XOR-CiM: An Efficient Computing-in-SOT-MRAM Design for Binary Neural Network Acceleration

Mehrdad Morsali†, Ranyang Zhou†, Sepehr Tabrizchi‡, Arman Roohi‡ and Shaahin Angizi†
†Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102, USA
‡School of Computing, University of Nebraska–Lincoln, Lincoln NE, USA
mm2772@njit.edu, aroohi@unl.edu, shaahin.angizi@njit.edu

2023 24th International Symposium on Quality Electronic Design (ISQED) | 979-8-3503-3475-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISQED57927.2023.10129322

***Abstract*—In this work, we leverage the uni-polar switching behavior of Spin-Orbit Torque Magnetic Random Access Memory (SOT-MRAM) to develop an efficient digital Computing-in-Memory (CiM) platform named XOR-CiM. XOR-CiM converts typical MRAM sub-arrays to massively parallel computational cores with ultra-high bandwidth, greatly reducing energy consumption dealing with convolutional layers and accelerating X(N)OR-intensive Binary Neural Networks (BNNs) inference. With a similar inference accuracy to digital CiMs, XOR-CiM achieves ∼4.5× and 1.8× higher energy-efficiency and speed-up compared to the recent MRAM-based CiM platforms.**

***Index Terms*—SOT-MRAM, computing-in-memory, binary neural networks**

I. Introduction

Binary Neural Networks (BNNs) have been developed as a solution to eliminating the need for massive Multiply-ACcumulate (MAC) operations and memory usage of Convolutional Neural Networks (CNNs) by forcing the inputs/weights/gradients to be binary specifically at the forward propagation level. BinaryConnect [1] uses binary weights to train deep neural networks on MNIST and CIFAR-10 data sets, with near state-of-the-art results. In BinaryNet [2], weights and activations are binarized as extensions to BinaryConnect. The XNOR-NET [3] algorithm offers a simple and accurate solution for large-scale data-sets and produces almost identical results as AlexNet's full-precision results.

From the hardware design point of view, the isolated processing and memory units connected via data buses, in the von-Neumann architecture, impose many challenging problems such as long memory access delay, the limited bandwidth of the memory, significant congestion at I/Os, massive data communication energy, and huge leakage current power consumption for storing network data in customary volatile memory [4], [5]. To address these concerns, Computing-in-Memory (CiM) CNN accelerators, as a potentially viable way to address the so-called memory wall challenge, have been widely explored [4], [6]–[8]. The main idea of CiM is to embed logic units within memory to process data by leveraging the inherent parallel computing mechanisms and exploiting large internal memory bandwidth. It could lead to remarkable savings in off-chip data communication energy and latency. An ideal CiM architecture should be capable of performing bulk bit-wise operations used in a wide spectrum of applications [8], [9]. The CiM architectures have recently become even more popular when integrating with emerging Non-Volatile Memory (NVM) technologies. Spin-Orbit Torque Magnetic Random Access Memory (SOT-MRAM) [5] is one of the most promising NVMs offering non-volatility, low switching energy, superior endurance, excellent retention time, and high integration density. IMCE [5], CMP-PIM [7], and GraphS [10] leverage bi-polar switching SOT-MRAM designs. They can activate two or more memory rows storing weights and inputs and execute a reduced-cycle and parallel X(N)OR operations on the Bit-lines required in BNNs through modified sense amplifiers. However, storing input feature maps of BNNs on-chip could impose extra write power that can be avoided by employing recent design methods such as [11], [12].

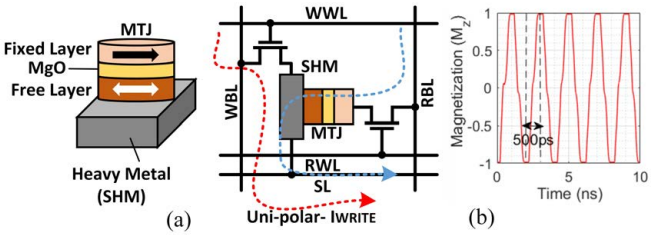

Fig. 1. (a) SOT-MTJ device and SOT-MRAM bit-cell with uni-polar switching, (b) Fast and consecutive uni-polar switching of SOT-MRAM.

