A Unified Digital Twin Platform for the Experiment Education of "The 101 Plans" Hardware Courses

Yu Huang[0009−0009−5865−8714], Zonghui Li*, Ke Xiong, and Ziyao Shen

School of Computer Science and Technology, Beijing Jiaotong University, China
ketted@bjtu.edu.cn, lizonghui@bjtu.edu.cn, kxiong@bjtu.edu.cn, 23331192@bjtu.edu.cn

Abstract. This paper presents a Digital Twin-based teaching platform that supports the main hardware courses for undergraduate education. It is part of "The 101 Plans" initiative, which aims to unify and improve computer hardware experimental courses. We develop a Digital Twin platform and a series of RISC-V-based courses on computer organization and system architecture. The platform supports hybrid teaching with physical experiment board and a Digital Twin client when the physical boards are not in hands. This setup overcomes equipment constraints, allowing for centralized hardware management and improved maintenance and efficiency. The platform can support hardware courses from digital circuits to CPU design and CPU performance optimization and finally covers System-on-Chip (SoC) development and operation system booting. We have written a series of detailed experimental guides for these courses. It combines clear instructional guidance with carefully designed problems. Some engineering challenges are intentionally required to encourage independent problem-solving and deeper hands-on engagement. The platform has been granted three patents and serves as the official competition platform for the RISC-V Cup track of the 2025 China College IC Competition. A pilot run of the Computer Organization course at Beijing Jiaotong University this year received positive student feedback and demonstrated effective learning outcomes.

Keywords: Hardware Education · RISC-V · Digital Twin · Online-Offline Hybrid Teaching · Computer Organization.

* Corresponding author

1 Introduction

"The 101 Plans"[1] is a pilot initiative launched by the Chinese Ministry of Education to reform undergraduate computer science education. As part of efforts to develop first-class core courses that are advanced, innovative, and challenging, experimental teaching plays a vital role in strengthening students practical abilities in computer systems. However, compared to software-focused courses, hardware experiments often present greater challenges, from complex curriculum design to the high cost of hardware. Therefore, hardware courses have long been regarded as a weak area in the cultivation of system-level capabilities in undergraduate computer education.

In recent years, the open-source RISC-V instruction set architecture has emerged as a game-changer in computer architecture education and research. As an open standard, anyone can use RISC-V as a building block in their open or proprietary solutions and services [2]. With its modular design, high scalability, and open-access nature, RISC-V offers an ideal foundation for hardware experiment design.

In response to current challenges in experimental teaching, we have developed both a Digital Twin platform and a corresponding series of courses on computer organization and system architecture based on the RISC-V architecture.

To support both in-person and remote teaching, the platform combines real hardware boards for offline use with Digital Twin software that collects hardware data and presents hardware behavior. Even when students do not have access to physical devices, they can still complete experiments in a nearly identical environment. This setup helps reduce equipment limitations and makes it easier to manage and maintain the hardware centrally.

A structured and progressive practice pathway is embedded throughout the curriculum, beginning with digital logic and hardware programming in Digital Circuit, moving to CPU architecture design in Computer Organization, exploring performance optimization in Computer Architecture, and culminating in operating system-level experiments in SoC Design with OS Integration. This integrated approach enables students to understand the principles of hardware-software co-design and enhances their practical skills in hardware development and system implementation.

Recognizing the critical role of the Computer Organization course, we piloted its reformed version at our institution in the 2025 spring semester. During the process, the course and platform were well adopted by students.

The rest of the paper is organized as follows. Section 2 presents motivation and related works. Section 3 presents an overview of the proposed teaching platform and course system. Section 4 presents the design and implementation of the platform, including its architecture and core systems. Section 5 introduces the structure and content of our proposed hardware experimental curriculum. The computer organization course is used as a representative case study to demonstrate the course design in practice. Section 6 introduces platform deployment and educational impact. Finally, we conclude our work in Section 7.

