Embedded Transformer Hetero-CiM: SRAM CiM for 4b Read/Write-MAC Self-attention and MLC ReRAM CiM for 6b Read-MAC Linear&FC Layers

Naoko Misawa, Tao Wang, Chihiro Matsui, and Ken Takeuchi
Department of Electrical Engineering and Information Systems, The University of Tokyo, Tokyo, Japan
E-mail: misawa@co-design.t.u-tokyo.ac.jp

***Abstract*—Heterogeneously integrated SRAM & ReRAM Computation-in-Memories (CiMs) are proposed for Transformer models. The emerging transformer models including LLMs such as ChatGTP are composed of 1) linear & FC layers that only read weights of MAC and 2) self-attention that both reads and writes weights of MAC. To meet these diverse requirements and achieve compact transformer models that can be embedded at the edge, this paper proposes Transformer Hetero-CiM. Proposed Transformer Hetero-CiM is composed of 1) SRAM CiM for 4-bit Read/Write-MAC Self-attention and 2) MLC ReRAM CiM for 6-bit Read-MAC linear & FC Layers. By the optimal mix & match of low energy write and endurance free SRAM CiM and high capacity/low cost MLC ReRAM, the optimal 3D-integration Transformer system of the edge AI is achieved. Proposed Transformer Hetero-CiM reduces the circuit area by 89.1% and 45.3 % compared with transformer models that intensively use SRAM CiMs or ReRAM CiMs, respectively. Furthermore, proposed Transformer Hetero-CiM improves inference accuracy by 1.1% compared with intensive SRAM CiMs and intensive ReRAM CiMs.**

Keywords— Transformer, Computation-in-Memory (CiM), ReRAM, SRAM

I. Introduction

Emerging transformer models such as Large Language models (LLMs) typified ChatGPT and Vision Transformer (ViT) achieve high inference accuracy and parallel computation due to self-attention mechanism [1]. Transformer models are mainly composed of 1) linear & fully-connected (FC) layers and 2) self-attention. In linear & FC layers, multiple-accumulate (MAC) operation is performed using fixed weights. On the other hand, self-attention is the mechanism to obtain the correlations between each input tokens, such as words or image pixels. Even correlations between distantly located tokens can be obtained. Thanks to self-attention, transformer models do not require recurrent layers or convolution layers. In self-attention, the weights of MAC operation are changed in each input. To embed transformer models at edge devices such as mobile phones or IoT devices, Computation-in-Memory (CiM) can be applied to MAC operation to reduce memory size and to accelerate calculation [2, 3]. However, the two different mechanism of linear & FC layers and self-attention in transformer models make CiM adaptation complicated. Especially, applying CiM to self-attention is more considerable than that to linear & FC layers because self-attention, whose weight values of MAC operation change depending on input, require writing weights in CiM. Thus, in this paper, Transformer Hetero-CiM that integrates ReRAM CiMs for Read-MAC operation in linear & FC layers and SRAM CiMs for Read/Write operation in self-attention as shown Fig. 1(a). Fig 1(b) shows 3D-integration of proposed Transformer Hetero-CiM. To solve the problem of a large number of parameters in transformer models, stacking CiMs in 3D enables more compact edge-embedded transformer models.

II. Step-by-Step Transformer Hetero-CiM Design Methodology

Fig. 2 shows step-by-step Transformer Hetero-CiM design methodology. Fig. 2(a) shows training and inference for adapting transformer models to proposed Transformer Hetero-CiM. Fig. 2(b) shows 3 steps of proposed Transformer Hetero-CiM design methodology. In Step 1, the overall configuration where and which memory CiMs are applied in a transformer model is determined. In Step 2, training and inference items such as fine-tuning or *N*-bit quantization aware training (QAT) for self-attention are approximately determined. In Step 3-1, training and inference items are determined for both linear & FC layers and self-attention based on Step 2. In Step 3-2, training and inference items for respective linear & FC layers and self-attention are determined.