In this work, we present an efficient CiM platform named XOR-CiM. XOR-CiM converts typical SOT-MRAM sub-arrays based on a uni-polar switching mechanism to massively parallel computational cores to accelerate X(N)OR-intensive applications. The main contributions of this work are listed here. (i) We propose a novel fast in-memory X(N)OR mechanism based on uni-polar switching SOT-MRAM; (ii) We develop micro-architecture and circuits required to convert every SOT-MRAM array to a potential computational core; (iii) We take BNN as a potential application and show how XOR-CiM can process such networks in parallel computational sub-arrays.

II. SOT-MTJ with Uni-polar Switching

SOT-MTJ device as shown in Fig. 1(a) is a composite structure of Spin Hall Metal (SHM) and Magnetic Tunnel Junction (MTJ). The resistance of MTJ with parallel magnetization in both magnetic layers (data-'1') is lower than that of MTJ with anti-parallel magnetization (data-'0'). Each SOT-MRAM cell located in our proposed computational sub-arrays is a SOT-MTJ associated with the Write Word Line (WWL), Read Word

Line (RWL), Write Bit Line (WBL), Read Bit Line (RBL), and Source Line (SL) as shown in Fig. 1(a). The deterministic switching of perpendicular SOT-MTJ requires an external in-plane magnetic field that hampers its application for a scalable SOT-MRAM design [13]. To mitigate this issue, state-of-the-art designs adopted in-plane SOT-MTJ structures to avoid random switching in perpendicular SOT-MTJ [14] or used Spin Transfer Torque (STT) assisted SOT switching methods [15]. Recently, field-free, deterministic, and high-speed SOT-MTJ switching mechanisms have been proposed leveraging uni-polar switching current [13], [16]. The magnetization dynamics of the MTJ's free layer under such a SOT can be given by the following Landau–Lifshitz–Gilbert (LLG) equation [16], [17]:

$$\frac{\partial \vec{m}}{\partial t} = -\gamma\mu_0 \vec{m}\times\overrightarrow{H_{eff}} + \alpha\vec{m}\times\frac{\partial \vec{m}}{\partial t} - \lambda_{DL}\xi J\vec{m}\times(\vec{m}\times\vec{\sigma}) - \lambda_{FL}\xi J\vec{m}\times\vec{\sigma} \quad (1)$$

where $\vec{m}$ denotes the unit vector of the free layer, γ and μ_0 are the gyromagnetic ratio and the vacuum permeability, respectively. $\overrightarrow{H_{eff}}$ represents the effective field and α the is Gilbert damping constant. Here, λ_{DL} and λ_{FL} represent the strengths of the damping-like and field-like torque, respectively. $\vec{\sigma}$ is the unit vector of the SOT-induced spin polarization. J is the current density and ξ is device-dependent parameter. It is experimentally shown that with a proper $\lambda_{FL}/\lambda_{DL}$ ratio, the uni-polar SOT-MTJ switching can be achieved. Therefore magnetization direction of the free layer can be periodically changed regardless of the polarity of the charge current flowing through the SHM [13], [18], [19]. Our macrospin simulation result in Fig. 1(b) shows how a fast (<1 ns) and consecutive (10 write operations) uni-polar switching of SOT-MRAM can be achieved.

