



# 香山社区：2024~2025

---

徐泽凡

香山开源社区 / 北京开源芯片研究院

2025 年 7 月 16 日

# 回顾：第一届香山社区大会

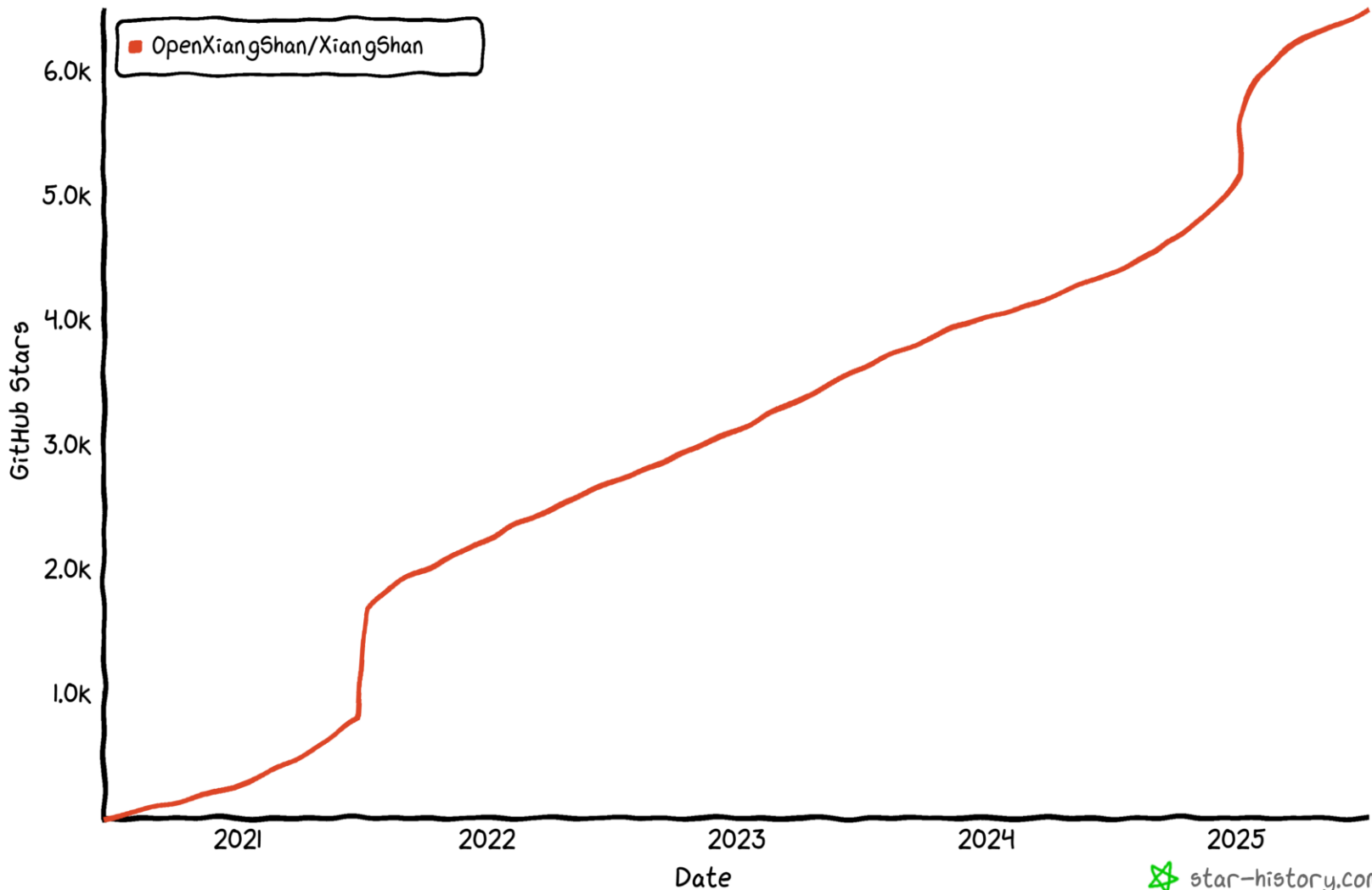


2024年 第四届 RISC-V 中国峰会 杭州黄龙饭店



# 香山仓库的星标情况

Star History



2025-07-16

香山团队 The XiangShan Team

star-history.com

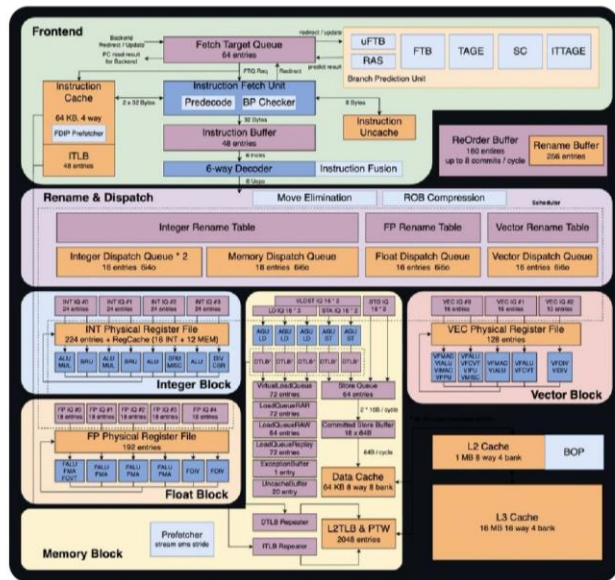


George Hotz  
@realGeorgeHotz

关注

Why is the top performing open source CPU (XiangShan) Chinese? Where's the American project to beat this? Gonna lose so hard if America keeps investing in scams, instead invest in things like this.

