IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 4 October 2024, accepted 26 October 2024, date of publication 30 October 2024, date of current version 13 November 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3487975

RESEARCH ARTICLE

Union SRAM: PVT Variation Auto-Compensated, Bit Precision Configurable Current Mode 8T SRAM in Memory MAC Macro

HONGGU KIM, YERIM AN, RYUNYEONG KIM, SUNYOUNG KIM, AND YONG SHIM
School of Electrical and Electronic Engineering, Chung-Ang University, Dongjak-gu, Seoul 06974, Republic of Korea

Corresponding author: Yong Shim (yongshim@cau.ac.kr)

This work was supported in part by the Chung-Ang University Graduate Research Scholarship in 2022, in part by the National Research Foundation of Korea (NRF) through Korean Government [Ministry of Science and ICT (MSIT)] under Grant 2021R1C1C100875214, and in part by the National Research Council of Science and Technology (NST) through Korean Government (MSIT) under Grant GTL24041-000.

ABSTRACT SRAM-based Compute-In-Memory (CIM) has two main paradigms: Digital domain and Analog domain, where both have been extensively explored to overcome the von-Neumann bottleneck and enhance energy efficiency. Digital CIM offers robustness and dynamic bit-precision through bit-wise and bit-serial computing, but suffers from limited throughput due to multi-cycle operations and degraded area density due to large hardware footprints. In contrast, Analog CIM offers the significantly improved throughput and area density for the analog computing nature and simple logic structure. However, the weight and input data are constrained by fixed bit-precision, limiting the flexibility in DNN applications. Additionally, Analog CIM is susceptible to process, voltage, and temperature (PVT) variations, resulting in potential accuracy degradation. We present a solution to the limitations of analog domain SRAM CIM in dynamic bit-precision configurability and PVT variation vulnerability. Our proposed 4b/8b bit-precision configurable analog current-mode 8T SRAM CIM architecture enhances DNN application flexibility. We also introduce PVT variation auto-compensation scheme, effectively maintaining precise computing accuracy of the analog domain CIM. Post-layout simulations confirm the architecture's efficacy, achieving a throughput of 170 to 793.4 GOPs, area efficiency of 0.227 to 1.06 TOPs/mm^2, and energy efficiency of 5.1 to 23.76 TOPs/W. Additionally, software-level simulation on the CIFAR-10 dataset demonstrates 95.02 percent classification accuracy.

INDEX TERMS SRAM compute-in-memory, DNN accelerator, PVT auto-compensated MAC macro.

I. INTRODUCTION

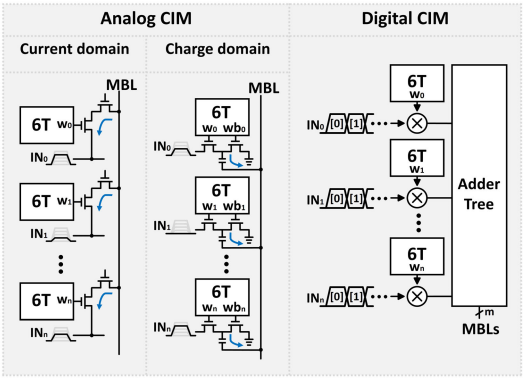
Recent advancements in Deep Neural Networks (DNNs) have led to a substantial escalation in data processing requirements. However, the device dimension and voltage scalings have approached its physical limits [1], failing to meet the increasing demands for data processing and transfer. This has further exacerbated the energy bottleneck inherent in von-Neumann architecture [2]. To address this energy problem, Compute-In-Memory (CIM) based DNN accelerators have gained attention, offering a promising solution for efficiently implementing data-intensive machine learning algorithms on power-constrained hardware like edge devices, while enhancing energy efficiency.

SRAM based CIM is one of the most promising candidate for such DNN accelerator. As can be seen on Fig. 1(a), several types of SRAM based CIM structure has been investigated, namely Analog CIM and Digital CIM, where the Analog CIM is further categorized into two groups: current domain [3], [4], [5], [6], [7], [8], [9], [10] and charge domain [11], [12], [13], [14], [15], [16], [17].

Analog CIM schemes, current domain CIM and charge domain CIM, commonly exhibit notably high throughput, owing to the inherent parallelism of the analog MAC operation. However, these schemes are typically constrained to fixed-bit precision, limiting their adaptability across

The associate editor coordinating the review of this manuscript and approving it for publication was Fabian Khateb.

(a)



(b)

	Analog CIM		Digital CIM
	Current	Charge	
Throughput	☺	☺	☹
Area density	☺	☺	☹
Robustness	☹	☺	☺
Bit-precision Configurability	☹	☹	☺

☺ : excellent ☺ : moderate ☹ : poor

FIGURE 1. (a) Categorization of SRAM based Compute-in-Memory (CIM) and (b) performance comparison table of Analog CIM and Digital CIM.

diverse DNN applications. Besides, current domain CIM and charge domain CIM exhibit slightly different performances in terms of the area density and the robustness.

Current domain CIM scheme incorporates only few more transistors (typically 6 to 8) than the conventional 6T SRAM cell, giving a high benefit in terms of the area density. However, the bitcell current is highly susceptible to Process, Voltage and Temperature (PVT) variations, leading to considerable inaccuracy. Additionally, other analog peripheral circuits, such as Digital-to-Analog Converters (DACs), are also vulnerable to PVT variations, contributing to non-linearities in the MAC operations.

In contrast, the charge-domain CIM scheme typically involves more than 8 transistors along with Metal-Oxide-Metal (MOM) capacitor. While this slightly reduces area efficiency compared to current-domain CIM counterparts, the charge-domain scheme offers improved robustness. The MOM capacitors exhibit excellent robustness against PVT variations, thereby enhancing bitcell reliability. Nevertheless, the peripheral analog circuits remain susceptible to PVT-induced inaccuracies, leaving the concerns of the inaccuracy associated with the computational inaccuracies.

To resolve the adverse effect of PVT variations in the Analog CIM architectures, PVT calibration scheme using calibration engine for analog domain CIM architecture has been proposed in [18]. The calibration engine proposed in [18] aligns the starting and ending points of MAC operation transfer curve to calibrate the distorted MAC operation range due to PVT variations. However, this scheme increases hardware level control overhead which makes the overall system complicated. Meanwhile, configurable Dot-product (DP) duration scheme to compensate PVT variations has been proposed in [19]. DP duration for MAC operation is tuned during training phase to fit the correct MAC operation range with the existence of PVT variations. But this technique necessitates both software level and system level controls, which increases overall system complexities.

On the other hand, Digital CIM [20], [21], [22], [23], [24], [25] exhibits excellent robustness against PVT variation for the digital domain computational flow. Furthermore, by leveraging bit-wise and bit-serial computation scheme, bit precision configurability is very high in digital CIM. However, digital CIM requires multiple cycles to complete the multi-bit MAC operation in the bit-wise manner, resulting in the degradation in computational throughput. Furthermore, the area efficiency is far degraded due to large hardware footprints, consisting of multipliers and the adder trees distributed across the entire SRAM array. At this point it can be concluded that Analog CIM and Digital CIM typically have common trade-off between throughput/area density and robustness/bit-precision configurability, as can be seen on the table in Fig. 1(b).

In this work, we propose PVT auto-compensated, bit precision configurable current mode 8T SRAM in memory MAC macro. The main features of this work are described as follows

- To make a breakthrough in the limitation of fixed bit-precision configuration inherent in analog domain CIM macro, we enabled 4-bit, 8-bit configurability to input and weight data to facilitate machine learning algorithm application.
- The name Union SRAM implies the collaboration of whole circuit components (input DAC, SRAM array and output IV converter) towards the process and temperature auto-compensation process. This auto-compensation scheme does not necessitate any other hardware or software level components, simplifying the overall system.
- UNION SRAM structure along with voltage (V_{DD} and V_{SS}) variations auto-adjusted reference voltage of the 3-bit FLASH ADC effectively compensates the voltage variations.

To sum up, our work targets at resolving the main drawbacks of analog CIM, poor dynamic bit-precision configurability and robustness, while still leveraging high throughput and area density benefits.

In section II, The basic differences between 8T SRAM cell and 6T SRAM cell are discussed, where 8T SRAM cell was used in our work. In section III, the overall architecture structure and 4-bit parallel and 4-bit serial operation flows are presented. Additionally, we describe the process and temperature auto-compensation mechanisms of the UNION SRAM structure, as well as the techniques for auto-compensation of voltage (V_{DD} and V_{SS}) variations. In section IV, the performance overview of designed CIM architecture are discussed. Lastly in section V, the performance comparison with other state-of-the-art researches are discussed.

