Deep Neural Network Inference with Analog In-Memory Computing

Manuel Le Gallo
Staff Research Scientist, IBM Research Europe

MS2F - Emerging Computing Technologies
for Next-Generation High-Performance
Computing, PASC25

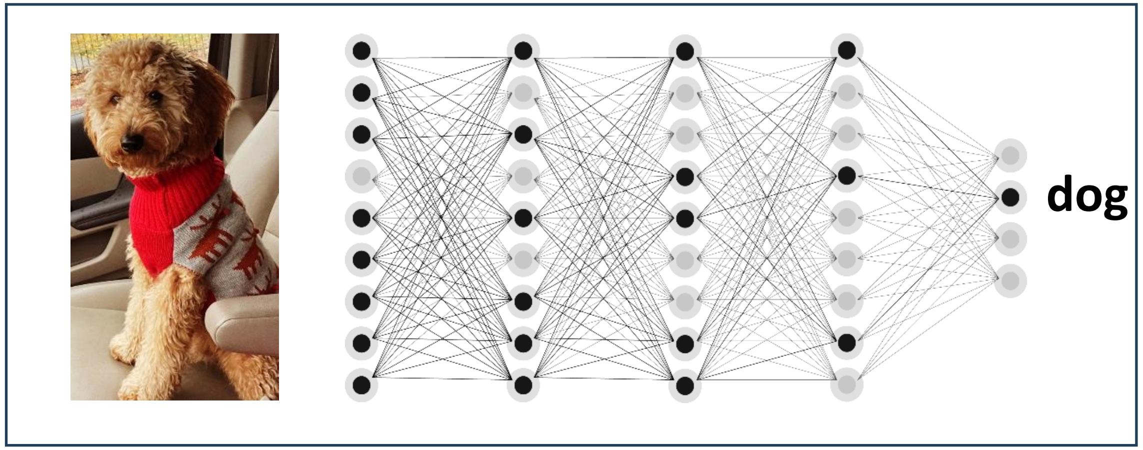
June 16, 2025

Outline

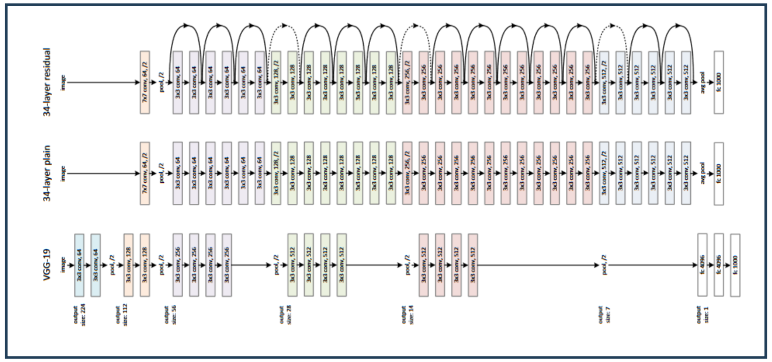
- Introduction to deep learning with in-memory computing
- IBM HERMES Project Chip
- Software/applications advancements towards next-gen AIMC accelerators
- 2 promising architecture advancements towards next-gen AIMC accelerators
- Conclusion

A revolution fueled by deep neural networks

Artificial deep neural networks

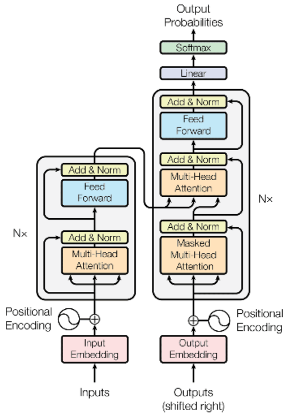


Convolutional neural



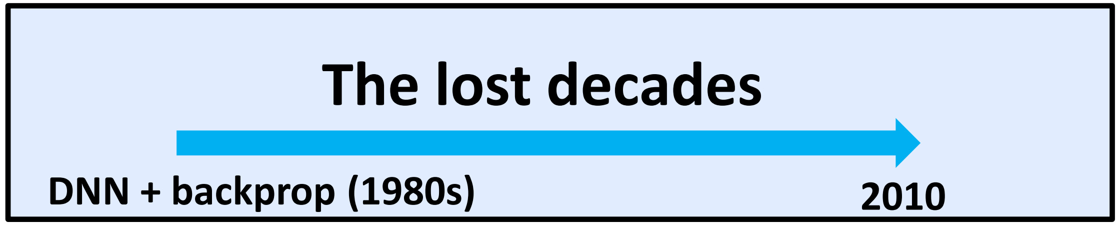
He et al., "Deep Residual Learning for Image Recognition", CVPR (2016)

Transformers



Vaswani et al., "Attention is all you need", NeurIPS (2017)

Hardware challenges for DNNs

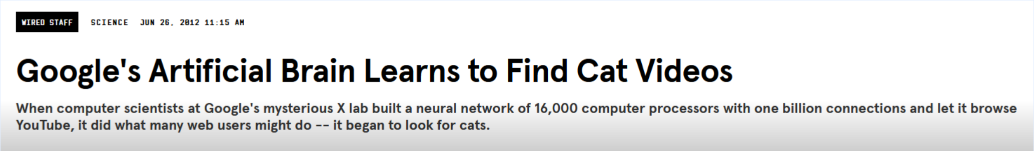


The lost decades

DNN + backprop (1980s)

2010

The arrival of Graphical Processing Units



WIRED STAFF SCIENCE JUN 26, 2012 11:15 AM

Google's Artificial Brain Learns to Find Cat Videos

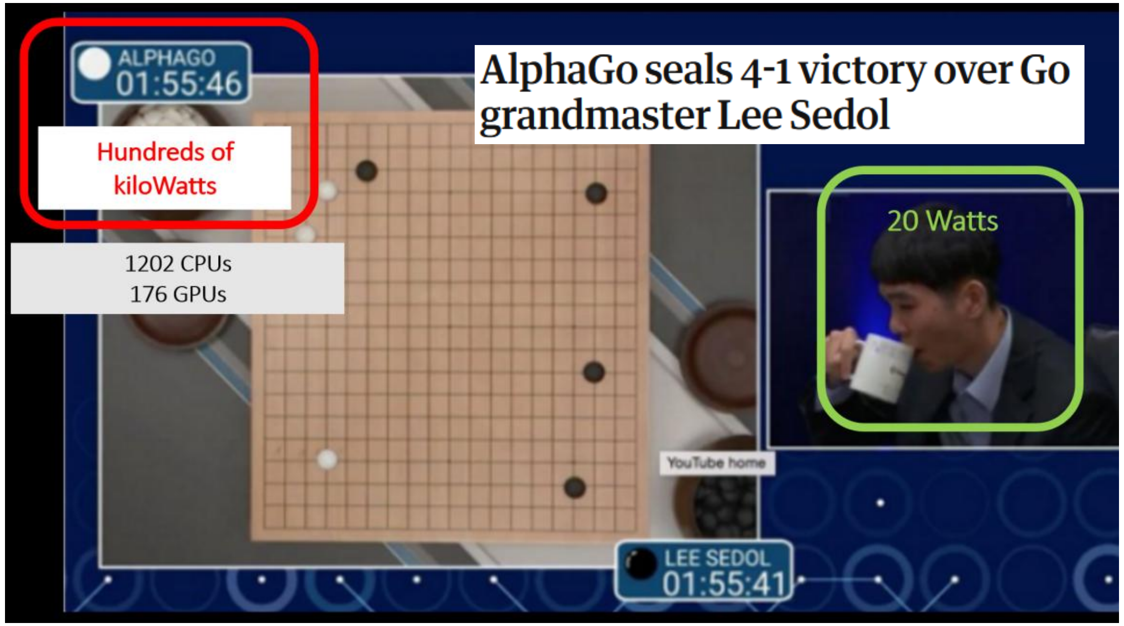
When computer scientists at Google's mysterious X lab built a neural network of 16,000 computer processors with one billion connections and let it browse YouTube, it did what many web users might do -- it began to look for cats.

16,000 Central Processing Units (CPUs) (2010)

→ 48 Graphical Processing Units (GPUs) (2012)

Hooker, "The hardware lottery", Comm. ACM (2021)

Dally et al., "Evolution of the Graphics Processing Unit", IEEE Micro (2021)



AlphaGo seals 4-1 victory over Go grandmaster Lee Sedol

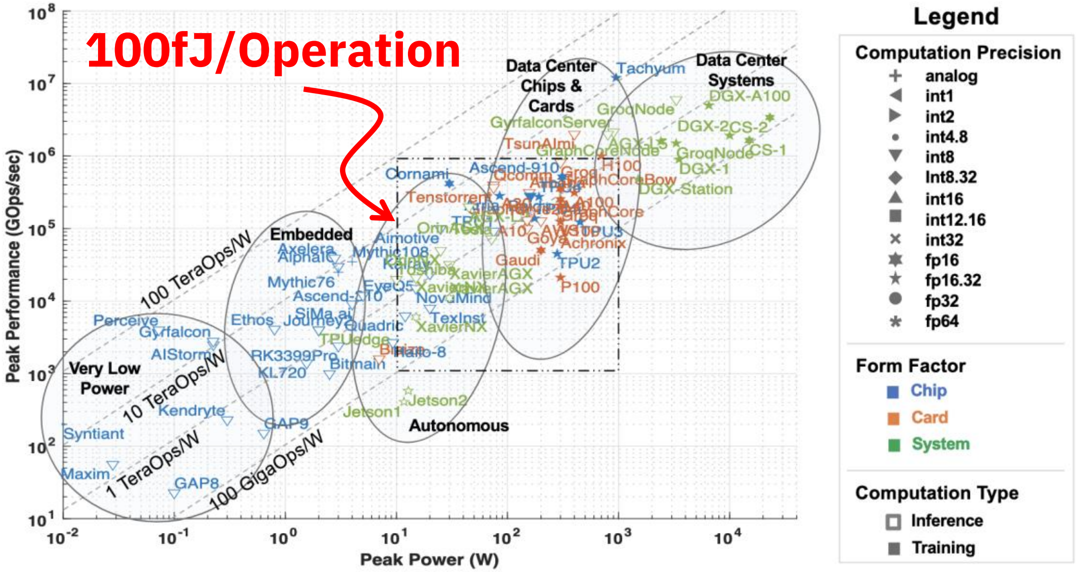
Hundreds of kiloWatts

1202 CPUs
176 GPUs

20 Watts

Silver et al., "Mastering the game of Go without human knowledge", Nature (2017)

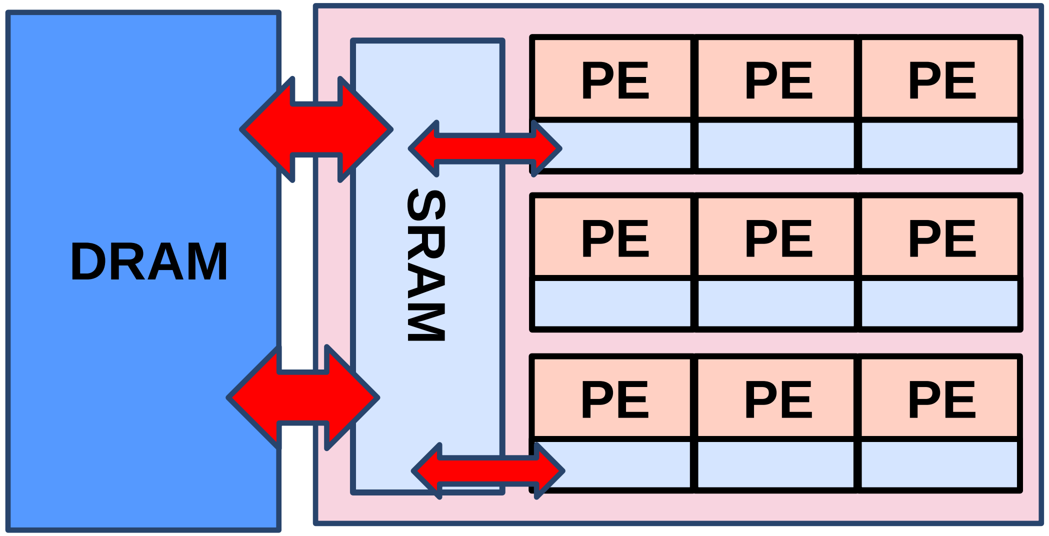
The plateauing of energy efficiency



100fJ/Operation

Reuther et al., "AI and ML Accelerator Survey and Trends", IEEE HPEC (2022)

~150-200fJ/Operation



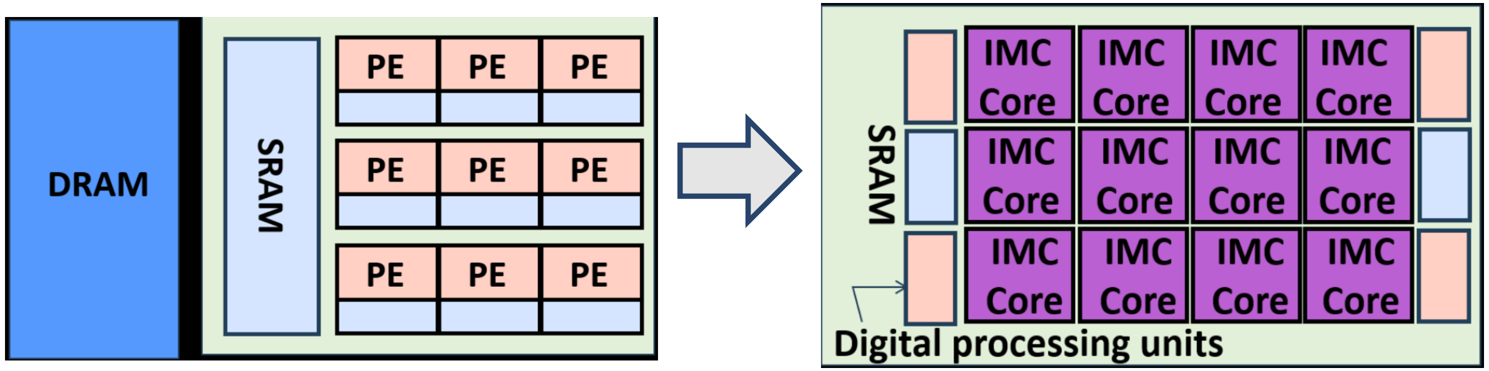
> 100 MB
40-100pJ/byte

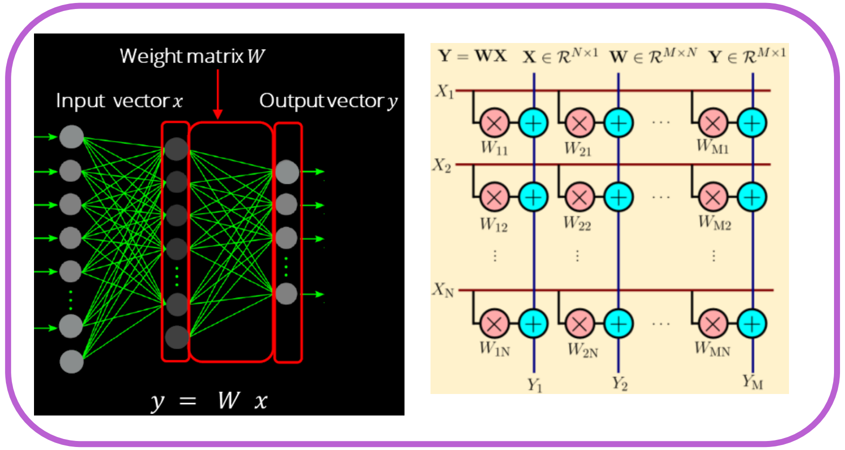
10-100KB
1-10pJ/byte

Murmann, "Mixed-Signal Computing for Deep Neural Network Inference", IEEE TVLSI (2020)

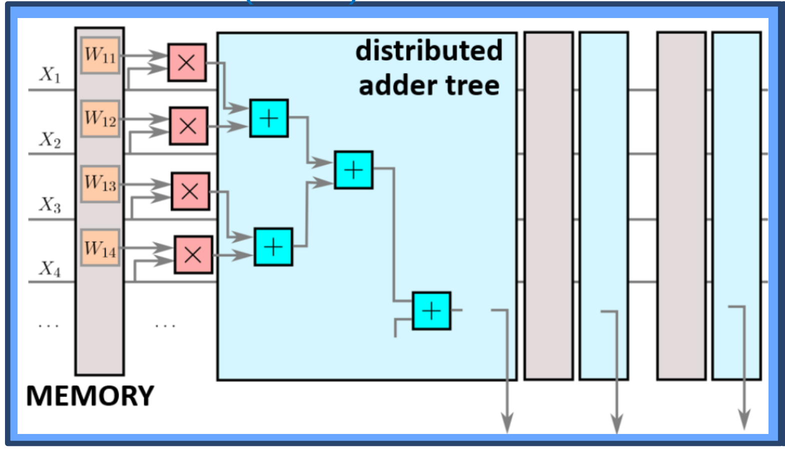
In-memory computing-based DNN accelerator

- Can we architect a DNN accelerator where the synaptic weights are kept stationary?

