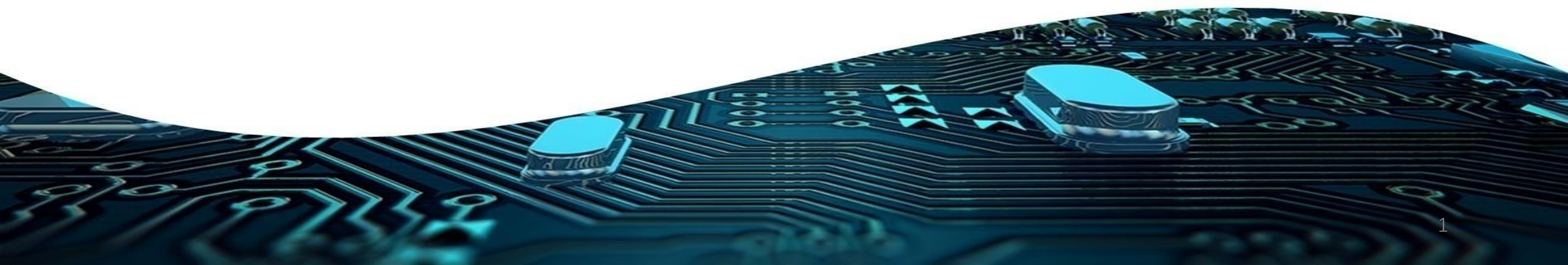




开源RISC-V芯片设计与实现

中科院上海微系统所
2018-12-6 郑云龙



2018年热点之一：RISC-V

RISC-V在全球范围内的热度

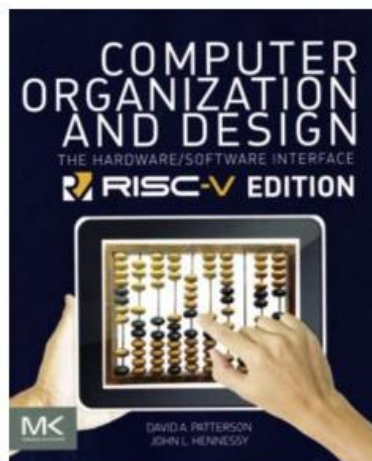
● 计算机体系结构泰斗Hennessy和Patterson获得2017年图灵奖，且双双进驻Google

- [《RISC-V与DSA！计算机架构宗师Patterson与Hennessy 演讲实录》](#)
- [《Hennessy与Patterson双双进驻Google，是计算机科学新时代的曙光》](#)
- [《2017图灵奖得主揭晓：体系结构大师获奖 谷歌成赢家》](#)
- 经典英文教材《计算机组成与设计——David A.Patterson / John L.Hennessy》改为RISC-V版本
- 经典英文教材《计算机体系结构量化研究方法——David A.Patterson / John L.Hennessy》改为RISC-V版本

● 被全球范围内的大学陆续采纳为教材

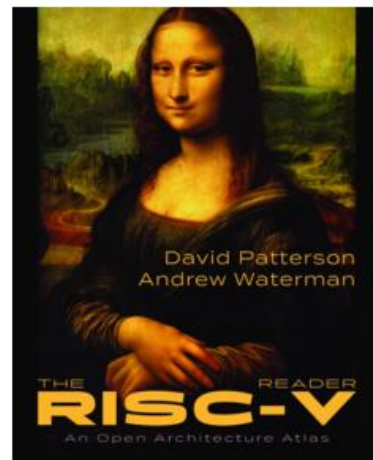
- 世界各地的很多一流大学都已经开始基于RISC-V开展体系结构研究。粗略统计，加州大学伯克利分校、MIT、普林斯顿大学、康奈尔大学、华盛顿大学、英国剑桥大学、瑞士苏黎世联邦理工学院、德州大学奥斯汀分校等一流大学在过去7届RISC-V研讨会上已累计发表了70余项前沿研究工作
- 全美国大学相关教程均逐步改为RISC-V授课，中国很多大学正在开始采用RISC-V授课，若干年后每一个电子工程系和计算机系毕业生都会熟练RISC-V

"Patterson and Hennessy brilliantly address the issues in the ever changing computer hardware architecture."
Professor Jae Oh, Syracuse



AUTHORS David Patterson & John Hennessy
AVAILABILITY Now

"I like RISC-V and this book as they are elegant—brief, to the point, and complete."
C. Gordon Bell, Designer of the PDP-11 and VAX-11 instruction set architectures



AUTHORS David Patterson & Andrew

"Computer Architecture: A Quantitative Approach is a classic that, like fine wine, just keeps getting better."
James Hamilton, Amazon Web Services



AUTHORS John Hennessy & David Patterson
AVAILABILITY December 15,

2018年热点之一：RISC-V

RISC-V在全球范围内的热度

- 被多个国家采纳为国家标准指令集

- [包云岗：关于RISC-V成为印度国家指令集的一些看法](#)
- [以色列关于RISC-V的Genpro计划](#)
- 美国国防部国防高等研究计划(DARPA)采用RISC-V

- 日益引起了产业界的广泛关注

- [《ARM太贵，80多家科技巨头悄然站队开源芯片架构RISC-V》](#)
- [《三星准备丢开ARM，开发RISC-V架构自主CPU内核》](#)
- [《NVIDIA Is Building Its Next-Gen Falcon Controller Using RISC-V》](#)
- [《Western Digital Gives A Billion Unit Boost To Open Source RISC-V CPU》](#)

- 硅谷RISC-V知名新创公司相继获得巨额融资

- [《美国SiFive完成5060万美元C轮融资》](#)
- [《Esperanto exits stealth mode, aims at AI with a 4096-core 7nm RISC-V》](#)



2018年热点之一：RISC-V

RISC-V Foundation Growth History
August 2015 (7) to April 2018 (150+)



2018年热点之一：RISC-V



2018年热点之一：RISC-V



**中国开放指令生态 (RISC-V) 联盟在乌镇世界
互联网大会上宣布成立**

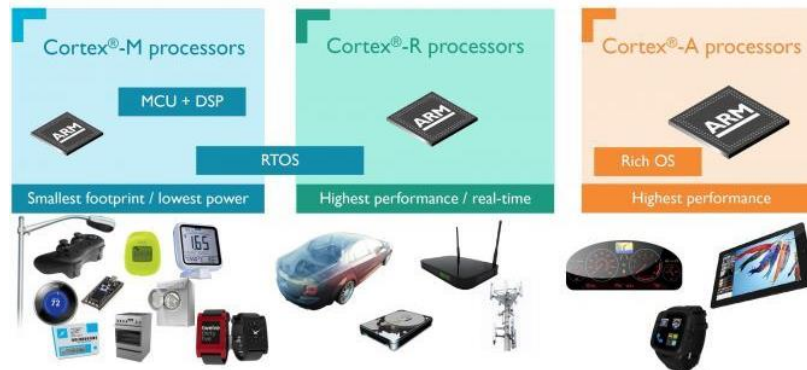
Agenda

- RISC-V的诞生
- RISC-V指令集简介
- RISC-V开源生态发展趋势
- RISC-V行业应用普及现状
- RISC-V普及真正推动力

Agenda

- **RISC-V的诞生**
- RISC-V简介
- RISC-V开源生态发展趋势
- RISC-V行业应用普及现状
- RISC-V普及真正推动力

RISC-V诞生



处理器技术现状背景

存在问题 在2010年，已存在ARM, x86, MIPS, SPARC, Power, 5种处理器技术

1. 复杂且不开源，处理器内部结构不清楚，可能存在后门漏洞，无法安全可控 (x86)
2. 存在专利壁垒，昂贵的知识产权授权费用 (ARM)
3. 属于商业公司产品，若公司被收购或者面临危机，则整个技术体系存在长期支持风险 (SPARC)
4. 商业公司以盈利为目的，不利于技术标准推广普及，由于容易与上下游公司产生利益冲突，缺乏完善的产业链生态支持 (MIPS、Power)

| 领域 | 主流架构 |
|---------------------------|------------------------------------|
| • 服务器 (Server) 领域 | • 以Intel公司x86架构的高性能CPU占垄断地位 |
| • 桌面个人计算机 (PC) 领域 | • 以Intel或者AMD公司x86架构的CPU占垄断地位 |
| • 嵌入式移动手持设备 (Mobile) 领域 | • 以ARM Cortex-A架构占垄断地位 |
| • 嵌入式实时设备 (Real Time) 领域 | • ARM架构占最大份额，其他RISC架构的嵌入式CPU也有不错表现 |
| • 深嵌入式 (Deep Embedded) 领域 | • ARM架构占最大份额，其他RISC架构的嵌入式CPU也有不错表现 |

RISC-V诞生

发源于基金会成立

● 2010年发源于伯克利大学Krste教授项目

- 由于受够了现在处理器架构的复杂性和相关知识产权的限制
- 由计算机体系结构领域的泰斗David Patterson的大力支持和推动
- 伯克利大学决定发明一种全新的指令集架构，从RISC-I开始设计，不断发展至RISC-V
- RISC-V (英文读作“risk-five”)，“V”包含两层意思：
 - 一是这是Berkeley从RISC I开始设计的第五代指令集架构
 - 二是它代表了变化(Variation)和向量(Vectors)

● 具体解决方案:

- 完全开源
- 免费授权，不存在专利壁垒
- 成立永久的非营利性基金会管理运营
- 组织建立完整的上下游产业开发联盟
- 每年召开两次workshop国际技术讨论大会

● RISC-V 基金会2016年正式成立运行

- 创始会员包括Google、西部数据等大公司
- 目前中国的会员有：中科院计算所、台湾晶心 (Andes)、华为、台湾联发科、C-Sky (杭州中天微)、中兴微电子等

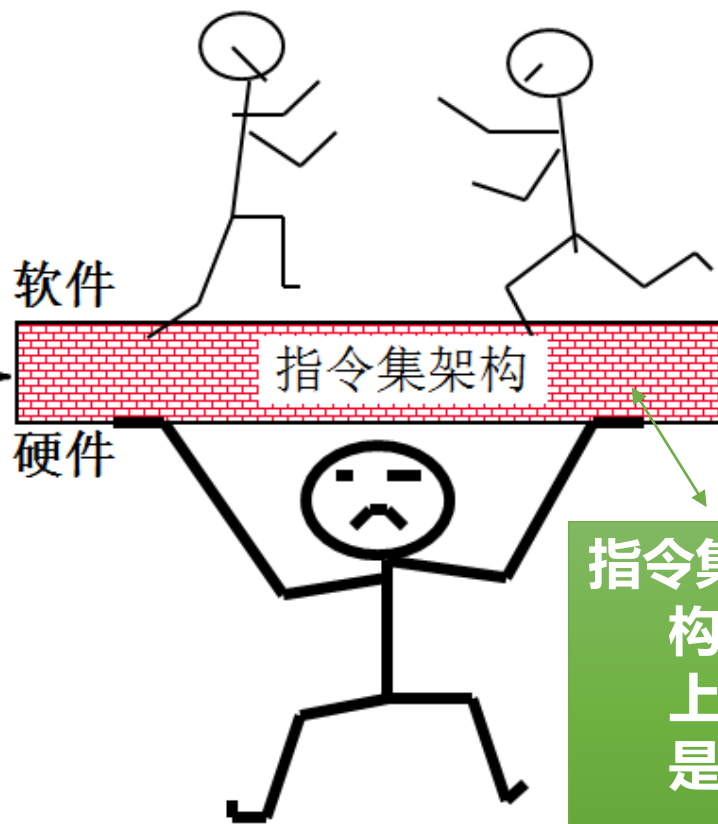


RISC-V诞生

处理器灵魂——指令集架构 (ISA)

- ◆数据类型
- ◆存储模型
- ◆软件可见的处理器状态
 - 通用寄存器 (General registers)
 - PC (Program Counter)
 - 处理器状态 (Processor status)
- ◆指令集
 - 指令类型与编码 (Instructions and formats)
 - 寻址模式 (Addressing modes)
 - 数据结构 (Data structures)
- ◆系统模型
 - 状态 (States)
 - 特权级别 (Privilege Level)
 - 中断和异常 (Interrupts and Exceptions)
- ◆外部接口
 - 输入输出接口 (IO)
 - 管理 (Management)

指令集架构，有时简称为“架构”或者称之为“处理器架构”，是处理器的灵魂



指令集架构不仅仅是一组指令的集合，它还要定义任何软件程序员需要了解的硬件信息，包括支持的数据类型、存储器 (Memory) 和寄存器状态、寻址模式与存储器模型等等

指令集架构可以理解为一个抽象层，构成处理器底层硬件与运行于其上的软件之间的桥梁与接口，也是现在计算机处理器中最重要的一个抽象层

RISC-V诞生

处理器架构领域的革命

- RISC-V 通用,简单, 开放和免费 , 使用 BSD License 开源协议!!!



