14.1 A 22nm 104.5TOPS/W μ-NMC-Δ-IMC Heterogeneous STT-MRAM CIM Macro for Noise-Tolerant Bayesian Neural Networks

De-Qi You*[1], Win-San Khwa*[2], Bo Zhang[3], Fang-Yi Chen[1], Andrew Lee[1], Yu-Cheng Hung[1], Yi-Ming Li[1], Yu-Hui Wang[1], Chung-Chuan Lo[1], Ren-Shuo Liu[1], Kea-Tiong Tang[1], Chih-Cheng Hsieh[1], Yu-Der Chih[4], Tsung-Yung Jonathan Chang[4], Meng-Fan Chang[1,2]

[1]National Tsing Hua University, Hsinchu, Taiwan
[2]TSMC Corporate Research, Hsinchu, Taiwan
[3]TSMC Corporate Research, San Jose, CA
[4]TSMC, Hsinchu, Taiwan

*Equally Credited Authors (ECAs)

Compute-in-memory (CIM) macros [1-5] for convolutional neural networks (CNNs) [6-7] and vision transformers (ViTs) [8] enable high-performance computing for energy-efficient edge-AI devices. Image recognition applications are constrained by inference accuracy degradation or misjudgments due to environmental noise. Bayesian neural networks (BNNs) [9], which represent weights using its mean (μ-weight) and difference from mean (Δ-weight) outperform CNN and ViT models in terms of noise tolerance, making them promising candidates for edge-AI devices dealing with noisy inputs. Using digital circuits to implement BNN [10-12] imposes a tradeoff between inference accuracy and performance, power and area (PPA).

Combining the CIM structure with STT-MRAM is a promising approach for implementing BNNs, thanks to STT-MRAM's stochastic behavior during write operations, its high endurance, and nonvolatility. As shown in Fig. 14.1.1, implementing a BNN using an STT-MRAM CIM imposes several challenges: (1) The exploration of CIM structures – near-memory-compute (NMC) and in-memory-compute (IMC) – is essential for handling MAC computation with μ/Δ-weight and optimizing the tradeoff between PPA and accuracy; (2) The need for accurate and low-power write operations in STT-MRAM devices is critical due to small write-monitor sense margins (WM-SMs) caused by low TMR, especially when using write stochastics for Δ-weight generation under a given normal, N(μ,σ), distribution; and (3) The degradation of IMC sense margins, caused by near-far effects on BL and a high number of activated WLs, significantly impacts IMC readout accuracy.

This paper proposes the following solutions: (1) A heterogeneous CIM structure using NMC for μ-compute and IMC for Δ-compute (μN-ΔI) to achieve high energy efficiency and inference accuracy; (2) A self-compare write-termination (SCWT) scheme to enhance WM-SMs for accurate Δ-weight generation with a minimal PPA impact; and (3) 2D clamping-voltage scaling with sense-margin compensation (2D-CVS-SMC) that improves IMC readout accuracy while reducing power consumption. The test chip, fabricated using foundry-provided 22nm STT-MRAM, achieves 104.5TOPS/W under an 8bIN - 8bμ - 8bΔ - 27bOUT configuration. In the presence of normalized RGB noise, σ=0.01, the inference accuracy degradation of the μN-ΔI STT-MRAM CIM is 2.55× lower than a simulated CNN.