翻译推文

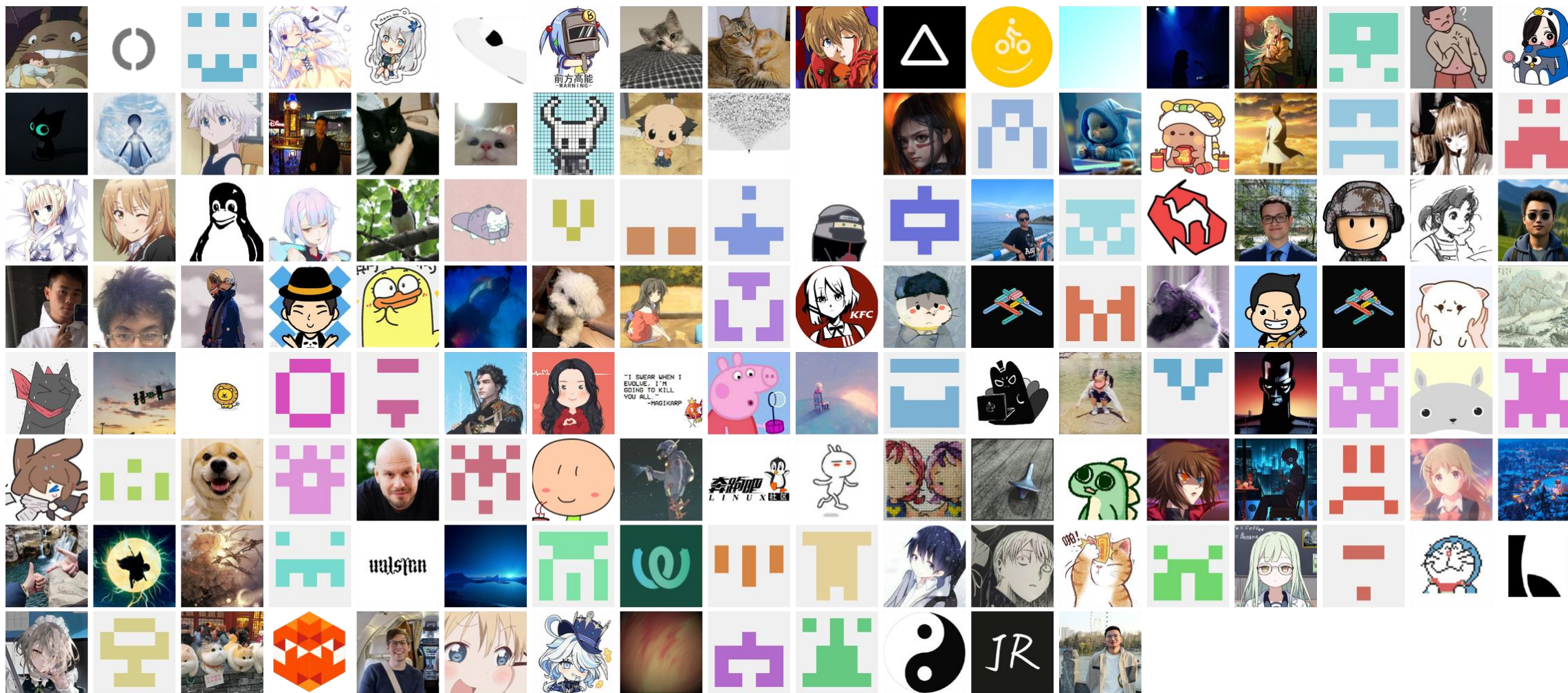


25年1月2日, 11:38 下午 · 51万 查看

487 转推 78 引用 5,369 喜欢 2,052 书签



# 香山主要仓库的贡献者 (2020 年 6 月至今)

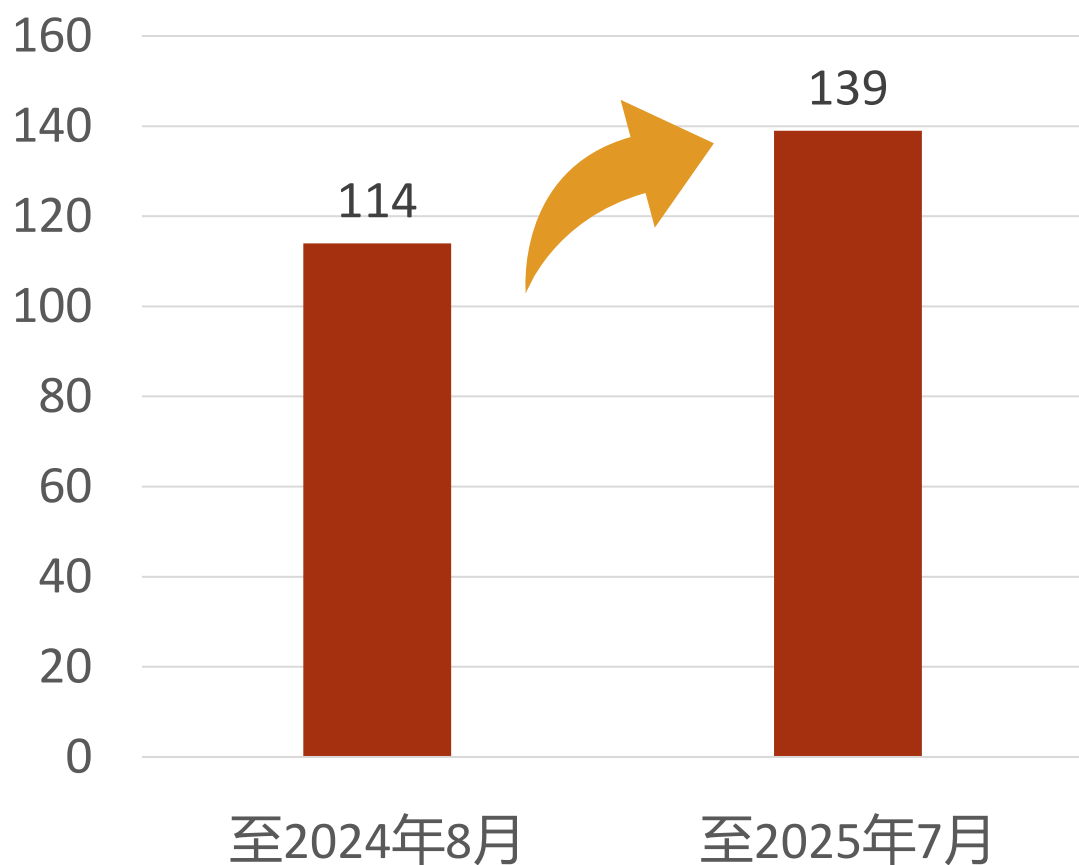


根据 OpenXiangShan 下 XiangShan, coupledL2, difftest, fudian, huancun, openLLC, utility, yunsuan, NEMU, tl-test, tl-test-new, env-scripts, nexus-am, XiangShan-doc 仓库统计, 共 139 位, 可能有遗漏。

