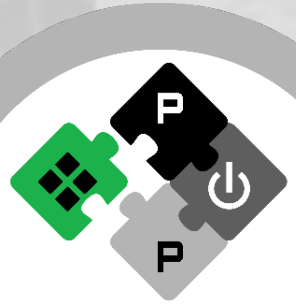


Flexible and Scalable Acceleration Techniques for Low-Power Edge Computing

2nd Italian Workshop on Embedded Systems

Università degli Studi di Roma "La Sapienza„

8.9.2017

Francesco Conti[1,2], Davide Rossi[1], Luca Benini[1,2]

f.conti@unibo.it

ALMA MATER STUDIORUM A.D. 1088
UNIVERSITÀ DI BOLOGNA

[1]Energy Efficient Embedded Systems Laboratory

ETH zürich

[2]Integrated Systems Laboratory

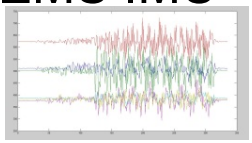
Computing for the Internet of Things

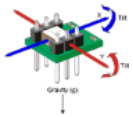
Battery + Harvesting powered
→ a few mW power envelope

Computing for the Internet of Things

Sense

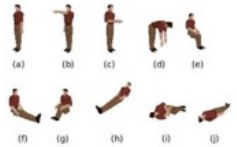
MEMS IMU



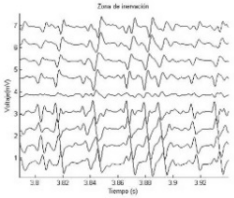


MEMS Microphone

ULP Imager



EMG/ECG/EIT



100 µW ÷ 2 mW

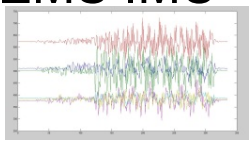
Battery + Harvesting powered
→ a few mW power envelope

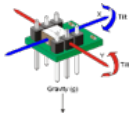
PULP

Computing for the Internet of Things

Sense

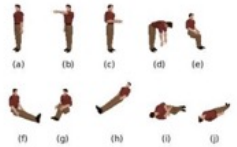
MEMS IMU



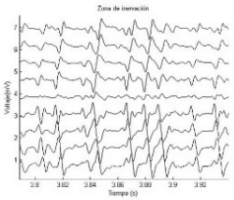


MEMS Microphone

ULP Imager

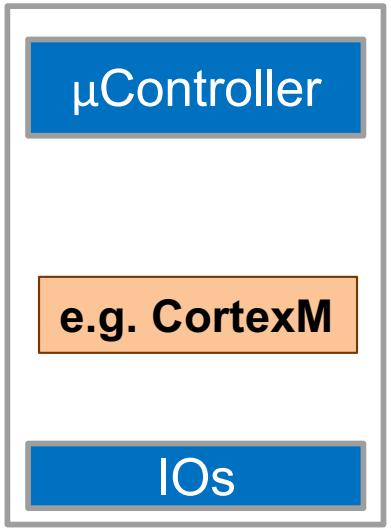


EMG/ECG/EIT



100 µW ÷ 2 mW

Analyze and Classify



1 ÷ 25 MOPS
1 ÷ 10 mW

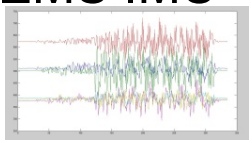
Battery + Harvesting powered
→ a few mW power envelope

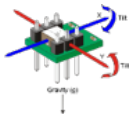
PULP

Computing for the Internet of Things

Sense

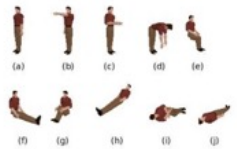
MEMS IMU



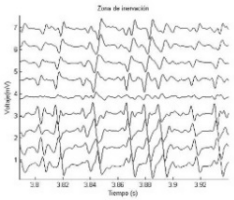


MEMS Microphone

ULP Imager



EMG/ECG/EIT



100 µW ÷ 2 mW

Analyze and Classify

µController

e.g. CortexM

IOs

1 ÷ 25 MOPS
1 ÷ 10 mW

Battery + Harvesting powered
→ a few mW power envelope

Transmit

Short range, medium BW

Low rate (periodic) data

SW update, commands

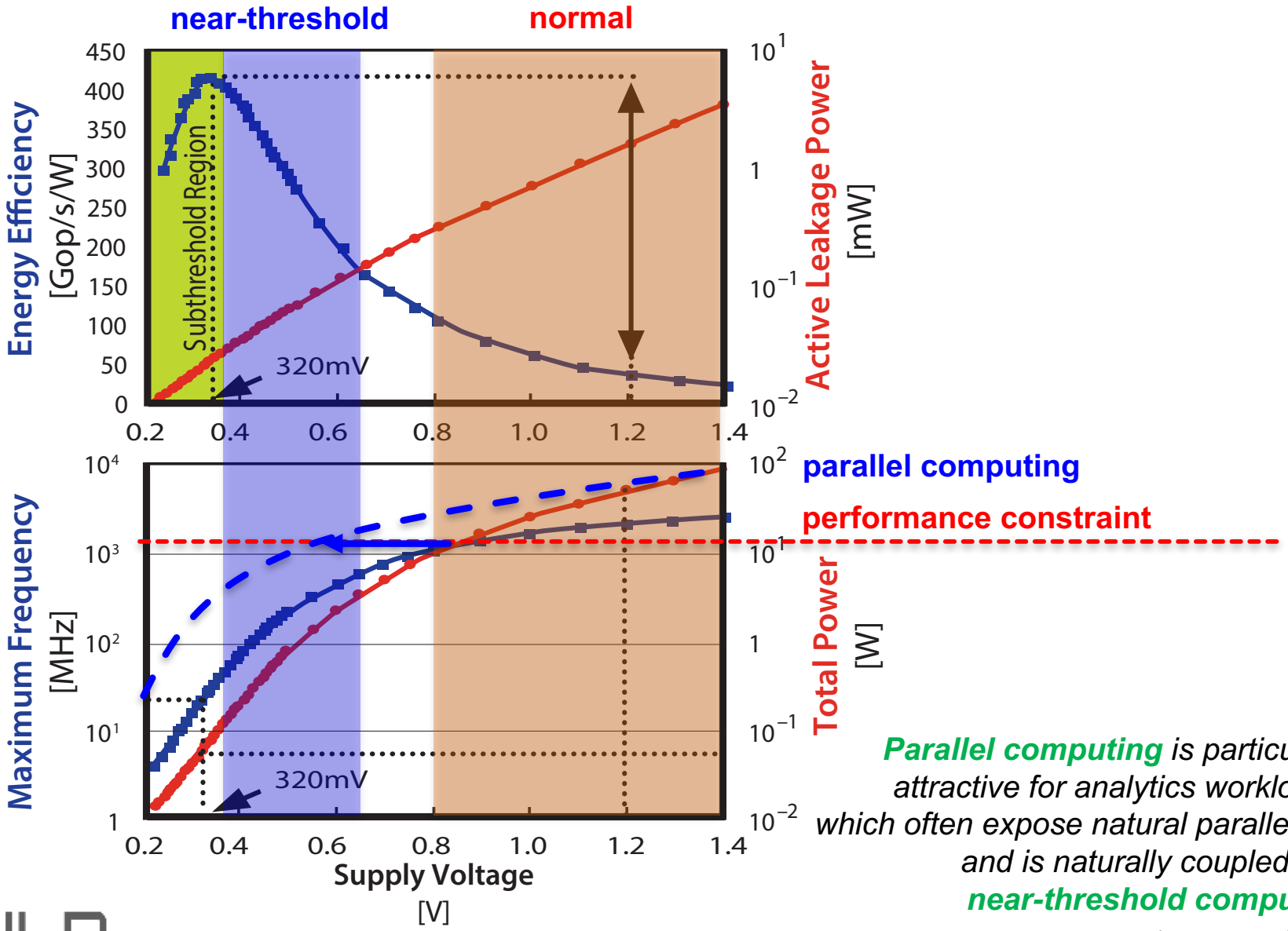
Long range, low BW

LoRa SEMTECH

Idle: ~1µW
Active: ~ 50mW

PULP

The Road to Efficiency



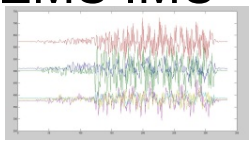
Adapted from Borkar and Chien, The Future of Microprocessors, *Communications of the ACM, May 2011*

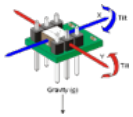
Parallel computing *is particularly attractive for analytics workloads, which often expose natural parallelism, and is naturally coupled with* ***near-threshold computing***

Computing for the Internet of Things

Sense

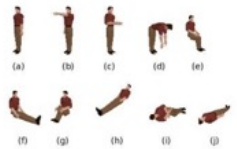
MEMS IMU



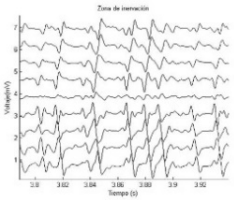


MEMS Microphone

ULP Imager

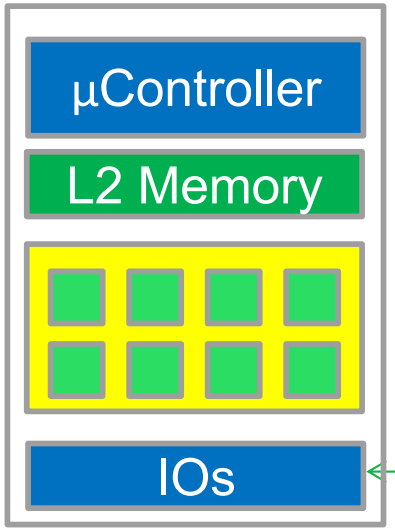


EMG/ECG/EIT



100 µW ÷ 2 mW

Analyze and Classify



1 ÷ 2000 MOPS
1 ÷ 10 mW

Battery + Harvesting powered
→ a few mW power envelope

Transmit

Short range, medium BW

Low rate (periodic) data

SW update, commands

Long range, low BW

LoRa SEMTECH

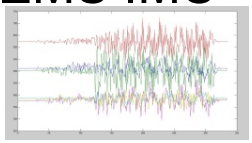
Idle: ~1µW
Active: ~ 50mW

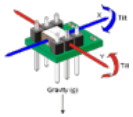
PULP

Computing for the Internet of Things

Sense

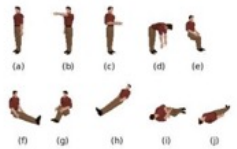
MEMS IMU



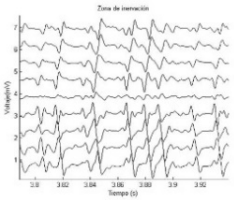


MEMS Microphone

ULP Imager

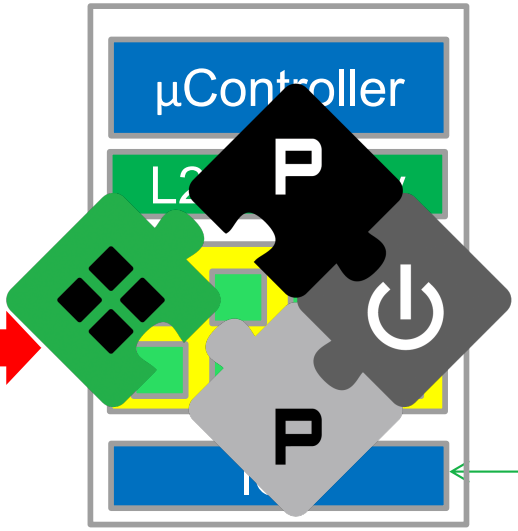


EMG/ECG/EIT



100 µW ÷ 2 mW

Analyze and Classify



1 ÷ 2000 MOPS
1 ÷ 10 mW

Battery + Harvesting powered
→ a few mW power envelope

Transmit

Short range, medium BW

Low rate (periodic) data

SW update, commands

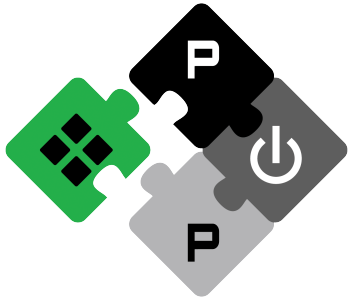
Long range, low BW

LoRa SEMTECH

Idle: ~1µW
Active: ~ 50mW

PULP

PULP architecture outline

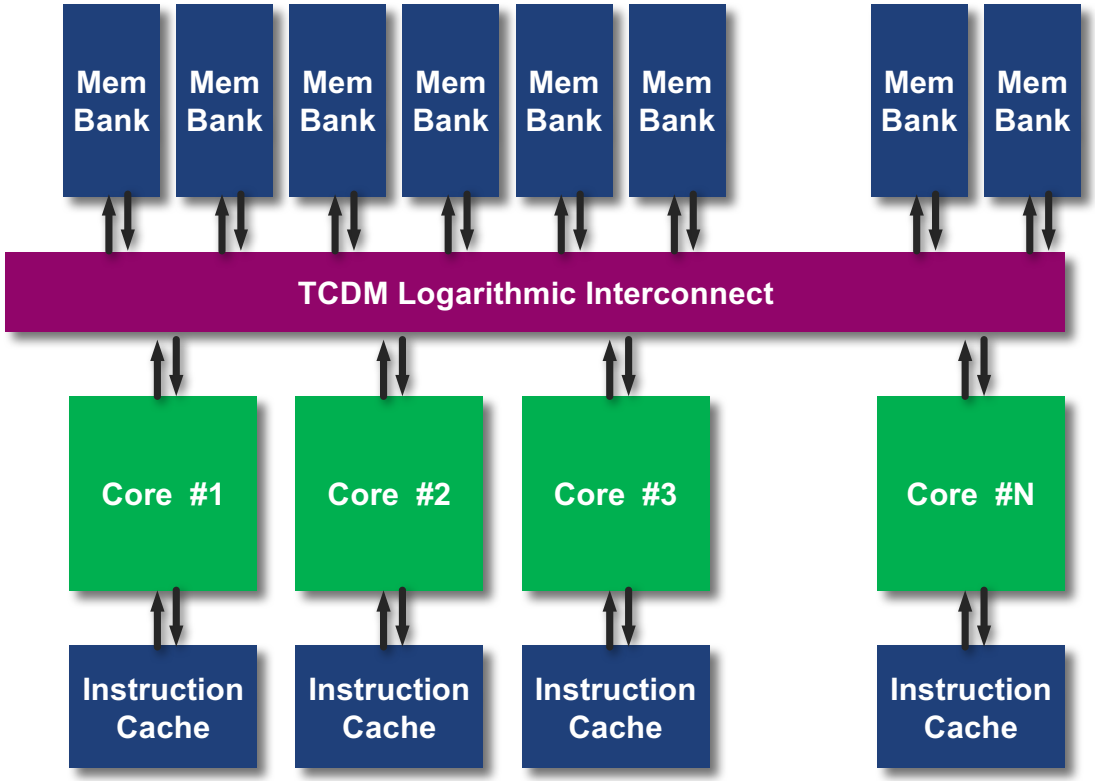


Parallel **U**ltra **L**ow **P**ower in a nutshell: **energy efficiency** for the IoT through

- near-threshold ULP execution
- parallel computing
- architecture targeted at low power

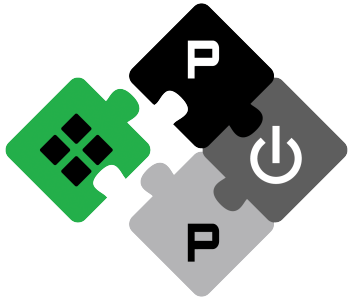
Targeting **100-1000 GOPS/W** of performance/Watt (>**100x** of current MCUs)

A joint effort of **University of Bologna**, **ETH Zurich** and other academic and industrial partners.



Parallel access to shared memory → Flexibility

PULP architecture outline

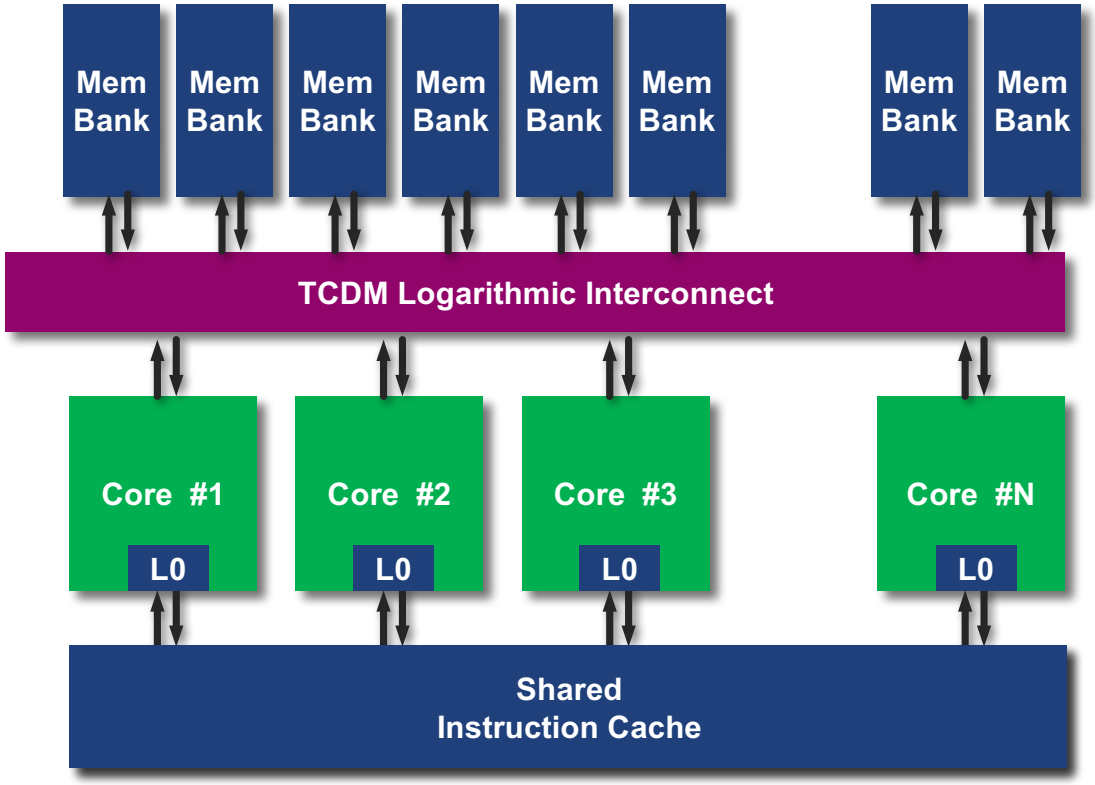


Parallel **U**ltra **L**ow **P**ower in a nutshell: **energy efficiency** for the IoT through

- near-threshold ULP execution
- parallel computing
- architecture targeted at low power

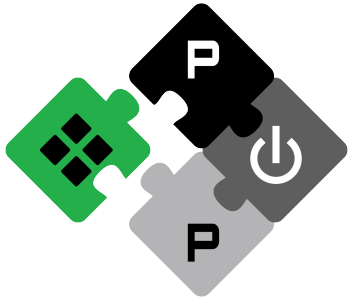
Targeting **100-1000 GOPS/W** of performance/Watt (>**100x** of current MCUs)

A joint effort of **University of Bologna**, **ETH Zurich** and other academic and industrial partners.



