DOI: https://doi.org/10.25140/2411-5363-2025-3(41)-401-409
UDC 681.5:004.77

Volodymyr Voytenko[1], Alexander Pisarevskiy[2]

[1]PhD in Technical Sciences, Docent, Associate Professor of Electronics, Automation, Robotics and Mechatronics Department
Chernihiv Polytechnic National University (Chernihiv, Ukraine);
researcher at the Department of Automatic Control
Lund University (Lund, Sweden)
E-mail: v.voytenko@stu.cn.ua. **ORCID:** http://orcid.org/0000-0003-1490-0600
ResearcherID: F-8698-2014. **Scopus Author ID:** 36167678700

[2]Research at the Department of Automatic Control
Lund University (Lund, Sweden)
E-mail: alexander.pisarevskiy@control.lth.se. **ORCID:** https://orcid.org/0009-0004-0019-1502

REMOTE DESIGN OF SAFETY CONTROLLERS FOR INDUSTRIAL AUTOMATION SYSTEMS

PLCs are the core of most industrial automation systems (IAS). Modern IAS development emphasizes hardware unification and increased software significance. Remote PLC programming boosts efficiency by reducing time and travel costs, while supporting advanced training. Using safety controllers in the example of IAS, this paper presents hardware and PLC-specific software tools enabling remote PLC programming and IAS monitoring. The results are applicable in IAS development and engineering education.

Keywords: *industrial automation systems (IAS); programmable logic controllers (PLC); safety controllers; PLC software; remote PLC programming; remote engineering education; microprocessor.*

Fig.: 6. References: 27.

Urgency of the research. The current stage of industrial development is characterized by the widespread use of automation technologies, which ensure high quality of mass-produced end-products while simultaneously reducing production costs. This automation largely relies on programmable logic controllers (PLCs) [1, 2], which, through associated input and output devices, integrate a variety of field equipment, including switches, sensors, converters, relays, motors and other devices. Modern PLCs [3, 4], in contrast to earlier relay-based controllers, are based on microprocessor technologies. This allows for the use of a unified structural framework to control diverse industrial processes, with customization for specific applications achieved through modifications to the software executed by the microprocessor.

In 2024, the global market for PLCs was valued at approximately USD 13 billion, representing a 7% increase compared to the previous year [5]. Consequently, PLC programming remains a crucial aspect of developing and configuring automation systems, particularly in sector, including: industrial manufacturing, automotive engineering, pharmaceuticals, and energy.

Target setting. Manufacturers of PLCs typically provide proprietary software tools for programming their controllers, which generally support the programming languages defined by the standard [6]. Most PLC software development environments follow a similar approach: a computer running the specific development tool is connected to the PLC via a programming interface. The application, developed and debugged in simulation mode, is then uploaded to the PLC. Afterward, an interactive process of testing and iterative refinement begins directly within the industrial automation system. This allows for verification of functionality and performance during interaction with the intended field devices.

Testing the PLC program within the specific configuration of field devices, including the length and characteristics of connection lines, power supply sources, and operational conditions, enables an accurate assessment of critical timing parameters, detection of developer errors, identification of equipment incompatibilities, and an overall evaluation of the industrial automation system's operability. If the outcome of this testing is unsatisfactory, it becomes necessary to return to the PLC software development environment and refine the program accordingly.

The challenge becomes even more critical when developing the safety system, a crucial component of any modern industrial automation system. In this context, numerous regulatory requirements must be met, typically achieved through the integration of additional PLCs, known as safety controllers. Industrial automation systems incorporating such safety controllers rely on specialized hardware and software tools to enable certification by stringent and comprehensive safety standards.

The physical separation of certain industrial automation system components from the PLC, the need to improve production technologies and/or safety systems, as well as reconfiguration for different product types, often necessitate regular updates to PLC programs. This, in turn, requires the frequent presence of developers on-site, which is economically inefficient.

This issue can be addressed by enabling remote access capabilities, both during the initial development and throughout the maintenance and operation phases of the industrial automation system. Such an approach can reduce labor costs associated with in-house developers or travel expenses for external specialists, while also minimizing their downtime.

Moreover, the ability to remotely program and verify PLC software opens new opportunities in education and the training of highly qualified specialists. It allows for reduction in the amount of expensive laboratory equipment by enabling individual students to access it at convenient times, facilitates resource sharing among educational institutions, enhances the specialization of laboratories and the quality of instruction, and improves safety in the context of pandemics and other contemporary disruptions.