Figure 14.1.2 illustrates the process for generating BNN weights using the μN-ΔI STT-MRAM CIM macro with the SCWT scheme. The BNN weights, N(μ,σ)'s, are generated through offline training. In Figure 14.1.2, 576 N(μ,σ)'s are considered as an example, with 576 μ-weights being directly stored in the μ-array. The user utilizes STT-MRAM write stochastics to generate corresponding 576 Δ-weights. After chip fabrication, the user first establishes a switching-probability (p) table for the STT-MRAM corresponding to the write voltage (V_W) and V_W pulse width (PW). The user then uses this table and the desired σ to generate the corresponding m-bit PW-code via software, which is input into the STT-MRAM stochastic Gaussian random number generator (SS-GRNG). The SS-GRNG consists of a PW generator, an FSM, and an SCWT circuit. The PW generator converts an m-bit PW-code into the V_W PW, which corresponds to a specific p. This PW is provided to the SCWT for central-limit theorem (CLT)-based Δ-weight generation [13]. SCWT tracks near-far effects on BL and PVT variations without the need for an on-/off-chip reference voltage or current. It samples the write cell current (I_{SAMP}) at the onset of the PW (before the STT-MRAM switches) and waits for a change in I_{DL} (ΔI_{DL}). If $\Delta I_{DL} \neq 0$, then the STT-MRAM cell has switched, and SCWT terminates the write process. This operation is repeated N_0 times, with the total switch count X used to calculate Δ-weight = X - $N_0 \times p$. The Δ-weights are then stored in the Δ-array. The user can regenerate a new set of Δ-weights as needed for different applications.

Figure 14.1.3 illustrates the proposed μN-ΔI scheme. Preliminary analysis reveals that the μ-compute error (μ-error) has a significant impact on inference accuracy, whereas Δ-compute error (Δ-error) has a negligible impact. Thus, we mapped μ-compute to NMC (more accurate) and Δ-compute to IMC (more energy-efficient). The μN-ΔI architecture uses a custom input register (IN-DFF) to reduce hardware area and power consumption; it also uses a dedicated weight mapping scheme to reduce latencies associated with heterogeneous computation between NMC and IMC. During computation, μ-NMC operates using word-wise inputs and bit-wise weights with only one WL activated at a time. This enables the completion of an 8bIN-1bμ MAC in 1 cycle; thus, an 8bIN-8bμ MAC requires 8 cycles. Conversely, Δ-IMC uses a bit-wise inputs, which activates multiple WLs simultaneously, and word-wise weights, enabling the completion of a 1bIN-8bΔ MAC in 1 cycle: an 8bIN-8bΔ MAC requires 8 cycles. Note that the μ-NMC and Δ-IMC share a 72 × 8b custom IN-DFF. For the 72-DL μ-NMC, each DL is mapped to an 8b input, fully utilizing the custom IN-DFF. On the other hand, Δ-IMC uses a 6b selection signal (IN-SEL [5:0]) to select 9 consecutive bit-wise inputs ($IN_{0\sim8}$[k]), sending it to 9 WLs for up to 9 BL accumulations. For instance, when performing a NN-layer computation, with 64 channels and 576 accumulations, the 8 μ-NMCs (72 DL/bank × 8 banks = 576b wide-IO) concentrate throughput entirely on a single channel MAC, producing a complete channel output every 8 cycles. This configuration enables the completion of 64-channel outputs in 512 cycles. Meanwhile, the 8 Δ-IMCs (9 BL accumulations, 8 MAC-channel/bank × 8 banks = 64 MAC-channels) distribute throughput evenly across the 64-channel MACs, allowing for the simultaneous completion of 64-channel outputs in 512 cycles. Finally, the μ-Δ combiner integrates and outputs 64 sets of μ-MAC and Δ-MAC results.

