Quantized Magnetic Domain Wall Synapse for Efficient Deep Neural Networks

Seema Dhull, *Graduate Student Member, IEEE*, Walid Al Misba, *Student Member, IEEE*, Arshid Nisar, *Graduate Student Member, IEEE*, Jayasimha Atulasimha, *Senior Member, IEEE,* Brajesh Kumar Kaushik*, Senior Member, IEEE*

***Abstract*— Quantization of synaptic weights using emerging non-volatile memory devices has emerged as a promising solution to implement computationally efficient neural networks on resource constrained hardware. However, the practical implementation of such synaptic weights is hampered by the imperfect memory characteristics, specifically the availability of limited number of quantized states and the presence of large intrinsic device variation and stochasticity involved in writing the synaptic states. This article presents on-chip training and inference of a neural network using quantized magnetic domain wall (DW) based synaptic array and CMOS peripheral circuits. A rigorous model of the magnetic DW device considering stochasticity and process variations has been utilized for the synapse. To achieve stable quantized weights, DW pinning has been achieved by means of physical constrictions. Finally, VGG8 architecture for CIFAR-10 image classification has been simulated by using the extracted synaptic device characteristics. The performance in terms of accuracy, energy, latency, and area consumption has been evaluated while considering the process variations and non-idealities in the DW device as well as the peripheral circuits. The proposed quantized neural network architecture achieves efficient on-chip learning with 92.4% and 90.4 % training and inference accuracy, respectively. In comparison to pure CMOS based design, it demonstrates an overall improvement in area, energy, and latency by 13.8×, 9.6×, and 3.5×, respectively.**

***Index Terms*— Magnetic domain wall, neuromorphic computing, quantized neural network, synapse, spin orbit torque.**

I. Introduction

Deep neural networks (DNNs) have proved to be successful in a number of applications ranging from image classification, speech recognition, time-series prediction, and spatiotemporal recognition tasks [1-2]. However, DNNs implemented in traditional von-Neumann computing platforms consume enormous energy [3] and incur high latency [4] because of separate memory and processing units as well as the need to shuttle data back and forth between these units. In-memory computing (IMC) [5-9] can obviate the need for data shuttling as the computation takes place in the memory itself and thus can be a solution for operating DNNs in resource constrained environments such as IoTs and edge devices. In IMC, non-volatile memory (NVM) devices are arranged in a crossbar array and the weights are mapped into the conductance states of the devices to perform the most intensive energy load of the DNN, the matrix vector multiplications (MVM), in a single time step [6, 9] using Kirchhoff's law. Towards this end, NVMs such as phase change memory (PCM) [7, 10-12], resistive random-access memory (RRAM) [13-15], ferroelectric memory (FeRAM) [16] and spintronic memory [17, 18] are shown to be area and energy-efficient solution for IMC and have been researched extensively. Among these devices, spintronic magnetic domain wall (DW) devices are very promising due to their higher energy efficiency, higher speed of operation, higher integration density, and CMOS compatibility [4, 18-24]. Such nanomagnetic memory devices can be controlled by voltage induced strain [25-31], voltage controlled magnetic anisotropy (VCMA) [32-34], current [35-36] and combination of voltage and current [37-39]. However, similar to other NVM devices, the DW devices suffer from non-linear, stochastic response and low resolution in the presence of practical constraints such as room temperature thermal noise, and lithographic imperfections [39, 40, 41]. Low resolution and stochastic responses of NVM devices are shown to significantly impact the DNN accuracy [42]. Several methods have been adopted in the literature to address these issues. Multiple devices per synapse have been used to increase the precision and address non-linearity of the conductance response [43-45]. Bit-slicing technique [46] is used to slice the input and weight matrices into several smaller bit slices. In [11], 3 transistors and 1 capacitor (3T1C) module is used to accumulate the smaller conductance update in the linear-operating region of the 3T1C and then periodically transfer them to the non-volatile PCM devices. However, with the methods mentioned above, all the weights are updated during the weight (conductance) update stage by sending overlapping pulses into the crossbar rows and columns, which results in low device endurance and high training energy cost. Recently, mixed precision training [47-48] is proposed where the energy intensive MVM and weight updates are performed in analog domain in an imprecise manner and the weight update calculations (computing the weight gradients) are performed in high precision memory units.

With the recent advent of quantized neural networks (QNNs) and its success in a number of neural network architectures such as fully connected neural networks (FCNNs) [49] and convolutional neural networks (CNNs) [50] for a range of tasks, quantization aware training can be performed to address the low-resolution issue of the DW devices. However, with QNN training, weight gradients need to be preserved in full precision to retain accuracy [49,50]. Thus, mixed precision framework is suitable for quantization-aware training, where the weight gradients are computed and preserved in full precision in a separate digital memory unit. Recent studies [52]

Seema Dhull (e-mail: seemadhull@ec.iitr.ac.in), Arshid Nisar (e-mail: arshid_l@ec.iitr.ac.in), and Brajesh Kumar Kaushik (e-mail: bkk23fec@iitr.ac.in) are with the Department of Electronics and Communication Engineering, Indian Institute of Technology Roorkee, Roorkee, Uttarakhand, 247667, India.

Walid Al Misba (e-mail: misbawa@vcu.edu), is with the Department of Mechanical and Nuclear Engineering, Virginia Commonwealth University, Richmond, VA 23284 USA.

Jayasimha Atulasimha (e-mail: jatulasimha@vcu.edu) is with the Department of Mechanical and Nuclear Engineering and the Department of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, VA 23284 USA.

Corresponding Author: Brajesh Kumar Kaushik

shows that such quantized training with extremely low resolution (less than 3-bit) and stochastic DW devices can achieve near equivalent accuracies to the floating- point precision (32-bit) FCNN with significant amount of energy savings. Large inter-state intervals among the limited states of the device conductances contribute to the significantly lower number of device updates and associated reduction in write energy. Although quantized training is shown for simpler FCNN, a more rigorous study of such quantized training with complex neural networks such as convolutional neural networks (CNNs) with appropriate hardware is lacking. Apart from device resolution, process variations and non-idealities in the peripheral circuitry can impact neural networks' accuracies. DW devices are typically accompanied with a magnetic tunnel junction (MTJ) to facilitate read operation that can be affected by process variation (i.e., tunnel barrier thickness) [53-54]. Moreover, peripheral circuits are used to convert analog read current sum to digital values (to be fed into the next layer neurons) and the digitally computed errors to analog values in forward and backward propagation, respectively. The low-resolution converters are desirable for peripheral operation as high-resolution ADC (or DAC) incur prohibitive area and energy cost and compromise the benefits of IMC [46,55]. However, decreasing the converter resolution introduces quantization loss during the read and write of the device and thus impacts the neural network accuracy. To obtain the efficient design of IMC framework without compromising accuracies, the impact of all the system-level non-idealities on the QNN needs to be quantified and assessed carefully from the standpoint of overall training cost. In our study, we perform system-level study for on-chip training of the quantized CNN by considering stochasticity in device operation, process variations, and non-idealities stemmed from the peripheral circuits. In addition, we demonstrate off-chip learning of the quantized CNN with stochastic DW devices, where a precursor CNN is first trained off-line separately, and the trained weights are transferred to the crossbar arrays to perform the actual inference.

