit inputs on a single wire. This reduces gate count and switching capacitance (Cdyn) compared to traditional all-digital implementation. The inputs are encoded by selecting a pulse of varying width depending on input code. The multiplication operation and accumulation is implemented using a digitally controlled switched-ring oscillator time-to-digital converter functioning as a time accumulator. The digital control allows for accumulation and quantization of two signals simultaneously, halving the required time to quantize a certain value. The proposed MAC is designed in a 28nm CMOS process and can achieve a simulated power efficiency of 0.32pJ/b, which is 1.8X better than what can be achieved by a single input gated ring oscillator (GRO) design.

KEYWORDS:
Time domain MAC, Convolutional Neural Network, time-to-digital converter, digital-to-time converter

1 | INTRODUCTION

Due to advances in hardware, machine learning has made a resurgence. Using newer and more powerful algorithms, machine learning is able to perform accurate pattern recognition and predictions in applications like speech, image, and facial recognition. Among these algorithms are deep neural networks, which require immense computational resources. Specifically, the convolutional neural network (CNN) has proven to be extremely powerful, allowing state of the art classification accuracy while maintaining low complexity relative to the traditional neural network implementations. The power consumption is primarily determined by the data movement from memory to the processing element and the computational power consumption in the processing element. The adoption in-memory or near memory computing has been shown to drastically reduce the data transport power consumption[1,2]. As such the computation energy is expected to be the dominant in near-/in-memory CNN architectures.

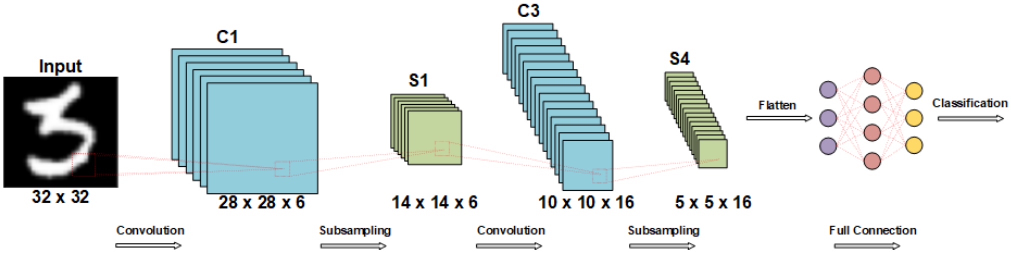
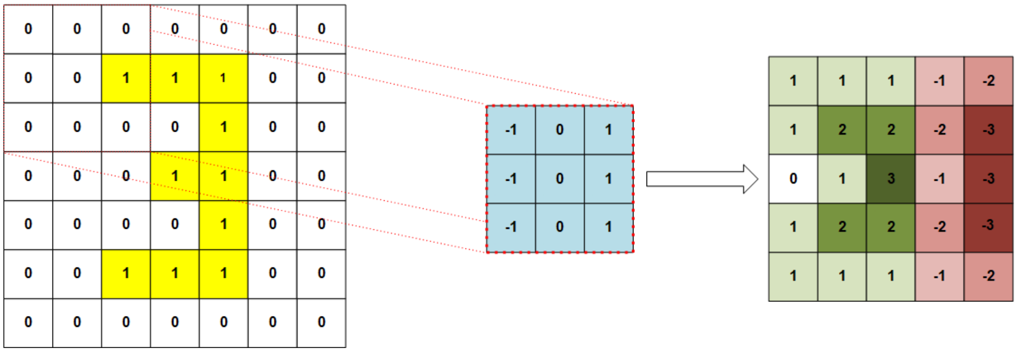


FIGURE 1 LeNet-5 CNN Architecture.



0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	0	0	0	1	0	0
0	0	0	1	1	0	0
0	0	0	0	1	0	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

-1	0	1
-1	0	1
-1	0	1

1	1	1	-1	-2
1	2	2	-2	-3
0	1	3	-1	-3
1	2	2	-2	-3
1	1	1	-1	-2

FIGURE 2 Convolution of image '3' with vertical edge detector filter.

A fundamental example of a CNN is LeNet-5[3], shown in Figure 1 , which was designed for early handwritten character recognition. LeNet-5 consists of convolutional layers and subsampling layers. Convolutional layers extract features from the dataset by sliding a 2-dimensional filter of weights around an image and at each location taking the Frobenius inner product, given by (1). The output of the inner product is then saved to the corresponding location of the resulting matrix.

