ETH zürich

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Sub-pJ-Operation Scalable Computing
The PULP Experience

Davide Rossi[1],

Antonio Pullini[2], Igor Loi[1], Francesco Conti[1], Andreas Traber[2], Michael Gautschi[2], Andreas Traber[2], Frank K. Gürkaynak[2], Davide Schiavone[2], Daniele Palossi[2], Alessandro Capotondi[1], Giuseppe Tagliavini[1], Andrea Marongiu[1,2], Germain Haugou[2], Eric Flamand[2], Luca Benini[1,2]

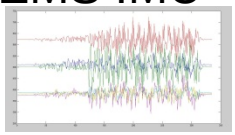
[1]DEI-UNIBO, [2]IIS-ETHZ

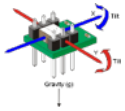
ETH zürich Computing for the Internet-of-Things

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Sense

MEMS IMU

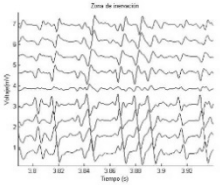




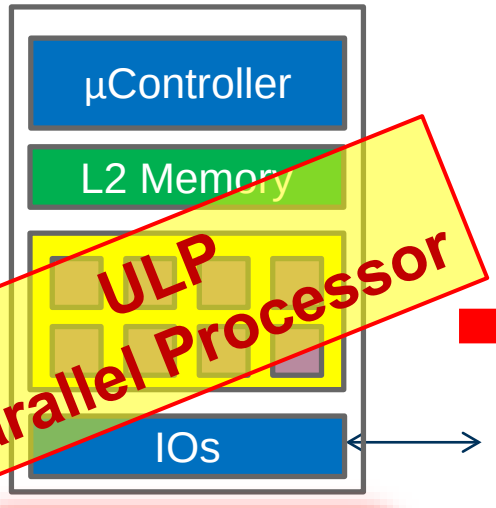
MEMS Microphone

ULP Imager

EMG/ECG/EIT



Analyze and Classify



1 ÷ 25 MOPS
1 ÷ 10 mW

Transmit

Short range, medium BW

Low rate (periodic) data

SW update, commands

Long range, low BW

LoRa SEMTECH

100 µW ÷ 2 mW

Battery + Harvesting powered
→ a few mW power envelope

Idle: ~1µW
Active: ~ 50mW

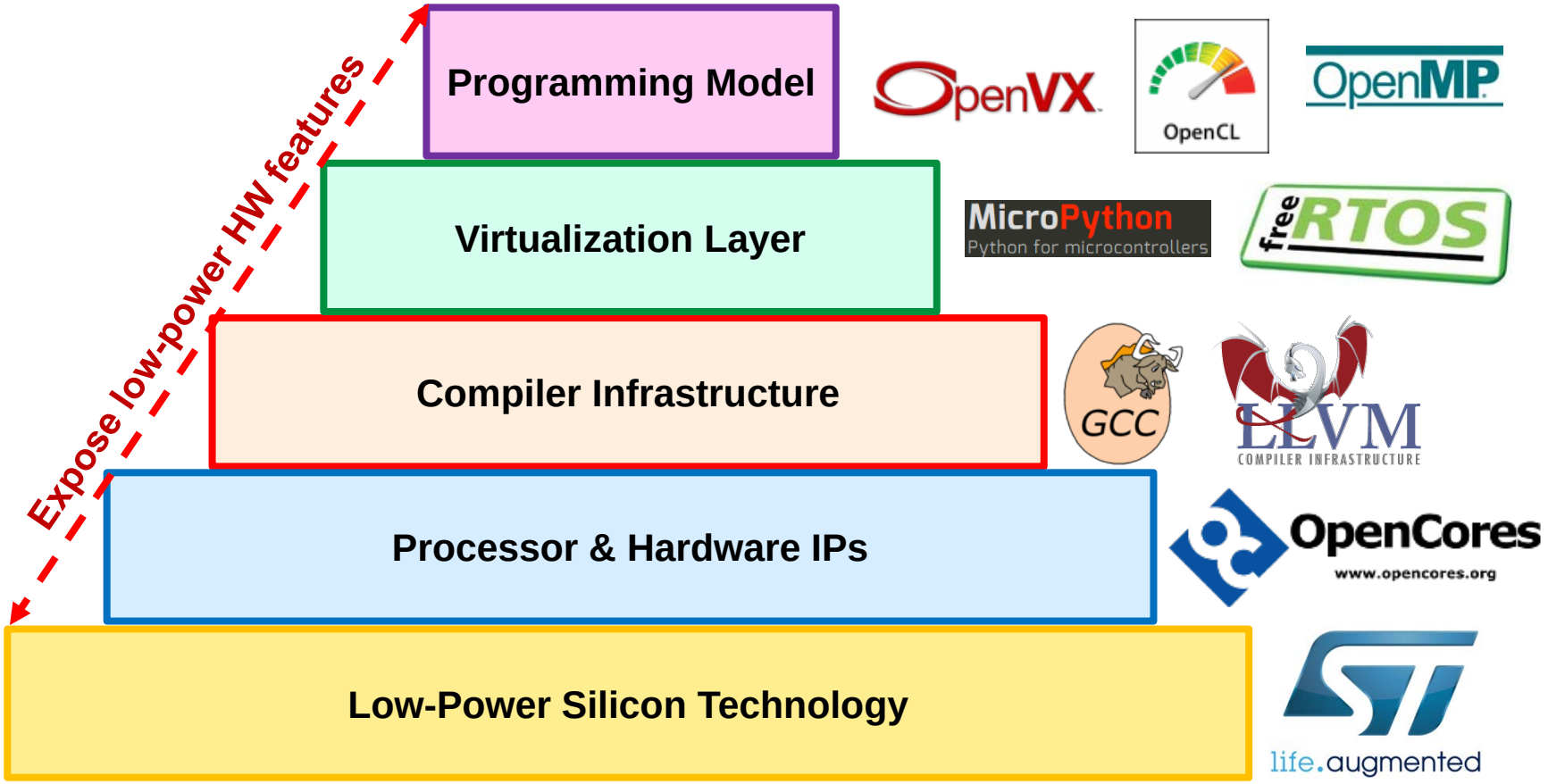
IoT: Near-Sensor Processing

	INPUT BANDWIDTH	COMPUTATIONAL DEMAND	OUTPUT BANDWIDTH	COMPRESSION FACTOR
Image				
Tracking: [*Lagroce2014]	80 Kbps	1.34 GOPS	0.16 Kbps	500x
Voice/Sound				
Speech: [*VoiceControl]	256 Kbps	100 MOPS	0.02 Kbps	12800x
Inertial				
Kalman: [*Nilsson2014]	2.4 Kbps	7.7 MOPS	0.02 Kbps	120x
Biometrics				
SVM: [*Benatti2014]	16 Kbps	150 MOPS	0.08 Kbps	200x

- ***Extremely compact output (single index, alarm, signature)***
- ***Computational power of ULP µControllers is not enough***
- ***Parallel worloads***

PULP:
pJ/op Parallel ULP computing

pJ/op is traditionally the target of ASIC + µControllers



Parallel + programmable + heterogeneous ULP computing

1mW-10mW active power

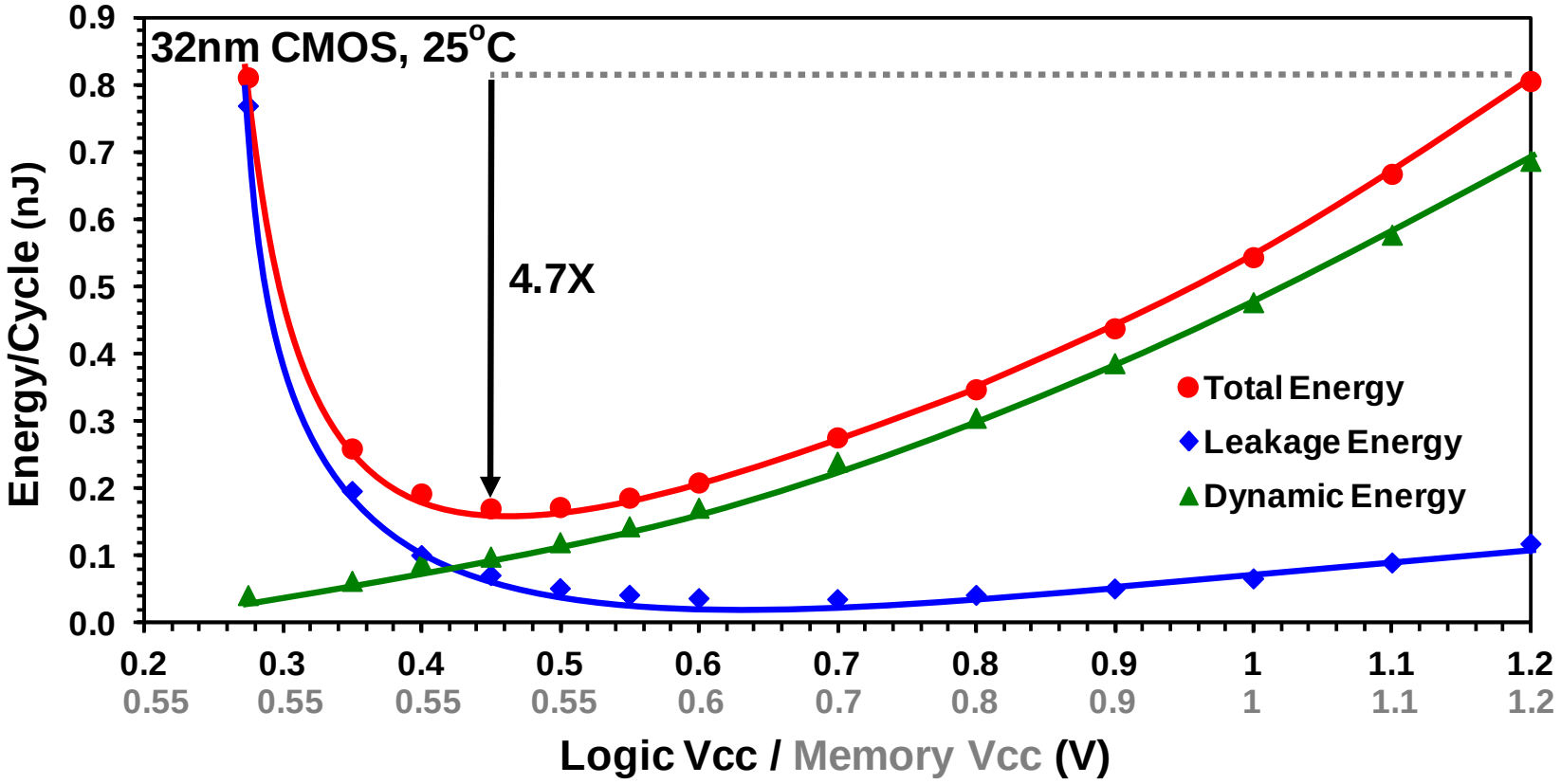
ETH zürich

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Near-Threshold Multiprocessing

Minimum energy operation

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**

Near-Threshold Multiprocessing

ETH zürich

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

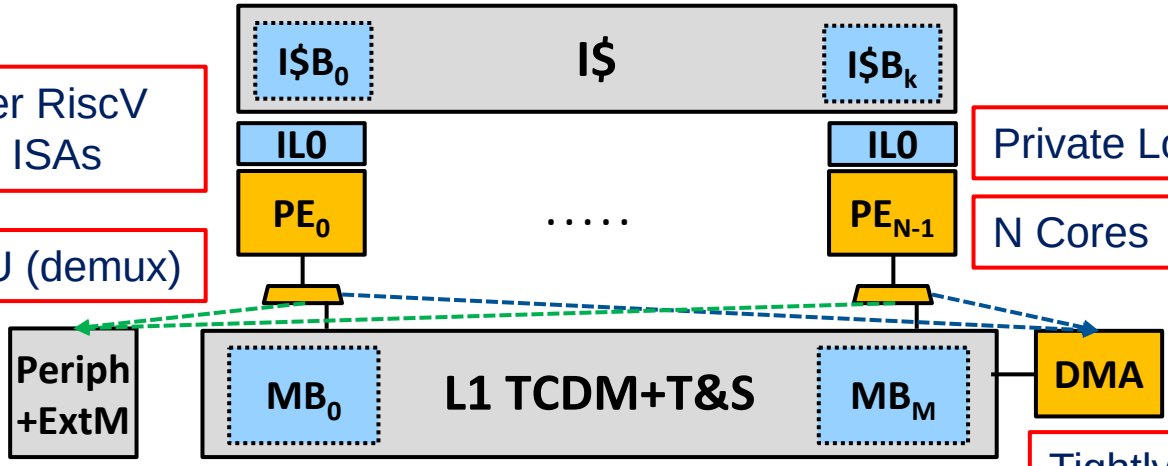
PULP
Parallel Ultra Low Power

Ultra-low power scalable computing

Shared L1 I$ with Multi-instruction load

4-stage, in-order RiscV and OpenRISC ISAs

Micro-MMU (demux)



