A Fast and Energy-Efficient Latch-Based Memristive Analog Content-Addressable Memory Architecture

Paul-Philipp Manea[1,2,3], Jim Ignowski[3], John Paul Strachan[1,2], Luca Buonanno[3]

[1] PGI-14, Forschungszentrum Jülich, Jülich, Germany
[2] RWTH Aachen University, Aachen, Germany
[3] Hewlett Packard Enterprise, Fort Collins, CO, USA

Memristive analog content-addressable memories (aCAMs) are a promising primitive for energy-efficient artificial intelligence hardware because they enable direct analog inequality evaluation inside memory, extending compute-in-memory beyond conventional vector-matrix multiplication [1, 2]. This makes them attractive for applications such as decision-tree and random-forest inference, similarity search, and other rule-based or associative workloads where data movement and repeated comparisons dominate the cost [3, 4].

The main limitation of most memristive aCAM designs is that they are based on static voltage-division during search, with the 6T2M cell being the most widely used representative [1]. Unlike latch-based evaluation, this approach relies on analog divider nodes to generate the comparison result and therefore suffers from static search current, limited output gain, and cumulative match-line crosstalk, which reduce sensing margin and constrain analog precision and scalability in large arrays. These effects become increasingly problematic as more cells are connected to the same match line, causing conventional architectures to degrade for higher-dimensional inference tasks.

In this work, we introduce a Strong Arm Latched Memristor (SALM) aCAM cell that replaces static voltage division with a dynamic current-race comparison. The latch provides strong regenerative gain, intrinsically stores the comparison result, and suppresses static search power. At the circuit level, SALM reduces read energy by 33 % at identical latency compared with 6T2M, while also providing significantly improved match-mismatch separation and robustness against match-line crosstalk.

Beyond the cell itself, the proposed latch-based design opens up an additional architectural degree of freedom. In particular, multiple 1T1R inequality-storage elements can be multiplexed onto a single latch, enabling a shared-latch array organization. This reduces the number of required latches and therefore improves area efficiency. The tradeoff is that the associated inequalities must be evaluated sequentially, so increasing the degree of multiplexing increases latency. As a result, the architecture exposes a clear and tunable tradeoff between latency, area, and energy, allowing the array organization to be adapted to the requirements of the target application rather than being fixed a priori.

We show that this trade-off can be exploited in a task-aware manner and result in an Optimization problem. Combining analytical models with a circuit-accurate behavioral model derived from SPICE lookup tables in 22 nm FD-SOI technology, we optimize the sequential-parallel array partitioning using workload-dependent mismatch statistics. Across representative datasets, this task-specific tuning achieves up to 50 % search-energy reduction at 3× latency while maintaining high inference accuracy, and we further demonstrate robust parallel scaling to 120 cells per match line. Integrated into the X-TIME decision-tree compiler flow [4], with multiple decision-tree benchmarks, these results show that SALM is not only a more efficient aCAM cell, but also the basis of a tunable array architecture that can be adapted to the target application.

[1] Li, C. et al., in: Nature Communications, 111, 2020.

[2] Molom-Ochir, T. et al., in: IEEE Transactions on Circuits and Systems I: Regular Papers, 728, 3971–3982, 2025, 3971–3982.

[3] Pedretti, G. et al., in: Nature communications, 121, 5806, 2021, 5806.

[4] Pedretti, G. et al., in: IEEE Journal on Exploratory Solid-State Computational Devices and Circuits, 10, 116–124, 2024, 116–124.