

开放原子开源大赛

面向RISC-V架构的AI开发框架

2023-2024  
CONTEST

构建与优化大赛



# 01 赛事介绍

## 关于openKylin

openKylin（开放麒麟）社区是由基础软硬件企业、非营利性组织、社团组织、高等院校、科研机构和个人开发者共同创立的一个开源社区，旨在以“共创”为核心、以“开源聚力、共创未来”为社区理念，在开源、自愿、平等、协作的基础上，通过开源、开放的方式与企业构建合作伙伴生态体系，共同打造桌面操作系统顶级社区，推动Linux 开源技术及其软硬件生态繁荣发展。

openKylin自成立起，就从源码级别自主构建了 RISC-V 开源桌面操作系统版本，目前能够支持 HiFive Unmatched、VisionFive2、LicheePi 和 SG2042 EVB 等多个硬件平台，支持 80% 以上市场主流 RISC-V 芯片，并与深度数智联合发布了全球首款搭载RISC-V笔记本电脑ROMA。同时，openKylin 社区研发团队还基于RISC-V指令集和GPU优化了UKUI 桌面环境，并适配了openKylin自研软件、Libreoffice、浏览器、影音播放和编程开发等常用软件。

## 赛事介绍

当今人工智能技术已经广泛应用于诸多领域，并呈现出快速发展的态势，而RISC-V架构则是一个具有潜力的开放式指令集架构（ISA）标准。面向RISC-V架构的AI开发框架构建与优化大赛是开放原子开源大赛设立的赛道，大赛由开放原子开源基金会、央视网、江苏省工业和信息化厅等承办，赛道由openKylin社区、麒麟软件有限公司承办，国防科技大学、信创海河实验室协办，玄铁、北京算能科技有限公司、广东赛昉科技有限公司赞助。通过本项赛事，旨在探索如何将人工智能技术与RISC-V架构相结合，以实现高性能和低功耗的目标，推动人工智能技术在不同领域的应用，提升AI算法在RISC-V架构下的运行性能和降低能耗，推进AI芯片的发展。

### 大赛承办单位



无锡高新区

...

### 赛道承办单位



### 协办单位



### 赞助单位



# 专家委员会

开源聚力，共创未来  
Open source cohesion ,co-creation future

openKYLin



主任

尼玛扎西

中国工程院院士、  
西藏大学计算机学院院长



陈渝

清华大学长聘副教授



周明辉

北京大学教授



章毅

四川大学教授、院长



张玉志

南开大学教授、院长



胡春明

北京航空航天大学教授、院长



于策

天津大学教授、副主任



杨建华

浙江大学教授



顾钊铨

哈尔滨工业大学教授



吴庆波

国防科技大学研究员



武延军

中科院软件所研究员、副所长



孙乐

中科院软件所研究员



乔保军

河南大学教授、院长



赵俊峰

内蒙古大学教授、副院长



陈莉君

西安邮电大学教授



刘宏

湖南师范大学教授



王伟

华东师范大学教授



张锦

长沙理工大学教授、院长



阳旺

中南大学副教授



肖景

湖南大学副教授



刘澎

中国开源软件推进联盟副主席  
兼秘书长



谭中意

开放原子开源基金会技术  
委员会主席



庄表伟

天工开物开源基金会执行秘书长、  
开源社执行长



陈阳

微软开源技术专家、  
开源社理事长



高鹏

算能科技有限公司高级副总裁



李春强

阿里巴巴达摩院RISC-V及生态软件  
生态负责人



孙彦邦

杭州进迭时空科技有限公司总裁



韩乃平

麒麟软件有限公司首席科学家、  
研高工



张铎

麒麟软件有限公司高级副总经理、  
高工



李震宁

麒麟软件有限公司副总经理、  
正高工



余杰

国防科技大学研究员、  
openKylin秘书长



## 赛事特点

### 等级高

开放原子开源大赛赛道

### 规模大

赛事邀请了多名院士以及权威机构坐镇，评委由学术界的教授、学者以及头部企业的专家担任，权威性有所保障

### 实用性强

比赛成果将进入openKylin操作系统软件仓库供全球用户下载使用

### 奖励丰富

大赛设有50万奖金以及若干数码奖励

## 往期大赛风采

# “麒麟杯”全国开源应用软件 开发大赛

该赛事已连续举办十二届，累计覆盖数百所高校，  
上千支队伍参与。大赛旨在加强高校人才和信息产业的衔接，培养更多优秀软件开发人才，培育一批优秀开源项目，进一步丰富国产操作系统的应用生态，共建信息技术应用创新产业生态，加快推进信息技术自主创新水平。

**500+**所高校

**100+**场技术活动

**5000+**支赛队

**7000+**位开发者

### 指导单位



### 承办单位

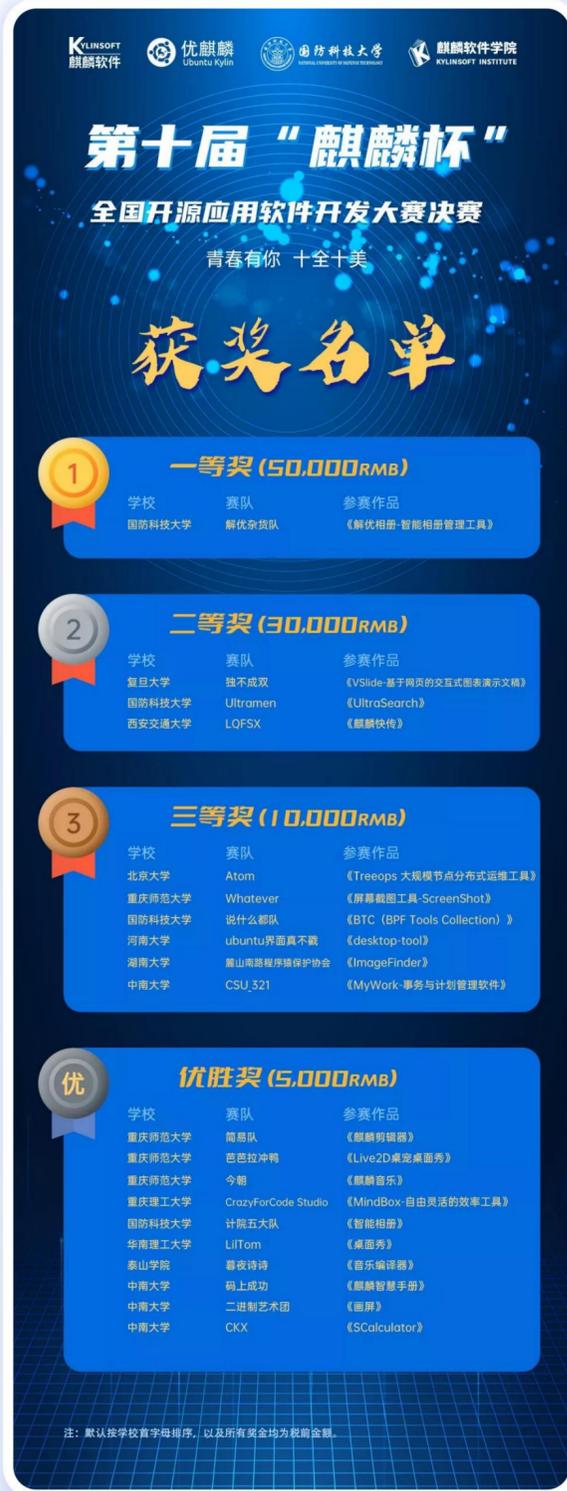


### 合作单位



### 赞助单位





**第十届“麒麟杯”**  
全国开源应用软件开发大赛决赛  
青春有你 十全十美

## 获奖名单

**1 一等奖 (50,000RMB)**

学校	赛队	参赛作品
国防科技大学	解忧杂货队	《解忧相册-智能相册管理工具》

**2 二等奖 (30,000RMB)**

学校	赛队	参赛作品
复旦大学	独不成双	《VSlide-基于网页的交互式图表演示文稿》
国防科技大学	Ultramen	《UltraSearch》
西安交通大学	LOFSX	《麒麟快传》