III. Step 1: Determination of The Overall Configuration of Transformer Hetero-Cim

Table I shows characteristics of SRAM CiM and ReRAM CiM. The advantage of ReRAM CiM is smaller cell size (10 - 50 F^2) [4, 5] compared with 120 - 200 F^2 for SRAM CiM [6]. On the other hand, SRAM CiM has the advantage of high Set/Reset endurance. Table II summarizes the requirements for CiM from transformer models. Because linear & FC layers require Read MAC operation and have a large number of weight parameters, high capacity/low cost MLC ReRAM CiM is suitable. Because self-attention requires Write/Read MAC operation and has a smaller number of weight parameters, low energy write and endurance free SRAM CiM is suitable.

Fig. 3(a) shows operation in linear & FC layers with ReRAM CiM. As for linear & FC layers, the weighs of trained model are used. Linear layer with W_{QKV} is shown as an example. Input matrix X is duplicated into three copies for the computation of query Q, key K, and value V. Q, K, and V are obtained by multiplying the matrix X by their respective weights W_Q, W_k, and W_V. Fig. 3(b) shows Read-MAC operation of MLC ReRAM CiM. Input matrix X corresponds to word-line (WL) of MLC ReRAM CiM. The weights of trained model after quantization are stored in MLC ReRAM. Output is obtained by bit-line current. As for inference in linear & FC layers, after MLC ReRAM is written to store the trained weights, MLC ReRAM CiMs are required to intensive Read-MAC operation.

Fig. 4(a) shows operation in self-attention. To map to SRAM CiMs, operation in self-attention is divided into three parts. First, matrix product of Q and K^T is calculated by QK-SRAM CiM. Next, Attention weight A is calculated with digital processors by scaling QK and applying Softmax function to the result of QK. Finally, head h is calculated with AV-SRAM CiM by multiplying V to Attention weight A. In QK-SRAM CiM, Q is input and K^T is the weights stored in SRAM (Fig. 4(b)) while in AV-SRAM CiM, Attention weight A is input and V is the weights stored in SRAM (Fig. 4(c)). Because Q, K and V are changed each input matrix X, the weights of SRAM CiMs are written and red out for MAC operation each input. In this work, ViT model called ViT-base model with 12 layers is used. The image classification task is evaluated for proposed Transformer Hetero-CiM.

IV. Analysis on Quantization and Error Toleance in Transfomer Models

To apply ReRAM CiM and SRAM CiM to transformer models, quantization and memory error tolerance need to be considered. Fig. 5 shows quantization and error injection methods. Fig. 5(a) shows initial weight value of MAC operation. Not to include outliers and to increase accuracy, clipping method is useful. After quantization of the initial weight value with/without clipping, the quantized weights (Fig. 5(b)) are stored in ReRAM or SRAM. Device errors are represented as Gaussian errors (Fig. 5(c)) and shift errors (Fig. 5(d)). Normalized step (n.s.) means unit when normalizing weights between -1 and 1. Fig. 6 shows an example of value distribution of (a) Q, (b) K, (c) V, and (d) Attention weight A

in Layer 12 in ViT for a given input. Layer 12 is the closest layer to output in transformer encoder. The distribution of Attention weight *A* is different from *Q*, *K*, and *V* because of scaling and Softmax function. Fig. 7(a) shows structure of ReRAM. Figs. 7(b)-(d) show measured current distributions. Because tail bits increase as Set/Reset cycles increase (Fig. 7(b)) [7], variance becomes larger (Fig. 7(c)). The tendency of write variation is represented by Gaussian errors. Shift errors represent the weight shift due to high endurance or data-retention (Fig. 7(d)) [8]. In this work, shift errors that shift the weights in the positive direction are considered.

