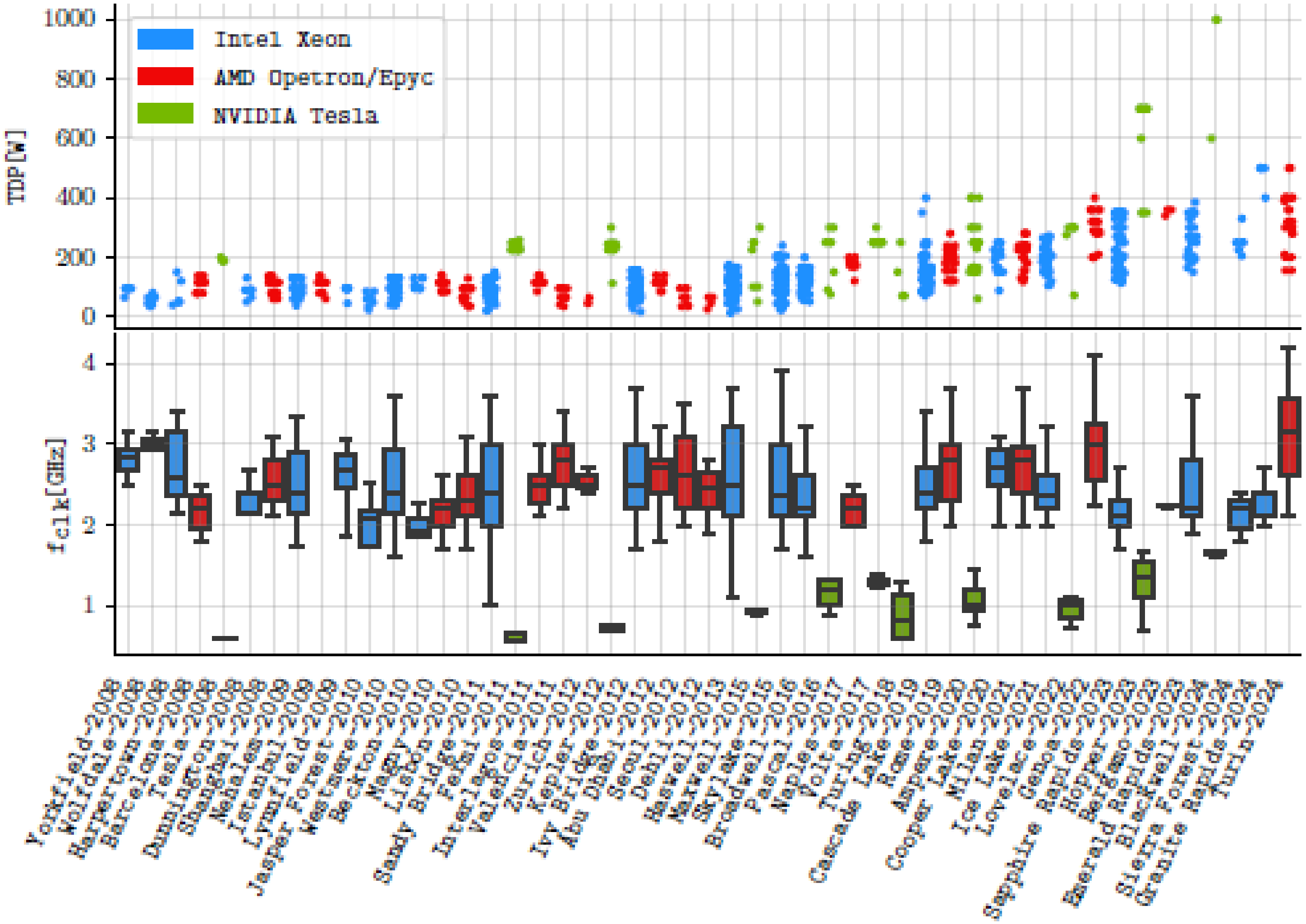
Energy efficiency and power management techniques for AI accelerators

Luca Benini
ETH Zurich, Univ. Bologna

ISSCC 2026 Forum 3.3

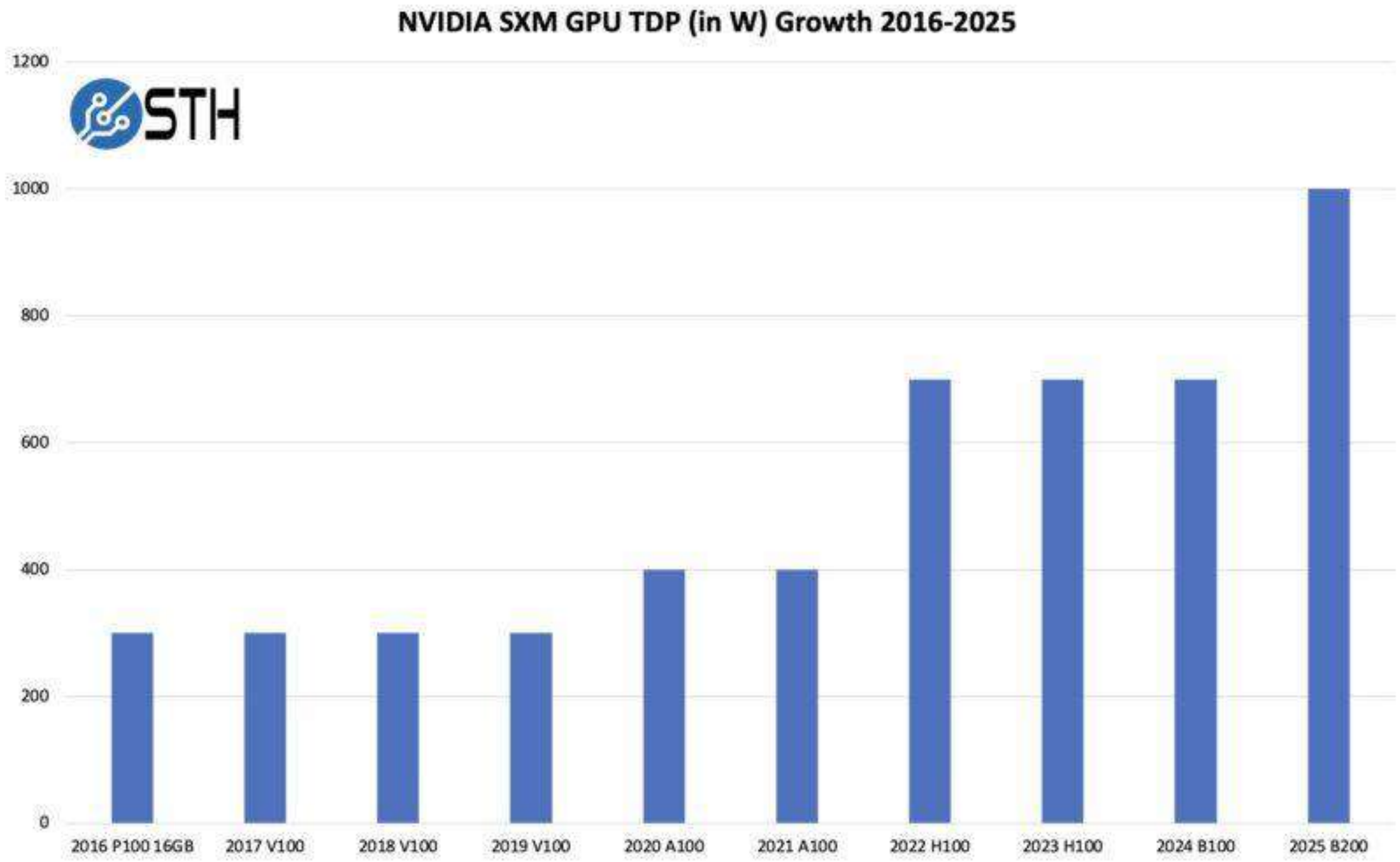
TDP of server Chips is growing

Clock speed and TDP of Intel, AMD, NVIDIA high-end CPUs/GPUs



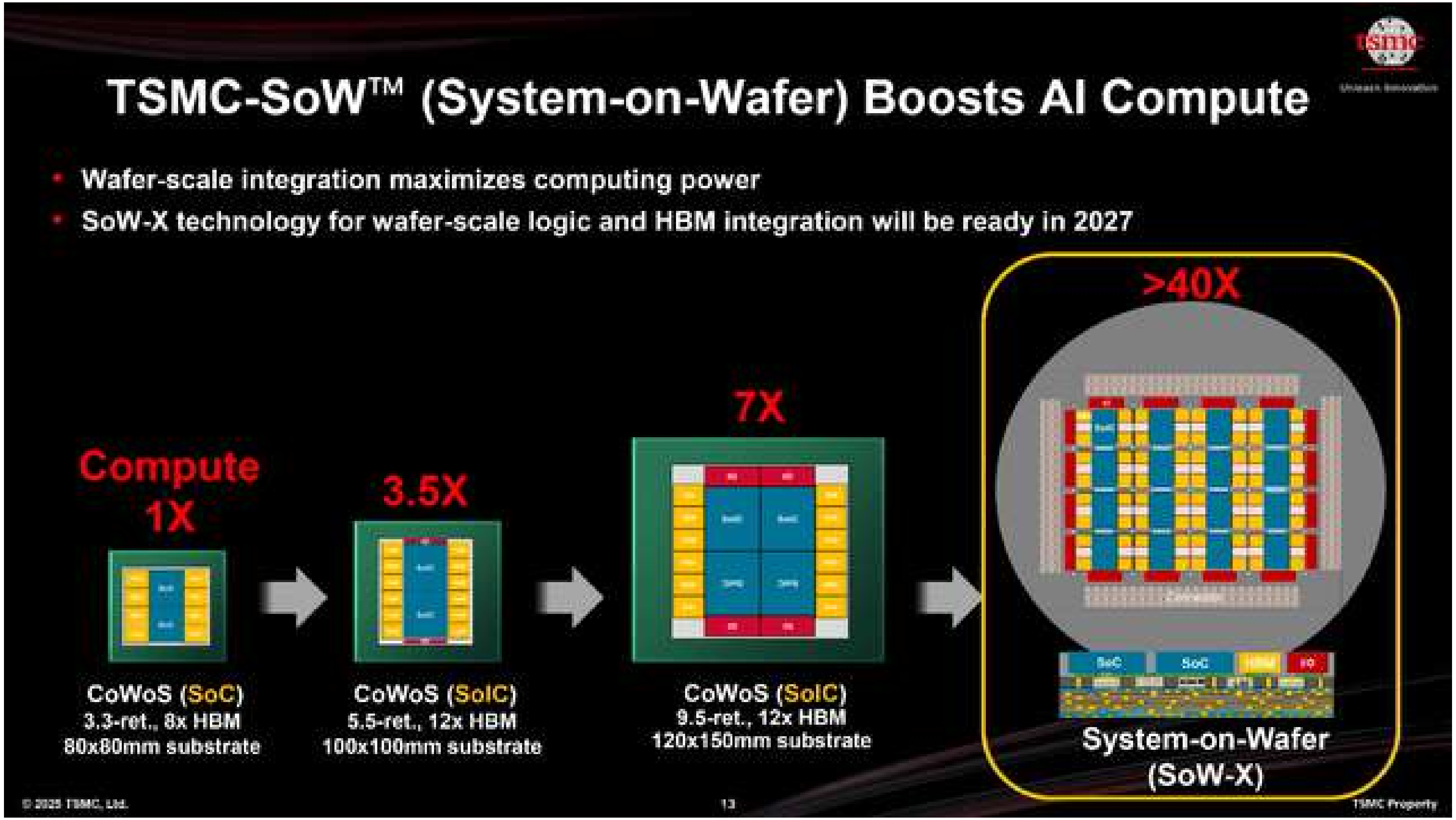
https://doi.org/10.24840/2183-6493_011-002_003138

TDP of server AI Chips is growing Faster



https://www.servethehome.com/why-servers-are-using-so-much-power-tdp-growth-over-time-supermicro-vertiv-intel-amd-nvidia

Why? Ultra-high density, 2.5→3D, WS integration

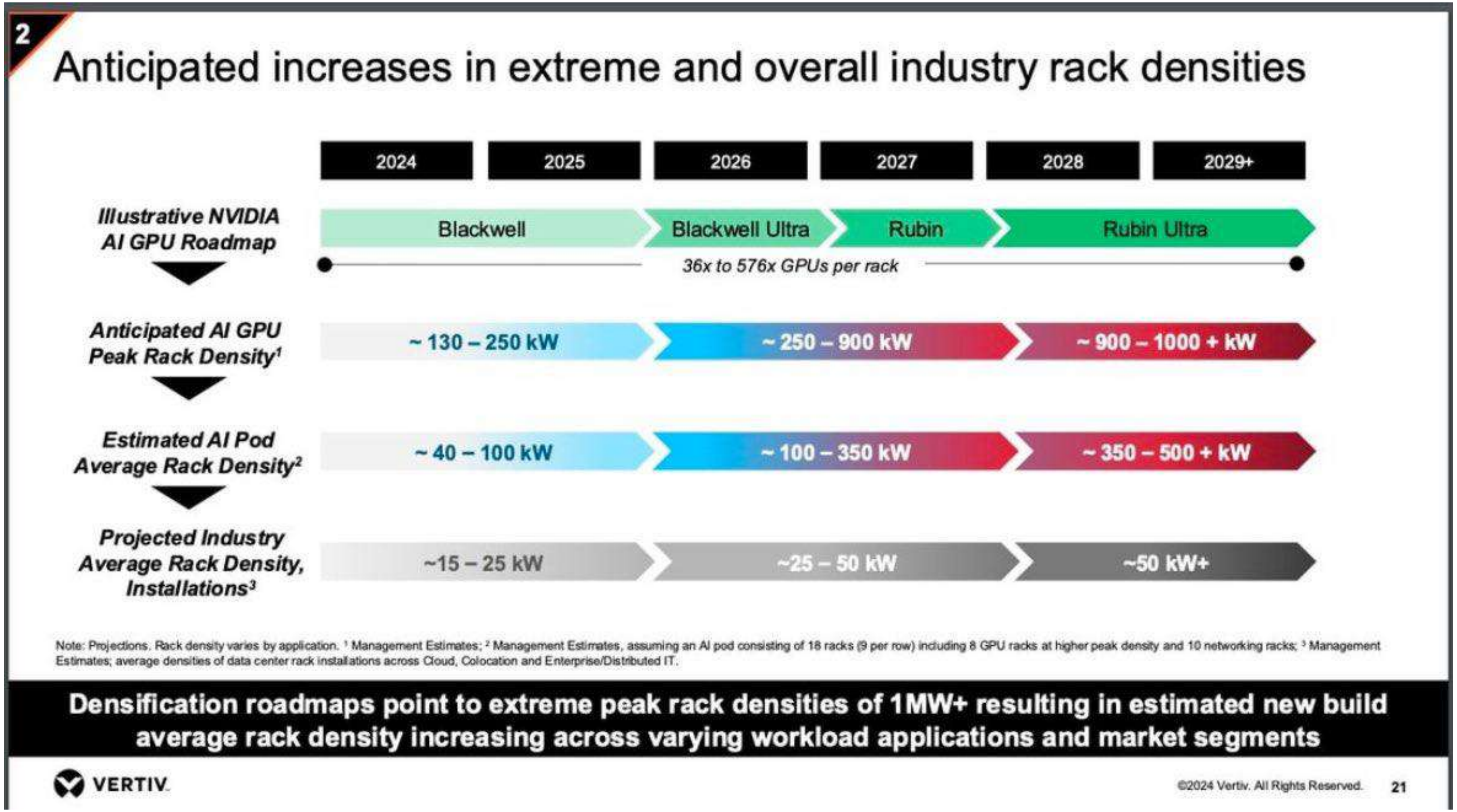


TSMC-SoW™ (System-on-Wafer) Boosts AI Compute

- Wafer-scale integration maximizes computing power
- SoW-X technology for wafer-scale logic and HBM integration will be ready in 2027

© 2025 TSMC, Ltd. 13 TSMC Property

The MW AI Rack (in 2029)



2

Anticipated increases in extreme and overall industry rack densities

2024 | 2025 | 2026 | 2027 | 2028 | 2029+

Illustrative NVIDIA AI GPU Roadmap

Blackwell → Blackwell Ultra → Rubin → Rubin Ultra

36x to 576x GPUs per rack

Anticipated AI GPU Peak Rack Density[1]

~ 130 – 250 kW → ~ 250 – 900 kW → ~ 900 – 1000 + kW

Estimated AI Pod Average Rack Density[2]

~ 40 – 100 kW → ~ 100 – 350 kW → ~ 350 – 500 + kW

Projected Industry Average Rack Density, Installations[3]

~15 – 25 kW → ~25 – 50 kW → ~50 kW+

Note: Projections. Rack density varies by application. [1] Management Estimates; [2] Management Estimates, assuming an AI pod consisting of 18 racks (9 per row) including 8 GPU racks at higher peak density and 10 networking racks; [3] Management Estimates; average densities of data center rack installations across Cloud, Colocation and Enterprise/Distributed IT.

Densification roadmaps point to extreme peak rack densities of 1MW+ resulting in estimated new build average rack density increasing across varying workload applications and market segments

VERTIV.

21

https://wccftech.com/ai-servers-can-reach-rack-density-of-1000-kw-likely-with-nvidias-next-gen-rubin-ultra-architecture/?utm_source=chatgpt.com

Why?

- □ GPU Density: AI systems rely on dense clusters of GPUs and TPUs with larger #MAC/Area at high avg. utilization → more power, heat than CPUs

- □ Continuous Workloads: AI training and inference → sustained high-power for extended periods, unlike the more variable workloads of classical computing

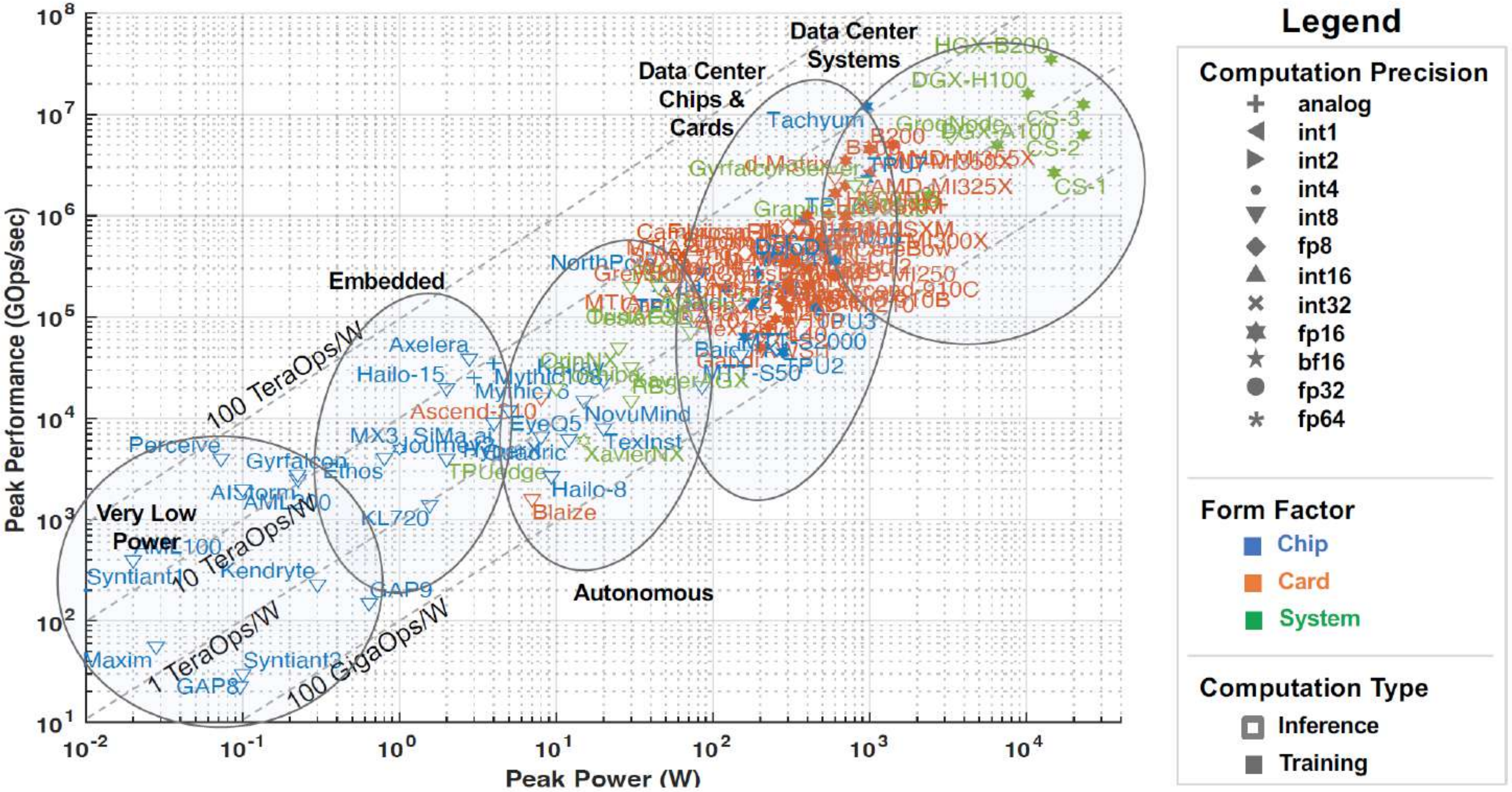
- □ Chip Proximity: To achieve maximum performance and low latency, components are packed closer together → larger power and thermal demands within a single rack package → rack