Actual scientific research and issues analysis. Due to the widespread PLCs adoptionof and the growing demand for specialists capable of working with them, as well as the increasing relevance of remote education, research in this field has been ongoing for quite some time. For instance, [7] the concept and architecture of an automation laboratory that enables users to remotely connect to a PLC is offered. The software for direct interaction with PLCs, referred to as PLC-specific software (PSS), is installed exclusively on the host computer, while the remote user's computer functions as an authorized client accessing the PSS. The network connection is established using Cisco VPN Client and Cisco VPN Server through a secure VPN tunnel.

In [8], a solution for remote monitoring of mechatronic devices is proposed by integrating PLCs with the Advantech WebAccess/SCADA software. In [9] and [10], PLCs are employed as intermediary components for collecting data from remote industrial process sensors and subsequently transmitting this information to end users.

Uninvestigated aspects of general problem definition. The main drawbacks of the aforementioned and other known solutions lie in the fact that the IAS developer working with PLCs does not interact directly with the instrumental PSS and lacks access to the full range of features provided by the programming environment. Special attention should be given to safety controllers (e.g., [11]), which are similar to conventional PLCs but possess specific characteristics due to their intended purpose. These specifics must be strictly observed during IAS development and are supported by dedicated PSS tailored for such controllers [12].

In the education context, it should be noted that when students rely on various intermediate or auxiliary software tools, they cannot fully acquire practical skills in working directly with a specific PSS. This includes essential steps, including locating the software, reviewing its documentation on the manufacturer's website, downloading, installing, and configuring it. As a result, future challenges may arise, i.e. difficulties in mastering new PLCs and PSS platforms or adapting to the continuous changes inherent in both this domain and computer technologies in general.

The research objective. The objective of this study is to develop an architecture for a hardware-software system that enables a safety subsystem developer of an industrial automation system to perform, from their local workstation equipped with the original instrumental PLC-specific software, any actions supported by the safety PLC manufacturer for program development, as well as to verify the results on a real remote IAS.

To achieve the stated objective, it is necessary to develop a generalized structure of the hardware components and connection schemes for individual elements, using specific predefined models of PLCs, field devices, and auxiliary components. Based on the obtained results, appropriate software tools must be selected, configured, refined, and adjusted to ensure the functionality of the proposed structure. Finally, a comprehensive verification of the developed architecture should be carried out, the results analyzed, and a plan for further actions formulated.

Achieving the stated objective will yield positive outcomes across various areas related to interaction with PLC-based industrial automation systems, including:

- Reducing working time and other expenses associated with the activities of IAS developers.
- Minimizing the amount of laboratory equipment required for training IAS developers.
- Enabling interaction with IAS at a convenient and safe time.

The statement of basic materials

Structure and hardware components of the PLC-based system under study. The structure of the PLC-based system under investigation is shown in Fig. 1. The system is built using components from SICK [11], specifically the *SICK Safety Controller FX3-CPU000000* (denoted as Main PLC Module in the figure) [13], which serves as a main control and interfacing module. Directly connected to it are the *FX3-GEPR00000* (Ethernet Gateway) [14] and the *FX3-XTIO84002* (Input/Output Module) [15], both of which can be configured via software. Additionally, the system includes the *UE410-4RO4* (Relay Module) [16], which can only be controlled through hardware using additional wired connections.

