IEEE TRANSACTIONS ON
EMERGING TOPICS
IN COMPUTING

Received 4 December 2021; revised 7 July 2022; accepted 24 August 2022.
Date of publication 5 September 2022; date of current version 7 June 2023.

Digital Object Identifier 10.1109/TETC.2022.3202113

High-Performance and Robust Spintronic/CNTFET-Based Binarized Neural Network Hardware Accelerator

MILAD TANAVARDI NASAB, ABDOLAH AMIRANY, (Member, IEEE), MOHAMMAD HOSSEIN MOAIYERI, (Senior Member, IEEE), AND KIAN JAFARI, (Member, IEEE)

The authors are with the Faculty of Electrical Engineering, Shahid Beheshti University, Tehran 19839-63113, Iran
CORRESPONDING AUTHOR: KIAN JAFARI (k_jafari@sbu.ac.ir.)

ABSTRACT The convolutional neural network (CNN) is a significant part of the artificial intelligence (AI) systems widely used in different tasks. The binarized neural networks (BNNs) reduce power consumption and hardware overhead to answer the demands for using AI in power-limited applications. In this paper, a BNN hardware accelerator is proposed. The proposed approach is based on a novel nonvolatile XNOR/XOR circuit designed using the magnetic tunnel junction (MTJ) and gate-all-around carbon nanotube field-effect transistor (GAA-CNTFET) devices. The nonvolatility of the proposed design leads to the elimination of external memory access that significantly decreases the data transmission delay and power dissipation. Moreover, it consumes low energy, which is very critical in battery-operated devices. Furthermore, the combinational read circuitry of the proposed design leads to high robustness to process variations. According to the simulation results, our proposed design has a logical error rate of 0.0164%, which is negligible and offers a significantly high network accuracy even in the presence of significant process variations. Our proposed hardware accelerator provides at least 13%, 29%, and 41% improvements regarding power, power delay product (PDP), and area compared to its state-of-the-art counterparts.

INDEX TERMS Spintronic, logic-in-memory, hardware accelerator, binarized neural network, GAA-CNTFET

I. INTRODUCTION

Recently, hardware implementation of neuromorphic computational architectures has widely been investigated due to the high performance and low power consumption [1]. Increasing the demands for using human-brain-inspired networks in electronic devices with limited hardware and power sources like mobile devices and IoT edge sensors requires area- and energy-efficient designs [2].

Most recent circuits have been designed based on the complementary metal-oxide-semiconductor (CMOS) transistors and memristors [1]. The efficiency of CMOS-based designs has been limited in deep nanoscale nodes due to the short channel effects and high power density. In addition, critical parameters like power consumption, throughput, integration capability, area, and endurance, make the hybrid CMOS-memristor designs relatively inefficient.

Using emerging technologies like carbon nanotube field-effect transistors (CNTFET) [3], [4] and magnetic tunnel junction (MTJ) [4] together with novel architectures like logic-in-memory (LiM) [5], [6] offers powerful hardware and energy savings. Moreover, using hardware-efficient approaches like binarized neural networks (BNNs) can reduce the power and area overheads and increase the performance of neuromorphic circuits [5]–[7]. Due to the lower leakage currents of GAA-CNTFET compared to the CMOS transistors and considering that in the applications using the proposed design, the system is idle for most of its life, GAA-CNTFET has been chosen to save energy and extend the system's operational time. Furthermore, the steeper voltage transfer characteristic (VTC) curve of the GAA-CNTFET-based inverter and the ability to modify the VTC diagram by changing the flat-band voltage lead to more efficient and reliable thresholding.

Due to numerous weights in neural networks and the necessity of accessing them in every cycle, power consumption, delay, and complexity will become critical in devices with limited hardware and power supply. Using LiM

architecture is one of the most promising solutions for addressing this issue. Integrating logic and memory in a single chip significantly reduces the delay and power for data transmission. Moreover, the LiM architecture reduces the routing complexity [8]. Accordingly, using the BNN implementation with the LiM architecture significantly reduces the delay, power, and design complexity [5], [6].

Beyond-CMOS emerging technologies can facilitate the design and implementation of efficient neural networks and LiM architectures. Due to the significant suppression of short channel effects and lower energy consumption, replacing CMOS transistors with CNTFETs can increase energy efficiency, a critical necessity in mobile devices [9], [10]. More importantly, the threshold voltage of a CNTFET can be adjusted by changing flat-band voltage (Vfb), which can be used in low power designs.