Shared I$ + L0 fetch buffer → Efficiency

PULP architecture outline

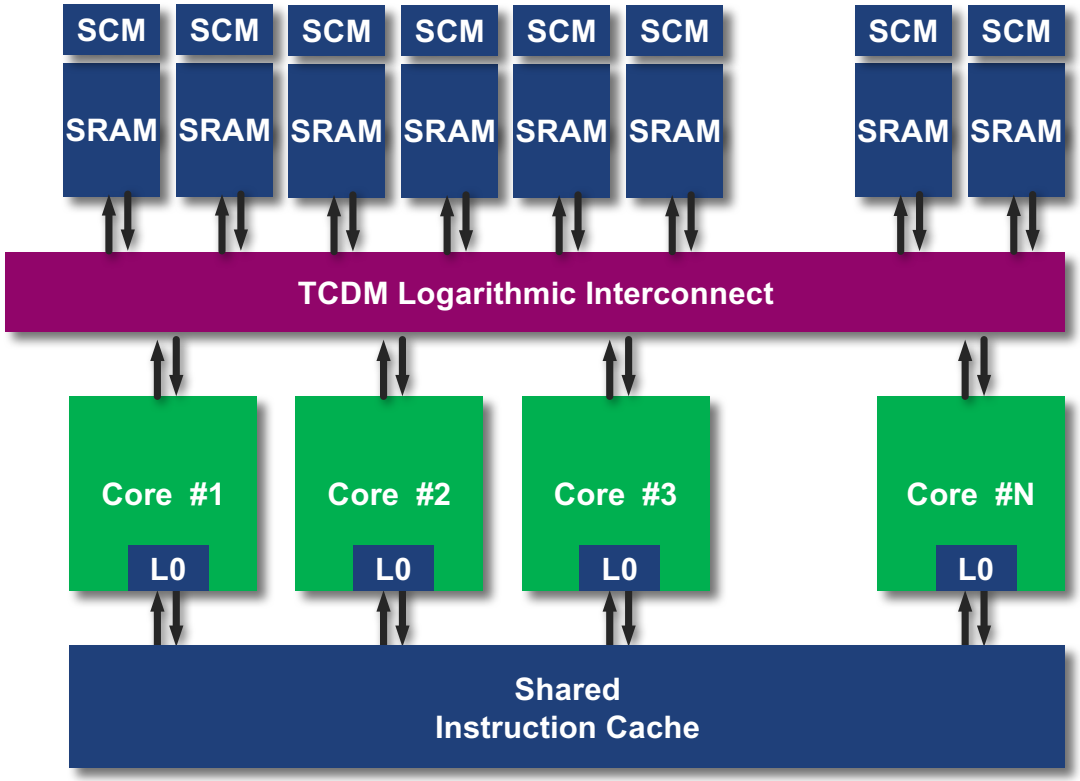


Parallel **U**ltra **L**ow **P**ower in a nutshell: **energy efficiency** for the IoT through

- near-threshold ULP execution
- parallel computing
- architecture targeted at low power

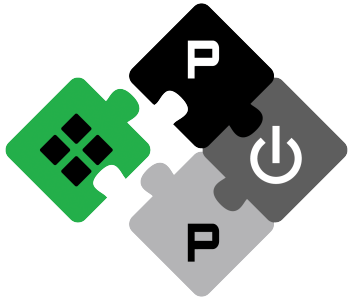
Targeting **100-1000 GOPS/W** of performance/Watt (>**100x** of current MCUs)

A joint effort of **University of Bologna**, **ETH Zurich** and other academic and industrial partners.



Hybrid memory: SRAM+SCM → can work at very low Vdd

PULP architecture outline

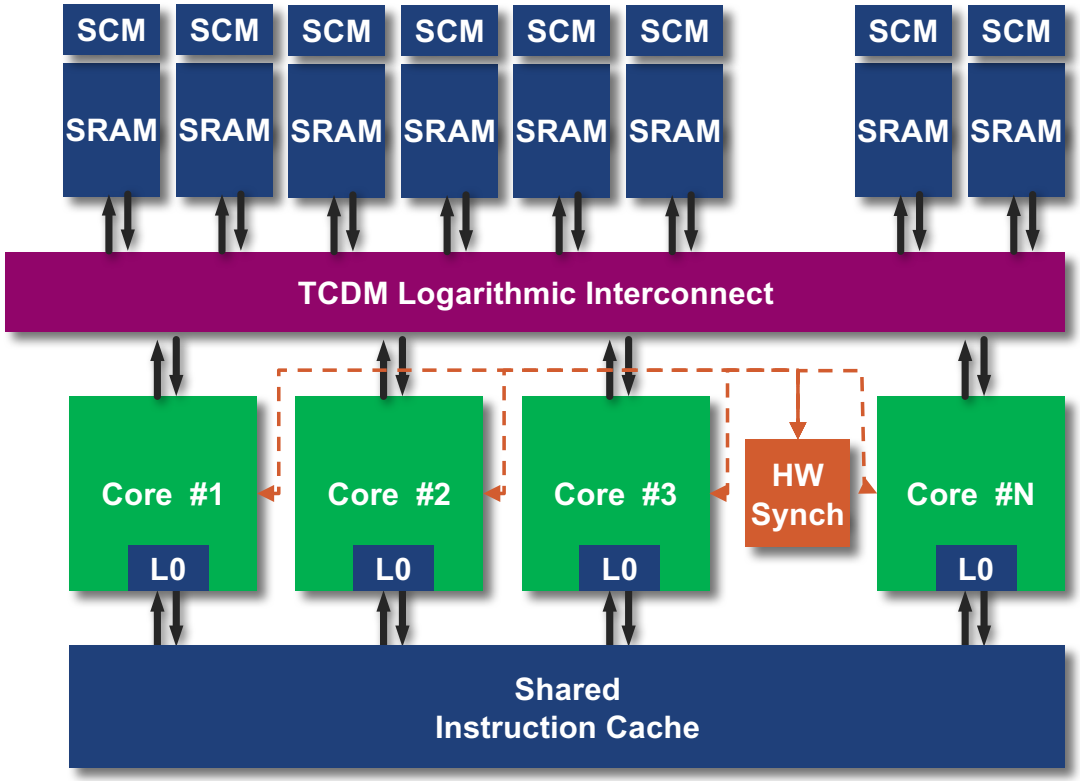


Parallel **U**ltra **L**ow **P**ower in a nutshell: **energy efficiency** for the IoT through

- near-threshold ULP execution
- parallel computing
- architecture targeted at low power

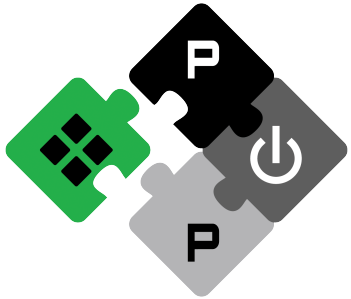
Targeting **100-1000 GOPS/W** of performance/Watt (>**100x** of current MCUs)

A joint effort of **University of Bologna**, **ETH Zurich** and other academic and industrial partners.



HW Synch → Faster core shutdown + parallelism

PULP architecture outline

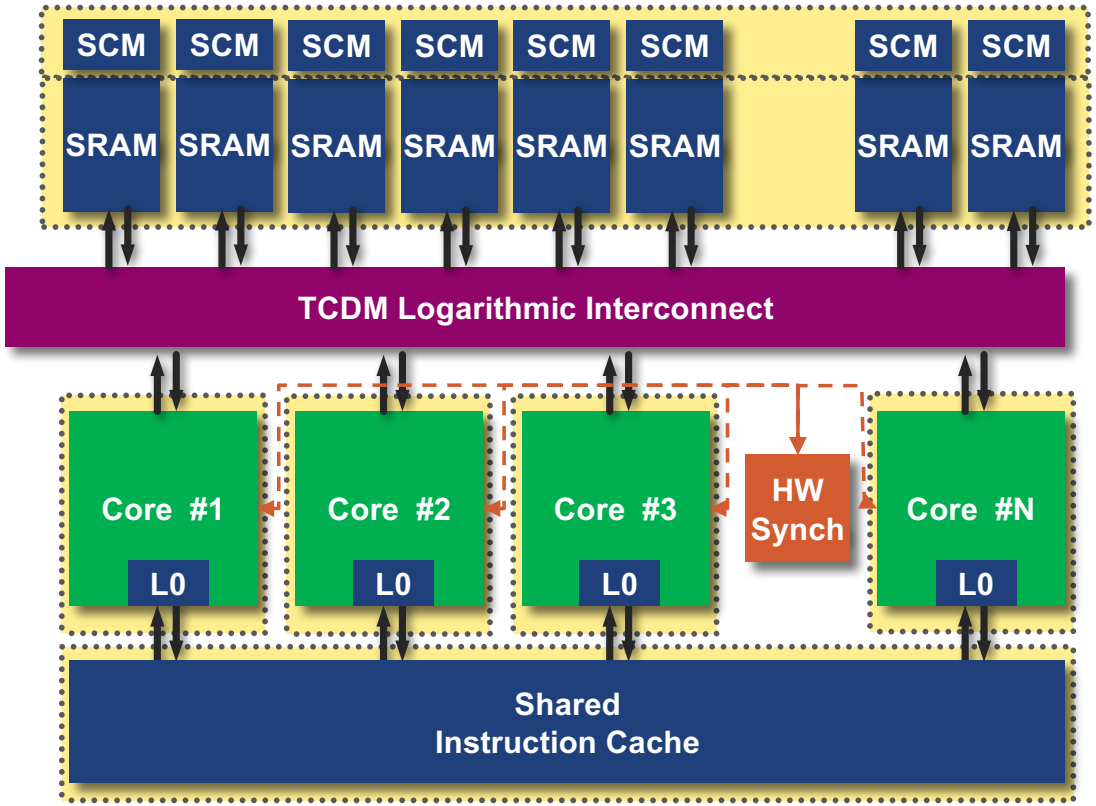


Parallel **U**ltra **L**ow **P**ower in a nutshell: **energy efficiency** for the IoT through

- near-threshold ULP execution
- parallel computing
- architecture targeted at low power

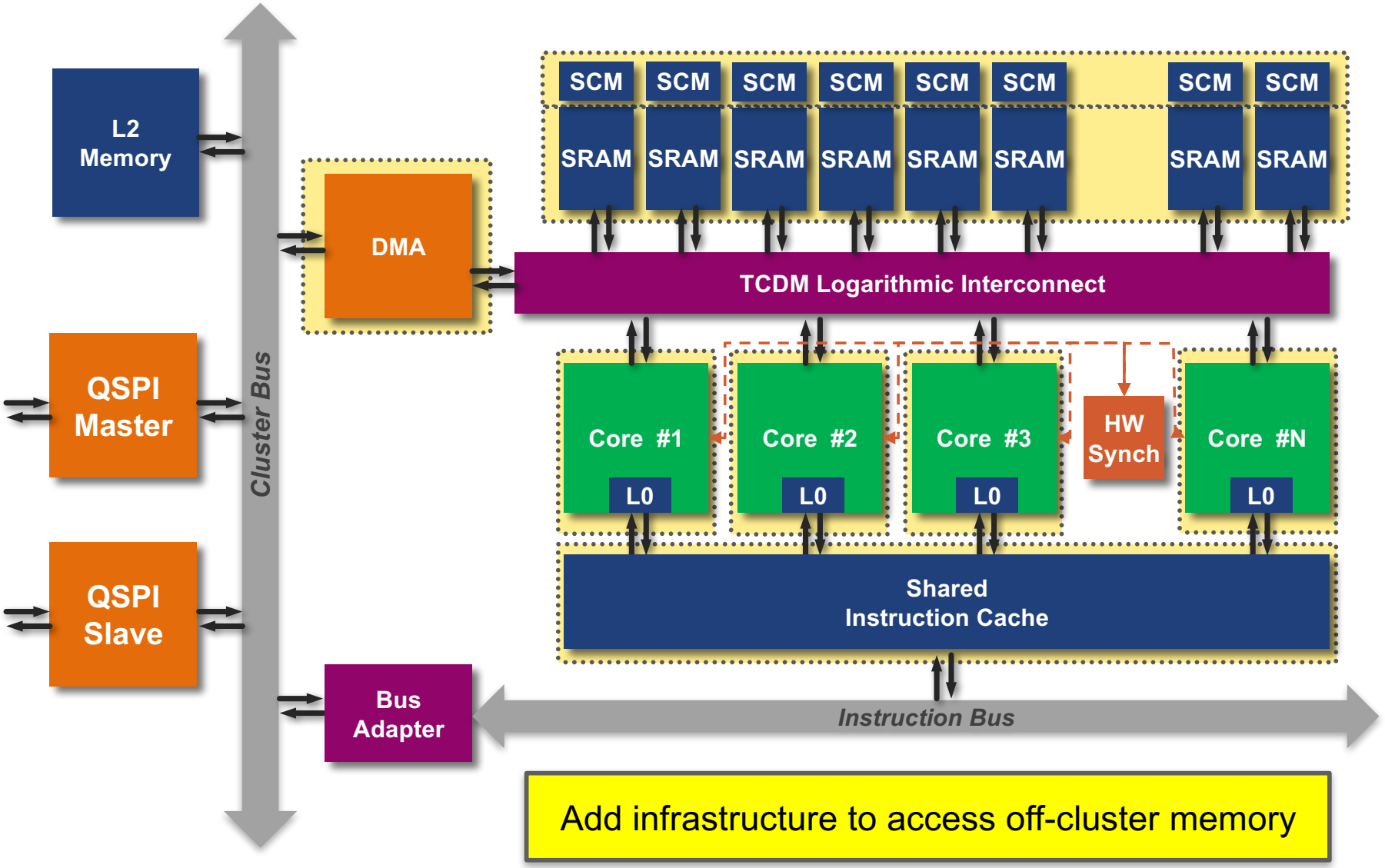
Targeting **100-1000 GOPS/W** of performance/Watt (>**100x** of current MCUs)

A joint effort of **University of Bologna**, **ETH Zurich** and other academic and industrial partners.



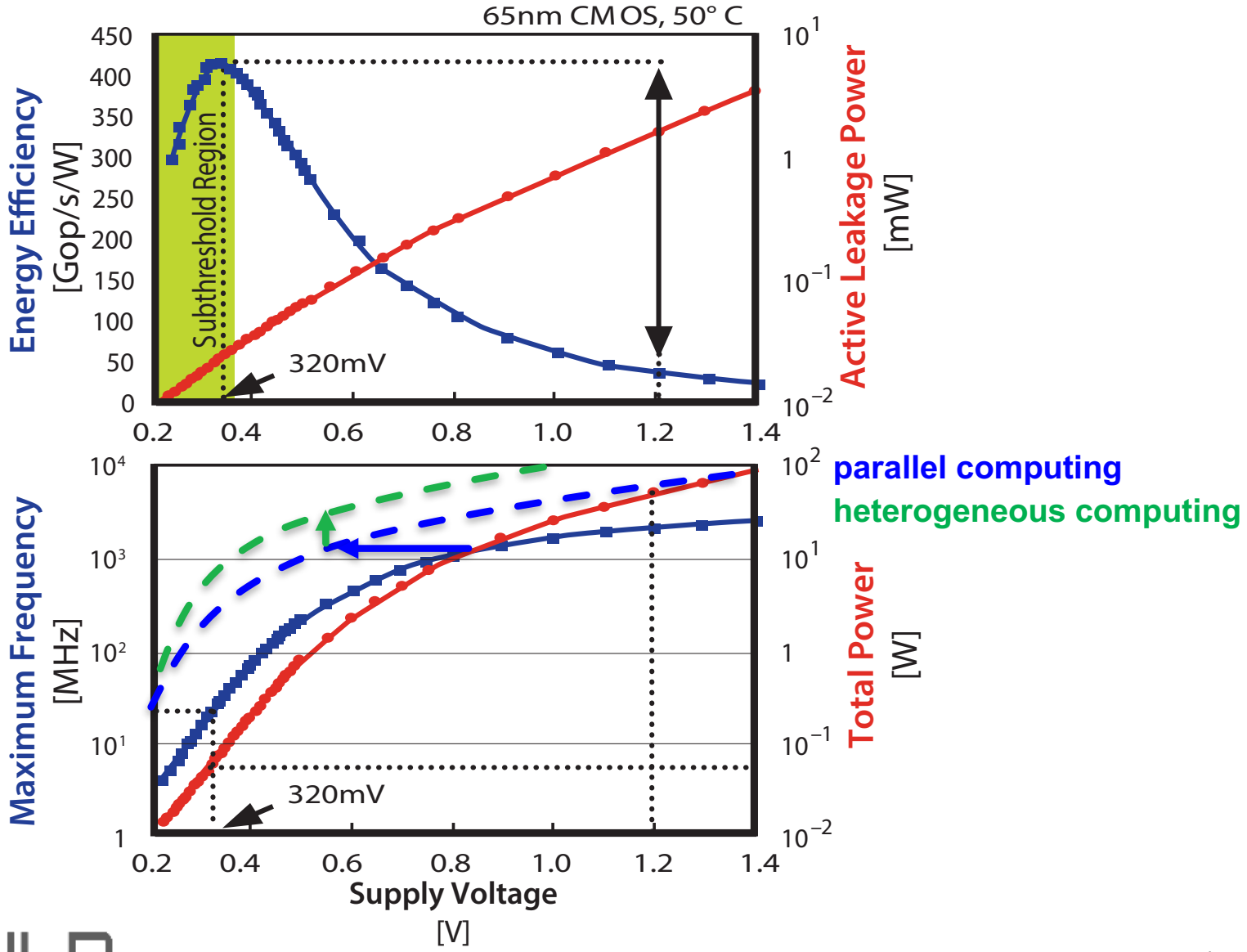
Fine-grain Clk-Gating + Body-Bias → Less Power

PULP architecture outline

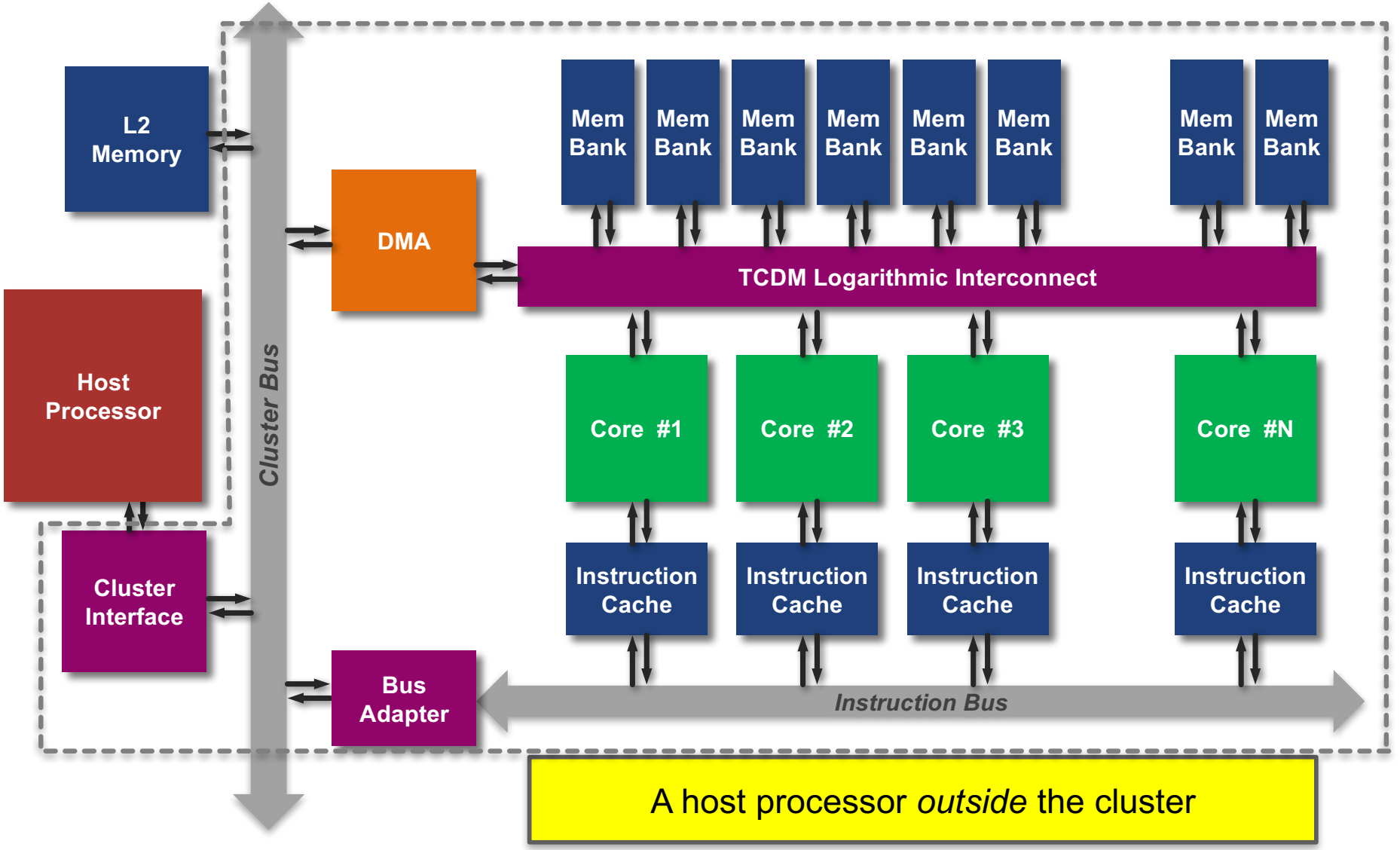


How to get even more efficient?