$$\langle X|W\rangle = Tr\{X^T W\} = \sum_{i,j} x_{ij} * w_{ij}. \tag{1}$$

Figure 2 shows a basic convolution of a vertical edge detector filter with a coarse representation of the number '3'. The result of this convolution is an output feature map with large values corresponding to locations where vertical edges are present. By cascading these layers in a neural network, CNNs can effectively extract higher level features such as facial traits from images. The convolutional layers dominate the computational cost in CNNs due to the multiply-and-accumulate (MAC) operation needed in these layers. In state-of-the-art networks, the number of MACs required within the convolutional layers spans from 666 million to 15.3 billion[4]. Thus, significant computational resources are required to perform training and inference using these networks. Thus, design of an efficient MAC engine is critical to efficient CNNs. Various approaches have been proposed to design an efficient MAC engine. Digital approaches to optimize dataflow while supporting reconfigurable kernel sizes[5] have been tried but suffer from large Cdyn when they are designed to support multi-bit precision, limiting power constrained applications. Analog voltage approaches have also been proposed to compactly represent signals on a single wire. These architectures also leverage in-memory techniques to minimize memory access cost. For example, one approach[6] writes input features to a bit-line using a digital-to-analog converter (DAC) and uses an adapted 10T bit-cell to control charging of local capacitors for multiplication. This operation is performed in parallel across a row of SRAM cells, which are then shorted together to perform averaging. Finally, an analog-to-digital converter (ADC) is used to retrieve the full sum. However, the finite voltage headroom and sensitivity of ADC and DAC limit the dynamic range and hence the supported input precision, reducing the accuracy of the network. Time domain multiplication using gated ring oscillators (GROs) for accumulation has been proposed as well[7,8,9]. These approaches represent signals in either pulse-width or time difference between edges. The pulse widths are multiplied by the weights then summed together before being quantized. This approach allows for compact representation of multi-bit information on a single wire, reducing Cdyn. However, due to the serial nature of these pulses, larger sums require larger periods to accumulate. To alleviate this, this paper proposes a time accumulator that is able to accumulate and quantize two signals simultaneously, achieving a 2x reduction in the period required to accumulate a given sum. This accumulator accepts pulses of varying widths from a two-step digital-to-time converter (DTC)[7], which accepts input feature values and encodes them in pulse-width. This paper is organized as follows. Section II discusses the background time domain processing and previous approaches. Section III presents the proposed time-domain architecture and dataflow. Section IV discusses simulation results. Section V concludes this paper and includes ideas for future work and avenues for improvement.

2 | TIME DOMAIN MAC

Time domain processing has shown promise in solving the problems present in voltage domain processing[7,8,9,10]. In the time domain, inputs can be encoded compactly on a single wire as a pulse-width modulated signal. As a result, Cdyn is reduced, resulting in reduced power consumption. Additionally, supply voltage can be reduced to near-threshold operation without reducing input precision, further reducing power consumption. To support more input bits, maximum pulse-width can be increased, resulting in reduced throughput. However, depending on the application, this could be acceptable. To perform a MAC operation in time domain, the digital input is encoded in a pulse-width modulated signal before being gated by the weight bits. A time domain MAC with binary weights is shown in Figure 3 (a). The gating operation can be performed using a simple AND gate, and corresponds to multiplication, where allowing the pulse to pass corresponds to multiplying by 1, and zeroing the pulse corresponds to multiplying by 0. The pulses are then summed together, corresponding to a long pulse, which can then be quantized.

While the binary multiplication in Figure 3 (b) is straightforward and can be accomplished by an AND gate[8] or multiplexer[10], the time accumulation is more complex and requires a circuit capable of summing multiple pulses. Previously demonstrated approaches to perform time accumulaion include a scheme[11] where a combination of comparators and current starving inverters

