

A near-threshold memristive computing-in-memory engine for edge intelligence

Received: 19 August 2024

Accepted: 11 June 2025

Published online: 01 July 2025

 Check for updates

Linfang Wang ^{1,2,7}, Weizeng Li ^{1,3,7}, Zhidao Zhou^{1,3,7}, Junjie An¹, Wang Ye^{1,3}, Zhi Li ^{1,3}, Hanghang Gao^{1,3}, Hongyang Hu^{1,3}, Jing Liu^{1,3}, Xiaoming Chen ⁴, Ling Li ⁵, Qi Liu ^{1,6}, Mingoo Seok², Chunmeng Dou ^{1,3} ✉ & Ming Liu^{1,6}

Memristive computing-in-memory and near-threshold computing are two unconventional computing paradigms that can potentially enhance the energy efficiency and real-time performance of edge devices. However, their scalability faces challenges, primarily due to process variation. Here, we report a 1-Mb, 16-macro near-threshold memristive computing-in-memory engine. The two-transistor-one-resistor cells provide strong cell current modulation capability with more than 120-times amplified resistance ratio. To mitigate variation issues, we compensate for transistor mismatches by leveraging the intrinsic variations in memristors. Additionally, we propose a charge stacking technique between multiple analog-to-digital converters to perform analog weight-and-combine operations with small energy and area overhead. Moreover, we introduce an inter-macro hybrid control scheme to reduce the task-level inference power. The fabricated chip can perform highly parallel analog computing over 256 input channels with a 2.4% relative standard deviation. It achieves a throughput up to 10.49 tera-operations per second and an energy efficiency up to 88.51 tera-operations per second per watt.

Edge hardware for artificial intelligence (AI) has experienced unprecedented growth, driven by the urgent need to reduce communication latency, improve energy-efficiency (EF), strengthen security and safeguard privacy in order to deliver high-quality AI services^{1–4}. Deploying AI on edge not only requires high EF to satisfy the stringent power constraints but also calls for high parallelism to improve the real-time performance. Memristive computing-in-memory (mCIM)^{5–46} offers a competent platform to enable edge intelligence. Particularly, its capability of high-density storage can enable to have all weights on the chip^{9–13} to eliminate the off-chip data traffic between on-chip cache memory and off-chip dynamic random-access memory^{14–17}. On the other hand, it can minimize large stand-by power required by the conventional volatile cache memory.

Previous works^{13,21–23} have made significant progress in designing and prototyping mCIM chips. However, enhancing computing parallelism remains an open research question due to the challenges posed by multiple aspects. First of all, there is a dilemma between the input parallelism and the computing accuracy in the analog multiply-and-accumulate (MAC) array. For example, the number of input parallelism for analog accumulation (N_{IN}) in the reported mCIM is relatively limited and usually no larger than $16^{12,20–22,27,33}$. This is because the MAC error quickly increases as N_{IN} increases, due to various device non-idealities^{21,27,36}, such as the IR drop due to the large summation current, the non-negligible high-resistive-state (HRS) leakage caused by the limited resistance-ratio (R-ratio) between the HRS and the low-resistive-state (LRS), and the accumulation of device-to-device variations in the computing results. Some previous works that can

¹State Key Lab of Fabrication Technologies for Integrated Circuits, Institute of Microelectronics of the Chinese Academy of Sciences, Beijing, China.

²Department of Electrical Engineering, Columbia University, New York, NY, USA. ³University of Chinese Academy of Sciences, Beijing, China. ⁴Institute of Computing Technology of the Chinese Academy of Sciences, Beijing, China. ⁵Institute of Software of the Chinese Academy of Sciences, Beijing, China.

⁶Frontier Institute of Chip and System, Fudan University, Shanghai, China. ⁷These authors contributed equally: Linfang Wang, Weizeng Li, Zhidao Zhou.

✉ e-mail: douchunmeng@ime.ac.cn

overcome the N_{IN} limitation, either by co-optimization of memristive devices with large R-ratios⁴¹ and high resistances⁴⁹ or by co-design with noise-tolerant algorithms⁵⁶, are summarized in Supplementary Table 1. Secondly, increasing the number of output parallelism (N_{OUT}) is hindered by the large hardware costs associated with the analog readout circuitry. As the readout precision increases, the energy and area cost of high-precision analog readout circuits exponentially increases⁴⁷. Also, additional analog processing circuits^{21,22,31,32,48,49} are required to weight-and-combine (WAC) the analog signals for multi-bit MAC computation. The energy and area consumptions of these circuits limit the maximum number of output ports that can be placed within a single macro. Lastly, integrating multiple macros can increase the parallelism at the system level. It is of importance to develop energy-saving techniques for the scalable CIM system, considering the limited power budget. Previous works have investigated the model-to-hardware mapping strategies and the dataflow control for multi-macro systems^{23,41,57,58}. In particular, prior architectural studies have demonstrated that the mixed precision approach can play a vital role in enabling highly efficient mCIM chips^{57,58}. However, hardware-centric studies of mixed precision multi-macro mCIM with low hardware cost and experimental validation remain challenging. On the other hand, the near-threshold (NVT) computing is another computing paradigm that can potentially play a key role in edge device. It can effectively reduce the power consumption by reducing the supply voltage (VDD). The performance loss due to reduced VDD can be regained through improving the system parallelism by adding more computing components^{50–52}. Despite these advantages, a long-standing challenge faced by the NVT computing is its high sensitivity to process variations, which can lead to unpredictable performance loss and reliability issues. The possibility of combining with mCIM remains unexplored. Transistors, which are used as selectors in the conventional one-transistor-one-resistor (1T1R) memristive array, need to be fully activated to readout the memristive resistance properly⁵³.

In this article, we combine the mCIM and NVT computing paradigms for highly energy-efficient and parallel computing. Specifically, we propose an NVT-operated two-transistor-one-resistor (2T1R) cell array. The NVT-operated transistors can modulate the cell current more strongly and thereby increasing the R-ratio of memristors. However, it also increases the variability in cell current. Therefore, we propose a mismatch-canceling scheme, where we use the memristor's variations to compensate for the transistor's threshold-voltage (V_{TH}) mismatch. We also propose an analog-to-digital converter (ADC) that can perform data conversion and analog WAC by reusing only one set of the capacitors, reducing the cost for analog readout and thereby increasing the room for greater parallelism. Last but not least, we propose an inter-macro control scheme that uses different WAC and quantizing approaches in different macros to lower the task-level inference power of a multi-macro system, which can help to extend its scalability under stringent energy constraints.

On this basis, we have verified a 1-Mb 16-macro NVT mCIM engine on silicon. The experiments demonstrate that it can perform highly efficient and parallel analog computing over a large number of input channels (256) with a small average relative standard deviation of 2.4%. The chip delivers a peak normalized throughput (TP) up to 10.49 tera-operations per second (TOPS) and achieves an optimal EF from 55.21 to 88.51 TOPS per watt (TOPS/W). It also demonstrates a higher level of computing parallelism compared to the earlier works. This work demonstrates a Mb-level NVT mCIM engine, emphasizing on the promise of combining mCIM and NVT computing for next-generation AI hardware with low power and high parallelism.

Results

Overall structure of the computing-in-memory engine

Fig. 1a shows the die photo of the 1 Mb NVT 2T1R mCIM engine, which is fabricated using a standard 180 nm logic process with back-end-of-

line (BEOL) integrated memristor cells. The typical cell properties and array-level characterization results can be found in the Supplementary Fig. 1. It contains 16 NVT mCIM macros, 96 kb SRAM buffer, I/O interface, and top control. Each mCIM macro has 64 b 2T1R cells, 16 8-bit (b) built-in sample-and-stack ADCs (BSS-ADCs) with pipelined developing and quantizing (PDQ) clampers, memory periphery circuits, and digital circuits for local control and computing. In the computing mode, the entire 2T1R cell array operates at the NVT region, which leads to reduced operational voltage, lowered summation current, and amplified signal ratio between HRS and LRS. More details about the circuit implementation of NVT mCIM engine can be found in Supplementary Fig. 2.

