



编者按：为帮助嵌入式、物联网和人工智能的从业者对嵌入式系统相关技术形成全面、客观的认识，本刊特开设“技术专题”栏目，力邀业内有影响力的专家聚焦当下热点，撰写技术与产业深度文章。近两年，伴随 AIoT 的蓬勃发展、错综复杂的政经环境，开源 RISC-V 指令集架构受到广泛关注，生态日益完善，但其应用发展之路还面临诸多挑战。本期专题围绕 RISC-V 处理器技术、产业生态现状与发展前景，以及 RISC-V MCU 芯片设计与应用等进行深入阐述。

RISC-V 处理器嵌入式开发概述

何小庆

(嵌入式系统联谊会, 北京 100191)

摘要：本文介绍了 RISC-V 指令集的起源和发展历史；讨论了各种 RISC-V CPU 核、SoC 设计平台和芯片产品的技术特点及选型要点；重点阐述了 RISC-V 给嵌入式系统带来的益处、RISC-V 指令扩展的特点与安全应用，介绍了支持 RISC-V 开源和商业软件开发工具和操作系统的现状；最后展望了 RISC-V 在教育 and 产业应用方面的发展趋势。

关键词：RISC-V；开源指令集；K210；SiFive；蜂鸟 E200；FreeRTOS

中图分类号：TP31

文献标识码：A

Overview of RISC-V Processor Embedded Development

He Xiaoping

(Embedded System Association, Beijing 100191, China)

Abstract: The article introduces the origin and development of the RISC-V instruction set, briefly describes the technical characteristics and selection guidelines of various RISC-V CPU cores, SoC design platforms and chip, features of RISC-V instruction extension and safety application. The article focuses on how RISC-V brings embedded systems the benefits, the open source and commercial software development tools and OS support the status of RISC-V, and finally looks forward to the trend of RISC-V in education and industry development.

Key words: RISC-V; open ISA; K210; SiFive; Hummingbird E200; FreeRTOS

引言

过去二十年，ARM 在移动和嵌入式领域成果丰硕，在 IoT 领域正逐渐确定其市场地位，其他商用架构（如 MIPS）逐渐消亡。不仅如此，ARM 还在进军 Intel 所在的 x86 市场，已经对传统 PC 和服务端领域造成一定压力。RISC-V 开源指令集的出现，迅速引起了产业界的广泛关注，科技巨头很看重指令集架构（CPU ISA）的开放性，各大公司正在积极寻找 ARM 之外的第二选择，RISC-V 无疑成为必然选择。RISC-V 被全球范围内的大学陆续采纳为教材以替代 MIPS 和 x86 架构，政府和企业采纳 RISC-V 为标准指令集，开源的 CPU 核和 SoC 芯片不断涌现，生态环境逐渐丰富，开发者社区愈来愈活跃。在错综复杂的国际政治经济环境的大背景下，芯片成为中国科技的新

制高点，CPU “自主可控”与“普世通用”存在天然的矛盾，而 RISC-V 开源指令集架构正在成为一种解决之道^[1]。

1 什么是 RISC-V？

一个 CPU 支持的指令和指令的字节级编码就是这个 CPU 的指令集（ISA），指令集在计算机软件和硬件之间搭起了一个桥梁。不同的 CPU 家族（如 x86、PowerPC 和 ARM）都有不同的 ISA，RISC-V 是其中唯一的开源 ISA。RISC-V 是一种开源的指令集架构，它不是一款 CPU 芯片，甚至不是一个完整的指令集，它是指令集规范和标准。RISC-V 起源于加州大学伯克利分校，在 2010 年夏季 Krste Asanovic 教授带领他的两位学生（Andrew Waterman 和 Yunsup Lee）启动了 3 个月的项目，针对 x86 和 ARM 架构复杂和 IP 授权的问题，旨在开发简化和开

