

RISC-V: Enabling Open Physical AI

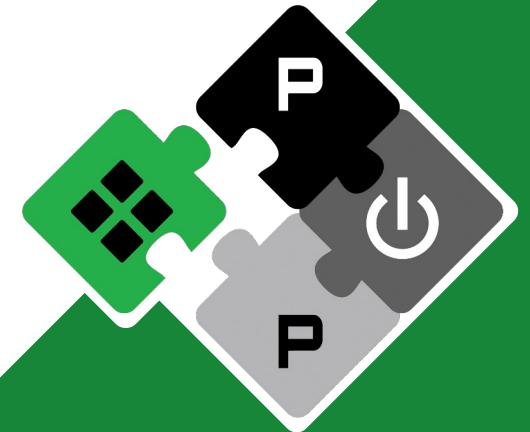
Luca Benini

lbenini@iis.ee.ethz.ch

luca.benini@unibo.it

PULP Platform

Open Source Hardware, the way it should be!



@pulp_platform 

pulp-platform.org 

youtube.com/pulp_platform 

Physical AI: Artificial Intelligence Everywhere



Humanoid



[Unitree26]

Interceptor Drone



[GeneralCherry26]

**Humanoid,
Interceptor Drone**
 $P_{avg} < 200 \text{ W}^*$

*[research.mobiusriskgroup.com/p/the-energy-diet-of-humanoid-robots]

Embodied (Physical) AI: Artificial Intelligence Everywhere



Nano-Drone



Smart Glasses

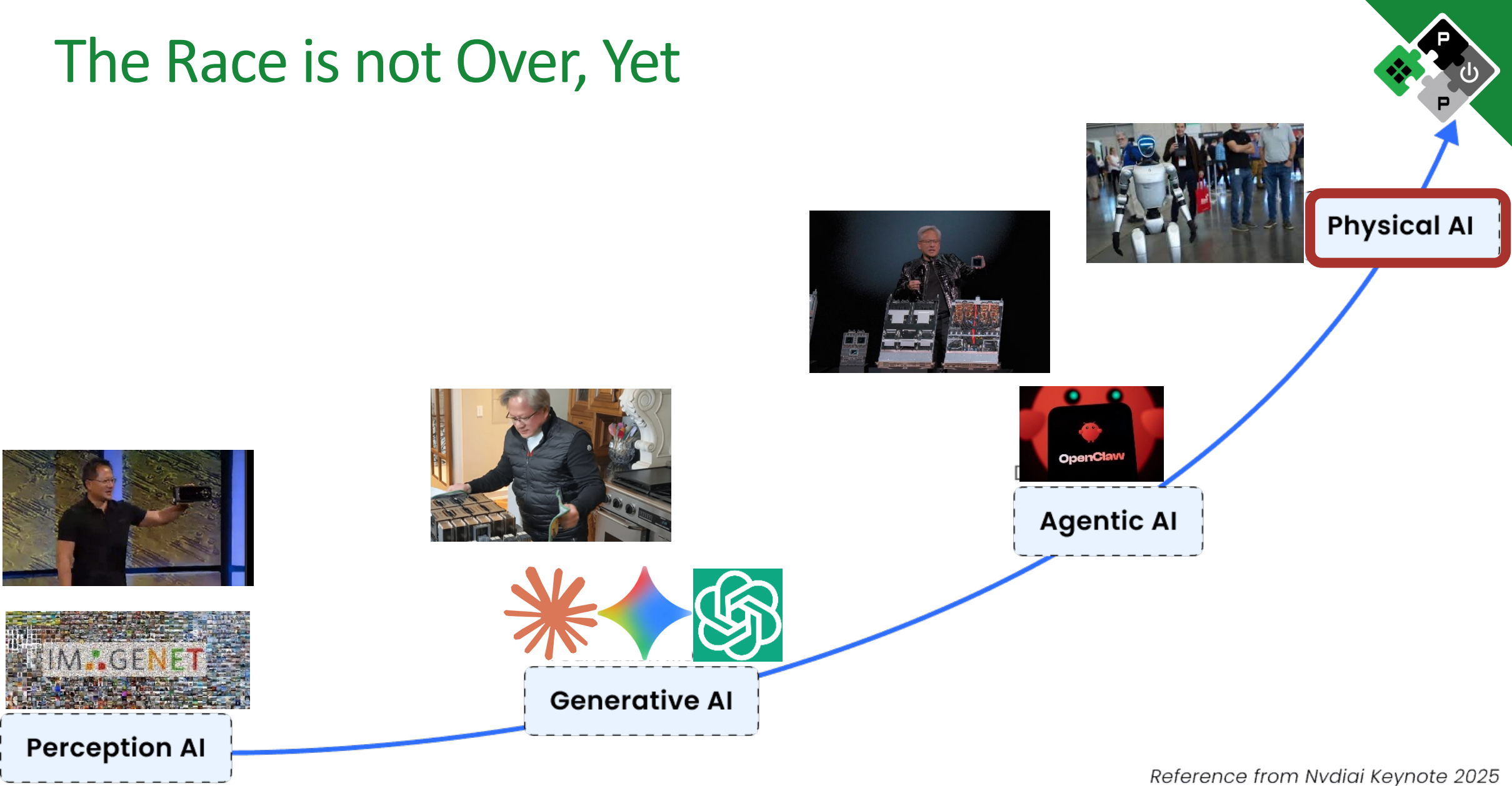


Nanodrone
 $P_{\text{avg}} < 150 \text{ mW}$

Smartglass
 $P_{\text{avg}} < 1.50 \text{ mW}$



The Race is not Over, Yet



Reference from Nvidiai Keynote 2025

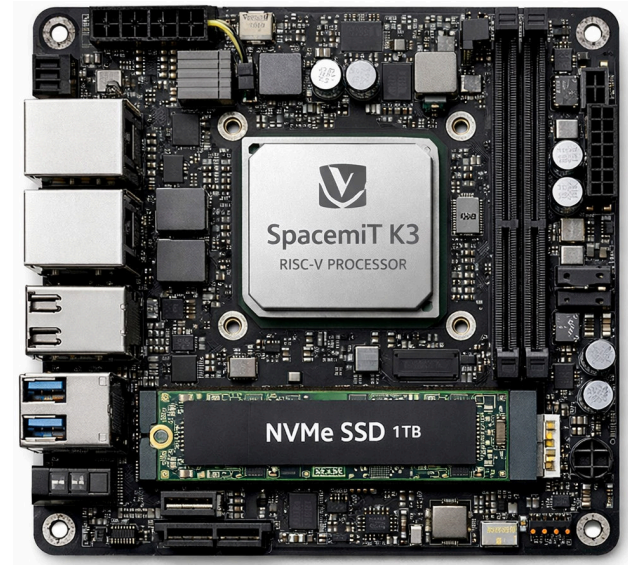
Innovation beyond “NVIDIA Gravity” is Challenging!

It's the software → **flexibility** key for fast evolution!

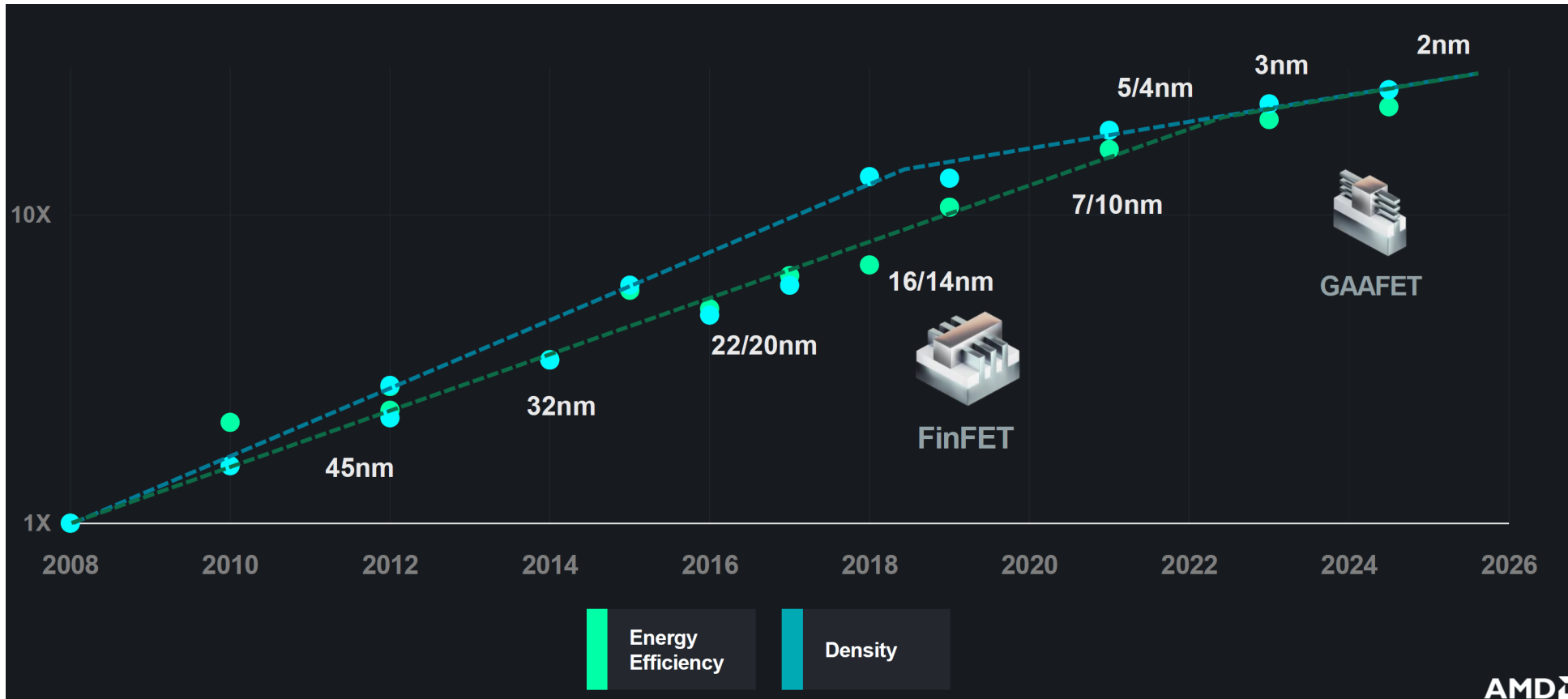
Need an **open standard** to counter a monopoly



RISC-V: The Free and Open RISC
Instruction Set Architecture



Efficiency Challenge



Model complexity
10x every **~2.5** years

Moore's Law
10x every **12** years!

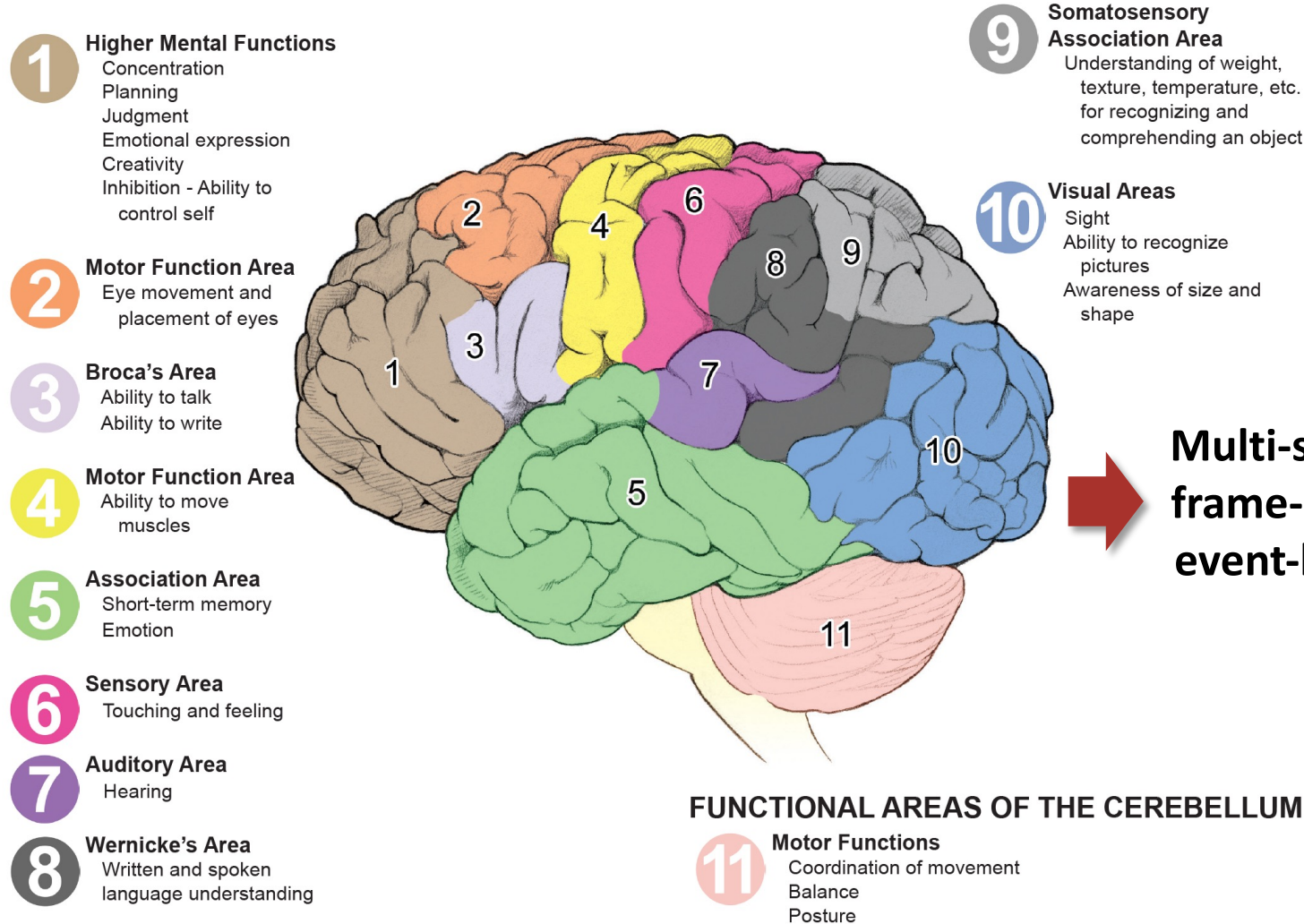


[AMD HotChips24]

Algorithm, Architecture, Design are key!

