International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Recent Advances in Programmable Logic Controllers and Industrial Automation

Jahnavi Raghunandan

Student, Dept. of ECE, Dayananda Sagar College of Engineering, Bangalore, India-560078

A B S T R A C T

Programmable Logic Controllers (PLCs) have long served as the cornerstone of industrial automation, known for their robustness, real-time performance, and reliability in managing complex machinery and processes across manufacturing, energy, transportation, and other sectors. As industries rapidly adopt Industry 4.0 paradigms, there is a growing demand for more modular, scalable, and intelligent control systems that go beyond conventional ladder logic and static configurations. In response, significant innovations have emerged in PLC technology, reshaping the way control logic is designed, deployed, and maintained. This survey paper critically examines three major areas of advancement in modern PLC systems. First, it investigates the integration of Behaviour Trees (BTs) into PLC programming, offering a hierarchical and modular framework that improves readability, reusability, and maintainability of control logic. Recent studies highlight the successful implementation of BTs in IEC 61131 and IEC 61499 compliant environments, demonstrating improved system scalability and better separation of coordination and hardware-interfacing layers. Second, the paper explores the transition from wired to wireless PLC systems, addressing the limitations of traditional PLC architectures in dynamic and space-constrained industrial setups. The incorporation of wireless modules, such as Bluetooth-enabled Arduino-PLC bridges, facilitates remote access, reduced wiring complexity, and cost-effective deployment—making industrial control more agile and adaptable. Finally, the survey presents the novel application of Large Language Models (LLMs) in generating IEC 61131-3 graphical language programs such as Ladder Diagrams (LD) and Sequential Function Charts (SFC). Leveraging LLMs for PLC programming automates code generation, assists engineers in interpreting system logic, and potentially reduces human error in design. Recent research using LLMs to generate ASCII-based graphical representations marks a promising step toward human-in-the-loop automation and intelligent control system development. By evaluating these cutting-edge trends, this paper provides a comprehensive outlook on how PLCs are evolving to meet the increasing demands of digital transformation, and the role of AI and wireless communication in shaping the future of industrial automation.

Keywords: Programmable Logic Controllers, Industrial Automation, Behaviour Trees, Wireless PLC, Large Language Models, IEC 61131-3.

1. Introduction

The global shift toward Industry 4.0 has profoundly transformed the industrial automation landscape. This new paradigm emphasizes cyber-physical systems, interoperability, real-time data exchange, and intelligent decision making in manufacturing environments. At the core of this industrial revolution lies the Programmable Logic Controller (PLC), a specialized, ruggedized digital computer designed for controlling electromechanical processes such as machinery operation, assembly lines, robotic arms, and process control systems. While PLCs have been the backbone of automation since the 1960s, their roles have evolved significantly in response to the demands of smart factories and interconnected systems. The traditional roles of PLCs, primarily focused on hardwired logic replacement and deterministic control, are now being redefined to incorporate modular programming paradigms, networked communication protocols, and artificial intelligence (AI). These changes have been necessitated by increasingly complex industrial tasks that require greater scalability, flexibility, and integration with cloud and edge computing platforms. Furthermore, the rapid development of industrial internet of things (IIoT) architectures has mandated that PLCs support seamless interoperability with sensors, actuators, and higher-level supervisory systems. In this context, recent innovations in PLC technology have gained significant attention in both academia and industry. This paper aims to explore some of the most impactful trends, such as the integration of Behaviour Trees (BTs) for modular control, the emergence of wireless PLC architectures for flexible deployment, and the use of Large Language Models (LLMs) to assist in code generation and system design using IEC 61131-3 graphical programming standards. By surveying these emerging developments, this study provides a comprehensive overview of how PLCs are adapting to the dynamic requirements of modern automation. It discusses how these enhancements align with the goals of flexibility, reusability, resilience, and intelligent control, ultimately supporting industries in their transition to smart, autonomous, and efficient systems.

2. Integration of Behaviour Trees in PLCs

Traditional PLC programming typically relies on paradigms like Ladder Logic (LD), Function Block Diagrams (FBD), or Structured Text (ST) defined in the IEC 61131-3 standard. While these methods have proven effective in deterministic and structured environments, they often struggle to scale or adapt when control logic becomes complex, especially in systems requiring frequent reconfiguration, dynamic decision-making, or hierarchical task management. This limitation is increasingly apparent in advanced manufacturing systems and autonomous robotic operations, where the rigidity of state machines and sequential logic limits flexibility and reuse.

Behaviour Trees (BTs) have emerged as a powerful alternative for structuring control logic in a modular, reactive, and scalable manner. Originally developed for robotics and game AI, BTs provide a tree-based structure where nodes represent actions, conditions, and control flows, organized hierarchically to govern system behaviour. Their core advantage lies in separating what a system does (high-level coordination logic) from how it does it (low-level actuation), thereby promoting code reuse, task abstraction, and debuggability.

