Implementation of educational tools for microprocessor systems design: fostering digital competence among engineering students

Viktoriia V. Tkachuk, Yuliia V. Yechkalo, Vladyslav O. Fedotov and Sergii A. Ruban

Kryvyi Rih National University, 11 Vitalii Matusevych Str., Kryvyi Rih, 50027, Ukraine

Abstract

This research examines the effectiveness of educational tools for the design of microprocessor systems in developing digital competence among engineering students. Through a quasi-experimental study that compares traditional teaching methods with an integrated approach using MCU 8051 IDE, STC-ISP, and PROTEUS DESIGN tools, we demonstrate significant improvements in student performance and skill development. The experimental group showed an increase in average academic scores and substantial enhancement in digital competencies compared to the control group. Our methodology emphasizes a practice-oriented approach that balances theoretical knowledge with hands-on experience through laboratory-based learning, simulation environments, and hardware implementation. The findings contribute to best practices in engineering education by establishing a comprehensive framework to develop digitally competent graduates prepared for industry challenges in the design of microprocessor systems. This research addresses the persistent gap between academic instruction and professional practice while aligning with European Digital Competence Standards adapted for engineering contexts.

Keywords

microprocessor systems, digital competence, engineering education

1. Introduction

The rapid evolution of microprocessor technologies and their expanding applications across industries requires a comprehensive reassessment of educational approaches in engineering curricula. As digital systems become increasingly embedded in all aspects of modern infrastructure, there exists a critical need to equip future engineers with both theoretical knowledge and practical competencies in microprocessor systems design. This educational challenge has been further complicated by recent global disruptions, including the COVID-19 pandemic [1] and Russian invasion [2], which has accelerated the demand for innovative teaching methodologies and virtual learning environments. The article addresses these pressing concerns by synthesising contemporary research on effective pedagogical strategies, highlighting the significance of project-based learning, cross-platform development, and industry-aligned curriculum restructuring. Particularly noteworthy is the emphasis on digital competence frameworks that provide structured approaches to developing the essential skills required for engineering graduates to navigate the complexities of modern technological landscapes and contribute meaningfully to their advancement.

Teaching microprocessor systems design to engineering students presents several challenges that educators must address to ensure effective learning outcomes. As noted by Pellicano et al. [3], "students struggle to learn embedded systems, connect embedded topics between courses, and apply those topics to real-world applications, thus facilitating a need for Modular Embedded Tools integrated within an electrical and computer engineering curriculum".

CTE 2024: 12th Workshop on Cloud Technologies in Education, co-located with the 6th International Conference on History, Theory and Methodology of Learning (ICHTML 2025), May 12, 2025, Kryvyi Rih, Ukraine

viktoriya.tkachuk@gmail.com (V. V. Tkachuk); uliaechk@gmail.com (Y. V. Yechkalo); fedotov@knu.edu.ua (V. O. Fedotov); serhii.ruban@knu.edu.ua (S. A. Ruban)
0000-0002-5879-5147 (V. V. Tkachuk); 0000-0002-0164-8365 (Y. V. Yechkalo); 0000-0002-6536-5591 (V. O. Fedotov); 0000-0002-4495-6667 (S. A. Ruban)

The development of effective microprocessor systems education requires both structured methodological approaches to learning resources and sophisticated analytical methods to evaluate. Shebanin et al. [4, 5] present a mathematical model for informational resources in distance learning systems that provides valuable structural insights, while their earlier work on fuzzy predicates and quantifiers in informational resource modeling offers a robust methodological foundation for assessing digital competence development among engineering students.

A significant challenge identified by Hertzog and Swart [6] report that standardising on Arduino microprocessors led to improved academic outcomes, with "more than 90% of the students successfully completing this design-based module, while 70% felt that it really helped them better understand the theoretical knowledge". Collaborative learning environments have proven to be effective in engineering education. Solesvik et al. [7] demonstrate how joint digital simulation platforms enhance learning outcomes and preparedness, a concept directly applicable to our implementation of simulation-based tools like PROTEUS DESIGN in microprocessor education.

The COVID-19 pandemic created additional obstacles for hands-on learning. Zhang [8] describes how the emergency remote teaching situation necessitated innovative approaches: "For project-based electrical and computer engineering courses that involve hardware components and group work, additional challenges include limited or no access to facilities for experimental work; students cannot meet in person to conduct group work, especially for projects involving hardware integration". During the pandemic, virtual environments became crucial for maintaining practical learning experiences. Deepa et al. [9] highlight the "effectiveness of using open source emulating environments like Edsim51 and EMU8086 for providing a virtual laboratory experience to the students". These environments provided "complete visualization of the internal functionality of the microprocessor/microcontroller architecture and also enhanced practical exposure of the students".

Modern processors' complexity presents another notable challenge. Kostadinov and Bencheva [10] argue that "teaching processor design is an ongoing challenge for instructors due to the ever-increasing complexity of modern processors". Their response was to develop "a family of FPGA-based processors and set of tools for instructional use" that simplified the learning process while maintaining practical relevance. Bridging theoretical knowledge and practical application remains a persistent challenge in engineering education. Ferlin and Pilla [11] identify "a great problem: the migration from theory to practice, from the abstract concepts to laboratory experimentation". Their solution involved "driving students from classroom theory to laboratory experimentation" through various tools, including hardware description languages and FPGA implementations. Dabroom et al. [12] acknowledge this challenge: "Microcontroller based system courses are usually attended by a wide range of students who came not only from different backgrounds but also with different programming skills". They addressed this diversity by designing a course that incorporated graphical programming tools for beginners while allowing advanced students to use high-level languages.

Maintaining student interest is crucial for effective learning. Parikh [13] observes that "difficulty arises when traditional classroom teaching with examples and non-creative projects instill boredom amongst students". To combat this, Parikh introduced an autonomous robot platform that allowed students to apply microcontroller concepts in engaging ways. Deepa et al. [14] emphasise that "fast-paced technological advancements lays more emphasis on embedded system design to real-world applications". Their approach involved offering additional value-added courses and implementing Agile-based project learning to bridge the gap between academic learning and industry requirements. Brown [15] highlights the importance of balancing new technologies with fundamental concepts: "Without these concepts, students will be unlikely to develop an intuitive understanding of the nature of logic circuits or processor".

As engineering education continues to evolve to meet industry demands, the implementation of specialized educational tools becomes increasingly important. Bakum and Tkachuk [16] emphasize that "the formation of future mining engineers' professional competence requires an innovative educational system", a principle equally applicable to developing digital competence in microprocessor systems education. Their research on innovative approaches in technical engineering education provides a valuable framework for understanding how specialized tools can bridge theoretical knowledge and practical application – a central concern in fostering comprehensive digital competence among engineering students.

