# Enhancing the Open-Source P-Mesh Cache Coherence System for Open ISAs

Jonathan Balkind (@jbalkind)

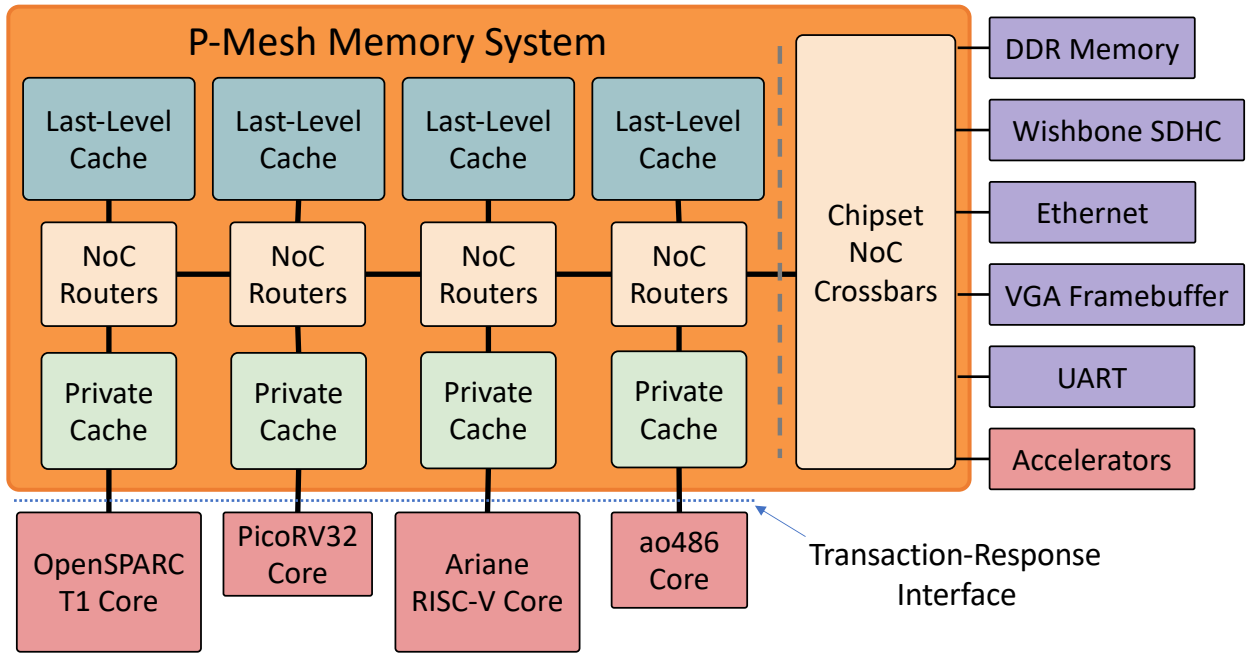
Princeton University



OpenPiton

# P-Mesh Cache Coherence System

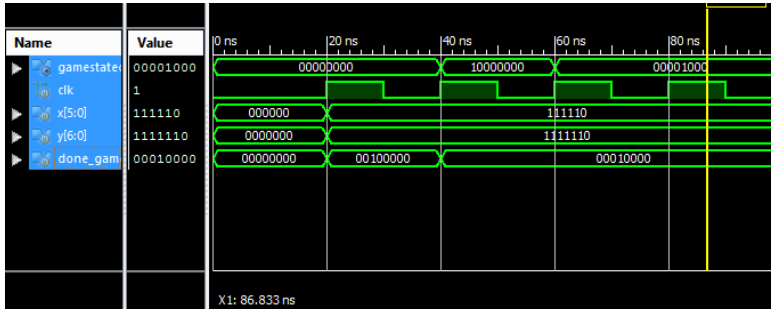
- Cache coherence system for manycore SoCs
- Scalable to ½ billion cores
- Transaction-Response Interface (TRI) for connecting new cores
- >10 cores connected with 5 different ISAs
- Support for heterogeneous-ISA systems
- Peripherals & accelerators connect with standard interfaces

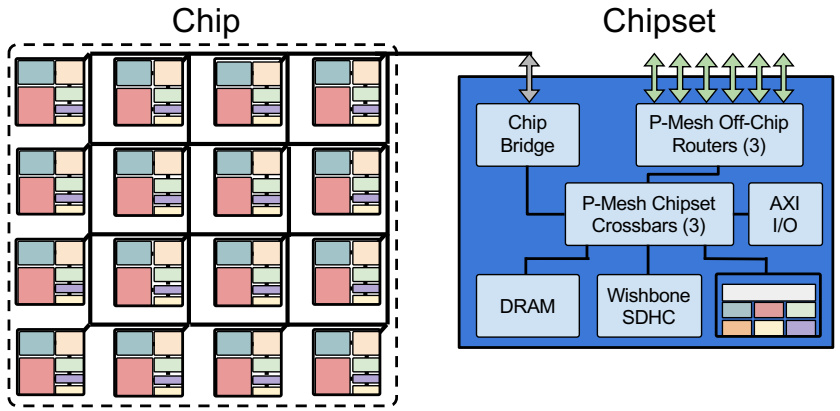





OpenPiton

# OpenPiton




- Open source manycore (since 2015)
- Written in Verilog RTL
- P-Mesh coherence scales to ½ billion cores
- Configurable core, uncore
- Validation suite with thousands of tests
- Simulation in VCS, ModelSim, Incisive, Verilator, Icarus, Riviera, Vivado
- Includes synthesis and back-end flow
- ASIC & FPGA verified, characterised [HPCA 2018]
- Runs full stack multi-user Debian Linux

