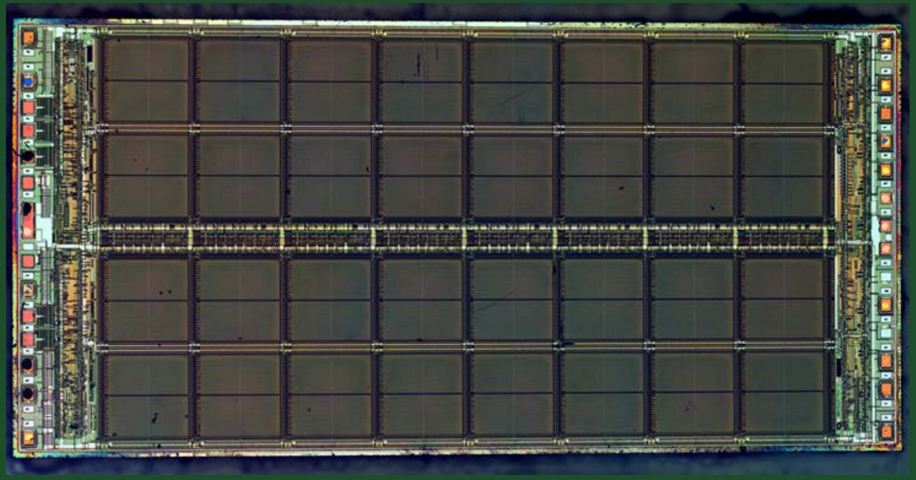
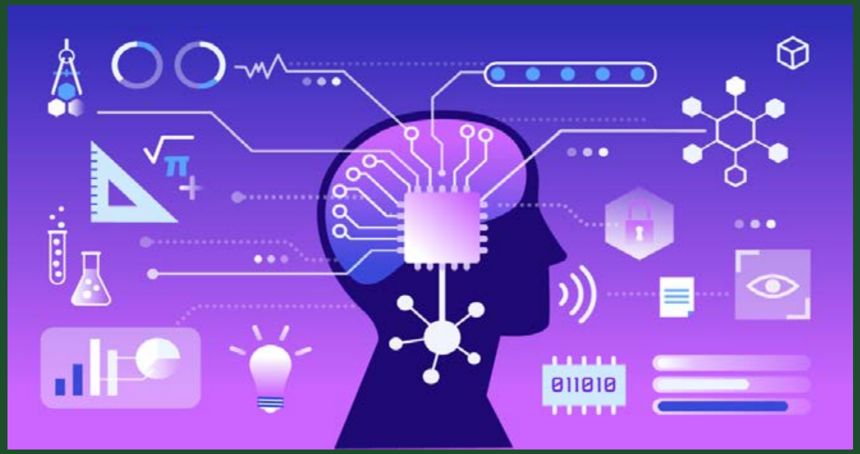
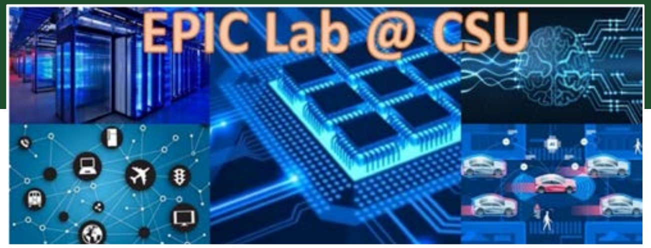
Stochastic In-DRAM Acceleration of LLMs



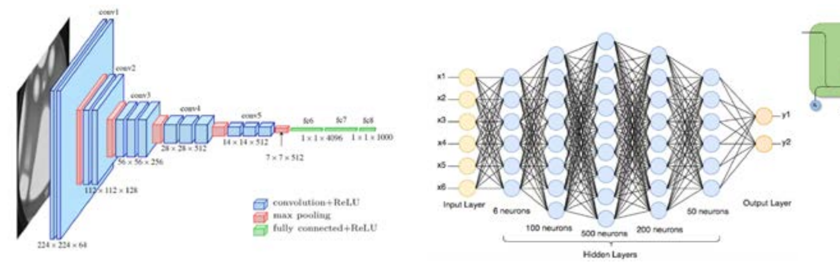
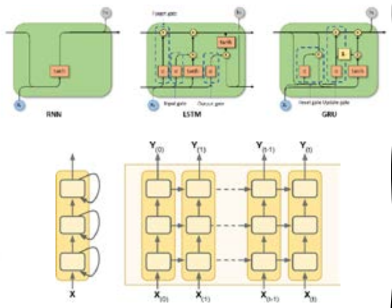




MPSoC Workshop 2025

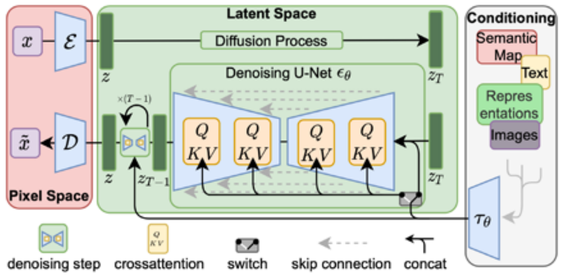
Sudeep Pasricha
Aram and Helga Budak Professor
FIEEE, FAAIA, FAIIA, ACM Distinguished Member
Director, Embedded, High Performance, and Intelligent Computing (EPIC) Lab
Department of Electrical and Computer Engineering
Colorado State University, Fort Collins, CO, USA

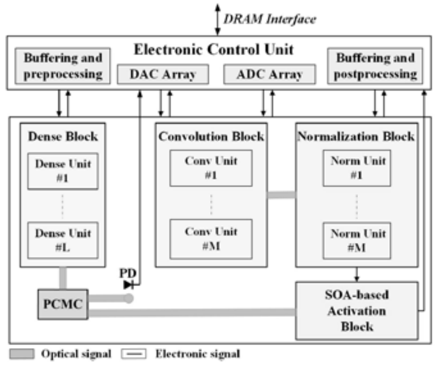
Recent Work: Family of Optical AI Accelerators

CNN, DNN, RNN Accelerators

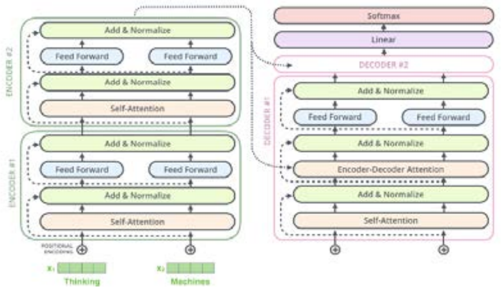
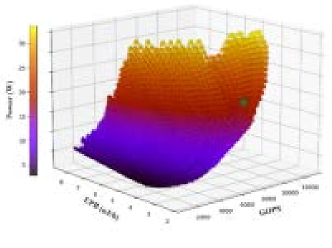
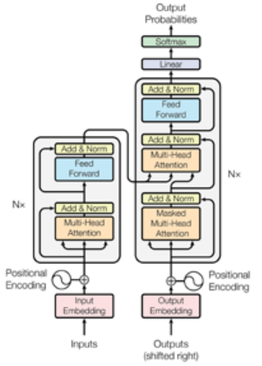
- F. Sunny, M. Nikdast, S. Pasricha, "Cross-Layer Design for AI Acceleration with Non-Coherent Optical Computing", ACM GLSVLSI, 2023.
- F. Sunny, M. Nikdast and S. Pasricha, "RecLight: A Recurrent Neural Network Accelerator With Integrated Silicon Photonics", IEEE ISVLSI, 2022.
- F. Sunny, A. Mirza, M. Nikdast, S. Pasricha, "CrossLight: A Cross-Layer Optimized Silicon Photonic Neural Network Accelerator", IEEE/ACM DAC, 2021.





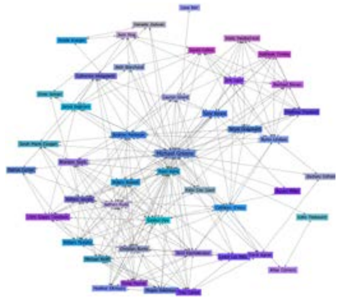
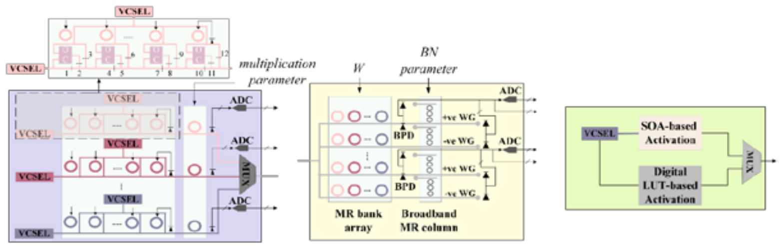
Generative AI Accelerators

- T. Suresh, S. Afifi, S. Pasricha, "Diffusion Neural Network Acceleration with Silicon Photonics" under review, 2025.
- T. Suresh, S. Afifi, S. Pasricha, "PhotoGAN: Generative Adversarial Neural Network Acceleration with Silicon Photonics" IEEE ISQED, 2025.

LLM/Transformer Accelerators

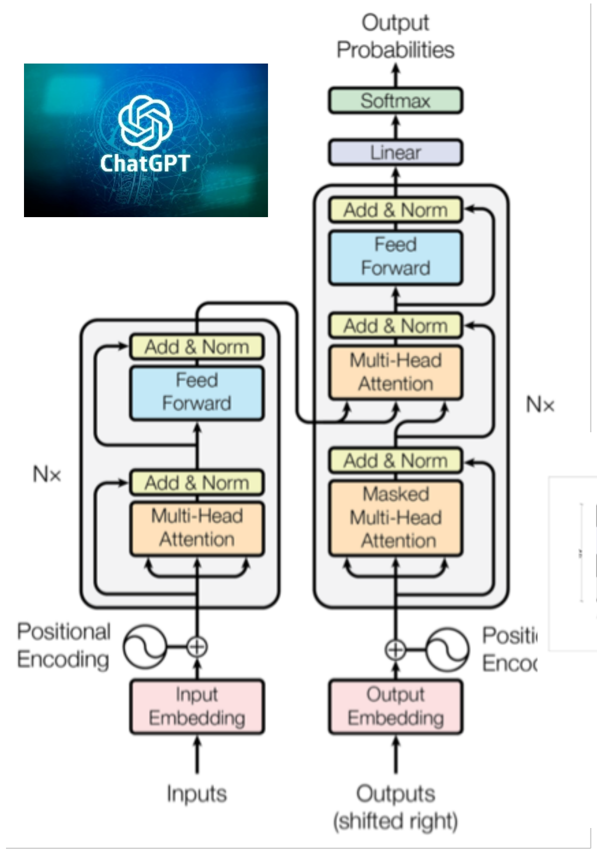
- S. Afifi, S. Pasricha, M. Nikdast, "Shedding Light on LLMs: Harnessing Photonic Neural Networks for Accelerating LLMs", IEEE ICCAD, Nov 2024.
- S. Afifi, F. Sunny, M. Nikdast, S. Pasricha, "TRON: Transformer Neural Network Acceleration with Non-Coherent Silicon Photonics", ACM GLSVLSI, 2023.