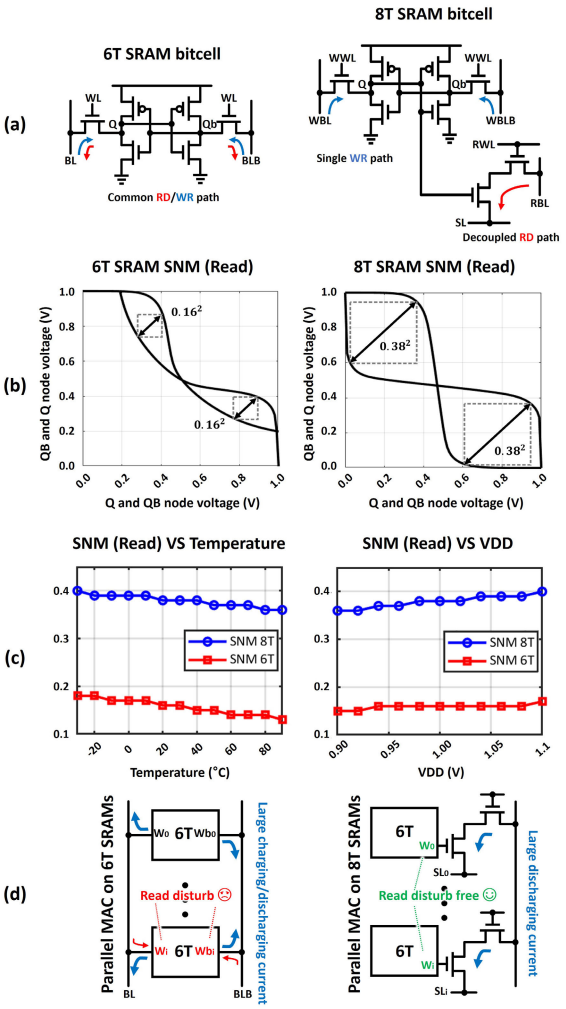


FIGURE 2. (a) Conventional 6T SRAM bitcell (left) and 8T SRAM bitcell (right), (b) Read Static Noise Margin (SNM) for 6T SRAM bitcell and 8T SRAM bitcell, (c) SNM variation with temperature and VDD sweep and (d) illustration of the parallel MAC operation on the two SRAM cell types and the related read disturb issue.

II. SRAM BASICS: 6T SRAM BITCELL AND 8T SRAM BITCELL

Fig. 2(a) shows the basic structure of the conventional 6T SRAM bitcell (left) and the 8T SRAM bitcell (right), employed in this work. The conventional 6T SRAM bitcell typically suffers from the data read or write failure due to small Static Noise Margin (SNM). On the contrary, the 8T SRAM bitcell mitigates this issue by decoupling the read path through the addition of a read port, consisting of two series-connected NMOS [26]. Therefore it is able to perform read operation in the 8T SRAM bitcell without the read failure. Moreover, since the read path is decoupled from the write path, it is able to optimize the write operation of the internal 6T SRAM by increasing the access transistors (connected to WWL signals), at the cost of increased cell area and hold-mode leakage. Fig. 2 (b) illustrates the read SNM of both SRAM bitcell structures. While SNM of 6T SRAM cell is small by 0.16^2, the 8T SRAM bitcell achieves a significantly higher SNM by 0.38^2, thanks to the decoupled read port.

Figure 2 (c) presents the variation in read SNM for each SRAM bitcell structure under different operating conditions. As temperature increases and supply voltage (VDD) decreases, the read SNMs for both structures exhibit a similar declining trend. However, the 6T SRAM bitcell's substantially lower SNM makes it more susceptible to read failures under these conditions, whereas the 8T SRAM cell retains stable read functionality in the practical read operation scenarios [26].

In the context of analog-domain Compute-In-Memory (CIM) architectures, performing parallel MAC operation across multiple bitcells is crucial. As illustrated in Figure 2 (d) (left), implementing such parallel operations in 6T SRAM-based CIM architectures is challenging due to the risk of data corruption caused by significant charging/discharging currents on the bitlines. In contrast, as shown in Fig. 2 (d) (right), 8T SRAM cell's decoupled read ports prevent data corruption by isolating the stored data from the read bitline currents. This isolation allows the implementation of parallel MAC operations, making the 8T SRAM bitcell more suitable to CIM applications.

III. OVERALL ARCHITECTURE

Fig. 3 describes the overall architecture. To realize MAC operation in common DNN applications, input activations are configured in the form of analog voltages by DACs and weights are stored in 8T SRAM arrays. There are totally 128 DACs each supporting 4 bit precision, for bigger the input value applied, smaller the DAC output voltage than reference voltage (V_{REF}) is injected to source lines of 8T SRAM array.

The whole 8T SRAM array storing weight parameters consists of 16 Banks (from Bank 0 to Bank 15), each having 4 columns of 8T SRAM bit-cells. In each row of one bank, there is one sign bit as was demonstrated in TWIN 8T [4] with readport scaled up by X2, while remaining 3 bit represents the magnitude with X1, X2 and X4 scaled up readport bitcells. 1 bit sign column and 3 bit magnitude columns have its own MAC-bitline (MBL), MBLS and MBLM. When an input value, represented as a voltage, is applied to the SLs and the corresponding weight stored in an 8T SRAM bitcell is 1, the multiplication of the input and the weight occurs at the 8T SRAM read port. The resulting current is then accumulated on the MBL, as illustrated in the inset of Fig. 3.

Two series-connected NMOS transistors that consist of read port of 8T SRAM cell operate in deep triode region for MAC operation linearity. The necessity for these NMOS transistors to operate in deep triode region stems from their role in enabling a linear increment of the bitcell current in response to increasing the input voltages (V_{IN}) that is fed to the drain node of the two NMOS transistors. If these two series connected NMOS transistors do not operate in deep triode region, the bitcell current will not increase in the linear

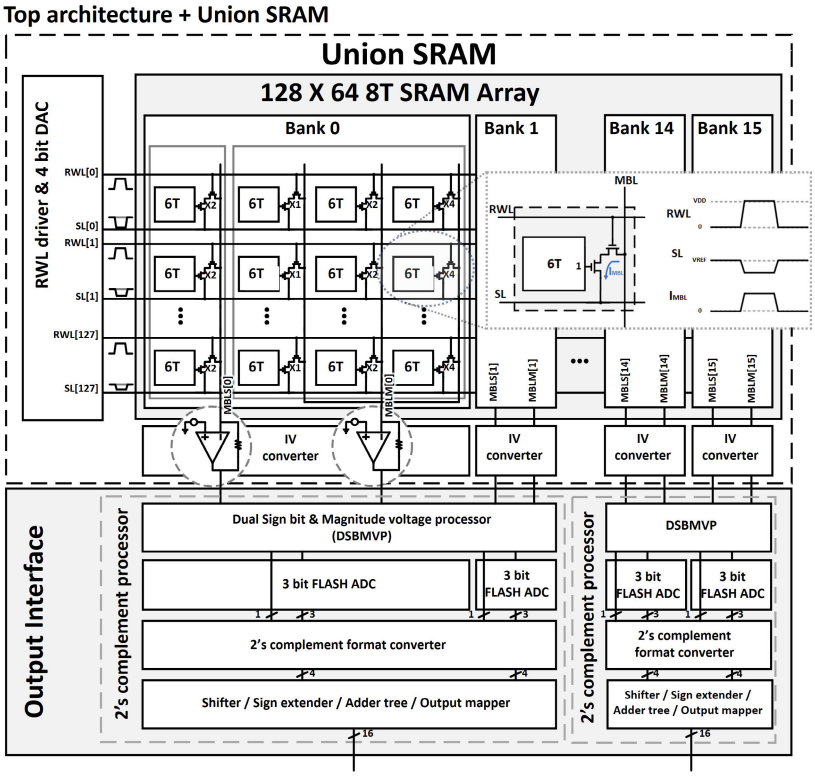


FIGURE 3. Top architecture of proposed 4/8 bit precision configurable current mode 8T SRAM CIM macro with union SRAM structure.

fashion with the increasing drain node voltages (V_{IN}), leading to non-linearities in the MAC operation. Deep triode region of 8T SRAM read port is guaranteed by fixing drain-source voltage of 8T SRAM readport using virtual short mechanism of OP AMP at IV converter stage. Positive input node of this OP AMP is fed with V_{REF} (200mV), and MBL voltage is clamped to V_{REF}, leading to constant drain-source voltage of 8T SRAM readport.