Figure 14.1.4 illustrates the proposed 2D-CVS-SMC scheme in Δ-IMC: consisting of 2D clamping-voltage scaling (2D-CVS) and 2D sense-margin compensation (2D-SMC) to enhance MAC sense margins (SM) and reduce ADC power consumption during MAC computation. Unlike the scheme in [14], which adjusts the clamping-voltage (V_{CLP}) based only on the WL address (XA), 2D-CVS adjusts V_{CLP} in two dimensions, based on XA and the number of activated WLs (N_{WL}). The 2D adjustment reduces the average BL current and improves MAC energy efficiency (EF_{MAC}), while not degrading the worst-case SM. Following chip fabrication, the 2D-CVS establishes a 2D-V_{CLP} scaling table for each Δ-array; where, a 2b XA [7:6] and a 4b N_{WL} (representing the number of logic-1s in $IN_{0\sim8}$[k]) correspond to a 3b bias index. This index is stored in the STT-MRAM and is accessed on wake up, where it selects 1 of 8 V_{CLP} levels (V_{CLP} [7:0]) to be $V_{2D\text{-}CLP}$. For example, for the case (case-1) where XA is far and N_{WL} is large, the bias index is larger, corresponding to a larger $V_{2D\text{-}CLP}$; and for case-2, where XA is near and N_{WL} is small, the bias index is smaller, resulting in a smaller $V_{2D\text{-}CLP}$. The $V_{2D\text{-}CLP}$ selection operation is performed during WL development so that it does not contribute to overall latency. The 2D-SMC operation is divided into four phases (PH0 - 3): in PH0 (standby, overlapping with 2D-CVS): C1 is charged to V_{DD}, while X0 and X1 are discharged to V_{SS}. In PH1 (current sampling): SW1 is turned on to sample the MAC current (I_{MAC}) from the Δ-array. C1 records the voltage that mirrors I_{MAC} from P1 to P2. In PH2 (distortion catch): SW1 is turned off, while SW2 and SW3 are turned on. V_{X0} is determined by I_{MAC} mirrored from P2 to P3 and the duration for which SW2 is on. A larger MAC value (MACV) results in a larger I_{MAC}, and thus a larger V_{X0}. The duration of SW2 can be adjusted according to test results to mitigate process variation. V_{X1} is determined by the difference between the 2D reference current ($I_{2D\text{-}REF0}$) generated from the Δ-array (with the same XA and N_{WL}) and I_{N4}. $I_{2D\text{-}REF0}$ tracks near-far effects on BL, so if $I_{2D\text{-}REF0}$ is smaller at a far XA, V_{X1} is smaller, leading to a larger gate-to-source voltage for N5 ($V_{X0\text{-}X1}$), and vice versa. In PH3 (margin compensation and offset cancellation): SW2 is turned off. The compensation current ($I_{2D\text{-}BOOST}$) determined by $V_{X0\text{-}X1}$ is mirrored from P4 to P5. A current subtractor then cancels the offset currents of $I_{2D\text{-}REF0}$, resulting in $I_{2D\text{-}OUT} = (I_{MAC} + I_{2D\text{-}BOOST} - I_{2D\text{-}REF0})$. With a smaller sense current, $I_{2D\text{-}OUT}$, the ADC power consumption is consequently reduced.

Figure 14.1.5 illustrates the simulated performance of the proposed schemes. SCWT enhances WM-SM by 3.25 to 3.74× and reduced write energy by 2.46 to 3.11×, compared to both the baseline and conventional write termination [15] schemes. μN-ΔI decreases macro-MAC power by 43% with an inference accuracy loss of only 0.05%, compared to μ-Δ all NMC. 2D-CVS-SMC improves IMC EF_{MAC} by 2.48× and MAC SM by 1.90×, compared to conventional static-BL clamping. The overall macro-level EF_{MAC} increases by 1.72× with only a 0.03% inference accuracy loss, compared to μ-Δ all NMC.

Figure 14.1.6 presents measurement results from the 22nm STT-MRAM CIM macro and a comparison table to prior silicon-verified works. When the ResNet-20 model is applied to the CIFAR-100 dataset, the measured inference accuracy degradation is 1.19% with normalized RGB noise N(0, 0.01), which is 2.55× lower than the simulated CNN. The Shmoo plot for the STT-MRAM CIM macro confirms an 18ns computing latency ($T_{CD\text{-}MAC}$) from a 0.8V supply using an 8bIN - 8bμ - 8bΔ - 27bOUT configuration. Figure 14.1.7 presents the die photo and a summary table of key metrics.

Acknowledgement:
The authors would like to thank NTHU-TSMC JDP, NSTC, and TSRI for financial and manufacturing support.

