AIOT技术挑战与RISC-V 处理器机遇

AIOT technical challenges and RISC-V processor opportunities

李东江

2020年9月

本节提要

1. AIOT 时代嵌入式技术的挑战
2. RISC-V 架构、SoC和平台技术
3. RISC-V 处理器生态和AIOT 开发

Embedded

AIOT 时代嵌入式技术的挑战

嵌入式系统展望

拥抱AI

拥抱AI 相信已是当今电子信息企业的共识。AI技术的诞生，存在于一个个被规划好的数据集中。而我们想让AI落地，在工厂、机场、车站、办公室、医院、教室等现实场景中发挥作用，嵌入式系统将与AI“三要素”– 算力、数据、算法合作找到合适解决方案

万物互联 5G时代来到，万物互联成为基础设 施。接入节点的数量呈现几何级数 的增长，接入节点的安全性成同样 趋势的下降。5G时代下的互联互通，还有赖于很多其他的通信技术实 现连接，比如窄带通信、近场通信 等在不同场景下叠加和融合、安全之忧是一场大考

AIOT

AIOT – 万物互联到万物智联

- 什么是AIOT？
  - **市场层面看**：AIOT是 AI技术与IoT在实际应用中的落地融合
  - **技术层面看**：AIOT 是把AI 技术嵌入到IOT 设备之中
  - AIOT 的典型应用

智慧工厂

对分布各地的企业工厂在生产过程中的半成品、成品进行自动化的质量检测

智慧零售

应用于广泛分布式部署的无人货柜、商铺智能导购等场景，需要迅速识别商品人员

智慧农牧

养猪场分散各地，地点偏远，网络环境较差，需要实现自动盘点与精确料肉比控制

安防监控

边缘侧迅速识别口罩帽子等特定人员，及跌倒等动作检测，降低客户带宽云存成本

AIOT时代的技术挑战

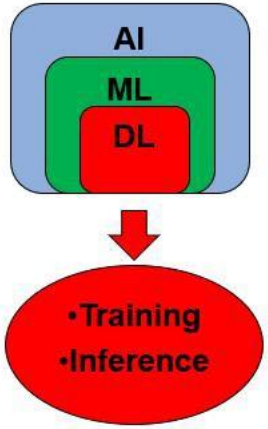
- AIOT 系统融入了“人”，人机交互走向**人机融合**
  - **功能安全**更加重要
  - **信息安全**依然重要
- “物”从传统封闭系统走向开放互联网世界
  - 封闭系统的安全应对开放的挑战
  - 封闭系统**实时性和确定性**如何在开放系统中实现
- AIOT是在AI 技术在IoT实现-**边缘计算**是解决方案
- AIOT≠ AI+IOT，是人、机、物 高度融合的物理信息系统（CPS)