Efficiency via Heterogeneity: Multi-Domain Specialization

Brain-inspired: Multiple areas, different structure different function!



Multi-sensor
frame-based
event-based

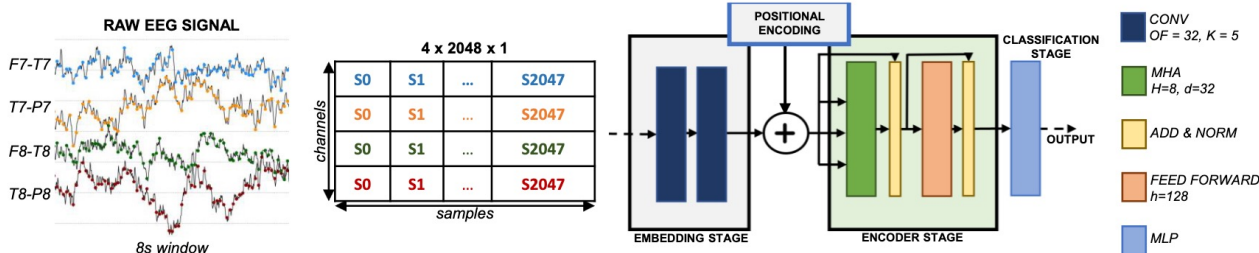
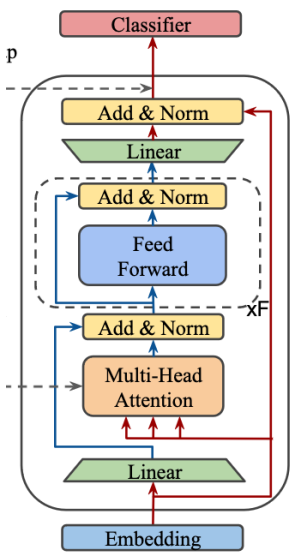
Perception

Multi-modal
Agentic
reasoning

A Fast-Evolving Model Zoo



[Z. Sun et al.] MobileBERT
Encoder Transformer



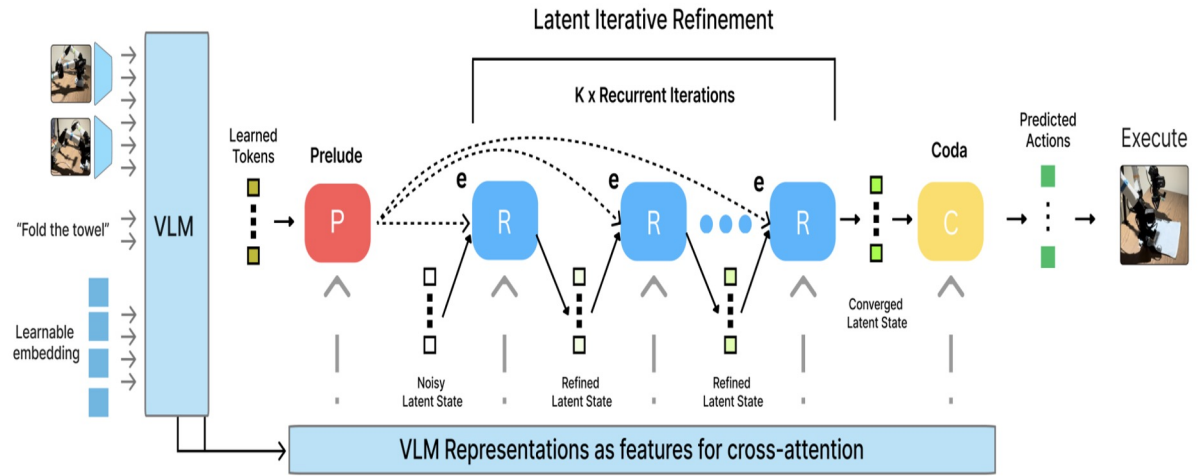
[P. Busia et al.] EEGFormer
Encoder Transformer



DINOv2: Learning Robust Visual Features without Supervision
[M. Oquab et al.]
Encoder



Auto-regressive



Recurrent

Domain-Specific ISA Extensions



Instruction set: open and extensible by construction (great!)

8-bit Convolution

Vanilla

N

```
addi a0,a0,1
addi t1,t1,1
addi t3,t3,1
addi t4,t4,1
lbu  a7,-1(a0)
lbu  a6,-1(t4)
lbu  a5,-1(t3)
lbu  t5,-1(t1)
mul  s1,a7,a6
mul  a7,a7,a5
add  s0,s0,s1
mul  a6,a6,t5
add  t0,t0,a7
mul  a5,a5,t5
add  t2,t2,a6
add  t6,t6,a5
bne  s5,a0,1c000bc
```

RISC-V
core

Specialized for AI → Mixed precision SIMD (16-2bit)

N/4

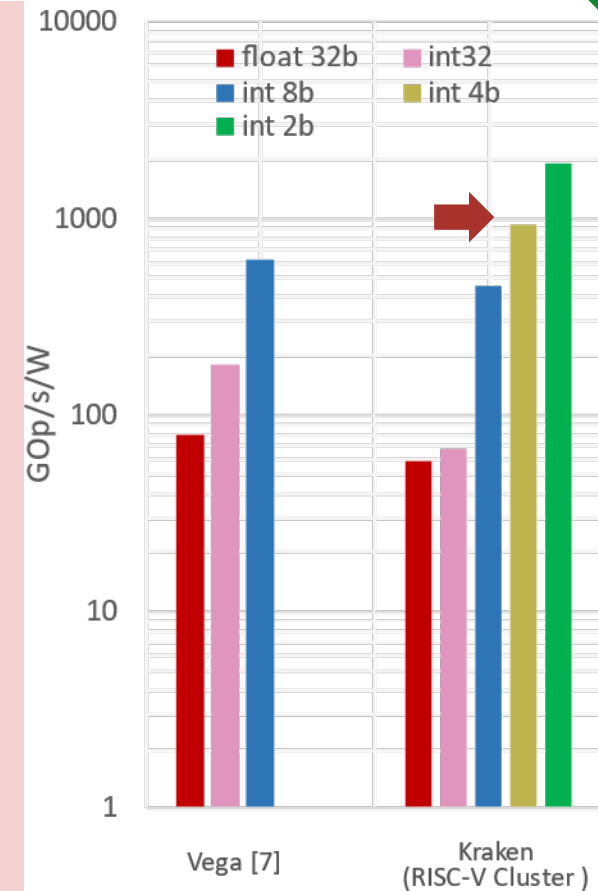
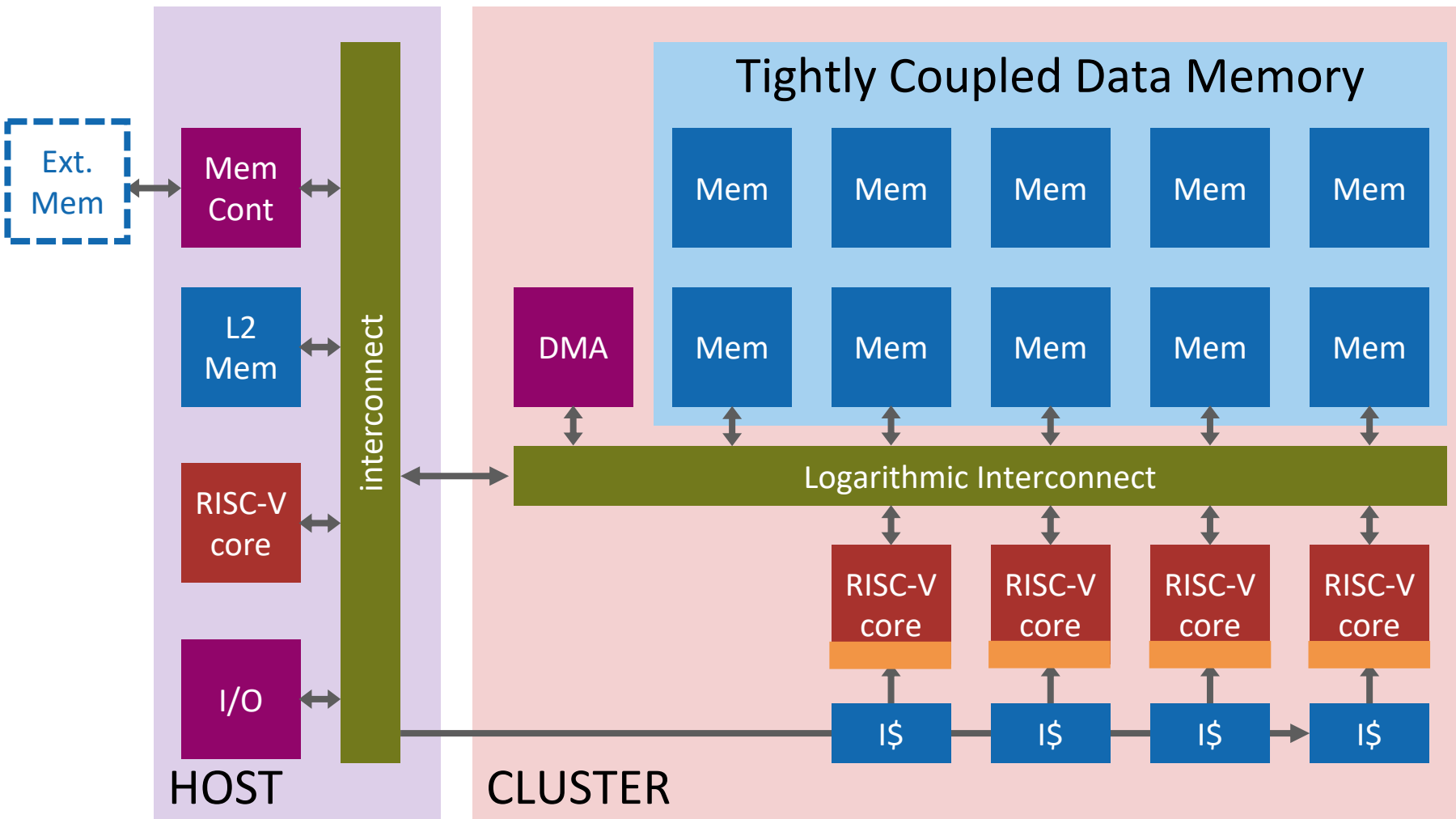
```
Init NN-RF (outside of the loop)
lp.setup
pv.nnsdotup.h s0,ax1,9
pv.nnsdotsp.b s1,aw2,0
pv.nnsdotsp.b s2,aw4,2
pv.nnsdotsp.b s3,aw3,4
pv.nnsdotsp.b s4,ax1,14
end
```

