IEEE Journal on Exploratory Solid-State Computational Devices and Circuits

Received 19 December 2022; revised 3 February 2023 and 7 March 2023; accepted 8 March 2023.
Date of publication 14 March 2023; date of current version 29 May 2023.

Digital Object Identifier 10.1109/JXCDC.2023.3255788

A High-Parallelism RRAM-Based Compute-In-Memory Macro With Intrinsic Impedance Boosting and In-ADC Computing

TIAN XIE[1] (Graduate Student Member, IEEE), SHIMENG YU[2] (Senior Member, IEEE), and SHAOLAN LI[1] (Member, IEEE)

[1]Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA
[2]Georgia Institute of Technology, Atlanta, GA 30332 USA

CORRESPONDING AUTHOR: T. XIE (txie65@gatech.edu)

This article has supplementary downloadable material available at https://doi.org/10.1109/JXCDC.2023.3255788, provided by the authors.

ABSTRACT Resistive random access memory (RRAM) is considered to be a promising compute-in-memory (CIM) platform; however, they tend to lose energy efficiency quickly in high-throughput and high-resolution cases. Instead of using access transistors as switches, this work explores their analog characteristics as common-gate current buffers. So the cell current can be minimized and the output impedance is boosted. The idea of In-ADC Computing (IAC) is also proposed to further decrease the complexity of the peripheral circuits. Benefiting from the proposed ideas, a pretrained VGG-8 network based on the CIFAR-10 dataset can be implemented, and an accuracy of 87.2% is achieved with 8.9 TOPS/W energy efficiency (for 8-bit multiply-and-accumulate (MAC) operation), demonstrating that the proposed techniques enable low-distortion partial sum results while still being able to operate in a power-efficient way.

INDEX TERMS High parallelism, In-ADC Computing (IAC), in-memory computing, intrinsic impedance boosting (IIB), resistive random access memory (RRAM).

I. INTRODUCTION

COMPUTE-IN-MEMORY (CIM) is a fast-rising paradigm of neural network hardware accelerators for their potential of overcoming the "memory wall" bottleneck [1], [2], [3], [4]. Among them, resistive random access memory (RRAM)-based CIM has deemed a promising direction, thanks to several advantages [5]. Compared to other eDRAM or SRAM CIM counterparts, the non-volatile characteristic of RRAM allows fast and low-power systems to wake up. The wide programmable range and inherent voltage–current (V–I) conversion of the RRAM cell naturally facilitate parallel multiply-and-accumulate (MAC) operation, which improves the computation throughput [6]. Moreover, the RRAM cell is highly compact and can be designed in high density, indicating a low-cost production [7], [8].

On the other hand, due to its intense analog nature, it is highly nontrivial for RRAM-CIM to implement large deep neural network (DNN) models, especially with both high parallelism and high energy efficiency. The critical bottleneck lies in the design of the "read circuits" (RCs), which typically consists of the RRAM interface circuit, multiplexer, and the analog-to-digital converter (ADC), as shown in Fig. 1. There are several stringent requirements that the RC needs to address to unlock higher computing capability. First, it needs to handle large subarray output current variation with little distortion and provide relatively fine quantization to preserve computation accuracy [9], [10]. Second, it must be compact in area, ideally matching the RRAM subarray column pitch, as full "column-parallel" operation is essential for achieving high throughput. Last but not least, the RC should maintain low power and low read latency, so as to ensure RRAM's suitability for edge computing. These requirements are tightly coupled with the fundamental tradeoff in analog circuit design and, thus, highly challenging to optimize simultaneously.

Recently reported RRAM-based CIM works have shown a growing focus on the RC design to address these limitations. Xue et al. [11] leverage a current-mode ADC and a current-mirror network for pre-ADC signal processing, which helps reduce the number of read and ADC operations. However, the large offset nature of current mode circuits restricts the

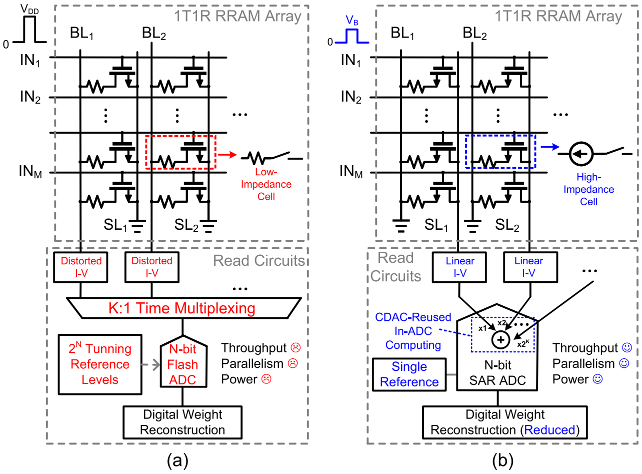


FIGURE 1. Comparison between (a) conventional RRAM-based CIM macro and (b) proposed RRAM CIM macro.

parallelism to only nine rows. In addition, the RC consumes a large area and requires 32:1 time multiplexing to share among columns, which significantly limits the throughput. Yin et al. [12] and He et al. [13] demonstrate a read scheme using a simple voltage-divider-based network with voltage-mode flash ADC that can reduce the multiplex ratio. These designs, nevertheless, suffer from large read power issues and are prone to read disturbance. Input-aware current control [10] and sparsity-aware clamping [14] are later developed to improve the voltage-mode read performance, but they come at the expense of reduced parallelism. In another work [15], a single-slope ADC is directly used as the RC through its embedded V–I conversion. While it achieves good parallelism with high A/D resolution, the throughput and chip area are severely undermined. In addition to the above limitations, most existing RRAM-CIM macros also share a common drawback that the analog MAC results suffer from considerable distortion. To compensate for this, they rely on externally generated and calibrated ADC reference levels, which are impractical for compact and energy-constraint applications, such as sensor nodes. Therefore, further research is expected to solve the tradeoffs between power, speed, and area.

In this article, we present an innovative RRAM-based CIM macro that unifies *accuracy, compactness, and energy efficiency*. We propose an intrinsic impedance boosting (IIB) technique, which exploits the access switch's analog property and turns it into a common-gate (CG) current buffer. This technique enables accessing a large number of rows and computing large MAC values with little distortion using simple interface circuits. The idea of the In-ADC Computing (IAC) technique is also proposed, which reuses the successive-approximation-register (SAR) ADC's capacitor to rebuild multibit-weight MAC results in the charge domain within the sampling process. This not only effectively reduces both the total A/D conversions number and digital shit-and-add overhead but also achieves column parallel A/D without time multiplexing and an extremely compact layout. Fig. 1 provides a high-level comparison between the conventional RRAM CIM macro with the proposed one. With the proposed techniques, an accuracy of 87.2% is achieved based on the VGG-8 network for CIFAR-10 applications. The simulated energy efficiency is 8.9 TOPS/W (for 8 × 8 bit MAC), and the throughput is 256 GOPS, which indicates that the proposed techniques enable low-distortion, high-parallelism, and power-efficient RRAM-based CIM macro.