2 Motivation and Related Work

Although engineering-oriented education and hands-on learning are gaining more attention in China, several long-standing issues remain in the teaching of computer system courses. These challenges can be broadly grouped into three areas:

1) Lack of integration: Many courses are taught separately, without a clear link between them. Students often struggle to see the connection between the content of one course and another.

2) Overemphasis on theory: Teaching often focuses too much on theory and explaining fundamental concepts but not enough on completing real projects [3,7]. Students typically learn through slides, equations, and simple examples rather than working with real hardware or writing programs for actual systems.

3) Weak self-research: In many courses, students follow instructions to complete small lab tasks on pre-made platforms. While many labs help them understand basic ideas, they do not give students the chance to design and build a system from scratch. There are a few open-ended tasks where students can try their ideas, explore different solutions, and learn how to solve real problems[4].

To better understand and address these issues, we reviewed recent efforts by researchers from leading Chinese universities. Yuan et al.[5] proposed a project-based curriculum on high-performance processor design with I/O and exception handling, but its complexity challenges students with weaker foundations. Qin et al.[8] proposed using a unified FPGA platform to connect multiple courses, though specific implementation practices and results were not well documented. Zhang et al.[6] proposed MIPS-based experiments and developed an intelligent evaluation platform, though details on implementation and outcomes were limited.

Building on insights from previous work, we identified three key requirements for a more effective teaching model. First, the curriculum should be well-integrated across related courses, guiding students progressively from basic digital logic to complete system design. Second, we must consider the cost and complexity of managing hardware resources, which often make it challenging to ensure that every student has access to hands-on use of physical circuit boards[9]. Third, the courses should emphasize meaningful, engineering-driven tasks that go beyond step-by-step instructions and encourage students to tackle real-world design problems.

To address these challenges, we developed a Digital Twin platform that lowers hardware costs by reducing the need for physical FPGA boards. Students can complete hardware programming in a virtual environment with behavior close to real devices. Based on this, we designed a series of structured courses focused on computer organization and system architecture, combining guided tasks with open-ended problems to strengthen practical skills and independent thinking.

3 Overview of the Platform and Courses

We developed four hardware-focused courses namely, Digital Circuit, Computer Organization, Computer Architecture, and SoC Design with OS Integration,

that follow the full hardware practice path of "The 101 Plans". These courses guide students from basic logic to CPU design and building a SoC system with RT-Thread OS. To support flexible learning, we provide a hybrid environment that includes a physical FPGA board and a Digital Twin client software. Digital Twin platform is composed of four main modules: the Client, the Remote Control System (RCS), the Data Communication System (DCS), and the Student Experiment System (SES). Fig. 1 illustrates the overall structure, featuring the progression through four courses alongside the hybrid hardware course Platform. Technical details are provided in Sections 4 and 5.

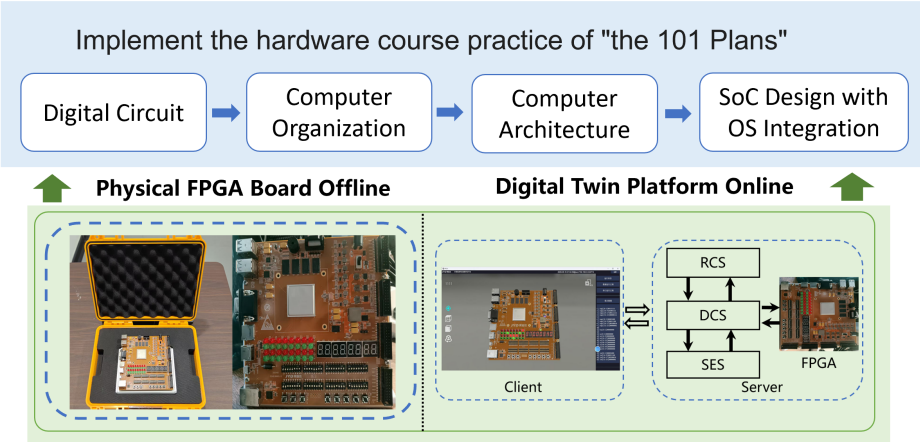

Fig. 1. Overview of the Hybrid Hardware Course Platform for "The 101 Plans"