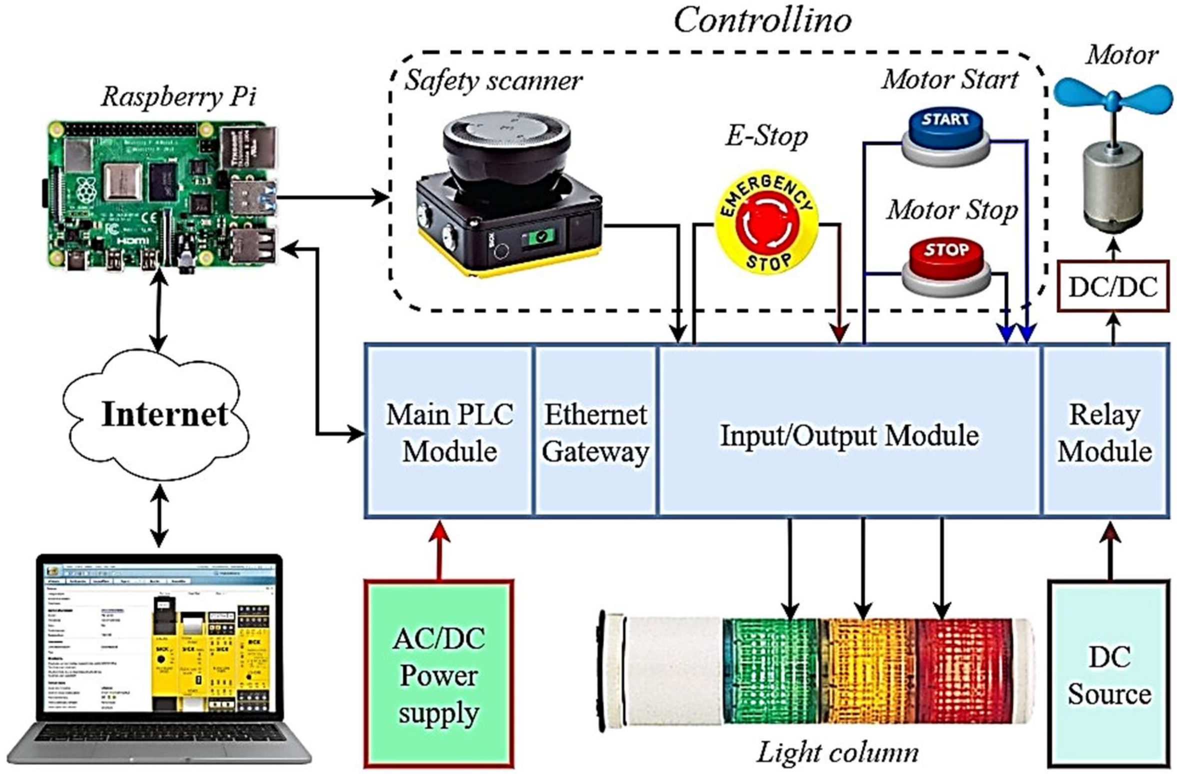

Fig. 1. Structure of the PLC-Based System Under Study

Source: developed by the authors.

The industrial compatible PLC *Controllino* [17] is applied for substitution of typical infrastructure components of industrial safety systems, such as *Emergency stop* button, *Motor Start* and *Motor Stop* buttons, and *Safety scanner*. Also, the real LED-based *Light column* is connected to the outputs of the *Safety PLC*. The opening and closing of a pair of built-in *Controllino* relays emulates for the *Safety PLC* the effect of pushing and releasing a typical 2-chain *E-Stop button* like *ES21* [18], which, according to the industrial safety standards, opens and closes the electrical loop between built-in special signal sources to *Safety PLC* and safe inputs of the same PLC. Another couple of built-in relays produce an effect of pressing of *Motor Start* and *Motor Stop* buttons, which correspond to a high logical level applied to non-safe inputs of *Safety PLC*. The pair of digital outputs of *Controllino* emulates the OSSD1 and OSSD2 signals like on the output of *nanoScan3* safety laser scanner [19]. The logic outputs of the input/output module are connected to three signal lamps – red, yellow, and green located on a *Light column*, as well as to the *Relay module*, which connects the *Motor* via a DC/DC converter *DTJ2024D05* [20] to power. Closing the relay upon command from the main module supplies power to the converter input, thereby starting the motor. A single-board computer, *Raspberry Pi 4* with 8 *GB* of RAM [21], is a key element of experimental setup, connecting main units to a unified system. The entire setup is powered by an AC/DC Power supply unit *WDR-120-24* [22].

Fig. 2 shows the physical appearance of the connected hardware components during the execution of the test program.

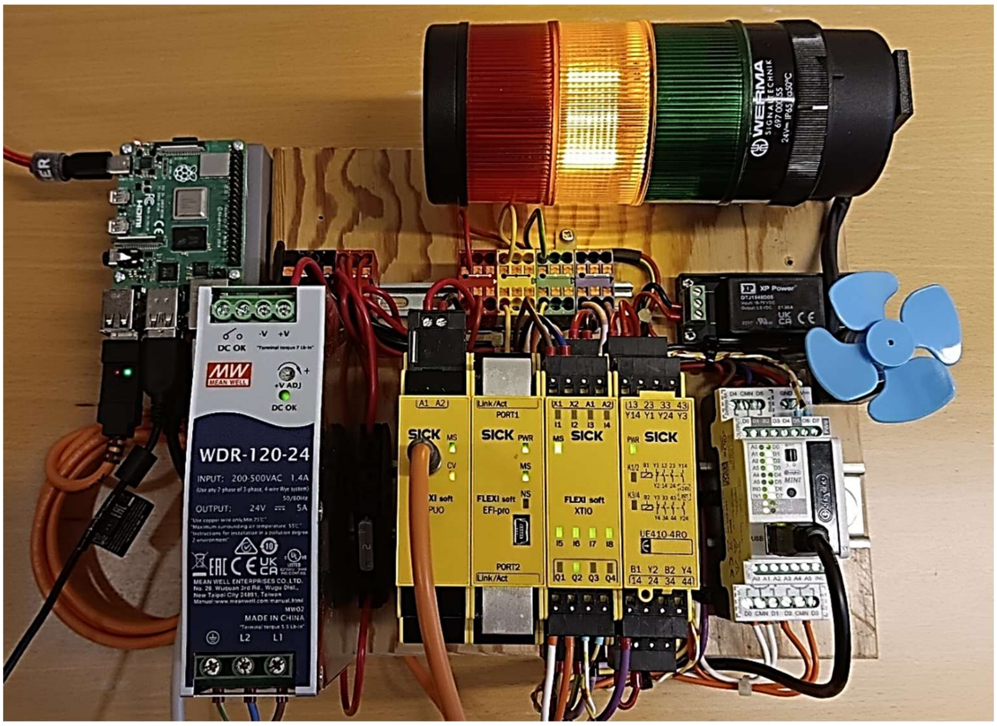

Fig. 2. Physical view of the connected hardware components

Source: developed by the authors.