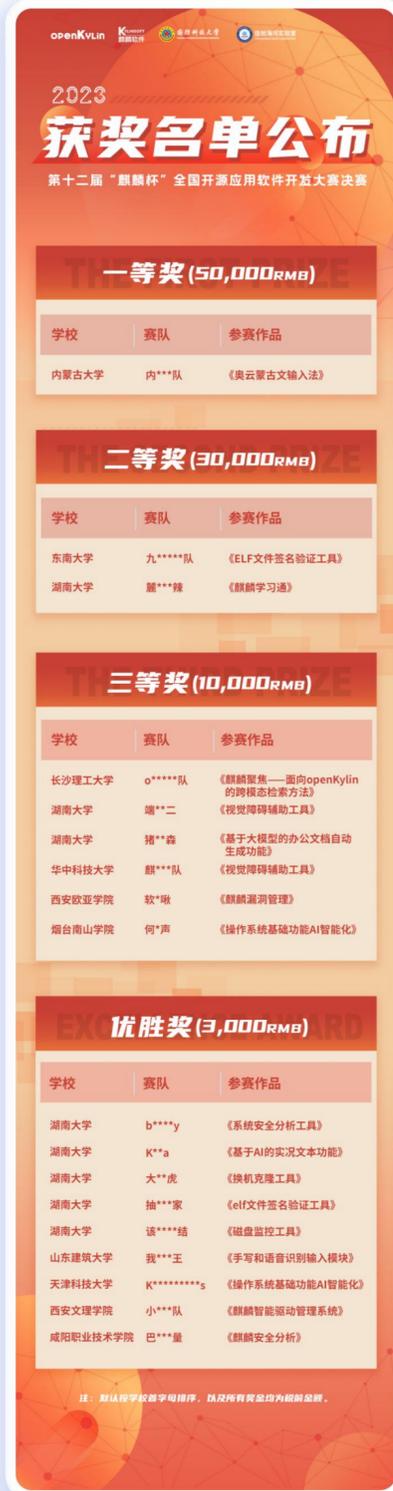
**3 三等奖 (10,000RMB)**

学校	赛队	参赛作品
北京大学	Atom	《Treeops 大规模节点分布式运维工具》
重庆师范大学	Whatever	《屏幕截图工具-ScreenShot》
国防科技大学	说什么都是	《BTC (BPF Tools Collection)》
河南大学	ubuntu界面真不美	《desktop-tool》
湖南大学	湖南南程序保护协会	《ImageFinder》
中南大学	CSU_321	《MyWork-事务与计划管理软件》

**优 优胜奖 (5,000RMB)**

学校	赛队	参赛作品
重庆师范大学	周易队	《麒麟剪辑器》
重庆师范大学	芭芭拉冲锋	《Live2D桌宠桌面秀》
重庆师范大学	今朝	《麒麟音乐》
重庆理工大学	CrazyForCode Studio	《MindBox-自由灵活的效率工具》
国防科技大学	计院五大队	《智能相册》
华南理工大学	LITom	《桌面秀》
泰山学院	暮夜诗诗	《音乐编辑器》
中南大学	码上成功	《麒麟智慧手册》
中南大学	二进制艺术团	《画廊》
中南大学	CKX	《SCalculator》

注：默认按学校首字母排序，以及所有奖金均为税前金额。



**2023 获奖名单公布**  
第十二届“麒麟杯”全国开源应用软件开发大赛决赛

**一等奖 (50,000RMB)**

学校	赛队	参赛作品
内蒙古大学	内***队	《奥云蒙古文输入法》

**二等奖 (30,000RMB)**

学校	赛队	参赛作品
东南大学	九***队	《ELF文件签名验证工具》
湖南大学	麓***楼	《麒麟学习通》

**三等奖 (10,000RMB)**

学校	赛队	参赛作品
长沙理工大学	o****队	《麒麟聚焦——面向openKylin的跨模态检索方法》
湖南大学	端**二	《视觉障碍辅助工具》
湖南大学	猪**森	《基于大模型的办公文档自动生成功能》
华中科技大学	麒**队	《视觉障碍辅助工具》
西安欧亚学院	软**墩	《麒麟漏洞管理》
烟台南山学院	何**声	《操作系统基础功能AI智能化》

**EX 优胜奖 (3,000RMB)**

学校	赛队	参赛作品
湖南大学	b****y	《系统安全分析工具》
湖南大学	K**a	《基于AI的实况文本功能》
湖南大学	大**虎	《换机克隆工具》
湖南大学	抽**家	《elf文件签名验证工具》
湖南大学	语***结	《磁盘监控工具》
山东建筑大学	我***王	《手写和语音识别输入模块》
天津科技大学	K****队	《操作系统基础功能AI智能化》
西安文理学院	小***队	《麒麟智能驱动管理系统》
咸阳职业技术学院	巴***量	《麒麟安全分析》

注：默认按学校首字母排序，以及所有奖金均为税前金额。

2021年，**第十届“麒麟杯”全国开源应用软件开发大赛**有来自全国 35 所高校的 102 组参赛队伍报名。大赛指导委员会邀请了**倪光南、陆首群、吾守尔·斯拉木、郑纬民、廖湘科**等院士、专家助阵，并由来自计算机开源和教育领域的**17 位知名专家组成评审**团队进行综合评审，促进了一批优秀开源项目的诞生与发展。

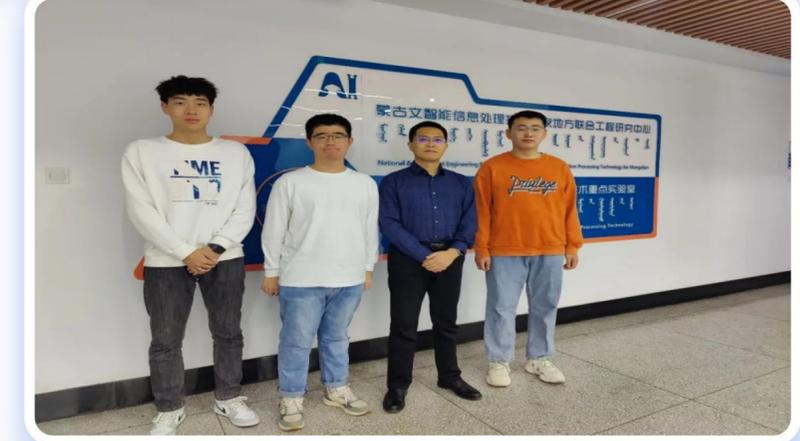
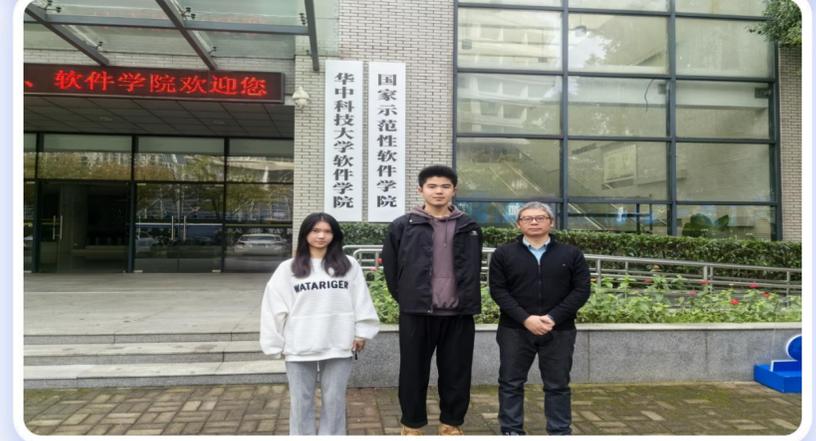
2022年，**第十一届麒麟杯大赛**作为第五届中国软件开源创新大赛“开源项目创新”赛道重要组成部分，共有**4409 支队伍报名参赛**，覆盖全国 648 所高校，其中包括清华大学、湖南大学、华中科技大学、国防科技大学等在内的 31 所985高校、76 所211高校以及 468 所本科院校。

2023年，**第十二届麒麟杯大赛**经过6个月的角逐已圆满落幕，累计覆盖全国上百所高校，共举办10余场线下宣讲活动，累计吸引全国**60 余所高校 345 支参赛队伍参与**。





# 往期大赛 开发者风采





## 02 赛题介绍

## 赛题任务

本赛题要求开发者能够基于openKylin开源操作系统和 RISC-V 硬件平台：

- 1) **框架适配**：适配运行一款以上主流AI框架，包括但不限于TensorFlow、PyTorch、Caffe、Keras、MXNet、PaddlePaddle、Mindspore等；
- 2) **应用开发**：基于适配的AI框架，能够运行有代表性的AI模型，或者开发AI应用软件，使AI应用能够运行在openKylin系统上；
- 3) **框架优化**：针对RISC-V芯片硬件特性，研究面向RISC-V架构的AI框架加速优化技术，从而有效利用RISC-V硬件特性为人工智能任务的运行提供高效的AI基础环境支撑。（此项可选，加分项）