4 Architecture of Digital Twin Platform

The Digital Twin platform enables students without physical FPGA access to complete experiments remotely while maintaining familiar interaction patterns. It includes four modules: Client, RCS, DCS, and SES, as previously mentioned above. The Client provides an interface similar to a physical FPGA board and connects to the RCS via the internet to retrieve FPGA status. Students write hardware description language (HDL) code in the SES, which, together with the DCS, is used to program the FPGA. The DCS communicates with the RCS through a serial connection to keep the FPGA synchronized. By operating the Client, students can remotely control and interact with the FPGA in real-time. Fig. 2 illustrates the overall architecture of the Digital Twin platform, highlighting the interactions among three subsystems, the Client and the FPGA.

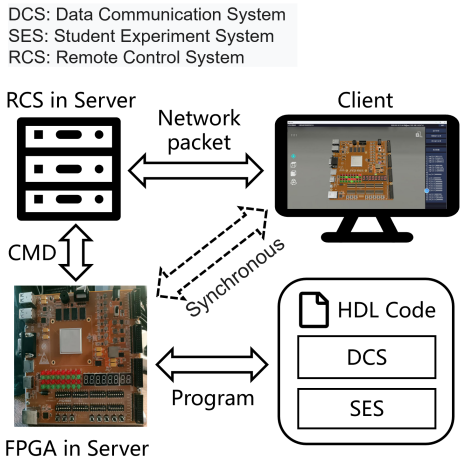

Fig. 2. Architecture of Digital Twin Platform

4.1 Design of Remote Control System

RCS sends control commands to DCS programmed into the FPGA via serial. To maintain stable control and synchronize data, the system employs periodic scheduling to manage the sending of commands and the receipt of responses. This timing prevents response errors and resource conflicts. Meanwhile, a status check automatically identifies and resolves any discrepancies between the client and FPGA in the server, ensuring accurate operations and consistent platform behavior.

4.2 Design of Data Communication System

DCS uses UART (8N1) and a simple command set to exchange control and status data with the RCS, ensuring stable transmission through dejittering and timed sampling.

To unify the board's complex pin layout, all 144 I/O pins are mapped into four peripheral groups: 64 DIP switches, 8 buttons, 32 LEDs, and 8 seven-segment displays. Switches and buttons are virtualized as writable inputs, while LEDs and seven-segment displays are mapped to real FPGA outputs and updated in real-time.

Two commands suffice. READ (opcode 0x80) returns all peripheral states in 18-byte. WRITE uses 1 byte to set a virtual input: the MSB indicates value, and the lower 7-bit selects one of 72 input addresses (0x01-0x40 for switches, 0x41-0x48 for buttons). As shown in Fig. 3, both commands fit in a single UART frame with no handshaking, and invalid opcodes are ignored.

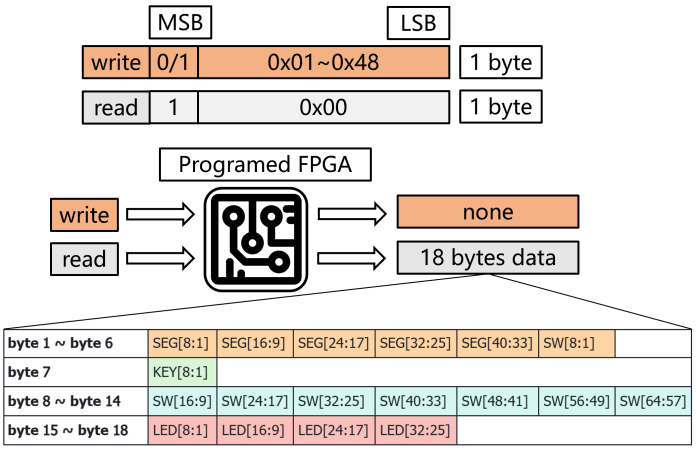

Fig. 3. WRITE & READ Command Structure in DCS

4.3 Design of Student Experiment System

SES provides a standardized programming interface for students to complete their designs, which includes several internal modules, as shown in Fig. 4. Students can only interact with the student module, which encapsulates internal complexity and streamlines the process of development.