AI 芯片: AIOT 系统的灵魂

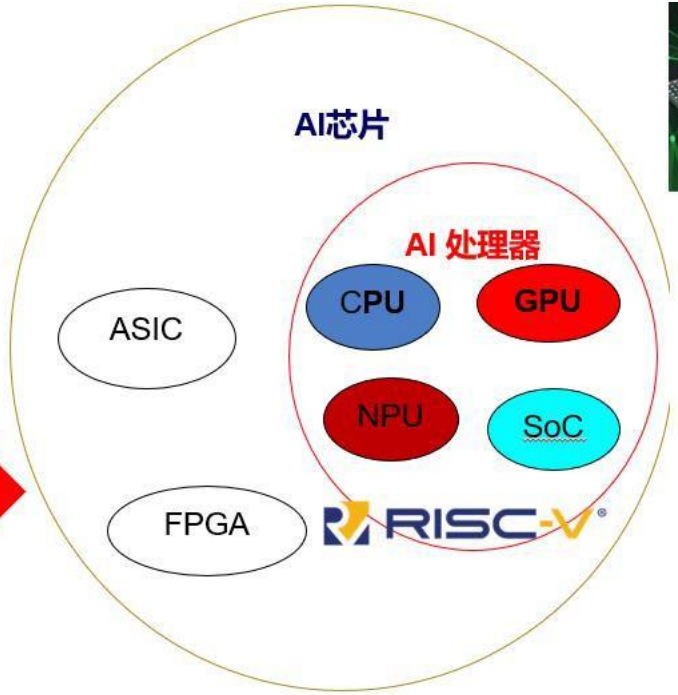
阿里云计算
Alibaba Cloud Computing

amazon web services

FACE++



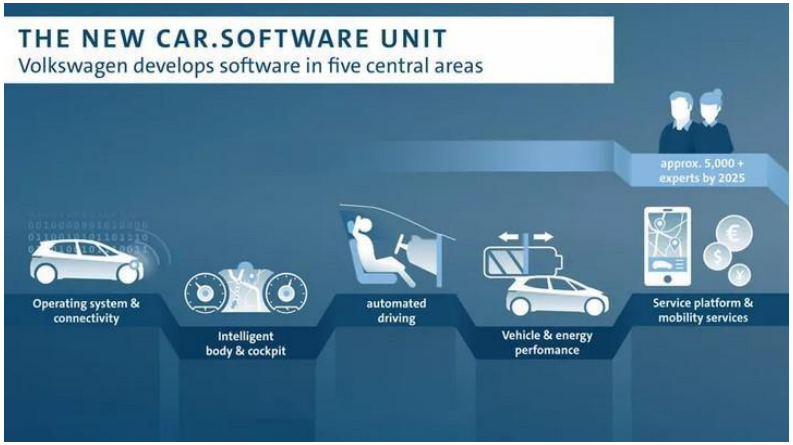
加速



操作系统：AIOT基石

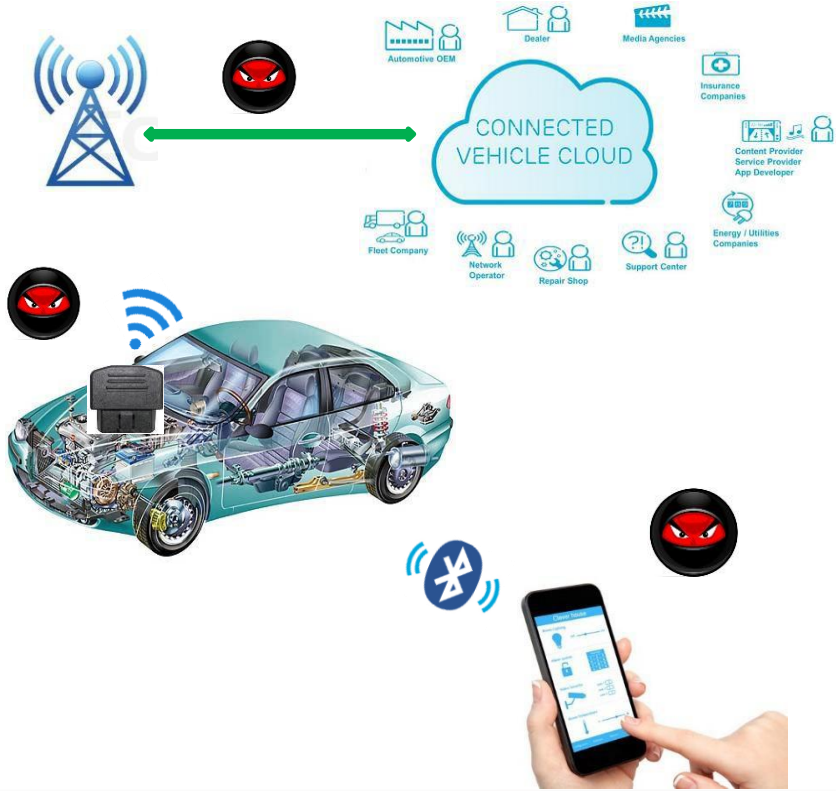
- 操作系统技术面临挑战
  - 自动驾驶驱动-功能安全认证 **(Safety)**
  - 联网时代信息安全-形势严峻 **(Security)**
  - 实时性和确定性（低延时）
  - 体系架构可伸缩
  - 支撑硬件发展-多核、异构、AI 和GPU…
  - 内核技术（微内核和宏内核并存）
  - Linux 和 AGL（Automotive Grade Linux）
  - 规模应用需要虚拟化和容器技术
  - 新一代的编译技术

