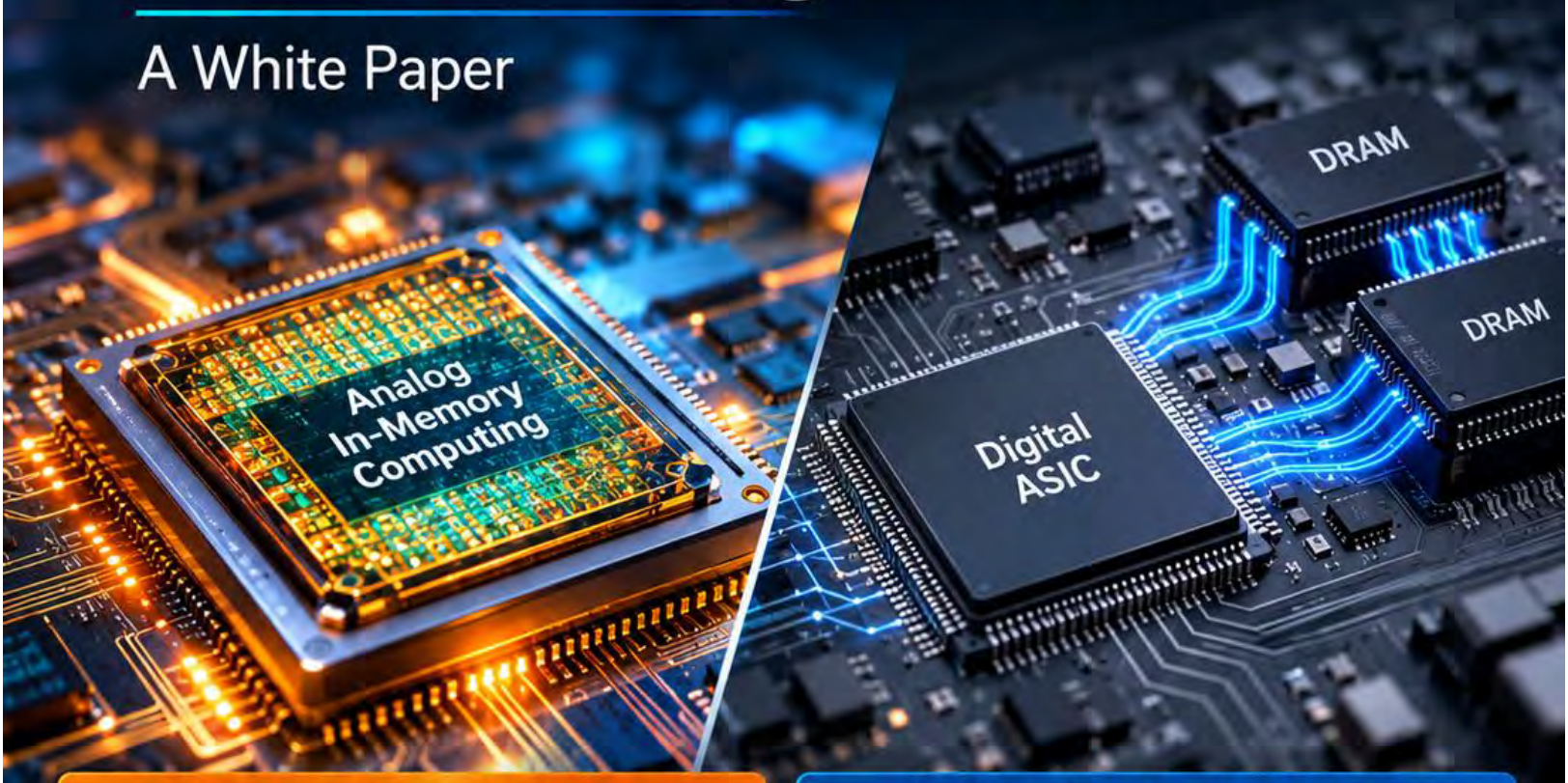
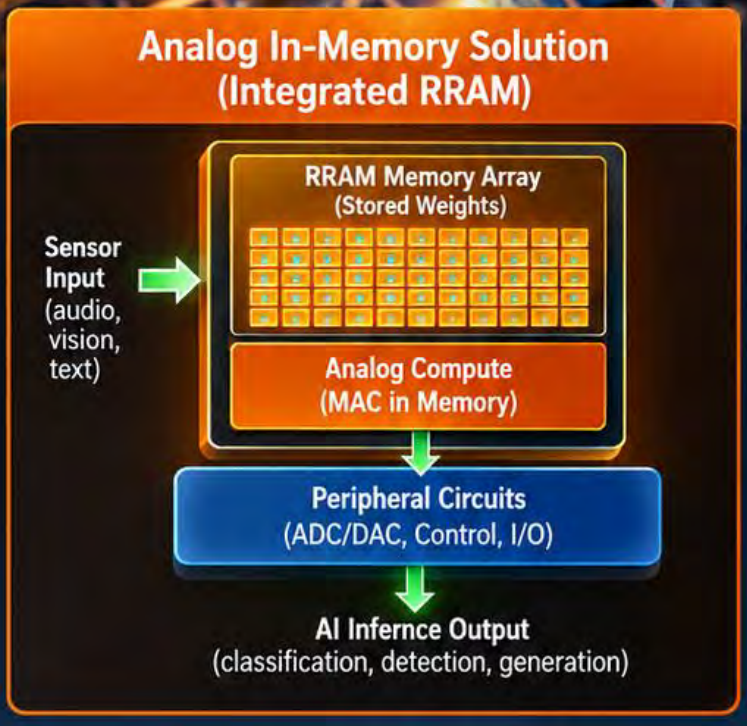