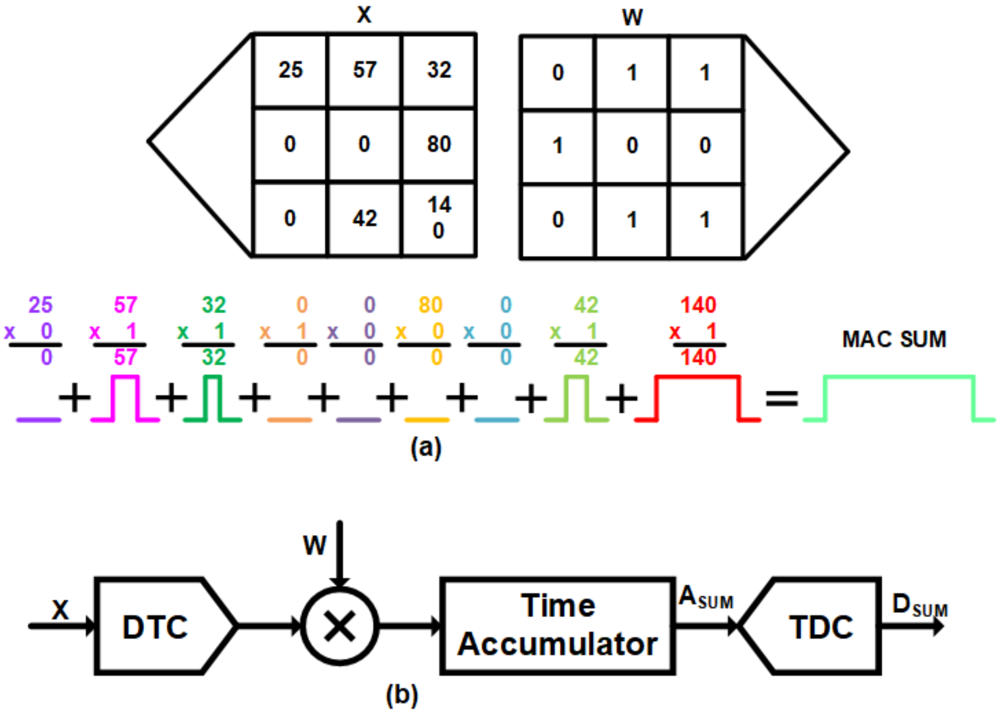


FIGURE 3 Time domain MAC (a) multiply-accumulate operation with binary weights and (b) conceptual schematic with digital input and output.

are used to sum input pulses in voltage domain. This approach performs accummulation with good accuracy but suffers from high power consumption due to the use of voltage comparators. Another approach[10] employs a cascade of delay cell stages with current starved inverters to perform accumulation. Interstage muliplexers propagate the polarity of the sum or invert it to perform multiplication by (+1) or (-1). This approach works well for binarized neural networks because there is no loss in accuracy as long as the polarity of the final sum is correct. When proportional accumulated results are needed, this approach will add significant nonlinearity due to delay variations and loss down the delay cell chain. Thus, there will be a limit on the number of terms that could be accumulated without impacting accuracy. This paper focuses on more digital control of pulse-widths to maintain linearity, as the accumulation of many terms results in significant loss due to nonlinearity.

To implement a more linear and power efficient time accummulation, the authors of the GRO-based MACs[7,8] proposed using a bi-directional gated ring oscillator (GRO), which is a modification of a previously presented time-to-digital converter (TDC)[12]. The GRO was repurposed to perform accumulation of pulses based on the observation that a string of tri-state inverters can function as a time-register, enabling time storage[13]. Figure 4 shows a GRO-based time accummulator with a follow-on counter to perform the quantization of the MAC output. It consists of an odd numbered ring of tri-state inverters. Assume that the nodes are initialized at 0 with the exception of the first node which is 1. When an input signal is asserted on the enable signal of the inverters, the ring oscillates and advances the phase along the ring as shown in Figure 4 . As the input falls low, the nodes are frozen. When the phase reaches the end of the ring, the input to the counter toggles from low to high, incrementing the output.

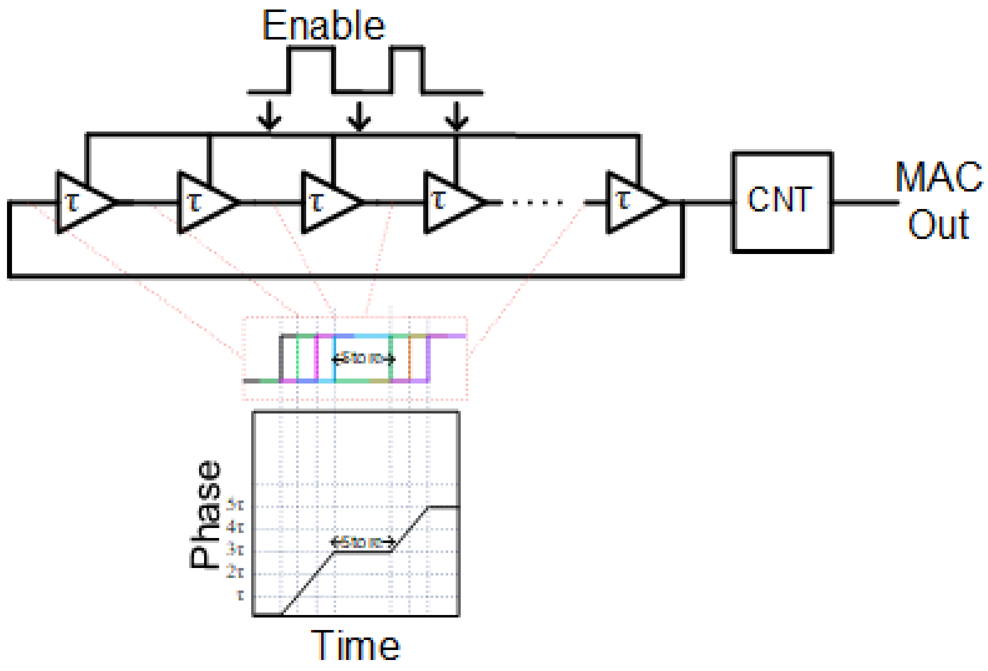


FIGURE 4 GRO-based time accumulator showing phase advancement.

To obtain the final result, the counter value should be scaled by the number of nodes in the ring. Thus, this structure performs accumulation of pulse widths.