The integration of advanced educational technologies in engineering education has become increasingly crucial, with curriculum adaptation to technological advances being particularly important. Kondratenko et al. [17, 18], Talaver and Vakaliuk [19], Bondar et al. [20], Haranin and Moiseienko [21], Korotun et al. [22], Osadcha and Shumeiko [23], Shumeiko and Osadcha [24], Tarasova and Doroshko [25] emphasise that the extensive implementation of AI in educational processes and systems has caused fundamental changes in modern higher education, while discussing the symbiotic relationship between AI and higher education systems. Building on this foundation, Kondratenko et al. [17] propose methodological approaches for university curricula modernisation based on advancements in information and communication technologies, providing a framework that can be effectively applied to microprocessor systems education and the development of digital competencies among engineering students. The integration of innovative educational technologies has proven effective across various engineering disciplines. Tkachuk et al. [26] demonstrated that "Augmented and Virtual Reality Tools in Training Mining Engineers" significantly enhanced practical skills acquisition and spatial understanding among engineering students. Their findings on immersive technology implementation provide valuable insights for our research on microprocessor systems education, particularly regarding the development of interactive learning environments that bridge theoretical knowledge and practical application.

Research indicates several effective approaches to teaching microprocessor systems design, with hands-on, project-based learning emerging as particularly valuable. Project-based learning has proven effective in developing both technical and soft skills. Metri et al. [27] note that their laboratory-focused course allowed students to combine "learning of microcontroller, control systems and power electronics courses," providing a comprehensive understanding of interconnected engineering concepts. Schneider and Peterson [28] detail how students "develop a complete microprocessor system for audio recording and playback" that requires "both hardware and software development; a true co-design experience". Such projects integrate multiple concepts and simulate real-world engineering challenges. McLauchlan [29] restructured a microprocessor-based control class to "include a practical design project as opposed to only simulations" because "students learn more and get more engaged in a project-oriented learning environment". This approach allowed students to "gain a greater understanding of the material given a project that will engage them in the design activity".

Jansen and Dusch [30] describe a master course where "students are guided in their own development of a tiny microprocessor in 6 concluding tasks during no more than 10 weeks, using modern design tools". This approach provides "deep insight into actual computer architecture, how processors are working, the interdependence of hard- and software as well as into the software tool chain". Rashevska and Tkachuk [31] emphasize the practical applications of microprocessor control systems, their work demonstrates the crucial connection between theoretical microprocessor education and real-world industrial applications, underlining that "engineers must possess both theoretical knowledge and practical skills in implementing advanced control algorithms through microprocessor-based systems. Sin [32] discusses transitioning from traditional discrete microprocessors to soft processor cores, which provide "flexibility that was previously non-existent" through a graphical user interface to configure processors with desired peripherals.

Cross-platform development provides valuable experience with diverse technologies, thus He and Hsieh [33] describe teaching students "to build teaming robots by combining the Cortex controllers with ROBOTC programming environment under Windows and the Raspberry Pi (in ARM cores) using Python under Linux". This approach helps students "develop a cross-platform software and hardware design" across multiple programming environments. Industry-relevant curriculum updates ensure graduates possess needed skills. Radu and Dabacan [34] describe restructuring digital design courses "to cope with the increased demands of the industry" by placing "a strong emphasis in the study of modern tools, technologies and current industrial practices". Sheng [35] emphasises the importance of "continuously improving teaching methods by considering assessment criteria, students' course evaluation/feedback, and changes in the students group. This approach helped "prepare students for their future career by teaching them state-of-the-art tools and techniques". Wang et al. [36] note that design-build-test projects not only "provide an alternative assessment method for students who may not excel on written quizzes and exams" but also teach students valuable technical communication skills.

Digital competence frameworks [37, 38] provide structured approaches to developing necessary skills for modern engineers. Lyngdorf et al. [39] developed a matrix to embed digital competences into engineering curricula by "reviewing existing frameworks of digital literacy and competency and modifying them to suit the context of problem- and project-based learning in engineering education". This matrix includes "three categories of digital competences (general academic competences, problem-based learning competences, and discipline-specific competences) and an interdependent taxonomy of digital competences (user competences, development competences and reflexive competences)". Hoefele [40] argues that the European Digital Competence Framework (DigComp) needs to be adapted for engineering education contexts: "It demands validation and implementation considering regional, local, sectoral and education-specific circumstances and should respond to needs at the level of practice. Thus, it is necessary to adapt the existing digital competency frameworks to the specific circumstances of engineering education and to validate them. Sánchez et al. [41] applied the DigComp framework in an Engineering in Industrial Organization degree programme "to disseminate the DigComp structure and content and to give visibility to the work and acquisition of Digital Competences directly related to this academic context". Różewski et al. [42] matched DigComp with the Framework of Visual Literacy Competences for Engineering Education, illustrating its application in various educational RD projects. This integration provides a more comprehensive approach to developing digital competencies among engineering students.

The importance of continuous professional development and lifelong learning [43] in the field of microprocessor systems design has been emphasized by Papadakis et al. [44], who discuss how information and communication technologies can advance these areas. This perspective aligns with the need for engineering education to prepare students for continuous adaptation to evolving technologies and industry requirements. For future engineers, Soloshych et al. [45] implemented a "competence and resource-oriented approach to the development of digital educational resources for the formation of digital competence". This approach provides for the creation of a set of different types of digital educational resources that encourage teachers to improve the content of these digital educational resources in combination with software applications and digital technologies. Draghici et al. [46] describe a collaborative initiative involving European universities to develop a training program for students and extend their digital competencies.

Research indicates that implementing appropriate educational tools and approaches leads to significant benefits for engineering students. Well-designed educational tools increase student engagement and competence in embedded systems. Nakkar [47] found that the implementation of design-based tutorials on a System-on-Chip platform showed "an increase in student knowledge about FPGA, embedded systems, SoC, and software-hardware co-design". Thus, we analysed several key insights for using educational tools for microprocessor systems design. As technologies continue to evolve, ongoing adaptation of educational tools and methodologies will remain essential for developing digitally competent engineering graduates ready to address complex real-world challenges.

Despite considerable advancements in microprocessor systems design education, several critical gaps remain in current pedagogical approaches that directly impact the objectives of this research. There exists a significant disparity between theoretical classroom instruction and industry requirements, particularly regarding the practical application of tools like MCU 8051 IDE, STC-ISP, and PROTEUS DESIGN. Current educational frameworks often fail to effectively integrate these tools to enhance students' digital competence according to European standards. Furthermore, existing digital competence frameworks, though providing structural guidance, often lack contextual adaptation for specialized engineering disciplines like microprocessor systems design and fail to account for how educational tools can bridge the gap between academic knowledge and professional practice requirements.

This research aims to address these gaps by analyzing the effectiveness of existing educational tools in teaching microprocessor systems design, evaluating how these tools enhance students' digital competence within European standards, and assessing their impact on students' technical skills and readiness for professional engineering practice.

