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

香山开源处理器社区
XiangShan Open-Source Processor Community



XSAI: 香山原生的通推一体 AICPU

王凯帆

北京开源芯片研究院

2026年10月18日 @RISC-V Summit China 2026

目录

一 XSAI：开发理念

二 XSAI：微架构设计

三 XSAI：编程范式与软件生态

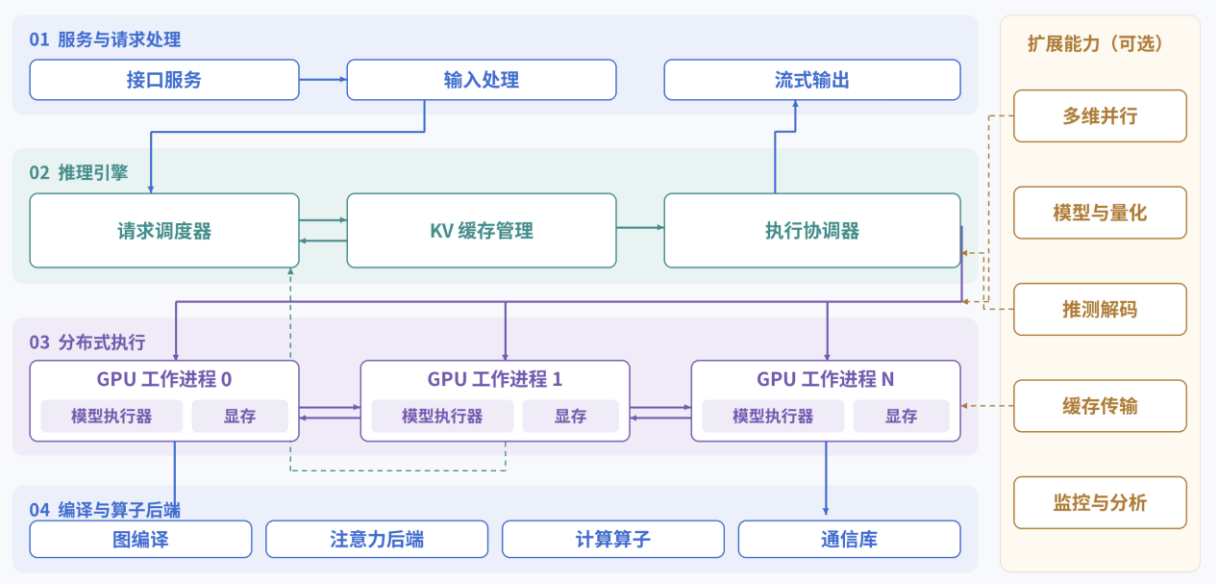
四 XSAI：基础设施

五 XSAI：设计评估

AI 不止于矩阵单元

趋势一

AI 推理负载正在变得复杂化、系统化

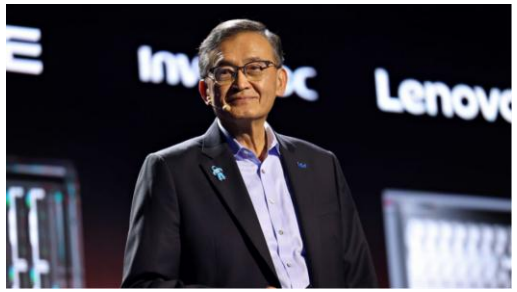


案例：vLLM 推理框架系统架构示意图

趋势二

Agentic AI 场景下 CPU 越来越成为瓶颈

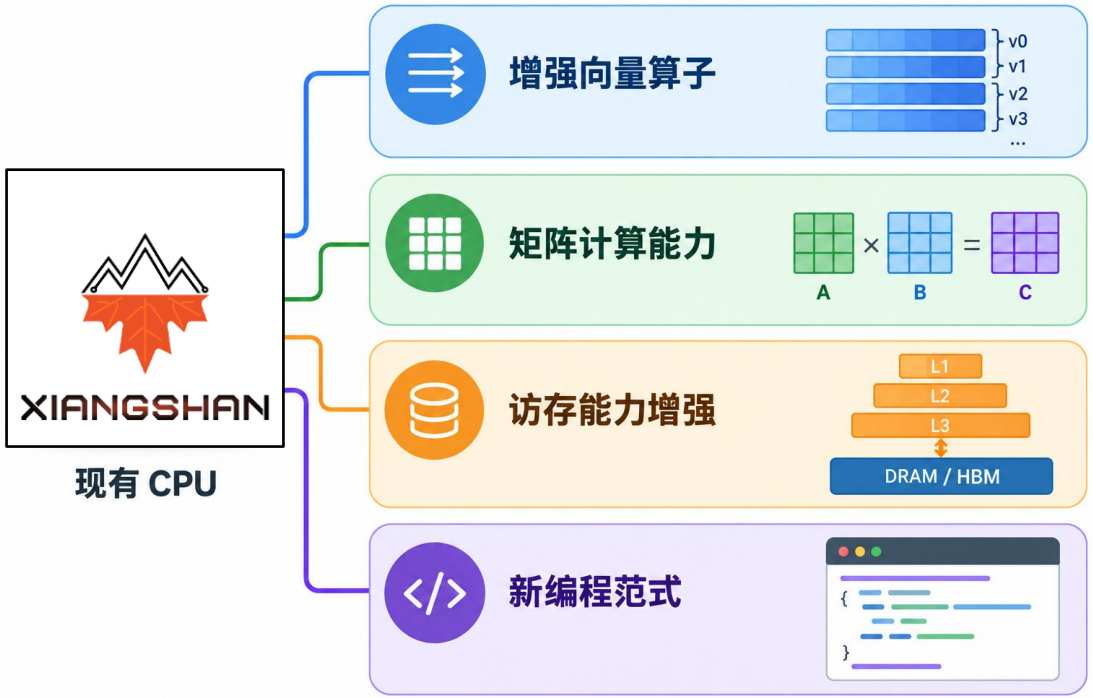




CPUs:GPUs = 1:8 → 1:1 → 4:1

CPU 原生的 AI 解决方案

- **通推一体：扩展 CPU 的 AI 推理能力，兼顾通用计算与推理任务**
- **依托 RISC-V 开源开放生态，实现软硬协同极致优化**



理念优势
• 算力同时支撑通推，硅片利用率高
• 异步矩阵指令与乱序操作调度
• 低 Kernel 启动与反馈延迟
• 兼容 CPU 编程模型，低编程复杂度
• 复用 CPU 编译器与自动向量化能力