III. Proposed XOR-CiM Design

A. Overview

XOR-CiM is developed as a high-performance and energy-efficient accelerator for X(N)OR-intensive applications such as BNNs [3]. The overall architecture of XOR-CiM is shown in Fig. 2(a) mainly consisting of the kernel storage array, result array, and a Computation Control Unit (CC). Each array is composed of 1024×512 SOT-MRAM cells connected to a voltage driver, row decoder, and sense amplifier unit. We propose to store only one of the repetitively-used operands, e.g., shared weights in the BNNs in the kernel storage while the second operand (activation) could be fed into to accelerator through the CC unit. With XOR-CiM, bit-wise convolutions in BNNs can be supported efficiently thanks to the following equation, which computes the dot-product of two vectors, *A* and *W* using *XNOR* and *Bitcount* as *A*⊛*W* = *BitCount(XNOR(A,W))*. Here ⊛ represents the binarized convolution using bit-wise logic and bit-count operations and activation-*A* and weight-*W* are vectors ∈{0,1}. A fast and parallel in-memory X(N)OR operation can be accomplished through a new mechanism discussed in the next subsection. After XNOR computation, to perform the Bit-Count task, the SOT-MRAM result array storing XNOR results will be read out. The output of sensory circuitry is then connected to a CMOS counter unit. In this way, the counter can count the number of ones in the resultant bit vector to generate the output feature maps.

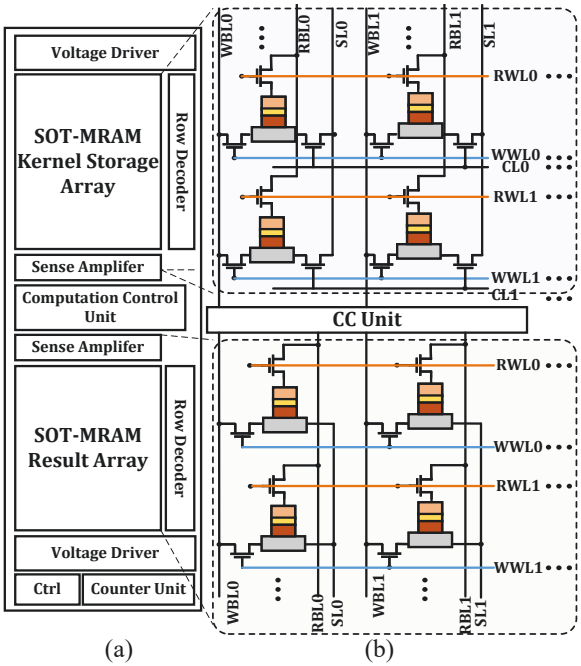

Fig. 2. (a) The proposed XOR-CiM architecture with (b) SOT-MRAM-based kernel storage and result storage arrays.

B. Fast In-memory X(N)OR Design

Leveraging ultra-fast deterministic uni-polar switching of the SOT-MRAM, a fast in-memory X(N)OR design is proposed herein inspired by [19], [20]. In uni-polar switching, each time a writing current with a large enough magnitude is applied to the SOT-MTJ through the SHM, regardless of the direction of the applied current, the magnetization switches periodically. With this observation, we realized that if the number of switching pulses is even, the MTJ's final state will be equal to its initial state. And, if the number of switching pulses is an odd number the MTJ would end up in a state opposite to its initial state. As the XOR function gives '1' output when the number of '1' inputs is odd, the uni-polar switching feature of SOT-MTJs can be exploited to implement this logic efficiently. The proposed X(N)OR circuit implementation is shown in Fig. 3(a). The core of our design is composed of two SOT-MRAMs and a CC unit part. The W-SOT-MRAM in the kernel storage array stores the BNN weight and the result of the X(N)OR operation is written at Y-SOT-MRAM in the result array. Hereafter, we assume the SOT-MRAM with an anti-parallel state (i.e., high-resistance), represents logic '0', and the SOT-MRAM with an parallel state represents logic '1' (i.e., low-resistance).

