37.6 A 22nm 60.81TFLOPS/W Diffusion Accelerator with Bandwidth-Aware Memory Partition and BL-Segmented Compute-in-Memory for Efficient Multi-Task Content Generation

Yiqi Jing[1], Jiaqi Zhou[1], Yiyang Sun[1], Siyuan He[1], Peiyu Chen[2,3], Ru Huang[1], Le Ye[1,2], Tianyu Jia[1]

[1]Peking University, Beijing, China
[2]Advanced Institute of Information Tecnhology of Peking University, Hangzhou, China
[3]Nano Core Chip Electronic Technology, Hangzhou, China

2025 IEEE International Solid-State Circuits Conference (ISSCC) | 979-8-3315-4101-9/25/$31.00 ©2025 IEEE | DOI: 10.1109/ISSCC49661.2025.10904573

Initially applied for image synthesis [1], Diffusion Models (DMs) have been rapidly expanded into many content-generation tasks, e.g. 3D scenes [2-3] or video [4], and deliver exceptional performance. Figure 37.6.1 provides an overview of DM architecture, which typically processes random noisy input through multiple, i.e. 20-50, denoising steps to generate the desired output content. Each denoising step incorporates a U-Net structure with a down-sampling encoder and an up-sampling decoder, which contains repetitive transformer blocks. To support diverse content generation, multi-view [5] or temporal attention blocks [6] are integrated to enhance 3D scene or video-frame consistency. Due to the large number of denoising steps, generating a single piece of content consumes significant latency, e.g. ~70s for a 4 seconds of 6fps video on an A100 GPU. To improve the hardware performance and efficiency, compute-in-memory (CIM) accelerators have been developed for transformers [7-9] or DMs [10]. However, for several reasons, it is still challenging to use existing CIM-based accelerators for practical image or multi-task DMs. First, a significant operational intensity (OpI) variation exists along DM layers leading to dynamic memory bandwidth (BW) requirements. The DMs also require excessive data storage, i.e. ~10× larger than VAEs and GANs . Second, the impressive CIM macro efficiency is often significantly degraded at the system-level due to the diminishing reuse rate for large AI models. Moreover, ~59% of CIM macro power is consumed on data access rather than computation. Third, emerging DM tasks for 3D or video require additional consistency operations, i.e. multi-view/frame attention, for smooth transitions across viewpoints or frames. Such consistency attention is quite computationally-intensive and comprises ~31% of the total operations.

To address these challenges, this paper presents a digital CIM-based accelerator for multi-task DMs with following key features: 1) a dynamic BW-aware memory partitioning scheme is developed with dense on-chip eDRAM storage to optimize CIM utilization and reduce EMA, 2) a bitline (BL)-segmented CIM cluster is designed with reuse-aware weight reordering to enhance system efficiency, and 3) a hierarchical consistency optimization flow is presented to minimize frame/pixel-level operations to improve performance. Overall, these innovations enable our chip to achieve a 60.81TFLOPS/W system efficiency, which has 1.4× better performance than a prior image DM chip [10] and our chip also shows promising performance for more diverse content-generation tasks.

Figure 37.6.2 shows the overall architecture of our chip. It comprises a CIM acceleration subsystem with 9 CIM clusters, each contains four 24Kb digital-CIM macros, 3Mb on-chip eDRAM, a multi-frame consistency management (MFCM) unit, a host RISC-V CPU and peripheral circuits. Direct connections are implemented between CIM clusters to support pipeline parallelism and crossbar (Xbar) interfaces are used to connect eDRAM. A reuse-aware weight update scheduler is incorporated inside the CIM subsystem to improve the data reuse rate for all clusters. The eDRAM is designed into four splittable banks and each has 8×3 32Kb 3T gain-cell arrays. A leak-tracking reference column is added in each array to enhance Vref accuracy and extend eDRAM retention time (Fig. 37.6.3). A dynamic BW-aware partitioning module is designed with an interconnect coupler to support flexible memory bank partitioning to adapt to varying OpI of different DM layers. The MFCM performs the hierarchical consistency optimizations, which contains a progressive view extension unit for frame-level optimizations, along with a spatial-temporal compress unit and epipolar-attention sparse unit to reduce pixel-level computations.