The student module provides a unified interface with clock, reset, and standard I/O ports (LEDs, displays, switches, buttons), using consistent signal naming to simplify development. Inputs are virtualized for remote use, while outputs remain connected to the FPGA for real-time feedback. Students integrate their designs into Student_top.sv to complete their design.

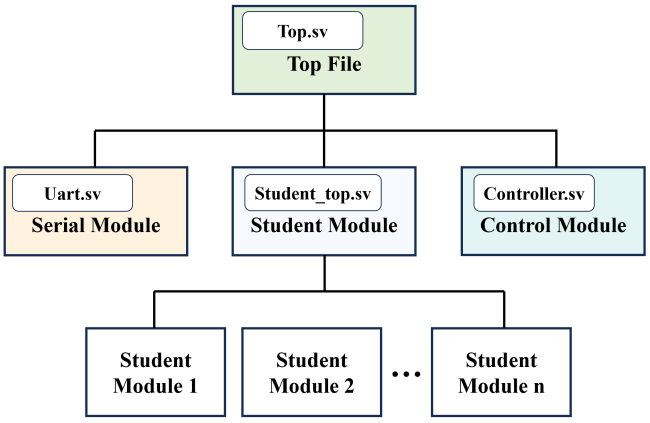

Fig. 4. Module Hierarchy of SES

5 A Case Study of Computer Organization in Four Integrated Hardware Courses

To address the problems of fragmented courses, overemphasis on theory, and weak independent research, we designed a clear, step-by-step practical course framework that connects separate course units into a closed teaching loop.

The curriculum guides students from digital circuit basics to advanced system-level design. It begins with foundational hardware programming, followed by designing a single-cycle CPU implementing the RV32I instruction set verified on hardware. Next, students enhance performance with pipelining, branch prediction, and caching. Finally, they integrate a floating-point unit, interrupt handling, AXI bus, and DMA to build a complete SoC and run an OS on their custom processor. This approach emphasizes hands-on engineering over traditional textbook methods. Table 1 summarizes all experimental designs and details of the four courses.

As an important example among these courses, the Computer Organization course was newly launched at Beijing Jiaotong University in the 2025 spring semester and serves as a case study in the paper. The experimental component of the course is divided into three stages.

Stage 1: Learning hardware basics and RISC-V ISA;

Stage 2: Implementing a CPU with a minimal 7-instruction RISC-V subset;

Stage 3: Expanding to a 24-instruction subset of RV32I (MiniRV) validated by Trace testing and hardware deployment.

5.1 Stage 1: Basic knowledge

Stage 1 aims to equip students with the fundamental knowledge and tools of hardware design. The "Hardware Programming Basics" module introduces SystemVerilog, Vivado 2023.2, and FPGA usage. Student learning is assessed through quizzes and programming exercises. (e.g., Fig. 5). Before designing the CPU, students complete a "RISC-V Assembly Programming" lab to understand the instruction set, preparing them for software hardware co-debugging. Experiments provide clear guidance with questions to stimulate thinking. Lab reports are managed via the Feishu platform (similar to GitBook) with step-by-step instructions. Some engineering details are omitted intentionally to encourage independent project building and skill development.

5.2 Stage 2: Design and verification of a core instruction set

After completing the foundational modules, students move on to designing a simplified RISC-V CPU based on a minimal set of seven core instructions. This stage emphasizes a bottom-up approach to building essential CPU components, including:

1. **Arithmetic unit design:** Modeling basic operations and extending to components like multipliers;