In XOR-CiM, the *W* and *A* are applied by the CC unit to the Y-SOT-MRAM as switching currents in two stages as indicated in Fig. 3(a). As a result, two back-to-back write operations are performed on Y-SOT-MRAM and the final state of the Y-SOT-

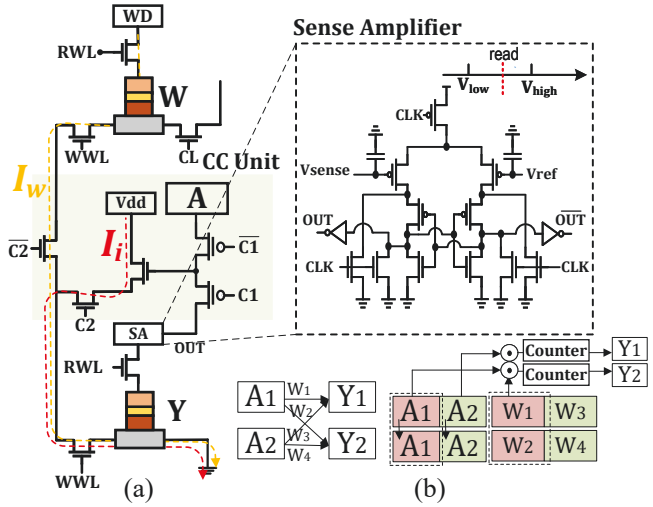

Fig. 3. (a) The proposed in-memory X(N)OR circuit implementation. Here, W, A, and Y represent weight, activation, and output, respectively. (b) Sample BNNs mapping technique.

TABLE I
PROPOSED XOR TRUTH TABLE

Weight	Activation	Y(t)	Y(t+t_0)	Y(t+2t_0)(XOR)
'0'	'0'	'0'	'0'	'0'
'0'	'1'	'0'	'0'	'1'
'1'	'0'	'0'	'1'	'1'
'1'	'1'	'0'	'1'	'0'

MRAM will be the XOR result. Considering the initial state of Y-SOT-MRAM as logic '0' (at t_0), in the first writing stage, the CC unit applies the weight current (I_w). If the W-SOT-MRAM stores logic '1' data with low resistance, the amplitude of the I_w is high enough to switch the Y-SOT-MRAM. Otherwise, the I_w current can't switch the Y-SOT-MRAM and at t+t_0, it remains in its initial state. In the second writing stage, the CC unit applies activation current (I_A) based on the *A*'s binary value. If the *A* is '1', a large enough I_A passes through Y-SOT-MRAM and switches its magnetization, otherwise if *A* data is '0', no writing current is applied by CC unit to Y-SOT-MRAM and the output remains unchanged at t+2t_0. After these two back-to-back writing stages, the Y-SOT-MRAM can be read using a pre-charged sense amplifier as shown in Fig. 3(a). Since SA generates data and its complementary logic at the same time, we can obtain both XOR and XNOR functions. The truth table of the proposed XOR scheme with timing considerations is illustrated in Table I. Based on Table I, in every writing cycle the *Y*, changes if the corresponding writing reference data is '1'. Thus, after two cycles, the result would be the XOR of the *W* and *A*. In the next section, more details will be provided about the X(N)OR operation. Fig. 3(b) gives a sample data mapping method for XOR-CiM where a one-layer neural network computation can be easily converted to a fully parallel scheme. As shown, activation can be applied in parallel through the CC unit to the kernel storage array to perform XNOR logic.

IV. PERFORMANCE EVALUATION

The XOR-CiM's memory sub-array organization has been configured with 1024 rows and 512 columns per mat organized in an H-tree routing manner, 2×2 mats per bank, 8×8 banks per group; in total 16 groups. At the device level, we jointly use the Non-Equilibrium Green's Function (NEGF) and LLG with spin Hall effect equations to model SOT-MRAM bit-cell [21], [22] with the device parameters listed in Table II. At the circuit level, a Verilog-A model of a uni-polar SOT-MRAM device is developed to co-simulate with the interface CMOS circuits at 45nm in SPICE. At the architectural-level, we modified the NVSim [23] to report performance for XOR-CiM operations with input from the device/circuit level results. At the application level, a behavioral-level simulator is developed in Matlab to calculate the latency and energy that XOR-CiM spends on BNNs with a mapping optimization framework to maximize the performance according to the available resources.

TABLE II
DEVICE PARAMETERS USED IN THE SIMULATION.