Private Loop/Prefetch Buffer

N Cores

Tightly Coupled DMA

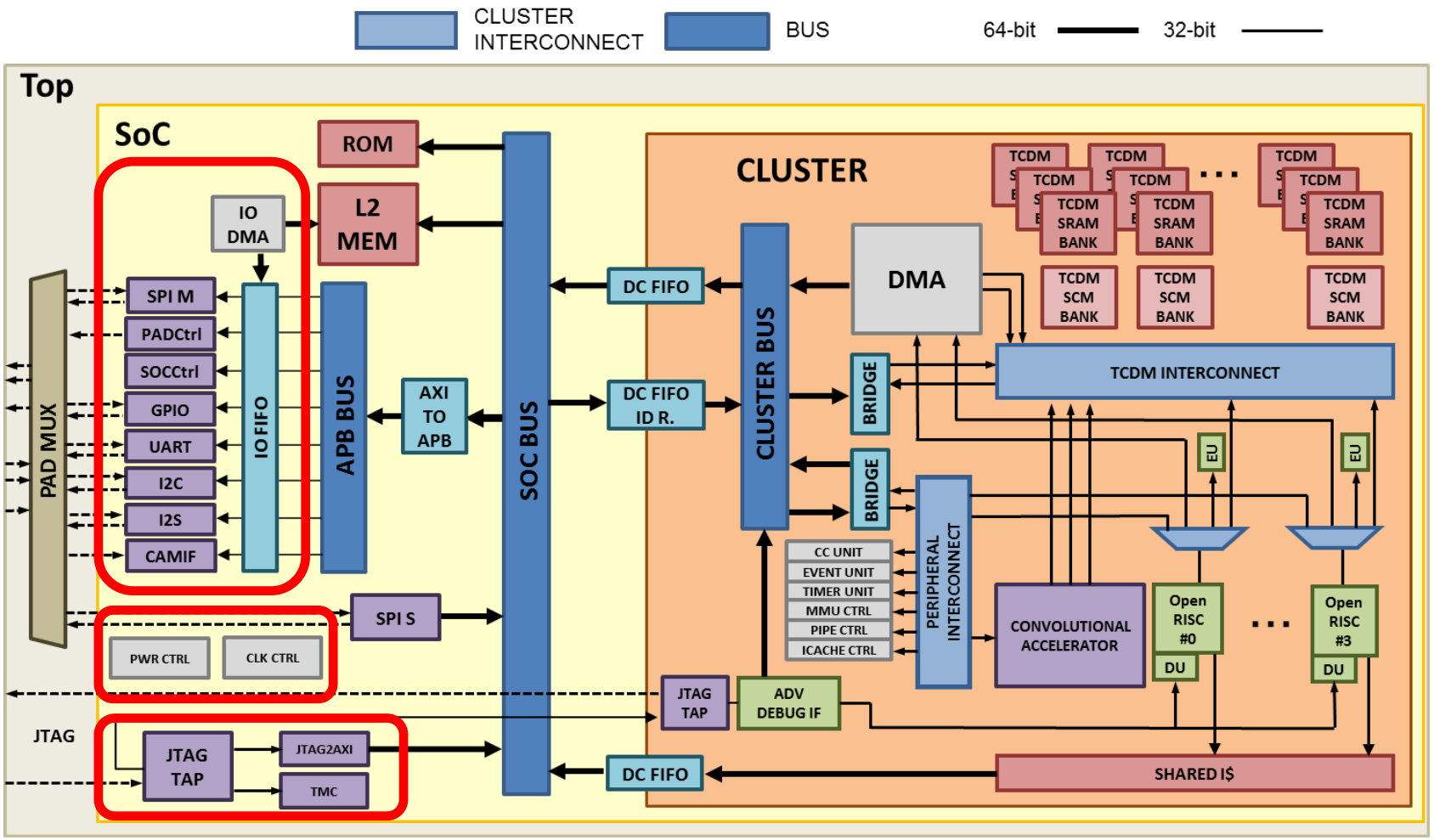
Shared L1 DataMem + Atomic Variables

1.. 8 PE-per-cluster, 1…32 clusters

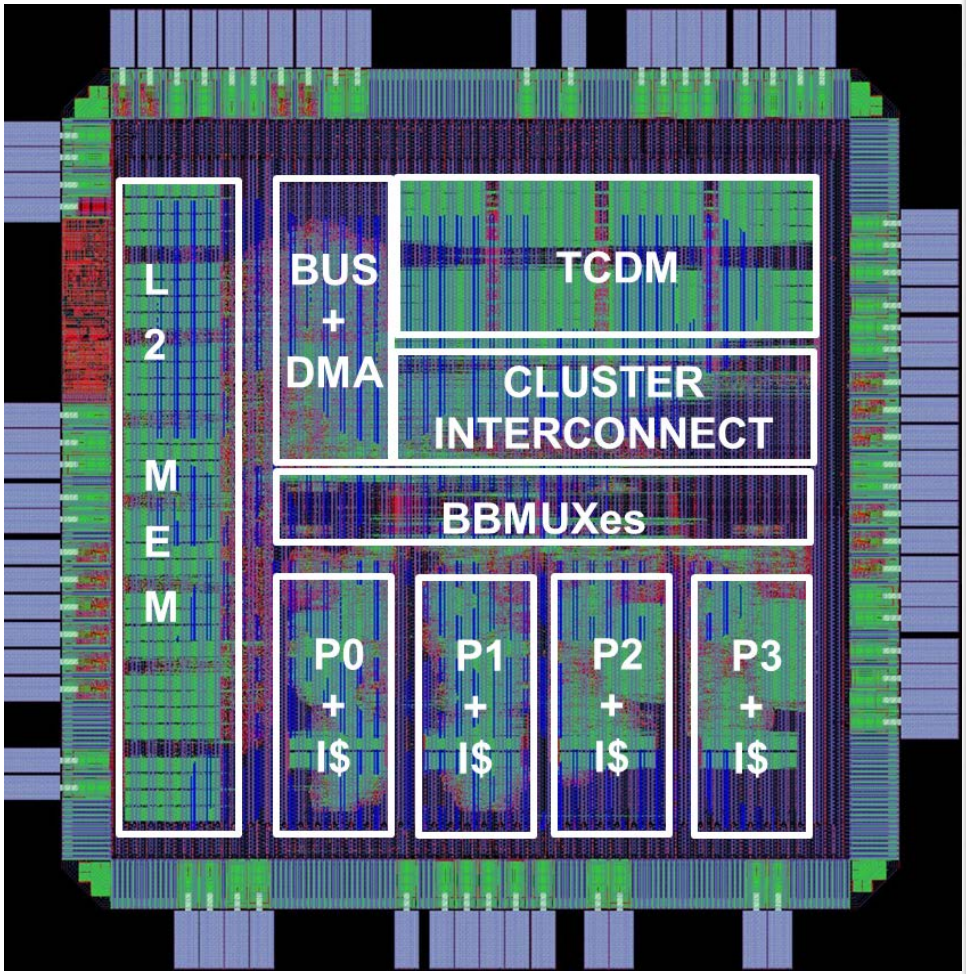
NT but parallel → Max. Energy efficiency when Active

+ strong PM for (partial) idleness

SoC Architecture



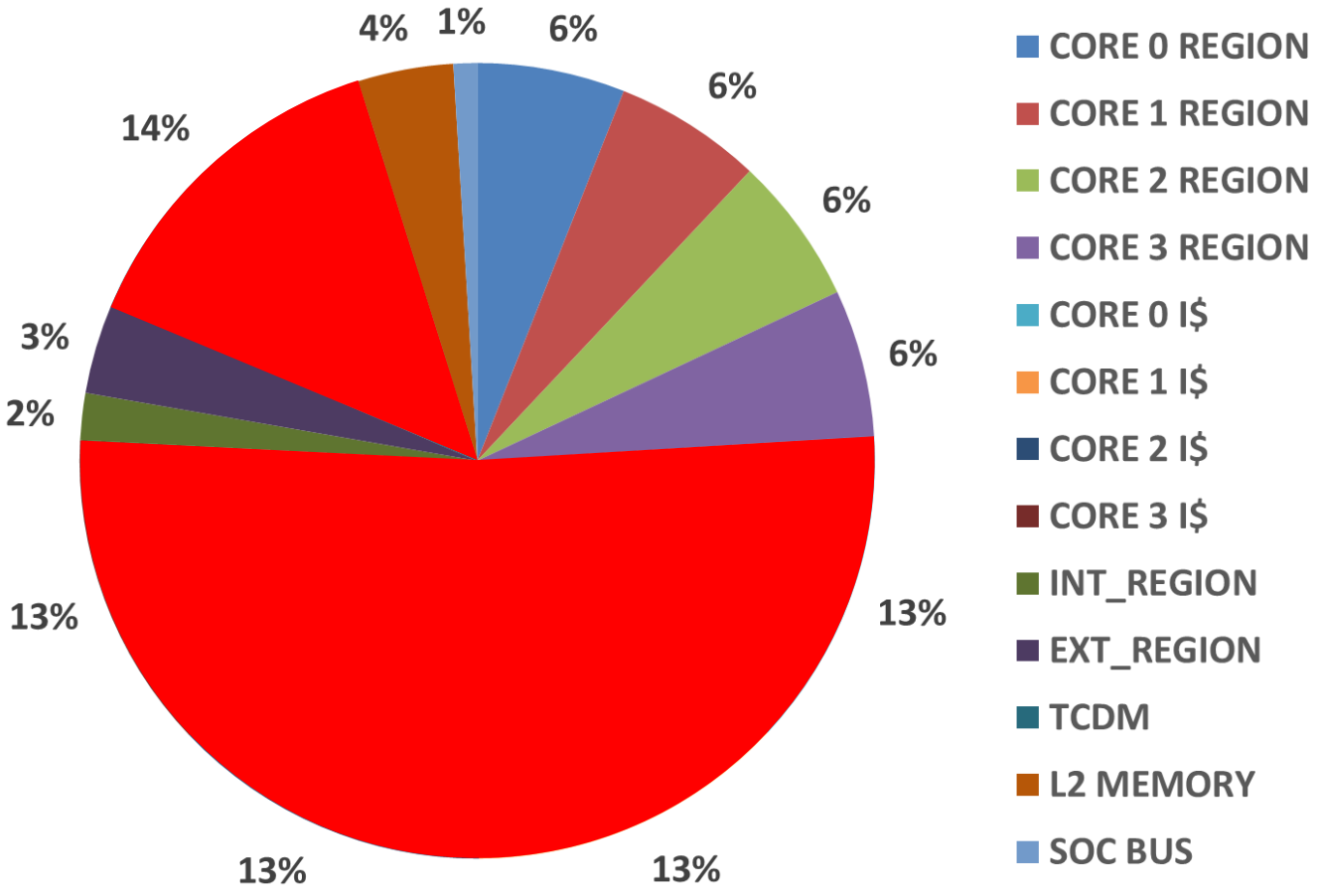
PULPv1



CHIP FEATURES	
Technology	28nm FDSOI (RVT)
Chip Area	3mm^2
# Cores	4xOpenRISC
I$	4x1kbyte (private)
TCDM	16 kbyte
L2	16 kbyte
BB regions	6
VDD range	0.45-1.2V
VBB range	-1.8V - +0.9V
Perf. Range	1 MOPS-1.9GOPS
Power Range	100 µW -127 mW
Peak Efficiency	60 GOPS/W@0.5V*

ISSCC15 (student presentations), Hot Chips 15, ISSCC16 (paper+student presentation)

Mmory is the bottleneck

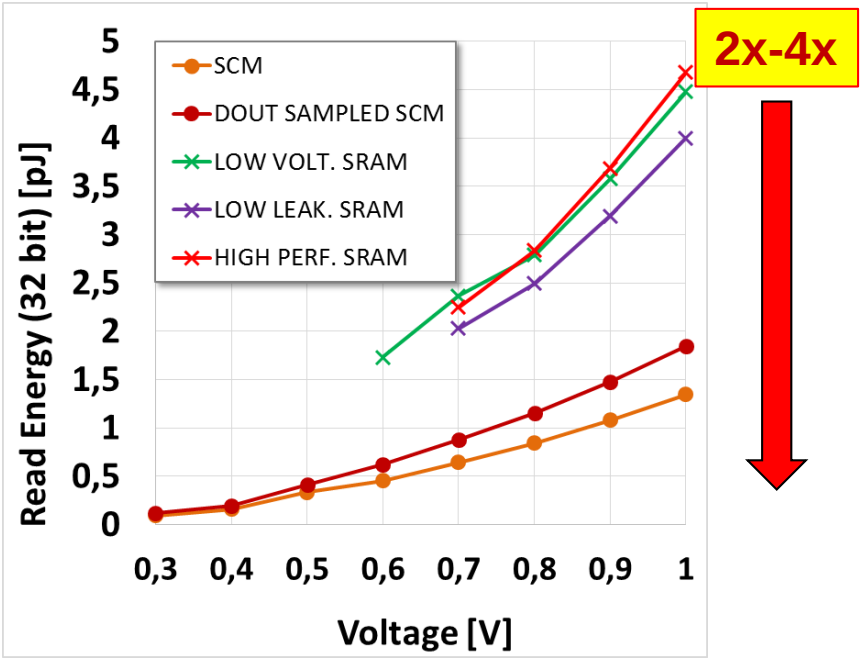


I$s + TCDM consume > 60% of total power!!!