Figure 37.6.3 illustrates the architecture and operation of our BW-aware memory partitioning. A two-step partitioning scheme is developed for each runtime subtask, i.e. tiled matrix workload in DM layers. First, the computation and memory resource demand for each subtask are evaluated based on operational intensity and recorded in a resource-aware subtask table. Second, the data arrangement within eDRAM, i.e. eDRAM bank partitioning plan, is determined by the required BW and stored in a memory-partition table. To maximize utilization, a time-multiplexed BW distribution method is adopted to decouple BW from memory capacity, e.g. data for subtask 5 is stored equally in ten eDRAM columns to increase available peak BW. Compared to a conventional fixed memory allocation (Case 1), which evenly distributes resources to subtasks, our BW-aware partitionng jointly optimizes CIM and BW utilization for subtasks with diverse OpI, e.g., subtasks 3-5 have improved capacity and BW utilization by 1.98× and 2.76× in Case 2. Based on the partitioning plan, area-efficient crossbar and interconnect couplers are reconfigured for proper topology and flow control. The coupler determines source and destination addresses through a loop decoder, while a credit noter monitors eDRAM refreshes and CIM backpressure to ensure reliable data transfer. Overall, the BW-aware memory partitioning improves CIM and BW utilization by 1.27-9.54× and 1.51-10.19× across DM layers for a Wonder3D model, leading to a total 2.68× performance gain with only 3.7% area overhead.

Figure 37.6.4 illustrates the implementation details of the CIM cluster and the weight-update scheduler. Each cluster integrates four BL-segmented CIM macros, an aggregator unit, and a local NoC. Each macro contains six 128×32 6T SRAM MAC-arrays, pre/post-processing units, a weight-alignment unit, and I/O buffers. The CIM array is designed with an architecture of 4×64 weight subarrays, which consist of 16 SRAM cells with a local stationary unit (LSU), and a LUT-bypass adder tree. To alleviate costly data access, a 2-stage BL-segmentation technique is incorporated, which segments BL based on operational addresses at both the MAC-array level and a finer-grained subarray level. This approach reduces the effective BL loading and unnecessary precharging during SRAM access to enhance CIM efficiency by 27%. A LUT-bypass adder tree is designed by leveraging a 4b sparse LUT as a multiplier and a first-stage adder to reduce high dynamic transitions. A bypass adder is used as the second-stage adder to skip zeros, resulting in 11% power reduction. During computation, the weight-update scheduler supports simultaneous computation with our reuse-aware weight reordering, in which a top-k module is used to generate a sparse attention pattern to indicate the weights to be reused in the CIM subarray. The pattern is first row-wise reordered via a reuse-driven activation sorter by similarity comparisons, forming the CIM's activation sequence. Then, a column-wise reordering based on weight lifespan decides the weight-update sequence with the least-recently-used queue-replacement policy, thereby reducing memory access for attention layers by 29%. Overall, the BL-S CIM macro achieves 1.31× performance gain and 54% energy savings by the above techniques.

Figure 37.6.5 depicts our MFCM scheme for efficient content generation. Conventional multi-view DMs denoise frames from all viewpoints and apply multi-view attention on pixels to ensure consistency, resulting in significant overhead. We develop a hierarchical consistency computation flow with both frame-level and pixel-level optimizations with following three stages. Stage 1 adopts a progressive view extension technique, which utilizes fewer viewpoint frames at initial timesteps to reduce frame-level computations, e.g. only 2 frames at timestep 5. New frames from different viewpoints will be added by duplicating the previous frame once its similarity with reference frame (calculated by a frame compare unit) reaches predefined threshold. Stage 2 reduces pixel-level computation by skipping background and trivial pixels using a spatial-temporal compression unit. The target object is segmented from the background using RGBA values in salient object detection unit (SODU) based on a spiral search pattern. Unmodified pixels are further skipped in trivial-pixel sparsity unit (TPSU) by similarity assessment across denoising iterations. Stage 3 further leverages a pixel-level epipolar-attention mechanism to minimize irrelevant inter-frame interactions. The epipolar solver controls a 3×3 MAC array to generate the pixels and epipolar line on each view plane associated with the target light ray. A sparse attention is performed between the epipolar line and the pixel, e.g. pixel on the P3 frame and the epipolar line on the P0 frame, to mask out irrelevant regions and reduce computations. The above hierarchical consistency optimizations bring us a 3.71× speedup and 68% energy saving.

