Light-CIM: A Lightweight ADC/DAC-Fewer RRAM CIM DNN Accelerator with Fully-Analog Tiles and Non-Ideality-Aware Algorithm for Consumer Electronics

Chenyang Zhao, Jinbei Fang, Jingwen Jiang, Xiaoyong Xue, *Member, IEEE,* , Xiaoyang Zeng, *Member, IEEE,*

***Abstract*—Neuromorphic computing has emerged as a revolutionary technology in consumer electronics, with computing-in-memory (CIM) attracting considerable attention for its potential to minimize data transfer. However, most CIM accelerators necessitate numerous digital-to-analog converters (DACs) and analog-to-digital converters (ADCs) for mixed-signal data processing, resulting in substantial area and energy overheads. This study introduces a lightweight CIM accelerator, Light-CIM, which operates with fully analog tiles and employs a non-ideality-aware algorithm. A fully analog tile consists of one-transistor-one-resistor (1T1R) arrays based on resistive random access memory (RRAM) and customized analog peripheral circuits for data processing. The intra-tile data computation, transfer, and buffering are all in analog voltage, current, or RRAM resistance, thus eliminating costly DACs and ADCs for intermediate data conversions in conventional CIM accelerators. The fully analog approach significantly reduces power consumption attributed to ADCs, accounting for only 2.5% of the total power consumption. Additionally, a non-ideality-aware training algorithm is employed to enhance the robustness of the hardware system. It models and incorporates non-idealities of circuits in software training, including read nonlinearities, mismatches, variations, and noises in the hardware analog data flow. Experimental results demonstrate that Light-CIM achieves accuracy close to software performance in various NN models. Light-CIM accomplishes a compute density of 3.91 TOPS/mm^2 and an energy efficiency of 3.08 TOPS/W, both highly competitive compared to state-of-the-art works.**

***Index Terms*—Consumer Electronics, Neuromorphic Computing, Computing-in-Memory, DNN Accelerator, Fully-Analog Tile, Non-Ideality-Aware Algorithm.**

I. Introduction

.

CONSUMER electronics, ranging from smartphones to edge devices and internet of things (IoT) gadgets, are witnessing a paradigm shift towards applications demanding complex computations, particularly in the realm of artificial intelligence (AI) and machine learning (ML). With the growing demand for energy-efficient and high-performance computing solutions, neuromorphic computing technologies have emerged as a promising consumer electronics approach. One such technology, computing-in-memory (CIM), has garnered significant attention due to its capacity to minimize data transfer between the processor and memory, coupled with its efficient execution of multiply-and-accumulate (MAC) operations. These features contribute to improved energy efficiency and reduced latency compared to conventional von Neumann architectures [1]. Specifically, CIM provides a viable solution for accelerating deep neural network (DNN) computations in consumer electronics with limited area and power budgets. Among the various emerging memory technologies, resistive random-access memory (RRAM) has attracted interest for CIM implementations due to its high density, substantial on/off ratio, analog programmability, and compatibility with complementary metal-oxide-semiconductor (CMOS) processes. Several RRAM-based CIM accelerators have already demonstrated potential for AI edge applications [2]–[4]. However, most existing CIM accelerators, such as ISAAC, are constituted of mixed-signal circuits and require numerous analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) for data conversion [1], [5], [6]. This may lead to considerable area costs and energy consumption, impairing the gains of CIM.

Recently, several efforts have been made to address the above-mentioned issue. DigitalPIM constructs memory blocks that can internally support essential operations in memory without DACs and ADCs [7]. This significantly reduces processing costs while providing high scalability and parallelism. FloatPIM also eliminates the use of DACs and ADCs using the same approach as DigitalPIM and further supports floating-point representation [8]. MOSAIC adopts the mechanism of bit splitting and employs a 1-bit word-line (WL) driver to replace DAC and a 1-bit sense amplifier (SA) to replace ADC for the minimal area of peripheral circuits [10]. Lattice moves the computation between the feature maps and the weights of network layers to a CMOS peripheral circuit to eliminate the costly DACs and ADCs [9]. ARCHON designs analog neuronal computation units based on time-domain conversions and charge accumulation, with the help of capacitor-based analog memory, managing to implement neural network (NN)

This work was supported in part by the National Key R&D Program under Grant 2023YFB4404700, in part by the National Natural Science Foundation of China under Grant 62274038, in part by the Science and Technology Commission of Shanghai Municipality under Grant 21TS1401200 and Grant 22ZR1407100, and in part by State Key Laboratory of Integrated Chips and Systems under Grant SKLICS-Z202315. (Chenyang Zhao and Jinbei Fang contributed equally to this research; Corresponding author: Xiaoyong Xue).

C. Zhao, J. Fang, J. Jiang, X. Xue, and X. Zeng are with the State Key Laboratory of Integrated Chips and Systems, Fudan University, Shanghai, 200438, China (e-mail: xuexiaoyong@fudan.edu.cn).

acceleration without any analog-to-digital conversions. Some works attempt to use buffer RRAM arrays or buffering capacitors to store intermediate results directly from the computing arrays without passing through ADCs [15], [18]. There are also attempts based on exploiting spiking neural network (SNN) intrinsic features or designing analog MAC cells from scratch [19], [20]. Meanwhile, BRAHMS performs all multiplications, and activation/pooling operations in the analog domain [25]. Thus, only retained analog values after activation/pooling require A/D conversion. Fuse-and-Mix introduces a magnetic tunnel junction (MTJ)-based analog content-addressable memory and devises a fused analog activation unit based on MACAM that can simultaneously achieve nonlinear activation and A/D conversion with substantial energy reduction [26].

However, DigitalPIM and FloatPIM are both based on voltage division between RRAM cells in series or parallel connection for logic operation, which may suffer from small operation margins and bring write disturbance to the RRAM cells. Besides, the RRAM resistance is often subject to read nonlinearity, which can further influence the reliability of the operation. MOSAIC may encounter serious discrepancies when combining the multi-bit results after nonlinear processing. Lattice is, in fact, a form of near-memory computing, which reduces DACs and ADCs to improve area efficiency but fails to exploit the parallelism enabled by the array-based MAC computation. ARCHON deploys the computing data path outside the memory array and also faces the problem of confined parallelism and performance compared to CIM-based works. Works like [15], [18] explore the benefits of analog computing at the macro or processing element level, which may obtain a limited improvement in hardware efficiency. BRAHMS is susceptible to signal noise as the analog signals need to be routed through analog content-addressable memory three times for nonlinear projection, pooling, and A/D conversion. Fuse-and-Mix can only realize ReLU-α instead of supporting general nonlinear functions like BRAHMS.

After studying the architectures of mainstream ADC-based CIM AI accelerators, we found that existing works usually place their DACs and ADCs at the macro level, to satisfy the high-resolution requirements of intermediate results from computing arrays. While the lower hierarchical level the conversions are conducted, the more DACs and ADCs are needed, causing larger area and energy overheads, as illustrated in **Fig. 1(a)**. CASCADE [15] manages to save the macro-level data conversions but fails to extend this feature to higher levels or in a wider range of functions, as shown in **Fig. 1(b)** and **Fig. 1(c)**. To our knowledge, it remains unexplored for CIM-based accelerators to realize a tile-level fully analog data flow with customized analog processing elements. In this work, we propose a lightweight ADC/DAC-fewer CIM accelerator, abbreviated as Light-CIM, with fully analog tiles (FANTs) and a non-ideality-aware training algorithm for cost-efficient AI applications in consumer electronics. The FANTs mainly consist of RRAM-based one-transistor-one-resistor (1T1R) arrays and customized analog circuits, to execute operations required by NNs. The DACs and ADCs are only used as the tile interface, participating in communications with other FANTs, and thus significantly reducing the hardware costs, as depicted