The well-defined digital control pulses, with modulated pulse widths, provide the gating signals for the GRO. Thus, time accumulation is achieved serially with high linearity. The GRO architecture also ensures long series of time accummulations can be performed without explosion of circuit area or power consumption. However, the challenge presented by this approach is the computation time needed to complete the accummulation. The maximum clock frequency that the MAC can run is limited by the time needed to serially accummulate the widest pulses, leading to lower throughput. Thus, a MAC architecture that maintains the linearity advantages of the GRO-based MAC but improves the throughput is needed.

3 | PROPOSED SWITCHED-RING OSCILLATOR (SRO) MAC

To speed up the computation time of the MAC without significantly degrading its area and power efficiency, a digital concurrent accumulation is proposed. This will require modifying the GRO from ON/OFF operation to multi-rate operation similar to the switched-ring oscillator (SRO)[14]. The proposed MAC uses digitally controlled SRO (DCSRO) to ensure linearly variable frequency of oscillation. A 2-speed realization of this concept is demonstrated in this paper, as shown in Figure 5 . The enable signal (En) is turned on when either of the two modulated time inputs are high. However, unlike the GRO, a separate control is added to detect the number of inputs that are simultaneously high. This gain control signal (Gn) selects between two paths for

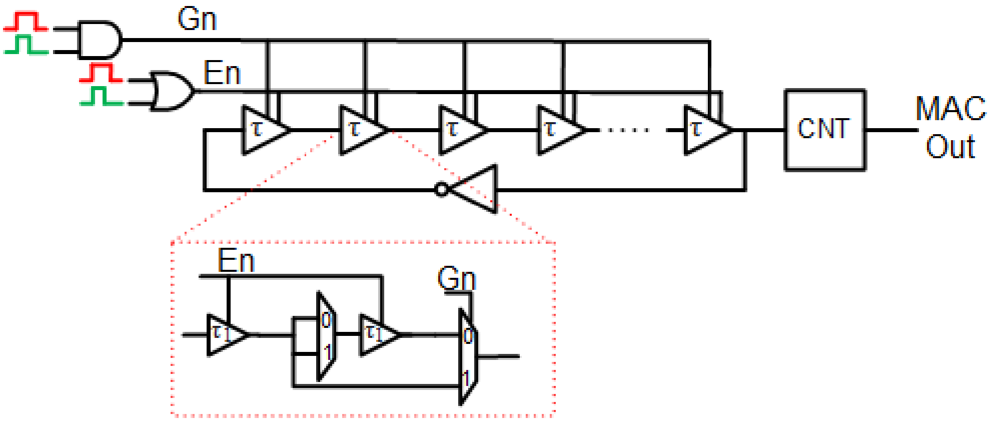


FIGURE 5 The DCSRO MAC showing the 2-speed delay unit.

the delay cell. This modulates the delay such that

$$\tau = (\tau_1 + \tau_{mux}) + (1 - G_n)(\tau_1 + \tau_{mux}). \tag{2}$$

where τ_1 and τ_{mux} are the delays of the sub delay cell and the multiplexer, respectively. To ensure proper ring polarity regardless of ring speed, buffers instead of inverters are used for the delay cells. This limits the minimum delay of a cell, assuming the inverter delay of τ_{inv}, to

$$\tau_{min} = 2\tau_{inv} + \tau_{mux}. \tag{3}$$

However, this limitation will have minimal impact of speed of operations given that the minimum delay will be in the order of a few 10s of picoseconds. This is expected to be reduced significantly with each process technology node advancement. The other consequence of using buffers for delay cells is that odd number of inverters in the ring is not achievable with delay cells alone. To address this, a solitary inverter is added to complete the ring to ensure oscillation. The impact of this inverter on the frequency of oscillation is suppressed by the fact that the loop usually has large number of delay cells. For our design, the loop contains 32 non-inverting delay cells.

Continuing with the theme of throughput enhancement while maintaining the energy efficiency of the proposed time domain MAC, a DTC design that leverages the concurrent accumulation was conceived. The delay line DTC approach of Figure 6 allows one to share the delay across multiple inputs, thus amortizing the area and power of the delay line. A clock is input to the delay line, which is then tapped and ANDed with the inverse of the signal. The output is a window corresponding to the overlap of the two signals with width $n\tau$, with n being the number of delay elements to the tap, and τ is the cell delay. The output is multiplexed and selected by the input feature performing the encoding. This DTC architecture allows the scaling of the number of concurrent inputs accumulated without significant area and power overhead. Moreover, the delay cells of the delay can be realized with the same buffers employed in the DCSRO to ensure the stable gain over process, voltage and temperature (PVT)