目录

一 XSAI：开发理念

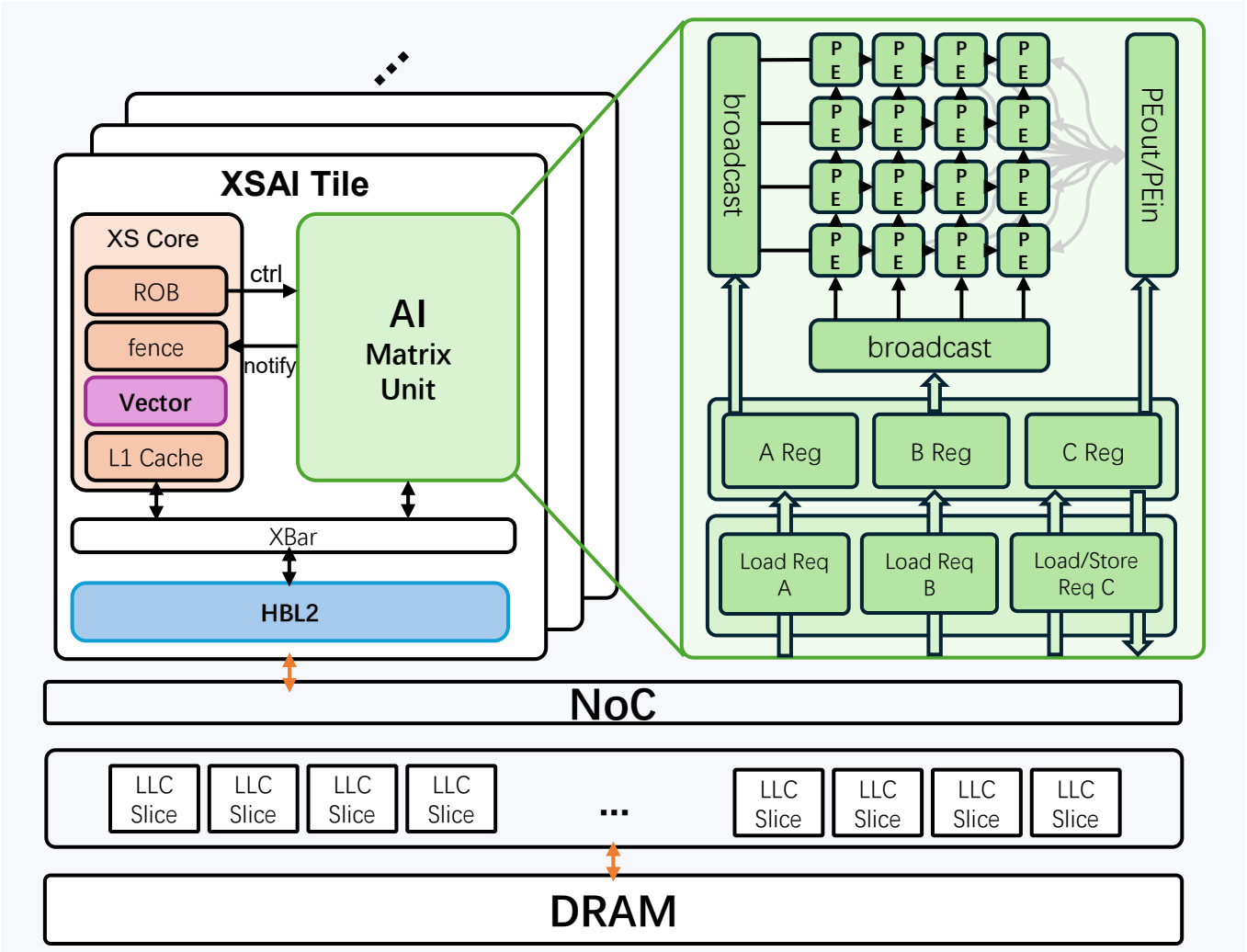
二 XSAI：微架构设计

三 XSAI：编程范式与软件生态

四 XSAI：基础设施

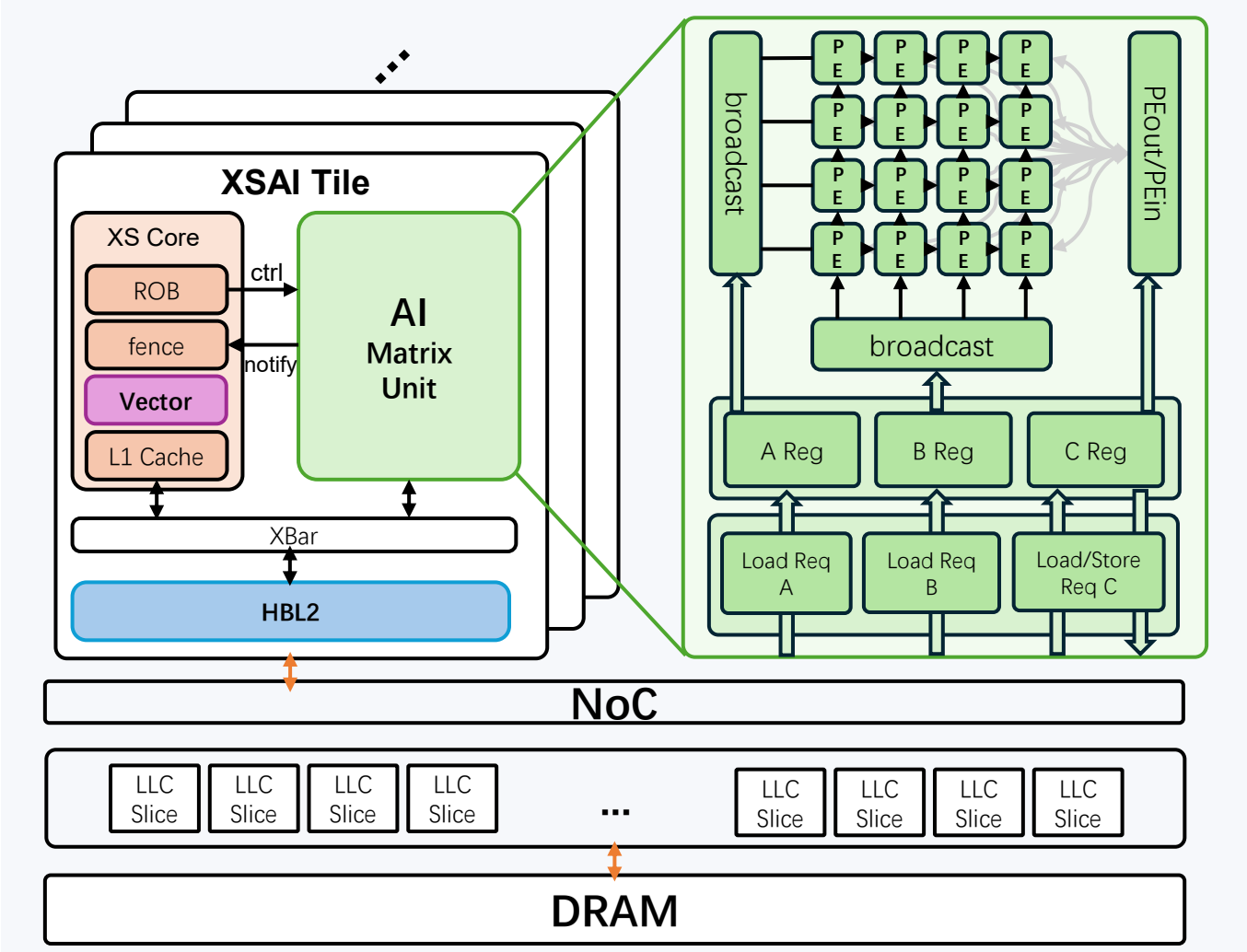
五 XSAI：设计评估

微架构总览



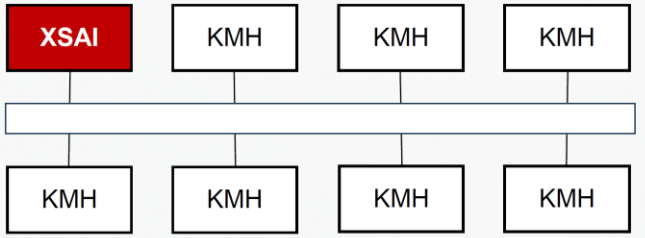
核心规格（可配置）	
CPU 控制核	香山昆明湖v2r2 SPECint2006 > 15.0/GHz
向量	昆明湖乱序RVV向量， 2*128 bit RVV1.0， 支持BF16
矩阵	1~64 TOPS FP4, 0.5~32 TOPS INT8/FP8
数据格式	TF32/FP16/BF16/FP8/INT8/MXFP4
高带宽 L2	容量 1~2M 可配，最高 1.28TB/s
总线	CHI/TL 兼容，适配 Ring/Mesh NoC