## 旨在探索如何将人工智能技术与RISC-V架构相结合，以实现高性能AI应用的目标

### AI加速优化 技术层面

针对RISC-V架构中可能存在的性能瓶颈，可以提出针对性的优化策略，如指令流水线优化、指令重排、硬件加速等技术，从而提高AI框架在RISC-V架构下的运行效率

### AI应用 功耗层面

在当前移动端设备、物联网和边缘计算等应用场景下，功耗已经成为限制因素之一，研究面向RISC-V架构的AI框架适配与优化技术可以提高功耗效率，降低能耗，使得在低功耗情况下运行AI应用成为可能。

### AI芯片 发展层面

针对RISC-V架构特点进行优化，可以有效推动AI芯片的发展和普及，降低成本，使得更多的企业和开发者能够使用AI芯片

### AI应用 市场层面

将RISC-V架构与AI技术相结合可以推动RISC-V在此领域的应用，进一步扩大RISC-V架构的市场



PS：具体流程以活动官网为准

- (一) 报名阶段：**各参赛队伍在大赛网站在线报名。
- (二) 初赛阶段：**初赛提交内容包括但不限于：参赛作品（作品申报书、作品介绍PPT、其它展示产品/解决方案的主要内容、效果等）。
- (三) 初赛评审：**评审专家根据初赛评审规则打分，排名前20名的队伍将进入决赛。
- (四) 决赛阶段：**晋级决赛的队伍，可在初赛作品基础上进一步优化和完善。
- (五) 决赛评审：**按评审细则进行打分，最终团队成绩，将在总决赛中的作品答辩环节的综合评分后得出评出各类奖项。

# 大赛报名



## 报名说明

- 赛制: 初赛-决赛
- 参赛对象: 国内外各相关单位、高校、个人等均可免费报名参赛, 自由组队, 人数不限
- 所有参赛成员 (队长、成员), 均需签署社区CLA (签署个人CLA (cla.openkylin.top))
- 为每个参赛队提供RISC-V云平台或开发板



扫一扫 进入活动页面报名

添加好友, 并发送“开放原子开源大赛”入群交流

## 大赛官网

[https://www.openkylin.top/community/ai\\_contest\\_24.html](https://www.openkylin.top/community/ai_contest_24.html)



高校



企业



个人

# 作品提交

## 提交方式

## 初赛提交

## 决赛提交

入围决赛选手按照竞赛专家组建议，进一步完善作品，细化相关文档

### 大赛作品提交地址为：

<https://atomgit.com/openkylin>

要求参赛队伍在AtomGit上建仓存放作品资料，提交作品前需签署社区个人 CLA

(<https://cla.openkylin.top>)，并将仓库链接同步给到工作人员。



### 项目提交

初赛提交&决赛提交

作品申报书

作品介绍PPT

其他展示产品

解决方案的主要内容、效果等

- a. 作品标题、摘要、软件分类、应用领域、开放源码组织认可的开放源码许可证类型（可多个）等软件基本信息、作品概述等。
- b. 软件背景及应用领域；
- c. 同类软件国内外现状及分析对比；
- d. 作品特点和设计思路；
- e. 功能描述；
- f. 体系结构和关键技术点；
- g. 功能模块设计；
- h. 软件部署及维护说明；
- i. 测试报告用户手册；
- j. 总结。

## 奖项设置



**50万** 奖金花落谁家?

二等奖

4组

**4W**

openKylin社区暑期实习

一等奖

2组

**8W**

openKylin社区offer

优秀奖

优秀指导奖

2000元

三等奖

9组

**2W**

openKylin社区面试直通车

为进一步提升赛题吸引力，出题单位为获奖队伍提供绿色招聘通道：

一等奖: openKylin社区offer, 额外赠送每组2台ROMA笔记本;

二等奖: openKylin社区暑期实习;

优秀奖: (若干组) openKylin 官方电子证书

三等奖: openKylin社区终面机会。

优秀指导奖: (若干名) openKylin 官方电子证书

**额外奖励计划**

## 经验++

扩展产品项目研发经验、创新意识和团队精神



## 能力++

专业开发工程师关于Qt、Java、Python等系列辅导课程



## 升学++

鼓励参赛学生群体在操作系统与软件领域的升学

## 就业++

创造参赛学生群体在信息技术相关行业的就业机会



## 奖励++

通过奖金、成果分享等多种形式回馈学生的学习与生活



# 作品要求

1

参赛作品须符合本届大赛赛题要求并**符合所报赛题方向**，作品名称应能**体现赛题主要特征**。

2

参赛作品必须为**原创作品**，并未侵犯任何他人的专利权、著作权、商标权及其他知识产权，且不得违反国家相关法律法规，否则将取消参赛资格。

3

参赛作品须**开放源代码**，遵循开源项目的开源许可证，如GPL、MIT、BSD、Mozilla、LGPL、MulanPSL等

4

作品应能**正确运行**并可达到预期结果。作品应与设计文档描述的功能一致，如未能实现设计文档中描述的所有功能，应注明未实现功能及其所占比例和重要程度。

5

作品必须可分离原开发环境，**具备可演示的运行环境**，能够在组委会提供的测试账户上编译/运行或自行提供编译/运行环境。

6

作品允许使用他人开源代码，但必须在保持原有版权的基础上，**注明出处**和其他开源代码的依赖关系。提交作品时，必须在分析设计文档中明确说明作品中其他开源代码的作用及所占比例。

7

参赛作品应具备解决面向RISC-V架构的AI开发框架构建的相关能力，应具备支持运行业界主流 AI 模型的能力

8

参赛作品应具备基于适配的AI框架，开发AI应用软件或使AI应用能够运行在openKylin系统上的能力

9

参赛作品的创新性、实用性及产品落地应用可行性

10

参赛团队需要提供足够详细的说明文档指导其他开发者运行/使用这个项目

11

参赛作品具备AI开发框架的加速优化能力，且性能较优化前方案的有明显提升



# 03 案例实践

# 实践流程



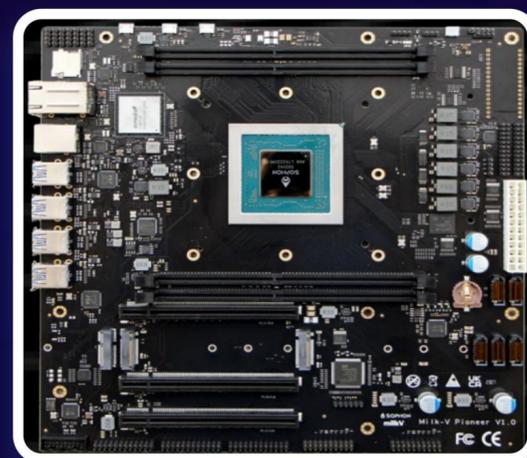
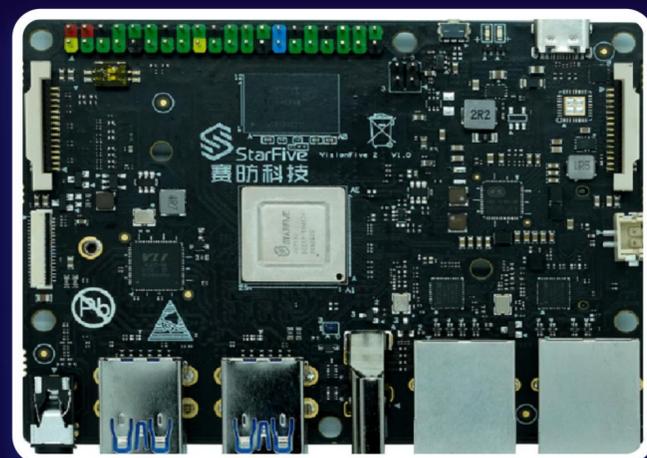
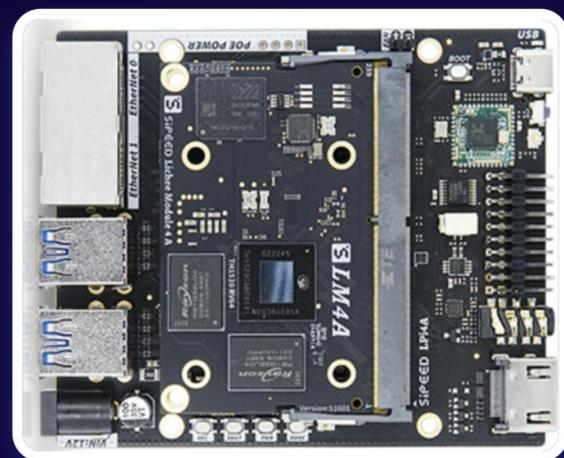
准备编译环境