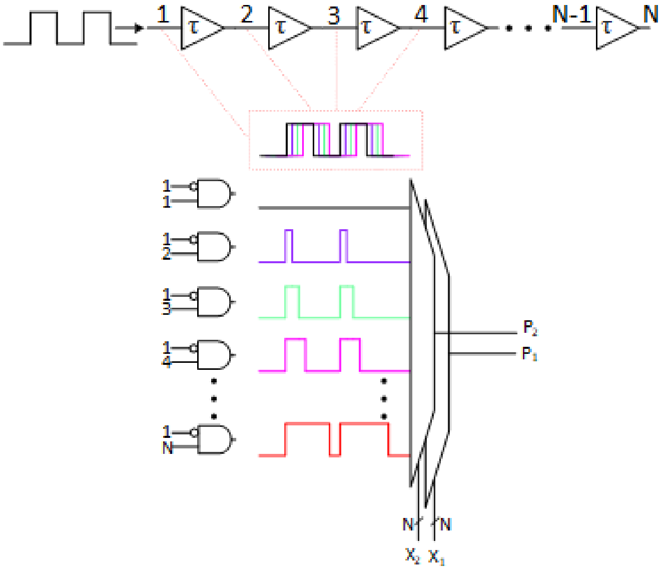


FIGURE 6 Delay line DTC that allows the sharing of pulse generators.

variations.

$$Gain = \frac{\tau_{DTC}}{\tau_{SRO}}. \tag{4}$$

To verify the operation of the DCSRO time accumulator, two input pulses are applied to the circuit. The oscillations along the internal nodes within the DCSRO are shown in Figure 7 . As both inputs are held high, the phase can be seen advancing corresponding to the dense rising and falling edges. As one of the inputs falls low and the other remains high, the phase can be seen advancing with longer delay. The shorter delay is approximately 34.9ps, and the longer delay is approximately 70.3ps. Thus, the phase advancement when both pulses are on is 0.496 times as fast as when only one pulse is on, giving a 0.85% non-linearity effect in schematic. As both inputs are low, the ring is frozen and there is no phase advancement, freezing counter. The glitches along the output as one input falls was filtered after post layout parasitics are added and will have minimal impact on the convolution output.

The other consideration is integrating bipolar operations to the DCSRO MAC. Both GRO-based MACs[7,8] used bidirectional delay cells to avoid a dedicated subtraction circuit. Given the focus on throughput enhancement, we opted to use two DCSRO stages for accumulating different polarities independently, then subtracting the results at the end. The circuit diagram of the complete MAC architecture is shown in Figure 8 . Two banks of the DCSROs of Figure 5 are used to accumulate time inputs of different polarities. Binary weights are used in this architecture to simplify the dataflow with little impact on accuracy[15]. A de-multiplexer performs the dual task of weight multiplication and polarity sorting.

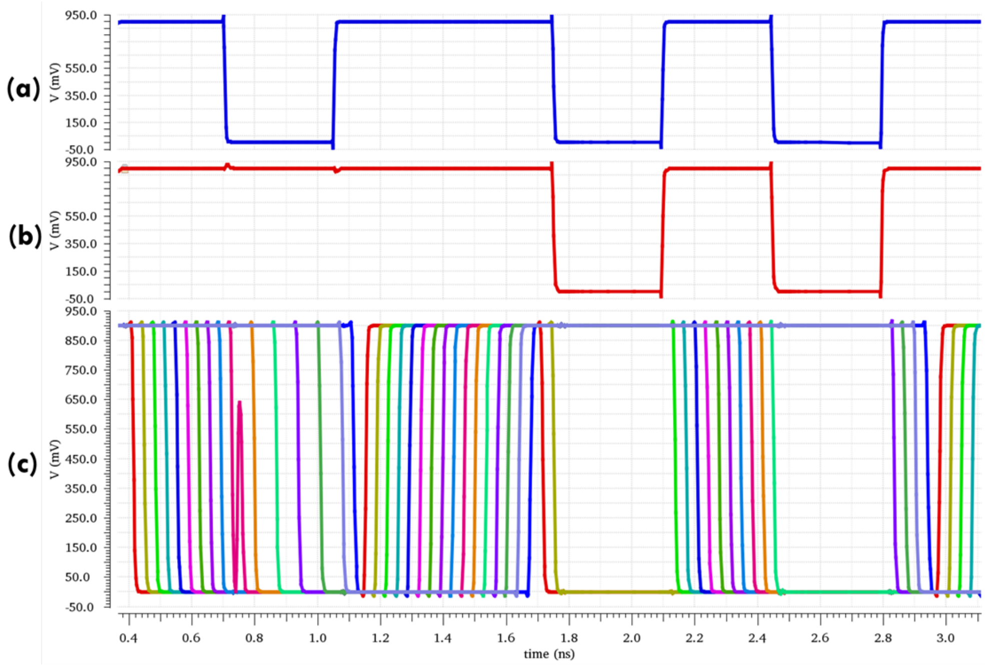


FIGURE 7 The DCSRO MAC DCGRO input and oscillations: a) Input pulse 1, b) Input pulse 2, c) Oscillations along the ring.