vw.OS
Car.software



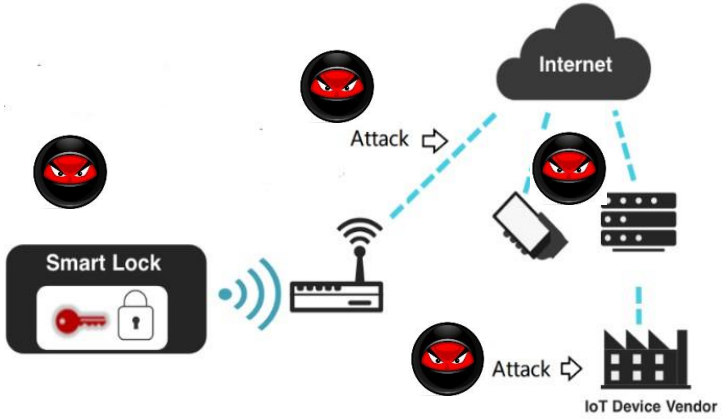
安全：为AIOT 保驾护航

- 更多日常活动因为攻击而可能终断。比如大　量**可穿戴的医疗健康设备**，都通过智能手机　接入互联网，攻击导致设备故障或将危机人 们的健康甚至生命
- 互联网和**大数据**通过传感器收集到了大量物的信息，其内容更加广泛，一旦**信息泄漏**危害更大，比如我们驾车的**汽车的位置，个人信息和疾病信息**
- **电网**、**交通运输**、**核电站**和环境监测等**关键系统**若遇到黑客的攻击，将是毁灭性的危害