编译/运行AI框架



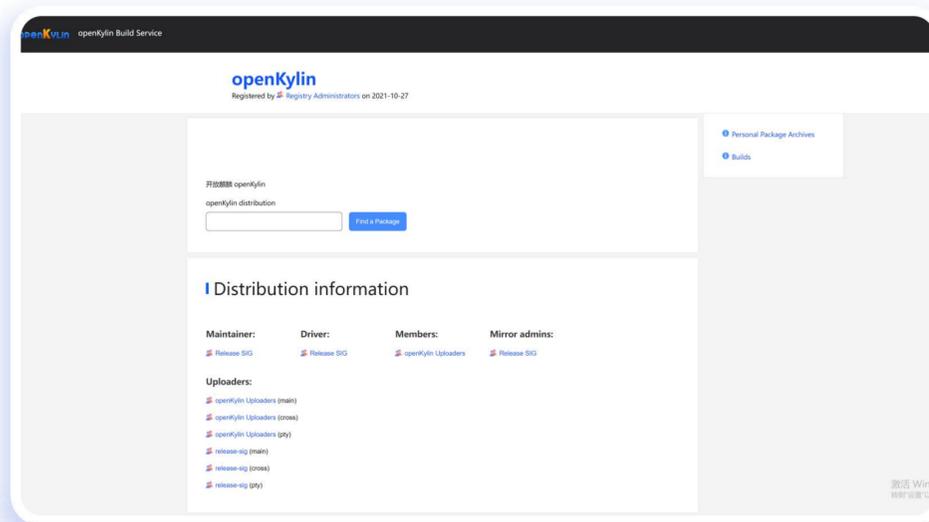
应用开发/框架优化





## 01. 本地/虚拟机编译

本地编译是指在搭载openKylin系统的RISC-V主机上进行编译，提供算能EVB开发板远程访问支持。



## 02. OKBS远程编译

远程编译是利用OKBS平台进行编译，不需要在本地进行编译操作，使用和配置参考openKylin社区。



## 03. 远程容器编译

通过ssh远程登录算能RISC-V服务器，支持20人并发访问，平均每个用户32个core，编译环境为独立的docker容器。

## ➤ 步骤1：填写资产领用申请表

下载渠道：大赛官网-赛事资料详情页

获取地址：

[https://www.openkylin.top/community/ai\\_contest\\_24.html](https://www.openkylin.top/community/ai_contest_24.html)



## ➤ 步骤2：提交资产领用申请表

可申领资产：赛昉、荔枝派开发板。  
将申请表信息填写完整后，发送至  
openKylin（开放麒麟）官方邮箱：  
contact@openkylin.top

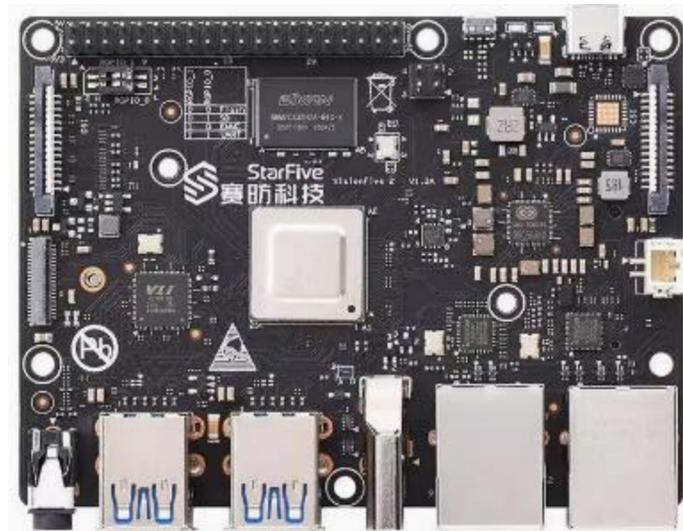
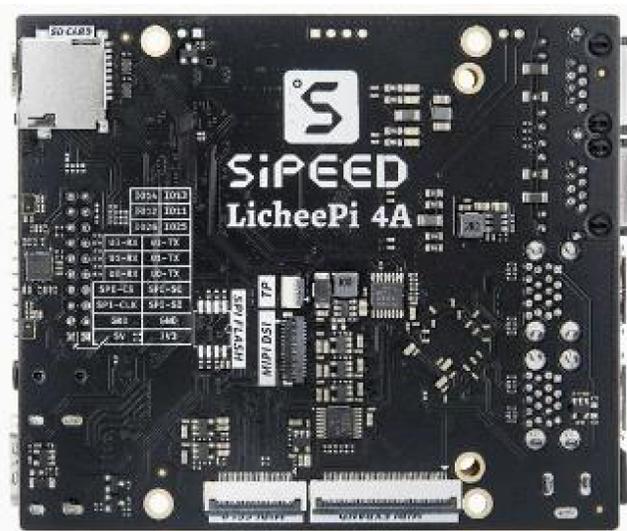
社区将信息登记核对后将开发板寄送给  
参赛开发者

## ➤ 步骤3：发送资产签收凭据

请申请人在收到开发板后拍照确认，并手写  
“我已收到开发板+本人姓名”发送至社区官方  
邮箱（contact@openkylin.top）作为签收凭据  
(开发板实物图+手写凭证)进行存档。

## 步骤4：注意事项

由于赛昉、荔枝派开发板成本较高，请申请领用的开发者们妥善保管(开发板+包装盒等所有零件)，并在赛事结束后尽快归还。



## 步骤5：开发板寄回

退还前请确保开发板上存储的数据已转移或备份。

联系社区资产管理小K(电话&微信同号：17863922737)进行开发板退还登记。

## 免责声明

- 1、领用资产仅限 面向RISC-V架构的AI开发框架构建与优化大赛赛事期间提供使用，并非免费赠送，请同学们提前规划好使用时间，切勿将重要数据保存在短期借用的开发板上。社区不承担任何关于数据资料丢失的责任。
- 2、严禁使用openKylin社区产品进行任何商业或者违法行为，一经发现会立即回收相应资源，并保留进一步追责的权利。

# 本地/虚拟机编译例子—Pytorch

## 步骤1：本机安装编译工具链

```
sudo apt-get install libblas-dev m4 cmake cython3 ccache
```

## 步骤2：下载Pytorch源代码

```
git clone --recursive https://github.com/pytorch/pytorch.git  
cd pytorch  
git submodule sync  
git submodule update --init --recursive
```

## 步骤3：修改编译配置文件

### 1、修改aten/src/ATen/CMakeLists.txt文件：

```
@@ -368,7 +368,8 @@ endif(MKLDNN_FOUND)  
368 368  
369 369 list(APPEND ATen_CPU_DEPENDENCY_LIBS cpuinfo)  
370 370  
371 - if(NOT MSVC AND NOT EMSCRIPTEN AND NOT INTERN_BUILD_MOBILE)  
371 + if(FALSE)  
372 + #if(NOT MSVC AND NOT EMSCRIPTEN AND NOT INTERN_BUILD_MOBILE)  
372 373 # Preserve values for the main build  
373 374 set(_aten_sleef_build_shared_libs ${BUILD_SHARED_LIBS})  
374 375 set(_aten_sleef_build_tests ${BUILD_TESTS})
```

### 2、修改caffe2/CMakeLists.txt文件：

```
@@ -1664,7 +1664,8 @@ if(BUILD_TEST)  
1664 1664 if(NOT MSVC)  
1665 1665 add_executable(${test_name}_${CPU_CAPABILITY} "${test_src}" ../aten/src/ATen/native/quantized/AffineQuantizerBase.cpp)  
1666 1666 # TODO: Get rid of c10 dependency (which is only needed for the implementation of AT_ERROR)  
1667 - target_link_libraries(${test_name}_${CPU_CAPABILITY} c10 sleef gtest_main)  
1667 + # target_link_libraries(${test_name}_${CPU_CAPABILITY} c10 sleef gtest_main)  
1668 + target_link_libraries(${test_name}_${CPU_CAPABILITY} c10 gtest_main)  
1668 1669 if(USE_FBGEMM)  
1669 1670 target_link_libraries(${test_name}_${CPU_CAPABILITY} fbgemm)  
1670 1671 endif()
```

