An eDRAM Digital In-Memory Neural Network Accelerator for High-Throughput and Extended Data Retention Time

Inhwan Lee[1], Jehun Lee[2], Jaeyong Jang[2], and Jae-Joon Kim[2†]
[1]Pohang University of Science and Technology, Pohang, Korea [2]Seoul National University, Seoul, Korea
[1]inhwan.lee@postech.ac.kr
[2]{jehun.lee, jaeyongjang, kimjaejoon}@snu.ac.kr
[†]Corresponding Author

***Abstract*—Computing-in-Memory (CIM) optimizes multiply-and-accumulate (MAC) operations for energy-efficient acceleration of neural network models. While SRAM has been a popular choice for CIM designs due to its compatibility with logic processes, its large cell size restricts storage capacity for neural network parameters. Consequently, gain-cell eDRAM, featuring memory cells with only 2-4 transistors, has emerged as an alternative for CIM cells. While digital CIM (DCIM) structure has been actively adopted in SRAM-based CIMs for better accuracy and scalability than analog CIMs (ACIM), previous eDRAM-based CIMs still employed ACIM structure since the eDRAM CIM cells were not able to perform a complete digital logic operation. In this paper, we propose an eDRAM bit cell for more efficient DCIM operations using only 4 transistors. The proposed eDRAM DCIM structure also maintains consistent and accurate output values over time, improving retention times compared to previous eDRAM ACIM designs. We validate our approach by fabricating an eDRAM DCIM macro chip and conducting hardware validation experiments, measuring retention time and neural network accuracy. Experimental results show that the proposed eDRAM DCIM achieves 3× longer retention time than state-of-the-art eDRAM ACIM designs, along with higher throughput without accuracy loss.**

I. Introduction

Computing-in-memory (CIM) enhances the efficiency of neural network computing by allowing simultaneous multiply-and-accumulate (MAC) operations directly within memory cells. Among the various memory technologies available for CIM applications, SRAM cells are commonly used due to their compatibility with CMOS logic processes [1]–[3]. However, the relatively large size of SRAM-based CIM cells limits memory capacity as model complexity increases. In contrast, gain-cell eDRAM offers greater memory capacity, using only 2-4 transistors per cell [4]–[7], while maintaining compatibility with logic processes. Meanwhile, digital CIMs (DCIM) have gained traction over traditional analog CIMs (ACIM) because DCIMs, which rely on digital adder trees, avoid the accuracy losses caused by the analog-to-digital converters (ADCs) required in ACIMs [8]–[10]. Despite the growing preference for DCIMs, most existing eDRAM CIM designs are still ACIM-based. This is because prior eDRAM CIM cell architectures, optimized for current-based or capacitive-dividing methods, cannot easily accommodate the digital logic operations required for DCIMs.

To bridge this gap, we propose a gain-cell eDRAM architecture tailored for DCIM operations that can execute full digital logic using only four transistors while maintaining a compact cell size. The proposed design enables high-speed operations without timing issues, as the multiplication results performed within the cells can be efficiently accumulated through the adder tree when inputs are applied. Moreover, even with dynamic storage data, the results from cell multiplications are fed through the adder tree, ensuring accurate MAC values over time. This significantly reduces the need for refresh operations due to the improved memory retention time. Finally, our architecture produces full-range digital MAC outputs via the adder tree, eliminating potential accuracy loss.

Our key contributions can be summarized as follows:

- We introduce an eDRAM bit cell optimized for DCIM structures, capable of performing precise digital logic gate operations for MAC without any accuracy degradation.
- We show that our eDRAM DCIM structure maintains consistent and accurate MAC values over an extended period, thanks to enhanced retention time, surpassing the performance of previous eDRAM ACIM designs.
- We fabricate an eDRAM DCIM chip and evaluate its neural network accuracy and retention time, demonstrating improved throughput and retention, with no loss in accuracy compared to previous eDRAM CIM designs.

II. Review of previous eDRAM cell designs for analog computing-in-memory

In this section, we review previously reported electrical current- and capacitor-based eDRAM ACIM designs focusing on retention time and operating speed. Additionally, we describe why previous eDRAM ACIM designs are not suitable for actual hardware implementation.