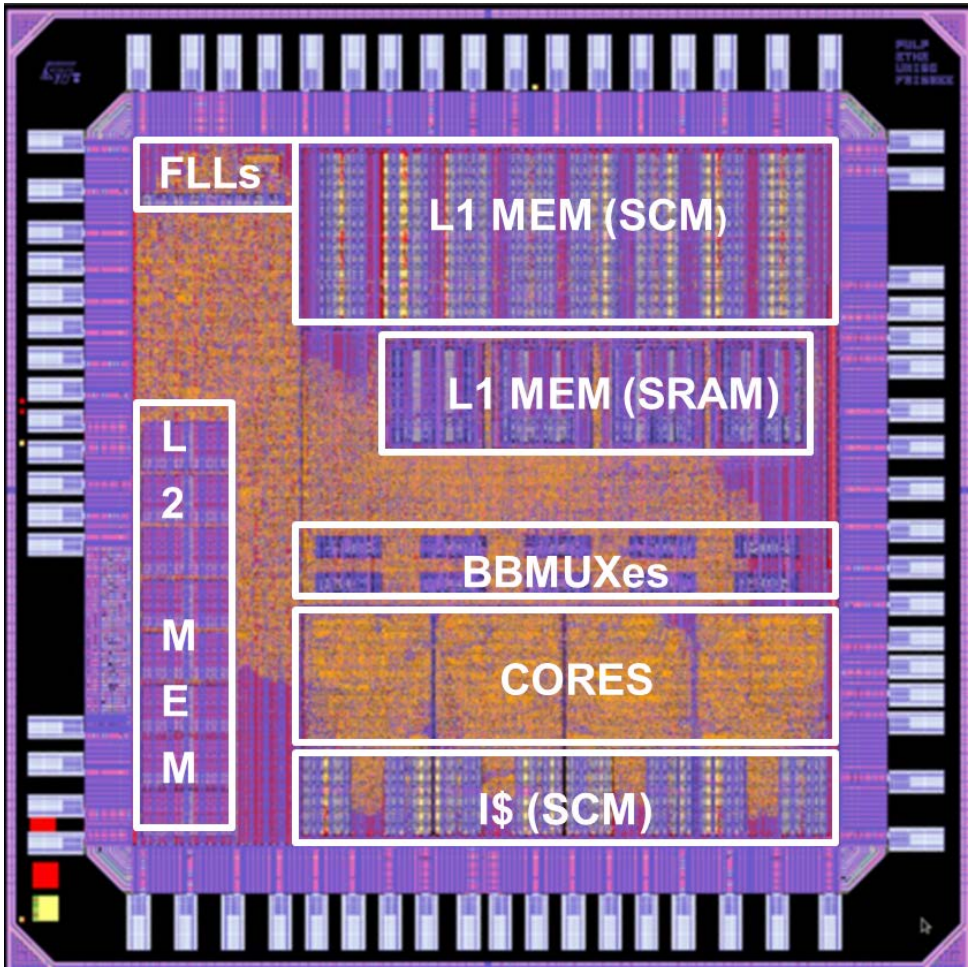
ULP (NT) Bottleneck: Memory

- "Standard" 6T SRAMs:
  - High VDDMIN
  - Bottleneck for energy efficiency
- Near-Threshold SRAMs (8T)
  - Lower VDDMIN
  - Area/timing overhead (25%-50%)
  - High active energy
  - Low technology portability
- Standard Cell Memories:
  - Wide supply voltage range
  - Lower read/write energy (2x - 4x)
  - Easy technology portability
  - Major area overhead (2x)

256x32 6T SRAMS vs. SCM



PULPv2



CHIP FEATURES	
Technology	28nm FDSOI (LVT)
Chip Area	3mm^2
# Cores	4xOpenRISC
I$ (SCM)	4x1kbyte (private)
TCDM	32 + 8 Kbyte
L2	64 kbyte
BB regions	10
VDD range	0.3-1.2V (0.45-1.2V)*
VBB range	0V-2V
Perf. Range	1 MOPS - 3.3 GOPS
Power Range	10µW - 480 mW
Peak Efficiency	193 GOPS/W@0.46V

Taped out Nov. 2014!

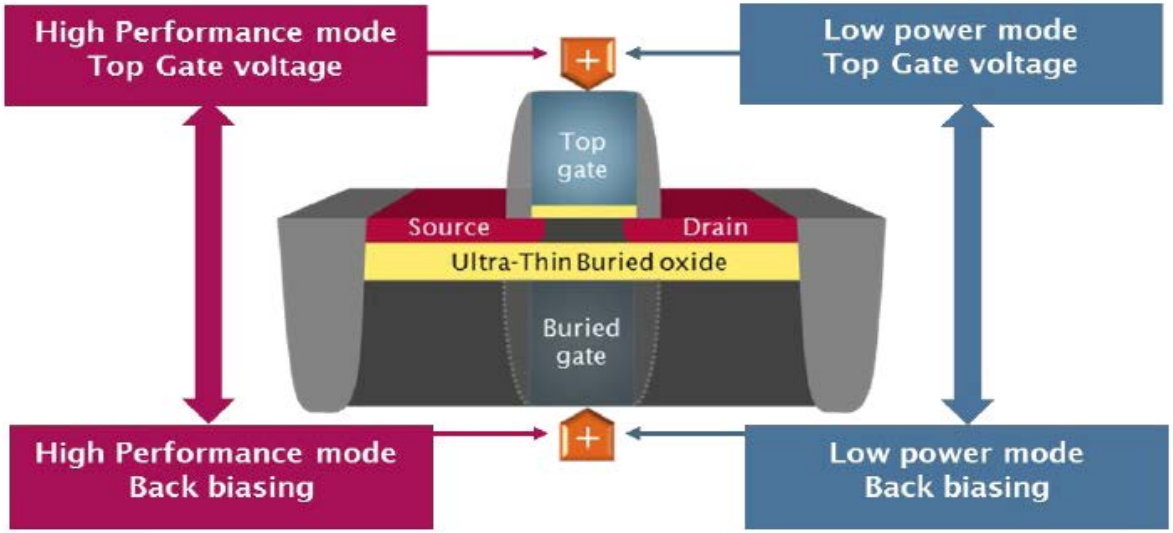
ETH zürich

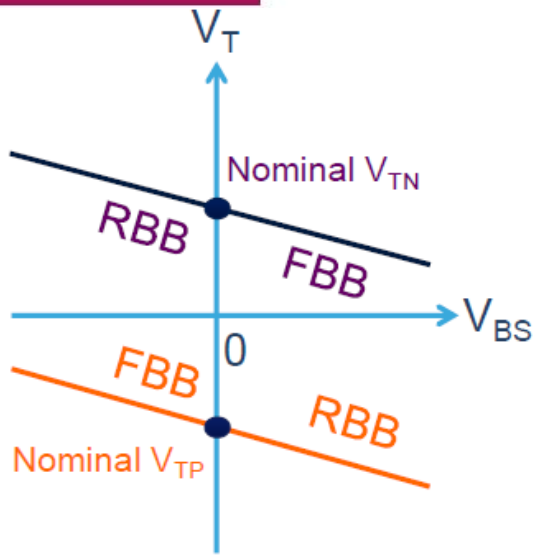
PULPv2

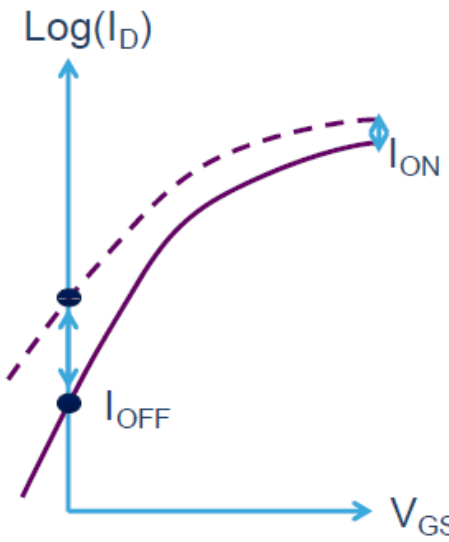
ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

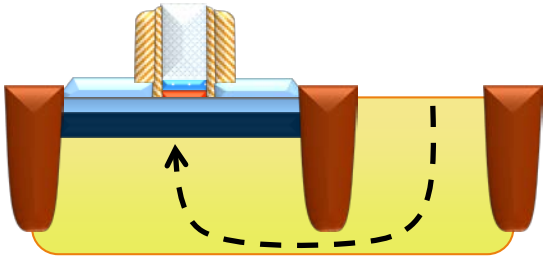
	[2]	[3]	[4]	[5]	This Work
Technology	CMOS 32nm	CMOS 28nm LP	FD-SOI 28nm flip-well	FD-SOI 28nm conventional-well	FD-SOI 28nm flip-well
Data format	2x 32-bit superscalar	4x 32-bit VLIW	32-bit	32-bit	32-bit
# of cores	1	1	1	4	4
I$/D$/L2	8K/8K/n.a.	16K/32K/256K	4K/4K/n.a.	1Kx4/16K/16K	1Kx4/48K/64K
Voltage range (SRAMs)	0.28V – 1.0V (0.5V – 1.0V)	0.6V - 1.05V	0.4V - 1.3V	0.44V – 1.2V (0.54V – 1.2V)	0.32V – 1.15V (0.45V – 1.15V)
Max frequency	915 MHz	1.2 GHz	2.6 GHz	475 MHz	825 MHz
Best power density	170 µW/MHz	58 µW/MHz	62 µW/MHz	65 µW/MHz	**20.7 µW/MHz**
Best performance	1.8 GOPS	3 GOPS	2.6 GOPS	1.8 GOPS	**3.3 GOPS**
Peak energy efficiency (MAX)	11.7 MOPS/mW @ 50 MOPS	43.1 MOPS/mW @ 230 MOPS	16.1 MOPS/mW @ 460 MOPS	60 MOPS/mW @ 25.6 MOPS	**193 MOPS/mW @ 162 MOPS**

Near threshold FDSOI technology



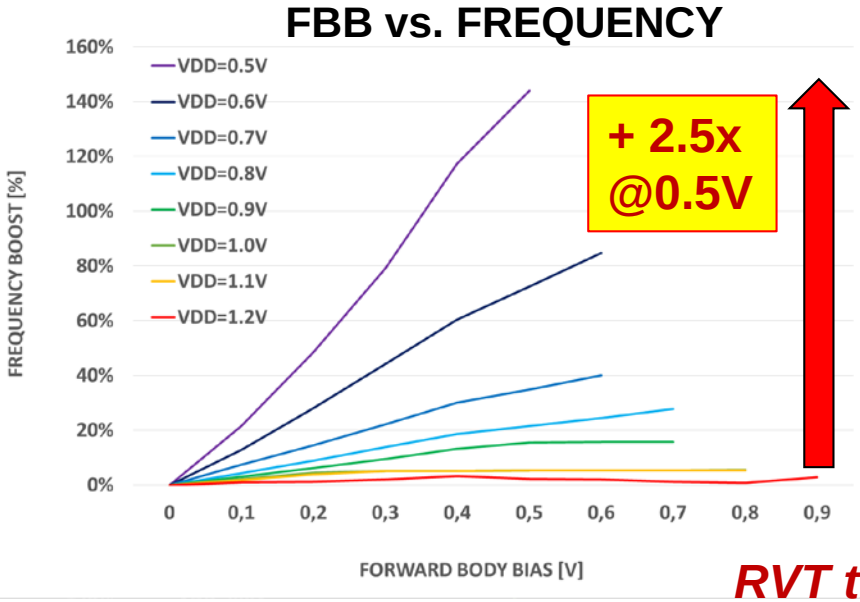


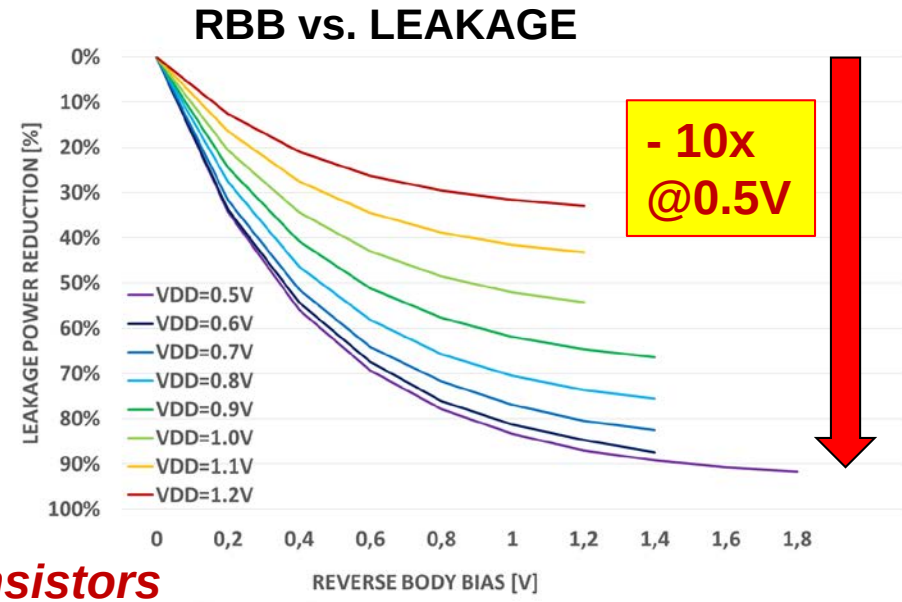




Body bias: Highly effective knob for power & variability management!