The rest of this article is organized as follows. Section II provides background. In Section III, the idea of RRAM IIB is introduced, and its design considerations are analyzed. In Section IV, the idea of IAC is proposed. Section V presents the system-level simulation results, and Section VI concludes this article.

II. RRAM-BASED CIM BACKGROUNDS

The core idea behind CIM is to leverage a crossbar structure to conduct massive parallel MAC operations. In the context of RRAM-CIM, the weights are represented by the resistance/conductance of the crossbar cells. Ideally, the inputs can be applied across the resistors, and the generated currents represent the computation result. Such implementation needs only resistors and can support input and weight with arbitrary resolution. Nonetheless, to avoid the sneak-path current issue and maintain better retention/reliability, practical RRAM-CIM macros are commonly implemented using a 1T1R array with binary cellwise computation [10], [11], [12], [13], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], as illustrated in Fig. 2(a). In this scheme, RRAM resistances are programmed between only two states, where the high-resistance state (HRS) represents logic “0” and the low-resistance state (LRS) represents logic “1”. Binary wordline (WL) voltages serve as inputs to turn on/off the access transistor. Only when RRAM is in LRS and the access transistor is turned on, a substantial cell current can be generated, which maps the binary multiplication. To implement multibit MAC under this scheme, the inputs and weights need to be decomposed to perform bitwise multiplications, as depicted in Fig. 2(b). The input bits follow a bit-serial manner, applied one by one to the WL sequentially, while the weight bits are parallelly programmed across adjacent columns (and unweighted). The complete MAC results are rebuilt via weighted-summing the partial results across the columns and cycles, which is typically done using digital shift-and-add (S&A) after the A/D.

On the other hand, despite providing good robustness, the binary computing RRAM scheme is subject to several shortcomings in terms of energy efficiency. The first issue is the large number of ADC operations required. It can be seen that the number of ADC firings grows proportionally to both the input and weight resolution. For example, if 4-bit inputs and 4-bit weights are utilized, a full MAC result requires 16 ADC firings. The second challenge lies in the power consumption of the RRAM array itself and the I–V interface. The low LRS resistance of RRAM is typically in the range of a few kΩ. With high parallelism (i.e., the number of simultaneously accessed rows), the output current can be as large as tens

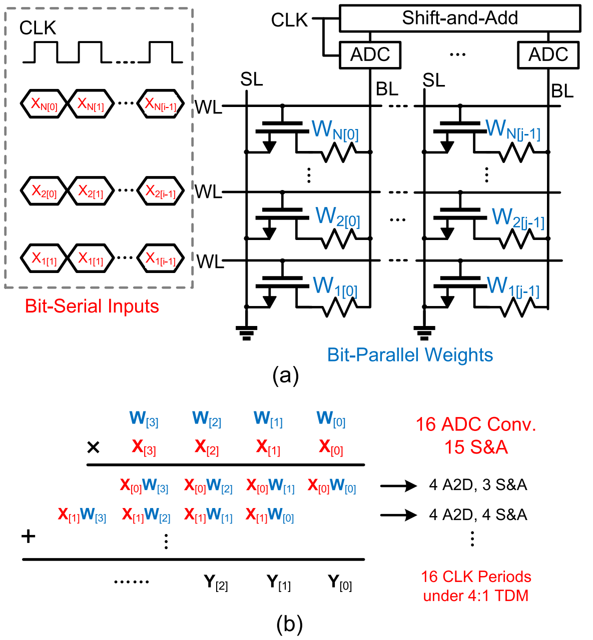
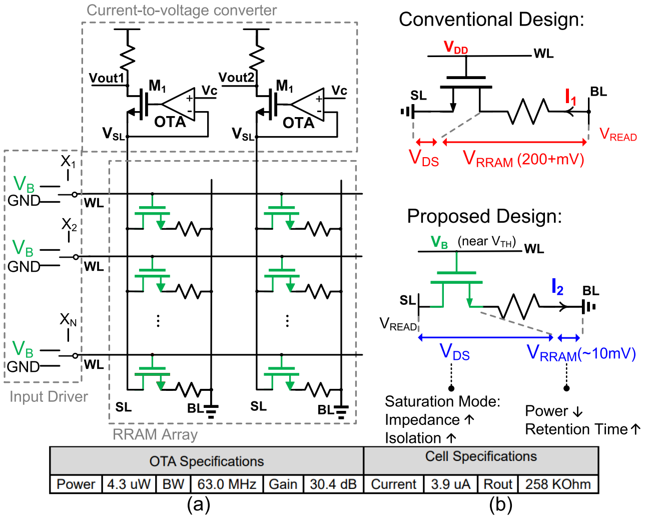


FIGURE 2. **(a) Conventional serial input parallel weight RRAM macro schematic. (b) Flow diagram of the operation.**



OTA Specifications						Cell Specifications			
Power	4.3 uW	BW	63.0 MHz	Gain	30.4 dB	Current	3.9 uA	Rout	258 KOhm

FIGURE 3. **(a) IIB-based RRAM macro and RCs. (b) Comparison between conventional RRAM cell and proposed IIB-based RRAM cell.**

of mA, and the lumped output resistance can be down to tens of Ω. Assuming that no distortion correction is employed, the I–V circuit must provide very low input impedance to guarantee low distortion during read-out, leading to several mW power consumption per column. While this can be relaxed by utilizing the ADC reference levels to compensate for the distortion, the overhead is just moved to the ADC end. Large currents can also bring an extra source of errors due to the IR drop along the long array lines. In essence, these drawbacks impose a steep tradeoff between energy efficiency, parallelism, and throughput for RRAM CIM design, limiting the scalability of RRAM CIM to high-resolution DNN applications. This motivates us to develop solutions to improve both the array-level design (see Section III) and interface design (see Section IV).

III. PROPOSED IMPEDANCE BOOSTED RRAM SUBARRAY

In existing RRAM CIM macros, the access transistors are always fully turned on in the triode mode as simple switches. While this is helpful for fast current development when the RRAM works as memory, it loses the impedance and current regulation capability in the saturation mode that can be useful for analog computing. Motivated by this, we propose the IIB technique, which exploits the access transistor's saturation-mode properties to solve the steep tradeoff between power and parallelism.