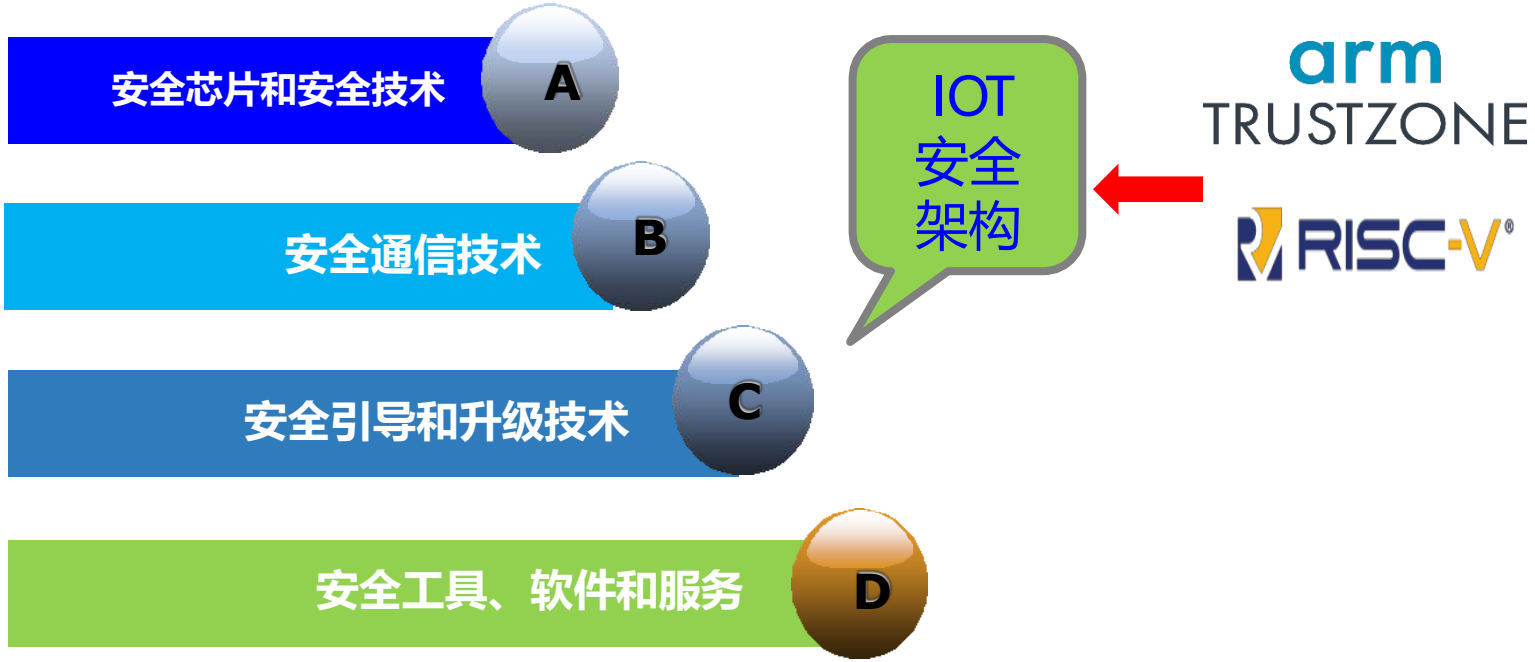
IOT系统安全的目标

- 物联网安全性旨在保护**代码**、**数据**和**系统功能**
  - **代码保护**是保证固件知识产权及其代码的完整性
  - **数据保护**是保证用户数据的机密性并 避免身份盗用
  - **物联网资产**保护，比如健康数据和位置信息，用户账号和密码，交易记录和密钥，以及设备和用户身份等
  - **系统功能安全**应受到重视，以避免设备故障及服务故障



IOT安全架构关键技术

构建安全物联网架构是为了应对多种形式的攻击，安全架构必须在系统中实现多种类型的安全机制，应在设备**硬件**、**软件**两个方面考虑整体安全性

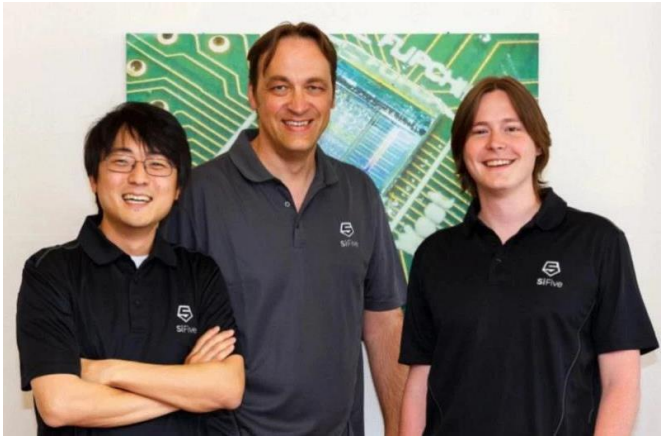


RISC-V® Architecture

RISC-V 架构、SoC和平台技术

什么是RISC-V？

- RISC-V 是一种**开源指令集架构**，它不是一款CPU芯片
- 一个CPU支持的**指令和指令编码**就是这个CPU的指令集（称为ISA）
  - 指令集在软件和硬件设计者之间提供了一个接口。不同的CPU家族：X86、PowerPC和ARM，都有不同的ISA，**RISC-V 是其中唯一的开源ISA**
- RISC-V 起源于加州大学伯克利分校，采用开源BSD 授权 ,任何企业、高校和个人都可以遵循**RISC-V架构指南**设计自己的CPU
- 计算机界泰斗 David Patterson 大力支持RISC-V



主要开发人员： Krste Asanovic 教授（中）、Andrew Waterman （右）和 Yunsup Lee （左）

RISC-V 的发展历程

- 在经过多年的研究以及使用MIPS，SPARC和x86的研究项目之后，2010年**加州大学伯克利分校**的架构小组选择ISA进行下一个研究项目
- 当时项目评估，显而易见的选择：**x86和ARM**
  - 结论：x86-复杂和IP问题；ARM-复杂和IP问题，2010年时候没有64位
- 2010年夏季启动了“ 3个月项目 目标” **开发纯净版ISA “**
  - 设计师： Andrew Waterman , Yunsup Lee, David Patterson, Krste Asanovic
- 四年后，2014年5月，发布了基本用户规范- 在此过程中有许多流片和一些论文
- 名称RISC-V（发音为“risk-five”）代表 **第五代伯克利RISC ISA**

RISC-V指令集介绍

- RISC-V的指令集是**模块化**的组织方式，每个模块使用一个英文字母来表示
- RISC-V最基本、也是唯一强制要求实现的指令集是由 I 字母表示的整数指令子集。使用该整数指令子集，便能够实现完整的软件编译器。其他的指令子集部分均为可选的模块，其代表性的模块包括**M/A/F/D/C，**比如 **RV32IMAC**

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
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

指令集文档”和“特权架构文档

- The RISC-V Instruction Set Manual Volume I: Unprivileged ISA
- The RISC-V Instruction Set Manual Volume II: Privileged Architecture

