"One Student One Chip" Initiative: Learn to Build RISC-V Chips from Scratch with MOOC

Zeyu Gao[1,2] and Xiaoke Su[1]

[1]Institute of Computing Technology, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences

Abstract

The "One Student One Chip" (OSOC) initiative was launched by the University of Chinese Academy of Sciences in 2019. The initiative guides students through designing a RISC-V processor chip from scratch, including tape-out, developing a simple operating system, running it on the chip, running the real game *Legend of Sword and Fairy*, and completing the physical design process using open-source EDA tools. This enables students to understand the entire processor chip design process. As of February 2025, the total number of OSOC enrollments has exceeded 11,000. This report introduces the implementation of the "One Student One Chip" initiative and the outcomes of open-source chip talent cultivation.

Introduction

Recent advancements in RISC-V have significantly accelerated the demand for chip talents, resulting in a growing talent gap. As a result, chip companies and research institutions now require higher quality and greater quantities of professionals. University chip talent training systems must not only enhance training quality but also scale up to meet this demand.

However, the construction of the chip talent training system faces several challenges:

(1) The high cost of training chip design talents makes it unaffordable for many universities.

(2) Comprehensive processor chip design requires interdisciplinary knowledge, integrating concepts from computer science (CS) and electronics (EE).

(3) Relying solely on theoretical knowledge is insufficient; hands-on practice is crucial.

(4) Processor chip design spans multiple levels of abstraction, from applications to transistors, resulting in a steep learning curve.

To tackle these challenges, we introduced the "One Student One Chip" (OSOC) initiative, aiming to provide hands-on experience and bridge the gap between theoretical knowledge and practical application.

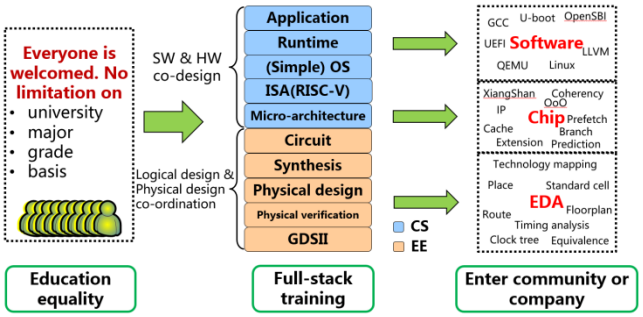

Figure 1 Overview of the "One Student One Chip" Initiative

As shown in Figure 1, the OSOC initiative is a large-scale, practice-oriented open learning program based on open-source components. It guides students through the design of a tape-out-capable RISC-V processor chip from scratch. Participants develop a simple operating system and run the real game *Legend of Sword and Fairy* on their chips, while completing the physical design process using open-source EDA tools. The initiative's primary goal is to provide students with hands-on experience in the entire processor chip design process.

The OSOC initiative is open to all individuals, including students and graduates, and offers free access to learning materials. Additionally, students who complete the program are provided with free tape-out opportunities.

To date, the initiative has completed six seasons, attracting over 10,000 participants and training hundreds of processor chip design talents. It has also contributed to the development of a new model for chip talent training.

Methodologies

To address the challenges in chip talent development, we employed the following strategies:

(1) Leveraging open-source components to lower the entry barrier for chip talent development. This involves designing processors based on RISC-V, integrating open-source IPs, and utilizing open-source EDA tools. Students are also encouraged to contribute back to the open-source community.

(2) To bridge computer science (CS) and electronic information engineering (EE) courses, we designed a learning schedule, as shown in Figure 2.

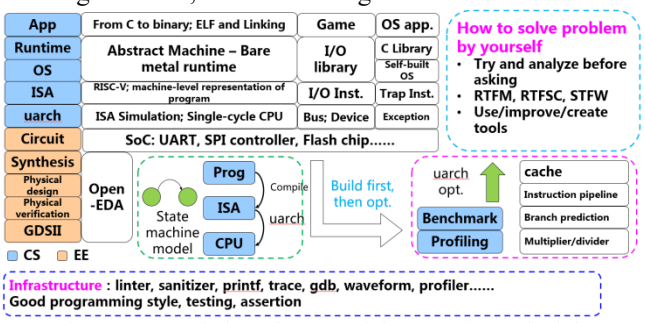

Figure 2 Knowledge Diagram of the Learning Schedule

Methodologies

To address the challenges in chip talent development, we employed the following strategies:

(1) Leveraging open-source components to lower the entry barrier for chip talent development. This involves designing processors based on RISC-V, integrating open-source IPs, and utilizing open-source EDA tools. Students are also encouraged to contribute back to the open-source community.

(2) To bridge computer science (CS) and electronic information engineering (EE) courses, we designed a learning schedule, as shown in Figure 2.