微架构总览



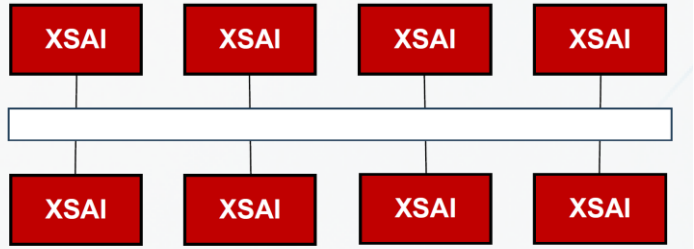
在 SoC 中灵活组合

1*XSAI +N*KMH



具身智能、边缘计算

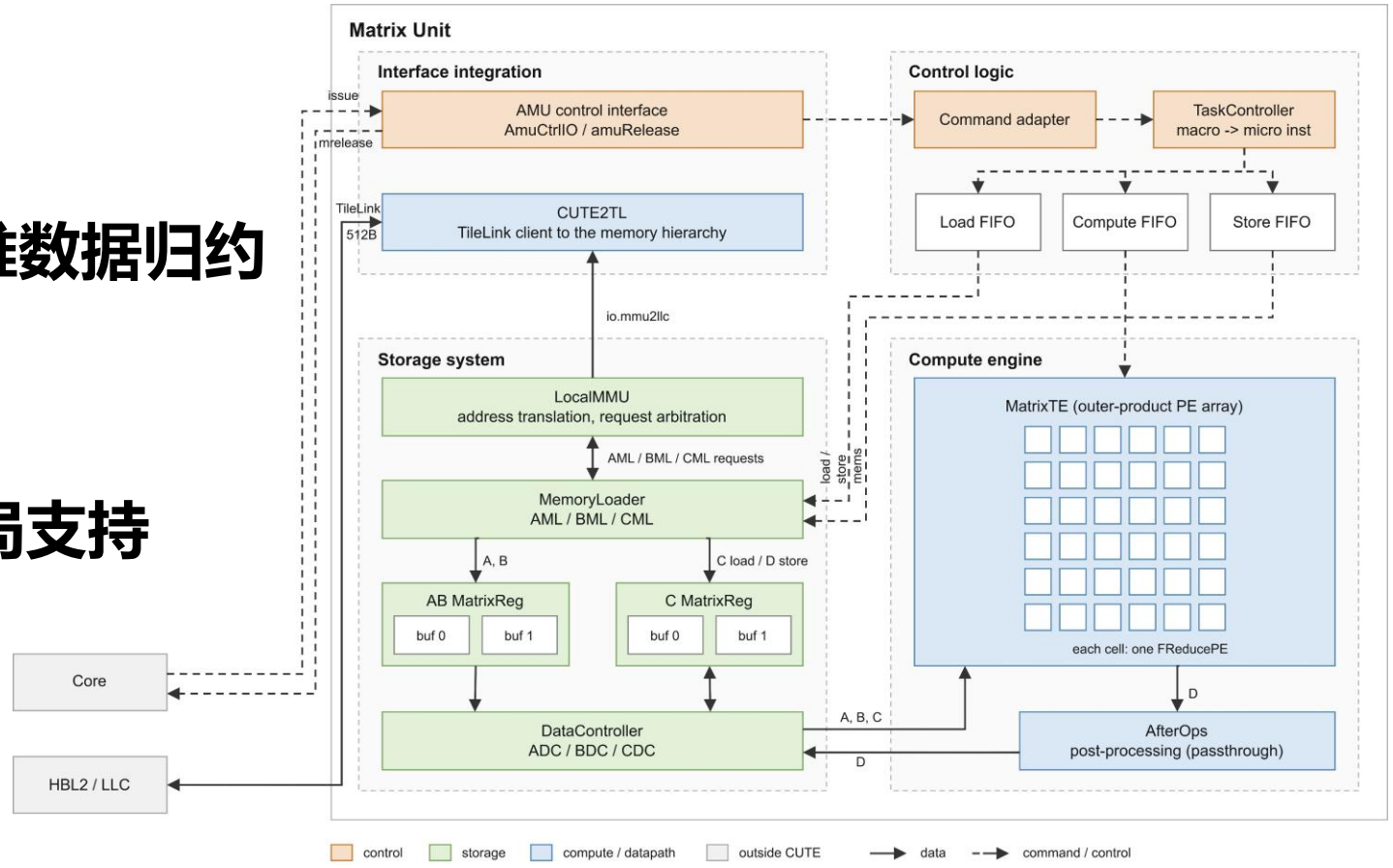
N*XSAI



云端推理服务器，个人智能主机

亮点① 矩阵运算单元

- 外积数据流，降低访存压力
- 行列数据广播，PE 完成 K 维数据归约
- 计算与访存解耦并行
- 支持依赖感知的乱序发射
- 多精度计算与灵活的数据布局支持



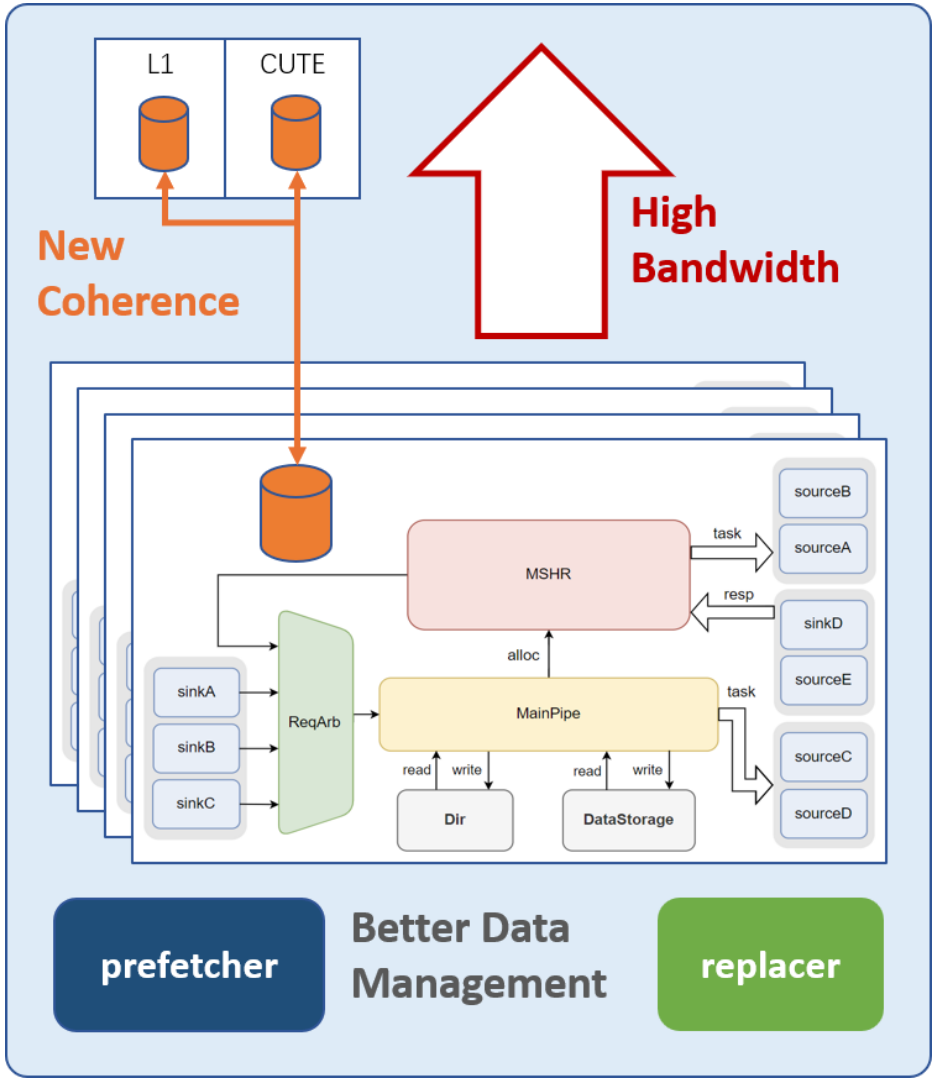
基于开源项目 OpenCUTE https://github.com/OpenCUTE/CUTE 进一步开发与优化

亮点② 高带宽 L2（HBL2）

- **权衡容量与延迟，L2 是矩阵单元合适连接层级**

	首要需求	适合缓存层级
标量/向量单元	低延迟	L1 Cache
矩阵运算	缓存容量和命中带宽	**L2 Cache**

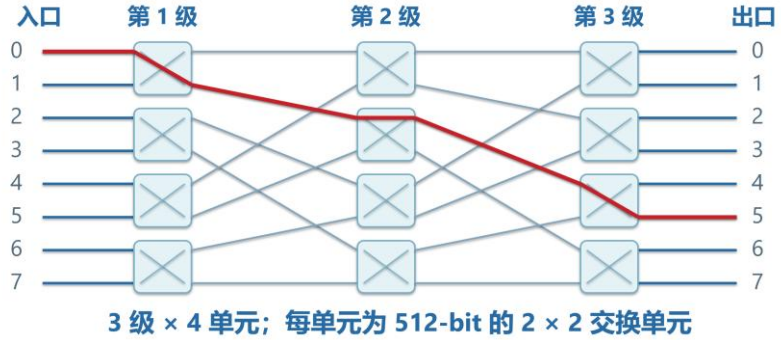
- **拓宽总线以满足矩阵单元的访存带宽需求**
  - 8 bank * 64B 访存
- **AI 亲和的缓存管理特性**
  - 轻量级缓存一致性管理
  - 面向推理访存模式的预取算法