The proposed NVT mCIM engine has the following features. Firstly, Fig. 1b conceptually shows the threshold-voltage mismatch canceling (V_{TH} -MC) programming scheme. While the superposition of memristor variation and transistor V_{TH} mismatch tends to exacerbate the cell current fluctuations, the V_{TH} -MC scheme can reverse correlate the two different variation sources to reduce the overall variability of the cell currents. Secondly, Fig. 1c illustrates the proposed BSS-ADCs. While conventional macros perform WAC, sampling, and quantizing with separate components, the proposed BSS-ADC can carry out all three functions using its internal capacitive array to reduce the energy and area costs. Moreover, its energy and area overheads for WAC operations increase only linearly with the WAC precision (N_{WAC}), rather than the exponential increase of the conventional approaches. Besides, by pipelining the developing and quantizing phase, the overall latency can be reduced by the proposed PDQ clamper. Lastly, as shown in Fig. 1d, different from the conventional multi-macro CIM systems that applying the same accumulation and quantizing approach for all macros, the proposed inter-macro hybrid control (IM-HC) scheme can lower the parallel processing power without compromising inference accuracy by mixing the high precision (HP) and the high efficiency (HE) modes for precision- and efficiency-oriented workloads, respectively. These key concepts at different levels from array, macro, to system collaboratively enable the large-scale NVT mCIM engine.

2T1R cell array with mismatch canceling

We propose the NVT 2T1R cell array to combine the NVT computing with the memristive array. Unlike the conventional 1T1R cell having one IO transistor (T1) and one memristor, the proposed 2T1R cell incorporates one additional core transistor (T2), as shown in Fig. 2a. In the memory operation, T1 is used as the selector for read and write through the source-line and bit-line (BL) like the conventional 1T1R cell, and T2 is disabled by grounding its source and drain. In the computing operation, T2 is used to amplify its gate voltage depending on the voltage division between T1 and the memristor. By this means, the resistive switching of memristors is converted into the threshold voltage shift in the 2T1R cells, which leads to strong capabilities of current modulation and signal amplification.

By operating T2 in the NVT region, the 2T1R cell can exponentially modulate the transpose bit-line (TBL) current (I_{TBL}) as a function of the memristor resistance (R_{CELL}). Fig. 2b shows the comparison between I_{BL} and I_{TBL} as a function of the word-line (WL) voltage (V_{WL}). It shows that the HRS and LRS can be differentiated without fully activating T1 in the 2T1R cell because T1 is used as a voltage divider. Compared with the 1T1R counterpart, the 2T1R cell can achieve more than 98% reduced on-state cell current and 47% reduced V_{WL} . Furthermore, by optimizing the WL and BL biasing, the current ratio of I_{TBL} between HRS and LRS can be improved by more than 120 times compared to that of I_{BL} , as shown in Fig. 2c. Consequently, the 2T1R cell can effectively suppress the analog MAC errors caused by the small signal ratio. More details about the NVT computing method of 2T1R cell array and its effect on reduced MAC errors can be found in Supplementary Fig. 3. Additionally, since T2 operates in the saturation region in the activated 2T1R cell, the cell current is insensitive to clamping voltage fluctuations,

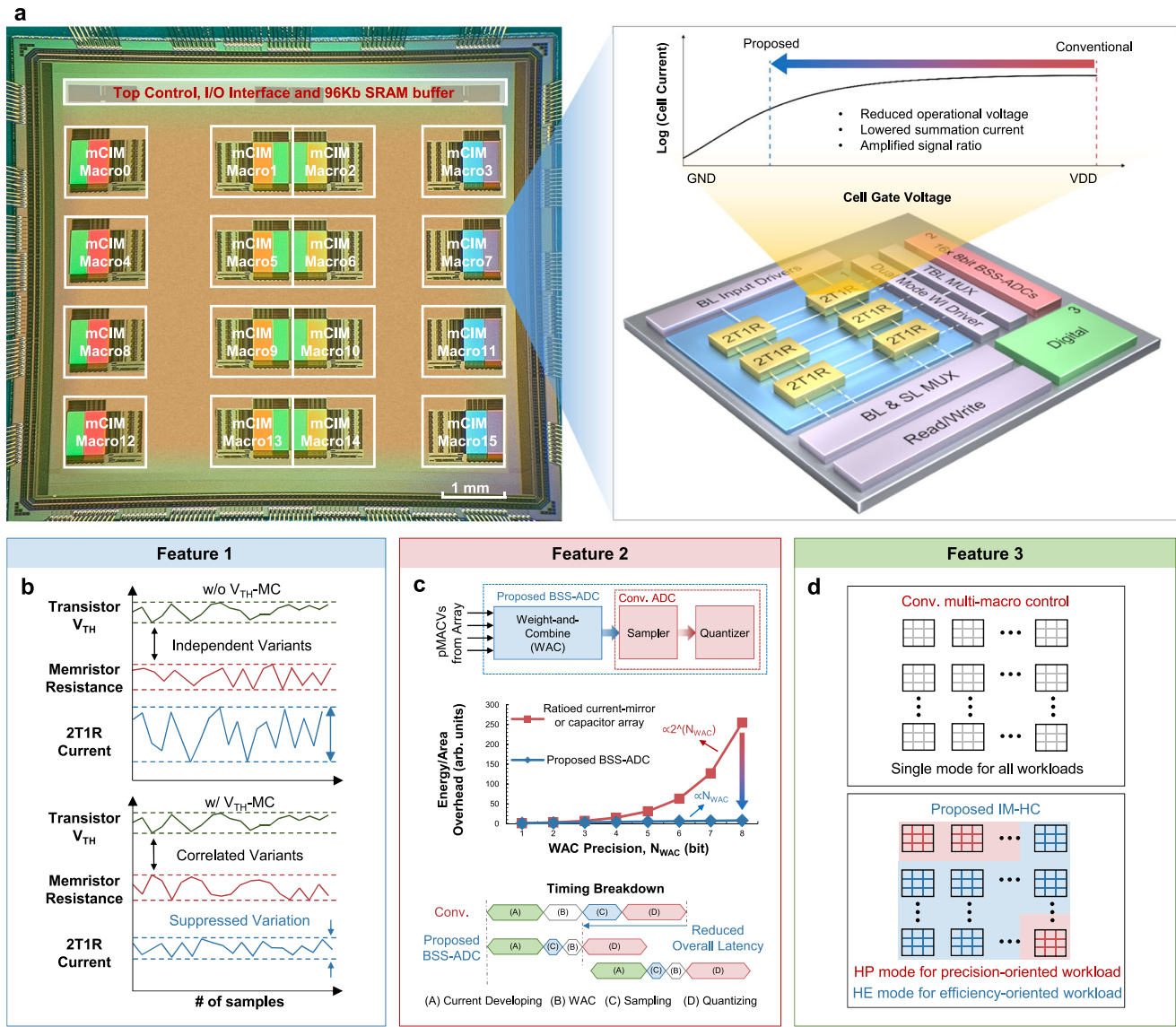


Fig. 1 | Proposed 1-Mb 16-Macro NVT mCIM engine and illustrations of the key enabling concepts. a Die micrograph, macro structure and the NVT operational scheme (Scale bar: 1 mm). **b** V_{TH} -MC scheme that reduces cell current fluctuations by establishing correlation between transistor V_{TH} and memristor resistance. **c** BSS-

ADC that supports WAC, sampling and quantizing with reduced energy/area overhead and latency compared to the conventional ones. **d** IM-HC that sets different macros independently to HP- or HE-mode to process precision- or energy-orientated workloads.

enabling the use of a low-cost static clamping scheme. A detailed analysis of the impact of different clamping schemes can be found in Supplementary Fig. 4.