放的指令集架构。

RISC-V 基金会创建于 2015 年,是一家非盈利组织。基金会董事会由 Bluespec、Google、Microsemi、NVIDIA、NXP、UC Berkeley、Western Digital 七家单位代表组成,主席目前是 Krste Asanovi 教授。基金会为核心芯片架构制定标准和建立生态,标准是公开免费下载的。基金会旗下有 300 家以上的付费成员,包括高通、NXP、阿里巴巴和华为等,RISC-V 基金会成员可以使用 RISC-V 商标,RISC-V 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。RISC-V 基金会总部从美国迁往瑞士,并于 2020 年 3 月完成在瑞士的注册,基金会更名为 RISC-V 国际基金会(RISC-V International Association)^[2]。

十年来 RISC-V 蓬勃发展,在 CPU IP 核、平台、SoC 芯片和应用方面都有了相当迅速的发展。比如,应用上有西部数据设计的 SSD 和 HDD 控制器(内核是 SweRV Core),中科蓝芯开发的 TWS 蓝牙耳机芯片,嘉楠科技的 K210 AIoT 芯片,沁恒电子的蓝牙 MCU、32 位通用 MCU 和高速接口的 MCU,它们的内核都是 RISC-V。

2 RISC-V 指令集介绍

RISC-V 的指令集使用模块化的方式进行组织,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,比如某款 RISC-V 处理器内核是 RV32IMAC,即代表实现了 I/M/A/C 指令集。部分指令集描述见表 1。

表 1 RISC-V 部分指令集描述

| 基本指令集 | 指令集 | 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 |

RISC-V 指令集发展变化中,32I 和 64I 已经冻结,MAFDQC 指令扩展冻结了,有些指令集(如 32E、128I、LBJTPV 和 ZAM 原子访问扩展)还在开发中。指令集扩展是 RISC-V 的技术特色,广泛征求会员单位和产业界的意见是 RISC-V 发展合理的路径^[3]。

3 RISC-V 处理器核、SoC 平台和 SoC 芯片

3.1 RISC-V 处理器核

在详细阐述 RISC-V 处理器嵌入式开发之前,我们首先梳理一下几个概念:RISC-V 处理器核心(Core,简称核),SoC 平台和 SoC 芯片,以及开发者如何选择它们。自 RISC-V 架构诞生以来,市场上已有数十个版本的 RISC-V 核和 SoC 芯片,其中,某些是开源免费,某些是商业公司开发用于内部项目,还有一些是商业公司开发的处理器核和平台。西部数据的 SweRV 架构(RV32IMC)是一个 32bit 顺序执行指令架构,具有双向超标量设计和 9 级流水线,采用 28 nm 工艺技术实现,运行频率高达 1.8 GHz,可提供 4.9 CoreMark/MHz 的性能,略高于 Arm 的 Cortex-A15,已经在西部数据的 SSD 和 HDD 控制器上使用,SweRV 项目是开源的项目。

典型的开源 RISC-V 核有 Rocket Core,它是美国加州大学伯克利分校开发的一个经典的 RV64 设计,伯克利分校还开发了一个 BOOM Core,与 Rocket Core 不同的是,它面向更高的性能。苏黎世理工大学(ETH Zurich)开发的 Zero-riscy 是经典的 RV32 设计,该大学另外一款 R15CY Core 可配置成 RV32E,面向的是超低功耗、超小芯片面积的应用场景。由 Clifford Wolf 开发 RISC-V Core PicoRV32 侧重于追求面积和 CPU 频率的优化。

开源的核用于研究和教学很合适,但是用于商业芯片设计还有许多工作要做。SiFive(美国赛昉科技)由 Yun-sup Lee 创立,他也是 RISC-V 的创始人之一。2017 年公司发布首个 RISC-V 核和 SoC 平台家族,以及相关支持软件和开发板,这些芯片包括采用 28 纳米制造技术的 64 位多核 CPU U500,支持 Linux 操作系统,以及采用 180 制造纳米技术的多外设低成本 IoT 处理器核 E300。开发 RISC-V 处理器核的厂商还包括 Codaip、Syntacore、T-Head(平头哥)、Andes(晶芯科技),以及创业公司芯来科技和优矽科技。