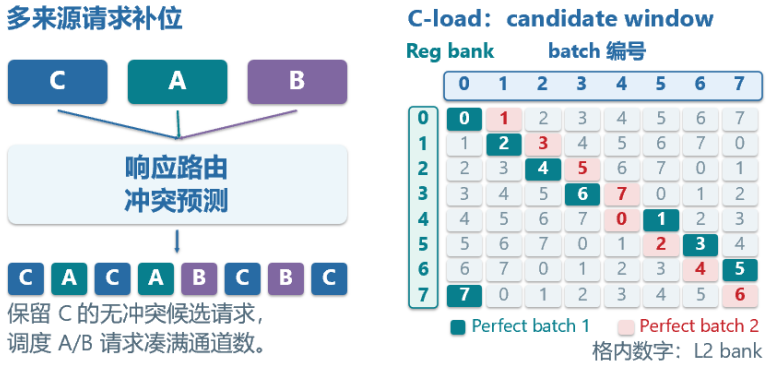
亮点③ 缓存互联网络

- **高带宽 L2 带来的挑战：512B 宽数据跨 8 bank 的全互联开销与高阻塞**

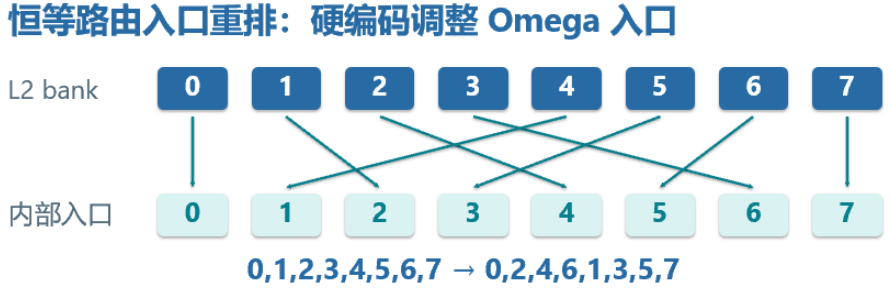
优化一：以 O(N log N) 的互联规模实现 8 通道并发返回



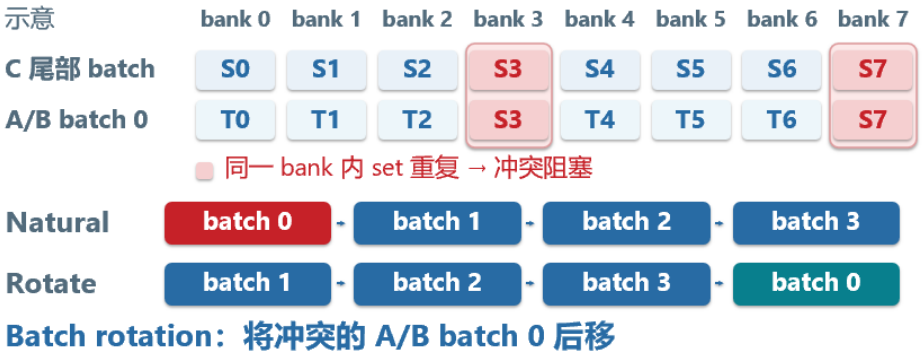
优化二：基于返回网络冲突预测的智能调度



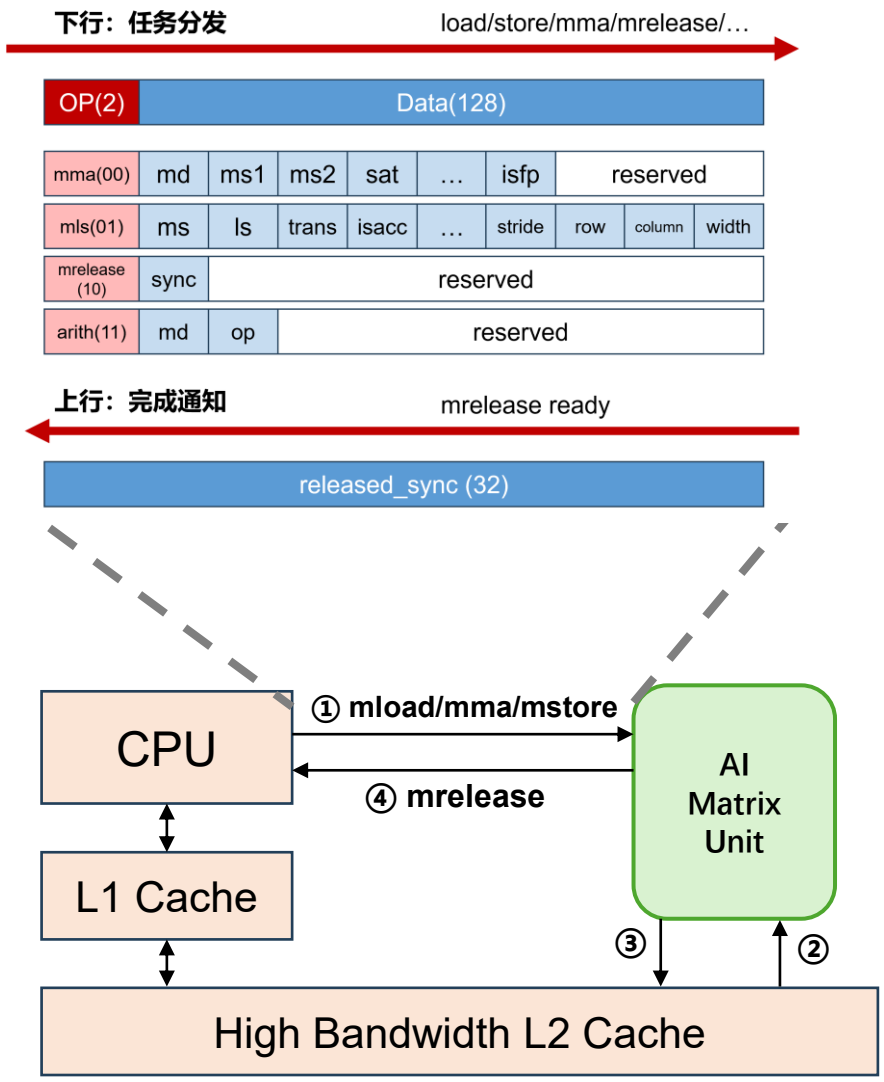
优化三：路由入口重排与 set 错峰



i-to-i 八路全通过；不增加交换级或开关数量。



亮点④ CPU与矩阵解耦交互接口



汇编代码及接口信号举例：

下行：XS->AI Matrix Unit

Matrix load(0)/store(1)

op(2)	md(4)	ls	trans	isacc	…	widths(3)
0x1	0x0	0	0	1		0x2

MMA

op(2)	reserved	md(4)	sat	ms1(4)	…	issigned
0x0		0x0	0	0x0		1

mrelease

op(2)	reserved	token(5)
0x2		0x01

上行：AI Matrix Unit -> XS

released_token(32)							
31	30	29	…	3	2	1	0
0	0	0		0	0	1	0

```
mlce32.m acc0, (r1), r2
mlae8.m tile0, (r3), r4
mlbe8.m tile1, (r5), r6
mqma.mm acc0, tile0, tile1
msce32.m acc0, (r7), r8
mrelease tok1
```

