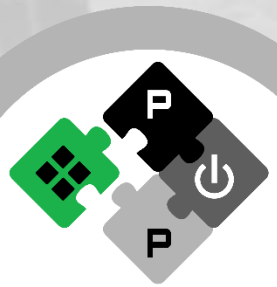


Energy efficient computing from Exascale to MicroWatts: The RISC-V playground

Bologna, Euroexascale Workshop *20.01.2020*

Luca Benini[1,2]

[1]*Department of Electrical, Electronic and Information Engineering*

ETH zürich

[2]*Integrated Systems Laboratory*

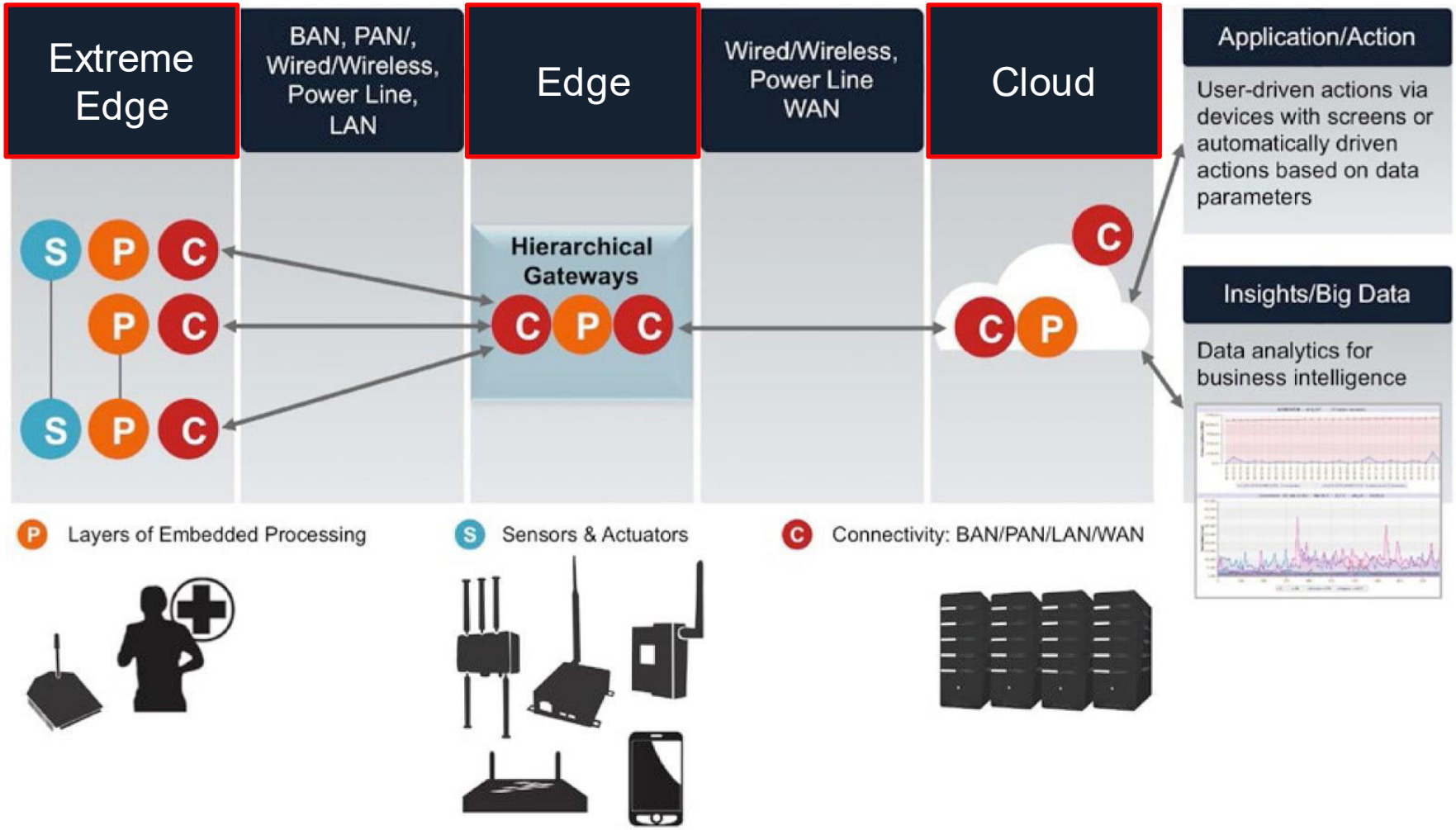
erc European Research Council

European Commission | Horizon 2020 European Union funding for Research & Innovation

nano-tera.ch

FNSNF Fonds national suisse Schweizerischer Nationalfonds Fondo nazionale svizzero Swiss National Science Foundation

Computing Continuum



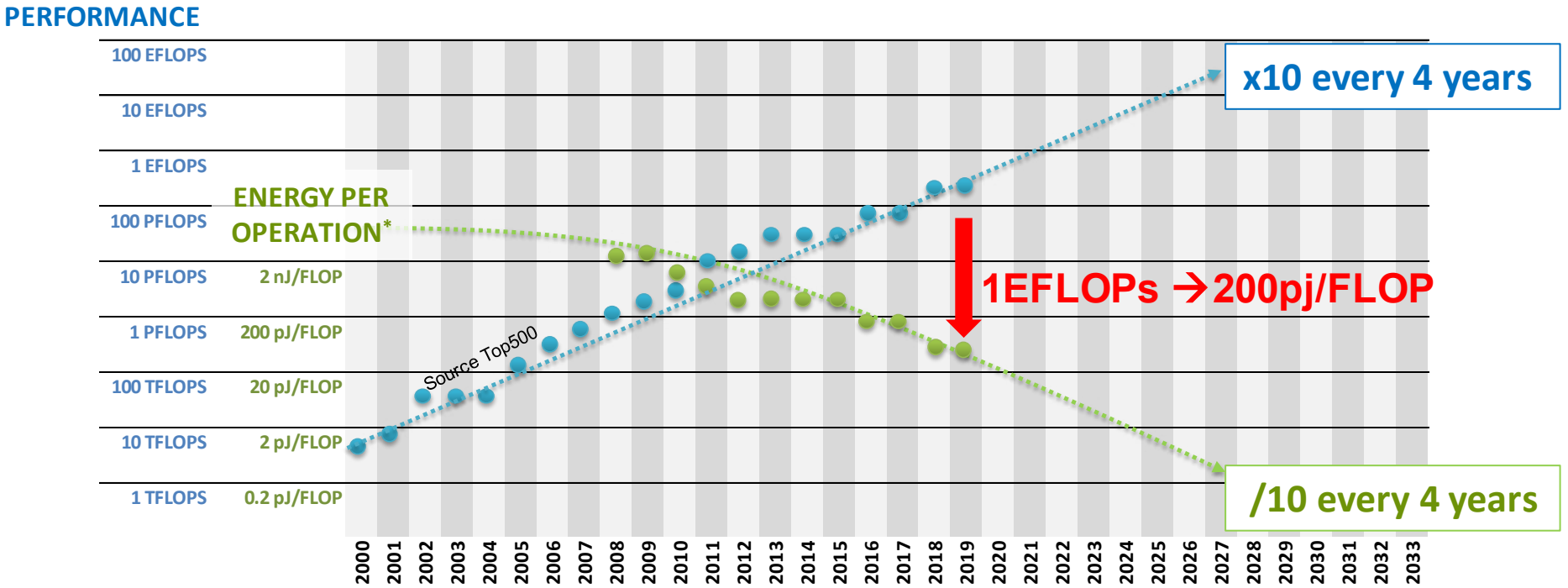
Near-Sensor Processing **Embedded Processing** **Cloud processing**

PULP

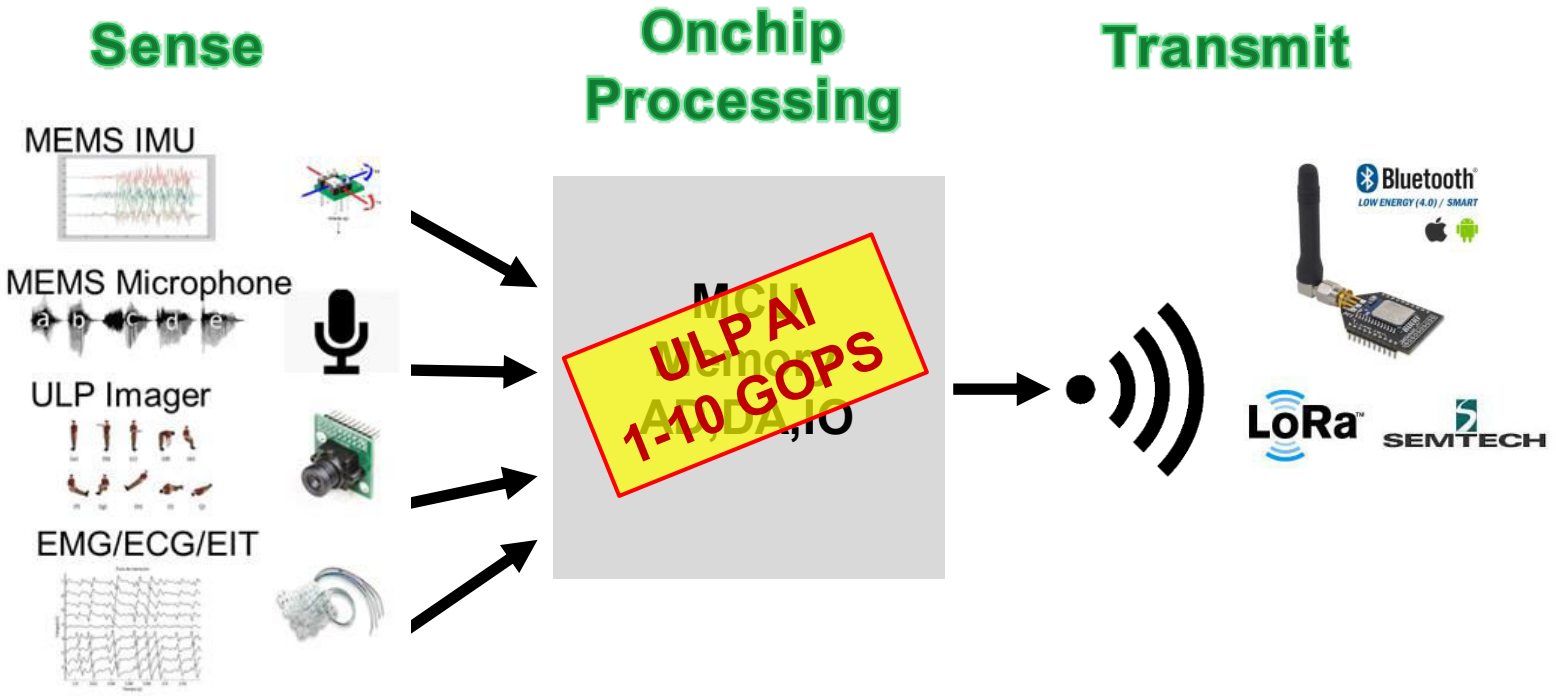
Energy efficiency challenge: Exascale

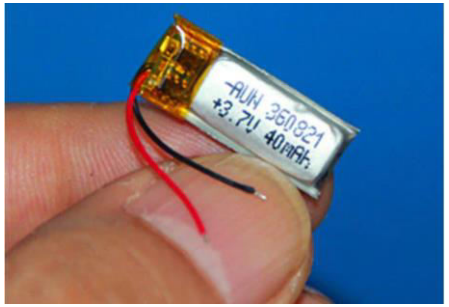
HPC is now power-bound→ need 10x energy efficiency improvement every 4 years

*20MWatt supercomputer: Performance & EnOP



Energy efficiency Challenge: Extreme Edge





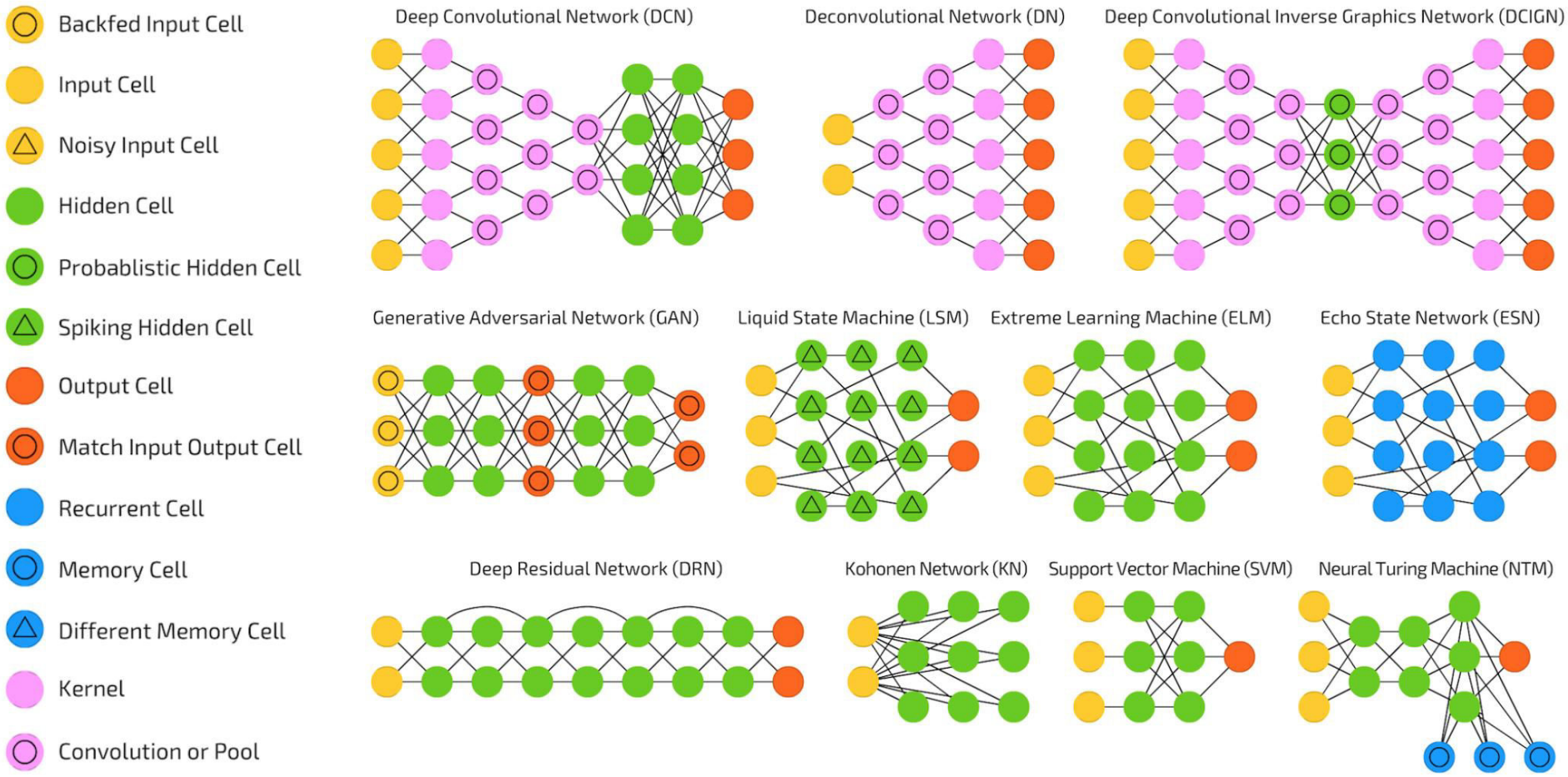
100mA/h 1month
1% duty cycle

10uA sleep
10mA active

1-10pJ/OP

PULP

Flexibility needed!



asimovinstitute.org/neural-network-zoo

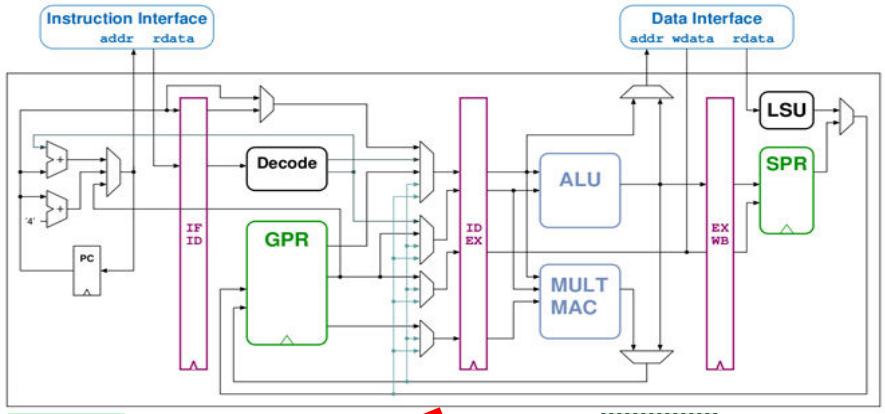
+ FFT, PCA, Mat-inv,…

PULP

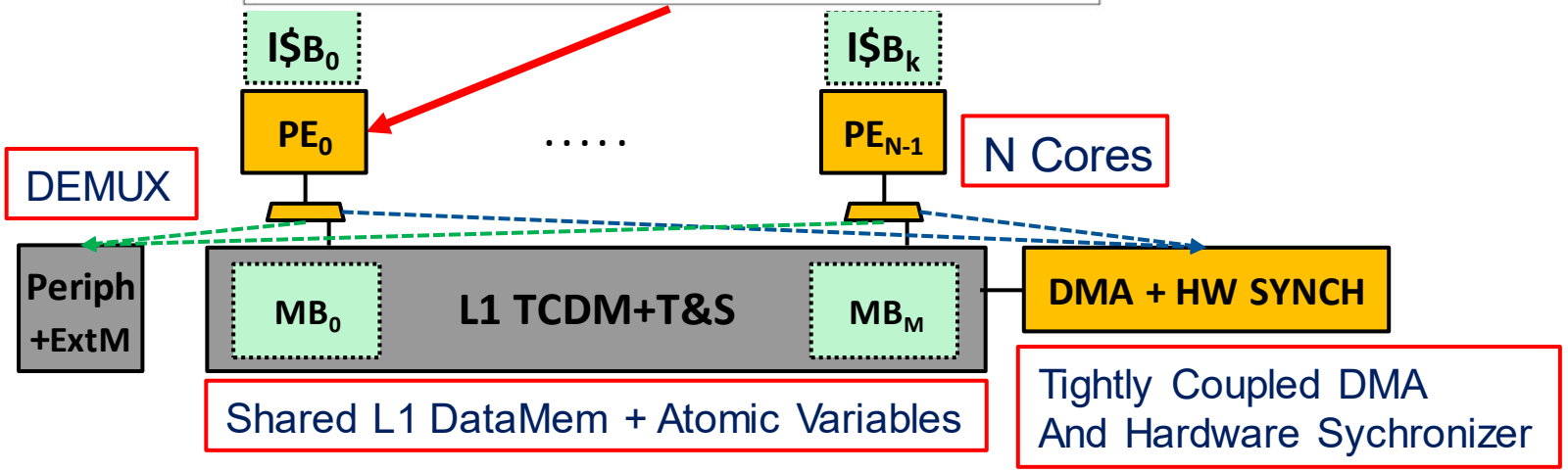
Near-Threshold Multiprocessing on an Open ISA

Need flexibility + energy efficiency ➡ Processor + Low Vdd + Parallel + ISA extensions

Shallow Pipeline
3 stages+1for LD/ST



RISC-V®



Need extensible ISA, Need full access to “deep” core interfaces, need to tune pipeline,
RV32IMC + **New, Open Microarchitecture + ISA Extensions**

PULP

Xpulp extensions

<32-bit precision → **SIMD2/4 → x2,4 efficiency & memory size**

Risc-V ISA is extensible *by construction* (great!)