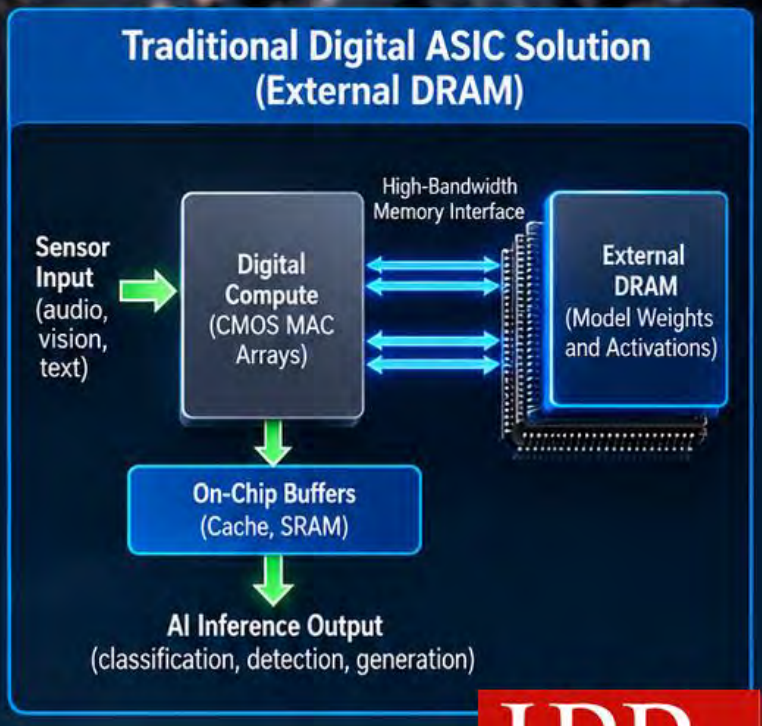
Rethinking Edge AI Costs: RRAM vs. Digital ASICs

A White Paper







JPR
Jon Peddie Research

Table of Contents

Rethinking Edge AI Costs: RRAM vs. Digital ASICs 3
Introduction 3
Summary 6

Table of Figures

Figure 1. Digital ASIC with optional external model memory. 4
Figure 2. RRAM arrays combine weight storage with analog computation. 5

Table of Tables

Table 1. Comparative cost factors for analog RRAM and digital ASIC architectures. 6

© Copyright Jon Peddie Research 2026

Rethinking Edge AI Costs: RRAM vs. Digital ASICs

Introduction

Edge AI economics depend on the complete system: processor silicon, memory, board area, energy, software, and manufacturing yield. Analog in-memory computing (A-IMC) places neural network weights in resistive random-access memory (RRAM) arrays and performs selected computations near those weights. Digital ASICs use established CMOS logic and memory hierarchies, often with external DRAM. Each architecture brings different cost, precision, integration, and sourcing considerations. This paper examines the conditions under which either approach can deliver a lower total cost of ownership.

The silicon economics of the memory bottleneck

Many digital AI accelerators separate arithmetic from substantial portions of model storage. Weight transfers across memory interfaces consume power and can limit throughput, particularly for repeated inference. The resulting system may require external DRAM, associated power management, routing, and packaging resources. Digital implementations can also keep smaller models entirely in on-chip SRAM or embedded memory; external DRAM is a design choice, not a universal requirement.

Advanced process nodes can improve density and energy efficiency, while mask sets, verification, IP, and engineering raise nonrecurring costs. Product teams should compare architectures at a common process maturity, workload, volume, and precision target. Per-chip costs alone do not capture the cost of the board, cooling, software integration, or lifecycle support.

How the digital ASIC solution works

A digital ASIC executes matrix operations in CMOS multiply-accumulate units. On-chip SRAM or caches stage activations and weights; an external DRAM interface supplies additional capacity when the model and intermediate data exceed local storage. Quantization, tiling, compression, and compiler scheduling can reduce memory traffic. Memory bandwidth, DRAM capacity, and software efficiency jointly determine practical system cost.

© Copyright Jon Peddie Research 2026

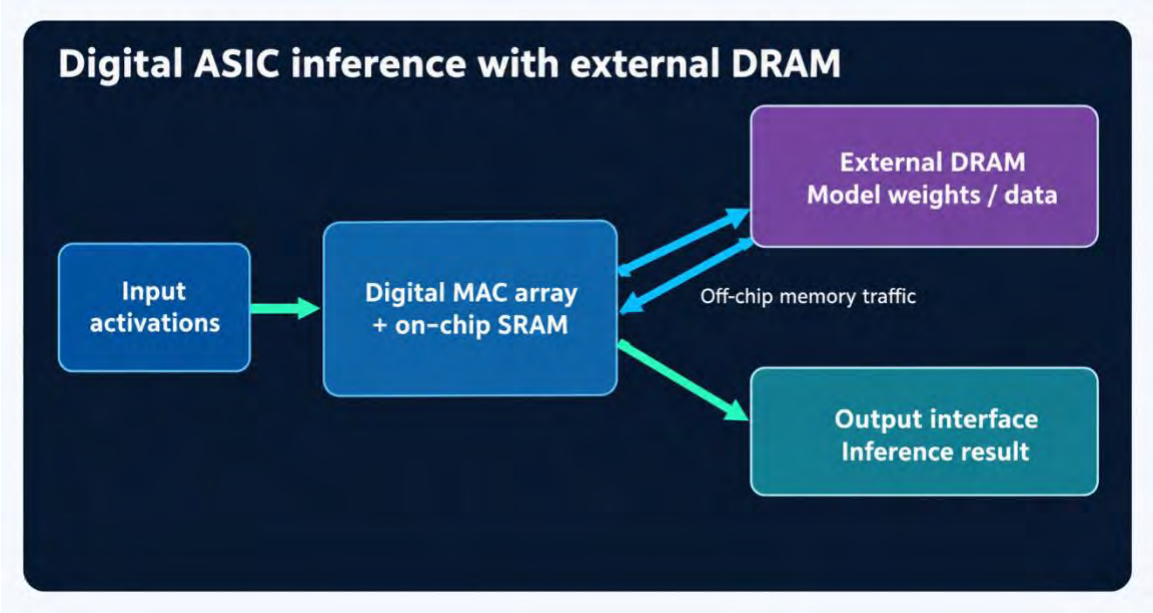


Figure 1. Digital ASIC with optional external model memory.

How the analog RRAM solution works

An RRAM A-IMC array stores weights as resistance states and uses electrical behavior to execute portions of matrix multiplication. The architecture reduces repeated weight movement for supported operators. Peripheral analog-to-digital conversion, control logic, activation storage, calibration, and error correction remain necessary. The effective accuracy and power advantage depend on device variation, drift, temperature, endurance, and workload mapping.

© Copyright Jon Peddie Research 2026

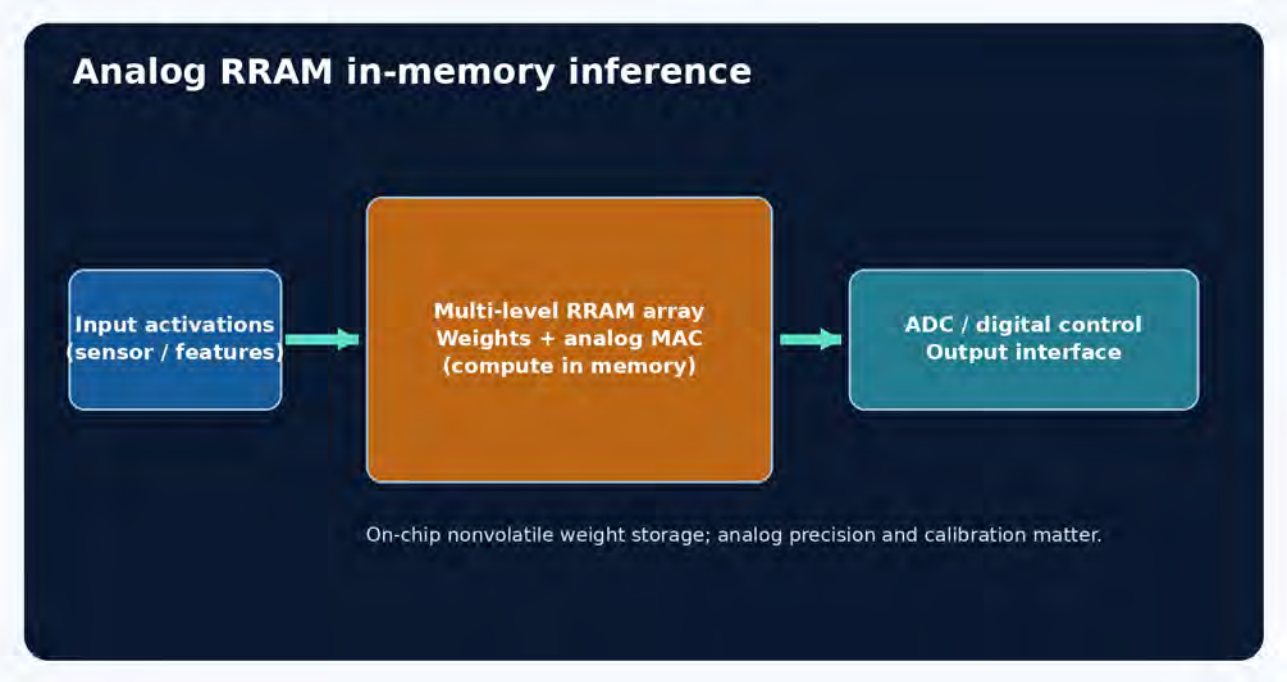


Figure 2. RRAM arrays combine weight storage with analog computation.

Bill of materials and system integration

An RRAM design may avoid dedicated external weight DRAM when its on-chip capacity accommodates the chosen model and when activations and buffering fit the remaining memory hierarchy. Larger workloads may still need off-chip memory. Reduced memory components can simplify PCB routing and power delivery, but converter circuits, testing requirements, packaging, and any RRAM process additions offset part of the savings. A digital ASIC can achieve a similarly compact design for small enough models.

Manufacturing, yield, and test

Conventional digital CMOS benefits from mature fault screening, characterization, and production infrastructure. Multi-level RRAM adds analog-state programming and validation. Foundries may require additional process steps, masks, and materials. Device variability and calibration can affect throughput, yield, and test time. Claims that the analog yield curve will converge with CMOS need production evidence at the intended process node and volume.

Economic comparison

The economics of edge AI processors extend beyond the cost of manufacturing the silicon. Memory requirements, packaging, PCB complexity, power consumption, thermal management, and production testing all contribute to the total cost of ownership. Analog RRAM and digital ASIC architectures distribute these costs differently. RRAM integrates weight storage and computation, potentially reducing external memory requirements and associated system costs.

© Copyright Jon Peddie Research 2026

Digital ASICs benefit from established manufacturing processes, mature design tools, and predictable production yields. The following table compares the principal cost factors and their implications for system designers and enterprise deployments.

Dimension	RRAM A-IMC	Digital ASIC	Decision metric
NRE and process	Potential added memory-process development and qualification	Established CMOS flows; node choice affects masks and IP	Total engineering and mask spend
Memory BOM	Can reduce external weight memory for fitting models	External DRAM optional, depending on model and SRAM	System BOM at target workload
Board and packaging	Potentially fewer memory interfaces	May need DRAM routing and power rails	Board area, layers, assembly cost
Test and calibration	Analog programming, drift, and precision validation	Mature digital production tests	Yield-adjusted test cost
Energy and thermal	Lower weight movement; ADC overhead remains	Optimized digital MACs; memory traffic varies	Measured joules/inference and cooling
Software	Operator mapping, quantization, toolchain maturity	Established compilers and frameworks vary by vendor	Porting effort and lifecycle support

Table 1. Comparative cost factors for analog RRAM and digital ASIC architectures.

Operating life and environmental considerations

System power affects cooling, enclosure size, power delivery, and operating expense. A-IMC can reduce energy consumed by weight transfers, yet its analog interfaces and conversion circuitry draw power. Digital accelerators can exploit clock gating, memory compression, and low-power modes. Cooling requirements follow measured system dissipation and ambient conditions. No architecture inherently guarantees passive cooling or longer service life.

Strategic sourcing and production risk

CIOs and procurement teams should assess process availability, foundry qualification, supply continuity, reliability, model support, and software portability. Standard digital flows generally offer a broader set of qualified manufacturing and verification options. Specialized RRAM implementations may constrain sourcing and require additional qualification. Pilot deployments should measure application latency, inference accuracy, system power, field stability, and total delivered cost.

Summary

The economic question is whether the reduction in data movement and external memory outweighs the additional process, calibration, integration, and software costs. A fair comparison holds model accuracy, throughput, duty cycle, deployment conditions, and manufacturing volume constant. RRAM A-IMC warrants evaluation for suitable edge inference workloads;

© Copyright Jon Peddie Research 2026

digital ASICs remain a strong reference because of their process maturity and broad software support.

RRAM in-memory computing offers a credible path to lower memory-transfer energy for targeted inference. The value proposition depends on fit: model size, supported operators, precision, and system configuration. Buyers should request measured joules per inference, yield-adjusted costs, and software demonstrations. A common benchmark suite and full system BOM will reveal the commercial trade-offs.

Growing interest in memory-centric inference points toward a potential inflection point in edge AI architecture. Production-worthy RRAM arrays, repeatable accuracy, mature compilation tools, and qualified manufacturing would make the approach easier to deploy. Architectural promise alone does not establish a market transition. Evidence will come from independent system benchmarks, production yields, multi-year reliability results, and commercial deployments that show lower total cost at equal application quality.

© Copyright Jon Peddie Research 2026

© Copyright Jon Peddie Research 2026. All rights reserved.

Reproduction in whole or in part is prohibited without written permission
from Jon Peddie Research.

This report is the property of Jon Peddie Research (JPR) and made available to a restricted number of clients only upon these terms and conditions.

Agreement not to copy or disclose. This report and all future reports or other materials provided by JPR pursuant to this subscription (collectively, "Reports") are protected by: (i) federal copyright, pursuant to the Copyright Act of 1976; and (ii) the nondisclosure provisions set forth immediately following.

License, exclusive use, and agreement not to disclose. Reports are the trade secret property exclusively of JPR and are made available to a restricted number of clients, for their exclusive use and only upon the following terms and conditions. JPR grants site-wide license to read and utilize the information in the Reports, exclusively to the initial subscriber to the Reports, its subsidiaries, divisions, and employees (collectively, "Subscriber"). The Reports shall, at all times, be treated by Subscriber as proprietary and confidential documents, for internal use only. Subscriber agrees that it will not reproduce for or share any of the material in the Reports ("Material") with any entity or individual other than Subscriber ("Shared Third Party") (collectively, "Share" or "Sharing"), without the advance written permission of JPR.

No material from JPR may be used for AI training. The term "AI training" refers to the process of teaching a machine learning model to make predictions or decisions based on data. That includes but is not limited to: 1. Supervised Learning: Training with labeled data (input-output pairs), such as Image classification, spam detection, sentiment analysis. 2. Unsupervised Learning: Training with unlabeled data to find patterns or groupings, such as Clustering (e.g. K-means), dimensionality reduction (e.g. PCA). 3. Self-Supervised Learning: A hybrid where the system generates labels from the data itself, such as Contrastive learning, masked language modeling (used in GPT, BERT). 4. Reinforcement Learning (RL): Agents learn by taking actions and receiving rewards or penalties, such as Game-playing AIs (e.g. AlphaGo), robotics, navigation. 5. Unsupervised Reinforcement Learning: Agents explore environments without a defined task or reward, such as Pretraining agents for general skills before fine-tuning. 6. Multi-Task Learning: Training one model to handle multiple tasks simultaneously, such as A model that can translate and summarize text. 7. Transfer Learning: Using a pre-trained model on a new, related task with minimal retraining, such as Fine-tuning GPT or BERT on a specific domain. 8. Few-Shot / One-Shot / Zero-Shot Learning, Few-shot: Trained on a very small number of examples. One-shot: Trained on just one example. Zero-shot: Performs tasks without any task-specific training. 9. Meta Learning ("Learning to Learn"): Training models that can adapt to new tasks with very little data, such as Model-Agnostic Meta-Learning (MAML). 10. Curriculum Learning: Training models by gradually increasing task difficulty. 11. Online Learning: Model updates continuously as new data arrives, such as Stock prediction systems. 12. Federated Learning: Training across decentralized devices without sharing raw data, such as Mobile keyboard prediction models. 13. Contrastive Learning: Learns by comparing positive and negative pairs of data. Used In: Self-supervised vision models like SimCLR, CLIP. Specialized AI Training Terms, Fine-tuning: Adjusting a pre-trained model on a new dataset. Pretraining: Initial broad training, usually on large datasets (e.g. GPT pretraining). Prompt Tuning / Instruction Tuning: Adjusting models using instructions or task-specific prompts. RLHF (Reinforcement Learning from Human Feedback): Used in GPT models to align behavior with human intent. Distillation: Training a smaller model to mimic a larger model. Active Learning: Selecting the most informative data points to label next. Semi-supervised Learning: Mix of labeled and unlabeled data for training.

Subscriber shall be liable for any breach of this agreement and shall be subject to cancellation of its subscription to Reports. Without limiting this liability, Subscriber shall be liable for any damages suffered by JPR as a result of any Sharing of any Material, without advance written permission of JPR.

Important Note re Sharing of Material. If Subscriber wishes to Share any Material with any Shared Third Party, it shall contact JPR and request permission to do so. JPR shall forthwith prepare an appropriate Sharing agreement, which shall set forth the name and contact information of the Shared Third Party, the terms and conditions of such Sharing, the compensation to JPR for such Sharing, and the agreement by Subscriber and the Shared Third Party that the Shared Third Party shall not, under any circumstances, Share any of the Material with any other entity or person.

Disclaimers. JPR Reports contain "reviews" of various products and information. The content of these Reports represent the interpretation and analysis of statistics and information that are either generally available to the public or released by entities or individuals deemed responsible. The information in the Reports is believed to be reliable but is not guaranteed as to its accuracy or completeness. The Reports do not endorse or attempt to sell any products and should not be interpreted differently. JPR assumes no responsibility for the correct or incorrect usage of any trademarks or service marks.

© Copyright Jon Peddie Research 2026