4b/4b/8b Precision Charge-Domain 8T-SRAM Based CiM for CNN Processing

Qibang Zang*†, Wang Ling Goh*, Yi Sheng Chong†, Anh Tuan Do†
*School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore
†Institute of Microelectronics, A*STAR, Singapore
*qibang001@ntu.edu.sg, *ewlgoh@ntu.edu.sg, †chong_yi_sheng@ime.a-star.edu.sg, †doat@ime.a-star.edu.sg

***Abstract*—Compute-in-memory (CiM) is a promising solution for solving the bottleneck of frequent data movement between the memory and processor in Von-Neumann architecture. In conventional multi-bit CiM architecture, when computing N-bit input and N-bit weight MAC operation, 2^N-1 cycles are needed for N-bit input modulation and normally 3-4 cycles with complex switch operation are needed for N-bit weight realization, which significantly degrades the final throughput and power efficiency. In this work, a C-2C DAC built in the 8T SRAM CiM array is designed for 4-bit weight and 4-bit input MAC operation, which can be completed in just one cycle. In the final power efficiency evaluation, our 4b/4b/8b CiM architecture attained up to 640 TOPS/W (normalized to 1b/1b/1b precision) which is a 6-10 times improvement as compared to the conventional multi-bit CiM architectures. The proposed architecture with 4b/4b/8b precision can provide 91.47% and 68.10% accuracy on CIFAR-10 and CIFAR-100 dataset classification, respectively.**

***Index Terms*—SRAM, compute-in-memory (CiM), multiply-and-accumulate (MAC).**

I. INTRODUCTION

Compute-in-memory (CiM) is a popular solution for Von-Neumann architecture bottleneck due to frequent data movement between the processor and memory. Binary neural network (BNN) or ternary neural network (TNN) based CiM [1]–[5] are popular since, even with the use of binary or ternary weight precision, some classification tasks can still achieve high accuracy. Examples of such are the MNIST and SVHN with $\geq 95\%$ accuracy and high power efficiency up to 800 TOPS/W [1]. However, when BNN or TNN are used for complex classification tasks, such as CIFAR-10 and CIFAR-100, the classification accuracy is not as satisfactory as the accuracy of MNIST and SVHN, which is always less than 85% [1]–[5]. A multi-bit CiM architecture is therefore necessary to support higher precision neural network processing for complex classification tasks.

In the existing multi-bit CiM works [6]–[11], some [7]–[9] require 2^N-1 pulses width/counts modulation to realize N-bit input causing a throughput degradation, while some [6], [7], [9], [10] need complex switch operations (normally 3-4 cycles) to achieve N-bit weight which also leads to a throughput degradation. Some CiM works [7]–[9], [11] that are based on SRAM discharge current (I_{dis}) accumulation for MAC operations, degrade MAC curve linearity and variation since I_{dis} is easily affected by process-voltage-temperature (PVT) conditions and has a relatively larger variation as illustrated in Fig. 1 (b) than voltage-domain CiM works [4], [5] or charge-domain CiM works [1]–[3], [6], [10].

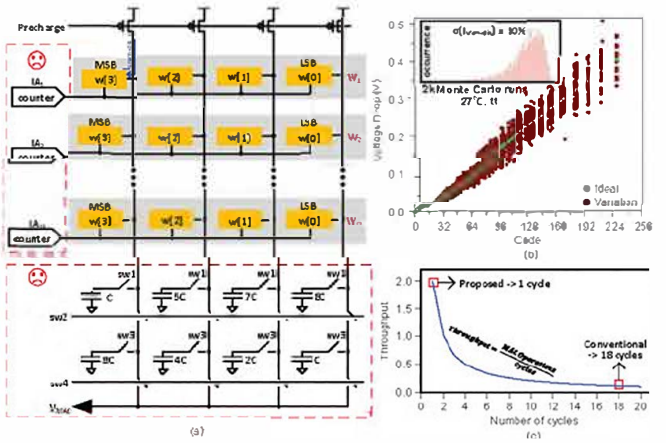


Fig. 1. (a) Existing 4b/4b CiM architecture [6]–[9]; (b) I_{dis} variation (σ = 10%) based on 2k Monte-Carlo runs in current-domain CiM [7]–[9] and its effect on the final 4b/4b MAC operation corresponding analog voltage (V_{MAC}); (c) A normalized throughput curve with conventional works [7]–[9] and this work highlighted.

