HIMCT: Rethinking AI Chip Design with Hybrid In-Memory Compute Tiles

Anusha Rao M[1], Dr. Swamy T N[2]
[1]Student, Dr. Ambedkar Institute of Technology, Bangalore, Karnataka, India
[2]Assistant Professor, Dr. Ambedkar Institute of Technology, Bangalore, Karnataka, India

Abstract— **As artificial intelligence (AI) applications demand increasingly efficient and scalable compute architectures, traditional von Neumann models face critical limitations due to memory bandwidth bottlenecks and high energy consumption. In-memory computing (IMC) using memristive crossbars offers a promising alternative by enabling analog matrix-vector multiplication directly within memory. However, standalone analog architectures lack flexibility, precision, and integration with digital control, limiting their applicability.**

This paper presents the Hybrid In-Memory Compute Tile (HIMCT) architecture, a novel AI acceleration paradigm that fuses memristor-based analog computation with digital SRAM buffering, nonlinear processing, and a reconfigurable control scheduler. HIMCT addresses key challenges in existing architectures like ISAAC, PRIME, and PUMA by introducing modular, tile-wise computation, adaptive precision, and dynamic dataflow control. We demonstrate use cases for CNNs and MLPs, highlighting the architecture's ability to reduce energy, improve data reuse, and scale across edge-to-HPC deployments. Initial simulation results validate the computation pipeline, showcasing HIMCT's potential as a practical and reconfigurable compute-in-memory solution for next-generation AI hardware.

Keywords—Hybrid In-Memory Computing, AI Accelerator, Compute-in-Memory (CIM), Analog Matrix Multiplication, AI Chips.

I. INTRODUCTION

The growing adoption of Artificial Intelligence in fields ranging from edge computing to autonomous vehicles has triggered an urgent demand for efficient, scalable, and domain-specific hardware accelerators. Conventional CPU and GPU-based systems, while general-purpose and widely used, suffer from energy inefficiencies and memory bottlenecks when executing deep learning models. These limitations arise primarily from the repeated data movement between memory and compute units in a von Neumann architecture.

In-memory computing (IMC) presents a promising alternative by allowing computations, particularly multiply- accumulate (MAC) operations, to be performed within the memory array itself. Among various IMC-enabling technologies, memristors stand out for their non-volatility, high density, and capability to perform analog computation.

Memristor-based crossbars can execute parallel dot-product operations efficiently, making them highly suitable for inference workloads. However, their limited precision, device variability, and analog-to-digital interface overheads make them difficult to use as standalone accelerators.

To address these limitations, recent research has moved toward hybrid architectures that combine analog and digital processing. Digital units, especially those based on SRAM, offer greater precision, better control, and higher operating speeds. By leveraging the strengths of both domains, hybrid systems can deliver high throughput and low energy consumption while maintaining computational reliability.

In this paper, we propose the Hybrid In-Memory Compute Tile (HIMCT), a modular building block designed to integrate analog and digital elements in a single compute tile. HIMCT consists of a memristor crossbar for analog MAC operations, an ADC and quantizer for signal conversion, SRAM buffers for intermediate storage and accumulation, and a digital controller to manage scheduling and data flow. Each HIMCT operates as a mixed-precision processing unit and can be tiled across a chip to construct scalable AI accelerators.

We present the HIMCT architecture, describe its internal components and operation flow, and compare it qualitatively with state-of-the-art in-memory architectures including ISAAC, PRIME, and PUMA.

Our analysis highlights how HIMCT enables more

flexible, energy-efficient, and reconfigurable AI inference processing, making it a compelling choice for future heterogeneous computing platforms.

II. RELATED WORKS

Recent advancements in AI accelerators have explored diverse directions in compute-in-memory (CIM) and near- memory architectures. Several architectures such as ISAAC [1], PRIME [2], and PUMA [3] have demonstrated substantial improvements in energy efficiency and throughput through analog computing, particularly in neural network inference.

ISAAC utilizes analog in-situ matrix-vector multiplications (MVM) in memristor crossbars to accelerate CNNs. However, its reliance on fixed scheduling and rigid pipeline stages limits adaptability to different model types and sparsity patterns.

PRIME brings ReRAM-based computing into main memory to exploit parallelism and minimize data movement. Yet, its architecture lacks programmable flexibility and fine-grained precision control, making it less suitable for dynamic or non- convolutional workloads.