A. CIRCUITS AND OPERATIONS

The schematic of the proposed IIB-RRAM is shown in Fig. 3(a). Note that this idea retains the WL-input 1T1R scheme; hence, it is fully compatible with the current RRAM array. In contrast to the common approach in Fig. 3(b), which uses the bit-lines (BLs) for output with the source-lines (SLs) grounded, the proposed design adopts a swapped connection (using the SLs as output and BLs as ground). On top of this, instead of being driven all the way to VDD, the WLs are connected to a bias voltage (V_B), which is slightly above the threshold voltage. This arrangement brings two key benefits. First, the access transistors in the RRAM array are designed as CG current buffers that amplify the output impedance. Second, the saturation mode of the access transistor isolates the SL voltage from the RRAM cell voltage, enabling the current reduction of each cell.

Without loss of generality, we can use the square-law model to gain intuition. Despite being inaccurate for exact current calculation, it can well represent the trend. Assuming that an RRAM cell is programmed to LRS and logic "1" is applied to WL (WL voltage is V_B), the value of cell current and the cell output resistance observed from the SL ($R_{o,\text{cell}}$) can be expressed as

$$I_{\text{cell}} \approx \frac{1}{R_L^2 K_n} - \frac{1}{R_L}\sqrt{\left(\frac{1}{R_L K_n}\right)^2 - (V_B - V_{\text{TH}})^2} \qquad (1)$$

$$R_{o,\text{cell}} \approx R_{\text{RRAM}} \cdot g_m r_o \qquad (2)$$

where V_{TH} and K_n are the threshold voltage and the lumped process coefficient of the access transistor, R_L is the LRS resistance, g_m denotes the transistor transconductance, and r_o is the small-signal output resistance of the access transistor. From (1), the drain and source voltages are isolated in the saturation mode. By making V_B close to and only slightly above V_{TH}, the voltage drop across the RRAM cell is kept very small, thus facilitating a much lower cell current. The choice of V_B is discussed in Section III-B. From (2), it can be known that the equivalent output impedance from drain to source is amplified by a $g_m r_o$ term, which is usually called "intrinsic gain." Typically, the intrinsic gain of access transistors can reach a few tens to hundreds due to the extra channel length to support high-voltage programming. In short, the IIB technique transforms the RRAM array into a small-value high impedance current mode digital-to-analog converter (DAC) with computation capability. It simplifies the interface design, making the simultaneous design of high parallelism and low power possible.

Taking advantage of the optimized array current and impedance, this work employs a structure resembling a gm-boosted CG amplifier to collect the column MAC current and convert it to voltage, as illustrated in Fig. 3(a). Note that, though this transimpedance stage may share similarities with the current interfaces in some existing works [11], the proposed IIB technique makes their specifications less demanding. In traditional designs without IIB, the read voltage at V_{SL} must be clamped very stably through a strong auxiliary amplifier to prevent creating nonlinear current (by boosting the gm to reduce input impedance of CG amplifier). Thanks to IIB boosting the array impedance and reducing the cell current, V_{SL} can be easily stabilized. Therefore, we are able to use a small CG transistor M_1 and a five-transistor OTA to guarantee robust operation.

To verify the effectiveness of the IIB, we simulate a 128-parallelism XNOR-based [12] RRAM array using Cadence Spectre. The RRAMs are implemented using the ASU RRAM model [29], [30], with the transistor model from TSMC 28-nm CMOS. Based on simulation results, an output voltage versus MAC value transfer curve can be obtained, as shown in Fig. 4(a). To facilitate comparison, we also simulate two baseline RRAM read schemes and plot the results alongside in Fig. 4(a). The first baseline is current-mode-based RCs [11], and we convert the output current to voltage through linear mapping. The second one is divider-based RCs [12]. Note that both two baselines did not use the IIB-based technique, so the cell current is relatively high compared to our work, which results in nonlinearity. The output voltage range is normalized to [0, 1] for a fair comparison. It can be seen that our proposed technique produces a transfer curve that exhibits minimum distortion over a large MAC range compared to the baselines. We further quantify the linearity by calculating the code distance versus MAC value, as shown in Fig. 4(b). It proves that the code distance in our work keeps a constant, and DNN computational error due to distortion will be minimized. We also examine the read voltage on the SL with IIB enabled and disabled, respectively, as shown in Fig. 4(c). Without IIB, the SL voltage fluctuates with the partial sum results, and voltage variation can reach up to 0.46 V, leading to large second-order effects on cell current. However, with the help of IIB, the SL voltage is stable over a large range (variation is less than 60 mV). Thus, the channel-length modulation effect of each cell can be minimized. In conclusion, compared to prior switch-based CIM macros, our work keeps a linear transfer curve even with large parallelism. To explore the linearity performance of the proposed IIB technique, the output voltage is tested under different parallelism of 128, 256, 512, and 1024, as shown in Fig. 4(d). The nonlinearity has a minor effect on the output voltage even if 512 parallelisms are applied, indicating that IIB makes large parallelism possible.

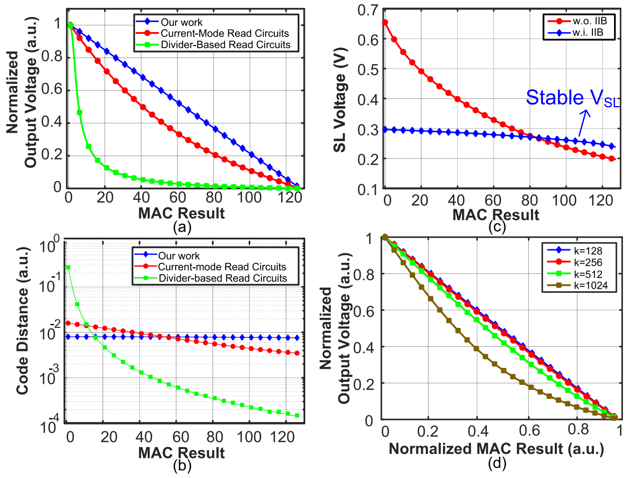


FIGURE 4. (a) Comparison of output voltage between our work and other works. (b) Comparison of code distance. (c) SL voltage comparison with/without IIB. (d) Output linearity under different parallelisms.

To eliminate the effect of the process, voltage, and temperature (PVT) variations, instead of using a fixed V_B, we propose a replica bias voltage generation circuit, as shown in Fig. 5(a). This structure ensures the similarity of working conditions between the bias generation branch and the RRAM cells, thus allowing the RRAM cell current to be well defined by the reference (I_{ref}) regardless of PVT variation. Based on a 1000-point simulation across different PVT conditions, the standard deviation of cell current is less than 0.05% of the mean value, which proves the robustness of the proposed circuits. It is worthwhile to mention that this bias generation scheme can be extended to support a multibit-per-cycle input scheme. For example, Fig. 5(b) shows a 2-bit-per-cycle example, where four different bias voltages (including GND) are generated by current sources with power-of-two weighted strength. Each 2-bit input serves as a control signal to choose from GND to V_{B1-3}. Note that the bias generation circuit can be shared by all rows, so the bias overhead is negligible.