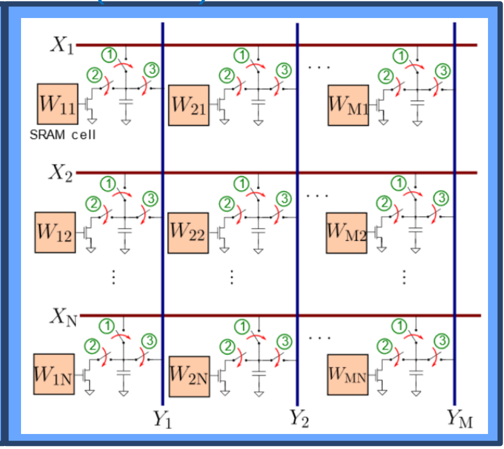



Weight matrix W
Input vector x
Output vector y
$y = W x$
$\mathbf{Y} = \mathbf{WX}$ $\mathbf{X} \in \mathcal{R}^{N \times 1}$ $\mathbf{W} \in \mathcal{R}^{M \times N}$ $\mathbf{Y} \in \mathcal{R}^{M \times 1}$
X_1
X_2
X_N
W_{11} W_{21} W_{M1}
W_{12} W_{22} W_{M2}
W_{1N} W_{2N} W_{MN}
Y_1 Y_2 Y_M

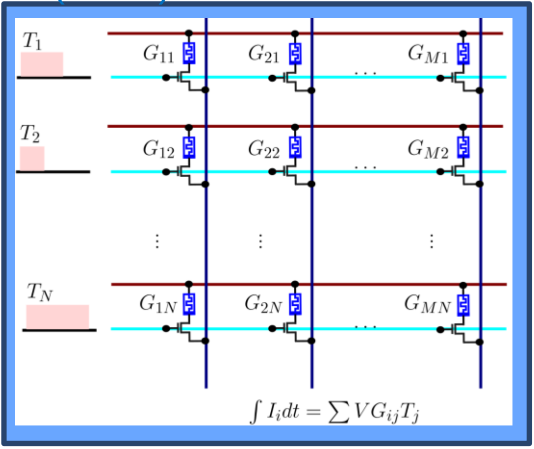
Digital In-memory Computing (DIMC) with SRAM



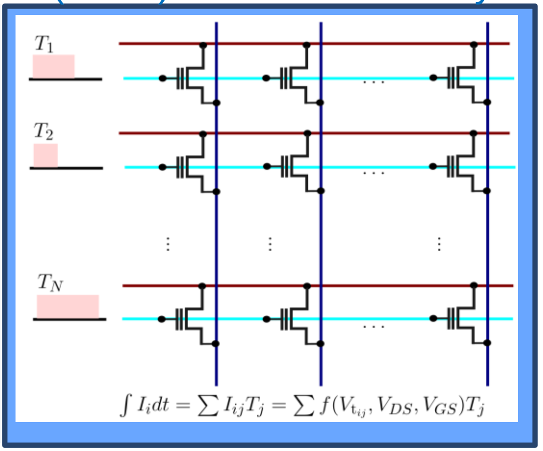
Analog In-memory Computing (AIMC) with SRAM


X_1 X_2 X_N
W_{11} W_{21} W_{M1}
SRAM cell
W_{12} W_{22} W_{M2}
W_{1N} W_{2N} W_{MN}
Y_1 Y_2 Y_M

Analog In-memory Computing (AIMC) with PCM/ReRAM

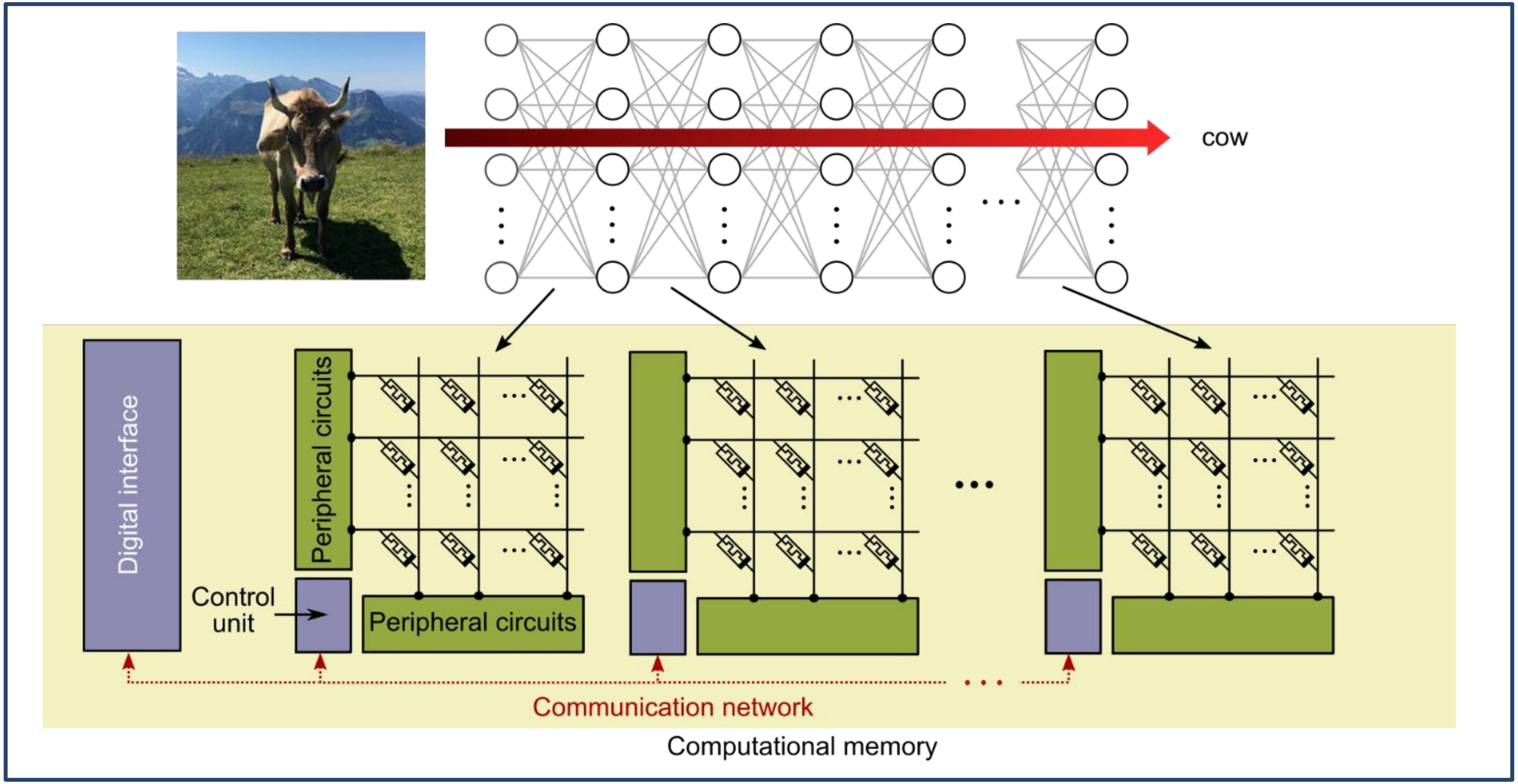

T_1 T_2 T_N
G_{11} G_{21} G_{M1}
G_{12} G_{22} G_{M2}
G_{1N} G_{2N} G_{MN}
$\int I_i dt = \sum V G_{ij} T_j$

Analog In-memory Computing (AIMC) with Flash memory


T_1 T_2 T_N
$\int I_i dt = \sum I_{ij} T_j = \sum f(V_{t_{ij}}, V_{DS}, V_{GS}) T_j$

Sebastian et al., "Memory devices and applications for in-memory computing", Nature Nano (2020)

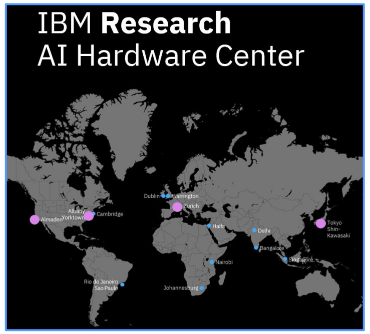
In-memory computing-based DNN accelerator



Eleftheriou et al., “Deep learning acceleration based on in-memory computing”, IBM JRD (2019)

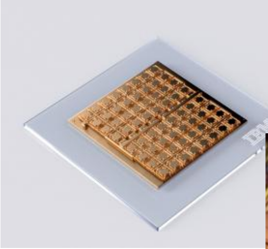
Sebastian et al., “Memory devices and applications for in-memory computing”, Nature Nanotech. (2020)

IMC-based DNN accelerators @ IBM Research - Zurich

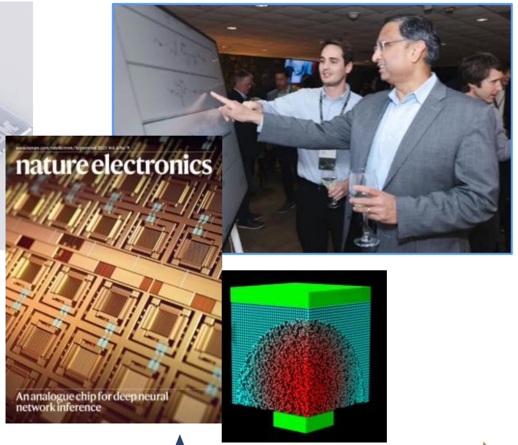


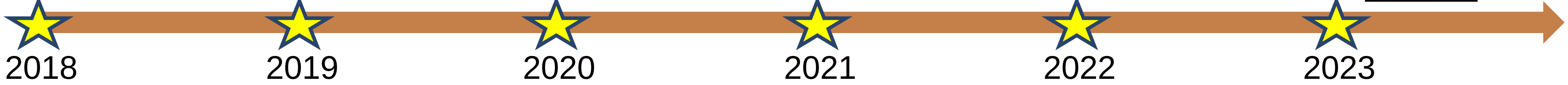
IBM AIHWKit

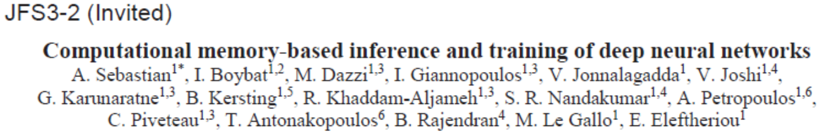
Master tomorrow's
AI Hardware today.
Run an AI algorithm where computation and storage coexist on a single chip.



HERMES



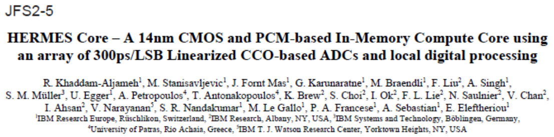




JFS3-2 (Invited)

Computational memory-based inference and training of deep neural networks

A. Sebastian[1*], I. Boybat[1,2], M. Dazzi[1,3], I. Giannopoulos[1,3], V. Jonnalagadda[1], V. Joshi[1,4], G. Karunaratne[1,3], B. Kersting[1,5], R. Khaddam-Aljameh[1,3], S. R. Nandakumar[1,4], A. Petropoulos[1,6], C. Piveteau[1,3], T. Antonakopoulos[6], B. Rajendran[4], M. Le Gallo[1], E. Eleftheriou[1]

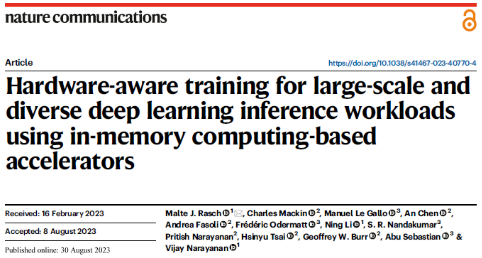


JFS2-5

HERMES Core – A 14nm CMOS and PCM-based In-Memory Compute Core using an array of 300ps/LSB Linearized CCO-based ADCs and local digital processing

R. Khaddam-Aljameh[1], M. Stanisavljevic[1], J. Fornt Mas[1], G. Karunaratne[1], M. Braendli[1], F. Liu[2], A. Singh[1], S. M. Müller[3], U. Egger[1], A. Petropoulos[4], T. Antonakopoulos[4], K. Brew[2], S. Choi[2], I. Ok[2], F. L. Lie[2], N. Saulnier[2], V. Chan[2], I. Ahsan[2], V. Narayanan[5], S. R. Nandakumar[1], M. Le Gallo[1], P. A. Francese[1], A. Sebastian[1], E. Eleftheriou[1]

[1]IBM Research Europe, Rüschlikon, Switzerland, [2]IBM Research, Albany, NY, USA, [3]IBM Systems and Technology, Böblingen, Germany, [4]University of Patras, Rio Achaia, Greece, [5]IBM T. J. Watson Research Center, Yorktown Heights, NY, USA



nature communications

Article

https://doi.org/10.1038/s41467-023-40770-4

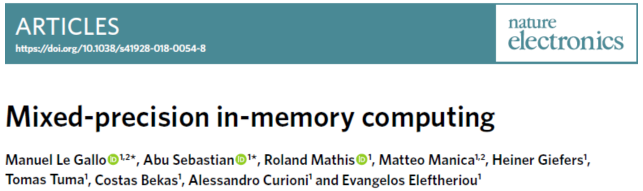
Hardware-aware training for large-scale and diverse deep learning inference workloads using in-memory computing-based accelerators

Received: 16 February 2023

Accepted: 8 August 2023

Published online: 30 August 2023

Malte J. Rasch[1]✉, Charles Mackin[2], Manuel Le Gallo[3], An Chen[2], Andrea Fasoli[2], Frédéric Odermatt[3], Ning Li[1], S. R. Nandakumar[3], Pritish Narayanan[2], Hsinyu Tsai[2], Geoffrey W. Burr[2], Abu Sebastian[3] & Vijay Narayanan[1]



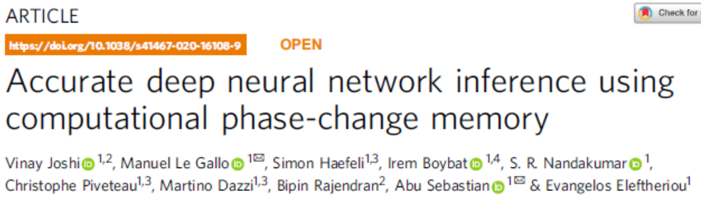
ARTICLES

https://doi.org/10.1038/s41928-018-0054-8

nature electronics

Mixed-precision in-memory computing

Manuel Le Gallo[1,2*], Abu Sebastian[1*], Roland Mathis[1], Matteo Manica[1,2], Heiner Giefers[1], Tomas Tuma[1], Costas Bekas[1], Alessandro Curioni[1] and Evangelos Eleftheriou[1]



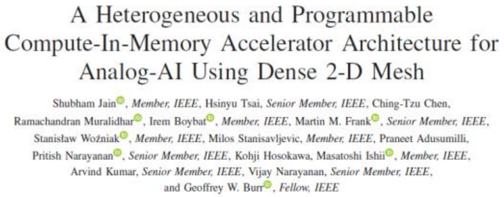
ARTICLE

https://doi.org/10.1038/s41467-020-16108-9 OPEN

Check for

Accurate deep neural network inference using computational phase-change memory

Vinay Joshi[1,2], Manuel Le Gallo[1]✉, Simon Haefeli[1,3], Irem Boybat[1,4], S. R. Nandakumar[1], Christophe Piveteau[1,3], Martino Dazzi[1,3], Bipin Rajendran[2], Abu Sebastian[1]✉ & Evangelos Eleftheriou[1]