 979-8-3315-4101-9/25/$31.00

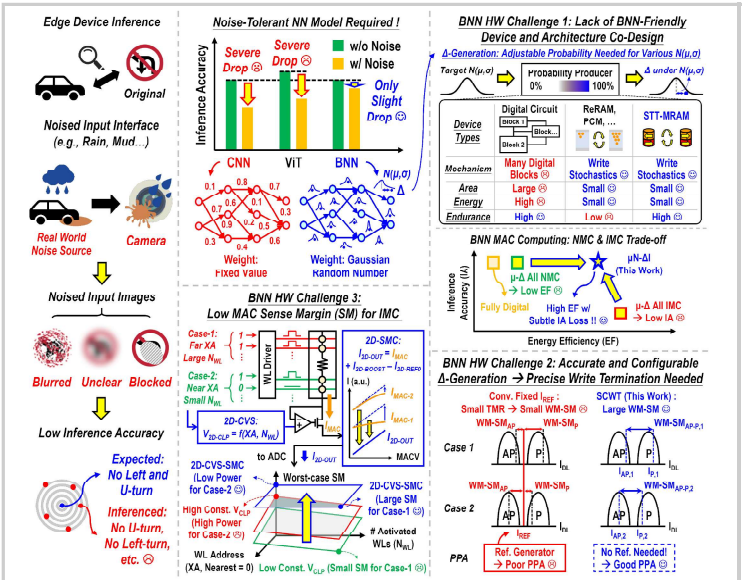


Figure 14.1.1: Advantages of BNN software and challenges of hardware implementation.

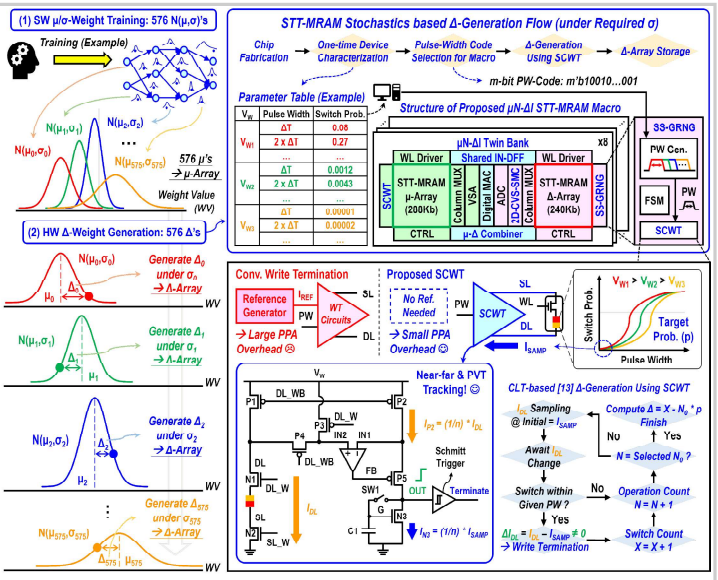


Figure 14.1.2: BNN-weight-generation process flow within the μN-ΔI STT-MRAM CIM macro using SCWT.

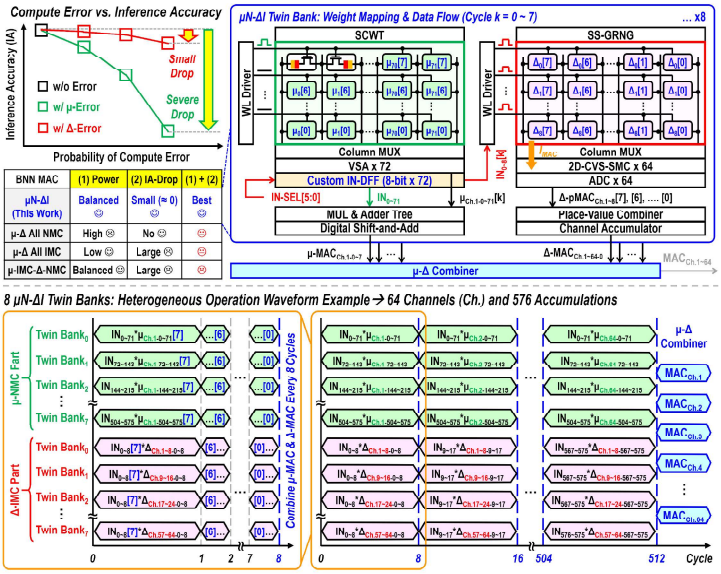


Figure 14.1.3: BNN-MAC computation flow within the μN-ΔI STT-MRAM CIM macro.