V. Step 2: Approximate Determination of Traininsg & Inference Items for SRAM CiM for Self-attention

In Step 2, regarding self-attention, which is the most important function of transformer models for high accuracy and parallel processing, training and inference items are approximately determined. As for self-attention, QK-SRAM CiM calculates the matrix product of *Q* and K^T while AV-SRAM CiM calculates the weighted sum of Attention weight *A* and *V*. Firstly, it is determined which of *Q* and *K* is used as input and weights. Second, as shown in Fig. 2, training items such as fine-tuning or *N*-bit QAT and inference items such as bit precision and with or without clipping are determined. These items are determined by considering quantization and error tolerance of parameters of self-attention. 95% inference accuracy is defined as criterion. Fig. 8 shows inference accuracy when each parameter of self-attention, *Q*, *K*, *V*, and Attention weight *A* is quantized with clipping. To examine the tolerance of each parameter, if *Q* is quantized, other parameters are not quantized. With fine-tuning, acceptable bit precision is 2 bits for both *Q* and *K* (Fig. 8(a)). Acceptable bit precision of *V* is 3 bits while acceptable bit precision of Attention weight *A* is 6 bits. 6-bit QAT improves inference accuracy of *V* and Attention weight *A* although acceptable bit precision of all parameters is same as that of fine-tuning (Fig. 8(b)). Therefore, 6-bit QAT is better than fine-tuning. Fig. 9 shows inference accuracy when quantization from 1- to 8-bit precision of (a) *Q* and *K* for QK-SRAM CiM and (b) Attention weight *A* and *V* for AV-SRAM CiM, respectively. Both *Q* and *K* are tolerant to quantization. AV-SRAM CiM has low quantization tolerance of Attention weight *A*, resulting inference accuracy is lower even if bit precision of *V* is 8 bits. Fig. 10 shows inference accuracy when (a) Gaussian errors and (b) shift errors are injected. To evaluate without being affected by quantization, *N*-bit QAT and clipping are not performed. As for *Q* and *K*, *Q* is the most tolerant to Gaussian errors and *K* is the most tolerant to shift errors. Attention weight *A* is also the worst in terms of both Gaussian and shift errors. From the results of Step 2, *Q* as input and *K* as weights for QK-SRAM CiM, and Attention weight *A* as input and *V* as weights for AV-SRAM CiM are determined.

VI. Step 3: Final Determination of Training & Inference Items for Transformer Hetero-CiM

In Step 2, input and weight of SRAM CiM for self-attention are determined. Next, in Step 3, both ReRAM CiM for linear & FC layer and SRAM CiM for self-attention are considered as a whole system.

A. Step 3-1 Determination Commonly Applicable Training & Inference Items for ReRAM and SRAM CiMs

To consider training and inference items commonly applicable to both ReRAM and SRAM CiMs, linear & FC layers and self-attention are quantized with same bit precision. Fig. 11 shows inference accuracy when weights in linear & FC layers and self-attention are quantized in same bit precision (a) without and (b) with clipping. As shown in Fig. 11(a), acceptable bit precision for most QAT is 6 bits. However, applying clipping in inference improves 5 bits (Fig. 11(b)). In 5-bit precision with clipping in inference, 6-bit QAT has the highest inference accuracy. To summarize the results in Step 3-1, for liner & FC layers and self-attention, which are ReRAM and SRAM CiMs, 6-bit QAT for training and 5-bit precision with clipping for inference are optimal.

B. Step 3-2 Final Optimization of Training & Inference Items for Transformer Hetero-CiM

Finally, based on Step 3-1, training and inference items are optimized for each ReRAM and SRAM CiM to achieve Transformer Hetero-CiM. To make more compact edge-embedded Transformer Hetero-CiM, cell size needs to be considered. Because cell size of SRAM is approximately five times larger than that of ReRAM, total CiM area can be reduced by applying the smaller bit precision of SRAM than ReRAM. According to the result of Step 3-1, 5-bit precision is optimal. Therefore, in inference, 4-bit precision of SRAM and higher bit precision of ReRAM are evaluated to maintain high inference accuracy (Fig. 12(a)). Even if 4-bit precision of SRAM with clipping is used in inference, 6-bit QAT in training and 6-bit precision of ReRAM with clipping in inference achieves 95% inference accuracy. Because 4-bit precision of SRAM is acceptable when 6-bit QAT is used in training, further lower bit precision of SRAM is evaluated by increasing bit precision of ReRAM as shown in Fig. 12(b). If 2-bit precision is used for SRAM in inference, inference accuracy catastrophically degrades regardless bit precision of ReRAM. If 3-bit precision of SRAM is used in inference, inference accuracy is not higher than the accuracy when 4-bit precision of SRAM and 6-bit precision ReRAM are applied. Furthermore, by applying MLC ReRAM embedded transformer models becomes even smaller. Thus, proposed Transformer Hetero-CiM is composed of SRAM CiM for 4-bit Read/Write-MAC Self-attention and MLC ReRAM CiM for 6-bit Read-MAC Linear & FC Layers.