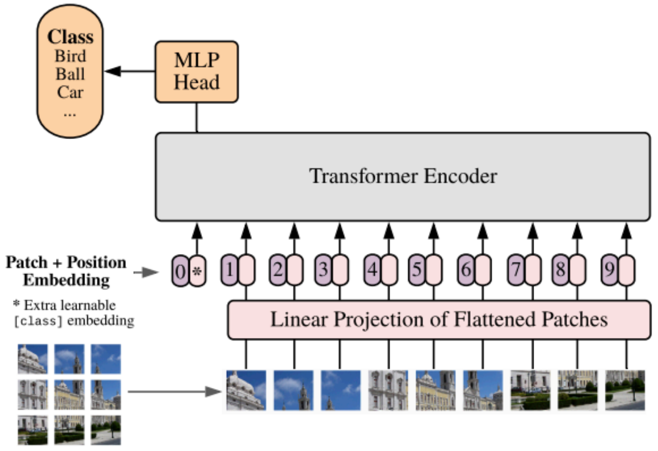



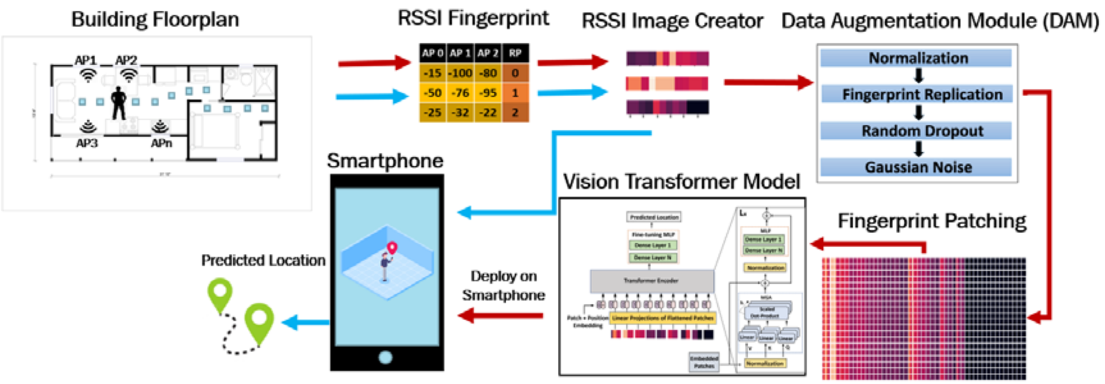
Graph Network Accelerators

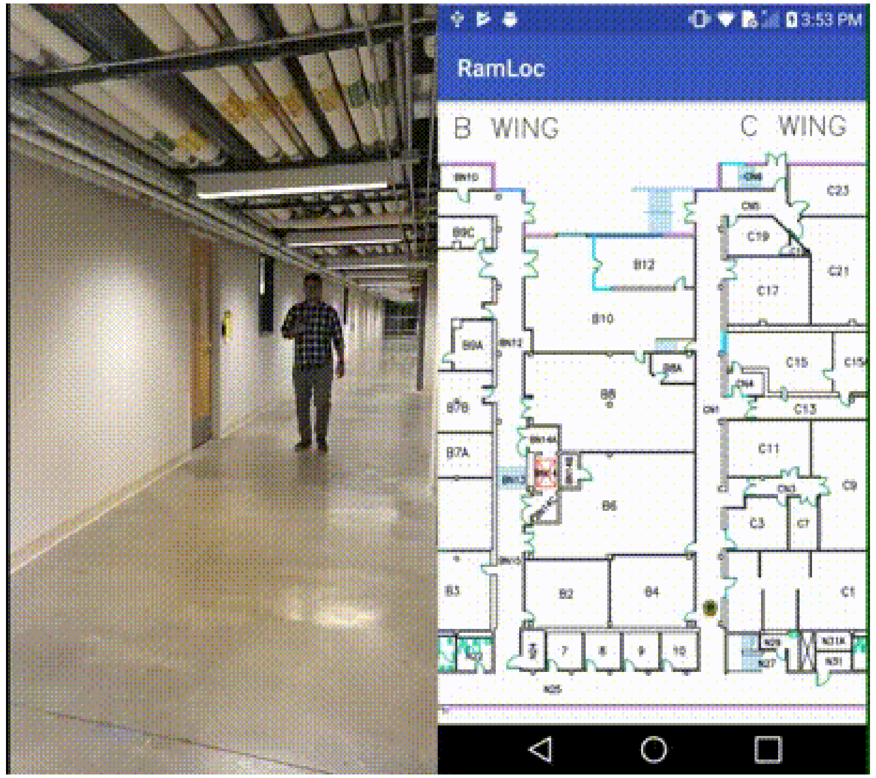
- S. Afifi, F. Sunny, M. Nikdast, S. Pasricha, "Accelerating Neural Networks for Large Language Models and Graph Processing with Silicon Photonics", IEEE/ACM DATE, 2024
- S. Afifi, F. Sunny, A. Shafiee, M. Nikdast, S. Pasricha, "GHOST: A Graph Neural Network Accelerator using Silicon Photonics", ACM TECS (ESWEEK), 2023.

Transformer Neural Networks









A. Singampalli, D. Gufran, S. Pasricha, "[CIELO: Class-Incremental Continual Learning for Overcoming Catastrophic Forgetting with Smartphone-based Indoor Localization]", ***IEEE Access, 2025***

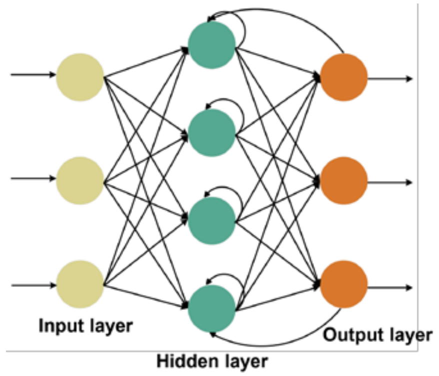
D. Gufran, S. Tiku, S. Pasricha, "[STELLAR: Siamese Multi-Headed Attention Neural Networks for Overcoming Temporal Variations and Device Heterogeneity with Indoor Localization]", ***IEEE Journal of Indoor and Seamless Positioning and Navigation, 2024.***

D. Gufran, S. Tiku, S. Pasricha, "[VITAL: Vision Transformer Neural Networks for Smartphone Heterogeneity Resilient and Accurate Indoor Localization]", ***IEEE/ACM Design Automation Conference (DAC), Jul 2023.***

Transformer Neural Networks

Recurrent Neural Networks

- Designed to processes sequential data
- Drawbacks:
  Vanishing gradient: its “memory” not that strong when remembering old connections

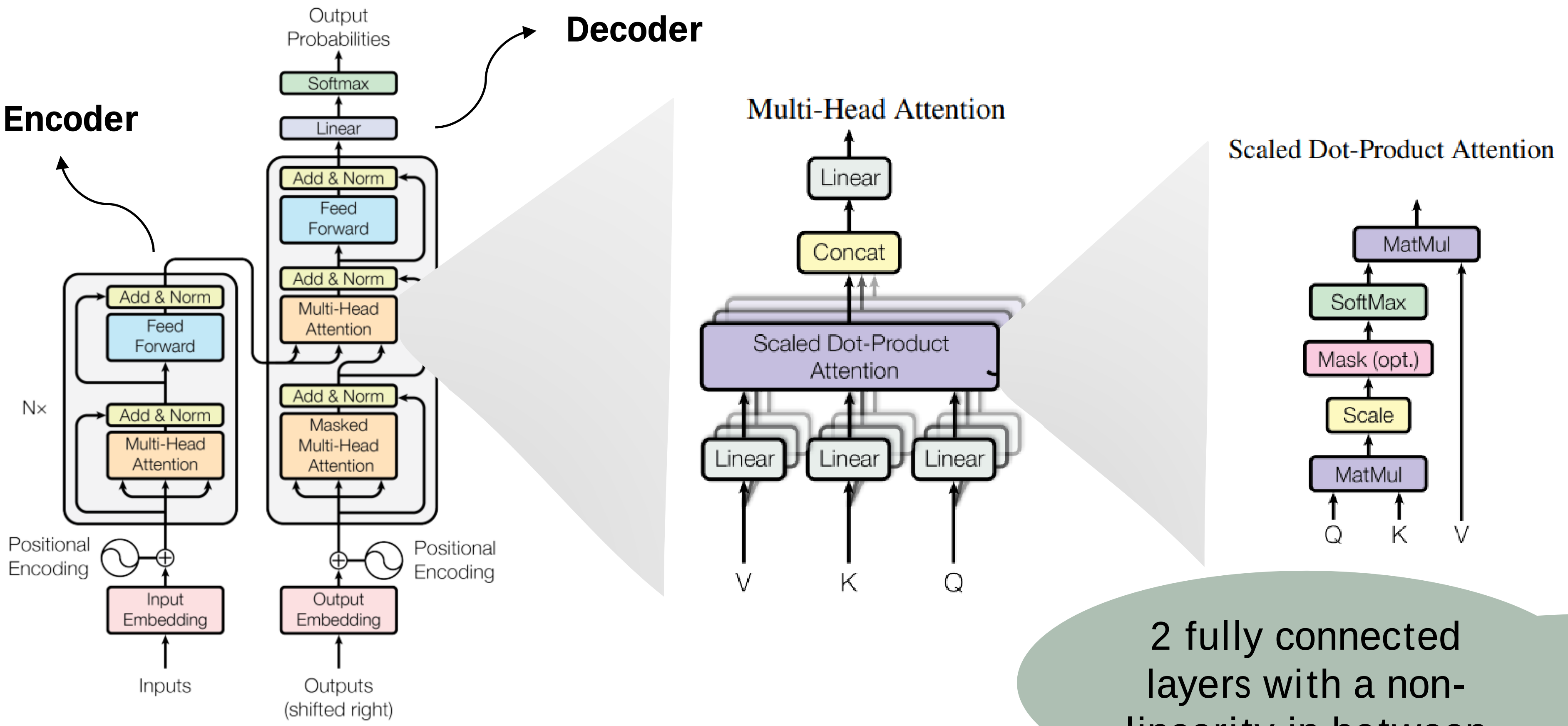


Transformer Neural Networks

- Uses the attention mechanism
- Swiftly established as the model of choice for NLP problems
- Being integrated into vision tasks
- Already implemented in many prominent applications
- Challenge:
  Transformers can be massive, requiring high computational and data movement support

Google Translate
ChatGPT
duolingo
TESLA

Transformer Model Acceleration



$Q = XW_Q$

$K = XW_K$

$V = XW_V$

$$Head(X) = Attention(Q, K, V) = Softmax\left(\frac{KQ^T}{\sqrt{d_K}}\right).V$$

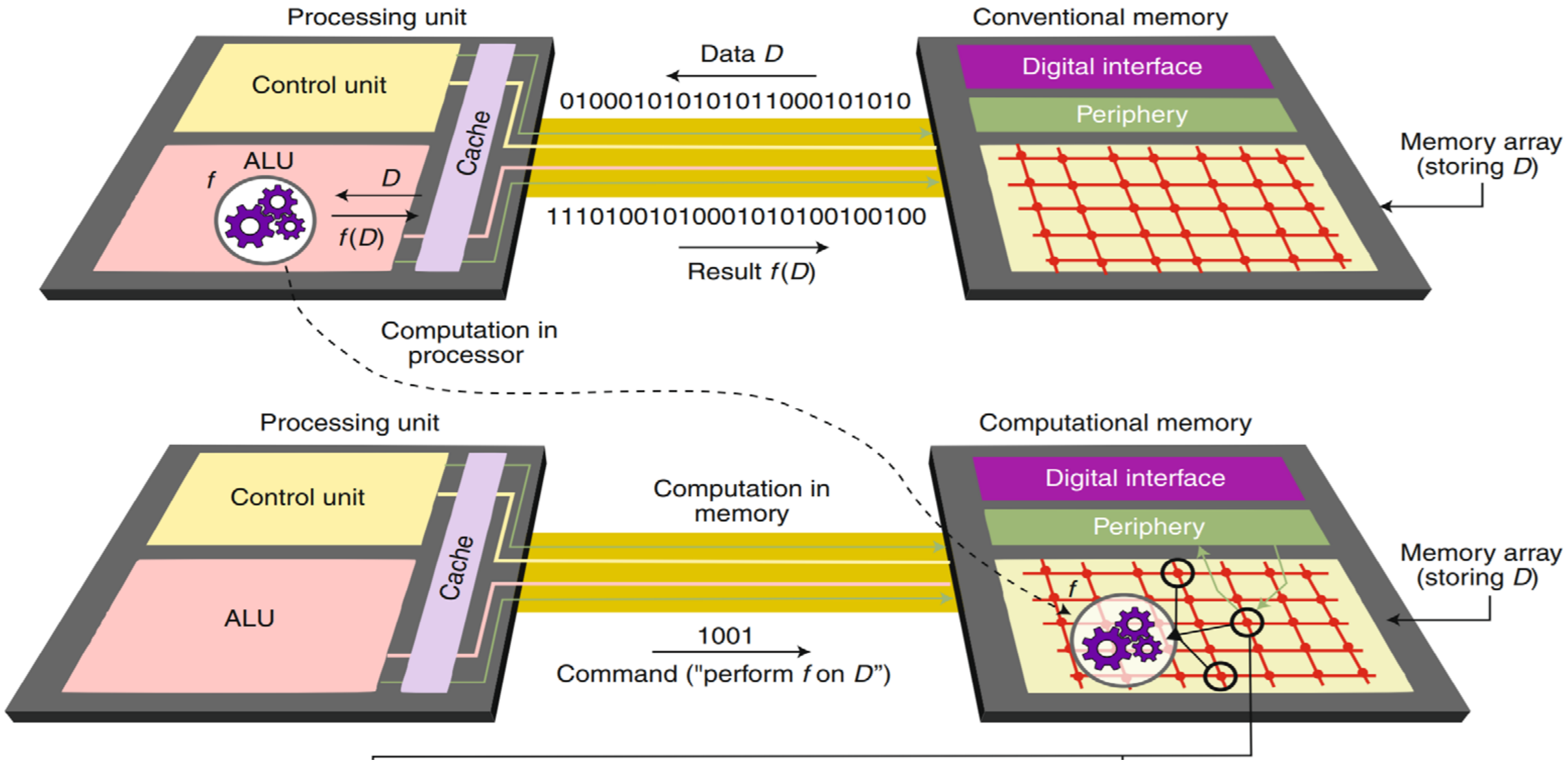
$$FF(Y) = RELU(YW_1 + b_1)W_2 + b_2$$

2 fully connected layers with a non-linearity in between

- Acceleration challenges
  - Larger parameter counts than other neural network types, e.g., CNNs, RNNs, …
  - Quadratic scaling of memory and computational demands with sequence length in self-attention
  - Larger batch sizes exacerbate these overheads even further