Outline

- Boosting efficiency for AI workloads
- Managing idleness and heterogeneity in accelerated systems
- Using AI for managing AI
- Conclusions, future perspectives

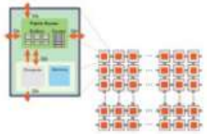
The Cambrian Explosion

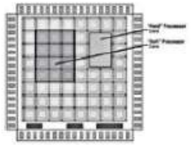
Domain-Specific Architectures

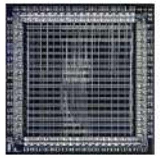


Reuther25 https://arxiv.org/abs/2510.20931v1

Domain-Specific Architecture Classes







	CPU Core	CPU Core Vector Engine	Parallel Thread Accelerator (GPUs)	Tensor Array Accelerator (TPU, Groq, TensorCore)	Microcore Mesh Accelerator (Cerebras, Tenstorrent)	Computational Block Accelerator (FPGA)	Custom Dataflow Accelerator (ASIC)
Technology label	CPU	Vector	GPU	Tensor	Manycore	FPGA	ASIC
ALUs per core	1-4	8-32	32-64	8x8 to 256x256	1 to 4	Various	App. specific
Cores per processor	4-64	4-64	8-128	1 to 4	100s to 1M	Various	App. specific
Parallel performance	Low	Medium Low	Medium High	High	High	High	Very High
Comp. efficiency (Ops/W)	Low	Medium Low	Medium	High	High	High	Very High
Comp. flexibility	Very High	Medium Low	Medium	Medium Low	Medium	Low	Very Low
Computation scheduling	Dynamic by instruction	Dynamic by instruction	Dynamic by kernel	Static by kernel	Static by kernel	Fixed	Fixed
Code redesign	Seconds	Seconds	Seconds/Minutes	Minutes	Minutes/Hours	Hours	Months

Increasing application specificity -> greater parallel performance -> narrower application/computational kernel enablement

Reuther25 https://arxiv.org/abs/2510.20931v1

Heterogeneous Architecture

Multiple Scales of acceleration

Extensions to processor cores

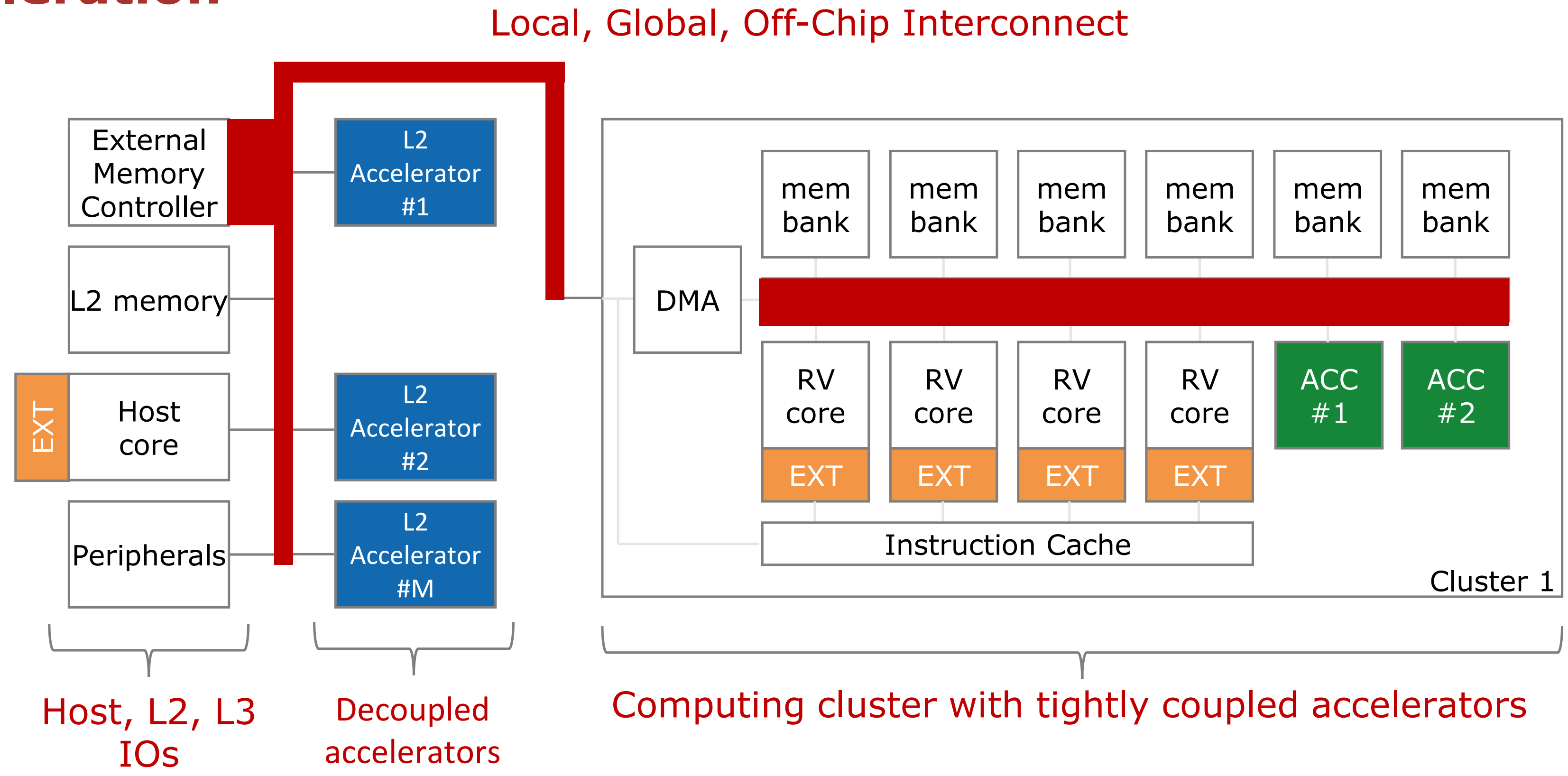
- Explore new extensions
- Efficient implementations

Shared-memory Accelerators

- Domain specific
- Local memory

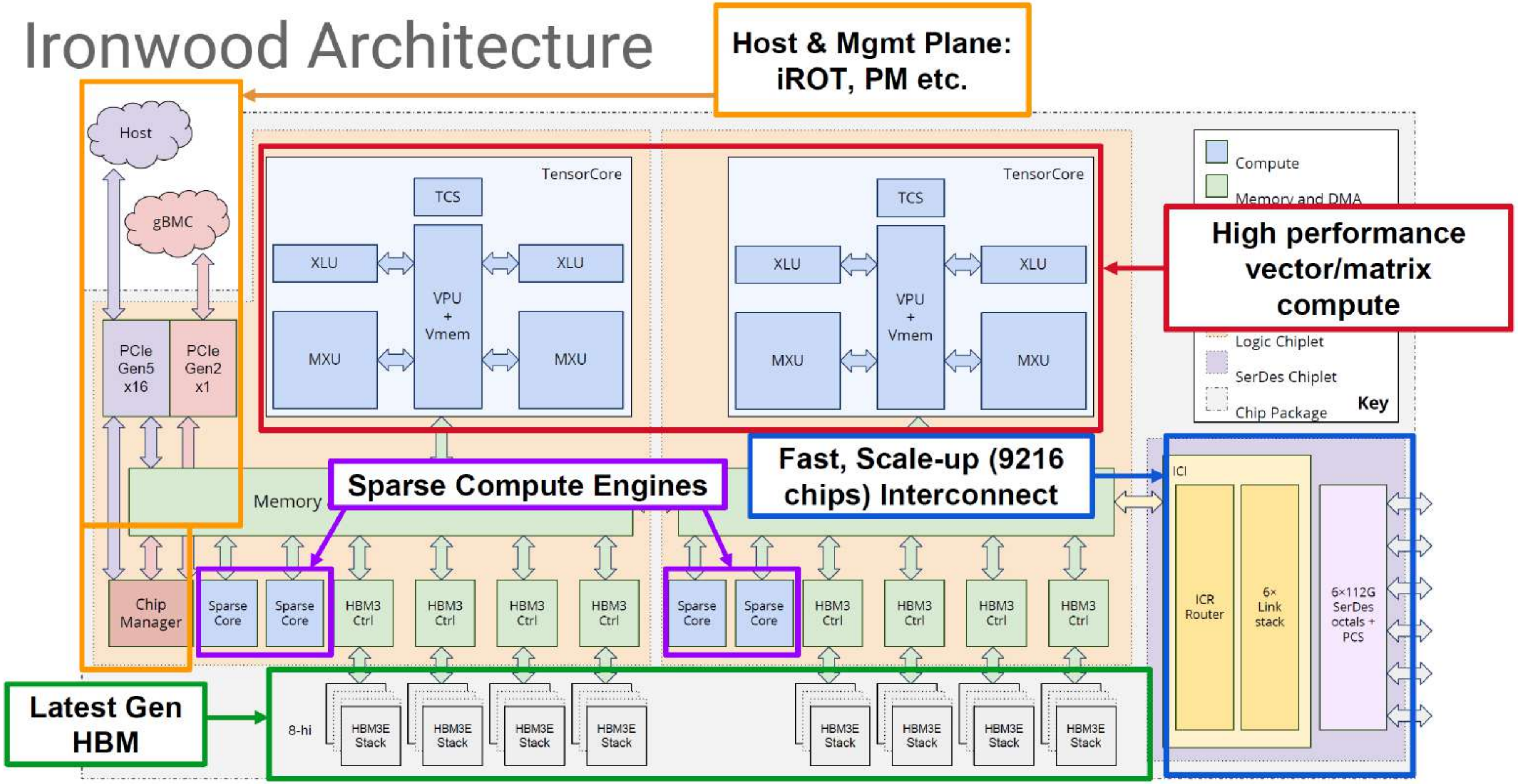
Multiple Decoupled Accelerators

- Communication
- Synchronization



Local, global, package, system

A notable Example: Google’s Ironwood



Host & Mgmt Plane: iROT, PM etc.

High performance vector/matrix compute

Sparse Compute Engines

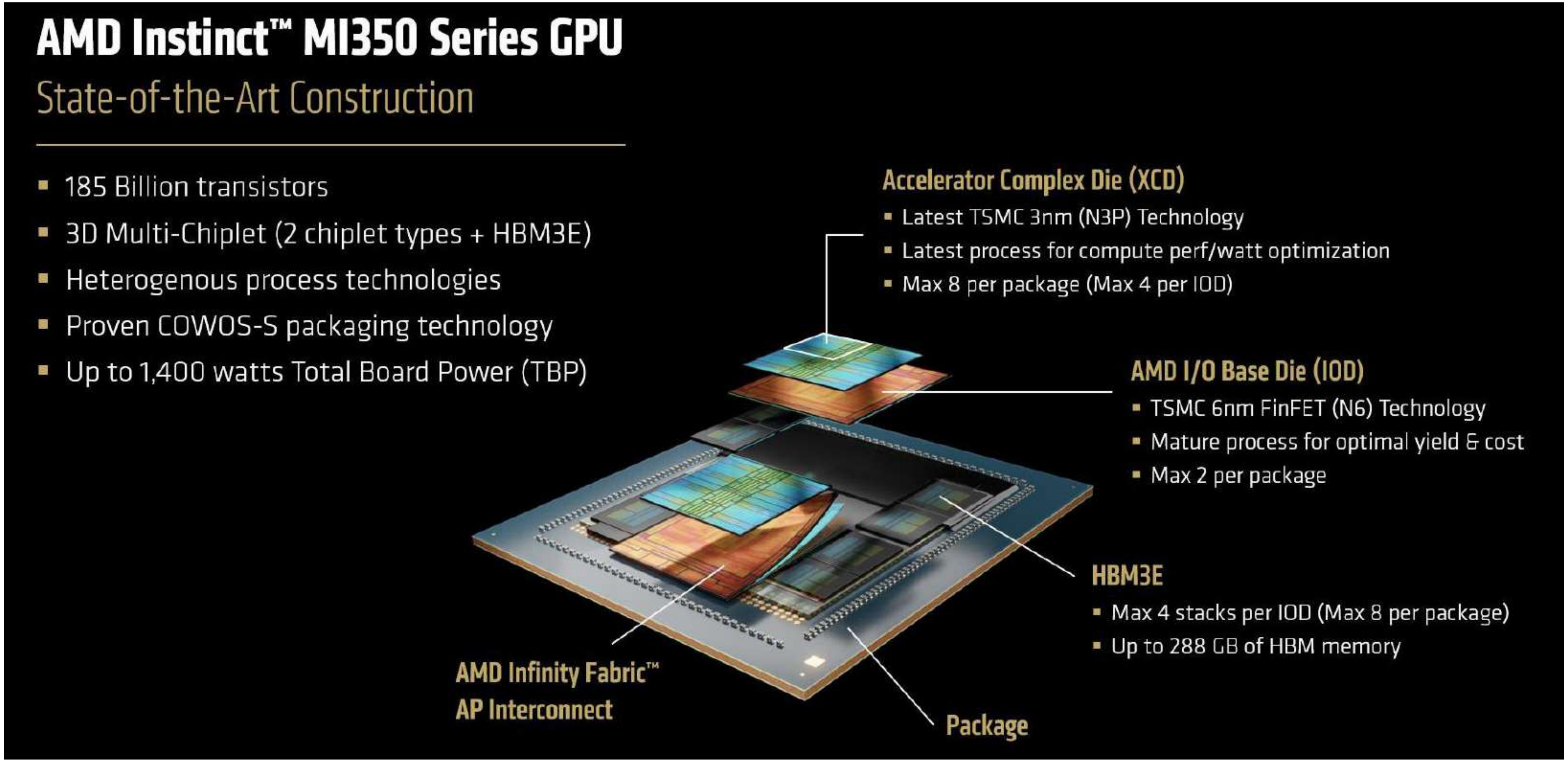
Fast, Scale-up (9216 chips) Interconnect

Latest Gen HBM

Ironwood Chip

- First dual compute die TPU, 4614 TFLOPS of FP8
  - >10x compute compared to TPU v5p
  - Capable of large scale pretraining of foundation models
- 8 stacks of HBM3E, peak BW 7.3 TB/s, capacity 192 GiB
  - Optimized for serving latest generation thinking models
- 1.2 TBps I/O to gluelessly scale-up to 9216 chips
- Industry leading cold plate thermal solution
- Integrated root-of-trust (iROT) for secure computing
- Functional BIST & silent data corruption (SDC) mitigation
- Logic repair to improve yield
- Dynamic voltage/frequency scaling for efficient perf/W
- AI based ALU circuits, floor plan optimization

Looking forward: more (3D) Heterogeneity



AMD Instinct™ MI350 Series GPU

State-of-the-Art Construction

- 185 Billion transistors
- 3D Multi-Chiplet (2 chiplet types + HBM3E)
- Heterogenous process technologies
- Proven COWOS-S packaging technology
- Up to 1,400 watts Total Board Power (TBP)

Accelerator Complex Die (XCD)

- Latest TSMC 3nm (N3P) Technology
- Latest process for compute perf/watt optimization
- Max 8 per package (Max 4 per IOD)

AMD I/O Base Die (IOD)

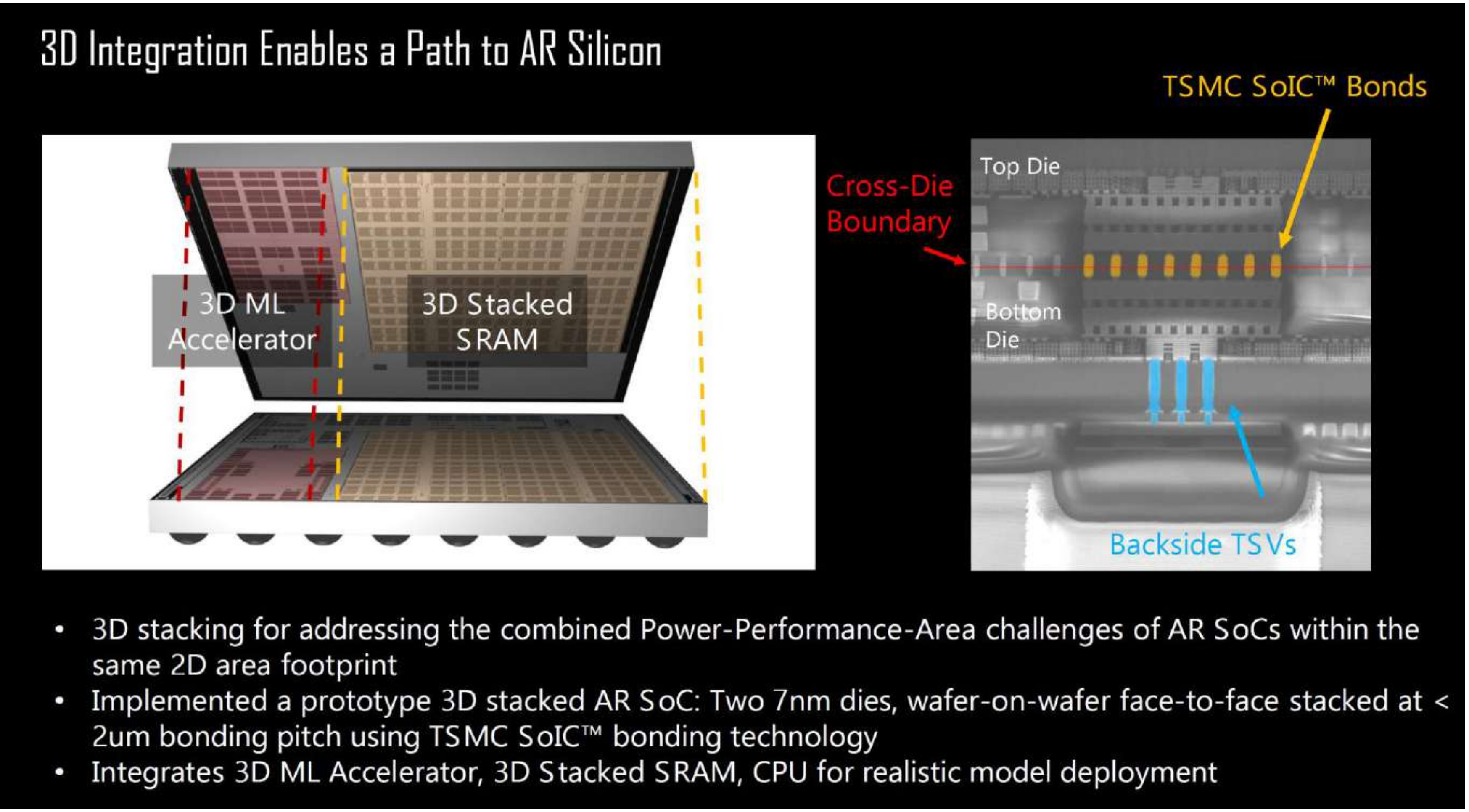
- TSMC 6nm FinFET (N6) Technology
- Mature process for optimal yield & cost
- Max 2 per package

HBM3E

- Max 4 stacks per IOD (Max 8 per package)
- Up to 288 GB of HBM memory

AMD – Hot Chips 2025

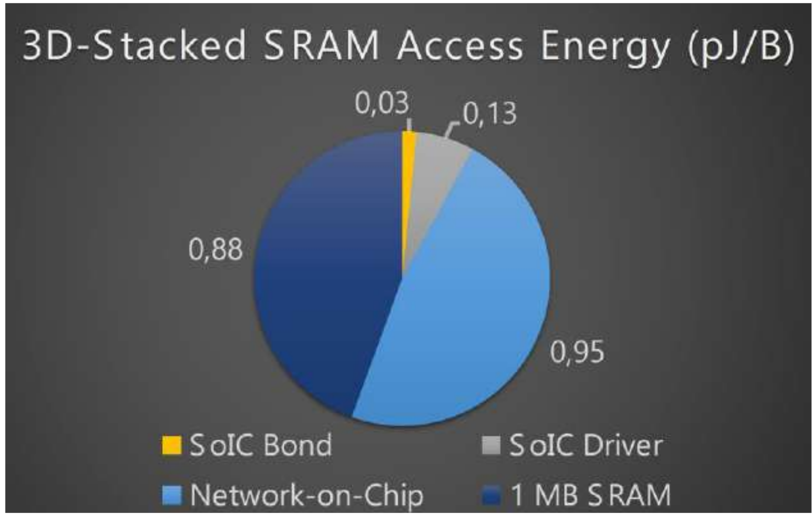
Looking forward: more (3D) Heterogeneity