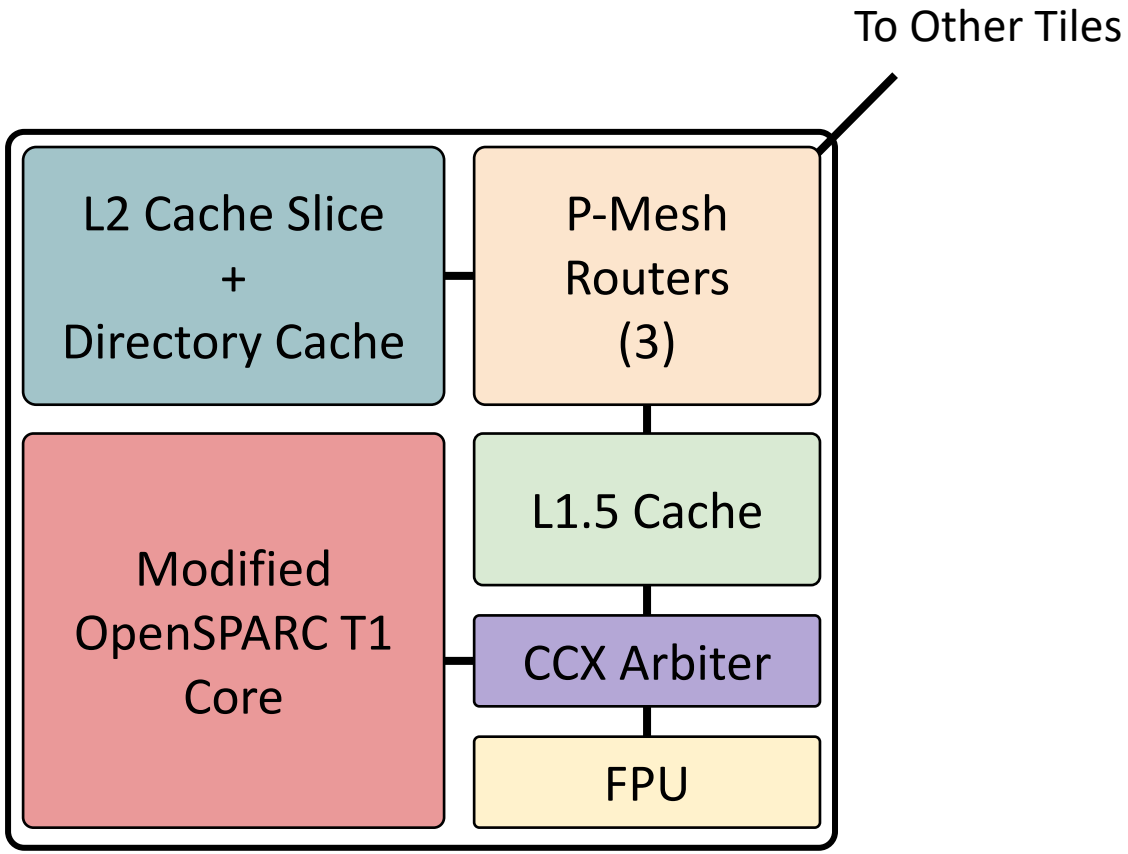
XILINX®

debian

OpenPiton

# OpenPiton

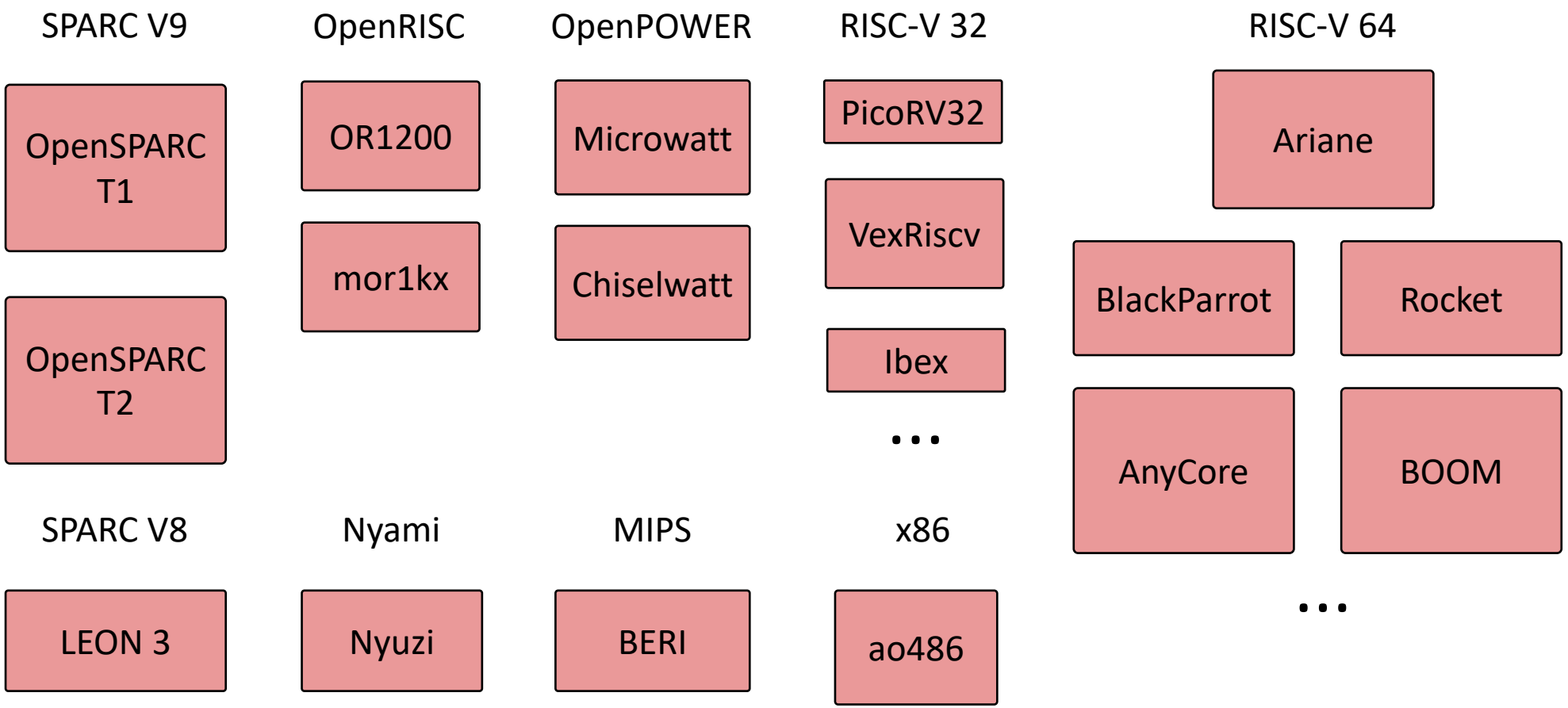
- P-Mesh previously supported only the OpenSPARC T1 core
- Featured numerous SPARC-specific specialisations
  - CCX was coupled to core implementation
- How to connect other cores?
- How to support other ISAs?






