



Methods and Tools Used in Education for Autonomous and Collaborative Development and Simulation of Embedded Digital Systems

Michael Dolinsky^{1,*}, , 

¹*Faculty of Mathematics and Technologies of Programming, Francisk Skorina Gomel State University, Gomel, Belarus*

Article History

Received: 11 August, 2026

Revised: 21 September, 2026

Accepted: 23 September, 2026

Published: 29 September, 2026

Abstract:

This article provides a literature review on methods and tools for autonomous and collaborative development and simulation of hardware and software for embedded digital systems that are used in real university practice. Particular attention is paid to such issues as: options for the content and methods of training; the importance of software and hardware training tools; tools for teaching digital logic design; tools for teaching programming of microprocessors / microcontrollers; tools for teaching collaborative development of software and hardware of embedded systems. An important area of the research is the development and implementation of online training systems. It is noted that the experience of using generative artificial intelligence to create formal models of digital systems in hardware description languages and program texts for microcontrollers/microprocessors in the following languages is rapidly developing.

Keywords: Literature review, embedded digital systems, software, hardware, microprocessor, teaching and learning.

1. INTRODUCTION

At present, there is a significant gap between the requirements for knowledge, skills and abilities of specialists working in the field of hardware and software development for embedded digital systems and the knowledge, skills and abilities possessed by university graduates. Another problem is the significant dropout of students who are not able to cope with the requirements of the universities themselves for graduates of these specialties. The solution to these problems is seen in modernizing the content of academic subjects, changing teaching methods, and improving the technological tools used in training.

This article provides a literature review on methods and tools for autonomous and collaborative development and simulation of hardware and software for embedded digital systems used in real university practice. Familiarization with such a review can help university specialists make a choice of content, methods and teaching aids that best suit the practice of a particular university/teacher. Since 2022 author subscribed on Google Scholar newsletter on themes “basics of digital electronics”, “programming of microcontrollers”, “teach programming”. So, author have seen letters about thousands of papers from 250+ SCOPUS-indexed journals.

Particular attention in this article is paid to such issues as: options for content and methods of training; the importance of software and hardware training tools; digital logic design training tools; microprocessor/microcontroller programming training tools; training tools for collaborative development of software and hardware for embedded systems. The introduction of new information technologies into educational practice makes the development of online learning systems an important area for research.

The rapid development of large language models and generative artificial intelligence tools based on them leads to the use of generative artificial intelligence to create formal models of digital systems in hardware description languages and program texts for microcontrollers/microprocessors in assembly, C and Python.

2. OPTIONS FOR CONTENT AND TEACHING METHODS

Current issues of digital systems include the following topics [1]: new accelerator architectures, high-performance embedded systems, multi-core systems/architectures, embedded reconfigurable processors, instrumentation software, compilation optimization, code generation for reconfigurable

*Address correspondence to this author at Faculty of Mathematics and Technologies of Programming, Francisk Skorina Gomel State University, Gomel, Belarus; E-mail: dolinsky@gsu.by



© 2026 Copyright by the Author.

Licensed as an open access article using a [CC BY 4.0 license](https://creativecommons.org/licenses/by/4.0/).

architectures, architecture synthesis based on functional language descriptions, embedded parallel systems and multiprocessors on a chip, in-memory/near-memory processing, software-defined network on chips.

As part of the 2025 IEEE Kirchhoff Award, a special session was held devoted to the development of the discipline “Embedded Systems” [2].

Below is an overview of the content and teaching methods in various universities around the world in disciplines related to the issues of digital systems.

The course “Digital IC Design” is taught at Phetchaburi Rajabhat University (Thailand) [3]. The training materials include: a textbook, a toolkit, a simulator, workbooks, and assessment tests. The following topics are covered in this course: number systems and coding, logical operations and logical gates, Boolean algebra, logical minimization, combinational circuits, memory elements, circuits with memory, designing digital devices using CPLD and FPGA, VHDL language, modeling and simulation. The Melon S3 board containing an FPGA (500,000 gates), 360Kb of memory, an ESP8266 microcontroller, toggle switches, indicators, communication with a computer, and software with VHDL/Verilog support is used in the training process.

The course “Fundamentals of Digital Logic” is taught at Huazhong Agricultural University, Wuhan, China [4]. The authors set themselves the goals of teaching students in the following areas: theoretical foundations of digital logic circuits; computer aided design technology; basic syntax of the Verilog language for describing digital systems; methods for developing combinational circuits and circuits with memory; building circuits using design systems.

The article [5] analyzes the practical plan of the theoretical course “Computer Organization and Architecture” (Beijing University of Posts and Telecommunications, China) from the perspective of teaching content and methods, the platform used, in order to improve students’ understanding of the internal logic of computers. It integrates knowledge of the topics “Digital Logic and Digital Microcircuits”, “Principles of Computer Organization”, “Computer Architecture”. The course includes the completion of six practical tasks: combination schemes; memory circuits; data paths in a computer; microprogramming; assembler; hardware description language.