RISC-V
core

15x less instructions than Vanilla
90%+ ALU Utilization

Specialization Cost: Power, Area: 1.5x↑ Time 15x↓ → E = PT 10x ↓

A Cluster of Domain Specific Cores



1TOP/s/W
2b/4b OPS

Heterogeneous, Multiscale Domain-Specific Architecture



Multiple Scales of Specialization

Extensions to processor cores

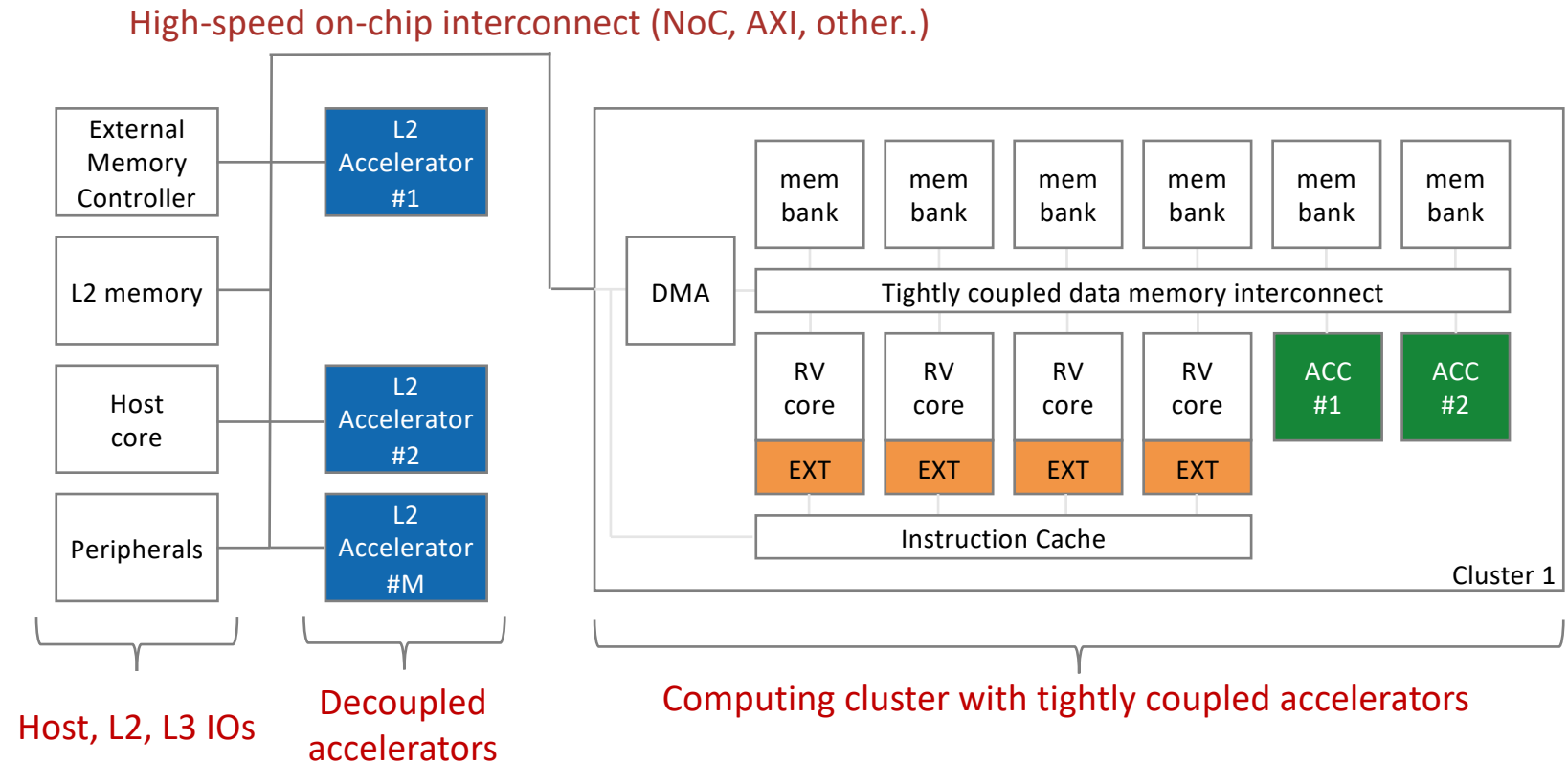
- Tightly-coupled, fine sychro
- Shared L0

Tightly coupled Accelerators

- Fast offload, coarse sychro
- Shared L1

Decoupled Accelerators

- Decoupled
- Shared L2 (or higher)



RISC-V is a key enabler → max agility, enabling SW build-up, without vendor lock-in

Specialization in Perspective



Using 22FDX tech, NT@0.6V, High utilization, minimal IO & overhead

Energy-Efficient RV Core $\rightarrow$ **20pJ (8bit)**



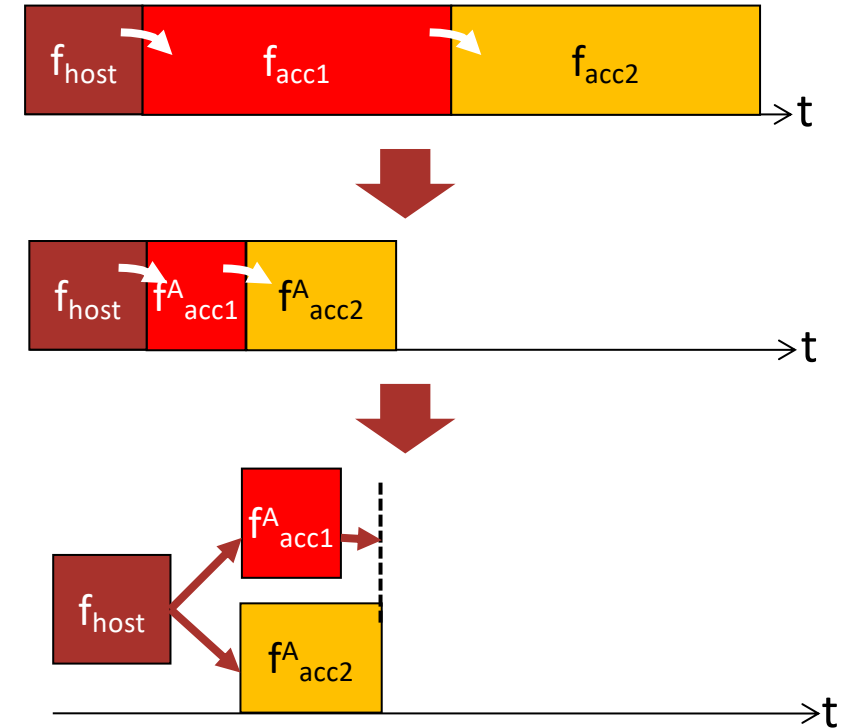
ISA-based 10-20x $\rightarrow$ **1pJ (4bit)**



Tightly coupled 10-20x $\rightarrow$ **100fJ (4bit)**

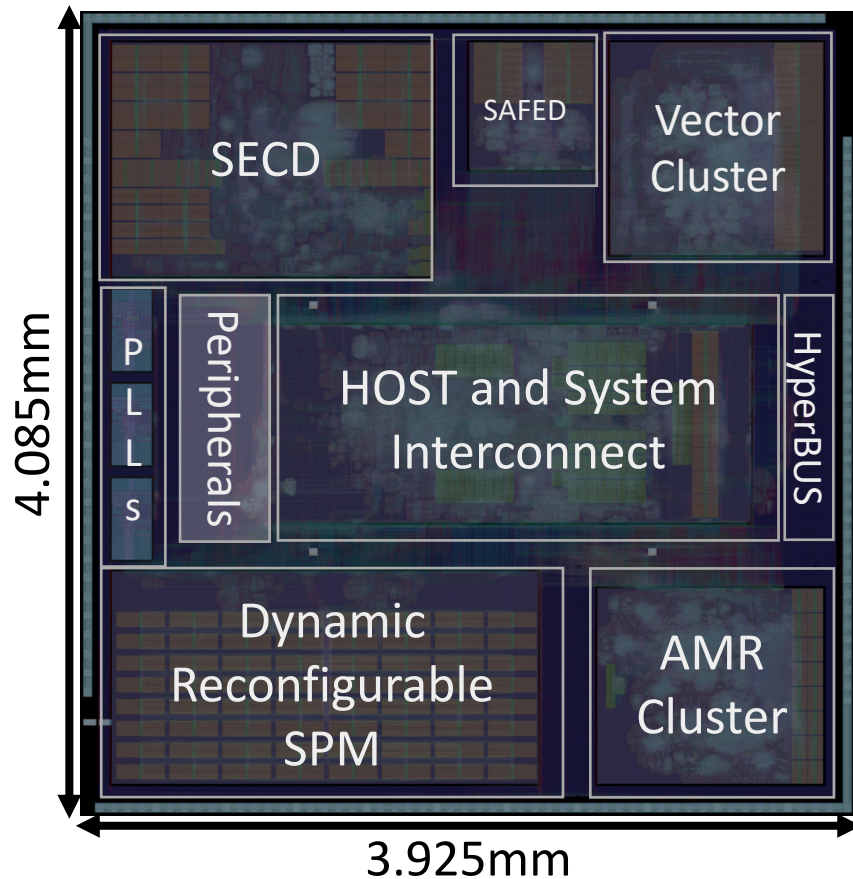


Decoupled 100x $\rightarrow$ **1fJ (ternary)**

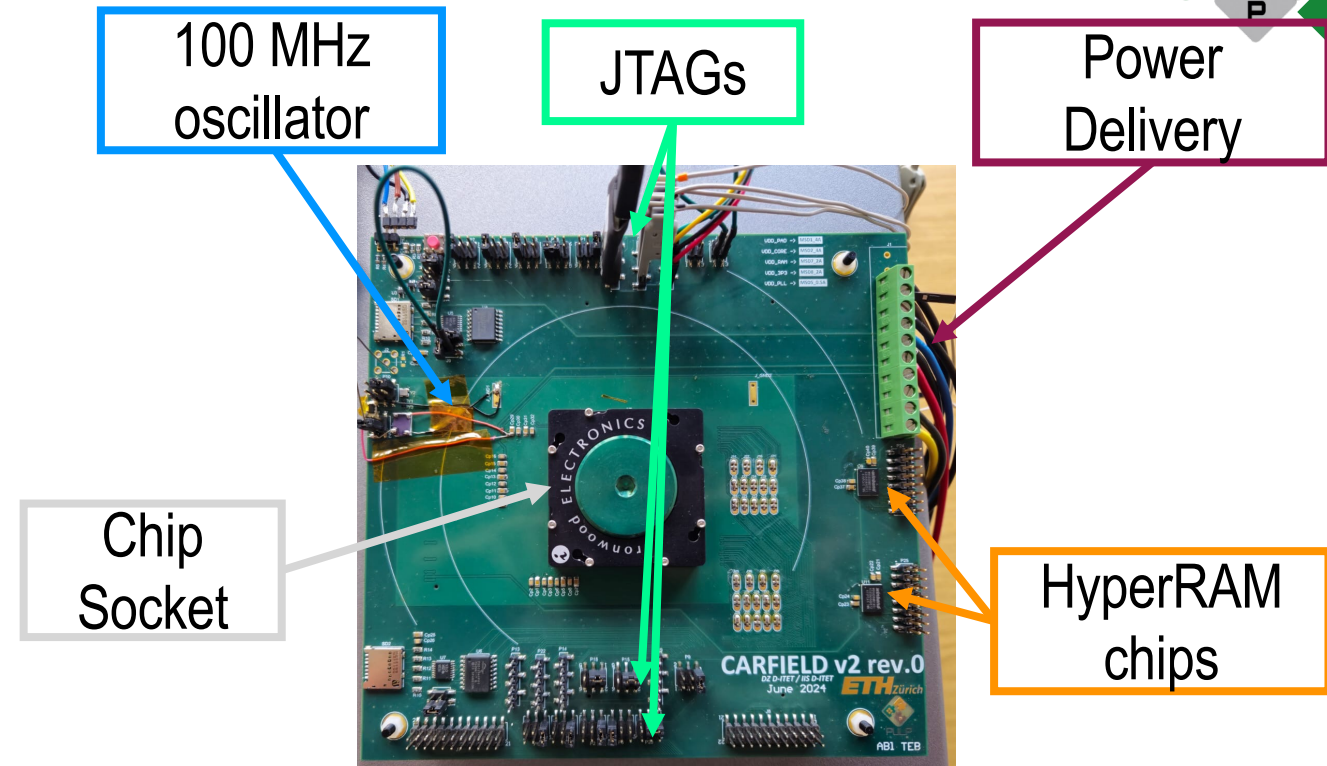


Specialization game: efficiency vs. utilization Tradeoff with Amdahl's Law

Carfield SoC in Intel16 FinFet



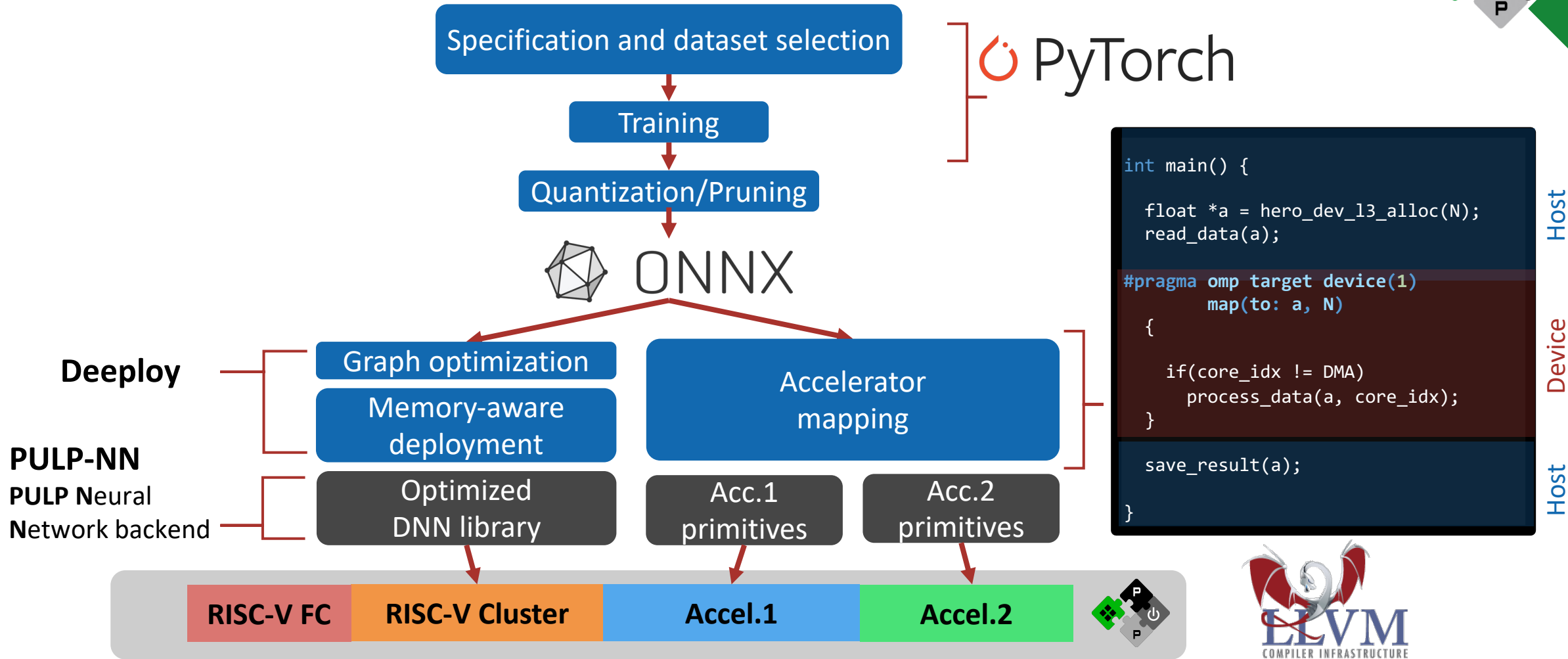
Up to 1 GHz, **1.2W** @ 0.8V



A. Garofalo et al., "A Reliable, Time-Predictable Heterogeneous SoC for AI-Enhanced Mixed-Criticality Edge Applications," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 72, no. 11, pp. 1625-1629, Nov. 2025