Table 1. Table of Four Integrated Hierarchical Hardware Courses

Course	**Lecture Topic**	**Main Experiment**
Digital Circuit	Basics of Hardware Description Languages	Basic Syntax of SystemVerilog
		Vivado Usage
		FPGA Development Kit
	Digital Systems Experiment	Logic Gate
		Combinational Logic Circuits
		Sequential Logic Circuits
	Comprehensive Experiment	Intelligent Quiz Buzzer
		Digital Clock
Computer Organization	Arithmetic Logic Unit(ALU)	Two's Complement Arithmetic
		ALU Implementation
	Memory	Usage of Block RAM in Vivado
		Bit/Word Extension
	Datapath in RV32I CPU	Instruction Analysis
		Datapath Component Design
	CPU Controller	μ-operation Analysis
		Controller Design
	Validate Design On FPGA	Bus and I/O Implementation
		RV32I Assembly Verification
Computer Architecture	Single-Cycle RV32I CPU	ISA Introduction
		Experimental Environment Setup
		Implementation of RV32I CPU
	Pipeline Experiment	Classic Five-Stage Pipeline
		Hazard Handling
		FPGA On-Board Verification
	Optimization Techniques	Branch Prediction
		Out-of-Order(Optional)
		FPGA On-Board Verification
	Design of Cache	Mapping Rules
		Replacement Policies
		Performance Evaluation
SoC Design with OS Integration	RISC-V ISA	Basic Concepts of ISA
		Basic Concepts of Architecture
		Evaluation Methods
	Floating-Point Unit(FPU)	Fixed-Point Addition
		Floating-Point Addition
		FPU Implementation
	Interrupt	CSR Register
		Core Local Interruptor System
		Implementation
	AXI Bus and DMA	AXI Protocol Overview
		an AXI-Based DMA Controller
		Connection to DDR Memory
	OS Integration	Cross-Compilation Environment
		Compilation of RT-Thread Nano
		On-Board Verification

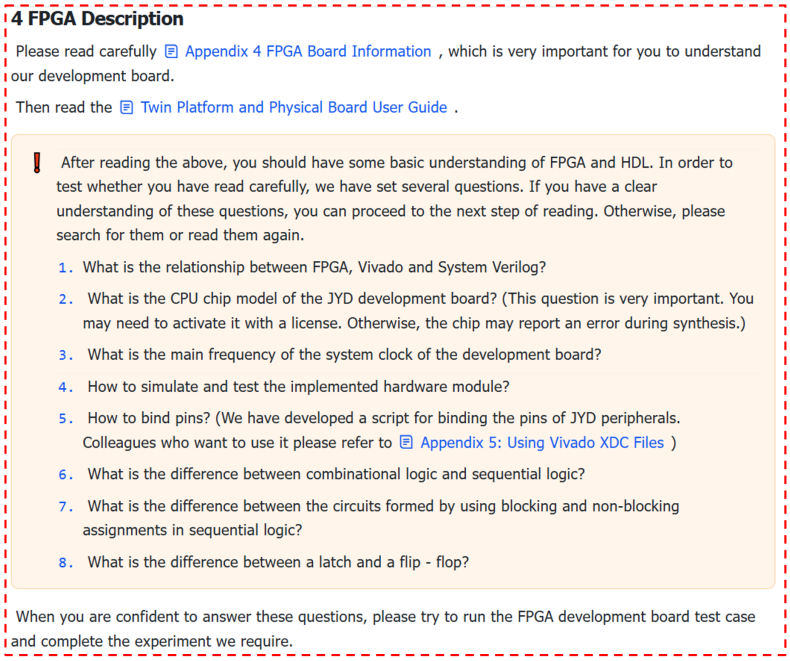

Fig. 5. Some Guiding Questions in Stage 1

2. **Data path design:** Learn the interconnections and data flow control between different modules to build a simplified but functional CPU;
3. **Memory module expansion:** Use horizontal expansion and vertical expansion to achieve larger storage capacity;
4. **Controller design:** Implementing the controller based on instruction analysis;
5. **Single-cycle CPU integration test:** Integrate the above modules to build a complete processor system and pass the basic integration test code.

5.3 Stage 3: Implementation of MiniRV and on-board verification

In this stage, students expand their CPU to support the 24-instruction MiniRV subset, encountering more complex datapath and control requirements. They learn to adjust the processor structure to accommodate extended instruction semantics.

