In-memory computing for deep-learning acceleration

Evangelos Eleftheriou
CTO & Co-founder, Axelera AI

The AI revolution

A portrait photo of a kangaroo wearing an orange hoodie and blue sunglasses standing on the grass in front of the Sydney Opera House holding a sign on the chest that says Welcome Friends!

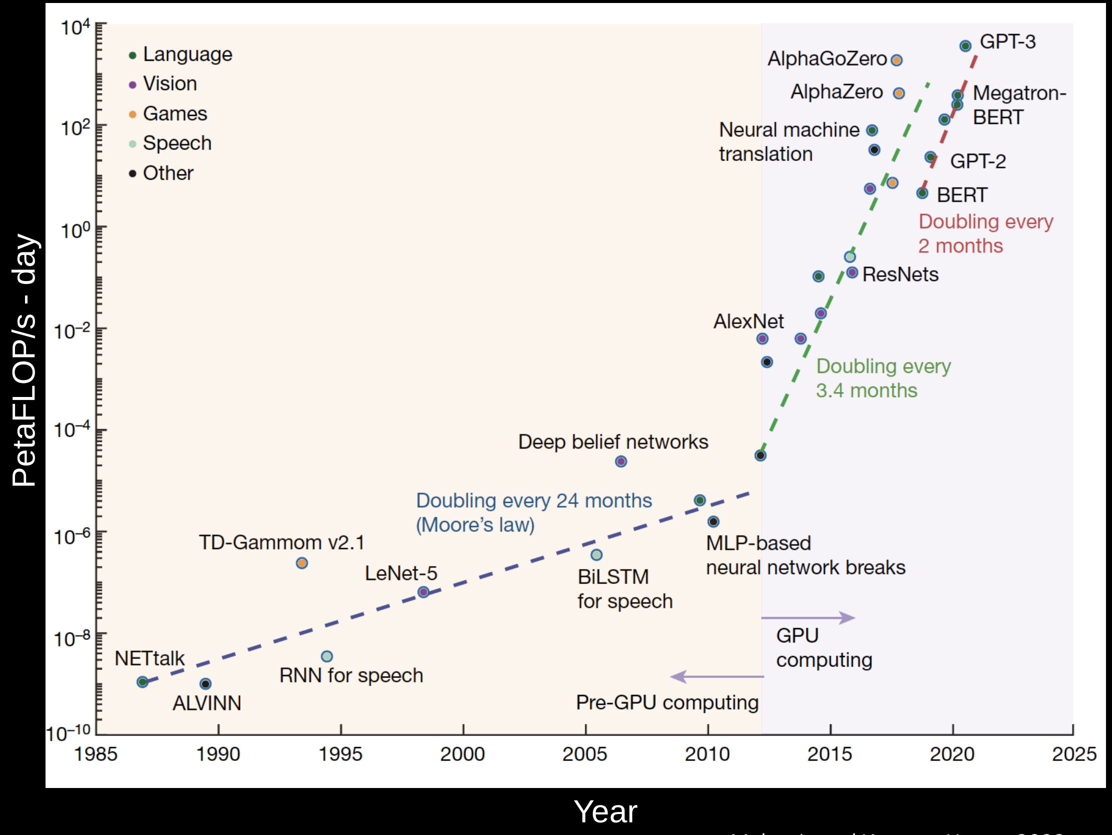
AI is revolutionizing automized execution of many cognitive tasks

- ML algorithms at times exhibit above-human accuracy for certain tasks
- ML algorithms can create realistic images from a text input

https://parti.research.google

Compute demands for AI

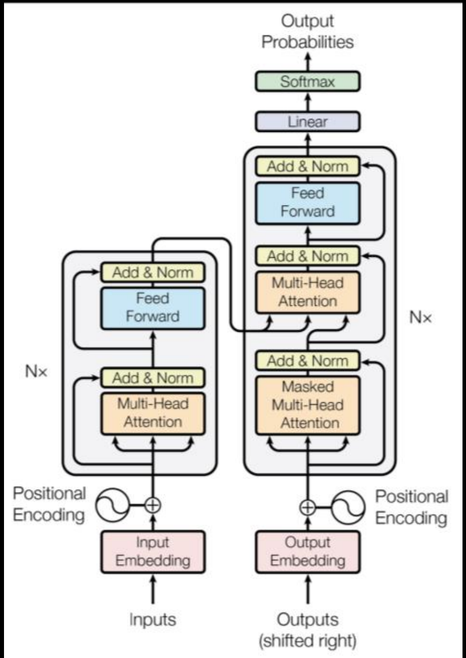
- Compute requirements for large AI training jobs are doubling every 2 months
- Unsustainable without significant hardware and software innovation



Mehonic and Kenyon, *Nature*, 2022

DL's computational efficiency problem

Transformer model

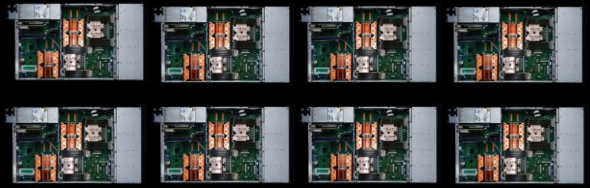


Vaswani et. al., *NIPS,* 2017

1 Transformer (big) training run, is ~1 weeks of home energy consumption

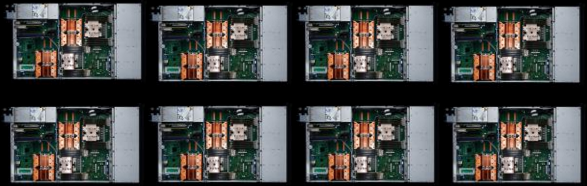
Transformer (base) 65M parameters

8 GPUs
0.5 days
~27 kWh
26 lbs CO_2



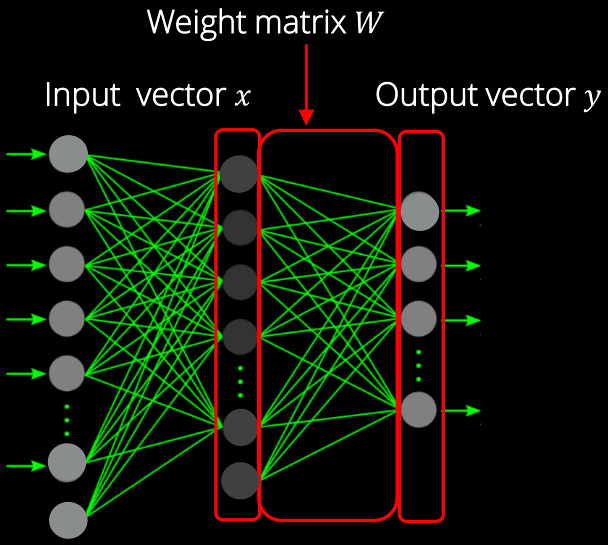
Transformer (big) 213M parameters

8 GPUs
3.5 days
~201 kWh
192 lbs CO_2



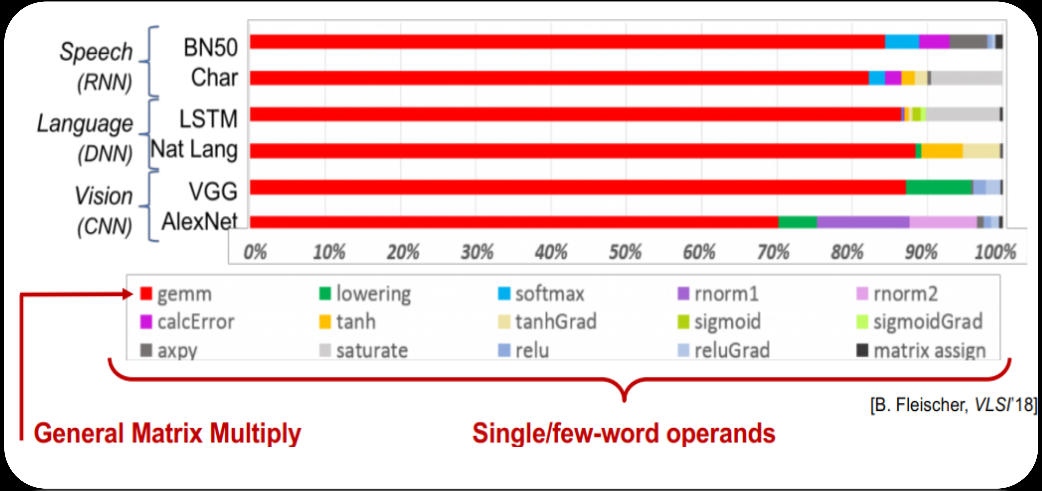
Strubell, Ganesh, McCallum, *arXiv*:1906.02243, 2019