Another critical challenge faced in realizing a large-scale NVT analog mCIM chip is that cell current variations tend to increase further due to the presence of memristor variations and transistor mismatch. To deal with this issue, we propose the V_{TH} -MC programming scheme along with the 2T1R cell to cancel out the influences induced by the variations of cell resistance and V_{TH} mismatch. Fig. 2d shows the V_{TH} -MC flow, including the coarse- and fine-tuning steps. The coarse-tuning step is used to search the optimal set condition for the target cell current (I_{TARGET}). It performs multiple reset-and-set (R/S) cycles with an incremental compliance current (I_C) in the set process till the minimum cell current deviation (ΔI), defined by the difference between I_{TARGET} and measured I_{TBL} , is less than 30% of I_{TARGET} . The fine-tuning step is used to compensate for the V_{TH} mismatch using the intrinsic cycle-to-cycle (C2C) variations of the memristor. It keeps carrying out R/S cycles using the identical voltage pulses with the optimal set condition determined by the previous step, till ΔI is less

than 10% of I_{TARGET} . More details about the operational flow of the V_{TH} -MC scheme, the modulations of the cell resistances and the retention characteristics in the resistance tuning process can be found in Supplementary Fig. 5. Unlike previous iterative programming schemes, the proposed V_{TH} -MC scheme leverages intrinsic C2C variations in memristors to compensate for transistor mismatch. Consequently, the presence of C2C variation effects does not necessarily increase the number of iteration programming cycles in the proposed scheme. A comparison with several previous iterative programming methods^{15,41,43,59} can be found in Supplementary Table 2. The dependency of the proposed 2T1R cell current on temperature is shown in Supplementary Fig. 6, which provides a guideline for mitigating the influence of temperature by adjusting the operational voltages or the readout circuits of the mCIM chip.

Fig. 2e shows the I_{TBL} versus V_{BL} of 30 2T1R cells at LRS and HRS using different programming schemes where a clear memory window can be observed between HRS and LRS in both cases. The relative standard deviation (σ/μ) of the on-state I_{TBL} current can be lowered from 41.5 to 5.43% after applying the V_{TH} -MC scheme. Fig. 2f further

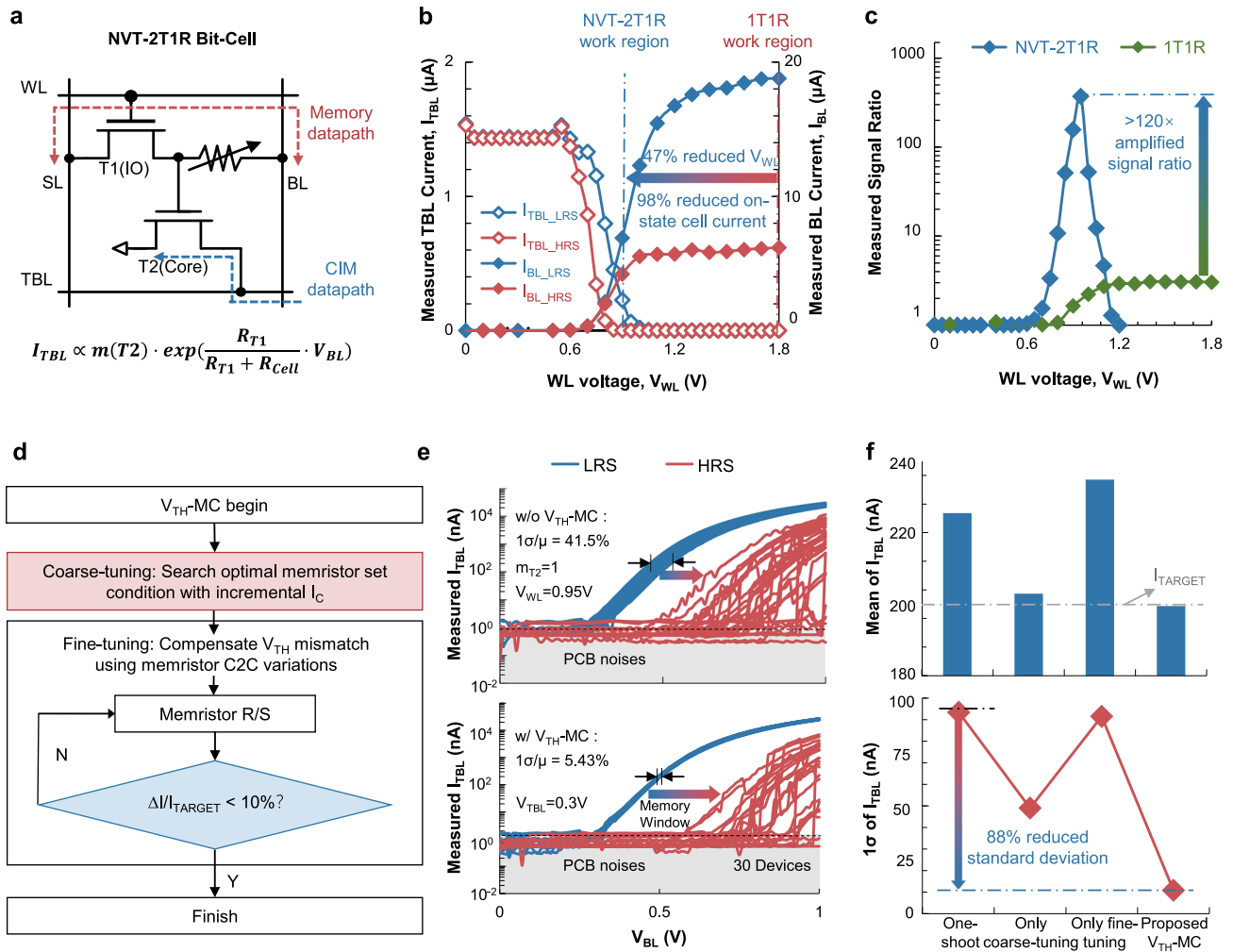


Fig. 2 | 2T1R NVT computing cell properties and the mismatch-canceling scheme. a Bit-cell structure and its principle. **b** Measured BL and TBL current versus WL voltage at HRS and LRS. **c** Measured signal ratio of BL and TBL current. Signal ratio is improved by more than 120 times compared to that on BL. **d** Operational flow of the proposed V_{TH} -MC scheme including coarse- and fine-tuning steps.

e Measurements of TBL current versus BL voltage on 30 2T1R cells in the cell array before and after performing the V_{TH} -MC scheme. **f** Comparison on the mean and standard deviation values of TBL current with different programming schemes, including one-shot, coarse-tuning, fine-tuning and V_{TH} -MC. R_{T1} , resistance of T1; R_{Cell} , resistance of memristor; I_{TARGET} , target cell current.

compares the influences of different programming steps in terms of the mean value and standard deviation (σ) of the I_{TBL} distribution at LRS. It can be seen that the coarse-tuning step can contribute to lowered the ΔI , and the fine-tuning step can further suppress the σ , though the trimming range of fine-tuning step is limited. The combination of coarse- and fine-tuning steps can lead to 98% reduced ΔI and 88% reduced σ compared to the conventional one-shot programming with fixed biasing conditions.

Efficient analog processing and readout circuits

We propose the BSS-ADC with PDQ clamper to reduce the costs for analog processing and readout, as shown in Fig. 3a. The PDQ clamper consists of a cascade current mirror and three n-type transistors (N1, N2, and N3), including N1 for voltage clamping (V_{CLP}), N2 for current sampling (SAM), and N3 for voltage reset (RST). It is worth noticing that incorporating N2 allows the BSS-ADC to perform quantization without being influenced by the current change in the array after sampling. The BSS-ADC consists of a binary-weighted digital-to-analog converters (DAC) array controlled by the common-mode voltage (V_{CM}), high- and low-reference-voltage ($V_{REF,H}$ and $V_{REF,L}$), a comparator, and modified successive approximation register (SAR) logics. Unlike conventional SAR ADC, two additional switches (SW1 and SW2) are introduced at the bottom plate and top plate of the MSB capacitor

(C_{MSB}) in the DAC array. The BSS-ADC can perform the WAC operation of partial charges by adding these two switches with a small additional area overhead.