A. Overview of eDRAM analog CIM cells operation

Fig. 1(a) and (b) illustrate basic gain-cell eDRAM structures utilizing current-based eDRAM cells [4], [5], [11]. During inference operation, the read transistor conditionally discharges the pre-charged CBL voltage based on the input and weight multiplication result, yielding the final MAC value. However, the impact of dynamic storage node voltage on the CBL voltage makes the analog MAC value highly sensitive to

978-3-9826741-0-0/DATE25/© 2025 EDAA

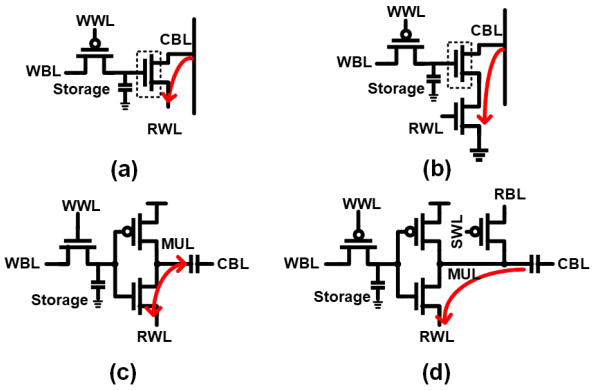


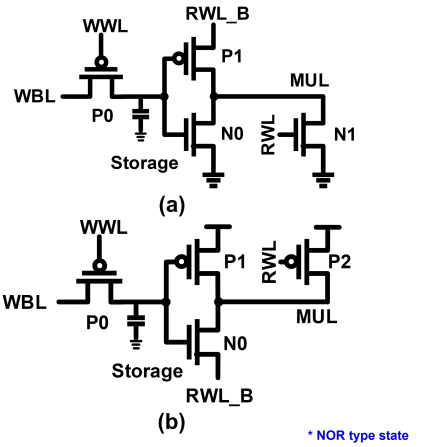
Fig. 1. The cell structures of previous analog eDRAM CIM cells. (a) Current-based 2T1C [11], (b) Current-based 3T1C [5], (c) Capacitor-based 3T2C [6], (d) Capacitor-based 4T2C [7]

data loss at the storage node, necessitating frequent refresh operations for CIMs. Moreover, timing challenges with control signals hamper operation speed.

Fig. 1(c) and (d) depict capacitor-based 3T2C [6] and 4T2C [7] eDRAM cells. A metal-oxide-metal capacitor (MOMCAP) is added to the MUL node of each cell to enhance capacitive coupling with the computing bit line (CBL). The operation involves pre-charged CBL, with inputs and weights generating analog voltage values. The NAND-like buffer structure ensures noise robustness. However, in the 3T2C structure, NMOS pull-up causes threshold voltage loss, incurring a shorter retention time than the 4T2C. In both 3T2C and 4T2C cells, stacking two MOMCAPs in a small eDRAM cell reduces the CBL voltage range, thereby leading to smaller ADC sensing margins and larger accuracy loss.

B. Hardware implementation issues in eDRAM ACIM

EDRAM ACIM structures have two main drawbacks that make hardware implementation challenging. First, they rely on ADCs, which are prone to variations. As a result, ACIMs are known to experience accuracy loss. Second, the dynamic nature of the storage nodes causes the analog MAC values to drift over time, requiring frequent cell refresh operations. These issues complicate hardware design and stability. As a result, there has been growing interest in adder tree-based DCIM structures, which use digital MAC operations to achieve full-range MAC values without accuracy loss. However, this approach requires memory cells capable of performing digital logic operations, which traditional eDRAM ACIM cells, such as 2T1C and 3T1C, cannot support due to their NMOS-only storage nodes. Even 3T2C and 4T2C cells, which include both NMOS and PMOS storage nodes, face challenges with capacitors at the output node that hinder digital MAC operations. In summary, while ACIM benefits from smaller gain-cell eDRAM sizes, its time-varying analog MAC values, ADC sensitivity, and the complexity of hardware implementation highlight the need for new eDRAM DCIM structures capable of accurate digital operations and greater resilience to variations.



* NOR type state

	Write	Read	Hold
WWL	0	1	1
WBL	0 or 1	0	0
RWL	0 (1)	1 (0)	0 (1)
MUL	1 (0)	0 or 1	1 (0)

(c)

Fig. 2. Bit cell structures for the proposed eDRAM CIM: (a) NOR type and (b) NAND type, (c) Bias conditions for the memory operation

III. Proposed eDRAM Cell for DCIM Operation

In this section, we introduce an eDRAM CIM cell specifically designed for DCIM applications, capable of performing entirely digital logic operations. Then, we explain how the proposed eDRAM DCIM structure improves retention time, operation speed, and accuracy of operations compared to previous eDRAM ACIM structures.