2. Objectives of research

The specific of research objectives are as follows:

1. To analyse the effectiveness of existing educational tools (MCU 8051 IDE, STC-ISP and PROTEUS DESIGN) in teaching microprocessor systems design and addressing common challenges such as balancing theoretical knowledge with practical application.
2. To evaluate how the utilisation of these educational tools enhances students' digital competence within the framework of European Digital Competence standards adapted for engineering education contexts.
3. To assess the impact of standardised microcontroller platforms and simulation environments on students' technical skills, problem-solving abilities, and readiness for professional engineering practice in the field of microprocessor systems design.

3. Implementation methodology of educational tools for microprocessor systems design

This research employs a practice-oriented approach to developing digital competence among engineering students through the implementation of a comprehensive microprocessor systems learning environment. The methodology focuses on integrating theoretical knowledge with practical skills using specialised software tools and hardware platforms. The research was conducted within the framework of the course "Electronics and Microprocessor Technology" at Kryvyi Rih National University.

The educational process is structured around three primary software tools, each serving a distinct purpose in the microcontroller development lifecycle. The first tool is MCU 8051 IDE, an integrated development environment for Intel 8051-based microcontrollers that supports both C and assembly language programming. This environment includes a built-in simulator with extensive debugging capabilities including step-by-step execution, interrupt viewing tools, external memory and code memory viewing utilities, and simulation capabilities for peripheral electronic devices such as LEDs, LED displays, matrices, and LCD displays (figure 1).

The second essential tool is STC-ISP, a microcontroller programming utility that allows selection of microcontroller type, COM port configuration and baud rate adjustments, as well as program code selection and writing to microcontroller memory (figure 2).

The third component is PROTEUS DESIGN, an electronic circuit design automation software package. This package comprises ISIS, a program for synthesis and simulation of electronic circuits, and ARES, a program for PCB (printed circuit board) development. PROTEUS DESIGN offers simulation capabilities for programmable devices including microcontrollers, microprocessors, and DSPs (figure 3).

The learning framework follows a sequential process that mirrors professional microcontroller development workflows. Students begin with project creation and configuration by establishing new projects in the MCU 8051 IDE with appropriate configuration settings. They then move to program development and testing, where they develop programs in assembly language (as illustrated by the example program in the document) and test them through simulation.

The next phase involves step-by-step simulation, where the microcontroller operation is simulated in a detailed mode to understand the execution flow and identify potential issues. Following successful simulation, students proceed to hardware programming. In this stage, they configure the STC-ISP utility to open the compiled program file, connect to the A2 educational board, program the microcontroller, and verify program operation on physical hardware.

After verifying basic functionality, students engage in circuit design and simulation using PROTEUS DESIGN. During this phase, they create a project for a microcontroller circuit with LEDs, test the circuit operation through simulation, and apply the compiled program code from MCU 8051 IDE. The final

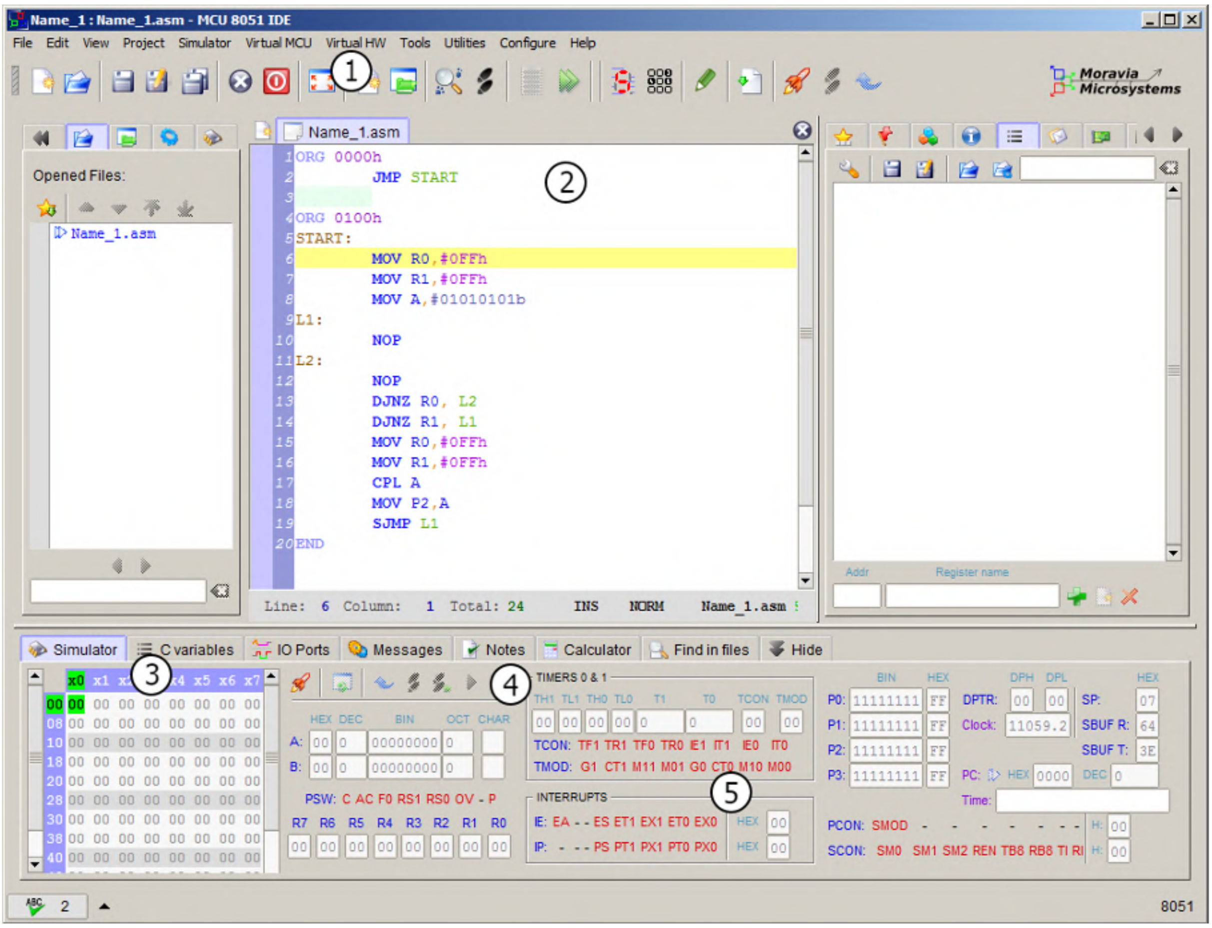

Figure 1: Screenshot of the MCU 8051 microcontroller programming utility.