3D Integration Enables a Path to AR Silicon

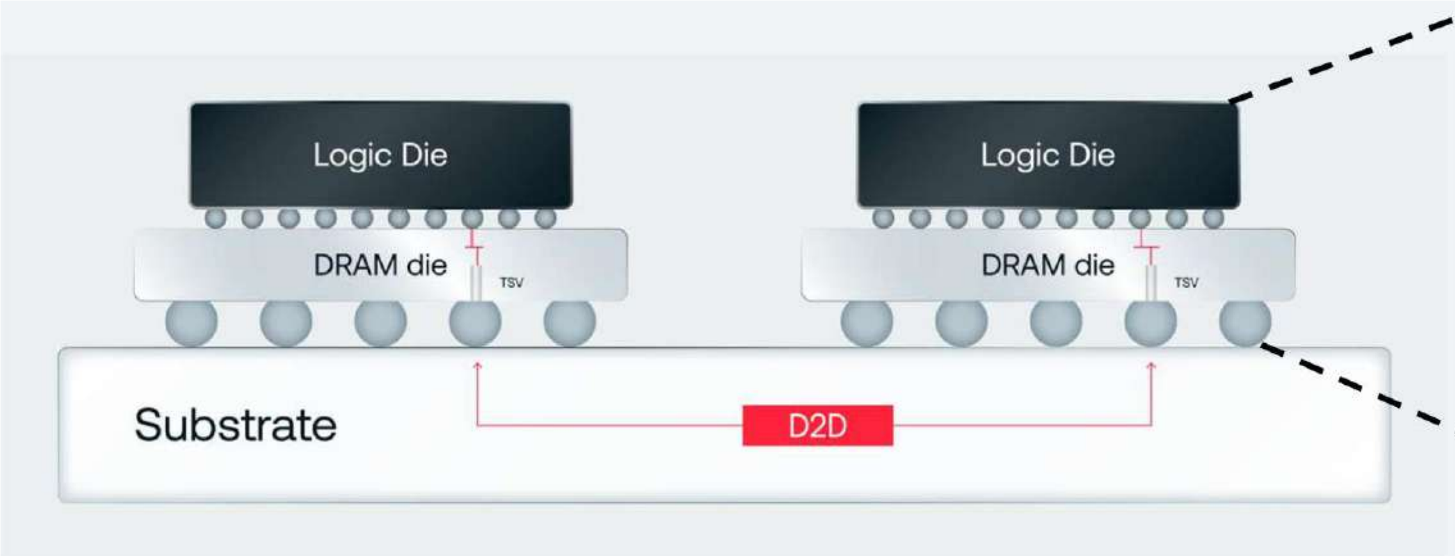
TSMC SoIC™ Bonds
3D ML Accelerator
3D Stacked SRAM
Cross-Die Boundary
Top Die
Bottom Die
Backside TSVs

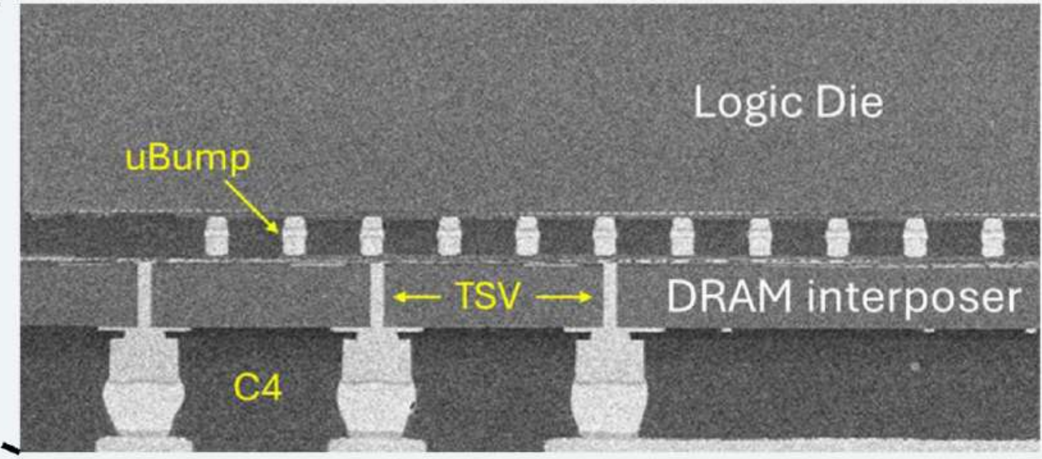
- 3D stacking for addressing the combined Power-Performance-Area challenges of AR SoCs within the same 2D area footprint
- Implemented a prototype 3D stacked AR SoC: Two 7nm dies, wafer-on-wafer face-to-face stacked at < 2um bonding pitch using TSMC SoIC™ bonding technology
- Integrates 3D ML Accelerator, 3D Stacked SRAM, CPU for realistic model deployment



META – ISSCC24 (Beigne et al.)

Looking forward: more (3D) Heterogeneity

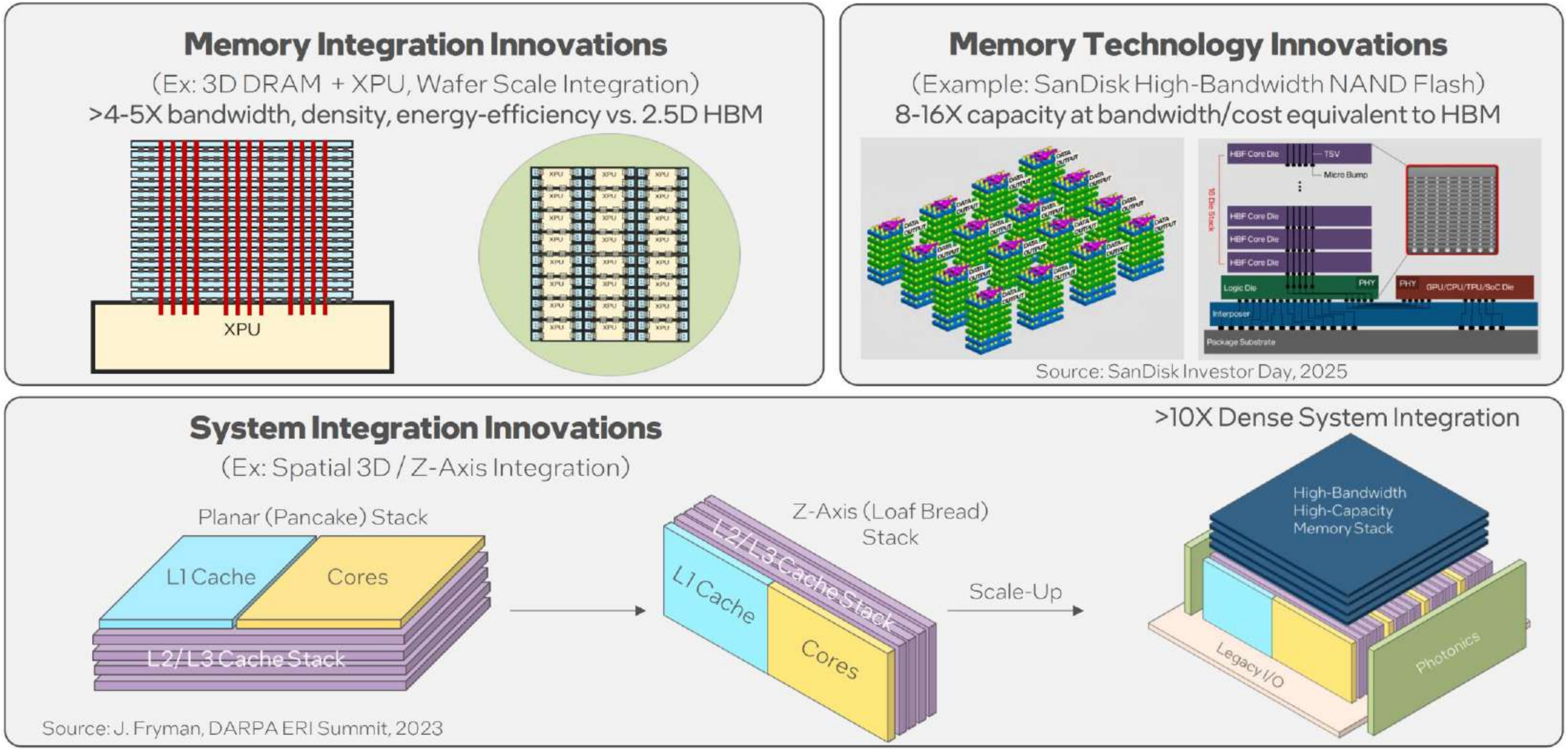




- Top die: TSMC N5 logic
- Bottom die: 3D DRAM
- Integration: 36µ Face to Face (F2F) stacking
- Proven low cost, high volume, high yield process

D-MATRIX – Hot Chips 2025

Looking forward: more (3D) Heterogeneity



Memory Integration Innovations

(Ex: 3D DRAM + XPU, Wafer Scale Integration)

>4-5X bandwidth, density, energy-efficiency vs. 2.5D HBM

Memory Technology Innovations

(Example: SanDisk High-Bandwidth NAND Flash)

8-16X capacity at bandwidth/cost equivalent to HBM

Source: SanDisk Investor Day, 2025

System Integration Innovations

(Ex: Spatial 3D / Z-Axis Integration)

>10X Dense System Integration

Source: J. Fryman, DARPA ERI Summit, 2023

INTEL CHISIC 2025

Efficiency Quest – Summary

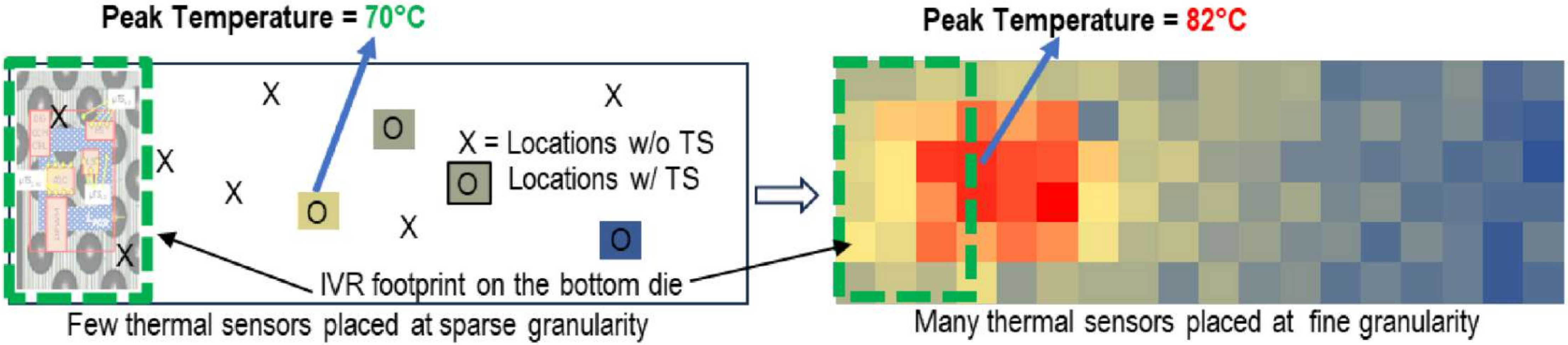
- □ Huge variety of domain-specific architectures → PM flexibility is crucial
- □ Heterogeneity is key → fine grained and heterogeneous power managers with lots of sensor data streams and multiple concurrent, heterogeneous PM tasks
- □ 3D integration → closed-loop massive MIMO power and thermal management is essential – active cooling is likely going fine-grained

Outline

- □ Boosting efficiency for AI workloads
- □ Managing idleness and heterogeneity in accelerated systems
- □ Using AI for managing AI
- □ Conclusions, future perspectives

Ultra-fine grained (digital) sensing (Temp)

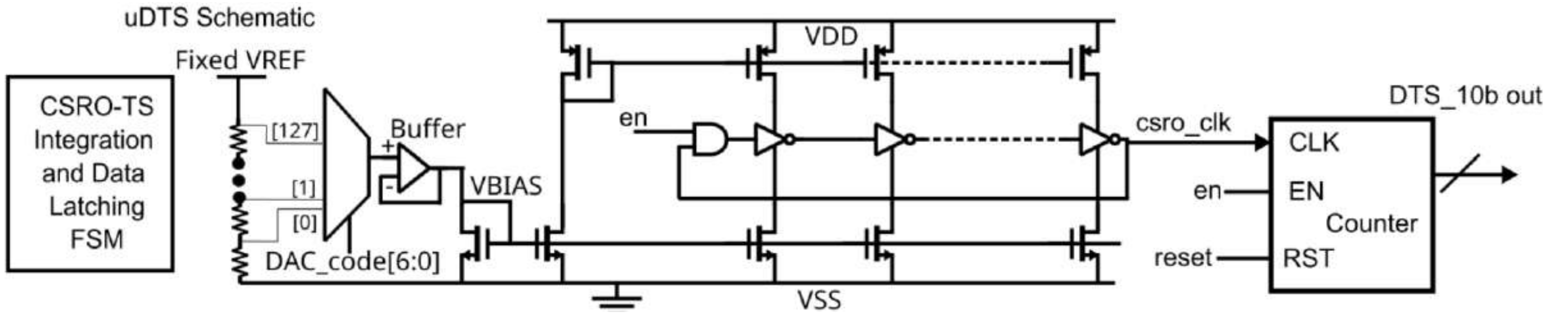
- Micro-scale Sensors placed in a fine-grained array improve resolution and enable better thermal decisions


Peak Temperature = 70°C

Peak Temperature = 82°C

Few thermal sensors placed at sparse granularity

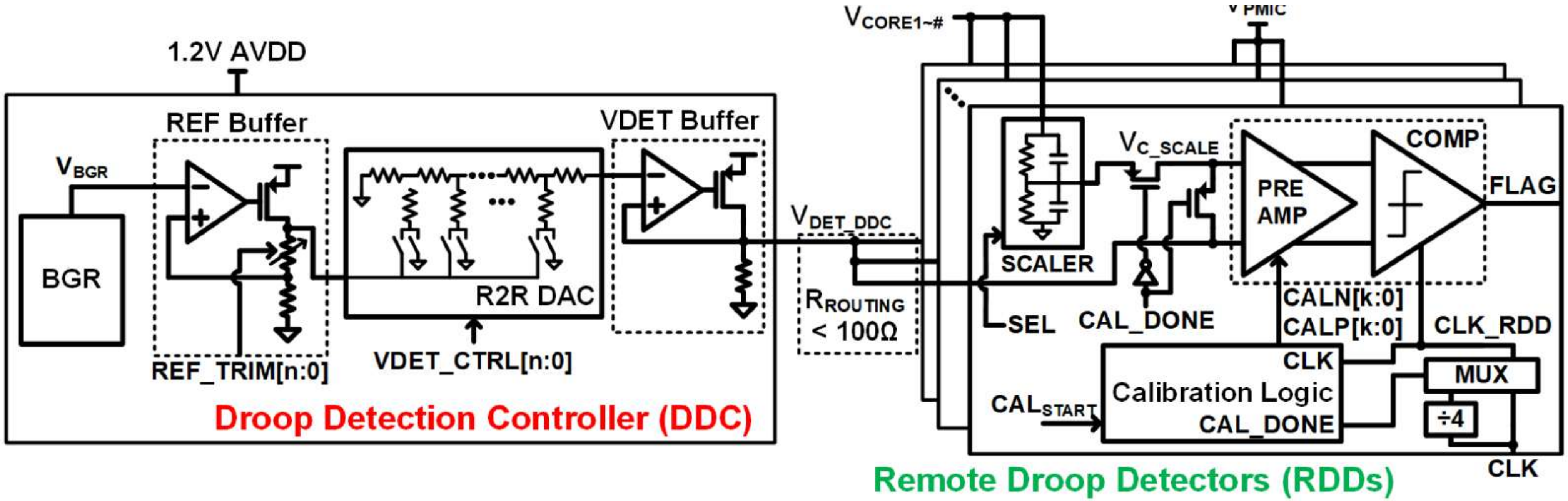
Many thermal sensors placed at fine granularity



- **Linearity and sensitivity of the CSRO-TS are programmed through VBIAS**
- **Programmable integration time enables noise robustness**

INTEL ISSCC25 8.8

Ultra-fine grained (mixed-signal) sensing (Vdd droop)

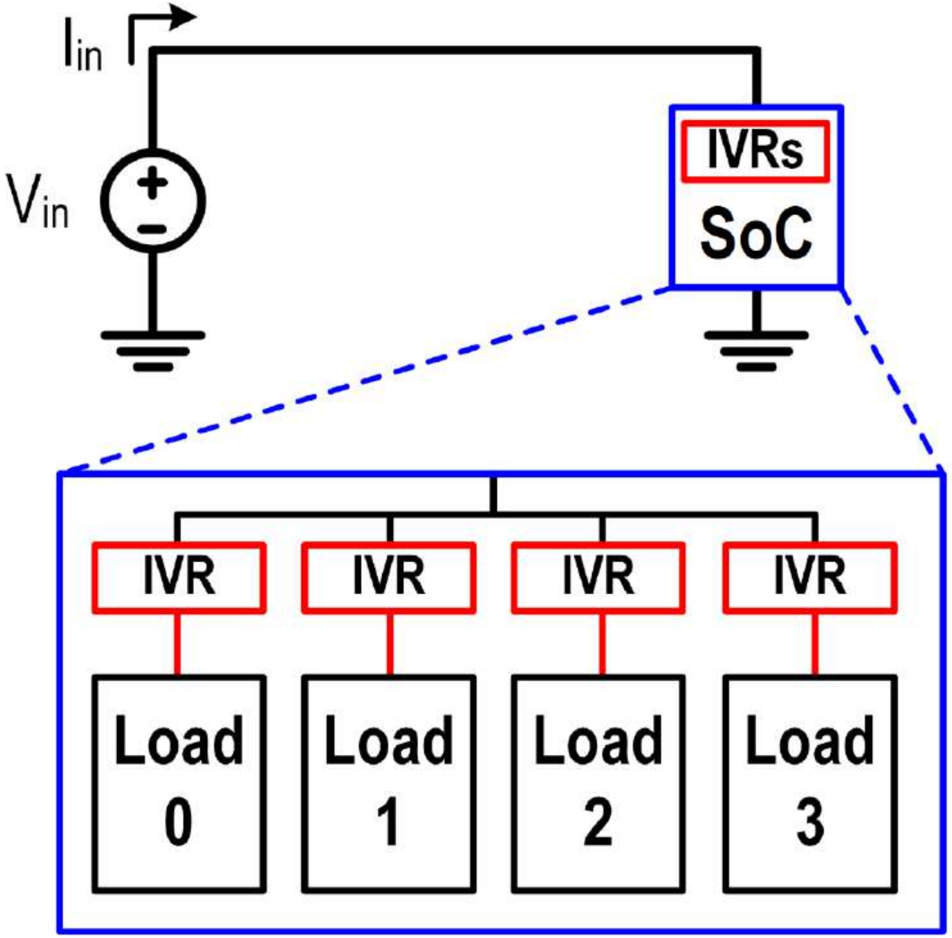


- DDC: Variation-tolerant reference generation → High Accuracy
- RDD: Self-calibrated comparator

  High-speed comparator & Scaler → High-Speed
- Multiple RDDs share a single DDC. → Area Efficiency