Symbol	Quantity	Values
α	Damping coefficient	0.3
θ_{SHM}	Spin Hall angle	0.3 [24]
H_k	Perpendicular magnetic anisotropy	$2.2 \times 10^5 A/m$
M_s	Saturation magnetization	$1 \times 10^6 A/m$
t_{MgO}	MgO thickness	0.8 nm
t_{sl}	Free layer thickness	1.2 nm
RA	MTJ Resistance area product	$10 \times 10^{-12} \Omega m^2$
TMR	Tunnel Magneto resistance	240%
ρ_{SHM}	Resistivity of SHM (W)	$200 \mu \Omega cm$ [25]
$(L.W)_{MTJ}$	MTJ dimention	$20 \times 20 nm^2$
$(L.W.t)_{SHM}$	SHM dimension	$60 \times 35 \times 0.5 nm^3$

A. Functionality Analysis

To evaluate the functionality of the XOR-CiM's X(N)OR operation, transient simulations for all four possible combinations of *W* and *A* are conducted. As mentioned above, for each XOR operation we need to have two back-to-back writes. Considering an extra writing step for the initialization, in our scheme, three writing steps are demanded. Each writing stage itself has two sub-stages, switching and relaxation times. The switching time was set to 200 ps in this work. After each switching, SOT-MRAM devices inherently take a relaxation time to continue changing their magnetization completely, to be relaxed in a stable state [26]. If we apply another current to a SOT-MRAM before it reaches a stable condition, false data may be written on the MTJ. To prevent writing errors, we have set 300 ps as relaxation time after each switching. So, every complete writing step can be done in 500 ps (t_0) and the total time for finalizing an X(N)OR operation in our design is 1.5 ns. Utilizing a pre-charged sense amplifier for reading the result in 200 ps, the total time for an X(N)OR operation considering reading the result would be 1.7 ns. Sample waveform of the main controlling signals, the magnetization direction of the SOT-MRAM, and its corresponding output data for all four possible combinations are demonstrated in Fig. 4. As shown in Fig. 4, the initial state of the Y-SOT-MRAM is considered as the parallel state (Mz=1.0), which represents logic '1'. Therefore, at the initialization step which begins at 200 ps, the magnetization of Y-SOT-MRAM switches to an anti-parallel state (Mz=-1.0) representing logic '0' to get it ready for our

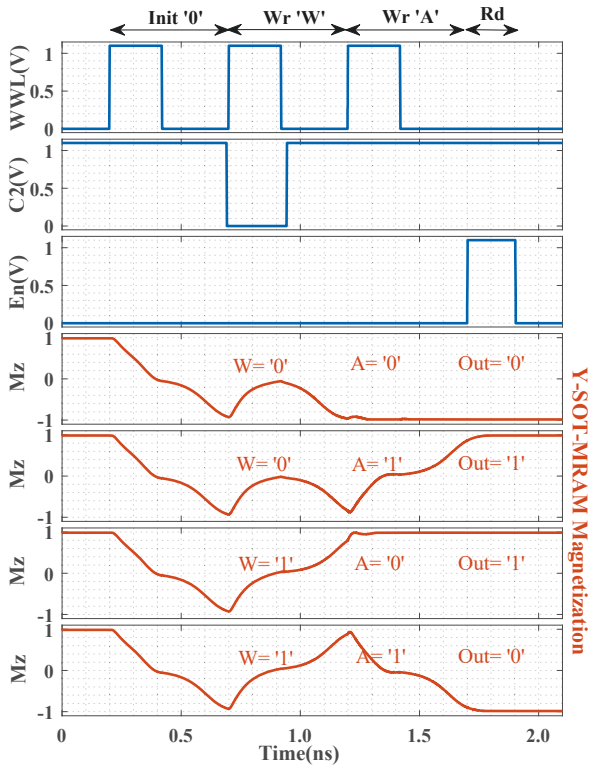

Fig. 4. Transient simulation waveforms.