Breakdown of arithmetic operations



$$y = W x$$

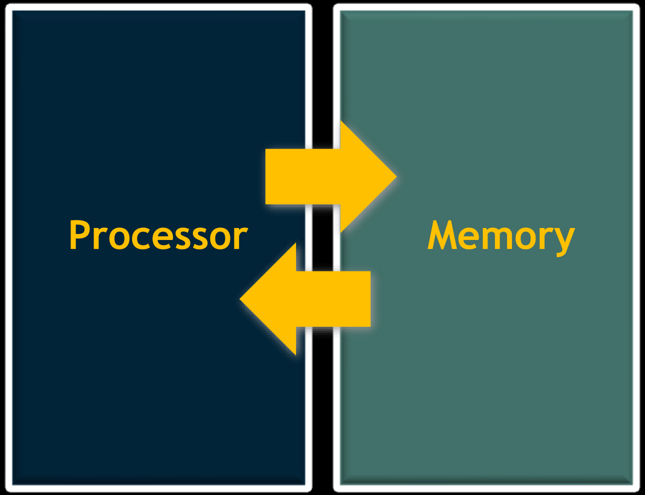
Matrix-vector multiplications constitute 70-90% of the total deep learning operations



Source: https://www.ibm.com/blogs/research/2018/06/approximate-computing-ai-acceleration/
Fleischer, Shulka , *IBM Research Blog*, 2018

Moving data dominates power consumption

Conventional von Neumann computing architecture



Cost of data transfer

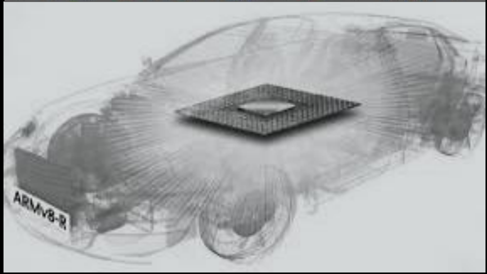
Operation	Energy (pJ)	Relative Energy Cost
8b Add	0.03	
16b Add	0.05	
32b Add	0.1	
16b FP Add	0.4	
32b FP Add	0.9	
8b Multiply	0.2	
32b Multiply	3.1	
16b FP Multiply	1.1	25X
32 FP Multiply	3.7	
32b SRAM Read (8KB)	5	
32b DRAM Read	640	

Dally, *ScaledML*, 2019
Horowitz, *ISSCC*, 2014

Efficiency matters even more at the Edge …

- AI for mobile devices, e.g., authentication, speech recognition, mixed/augmented reality
- Embedded processing for the Internet of Everything, e.g., smart cities and homes
- Embedded processing for prosthetics, wearables and personalized healthcare
- Real-time Video Analytics for Autonomous Navigation and control

… especially for energy and memory constrained embedded applications



Google Images;
S. Kulkarni et al, *MWSCAS*, 2017

AI Systems: Trends & Opportunities

Key trends

→ Energy to move data dominates compute energy

→ Neural network complexity increases exponentially

→ Neural networks are dominated by MVMs

Opportunities

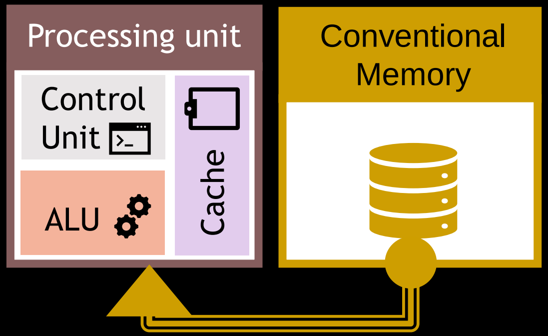
★ Minimize data movement by performing computation directly (or nearby) where the data resides

★ Introduce novel computational primitives that facilitate the DL workloads

In-Memory Computing (IMC) for DL

matrix-vector-multiplication (MVM) $\boldsymbol{A} \times \vec{x} = \vec{y}$

von Neumann-based

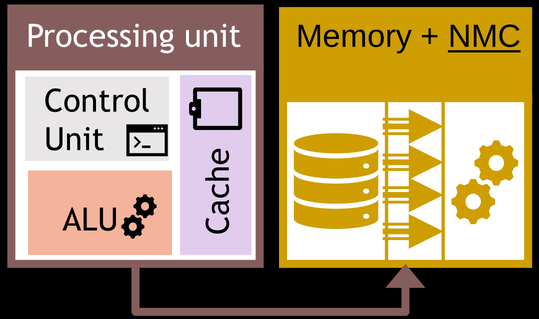


Send matrix $\boldsymbol{A}$

off-chip communication

element-wise processing $\Longrightarrow \mathcal{O}(\mathrm{N}^2)$

Near-memory computing (NMC)

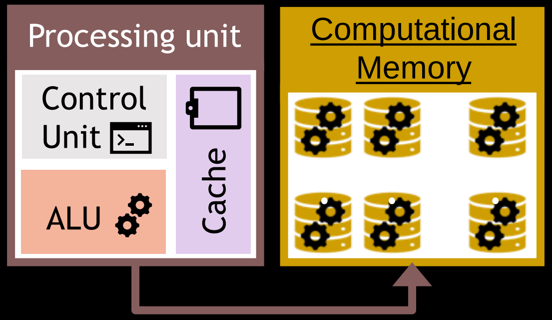


Multiply by vector $\vec{x}$

on-chip communication

vector-wise processing $\Longrightarrow \mathcal{O}(\mathrm{N})$

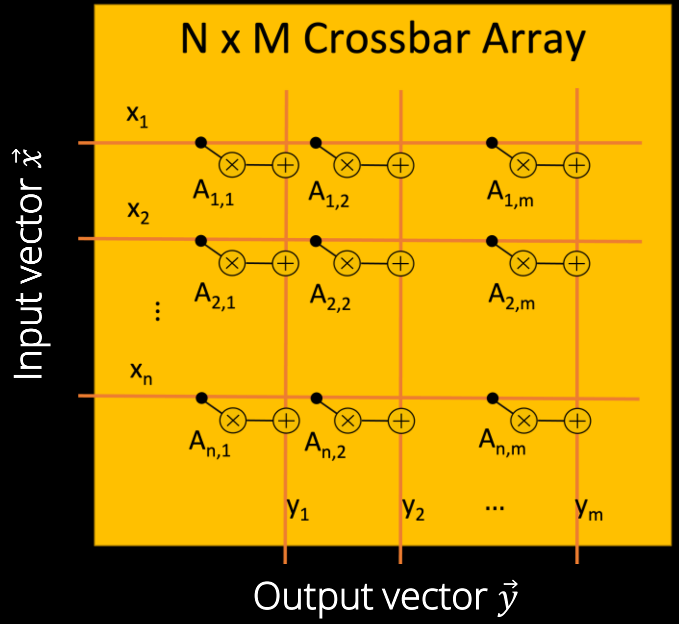
In-memory computing (IMC)



