北京开源芯片研究院
BEIJING INSTITUTE OF OPEN SOURCE CHIP

Open-Source at BOSC: Achievements and Challenges

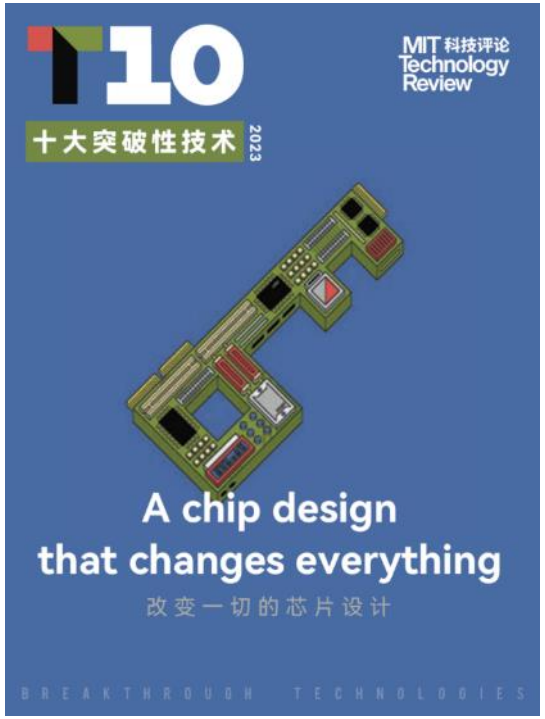
Yungang Bao

June 26, 2024

RISC-V: A new wave of revolution in chip design

- **One of 10 Breakthrough Technologies 2023 by MIT Technology Review**

A chip design that changes everything: "Computer chip designs are expensive and hard to license. That's all about to change thanks to the popular open standard known as RISC-V."

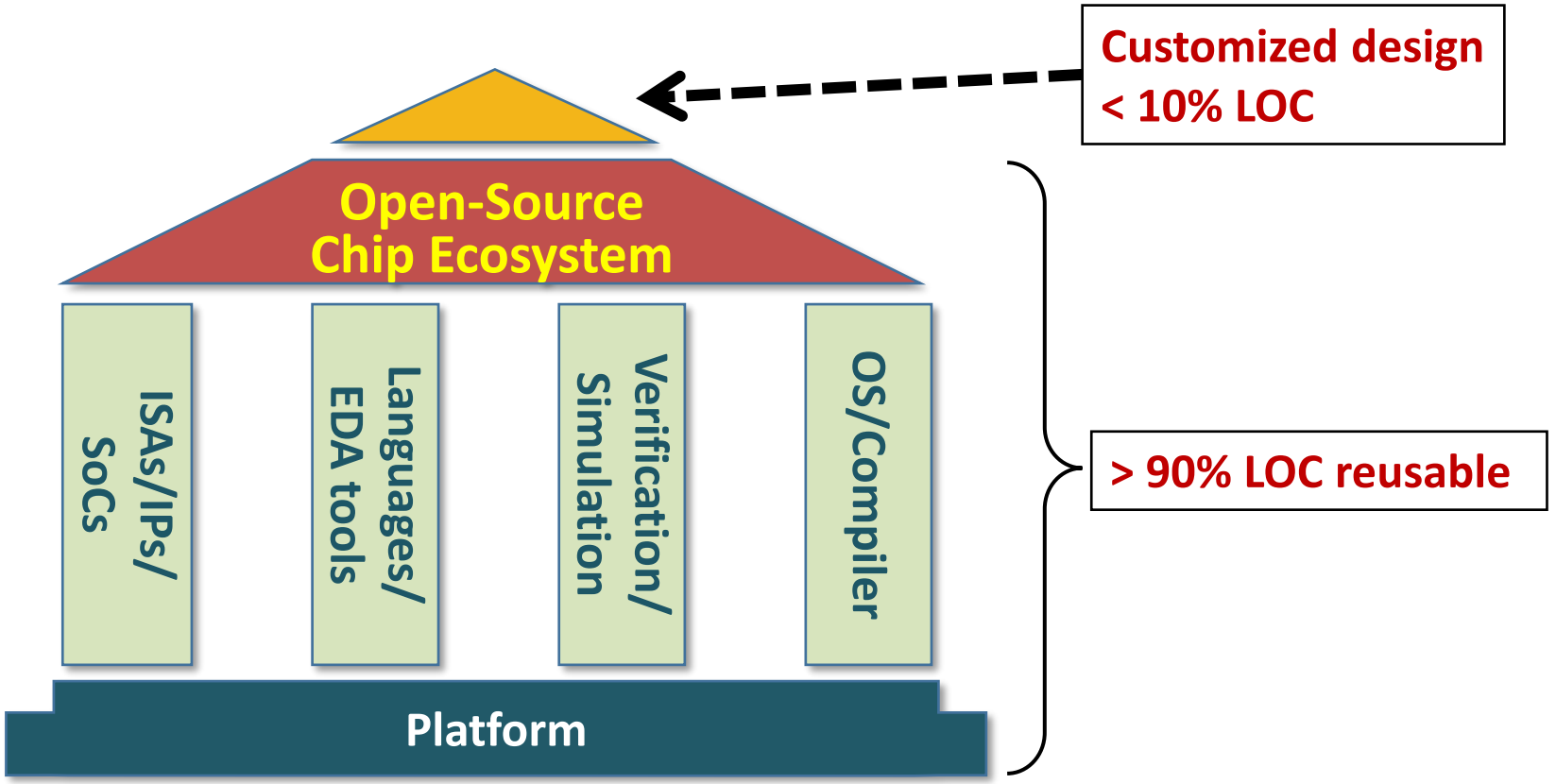


《MIT Technology Review》10 Breakthrough Technologies 2023:

(1) James Webb Space Telescope
(2) CRISPR for high cholesterol
(3) AI that makes images
(4) Organs on demand
(5) Abortion pills via telemedicine
(6) A chip design that changes everything
(7) Ancient DNA analysis
(8) Battery recycling
(9) The inevitable EV
(10) Mass-market military drones

Open-Source Chip Ecosystem (OSCE)

- **Lower the barrier of chip development by saving time-to-market** and the **cost of IPs, EDA tools and engineers**



Yungang Bao, The Four Steps to An Open-Source Chip Design Ecosystem, ACM SIGARCH Visioning Workshop, June 2019

BOSC's Mission

- In 2021, Beijing Institute of Open Source Chip (**BOSC**) was founded by **16 companies**
- BOSC's mission is to promote the development of OSCE by playing the role of the **bridge** between the academy and the industry

BOSC focuses on L2 & L3 of OSCE

3 Open-source Tools

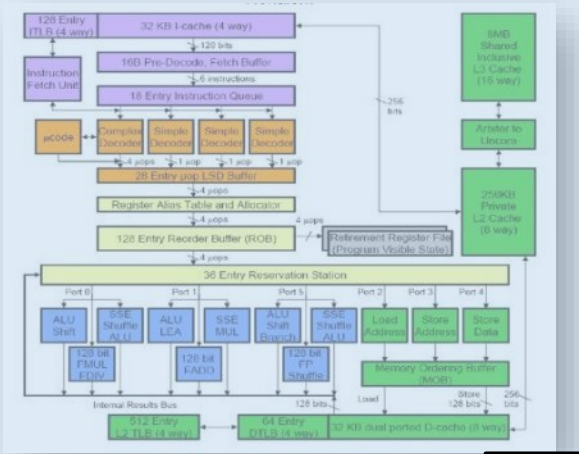
Microarch. → Engineering → EDA Tools

ISA Spec.