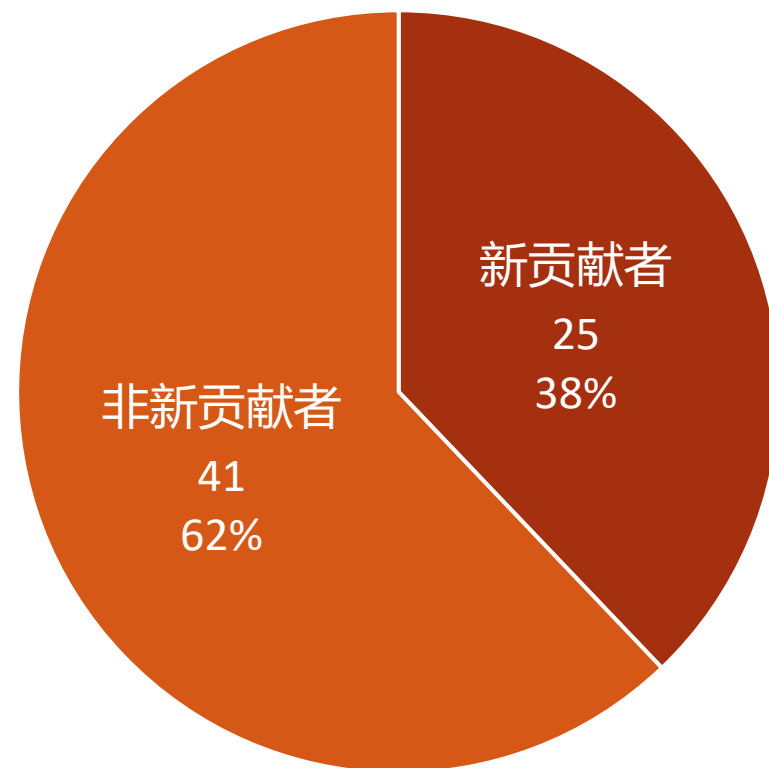


# 香山主要仓库的贡献者

2020年以来的贡献者数



2024年8月至2025年7月  
新贡献者占全部贡献者的比例

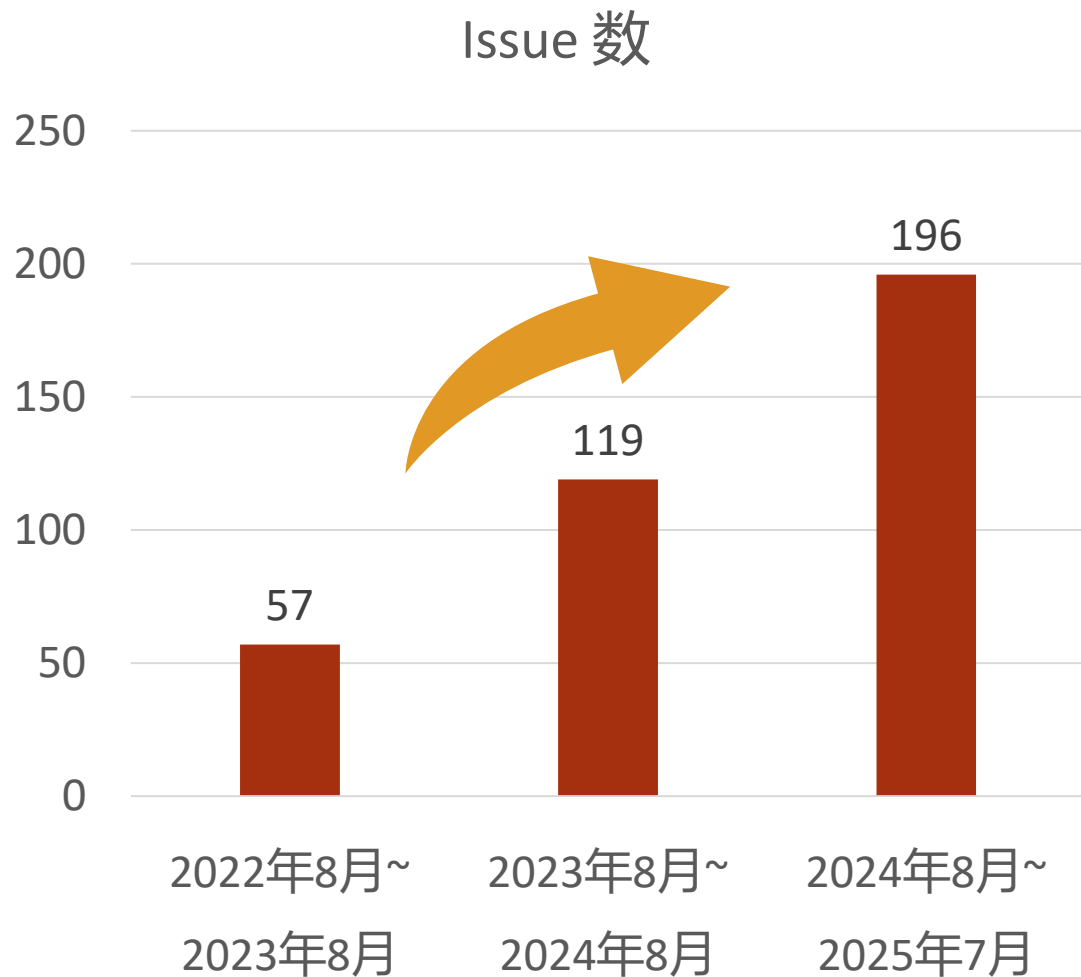


# QQ 群和微信群

- QQ 群已经接近 1200 人
- 5 个微信群，总计接近 1000 人
- QQ 群抽奖!
- 奖品：《香山源代码剖析》



# GitHub Issue & Discussion



- 首条回复时间
  - 平均 25 小时
  - 中位数 13 小时
- 争取进一步提升 Issue 回复速度
- 欢迎大家来提 Issue



# 编译器上游

- “香山昆明湖” 硬件描述并入 LLVM 和 gcc 主线

```
gcc-mirror / gcc
Code Pull requests 48 Actions Projects Wiki Security Insights
Commit f0cd40f
3 people committed on Jun 6
RISC-V: Support -mcpu for XiangShan Kunminghu cpu.
This patch adds support for the XiangShan Kunminghu CPU in GCC, allowing
the use of the '-mcpu=xiangshan-kunminghu' option.
XiangShan-KunMingHu is the third-generation open-source high-performance
RISC-V processor.[1] You can find the corresponding ISA extension from the
XiangShan Github repository.[2] The latest news of KunMingHu can be found
in the XiangShan Biweekly.[3]
[1] https://github.com/OpenXiangShan/XiangShan-User-Guide/releases.
[2] https://github.com/OpenXiangShan/XiangShan/blob/master/src/main/scala/xiangshan/Parameters.scala
[3] https://docs.xiangshan.cc/zh-cn/latest/blog
A dedicated scheduling model for KunMingHu's hybrid pipeline will be
proposed in a subsequent PR.
gcc/ChangeLog:
* config/riscv/riscv-cores.def (RISCV_TUNE): New cpu tune.
(RISCV_CORE): New cpu.
* doc/invoke.texi: Ditto.
gcc/testsuite/ChangeLog:
* gcc.target/riscv/mcpu-xiangshan-kunminghu.c: New test.
Co-Authored-By: Jiawei Chen <jiawei@iscas.ac.cn>
Co-Authored-By: Yangyu Chen <cyycyyself.name>
Co-Authored-By: Tang Haojin <tanghaojin@outlook.com>
master
```

```
llvm / llvm-project
Code Issues 5k+ Pull requests 4.4k Actions Security Insights
Commit 0e3e0bf
4 people authored 3 days ago · 21 / 23 · Verified
[RISCV] Add processor definition for XiangShan-KunMingHu-V2R2 (#123193)
XiangShan-KunMingHu is the third generation of Open-source