Samsung ISSCC25 8.4

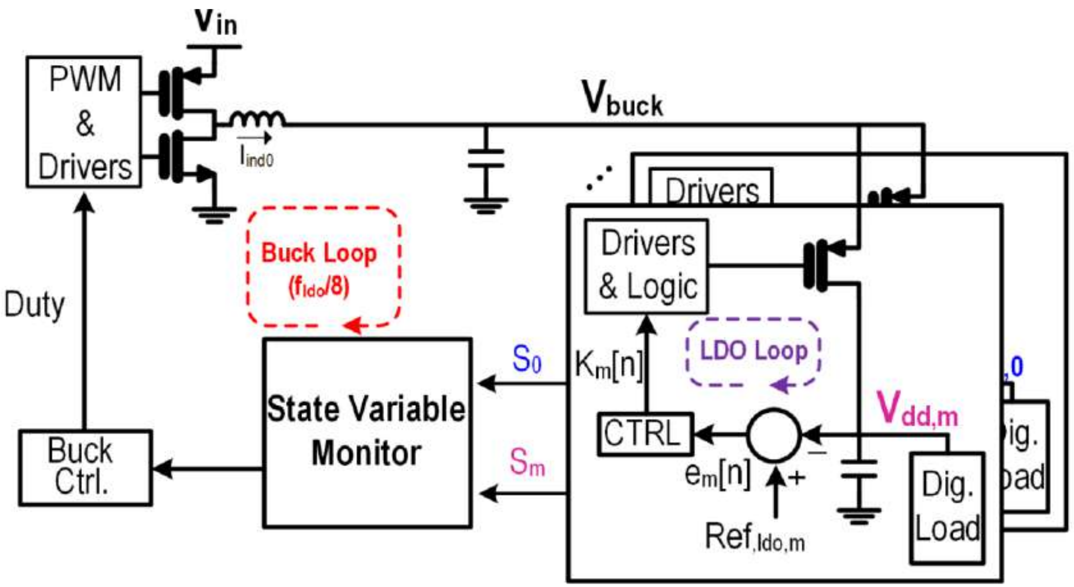
Fine-grained Regulation (Vdd)



- **Minimize <u>total</u> energy dissipation**
  - Voltage Regulator (VR)
  - Load (SoC):
- **Fine-grained DVFS**
  - Improved η_{soc}
  - Needs multiple VRs
- **Power Distribution Network → IVR**
  - Scalability vs η_{VR} tradeoffs

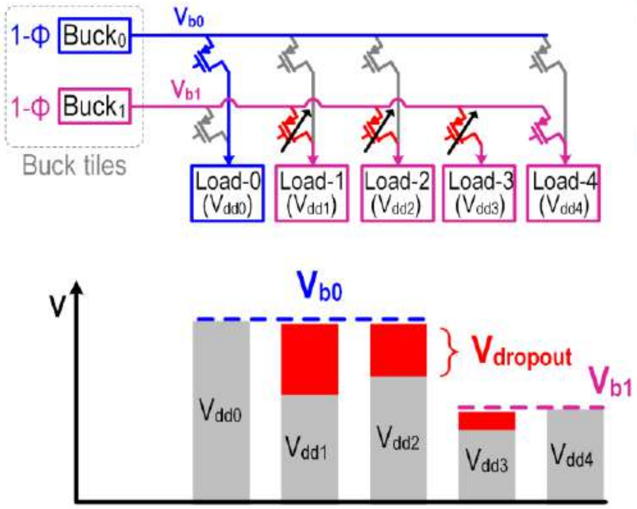
Gatech ISSCC25 8.3

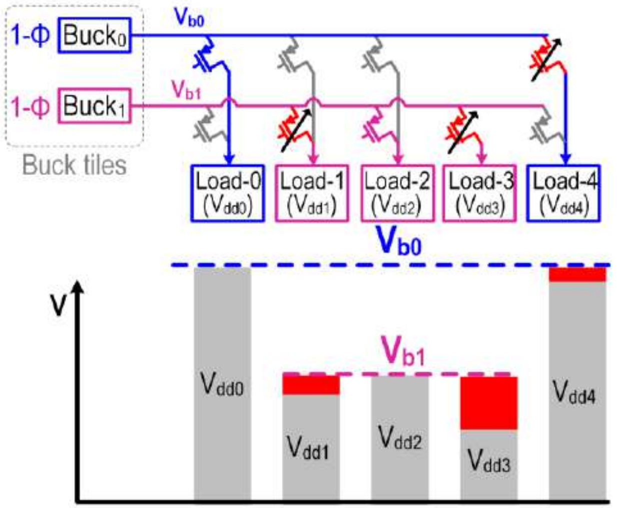
Fine-grained Regulation (Vdd)



- Sample State Variables from all LDOs
- **Establish the lowest V_{buck} to satisfy all V_{dd} targets**

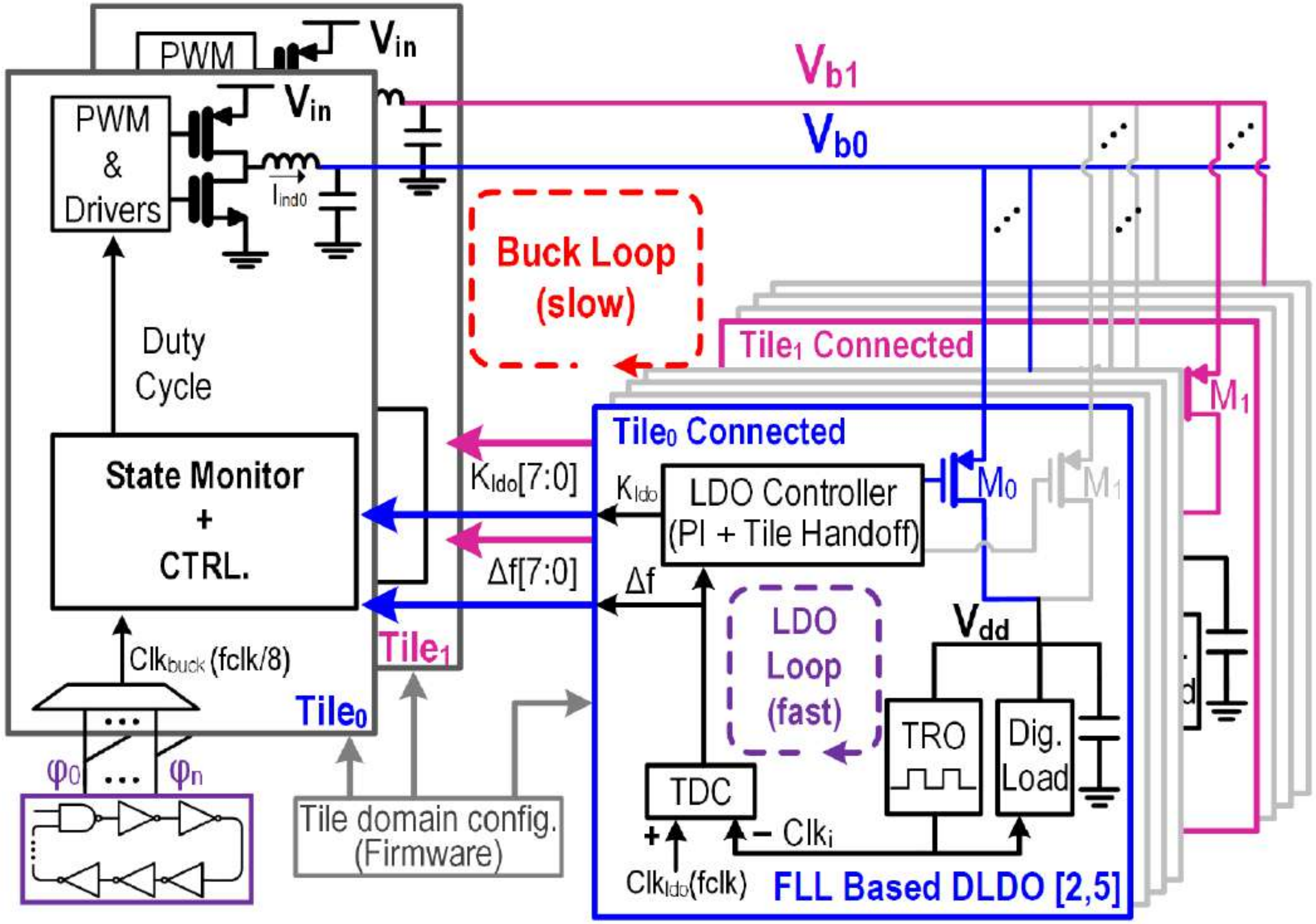
- **Partial Crossbar**
  - Near-optimal domain assignment
- **Runtime Reconfigurable connectivity**





Gatech ISSCC25 8.3

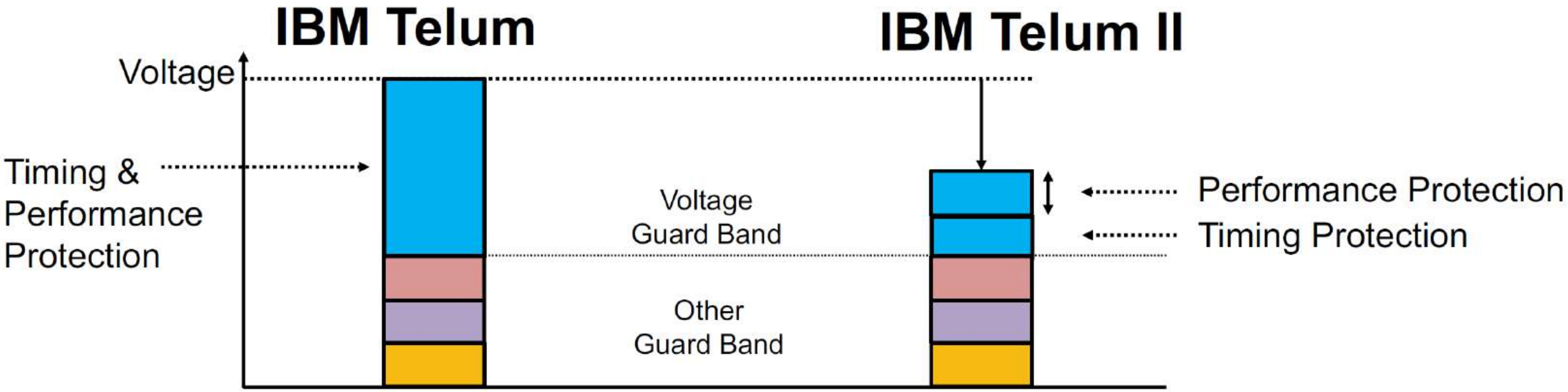
Fine-grained Regulation (Vdd)



- Domain runtime re-configurability (M_0, M_1)
- Controller for VR and hot-swapping domain transition
- LDO state variable broadcasting to each buck tile
- Phase selection for 1ϕ or 2ϕ Buck

Gatech ISSCC25 8.3

MIMO Control: IBM Telum2 (Z-systems)



Timing & Performance Protection

- lumped together
- static V_{SET} and droop mitigation

Timing Protection

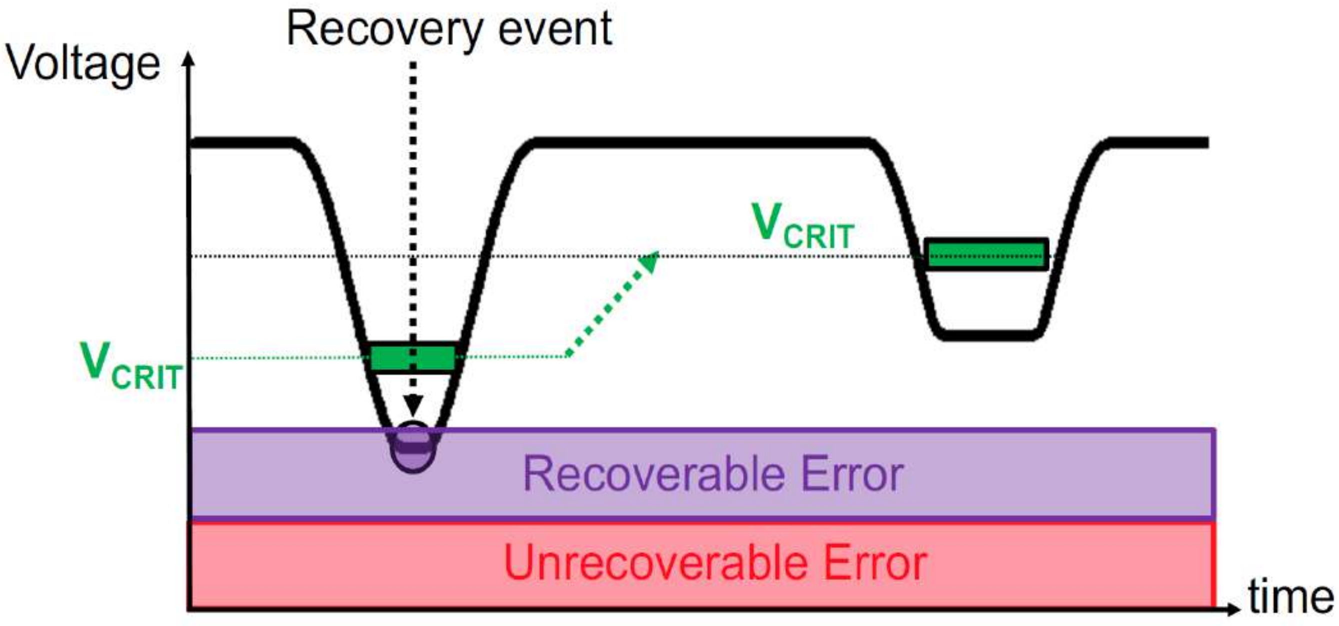
- Droop mitigation only

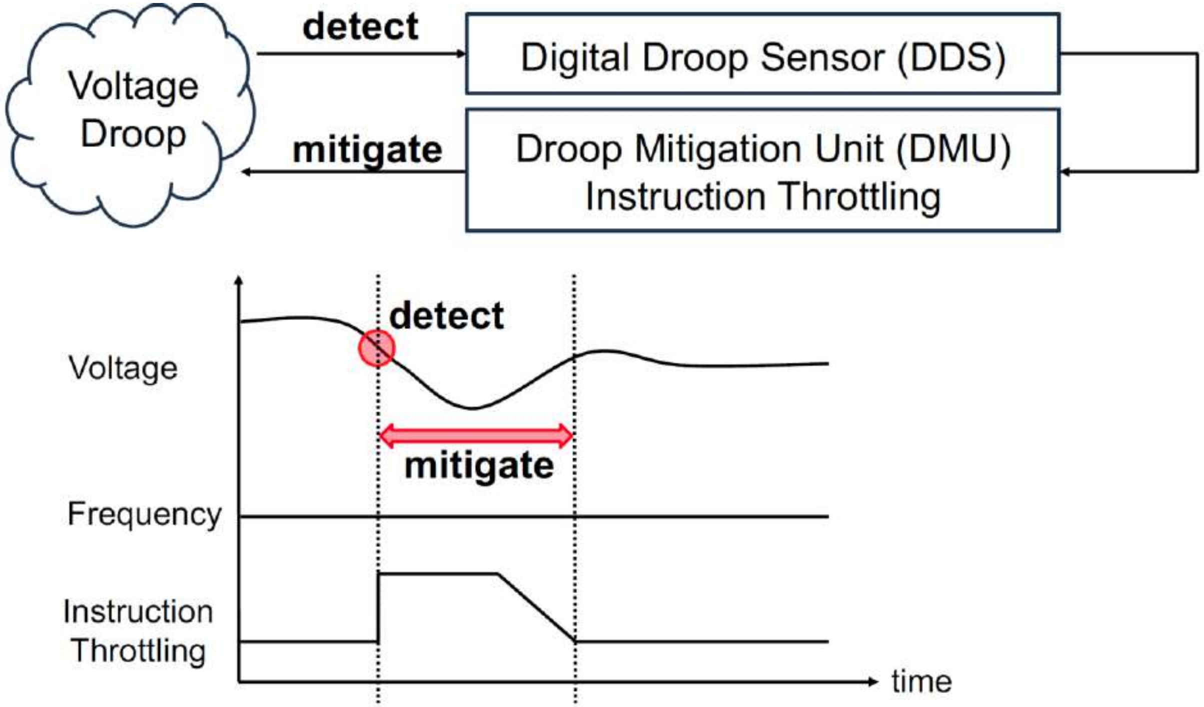
Performance Protection

- Dynamic
- Workload dependent

IBM ISSCC25 8.1

MIMO Control: IBM Telum2 (Z-systems)





- □ Multiple interacting control loops
  - Performance Loop
  - Voltage droop loop
  - Recoverable error loop (exploiting RAS feature – incl. robust core recovery)

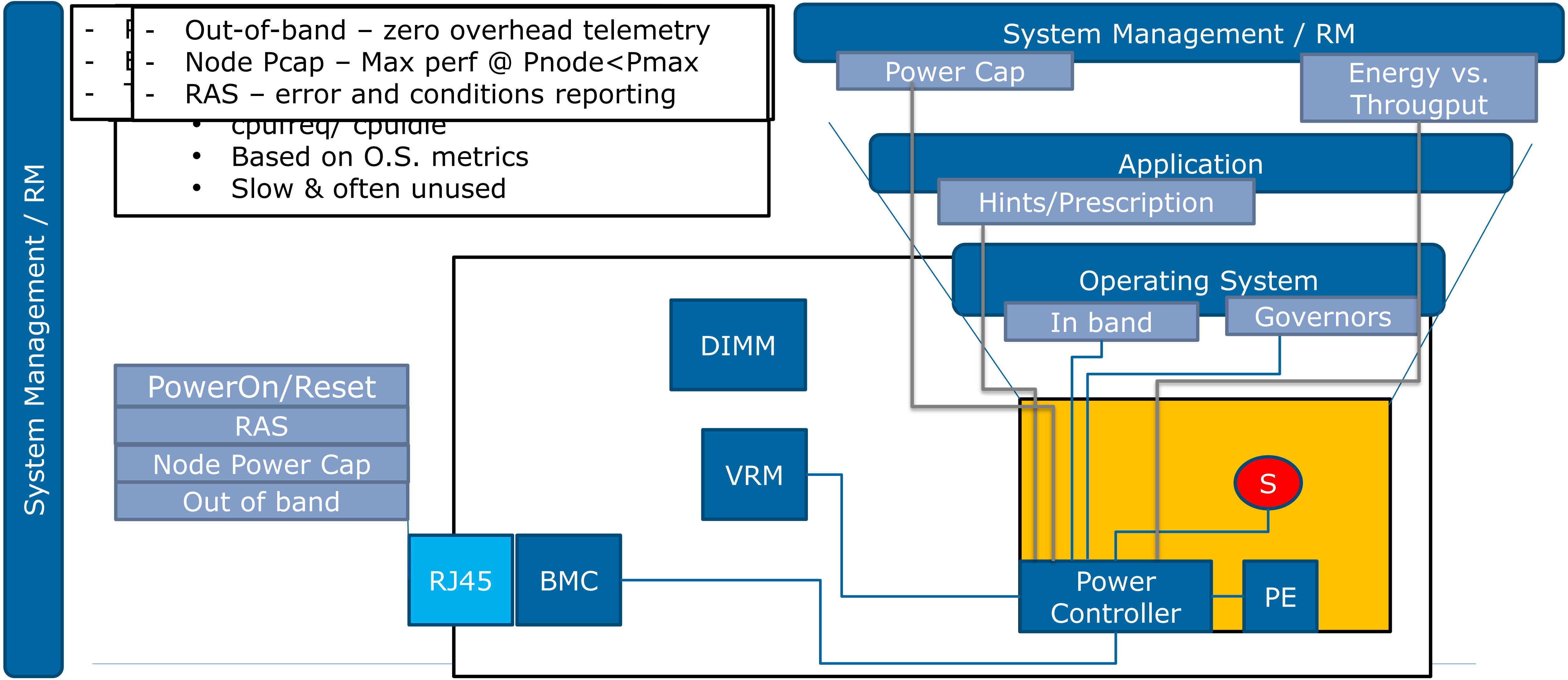
IBM ISSCC25 8.1

PM: System-Level View

- F
- E
- T

- Out-of-band – zero overhead telemetry
- Node Pcap – Max perf @ Pnode<Pmax
- RAS – error and conditions reporting

- cpufreq/ cpuidle
- Based on O.S. metrics
- Slow & often unused

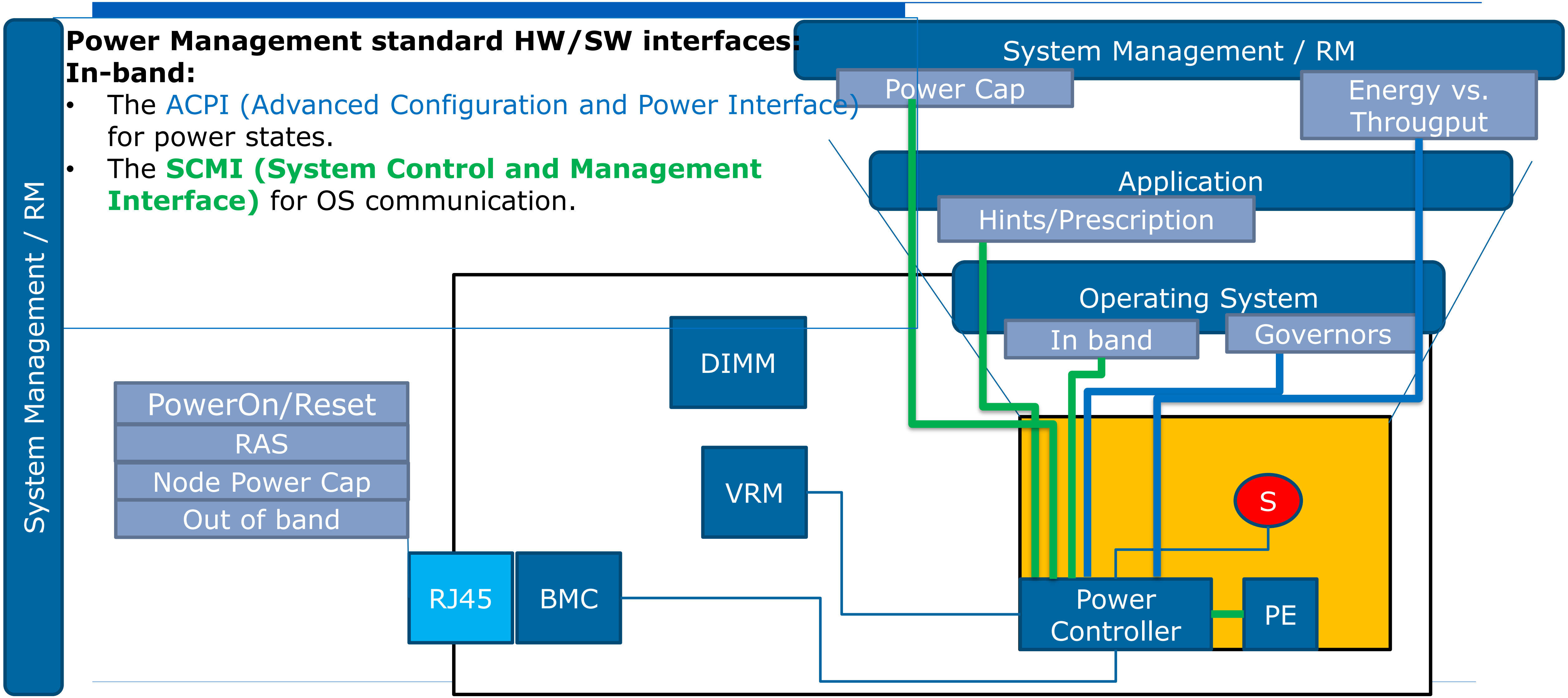


PM: System-Level View

Power Management standard HW/SW interfaces:

In-band:

- The ACPI (Advanced Configuration and Power Interface) for power states.
- The **SCMI (System Control and Management Interface)** for OS communication.