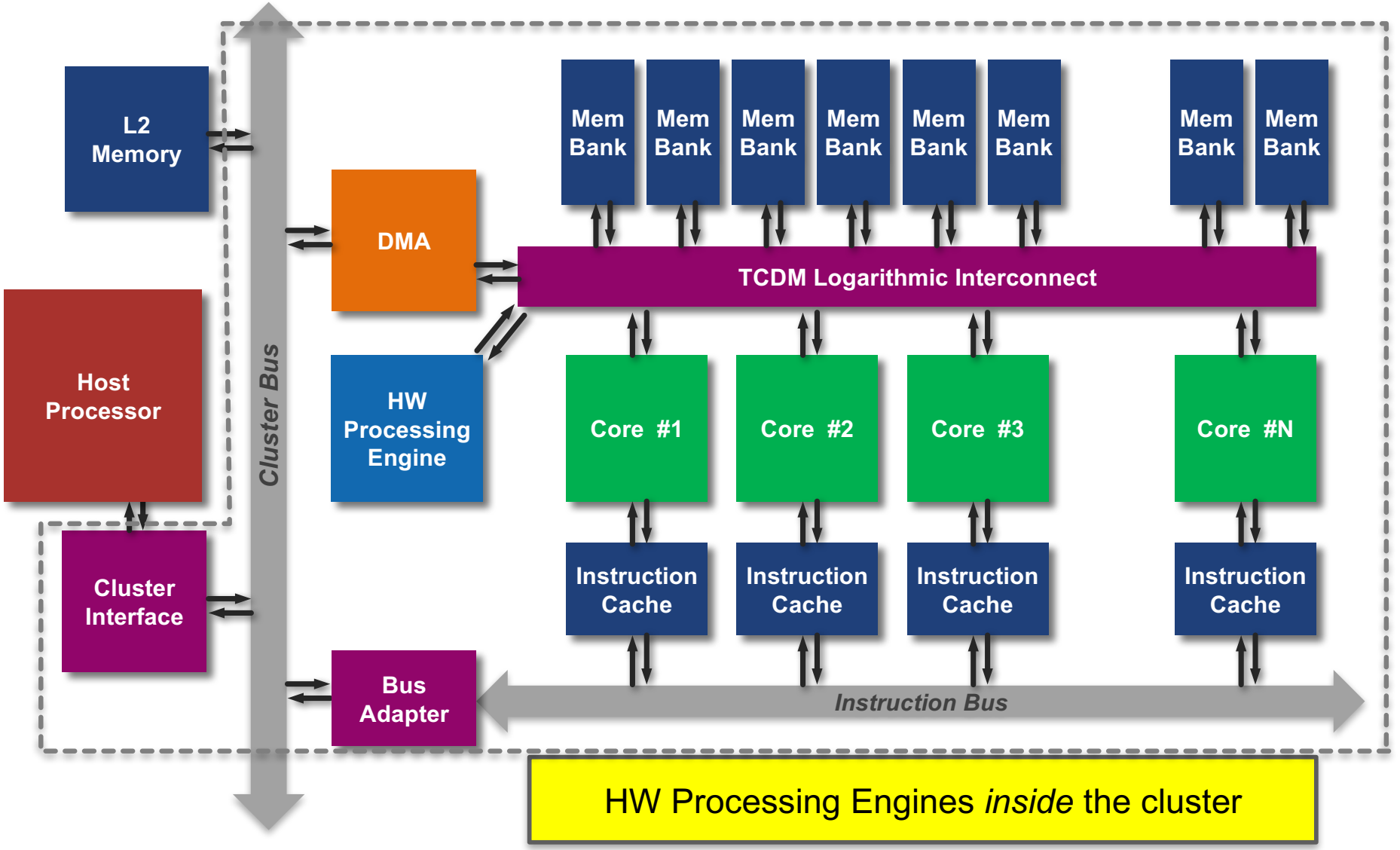
Adapted from Borkar and Chien, The Future of Microprocessors, *Communications of the ACM, May 2011*



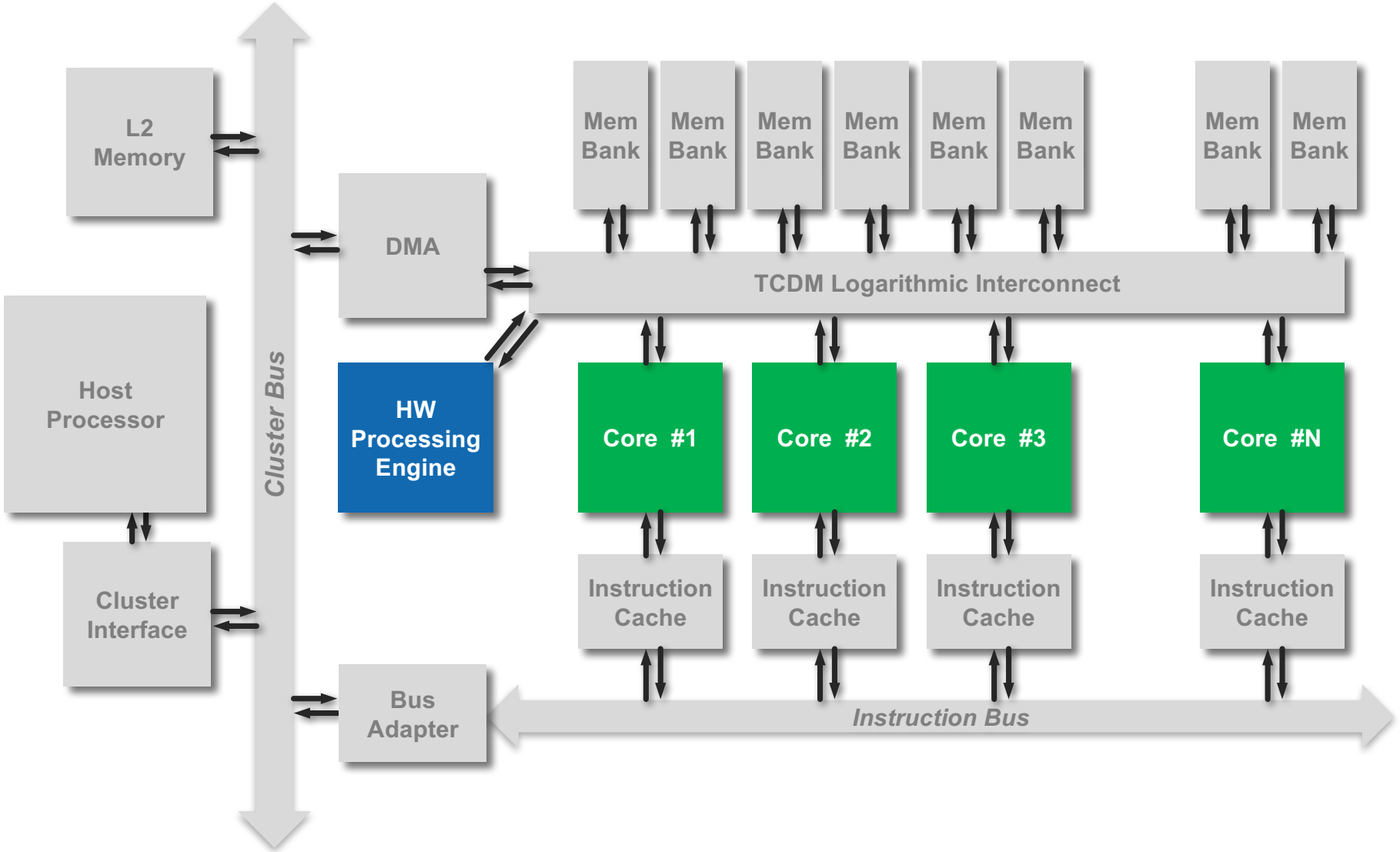
HW Acceleration in Tightly-Coupled Clusters



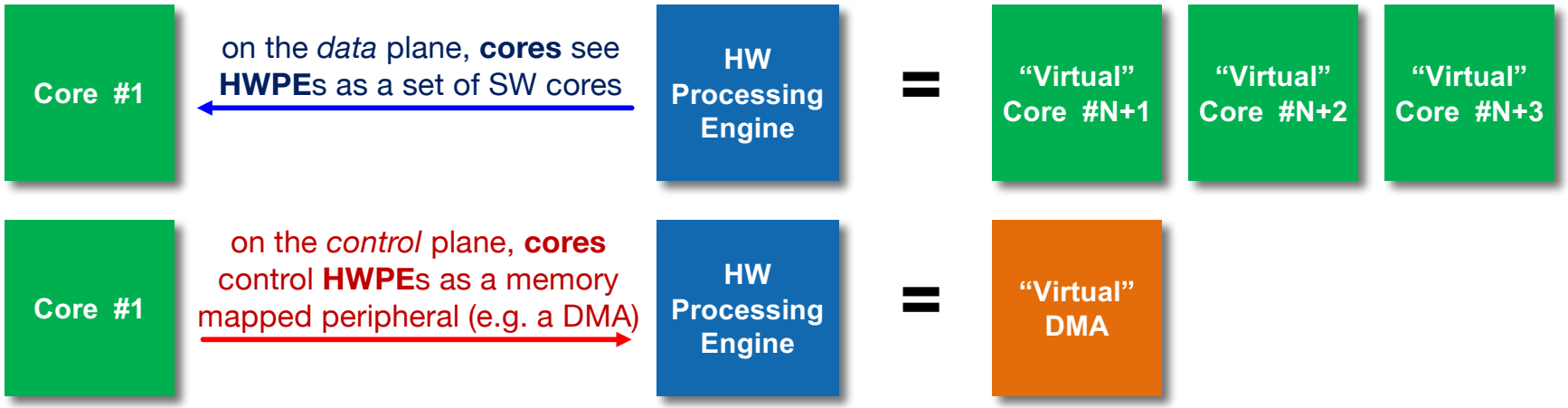
HW Acceleration in Tightly-Coupled Clusters



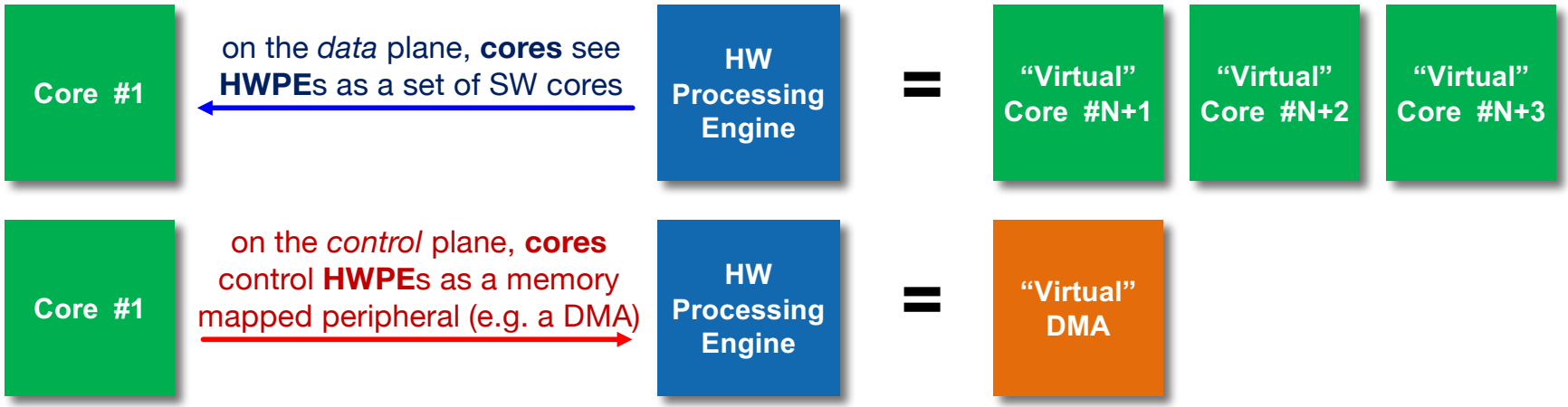
HW Acceleration in Tightly-Coupled Clusters

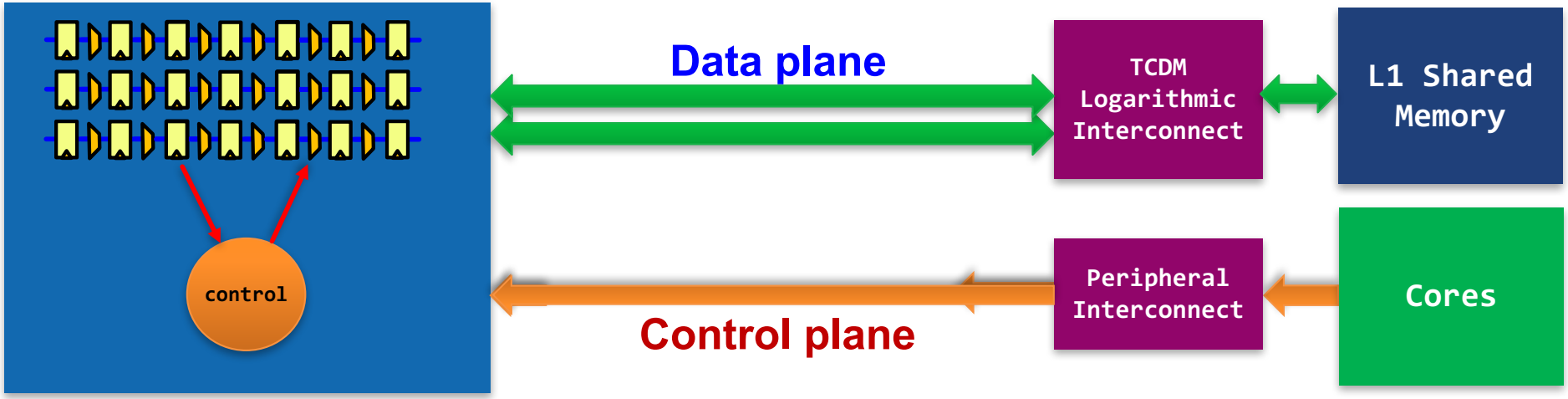


HW Processing Engines

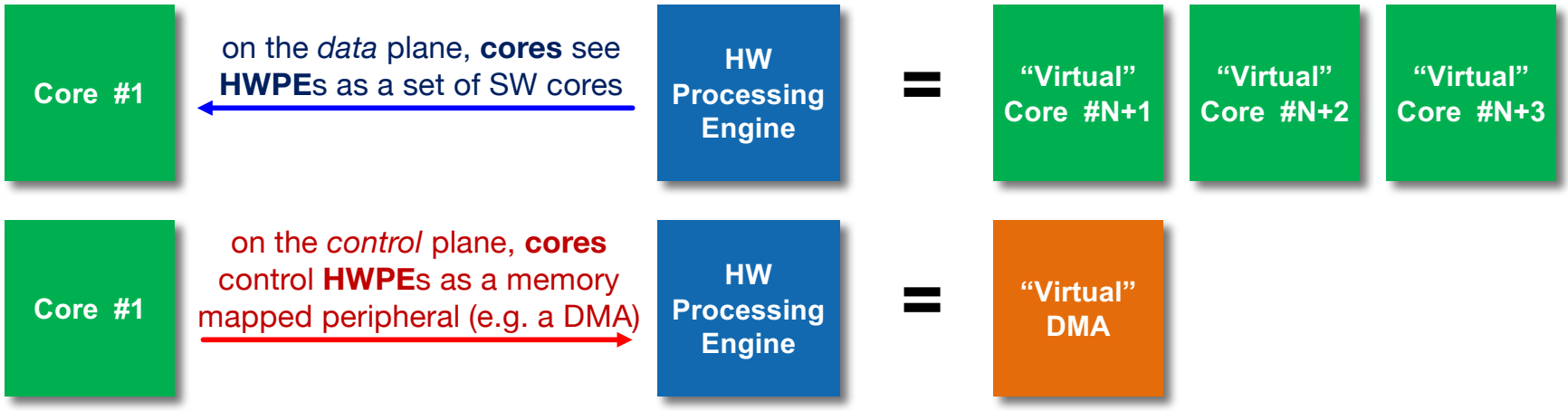


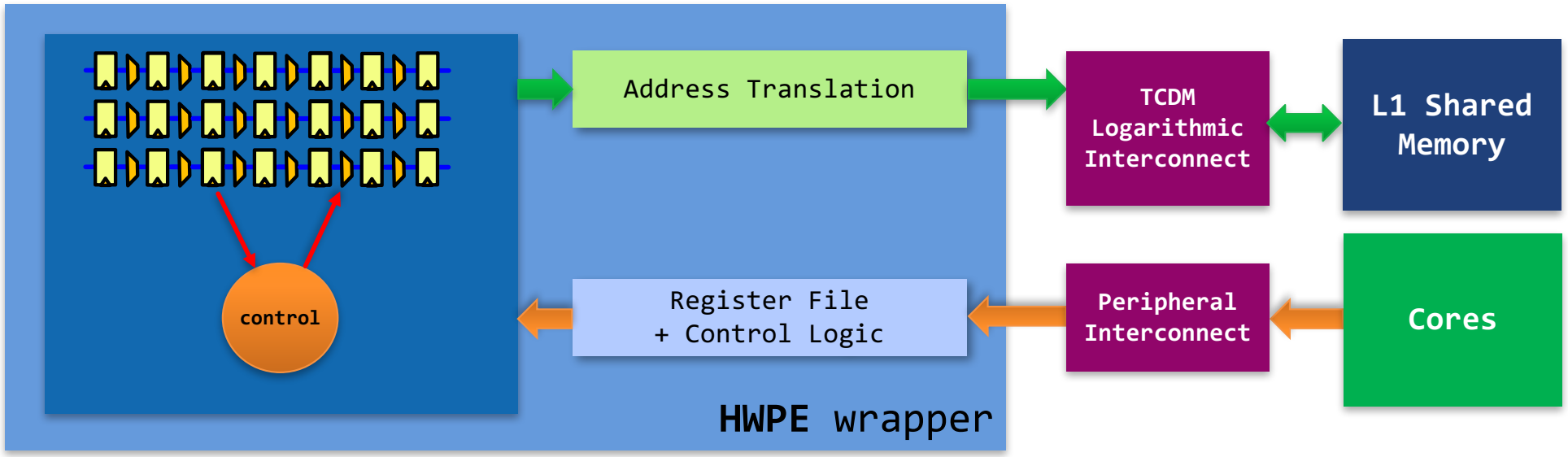
HW Processing Engines





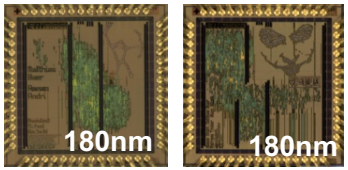
HW Processing Engines

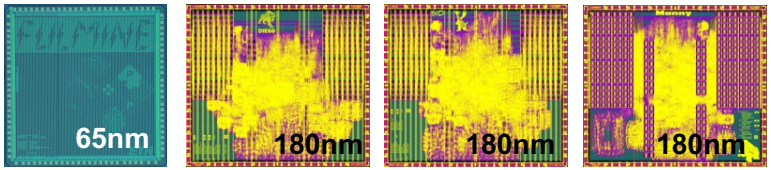


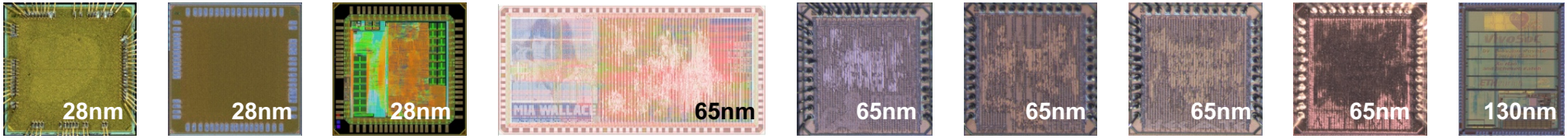


PULP: a busy silicon schedule 2013-2???

- ***ST 28nm FDSOI***
  - PULP1
  - PULP2
  - PULP3 **(on board)**
- ***UMC 65nm***
  - Artemis, Hecate, Selene, Diana - FPU
  - Mia Wallace – full system **(on board)**
  - Imperio - PULPino chip **(on board)**
  - Fulmine – secure smart analytics **(on board)**
  - Patronus – tiny cores **(taped out)**
- ***GF 28nm***
  - Honey Bunny – first RISC-V based **(on board)**
- ***GF 22nm***
  - Ariane – RISC-V 64bit core **(under development)**
  - Quentin – second-gen PULPino MCU **(under development)**
- ***UMC 180nm***
  - Sir10us
  - Or10n
- ***SMIC 130nm***
  - VivoSoC
  - VivoSoC2 **(on board)**
- ***ALP 180nm***
  - Diego
  - Manny
- ***TSMC 40nm***
  - Mr. Wolf **(taping out)**



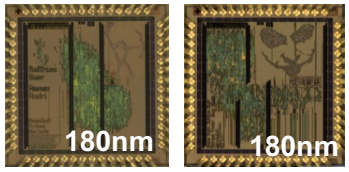


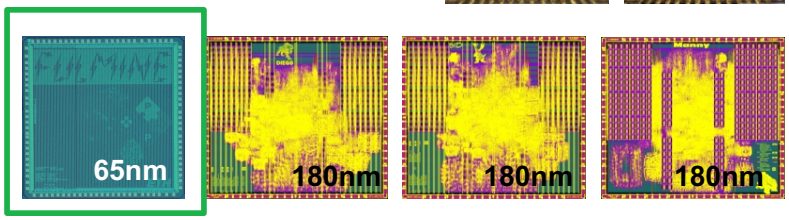


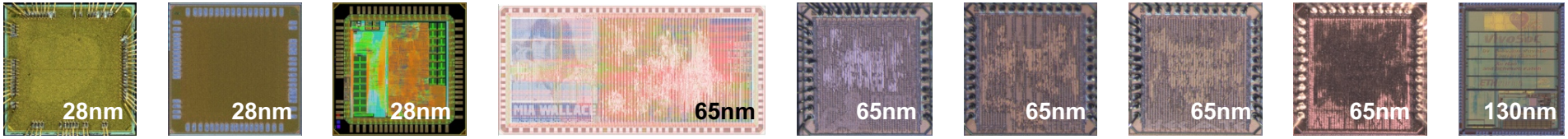
PULP

PULP: a busy silicon schedule 2013-2???