B. DESIGN CONSIDERATIONS AND TRADEOFFS

The key design consideration of the proposed technique lies in the choice of the access transistor turn-on voltage V_B or, equivalently, the cell reference current I_{ref}. In the ideal case, I_{ref} can be arbitrarily small, as the access transistor will always operate in either the saturation or subthreshold mode, keeping the IIB effective. In practice, a lower bound for I_{ref} is limited by the following three factors: 1) thermal noise; 2) random mismatch; and 3) output swing and latency.

To illustrate the tradeoff between I_{ref} and thermal noise, we, hereby, introduce the concept of peak SNR ($\mathrm{SNR}_{\mathrm{peak}}$), defined as

$$\mathrm{SNR}_{\mathrm{peak}} = I_{\max}[k]^2/(P_n[k] + E_Q^2) \tag{3}$$

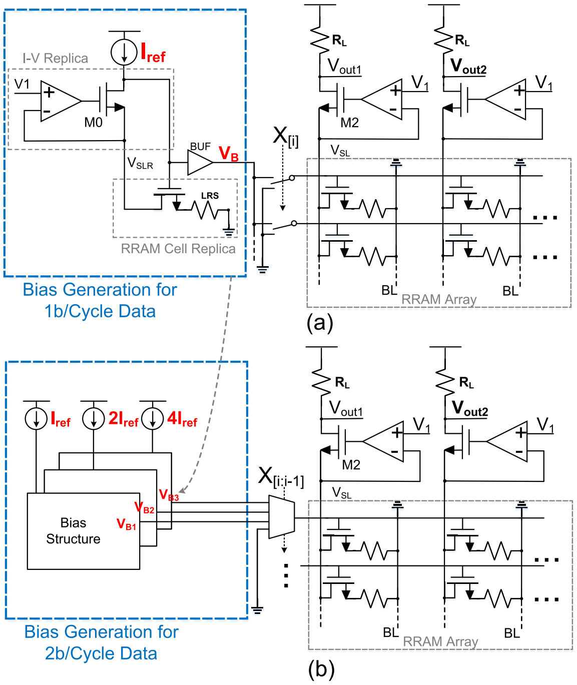


FIGURE 5. **(a) Bias voltage generation circuit for binary inputs. (b) Bias voltage generation circuit for multibit inputs (2-bit for example).**

where $I_{\max}[k]$ and $P_n[k]$ are the maximum column current and electrical noise power under parallelism k, respectively. E_Q represents the quantization noise of ADC in the current domain [31]. SNR$_{\text{peak}}$ can be viewed as a measurement matrix of the accuracy for reading out and digitizing the analog MAC value from the RRAM array. Without loss of generality, we assume the on-state cell current equals to I_{ref}, and the OFF-state current is zero. In addition, we also assume that the ADC's resolution is $\log_2(k)$ bits such that the current-referred quantization step is also I_{ref}. Then, the following expression can be obtained:

$$I_{\max}[k] = kI_{\text{ref}} \tag{4}$$

$$P_n[k] \approx k \times \text{BW} \times \left[4KT\gamma g_m\left(\frac{g_m}{1+g_mR_{\text{LRS}}}\right)^2 + \frac{4KT}{R_{\text{LRS}}}\left(\frac{g_mR_{\text{LRS}}}{1+g_mR_{\text{LRS}}}\right)^2\right] \tag{5}$$

$$E_Q^2 = I_{\text{ref}}^2/12 \tag{6}$$

where BW denotes the bandwidth of the RCs. K is Boltzmann's constant, and T is the temperature in Kelvin. γ is the "excess noise coefficient" [32], g_m is the transconductance of access transistor, and R_{LRS} is the LRS resistance of RRAM.

Based on (4)–(6), since $I_{\max}[k]$ and E_Q^2 are proportional to I_{ref}, reducing I_{ref} causes more thermal noise contribution, thus causing SNR$_{\text{peak}}$ degradation. To visualize it, Fig. 6(a) plots the simulated SNR of our testbench design at the maximum MAC value (where the SNR drop is the largest) as a function of I_{ref}. Therefore, cell current needs to be large enough to reduce thermal noise effects.

In addition to thermal noise, random mismatch also plays an important role in determining I_{ref}. Although the proposed bias generation circuit in Fig. 5 mitigates PVT variation issues, random mismatch effects between the bias branch and RRAM cells, such as threshold voltage mismatch and size mismatch, still induce deviations in the RRAM cell current. In Fig. 6(b), a 1000-point Monte Carlo simulation with mismatch is performed, and the current mismatch statistic is collected as a function of I_{ref}. It shows that the cell current standard deviation can reach up to 5.21% of I_{ref} with a 1-μA reference current, and increasing I_{ref} has the benefit of reducing the ratio of cell current standard deviation to I_{ref}. The reason is that the higher I_{ref} results in a higher overdrive voltage, which suppresses the current deviation.

Finally, the output swing and latency are also affected by I_{ref}. A larger output swing is desired not only because it has better noise rejection characteristics, but also it relaxes the ADC requirements and makes ADC easy to design. A low I_{ref} with a high R_L are expected to generate a high output swing and keep a low power consumption. However, the intrinsic tradeoff between output swing and latency limits R_L to be too large. In Fig. 6(c), the output voltage swing versus I_{ref} is plotted under different load resistances. When I_{ref} is low, the output swing is mainly limited by insufficient voltage drop over R_L. For a high I_{ref}, the output swing is undermined by distortion from nonlinearity. Thus, we need to choose a moderate I_{ref} to provide enough output swing and prevent distortion.

Based on the discussions above, we select I_{ref} to be 3.9 μA for the balance of SNR, random mismatch, and output swing. In this case, the SNR$_{\text{peak}}$ reaches 52.6 dB, which means the signal power is almost 160k times larger than the noise power, so thermal noise will not interfere with signals. To maximize the output swing and reduce the latency, a load resistance of 1800 Ω is chosen, so a 0.63-V output swing of the RC can be obtained, which is efficient to drive ADC and keeps output voltage with good linearity. In addition, the output latency is also determined by the load resistance and cell current. As shown in Fig. 6(a), as long as I_{ref} is not too small, the SNR is dominated by quantization error instead of the RC noise. In this region, we do not necessarily need to trade the sampling BW, so sub-ns latency can still be achieved. In this work, 0.43-ns latency is obtained if 3 fF is used as unit capacitance in CDAC, which enables a high-throughput design. The cell current variation attributed to transistor mismatch and PVT variations is limited within 3%, which is negligible compared to RRAM device variation [22], [33]. The comparison of the operating cell current is illustrated in Table 1. Compared to the conventional RRAM cell, the proposed IIB-based cell reduces the ON-state and OFF-state currents by 10.3× and 2.6×, respectively. Our proposed IIB-based RRAM cell helps to reduce I_{ON} and I_{HRS}; however, the ON–OFF ratio becomes smaller, indicating a more severe ambiguity issue. Fortunately, this problem can be solved by