The Linley
Group
Announces
Winners of
Annual Analysts'
Choice Awards



- RISC-V架构的目标是：
 - 成为一种完全开放的指令集架构，可以被任何学术机构或商业组织所自由使用
 - 成为一种真正适合硬件实现且稳定的标准指令集架构

Agenda

- RISC-V的诞生
- **RISC-V指令集简介**
 - 模块化
 - 极简
 - 可扩展
- RISC-V开源生态发展趋势
- RISC-V行业应用普及现状
- RISC-V普及真正推动力

RISC-V指令集简介

模块化

● RISC-V的模块化指令集组织方式

| 基本指令集 | 指令数 | 描述 |
|----------|-----|-------------------------------------------------------|
| • RV32I | 47 | • 32位地址空间与整数指令，支持32个通用整数寄存器 |
| • RV32E | 47 | • RV32I的子集，仅支持16个通用整数寄存器 |
| • RV64I | 59 | • 64位地址空间与整数指令，及一部分32位的整数指令 |
| • RV128I | 71 | • 128位地址空间与整数指令，及一部分64位和32位的指令 |
| 扩展指令集 | 指令数 | 描述 |
| • M | 8 | • 整数乘法与除法指令 |
| • A | 11 | • 存储器原子（Atomic）操作指令和Load-Reserved/Store-Conditional指令 |
| • F | 26 | • 单精度（32比特）浮点指令 |
| • D | 26 | • 双精度（64比特）浮点指令，必须支持F扩展指令 |
| • C | 46 | • 压缩指令，指令长度为16位 |

RISC-V指令集简介

极简

● RISC-V的设计原则——大道至简

| 特性 | x86或ARM架构 | RISC-V |
|------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 架构篇幅 | • 数千页 | • 少于三百页 |
| 指令数目 | • 指令数繁多，不同的架构分支彼此不兼容 | • 一套指令集支持所有架构。基本指令子集仅40余条指令，以此为共有基础，加上其他常用模块子集指令总指令数也仅几十条 |
| 易实现性 | • 硬件实现的复杂度高 | • 硬件设计与编译器实现非常简单： <ul style="list-style-type: none">➢ 仅支持小端格式➢ 存储器访问指令一次只访问一个元素➢ 去除存储器访问指令的地址自增自减模式➢ 规整的指令编码格式➢ 简化的分支跳转指令与静态预测机制➢ 不使用分支延迟槽（Delay Slot）➢ 不使用指令条件码（Conditional Code）➢ 运算指令的结果不产生异常（Exception）➢ 16位的压缩指令有其对应的普通32位指令➢ 不使用零开销硬件循环 |

RISC-V指令集简介

可扩展

• 可扩展

- 预留的指令编码空间
- 预定义的Custom指令

| inst[15:13] | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 | |
|-------------|----------|-----------------------|------|-----------------------|-----------------|----------------------|------|-----------------------|-----------------------|
| inst[1:0] | | | | | | | | | |
| 00 | ADDI4SPN | FLD FLD LQ | LW | FLW LD LD | <i>Reserved</i> | FSD FSD SQ | SW | FSW SD SD | RV32 RV64 RV128 |
| 01 | ADDI | JAL ADDIW ADDIW | LI | LUI/ADDI16SP | MISC-ALU | J | BEQZ | BNEZ | RV32 RV64 RV128 |
| 10 | SLLI | FLDSP FLDSP LQ | LWSP | FLWSP LDSP LDSP | J[AL]R/MV/ADD | FSDSP FSDSP SQ | SWSP | FSWSP SDSP SDSP | RV32 RV64 RV128 |
| 11 | >16b | | | | | | | | |

| inst[4:2] | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|-----------|--------|----------|-----------------|----------|--------|-----------------|-----------------------|---------|
| inst[6:5] | | | | | | | | (> 32b) |
| 00 | LOAD | LOAD-FP | <i>custom-0</i> | MISC-MEM | OP-IMM | AUIPC | OP-IMM-32 | 48b |
| 01 | STORE | STORE-FP | <i>custom-1</i> | AMO | OP | LUI | OP-32 | 64b |
| 10 | MADD | MSUB | NMSUB | NMADD | OP-FP | <i>reserved</i> | <i>custom-2/rv128</i> | 48b |
| 11 | BRANCH | JALR | <i>reserved</i> | JAL | SYSTEM | <i>reserved</i> | <i>custom-3/rv128</i> | ≥ 80b |

RISC-V指令集优点

包括以下几个方面:

1. RISC-V属于免费开源架构, 无须付费
2. RISC-V的ISA比起其他ISA, 指令简单, 因此大大降低芯片设计验证工作。
3. RISC-V属于非营利基金会, 不用担心ISA突然发生很大变化或者消失, 可持使用
4. RISC-V在不同行业应用中, 芯片设计实现更高效, 面积、功耗和性能更有优势
5. RISC-V可作为SoC芯片核的基础ISA, 具有很好的扩展性, 可以随意按照需求扩展

Agenda

- RISC-V的诞生
- RISC-V指令集简介
- **RISC-V开源生态发展趋势**
- RISC-V行业应用普及现状
- RISC-V普及真正推动力

RISC-V开源和商用IP内核爆发式发展

RISC-V全球开源和商用IP内核爆发式增长

| 名称 | 属性 | 类型 | 开发者 | 简介 |
|------------|------|------------|-----------------|---------------------------------------------------------|
| • Rocket | • 开源 | • 内核 | • 伯克利大学 | • 5级流水按序单发射, 使用Chisel语言开发, 由Rocket-Chip SoC Generator生成 |
| • BOOM | • 开源 | • 内核 | • 伯克利大学 | • 超标量乱序双发射, 使用Chisel语言开发, 由Rocket-Chip SoC Generator生成 |
| • Freedom | • 开源 | • SoC | • SiFive公司 | • 使用Rocket内核 |
| • LowRISC | • 开源 | • SoC | • LowRISC组织 | • 使用Rocket内核 |
| • PULPino | • 开源 | • Core和SoC | • PULP组织 | • 有三种不同型号的内核, 涵盖不同等级 |
| • PicoRV32 | • 开源 | • Core | • Clifford Wolf | • 专为FPGA优化的超小面积内核 |
| • SCR1 | • 开源 | • Core | • Syntacore公司 | • 最低功耗版本开源, Syntacore公司还提供其他商用版本 |
| • ORCA | • 开源 | • Core | • VectorBlox公司 | • 使用VHDL开发 |
| • Andes | • 商用 | • Core | • Andes公司 | • 台湾Andes公司的下一代架构采用RISC-V架构 |
| • Cudasip | • 商用 | • Core | • Cudasip公司 | • 提供各种型号IP |
| • SiFive | • 商用 | • Core | • SiFive公司 | • 提供各种型号IP |
| • 蜂鸟E203 | • 开源 | • Core | • Bob Hu | • 蜂鸟E203开源 |

RISC-V发展现状

发展现状

● 被全球范围内的大学陆续采纳为教材

- 世界各地的很多一流大学都已经开始基于RISC-V开展体系结构研究。粗略统计，加州大学伯克利分校、麻省理工学院、普林斯顿大学、康奈尔大学、华盛顿大学、英国剑桥大学、瑞士苏黎世联邦理工学院、德州大学奥斯汀分校等一流大学在过去7届RISC-V研讨会上已累计发表了70余项前沿研究工作
- 经典英文教材《计算机组成与设计——David A.Patterson / John L.Hennessy》改为RISC-V版本
- 经典英文教材《计算机体系结构量化研究方法——David A.Patterson / John L.Hennessy》改为RISC-V版本
- 全美国大学相关教程均逐步改为RISC-V授课，若干年后每一个电子工程系和计算机系毕业生都会熟稔RISC-V

● 被多个国家采纳为国家标准指令集

- [包云岗：关于RISC-V成为印度国家指令集的一些看法](#)
- [以色列关于RISC-V的Genpro计划](#)
- 美国国防部国防高等研究计划(DARPA)采用RISC-V
- 中国中标基金/中国RISCV产业联盟/中国RISCV技术联盟

● 日益引起了产业界的广泛关注

- ARM在移动和嵌入式领域完胜，其他商用架构（如MIPS）逐渐消亡；ARM正在激进布局进军Intel所在的x86市场，已经对传统PC和服务器领域造成巨大冲击。
- 而巨头大公司也很看重CPU ISA（指令集架构）的开放性，各大巨头公司正在积极寻找ARM之外的第二选择，**RISC-V成为时代之必然产物。**
- 亚洲第一的微处理器IP厂商台湾晶心科技（Andes）下一代架构采用RISC-V
- [《三星准备丢开ARM，开发RISC-V架构自主CPU内核》](#)
- [《NVIDIA Is Building Its Next-Gen Falcon Controller Using RISC-V》](#)
- [《Western Digital Gives A Billion Unit Boost To Open Source RISC-V CPU》](#)
- [《Hennessy与Patterson双双进驻Google，是计算机科学新时代的曙光》](#)
- [《华米科技-全球智能可穿戴第一颗AI芯片，智能手环用上了RISC-V架构》](#)

RISC-V国内发展现状

国内发展现状

● 国内社区活跃

CNRV微信群四群爆满，CNRV社区定期发布双周报



CNRV
为推广RISC-V尽些薄力

CNRV 社区群 <https://cnrv.io/> [View On GitHub](#)

关于

本站点希望能够为国内的RISC-V开发者和爱好者提供便利。

RISC-V双周简报

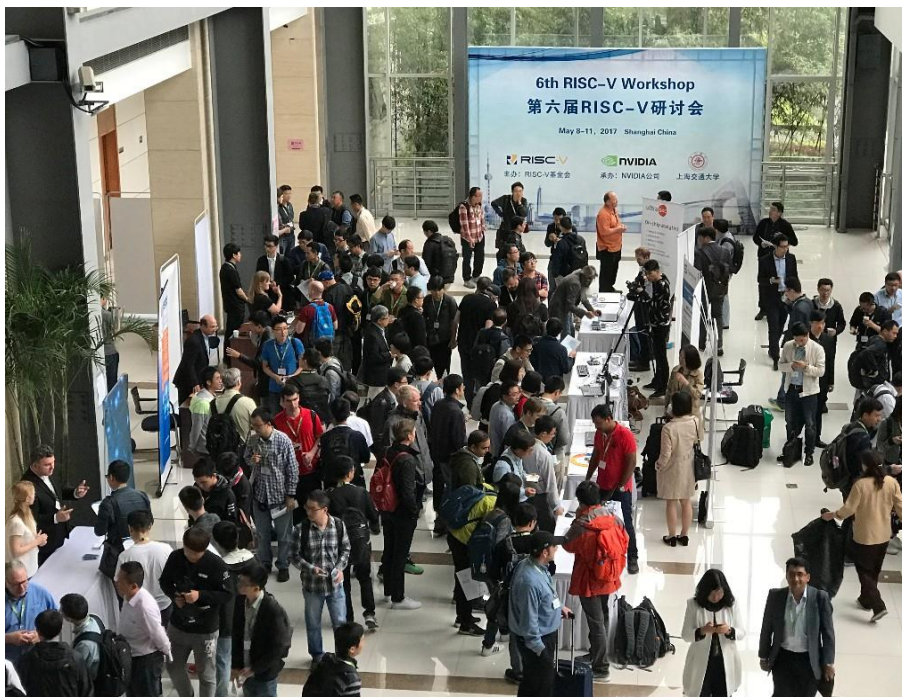
国内的RISC-V爱好者利用GitHub协作的方式，以双周简报的方式为大家带来最新的RISC-V相关资讯，同时在微信公众号和CNRV站点上发布。内容覆盖RISC-V邮件列表、行业新闻、项目进展以及各类点评。也欢迎大家关注CNRV公众号获取最新信息。

- 第0x10弹(2018-02-01): 走在“时尚的前沿” [繁体]
- 第0x0f弹(2018-01-18): Google开源RISC-V CPU [繁体]
- 第0x0e弹(2018-01-04): Intel漏洞是非多! [繁体]
- 第0x0d弹(2017-12-21): Esperanto Technologies会成功么? [繁体]
- 第0x0c弹(2017-12-07): Workshop上干活多多 [繁体]
- 第0x0b弹(2017-11-23): 3家RISC-V相关的公司上榜“EE Times Silicon 60: Startups to Watch” [繁体]
- 第0x0a弹(2017-11-09): RISC-V的好汉们就快凑齐一桌麻将了 [繁体]
- 第0x0a弹(2017-10-26): 国内商业公司玩RISC-V毫不逊色!
- 第0x09弹(2017-10-12): 妥妥的跑Linux的RISC-V板子明年就来~
- 第0x08弹(2017-09-28): 悠悠的唱着最炫开源风~

• 往期存档

● 国内Workshop活动

2017年5月于上海交大举行的Workshop



2018年6月于上海复旦举办的RISC-V上海



RISC-V Day in Shanghai

June 30, 2018



● 频频见诸报道

《半导体行业观察》，《EETOP》等

国内第一款开源RISC-V 处理器核 蜂鸟E203

芯来科技

https://github.com/SI-RISCV/e200_opensource

Hummingbird E200 Opensource Processor Core

About

This repository hosts the project for open-source hummingbird E200 RISC processor Core.

The Hummingbird E200 core is a two-stages pipeline based ultra-low power/area implementation, which has both performance and areas benchmark better than ARM Cortex-M0+ core, makes the Hummingbird E200 as a perfect replacement for legacy 8051 core or ARM Cortex-M cores in the IoT or other ultra-low power applications.

To boost the RISC-V popularity and to speed up the IoT development in China, we are very proud to make it open-source. It is the first open-source processor core from China mainland with industry level quality and state-of-art CPU design skills to support RISC-V instruction set.

Our ambition is to make "Hummingbird E200" become next 8051 in China, please go with us to make it happen.

Usages and Applications

The open-source Hummingbird E200 core can be a perfect candidate for the following fields:

- Replace legacy 8051 core for better performance.
- Replace Cortex-M core for lower cost.
- Also, the Hummingbird E200 core as a simple ultra-low power core and SoC, which is "蜂鸟虽小、五脏俱全", with detailed Docs and Software/FPGA Demos, hence, it will be a perfect example for lab practice in university or entry-level studying.

Detailed Introduction

We have provided very detailed introduction and quick start-up documents to help you ramping it up.

The detailed introduction and the quick start documentation can be seen from `e200_opensource/doc` directory.

By following the guidences from the doc, you can very easily start to use Hummingbird E200 processor core and demo SoC.