示例片段

控制核心与矩阵单元的典型交互过程（GEMM）：

① 处理器核向矩阵单元发送控制信号：mload/mma/mstore

② 矩阵单元从 HBL2 取数，并完成矩阵运算

③ 矩阵单元向 HBL2 存数

④ 矩阵单元通过 mrelease 机制修改核内特定寄存器，从而通知处理器核计算完成

目录

一 XSAI：开发理念

二 XSAI：微架构设计

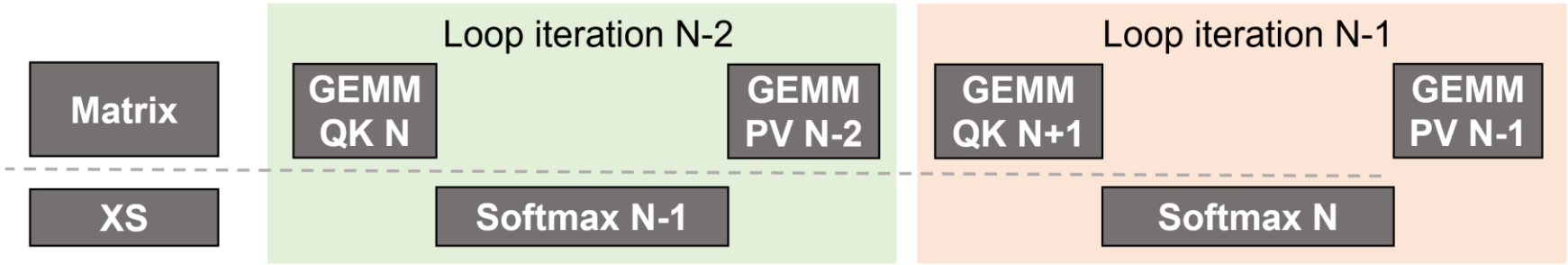
三 XSAI：编程范式与软件生态

四 XSAI：基础设施

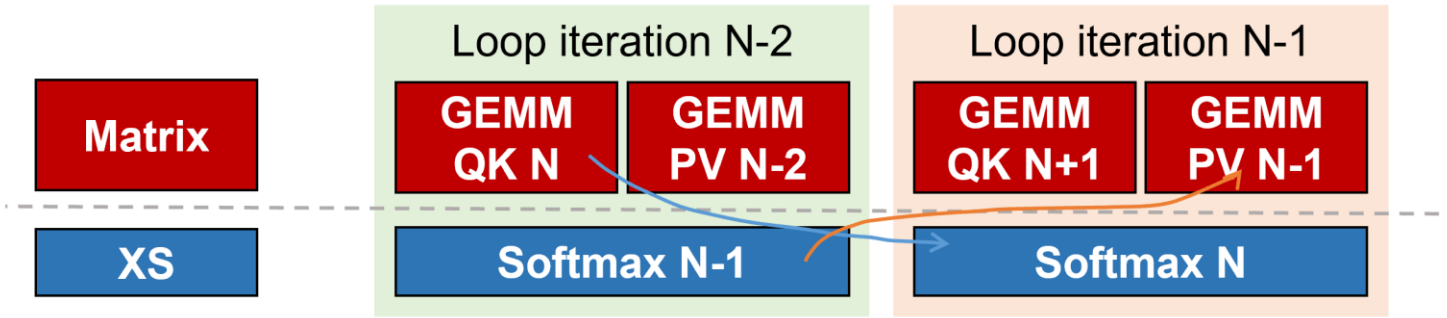
五 XSAI：设计评估

CPU 编程范式下的矩阵运算

- 异步矩阵指令提交：支持矩阵运算与其他运算并行化，提升吞吐
- Acquire-Release 同步模式：编程友好，软硬件协同维护一致性



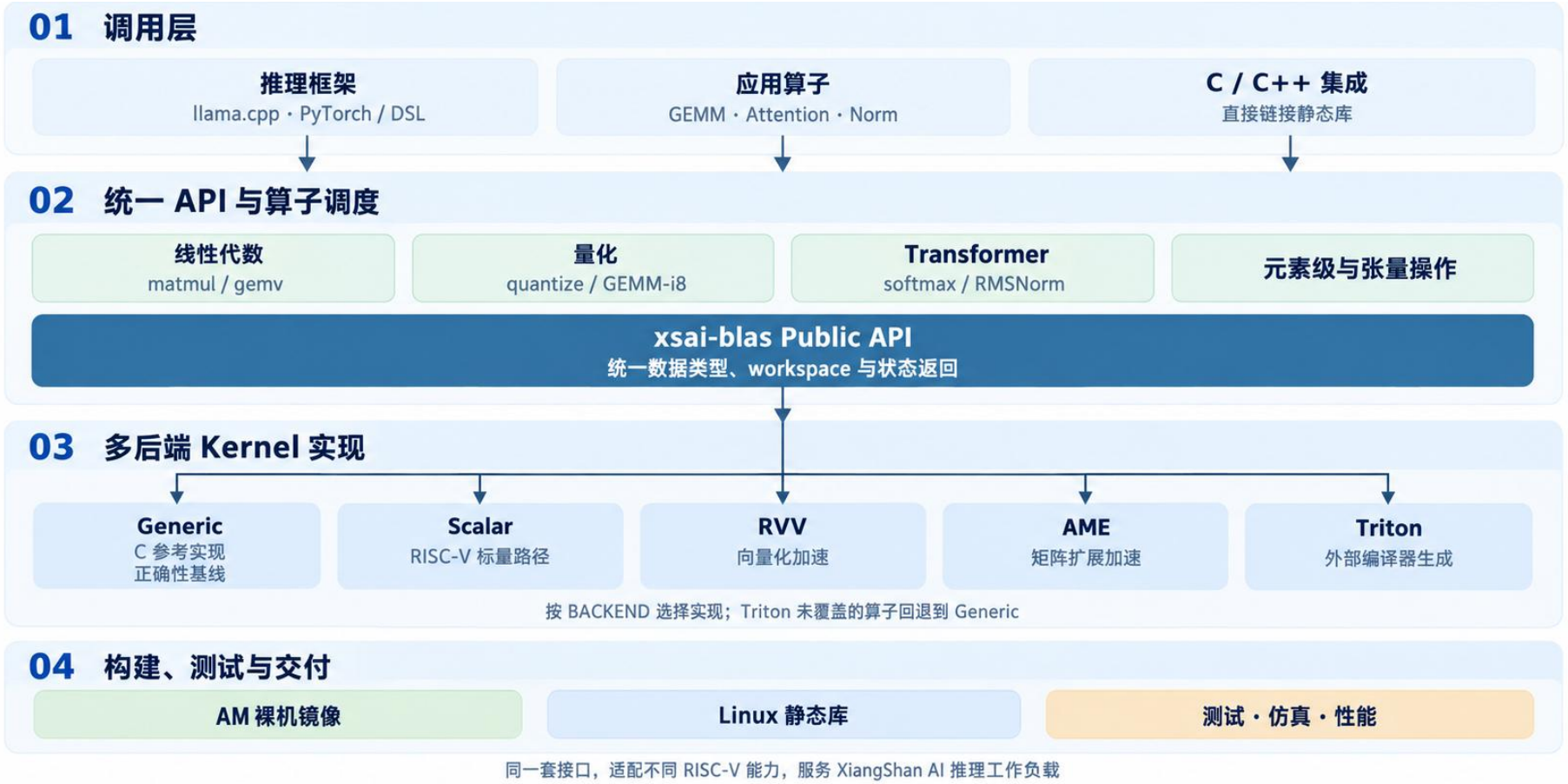
无异步矩阵：GEMM 与 Softmax **无法重叠执行**




```
// main loop
for (int i = 0; i < n - 2; ++i) {
  gemm_qk(i + 2);
  mrelease(tok);
  gemm_pv(i);
  macquire(tok, i + 2);
  softmax(i + 1);
}
```

支持异步矩阵: **支持 GEMM 与 softmax 的流水并行**

XSAI-BLAS: 高性能算子库



01 统一入口

框架只对接一组 plan API
tensor 以 DLPack 传入

02 运行时选择

同一库内包含 generic、RVV、AME
按 capability + heuristic 选择实现

03 计划一次，执行多次

冻结 implementation
重复执行时跳过重新选择

软件栈整体开发情况


模型负载
Qwen / LLaMA / DeepSeek 等大模型
推理框架
llama.cpp / ggml
主要适配入口
PyTorch
生态拓展
算子开发
算子库与手工优化
xsai-blas
轻量级算子库 · AME 加速
DSL 算子编译
Triton
TileLang
算子描述 → IR → 底层编译
编译工具链
buddy-compiler / LLVM 21 + AME
算子编译 · 汇编 / C / C++ · AME Intrinsics
指令集规范
RISC-V + RVV + AME
社区 AME v0.6 草案拓展
硬件平台
XSAI CPU + 向量单元 + 矩阵单元 + L2
系统运行支撑
Linux
上下文与异常管理
OpenSBI
硬件初始化 · PMU
设备树 · RootFS
开发与验证工具
裸机 MicroBenchmark
QEMU User / System
NEMU + DiffTest
RTL 仿真 / FPGA
生态共建
XSAI × RuyiAI
已达成合作备忘录，共同建设 XSAI 软件生态