As can be seen on Fig. 3, the IV converter has been designed with OP AMP with feedback resistor. The feedback resistor of OP AMP placed underneath the sign bit column is scaled up by 4 times to realize X8 scaled up signed MAC operation together with X2 scaled up sign bit 8T SRAM bitcell. With input vectors multiplied by weights stored in 8T SRAMs, a pair of MAC values (in voltage) of each 1-bit sign column and 3-bit magnitude column resulting at the output of IV converter can be expressed as follows.

$$V_{sign} = 8 * \sum_{i=1}^{128} IN_i * W_i[0] \tag{1}$$

$$V_{mag} = \sum_{i=1}^{128} IN_i * (W_i[1] + 2 * W_i[2] + 4 * W_i[3]) \tag{2}$$

With all RWL drivers, 4 bit input DACs and 8T SRAM arrays simultaneously activated, fully-parallel MAC operations are realized within this CIM architecture.

A. 4 BIT-WISE PARALLEL OPERATION: 4/8 BIT WEIGHT CONFIGURATION

Two pair of MAC values (representing sign and magnitude) from two adjacent Banks are fed to one 2's complement processor. The first stage of 2's complement processor is Dual Sign Bit and Magnitude Voltage Processor (DSBMVP) as can be seen on Fig. 4(a), consisting of one 8b W MAC sign decision module (8bMSDM) and two Single MAC Sign and VOUT processor (SMSVP).

Fig. 4(b) and (d) show the MAC operation when 4 bit weight is configured, with 8bMSDM deactivated. Specifically, Fig. 4(d) illustrates an example when the MAC operation results come from the multiplications of 4 bit inputs (X0 = 1 and X1 = 2) with 4 bit weights (W0 = −3 and W1 = 1), which should result in digital codes −3 and 2 respectively, after the data conversion through DSBMVP, 3 bit FLASH ADC and 2's complement format converter. Initially, the MAC operation results (multiplications of X and W) come from an 8T SRAM array in current form and are converted to the voltage domain via an OP AMP based IV converter. Once the voltage output of IV converter is settled down, each SMSVP individually compares the magnitude of signed MAC value ($V_{MACS}[0]$ and $V_{MACS}[1]$) and magnitude MAC value ($V_{MACM}[0]$ and $V_{MACM}[1]$) to decide the sign bit (SIGN[0] and SIGN[1]) of corresponding MAC operation. Fig. 4(d) assumes that $V_{MACS}[0]$ is bigger than $V_{MACM}[0]$ at left SMSVP resulting in SIGN[0] = 1,

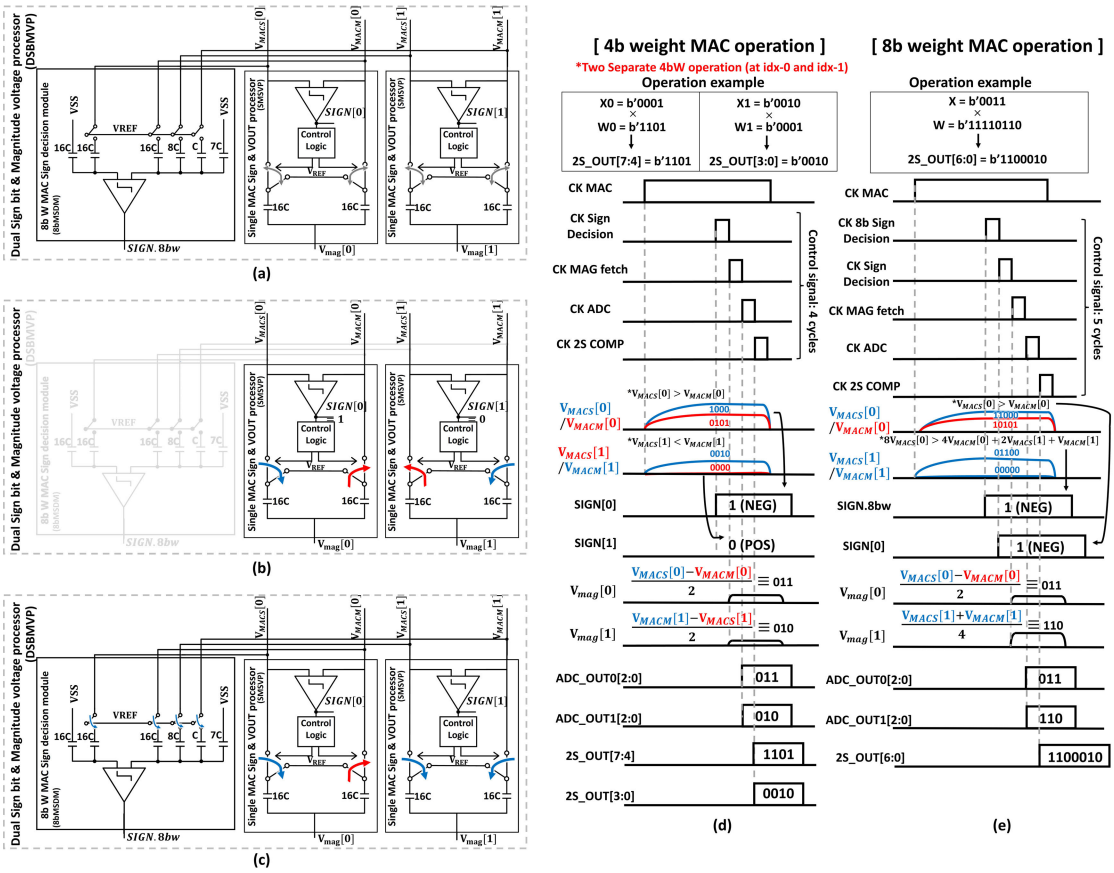


FIGURE 4. (a) Schematic of Dual Sign bit and Magnitude voltage processor (DSBMVP) and its operation (b) when weight is configured to be 4 bit precision with X0 = b'0001 multiplied by W0 = b'1101 resulting in 2S_OUT[7:4] = b'1101 and X1 = b'0010 multiplied by W1 = b'0001 resulting in 2S_OUT[3:0] = b'0010 and (c) when weight is configured to be 8 bit precision with X = b'0011 multiplied by W = b'11110110 resulting in 2S_OUT[6:0] = b'1100010.

while $V_{MACS}[1]$ is bigger than $V_{MACM}[1]$ at right SMSVP resulting in SIGN[1] = 0.

Depending on the resulting sign bit, following control logic and switches add and subtract those voltage magnitudes, resulting in voltage outputs ($V_{mag}[0]$ and $V_{mag}[1]$). In case of Fig. 4(d), $V_{mag}[0]$ is equal to $V_{MACS}[0]$ subtracted by $V_{MACM}[0]$, because $V_{MACS}[0]$ is bigger than $V_{MACM}[0]$ and the sign bit (SIGN[0]) is 1. On the other hand, $V_{mag}[1]$ is equal to $V_{MACM}[1]$ subtracted by $V_{MACS}[1]$, because $V_{MACM}[1]$ is bigger than $V_{MACS}[1]$ and the sign bit (SIGN[1]) is 0.

These V_{mag}s are fed to 3-bit FLASH ADC, resulting in 3 unsigend magnitude bits (ADC_OUT0[2:0] = 011 and ADC_OUT1[2:0] = 010). Resulting sign bits and unsigned 3 bit magnitude codes are now adjusted to 2's complement format by 2's complement format converter, resulting in 4 bit signed outputs (2S_OUT[7:4] = 1101 and 2S_OUT[3:0] = 0010) on a each column.

Meanwhile, Fig. 4(c) and (e) show the MAC operation with 8 bit weight configured. Specifically, Fig. 4(e) illustrates an example when the MAC operation results come from the multiplications of 4 bit inputs (X = 3) and 8 bit weights (W = −10), which should result in digital codes −30. Banks located at even column inside 8T SRAM array are configured with unsigned 4 bit LSB magnitudes, while Banks located at odd columns configure remaining 4 bit MSB values, 1 sign bit and unsigned 3 magnitude bits. To compute sign bit (SIGN.8bw) based on theses 8 bit data, 8bMSDM is activated which brings each MAC value via charge coupling scheme. Each MAC value is fetched with binary weighted manner regarding its placement value, and the sign bit corresponding to 8 bit weight MAC operation (SIGN.8bw) is decided. Fig. 4(e) shows the case that $8*V_{MACS}[0]$ is bigger than $4*V_{MACM}[0]+2*V_{MACS}[1]+V_{MACM}[1]$, resulting in SIGN.8bw = 1 at the output of 8bMSDM.