Efficiency + Safety, Predictability, Security. All are key for Physical AI SoCs

Let's not Forget Software....

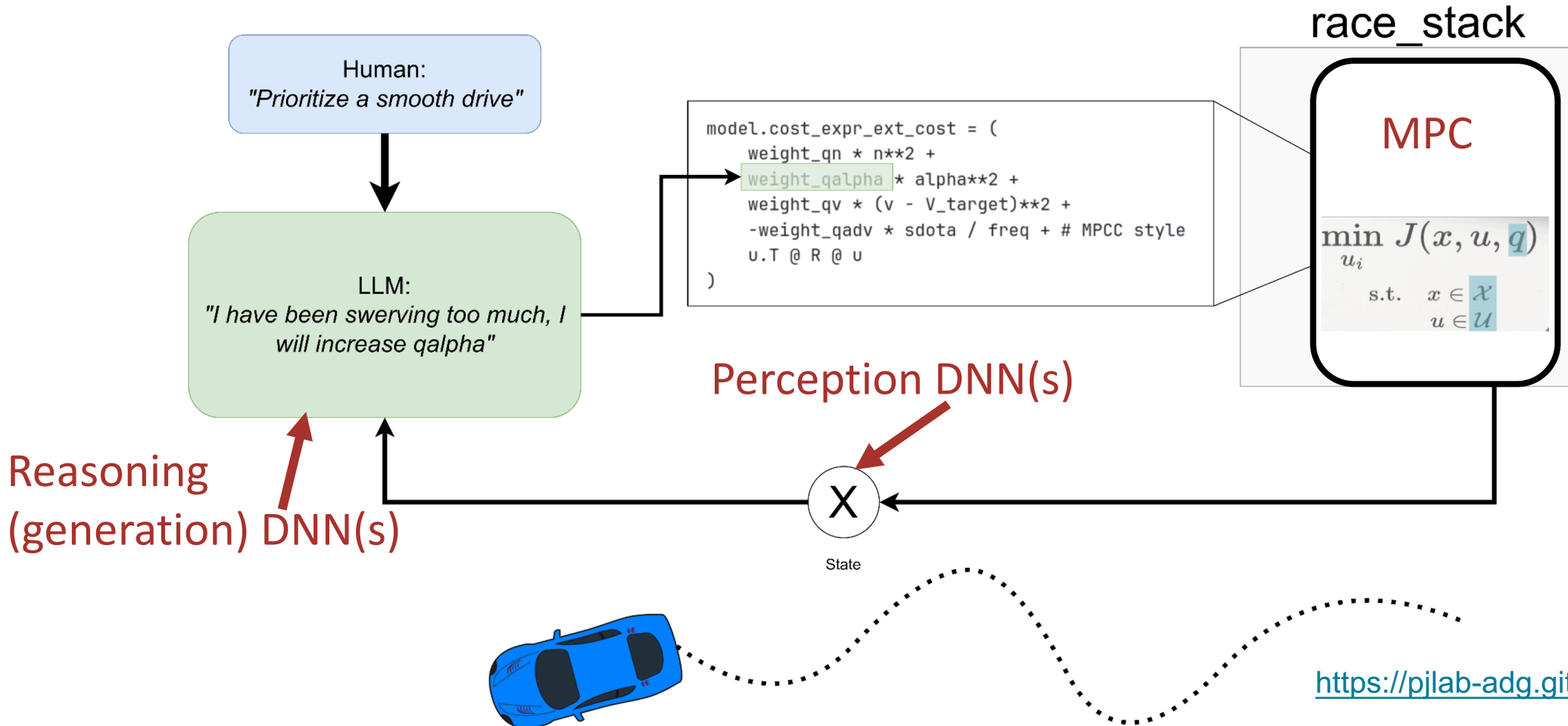


Open HW-SW platforms & standards are key for open innovation vs “AI Monopoly”

Beyond Perception and Fusion: Agentic Reasoning

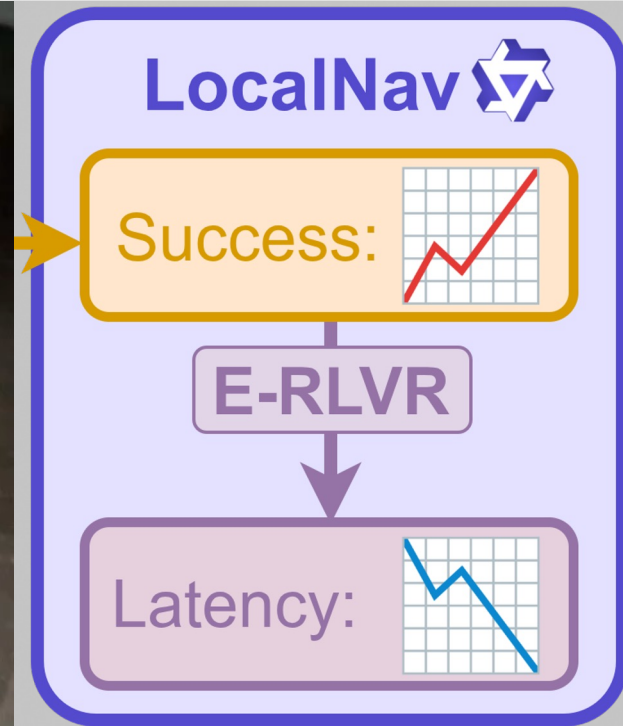


LLM Reasoning on Human Commands & Robot Observations



<https://pjlabor-adg.github.io/DiLu/>

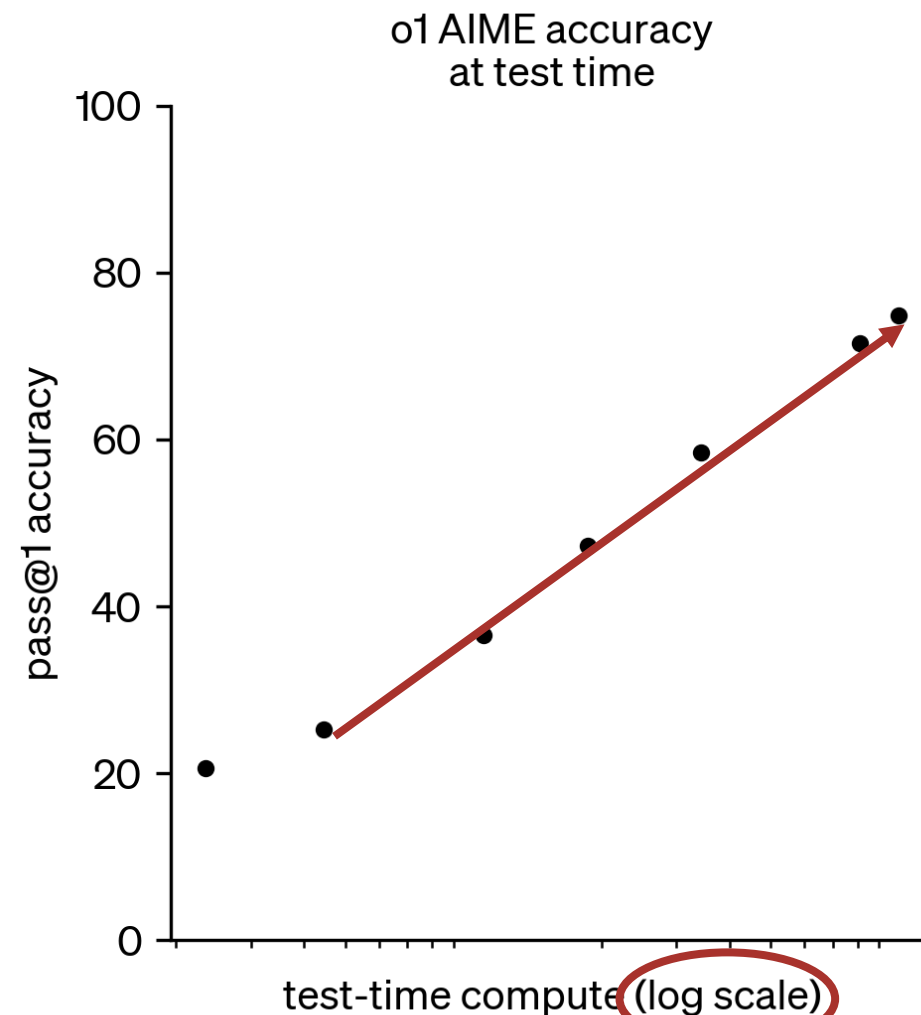
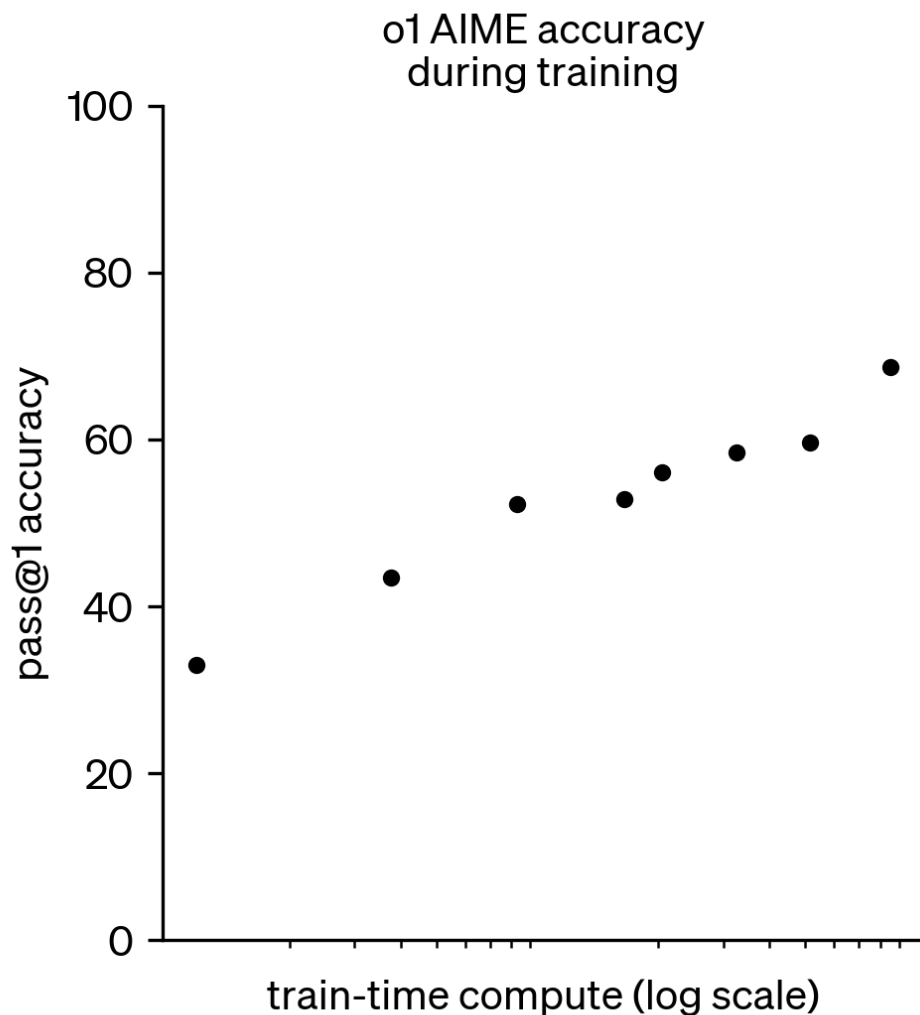
Beyond Perception and Fusion: Agentic Reasoning