The teaching methodology is based on the pattern “Give priority to practice and integrate the virtual with the real.” The course “Microcontroller Technology” is taught at Guangdong University of Science and Technology, Dongguan, China [6]. In response to students’ difficulties in completing the course “Microcontroller Technology”, the teaching is based on the method OBE-ADDIE (Outcomes-based Education & Analysis – Design – Development – Implementation - Evaluation). The course “Microcontroller technologies” is based on the study and application of the STM32 microcontroller. The ADDIE method includes three main aspects: “What to teach”; “How to teach”; “How to evaluate”.

Updated course objectives - students should be able to:

1. Describe and analyze the knowledge required to develop embedded systems based on ARM processors, must be familiar with the basic structure and external contacts of various models of STM32 series microcontrollers.

2. Use debugging tools for STM32 processors, together with debug boards.

3. Analyze design requirements and utilize internal microcontroller resources such as external input and output pins, system interrupts, timers, *etc.*

Superficial understanding and declining interest in subjects such as computer architecture require a change in teaching methods, so at University Tun Hussein Onn (Malaysia), a Verilog model of a RISC processor is being developed for educational purposes within the disciplines of Computer Architecture and Microprocessors and Microcontrollers [7]. Processor is 8-bit, executes 16 types of instructions, designed using Intel Quartus and Verilog HDL. Simulation and verification are performed using ModelSim (Altera).

The digital systems development training at the Bundeswehr University (Munich, Germany) [8] includes courses such as “Digital Circuit Design”, “Digital Systems Design”, “Systems on a Chip”, Python. The first two courses are studied in the bachelor’s program, “Systems on a Chip” in the master’s program, Python is an additional elective course. Initially, the Digital Circuit/System Design course involves designing a small portion of a RISC-V processor using VHDL/FPGA. The Systems on a Chip course emphasizes the combined development of software and hardware. In the Python course, advanced students develop support tools or simulators to improve their understanding of the RISC-V architecture.

In United International University (Dhaka, Bangladesh) a training program is proposed in the laboratory course on microprocessors and microcontrollers that is close in content to real developments [9].

The Master’s degree program at Apollo University (India) “VLSI Design and Embedded Systems” is presented in [10].

The RVfpga course from Imagination University is presented in [11]. RISC-V is an open standard architecture of the processor instruction set based on the principles of the reduced instruction set computing (RISC). RISC-V was released in 2010 at the University of California, Berkeley. Unlike proprietary architectures, RISC-V is free and open, which allows anyone to design, manufacture, and sell RISC-V chips and software for them. The modularity of RISC-V allows it to be used in a wide range of applications from small embedded systems to powerful supercomputers. A growing ecosystem and community support make the RISC-V architecture attractive for use in scientific research, education, and commercial development. The RVfpga course offers a good introduction to computer architectures using RISC-V and FPGA. The course materials, including a detailed description of lab work, are posted free of charge on the Imagination University Program website. The RVfpga course has been adapted into a MOOC on the edX platform.

The following articles focus on improving the quality of learning through changes in teaching methods. The authors of the following articles emphasize the need to modernize teaching methods in order to enhance their effectiveness and propose their own approaches to addressing the issue.

Ma *et al.* [12] recommend using project-based learning, paying special attention to each of the following design stages: implementation - evaluation - feedback - improvement.

Awwad [13] report on the author's application of active learning strategy to improve academic performance in electronic engineering disciplines. The author has achieved significant impact on student engagement, growth of their practical skills, and learning outcomes. Although traditional lecture-based methods have been used for a long time, they are not sufficiently engaging for students and do not provide quality learning. The author integrates active learning strategies such as project-based learning; use of simulation software; flipped classroom.

Arredondo *et al.* [14] presents the result of an innovative approach to teaching based on the creation of educational videos by students and their evaluation by students and teachers in an anonymous mode within the framework of the discipline "Electrical circuits".

3. THE IMPORTANCE OF SOFTWARE AND HARDWARE IN EDUCATION

The use of interactive web-based learning is gaining popularity as it allows students to actively engage with educational content, offering flexibility in terms of time and space, allowing students to move through the learning material at their own pace [15].

The use of simulation of electronic systems improves the quality and reduces the cost of the learning process [16].

Virtual online simulation laboratories increase the effectiveness of the learning process through personalization, formative and final assessment [17].

The integration of software for simulating electronic circuits in the study of the fundamentals of electronics significantly improves the quality of the educational process and the competence of graduates [18].

A comparative analysis of the advantages and disadvantages of physical and virtual training in the design of electronic devices is presented in the article [19].

4. DIGITAL LOGIC DESIGN TEACHING TOOLS

Processor Development Platform RISC-V in register transfer language is presented in [20]. The authors developed the RIVL platform to support the processor development process, which integrates processor design in register transfer language, integration, verification and placement of components on the chip. The platform was tested on 60 RISC-V processors.