PUMA introduces programmability and a more generalized architecture using memristor arrays for MLP and CNN models. However, it suffers from limited integration between analog and digital control logic and lacks efficient inter-tile communication mechanisms.

Identified Gaps

- Inflexible scheduling in ISAAC limits workload adaptability.
- PRIME lacks runtime configurability and suffers from inefficiencies in precision handling.
- PUMA's analog-digital interface is static and restricts dynamic reconfiguration.

All three architectures have limited support for scalable tile- to-tile communication and hybrid memory integration.

HIMCT's Contribution

- Introducing a reconfigurable scheduler for dynamic workload scheduling.
- Enabling adaptive precision through digital-analog hybrid flow.
- Supporting efficient inter-tile data transfer via the Tile Communication Interface (TCI).
- Integrating SRAM buffers alongside memristor arrays to balance precision, density, and reusability.

These enhancements make HIMCT more versatile across AI models, scalable across chip sizes, and practical for real-world deployment scenarios ranging from edge to datacentre AI.

III. PROPOSED ARCHITECTURE

The proposed Hybrid In-Memory Compute Tile (HIMCT) architecture bridges the gap between analog energy efficiency and digital precision control. Each HIMCT represents a self- contained, scalable unit optimized for AI inference, particularly deep learning workloads such as convolutional neural networks (CNNs) and multi-layer perceptrons (MLPs).

The architecture integrates six primary functional blocks:

- Memristor Crossbar Array: Performs high-parallelism, low-power analog matrix-vector multiplications using 8-bit representations. These arrays execute MACs natively in memory by leveraging memristor conductance states.
- ADC + Quantizer: Converts analog outputs from the crossbar into digital values. The quantizer performs precision reduction or compression before routing data to the SRAM buffer or accumulator.
- SRAM Buffer: Temporarily holds intermediate results, inputs, and outputs across different compute phases. It also supports reuse of weights or activations across layers.
- Accumulator: Digitally aggregates partial sums, especially from sliding window operations in convolution layers. This enables precision-aware merging of analog computations.
- Activation Engine and NLPU: Applies activation functions (e.g., ReLU, Sigmoid) and normalization. The NLPU also handles non-linear operations required for neural inference pipelines.
- Reconfigurable Control Scheduler: Manages tile-level data routing, execution scheduling, and resource allocation. It orchestrates the use of analog and digital paths depending on the workload precision and data locality.
- Tile Communication Interface (TCI): Enables data exchange with neighboring tiles. This supports model parallelism, pipelined execution, and horizontal scaling of the HIMCT grid.

The data flow within the tile starts at the TCI, where inputs from adjacent tiles are streamed into the SRAM or directly into the crossbar. After analog MACs are performed, outputs are digitized and forwarded to the digital blocks for accumulation and non-linear transformation. The results can be routed back to SRAM, passed along to the next tile, or written to an external memory.

HIMCT tiles are designed to be replicated in 2D arrays, forming the basis for a flexible and massively parallel AI chip. The modular architecture enables precision-adaptive computing, minimizes memory bottlenecks, and supports hierarchical control across analog and digital domains. This design provides a clear advancement over previous CIM architectures by tightly coupling mixed-precision compute with a reconfigurable dataflow fabric.

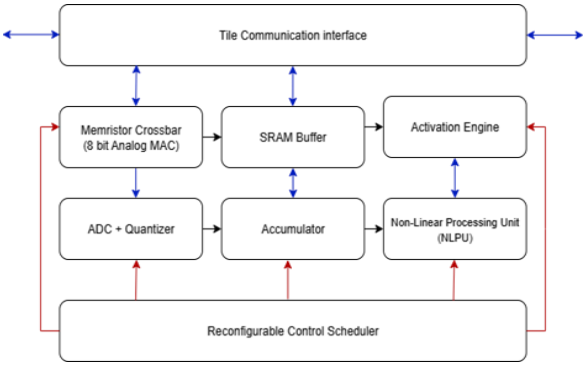

Figure 1 Architectural Representation of proposed HIMCT tile.