Spintronic devices, like MTJs, are promising candidates for implementing neural networks and have received much attention in recent years. Non-volatility, intrinsic immunity to soft errors, high density, and low power consumption of MTJs make them efficient for low-power LiM architectures [8]. Moreover, the fabrication process and resistance of MTJs are compatible with the CMOS and CNTFET devices [4]. The MTJs do not impose much area overhead, as they are fabricated on a separated layer above the transistors. Moreover, MTJs are also superior in reliability and endurance compared to memristors [3], [4].

This paper proposes a high-performance nonvolatile XOR/XNOR circuit for hardware implementation of the binarized neural network accelerator. The proposed design offers low hardware overhead and superior energy efficiency. Furthermore, the proposed design does not need complex and sensitive circuits like sense amplifiers (SAs) and pre-charge sense amplifiers (PCSAs). Consequently, our proposed design occupies a smaller area than its state-of-the-art counterparts. Moreover, our design is resilient to significant process variations, making the BNN hardware implementation invulnerable to process variations. Consequently, higher accuracy and performance are reached. Our investigations reveal that by considering the error rate in network validation, even a tiny logical error rate can significantly decrease the accuracy of the hardware-implemented network. The proposed design's efficient delay, power, and area make it a promising candidate for binary network hardware accelerator. It is noteworthy that this design needs neither an analog-to-digital converter nor a digital-to-analog converter. Moreover, auxiliary circuits like adders and comparators needed to implement the whole network can be simply realized using CNTFETs. The contributions of this paper can be summarized as:

- Designing a nonvolatile XNOR/XOR gate for a binarized neural network hardware implementations
- Proposing an efficient architecture for hardware implementation of binarized neural networks
- Investigating the effect of the logical errors caused by the process variations on the accuracy of the binarized neural networks

The rest of the paper is organized as follows: The fundamental backgrounds of the study are briefly reviewed in Section II. The previous BNN hardware accelerators are reviewed in Section III. In Section IV, the proposed design is described in detail. The results of the functional and Monte-Carlo simulations, neural network implementation (using PyTorch package), and the effect of logical errors on the performance and accuracy of the network (considering the logical error rates in the validation process) are presented in Section V. Finally, Section VI concludes the paper.

II. BACKGROUND

A. MAGNETIC TUNNEL JUNCTION

MTJ is a nonvolatile magnetic memory element widely used in magnetic-based circuits. As described in [2], MTJ consists of two ferromagnetic layers known as fixed and free separated by a thin oxide barrier. MTJ has two operation modes. If the fixed and free layers have the same magnetic direction, the MTJ device is in parallel mode and shows a low resistance (Rp). Otherwise, MTJ is in antiparallel mode and shows a high resistance (Rap). The difference between the resistances of an MTJ in the parallel and antiparallel modes is indicated by a criterion called tunnel magnetoresistance (TMR) ratio, defined as $(R_{AP}\text{-}R_P)/R_P$. A TMR ratio of 604% for the in-plane MTJ and 249% for the perpendicular MTJ [11] was obtained at room temperature.

A critical parameter of an MTJ is retention time. MTJs are used as nonvolatile memories, so they must have a reasonable retention time. The retention time of an MTJ is calculated as

$$\tau = \tau_0.\exp(\Delta),\ \ \tau_0 = 1\ ns\ and\ \Delta = \frac{H_k M_s A_r t}{2k_B T} \tag{1}$$

where K_B is Boltzmann constant, T is temperature, H_K is uniaxial anisotropy, M_S is saturation magnetization, A_r is the area of MTJ, t is the thickness of the free layer, and Δ is thermal stability. For a storage class memory, thermal stability should be greater than 75 [12].

B. CARBON NANOTUBE FIELD-EFFECT TRANSISTOR

The structure of a gate-all-around (GAA) CNTFET is shown in [3]. The GAA-CNTFET is a promising alternative for CMOS transistors. Besides the high gate control, electrons and holes have identical mobilities in CNT, and hence, there is no need to size the transistors to compensate for the lower mobility of holes. This feature results in symmetric voltage transfer characteristics for a minimum-size GAA-CNTFET-based inverter. Moreover, the threshold voltage of CNTFET is adjustable through the flat-band voltage (Vfb), which can be used in low-power design. In addition, due to the perfect short channel effect suppression, CNTFET shows a high transconductance, which leads to a sharper transient region in voltage transfer characteristics and higher voltage gain in the CNTFET-based circuits [3], [4]. Accordingly, CNTFET is a promising alternative for the CMOS transistors at deep nanoscale dimensions, especially in battery-operated systems.