A Heterogeneous and Programmable Compute-In-Memory Accelerator Architecture for Analog-AI Using Dense 2-D Mesh

Shubham Jain, *Member, IEEE*, Hsinyu Tsai, *Senior Member, IEEE*, Ching-Tzu Chen, Ramachandran Muralidhar, Irem Boybat, *Member, IEEE*, Martin M. Frank, *Senior Member, IEEE*, Stanisław Woźniak, *Member, IEEE*, Milos Stanisavljevic, *Member, IEEE*, Praneet Adusumilli, Pritish Narayanan, *Senior Member, IEEE*, Kohji Hosokawa, Masatoshi Ishii, *Member, IEEE*, Arvind Kumar, *Senior Member, IEEE*, Vijay Narayanan, *Senior Member, IEEE*, and Geoffrey W. Burr, *Fellow, IEEE*

erc
European Research Council
Established by the European Commission

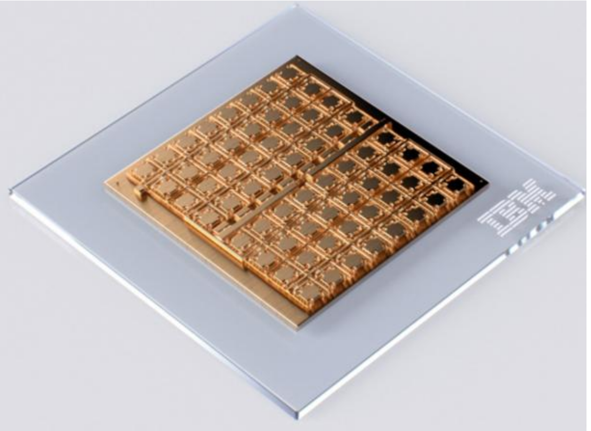
THE FRAMEWORK PROGRAMME FOR RESEARCH AND INNOVATION
HORIZON 2020

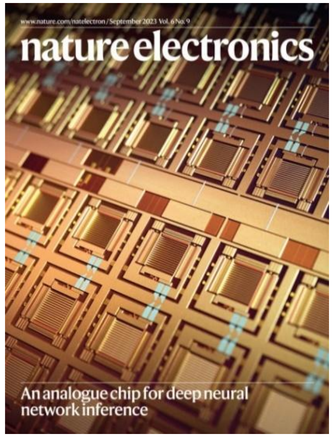
Horizon Europe

FNSNF
FONDS NATIONAL SUISSE
SCHWEIZERISCHER NATIONALFONDS
FONDO NAZIONALE SVIZZERO
SWISS NATIONAL SCIENCE FOUNDATION

IBM's Recent AIMC-based AI Chip Prototypes

IBM HERMES project chip



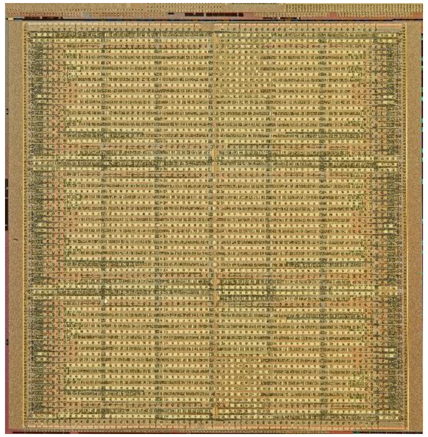


nature electronics

An analogue chip for deep neural network inference

Le Gallo et al., Nature Electronics (2023)

- 64 tiles with 256x256 crossbar arrays
- ADCs integrated per tile along with digital processing units
- A digital communication fabric

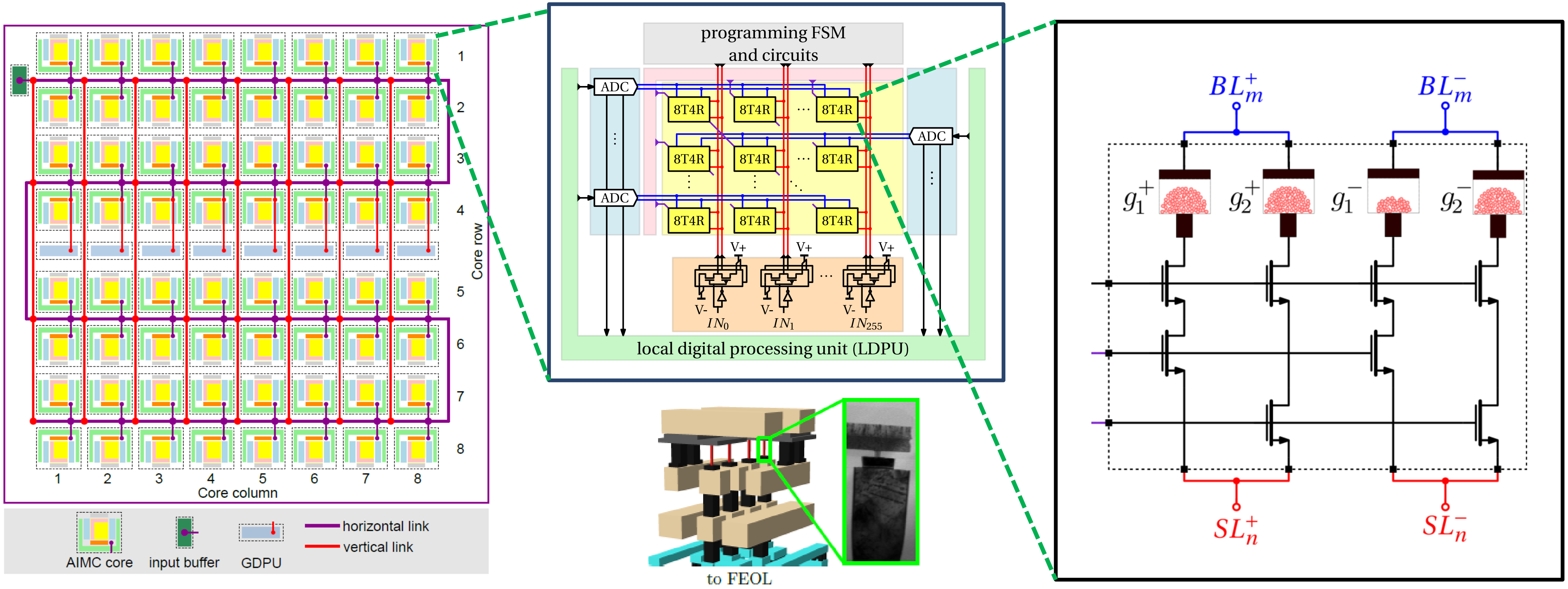


Ambrogio et al., Nature (2023)

- 34 tiles with 512x512 crossbar arrays
- ADCs not integrated per tile
- Analog communication: Uses a 2D routing mesh to transmit data in duration format

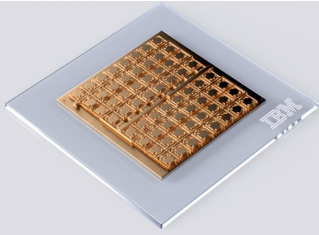
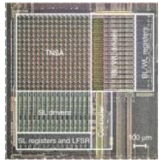
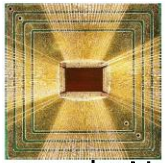
Fabricated in 14nm CMOS technology node with embedded phase-change memory in the back-end

IBM Hermes Project Chip: architectural overview



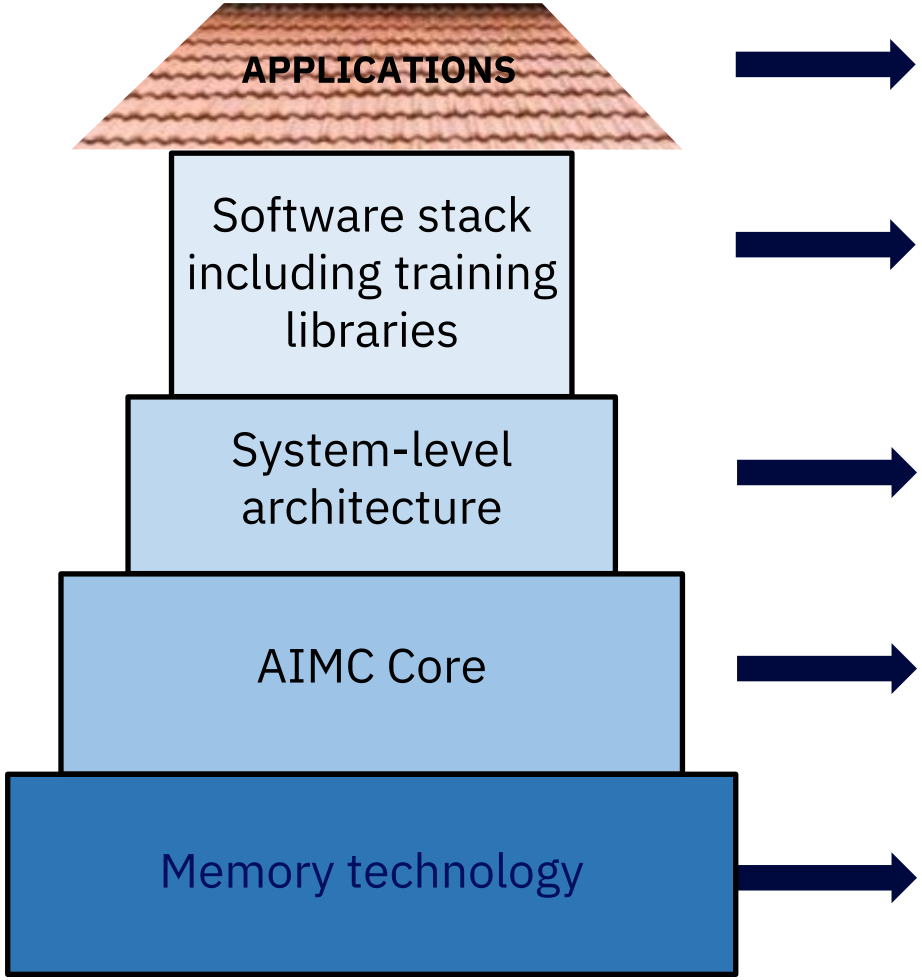
- Each of the 64 cores comprises 256x256 crossbar arrays of unit cells with peripheral circuitry
- Each core has 256 integrated CCO-based ADCs and 32 current DACs for programming
- Each unit cell comprises four phase-change memory devices (Total: 16M PCM devices)
- On-chip local and global digital processing as well as a communication fabric

Performance

			Kwa et al., ISSCC (2022)	 Wan et al., Nature (2022)	 Hung et al., Nature Electr. (2021)
CMOS technology	14 nm		40nm	130 nm	22 nm
AIMC device	PCM		PCM	RRAM	RRAM
Chip area (mm^2)	144		18	159	6
Number of cores (core size)	64 (256x256)		8 (256x1024)	48 (256x256)	8 (1024x512)
Number of weights	4.2M		400k	1.6M	524k
Input/weight/output precision	8b/Analog/8b		8b/8b/19b	4b/Analog/6b	8b/8b/14b
CIFAR-10 accuracy	-	92.81%	91.89%	85.66%	92.01%
Peak MVM throughput (TOPS)	63.1	16.1	0.475	0.754	0.0337
MVM TOPS/W	9.76	2.48	20.5	16	15.6
MVM GOPS/mm^2	1550	400	26.4	4.7	5.61
Non-MVM operations supported	ReLU/sigmoid/tanh, batch norm., LSTM hidden state, partial sum accumulation		-	ReLU/sigmoid/tanh	-

- Highest CIFAR-10 accuracy
- >15x higher MVM throughput per area than SoA resistive-memory chips
- MVM TOPS/W somewhat lower due to large number of ADCs and LDPUs (>75% of power)

Towards next gen AIMC chips



APPLICATIONS

Software stack including training libraries

System-level architecture

AIMC Core

Memory technology

- Neural architecture search for DNN inference
- Applications beyond DNN inference that can leverage AIMC chips

- Improved yet light-weight hardware-aware training libraries
- Software stack for fully weight-stationary architectures

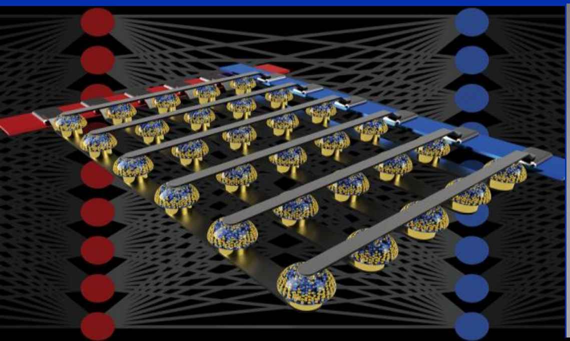
- A heterogeneous compute architecture
- Efficient communication fabric for activations

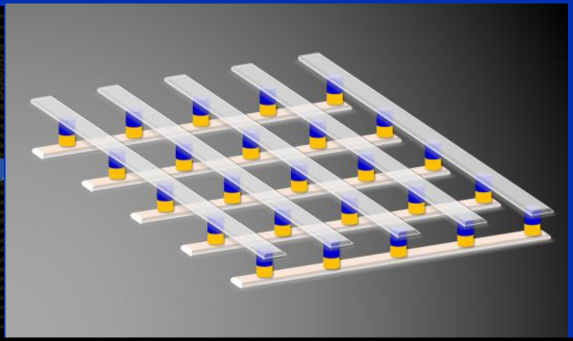
- More efficient data converters
- Lower the integration time

- High compute density (dense integration)
- Higher compute precision
- Ultra-high on-chip weight capacity

Software: IBM Analog AI Hardware Acceleration Kit

https://aihw-composer.draco.res.ibm.com/





Overview

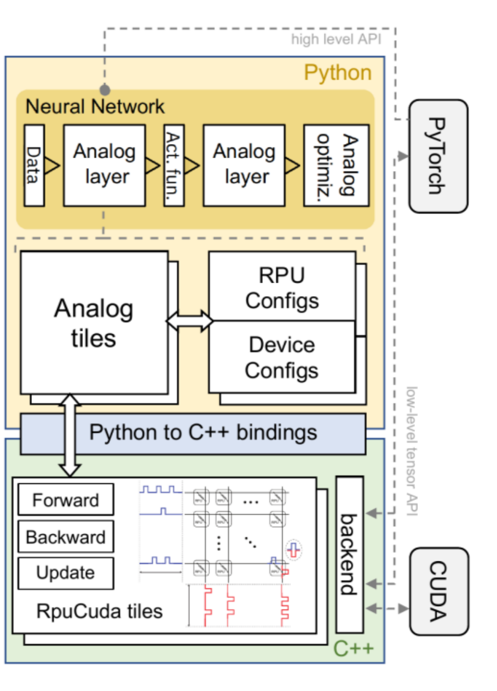
- Analog crossbar simulator that focuses on the algorithmic level and algorithmic advances of Analog AI
- Analog AI training and inference simulations
- Bring your own models and datasets to evaluate the impact of emerging analog AI hardware on your DL workloads using the flexibility of PyTorch

M. Rasch et al., AICAS, 2021

Le Gallo et al., APL Machine Learning, 2023

Capabilities:

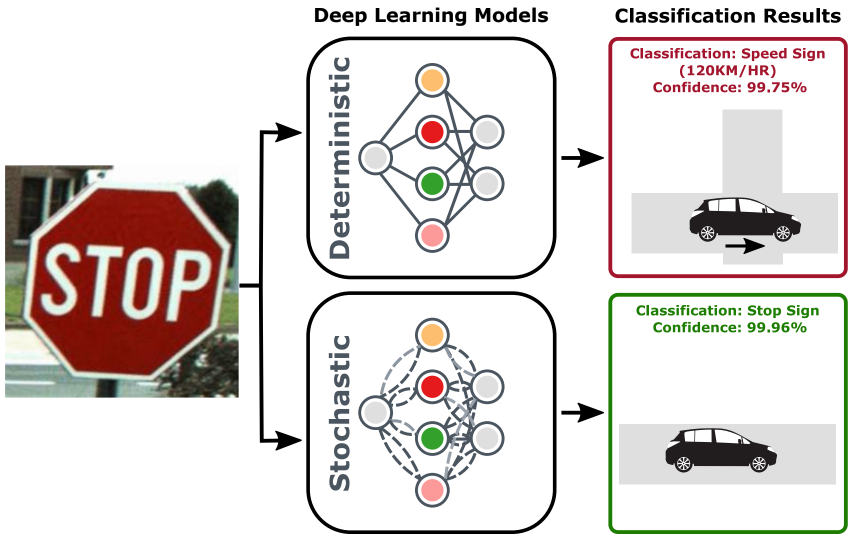
- Simulate analog neural network operation including forward/backward pass and update
- Abstract functional models of material characteristics with adjustable parameters
- Hardware-aware training for inference capability
- Inference simulator with drift and statistical (programming) noise models calibrated on hardware
- Full GPU support and substantial online documentation