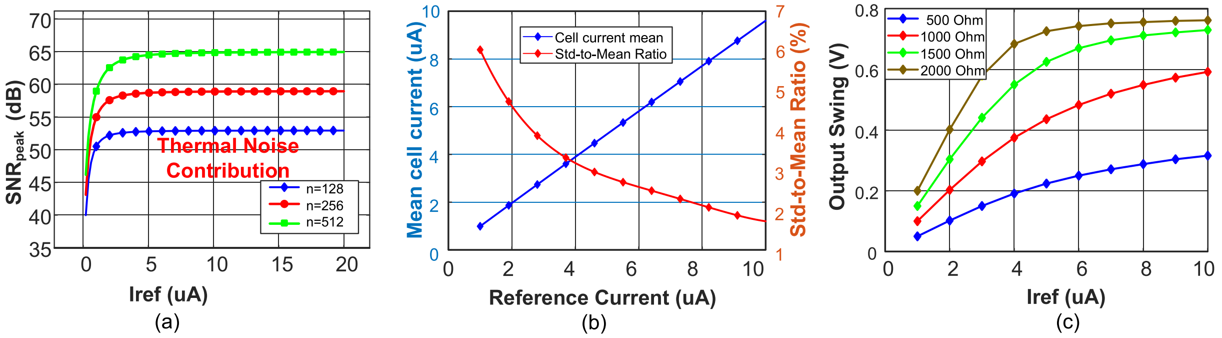


FIGURE 6. (a) SNR simulation under different I_{ref}'s. (b) 1000-point simulation for current deviation under different reference currents. (c) Simulation result of output swing under different I_{ref}'s and R_L's.

TABLE 1. Operating current comparison.

	Conventional Cell*	IIB Cell
I_{ON}**	40.0-uA	3.9-uA
I_{OFF}	949-fA	359-fA
I_{HRS}***	991-nA	291-nA

* The conventional cell currents are simulated under 0.5 V BL voltage and 1 V WL voltage.

** On-state and off-state RRAM resistances are 5K Ohm and 500K Ohm, respectively [29][30].

***I_{HRS} represents the cell current under high WL voltage and RRAM HRS state.

implementing XNOR RRAM cells, which encode two cells as one weight [12].

IV. PROPOSED ADC DESIGN WITH IN-ADC COMPUTING

In most existing RRAM-based CIM macros, flash ADCs are commonly employed [10], [11], [12], [13], [17], [18], [19] for their flexibility in individual transition level tuning to compensate for the readout distortion (more details in the Supplementary Material). With the read-out linearity greatly improved through the IIB technique, such distortion compensation can be obviated. This allows us to employ energy- and area-efficient ADC options, the voltage-mode SAR architecture. It saves the large overhead for tunable transition voltage generation and consumes less energy compared to the flash ADC at 5 b and beyond [34].

We observe that the CDAC capacitors are binary-weighted in SAR ADCs, which inherently turns the voltage across each capacitor into a weighted charge and adds up naturally during the SAR conversion. This characteristic allows weight reconstruction to be done inside the ADC, named IAC. Fig. 7(a) demonstrates one possible mode of IAC (Mode A), which performs a one-shot weighted summation of a 1-b-I-4-b-W MAC operation on a 5-b SAR ADC. When the first input signal $X[0]$ drives the WL, it multiplies with binary weights: W[0], W[1], W[2], and W[3], respectively. Then, the readout voltages of V_3, V_2, V_1, and V_0 can be sampled to different capacitors with a capacitance ratio of 2, while other capacitors are connected to a fixed dc voltage V_{CM}. Note that this configuration is generally referred to as bottom-plate sampling (BPS) in ADC design terminology. Following the sampling, the comparator side of the CDAC will be floated, while the input side merges to V_{CM}. This will initiate charge redistribution and create the weighted sum of V_3, V_2, V_1, and V_0 as V_{x0}, which can be expressed as

$$V_{x0} = \frac{31}{16}V_{\mathrm{CM}} - \frac{1}{2}V_3 - \frac{1}{4}V_2 - \frac{1}{8}V_1 - \frac{1}{16}V_0 \qquad (7)$$

where V_{x0} is the initial comparator input voltage (V_x) after BPS. Note that the equation suggests a negative weighted sum. This, in fact, is useful because $V0$–$V3$ have a negative slope over MAC value (see Fig. 4). The negative weighted sum allows the ADC output to be proportional to the MAC value. In addition, the comparator negative input voltage V_{center} is chosen to be the average voltage of the largest and smallest V_{x0}, which removes the dc offset due to V_{CM} and the I–V's inherent dc-level. With Mode A, only one ADC is required for four columns, so the ADC area and power overhead are reduced by 4× compared to the conventional A-D method. It also avoids the use of multiplexers, which eliminates the tradeoff between latency and area consumption.

To further increase the throughput, we propose Mode B IAC, as shown in Fig. 7(b). In this scheme, the DAC is divided into two parts, where the size ratio is 2:1. When $X[0]$ is connected to the array, its MAC results will be sampled by the smaller part of the DAC. Then, the top plate (i.e., the input side) of the DAC will be kept floating. Then, $X[1]$ is fed into the array, and its MAC result will be sampled to 2× capacitance DAC, constructing a 2× weight of data reconstruction. With Mode B, a 4 b × 4 b MAC requires only four samplings and two conversions of a 6-bit SAR ADC.

The operation of different modes of IAC is shown and compared in Fig. 7(c). The Mode A scheme collects the MAC results from every column and conducts S&A operation by utilizing the DAC capacitance ratio. For Mode B, an extra 2× DAC is implemented to help reconstruct the input weight. The advantages of Mode B are given as follows: 1) since only two 6-bit results are obtained for the subsequent process, less digital calculation is needed and 2) Mode B also reduces the latency of the CIM macro, as shown in Fig. 7(d). For Mode A, since each A-D conversion time is 4 ns, a total of 16 ns is needed to finish a 4-bit MAC operation. Note that, since the read-out circuits are disconnected with ADC after sampling, the next input can be fed to the array at the beginning of the ADC conversion phase, as shown in Fig. 7(d). In other words, the next MAC result can be calculated simultaneously