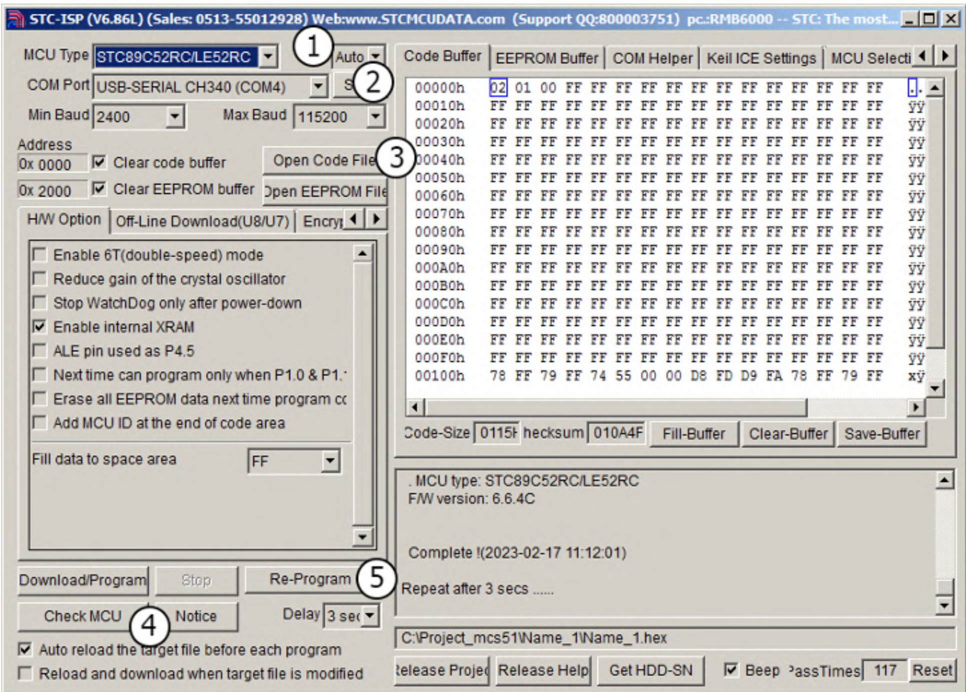

Figure 2: Screenshot of the STC-ISP microcontroller programming utility interface.

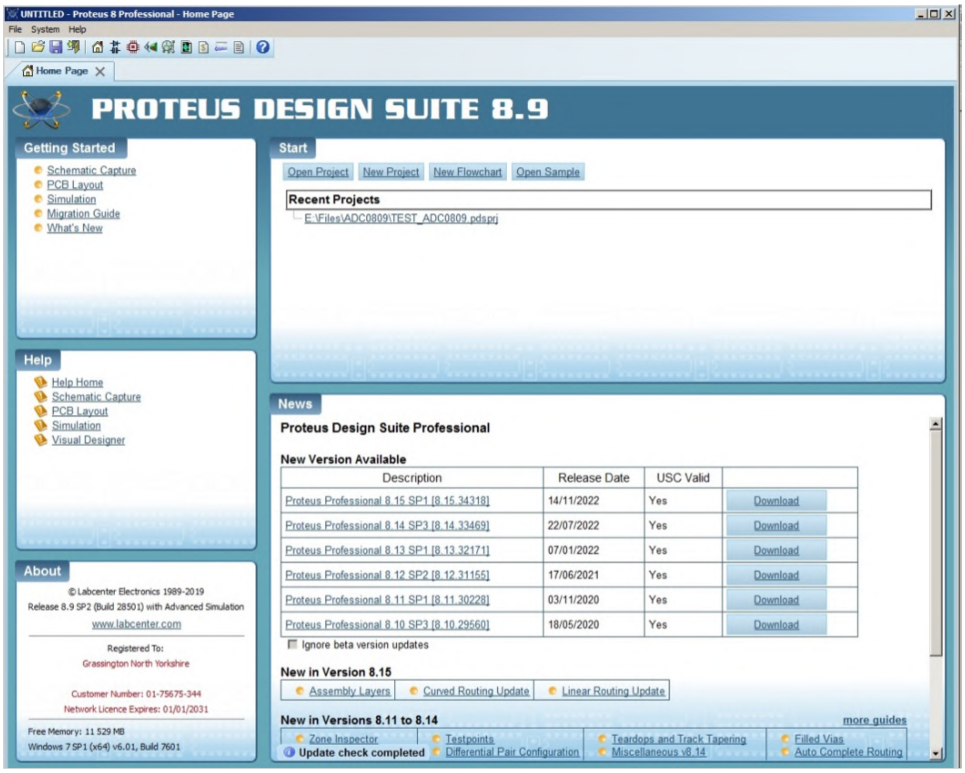

Figure 3: External view of the PROTEUS DESIGN software package interface.

stage involves documentation and reporting, where students document their work and create laboratory reports that reflect on their learning process and outcomes.

The methodology incorporates the A2 educational board as the primary hardware platform for practical implementation. This board enables students to test their programs on physical microcontrollers after simulation, providing a complete learning cycle from code development to real-world implementation.

The pedagogical approach emphasises progressive skill development, moving from basic programming concepts to complex circuit design. It embraces iterative learning through the use of simulation before hardware implementation to refine understanding. Hands-on experience bridges theoretical knowledge with practical application, while exposure to the complete development cycle familiarises students with all stages of microcontroller system development. Additionally, the approach fosters documentation skills through the development of technical communication abilities in the reporting phase.

The methodology targets key competencies identified in the curriculum:

- The ability to solve complex specialised tasks in electrical engineering or during the learning process, involving the application of theories and methods from physics and engineering sciences, characterised by complexity and uncertainty of conditions;
- The ability to apply knowledge in practical situations;
- The ability to search, process and analyse information from various sources;
- The ability to identify, formulate and solve problems;
- The ability to solve practical tasks using methods of mathematics, physics and electrical engineering;
- The ability to apply application software, microcontrollers and microprocessor technology to solve practical problems in professional activities.

This methodology creates a comprehensive learning environment that develops both technical skills specific to microcontroller programming and broader digital competencies applicable across engineering disciplines. To thoroughly analyse the effectiveness of MCU 8051 IDE, STC-ISP and PROTEUS DESIGN in teaching microprocessor systems design, the following seven criteria have been developed (table 1).

Table 1
Evaluation criteria for educational tools in microprocessor systems design.

№	Criteria	Interpretation
1	Simulation fidelity	The accuracy and comprehensiveness with which the tool simulates real microprocessor behaviour, including timing, interrupts, and peripheral interactions
2	User interface accessibility	The intuitiveness of the interface, ease of navigation, and learning curve for new users, particularly engineering students with varying levels of prior experience
3	Debugging capabilities	The range and effectiveness of debugging features, including step-by-step execution, register/memory monitoring, and breakpoint functionality
4	Hardware integration	The ability to interface with physical hardware components, supporting the transition from simulation to real-world implementation
5	Comprehensive documentation and learning resources	The availability and quality of documentation, tutorials, examples, and other learning materials that support independent learning
6	Cross-platform compatibility	The ability to operate across different operating systems and hardware configurations, ensuring accessibility for all students
7	Industry relevance	The extent to which skills developed using the tool are transferable to industry contexts and align with current professional practices

Based on the established evaluation criteria, a comprehensive comparative analysis of educational tools for microprocessor systems design has been developed (table 2). This analysis serves to provide educators and institutions with evidence-based insights for selecting appropriate tools that enhance the teaching and learning experience in microprocessor education.