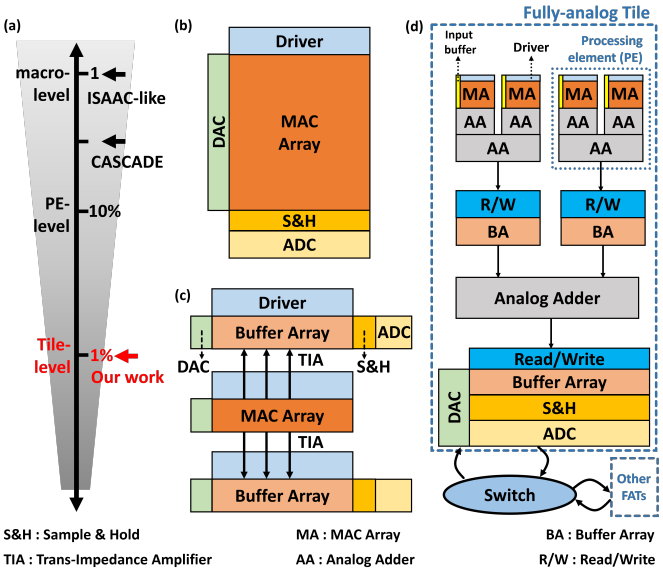

Fig. 1. Comparison of architectural approaches and advancements. (a) The theoretical remaining percentages of DACs and ADCs with different level placement schemes; (b) Macro-level ADC-placement in conventional ISAAC-like architecture [5]; (c) ADC-placing scheme used in CASCADE [15]; (d) The architecture of proposed Light-CIM accelerator.

in **Fig. 1(d)**. Because analog computing is sensitive to the non-idealities in RRAM devices and analog circuits, a circuit-algorithm co-optimization is further proposed by employing a non-ideality-aware training algorithm to improve robustness.

Compared to prior works, we have made the following contributions. **Firstly**, the architecture of Light-CIM is presented with fully analog data flow within tiles. Intra-tile data are all processed, transferred, and buffered in the analog domain. This means that either inputs, weights, or the intermediate data directly from RRAM arrays are all in analog voltage, current, or RRAM resistance, fundamentally eliminating the use of DACs and ADCs within tiles, and thus boosting energy and area efficiencies. **Secondly**, we systematically provide the analog peripheral circuits of Light-CIM to realize fully analog intra-tile data flow. RRAM-based analog buffers are used as intermediate data storage at different hierarchical levels. Input buffers are designed for current-voltage conversion before the inputs are fed into the computing RRAM arrays. We also design different analog processing elements to implement the merging of partial sums, non-linear activation, and max-pooling functions. **Thirdly**, a non-ideality-aware training algorithm is proposed to improve the robustness of Light-CIM by bringing the major non-ideality models of RRAM devices and analog circuits into the training process. The simulation results of MLP, VGG-8, VGG-11, and ResNet-18 show that the accuracy can be recovered to be close to software performance on MNIST, CIFAR-10, and CIFAR-100 datasets.

The subsequent sections of this paper are organized as follows: **Section II** provides an overview of related work and outlines the challenges. **Section III** presents the overall architecture and main analog circuits of Light-CIM. The non-ideality-aware training algorithm is expounded in **Section IV**. Evaluation and discussions are conducted in **Section V**. The

paper concludes with **Section VI**, presenting final remarks and avenues for future research.

II. Related Work and Challenges

A. Related Work

Light-CIM is based on the classical CIM architecture ISAAC but in an intra-tile fully-analog fashion [5]. As analyzed above, although works like DigitalPIM, FloatPIM, MOSAIC, Lattice and Neuro-CIM are promising for reducing or eliminating the use of DACs and ADCs by exploiting stateful logic, bit-splitting, near-memory computing and SNN intrinsic features, the difficulties still need to be overcome in terms of operation reliability, MAC computation efficiency, limited parallelism or network accuracy [7]–[10], [20]. Works like [15], [18] employ a similar buffer RRAM or capacitor scheme with our work but limit the attempt at the macro or processing element level, and also face problems such as RRAM non-idealities, and circuit nonlinearities, which are not well dealt with.

Furthermore, most demonstrated CIM test chips still adhere to the CIM architectures of PRIME or ISAAC [1], [5], [6]: the memory array handles MAC computation in an analog fashion, while peripheral digital circuits manage other DNN function computations, such as adding/subtracting, nonlinear activation, pooling, etc. DACs and ADCs facilitate the data conversion between the digital domain and the analog domain. Consequently, this work endeavors to employ analog circuits to realize the DNN functions previously executed by digital circuits at the tile level. Thus, intra-tile data conversions can be eliminated, significantly reducing the dependence on DACs and ADCs at the chip level.

B. Challenges for Light-CIM

Given the adoption of a fully analog approach for DNN acceleration within tiles, several challenges demand attention. Firstly, the analog circuits for DNN function computations are susceptible to mismatch, variation, noise, and so on. Secondly, the intermediate data needs to be stored in an analog fashion. For convolution operations, a volume of convolution results after nonlinear processing needs to be cached in analog storage media before transferring to the next NN layer. Thirdly, the non-ideality in RRAM may impact network performance [11]. For RRAM used as a synapse in CIM accelerators, read linearity is particularly critical. The resistances of high/low resistance states (HRS/LRS) will change with the applied voltage, potentially severely impairing inference accuracy. In PRIME, read nonlinearity is not considered even when the WL drivers of DAC apply varying voltages on RRAM cells. ISAAC and recent test chips employ constant voltage on RRAM cells to avoid the effect of read nonlinearity. Our proposed accelerator Light-CIM directly utilizes analog voltages as input on RRAM without DACs, which will suffer from the effect of read nonlinearity. In summary, overcoming challenges from analog computing circuits, analog data buffers, and RRAM devices is imperative to make Light-CIM functional.

III. Light-CIM Architecture and Circuit

A. Overall Architecture

The Light-CIM architecture is structured hierarchically, comprising four levels: the chip (**level-1**), fully analog tile (FANT) (**level-2**), processing element (PE) (**level-3**), and array (**level-4**), as illustrated in **Fig. 2(a)**. ***At the chip level***, Light-CIM integrates FANTs, switches, the global controller, and the chip I/O interface. The on-chip concentrated mesh topology is employed, with four FANTs sharing a switch [5]. ***At the FANT level***, there are PE modules for vector-matrix multiplications (VMMs), the current mirror adder for data accumulation across different PEs/FANTs, the tile analog buffer for storing input and output analog data from various FANTs/PEs, the analog ReLU module for nonlinear activation, the analog max-pooling module for max-pooling operation, and the FANT controller. ***At the PE level***, there are multiple binary CIM arrays, the PE analog buffer for storing input and output analog data from different PEs/arrays, the current mirror adder dedicated to inter-array data accumulation, and the PE controller. ***At the array level***, there is a 1T1R RRAM array for CIM operation, the input buffer for caching the analog input vectors, and the current mirror adder for shifting and adding the readout current from different bit-lines (BLs). Noteworthy is that analog buffers also adopt the 1T1R RRAM array but leverage an analog programming scheme for analog data storage [12], [13]. Controllers in different layers primarily furnish control signals for decoders, clocks, and enable signals. Further details on each module are expounded in **Section IV**.

The workflow of Light-CIM is described as follows. The batch images captured by sensors are inscribed into the tile analog buffers in the form of analog data. Subsequently, these data are allocated to relevant PE analog buffers based on the mapping scheme. The data in the PE analog buffers are then loaded into each 1T1R array in the form of analog voltage through the input buffers. Pre-trained weights are preloaded into the 1T1R array. After the MAC computations by the array and necessary weighted sum operations, the results undergo processing by analog ReLU modules for nonlinear activation. Following this, analog max-pooling modules realize the reduction of parameter counts if required by the NN. The pooled results find residence in the analog buffers, serving as input for the next NN layer or as the final output of the intra-tile (i.e., intra-FANT) processing. This delineation of analog data flow across different hierarchical levels, as indicated by the blue lines with arrows in **Fig. 2(a)**, underscores that processing, transfer, and buffering all unfold in the analog domain from the array level to the FANT level. This fundamentally eliminates the data conversions executed at lower levels in conventional accelerator designs. A small number of ADCs are still retained at the FANT interfaces, enabling Light-CIM to convert intra-tile analog data into inter-tile digital data. On-chip concentrated mesh is used for fast and reliable digital inter-tile communication. The regular grid structure and efficient interconnectivity of concentrated mesh allow for the easy addition of new FANTs to the system without significant modifications nor compromising the performance of the overall architecture.

