The Parallel Ultra Low Power Platform

RISC-V Tutorial at HotChips 2019

18 Aug 2019

Fabian Schuiki

and the entire PULP team

[1]Department of Electrical, Electronic
and Information Engineering

[2]Integrated Systems Laboratory

pulp-platform.org

Parallel Ultra Low Power (PULP)

- Project started in 2013 by Luca Benini
- A collaboration between University of Bologna and ETH Zürich
  - Large team. In total we are about 60 people, not all are working on PULP
- Key goal is

How to get the most BANG for the ENERGY consumed in a computing system

- We were able to start with a clean slate, no need to remain compatible to legacy systems.

How we started with open source processors

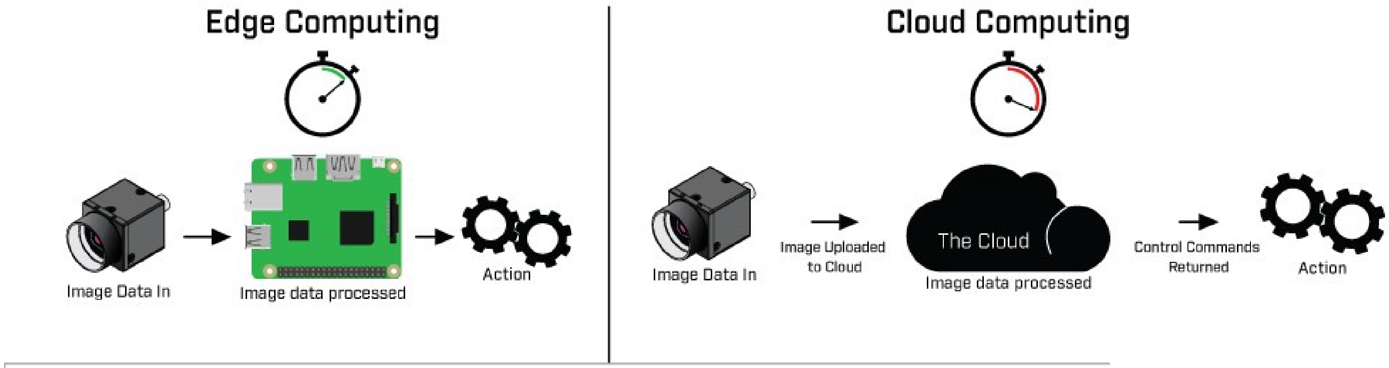
- Our research was not developing processors…
- … but we needed good processors for systems we build for research
- Initially (2013) our options were
  - Build our own (support for SW and tools)
  - Use a commercial processor (licensing, collaboration issues)
  - Use what is openly available (OpenRISC,..)
- We started with OpenRISC
  - First chips until mid-2016 were all using OpenRISC cores
  - We spent time improving the microarchitecture
- Moved to RISC-V later
  - Larger community, more momentum
  - Transition was relatively simple (new decoder)

OpenCores
www.opencores.org

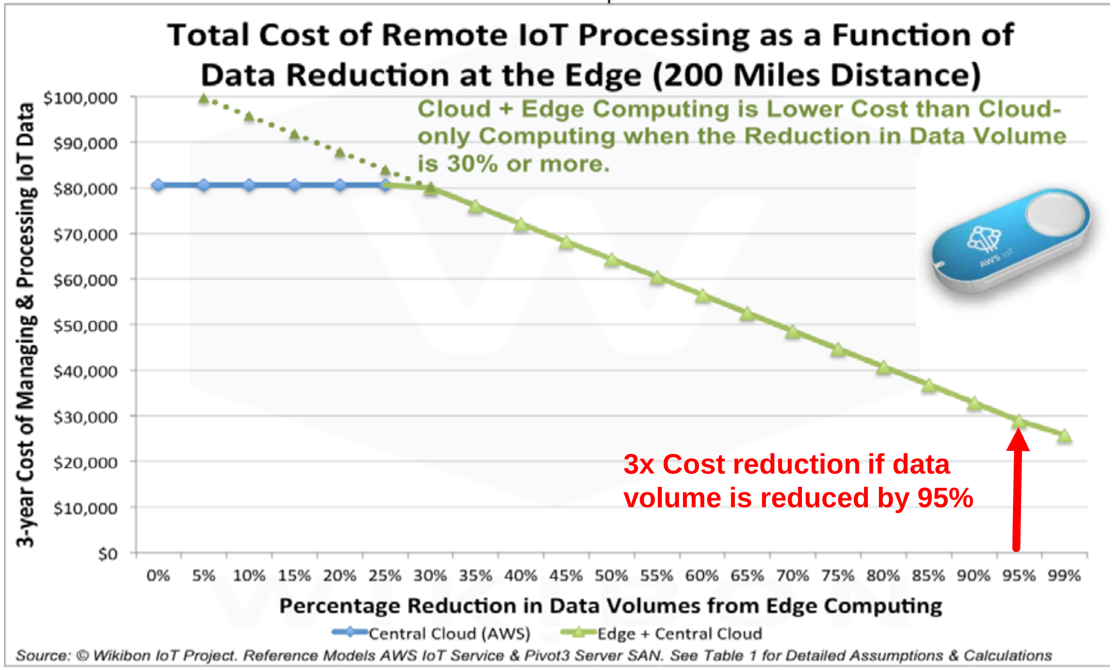
RISC-V

Motivation: Cloud → Edge → Extreme Edge AI

Latency, Privacy



Cost



Total Cost of Remote IoT Processing as a Function of Data Reduction at the Edge (200 Miles Distance)

Cloud + Edge Computing is Lower Cost than Cloud-only Computing when the Reduction in Data Volume is 30% or more.

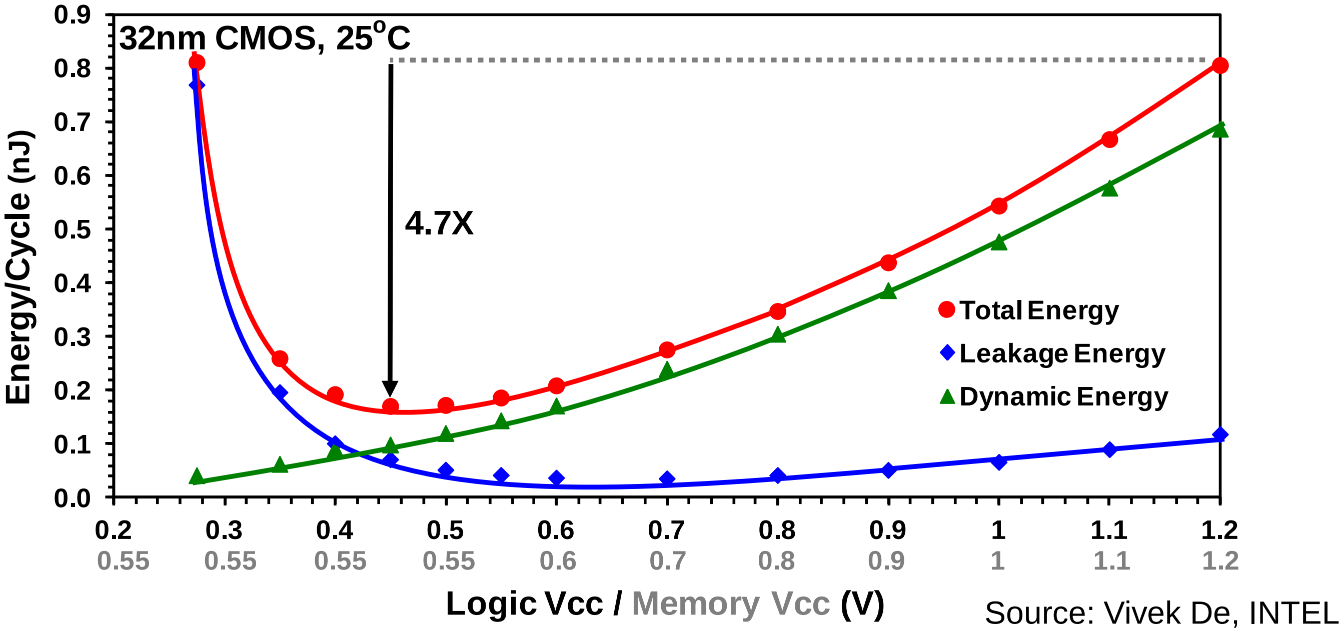
3x Cost reduction if data volume is reduced by 95%

Source: © Wikibon IoT Project. Reference Models AWS IoT Service & Pivot3 Server SAN. See Table 1 for Detailed Assumptions & Calculations

Extreme edge AI challenge:

- **AI capabilities below 1 pJ/op (MCU power envelope)**
- **Mops to Tops**
- **Beyond fp32/fp64**

2013: Parallel Ultra Low Power → PULP!



Source: Vivek De, INTEL – Date 2013

Near-Threshold Computing (NTC):

1. **Don't waste energy pushing devices in strong inversion**
2. **Recover performance with parallel execution**
3. **Core with 'naked' L1 interface to create cluster coupled at L1 level**
4. **Manage Leakage, PVT variability and SRAM limiting NT!**

Need Strong ISA, Need full access to "deep" core interfaces, need to tune pipeline!
OPEN ISA: **RISC-V** RV32IMC + **New, Open Microarchitecture → RI5CY!**

Bespoke ISA needed! Enter Xpulp extensions

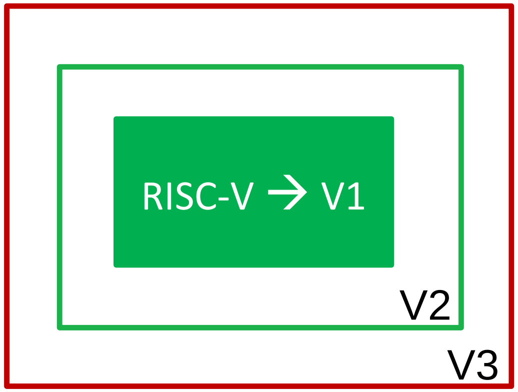
<32-bit precision → **SIMD2/4 → x2,4 efficiency & memory size**

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

V1 Baseline RISC-V RV32IMC

V2 HW loops
Post modified Load/Store
Mac

V3 SIMD 2/4 + DotProduct + Shuffling
Bit manipulation unit
Lightweight fixed point **(EML centric)**



25 kGE → 40 kGE (1.6x)

M. Gautschi et al., "Near-Threshold RISC-V Core With DSP Extensions for Scalable IoT Endpoint Devices," in IEEE TVLSI, Oct. 2017.

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(
   lb     x3, 0(
   addi   x10,x1
   addi   x11,x1
   add    x2, x3
   sb     x2, 0(
   addi   x4, x4
   addi   x12,x1
bne       x4, x5
```

11 cycles/output

Auto-incr load/store