SI-RISCV / e200_opensource

Watch 54 Unstar 195 Fork 71

Code Issues 3 Pull requests 1 Projects 0 Wiki Insights

The Ultra-Low Power RISC Core

risc-v ultra-low-power cpu core china verilog

87 commits 1 branch 0 releases 2 contributors Apache-2.0

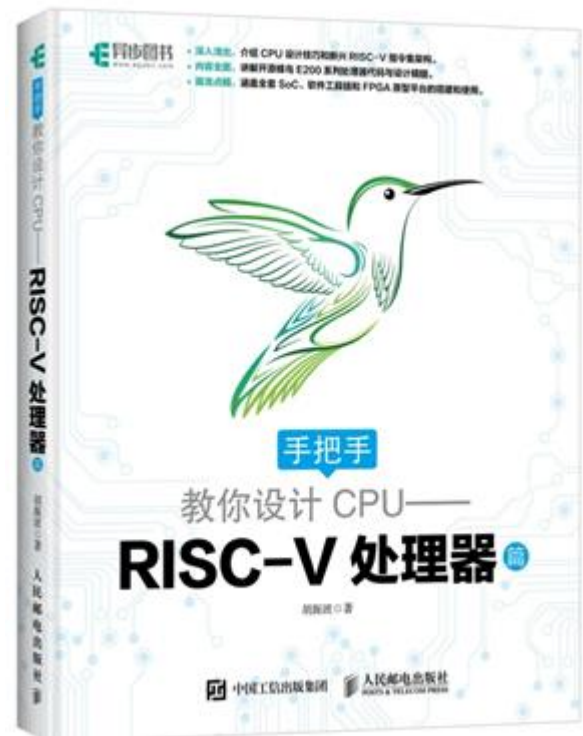
Branch: master New pull request Create new file Upload files Find file Clone or download

| SI-RISCV Update README.md | Latest commit 92d9c4 on 19 May |
|---------------------------|--------------------------------------------------------------------------------------|
| boards | add a placeholder boards directory 5 months ago |
| doc | Update README.md a month ago |
| fpga | update the prebuilt mcs to easy user download it directly 4 months ago |
| prebuilt_tools | add the prebuilt tools link to easy user 5 months ago |
| riscv-tools | add the missing tests 5 months ago |
| rtl/e203 | the original freedom-e310 chisel generated QSPI have a bug in Quad-mo... a month ago |
| sirv-e-sdk | add the missing header file 5 months ago |
| tb | enlarge the interrupt generate period in tb 5 months ago |
| vsim | update makefile 5 months ago |
| .gitignore | update ignore list 5 months ago |
| LICENSE | upload e203 RTL codes with copyright notice and LICENSE 9 months ago |
| README.md | Update README.md a month ago |

国内第一本RISC-V中文教材

国内第一本RISC-V书籍胡振波

可在京东淘宝购买



第一部分 CPU 与 RISC-V 综述

| | |
|--------------------------------------------|----|
| 第 1 章 一文读懂 CPU 之三生三世 | 2 |
| 第 2 章 大道至简——RISC-V 架构之魂 | 29 |
| 第 3 章 乱花渐欲迷人眼——盘点 RISC-V 商业版本与开源版本 | 46 |
| 第 4 章 开源 RISC-V——蜂鸟 E200 系列超低功耗 Core & SoC | 54 |

第三部分 使用 Verilog 进行仿真和在 FPGA SoC 原型上运行软件

| | |
|---------------------------------|-----|
| 第 17 章 冒个烟先——运行 Verilog 仿真测试 | 292 |
| 第 18 章 套上壳子上路——实现 SoC 和 FPGA 原型 | 302 |
| 第 19 章 画龙点睛——运行和调试软件示例 | 321 |
| 第 20 章 是骡子是马？拉出来遛遛——运行跑分程序 | 332 |

附录部分 RISC-V 架构详述

| | |
|--------------------------|-----|
| 附录 A RISC-V 架构指令集介绍 | 342 |
| 附录 B RISC-V 架构 CSR 寄存器介绍 | 374 |
| 附录 C RISC-V 架构的 PLIC 介绍 | 384 |
| 附录 D 存储器模型背景介绍 | 392 |
| 附录 E 存储器原子操作指令背景介绍 | 397 |
| 附录 F RISC-V 指令编码列表 | 400 |
| 附录 G RISC-V 伪指令列表 | 404 |

第二部分 手把手教你使用 Verilog 设计 CPU

| | |
|------------------------------------|-----|
| 第 5 章 先见森林,后观树木——蜂鸟 E200 设计总览和顶层介绍 | 65 |
| 第 6 章 流水线不是流水账——蜂鸟 E200 流水线介绍 | 78 |
| 第 7 章 万事开头难吗——一切从取指令开始 | 88 |
| 第 8 章 一鼓作气,执行力是关键——执行 | 117 |
| 第 9 章 善始者实繁,克终者盖寡——交付 | 161 |
| 第 10 章 让子弹飞一会儿——写回 | 170 |
| 第 11 章 哈弗还是比亚迪——存储器架构 | 178 |
| 第 12 章 黑盒子的窗口——总线接口单元 BIU | 203 |
| 第 13 章 不得不说的故事——中断和异常 | 217 |
| 第 14 章 最不起眼的,其实是最难的——调试机制 | 246 |
| 第 15 章 动如脱兔,静若处子——低功耗的诀窍 | 260 |
| 第 16 章 工欲善其事,必先利其器——RISC-V 可扩展协处理器 | 276 |

更多的RISC-V技术介绍

(微信公众号: 硅农亚历山大)



实例讲解进驻Google两位大神主推的异构计算与RISC-V
John Hennessy和David Patterson两位计算机体系结构的泰山北斗双双进驻Google



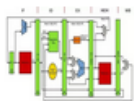
大道至简——RISC-V架构之魂(上)
本文为《RISC-V CPU设计》专栏和《RISC-V嵌入式软件开发》专栏系列文章之一



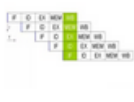
大道至简——RISC-V架构之魂(中)
本文为《RISC-V CPU设计》专栏和《RISC-V嵌入式软件开发》专栏系列文章之一



大道至简——RISC-V架构之魂(下)
本文为《RISC-V CPU设计》专栏和《RISC-V嵌入式软件开发》专栏系列文章之一



八一八处理器流水线(上)?
本文将讨论处理器的一个重要的基础知识:“流水线”。本文将简要介绍处理器的一些



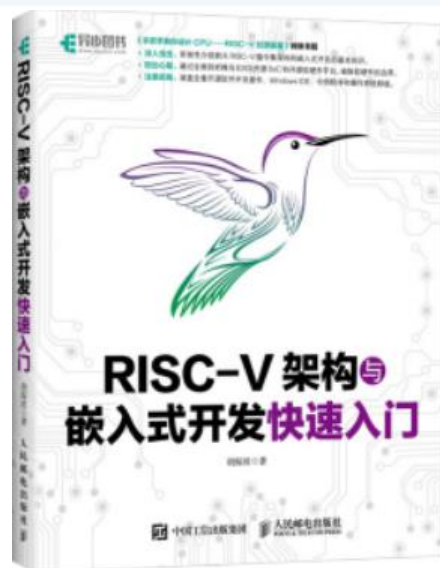
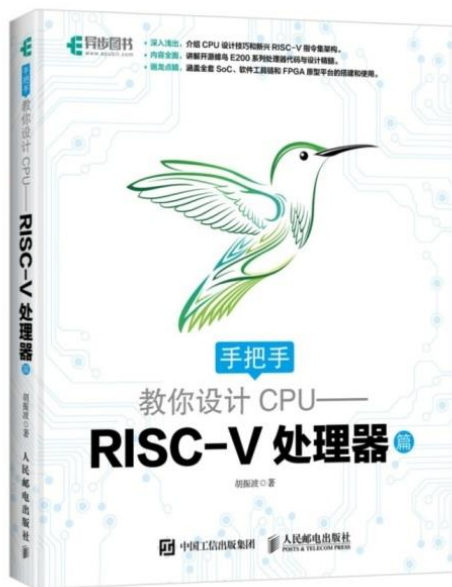
八一八处理器流水线(下)?
本文将讨论处理器的一个重要的基础知识:“流水线”。本文将简要介绍处理器的一些



动如脱兔,静若处子——处理器低功耗设计的诀窍
低功耗机制对于处理器而言至关重要。本章将对处理器的低功耗技术加以概述。

市,越来越多的爱好者开始使用开源的蜂鸟E203 RISC-V处理核,很多初学者留言询问有关RISC-V工具链使用的问题,因此本公众号将开始陆续发表若干篇有关RISC-V软件工具链使用的文章,包括:

- RISC-V嵌入式开发准备篇1: 编译过程简介
- RISC-V嵌入式开发准备篇2: 嵌入式开发的特点介绍
- RISC-V嵌入式开发入门篇1: RISC-V GCC工具链的介绍
- RISC-V嵌入式开发入门篇2: RISC-V汇编语言程序设计
- RISC-V嵌入式开发上手篇: 基于HBird-E-SDK平台的软件开发与运行
- RISC-V嵌入式开发实践篇: 运行开源蜂鸟E200 MCU更多示例程序
- RISC-V嵌入式开发新奇篇: 基于Windows Eclipse IDE的软件开发与运行
- RISC-V嵌入式开发升华篇: 基于开源蜂鸟E200 MCU移植RTOS



RISC-V的发展趋势

开源技术现状

目前关于操作系统、软件、网络协议等技术领域都有相应的开源计划，成熟的社区组织领导了技术创新，并促使其成功发展

| 领域 | 标准 | 开源, 免费 |
|-----------------|----------|-----------------|
| 操作系统 OS | Posix | Linux , FreeBSD |
| 编译器 Compiler | C | gcc, LLVM |
| 数据库 Database | SQL | MySQL |
| 网络协议 Network | TCP/IP | 众多 |
| 处理器 CPU | RISC-V ? | RISC-V ? |

RISC-V的发展趋势

开源硬件发展趋势

开源技术正在从软件、操作系统的开源成功——>走向硬件开源

| 领域 | 主流, 商业 | 开源, 免费 | 下一代 |
|----------------|------------------|---------------------------------|------------------|
| 操作系统 OS | Microsoft-Window | Linux , FreeBSD , Android | Zephyr ACRN |
| 硬件 Hardware | Intel-PC | Arduino/Raspber ry | RISC-V? ? |

- 降低成本, 提高可靠性, 缩短开发周期
- 开源的Linux平台是新一代手机操作系统Android迅速发展的基础保障
- 开源的**RISC-V** 平台是下一代碎片化应用**AIOT** (智能物联网) 的基础?

Agenda

- RISC-V的诞生
- RISC-V简介
- RISC-V开源生态发展趋势
- **RISC-V行业应用普及现状**
- RISC-V普及真正推动力

RISC-V行业应用

全球范围内使用RISC-V芯片的商业大公司

| 公司 | 使用 |
|----------|---------------------------|
| • 西部数据 | • 公开宣传将替换其所有产品的处理器为RISC-V |
| • NVIDIA | • GPU内部Falcon控制内核 |
| • 三星 | • 芯片内微控制器内核 |
| • Ceva | • IP内微控制器内核 |
| • Google | • 芯片内微控制器内核 |
| • Micron | • 存储器芯片内控制器核 |

● 注意:

- 以上表格仅列出明确发布了新闻稿的知名大公司
- 行业内已经大范围内使用RISC-V充当嵌入式控制内核

RISC-V行业应用

| 地区 | 公司 | 商业模式 |
|------------------|-----------------------------------------------------------------|---------------------------------------------------------------------|
| • US | • SiFive • Microsemi • BlueSpec • Cortus • Espranto | • IP, 开发板, 芯片 • FPGA (嵌入RISC-V) • IP • IP、解决方案 • 人工智能芯片 |
| • EU | • Codaip • UltraSoC • GreenWaves | • IP • IP • IoT边缘计算芯片 |
| • Russia | • Syntacore | • IP |
| • Taiwan | • Andes | • IP |
| • China Mainland | • 中天微 (平头哥半导体公司) | • IP/芯片 |

● 注意:

- 以上表格仅列出比较知名的RISC-V商业公司, 更多RISC-V公司可于网络上查阅
- 以上表格未包含开放组织, 譬如PULP、LowRISC等

RISC-V潜在的行业应用方向

RISC-V与X86和ARM的互补共赢关系

- 在特定专用领域（SSD控制器、FPGA软核）、新兴的领域（譬如IoT、AI、区块链、安全加密、边缘计算、数据中心等）是RISC-V应用发展方向

新兴领域的应用

新兴领域的应用

- 代表芯片一：华米手环智能穿戴AI处理器



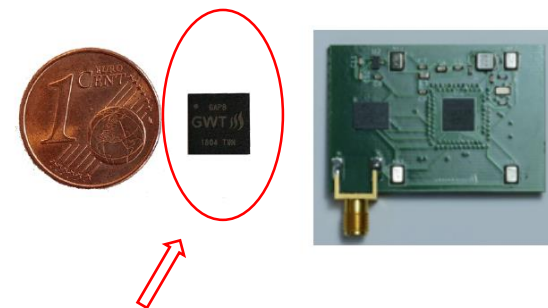
- 代表芯片二：GreenWaves IoT应用处理器

GAP8 delivers this computing energy efficiency and cost breakthrough

- **Ultra low power**
20x better than the state-of-the-art on content understanding applications.
- **Low cost system solution**
2-3x less than alternatives.
- **Flexible**
Fully programmable.
Not just CNN.
- **Low installation cost**
Years on a battery
10-100x less than wired installations