OpenPiton

# There are a lot of open cores to build with!

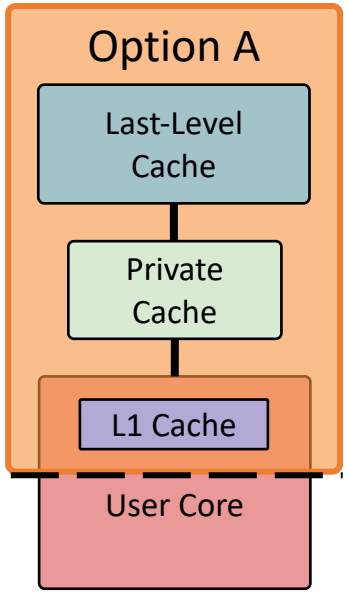





OpenPiton

# Core Interface Options

- Several interface options with different trade-offs
- We want to minimise effort to connect new cores



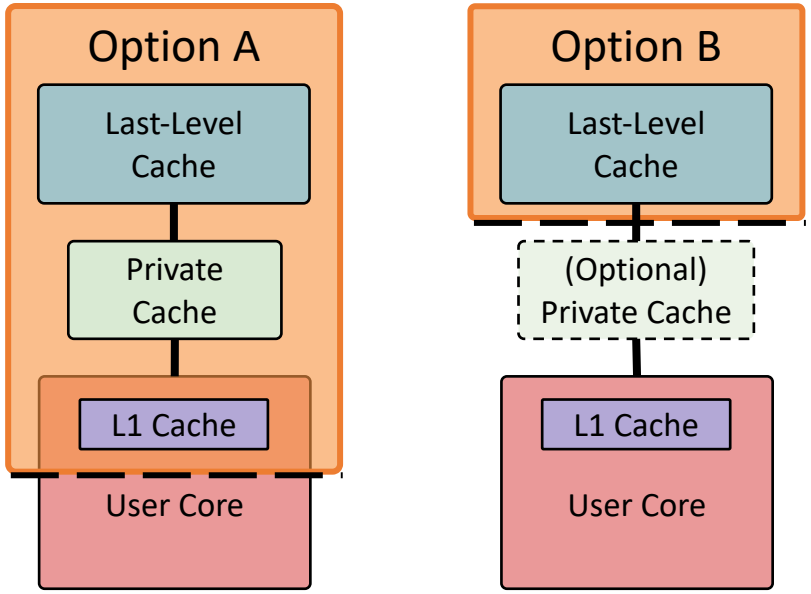

Provide 3 cache levels
Interface before L1



OpenPiton

# Core Interface Options

- Several interface options with different trade-offs
- We want to minimise effort to connect new cores




Provide 3 cache levels
Interface before L1

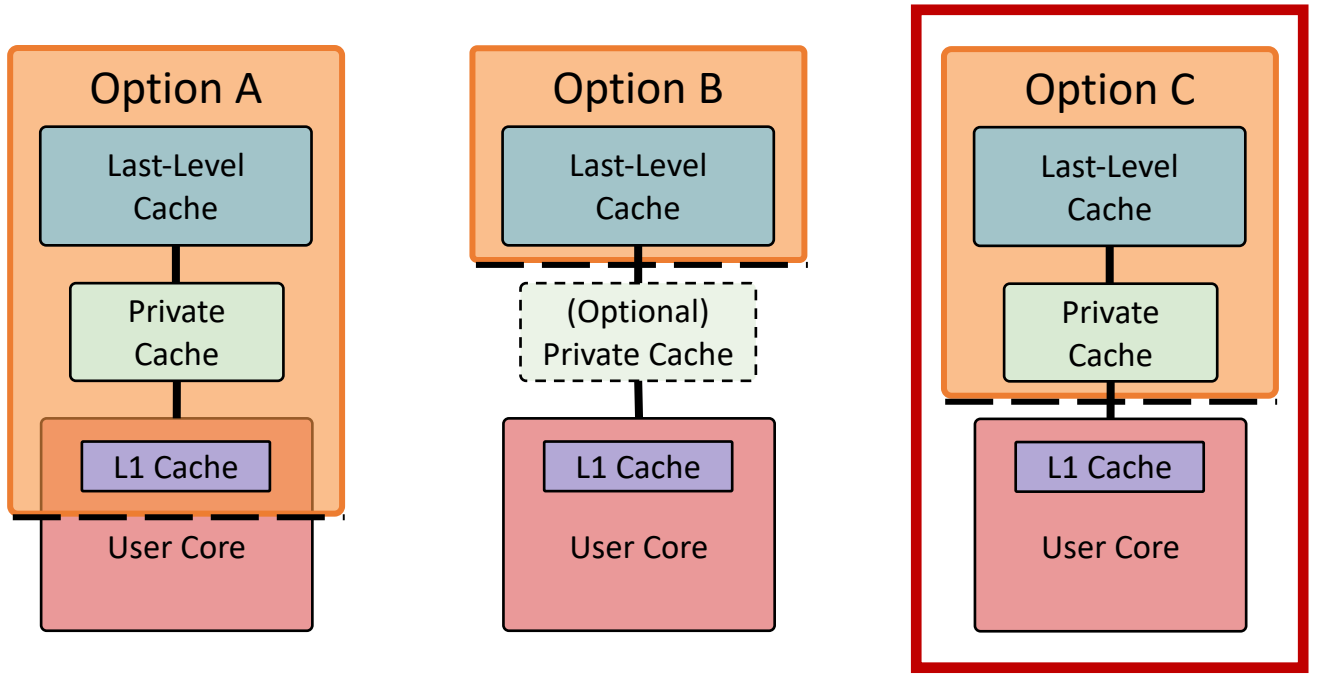
Provide 1 cache level
Interface before LLC



OpenPiton

# Core Interface Options

- Several interface options with different trade-offs
- We want to minimise effort to connect new cores




Provide 3 cache levels
Interface before L1

Provide 1 cache level
Interface before LLC

Provide 2 cache levels
Interface after L1

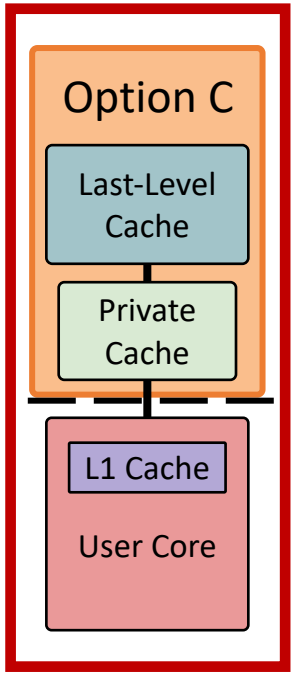


OpenPiton

# Core Interface Options

- Several interface options with different trade-offs
- We want to minimise effort to connect new cores
- We define the Transaction-Response Interface (TRI)
  - Simple, extensible, low overhead

| Core to Cache | Cache to Core | Subset |
|---|---|---|
| Load | Load Response | Load/Store |
| Store | Store Ack | Load/Store |
| Instruction Miss | Instruction Return | Instruction |
| | Invalidation Request | Invalidation |
| Atomic | Atomic Response | Atomic |
| Outbound Interrupt | Inbound Interrupt | Interrupt |