The readout process of BSS-ADC, with and without WAC, is illustrated by its waveform (Fig. 3b) and configuration (Fig. 3c). A typical readout process can be divided into the current developing phase (PO), the sampling phase (PI) and the quantizing phase (P2). In the standard readout process without WAC, the BSS-ADCs operate independently. In PO, the SAM and RST of the PDQ clamper are kept off during the I_{TBL} developing, which allows for pipelining of PO and P2 in different clock (CLK) cycles to reduce the overall latency; In PI, SW1 is turned off and SW2 is turned on. The RST and SAM are sequentially activated to reset the voltage on the top plates and directly sample I_{TBL} using the DAC array, respectively; In P2, the comparator is triggered to quantize the sampled voltage (V_{SAM}) by the asynchronous SAR logics for the 8-bit CIM outputs (COUT[7:0]). In the readout process with WAC, four BSS-ADCs are operated as a group. The operation is similar to the standard operation except the differences as follows. In PI, both SW1 and SW2 are turned on after sampling. This enables the WAC of four V_{SAM} ($V_{SAM0} - V_{SAM3}$) in different ADCs (ADC0 - ADC3) by leveraging the charge stacking. As a result, V_{SAM3} equals $V_{SAM3}/2 + V_{SAM2}/4 + V_{SAM1}/8 + V_{SAM0}/16$. Notice that in order to scale the V_{SAM3} to the same ADC input range, the sampling time of BSS-ADC with WAC is

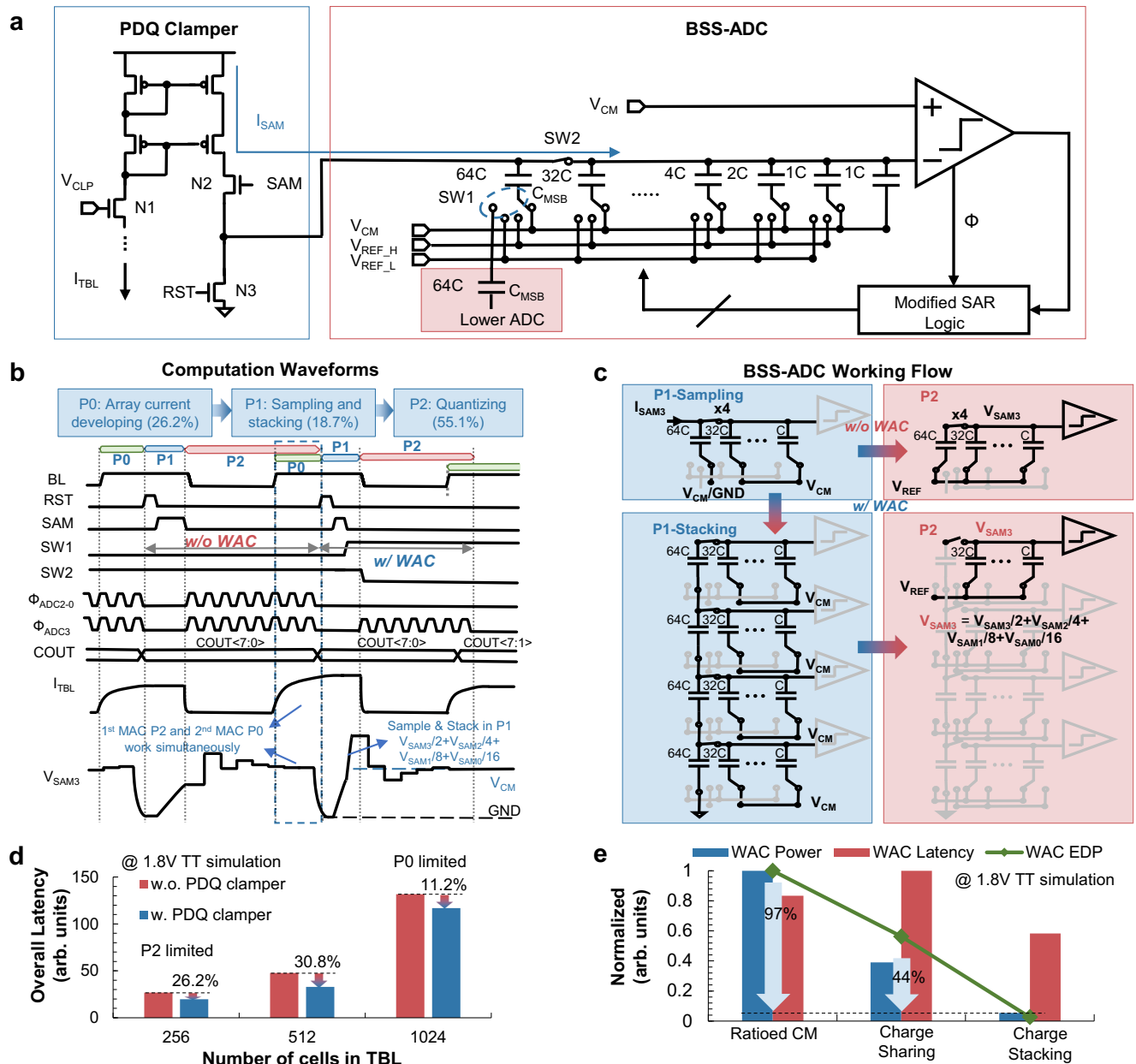


Fig. 3 | Structure and operations for proposed BSS-ADC. a Schematic of the BSS-ADC with a PDQ clumper. **b** Operational waveforms of the different readout phases. The time proportion of each phase is also labeled in the figure. **c** Illustration of the BSS-ADC configurations in sampling and quantizing phases. The analog WAC is achieved by capacitors charge stacking. **d** The impact of the PDQ clumper on the

overall ADC readout latency over different numbers of cells in TBL. **e** Comparison on the power, latency and EDP for the WAC operation between the conventional approaches (ratioed CMs or capacitors) and the proposed capacitors charge stacking.

reduced to the half of that in the case without WAC. In P2, SW2 is turned off to maintain the V_{SAM3} and only the comparator in ADC3 is triggered for the 7-bit CIM outputs (COUT[7:1]).

Fig. 3d shows the performance evaluations of the PDQ clumper. By pipelining the P0 and P2 phases, the overall readout latency can be reduced from 11.2 to 30.8% depending on the array size. The effect of latency reduction is maximized when the durations of P0 and P2 are similar. Fig. 3e further compares different WAC schemes in terms of WAC power, latency and energy-latency-product (EDP). More details about the comparison between different WAC schemes^{31,48,60} can be found in Supplementary Fig. 7. The conventional ratioed current mirrors (CMs) approach tends to incur a large WAC power consumption due to the introduction of additional current duplication branches. The conventional charge sharing approach shows relatively increased

WAC power and latency compared to the proposed charge stacking approach because it usually requires redundant compensation capacitors to form the sampling capacitors. Consequently, the proposed charge stacking approach achieves 97 and 44% reduced WAC EDP compared to the previous approaches based on ratioed CMs and capacitors charge sharing, respectively. Supplementary Note 1 provides additional details on the latency and power data from Fig. 3d, e.