开源的RISC-V架构处理器核（Core）

- **Rocket Core**
  - 伯克利开发，经典的RV64 设计
- **Zero-riscy**
  - 苏黎世理工，经典的RV32 设计
- **R15CY**
  - 苏黎世理工，可配置成RV32E，面向的是超低功耗、超小面积的场景
- **PicoRV32**
  - 由Clifford Wolf开发RISC-V Core，重点在于追求面积和频率的优化
- **BOOM**
  - 伯克利开发，与Rocket Core不同的是，BOOM Core面向更高的性能
- 中国有**蜂鸟E200**和**WuJian100** 开源项目

SiFive

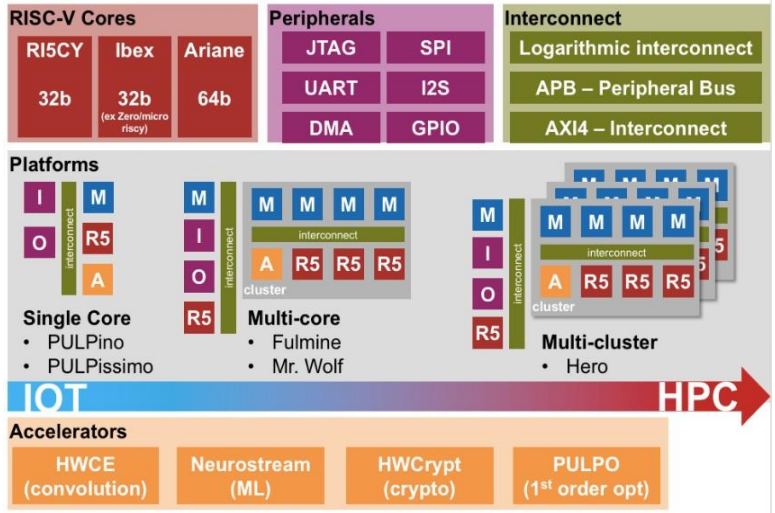
ETH zürich

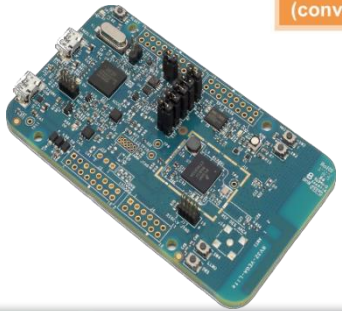
商业RISC-V 处理器核 （IP Core ）

- **SiFive（赛昉科技）**
  - E31/34, S51/54, U54/54-MC E76/U74
- **Andes （台湾晶芯科技）**
  - AX25,A25, D25F,A25MP,N22,N25
- **芯来科技**
  - N200/300/600/900, NX600/900,UX600/900
- **平头哥 T-head**
  - XuanTie C910，XuanTie E902
- SCR1 CORE
  - Syntacore公司使用Verilog设计的一款极低功耗开源RISC-V处理器核
- 商业IP 公司有自己IP Core 开发平台 （Platform）

开源RISC-V 处理器SoC 平台（Platform）

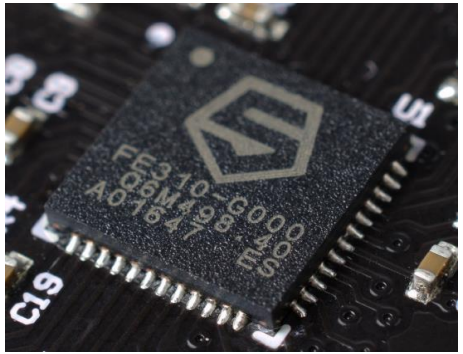
- ETH Zurich **PULPino**
  - Core： RI5CY, Zero-risky
- **LowRISC**
  - Core ： RV32IM
- **Rocket Chip**
  - 基于Chisel语言开发的开源SoC生成器
  - Core: Rocket core/Boom