The comparative table shows that each tool has distinct strengths: MCU 8051 IDE excels in simulation fidelity, debugging, and cross-platform compatibility; STC-ISP is strongest in user interface accessibility, hardware integration, and industry relevance; PROTEUS DESIGN performs best in simulation fidelity, documentation, and industry relevance. This comparative analysis provides a structured evaluation of each tool's strengths and limitations across the seven key criteria, enabling informed decisions about their implementation in microprocessor systems design education (figure 4, 5).

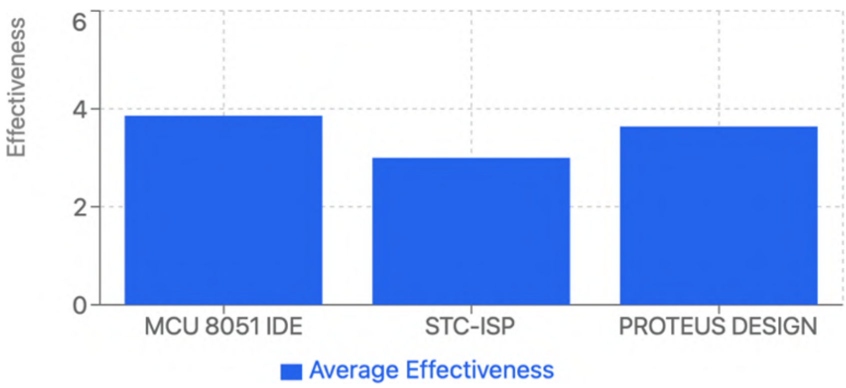

Figure 4: Average effectiveness of educational tools for microprocessor design.

Key findings from effectiveness analysis show:

- MCU 8051 IDE (average: 3.86) performs consistently across most criteria with notable strengths in simulation and debugging.
- STC-ISP (average: 3) excels in hardware integration and interface accessibility but lacks in simulation capabilities.

Table 2
Comparative analysis.

Criteria	MCU 8051 IDE	STC-ISP	PROTEUS DESIGN
Simulation fidelity	High – Accurate CPU simulation with comprehensive register and memory visualisation	Low – Not primarily a simulation tool, focused on programming only	Very High – Complex circuit simulation with accurate component modelling
User interface accessibility	Medium – Relatively intuitive but dated interface with moderate learning curve	High – Simple, focused interface with clear workflow	Low – Complex interface with steep learning curve requiring significant prior knowledge
Debugging capabilities	High – Step-by-step execution, register monitoring, memory inspection, and breakpoints	Low – Limited to basic verification of programming	High – Circuit-level debugging with signal tracing and component state monitoring
Hardware integration	Medium – Supports programming real hardware but requires additional tools	High – Direct hardware programming capabilities with good device support	Low – Primarily simulation-focused with limited direct hardware integration
Comprehensive documentation	Medium – Adequate documentation but limited advanced tutorials	Low – Basic operational documentation with few learning resources	High – Extensive documentation, tutorials, and community resources
Cross-platform compatibility	High – Works across Windows, Linux, and macOS	Medium – Primarily Windows-focused with limited support for other platforms	Medium – Best on Windows with partial support for other platforms
Industry relevance	Medium – Teaches fundamental concepts but uses somewhat outdated interfaces	High – Uses actual programming tools similar to industry practice	High – Simulation approach mirrors industry design workflows

- PROTEUS DESIGN (average: 3.64) shows strengths in simulation fidelity and documentation but has limited hardware integration.

Overall effectiveness: MCU 8051 IDE has the highest average effectiveness, followed by PROTEUS DESIGN, and STC-ISP.

The radar chart in provides a more nuanced comparison across seven specific criteria, revealing the unique strengths and weaknesses of each tool:

- MCU 8051 IDE: demonstrates balanced performance across most evaluation criteria; shows particular strengths in simulation fidelity and debugging capabilities; maintains solid performance in cross-platform compatibility; shows moderate performance in documentation and resources;
- STC-ISP: excels significantly in hardware integratio; shows good performance in user interface accessibility; demonstrates notable weakness in simulation fidelity; has limited cross-platform compatibility; shows moderate performance in debugging capabilities;
- PROTEUS DESIGN: demonstrates excellent simulation fidelity (highest score in this category); provides strong documentation and resources; shows good industry relevance; has limited hardware integration capabilities; maintains reasonable user interface accessibility.

These findings suggest important considerations for implementing these tools in microprocessor systems design education:

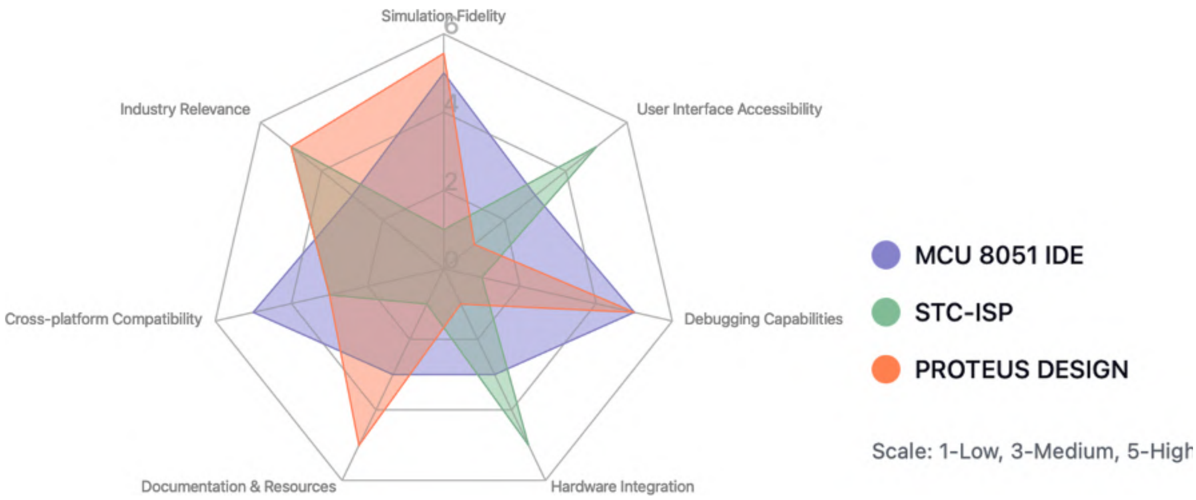

Figure 5: Criteria-based effectiveness comparison.

- For comprehensive learning covering both simulation and practical implementation, MCU 8051 IDE offers the most balanced approach;
- For courses emphasizing hardware interaction and real-world implementation, STC-ISP provides superior hardware integration;
- For detailed simulation-focused learning with strong documentation support, PROTEUS DESIGN would be most effective.