12.1 % 1000 output tokens

1.16 % 1000 latency

Generative AI Scaling Laws



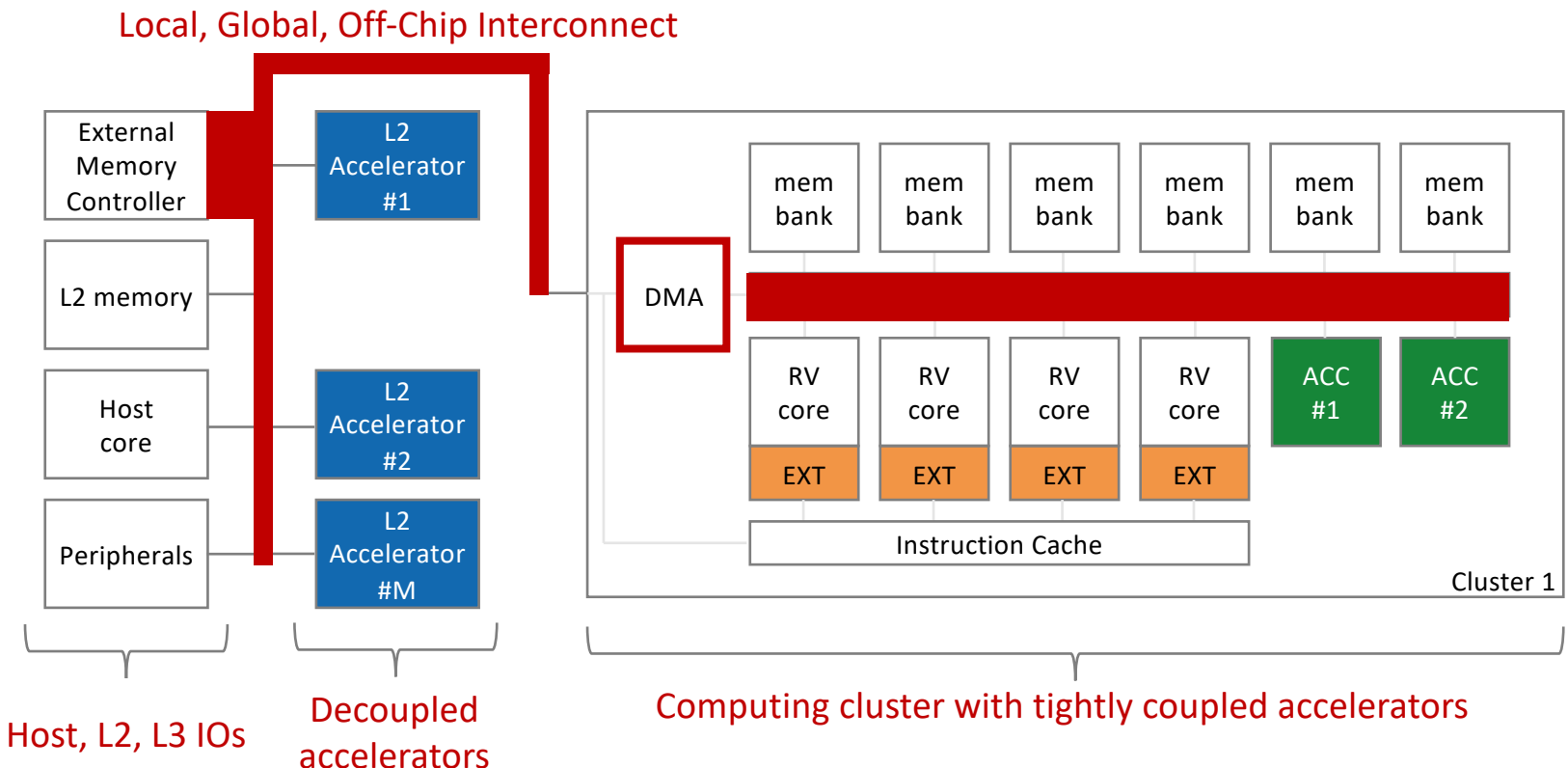
[openai.com/index/learning-to-reason-with-llms/]



There is no Other Way to Go, but UP

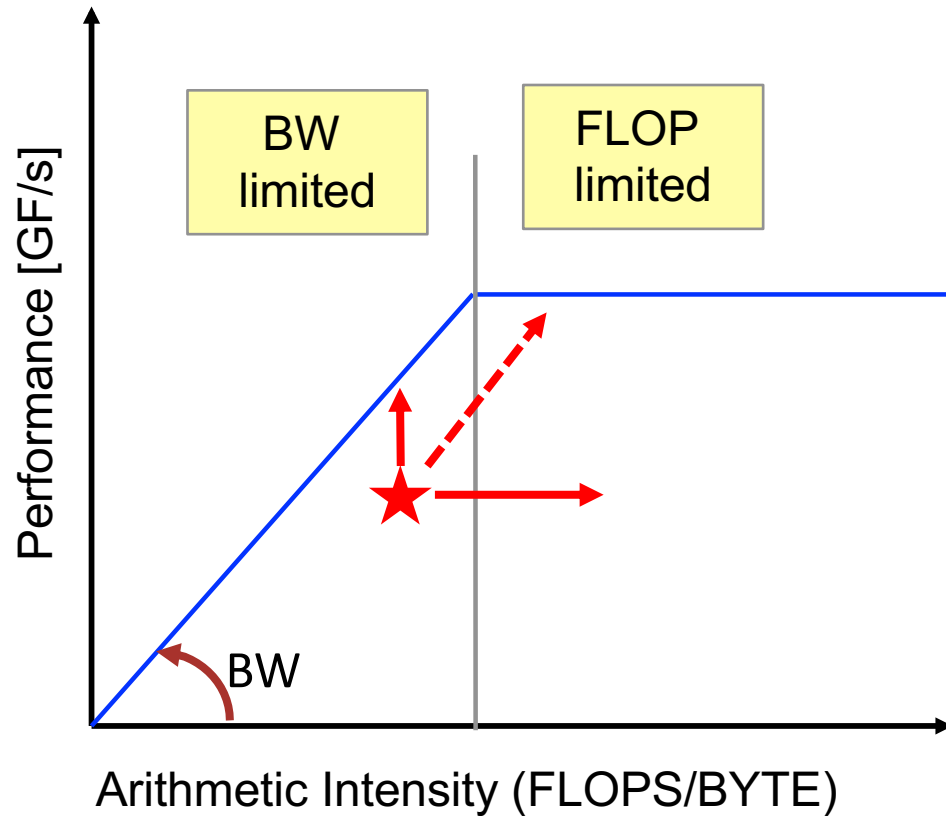
Multiple Scales of specialization

- Extensions to processor cores
- Tightly-coupled, fine sychro
 - Shared L0
- Tightly Coupled Accelerators
- Fast offload, coarse synchro
 - Shared L1
- Multiple Decoupled Accelerators
- Decoupled
 - Shared L2 (or higher)



Specialize interconnects too! Latency (and Bandwith) Challenge!

Patterson's Law (aka Roofline), Little's Law



Mean jobs in system = arrival rate x mean response time



BufferSize (B) = Bandwidth (B/s) x Latency (s)

Eg. 1KB/ns x 100 (ns) → 100KB

Need IO-bandwidth, on-chip Buffering, and latency tolerant Architecture

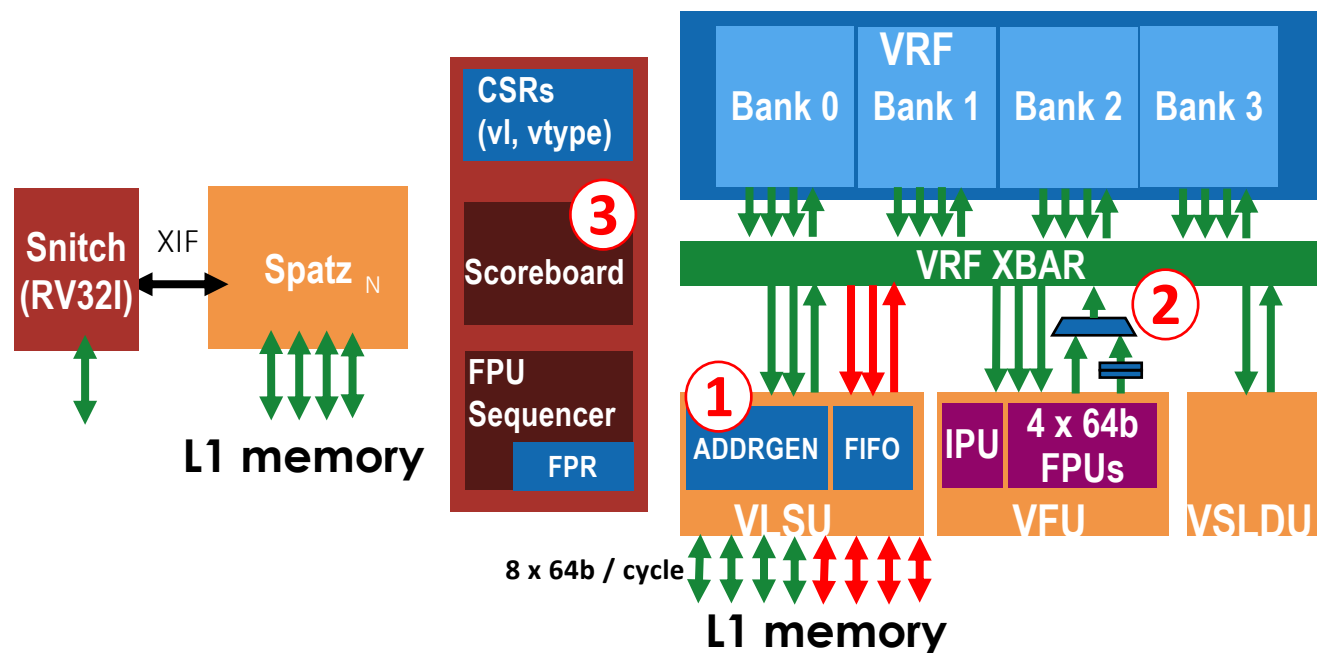


Spatz: RVV Unit for Snitch

Set of uA optimizations for VPEs

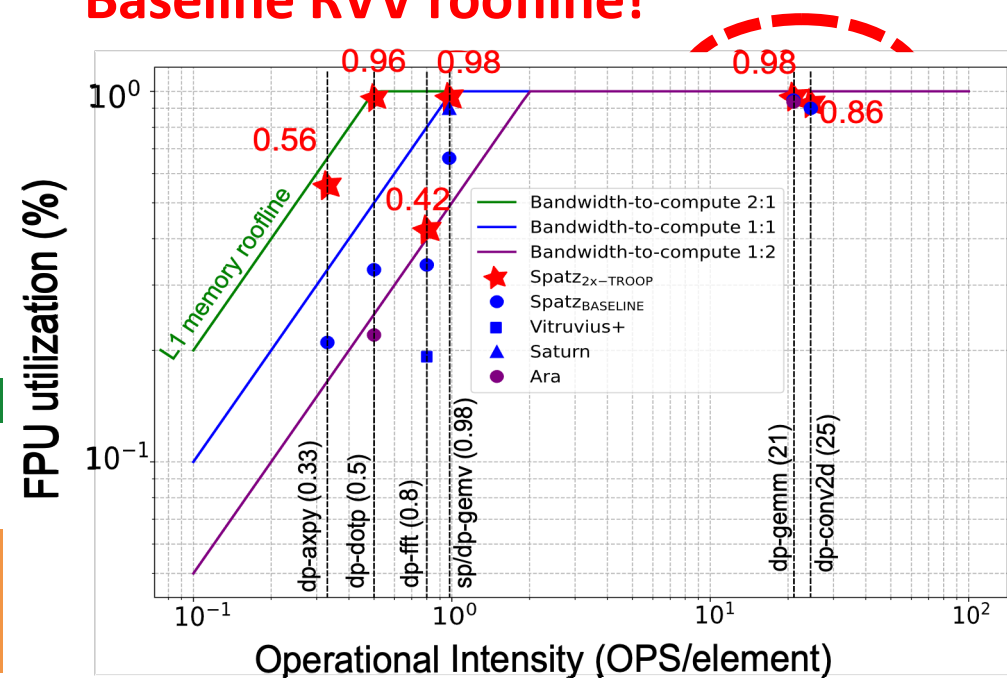
- Decoupled VLSU interfaces
- Dynamic Priority allocation and buffering
- Improved vector chaining

Fully utilize L1-memory BW + tolerate L1 latency (Troop)



Troop reaches L1-memory roofline!