high-performance RISC-V processor developed by Beijing Institute of Open
Source Chip (BOSC), and its latest version is V2R2.
The KunMingHu manual is now available at
https://github.com/OpenXiangShan/XiangShan-User-Guide/releases.
It will be updated on the official XiangShan documentation site:
https://docs.xiangshan.cc/zh-cn/latest
You can find the corresponding ISA extension from the XiangShan Github
repository:
https://github.com/OpenXiangShan/XiangShan/blob/master/src/main/scala/xiangshan/Parameters.scala
If you want to track the latest performance data of KunMingHu, please
check XiangShan Biweekly: https://docs.xiangshan.cc/zh-cn/latest/blog
This PR adds the processor definition for KunMingHu V2R2, developed by
the XSCC team https://github.com/orgs/OpenXiangShan/teams/xscc.
The scheduling model for XiangShan-KunMingHu V2R2 will be submitted in a
subsequent PR.
-----
Co-authored-by: Shenglin Tang <tangshenglin@ict.ac.cn>
Co-authored-by: Xu, Zefan <ceba_robot@outlook.com>
Co-authored-by: Tang Haojin <tanghaojin@outlook.com>
main (#123193)
```



# Tutorial

- 过去一年里，我们总共举办了 6 场 Tutorial

会议	时间	地点
MICRO 2024	2024-11-02	美国奥斯汀
HPCA 2025	2025-03-02	美国拉斯维加斯
ASPLOS 2025	2025-03-30	荷兰鹿特丹
RVSE 2025	2025-05-12	法国巴黎
ISCA 2025	2025-06-21	日本东京
APPT 2025	2025-07-14	希腊雅典

- 我们不断迭代升级 Tutorial 流程
- 接下来的一年，我们还将继续举办 Tutorial





# 开源软件供应链点亮计划：开源之夏 2025

- 首次参加开源之夏活动
- 发布 9 个赛题
  - 6 个万众一芯 UT 验证
  - 2 个香山基础设施
  - 1 个 XS-GEM5
- 其中 6 个赛题收到 9 份申请书
- 赴多所高校参与宣讲活动
  - Special Thanks @Anzooooo





# GLCC 确实开源编程夏令营

- 2024、2025 连续两年参与
- 赛题与万众一芯、XS-GEM5等相关
- 其中 2024 年：
  - 香山开源社区荣获“优秀社区”
  - 安旭老师荣获“优秀社区管理员”



# RISC-V 在线黑客松 (Hackathon)

- 2025 年 4 月
- 中国唯二的线下黑客松
- 微信群 59 人，6 人线下参赛
- 赛题：香山处理器上的 ChaCha20 加密算法优化
- <https://github.com/cyyself/chacha20-xiangshan>
- Special Thanks @cyyself







# 文档建设 (1): 致谢列表

- <https://docs.xiangshan.cc/acknowledgements/>
- 列出了 32 篇香山设计中使用到的微架构工作
- 文章与 RTL 实现源码建立双向链接
- 2024 年 11 月上线
- 向这些工作的作者表示感谢!