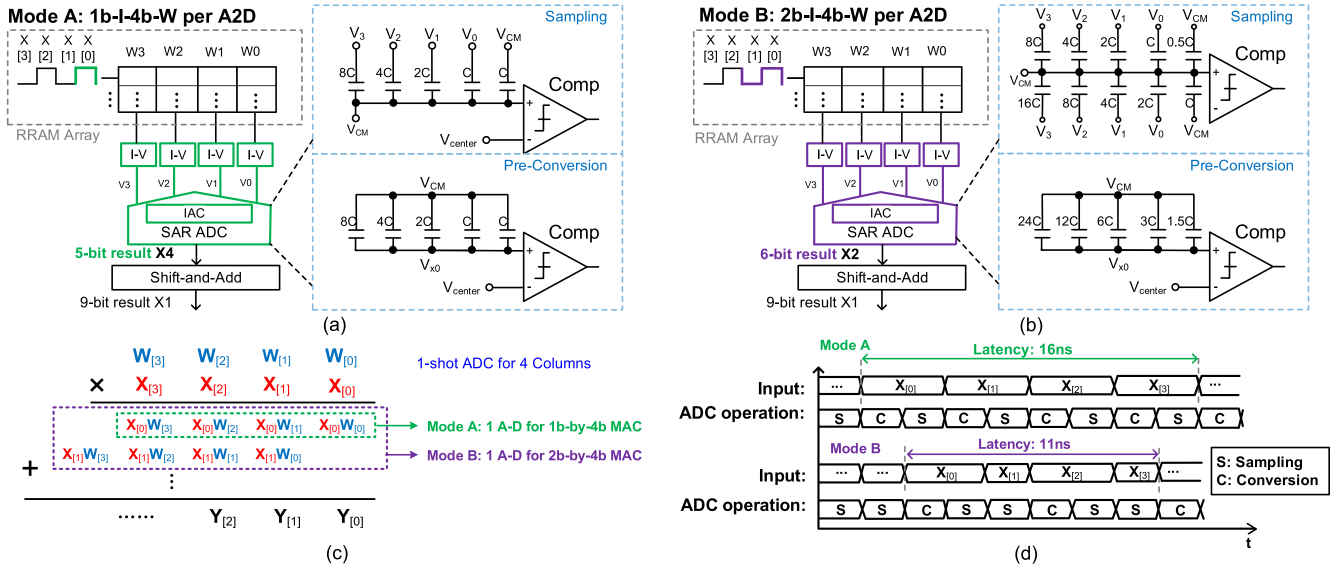


FIGURE 7. (a) Schematic of RRAM array with IAC-based data reconstruction (Mode A). (b) High throughput IAC schematic (Mode B). (c) Operation of 4-bit MAC operation with IAC. (d) Timing diagram of 4-bit MAC operation with IAC (Mode A and Mode B).

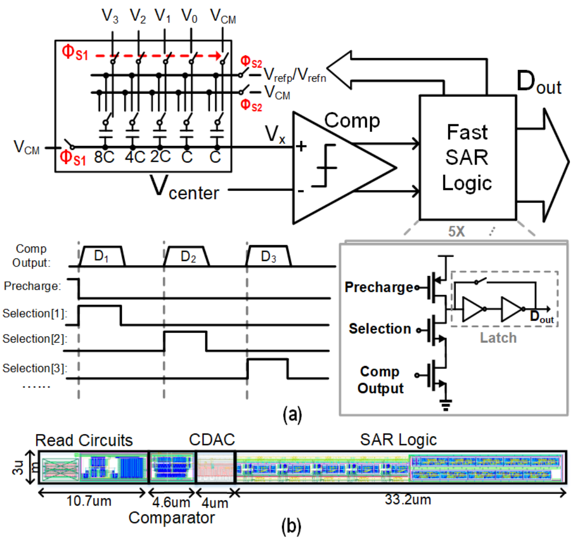


FIGURE 8. (a) Schematic 5-bit ADC with 4-bit IAC with latch-based fast SAR logic. (b) Layout of proposed SAR ADC.

with the ADC conversion phase. For Mode B, since the SAR conversion time is reduced by half, the total latency is only 11 ns, which means the throughput is improved by 45%.

In essence, the IAC-SAR approach allows ADC sharing across columns without the cost of time multiplexing. This not only retains throughput but also relaxes the area constraint on the ADC design. Still, because state-of-the-art RRAM technology can produce an array pitch to be as small as approximately 0.25 μm [36], careful design practice is needed to ensure a small ADC area. The detailed ADC schematic and waveform are shown in Fig. 8, where we devise the following strategies.

1) Synchronous timing is adopted for this design so that the clocking generation circuits can be shared by all ADCs.
2) We apply the Vcm-based switching technique [37] to eliminate the MSB capacitor and its control logic.
3) Fast SAR logic is utilized, which is based on latch instead of D-FFs, so the SAR logic circuits can be simplified [38].
4) A compact CDAC layout reported in [39] is implemented. Furthermore, the CDAC capacitors, made on the metal layers, are overlappingly placed with transistors to further reduce area consumption. In this work, the ADC's total length is 48.5 μm, and the width is only 3 μm, which makes it easily match the pitch width of the RRAM array.

V. SYSTEM-LEVEL SIMULATION RESULTS

The performances of our work and other state-of-the-art works are summarized and compared in this section. We benchmark our performance through SPICE simulation and NeuroSim simulator [42], [43]. We first train a VGG-8 network based on the CIFAR-10 dataset on 8-bit precision and get 90.8% accuracy as the baseline. Then, we use an RRAM-based CIM macro for inference. For the circuit simulation, we use Cadence Virtuoso EDA software and test the performances under TSMC 28-nm CMOS technology.

First, we compare the energy performance of proposed ideas with different configurations, as shown in Table 2. For the simulation setup, the array size of 256 × 128 (with XNOR RRAM cells) and the multiplexing ratio of 1:1 are chosen as macro structure. The ADC resolution is chosen to be 5 bit to provide enough inference accuracy and avoid consuming too much area and power. With Mode-A IAC, the ADC energy to finish 128 8 × 8 MAC is 81 pJ, and the area consumption is only 1164 μm^2, which is reduced by 4× compared to the conventional method. Mode B further reduces the latency by 31.25% compared to no IAC and Mode A cases. However, the tradeoff that it needs is a 3× sized CDAC, which burns

TABLE 2. Simulated performance for different configurations.

Setup		Performance			
Input Resolution (bit)	IAC Mode*	ADC Energy** (pJ)	ADC Area (um^2)	Energy Efficiency (TOPS/W)	Throughput (GOPS)
1	No IAC	324.0	4656	3.8	128.0
	Mode A	81.0	1164	6.1	128.0
	Mode B	75.2	1245	8.2	186.2
2	No IAC	162.0	4656	6.1	256.0
	Mode A	40.5	1164	8.9	256.0
	Mode B***	37.6	1245	12.2	372.4

*4-bit IAC is applied.

**The energy consumed to finish 128 8x8 bit task.

***In this case, it needs extra ADC resolution (at least 8-bit ADC) to guarantee the same inference accuracy, which will degrade the energy efficiency. So we select 2-bit input with IAC mode A as our result.