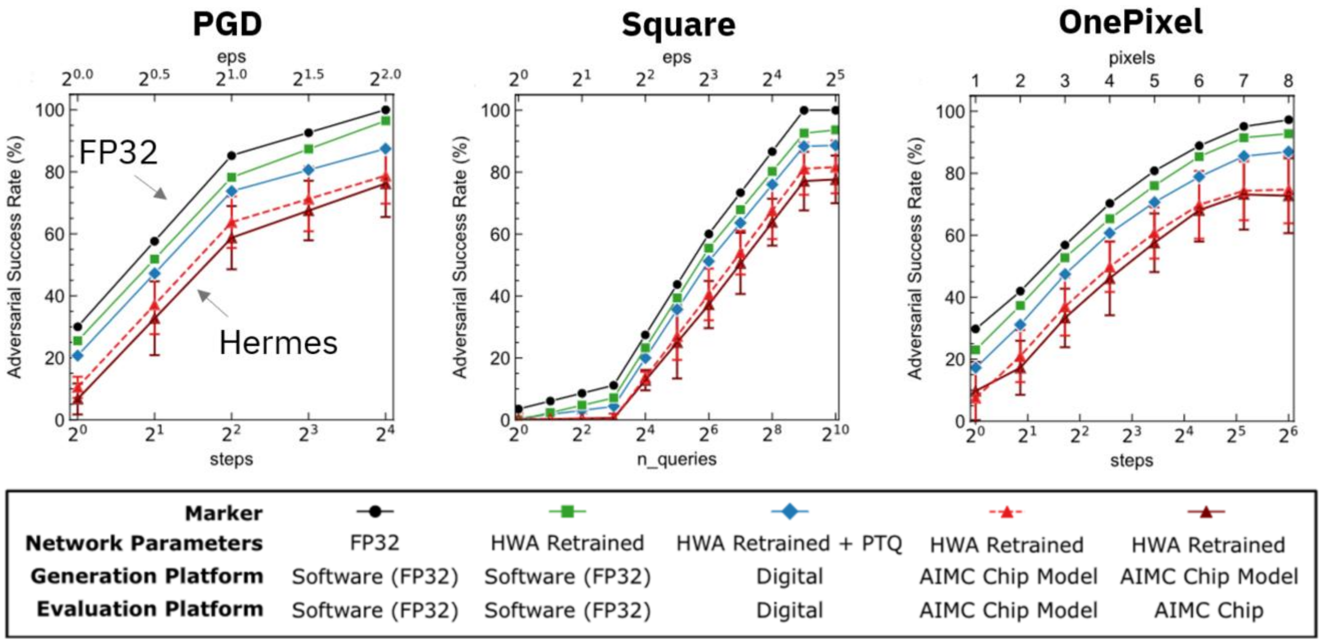


Roadmap

- Additional neural network layers
- Algorithmic advances to improve training and inference accuracy
- Premium hardware demonstrations
- Real hardware demonstrations
- Analog materials device builder

Applications: Inherent adversarial robustness of AIMC





Marker	●	■	◆	▲	▲
Network Parameters	FP32	HWA Retrained	HWA Retrained + PTQ	HWA Retrained	HWA Retrained
Generation Platform	Software (FP32)	Software (FP32)	Digital	AIMC Chip Model	AIMC Chip Model
Evaluation Platform	Software (FP32)	Software (FP32)	Digital	AIMC Chip Model	AIMC Chip

Lammie et al., Nature Comm. (2025)

- Experimental proof that networks implemented on AIMC hardware are inherently more robust to adversarial attacks than when implemented on digital hardware
- Experimentally validated on the IBM HERMES project chip
- The cause of this additional robustness is the intrinsic noise of the AIMC devices

Promising Application Domains

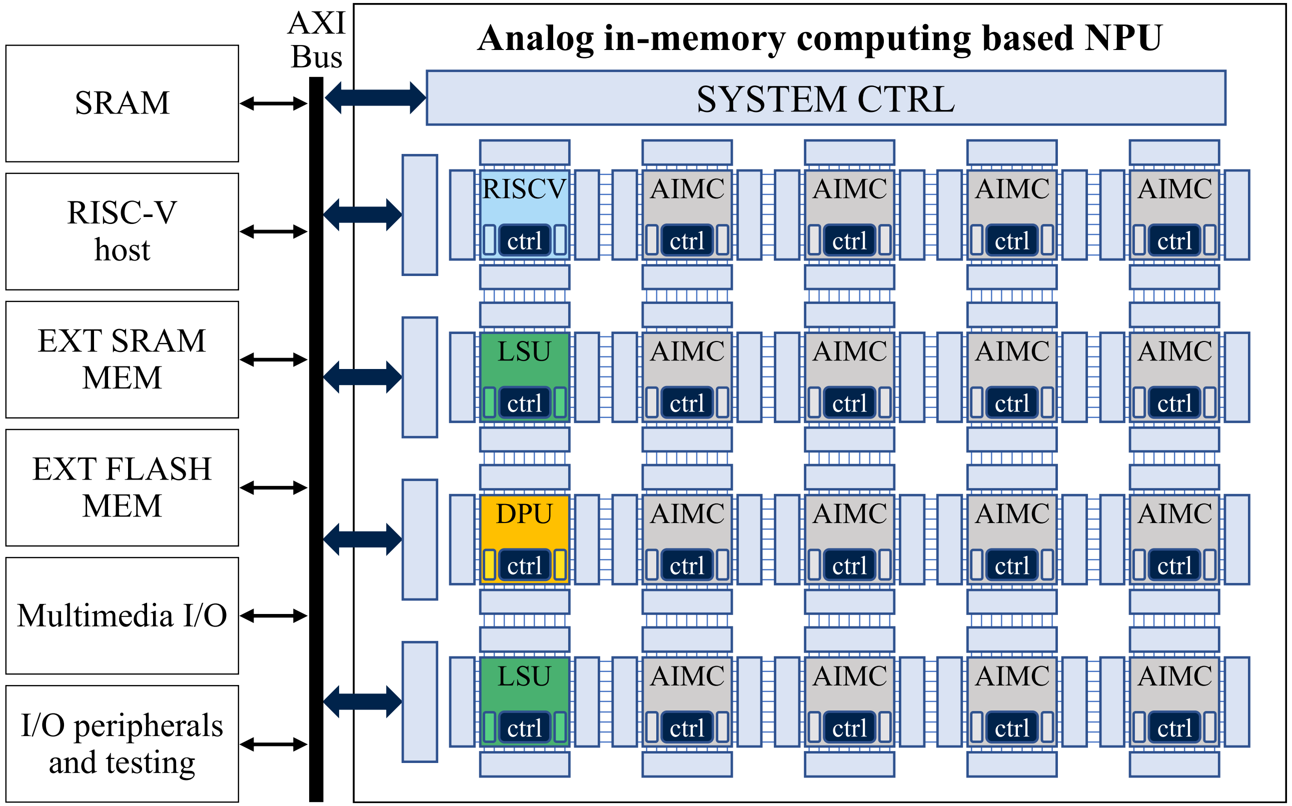
Embedded neural processing units

Boybat et al., "Heterogeneous Embedded Neural Processing Units Utilizing PCM-based Analog In-Memory Computing", IEDM (2024)

Stand-alone accelerators

Büchel et al., "Efficient Scaling of Large Language Models with Mixture of Experts and 3D Analog In-Memory Computing", Nature CS (2025)

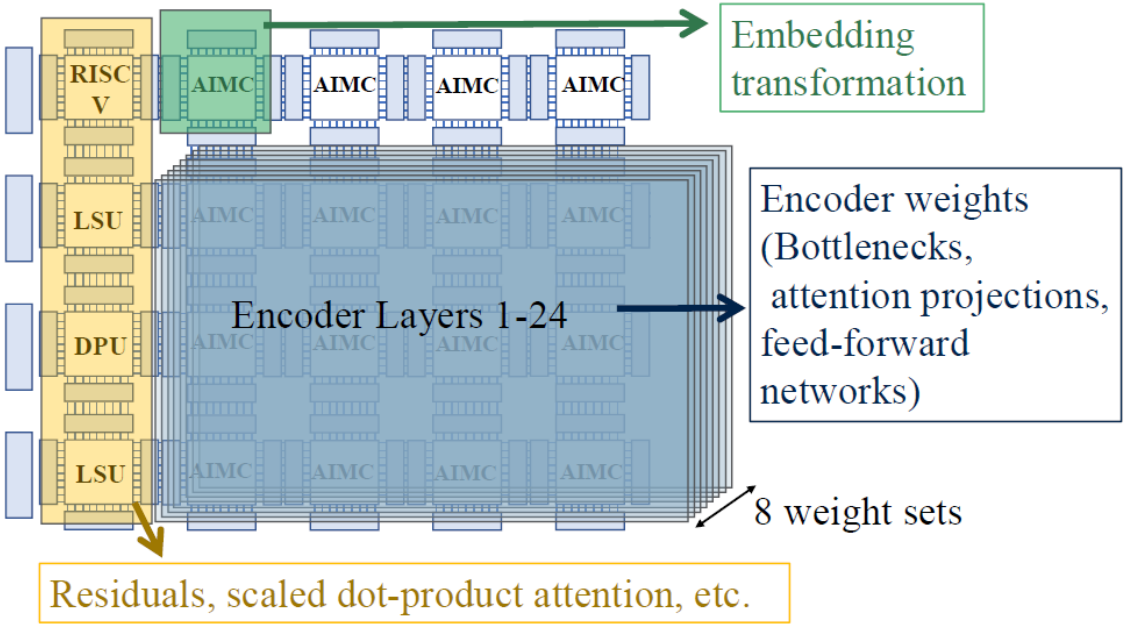
An AIMC-based NPU Architecture



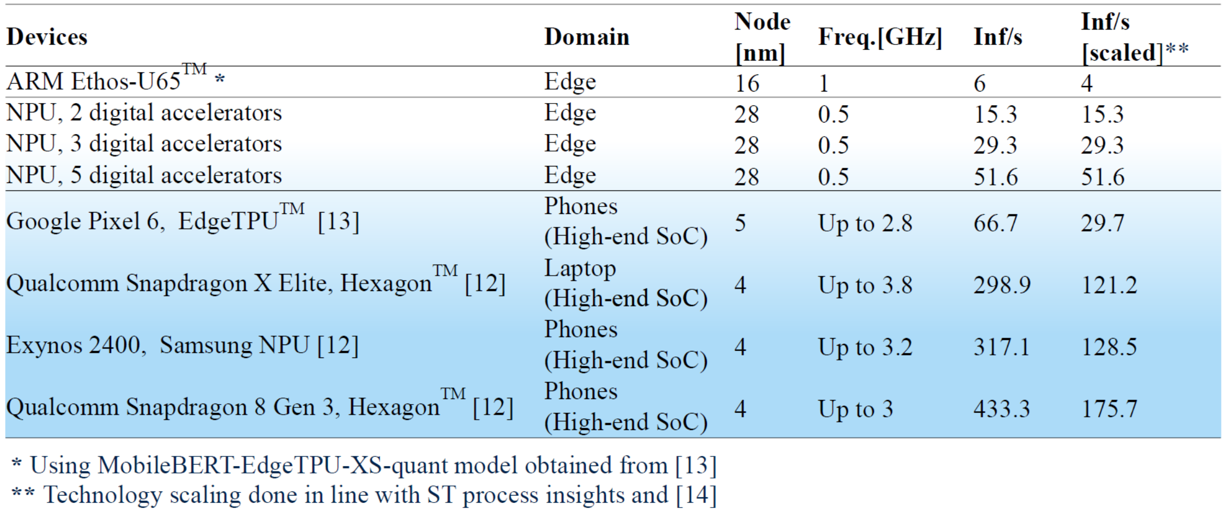
Boybat et al., IEDM (2024)

- Multiple AIMC and Digital accelerator nodes interconnected by a communication fabric
- Different flavors of custom and programmable digital accelerator nodes
- The NPU configuration is estimated to have ~30 mm² area and ~1W average power dissipation on ST's 28nm FD-SOI technology at 500MHz

NPU Architecture: Performance



RISC V
AIMC
AIMC
AIMC
AIMC
Embedding transformation
LSU
DPU
LSU
Encoder Layers 1-24
Encoder weights (Bottlenecks, attention projections, feed-forward networks)
8 weight sets
Residuals, scaled dot-product attention, etc.



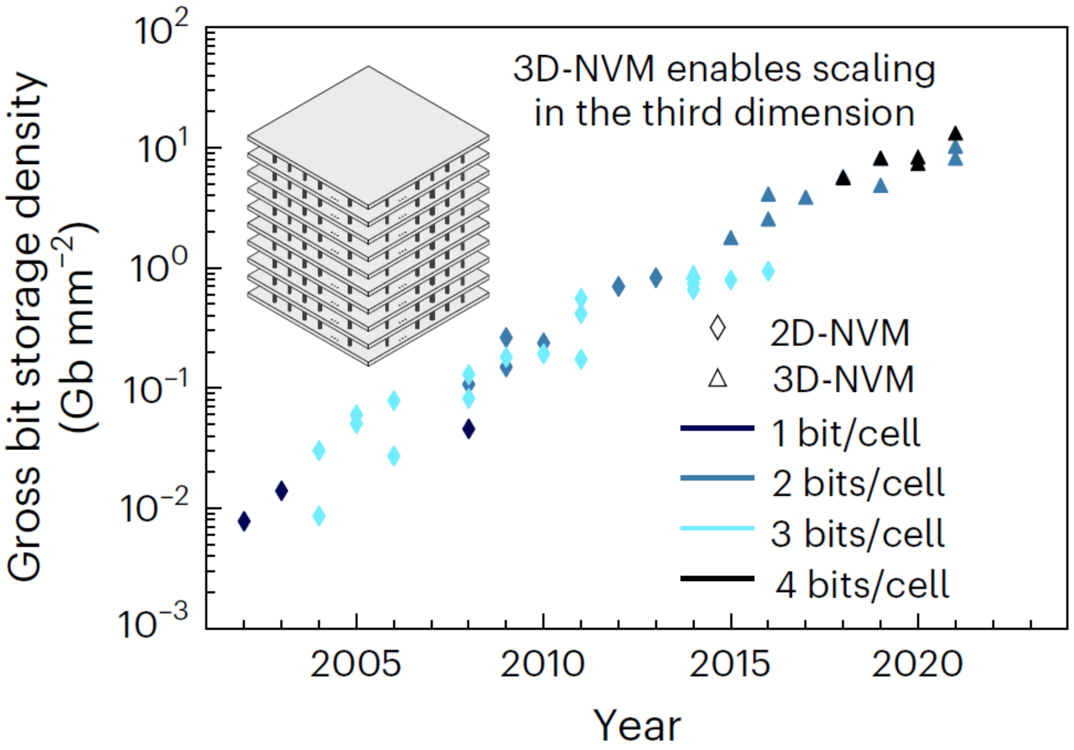
Devices	Domain	Node [nm]	Freq.[GHz]	Inf/s	Inf/s [scaled]**
ARM Ethos-U65™ *	Edge	16	1	6	4
NPU, 2 digital accelerators	Edge	28	0.5	15.3	15.3
NPU, 3 digital accelerators	Edge	28	0.5	29.3	29.3
NPU, 5 digital accelerators	Edge	28	0.5	51.6	51.6
Google Pixel 6, EdgeTPU™ [13]	Phones (High-end SoC)	5	Up to 2.8	66.7	29.7
Qualcomm Snapdragon X Elite, Hexagon™ [12]	Laptop (High-end SoC)	4	Up to 3.8	298.9	121.2
Exynos 2400, Samsung NPU [12]	Phones (High-end SoC)	4	Up to 3.2	317.1	128.5
Qualcomm Snapdragon 8 Gen 3, Hexagon™ [12]	Phones (High-end SoC)	4	Up to 3	433.3	175.7