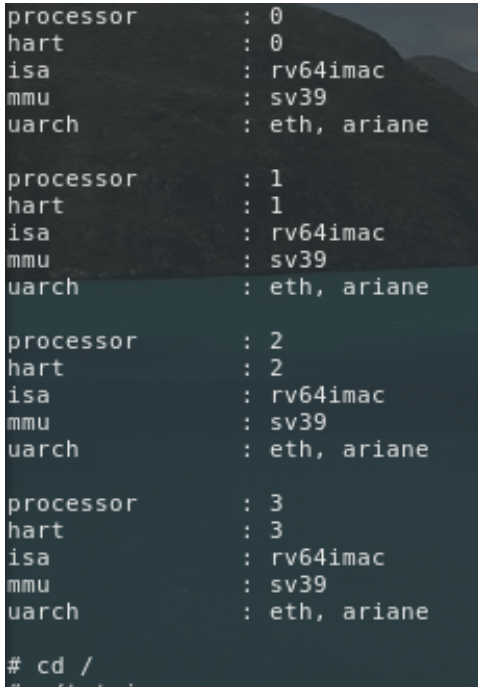

OpenPiton

# Cores Connected With TRI Subsets

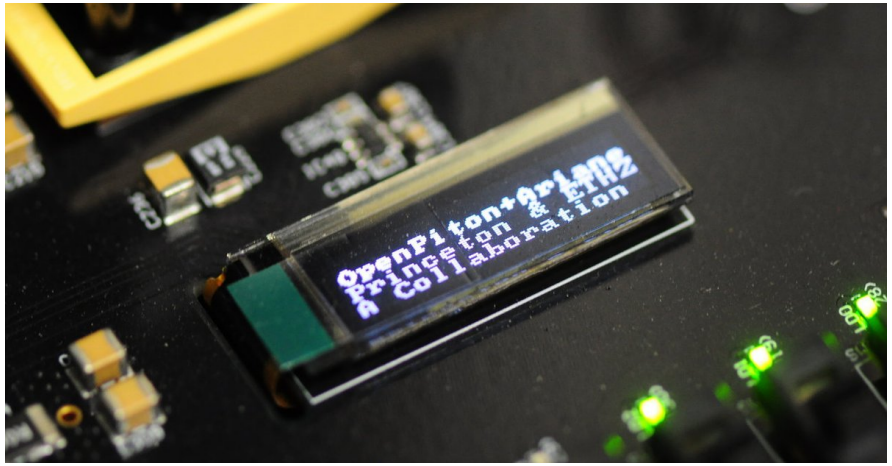
- OpenSPARC T1 (SPARC V9)
  - All subsets
- Ariane (RISC-V 64)
  - All subsets
- BlackParrot (RISC-V 64)
  - All subsets
- PicoRV32 (RISC-V 32)
  - Load/store + Atomic
- ao486 (x86)
  - Instruction + Load/store
- QEMU (Many ISAs)
  - Load/store
- VexRiscv (RISC-V 32)
  - Instruction
- Microwatt (OpenPOWER)
  - Instruction + Load/store*

| Core to Cache | Cache to Core | Subset |
|---|---|---|
| Load | Load Response | Load/Store |
| Store | Store Ack | Load/Store |
| Instruction Miss | Instruction Return | Instruction |
| | Invalidation Request | Invalidation |
| Atomic | Atomic Response | Atomic |
| Outbound Interrupt | Inbound Interrupt | Interrupt |

# OpenPiton + Ariane



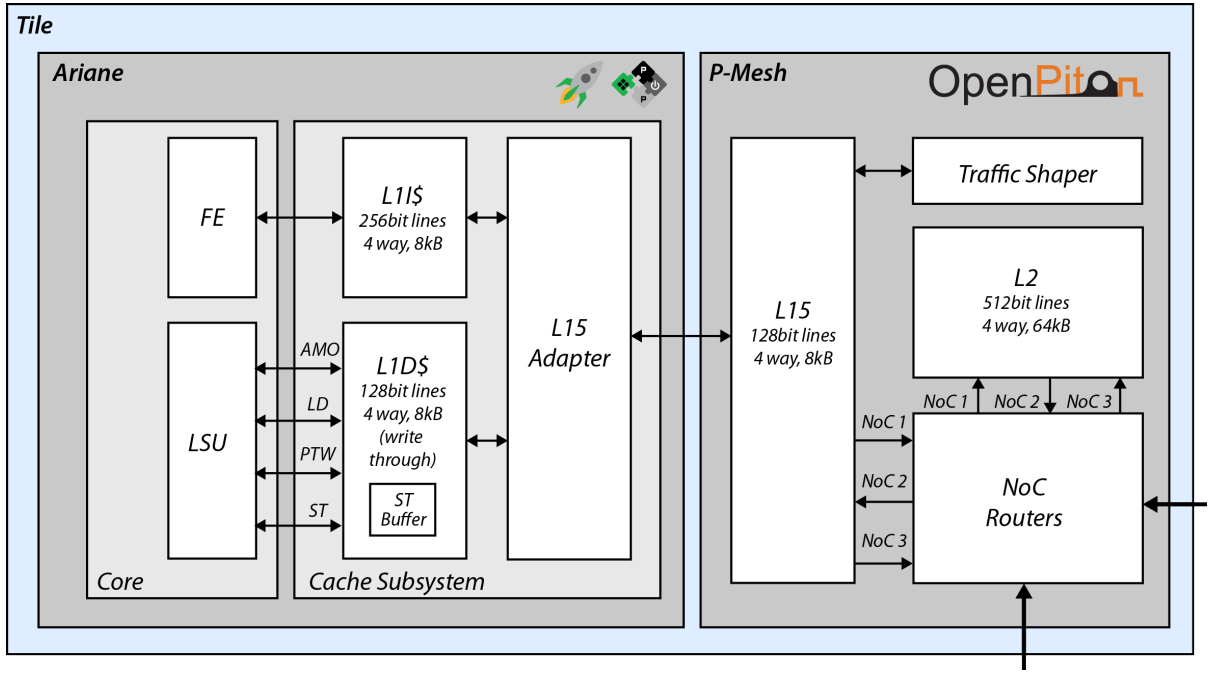

- Collaboration between Princeton University and ETH Zürich
- Goal: to develop a permissively licensed, Linux capable manycore research platform based on RISC-V
  - Based on mature, extensible designs
  - Booted SMP Linux in <6 months
- Ariane
  - RV64GC Core (with extensions)
  - Linux capable
- OpenPiton
  - Manycore research platform
  - P-Mesh cache coherence and NoC