SMSVP located at even column simply adds up both MAC voltage values ($V_{MACS}[1]$ and $V_{MACM}[1]$) to obtain unsigned MAC results from LSB 4 bit columns, resulting in $V_{mag}[1]$. On the contrary, SMSVP located at odd column compares $V_{MACS}[0]$ and $V_{MACM}[0]$, and adds up the bigger one and subtracts the smaller one to obtain unsigned MAC results from MSB 4 bit columns, resulting in $V_{mag}[0]$. Each resulting voltages ($V_{mag}[0]$ and $V_{mag}[1]$) are fed to 3-bit FLASH ADCs, resulting in unsigned 6 bit magnitude outputs

(ADC_OUT0[2:0] = 011 and ADC_OUT1[2:0] = 110). Note that top reference voltage of FLASH ADC at even column is scaled up by 2 times to adjust to proper operating range.

Resulting one sign bit and unsigned 6 bit magnitude outputs are lastly converted to 2's complement format by 2's complement format converter, resulting in 7 bit signed outputs (2S_OUT[6:0] = 1100010). As can be seen on Fig. 4(e), 8b weight MAC operation mode takes one more clock cycle due to CK 8b Sign Decision clock which is needed for 8b weight sign decision.

B. 4BIT-WISE SERIAL OPERATION: 4/8 BIT INPUT CONFIGURATION

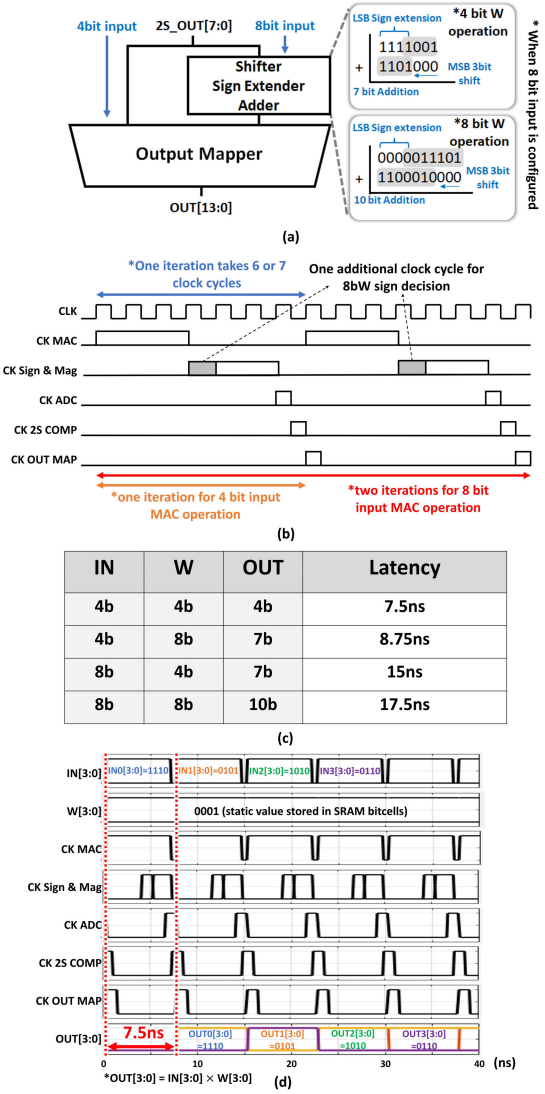
There are two common schemes to support multiple bit precision for input activation, adopting bit serial operation or adopting DAC. Adopting bit serial operation is superior to adopting DAC in terms of flexible bit precision configurability, while it suffers from degraded throughput. On the contrary, adopting DAC enhances throughput while it only supports fixed bit precision.

In our architecture, to enable flexible bit precision configurability while not to degrade throughput, we adopted 4 bit-wise serial operation which utilizes both DACs and bit-serial operation principle. Furthermore, the last block of output interface, shown in Fig. 5(a), consisting of shifter, sign extender, adder tree and output mapper is set to compute and stream out the final digital output bits in a 4 bit wise sequential manner, so as to enable the operation with different bit precision of input data (4 bit or 8 bit).

As can be seen on Fig. 5(a), if input data is 4 bit configured, then only output mapper works and it simply buffers out the MAC output in the form of digital bits. If input data is set to 8 bit, then the bit shifter, sign extender, adder tree and output mapper sequentially operate to stream out the final digital output bits. By configuring different bit precision for both input and weight data, the output bit-width dynamically changes from 4 to 10 bit.

Fig. 5(b) shows the timing diagram of our MAC macro, with system clock operating at 800MHz frequency in our architecture. Notably, the critical pulses, specifically the MAC pulse (CK_MAC) and CK Sign & Mag pulse, necessitate a substantial number of clock cycles (ranging from 5 to 6 cycles) to accommodate the settling time requirements of the OP AMPs in the input DAC and IV converter, as well as the switching capacitors in the DSBMVP, across all PVT scenarios. Also note that one more clock cycle is required during 8-bit W MAC operation due to 8bW sign decision phase.

Furthermore, as can be seen on Fig. 5(b), only one iteration is required for 4 bit input mode MAC operation, and two iterations are required for 8 bit input mode MAC operation to enable 4-bit wise serial accumulation operation. Taking these timing requirements into account, table in Fig. 5(c) shows the latency of MAC operation and output bit width with different input and weight bit precisions. Fig. 5(d) shows the operation



IN	W	OUT	Latency
4b	4b	4b	7.5ns
4b	8b	7b	8.75ns
8b	4b	7b	15ns
8b	8b	10b	17.5ns

FIGURE 5. (a) Operation of Shifter, Sign Extender, Adder and Output Mapper with different bit precision, (b) Timing Diagram of MAC operation, (c) Table that shows the latency of MAC operation with different bit precision and (d) MAC operation post layout simulation when 4 bit input and 4 bit weight is configured.

of 4 bit input and 4 bit weight configured MAC operation where the output code is valid in 7.5ns. It is worth to note that 4-bit wise serial operation maintains consistent system operation with different bit-width configuration.

C. UNION SRAM: PROCESS AND TEMPERATURE VARIATIONS AUTO-COMPENSATION

Recent reasearches [18], [19] have proposed both hardware and software level approaches to calibrate the adverse effect of PVT variations in analog CIM, which increased hardware and software overhead for the system control. In auto-compensation scheme proposed in our work, there is no extra hardware or software overhead, making overall system simple.

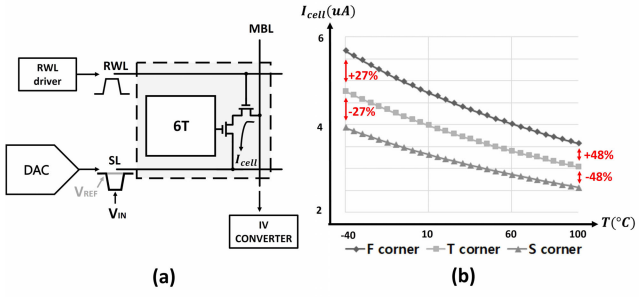


FIGURE 6. (a) Normal implementation of multiplication operation within current-based 8T SRAM cell, (b) bit-cell current variation in the presence of process corner shift and temperature variation.

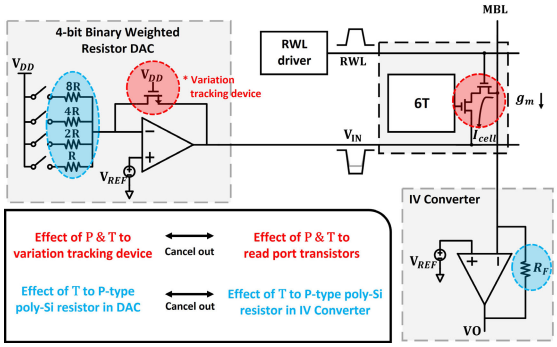


FIGURE 7. PVT variation auto compensated configuration of input DAC and IV converter along with the 8T SRAM cell.

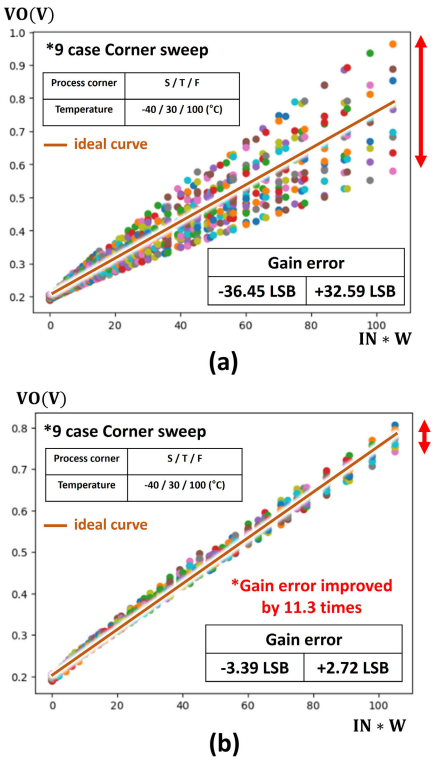


FIGURE 8. Process and temperature swept 9 corner simulation results of (a) MAC operation linearity curve without auto-compensation (b) MAC operation linearity curve with auto-compensation.