C. BINARIZED NEURAL NETWORKS

Efficient hardware implementation of convolutional neural network (CNN) is challenging in hardware and energy limitations applications. Binarization is one of the solutions to overcome this issue [7]. Binarization aims to find the best approximation of CNN by using bit-wise operations instead of floating-point operations. A CNN architecture can be represented as a triplet <X, W, *>, where $X = X_n(n = 1,2,\ldots, N)$ and X_n is the input of the nth layer of CNN. Also, $W = Wm(m = 1,2,\ldots, M)$ and Wm is the m^{th} filter in the nth layer, and M is the number of filters in the nth layer. Moreover, * represents the convolution operation of X and W. There are two approaches to binarization; the first is to binarize weights, and the second is to binarize weights and inputs. In XNOR-Network [7], the second approach is used. W can be estimated by a binary filter $B \in +1, -1$ and a scaling factor α which can be calculated as

$$W = B \times \alpha \tag{2}$$

$$B = sign\{W\} \tag{3}$$

$$\alpha = \frac{1}{k}\|W\|_{n_m} \tag{4}$$

where k is the total number of weights of the m^{th} filter in the n^{th} layer. Also, by using Eq. (5), (6), and (7), the input can be binarized as

$$X^T = H^T \times \beta \tag{5}$$

$$H = sign\{X\} \tag{6}$$

Consequently, the convolution between input X and filter W can be approximated as given by Eq. (8):

$$\beta = \frac{1}{k}\|X\|_{n_m} \tag{7}$$

$$X * W \cong (sign\{X\} \circledast sign\{W\}) \odot \alpha\beta = (H \circledast B) \odot \alpha\beta \tag{8}$$

where $\circledast$ is the XNOR-bitounting operator and $\odot$ is the element-wise product.

III. PREVIOUS WORK

An XNOR/XOR gate based on STT-MTJ for implementing BNN was proposed in [5]. This design's state reading is based on the pre-charge sense amplifier (PCSA). Although the state reading is fast in PCSA-based circuits and has low static power, it imposes a large area overhead. Moreover, it is sensitive to process variations and energetic particle strikes. Also, it needs a pre-charge cycle for each state reading, which increases power consumption. Considering that a massive number of XNOR gates are needed in implementing a BNN, a slight increase in the occupied area and power consumption will significantly increase the total area and power consumption of the BNN hardware.

Another XNOR/XOR gate is proposed in [6] based on SOT-MTJ. This design uses voltage division for the state reading. In this design, it is required to write the input value on MTJ in each cycle of the XNOR/XOR operation. This writing operation will lead to a significant power overhead. Moreover, the reference voltages with slight differences and an analog comparator in the design structure proposed in [6] are susceptible to process variations. Moreover, this XNOR gate needs a negative supply voltage and two reference voltages. In addition, it requires two operational amplifiers, which increase area overhead.

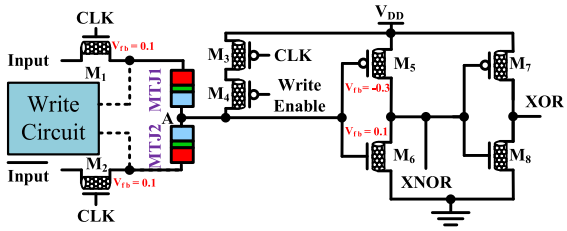

FIGURE 1. The proposed NV-LiM spintronic/CNTFET-based XNOR/XOR gate.

In [13], an XNOR/XOR gate based on memristors is proposed. This gate was designed using two 1-transistor 1-RRAM (1T-1R) cells and a current sense amplifier (CSA). Using a sense amplifier can reduce the read delay, but it is vulnerable to process variations. Furthermore, the CSA circuit consumes much energy and leads to a large overhead area, which is not desirable in implementing BNN in mobile and battery-operated devices.

Considering that the weights have binary values in a BNN, sensitivity to process variations can drastically affect the performance and accuracy of the neural network. Due to the high sensitivity of the designs presented in [5], [6], [13] to process variations, it will have a considerable negative effect on the performance and accuracy of the hardware-implemented network.

IV. PROPOSED DESIGN

The proposed energy and area-efficient nonvolatile XNOR/XOR circuit based on the LiM architecture is shown in Figure 1. This XNOR/XOR circuit works based on voltage division, and the functionality of the proposed gate is independent of the type of MTJ used. Also, every MTJ writing method can be used in this design. However, in the binarized neural network application, since the weights are written one time on MTJs, a simple switching method like STT can be used efficiently. Furthermore, as the nonvolatile storing elements are in the circuit, there is no need to access external memory. Accordingly, the network's overall delay, power, and area overhead are reduced, and the connections will be more straightforward.