A. Cell structure

To provide full digital logic operations, we propose eDRAM cell structures with NOR and NAND types. Fig. 2(a) and (b) depict the proposed eDRAM cells performing NOR and NAND operations, respectively. Both types consist of 4 transistors, with the difference lying in how the multiplication (MUL) node is connected either with NMOS or PMOS transistors in series. To minimize subthreshold leakage current, a PMOS access transistor (P0) is utilized for data writing. Additionally, the PMOS transistor (P1) and NMOS transistor (N0) are connected in series, with the data written through P0 stored at the gate capacitance of these two transistors. Lastly, the NMOS transistor (N1) is connected to the output MUL node for the NOR structure, and the PMOS transistor (P2) is connected to the output MUL node for the NAND structure. Moreover, to reduce cell area overhead while increasing retention time, we stack MOMCAPs on the storage nodes of the proposed eDRAM cells.

B. General memory operations

Since the target weight parameters need to be programmed to CIM cells, a CIM cell must also perform conventional memory operations. The proposed eDRAM CIM cells can perfectly execute typical memory operations, and the bias conditions for memory operations are described in Fig. 2 (c).

- *Memory write*: Both NOR and NAND types perform write operations under the same conditions. Firstly, the write access transistor (P1) is turned on, and the desired data is applied to the storage node through the write bit line (WBL). To prevent threshold voltage loss, a voltage lower than VSS is applied to the write word line (WWL).

- *Memory read*: Unlike conventional memory cells, the proposed eDRAM DCIM cells can read data without using sense amplifiers. For simplicity, we explain the read operations for the NAND type, with the NOR type described in parentheses. In the NAND (NOR) type, applying VDD (VSS) to the read word line (RWL) turns off P2 (N1). If the stored data is '1', P1 is turned off and N0 is turned on, pulling the MUL node down. Conversely, if the stored data is '0', P1 is turned on and N0 is turned off, pulling the MUL node up. Data '0' and '1' can be read out through the adder tree connected to the MUL node. Simultaneously, Other memory cells connected to the adder tree have VSS (VDD) applied to RWL, causing P2 (N1) to turn on regardless of the stored data. Consequently, MUL nodes of other memory cells are pulled up (down) regardless of the stored data, ensuring there is no data conflict between cells.

- *Memory hold*: To hold data, all transistors associated with the memory cell must be turned off. Therefore, VDD and VSS are applied to WWL and WBL, while VDD and VSS are applied to RWLs of the NOR and NAND types, respectively.

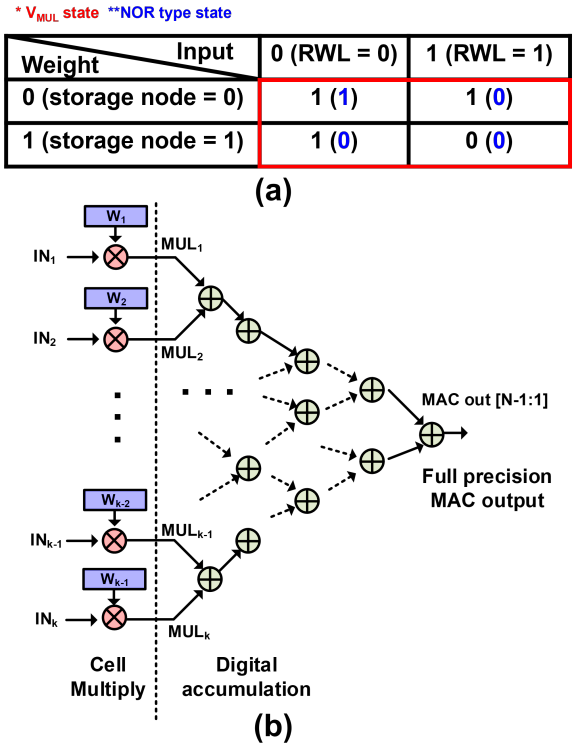


Fig. 3. (a) Truth table for CIM operation, (b) Overall DCIM structure for a single column

C. Digital CIM operations

Our goal is to accelerate neural network computing using CIM operations by mapping inputs and weights to the RWL and storage nodes, respectively, as illustrated in Fig. 3(a). We focus on explaining the operations of a NAND-type eDRAM DCIM cell (with NOR-type operations noted in parentheses). When the input is '0' ('1'), VSS (VDD) is applied to RWL, turning on P2 (N1) and keeping the MUL node at VDD (VSS) regardless of the weight value. If the input is '1' ('0') and the weight is '0' ('1'), VDD (VSS) is applied to RWL, turning off P2 (N1) but turning on P1 (N0), resulting in the MUL node being at VDD (VSS). When both the input and weight are '1' ('0'), P2 (N1) is turned off, and VSS (VDD) is applied to RWL_B, turning on N0 (P1) and causing the MUL node to become VSS (VDD). Consequently, the MUL node discharges (charges) when both the input and weight are '1' ('0') and remains charged (discharged) otherwise, effectively performing a digital NAND (NOR) operation for multiplication. In DCIM operation, multiplication results from each cell are obtained through an adder tree to produce the final MAC result after digital summation. Unlike ACIM operations, DCIM does not require internal timing control signals to develop the MAC value. Multiplication is directly performed with the stored weights when the input is applied, enabling high-speed operation. As shown in Fig. 3(b), within a single column, the multiplication results from each cell are not combined on a common bit line to form an analog MAC value, as in ACIM. Instead, the output nodes from each cell are fed into an adder for digital summation, resulting in the final output value in digital form. The final MAC value is simply expressed by eqs (1) and (2).