Fig. 6(a) shows the key components for single MAC operation of the proposed current-based 8T SRAM MAC opeation, with DAC connected to SLs, and MBL connected to the input of IV converter. Basically, the bit cell current (I_{cell}) that represents the multiplication of input and weight can be expressed as follows.

$$I_{cell} = g_m * V_{IN}. \tag{3}$$

where g_m denotes the transconductance of read port of 8T SRAM cell operating in deep triode region, and V_{IN} denotes the voltage fed into the SLs. Here, one of the critical issue with current-based SRAM CIM appears, the effect of process variation (P) and temperature variation (T) to the transconductance of this read port. As can be seen in Fig. 6(b), current flowing across the transistors fluctuates from 27 percent to 48 percent in absolute value with temperature variation and process shift under TSMC 65nm chip fabrication environment. Taking these environmental parameter into consideration, the bit-cell current can be expressed as the formula below.

$$I_{cell} = (g_m * f(P) * g(T)) * V_{IN}. \tag{4}$$

where $f(P)$ and $g(T)$ denote bitcell transcanductance variation due to process shift and temperature variation, respectively. To remove these variabilities, we implemented input DAC and IV converter as Fig. 7.

Input DAC has been configured by a common binary weighted resistor DAC, with feedback resistor replaced by a transistor (variation tracking device) operating in deep triode region. With this DAC configuration, the bit cell current equation can be modified as follows

$$I_{cell} = (g_m * f(P) * g(T)) * (V_{IN} * f^{-1}(P) * g^{-1}(T)). \tag{5}$$

resulting in

$$I_{cell} = g_m * V_{IN}. \tag{6}$$

Because the variation tracking device in the input DAC is also influenced by P and T, V_{in} level is also shifted, but with the opposite tendency. For example, if P and T scales down the transconductance of the nmos transistors, then the magnitude of V_{IN} is scaled up with the same rate since the effective resistance across this device is increased proportionally, cancelling out the distortions due to P and T. Furthermore, P-type poly-Si resistors placed in DAC and IV converter are also affected by temperature variation. But, in the same manner, the effects of T in the P-poly resistors in DAC and IV converter are cancelled out.

Fig. 8(a) and (b) show process and temperature swept 9 corner simulation results that represents MAC operation linearity curves on output voltage node of IV converter (VO). Fig. 8(a) shows the case when the voltage generated by input DAC and the feedback resistor R_F on the IV

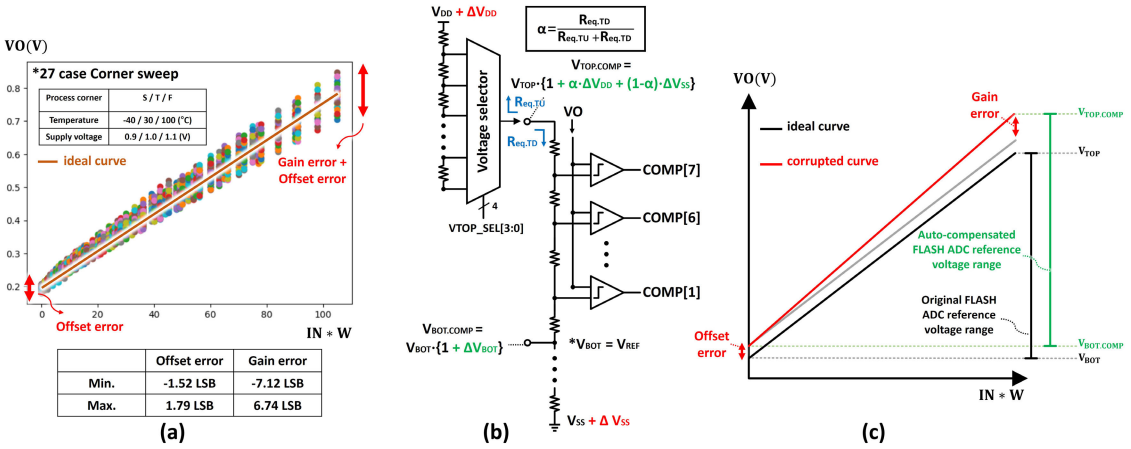


FIGURE 9. **(a) Simulation results of the MAC operation linearity curve under 27 PVT corner sweeps, incorporating the proposed auto-compensation configuration within the input DAC, 8T SRAM, and IV converter, (b) Design of 3-bit FLASH ADC with V_{DD} and V_{SS} variations auto-tracking reference voltage resistor ladder and (c) V_{DD} and V_{SS} variations auto-compensated FLASH ADC reference voltage range.**

converter are not variant with process and temperature variation, and Fig. 8(b) shows the case where the proposed auto-compensation structure is implemented. As can be seen on Fig. 8(a), the MAC operation curve is shifted significantly depending on the process shift and temperature variation, but Fig. 8(b) shows that our design ensures fully-compensated MAC operation linearity curve. With this auto-compensation scheme, we could achieve 11.3 times enhanced gain error, where the gain error denotes the difference between last points of real MAC operation curve and ideal MAC operation curve.

Besides, the choice to implement a binary-weighted resistor DAC over other commonly used, power and area efficient DAC structures, such as the R-2R DAC, is driven by three key factors:

- In terms of unit resistor count, a 4-bit binary-weighted resistor DAC requires 15 unit resistors, while R-2R DAC needs 13 unit resistors. The difference in the number of unit resistors is minimal (only 2 additional resistors) due to the low bit resolution by 4 bit.
- The binary-weighted DAC necessitates only a single analog switch per resistor branch, connecting each branch to the V_{DD} signal. In contrast, the R-2R DAC requires two analog switches per branch to toggle between V_{REF} and V_{DD} signals. This doubles the number of analog switches, worsening the area efficiency.
- The binary-weighted resistor DAC requires a simple V_{REF} signal source to drive only the capacitive load (gate of OP AMP). However, the R-2R DAC demands a more complex V_{REF} source to drive both the capacitive load and the resistive network formed by the R-2R branches. This imposes additional design constraints, increasing the complexity of the V_{REF} signal source and the associated power consumption.

For these three considerations, the 4-bit binary-weighted resistor DAC was selected for the energy and area efficient design.

D. SUPPLY VOLTAGE (V_{DD}) AND GROUND (V_{SS}) VARIATIONS AUTO-COMPENSATION

Although the process and temperature variations are effectively compensated through the proposed UNION SRAM structure, still the supply voltage (V_{DD}) and ground (V_{SS}) variations need to be considered which may cause the distortions in the MAC operation linearity. As discussed in section II.C, the transconductance distortion in the bitcell transistors due to the V_{DD} variation is compensated through the variation tracking device in 4-bit Binary Weighted Resistor DAC. For instance, if the V_{DD} slightly increases due to supply noise, then the transconductance of the bitcell increases proportionally. Meanwhile, the effective resistance of the variation tracking device also gets bigger at the same rate, resulting in the compensated bitcell current.

However, the variation factor introduced by the V_{DD} signal, which is connected to the binary weighted resistors via analog switches, remains uncompensated. Furthermore, V_{REF} is shared with V_{BOT} of the FLASH ADC reference voltages, as depicted in Fig. 9(b). This voltage is also susceptible to distortions due to variations in V_{DD} and V_{SS}. Consequently, these two uncompensated voltage variation factors induce distortions in the linearity curve of the MAC operation.

Figure 9(a) presents the simulation outcomes across 27 PVT corners, considering 3 distinct variations per each corner. These results are obtained with the implementation of the proposed auto-compensation technique, which effectively mitigates the impact of process and temperature fluctuations, but the V_{DD} and V_{SS} variations are not fully compensated. The unresolved V_{DD} variations in the input DAC stage

lead to increased gain errors in the MAC linearity curve. Additionally, the distorted V_{REF}, which determines the initial point of the MAC operation range, introduces further offset errors in the MAC linearity curve. In summary, the distortion rate in the MAC operation range due to voltage variations can be formulated as follows.

$$\Delta MAC = \frac{\Delta V_{DD} - \Delta V_{BOT}}{V_{DD} - V_{BOT}}, \tag{7}$$

where ΔV_{DD} and ΔV_{BOT} are resulted due to the connection of distorted V_{DD} and V_{REF} (=V_{BOT}) signals to the input DAC stage as illustrated in Fig. 7.