Figure 37.6.6 shows the measurement results of the 22nm CIM processor. Multiple content-generation tasks are evaluated using SOTA DMs, i.e. SD-v1.5, Wonder3D, SVD, based on customized hybrid BF16-W4A8 quantization. Compared to a SOTA image DM accelerator [10], our chip achieves a 1.4× performance improvement. Since there is no prior accelerator for 3D or video, we also provide the execution time for 3D and video models with high system efficiency. The BW-aware partitioning, BL-S macros and MFCM together contribute to 13.03× performance and 3.69× system-efficiency improvement, leading to a 49.74-60.81TFLOPS/W system efficiency (1.52× better than [10]). Our CIM macro FoM, which considers both energy and area efficiency, is 1.28× higher than the Booth8 CIM in [10]. The system FoM is 1.44× and 3.82× higher than SOTA CIM [8] and DM [11] accelerators, illustrating better optimized CIM computation and BW utilization. Figure 37.6.7 shows the die photo and more specifications.

Acknowledgement:
This work was supported in part by NSFC Grant 92164301, Grant 62225401, and Grant U23A6007; Zhejiang Provincial Key R&D program under Grant 2021C01035; Grant QYJS-2023-2401-B, and Grant QYJS-2023-2402-B. Corresponding authors: Tianyu Jia and Le Ye.

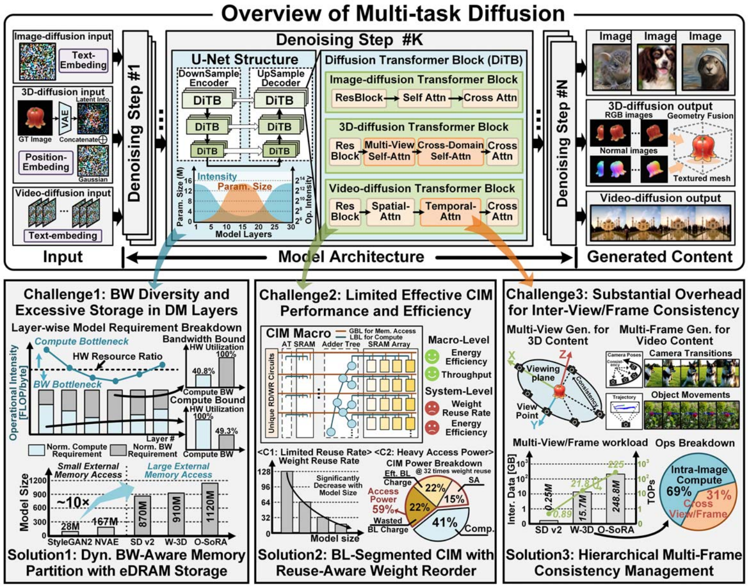


Figure 37.6.1: Overview of diffusion models for multi-task content generation and the deployment challenges.

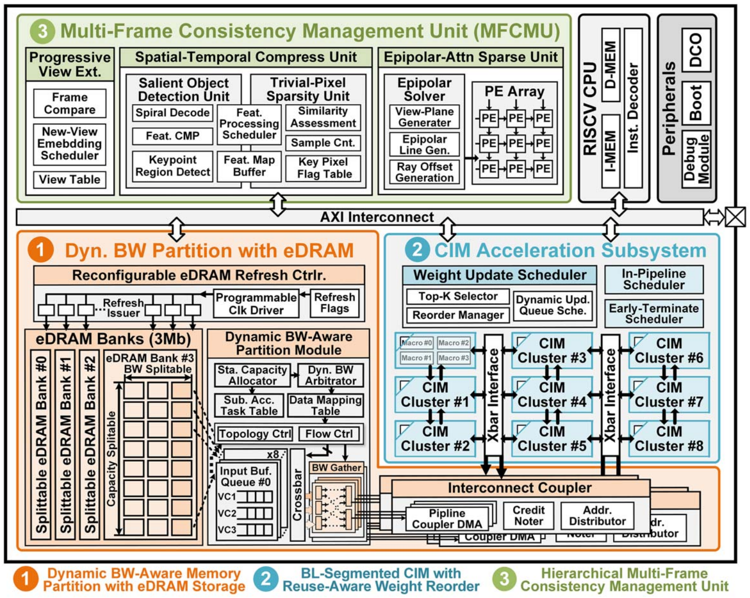


Figure 37.6.2: Overall chip architecture.