Fig. 1 (a) shows the conventional 4b input and 4b weight CiM architecture [6]–[9]. The realization of 4b input requires 2^3-1 cycles for pulse width/counts modulation while the 4b weight realization necessitates 3-4 cycles with complex switch operations. In addition, the current-domain MAC computation is easily affected by I_{dis} variation (σ), as illustrated in Fig. 1 (b). We can see that 10% I_{dis} variation obtained from 2k Monte Carlo runs degrades the 4b/4b MAC operation, i.e., the corresponding analog voltage (V_{MAC}) linearity and variation. Fig. 1 (c) shows the normalized throughput curve plotted according to the equation $Throughput = \frac{MAC\ Operations}{Cycles}$ between the architecture shown in Fig. 1 (a) and this work, and to be discussed later in the paper.

The contributions of this work are as follow: 1) a C-2C DAC built in the 8T SRAM CiM array is proposed to solve throughput degradation issues, where only one cycle is required to complete multi-bit MAC operation, therefore achieving as high power efficiency (TOPS/W) as the BNN or TNN based CiM works [1]–[5], 2) charge-domain multi-bit CiM architecture is designed with no significant MAC

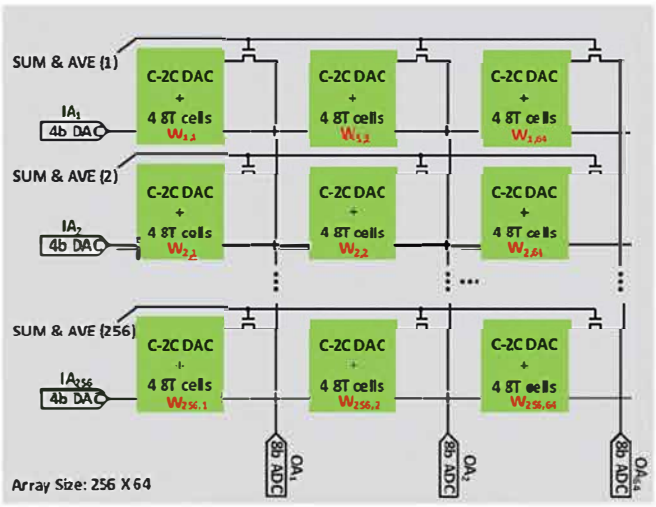


Fig. 2. The overall CiM architecture with array size 256×64 for convolutional neural network (CNN) mapping and processing.

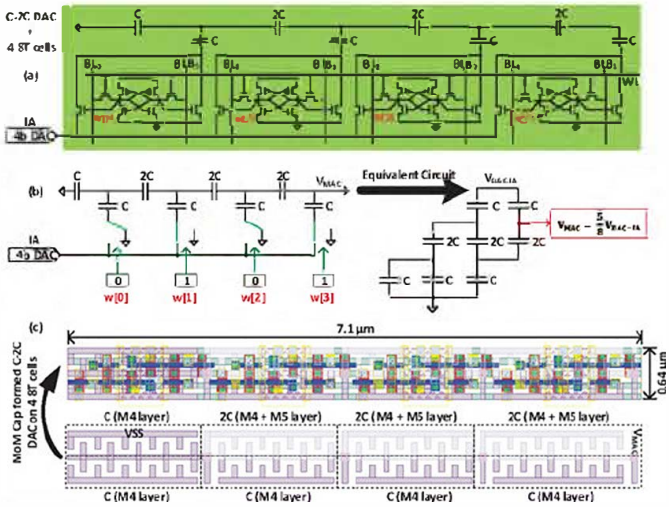


Fig. 3. (a) Schematic diagram of the built-in C-2C DAC and four 8T SRAM cells; (b) Illustration of the built-in C-2C DAC and 4 8T SRAM cells function as a 4b/4b MAC unit with W[3:0] = '1010'; (c) The logic rule based layout of the built-in C-2C DAC and four 8T SRAM cells.

curve linearity and variation degradation as current-domain CiM works [7]–[9].