XOR operation. After this step, the CC unit opens the path for the I_w by setting the C2 signal to '0'. If W='0', which means that the W is in an anti-parallel state (high-resistance), I_w can't switch the state of the Y-SOT-MRAM in 200 ps, and it remains in anti-parallel state after first XOR writing cycle. Otherwise, the Y-SOT-MRAM switches to a parallel state (Mz=1.0). Then, by setting C2 to '1', I_A passes through Y. If the A is '1', a current large enough will pass through the Y-SOT-MRAM and switches it. By setting the En signal to '1', corresponding output data can be read from Y. As discussed, XOR-CiM requires only 1.7 ns to execute X(N)OR operation. We simulated all four possible combinations of W and A and measured the power consumption of each possibility. XOR-CiM consumes on average 15.97μW power to perform X(N)OR. Considering 1.7 ns operation latency, the average energy consumption of our proposed design is 27.16 fJ.

B. BNN Acceleration

We compare XOR-CiM with other possible BNN acceleration solutions based on MRAM, ReRAM, and DRAM. Enlarging the chip area brings a higher performance for XOR-CiM and other designs due to the increased number of sub-arrays or computational units, though the die size directly impacts the chip cost. Therefore, to have a fair comparison, the normalized ISO-capacity results will be reported henceforth. ***MRAM:*** We developed an X(N)OR-friendly CMP-PIM-like CiM [7] with two-row activation and an MnM-like CiM [12] with one-row activation based on bi-polar SOT-MTJs to perform BNNs. ***DRAM:*** We developed Ambit- [27] and DRISA-like [6] accelerators for BNNs. Ambit leverages a triple-row activation mechanism to implement majority gate-based logic. An X(N)OR logic with this mechanism can take up to 7 cycles. This is due to inevitable row initializations and destructive operations in DRAM. As for DRISA, the 1T1C method with XOR add-on support has been selected for comparison. ***ReRAM:*** An MPIM-like [28] accelerator with 256 sub-arrays and one buffer sub-array per bank was considered for evaluation. For evaluation, NVSim simulator [23] was extensively modified to work with Design Compiler [29] to emulate MPIM functionality. Note that the default NVSim's ReRAM cell file (.cell) was adopted for the assessment.

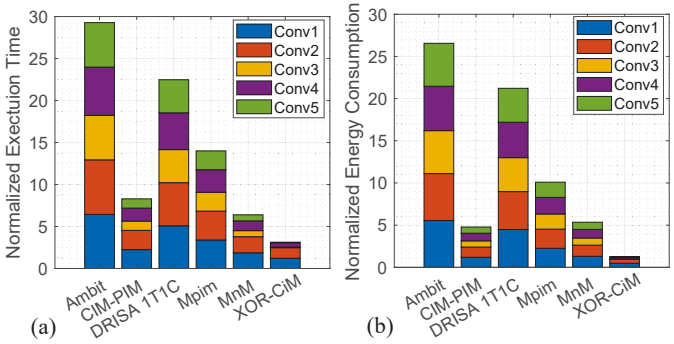

Fig. 5. (a) Normalized execution time, and (b) Energy consumption of various CiMs running AlexNet kernels.

Figure 5(a) reports the normalized execution time breakdown (based on convolutional layers) of XOR-CiM and the under-test CiM platforms running the binary-version AlexNET for SVHN data-set. We observe that XOR-CiM outperforms the fastest MRAM-based platform (MnM [12]) on average by ~1.8× speedup. XOR-CiM with reduced-cycle and intrinsic XNOR2 operation also achieves remarkable improvement over other CiM counterparts, for example, achieving ~5.7× speedup over DRISA [6]. Figure 5(b) reports the normalized energy consumption of XOR-CiM and various CiM platforms. We observe that XOR-CiM notably reduces the energy consumption for running X(N)OR-based operations compared with other CiM platforms. XOR-CiM obtains 4.5× energy saving over the most energy-efficient MRAM platform (i.e., CMP-PIM [7]). As compared with MPIM [28], our design shows ~5.2× reduction in energy consumption.