Architecture and software components of the studied PLC-based system. The deployed software architecture generally resembles a typical three-tier web architecture, as used, for example, in [23], and illustrated in Fig. 3.

The most critical component of the system is *Raspberry Pi*, which runs on the Debian-based *Raspberry Pi OS* (32-bit). The three virtual servers provide Internet-based access and interaction between remote users and experimental hardware.

- The web server *Apache2* is listening for remote TCP connections on port 80. It provides an interactive web interface, which contains software buttons and an embedded video stream. Each pressing of a software button is being processed by *PHP/AJAX* handler to a specific code and forwarded via USB interface to *Controllino*.
- The streaming server *MJPG-streamer* [24] listens for TCP connections on port 8080 and allows a web server to capture and broadcast the video stream from a web camera.

- The *USB/IP Linux* package runs the process which accepts TCP connections on port 3240 via the Internet from installed on user computer *USB/IP client* [25].

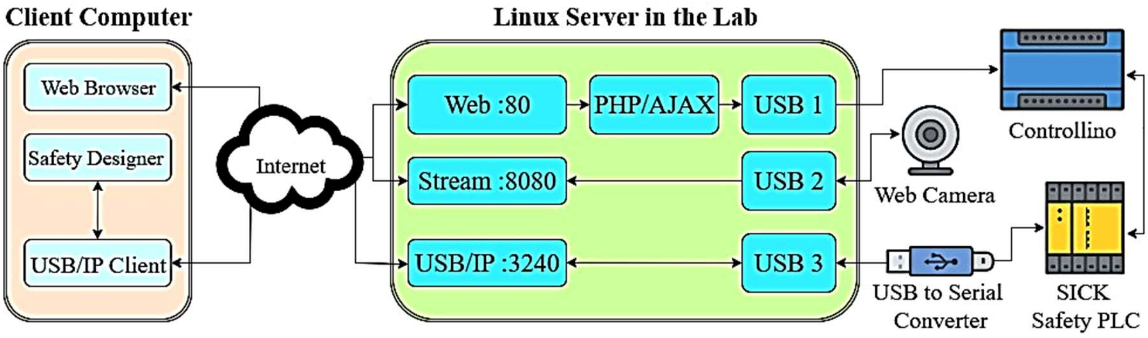

Fig. 3. Architecture of remote interaction with electronic equipment
Source: developed by the authors

On the client PC side (*Windows 10 Pro*, *version 22H2, build 19045.5965; Intel(R) Core(TM) i7-4770* CPU @ 3.40 GHz, 16.0 GB RAM), in addition to the *Sick Safety Designer* PLC development environment, the *USB/IP Client for Windows* [25] is also installed. This setup enables *Sick Safety Designer* on the client PC to access the *Safety PLC* remotely via the standard driver as if it has a virtual USB cable connected directly to the PLC.

A screenshot of the *Sick Safety Designer* PSS window in PLC configuration view mode is presented in Fig. 4.

