PEZY-SC4s

The Fourth Generation MIMD Many-core Processor with High Energy Efficiency and Flexibility for HPC and AI Applications

Naoya Hatta†, Shuntaro Tsunoda, Kouhei Uchida,

Taichi Ishitani, Toru Koizumi, Ryota Shioya, Kei Ishii

† PEZY Computing, K.K.

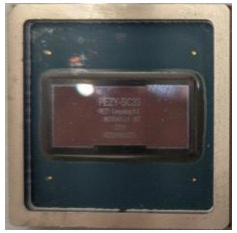
PEZY Computing, K.K.

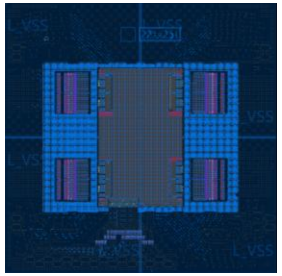
Established
- 2010

Business
- Develop supercomputer system
  - Microprocessors and electronic devices
  - Immersion cooling systems
  - Genome analysis and medical imaging software

Location
- Tokyo, Japan

History of PEZY-SCx



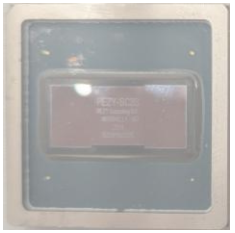


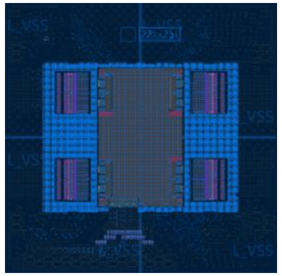
Package Layout

	PEZY-SC	PEZY-SC2	PEZY-SC3	PEZY-SC3s	PEZY-SC4s
Release	2014	2016	2020	2021	2026
Process	28 nm	16 nm	7 nm	7 nm	5 nm
Core	1024	2048	4096	512	2048
Performance	0.75 TFLOPS	4.1 TFLOPS	19.7 TFLOPS	2.0 TFLOPS	24.6 TFLOPS
Mem Bandwidth	154 GB/s	102 GB/s	1228 GB/s	614 GB/s	3277 GB/s
PCIe	Gen3 x 32	Gen4 x 32	Gen4 x 48	Gen4 x 4	Gen5 x 16
The Green500	Ranked **1st** (2015-2016)	Ranked **1st** (2017-2018)	Ranked 12th (2021)		

* Double Precision

History of PEZY-SCx





Package Layout

	PEZY-SC	PEZY-SC2	PEZY-SC3	PEZY-SC3s	PEZY-SC4s
Release	2014	2016	2020	2021	2026
Process	28 nm	16 nm	7 nm	7 nm	5 nm
Core	1024	2048	4096	512	2048
Performance	0.75 TFLOPS	4.1 TFLOPS	19.7 TFLOPS	2.0 TFLOPS	24.6 TFLOPS
Mem Bandwidth	154 GB/s	102 GB/s	1228 GB/s	614 GB/s	3277 GB/s
PCIe	Gen3 x 32	Gen4 x 32	Gen4 x 48	Gen4 x 4	Gen5 x 16
THE GREEN 500	Ranked **1st** (2015-2016)	Ranked **1st** (2017-2018)	Ranked 12th (2021)		

* Double Precision

Agenda

Architecture of PEZY-SCx Series

Implementation of PEZY-SC4s

Software

Evaluation of PEZY-SC4s

Summary and Future plans

Agenda

Architecture of PEZY-SCx Series

Implementation of PEZY-SC4s

Software

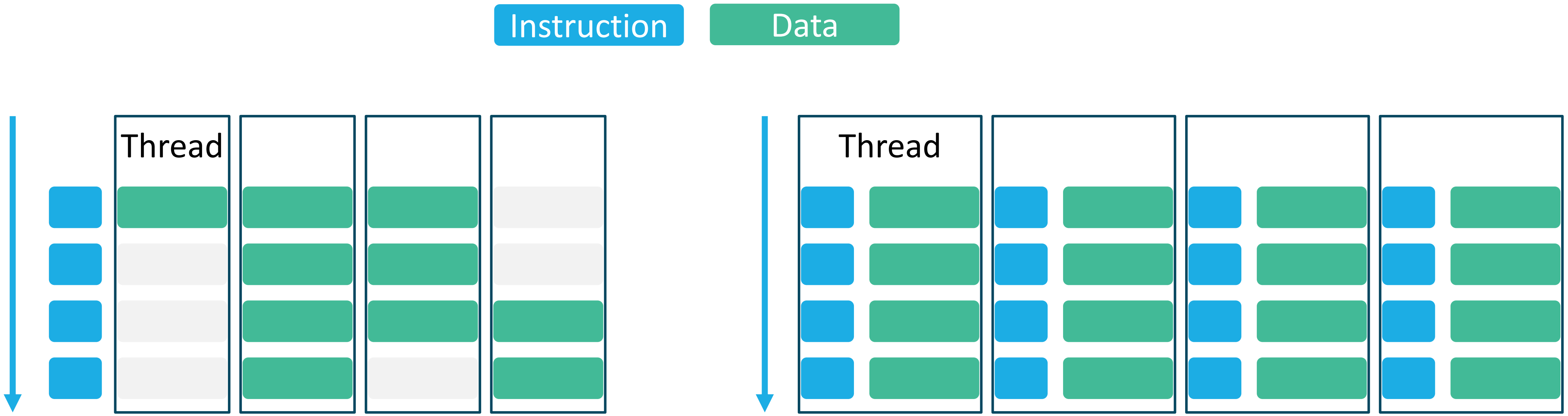
Evaluation of PEZY-SC4s

Summary and Future plans

Concept of PEZY-SCx

Accelerating "Single Program, Multiple Data (SPMD)" applications

- Based on "Multiple Instruction, Multiple Data (MIMD)" architecture
- MIMD is more efficient for applications with highly independent threads



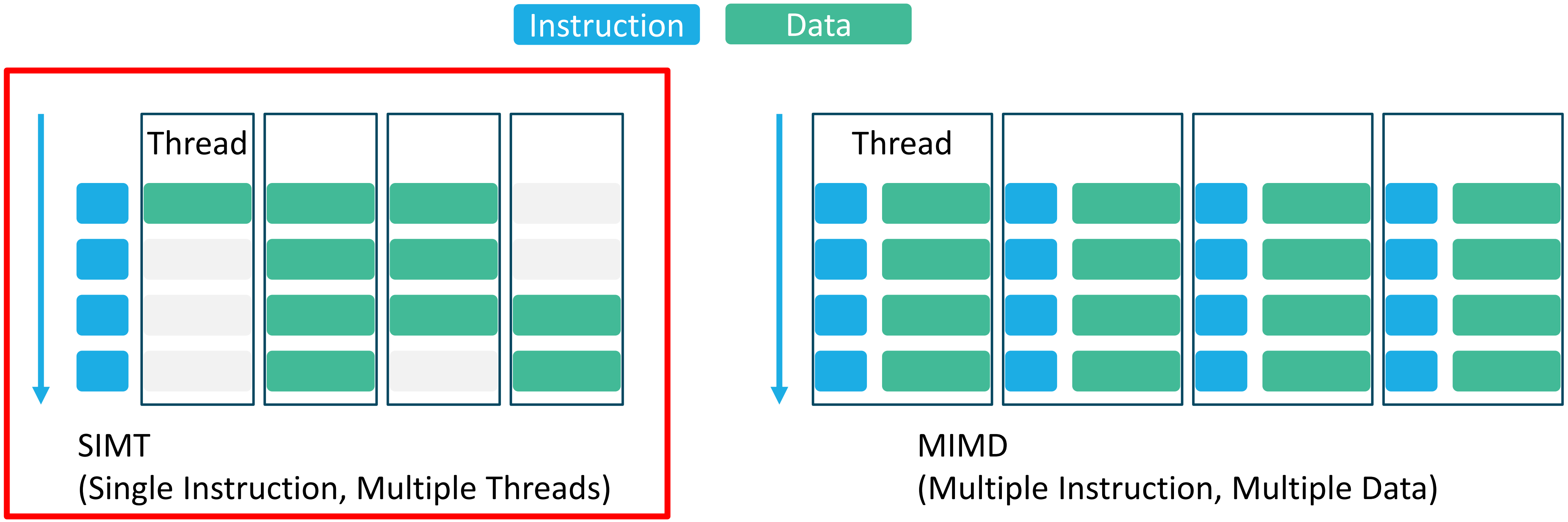
SIMT
(Single Instruction, Multiple Threads)

MIMD
(Multiple Instruction, Multiple Data)

Concept of PEZY-SCx

Accelerating "Single Program, Multiple Data (SPMD)" applications

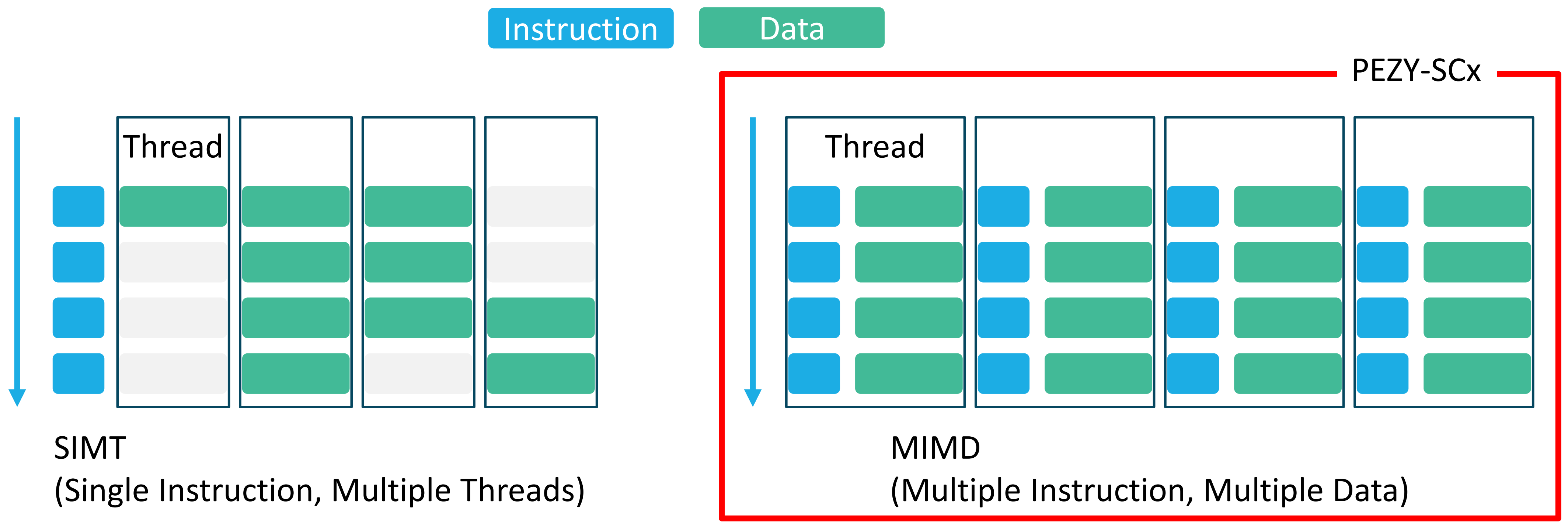
- Based on "Multiple Instruction, Multiple Data (MIMD)" architecture
- MIMD is more efficient for applications with highly independent threads