II. 4b/4b/8b Charge-Domain CiM Design

Fig. 2 (a) shows the overall CiM architecture with array size 256×64 for convolutional neural network (CNN) mapping and processing. The overall CiM architecture consists of an external 4b DAC for input activation (IA) generation in each row, the built-in C-2C DAC and 4 8T SRAM cells formed compute-in-memory array, and the 8b SAR ADC for output activation (OA) in each column. Each bitcell of this array consists of a C-2C DAC for processing 4b/4b MAC operation and four 8T SRAM cells for 4b weight storage to control the C-2C DAC inputs. In this array, the maximum simultaneously opening rows are 9 and each row is controlled by the summation and average (SUM and AVE) signals. Once the corresponding analog voltage (V_{MAC}) at each column of 4b/4b MAC operation is ready, the 8b SAR ADC will transfer V_{MAC} to the 8b digital outputs.

Fig. 3 (a) shows the schematic of the proposed built-in C-2C DAC and four 8T SRAM cells. The 4b weight W[3:0] (W[3] represents MSB and W[0] represents LSB) is stored in 4 6T SRAM cells, which controls corresponding two NMOS access transistors. When W = '1' and $\overline{W}$ = '0', the 4b input corresponding voltage (V_{DAC-IA}) generated by the 4b DAC is conductive and is applied to the built-in C-2C DAC; when W = '0' and $\overline{W}$ = '1', GND is conductive and is applied to the built-in C-2C DAC. Fig. 3 (b) provides an example to illustrate how the built-in C-2C DAC and four 8T SRAM cells function as a 4b/4b MAC unit. If the stored weight W[3:0] = '1010', when W[3] is '1', V_{DAC-IA} will be selected and applied to the built-in C-2C DAC; when W[2] is '0', GND will be selected and applied to the built-in C-2C DAC; when W[1] is '1', V_{DAC-IA} will be selected and applied to the built-in C-2C DAC; when W[0] is '0', GND will be selected and applied to the built-in C-2C DAC. Therefore, the equivalent circuit can be deduced and the output voltage is resolved as $V_{MAC} = \frac{5}{8} * V_{DAC-IA}$. Fig. 3 (c) shows the logic rule based layout of the built-in C-2C DAC and four 8T SRAM cells. The built-in C-2C DAC is realized by Metal-Oxide-Metal (MOM) based sandwich and fringing structure, which avoid the area overhead issue. According to parallel plate capacitor equation, $C = \frac{\varepsilon_0 \times A}{d}$, when area doubles capacitance doubles. The unit capacitor C is formed using a single M4 layer and 2C is formed using both M4 layer and M5 layer.

Fig. 4 (a) gives the MOM capacitor value extracted from the layout shown in Fig. 3 (c) and evaluates the MOM capacitor variation (σ). We can see that the average capacitor C value is 0.488 fF with a maximum variation (σ) of 3.7% and the average capacitor 2C value is 0.938 fF with a maximum variation (σ) of 2.1%, the capacitor variation (σ) mainly comes from the parasitic capacitance ($C_{parasitic}$) between C and 2C, causing mismatch in the layout. Fig. 4 (b) shows the 3D plot for 4b/4b MAC value corresponding analog voltage (V_{MAC}) with the capacitor variation (σ) shown in Fig. 4 (a) when the number of rows turned on simultaneously increases from 0 to 9, according to the equation $V_{MAC} = \frac{rows}{ROWS} * \{V_{DAC-IA} \frac{\sum_{i=0}^{N-1} IN[i]*2^i}{2^N}\} \{\frac{\sum_{j=0}^{N-1} W[j]*2^j}{2^N}\}$, where V_{DAC-IA} represents the input DAC power supply and is $\frac{VDD}{2}$ since the two access NMOS are controlled by 6T SRAM cells which allows conducting at the voltage lower than $\frac{VDD}{2}$.

Fig. 5 (a) shows a comparison table of existing DAC structures used in SAR ADC. Binary-weighted DAC is widely used in SAR ADC due to its straight forward structure and small parasitic capacitor. However, the number of capacitor value used in binary-weighted DAC is determined by the resolution. In order to alleviate the capacitance mismatch effect and reduce the number of capacitor value used in binary-weighted DAC, the resolution cannot be more than 8 bits. The split-weighted DAC structure uses a bridge capacitor to