3.2 RISC-V SoC 平台

知名的 RISC-V 处理器 SoC 平台有瑞士理工大学的 PULPino、开源项目 LowRISC 以及 Rocket Chip-伯克利分校基于 Chisel 开发的开源 SoC 生成器。芯来科技胡振波发起的蜂鸟 E200 开源项目^[4],配合他的图书,是在国内知名度非常高的开源软核 SoC 平台之一(见图 1)。

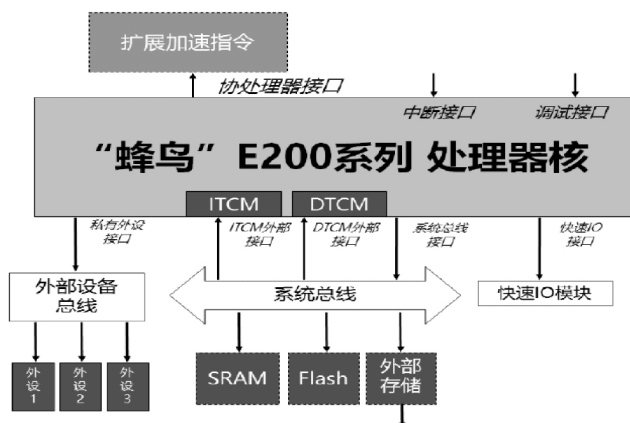


图1 蜂鸟 E200 RISC-V 处理器

3.3 RISC-V SoC 芯片

RISC-V 处理器 SoC 芯片近年发展迅速,知名度较大的通用性 SoC 芯片有兆易创新开发的 GD32VF103 MCU 芯片,该芯片基于芯来科技 Bumblebee 内核(RV32IMAC)。GD32VF103 系列提供了 108 MHz 的运算主频,16~128 KB 的片上闪存和 6~32 KB 的 SRAM,有 4 个 16 位通用定时器、2 个 16 位基本定时器和 2 个多通道 DMA 控制器。GD32VF103 MCU 全新设计的中断控制器(ECLIC)提供了多达 68 个外部中断并可嵌套 16 个可编程优先级,以增强高性能控制的实时性。

GD32VF103 MCU 开发板有 GD32VF103V-EVAL 全功能评估板以及 GD32VF103-START 入门级学习板。除此之外,还有芯来科技开发的 RV-START,以及 Sipeed Longan Nano 开发板^[5]和 IAR 最新的 IAR RISC-V GD32 EVAL 评估套件^[6],见图 2。



图2 IAR RISC-V GD32V 评估套件

嘉楠科技 K210 是一个 AIoT SoC 芯片。K210 包含

RISC-V 64 位双核 CPU,采用双 RV64 GC Core,MAFD ISA 指令标准扩展。K210 包含 KPU 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,K210 的 FFT 加速器是用硬件的方式来实现^[7]。

NXP RV32M1 集成了 4 个核: RISC-V RI5CY 核, RISC-V ZERO-RISCY 核, ARM Cortex-M4 核和 ARM Cortex-M0+ 核。从专业人士视角看, RV32M1 更像是工程实验样品,供开发者评估使用。为此, NXP 为开发者创建 <https://open-isa.org/社区>,维护工具链和软件生态,通过赠送开发板和举办大赛为开发者学习 RISC-V 嵌入式开发提供了便利,在早期市场培育期发挥了重要的作用^[8],见图 3。