- ***ST 28nm FDSOI***
  - PULP1
  - PULP2
  - PULP3 **(on board)**
- ***UMC 65nm***
  - Artemis, Hecate, Selene, Diana - FPU
  - Mia Wallace – full system **(on board)**
  - Imperio - PULPino chip **(on board)**
  - Fulmine – secure smart analytics **(on board)**
  - Patronus – tiny cores **(taped out)**
- ***GF 28nm***
  - Honey Bunny – first RISC-V based **(on board)**
- ***GF 22nm***
  - Ariane – RISC-V 64bit core **(under development)**
  - Quentin – second-gen PULPino MCU **(under development)**
- ***UMC 180nm***
  - Sir10us
  - Or10n
- ***SMIC 130nm***
  - VivoSoC
  - VivoSoC2 **(on board)**
- ***ALP 180nm***
  - Diego
  - Manny
- ***TSMC 40nm***
  - Mr. Wolf **(taping out)**

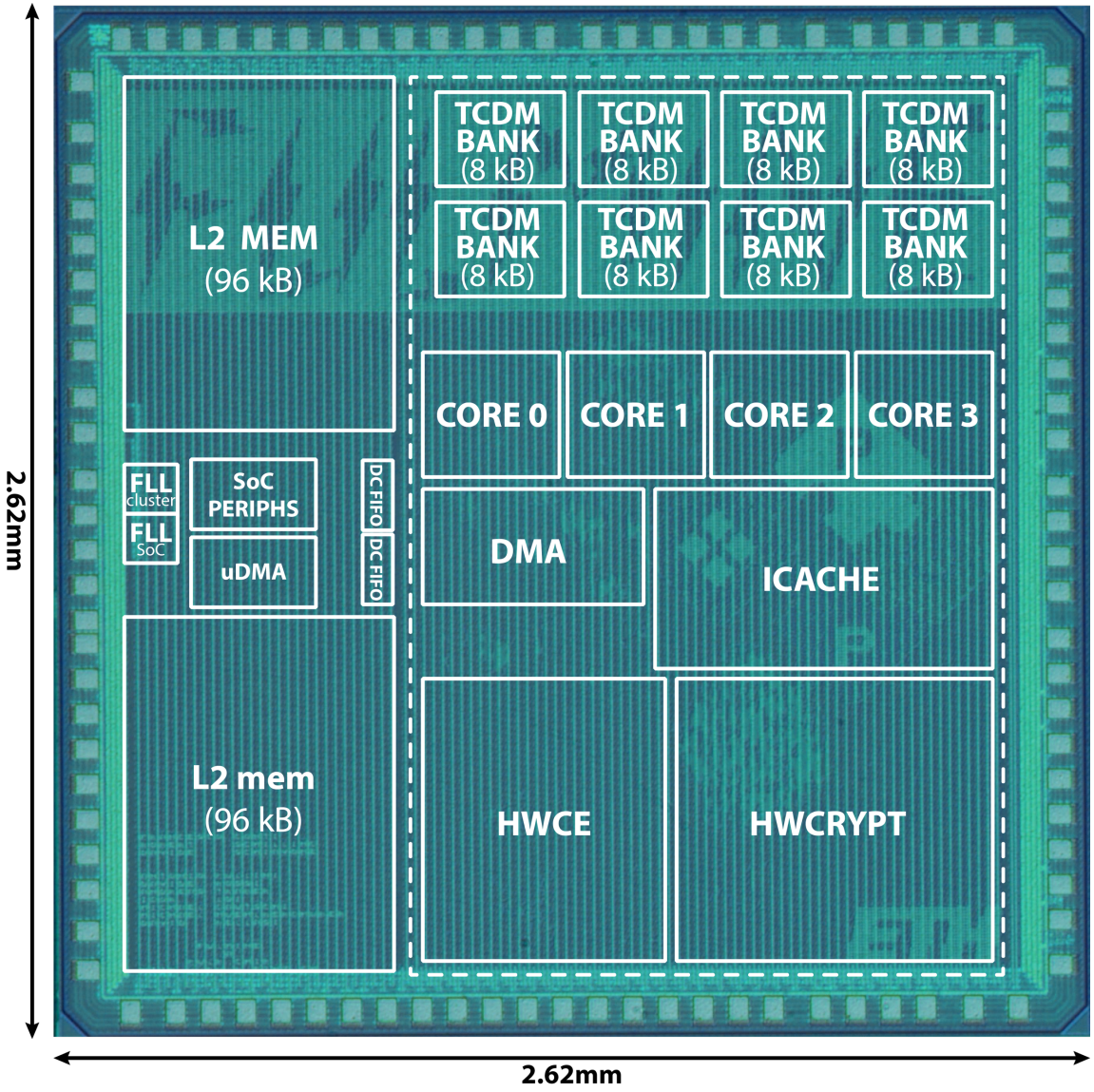






PULP

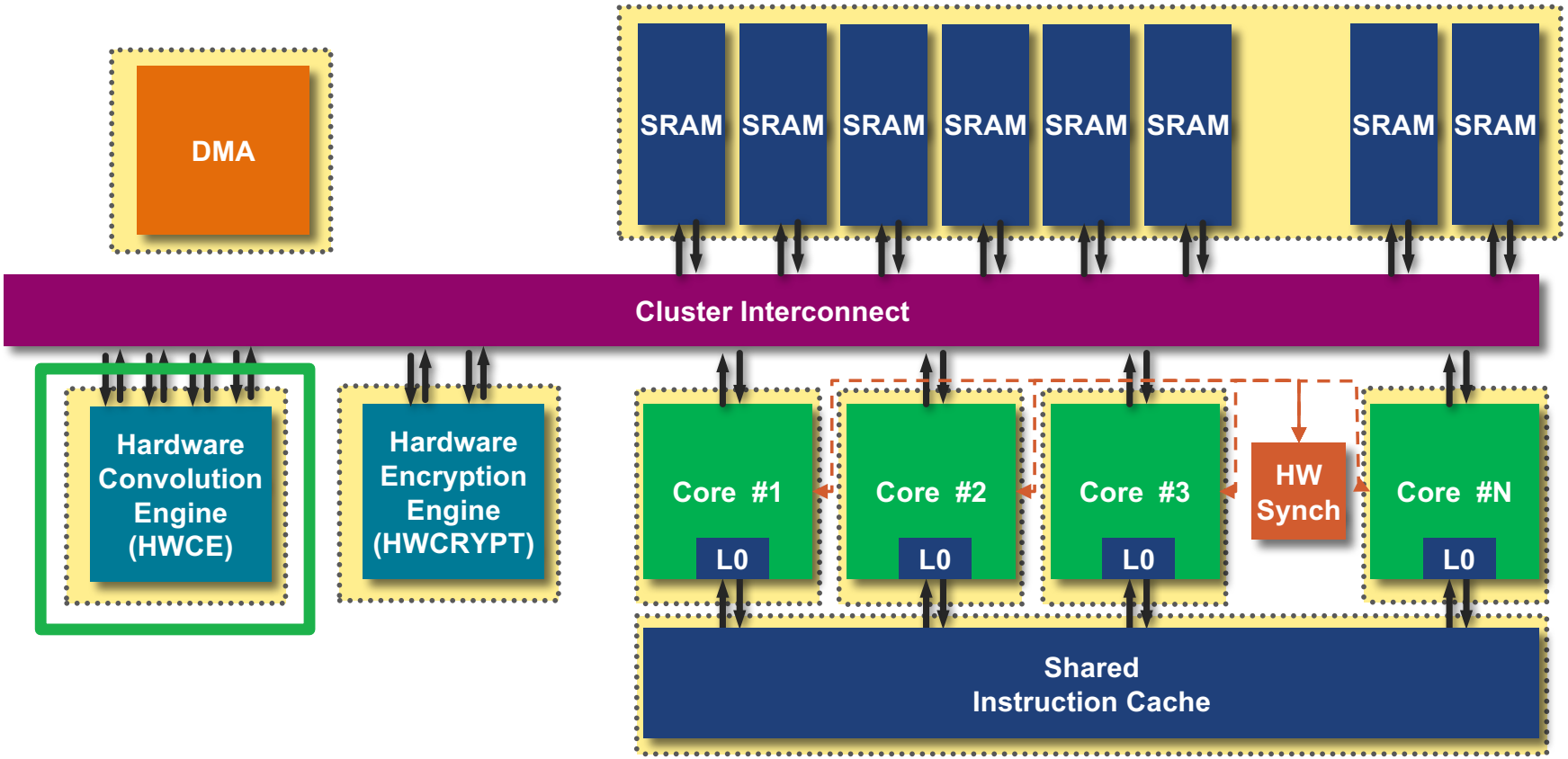
The *Fulmine* System-on-Chip



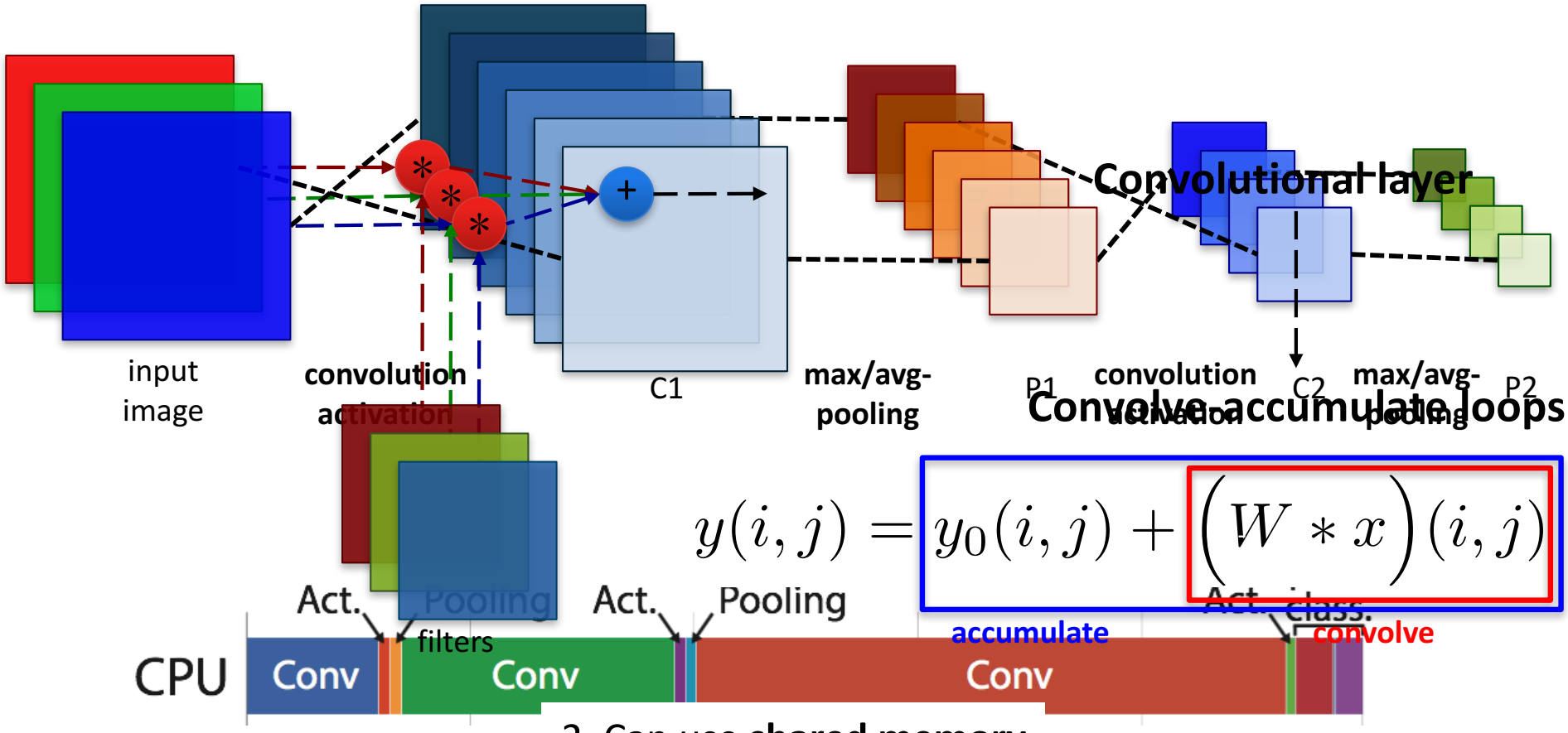
Fulmine SoC:

- UMC 65nm technology
  - 6.86 mm^2
- 4 cores, 2 accelerators
  - **HWCE** for 3D conv layers
  - **HWCRYPT** for AES
  - **DSP-optimized** cores
- 64 kB of L1, 192 kB of L2
- uDMA for I/O with no SW intervention
  - QSPI master/slave
  - I^2C
  - I^2S
  - UART

The *Fulmine* PULP cluster for Secure Smart Analytics



HWPEs for CNNs?

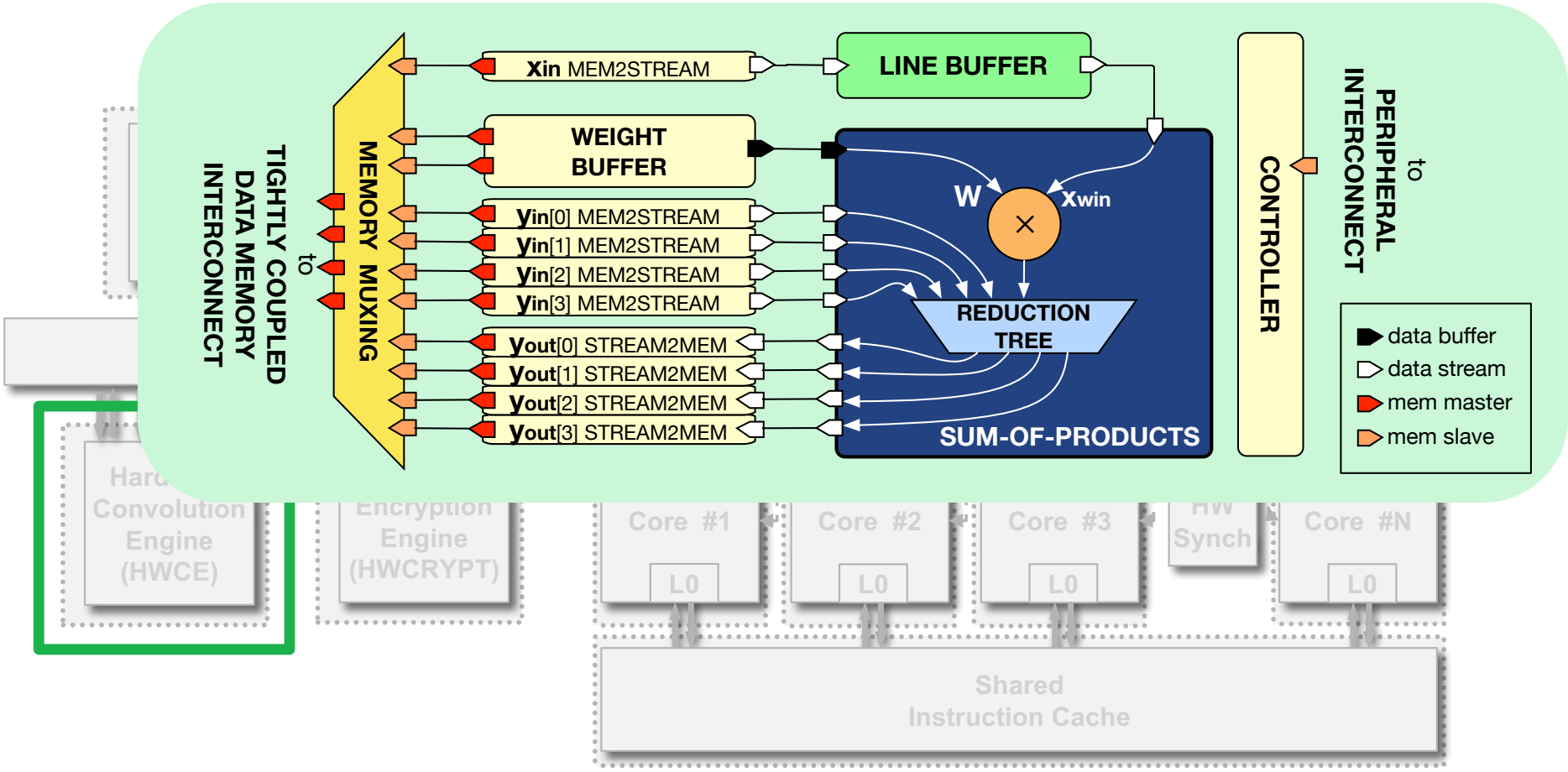


1. Suitable for **streaming** implementation

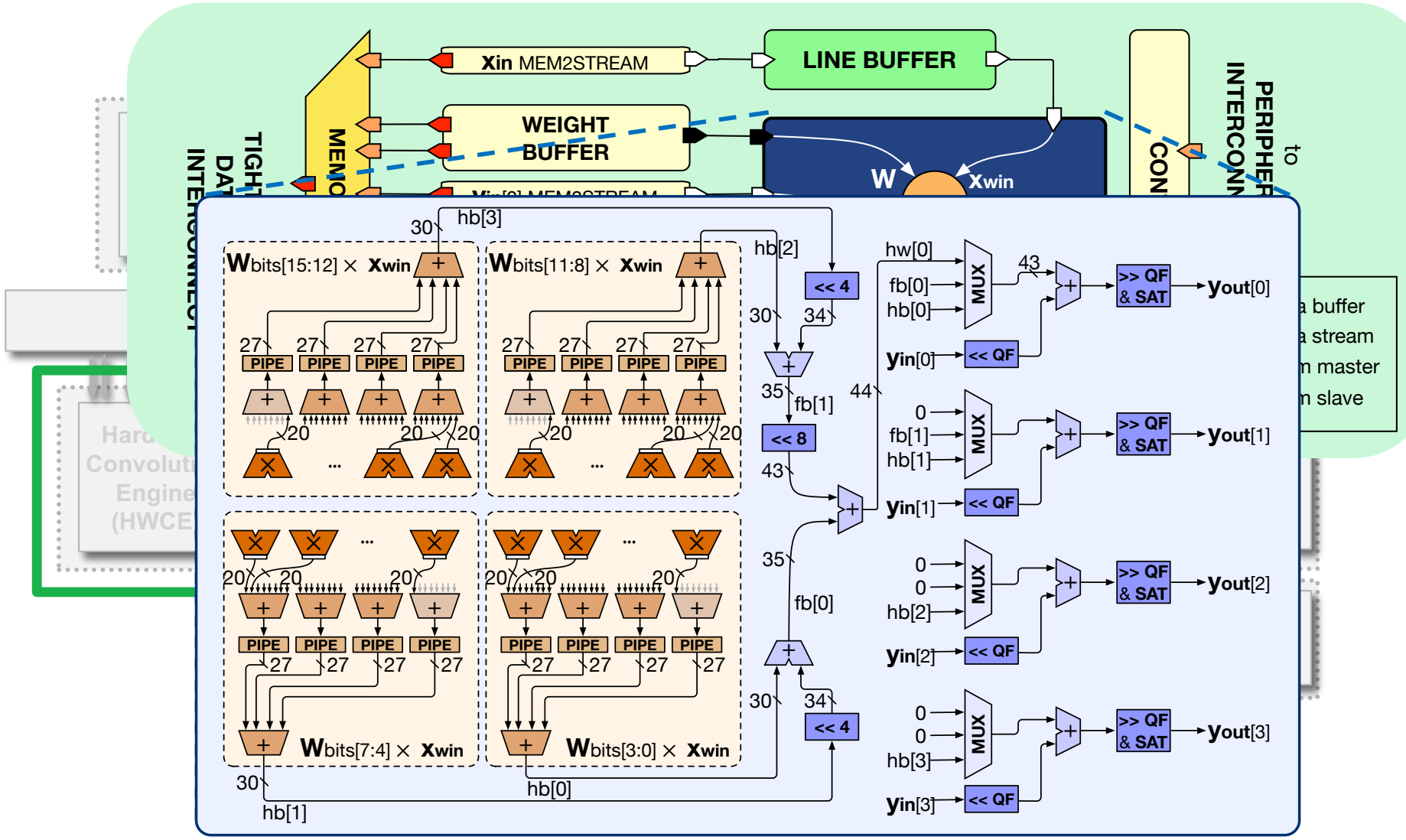
2. Can use **shared memory** for intermediate results (i.e. accumulation)