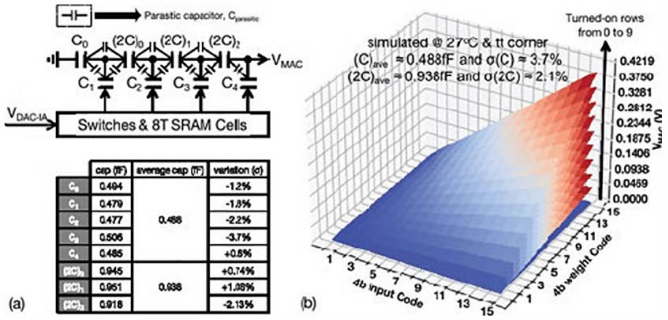


Fig. 4. (a) The MOM capacitor C and 2C value and the variation analysis based on the layout given in Fig. 3 (c); (b) The 3D plot for 4b/4b MAC value corresponding analog voltage (V_{MAC}) when the number of rows turned on simultaneously increasing from 0 to 9.

split the MSB capacitor array and LSB capacitor array to alleviate the capacitance mismatch effect, and reduces the number of capacitor value in high resolution applications at, say 9 to 12 bits. The C-2C DAC structure has only two kinds of capacitor values, but the parasitic capacitance is large which limits its resolution, say 3 to 6 bits. Fig. 5 (b) shows the overall architecture of an 8b SAR ADC, consists of a boostrapped switch for high speed sampling, an 8b binary-weighted DAC, an 8b SAR logic and a self-calibrated single-ended sense amplifier (SA). Fig. 5 (c) shows the operation principle of the SA. In auto-zero phase, the inverter input and output are connected to store the offset voltage on C_{AZ} and a short current (I_{short}) flows through PMOS to NMOS since both PMOS and NMOS are switched on when the inverter input and output are connected. In the comparison phase, the inverter input and output are disconnected, and the comparison decision is made when $V_{MAC} > V_{cDAC}$ the SA output is '1', when $V_{MAC} < V_{cDAC}$ the SA output is '0'. Thanks to auto-zero phase, the SA can achieve small offset (1.7 mV) with small area.

Fig. 6 (a) shows the Monte-Carlo results of the 8b SAR ADC. In the simulation, an analog input signal has a full-scale range that increases from 0 to VDD and the corresponding 8b digital output increases from 00000000 to 11111111. The average DNL (LSB) and INL (LSB) of the Monte-Carlo runs are calculated, which is -0.2/+0.2 and -0.2/+0.05, respectively. In addition, several defects occurs in around FSR/2, but these defects do not affect the final software inference accuracy. Fig. 6 (b) shows the power breakdown analysis for the 8b SAR ADC, we can see that the SA power occupies 67.37% of the ADC total power consumption, SAR control logic occupies 28.77% of the ADC total power consumption and the 8b binary-weighted DAC only occupies 3.87% of the ADC total power consumption. Fig. 6 (c) shows a performance summary of the 8b SAR ADC, our work has a relatively better linearity compared to other SAR ADC works [12]–[14], and this is because the SA used in our work supports full-scale range comparison and has smaller offset due to self-calibration. As for the figure of merit (FoM) which can be calculated using the equation $FoM = \frac{power}{f_s \times 2^{ENOB}}$, our work achieves

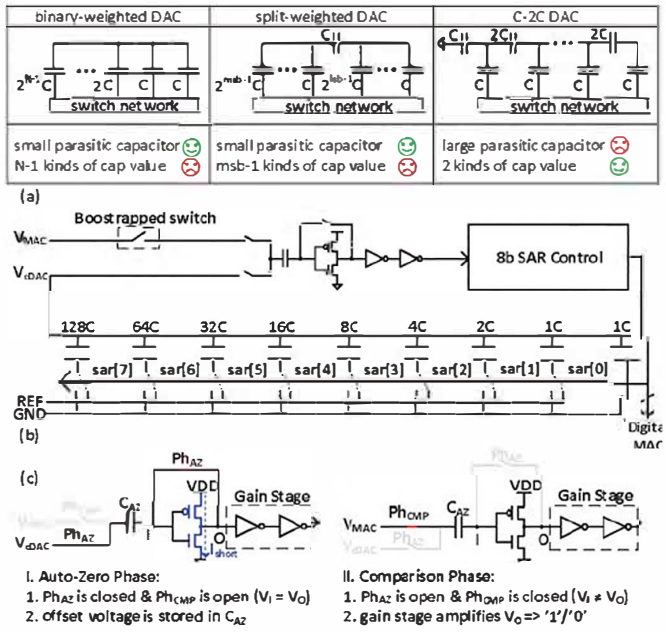


Fig. 5. (a) A comparison table of existing DAC structures used in SAR ADC; (b) the overall architecture of 8b SAR ADC; (c) the operation principle of the a self-calibrated single-ended SA.