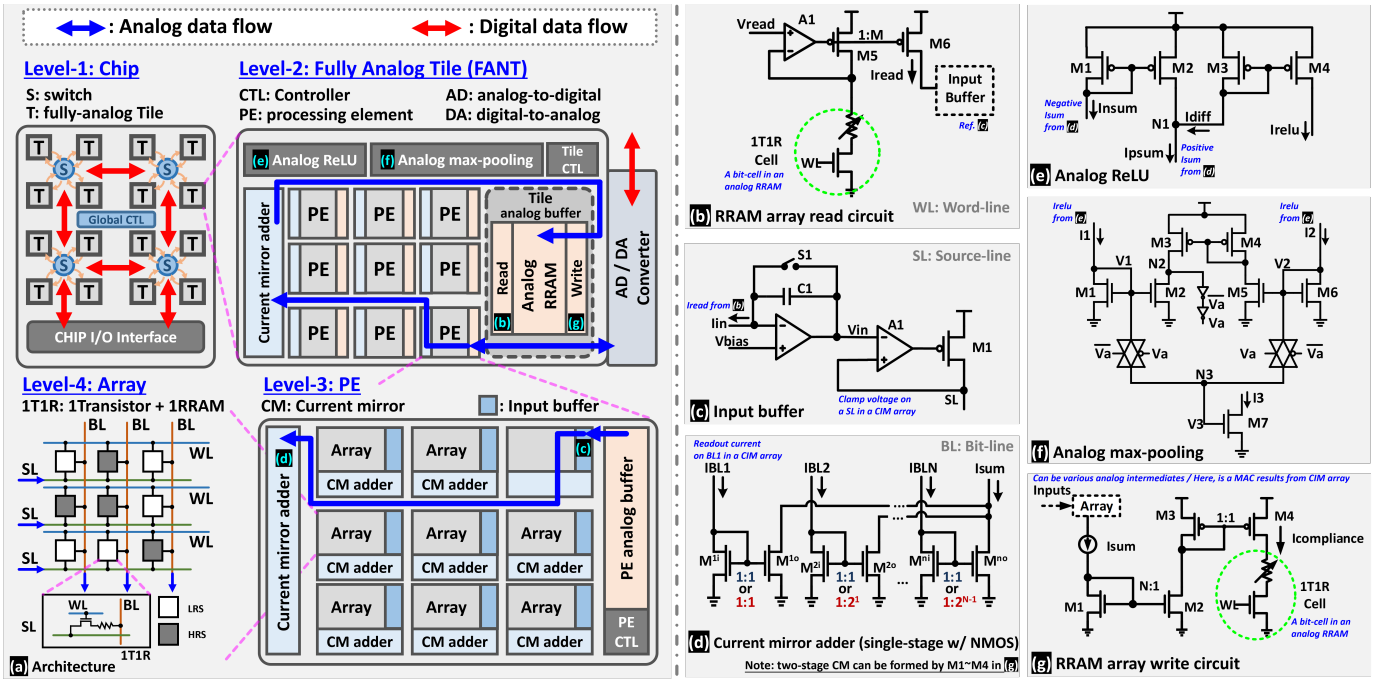

Fig. 2. Architecture and key circuits of the proposed Light-CIM accelerator. (a) Architectural overview of the proposed Light-CIM accelerator; (b) Read circuit for the RRAM array; (c) Input buffer circuit; (d) Current mirror adder circuit; (e) Analog rectified linear unit (ReLU) circuit; (f) Analog max-pooling circuit. (g) Write circuit for the RRAM array.

The distinctive FANT structure presents notable improvements in efficiency, particularly when applied to middle-sized or small-sized NNs. In situations where the majority of layers can be accommodated within a single FANT, an intra-tile layer mapping scheme alleviates the need for high-resolution ADCs and DACs, which play a pivotal role in facilitating intermediate data transfer across FANTs. In scenarios where each layer of an NN model can be mapped into a FANT, the resolution of ADCs and DACs across the entire chip is determined by the precision of activations and weights, a parameter that can be relatively modest in edge AI applications. For instance, considering the architecture of Light-CIM, in the context of a binarized neural network (BNN) where each layer fits almost neatly into a single FANT, 1-bit SAs can effectively replace all the ADCs in Light-CIM. This transition not only avoids the latency issues observed in PRIME but also circumvents data flow discrepancies encountered in MOSAIC. Furthermore, this optimized BNN design retains the advantage of a significant reduction in the frequency of data conversions, resulting in a marked enhancement of hardware efficiency. It should be noted that Light-CIM is designed as an inference accelerator for low-power consumer electronics, in which NNs are pre-trained and subsequently loaded onto the devices for inference. Further deployment of the full training process on Light-CIM needs to tackle the intrinsic limited RRAM endurance ($10^6 \sim 10^9$ [22]) without retention degradation to make a practical product for consumers, considering the frequent weight updates brought by on-chip training.

B. Analog Circuit Modules for Light-CIM

The analog buffers at both the FANT and PE levels are constructed using a 1T1R RRAM array, incorporating read and write circuits, along with ancillary modules such as decoders, selectors, and essential control circuits. The 1T1R analog RRAM array functions as the repository for analog input and output data from each NN layer. The read circuit extracts analog data from the 1T1R array in its current state utilizing an active clamper circuit (A1 and M5), as depicted in **Fig. 2(b)**. Maintaining a constant read voltage is crucial to nullify the impacts of read nonlinearity in analog RRAM. Notably, both the write and read currents undergo appropriate scaling. Programming the analog RRAM involves adjusting the set current compliance [12], [13]. The write circuit of the analog buffer module, illustrated in **Fig. 2(g)**, mainly relies on the current mirror structure, taking as input the processed MAC results from the array or data from other analog buffers in its current form. The incorporation of a write verify step can enhance programming accuracy [13]. A primary concern associated with utilizing analog RRAM for intermediate storage pertains to its relatively restricted endurance. However, this endurance challenge can be mitigated by striking a balance between endurance and retention of RRAM through careful adjustment of the compliance current [17]. In our design, the endurance can be elevated to an impressive 10^{20} times when retention is reduced to 10 ms, a level deemed acceptable for intermediate data storage requirements.

The analog RRAM circuits demonstrate versatility, extending their applicability to CIM arrays for MAC computations with a constant compliance current, implying that only 1-bit programming is performed for each analog RRAM cell.

This suggests that to simplify the practical implementation, the same type of analog RRAM can be used for both the input buffer and the CIM array. However, it is also feasible to equip the input buffer and CIM array with analog RRAM and dedicated binary RRAM, respectively, albeit potentially increasing hardware manufacturing costs [23]. Additionally, if analog RRAM is employed in the CIM array, there might be challenges in endurance and retention compared to binary RRAM. This can be mitigated by controlling the frequency of weight updates. This is because, generally, the hardware resources used for CIM may not be sufficient to map an entire NN model. Under such circumstances of frequent weight updates, excessively long retention times may not be necessary, providing an opportunity for analog RRAM to play a role.

The input buffer module in Light-CIM, as shown in **Fig. 2(c)** is an analog integrator followed by a clamper, which is used to integrate the read current, denoted as I_{in}, into an input analog voltage, represented by V_{in}. After V_{in} is clamped, a stable analog voltage is applied on the source-line (SL) of the CIM array while BL is clamped at a fixed voltage, thus ensuring a consistent analog voltage difference across the two ends of the RRAM cell used for MAC operation as the analog input.

The current mirror adder within the Light-CIM architecture serves two distinct purposes: the array level and the PE/FANT level. At the array level, the current mirror adder is intended for shifting and accumulating the output results of BLs within the CIM array. Each RRAM cell in the 1T1R array stores 1-bit weight data, thereby necessitating n-column BLs for hardware mapping of n-bit weights. Following the mapping pattern from the least significant bit (LSB) to the most significant bit (MSB), the outputs on these BLs undergo shifting and accumulation in the analog domain, through a set of current mirrors designed in proportions of $1:2^0, 1:2^1, \ldots, 1:2^{n-1}$, as illustrated in **Fig. 2(d)**. For the PE/FANT level, the current mirror adders mainly perform accumulation operations and generally do not involve shifting operations. The structure presented in **Fig. 2(d)** is retained, employing a set of current mirrors with a proportional ratio of $1:1, 1:1, \ldots 1:1$. This configuration allows the current mirror adders to provide only the accumulation functionality. This design caters to the general requirement of analog output accumulation at the PE/FANT level. Note that **Fig. 2** only shows the sketch schematic. In real situations, multi-stage current mirrors are used when the magnification ratio is too large to save area. Besides, the cascade structure is applied to mitigate the influence of channel-length modulation in the 28nm process.