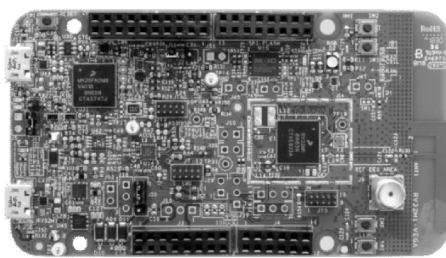


图3 NXP RISC-V 织女星开发板

Microchip PolarFire SoC 芯片是一款低成本、多核 RISC-V SoC FPGA,包含了 4 个 64-bit RV64GC RISC-V 应用核,可运行 Linux,一个单核 RV64IMAC 做实时和监控任务,这是一款十分适合工业控制和物联网应用的开发平台。

3.4 RISC-V 核、平台和芯片究竟该如何选择?

近期关注 RISC-V 处理器研究、开发和教学的人士开始增多,笔者的建议:

① 芯片设计者可选择 RISC-V 核和 SoC 平台构建自己的芯片。比如,使用 PULPino 平台开发 SoC 芯片,内核使用 RI5CY 和 Zero-riscy,国内企业和高校研究项目都有在使用。

② 建议嵌入式和物联网系统开发者使用 RISC-V SoC 芯片。比如,选择 GD32VF103 系列 MCU 芯片做嵌入式项目开发, GD32VF103 有多款开发板和开发工具链支持。AIoT 应用可以选择 K210, K210 开发软件 SDK 非常成熟,支持 FreeRTOS 和裸机,最近 Linux 5.8 正式将 K210 RISC-V 纳入主线, K210 已经成功应用在人脸识别和智能抄表等机器视觉和机器听觉领域。

③ 高校和研究机构可以选择开源 RISC-V 核在 FPGA 平台上进行计算机体系架构、操作系统、编译技术以及嵌入式系统的教学和研究工作。比如, Arty FPGA 开发板上实现一个 SiFive 开源 Freedom E310 微控制器的技术已经非常成熟,相应软件工具链支持得很好^[9],见图 4。

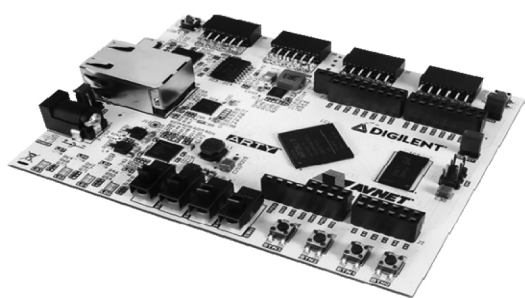


图4 可以配置 RISC-V 软核的 Arty FPGA 开发板

4 RISC-V 给嵌入式系统带来的优势

我们上一节讨论了 RISC-V 处理器核、平台和芯片。显然,嵌入式与物联网以及 AIoT 是 RISC-V 最活跃的应用市场,RISC-V 给嵌入式系统带来许多优势,我仅就以下三点加以阐述:

① 开源和免费。开源是新的经济方式,是成功的商业之道,也是学生和工程师学习的最佳途径。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,免费则可以降低芯片设计门槛,让草根开发者进入芯片设计领域。

② 简单和灵活。RISC-V 基础的指令集有 50 条,模块化的 4 个基本指令集能让设计者开发出简化的 RISC-V CPU,代码密度和功耗都很低的芯片,可覆盖从 8051~ARM A 系列各种嵌入式处理器。

③ 高效和安全。RISC-V 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。物联网保护的一种通用的途径是分层,分为信任执行环境(TEE)和非信任环境。RISC-V ISA 的设计将 TEE 硬件要求定义为标准规范的一部分,可以在任何 RISC-V 芯片上实现,包括配置物理内存保护(PMP),PMP 类似 ARM 处理器内存保护单元(MPU)。

HEX-FIVE 公司在 RV32 Core 做了 IoT 分区保护的应用案例^[10],见图 5。软件可参考开源代码 <https://github.com/hex-five/multizone-secure-iot-stack>。从图中可以看到,SoC 芯片是 X300 Bitstream,采用开源 Rocket RV32IMAC 内核, Digilent ARTY A7 FPG 开发板配有以太网。软件上 Multi-Zone 安全可信执行环境配置成 4 个区:Zone 1 区,运行 FreeRTOS 三个任务,分别是 CLI