```
mv    x5, 0
mv    x4, 100
Lstart:
   lb    x2, 0(
   lb    x3, 0(
  addi  x4, x4
  add   x2, x3
  sb    x2, 0(
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(x1
```

5 cycles/output

Packed-SIMD

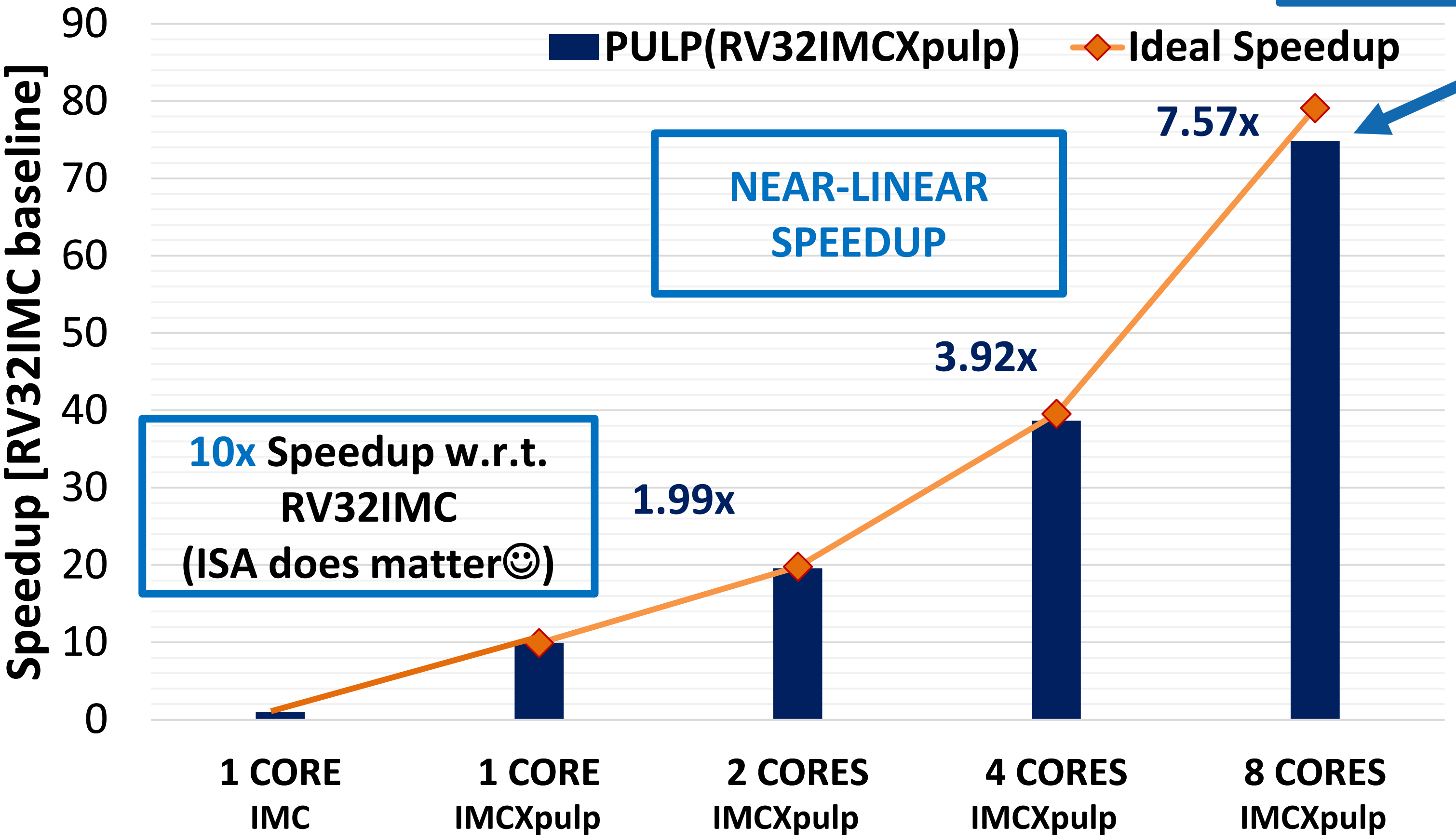
```
lp.setupi 25, Lend
  lw   x2, 0(x10!)
  lw   x3, 0(x11!)
  pv.add.b x2, x3, x2
Lend: sw x2, 0(x12!)
```

1,25 cycles/output

Results: RV32IMCXpulp vs RV32IMC

8-bit Convolution Results

Overall Speedup of **75x**



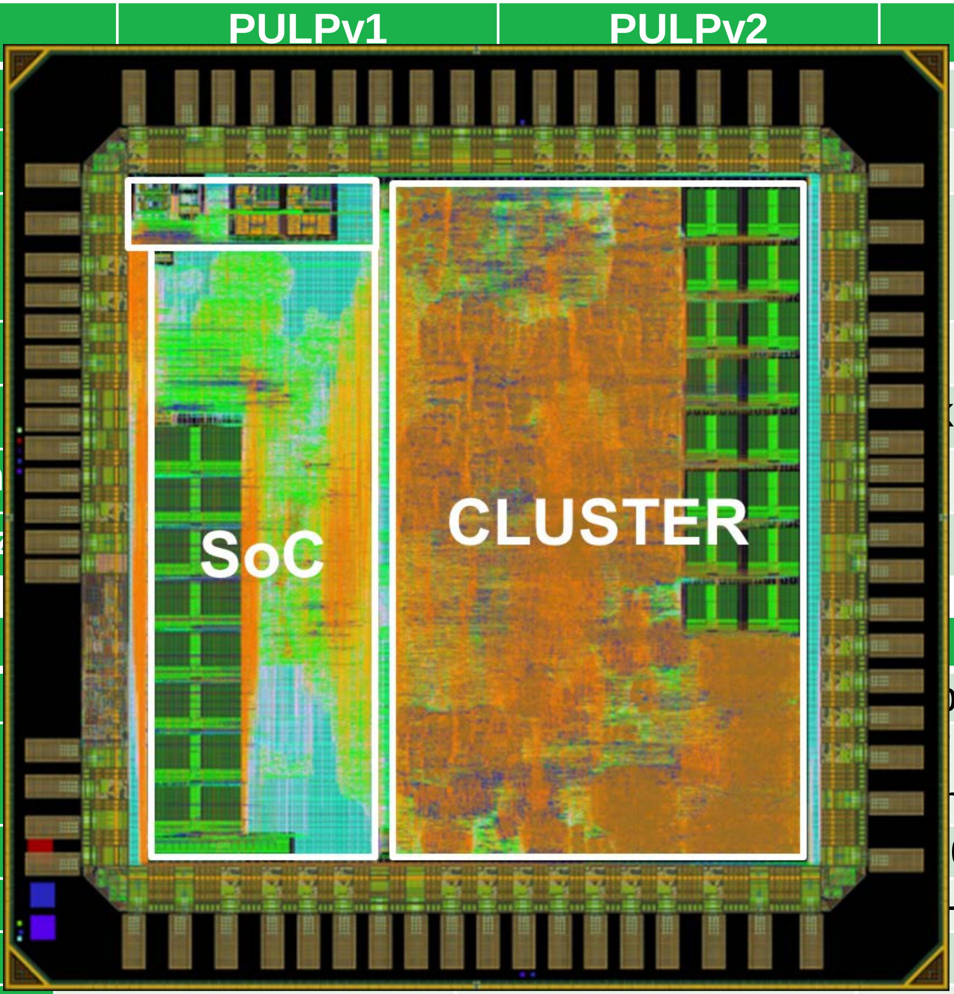
NEAR-LINEAR SPEEDUP

10x Speedup w.r.t. RV32IMC (ISA does matter☺)

PULP-NN: an open Source library for DNN inference on PULP cores

The Evolution of the ‘Species’

	PULPv1	PULPv2	PULPv3
# of cores			4
L2 memory			128 kB
TCDM			32kB SRAM 16kB SCM
DVFS			yes
I$			kB SCM shared
DSP Extension			yes
HW Synchroniz			yes



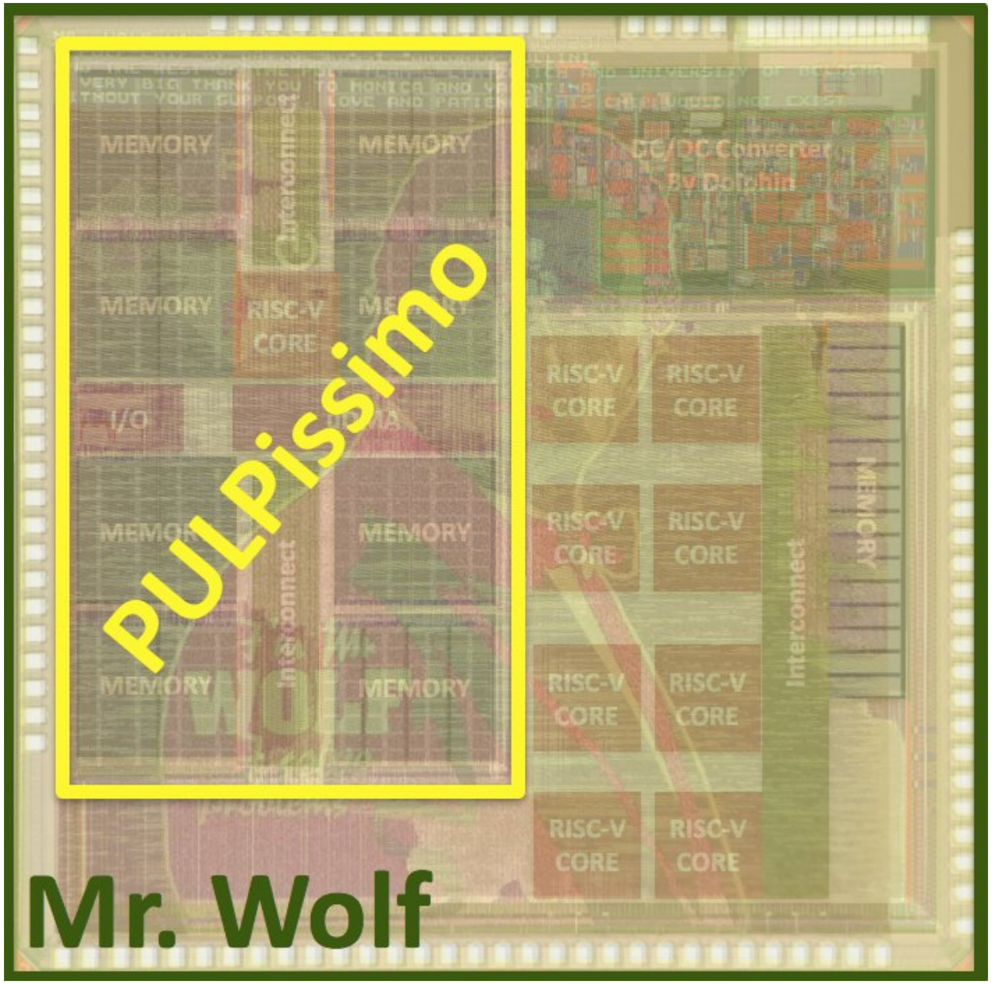
			PULPv3
Status			ost tape out
Technology			D-S nventional well
Voltage range			0.5V - 0.7V
BB range			-1.8V - 0.9V
Max freq.			200 MHz
Max perf.	1.9 GOPS	4 GOPS	1.8 GOPS
Peak en. eff.	60 GOPS/W	135 GOPS/W	385 GOPS/W

2.6pJ/op

Mr. Wolf Chip Results: Heterogeneous Computing Works

Technology	CMOS 40nm LP
Chip area	10 mm^2
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 µW*
SoC Active	*0.8 - 1.1*	*32 kHz - 450 MHz*	*0.97 - 38 mW*
Cluster Active	*0.8 - 1.1*	*32 kHz - 350 MHz*	*1.6 - 153 mW*

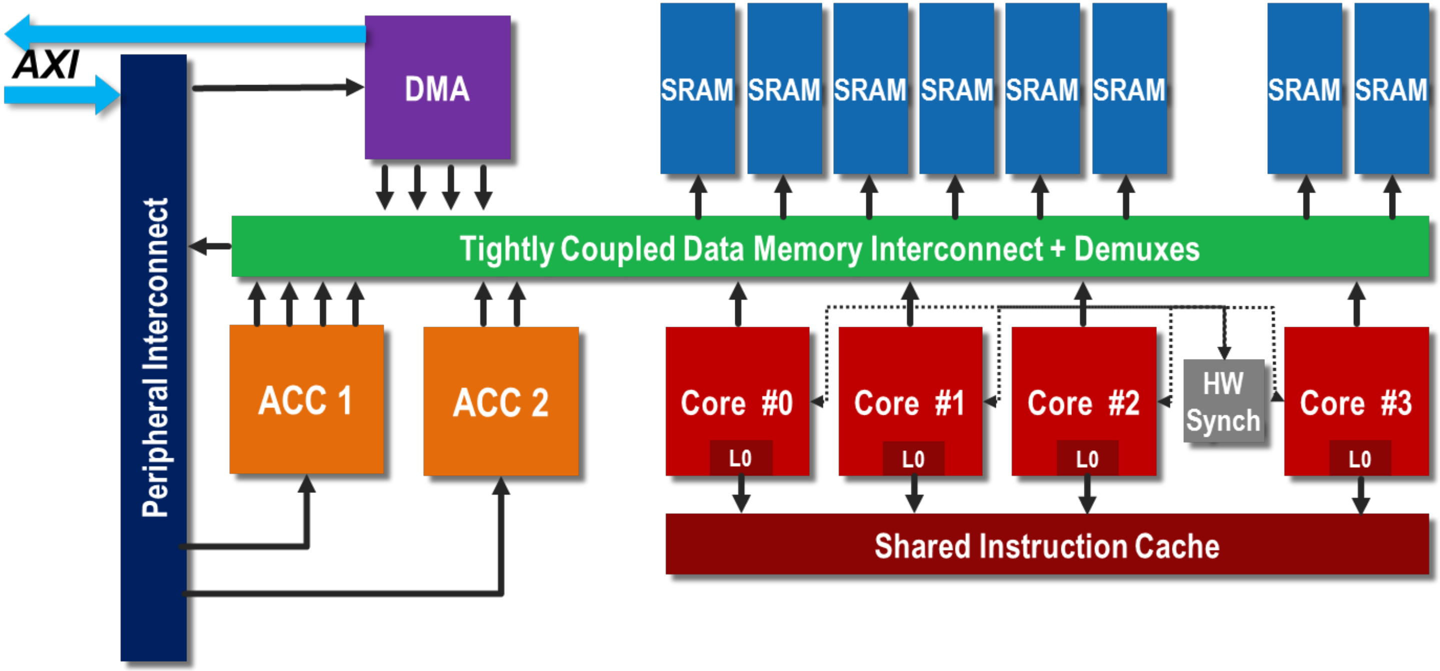


A. Pullini, D. Rossi, I. Loi, A. Di Mauro, L. Benini, "Mr.Wolf: a 1 GFLOP/S Energy-Proportional Parallel Ultra Low Power SoC for IoT Edge Processing", ESSCIRC 2018.

What Kind of Acceleration: Shared memory accelerators

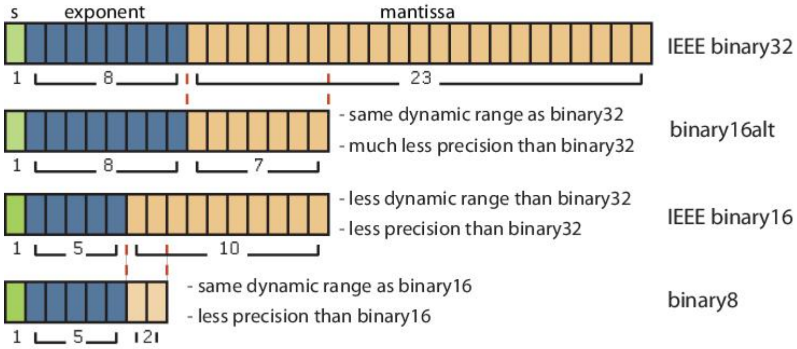
Coarse-Grained Shared-Memory Accelerators

- DFGs mapped In Hardware (ILP + DLP) →Highest Efficiency, Low Flexibility
- Sharing data memory with processor for fast communication → low overhead
- Controlled through a memory-mapped interface
- Typically one/two accelerators shared by multiple cores



What About Floating Point Support?

- F (single precision) and
  D (double precision) extension in RISC-V
- Uses separate floating point register file
  - specialized float loads (also compressed)
  - float moves from/to integer register file
- Fully IEEE compliant
- Alternative FP Format support (<32 bit)



IEEE binary32

- same dynamic range as binary32
- much less precision than binary32

binary16alt

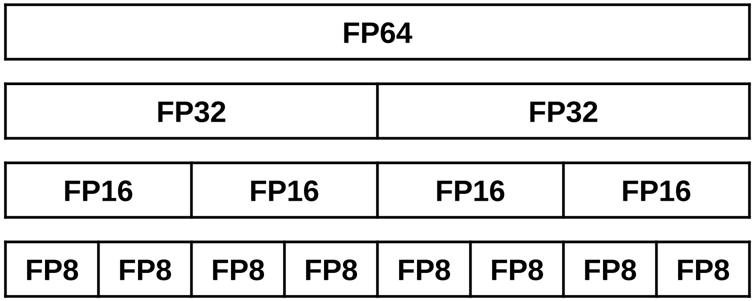
- less dynamic range than binary32
- less precision than binary32

IEEE binary16

- same dynamic range as binary16
- less precision than binary16

binary8

Packed-SIMD support for all formats



FP64

FP32 | FP32

FP16 | FP16 | FP16 | FP16

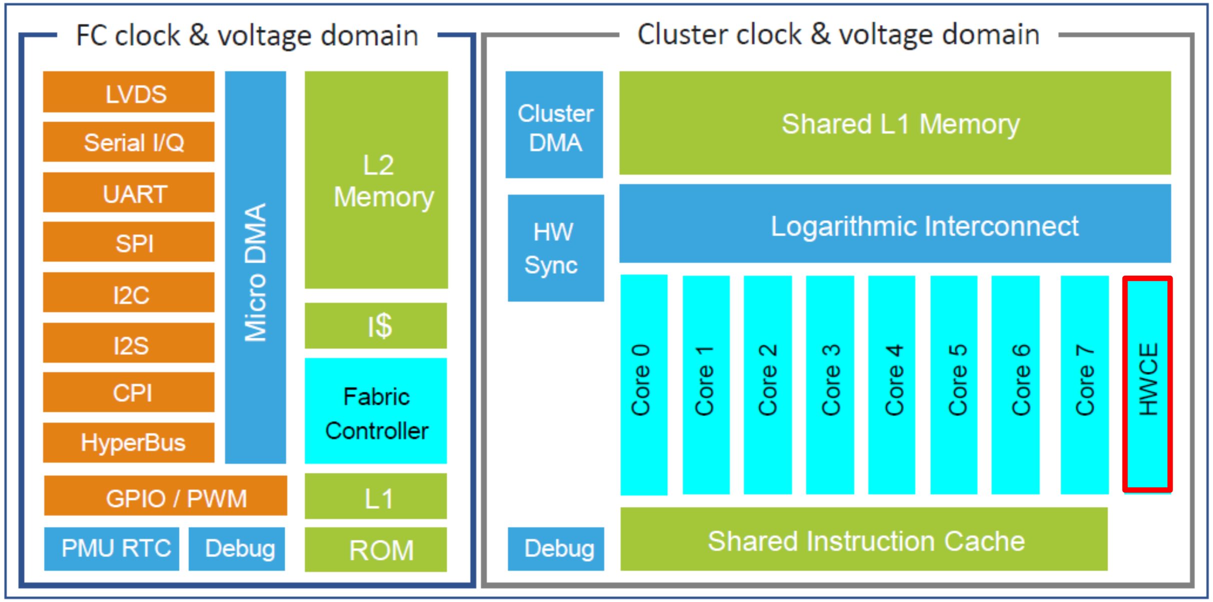
FP8 | FP8 | FP8 | FP8 | FP8 | FP8 | FP8 | FP8

Unified FP/Integer register file

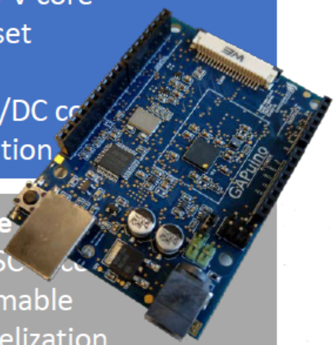
- Not standard
- up to 15 % better performance
  - Re-use integer load/stores (post incrementing ld/st)
  - Less area overhead
  - Useful if pressure on register file is not very high (true for a lot of applications)

PULP cluster+MCU+HWCE(V1) → GWT's GAP8 (55 TSMC)

Two independent clock and voltage domains, from 0-133MHz/1V up to 0-250MHz/1.2V



MCU Function
- Extended RISC-V core
- Extensive I/O set
- Micro DMA
- Embedded DC/DC co
- Secured execution



Computation engine
- 8 extended RISC co
- Fully programmable
- Efficient parallelization
- Shared instruction cache
- Multi channel DMA
- HW synchronization
- → HW convolution Engine

What	Freq MHz	Exec Time ms	Cycles	Power mW
40nm Dual Issue MCU	216	99.1	21 400 000	60
GAP8 @1.0V	15.4	99.1	1 500 000	3.7