In-DRAM Computing

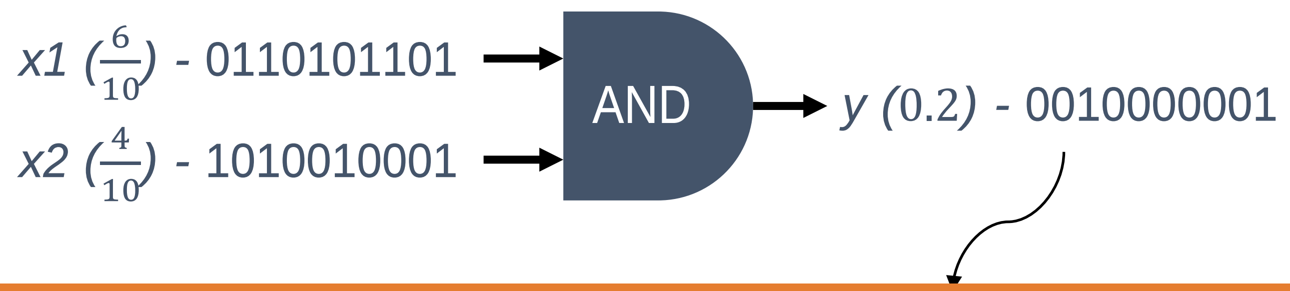
- Minimizing data movement by computing closer to where data resides (inside DRAM) can significantly benefit high memory usage transformer workloads



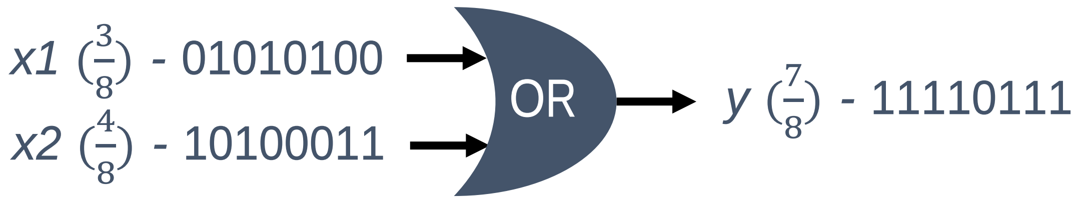
Stochastic Computing

- In stochastic computing, numbers are represented by the probability of the appearance of “1”s in the bitstream
- An n-bit binary number → represented stochastically by a 2^n element vector
- **Complex operations** can be done using **simple logic gates**
  - Amenable to simplified implementations, crucial for in-DRAM acceleration

Multiply Operation



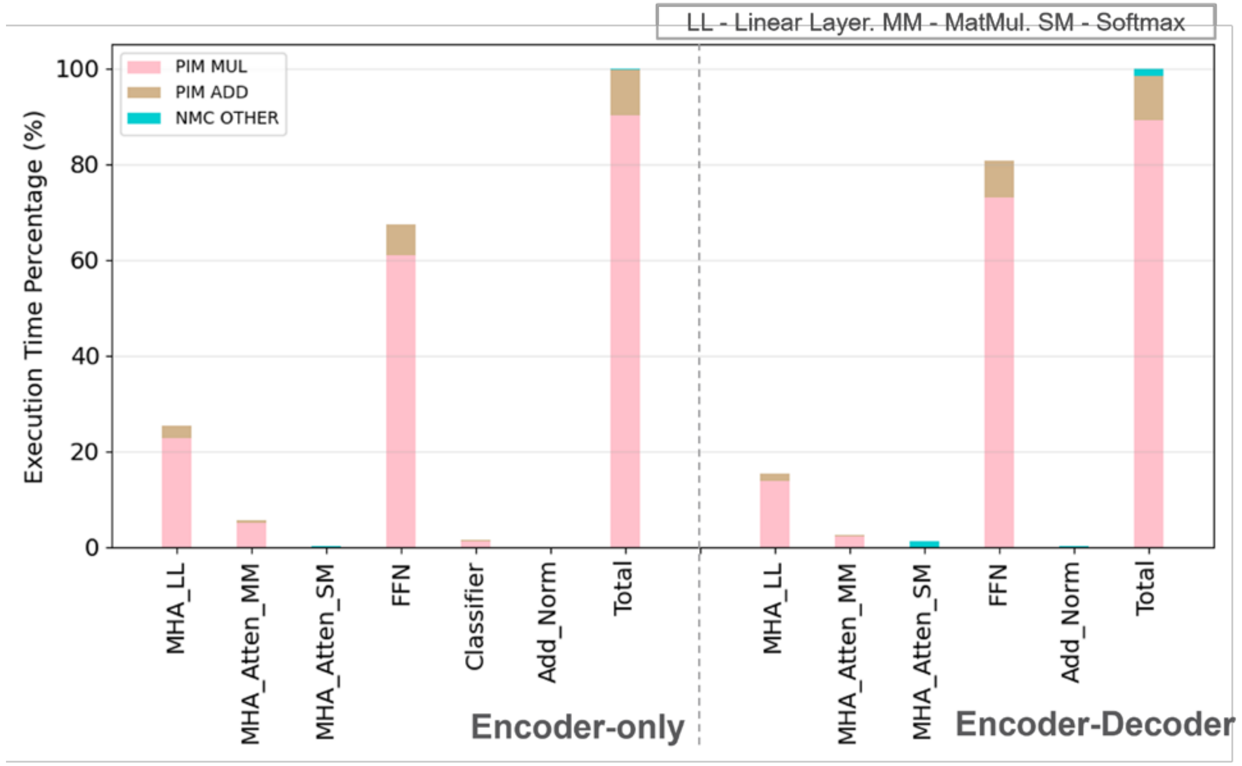
Addition Operation



Integrating stochastic computing with in-DRAM computing offers an interesting new approach to accelerate LLMs

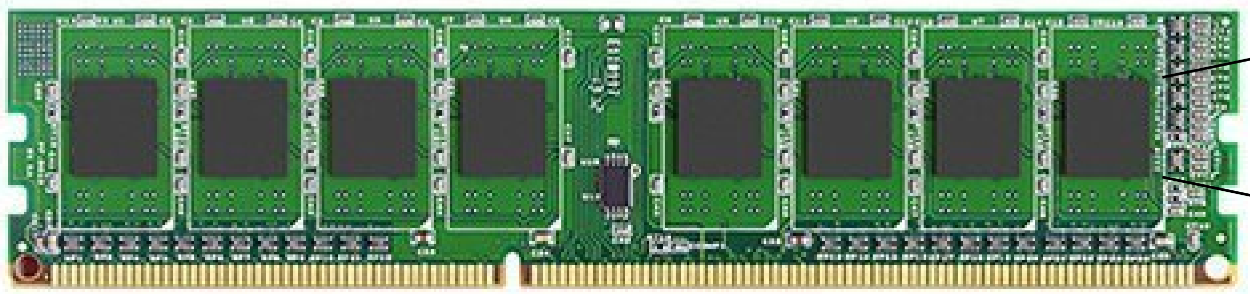
Stochastic In-DRAM Acceleration Challenges

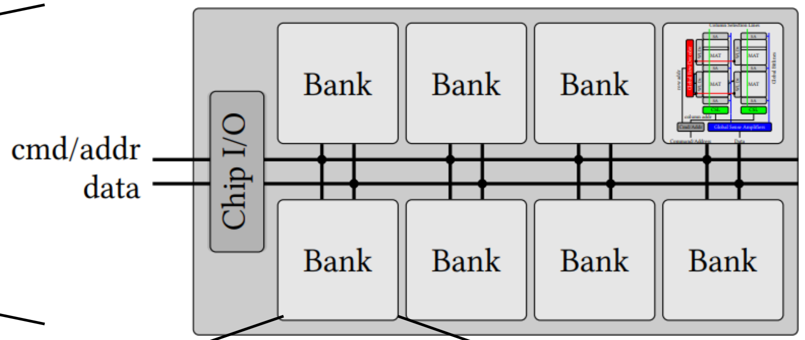
- **Implementing MAC operations within DRAM**
  - Existing implementations decompose MAC into multiple functionally complete memory operation cycles (MOCs; activate-activate-precharge)
  - Long latencies; a single MUL takes 1600ns in DRISA [S. Li et al;. MICRO 2017]



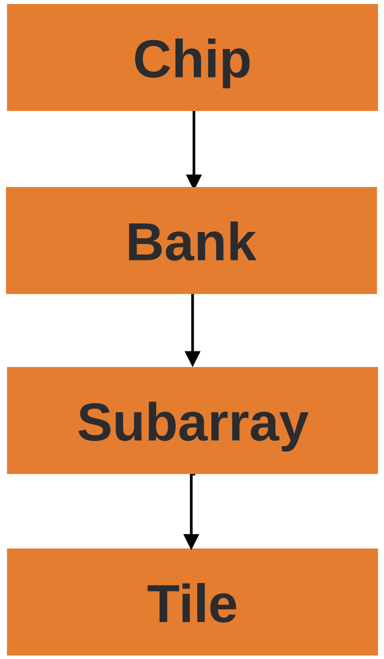
- **Storage Overhead**
  - SC requires $O(2^N)$ storage overhead as representing an N-bit real value requires 2^N bits
  - Can reduce parallelism
- **Stochastic Computational Error**
  - Can impact the overall inference accuracy
  - Trade-off exists between accuracy and hardware resources for encoding/decoding
- **Stochastic-to-Binary (S_to_B) Conversion**
  - Frequent S_to_B conversions are needed
  - Pop-Count (PC) creates several challenges related to area, power and latency