V1 Baseline RISC-V RV32IMC

HW loops

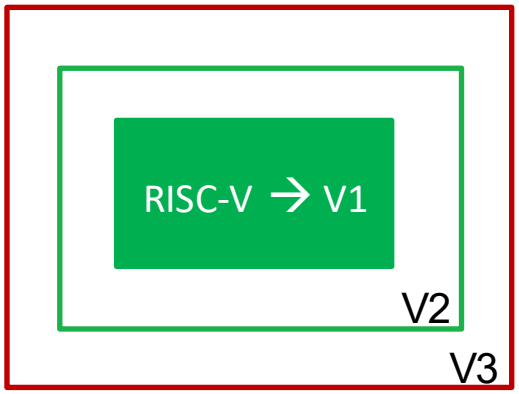
V2 Post modified Load/Store

Mac

V3 SIMD 2/4 + DotProduct + Shuffling

Bit manipulation unit

Lightweight fixed point



25KG → 40KG (1.6x)

RI5CY – are xPULP ISA Extensions (1.6x) worthwhile?

```
for (i = 0; i < 100; i++)
    d[i] = a[i] + b[i];
```

10x on 2d convolutions …YES!

Baseline

```
mv   x5, 0
mv   x4, 100
Lstart:
  lb   x2, 0(x10)
  lb   x3, 0(x11)
  addi x10,x10, 1
  addi x11,x11, 1
  add  x2, x3, x2
  sb   x2, 0(x12)
  addi x4, x4, -1
  addi x12,x12, 1
bne    x4, x5, Lstar
```

11 cycles/output

Auto-incr load/store

```
mv   x5, 0
mv   x4, 100
Lstart:
  lb   x2, 0(x10!)
  lb   x3, 0(x11!)
  addi x4, x4, -1
  add  x2, x3, x2
  sb   x2, 0(x12!)
bne    x4, x5, Lstart
```

8 cycles/output

HW Loop

```
lp.setupi 100, Lend
  lb   x2, 0(x10!)
  lb   x3, 0(x11!)
  add  x2, x3, x2
Lend: sb x2, 0(x12!)
```

5 cycles/output

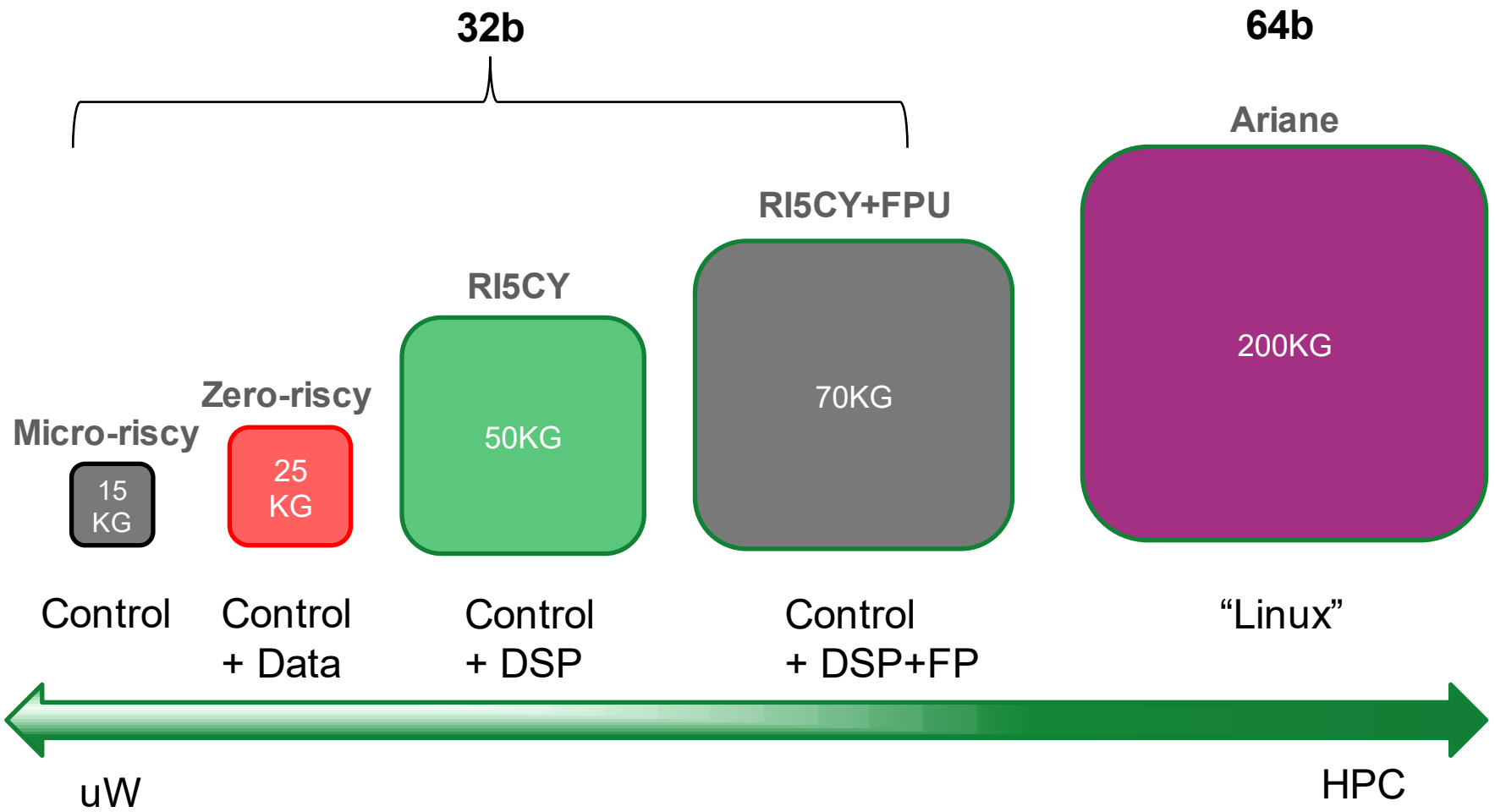
Packed-SIMD

```
lp.setupi 25, Lend
  lw   x2, 0(x10!)
  lw   x3, 0(x11!)
  pv.add.b x2, x3, x2
Lend: sw x2, 0(x12!)
```

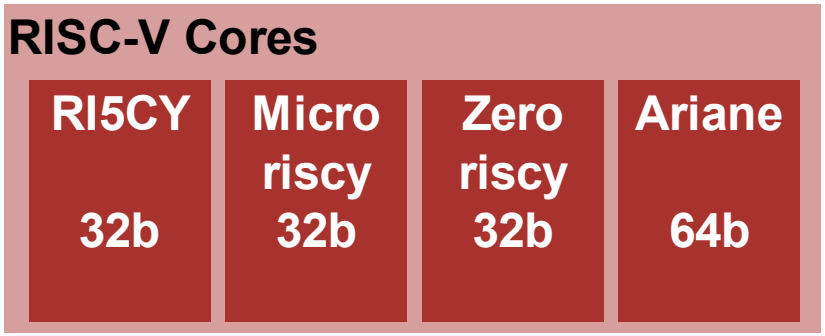
1,25 cycles/output

PULP

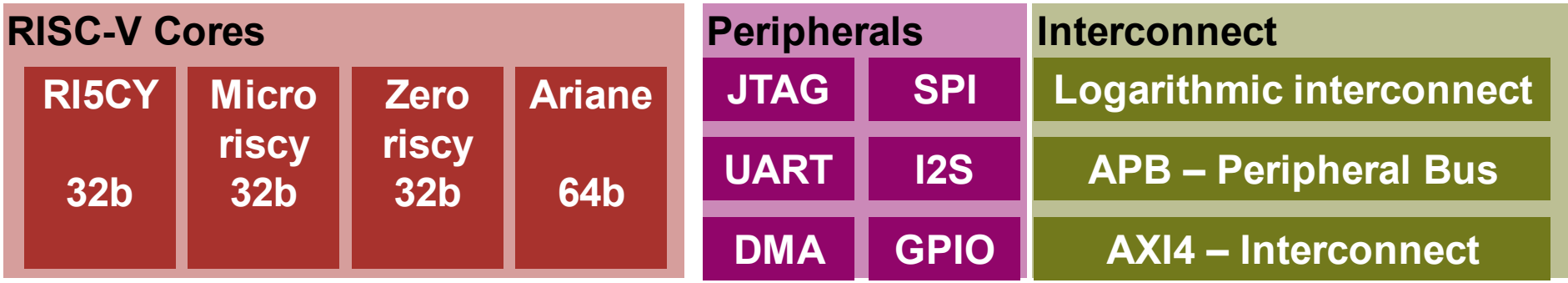
Need a “range of cores”

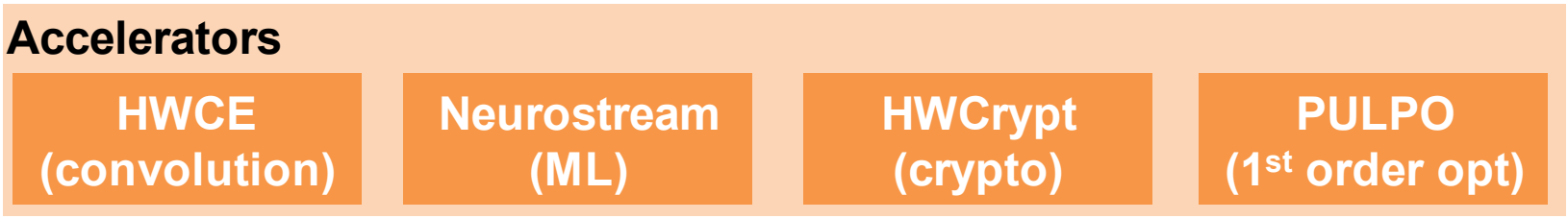


RISC-V cores


RISC-V Cores
RI5CY
32b
Micro
riscy
32b
Zero
riscy
32b
Ariane
64b

Cores+IOs+Intercos+Accelerators





All these components are combined into platforms

RISC-V Cores

- RI5CY 32b
- Micro riscy 32b
- Zero riscy 32b
- Ariane 64b

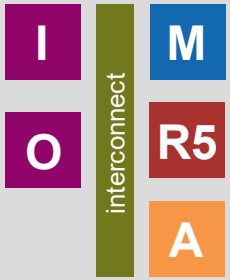
Peripherals

- JTAG
- SPI
- UART
- I2S
- DMA
- GPIO

Interconnect

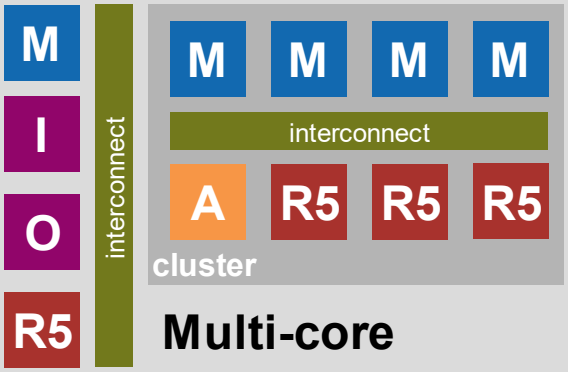
- Logarithmic interconnect
- APB – Peripheral Bus
- AXI4 – Interconnect