目录

一 XSAI：开发理念

二 XSAI：微架构设计

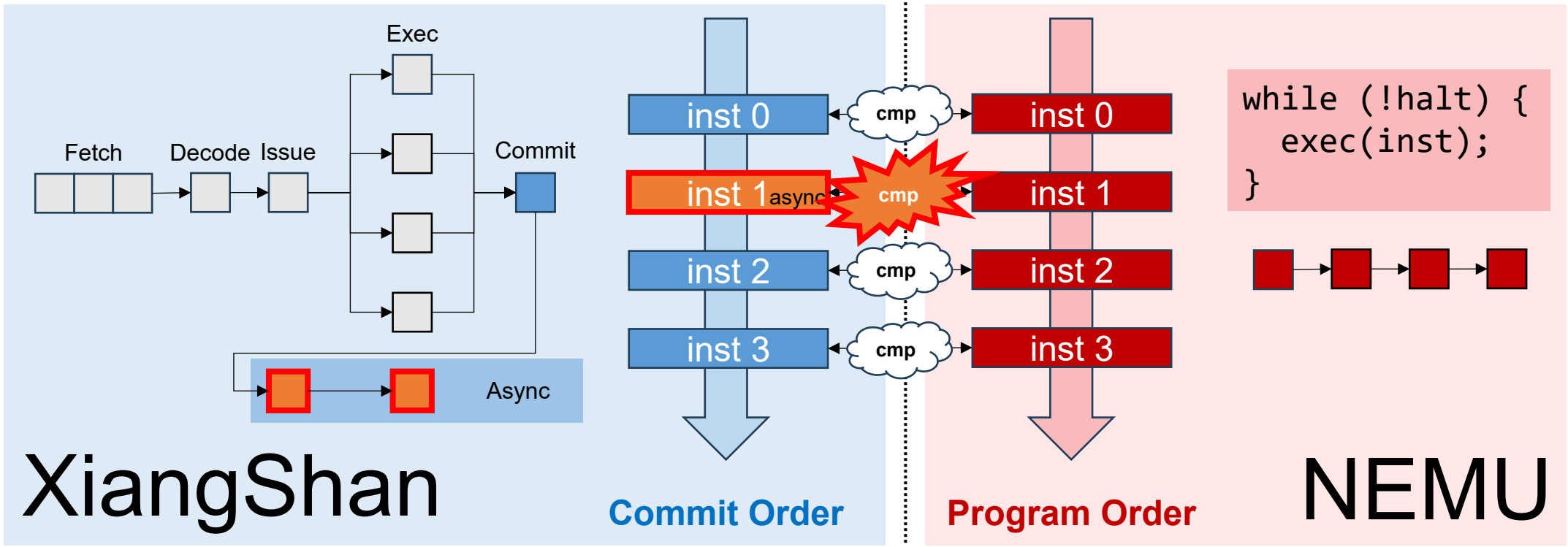
三 XSAI：编程范式与软件生态

四 XSAI：基础设施

五 XSAI：设计评估

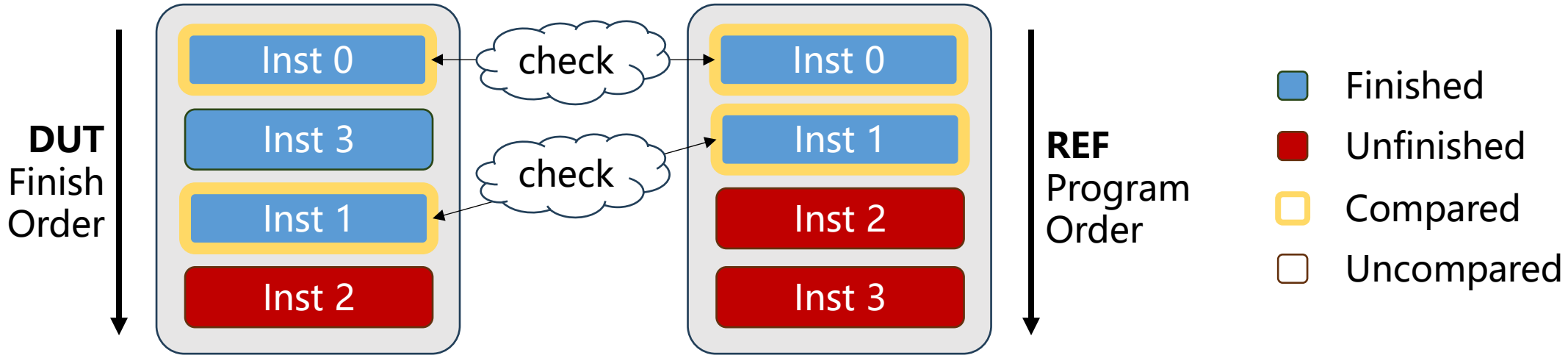
功能对比：XSAI Difftest 解决方案

- 异步矩阵指令在指令提交阶段尚未产生执行效果，无法直接比较结果



功能对比：XSAI Difftest 解决方案

- **解决方案：使用软件 ROB 维护指令生效顺序**
  - 软件层面重排指令，仅重排异步指令
  - 比对已完成指令序列前缀，上报差异
  - 相比 GPGPU Difftest 方案有更低的 Overhead：~1%



性能评估：基于采样的评估方法

完整工作负载

- 准确率高
- 随 LLM 规模增长，无法在高精度模拟器/ RTL 仿真上运行完

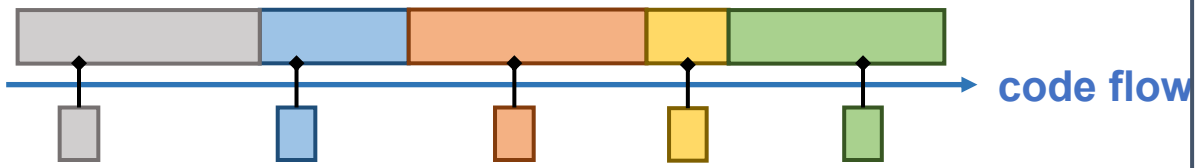
算子级 Benchmark

- 仿真规模小、速度快
- 需人工提取各类算子
- 无法反映系统级优化的综合收益

采样性能评估

- 仅详细仿真代表性片段
- 保留真实的程序特征
- 兼顾准确率与仿真速度

指令集模拟器
完整负载的功能仿真
(采样)



高精度模拟器/ RTL
程序片段的性能仿真
(评估)



$$CPI = \sum_{i=0}^{n} CPI_i \times weight_i$$

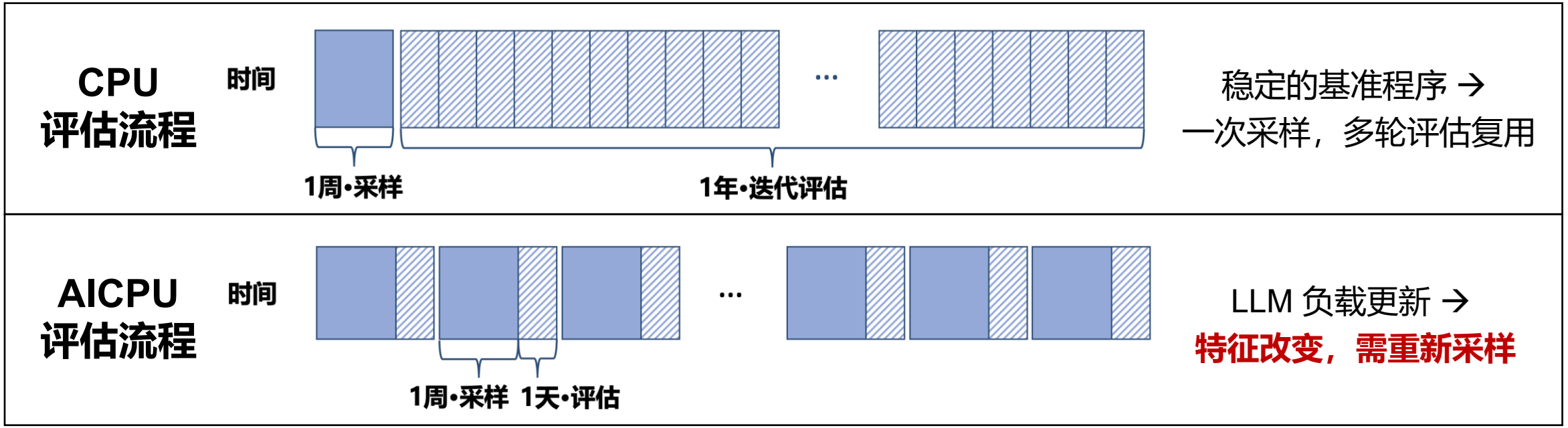
- **单核 CPU 2% 以内误差，500×以上加速** [1]
- **多核CPU、GPU等场景有效**
- **工业界广泛应用**

[1] Han, Chenji, et al. "MultiPoint: Enabling scalable pre-silicon performance evaluation for multi-task workloads." BenchCouncil (2024).