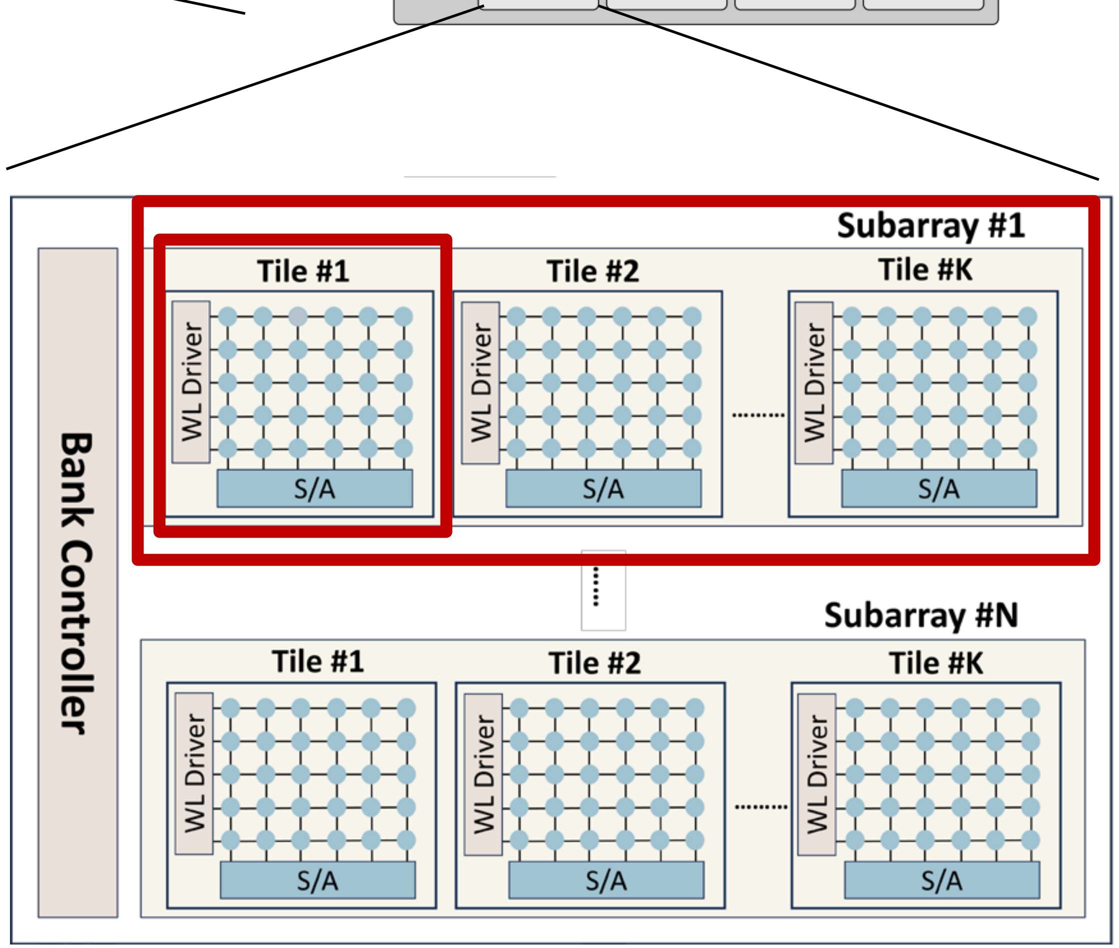
Background: DRAM Structure





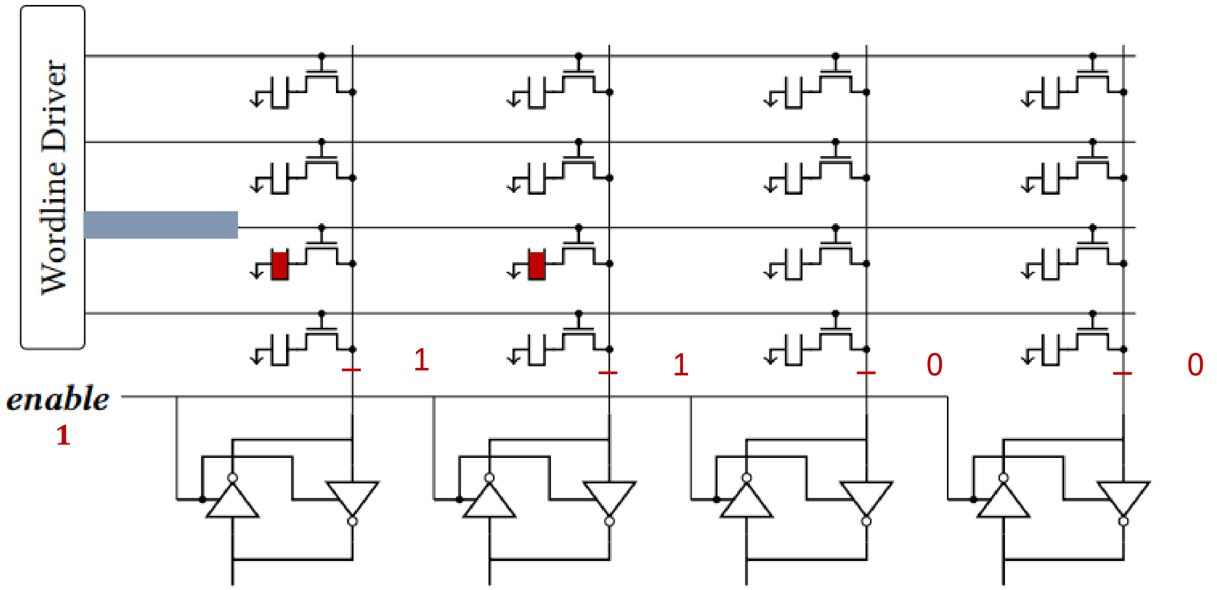
- A DRAM chip has a hierarchical architecture:





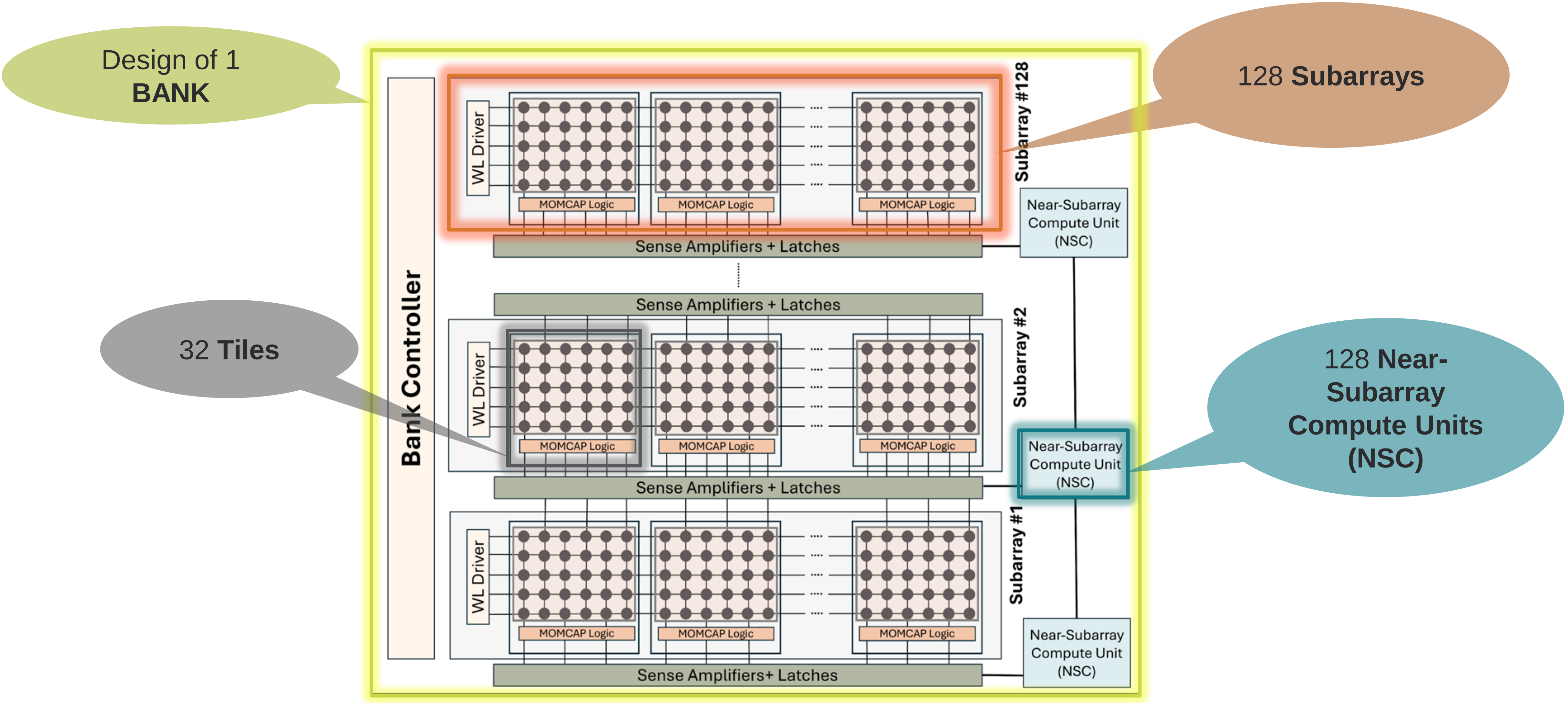
Background: DRAM Operation

- Read Operation
  1. *Pre-charge stage*:
     - bit-lines are pre-charged to $\frac{vdd}{2}$
  2. *Activate stage*:
     - Target cells are activated using the word-lines control signals (WL)
     - **Charge sharing phase**: charge is distributed between the cell and bit-line capacitance
     - **Sense amplifier (SA)** is activated to detect and amplify the subtle voltage variation
     - **Restore phase**: The sensed voltage variation is then amplified by the SA and reinstated to the target cells



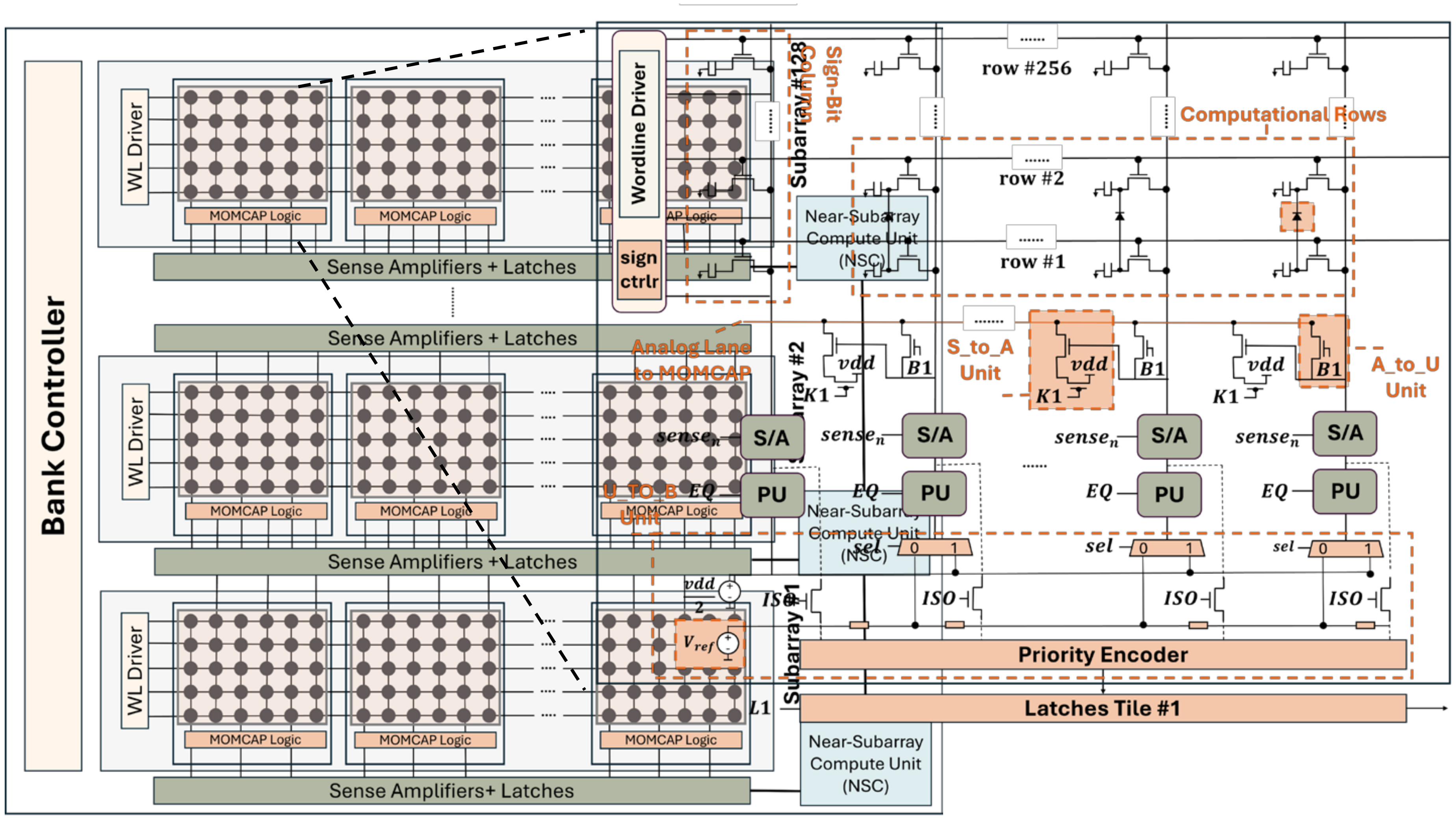
- Write Operation
  - SAs read and amplify data from the DRAM chip's internal bus
  - Charge is written to target cells during **Restore Phase**

ARTEMIS Architecture



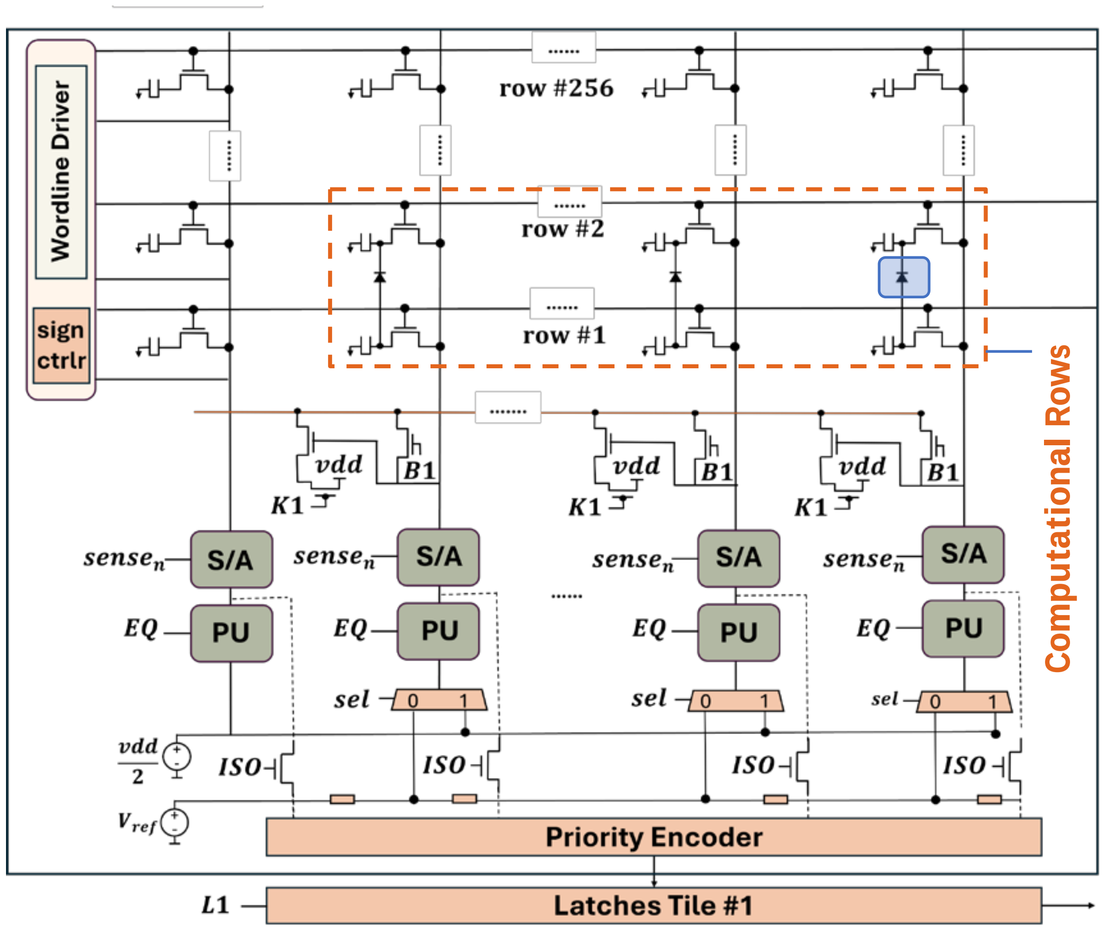
S. Afifi, I. Thakkar, S. Pasricha, "ARTEMIS: A Mixed Analog-Stochastic In-DRAM Accelerator for Transformer Neural Networks", ***IEEE/ACM CASES (ESWEEK), Oct 2024.***