However, when these corrupted MAC voltage values (V_O) are delivered to the FLASH ADC stage via DSBMVP, the voltage variations are effectively compensated through the auto-modulation of the FLASH ADC's reference voltage range. Fig. 9(b) illustrates the structure of the 3-bit FLASH ADC. V_{TOP} voltage is dynamically selected between 0.4 V and 1 V, generated by a resistor ladder through a 4-bit voltage selector, while V_{BOT} (equivalent to V_{REF}) is fixed at 0.2 V. Considering the variations in V_{DD} and V_{SS}, the distortion rate in the FLASH ADC reference voltage range is expressed by the following formula.

$$\Delta V_{REF_ADC} = \frac{\Delta V_{TOP} - \Delta V_{BOT}}{V_{TOP} - V_{BOT}} \tag{8}$$

By substituting ΔV_{TOP} with ΔV_{DD}, the equation (8) can be reformulated as follows, where α represents the resistive division at the top node of the resistor ladder, as depicted in Fig. 9(b).

$$\Delta V_{REF_ADC} = \frac{(\alpha \cdot \Delta V_{DD} + (1-\alpha) \cdot \Delta V_{SS}) - \Delta V_{BOT}}{\alpha \cdot V_{DD} - V_{BOT}} \tag{9}$$

$$\alpha = \frac{R_{eq.TD}}{R_{eq.TU} + R_{eq.TD}} \tag{10}$$

If the resistance seen at the top reference node towards ground potential ($R_{eq.TD}$) is much larger than the resistance towards V_{DD} potential ($R_{eq.TU}$), then α approaches to the value of 1. Consequently, the equation (9) can be simplified as follows, which is identical to the distortion rate in the MAC operation range expressed in equation (7).

$$\Delta V_{REF_ADC} \approx \frac{\Delta V_{DD} - \Delta V_{BOT}}{V_{DD} - V_{BOT}} \tag{11}$$

If this condition is met, as the MAC voltage values (VO) are distorted due to the V_{DD} and V_{SS} variations, V_{TOP} and V_{BOT} are also shifted almost at the same rate. Consequently, as illustrated in Fig. 9(c), the FLASH ADC reference voltage range (from $V_{BOT.COMP}$ to $V_{TOP.COMP}$) is auto adjusted to align with the distorted range of the corrupted curve (VO), resulting in voltage (V_{DD} and V_{SS}) variations auto-compensated digital output codes.

In our investigation considering practical DNN application scenarios, V_{TOP} is nominally set to 700 mV. In this case, $R_{eq.TD}$ is 2.33 times larger than $R_{eq.TU}$, which is sufficient to effectively implement the voltage variation auto-compensation mechanism.

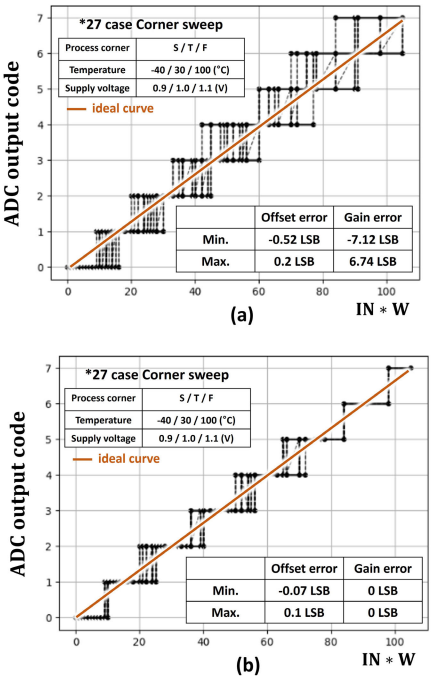


FIGURE 10. Transfer curve of the 3-bit FLASH ADC based on the 27 corner simulations (a) without ADC reference voltage range auto-adjustment and (b) with ADC reference voltage range auto-adjustment.

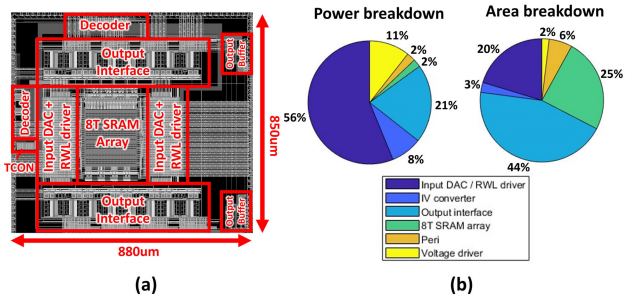


FIGURE 11. (a) Top layout of designed union SRAM based CIM architecture and (b) power and area breakdown of the overall architecture.

To reflect practical DNN application scenarios, V_{TOP} was set to 700mV, and 27 corner simulations were conducted. Fig. 10(a) shows the transfer curve of the 3-bit FLASH ADC under these conditions, where V_{TOP} and V_{BOT} are invariant with voltage (V_{DD} and V_{SS}) variations. Note that start-to-end point of the transfer curves exhibit severe distortions mainly due to not compensated voltage variations. On the other hand, Fig. 10(b) shows the transfer curve of the 3-bit FLASH ADC with the incorporation of the auto-adjustment of FLASH ADC reference voltage range to the voltage variations. The resulting transfer curve exhibits significantly improved offset and gain errors after the digitization process of the ADC operation.

IV. HARDWARE PERFORMANCE ANALYSIS

Fig. 11(a) shows the top layout of our 8T SRAM MAC macro designed with TSMC 65nm GP process. The total area of our

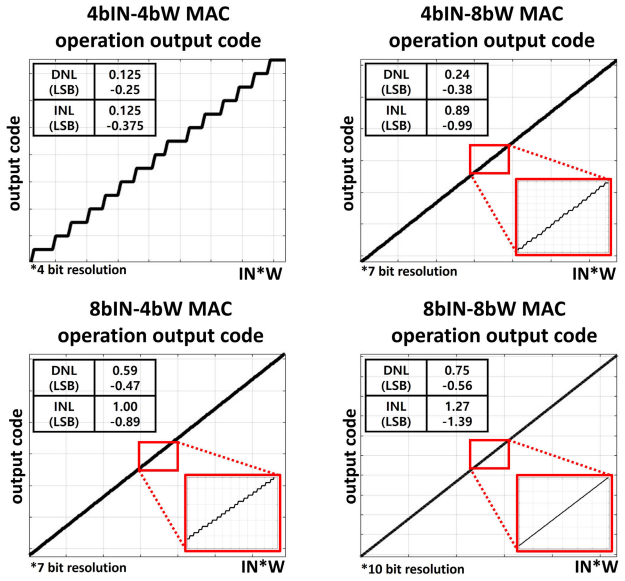


FIGURE 12. Output code of the overall system depending on different bit precision configurations.

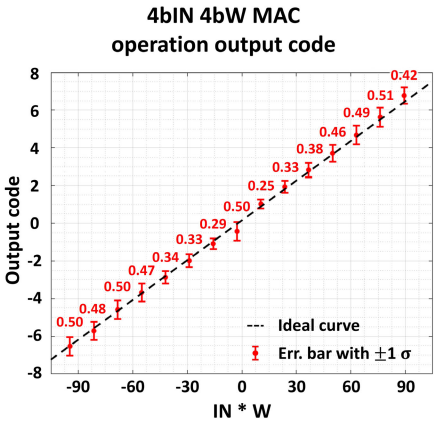


FIGURE 13. 500 Monte-Carlo simulation results of the overall system with 4bIN/4bW configuration.

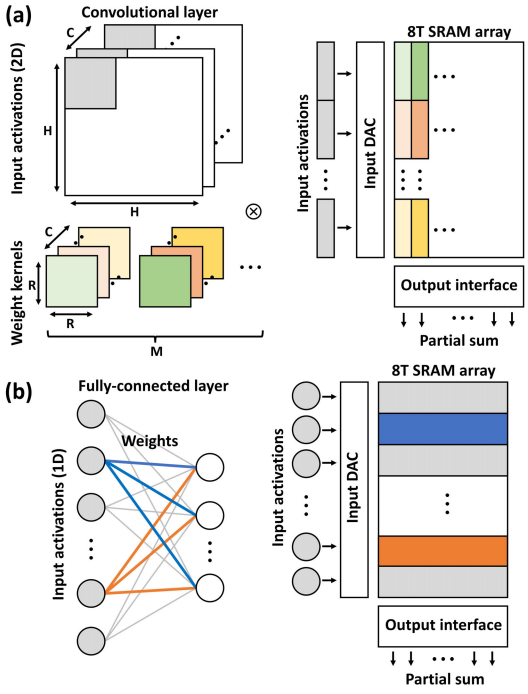


FIGURE 14. (a) Illustration of mapping convolutional layer weight kernels to UNION SRAM MAC macro and (b) illustration of mapping fully-connected layer weight matrix to UNION SRAM MAC macro.

chip is 0.748 mm^2. The total power consumption of the entire system is 22.89mW based on post layout simulation.