Concept of PEZY-SCx

Accelerating "Single Program, Multiple Data (SPMD)" applications

- Based on "Multiple Instruction, Multiple Data (MIMD)" architecture
- MIMD is more efficient for applications with highly independent threads



Our MIMD architecture

Processor elements (PEs) that utilize many threads

- Fine-grained multithreading
- Coarse-grained multithreading

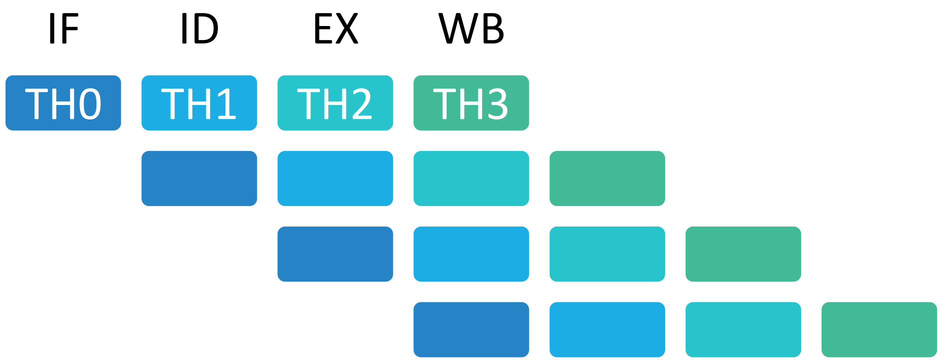
Data supply for many threads

- Local memory storage
- Amplifying bandwidth with hierarchical cache

Thread synchronization

- Explicit thread and cache synchronization
- Chip-level data operation

Fine-grained Multithreading



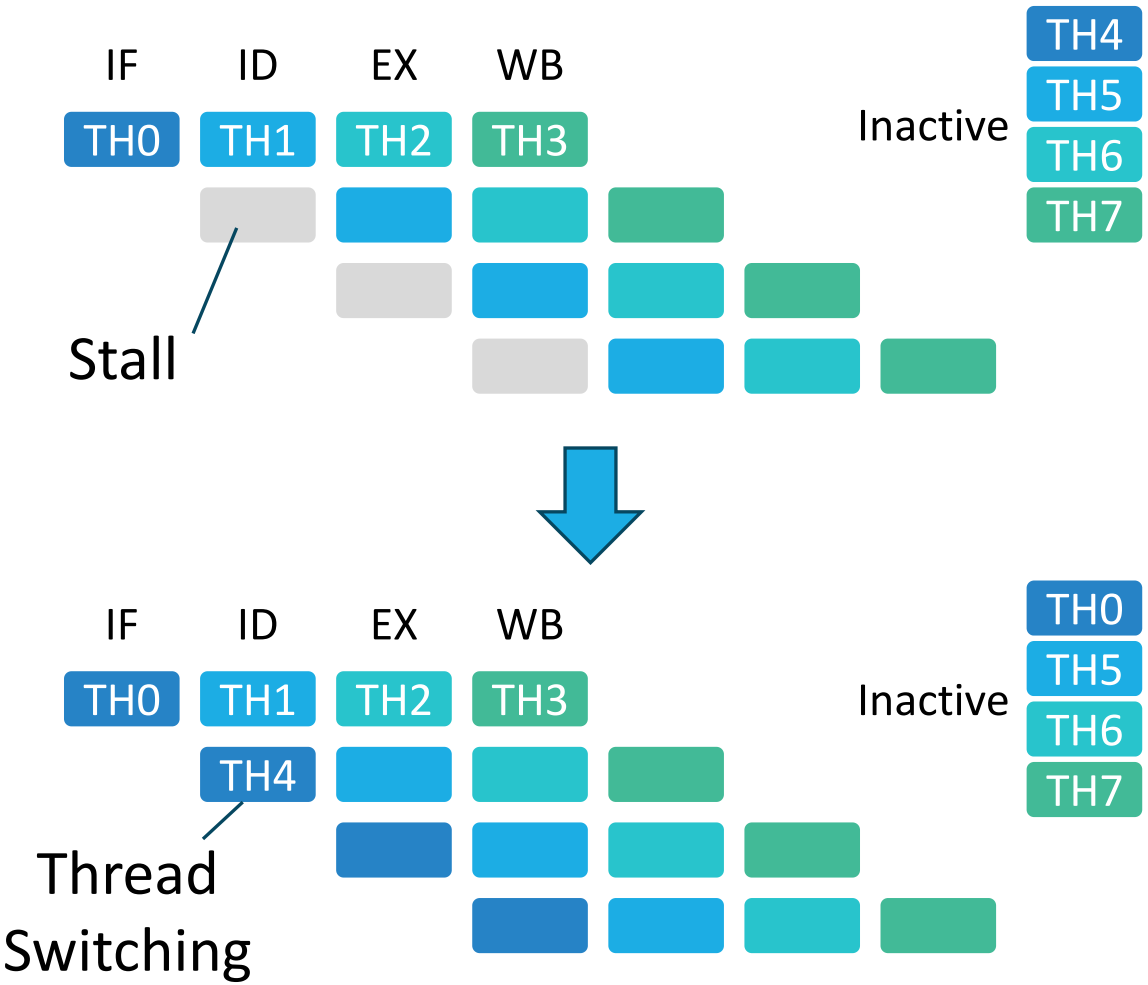
* Simplified pipeline
for explanation

"Barrel Processor"

- Each pipeline stage can be filled by different threads
- Branch prediction and out-of-order issuing are not necessary
- Short latency (a few cycle) can be hidden

Processor elements become compact and efficient

Coarse-grained Multithreading



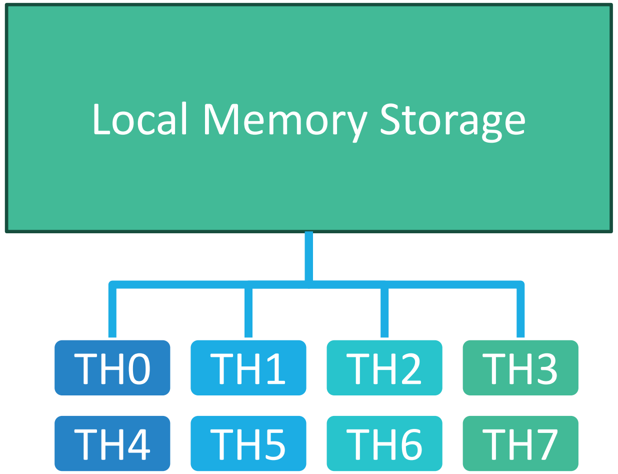
Each thread in pipeline consists of a pair of threads

- Active and inactive threads
- Active/inactive states can be switched

Thread switching can hide long memory latency

- Thread switching instruction
- Instruction with switching flag
  - e.g. Load from memory

Local Memory Storage



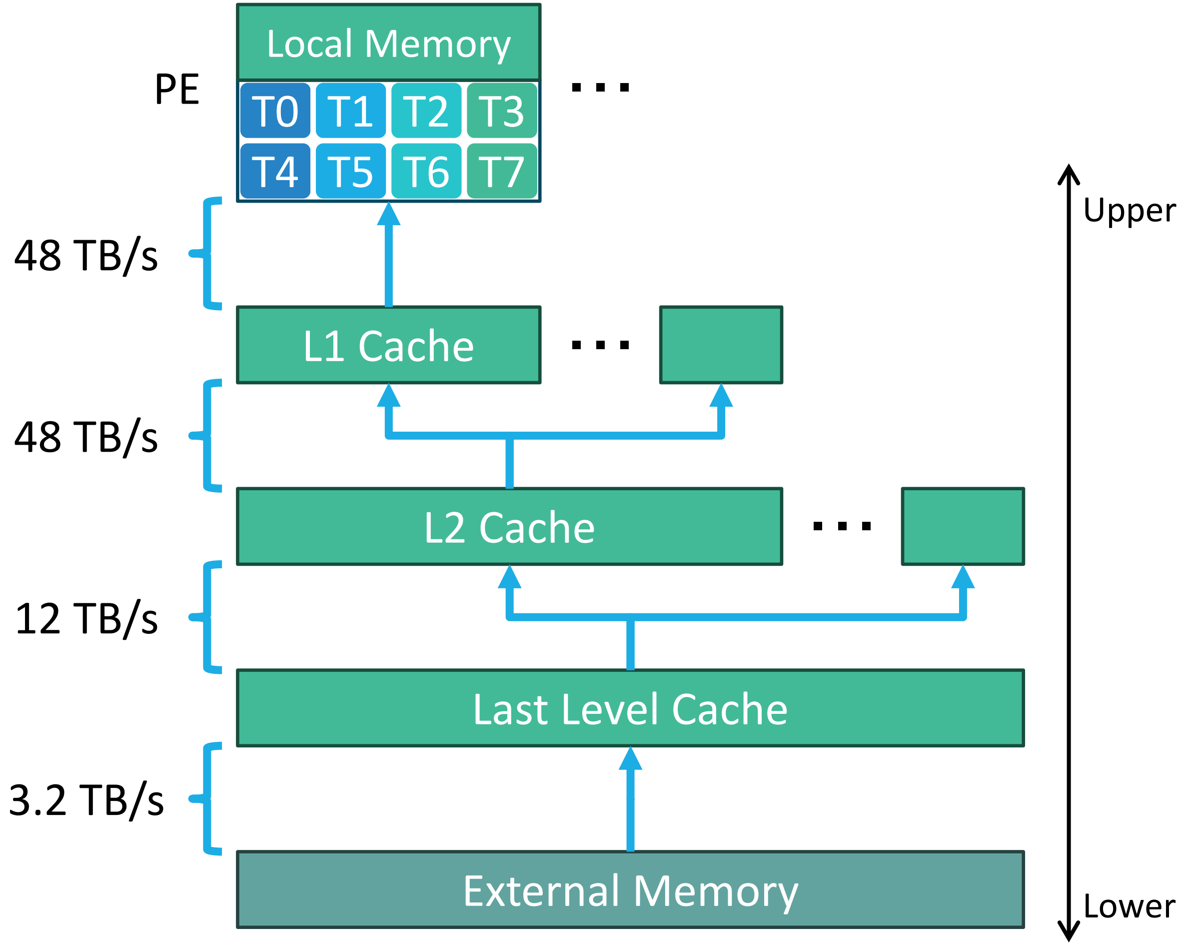
Memory storage for

- "Stack Region"
  - Automatically used by compiler
- Data with high locality
  - Explicitly usable by users