Near Threshold + Body Biasing Combined





RVT transistors

- State retentive (no state retentive registers and memories)
- Ultra-fast transitions (tens of ns depending on n-well area to bias)
- Low area overhead for isolation (3µm spacing for deep n-well isolation)
- Thin grids for voltage distribution (small transient current for wells polarization)
- Simple circuits for on-chip VBB generation (e.g. charge pump)

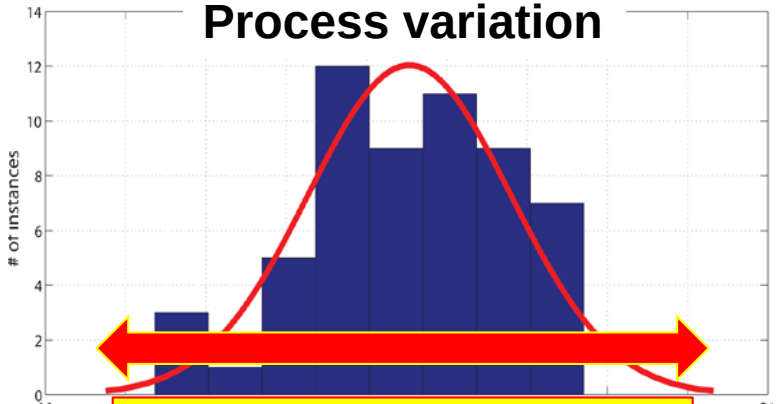
But even with aggressive RBB leakage is not zero!

Body Biasing for Variability Management

ETH zürich

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

Process variation



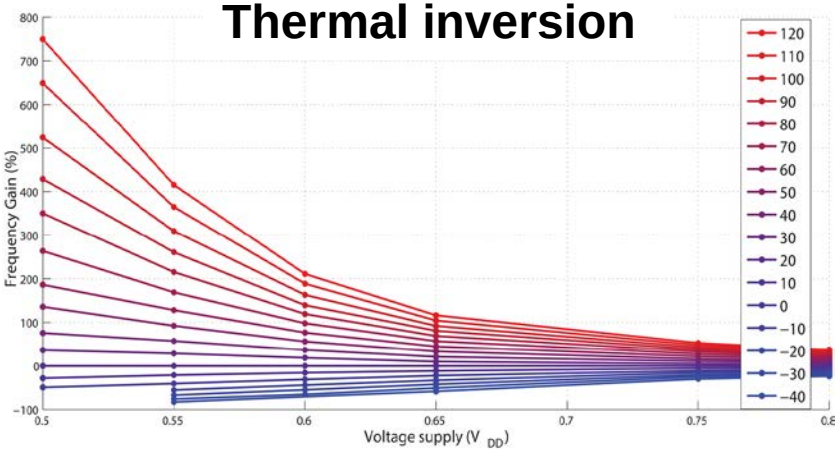
25 MHz ± 7 MHz (3σ)

120° C

100x @0.5V

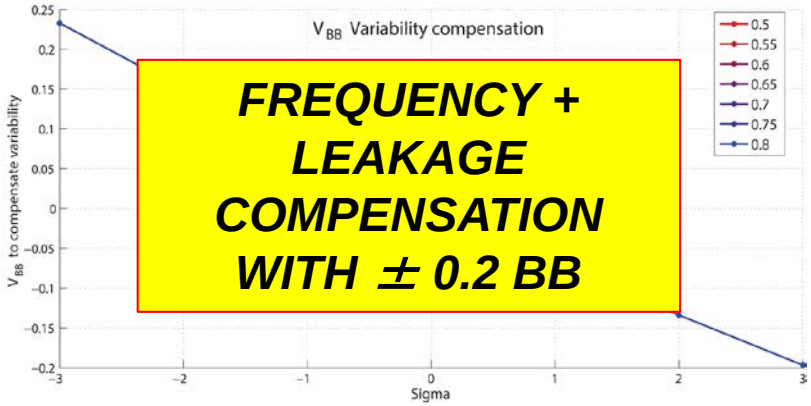
-40° C

Thermal inversion

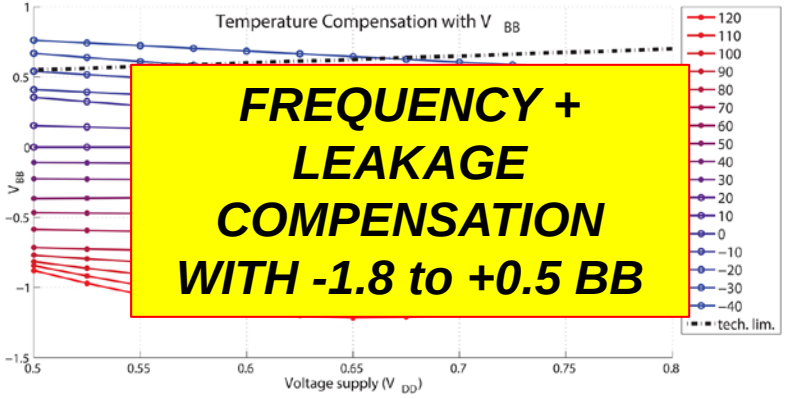


RVT transistors

FBB/RBB



FREQUENCY + LEAKAGE COMPENSATION WITH ± 0.2 BB



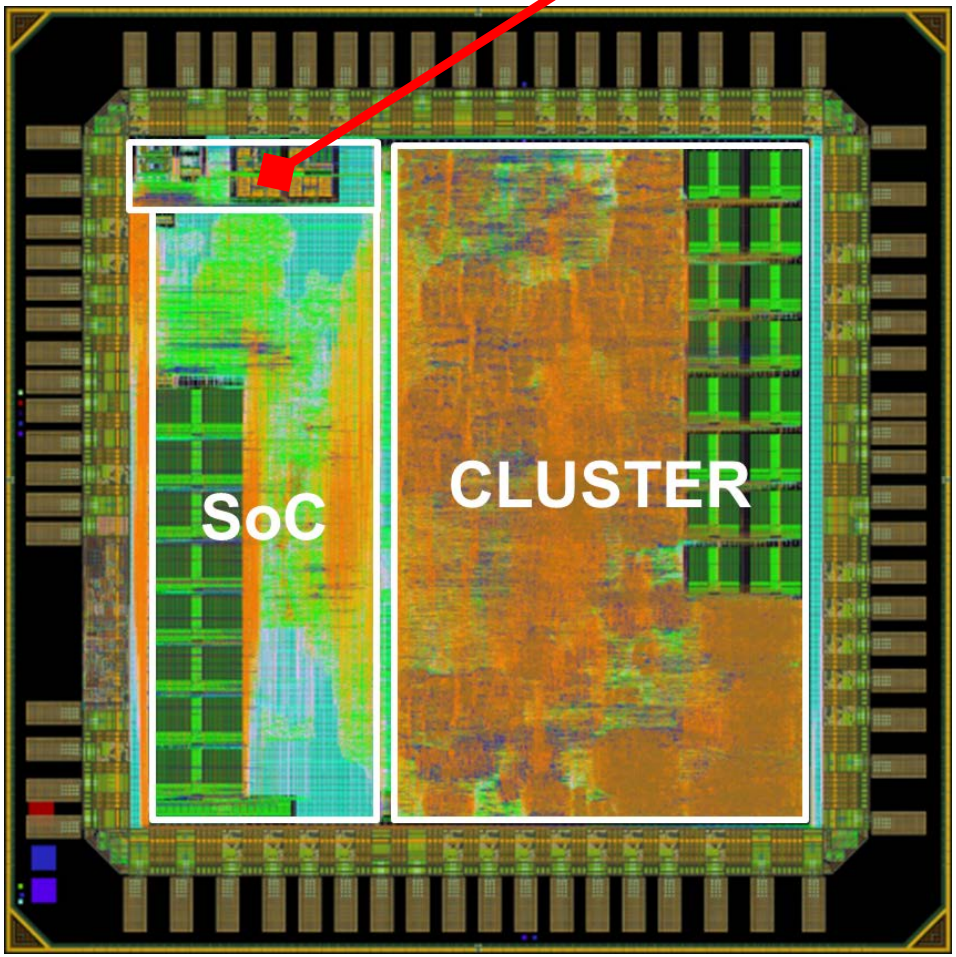
FREQUENCY + LEAKAGE COMPENSATION WITH -1.8 to +0.5 BB

ETH zürich

PULPv3

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

FLLs + BBGEN + PMBs + Multiprobes



CHIP FEATURES	
Technology	28nm FDSOI (RVT)
Chip Area	3mm^2
# Cores	4xOR1ON
I$	4 kbyte (shared)
TCDM	64 + 8 kbyte
L2	128 kbyte
BB regions	2 (SoC, Cluster)
VDD range	0.4-0.7V
VBB range	-1.8V - +0.9V
Perf. Range	1 MOPS - 1.8 GOPS*
Power Range	20µW - 5.6 mW*
Peak Efficiency	387 GOPS/W@0.5V*

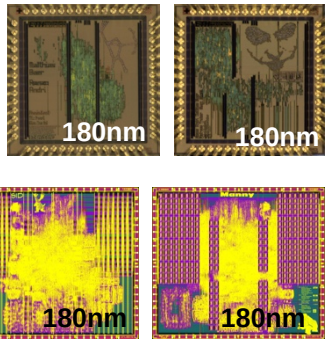
under test NOW!

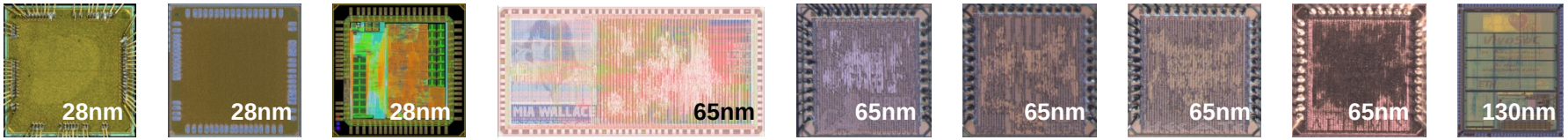
****Cluster, Estimated***

Building PULP: silicon roadmap

- **Main PULP chips** (ST 28nm FDSOI)
  - PULPv1
  - PULPv2
  - PULPv3 (under test)
  - PULPv4 (in progress)
- **PULP development** (UMC 65nm)
  - Artemis - IEEE 754 FPU
  - Hecate - Shared FPU
  - Selene - Logarithic Number System FPU
  - Diana - Approximate FPU
  - Mia Wallace – full system
  - *Imperio* - PULPino chip
  - *Fulmine* – Secure cluster (Jan 2016)
- **RISC-V based systems** (GF 28nm)
  - Honey Bunny (Nov 2015)

- **Early building blocks** (UMC180)
  - Sir10us
  - Or10n
- **Mixed-signal systems** (SMIC 130nm)
  - VivoSoC
  - EdgeSoC (in planning)
- **IcySoC chips approx. computing platforms** (ALP 180nm)
  - Diego
  - Manny
  - Sid