性能评估：基于采样的评估方法

- 模型高频更新
- 推理框架优化多样
- AI 扩展指令不断演进

→ LLM 推理负载更新频繁



挑战：采样效率成为瓶颈，而现有加速方法不适用于快速演进的 AI 指令

性能评估：基于采样的评估方法

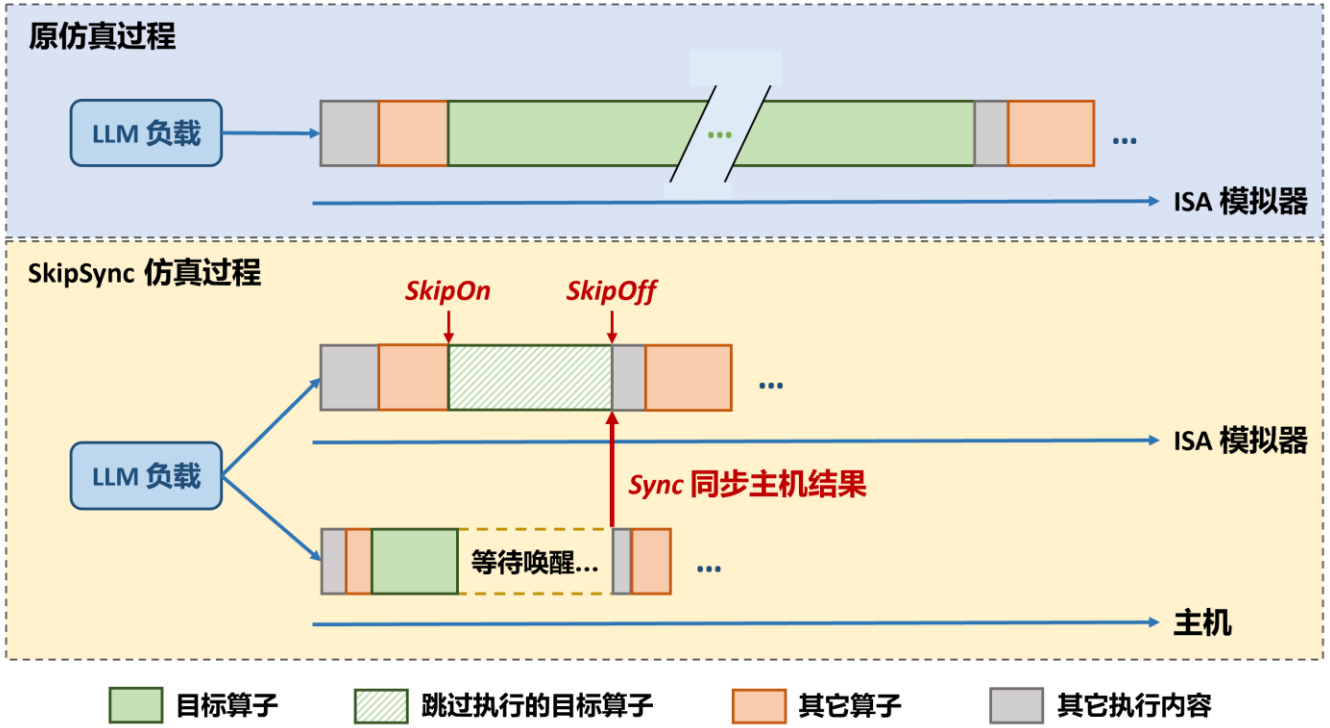
- **解决方案：SkipSync**

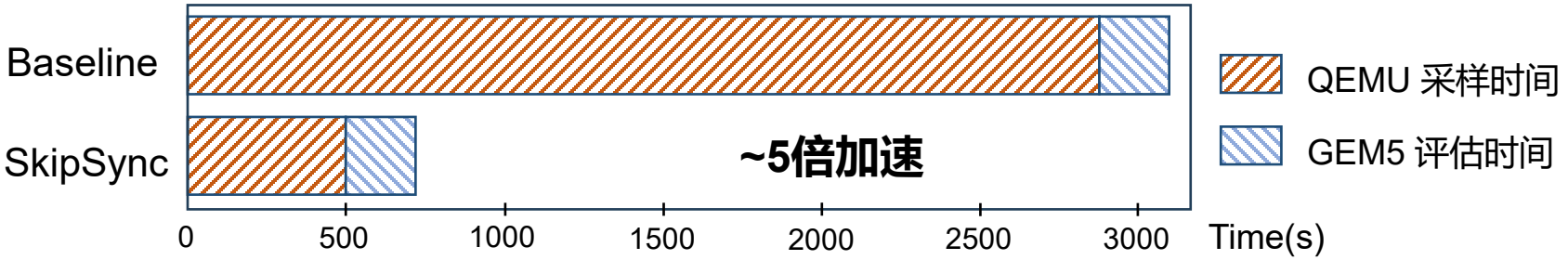
➢ **Skip**
- 在目标算子内跳过加速器指令的执行阶段

➢ **Sync**
- 以算子力度将宿主机计算结果同步到模拟器

✓ **仅需少量原语插入保证采样准确性**

✓ **对快速变化的加速器指令可扩展性强**





目录

一 XSAI：开发理念

二 XSAI：微架构设计

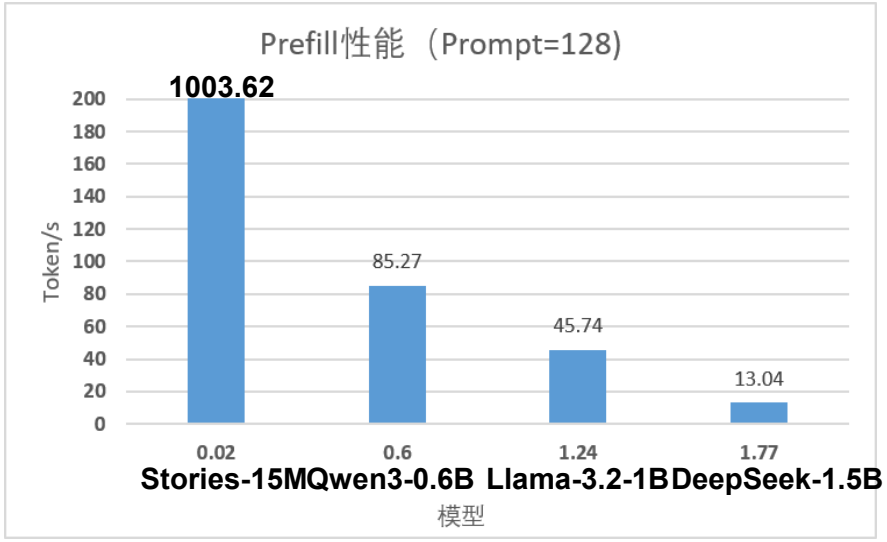
三 XSAI：编程范式与软件生态

四 XSAI：基础设施

五 XSAI：设计评估

性能评估

模型	参数量	量化	pp128 t/s	128 tok 耗时
Stories-15M	24.4M	Q8_0	1003.62±9.8	0.13 s
Qwen3-0.6B	0.60B	Q8_0	85.27±6.8	1.50 s
Llama-3.2-1B	1.24B	Q8_0	45.74±9.8	2.80 s
DeepSeek-R1-Distill-1.5B	1.77B	Q8_0	13.04±60.0	9.82 s