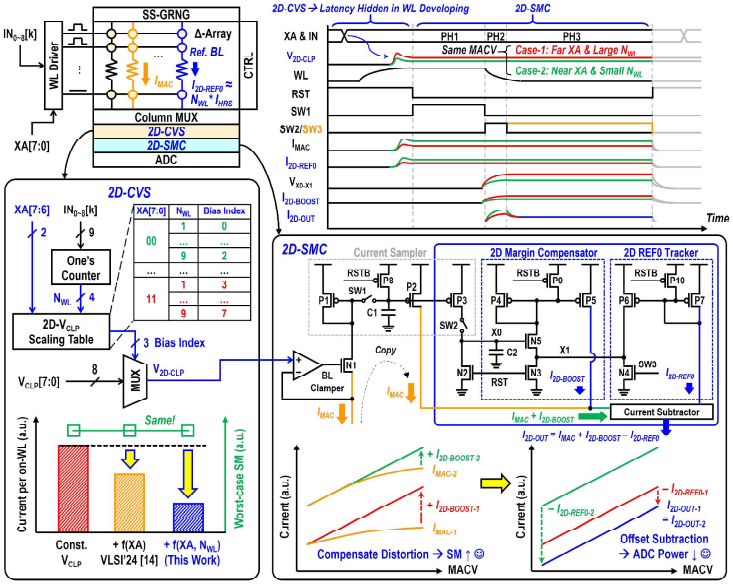


Figure 14.1.4: IMC readout with 2D-CVS-SMC.

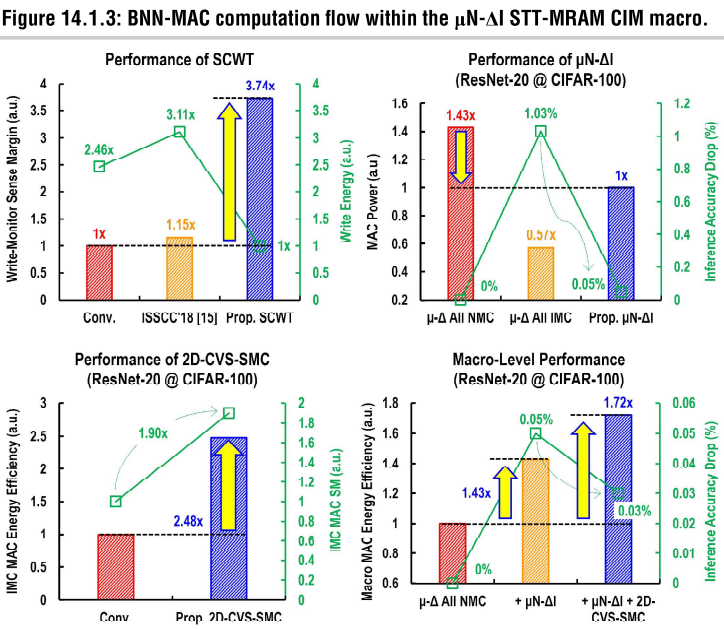
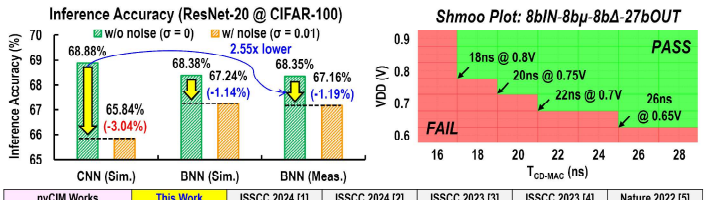