Scalable multi-macro-based computing-in-memory system
The inter-macro collaborative strategy to lower system power is of importance to enable highly scalable mCIM system. Leveraging the features of BSS-ADC, we propose the IM-HC schemes that hybridizes different working modes, namely the HP and HE modes, for different workloads in AI models. The dataflows for performing the 4-bit integer

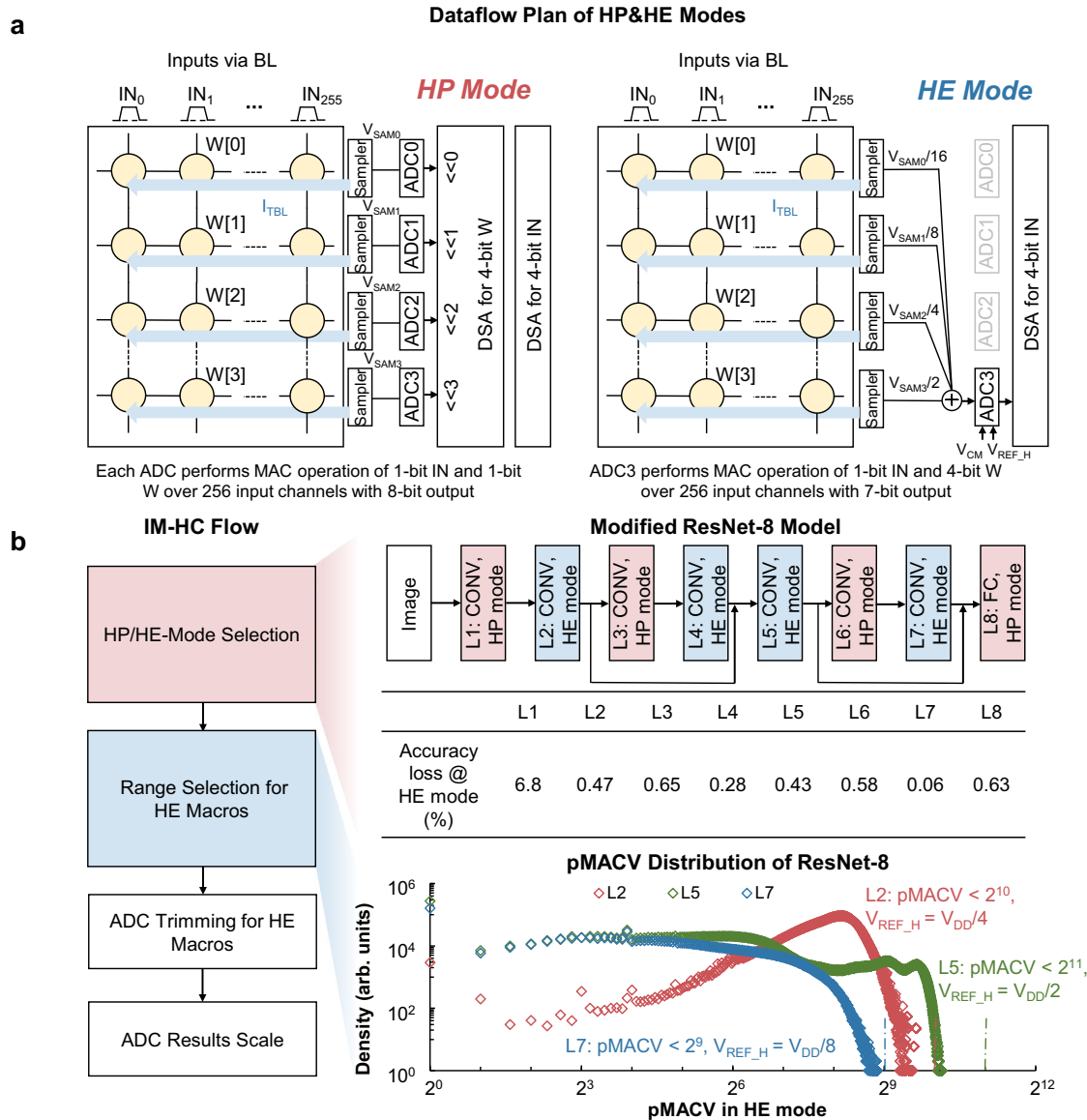


Fig. 4 | Illustration of the proposed IM-HC scheme. a Dataflow of the NVT mCIM macro in the HP and the HE modes for INT4 MAC operation over 256 accumulation channels. The multi-bit weights are processed by the ISA in the HP mode and by the WAC in the HE mode. **b** Processing flow for the IM-HC scheme, including the mode selection, the range selection, the ADC trimming and the ADC results scaling step. The flow is illustrated using a ResNet-8 model trained on the CIFAR-10 dataset

as an example. The mode selection step is performed by analyzing the influences of different quantizing methods on the model accuracy layer-wisely. The range selection step is carried out by predicting the pMACV distribution in HE mode across different neural network layers using the training dataset. In HE mode, the pMACV corresponds to the MAC results of 1-bit input and 4-bit weight, with an accumulation length of 256.

(INT4) MAC operation are shown in Fig. 4a. In both HP and HE modes, the INT4 input (IN) data are bit-wise sent to the BL of the cell array through 4 computing cycles, the INT4 weight (W) data are mapped into four different rows. In the HP mode, the WAC function is disabled and each BSS-ADC works independently. It uses two levels of digital adder-and-shifters (DSAs) to perform the shift and add operation. The first level of DSA is used to combine the partial sum data output by different rows for INT4 W, while the second one is adopted to combine the partial sum data generated by 4 computing cycles for INT4 IN. In the HE mode, the WAC function of BSS-ADC is enabled, allowing the partial sum signals on different rows to be combined in the analog domain for INT4 W. It only uses one level of DSA to combine the partial sum data for INT4 IN.

The operational flow of the IM-HC scheme is shown in Fig. 4b using a modified ResNet-8 model trained on the CIFAR-10 dataset⁵⁴ as an example. First of all, mode selection is carried out to identify

different types of workloads. Here, we evaluate the inference accuracy of the model by changing the quantization strategy from layer-1 (L1) to layer-8 (L8) layer-wisely according to the dataflow in the HP and HE mode. The overall accuracy is more subjective to the quantization precision of the L1, L3, L6, and L8 compared to the other layers. Consequently, the macros for L1, L3, L6, and L8 processing are configured in the HP mode, whereas those for L2, L4, L5, and L7 processing are set to the HE mode. Secondly, in order to suppress the accuracy loss, we analyze the typical partial MAC value (pMACV) distribution of the HE-processed workloads. For instance, the pMACV distribution range of L5 is two times of that of L2 and hence needs a wider quantization range to improve the inference accuracy. Thirdly, the quantization ranges of the ADCs in the HE macros are trimmed by modulating their $V_{REF,H}$ and V_{CM} according to the pMACV distribution of the processing layers. Lastly, depending upon the quantization range, the ADC outputs are re-scaled by digital shifters to generate the