To store the logic '1' ('0') as a weight in the MTJ devices, MTJ1 should be in the antiparallel (parallel) mode, and MTJ2 should be in the parallel (antiparallel) mode. The write circuit performs the write operation once after the training process.

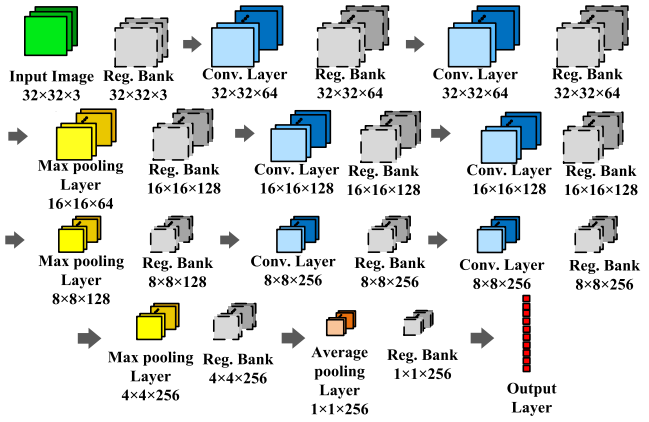

FIGURE 2. The architecture of the proposed BNN hardware accelerator.

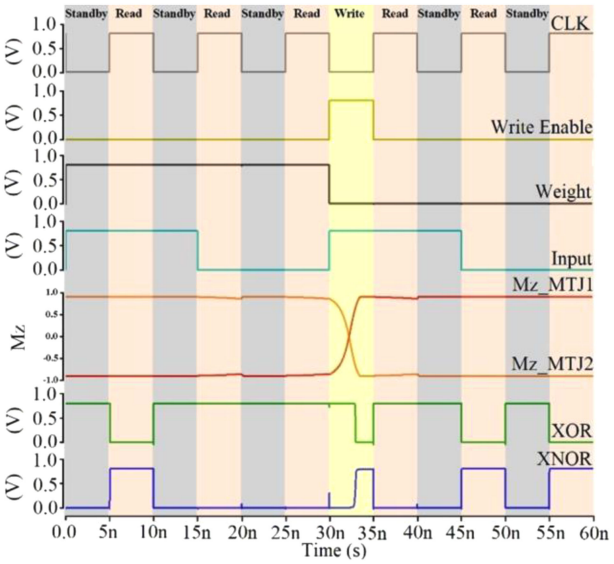

FIGURE 3. Simulation result of the proposed NV-LiM XNOR/XOR gate.

To perform the XNOR/XOR operation (read), the signal CLK is asserted, and the input signal and its complement are fed to the circuit through the transistors M1 and M2, respectively. The XOR of the input and the weight stored in MTJs appears at node A, connected to an inverter gate that corrects the voltage swing and enhances drivability. The output of the first inverter gate is the XNOR of the input and the stored weight. Consequently, the second inverter gate generates the XOR of the input and the stored weight.

When the circuit is in standby mode (no write or read operation is performed), the write circuit is disconnected from MTJs, and the transistors M1 and M2 are off. Consequently, node A becomes float, and its voltage will settle to an intermediate voltage. To prevent the inverters from consuming high static power in this situation, node A is connected to the power supply through p-type transistors M3 and M4. The gates of these transistors are connected to the CLK and Write_Enable signals. Accordingly, when the circuit is in the standby mode (CLK and Write_Enable signals are '0'), transistors M3 and M4 are ON, and the voltage of node A will be equal to the power supply voltage. Moreover, using GAA-CNTFETs with low leakage currents leads to a significantly low leakage power during the standby mode. Leakage power is critical in battery-operated devices, especially those with a significant standby mode, like IoT edge sensors.

The CLK signal is '0' in the write mode, and the Write_Enable signal is '1'. Accordingly, the transistor M4 disconnects the path between the power supply and node A. Moreover, transistors M1 and M2 are off, and the write circuit is connected to the MTJs. The proposed design is compatible with all MTJ write methods, and hence, a simple and efficient switching method like STT can be used. Also, it is noteworthy that the weights are written on MTJs once after the training process in the neural networks, and the network is almost always in the read or standby modes.

Suppose that MTJ1 is in the antiparallel (parallel) mode and MTJ2 is in the parallel (antiparallel) mode (weight '1' ('0') is stored). Let the input be '1'. Accordingly, due to the resistance values of MTJ1 and MTJ2, the voltage of node A will be lower (higher) than the switching threshold of the first inverter, and the XNOR output will be '1' ('0'). For input '0', the voltage of node A will be lower (higher) than the switching threshold of the first inverter gate, and the XNOR output will be '0' ('1').