This schedule also aims to connect software and hardware, helping students gain a deep understanding of how computer systems support application execution through processors, instruction sets, runtime environments, operating systems, and library functions. Additionally, it fosters the ability to design processor chips across the entire flow, from architecture design, RTL development, and SoC integration to verification, synthesis, physical design, and ultimately generating GDSII for chip manufacturing.

(3) By combining theory and practice, we incorporate cutting-edge research achievements in processor chip design into the teaching materials. This includes agile development languages, agile testing methods, and formal verification techniques. These approaches enable students to quickly identify bugs during design practices and analyze performance bottlenecks across the entire system.

(4) The learning schedule is divided into several stages: preliminary, basic, advanced, and specialist. The preliminary stage helps students with limited prior knowledge transition smoothly to the subsequent stages, reducing the learning curve. Additionally, each stage uses games as test targets for processor chip design, fully engaging students' learning interests.

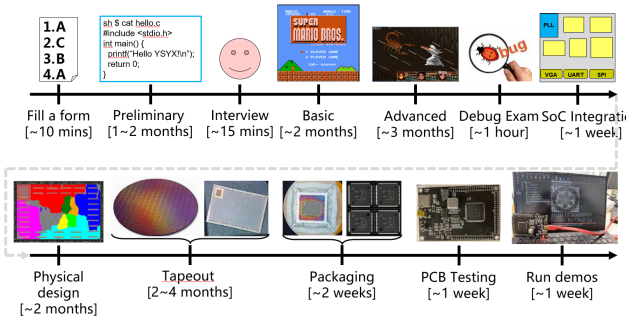

Figure 3 Learning Roadmap

Figure 3 illustrates the learning roadmap. To participate in the initiative, students first fill out an application form. We do not reject any applications, ensuring everyone has a chance to learn. After applying, students enter the preliminary stage, where they learn Linux basics, C language, digital circuits, and HDL. Following the preliminary stage, students undergo an interview with TAs to assess their understanding. If needed, TAs provide suggestions, and students can retake the interview after further study.

The next phase is the basic stage, where students design a single-cycle RISC-V processor and run Super Mario. The advanced stage follows, with the goal of designing a pipelined RISC-V processor that supports caches and the AXI bus, while running a simple self-developed operating system and even a real game.

The specialist stage (not shown in Figure 3) focuses on implementing MMU, booting Linux, and optimizing the processor. Currently, this stage is optional. After completing the learning process, students can apply for a debug exam. Upon passing, they can integrate their processor into an SoC, sharing the same SoC and devices with others. Students can also opt to complete physical design using open-source EDA tools, though our team currently uses commercial EDA tools to generate GDSII and tape-out. Finally, students will receive a PCB board with their own chip.

Result

To date, the OSOC initiative has completed six seasons, receiving over 11,000 applications. More than 1,000 students have passed the interview, and over 100 have completed the program and had their chips taped-out. Figure 4 shows some demos of students' chips. A few students have also completed the specialist stage and successfully booted Linux Debian on their chips.

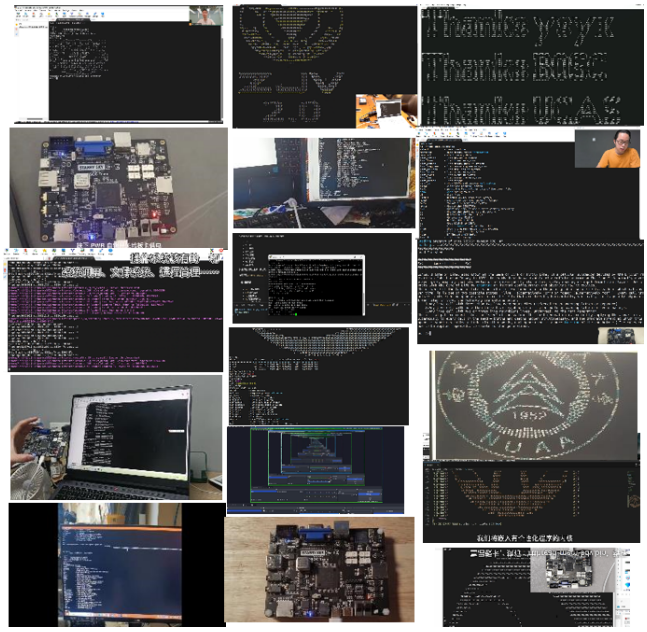

Figure 4 Demos from Students and their Chips

Indeed, the real chip is a byproduct of the initiative, with the primary focus being the growth of students during the learning process. Feedback from students highlights their ability to manage more complex projects, solve new problems independently, and gain increased confidence and patience. For example, one student shared that the initiative's comprehensive training program helped him develop a solid understanding of processors and computer systems, enabling him to quickly engage with another research project. Moreover, when facing challenges in the project, he was able to efficiently solve problems by applying the practical experience gained from the initiative.