RISC-V®

1 Open ISA

Docs

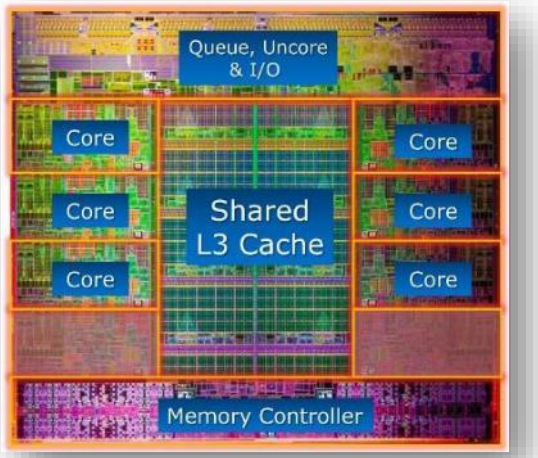


RTL codes

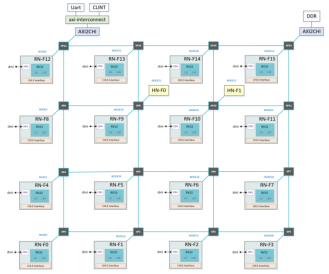
```
component DebugCoreTop is
  port (
    -- Trigger and Data
    cu_Clk      : in    std_logic_vector(2 downto 0) := (others => '0');
    cu0_Trig    : in    t_trig_0 := (others => (others => '0'));
    cu1_Trig    : in    t_trig_1 := (others => (others => '0'));
    cu2_Trig    : in    t_trig_2 := (others => (others => '0'));
    cu0_Data    : in    t_data_0 := (others => (others => '0'));
    cu1_Data    : in    t_data_1 := (others => (others => '0'));
    cu2_Data    : in    t_data_2 := (others => (others => '0'));

    -- Downstream I2C
    SCL         : in    std_logic := '0';
    SDA         : inout std_logic := '0';

    -- Upstream
    gt_RefClk_p : in    std_logic := '0';
    gt_RefClk_n : in    std_logic := '0';
    gt_RX_p     : in    std_logic_vector(2 downto 0) := (others => '0');
    gt_RX_n     : in    std_logic_vector(2 downto 0) := (others => '0');
    gt_TX_p     : out   std_logic_vector(2 downto 0);
    gt_TX_n     : out   std_logic_vector(2 downto 0)
  );
end component;
```

2 Open-source Design

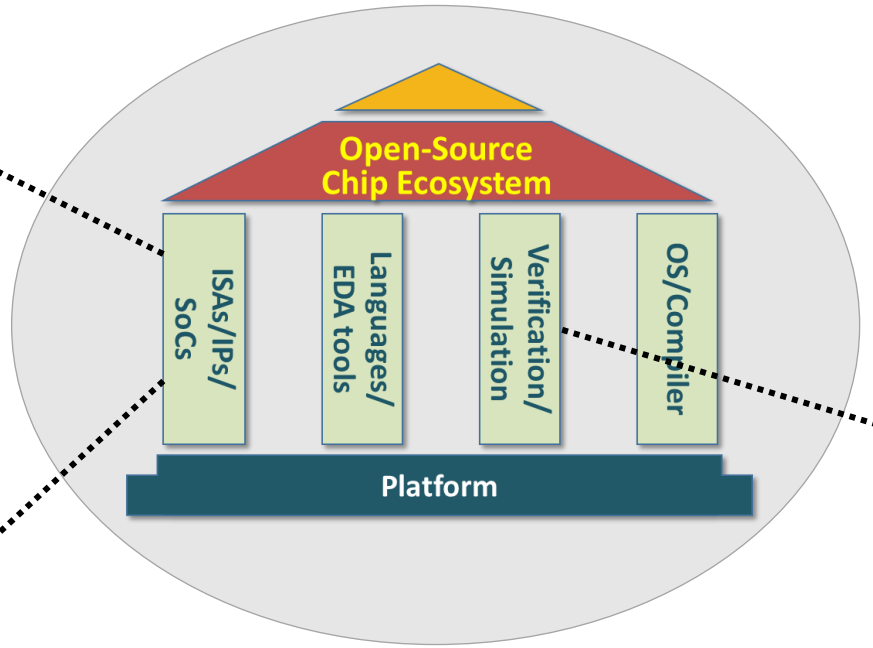
Layout

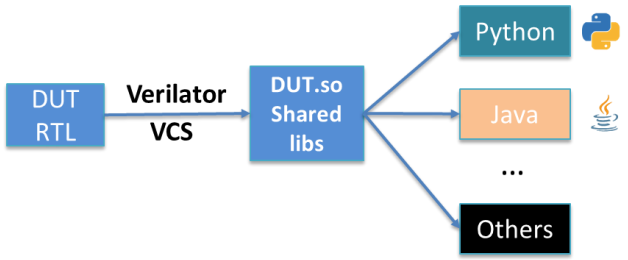


Open-source projects in BOSC



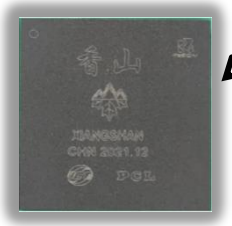
Open-source NoC





Open Verification Platform

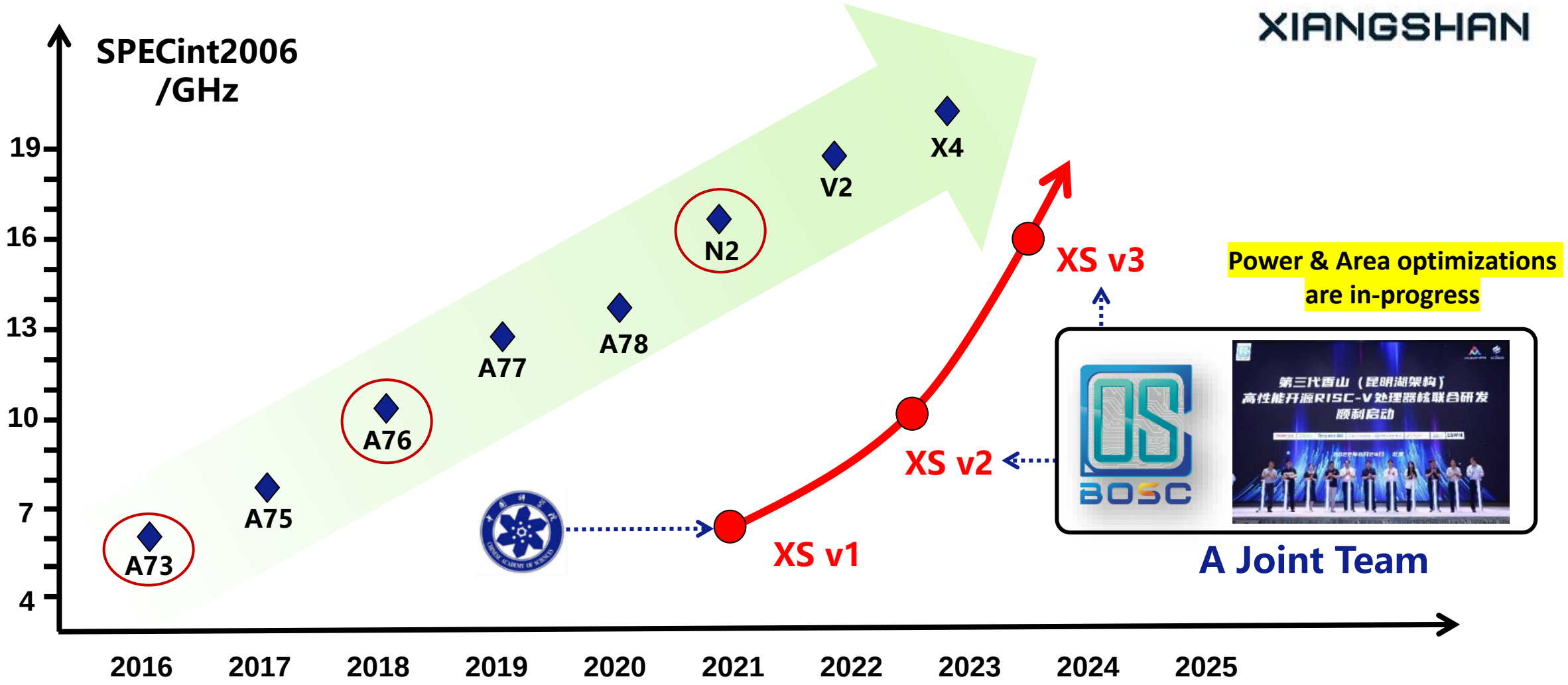
XIANGSHAN



Open-Source CPU Core

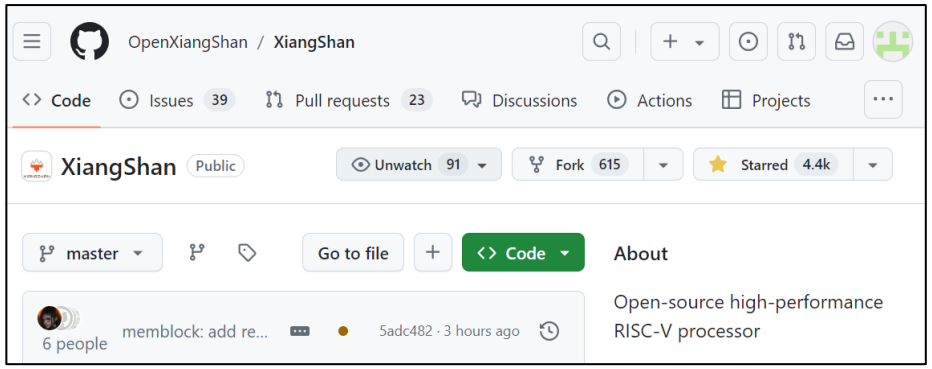
#1: Open Source Chip: The XiangShan Project