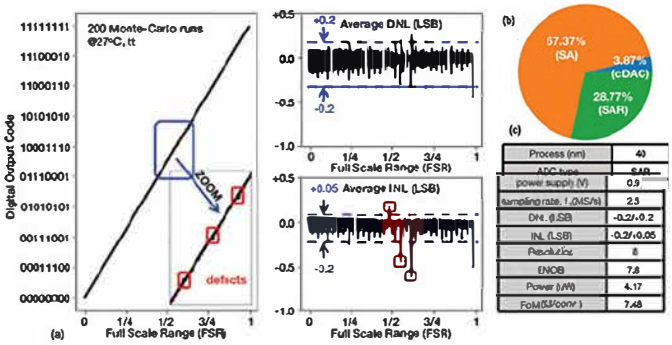


Fig. 6. (a) Monte-Carlo results of the 8b SAR ADC; (b) Power breakdown analysis of the 8b SAR ADC; (c) A performance summary of the 8b SAR ADC.

7.48 fJ/conv, which is higher than other works [12]–[14] after considering technology node scaling due to relatively high SA power consumption caused by I_{short} flowing through PMOS to NMOS directly in SA auto-zero phase.

III. RESULTS AND DISCUSSION

Fig. 7 shows the VGG-like neural network structure used for CIFAR-10 and CIFAR-100 image (32×32×3 pixels) classification, and convolutional layers mapping strategy on the proposed CiM architecture shown in Fig. 2. There are four convolutional layers in our VGG-like neural network structure. The number of filters and corresponding bitcells for weight storage at each convolutional layer are as follow: in conv1 layer, there are 64 filters occupying 9×64 bitcells for weight storage, in conv2 layer, there are 128 filters occupying 18×64 bitcells for weight storage, in conv3 layer, there are 256 filters

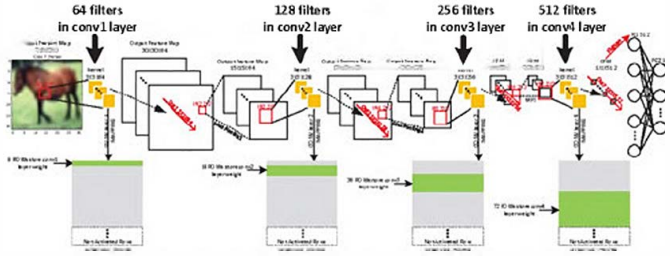
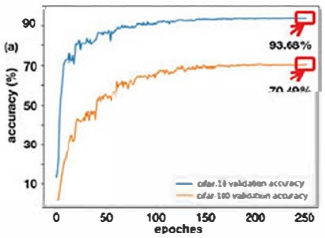


Fig. 7. The VGG-like neural network structure used for CIFAR-10 and CIFAR-100 image (32×32×3 pixels) classifications, as well as the four convolutional layers mapping strategy on the hardware shown in Fig. 2.



(b)

Dataset	MNIST	CIFAR-10	CIFAR-100
Neural Network	CNN	VGG-like	
FP accuracy	99.14%	93.68%	70.49%
1b/1b/1b accuracy/ Accuracy loss	97.39%/ -1.75%	80.02%/ -13.66%	55.05%/ -15.44%
4b/4b/8b accuracy/ Accuracy loss	98.77%/ -0.37%	91.47%/ -2.21%	68.10%/ -2.39%

Fig. 8. (a) The floating point validation accuracy curve of CIFAR-10 and CIFAR-100 image classification, respectively, using the VGG-like neural network; (b) A summary of floating point accuracy, binary accuracy/accuracy loss and 4b/4b/8b quantization accuracy/accuracy loss in this work for MNIST/CIFAR-10/CIFAR-100.

occupying 36×64 bitcells for weight storage, in conv4 layer, there are 512 filters occupying 72×64 bitcells for weight storage. The remaining rows are not activated.