As the read procedure of the proposed design is performed in a combinational manner requiring no SAs and PCSAs, our design is resilient to process variations. This robustness makes the BNN hardware implementation invulnerable to process variations. Consequently, the BNN hardware based on the proposed design will demonstrate the highest reachable accuracy, equal to the accuracy provided by the software-implemented network.

The architecture of the implemented binary neural network based on the proposed design is shown in Figure 2. The output data of each layer should be stored to be used as the input for the next layer. Therefore, register banks are used to sequence the data and make it possible to use the proposed design as a binarized neural network hardware accelerator.

The proposed binarized neural network hardware accelerator can be used in all of the architectures based on the convolutional neural networks [7]. The implemented network structure in this study is "2×(64C3)-MP2-2 × 2×(128C3)-MP2-2 × 2×(256)-MP2-2-AP4-2-256FC-Softmax" as illustrated in Figure 3. MP, AP, C, and FC stand for max pooling, average pooling, convolution, and full connection in this architecture, respectively. Moreover, MP2-2 denotes max pooling with window size 2×2 and stride2, AP4-2 means average pooling with window size 4×4 and stride 2, 2×(64C3) means two convolutional layers with kernel size 3×3 and 64 feature maps. Also, 256FC shows a fully connected layer with 256 neurons.

As the process variation is unavoidable during the fabrication process of the integrated circuits and can cause logical errors and, consequently, accuracy loss, the implemented

TABLE 1 The critical parameters of the GAA-CNTFET and MTJ.

Description		Value	Variation at the $\pm 3\sigma$ level
CNTFET	Gate length	15 nm	10%
	Gate width	15 nm	10%
	Gate oxide dielectric thickness	4 nm	10%
	CNT diameter	1.2 nm	10%
	CNT-metal contact length	20 nm	10%
MTJ	Oxide barrier thickness	1.1 nm	5%
	TMR under zero voltage bias	300%	10%
	Minimum TMR under operational voltage	270%	–
	Free layer thickness	2 nm	5%
	MTJ surface	60 nm × 60 nm	15%
	Resistant area product (RAP)	10	15%
	MTJ resistance under zero bias voltage	18.1 KΩ–71.7 KΩ	–

network is used to study the effect of the process variations and the logical errors on the accuracy of the hardware-implemented network. In addition, the software implementation gives a good insight into hardware implementation and its challenges.

V. SIMULATIONS

In this section, our proposed and previous designs are simulated, evaluated, and compared comprehensively regarding various aspects of performance and accuracy.

A. FUNCTIONAL SIMULATION AND COMPARATIVE ANALYSIS

The experimentally validated model of CNTFET [14] and the STT-MTJ model based on a numerical solution for the Landau-Lifshitz-Gilbert with dependable precision and the RRAM model [15], [16] have been used to simulate the circuits using HSPICE. Moreover, to have fair comparisons, the previous designs have also been redesigned and optimized based on CNTFETs. Table 1 shows the critical parameters of the MTJ and CNTFET devices [14]–[16]. It is worth mentioning that R_{on}, R_{off}, and C_{in} for a minimum-size CNTFET are 13KΩ, 760MΩ, and 19aF, respectively. Also, according to Eq. (1)., the thermal stability of the MTJ used in this design is 77.4, and the retention time is 4.3×10^{24} seconds.

Figure 3 illustrates the functionality simulation result of the proposed design. In this waveform, Mz_MTJ denotes the magnetic direction of MTJs, and Mz_MTJ '1' ('0') states that the MTJ is in the parallel (antiparallel) mode. When the CLK pulse is '1', the circuit is in the read mode, and the output is valid. When CLK is '0' and Write_Enable is '1', MTJs can be configured by the write circuitry, and when both the CLK and Write_Enable are '0', the circuit is in standby mode.

As shown in Figure 3, at 8ns (18ns), the input is '1' ('0'), the stored weight is '1', and the CLK is '1', and hence, the XOR logic is '0' ('1') and the XNOR logic is '1' ('0'). Moreover, at 38ns (48ns) which is after the change of the stored weight, the input is '1' ('0'), the stored weight is '0', and the CLK is '1' and consequently, the XOR logic is '1' ('0') and the XNOR logic is '0' ('1').

The simulation results of the proposed design and the related designs presented in [5], [6], [13] are given in Table 2. . The average power consumption, delay, power-delay product (PDP), estimated area, and power-delay-area product (PDAP) are evaluated and compared.