Multiply by vector $\vec{x}$

in-place parallel-processing $\Longrightarrow \mathcal{O}(1)$

In-Memory Computing (IMC) in a nutshell



In-memory Matrix-Vector Multiplication (MVM):

- The inputs $\vec{x}$ are applied at the rows
- The weights $A_{i,j}$ are stored in the memory
- The outputs $\vec{y}$ appear at the columns

In-place MVM operations with $O(1)$ time complexity

Burr, et al., *Adv. Phys. X,* 2017
Merrick-Bayat et al., *IEEE TNNLS*, 2017
Moons, *IEEE CICC,* 2018
Eleftheriou, et al., *IBM J. R&D,* 2019
Xia, Yang, *Nature Materials,* 2019
Sebastian, et. al.,, *Nature Nano,* 2020
Papistas *et al*., *IEEE CICC*, 2021

IMC memory technology trade-offs

Considerations for choosing the right memory

- Performance: TOPS & TOPS/W
- Density: die area, which affects cost
- Volatility, write time/energy & endurance: static weights or reloadable weights
- Stability (temperature, drift, noise): Accuracy; suitabilty for Edge applications
- Manufacturing process, compatibility: Supplier risk & cost
  Does it scale to lower technology nodes?

Comparison of best performances of commercial stand-alone memories in 2021

	SRAM*	DRAM	STT-RAM	PCM	ReRAM	NOR Flash
Cell Size (F^2)	~100	6-8	6-30	4/4L	6-30	6-30
Multibit	1	1	1	≥1	≥1	≥1
Endurance (cycles)	$\sim10^{16}$	$\sim10^{15}$	$\sim10^{15}$	$\sim10^{7}$	$\sim10^{6}$	$\sim10^{5}$
Read Time (ns)	~1	~10	~10	10-100	~100	10-100
Write Time (ns)	~1	~10	~10	10-100	~100	~1000
Write Energy (Energy/bit)	~1fJ	~10fJ	~100fJ	~10pJ	~100fJ	~100pJ

Lanza et. al., *Science* 2022
F: represents feature size, L: denotes number of layers
*Embedded

System design trade-offs

Energy efficiency vs. Accuracy

- Low effective precision of weights/activations increases efficiency but decreases accuracy
- Analog architectures require high resolution DACs/ADCs for high accuracy impacting energy efficiency

Endurance & noise effects vs. training

- Memory cycling endurance determines suitability for training and/or inference applications
- Noise and nonlinear effects affect precision of MVM, thus dictating complex "HW Aware Training" schemes

Compute density vs. re-programmability

- The smallest cell-size memory technologies exhibit high write-latency precluding re-programmability
- With fast re-programmability, there is no need to map entire DNNs onto multiple crossbar arrays, which affects compute density

Scalability

- Mature technologies can scale better with technology node
- Compatibility with CMOS crucial for successful commercialization of the IMC technology

Using SRAM as example

	SRAM	DRAM	STT-RAM	PCM	ReRAM	NOR Flash
Cell Size (F^2)	~100	6-8	6-30	4/4L	6-30	6-30
Multibit	1	1	1	≥1	≥1	≥1
Endurance (cycles)	~10^{16}	~10^{15}	~10^{15}	~10^{7}	~10^{6}	~10^{5}
Read Time (ns)	~1	~10	~10	10-100	~100	10-100
Write Time (ns)	~1	~10	~10	10-100	~100	~1000
Write Energy (Energy/bit)	~1fJ	~10fJ	~100fJ	~10pJ	~100fJ	~100pJ

+

- Fastest read time → highest **performance**
- Fastest write time → re-programmability
- Highest endurance → longevity
- Low noise, no drift → better accuracy
- Standard manufacturing process → scalability

–

- Largest cell size → low density
- Idle and retention power → high power consumption

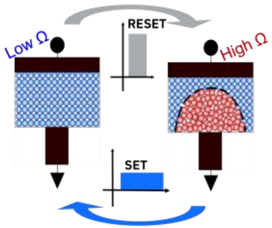
Phase-Change Memory (PCM)

Principle: Two distinct solid phases of a Ge-Sb-Te metal alloy to store a bit

- Transition between phases by controlled heating and cooling
- Intermediate phases to obtain a continuum of different states or resistance levels
- Well understood device physics and successfully commercialized technology

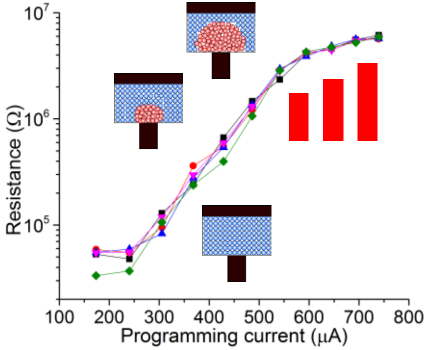
Key enablers:

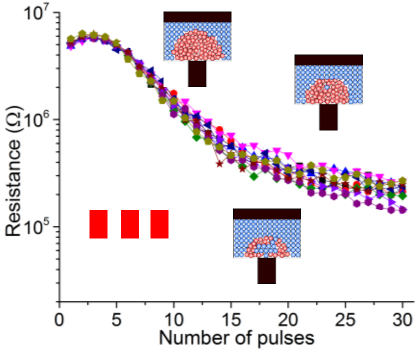
- *Multilevel memory capability*: Analog storage device; but with drift and noise
- *Accumulative behavior*: Nonvolatile nanoscale integrator; but stochastic and nonlinear



Coductance Range:
~50KΩ – 50MΩ

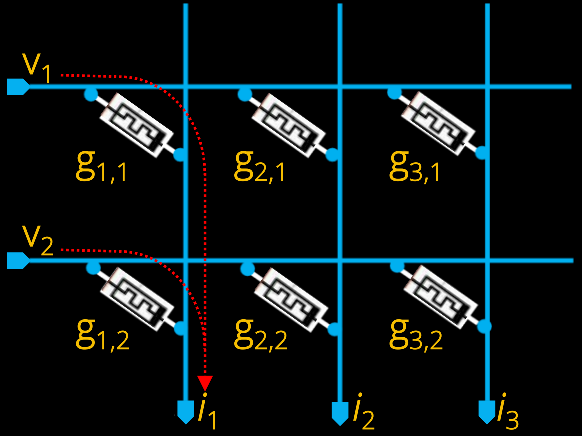
Read speed: ~100ns,
Write speed: ~100ns





Sebastian, Le Gallo, Eleftheriou , *J. Phys. D: Appl. Phys.*, 2019

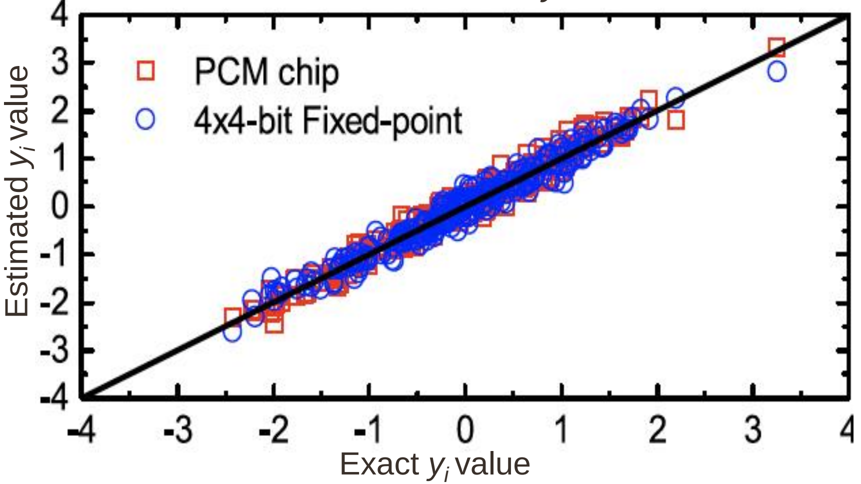
MVM using PCM technology



– Matrix elements → conductances $g_{m,n}$
– Input vector → read-voltage pulse v_m
– Currents i_n → result vector