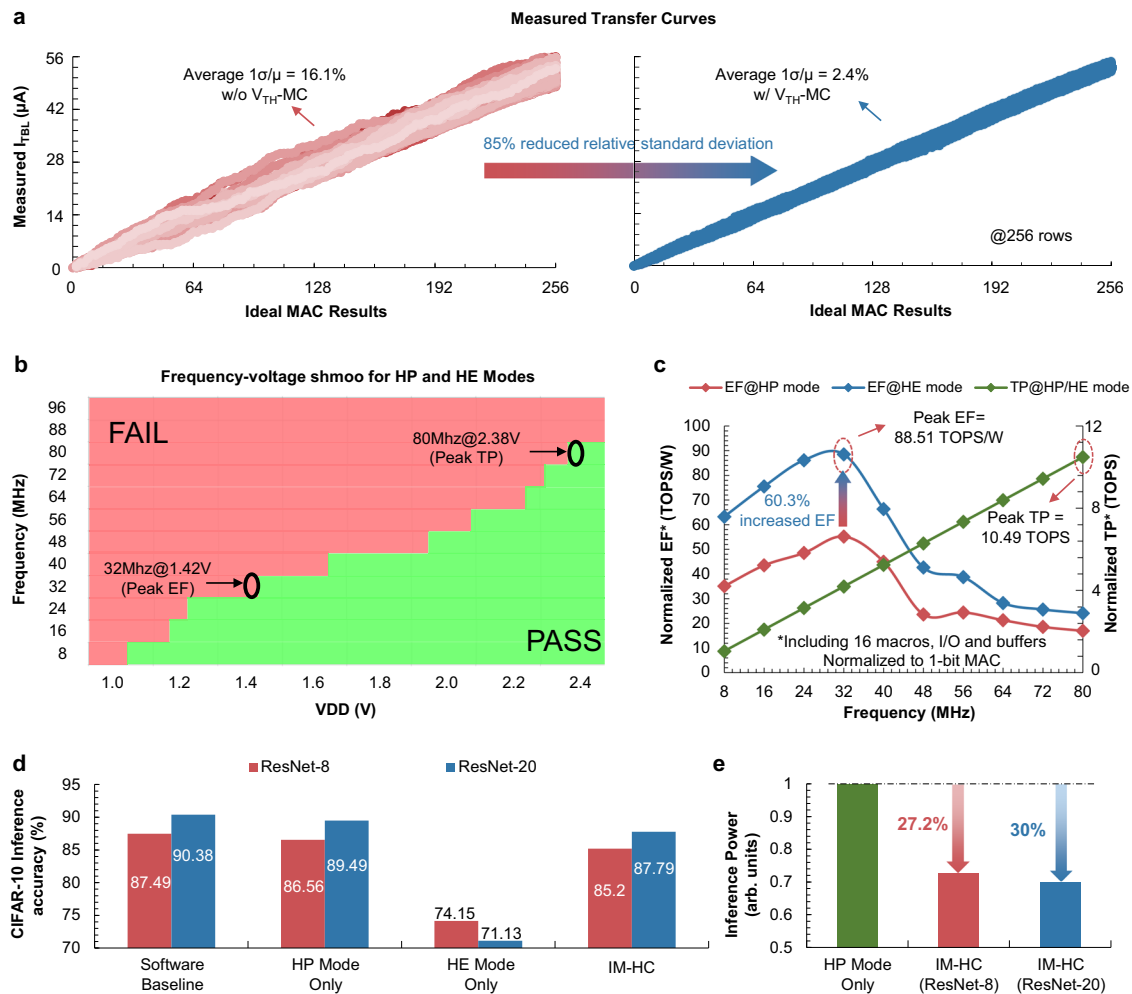


Fig. 5 | Performance of the fabricated NVT mCIM engine. **a** Measured statistics transfer curves over 256 rows w/o and w/ the V_{TH-MC} scheme. **b** Measured shmoo plot. **c** Measured system-level EF and TP at different frequency. **d** and **e** Measured inference accuracy (**d**) and power consumption (**e**) for processing the ResNet-8 and the ResNet-20 models (both trained on the CIFAR-10 dataset) with different inter-

macro controlling schemes, including the HP mode only, the HE mode only and the IM-HC scheme. Using the IM-HC scheme instead of the HP-mode only scheme leads to a marginal decrease in accuracy (1.36% and 1.7% for ResNet-8 and ResNet-20, respectively) but considerably reduces power consumption by 27.2% and 30%, respectively.

corresponding pMACVs. More details about the ADC range selection and trimming can be found in Supplementary Fig. 8. A detailed explanation of the pMACV distribution evaluation is presented in Supplementary Note 2. The pMACV distributions across multiple layers in ResNet-20, along with the effects of different quantization strategies on model inference accuracy, are presented in Supplementary Fig. 9. The illustration of the mode selection step for the IM-HC scheme can be found in Supplementary Note 3.

Performance of the fabricated test-chip

The test-chip is demonstrated on silicon using a 180 nm logic process with BEOL-integrated memristor cells (see Methods for further details). The statistic transfer curves of the test-chip with and without V_{TH-MC} scheme are measured over 256 rows by sweeping the input codes, as shown in Fig. 5a. Highly linear analog current summation can be achieved in both cases. Additionally, the V_{TH-MC} scheme can achieve a relative standard deviation of 2.4% on average, which is 85% reduced compared to that of the case without using the scheme. The shmoo plot (Fig. 5b) shows that the chip can work with a frequency range between 8 to 80 MHz by changing its VDD. The computing engine can deliver a peak normalized TP of 10.49 TOPS at 80 MHz. Fig. 5c further shows the measured normalized EF and TP of the whole chip as a function of frequency (Methods). It achieves an optimal

normalized system-level EF of 88.51 TOPS/W in the HE mode, which is 60.3% improved compared to that of the HP mode. It is worth mentioning that, for a fair comparison of the efficiency between the HP and HE modes, the EF and TP values reported here are normalized to 1-bit precision operations. Additionally, a detailed power breakdown of the test-chip at peak EF can be found in Supplementary Fig. 10. We have developed a neural network inference system based on the test-chip (see Supplementary Fig. 11 for further details). The inference accuracy and power of the test-chip to process the ResNet-8 and the ResNet-20⁵⁵ models for classifying the CIFAR-10 dataset are experimentally measured. The inference testing results are shown in Fig. 5d and Fig. 5e. Using the IM-HC scheme instead of the HP mode only scheme leads to a marginal decrease in accuracy (1.36 and 1.7% for ResNet-8 and ResNet-20, respectively) but considerably reduces power consumption by 27.2 and 30%, respectively. Also, we speculate that more power can be saved as the neural network model scales up because the proportion of the HE-processed workloads tends to continually increase. More details about neural network processing based on the proposed chip are illustrated in the Methods section and the Supplementary Fig. 12. The details of model-to-hardware weight mapping and computing dataflow are shown in Supplementary Fig. 13. Simply put, the IM-HC scheme can effectively reduce the power consumption of the highly parallel CIM systems based on multiple macros with a negligible

accuracy loss. The benchmark of this work can be found in the Supplementary Table 3. It experimentally demonstrated highly linear NVT analog MAC operation over a large number of accumulation channels ($N_{IN} \geq 256$) with small variations ($1\sigma/\mu = 2.4\%$). Besides, this work also demonstrates a higher level of computing parallelism compared to the earlier works.

Discussion

We have reported a highly energy-efficient and parallel 1Mb 16-macro NVT 2T1R mCIM engine enabled by cross-layer co-design. At the array-level, the combination of the 2T1R cell and the V_{TH-MC} scheme can lead to NVT memristive array with small cell current, large signal ratio, and suppressed variations. At the macro-level, the BSS-ADC with a PDQ clamper can effectively reduce the energy- and area-costs for analog processing and readout with reduced overall readout latency. At the system-level, the IM-HC scheme is able to reduce the power consumption with a negligible accuracy loss in multi-macro mCIM systems by configuring the macros in different modes. The highly parallel and linear analog MAC operations in the NVT region with a small relative deviation of 2.4% are statistically characterized. The whole chip delivers a peak normalized TP up to 10.49 TOPS and achieves an optimal energy efficiency from 55.21 to 88.51 TOPS/W. This work demonstrates a path to overcome the long-standing variation challenge faced by mCIM and NVT computing paradigms, which opens the design space to explore for next-generation edge AI hardware.

Methods

Test-chip fabrication

In this work, the memristive chip has a 1 polysilicon-layer and 6 metal-layers (1P6M) structure. It is fabricated by a standard 180 nm logic process in a commercial foundry and an in-house developed memristor technology in a pilot process line. The layers up to and including the fifth via-layer (V5) are fabricated by the foundry process. The layers above V5 are fabricated by the pilot line, including the memristor cells, the sixth metal layer (M6), and the passivation layer (PA). The memristor cell is integrated between the bottom via connected to the fifth metal-layer (M5) and the top via connected to the sixth metal-layer (M6). Each memristor cell consists of a titanium nitride (TiN) bottom-electrode layer, a transition-metal-oxide-based resistive switching layer, and a TiN top-electrode layer.

Current-voltage characterization of the memristive cells

The test-chip is packaged using chips-on-board technology and mounted on a customized 256-pin universal printed circuit board (PCB) with a socket (Yamaichi NP89-44111). A Keysight N6705B DC power analyzer was used to provide the supply voltages (VDD) for the customized PCB. The working modes and the memory address can be selected by connecting the corresponding pins either to the ground or to VDD. For I-V measurement, the test-chip is set to a test-mode so that the selected memristive cells in the cell array can be directly accessed. Then, to precisely measure the sub-threshold characteristics, the I-V curves of the memristor cells inside the cell array of the test chip are measured by connecting the corresponding pins to the supply/measurement units of a standard semiconductor analyzer (Keysight B1500A).

Chip measurements with the automated test equipment

To probe the chip functions statistically, we build a customized load board of the test-chip for an ATE equipment (YTEC S100). The checkerboard memory testing, transfer curves and shmoo testing in this work are done by the ATE. The shmoo plot is carried out by measuring the functional CLK frequency with varying VDD. The resolution of VDD and the CLK frequency were set at 0.02 V and 8 MHz, respectively.