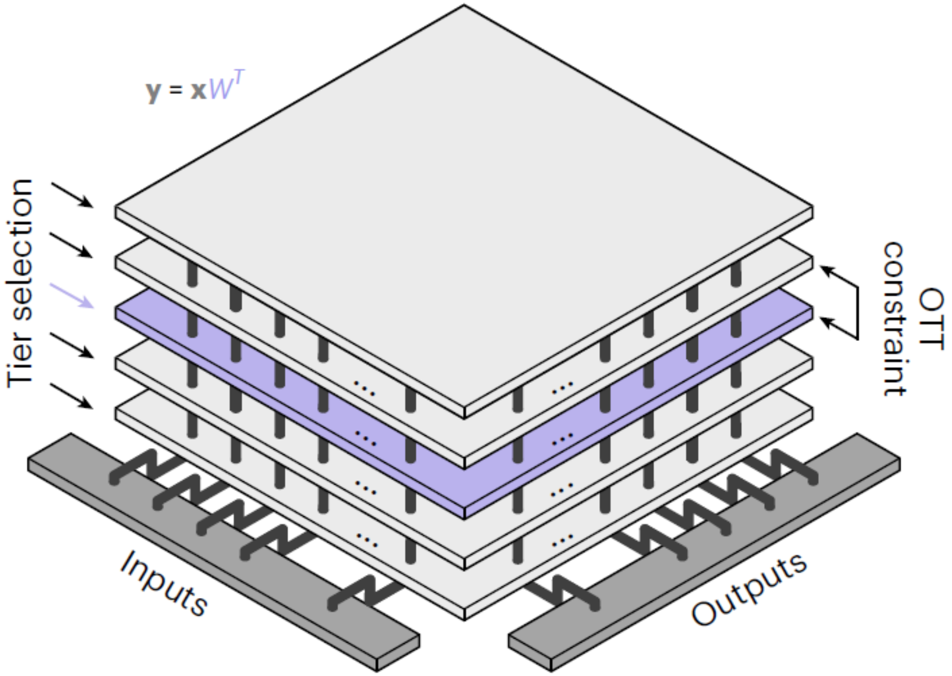
* Using MobileBERT-EdgeTPU-XS-quant model obtained from [13]
** Technology scaling done in line with ST process insights and [14]

Boybat et al., IEDM (2024)

- Supports a wide range of Neural Network (NN) models, including CNNs, LSTMs, and even Transformers
- The NPU configuration is estimated to deliver competitive inference throughput approaching the performance of high-end SoCs for mobile devices

Stand-alone Accelerators Based on 3D AIMC





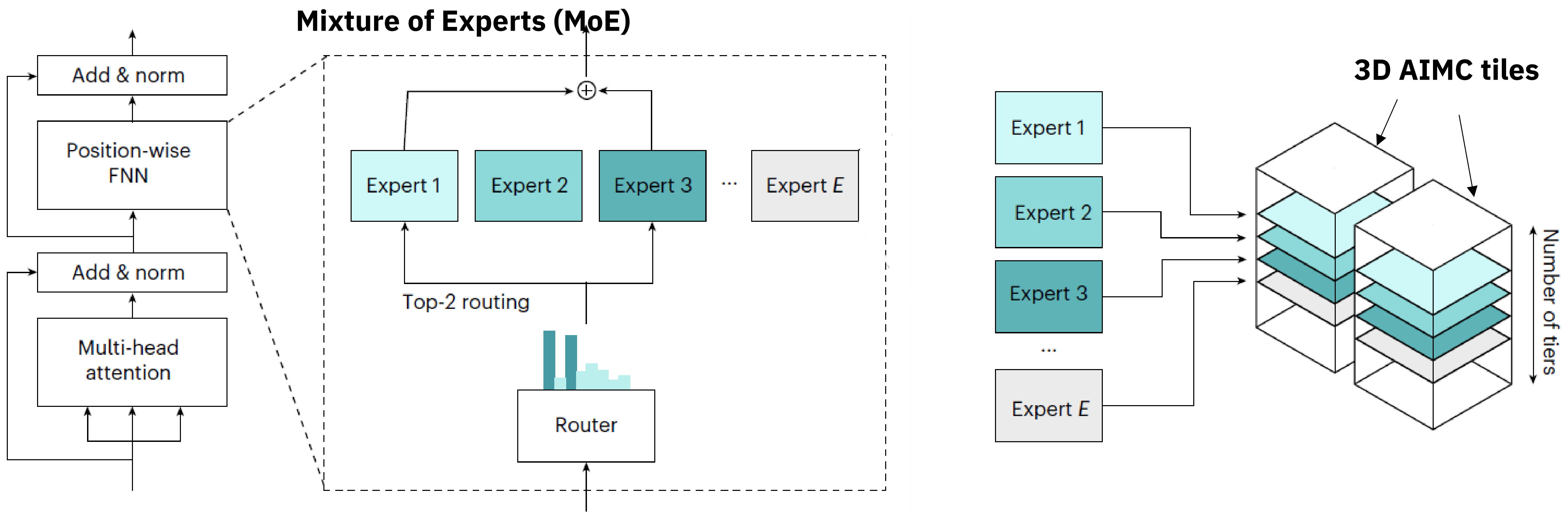
Buechel et al., Nature Computational Science (2025)

Shim et al., Proc. MEMSYS (2020)

Bavandpour et al., Neuromorphic Computing and Engineering (2021)

- Dramatic improvements in storage density for 3D non-volatile memory
- 3D non-volatile memory will be a game-changer for AIMC
- Would facilitate full-weight stationarity even for very large networks

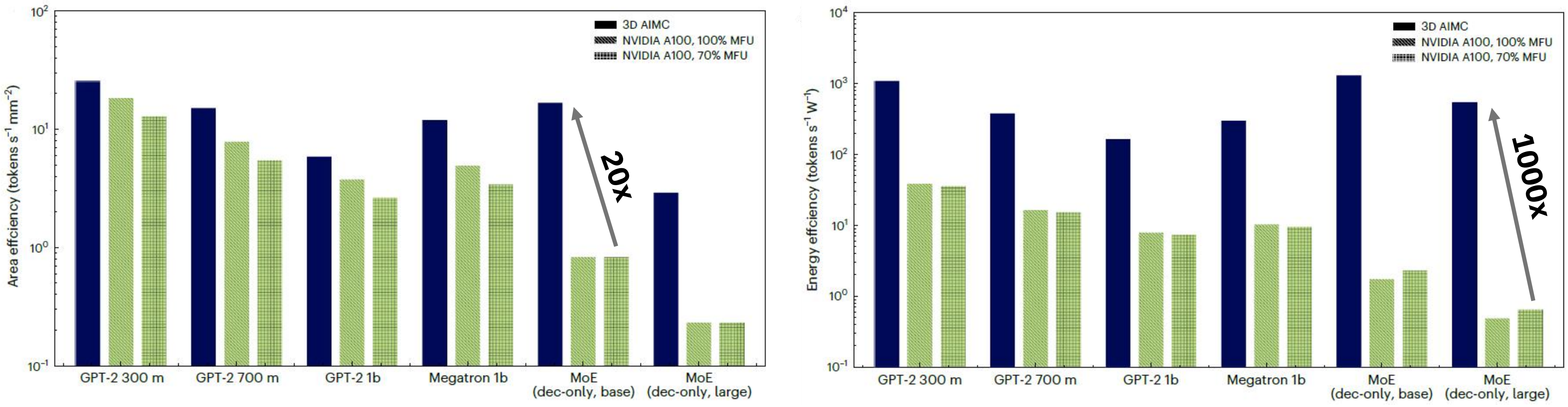
Mixture of Experts (MoEs) with 3D AIMC



Buechel et al., Nature Computational Science (2025)

- MoEs replace each feedforward network in a Transformer with multiple expert networks
- Faster inference than dense networks with less parameters and higher accuracy
- Each expert can be implemented on a tier of a 3D AIMC tile -> **Ideal fit for 3D AIMC!**

Mixture of Experts with 3D AIMC: Performance



Throughput-per-area

Throughput-per-watt

Buechel et al., Nature Computational Science (2025)

- Estimates from simulations show that throughput-per-area of 3D AIMC is up to 20x higher than A100 GPU, and energy efficiency is up to 1000x higher than A100 GPU

Summary

- **In-memory computing is arguably the next key breakthrough towards further improving the computational efficiency of deep neural networks**
  - Brain-inspired: Stationary synapses and analog processing
  - Arguably the only way to run LLMs on a thumb-drive sized system
- **Advanced research prototype chips in 14nm CMOS technology with embedded PCM**
  - Demonstrate the feasibility of achieving software-equivalent accuracies
  - Seamless interface of analog computing with digital processing units
- **Software/applications advancements**
  - AIHWKIT software library can be used to benchmark the accuracy of large networks on AIMC, now supporting LLMs
- **Two promising application domains**
  - Embedded neural processing units based on AIMC
  - Stand-alone AIMC accelerators with enormous weight capacity

Contributors

- In-memory computing group, IBM Research – Zurich,
- IBM Research – Almaden, CA, USA
- IBM Research – Albany, NY, USA
- IBM Research – Tokyo, Kawasaki, Japan
- IBM TJ Watson Research Center, NY, USA
- ETH Zürich, EPFL, Patras, KCL, Oxford, WWU Münster, Groningen etc.

THE FRAMEWORK PROGRAMME FOR RESEARCH AND INNOVATION

HORIZON 2020

Horizon Europe

FNSNF

FONDS NATIONAL SUISSE
SCHWEIZERISCHER NATIONALFONDS
FONDO NAZIONALE SVIZZERO
SWISS NATIONAL SCIENCE FOUNDATION

erc

European Research Council
Established by the European Commission

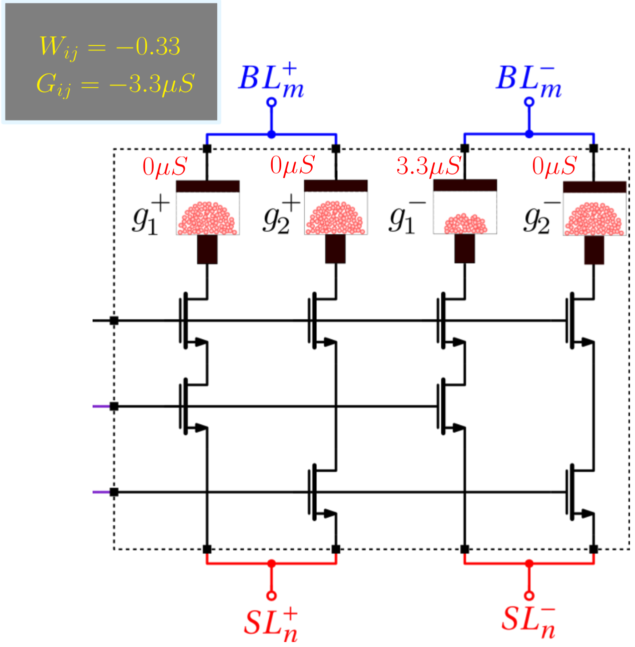
IBM **Research**
AI Hardware Center

Dublin
Warrington
Zurich
Almaden
Albany
Yorktown
Cambridge
Haifa
Delhi
Tokyo
Shin-
Kawasaki
Bangalore
Singapore
Nairobi
Rio de Janeiro
Sao Paulo
Johannesburg

DARPA

Backup

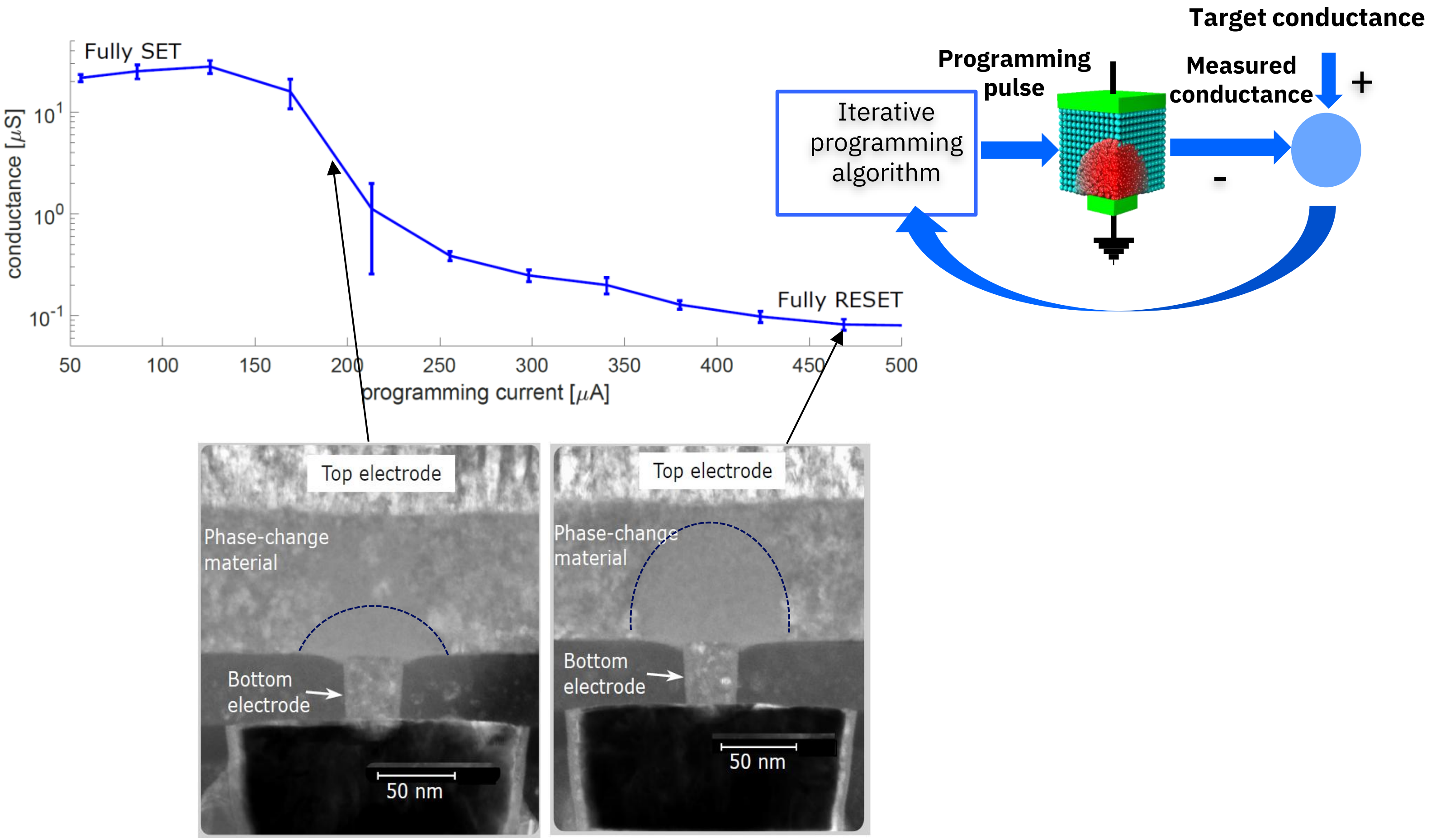
Programming to a target conductance value



$$W_{ij} \in [-1, 1]$$

$$G_{ij} = W_{ij} G_{\max}$$

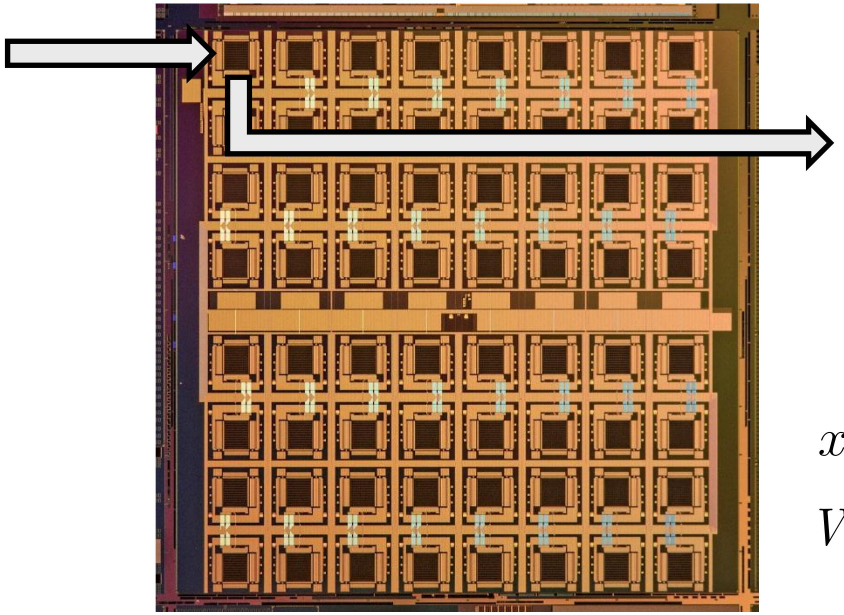
$$G_{ij} = (g_1^+ + g_2^+) - (g_1^- + g_2^-)$$