Fig. 8 (a) shows the floating point validation accuracy curve of CIFAR-10 and CIFAR-100 image classification, respectively, using the VGG-like neural network shown in Fig. 7. We can see that the floating point accuracy of CIFAR-10 and CIFAR-100 is 93.68% and 70.49% respectively. However, after 4b/4b/8b quantization in this work, CIFAR-10 and CIFAR-100 image classification accuracy drops to 91.47% and 68.10% with 2.21% and 2.39% accuracy loss, respectively, as compared to the floating point accuracy as summarized in Fig. 8 (b). Meanwhile, MNIST/CIFAR-10/CIFAR-100 binary accuracy are analyzed as a comparison to our work. We can see in Fig. 8 (b) that the binary accuracy for MNIST/CIFAR-10/CIFAR-100 are 97.39%/80.02%/55.05% with -1.75%/-13.66%/-15.44% accuracy loss, respectively, as compared to the floating point accuracy. Therefore, when applied to complex image classification task (CIFAR-10/CIFAR-100) higher precision neural network can significantly improve the final inference accuracy, but, when applied to less complex image classification (MNIST), the binary neural network is sufficient.

Fig. 9 shows the comparison between this work and other state-of-art CiM works. This work proposed a built-in C-2C DAC multi-bit CiM architecture to solve the throughput and power efficient degradation issue in conventional multi-bit CiM works [6]–[8], [10], [11], [15]. We can see that in the conventional multi-bit CiM work [6], [8], [11], [15], the power efficiency is much lower than that of BNN/TNN works [1]–[3], [5]. Moreover, our work uses the 4b/4b/8b precision MAC

	[8] TCAS-I'20	[6] ISCAS'21	[15] AICAS'20	[5] JSSC'20	[11] JSSC'20	This work
Process (nm)	65	65	40	65	55	40
Cell Type	6T	8T	6T	12T	8T	8T
Logic Rule Layout	Yes	Yes	No	Yes	No	Yes
Cell Area	343 f²	657 f²	240 f²	927 f²	318 f²	710 f²
Operation Domain	Current	Charge	NA	Charge	Current	Charge
# of rows truned on simutaneously	NA	128	NA	256	9/18	9
IN/W/Out bits	5/5/4	4/4/4	8/8/8	1/1/3.64	1/2/3 2/5/5 4/5/7	4/4/8
Power efficient (TOPS/W) normalize to 1b ops	NA	69.33	196.2	403	18.37~72.03	640
Dataset	CIFAR-10	NA	ImageNet	CIFAR-10	CIFAR-10	CIFAR10
Network	VGG-like	NA	MobileNetV2	VGG-like ResNet-14	CNN	VGG-like
Accuracy	88.83%	NA	NA	87.23% 85.72%	85.56% 90.2% 90.42%	91.47%

Fig. 9. Comparison table between the proposed work and the state-of-art CiM works

operation, where the final evaluated power efficiency does not suffer much as the conventional ones [6]–[8], [10], [11], [15], since our proposed built-in C-2C DAC and 8T SRAM cells structure consumes less power which is 543.7 pW due to low C-2C DAC input voltage (from 0 to $\frac{VDD}{2}$) and small capacitor dynamic power (small MOM capacitor value), and the main power consumption comes from the 8b SAR-ADC and the 4b input activation DAC. In addition, our proposed built-in C-2C DAC does not require extra area and the average 8T SRAM cell area is 710 f^2 which is reasonable compared to other works [5], [6], [8], [11], [15]. In our work, the software inference accuracy using 4b/4b/8b VGG-like neural network for CIFAR-10 is 91.47% which is competitive compared to other multi-bit CiM works [8], [10], [11].

IV. Conclusion

This paper proposed a 4b/4b/8b precision charge-based CiM architecture for convolutional neural network processing. Compared to conventional multi-bit CiM works [6]–[11], which requires $2^N - 1$ cycles for N-bit input pulse width/counts modulation and 3-4 cycles with complex switch operation for N-bit weight realization, this work mandates just one cycle to complete the 4b/4b MAC operations, which is credited to the C-2C DAC built in the 8T SRAM CiM array, significantly improves the throughput of the multi-bit input and multi-bit weight MAC operation. In the final power efficiency evaluation, our work achieves 640 TOPS/W, which is 6 to 10 times higher than the conventional designs [6]–[8], [11]. In addition, the built-in C-2C DAC in our work does not call for extra area since the capacitor is based on MOM sandwich and fringing structure.

V. Acknowledgement

This research is supported by Programmatic grant no. A18A6b0057 (SpOT-Lite), Singapore RIE2020 Advanced Manufacturing and Engineering domain.

