RISC-V指令集开源软件生态

报告人：吴伟（ @lazyparser on Github/Bilibili）
中国科学院 软件研究所 智能软件研究中心 PLCT实验室

欢迎参与我们的开源项目 https://github.com/isrc-cas/

如果希望参与RISC-V开源软件项目开发，欢迎加我微信 fangzhang1024 询问 ☺

关于我

- 年少有志苦练**屠龙技**；如今天天写PPT；没有设计过CPU/SoC
- RISC-V：2016年开始关注、赞助、沙龙、演讲、**培训**
- **PLCT Lab**：编译器、虚拟机、模拟器
- HelloGCC、HelloLLVM、**OSDT**社区

本次报告观点仅代表个人及 PLCT Lab；不代表ISRC或ISCAS或国家

今天演讲的主题内容（Takeaway）

- RISC-V 已经成为未来的主流架构(之一)
  - 任何芯片公司都应投入；任何软件公司都需要适配；从业者应该立刻开始学习
- 拥抱开源软件并积极成为贡献者是企业生存、胜出的必要条件
  - 要么积极融入开源社区，要么被开源软件吞噬；不被upstream支持等于活在ICU
- RISC-V 软件生态领域依然有大量机会可以把握
  - 以中国科学院软件研究所PLCT实验室在RISC-V领域的贡献和成果为例

为什么突然冒出来一个 RISC-V？

- 其实收钱的、开源自由不要钱的ISA，过去现在未来都有
  - OpenRISC、SPARC V32、Wild MIPS
- 国内也有不少指令集，例如 C-SKY
  - 所有知名ISA国内都有备胎
- RISC-V 是众多新ISA脱颖而出的一个，「大家」都买账

思考：“大家”是谁？

RISC-V Timeline

RISC-V 已经成为未来的主流架构

是因为 RISC-V 在合适的时间、合适的地点、提出了足够好的设计

- 摩尔定律 → Domain-Specific Architecture
- DSA 需要：基本的软件支持、大量的定制指令编码空间

为什么在RISC-V之前那么多个ISA都没有成为主流？

RISC-V 已经成为未来的主流架构

是因为 RISC-V 在合适的时间、合适的地点、提出了足够好的设计

- 伯克利
- Krste 教授的研究组

如果 RISC-V 诞生在中国，我们还需要补上哪些技术人才？

RISC-V 已经成为未来的主流架构

是因为 RISC-V 在合适的时间、合适的地点、提出了足够好的设计

- 提出了模块化设计的概念
  - 最基础的RV32I仅使用了40+条指令编码：一名大三学生一周就能完成一个能跑的设计
- 提供了高度灵活的配置空间

图源：Krste 演讲： RISC-V State of Union https://www.youtube.com/watch?v=CKCIDe7f

RISC-V 已经成为未来的主流架构

是因为 RISC-V 在合适的时间、合适的地点、提出了足够好的设计

- 提出了模块化设计的概念
  - 最基础的RV32I仅使用了40+条指令编码：一名大三学生一周就能完成一个能跑的设计
- 提供了高度灵活的配置空间
- 同时平台标准提供了足够多的软件支持

图源：Krste 演讲： RISC-V State of Union https://www.youtube.com/watch?v=CKCIDe7f

RISC-V 已经成为未来的主流架构

是因为 RISC-V 在合适的时间、合适的地点、提出了足够好的设计

- 提出了模块化设计的概念
  - 最基础的RV32I仅使用了40+条指令编码：一名大三学生一周就能完成一个能跑的设计
- 提供了高度灵活的配置空间
- 同时平台标准提供了足够多的软件支持

图源：Krste 演讲： RISC-V State of Union https://www.youtube.com/watch?v=CKCIDe7f

RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
  - 在 RISC-V 之前，主流ISA对硬件自由创新禁止，尤其是64位
  - 相比于 OpenRISC、SPARC 等架构，RISC-V 提供了更好的可扩展的设计

图源：Krste 演讲： RISC-V State of Union

RISC-V Ecosystem

Software

Open-source software:
Gcc, binutils, glibc, Linux, BSD, LLVM, QEMU, FreeRTOS, ZephyrOS, LiteOS, SylixOS, …

Commercial software:
Lauterbach, Segger, IAR, Micrium, ExpressLogic, Ashling, AntMicro, Imperas, UltraSoC …