The rest of the paper is organized as follows. Section II presents the micromagnetic and circuit implementation of the

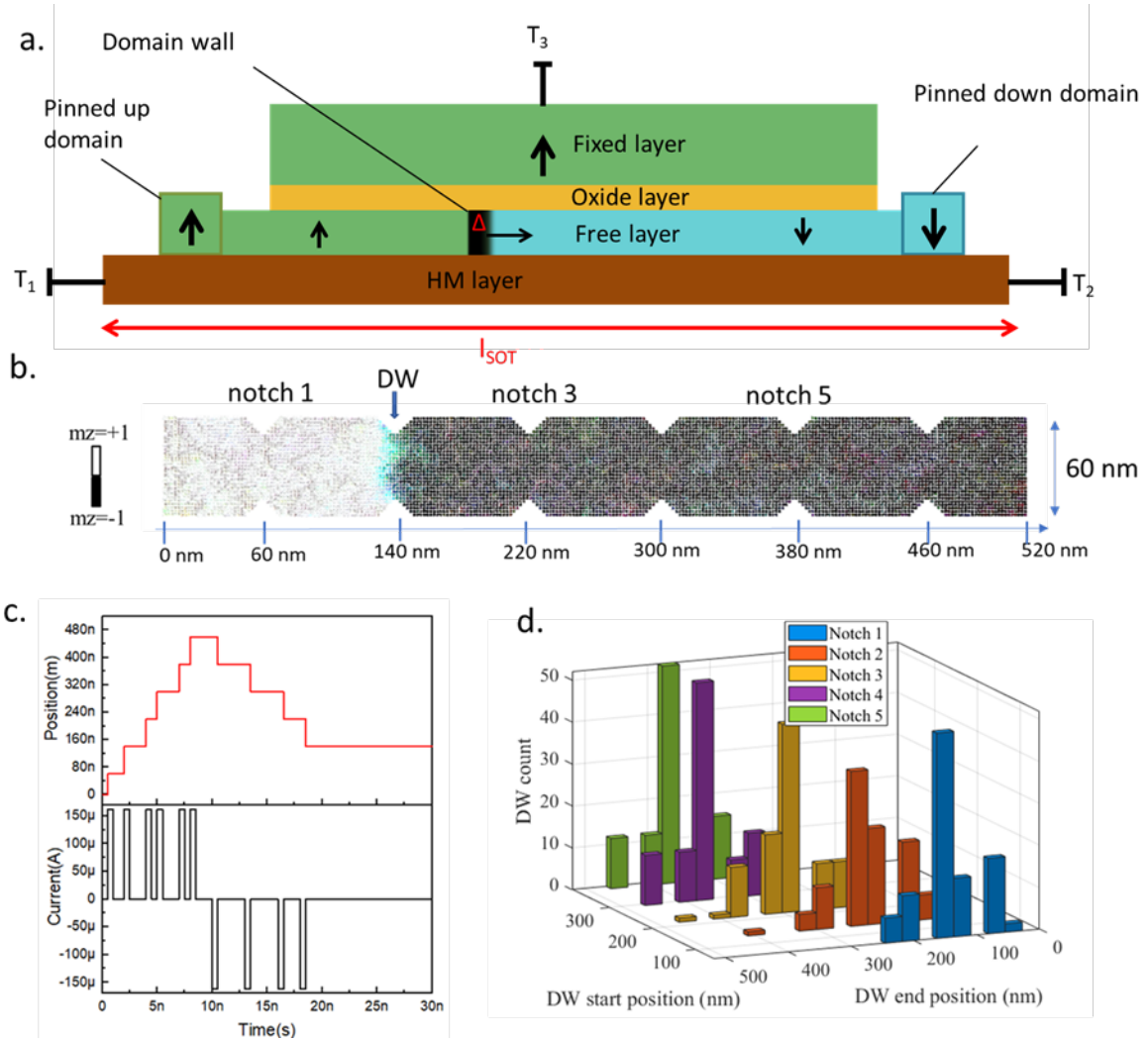

Fig. 1 SOT driven magnetic domain wall synapse (a) structure of notched domain wall device (b) micromagnetic simulations of notched constrictions in free layer where domain wall moves by means of SOT current (c) Circuit implementation of the notched DW synapse and its position corresponding to current pulse (d) Probability of DWs being pinned at locations after applying a fixed magnitude and duration current pulse when started from a particular notch position. The starting positions of the DWs are shown in the legends and the target pinning positions of the DWs are the subsequent notches.

magnetic DW device based quantized synapses. In Section III, a quantized CNN is presented using the magnetic DW device based synaptic arrays. The performance analysis of CNN is explained in section IV. Further, the effect of various variations on the accuracy of the network is evaluated in section V. Finally, section VI draws the conclusion.

II. Magnetic Domain Wall as Synapse

We propose a synapse with a magnetic racetrack that hosts a DW, as shown in Fig. 1a. A fixed amplitude and duration current pulse applied through the heavy metal layer that exerts spin orbit torque (SOT) in the adjacent magnetic racetrack and sets the DW into motion. The DWs are translated to different distances by applying different numbers of current pulses. The magnetization information of the racetrack is read by MTJs, where the MgO tunneling layer is sandwiched between the bottom racetrack free layer and the top reference layer. Magnetic regions pinned up and down are positioned at both ends of the free layer to prevent DW from being annihilated from the racetrack. The free layer including the heavy metal layer for injecting current is considered to be Pt/Co/Ni. This is motivated from the higher DW velocities exhibited in racetrack including Pt/Co due to higher interfacial Dzyaloshinskii-Moriya interaction (DMI) and SOT coupling [56-59]. The racetrack dimension is considered to be 520 nm × 60 nm × 1nm. Engineered notches are placed at regular intervals in the racetrack to arrest the DWs in stable locations. The notch pitch is selected to be 80 nm with left-most notch placed at 60 nm. Prototype MTJ with 250 nm wide and 500 nm long racetrack is reported [60]. We chose a 60 nm wide racetrack as it offers low area and decreases the likelihood of MTJ breakdown with pinholes and defects. Feature size less than ~ 50 nm is feasible and demonstrated for MTJs with perpendicular magnetic anisotropy [61]. Racetrack with trapezoidal shape geometry [60] and meander segments [62] are explored. However, linear conductance update is not possible in those racetracks which could incur additional operation during neural network learning to accommodate non-linear weight update. In contrast, rectangular shape racetrack offers linear conductance update when the artificial pinning sites are regularly placed. Artificial pinning sites with nonmagnetic metal diffusion [63], non-magnetic ion implantation [64], interfacial DMI modulation [65] and engineered notches [66] are investigated. Pinning site with engineered notches are more practical to implement as it can be done in one-step lithography. We use triangular shape notches with side length of ~ 23 nm × 8 nm × 8 nm. The physical DW width in our racetrack is calculated to be $\sim\pi\Delta = \pi\sqrt{\frac{A_{ex}}{K_u - \frac{\mu_0 M_S^2}{2}}} \approx 32$ nm (see simulation parameter in Table I). We want to capture a larger portion of the DWs within the notch. Larger notch side lengths are possible; however, they can increase the depinning current. Notch pitch could be set to $\pi\Delta\sim$ 32 nm to get distinct pinning locations along the racetrack. However, due to the presence of DMI (from heavy metal/ferromagnet interface such as in Pt/Co interface in the Pt/Co/Ni racetrack) in the system, the DW undergoes significant titling [67] and could therefore escape from the pinning sites. To counter the effect of DW titling we set the notch pitch ~ 80 nm which is higher than the physical DW width. Notch pitch could be further increased; however, it can increase the footprint of the device.