VII. Conclusion

Table III summarizes proposed Transformer Hetero-CiM compared with conventional ViT, intensive SRAM CiMs or SLC ReRAM CiMs. Proposed Transformer Hetero-CiM reduces the circuit area by 89.1% compared with intensive SRAM CiMs and 45.3% compared with intensive SLC ReRAM CiMs. Furthermore, proposed Transformer Hetero-CiM improves by 1.1% inference accuracy rather than intensive SRAM CiMs and SLC ReRAM CiMs. Fig. 13(a) shows design of proposed Transformer Hetero-CiM. Figs. 13(b) and (c) show inference accuracy when (b) Gaussian errors and (c) shift errors are injected into proposed Transformer Hetero-CiM. Because proposed Transformer Hetero-CiM is designed step-by-step and Attention weight *A* is corresponded to input of SRAM CiM, error tolerance of self-attention is improved (Table IV). This Transformer Hetero-CiM achieves the best energy efficiency and MAC performance. First, as for the energy efficiency, proposed Transformer uses the minimum energy solution of SRAM CiM for self-attention and ReRAM CiM for Linear & FC layers. Similarly for the performance (mainly as for the throughput) of CiM, both most energy efficient and fastest CiM combinations of SRAM & ReRAM mix & match is proposed. By incorporating digital processor for the CiM unfriendly functions, cost including memory capacity, MAC performance and energy efficiency is maximized by proposed Transformer Hetero-CiM.

Acknowledgment

This paper is based on results obtained from a project, JPNP23015, subsidized by the New Energy and Industrial Technology Development Organization (NEDO).

References

[1] A. Dosvitsky et al., *ICLR*, 2021, pp. 1-22. [2] R. Yamaguchi et al., *SSDM*, 2023, pp. 495-496. [3] R. Guo et al., *A-SSCC*, 2023, pp. 1-3. [4] Y. -C. Luo et al., *IEEE Transactions on Electron Devices*, vol. 71, no. 1, pp. 916-921, 2024. [5] C. -W. S. Yeh and S. S. Wong, *IEEE J. Solid -State Circuits*, vol. 50, no. 5, pp. 1299-1309, 2015. [6] J. Boukhobza et al., *ACM Trans. Des. Autom. Electron. Syst.*, vol. 23, no. 2, 2017. [7] K. Taoka et al., *SSDM*, 2021, pp. 119-122. [8] Y. Yamaga et. al., *IEEE Symp. VLSI Tech*., 2018, pp. 109-110.