Shared by all threads within a PE

- Data synchronization among threads

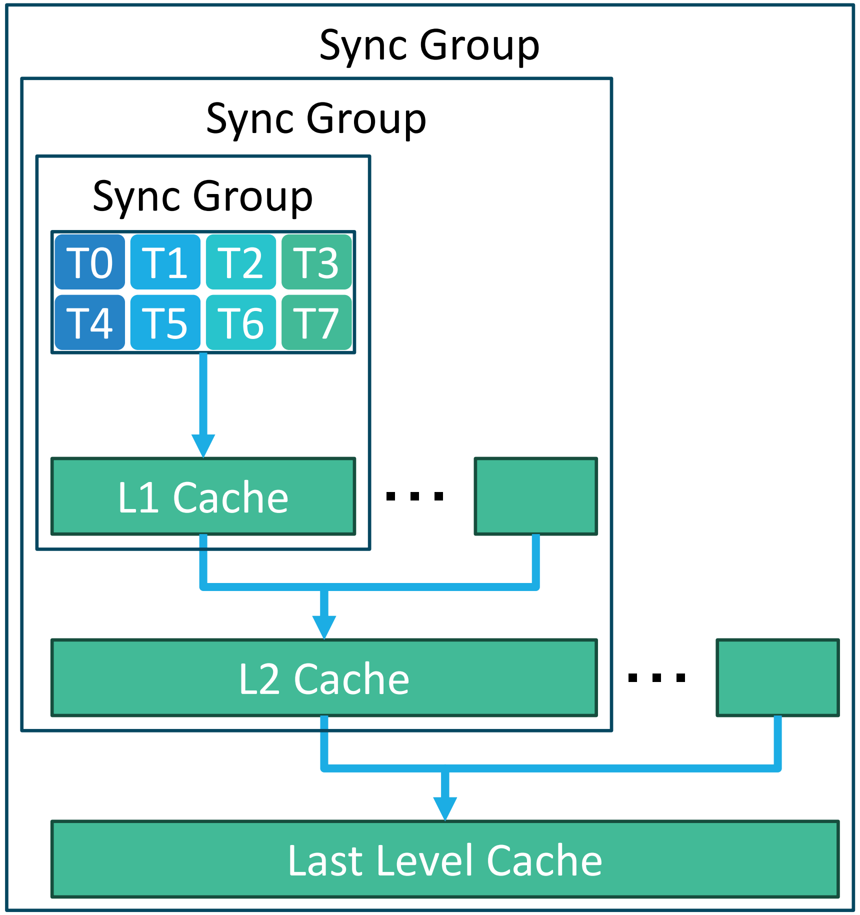
Amplifying Bandwidth with Hierarchical Cache



Each cache has many upper caches

- For example, a L2 cache is connected to many L1 caches
- Cache lines are repeatedly accessed by upper caches
  - even if the line is used once by each thread
- Caches can provide more bandwidth than lower hierarchies

Explicit Thread and Cache Synchronization



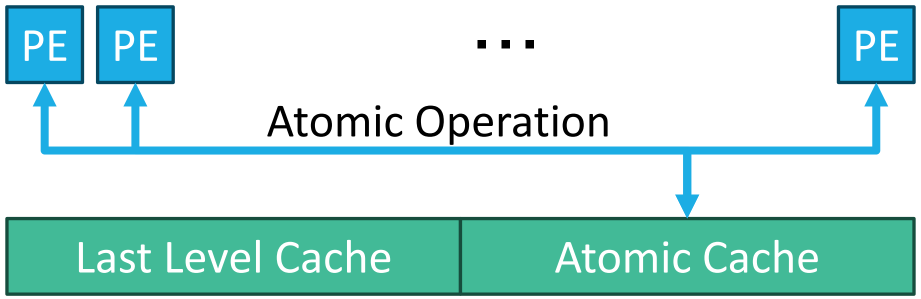
Instruction-based synchronization

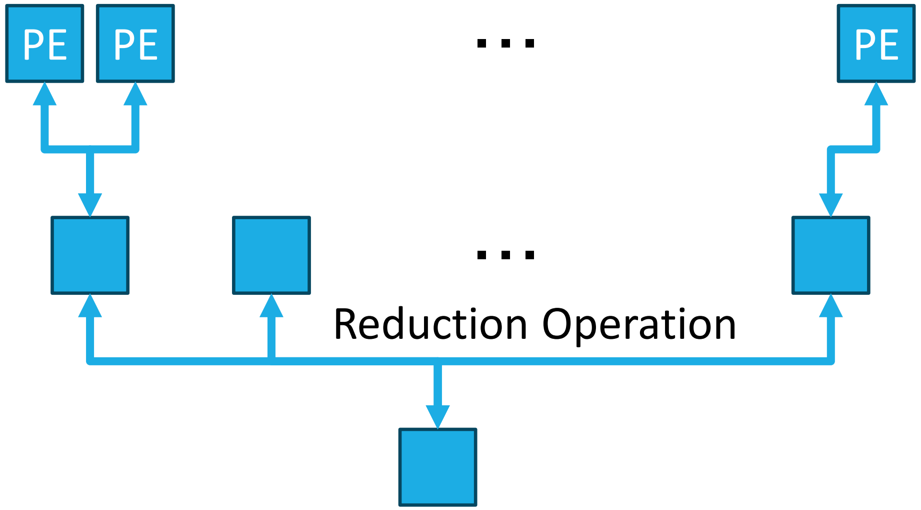
- Sync/Flush instruction
  - Synchronize all Program Counters (PC) in the group
  - Flush all dirty lines in cache and synchronize PCs
- Configurable synchronization group by instruction operand

Automatic cache coherency mechanism is not required

- Less complexity and more bandwidth

Chip-level Data Operation





Atomic

- Atomic cache near Last Level Cache
- Supported operations
  - exchange, CAS, add, sub, min, max, inc, dec

Reduction

- Dedicated tree network for whole-chip reduction
- Supported operations
  - add, max, min, and, or

Architecture Summary

Efficient PEs with hierarchical caches

- Fine/coarse-grained multithreading
- Explicit cache synchronization

Features to boost performance

- Local memory storage
- Atomic and reduction

Usability comparable to general microprocessor

- Easily bring up software using compiler technology
- Tune the software to utilize performance-boosting features