The power consumption and delay of the design proposed in [6] are higher than the other designs. It needs to write the input on MTJs in each read cycle, which significantly increases the delay and energy dissipation. It also has two operational amplifiers with two current sources, significantly increasing the power consumption. However, in the proposed design and the designs presented in [5], [13], the learned weights are written on MTJs once after the learning process. Also, the PCSA-based and CSA-based read circuits occupy a large area. Moreover, the pre-charge cycle in each read state and the approach used in the designs proposed in [5], [13] consume higher power than the proposed design. The design proposed in [13] also needs a current source that increases power consumption and area overheads. Furthermore, the designs proposed in [5], [13] are more susceptible to process variations, which causes logical errors and consequently degrades the accuracy and performance of the network.

Our proposed design does not use read circuits with significant area overhead like operational amplifiers or PCSA/CSA-based circuits. Consequently, its area and power dissipation are significantly lower than the designs proposed in [5], [6], and [13]. Accordingly, the proposed design can be considered a promising binarized neural network hardware accelerator in battery-operated devices or other devices with hardware and energy resources limitations.

According to Table 2, our proposed design reduces the power consumption and PDP by 13% to 93% and 29% to 99%, respectively. Moreover, our design improves the area and PDAP, at least by 41% and 58%, respectively, compared to the state-of-the-art counterparts. The proposed design reduces the delay by 18% and 99% compared to the proposed design in [5] and [6], respectively. However, the proposed design in [13] has a lower delay than the proposed design in this paper.

TABLE 2 Performance comparison of the NV-LiM XNOR/XOR circuits.

Design	[6]	[5]	[13]	Proposed
Average Power (nW)	18127	1281	4786	1114
Static Power (nW)	18085	1264	4767	1095
Delay (ps)	2164	11	4	9
PDP (aJ)	39226.82	14.09	19.14	10.02
Area (nm^2)	20876	20876	24560	12280
PDAP ($fJ.nm^2$)	818899.09	294.14	470.07	123.04

B. MONTE-CARLO SIMULATIONS

Monte-Carlo simulations have been carried out to validate the functionality of the designs in the presence of process variations. Table 1 shows that Gaussian distribution and variations at the 3σ level have been considered for the process parameters [2]. Also, the variations for the design presented in [13] have been considered according to the details given in [17]. As the total number of XNOR/XOR gates in the implemented network is 8064, Monte-Carlo simulations with a reasonable number of 10,000 runs have been performed for each design. However, due to the error rate of 0 for the proposed design in this simulation, another Monte-Carlo simulation considering 1,000,000 random samples has also been conducted for the proposed design.

The logical error as a critical parameter has been investigated in the Monte-Carlo simulations to evaluate the impact of the process variations on the functionality of the NV-Lim XNOR/XOR circuits. Due to the binary weights in BNNs, an erroneous weight can significantly affect the accuracy of the network, and hence, the errors caused by the process variations are undesirable. The logical error rate has been calculated by checking the output of each design, considering all of the 10,000 simulation runs.

The results of the Monte-Carlo simulations, for 10,000 Monte-Carlo simulations, are given in Table 4. The Monte-Carlo simulation results show at least 12% improvements in power consumption. Also, our proposed design provides 10% and 99% lower delay than [5] and [6], respectively. However, the design proposed in [13] has a lower delay. Moreover, due to the significantly lower sensitivity of our proposed design to process variations than its counterparts, it reaches the error rate of 0 in the Monte-Carlo simulations with 10,000 runs.

To better estimate the error rate for the proposed design, another Monte-Carlo simulation has been performed considering 1,000,000 random samples, indicating that the proposed design has a failure rate of 0.0164%. Considering the total number of XNOR gates in the implemented network, 8064, this failure rate is negligible. The accuracy of the network implemented using our proposed design is approximately equal to the accuracy of the software implemented network (only a 0.89% reduction in accuracy). However, the design proposed in [20] shows an error rate of 10.5% in the presence of process variations. This is mainly due to the sensitivity of the PCSA circuit to the process variations. Moreover, in this reading strategy, the resistance of each discharge path can easily change by the process variations, which can lead to logical errors. The error rate of the design proposed in [6] is 29.7%, mainly due to the slight differences between the reference voltages and the use of analog operational amplifiers. A slight variation in critical parameters can change the voltages so that logical errors occur. Also, the error rate of the design proposed in [13] is 20.8%, which is primarily due to the sensitivity of the CSA circuit to the process variations.

TABLE 3 Monte-carlo simulation results of the LiM XNOR/XOR circuits.