XIANGSHAN



One of the most active open-source chip projects

- **XiangShan** has been one of the most active open-source chip projects
  - **>4400** Stars, > **600** Forks

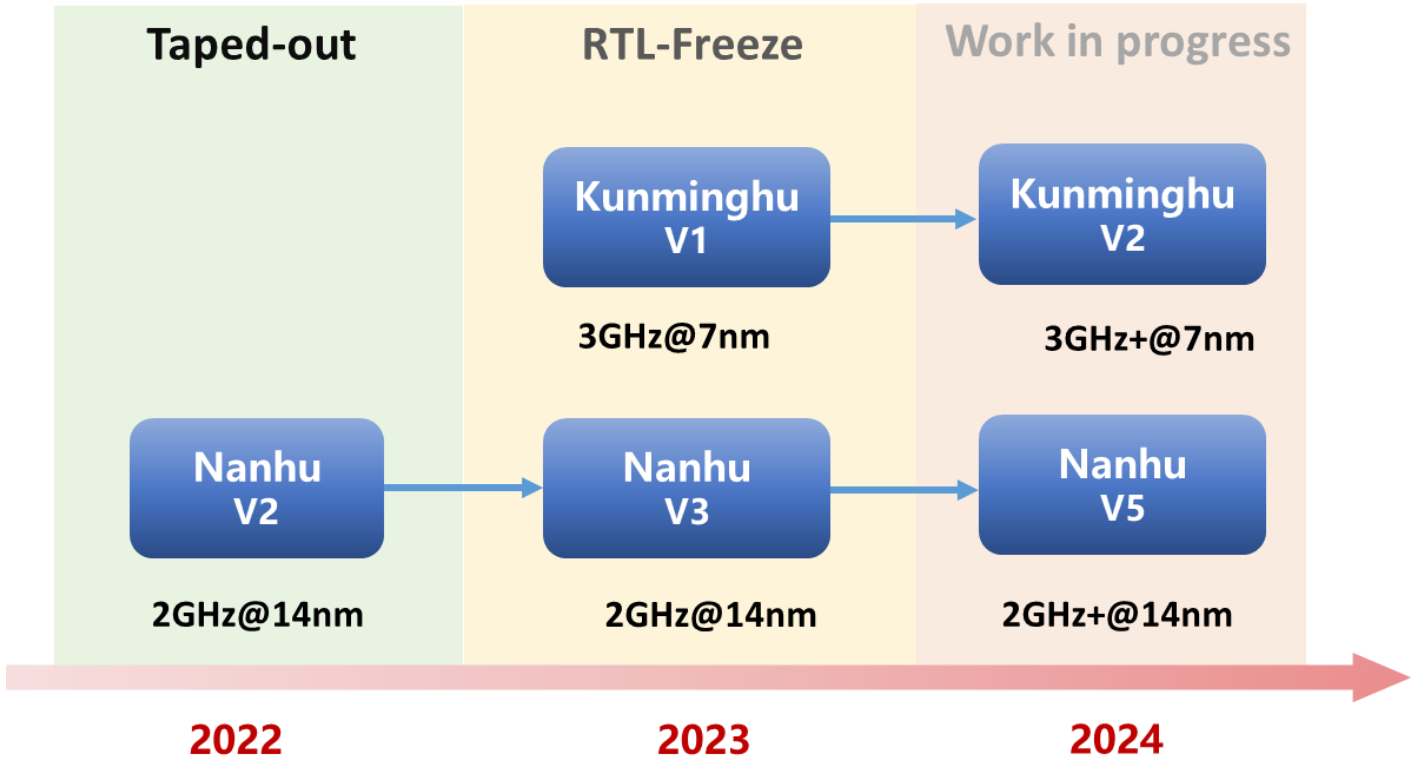


Data collected on June 16, 2024

CPU Core: Two Tiers Roadmap

Highlights

- Highly configurable with Chisel and agile development toolchain
- Industrial-grade µarch design and development workflow



XS v3 (Kunminghu) Architecture

- ➢ Designed for ultimate performance
- ➢ Targeting server/data-center segment

- RVA-23 profile compatible
- Leading RISC-V features (H/V ext.)