GAP8

The IoT Application Processor



FPGA领域的应用

集成处理器CPU核心的FPGA

FPGA硬核



FPGA软核

MicroBlaze

NIOS

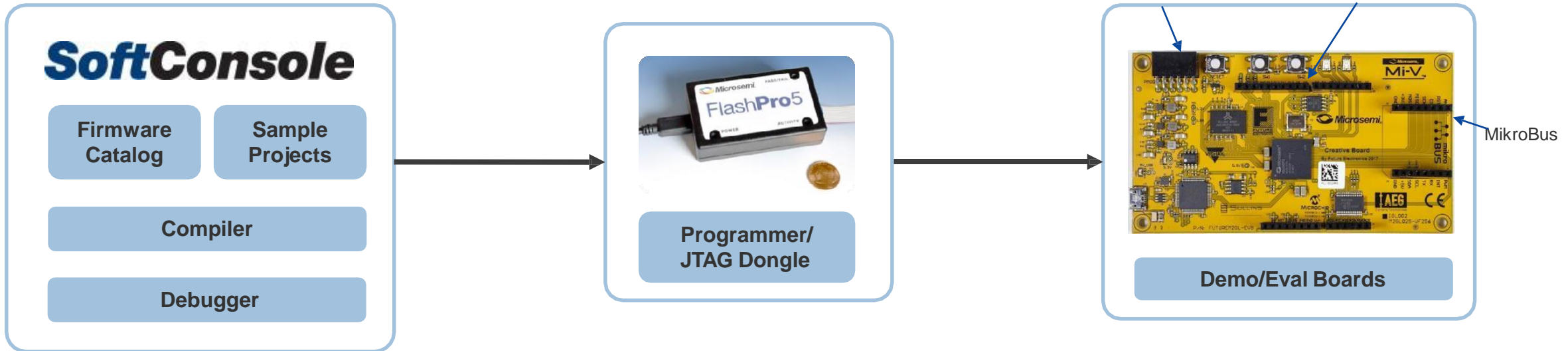
FPGA中需要软核完成控制——FPGA缺失共用的CPU软核

CPU软核的需求——开源、易修改



Microsemi RISC-V软核CPU

Microsemi: Mi-V Eclipse IDE Design Flow



- A single tool chain for RISC-V and ARM MCUs
 - Easy migration from ARM to RISC-V
- Running on Linux or Windows Hosts
- Bundled with example projects and RTOSs
- <https://github.com/RISCV-on-Microsemi-FPGA>



RISC-V的行业应用现状

- 如果只是在特定领域和新兴领域
- RISC-V相较于已有处理器指令集并没有显著优势
- Intel&ARM会在既有垄断领域继续保持强势
- RISC-V并不能成为主流，迅猛发展

Agenda

- RISC-V的诞生
- RISC-V简介
- RISC-V开源生态发展趋势
- RISC-V行业应用普及现状
- **RISC-V普及真正推动力**

RISC-V的出路和真正的优势

在哪里？

自主可控与软硬件协同设计

RISC-V诞生的时代背景

- **互联网公司**之前一直在搭建生态抢夺流量入口，让用户可以更方便地使用自己的服务，巨头们都想到了搭建自己的硬件平台作为自己服务的入口，互联网巨头追求硬件性能极致化以实现差异化的用户体验用来吸引客户，**需要定制有差异化的芯片**
- **在摩尔定律逐渐失效的今天**，靠工艺制程提升来增进芯片性能的成本实在太高，因此需要更多根据应用做专用设计，靠异构计算架构设计来进一步提升性能，**软硬件协同设计**
- **面临着X86不授权，ARM 有条件 and 期限的架构授权（64位CPU IP千万级费用），系统公司** Google、WestDigital**需要芯片自主设计与可控**
- 例如Intel的CPU，Qualcomm的Snapdragon等等，但是这样的通用平台在今天已经无法满足移动设备对于性能和能效比的需求，尤其是在AR/VR、终端人工智能等新兴应用，而**Apple的A系列处理器芯片、iOS、APPstore组合推出，促使了iphone巨大成功**

RISC-V可能成为真正能够实现国产自主的指令集架构

国产处理器现状

- ARM、MIPS、x86、Power PC等任何一种国外商用处理器架构皆不能真正实现国产自主可控
- 而国内各种自定义的指令集架构不够普世通用无法形成其生态
- CPU “国产自主可控” 与 “普世通用生态” 存在天然的矛盾——**只有RISC-V能够完美解决之**

| 指令集架构 | 公司 |
|-----------------|----------------------------------------------|
| • ARM v7/v8-A | • 飞腾——国防科大 • 展讯 • 华为 • 华芯通——贵州与高通合资 |
| • Intel/AMD x86 | • 北大众志 • 兆芯——威盛内资 • 海光——天津与AMD合资 |
| • MIPS | • 龙芯 • 君正 |
| • IBM Power PC | • 中晟宏芯 |
| • DEC Alpha | • 申威 |
| • C-Sky | • C-Sky |

RISC-V自主可控的意义

RISC-V带来的自主可控新机遇

倪光南院士提出了CPU自主可控“核心三要素”，是否是真正自主可控主要聚焦三点：

- CPU研制单位是否符合安全保密要求
- CPU指令系统是否可持续自主发展
- CPU核心源代码是否是自己编写

◆ RISC-V通过开源的方式显著降低中国创新的代价，促进自主可控芯片的开发

● RISC-V的优点：

- 全世界通用的标准处理器架构，生态优势长期向好
- 架构开放免费，消除了处理器架构的专利壁垒，降低了创新的门槛
- 具备低成本的优势，降低了创新的代价
- 可定制可扩展，能够快速响应“碎片化和越来越以应用为中心”的新需求

● 基于上述原因：

- 越来越多国外研究机构正在使用RISC-V设计处理器内核试图进入国防各个领域。譬如DARPA（美国）、IIT（印度）。
- 越来越多国内商业公司与研究机构开始使用RISC-V内核用于芯片产品。

互联网产业的巨大成功得益于拥抱开源软件

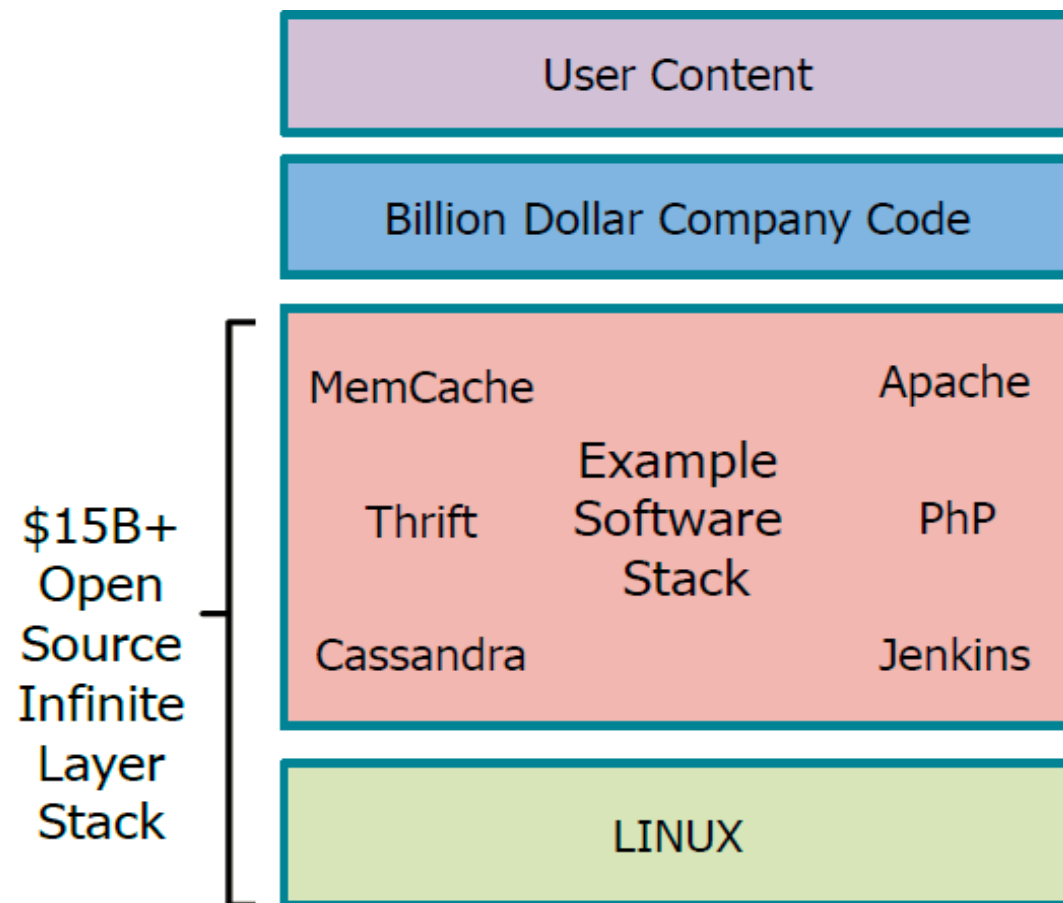
价值150亿美元的开源软件生态带来两大价值：

- ★ 降低互联网创新的门槛

3-5位开发人员用几个月即可快速开发创新业务，滴滴、摩拜等

- ★ 推动互联网企业快速成长

Google 8年10 billion营收



当前硬件的开发周期——以年为单位的设计迭代



Boards

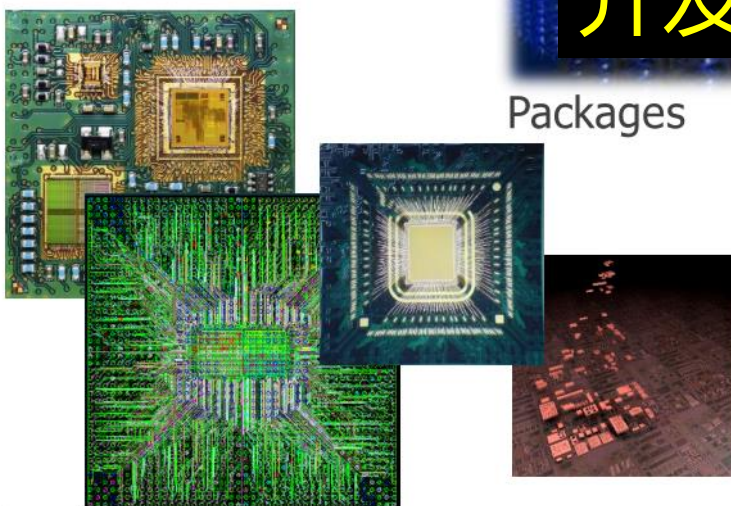
Sources: NVIDIA, Analog Devices, Raspberry Pi



Boxes

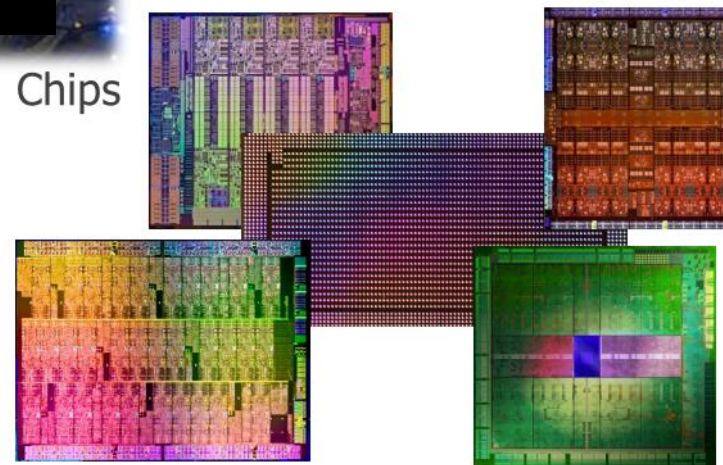
Sources: Adafruit, Adapteva, Bezar, Curtiss-Wright, Dawn VME