Table I
Device Parameters Of The Proposed Model

Parameter	Description	Value
α	Damping parameter	0.015 [69]
M_s	Saturation magnetization	$1\times10^6 A/m$ [70]
K_u	Anisotropy constant	$8.2\times10^5 J/m^3$ [70]
A	Exchange constant	$2\times10^{-11} J/m$ [71, 72]
D	DMI parameter	0.6×10^{-3} J/m^2 [73]

We simulated the evolution of DW position in the racetrack in the presence of room temperature (T=300K) thermal noise using the micromagnetic simulation tool MuMAX3 [68] and the set of parameters listed in Table I derived from experimental studies [59, 69-73]. The cell size for the simulation is assumed to be 2nm × 2nm × 1nm which is well within the ferromagnetic exchange length. The micromagnetic configuration of the racetrack-free layer is shown in Fig. 1b when the DW is pinned at second notch position. The simulation details are presented in supplementary section S4. Initially, the DWs are assumed to be pinned at different notch locations. A current pulse of amplitude 90×10^{10} A/m^2 (effective spin current is 90×10^9 A/m^2 assuming spin hall angle of Pt to be 0.1) and duration 0.5 ns is applied to the heavy metal which is sufficient to depin the DW and drive it to the next notch locations. We note that the DW velocities in our racetrack for different current densities are computed and compared with fabricated devices [59]. A qualitive match is observed in the trend of DW velocity (see supplementary section S5). Without stochastic behavior, the DW movement with the current pulse would be as illustrated in Fig. 1c. However, thermal noise and DW titling due to DMI [67] results in stochastic DW motion and consequently the DWs are pinned at different notch locations instead of being pinned at the intended notch when they start from a particular notch position as can be seen in Fig. 1d. We note, the DMI moves the DW in the presence of SOT current as it helps to stabilize the Neel domain wall [74]. Without the DMI, the DW becomes Bloch wall, where the centre (mid-plane) magnetization of the DW is parallel to the polarization of the current (direction of the spins). Spins aligned parallel to the DW magnetization do not exert any torque, thus barely move the DW with SOT current. In contrast, in Neel wall, the DW magnetization aligns perpendicular to the spins, thus maximum torque is exerted due to SOT, which helps to move the DW. Once the equilibrium end positions of the DWs are known, the conductances of the MTJ can be derived from average magnetization of the racetrack using the following equation:

$$G^{synapse} = \frac{G_{max} + G_{min}}{2} + \frac{G_{max} - G_{min}}{2} < m_z > \quad (1)$$

where, $< m_z >$ is the average magnetization moment of ferromagnetic racetrack along z-direction and reference ferromagnetic layer magnetization is assumed to point downward. G_{max} and G_{min} are the maximum and minimum conductance of the synapse, respectively (see supplementary section S7 for details). We note that, apart from thermal noise,

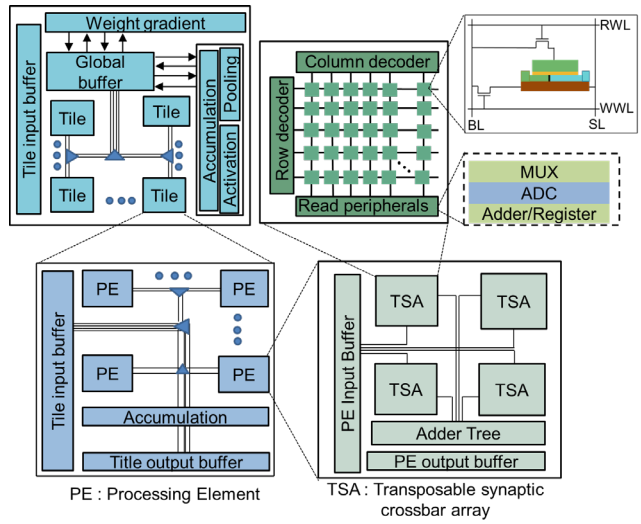

Fig. 2 Architecture level representation of on-chip learning hardware

defects can exist in the racetrack which can also influence the distribution of the equilibrium DW positions. To analyze the role of defects, we use Voronoi tessellation and change the anisotropy constant in different regions of the racetrack. The results are presented in supplementary section S6. The distribution of equilibrium DW positions in racetrack with defects is similar to racetrack without defects as long as the anisotropy variation is low which is expected in a small-scale device.

III. Deep Neural Network using DW synapse

The IMC architecture speeds up CNN processing by performing MVM within the memory crossbar array itself. The primary principle of analog IMC is to store the weights in the form of conductance states of a memory cell to perform the functionality of a synapse. In this manuscript, a notched magnetic DW device is used as a synapse to store 5 conductance states, as shown in Fig. 1(b), and enable the network to learn and classify at the circuit level. The hardware architecture to implement on-chip learning of the network is shown in Fig. 2. It is composed of crossbar arrays aided with peripheral read and write circuits, ADCs, MUX, and adders to form a transposable synaptic array (TSA). Multiple TSAs are arranged in H-routing manner with integrated buffers to form processing elements (PEs) that are further arranged in the form of tiles. The top-level architecture has multiple tiles with separate units for weight gradient, global buffer, accumulation, activation and pooling computations. The weight update is performed in row-by-row manner sequentially, whereas the inference is performed in parallel mode by activating all the columns simultaneously. Write and read word lines (WWL/RWL) control the access transistors to select a particular synaptic device for its write and read operations. The column multiplexer employs column sharing because of the energy and area overheads of ADC. Here, one ADC is shared by eight columns. Along each column, the output vectors are produced as the analog partial current sum that is converted to digital values by the ADC. To get the final sum values from the multistate weights and input multiplication, shift-and-add digital modules are used.

VGG-8 architecture is used to classify 32×32-coloured images from CIFAR-10 dataset as shown in Fig. 3. It consists of 6 convolution layers appended by 2 fully connected layers to perform CIFAR-10 image classification. The architecture uses maximum pooling layers with kernel of 2×2 after each convolution layer. Input voltages proportional to 1024 input features converted from a 32 × 32 image, are applied to the crossbar. Read currents, corresponding to the product of each input element and synaptic weight of the bit cell, add up using Kirchhoff's current law and feed to the activation function

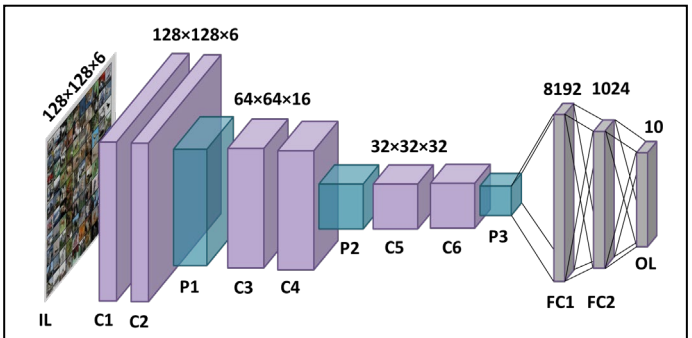

Fig. 3 Schematic of the VGG-8 model used for image classification from CIFAR-10 dataset.

Table II
Mapping algorithm

1. Unrolling: Input feature map (IFM) [W x W x D], output feature map (OFM) [W x W x N] with 3D kernels unrolled column wise [1: N]. Kernels of [K x K x D x N] arranged in [K x K] sub-matrices each with D rows and N columns. The sub-matrices of K x K are computed in parallel for generating OFMs.
2. In first cycle, first element in every OFM is generated with respect to 1 x 1 x N by sum of dot products. After W x W cycles, complete OFM of layer <n> is generated when IFM slides for all kernels.
3. Loss function (L) is defined as $L = \|(T - Y)\|^2$ where, T and Y represent desired and achieved output, respectively.
4. Perform feed-forward operation, the input [I] is fed into the network arranged in crossbars (weight matrix [X]).
   Output vector (I_h) from crossbar after MAC operation = X.I + bias is passed through activation and read peripherals to give input for subsequent layer.
5. Perform backpropagation:
   Weight updating by SGD:
   $W_{new} = W_{old} - \eta . \frac{\partial L}{\partial W}$ = clip (update with stochastic rounding (W_{old}, η , g), -1, 1).
   a. Clip function [75]: The floating-point number *n* is quantized into *B* bit signed integer representation bounded in range of [-1+σ(B), 1-σ(B)] by:
      $Q(n, B) = \text{clip}\left\{\sigma(B).\text{round}\left[\frac{n}{\sigma(B)}\right], -1 + \sigma(B), 1 - \sigma(B)\right\}$
      where $\sigma(B) = 2^{1-B}$, $B \in \mathbb{N}_+$
   b. Stochastic rounding function is described as:
      $$\begin{cases} Prob(n = floor(n)) = ceil(|n|) - |n| \\ Prob(n = ceil(n)) = |n| - floor(|n|) \end{cases}$$
6. Quantization is performed on weight (W) and activation (a) corresponding to synaptic weight states both in feed forward and backpropagation as:
   $$clip(x, a, b) = \min(max(x, a), b)$$
   $$\Delta = \frac{b-a}{n-1}$$
   $$q = \left[round\left(\frac{clip(x,a,b)-a}{\Delta}\right)\right] \times \Delta + a$$