The analog ReLU circuit consists of two analog current mirrors, as illustrated in **Fig. 2(e)**. Based on Kirchhoff's current law, the input current at a circuit node equals the output current, resulting in $I_{relu} = I_{psum} - I_{nsum}$. Due to the unidirectional flow of current from source to drain in the PMOS terminals of the current mirror formed by M3 and M4 (as the output voltage cannot exceed VDD), if $I_{nsum} > I_{psum}$, the analog ReLU module forces I_{relu} to zero, realizing the ReLU function.

The analog max-pooling circuit is also based on the analog current mirror, as depicted in **Fig. 2(f)**. Input currents, I_1 and I_2, are duplicated to node N2 through the current mirror. If $I_1 > I_2$, the potential at N2 approximates VSS; if $I_1 < I_2$, the potential at N2 approximates VDD. Subsequently, two inverter stages compensate for N2 voltage to achieve rail-to-rail operation. When $I_1 > I_2$, the left switch opens, copying I_1 to I_3; conversely, when $I_1 < I_2$, I_2 is copied to I_3. The proposed Light-CIM supports a customizable size of max-pooling operation, accomplished through multiple invocations of foundational 2-to-1 analog max-pooling modules. For instance, a 4-to-1 max-pooling can be realized using three 2-to-1 modules. If max-pooling operations are unnecessary, the analog max-pooling module can be bypassed through the control module.

IV. Non-Ideality-Aware Training Algorithm

A. Algorithm

Due to the intra-tile fully-analog fashion, Light-CIM is unable to cope with the hardware non-idealities like the traditional mixed-signal accelerators, which process data mainly in the digital domain. Analog data paths in Light-CIM may encounter various non-ideality issues, including read nonlinearities, mismatches, variations, noises, and so on. The above-mentioned non-idealities can severely harm the performance of Light-CIM, and the models trained with traditional algorithms may not work at all in extreme circumstances.

To confront this challenge, we abstract the analog data path from layer l-1 to layer l, as illustrated in **Fig. 3**. Starting from the analog buffer that stores the outputs of layer l-1 and ending with writing to the analog buffer storing the outputs of layer l, the data path is characterized by read nonlinearities exhibited by 1T1R arrays and all analog circuits along the data path. Meanwhile, circuit mismatches, variations, and noises also exist widely in the whole analog path. After analyzing and modeling these major non-idealities along the analog path, we bring these non-ideality models into the training process of NNs.

By analyzing the locations where various non-ideal circuit characteristics appear in the analog dataflow, it can be found that they originate from the output of the CIM array and end at the output of the analog ReLU or pooling module. Therefore, by customizing the nonlinear activation function, the nonlinear offset and random fluctuations caused by mismatch/variation/noise can be applied to the linear tensor received by the nonlinear activation function before the

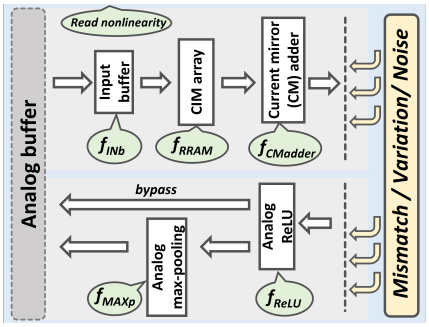

Fig. 3. data flow of Light-CIM with non-idealities.

ReLU judgment is made. The customized activation function, incorporating various non-idealities, is applied to each NN layer of the DNN model according to the hardware mapping scheduling.

To bolster the robustness of NNs against mismatch, variation, and noise, their mathematical models are expressly characterized solely during the forward propagation process. This operation concurrently serves as a regularization term, mitigating the risk of network overfitting.

Modeling approach involves constructing the analog circuit schematic using the Virtuoso tool, invoking MOS transistor models provided by the foundry. Mathematical models for variations and mismatches are derived from statistical analyses of Monte Carlo (MC) simulations across critical paths in our analog circuit design. Mathematical models for noises are also derived from simulation results of circuit noises in the complete data flow @ tt corner and 27°C temperature conditions. The mathematical modeling of read nonlinearity in each analog circuit is based on simulation results for the individual analog circuit modules, conducted @ tt corner and 27°C temperature conditions.

B. Modeling

We implemented a comprehensive analog circuit simulation for the Light-CIM architecture at the 28nm node. The core designs of the analog circuits employed in this work revolve around the proportional current mirror structure and integrators constituted by the basic operational amplifier. Through rigorous circuit analysis, we explored various non-ideal effects, encompassing the read nonlinearities in the 1T1R arrays and analog peripheral circuits, as well as addressing issues of mismatch, variation, and noise. It is noteworthy that for the modeling of RRAM cell variations, we leveraged the models and parameters provided by NeuroSIM [14].

Typically, the synaptic device resistance is nonlinear under different applied voltages due to the tunneling or hopping nature of the electrons in the cross-point junction. 1T1R cell is modeled based on the work [11] and different IV curves are obtained by simulation, as shown in the left part of **Fig. 4(a)**. In this work, the difference in RRAM conductance of LRS and HRS is taken as a variable to analyze the read-nonlinearity of the 1T1R cell. We define the read-nonlinearity as the ratio between the real conductance difference and the ideal one, which corresponds to f_{RRAM} in **Fig. 3**. The read-nonlinearity for nine combinations of HRS: 5 MΩ, 2 MΩ, 1 MΩ, and LRS: 200 kΩ, 50 kΩ, 20 kΩ is analyzed, as shown in the right part of **Fig. 4(a)**. Note that the nonlinearity of 1T1R cell is mainly determined by LRS, and different HRSs only make the nonlinearity fluctuate in a small range. Specifically, for LRS values of 200 kΩ, 50 kΩ, and 20 kΩ, the nonlinearity offset ranges were about 100%∼275%, 100%∼225%, and 100%∼170%, respectively. As the input voltage increases, the trend of nonlinearity for all three exhibit monotonic increases. For the two groups with LRS at 200 kΩ and 50 kΩ, the read nonlinearity becomes more pronounced with increasing input voltage. In the case of the 20 kΩ LRS group, the read nonlinearity shows a tendency to saturate as the input voltage increases. The final read-nonlinearity of 1T1R RRAM is fitted with the polynomial function related to the read voltage.

Subsequently, attention shifts to the read nonlinearity modeling for the current mirror adder. We conducted separate circuit simulations for different ratios of single-stage and two-stage current mirror adders to analyze their read nonlinearity. The explored ratio range spans from $1:1$, $1:2$, up to $1:2^7$, indicating that the proposed Light-CIM can support parallel MAC operations for up to 8-bit weights. For higher precision NN models, a time-multiplexed serial approach can be employed to shift and accumulate results from multiple 8-bit weight sets. The read nonlinearity, defined as the deviation of the actual output relative to the ideal output, is modeled using the power function mathematically as $f_{CMadder}^{single-stage}$ and $f_{CMadder}^{two-stage}$, using as shown in **Fig. 4(b)**, with deviation ranges of 125%∼95% and 130%∼90%, respectively. As the input current increases, the decreasing trend of nonlinearity for both cases gradually slows down.

Following that is the read nonlinearity modeling for the input buffer module. The nonlinearity in the output of the input buffer is defined as the deviation of the actual output voltage from the ideal output voltage. The mathematical model using the power function, denoted as f_{INb}, is presented in **Fig. 4(c)**, and its deviation range spans from 170% to 90%. As the input current increases, the decreasing trend is initially slowed, but after surpassing 25 μA, it intensifies. This suggests that the input buffer has an upper limit in its effective operating range.

Then, the discussion extends to the read nonlinearity modeling for the analog ReLU module. The read nonlinearity of the analog ReLU circuit, defined as the deviation of the actual output current relative to the ideal output current, is characterized by the mathematical model f_{ReLU} using the power function, as depicted in **Fig. 4(d)**. The deviation range spans from 106% to 99%. As the input current increases, the decreasing trend gradually slows down.