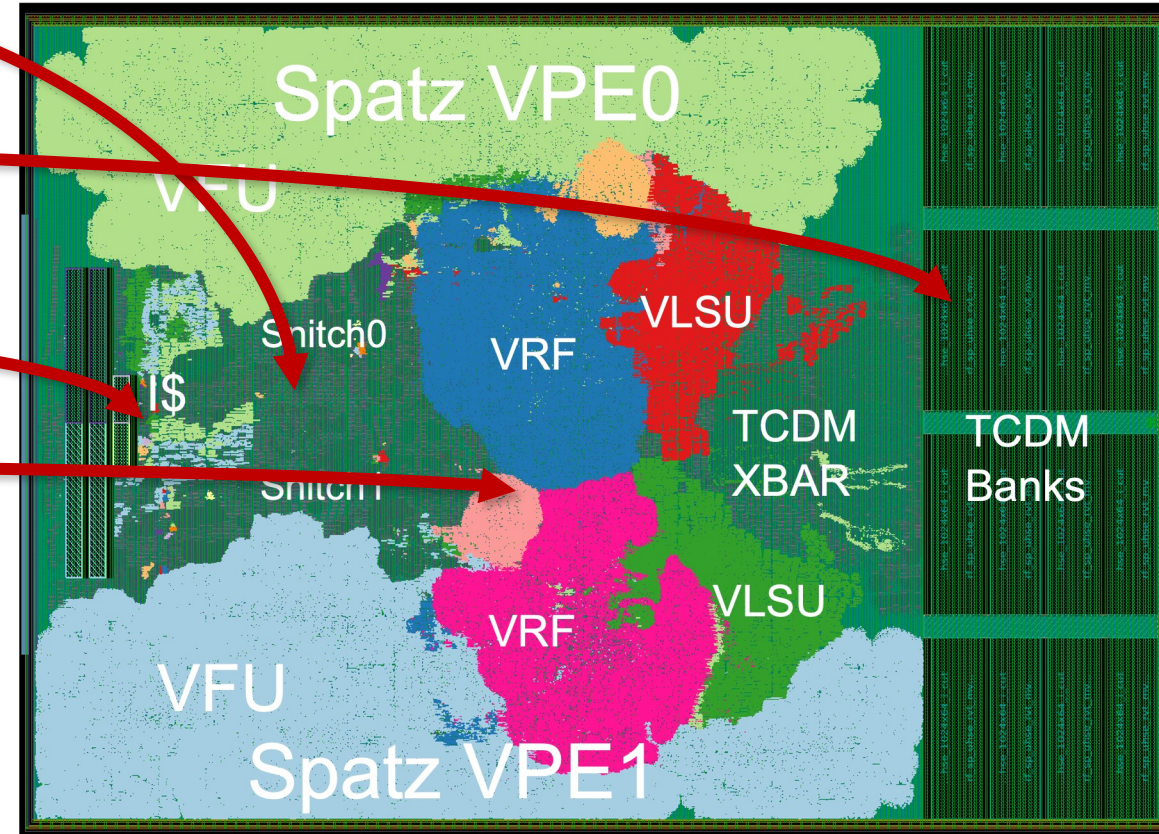
Baseline RVV roofline!



Snitch Cluster: The Fundamental Compute Block



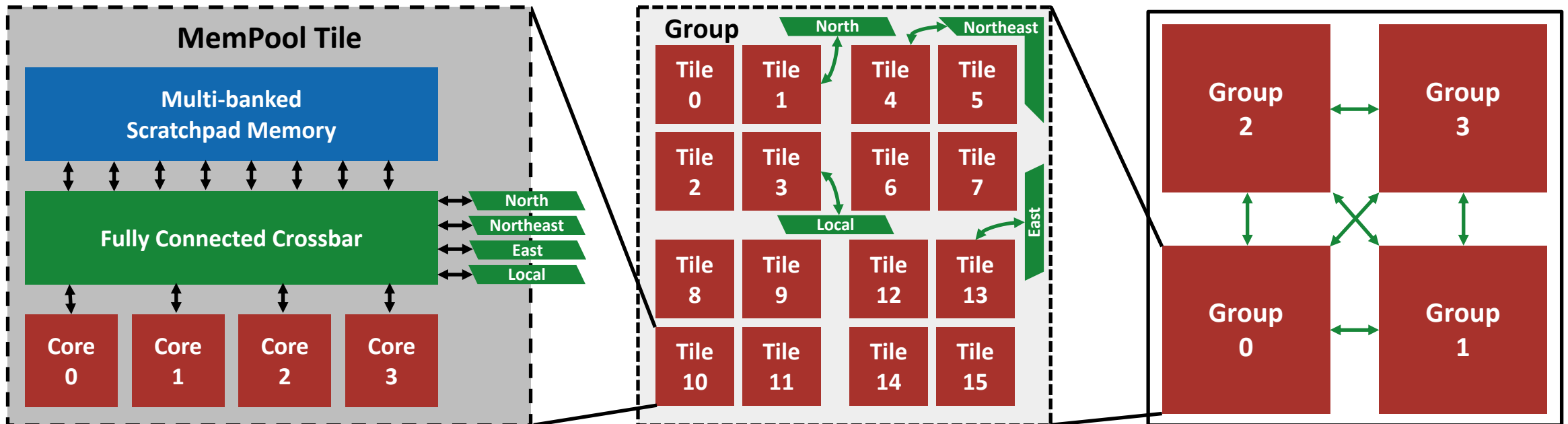
- **2 Snitch+Spatz compute cores**
 - 8 SIMD FPU
- **128 KiB Shared TCDM**
 - 32-bank, low-latency shared scratchpad
- **Shared I-cache and peripherals**
- **DMA engine**
 - 512b interface to interconnect
 - HW support for multi dim. transfers
- **Shared DMA (10% overhead) for latency tolerance of L2+→L1: 100s vs 10s cycles**



How Large should a Cluster be?



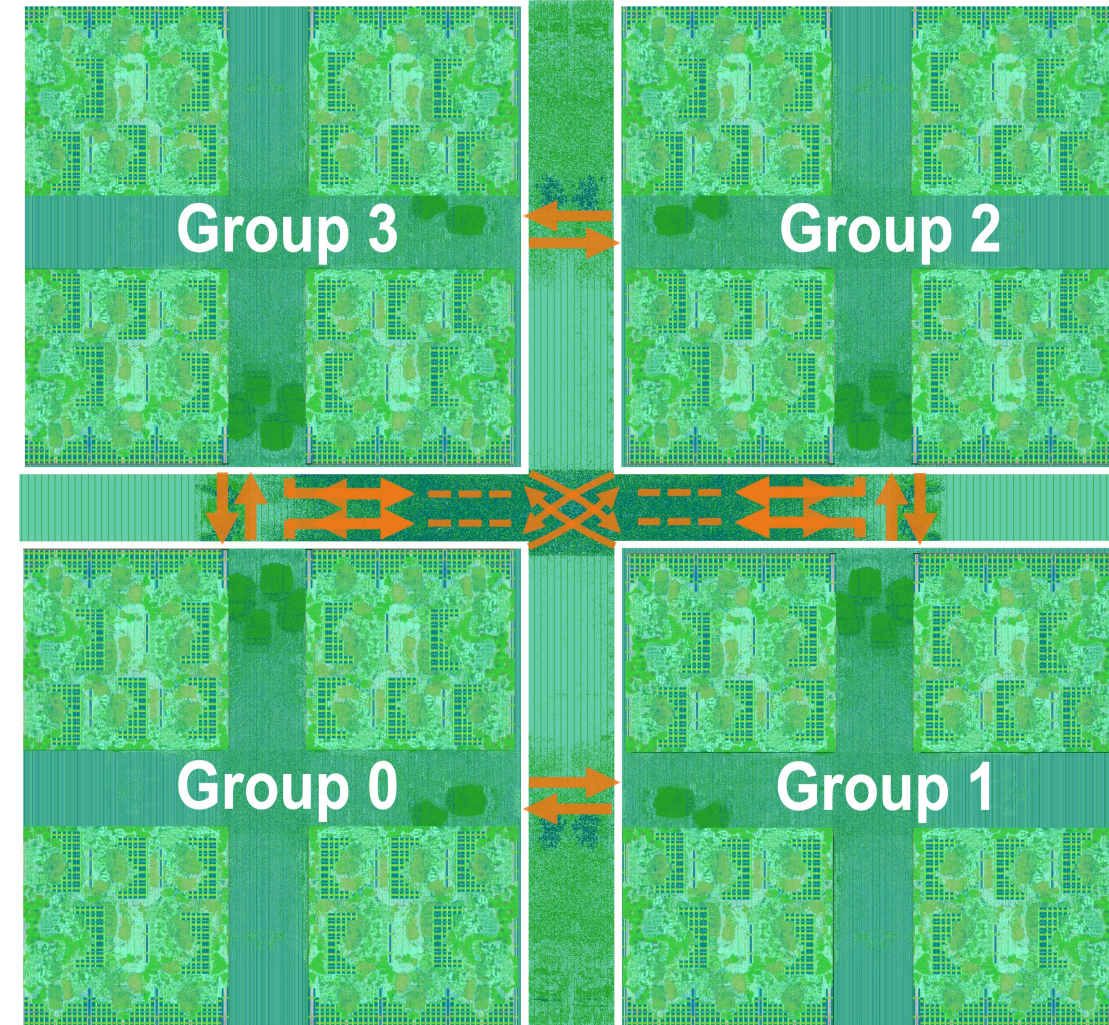
- The push for “Larger”
 - Better global latency tolerance if $L1_{\text{size}} > 2 \times L2_{\text{latency}} \times L2_{\text{bandwidth}}$ (Little’s law + double buffer)
 - Smaller data partitioning overhead
 - Larger Compute/Boundary bandwidth ratio: N^3/N^2 for MMUL grows linearly with N!
- A large “MemPool”: 256+ cores and 1+ MiB of shared L1 data memory



TeraPool: Physically feasible 1024-cores cluster



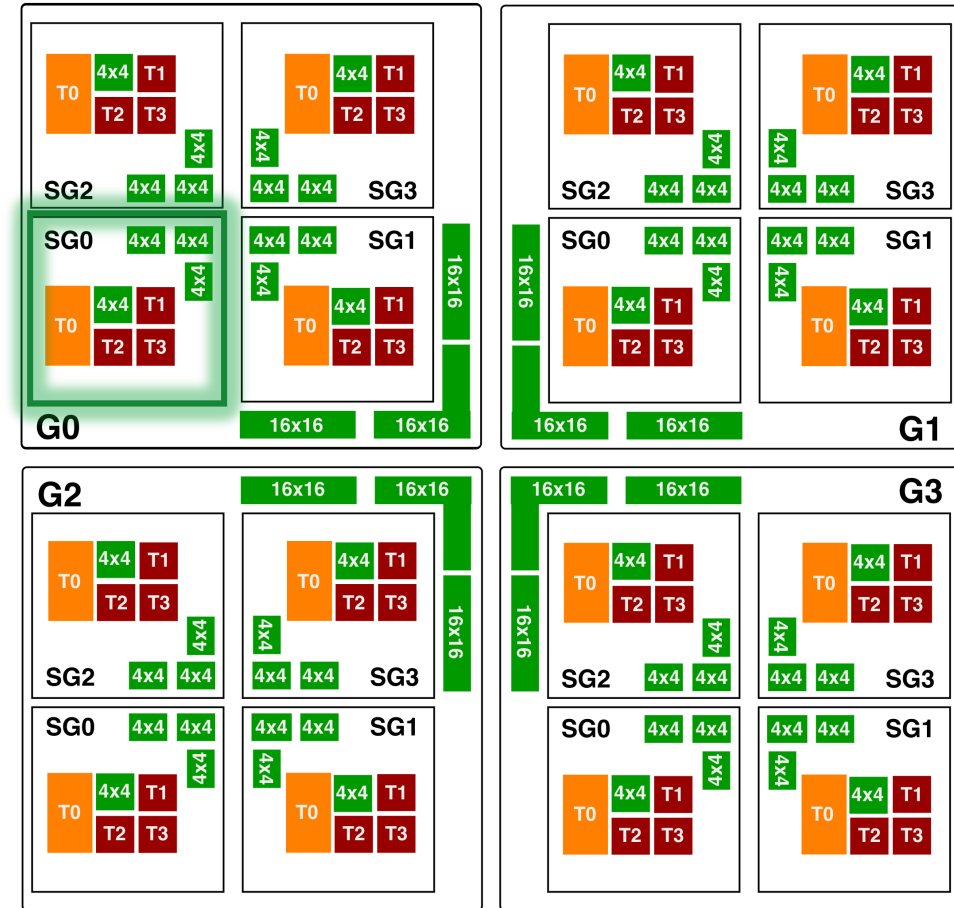
- In GlobalFoundries' 12P+ FinFET, **11 cycles** latency, **740MHz@WC 920MHz@TC**, in 8.3 x 8.3mm²
- **1.89TOPS** peak-performance, **3.2X** on SDR kernels vs MemPool (256-cores)
- Interconnect-scaling does not impact energy efficiency



TensorPool: A Domain Specific Mempool for Physical AI



- Hierarchical Implementation
 - 4 PEs (+ 1 **Tensor Engine**) / Tile with latency tolerant burst interface
 - 4 Tiles / SubGroup
 - 4 SubGroups/Group
 - 4 Groups / Cluster
- Peak Performance:
 - (1 TE + 4x4 PEs) / SG
 - $(1 \times 256_{TE} + 2_{f16} \times 16_{PE}) \times 16 \text{ SGs}$
 $= \mathbf{4608} \text{ MAC}_{f16b} / \text{Cycle/Cluster}$