Agenda

Architecture of PEZY-SCx Series

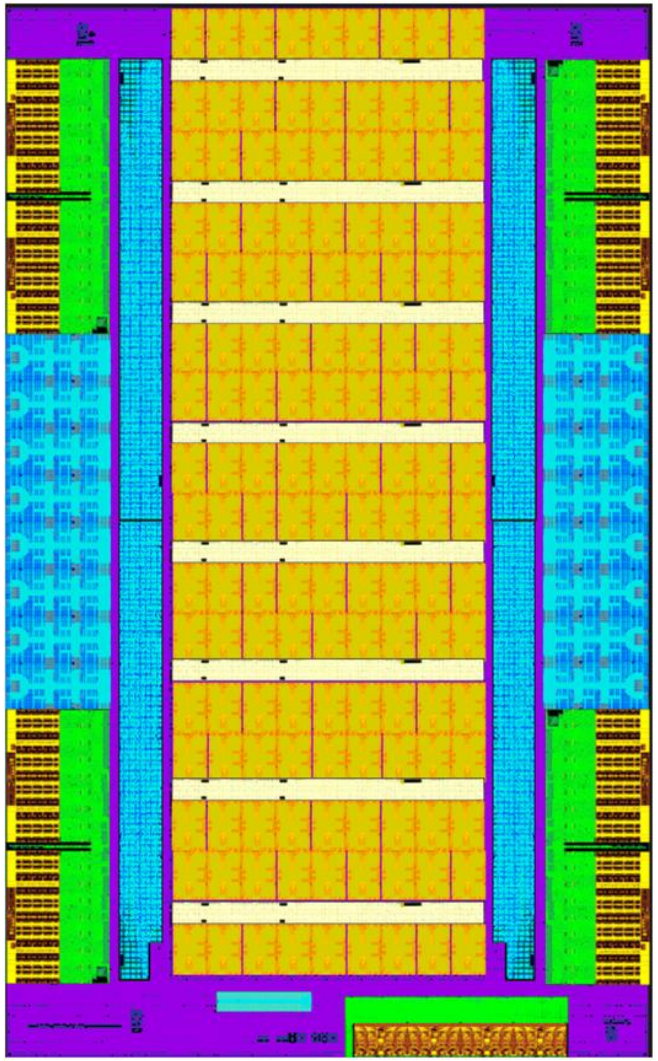
Implementation of PEZY-SC4s

Software

Evaluation of PEZY-SC4s

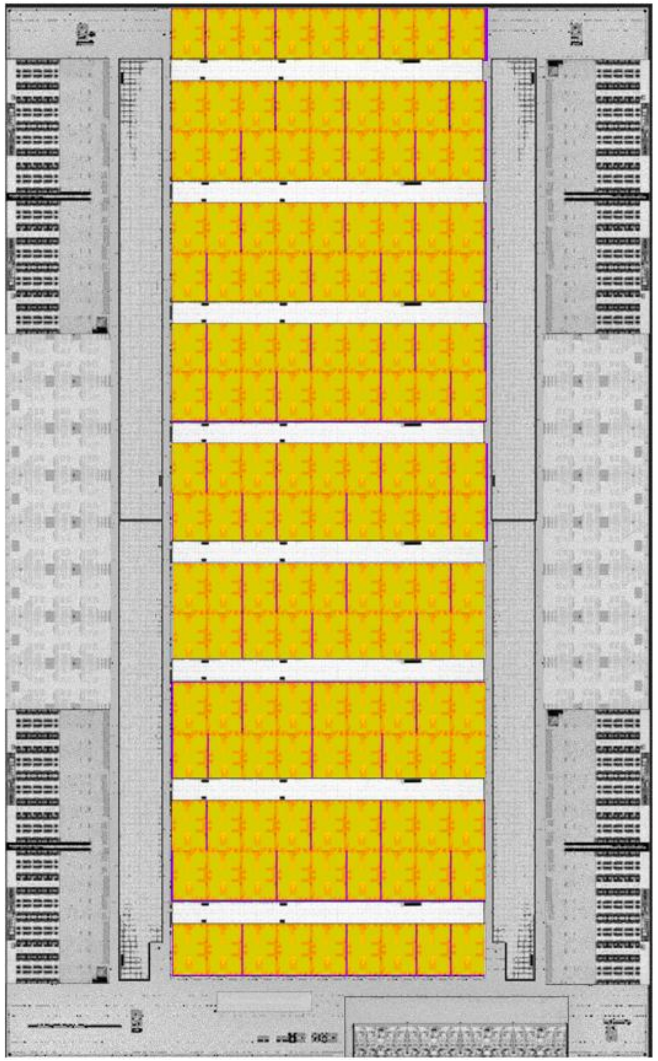
Summary and Future plans

Overview



Process	: TSMC 5 nm FinFET
Die size	: 18.4 mm x 30.2 mm
Gate Count	: 4.8 billion gates
SRAM Cell	: 1.6 Gbits

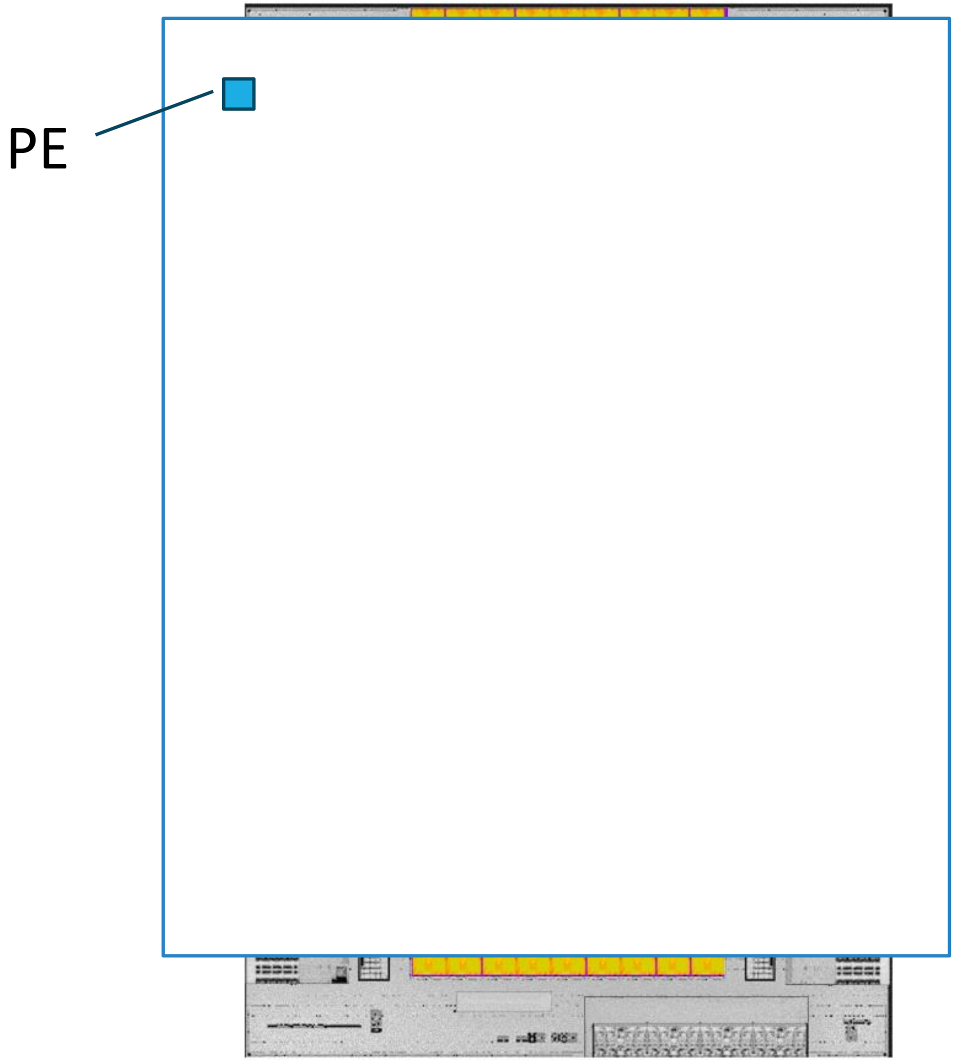
Processor Elements



Primary computing resources

- 2,048 PEs (16,384 threads)
- Hierarchical structure of PEs and caches

Processor Element (PE)



RISC-like ISA

- Integer arithmetic
- Floating-point arithmetic
  - Double, float, half, BF16

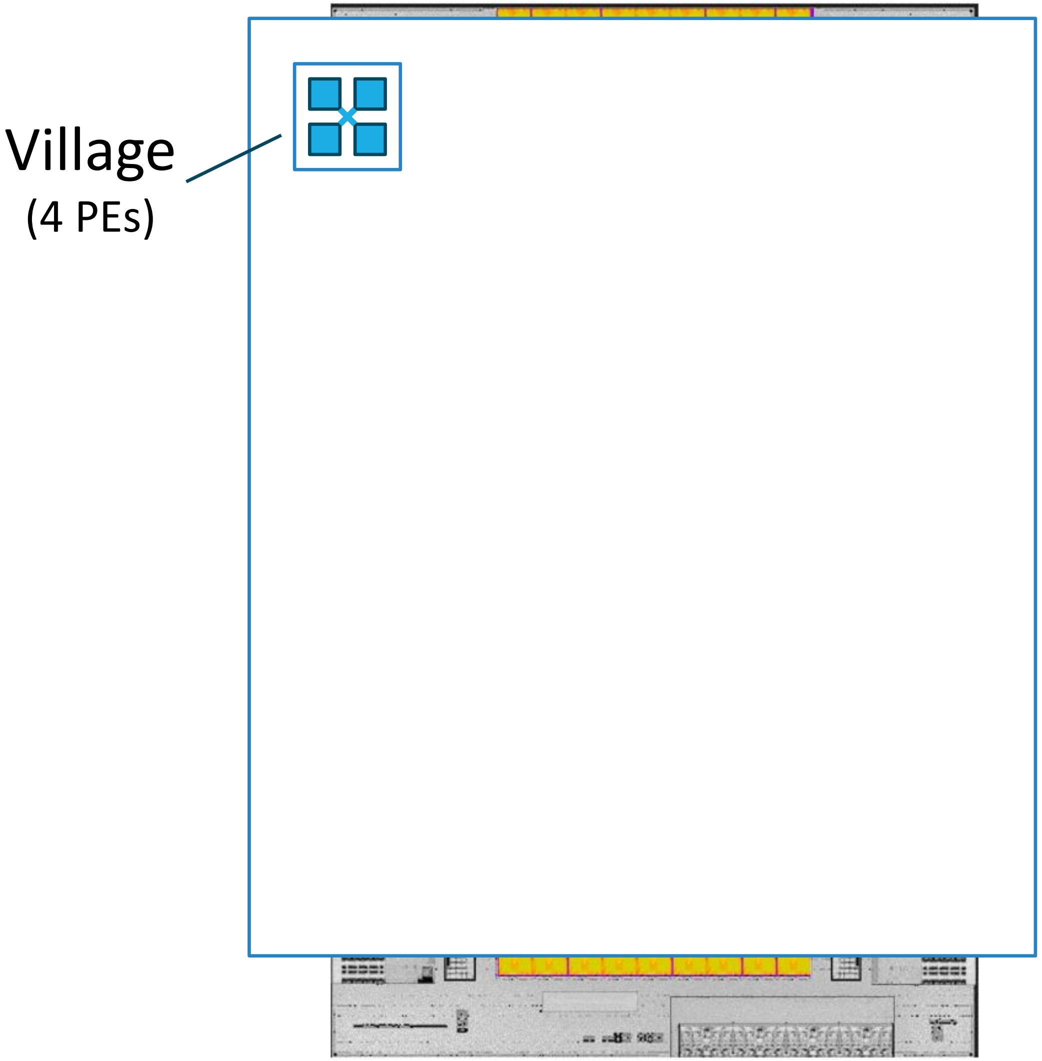
Resources

- Hardware threads : 8
- L1 I-Cache : 4 KB
- L1 D-Cache : 4 KB
- Local Storage : 24 KB