RISC-V Foundation

ISA specification | Golden Model | Compliance

Hardware

Open-source cores:
Rocket, BOOM, RI5CY, Ariane, PicoRV32, Piccolo, SCR1, Shakti, Serv, Swerv, Hummingbird, …

Commercial core providers:
Alibaba, Andes, Bluespec, Cloudbear, Codasip, Cortus, InCore, Nuclei, SiFive, Syntacore, …

Inhouse cores:
Nvidia, WD, +others

图源：Krste 演讲： RISC-V State of Uni

图源：Krste 演讲： RISC-V State of Uni

内卷警告：所有还在做自研的指令集的朋友，该思考下出路了

图源：Krste 演讲： RISC-V State of Uni

RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
- **设计优美简洁，支持了从教学、研究到工业界的无缝切换（懒是进步之源）**
  - 前人成功的案例，有Linux和LLVM
  - 在这之前，教育用MIPS/SPARC，进入产业之后变成X86/Arm，研究成果无法直接转化

图源：Krste 演讲： RISC-V State of Union

RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
- 设计优美简洁，支持了从教学、研究到工业界的无缝切换
- 左手成立RISC-V基金会，右手创办SiFive

RISC-V 自身做对了什么

- 左手成立RISC-V基金会，右手创办SiFive
  - 兼顾ISA中立和推动商业化

https://commons.wikimedia.org/wiki/File:Technology-Adoption-Lifecycle.png

RISC-V 自身做对了什么

- 基金会和SiFive：兼顾ISA中立和推动商业化

https://commons.wikimedia.org/wiki/File:Technology-Adoption-Lifecycle.png

RISC-V 自身做对了什么

- 清晰的定位：指令集自由开放，作为硬件和软件之间“交谈”的接口
- 设计优美简洁，支持了从教学、研究到工业界的无缝切换
- 左手成立RISC-V基金会，右手创办SiFive
- **顺应了时代的潮流：DSA导致的设计方式的改变，引发对ISA的新需求**
  - “RISC-V 的成功是因为（出现了）新的商业模式” —— Krste 教授

https://spectrum.ieee.org/view-from-the-valley/computing/hardware/david-patterson-says-its-time-for-new-computer-architectures-and-software-

那么，为何是伯克利搞成了RISC-V ，而不是中国的高校或企业？

- 一呼天下应的出身（代表着技术领域的信用货币）？
- 有钱、正确的烧钱（进行了十几次流片*）？
- 符合时宜的设计理念（模块化、定制化）？
- 简洁（连一个本科生都可以接近独立的设计出短流水线MCU）？
- 非常注重软件栈、借力？
- 天下桃李，悉在公门矣？

* https://blog.csdn.net/moxibingdao/article/details/106667615

那么，为何是伯克利搞成了RISC-V ，而不是中国的高校或企业？
我们缺少什么？如何补缺？如何追赶？

- 一呼天下应的出身（代表着技术领域的信用货币）？
- 有钱、正确的烧钱（进行了十几次流片*）？
- 符合时宜的设计理念（模块化、定制化）？
- 简洁（连一个本科生都可以接近独立的设计出短流水线MCU）？
- 非常注重软件栈、借力？
- 天下桃李，悉在公门矣？

* https://blog.csdn.net/moxibingdao/article/details/106667615

RISC-V 已经成为未来的主流架构：要么在车里，要么在车底

拥抱开源软件：软件吞噬世界，开源软件吞噬软件

1 https://developer.ibm.com/blogs/how-open-source-software-is-eating-the-world/
2 https://algorithmia.com/blog/wp-content/uploads/2016/06/Screen-Shot-2016-06-08-at-3.35.53-PM-1024x
3 https://developer.android.com/guide/platform

拥抱开源软件：软件吞噬世界，开源软件吞噬软件

是什么让开源不可避免？

软件系统的规模

1 https://developer.ibm.com/blogs/how-open-source-software-is-eating-the-world/
2 https://algorithmia.com/blog/wp-content/uploads/2016/06/Screen-Shot-2016-06-08-at-3.35.53-PM-1024x730.pn
3 https://developer.android.com/guide/platform

拥抱开源软件，或者被吞噬

- 开源软件正在吞噬世界
  - GNU/Linux、FreeRTOS、*BSD
  - Hadoop/Spark、Docker/k8s……
- **不被开源软件支持将付出极大的维护成本**
  - 进而失去竞争优势被淘汰