Platforms



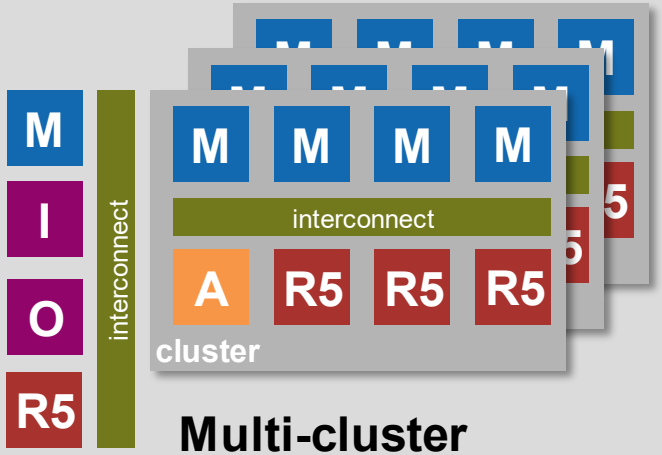
Single Core

- PULPino
- PULPissimo



Multi-core

- Fulmine
- Mr. Wolf



Multi-cluster

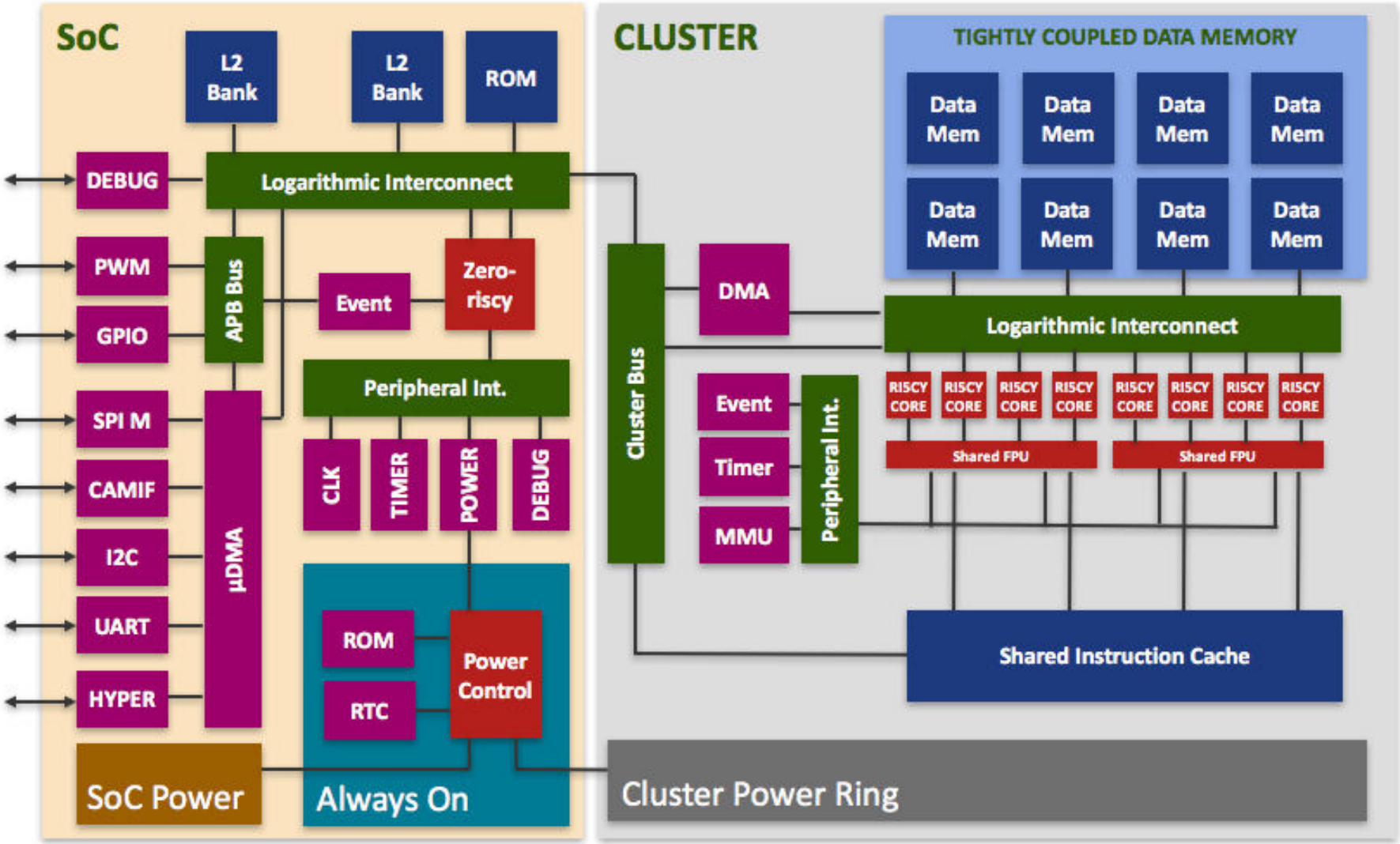
- Hero

uW → HPC

Accelerators

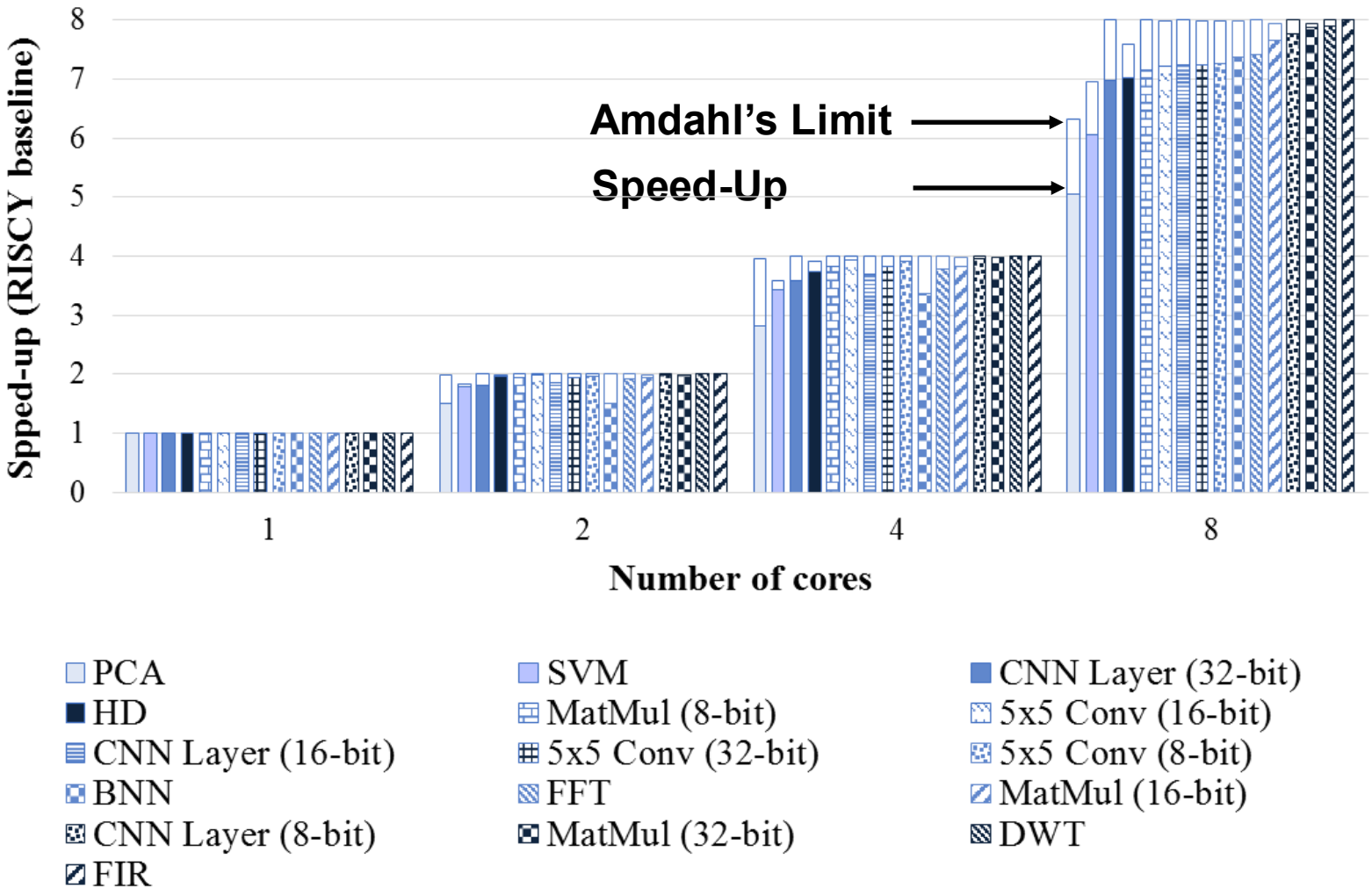
- HWCE (convolution)
- Neurostream (ML)
- HWCrypt (crypto)
- PULPO (1st order opt)

At the extreme edge: The Mr Wolf IoT Processor



PULP

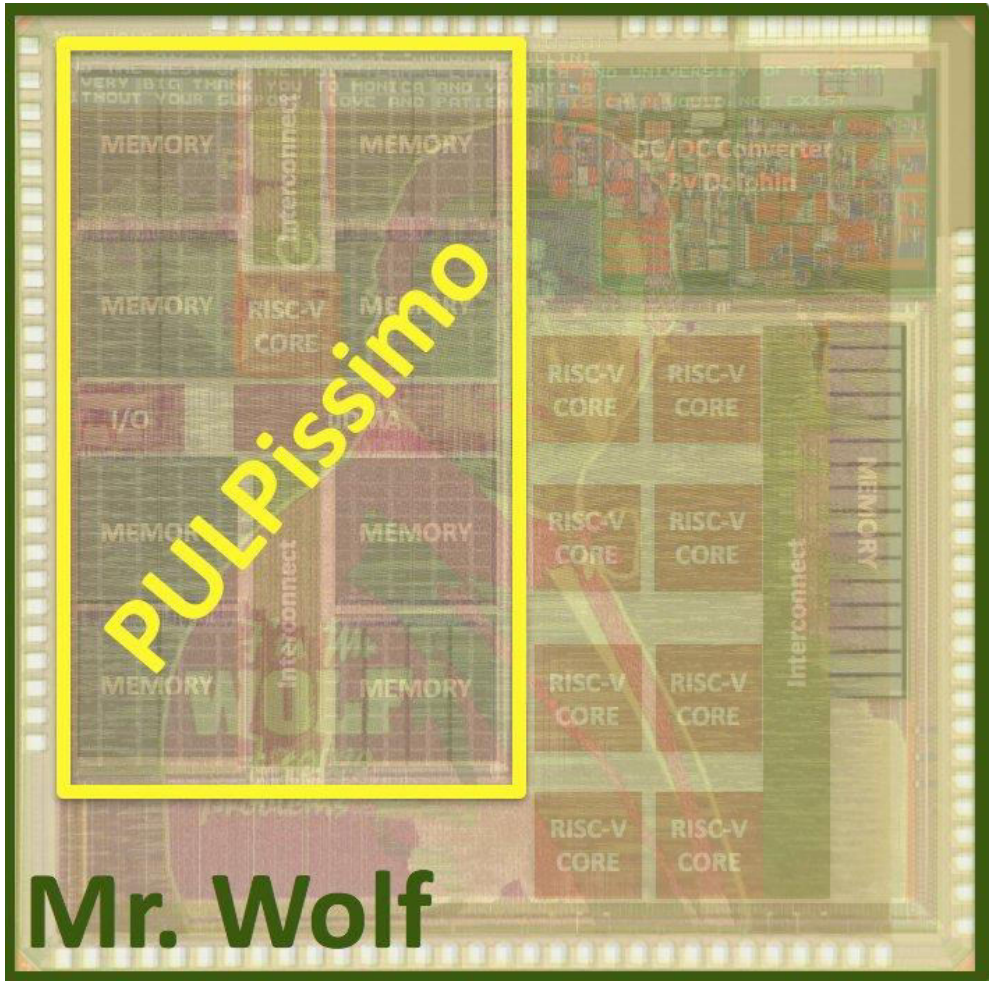
8-Processor PULP Cluster: Parallel Speed-up



Mr. Wolf Chip Results: Heterogeneous Computing Works

Technology	CMOS 40nm LP
Chip area	10 mm²
VDD range	0.8V - 1.1V
Memory Transistors	576 Kbytes
Logic Transistors	1.8 Mgates
Frequency Range	32 kHz – 450 MHz
Power Range	72 µW – 153 mW

Power Managent (DC/DC + LDO)	VDD [V]	Freq.	Power
Deep Sleep	*0.8*	*n.a.*	*72 µW*
Ret. Deep Sleep	*0.8*	*n.a*	*76.5 - 108 mW*
SoC Active	*0.8 - 1.1*	*32 kH 450 M*	*0.97 - 38 mW*
Cluster Active	*0.8 - 1.1*	*32 kH 350 M*	*1.6 - 153 mW*

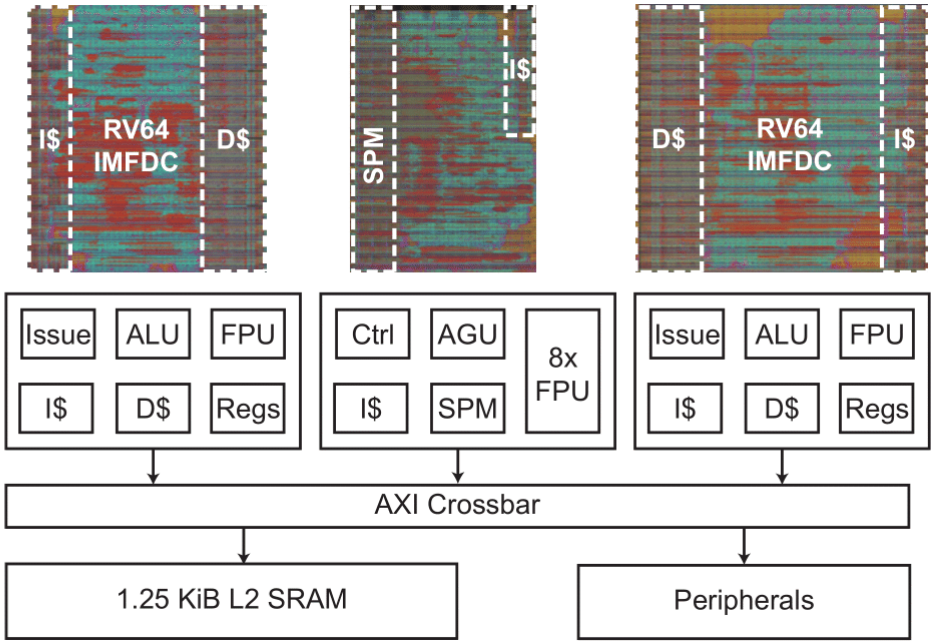


Max perf 16.4 GOp/s, Max En.Eff. 274 MOp/s/mW

Back to HPC: Kosmodrom

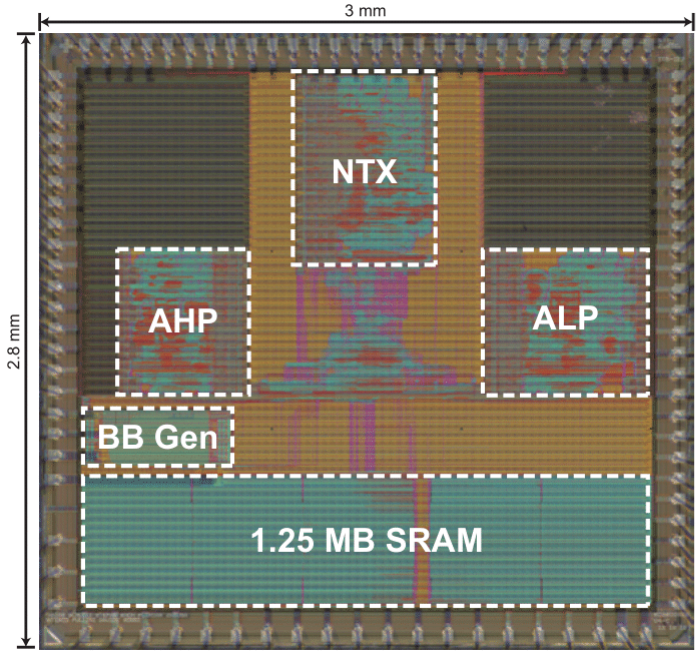
Need 64 bit address space, virtual memory, FP & DP, scaled technology

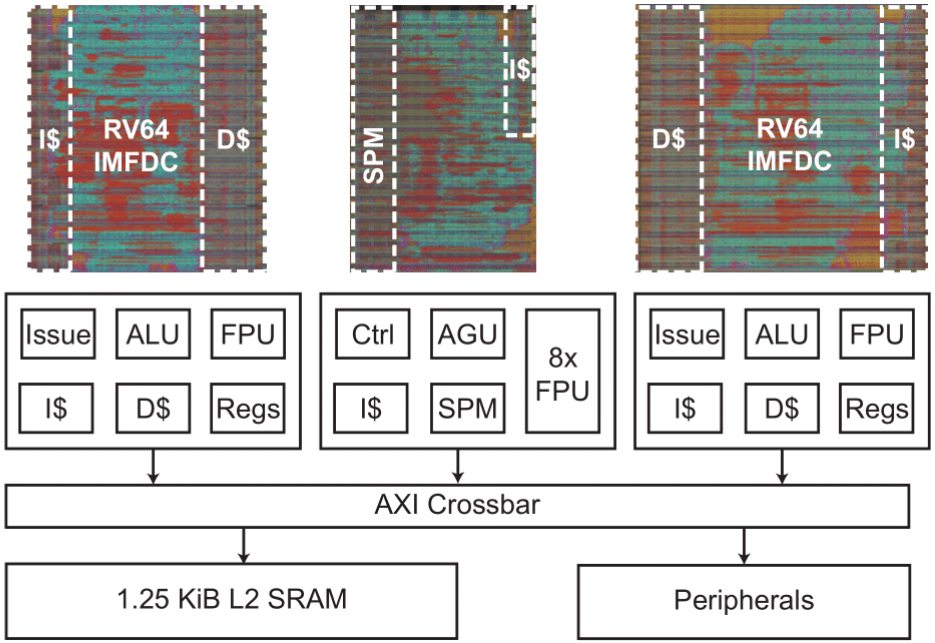
- 22nm FDX technology
- Two application-class RISC-V Ariane cores [1] - DP
  - RV64GCXsmallfloat
  - General purpose workloads
- Network Training Accelerator (NTX) [2] - FP
  - Accelerates oblivious kernels:
    - Deep neural network training
    - Stencils
    - General linear algebra workloads
- 1.25 MiB of shared L2 memory
- Peripherals



PULP

Architecture: Floorplan





Architecture: Ariane RISC-V Cores

- RV64GC, 6-stage, in-order, out-of-order execute
- 16 KiB instruction cache, 32 KiB data cache
- Transprecision floating-point unit (TP-FPU) [3]
  - double-, single- and half-precision FP formats
  - Two custom formats FP16alt and FP8
  - All standard RISC-V formats as well as SIMD
- Two different implementations:
  - Ariane High Performance (AHP): tuned for high-performance applications
  - Ariane Low Power (ALP): tuned for light, single-threaded applications

Architecture: Network Training Accelerator (NTX)