PM: System-Level View

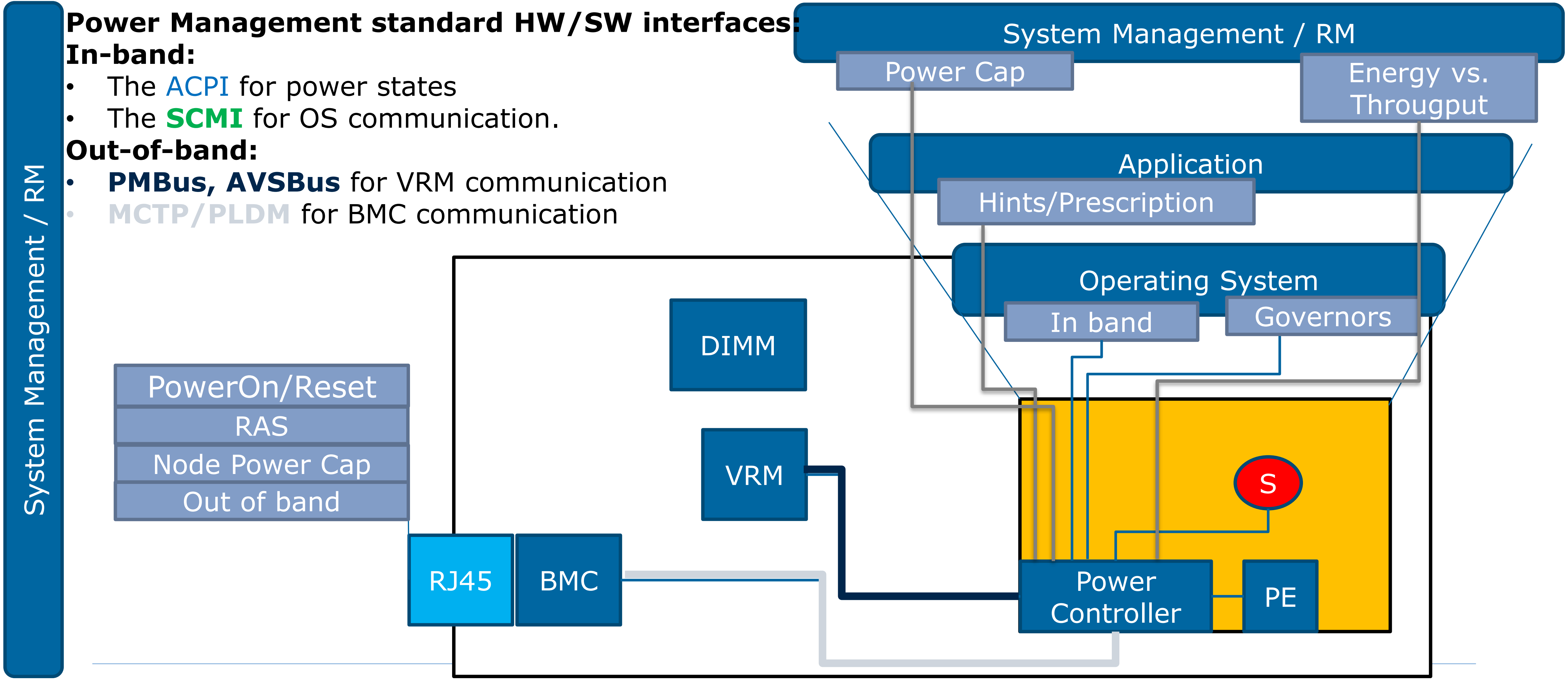
Power Management standard HW/SW interfaces:

In-band:

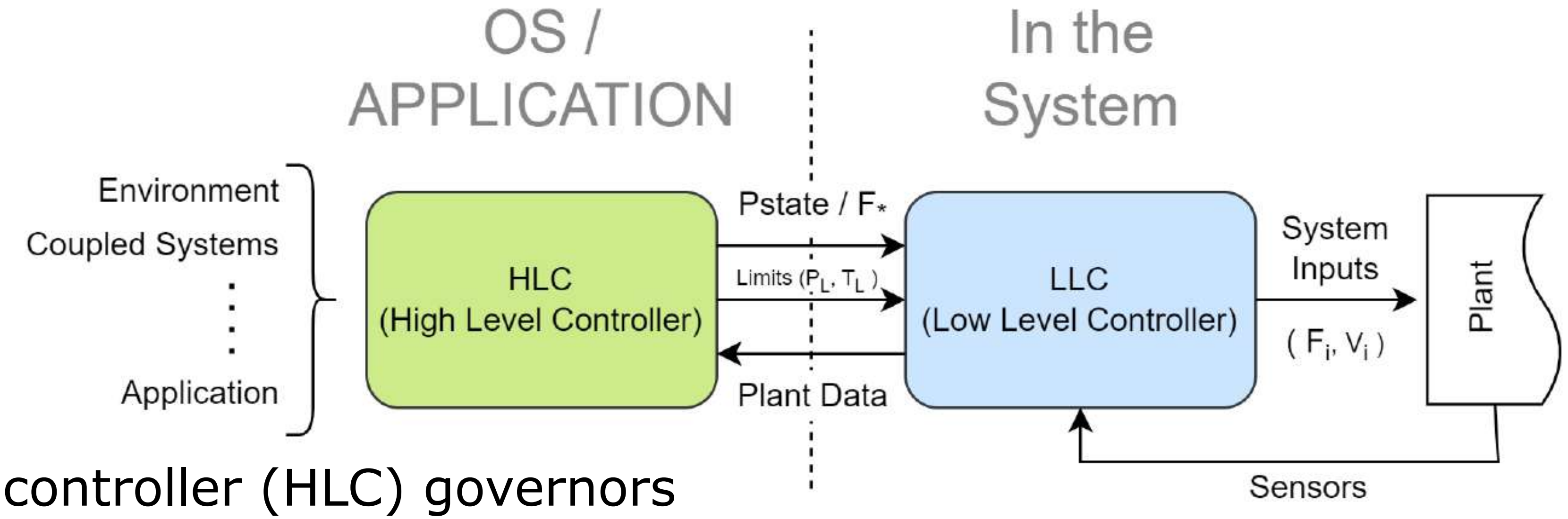
- The ACPI for power states
- The **SCMI** for OS communication.

Out-of-band:

- **PMBus, AVSBus** for VRM communication
- **MCTP/PLDM** for BMC communication

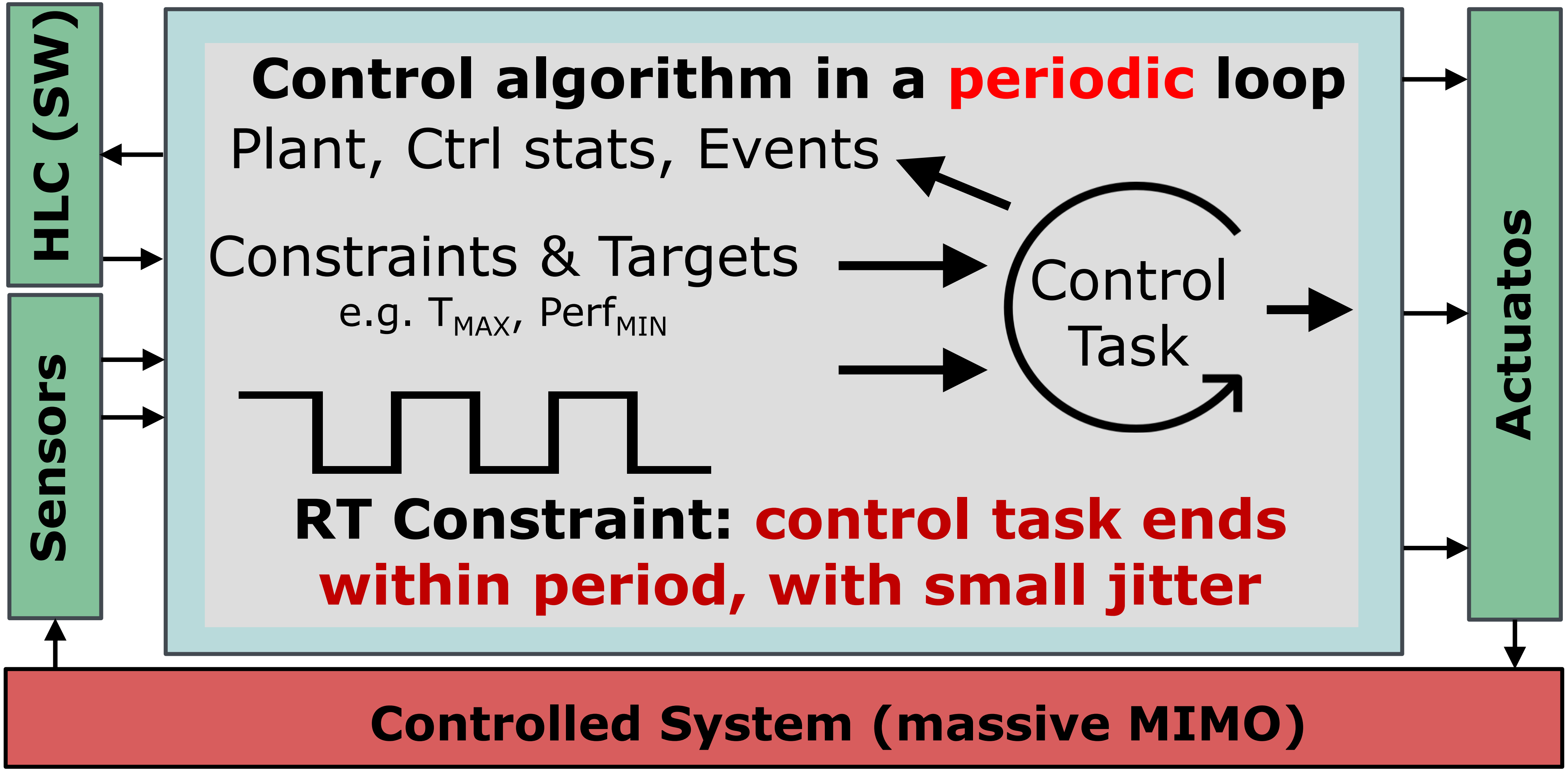


PM: System-Level View

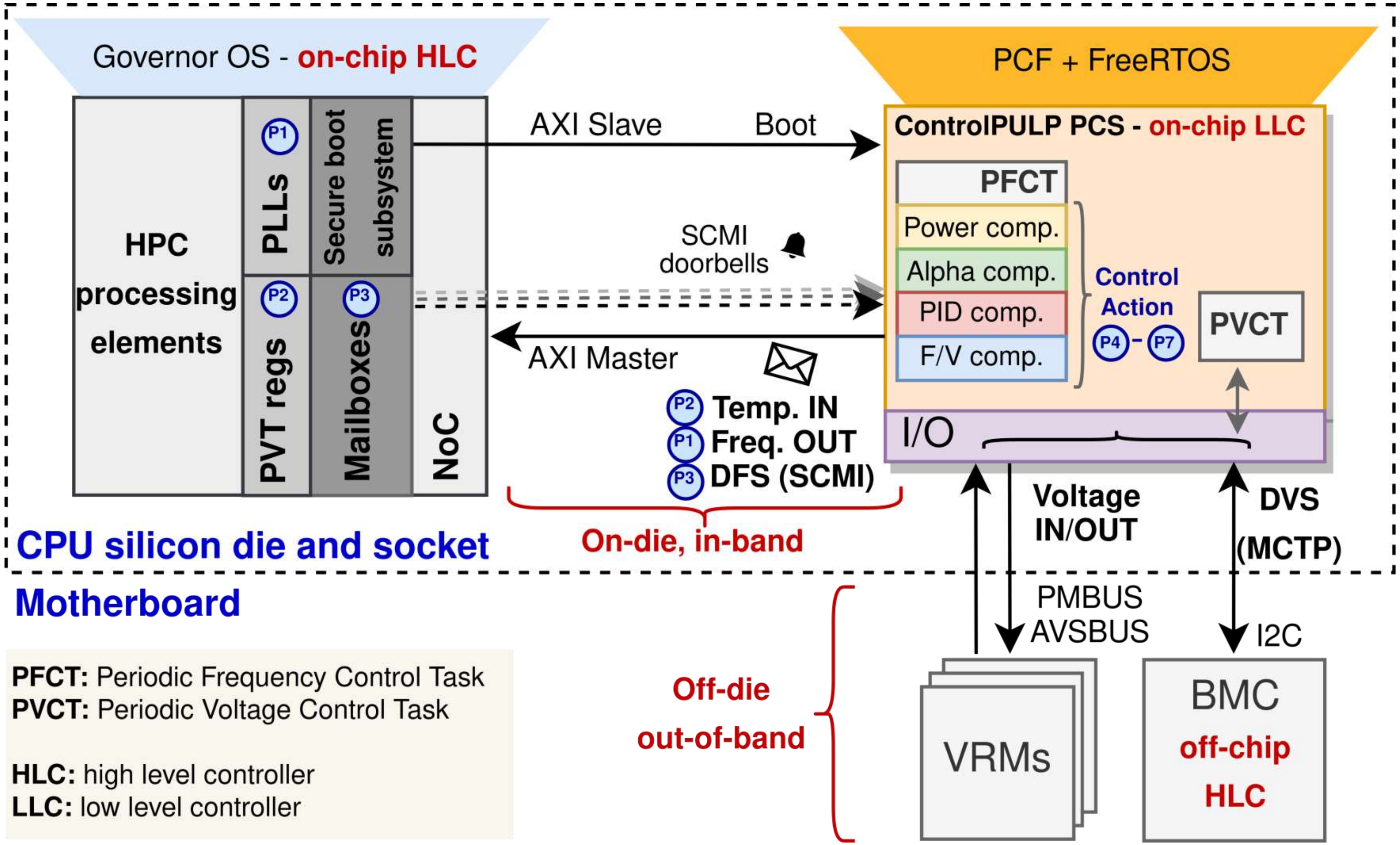


- □ High level controller (HLC) governors
  - Operating System (OS): e.g., OSPM in the Linux kernel
  - Baseboard Management Controller (BMC): e.g., Intel Aspeed off-chip regulator
- □ Low level controller (LLC) embedded platform
  - Power control subsystem (PCS): typically, a single-core MCU with < 1MiB memory footprint
  - Power control firmware (PCF): FW routine on top of the PCS

Focus on LLC: Function


HLC (SW)
Sensors
Control algorithm in a periodic loop
Plant, Ctrl stats, Events
Constraints & Targets
e.g. T_{MAX}, $Perf_{MIN}$
Control Task
RT Constraint: control task ends within period, with small jitter
Actuatos
Controlled System (massive MIMO)

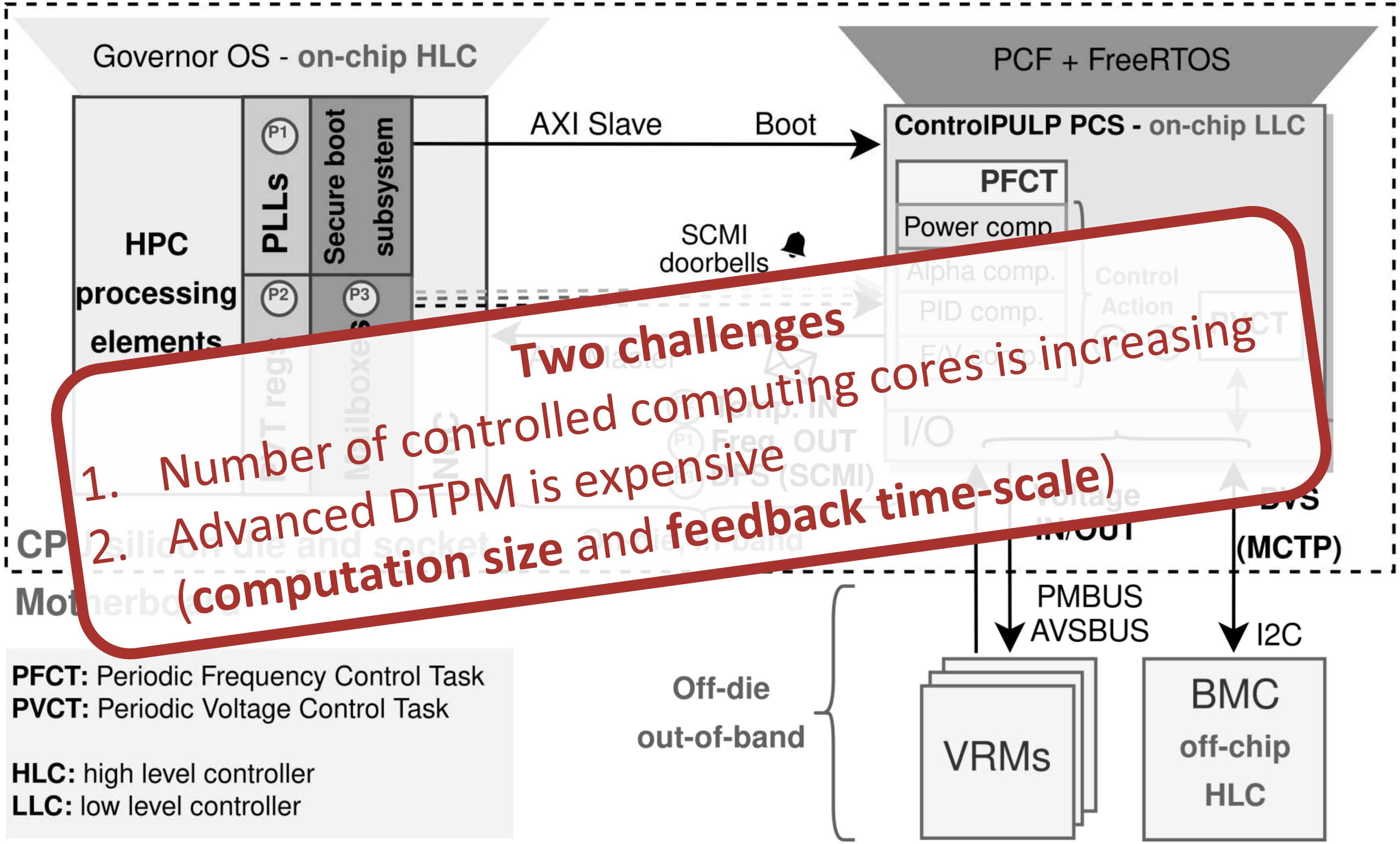
ControlPULP: Open-source Controller for PM