Precision equivalent to 4-bit fixed point arithmetic

$$\boldsymbol{A} \times \vec{x} = \vec{y}$$



- A is a 256X256 Gaussian matrix coded in a PCM chip
- $\vec{x}$ is a 256-long Gaussian vector applied as voltage

Measurements using Fusion IBM's 1st gen analog AI chip, 1M PCM devices, 90nm CMOS

Le Gallo, et. al., *IEEE Trans. on Electron Devices,* 2018

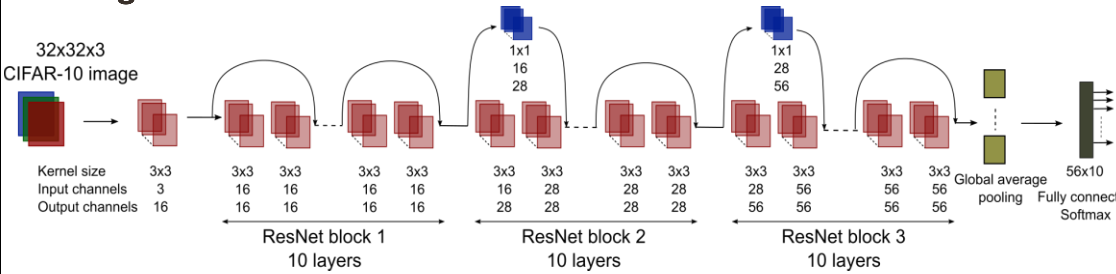
Inference on PCM-based IMC

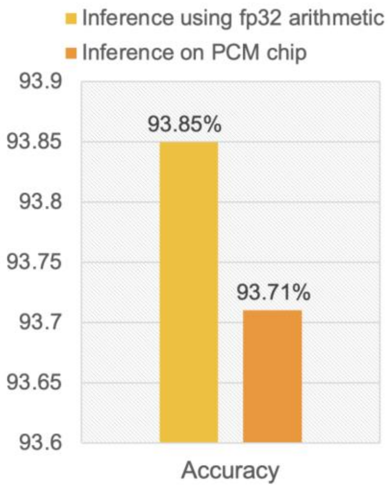
”Hardware-aware training”

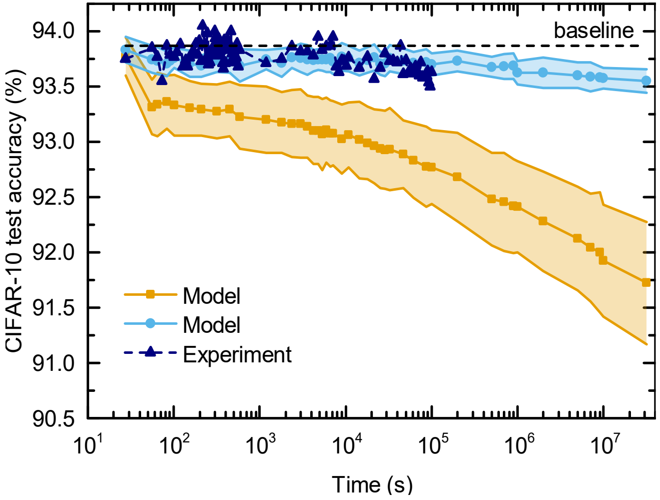
- Custom training approach needed to account for the conductance distributions
- Incorporation of “injective” noise and drift compensation techniques during training

“Almost” SW equivalent accuracies can be achieved over a long time

Image classification: ResNet-32 trained on CIFAR-10







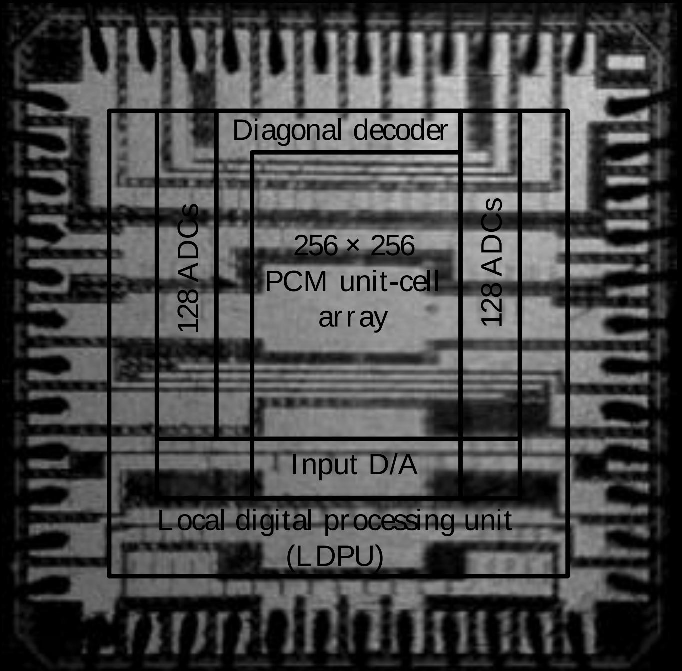
V. Joshi, et al., *Nature Communications*, 2020

PCM-based IMC core

Hermes: IBM's 2nd generation analog AI chip

- 256 x 256 PCM unit-cell array
- 4 PCM devices per unit cell
- Local digital processing unit
- 14 nm CMOS technology
- INT8 arithmetic

Unit-cell	8T4R
Input/weight/output bits	8b/Analog/8b
Throughput (TOPS)	1.008
Energy efficiency (TOPS/W)	10.5
Area efficiency (TOPS/mm^2)	1.59

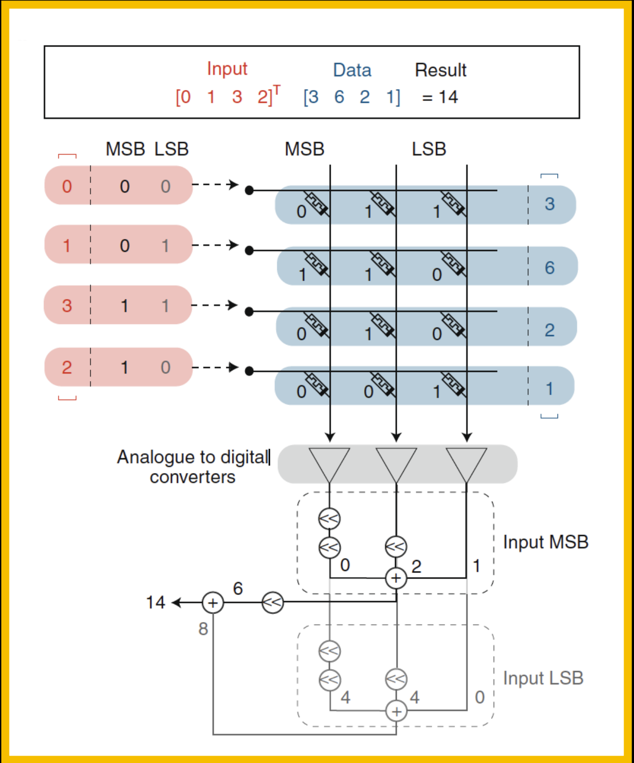


Khaddam-Aljameh et. al., *VLSI Technology Symposium*, 2021

"Bit-slicing" for high precision

Principle:

- Construct an MVM crossbar array from sub-arrays representing smaller bit widths
- Each sub-array processes one bit field or '*slice*' of an operand
  - Map an *n*-bit element of a weight matrix ➔ onto n binary memory cells – *n bit-slices*
  - Map an *m*-bit element of an input activation ➔ onto *m bit-slices*
  - Multiply in-place ***activation "bit-slices"*** with ***matrix weight "bit-slices"***
  - Combine partial products via shift-and-add reduction networks

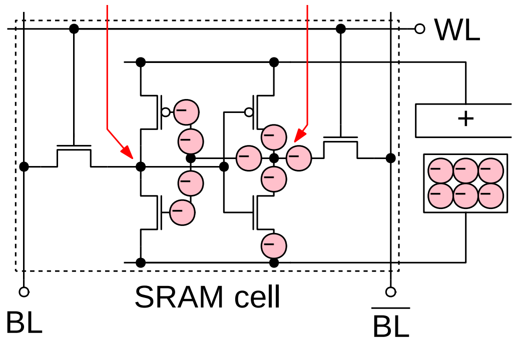


Tradeoff between precision and compute density

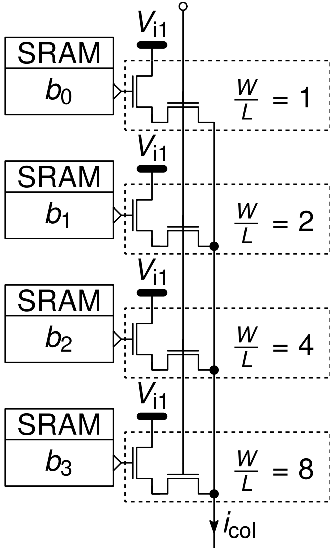
Sebastian, Le Gallo, Khaddam-Aljameh, Eleftheriou, *Nature Nano.*, 2020
Le Gallo et. al., *Neuromorphic Comput. Eng.*, 2022

Analog SRAM-based IMC

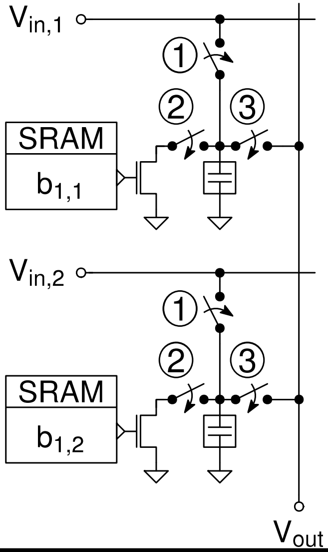
6T SRAM cell



Current-based
(SRAM-controlled current sources)



Charge-based
(SRAM + switched capacitors)



- volatile (persistent) binary storage element
- read/write speed: ~1ns @ 14nm node

✗ prone to device mismatch
✗ prone to voltage drop (IR)

✓ low metal cap. mismatch
✓ no significant voltage drop

SRAM & switched-cap approach

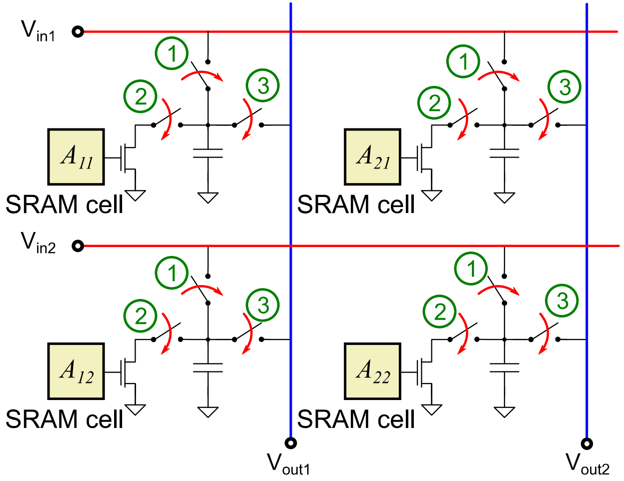
$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

MAP to SRAM content

MAP to cap voltage

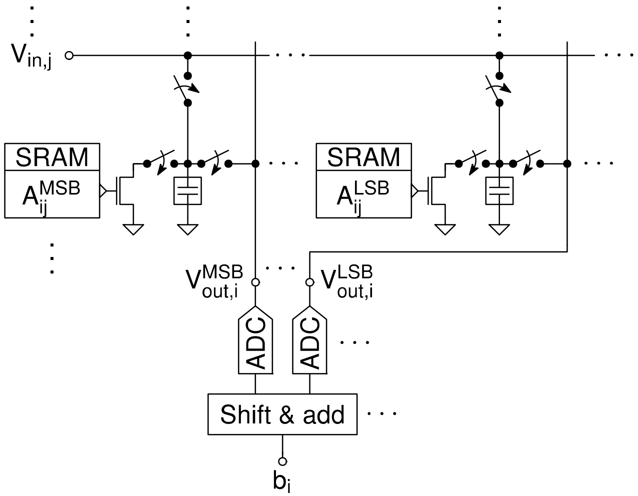
DECIPHER from voltage along the BL

"Charge sharing principle"



Biswas et al., *ISSCC,* 2018
Valavi et al., *JSSC,* 2019
Khaddam-Aljameh et.al., IEEE *TVLSI,* 2020

… with bit-slicing



SRAM cells used to store the elements of a binary matrix

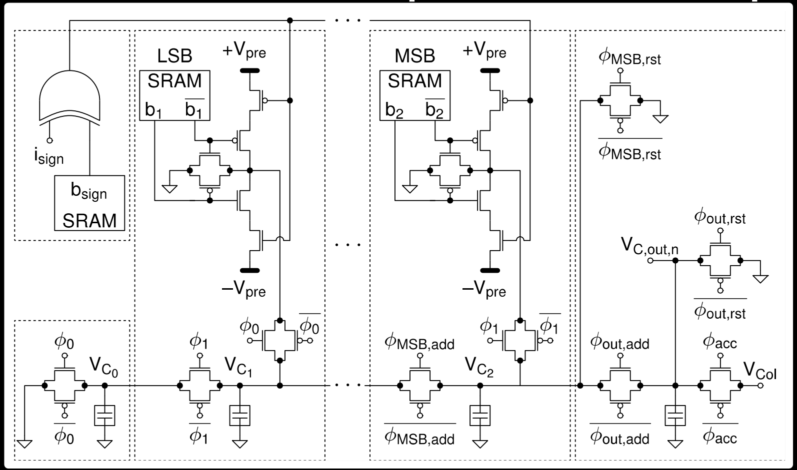
- Step 1: Capacitors charged to input values
- Step 2: Capacitors associated with value 0 are discharged
- Step 3: Capacitors shorted along the columns

For multi-bit weights:

- Step 4: A/D conversion
- Step 5: Bit-shift/add results
- Step 6: Summing up

An alternative SRAM scheme

Interleaved switched-capacitor-based multiplier

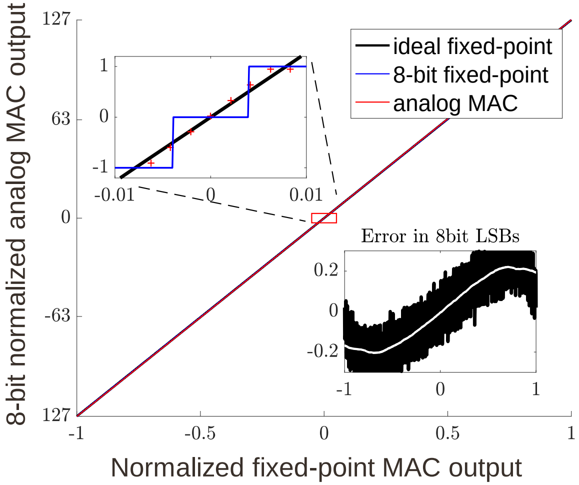


Principle:

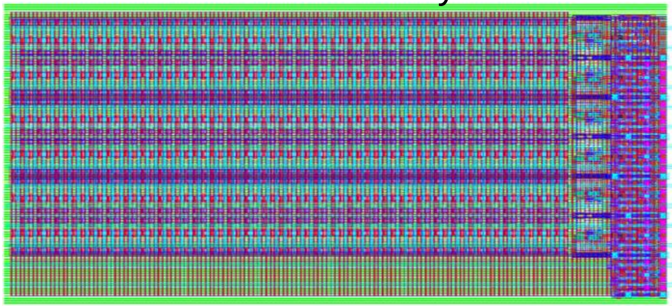
- Pipeline DAC: Generates weight proportional voltage V_w
- Switched-Cap DAC: Multiplies V_w with the input bits

In-memory MVM with precision that scales linearly in Area, Time, and Power

INT8 weight/activations, 512x512 MVM
14nm transistor-level *Spectre* simulation

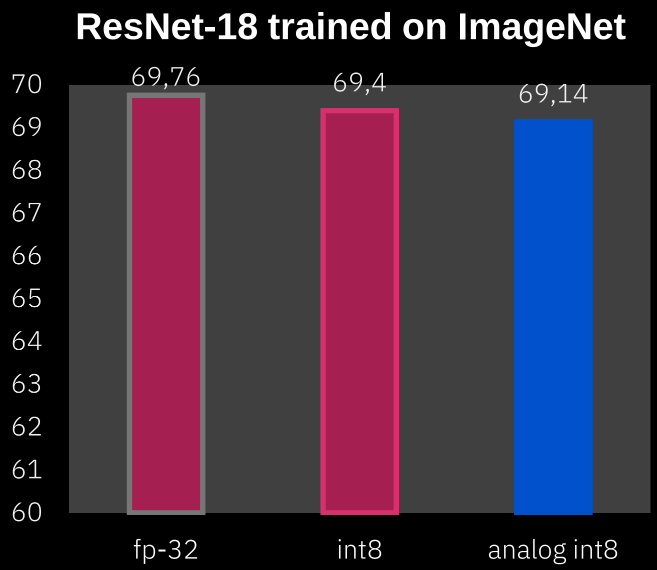


14 nm CMOS layout

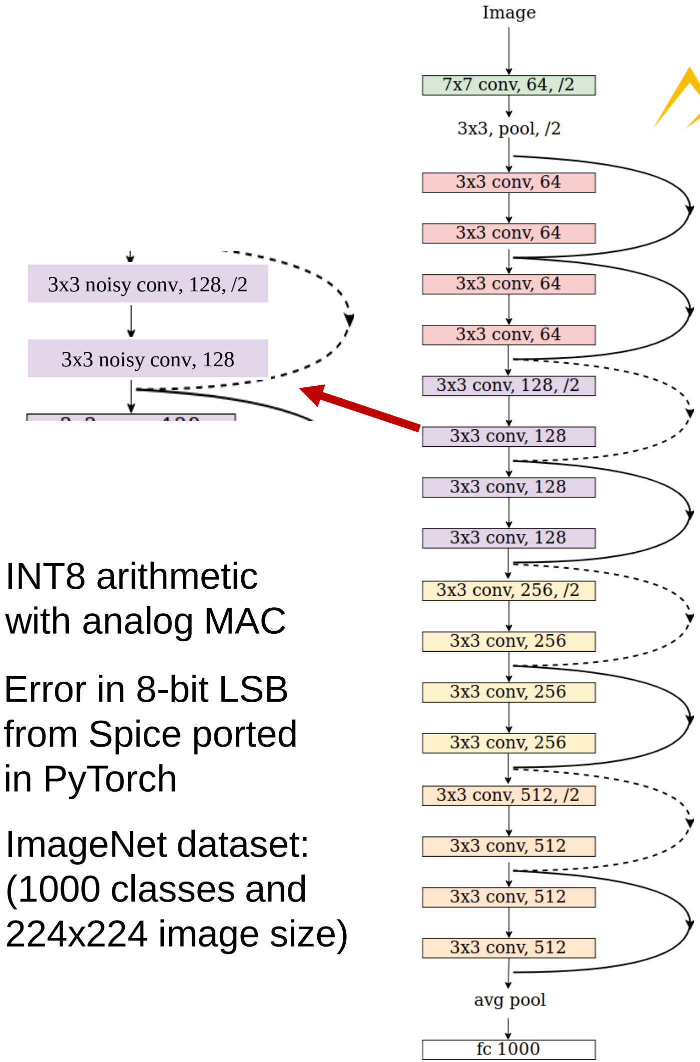


Khaddam-Aljameh et. al., IEEE *TVLSI*, 2020

Inference on interleaved switched-cap-based IMC



- Int8 model with “noisy convolutions” achieves 0.26% lower accuracy compared to ideal noiseless model
- No retraining or recalibration was applied to the model after post-training quantization



INT8 arithmetic with analog MAC

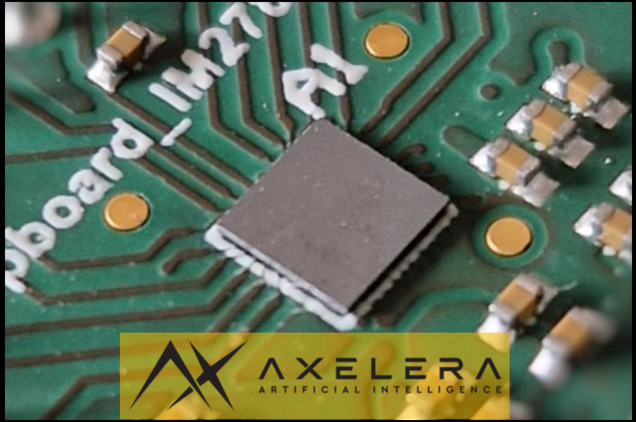
Error in 8-bit LSB from Spice ported in PyTorch

ImageNet dataset: (1000 classes and 224x224 image size)