拥抱开源软件，或者被吞噬
开源软件的世界很残酷的，只有头部的社区容易吸引用户和贡献者

- **开源软件正在吞噬世界**
  - GNU/Linux、FreeRTOS、*BSD
  - Hadoop/Spark、Docker/k8s……
- **不被开源软件支持将付出极大的维护成本**
  - 进而失去竞争优势被淘汰

拥抱开源软件

- **成功的ISA架构在开源软件领域都有着巨大的投入**
  - Intel、Arm 有着长期巨大的投入，投入不够一定会被打败
  - 就像是武林高手对掌拼内力，谁先撤力谁出大问题

拥抱开源软件

- **成功的ISA架构在开源软件领域都有着巨大的投入**
  - Intel、Arm 有着长期巨大的投入，投入不够一定会被打败
- **RISC-V 做对了两件事让其成为未来的主流**
  - 早期的团队研发实力和伯克利的领袖光环，**实现了基本工具链支持**
  - 在2017年之后顺应产业界的需求从芯片创新转向以软件为中心

RISC-V 正确的转向

RISC-V

Changing Priorities

2010

1. Be simple, efficient, extensible
2. Revisit legacy design decisions
3. Have basic software

(computer-architecture-driven project)

2020

1. Run all software
2. Be feature complete
   - see #1
3. Be stable
   - see #1
4. Support innovation
   - conflicts with #1,#2,#3?

(software-driven project)

RISC-V 正确的转向

或许这就是
国内的技术领袖们
稀缺的远见

RISC-V 正确的转向：或许这是我们（国内）稀缺的远见

- 硬件出身的专家往往看不见**软件生态才是决定生态生死的国王**

- 而 RISC-V 初创团队是硬件出身，却能够
  在2010年就看到软件占据了系统成本的大头
  在2020年明确提出将从体系架构主导变更为软件主导

RISC-V 正确的转向：或许这是我们（国内）稀缺的远见

- 硬件出身的专家往往看不见软件生态才是决定生态生死的国王

- 而 RISC-V 初创团队是硬件出身，却能够
  在2010年就看到软件占据了系统成本的大头
  在2020年明确提出将从体系架构主导变更为软件主导

国内的许多企业，提前跑了多年，为什么没有成为RISC-V？

拥抱开源软件：怎么拥抱？

- 开源软件：会跑起来和会自己修改、演化是两个不同的境界
- 充分的消化、吸收，变成自己的知识

拥抱开源软件：怎么拥抱？

- 开源软件：会跑起来和会自己修改、演化是两个不同的境界
  - 「国外一开源，国内就自主可控」是劣币驱逐良币
- 充分的消化、吸收，变成自己的知识
  - 尊重知识消化吸收的规律，避免拔苗助长

拥抱开源软件：怎么拥抱？

- 开源软件：会跑起来和会自己修改、演化是两个不同的境界
  - 「国外一开源，国内就自主可控」是劣币驱逐良币
  - 国内想要超越，就不能一直搭顺风车：请**不要裁掉基础软件工程师！**
- 充分的消化、吸收，变成自己的知识
  - 尊重知识消化吸收的规律，避免拔苗助长
  - **技术结果导向**，在早期识别浑水摸鱼的南郭先生
  - **能者多得**，而不仅仅是能者多劳

面向RISC-V的开源软件生态：国内的贡献度起步迟了

- **重要的开源软件的 RISC-V Ports 都已经有志愿者维护**
  - GNU Toolchain、glibc/newlib、Linux Kernel、Clang/LLVM
- **谁在上游（Upstream）贡献了代码，谁就掌控了话语权**
  - 欧洲的EPI和ZTH、北美的Google和WDC、印度的IIT、中国台湾的Andes

面向RISC-V的开源软件生态：国内的贡献度起步迟了

- 重要的开源软件的 RISC-V Ports 都已经有志愿者维护
  - GNU Toolchain、glibc/newlib、Linux Kernel、Clang/LLVM
- 谁在上游（Upstream）贡献了代码，谁就掌控了话语权
  - 欧洲的EPI和ZTH、北美的Google和WDC、印度的IIT、中国台湾的Andes
- **不被上游维护的代码就像是活在ICU里：费用昂贵、死亡率高**
  - 举例：几乎所有国内的芯片厂商和系统厂商