Lastly, the discussion encompasses the read nonlinearity modeling for the analog max-pooling module. The read nonlinearity of the basic 2-to-1 analog max-pooling, defined as the deviation of the actual output current relative to the ideal output current, is characterized by the mathematical model f_{MAXp} using the power function, as shown in **Fig. 4(e)**. The deviation range spans from 125% to 95%. As the input current increases, the decreasing trend of the nonlinearity gradually approaches linearity.

Fig. 4(f) presents 4k-point MC simulation results of output currents in the complete analog path, encompassing all analog circuit modules. Since the MC method is a numerical simulation of statistically varying inputs to reproduce stochastic variables preserving the specified distributional property outcomes [21], such as the 3σ (∼99.7%), the MC simulation can offer an objective and comprehensive reflection of circuit behavior under the combined influences of different PVT conditions, variations, and noise. Therefore, we conducted a thorough 4k-point MC simulation on a critical path from input to output in our proposed Light-CIM architecture. Note that the percentage of maximum deviation decreases as the output current increases from 1.5 μA to 60 μA. Since accurate computations involving strong neuron (large activation) and

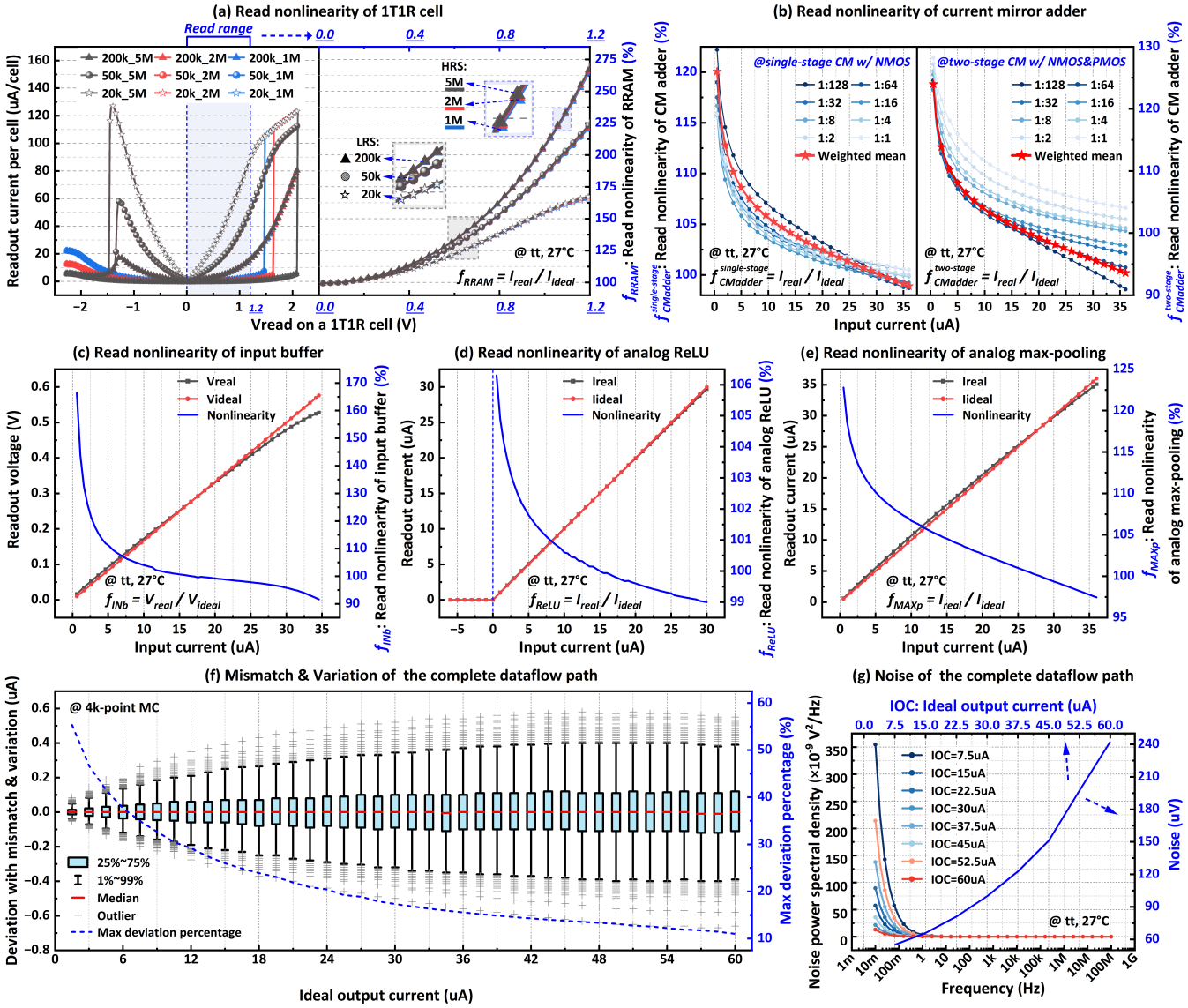

Fig. 4. Non-idealities analysis in the proposed Light-CIM. (a) 1T1R cell read nonlinearity modeling: Left: Transfer characteristic curve (TCC) showing the relationship between input voltage and readout current. Right: Read nonlinearity curve (RNC) indicating disparities between real and ideal readout current. (b) Current mirror adder read nonlinearity modeling: Left (RNC): Single-stage current mirror. Right (RNC): Two-stage current mirror. (c) Input buffer read nonlinearity modeling: Left Y-axis: TCC correlating input current and readout voltage. Right Y-axis: RNC revealing differences between real and ideal readout voltage. (d) / (e) Analog ReLU / max-pooling read nonlinearity modeling: Left Y-axis: TCC depicting input current and readout current relationship. Right Y-axis: RNC showing distinctions between real and ideal readout current. (f) Mismatch and variation modeling: Based on 4k-point MC simulation results of output current in the complete data flow path, covering all analog circuits. (g) Noise modeling: Analyzed through power spectral density (PSD) for noise frequencies from 10^{-2} to 10^{8} HZ.

synaptic (large weight) significantly impact the overall NN model accuracy [24], a larger output current corresponds to a lower error ratio is beneficial for hardware accuracy. Based on the simulation results, we model the mismatch and variation as a normal distribution in which the mean and standard deviation correlate to the outputs.

The circuit noises are also simulated on the whole analog data flow as shown in **Fig. 4(i)**. We sweep the frequency from 10^{-2} to 10^{8} HZ, and integrate the noise power spectral density (PSD) to obtain the circuit noise results under different outputs. The frequency shown on the x-axis in Fig. 4(i) does not refer to the system operating frequency but rather the frequency used in the AC simulation. The purpose of this figure is to illustrate the noise distribution characteristics across different frequency bands, and it is not intended to guide the selection of the system frequency.

V. Evaluation and Discussion

A. Benchmark

To verify the Light-CIM accelerator with the proposed non-ideality-aware training algorithm, we adopt four NN models of different sizes as benchmarks. A four-layer MLP with the synapse numbers 784, 256, 256, and 10 is used for recognizing the MNIST dataset. VGG-8 and VGG-11 are

adopted to recognize the CIFAR-10 dataset. ResNet-18 is used to classify the CIFAR-100 dataset. The activation function used for these networks is ReLU. The max-pooling layer is processed at the end of certain layers in VGG-8, VGG-11, and ResNet-18 to down-sample the feature maps. Note that some unsupported functions like average-pooling and residual block are considered to be performed in the digital domain outside FANTs in Light-CIM.

B. Experimental Setup

To make a fair comparison, the proposed Light-CIM accelerator consists of 168 FANTs, each of which consists of 12 PEs; every PE consists of 8 sub-arrays with a size of 128×128 cells which shares the same hardware configuration with ISAAC [5]. The 1T1R cells adopted in CIM arrays are with different resistance state combinations modeled in **Section IV-B**. The maximum read voltage is set to be 0.5 V or 1 V, depending on the specific test. The analog RRAM model used in the analog buffer is referred to [12]. The variation model of RRAM is extracted from NeuroSIM [14] and is set to $3\sigma = 20\%$. The precision of weight is 8-bit, which is implemented by two groups of 8 1-bit cells representing the positive and negative weight data respectively. The core/IO supply voltage is 0.9V/1.8V and the operation frequency is 10 MHz. The power and area results of the peripheral circuits proposed in this work are obtained based on the circuit design in the 28nm process. Based on the aforementioned configuration and model, we adopted a methodology inspired by the NeuroSIM simulator [14], evaluating the area efficiency and energy efficiency of Light-CIM at the 28nm process.