Hardware 开发流程



Packages

Sources: Axis, Adapteva



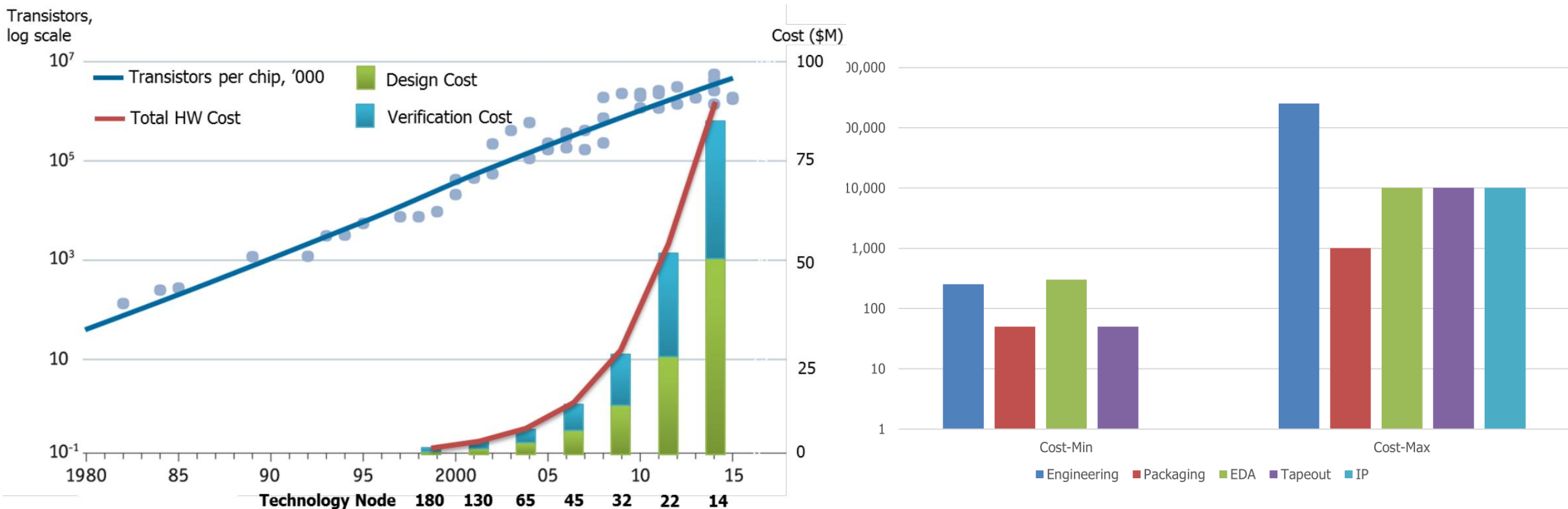
Chips

Sources: Intel, Oracle, NVIDIA

随摩尔定律发展芯片开发成本急剧上升，不可持续

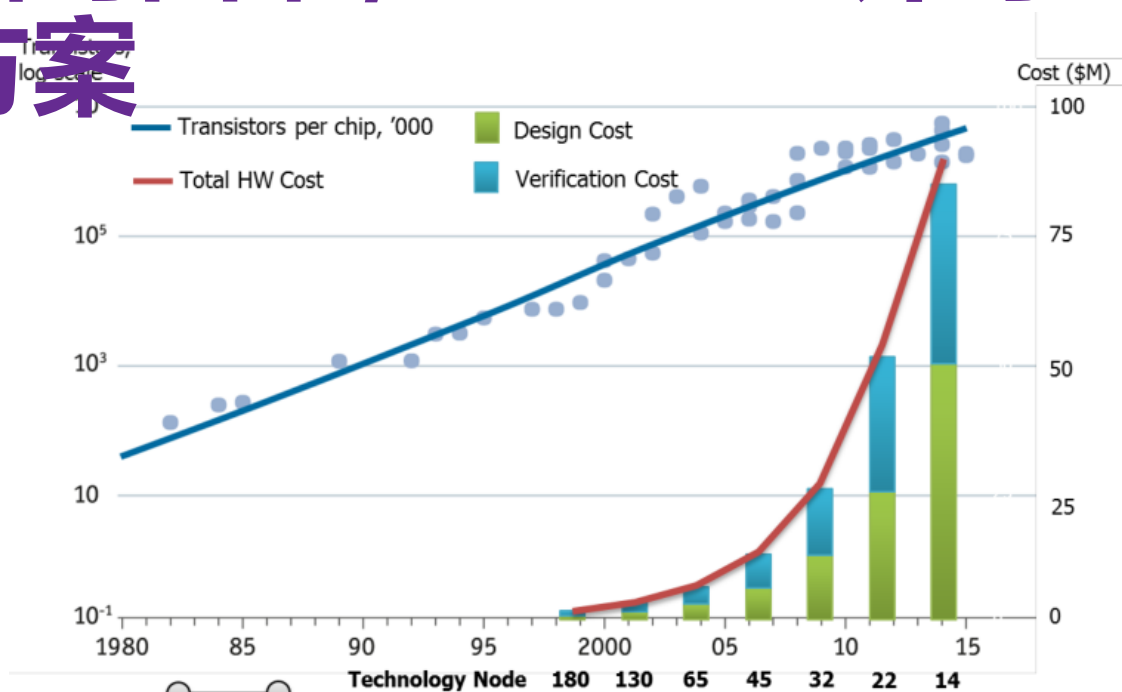
芯片设计费用越来越高

上亿元的NRE研发经费，数千名员工的协同开发，摩尔定律即将走到了尽头，但硬件性能提升的需求并未改变，产品的迭代周期反而越来越快



Source: Andreas Olofsson, Intelligent Design of Electronic Assets (IDEA), 2017

面对困难，DARPA针对芯片设计提出的解决方案



MOSIS



Fabless companies

New procedures for physical design and verification will lower the design barrier, enabling rapid specialization

Intelligent Design of Electronic Assets (IDEA)

- No human in the loop" 24-hour layout generation for mixed signal integrated circuits, systems-in-package, and printed circuit boards. Machine generated layout of electrical circuits and systems

Push Open Source Hardware (POSH)

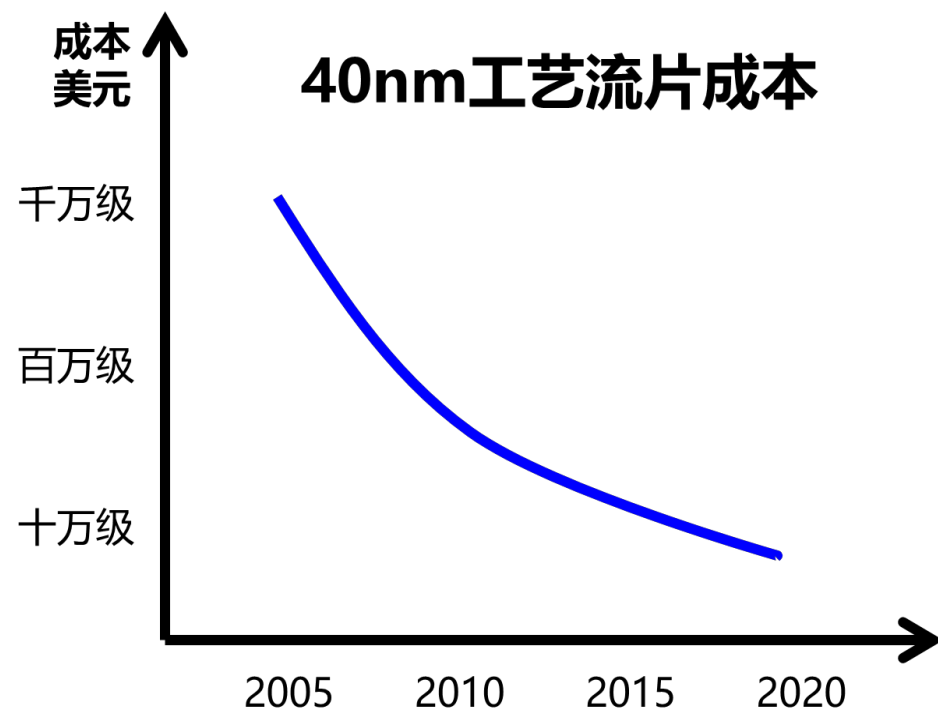
- An open source System on Chip (SoC) design and verification eco-system that enables cost effective design of ultra-complex SoCs.

- 异构计算
- 软硬件协同
- 更高层次抽象
- 标准化接口
- 自动综合工具
- 改善流程
- 方便编程语言

The 1980 's DARPA MOSIS effort removed fab cost and fab access barriers and launched the fabless industry. The ERI Design effort will address today' s design complexity and cost barriers, creating the environment needed for the next wave of US semiconductor innovation.

芯片开源：借鉴开源软件思路

- ✓ 降低芯片设计的人力、EDA、IP成本
- ✓ 成熟工艺制造成本会因摩尔定律放缓而自然下降
- ✓ 降低芯片设计成本门槛成为可能



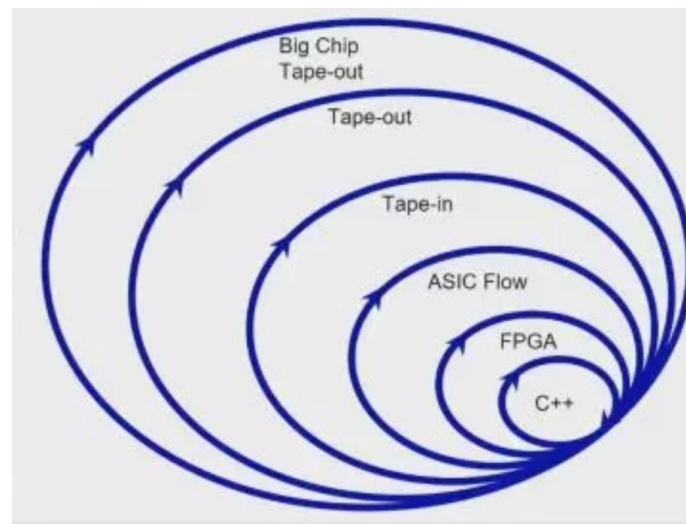
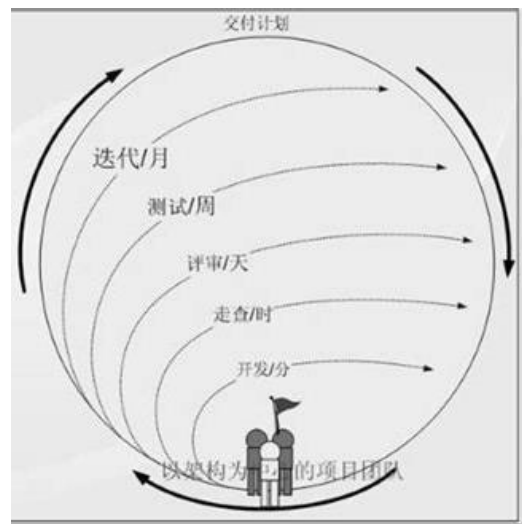
如果芯片设计门槛降低几个数量级

- ✓ **目标**: 3-5人的小团队, 在3-4个月内, 使用几万到几十万人民币, 研制出一款有市场竞争力的领域专用芯片
- ✓ 颠覆芯片开发模式, 催生软硬件协同设计新模式, 繁荣芯片创新生态

| Challenge | Industry | Hurdle | Cost | Future |
|----------------------|----------|------------|----------|----------|
| IP | \$5B | Risk | \$1M+ | \$0? |
| EDA | \$10B | Complexity | \$1M+ | \$0? |
| Engineering | \$20B? | Time | 9 months | 24hrs? |
| Packaging | \$13B | Logistics | \$50K | \$0? |
| Manufacturing | \$40B | Logistics | \$2M+ | \$1,000? |

下一个时代：软硬件协同的敏捷开发新模式

- ✓ 传统开发模式：软件以月为迭代周期，硬件以年为迭代周期
- ✓ 颠覆性新模式：软硬件协同开发，均以月为迭代周期



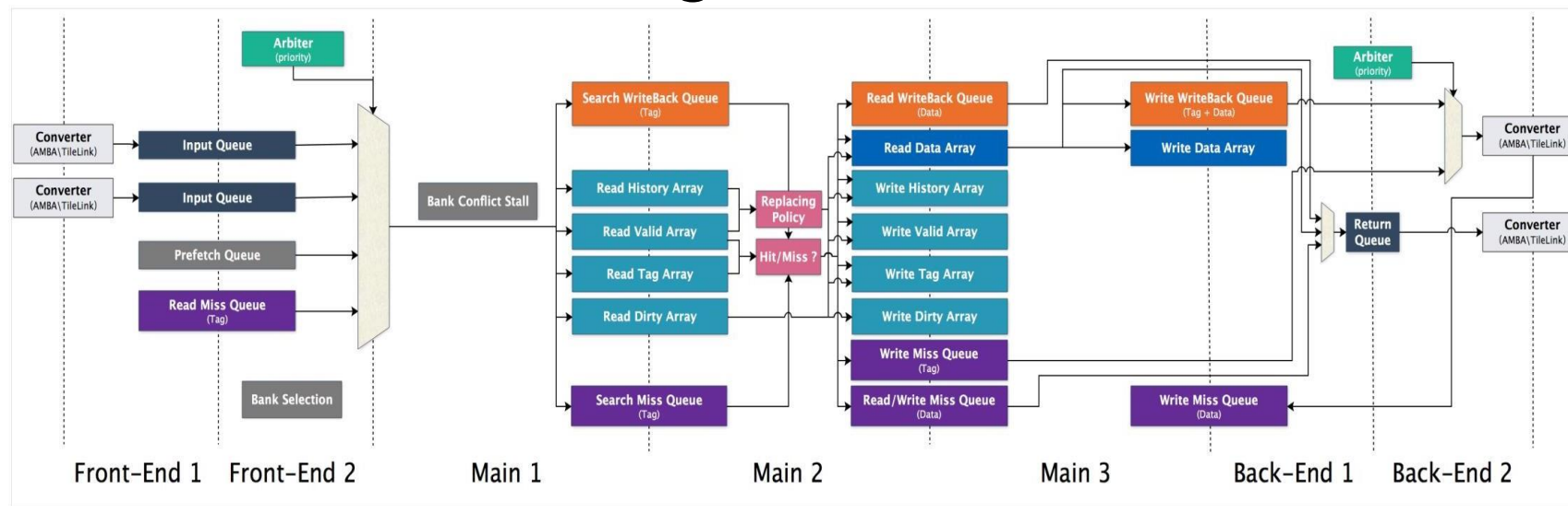
A New Golden Age
for Computer
Architecture,
Hennessy and
Patterson, 2017
ACM A.M. Turing
Award Lecture

- 解决物联网碎片化（昆虫纲悖论）的新思路——不是提供某种具体的通用芯片方案，而是提供**针对特定需求的快速软硬件协同开发能力**

RISC-V 芯片敏捷开发——中科院包云岗教授案例

同一个任务：设计CPU L2 Cache

- 一位资深工程师
- 熟悉 OpenSparc, Xilinx Cache
- 使用Verilog
- 花了数周时间完成实现存在bug
- 一位大四本科生
- 做过CPU课程设计
- 使用Chisel
- 花了3天时间完成实现, 可启动Linux