任务、PWM LED 任务、机器人手臂控制任务;Zone 2 区,移植以太网,运行 PicoTCP 软件协议,TCP 信道进行了 TLS 加密;Zone 3 区,运行 WolfSSL TLS 1.3,存储信任根,加密密钥、密码以及受保护的文件等放在这个区域;Zone 4 区,UART 本地终端应用。

5 RISC-V 嵌入式软件生态系统

RISC-V 软件生态是以开源为主,商业软件作为辅助支撑力量。开源 GNU 工具链支持 RISC-V,包括 riscv gcc 编译器、riscv binutils 二进制工具链接器汇编器、riscv gdb 调试工具以及 OpenOCD,这是一款运行于 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。

5.1 开源 GNU 工具链的软件

目前,市场上开源的 GNU 工具软件常见的有 SiFive Freedom Studio、AndesSight 和 Nuclei Studio IDE。这些软件基本针对自家企业 RISC-V 处理器核开发和优化,集成开发环境基于 Eclipse 开发而成。如果开发者有兴趣,可以自己下载 jdk-8u101-windows-x64.exe, Eclipse IDE for C/C++ developers, GNU MCU Eclipse Windows Build Tools, OpenOCD,以及 riscv32-unknown-elf-gcc,搭建一个 RISC-V 开发环境。QEUM 处理器模拟器已经支持 RV32 和 RV32 指令集,这对于 CPU 种类繁多的 RISC-V 家族是一件好事,开发者可以执行软件代码和操作系统,比如 FreeRTOS、Zephyr 和 Linux^[11]。

商业嵌入式软件公司已经开始支持通用的 RISC-V ISA RV32 以及一些公司的处理器核和芯片,其中, GD32VF103 芯片和 SiFive E310、瑞典 IAR 的 IAR Embedded Workbench for RISC-V 和德国 Segger Embedded Studio RISC-V 是目前市场知名度比较高的几款产品。