Clock frequency

- 1.5 GHz

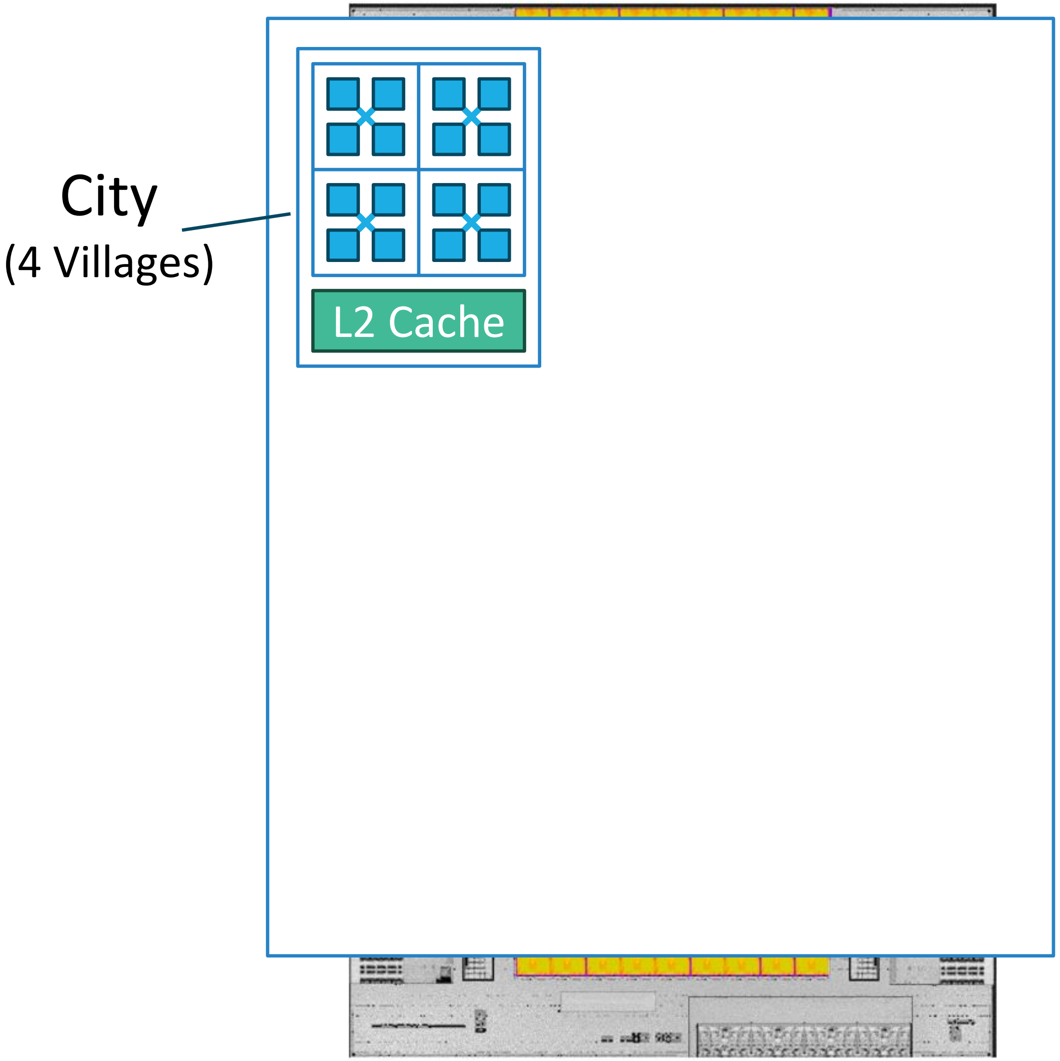
Hierarchy of PEs



Village (4 PEs)

- Shares local memory storage

Hierarchy of PEs



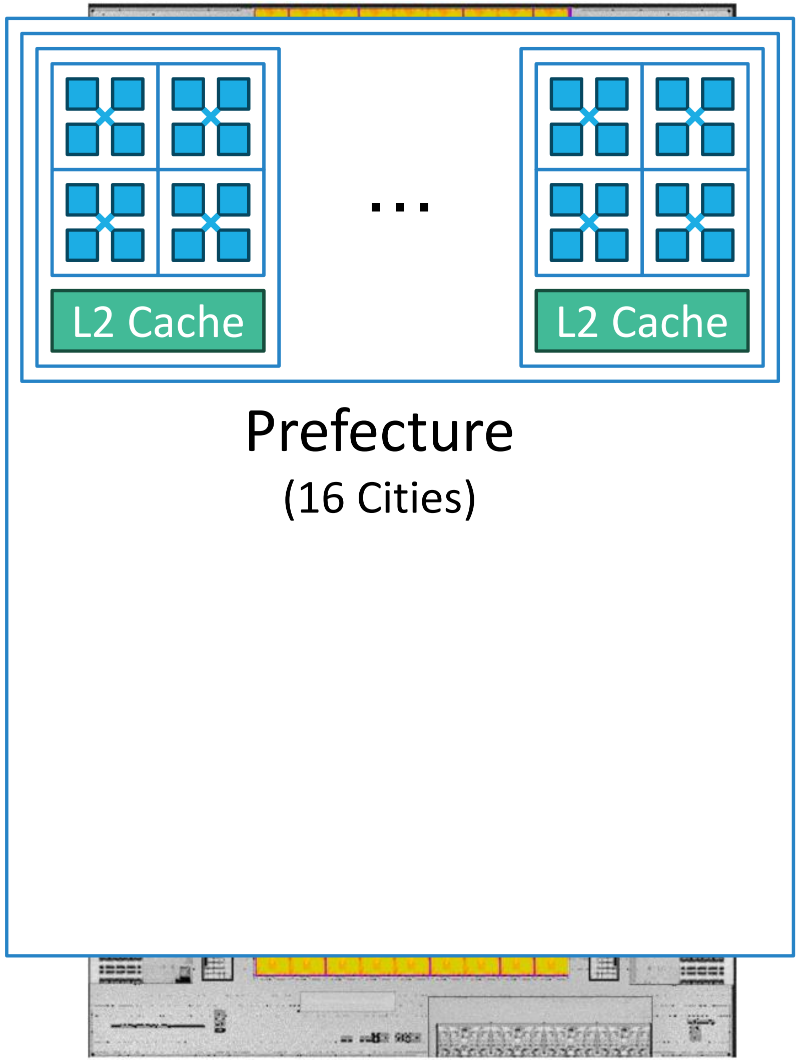
Village (4 PEs)

- Shares local memory storage

City (4 Villages)

- L2 I-Cache : 32 KB
- L2 D-Cache : 64 KB

Hierarchy of PEs



Village (4 PEs)

- Shares local memory storage

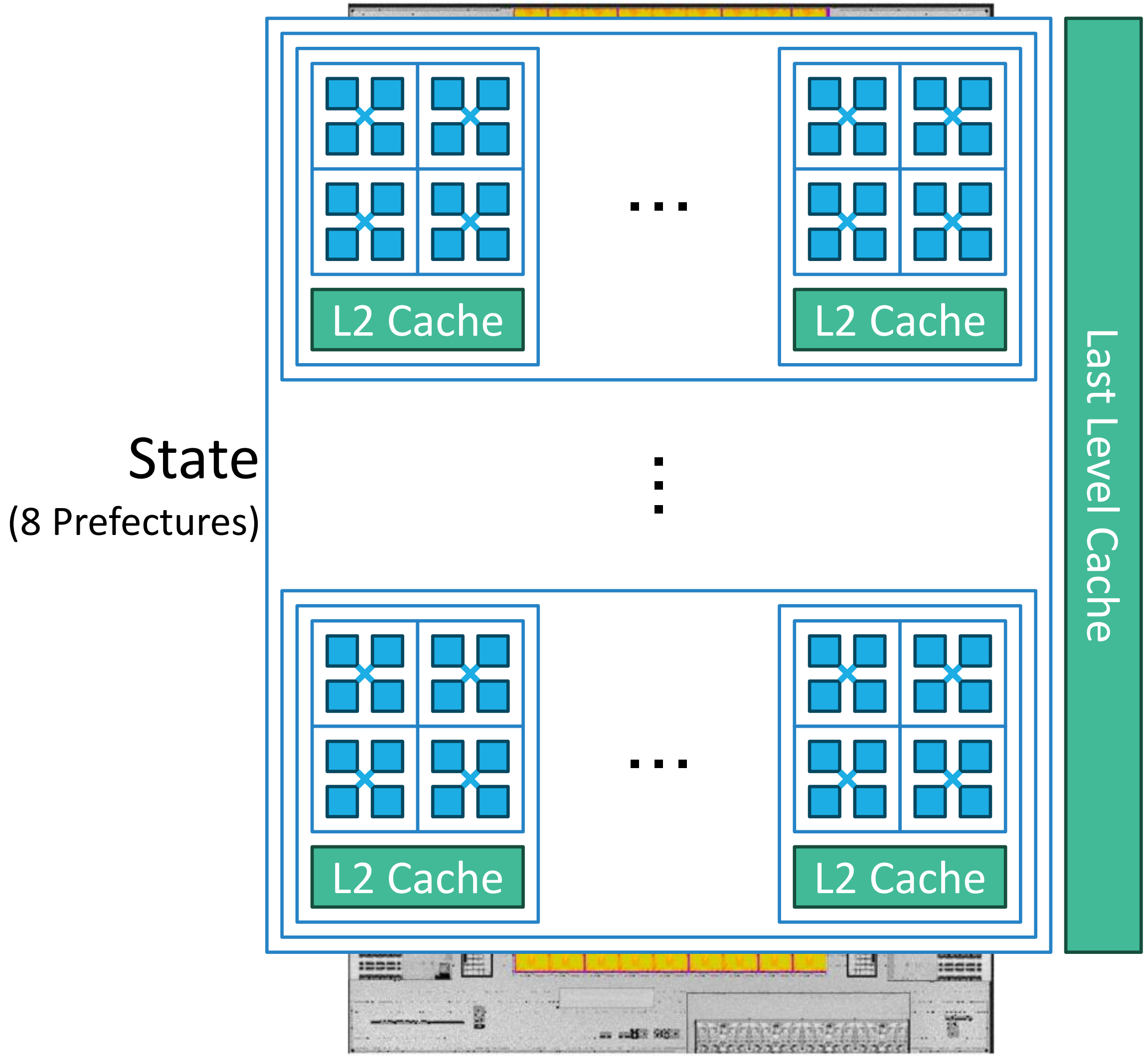
City (4 Villages)

- L2 I-Cache : 32 KB
- L2 D-Cache : 64 KB

Prefecture (16 Cities)