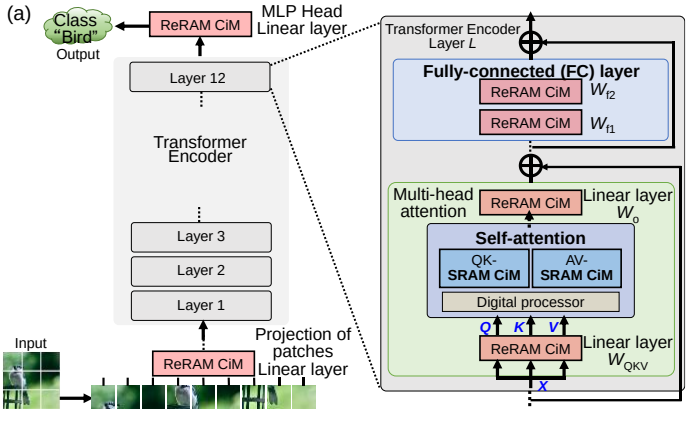


Fig. 1 Proposed embedded Transformer Hetero-Computation-in-Memory (CiM). (a) The details of proposed Hetero-CiM for Vision Transformer (ViT) that integrates SRAM CiM for 4-bit Read/Write-MAC self-attention and MLC ReRAM CiM for 6-bit Read-MAC linear & FC layers. (b) 3D-integration of proposed Hetero-CiM.

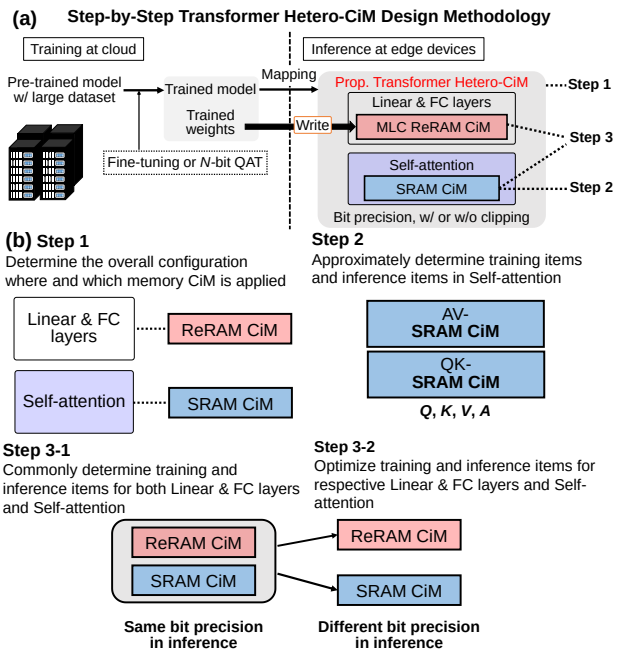


Fig. 2 Step-by-step Transformer Hetero-CiM Design Methodology. (a) Training and inference of proposed Transformer Hetero-CiM. (b) Proposed Transformer Hetero-CiM design methodology at each step.

Table I Characteristics of ReRAM and SRAM CiMs

ViT	Linear & FC layers	Self-attention
CiM type	ReRAM CiM	SRAM CiM
Cell size (F^2)	10-50 (30)	120-200 (150)
Set/ Reset Endurance	10^8-10^{11} cycles	Endurance free

The numbers of parentheses for calculations in Table III

Table II Requirements from ViT to CiM

ViT	Inference	Weights for MAC operation	Requirements to CiM
Linear & FC layers	Read-MAC operation	85.5M	-High read cycles -High capacity -Long data-retention
Self-attention	Read/Write-MAC operation	3.61M	-High Set/Reset endurance -Low energy write

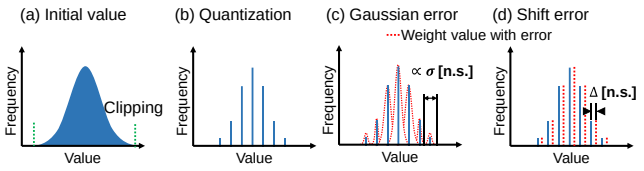


Fig. 5 Values in transformer models corresponding to weights of CiM. (a) Initial value w/ or w/o clipping. (b) Quantized value. Quantized value with (c) Gaussian error injected and (d) shift error injected.