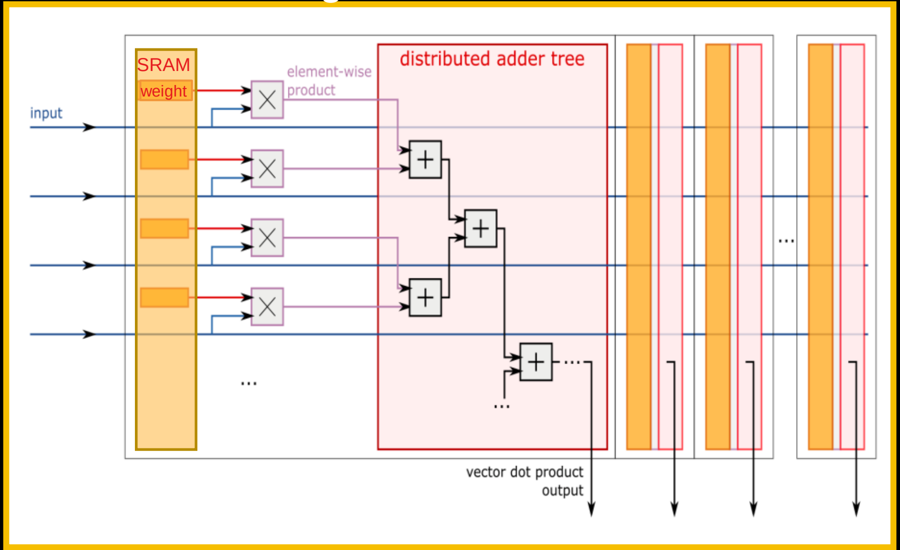
Digital SRAM-based IMC

Thetis Core: Axelera's 1st generation digital IMC chip

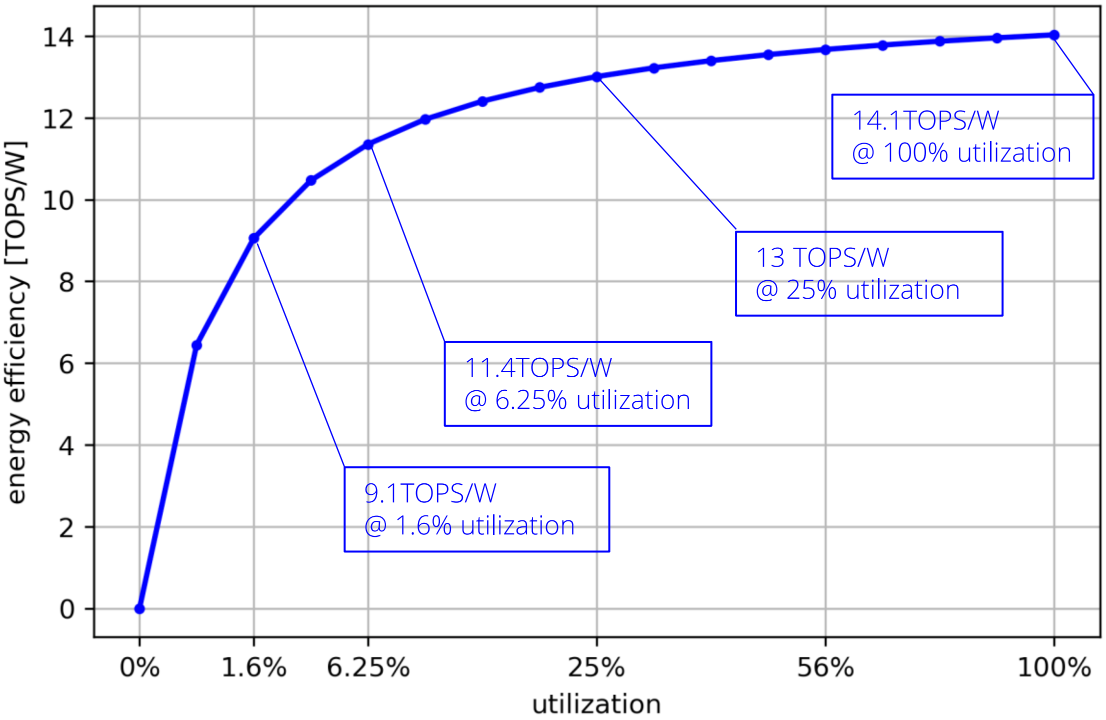
- Area: 8.6 mm^2
- Throughput: 39.3 TOPs
- Energy efficiency: 14.11 TOPs/W
- Energy efficiency (normalized 1bIN-1bW): 903 TOPS/W
- INT8 arithmetic



High-level architecture



Thetis core: energy efficiency vs. utilization



By reducing utilization from 100% to 25%, the energy efficiency drops by only 7%

For all practical use cases the energy efficiency remains "almost" constant

Inference on digital SRAM-based IMC

No need for costly ”quantization aware” or “HW aware” training

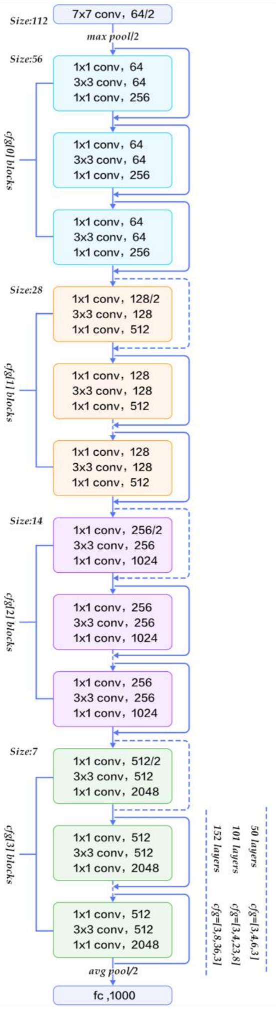
- Calibrate pre-trained model using small subset of training data
- Use statistics to compute clipping ranges and scaling factors

Image classification accuracy on ImageNet

Network	FP-32 accuracy	Axelera-AI Int-8 accuracy
ResNet-18	69.76	69.57 (-0.19)
ResNet-34	73.31	73.21 (-0.10)
ResNet-50	76.13	76.03 (-0.10)

A “*calibrated model*” running on digital SRAM-based IMC with INT8 arithmetic delivers FP32 iso-accuracy

ResNet-50



The state-of-the-art in IMC

Device	PCM	PCM	RRAM	MRAM	A-SRAM	A-SRAM	Digital CMOS	D-SRAM
CMOS technology	14nm	40nm	22nm	22nm	16nm	28nm	16nm	12nm
Input/weight/output precision	8b/analog/8b	8b/8b/19b	8b/8b/14b	1b/1b/4b	4b/4b/8b	8b/8b/22b	8b/8b/8b	8b/8b/32
Energy efficiency (TOPS/W)	10.5	20.5	15.6	5.1	121	27.75	0.96	14.11
Energy efficiency (TOPS/W) (normalized: 1bIN-1bW)	336	1312	998.4	5.1	1936	1776	61.44	903
Area efficiency (TOPS/mm^2)	1.59	0.026	0.005	0.758	2.67	0.1	1.29	6.64

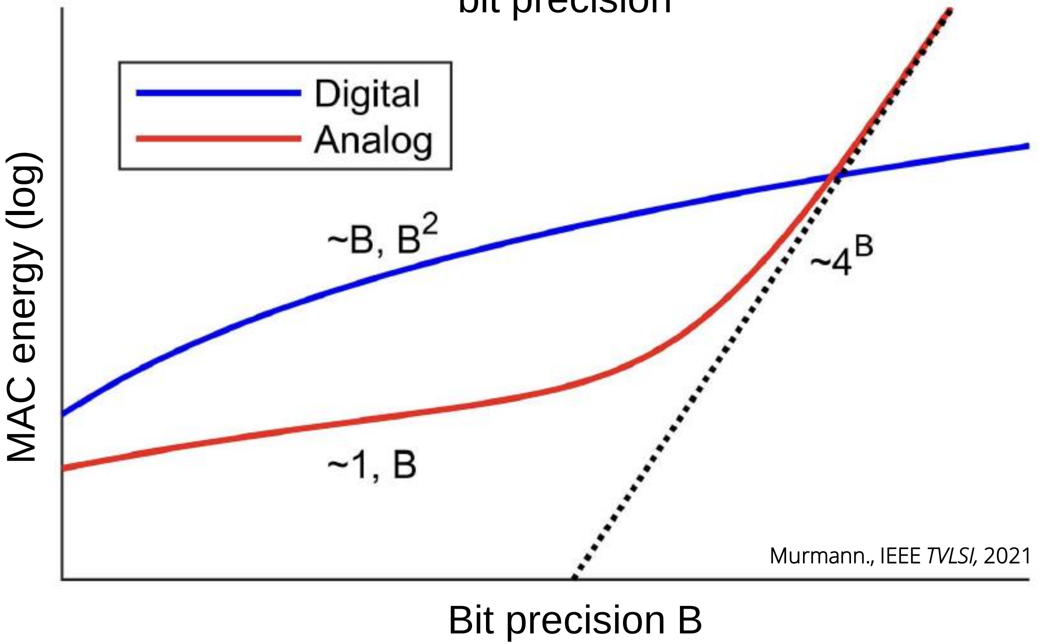
A-SRAM: Analog SRAM-based IMC
D-SRAM: Digital SRAM-based IMC

Lanza et. al., *Science*, 2022

Khaddam-Aljameh et al. *VLSI* 2021
Khwa et al. *ISSCC* 2021
Hung et al. *Nat. Electron.* 2021
Deaville et al. *ESSCIRC* 2021
Jia et al. *ISSCC* 2021
Wu et al. *ISSCC* 2022
Zimmer et al. *JSSC* 2020
Axelera AI May 2022

Analog or digital MAC?