Table III
Latency consumption for CNN

Layer	ADC latency(s)	Accumulation latency(s)	Synaptic array latency(s)	Weight gradient latency(s)	Weight update latency (s)
1	0.129774	0.0329325	0.201507	1.18366	9.54E-06
2	1.070640	1.76489	0.967181	9.32001	1.85E-05
3	0.233162	0.384354	0.211583	6.21413	2.65E-05
4	0.466323	0.480683	0.421276	6.21281	4.15E-05
5	0.085651	0.088286	0.077722	5.32607	13.28E-05
6	0.171302	0.130426	0.1548	5.32651	10.53E-05
7	0.004758	2.04022	0.004322	0.00810	1.86E-05
8	0.000594	0.000623	0.000542	0.00084	1.40E-07
Total	2.16221	4.92242	2.03893	33.5922	3.53E-04
Total/image	2.89E-07	6.56E-07	2.72E-07	4.47E-06	4.7E-11

Table IV
Energy consumption for CNN

Layer	ADC energy (J)	Accumulation energy (J)	Synaptic Array energy(J)	Weight gradient energy (J)	Weight update energy(J)
1	0.0174728	0.00817203	0.00456215	0.0487068	6.38E-08
2	0.289625	0.180987	0.0701857	1.76747	5.51E-08
3	0.115211	0.0788298	0.030202	0.737181	1.02E-07
4	0.214977	0.153347	0.0598064	1.43143	1.87E-07
5	0.0789722	0.05518	0.021961	0.529648	4.31E-07
6	0.1543	0.109247	0.0437938	1.0542	7.13E-07
7	0.0145854	0.0595865	0.00431061	0.15844	1.41E-06
8	5.37E-05	6.10E-05	2.27E-05	0.000193544	7.66E-09
Total	0.885197	0.64541	0.234844	5.72727	2.97E-06
Total/ image	1.18E-07	8.60E-08	3.13E-08	7.6E-07	3.96E-13

Table V
Area consumption for CNN

Device	Total Area(m^2)	Total IMC Area (m^2)	Routing Area(m^2)	ADC Area(m^2)	Accumulation Area(m^2)	Other Logic & Storage Area(m^2)	Weight Gradient Area(m^2)
DW	17.28 E-05	7.44E-06	2.86E-05	3.88E-05	3.69E-05	3.39E-05	2.72E-05
SRAM	239.77E-05	16.82E-05	22.2E-05	42.1E-05	130.4E-05	26.03E-05	2.04E-05

circuit at each output node. Hence, the MVM is performed within the crossbar array. Furthermore, the stochastic gradient descent (SGD) approach is employed to calculate the weight update at each output node using a weight update circuit. The weight change calculated at each output node is then multiplied by the inputs using the multiplier circuit. A current corresponding to the multiplier's output acts as a writing current on the DW synapse and modifies its conductance. Furthermore, a mapping algorithm, as shown in Table II, has been developed to map hardware architecture. The proposed DW device provides synaptic weight states, therefore, we adopted weight and activation quantization in our training algorithm. To accomplish this, the following set of functions as reported in [49] are used:

$$q = \left[round\left(\frac{clip(x,a,b)-a}{\Delta}\right)\right] \times \Delta + a \qquad (2)$$

Where $clip(x,a,b) = \min(max(x,a),b)$, $\boldsymbol{q}$ is the quantized value of the real valued number $\boldsymbol{x}$, $[\boldsymbol{a}; \boldsymbol{b}]$ is the quantization range and $\boldsymbol{n}$ is the level of quantization. $\Delta = \frac{b-a}{n-1}$, is the scaling factor that essentially divides a given range of real values into a number of partitions. The weights are initialized with random distribution of quantized states. The weights are updated using the stochastic gradient descent (SGD) algorithm which computes the weight gradient and losses. The error is computed by gradient of loss function (root mean square) with respect to activation and then this error is back propagated from $(n+1)^{th}$ layer to n^{th} layer. During backpropagation, the error is encoded into input voltage pulses and then MAC operation is performed with prior stored kernels. The error in the corresponding layer is propagated to each of the sub-array matrices according to the spatial location of kernels and each crossbar considers the encoded error as its input. The details of the algorithms are presented in section S3 of supplementary material.

IV. Performance of CNN

Micromagnetic simulations of notched DW device have been carried out using Mumax3. Furthermore, a Verilog-A based SPICE model of the notched DW device has been developed for performing circuit level simulations. Then, synaptic crossbar array of size 128×128 has been implemented on circuit

simulator for each layer separately. The output of each column of crossbar array is connected to an output neuron. After every iteration, the outputs generated by the circuit simulator are then passed to an in-house script that performs digital operations (ReLU activation function,) in the forward computation as well as the weight-update calculation in back-propagation.

The complete set of performance metrics is achieved for 125 epochs. Table III and IV show the layer wise latency and energy consumption, respectively, for various CNN modules/operations such as ADC, accumulation, synaptic array, weight gradient calculation, and weight update. Four operations: feedforward, error computation, gradient computation, and weight update, mainly contribute to the total energy and latency. The weight gradient computation dominates in the total latency and energy because of the repeated write and read operations of activation functions and errors to compute weight gradients. It is observed that the energy and latency utilized in weight updates is very less compared to other units. The reason for this is that weight update operations are performed only once per batch. Table V presents the area breakdown incurred by crossbar array, routing, ADCs, accumulation module, weight gradient storage array, and other peripheral circuitry in case of both DW and CMOS based CNNs. From the table, we can see that ADC and weight gradient computation units occupy large area as these are built up by conventional technology arrays that are separate from CIM arrays. Other logic and storage circuits also consume large areas as these provide data transfer during each operation. The accumulation area comprises PE- and tile-level adder trees for weight gradient unit and transposable synaptic-arrays. In order to access number of quantized DW states on the CNN performance, the proposed DW based CNN has been compared with CMOS based CNN utilizing same simulation approach. Fig. 4 shows the relation between the number of training epochs and accuracy of 5-state DW weight precision-based CNN and 32-bit CMOS weight precision-based network. It is observed that 5-state DW based NN achieves comparable accuracy as that of 32-bit CMOS based design. However, the DW based CNN is 13.8×, 9.6×, and 3.5× more efficient in terms of area, energy, and latency, respectively, in contrast to 32-bit CMOS based CNN, as shown in Table VI. For having a fair comparison, the proposed 5-state DW based CNN is also compared with 3-bit CMOS based design. The results show that DW based CNN is 2.5×, 1.8×, and

Table VI
Performance comparison of proposed DW based CNN with SRAM based design

Synaptic Device	Total Area ($\times 10^{-5}$ m^2)	Total latency (s)	Total energy (J)
DW	17.28	43.716	7.492
SRAM	239.8	154.79	71.80

Table VII
Impact on accuracy due to ADC precision

ADC precision (bits)	Accuracy (%)
10	90.9
8	90.4
6	86.0
4	10.0

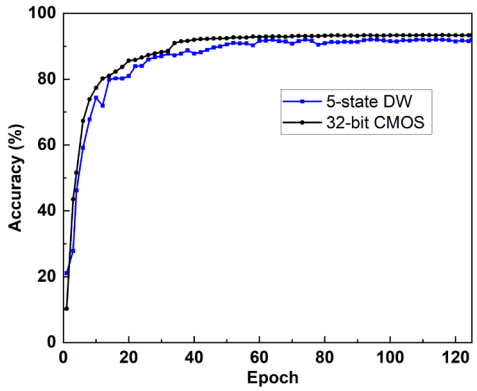

Fig. 4 Accuracy achieved in 125 epochs for 5-state and 32-bit DW device precision

2.9× more efficient in terms of energy, latency, and area, respectively without affecting the accuracy. Also, to evaluate the effect of ADC precision on accuracy, the training and inference of the proposed CNN has been performed for varying number of ADC precision bits as shown in Table VII. The results show that 8-bit ADC precision is sufficient to achieve 90.9% accuracy in the proposed DW based CNN.