- Redundancy by enabling 16 out of 18 cities

Hierarchy of PEs



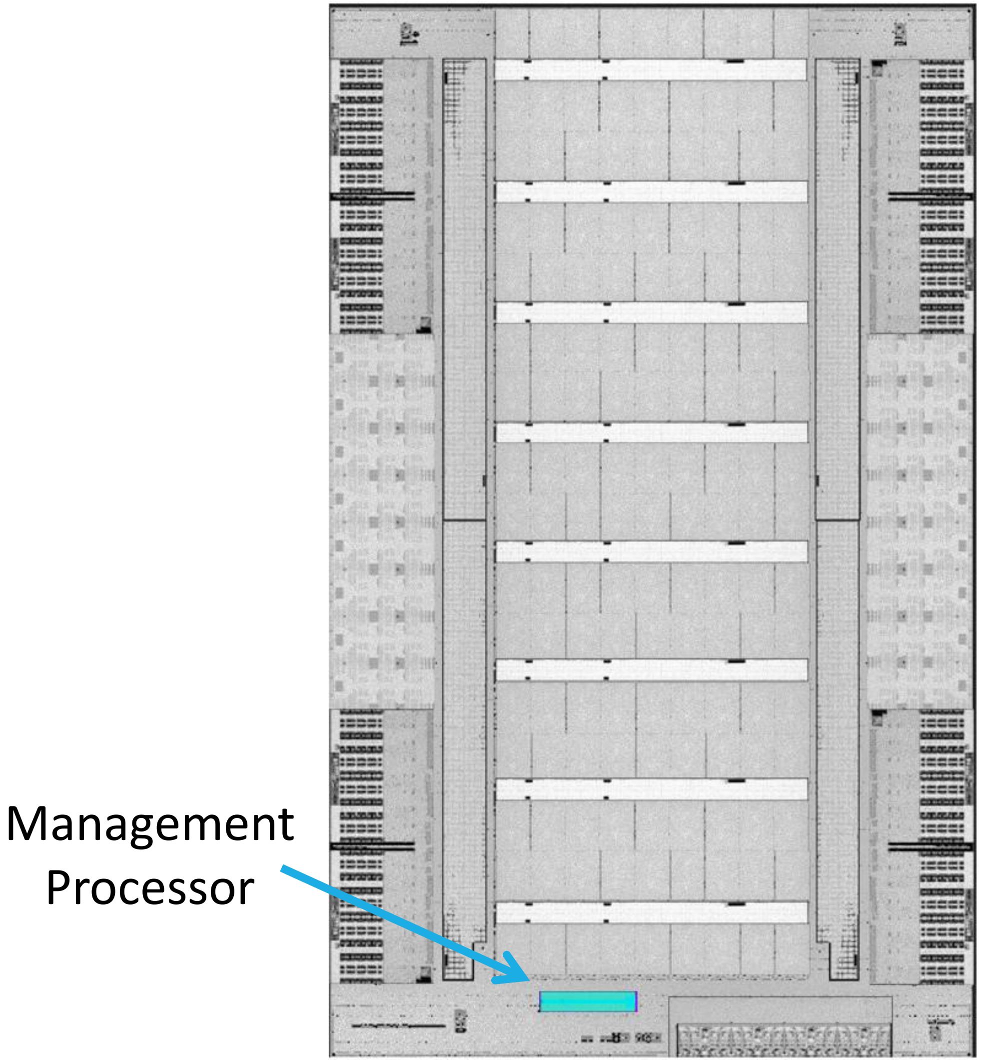
Village (4 PEs)
- Shares local memory storage

City (4 Villages)
- L2 I-Cache : 32 KB
- L2 D-Cache : 64 KB

Prefecture (16 Cities)
- Redundancy by enabling 16 out of 18 cities

State (8 Prefectures)
- Last Level Cache : 64 MB

Management Processor



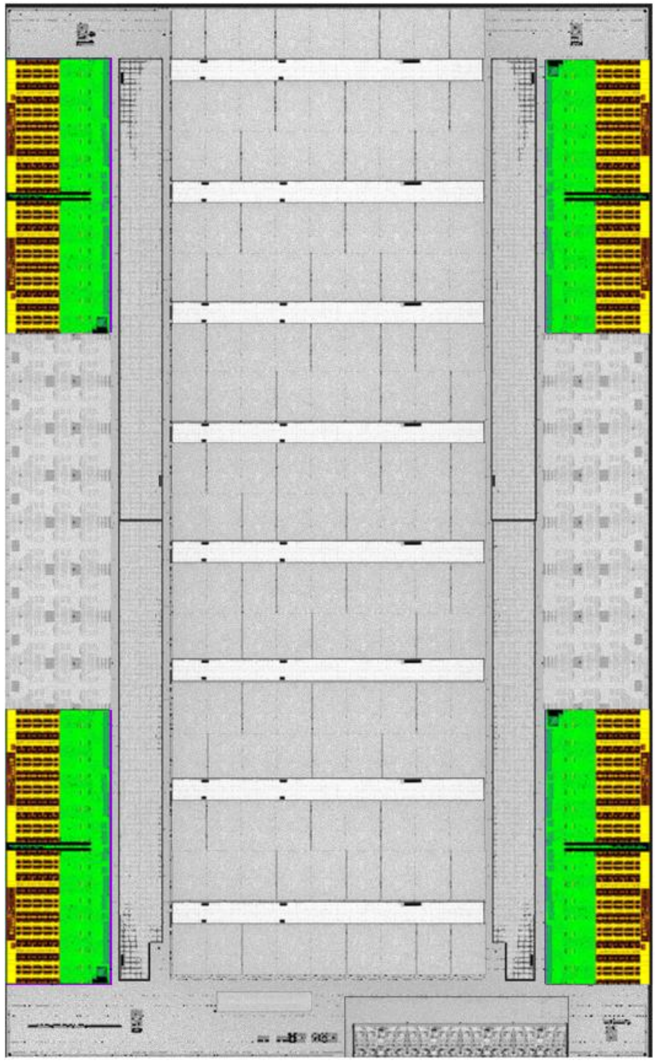
RISC-V processor

- 4 cores
- 1.5 GHz

Rocket Core

- Open source RISC-V implementation
  - https://github.com/chipsalliance/rocket-chip
- In-order scalar processor

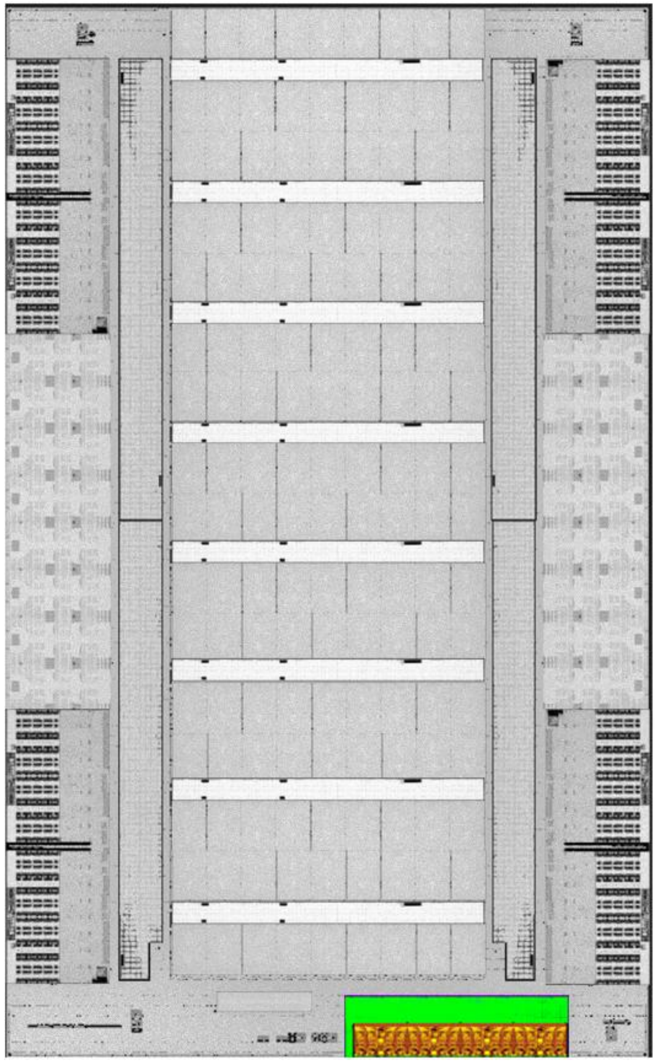
External Memory



HBM3

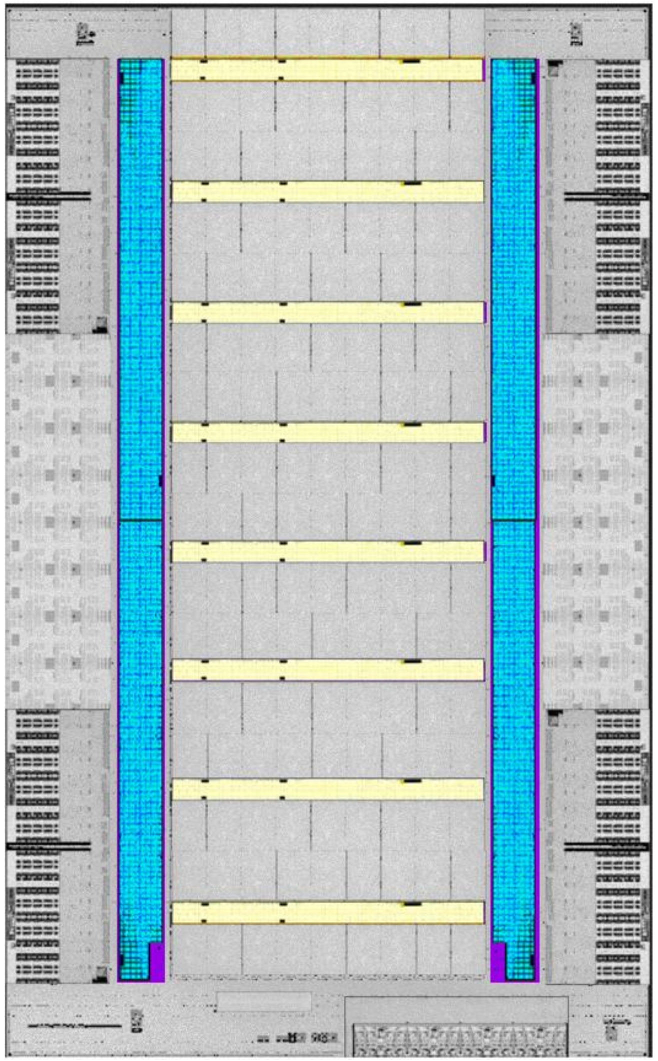
- 4 devices
- Bandwidth : 3.2 TB/s
- Capacity : 96 GB