$$MAC\ value\ for\ NAND\ type = \sum I_i W_i \tag{1}$$

$$MAC\ value\ for\ NOR\ type = \sum Ib_i Wb_i \tag{2}$$

where I_i,Ib_i, W_i, and Wb_i are input, inverted input, weight, and inverted weight for the cell at the k-th row, respectively

D. Retention time analysis

The proposed eDRAM cells perform accurate digital logic gate operations with exceptional noise resistance and stability against variations. This is achieved by generating precise digital MAC values through an adder tree, avoiding the instability of analog values. Although dynamic storage nodes require periodic refreshes due to leakage, the proposed DCIM operations drastically increase the retention time, thereby reducing the refresh burden. To explain the retention time characteristics, the worst-case condition for the NAND-type cell is depicted in Fig. 4, illustrating their DCIM operations at the logic gate level in the first adder tree stage. Without loss of generality, we examine the worst-case scenario for the retention time in the NAND-type cell only. A similar analysis can also be done for NOR-type cells. In this setup, we assume that only node A of the full adder (FA) is active (inputs B and Cin are set to '0'), allowing the logical result from the eDRAM cell to directly become the SUM output of the FA. To develop the worst-case condition for data retention, the NAND-type cell is first set with input '0' and weight '1' for a long time. In this case, the cell slowly starts to lose charge via the subthreshold leakage

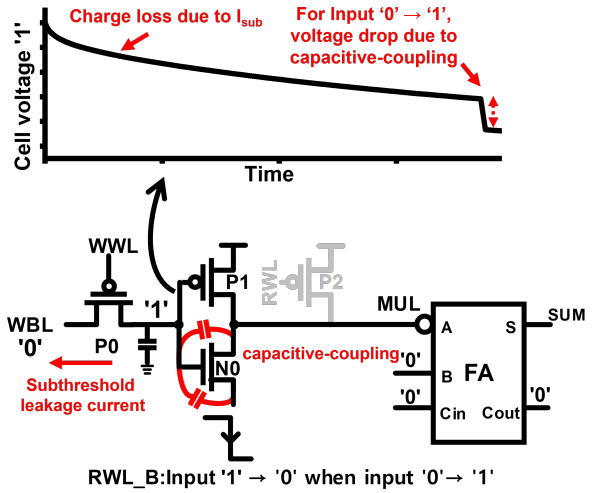


Fig. 4. The worst-case condition for data retention in the proposed eDRAM cell-based DCIM path consisting of a cell and the first stage adder tree.

current through the access transistor P0. When input becomes '1', the RWL_B value changes from '1' to '0', which causes a further voltage drop at the storage node due to capacitive coupling between the source/drain and the gate node of the N0 transistor. If this combined effect of leakage current and capacitive coupling makes the storage node voltage too low, an error at the output node may occur. Note that output data becomes correct regardless of weight data corruption when input '0' is applied because P2 turns on in such a case, keeping the MUL node at '1' regardless of weight data value.

Fig. 5 illustrates the time-dependent change of weight, MUL, and the SUM output of the FA in the proposed NOR and NAND cells under the worst-case retention conditions. The results are from a 1K Monte-Carlo simulation. In Fig. 5(a), the weights of the NOR cell ('0') and the NAND cell ('1') degrade over time due to leakage currents and the capacitive coupling. Despite the cell data loss, the MUL node voltage in the proposed eDRAM cell experiences slower change with time thanks to the regenerative characteristics of digital logic gates, as shown in Fig. 5(b). NAND cells exhibit longer retention times because NMOS transistors have higher mobility than PMOS transistors, leading to a lower trip voltage in the buffer structure for the same cell size. Consequently, NAND cells storing a weight of '1' are more resilient to noise compared to NOR cells storing a weight of '0'. Unlike previous ACIM structures, these MUL values are not connected to a single bit line but are connected to digital adders, where they are summed together. The MUL values are buffered and transferred through the FA. As a result, the final SUM output of the FA does not experience the voltage change until a significantly longer time than that of a single memory cell and the corresponding MUL node, as shown in Fig. 5(c). In summary, the NOR-type cell shows a retention time of 120 μs, and the NAND-type cell shows that of 332 μs. Thus, the proposed eDRAM DCIM operation not only ensures reliable and accurate performance against variations but also extends retention time significantly.