## Acknowledgments

This list outlines the techniques used in the XiangShan RTL codes.



### Instruction Fetching

- [1] Glenn Reinman, Todd Austin, and Brad Calder. "[A scalable front-end architecture for fast instruction delivery.](#)" 26th International Symposium on Computer Architecture (ISCA). 1999. [\[RTL Codes\]](#)
- [2] Alex Ramirez, Oliverio J. Santana, Josep L. Larriba-Pey, and Mateo Valero. "[Fetching instruction streams.](#)" 35th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO). 2002. [\[RTL Codes\]](#)

### Instruction Prefetch

- [1] Glenn Reinman, Brad Calder, and Todd Austin. "[Fetch directed instruction prefetching.](#)" 32nd Annual ACM/IEEE International Symposium on Microarchitecture (MICRO). 1999. [\[RTL Codes\]](#)
- [2] Yasuo Ishii, Jaekyu Lee, Krishnendra Nathella, and Dam Sunwoo. "[Rebasing instruction prefetching: An industry perspective.](#)" IEEE Computer Architecture Letters 19.2: 147-150. 2020. [\[RTL Codes\]](#)
- [3] Yasuo Ishii, Jaekyu Lee, Krishnendra Nathella, and Dam Sunwoo. "[Re-establishing fetch-directed instruction prefetching: An industry perspective.](#)" 2021 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS). 2021. [\[RTL Codes\]](#)

## 文档建设 (2): 用户手册

- 适用于最新的昆明湖 V2R2 版本
  - 面向软件开发者, 指导系统和应用软件开发
  - 面向硬件集成商, 指导处理器核集成
  - 全文约 80 页
- 
- <https://docs.xiangshan.cc/projects/user-guide/>
  - PDF & 网页 双格式
  - 2025 年 2 月上线
  - AI 翻译英文版也已上线



### 香山开源处理器 用户手册

适用于 昆明湖 V2R2

873f8ad  
2025 年 7 月 14 日

# 文档建设 (3): 设计文档

- 适用于最新的昆明湖 V2R2 版本
  - 面向硬件开发者、处理器工程师
  - 全面介绍香山处理器的微架构、各模块实现
  - 全文约 550 页
- 
- <https://docs.xiangshan.cc/projects/design/>
  - PDF & 网页 双格式
  - 2025 年 4 月上线
  - AI 翻译英文版也已上线