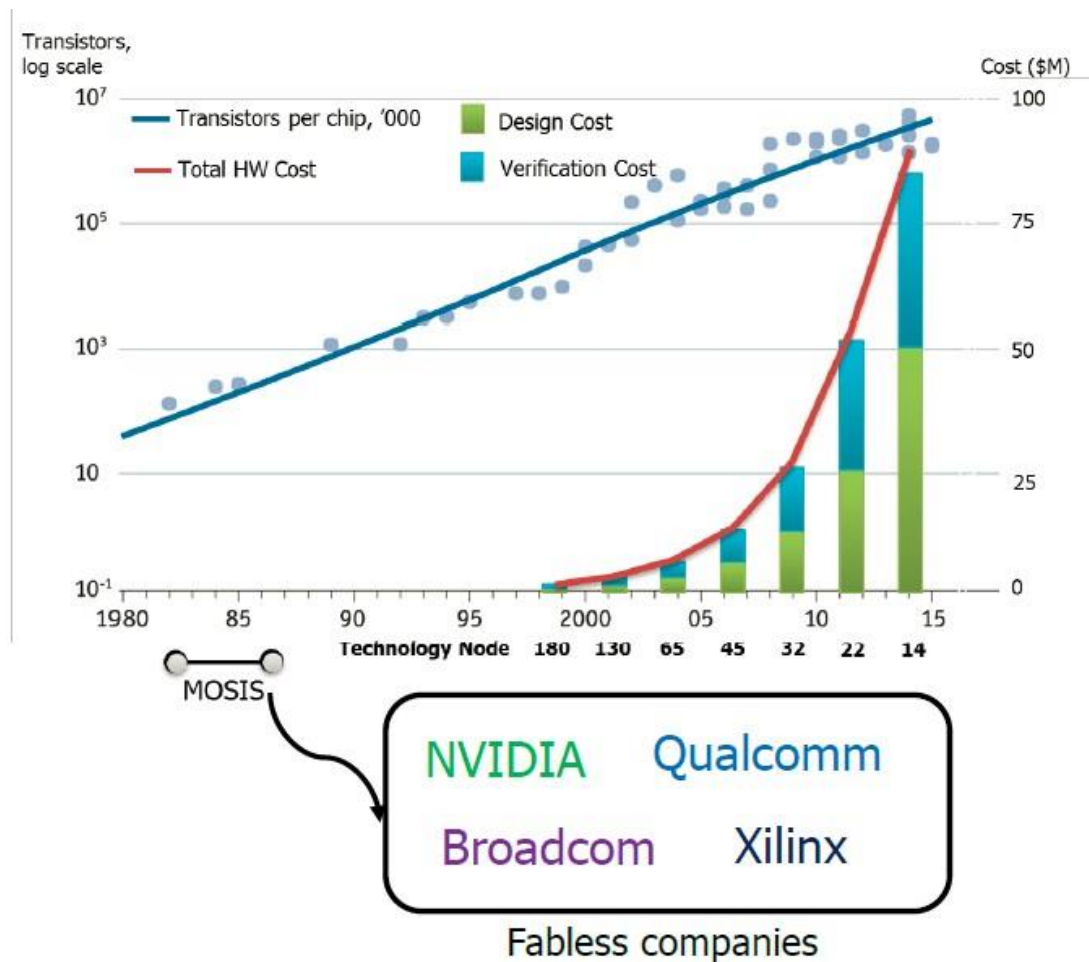
DAPAR促进了半导体产业的分工——设计、制造

★ 芯片设计企业

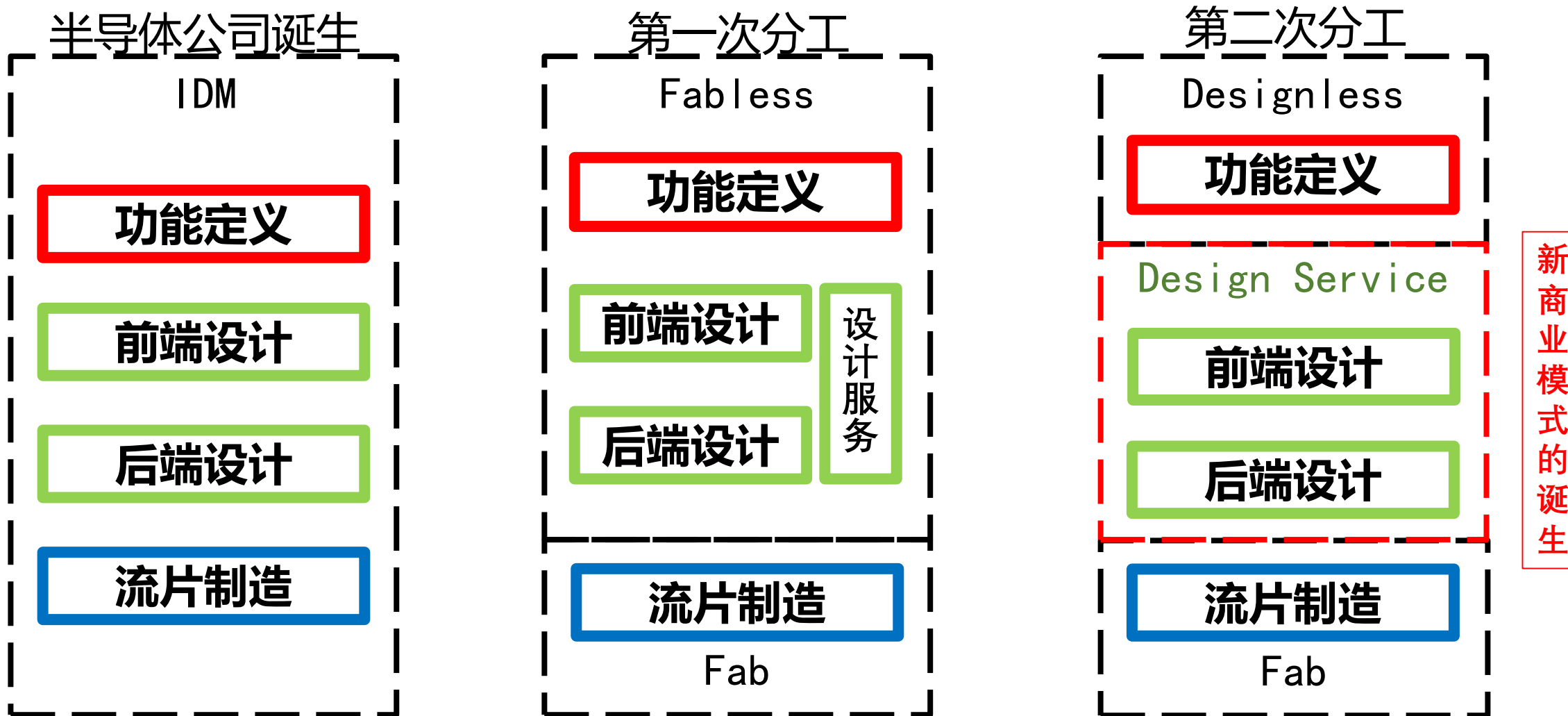
★ NVIDIA,
Qualcomm, Broadcom
Xilinx, AMD

★ 芯片代工企业

★ TSMC, GF, 中芯国际



RISC-V即将催生半导体产业又一次的分工洗牌



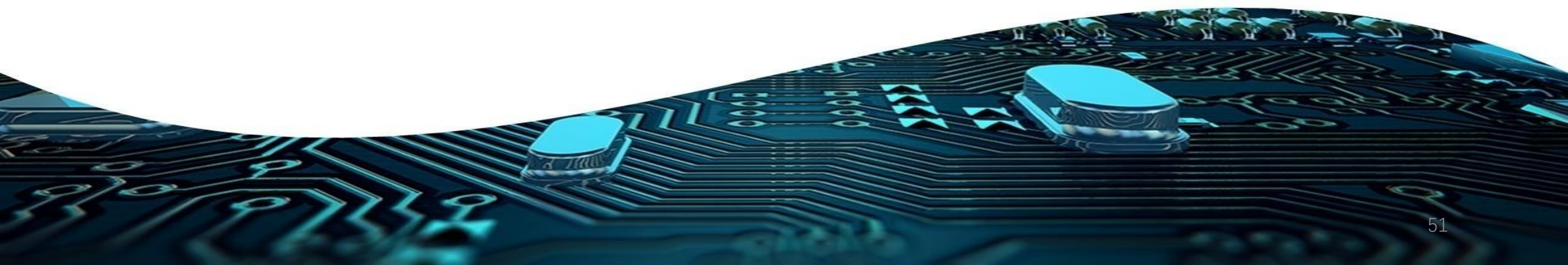
历史上，半导体公司从传统的IDM走到Fabless模式，主要是因为Fab开销过高，成为了半导体公司发展的包袱，而代工厂则提供了一个非常灵活的选项，同样到了今天，芯片设计高昂的费用和时间成本，促使了半导体产业又一次分工

Agenda

- RISC-V的诞生
- RISC-V简介
- RISC-V开源生态发展趋势
- RISC-V行业应用普及现状
- **我们的工作**

个人介绍

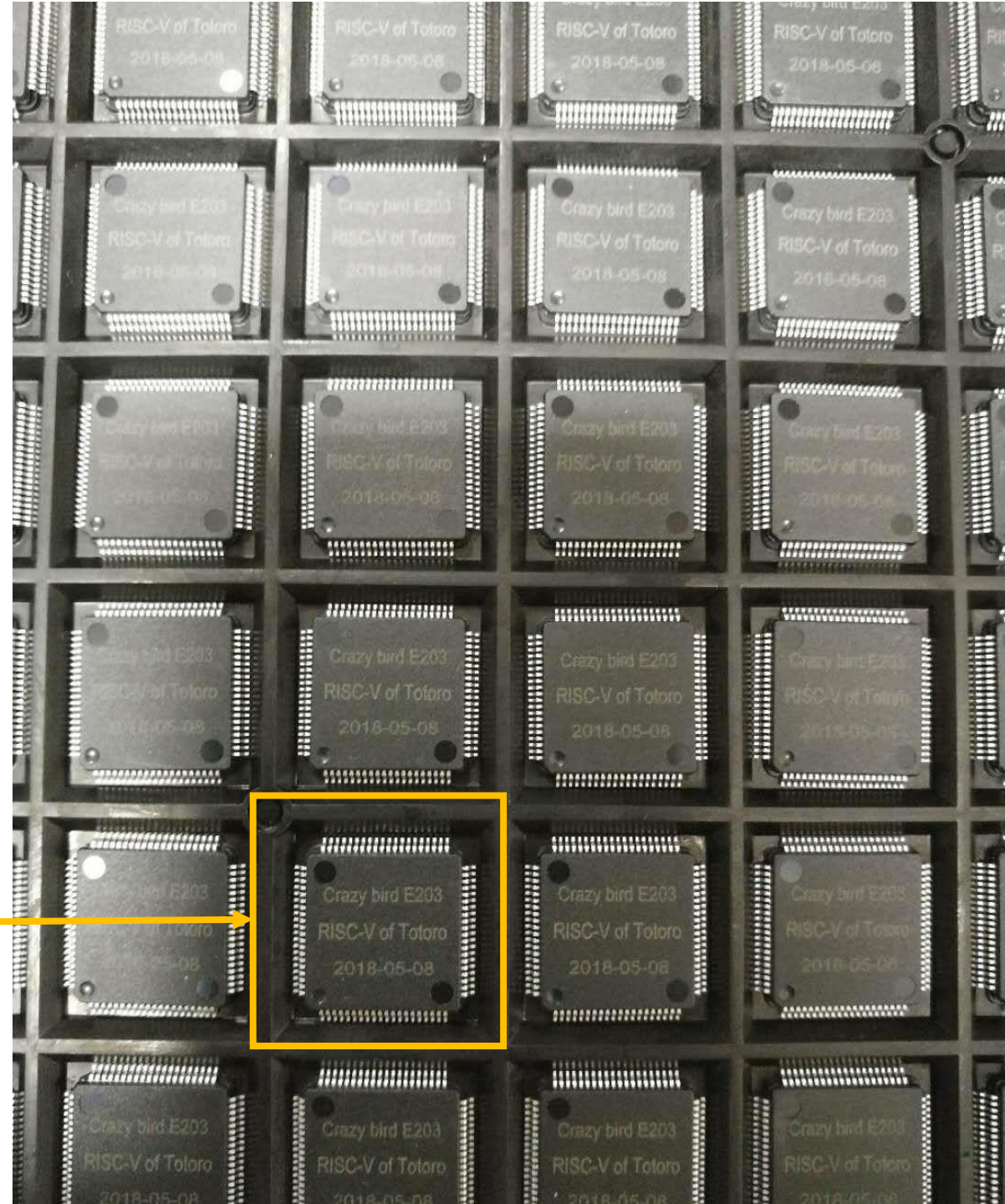
郑云龙，中科院上海微系统所博士，博士后。参与了国内第一颗RISC-V开源处理器内核蜂鸟E203 SOC设计实现，专注从事集成电路高可靠性研究和微处理器芯片SOC设计，拥有丰富的芯片设计与流片经验。