面向RISC-V的开源软件生态：仍有大量机会，来吧，少年！

男女不限、年龄不限、内心有一团火

面向RISC-V的开源软件生态：仍有大量机会，来吧，少年！

- **基础软件领域的狂欢：已经很久没有这么多轮子可以发明了**
  - 各种编程语言的编译器 x N
  - 语言虚拟机和运行时 x N
  - 系统虚拟机、指令集模拟器 x N
  - 全系统性能分析和优化、软硬件协同演进的定制和优化
  - 二进制工具 x N
  - 全新的系统安全视角和可能性

    请务必看看 bunnie 的 keynote https://www.youtube.com/watch?v=zXwy65d_tu8

国内的企业有哪些机会？

- MCU领域已经形成了成功的模式
- 通用计算（PC/Server）和AIoT领域正在发力
- 纯CPU创业会很辛苦，因为有大财团互联网公司在旁边准备投你的友商
- 各类AI加速器初创公司还有机会，但是如何面对NVDLA的坚壁清野
- 目前国内的差距主要体现在软件能力，~~而不是硬件研发能力上(我不清楚)~~
- 但是纯做开源软件并不能养活自己，这是个问题

ISCAS 中国科学院软件研究所 Institute of Software Chinese Academy of Sciences
智能软件研究中心 Intelligent Software Research Center

RISC-V生态的边界正在经历一个商业场景的变迁：卖硬件到卖系统

Open-source software:
Gcc, binutils, glibc, Linux, BSD, LLVM, QEMU, FreeRTOS, ZephyrOS, LiteOS, SylixOS, ...

Commercial software:
Lauterbach, Segger, IAR, Micrium, ExpressLogic, Ashling, AntMicro, Imperas, UltraSoC ...

RISC-V Foundation | ISA specification | Golden Model | Compliance

Hardware

Open-source cores:
Rocket, BOOM, RI5CY, Ariane, PicoRV32, Piccolo, SCR1, Shakti, Serv, Swerv, Hummingbird, ...

Commercial core providers:
Alibaba, Andes, Bluespec, Cloudbear, Codasip, Cortus, InCore, Nuclei, SiFive, Syntacore, ...

Inhouse cores:
Nvidia, WD, +others

Open-source software:
Gcc, binutils, glibc, Linux, BSD, LLVM, QEMU, FreeRTOS, ZephyrOS, LiteOS, SylixOS, ...

Commercial software:
Lauterbach, Segger, IAR, Micrium, ExpressLogic, Ashling, AntMicro, Imperas, UltraSoC ...

RISC-V Foundation | ISA specification | Golden Model | Compliance

Hardware

Open-source cores:
Rocket, BOOM, RI5CY, Ariane, PicoRV32, Piccolo, SCR1, Shakti, Serv, Swerv, Hummingbird, ...

Commercial core providers:
Alibaba, Andes, Bluespec, Cloudbear, Codasip, Cortus, InCore, Nuclei, SiFive, Syntacore, ...

面向RISC-V的开源软件生态：以软件所的贡献为例

- **基础软件领域的国家队，有义务为国内企业提供「开源软件公共品」**
  - 编译器领域：RISC-V 向量扩展的LLVM参考实现
  - 虚拟机领域：V8 for RISC-V （V8 是 Chrome 使用的JS引擎）
  - 模拟器领域：面向国内厂商的QEMU、Spike支持
  - 应用领域：OpenCV for RISC-V
  - 操作系统领域：开源供应链安全计划、点亮计划、暑期2020计划
    （软件所操作系统团队成果，非PLCT工作）

PLCT Lab 的使命

程序语言与编译技术实验室（PLCT）致力于成为编译技术领域的开源领导者，推进开源工具链及运行时系统等软件基础设施的技术革新，具备主导开发和维护重要基础设施的技术及管理能力。与此同时，努力成为编译领域培养尖端人才的黄埔军校，推动先进编译技术在国内的普及和发展。

RISC-V Vector Extension in LLVM

https://github.com/isrc-cas/rvv-llvm

- 实现RISC-V 向量扩展指令集支持，维护适用于国内RISC-V芯片厂商的定制版本
- 以RVV作为切入点，逐步参与RISC-V基金会技术决策
- 为本土小芯片厂商提供工具链及开发工具的支持，填补短板
- 作为实战培训项目培养编译器开发新员工并壮大团队

V8 for RISC-V

https://github.com/v8-riscv/v8

- Google V8 是目前全球占有率第一大 JavaScript 引擎，是 Node.js 生态领域的基础
  - V8 和 Node.js 的 RISC-V 支持被列在了 RISC-V 基金会的官网许愿清单中
- 建立一支掌握虚拟机关键技术的小队，能够为国内企业提供支持
- 尽可能多的培养掌握虚拟机开发能力的新人