V. Conclusion

In this paper, we proposed an efficient computing-in-memory (CiM) platform named XOR-CiM. Utilizing the unipolar switching behavior of SOT-MRAM, the presented design converts typical MRAM sub-arrays to massively parallel computational cores with ultra-high bandwidth. This platform is capable of greatly reducing energy consumption dealing with convolutional layers and accelerating X(N)OR-intensive binary neural networks. Compared with the recent digital MRAM-based CiM accelerators, XOR-CiM achieves ~4.5× and 1.8× higher energy-efficiency and speed-up, With the same inference accuracy on SVHN data-set.

Acknowledgements

This work is supported in part by the National Science Foundation under Grant No. 2228028, 2216772, and 2216773.

REFERENCES

[1] M. Courbariaux, Y. Bengio, and J.-P. David, "Binaryconnect: Training deep neural networks with binary weights during propagations," in *Advances in Neural Information Processing Systems*, 2015, pp. 3123–3131.

[2] M. Courbariaux, I. Hubara, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks: Training deep neural networks with weights and activations constrained to +1 or-1," *arXiv:1602.02830*, 2016.

[3] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 525–542.

[4] P. Chi, S. Li, C. Xu, T. Zhang, J. Zhao, Y. Liu, Y. Wang, and Y. Xie, "Prime: A novel processing-in-memory architecture for neural network computation in reram-based main memory," in *ISCA*. IEEE Press, 2016.

[5] S. Angizi, Z. He, F. Parveen, and D. Fan, "Imce: energy-efficient bit-wise in-memory convolution engine for deep neural network," in *Proceedings of the 23rd ASP-DAC*. IEEE Press, 2018, pp. 111–116.

[6] S. Li, D. Niu, K. T. Malladi, H. Zheng, B. Brennan, and Y. Xie, "Drisa: A dram-based reconfigurable in-situ accelerator," in *Micro*. ACM, 2017, pp. 288–301.

[7] S. Angizi, Z. He, A. S. Rakin, and D. Fan, "Cmp-pim: an energy-efficient comparator-based processing-in-memory neural network accelerator," in *55th DAC*. ACM, 2018, p. 105.

[8] S. Li, C. Xu, Q. Zou, J. Zhao, Y. Lu, and Y. Xie, "Pinatubo: A processing-in-memory architecture for bulk bitwise operations in emerging non-volatile memories," in *DAC*. IEEE, 2016.

[9] S. Angizi, Z. He, A. Awad, and D. Fan, "Mrima: An mram-based in-memory accelerator," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 5, pp. 1123–1136, 2019.

[10] S. Angizi, J. Sun, W. Zhang, and D. Fan, "Graphs: A graph processing accelerator leveraging sot-mram," in *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2019, pp. 378–383.

[11] K. Kim, H. Shin, J. Sim, M. Kang, and L.-S. Kim, "An energy-efficient processing-in-memory architecture for long short term memory in spin orbit torque mram," in *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. IEEE, 2019, pp. 1–8.

[12] A. Sridharan, F. Zhang, and D. Fan, "Mnm: A fast and efficient min/max searching in mram," in *GLSVLSI 2022*, 2022, pp. 39–44.

[13] T. Yang, M. Yang, L. Zhao, J. Gao, Q. Xiang, W. Li, F. Luo, L. Ye, and J. Luo, "Field-free deterministic writing of spin-orbit torque magnetic tunneling junction by unipolar current," *IEEE Electron Device Letters*, vol. 43, pp. 709–712, 2022.

[14] K. C. Chun, H. Zhao, J. D. Harms, T.-H. Kim, J.-P. Wang, and C. H. Kim, "A scaling roadmap and performance evaluation of in-plane and perpendicular mtj based stt-mrams for high-density cache memory," *IEEE journal of solid-state circuits*, vol. 48, pp. 598–610, 2012.