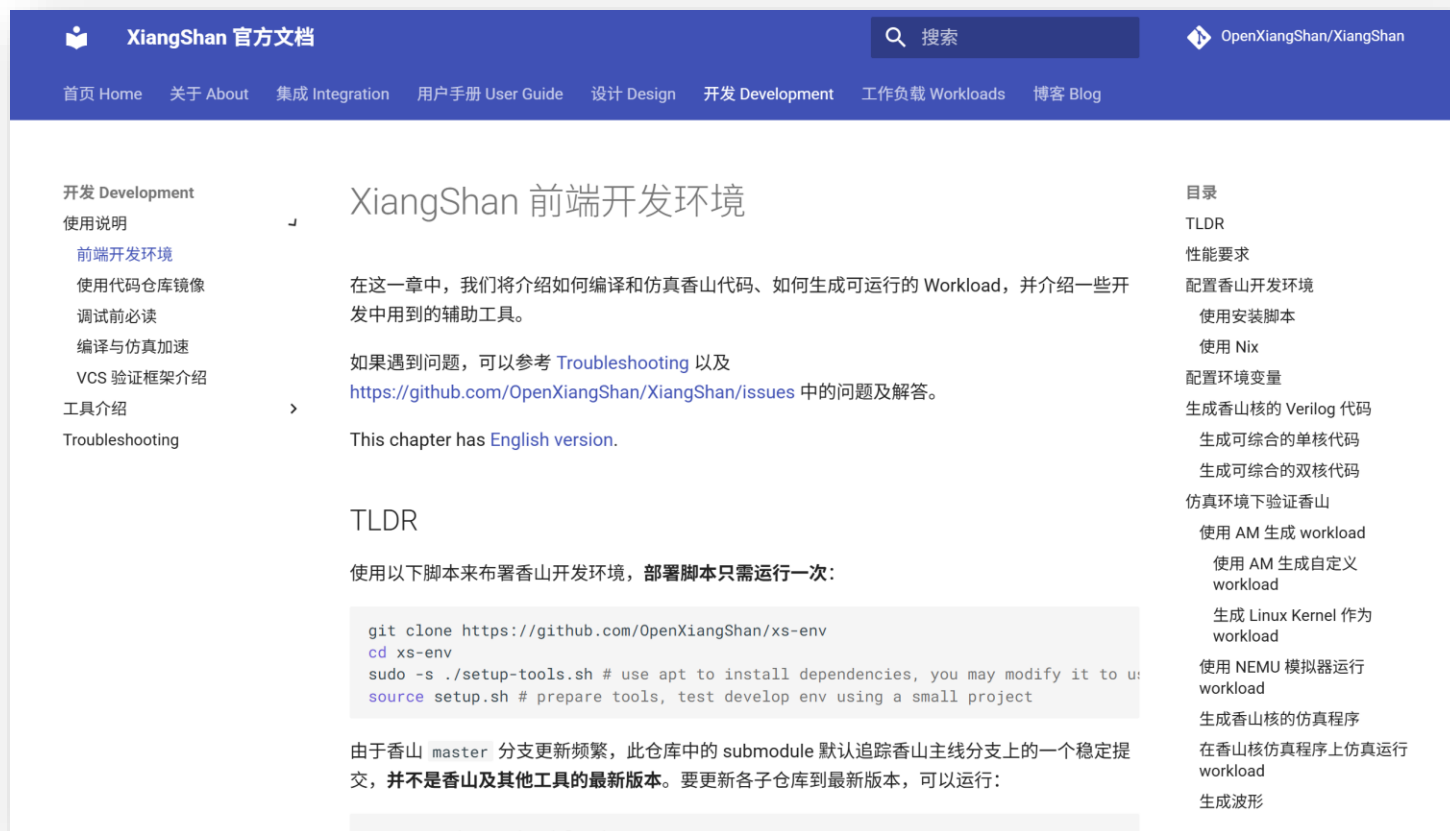
## 香山开源处理器 设计文档

适用于 昆明湖 V2R2

52fc49d  
2025 年 7 月 15 日

# 新文档基础设施 (1): 回顾现状

- 香山现有文档: **Markdown** 语言编写, **Mkdocs** 生成静态网页
- 使用 **ReadTheDocs** 服务部署于 <https://docs.xiangshan.cc>