C. Influence of Non-idealities & Algorithm Effectiveness

The results in **Fig. 5** display how the abovementioned non-idealities harm the hardware performance without the proposed algorithm. In the simulation the synaptic cells with the resistance state combination of 200 kΩ (LRS) and 2 MΩ (HRS) are chosen to prove the effectiveness in the worst case of RRAM read nonlinearity discussed in **Section IV**. At this point, the upper limit for the read voltage is configured to be 0.5 V.

Taking the example of VGG-8 (VGG-11) recognizing CIFAR-10, for NNs trained with traditional algorithms, the ideal accuracy on software reaches 89.72% (90.11%). Upon directly introducing the impact of RRAM variation to the NN model trained with the traditional algorithm, the accuracy sharply drops to 47.95% (49.5%) due to the low robustness of the NN model to hardware non-ideal features. Further introducing the effect of read nonlinearity on the periphery circuits leads to a continued decrease in accuracy to 45.4% (46.71%). Subsequently, introducing the impact of RRAM read nonlinearity further reduces accuracy to 39.55% (42.25%). Finally, considering the complete impact of variation, mismatch, and noise along the analog data path results in a further decrease in accuracy to 32.15% (36.12%). Meanwhile, the comprehensive inclusion of all modeled non-idealities eventually deteriorates accuracy to 97.42% for MLP, 36.12% for VGG-11, and 13.76% for ResNet-18. These results underscore

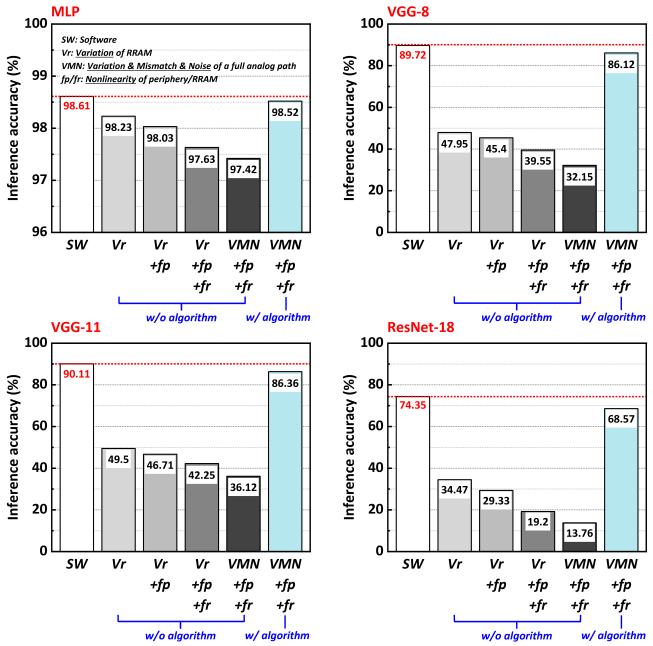

Fig. 5. Influence of RRAM read nonlinearity and peripheral circuit non-ideality on network accuracy (a) MLP for MNIST; (b) VGG-8 for Cifar-10; (c) VGG-11 for Cifar-10; (d) ResNet-18 for Cifar-100.

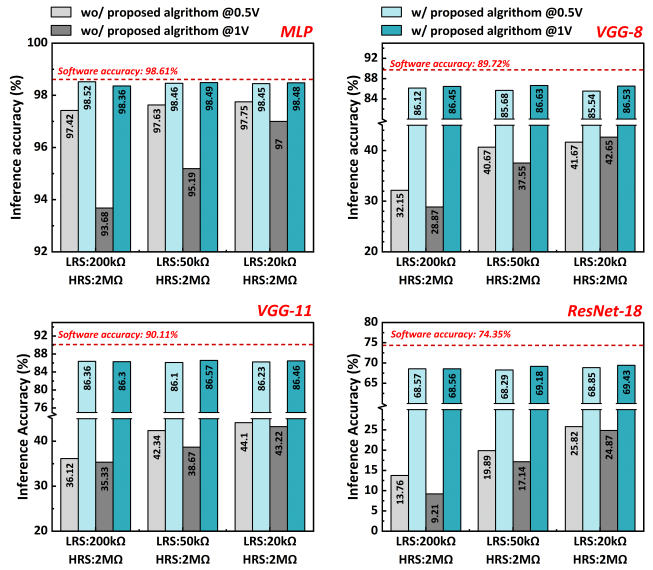

Fig. 6. Influence of different RRAM read nonlinearities on network accuracy (a) MLP for MNIST; (b) VGG-8 for Cifar-10; (c) VGG-11 for Cifar-10; (d) ResNet-18 for Cifar-100.

that, without the aid of our non-ideality-aware algorithm, the performance of NNs operating on the proposed Light-CIM can be significantly compromised. Notably, the accuracy drops in larger NN models are much more pronounced given the same non-idealities. The scenario transforms for NNs operating on hardware after being retrained with our proposed non-ideality-aware algorithm. The accuracy can be restored on hardware to 98.52%, 86.12%, 86.36%, and 68.57% for MLP, VGG-8, VGG-11, and ResNet-18, respectively, approaching the levels observed on the software side.

Note that the accuracy drop on ResNet-18 for CIFAR-100 is 5.78%, which exhibits a small gap compared to the ideal software accuracy. We would like to clarify that the reported accuracy loss in **Fig. 5** occurs in the worst-case scenario, where the weights of the four NNs are assumed to be mapped in a single FANT. The assumption that a single FANT to be large enough to map the whole NN model helps to reveal the upper limit of accuracy loss after recovered by our proposed algorithm. However, in real situations, for larger NNs like VGG-11 and ResNet-18, it is unlikely to map their weights onto one single FANT in most cases, and often requires multiple FANTs to distribute the computations of different layers. The analog data within each FANT is converted to digital data using the DACs and ADCs at the tile interfaces. This digital data is then transmitted over the mesh interconnect to the destination tile, where it is converted back to analog data for further processing. The analog-to-digital conversions between FANTs can avoid cascading errors from multi-layer analog computations, effectively decomposing a deep network into multiple shallow NN layers implemented by analog computation, thus achieving better results according to the number of FANTs used for mapping.

For large models that exceed the capacity of a single tile, Light-CIM leverages a traditional multi-tile mapping approach to ensure scalability. The mapping flow involves partitioning the model across multiple tiles based on the available resources and the model's computational graph. The partitioning process aims to minimize inter-tile communication while maximizing the utilization of intra-tile resources. Once the model is partitioned, each tile is assigned a specific portion of the model to execute.

We further investigate the impact of different RRAM read nonlinearities on NN accuracy, as depicted in **Fig. 6**. The extent of read nonlinearities primarily hinges on the combination of LRS/HRS, as well as the read voltage. As illustrated in **Fig. 4**, the choice of HRS has minimal impact on device behavior; hence, HRS is maintained at a constant value of 2 MΩ. RRAM cells with LRS values of 20 kΩ, 50 kΩ, and 200 kΩ are selected for simulation. The maximum read voltage is set at 0.5 V or 1 V. Other parameters remain consistent, including the variation, mismatch, and noise of the full analog path, as well as the nonlinearity of periphery circuits. **Fig. 6** illustrates that, in comparison to RRAM cells with smaller LRS values, those employing larger LRS values exhibit more pronounced RRAM read nonlinearity. Simultaneously, in the absence of our proposed algorithm, the hardware accuracy tends to increase initially and then decrease with increasing read voltage. In summary, the overall hardware accuracy is influenced by the complex interplay of various non-ideal characteristics. Therefore, the accuracy results reflect combined effects and may not exhibit a strictly linear correlation with a single non-ideal characteristic. Higher read voltages and larger LRS values exacerbate the read nonlinearity of the RRAM-based CIM array.

Furthermore, under different resistance combinations and read voltages, our proposed algorithm consistently demonstrates effective recovery across various hardware conditions and NN benchmarks. This provides a robust foundation for the hardware implementation of the intra-tile fully analog data flow in Light-CIM.