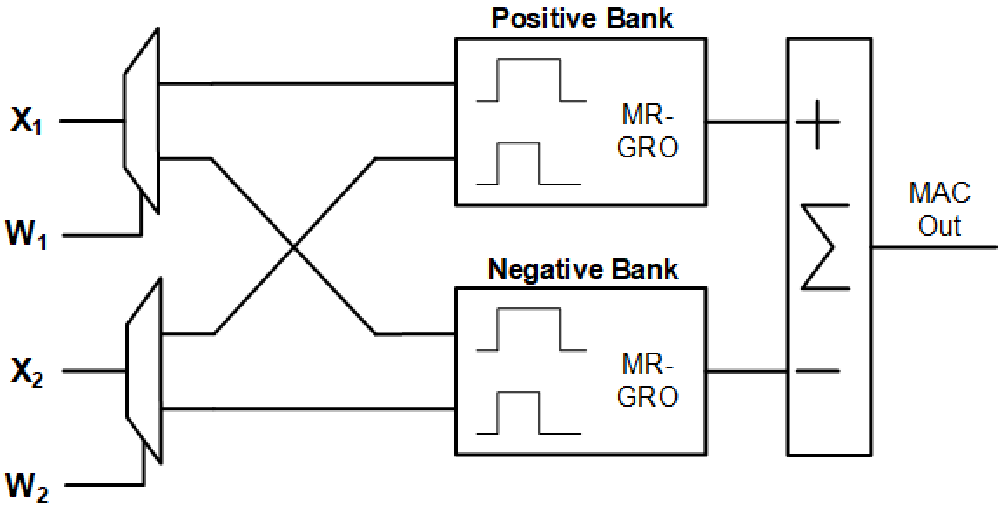


FIGURE 8 The DCSRO MAC showing bipolar realization.

4 | SIMULATION RESULTS

The testbench of Figure 9 was used to characterize the performance of the proposed DCSRO. First, a 5bit realization of the DTC of Figure 6 was used to generate the dual input pulses at 400MHz clock frequency. Simulations in a 28nm CMOS process yielded a DTC with -0.23LSB/+0.07LSB differential nonlinearity (DNL) and 0.50LSB/+0.02LSB integral nonlinearity (INL). The least significant bit (LSB) is calculated as max pulse width divided by 32 for the 5bit design. With positive weights, a

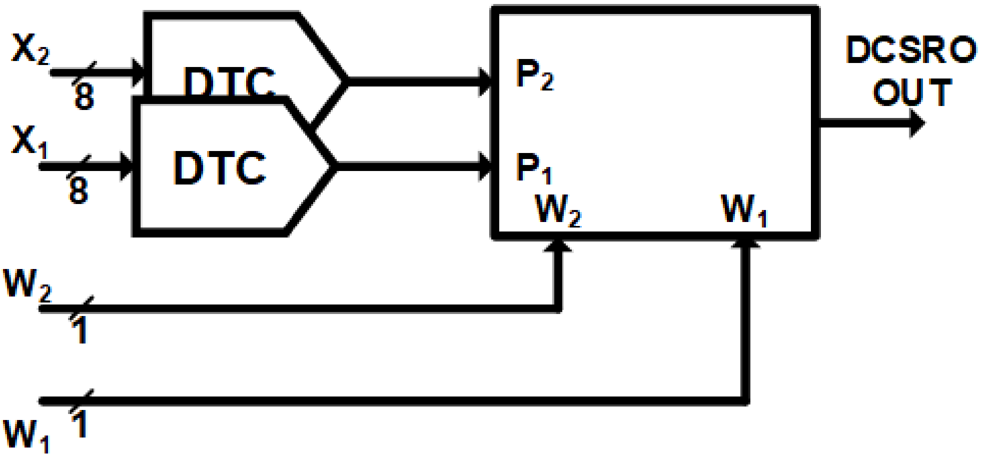


FIGURE 9 Simulation testbench to characterize the DCSRO MAC.

set of inputs were chosen such that the expected final accumulation result is 1984. The output of the counter at the end of the convolution is measured to be 29, corresponding to a value of 1856 after scaling, resulting in a compression of 7%.

To separate the DCSRO performance from the nonlinearity of the DTC, an 8b DTC based on a two-step architecture[7] was designed. Two copies were used to generate the two concurrent inputs for accumulation, as shown in Figure 9 . The DNL and INL of the 8bit DTC was simulated to be -0.32LSB/+0.09LSB and -0.21LSB/+0.50LSB, respectively as can be seen in Figure 10 . It instructive to note that while the magnitudes of the DNL and INL are close in LSBs, the LSB of the 8bit design is 8 times smaller. This reduces the amount of input nonlinearity that can be accumulated by the DCSRO MAC.