Versus ARM Neoverse N2

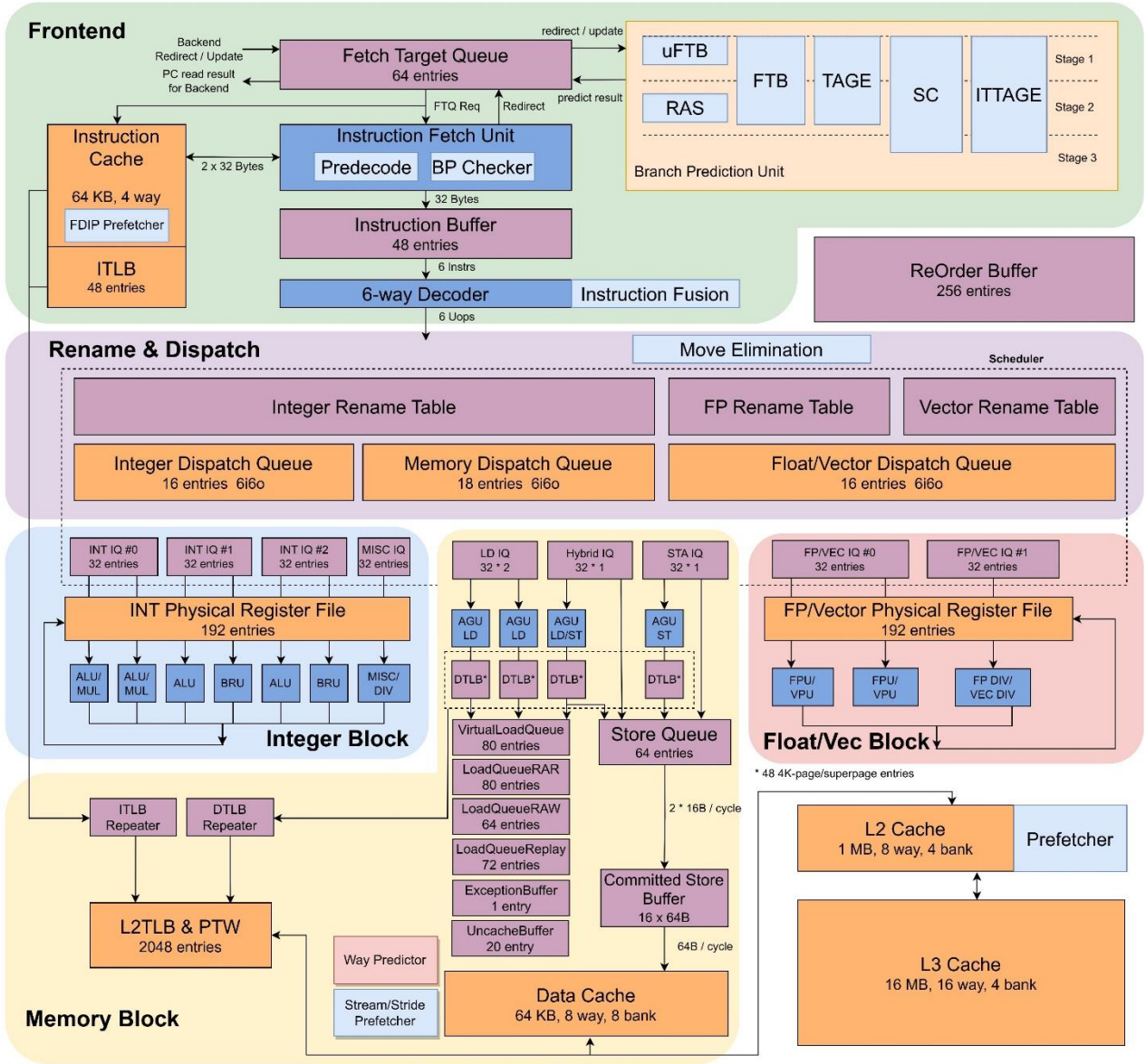
XS v2 (Nanhu) Architecture

- ➢ Designed for power/area efficiency
- ➢ Targeting industry-control segment

- RVA-20 profile compatible
- Already taped out and tested

Versus ARM Cortex A76

µArch Overview of XS v3 (Kunminghu) Architecture



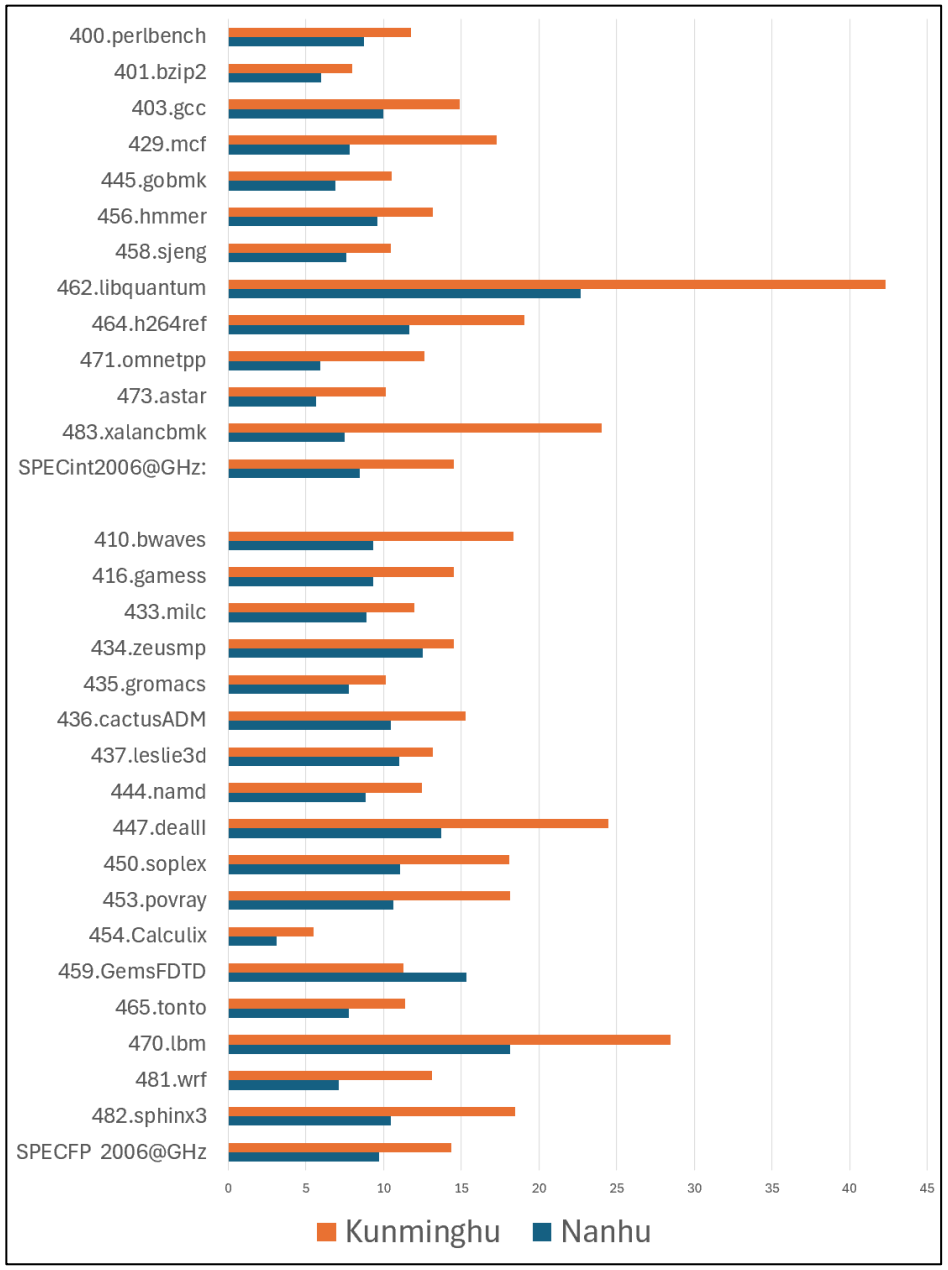
- **ISA extension support**
  - Vector extension
  - Hypervisor extension
- **Decoupled frontend**
  - Eliminate great fetch bubbles
  - Instruction prefetching friendly
- **Large outstanding instruction window**
- **Low latency & high BW $ access**
  - Closely-coupled load/store pipe & L1/L2$
  - Multi-level composite prefetchers
  - Banked architecture
- **Support CHI/TileLink coherence port**

Performance Evaluation

- **Methodology: RTL simulation using checkpoints selected via Simpoint**
  - Compiler: GCC 12 -O3, RV64GCB, jemalloc
  - CPU Config: 3GHz, 16MB L3
  - Memory modeled by DRAMsim3 DDR4@3200, 70ns latency
    - dual channel, x8, 4 bank groups
- **Evaluation results**

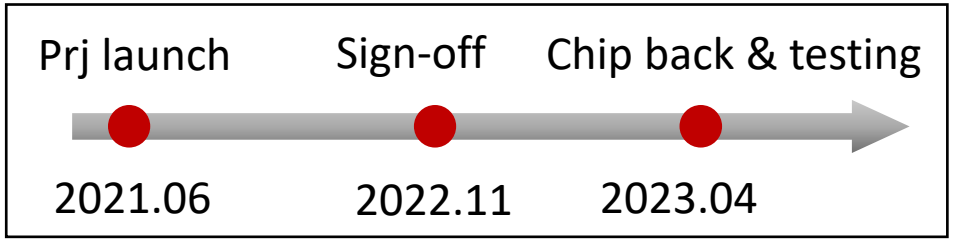
Base score	SPECint 2006	SPECfp 2006
Nanhu v2@2.5GHz	**25.41**	**29.13**
Kunminghu v1@3GHz	**44.98 (49.96*)**	**45.72**