### 3、修改test/cpp/api/CMakeLists.txt：

```
@@ -47,6 +47,7 @@ if(USE_CUDA)  
47 47 endif()  
48 48  
49 49 add_executable(test_api ${TORCH_API_TEST_SOURCES})  
50 + target_compile_options(test_api PUBLIC -Wno-nonnull)  
50 51 target_include_directories(test_api PRIVATE ${ATen_CPU_INCLUDE})  
51 52 target_link_libraries(test_api PRIVATE torch gtest)  
52 53 if(NOT MSVC)
```

## 本地/虚拟机编译例子—Pytorch

### ➤ 步骤4：配置环境变量并编译

```
1 export USE_CUDA=0 #不编译 CUDA 版本
2 export USE_DISTRIBUTED=0 #不支持分布式
3 export USE_MKLDNN=0 #不支持 MKLDNN
4 export MAX_JOBS=16 #最大线程数，按需填写
```

### ➤ 步骤5：源代码编译

```
sudo python3 setup.py develop --cmake
```

参考：

知乎

首发于  
RISC-V && AI

#### 如何在RISC-V平台编译pytorch



山大智研院  
山东大学 软件学院教授

知乎

首发于  
RISC-V && AI

#### 如何在RISC-V平台编译paddle



山大智研院  
山东大学 软件学院教授

## 本地/虚拟机编译例子—TensorFlow

### ➤ 步骤1：安装Python和TensorFlow软件包依赖

```
sudo apt install python3-dev python3-pip
pip install -U --user pip numpy wheel
pip install -U --user keras_preprocessing --no-deps
```

### ➤ 步骤2：安装Bazel

TensorFlow的编译使用Google的自研工具bazel，由于bazel和debian打包逻辑相反，经过Debian-Med 团队和 Google 的共同努力，将bazel打包为bazel-bootstrap，目前最新版本为4.2.3，并支持riscv64运行，debian地址为Index of /debian/pool/main/b/bazel-bootstrap/ ([ustc.edu.cn](http://ustc.edu.cn))，已上传到PPA [okbs:~haihe-riscv64-team/tensorflow-depend](https://ppa.launchpad.net/okbs/~haihe-riscv64-team/tensorflow-depend)中；

bazel的安装，共涉及到约50个依赖包，已上传到PPA [okbs:~haihe-riscv64-team/tensorflow-depend](https://ppa.launchpad.net/okbs/~haihe-riscv64-team/tensorflow-depend)中；

将上述PPA添加到系统软件源后，使用apt安装Bazel。

```
sudo add-apt-repository okbs:~haihe-riscv64-team/tensorflow-depend
sudo apt-get update
sudo apt-get install bazel-bootstrap
```

## 本地/虚拟机编译例子—TensorFlow

### ➤ 步骤3：下载TensorFlow源代码

由于当前bazel的最高版本为4.2.3，可以支持编译TensorFlow的最高版本为2.8.4，可以通过<https://github.com/tensorflow/tensorflow/releases>下载指定版本的源码。

### ➤ 步骤4：配置buid

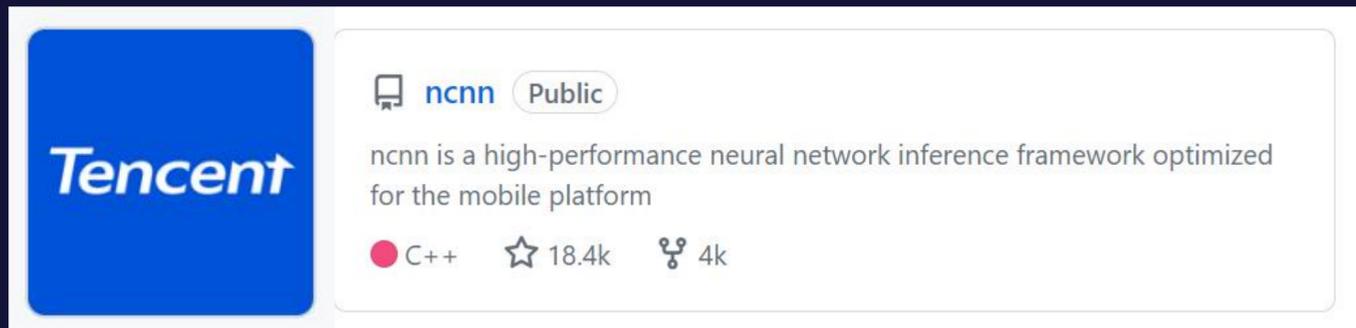
通过运行 TensorFlow 源代码树根目录下的 `./configure` 配置系统 build。此脚本会提示您指定 TensorFlow 依赖项的位置，并要求指定其他构建配置选项（例如，编译器标记，是否编译支持CUDA的版本等），具体示例可见TensorFlow官网中“[从源代码构建](#)”部分。

### ➤ 步骤5：构建pip软件包

```
bazel build -c opt //tensorflow/tools/pip_package:build_pip_package
```

# 面向端侧的AI框架

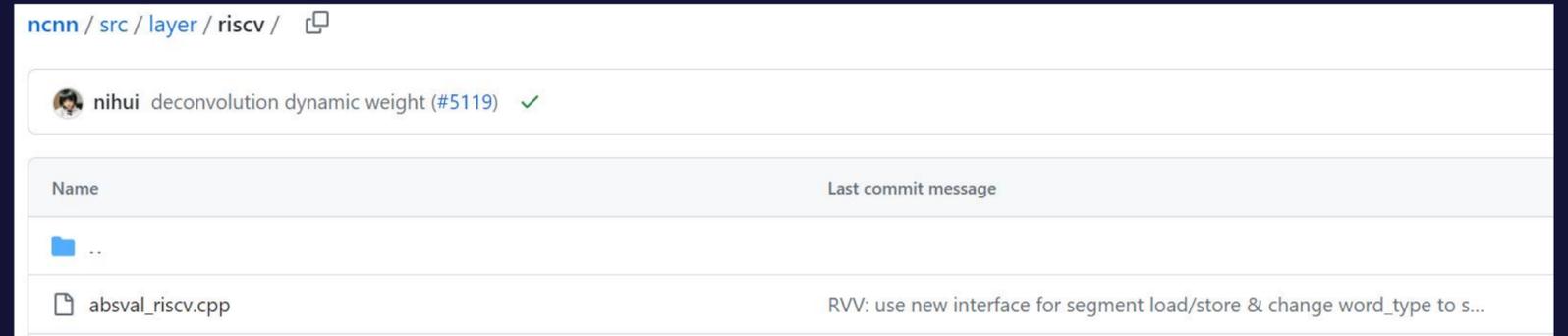
## ➤ NCNN



**Tencent** **ncnn** Public

ncnn is a high-performance neural network inference framework optimized for the mobile platform

C++ 18.4k 4k

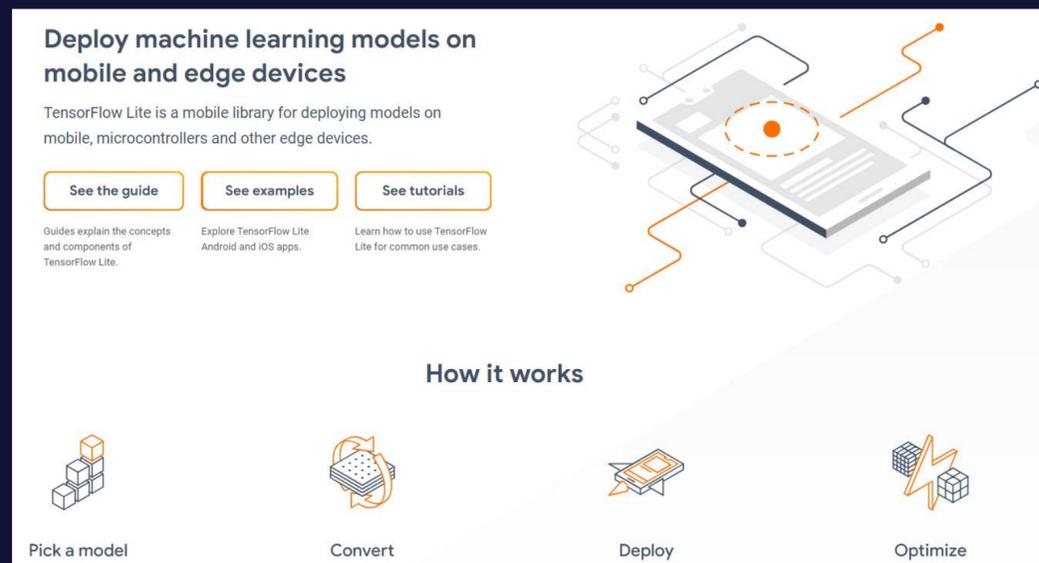


ncnn / src / layer / riscv /

nihui deconvolution dynamic weight (#5119) ✓

Name	Last commit message
..	
absval_riscv.cpp	RVV: use new interface for segment load/store & change word_type to s...