4. Experimental assessment of educational tools' effectiveness in developing engineering students' digital competences

The study employed a comparative analysis between an experimental group (26 students) utilising the new educational technologies and a control group (26 students) following traditional teaching methods.

The experimental group's curriculum incorporated: hardware-software co-design platforms; project-based learning scenarios; industry-aligned problem-solving tasks; collaborative design challenges; self-directed learning components; simulation and physical prototyping tools, authentic assessment methods. Student performance was evaluated through: standardised examinations; project assessments; digital competence rubrics; self-efficacy surveys; industry-readiness evaluations.

For a well-substantiated analysis of the effectiveness of implementing educational tools in the teaching process of the microprocessor system, a comparative study of academic performance between the experimental and control groups of students was conducted (table 3).

Table 3
Comparison of academic performance metrics.

Performance metric	Experimental group	Control group	Improvement (%)
Average score	72.87	61.94	17.65%
Pass rate	100.00%	85.00%	17.65%
Excellent rate	26.09%	18.26%	42.86%
Good rate	26.09%	22.17%	17.68%
Satisfactory rate	47.83%	52.61%	-9.09%*

Note: The lower percentage in the satisfactory rate category for the experimental group reflects a positive outcome, as more students achieved higher grades.

The experimental group demonstrated a balanced distribution across grade categories: A (excellent):

6 students (26.09%); B (very good): 2 students (8.70%); C (good): 4 students (17.39%); D (satisfactory): 6 students (26.09%); E (sufficient): 5 students (21.74%). The experimental group demonstrated substantial improvements in academic performance compared to the control group. The most notable improvement was in the proportion of students achieving excellent grades (42.86% increase), suggesting that the implemented tools were particularly effective in supporting high-level learning outcomes. The 100% pass rate in the experimental group further demonstrates the effectiveness of the approach in supporting all learners. In addition to improving academic performance, an important aspect of the research was the assessment of students' digital competence development in accordance with European standards. For an objective measurement of digital skills proficiency levels, an evaluation was conducted using a 10-point scale across various competence areas. The results of this assessment are presented in the table below, which reflects significant differences between the experimental group that studied using specialised educational tools and the control group (table 4).

Table 4
Digital competence ratings.

Competence area	Experimental group	Control group	Improvement (%)
Problem solving	8.7	7.1	22.54%
Technical proficiency	8.5	6.8	25.00%
Independent learning	8.2	6.4	28.12%
Collaboration	8.9	7.3	21.92%
Industry readiness	8.3	6.5	27.69%

All measured areas of digital competence showed significant improvement, with independent learning skills showing the most substantial gains (28.12%). This suggests that the new educational tools successfully foster self-directed learning capabilities. The high ratings for industry readiness (8.3/10) indicate that the approach effectively bridges the gap between academic learning and professional practice. The experimental implementation of innovative educational tools for microprocessor systems design yielded: significantly higher academic performance (17.65% increase in average scores); enhanced digital competence across all measured domains. This study provides compelling evidence that well-designed educational tools for microprocessor systems design can significantly enhance digital competence development among engineering students, with measurable improvements in both academic performance and industry-relevant skills.

5. Conclusions

This research demonstrates that a comprehensive approach to microprocessor systems design education, integrating specialised software tools and hardware platforms, significantly enhances digital competence development among engineering students. The comparative analysis of MCU 8051 IDE, STC-ISP, and PROTEUS DESIGN revealed distinct complementary strengths, supporting the implementation of a multi-tool learning environment that mirrors professional workflows.

The quasi-experimental study provided compelling evidence of the effectiveness of this approach, with the experimental group demonstrating marked improvements across all measured metrics. The 17.65% increase in average academic performance, combined with a 100% pass rate and 42.86% improvement in excellent grades, validates the pedagogical framework developed in this research.

Particularly significant was the development of digital competencies aligned with industry requirements. The substantial improvements in problem-solving (22.54%), technical proficiency (25%), independent learning (28.12%), collaboration (21.92%), and industry readiness (27.69%) suggest that the implemented methodology successfully bridges the gap between academic knowledge and professional practice.

Declaration on Generative AI

AI served as an assistant in research development, specifically for editing and translation purposes. All theoretical constructs, data interpretations, and conclusions were developed by researchers, ensuring that AI enhanced rather than replaced human expertise in the research process.

References

[1] M. Velykodna, Psychoanalysis during the COVID-19 pandemic: Several reflections on countertransference, Psychodynamic Practice 27 (2021) 10–28. doi:10.1080/14753634.2020.1863251.

[2] O. Lushchak, M. Velykodna, S. Bolman, O. Strilbytska, V. Berezovskyi, K. B. Storey, Prevalence of stress, anxiety, and symptoms of post-traumatic stress disorder among Ukrainians after the first year of Russian invasion: a nationwide cross-sectional study, The Lancet Regional Health - Europe 36 (2024) 100773. doi:10.1016/j.lanepe.2023.100773.

[3] F. G. Pellicano, M.-I. Carnascialli, C. Martinez, Establishing Baseline Data on Student Success in Embedded Systems Education, in: 2022 IEEE Frontiers in Education Conference (FIE), 2022, pp. 1–4. doi:10.1109/FIE56618.2022.9962525.

[4] V. Shebanin, I. Atamanyuk, Y. Kondratenko, Y. Volosyuk, Development of the Mathematical Model of the Informational Resource of a Distance Learning System, in: O. Chertov, T. Mylovanov, Y. Kondratenko, J. Kacprzyk, V. Kreinovich, V. Stefanuk (Eds.), Recent Developments in Data Science and Intelligent Analysis of Information, Springer International Publishing, Cham, 2019, pp. 199–205. doi:10.1007/978-3-319-97885-7_20.

[5] V. Shebanin, I. Atamanyuk, Y. Kondratenko, Y. Volosyuk, Application of fuzzy predicates and quantifiers by matrix presentation in informational resources modeling, in: 2016 XII International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH), 2016, pp. 146–149. doi:10.1109/MEMSTECH.2016.7507536.

[6] P. E. Hertzog, A. J. Swart, Arduino — Enabling engineering students to obtain academic success in a design-based module, in: 2016 IEEE Global Engineering Education Conference (EDUCON), 2016, pp. 66–73. doi:10.1109/EDUCON.2016.7474533.

[7] M. Solesvik, O. J. Borch, Y. Kondratenko, Joint Digital Simulation Platforms for Safety and Preparedness, in: Y. Luo (Ed.), Cooperative Design, Visualization, and Engineering, Springer International Publishing, Cham, 2018, pp. 118–125. doi:10.1007/978-3-030-00560-3_16.

[8] X. Zhang, Resilient Course Design for Teaching a Project-based Engineering Course Online, in: ASEE Annual Conference and Exposition, Conference Proceedings, 2021. URL: https://par.nsf.gov/servlets/purl/10320028.

[9] M. Deepa, P. Reba, G. Santhanamari, N. Susithra, Enriched blended learning through virtual experience in microprocessors and microcontrollers course, Journal of Engineering Education Transformations 34 (2021) 642–650. doi:10.16920/jeet/2021/v34i0/157236.