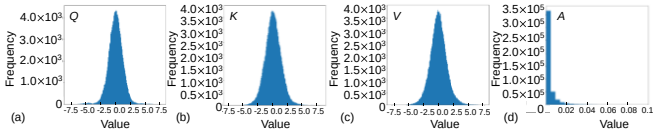
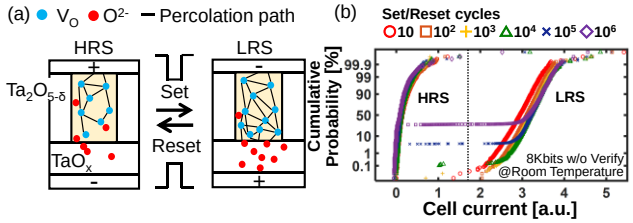


Fig. 6 Initial value distribution of Layer 12 in ViT with fine-tuning. (a) *Q*, (b) *K*, (c) *V*, and (d) Attention weight *A*.



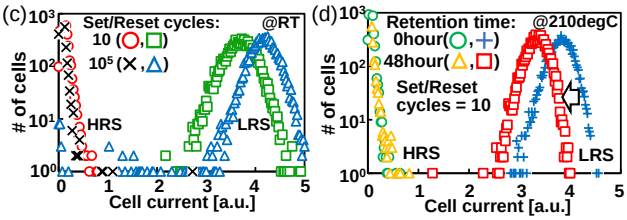


Fig. 7 (a) Structure of ReRAM. Measured ReRAM cell current distributions with (b) 10 to 10^6 Set/Reset cycles, (c) 10 and 10^5 Set/Reset cycles, and (d) during data-retention.

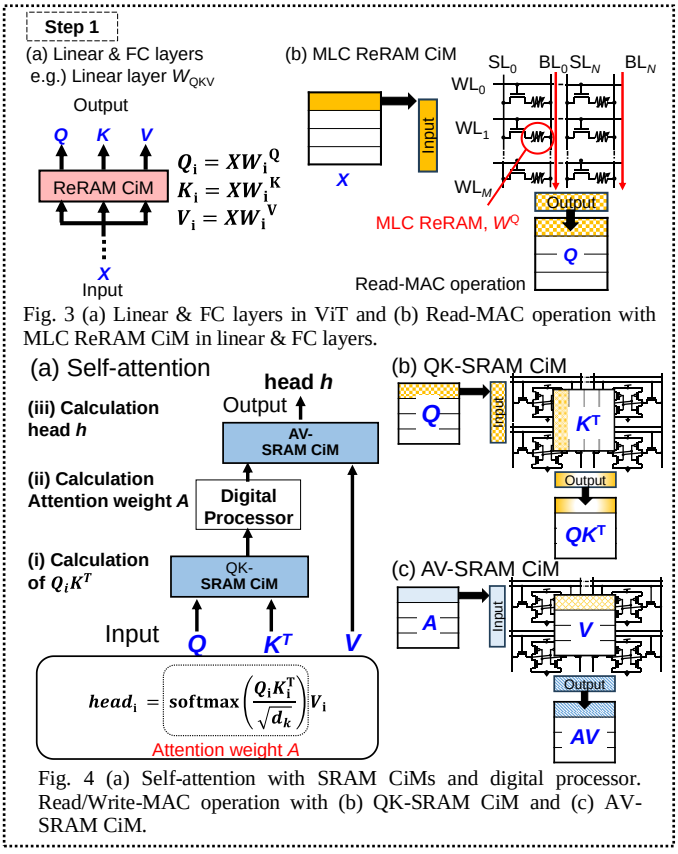


Fig. 3 (a) Linear & FC layers in ViT and (b) Read-MAC operation with MLC ReRAM CiM in linear & FC layers.

Fig. 4 (a) Self-attention with SRAM CiMs and digital processor. Read/Write-MAC operation with (b) QK-SRAM CiM and (c) AV-SRAM CiM.

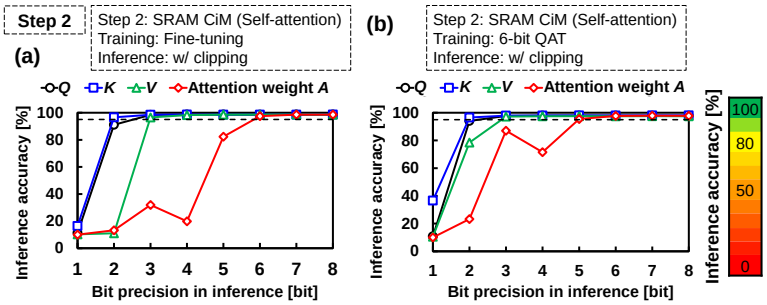


Fig. 8 Inference accuracy when each parameters of self-attention for SRAM CiM is quantized in the case of (a) fine-tuning and (b) 6-bit QAT. If Q is quantized, the other parameters, K, V, and Attention weight A are 32 bits.