External Interface



PCIe Gen5
- 16 lanes
- Bandwidth : 64 GB/s

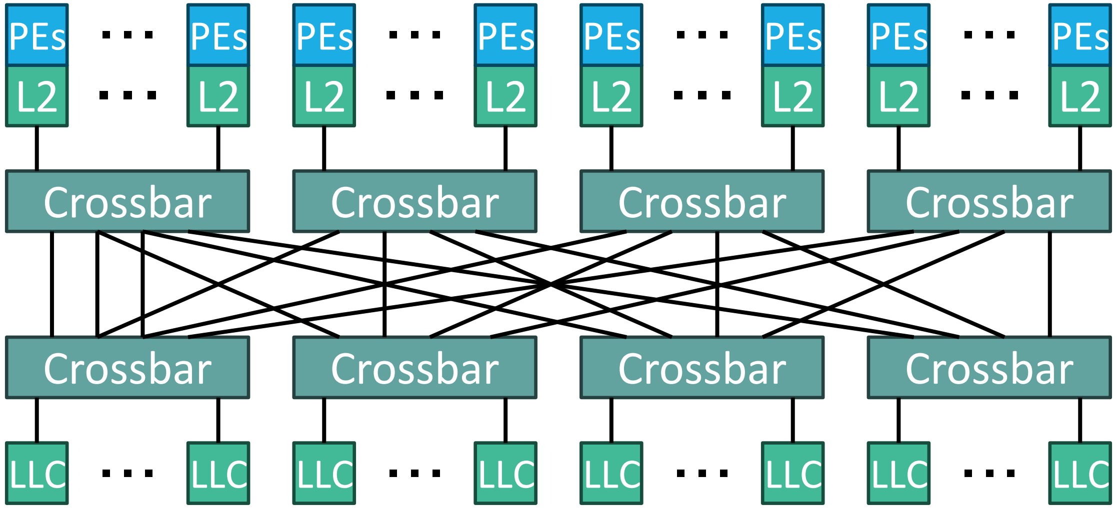
Internal Bus



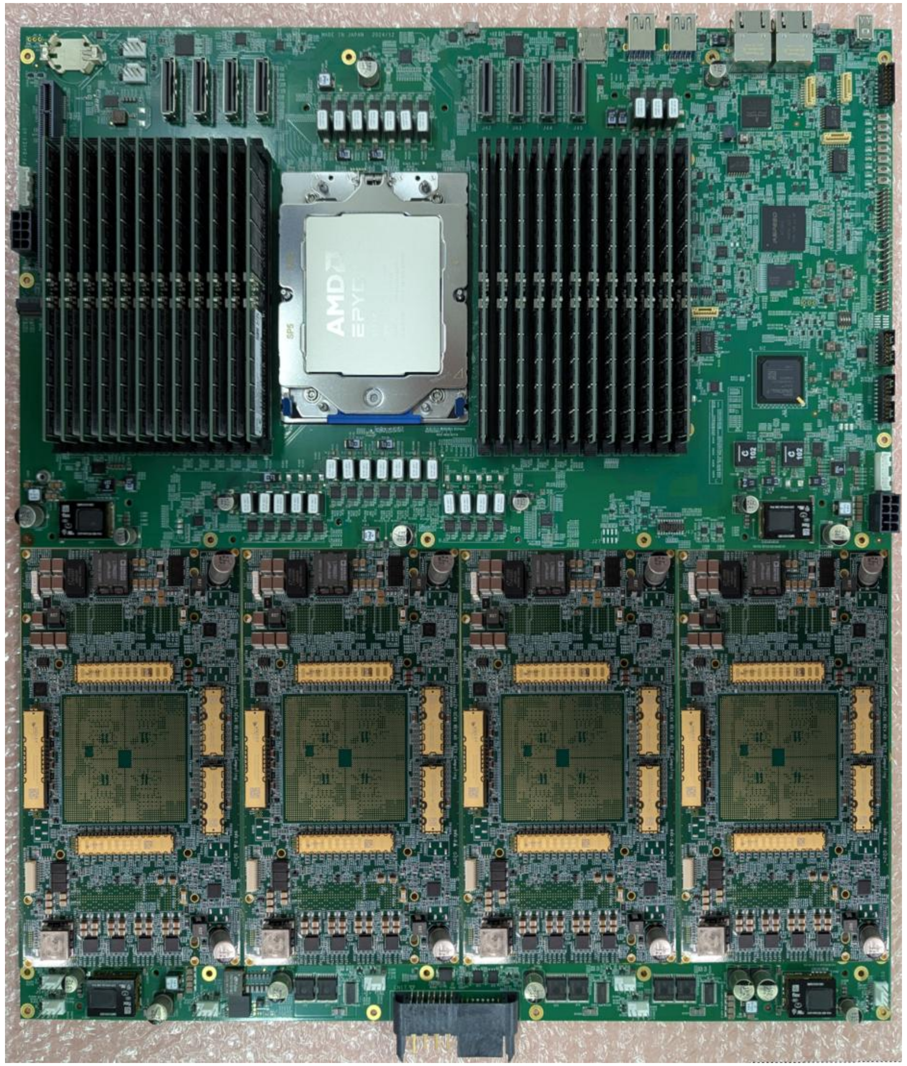
Custom Bus Architecture

- Bandwidth (Read) : 12 TB/s
- Bandwidth (Write) : 6 TB/s

Crossbar-based connection



System Development



Module/system board for the supercomputer system is ready

Node with host CPU and PEZY-SC4s

- AMD EPYC 9555P : 1
- PEZY-SC4s : 4
- NDR InfiniBand

Planned system configuration

- Nodes : 90
- Total PEs : 737,280
- R_{peak} : 8.6 PFLOPS (Double Precision)

Agenda

Architecture of PEZY-SCx Series

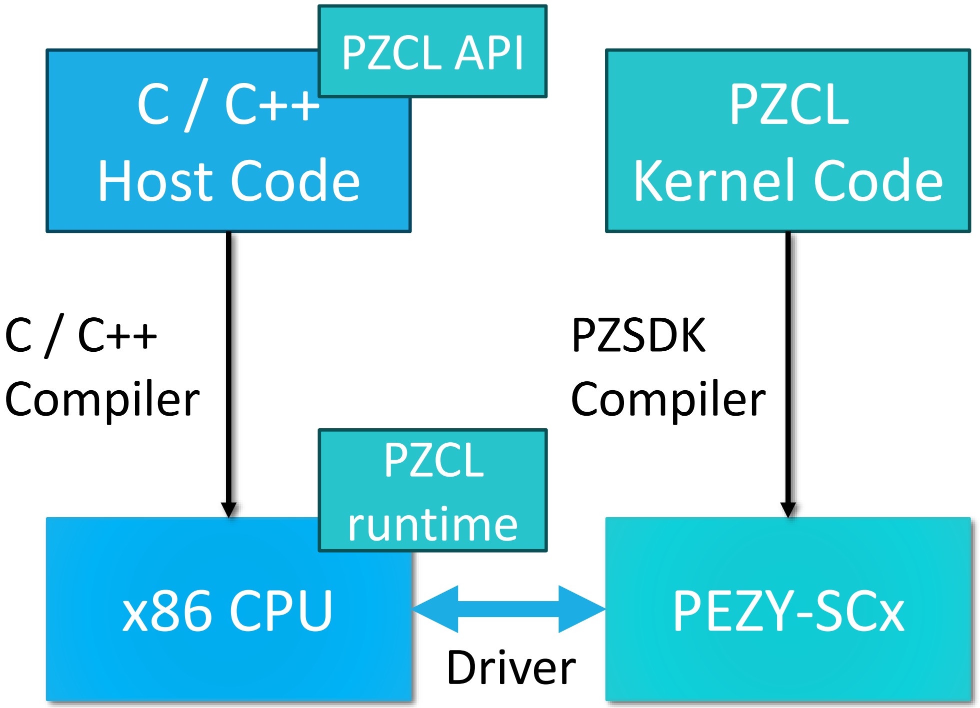
Implementation of PEZY-SC4s

Software

Evaluation of PEZY-SC4s

Summary and Future plans

Software Development Kit: PZSDK



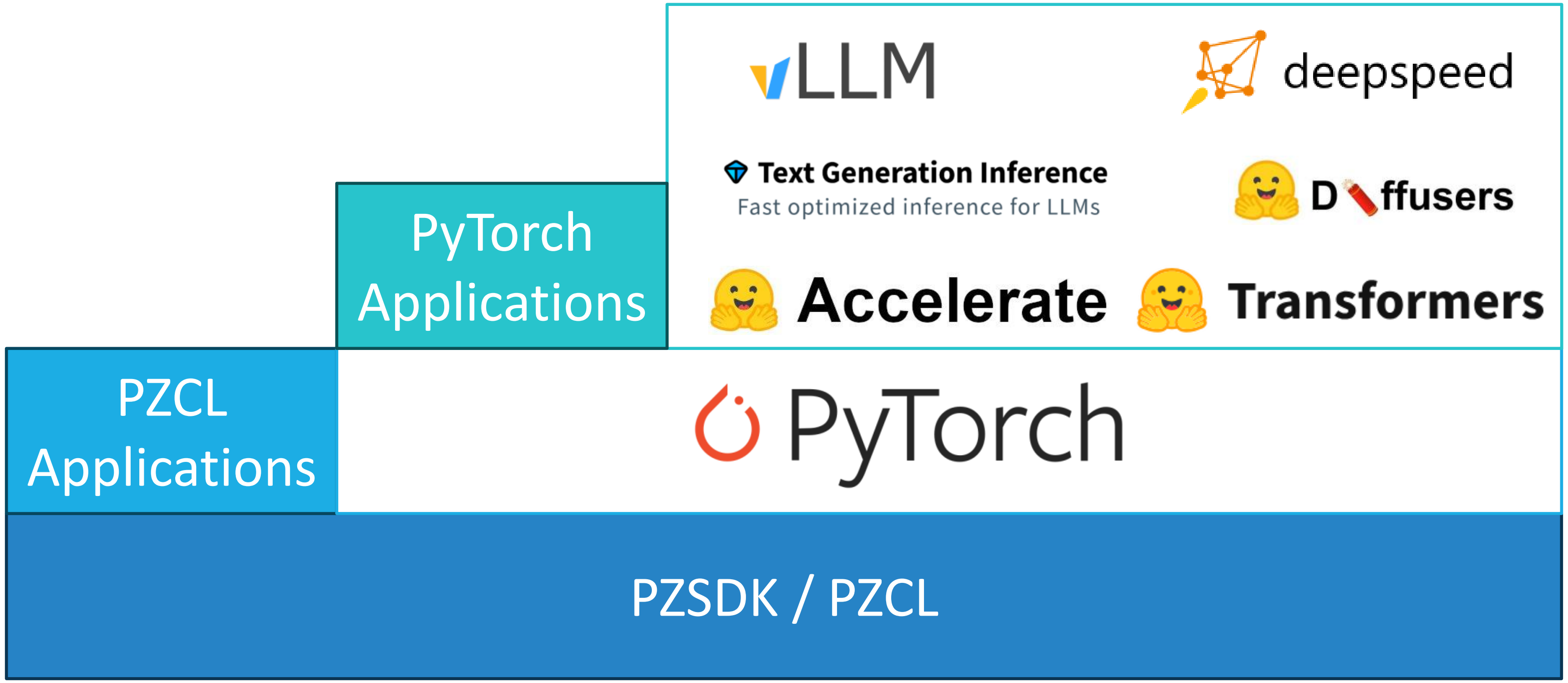
PZCL: OpenCL-like Programming API

- Host code : C / C++ with PZCL API
- Kernel code : PZCL C (OpenCL C-like)