GAP8 @1.2V	175	8.7	1 500 000	70
GAP8 @1.0V w HWCE	4.7	99.1	460 000	0.8

11 X

16 X

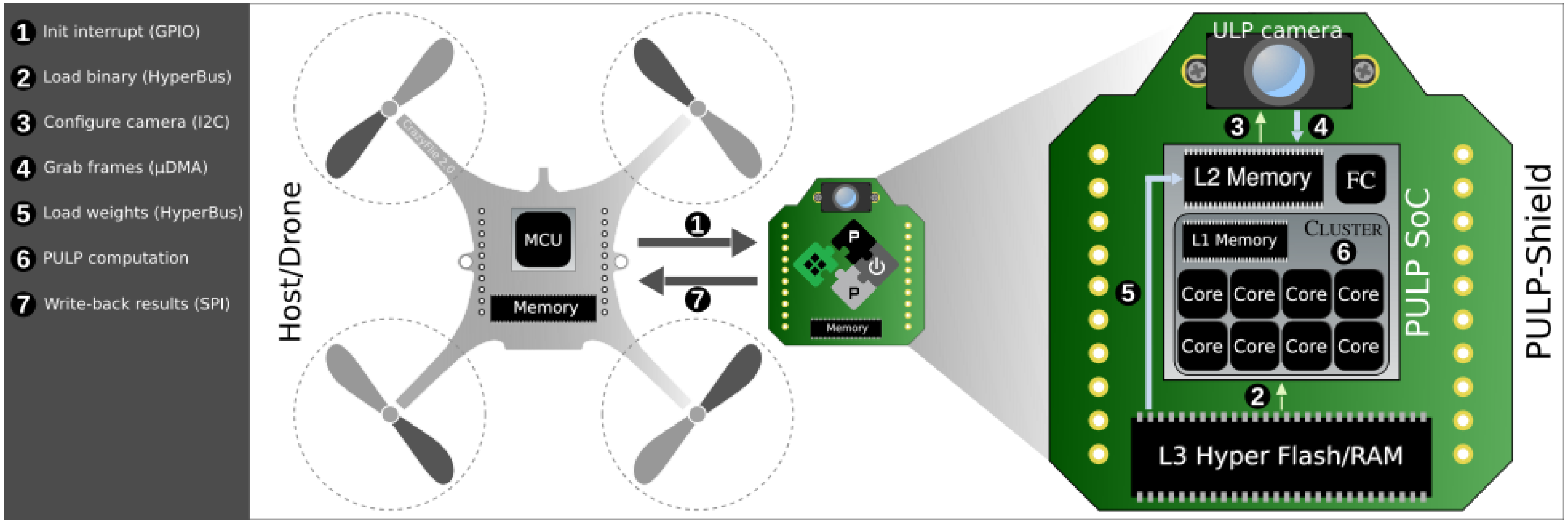
GREENWAVES TECHNOLOGIES

4x More efficiency at less than 10% area cost

PULP

ETH

New Application Frontiers: DroNET on NanoDrone

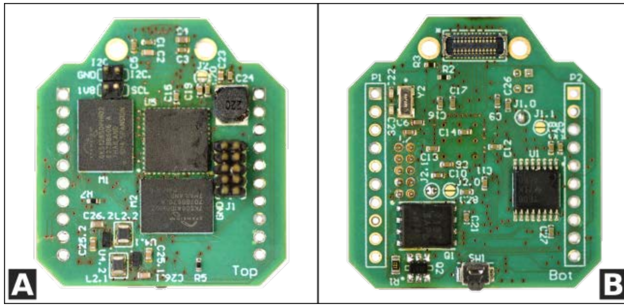


1 Init interrupt (GPIO)
2 Load binary (HyperBus)
3 Configure camera (I2C)
4 Grab frames (µDMA)
5 Load weights (HyperBus)
6 PULP computation
7 Write-back results (SPI)

Pluggable PCB: PULP-Shield

- ~5g, 30×28mm
- GAP8 SoC
- 8 MB HDRAM
- 16 MB HFlash
- QVGA ULP HiMax camera
- Crazyflie 2.0 nano-drone (27g)

Credit: Frank K. Gürkaynak & Daniele Palossi

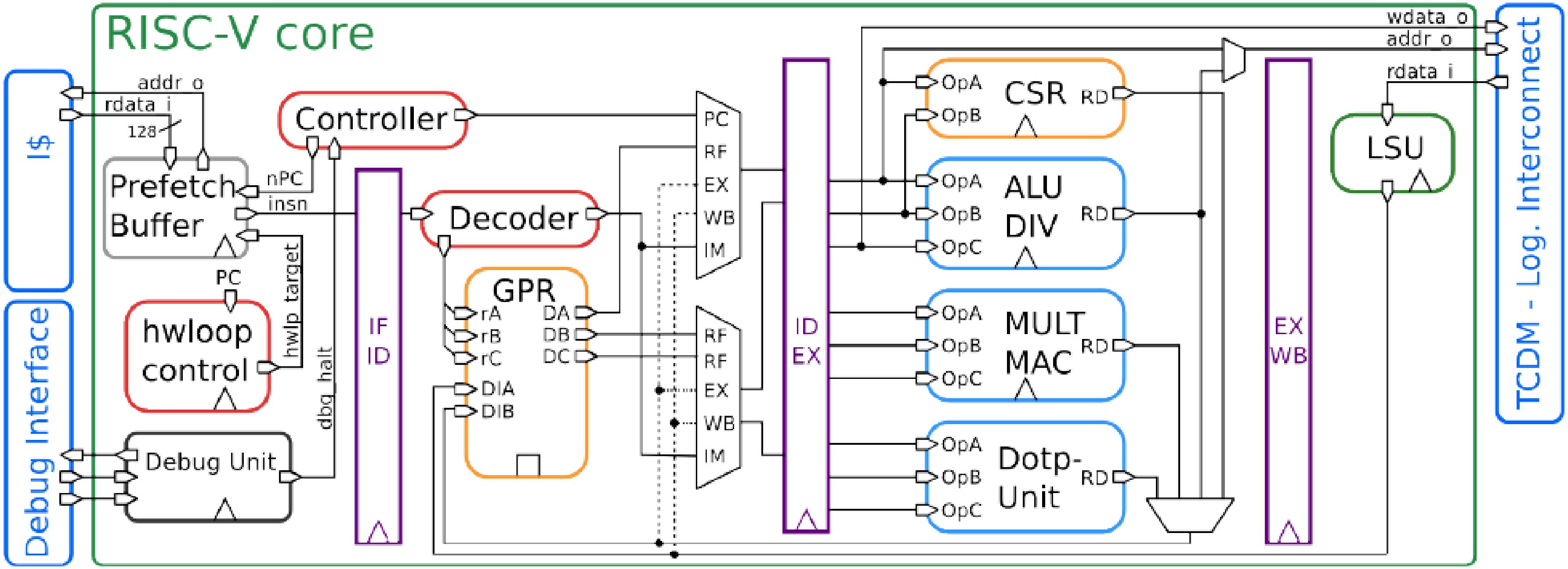


Only onboard computation for autonomous flight + obstacle avoidance
no human operator, no ad-hoc external signals, and no remote base-station!

https://youtu.be/57Vy5cSvnaA

The Cores

RI5CY – Our workhorse 32-bit core



- 4-stage pipeline, optimized for energy efficiency
- 40 kGE, 30 logic levels, Coremark/MHZ 3.19
- Includes various extensions (Xpulp) to RISC-V for DSP applications

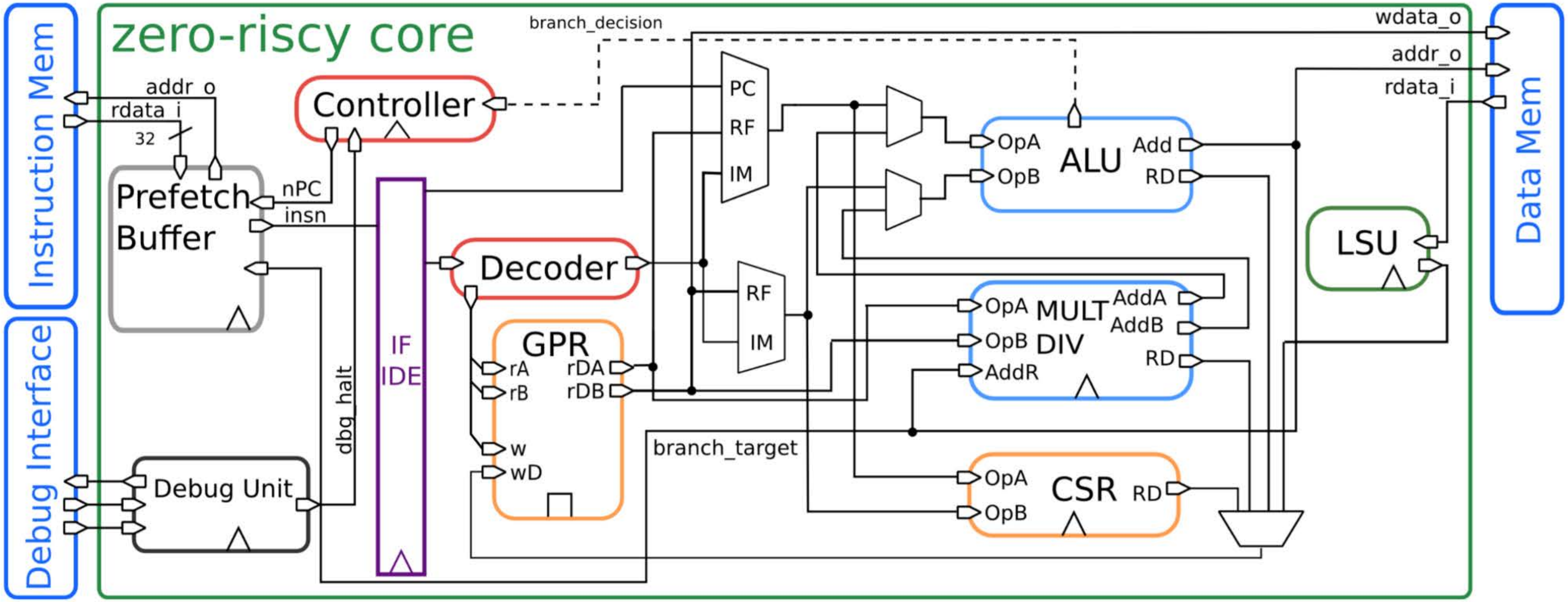
Our extensions to RI5CY (with additions to GCC)

- **Post–incrementing** load/store instructions
- **Hardware Loops** (`lp.start`, `lp.end`, `lp.count`)
- **ALU instructions**
  - Bit manipulation (count, set, clear, leading bit detection)
  - Fused operations: (add/sub-shift)
  - Immediate branch instructions
- **Multiply Accumulate** (32x32 bit and 16x16 bit)
- **SIMD instructions** (2x16 bit or 4x8 bit) with scalar replication option
  - add, min/max, dotproduct, shuffle, pack (copy), vector comparison

For 8-bit values the following can be executed in a single cycle (**pv.dotup.b**)

$$Z = D_1 \times K_1 + D_2 \times K_2 + D_3 \times K_3 + D_4 \times K_4$$

Enter Zero/Micro-riscy (Ibex), small core for control

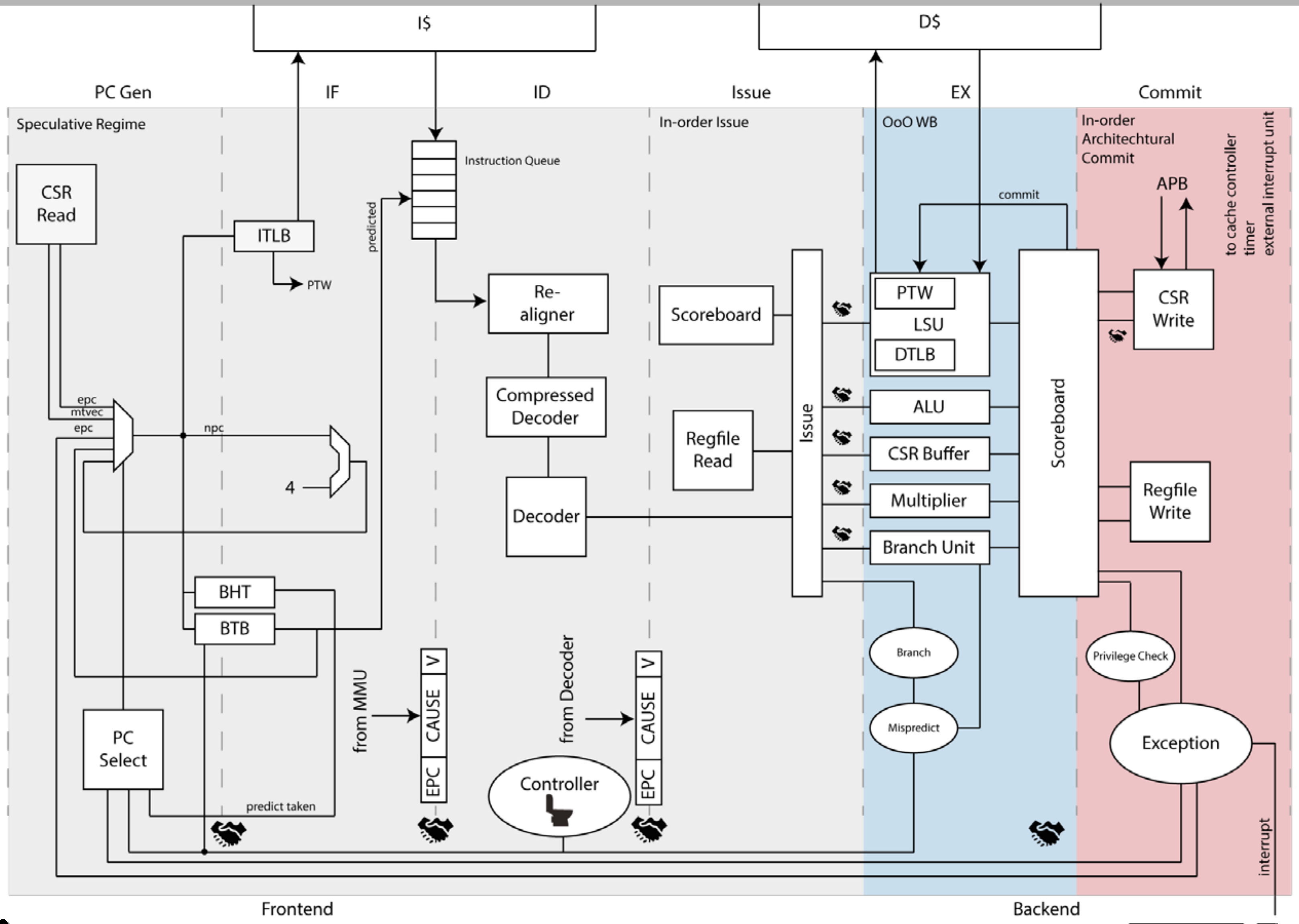


- Only 2-stage pipeline, simplified register file
- **Zero-Riscy** (RV32-ICM), 19kGE, 2.44 Coremark/MHz
- Micro-Riscy (RV32-EC), 12kGE, 0.91 Coremark/MHz
- Used as SoC level controller in newer PULP systems

Finally the step into 64-bit cores

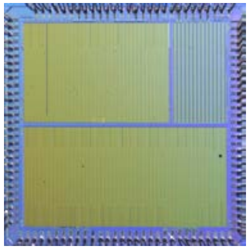
- For the first 4 years of the PULP project we used only 32bit cores
  - Luca once famously said “*We will never build a 64bit core*”.
  - Most IoT applications work well with 32bit cores.
  - A typical 64bit core is much more than 2x the size of a 32bit core.
- But times change:
  - Using a 64bit Linux capable core allows you to share the same address space as main stream processors.
    - We are involved in several projects where we (are planning to) use this capability
  - There is a lot of interest in the security community for working on a contemporary open source 64bit core.
  - Open research questions on how to build systems with multiple cores.

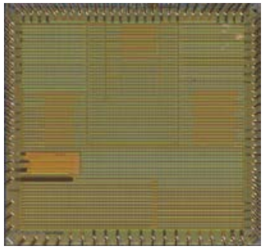
ARIANE: Our Linux Capable 64-bit core

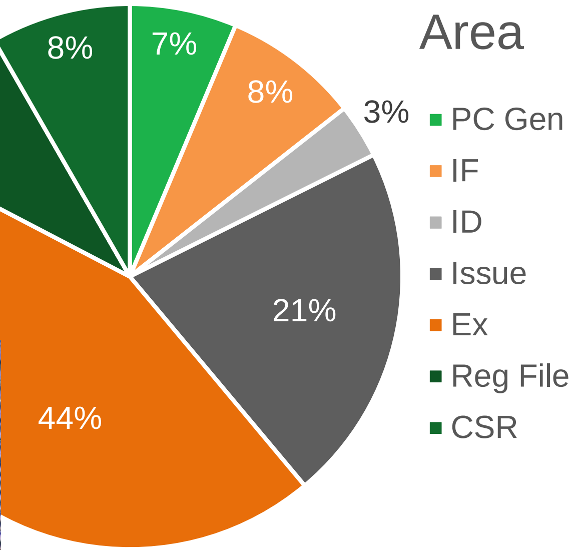


Main properties of Ariane

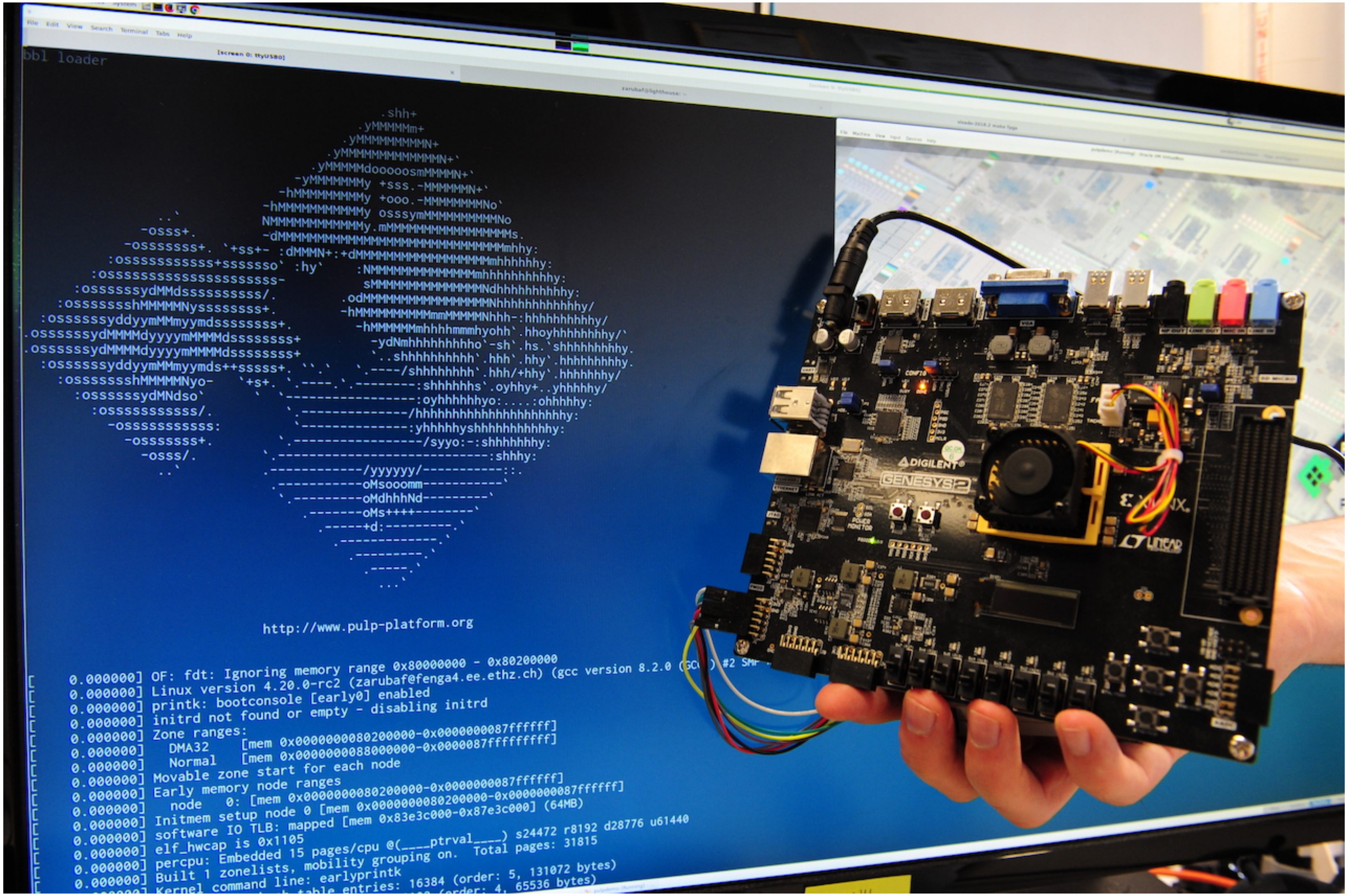
- Tuned for high frequency, 6 stage pipeline, integrated cache
  - In order issue, out-of-order write-back, in-order-commit
  - Supports privilege spec 1.11, M, S and U modes
  - Hardware Page Table Walker
- Implemented in GF 22FDX (Poseidon, Kosmodrom, Baikonur), and UMC65 (Scarabaeus)
  - In 22nm: ~1 GHz worst case conditions (SSG, 125/-40C, 0.72V)
  - 8-way 32kByte Data cache and 4-way 32kByte Instruction Cache
  - Core area: 175 kGE





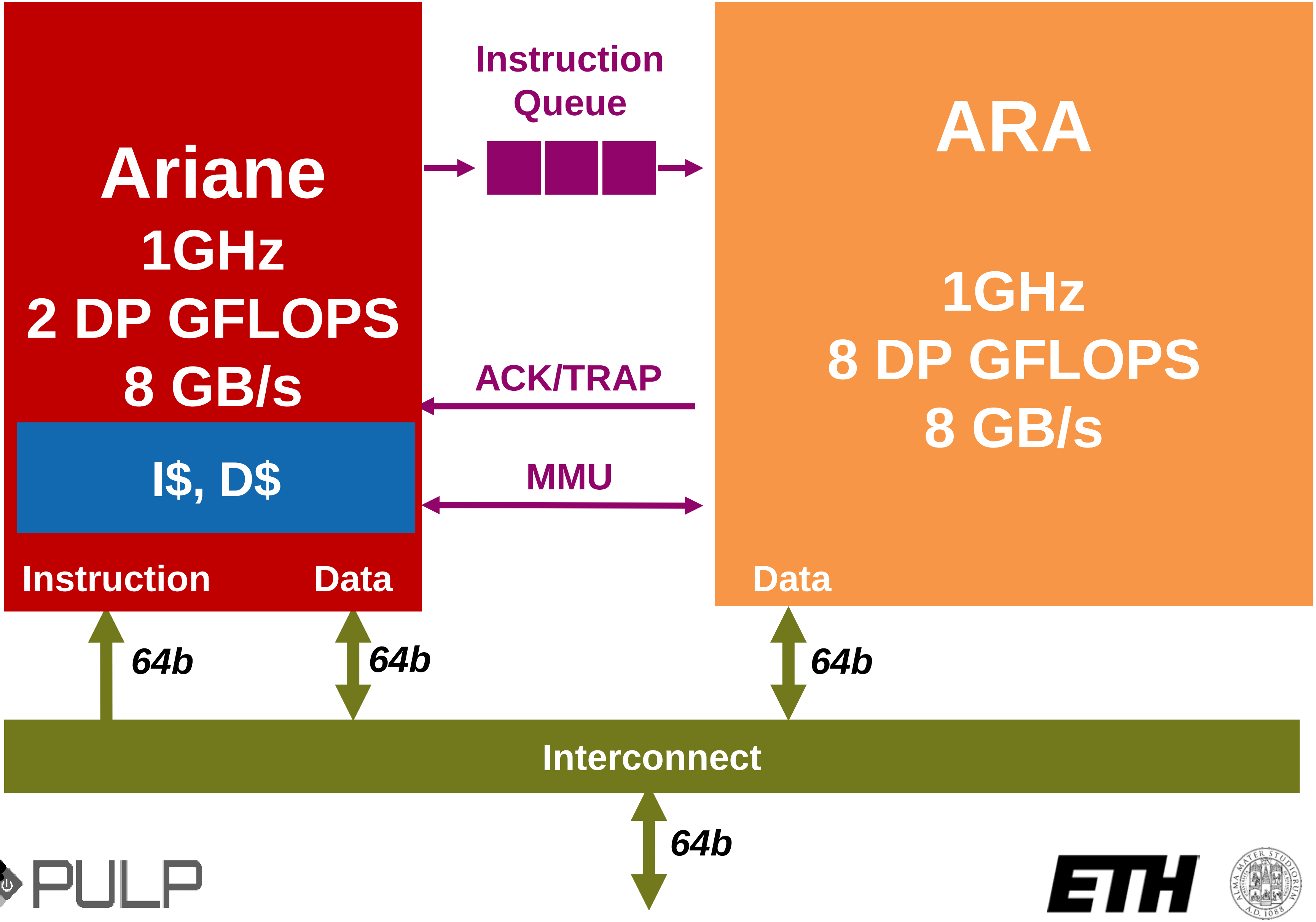


Ariane booting Linux on a Digilent Genesys 2 board

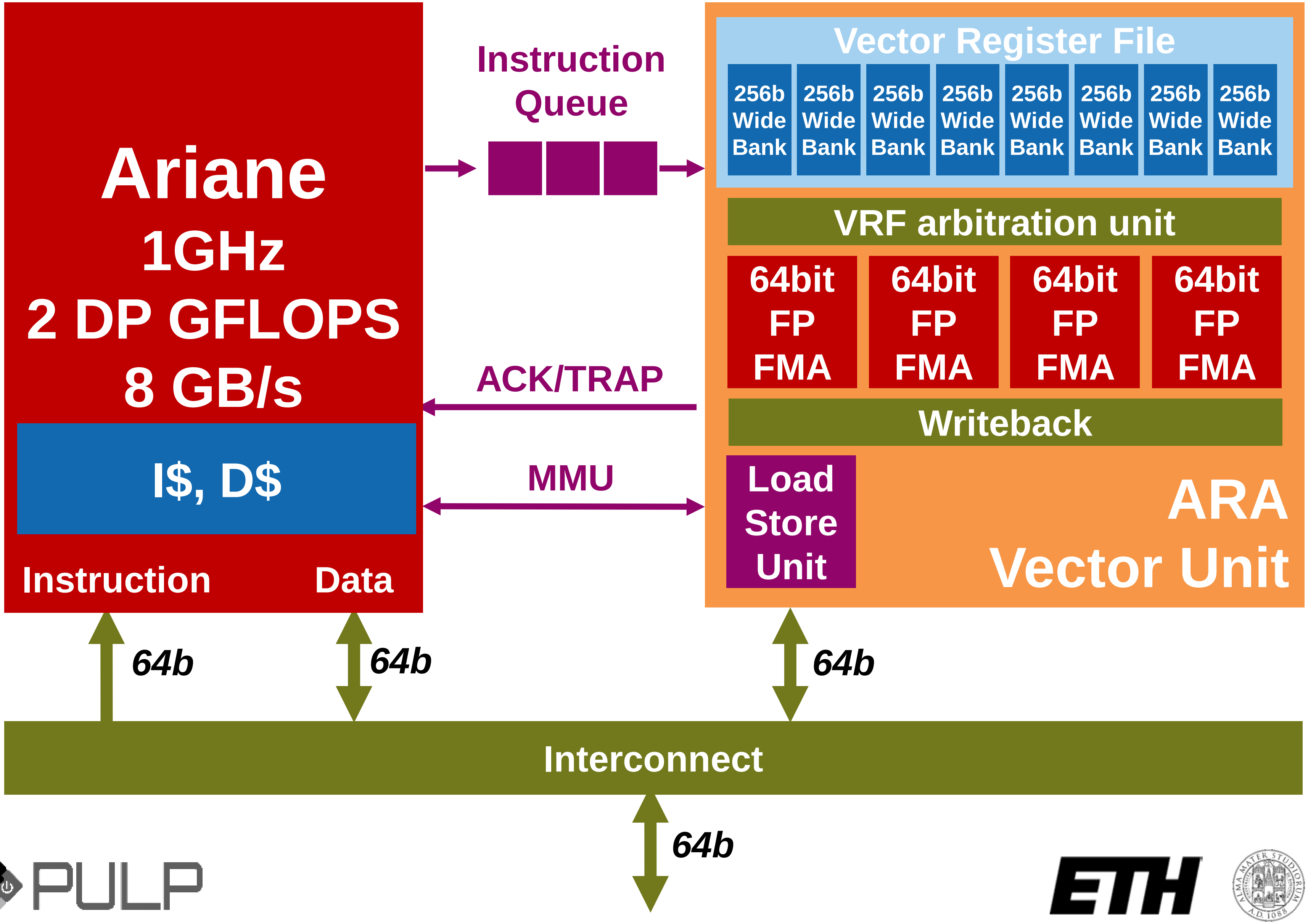