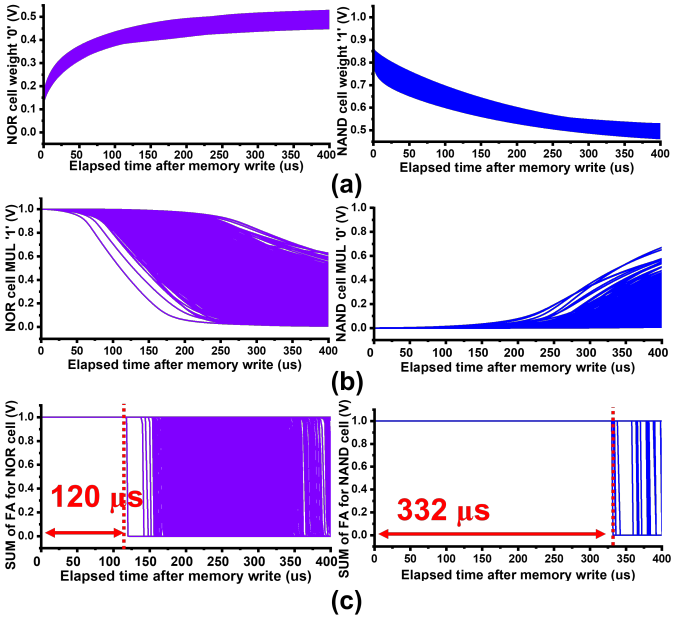


Fig. 5. Retention time analysis of the proposed NOR-type (Left) and NAND-type (Right) DCIMs: (a) cell storage node voltage changes over time (b) MUL node voltage changes over time (c) changes of the SUM output voltage of FA over time

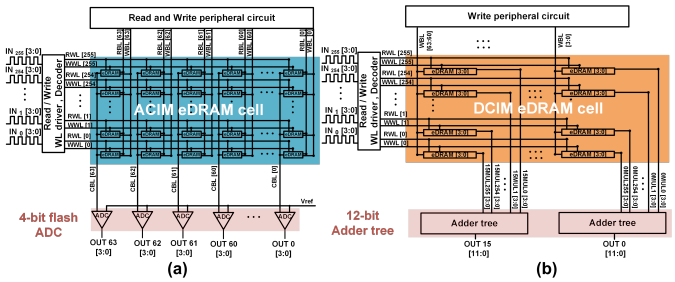


Fig. 6. Overall architecture for (a) ACIM and (b) DCIM structure

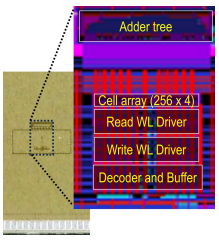
IV. Evaluation

We evaluate previous ACIM designs and the proposed eDRAM DCIM architecture, focusing on retention time, throughput, and energy efficiency. We fabricated the proposed eDRAM DCIM array and measured retention time and operating frequency, while previous ACIM designs were validated through simulations only. For a fair comparison, we reproduced previous ACIM structures in a 28nm CMOS technology. We chose the NAND-type cell over the NOR-type cell for the eDRAM DCIM due to its longer retention time.

The array size is 256x64 (Fig. 6). 4-bit inputs were applied bit-serially, and 4-bit weights were stored in multiple columns in a bit-parallel fashion. In the ACIM structure, each column uses a 4-bit flash ADC (Fig. 6(a)), while the DCIM structure allocates a 12-bit adder tree for every 4 columns (Fig. 6(b)). Both designs use 1 V supply voltage except the adder tree in the DCIM which uses 0.8 V. The VGG-9 model was tested on the CIFAR-10 dataset to assess neural network accuracy and hardware performance.

A. Retention time

The retention time of eDRAM is a key factor in determining how many CIM operations it can perform without a refresh op-



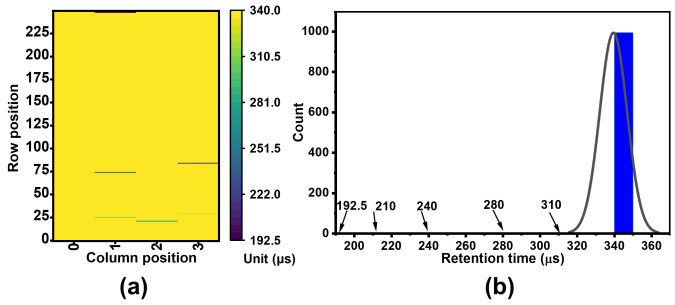
(a)