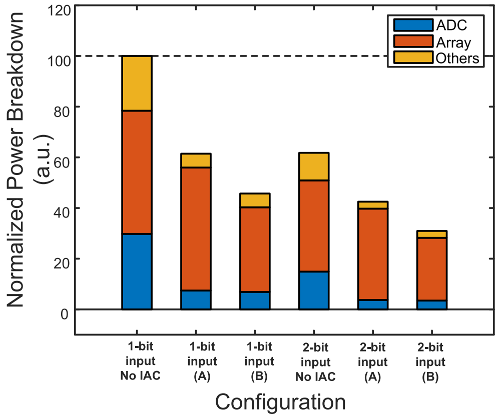


FIGURE 9. Power breakdown of different configurations.

3× CDAC power. In addition, the ADC resolution should be extended to maintain the same inference accuracy with 2-bit input and 4-bit Mode B IAC, which results in an exponentially increased CDAC power. The detailed normalized power breakdown under different configurations is compared in Fig. 9. In this work, we adopt a 2-bit input scheme with a 4-bit IAC (Mode A) to achieve a balance between energy efficiency, area, and accuracy.

Then, we compare our work with other state-of-the-art works [12], [17], [18], as shown in Table 3. In our work, we adopt 2-bit input and 1-bit weight; then, we use 5-bit ADC to digitize the result. A voltage-mode SAR ADC with IAC is proposed for A-D conversion and data reconstruction. Just as discussed in Section III, the output swing can be adjusted by changing R_L resistance, and 128 cells can be turned on simultaneously. Benefiting from the IIB technique, the cell operating current can be saved by 90%, and thus, the array energy can be minimized. The latency in our work comes from input digital delay, read delay, and analog-to-digital conversion delay. However, these delays can be implemented in a pipeline way (the SAR conversion can be processed simultaneously with RRAM array computing), so the final read delay will be 4 ns. For energy efficiency, we normalize all the work to 8-bit MAC. According to the simulation result, the energy efficiency of the proposed CIM macro is 8.9 TOPS/W due to power-efficient RRAM design and IAC, which is the highest among all the state-of-the-art works. Benefiting from the low distortion RRAM array and RCs' design techniques, the accuracy of 87.2% is achieved on the CIFAR-10 dataset.

TABLE 3. Performance comparison of different works.

	Our Work	ISSCC 2022 [17]	TED 2020 [12]	ISSCC 2019 [18]
Macro Precision	**2-1-5**	1-1-4	1-1-3	2-3-4
Data Reconstruction	**IAC**	Digital	Digital	Analog
Parallelism	**128**	256	64	9
Mux ratio	**1:1**	16:1	8:1	1:1
Read latency(ns)	**4**	5.3	6.5	14.6
Energy Efficiency (TOPS/W) *	**8.9****	26.6 (0.4) **	24.1 (0.4) **	21.9 (2.1) **
Accuracy	**CIFIR-10: 87.2%**	N/A	CIFAR-10: 83.5%	CIFIR-10: 88.5%

* The energy efficiency of this work is based on simulation results while others are measurement results.

**Normalized energy efficiency under 8-bit MAC.

VI. CONCLUSION

This article presents an intrinsic impedance-boosted RRAM array and its peripheral circuit design. This design reuses the access transistors as CG current buffers, which reduces the cell current and enables a linear read voltage with low complexity. In addition, a compact voltage mode SAR ADC with IAC further reduces the complexity of peripheral circuits and saves power. The proposed ideas make our RRAM macro achieve 87.2% inference accuracy while operating under 8.9 TOPS/W energy efficiency for 8-bit MAC operation.

REFERENCES

[1] W. A. Wulf and S. A. McKee, "Hitting the memory wall: Implications of the obvious," *ACM SIGARCH Comput. Archit. News*, vol. 23, no. 1, pp. 20–24, Mar. 1995.

[2] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, Dec. 2017.

[3] S. Srikanth, T. M. Conte, E. P. DeBenedictis, and J. Cook, "The superstrider architecture: Integrating logic and memory towards non-von Neumann computing," in *Proc. IEEE Int. Conf. Rebooting Comput. (ICRC)*, Nov. 2017, pp. 1–8.

[4] C. S. Lent et al., "Molecular cellular networks: A non von Neumann architecture for molecular electronics," in *Proc. IEEE Int. Conf. Rebooting Comput. (ICRC)*, Oct. 2016, pp. 1–7.

[5] S. Yu, "Resistive random access memory (RRAM)," *Synthesis Lect. Emerg. Eng. Technol.*, vol. 2, no. 5, pp. 1–79, 2016.

[6] D. Ielmini, "Brain-inspired computing with resistive switching memory (RRAM): Devices, synapses and neural networks," *Microelectron. Eng.*, vol. 190, pp. 44–53, Apr. 2018.

[7] H. Akinaga and H. Shima, "Resistive random access memory (ReRAM) based on metal oxides," *Proc. IEEE*, vol. 98, no. 12, pp. 2237–2251, Dec. 2010.

[8] A. Chen, "A review of emerging non-volatile memory (NVM) technologies and applications," *Solid-State Electron.*, vol. 125, pp. 25–38, Nov. 2016.

[9] S. Yu, X. Sun, X. Peng, and S. Huang, "Compute-in-memory with emerging nonvolatile-memories: Challenges and prospects," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Mar. 2020, pp. 1–4.

[10] J.-H. Yoon, M. Chang, W.-S. Khwa, Y.-D. Chih, M.-F. Chang, and A. Raychowdhury, "A 40 nm 64 Kb 56.67 TOPS/W read-disturb-tolerant compute-in-memory/digital RRAM macro with active-feedback-based read and in-situ write verification," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, vol. 64, Feb. 2021, pp. 404–406.

[11] C.-X. Xue et al., "Embedded 1-Mb ReRAM-based computing-in-memory macro with multibit input and weight for CNN-based AI edge processors," *IEEE J. Solid-State Circuits*, vol. 55, no. 1, pp. 203–215, Jan. 2020.

[12] S. Yin, X. Sun, S. Yu, and J.-S. Seo, "High-throughput in-memory computing for binary deep neural networks with monolithically integrated RRAM and 90-nm CMOS," *IEEE Trans. Electron Devices*, vol. 67, no. 10, pp. 4185–4192, Oct. 2020.

[13] W. He et al., "2-bit-per-cell RRAM-based in-memory computing for area-/energy-efficient deep learning," *IEEE Solid-State Circuits Lett.*, vol. 3, pp. 194–197, 2020.

[14] L. Wang et al., "Sparsity-aware clamping readout scheme for high parallelism and low power nonvolatile computing-in-memory based on resistive memory," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2021, pp. 1–4.

[15] Q. Liu et al., "A fully integrated analog ReRAM based 78.4 TOPS/W compute-in-memory chip with fully parallel MAC computing," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 500–502.

[16] M. Chang et al., "A 40 nm 60.64 TOPS/W ECC-capable compute-in-memory/digital 2.25 MB/768 KB RRAM/SRAM system with embedded cortex M3 microprocessor for edge recommendation systems," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2022, pp. 1–3.