面向国内RISC-V厂商的QEMU支持计划

https://github.com/isrc-cas/plct-qemu

- 为目前国内RISC-V初创企业提供QEMU模拟器的技术支持
- 建立一支有能力快速掌握QEMU核心技术的小队，并形成培养梯队
- 为国内芯片厂商输送模拟器、仿真器方面的技术骨干

QEMU

Optimize OpenCV for RISC-V

https://github.com/opencv/opencv

- 融入OpenCV社区，成为OpenCV社区在RV方面的贡献者
- 与OpenCV的国内研发团队形成战略合作，相互扶持
- 初步完成 RISC-V 向量扩展的支持，随 v4.5 之后的版本发布（大概）
  - 正在进行进一步的bugfix和性能的优化
  - 由于 RISC-V 向量扩展仍在演化中，OpenCV项目也会持续更新和维护

OpenCV

敬请关注：开源软件供应链点亮计划（软件所操作系统团队成果，非PLCT工作）

对标谷歌暑期编程计划
GSoC

- 活动规模：34 所高校、42 个开源社区、92 家组织机构、246 位社区核心开发人员（社区导师）、388 个开源项目
- 技术范围：8 大技术方向，操作系统、内核与编译器、人工智能、分布式系统、大数据与数据管理、容器与虚拟化、Web 技术、编程开发
- 华为参与：7个开源社区（占比16.7%），126个开源项目（占比32.4%），5大技术方向（操作系统、内核与编译器、容器与虚拟化、分布式系统、人工智能）
- 已有成果：146 位同学提交了结项报告，已有 3 位同学入选 Apache APISIX 及社区 Committer

暑期2020活动已经圆满收官，明年将重点支持RISC-V生态内的开源软件

观看PLCT/OSDT的技术分享、一起学习和进步

- https://space.bilibili.com/296494084/video
- 已更新技术分享150多个，每周三技术分享后上传
- 每周五、周六下午3点有编译技术入门与实战讨论班（线上），欢迎加入

每个月1号和16号，「PLCT开源进展」半月刊更新最新工作进展

- https://www.zhihu.com/column/plct-lab
  - https://github.com/isrc-cas/PLCT-Weekly

软件所PLCT实验室

小乖他爹 · 34 篇内容 •••

PLCT开源进展·第10期·2020年10月01日

卷首语 感谢最近刚刚进入RISC-V国际基金会董事会的包云岗老师，将我们引荐给了基金会的CTO并进行了几次技术交流。现在，RISC-V基金会终于知道软件所PLCT实验室的存在了。 通过这几次会议，我才注意到所有的RISC-V基金会的技术会议实际上都是公开的。不仅有列表列出了所有的时间，甚至提供了一个 ICS 日历文件，让我... 阅读全文

▲ 赞同 26 ▼ 1 条评论 分享 收藏

欢迎使用、参与开发、提问、报告bugs、提 Pull Requests (^o^)/

- PLCT Weekly https://github.com/isrc-cas/PLCT-Weekly
- C910 LLVM 支持 https://github.com/isrc-cas/c910-llvm
- RISC-V Vector Extension Support（rvv-llvm） https://github.com/isrc-cas/rvv-llvm
- V8 for RISC-V https://github.com/v8-riscv/v8
- 永恒天平 https://github.com/isrc-cas/eternal-balance/
- PLCT 公开报告 https://github.com/isrc-cas/PLCT-Open-Reports
- QuickJS for RISC-V https://github.com/isrc-cas/quickjs-riscv
- PLCT-QEMU https://github.com/isrc-cas/plct-qemu
- Flounder https://github.com/isrc-cas/flounder
- OpenCV for RISC-V https://github.com/isrc-cas/opencv-riscv
- pacific（方舟编译器玩具运行时） https://github.com/isrc-cas/pacific
- rvv-benchmark https://github.com/isrc-cas/rvv-benchmark
- 工具箱 https://github.com/isrc-cas/PLCT-toolbox

谢 谢 ！

- 报告人：吴伟（ @lazyparser on Github/Bilibili）
- 中国科学院 软件研究所 智能软件研究中心 PLCT实验室
- 欢迎参与我们的开源项目 https://github.com/isrc-cas/

如果希望参与RISC-V开源软件项目开发，欢迎加我微信 fangzhang1024 询问 ☺