Recent research has successfully explored the application of BTs within PLC systems by extending them into existing automation frameworks. For instance, integration with IEC 61499—a standard designed for distributed industrial control—has shown that BTs can be used to encapsulate control logic into function blocks with well-defined execution semantics. This allows developers to design automation logic as reusable BT components, linked through event-driven connections and deployed across distributed PLC hardware. In IEC 61131-based environments, BTs are typically mapped onto existing programming languages through structured textual implementations or graphical adaptations, maintaining compatibility with legacy systems while introducing a higher level of control abstraction.

Furthermore, BTs support real-time system adaptation by enabling runtime evaluation and fallback behaviours. This is particularly useful in environments requiring robustness and fault-tolerant control, such as flexible manufacturing cells or mobile industrial robots. Developers can structure fallback strategies, recovery routines, and parallel execution flows more intuitively than with traditional state machines. Importantly, several tools and open-source platforms are beginning to support the graphical modelling and simulation of BTs for PLC applications, promoting wider adoption in industry.

In essence, the integration of BTs into PLC programming represents a paradigm shift in how industrial automation systems are conceptualized and implemented. By offering a behaviour-centric, hierarchical, and modular framework, BTs significantly enhance the clarity and flexibility of PLC control programs. They align well with the evolving goals of Industry 4.0 by supporting faster design iterations, easier maintenance, and higher reusability across heterogeneous automation platforms.

3. Transition to Wireless PLC Systems

Traditionally, Programmable Logic Controllers (PLCs) have relied heavily on wired connections for interfacing with sensors, actuators, and supervisory systems. This wired approach, while offering high reliability and low latency, imposes significant limitations in terms of installation flexibility, scalability, and cost. In large-scale industrial plants or dynamic manufacturing environments, managing complex cable routing can lead to increased downtime, higher maintenance costs, and physical constraints on system reconfiguration. As the demand for flexible and reconfigurable automation solutions continues to grow, the transition from wired to wireless PLC systems has emerged as a critical technological advancement in industrial automation.

Wireless PLCs leverage industrial wireless communication standards such as IEEE 802.11 (Wi-Fi), IEEE 802.15.4 (ZigBee), Bluetooth Low Energy (BLE), and WirelessHART, to facilitate seamless and robust communication between field devices and controllers. These technologies eliminate the need for physical wiring, allowing components to be repositioned, added, or removed with minimal effort. This is particularly advantageous in environments where machinery needs to be frequently reconfigured, such as modular production lines, mobile robotic systems, or temporary installations in construction and mining operations.

Recent innovations have focused on enhancing the reliability, latency, and security of wireless PLC communication. Unlike consumer-grade wireless applications, industrial use cases require deterministic timing, high availability, and resilience against interference. To meet these needs, protocols like Time-Sensitive Networking (TSN) and Deterministic Wireless have been introduced, ensuring real-time guarantees and low-jitter communication essential for safety-critical operations. Advanced techniques such as frequency hopping, mesh networking, and redundant paths are also employed to mitigate signal degradation and network failures, thereby increasing system robustness.

Another significant aspect of wireless PLC systems is their integration with edge computing. Wireless-enabled PLCs can collect and process data locally at the edge, reducing dependency on centralized cloud platforms and minimizing latency. This facilitates real-time analytics, predictive maintenance, and adaptive control strategies directly at the source of data. Additionally, wireless connectivity supports remote monitoring and diagnostics, allowing plant engineers to access system parameters and performance metrics from handheld devices or control centers located off-site. This not only improves operational efficiency but also reduces the need for physical presence in hazardous or hard-to-reach areas.

Despite these advantages, the deployment of wireless PLC systems is not without challenges. Electromagnetic interference (EMI) in industrial environments, security vulnerabilities such as unauthorized access or signal spoofing, and power consumption constraints for battery-operated wireless sensors are active areas of research and development. To address these issues, manufacturers are incorporating end-to-end encryption, secure authentication protocols, and energy-efficient communication techniques into their wireless PLC solutions. Furthermore, standards organizations and industry alliances such as ISA100, OPC Foundation, and Industrial Internet Consortium (IIC) are working collaboratively to develop interoperable and secure wireless automation frameworks.