A web system (with open-source texts) for designing digital devices, based on the SHDL language and translating it into

VHDL is presented in [21]. The students' work in standard VHDL caused great difficulties, so the authors developed the SHDL (Small Hardware Description Language) - as a simplified version of VHDL. Using the SHDL language reduces the number of student errors in syntax, simulation and synthesis. Using the SHDL language and the web system, the authors created many digital circuits to prepare students for developing their own projects. The process of designing devices by students consists of describing components, simulation, synthesis, and implementation on programmable devices.

An FPGA tool for designing, modeling and evaluating an object recognition system is presented in [22]. Computer vision is characterized by computationally intensive algorithms and strict requirements for real-time execution. FPGAs (Field Programmable Gate Arrays) are based on a parallel paradigm that allows efficient hardware design and are therefore ideal for solving computer vision problems. It is therefore clear that students should be taught this as well.

The mobile specialized game Go Electronics for teaching digital electronics is presented in [23]. According to the authors, the evolution of teaching technology requires modernization of teaching methods. The work describes the game Go Electronic, developed to improve the teaching of digital electronics, which is quite difficult for many students using traditional methods. In this game, students consistently answer questions on the phases of project development: requirements analysis, system design, implementation, integration, testing, support.

The study of combinational and sequential circuits using a special board based on Arduino is presented in [24].

Improving the quality of teaching logical design based on Arduino is presented in [25]. This article presents a low-cost set of tools for teaching the design of digital combinational and memory circuits based on the Arduino kit, which is used both as a programmable signal generator and as a logic analyzer. For experiments, 74HC series logic chips are connected to the Arduino UNO R3. Students are asked to test the 74HC04 inverter, 74HC08 AND gates, 74HC32 OR gates, 74HC86 XOR gates, 74HC138 decoder, 74HC151 multiplexer, 74HC74 D-trigger.

Usage PyEDA and Logisim in the course "Design of Digital Systems" are presented in [26]. According to the authors, digital electronic devices penetrate into all spheres of our lives. To meet new requirements, engineering programs often include at least one corresponding course. "Design of Digital Systems" is a basic subject for students studying computer science. It includes Boolean algebra, logical minimization, binary arithmetic, analysis and synthesis of digital systems, providing the ability to design and test them. At the same time, this subject causes difficulties for many students. The authors present two adapted software packages PyEDA and Logisim, which increase the effectiveness of teaching the course "Design of Digital Systems". PyEDA is a Python library for automating the design and simulation of electronic devices, including support for Boolean algebra, truth tables, minimization of Boolean functions. In particular, it allows you to minimize Boolean functions. Logisim allows you to design (draw circuits) and simulate digital devices.

An overview of the software used for hardware training in Pakistan is presented in [27]. MATLAB, Simulink, Python are mentioned as the main ones.

The use of virtual reality in the process of teaching DC electrical circuits is presented in [28].

5. MICROPROCESSOR/MICROCONTROLLER PROGRAMMING TRAINING TOOLS

Studying microcontroller programming - a graphical or textual start? - this is the question asked by the authors of [29]. They claim that students achieve the best results with textual programming.

A remote laboratory based on Raspberry Pi Pico is presented in [30]. Raspberry Pi Pico, based on the RP2040 processor, is an easy-to-use microcontroller development board that is very convenient for learning the basics of electronics to first-year students. This article presents a remote laboratory for blended learning developed by the authors using Raspberry Pi Pico for the course "Digital Circuits and Systems". Due to the impact of COVID-19, the number of students attending laboratory classes has decreased, so the authors provided an alternative by using the online Raspberry Pi Pico simulator from Wokwi for those students who could not physically attend the laboratory. The entire learning process includes three dependent sessions. Upon completion of these, the student should understand basic digital logic and electronic circuits by developing a simplified traffic light controller using Raspberry Pi Pico and programming in Python.

Online learning platforms are beginning to play a significant role in improving the quality of education. Xin *et al.* [31] presents the author's development of an online platform for training and assessment, using the full development stack, including a knowledge graph, principles of sensor operation, communication protocols, and microcontroller programming (STM32, Arduino, MicroPython).

Rezo *et al.* [32] consider the use of a simulator program to improve online courses in microprocessor programming. Compared to traditional laboratories, simulator programs offer cost-effective, scalable platforms for simulating various processors. According to the authors, such training allows for an effective bridge from theory to practice in teaching microprocessor programming, supports active learning, and develops critical thinking.

MOOC (Massive Open Online Course) is a type of distance education that offers a large set of online courses to anyone, with no limits on the number of participants and open access via the Internet. MOOC courses can cover a wide range of topics, from humanities to technical sciences, and are often provided for free or for a nominal fee. With the introduction of MOOCs, more and more courses are connected to online platforms that provide shared access to learning resources and ensure continuous learning. [33] describes the creation of an online MOOC course on programming the STM32 microcontroller.