Provided software components

- PZSDK compiler based on LLVM
- PZCL runtime library
- Driver

Software Stack



Porting Examples

Genome analysis pipeline

- GATK (Genome Analysis Toolkit) Best Practices
- 33 min/sample with PEZY-SC3 x 4
  - More than twice the performance of NVIDIA H100 (37 min/sample with H100 x 8)

LLM (Large Language Model) applications

- Several LLM models already run on PEZY-SC3
- Supported models
  - Gemma3, Llama3, Qwen2, Stable Diffusion 2, HuBERT, Vision Transformer

Agenda

Architecture of PEZY-SCx Series

Implementation of PEZY-SC4s

Software

Evaluation of PEZY-SC4s

Summary and Future plans

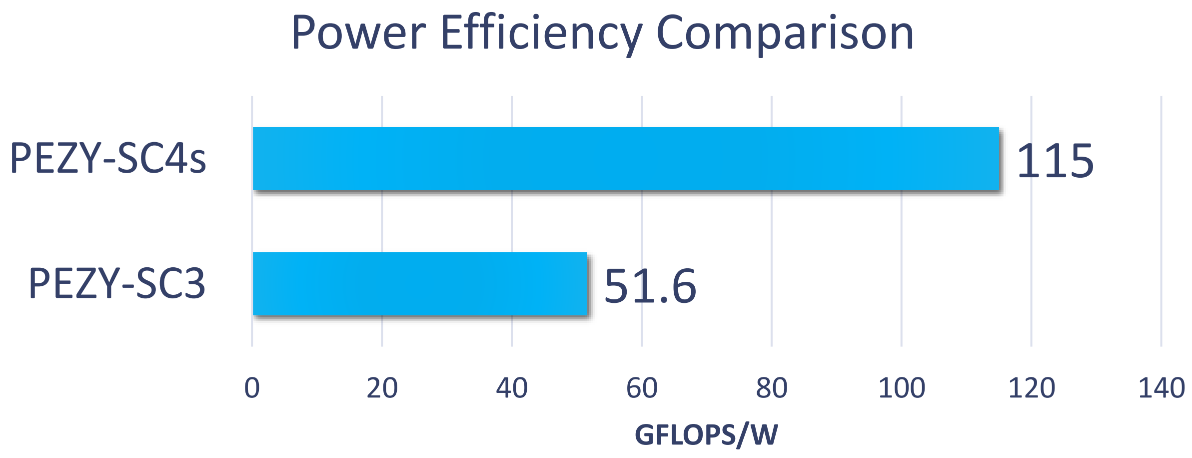
Evaluation of PEZY-SC4s

Power Efficiency of DGEMM

Performance of Smith-Waterman

Memory Bandwidth

Power Efficiency of DGEMM



Power estimation with gate-level netlist

- Simulator : Synopsys VCS
- RC extraction : Synopsys StarRC
- Power estimation : Synopsys PrimeTime PX

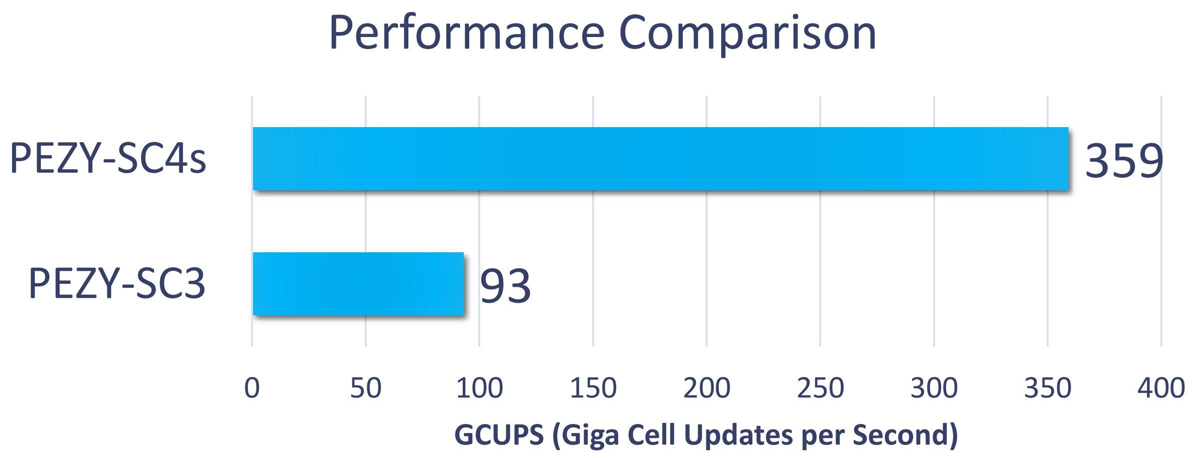
Benchmark program

- DGEMM (Double-precision GEneral Matrix Multiply)

Result

- Performance : 24.4 TFLOPS
- Efficiency : 99.2 %
- Power : 212 W (PEs only)
- Power Efficiency : 115 GFLOPS/W

Performance of Smith-Waterman



Performance estimation with RTL

- Simulator : Synopsys VCS

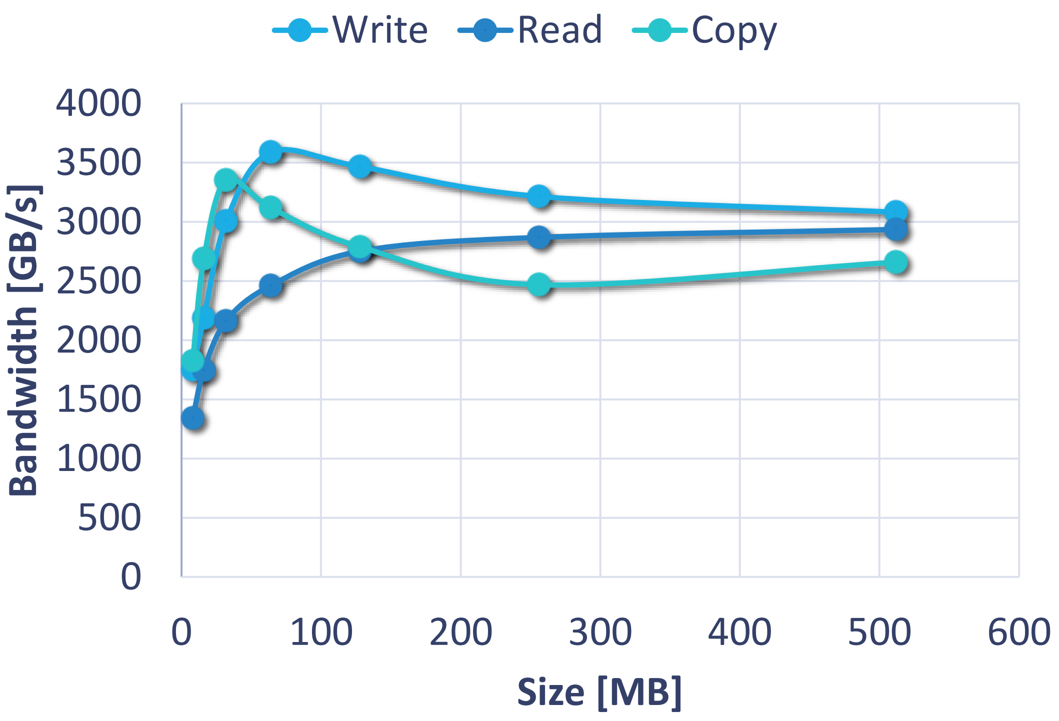
Benchmark program

- Smith-Waterman (Genome sequence alignment)

Result

- Performance : 359 GCUPS

Memory Bandwidth



Bandwidth evaluation

- Emulator : Synopsys ZeBu Server 5

Benchmark program

- Read (HBM to PEs)
- Write (PEs to HBM)
- Copy (HBM to PEs to HBM)

Result (512 MB)

- Read : 2.9 TB/s (91 %)
- Write : 3.0 TB/s (94 %)
- Copy : 2.6 TB/s (81 %)

Agenda

Architecture of PEZY-SCx Series

Implementation of PEZY-SC4s

Software

Evaluation of PEZY-SC4s

Summary and Future plans

Summary

Architecture of PEZY-SCx Series

- Optimized microarchitecture for MIMD processor

Implementation of PEZY-SC4s

- TSMC 5 nm FinFET, 4.8 billion gates

Software

- PZSDK with PyTorch support
- Several ported software packages, including major LLM models

Evaluation of PEZY-SC4s

- Performance and power efficiency significantly surpass PEZY-SC3

Future Plans

PEZY-SC5: The Fifth Generation of PEZY-SCx

- Currently under development
- Process : 3 nm or finer
- Release : scheduled for 2027

Veryl: A New Hardware Description Language as an Alternative to SystemVerilog

- We are developing Veryl as an Open Source Software
- Core components of PEZY-SC5 are being developed using Veryl
- https://veryl-lang.org/