```
bbl loader

http://www.pulp-platform.org

[    0.000000] OF: fdt: Ignoring memory range 0x80000000 - 0x80200000
[    0.000000] Linux version 4.20.0-rc2 (zarubaf@fenga4.ee.ethz.ch) (gcc version 8.2.0 (GC ) #2 SMP
[    0.000000] printk: bootconsole [early0] enabled
[    0.000000] initrd not found or empty - disabling initrd
[    0.000000] Zone ranges:
[    0.000000]   DMA32    [mem 0x0000000080200000-0x0000000087ffffff]
[    0.000000]   Normal   [mem 0x0000000088000000-0x0000087fffffffff]
[    0.000000] Movable zone start for each node
[    0.000000] Early memory node ranges
[    0.000000]   node   0: [mem 0x0000000080200000-0x0000000087ffffff]
[    0.000000] Initmem setup node 0 [mem 0x0000000080200000-0x0000000087ffffff]
[    0.000000] software IO TLB: mapped [mem 0x83e3c000-0x87e3c000] (64MB)
[    0.000000] elf_hwcap is 0x1105
[    0.000000] percpu: Embedded 15 pages/cpu @(____ptrval____) s24472 r8192 d28776 u61440
[    0.000000] Built 1 zonelists, mobility grouping on.  Total pages: 31815
[    0.000000] Kernel command line: earlyprintk
                        table entries: 16384 (order: 5, 131072 bytes)
                                              (order: 4, 65536 bytes)
```

Extreme FP Performance: The “V” Extension

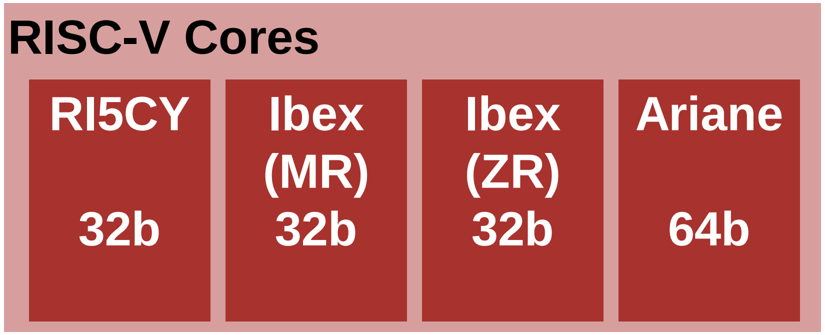


Extreme FP Performance: The "V" Extension



The Platforms

Making PULP: Cores



RISC-V Cores

RI5CY 32b

Ibex (MR) 32b

Ibex (ZR) 32b

Ariane 64b

Making PULP: Cores + Peripherals/Acc.

RISC-V Cores

- RI5CY 32b
- Ibex (MR) 32b
- Ibex (ZR) 32b
- Ariane 64b

Peripherals

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

Interconnect

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

Accelerators

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

Making PULP: Cores + Peripherals/Acc. = Platforms

RISC-V Cores

RI5CY 32b	Ibex (MR) 32b	Ibex (ZR) 32b	Ariane 64b

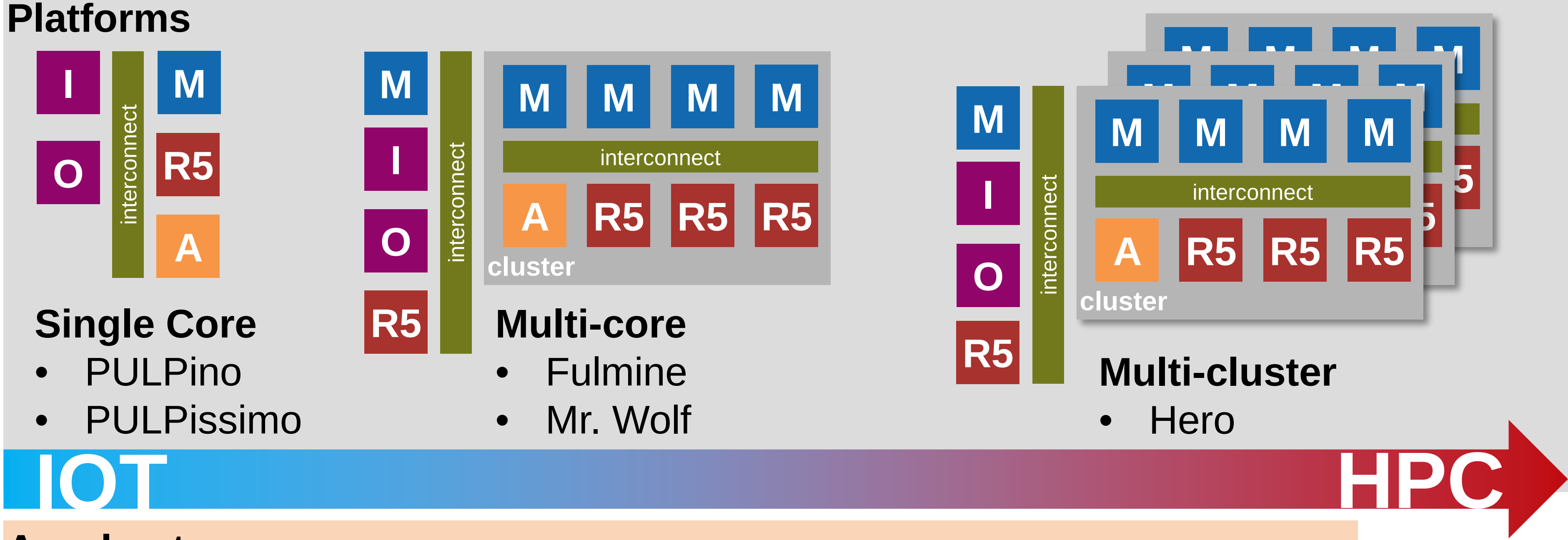
Peripherals

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

Accelerators