Hands-on experience is very important in engineering education. However, the establishment of physics laboratories

requires large financial resources, which are lacking in universities with low financial resources. [34] discusses a new method of experimental learning using Virtual Simulation Machine (VSM) to simulate microcontrollers.

The creation of an open instruction set for the RISC-V architecture has made it possible to design high-performance processor cores by extending a relatively simple architecture with custom functionality specific to the corresponding application. Designing and simulating such applications is traditionally done using hardware description languages, which is a time-consuming and labor-intensive process. [35] presents SyncRim, a high-level synchronous circuit simulator that omits irrelevant details of the processor simulation model, allowing rapid prototyping of any processor core extension. It is shown how a synchronous circuit simulator can be developed using the Rust programming language and how this simulator can be used for joint development of software and hardware based on the RISC-V core. The paper also provides a brief description of popular alternative open-source processor simulators. QEMU – supports dozens of processor models – is based on dynamic binary translation, which means that the instructions of the target architecture are transpiled to the host architecture at runtime. This allows to omit the details of the hardware implementation, which significantly increases efficiency, but complicates the joint development of software and hardware. ReNode – a high-level simulator that supports software debugging for multiprocessor networks, but does not support joint debugging of software and hardware. QtRvSim – a graphical simulator that supports many RISC-V models at a low level, displaying signals between components. Verilator – a Verilog/SystemVerilog simulator. In essence, Verilator converts HDL descriptions of hardware, providing the lowest level of abstraction.

Newhall *et al.* [36] present ASM Visualizer - a tool designed to help students learn assembly programming. ASM Visualizer allows you to execute an assembler program by instructions and observe changes in registers, stack and RAM. ASM Visualizer is built on a client-server architecture. The client works in a web browser to edit programs and display their execution, and the server compiles and executes programs, and forms data traces for displaying them on the client. Supported processor models are x86, x86_64, ARM_64, [37] considers MicroPython as an educational tool for teaching microcontroller programming and embedded systems development. The article presents several lab examples written in MicroPython for Raspberry Pi Pico, RP2040. The arguments for and against such an approach to teaching microcontroller programming and developing embedded systems based on them are considered

Eliza *et al.* [38] describes a tool the authors created for teaching students how to program Arduino-based microcontrollers.

Yusop *et al.* [39] presents a literature review on Arduino application development in education.

Arduino provides students with opportunities to improve their coding skills. [40] describes a special software tool created by the authors called AABC (Assessing Arduino Basics in Coding) for assessing Arduino coding skills.

Kissich *et al.* [41] present the authors test system ATTEST, which is used in the course “Real-Time Operating Systems” for two semesters in Graz University of Technology, Austria. The response time is quite short, despite the technical limitations of the used board based on the MSP430 microcontroller. The evaluation is carried out on the basis of carefully prepared tests. Each student package (commit in GIT) is compiled individually for each of the prepared tests, which remain a “black box” for students. The built code is executed on a real device due to the lack of accurate simulation models. The authors point out that the MSP430 is additionally equipped with an oscilloscope, which is used in some tests. Each test has its own report on its run. Students receive some advice on possible sources of errors; teachers have full access to all logged details. The test system has undeniable advantages in assessing students' work and provides direct support. In addition, anonymized data can help teachers study how students learn.

Steiner *et al.* [42] developed an automatic microcontroller program evaluation system to generate fast and accurate feedback for student submissions in embedded software development courses.

The course “Microcontrollers and Embedded Systems” (Marwadi University, India) requires practical implementation of hardware, software and their integration. In doing this work, one has to use many tools, such as editors, compilers, debuggers. As an assessment of this work, the authors of [43] suggest using “open book examination”, when one is allowed to use any sources when preparing an answer to the exam.

Microcontrollers are widely used in agriculture. However, most students in agricultural colleges do not know how to program well enough to use these devices in their academic program or career. At the same time, generative artificial intelligence (GenAI) chatbots can write quite complex programs for microcontrollers if they are given adequate requests.

Johnson *et al.* [44] describe a study in which students had to complete their assignment on developing programs for a microcontroller using GenAI. 9 out of 11 student pairs successfully completed their assignment.

Developing software for embedded digital systems requires a lot of developer knowledge, huge labor and time costs, and provokes errors in projects. To solve this problem, the authors of [45] developed the EmbedGenius system - a platform for fully automatic software development. The main idea is to use LLM. During the EmbedGenius testing process, about 350 tasks were completed with 95% accuracy.

6. TEACHING TOOLS FOR COLLABORATIVE DEVELOPMENT OF SOFTWARE AND HARDWARE FOR EMBEDDED DIGITAL SYSTEMS

Pfau *et al.* [46] presents a technology platform for teaching SoC (system on a chip) design. Students write their code in Verilog for digital and analog projects. They then simulate

them, debug the software and hardware. The project is then prototyped on FPGA and transferred to ASIC. The project is completed by creating technical documentation for it.