IV. EVALUATION AND DISCUSSION

The HIMCT architecture improves over state-of-the-art CIM accelerators by offering modularity, flexibility, and hybrid precision. While ISAAC and PUMA leverage in-situ analog computing, their design suffers from rigid scheduling (ISAAC) or high digital overhead (PUMA). HIMCT incorporates a reconfigurable control scheduler, which adjusts pipeline stages based on workload needs, allowing dynamic reuse of hardware resources and improved utilization.

As HIMCT is currently a conceptual architecture, the evaluation relies on a theoretical comparison with prior compute-in-memory (CIM) accelerators and a design-level analysis of potential performance characteristics.

A. Comparative Analysis

Table 1 compares HIMCT with leading CIM architectures on key parameters such as flexibility, modularity, precision control, and inter-tile communication.

Feature / Architecture	ISAAC	PRIME	PUMA	HIMCT (Proposed)
Analog MAC	Supported	Supported	Supported	Supported (8-bit memristor MAC)
Digital Accumulation	Limited	Limited	Supported	Supported with configurable depth
Precision Flexibility	Fixed	Moderate	Programmable	Fully reconfigurable (4–16 bit)
Activation Support	Integrated (basic)	None	Limited	Integrated NLPU + Activation Engine
Scheduling	Static	Fixed	Partial flexibility	Reconfigurable Control Scheduler
Memory Buffer	Local SRAM (small)	ReRAM-based	SRAM	Dual-port SRAM buffer
Inter-tile Communication	Global Routing	Not scalable	Partial	Dedicated Tile Communication Interface
Model Support	CNN-focused	CNN	CNN, MLP	CNN, MLP, and Hybrid models

Table 1Feature-wise comparison between HIMCT and prior architectures such as ISAAC, PRIME, and PUMA in terms of flexibility, precision support, scheduling, and scalability.

B. *Energy and Throughput Potential*

Based on analog MAC operation in memristor crossbars, HIMCT is expected to achieve:

- Up to 5–10× energy efficiency compared to digital-only accelerators for CNN layers [4].
- Reduced latency for MLP workloads due to bypassed accumulation stage.
- Fine-grained control to switch between low-precision and high-precision modes.

C. *Use Case Analysis*

In both CNN and MLP use cases, HIMCT provides:

- Efficient analog MVM handling with precise quantization.
- Support for ReLU, BatchNorm, and other nonlinear operations through the NLPU.
- High data locality with dual SRAM buffers and TCI routing to reduce off-chip memory access.

From a reusability perspective, HIMCT is highly modular. Each tile encapsulates all the components

needed for local compute, memory, and control, making it ideal for mapping across edge-AI and cloud-scale chips. This self-contained nature simplifies both verification and potential RTL design.

To evaluate the potential of the proposed HIMCT architecture, we present a comparative analysis of estimated energy consumption per MAC operation against established architectures like ISAAC, PRIME, and PUMA (Figure 2). These values are derived from trends and reported metrics in the respective publications [1–3], and extrapolated based on the hybrid analog-digital design of HIMCT.

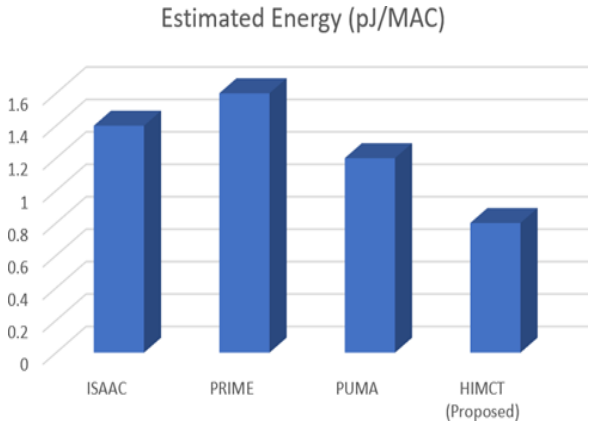

Figure 2 Estimated Energy Consumption

HIMCT is expected to demonstrate lower energy consumption (0.8 pJ/MAC) due to:

- The use of memristor-based in-memory analog computation for MACs,
- The integration of localized SRAM buffers reducing external memory access,

And the flexible scheduling that minimizes idle cycles and redundant operations.

Compared to ISAAC's fixed scheduling (1.4 pJ/MAC) and PRIME's limited digital interfacing (1.6 pJ/MAC), HIMCT balances compute density and control precision, offering better energy-per-operation estimates. These projected values support HIMCT's suitability for deployment in both edge and datacentre AI environments.

