The Rise and Popularity of RISC-V

Saroj Rout, *Senior Member, IEEE* and Chandra Shekhar

***Abstract*—RISC-V, a Reduced Instruction Set Computer (RISC) architecture has emerged as the architecture of choice for power-conscious devices such as tablets, mobile phones, and IoT embedded systems. This article looks at the evolution of RISC-V to give readers a historical perspective on its enormous success.**

***Index Terms*—RISC-V, RISC, Computer Architecture**

I. Introduction

SYSTEMS-ON-A-CHIP (SoC) with integrated processors are becoming ubiquitous. One such SoC is a microcontroller. It is hard to find an electronic system without a microcontroller in it, e.g. IoT, appliances, watches, toys, etc. They vary in complexity depending on the target applications. For example, Atmel's ATtiny series of microcontrollers are 8-bit processors starting with just 512-byte Flash and 32-byte SRAM running at 10 MHz clock speed [1]. On the other end of the spectrum are the popular STMicroelectronic STM32 series microcontrollers, which are based on Arm's Cortex-M 32-bit processor with up to 128 kB of L2 cache and more than 600 kB of SRAM running at 650 MHz [2].

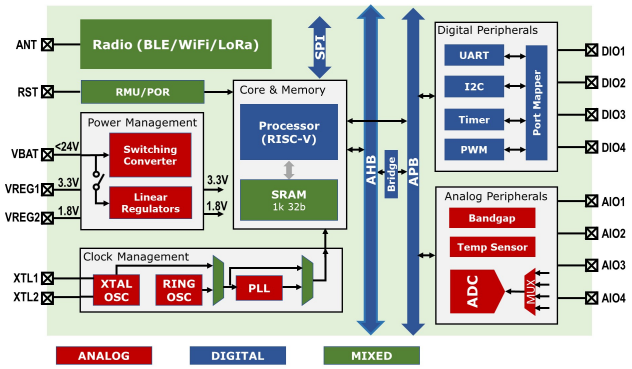

Fig. 1. Architecture of a microcontroller intended for wireless IoT nodes.

Fig. 1 shows an architecture of a microcontroller designed for IoT wireless nodes. It contains a core processor, a cache (SRAM), analog peripherals (bandgap voltage reference, temperature sensor, and analog-to-digital converter(ADC)), digital peripherals (UART, SPI, PWM), power management, clock management (PLL, crystal oscillator) and RF Radio (Bluetooth, WiFi, LoRa [3], [4]). Apart from the processor, all other blocks (typically called IPs) can be either designed from scratch by mixed-signal engineers or licensed from a vendor. When it comes to the processor, it becomes a little complicated. You can use a very old instruction set architecture (ISA) and implement it in your target technology. One such popular ISA is Intel's 8051 microcontroller architecture developed in 1980s [5]. It has been around for several decades and has proven to be reliable, efficient, and versatile. Its compact size, low power consumption, and wide range of peripherals make it a popular choice for many embedded system applications. Additionally, there is a large community of developers and engineers who are familiar with the 8051 architecture, making it easier to find support and resources for development.

Companies like Arm, IBM, and Intel have very successful ISAs, but patents on key aspects of their ISAs prevent others from using them without licenses, which imposes a prohibitive cost for most educational, research organization or companies trying to target low-volume markets. Note that ISA is not an implementation of the hardware or software (OS, drivers, compilers). It is a standardization of the interface specification, so development tools and design implementations can be reused and shared, provided that they are open and free. And both hardware and software can be implemented in three ways: proprietary, licensed open-source, or free open-source.

There are three free and open-source RISC ISAs currently available for use [6]:

1) *SPARC V8* from Sun Microsystems that was made into a IEEE standard in 1994.
2) *OpenRISC* is a GNU open-source effort started in 2000, with 64-bit ISA completed in 2011.
3) *RISC-V* (pronounced "RISC 5") a BSD open-source effort started in 2010 at the University of California at Berkeley by Krste Asanović, David A. Patterson and their graduate students Andrew Waterman and Yunsup Lee partly inspired by ARM's IP restrictions together with the lack of 64-bit addresses and overall baroqueness of ARM v7.

Before we look at the history of RISC-V and why it prospered compared to it's predecessors SPARC V8 and OpenRISC, it is worth taking a quick look at the history of microprocessors and how RISC has emerged as the goto architecture for power-conscious devices such as tablets, mobile phones, and embedded systems.