V. Process Device-to-Device Variations on Accuracy

In order to implement CNN in hardware, the input vector [V_1……..V_m] is fed into the network arranged in crossbars architecture of the domain wall based synaptic devices as shown in Fig. 5. Weights are represented by the conductance of the notched DW device ($G^{synapse}$). The notched DW synapse offers linear and exclusively positive weight changes. This is because the weight change depends upon the conductance associated with the position of the DW, and the corresponding conductance values can only be positive. As a result, the synaptic weight modification is limited to positive values ranging from G_{min} to G_{max}. However, we require weights to be updated in both directions (positive and negative). Therefore, to implement both the positive and negative linear weight update, an extra conductance ($G_{parallel}$) is added in parallel to each of the synapses and a negative of the input voltage is applied on the $G_{parallel}$ [22]. The synapse conductance can be derived using Kirchhoff's current law for one column as shown in the following:

$$I_{eq} = -V_1 G_{parallel} + V_1 W_{1,1} \dots\dots\dots - V_m G_{parallel} + V_m W_{m,1} \quad (3)$$

where, $G_{parallel} = (G_{max} + G_{min})/2$ is the average of maximum and minimum conductance achieved by the DW device. To solve it for single synapse:

$$I_{eq,1} = -V_1 G_{parallel} + V_1 W_{1,1} \quad (4)$$

$$\frac{I_{eq,1}}{V_1} = G_1^{eq,synapse} = -G_{parallel} + W_{1,1} \quad (5)$$

$W_{1,1}$ is $G^{synapse}$ corresponding to the position of the DW (as shown in Eq. 1). Hence,

$$G_1^{eq,synapse} = -\frac{G_{max}+G_{min}}{2} + G^{synapse} \quad (6)$$

Equivalent resistance can be expresses as:

$$R^{eq,synapse} = \frac{2R_{max}R_{min}R^{synapse}}{2R_{max}R_{min} - R^{synapse}R_{max} - R^{synapse}R_{min}} \quad (7)$$

Where, $R^{synapse}$ is the DW device resistance corresponding to the position of the DW i.e., $R^{synapse} = R_P R_{AP}/R_P + R_{AP}$. R_{max} and R_{min} are corresponding to G_{max} and G_{min}, respectively. Hence, linear weights in both directions (positive and negative weights) are achieved as shown in Table VIII.

There are two main reasons that result in stochastic behavior in DW conductance state, therefore affecting the NN accuracy; 1. Thermal noise and DW tilting as discussed in section II. 2. Variation in notched DW device parameters such as free layer, fixed layer, heavy metal layer, and oxide layer dimensions and transistor width and length. To assess the effect of conductance variation of synaptic devices, 1000 Monte Carlo simulations have been performed with the variation in TMR, oxide thickness, MTJ area, width and length of the transistor as shown in Table IX. The statistical distributions of 5 linear conductance change states of the proposed synapse are shown in Fig. 6. Furthermore, deviation from the 5 conductance distributions is obtained and its effect on the DNN accuracy is evaluated. It is observed that the deviation in the conductance below 20% is enough to achieve the accuracy of 88.6% as shown in Table X.

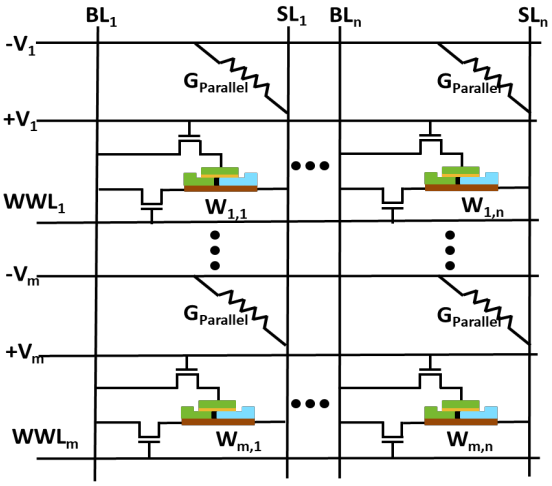

Fig. 5 Circuit to achieve linear weight updates.

Table VIII
Synapse conductance corresponding to DW position

DW Position (nm)	MTJ Resistance (kΩ)	Conductance corresponding to MTJ resistance ($\times 10^{-5}$ S)	Conductance update in both directions ($\times 10^{-5}$ S)
60	29.8	3.3	-8.99
140	12.7	7.8	-4.51
220	8.1	12.3	-0.02
300	5.9	16.9	4.60
380	4.7	21.2	9.11

Table IX
Percentage variations in process parameters

MTJ parameters	Variation (%)	References
Free layer thickness	5	[76], [77], [78]
Oxide barrier thickness	2	[76], [77], [79]
TMR	2	[80]
Width of the transistor	5	[76], [81]
Length of the transistor	5	[76], [81]

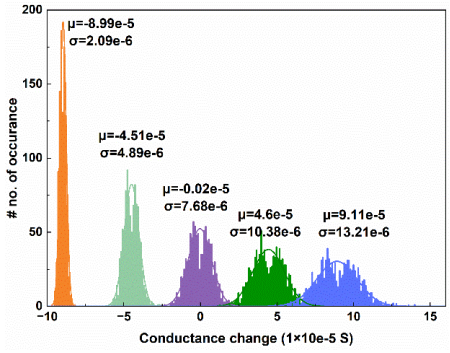

Fig. 6 Statistical distribution of synapse conductance considering process variations.

Table X
Change in accuracy due to device conductance variation.

Device conductance variation (%)	Accuracy (%)
0	90.4
10	89.4
20	88.6
30	86.6
40	85.7
50	80.4

VI. Conclusion

A quantized neural network using a magnetic domain wall (DW) synaptic array and CMOS peripheral circuits has been implemented. The magnetic domain wall model is rigorously studied by considering stochasticity and process variation to implement the synapse. By using extracted synaptic device characteristics, a VGG8 neural network architecture has been implemented for CIFAR-10 data classification. The proposed neural network architecture achieves efficient on-chip learning with 90.4% inference accuracy. The algorithmic level quantization and optimization shows the efficiency of the presented work for next-generation hardware implementations. In comparison to pure CMOS 32-bit based design it shows an overall improvement in area, energy, and latency by 13.8×, 9.6×, and 3.5×, respectively.

Acknowledgement

S. Dhull, A. Nisar, and B. K. Kaushik acknowledge funding from Science and Engineering Research Board, Department of Science and Technology, Government of India under Grant CRG/2019/004551.

W. A. Misba and J. Atulasimha acknowledge funding from the US National Science Foundation grant: NSF ECCS 1954589, Virginia State's Commonwealth Cyber Initiative Grant Award #VV-1Q24-009 and Virginia Commonwealth University's Breakthrough grant on Energy Efficient Neurocomputer (E2N).