CAD (Computer-Aided Design) systems are very important in education, but traditional CAD labs have disadvantages such as limited access, high cost, space and time constraints. Therefore, virtual CAD labs are becoming more widespread. The authors of [47] describe their virtual CAD lab built on the FreeCAD system (open source), which provides a flexible, cheap and widely accessible learning environment that overcomes the disadvantages of physical CAD labs.

Antonescu *et al.* [48] presents a project (software and hardware) designed as a universal learning platform (ULP) for face-to-face and remote learning. The ULP provides a laboratory basis for several e-learning courses, offering measurement tools, multiplexing and reconfiguration of chips, and software control of prerequisites. The ULP has been in use since 2020 and has increased the success rate of e-learning courses from 62% to 91%.

The Internet-of-Things (IoT) has created a complex environment where hardware and software interact in complex ways. To simplify the development of applications for the IoT, WebAssembly (Wasm), a binary instruction format for a stack-based virtual machine, was developed. Wasm is designed as a portable target architecture for programming languages, enabling web development of client-side and server-side applications. Ribeiro *et al.* [49], present WASMICO, a software for microcontrollers that supports the full application lifecycle. WASMICO uses the Wasm3 interpreter and FreeRTOS to provide parallel execution of WebAssembly tasks. Based on such containerization, developers can write programs in various programming languages, compile them to WebAssembly using existing tools, and remotely download, run, and manage all tasks *via* an HTTP API and command-line interface.

CONCLUSION

The article provides a review of the literature on methods and tools for autonomous and collaborative development and simulation of hardware and software for digital systems that are used in real university practice. Particular attention is paid to such issues as options for the content and methods; the importance of software and hardware training tools; tools for teaching digital logic design; tools for teaching programming of microprocessors/microcontrollers; tools for teaching collaborative development of software and hardware for embedded systems.

An important area of development is the development and implementation of online training systems.

It is noted that the experience of using generative artificial intelligence to create formal models of digital systems in hardware description languages and program texts for microcontrollers/microprocessors in assembly, C, Python is rapidly developing.

LIST OF ABBREVIATIONS

ARM	=	Advanced RISC Machine
ADDIE	=	Analysis – Design – Development – Implementation - Evaluation
CAD	=	Computer-Aided Design
CPLD	=	Complex Programmable Logic Device
FPGA	=	Field Programmable Gate Array
IoT	=	Internet of Things
MOOC	=	Massive Open Online Course
OBE	=	Outcomes-Based Education
RISC	=	Reduced Instruction Set Computer
SoC	=	System on Chip
SHDL	=	Small Hardware Description Language
ULP	=	Universal Learning Platform
VSM	=	Virtual Simulation Machine

AUTHOR'S CONTRIBUTION

M.D. contributed to the conceptualization and design of the study, data collection, analysis and interpretation of the findings, and preparation of the manuscript.

CONSENT FOR PUBLICATION

Not applicable.

AVAILABILITY OF DATA AND MATERIALS

The data will be made available on reasonable request by contacting the corresponding author [M.D.].

FUNDING

None.

CONFLICT OF INTEREST

The author declares no conflicts of interest.

ACKNOWLEDGEMENTS

Declared None.

DECLARATION OF AI

AI-assisted tools were used solely for manuscript enhancement, including language improvement and editorial refinement. All AI-assisted modifications were carefully reviewed and verified, and full responsibility for the accuracy, originality, and integrity of the final manuscript is accepted.