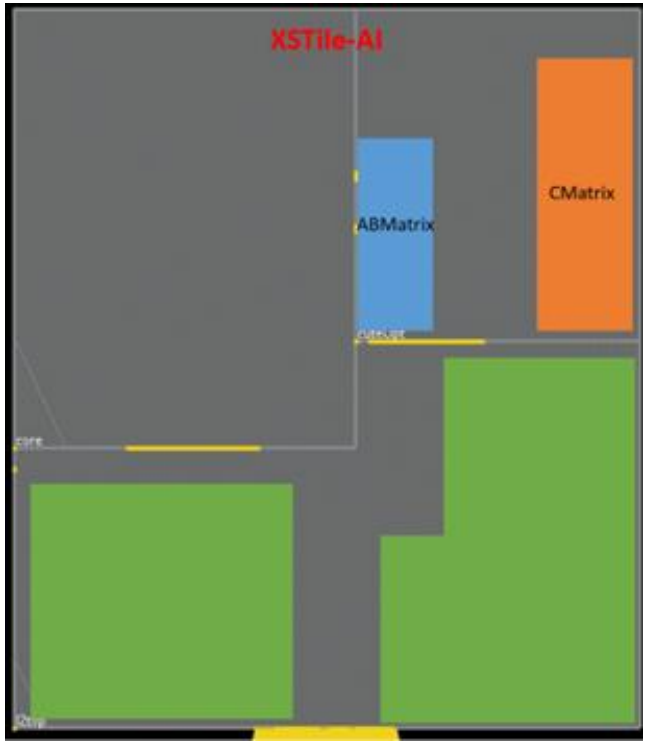
平台	XSAI RISC-V（SBI PMU 计数 · 仿真环境）
后端	CPU · 单线程　AI矩阵算力：8TOPS INT 8 @2GHz
工具	llama-bench · -p 128 -n 128 -r 1 --no-warmup
模式	Q8_0 量化 · --fake-like

通算性能：维持昆明湖 V2R2 性能表现

SPECint 2006/GHz: 14.887->**14.976** (+0.58%)

SPECfp 2006/GHz: 14.950->**14.905** (-0.29%)

PPA 评估



7nm工艺下

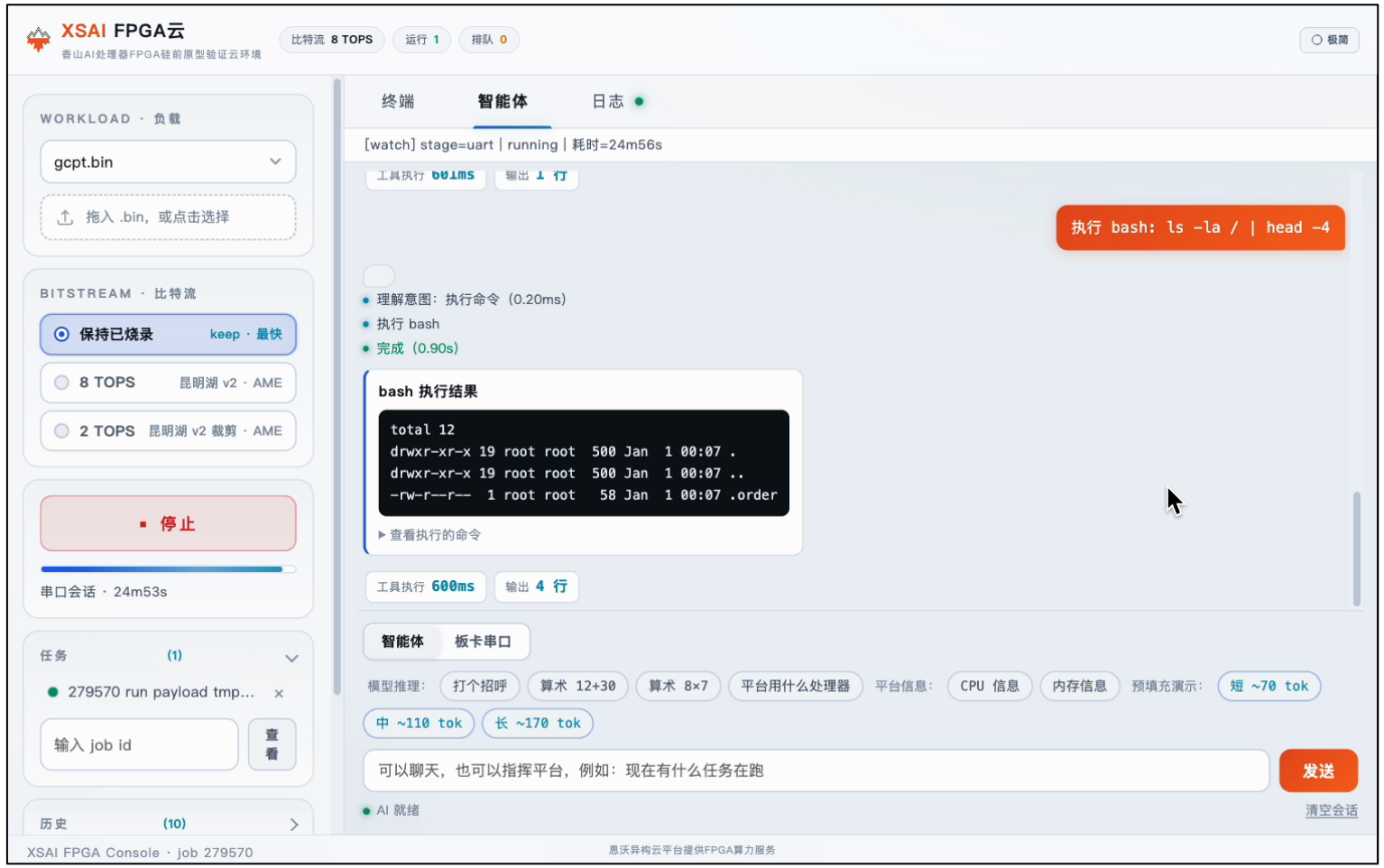
香山核：2.8 GHz
L2 Cache： 2.5 GHz
矩阵单元：2.5 GHz

SoC 方案

1. 全核同频至 2.5GHz
2. 异步桥，性能损失约 2%

	XSCore	CUTE	L2	Total
Area	0.87mm^2	0.50mm^2	1.5mm^2	2.87mm^2
Floorplan	1.72mm^2	0.89mm^2	2.67mm^2	5.28mm^2
Density	50.58%	56.17%	56.01%	
Power(*Est.*)	1.16*W*	0.67*W*	0.66*W*	2.49*W*

DEMO：XSAI FPGA 云



- ISA: RISC-V指令集+AME矩阵扩展
- AI矩阵算力：8TOPS@2GHz
- OS: OpenRUYI
- FPGA：1902 FPGA@50 MHz

欢迎大家来开芯院
5 楼 B04 展台体验！

生态共建

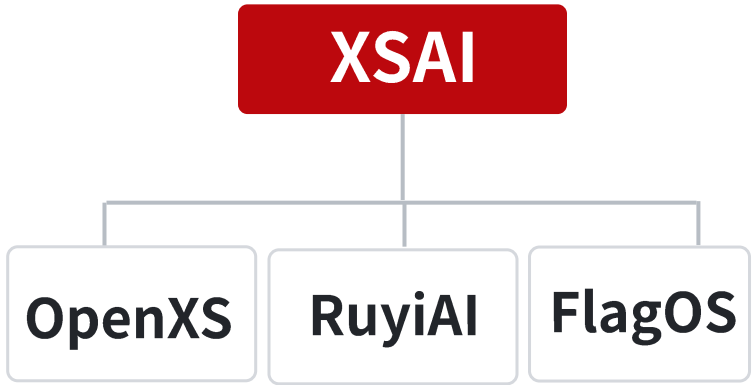
01 全栈开源

覆盖规范、实现与开发工具

- 指令集
- RTL 代码
- 设计文档
- 模拟器
- 软件栈
- 基础设施

02 生态协作

积极融入 RISC-V AI 生态



03 架构演进

架构设计中

CPU 交互	更低耦合
Element-wise	独立处理单元
数值格式	支持更低精度

欢迎社区参与 XSAI 共建

以开源共享成果，以协作加快下一代架构迭代

GitHub 仓库

设计文档