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

The PULP platforms put everything together

RISC-V Cores

RI5CY 32b

Ibex (MR) 32b

Ibex (ZR) 32b

Ariane 64b

Peripherals

JTAG

SPI

UART

I2S

DMA

GPIO

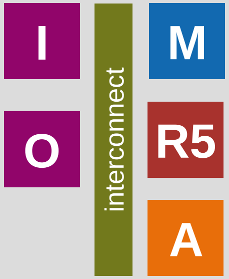
Interconnect

Logarithmic interconnect

APB – Peripheral Bus

AXI4 – Interconnect

Platforms



I

O

interconnect

M

R5

A

Single Core

- PULPino
- PULPissimo

Accelerators

HWCE (convolution)

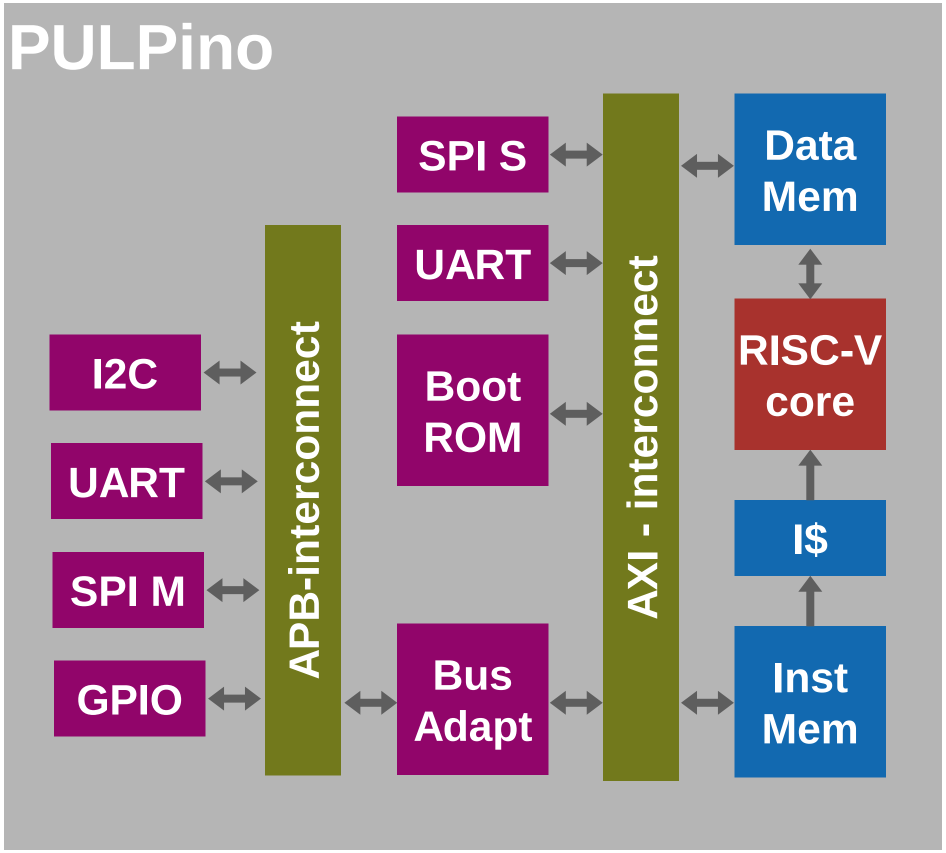
Neurostream (ML)

HWCrypt (crypto)

PULPO (1st order opt)

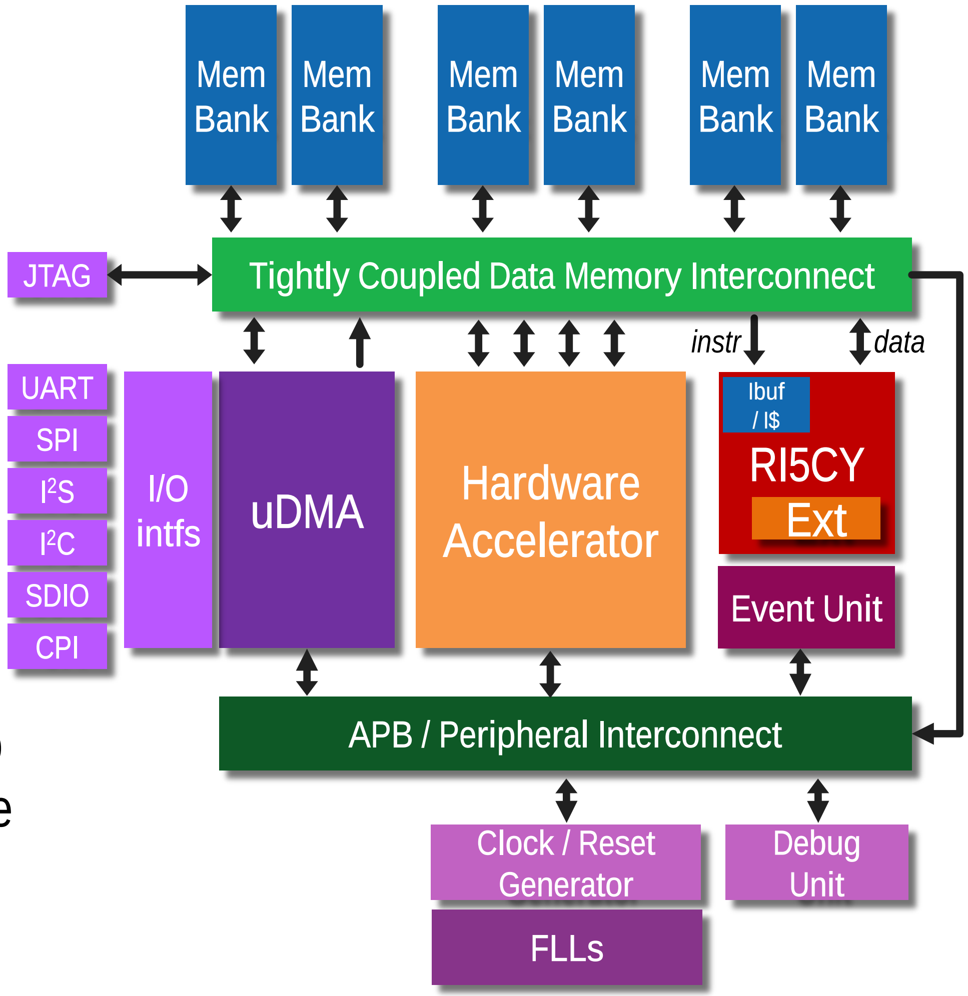
PULPino: Our first single core platform

- Simple design
  - Meant as a quick release
- Separate Data and Instruction memory
  - Makes it easy in HW
  - Not meant as a Harvard arch.
- Can be configured to work with all our 32bit cores
  - RI5CY, Zero/Micro-Riscy (Ibex)
- Peripherals copied from its larger brothers
  - Any AXI and APB peripherals could be used



PULPissimo: The improved single core platform

- Shared memory
  - Unified Data/Instruction Memory
  - Uses the multi-core infrastructure
- Support for Accelerators
  - Direct shared memory access
  - Programmed through APB bus
  - Number of TCDM access ports determines max. throughput
- uDMA for I/O subsystem
  - Can copy data directly from I/O to memory without involving the core
- Used as a SoC/fabric controller in larger systems



The main PULP systems we develop are cluster based

RISC-V Cores

- RI5CY 32b
- Ibex (MR) 32b
- Ibex (ZR) 32b
- Ariane 64b

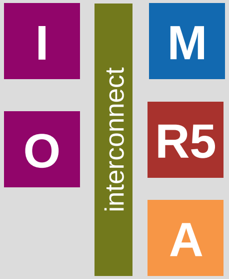
Peripherals

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

Interconnect

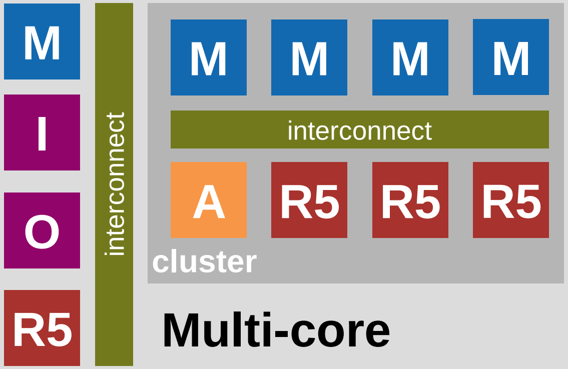
- Logarithmic interconnect
- APB – Peripheral Bus
- AXI4 – Interconnect

Platforms



Single Core

- PULPino
- PULPissimo



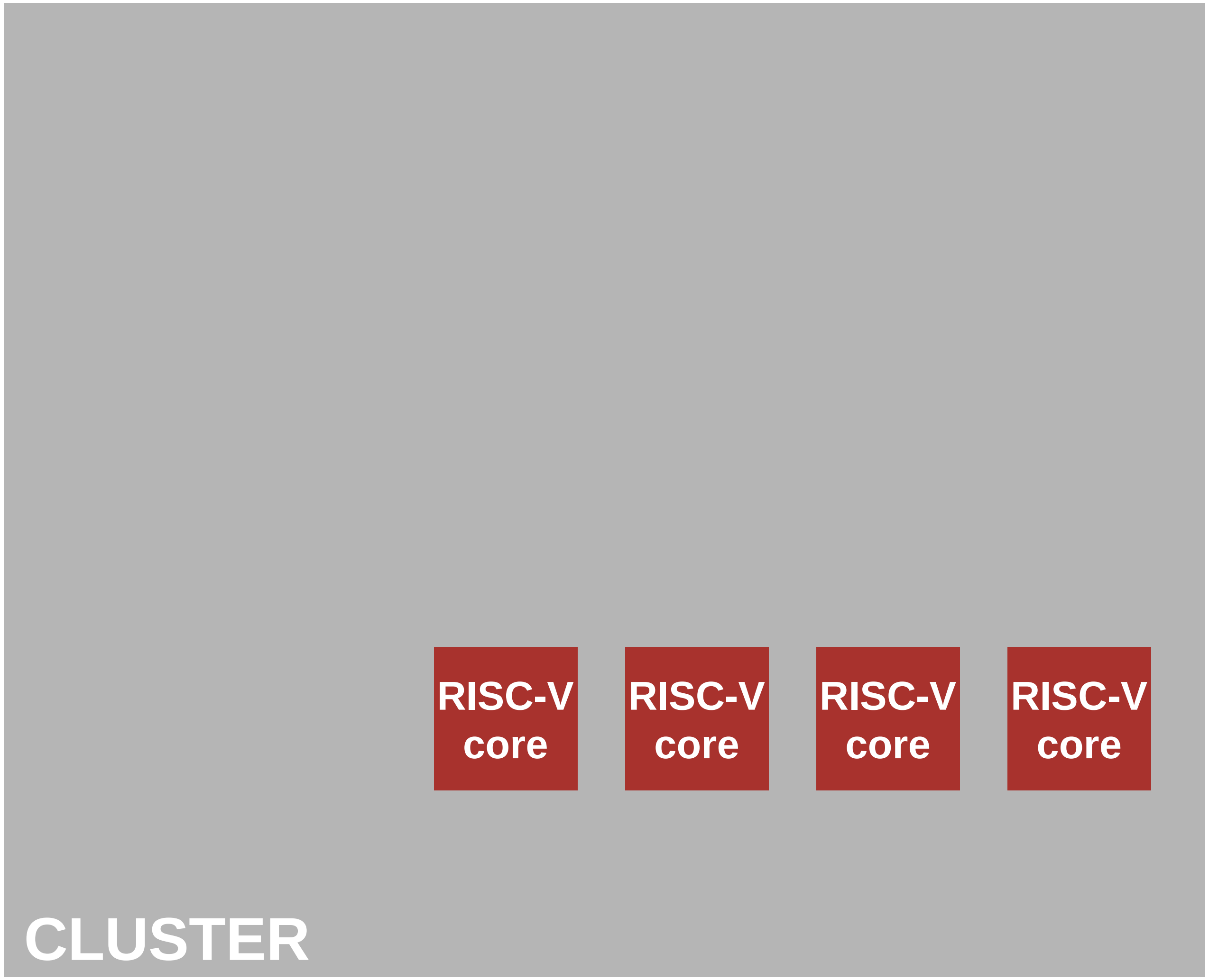
Multi-core

- Fulmine
- Mr. Wolf

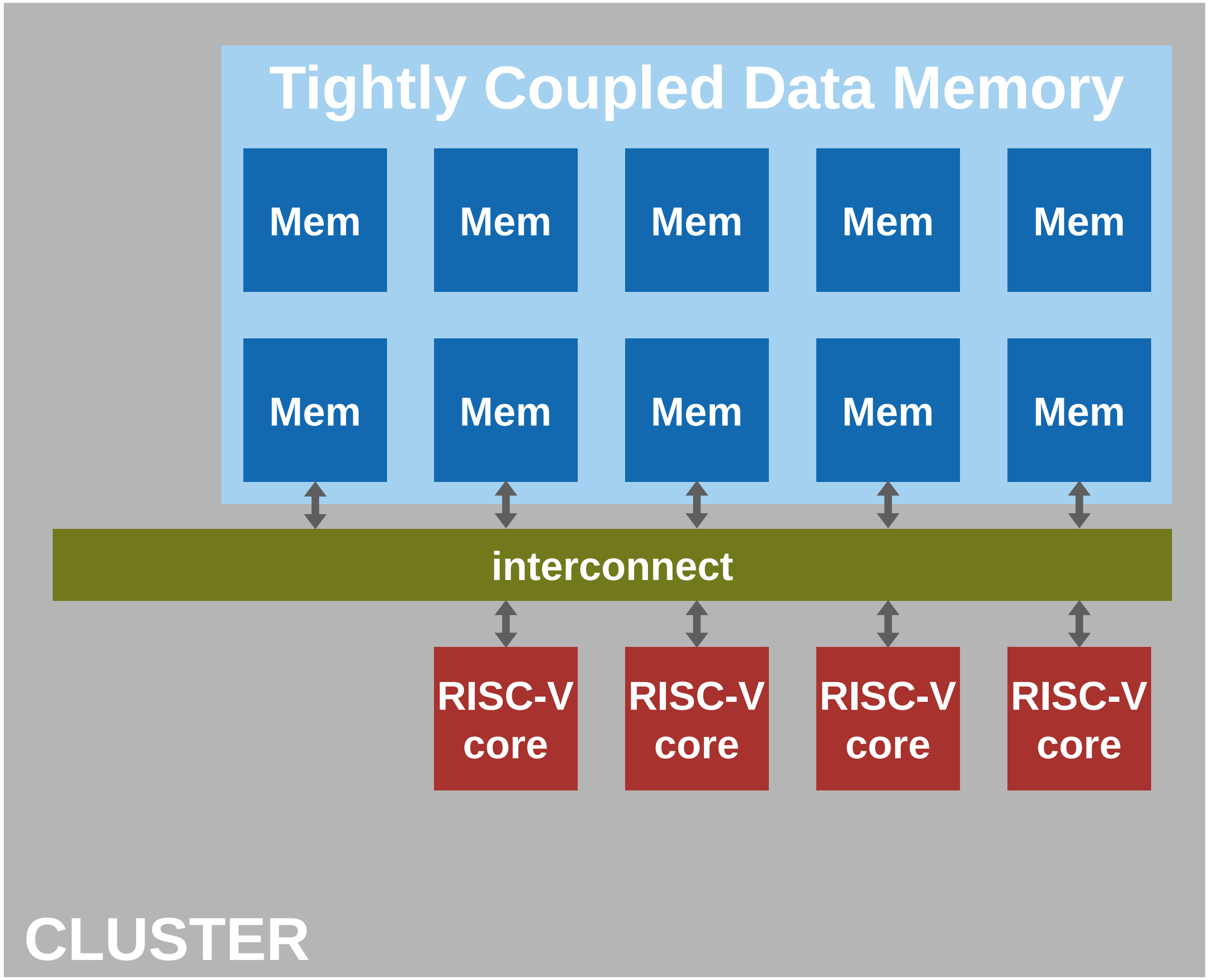
Accelerators

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

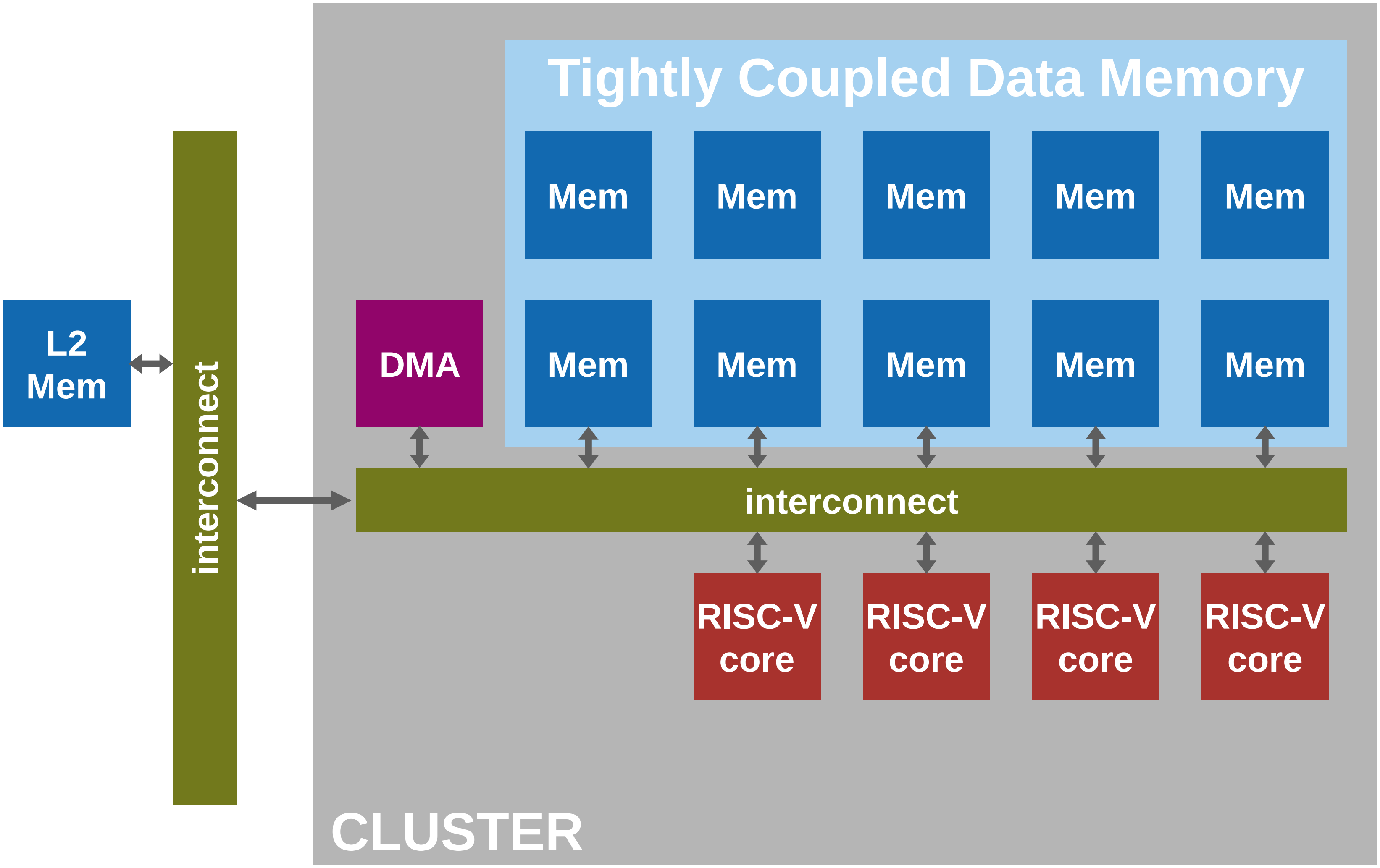