PFCT: Periodic Frequency Control Task
PVCT: Periodic Voltage Control Task

HLC: high level controller
LLC: low level controller

ControlPULP: Open-source controller for PM



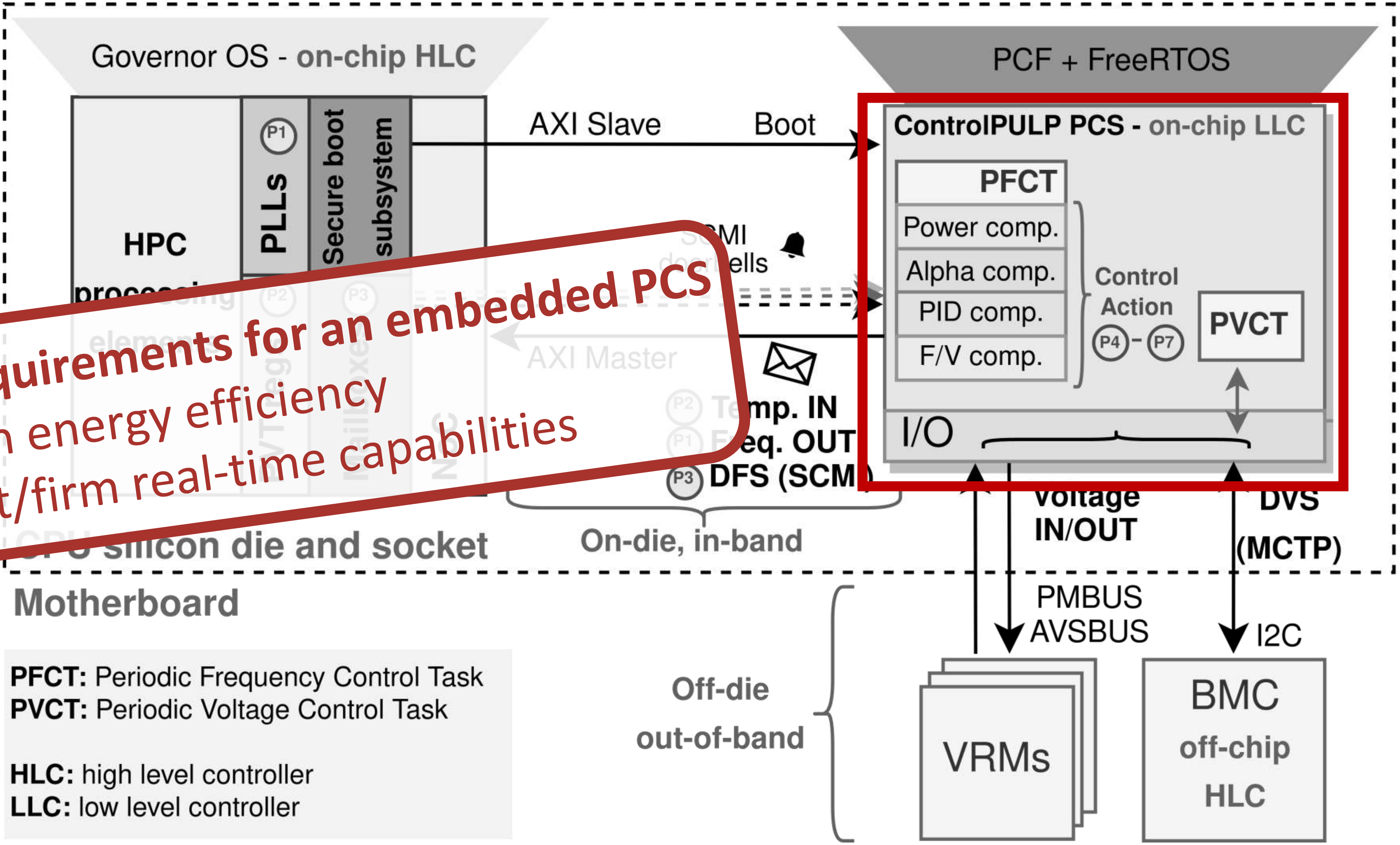
PFCT: Periodic Frequency Control Task
PVCT: Periodic Voltage Control Task

HLC: high level controller
LLC: low level controller

Two challenges

1. Number of controlled computing cores is increasing
2. Advanced DTPM is expensive (**computation size** and **feedback time-scale**)

ControlPULP: Open-source controller for PM



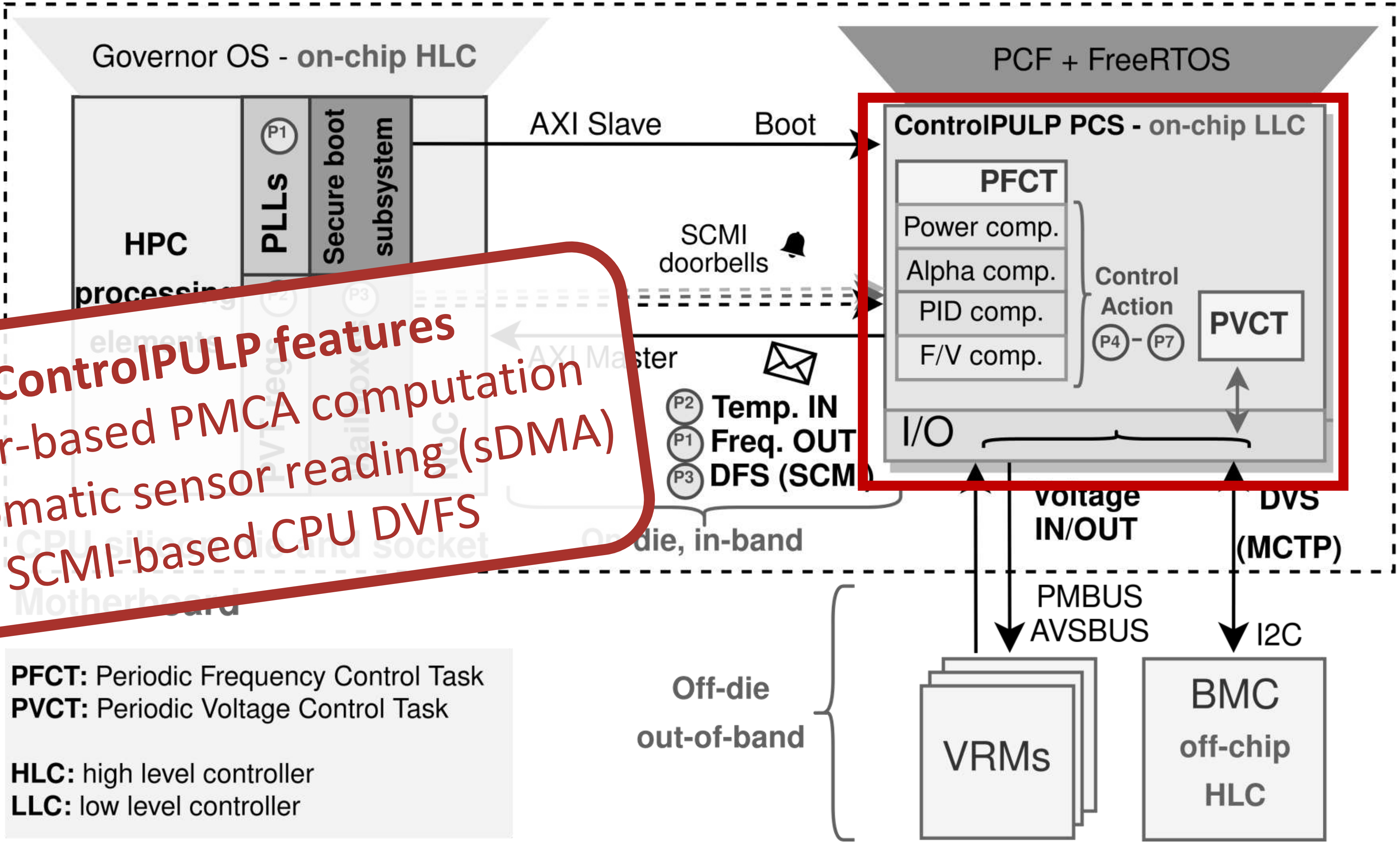
PFCT: Periodic Frequency Control Task
PVCT: Periodic Voltage Control Task

HLC: high level controller
LLC: low level controller

Two requirements for an embedded PCS

1. High energy efficiency
2. Soft/firm real-time capabilities

ControlPULP: Open-source controller for PM



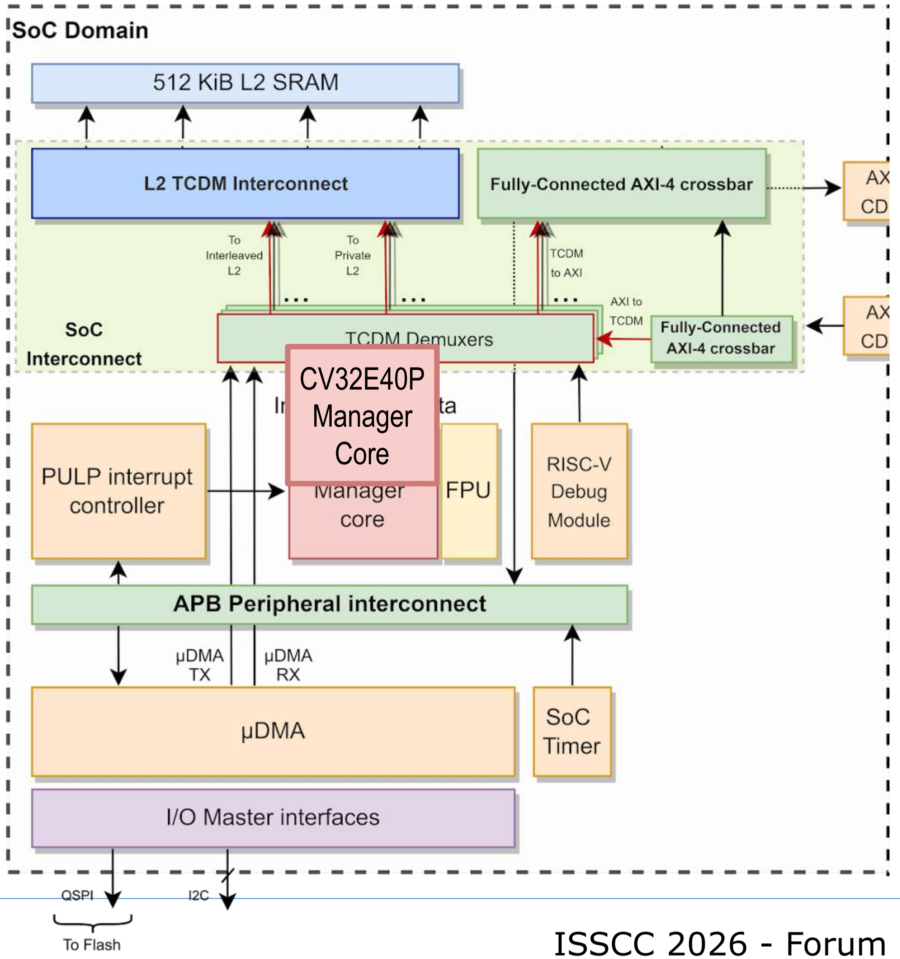
PFCT: Periodic Frequency Control Task
PVCT: Periodic Voltage Control Task

HLC: high level controller
LLC: low level controller

ControlPULP features

- Timer-based PMCA computation
- Automatic sensor reading (sDMA)
- Fast SCMI-based CPU DVFS

ControlPULP Architecture



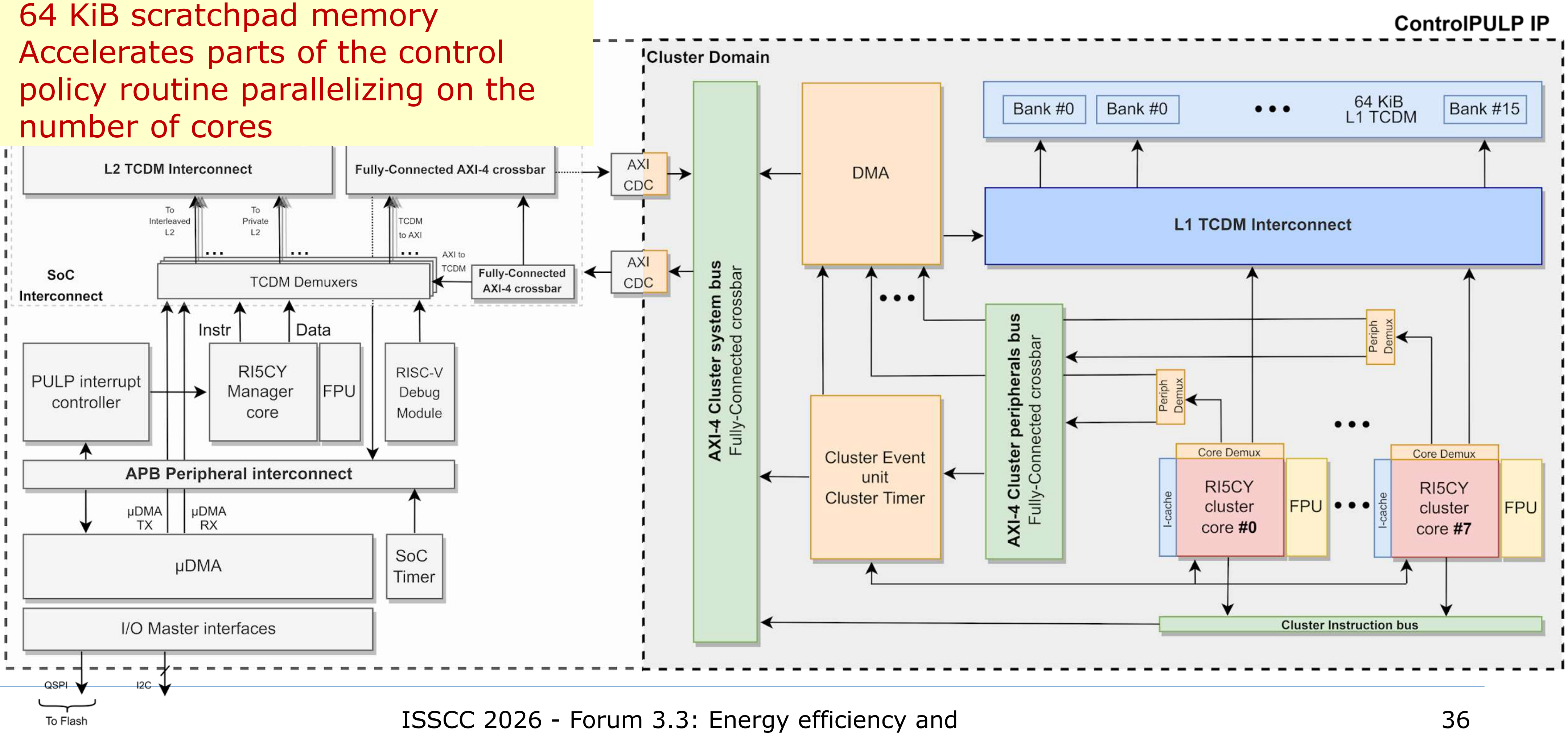
Single-core subsystem

- 32-bit CV32E40P
- 512 KiB scratchpad memory
- Executes the main control policy routine
- Offloads tasks to accelerator cluster

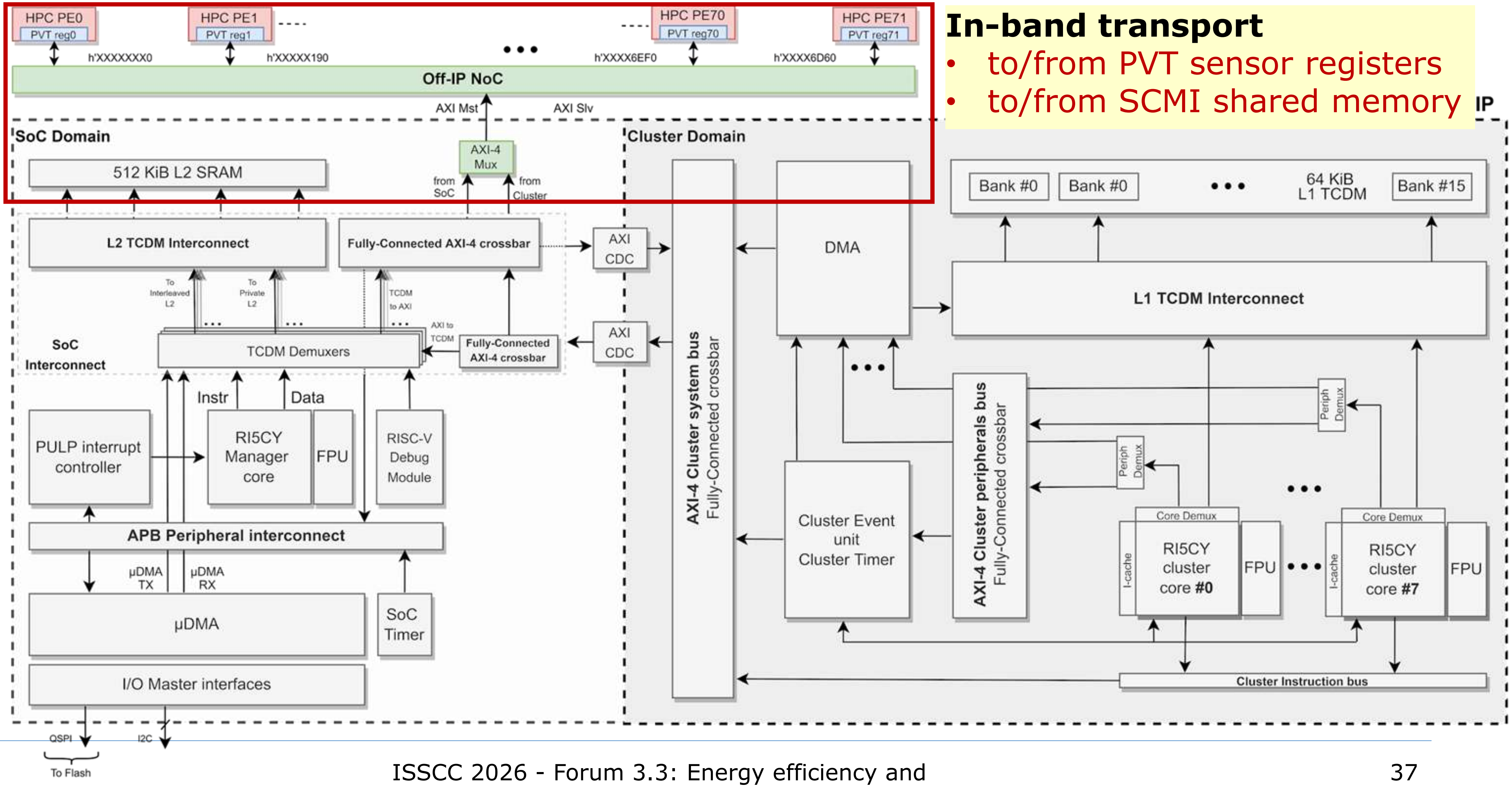
ControlPULP Architecture

Multi-core cluster subsystem

- 8 32-bit CV32E40P
- 64 KiB scratchpad memory
- Accelerates parts of the control policy routine parallelizing on the number of cores