3. Target **one** case in **HW**, but manage **all** by **SW**

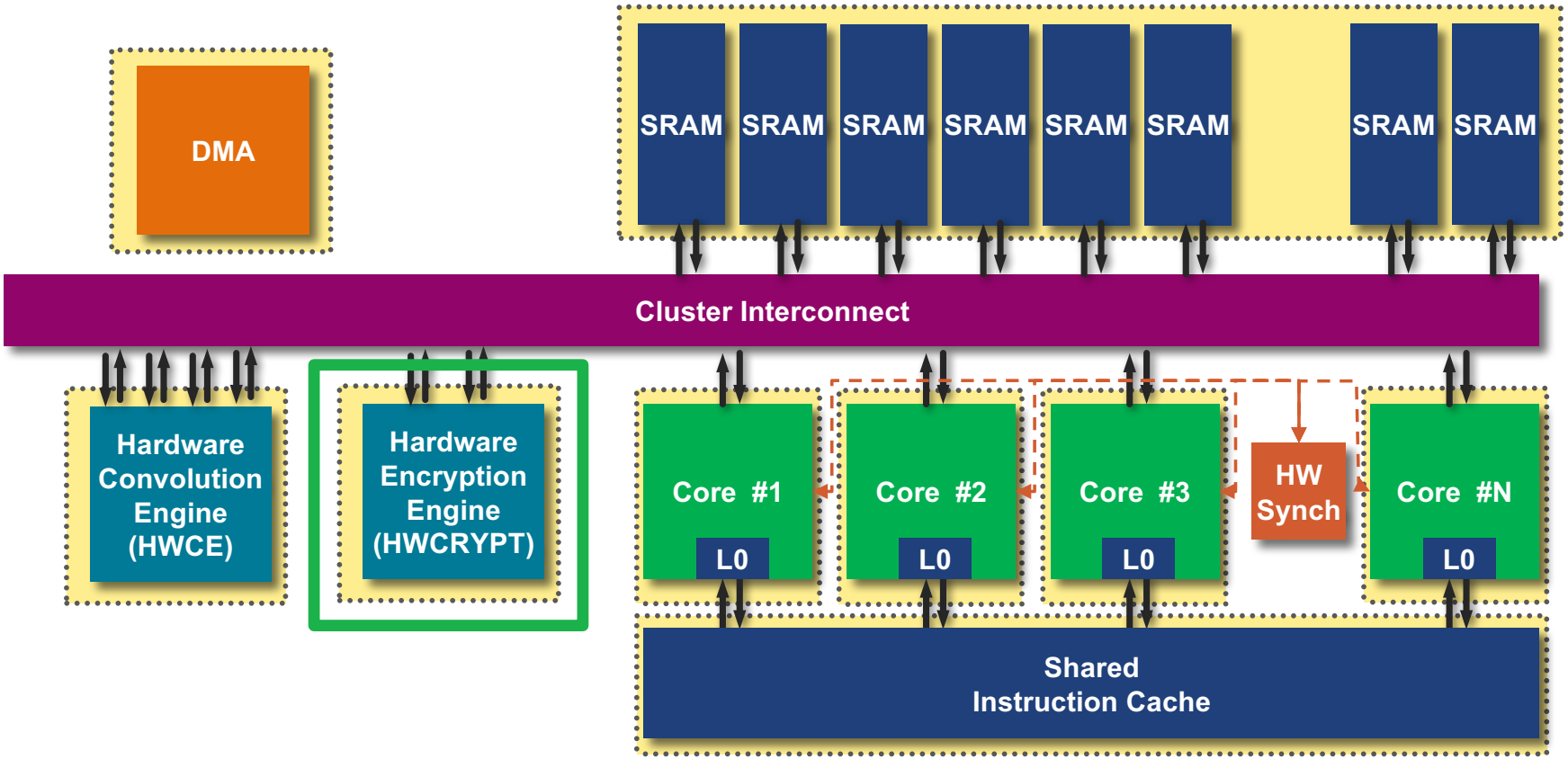
The *Fulmine* PULP cluster for Secure Smart Analytics



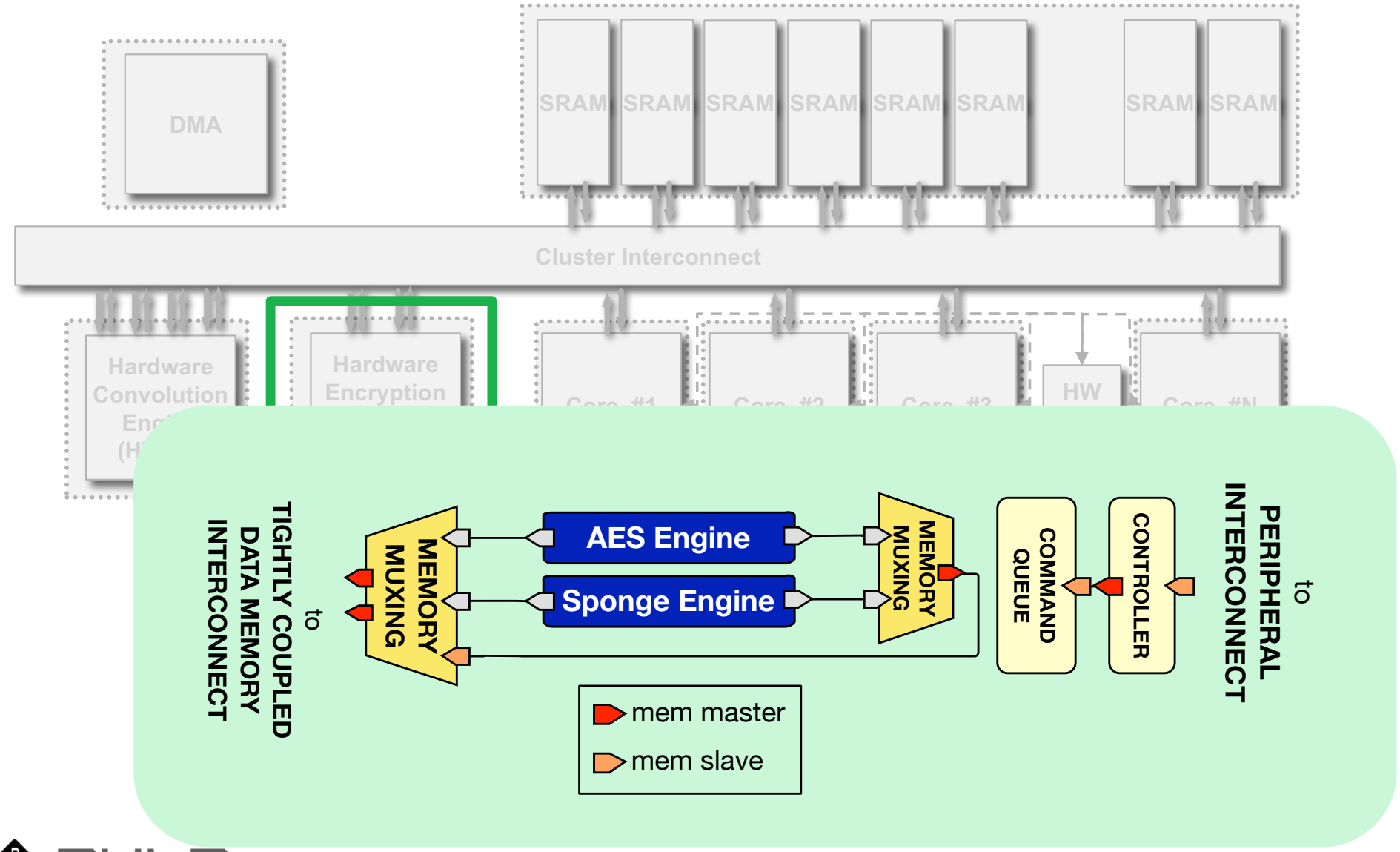
The *Fulmine* PULP cluster for Secure Smart Analytics



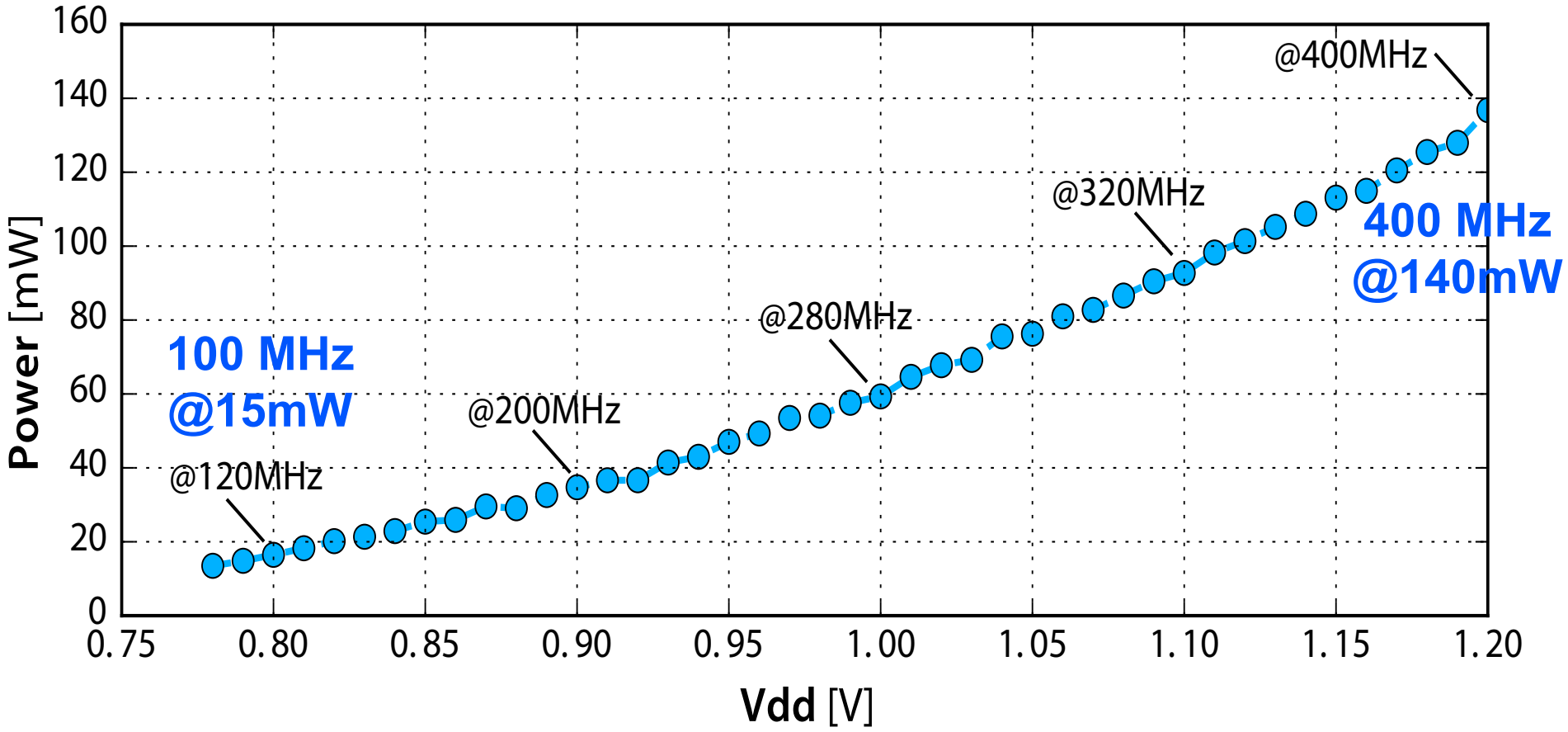
The *Fulmine* PULP cluster for Secure Smart Analytics



The *Fulmine* PULP cluster for Secure Smart Analytics



Fulmine SoC performance and power envelope

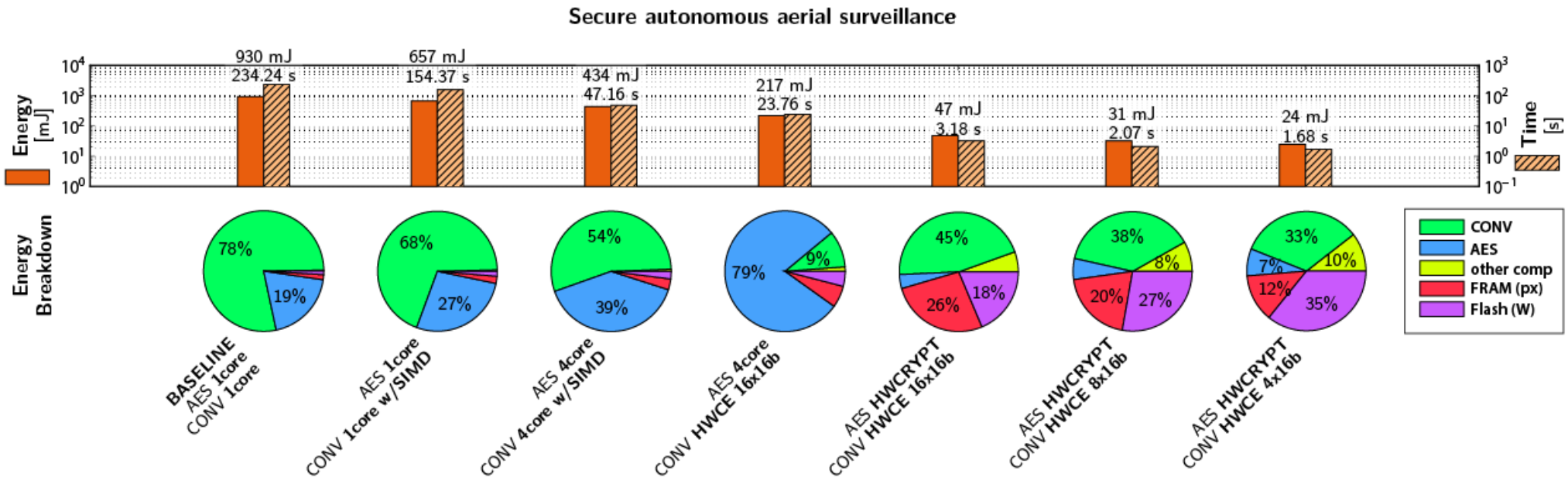


State-of-the-Art comparison

CRYPTOGRAPHY	OUR WORK	Zhang et al. [2] VLSI'16	Mathew et al. [1] JSSC'15 @ 0.9V	Mathew et al. [1] JSSC'15 @ 0.43V
Technology	UMC 65nm LL 1P8M	TSMC 40nm	Intel 22nm	Intel 22nm
Operating Point	0.8V, 84MHz	0.9V, 1.3 GHz	0.9V, 1.13 GHz	0.43V, 324 MHz
Area	5.75 mm^2 (SoC) 0.56 mm^2 (HWCRYPT)	0.42 mm^2 (AES)	0.19 mm^2 (AES)	0.19 mm^2 (AES)
Power	27 mW (SoC)	4.39 mW (AES)	13 mW (AES)	428 µW (AES)
Performance	1.76 Gbit/s	0.446 Gbit/s	0.432 Gbit/s	0.124 Gbit/s
Energy Efficiency	65.2 Gbit/s/W	13 Gbit/s/W	33.2 Gbit/s/W	289 Gbit/s/W
Supported Schemes	AES-XTS, AES-ECB, Keccak-f400, LR masking/shuffling	AES-ECB	AES-ECB	AES-ECB

CNN	OUR WORK	Eyeriss [3] ISSCC'16	Sim et al. [4] ISSCC'16
Technology	UMC 65nm LL 1P8M	TSMC 65nm LP 1P9M	65nm 1P8M
Operating Point	0.8V, 84MHz	1V, 200MHz	1.2V, 128MHz
Area	5.75 mm^2 (SoC) 0.35 mm^2 (HWCE)	12.25 mm^2	16 mm^2
Power	14 mW (SoC)	288 mW	45 mW
Performance	1.85/3.44/4.64 GMac/s	21.4 GMac/s	32 GMac/s
Energy Efficiency	132/246/331 GMac/s/W	74.3 GMac/s/w	710 GMac/s/W
Arithmetic Precision	Fixed point 16/8/4x16 bits	Fixed point 16x16 bits	Fixed point 16x16/24x24 bits

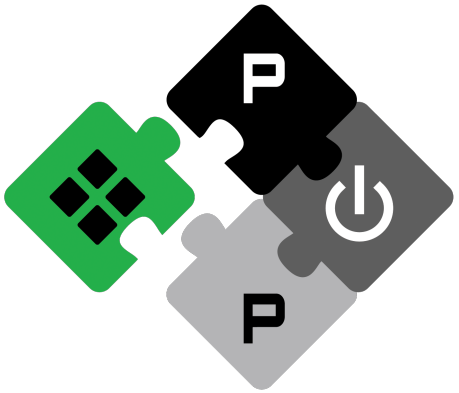
Example application: secure aerial surveillance



Fulmine silicon measurements for CONV, AES, DMA + datasheet values for COTS FRAM, Flash

- **ResNet-based CNN** secured at the cluster boundary with **AES** encryption
- An example application for a smart endnode mounting a Fulmine chip…