ARTEMIS Tile Architecture



MAC Operation

- **Multiplications**
  - Main Challenge: output precision
  - Solution: deterministic stochastic multiplication technique using transition-coded-unary (TCU) numbers
    - TCU number: stochastic number with all the 1's grouped (0000111111)
    - MAE: 0.039, Max. Error: 0.123



1) Copy operand 1 to computation row #1

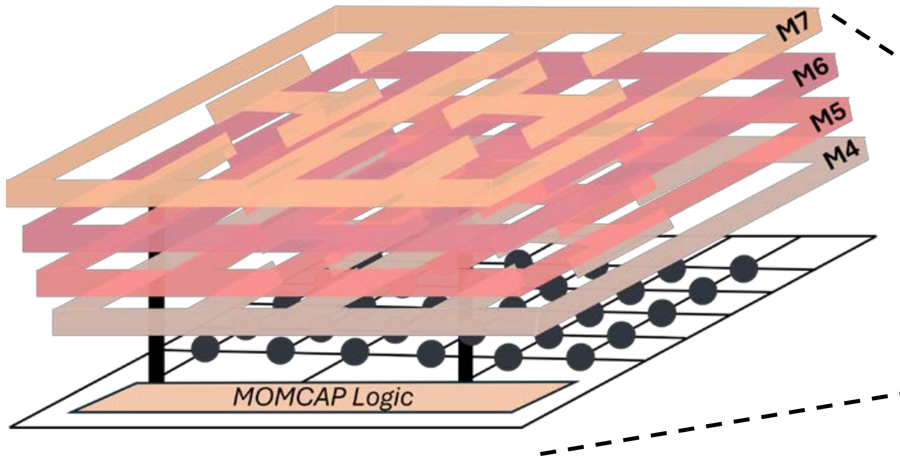
2) Copy operand 2 to computation row #2

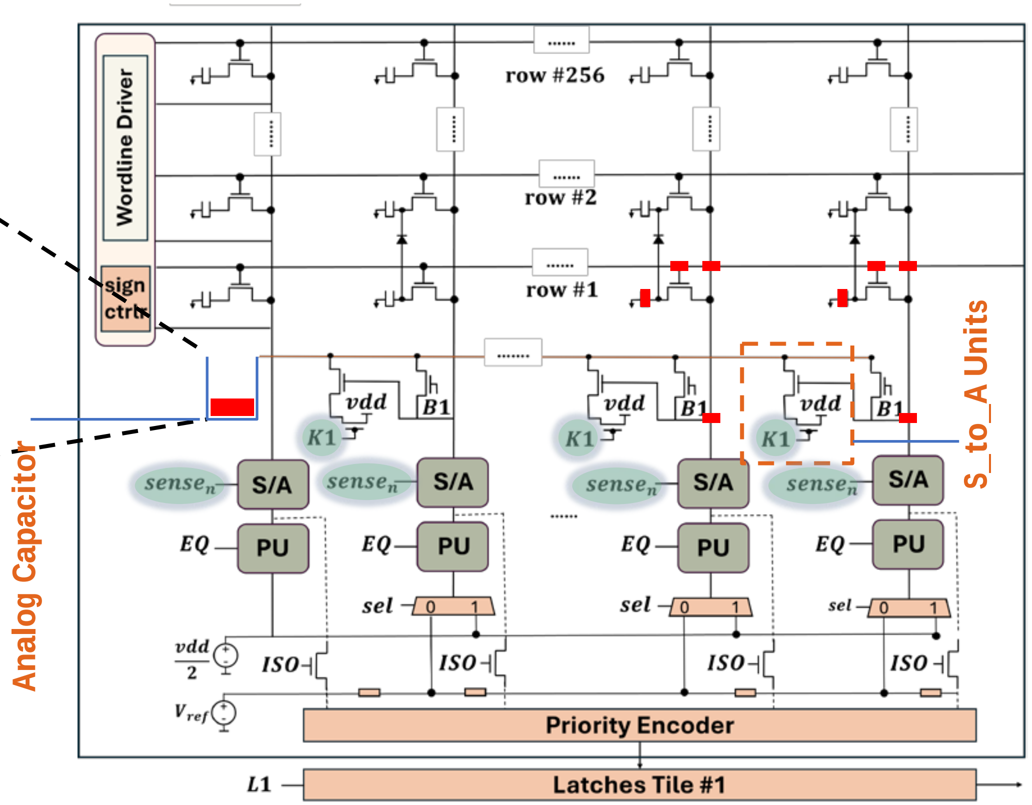
AND (stochastic multiply) operands using in-DRAM computing and result is stored in row #1

Multiplication Performed in two MOCs

S/A – Sense Amplifier.
PU – Precharge Unit.
TCU – transition-coded-unary

MAC Operation

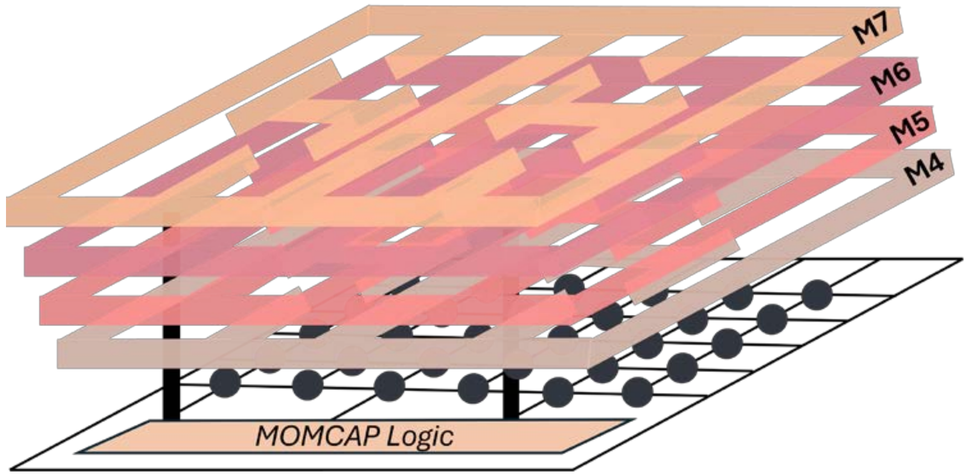




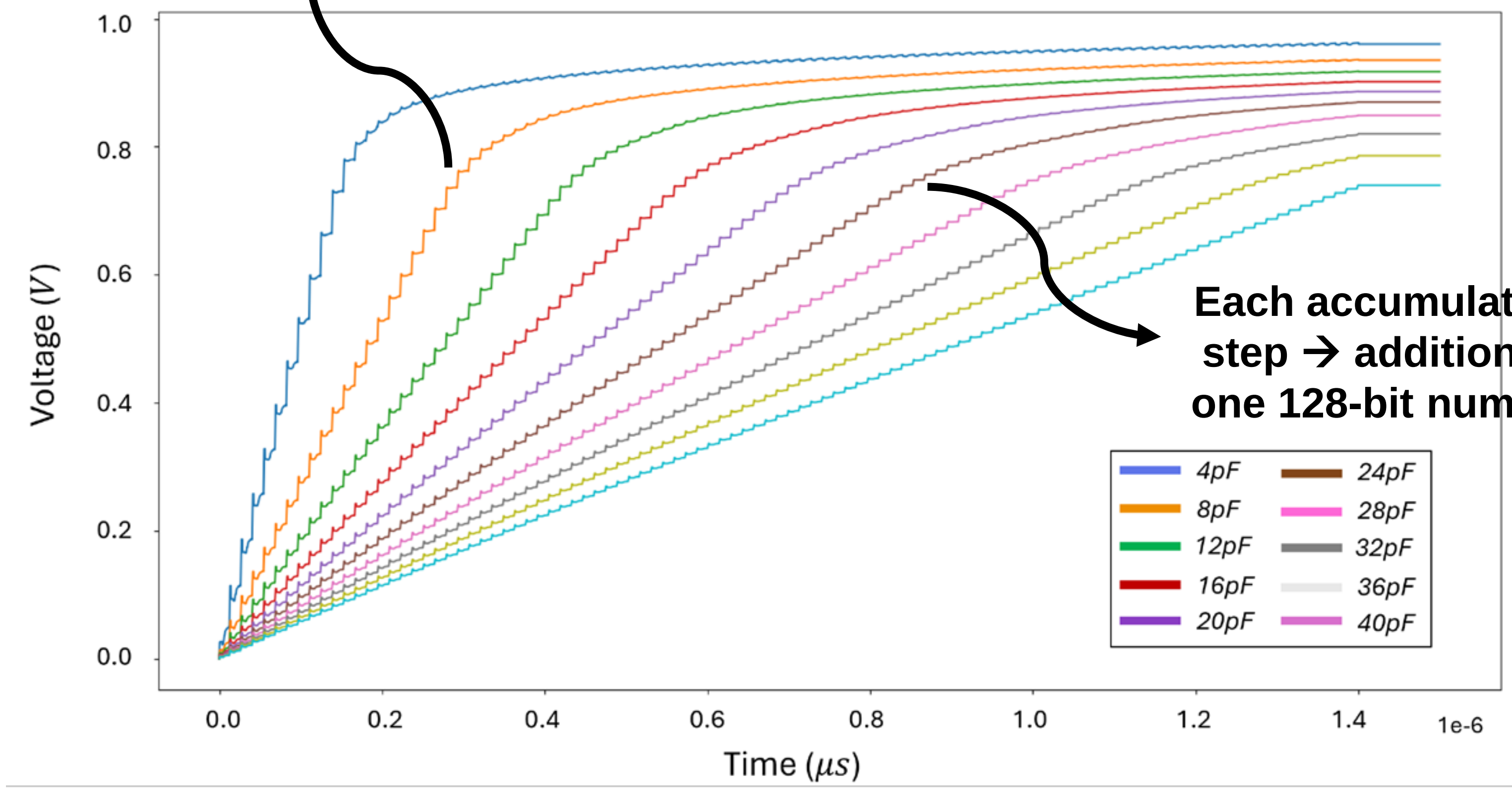
- **Accumulations**
  - Main Challenges: Stochastic-based addition introduces large errors
  - Solution: temporal analog accumulations
    - MUL outputs are accumulated using analog capacitor Using stochastic-to-analog (*S_to_A)* unit
    - Charge on the capacitor → corresponds to the number of '1's
    - Using H-shaped MOMCAP
    - Area of MOMCAP = Tile Area
    - MAE: 0.00085
    - Max. Error: 0.0729

S/A – Sense Amplifier.
PU – Precharge Unit.
S_to_A – Stochastic-to-Analog.
MOMCAP – Metal-Oxide-Metal Capacitor