TensorPool PnR in TSMC's N7



- **6x** more throughput on GEMM
(2x FMAs and 3x utilization 89% vs 30%)
- **9.1x** Area&Energy Efficiency

	TensorPool	TeraPool
Node	7nm	12nm
Area [mm2]	26.6	81.7
T [ns]	1.1	1.1
Peak FLOPs/cycle (TEs + PEs)	9,216	4,096
Peak GFLOPs/s	8,378.18	3723.63
GEMM Utilization	88.94%	29.74%
GEMM FLOPs/cycle	7286	1218
GEMM Power	4.3	6.3
GEMM GFLOPs/s/W/mm2	57.53	6.24

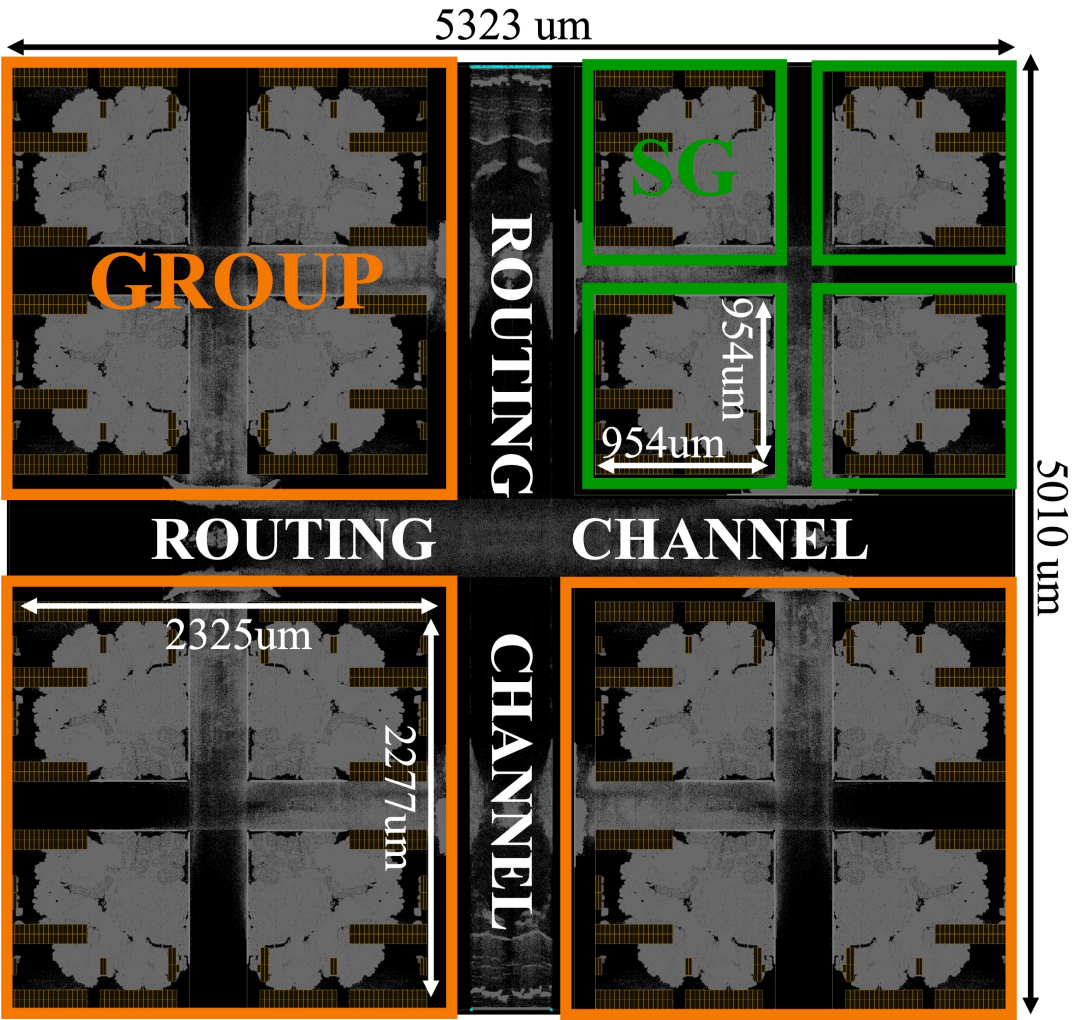
2.25x

1.54x

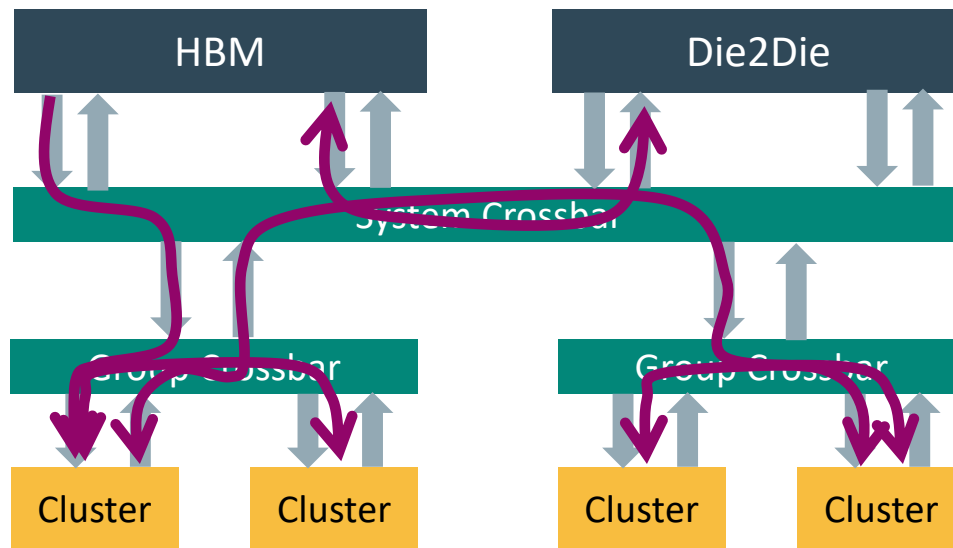
3x

6x

9.1x



Scaling UP: Efficient and Flexible Data Movement



- **Problem:** HBM Accesses are critical in terms of

- Access energy
- Congestion
- High latency

- Instead reuse data on lower levels of the memory hierarchy

- Between **clusters**
- Across **groups**

- Smartly distribute workload

- **Clusters:** Tiling, Depth-First
- **Chiptlets:** E.g. Layer pipelining

Big trend!

Physically Scalable NoC: FlooNoC

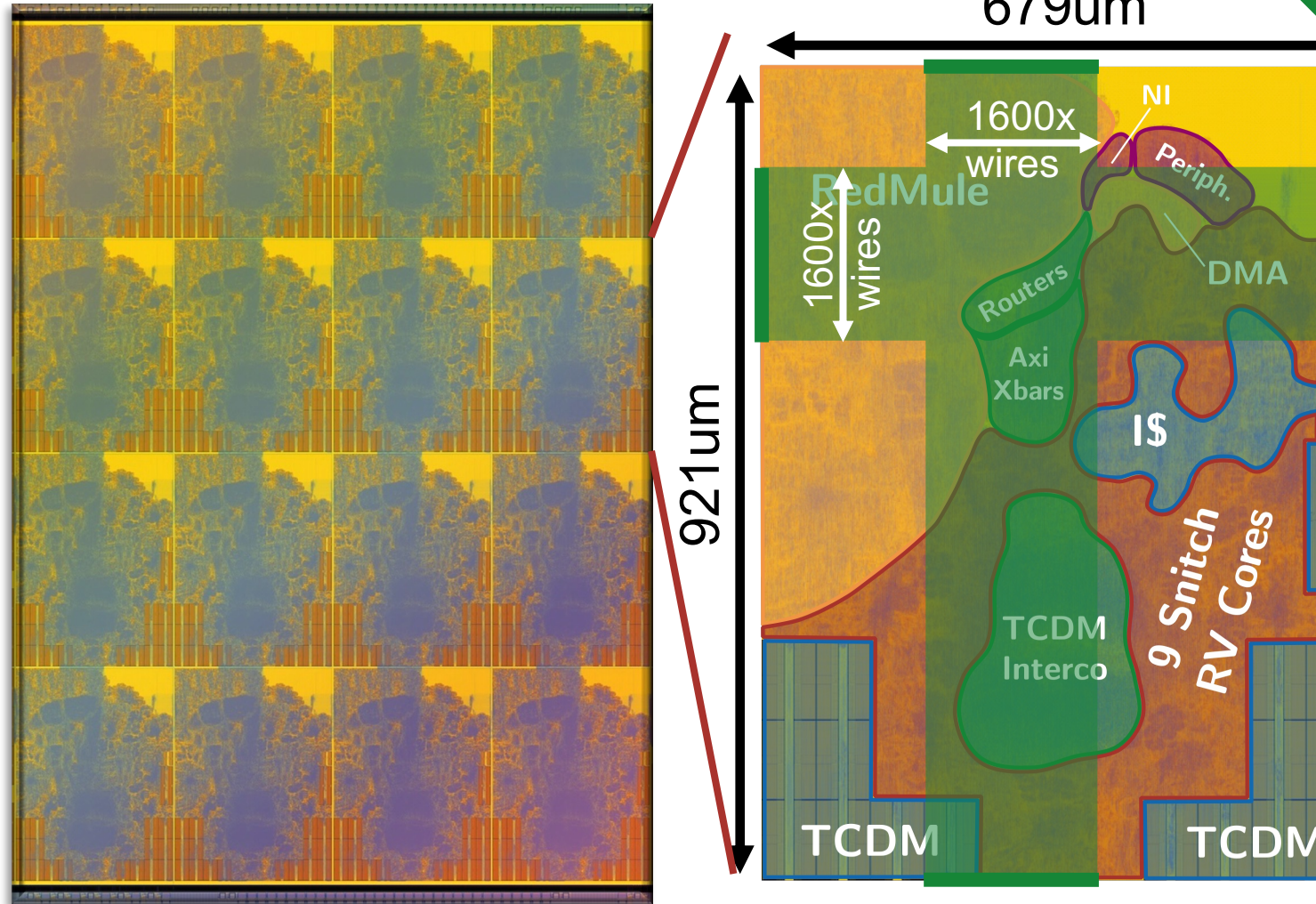


- **Key Ideas in *FlooNoC***

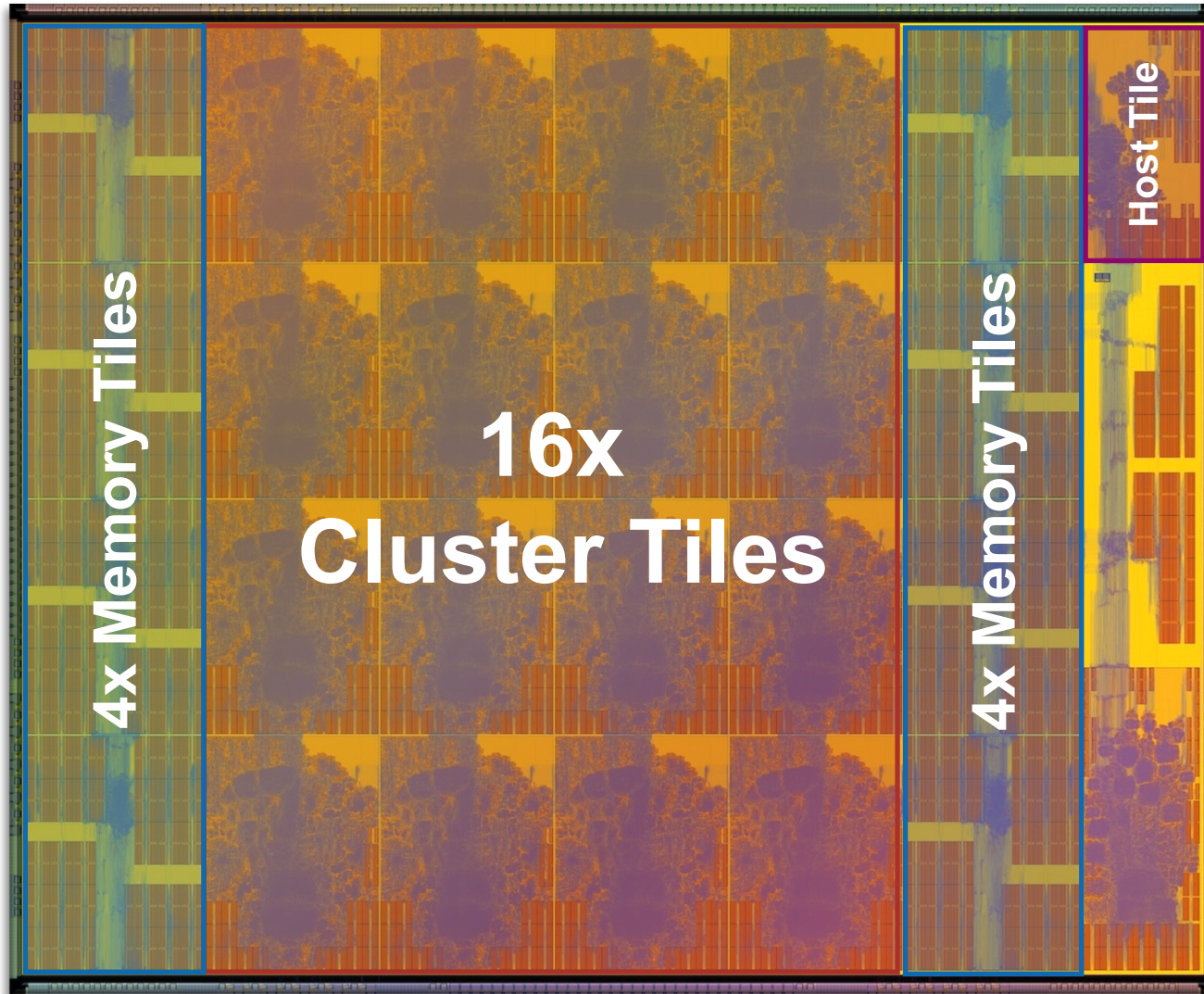
- Fully AXI4 compatible
- Wide physical channels (multiplane vs. VC)
- Over-the-Macro-Routing
- Only 3% of std. cell area taken up by NoC
- Used only ~10% of available routing resources