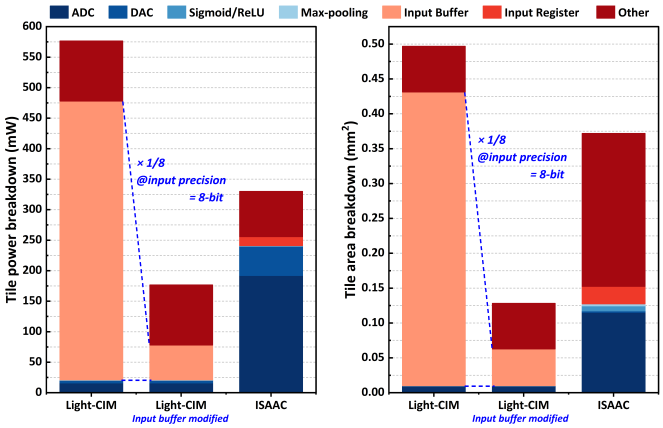

Fig. 7. The FANT breakdowns of (a) power consumption and (b) area consumption.

D. Hardware Evaluation and Comparison

We evaluate the power and area consumptions at the 28nm node, and make an end-to-end tile-level comparison with a CIM architecture adopting the digital process method, as shown in **Fig. 7**. Here ISAAC is chosen because its similar hardware configuration to Light-CIM makes it a great digital counterpart. Notably, while the actual tile (i.e., FANT) power consumption of Light-CIM appears larger than that of ISAAC, a fair assessment must consider that Light-CIM, benefiting from its analog input/output scheme, can achieve n-times higher throughput than ISAAC, given an n-bit input precision. Assuming $n = 8$, Light-CIM can effectively reduce the number of input buffers by $7/8$ while maintaining the same throughput as ISAAC. Consequently, the power/area consumption attributed to Input Buffers can be decreased by approximately 87.5%, denoted as *Light-CIM (input buffer modified)* in **Fig. 7**. This observation underscores that, although digital processing units can mitigate non-idealities, they may introduce ADCs at lower architectural levels, thereby dominating hardware overheads. In contrast, Light-CIM successfully mitigates ADC-related overheads to acceptable levels, thanks to the proposed intra-tile fully-analog data flow.

We provide a detailed comparative analysis of our proposed Light-CIM architecture against several state-of-the-art works in **Table I**. The works [8], [18], and [19] share the common objective of eliminating the need for ADCs through diverse methodologies. FloatPIM, detailed in [8], achieves ADC elimination by employing NOR-based stateful logic, facilitating precise floating-point CNN training. It attains an impressive 298× energy efficiency compared to GPU counterparts. The strategy in [18] adopts ADC-free pulse-width modulation in RRAM-based CIM, resulting in substantial gains in energy efficiency. The work by [19] introduces a 4T1C all-analog MAC cell, achieving a mere 0.67 fJ for a circuit-level MAC operation. Due to the evaluation being confined to the cell-level, [19] achieved the EE of 2989 TOPS/W. CASCADE, as proposed by [15], addresses multi-bit ADC challenges by

TABLE I
COMPARISON WITH STATE-OF-THE-ART WORKS

	ISCA'19 [8]	**MICRO'19** [15]	**VLSI'22** [18]	**TCAS-I'23** [19]	**JSSC'23** [20]	**Light-CIM**
Technology	28nm	65nm	40nm	22nm	28nm	**28nm**
Array structure	**RRAM crossbar**	**RRAM crossbar**	**RRAM 1T1R**	**4T1C MAC cell**	**8T SRAM**	**RRAM 1T1R**
Activation precision (bit)	Float-32 bfloat 16 Fixed-32 Fixed-16	16	Analog	up to 6	1∼8	**Analog**
Weight precision (bit)	Float-32 bfloat 16 Fixed-32 Fixed-16	16	2	up to 6	1/4/8	**1∼8**
Sensing scheme	-	TIA+ADC	PWM Conversion	-	Comparator	**Tile-interface ADC**
Sensing presicion (bit)	-	10	-	-	1	**8**
ADC optimization scheme	Eliminated	Macro-level eliminated	Eliminated	Eliminated	Replaced	**Tile-level eliminated**
EE (TOPS/W)	0.7∼0.82	2.2	26.97	2989 (cell-level)[*i]	310.4	3.08
AE (TOPS/mm^2)	0.3∼2.39	0.1	22.5	272/332 (cell-level)[*i]	546.1	3.91
Normalized EE (TOPS/W)[*ii]	179.2∼209.9	563.2	431.5	2989 (cell-level)[*i]	310.4	**394.2**
Normalized AE (TOPS/mm^2)[*ii]	76.8∼611.8	25.6	360	272/332 (cell-level)[*i]	546.1	**500.5**
FoM[*iii]	13,763∼128,417	14,418	155,340	813,008/992,348 (cell-level)[*i]	169,509	**197,297**

[*i] Taking only the analog computing cell into account, not including complete system performance;
[*ii] Normalized to 1-bit activation and 1-bit weight. 'Analog' is calculated as 16-bit;
[*iii] FoM = Normalized EE × Normalized AE.

leveraging robust analog data flow and analog RRAM buffers. While similar to Light-CIM in eliminating ADCs, CASCADE implements macro-level ADC elimination, resulting in high energy efficiency (2.2 TOPS/W) but relatively lower area efficiency (101 GOPS/mm^2) at 16-bit activations/weights. Neuro-CIM, presented in [20], supports SNN dataflow, incorporating early stopping schemes and integrating analog and digital networks for neuromorphic computing. Neuro-CIM exploits the intrinsic features of SNN, replacing ADCs with 1-bit comparators, achieving both high energy and area efficiency. The proposed Light-CIM is evaluated to achieve an energy efficiency of 3.08 TOPS/W and an area efficiency of 3.91 TOPS/mm^2 at 8-bit weights and analog inputs. Importantly, these metrics are competitive with the aforementioned state-of-the-art works. Notably, Light-CIM and Float-PIM are assessed at the processor chip level, while other works are evaluated at the macro or PE level, without considering chip-level data communication overheads.Furthermore, compared to the other five works, the proposed Light-CIM is the only one employing the non-ideality-aware algorithm for hardware-software co-optimization. In comparison to the work [18], which also evaluated hardware accuracy for ResNet-18/CIFAR-100, our work demonstrates a lower accuracy loss.

VI. CONCLUSION AND FUTURE WORK

In this work, we proposed an ADC/DAC-fewer CIM accelerator Light-CIM with customized analog circuits implementing different functions, for NN acceleration in consumer electronics. The intra-tile fully analog data flow reduces DACs and ADCs, significantly improving the area and energy efficiencies. Although analog computing is sensitive to non-idealities in emerging memory devices and analog circuits, our proposed training algorithm helps to recover the accuracy loss. The work scheme of Light-CIM has been verified on MLP, VGG-8, VGG-11, and ResNet-18. We would like to extend the fully analog CIM application to on-chip training or deeper NNs for more complex benchmarks in the future.

REFERENCES

[1] P. Chi, S. Li, C. Xu, T. Zhang, J. Zhao, Y. Liu, Y. Wang, and Y. Xie, "PRIME: A Novel Processing-in-Memory Architecture for Neural Network Computation in ReRAM-Based Main Memory," *ACM SIGARCH Computer Architecture News*, vol. 44, no. 3, pp. 27–39, 2016.

[2] W.-H. Chen, K.-X. Li, W.-Y. Lin, K.-H. Hsu, P.-Y. Li, C.-H. Yang, C.-X. Xue, E.-Y. Yang, Y.-K. Chen, Y.-S. Chang *et al.*, "A 65nm 1Mb nonvolatile computing-in-memory ReRAM macro with sub-16ns multiply-and-accumulate for binary DNN AI edge processors," in *2018 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2018, pp. 494–496.

[3] C.-X. Xue, W.-H. Chen, J.-S. Liu, J.-F. Li, W.-Y. Lin, W.-E. Lin, J.-H. Wang, W.-C. Wei, T.-W. Chang, T.-C. Chang *et al.*, "24.1 A 1Mb Multibit ReRAM Computing-In-Memory Macro with 14.6ns Parallel MAC Computing Time for CNN Based AI Edge Processors," in *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2019, pp. 388–390.