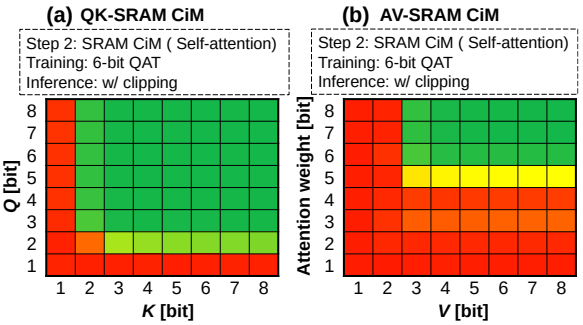


Fig. 9 Inference accuracy with different bit precision of (a) Q and K in QK-SRAM CiM and (b) Attention weight A and V in AV-SRAM CiM.

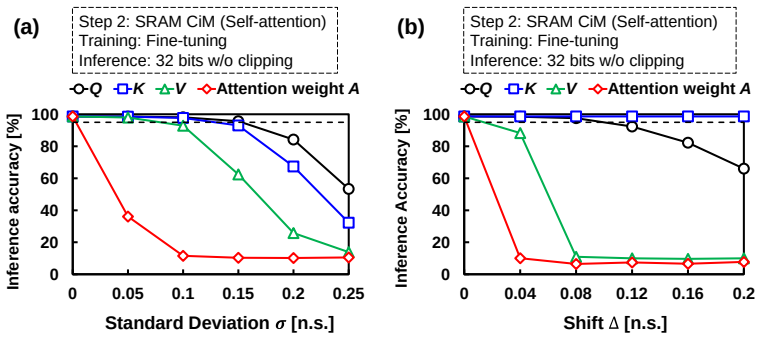


Fig. 10 Inference accuracy when (a) Gaussian errors and (b) shift errors are injected to each parameters of self-attention for SRAM CiM in inference.

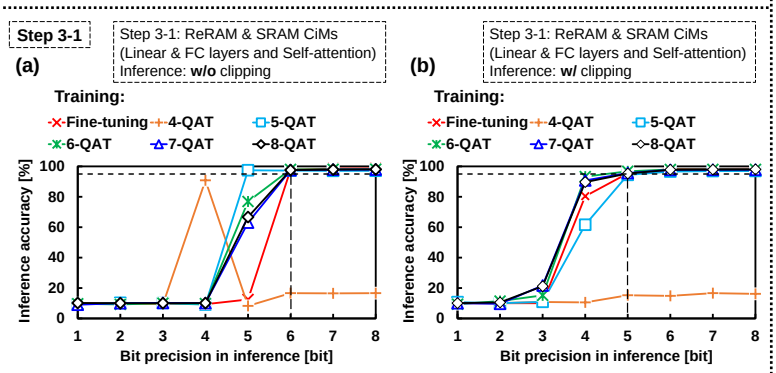


Fig. 11 Inference accuracy with quantization in Linear & FC layers and self-attention for both ReRAM and SRAM CiMs (a) without clipping and (b) with clipping.