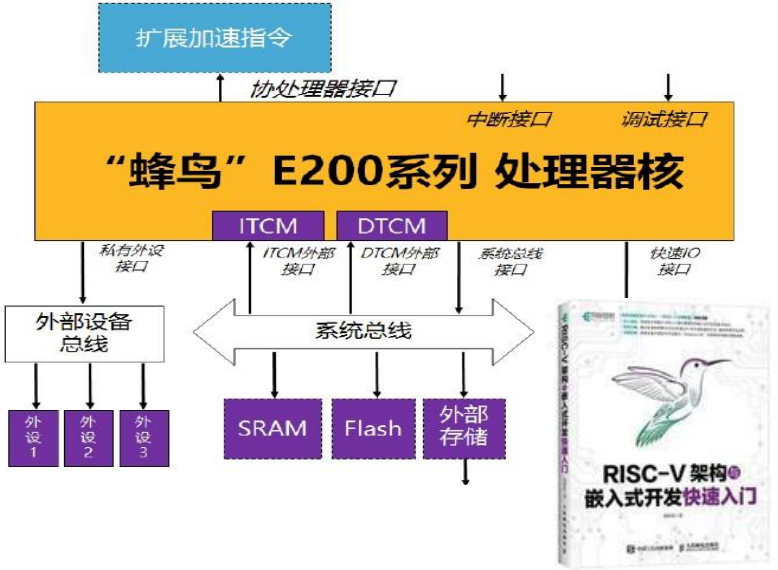


开源 RISC-V 嵌入式处理器（SoC Core）

- **Freedom E310-** SiFive 的 RV32 Core
  - Freedom E310是Freedom everywhere的子系列 E300的一个流片实例，目标应用场合是MCU和IOT，处理器 Core 是E31,支持RV32IMAC指令集。采用180nm工艺流片成功，主频可以达到320M以上，芯片有目前三个型号，**既有Core 还有 工程样片**



- **蜂鸟E200** - 芯来科技胡振波开源项目
  - 国内知名度非常高的开源**软核SoC**

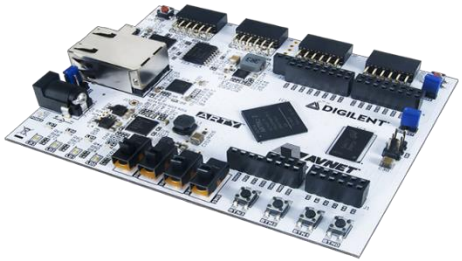


商业RISC-V 嵌入式处理器（SoC）

- **GD32VF103**
  - 兆易开发基于芯来Bumblebee 大黄蜂核（RV32IMAC）芯片
- **Kendryte K210**
  - K210 包含 RISC-V 64 位双核 CPU（Rocket core），每个核心内置独立 FPU，是一个典型 AIOT SoC
- **NXP RV32M1**
  - 集成了4个cores: RISC-V RI5CY core, RISC-V ZERO-RISCY core, ARM CortexM4 core 和 ARM Cortex-M0+ core
- **Microchip PolarFire SoC**
  - 低成本, 多核RISC-V SoC FPGA，4个 64-bit RV64GC RISC-V cores 可运行Linux, 一个单核 RV64IMAC 做real-time 任务

Core 、Platform和SoC软核和SoC 芯片的选择

- **芯片设计者可选择RISC-V Core 和SoC Platform 构建自己的芯片**
  - 比如使用 PULPino 平台开发 SoC 芯片，内核使用 RI5CY, Zero-risky，国内企业和高校研究项目在用
- **嵌入式和物联网系统开发者可以使用 RISC-V SoC 芯片**
  - 比如，选择GD32VF103 系列 芯片做嵌入式项目开发，在某些应用上性价比或许优于ARM CM3芯片，开发板有5种以上
- **高校和研究机构可以选择开源RISC-V Core 或者 SoC 软核在FPGA平台上进行计算机体系架构、OS 和编译技术教学和研究工作**
  - 比如，Arty FPGA 开发板上实现一个 SiFive 开源Freeedom E310 MCU 并有相应软件工具链支持
  - 比如，T-Head FPGA 平台实现 一个开源 wujian 100 MCU 开发



RISC-V

Embedded

RISC-V 处理器生态和AIOT 开发

RISC-V为 AIOT 嵌入式开发带来什么?