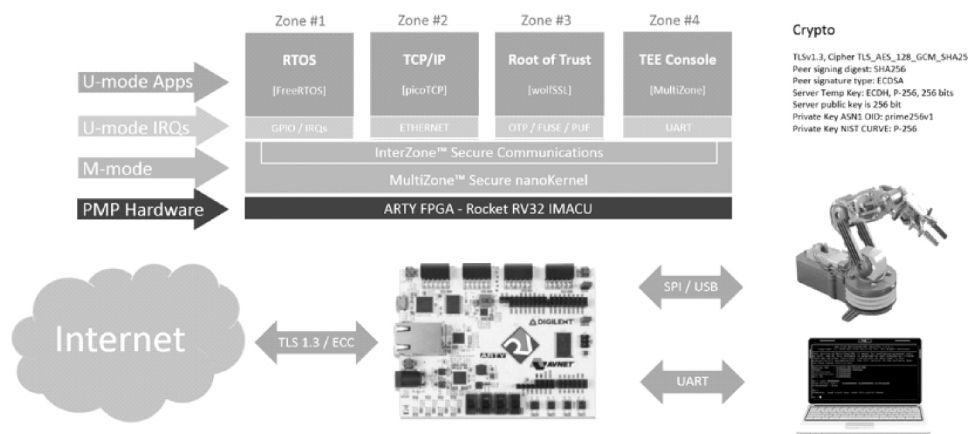


图5 MultiZone Security IoT Stack RISC-V 平台应用案例

5.2 IAR Embedded Workbench

这是一款与 KEIL MDK 在嵌入式和 MCU 市场拥有同样知名度的开发工具,针对代码尺寸和效率提供了出色的优化能力,开发者完全可以信赖这个工具来做精美的编译,分析和调试应用代码。IAR Embedded Workbench for RISC-V 最新版本是 1.30,该版本有以下特点:编译器和运行库的进一步优化,运行库中包含了对软件乘除法的实现,可以支持不带 M Extension(硬件乘法指令)的芯片。支持 P Extension(Packed SIMD)DSP 指令,支持 DSP 和 Packed SIMD specification(草案),Intrinsic functions 支持 Andes DSP libraries。进一步完善 Trace 调试功能,支持基于 Nexus IEEE-ISTO 5001 协议的 Trace 调试环境,支持 SiFive Insignia 解决方案,值得注意的是,许多 RISC-V SoC 芯片没有支持 Trace。中断向量表自动建立,无需手工编写汇编代码,支持 SiFive、Andes 和 GigaDevice 处理器核和芯片。在已经支持 Andes D25F、CloudBear BM-310、SiFive、Microchip、Syntacore 基础上支持 GigaDevice GD32V 系列 10 余款 MCU。

5.3 Segger Embedded Studio

早在 2017 Embedded Studio 开始支持 RISC-V 时,Segger 公司知名的嵌入式产品是硬件 JLINK 调试器,当然 Embedded Studio 是使用 JLINK,也可以通过 GDB 支持其他调试接口(比如 OpenOCD 和 GD-Link)。Embedded Studio 单核 RV32 指令集支持很完善,包括 RV32I、RV32IMA、RV32IMAC、RV32IMAF、RV32IMAFc、RV32G 和 RV32GC,Embedded Studio 支持芯来科技 RISC-V 处理器核(见图 6)。据悉 RV64 和多核已经在测试中。Embedded Studio 支持与主机调试器间高速通信(RTT),支持 Segger Systemview 软件分析工具。

Embedded Studio 是一个集成开发工具,Segger 官方称这个工具使用的编译器是 Clang/LLVM 和 GCC C/C++ 编译器,支持外部工具链,比如芯来科技 Embedded Studio 工程中缺省配置使用的是芯来自己优化的 GCC 工具链。Segger 创始人 Rolf Segger 在 2020 年 2 月一篇名为“The SEGGER Compiler”的博文中指出:“我们三个选择:从头开始构建,基于 GCC 构建,基于 CLANG/LLVM 构建。Clang 具有更现代的设计,并且通常被认为比 GCC 更先进。除此之外,Clang 还获得了更为宽松的许可,该许可允许制造可商购的衍生物。ARM/

Keil 从 Clang 派生了一个编译器,从而停止了其先前专有编译器的开发。考虑到所有这些,我们决定更深入地研究 Clang,以使用它创建我们自己的编译器为目标。”^[12]不难看出,Segger 编译器指的就是一个“Clang 交叉编译器”,Segger 将所有内容打包到一个易于下载的“软件包”中,完全集成并且可以立即使用。