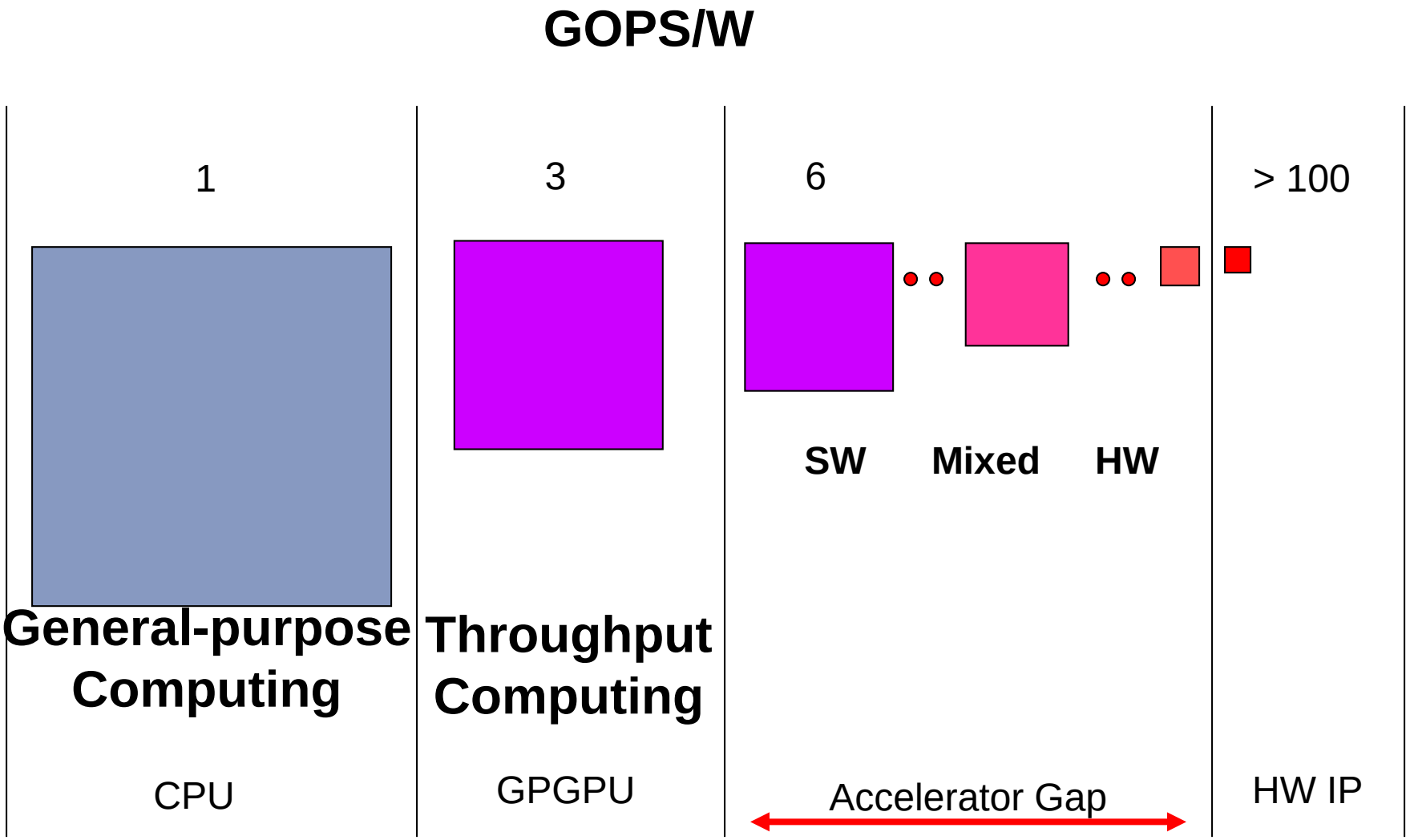


ETH zürich

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Pushing Beyond pJ/OP

Recovering more silicon efficiency



Closing The Accelerator Efficiency Gap with Agile Customization

Learn to Accelerate

- Brain-inspired (deep convolutional networks) systems are high performers in many tasks over *many domains*

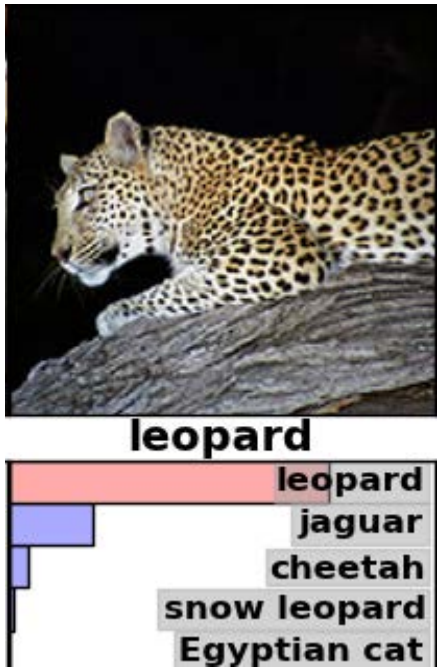
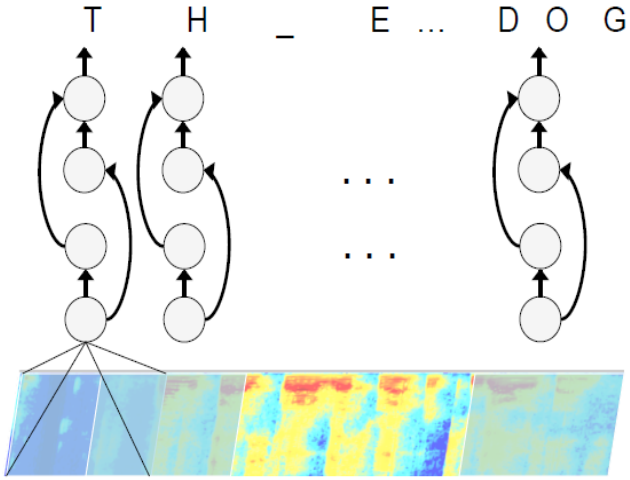


Image recognition
[Russakovsky et al., 2014]

CNN:
93.4% accuracy
(Imagenet 2014)
Human:
85% (untrained),
94.9% (trained)

[Karpahy15]

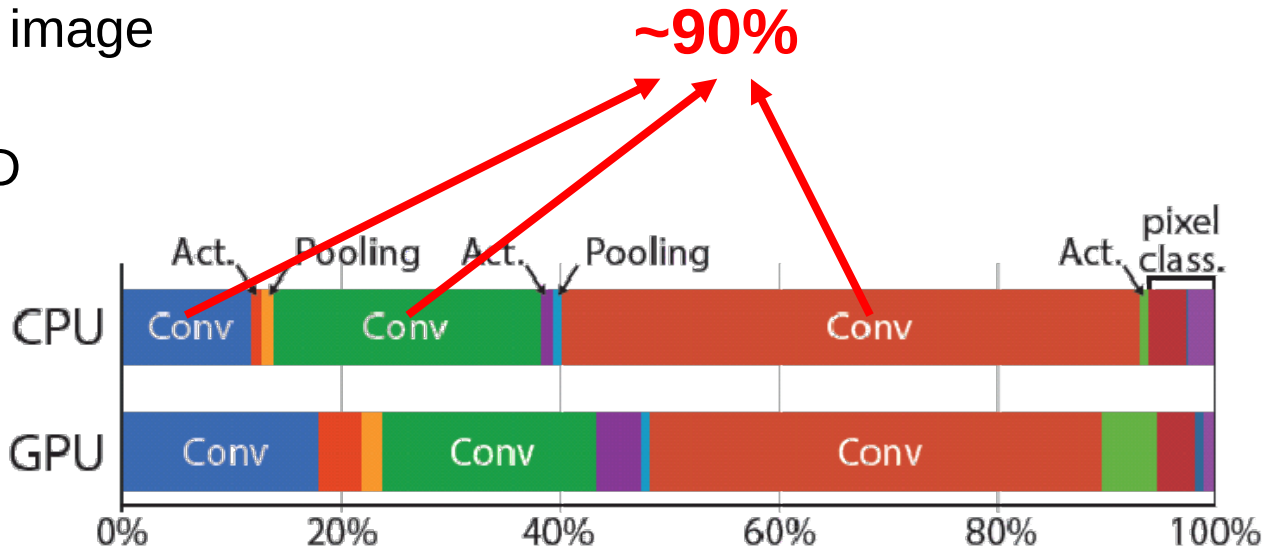


Speech recognition
[Hannun et al., 2014]

- **Flexible** acceleration: learned CNN weights are “the program”

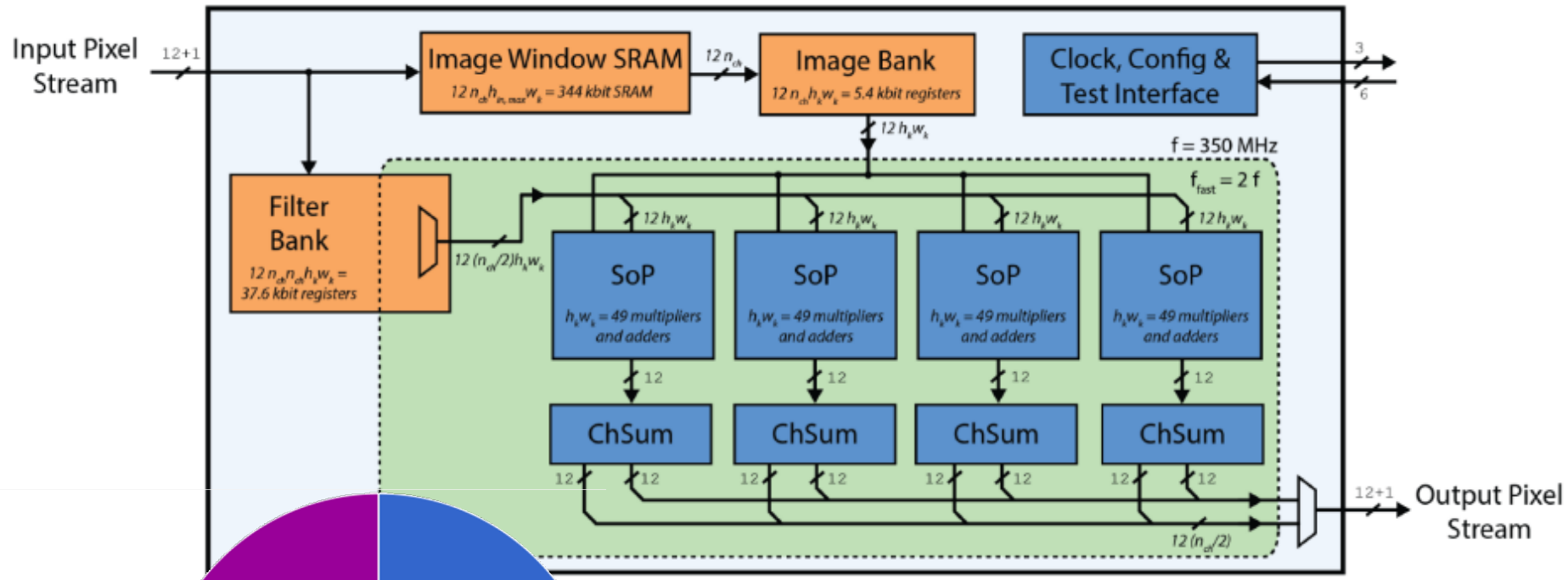
Computational Effort

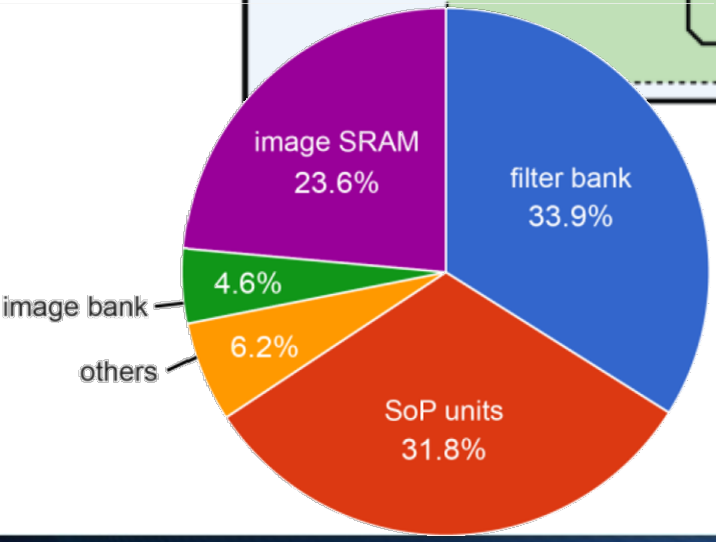
- Computational effort
  - 7.5 GOp for 320x240 image
  - 260 GOp for FHD
  - 1050 GOp for 4k UHD