II. Brief History of Microprocessors

The birth of a microprocessor took place around the end of 1960s when a Japanese calculator maker Business Computer Corporation (Busicom) contracted with Intel, then a small startup, for custom chips for a calculator. That gave birth to Intel 4004 which is widely accepted as the first microprocessor, and that calculator was launched in early 1971 with Intel 4004 in it along with chips for storage and I/O [7]. In early 1970, Computer Terminal Corporation (CTC), a company based in San Antonio, TX, USA, arranged Intel to

Dr. Saroj Rout is with Department of Electronics Engineering, Silicon Institute of Technology, Bhubaneswar, India (https://sroutk.github.io)

Dr. Chandra Shekhar currently serves as the Professor Emeritus at Birla Institute of Technology & Science, Pilani, Rajasthan, India

build a single MOS chip to replace their discrete-based 8-bit processor for their general-purpose computer, Data Point 2200. Although Intel went ahead and built the chip based on the Data Point 2200 architecture, the project was suspended as CTC contracted Texas Instruments (TI) for the job [1]. TI developed the TMC 1795 processor for CTC which was rejected after testing the chip. Eventually, TI abandoned the project after failing to market it to other companies. In the mean time, Intel 8008 was successfully working at the end of 1971, but CTC had lost interest in that project and gave up its exclusive right to design. Intel then commercialized 8008 in April 1972, and went on to become a successful product. In 1974 the 8008 spawned the Intel 8080, which in turn heavily influenced the Intel 8086 and today's x86 architecture, which went on to dominate the personal computer (PC) and the server market [8]. So the history is certainly very interesting, where it all started with CTC's Data Point 2200 computer, which eventually established Intel as a microprocessor company and its x86 architecture as a leading architecture in the years ahead.

From 1978 until 1988, the complex instruction set computer (CISC) architecture [9] dominated the market, with performance improvement averaging at 15% per year. This improvement mainly owed to Moore's law, which ensured the doubling of transistors per unit area every 18 months, and Dennard's scaling theory to ensure that the power consumed in the chip also remained about the same.

From 1988-2003, the focus shifted to improving single-processor performance by exploiting instruction level parallelism (ILP), which led to Reduced Instruction Set Computer (RISC) architecture [10] becoming mainstream. Single processor performance improved tremendously with the use of pipelining for single-cylce execution, branch prediction, out-of-order execution, on-chip caches, multilevel on-chip caches, superscalar processors, and VLIW. The continuing benefit of Moore's law and the improvement in Dennard scaling performance was averaged @ 40 % per year. This performance enhancement was the result of the change in the RISC Instruction Set Architecture (ISA) and microarchitectures based on pipelined execution. RISC offered a simplified ISA that restricted arithmetic and logic operations to register operands as source and destination operands. Memory access was limited to two instructions, load and store, without performing any logic or arithmetic operation on the operands. Memory addressing modes were fewer and simpler compared to CISC. The RISC ISA also allowed for ILP through pipelining of instructions that gives a throughput of one instruction per cycle, in contrast to a CISC architecture taking multiple cycles for an instruction to execute. Additionally, RISC ISA also allowed for Superscalar Processing where more than one instruction can be fed to the pipeline in parallel, albeit some challenges involved in doing so.

III. The Emergence of RISC-V as Free, Open ISA

For an open-source ISA to have a long and successful lifespan, the founders of RISC-V were wise to look for a proven commercial record. This quickly eliminated some unsuccessful ones or those that have slowly died down, including the *stack* ISA, *VLIW*, Multiflow, and any new CISC ISAs. While 80x86 had dominated the PC market, *RISC-style* load-store ISAs date back to Seymour Cray's CDC 6600 and RISC had dominated the mobile and tablets of the post-PC Era. Therefore, RISC naturally emerged as a free, open ISA. The founders and the community also felt the need to start with a clean slate with a new RISC ISA to correct some of the mistakes of its predecessors. Some of the learning from the mistakes were: *leaving out essentials* such as the byte load / store or the half word load / store in the initial MIPS I, *making the ISA bulky* by including too much like the register windows in SPARC, or *microarchitectural dependent ISA* like the delayed branch in MIPS.

Given that the longevity of ISA is in decades, it was essential to look at the future technology landscape to predict some essential features of this new ISA. Three technology platforms seemed to dominate: Internet of Things (IoT), Personal Mobile Devices, and Warehouse-Scale Computers (WSCs). To target these technologies, it will be essential to have four key features from the new ISA:

- *Base-plus-extension ISA* with a small core set of instructions for cost-sensitive applications, optional extensions with standard ISA additions to customize SoCs, and space for entirely new set of instructions application-specific SoCs.
- *Compact instruction set encoding* for cost-sensitive applications such as IoT that require smaller memory.
- *Quadruple-precision (QP) as well as SP and DP* support for wide-ranging data size requirements from WSCs to IoT.
- *128-bit addressing as well as 32-bit and 64-bit.*

				Address			Software			
	Base+Ext	Compact	Quad FP	32-bit	64-bit	128-bit	GCC	LLVM	Linux	QEMU
SPARC V8			√	√			√	√	√	√
OpenRISC				√	√		√	√	√	√
RISC-V	√	√	√	√	√	√	√	√	√	√

Fig. 2. Scores for three free open ISAs using four criteria plus listing critical compilers and OS ports.