* With compiler optimizations

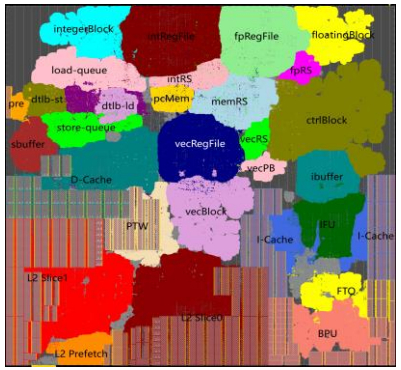


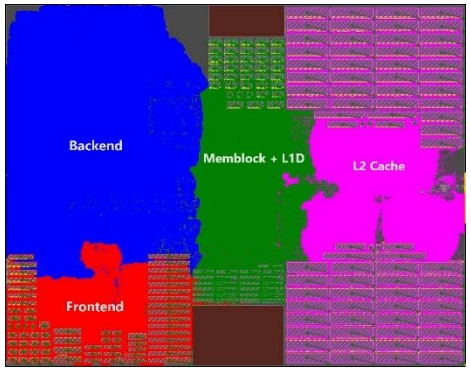
Tape-out Status

- **Nanhu v2 has been taped out**

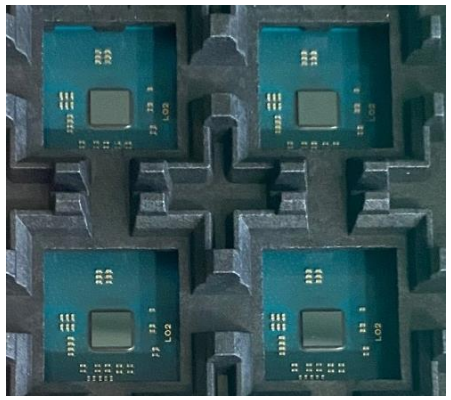


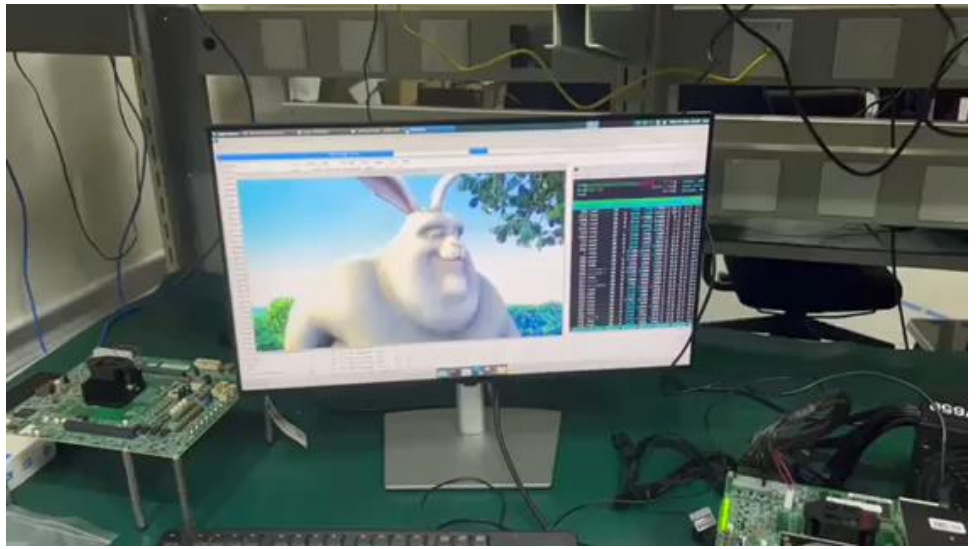
- **Nanhu v2 Chip Evaluation**
  - **2.5GHz**, SPECCPU06 **~10/GHz**
- **Ready to tape-out next**
  - XS Nanhu v3
  - XS Kunminghu v1





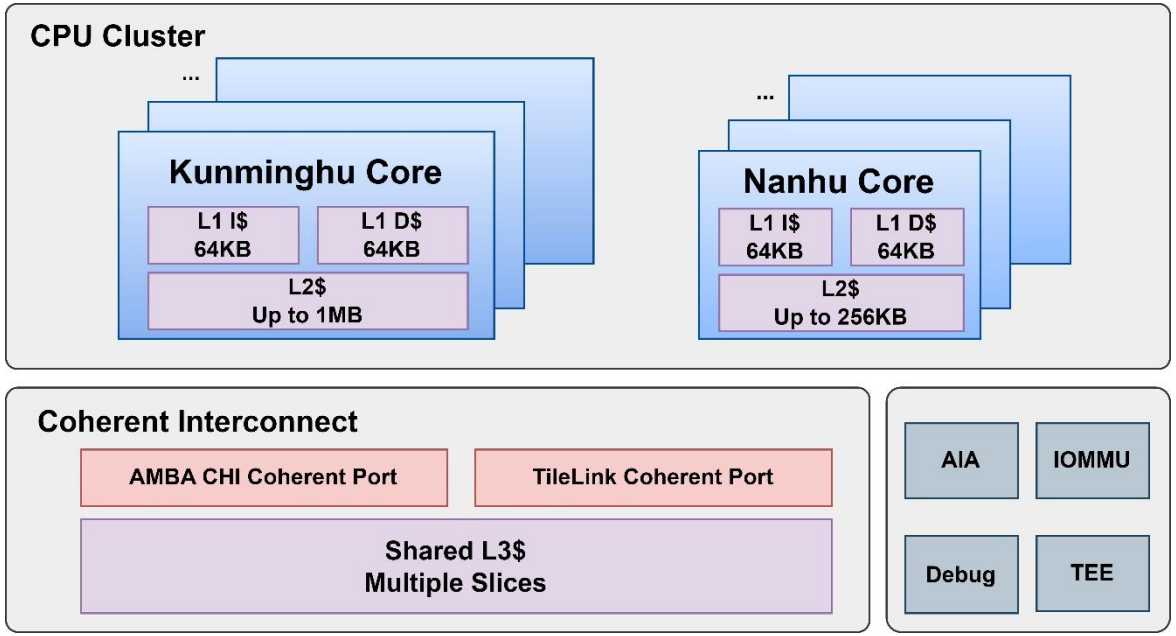
Floorplan of Nanhu v3 & Kunminghu v1 (single core)