PULP cluster contains multiple RISC-V cores



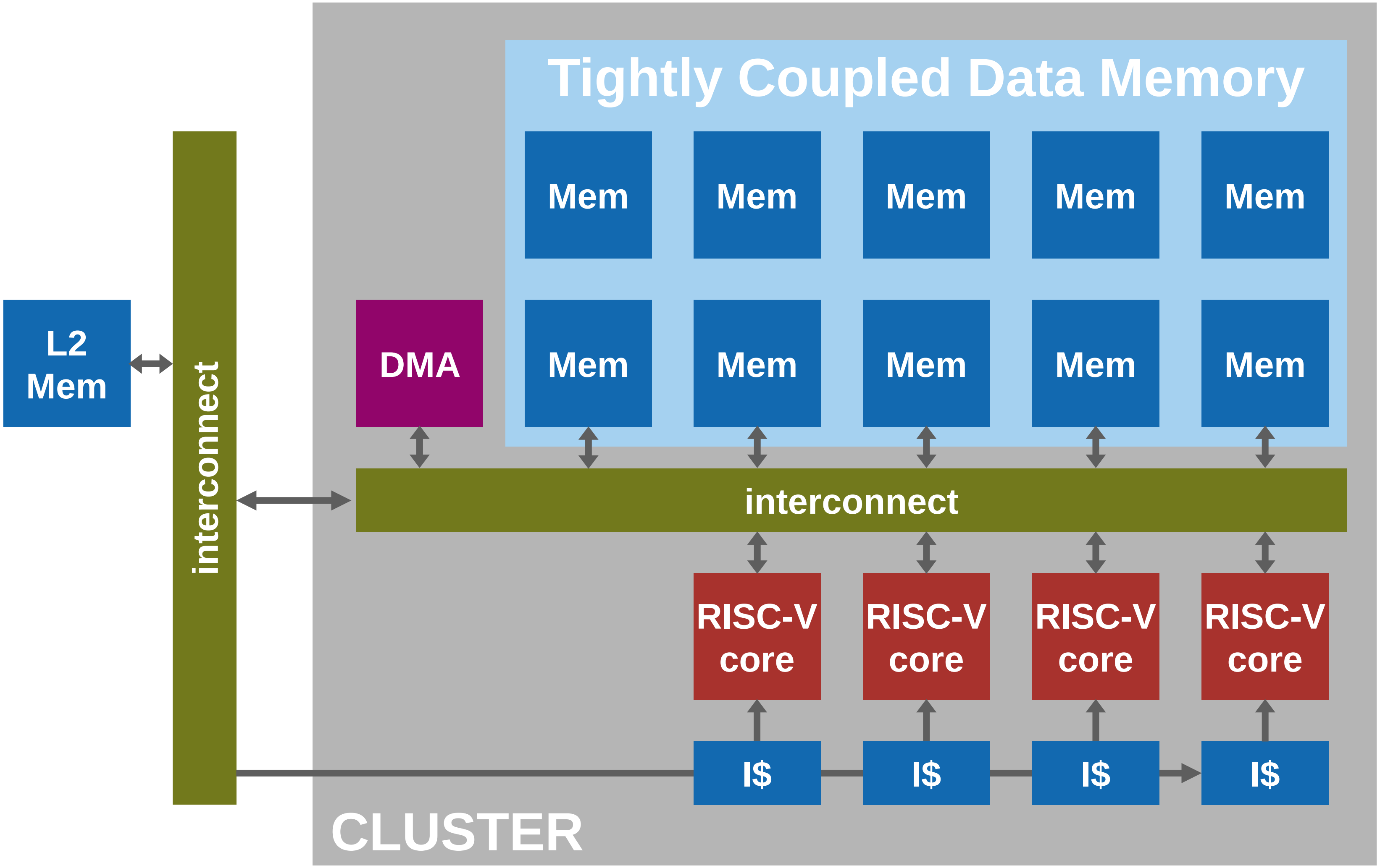
All cores can access all memory banks in the cluster



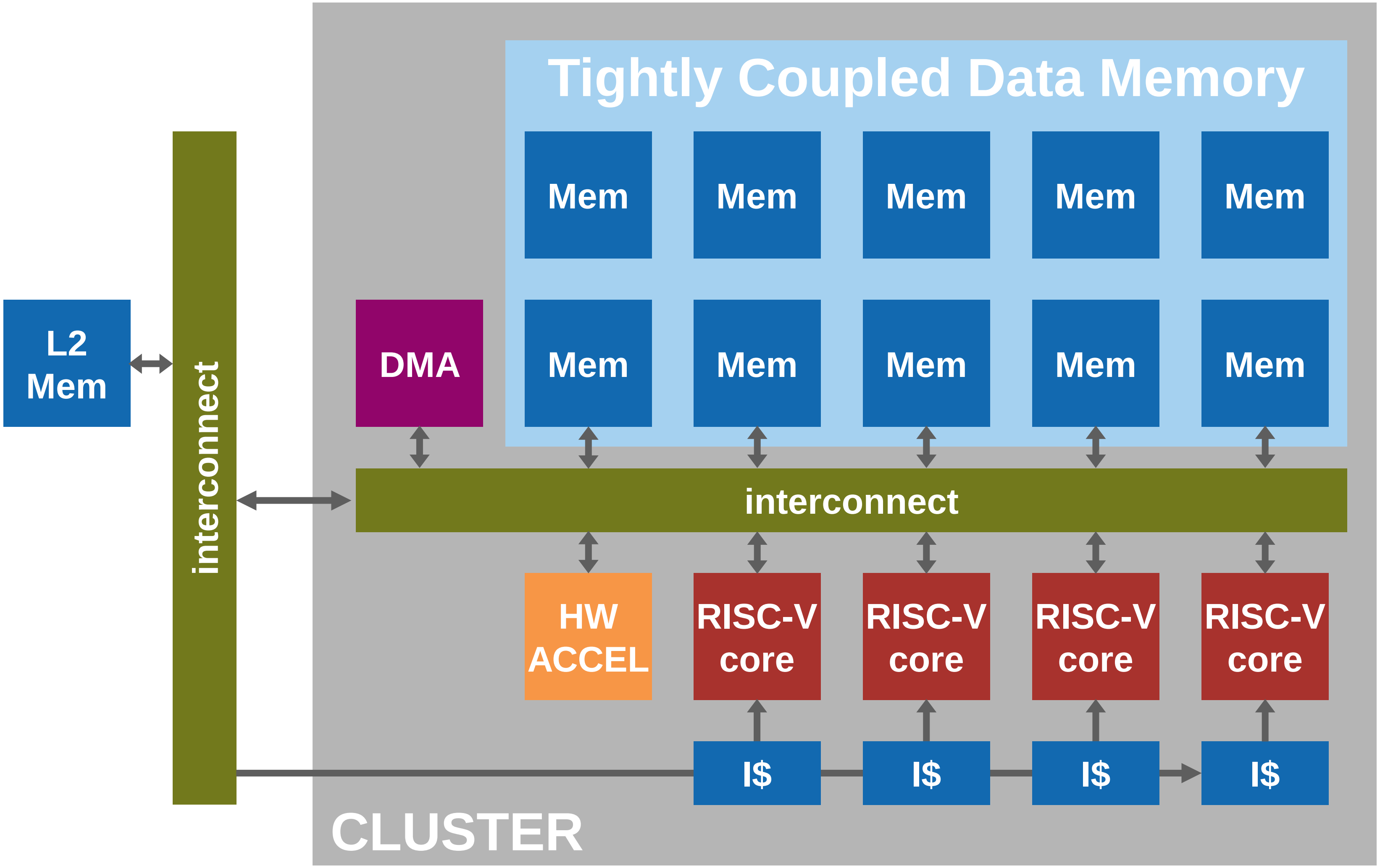
Data is copied from a higher level through DMA



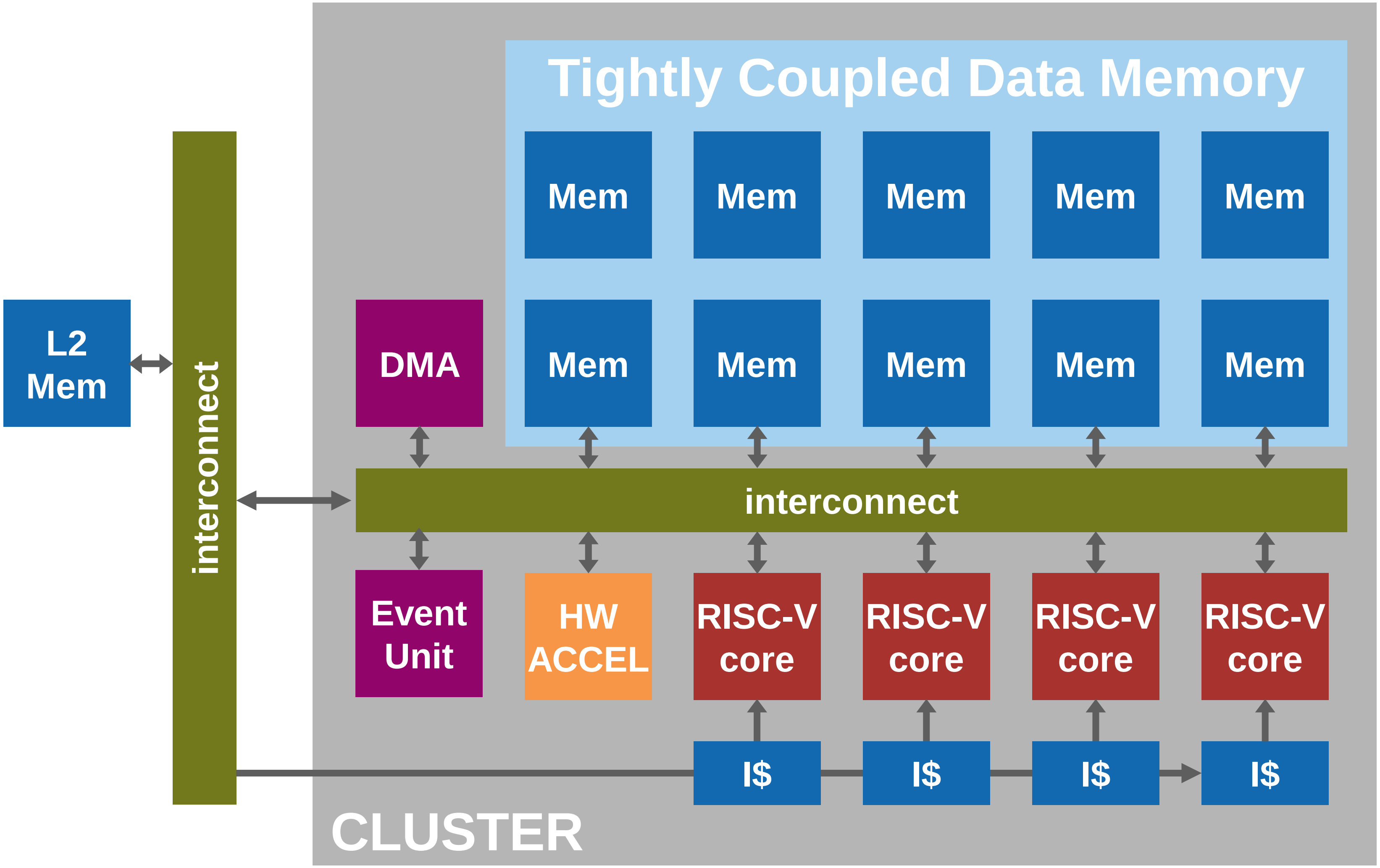
There is a (shared) instruction cache that fetches from L2



Hardware Accelerators can be added to the cluster



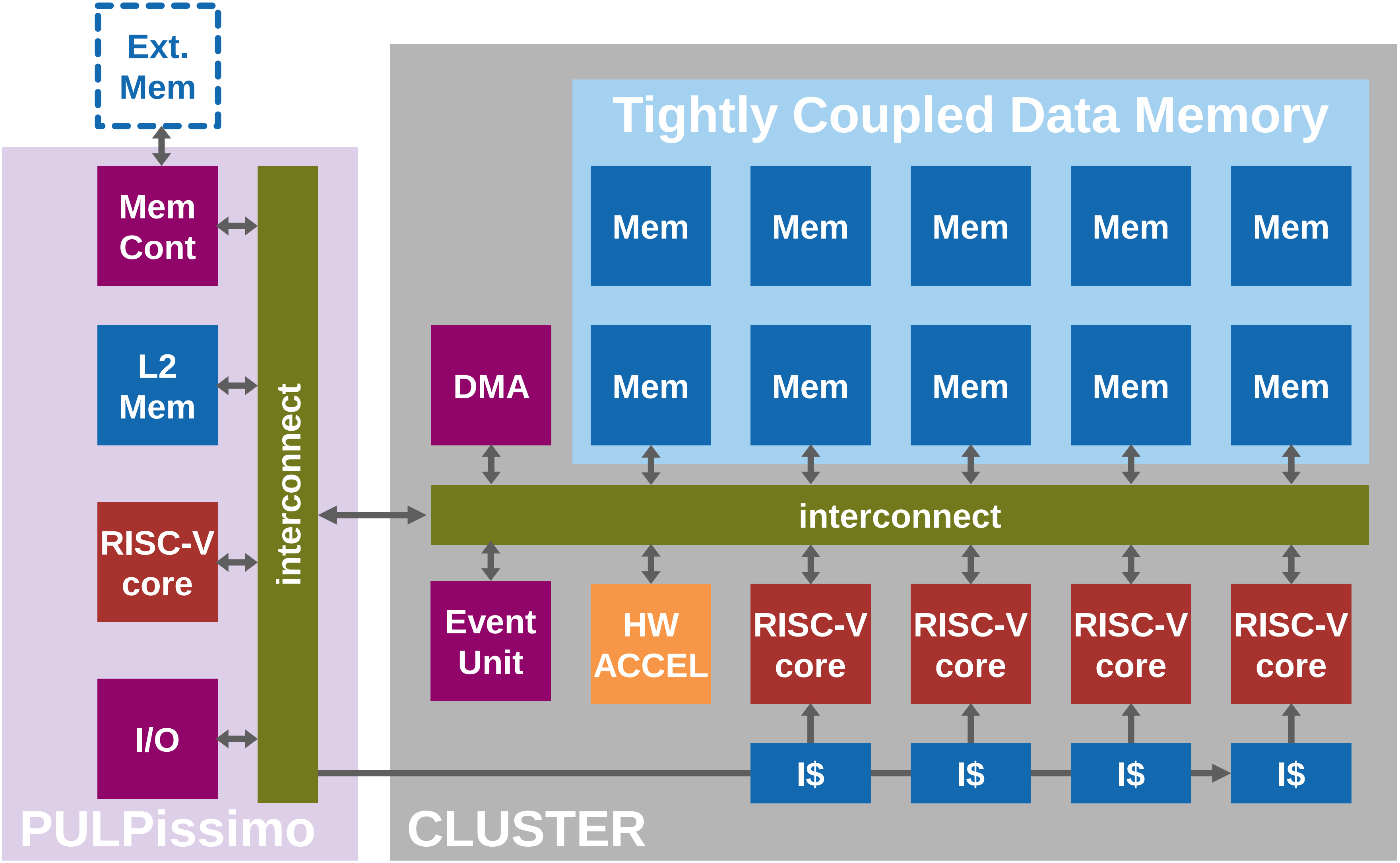
Event unit to manage resources (fast sleep/wakeup)



PULP

ETH

An additional microcontroller system (PULPissimo) for I/O



Finally multi-cluster PULP systems for HPC applications

RISC-V Cores

RI5CY	Ibex (MR)	Ibex (ZR)	Ariane
32b	32b	32b	64b

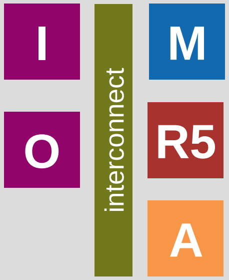
Peripherals

JTAG	SPI
UART	I2S
DMA	GPIO

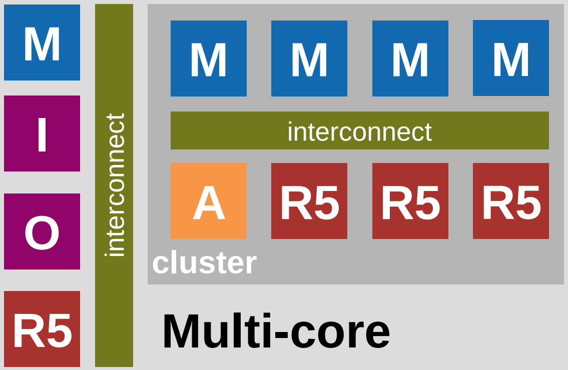
Interconnect

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

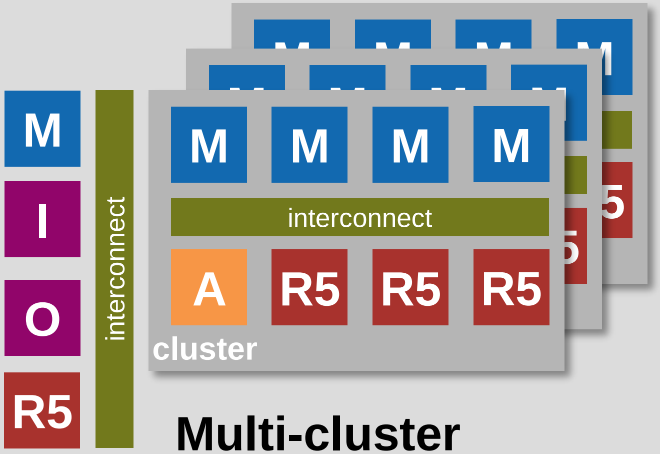
Platforms



Single Core
- PULPino
- PULPissimo



Multi-core
- Fulmine
- Mr. Wolf



Multi-cluster
- Hero

IOT → HPC

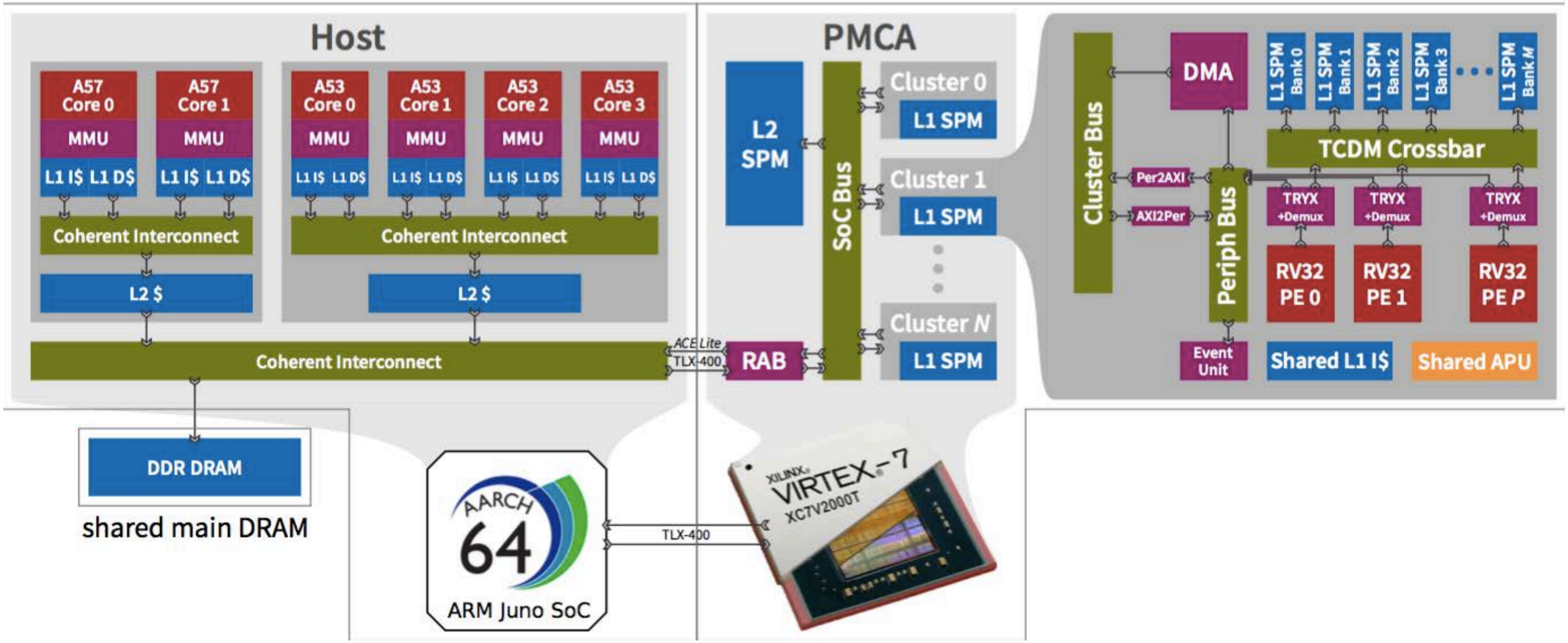
Accelerators

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

Heterogeneous Research Platform

industry-standard, hard-macro
Arm Cortex-A Host processor

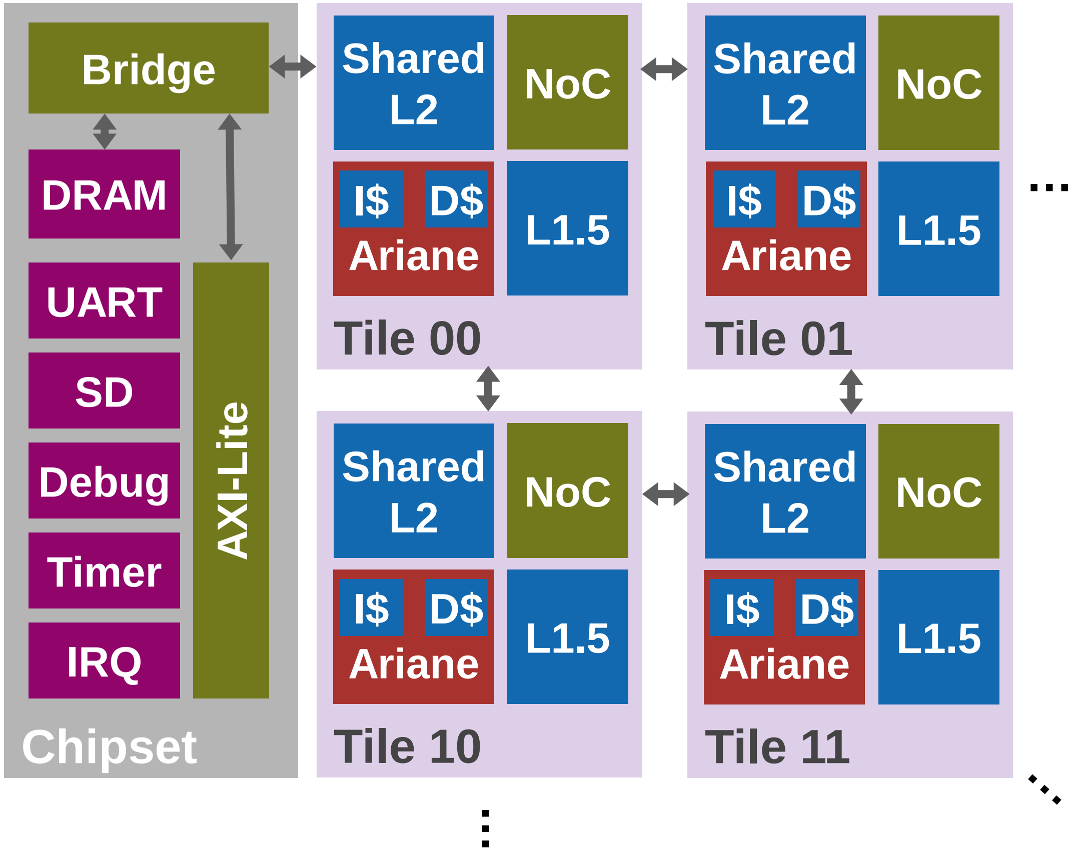
scalable, configurable, modifiable FPGA implementation
of PULP (silicon-proven, cluster-based PMCA with RISC-V PEs)



- First released in 2018
- Allows a PULP cluster to be connected to a host system

OpenPiton and Ariane together, the many-core system

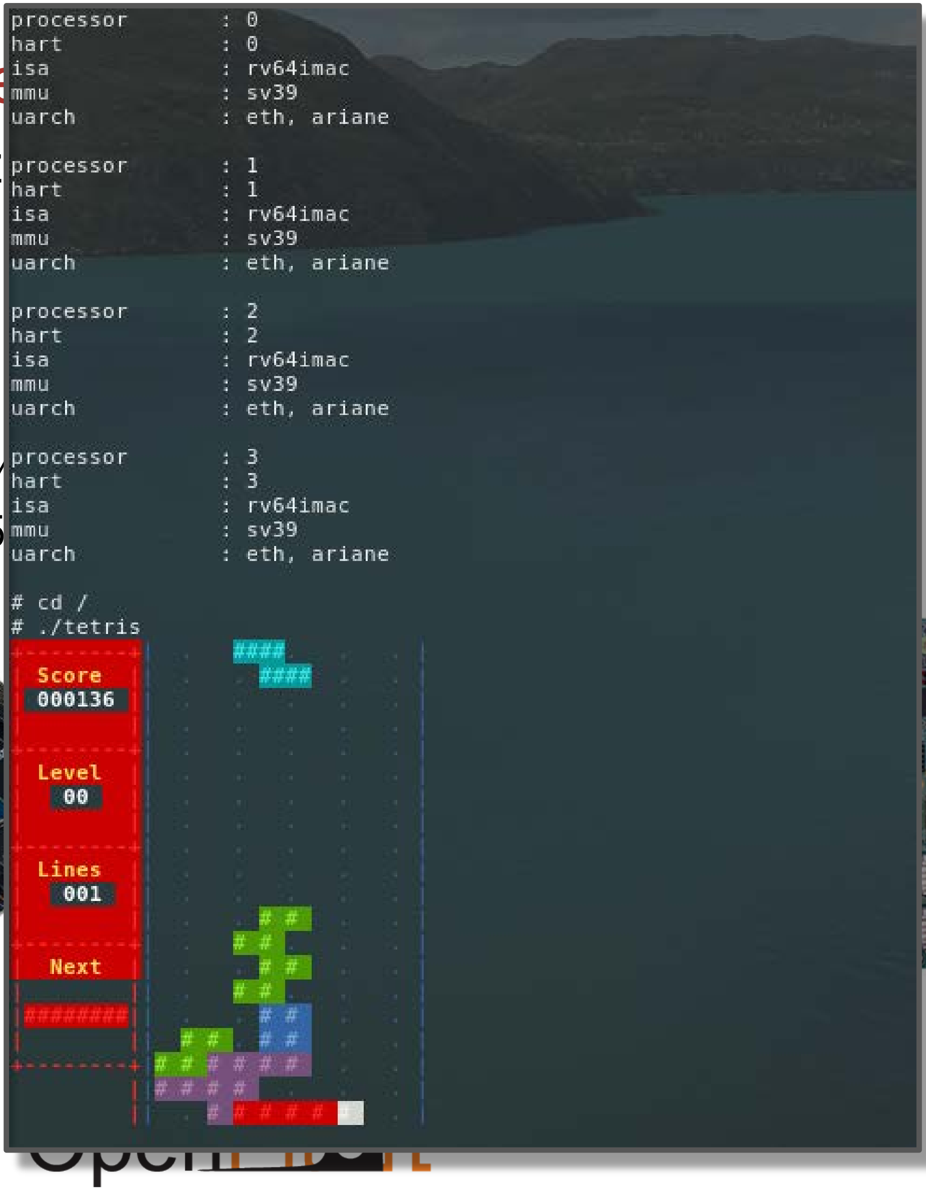
- OpenPiton
  - Developed by Princeton
  - Originally OpenSPARC T1
  - Scalable NoC with coherent LLC
  - Tiled Architecture
- Still work in progress
  - Bare-metal released in Dec ’18
  - Update with support for SMP Linux will be released soon



OpenPiton+Ariane mapped to FPGA

Digilent Gene

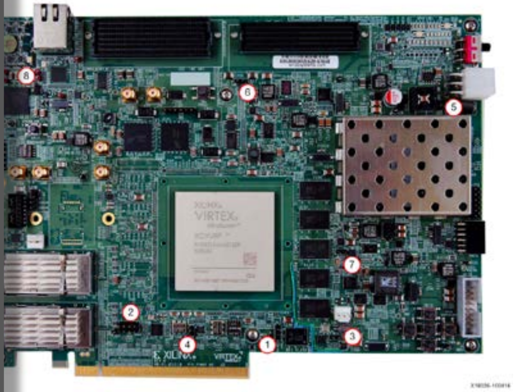
- Core: 66 MHz
- Up to 2 cores
- 8 GiB DDR3
- 1 core config:
  - 85k LUT (42%
  - 67 BRAM (15