[4] C.-X. Xue, T.-Y. Huang, J.-S. Liu, T.-W. Chang, H.-Y. Kao, J.-H. Wang, T.-W. Liu, S.-Y. Wei, S.-P. Huang, W.-C. Wei *et al.*, "15.4 A 22nm 2Mb ReRAM Compute-in-Memory Macro with 121-28TOPS/W for Multibit MAC Computing for Tiny AI Edge Devices," in *2020 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2020, pp. 244–246.

[5] A. Shafiee, A. Nag, N. Muralimanohar, R. Balasubramonian, J. P. Strachan, M. Hu, R. S. Williams, and V. Srikumar, "ISAAC: A Convolutional Neural Network Accelerator with In-Situ Analog Arithmetic in Crossbars," *ACM SIGARCH Computer Architecture News*, vol. 44, no. 3, pp. 14–26, 2016.

[6] L. Song, X. Qian, H. Li, and Y. Chen, "PipeLayer: A Pipelined ReRAM-Based Accelerator for Deep Learning," in *2017 IEEE international symposium on high performance computer architecture (HPCA)*. IEEE, 2017, pp. 541–552.

[7] M. Imani, S. Gupta, Y. Kim, M. Zhou, and T. Rosing, "DigitalPIM: Digital-based Processing In-Memory for Big Data Acceleration," in *Proceedings of the 2019 on Great Lakes Symposium on VLSI*, 2019, pp. 429–434.

[8] M. Imani, S. Gupta, Y. Kim, and T. Rosing, "FloatPIM: In-Memory Acceleration of Deep Neural Network Training with High Precision," in *2019 ACM/IEEE 46th Annual International Symposium on Computer Architecture (ISCA)*. IEEE, 2019, pp. 802–815.

[9] Q. Zheng, Z. Wang, Z. Feng, B. Yan, Y. Cai, R. Huang, Y. Chen, C.-L. Yang, and H. H. Li, "Lattice: An ADC/DAC-less ReRAM-based Processing-In-Memory Architecture for Accelerating Deep Convolution Neural Networks," in *2020 57th ACM/IEEE Design Automation Conference (DAC)*. IEEE, 2020, pp. 1–6.

[10] H. Kim, Y. Kim, S. Ryu, and J.-J. Kim, "Algorithm/Hardware Co-Design for In-Memory Neural Network Computing with Minimal Peripheral Circuit Overhead," in *2020 57th ACM/IEEE Design Automation Conference (DAC)*. IEEE, 2020, pp. 1–6.

[11] P.-Y. Chen and S. Yu, "Compact Modeling of RRAM Devices and Its Applications in 1T1R and 1S1R Array Design," *IEEE Transactions on Electron Devices*, vol. 62, no. 12, pp. 4022–4028, 2015.

[12] D. Liu, H. Cheng, X. Zhu, G. Wang, and N. Wang, "Analog Memristors Based on Thickening/Thinning of Ag Nanofilaments in Amorphous Manganite Thin Films," *ACS applied materials & interfaces*, vol. 5, no. 21, pp. 11 258–11 264, 2013.
[13] H.-S. P. Wong, H.-Y. Lee, S. Yu, Y.-S. Chen, Y. Wu, P.-S. Chen, B. Lee, F. T. Chen, and M.-J. Tsai, "Metal-oxide RRAM," *Proceedings of the IEEE*, vol. 100, no. 6, pp. 1951–1970, 2012.
[14] P.-Y. Chen, X. Peng, and S. Yu, "NeuroSim: A Circuit-Level Macro Model for Benchmarking Neuro-Inspired Architectures in Online Learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 12, pp. 3067–3080, 2018.
[15] T. Chou, W. Tang, J. Botimer, and Z. Zhang, "CASCADE: Connecting RRAMs to Extend Analog Dataflow In An End-To-End In-Memory Processing Paradigm," in *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, 2019, pp. 114–125.
[16] J.-O. Seo, M. Seok, and S. Cho, "ARCHON: A 332.7TOPS/W 5b Variation-Tolerant Analog CNN Processor Featuring Analog Neuronal Computation Unit and Analog Memory," in *2022 IEEE International Solid-State Circuits Conference (ISSCC)*, vol. 65. IEEE, 2022, pp. 258–260.
[17] J. Yang, D. Chen, Q. Ding, J. Fang, X. Xue, H. Lv, X. Zeng, and M. Liu, "A Novel PUF Using Stochastic Short-Term Memory Time of Oxide-Based RRAM for Embedded Applications," in *2020 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2020, pp. 39–2.
[18] H. Jiang, W. Li, S. Huang, and S. Yu, "A 40nm Analog-Input ADC-Free Compute-in-Memory RRAM Macro with Pulse-Width Modulation between Sub-arrays," in *2022 IEEE Symposium on VLSI Technology and Circuits (VLSI Technology and Circuits)*, pp. 266–267, IEEE, 2022.
[19] R. Nägele, J. Finkbeiner, V. Stadtlander, M. Grözing, and M. Berroth, "Analog Multiply-Accumulate Cell With Multi-Bit Resolution for All-Analog AI Inference Accelerators," *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2023.
[20] S. Kim, S. Kim, S. Um, S. Kim, K. Kim, and H.-J. Yoo, "Neuro-CIM: ADC-Less Neuromorphic Computing-in-Memory Processor With Operation Gating/Stopping and Digital–Analog Networks," *IEEE Journal of Solid-State Circuits*, 2023.
[21] J.-F. Huang, V. C. Chang, S. Liu, K. Y. Doong, and K.-J. Chang, "Modeling Sub-90nm On-Chip Variation Using Monte Carlo Method for DFM," in *2007 Asia and South Pacific Design Automation Conference*, pp. 221–225, IEEE, 2007.
[22] A. Sebastian, M. Le Gallo, R. Khaddam-Aljameh, and E. Eleftheriou, "Memory Devices and Applications for In-Memory Computing," *Nature nanotechnology*, vol. 15, no. 7, pp. 529–544, 2020.
[23] Y. Li, J. Tang, B. Gao, J. Yao, A. Fan, B. Yan, Y. Yang, Y. Xi, Y. Li, J. Li *et al.*, "Monolithic Three-Dimensional Integration of RRAM-Based Hybrid Memory Architecture for One-Shot Learning," *Nature Communications*, vol. 14, no. 1, p. 7140, 2023.
[24] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.
[25] T. Song, X. Chen, X. Zhang, and Y. Han, "Brahms: Beyond conventional rram-based neural network accelerators using hybrid analog memory system," in *2021 58th ACM/IEEE Design Automation Conference (DAC)*. IEEE, 2021, pp. 1033–1038.
[26] H. Zhu, K. Zhu, J. Gu, H. Jin, R. T. Chen, J. A. Incorvia, and D. Z. Pan, "Fuse and mix: Macam-enabled analog activation for energy-efficient neural acceleration," in *Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design*, 2022, pp. 1–9.

Chenyang Zhao received the B.S. degree from North University of China, Taiyuan, China, in 2019. She is currently pursuing a Ph.D. degree in the State Key Laboratory of ASIC and Systems, at Fudan University, Shanghai, China. Her current research interest is in the chip design of neural network accelerators based on the computing-in-memory architecture.

Jinbei Fang received the B.S. degree from Fudan University, Shanghai, China, in 2020. He is currently pursuing a Ph.D. degree in the State Key Laboratory of ASIC and Systems at Fudan University. His current research interests focus on computing-in-memory neural networks accelerators.

Jingwen Jiang received the B.S. degree from Fudan University, Shanghai, China, in 2021. She is currently pursuing a Ph.D. degree with the State Key Laboratory of ASIC and Systems at Fudan University. Her current research interests include computing-in-memory architecture and neuromorphic circuits and systems.

Xiaoyong Xue (M'12) received a Ph.D. degree in microelectronics from Fudan University, Shanghai, China, in 2011. He joined the Department of Microelectronics, at Fudan University, as a postdoctoral research fellow. He is now an associate professor at Fudan University. His research interests include high-performance memory/storage, in-memory computing circuits, and systems.

Xiaoyang Zeng (M'07) received the B.Sc. degree from Xiangtan University, Xiangtan, China, in 1996, and the Ph.D. degree (Hons.) from the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, in 2001. He is the Chair Professor and Executing Director of the State-Key Laboratory of ASIC and System. His research fields include information security chips, baseband processing technologies for wireless communication, mixed-signal IC designs, and ultra-low power IC methodology.