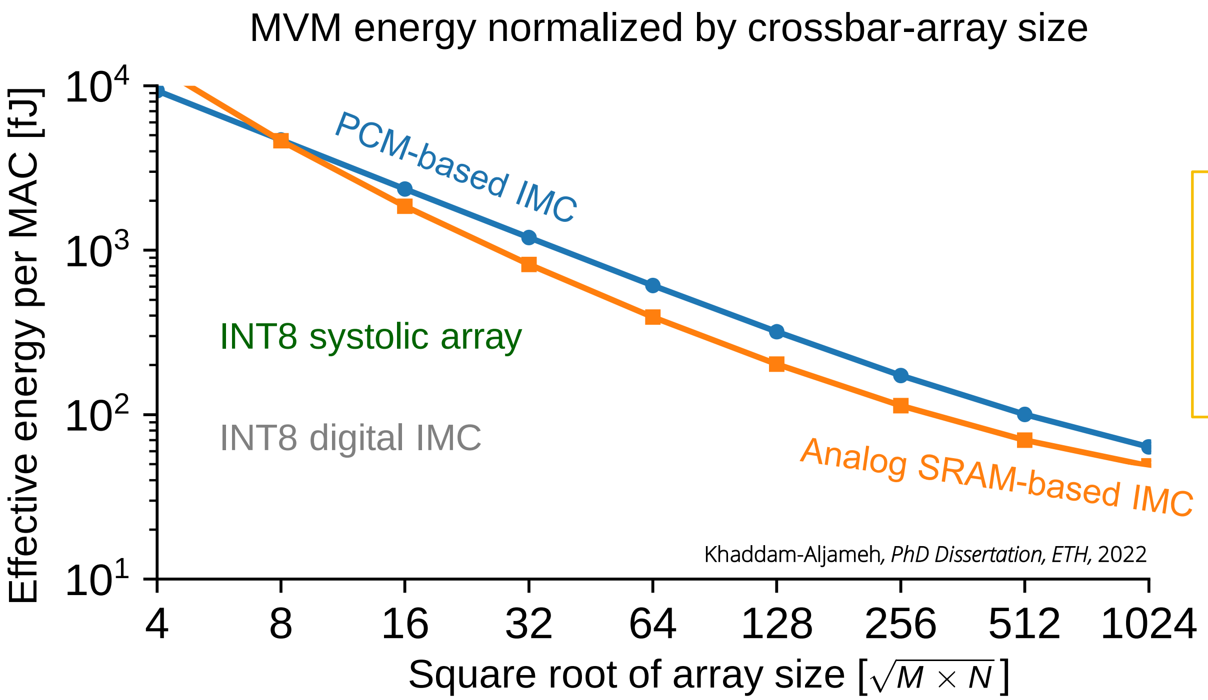
Qualitative plot of energy cost for MAC operations vs. bit precision



- Analog computations are more efficient than digital for low bit precisions
- Analog energy cost rises steeply for high bit precisions

Below 8-bit precision, analog realizations can be superior to digital ones

Analog or digital IMC?

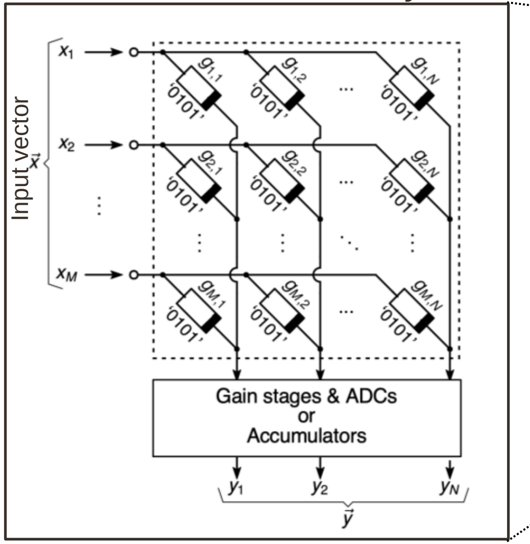


14 nm CMOS technology
- PCM-based MVM: INT8/analog/INT8
- Analog SRAM-based MVM: INT8/INT8/INT8

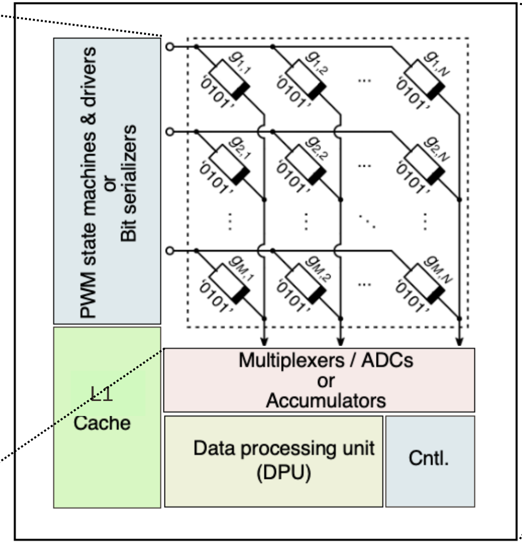
For practical crossbar-array sizes and INT8 weight/activations, digital IMC can be more energy efficient than analog IMC

IMC co-processor architecture

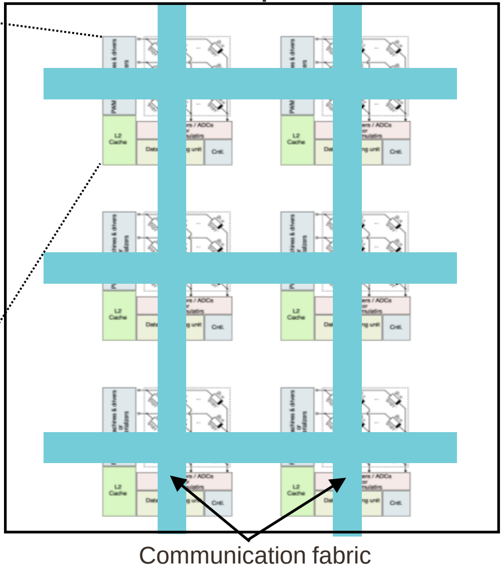
IMC crossbar array



IMC core



IMC co-processor



- Crossbar arrays with analog or digital memory cells
- "Bit-slicing" techniques to alleviate precision issues

- IMC array for matrix vector multiplications (MVM)
- DPU for element-wise vector operations, vector reduction functions, and activations

- 2-D mesh topology for systems with a large number of cores
- Fully-connected topology for systems with a small-number of cores

Concluding remarks

- The specific requirements that memory devices need to fulfill when used for IMC depend highly on the application
- Further improvements needed to make memristive IMC competitive against custom digital accelerators and SRAM-based IMC
  - Compute densities in excess of 7 TPOS/mm^2
  - Compute precision of at least 5- to 6-bit fixed-point arithmetic
- Analog IMC appears to require sophisticated HW-aware training to achieve FP32 iso-accuracies
- Digital IMC with INT8 arithmetic offers high throughput, high energy efficiency, high compute density and FP32 iso-accuracy without retraining