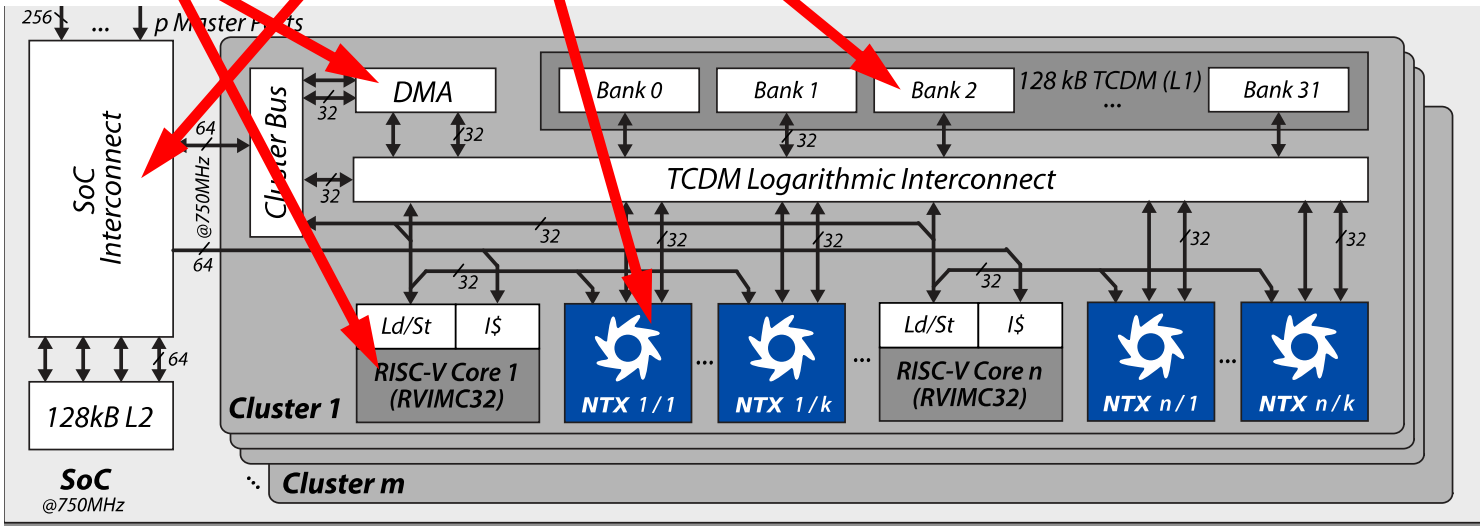
- "Network Training Accelerator"
  - 32 bit float streaming co-processor (IEEE 754 compatible)
  - Custom 300 bit "wide-inside" Fused Multiply-Accumulate
  - 1.7x lower RMSE than conventional FPU
  - 1 RISC-V core ("RI5CY") and DMA
  - 8 NTX co-processors
  - 64 kB L1 scratchpad memory (comparable to 48 kB in V100)

Key ideas to increase hardware efficiency:

- Reduction of von Neumann bottleneck (**load/store elision** through streaming)
- **Latency hiding through DMA-based double-buffering**

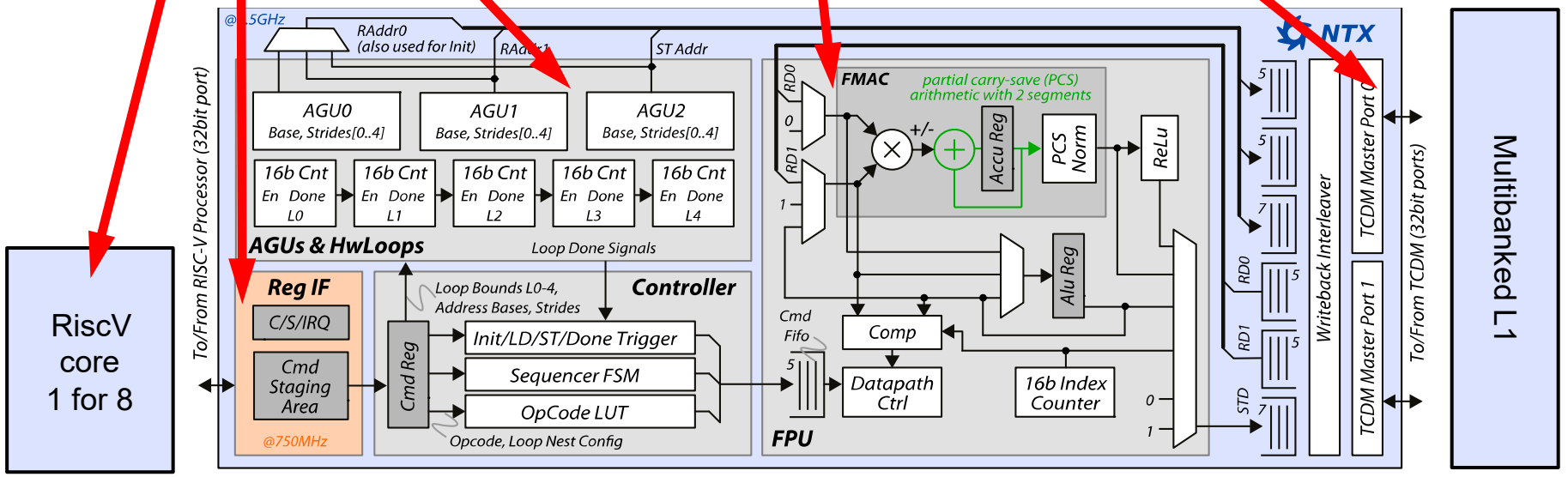
Flexible Architecture NTX accelerated cluster

- **1 processor** core controls **8 NTX** coprocessors
- Attached to 128 kB shared **TCDM** via a logarithmic interconnect
- **DMA** engine used to transfer data (double buffering)
- Multiple clusters connected via interconnect (crossbar/NoC)



Network Training Accelerator (NTX)

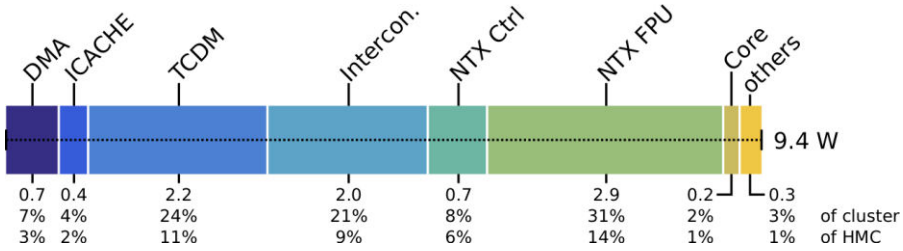
- Processor configures Reg IF and manages DMA double-buffering in L1 memory
- Controller issues AGU, HWL, and FPU micro-commands based on configuration
- AGUs generate address streams for data access
- FMAC with extended precision + ML functions
- Reads/writes data via 2 memory ports (2 operand and 1 writeback streams)



Again: specialized “deep interfaces” + Instruction extensions

NTX Power Breakdown & GPU SM Comparison

- NTX dissipates significant fraction of power in its FPU (more is better):
  - 31% of cluster
  - 14% of entire HMC
  - Recall: GPU is just around 5% [1]
- Compared to NVIDIA Volta GPU [2]:
  - Register file in GPU holds registers and thread-local data
  - Each register read/write is an SRAM access
  - Register and data accesses compete for SRAM



1 Volta SM	8 NTX cl.
64 FPUs	64 FPUs
256 kB RF 128 kB L0 Cache	512 kB TCDM
32-2048 threads	8 threads

Volta Assembly

```
LDS  R2, [R0]
LDS  R3, [R1]
FFMA R4, R2, R3, R2
```

2 mem. acc. ("[…]")
8 reg. acc.

= 10 SRAM hits total

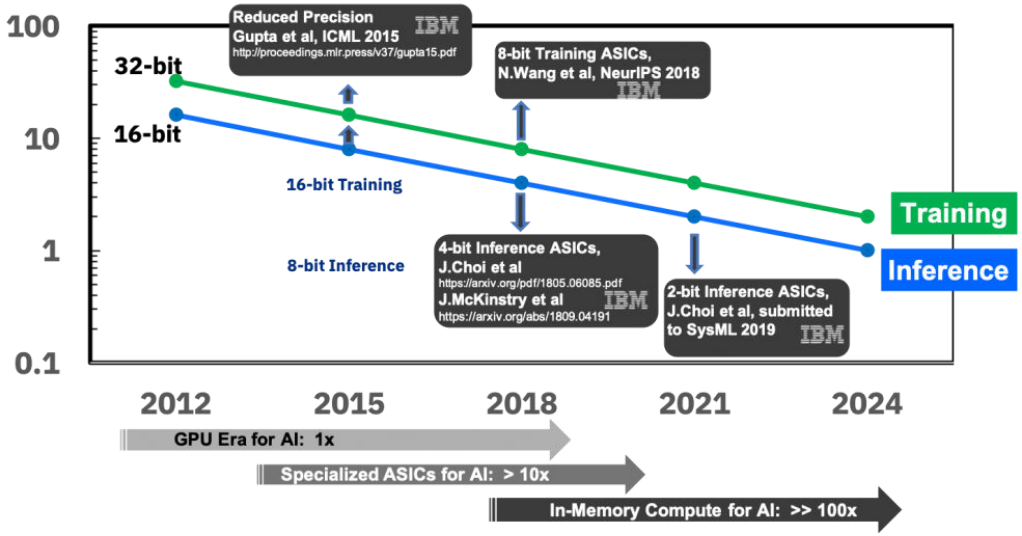
NTX Pseudocode

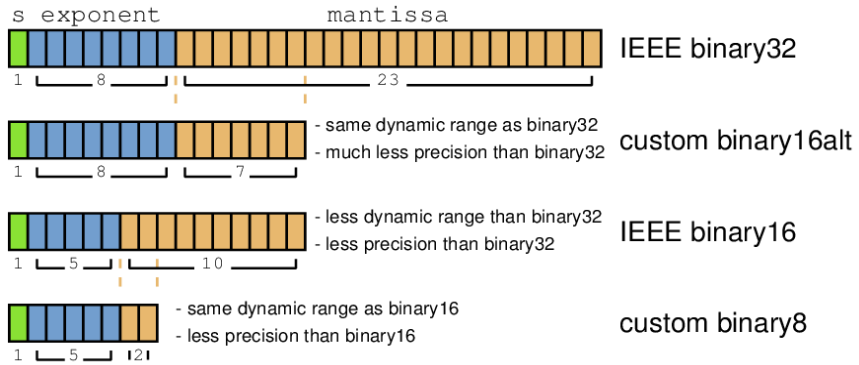
```
FMAC accu, [AGU0], [AGU1]
```

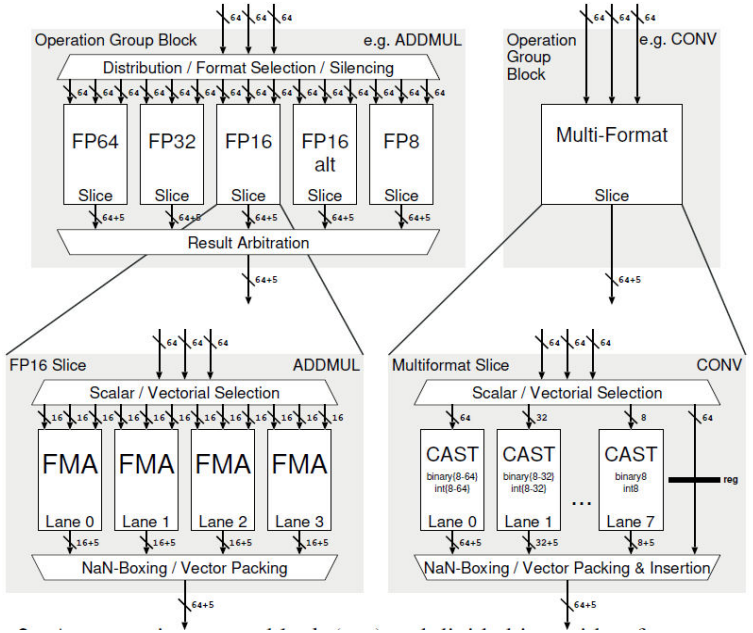
2 mem. acc. ("[…]")
0 reg. acc.
(+ addr. calc for free)

= 2 SRAM hits total

Low-Bitwidth Floating point Formats



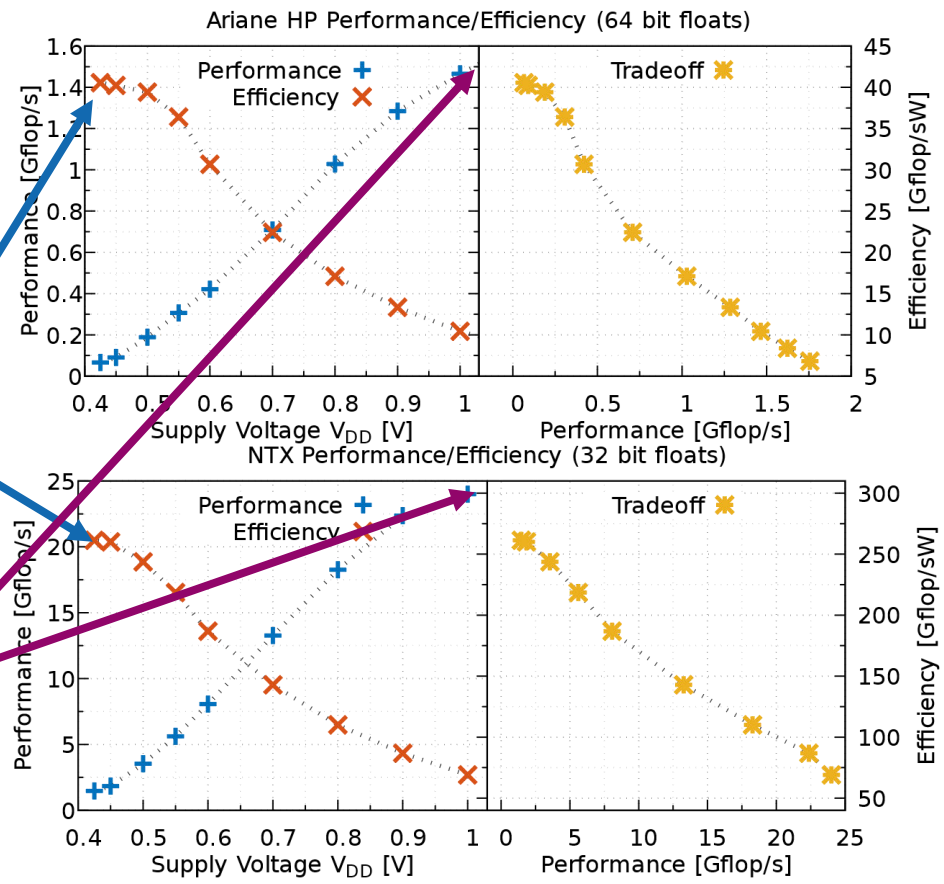




- Flexible (cycle by cycle) precision modulation (FP)
- Save precious DRAM bandwidth
  - Custom number formats
    - Use float8, float16, float16alt

Chip Measurements Results: Core vs. Accelerator

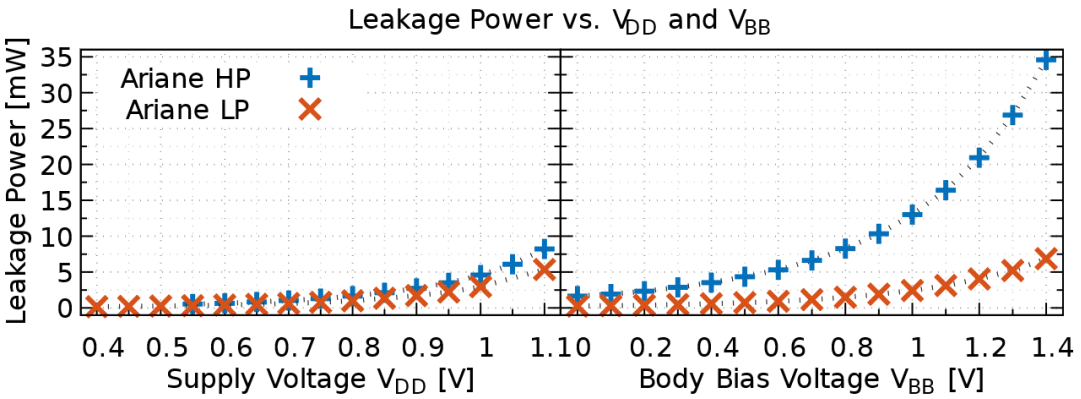
- General purpose cores to run operating system
- Specialized workloads are off-loaded to the NTX
- **NTX** is **6x** more energy efficient (41 vs 266 Gflop/sW) for oblivious kernels
  - Accelerator
  - FP precision
- **NTX** provides **18x** the performance (1.5 vs 24 Gflop/s)



Results: Cell Library (Technology)

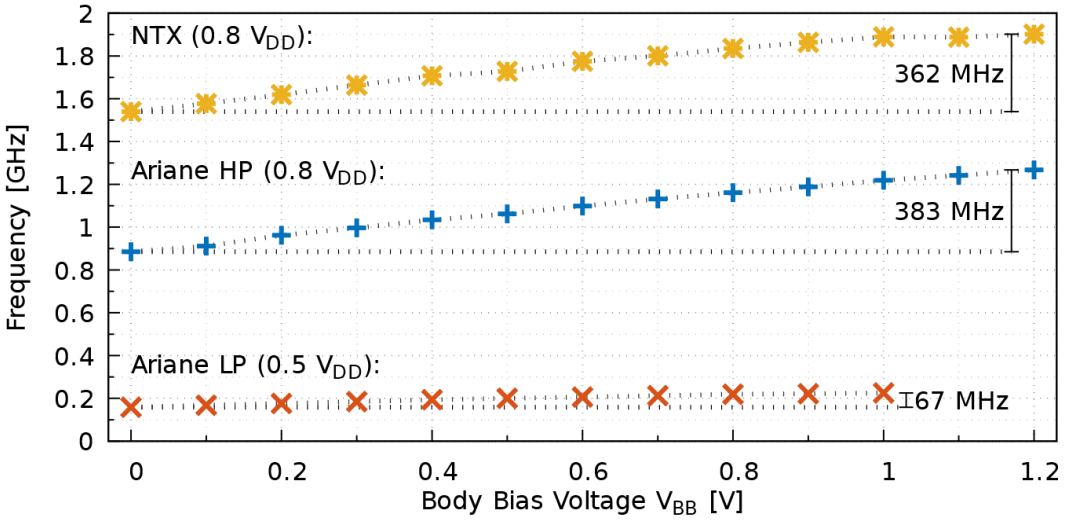
- AHP and ALP manufactured with two different cell libraries
- **AHP** tuned for **high-performance**, using fast, short-channel transistors
  - 20nm, 24nm, 28nm
  - 0.8V nominal voltage
  - Fast, single-ported SRAMs for caches
- **ALP** tuned for **minimizing power**, using slower, lower leakage cells
  - 28nm, 32nm, 36nm
  - 0.5V nominal voltage
  - Low power, single-ported, dual-supply SRAMs