[10] N. Kostadinov, N. Bencheva, Educational Tools for Processor Design Classes: A Case Study, in: 2023 32nd Annual Conference of the European Association for Education in Electrical and Information Engineering (EAEEIE), 2023, pp. 1–4. doi:10.23919/EAEEIE55804.2023.10182179.

[11] E. P. Ferlin, V. Pilla, Microprocessors: from theory to practice, a didactic experience, in: 34th Annual Frontiers in Education, 2004. FIE 2004., volume 3, 2004, pp. S1D/4–S1D/7. doi:10.1109/FIE.2004.1408687.

[12] A. Dabroom, W. M. Refie, R. Matmti, Microcontroller-based learning kit: Course design using constructive alignment principles, in: 21st Mediterranean Conference on Control and Automation, 2013, pp. 558–566. doi:10.1109/MED.2013.6608777.

[13] C. Parikh, Autonomous robot: An intellectual way of infusing microcontroller fundamentals into sophomore students, in: 2015 IEEE Frontiers in Education Conference (FIE), 2015, pp. 1–4. doi:10.1109/FIE.2015.7344261.

[14] M. Deepa, N. Susithra, G. Santhanamari, P. Reba, Leveraging Agile Framework for a Project Based

Learning Environment in Embedded Systems Design Course, Journal of Engineering Education Transformations 37 (2024) 517–522. URL: https://www.journaleet.in/index.php/jeet/article/view/2464.
[15] S. Brown, Leveraging FPGA Technology for Digital Logic and Embedded Systems Education, in: 2007 IEEE International Conference on Microelectronic Systems Education (MSE'07), 2007, pp. 159–159. doi:10.1109/MSE.2007.61.
[16] Z. Bakum, V. Tkachuk, Mining engineers training in context of innovative system of Ukraine, Metallurgical and Mining Industry 6 (2014) 29–34.
[17] Y. P. Kondratenko, V. I. Slyusar, M. B. Solesvik, N. Y. Kondratenko, Z. Gomolka, Interrelation and inter-influence of artificial intelligence and higher education systems, in: Y. P. Kondratenko, A. I. Shevchenko (Eds.), Research Tendencies and Prospect Domains for AI Development and Implementation, River Publishers, New York, 2024, pp. 31–58.
[18] Y. P. Kondratenko, D. Simon, I. P. Atamanyuk, University Curricula Modification Based on Advancements in Information and Communication Technologies, in: V. Ermolayev, A. Spivakovsky, M. S. Nikitchenko, A. Ginige, H. C. Mayr, D. Plexousakis, G. Zholtkevych, O. Burov, V. S. Kharchenko, V. Kobets (Eds.), Proceedings of the 12th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer, Kyiv, Ukraine, June 21-24, 2016, volume 1614 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2016, pp. 184–199. URL: https://ceur-ws.org/Vol-1614/paper_18.pdf.
[19] O. V. Talaver, T. A. Vakaliuk, A model for improving the accuracy of educational content created by generative AI, in: S. O. Semerikov, A. M. Striuk, M. V. Marienko, O. P. Pinchuk (Eds.), Proceedings of the 7th International Workshop on Augmented Reality in Education (AREdu 2024), Kryvyi Rih, Ukraine, May 14, 2024, volume 3918 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 149–158. URL: https://ceur-ws.org/Vol-3918/paper372.pdf.
[20] K. Bondar, O. S. Bilozir, O. Shestopalova, V. A. Hamaniuk, Bridging minds and machines: AI's role in enhancing mental health and productivity amidst Ukraine's challenges, in: S. O. Semerikov, A. M. Striuk, M. V. Marienko, O. P. Pinchuk (Eds.), Proceedings of the 7th International Workshop on Augmented Reality in Education (AREdu 2024), Kryvyi Rih, Ukraine, May 14, 2024, volume 3918 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 43–59. URL: https://ceur-ws.org/Vol-3918/paper020.pdf.
[21] O. M. Haranin, N. V. Moiseienko, Adaptive artificial intelligence in RPG-game on the Unity game engine, in: A. E. Kiv, S. O. Semerikov, V. N. Soloviev, A. M. Striuk (Eds.), Proceedings of the 1st Student Workshop on Computer Science & Software Engineering, Kryvyi Rih, Ukraine, November 30, 2018, volume 2292 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018, pp. 143–150. URL: https://ceur-ws.org/Vol-2292/paper16.pdf.
[22] O. V. Korotun, T. A. Vakaliuk, A. M. Makhno, Tools for Teaching the R Programming Language to Bachelors of Computer Science in the Period of Distance Learning, in: E. Smyrnova-Trybulska, N.-S. Chen, P. Kommers, N. Morze (Eds.), E-Learning and Enhancing Soft Skills: Contemporary Models of Education in the Era of Artificial Intelligence, Springer Nature Switzerland, Cham, 2025, pp. 309–330. doi:10.1007/978-3-031-82243-8_18.
[23] K. P. Osadcha, N. V. Shumeiko, Artificial intelligence, the labor market, and education for sustainable development: the points of intersection, IOP Conference Series: Earth and Environmental Science 1415 (2024) 012015. doi:10.1088/1755-1315/1415/1/012015.
[24] N. V. Shumeiko, K. P. Osadcha, Application of artificial intelligence in higher education institutions for developing soft skills of future specialists in the sphere of information technology, Journal of Physics: Conference Series 2871 (2024) 012027. doi:10.1088/1742-6596/2871/1/012027.
[25] O. Y. Tarasova, V. S. Doroshko, Methodological foundations of teaching the basics of artificial intelligence to lyceum students, in: S. O. Semerikov, A. M. Striuk, M. V. Marienko, O. P. Pinchuk (Eds.), Proceedings of the 7th International Workshop on Augmented Reality in Education (AREdu 2024), Kryvyi Rih, Ukraine, May 14, 2024, volume 3918 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 240–249. URL: https://ceur-ws.org/Vol-3918/paper184.pdf.
[26] V. Tkachuk, Y. Yechkalo, D. Brovko, W. Sobczyk, Augmented and Virtual Reality Tools in Training