## ➤ TensorFlow Lite



**Deploy machine learning models on mobile and edge devices**

TensorFlow Lite is a mobile library for deploying models on mobile, microcontrollers and other edge devices.

[See the guide](#) [See examples](#) [See tutorials](#)

Guides explain the concepts and components of TensorFlow Lite. Explore TensorFlow Lite Android and iOS apps. Learn how to use TensorFlow Lite for common use cases.

**How it works**

- Pick a model
- Convert
- Deploy
- Optimize

### Towards Deep Learning using TensorFlow Lite on RISC-V

Marcia Sahaya Louis  
marcia93@bu.edu  
Boston University

Zahra Azad  
zazad@bu.edu  
Boston University

Leila Delshadtehrani  
delshad@bu.edu  
Boston University

Suyog Gupta  
suyoggupta@google.com  
Google Inc.

Pete Warden  
petewarden@google.com  
Google Inc.

Vijay Janapa Reddi  
vj@eecs.harvard.edu  
Harvard University

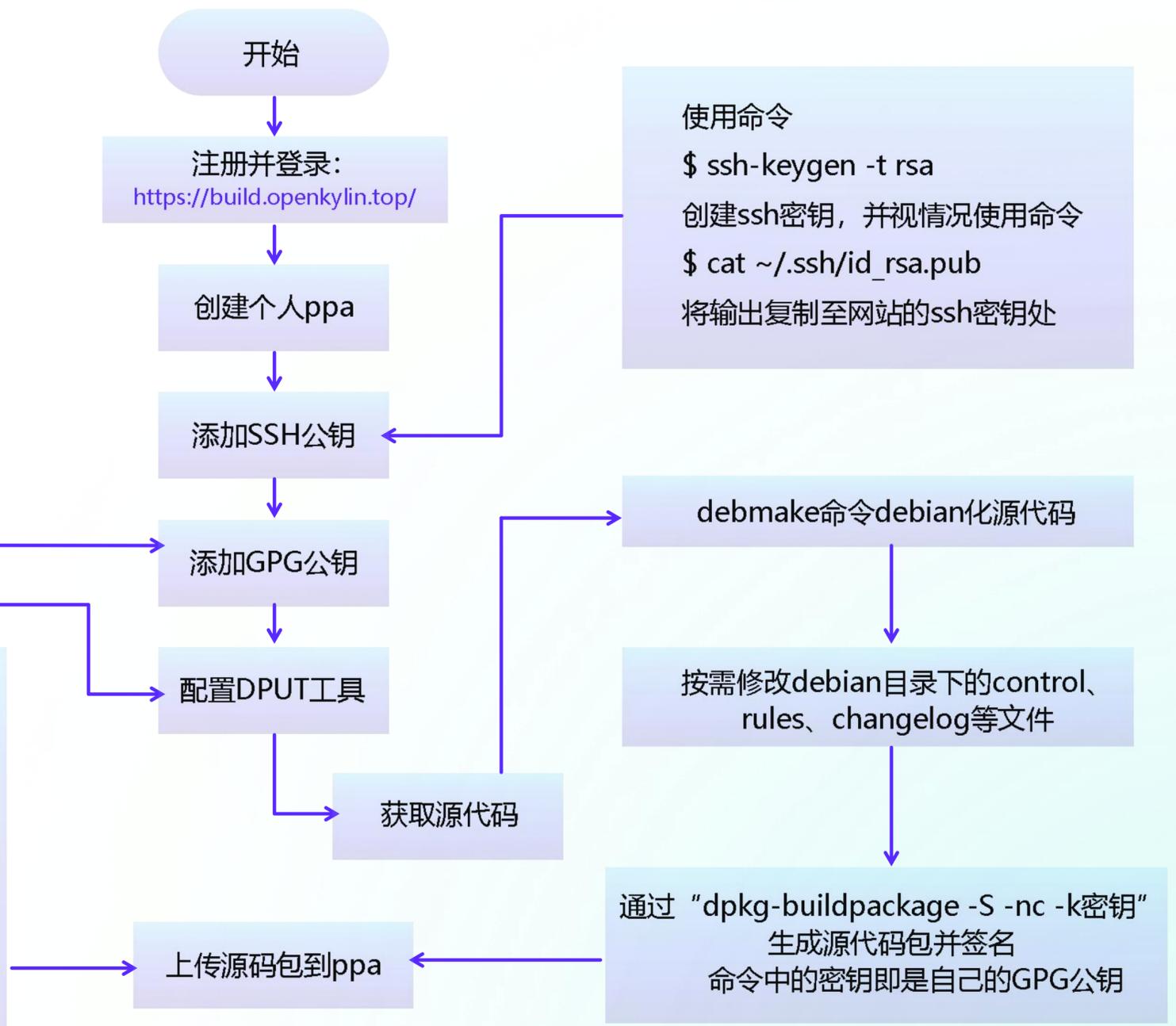
Ajay Joshi  
joshi@bu.edu  
Boston University

## okbs配置并远程编译使用流程

创建公钥命令  
\$ gpg --full-gen-key  
根据提示补充相关信息，默认将会选择RSA类型，3072位长度且永不过期  
请填写姓名和常用邮箱  
创建完成后使用命令  
\$ gpg --fingerprint  
可以查看当前系统中密钥  
将密钥上传到 hkp://keyserver.build.openkylin.top:11371 执行命令  
\$ gpg --keyserver hkp://keyserver.build.openkylin.top:11371 --send-keys <keyid 后8位即可>  
上传后将创建密钥拷贝到网站的gpg密钥处

源码包的changelog 文件中，系列代号设置为 yangtze  
在执行 dput 前请保证系统环境中安装了 paramiko  
\$ sudo pip3 install paramiko  
使用该命令安装依赖  
生成source.changes后，按照ppa页面的提示执行dput命令  
例如：  
\$ dput okbs:~xiewei/ppa <source.changes>  
dput执行成功后，需要等待服务器处理上传的文件，可在 <http://archive.build.openkylin.top/dput-logs/> 查看处理结果。

修改本地 ~/.dput.cf 或 /etc/dput.cf，添加如下内容：  
LOGIN改为自己的okbs帐号对应ID  
[okbs]  
fqdn=upload.build.openkylin.top:2121  
method=sftp  
incoming=%(okbs)s  
login=LOGIN