OpenPiton

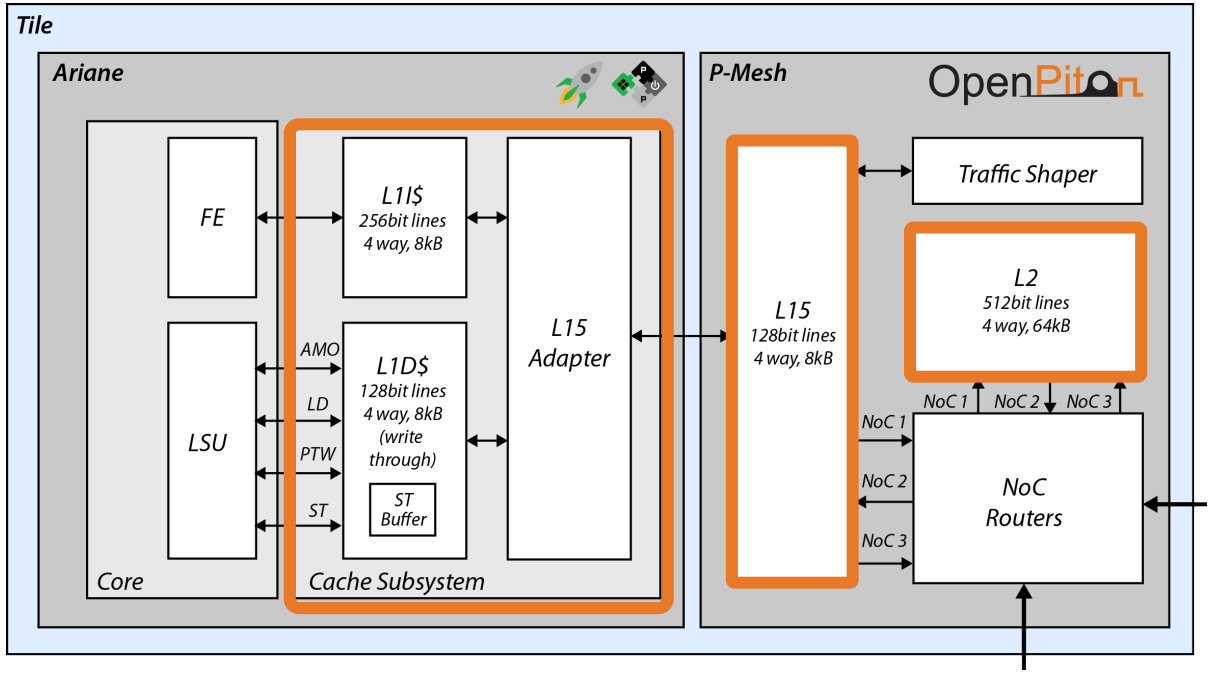
# OpenPiton+Ariane Tile






OpenPiton

# OpenPiton+Ariane Cache Modifications

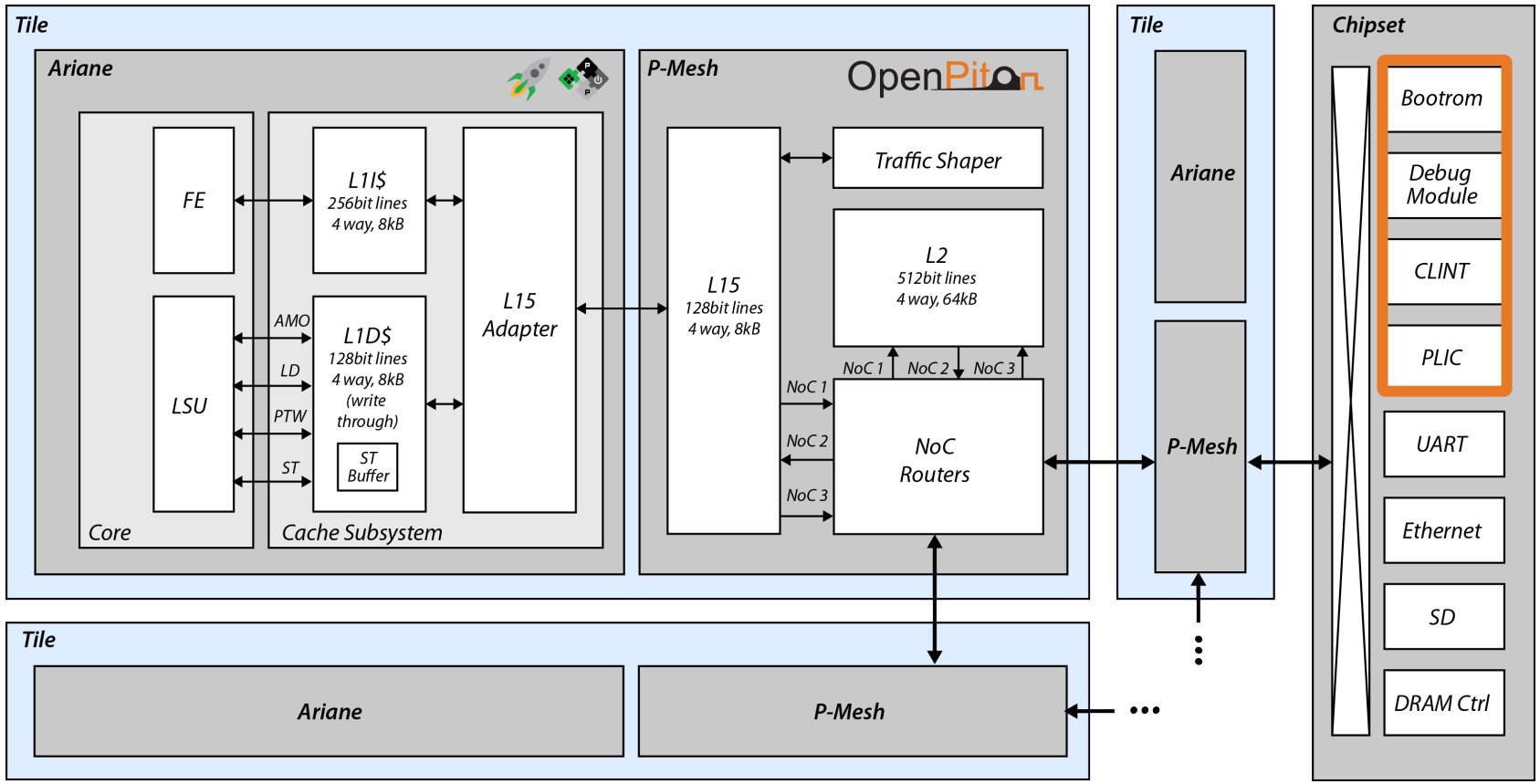



- New write-through cache subsystem with invalidations and the TRI interface
- LR/SC in L1.5 cache
- Fetch-and-op in L2 cache