MOMCAP Design



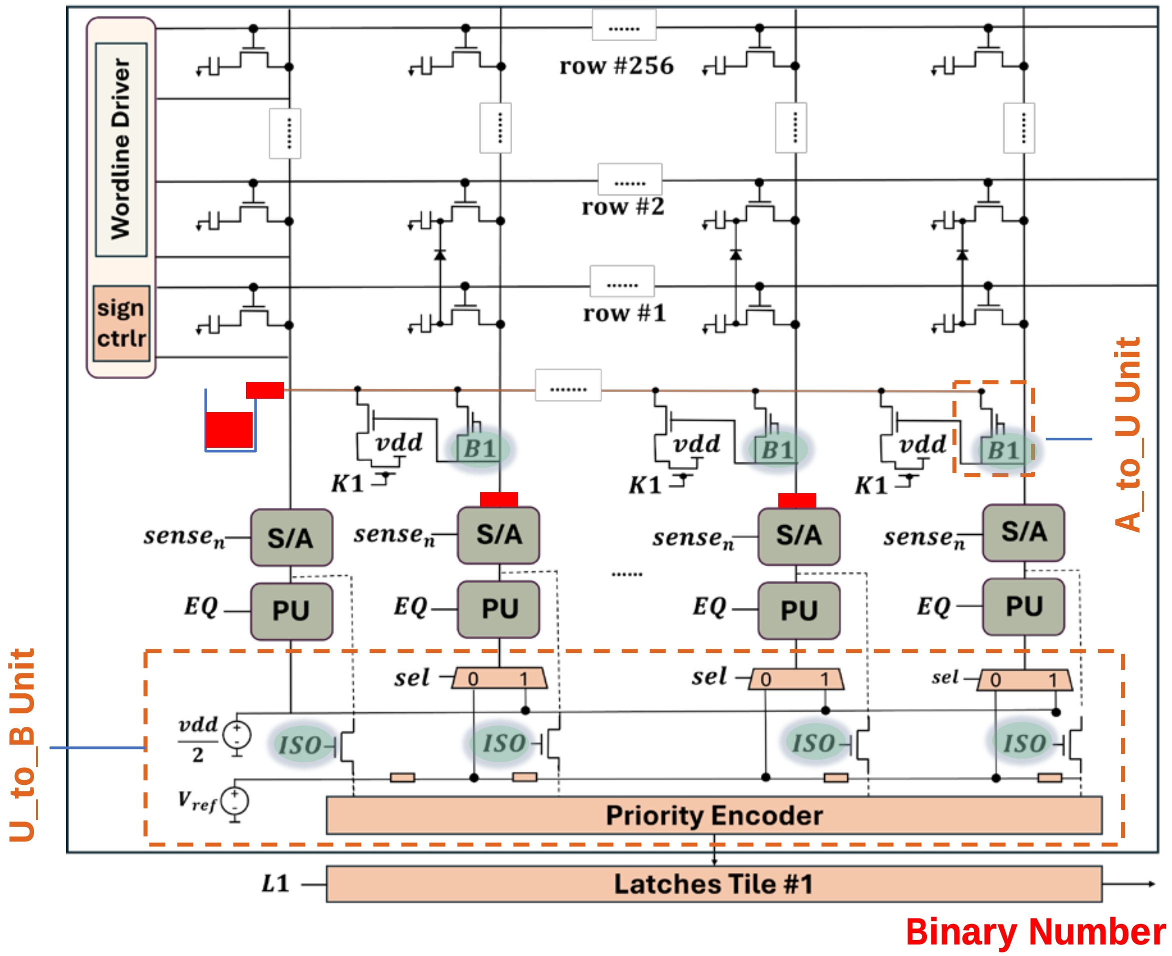
- We analyzed the voltage behavior of charge accumulation on the MOMCAP across a spectrum of capacitance values
  - Modeled and simulated 128 bit-lines alongside the tile's circuits utilizing LTSPICE
  - Increased capacitance enhances the capacitor's ability to accommodate a greater number of accumulations
  - But higher capacitance leads to a larger area overhead
  - We selected a MOMCAP size aligning with ARTEMIS' tile area of 338µm^2, which corresponds to an 8pF capacitance
  - This enables the accumulation of 20 consecutive dot products per MOMCAP



Analog to Binary Data Conversion

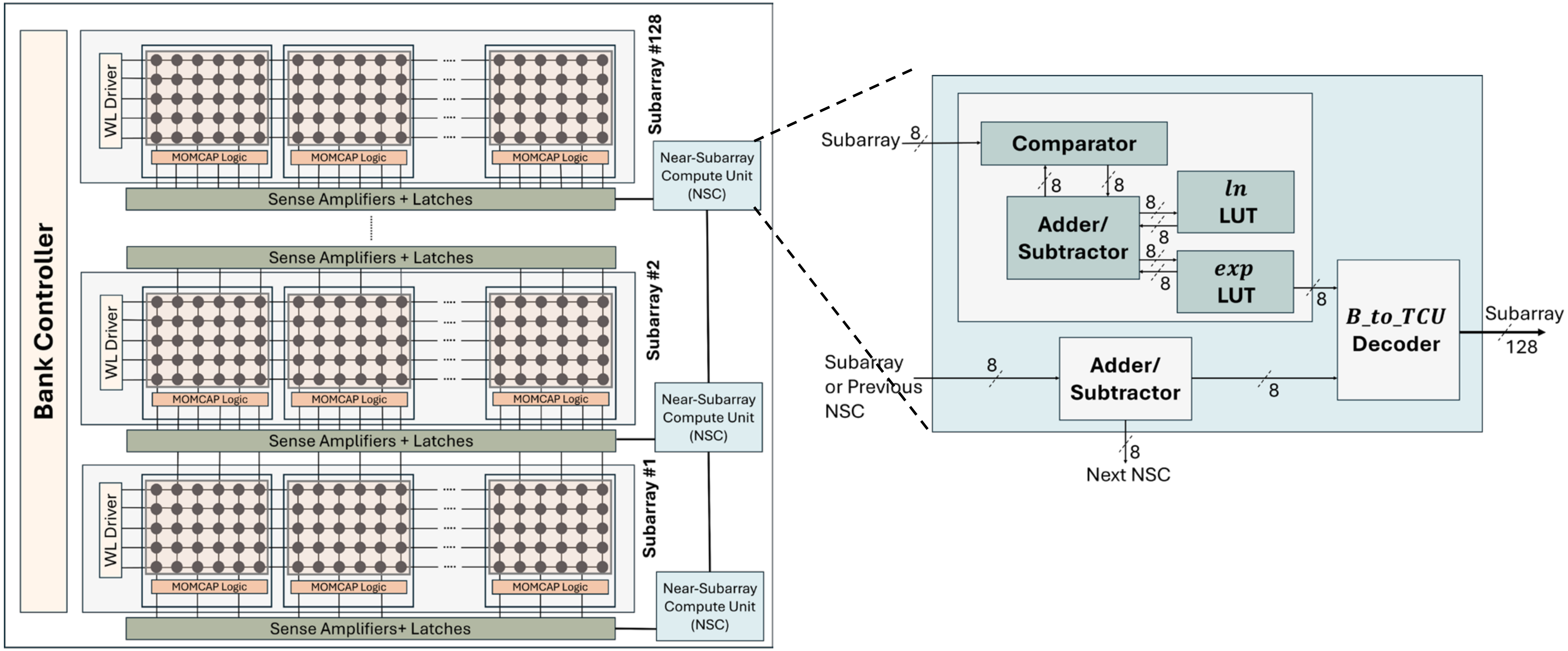
- Analog data stored in MOMCAPs need to be converted to binary numbers
  - **Analog-to-unary (A_to_U) unit:**
    - Toggle *B1* to connect MOMCAP to tile's bit-lines
  - **Unary-to-binary (U_to_B) unit:**
    - S/As are repurposed as voltage comparators Priority encoder generates binary number
    - MAE: 0.00037
    - Max. Error: 0.0062

ARTEMIS efficiently mitigates unary-to-binary (U_to_B) data conversion challenges



S/A – Sense Amplifier.
PU – Precharge Unit.
A_to_U – Analog-to-Unary.

ARTEMIS: Near-Subarray Compute Unit

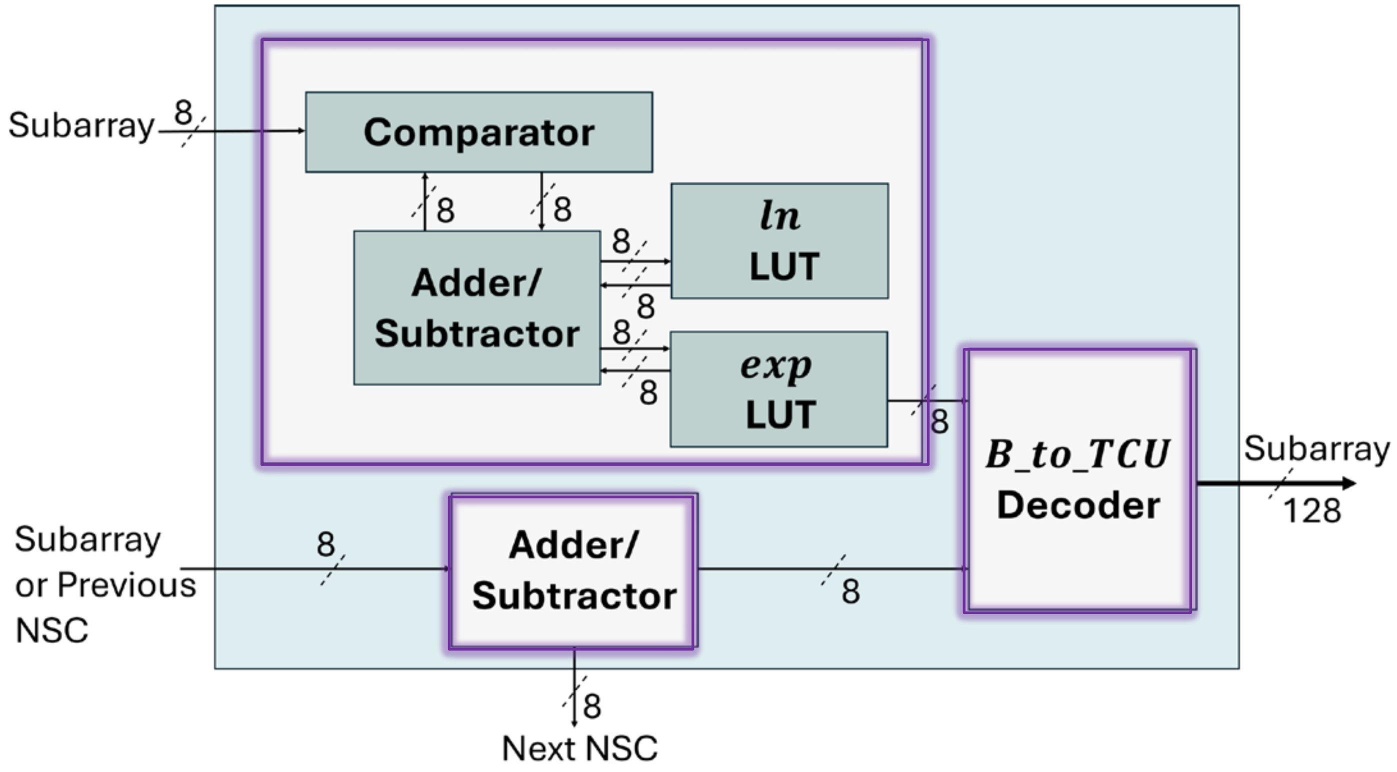


Near-Subarray Compute Unit (NSC)

1. Reduction Operations
   - Addition of partial sums
2. Softmax
   - Comparator
   - Adder/subtractor
   - ln LUT
   - exp LUT
3. B_to_TCU Decoder
   - Prepare inputs to next operations/layers

$$Softmax(y_i) = \frac{\exp(y_i - y_{max})}{\sum_{j=1}^{D} \exp(y_j - y_{\max})},$$

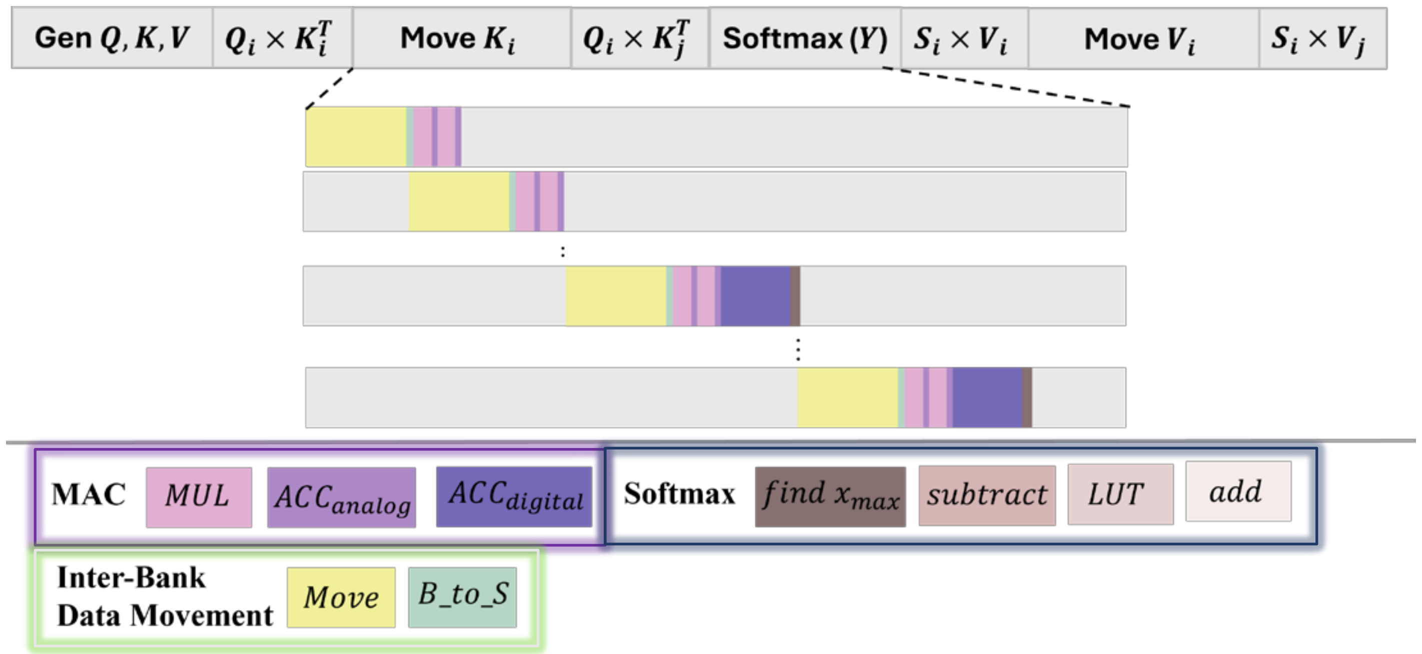
$$= \boxed{\exp}\left(y_i \boxed{-} \boxed{y_{max}} \boxed{-} \boxed{\ln} \left(\boxed{\sum_{j=1}^{D}} \boxed{\exp}\left(y_j \boxed{-} \boxed{y_{max}} \right) \right) \right),$$



B_to_TCU – binary to transition-coded-unary.
LUT – look-up table.