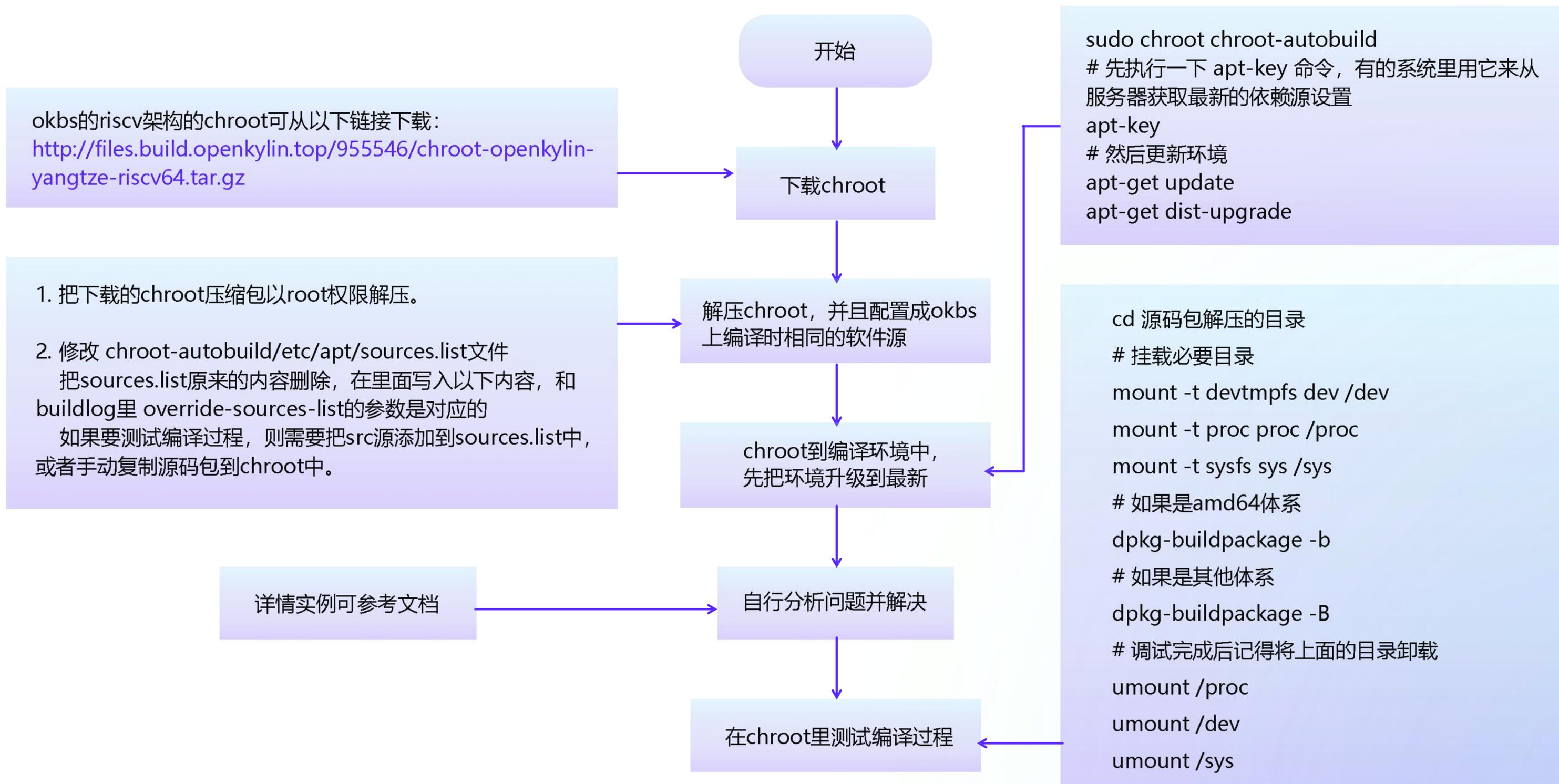


使用命令  
\$ ssh-keygen -t rsa  
创建ssh密钥，并视情况使用命令  
\$ cat ~/.ssh/id\_rsa.pub  
将输出复制至网站的ssh密钥处

debmake命令debian化源代码  
按需修改debian目录下的control、rules、changelog等文件

通过 "dpkg-buildpackage -S -nc -k密钥"  
生成源代码包并签名  
命令中的密钥即是自己的GPG公钥

## 调试okbs编译错误





## dput上传时提示 unknown method:sftp

dput源包错误，将本地安装的源包卸载，并安装如下dput包  
[https://launchpad.net/ubuntu/+source/dput/1.1.0ubuntu2.1/+build/26381494/+files/dput\\_1.1.0ubuntu2.1\\_all.deb](https://launchpad.net/ubuntu/+source/dput/1.1.0ubuntu2.1/+build/26381494/+files/dput_1.1.0ubuntu2.1_all.deb)

## dput上传时提示 Unable to negotiate with 124.126.103.198 port 2121: no matching host key type found. Their offer: ssh-rsa

在~/.ssh目录下修改config文件。（如果是root用户并且没有这个文件夹可以手动创建，亲测有效）如果没有config就创建一个config文件，增加以下内容：

```
Host *
  PubkeyAcceptedKeyTypes +ssh-rsa
  HostKeyAlgorithms +ssh-rsa
```

第一行说明对所有主机生效，第二行是将ssh-rsa加会允许使用的范围，第三行是指定所有主机使用的都是ssh-rsa算法的key

## 上传源代码时提示: gpg: ../cargo\_0.66.0+ds1-ok1\_source.changes: error 58: gpgme\_op\_verify gpgme\_op\_verify: GPGME: No data

重新签名.changes文件即可：

```
debsign -k "GPG公钥" <filename>.changes
```

## 上传到ppa时提示类似错误: Unable to find cargo\_0.66.0+ds1.orig-vendor.tar.gz in upload or distribution.Unable to find cargo\_0.66.0+ds1.orig.tar.gz in upload or distribution.

这是因为上传的source.changes文件中，没有附上对应文件的词条。在source.changes文件中加上相应文件的签名即可。签名后需要对source.changes文件进行重签名并且重新上传到ppa



01.

XLA: 在RISC-V架构, 适配使用TensorFlow的编译优化技术XLA, 改进执行速度和内存使用, 加速模型的训练和推理;

02.

定点运算优化: 在RISC-V架构中, 浮点运算比定点运算性能开销更大, 在不显著影响模型准确性的前提下, 可以通过将浮点运算转化成定点运算来提高性能;

03.

多线程和并行计算: 在RISC-V架构上实现多线程并行, 充分利用RISC-V的多核特性, 以加速任务的执行;

04.

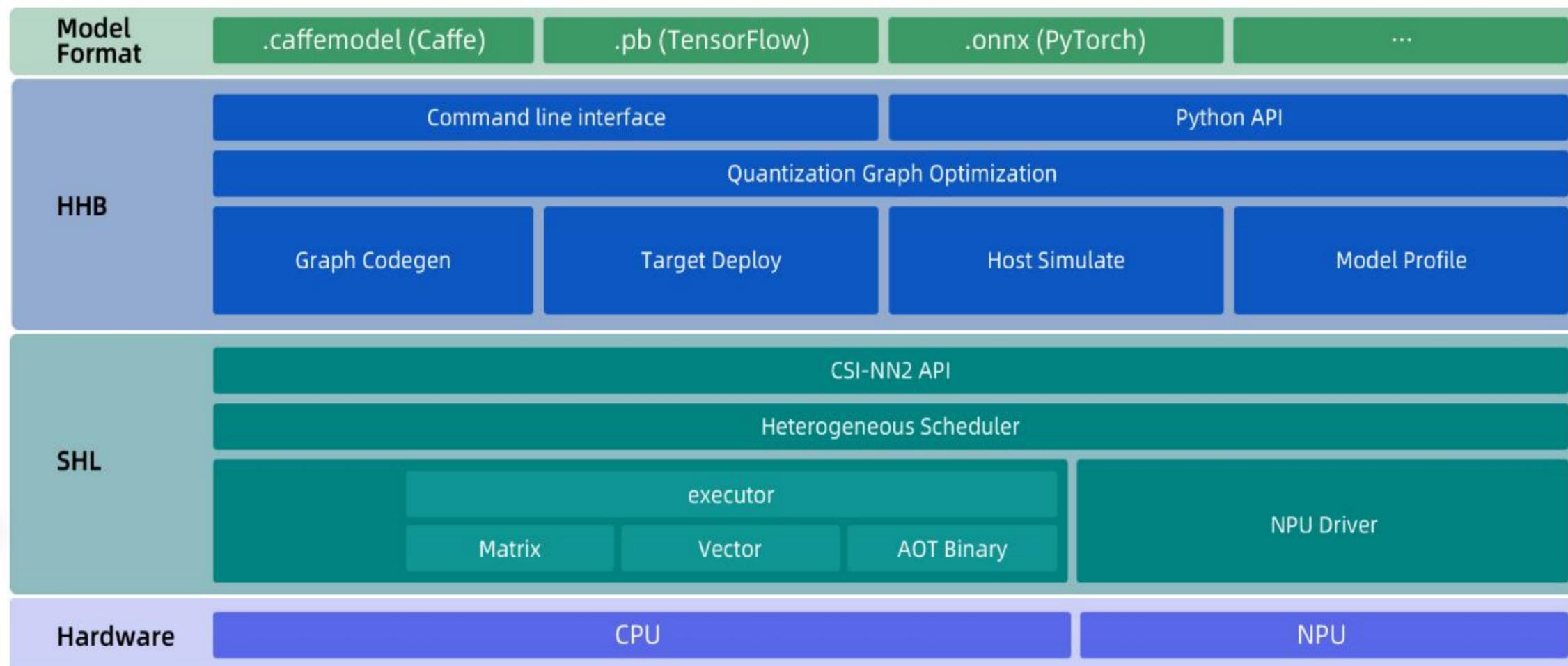
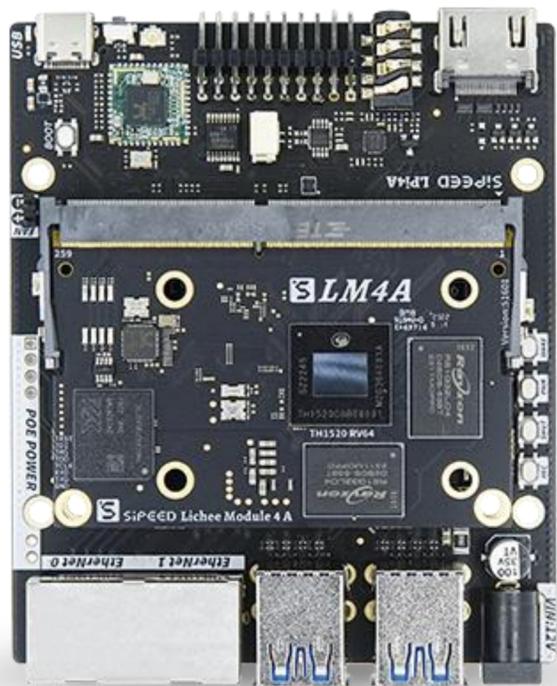
数学库和核心算法优化: 针对RISCV架构特性优化核心算法, 比如 BLAS 和 LAPACK等线性代数库等。