Technology	28 nm CMOS
Supply voltage (V)	0.8 ~ 1.0
Frequency (MHz)	47 ~ 70
Input / Weight / Output	4/4/ 12
Array size	256x4 (1kb)

(b)

Fig. 7. (a) eDRAM DCIM array chip photograph and layout, (b) Summary table of eDRAM DCIM chip

eration. To measure the retention time of the proposed eDRAM DCIM architecture, we fabricated a dedicated eDRAM chip as shown in Fig. 7. Fig. 8(a) shows a heatmap of the measured retention time from the eDRAM DCIM chip, corresponding to the locations of the memory cells. Fig. 8(b) shows the distribution of the retention time measured from multiple cells. Due to the proposed eDRAM DCIM operation, the retention time of most cells exceeds 300 µs, with the shortest measured retention time being 192 µs.



(a) (b)

Fig. 8. Retention time measurement result from the fabricated chip: (a) Heatmap according to the locations of memory cells, (b) distribution of retention time

Next, we perform neural network simulations based on both simulated and measured retention times, comparing the time-dependent inference accuracy of the proposed DCIM with previous ACIM designs, considering the presence or absence of variations. The accuracy simulation results without considering cell variations are shown in Fig. 9(a). A common observation for all ACIM structures is that due to partial sum quantization in ADC, accuracy is lower than the software baseline even at t=0. In the case of 2T1C and 3T1C structures, the accuracy sharply decreases within 2-3 μs since the weight value change with time directly affects the output nodes. 3T2C and 4T2C, on the other hand, have buffer structures capable of absorbing a certain amount of noise within the cell itself, thereby maintaining accuracy longer than 2T1C and 3T1C. However, 3T2C, performing pull-up operations with NMOS transistors, continuously experiences threshold voltage loss over time, resulting in the shorter retention time than 4T2C. Unlike other ACIM cells, 4T2C maintains accuracy up to 100 μs, which is 3-50 times longer. However, thanks to the noise recovery capability of the digital-style design, the accuracy of the proposed structure remains identical to the baseline without any accuracy loss for a long time, being maintained up to 340 μs, 3.4 times longer than the 4T2C design.

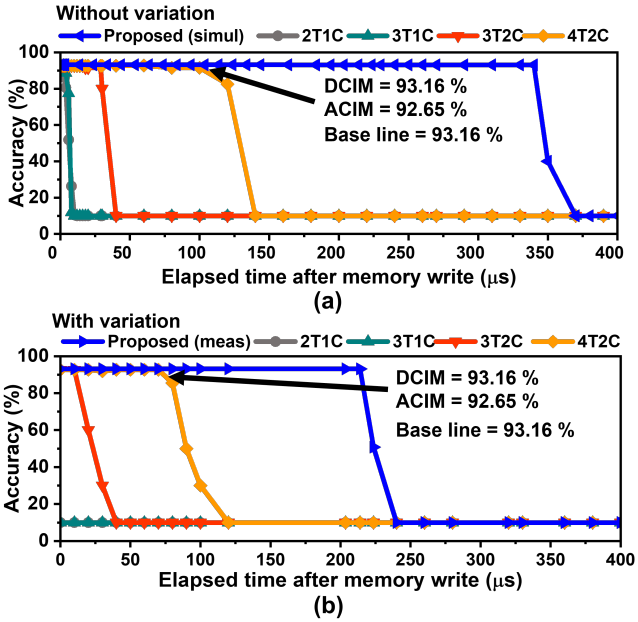


Fig. 9. Test accuracy measurement over elapsed time after memory write for CIFAR-10 dataset (a) without variation and (b) with variation

The superiority of the proposed design in terms of retention time becomes more pronounced when cell variations are considered. Fig. 9(b) shows the results of accuracy simulations over time, considering variation information from the heatmap shown in Fig. 8(a). For 2T1C and 3T1C, variations in memory cells directly affect the output values, preventing them from maintaining accuracy even right after memory write operations. While the 3T2C and 4T2C are more resilient to variations than 2T1C and 3T1C, they still suffer significantly higher accuracy loss over time compared to the case without cell variation. In contrast, the proposed structure maintains high accuracy over a much longer time even with the extra variations and noise, achieving a retention time of 210 μs, which is three times longer than the 4T2C ACIM.