REFERENCES

- [1] L. Carro, F. Regazzoni, and C. Pilato, "Embedded Computer Systems: Architectures, Modeling, and Simulation: 24th International Conference, SAMOS 2024, Samos, Greece, June 29–July 4, 2024, Proceedings," Springer Nature, pp. 265, 2025, <https://doi.org/10.1007/978-3-031-78377-7>
- [2] M. Wolf, D. Atienza, and R. K. Gupta, "The Development of Embedded Computing as a Discipline [Special Section on 2025 IEEE Kirchhoff Award]," *IEEE Circuits Syst. Mag.*, vol. 25, no. 2, pp. 32–38, Second Quarter 2025, <https://doi.org/10.1109/MCAS.2025.3532884>
- [3] S. Inchan and S. Akatimagool, "A Blended Laboratory-Based Learning Model for Embedded Systems Education," *J. Tech. Educ. Train.*, vol. 17, no. 1, pp. 1–13, 2025, Available from: <https://publisher.uthm.edu.my/ojs/index.php/JTET/article/view/17516>
- [4] S. Liu, R. Zhai, and H. Peng, "Research on Digital Logic Circuit Teaching for Training System Abilities in Computer Science Majors," in *Proc. 4th Int. Conf. New Media Development and Modernized Education (NMDME 2024)*, *Adv. Intell. Syst. Res.*, Art. no. 188, 2024, https://doi.org/10.2991/978-94-6463-600-0_53
- [5] T. Diao and J. Li, "An Exploration of Practical Teaching of 'Computer Organization and Structure' for the Software Engineering Major," in *Proc. 4th Int. Conf. Education, Language and Art (ICELA 2024)*, *Adv. Soc. Sci. Educ. Humanit. Res.*, Art. no. 907, 2024, https://doi.org/10.2991/978-2-38476-364-1_25
- [6] W. Yu, "Study on Teaching Reform of 'Microcontroller Technology' Course Based on OBE-ADDIE," *Curriculum Teach. Methodol.*, vol. 7, no. 8, pp. 143–148, 2024, <https://doi.org/10.23977/curtm.2024.070822>
- [7] K. C. Lau, M. Ab. Rahman, and C. U. Eh Kan, "Design and Implementation of Pipelined RISC Processor for Educational Purpose," *Evol. Electr. Electron. Eng.*, vol. 5, no. 2, 2024, Available from: <https://periodical.uthm.edu.my/index.php/eeee/article/view/17623>
- [8] F. Sprang, T. Latzel, and F. Englberger, "FPGA and RISC-V integration in electrical engineering curriculum," in *ICERI 2024 Proceedings*, pp. 2560–2568, 2024, <https://doi.org/10.21125/iceri.2024.0682>
- [9] F. Hafiz, M. Emon, A. Hossain, S. Mukta, S. Islam, and S. Shatabda, "Design of a microprocessors and microcontrollers laboratory course addressing complex engineering problems and activities," *Comput. Appl. Eng. Educ.*, vol. 33, no. 2, Art. no. e70006, 2025, <https://doi.org/10.1002/cae.70006>
- [10] Apollo. Master's Degree Program in VLSI Design and Embedded Systems at Apollo University (India), 2023, Available from: <https://apollouniversity.edu.in/wp-content/uploads/2023/10/M-TECH-SULLUBUS-VLSI-ES.pdf>