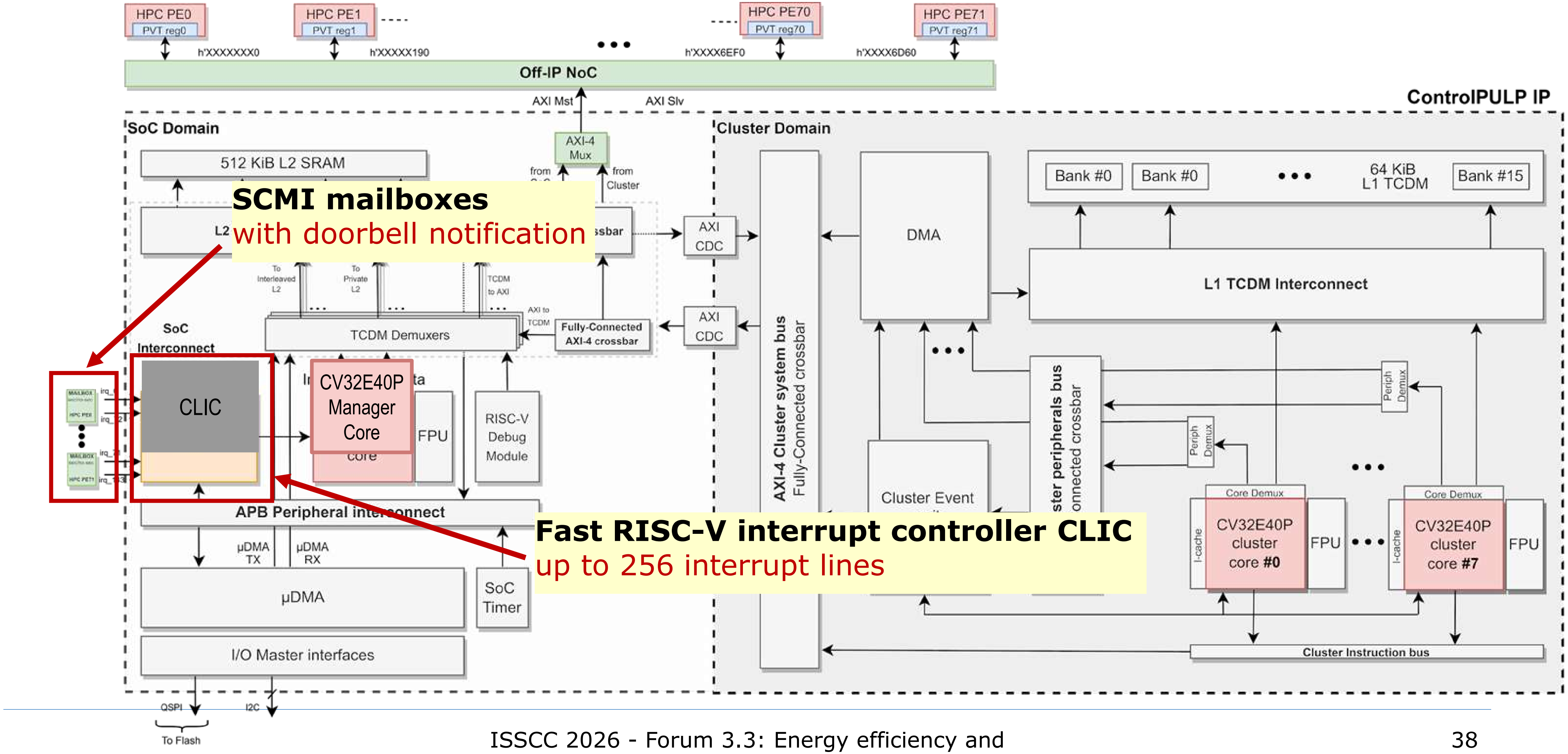
ControlPULP Architecture



In-band transport

- to/from PVT sensor registers
- to/from SCMI shared memory

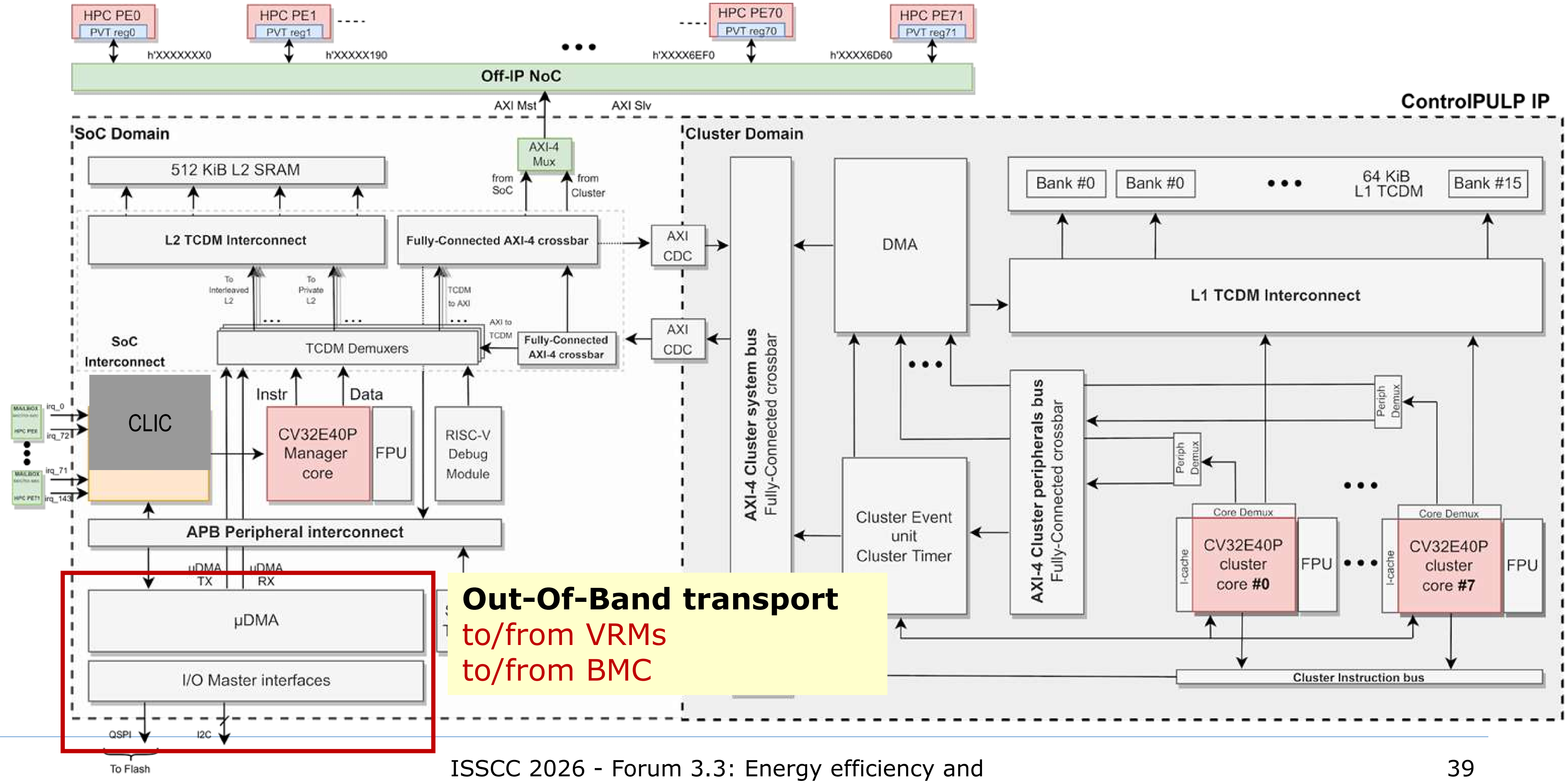
ControlPULP Architecture



SCMI mailboxes
with doorbell notification

Fast RISC-V interrupt controller CLIC
up to 256 interrupt lines

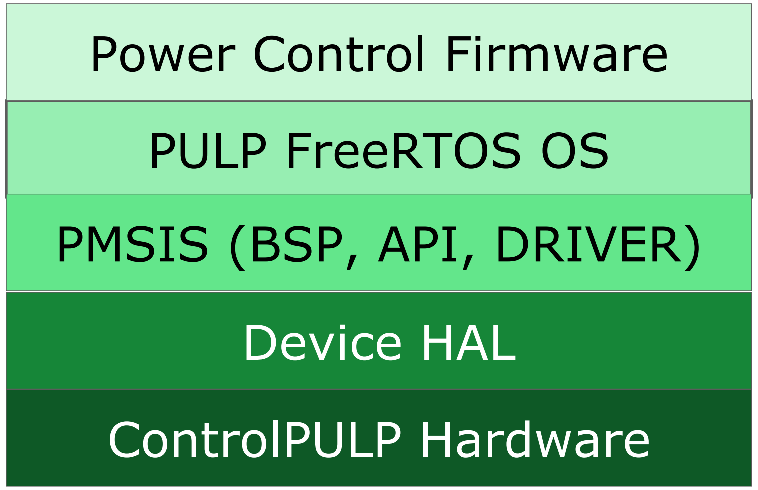
ControlPULP Architecture



Out-Of-Band transport
to/from VRMs
to/from BMC

ControlPULP: Software Stack

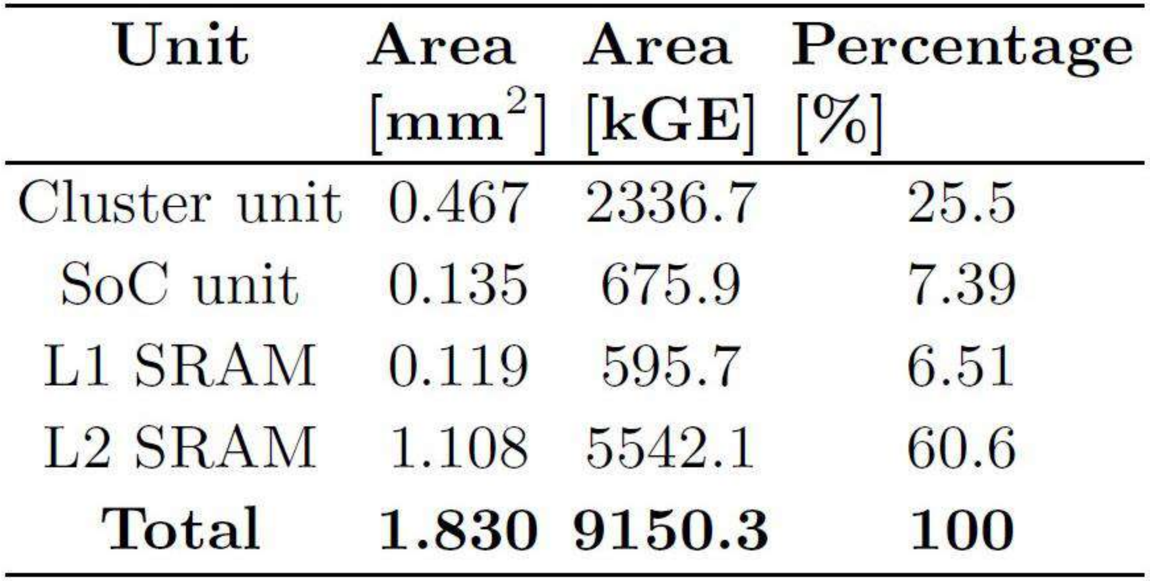
- □ Complete (application to hardware) software stack
- □ With a Real-time operation system, FreeRTOS



- □ Real-time OS (FreeRTOS) schedules tasks with periodic SysTick
- □ PCF (control policy) example tasks:
  - Periodic Frequency Control task: 2kHz, reads temperatures, computes DVFS, applies frequency on a per-PE basis
  - Periodic Voltage Control task: 8kHz, reads power rails consumption, applies voltage to groups of PEs
  - Communication Control task: SCMI, ACPI, MCTP communication

ControlPULP Implementation

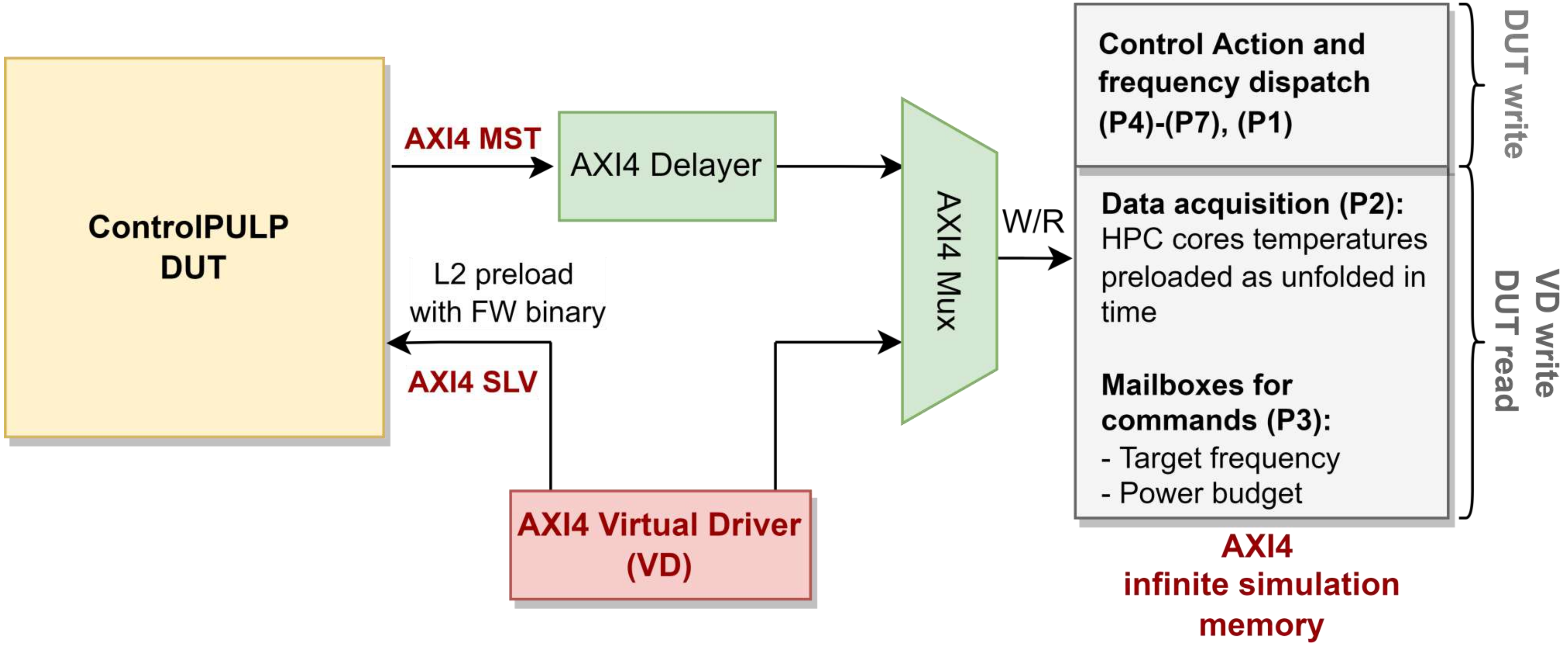
- GF22 synthesis: one manager core, one cluster, 512KiB + 64KiB @500 MHz, TT
- Total Area of 9.1 MGE
- Estimated < 1% of a HPC processor die in modern technology node



Unit	Area [mm^2]	Area [kGE]	Percentage [%]
Cluster unit	0.467	2336.7	25.5
SoC unit	0.135	675.9	7.39
L1 SRAM	0.119	595.7	6.51
L2 SRAM	1.108	5542.1	60.6
Total	**1.830**	**9150.3**	**100**

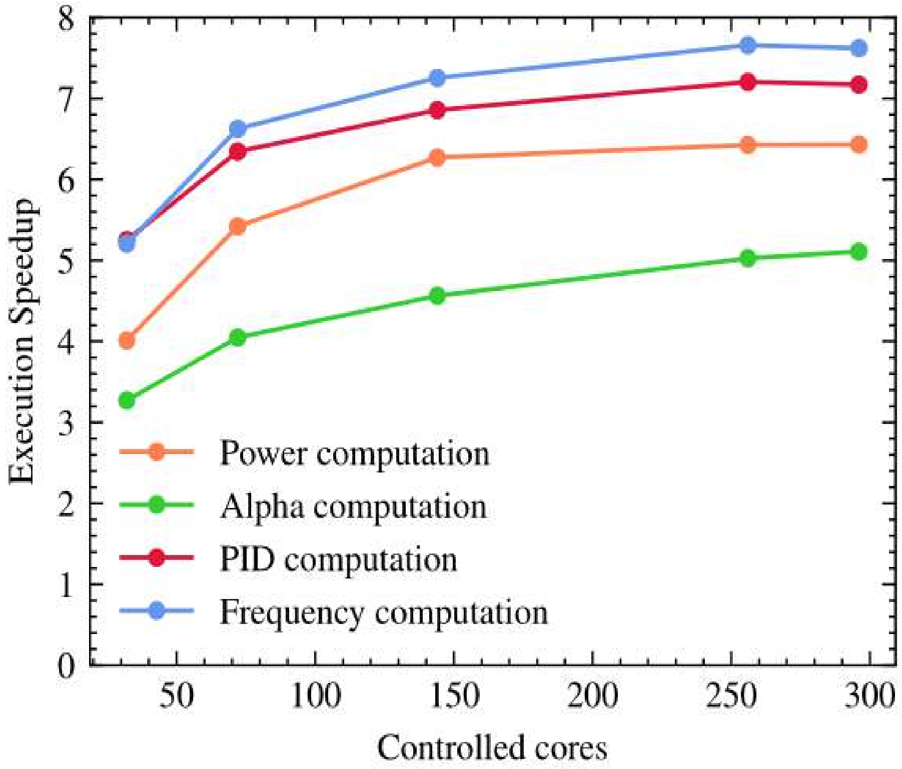
ControlPULP Performance

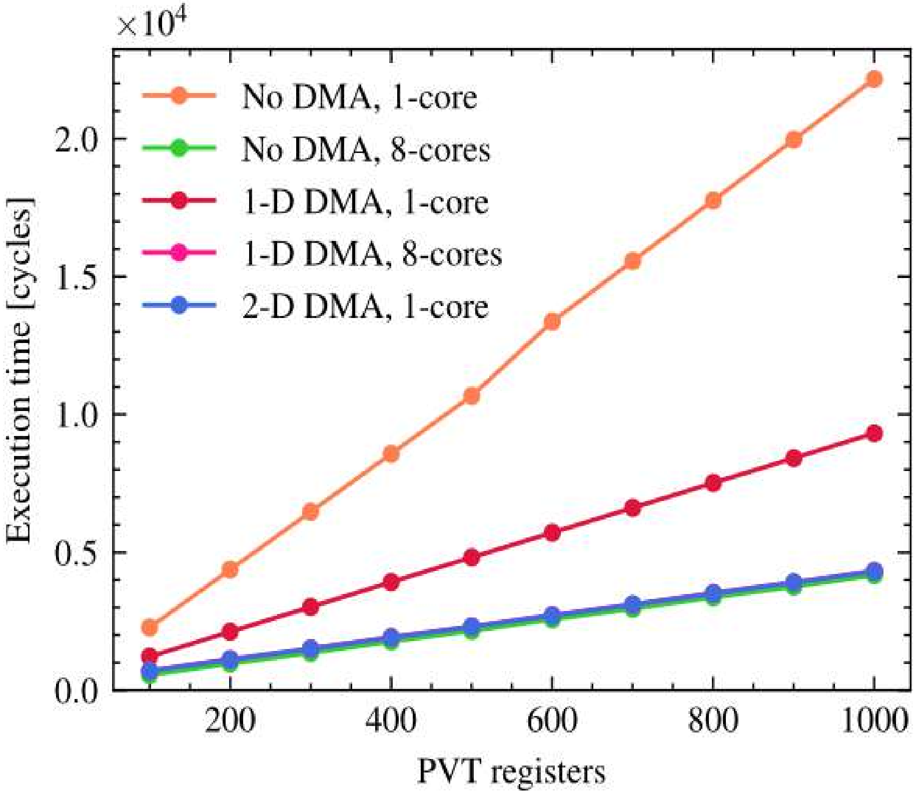
- □ NoC latency and controlled system are modeled in a testbench environment
- □ Evaluate: multi-core speedup (performance) and interrupt handling reactiveness (latency)


ControlPULP DUT
AXI4 MST
AXI4 Delayer
L2 preload with FW binary
AXI4 SLV
AXI4 Virtual Driver (VD)
AXI4 Mux
W/R
Control Action and frequency dispatch (P4)-(P7), (P1)
Data acquisition (P2): HPC cores temperatures preloaded as unfolded in time
Mailboxes for commands (P3):
- Target frequency
- Power budget
DUT write
VD write
DUT read
AXI4 infinite simulation memory