OpenPiton

# OpenPiton+Ariane Platform Support



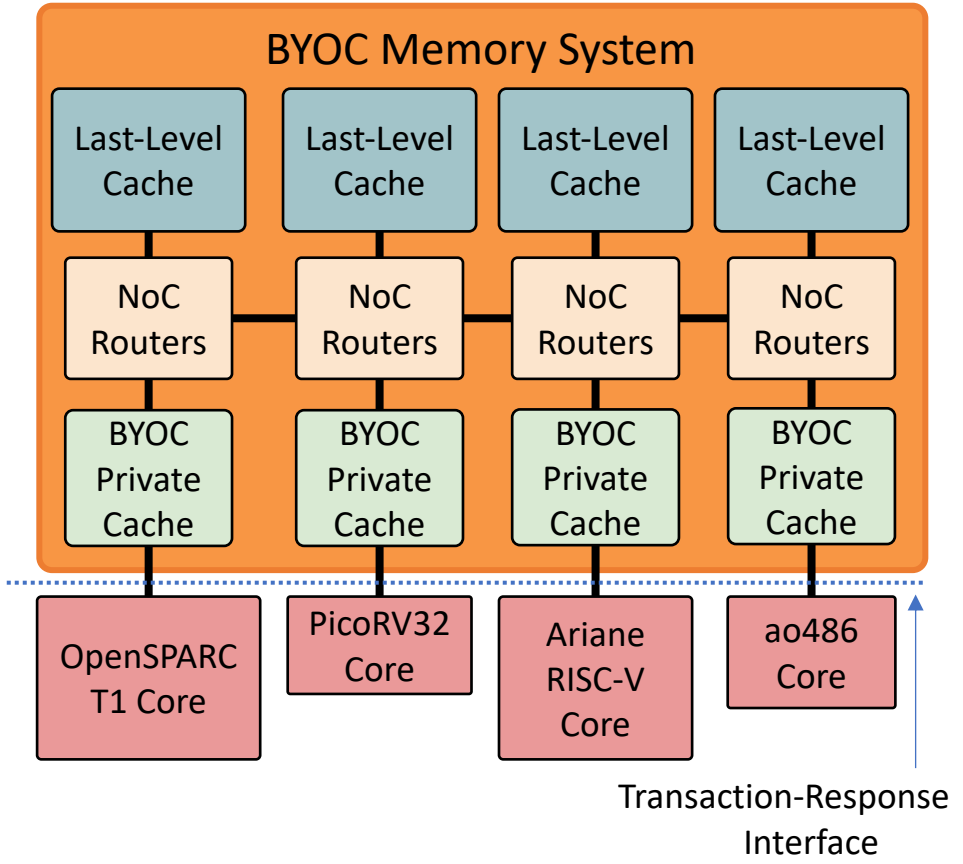

- Bootrom auto-generated with device tree from configuration
- RISC-V Debug
  - OpenOCD + GDB
  - Bootloading
- CLINT
- PLIC
  - lowRISC rv_plic



OpenPiton

# BYOC Heterogeneous-ISA System


- Innovations:
- We connect all cores in one open-source system!
  - Previous systems only allow cores of one ISA/microarchitecture
- Open-source, cache coherent memory system explicitly for heterogeneous ISA
- Core interface to cache coherence (TRI)
- First open-source, general-purpose, heterogeneous-ISA processor

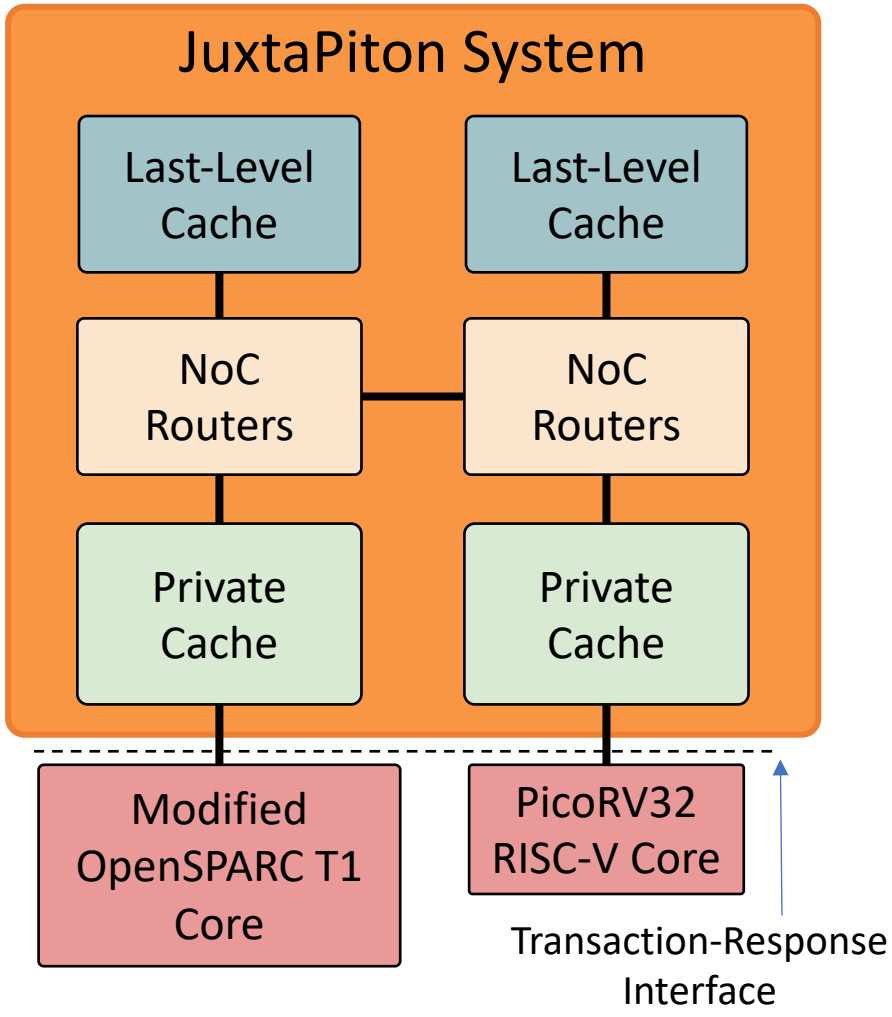





OpenPiton

# Proof of Concept: JuxtaPiton

- World's first open-source, general-purpose, heterogeneous-ISA processor
- 64 bit OpenSPARC T1 and 32 bit PicoRV32 cores coherently share memory
- OpenSPARC T1 offloads 32 bit RISC-V binaries to low-power PicoRV32 core
  - Syscalls proxied by OpenSPARC core
- Open-source as part of OpenPiton