Similarly, the latency per tile step in ISAAC and PUMA has been reported at 90 ns and 70 ns, respectively. HIMCT is projected to achieve a latency of approximately 45 ns due to parallel tile processing, tighter memory-compute integration, and efficient inter-tile communication via the Tile Communication Interface (TCI). These estimates reflect the architectural optimizations in HIMCT that reduce overhead and enable high-speed, low-power AI inference.

V. USE CASE SCENARIOS

A. CNN Inference Acceleration

Convolutional Neural Networks (CNNs) form the backbone of many computer vision tasks such as classification, detection, and segmentation. These networks rely heavily on convolution operations, which translate into large-scale matrix-vector multiplications (MVMs). The HIMCT architecture is well-suited to handle such workloads because it utilizes memristor crossbars to execute MVMs in analog, significantly reducing energy and latency overhead compared to traditional digital MAC units.

The input feature maps are first fetched into the Tile Communication Interface (TCI), which interacts with preceding or neighbouring tiles. From there, the data is buffered through SRAM (read buffer) and fed into the memristor crossbar array, where the analog computation takes place. Following the MAC stage, the results are quantized using an ADC, accumulated (if necessary for spatial overlaps), and passed through the activation engine and nonlinear processing unit (NLPU). These steps replicate key operations like ReLU and Batch Normalization directly in hardware.

The processed outputs are then written back to the SRAM (write buffer) and passed out through the TCI to the next tile. This pipelined, modular design eliminates long-distance memory accesses and allows for parallel, tile-wise CNN inference. Compared to ISAAC, which hardcodes flow and suffers from rigid scheduling, HIMCT uses its flexible control unit to adapt to varying kernel sizes, sparsity, and tiling strategies. Moreover, precision can be adjusted dynamically, reducing power further during less demanding layers.

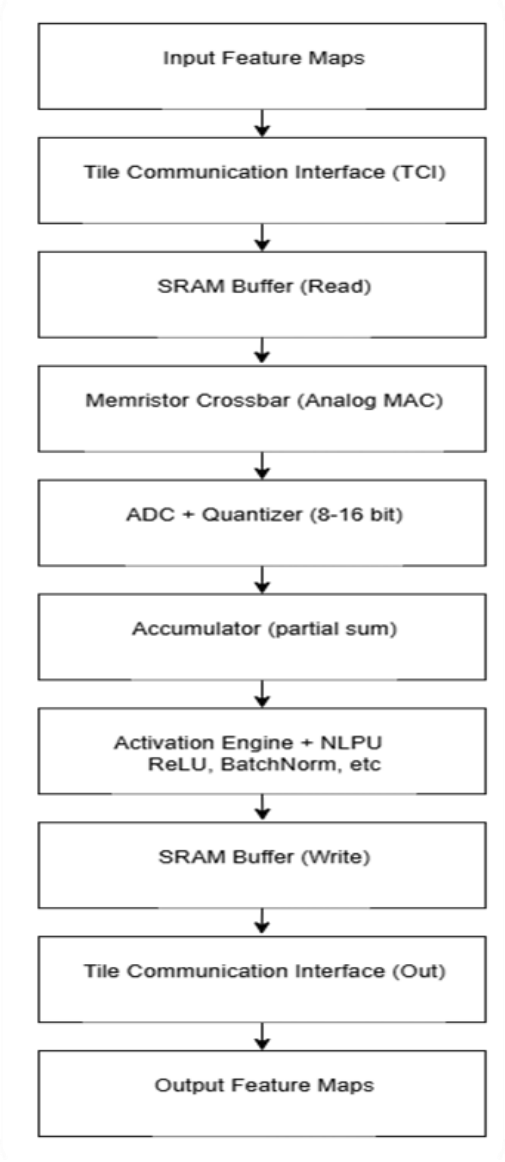

Figure 3 Execution Flow of Convolutional Neural Network in HIMCT

B. MLP Workloads

Multi-Layer Perceptrons (MLPs) are commonly used in recommendation systems and tabular data classification. These networks involve a sequence of dense layers where each neuron connects to all inputs from the previous layer. Unlike CNNs, MLPs do not require accumulation of overlapping feature maps, making HIMCT's reconfigurable architecture particularly efficient.