Demonstration for deep neural network inference

We develop an inference system based on the fabricated test-chip to demonstrate deep neural network inference, as shown in the Supplementary Fig. 11. The system consists of a personal computer (PC) as the host, a Xilinx ZCU106 Field Programmable Gate Array (FPGA) main board for the local control, and an expansion board to accommodate the test-chip. In the host PC, a user interface with different operation modules has been developed using LabVIEW. These modules can support the test-chip for different tasks, such as weight mapping, serial peripheral interface configuration and parallel MAC operations. The main board communicates with the PC through a PCIe module. It allows for the exchange of control signals, inputs and output data between the host PC and the test-chip. The expansion board includes the power modules and the DACs to generate a proper power supply and biasing for the test-chip.

Implementations of ResNet models

For the CIFAR-10 image classification tasks, two ResNet models are employed on chip for demonstration, including the standard ResNet-20 model and a modified ResNet-8 model. The CIFAR-10 dataset consists of 50,000 training images and 10,000 testing images, each belonging to one of ten object classes. The modified ResNet-8 includes seven convolutional layers and one fully connected layer, with batch normalizations and ReLU activations between the layers. The ResNet-20 includes 19 convolutional layers and one fully connected layer, also with batch normalizations and ReLU activations between the layers. These models are trained using the PyTorch framework. The inputs and weights of all convolutional and fully connected layers are quantized to a 4-bit fixed-point format. We merge the batch normalization parameters into convolutional weights and biases after training. All the parameters of the ResNet-8 are mapped on a single NVT mCIM engine, including those of the convolutional layers and the fully connected layers, while all the parameters of the ResNet-20 are mapped on two NVT mCIM engines. Other operations, including shift-and-add, accumulation, dummy output subtraction, shortcut, activation, and pooling are implemented on a Xilinx ZCU106 FPGA integrated on the same board as the NVT mCIM engine. These operations contribute only a small fraction of the total computation, and integrating their implementation in digital CMOS would incur negligible overhead. The choice of FPGA implementation provides flexibility during testing and development.

To implement the weights of a four-dimensional convolutional layer with dimension height (H), width (W_d), input channel (C_{IN}) and output channel (C_{OUT}) on two-dimensional memristive arrays, the first three dimensions are flattened into a one-dimensional vector and the bias term (B) of each output channel is appended to each vector. Under the IM-HC scheme, the parameters of a convolutional layer are converted into a conductance matrix of size $(HW_d C_{IN} + B, C_{OUT})$. Several model-to-hardware mapping schemes are used to improve TP and facilitate the pipelined processing of multiple inference tasks as shown in the Supplementary Fig. 13, including (1) duplicating weights among different macros for the layers (L1-L2 in ResNet-8, L1-L7 in ResNet-20) having small amounts of parameters (N_{PA}) but large input feature maps to improve the input parallelism, (2) splitting weights between different macros for the layers (all the layers in ResNet-8 and ResNet-20) having C_{OUT} larger than the output channels of a macro (N_{OUT}) to improve the output parallelism, (3) dividing weights into different pipelined stages for the layers (L5-L7 in ResNet-8, L14-L19 in ResNet-20) having C_{OUT} more than $N_M \times N_{OUT} / N_P$, where N_M is the number of the macros in the system and N_P is the number of pipeline stages.

Data availability

All data that support the findings of this study are included in the main text and supplementary information. Source data are provided with this paper.

Code availability

All codes used in simulations supporting this article are included in the supplementary information.

References

- Subramanian, I. & Vwani, R. AI computing reaches for the edge. *Science* **382**, 263–264 (2023).
- Tomas, S. AI chips that flip. *Nat. Electron.* **6**, 178 (2023).
- Skillman, A. & Edsö, T. A technical overview of cortex-M55 and Ethos-U55: arm's most capable processors for endpoint AI. In: *Proc. IEEE Hot Chips 32 Symposium (HCS) 1-20* IEEE, (2020).
- Yakubo, Y. et al. Crystalline oxide semiconductor-based 3D bank memory system for endpoint artificial intelligence with multiple neural networks facilitating context switching and power gating. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers 12-14* (IEEE, 2023).
- Wang, Z. et al. Resistive switching materials for information processing. *Nat. Rev. Mater.* **5**, 173–195 (2020).
- Xiao, T. P. et al. Analog architectures for neural network acceleration based on non-volatile memory. *Appl. Phys. Rev.* **7**, 031301 (2020).
- Lanza, M. et al. Memristive technologies for data storage, computation, encryption, and radio-frequency communication. *Science* **376**, eabj9979 (2022).
- Sun, Z. et al. A full spectrum of computing-in-memory technologies. *Nat. Electron.* **6**, 823–835 (2023).
- Mochida, R. et al. A 4M synapses integrated analog ReRAM based 66.5 TOPS/W neural-network processor with cell current controlled writing and flexible network architecture. In: *Proc. Symposium on VLSI Technology, Digest of Technical Papers 175–176* (IEEE, 2018).
- Wang, Z. et al. An all-weights-on-chip DNN accelerator in 22nm ULL featuring 24×1 Mb eRRAM. In: *Proc. IEEE Symposium on VLSI Circuits 1-2* (IEEE, 2020).
- Prabhu, K. et al. CHIMERA: A 0.92-TOPS, 2.2-TOPS/W edge AI accelerator with 2-MByte on-chip foundry resistive RAM for efficient training and inference. *IEEE J. Solid State Circuits* **57**, 1013–1026 (2022).
- Chang, M. et al. A 40nm 60.64TOPS/W ECC-capable compute-in-memory/digital 2.25MB/768KB RRAM/DRAM system with embedded cortex M3 microprocessor for edge recommendation systems. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers 1–3* (IEEE, 2022).
- Huang, W.-H. et al. A Nonvolatile AI-Edge Processor with 4MB SLC-MLC Hybrid-Mode ReRAM Compute-in-Memory Macro and 51.4-251TOPS/W. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers 15-17* (IEEE, 2023).
- Shafiee, A. et al. ISAAC: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars. In: *Proc. ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA) 14–26* (IEEE, 2016).
- Yao, P. et al. Fully hardware-implemented memristor convolutional neural network. *Nature* **577**, 641–646 (2020).
- Wan, W. et al. A 74 TMACS/W CMOS-RRAM neurosynaptic core with dynamically reconfigurable dataflow and in-situ transposable weights for probabilistic graphical models. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers 498–500* (IEEE, 2020).
- Zhang, W. et al. Edge learning using a fully integrated neuro-inspired memristor chip. *Science* **381**, 1205–1211 (2023).
- Wen, T. et al. Fusion of memristor and digital compute-in-memory processing for energy-efficient edge computing. *Science* **384**, 325–332 (2024).
- Cai, F. et al. A fully integrated reprogrammable memristor-CMOS system for efficient multiply-accumulate operations. *Nat. Electron.* **2**, 290–299 (2019).
- Chen, W. H. et al. CMOS-integrated memristive non-volatile computing-in-memory for AI edge processors. *Nat. Electron.* **2**, 420–428 (2019).
- Xue, C. X. et al. A CMOS-integrated compute-in-memory macro based on resistive random-access memory for AI edge devices. *Nat. Electron.* **4**, 81–90 (2021).
- Hung, J. M. et al. A four-megabit compute-in-memory macro with eight-bit precision based on CMOS and resistive random-access memory for AI edge devices. *Nat. Electron.* **4**, 921–930 (2021).
- Wan, W. et al. A compute-in-memory chip based on resistive random-access memory. *Nature* **608**, 504–512 (2022).
- Huo, Q. et al. A computing-in-memory macro based on three-dimensional resistive random-access memory. *Nat. Electron.* **5**, 469–477 (2022).
- Wang, C. et al. Parallel in-memory wireless computing. *Nat. Electron.* **6**, 381–389 (2023).
- Zhao, H. et al. Energy-efficient high-fidelity image reconstruction with memristor arrays for medical diagnosis. *Nat. Commun.* **14**, 2276 (2023).
- Yoon, J.-H. et al. A 40-nm, 64-Kb, 56.67 TOPS/W voltage-sensing computing-in-memory/digital RRAM macro supporting iterative write with verification and online read-disturb detection. *IEEE J. Solid-State Circuits* **57**, 68–79 (2022).
- Yan, B. et al. RRAM-based spiking nonvolatile computing-in-memory processing engine with precision-configurable in situ nonlinear activation. In: *Proc. Symposium on VLSI Technology, Digest of Technical Papers T86–T87* (IEEE, 2019).
- Xue, C. X. et al. A 1Mb multibit ReRAM computing-in-memory macro with 14.6ns parallel MAC computing time for CNN based AI edge processors. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers 388–390* (IEEE, 2019).
- Liu, Q. et al. A fully integrated analog ReRAM based 78.4TOPS/W compute-in-memory chip with fully parallel MAC computing. In: *Proc. International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers 500–502* (IEEE, 2020).
- Xue, C.-X. 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: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers 244–245* (IEEE, 2020).
- Xue, C.-X. et al. 16.1 A 22nm 4Mb 8b-precision ReRAM computing-in-memory macro with 11.91 to 195.7TOPS/W for tiny AI edge devices. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers 245–247* (IEEE, 2021).
- Hung, J.-M. et al. An 8-Mb DC-current-free binary-to-8b precision ReRAM nonvolatile computing-in-memory macro using time-space-readout with 1286.4-21.6TOPS/W for edge-ai devices. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers 1–3* (IEEE, 2022).
- Correll, J. M. et al. An 8-bit 20.7 TOPS/W multi-level cell ReRAM-based compute engine. In: *Proc. IEEE Symposium on VLSI Technology and Circuits (VLSI Technology and Circuits) 264-265* (IEEE, 2022).
- Chang, M. et al. A 73.53TOPS/W 14.74TOPS heterogeneous RRAM in-memory and SRAM near-memory SoC for hybrid frame and event-based target tracking. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers 426-428* (IEEE, 2023).
- Ye, W. et al. A 28-nm RRAM computing-in-memory macro using weighted hybrid 2T1R cell array and reference subtracting sense amplifier for AI edge inference. *IEEE J. Solid-State Circuits* **58**, 2839–2850 (2023).
- Yin, S., Sun, X., Yu, S. & Seo, J.-S. High-throughput in-memory computing for binary deep neural networks with monolithically integrated RRAM and 90-nm CMOS. *IEEE Trans. Electron Devices* **67**, 4185–4192 (2020).