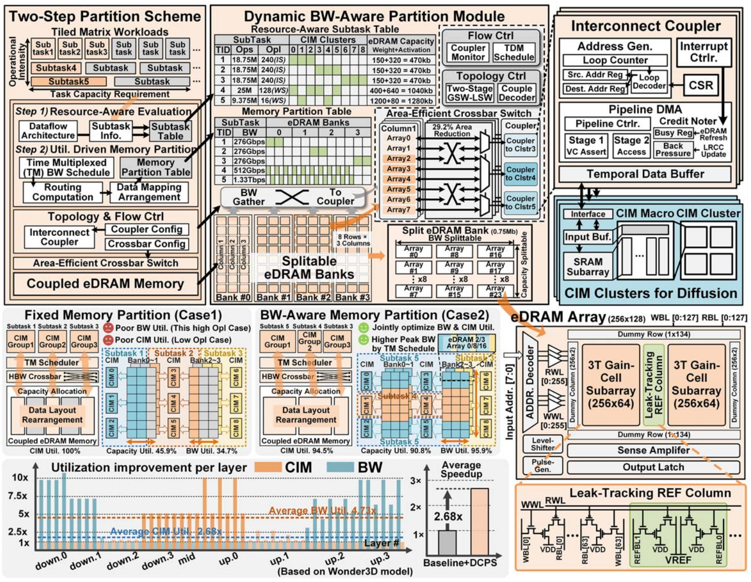


Figure 37.6.3: Dynamic BW-aware memory partitioning scheme with eDRAM to jointly improve CIM and BW utilization.

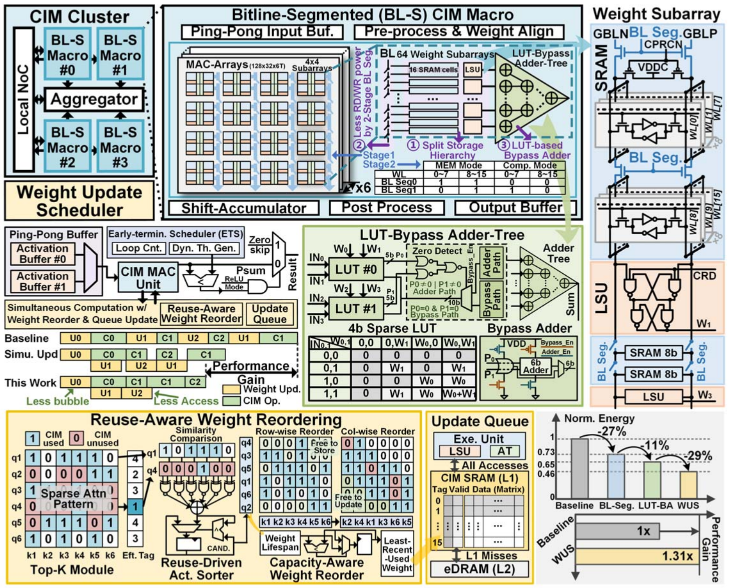


Figure 37.6.4: Bitline-segmented (BL-S) CIM macro and weight update scheduler with reuse-aware reordering update.

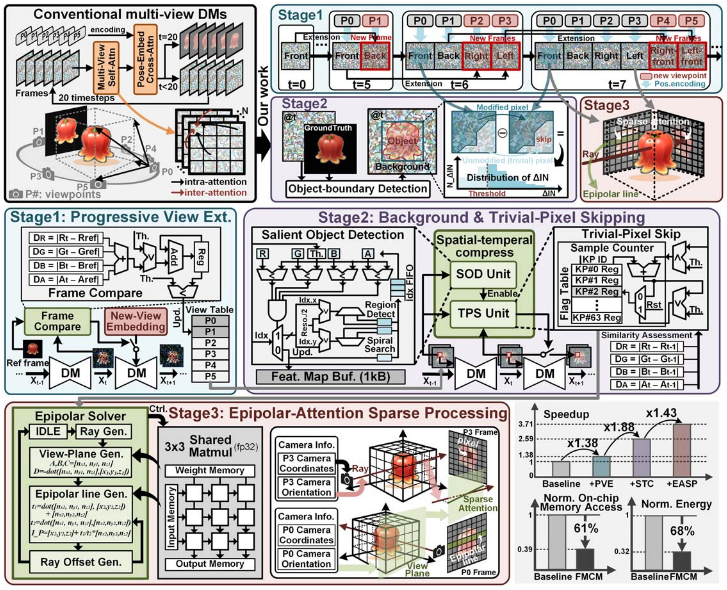
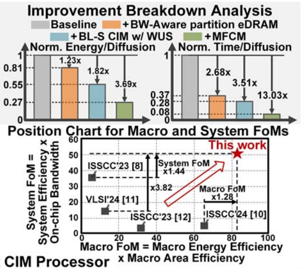


Figure 37.6.5: Multi-frame consistency management with hierarchical optimizations for diverse content generation.