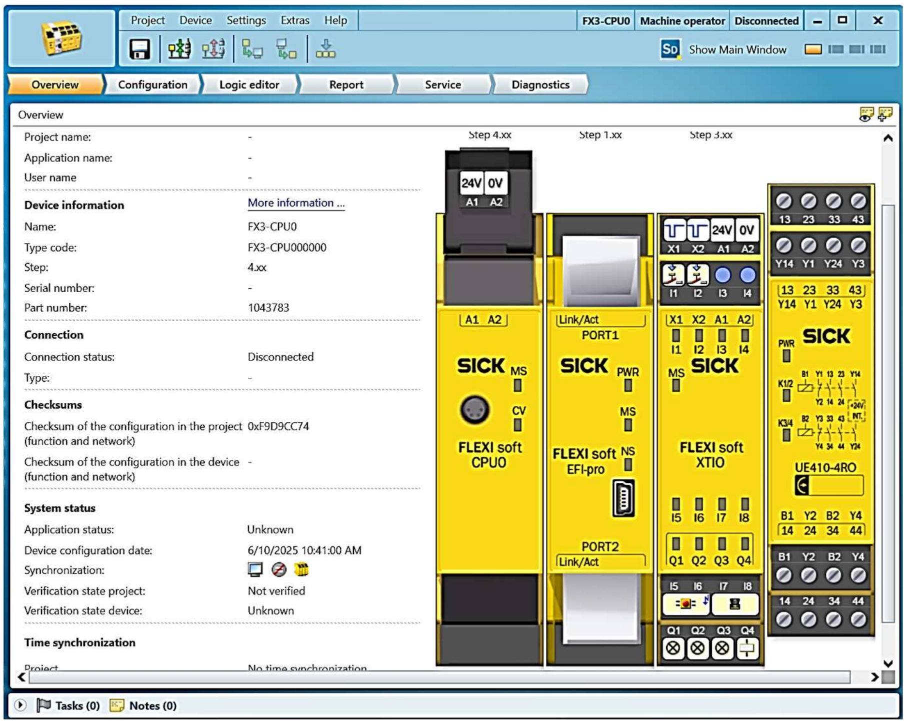

Fig. 4. Screenshot of the Sick Safety Designer window in PLC configuration view mode
Source: developed by the authors.

Functional opportunities. The hardware and software components of the system described above allow their main functions to be implemented.

- Program and debug the *Safety PLC* from a remote computer as if his computer were connected to the PLC using a USB cable.

- Change remotely via web interface the state of different inputs of *Safety PLC* as if the real corresponding hardware were connected to *Safety PLC*.
- Get visual feedback, observe indicators on the *Safety PLC*, *Light column,* and propeller of electric motor through the video stream from web camera.

The web interface, presented in Fig. 5, allows control of the virtual input devices by emulating via the *Controllino* corresponding input signals for the *I/O module*. This setup enables remote testing of the PLC program's response to the pressing and releasing of the *E-Stop* button, introducing a fault (breaking one of the wires of the safety chain) and fixing it. It also allows for initiating or canceling the confirmation from *Safety scanners*. In case the user program, which is uploaded remotely into the *Safety PLC,* is correct and the safety chain is correctly "assembled", the *Motor* can be started and stopped by pressing of corresponding buttons.

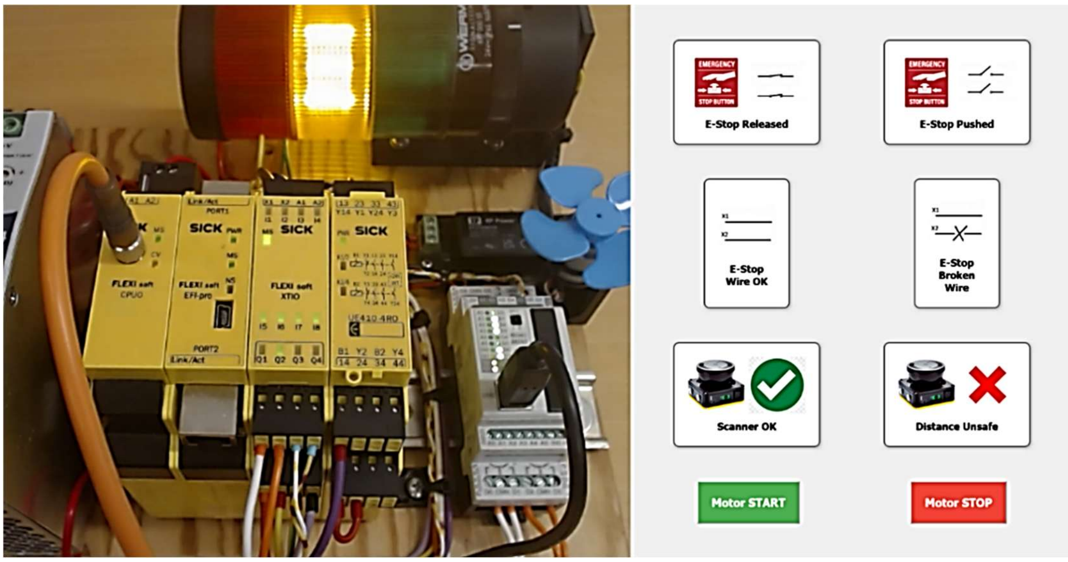

Fig. 5. Screenshot of the client browser window in PLC monitoring mode
Source: developed by the authors.

Fig. 6 illustrates the procedure of visual PLC programming using the *Sick Safety Designer* software.

While monitoring the system in real time, it is also possible to work with *Sick Safety Designer* to create a new project or modify the current one and upload it to the internal memory of the PLC.