38. Tsai, H. et al. Inference of long-short term memory networks at software-equivalent accuracy using 2.5m analog phase change memory devices. In: *Proc. Symposium on VLSI Technology*. T82–T83 (IEEE, 2019).
39. Joshi, V. et al. Accurate deep neural network inference using computational phase-change memory. *Nat. Commun.* **11**, 2473 (2020).
40. Khaddam-Aljameh, R. et al. HERMES core-A 14nm CMOS and PCM-based in-memory compute core using an array of 300ps/LSB linearized CCO-based ADCs and local digital processing. In: *Proc. Symposium on VLSI Circuits, Digest of Technical Papers* 1-2 (IEEE, 2021).
41. Le Gallo, M. et al. A 64-core mixed-signal in-memory compute chip based on phase-change memory for deep neural network inference. *Nat. Electron.* **6**, 680–693 (2023).
42. Ishii, M. et al. On-chip trainable 1.4M 6T2R PCM synaptic array with 1.6K stochastic LIF neurons for spiking RBM. In: *Proc. International Electron Devices Meeting (IEDM), Technical Digest* 14.2.1–14.2.4 (IEEE, 2019).
43. Narayanan, P. et al. Fully on-chip MAC at 14 nm enabled by accurate row-wise programming of PCM-based weights and parallel vector-transport in duration-format. *IEEE Trans. Elect. Dev.* **68**, 6629–6636 (2021).
44. Jung, S. et al. A crossbar array of magnetoresistive memory devices for in-memory computing. *Nature* **601**, 211–216 (2022).
45. Chiu, Y. C. et al. A CMOS-integrated spintronic compute-in-memory macro for secure AI edge devices. *Nat. Electron.* **6**, 534–543 (2023).
46. Soliman, T. et al. First demonstration of in-memory computing crossbar using multi-level cell FeFET. *Nat. Commun.* **14**, 6348 (2023).
47. Murmann, B. Introduction to ADCs/DACs: metrics, topologies, trade space, and applications. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Short Course Presentation* (IEEE, 2022).
48. Dong, Q. et al. 15.3 A 351TOPS/W and 372.4GOPS compute-in-memory SRAM macro in 7nm FinFET CMOS for machine-learning applications. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers* 242–244 (IEEE, 2020).
49. Wu, P.-C. et al. A 22nm 832Kb hybrid-domain floating-point SRAM In-memory-compute macro with 16.2–70.2TFLOPS/W for high-accuracy AI-Edge Devices. In: *Proc. IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers* 126–128 (IEEE, 2023).
50. Kaul, H. et al. Near-threshold voltage (NTV) design — opportunities and challenges. In: *Proc. Design Automation Conference (DAC)* 1149–1154 (IEEE, 2012).
51. Pinckney, N., Blaauw, D. & Sylvester, D. Low-power near-threshold design: techniques to improve energy efficiency. *IEEE Solid-State Circuits Mag.* **7**, 49–57 (2015).
52. Cerqueira, J. P. et al. Catena: a near-threshold, sub-0.4-mW, 16-core programmable spatial array accelerator for the ultralow-power mobile and embedded internet of things. *IEEE J. Solid-State Circuits* **55**, 2270–2284 (2020).
53. Dou, C. et al. Nonvolatile circuits-devices interaction for memory, logic and artificial intelligence. In: *Proc. Symposium on VLSI Technology* 171–172 (IEEE, 2018).
54. Krizhevsky, A. Learning multiple layers of features from tiny images. <https://www.cs.toronto.edu/~kriz/cifar.html> Learning Multiple (2009).
55. He, K. et al. Deep residual learning for image recognition. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 770–778 (IEEE, 2016).
56. Cai, F. et al. Power-efficient combinatorial optimization using intrinsic noise in memristor Hopfield neural networks. *Nat. Electron.* **3**, 409–418 (2020).
57. Nandakumar, S. R. et al. Mixed-precision deep learning based on computational memory. *Front. Neurosci.* **14**, 406 (2020).
58. Huang, S. et al. Mixed precision quantization for ReRAM-based DNN inference accelerators. In: *Proc. 26th Asia and South Pacific Design Automation Conference* 372–377 (IEEE, 2021).
59. Rao, M. et al. Thousands of conductance levels in memristors integrated on CMOS. *Nature* **615**, 823–829 (2023).
60. Khaddam-Aljameh, R. et al. An SRAM-based multibit in-memory matrix-vector multiplier with a precision that scales linearly in area, time, and power. *IEEE Trans. Very Large Scale Integr. Syst.* **29**, 372–385 (2021).

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant No. 92364202, 62488101, U2441247, and 62425407.

Author contributions

C.D. and L.W. conceived the initial idea and made the research plan. L.W., W.L., Z.Z., and C.D. designed the test-chip and carried out the measurements. L.W., W.L., Z.Z., J.A., Y.W., Z.L., H.G., H.H., J.L., C.D., and M.S. analyzed and interpreted the data. L.W., W.L., Z.Z., X.C., L.L., C.D., and M.S. wrote and revised the manuscript. C.D., Q.L., and M.L. supervised the project. All authors have discussed the results and commented on the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41467-025-61025-4>.

Correspondence and requests for materials should be addressed to Chunmeng Dou.

Peer review information *Nature Communications* thanks Giacomo Pedretti and the other anonymous reviewer(s) for their contribution to the peer review of this work. A peer review file is available.

Reprints and permissions information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025