B. Runtime Comparison

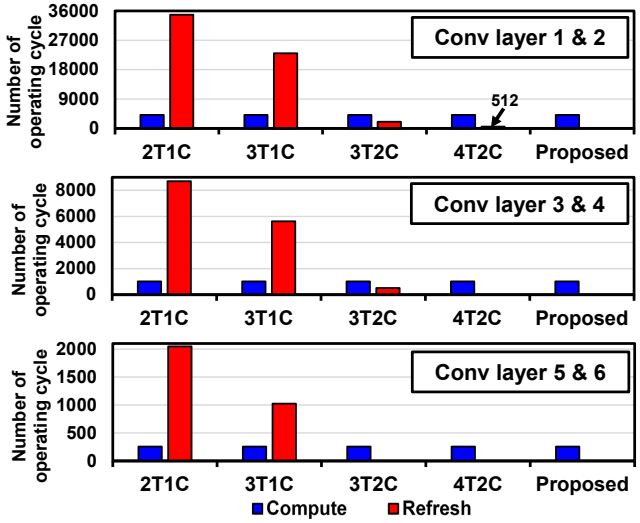


Fig. 10. Number of operation cycles for each convolution layer of the VGG-9 model for various CIM arrays

Neural network operations in the eDRAM array heavily depend on both retention time and operating speed. To assess how many operations can be performed within the retention time, we conducted cycle simulations for neural network operations on eDRAM CIM arrays, considering both retention time and operation frequency. Retention time was based on how long the eDRAM array maintained accuracy without variation. Operation frequencies were determined through simulations and chip measurements: 30 MHz for the ACIM structure at 1 V and 47 MHz for the DCIM structure at 0.8 V. The simulations center on convolution layer operations. Inputs are applied after weights are written into the CIM array, with rows representing input channels and columns representing output channels. Therefore, the number of operation cycles depends on the spatial size of the input image and precision, as shown in Fig. 10 for the VGG-9 model. Computing cycles remain constant across different designs due to identical inputs. And, they decrease in deeper layers as the input size reduces. The 2T1C and 3T1C designs have short retention times of 2 and 3 μs, causing their refresh cycles to exceed computing cycles significantly. In layers 1 to 4, refresh consumes 8.5 and 5.6 times longer time than computing operations for the 2T1C and 3T1C designs, respectively, and in layers 5 and 6, refresh takes 8 and 4 times longer time than computing for the 2T1C and 3T1C designs, respectively. The 3T2C and 4T2C have improved retention times of 30 and 100 μs but still require some amount of refreshes during computing operations. For 3T2C, refresh cycles are half the computing cycles for layers 1 to 4, with none needed for layers 5 and 6. For 4T2C, refresh cycles are a quarter of the computing cycles for layers 1 and 2, while no refresh is needed for layers 3 to 6. However, the proposed structure eliminates the need for refresh cycles across the entire convolution layer, enabling higher operational frequencies and much longer retention times. So, the proposed design allows continuous operations without interruptions for refresh operations.

C. Throughput and energy efficiency

Fig. 11 shows the throughput and energy efficiency of different eDRAM CIM designs for each convolution layer of the VGG9 model. In Fig. 11(a), 2T1C and 3T1C exhibit very low throughput in all convolution layers due to frequent refresh operations from short retention times and low operation frequencies. The 3T2C and 4T2C designs achieve better throughput with fewer refresh cycles. However, the proposed eDRAM DCIM shows a higher throughput than all previous eDRAM ACIMs thanks to its longer retention time and higher operation frequency. Specifically, the proposed structure achieves 15–16×, 8–11×, 1.6–2.5×, and 1.6–1.9× higher throughput than the 2T1C, 3T1C, 3T2C, and 4T2C designs, respectively. As shown in Fig. 11(b), the proposed eDRAM structure, despite its improved retention time and high throughput, exhibits slightly lower energy efficiency compared to 3T2C and 4T2C, while outperforming 2T1C and 3T1C ACIMs by 1.7× and 1.45×, respectively. This is partly due to the fact that the proposed DCIM operates at a higher frequency, causing power consumption to increase proportionally

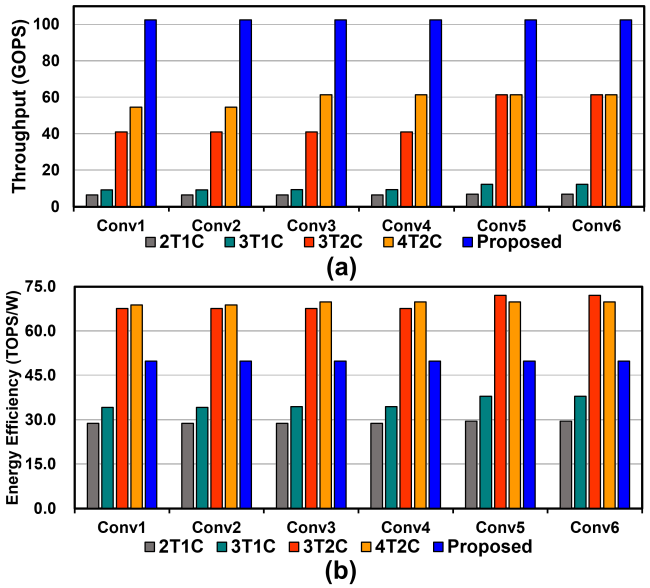


Fig. 11. Comparison of the proposed DCIM over previous ACIMs in terms of (a) throughput and (b) energy efficiency for each convolution layer of the VGG-9 model

with the frequency. Although the proposed structure has about 1.3× lower energy efficiency than 3T2C and 4T2C, its higher network accuracy due to accurate computation and higher throughput justify the slightly reduced energy efficiency.