ControlPULP Performance

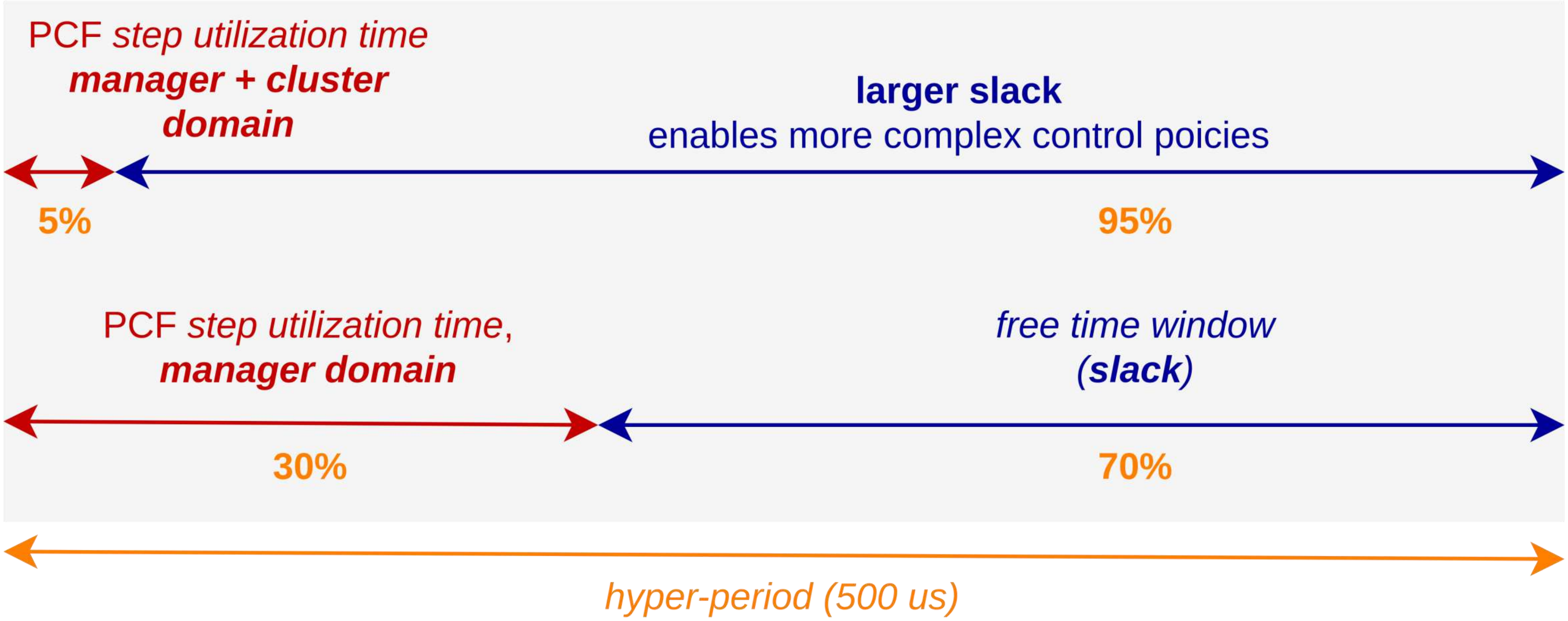
- Programmable accelerator and DMA driven PCF speedup:
  about 5x faster than single-core execution for 72 controlled cores (massive MIMO)





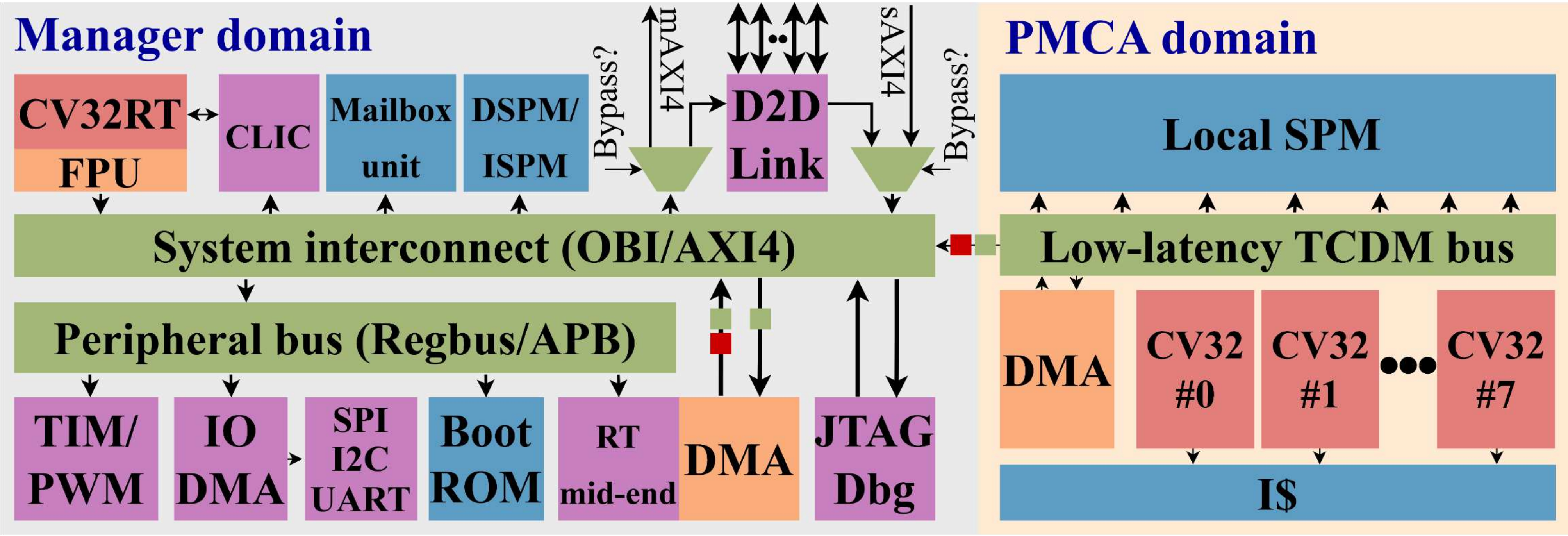
Why Programmable PM Accelerarator?

- □ Programmable accelerator (cluster) advantage for real-time scenarios

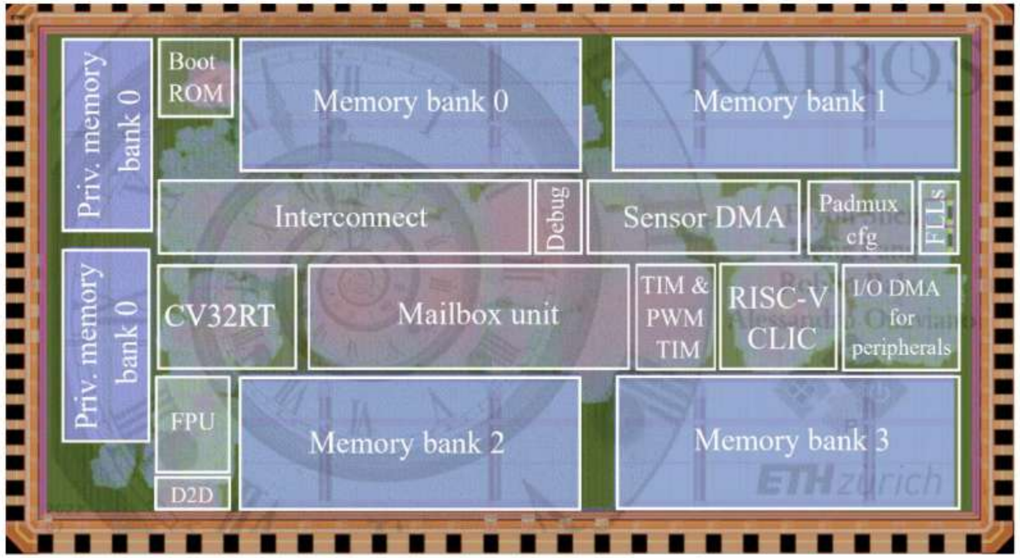

PCF *step utilization time*
manager + cluster domain
larger slack
enables more complex control poicies
5%
95%
PCF *step utilization time*,
manager domain
free time window
*(**slack**)*
30%
70%
hyper-period (500 us)

From single to multi Die: ControlPULPlet

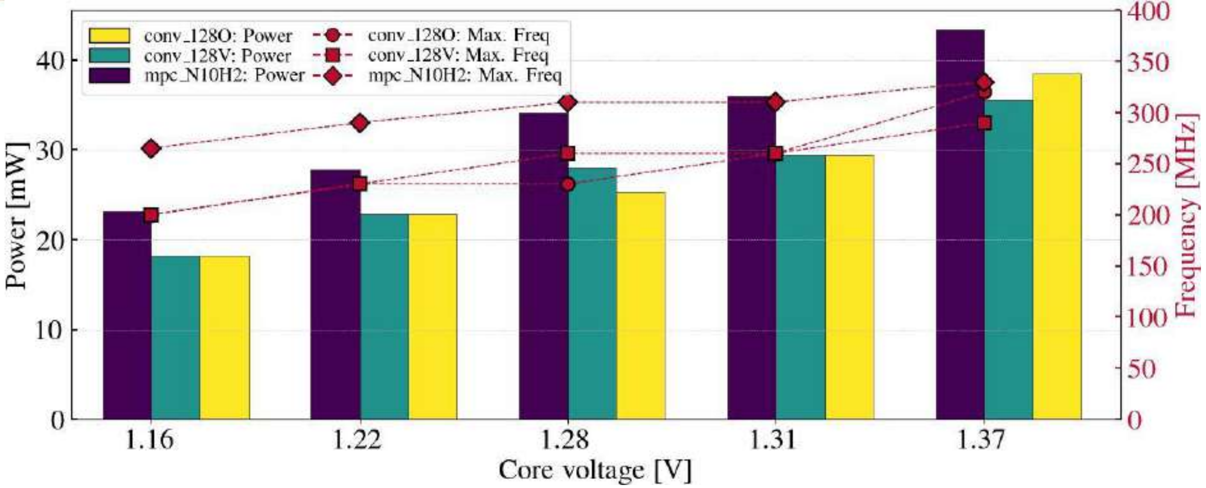
- □ CV32RT: fast interrupt handling in RISC-V
- □ D2D (die-to-die) connection for chiplet-based power control
- □ Scatter-gather DMA for N-dimensional periodic sensor reading
- □ 64x mailboxes for fast message exchange (e.g., SCMI)



Kairos, a Proxy in TSMC65 CMOS



ISA	rv32imafcxpulpv3xfastirq
Core	CV32RT [12]
On-chip SPM	448 KiB
Technology	TSMC65
Chip area	7.2 mm^2
V_{DD} core/V_{DD} IO	1.2 V/2.5 V
Frequency range	20 MHz to 380 MHz
Power envelope	$<$ 45 mW



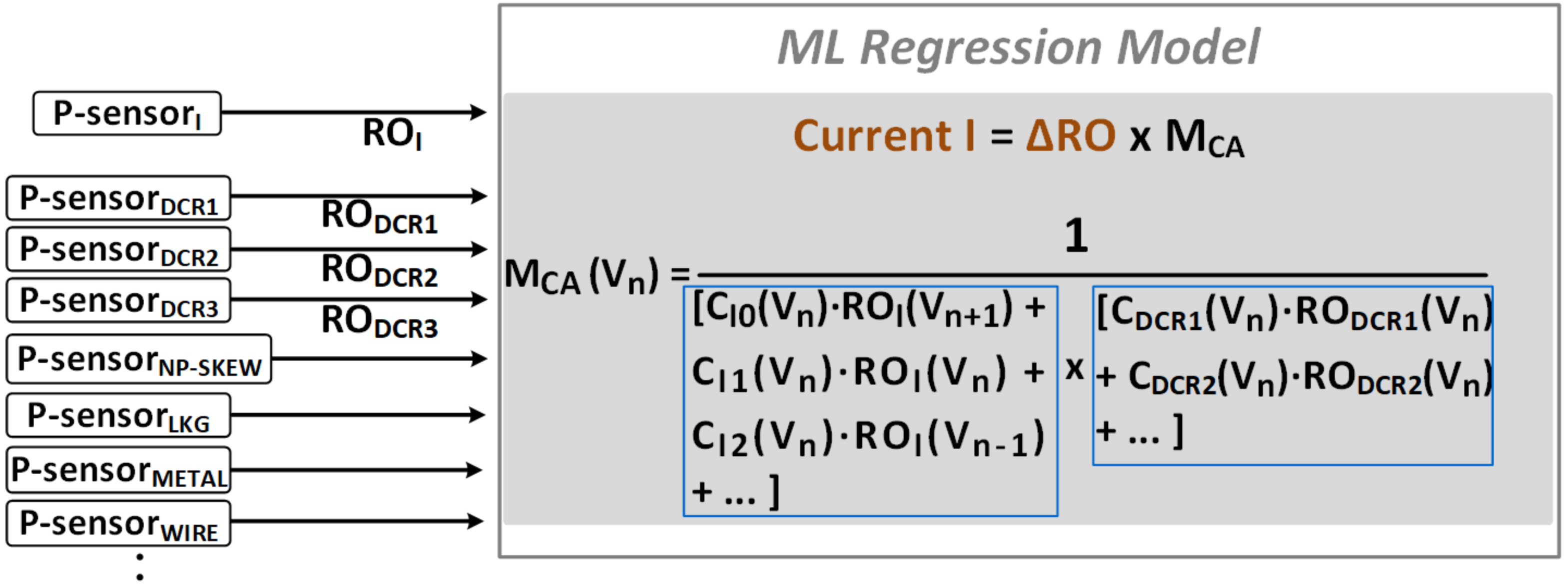
A. Ottaviano et al. TVLSI25

Outline

- □ Boosting efficiency for AI workloads
- □ Managing idleness and heterogeneity in accelerated systems
- □ Using AI for managing AI
- □ Conclusions, future perspectives

ML Based learning for Non-linear Sensors

Current sensing using Count-to-Ampere Models (MCA) based on Ring-oscillator sensors



Current I = ΔRO x M_{CA}

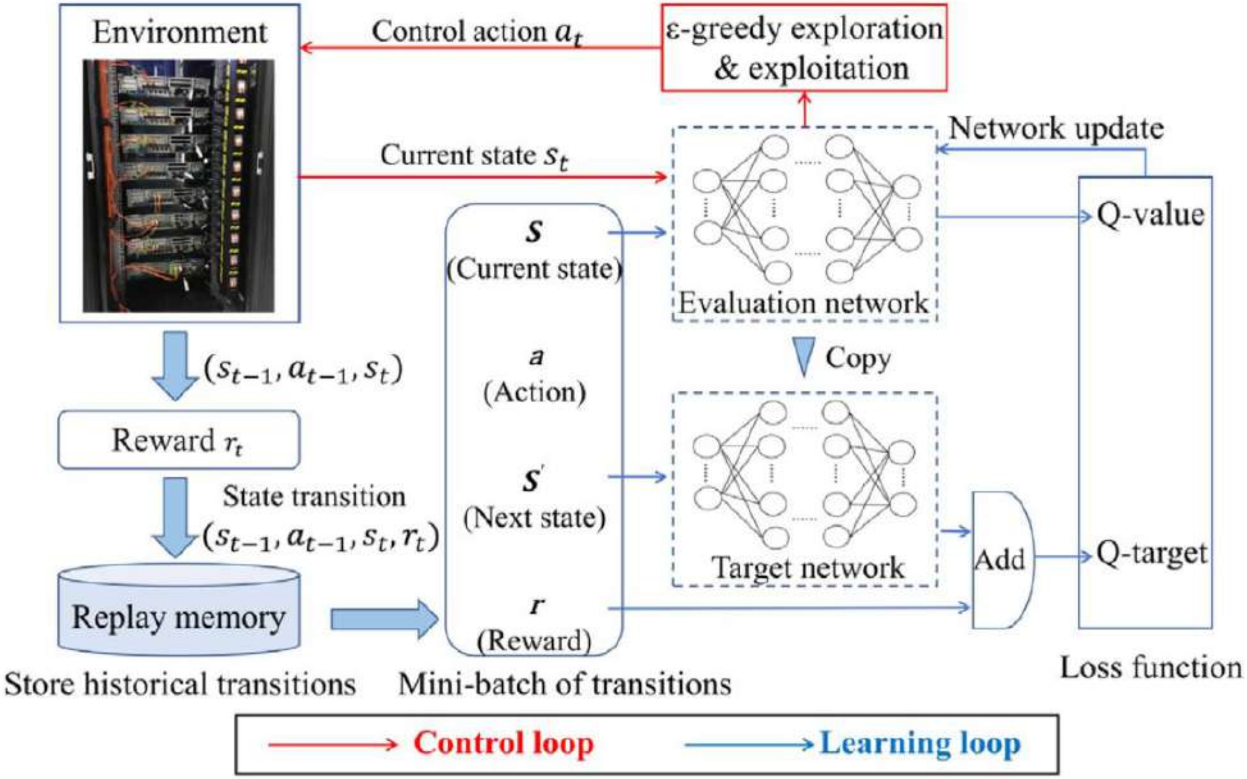
$$M_{CA}(V_n) = \frac{1}{\boxed{\begin{array}{l}[C_{I0}(V_n)\cdot RO_I(V_{n+1}) + \\ C_{I1}(V_n)\cdot RO_I(V_n) + \\ C_{I2}(V_n)\cdot RO_I(V_{n-1}) \\ + \ldots]\end{array}} \times \boxed{\begin{array}{l}[C_{DCR1}(V_n)\cdot RO_{DCR1}(V_n) \\ + C_{DCR2}(V_n)\cdot RO_{DCR2}(V_n) \\ + \ldots]\end{array}}}$$

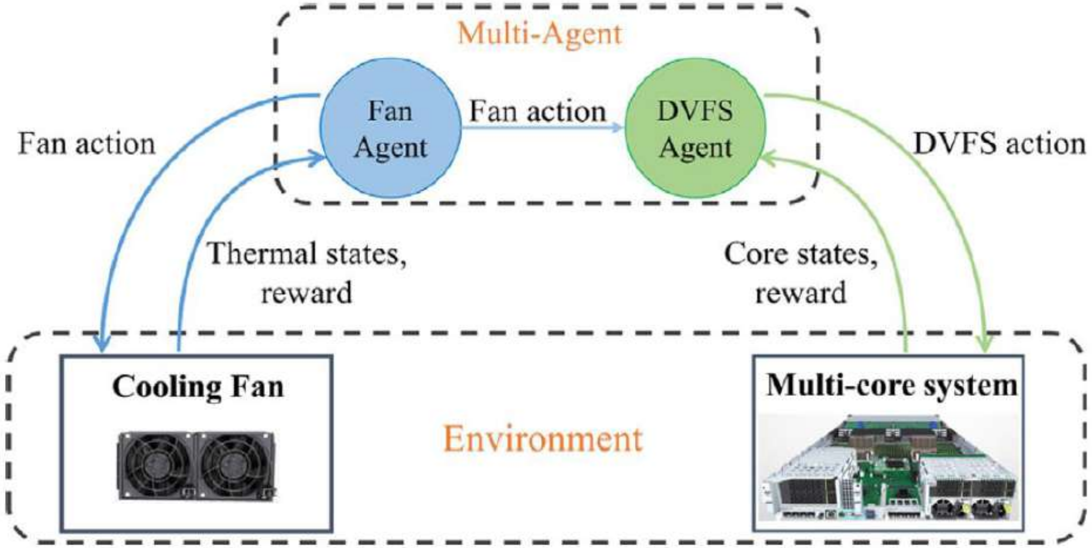
Mediatek ISSCC25 8.1

- □ ML calibration model leverages silicon data to refine the MCA
  - ■ Scope-measured current I as golden to train the coefficients of ΔRO (RO) from the I-sensor (P-sensors)
  - ■ Multi-scenarios, including generic benchmarks like Geekbenchv6, Antutu, and SpecInt2k6 etc.
- □ Good tolerances against voltage and temperature fluctuations
- □ Statistical ML used to eelect the most relevant RO terms (i.e., P-sensors) to simplify the coefficient terms of the model
- □ Device-specific fine tuning also possible

Advanced AI-based Policies

□ Control law can be learned (e.g. RL) offline and fine-tuned online





www.sciencedirect.com/science/article/abs/pii/S2210537924000222

Outline

- □ Boosting efficiency for AI workloads
- □ Managing idleness and heterogeneity in accelerated systems
- □ Using AI for managing AI
- □ Conclusions, future perspectives

Summing Up & Future Perspectives

- PM is more critical than ever for AI hardware
  - Thermal bottleneck
  - Power delivery bottleneck
  - Impacts the AI bottom line (cost per token)
- Key requirements for PM in AI hardware
  - Massive MIMO control (fine-grained & scaled-up)
  - Heterogeneous (processors, accelerators, PIM, PIN…)
  - Multi-die (2.5D → 3.5D)
- Real-time, (multi-die) scalable and programmable PM controller architecture is needed (open platform is advisable)
- AI for power managing AI is a major trend
  - Learn complex non-linear functions for sensing and actuation
  - Learn optimal policies (e.g. RL)
  - Online learning for fine-tuning device-specific calibration