- Differential configuration to facilitate storage of bipolar weights
- Synaptic weights stored in terms of analog conductance values of one or two devices per polarity
- Iterative programming algorithms used to achieve a target conductance value

Matrix-vector multiplication

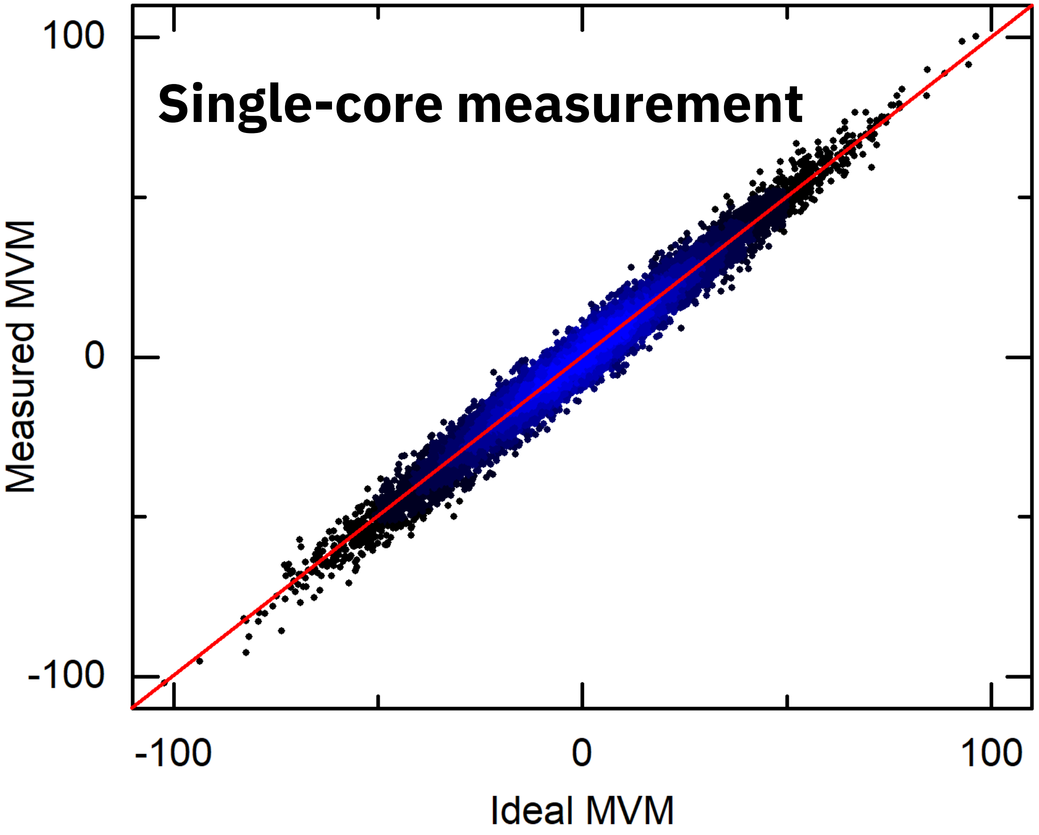
$$\begin{bmatrix} x_1^k \\ x_2^k \\ \vdots \\ x_n^k \end{bmatrix}$$



$$\begin{bmatrix} y_1^k \\ y_2^k \\ \vdots \\ y_m^k \end{bmatrix}$$

$$x_i^k \in [-1, 1]$$

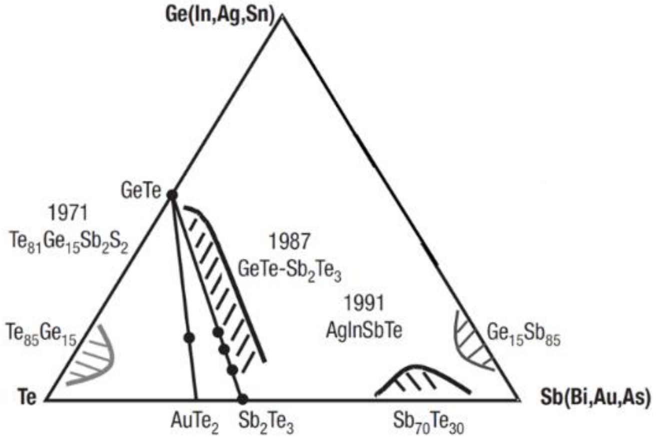
$$W_{ij} \in [-1, 1]$$



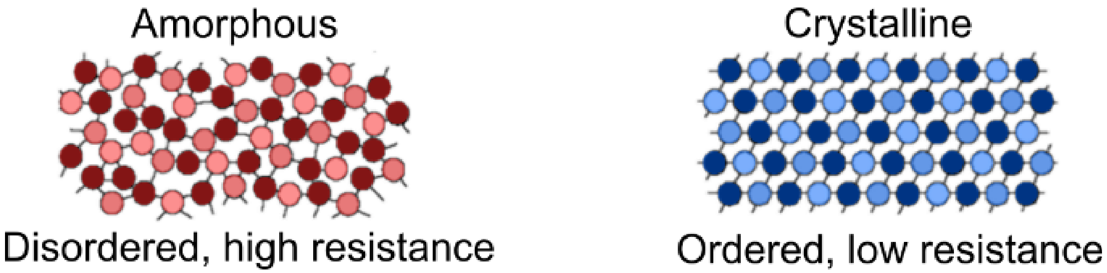
- 256-dimensional input vectors of 8-bit resolution
- Fully stationary synaptic weights stored in terms of analog conductance values
- 256-dimensional output vectors of 8-bit resolution
- Each MVM operation takes either 128ns (~1 TOPS) or 512ns (~0.25 TOPS)
- Achieved precision between 3 and 4 bits compared with 8-bit input/output and N-bit weight digital computation

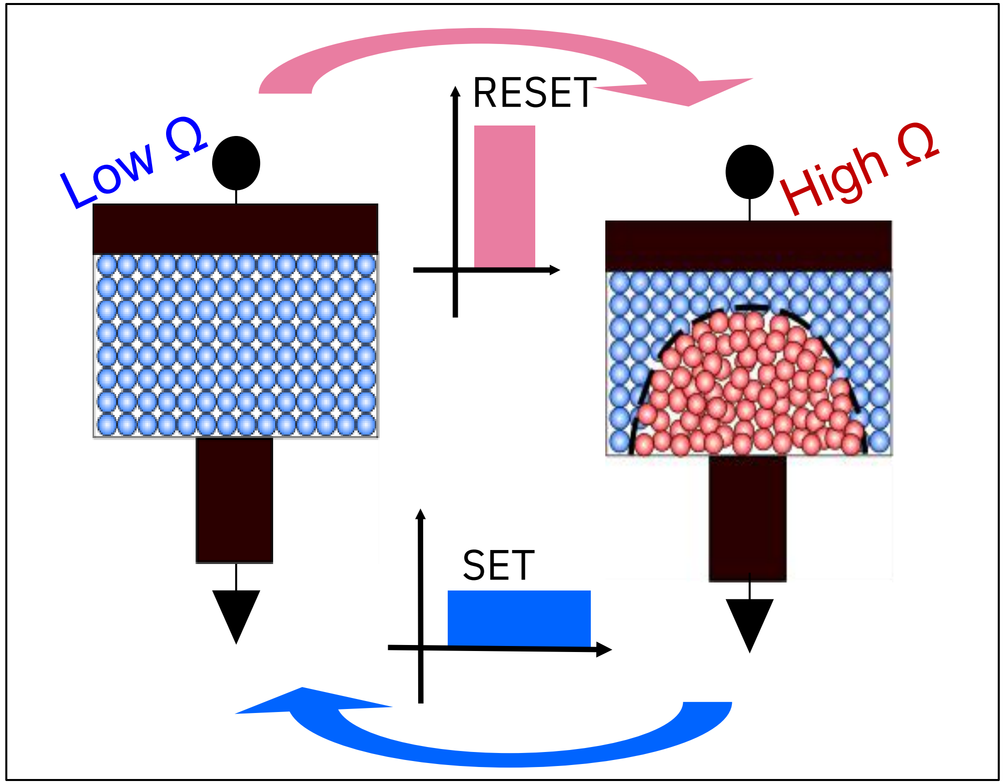
Phase-change memory

Commonly used phase change materials



Wuttig & Yamada, Nature Materials, 2007
Le Gallo et al., J. Phys. D, 2020



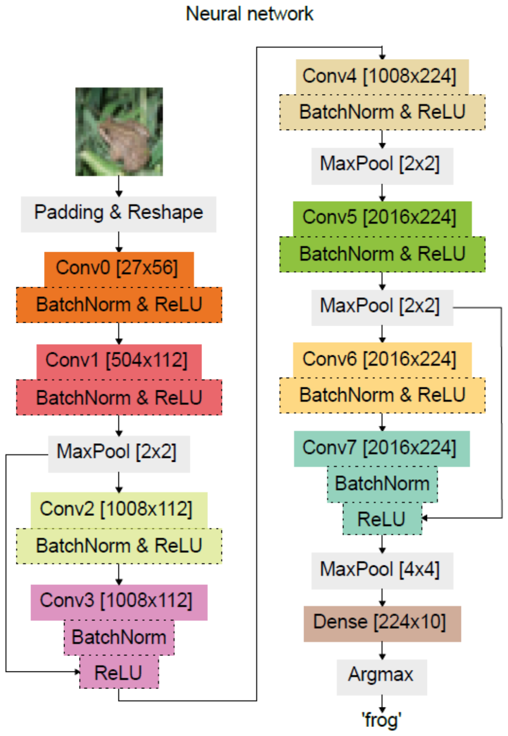


- A nanometric volume of phase change material between two electrodes
- “WRITE” Process
  - ✓ By applying a voltage pulse the material can be changed from crystalline phase (SET) to amorphous phase (RESET)
- “READ” process
  - ✓ Low-field electrical resistance

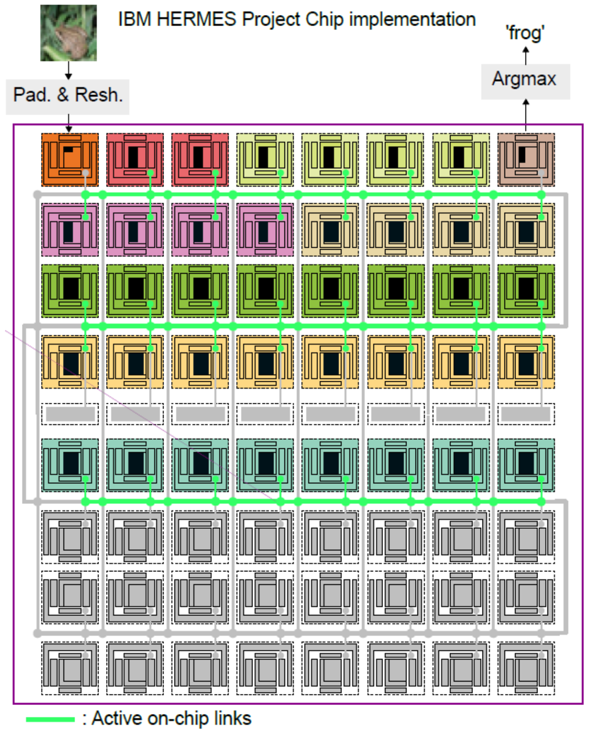
CIFAR-10 image classification

- Resnet-9 CNN with **1’866’536** weights for CIFAR-10 image classification (**40 cores**)
- All convolutional layers implemented with fully on-chip analog/digital computations
- Software accuracy: **93.67%**
- On-Chip accuracy: **92.81%**

Neural network



Hermes implementation



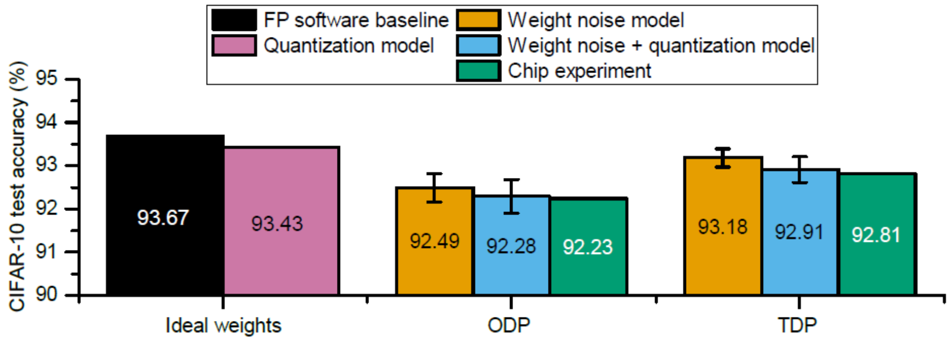
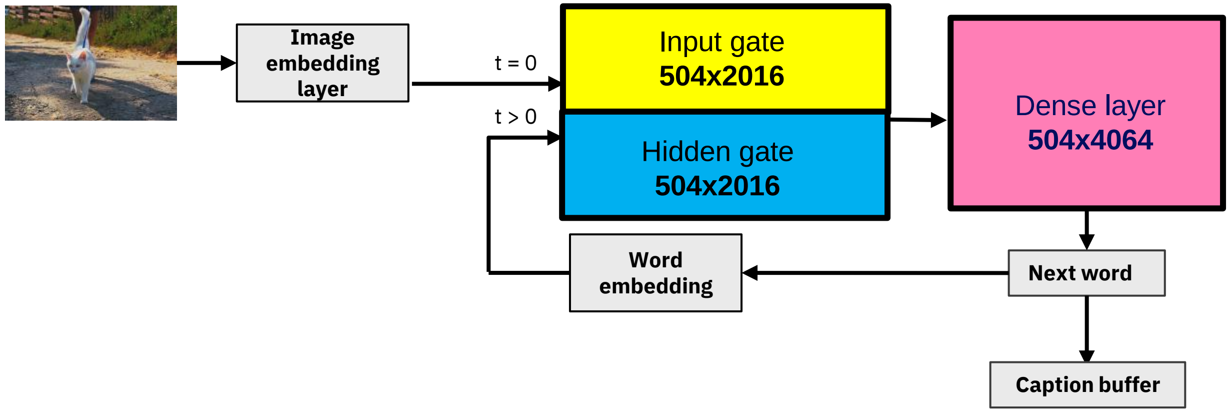