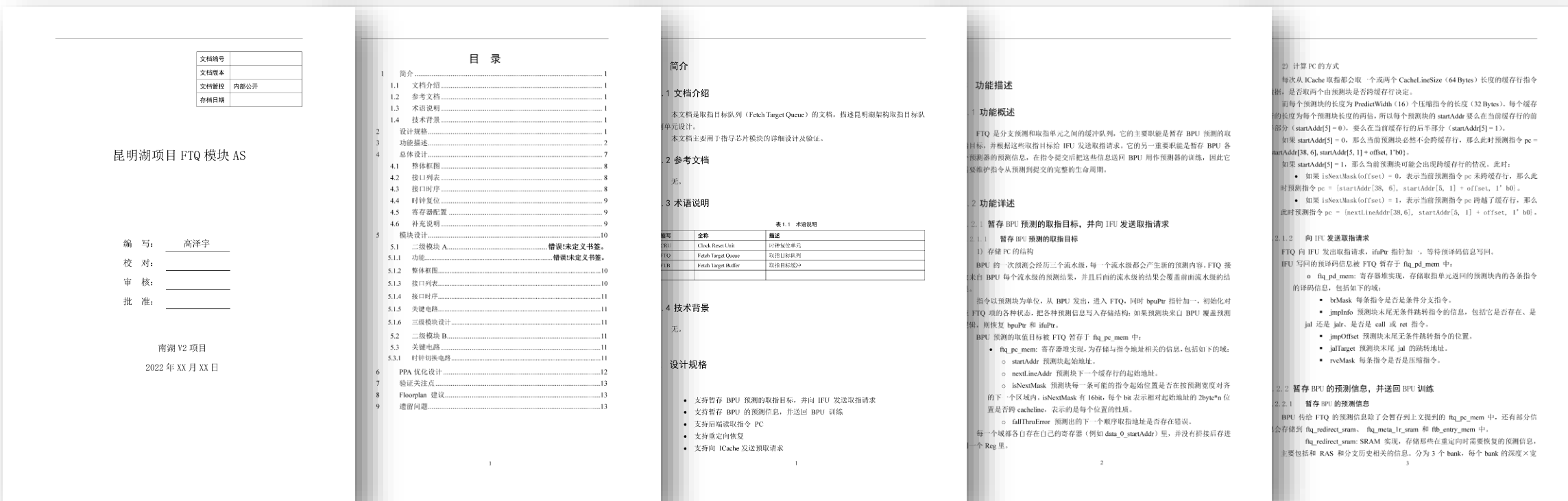






# 新文档基础设施 (1): 回顾现状 (续)

- 南湖 V2 / 昆明湖 V1 验证文档: **使用 Microsoft Word 编写**
- 符合工业界习惯, 方便验证团队在无网络环境下使用
- 版本管理、团队协作、统一格式皆困难, 偏离开源社区实践和习惯**