Mining Engineers, Inzynieria Mineralna (2023) 137–146. doi:10.29227/IM-2023-01-17.
[27] R. A. Metri, V. R. S. V. Bharathpulavarthi, I. Srikanth, C. L. Bhattar, Microcontroller Laboratory Practices through Project-Based Learning , Journal of Engineering Education Transformations 31 (2018) 169–178. doi:10.16920/jeet/2018/v31i3/120788.
[28] S. J. Schneider, S. J. Peterson, Utilizing a system-on-chip project as a capstone experience in a microprocessors course, in: ASEE Annual Conference and Exposition, Conference Proceedings, 2012. URL: https://peer.asee.org/utilizing-a-system-on-chip-project-as-a-capstone-experience-in-a-microprocessors-course.pdf.
[29] L. McLauchlan, Design-oriented course in microprocessor-based controls, in: ASEE Annual Conference and Exposition, Conference Proceedings, 2007. URL: https://strategy.asee.org/design-oriented-course-in-microprocessor-based-controls.pdf.
[30] D. Jansen, B. Dusch, Every student makes his own microprocessor, in: 10th European Workshop on Microelectronics Education (EWME), 2014, pp. 97–101. doi:10.1109/EWME.2014.6877404.
[31] N. Rashevska, V. Tkachuk, Technological conditions of mobile learning at high school, Metallurgical and Mining Industry 7 (2015) 161–164. URL: https://www.metaljournal.com.ua/assets/Journal/english-edition/MMI_2015_3/021%20Rashevska.pdf.
[32] M. L. Sin, On the use of a soft processor core in computer engineering education, in: ASEE Annual Conference and Exposition, Conference Proceedings, 2006. URL: https://peer.asee.org/on-the-use-of-a-soft-processor-core-in-computer-engineering-education.pdf.
[33] S. He, S.-J. Hsieh, Multi-sensors for robot teaming using raspberry Pi and VEX robotics construction kit, in: ASEE Annual Conference and Exposition, Conference Proceedings, 2018. URL: https://peer.asee.org/multi-sensors-for-robot-teaming-using-raspberry-pi-and-vex-robotics-construction-kit.pdf.
[34] M. Radu, M. A. Dabacan, Restructuring digital design courses in electrical and computer engineering technology programs, preparing the engineer of 2020, in: ASEE Annual Conference and Exposition, Conference Proceedings, 2016. URL: https://tinyurl.com/4rju9e7w.
[35] J. Sheng, Continuous improvement in teaching microprocessor systems design a review of efforts in using different tools, techniques and methods to satisfy students' needs, in: ASEE Annual Conference and Exposition, Conference Proceedings, 2020. URL: https://monolith.asee.org/public/conferences/172/papers/29716/download.
[36] A. Wang, L. Van Den Einde, N. Delson, Gotta Catch'Em All: Learning Graphical Communications through an Introductory Hands-on Design-Build-Test Project in a Hybrid Learning Environment, in: ASEE Annual Conference and Exposition, Conference Proceedings, 2023. URL: https://tinyurl.com/bdea94ue.
[37] I. V. Lovianova, N. Y. Hrebin-Krushelnytska, R. Y. Kaluhin, A. V. Krasnoshchok, O. O. Kozhukhar, Formation of digital competence of specialists in socionomic professions as a pedagogical problem, in: S. Papadakis (Ed.), Proceedings of the 11th Workshop on Cloud Technologies in Education (CTE 2023), Kryvyi Rih, Ukraine, December 22, 2023, volume 3679 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 209–223. URL: https://ceur-ws.org/Vol-3679/paper40.pdf.
[38] M. V. Moiseienko, N. V. Moiseienko, O. O. Lavrentieva, Developing pre-service teachers' digital competence through informatics disciplines in teacher education programs, in: S. O. Semerikov, A. M. Striuk (Eds.), Proceedings of the 6th International Workshop on Augmented Reality in Education (AREdu 2023), Kryvyi Rih, Ukraine, May 17, 2023, volume 3844 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 45–52. URL: https://ceur-ws.org/Vol-3844/paper11.pdf.
[39] N. E. Lyngdorf, L. B. Bertel, E. Lindsay, A Matrix for Making Sense of Digital Competences in Formal Engineering Education Curricula, in: SEFI 2022 - 50th Annual Conference of the European Society for Engineering Education, Proceedings, 2022, pp. 2041–2046. URL: https://vbn.aau.dk/en/publications/a-matrix-for-making-sense-of-digital-competences-in-formal-engine.
[40] J. Hoefele, European Digital Competence Frameworks and Engineering Pedagogy, in: M. E. Auer, H. Hortsch, O. Michler, T. Köhler (Eds.), Mobility for Smart Cities and Regional Development - Challenges for Higher Education, Springer International Publishing, Cham, 2022, pp. 800–811. doi:10.1007/978-3-030-93907-6_87.

[41] A. L. E. Sánchez, L. M. E. Escaño, M. O. Carmona, M. R. Andrés, J. B. Mora, C. B. Olave, Digital Competences. The challenge of their acquisition and verification in the University Degree, in: 2022 Congreso de Tecnología, Aprendizaje y Enseñanza de la Electrónica (XV Technologies Applied to Electronics Teaching Conference), 2022, pp. 1–6. doi:10.1109/TAEE54169.2022.9840546.

[42] P. Różewski, M. Kieruzel, T. Lipczyński, M. Prys, Framework of Visual Literacy Competences for Engineering Education Discussed in the Scope of DigComp framework with Examples from Educational R&D Projects, Procedia Computer Science 192 (2021) 4441–4447. doi:10.1016/j.procs.2021.09.221.

[43] O. Chaika, N. Sharmanova, Innovative efficiency models in foreign language communication: Lifelong learning and multiculturalism, in: Lifelong Learning: Models and Methods of Implementation, PC TECHNOLOGY CENTER, 2023, pp. 59–88. doi:10.15587/978-617-7319-70-1.CH3.

[44] S. Papadakis, S. O. Semerikov, Y. V. Yechkalo, V. Y. Velychko, T. A. Vakaliuk, S. M. Amelina, A. V. Iatsyshyn, M. V. Marienko, S. M. Hryshchenko, V. V. Tkachuk, Advancing lifelong learning and professional development through ICT: insights from the 3L-Person 2023 workshop, in: S. Papadakis (Ed.), Proceedings of the VIII International Workshop on Professional Retraining and Life-Long Learning using ICT: Person-oriented Approach (3L-Person 2023), Virtual Event, Kryvyi Rih, Ukraine, October 25, 2023, volume 3535 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 1–16. URL: https://ceur-ws.org/Vol-3535/paper00.pdf.

[45] I. Soloshych, M. Grynova, N. Kononets, I. Shvedchykova, I. Bunetska, Competence and Resource-Oriented Approaches to the Development of Digital Educational Resources, in: 2021 IEEE International Conference on Modern Electrical and Energy Systems (MEES), 2021, pp. 1–5. doi:10.1109/MEES52427.2021.9598603.

[46] A. Draghici, D. Fatol, L. Ivascu, M. Mocofan, Multimedia Skills Development to Support Engineering Studies, in: C.-Y. Huang, R. Dekkers, S. F. Chiu, D. Popescu, L. Quezada (Eds.), Intelligent and Transformative Production in Pandemic Times, Springer International Publishing, Cham, 2023, pp. 831–840. doi:10.1007/978-3-031-18641-7_77.

[47] M. Nakkar, Design based tutorials for System-on-Chip teachings, in: 2009 International Conference on Engineering Education (ICEED), 2009, pp. 117–121. doi:10.1109/ICEED.2009.5490602.