- [11] D. Chaver, S. Harris, L. Piñuel, O. Kindgren, R. Kravitz, J. I. Gómez, F. Castro, K. Olcoz, J. Villalba, A. Grinshpun, F. Gabbay, L. Seed, R. Duarte, M. López, O. Alonso, and R. Owen. Teaching Experiences using the RVfpga Package. 2024, Available from: <https://arxiv.org/pdf/2411.14954>
- [12] H. Ma, L. Li, Y. Ma, X. Liu, and S. Yang, "Exploration and application of project-based teaching integrating scientific research achievements in electronic comprehensive design course," *Comput. Appl. Eng. Educ.*, vol. 33, no. 33, Art. no. e70047, 2025, <https://doi.org/10.1002/cae.70047>
- [13] F. Awwad, "Enhancing electronics courses education: Active learning strategies for undergraduate engineering students," *Int. J. Eng. Pedagogy (iJEP)*, vol. 15, no. 2, pp. 42–73, 2025, <https://doi.org/10.3991/ijep.v15i2.51739>
- [14] F. Arredondo, B. García, and R. Lijo. (2025). Learning Through Explanation: Producing and Peer-Reviewing Videos on Electric Circuits Problem Solving, *IEEE Transactions on Education*, vol. 68, no. 1, pp. 67–78, 2025, <https://doi.org/10.1109/TE.2024.3454008>
- [15] J. Handhika, "Learning the basic skills in digital electronics technology: An interactive web-based application for pre-service technical-vocational educators," in *Proc. 4th Int. Conf. Educ. Technol. (ICETECH 2023), Atlantis Highlights in Social Sciences, Education and Humanities*, 2023, https://doi.org/10.2991/978-94-6463-554-6_9
- [16] C. Tokatlidis, S. Rapti, S. Tselegkaridis, T. Sapounidis, and D. Papakostas, "Virtual environment in engineering education: The role of guidance, knowledge and skills development in electronic circuits teaching," *Educ. Sci.*, vol. 14, no. 12, Art. no. 1336, 2024, <https://doi.org/10.3390/educsci14121336>
- [17] J. Chang and D. Liu, "Optimizing learning outcomes: A comprehensive approach to virtual simulation experiment teaching in higher education," *Int. J. Human-Comput. Interact.*, vol. 41, no. 4, pp. 2114–2134, 2025, <https://doi.org/10.1080/10447318.2024.2314825>
- [18] A. Gonzales and G. Gonzales, "Exploring users' acceptance of electronic circuits simulation: Implications to teaching basic electronics," *Social Sci. Humanit. Open*, vol. 9, Art. no. 100821, 2024, <https://doi.org/10.1016/j.ssaho.2024.100821>
- [19] C. Tokatlidis, S. Tselegkaridis, S. Rapti, T. Sapounidis, and D. Papakostas, "Hands-on and virtual laboratories in electronic circuits learning: Knowledge and skills acquisition," *Information*, vol. 15, no. 11, Art. no. 672, 2024, <https://doi.org/10.3390/info15110672>
- [20] L. Xu *et al.*, "RIVL: A low-cost SoC agile development platform for multiple RISC-V processors design and verification," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 72, no. 4, pp. 1668–1678, 2025, <https://doi.org/10.1109/TCSI.2024.3509634>
- [21] A. Trost, J. Jamšek, and A. Žemva, "SHDL: A hardware description language and open-source web tool for online digital systems design teaching," *Electronics*, vol. 12, no. 2, Art. no. 425, 2023, <https://doi.org/10.3390/electronics12020425>
- [22] E. Guzmán-Ramírez, I. García, C. Pacheco, and E. Guerrero-Ramírez. An FPGA-based tool for supporting the design, modeling, and evaluation of hybrid object recognition systems on computer engineering courses, *Comput. Appl. Eng. Educ.* 32, e22726, 2024, <https://doi.org/10.1002/cae.22726>
- [23] N. H. Hasmizan, M. N. A. Azman, M. J. R. Prestoza, and M. S. Othman, "Design and development of mobile teaching aids using Go-based electronic games for teaching digital electronics in higher education," *Int. J. Interact. Mobile Technol. (IJIM)*, vol. 19, no. 3, pp. 4–21, 2025, <https://doi.org/10.3991/ijim.v19i03.51691>
- [24] Lee, E. Development and implementation of an arduino-based digital logic circuit practice board module. *Educ Inf Technol.* vol. 30, pp 21249–21282, 2025, <https://doi.org/10.1007/s10639-025-13604-7>
- [25] S. Gulgonul, "Enhancing logic design education with a low-cost Arduino take-home lab," 2025, <https://doi.org/10.21203/rs.3.rs-6663829/v1>
- [26] G. Wang and Y. Zhao, "Enhancing teaching effectiveness and learning experience of digital circuit design using multiple tools," in *Proc. 2023 ASEE Annu. Conf. Expo.*, American Society for Engineering Education, 2023, Available from: <https://peer.asee.org/enhancing-teaching-effectiveness-and-learning-experience-of-digital-circuit-design-using-multiple-tools.pdf>
- [27] A. R. Bali and Y. Mughal, "The impact of different computing tools in electrical engineering learning," *J. Eng. Comput. Intell. Rev.*, vol. 3, no. 1, pp. 52–67, 2025, Available from: <https://jecir.com/index.php/jecir/article/view/12>
- [28] D. A. Albarracin-Acero, F. A. Romero-Toledo, C. E. Saavedra-Bautista, and E. A. Ariza-Echeverri, "Virtual reality in the classroom: Transforming the teaching of electrical circuits in the digital age," *Future Internet*, vol. 16, no. 8, Art. no. 279, 2024, <https://doi.org/10.3390/fi16080279>
- [29] F. Vrbaničič and S. Kocijančič, "Strategy for learning microcontroller programming: A graphical or a textual start?," *Educ. Inf. Technol.*, vol. 29, pp. 5115–5137, 2024, <https://doi.org/10.1007/s10639-023-12024-9>
- [30] Z. C. M. Davidson, S. Dang, and X. Vasilakos, "Blended laboratory design using Raspberry Pi Pico for digital circuits and systems," *IEEE Trans. Learn. Technol.*, vol. 17, pp. 1170–1183, 2024, <https://doi.org/10.1109/TLT.2024.3363230>
- [31] G. Xin, H. Xie, and S. Kang, "Design and development of an intelligent connected access integrated training and assessment platform," *J. Electr. Syst.*, vol. 20, no. 2, pp. 900–908, 2024, <https://doi.org/10.52783/jes.1250>