- AHP: **0.85 GHz**, 50.2 mW total power, 4.6 % leakage power
- ALP: 0.175 GHz, **6.2 mW** total power, **2.8 % leakage power**



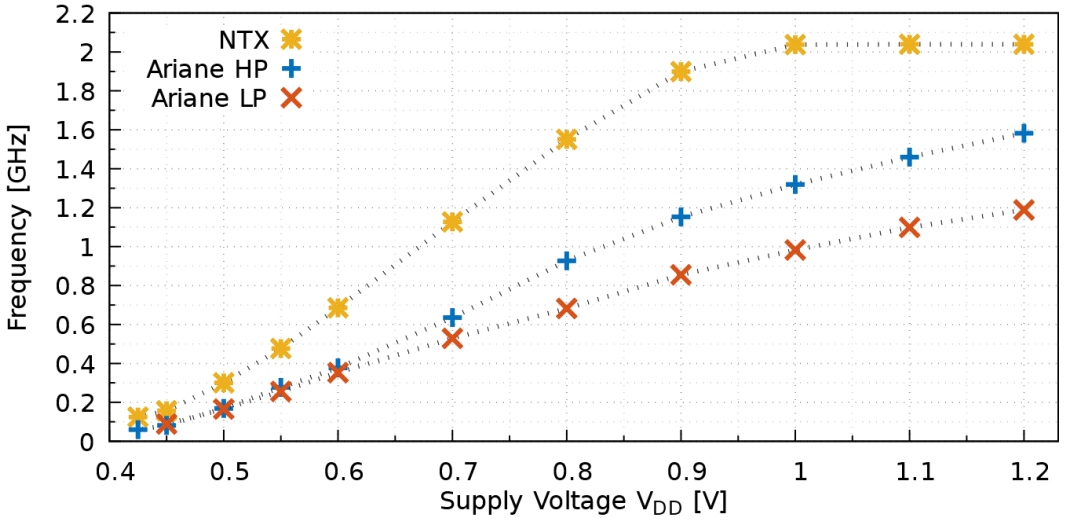
Results: Body Bias Voltage

- Flip-well transistors enable forward body bias (FBB)
- FBB lowers V_{TH}
- Up to 383 MHz speed-up
- FBB can be used for frequency centering at low voltage for ALP
- Extra frequency boost at high supply voltages for the AHP and NTX



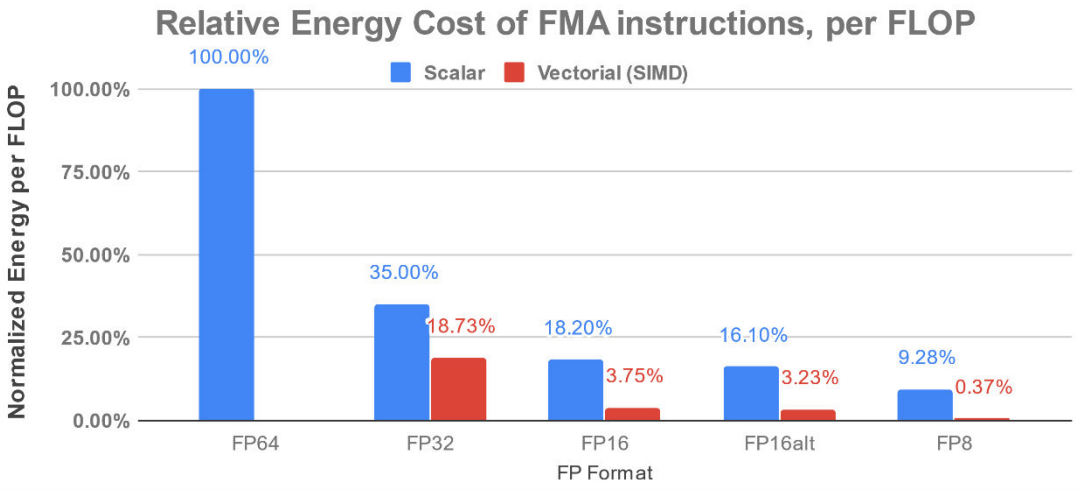
Results: Supply Voltage

- **AHP** runs up to 1.6 GHz
- **NTX** runs up to 2 GHz (limited by the on-chip clock generation capabilities)
- **ALP** tuned for background-tasks at lower speeds
  - 175 MHz at 0.5V (nominal voltage)
- At high frequencies AHP becomes more efficient
- At low speeds ALP is more efficient due to reduced leakage
- **AHP** and **ALP** cover complementary operating conditions



Results: FP Precision and Energy trade-off

- Trade-off floating point precision for instruction energy
- Energy cost of FP operations is super linearly proportional to data width
- Smaller FP formats take less latency to complete
- SIMD style vectors yield higher throughput
- Improve **energy to solution** and **time to solution** up to **7.95x** and **7.6x** for FP8 workloads



Summary on Kosmodrom: State of the Art

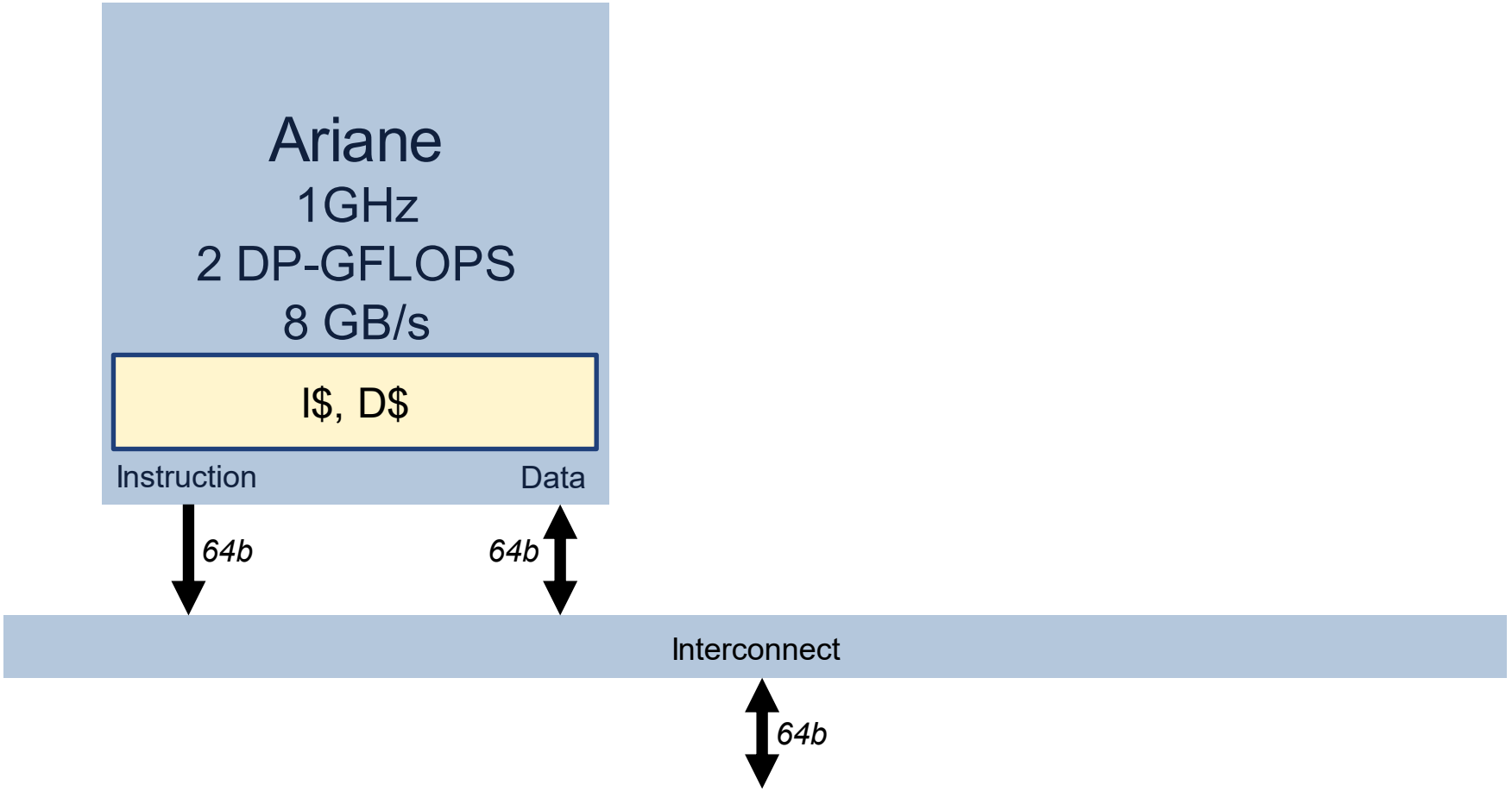
- We achieve **higher energy-efficiency** for AHP and ALP than competitive RISC-V processors (Rocket)
- Ariane contains slightly larger caches (32 KiB compared to 16 KiB)
- The ALP implementation is penalized because of less mature cell libraries available to us (7k cells vs 2k cells)
- NTX achieves a **2x** gain in energy-efficiency compared to Tesla V100

		AHP	ALP	NTX
Nominal VDD	[V]	0.8	0.5	0.8
Frequency	[GHz]	0.85	0.175	1.55
Area	[mm^2]	**0.4**	0.5	0.5
Area*	[MGE]	**1.95**	3.96†	2.8
Total Power	[mW]	50.2	**6.2**	160.0
Leakage Power	[%]	4.6	**2.8**	5.5
Lkg. Power / Area (0.5 V	[mW/mm^2]	1.25	**0.36**	3
Energy/Instr.	[pJ]	24.4	22.7	**3.8**
Max. Eff.	[Gflop/s W]	41	44	**266**
Max. Perf.	[Gflop/s]	1.5	1.3	**24**
Area Eff.	[Gflop/s mm^2]	3.75	2.6	**48**
SLVT/LVT	[%]	**26/74**	83/17	60/40
FP Formats		8/16/ 16alt/ 32/64	8/16/ 16alt/ 32/64	32

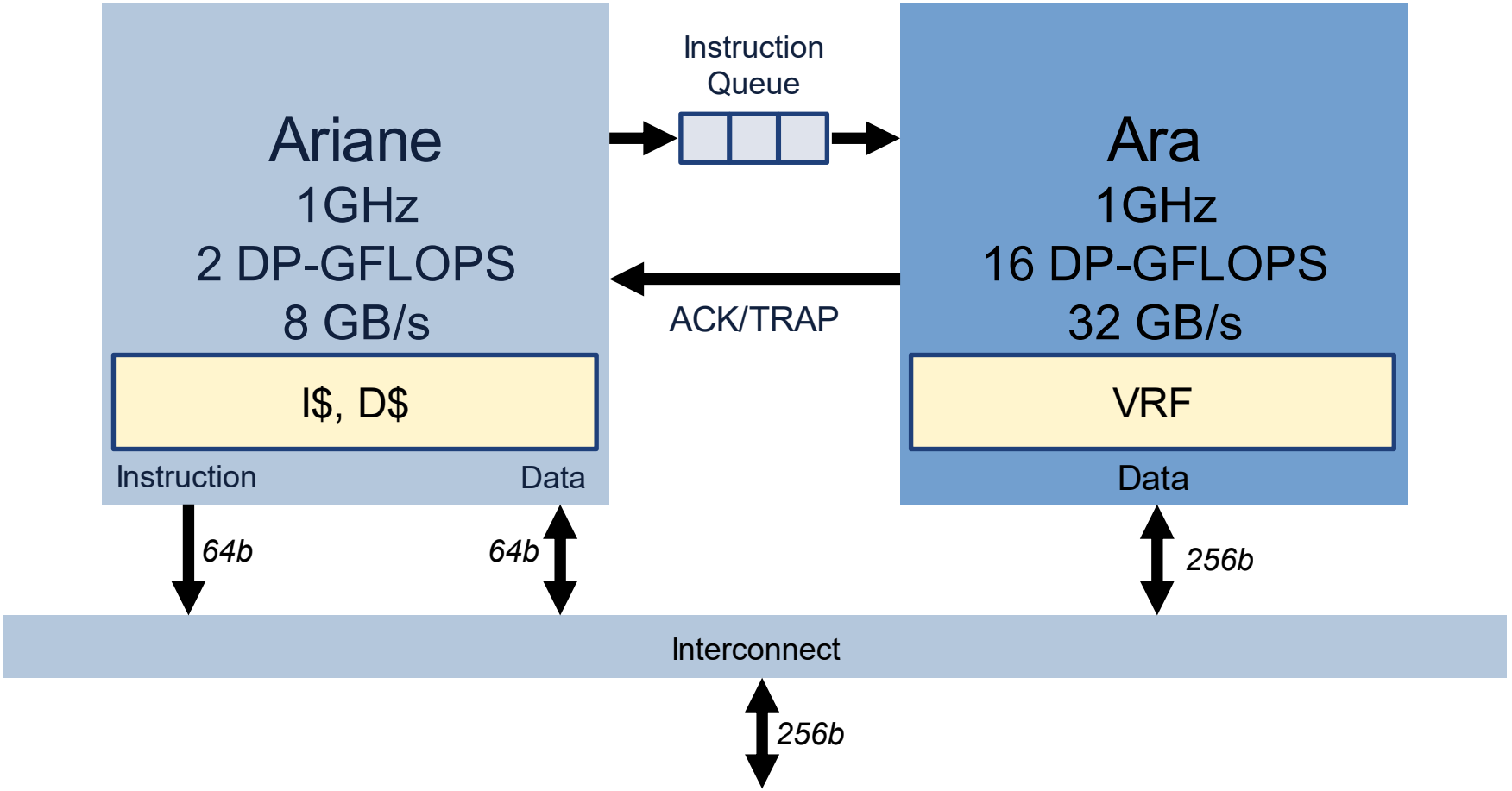
	AHP [us]	ALP [us]	NTX [us]	Cortex A53[14]	Rocket 64b[15]	Tesla V100§	Xeon 8180§
Node/V_{DD}	22/0.45	22/0.45	22/0.45	16/0.8	40/0.65	12/1.0	14/0.9
32 bit floats							
Energy Eff.†	93 ‖	98 ‖	266	38.7 *	16.7 ¶	122	21.9
Area Eff.‡	7.5	5.2	47.1	8.7 *	7.3 ¶	20.5	3.57
64 bit floats							
Energy Eff.†	41	44	—	19.4 *	16.7 ¶	61	11.0
Area Eff.‡	3.75	2.6	—	4.4 *	7.3 ¶	10.3	1.79

† Gflop/s W; ‡ Gflop/s mm^2 (node-scaled); § our estimates;
* assuming NEON; ¶ no SIMD ‖ extrapolated from 64 bit

Efficient & High performance DP: Ariane is not Enough!