Thanks for your attention…



http://www.pulp-platform.org

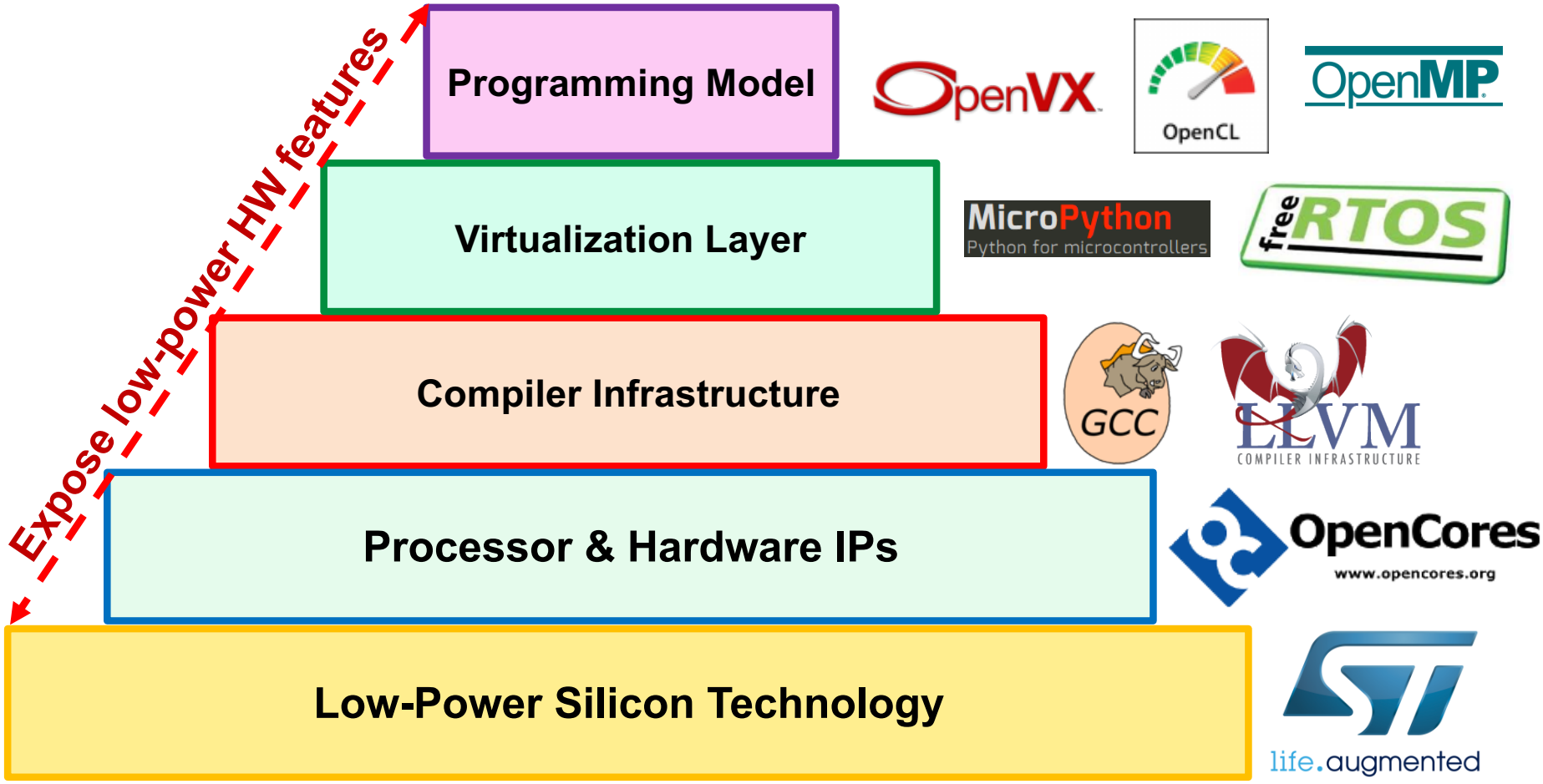
GitHub: pulp-platform

pulp-info@list.ee.ethz.ch

BACKUP SLIDES

PULP: pJ/op Parallel ULP computing

pJ/op is traditionally the target of ASIC + super-small research µControllers

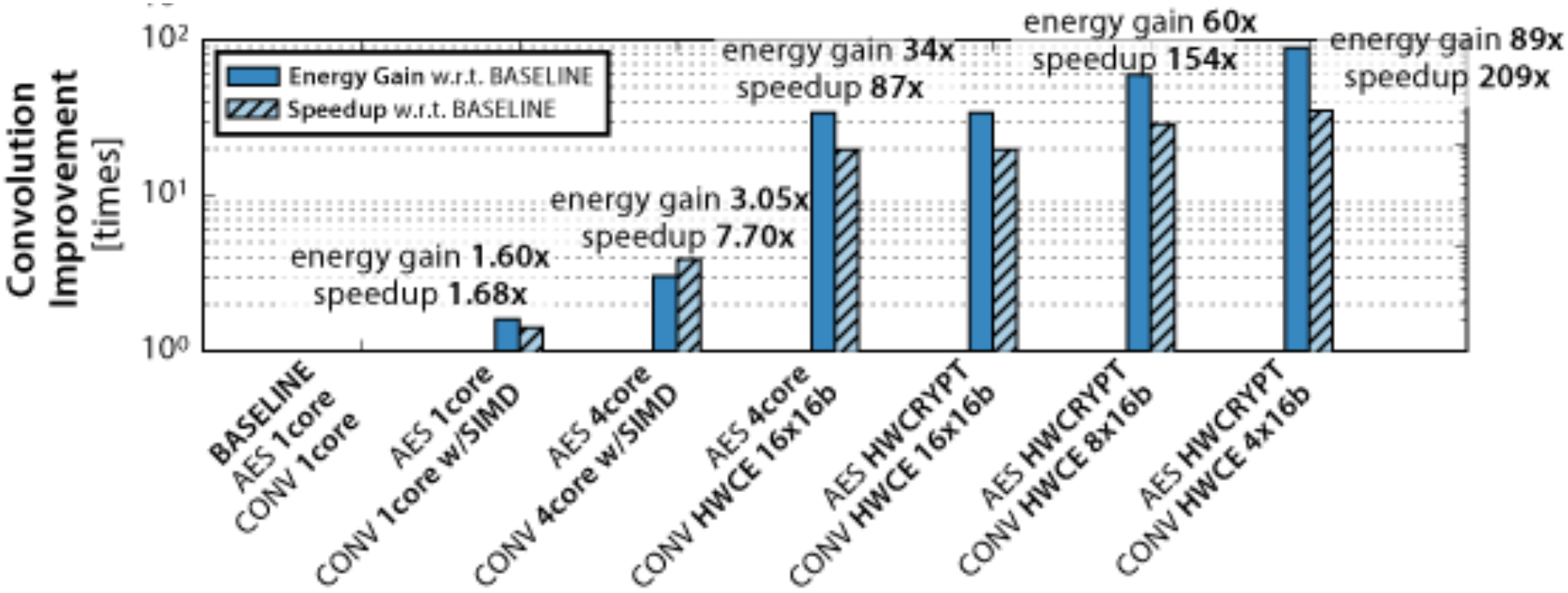


Parallel + Programmable + Heterogeneous ULP computing

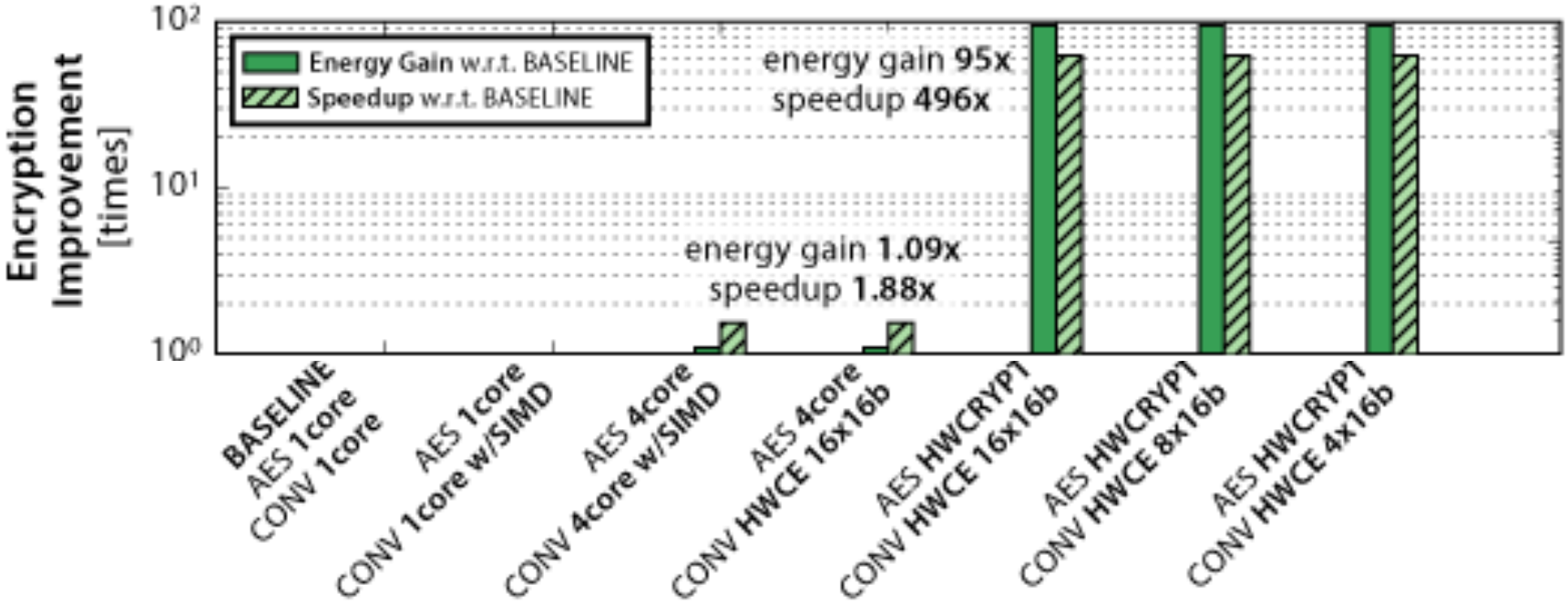
1mW-10mW active power

PULP

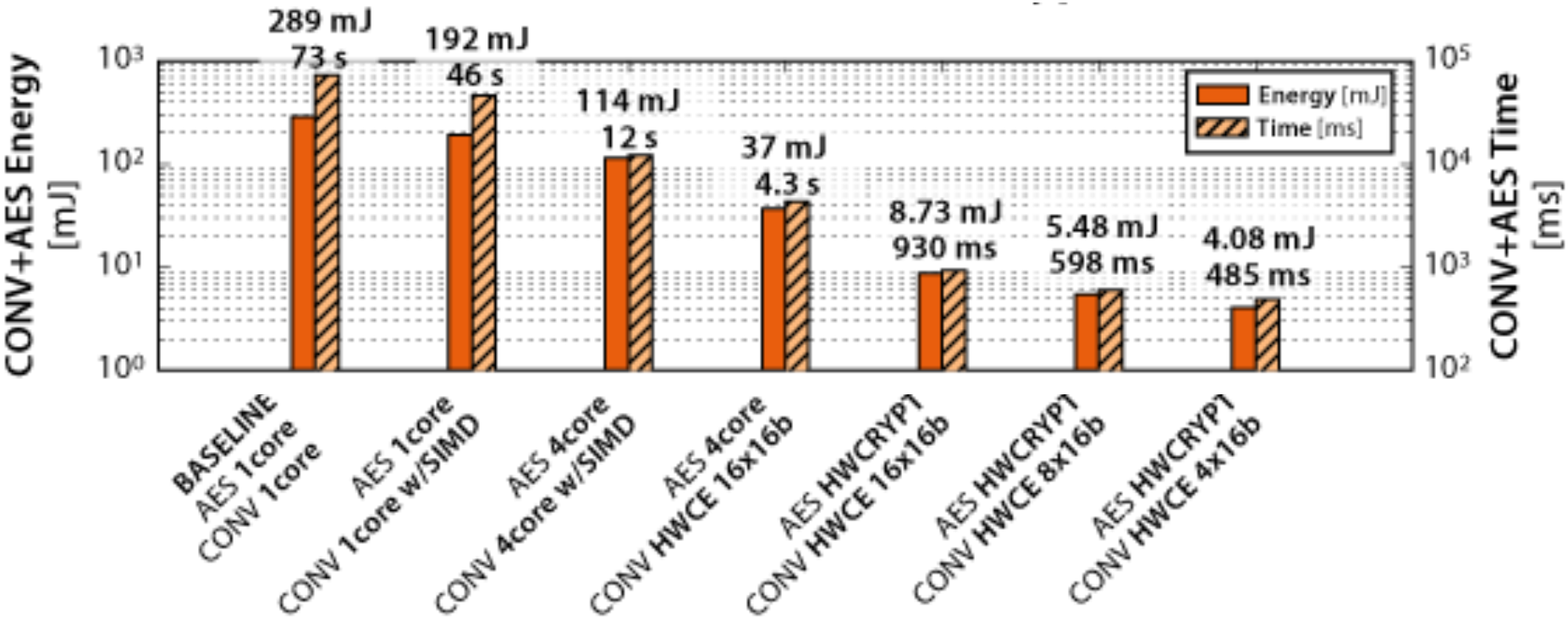
An example: Secured AlexNet



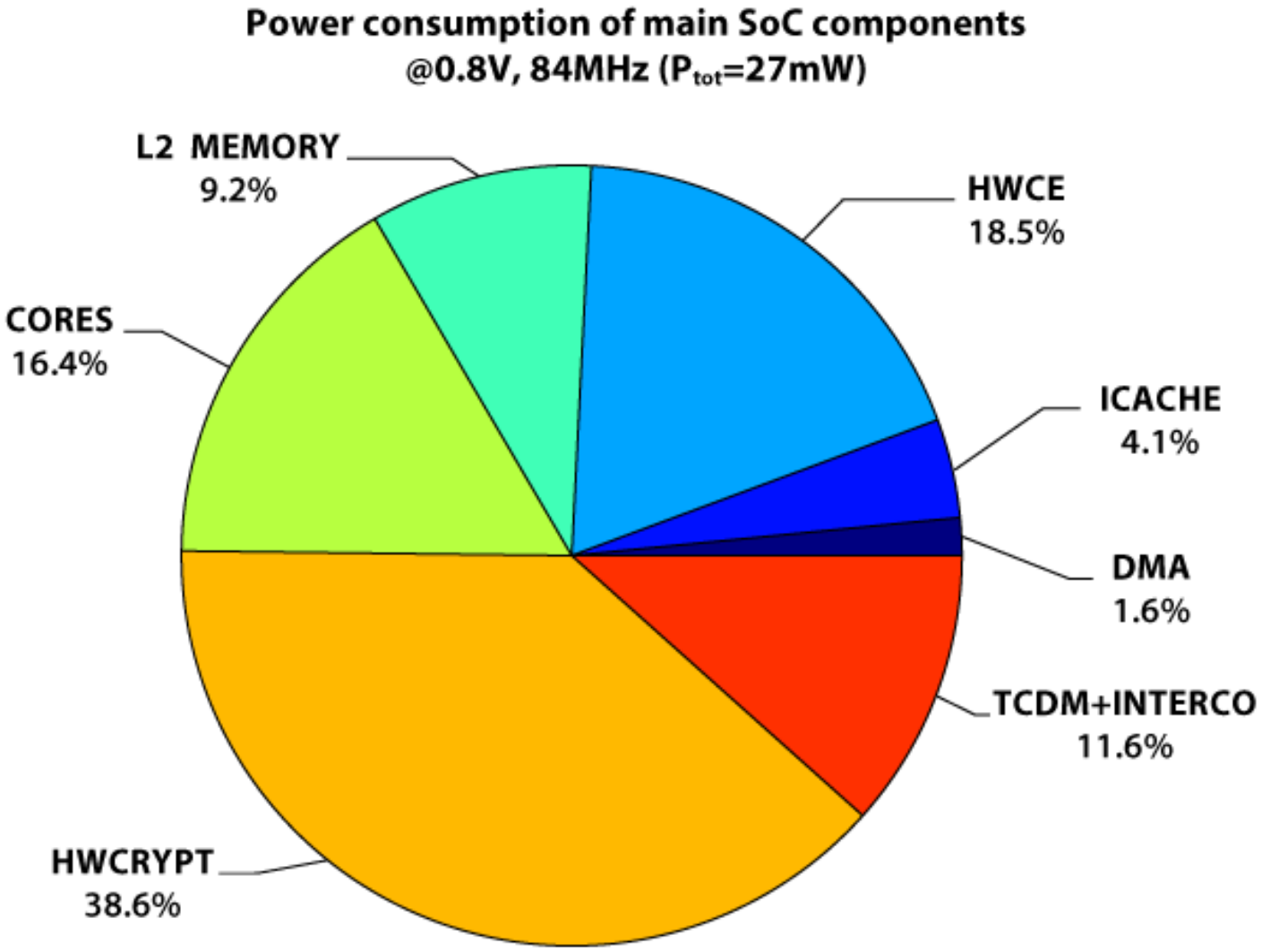
An example: Secured AlexNet



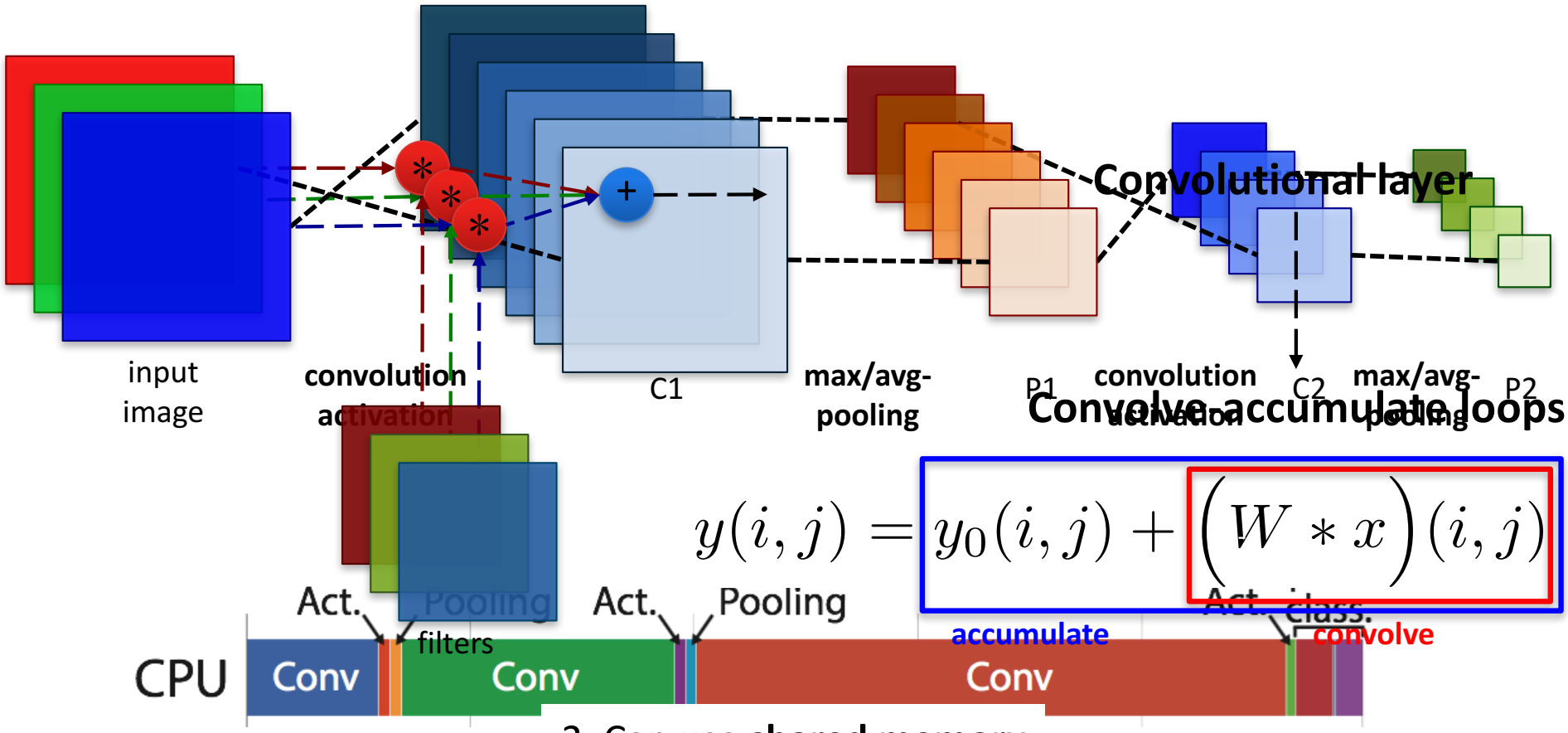
An example: Secured AlexNet



Power Breakdown



Accelerating CNNs



1. Suitable for **streaming** implementation

2. Can use **shared memory** for intermediate results (i.e. accumulation)

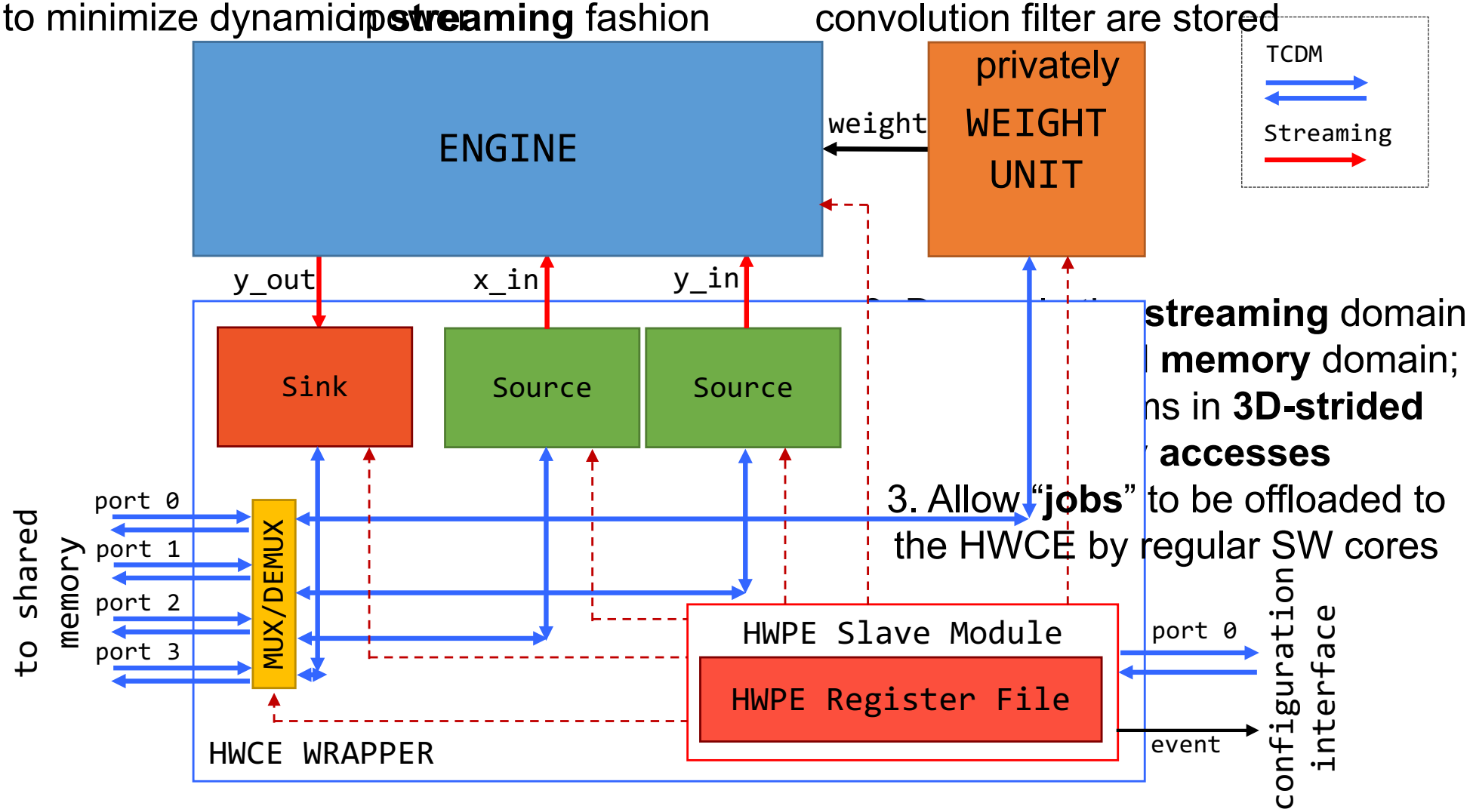
3. Target **one** case in **HW**, but manage **all** by **SW**