[17] S. D. Spetalnick et al., "A 40 nm 64 kb 26.56 TOPS/W 2.37 Mb/mm^2 RRAM binary/compute-in-memory macro with 4.23× improvement in density and >75% use of sensing dynamic range," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2022, pp. 1–3.

[18] C.-X. Xue et al., "A 1 Mb multibit ReRAM computing-in-memory macro with 14.6 ns parallel MAC computing time for CNN based AI edge processors," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2019, pp. 388–390.

[19] W. Li, X. Sun, H. Jiang, S. Huang, and S. Yu, "A 40 nm RRAM compute-in-memory macro featuring on-chip write-verify and offset-cancelling ADC references," in *Proc. IEEE 47th Eur. Solid State Circuits Conf. (ESSCIRC)*, Sep. 2021, pp. 79–82.

[20] H. Jiang, W. Li, S. Huang, and S. Yu, "A 40 nm analog-input ADC-free compute-in-memory RRAM macro with pulse-width modulation between sub-arrays," in *Proc. IEEE Symp. VLSI Technol. Circuits (VLSI Technol. Circuits)*, Jun. 2022, pp. 266–267.

[21] J. M. Correll et al., "A fully integrated reprogrammable CMOS-RRAM compute-in-memory coprocessor for neuromorphic applications," *IEEE J. Explor. Solid-State Comput. Devices Circuits*, vol. 6, pp. 36–44, 2020.

[22] J. M. Correll et al., "An 8-bit 20.7 TOPS/W multi-level cell ReRAM-based compute engine," in *Proc. IEEE Symp. VLSI Technol. Circuits (VLSI Technol. Circuits)*, Jun. 2022, pp. 264–265.

[23] W.-H. Chen et al., "A 16 Mb dual-mode ReRAM macro with sub-14 ns computing-in-memory and memory functions enabled by self-write termination scheme," in *IEDM Tech. Dig.*, Dec. 2017, pp. 28.2.1–28.2.4.

[24] Y. Chen, L. Lu, B. Kim, and T. T.-H. Kim, "A reconfigurable 4T2R ReRAM computing in-memory macro for efficient edge applications," *IEEE Open J. Circuits Syst.*, vol. 2, pp. 210–222, 2021.

[25] C.-X. Xue et al., "A 22 nm 2 Mb ReRAM compute-in-memory macro with 121–28 TOPS/W for multibit MAC computing for tiny AI edge devices," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 244–246.

[26] F. Tan et al., "A ReRAM-based computing-in-memory convolutional-macro with customized 2T2R bit-cell for AIoT chip IP applications," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 67, no. 9, pp. 1534–1538, Sep. 2020.

[27] C.-X. Xue et al., "A 22 nm 4 Mb 8b-precision ReRAM computing-in-memory macro with 11.91 to 195.7 TOPS/W for tiny AI edge devices," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2021, pp. 245–247.

[28] W.-H. Chen et al., "CMOS-integrated memristive non-volatile computing-in-memory for AI edge processors," *Nature Electron.*, vol. 2, no. 9, pp. 420–428, Aug. 2019.

[29] P.-Y. Chen and S. Yu, "Compact modeling of RRAM devices and its applications in 1T1R and 1S1R array design," *IEEE Trans. Electron Devices*, vol. 62, no. 12, pp. 4022–4028, Dec. 2015.

[30] Z. Jiang, S. Yu, Y. Wu, J. H. Engel, X. Guan, and H.-S.-P. Wong, "Verilog-A compact model for oxide-based resistive random access memory (RRAM)," in *Proc. Int. Conf. Simulation Semiconductor Processes Devices (SISPAD)*, Sep. 2014, pp. 41–44.

[31] F. Maloberti, *Data Converters*. New York, NY, USA: Springer, 2007.

[32] A. van der Ziel, "Thermal noise in field-effect transistors," *Proc. IRE*, vol. 50, no. 8, pp. 1808–1812, Aug. 1962.

[33] Y. Pang et al., "A reconfigurable RRAM physically unclonable function utilizing post-process randomness source with $<6\times10^{-6}$ native bit error rate," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2019, pp. 402–404.

[34] H. Jiang, W. Li, S. Huang, S. Cosemans, F. Catthoor, and S. Yu, "Analog-to-digital converter design exploration for compute-in-memory accelerators," *IEEE Design Test*, vol. 39, no. 2, pp. 48–55, Apr. 2022.

[35] W.-H. Chen et al., "A 65 nm 1 Mb nonvolatile computing-in-memory ReRAM macro with sub-16 ns multiply-and-accumulate for binary DNN AI edge processors," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2018, pp. 494–496.

[36] C.-F. Lee, H.-J. Lin, C.-W. Lien, Y.-D. Chih, and J. Chang, "A 1.4 Mb 40-nm embedded ReRAM macro with 0.07 μm^2 bit cell, 2.7 mA/100 MHz low-power read and hybrid write verify for high endurance application," in *Proc. IEEE Asian Solid-State Circuits Conf. (A-SSCC)*, Nov. 2017, pp. 9–12.

[37] Y. Zhu et al., "A 10-bit 100-MS/s reference-free SAR ADC in 90 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 45, no. 6, pp. 1111–1121, Jun. 2010.

[38] C.-H. Chan, Y. Zhu, S.-W. Sin, U. S.-P. Ben, and R. P. Martins, "A 6 b 5 GS/s 4 interleaved 3 b/cycle SAR ADC," *IEEE J. Solid-State Circuits*, vol. 51, no. 2, pp. 365–377, Feb. 2016.

[39] N. Le Dortz et al., "A 1.62 GS/s time-interleaved SAR ADC with digital background mismatch calibration achieving interleaving spurs below 70 dBFS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 386–388.

[40] S. Xie, C. Ni, A. Sayal, P. Jain, F. Hamzaoglu, and J. P. Kulkarni, "eDRAM-CIM: Compute-in-memory design with reconfigurable embedded-dynamic-memory array realizing adaptive data converters and charge-domain computing," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2021, pp. 248–250.

[41] R. H. Walden, "Analog-to-digital converter survey and analysis," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 4, pp. 539–550, Apr. 1999.

[42] X. Peng, S. Huang, H. Jiang, A. Lu, and S. Yu, "DNN+NeuroSim V2.0: An end-to-end benchmarking framework for compute-in-memory accelerators for on-chip training," 2020, *arXiv:2003.06471*.

[43] X. Peng, S. Huang, Y. Luo, X. Sun, and S. Yu, "DNN+NeuroSim: An end-to-end benchmarking framework for compute-in-memory accelerators with versatile device technologies," in *IEDM Tech. Dig.*, Dec. 2019, pp. 32.5.1–32.5.4.