Origami chip

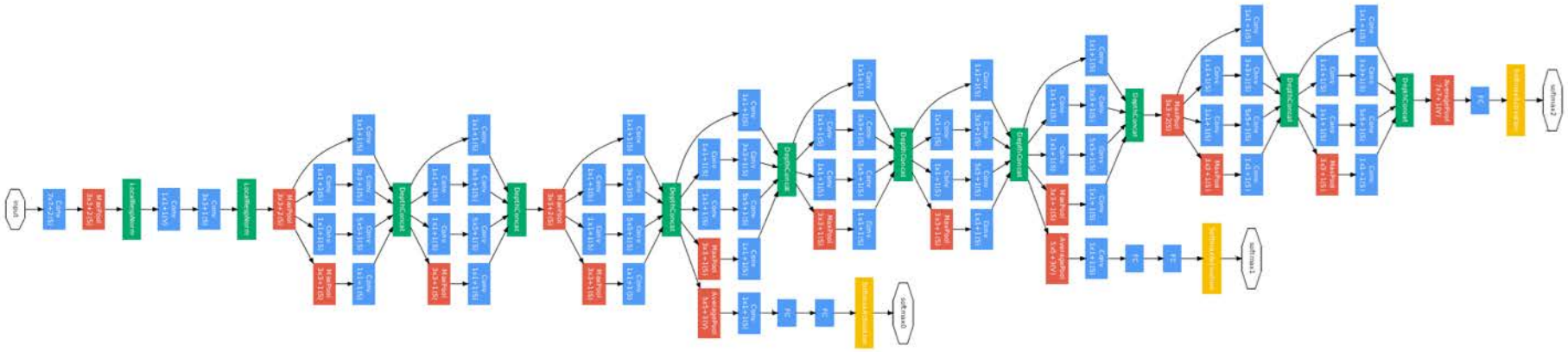
Origami: A CNN Accelerator





- FP needed?
  - 12-bit signals sufficient
  - Input to classification double-vs-12-bit accuracy loss $< 0.5\%$ (80.6% to 80.1%)

CNNs: typical workload



Example: **GoogLeNet** [ILSVRC 2014 winner]

~**7x10^6** parameters

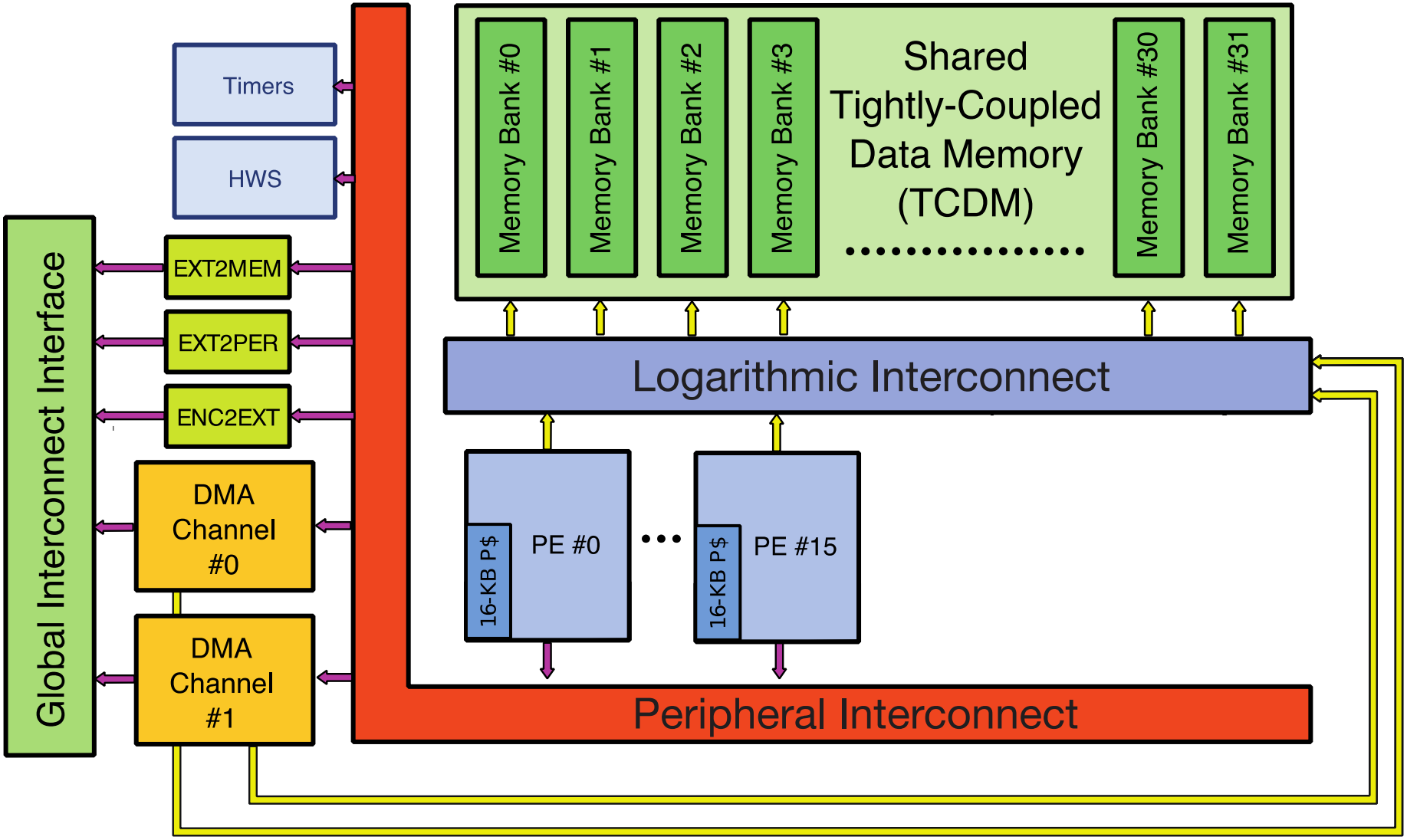
~**2.3x10^9** MAC operations on a **320x240** RGB image

Realtime (10 fps): ~**23 GMAC/s** performance

Realtime & Low-Power (10 fps @ 10mW): ~**2300 GMAC/s/W** efficiency

Origami core in 28nm FDSOI → GoogLeNet with ~10mW

Fractal Heterogeneity



Pushing Further: YodaNN[1]

- Approximation at the algorithmic side →Binary weights
- BinaryConnect [Courbariaux, NIPS15]
  - Reduce weights to a binary value -1/+1
  - Stochastic Gradient Descent with Binarization in the Forward Path

$$w_{b,stoch} = \begin{cases} -1 & p_{-1} = \sigma(w) \\ 1 & p_1 = 1 - p_{-1} \end{cases} \qquad w_{b,det} = \begin{cases} -1 & w < 0 \\ 1 & w > 0 \end{cases}$$

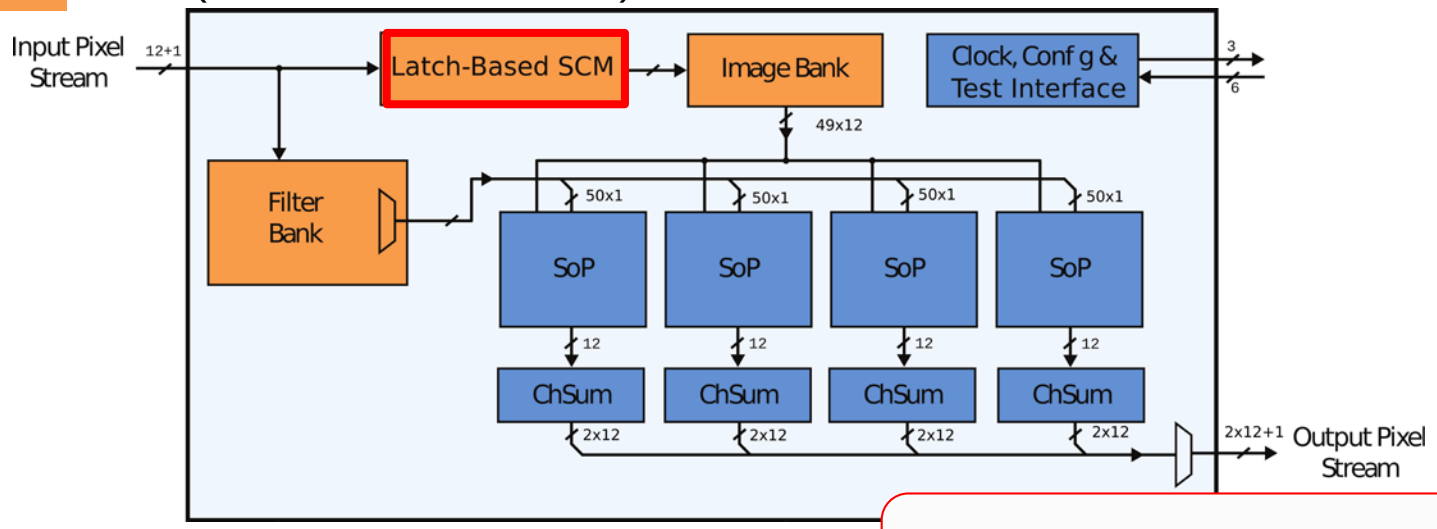
  - Learning large networks is still an issue with binary connect…

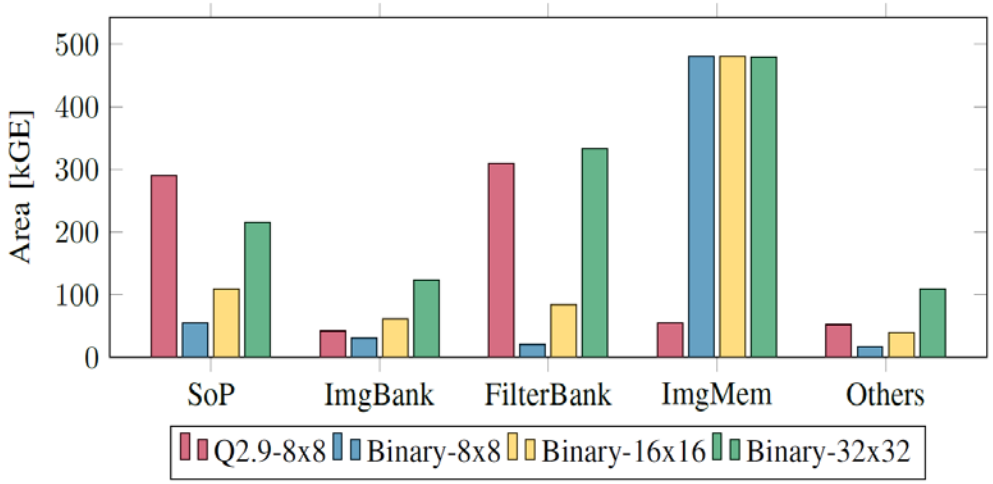
- Ultra-optimized HW is possible!
  - Power reduction because of arithmetic simplification
  - Major arithmetic density improvements
    - Area can be used for more energy-efficient weight storage
  - SCM memories for lower voltage → E goes with $1/V^2$

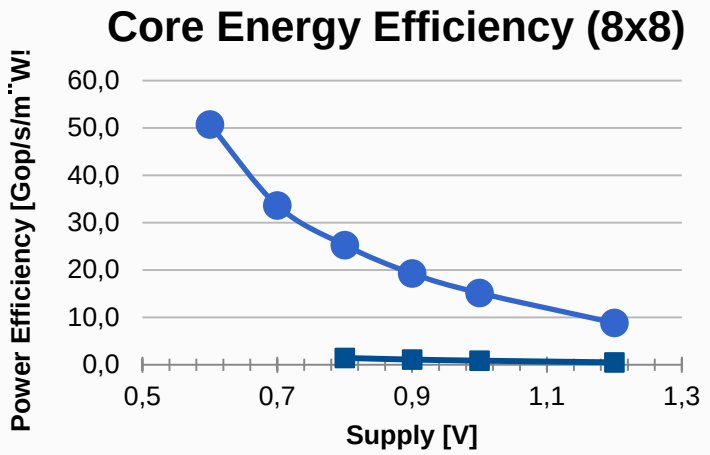
[1]After the Yedi Master from Star Wars - “Small in size but wise and powerful” cit. www.starwars.com

SCM for Energy efficiency

SCM (down to 0.6V in 65nm)