V. Conclusion

We propose an eDRAM bit cell for the DCIM neural network accelerator that ensures no loss of accuracy, making the first implementation capable of fully executing eDRAM DCIM operations within a single cell. We validate fully digital operations that are resilient to variations, offer increased retention time, and maintain accurate MAC operations over an extended time. We fabricated an eDRAM DCIM chip, and measurement results demonstrate significantly improved robustness compared to previous designs, achieving 3× longer retention time and 1.9× higher throughput without any accuracy loss. These improvements, along with the increased capacity for storing more model parameters, make our design well-suited for processing more complex neural network operations and highly applicable for real-world CIM hardware implementations.

Acknowledgment

This work was supported in part by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2023-00208606, NeuroHub+: Scheduler and Simulator for General In-Memory Neural Network Accelerators, No. 2022-0-00266, Development of Ultra-Low Power Low-Bit Precision Mixed-mode SRAM PIM, IITP-2023-RS-2023-00256081: artificial intelligence semiconductor support program to nurture the best talents), BK21 FOUR program at Seoul National University, Samsung Research Funding Center under Project SRFC-TC1603-53, and ISRC at Seoul National University. The EDA tool was supported by the IC Design Education Center(IDEC). (Corresponding Author: Jae-Joon Kim).

REFERENCES

[1] Z. Jiang *et al.*, "C3sram: An in-memory-computing sram macro based on robust capacitive coupling computing mechanism," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 7, pp. 1888–1897, 2020.

[2] S. Yin *et al.*, "Pimca: A 3.4-mb programmable in-memory computing accelerator in 28nm for on-chip dnn inference," in *2021 Symposium on VLSI Circuits*, 2021, pp. 1–2.

[3] J. Lee *et al.*, "Fully row/column-parallel in-memory computing sram macro employing capacitor-based mixed-signal computation with 5-b inputs," in *2021 Symposium on VLSI Circuits*, 2021, pp. 1–2.

[4] T. Yoo *et al.*, "A logic compatible 4t dual embedded dram array for in-memory computation of deep neural networks," in *2019 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, 2019, pp. 1–6.

[5] Z. Chen *et al.*, "15.3 a 65nm 3t dynamic analog ram-based computing-in-memory macro and cnn accelerator with retention enhancement, adaptive analog sparsity and 44tops/w system energy efficiency," in *2021 IEEE International Solid-State Circuits Conference (ISSCC)*, vol. 64, 2021, pp. 240–242.

[6] S. Kim *et al.*, "16.5 dynaplasia: An edram in-memory-computing-based reconfigurable spatial accelerator with triple-mode cell for dynamic resource switching," in *2023 IEEE International Solid-State Circuits Conference (ISSCC)*, 2023, pp. 256–258.

[7] I. Lee *et al.*, "In-memory neural network accelerator based on edram cell with enhanced retention time," in *2023 60th ACM/IEEE Design Automation Conference (DAC)*, 2023, pp. 1–6.

[8] Y.-D. Chih *et al.*, "16.4 an 89tops/w and 16.3tops/mm2 all-digital sram-based full-precision compute-in memory macro in 22nm for machine-learning edge applications," in *2021 IEEE International Solid-State Circuits Conference (ISSCC)*, vol. 64, 2021, pp. 252–254.

[9] H. Fujiwara *et al.*, "A 5-nm 254-tops/w 221-tops/mm2 fully-digital computing-in-memory macro supporting wide-range dynamic-voltage-frequency scaling and simultaneous mac and write operations," in *2022 IEEE International Solid-State Circuits Conference (ISSCC)*, vol. 65, 2022, pp. 1–3.

[10] H. Mori *et al.*, "A 4nm 6163-tops/w/b 4790-tops/mm2/b sram based digital-computing-in-memory macro supporting bit-width flexibility and simultaneous mac and weight update," in *2023 IEEE International Solid-State Circuits Conference (ISSCC)*, 2023, pp. 132–134.

[11] K. C. Chun *et al.*, "A 667 mhz logic-compatible embedded dram featuring an asymmetric 2t gain cell for high speed on-die caches," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 2, pp. 547–559, 2012.