Using the 8bit DTC, the DCSRO was tested with three different digital inputs, and the results included in Figure 11 . The clock frequency used was 50MHz. In the first test, one of the digital inputs of the MAC was tied to 0, and the other was swept from a decimal value of 0 to 99. The DCSRO output shown in Figure 11 (a) was compared to an ideal accumulator built using VerilogA. The result after 2µs of simulation time is a decimal value of 40, compared to the ideal expected result of 38.7. The second test has both the DCSRO inputs ramped together from codes 0 to 99. The result after 1µs simulation time stayed at 40, as shown in Figure 11 (b). This indicates no performance degradation from the proposed concurrent accumulation. To confirm this conclusion, a final test was run with one input ramping from 0 to 99, and the other going the other direction from 99 to 0. The output in Figure 11 (c) settled to same result.

Power consumption was measured to compare the efficiency of the proposed 2-input DCSRO with a single input GRO equivalent assuming the same input clock frequency of 50MHz. The power consumption of 210.6μW, 14.4μW and 4.5μW for the DTC, GRO and counter, respectively for the single input case. This yields a total power of 229.5μW and an efficiency of 0.574pJ/b at 8bit input and 50MHz. The power consumption of 225μW, 26.1μW and 4.5μW for the DTC, DCSRO and counter, respectively for the two-input case, respectively.

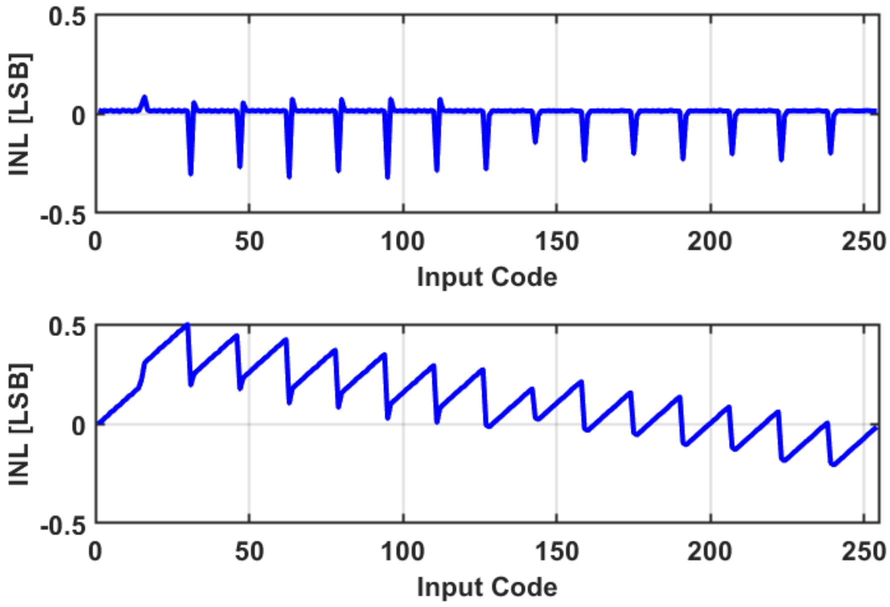


FIGURE 10 Simulation results showing the DNL and INL of an 8bit DTC used to characterize the performance of the proposed DCSRO MAC.

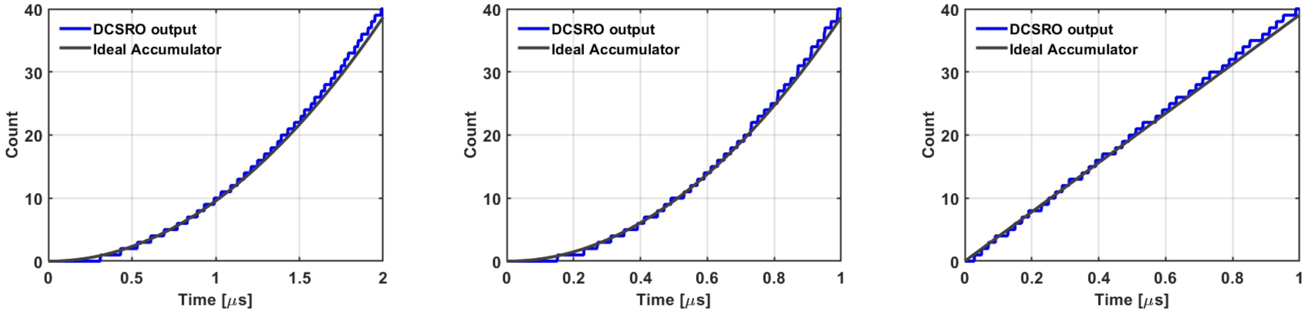


FIGURE 11 Simulation results showing accumulation accuracy of the proposed DCSRO MAC with (a) one input ramp, (b) two unidirectional input ramps and (c) two input ramps going the opposite directions.