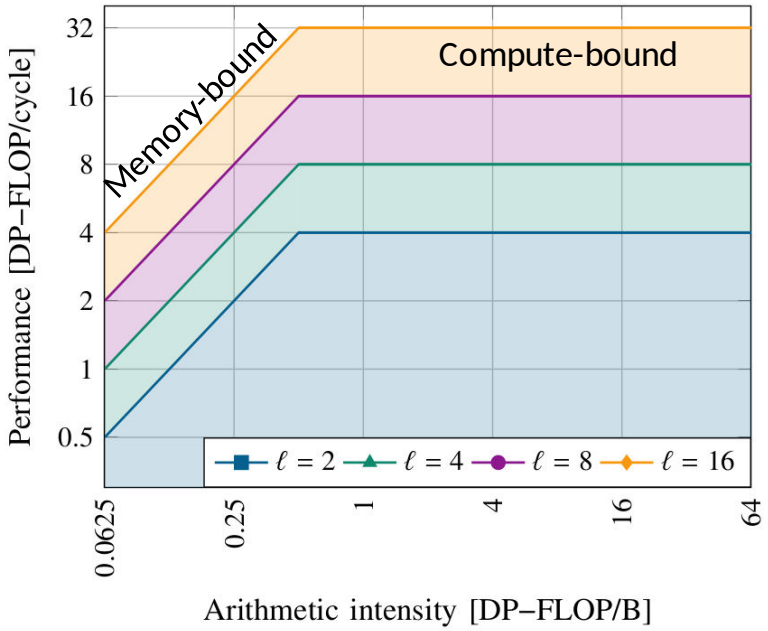


Enter ARA: Open-Source RISCV Vector Engine



Memory Bandwidth

- Arithmetic intensity
  - Operations per byte: data reuse of an algorithm
  - One FMA ⟶ two operations
- Memory-boundness and compute-boundness
- Ara targets 0.5 DP-FLOP/B
  - Memory bandwidth scales with the number of physical lanes



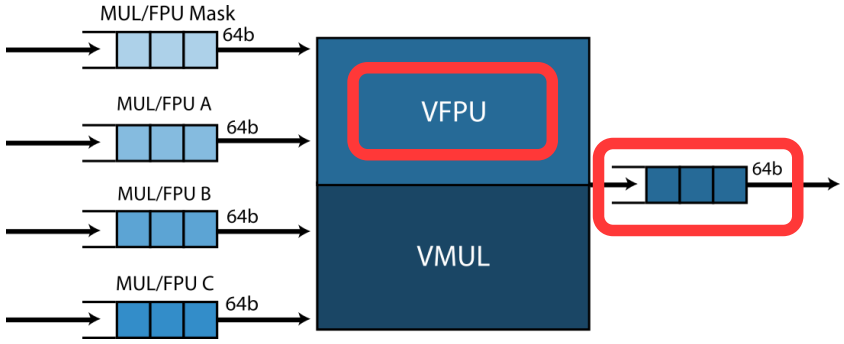
RISC-V Vector Extension

- RISC-V “V” Extension
  - Cray-like vector processing, opposed to packed-SIMD

- Ara is based on the version 0.5
  - Work is being done to update it to the latest version
  - Open-source in 2020 (Q1)

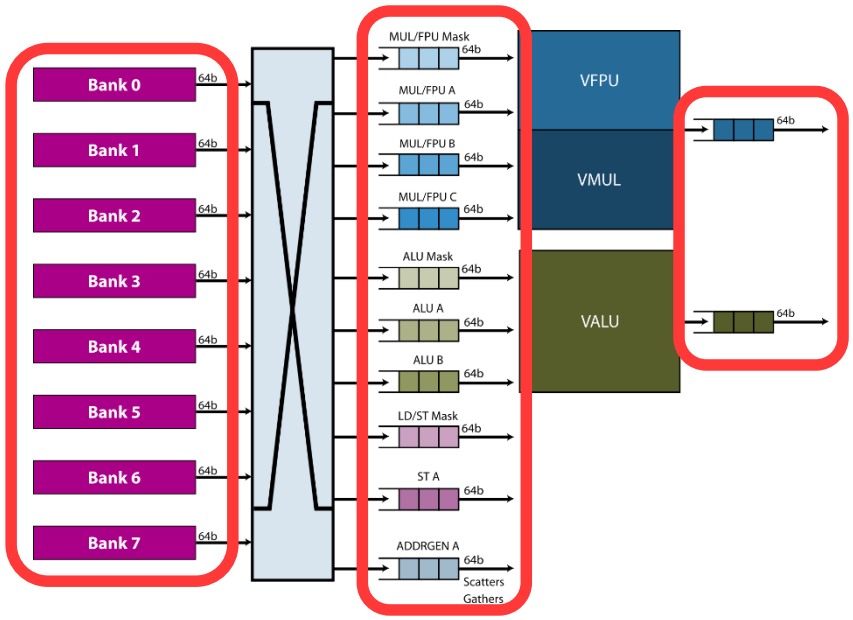
Ara main datapath elements

- ALU, MUL and FPU
- Transprecision functional units
  - Throughput of 64 bit per cycle
  - Packed-SIMD approach
- FPU
  - FP64, FP32, FP16, bfloat16
  - Independent pipelines for each data type
    - Each with a different latency



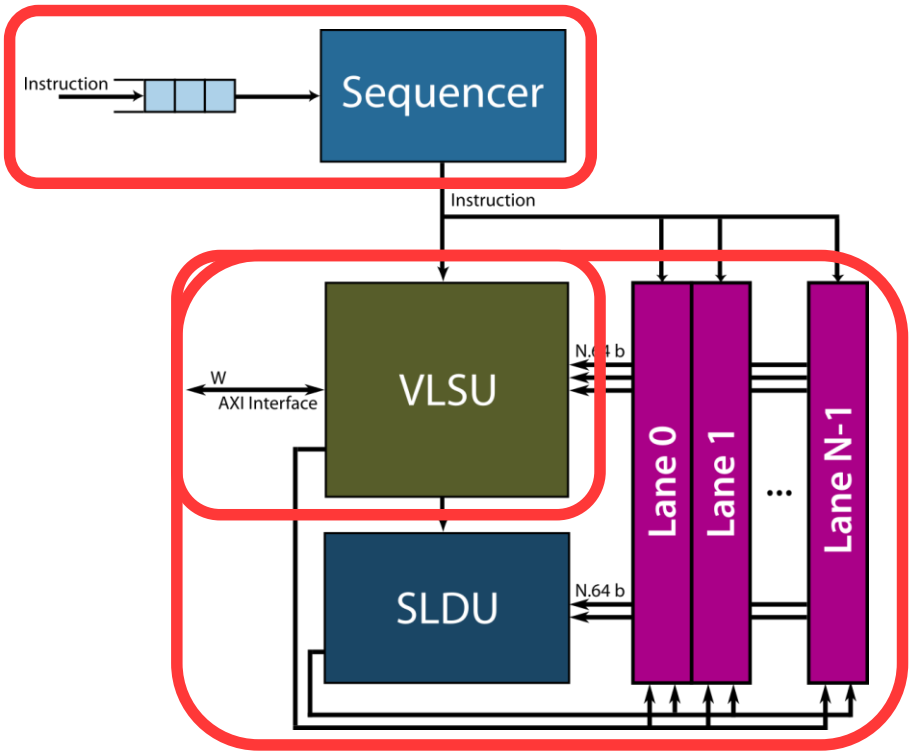
Vector Lane: base computational unit

- Per-lane Vector Register File
  - 8 x 1RW SRAM banks
  - Functional units only access their own section of the VRF
  - Requires an arbiter (banking conflicts)
- Operand queues
  - Hide latency due to banking conflicts on the VRF
  - One FIFO per operand per datapath unit: 10 x 64b queues
  - Similar queues for output operands



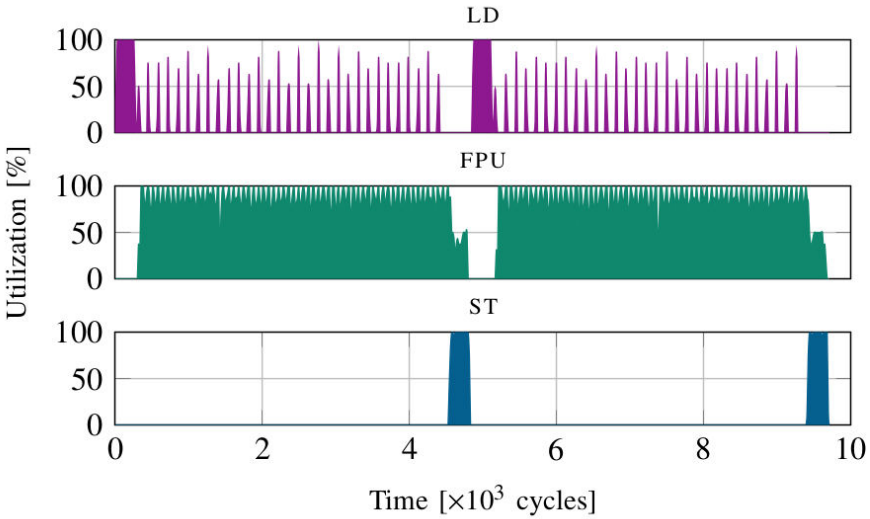
Ara with *N* identical vector lanes

- Instruction forked from Ariane's issue stage
  - Instructions are issued *non-speculatively*
  - Bookkeeping by the sequencer
- Load/Store and Slide Units access all the VRF
  - Connected to each lane
  - Scalability issue
- W = 32.N bits wide memory interface
  - Keep Ara performance per bandwidth ratio at 0.5 DP-FLOP/B



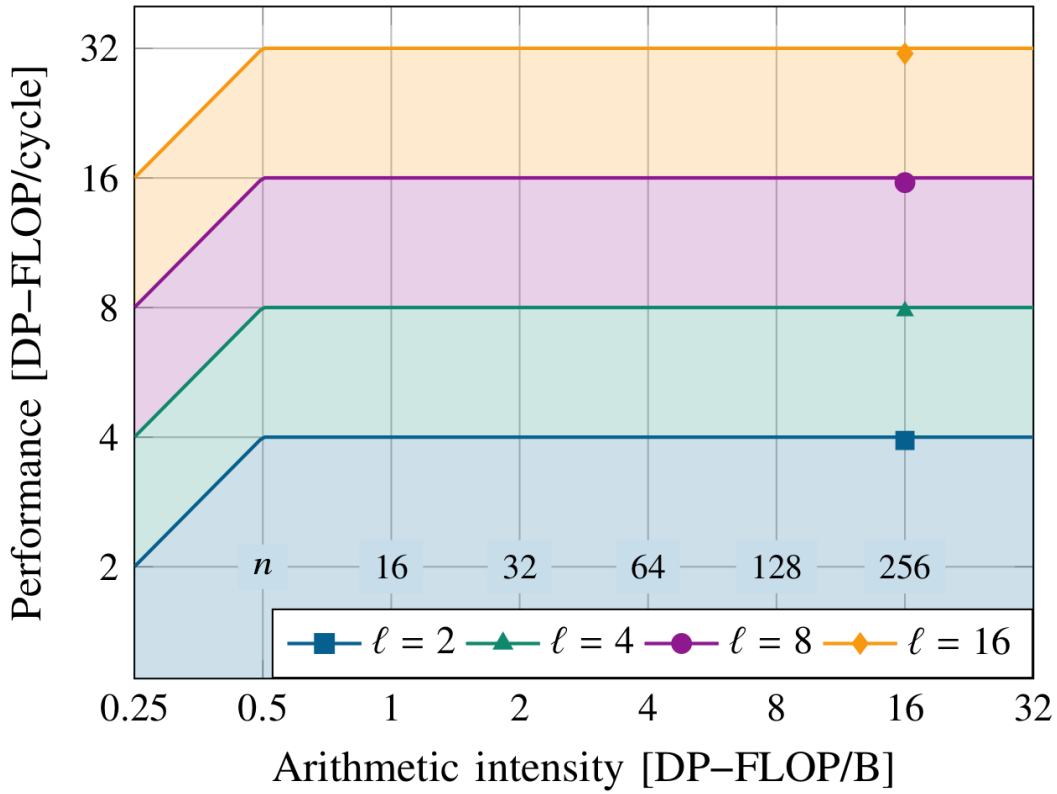
Matrix Multiplication on ARA

- DP-MATMUL
  - *n* x *n* double-precision matrix multiplication
  - C ← A · B + C
- 32n² bytes of memory transfers and 2n³ operations
  - *n/16* DP-FLOP/B
  - Compute-bound in Ara for *n* > *8*

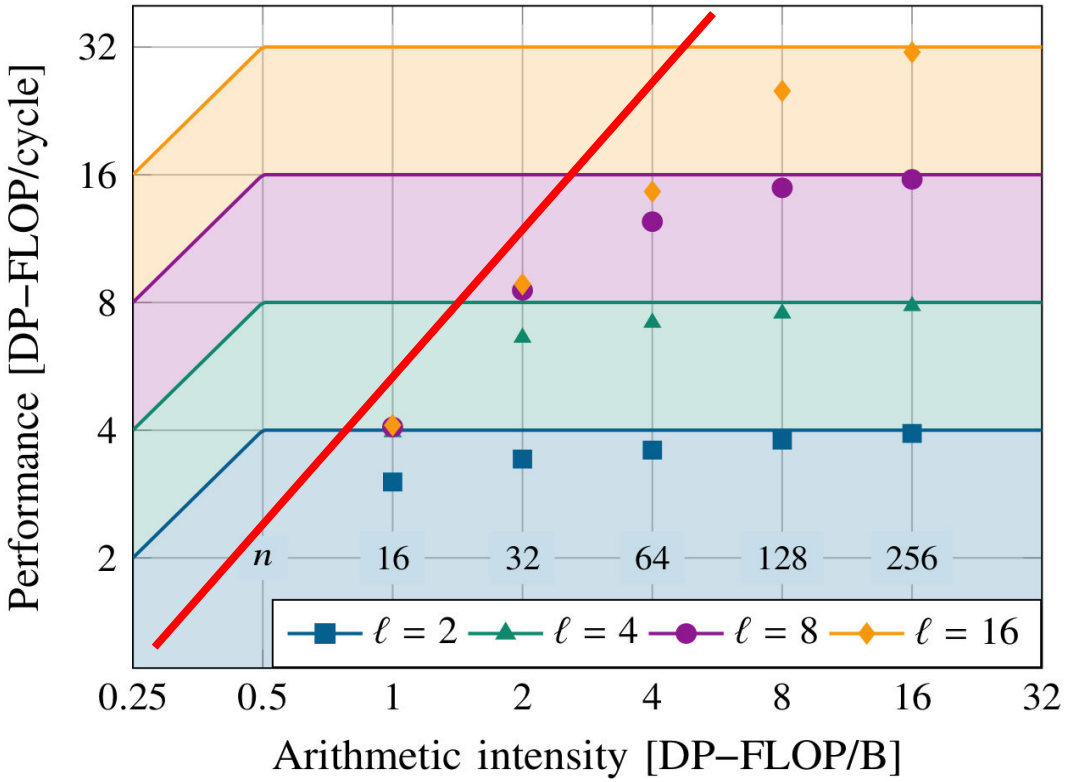


Functional unit's utilization for a 16x16 DP-MATMUL

Up to 98% Efficiency on *nxn* DP-MATMUL

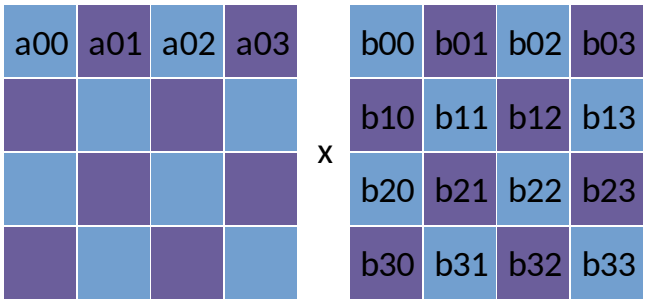


Up to 98% Efficiency on *nxn* DP-MATMUL (always?)



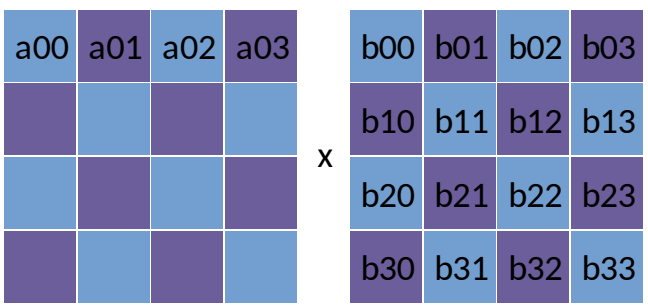
Matrix multiplication on Ara

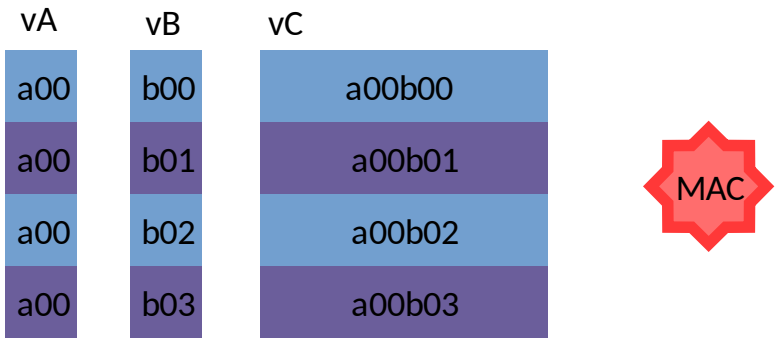
- Standard algorithm (row times column + reduction) is slow
  - Highly sequential
- Use a vector of reductions instead



Matrix multiplication on Ara

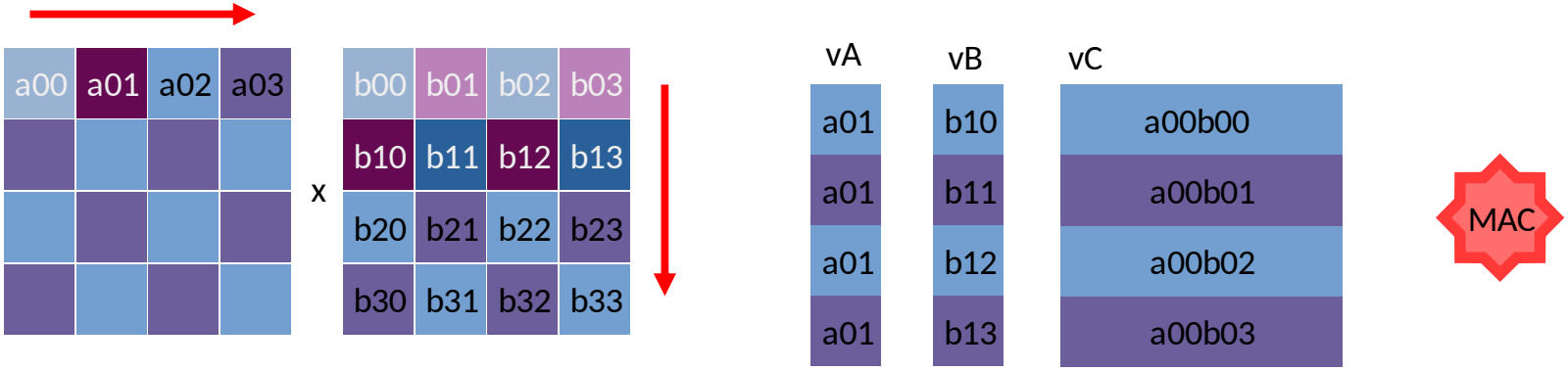
- Standard algorithm (row times column + reduction) is slow
  - Highly sequential
- Use a vector of reductions instead