```
processor       : 0
hart            : 0
isa             : rv64imac
mmu             : sv39
uarch           : eth, ariane

processor       : 1
hart            : 1
isa             : rv64imac
mmu             : sv39
uarch           : eth, ariane

processor       : 2
hart            : 2
isa             : rv64imac
mmu             : sv39
uarch           : eth, ariane

processor       : 3
hart            : 3
isa             : rv64imac
mmu             : sv39
uarch           : eth, ariane

# cd /
# ./tetris
  Score
  000136

  Level
   00

  Lines
   001

   Next
```

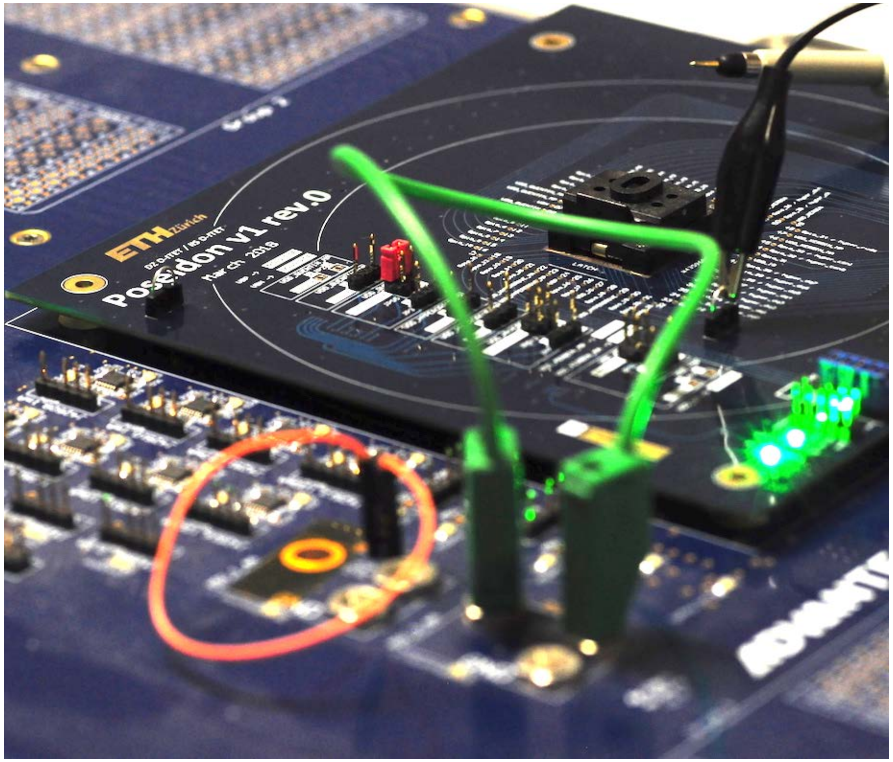
Xilinx VCU 118

- Core: 100 MHz
- Up to 16 cores
- 32 GiB DDR4
- (Available soon)



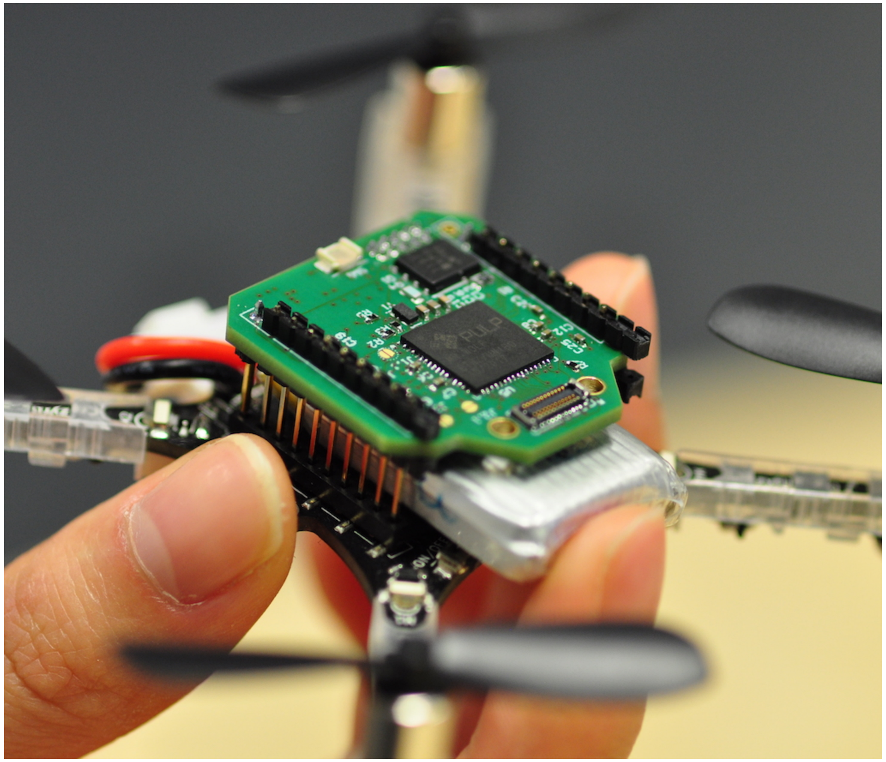
The Chips

We have designed more than 25 ASICs based on PULP



ASICs meant for applications

- More peripherals (SPI, Camera)
- More on-chip memory



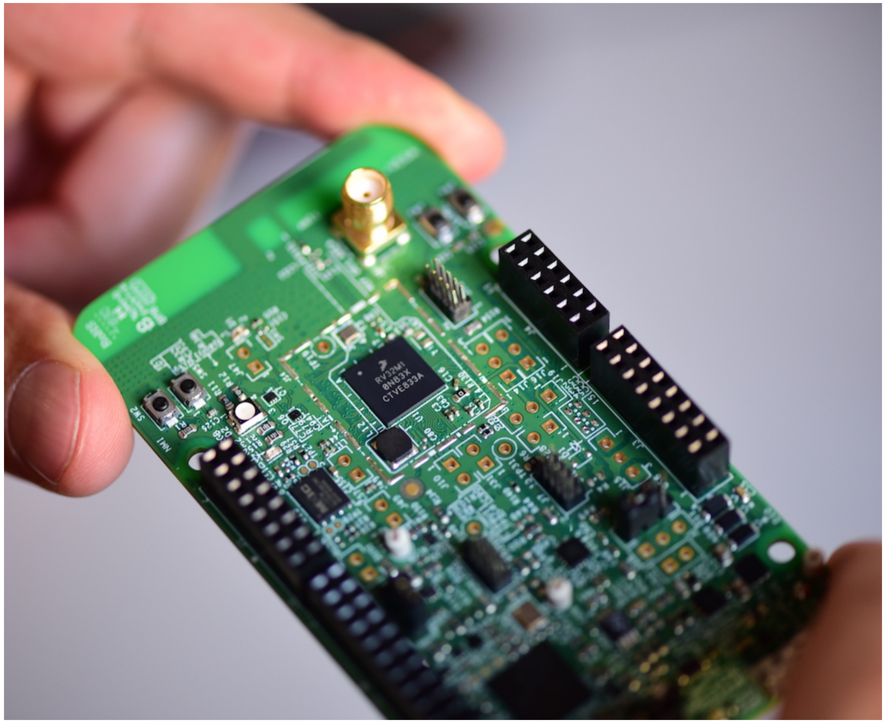
ASICs meant to go on IC Tester

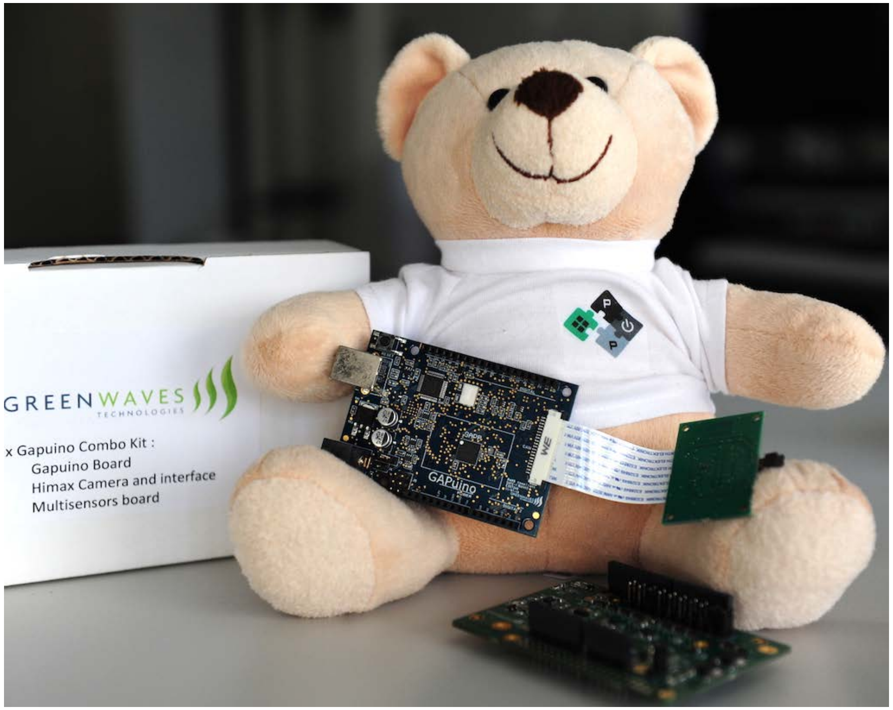
- Mainly characterization
- Not so many peripherals

You can buy development boards with PULP technology

VEGA board from open-isa.org

- Micro-controller board with RI5CY and zero-riscy

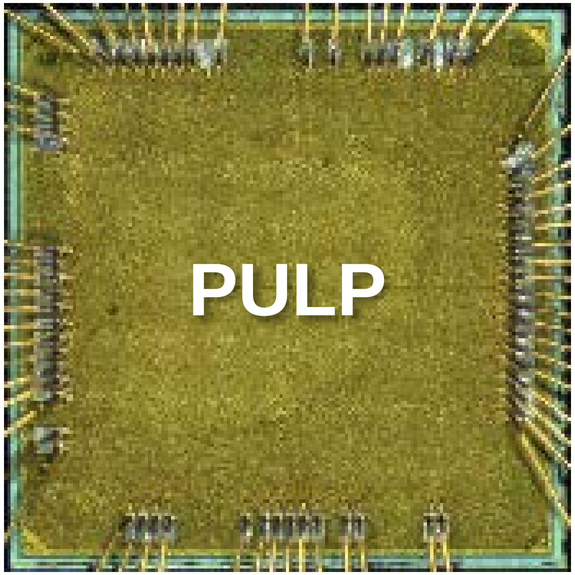


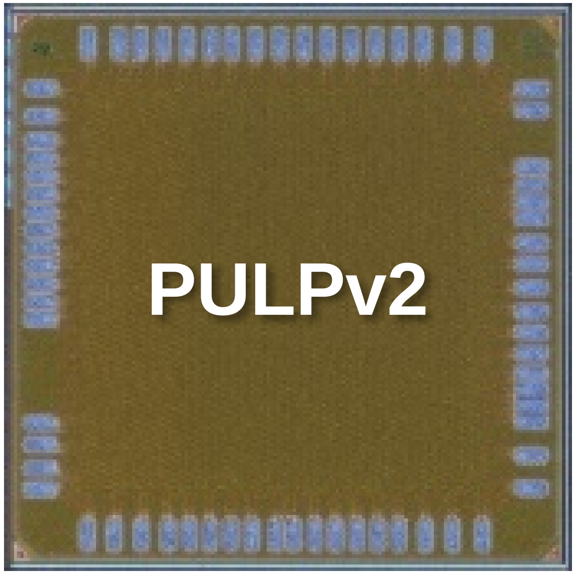


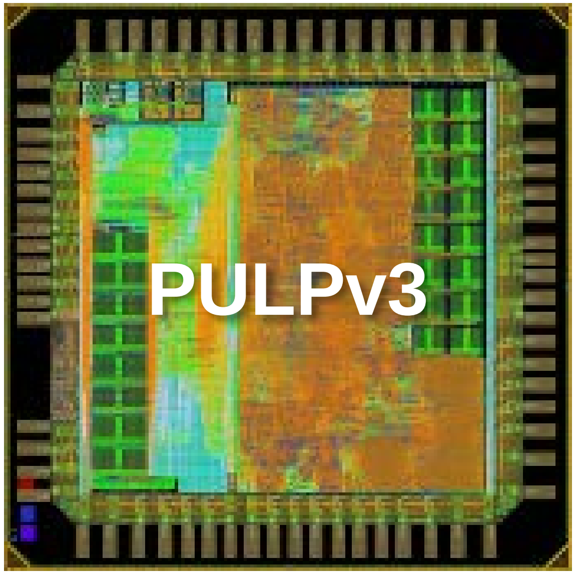
GAPUINO from Greenwaves

- PULP cluster system with 8+1 RI5CY cores

Brief illustrated history of selected ASICs

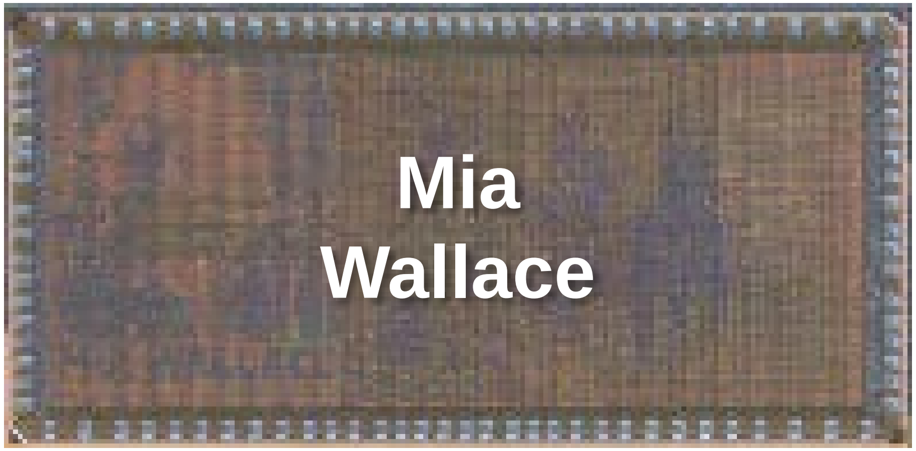






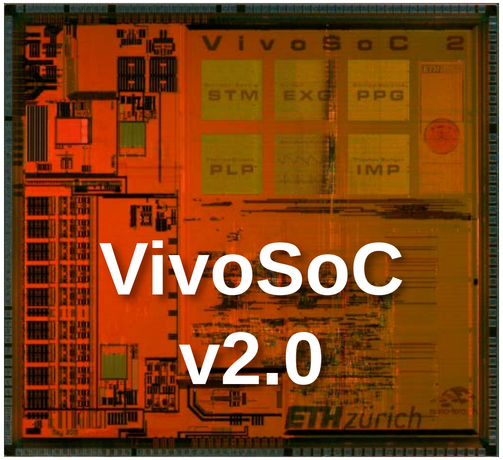
- All are 28 FDSOI technology, RVT, LVT and RVT flavor
- Uses OpenRISC cores
- Chips designed in collaboration with STM, EPFL, CEA/LETI
- PULPv3 has ABB control

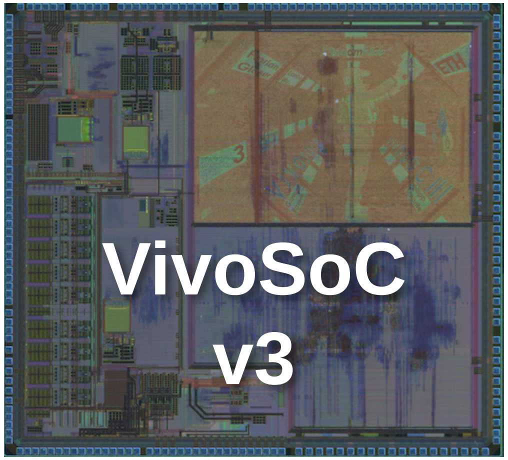
The first system chips, meant for applications



- First multi-core systems that were designed to work on development boards. Each have several peripherals (SPI, I2C, GPIO)
- Mia Wallace and Fulmine (UMC65) use OpenRISC cores
- Honey Bunny (GF28 SLP) uses RISC-V cores
- All chips also have our own FLL designs.

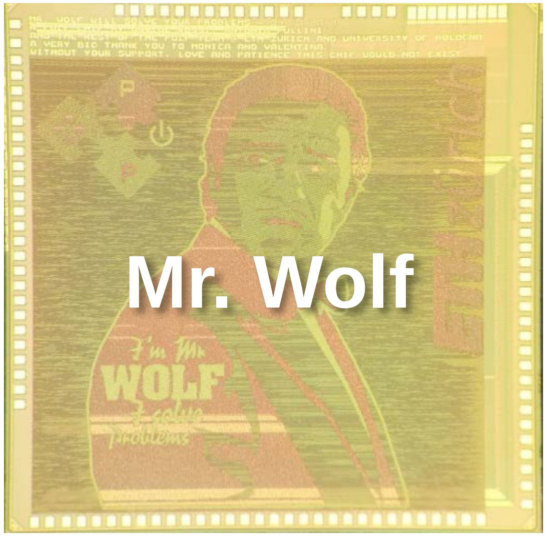
Combining PULP with analog front-end for Biomedical apps

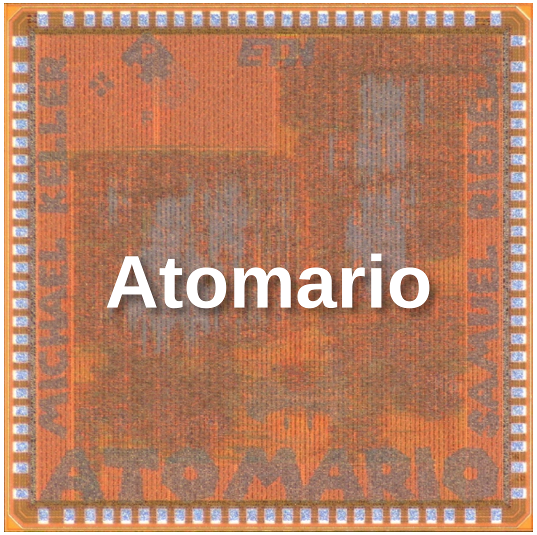


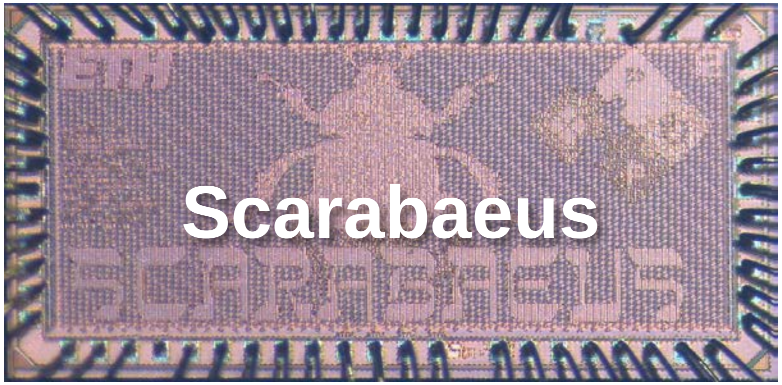


- Designed in collaboration with the Analog group of Prof. Huang at ETH
- All chips with SMIC130 (because of analog IPs)
- First three with OpenRISC, VivoSoC3 with RISC-V

The new generation chips from 2018

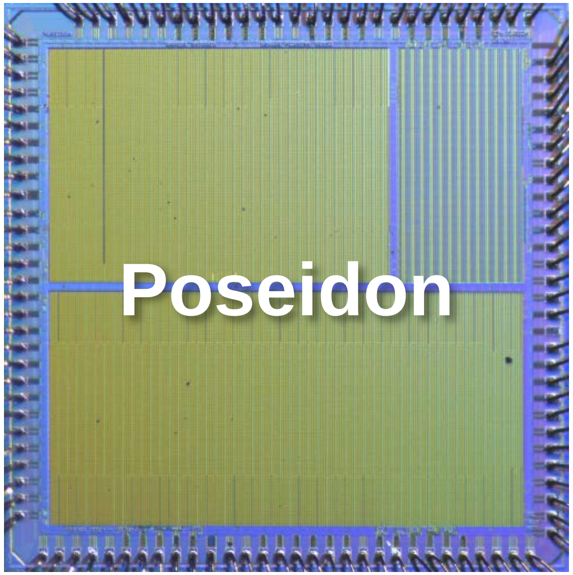


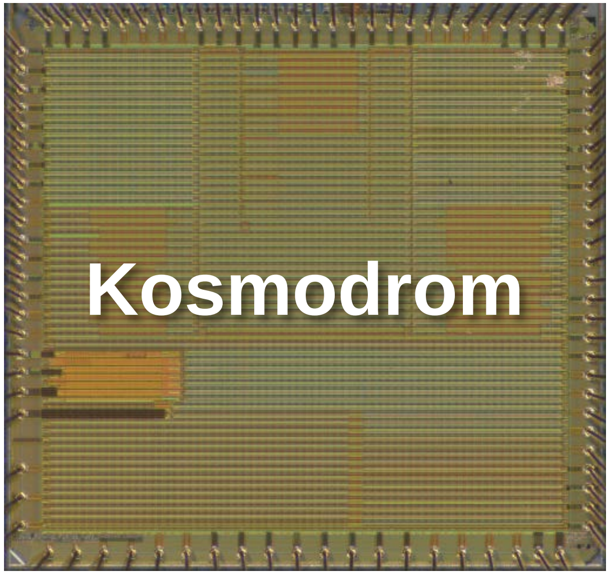


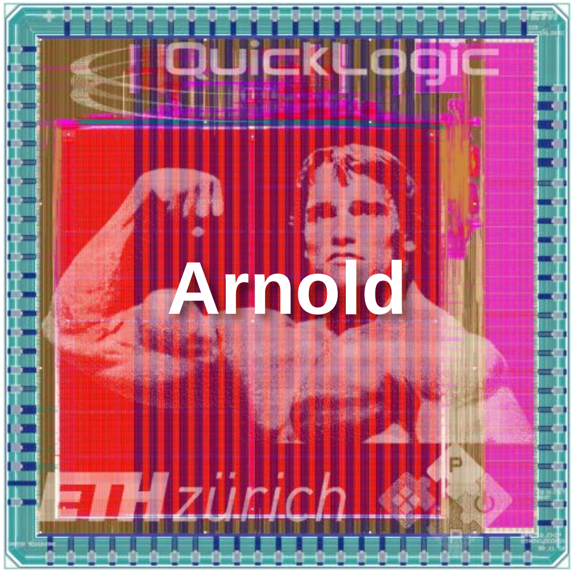


- System chips in TSMC40 (Mr. Wolf) and UMC65
- Mr. Wolf: IoT Processor with 9 RISC-V cores (Zero-riscy + 8x RI5CY)
- Atomario: Multi cluster PULP (2x clusters with 4x RI5CY cores each)
- Scarabaeus: Ariane based microcontroller

The large system chips from 2018

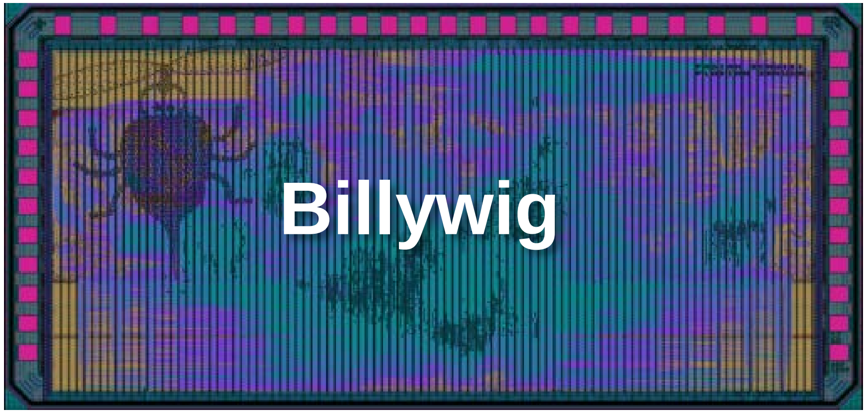


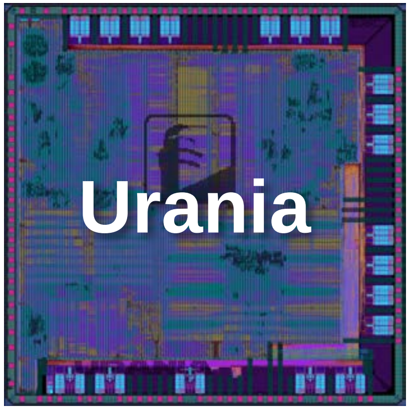


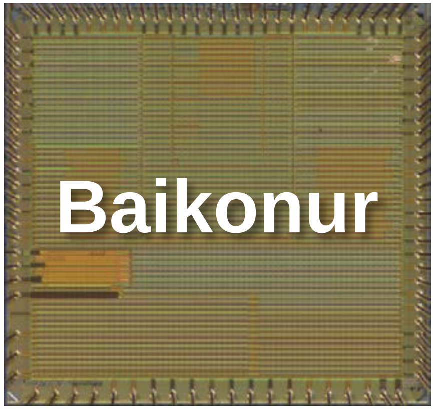


- All are Globalfoundries 22FDX, around 10 mm^2, 50-100 Mtrans
- Poseidon: PULPissimo (RI5CY) + Ariane
- Kosmodrom: 2x Ariane + NTX (FP streaming) accelerator
- Arnold: PULPissimo (RI5CY) + Quicklogic eFPGA

The next frontier from 2019





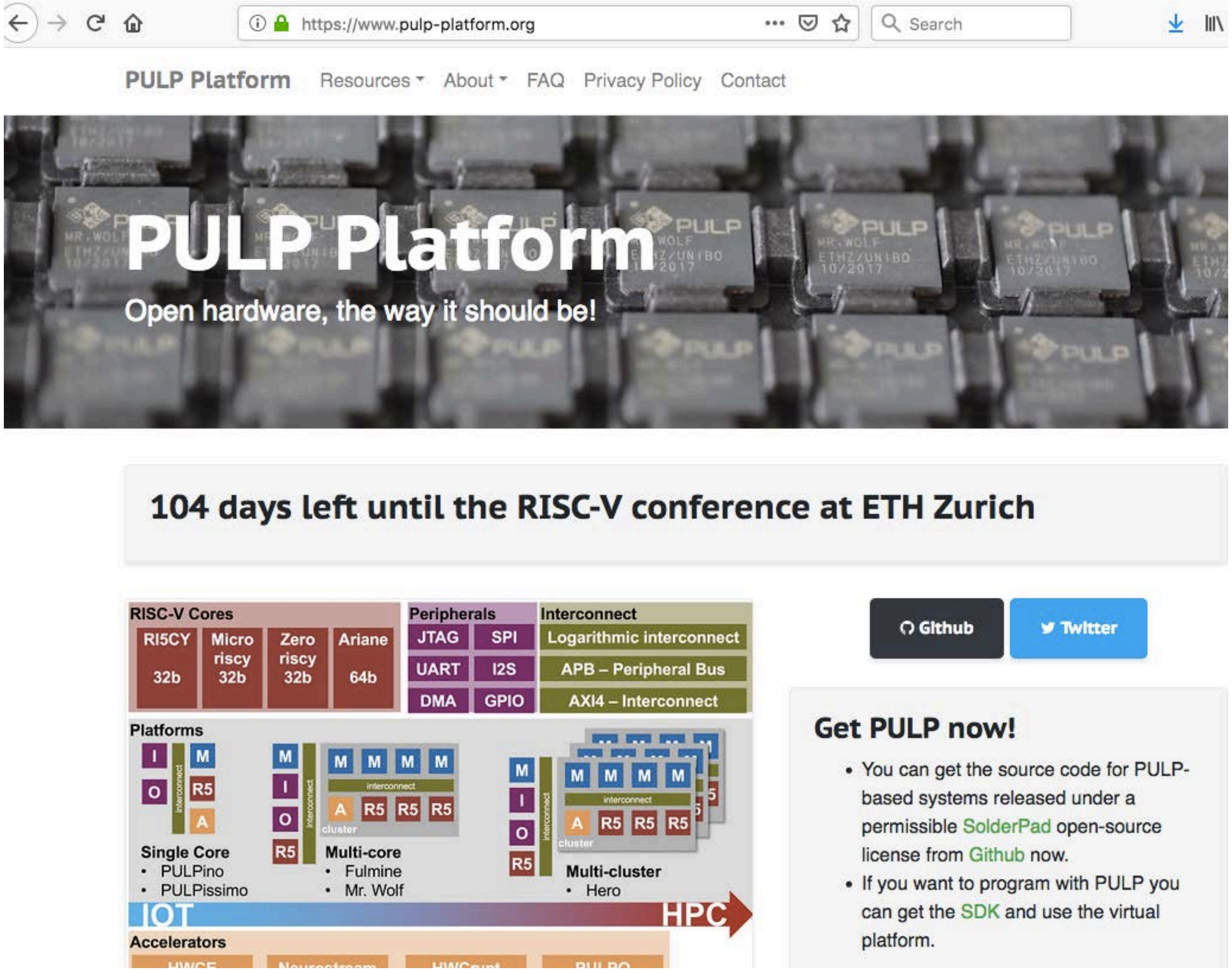


- UMC 65nm and Globalfoundries 22FDX
- **Billywig**: Streaming-enhanced RV32 cores for max. throughput, 3mm^2
- **Urania**: Ariane+PULP Het. SoC, plus custom DRAM controller, 16mm^2
- **Baikonur**: 2x Ariane + streaming-enhanced RV32 cores, 10mm^2

PULP

ETH

We firmly believe in Open Source movement



PULP Platform

Open hardware, the way it should be!