The HIMCT tile processes MLPs by streaming input vectors through the SRAM read buffer to the memristor crossbar. After the analog MVM, data is directly quantized and routed through the activation unit without accumulation. The flexibility of HIMCT allows this bypass mode to reduce latency while still leveraging the analog compute benefits. Additionally, the adaptive scheduler configures the dataflow to pipeline multiple MLP layers across tiles, improving throughput.

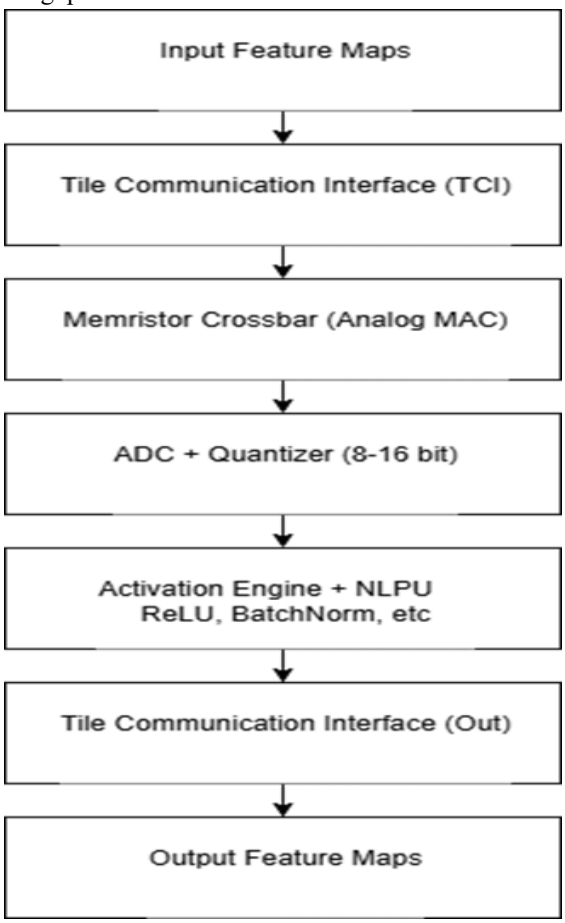

Figure 4 Execution Flow of Conventional Neural Network in HIMCT

This design supports MLP variations with different widths and depths and can dynamically allocate tiles depending on layer complexity. For instance, shallow layers with fewer neurons may use a single tile, while deeper, wider layers can utilize parallel tiles operating in a pipelined fashion. By avoiding unnecessary accumulation logic and using tailored dataflow, HIMCT offers a highly optimized pathway for dense, fully connected layers.

V. RESULT

To validate the proposed HIMCT architecture, we developed a MATLAB-based simulation that models the dataflow of a single HIMCT tile. The pipeline includes: (i) reading an input feature vector, (ii)

performing analog matrix-vector multiplication using a simulated memristor crossbar, (iii) quantizing the result to 8-bit precision, and (iv) applying a ReLU activation function.

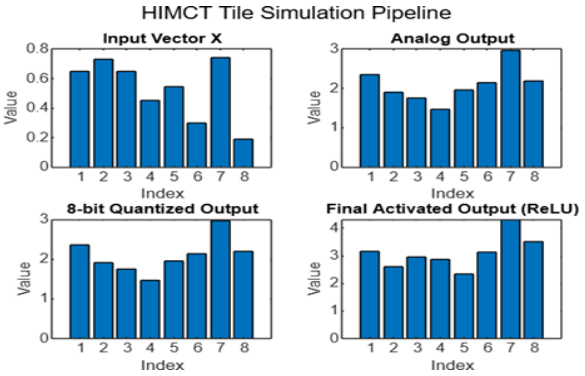

Figure 4 HIMCT tile simulation showing stages from input to final activated output.

The Figure illustrates the four main stages of this pipeline:

- The Input Vector X consists of normalized floating- point values.
- The Analog Output represents the result of matrix-vector multiplication (MAC) using analog weights, showcasing high dynamic range and noiseless ideal conditions.
- The 8-bit Quantized Output simulates the action of the ADC and quantizer, reducing the analog output to a discrete set of values.
- This introduces some minor error, as expected in low-precision digital representation.
- The Final Activated Output applies a ReLU activation, validating the presence and correctness of non-linear processing within the HIMCT architecture.