Matrix multiplication on Ara

- Standard algorithm (row times column + reduction) is slow
  - Highly sequential
- Use a vector of reductions instead



Matrix multiplication on Ara

- Load row *i* of matrix B into vB
- for (int *j* = 0; *j* < *n*; *j*++)
  - Load element A*[j, i]*
  - Broadcast it into vA
  - vC ← vA . vB + vC

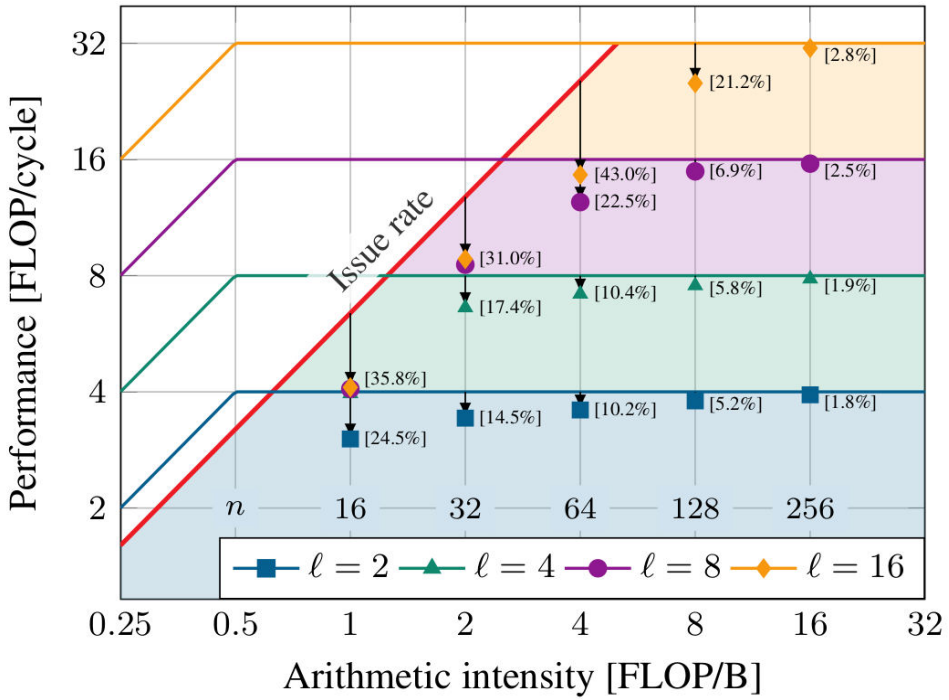
vld vB, 0(addrB)

(Unrolled loop)

- ld t0, 0(addrA)
- addi addrA, addrA, 8
- vins vA, t0, zero
- ***vmadd vC, vA, vB, vC***
- ld t0, 0(addrA)
- addi addrA, addrA, 8
- vins vA, t0, zero
- ***vmadd vC, vA, vB, vC***

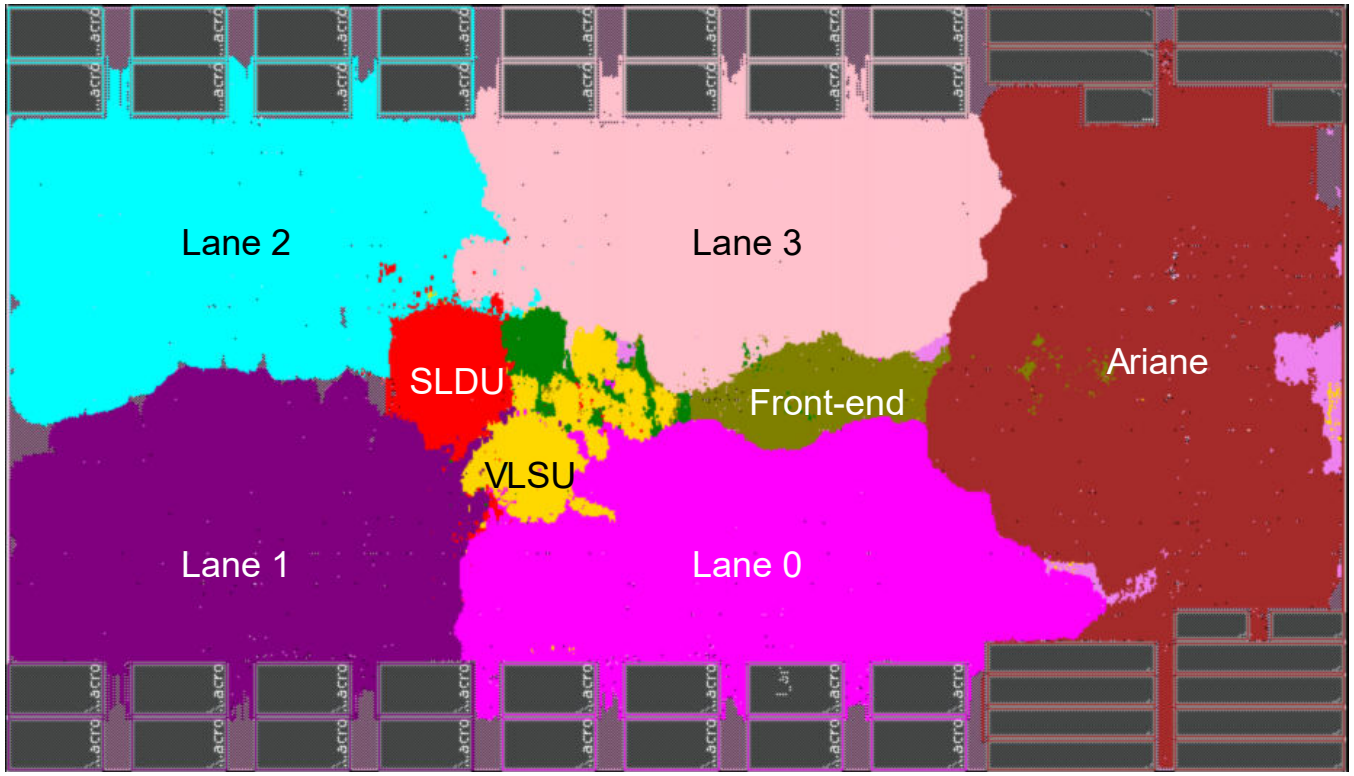
Issue rate performance limitation

- vmadds are issued at best every four cycles
  - Since Ariane is single-issue
- If the vector MACs take less than four cycles to execute, the FPUs starve waiting for instructions
  - Von Neumann Bottleneck
- This translates to a boundary in the roofline plot



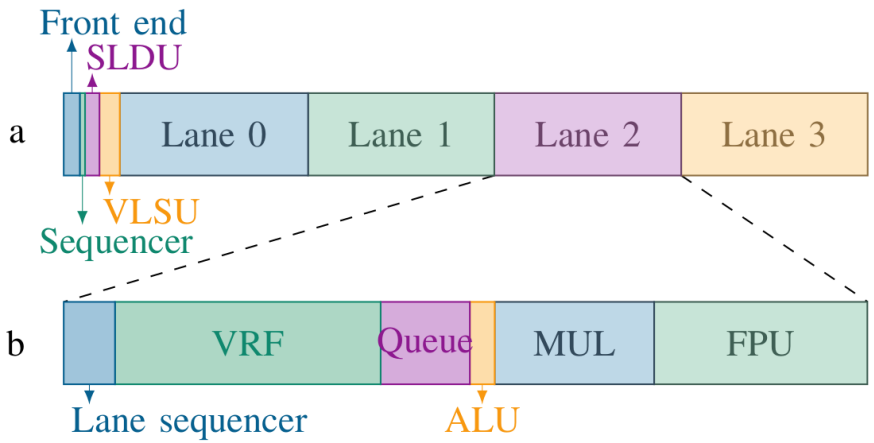
Ara: 4 lanes GF 22FDX 1.25 GHz implementation

(TT, 0.80V, 25˚C)



Ara: Figures of Merit

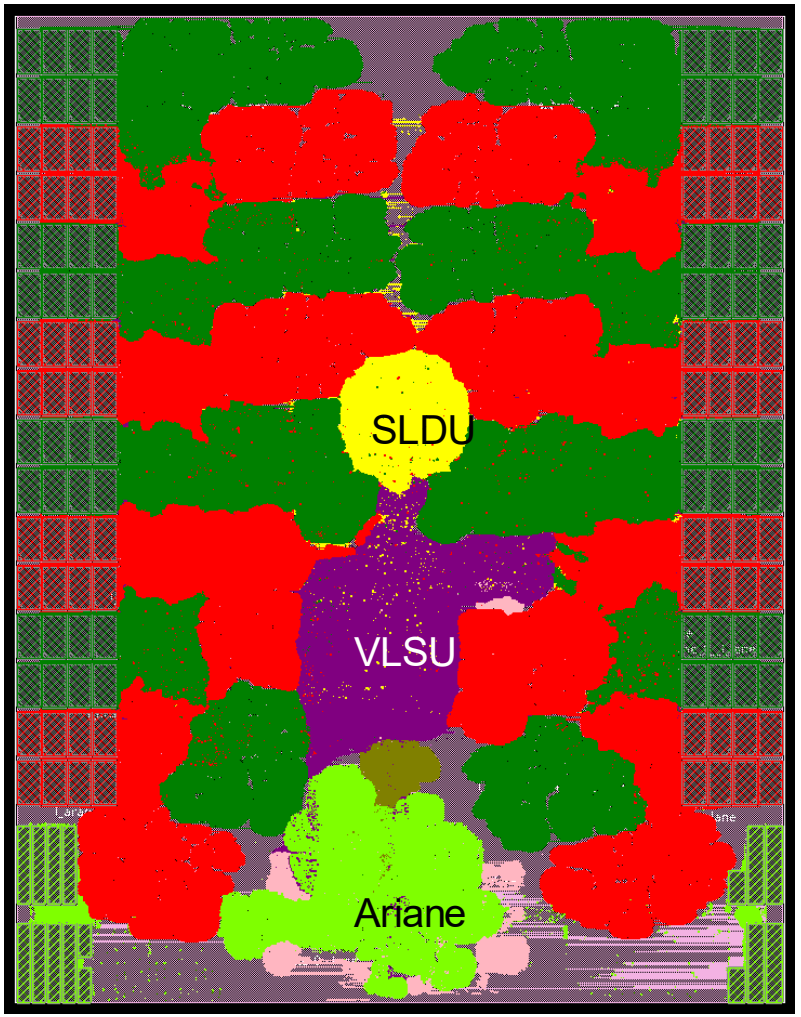
- Area breakdown



- Clock frequency
  - 1.25 GHz (nominal), 0.92 GHz (worst condition)
  - 40 gate delays
- Area: 3400 kGE
  - 0.68 mm^2
- 256 x 256 MATMUL
  - Performance: 9.8 DP-GFLOPS
  - Power: 259 mW
  - Efficiency: **38 DP-GFLOPS/W**
  - **~2.5X better than Ariane on same benchmark**

Ara: Scalability

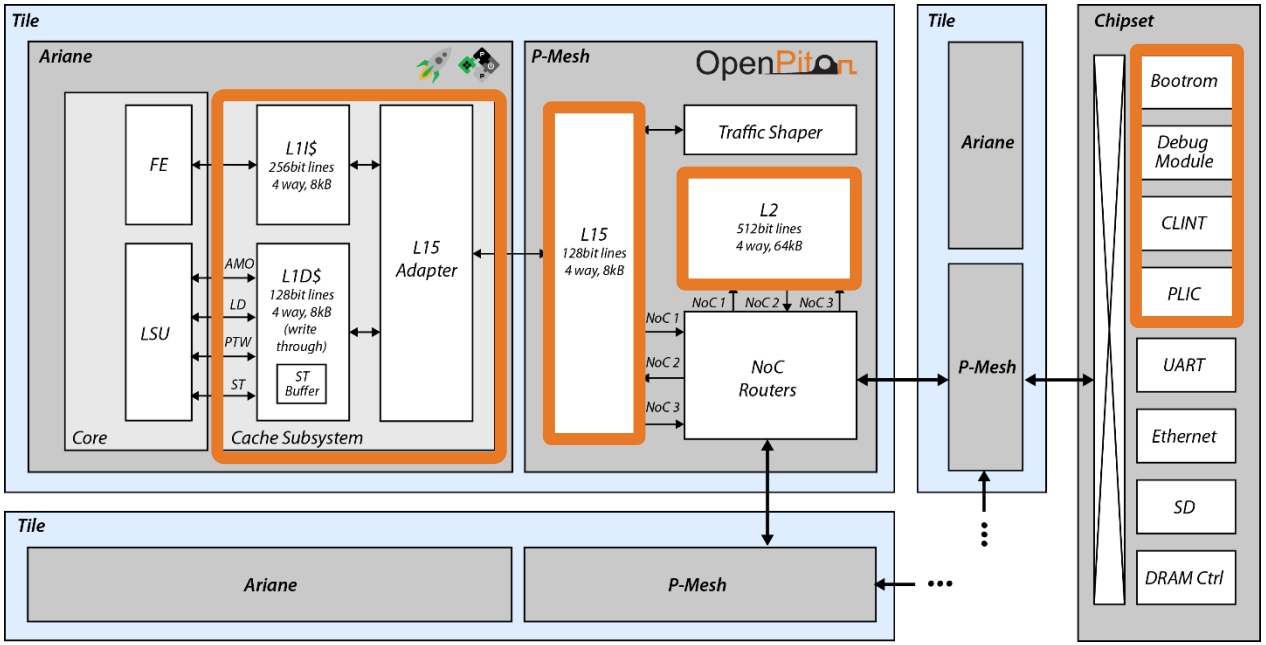
- Each lane is *almost* independent
  - Contains part of the VRF and its functional units
- Scalability limitations
  - VLSU and SLDU: need to communicate to all banks
- Instance with 16 lanes:
  - 1.04 GHz (nom.), 0.78 GHz (w)
  - 10.7 MGE (2.13mm²)
  - 32.4 DP-GFLOPS
  - **40.8 DP-GFLOPS/W**



16 ARAs give you 1TFLOP at 12W - NOT BAD!

OpenPiton+Ariane

If you are really passionate about cache coherent "scalable" machines…



- New write-through cache subsystem with invalidations and the TRI interface
- LR/SC in L1.5 cache
- Fetch-and-op in L2 cache
- RISC-V Debug
- RISC-V Peripherals

Note: ARA plugs in nicely at the L1 interface!