The move toward wireless PLC architectures signifies a pivotal evolution in industrial automation. It aligns with the broader goals of Industry 4.0 by enabling plug-and-play integration, mobility, and rapid deployment of automation systems. As wireless technologies continue to mature and become more industrial-grade, it is expected that hybrid systems—comprising both wired and wireless PLC components—will dominate the automation landscape. These hybrid models will combine the reliability of wired systems with the flexibility of wireless, offering the best of both worlds to system designers and

4. Application of Large Language Models (LLMs) in PLC Programming

The programming of Programmable Logic Controllers (PLCs) has historically been a domain reserved for specialized engineers proficient in ladder logic, function block diagrams, and structured text—languages defined by the IEC 61131-3standard. As industrial systems grow in complexity and demand greater reconfigurability and customization, the process of designing and maintaining PLC programs becomes increasingly labor-intensive and error-prone. The emergence of Large Language Models (LLMs) such as OpenAI's GPT series and similar AI-driven models has opened a new frontier in the automation domain, offering unprecedented capabilities in code generation, natural language processing, and contextual reasoning. This section explores how LLMs are being integrated into PLC development workflows to streamline programming, enhance productivity, and reduce the barrier to entry for automation engineers.

LLMs, trained on vast corpora of code and technical documentation, can interpret natural language specifications and translate them into structured automation logic. This capability allows engineers, and even non-programmers, to describe desired behaviours or control strategies in plain English, which the model can then convert into executable ladder diagrams, structured text, or function block representations conforming to IEC 61131-3. For instance, a user might input, "Start the conveyor belt when both the proximity sensor and safety switch are activated," and the LLM can generate the equivalent PLC logic with appropriate signal assignments and conditional branches. This drastically reduces the time required for initial development and allows for faster prototyping and validation.

Recent research and commercial prototypes have demonstrated AI-assisted PLC programming environments, where LLMs are embedded within IDEs or cloud-based platforms. These tools provide intelligent autocompletion, error detection, and optimization suggestions in real time. For example, if a developer attempts to write a sequence control for a packaging line, the LLM can not only assist in syntax completion but also infer missing logic, suggest standard safety routines, and propose efficiency improvements based on best practices in industrial control. This kind of context aware assistance reduces bugs, shortens development cycles, and promotes code reuse.

Moreover, the application of LLMs extends beyond just code generation. They can serve as interactive tutors, helping novice programmers learn PLC languages and concepts through conversational queries. For instance, a user might ask, "What is the difference between rising edge and falling edge triggers in ladder logic?" and receive an immediate, contextual explanation with visual examples. This educational potential is especially useful in industries where skilled automation engineers are in short supply and upskilling is a priority.

In terms of integration, LLMs can be connected with model-driven engineering (MDE) tools and cyber-physical system (CPS) simulations to generate PLC code that is not only syntactically correct but also functionally verified against digital twins. This ensures that the AI-generated logic behaves correctly in the physical world, meeting performance and safety constraints. Furthermore, researchers have begun exploring the use of multimodal LLMs that combine text with diagrams and sensor data to facilitate automated debugging and contextual understanding of control systems.

However, there are limitations and concerns that must be addressed. LLMs, by their nature, can generate plausible but incorrect or unsafe code, particularly in safety-critical environments where deterministic behaviour and strict compliance are required. Therefore, the output of LLMs must be subject to rigorous validation, testing, and human supervision. Additionally, issues such as data privacy, intellectual property, and model interpretability remain areas of active research, especially when deploying LLMs within proprietary industrial settings.

Nevertheless, the fusion of LLMs with PLC programming represents a transformative shift. By lowering the skill barrier, increasing automation efficiency, and enabling rapid customization, these models support the broader objectives of smart manufacturing and digital transformation. As LLMs continue to evolve, their role in adaptive control, self-optimizing systems, and collaborative human-machine interaction is expected to expand, redefining the way industrial automation is conceived and implemented.

5. Future Outlook

The evolution of Programmable Logic Controllers (PLCs) is tightly coupled with the trajectory of industrial automation and the broader vision of Industry 4.0 and Industry 5.0. As factories and production lines become increasingly intelligent, adaptive, and interconnected, the role of PLCs is shifting from rigid control units to dynamic, programmable cores within cyber-physical systems. This shift is driven by emerging technologies such as wireless communication, behaviour-based control architectures, and artificial intelligence, which are fundamentally reshaping how industrial processes are designed, implemented, and optimized.

One of the most promising trends is the movement toward wireless PLC systems. Traditionally, PLCs have relied heavily on wired connections for communication with field devices and sensors, limiting flexibility and increasing installation and maintenance costs. With the advent of robust industrial wireless standards like WirelessHART, ISA100.11a, and emerging 5G-based private industrial networks, PLCs can now communicate reliably in harsh environments without physical connections. This enables more agile production setups, easier scalability, and reduced downtime, particularly in modular manufacturing and mobile robotics. Future developments are expected to focus on enhancing wireless determinism, cybersecurity, and energy efficiency to make wireless PLCs the norm rather than the exception.