- RISC-V最大的优势是**开源和免费**
  - 开源是新的经济方式，是成功的商业之道， 学习最好的途径
  - ISA开源意味着开发者可以针对特定应用场景，创造自己的芯片架构
  - 免费可以降低入门的门槛，**让草根开发者进入芯片设计**
- RISC-V 第二个优势是**简单和灵活**
  - 基础的指令集有50条，模块化的4个基本指令集让设计者开发出很简化的RISC-V CPU，功耗可以很小，代码密度低，**覆盖51-ARM A系列 处理器**
- RISC-V 第三个优势是**高效和安全**
  - 通过预留编码空间和用户指令支持扩展的指令集
  - 通过指令集扩展实现**运算加速和物联网安全**

RISC-V 嵌入式软件生态

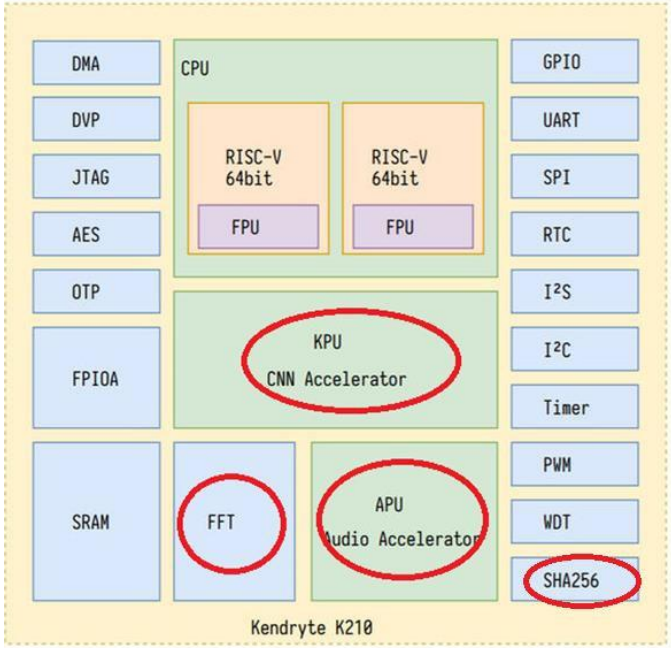
- **开源GNU 工具链支持 RISC-V：**
  - riscv gcc GCC 编译器
  - riscv binutils： 二进制工具 链接器 汇编器等
  - riscv gdb GDB 调试工具等
  - OpenOCD – 运行于PC上的开源调试软件，控制JTAG硬件，可以将它理解为一种GDB服务程序
  - 如 SiFive Freedom Studio AndesSight 和Nuclei Studio IDE
- **SEGGER - JLINK Probe 和 Embedded Studio RISC-V IDE**
- **IAR - IAR Embedded Workbench for RISC-V**
- **Embedded OS**
  - **FreeRTOS**、**Zephyr OS**、 Thread X、µC/OS、RIOT……
  - **RT-Thread**、Huawei LiteOS、Sylix OS…..

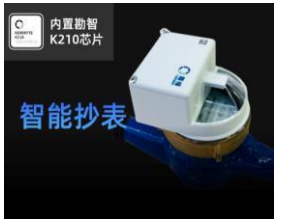
RISC-V 指令集扩展概述

RISC-V ISA 以模块化方式设计，ISA具有可根据需要启用或禁用的几组指令（ISA扩展），其中 一组很特殊，它没有预定义的指令，设计人员可以为他们想要加速应用程序添加所需的任何指 令，因为它不会破坏任何软件兼容性这是一项非常强大的功能，看下面芯来对扩展指令应用理解

RISC-V 架构 AIOT应用

- Kendryte K210集成了**机器视觉和机器听觉**的SoC芯片，采用台积电28nm工艺，采用双核64位处理器，具有更好的功耗性能、稳定性和可靠性
- Kendryte K210 定位于 **AI 与 IoT 市场的 SoC**，同时是使用非常方便的 MCU
- K210 是 双**RV64 GC** Core，**MAFD** ISA 指令标准扩展
- **KPU** 是通用神经网络处理器，内置卷积、可以对人脸或物体进行实时检测
- **FFT** 加速器是用硬件的方式来实现
- **SHA256** 加速器是 SHA-256 的计算单元





结论

- 许多新 RISC-V 芯片是针对 **AIoT 应用型SoC**
- **RISC-V 将与ARM 同行发展**，一个系统中同时使用ARM和RISC-V
- 嵌入式系统将**迎接RISC-V 时代**， IOT和 AIOT 产品开发是未来热点
- RISC-V 非常适合高校电子信息相关**研究项目和教育课程**，一种全新开源硬件模式，构建当今社会不可多得的**合作创新环境**!

"There are no serious technical or practical issues with RISC-V. It will eventually supplant x86 and ARM as the primary instruction set for microprocessors. It will fundamentally change the computing world."

- 华盛顿大学计算机工程学院 Michael Taylor 教授

Thank you !