Fig.2 tabulates the scores of the three free ISAs based on four criteria plus the list of critical compilers and OS ports. RISC-V meets all the intended requirements while being simple and clean. Because of these advantages, RISC-V gained a lot of momentum among the community and emerged as a clear winner as a choice for a free and open-source ISA. In particular, Chisel [11] from Berkeley was a highly productive open hardware design system that became part of many silicon successes, making the case for RISC-V even stronger.

The rise and popularity led to the RISC-V Foundation (www.riscv.org), a nonprofit corporation that was founded in 2015 to build an open and collaborative community of

[1]An interesting historical note: Intel was a startup with about 100 employees, and Texas Instruments was a large company with more than 45,000 employees

software and hardware innovators based on the RISC-V ISA. In November 2018, the RISC-V Foundation announced a joint collaboration with the Linux Foundation to provide operational, technical, and strategic support.

The adoption of RISC-V cores in different industries is predicted to reach 62 billion cores by 2025. A study in 2020 showed almost a quarter of ASIC and FPGA projects incorporating RISC-V cores [12].

IV. Conclusion

The history thus far has clearly indicated in favor of RISC-V as a free and open ISA. It has created a great opportunity by removing barriers related to design risk, cost of entry, partner limitations, and supply chain. The RISC community took a united approach to learn from previous mistakes of open ISAs and created an ISA for longevity. Given the increasing success of RISC-V, it can be safely envisioned to be the choice of ISA for most power-conscious and those with costs, including IoT, WSC, and mobile devices.

References

[1] Microchip, "ATtiny85: 8-bit AVR RISC-based Microcontroller." [Online]. Available: https://www.microchip.com/en-us/product/attiny85

[2] STMicroelectronics, "STM32: Arm Cortex-M 32-bit Microcontrollers (MCUs)." [Online]. Available: https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html

[3] R. S. Sinha, Y. Wei, and S.-H. Hwang, "A survey on LPWA technology: LoRa and NB-IoT," *ICT Express*, vol. 3, no. 1, pp. 14–21, Mar. 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405959517300061

[4] M. Bor, J. E. Vidler, and U. Roedig, "LoRa for the Internet of Things." AUT: Junction Publishing, Feb. 2016, pp. 361–366, conference Name: EWSN '16 Proceedings of the 2016 International Conference on Embedded Wireless Systems and Networks Meeting Name: EWSN '16 Proceedings of the 2016 International Conference on Embedded Wireless Systems and Networks Num Pages: 6. [Online]. Available: https://eprints.lancs.ac.uk/id/eprint/77615/

[5] "MCS-51 : 8051 microcontroller," Dec. 2023, page Version ID: 1192578186. [Online]. Available: https://en.wikipedia.org/wiki/MCS-51

[6] K. Asanovic and D. A. Patterson, "Instruction sets should be free: The case for risc-v," 2014. [Online]. Available: https://people.eecs.berkeley.edu/ krste/papers/EECS-2014-146.pdf

[7] F. Faggin, M. Hoff, S. Mazor, and M. Shima, "The history of the 4004," *IEEE Micro*, vol. 16, no. 6, pp. 10–20, Dec. 1996, conference Name: IEEE Micro. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/546561

[8] "Intel 8080," Jan. 2024, page Version ID: 1196785311. [Online]. Available: https://en.wikipedia.org/wiki/intel_8080

[9] E. Blem, J. Menon, and K. Sankaralingam, "Power struggles: Revisiting the RISC vs. CISC debate on contemporary ARM and x86 architectures," in *2013 IEEE 19th International Symposium on High Performance Computer Architecture (HPCA)*, Feb. 2013, pp. 1–12, iSSN: 1530-0897. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6522302

[10] D. A. Patterson and C. H. Sequin, "RISC I: A reduced instruction set VLSI computer," in *25 years of the international symposia on Computer architecture (selected papers)*, 1998, pp. 216–230.

[11] J. Bachrach, H. Vo, B. Richards, Y. Lee, A. Waterman, R. Avižienis, J. Wawrzynek, and K. Asanović, "Chisel: constructing hardware in a Scala embedded language," in *Proceedings of the 49th Annual Design Automation Conference*, ser. DAC '12. New York, NY, USA: Association for Computing Machinery, Jun. 2012, pp. 1216–1225. [Online]. Available: https://doi.org/10.1145/2228360.2228584

[12] C. Redmond, "RISC-V: The Open Era of Computing," 2021. [Online]. Available: https://riscv.org/wp-content/uploads/2021/05/RISC-V-New-Era-04-19-2021.pptx