疯鸟E203开源SoC芯片

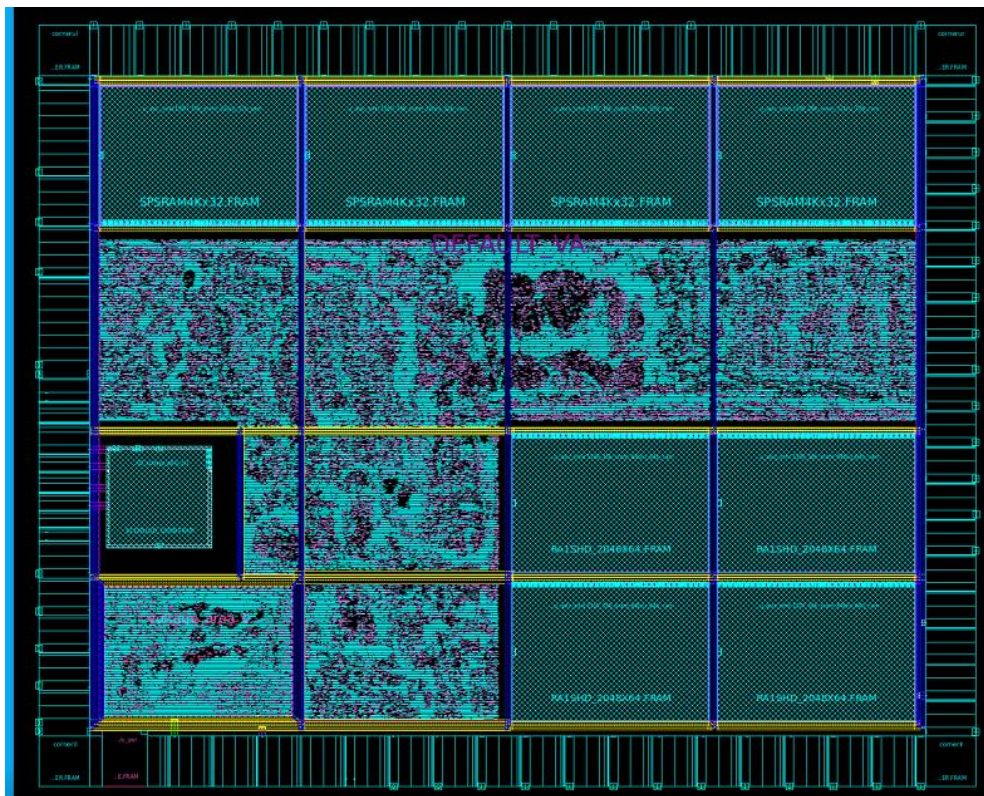
| | Features | Description |
|-------|-------------------------------------|------------------------|
| • CPU | ● 芯片的频率 (MHz) 区间 | 130nm SMIC 50 ~ 150MHz |
| | ● 基于E200系列处理器核 | |
| | ● 使用标准JTAG调试接口 | |
| | ● 支持GDB交互式软件调试能力 | |
| | ● 支持中断控制器 | |
| • 存储 | ● 片上ITCM-SRAM (指令) | 可配置大小, 可配置ECC保护 |
| | ● 片上DTCM-SRAM (数据) | 可配置大小, 可配置ECC保护 |
| | ● 系统SRAM | |
| | ● 可通过QSPI等接口外接其他片外Nor Flash 片外存储 | |
| • 外设 | ● 提供PWM | 3组 |
| | ● 提供SPI, QSPI | 3组 |
| | ● 提供GPIO | 32个pin脚 |
| | ● 提供UART | 1组 |
| | ● 提供WatchDog | 1组 |
| | ● 提供 RTC (Real Time Counter) | 1组 |
| | ● 提供计时器 (Timer) | 1组 |

基于蜂鸟的SOC验证芯片, 2018年5月MPW流片成功



疯鸟E203 SOC实际流片结果

SMIC 130nm工艺下, Signoff 100MHz工作频率



面积: 3.9x3.3平方毫米
功耗: ≤100毫瓦

| IP综合后, 实际面积对比 | 开源 E20E203核 | ARM Cortex-M3核 |
|------------------|----------------|-------------------|
| 包含ITCM/DTCM | 3,595,554 | |
| 不包含ITCM/DTCM | 167,883 | 392,206 |

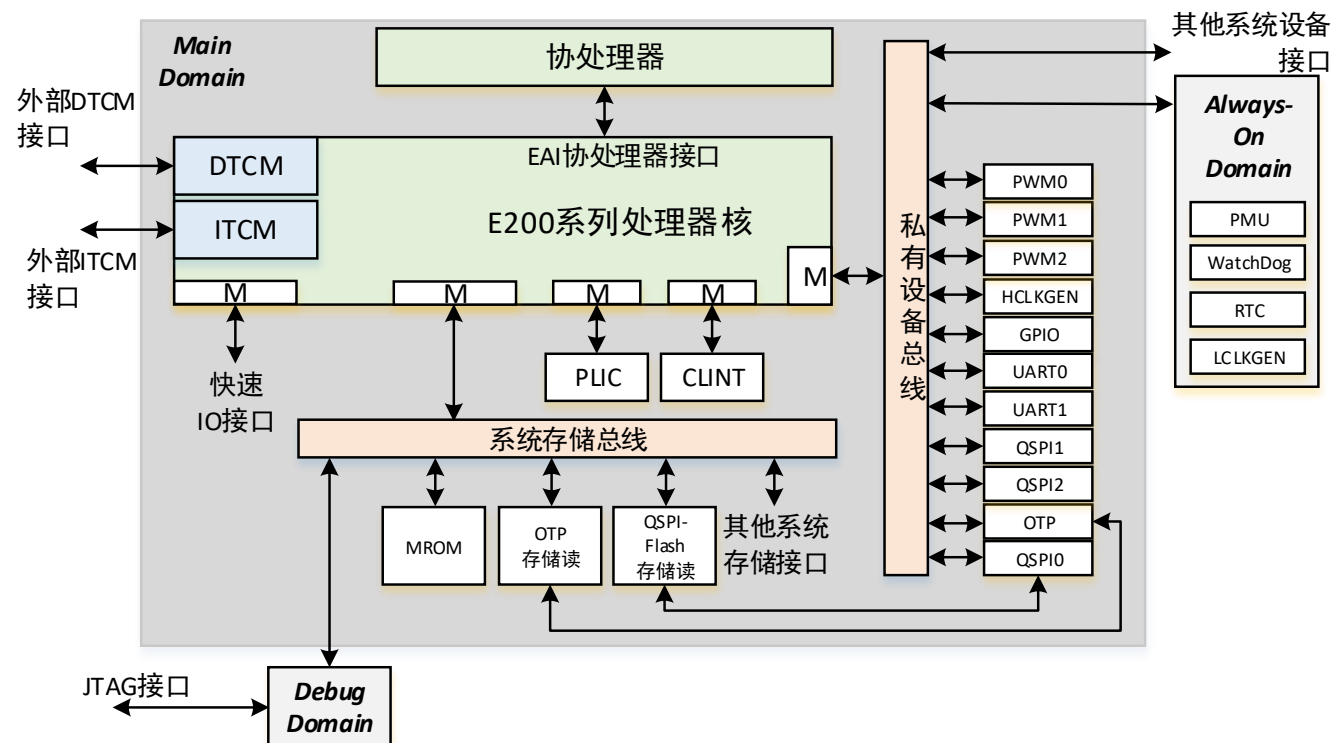
说明: M3架构和E203架构有所不同: M3的ITCM/DTCM是放在CPU外面的, E203核的ITCM/DTCM是放在CPU内部的

相同性能, 面积减小 57%

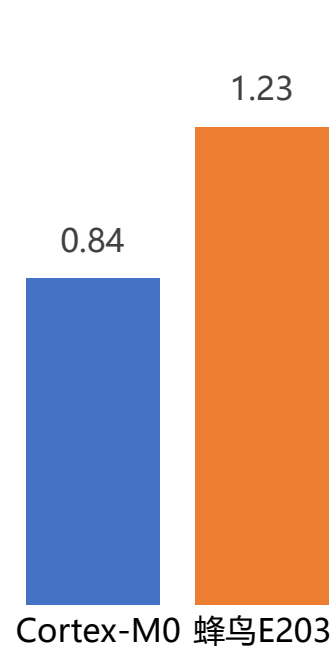
蜂鸟E203架构介绍

全球最小低功耗嵌入式通用32位RISC-V处理器SOC，提供丰富的外围接口支持PWM，SPI，QSPI，GPIO，UART，WatchDog，RTC（Real Time Counter），Timer

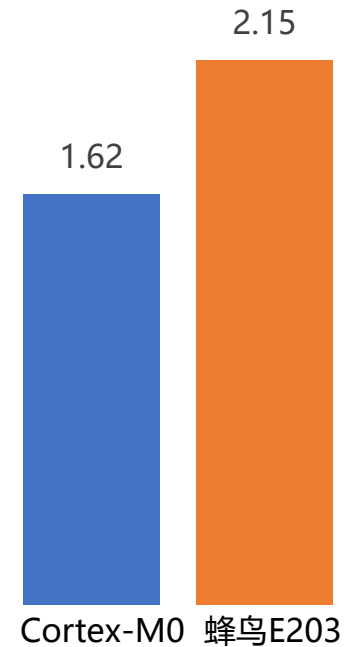
E203开源SOC原型（提供FPGA演示平台）



DhryStone得分 (越高越好)



CoreMarks得分 (越高越好)



蜂鸟RISC-V CPU 支持的RTOS

■ Open Source

- FreeRTOS
- RT-Thread
- Zephyr



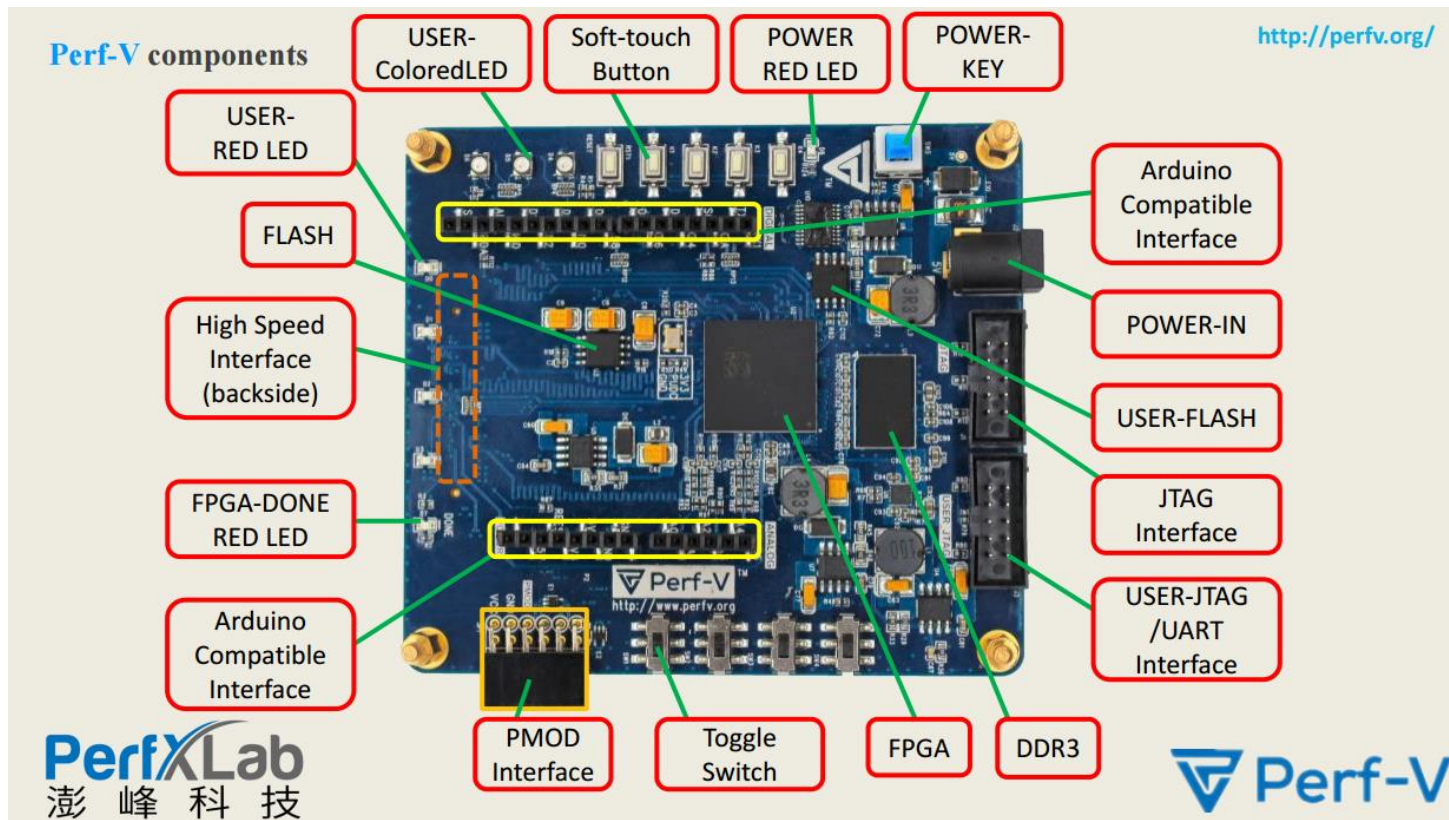
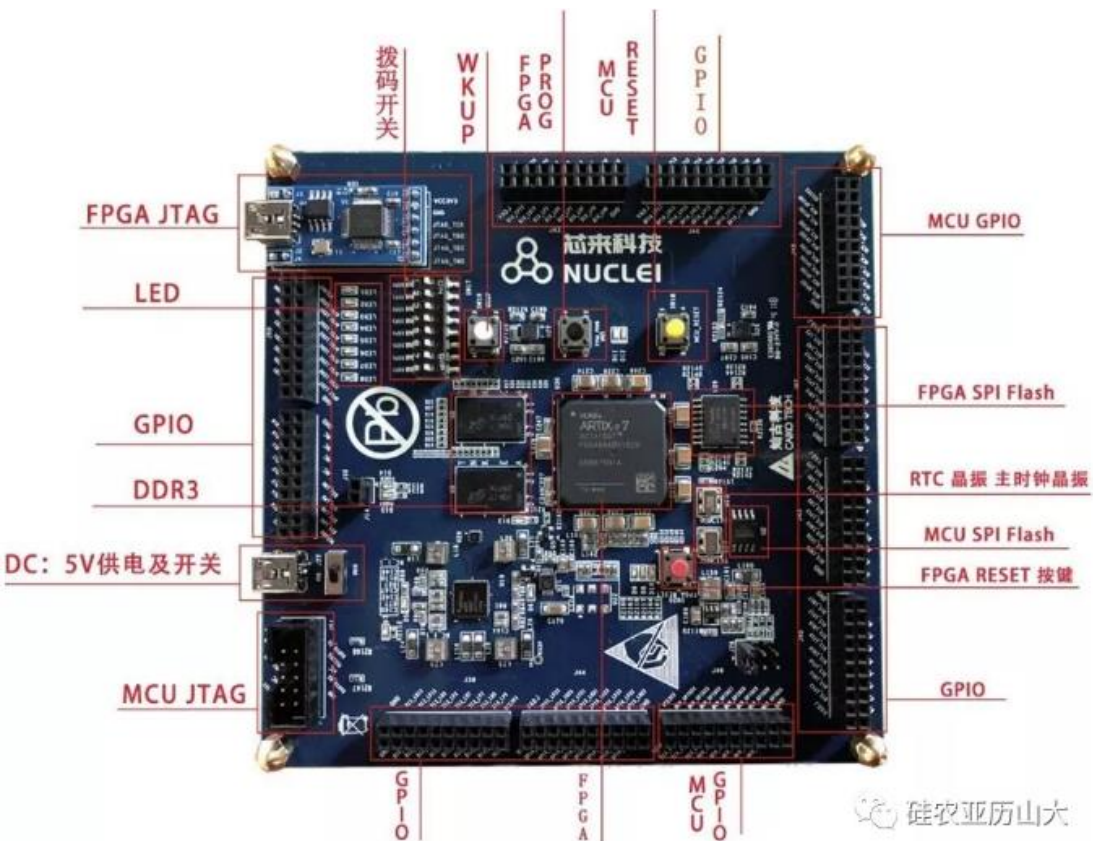
■ Commercial