Designs		[6]	[5]	[13]	Proposed
Power	Minimum (nW)	16348	897	3955	891
	Average (nW)	18127	1295	4799	1123
	Maximum (nW)	20501	1958	5871	1768
	σ/μ (%)	29.17	17.13	17.21	16.12
Delay	Minimum (ps)	2182	6	3	3
	Average (ps)	2641	10	5	9
	Maximum (ps)	3765	20	9	11
	σ/μ (%)	32.3	25.12	25.09	23.4
Error rate (%)		29.7	10.2	20.8	0

Furthermore, the resistance of RRAM is sensitive to process variation, which can change the amount of current passed through the device. Accordingly, as this current is used in the CSA part to generate the output, this variation can significantly increase the error rate. It is noteworthy that due to the lower sensitivity of the proposed design to variations, the CVs of its delay and power are less than the other designs.

C. BINARIZED NEURAL NETWORK APPLICATION

A BNN has been implemented using the PyTorch package in Python for analyzing the functionality and performance of the NV-LiM XNOR/XOR circuits in the hardware implementation of neural networks. The implemented network is shown in Figure 3. The results are used to study the impacts of the process variations and logical errors on the accuracy of the BNN hardware accelerators.

The logical error rate of the NV-Lim XNOR/XOR circuits (given in Table 4) have been applied to the weights of the implemented BNN to consider the impact of the inevitable process variations on the accuracy and performance of the BNNs. The simulation results of the hardware implementation of the BNNs with these weights and its software implementation are given in Table 4.

Due to the weight binarization in BNNs, the process variations can completely flip the value of the weights, which can drastically affect the accuracy of the network. Because our proposed design has no logical error, the accuracy of the implemented BNN based on our design is equal to the software implementation. However, due to logical errors in the gates proposed in [5], [6], [13], the accuracy of the implemented networks based on these gates is degraded. For our design and the designs proposed in [5], [6], [13], 100 BNN validations have been performed, and the logical error rates have been considered by changing the weights randomly.

TABLE 4 Result of the BNN implementation.

Architecture		Software implementation	[5]	[6]	[13]	Proposed
Accuracy (%)	Minimum	74.34	9.10	7.2	8.41	74.34
	Average		18.20	9.8	14.11	
	Maximum		26.20	12.89	16.81	
	σ/μ	0	9.3	10.8	9.8	0
Power (μW)		N/A	10329	146176	38594	8983
Delay (ns)			30	5816	11	24
PDP (pJ)			309	850159	424	215
Area (μm^2)			168	168	198	99

The validation results show a high sensitivity of the BNNs based on these designs to binary weight variations. Furthermore, it is noteworthy that the network's sensitivity to different weights is not the same. This point is considered by performing 100 validation runs to reach reliable results. The minimum, average, maximum, and σ/μ of the accuracies are given in Table 3. The results demonstrate that the proposed design has far higher accuracy than the other designs. Moreover, the results given in Table 3 indicate that our proposed design offers at least 13% lower power, 29% lower PDP, and 41% smaller area in comparison with its counterparts.

VI. CONCLUSION

A new nonvolatile spintronic LiM XNOR/XOR circuit based on CNTFETs and MTJs has been proposed in this paper. The proposed design is used in the hardware implementation of the XNOR network. Due to the massive number of XNOR gates in the structure of BNNs and the demand for using bio-inspired designs in applications with limited hardware and power resources, the proposed design can be used efficiently in such applications. The proposed NV-LiM gate reduces the power, PDP, area, and PDAP at least by 13%, 29%, 41%, and 58%, respectively, compared to its state-of-the-art counterparts. The impact of the logical errors caused by the process variations on the accuracy of BNNs has also been investigated. The accuracy of our proposed design with no logical error reaches the accuracy of the software-implemented network. Moreover, our proposed design offers at least 13% lower power, 29% lower PDP, and 41% smaller area for the complete implementation of the network compared to its state-of-the-art counterparts. Accordingly, the proposed design is a promising candidate for a hardware accelerator in binarized neural networks.

REFERENCES

[1] Z. I. Mannan, S. P. Adhikari, C. Yang, R. K. Budhathoki, H. Kim, and L. Chua, "Memristive imitation of synaptic transmission and plasticity," *IEEE Trans. Neural Netw. Learn. Syst.,* vol. 30, no. 11, pp. 3458–3470, Nov. 2019, doi: 10.1109/TNNLS.2019.2892385.

[2] A. Amirany, M. H. Moaiyeri, and K. Jafari, "Nonvolatile associative memory design based on spintronic synapses and CNTFET neurons," *IEEE Trans. Emerg. Topics Comput.,* vol. 10, no. 1, pp. 428–437, Jan.–Mar. 2020, doi: 10.1109/tetc.2020.3026179.