Hardware Convolution Engine

5. Fine-grain **clock gating** to minimize dynamic power

1. Perform **convolve-accumulate** in **streaming** fashion

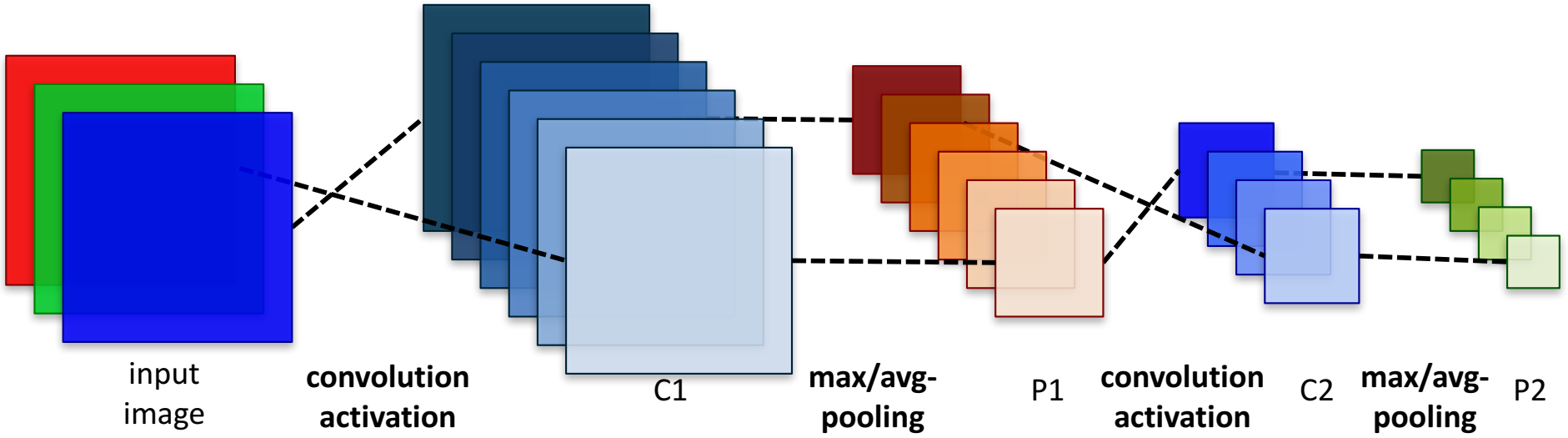
4. **Weights** for each convolution filter are stored privately



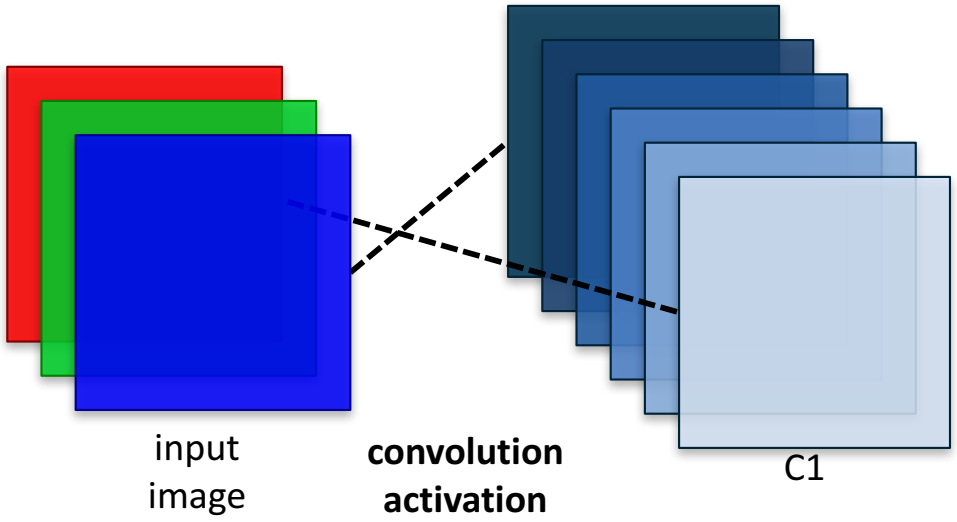
streaming domain
memory domain;
ns in **3D-strided**
accesses

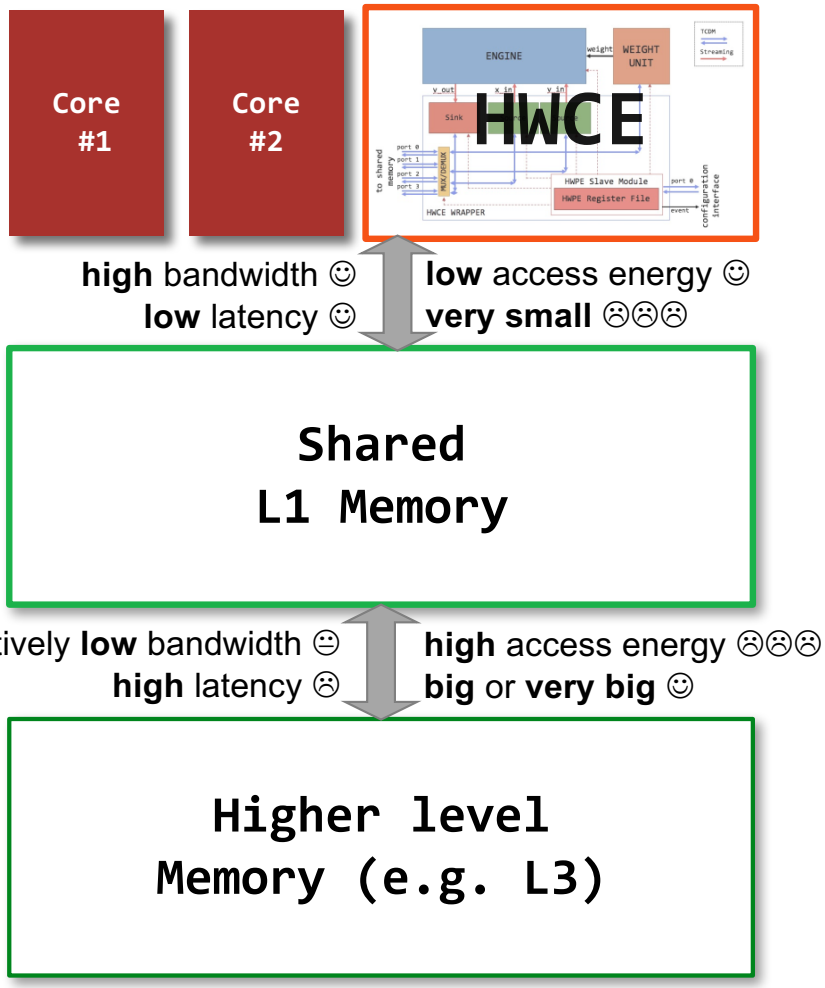
3. Allow "**jobs**" to be offloaded to the HWCE by regular SW cores

But how to map full CNNs on PULP?



But how to map full CNNs on PULP?

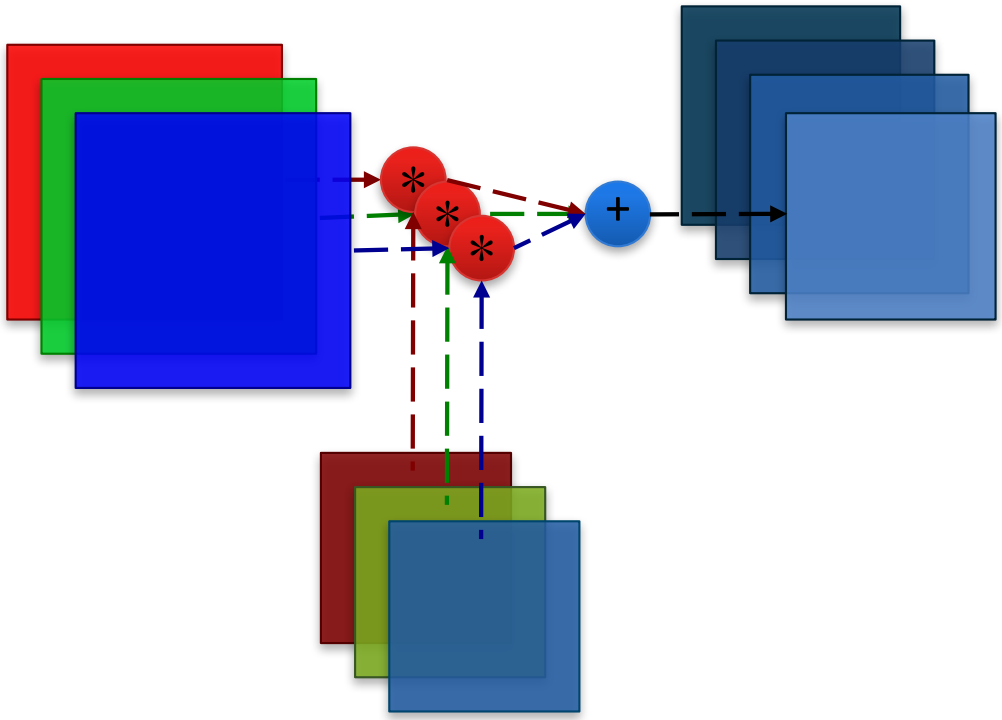


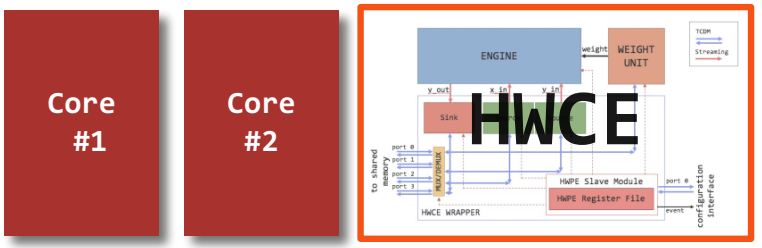


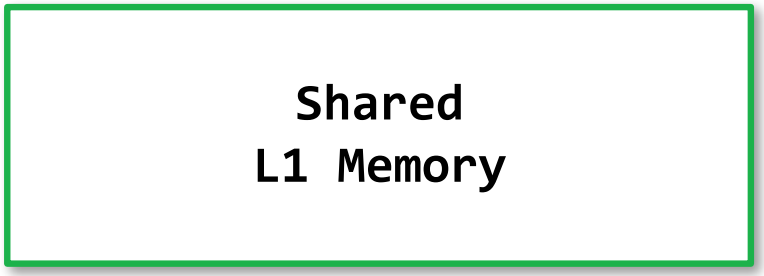
Essentially, a problem of optimizing **data exchange**

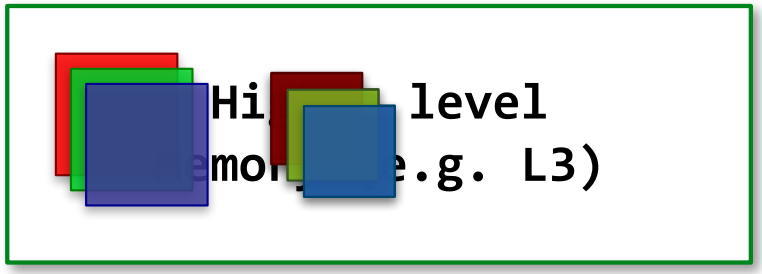
1. Maximize **data reuse**
2. Avoid **unneeded transfers** back and forth

Mapping CNNs on PULP



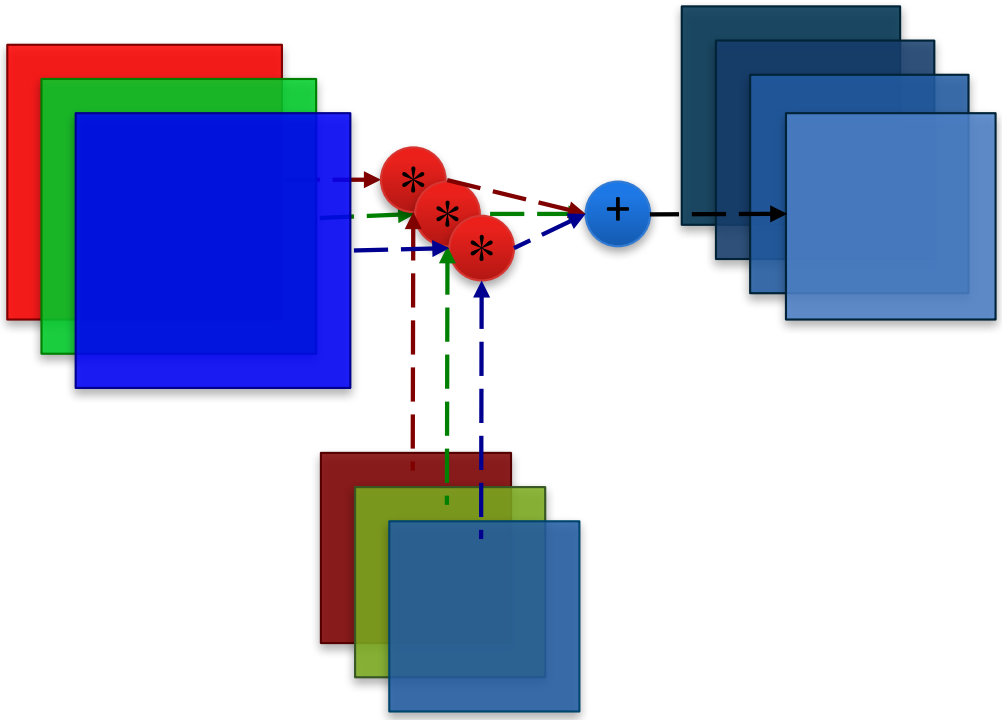


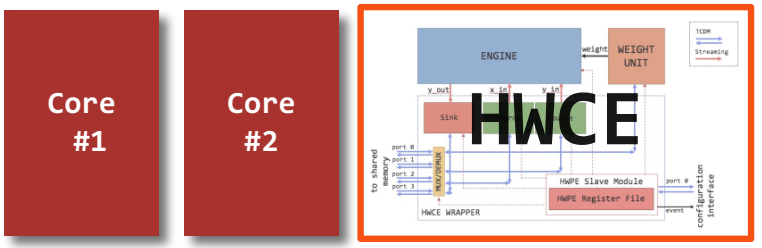


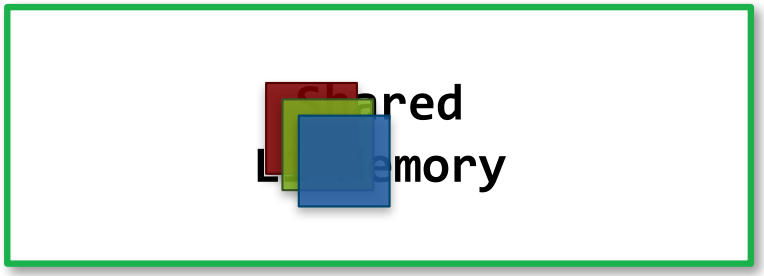


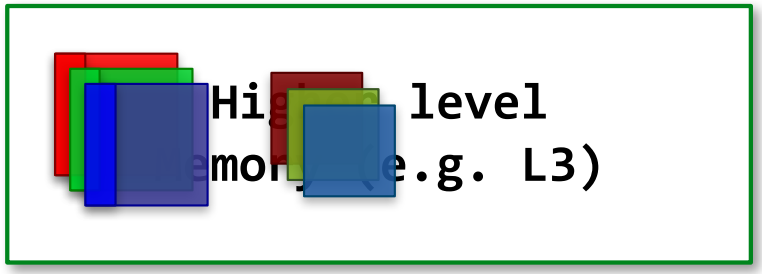
1. Copy **all weights** for current layer from L3→L1

Mapping CNNs on PULP



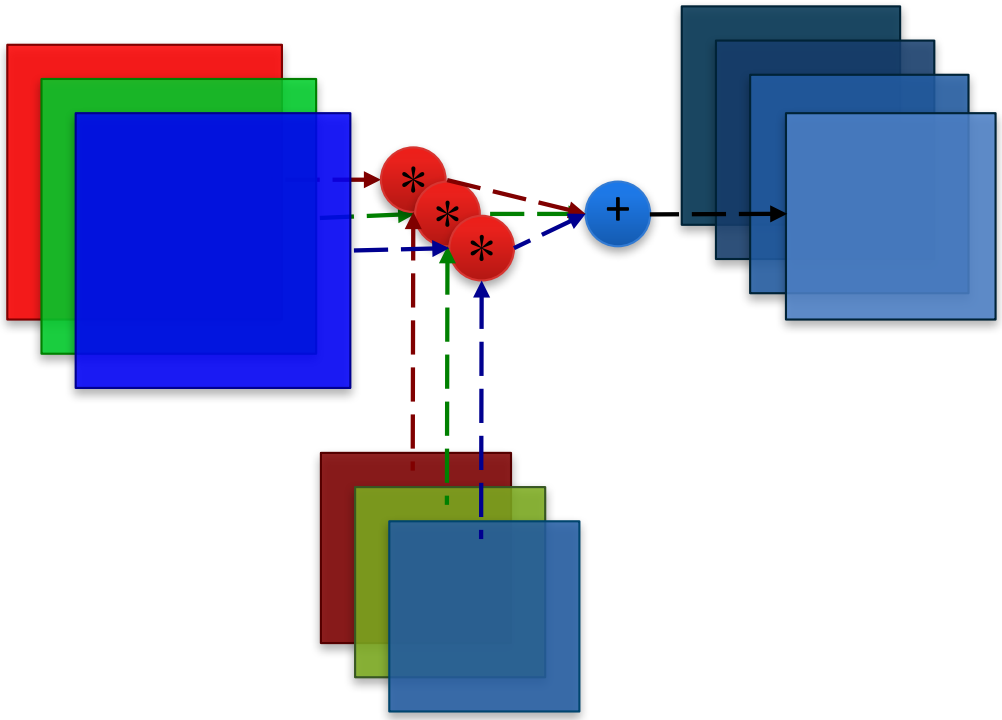


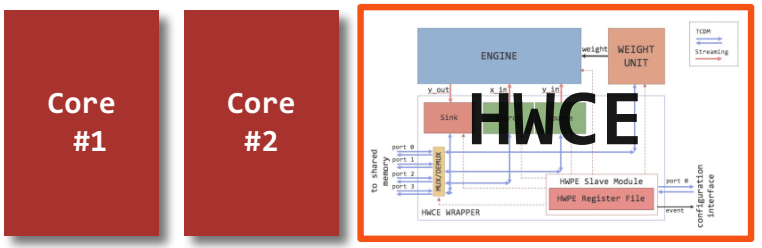


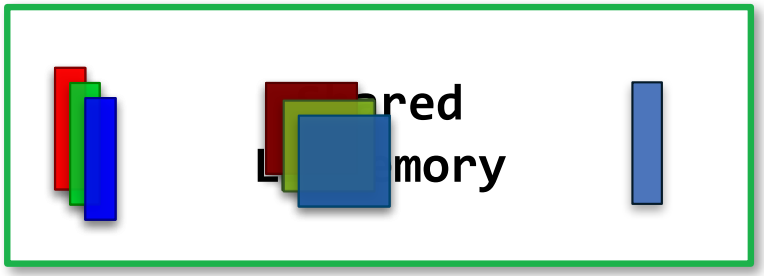


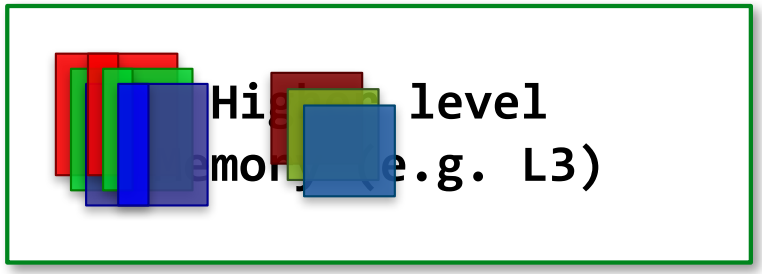
2. Copy input **tile 0** from L3→L1
(stripe of *N* input features)