Conclusions. As a result of the conducted research, the newest hardware-software system architecture has been developed and experimentally validated for the first time. This architecture enables the use of the original PLC development environment, *Sick Safety Designer*, both in standard local mode and remotely via the Internet. It allows the users to upload the created program to the industrial automation system located in the remote research or educational laboratory, as well as to monitor the operation of this system in real time.

The developed system is configured at the hardware level; however, due to the use of an intermediate industrial controller, it is possible to emulate specific safety scenarios within the industrial automation system under development and to perform hardware modifications remotely. This expands the potential applications of the proposed architecture in the educational process of training engineers in the fields of electronics and automation.

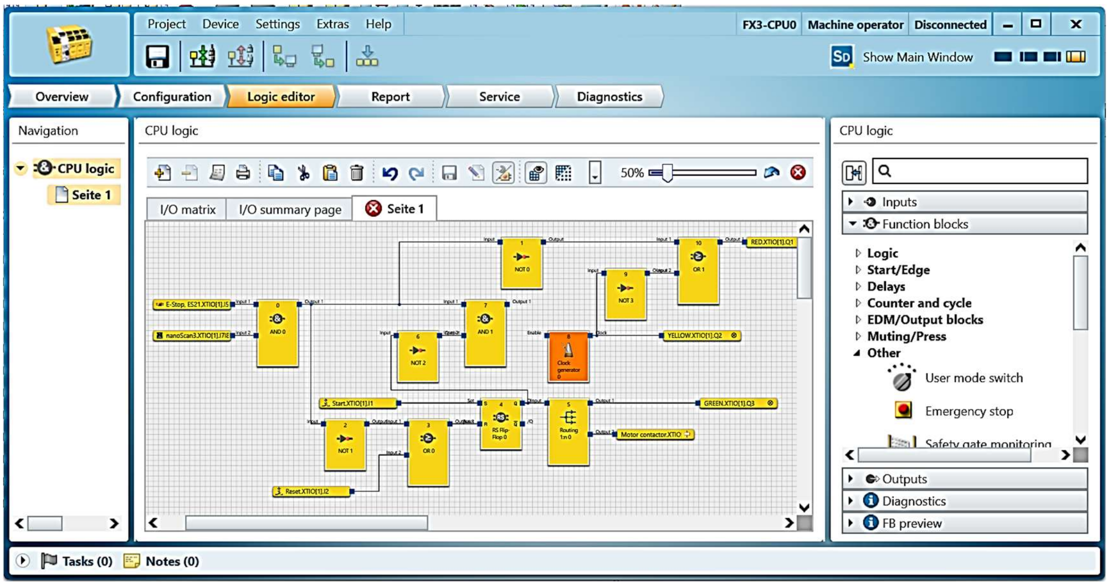

Fig. 6. Screenshot of the Sick Safety Designer window in Logic Editor mode
Source: developed by the authors.

The obtained practical results confirm the validity of the chosen strategy and demonstrate the feasibility of organizing remote work for industrial automation system (IAS) developers, as well as enabling students to conduct laboratory research in the distant - learning format.

To integrate the developed system into the educational process, it is necessary to extend its functionality by incorporating access control mechanisms, implementing a queuing system [26], and enabling interaction with a learning management system (e.g., *MOODLE* or *CANVAS*).

It is also advisable to experimentally verify the effectiveness of the proposed approach when working with other PLCs, as well as to implement remote interaction with laboratory equipment including various development boards (e.g., *STM32 Nucleo Board* and others). This would enable the execution of remote laboratory assignments for other engineering courses, including *Microcontrollers*, *Microprocessor Technology*, *Analog Electronics Devices*, *Electronic Monitoring and Visualization Tools*, and others [27].

Declaration of generative AI and AI-assisted technologies in the writing process. During the production of this work, the authors utilized Grammarly, Copilot, and DeepL services to enhance readability and language quality. After using these services, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Acknowledgements. The authors are highly grateful to the Crafoord Foundation for providing a grant.

References

1. Petruzella, F. D. (2023). *Programmable Logic Controllers*. (6th ed.). New York: McGraw-Hill.
2. Bolton, W. (2015). *Programmable Logic Controllers*. (6th ed.). Oxford: Newnes.
3. Electronicsinfos. (2024, July 24). *Top 11 PLC Brands of 2024*. https://www.electronicsinfos.com/2024/07/top-11-plc-brand-in-2024.html.
4. INSTBLOG. (2020, December 27). *Top 20 PLC Manufacturers: A List of PLC Brands and Their Rankings*. https://instrumentationblog.com/plc-manufacturers-plc-brands/.
5. Future Market Insights, Inc. (2024, Dec 4). *Programmable logic controller market*. https://www.futuremarketinsights.com/reports/programmable-logic-controller-market.
6. IEC Webstore. (n.d.). *IEC 61131-3:2025. Programmable controllers – Part 3: Programming languages.* https://webstore.iec.ch/en/publication/68533.