References

[1] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[2] V. Sze, Y. -H. Chen, T. -J. Yan,g and J. S. Emer, "Efficient processing of deep neural networks: a tutorial and survey," in *proceed. of IEEE*, vol. 105, no. 12, pp. 2295-2329, Dec. 2017, doi: 10.1109/JPROC.2017.2761740.

[3] A. Pedram, S. Richardson, M. Horowitz, S. Galal, and S. Kvatinsky, ‘‘Dark memory and accelerator-rich system optimization in the dark silicon era,’’ *IEEE Design Test*, vol. 34, no. 2, pp. 39–50, 2017.

[4] H.-S.-P. Wong and S. Salahuddin, ‘‘Memory leads the way to better computing,’’ *Nat. Nanotechnol*., vol. 10, no. 3, pp. 191–194, 2015.

[5] D. Ielmini and H. S. P. Wong, ‘‘In-memory computing with resistive switching devices,’’ *Nat. Electron*., vol. 1, no. 6, pp. 333–343, 2018.

[6] A. Sebastian, M. Le Gallo, R. Khaddam-Aljameh, and E. Eleftheriou, ‘‘Memory devices and applications for in-memory computing,’’ *Nat. Nanotechnol.*, vol. 15, pp. 529–544, 2020.

[7] G. W. Burr, R. M. Shelby, A. Sebastian, S. Kim, S. Kim, S. Sidler, K. Virwani, M. Ishii, P. Narayanan, A. Fumarola, L. L. Sanches, I. Boybat, M. Le Gallo, K. Moon, J. Woo, H. Hwang, and Y. Leblebici, ‘‘Neuromorphic computing using non-volatile memory,’’ *Adv. Phys.*, vol. 2, no. 1, pp. 89–124, Dec. 2016.

[8] C. Li, M. Hu, Y. Li, H. Jiang, N. Ge, E. Montgomery, J. Zhang, W. Song, N. Dávila, C. E. Graves, Z. Li, J. P. Strachan, P. Lin, Z. Wang, M. Barnell, Q. Wu, R. S. Williams, J. J. Yang, and Q. Xia, ‘‘Analogue signal and image processing with large memristor crossbars,’’*Nat. Electron.*, vol. 1, pp. 52–59, 2017.

[9] M. Hu, J. P. Strachan, Z. Li, E. M. Grafals, N. Davila, C. Graves, S. Lam, N. Ge, J. J. Yang, and R. S. Williams, ‘‘Dot-product engine for neuromorphic computing: Programming 1T1M crossbar to accelerate matrix-vector multiplication,’’ *in Proc. 53rd Annu. Design Autom. Conf.,* Austin, TX, USA, Jun. 2016, pp. 1–6.

[10] T. H. Lee, D. Loke, K.-J. Huang, W.-J. Wang, and S. R. Elliott, ‘‘Tailoring transient-amorphous states: Towards fast and power-efficient phasechange memory and neuromorphic computing,’’ *Adv. Mater*., vol. 26, no. 44, pp. 7493–7498, Nov. 2014, doi: 10.1002/adma.201402696.

[11] S. Ambrogio, P. Narayanan, H. Tsai, R. M. Shelby, I. Boybat, C. Di Nolfo, S. Sidler, M. Giordano, M. Bodini, N. C. P. Farinha, B. Killeen, C. Cheng, Y. Jaoudi, and G. W. Burr, ‘‘Equivalent accuracy accelerated neural-network training using analogue memory,’’ *Nat.*, vol. 558, no. 7708, pp. 60–67, Jun. 2018, doi: 10.1038/s41586-018-0180-5.

[12] M. Suri, O. Bichler, D. Querlioz, O. Cueto, L. Perniola, V. Sousa, D. Vuillaume, C. Gamrat, and B. DeSalvo, ‘‘Phase change memory as synapse for ultra-dense neuromorphic systems: Application to complex visual pattern extraction,’’ *in Proc. Int. Electron Devices Meeting*, Dec. 2011, p. 4, doi: 10.1109/IEDM.2011.6131488

[13] S. Yu, Y. Wu, R. Jeyasingh, D. Kuzum, and H.-S. P. Wong, ‘‘An electronic synapse device based on metal oxide resistive switching memory for neuromorphic computation,’’ *IEEE Trans. Electron Devices*, vol. 58, no. 8, pp. 2729–2737, Aug. 2011, doi: 10.1109/TED.2011.2147791.

[14] J. Woo, K. Moon, J. Song, S. Lee, M. Kwak, J. Park, and H. Hwang, ‘‘Improved synaptic behavior under identical pulses using AlOx /HfO2 bilayer RRAM array for neuromorphic systems,’’ *IEEE Electron Device Lett.*, vol. 37, no. 8, pp. 994–997, Aug. 2016, doi: 10.1109/LED.2016.2582859.

[15] P. Yao, H. Wu, B. Gao, S. B. Eryilmaz, X. Huang, W. Zhang, Q. Zhang, N. Deng, L. Shi, H.-S. P. Wong, and H. Qian, ‘‘Face classification using electronic synapses,’’ *Nat.Commun*., vol. 8, no. 1, pp. 1–8, May 2017.

[16] A.Chanthbouala, et al., “A ferroelectric memristor,” *Nat. Materials,* vol. 11, pp. 860–864, 2012.

[17] S.Lequeux et al., “A magnetic synapse: multilevel spin-torque memristor with perpendicular anisotropy,”*Sci. Reports*, vol. 6, no. 31510, 2016.

[18] A. F. Vincent et al., ‘‘Spin-transfer torque magnetic memory as a stochastic memristive synapse for neuromorphic systems,’’ *IEEE Trans. Biomed. Circuits Syst*., vol. 9, no. 2, pp. 166–174, 2015.

[19] D. Kaushik, U. Singh, U. Sahu, I. Sreedevi, and D. Bhowmik, ‘‘Comparing domain wall synapse with other non-volatile memory devices for on chip learning in analog hardware neural network,’’ *AIP Adv*., vol. 10, no. 2, pp. 1–7, Feb. 2020, Art. no. 025111.

[20] A. Sengupta, Y. Shim, and K. Roy, ‘‘Proposal for an all-spin artificial neural network: Emulating neural and synaptic functionalities through domain wall motion in ferromagnets,’’ *IEEE Trans. Biomed. Circuits Syst.*, vol. 10, no. 6, pp. 1152–1160, 2016.

[21] M.-C. Chen, A. Sengupta, and K. Roy, ‘‘Magnetic skyrmion as a spintronic deep learning spiking neuron processor,’’ *IEEE Trans. Magn.,* vol. 54, no. 8, Aug. 2018, Art. no. 1500207.

[22] D. Bhowmik, U. Saxena, A. Dankar, A. Verma, D. Kaushik, S. Chatterjee, and U. Singh, ‘‘On-chip learning for domain wall synapse based fully connected neural network,’’ *J. Magn. Magn. Mater*., vol. 498, Nov. 2019, Art. no. 1654342.

[23] M. Alamdar, T. Leonard, C. Cui, B. P. Rimal, L. Xue, O. G. Akinola, T. P. Xiao, J. S. Friedman, C. H. Bennett, M. J. Marinella, and J. A. C. Incorvia, ‘‘Domain wall-magnetic tunnel junction spin-orbit torque devices and circuits for in-memory computing,’’ *Appl. Phys. Lett.,* vol. 118, Mar. 2021, Art. no. 112401.

[24] D. Zhang, Y. Hou, L. Zeng, and W. Zhao, ‘‘Hardware acceleration implementation of sparse coding algorithm with spintronic devices,’’ *IEEE Trans. Nanotechnol*., vol. 18, pp. 518–531, 2019.