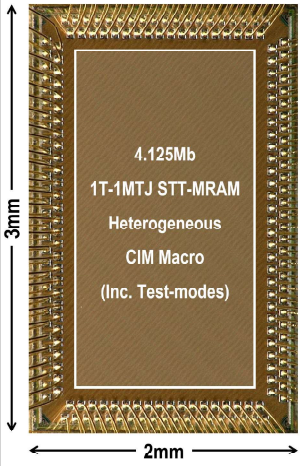
Figure 14.1.5: Simulated performance of the proposed schemes.



nvCIM Works	This Work	ISSCC 2024 [1]	ISSCC 2024 [2]	ISSCC 2023 [3]	ISSCC 2023 [4]	Nature 2022 [5]
Technology	22nm	22nm	14nm	22nm	28nm	28nm
NVM Device	STT-MRAM	ReRAM	Flash	STT-MRAM	STT-MRAM	STT-MRAM
Application	Bayesian NN/CNN	CNN	Plastic NN	CNN	CNN	CNN
Input Noise Tolerance	High ☺	Low ☹	Low ☹	Low ☹	Low ☹	Low ☹
CIM Mode	NMC-IMC Heterogeneous	IMC	IMC	NMC	IMC	IMC
Active Area (mm²)	3.49[1]	8.2	3.875	18	4.5	2
Capacity	4.125Mb	16Mb	32Kb	8Mb	2Mb	4Kb
Supply Voltage (V)	0.65-0.8	0.7-0.8	0.8	0.65-0.8	0.85-1	0.8-1
Input Precision	INT8	FP16, BF16	INT1/2/8	INT4/8	Binary	Binary
Weight Precision	INT8 (μ-weight) INT8 (Δ-weight)	FP16, BF16	INT5/8 (W_1) INT1 8 (W_2)	INT4/8	Binary	Binary
Accumulation Number	576 (NMC), 9 (IMC)	Not Reported	Not Reported	576	9	64
Output Precision	27b	FP32	20b (MVM) 28b (MEM-MVM)	16b/22b/26b	Binary/5b	4b
Throughput[2] (TOPS, TFLOPS)	0.128[3]	0.86 (BF16) 0.76 (FP16)	0.1536 (8bIN-8bW_1-1bW_2)	0.59 (INT8)	Not Reported	Not Reported
Energy Efficiency (TOPS/W, TFLOPS/W)	104.5	31.2 (BF16) 28.7 (FP16)	22.64 (8bIN-8bW_1-1bW_2)	46.4 (INT8)	35.2 (Binary)	405 (Binary)
Compute Density[4] (TOPS/mm², TFLOPS/mm²)	0.036	0.104 (BF16) 0.095 (FP16)	0.15 (8bIN-8bW_1-1bW_2)	0.033 (INT8)	0.55 (Binary)	4.43 (Binary)
Inference Accuracy Loss	0.03%[5] (CIFAR-100)	0.32%[6] (CIFAR-100)	1.38%[7] (Omniglot)	No (CIFAR-100)	2.9%[8] (CIFAR-10)	2.01%[9] (MNIST)

[1] Including test-modes. [2] One operation (OP) represents one multiplication or addition. [3] Peak performance measured @ 0.8V. [4] Compute Density = Throughput / Active Area. [5] SW baseline was 68.38% (using ResNet 20 BNN model under INT8 precision). [6] SW baseline was 69.8% (using ResNet 20 model under BF16 precision). [7] SW baseline was 79.42%. [8] SW baseline was 88.5% (using LeNet model under binary precision). [9] SW baseline was 95.24% (using VGG model under binary precision).

Figure 14.1.6: Measurement results and comparison table to prior works.