- [32] V. Režo, F. Gerhát, and M. Weis, "Use of modern software environments for online teaching of microprocessor programming," in *Proc. 21st Int. Conf. Emerg. eLearning Technol. Appl. (ICETA)*, Sary Smokovec, Slovakia, 2023, pp. 415–420, 2023, <https://doi.org/10.1109/ICETA61311.2023.10344061>
- [33] F. Wei, "Thoughts on online teaching of STM32," *Educ. Rev., USA*, vol. 8, no. 7, pp. 960–963, 2024, <http://dx.doi.org/10.26855/er.2024.07.011>
- [34] M. Anees, B. G. Nagaraja, Y. Thimmaraja, and P. Chandrashekar, "Experimental-based learning: Microcontroller simulation using virtual simulation machine," *J. Eng. Educ. Transform.*, vol. 38, pp. 199–209, 2025, Available from: https://www.researchgate.net/publication/388589107_Experimental_Based_Learning_Microcontroller_Simulation_Using_Virtual_Simulation_Machine
- [35] P. Dzialo, "SyncRim: Hardware/software co-design for the RISC-V architecture using high-level simulation in Rust," M.S. thesis, Luleå Univ. Technol., Luleå, Sweden, 2024, Available from: <https://www.diva-portal.org/smash/get/diva2:1895769/FULLTEXT01.pdf>
- [36] T. Newhall, K. C. Webb, I. Romea, E. Stavis, and S. J. Matthews, "ASM visualizer: A learning tool for assembly programming," in *Proc. 56th ACM Tech. Symp. Comput. Sci. Educ. (SIGCSE TS '25)*, pp. 840–846, 2025, <https://doi.org/10.1145/3641554.3701793>
- [37] B. Hur, "Python-based microcontroller architecture and microcontroller application education in engineering technology," in *Proc. 2025 ASEE Gulf-Southwest Annu. Conf.*, 2025, Available from: <https://peer.asee.org/55073.pdf>
- [38] F. Eliza, O. Candra, R. Mukhaiyar, R. Fadli, and Y. E. Wibowo, "Development of a microcontroller training kit to increase student learning motivation," *Eng. Technol. Appl. Sci. Res.*, vol. 15, no. 3, pp. 22319–22326, 2025, <https://doi.org/10.48084/etasr.10142>
- [39] N. Yusop, N. A. Mokhtar, and S. Sadikan, "Development of Arduino applications for IoT applications in software engineering education: A systematic literature review," *Bull. Electr. Eng. Inform.*, vol. 13, no. 3, pp. 1824–1831, 2024, <https://doi.org/10.11591/eei.v13i3.4506>
- [40] S. Tselegkaridis, T. Sapounidis, C. Tokatlidis, S. Rapti, and D. Papakostas, "AABC: A tool for assessing Arduino basic coding skills," *Comput. Appl. Eng. Educ.*, vol. 33, Art. no. 70029, 2025, <https://doi.org/10.1002/cae.70029>
- [41] M. Kissich, K. Weinbauer, and M. Baunach, "Improving embedded software development courses: A student attentive data-driven approach," in *Proc. 2025 Eur. Conf. Software Eng. Educ. (ECSEE)*, pp. 171–175, 2025, <https://doi.org/10.1145/3723010.3723013>
- [42] F. Steiner, B. Lueger, B. Wallisch, and T. Polzer, "Automated evaluation system for microcontroller assignments," in *Proc. 2023 IEEE Global Eng. Educ. Conf. (EDUCON)*, pp. 1–7, 2023, <https://doi.org/10.1109/EDUCON54358.2023.10125112>
- [43] N. Joshi, A. Ved, A. Bhogayata, T. Trivedi, T. Vora, N. Rachchh, and R. Doshi, "Diversifying assessment approaches: A success story of assessing microcontroller and embedded systems," *J. Eng. Educ. Transform.*, vol. 38, no. 2, pp. 343–349, 2025, <https://doi.org/10.16920/jeet/2025/v38is2/25041>
- [44] D. M. Johnson, W. Doss, and C. M. Estepp, "Agriculture students' use of generative artificial intelligence for microcontroller programming," *Nat. Sci. Educ.*, vol. 53, no. 2, Art. no. e20155, 2024, <https://doi.org/10.1002/nse2.20155>
- [45] H. Yang, M. Li, M. Han, Z. Li, and W. Xu, "EmbedGenius: Towards automated software development for generic embedded IoT systems," 2024, Available from: <https://arxiv.org/pdf/2412.09058>
- [46] J. Pfau, R. Leys, M. Neu, A. Serdyuk, I. Peric, and J. Becker, "A unified SoC lab course: Combined teaching of mixed-signal aspects, system integration, software development and documentation," in *Proc. 2023 IEEE Int. Symp. Circuits Syst. (ISCAS)*, pp. 1–5, 2023, <https://doi.org/10.1109/ISCAS46773.2023.10181679>
- [47] S. B. Khot, R. Magdum, P. Jadhav, R. Desavale, and S. Shriguppikar, "Bridging the gap: A case study on implementing and evaluating a virtual CAD modelling lab for enhanced accessibility and learning," *J. Eng. Educ. Transform.*, vol. 38, no. 2, pp. 505–512, 2025, <https://doi.org/10.16920/jeet/2025/v38is2/25062>
- [48] M. Antonescu, F.-S. Dumitru, and M. Enachescu, "The Universal Learning Platform (ULP): A hybrid IoT-based learning environment for automatic device and circuit characterization," *IEEE Access*, vol. 13, pp. 45168–45186, 2025, <https://doi.org/10.1109/ACCESS.2025.3550022>
- [49] E. Ribeiro, A. Restivo, H. S. Ferreira, and J. P. Dias, "WASMICO: Micro-containers in microcontrollers with WebAssembly," *J. Syst. Softw.*, vol. 214, Art. no. 112081, 2024, <https://doi.org/10.1016/j.jss.2024.112081>