Improvement Breakdown Analysis

Model		SD-v1.5	Wonder-3D	SVD
Task		2D Image	3D Modeling	3D/4D Video
Dataset		LION-5B (64*64)	Objaversr (32*32)	Panda-70M
Parameters (MB)		870	910	1504
Precision	Conv	INT8	INT8	INT8
	Attention	BF16	BF16	BF16
Baseline Accuracy		17.77(FID↓)	0.119(LPIPS↓)	136.87(FVD↓)
Chip Accuracy		18.69	0.116	141.13
Execution Time(s)[1]		13.092[3]	5.404	1392.641
Macro Energy Efficiency (TFLOPS/W)[2]		132.97	101.97	76.69
System Energy[2] Efficiency (TFLOPS/W)		60.81	54.91	49.74

1. Measured @478MHz, 0.9V; 2. Measured @180MHz, 0.6V;
3. 18.378s reported in [10], off-chip memory access is excluded.

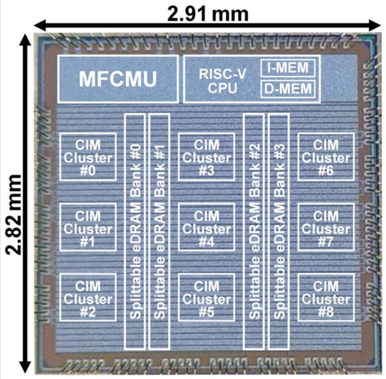


Comparison with State-of-the-art CIM Processor

	ISSCC'24 [10]	VLSI'24 [11]	ISSCC'23 [8]	ISSCC'23 [12]	This Work		
Technology [nm]	28	22	28	28	22		
Task	Diffusion	Diffusion	Transformer	CNN	Diffusion		
Content Generation	Image	Image	\	\	Image	3D	Video
Data Precision	INT10/16, FP/BF16	FP16	INT8/16	INT4/8, FP/BF16	INT4/8/12, FP8, BF16		
Chip Area [mm^2]	3.67	3.7	14.36	4.54	8.2		
Supply Voltage [V]	0.6 - 1.0	0.6 - 1.0	0.6 - 1.0	0.397 - 0.9	0.6 - 0.9		
Frequency [MHz]	50 - 540	120 - 540	85 - 275	10 - 400	180 - 478		
System Power [mW]	8.268 - 171.0	60 - 279	29.83 - 152.75	0.87 - 74.9	73.8 - 211.1		
Memory Size	200kB SRAM	272kB SRAM	192kB SRAM	512kB SRAM	3Mb eDRAM + 64kB SRAM		
On-Chip BW[1]	414.72	518.4	1126.4	819.2	2202.6		
BW Management	\	\	Static Model-aware	\	Dynamic BW-aware		
Frame Optimizations	Pixel-level (Single View)	Pixel-level (Single View)	\	\	Frame-level & Pixel-level (Multi-View)		
MAC Size	288kb + 64kb DCIM	432 PE	1024kb DCIM	64kb DCIM	864kb DCIM		
CIM Macro Efficiency [TFLOPS/W]	80.3 (FP16)	\	16.8 - 83.5 (INT16)	17.2 - 91.3 (FP16)	76.69 - 132.97 (BF16)[2]		
Effective System[3] Efficiency [TFLOPS/W]	40.06 (BF16)	1.57 - 48.95	10.7 - 53.5 (INT16)	2.83 - 15.0 (FP16)	60.81	54.91	49.74
Macro FoM[4]	65.1	\	4.09	30.74	83.63		
System FoM[5]	3.69	13.2	35.1	1.54	50.44		

1. Based on maximum reported frequency. 2. Measured @180MHz, 0.6V.
3. Considering CIM utilization based on available on-chip memory and bandwidth.
4. Macro FoM = Macro Energy Efficiency × Macro Area Efficiency, evaluated @ 0.75V, area is scaled to 22nm.
5. System FoM = System Energy Efficiency × On-chip Bandwidth, evaluated based on reported peak efficiency point.