Image caption generation

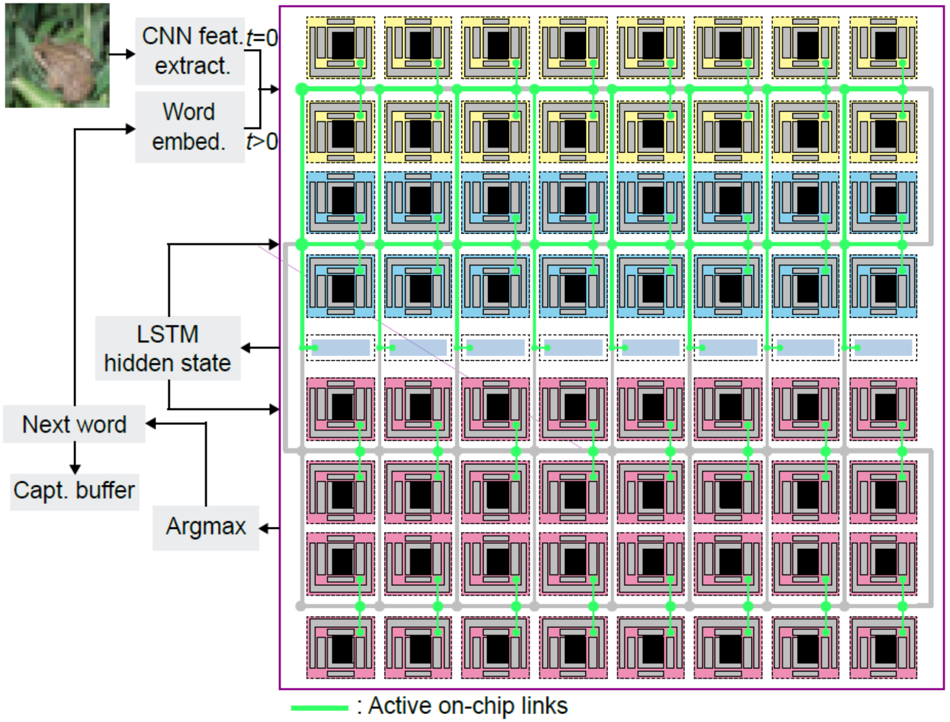
Neural network



t=0:	**Network IN**: {Image embedding}	**Network OUT:** “the”	Partial caption: “the”
t=1:	**Network IN**: “the”	**Network OUT:** “white”	Partial caption: “the white”
t=2:	**Network IN**: “white”	**Network OUT:** “cat”	Partial caption: “the white cat”
…			

- LSTM network with **4,080,384** weights used to generate image captions (**64 cores**)
- LSTM and dense layer computations on-chip
- Off-chip embedding layers and caption buffer

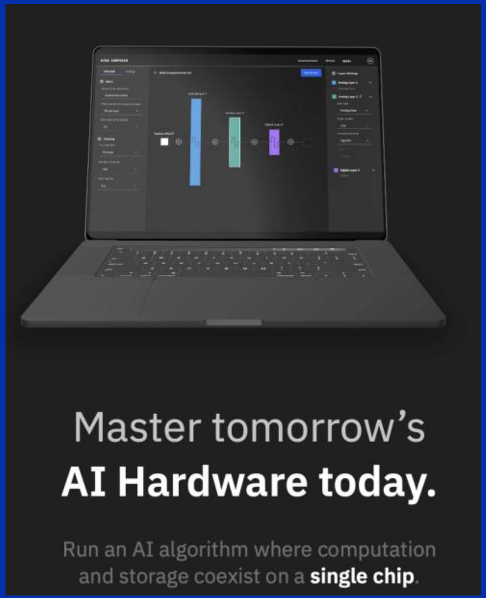
Hermes implementation



Impl.	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Software	0.534	0.340	0.206	0.135
Hermes	0.544	0.346	0.206	0.134

BLEU metric matches the n-grams of the produced caption with reference ones. Number between 0 and 1 (higher is better).

IBM Analog AI Hardware Composer



Master tomorrow's
AI Hardware today.

Run an AI algorithm where computation and storage coexist on a **single chip**.

https://aihw-composer.draco.res.ibm.com/

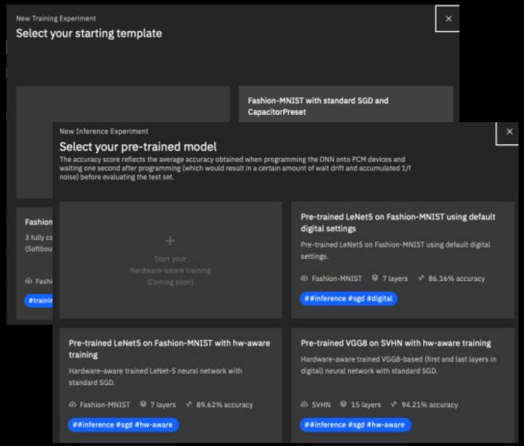
Interactive No-code Cloud Experience

- Cloud-based simulations
- Pre-configured analog device presets
- Show-case new algorithmic advancements
- Seamless interaction with GitHub open-source AIHWKIT

Roadmap

- Analog hardware access
- Backend-integration with AiMOS
- More neural network templates
- BYO Model and datasets
- Analog materials device builder
- Performance models

Templates for inference and training experiments



New Training Experiment

Select your starting template

Fashion-MNIST with standard SGD and CapacitorPreset

New Inference Experiment

Select your pre-trained model

The accuracy score reflects the average accuracy obtained when programming the DNN onto PCM devices and waiting one second after programming (which would result in a certain amount of wait drift and accumulated 1/f noise) before evaluating the test set.

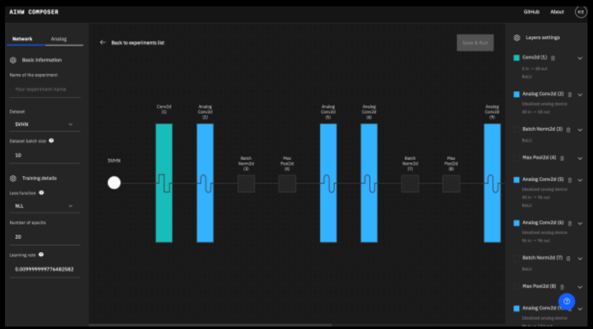
+ Start your Hardware-aware training (Coming soon)

Pre-trained LeNet5 on Fashion-MNIST using default digital settings
Pre-trained LeNet5 on Fashion-MNIST using default digital settings.
Fashion-MNIST · 7 layers · 86.16% accuracy
##inference #sgd #digital

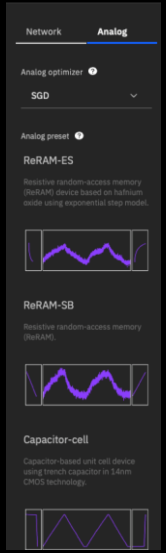
Pre-trained LeNet5 on Fashion-MNIST with hw-aware training
Hardware-aware trained LeNet-5 neural network with standard SGD.
Fashion-MNIST · 7 layers · 89.62% accuracy
##inference #sgd #hw-aware

Pre-trained VGG8 on SVHN with hw-aware training
Hardware-aware trained VGG8-based (first and last layers in digital) neural network with standard SGD.
SVHN · 15 layers · 94.21% accuracy
##inference #sgd #hw-aware

Compose your neural network and select various analog and neural network parameters



Select from a wide range of pre-configured device presets



Network | Analog

Analog optimizer: SGD

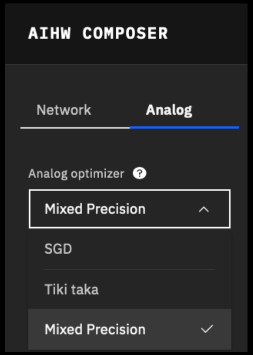
Analog preset

ReRAM-ES
Resistive random-access memory (ReRAM) device based on hafnium oxide using exponential step model.

ReRAM-SB
Resistive random-access memory (ReRAM).

Capacitor-cell
Capacitor-based unit cell device using trench capacitor in 14nm CMOS technology.

Analog-friendly optimizers



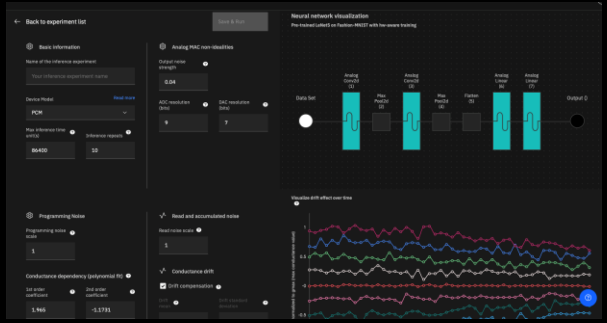
AIHW COMPOSER

Network | Analog

Analog optimizer: Mixed Precision

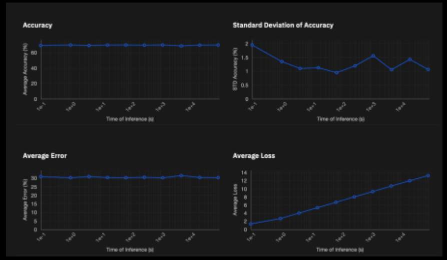
- SGD
- Tiki taka
- Mixed Precision ✓

Configure various noise models for inference simulation



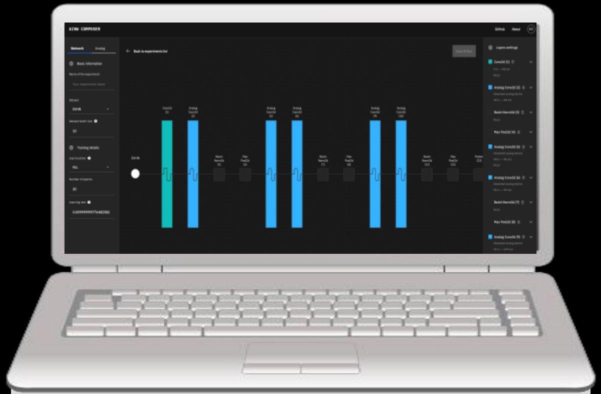
Back to experiment list
Basic information
Analog MAC non-idealities
Neural network visualization
Data Set
Output 0
Programming Noise
Read and accumulated noise
Conductance dependency (polynomial fit)
Conductance drift
Drift compensation

Visualize accuracy results



Manuel Le Gallo, IBM Research - Europe

IBM Analog Cloud Composer



Interactive Cloud Composer

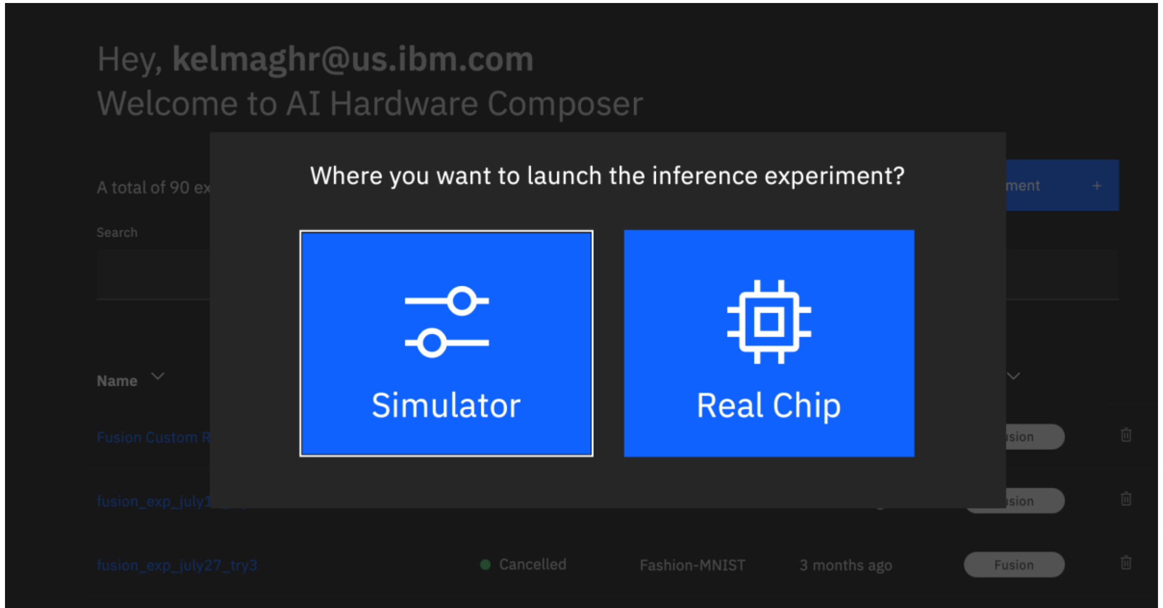
No code experience. Explore training with analog and neural networks

https://aihw-composer.draco.res.ibm.com

Released in January 2024

First of a Kind Analog Chip Access

A brand-new Interactive Experience for Inference on Real Analog AI Hardware



Hey, **kelmaghr@us.ibm.com**
Welcome to AI Hardware Composer

A total of 90 ex

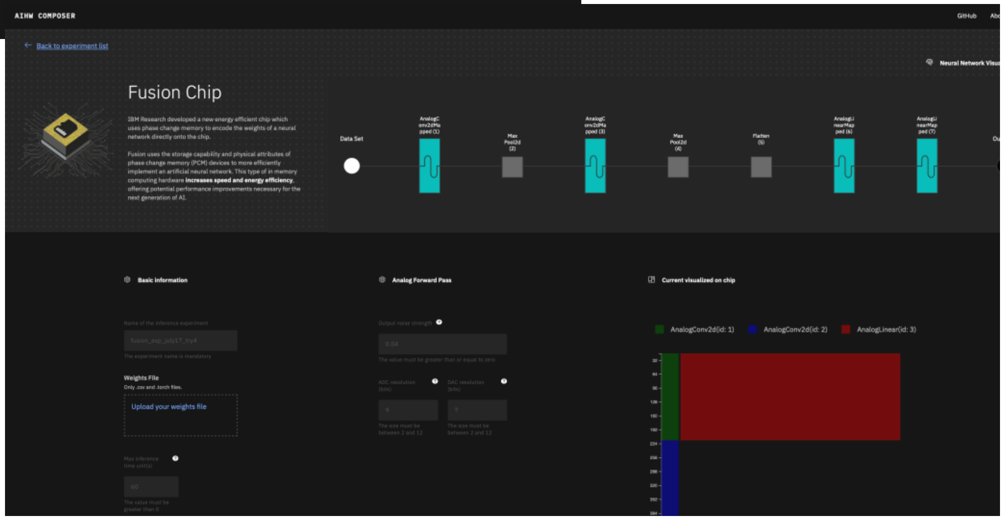
Search

Where you want to launch the inference experiment?

fusion_exp_july27_try3	Cancelled	Fashion-MNIST	3 months ago	Fusion
fusion_exp_july27_try2	Cancelled			

Free access to 1T1R chip with **1M PCM cells**

[Live Demo]

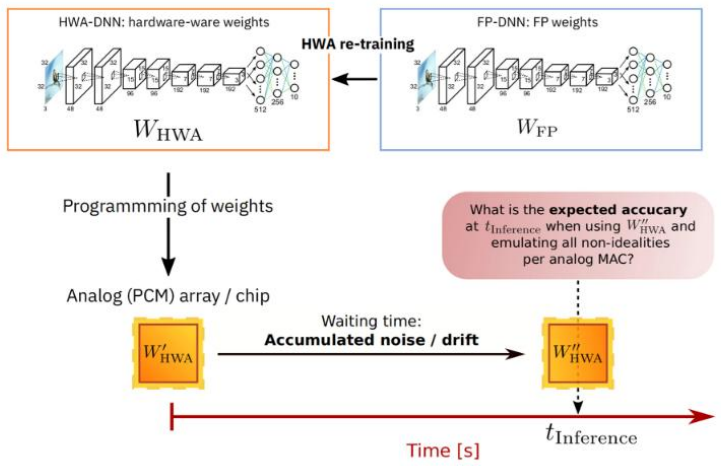


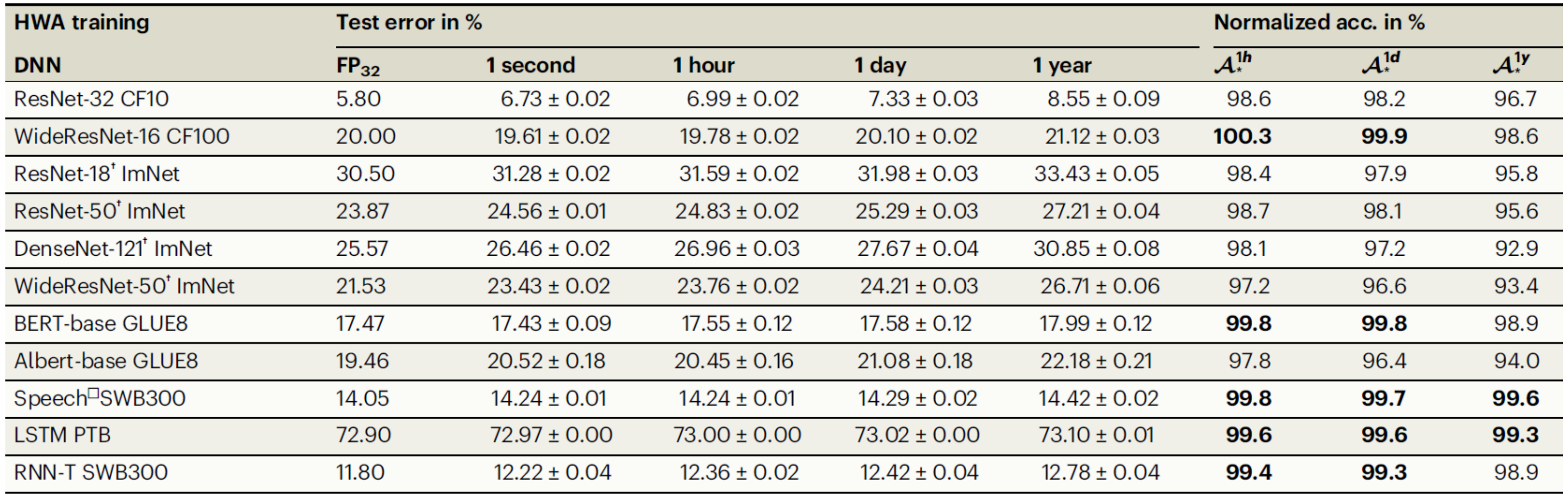
Fusion Chip

IBM Research developed a new energy efficient chip which uses phase change memory to encode the weights of a neural network directly onto the chip.