7. Eslami, A., Williams, A., Krauss, K., Rezaei, A. (2009, June). A remote access robotics and PLC laboratory for a distance learning program. Paper presented at the 2009 Annual Conference & Exposition, Austin, Texas. https://peer.asee.org/a-remote-access-robotics-and-plc-laboratory-for-distance-learning-program.

8. Shyr, W-J, Su, T-J, Lin, C-M. (2017). Development of Remote Monitoring and a Control System Based on PLC and WebAccess for Learning Mechatronics. *International Journal of Advanced Robotic Systems*. 2017, 10(2). https://doi.org/10.5772/55363.

9. Bakshi, S., Khairmode, G., Varkhede, N., Ayane, S. (2019, Mar). Monitoring and control of PLC based automation system parameters using IoT. *International Research Journal of Engineering and Technology* (IRJET). 2019, 6(3). https://www.irjet.net/archives/V6/i3/IRJET-V6I3124.pdf.

10. Aravind, C., Suji Prasad, S. J., Ponni Bala, M. (2020, Jan). Remote Monitoring and Control of Automation System with Internet of Things. *International Journal of Scientific & Technology Research*. 2020, 9(1). https://www.ijstr.org/final-print/jan2020/Remote-Monitoring-And-Control-Of-Automation-System-With-Internet-Of-Things.pdf.

11. SICK. (n.d.). *Safety controllers Flexi Soft*. https://www.sick.com/gb/en/catalog/products/safety/safety-controllers/flexi-soft/c/g186176.

12. SICK. (n.d.). *Safety controllers: Safety Designer*. https://www.sick.com/gb/en/catalog/products/safety/safety-controllers/safety-designer/safety-designer/ p/p674218?tab=detail.

13. SICK. (n.d.). *Safety controllers Flexi Soft. FX3-CPU000000*. https://www.sick.com/gb/en/catalog/products/safety/safety-controllers/flexi-soft/fx3-cpu000000/p/p80477?tab=detail.

14. SICK. (n.d.). *Safety controllers Flexi Soft. FX3-GEPR00000*. https://www.sick.com/gb/en/catalog/products/safety/safety-controllers/flexi-soft/fx3-gepr00000/p/p592553?tab=detail.

15. SICK. (n.d.). *Safety controllers Flexi Soft. FX3-XTIO84002*. https://www.sick.com/gb/en/catalog/products/safety/safety-controllers/flexi-soft/fx3-xtio84002/p/p80480.

16. SICK. (n.d.). *Safety controllers Flexi Soft. UE410-4RO4*. https://www.sick.com/gb/en/catalog/products/safety/safety-controllers/flexi-classic/ue410-4ro4/p/p54589.

17. CONTROLLINO. (n.d.). *CONTROLLINO MINI*. https://www.controllino.com/product/controllino-mini/, last accessed 2025/04/10.

18. SICK. (n.d.). *Safety switches. ES21*. https://www.sick.com/gb/en/catalog/products/safety/safety-switches/es21/c/g195527?tab=overview.

19. SICK. (n.d.). *Safety laser scanners. nanoScan3*. https://www.sick.com/gb/en/catalog/products/safety/safety-laser-scanners/nanoscan3/c/g507056?tab=overview.

20. XP Power. (n.d.). *Quality Power Supplies and Converters. DTJ20 series power product range*. https://www.xppower.com/product/DTJ20-Series?m=DTJ2024D05.

21. Raspberry Pi. (n.d.). *Raspberry Pi 4 model B*. https://www.raspberrypi.com/products/raspberry-pi-4-model-b/.

22. Mean Well. (n.d.). *Power Supply. WDR Series*. *WDR-120 Series*. https://www.meanwell.com/productSeries.aspx?i=52&c=7#tag-7-52.

23. Li, Y., Esche, S.K., & Chassapis, C. (2008). A scheduling system for shared online laboratory resources. *2008 38th Annual Frontiers in Education Conference, Saratoga Springs, NY, USA, 2008, pp. T2B-1-T2B-6, doi: 10.1109/FIE.2008.4720253.*

24. Grinberg, M. (2013, Oct 1). *How to build and run MJPG-streamer on the raspberry Pi*. miguelgrinberg.com. https://blog.miguelgrinberg.com/post/how-to-build-and-run-mjpg-streamer-on-the-raspberry-pi.

25. Vadimgrn/usbip-win2. (2025, May 30). *USB/IP client for Windows*. GitHub. https://github.com/vadimgrn/usbip-win2.

26. Liguori, C., Pietrosanto, A., Carratù, M., Paciello, V., De Santos, M., Di Leo, G., Ferro, M. (2024). Remote laboratories for measurement courses during the Covid-19 era. *AIP Conf. Proc.* 2816, 160003. https://doi.org/10.1063/5.0179433.