05.

内存管理和优化: 优化数据存储和内存访问模式, 减少内存占用、提高缓存效率; 降低内存碎片化, 优化内存分配和回收。



HHB (Heterogeneous Honey Badger) 是一套针对**玄铁**芯片设计平台的神经网络模型部署工具集。





```
开始连接到 sophgo@172.28.8.243 0.9
Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Mon Jul 25 12:12:50 UTC 2022

System load:  0.08          Processes:      107
Usage of /:   58.0% of 31.37GB  Users logged in:  0
Memory usage: 2%           IP address for ens3: 172.28.8.243
Swap usage:  0%

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

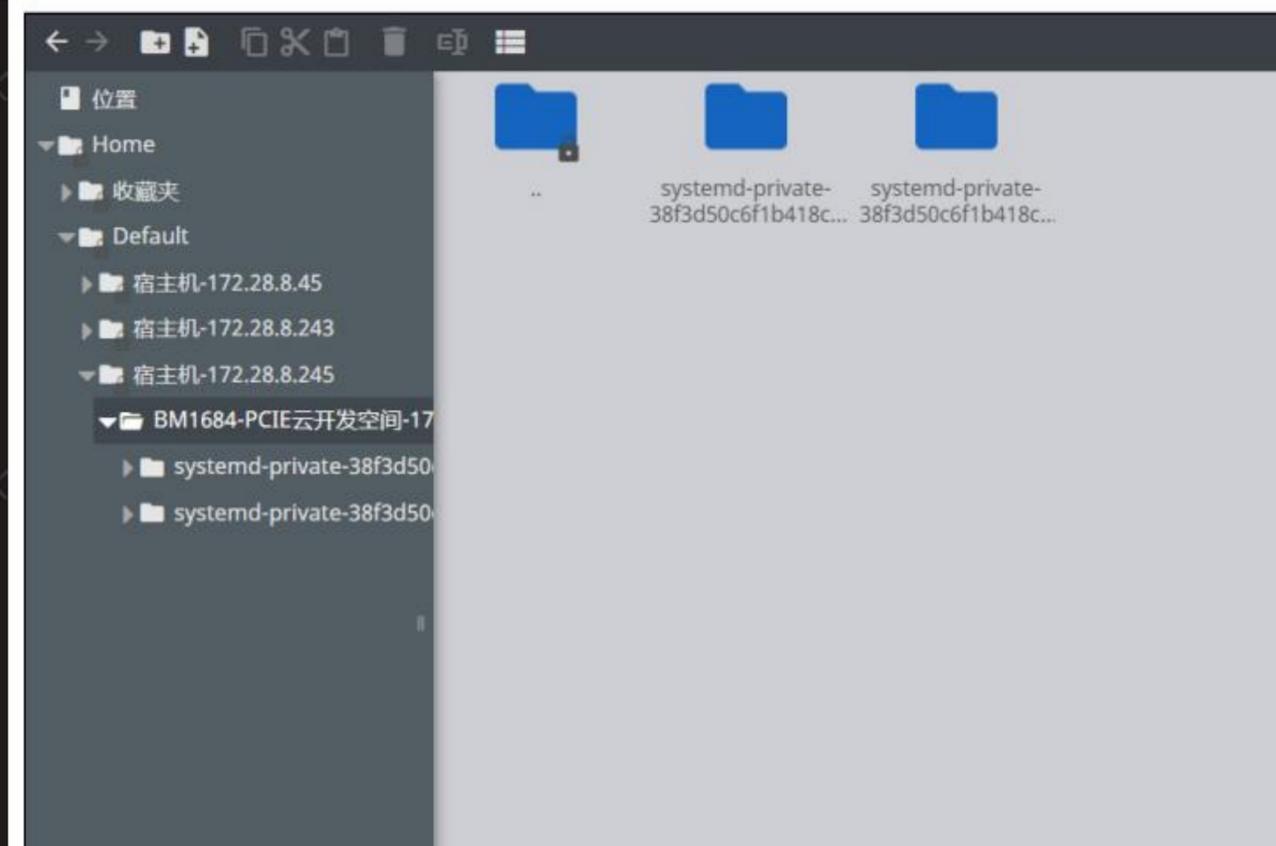
https://ubuntu.com/blog/microk8s-memory-optimisation

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
   https://ubuntu.com/livepatch

125 packages can be updated.
1 update is a security update.

New release '20.04.4 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** System restart required ***
Last login: Mon Jul 25 09:51:43 2022 from 172.28.141.100
sophgo@sophgo:~$
```



今日直播



## 开放原子开源大赛

# 面向RISC-V架构的AI开发框架 构建与优化大赛辅导培训

### 《面向RISC-V架构的AI开发 框架构建大赛介绍》

王文竹  
openKylin社区技术委员会委员  
信创海河实验室副研究员



### 《玄铁多媒体AI全栈软硬件平台》

贺晏安  
阿里巴巴达摩院 RISC-V及生态 技术专家



### 《算能RISC-V通用云开发空间 使用教程》

毛慧华  
北京算能科技有限公司 应用软件工程师



直播开始时间:

1月9日 19:00

腾讯会议: 638160352



扫码参会

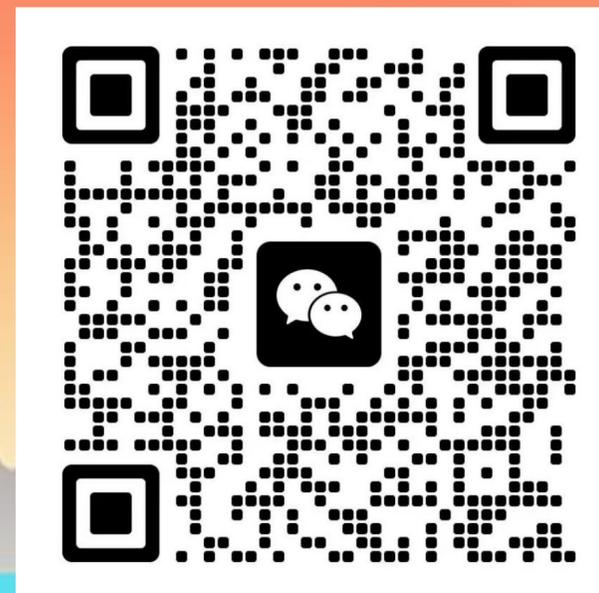
开源聚力，共创未来  
Open source cohesion ,co-creation future



# 我要报名



扫一扫 进入活动页面报名



添加好友，并发送“开放原子  
开源大赛”入群交流