[25] V. Sampath, N. D’Souza, D. Bhattacharya, G. M. Atkinson, S. Bandyopadhyay, and J. Atulasimha, Acoustic-Wave-Induced Magnetization Switching of Magnetostrictive Nanomagnets from Single-Domain to Nonvolatile Vortex States,Nano Lett. 2016, 16, 9, 5681–5687

[26] N. D’Souza, M. S. Fashami, S. Bandyopadhyay, and J. Atulasimha, Experimental Clocking of Nanomagnets with Strain for Ultralow Power Boolean Logic,Nano Lett. 2016, 16, 2, 1069–1075

[27] A. K. Biswas, S. Bandyopadhyay, J. Atulasimha, Complete magnetization reversal in a magnetostrictive nanomagnet with voltage-generated stress: A reliable energy-efficient non-volatile magneto-elastic memory, Appl. Phys. Lett. 105, 072408 (2014)

[28] K. Roy, S. Bandyopadhyay, J. Atulasimha, Hybrid spintronics and straintronics: A magnetic technology for ultra low energy computing and signal processing,Appl. Phys. Lett. 99, 063108 (2011)

[29] A. K. Biswas, S. Bandyopadhyay, and J. Atulasimha, Energy-efficient magnetoelastic non-volatile memory, Appl. Phys. Lett. 104, 232403 (2014)

[30] X. Li, D. Carka, C.- Liang, A. E. Sepulveda, \S. M. Keller, P. K. Amiri, G. P. Carman, and C. S. Lynch, Strain-mediated 180° perpendicular magnetization switching of a single domain multiferroic structure,Journal of Applied Physics 118, 014101 (2015)

[31] N. Lei, T. Devolder, G. Agnus, P. Aubert, L. Daniel, J.-V. Kim, W. Zhao, T. Trypiniotis, R. P. Cowburn, C. Chappert, D. Ravelosona and P. Lecoeur, Strain-controlled magnetic domain wall propagation in hybrid piezoelectric/ferromagnetic structures, Nature Communications volume 4, Article number: 1378 (2013)

[32] C. Grezes, F. Ebrahimi, J. G. Alzate, X. Cai, J. A. Katine, J. Langer, B. Ocker, P. Khalili Amiri, and K. L. Wang, Ultra-low switching energy and scaling in electric-field-controlled cle magnetic tunnel junctions with high resistance-area product,Appl. Phys. Lett. 108, 012403 (2016)

[33] Dhritiman Bhattacharya, Md Mamun Al-Rashid, Jayasimha Atulasimha,Voltage controlled core reversal of fixed magnetic skyrmions without a magnetic field, Scientific Reports volume 6, Article number: 31272 (2016)

[34] D. Bhattacharya, J. Atulasimha, Skyrmion-mediated voltage-controlled switching of ferromagnets for reliable and energy-efficient two-terminal memory,ACS Appl. Mater. Interfaces 2018, 10, 20, 17455–17462

[35] J. C. Slonczewski, Current-driven excitation of magnetic multilayers, J. Magn. Magn. Mater. 159, L1 (1996).

[36] K.-S. Ryu, L. Thomas, S.-H. Yang and S. Parkin, “Chiral spin torque arising from proximity-induced magnetization,” *Nature* Communications volume 5, Article number: 3910, 2014.

[37] W. A. Misba, M. M. Rajib, D. Bhattacharya, and J. Atulasimha, Acoustic-Wave-Induced Ferromagnetic-Resonance-Assisted Spin-Torque Switching of Perpendicular Magnetic Tunnel Junctions with Anisotropy Variation, Phys. Rev. Applied 14, 014088

[38] M. A. Azam, D. Bhattacharya, D. Querlioz, C. A. Ross, J. Atulasimha, Voltage control of domain walls in magnetic nanowires for energy-efficient neuromorphic devices, Nanotechnology 31 145201 (2020)

[39] W. A. Misba, T. Kaisar, D. Bhattacharya, J. Atulasimha, Voltage-controlled energy-efficient domain wall synapses with stochastic distribution of quantized weights in the presence of thermal noise and edge roughness,IEEE Transactions on Electron Devices,pp. 1658 - 1666, Volume: 69, Issue: 4, April 2022

[40] X. Jiang, L. Thomas, R. Moriya, M. Hayashi, B. Bergman, C. Rettner, and S. S. P. Parkin, ‘‘Enhanced stochasticity of domain wall motion in magnetic racetracks due to dynamic pinning,’’ *Nat.Commun*., vol. 1, no. 1, pp. 1–5, 2010.

[41] S. Dutta, S. A. Siddiqui, J. A. C.-Incorvia, C. A. Ross, and M. A. Baldo, “The Spatial Resolution Limit for an Individual Domain Wall in Magnetic Nanowires,” *Nano Lett.,*vol. 17, no. 9, pp. 5869–5874, 2017.

[42] G. W. Burr et al.,“Experimental demonstration and tolerancing of a large-scale neural network (165 000 synapses) using phase-change memory as the synaptic weight element,”*IEEE Trans. Electron Devices*, vol. 62, no. 11, pp. 3498–3507, 2015.

[43] I. Boybat et al., ‘‘Neuromorphic computing with multi-memristive synapses,’’ *Nat.Commun*., vol. 9, no. 1, pp. 1–12, Jun. 2018.

[44] S. Agarwal, R. B. Jacobs Gedrim, A. H. Hsia, D. R. Hughart, E. J. Fuller, A. A. Talin, C. D. James, S. J. Plimpton, and M. J. Marinella, ‘‘Achieving ideal accuracies in analog neuromorphic computing using periodic carry,’’ *in Proc. Symp. VLSI Technol*., Kyoto, Japan, Jun. 2017, pp. T174–T175.

[45] V. B. Desai, D. Kaushik, J. Sharda, and D. Bhowmik, “On-chip learning of a domain-wall-synapse-crossbar-array-based convolutional neural network,” *Neuromorph. Comput. Eng.,*vol. 2, no. 2 pp. 024006, 2022.

[46] M. L. Gallo, S R Nandakumar, L.Ciric, I.Boybat, R. K.-Aljameh, C.Mackin, and A. Sebastian, “Precision of bit slicing with in-memory computing based on analog phase-change memory crossbars,” *Neuromorph.Comput. and Eng.,* vol. 2, no. 1, 014009, 2022.

[47] M. L. Gallo, A. Sebastian, R. Mathis, M. Manica, H. Giefers, T. Tuma, C. Bekas, A. Curioni, and E. Eleftheriou, ‘‘Mixed-precision in-memory computing,’’ *Nat. Electron*., vol. 1, no. 4, pp. 246–253, Apr. 2018.

[48] S. R. Nandakumar et al., ‘‘Mixed-precision deep learning based on computational memory,’’ *Frontiers Neurosci.,* vol. 14, pp. 1–17, 2020.

[49] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, ‘‘Quantized neural networks: Training neural networks with low precision weights and activations,’’ *J. Mach. Learn. Res.,* vol. 18, no. 187, pp. 1–30, Apr. 2017.

[50] R. Krishnamoorthi, “Quantizing deep convolutional networks for efficient inference,” A whitepaper, arXiv:1806.08342, 2018.

[51] M. Courbariaux, Y. Bengio, and J.-P. David, ‘‘Binaryconnect: Training deep neural networks with binary weights during propagations,’’ *in Proc. 28th Int. Conf. Neural Inf. Process. Syst*., Montreal, BC, Canada, vol. 2, Dec. 2015, pp. 3123–3131.

[52] W. A.Misba, M. Lozano, D.Querlioz, and J.Atulasimha, “Energy efficient learning with low resolution stochastic domain wall synapse for deep neural networks,”*IEEE Access*, vol. 10, pp. 84946 - 84959, 2022.

[53] M. Sharad, C. Augustine, G. Panagopoulos, K. Roy, “Spin-based neuron model with domain-wall magnets as synapse,” *IEEE Trans. on Nanotech.*, vol. 11, pp. 843-53, 2012.