27. Vera, R.J., Galilea, J.L.L.G., Vicente, A.G., De-La-Llana-Calvo, A. & Munoz, I.B. (2024). A Flexible Framework for the Deployment of STEM Real Remote Laboratories in Digital Electronics and Control Systems. *IEEE Access*, *12*, 14563–14579. https://doi.org/10.1109/ACCESS.2024.3357991.

Отримано 23.07.2025

УДК 681.5:004.77

Володимир Павлович Войтенко[1], Олександр Ілліч Писаревський[2]

[1]кандидат технічних наук, доцент, доцент кафедри електроніки, робототехніки, автоматики та мехатроніки

Національний університет «Чернігівська політехніка» (Чернігів, Україна);
дослідник кафедри автоматичного керування
Лундський університет (Лунд, Швеція)
E-mail: v.voytenko@stu.cn.ua. **ORCID:** http://orcid.org/0000-0003-1490-0600
ResearcherID: F-8698-2014. **Scopus Author ID:** 36167678700

[2]Інженер-дослідник
Лундський університет (Лунд, Швеція)
E-mail: alexander.pisarevskiy@control.lth.se. **ORCID:** https://orcid.org/0009-0004-0019-1502

ДИСТАНЦІЙНА РОЗРОБКА КОНТРОЛЕРІВ БЕЗПЕКИ ДЛЯ ПРОМИСЛОВОЇ СИСТЕМИ АВТОМАТИЗАЦІЇ

Програмовані логічні контролери (PLC) є основою сучасних промислових систем автоматизації, а також широко застосовуються у транспорті, енергетиці, робототехніці, системах безпеки, водопостачання та водовідведення, в будівлях і спорудах, у медичних приладах. В основі побудови PLC лежить мікропроцесор, а тому розробка та супроводження системи на базі PLC потребує висококваліфікованих спеціалістів, які розуміють можливості PLC, особливості їхньої взаємодії з польовими пристроями та здатні працювати із специфічним інструментальним програмним забезпеченням PLC. Звідси випливає актуальність проблеми підвищення ефективності роботи таких спеціалістів, а також їхньої якісної підготовки, що потребує багато часу та вартісного обладнання.

Розвиток інформаційних і комп'ютерних технологій відкриває таку можливість у вирішенні зазначеної проблеми, як застосування дистанційних методів розробки, а також навчання роботі з віддаленими системами автоматизації на базі PLC. Цей підхід стає особливо актуальним в Україні, оскільки дозволяє не тільки економити на відрядженнях спеціалістів чи кількості потрібного лабораторного устаткування, але й підвищує безпеку завдяки розосередженню людей. Синергетичний ефект може бути досягнутий і за рахунок планування часу роботи як спеціалістів, так і обладнання, а також можливості їхньої узгодженої спільної взаємодії.

Особливу роль у промисловій автоматизації відіграють системи безпеки, оскільки виробничий травматизм є неприпустимим ані з суспільного, ані з економічного погляду. Ця стаття зорієнтована на дистанційну розробку та навчання роботи саме з контролерами безпеки, які, по суті, являють собою PLC, проте мають специфічні польові пристрої, використовують надійні протоколи зв'язку з PLC та потребують відповідної сертифікації перед тим, як бути запровадженими у виробництво. Мета виконаного дослідження – створення архітектури програмно-апаратної системи, яка дає можливість працювати з контролерами безпеки дистанційно, використовуючи стандартне інструментальне забезпечення PLC, а також і додаткові компоненти. Ці компоненти дозволяють в реальному часі спостерігати за віддаленою системою автоматизації та модифікувати і програму, і апаратну конфігурацію.

Для вирішення поставлених задач була розроблена структура системи безпеки на базі PLC, розгорнуте відповідне програмне забезпечення, а також проведені експериментальні дослідження, які підтверджують можливість дистанційної взаємодії з потрібними апаратними засобами, у тому числі під час виконання лабораторних завдань студентами інженерних спеціальностей.

Стаття може бути корисною для розробників систем автоматизації на базі PLC, а також використана в освітньому процесі.

Ключові слова: *промислові системи автоматизації; програмовані логічні контролери (PLC); контролери безпеки; програмне забезпечення PLC; дистанційна розробка програми для PLC; дистанційне інженерне навчання; мікропроцесор.*

Рис.: 6. Бібл.: 27.

Voytenko, V., Pisarevskiy, A. (2025). Remote design of safety controllers for industrial automation systems. *Technical sciences and technologies,* (3(41)), 401-409. DOI: https://doi.org/10.25140/2411-5363-2025-3(41)-401-409.