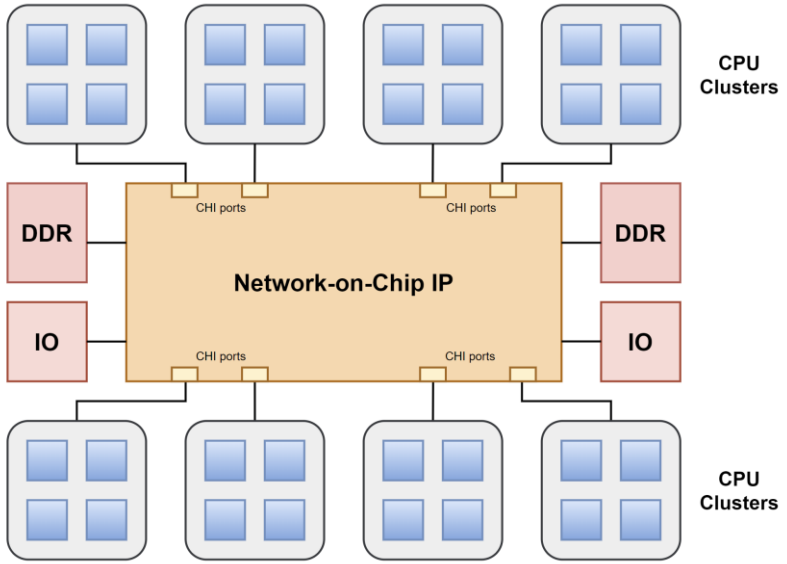




Nanhu v2 Test Chip Demo

#2: Open SoC Solutions



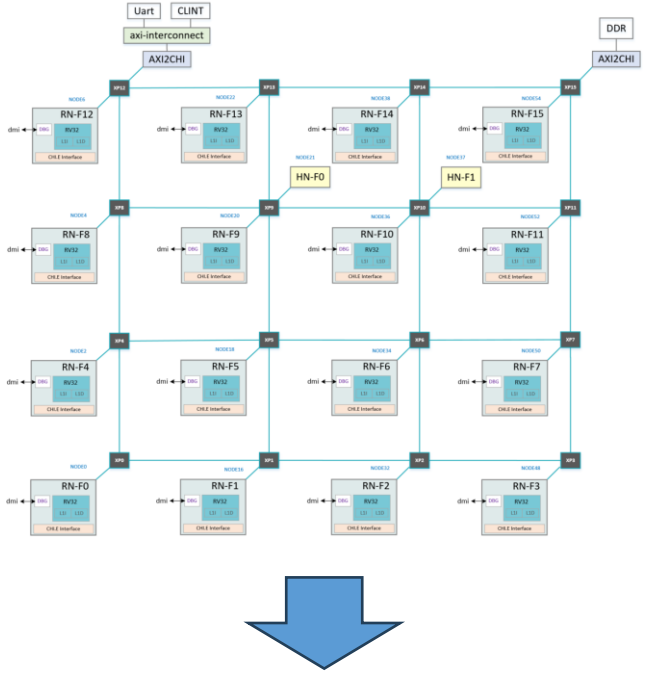


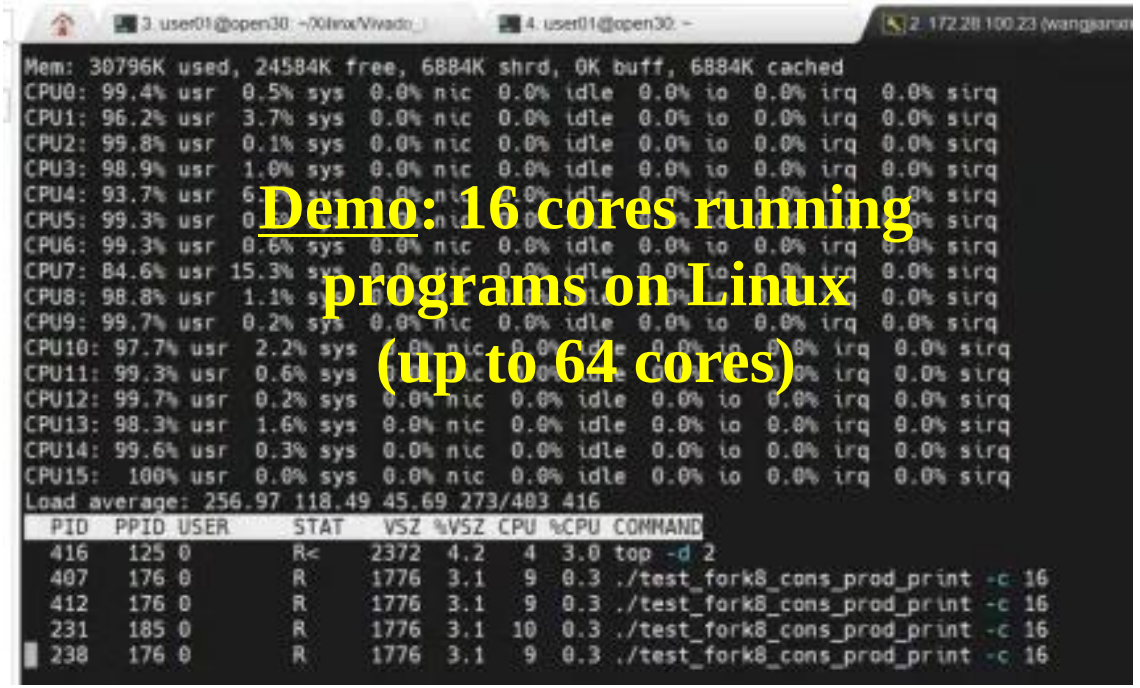
Support

- **RISC-V IOMMU Architecture**
- **RISC-V Advanced Interrupt Architecture**
  - MSI handling & interrupt virtualization
- **Proprietary TEE optimized for RISC-V**
- **AXI4/CHI/TileLink protocol**
- **Optional cluster-level shared L3 cache**
  - Directory based, Inclusive/non-inclusive
- **RISC-V Debug/Trace Architecture**
- **Ready for 3rd party NoC for current generation**
  - NoC for XiangShan work-in-progress

The OpenNoC Project

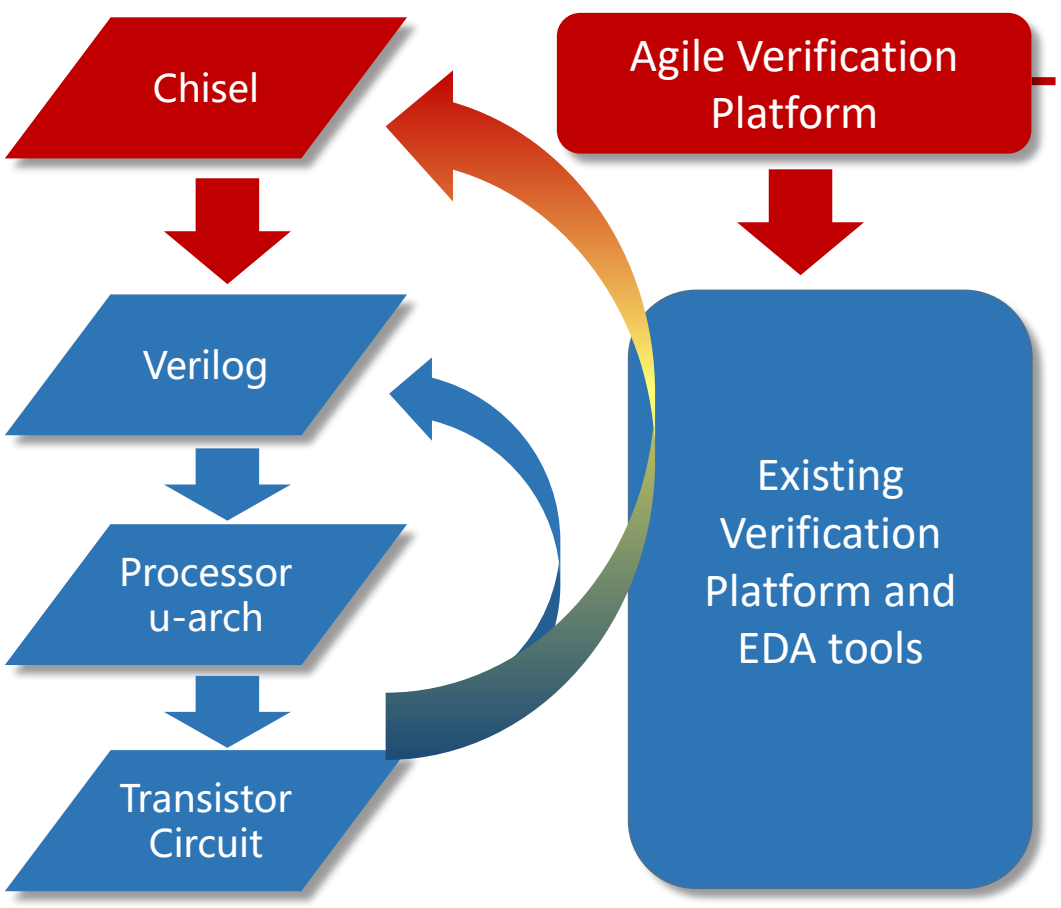
- Compliant with ARM AMBA 5 **CHI** 0050E.b
- Compliant with **MESI** Protocol Cache Coherence
- Use Non-Inclusive, Non-Exclusive policy
- Support **Snoop Filter**
- Support end-to-end QoS
- Support up to an **8×8** Mesh network
- Support up to **128** processors, accelerators, graphic processing units, etc.
- Support up to **16** internal memory controllers
- **256-bit** data channel
- Support up to **16** HN-Fs, with each HN-F's LLC size ranging from 0 to **32MB**
- Maximum Snoop Filter size is **8MB**
- Supports a maximum of **128 RNFs**





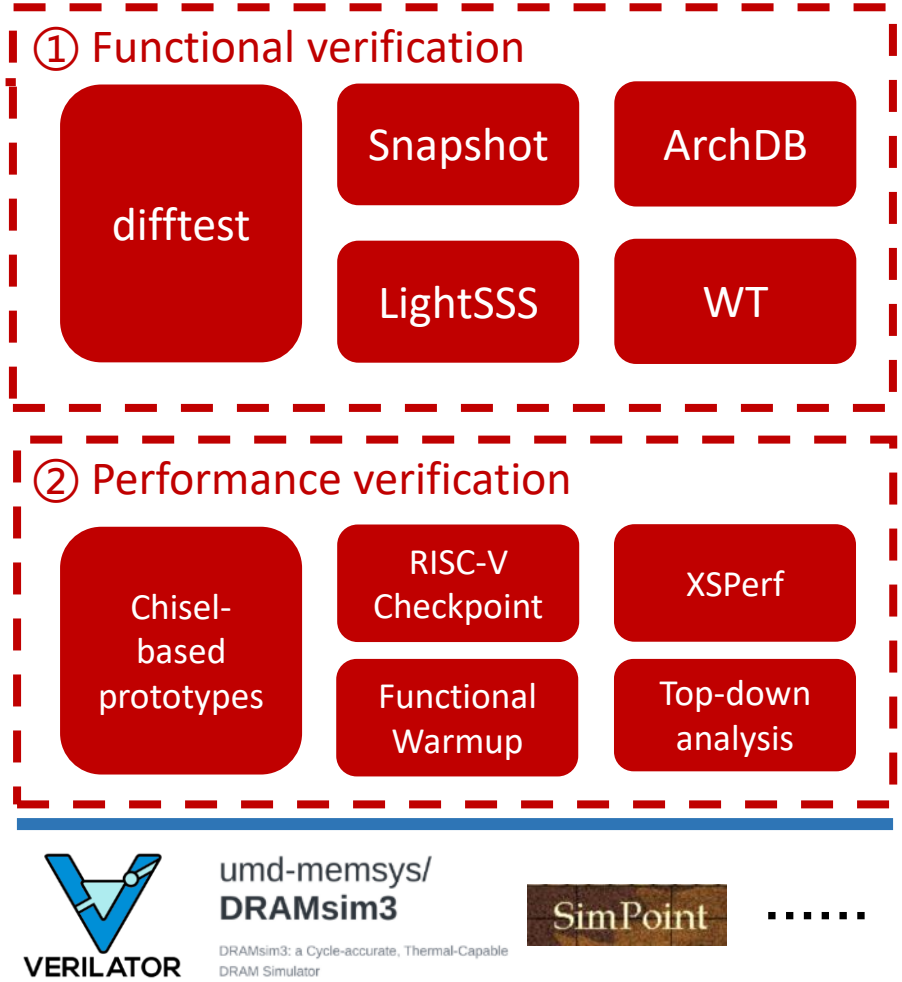
Demo: 16 cores running programs on Linux (up to 64 cores)