## 新文档基础设施 (2): 现状 → 需求

- 现在的文档基础设施不能满足香山社区发展的需要
- 需要搞一套新的基础设施，来实现：
- 保留当前写作习惯：**继续使用 Markdown**
- 适配社区和工业界：**支持输出 网站 & PDF 双版本文档**
- 面向国际：**支持多语言、便于翻译**
- 保持开源：**尽量使用开源工具**

## 新文档基础设施 (3): 使用 Markdown 生成 PDF

- 生成 PDF  $\approx$  排版，有以下几种常见路线：
- 转换成 HTML  $\rightarrow$  浏览器打印为 PDF
  - 浏览器的 CJK 排版水平属于图一乐，难堪大用
- 转换成 Word 等商业排版格式  $\rightarrow$  导出为 PDF
  - 商业工具有点.....而且难以自动化
- 转换成 LaTeX / typst 等开源排版语言  $\rightarrow$  生成 PDF
  - 比较靠谱。其中，LaTeX 对 CJK 排版的支持会更好一点
- 解决方案：使用 Pandoc 将 Markdown 转化为 LaTeX，再排版为 PDF
  - 用 Makefile 来管理构建流程

## 新文档基础设施 (3): 使用 Markdown 生成 PDF (续)

- 使用 Pandoc 将 Markdown 转化为 LaTeX, 再排版为 PDF
- **麻烦 1:** 多个不同层次的 md 文件如何组织成单一的 LaTeX
  - 使用 Pandoc 的 include-files 插件, 手动指定章节文件间的嵌套关系, 由插件自动平移 md 文件中标题的层次
- **麻烦 2:** 对于矢量插图, 网页需要 svg 格式, 但 LaTeX 只接受 PDF
  - 在构建过程中使用 librsvg2 将 svg 转成 PDF
- **麻烦 3:** TeX Live 太大, 如何集成到持续构建流程
  - 使用简化后的 TinyTex 发行版, 并且通过 Docker 镜像来加速构建
- 此外, 设计了一套专用的 LaTeX 模板: 封面、字体、排版细节.....



## 新文档基础设施 (4): 让 Markdown 支持更多特性

- Markdown 原生缺少许多特性, 存在许多方言
- Pandoc 和 Python-Markdown 支持的方言不一致
- 为此, 我们写了一系列
  - Python-Markdown 的插件
  - Pandoc 过滤器
  - 在 OpenXiangShan/docs-utils

特性	Pandoc	Python-Markdown
表标题	原生支持	✗
grid table	原生支持	插件对 CJK 有 bug
交叉引用	pandoc-crossref	✗
变量替换	✗	✗

插件	用途
remove_include	移除 Pandoc include-files 代码块
table_captions	支持表标题
grid_table	修复了上游 CJK 支持的 Bug
crossref	简易支持 pandoc-crossref 语法
replace_variables	支持变量替换

## 新文档基础设施 (5): 支持 i18n

- 让 Mkdocs 支持多语言
  - 为每个语言单独构建, 使用 ReadTheDocs 提供的多语言功能进行托管
  - 魔改 Mkdocs-Material 主题, 实现赞助限定的 Stay-on-Page 功能
  - Mkdocs-static-i18n 等插件存在硬伤, 没法采用
- 用 Weblate 作为协作翻译平台
  - Weblate 有限地支持翻译 Markdown 文件格式 (作为单语言格式)
  - 可以调用 OpenAI / Deepseek 进行 AI 翻译 (花了 2 块钱)

## 新文档基础设施 (6): 反思

- 似乎麻烦都与 Markdown 相关.....如果换成了 reStructuredText ?
- reST 语法规范且丰富，没有也不需要方言
- reST 有主线处理工具 docutils，Pandoc 也有较好支持
- reST 的网站生成器 Sphinx 功能全面且好用，可用主题也更多
- ReadTheDocs、Weblate 等平台完美支持 Sphinx 流程
- .....
- 当然，我们实际上并没有跑过 reST 的完整流程，可能有别的坑.....

谢谢！



# 谢谢！

---

**香山社区：2024~2025**

香山开源处理器社区 / 北京开源芯片研究院 徐泽凡

2025 年 7 月 16日