Execution Pipeline

- Execution bottleneck: inter-bank data movement
- ARTEMIS pipelines the following:
  - In-situ MAC operations within the DRAM tiles,
  - Data movement using the row of latches
  - NSC units' operations



ARTEMIS efficiently hides latencies of MAC and NSC operations

Experimental Setup

Model	Params	Layers	N	Heads	dmodel	dff
Transformer-base	52M	2	128	8	512	2048
BERT-base	108M	12	128	12	768	3072
Albert-base	12M	12	128	12	768	3072
ViT-base	86M	12	256	12	768	3072
OPT-350	350M	12	2048	12	768	3072

- **Detailed simulation-based analysis for each model and dataset**
  - **Software mapping**
    - Simulate layer-wise mapping for each transformer model and dataset.
  - **Hardware mapping**
    - Modeled all hardware components and in-DRAM operations
    - Area estimates → using CACTI
    - Per-tile circuits latencies → using LTSPICE
- **Five Transformer models considered**
  - **8-bit quantization (128-bit for SC) used**
- **Comparison to state-of-the-art accelerators**
  - TRANSPIM, ReBERT, HAIMA, FPGA_ACC, TPU, CPU, GPU

Results: Computational Error and Accuracy

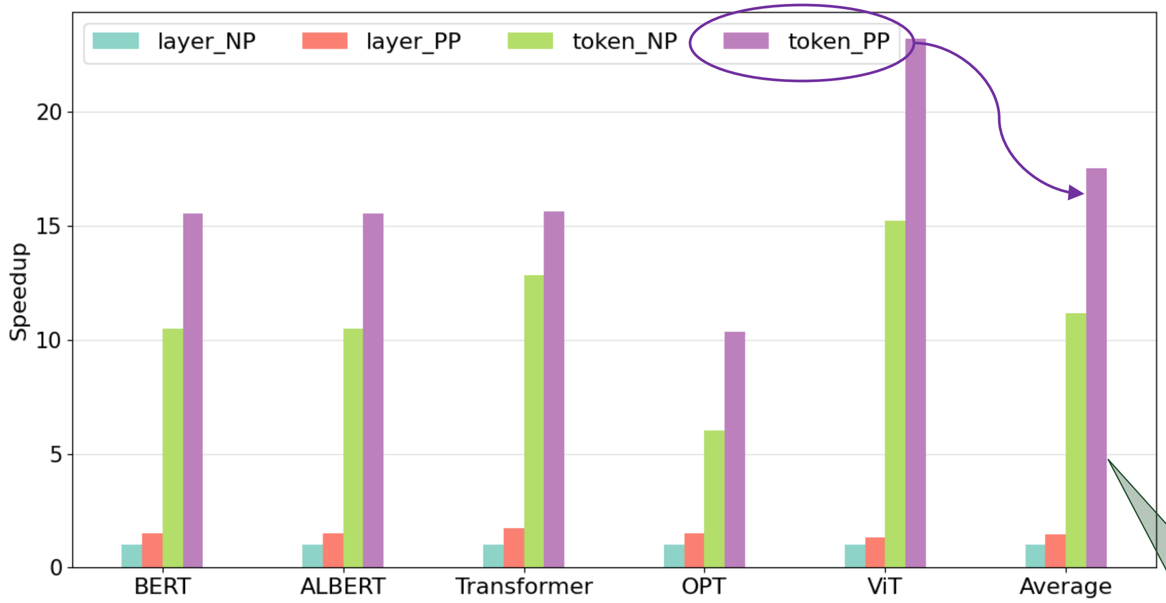
- To mitigate SC accuracy degradation issues:
  - ARTEMIS avoids stochastic additions
  - ARTEMIS utilized an optimized approach to stochastic multiplications
- We performed a detailed computational error analysis for each approximate block
- We performed a detailed accuracy analysis for the various models
- Minimal accuracy degradation, averaging at **1.4%** compared to FP32 and **0.5%** compared to quantized 8-bit models

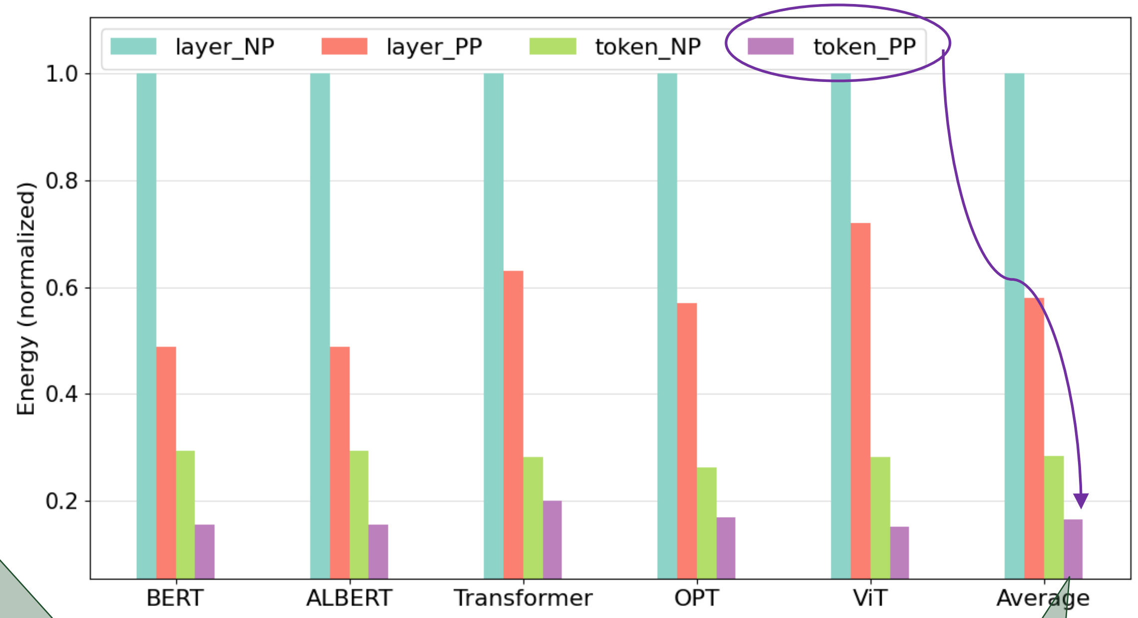
Block	MAE	Max Error	Calibration Accuracy
Stochastic MUL	0.039	0.123	4.68
Analog ACC	0.0085	0.0729	6.88
A_to_B	0.00037	0.00062	11.38
Softmax	0.0020	0.0078	8.20

Model	Dataset	FP32	Q(8-bit)	Q(8-bit) + SC
Transformer-base	70.90%	70.40%	69.45%	70.90%
BERT-base	87.00%	86.27%	85.92%	87.00%
Albert-base	86.07%	84.80%	84.51%	86.07%
ViT-base	97.60%	96.50%	96.20%	97.60%
OPT-350	18.07 (BLEU)	17.79 (BLEU)	17.49 (BLEU)	18.07 (BLEU)

Dataflow Sensitivity Analysis

layer – layer-based dataflow
token – token-based dataflow
NP – no pipelining
PP – pipelining





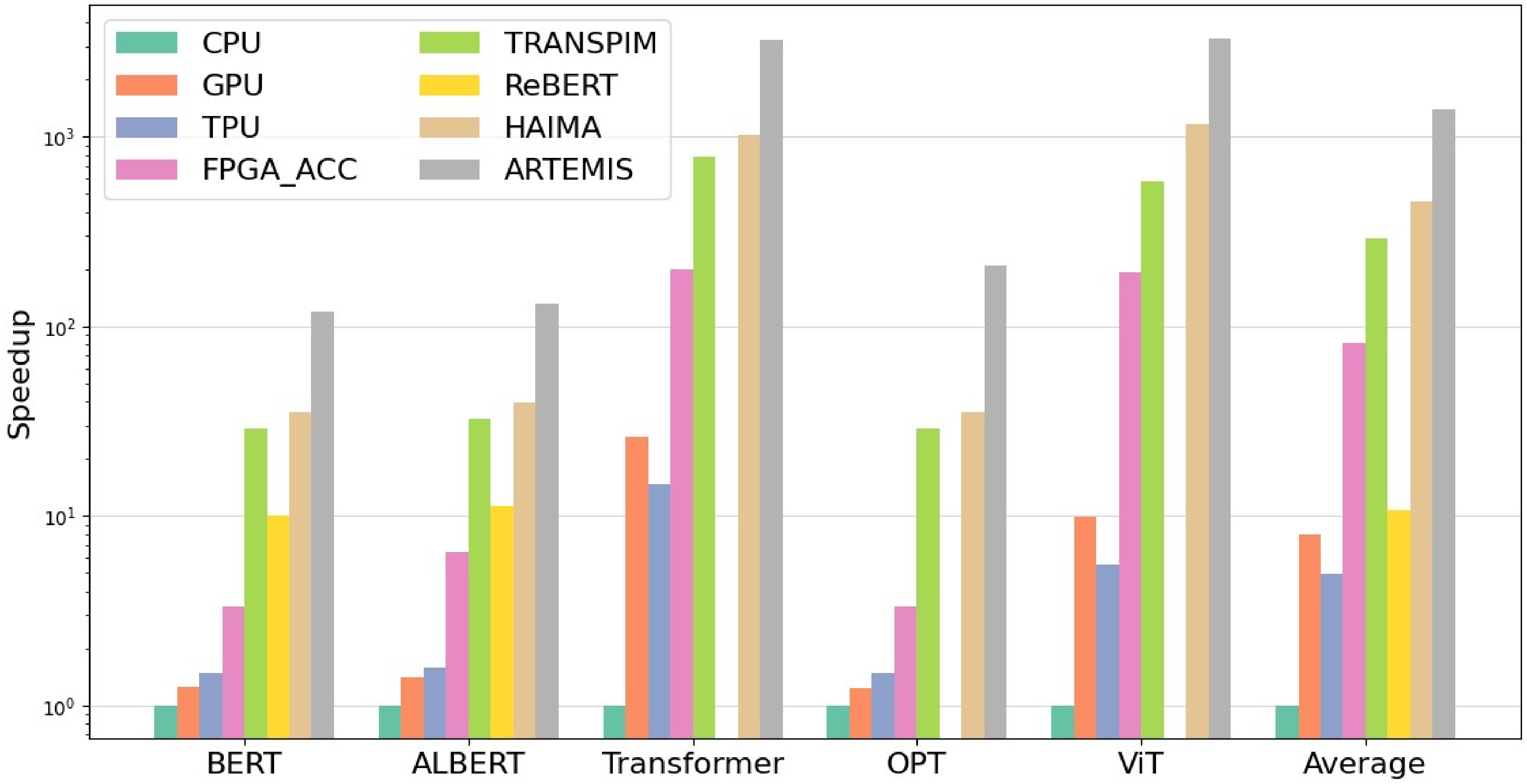
- Results normalized to baseline model (layer_NP)

Employing token-based dataflow and pipelining optimizations simultaneously results in the highest speedup and least energy values