Configurability Options

Component	Configurability Options	
Cores (per chip)	Up to 65,536	
Cores (per system)	Up to 500 million	
Core Type	OpenSPARC T1	**Ariane 64 bit RISC-V**
Threads per Core	1/2/4	**1**
Floating-Point Unit	FP64, FP32	**FP64, FP32**, FP16, FP8, BFLOAT16
TLBs	8/16/32/64 entries	Number of entries **(16 entries)**
L1 I-Cache	Number of Sets, Ways **(16kB, 4-way)**	
L1 D-Cache	Number of Sets, Ways **(8kB, 4-way)**	
L1.5 Cache	Number of Sets, Ways **(8kB, 4-way)**	
L2 Cache	Number of Sets, Ways **(64kB, 4-way)**	
Intra-chip Topologies	**2D Mesh**, Crossbar	
Inter-chip Topologies	2D Mesh, 3D Mesh, Crossbar, Butterfly Network	
Bootloading	SD/SDHC Card, UART, RISC-V JTAG Debug	

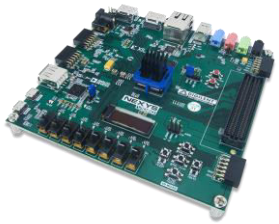
FPGA Prototyping Platforms

Available:

- Digilent Genesys2
  - $999 ($600 academic)
  - 1-2 cores at 66MHz
- Xilinx VC707
  - $3500
  - 1-4 cores at 60MHz
- Digilent Nexys Video
  - $500 ($250 academic)
  - 1 core at 30MHz

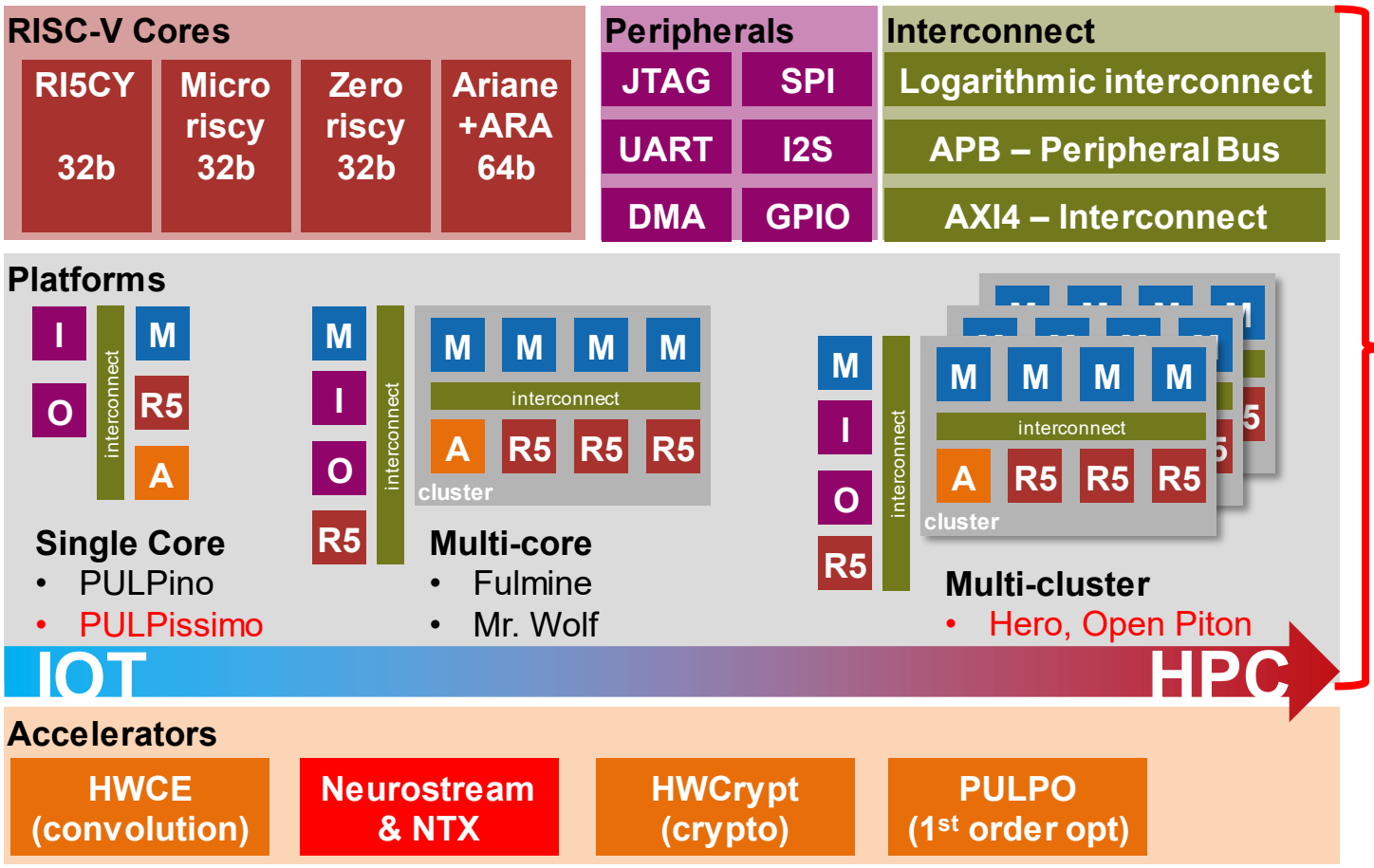
In progress:

- Xilinx VCU118, BittWare XUPP3R
  - $7000-8000
  - >100MHz
- Amazon AWS F1
  - Rent by the hour



A Computing Ecosystem Perspective

https://www.pulp-platform.org/ https://github.com/pulp-platform



OS HW Solderpad0.5

100s of users (academia + industry)

This is way too much for a university (or two)!

Academic Open-Source → Industrial Open source

OPENHW GROUP — PROVEN PROCESSOR IP

Rick O'Connor (OpenHW CEO, former RISC-V foundation director)

- OpenHW Group is a not-for-profit, global organization (EU,NA,Asia) driven by its members and individual contributors where HW and SW designers collaborate in the development of open-source cores, related IP, tools and SW such as the CORE-V Family of cores.
- OpenHW Group provides an infrastructure for hosting high quality open-source HW developments in line with industry best practices.

PULP — Parallel Ultra Low Power

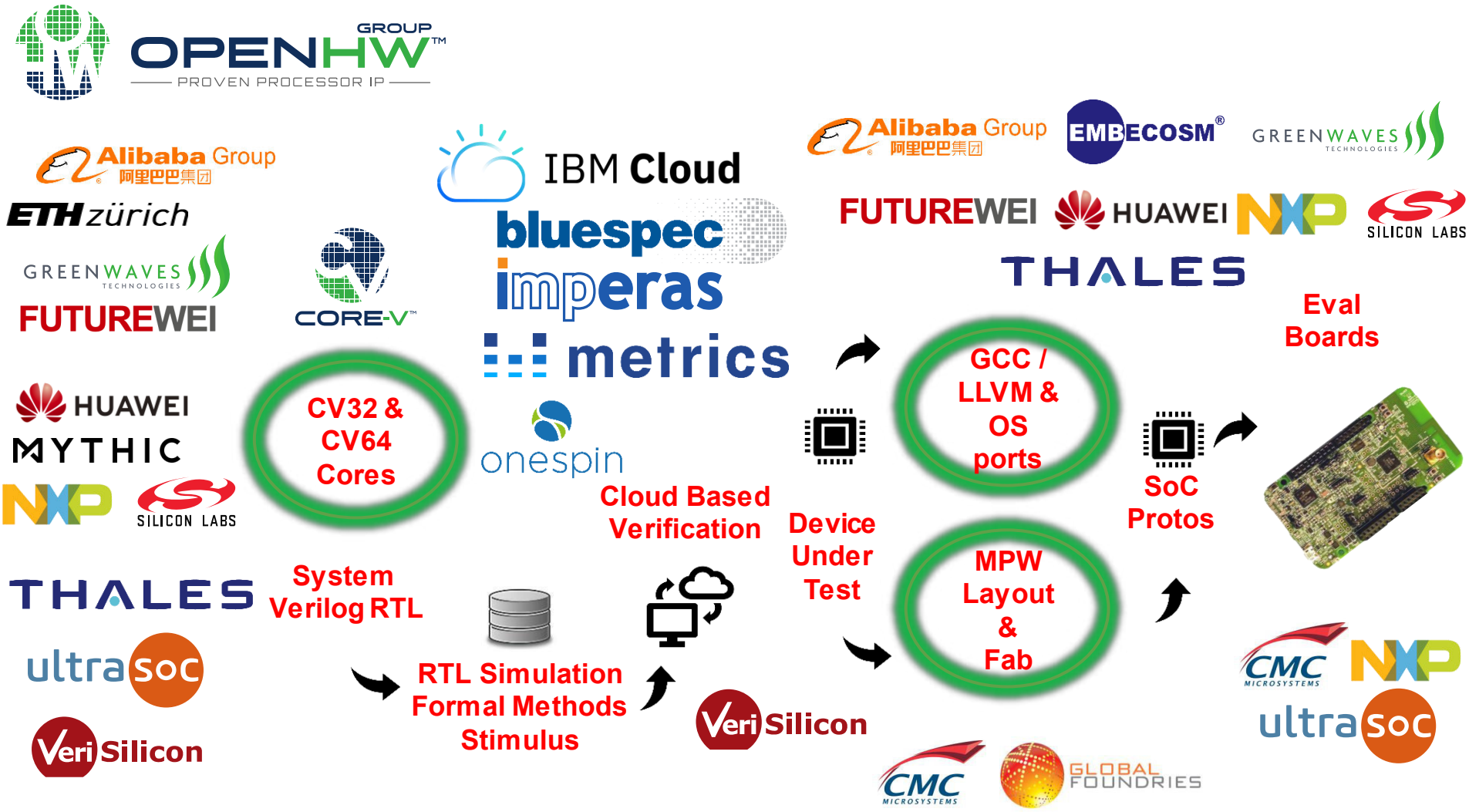
RI5CY, ARIANE

CORE-V™

Alibaba Group 阿里巴巴集团 · bluespec · BTA DESIGN SERVICES your project, our team · CMC · ETH zürich Integrated Systems Laboratory · EMBECOSM® · FUTUREWEI · GLOBAL FOUNDRIES · GREENWAVES TECHNOLOGIES · HUAWEI · imperas · metrics · MYTHIC · NXP · onespin · SILICON LABS · THALES · ultraSoC · VeriSilicon

PULP

OpenHW Group Ecosystem



A Vertical, Application-focused Open-Platform Approach

OpenTitan

More transparent, trustworthy, and secure RoT chip design

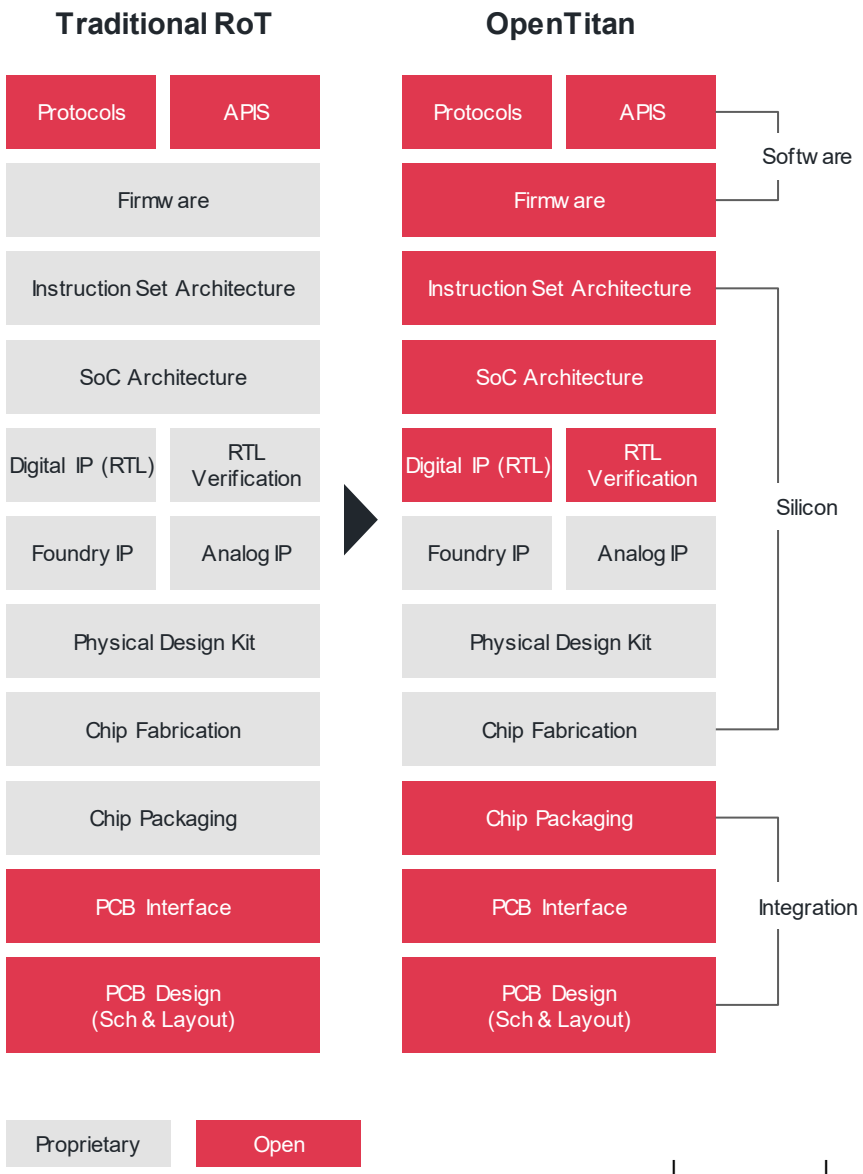
OpenTitan is the first open source silicon project building a transparent, high-quality reference design for silicon root of trust (RoT) chips.

Founding partners

Open HW enables a New Level of Openness in Security

Transparent:
Open implementation

- Transparency at the bottom; lower than any existing RoT solutions
- Transparency enables the community to proactively audit, evaluate, & improve the design
- Engineering: reference firmware, register-transfer level (RTL), design verification (DV), and integration guidelines



PULP

Feel the momentum!

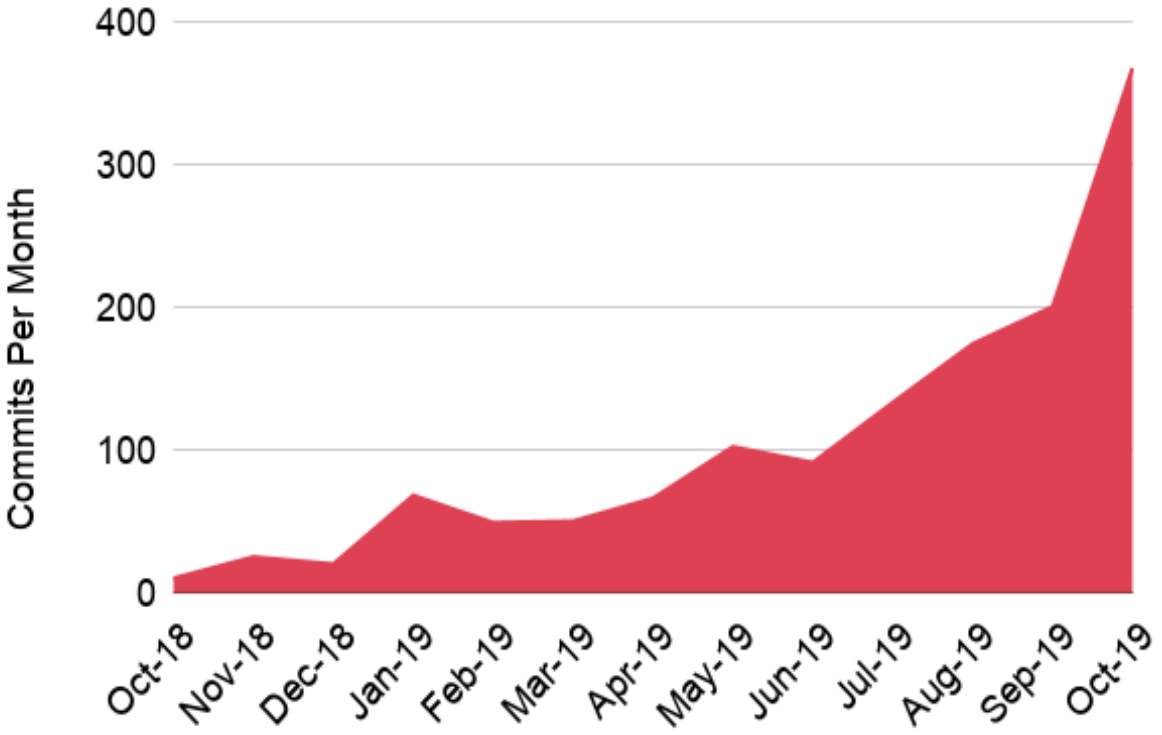
Ibex RISC-V core, flash interface, communications ports, cryptography accelerators, and more.

PULP Parallel Ultra Low Power → lowRISC

Zero-Riscy → Ibex

Vibrant repository

35+	Contributors
1300+	Contributions
470	GitHub Issues



HPC Vertical: The European Processor Initiative

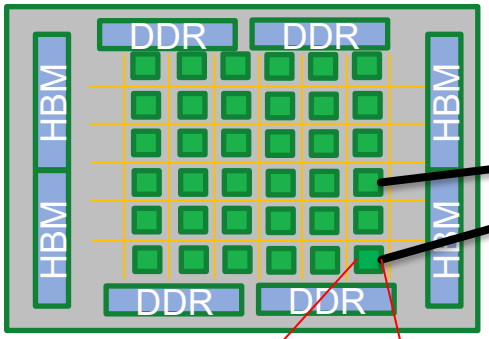
Europe Needs its own Processors

- Processors now control almost every aspect of our lives
- **Security** (back doors etc.)
- Possible **future restrictions on exports to EU** due to increasing protectionism
- **A competitive EU supply chain** for HPC technologies will create jobs and growth in Europe
- Sovereignty (data, economical, embargo)

- High Performance General Purpose Processor for HPC
- **High-performance RISC-V based accelerator**
- Computing platform for autonomous cars
- Will also target the AI, Big Data and other markets in order to be economically sustainable

PULP

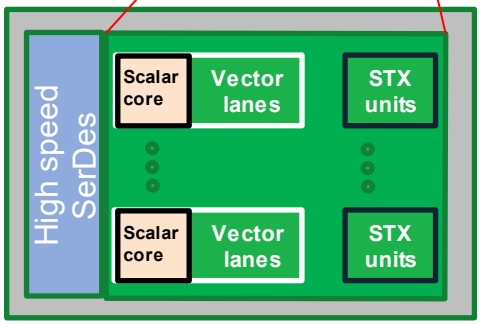
First Generation EPI chips



General Purpose Processor (GPP) chip

- 7 nm, chip-let technology
- **ARM-SVE** tiles
- **EPAC RISC-V** vector+AI accelerator tiles
- L1, L2, L3 cache subsystem + HBM + DDR

GPP power manager based on PULP!



RISC-V Accelerator Demonstrator Test Chip

- 22 nm FDSOI
- Only one RISC-V accelerator tile
- On-chip L1, L2 + off-chip HBM + DDR PHY
- Targets 128 DP GFLOPS (vector) **200+GOPs/W SP (STX)**

Scalar Core + STX units based on PULP!


The fun is
just beginning
http://pulp-platform.org