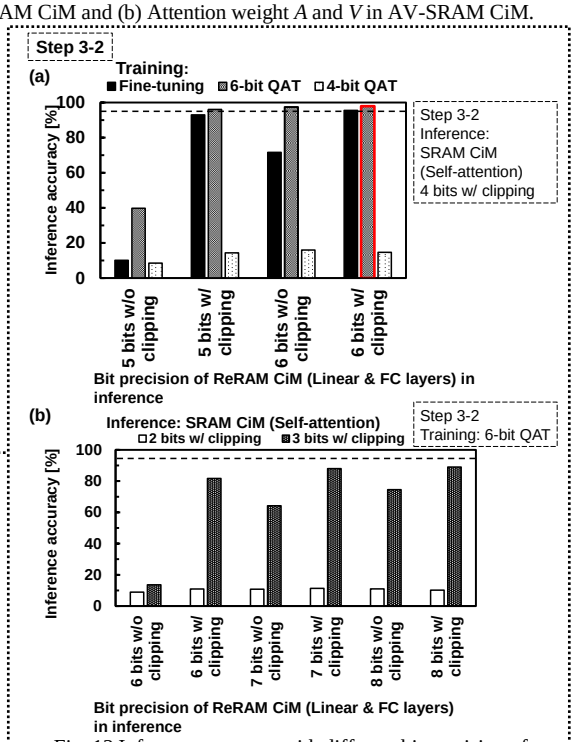


Fig. 12 Inference accuracy with different bit precision of ReRAM CiM when (a) bit precision of SRAM is 4 bits with clipping and (b) bit precision of SRAM is further smaller than 4 bits.

Table III Comparison of proposed Transformer Hetero-CiM with conv. ViT, SRAM CiMs, and ReRAM CiMs

Training: 6-bit QAT, Inference: w/ clipping

Inference		Conv. ViT	SRAM CiMs	SLC ReRAM CiMs	Proposed Transformer Hetero-CiMs		
					SLC ReRAM CiM	MLC ReRAM CiM	TLC ReRAM CiM
Linear & FC layers	Bit precision	32 bits	5 bits	5 bits	6 bits	6 bits	6 bits
	Memory bits	2.73G	427M	427M	513M	513M	513M
	Area [F²]	-	128G	25.7G	30.8G	15.4G	10.3G
Self-attention	Bit precision	32 bits	5 bits	5 bits	SRAM CiM		
					4 bits	4 bits	4 bits
	Memory bits	115M	18.1M	18.1M	14.4M	14.4M	14.4M
	Area [F²]	-	5.42G	1.08G	4.34G	4.34G	4.34G
Total area [F²]		-	134G	26.7G	35.1G	19.7G	14.6G
Inference accuracy [%]		98.2% [1]	96.8 (Fig. 11 (b))	96.8% (Fig. 11 (b))	+ 1.1%	97.9% (Fig. 12 (a))	

Step 1 Step 3-2 Step 3-1 - 45.3% - 89.1%

*Area = Memory bits × 2 (a differential pair) × Cell size

Table IV Acceptable bit-error rate (BER) of proposed Transformer Hetero-CiM

Error type	Linear & FC layers (ReRAM CiM)	Self-attention (SRAM CiM)	Linear& FC layers and self-attention (ReRAM & SRAM CiMs)
Gaussian σ [n.s.]	0.02	0.08	0.02
Shift Δ [n.s]	0.04	0.04	< 0.04

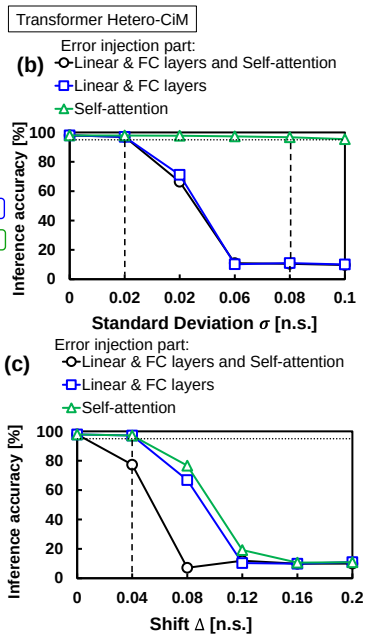


Fig. 13 (a) Design of proposed Transformer Hetero-CiM. Inference accuracy when (b) Gaussian errors and (c) shift errors are injected into proposed Transformer Hetero-CiM in inference.