如图 6 所示,Embedded Studio 的授权方法很友好,包括高校老师和学生在内的非商业使用都是免费的,不需要联网激活,同意使用协议后就能使用。

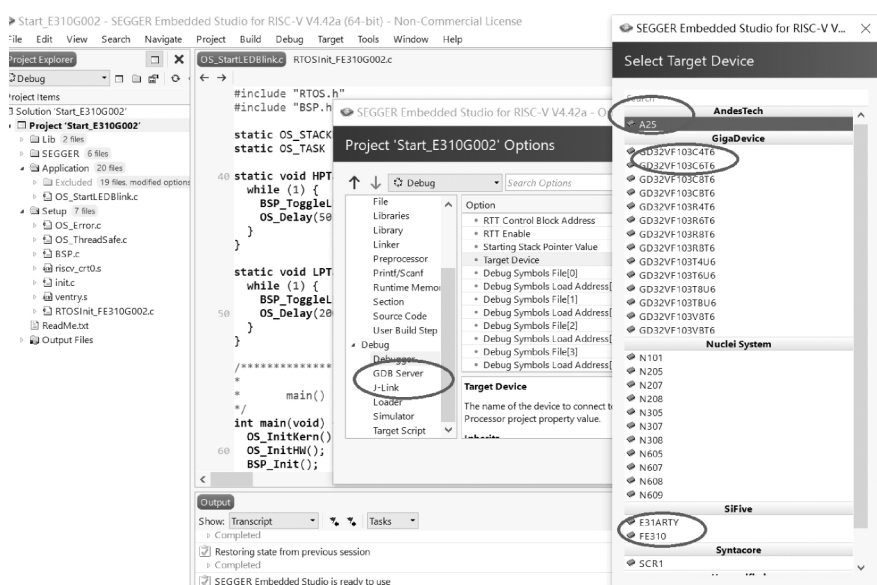


图 6 Embedded Studio for RISC-V

5.4 嵌入式操作系统

RISC-V 嵌入式处理器和物联网应用离不开嵌入式操作系统的支持,国外的 FreeRTOS、Zephyr OS、Thread X(现已被微软收购 Azure RTOS)、μC/OS、RIOT 都已经移植好的参考实例,比如:FreeRTOS 10.3 版本就有 NXP VEGA 和 SiFive freedom HiFive1-revB 开发板的移植,编译器支持 GCC 和 IAR; Segger embOS 有在 GD32VF103-Eval 开发板 embOS 和 emWin 演示; SiFive 在 Git 发布了 Amazon FreeRTOS 移植代码,目前有 SiFive Learn Inventor 教育开发板和 Andes Corvette-F1 N25 平台可运行。国内开源的 RT-Thread 有在 HiFive1-revB 演示, Huawei LiteOS 有在 GD32VF103-EVAL 展示其物联网操作系统, Sylix OS 支持 Andes RISC-V 核的开发板^[13]。

RISC-V 指令集架构要进入高端计算市场, Linux 内核的支持和进入 Linux 开源的主线(Linux tree)是至关重要的。目前在 Andes、西部数据和 SiFive 工程师的努力下有一些进展,比如 Fedora 和 Debian 在 SiFive HiFive Un-



leashed 上演示。RISC-V 是开源 Linux 开发者感兴趣的新处理器,由于缺乏经济实惠的 RISC-V 开发板,限制了开发人员在这种架构上的更多工作。Linux 内核对 RISC-V 的支持近期有所改善, Linux 5.8 版本正式支持 K210, 改变了过去 K210 只有 no-mmio Linux 支持的局面^[14], 重要的是, 社区可以很容易地找到一个 K210 (RV64GC) 的开发板。

6 总结: 教育和发展

2020 年 9 月, Imagination 宣布推出针对本科教学的 RISC-V 计算机体系结构课程, 课程名为“RVfpga: Understanding Computer Architecture”。该课程是 Imagination 与 Sarah Harris 副教授共同开发的, 她是备受欢迎的《数字设计与计算机体系结构》的合著者。Sarah Harris 说:“RISC-V 以各种可能的方式对前几代处理器进行了改善, 从功耗到性能, 还提升安全性。作为计算机体系结构的又一次巨大进步, 从基础层面上认识 RISC-V 对学生而言是很重要的。”^[15]

清华大学陈渝副教授和他的团队自 2019 年开始尝试使用 RUST 语言在 RISC-V 处理器上实现一个 rCore OS, 它是 uCore 的 RUST 移植版本。他们致力于使用现代编程语言, 提升 OS 的开发体验和质量, 探索未来 OS 的设计实现方式^[16]。笔者参与麦克泰联合 IAR 与 SiFive, 以及兆易创新和芯来科技推出了第二期和第三期嵌入式与物联网开发技术线上分享讲座课程, 课程旨在帮助开发者和高校教师、学生学习和掌握最新 RISC-V 处理器知识以及嵌入式开发技术和工具使用, 课程实验代码和科技均开放给社区^[17]。