Fig. 11(b) shows the breakdowns of the power and area of the designed chip. Since it is important to provide sufficient driving strength to SLs of 8T SRAM arrays for accurate MAC operation, input DACs consume significant portion of power consumption. Furthermore, the array peripherals (input DAC + RWL driver, output interfaces including ADC) take up a large portion of the area.

Fig. 12 shows the output code from the entire system with different bit precision, differing from 4bit to 10bit. Note that Differential Non-linearity (DNL) error and Integral Non-linearity (INL) error increase due to the amplification of the quantization errors with the bit-shift operations at the higher bit precision. To analyze the impact of FLASH ADC-induced offset errors, a Monte Carlo simulation with 500 iterations was performed using a 4-bit input and weight configuration. As shown in Fig. 13, the maximum observed standard deviation was 0.51 LSB. To assess the impact of these errors on classification accuracy, random offset errors with a 0.51 LSB standard deviation were injected into the MAC outputs of all layers in the ResNet-18 model. Testing on CIFAR-10 resulted in 94.96 percent accuracy, exhibiting only 0.06 percent drop from the 4-bit baseline model accuracy of 95.02 percent.

To specifically analyze the performance of the proposed UNION SRAM macro in the practical DNN application, we built software level IMC framework that effectively maps the weight kernels to the MAC macro tiles that reflect the hardware properties (e.g. quantization levels), as can be seen on the Fig. 14(a) and (b). For convolutional layers, both the input slices and the weight kernels are segmented and unrolled into UNION SRAM macros, as illustrated in Fig. 14(a). For fully-connected layers, the input vectors and weight matrices are divided and directly mapped to the UNION SRAM macros, as can be seen on Fig. 14(b). The partial sum outputs from each macro are subsequently accumulated and delivered to the next layer's input activations.

As can be seen on table 1, the IMC framework was applied to 4-bit ResNet-18 model for CIFAR-10 dataset classification, 95.02 percent of the accuracy has been achieved. Note that the FP32 baseline accuracy is 95.17 percent, demonstrating that the IMC based model approaches the accuracy performance of the software baseline. To further evaluate

TABLE 1. CIFAR-10 dataset classification accuracy RESNET-18 model.

	S/W baseline	IMC based			
Bit-precision	FP 32	INT 4			
Condition	-	Nominal	ff/1.1V/-30°C	ss/0.9V/90°C	
Accuracy (%)	95.17	95.02	w.o. comp	w. comp	w.o. comp	w. comp
			78.59 (-16.43)	94.78 (-0.24)	85.84 (-9.18)	94.13 (-0.89)

TABLE 2. Comparison table.

	JSSC'21 [15]	TCASI'23 [25]	ACCESS'24 [10]	This work			
Technology Node	65nm	65nm	55nm	65nm			
Measured / Simulated	Measured	Simulated	Measured	Simulated			
Supply Voltage	1.2 V	0.8 – 1.1 V	1.2 V	1.0 V			
CIM mode	Charge	Digital	Current	Current			
IN/W bit precision	4 / 1-5, 8	1-16 / 1-16	4 / 5	4 / 4	4 / 8	8 / 4	8 / 8
Output bit precision	7	9-24	5	4	7	7	10
Throughput (GOPs/8Kb)	71.675 (W:2b)	204.8 (1b/1b) 12.8 (1b/16b)	NA	793.4 (4b/4b) 170 (8b/8b)			
Energy Efficiency (TOPs/W)	49.4 (W:1b)	315.07 (1b/1b) 1.23 (16b/16b)	7.3	23.76 (4b/4b) 5.1 (8b/8b)			
Area Efficiency (TOPs/mm^2)	3.4 (W:2b)	2.5 (1b/1b) 0.027 (16b/16b)	0.07	1.06 (4b/4b) 0.227 (8b/8b)			
CIFAR-10 accuracy (%)	89*	NA	92.28*	95.02**			

* Deployed on hardware
** Simulated w/ hardware properties

the effectiveness of the proposed PVT auto-compensation scheme in mitigating accuracy degradation, we introduced estimated gain and offset errors into the MAC computation layers, including both convolutional and fully connected layers of the ResNet-18 model. Without the PVT variation compensation, the classification accuracy is severely degraded by 16.43 percent at ff/1.1V/-30°C condition. Meanwhile, when the proposed PVT auto-compensation scheme is applied, the classification drop is effectively prevented below 0.89 percent, highlighting the robustness of the scheme in maintaining model performance under adverse PVT conditions.

V. PERFORMANCE COMPARISON

Table 2 shows the performance comparison with recent SRAM CIM works. Analog domain SRAM based CIM architecture (charge mode and current mode) research works typically exhibit poor bit precision configurability. But they exhibit higher throughput compared to digital domain counter parts. On the contrary, digital domain SRAM based CIM architectures exhibit high bit-precision configurability from 1 to 16 bit. However, when compared to analog domain counterparts, digital domain CIM architectures exhibit lower throughput. It is also worth to note that analog domain CIM architectures are prone to PVT variations, where digital domain CIM architectures are not for its bit wise computing scheme.

Unlike other recent analog domain CIM works which mostly support fixed bit-precision, our work supports dynamic bit-precision for input (4b/8b), weight (4b/8b) and output (4b/7b/7b/10b), while achieving highest throughput among the other analog and digital domain CIM works. Consequently, we proved the efficacy of our analog domain SRAM based CIM architecture with throughput ranging from 170 to 793.4 GOPs, area efficiency ranging from 0.227 to 1.06 TOPs/mm^2 and energy efficiency ranging from 5.1 to 23.76 TOPs/W with different bit precision configurations.

VI. CONCLUSION

In this work, we demonstrated current accumulation based bit precision configurable 8T SRAM MAC macro for DNN acceleration application. To resolve the unflexible bit-precision configuration issue inherent in analog CIM architecture, we enabled 4/8 bit IN/W configurable design. Furthermore, we demonstrated an unified structure of SRAM MAC macro that fully compensates the MAC computing accuracy degradation due to the systematic PVT variations. The post layout simulation resulted in throughput of 170 - 793.4 GOPs, area efficiency of 0.227 to 1.06 TOPs/mm^2 and energy efficiency of 5.1 - 23.76 TOPs/W, with different bit precision configuration of input (4/8b), weight (4/8b) and output (4b/7b/7b/10b). Furthermore, we proved the efficacy of designed UNION SRAM based CIM architecture with software level simulation, with CIFAR-10 dataset classification accuracy of 95.02 percent using RESNET-18 model.

REFERENCES

[1] M. Horowitz, "1.1 computing's energy problem (and what we can do about it)," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, Feb. 2014, pp. 10–14.
[2] T. N. Theis and H.-S. P. Wong, "The end of Moore's law: A new beginning for information technology," *Comput. Sci. Eng.*, vol. 19, no. 2, pp. 41–50, Mar. 2017.
[3] A. Jaiswal, I. Chakraborty, A. Agrawal, and K. Roy, "8T SRAM cell as a multibit dot-product engine for beyond von Neumann computing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 11, pp. 2556–2567, Nov. 2019.
[4] 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 J. Solid-State Circuits*, vol. 55, no. 1, pp. 189–202, Jan. 2020.
[5] M. E. Sinangil, B. Erbagci, R. Naous, K. Akarvardar, D. Sun, W.-S. Khwa, H.-J. Liao, Y. Wang, and J. Chang, "A 7-nm compute-in-memory SRAM macro supporting multi-bit input, weight and output and achieving 351 TOPS/W and 372.4 GOPS," *IEEE J. Solid-State Circuits*, vol. 56, no. 1, pp. 188–198, Jan. 2021.
[6] Z. Lin, H. Zhan, Z. Chen, C. Peng, X. Wu, W. Lu, Q. Zhao, X. Li, and J. Chen, "Cascade current mirror to improve linearity and consistency in SRAM in-memory computing," *IEEE J. Solid-State Circuits*, vol. 56, no. 8, pp. 2550–2562, Aug. 2021.
[7] V. T. Nguyen, J.-S. Kim, and J.-W. Lee, "10T SRAM computing-in-memory macros for binary and multibit MAC operation of DNN edge processors," *IEEE Access*, vol. 9, pp. 71262–71276, 2021.
[8] A. Kneip, M. Lefebvre, J. Verecken, and D. Bol, "IMPACT: A 1-to-4b 813-TOPS/W 22-nm FD-SOI compute-in-memory CNN accelerator featuring a 4.2-POPS/W 146-TOPS/mm^2 CIM-SRAM with multi-bit analog batch-normalization," *IEEE J. Solid-State Circuits*, vol. 58, no. 7, pp. 1871–1884, Jul. 2023.
[9] 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 J. Solid-State Circuits*, vol. 54, no. 1, pp. 217–230, Jan. 2019.