OpenPiton

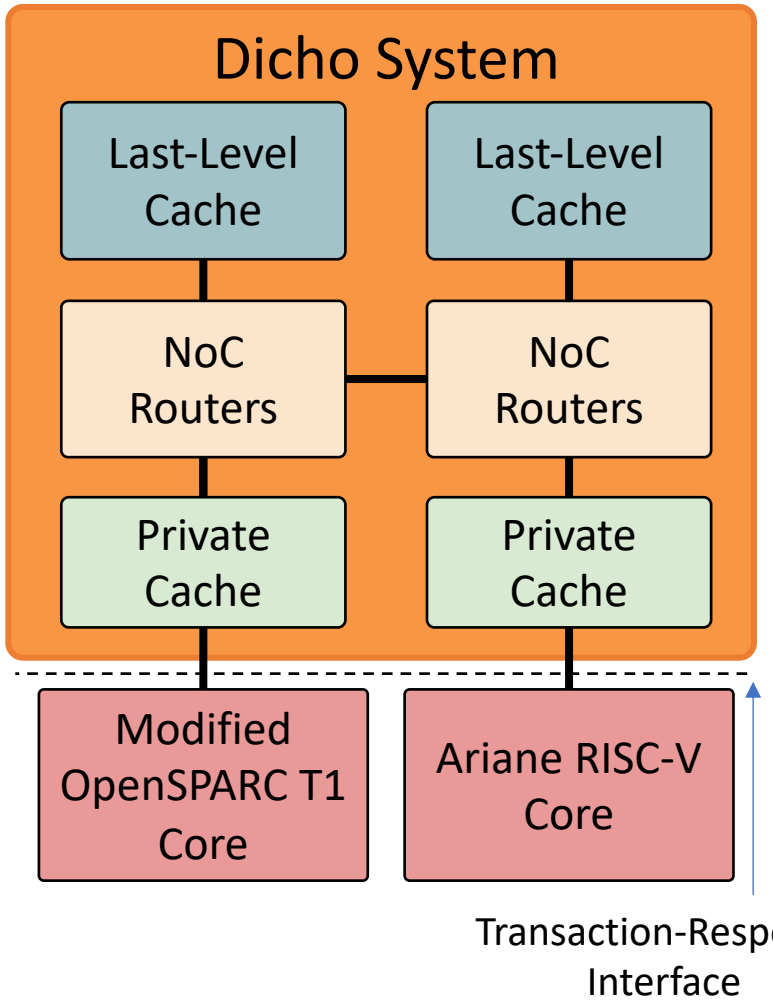
# Proof of Concept: Dicho

- 64 bit OpenSPARC T1 and 64 bit Ariane cores
- Fully hardware-coherent shared memory
  - Includes SPARC and RISC-V atomics
- Each core boots independent Linux kernel (SPARC + RISC-V)
- User-mode Linux applications share memory across the two cores/ISAs
- Artifact DOI:

https://doi.org/10.5281/zenodo.3563256

- Prototype available on GitHub:

https://github.com/bring-your-own-core/byoc







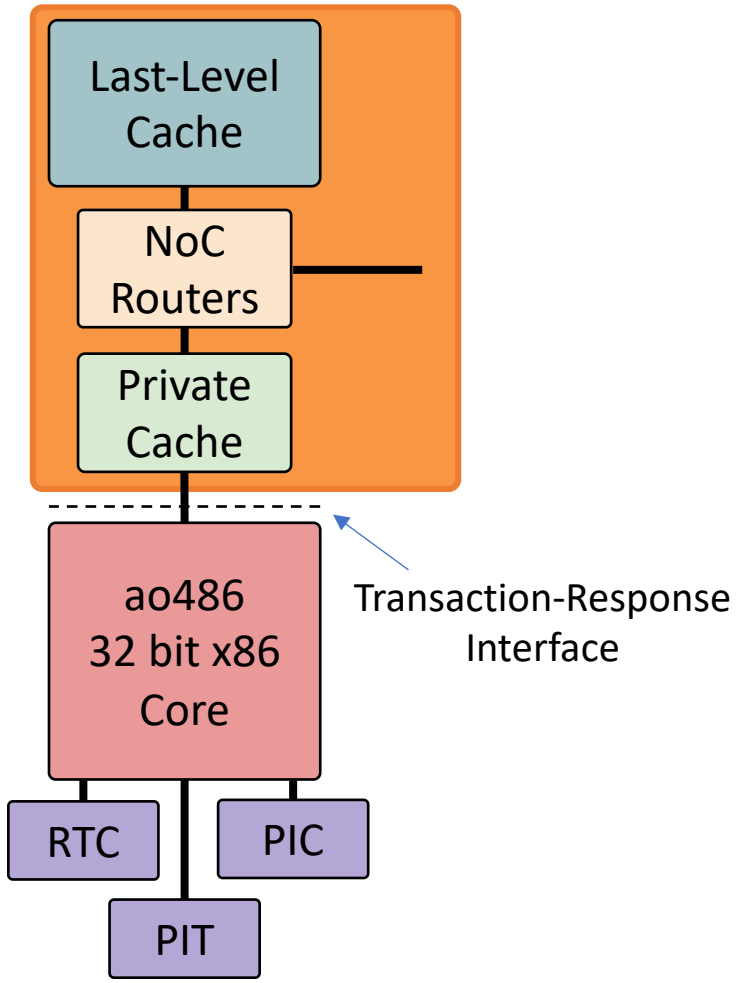



OpenPiton

# x86 in BYOC

- ao486 core implements 486 SX
  - Original SoC booted Windows 98, Linux 3.x
- Transduced Avalon memory and memory-mapped I/O operations to TRI
- Transduced I/O interface to legacy peripherals
- In BYOC, runs Bochs BIOS with simple OS on top
- Available on GitHub:

https://github.com/bring-your-own-core/byoc

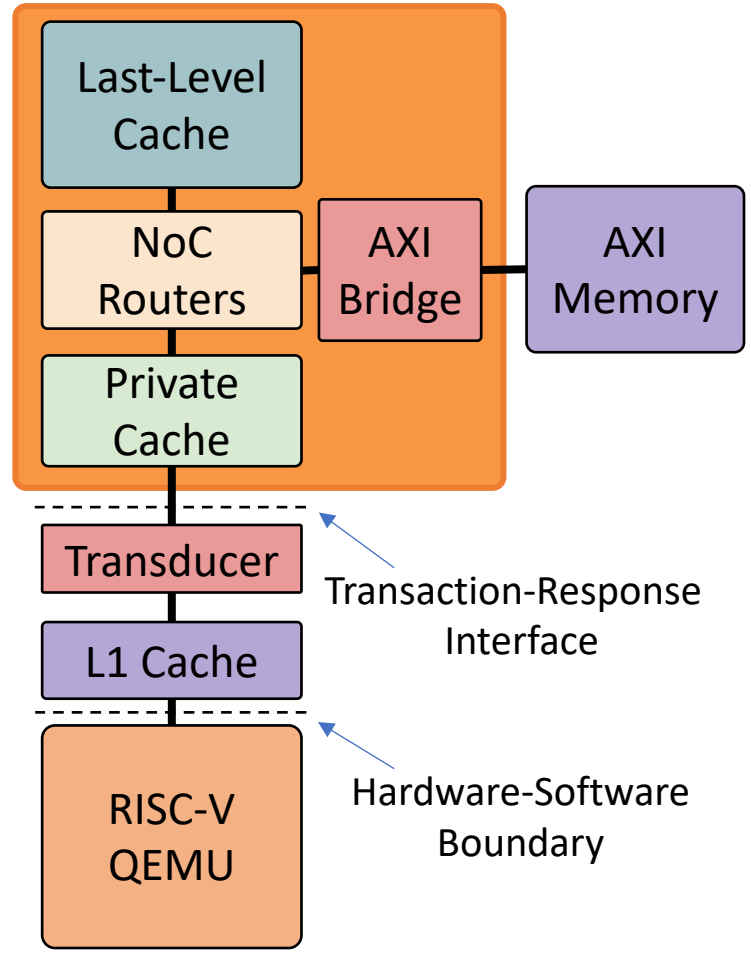





OpenPiton

# QEMU+BYOC

- Connected LibSystemCTLM-SoC to P-Mesh
- RISC-V QEMU (software) loads/stores to L1 cache in SystemC (hardware simulation)
- L1 cache transduced to TRI
- Potential: QEMU supports many ISAs with no open-source cores (e.g. x86-64, ARM)

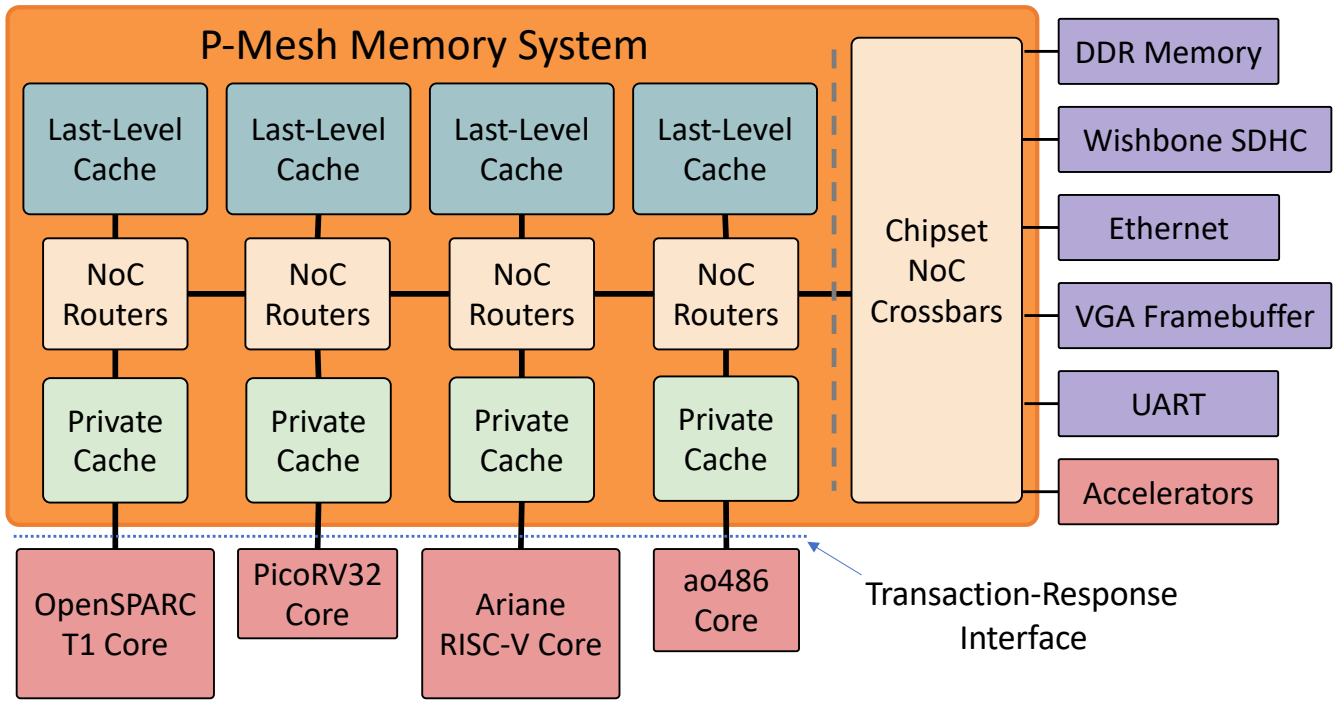





OpenPiton

# Building Heterogeneous Manycore SoCs

- OpenPiton/BYOC gives P-Mesh features to all cores, regardless of ISA
- We built an ISA-agnostic Hardware Transactional Memory system in P-Mesh
- Ongoing improvements:
  - Full-system Microwatt support!
  - Distributed interrupt controllers
  - More standard bridges
  - Core clusters per-tile






OpenPiton

# Support








This material is based on research sponsored by the NSF under Grants No. CNS-1823222, CCF-1217553, CCF1453112, CCF-1823032, and CCF-1438980, AFOSR under Grant No. FA9550-14-1-0148, Air Force Research Laboratory (AFRL) and Defense Advanced Research Projects Agency (DARPA) under agreement No. FA8650-18-2-7846, FA8650-18-2-7852, and FA8650-18-2-7862 and DARPA under Grants No. N66001-14-1-4040 and HR0011-13-2-0005. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Air Force Research Laboratory (AFRL) and Defense Advanced Research Projects Agency (DARPA), the NSF, AFOSR, or the U.S. Government.

# Enhancing the Open-Source P-Mesh Cache Coherence System for Open ISAs

Jonathan Balkind (@jbalkind)

Princeton University



OpenPiton