5 | CONCLUSION

In this article, a time domain MAC engine with near-memory weight storage is demonstrated. A digitally controlled switched-ring oscillator is proposed to parallelize the quantization and accumulation of time domain inputs. To accumulate signed weights, two DCSRO banks are used, and the weight multiplication controls the switching between them. The DCSRO is simulated to have under 1% mismatch between the two supported frequencies, allowing for linear accumulation of two pulses, achieving 2x throughput compared to previous throughput of the GRO alone. The overall computational sum is shown to be compressed by 7% for a 5bit DTC. This reduces to 3.3% with an 8bit DTC. More importantly, the compression was confirmed to not be related to the concurrent accumulation. Simulation results show a power efficiency of 0.320pJ/b which is better than a simulated efficiency of 0.574pJ/b of the single input GRO

5.1 | Acknowledgements

The authors would like to thank the Intel Corporation Circuits SRS for providing funding for this work.

References

1. Agrawal A, Jaiswal A, Lee C, Roy K. X-SRAM: Enabling In-Memory Boolean Computations in CMOS Static Random Access Memories. *IEEE Transactions on Circuits and Systems I: Regular Papers* 2018; 65(12): 4219-4232. doi: 10.1109/TCSI.2018.2848999

2. Jhang CJ, Xue CX, Hung JM, Chang FC, Chang MF. Challenges and Trends of SRAM-Based Computing-In-Memory for AI Edge Devices. *IEEE Transactions on Circuits and Systems I: Regular Papers* 2021; 68(5): 1773-1786. doi: 10.1109/TCSI.2021.3064189

3. Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 1998; 86(11): 2278-2324. doi: 10.1109/5.726791

4. Sze V, Chen YH, Yang TJ, Emer JS. Efficient Processing of Deep Neural Networks: A Tutorial and Survey. *Proceedings of the IEEE* 2017; 105(12): 2295-2329. doi: 10.1109/JPROC.2017.2761740

5. Chen YH, Krishna T, Emer JS, Sze V. Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks. *IEEE Journal of Solid-State Circuits* 2017; 52(1): 127-138. doi: 10.1109/JSSC.2016.2616357

6. Biswas A, Chandrakasan AP. CONV-SRAM: An Energy-Efficient SRAM With In-Memory Dot-Product Computation for Low-Power Convolutional Neural Networks. *IEEE Journal of Solid-State Circuits* 2019; 54(1): 217-230. doi: 10.1109/JSSC.2018.2880918

7. Toyama Y, Yoshioka K, Ban K, Maya S, Sai A, Onizuka K. An 8 Bit 12.4 TOPS/W Phase-Domain MAC Circuit for Energy-Constrained Deep Learning Accelerators. *IEEE Journal of Solid-State Circuits* 2019; 54(10): 2730-2742. doi: 10.1109/JSSC.2019.2926649

8. Sayal A, Nibhanupudi SST, Fathima S, Kulkarni JP. A 12.08-TOPS/W All-Digital Time-Domain CNN Engine Using Bi-Directional Memory Delay Lines for Energy Efficient Edge Computing. *IEEE Journal of Solid-State Circuits* 2020; 55(1): 60-75. doi: 10.1109/JSSC.2019.2939888

9. He Y, Choi M, Kim KK, Kim YB. A Time-Domain Computing-In-Memory Micro using Ring Oscillator. In: IEEE. ; 2021: 107-108

10. Miyashita D, Kousai S, Suzuki T, Deguchi J. A Neuromorphic Chip Optimized for Deep Learning and CMOS Technology With Time-Domain Analog and Digital Mixed-Signal Processing. *IEEE Journal of Solid-State Circuits* 2017; 52(10): 2679-2689. doi: 10.1109/JSSC.2017.2712626

11. Lee D, Lee D, Lee T, Kim YH, Kim LS. An integrated time register and arithmetic circuit with combined operation for time-domain signal processing. In: IEEE. ; 2015: 1830-1833

12. Hsu CM, Straayer MZ, Perrott MH. A Low-Noise Wide-BW 3.6-GHz Digital ΔΣ Fractional-N Frequency Synthesizer With a Noise-Shaping Time-to-Digital Converter and Quantization Noise Cancellation. *IEEE Journal of Solid-State Circuits* 2008; 43(12): 2776-2786. doi: 10.1109/JSSC.2008.2005704

13. Kim K, Yu W, Cho S. A 9b, 1.12ps resolution 2.5b/stage pipelined time-to-digital converter in 65nm CMOS using time-register. In: IEEE. ; 2013: C136-C137.

14. Elshazly A, Rao S, Young B, Hanumolu PK. A Noise-Shaping Time-to-Digital Converter Using Switched-Ring Oscillators—Analysis, Design, and Measurement Techniques. *IEEE Journal of Solid-State Circuits* 2014; 49(5): 1184-1197. doi: 10.1109/JSSC.2014.2305651

15. Courbariaux M, Bengio Y, David JP. BinaryConnect: Training Deep Neural Networks with Binary Weights during Propagations. In: NIPS'15. ACM. MIT Press; 2015; Cambridge, MA, USA: 3123–3131.

□