Chip Summary		
Technology	22nm CMOS process	
NVM Device	Foundry 1T-1MTJ STT-MRAM	
Application	Bayesian NN/CNN	
Test-chip Size	3mm x 2mm (including IO-pads and test-modes)	
Capacity	4.125Mb (8 sub-banks)	
Supply Voltage	0.65 – 0.8V	
Input Precision	INT8	
Weight Precision	INT8 (μ-weight), INT8 (Δ-weight)	
Accumulation Number	NMC Part	576
	IMC Part	9
Output Precision	27b *1	
Throughput (TOPS)	0.128 *2	
Energy Efficiency (TOPS/W)	104.5 *3 – 167.7 *4	
Compute Density (TOPS/mm^2)	0.036 *2	

*1 Bit-width after NMC & IMC outputs combination
*2 Peak performance measured @ 0.8V
*3 Performance measured @ 0.8V under 50% input sparsity (ResNet-20)
*4 Performance measured @ 0.65V using 90% input sparsity (ResNet-20)
*5 One operation (OP) represents one integer multiplication or addition

Figure 14.1.7: Die photo and key-metric summary table.

References:
[1] T.-H. Wen et al., "A 22nm 16Mb Floating-Point ReRAM Compute-in-Memory Macro with 31.2TFLOPS/W for AI Edge Devices," *ISSCC*, pp. 580-582, 2024.
[2] L. Wang et al., "A Flash-SRAM-ADC-Fused Plastic Computing-in-Memory Macro for Learning in Neural Networks in a Standard 14nm FinFET Process," *ISSCC*, pp. 582-584, 2024.
[3] Y.-C. Chiu et al., "A 22nm 8Mb STT-MRAM Near-Memory-Computing Macro with 8b-Precision and 46.4-160.1TOPS/W for Edge-AI Devices," *ISSCC*, pp. 496-498, 2023.
[4] H. Cai et al., "A 28nm 2Mb STT-MRAM Computing-in-Memory Macro with a Refined Bit-Cell and 22.4 - 41.5TOPS/W for AI Inference," *ISSCC*, pp. 500-502, 2023.
[5] S. Jung et al., "A crossbar array of magnetoresistive memory devices for in-memory computing," *Nature*, vol. 601, pp. 211-216, 2022.
[6] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *IEEE Conf. on Comp. Vision and Pattern Recognition*, pp. 770-778, 2016.
[7] M. Sandler, et al., "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *IEEE/CVF Conf. on Comp. Vision and Pattern Recognition*, pp. 4510-4520, 2018.
[8] S. Mehta and M. Rastegari, "MobileViT: Light-weight, General-purpose, and Mobile-friendly Vision Transformer," *Intl. Conf. on Learning Representations*, 2022.
[9] C. Blundell, et al., "Weight uncertainty in neural networks," *Intl. Conf. on Machine Learning*, pp. 1613-1622, 2015.
[10] H. Fan et al., "FPGA-Based Acceleration for Bayesian Convolutional Neural Networks," *IEEE TCAD*, vol. 41, no. 12, pp. 5343-5356, Dec. 2022.
[11] R. Cai et al., "VIBNN: Hardware Acceleration of Bayesian Neural Networks," *ACM Special Interest Group on Programming Languages Notices*, vol. 53, no. 2, pp. 476-488, Nov. 2018.
[12] H. Awano and M. Hashimoto, "BYNQNet: Bayesian Neural Network with Quadratic Activations for Sampling-Free Uncertainty Estimation on FPGA," *Design, Automation & Test in Europe Conference & Exhibition*, pp. 1402-1407, 2020.
[13] X. Jia et al., "An Energy-Efficient Bayesian Neural Network Implementation Using Stochastic Computing Method," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 35, no. 9, pp. 12913-12923, Sept. 2024.
[14] D.-Q. You et al., "A 22nm Nonvolatile AI-Edge Processor with 21.4TFLOPS/W using 47.25Mb Lossless-Compressed-Computing STT-MRAM Near-Memory-Compute Macro," *IEEE VLSI Symp. Tech. and Circuits*, 2024.
[15] Q. Dong et al., "A 1Mb 28nm STT-MRAM with 2.8ns read access time at 1.2V VDD using single-cap offset-cancelled sense amplifier and in-situ self-write-termination," *ISSCC*, pp. 480-482, 2018.

979-8-3315-4101-9/25/$31.00