Fusion uses the storage capability and physical attributes of phase change memory (PCM) devices to more efficiently implement an artificial neural network. This type of in memory computing hardware **increases speed and energy efficiency**, offering potential performance improvements necessary for the next generation of AI.

Software: Custom training for DNN inference


HWA-DNN: hardware-ware weights
HWA re-training
FP-DNN: FP weights
W_{HWA}
W_{FP}
Programmming of weights
What is the **expected accucary** at $t_{\mathrm{Inference}}$ when using W''_{HWA} and emulating all non-idealities per analog MAC?
Analog (PCM) array / chip
Waiting time:
Accumulated noise / drift
W'_{HWA}
W''_{HWA}
$t_{\mathrm{Inference}}$
Time [s]



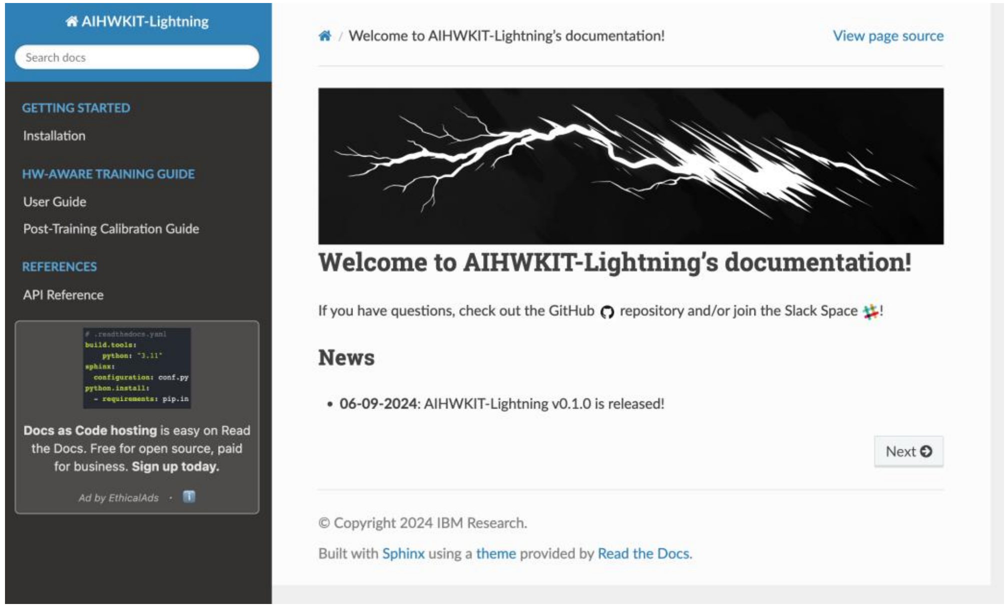
HWA training	**Test error in %**					**Normalized acc. in %**		
DNN	**FP$_{32}$**	**1 second**	**1 hour**	**1 day**	**1 year**	$\mathcal{A}_*^{1h}$	$\mathcal{A}_*^{1d}$	$\mathcal{A}_*^{1y}$
ResNet-32 CF10	5.80	6.73 ± 0.02	6.99 ± 0.02	7.33 ± 0.03	8.55 ± 0.09	98.6	98.2	96.7
WideResNet-16 CF100	20.00	19.61 ± 0.02	19.78 ± 0.02	20.10 ± 0.02	21.12 ± 0.03	**100.3**	**99.9**	98.6
ResNet-18† ImNet	30.50	31.28 ± 0.02	31.59 ± 0.02	31.98 ± 0.03	33.43 ± 0.05	98.4	97.9	95.8
ResNet-50† ImNet	23.87	24.56 ± 0.01	24.83 ± 0.02	25.29 ± 0.03	27.21 ± 0.04	98.7	98.1	95.6
DenseNet-121† ImNet	25.57	26.46 ± 0.02	26.96 ± 0.03	27.67 ± 0.04	30.85 ± 0.08	98.1	97.2	92.9
WideResNet-50† ImNet	21.53	23.43 ± 0.02	23.76 ± 0.02	24.21 ± 0.03	26.71 ± 0.06	97.2	96.6	93.4
BERT-base GLUE8	17.47	17.43 ± 0.09	17.55 ± 0.12	17.58 ± 0.12	17.99 ± 0.12	**99.8**	**99.8**	98.9
Albert-base GLUE8	19.46	20.52 ± 0.18	20.45 ± 0.16	21.08 ± 0.18	22.18 ± 0.21	97.8	96.4	94.0
Speech□SWB300	14.05	14.24 ± 0.01	14.24 ± 0.01	14.29 ± 0.02	14.42 ± 0.02	**99.8**	**99.7**	**99.6**
LSTM PTB	72.90	72.97 ± 0.00	73.00 ± 0.00	73.02 ± 0.00	73.10 ± 0.01	**99.6**	**99.6**	**99.3**
RNN-T SWB300	11.80	12.22 ± 0.04	12.36 ± 0.02	12.42 ± 0.04	12.78 ± 0.04	**99.4**	**99.3**	98.9

Joshi et al., Nature Comm. (2020); Rasch et al., Nature Comm. (2023)

- A custom “additive noise training” procedure required to deal with the lower-precision MVM operations
- The key idea is to inject noise to the synaptic weights in proportion to the synaptic weight noise during the forward pass of training
- Many larger-scale deep neural networks can be successfully retrained to show iso-accuracy with the floating-point implementation

Software: AIHWKIT-Lightning ⚡

- New lightweight library as a faster alternative to AIHWKIT with less features: https://github.com/IBM/aihwkit-lightning
- Aimed for training very large networks (e.g. LLMs)
- Can train Phi-3-Mini (3.8B parameters) on 1B tokens in under 6 hours using 96 V100 GPUs. 24h on 8 A100 GPUs.



AIHWKIT-Lightning

Search docs

GETTING STARTED

Installation

HW-AWARE TRAINING GUIDE

User Guide

Post-Training Calibration Guide

REFERENCES

API Reference

```
# .readthedocs.yaml
build.tools:
    python: "3.11"
sphinx:
  configuration: conf.py
python.install:
  - requirements: pip.in
```

Docs as Code hosting is easy on Read the Docs. Free for open source, paid for business. **Sign up today.**

Ad by EthicalAds ·

/ Welcome to AIHWKIT-Lightning's documentation!

View page source

Welcome to AIHWKIT-Lightning's documentation!

If you have questions, check out the GitHub repository and/or join the Slack Space !

News

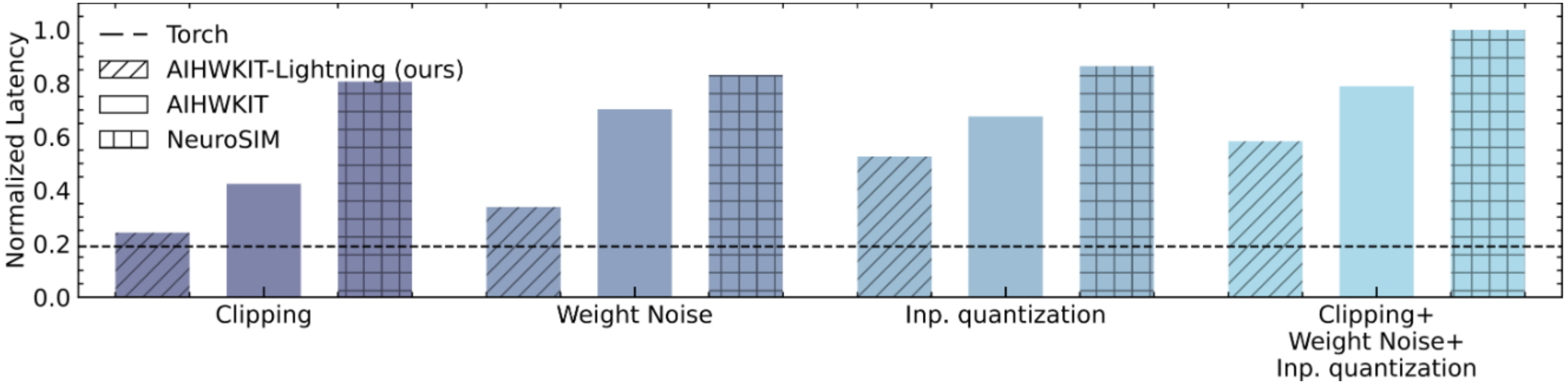
- **06-09-2024:** AIHWKIT-Lightning v0.1.0 is released!

Next

© Copyright 2024 IBM Research.

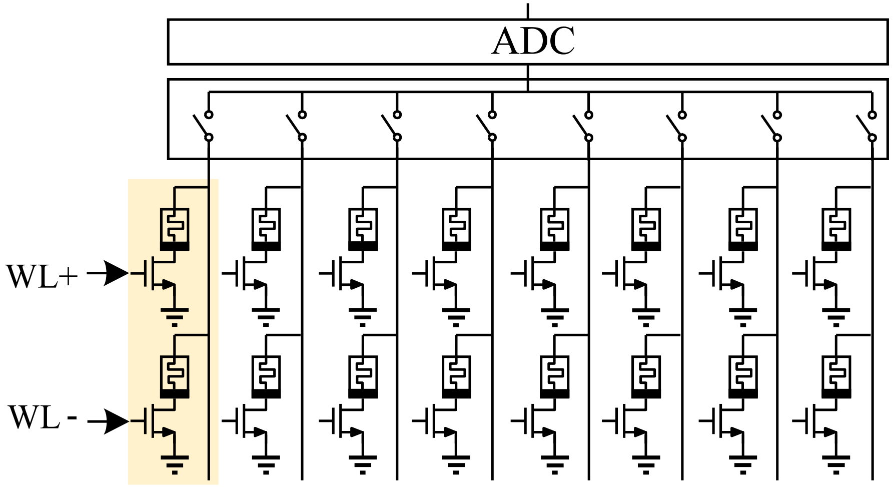
Built with Sphinx using a theme provided by Read the Docs.

~1.4x faster than AIHWKIT, ~2x faster than NeuroSIM

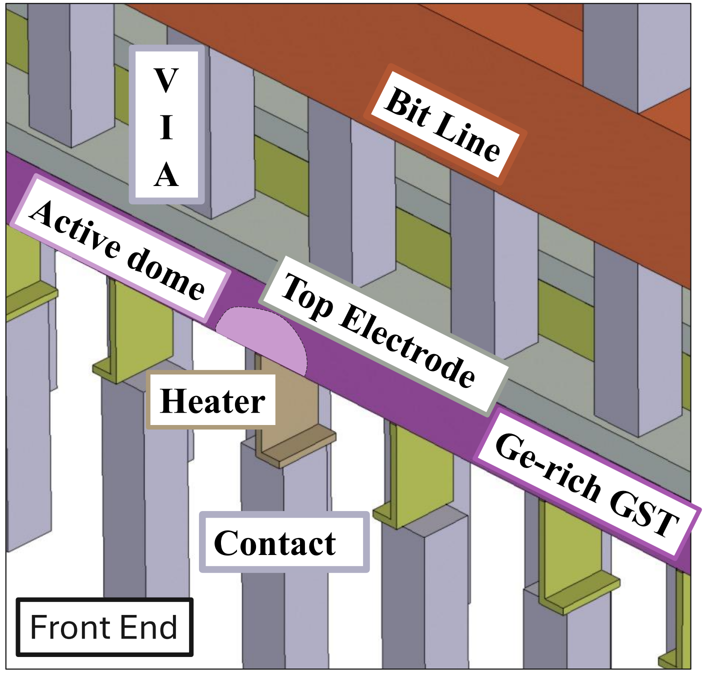


Buechel et al., NeurIPS MLNCP (2024)

AIMC Accelerator Node



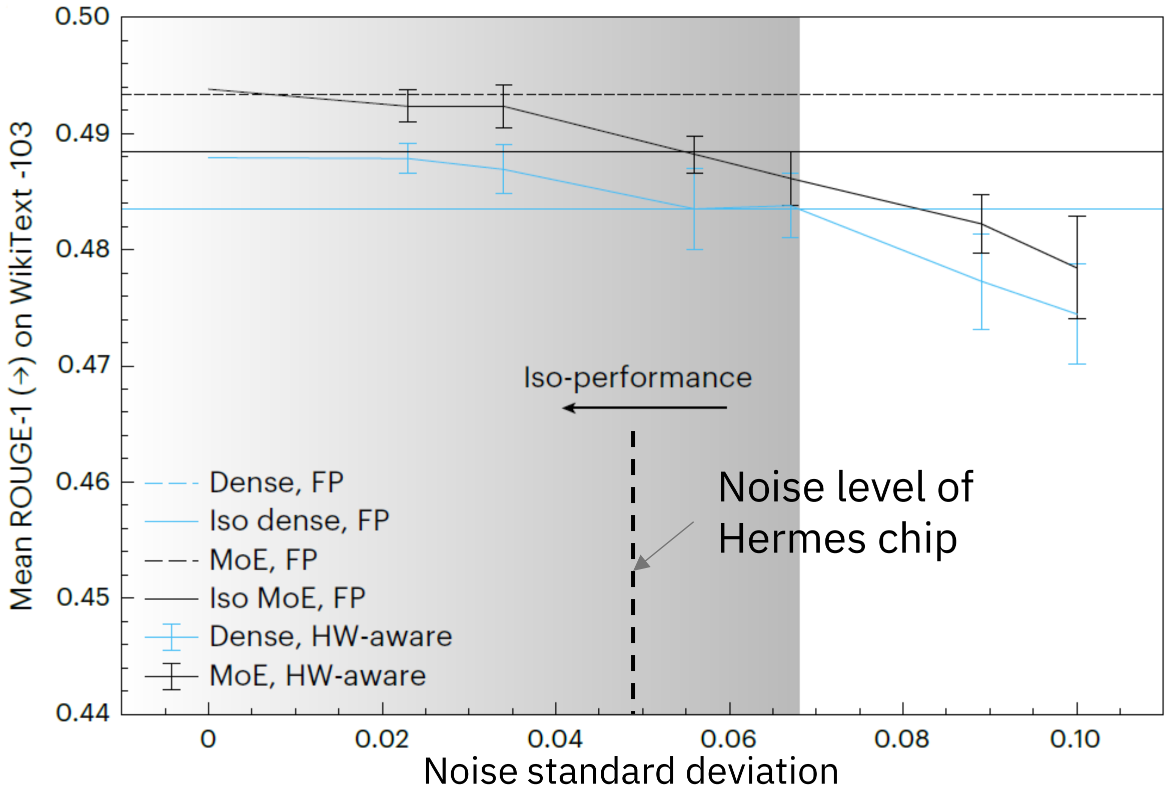
Max. capacity	Weights unsigned	Weights signed
Inputs unsigned	16 weight sets	8 weight sets
Inputs signed	16 weight sets	8 weight sets



Boybat et al., IEDM (2024)

- 512x512 array per AIMC node
- Potential for 8 different weight sets at lowest precision (>2M weights)
- Multiple PCM devices can be combined for higher compute precision
- Current-controlled oscillator-based ADCs

Mixture of Experts with 3D AIMC: Accuracy



Buechel et al., Nature Computational Science (2025)

- With hardware-aware training, we can get acceptable accuracies (within 99% of SW baseline) with an MoE for language modeling on WikiText-103 at noise levels comparable to Hermes chip