Mapping CNNs on PULP



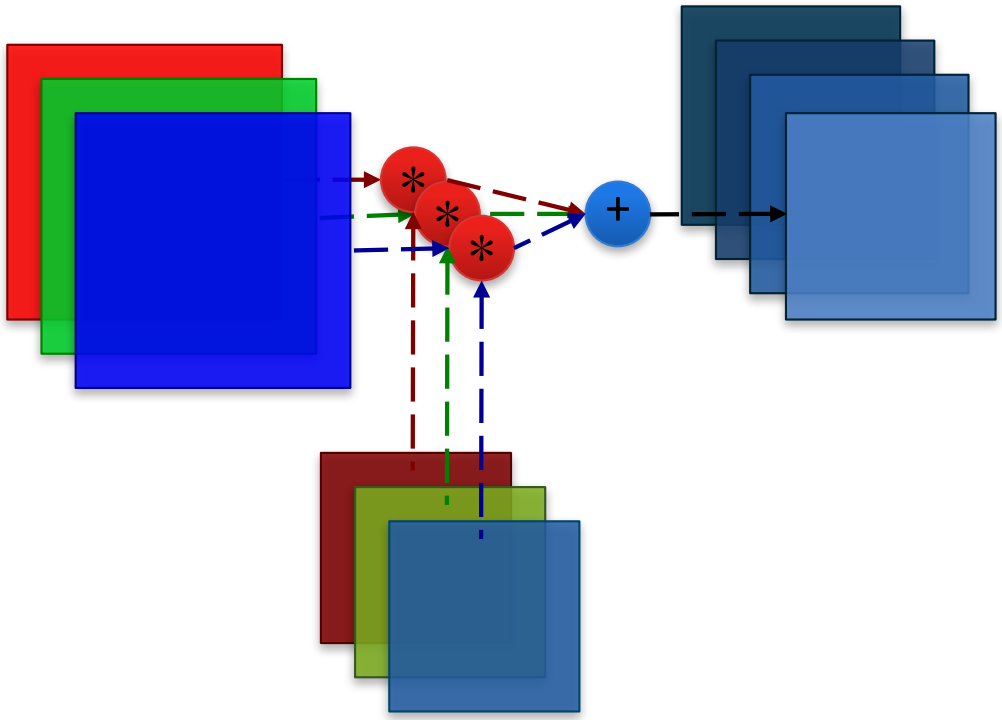


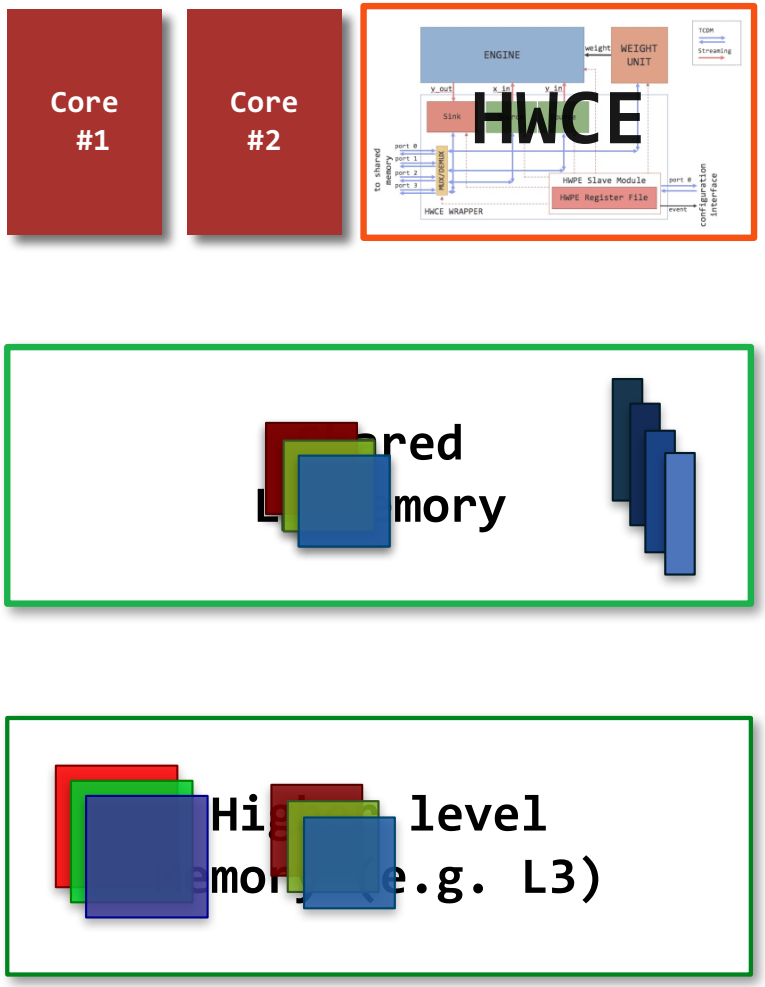




3. Copy input **tile 1** from L3→L1
+ **compute** on tile 0 (double buffering)

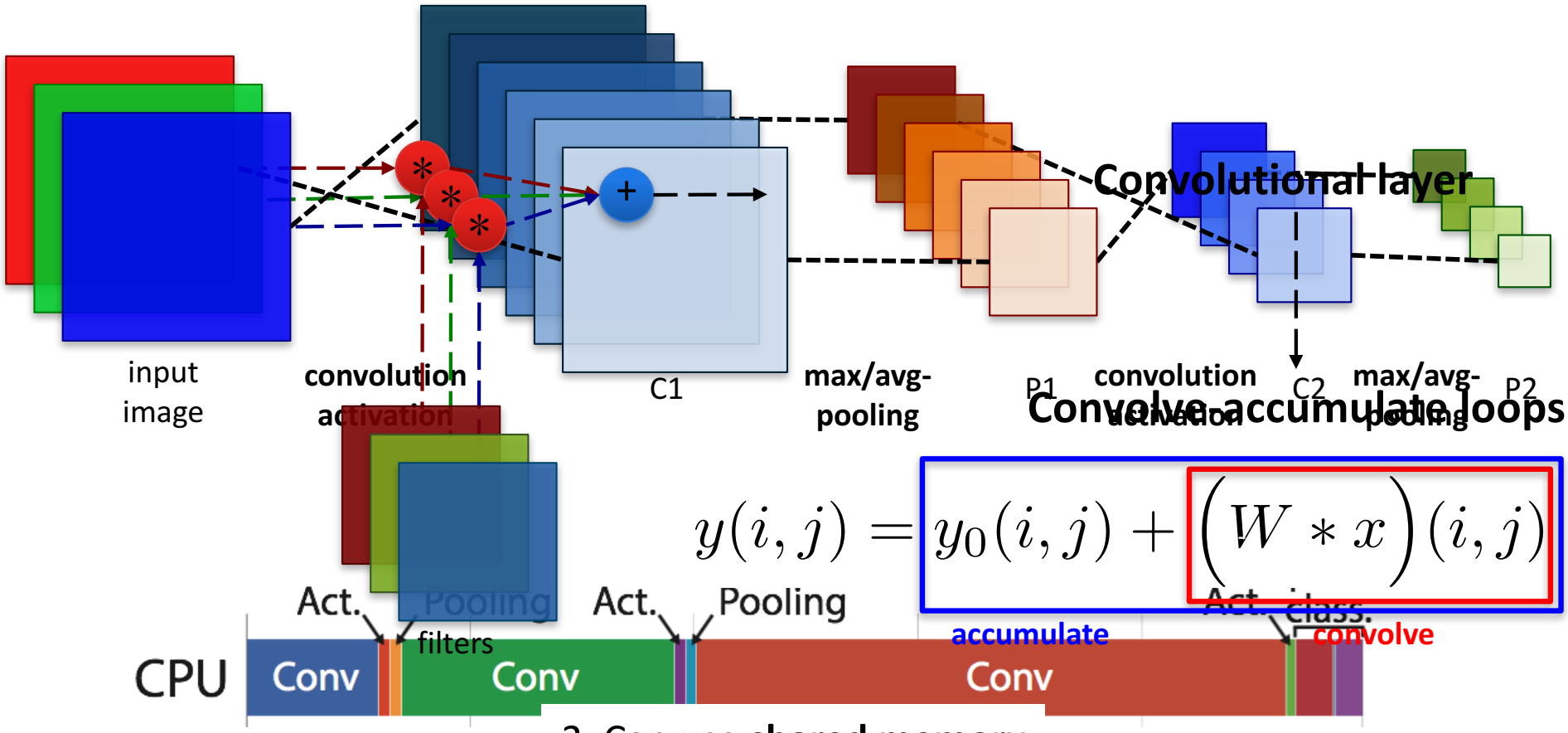
Mapping CNNs on PULP





3. When computation on a tile is **complete** for the given layer, write it back to L3

Accelerating CNNs

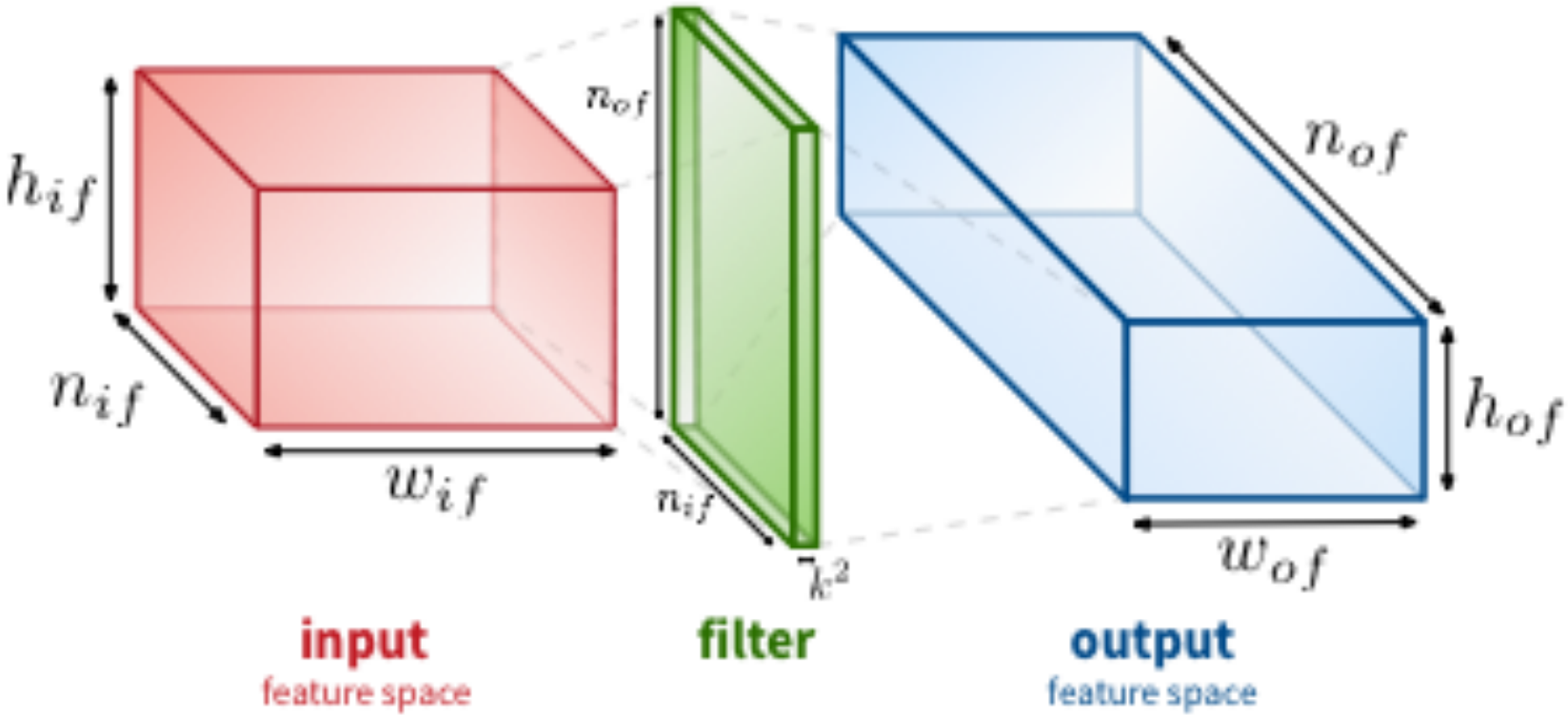


1. Suitable for **streaming** implementation

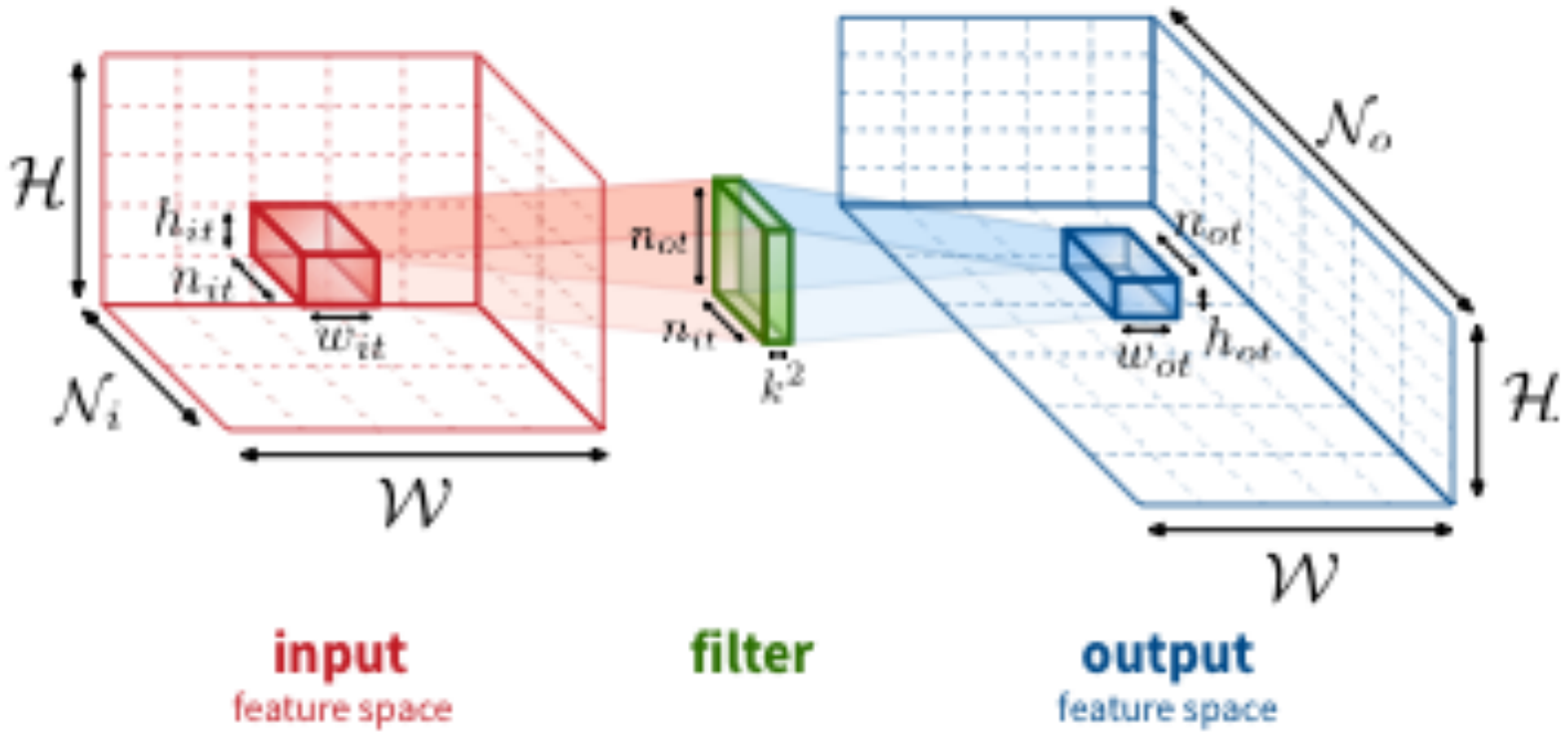
2. Can use **shared memory** for intermediate results (i.e. accumulation)

3. Target **one** case in **HW**, but manage **all** by **SW**

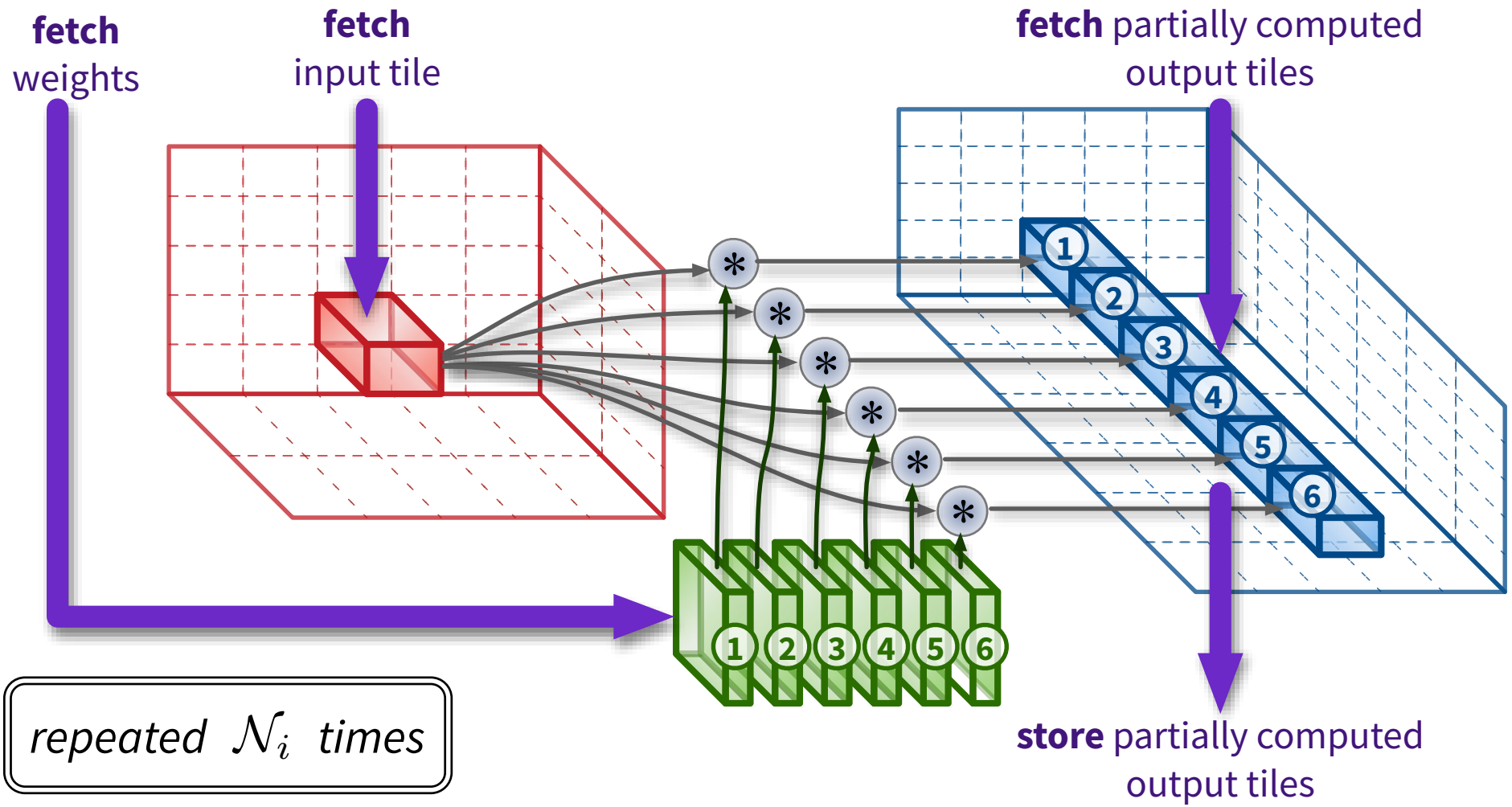
Tiling


h_{if}
n_{if}
w_{if}
n_{of}
n_{if}
k^2
n_{of}
h_{of}
w_{of}
input
feature space
filter
output
feature space

Tiling


$\mathcal{H}$
h_{it}
n_{it}
w_{it}
$\mathcal{N}_i$
$\mathcal{W}$
n_{ot}
n_{it}
k^2
$\mathcal{N}_o$
n_{ot}
w_{ot}
h_{ot}
$\mathcal{H}$
$\mathcal{W}$
input
feature space
filter
output
feature space

Tiling on Input Features



Tiling on Output Features