- Express Logic - ThreadX
- SiLabs - Micrium μ C/OSIII
- Segger - embOS



基于蜂鸟 SOC的FPGA开发板

基于Xilinx 的FPGA-based 开发板



基于蜂鸟 SOC的国产FPGA开发板

FPGA-based MCU Board

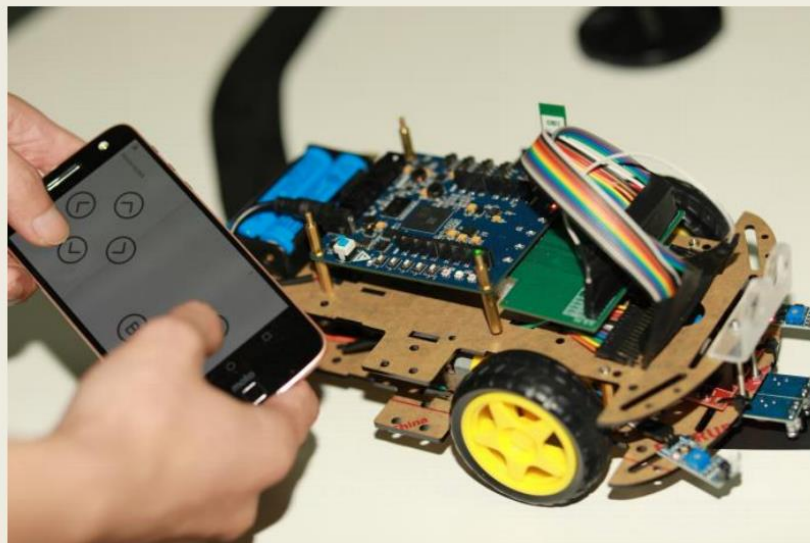
- Anlogic (上海安路) FPGA based MCU Board (<100RMB)
- PANGO (紫光同创) FPGA based MCU Board (<100RMB)

Lichee Tang



基于蜂鸟SOC的FPGA丰富应用开发

DEMO简介——基于RISC-V实现的智能寻迹小车



DEMO简介——贪食蛇游戏视频



RISC-V 完整的工具链与软件开发环境

• C/C++ Compiler

- Standard GNU GCC Toolchain (with both Linux and Windows version)

• Software Development Kit

- Hbird-E-SDK: based on GCC Toolchain, 调试工具链、中断控制器、JVM、LLVM、Python

• Windows and Linux IDE

- Eclipse C/C++ Development IDE
- Note: utilize the whole RISC-V ecosystem, any IDE supporting RISC-V will support Hummingbird E200 (because of standard ISA)

• Software Simulation

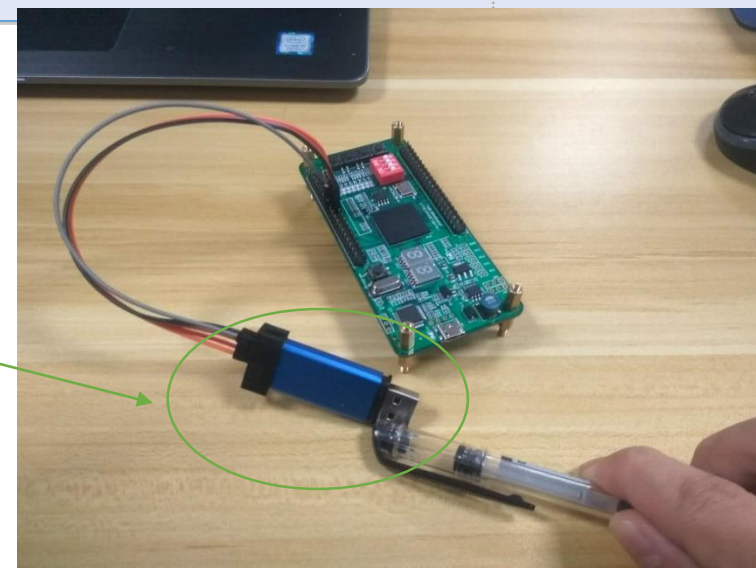
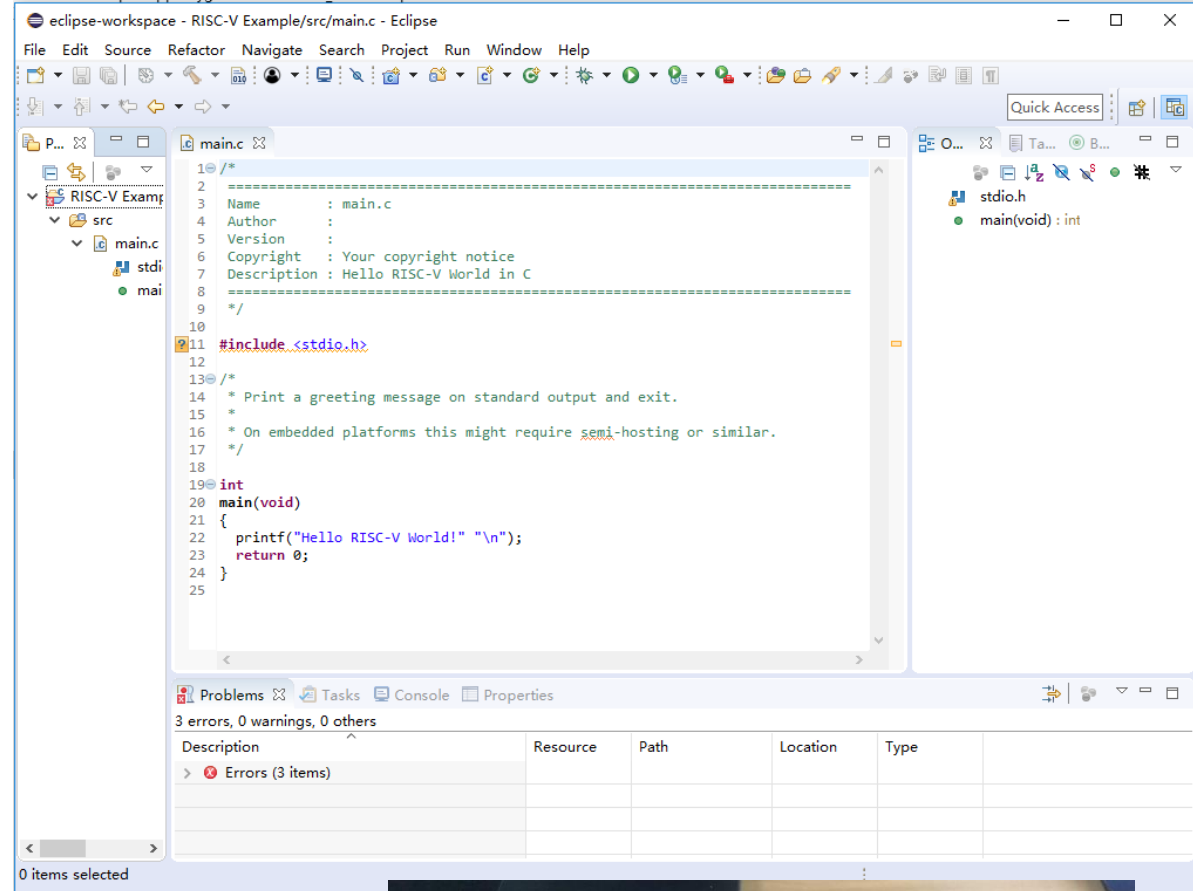
- Based on QEMU

• JTAG Debugger

- JTAG Debugger (USB-Disk size, Cost around 30 RMB)

• RTOS Support

- FreeRTOS
- RT-Thread
- Note: utilize the whole RISC-V ecosystem, any RTOS supporting RISC-V will support Hummingbird E200 (because of standard ISA)



芯片边缘计算，从PPA到APP

- **1.PPA**

- PPA是Performance, Power, Area的简称，集成电路的内部的微观概念，从1958年晶体管诞生之日起，随着摩尔定律和迪纳德定律的延续发展，**集成电路的PPA成为集成电路行业最重要指标，一直在产品中强调更高的性能，更低的功耗，更小的面积（成本）。**

- **2.APP**

- APP是**Application**, Price, Property的简称，集成电路的宏观的概念，现在对于芯片企业来讲要把芯片卖出去在微观的基础实现下，面对新的需求，**需更加关注应用市场，更加关注价格甚至性价比，关注产品整体的性能参数，在AIOT碎片化的市场中寻找出路。**

电子产品周期缩短，快速变化（2~3年）

以手机为例，传统上9个月左右出一款新机型，但是现在中国手机品牌可以做到4个月出一款新机型，且这样的新机型并非简单的外观、存储容量升级，而是要增加新的创新功能

瑞芯微变化的例子

步步高复读机——MP3/MP4——平板电脑——智能音箱

无法猜对市场需求，打造爆款芯片

APP软件面临相似的情形

一款软件APP的现状:

软件APP的开发过程, 版本的快速迭代, 生命周期急剧缩减

软件APP开发解决方案:

基于开源软件, 开源的linux OS、开源的Web设计工具、
python & C++

硬件的解决办法，提供共性可重构平台

- 芯片设计、硬件设计、软件设计（硅、硬、软全栈式解决方案）
- 利用搭积木的方法在6个月内推出芯片产品
- 芯片公司的消亡，系统公司、方案公司、新型芯片设计服务公司的崛起

协助硬件工程师，开发定义硬件产品

- 行业需求是固定的
- 产品需求是变化的

基于开源软件、开源硬件

- 面向工程师友好的树莓派开发板评估
- 软件定义产品的开发支持，python

我们的目标

敏捷开发，5位工程师可以做芯片

IP算法工程师，SOC设计工程师，后端工程师，封装测试工程师，嵌入式硬件工程师

- 为社区提供经过流片验证的高质量RISC-V开源核
- RISC-V处理器核IP、外围IP等
- 逐步构建一套基于开源工具链、开源IP、开源工艺库的SoC芯片设计流程
- 将商业版工具、IP逐渐替换为开源版，并提供更好的本地服务

● 范围

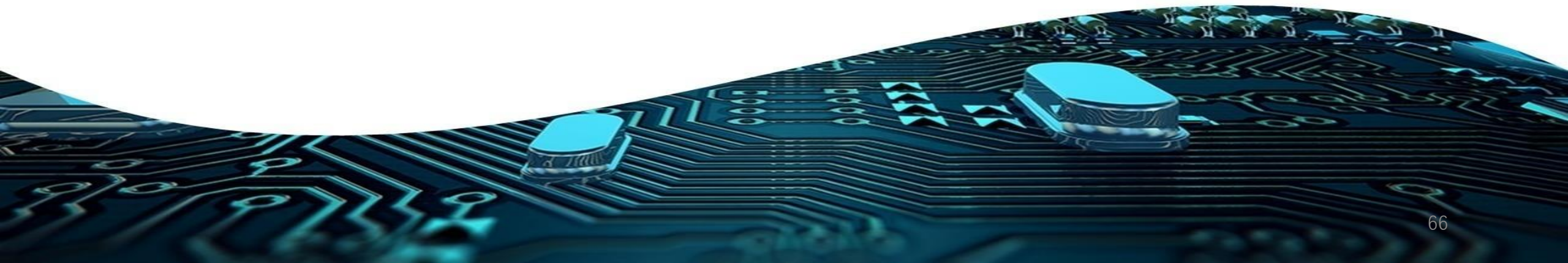
- 为研究机构、系统公司、中小企业提供RISC-V 开源处理器SOC设计与芯片定制开发
- 提供更先进的开源工具，提高芯片设计验证效率，降低芯片开发门槛，缩短芯片开发周期

● 优势

- 2018推出国内首颗开源RISC-V处理器SOC，蜂鸟E203 SOC具有超低功耗/面积特征，针对深度嵌入和物联网领域
- 已初步构建一套基于开源工具链的完整芯片与软件开发流程
- 芯片流片实测数据表明，与arm Cortex-M核相比，具有同等级更高的性能。

物联网边缘计算时代 芯片敏捷开发赋能者

芯可方科技公司



Q & A

Thanks