For verification, a trace test system is employed to compare key signals of the tested module against a golden reference at each clock cycle, enabling precise error detection. Upon passing the test, students deploy their CPU to an FPGA board, analyze the I/O address space, and correctly place the test program in memory to complete on-board execution.

Fig. 6 illustrates the overall setup: the left half depicts the architecture of the trace test system. In contrast, the right half shows the Digital Twin client interfacing with the physical FPGA board during program execution.

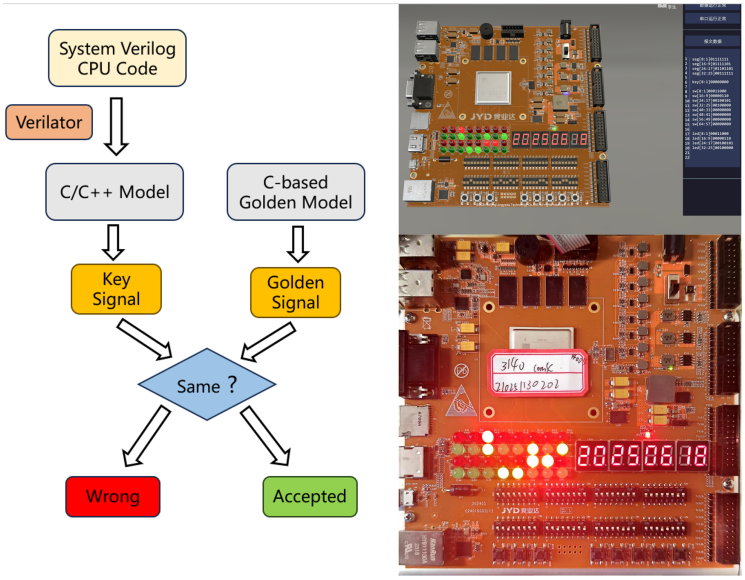

Fig. 6. Trace Test and Digital Twin System Overview

6 Platform Deployment and Educational Impact

The proposed experimental platform has demonstrated both technical and educational value. It has been granted three educational patents [10,11,12], and was exhibited at the 33rd Beijing Education Equipment Expo and the 61st China Higher Education Expo. In collaboration with JinYeda Corporation, the platform has been commercialized and adopted as the official platform in the RISC-V Cup track of the 2025 China College IC Competition [13]. With the support of this platform and curriculum system, Beijing Jiaotong University also participated in drafting the RVEI Committee's talent development framework [14].

As part of the pilot implementation, the Computer Organization course was launched in the "Zhan Tianyou Class" (70 students) this year. Students completed the whole design and implementation of a 32-bit RISC-V CPU. According to the final assessment results shown in Table 2, the average scores across all key experiments exceeded the preset target scores, with achievement ratio of 1.23 (equivalent to 85.75 out of 100). The course outcomes met the expected targets, demonstrating the effectiveness of the proposed approach.

Table 2. Course Experiment Evaluation Table

Experiment	Target Score	Average Score	Achievement Ratio
Arithmetic Logic Unit	14	18.07	1.29
Memory	14	17.77	1.26
Datapath in RV32I CPU	14	16.94	1.21
CPU Controller	14	16.79	1.20
Validate Design On FPGA	14	16.20	1.16
Total Score	70	85.75	1.23

7 Conclusions

The paper presents an integrated solution to challenges in hardware-oriented computer system education, such as course fragmentation, limited hands-on experience, and insufficient opportunities for independent design. A significant contribution is the construction of a layered and unified experimental curriculum, which connects four core hardware courses Digital Circuit, Computer Organization, Computer Architecture, and SoC Design with OS Integration through the platform. This structure enables a smooth progression from digital logic fundamentals to complete SoC design, including OS booting, addressing the fragmentation common in traditional curricula.

The platform is powered by a Digital Twin system that extends traditional offline labs to support real-time remote interaction with FPGA-based hardware. This approach provides flexible, scalable, and location-independent access to experimental resources while enabling centralized hardware management for enhanced reliability and maintenance.