[3] M. H. Moaiyeri and F. Razi, "Performance analysis and enhancement of 10-nm GAA CNTFET-based circuits in the presence of CNT-metal contact resistance," *J. Comput. Electron.,* vol. 16, no. 2, pp. 240–252, 2017, doi: 10.1007/s10825-017-0980-0.

[4] N. Yang, X. Wang, X. Lin, and W. Zhao, "Exploiting carbon nanotube FET and magnetic tunneling junction for near-memory-computing paradigm," *IEEE Trans. Electron Devices,* vol. 68, no. 4, pp. 1975–1979, Apr. 2021, doi: 10.1109/ted.2021.3059817.

[5] M. Natsui, T. Chiba, and T. Hanyu, "Design of an energy-efficient XNOR gate based on MTJ-based nonvolatile logic-in-memory architecture for binary neural network hardware," *Japanese J. Appl. Phys.,* vol. 58, 2019, Art. no. SBBB01, doi: 10.7567/1347-4065/aafb4d.

[6] A. Samiee, P. Borulkar, R. F. DeMara, P. Zhao, and Y. Bai, "Low-energy acceleration of binarized convolutional neural networks using a spin hall effect based logic-in-memory architecture," *IEEE Trans. Emerg. Topics Comput.,* vol. 9, no. 2, pp. 928–940, Apr.–Jun. 2019, doi: 10.1109/tetc.2019.2915589.

[7] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, *XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks*. Cham, Switzerland: Springer, 2016, pp. 525–542.

[8] A. Amirany, M. H. Moaiyeri, and K. Jafari, "Process-in-memory using a magnetic-tunnel-junction synapse and a neuron based on a carbon nanotube field-effect transistor," *IEEE Magn. Lett.,* vol. 10, no. 2, pp. 928–940, Second Quarter 2019, doi: 10.1109/lmag.2019.2958813.

[9] G. Hills et al., "Modern microprocessor built from complementary carbon nanotube transistors," *Nature,* vol. 572, no. 7771, pp. 595–602, Aug. 2019, doi: 10.1038/s41586-019-1493-8.

[10] M. D. Bishop et al., "Fabrication of carbon nanotube field-effect transistors in commercial silicon manufacturing facilities," *Nat. Electron.,* vol. 3, no. 8, pp. 492–501, 2020, doi: 10.1038/s41928-020-0419-7.

[11] M. Wang et al., "Current-induced magnetization switching in atom-thick tungsten engineered perpendicular magnetic tunnel junctions with large tunnel magnetoresistance," *Nat. Commun.,* vol. 9, no. 1, Feb. 2018, Art. no. 671, doi: 10.1038/s41467-018-03140-z.

[12] A. Nigam, C. W. Smullen, V. Mohan, E. Chen, S. Gurumurthi, and M. R. Stan, "Delivering on the promise of universal memory for spin-transfer torque RAM (STT-RAM)," in *Proc. IEEE/ACM Int. Symp. Low Power Electron. Des.*, 2011, pp. 121–126.

[13] X. Sun, S. Yin, X. Peng, R. Liu, J.-s. Seo, and S. Yu, "XNOR-RRAM: A scalable and parallel resistive synaptic architecture for binary neural networks," in *Proc. Des. Automat. Test Europe Conf. Exhib.*, 2018, pp. 1423–1428.

[14] C. S. Lee, E. Pop, A. D. Franklin, W. Haensch, and H. S. P. Wong, "A compact virtual-source model for carbon nanotube FETs in the Sub-10-nm regime—Part I: Intrinsic elements," *IEEE Trans. Electron Devices,* vol. 62, no. 9, pp. 3061–3069, Sep. 2015, doi: 10.1109/ted.2015.2457453.

[15] Y. Wang, Y. Zhang, E. Y. Deng, J. O. Klein, L. A. B. Naviner, and W. S. Zhao, "Compact model of magnetic tunnel junction with stochastic spin transfer torque switching for reliability analyses," *Microelectronics Rel.,* vol. 54, no. 9-10, pp. 1774–1778, 2014, doi: 10.1016/j.microrel.2014.07.019.

[16] Y. Zhang et al., "Compact modeling of perpendicular-anisotropy CoFeB/MgO magnetic tunnel junctions," *IEEE Trans. Electron Devices,* vol. 59, no. 3, pp. 819–826, Mar. 2012, doi: 10.1109/ted.2011.2178416.

[17] J. Reuben, D. Fey, and C. Wenger, "A modeling methodology for resistive RAM based on stanford-PKU model with extended multilevel capability," *IEEE Trans. Nanotechnol.,* vol. 18, pp. 647–656, 2019, doi: 10.1109/tnano.2019.2922838.