- **Big area/performance gains**

- *High Bandwidth: 629Gbps/link*
- *High Energy-Efficiency: 0.19pj/B/hop*



Here is Picobello: our 1st Gen Phy-AI Engine in TSMC 7nm



- 16 clusters totaling 144x RISC-V cores with FP8-FP64-bit support
- 8x 1MB of on-chip L2
- Linux capable CVA6 Host
- Peripherals (JTAG, SPI, I2C)
- Running at 1+ GHz (WC),
> 256 GFLOP/s (FP64)
> 2 TFLOP/s (FP8)
- Tape-out August 2025
- Part of the EU Pilot project

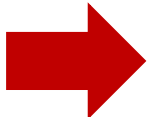
THE **EUPILOT**

MHA Mapping on our Fabric: Flat Attention



- Proposed Dataflow Schedule of MHA
 - We leverage all-cluster L1 for single head attention – Minimize I/O complexity
 - Gen.AI specialized NoC**
 - Matrix transpose engine for transposition of $(K \rightarrow K^T)$
 - Collective operations on NoC

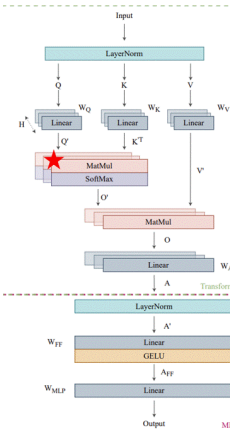
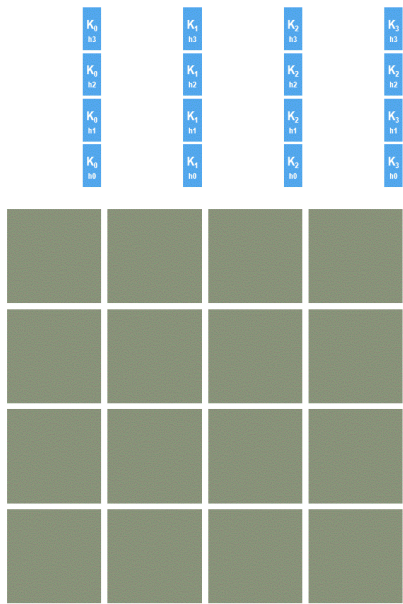
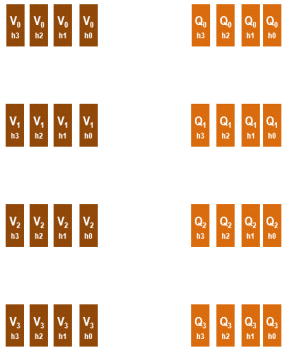
Total Runtime(ms)	
Baseline: Flash Attention for Each Head on Each Cluster	14.4
Flatten Attention (w/o NoC collective)	17.7
Flatten Attention (w/ NoC collective)	4.6



FlattenAtt Step: QK Matmul

- Note
 - After QKV projection, following steps focus only on one head and we process every head sequentially

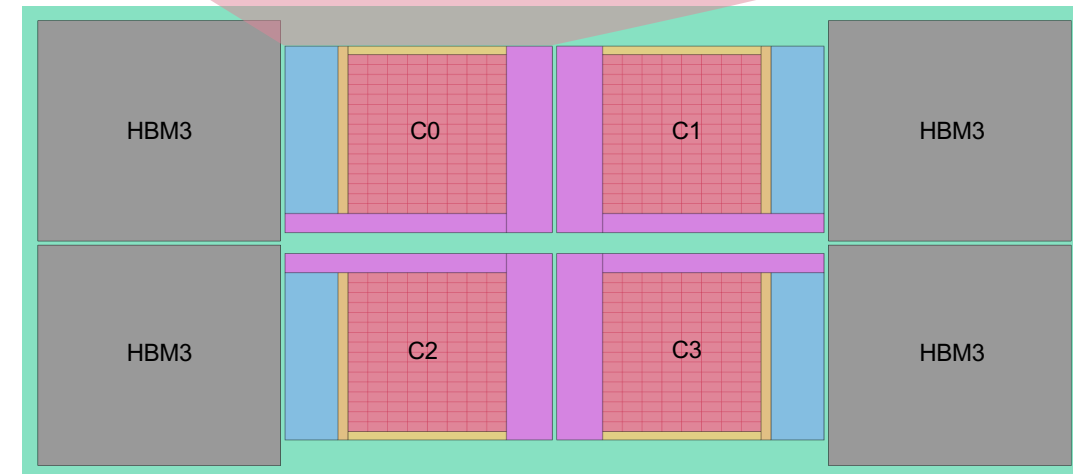
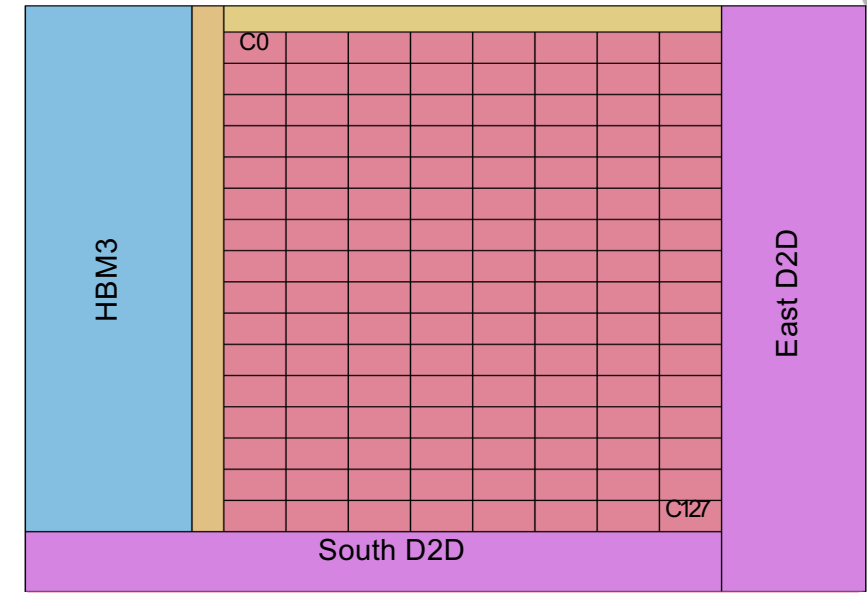
$$Q_{pre} \cdot K_{pre}^T$$



Ogopogo: A 7nm Quad-Chiplet Concept Architecture



- Expands *Picobello* to 16× more clusters
 - Four 16×8 chiplets in TSMC 7nm FinFET
 - Each with *two* D2D links and HBM3
 - 10.3 DP-TFLOP/s peak performance
- Lightweight NoC transport extensions
 1. In-router handling of collectives
 2. In-stream vector operations
- 19% higher node-normalized compute density than Nvidia B200
 - Perf. gap reduced to a matter of die size



All digital IPs are open ISA, open Source HW

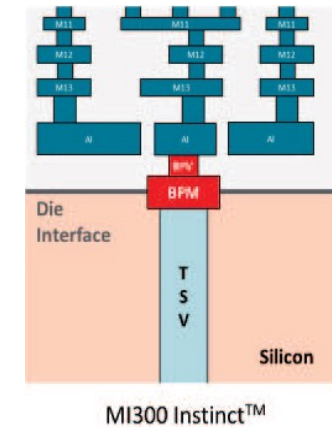
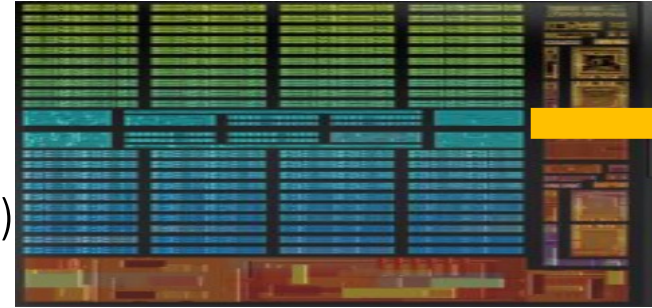
What's Next: 3D



Logic die
Memory + IO die

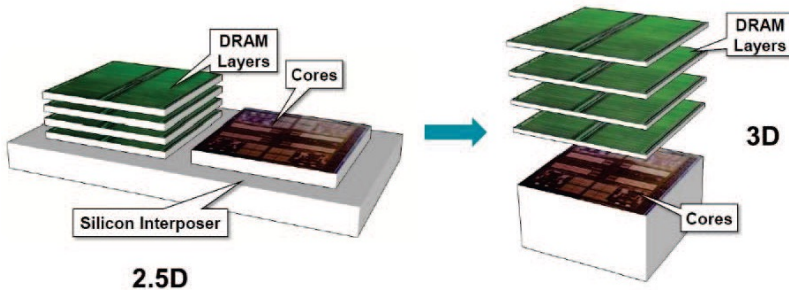
- 3.5D v1

- 3D stacking on logic + 2.5D HBM (AMD MI300)
- Face (top) to Back (bottom)
- Die (top) to Wafer (bottom)

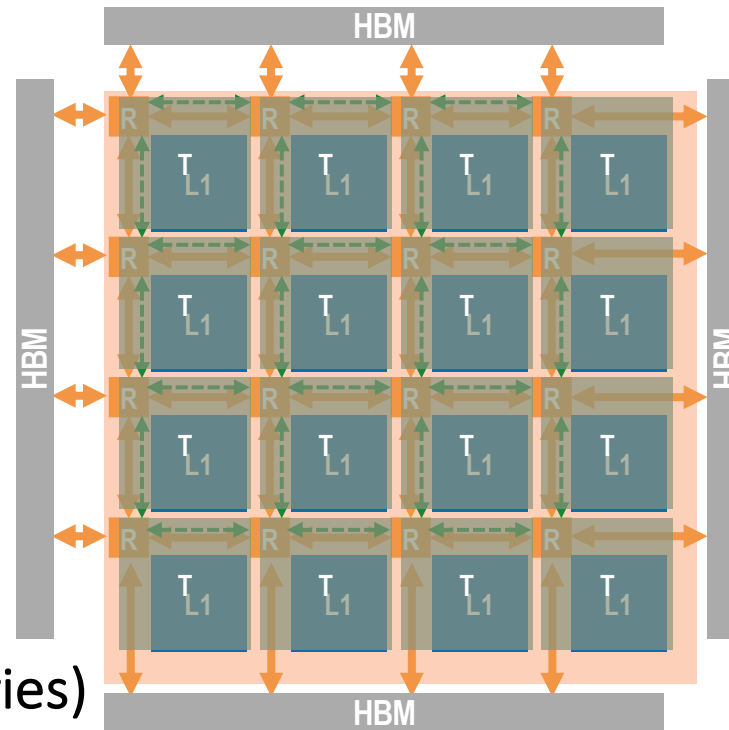


MI300 Instinct™

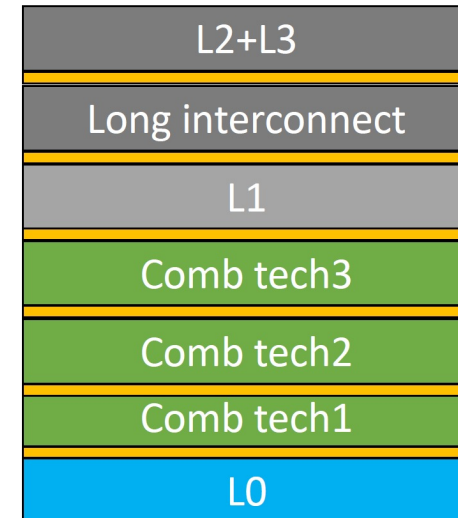
- 3.5D v2?



- Monolithic 3D (CMOS2.0+3D memories)



V1
SRAM+NOC+IO at the bottom



Technology is going “full 3D” → Open Phy.AI 3D platform



Thank You!



youtube.com/pulp_platform



pulp-platform.org



[@pulp_platform](https://twitter.com/pulp_platform)