As a representative plot, the newly restructured Computer Organization course demonstrated the effectiveness of this approach. Students completed a 32-bit RISC-V CPU, exceeding the expected learning goals. The platform has been granted three educational patents, has been adopted in national competitions, and has contributed to the RVEI Committees talent development framework.

Acknowledgments. This work was supported in part by the Undergraduate Teaching Reform Project under Grant LX20250081, the Postgraduate Teaching Reform Project under Grant YJSSQ20250095.

References

1. MOE, MOE holds meeting to review progress on "The 101 Plans", 22 April 2024. [Online]. Available: http://en.moe.gov.cn/news/press_releases/202404/t20240422_1127028.html [Accessed: 17 June 2025].
2. RISC-V International, About RISC-V, [Online]. Available: https://riscv.org/about/ [Accessed: 18 June 2025].
3. J. Kojmane and A. Aboutajeddine, "Enjoyeering Junior: A hands-on activity to enhance technological learning in an engineering dynamics course," 2016 International Conference on Information Technology for Organizations Development (IT4OD), Fez, Morocco, 2016, pp. 1-6, https://doi.org/10.1109/IT4OD.2016.7479267.
4. Matias Lopez-Rosenfeld, "Tell Me and I Forget, Teach Me and I May Remember, Involve Me and I Learn: Changing the Approach of Teaching Computer Organization," IEEE/ACM 1st International Workshop on Software Engineering Curricula for Millennials (SECM), Rio de Janeiro , Brazil, 2017, pp 68-71, https://doi.org/10.1109/SECM.2017.9.
5. Yuan, C.F., Tao, X.P., Wang, L., Gu, R., Li, J. (2018). Construction on curriculum experiment system for ability training of computer system. Experimental Technology and Management, 35 (06), 12-16. https://doi.org/10.16791/j.cnki.sjg.2018.06.004.
6. Qin, G.F., Ding, Z.J., Wang, L.S., Wei, Z.H., Zhang, D.D. (2020).Focusing on capacity building and continuously promoting experimental linkage. Experimental Technology and Management,37(05),142-145. https://doi.org/10.16791/j.cnki.sjg.2020.05.030.
7. S. J. Lee, A. Jung, J. Park and M. Yun, "Cost-efficient Hands-on Learning Design for Computer Organization Course," 2020 15th International Conference on Computer Science & Education (ICCSE), Delft, Netherlands, 2020, pp. 150-155, https://doi.org/10.1109/ICCSE49874.2020.9201854.
8. Zhang, L., Wang, J.P., Zheng, R., He, J., Qi, Y. (2021).Through experiment model of Computer organization course based on VerilogHDL intelligent evaluation platform. Experimental Technology and Management,38(03),236-241. https://doi.org/10.16791/j.cnki.sjg.2021.03.048.
9. P. P. Papazoglou, "Hardware Oriented Microprocessor Simulator," 2024 IEEE Global Engineering Education Conference (EDUCON), Kos Island, Greece, 2024, pp. 1-5, https://doi.org/10.1109/EDUCON60312.2024.10578582.
10. Zonghui Li et al., A computer hardware teaching platform for talent cultivation in universities, Chinese Patent CN202410450390.6, issued Jun. 4, 2024.
11. Zonghui Li et al., A computer hardware experimental box for talent cultivation in universities, Chinese Patent CN202420773756.9, issued Apr. 15, 2024.
12. Zonghui Li et al., An IoT teaching experimental platform for universities, Chinese Patent CN202410450411.4, issued May 31, 2024.
13. CICIEC, The topic of the 9th JiChuang Cup Competition - Jingyeda Cup, [Online]. Available: http://univ.ciciec.com/nd.jsp?id=879#_jcp=1 [Accessed: 18 June 2025].
14. RVEI Committee, Notice on Public Solicitation of Participating Units for the Drafting of the "RISC-V Talent Training Standard", [Online]. Available: https://mp.weixin.qq.com/s/0rNhrGCGUDhzqut7Q_cdTw [Accessed: 18 June 2025].