16x Energy efficiency improvement: 0.5pJ/OP →50GOPS/mW

ETH zürich

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

PULP infrastructure: not only hardware…

Using PULP: emulation

The PULP cluster is RTL-equivalent to the one used in ASICs

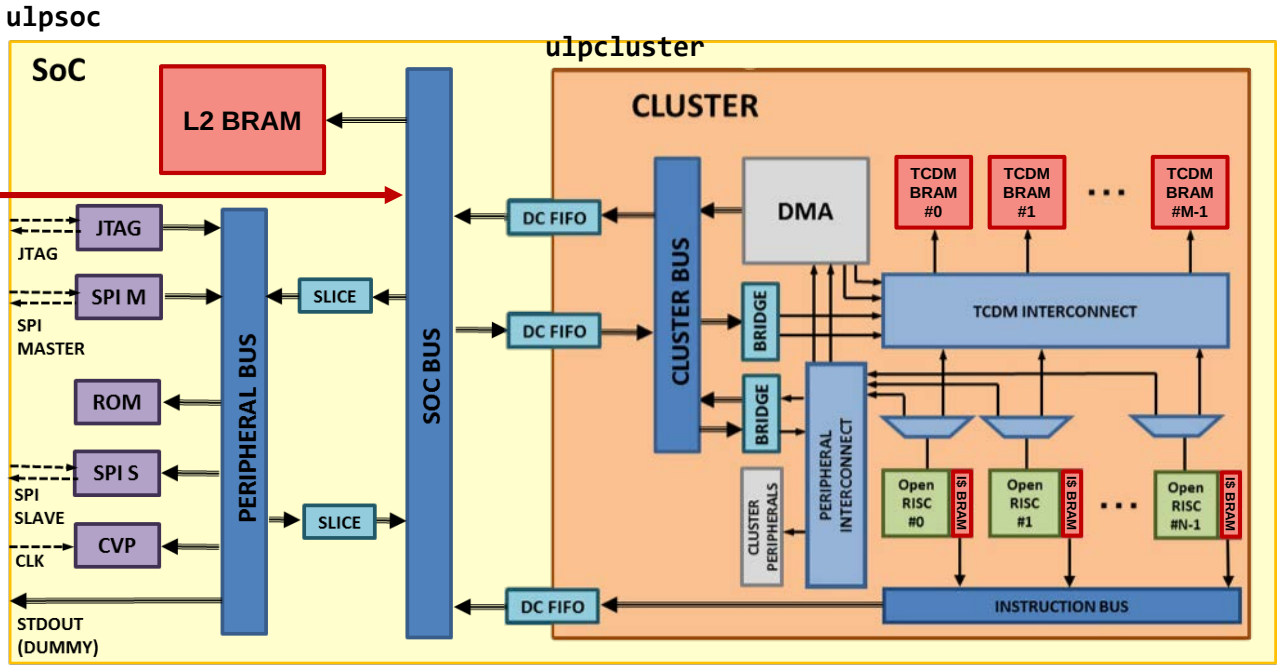
pulpemu

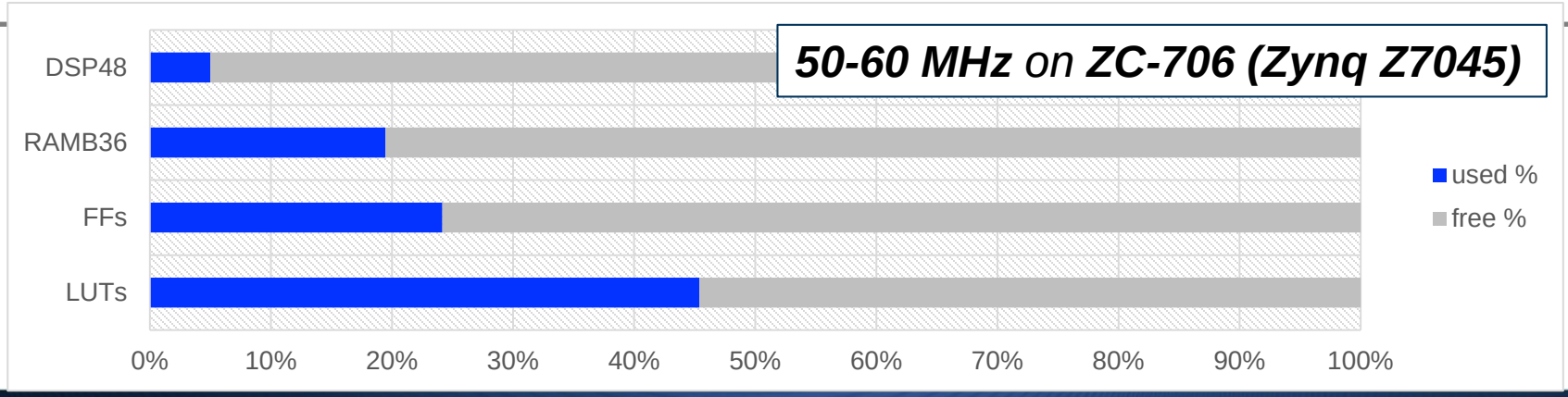
PULP is mapped in the ARM memory using a *mmap()* syscall

`rab bus`

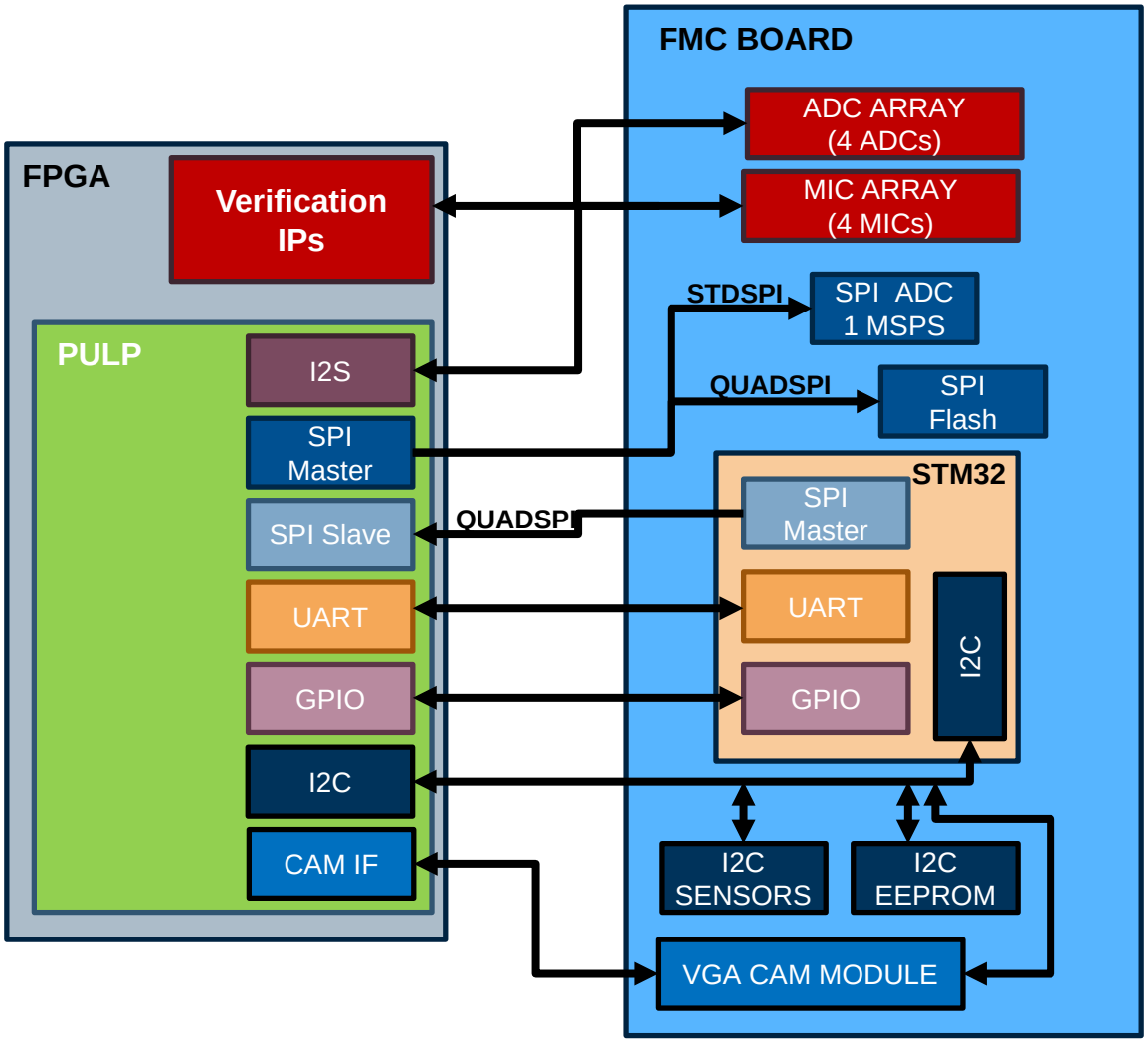
Zynq PS

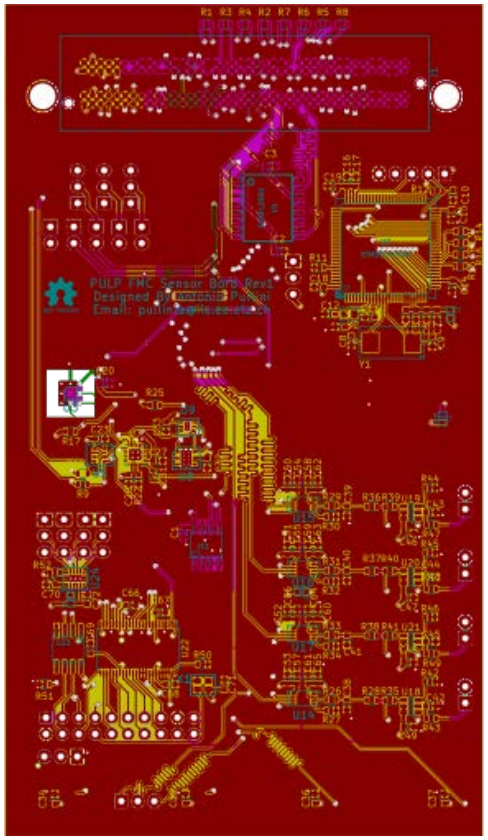
Runs **Ubuntu Linux** with PULPemu debug/support tool (*WinstonWolf*)
ARM as "smart traffic generator"





Using PULP: emulation





Board design is **open source**
(part of **PULPino** release)

Using PULP: toolchain support

Two **toolchains**:

- GCC 4.9
- LLVM 3.7

Transparent compiler support for most HW features:

- Hardware loops, Post-Increment, Register offset, MAC, vectorization…
- Bit-counting operations (`l.ff1`, `l.fl1`, `l.cnt`) supported through intrinsics
- All extensions can be disabled via compiler flags
- *libc* support (I/O based on semi-hosting)

Performance in industry standard benchmark:

- **2.37** CoreMark/MHz with **GCC**
- **2.16** CoreMark/MHz with **LLVM**

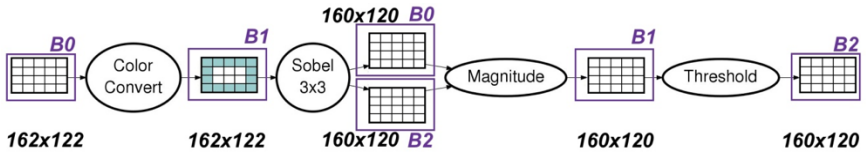
Using PULP: runtime support

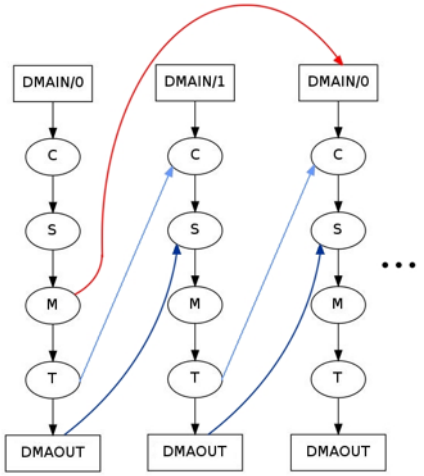
Full **OpenMP v3.0** support:

- Optimized for performance and power with HW support (event unit + HW test-and-set)
- 4-core parallel region: **70 cycles**
- barrier: **25 cycles**
- critical section: **60 cycles**

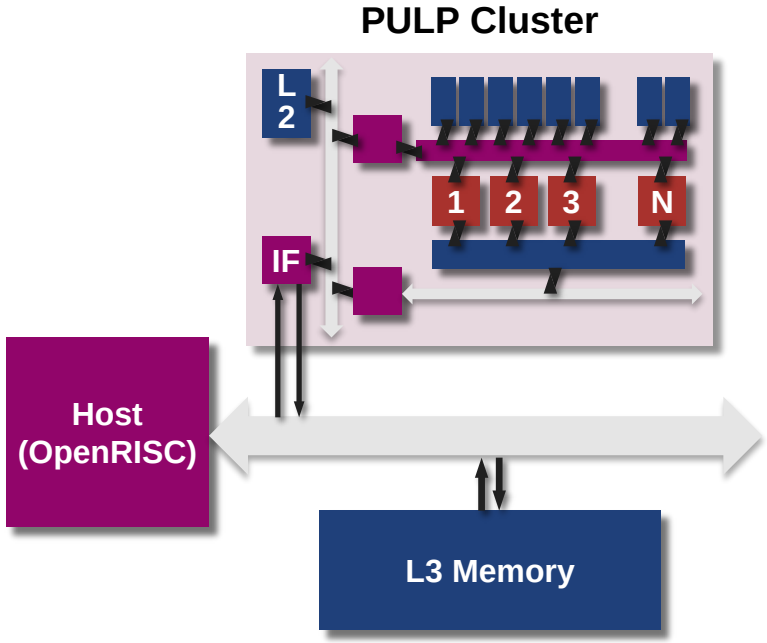
OpenVX support on PULP virtual platform:

- Vision acceleration layer for mobile/embedded, C-based
- Application = Directed Acyclic Graph with data as linkage
- *Khronos* runtime running on host OR10N
- Kernels are parallized with OpenMP on PULP
- Automatic tiling and insertion of DMA transfers





Using PULP: virtual platform



Virtual platform implementation:

- *C++* for fast native simulation
- *Python* for instantiation + configuration
- SWIG is used to generate C++ stubs from Python
- Any ISS can be integrated (*or1ksim*, *gdb* simulator, *riscv* on-going)

Timing model:

- Fully-event based, instances can generate events at specific time
- Includes timing models for interconnects, DMACs, memories…

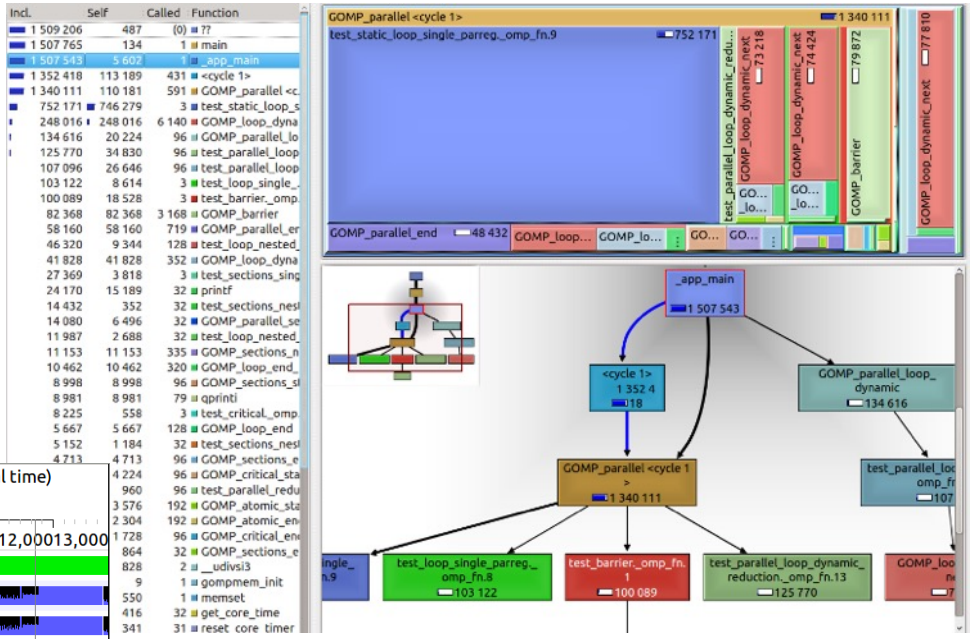
Simulation performance:

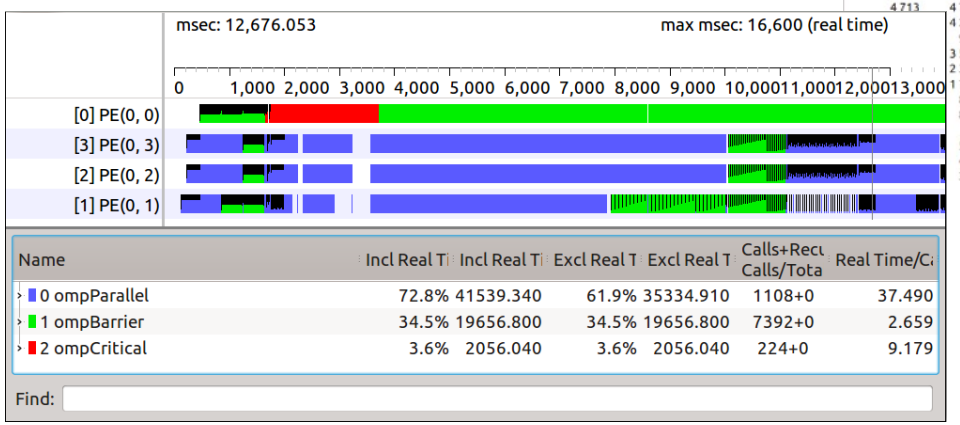
- Around 1MIPS simulation speed
- Functionally aligned with HW
- Timing accuracy is within **10-20%** of target HW

Using PULP: profiling & debug

Profiling based on **KCachegrind**:

- Supports PC traces from RTL and virtual platform (FPGA emulator on-going)
- Several events can be catched (PC, cycles, I$ miss, stalls…)



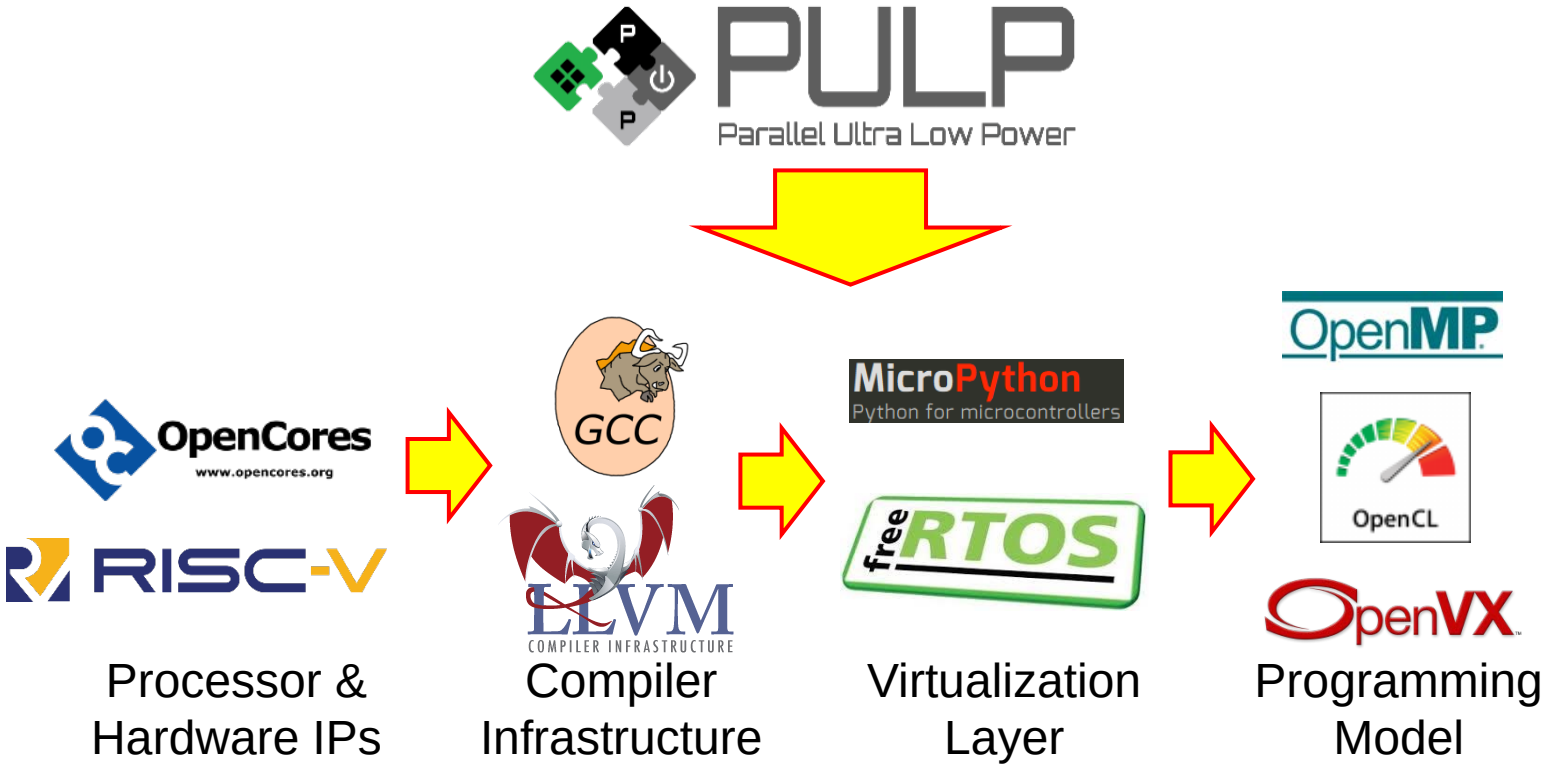


Debug with **GDB**:

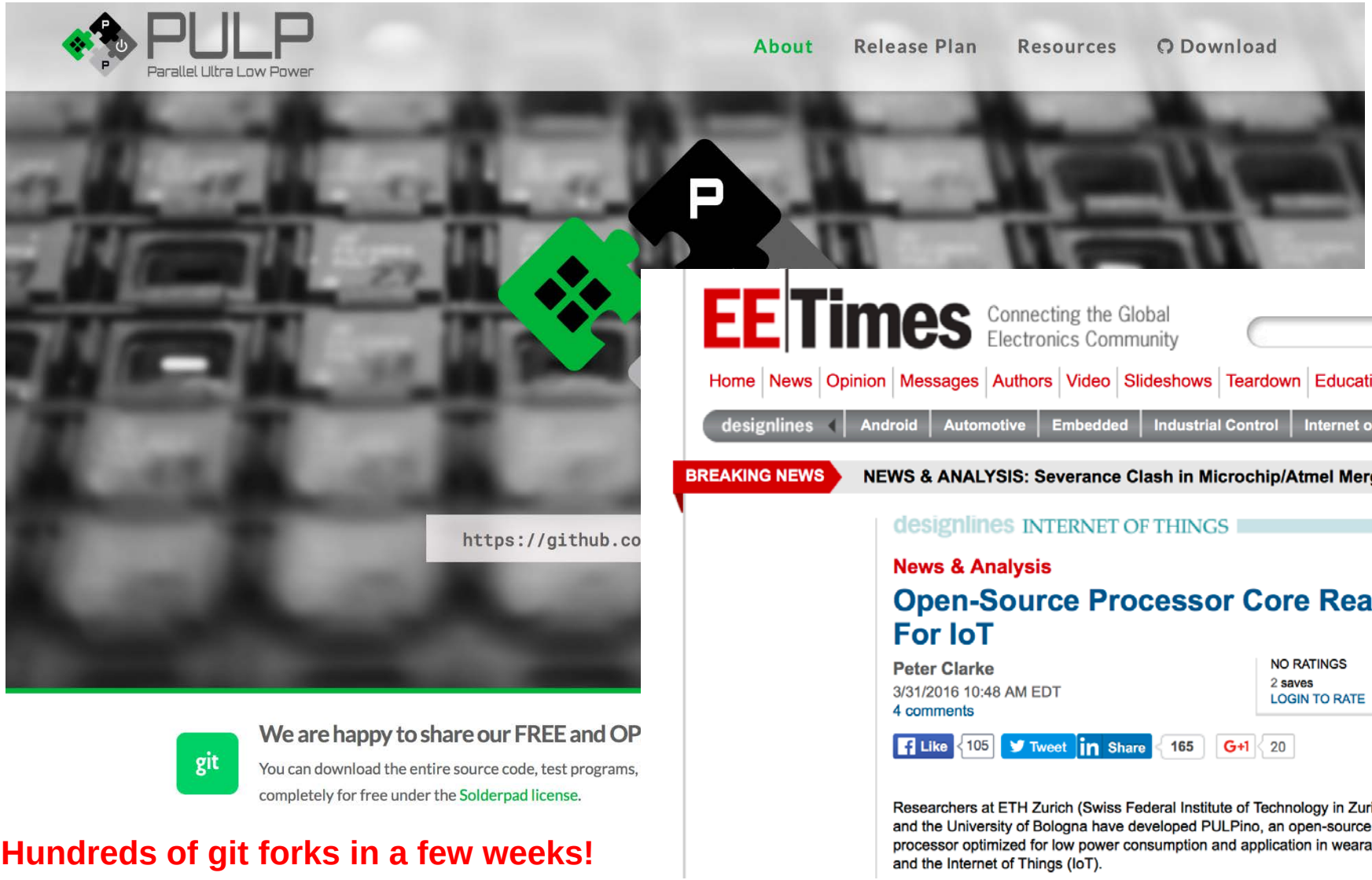
- Supports RTL and virtual platform
- Uses a bridge to inject JTAG requests
- Main GDB features work (step-by-step, breakpoints, introspection)
- Forked from *minsoc* project

Open Source
Parallel ULP computing for the IoT

(sub)-pJ/op computing platform **- let's make it Open!**



ETHZ and UNIBO released PULPino



Hundreds of git forks in a few weeks!

ETH zürich

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Thanks for your attention!!!

PULP
Parallel Ultra Low Power

www.pulp-platform.org
www-micrel.deis.unibo.it/pulp-project
iis-projects.ee.ethz.ch/index.php/PULP