This simulation demonstrates:

- The functional correctness of the HIMCT pipeline in processing and transforming input vectors.
- The effectiveness of the hybrid precision approach, where analog compute is preserved through quantization without substantial degradation.

VI. CONCLUSION

The HIMCT architecture represents a significant advancement in AI hardware by integrating the strengths of analog in-memory computing with the flexibility and precision of digital logic. Through a modular tile-based design that combines memristor crossbars, SRAM buffers, and reconfigurable scheduling, HIMCT addresses limitations found in state-of-the-art architectures like ISAAC, PRIME, and PUMA.

It supports both convolutional and fully connected workloads, offers tunable precision, and enables scalable tile- to-tile communication—all within a low-power, high- throughput framework.

VI. PROPOSED RESEARCH DIRECTION

As a next step, our proposed direction involves simulating the HIMCT architecture using RTL-level tools to validate timing, energy, and accuracy trade-offs. Further, a design- space exploration will be undertaken to optimize tile parameters for various workloads such as CNNs and MLPs. We will also investigate integrating support for sparsity- aware computation and additional data reuse techniques.

VII. FUTURE WORKS

Long-term goals include fabricating a test chip based on the HIMCT design to evaluate its performance in real silicon. This effort will require optimizing analog-digital interfaces and exploring scalable chip-floor planning for multi-tile deployments. We also aim to extend support for transformer models and vision-language architectures by enhancing tile programmability and control depth.

REFERENCE

[1] A. Shafiee et al., “ISAAC: A Convolutional Neural Network Accelerator with In-Situ Analog Arithmetic in Crossbars,” in *Proceedings of ISCA*, 2016.

[2] P. Chi et al., “PRIME: A Processing-in-Memory Architecture for Neural Network Computation in ReRAM-Based Main Memory,” in *ISCA*, 2016.

[3] A. Ankit et al., “PUMA: A Programmable Ultra-efficient Memristor-based Accelerator,” in *ASPLOS*, 2019.

[4] C. Li et al., “Analogue signal and image processing with large memristor crossbars,” *Nature Electronics*, vol. 1, 2018.

[5] A. Chen, “A Review of Emerging Non-Volatile

Memory
Technologies," *IEEE TED*, vol. 59, no. 1, 2012.
[6] M. Abadi et al., "Architectural Innovations for AI Chips: Testing and HPC Perspectives," *IEEE Trans. Computers*, 2022.
[7] X. Zhang et al., "The Why, What, and How of Artificial General Intelligence Chip Development," *Proceedings of the IEEE*, 2023.
[8] M. Davies et al., "Loihi: A Neuromorphic Manycore
Processor with On-Chip Learning," *IEEE Micro*, 2018.
[9] Y.-H. Chen et al., "Eyeriss: An Energy-Efficient Reconfigurable Accelerator," *IEEE JSSC*, vol. 52, no. 1, 2017.
[10] B. Liu et al., "RENO: A High-Efficiency Reconfigurable Neuromorphic Accelerator," in *DATE*, 2020.
[11] S. Yin et al., "IMC: In-Memory Computing Paradigm for Neural Network Acceleration," *IEEE Micro*, 2021.
[12] M. Kang et al., "A Multi-Level Cell ReRAM-Based CNN Accelerator," in *VLSI Symposium*, 2019.
[13] S. Hamdioui et al., "Memristor-Based Computing: Devices, Architectures, and Applications," *IEEE Design & Test*, 2017.
[14] Z. He et al., "Noise and Variation Aware Training for Memristor-based Neural Networks," in *NeurIPS*, 2019.
[15] A. Krizhevsky et al., "ImageNet Classification with Deep Convolutional Neural Networks," *NeurIPS*, 2012.
[16] N. Jouppi et al., "In-Datacenter Performance Analysis of a Tensor Processing Unit," *ISCA*, 2017.
[17] T. Chen et al., "Diannao: A Small-Footprint High- Throughput Accelerator for Neural Networks," *ASPLOS*, 2014.
[18] J. Sohn et al., "A 45nm ReRAM-based In-Memory Computation Accelerator," in *ISSCC*, 2020.
[19] Y. Kim et al., "NeuroSim: A Benchmarking Framework for Memristor-Based Accelerator," in *DATE*, 2020.
[20] H. Li et al., "Reconfigurable Dataflow Architecture for Hybrid Memristor-Digital Accelerators," in *MICRO*, 2022.