104 days left until the RISC-V conference at ETH Zurich

RISC-V Cores				Peripherals		Interconnect
RI5CY 32b	Micro riscy 32b	Zero riscy 32b	Ariane 64b	JTAG	SPI	Logarithmic interconnect
				UART	I2S	APB – Peripheral Bus
				DMA	GPIO	AXI4 – Interconnect

Platforms

Single Core
- PULPino
- PULPissimo

Multi-core
- Fulmine
- Mr. Wolf

Multi-cluster
- Hero

Accelerators

Get PULP now!

- You can get the source code for PULP-based systems released under a permissible SolderPad open-source license from Github now.
- If you want to program with PULP you can get the SDK and use the virtual platform.

First launched in February 2016 (Github)

All our development is on open repositories

Contributions from many groups

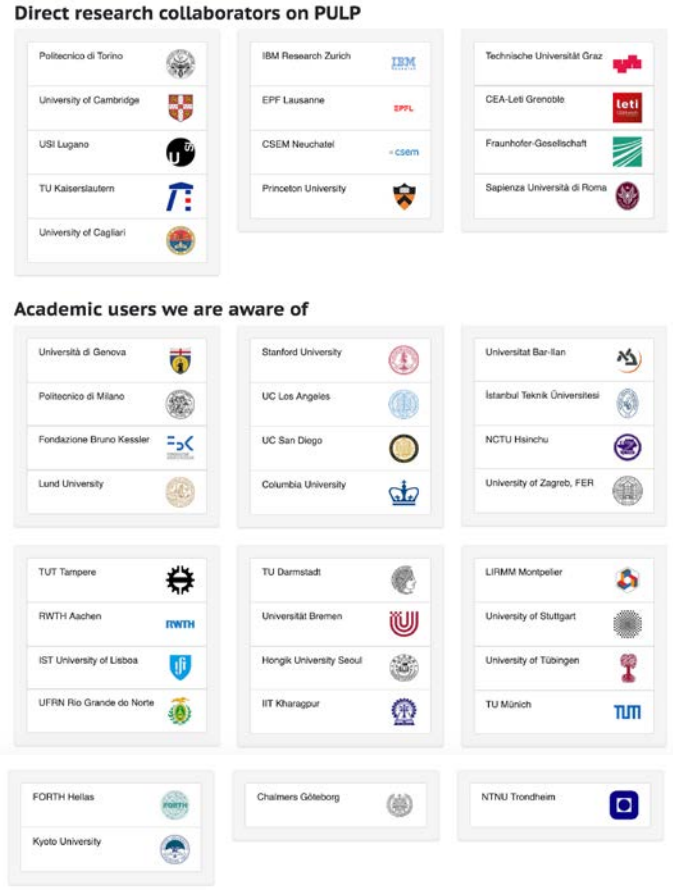
Open Hardware is a necessity, not an ideological crusade

- The way we design ICs has changed, big part is now infrastructure
  - Processors, peripherals, memory subsystems are now considered infrastructure
  - Very few (if any) groups design complete IC from scratch
  - High quality building blocks (IP) needed
- We need an easy and fast way to collaborate with people
  - Currently complicated agreements have to be made between all partners
  - In many cases, too difficult for academia and small enterprises
- Hardware is critical for security, we need to ensure it is secure
  - Being able to see what is really inside will improve security
  - Having a way to design open HW, will not prevent people from keeping secrets.

Silicon and Open Hardware fuel PULP success

- Many companies (we know of) are actively using PULP
  - They value that it is silicon proven
  - They like that it uses a permissive open source license

OPENHW GROUP — PROVEN PROCESSOR IP
lowRISC
Google
IBM
NXP
GREENWAVES TECHNOLOGIES
ST life.augmented
GLOBALFOUNDRIES
EMBECOSM
DOLPHIN INTEGRATION
CEVA
IQ-ANALOG
Mentor A Siemens Business
cadence
QuickLogic
onespin
imperas
SILICON LABS
axiomise predictable formal verification
ASHLING
ACP AG Advanced Circuit Pursuit
Valtrix SYSTEMS
antmicro
WITTENSTEIN



Direct research collaborators on PULP

Politecnico di Torino
University of Cambridge
USI Lugano
TU Kaiserslautern
University of Cagliari

IBM Research Zurich
EPF Lausanne
CSEM Neuchatel
Princeton University

Technische Universität Graz
CEA-Leti Grenoble
Fraunhofer-Gesellschaft
Sapienza Università di Roma

Academic users we are aware of

Università di Genova
Politecnico di Milano
Fondazione Bruno Kessler
Lund University

Stanford University
UC Los Angeles
UC San Diego
Columbia University

Universitat Bar-Ilan
İstanbul Teknik Üniversitesi
NCTU Hsinchu
University of Zagreb, FER

TUT Tampere
RWTH Aachen
IST University of Lisboa
UFRN Rio Grande do Norte

TU Darmstadt
Universität Bremen
Hongik University Seoul
IIT Kharagpur

LIRMM Montpelier
University of Stuttgart
University of Tübingen
TU Münich

FORTH Hellas
Kyoto University

Chalmers Göteborg

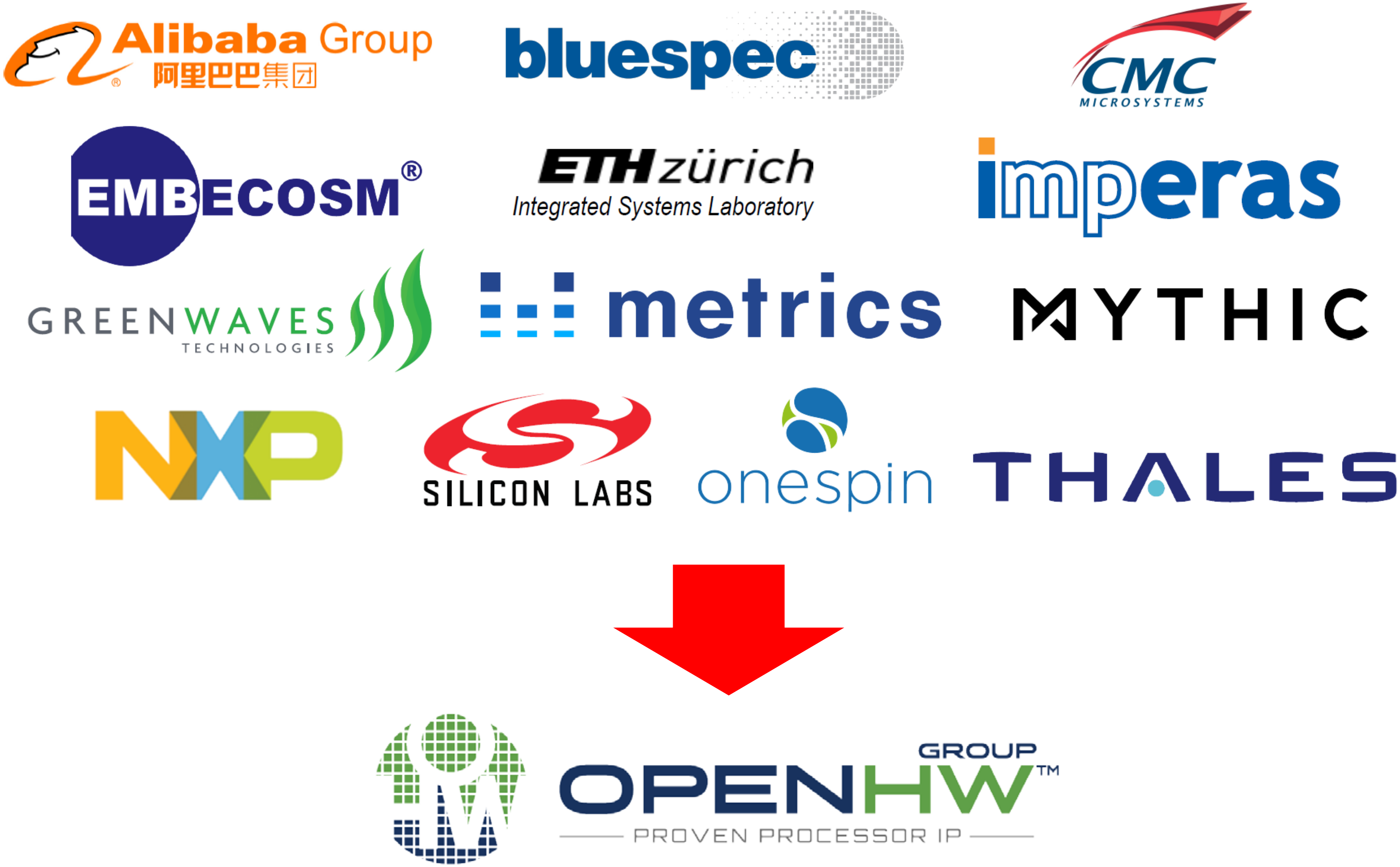
NTNU Trondheim

Micro/Zero-riscy is now Ibex

- LowRISC has agreed to maintain micro/zero riscy
  - Interested in using the core in their projects
  - They have a team that can provide support
  - ETH Zürich and University of Bologna will continue to contribute to Ibex
- Our core has grown and left the house
  - Alpine Ibex (Capra Ibex) is a mountain goat
    that is typical in the mountains of Switzerland

lowRISC

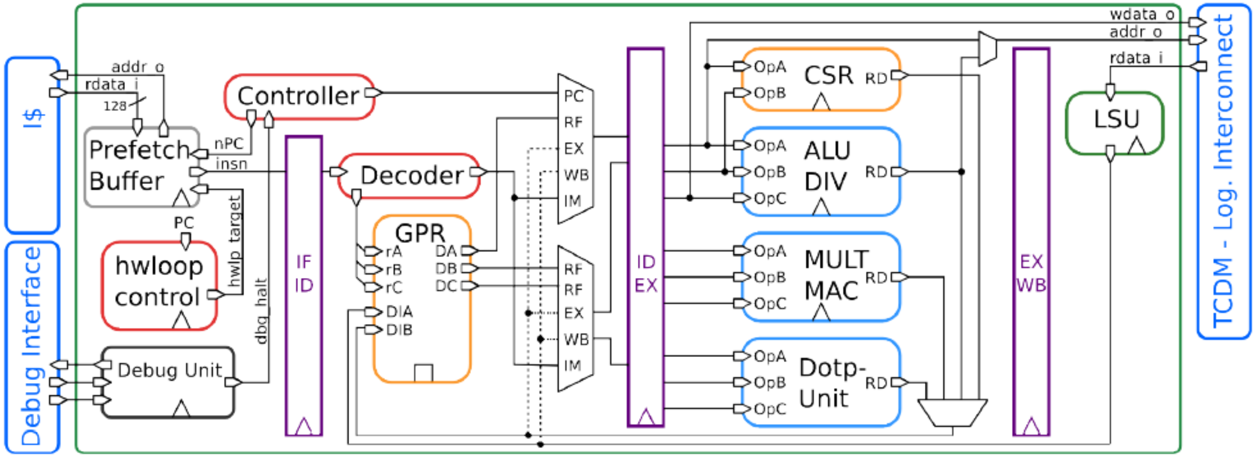
Non-Profit Open Hardware Group



OpenHW Group Charter

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

RI5CY core



CORE-V™

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

Thanks!

@pulp_platform pulp-platform.org asic.ethz.ch