REFERENCES

[1] H. Valavi, P. J. Ramadge, E. Nestler, and N. Verma, "A 64-tile 2.4-mb in-memory-computing cnn accelerator employing charge-domain compute," *IEEE Journal of Solid-State Circuits*, vol. 54, no. 6, pp. 1789–1799, 2019.

[2] Z. Jiang, S. Yin, J.-S. Seo, and M. Seok, "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.

[3] A. Biswas and A. P. Chandrakasan, "Conv-sram: An energy-efficient sram with in-memory dot-product computation for low-power convolutional neural networks," *IEEE Journal of Solid-State Circuits*, vol. 54, no. 1, pp. 217–230, 2019.

[4] X. Si, W.-S. Khwa, J.-J. Chen, J.-F. Li, X. Sun, R. Liu, S. Yu, H. Yamauchi, Q. Li, and M.-F. Chang, "A dual-split 6t sram-based computing-in-memory unit-macro with fully parallel product-sum operation for binarized dnn edge processors," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 11, pp. 4172–4185, 2019.

[5] S. Yin, Z. Jiang, J.-S. Seo, and M. Seok, "Xnor-sram: In-memory computing sram macro for binary/ternary deep neural networks," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 6, pp. 1733–1743, 2020.

[6] V. Sharma, J. E. Kim, Y.-J. Jo, Y. Chen, and T. T.-H. Kim, "And8t sram macro with improved linearity for multi-bit in-memory computing," in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2021.

[7] Q. Dong, M. E. Sinangil, B. Erbagci, D. Sun, W.-S. Khwa, H.-J. Liao, Y. Wang, and J. Chang, "15.3 a 351tops/w and 372.4gops compute-in-memory sram macro in 7nm finfet cmos for machine-learning applications," in *2020 IEEE International Solid- State Circuits Conference - (ISSCC)*, pp. 242–244, 2020.

[8] M. Ali, A. Jaiswal, S. Kodge, A. Agrawal, I. Chakraborty, and K. Roy, "Imac: In-memory multi-bit multiplication and accumulation in 6t sram array," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 8, pp. 2521–2531, 2020.

[9] N. Gupta, A. Joshi, D. Kushwaha, V. Menezes, R. Sachan, S. Dasgupta, and A. Bulusu, "A multibit mac scheme using switched capacitor based 3c multiplier for analog compute in-memory architecture," in *2022 29th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pp. 1–4, 2022.

[10] G. Yin, M. Zhou, Y. Chen, W. Tang, Z. Yang, M. Lee, X. Du, J. Yue, J. Liu, H. Yang, *et al.*, "A 65nm 8b-activation 8b-weight sram-based charge-domain computing-in-memory macro using a fully-parallel analog adder network and a single-adc interface," *arXiv preprint arXiv:2212.04320*, 2022.

[11] X. Si, J.-J. Chen, Y.-N. Tu, W.-H. Huang, J.-H. Wang, Y.-C. Chiu, W.-C. Wei, S.-Y. Wu, X. Sun, R. Liu, S. Yu, R.-S. Liu, C.-C. Hsieh, K.-T. Tang, Q. Li, and M.-F. Chang, "A twin-8t sram computation-in-memory unit-macro for multibit cnn-based ai edge processors," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 1, pp. 189–202, 2020.

[12] K.-H. Chang and C.-C. Hsieh, "A calibration-free 12-bit 50-ms/s full-analog sar adc with feedback zero-crossing detectors," *IEEE Journal of Solid-State Circuits*, vol. 54, no. 6, pp. 1624–1635, 2019.

[13] S. Ahn, J. Song, C. Lim, Y. Choi, S. Park, Y. Park, and C. Kim, "A 1 ms/s 9.15 enob low-power sar adc with triple-charge-sharing technique," in *2020 International SoC Design Conference (ISOCC)*, pp. 1–2, 2020.

[14] D. Verma, K. Shehzad, D. Khan, Q. U. Ain, S. J. Kim, D. Lee, Y. Pu, M. Lee, K. C. Hwang, Y. Yang, and K.-Y. Lee, "A design of 8 fj/conversion-step 10-bit 8ms/s low power asynchronous sar adc for ieee 802.15.1 iot sensor based applications," *IEEE Access*, vol. 8, pp. 85869–85879, 2020.

[15] C.-Y. Lo, P.-T. Huang, and W. Hwang, "Energy-efficient accelerator design with 3d-sram and hierarchical interconnection architecture for compact sparse cnns," in *2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, pp. 320–323, 2020.