RISC-V 的应用发展还面临许多挑战, 嵌入式系统需要的“通用和标准”开发平台, 目前市场上还非常少。RISC-V 进入高端应用, 更需要商业级开源软件, 比如 Linux 支持才刚开始, 短期很难看到 Android 可以运行在 RISC-V 处理器上。RISC-V 生态非常活跃, 但积累的成果还不足, 比如 RISC-V 中文图书只有胡振波的《手把手教你设计 CPU》一书受到广泛的欢迎。

RISC-V 产业界需要有远大的抱负和持久的耐心, 发展不仅是追求“芯片数量”增长, 还需要有数量多、门类齐全的“RISC-V Inside 产品”, 这样一来, RISC-V 开发者自然就会多了!

毫无疑问, RISC-V 非常适合用于高校电子信息相关研究项目和教育课程, 以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! 华盛顿大学计算机工

程学院 Michael Taylor 教授说:“RISC-V 没有重要的技术与实践有关的问题, 它将最终取代以 x86、ARM 为主的微处理器指令集, 从根本上改变计算的世界。”^[18]

参考文献

- [1] 胡振波. RISC-V 发展现状与应用[EB/OL]. [2020-09]. <https://mp.weixin.qq.com/s/7SBGzYSEg81vP0jL66GCKQ>.
- [2] History of RISC-V[EB/OL]. [2020-09]. <https://riscv.org/about/history/>.
- [3] DAVID PATTERSON, ANDREW WATERMAN. RISC-V 手册: 一本开源指令集的指南, 2018: 23-33.
- [4] 胡振波. RISC-V 架构与嵌入式开发快速入门[M]. 北京: 人民邮电出版社, 2018: 34-39.
- [5] RISC-V MCU 中文社区[EB/OL]. [2020-09]. <https://www.riscv-mcu.com/>.
- [6] IAR Getting started with IAR RISC-V GD32V Eval board [EB/OL]. [2020-09]. <https://www.iar.com>.
- [7] K210 技术规格书[EB/OL]. [2020-09]. <https://canaan-creative.com/developer>.
- [8] V32M1-VEGA Development Board User Guide[EB/OL]. [2020-09]. <https://github.com/open-isa-org/open-isa.org/releases/download/1.0.0/Documentation.zip>.
- [9] Artix-7 35T Arty FPGA 评估套件学习+SiFive risc-v 指令集芯片验证[EB/OL]. [2020-09]. <https://www.cnblogs.com/zjutlitao/p/9745365.html>.
- [10] MultiZone Security for RISC-V[EB/OL]. [2020-09]. <https://hex-five.com/multizone-security-sdk/>.
- [11] RISC-V Foundation. RISC-V - Getting Started P5, P14 Guide, 2019.
- [12] Rolf Segger. The SEGGER Compiler[EB/OL]. [2020-09]. <https://blog.segger.com/the-segger-compiler/>.
- [13] 何小庆. 3 种物联网操作系统分析与比较[J]. 微纳电子与智能制造杂志, 2020(3).
- [14] https://kernelnewbies.org/Linux_5.8#RISCV.
- [15] “RVfpga: Understanding Computer Architecture” includes teaching materials and hands-on exercises for students [EB/OL]. [2020-09]. <https://riscv.org/2020/09/imagination-announces-the-first-risc-v-computer-architecture-course/>.
- [16] rCore OS 开发文档[EB/OL]. [2020-09]. <https://rcore.gitbook.io/rust-os-docs/>.
- [17] 嵌入式与物联网开发技术线上讲座课程(第三期)[EB/OL]. [2020-09]. <http://hexiaoqing.net/courses/>.
- [18] Samuel Greengard Will RISC-V Revolutionize Computing? [J]. COMMUNICATIONS OF THE ACM, 2020, 63(5).

(责任编辑: 芦潇静 收稿日期: 2020-09-10)