[10] Z. Gu, S. Dou, H. You, Y. Zhan, S. Qiao, and Y. Zhou, "A dual-wordline 6T SRAM computing-in-memory macro featuring full signed multi-bit computation for lightweight networks," *IEEE Access*, vol. 12, pp. 35195–35203, 2024.
[11] 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 J. Solid-State Circuits*, vol. 55, no. 7, pp. 1888–1897, Jul. 2020.
[12] V. Sharma, J.-E. Kim, H. Kim, L. Lu, and T. T. Kim, "A reconfigurable 16Kb AND8T SRAM macro with improved linearity for multibit compute-in memory of artificial intelligence edge devices," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 12, no. 2, pp. 522–535, Jun. 2022.
[13] 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 J. Solid-State Circuits*, vol. 54, no. 6, pp. 1789–1799, Jun. 2019.
[14] H. Jia, H. Valavi, Y. Tang, J. Zhang, and N. Verma, "A programmable heterogeneous microprocessor based on bit-scalable in-memory computing," *IEEE J. Solid-State Circuits*, vol. 55, no. 9, pp. 2609–2621, Sep. 2020.
[15] Z. Chen, Z. Yu, Q. Jin, Y. He, J. Wang, S. Lin, D. Li, Y. Wang, and K. Yang, "CAP-RAM: A charge-domain in-memory computing 6T-SRAM for accurate and precision-programmable CNN inference," *IEEE J. Solid-State Circuits*, vol. 56, no. 6, pp. 1924–1935, Jun. 2021.
[16] B. Zhang, S. Yin, M. Kim, J. Saikia, S. Kwon, S. Myung, H. Kim, S. J. Kim, J.-S. Seo, and M. Seok, "PIMCA: A programmable in-memory computing accelerator for energy-efficient DNN inference," *IEEE J. Solid-State Circuits*, vol. 58, no. 5, pp. 1436–1449, May 2023.
[17] H. Zhang, S. He, X. Lu, X. Guo, S. Wang, Y. Du, and L. Du, "SSM-CIM: An efficient CIM macro featuring single-step multi-bit MAC computation for CNN edge inference," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 70, no. 11, pp. 4357–4368, Nov. 2023.
[18] S.-E. Hsieh, C.-H. Wei, C.-X. Xue, H.-W. Lin, W.-H. Tu, E.-J. Chang, K.-T. Yang, P.-H. Chen, W.-N. Liao, L. L. Low, C.-D. Lee, A.-C. Lu, J. Liang, C.-C. Cheng, and T.-H. Kang, "7.6 A 70.85–86.27TOPS/W PVT-insensitive 8b word-wise ACIM with post-processing relaxation," in *Proc. IEEE Int. Solid- State Circuits Conf. (ISSCC)*, Feb. 2023, pp. 136–138.
[19] A. Kneip, M. Lefebvre, J. Verecken, and D. Bol, "A 1-to-4b 16.8-POPS/W 473-TOPS/mm^2 6T-based in-memory computing SRAM in 22nm FD-SOI with multi-bit analog batch-normalization," in *Proc. IEEE 48th Eur. Solid State Circuits Conf. (ESSCIRC)*, Sep. 2022, pp. 157–160.
[20] 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 Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 11, pp. 4172–4185, Nov. 2019.
[21] Y.-C. Chiu, Z. Zhang, J.-J. Chen, X. Si, R. Liu, Y.-N. Tu, J.-W. Su, W.-H. Huang, J.-H. Wang, W.-C. Wei, J.-M. Hung, S.-S. Sheu, S.-H. Li, C.-I. Wu, R.-S. Liu, C.-C. Hsieh, K.-T. Tang, and M.-F. Chang, "A 4-Kb 1-to-8-bit configurable 6T SRAM-based computation-in-memory unit-macro for CNN-based AI edge processors," *IEEE J. Solid-State Circuits*, vol. 55, no. 10, pp. 2790–2801, Oct. 2020.
[22] X. Si et al., "A local computing cell and 6T SRAM-based computing-in-memory macro with 8-b MAC operation for edge AI chips," *IEEE J. Solid-State Circuits*, vol. 56, no. 9, pp. 2817–2831, Sep. 2021.
[23] H. Kim, T. Yoo, T. T. Kim, and B. Kim, "Colonnade: A reconfigurable SRAM-based digital bit-serial compute-in-memory macro for processing neural networks," *IEEE J. Solid-State Circuits*, vol. 56, no. 7, pp. 2221–2233, Jul. 2021.
[24] H. Fujiwara, H. Mori, W.-C. Zhao, M.-C. Chuang, R. Naous, C.-K. Chuang, T. Hashizume, D. Sun, C.-F. Lee, K. Akarvardar, S. Adham, T.-L. Chou, M. E. Sinangil, Y. Wang, Y.-D. Chih, Y.-H. Chen, H.-J. Liao, and T. J. Chang, "A 5-nm 254-TOPS/W 221-TOPS/mm^2 fully-digital computing-in-memory macro supporting wide-range dynamic-voltage-frequency scaling and simultaneous MAC and write operations," in *Proc. IEEE Int. Solid- State Circuits Conf. (ISSCC)*, vol. 65, Feb. 2022, pp. 1–3.
[25] H. Kim, J. Mu, C. Yu, T. T. Kim, and B. Kim, "A 1–16b reconfigurable 80Kb 7T SRAM-based digital near-memory computing macro for processing neural networks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 70, no. 4, pp. 1580–1590, Apr. 2023.
[26] L. Chang, R. K. Montoye, Y. Nakamura, K. A. Batson, R. J. Eickemeyer, R. H. Dennard, W. Haensch, and D. Jamsek, "An 8T-SRAM for variability tolerance and low-voltage operation in high-performance caches," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 956–963, Apr. 2008.

HONGGU KIM received the B.S. degree from the School of Electrical and Electronics Engineering, Chung-Ang University (CAU), Seoul, South Korea, in 2022. He is currently pursuing the integrated M.S. and Ph.D. degree in intelligent semiconductor engineering. His research interests include neuromorphic hardware and software co-optimization and analog compute-in-memory architecture.

YERIM AN received the B.S. degree from the School of Electrical and Electronics Engineering, Tech University of Korea (TUKOREA), South Korea, in 2022, and the M.S. degree in intelligent semiconductor engineering from Chung-Ang University (CAU), Seoul, South Korea, in 2024, where she is currently pursuing the Ph.D. degree in intelligent semiconductor engineering. Her research interest includes process-in-memory (PIM) architecture design.

RYUNYEONG KIM received the B.S. degree from the School of Electrical and Electronic Engineering, Chung-Ang University (CAU), Seoul, South Korea, in 2023, where he is currently pursuing the M.S. degree in intelligent semiconductor engineering. His research interest includes SRAM-based hybrid domain process-in-memory (PIM).

SUNYOUNG KIM received the B.S. degree from the School of Electronics Engineering, Kangwon National University (KNU), South Korea, in 2023. She is currently pursuing the M.S. degree in intelligent semiconductor engineering with Chung-Ang University (CAU). Her research interest includes process-in-memory (PIM) architecture design.

YONG SHIM received the B.S. and M.S. degrees in electronics engineering from Korea University, in 2004 and 2006, respectively, and the Ph.D. degree from the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, in 2018. He was a Memory Interface Designer at Samsung Electronics, Hwaseong, from 2006 to 2013. At Samsung, he has worked on the design and development of a memory interface for synchronous DRAMs (DDR1 and DDR4). He is currently working as an Assistant Professor with Chung-Ang University. Prior to joining Chung-Ang University, in 2020, he was an SRAM Designer at Intel Corporation, Hillsboro, OR, from 2018 to 2020, where he was involved in designing circuits for super-scaled next-generation SRAM cache design. His research interests include neuromorphic hardware and algorithm, in-memory computing, robust memory interface design, and emerging devices (RRAM, MRAM, and STO) based unconventional computing models.