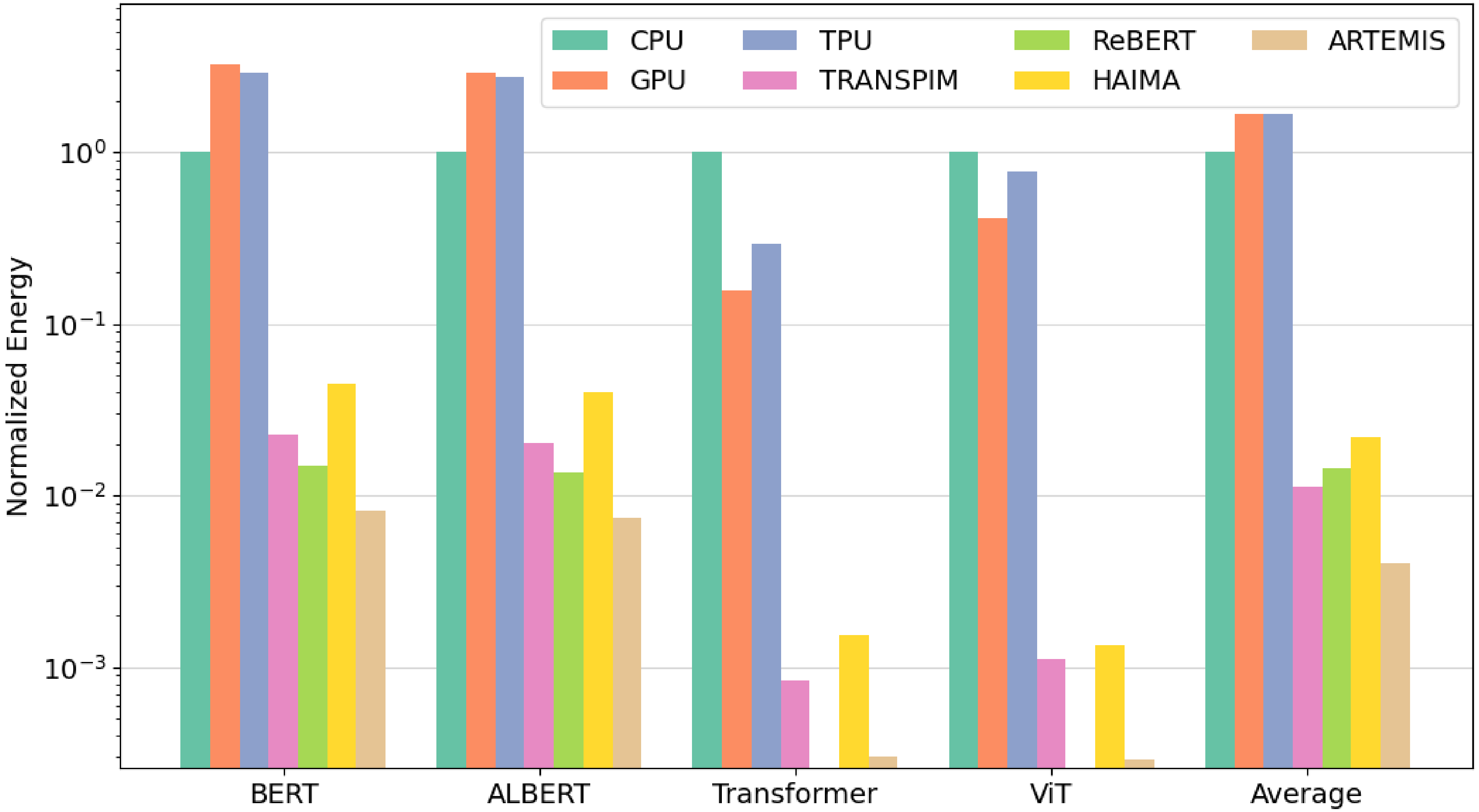
Token-based DF + PP improved speedup by **16.2×**

Token-based DF + PP reduced energy consumption by **3.5×**

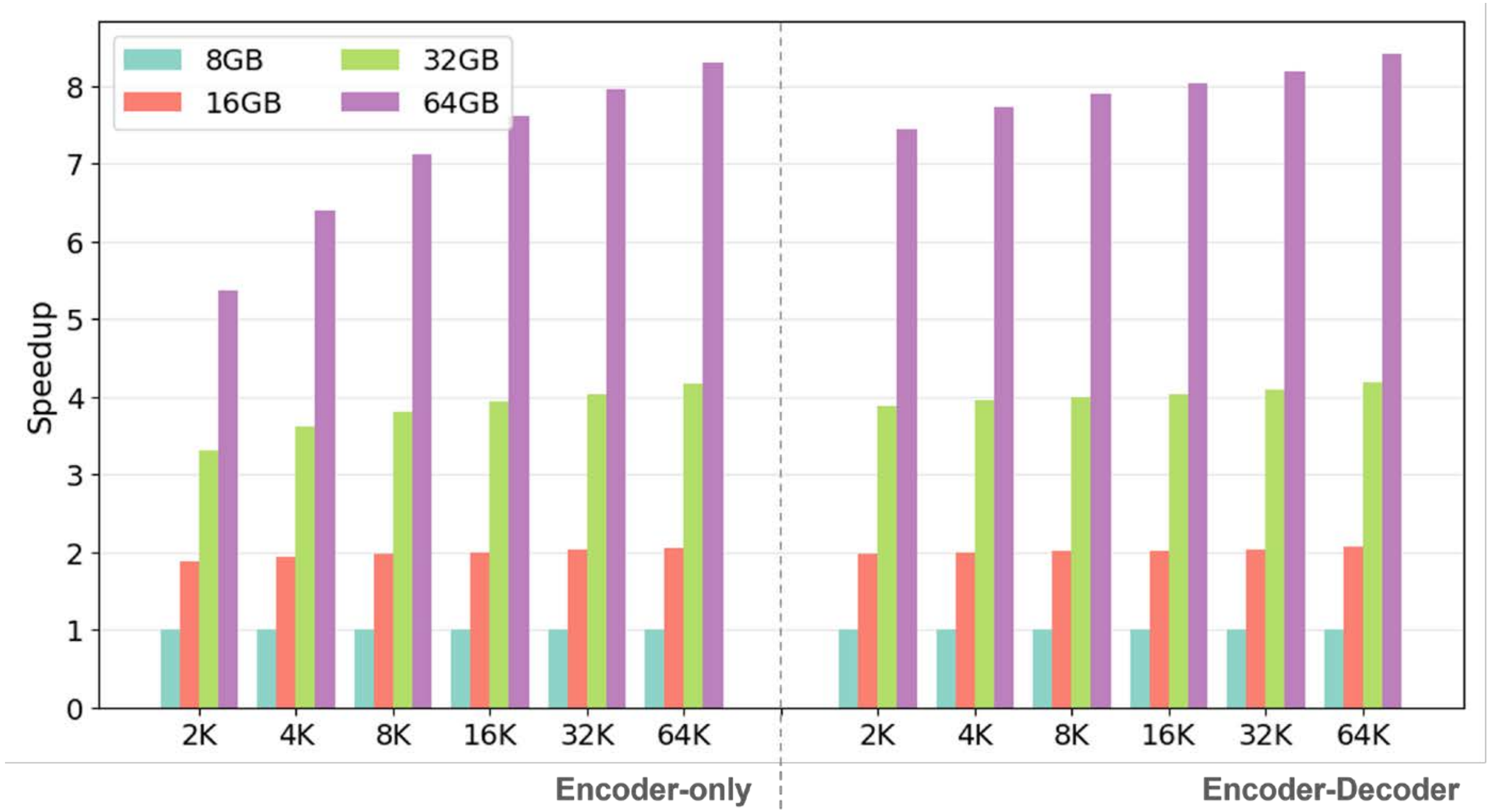
Results: Speedup



Results: Energy Efficiency



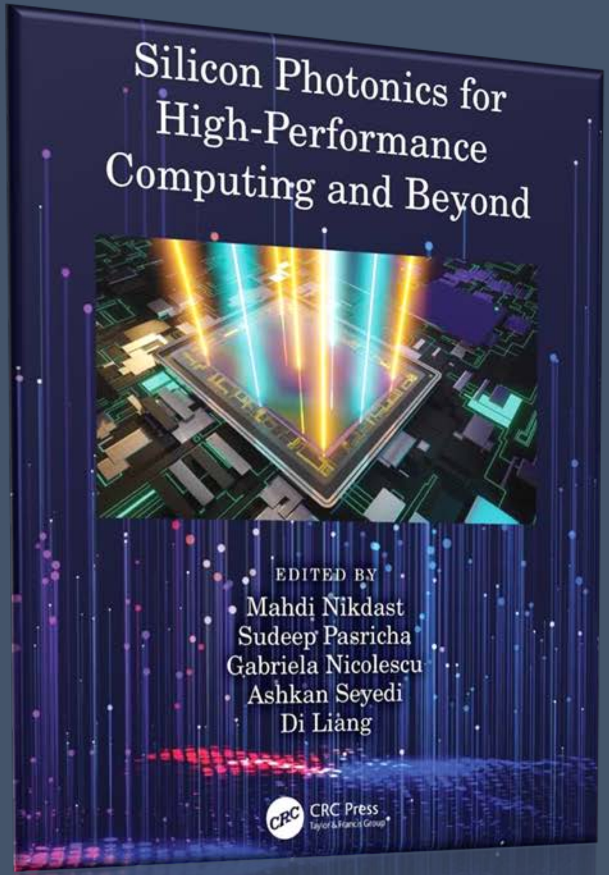
Results: Scalability Analysis

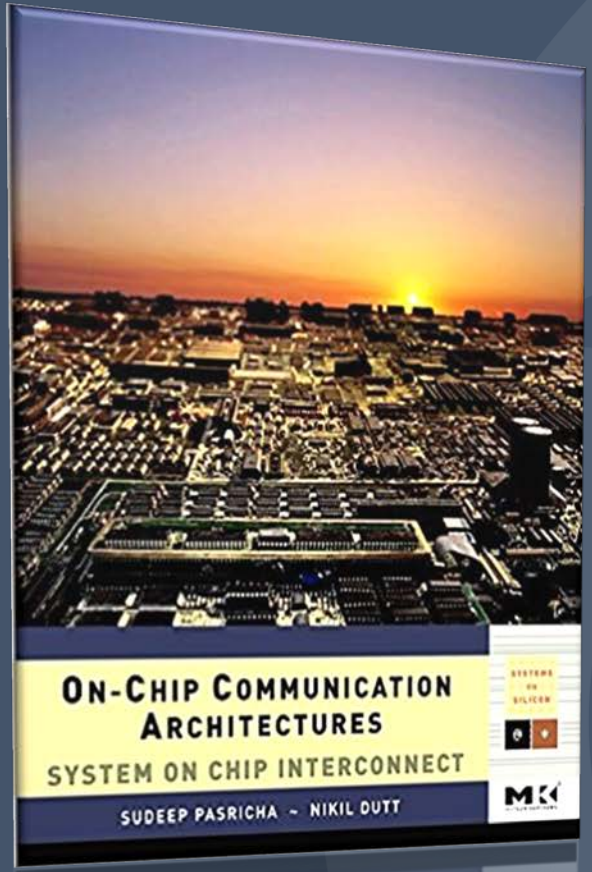


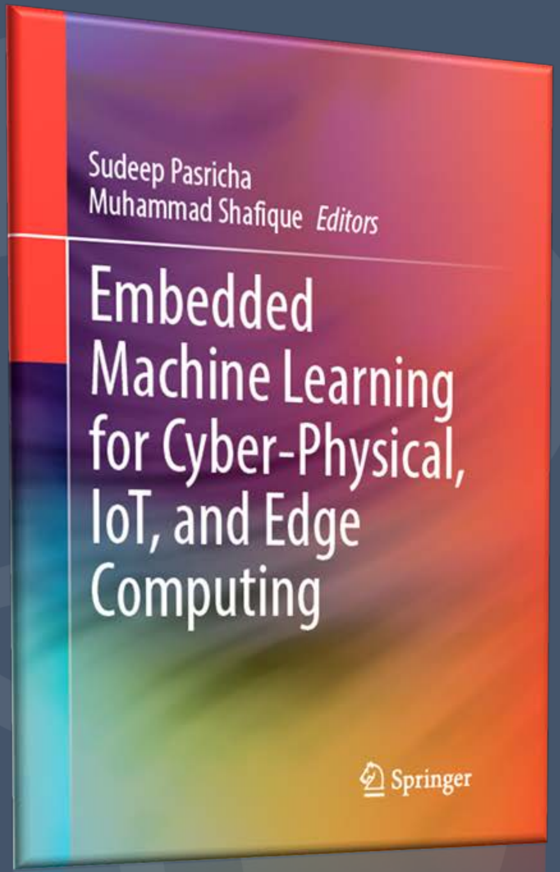
- Speedup obtained by employing additional HBM stacks for processing workloads of increasing input sequence lengths
  - ARTEMIS scales up well with increasing memory usage

Conclusions

- **ARTEMIS is the first in-DRAM hardware accelerator for transformer neural networks that combines stochastic and analog computing**
- **ARTEMIS can efficiently accelerate inference of Transformer neural networks with negligible accuracy degradation and overcome many transformer inference challenges**
- **Speedup improved by at least 3.6×**
- **Energy Efficiency improved by at least 1.8×**
- **ARTEMIS introduces a promising paradigm for energy-efficient LLM acceleration in edge devices**







Thank you

Sudeep Pasricha (sudeep@colostate.edu)

Colorado State University