#3: The Open Verification Platform



Design Flow

Verification Flow



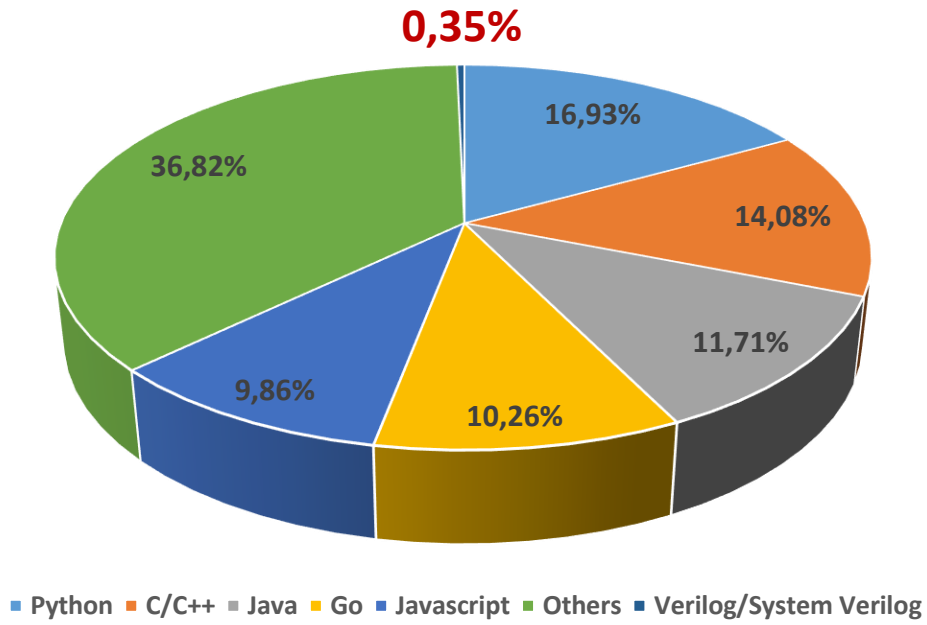
Agile hardware verification tools

Challenges in Chip Verification

- There are orders of magnitude fewer HW engineers than SW engineers

Survey

- Pull Requests (PRs) on GitHub related to **Verilog/SV** account for **0.35%**
  - **Python**: 16.93%
  - **C/C++:** 14.08
  - **Java**: 11.71%
  - **Go**: 10.25%

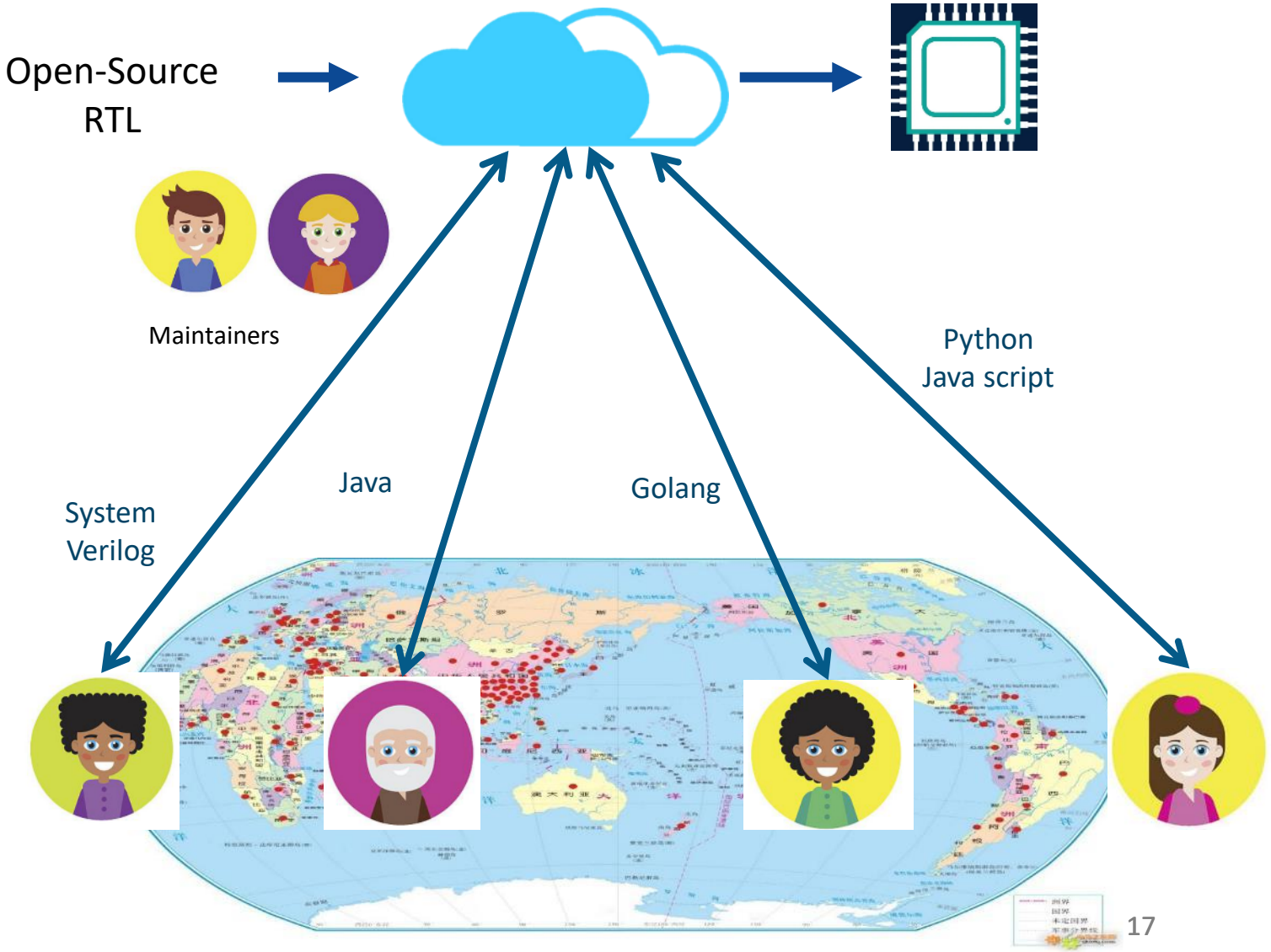


https://madnight.github.io/githut/#/issues/2024/1

The Open Verification Platform：
A Cloud-based Verification Platform for Open-source Chip projects

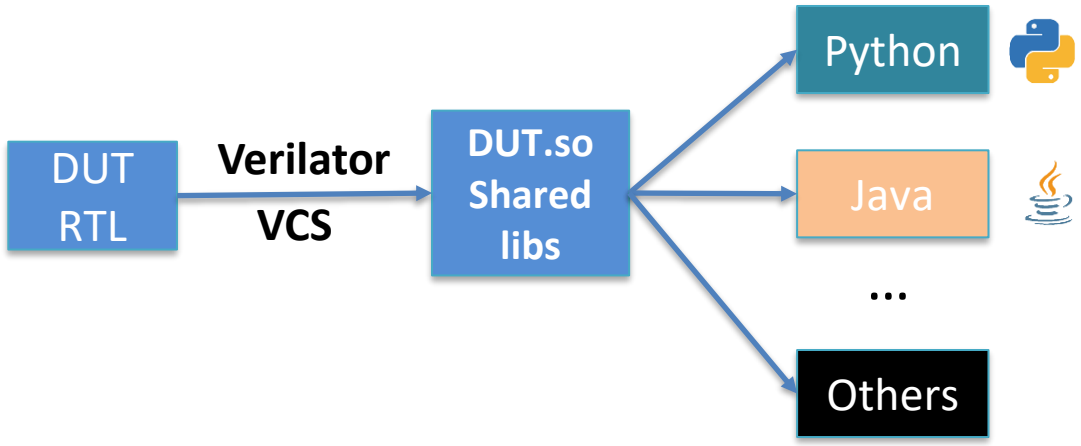
Features:

- Support crowdsourcing
- Based on cloud
- Support multi-languages
- Involve software engineers



Open Chip Verification with Multiple languages

- Convert RTL to Python, Java/Scala, C/C++, and Go
- Compatible with both software testing and chip verification environments




```
5  import UT_RAS as ras   # import RTL as python module (UT_RAS is generated by picker)
6  import ras_pins as p   # DUT wrapper functions (rest, pop, push, commit, redirect) based on pins
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
```