[15] M. Wang, W. Cai, K. Cao, J. Zhou, J. Wrona, S. Peng, H. Yang, J. Wei, W. Kang, Y. Zhang, J. Langer, B. Ocker, A. Fert, and W. Zhao, "Current-induced magnetization switching in atom-thick tungsten engineered perpendicular magnetic tunnel junctions with large tunnel magnetoresistance," *Nature communications*, vol. 9, pp. 1–7, 2018.

[16] L. Jiang, E. Deng, H. Zhang, Z. Wang, W. Kang, and W. Zhao, "A spintronic in-memory computing network for efficient hamming codec implementation," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 4, pp. 2086–2090, 2022.

[17] M. Kazemi, G. E. Rowlands, E. Ipek, R. A. Buhrman, and E. G. Friedman, "Compact model for spin–orbit magnetic tunnel junctions," *IEEE TED*, vol. 63, pp. 848–855, 2016.

[18] Z. Wang, B. Wu, Z. Li, X. Lin, J. Yang, Y. Zhang, and W. Zhao, "Evaluation of ultrahigh-speed magnetic memories using field-free spin–orbit torque," *IEEE Transactions on Magnetics*, vol. 54, pp. 1–5, 2018.

[19] H. Zhu, B. Wu, K. Chen, C. Yan, and W. Liu, "High-performance in-memory logic scheme using unipolar switching sot-mram," in *2022 IEEE 22nd International Conference on Nanotechnology (NANO)*, 2022, pp. 112–115.

[20] M. Zabihi, Z. I. Chowdhury, Z. Zhao, U. R. Karpuzcu, J.-P. Wang, and S. S. Sapatnekar, "In-memory processing on the spintronic cram: From hardware design to application mapping," *IEEE Transactions on Computers*, vol. 68, no. 8, pp. 1159–1173, 2018.

[21] X. Fong, Y. Kim, K. Yogendra, D. Fan, A. Sengupta, A. Raghunathan, and K. Roy, "Spin-transfer torque devices for logic and memory: Prospects and perspectives," *IEEE TCAD*, vol. 35, 2016.

[22] Z. Wang, W. Zhao, E. Deng, J.-O. Klein, and C. Chappert, "Perpendicular-anisotropy magnetic tunnel junction switched by spin-hall-assisted spin-transfer torque," *Journal of Physics D: Applied Physics*, 2015.

[23] X. Dong, C. Xu, Y. Xie, and N. P. Jouppi, "Nvsim: A circuit-level performance, energy, and area model for emerging non-volatile memory," in *Emerging Memory Technologies*. Springer, 2014, pp. 15–50.

[24] C.-F. Pai, L. Liu, Y. Li, H. W. Tseng, D. C. Ralph, and R. A. Buhrman, "Spin transfer torque devices utilizing the giant spin hall effect of tungsten," *Applied Physics Letters*, 2012.

[25] Y. Kim, S. H. Choday, and K. Roy, "Dsh-mram: differential spin hall mram for on-chip memories," *IEEE Electron Device Letters*, vol. 34, no. 10, pp. 1259–1261, 2013.

[26] W. Legrand, R. Ramaswamy, R. Mishra, and H. Yang, "Coherent subnanosecond switching of perpendicular magnetization by the fieldlike spin-orbit torque without an external magnetic field," *Phys. Rev. Applied*, vol. 3, p. 064012, Jun 2015.

[27] V. Seshadri, D. Lee, T. Mullins, H. Hassan, A. Boroumand, J. Kim, M. A. Kozuch, O. Mutlu, P. B. Gibbons, and T. C. Mowry, "Ambit: In-memory accelerator for bulk bitwise operations using commodity dram technology," in *Micro*. ACM, 2017, pp. 273–287.

[28] M. Imani, Y. Kim, and T. Rosing, "Mpim: Multi-purpose in-memory processing using configurable resistive memory," in *ASP-DAC*. IEEE, 2017, pp. 757–763.

[29] S. D. C. P. V. . Synopsys, Inc.