Figure 37.6.6: Measurement results and performance comparison table.



Technology	TSMC 22nm		
Chip area	2.91mm×2.82mm		
CIM Macro area	0.149mm×0.168mm×36		
CIM Size	864 kb		
Memory Size	3 Mb eDRAM + 64kB SRAM		
Weight prec.	INT4/8/16, FP8, BF16		
Activation prec.	INT4/8/16, FP8, BF16		
Voltage	0.6 - 0.9 V		
Frequency	180 - 478 MHz		
CIM Power	33.7 - 127.6 mW		
System Power	73.8 - 211.1 mW		
Performance[*1]	6.79 - 9.71 T(FL)OPS (Hybrid-precision)		
CIM Macro[*2] Energy Efficiency	76.69 - 132.97 TFLOPS/W (BF16/BF16)		
System[*2] Energy Efficiency	49.74 - 60.81 TFLOPS/W (Hybrid-Precision)		
Task	Image	3D	Video
Throughput	3.82 iter/s	3.70 iter/s	0.02 iter/s
Energy per DM iteration	29.23 mJ[*3] (Single-Frame)	39.72 mJ[*4] (12-Frames)	9.07 J (25-Frames)

1. Measured @ 0.9V, 478MHz. 2. Measured @ 0.6V, 160MHz.
3. Image Resolution: 64x64 4. Image Resolution: 32x32

Figure 37.6.7: Chip micrograph and specifications.

References:

[1] R. Rombach et al., "High-Resolution Image Synthesis with Latent Diffusion Models," *IEEE CVPR*, pp. 10674-10685, 2022.

[2] X.-Y. Zheng et al., "MVD^2: Efficient Multiview 3D Reconstruction for Multiview Diffusion," *ACM SIGGRAPH*, 2024.

[3] X. Long et al., "Wonder3D: Single Image to 3D using Cross-Domain Diffusion," *IEEE CVPR*, pp. 9970-9980, 2024.

[4] H. Ni et al., "Conditional Image-to-Video Generation with Latent Flow Diffusion Models," *IEEE CVPR*, pp. 18444-18455, 2023.

[5] Z. Deng et al., "MV-Diffusion: Motion-aware Video Diffusion Model," *ACM Multimedia*, pp. 7255-7263, 2023.

[6] H. Jeong et al., "VMC: Video Motion Customization using Temporal Attention Adaption for Text-to-Video Diffusion Models," *IEEE CVPR*, pp. 9212-9221, 2024.

[7] F. Tu et al., "A 28nm 15.59μJ/Token Full-Digital Bitline-Transpose CIM-Based Sparse Transformer Accelerator with Pipeline/Parallel Reconfigurable Modes," *ISSCC*, pp. 466-468, 2022.

[8] F. Tu et al., "MulTCIM: A 28nm 2.24μJ/Token Attention-Token-Bit Hybrid Sparse Digital CIM-Based Accelerator for Multimodal Transformers," *ISSCC*, pp. 248-250, 2023.

[9] S. Liu et al., "A 28nm 53.8TOPS/W 8b Sparse Transformer Accelerator with In-Memory Butterfly Zero Skipper for Unstructured-Pruned NN and CIM-Based Local-Attention-Reusable Engine," *ISSCC*, pp. 250-252, 2023.

[10] R. Guo et al., "A 28nm 74.34TFLOPS/W BF16 Heterogenous CIM-Based Accelerator Exploiting Denoising-Similarity for Diffusion Models," *ISSCC*, pp. 362-364, 2024.

[11] Y. Qin et al., "A 52.01 TFLOPS/W Diffusion Model Processor with Inter-Time-Step Convolution-Attention-Redundancy Elimination and Bipolar Floating-Point Multiplication," *IEEE VLSI Technology and Circuits*, 2024.

[12] J. Yue et al., "A 28nm 16.9-300TOPS/W Computing-in-Memory Processor Supporting Floating-Point NN Inference/Training with Intensive-CIM Sparse-Digital Architecture," *ISSCC*, pp. 252-254, 2023.