[54] X. Hu et al., "Process Variation Model and Analysis for Domain Wall-Magnetic Tunnel Junction Logic," *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, Seville, Spain, 2020, pp. 1-5, doi: 10.1109/ISCAS45731.2020.9180675.

[55] Z. Jiang. S. Yin, J.S. Seo, and M. Seok, “C3SRAM: An In-Memory-Computing SRAM Macro Based on Robust Capacitive Coupling Computing Mechanism,” *IEEE Jour. Of Solid-State Cir.*, vol. 55, no. 7, 2020.

[56] E. Raymenants et al., “Nanoscale domain wall devices with magnetic tunnel junction read and write,” *Nat. Elect.,* vol. 4, pp. 392–398, 2021.

[57] S. Emori, U. Bauer, S-M. Ahn, E. Martinez, and G. S. D. Beach, “Current driven dynamics of chiral ferromagnetic domain walls,” *Nat. Mater.,* vol. 12, pp. 611–616, 2013.

[58] M. Miron et al., “Fast current-induced domain-wall motion controlled by the Rashba effect,” *Nat. Mater.*, vol. 10, pp. 419–423, 2011.

[59] K-S. Ryu, L. Thomas, S-H. Yang, and S. Parkin, “Chiral spin torque at magnetic domain walls,” *Nature Nanotech.,* vol. 8, pp. 527-533, 2013.

[60] T. Leonard, S. Liu, M.Alamdar, H. Jin, C. Cui, O. G. Akinola, L. Xue, T. P. Xiao, J. S. Friedman, M. J. Marinella, C. H. Bennett, and J. A. C. Incorvia, “Shape-dependent multi-weight magnetic artificial synapses for neuromorphic computing,” *Adv. Electron. Mater., vol.* 8, p. 2200563, 2022.

[61] L. Xue *et al.*, “Process Optimization of Perpendicular Magnetic Tunnel Junction Arrays for Last-Level Cache beyond 7 nm Node,” *in proceedings IEEE Symposium on VLSI Technology*, 2018, Honolulu, HI, USA.

[62] D. Kumar, H. J. Chung, J. Chan, T. Jin, S. T. Lim, S.S. P. Parkin, R. Sbiaa, and S.N. Piramanayagam, “Ultra-low energy domain wall device for spin-based neuromorphic computing,” *ACS Nano.,* vol. 17, no. 7, pp. 6261–6274, 2023.

[63] T. L. Jin, M. Ranjbar, S. K. He, W. C. Law, T. J. Zhou, W. S. Lew, X. X. Liu, and S. N. Piramanayagam, “Tuning magnetic properties for domain wall pinning via localized metal diffusion,” *Sci. Rep.*, vol. 7, no. 1, pp. 1−9, 2017.

[64] T. Jin, D. Kumar, W. Gan, M. Ranjbar, F. Luo, R. Sbiaa, X. Liu, W. S. Lew, and S. N. Piramanayagam, “Nanoscale compositional modification in Co/Pd multilayers for controllable domain wall pinning in racetrack memory,”. *Phys. Stat. Solidi-Rap. Res. Lett.*, vol.12, no. 10, p. 1800197, 2018.

[65] D. Kumar, J. Chan, and S.N. Piramanayagam, “Domain wall pinning through nanoscale interfacial Dzyaloshinskii Moriya interaction,” *J. Appl. Phys*., vol. 130, no. 21, p. 213901, 2021.

[66] M. Kläui, C. A. F. Vaz, J. A. C. Bland, W. Wernsdorfer, G. Faini, and E. Cambril. “Domain wall pinning and controlled magnetic switching in narrow ferromagnetic ring structures with notches,” *J. of Appl. Phys.,* vol. 93, no. 10, 7885−7890, 2003.

[67] S. Liu, T. P. Xiao, C. Cui, J. A. C. Incorvia, C. H. Bennett, and M. J. Marinella, ‘‘A domain wall-magnetic tunnel junction artificial synapse with notched geometry for accurate and efficient training of deep neural networks,’’ *Appl. Phys. Lett*., vol. 118, May 2021, Art. no. 202405.

[68] A. Vansteenkiste, J. Leliaert, M. Dvornik, M. Helsen, F. Garcia-Sanchez, and B. V. Waeyenberge, ‘‘The design and verification of MuMax3,’’ *AIP Adv.,* vol. 4, no. 10, Oct. 2014, Art. no. 107133.

[69] J. M. Shaw, H. T. Nembach, and T. J. Silva, “Resolving the controversy of a possible relationship between perpendicular magnetic anisotropy and the magnetic damping parameter,” *Appl. Phys. Lett.,* vol. 105, no. 6, pp. 062406, 2014.

[70] M. Heigl, R. Wendler, S. D. Haugg, and M. Albrecht, “Magnetic properties of Co/Ni-based multilayers with Pd and Pt insertion layers,” *J. Appl. Phys.,* vol. 127, no. 23, p. 233902, 2020.

[71] C. Eyrich, W. Huttema, M. Arora, E. Montoya, F. Rashidi, C. Burrowes, B. Kardasz, E. Girt, B. Heinrich, O. N. Mryasov, M. From, and O. Karis, “Exchange stiffness in thin film Co alloys," *J. Appl. Phys*., vol. 111, no. 7, p. 07C919, 2012.

[72] S. P. Vernon, S. M. Lindsay, and M. B. Stearns, “Brillouin scattering from thermal magnons in a thin Co film,” *Phys. Rev. B*, vol. 29, pp. 4439, 1984.

[73] N. S. Gusev, A. V. Sadovnikov, S. A. Nikitov, M. V. Sapozhnikov, and O. G. Udalov, “Manipulation of the Dzyaloshinskii–Moriya Interaction in Co/Pt Multilayers with Strain,” *Phys.Rev. Lett.*, vol. 124, p. 157202, 2020.

[74] L. Liu, R. A. Buhrman, and D. C. Ralph, “Review and Analysis of Measurements of the Spin Hall Effect in Platinum”, *arXiv*:1111.3702, 2012.

[75] S. Wu, G. Li, F. Chen, and L. Shi L, “Training and inference with integers in deep neural networks,” arXiv preprint arXiv:1802.04680. 2018 Feb 13.

[76] Y. Zhang, X. Wang, H. Li, and Y. Chen, “STT-RAM cell optimization considering MTJ and CMOS variations,” *IEEE Trans. Magn.,* vol. 47, no. 10, pp. 2962–2965, Oct. 2011.

[77] W. Kang, L. Zhang, J. Klein, Y. Zhang, D. Ravelosona, and W. Zhao, “Reconfigurable codesign of STT-MRAM under process variations in deeply scaled technology,” *IEEE Trans. Electron Devices*, vol. 62, no. 6, pp. 1769-1777, 2015.

[78] X. Wang, W. Zhu, M. Siegert and D. Dimitrov, "Spin Torque Induced Magnetization Switching Variations," *IEEE Trans. on Magnetics*, vol. 45, no. 4, pp. 2038-2041, April 2009, doi: 10.1109/TMAG.2009.2015376.

[79] W. S. Zhao et al., “Failure and reliability analysis of STT-MRAM,” *Microelectron. Rel*., vol. 52, nos. 9–10, pp. 1848–1852, 2012.

[80] J. Li, C. Augustine, S. Salahuddin, and K. Roy, “Modeling of failure probability and statistical design of spin-orbit-torque transfer magnetic random-access memory (STT MRAM) array for yield enhancement,” *in Proc. Design and Automation Conf. (DAC)*, California, USA, 2008, pp. 278-283.

[81] W. Wen, Y. Zhang, Y. Chen, Y. Wang and Y. Xie, “PS3-RAM: A Fast Portable and Scalable Statistical STT-RAM Reliability/Energy Analysis Method,” *IEEE Trans. on Comp.-Aided Design of Int. Cir. and Sys.*, vol. 33, no. 11, pp. 1644-1656, 2014.