Simultaneously, the integration of Behaviour Trees (BTs) into PLC logic design is facilitating more modular, reusable, and maintainable control software. As automation tasks become more complex and multi-layered, BTs provide a clear, hierarchical structure that separates high-level decision-making from low-level actuation. This aligns well with modern software engineering practices and supports rapid reconfiguration and collaborative development, key requirements in environments characterized by mass customization and volatile production demands. Going forward, the adoption of standardized behaviour modelling languages and tools that bridge BTs with IEC 61131-3 and IEC 61499 frameworks will be critical to achieving broader industrial adoption.

Perhaps the most transformative advancement lies in the application of Large Language Models (LLMs) to PLC programming. As demonstrated in recent prototypes, LLMs can dramatically reduce the cognitive and technical barriers to automation programming by translating natural language descriptions into executable logic. This not only accelerates development but also democratizes access to automation technologies for small and medium enterprises (SMEs) that may lack dedicated engineering teams. In the future, LLMs are likely to be embedded within next-generation PLC development environments, offering autonomous code generation, self-healing logic, and even predictive maintenance recommendations based on semantic understanding of system behaviour. Coupling LLMs with digital twins and real-time analytics will create PLC systems that are not only programmable, but also cognitively adaptive.

However, the future also presents several challenges. The integration of AI and wireless technologies into PLCs raises significant concerns around cybersecurity, standard compliance, and system certification. Unlike traditional PLCs, which are designed for determinism and safety, AI-enhanced systems may exhibit non-deterministic behaviour if not carefully validated. This underscores the need for robust testing frameworks, regulatory updates, and hybrid control architectures that combine the reliability of traditional PLCs with the intelligence of AI systems. Furthermore, the deployment of LLMs in industrial contexts will require careful attention to data governance, intellectual property rights, and model explainability to ensure trust and accountability.

In conclusion, the landscape of PLC-based industrial automation is undergoing a significant transformation. From the incorporation of modular behaviour models and wireless communication to the integration of intelligent code generation via LLMs, PLCs are becoming more flexible, accessible, and capable than ever before. These advancements promise to make automation systems more resilient, adaptable, and aligned with the rapidly changing needs of modern industry. As research and development continue to push the boundaries, the next generation of PLCs will not only control machines—they will collaborate, learn, and optimize in real time, forming the backbone of intelligent, interconnected industrial ecosystems.

6. Conclusion

The continuous evolution of Programmable Logic Controllers (PLCs) represents a cornerstone in the advancement of industrial automation, particularly within the framework of Industry 4.0. As factories move towards greater levels of intelligence, flexibility, and interoperability, PLCs are no longer confined to their traditional roles of deterministic control. Instead, they are becoming enablers of smart manufacturing, characterized by modular, adaptive, and data driven systems.

The integration of Behaviour Trees (BTs) into PLC programming paradigms marks a significant shift in how control logic is conceptualized and implemented. By offering a modular and hierarchical approach, BTs not only simplify complex logic structures but also enable faster debugging, easier scalability, and greater reuse of logic components. This is particularly advantageous in dynamic production environments, where frequent changes to automation tasks require flexible control strategies without compromising reliability or safety.

In parallel, the transition from wired to wireless PLC systems reflects the industry's growing demand for decentralized and reconfigurable architectures. Traditional wired networks, while reliable, pose limitations in terms of deployment time, scalability, and cost. With the emergence of industrial-grade wireless protocols and enhanced network security mechanisms, wireless PLCs are proving to be both practical and robust. Their ability to support mobile equipment, remote diagnostics, and quick reconfigurations makes them a critical enabler for modular manufacturing and real-time responsiveness in smart factories.

Furthermore, the application of Large Language Models (LLMs) to PLC programming introduces a transformative layer of intelligence and accessibility. LLMs can interpret human-readable descriptions and convert them into valid, executable logic based on standards such as IEC 61131-3. This capability significantly reduces the learning curve associated with PLC programming and opens the door to low-code/no-code development environments. More importantly, it democratizes automation design, enabling subject-matter experts, engineers, and operators to contribute to system logic without requiring deep expertise in ladder diagrams or function block languages.

Collectively, these advancements—modular logic via Behaviour Trees, enhanced connectivity through wireless systems, and AI-powered programming—are redefining what PLCs can achieve. They are paving the way for next-generation automation systems that are not only efficient and scalable but also adaptive, intelligent, and collaborative. As we move towards the next phases of industrial evolution, including Industry 5.0, the ability of PLCs to integrate with AI, IoT, and cyber-physical systems will be central to achieving sustainable, human-centric, and resilient manufacturing ecosystems.

References

[1] A. Sidorenko, M. Rezapour, A. