# Cover Letter

AACIM: A 2785-TOPS/W, 161-TOP/mm$^2$, $< 1.17\%$-RMSE, Analog-In Analog-Out Computing-In-Memory Macro in 28 nm

Wan, R.; Lin, C.-T.; Zhang, B.; Tsividis, Y.; Seok, M. — IEEE JSSC, 2025

**General Topic.** The authors study analog-in analog-out (AIAO) computing-in-memory (CIM) for vector–matrix multiplication (VMM), with the goal of reducing energy and latency by avoiding repeated Digital-to-Analog/Analog-to-Digital (DAC/ADC) conversions in conventional mixed-signal CIM pipelines.

**Specific Activity.** The work builds a 28 nm AIAO CIM macro that performs $64\times32$ VMM using analog inputs and analog outputs, and proposes three circuit techniques: self-calibration writing (CSCW), discharge-current stabilization (DCS), and retention enhancement—to control accuracy under mismatch and PVT variation.

**Research Questions.**

- **RQ1** How can an AIAO CIM macro achieve low VMM error (reported via normalized RMSE / error statistics) without DAC/ADC conversion?
- **RQ2** How effectively do CSCW and DCS reduce programming/read non-idealities and improve robustness across supply and temperature?
- **RQ3** Is analog-state retention sufficient for practical deployment, and what refresh interval is implied by measured drift/hold time?

**Problem.** While analog CIM can be extremely energy-efficient, practical accuracy is limited by (i) programming variability of analog weight states, (ii) data-dependent discharge-current nonlinearity during read/compute, and (iii) finite analog-state retention. In AIAO CIM, these non-idealities directly distort the analog output, so the system must suppress both static error and drift while preserving the energy advantages of eliminating conversion overhead.

**Claims.** The authors claim that (1) an AIAO CIM macro can deliver sufficiently accurate VMM while removing DAC/ADC conversion overhead; (2) CSCW reduces write/programming error by closing a feedback loop that calibrates the stored state to a target discharging current; (3) DCS stabilizes the discharge current by regulating the read-bitline voltage with feedback, reducing output-dependent error spread; and (4) retention enhancement yields a usable hold time (on the order of hundreds of $\mu$s), enabling refresh-based operation.

**Evidence.** The authors support the claims with silicon measurements of (i) VMM transfer behavior and normalized RMSE (typical and worst-case), (ii) error statistics (e.g., output-dependent standard deviation $\sigma$) to quantify nonlinearity and level dependence, (iii) robustness sweeps across supply and temperature to show worst-case accuracy bounds, and (iv) measured retention/hold-time behavior across multiple weight codes and multiple chips. They further connect circuit-level non-idealities to application impact by using a measured-error-derived model (near-zero-mean error with output-dependent $\sigma$, plus drift based on retention) to estimate inference accuracy for quantized networks.

**Statistical Analysis.** The paper primarily uses descriptive experimental analysis rather than formal inferential statistics, mirroring the style in the course template: it reports quantitative error metrics (normalized RMSE and error $\sigma$), compares them across PVT conditions and operating modes, and summarizes retention behavior across codes/chips to bound drift and motivate refresh intervals. Task-level implications are evaluated via an explicit error/drift model driven by measured statistics rather than by hypothesis testing.