LCOV - code coverage report

		Hit	Total	Coverage
Current view:	top level			
Test:	merged.info	Lines:	1360	1541	88.3 %
Date:	2024-05-07 07:31:58	Functions:	0	0	-

Directory	Line Coverage			Functions	
out/picker_out_uFTB		87.8 %	1276 / 1454	-	0 / 0
rtl/common		100.0 %	15 / 15	-	0 / 0
rtl/uFTB		95.8 %	69 / 72	-	0 / 0

Generated by: LCOV version 1.14

Pytest test Code for a XiangShan module

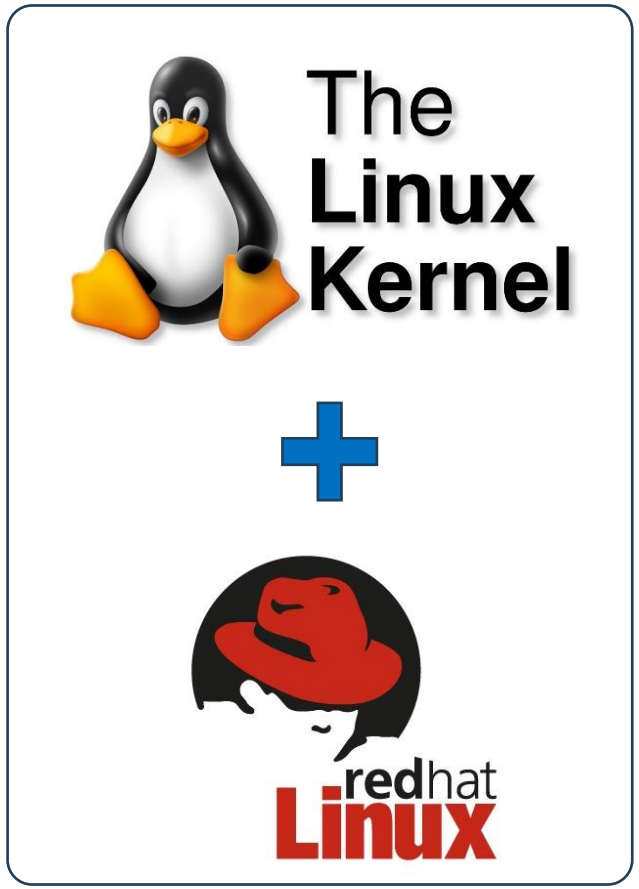
Tutorial Resources of BOSC's projects

- **Contents of tutorials**
  - Introduction to the microarchitecture and design concepts of XiangShan processor
  - Introduction to the infrastructures for XiangShan development
  - Hands-on development with typical use cases on XiangShan and MinJie

- **Open-access materials**
  - Presentations slides
    - https://github.com/OpenXiangShan/XiangShan-doc/tree/main/tutorial
  - Demonstration environment
    - https://github.com/OpenXiangShan/xs-env/releases
  - Demonstration videos (in Chinese)
    - https://www.bilibili.com/video/BV1c14y1s7LF
    - https://www.bilibili.com/video/BV1ph4y1T745

Collaboration opportunity: Linux + Redhat Mode

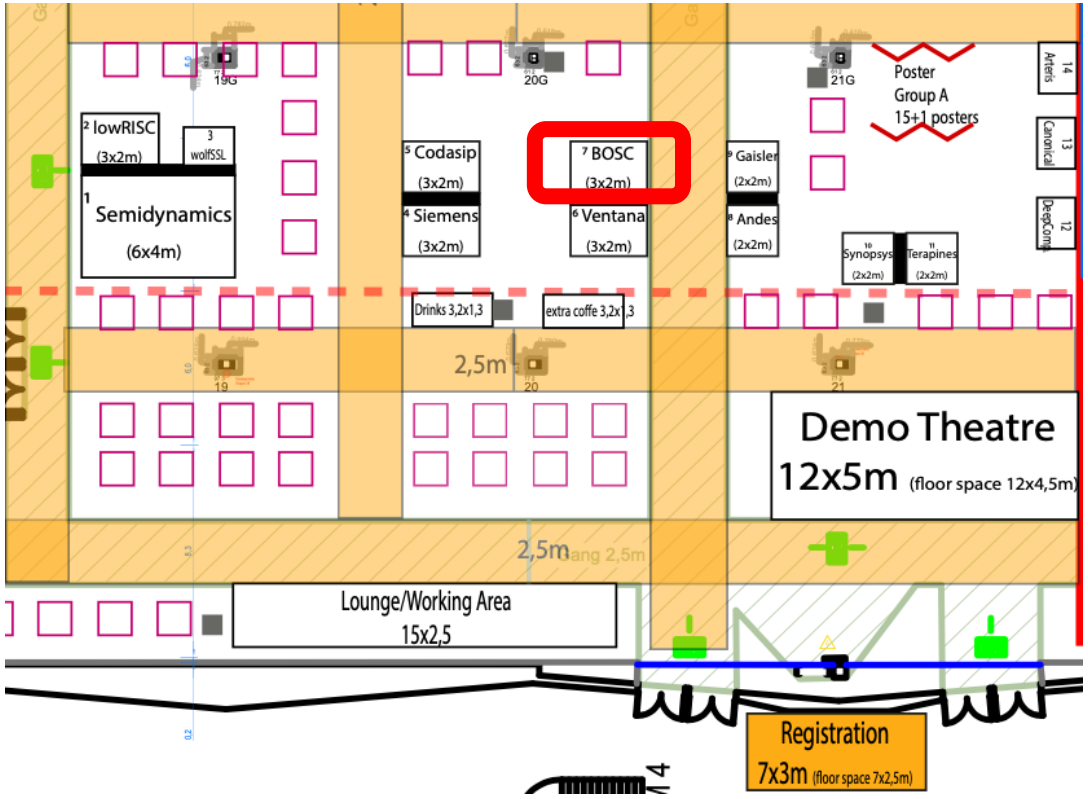
- Provide IPs and design service based on open-source RISC-V cores
- Reduce TCO by leveraging the efforts of open-source communities



V.S.

Welcome to BOSC Booth (No.7)

Booth No.7



7 Posters

1.XiangShan: Empowering Open-Source RISC-V Innovation with High Performance Processor and Agile Infrastructure

2.Calibrate GEM5 to Boost DevelopingXiangshan Processor

3.Open-Source at BOSC: Achievements and Challenges

4.Efficient Architecture Verification Framework with FPGA Acceleration for RISC-V Processors

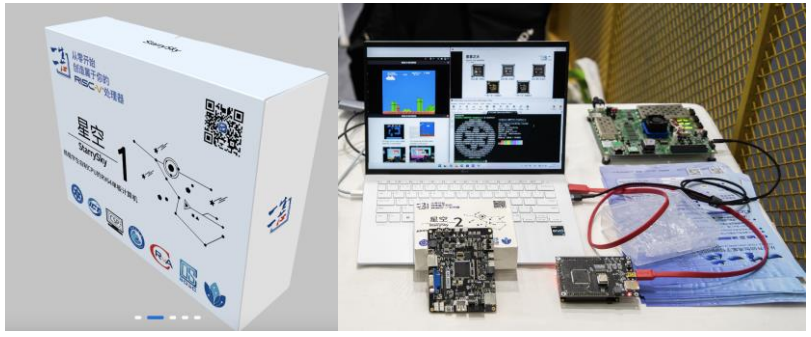
5.Efficient Verification Framework for RISC-V Instruction Extensions with FPGA Acceleration

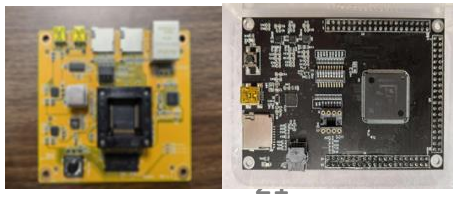
6."One Student One Chip" Initiative: Learn to Build RISC-V Chips from Scratch with MOOC （Booth Area）

7.Advancing SoC Design: A Groundbreaking Architecture for Minimizing Tape-Out Costs（Booth Area）

Dev Zone

SOC Designs from Student Participants of "One Student One Chip"






北京开源芯片研究院
BEIJING INSTITUTE OF OPEN SOURCE CHIP
Open Source Chip
Together for a Shared World

Thanks!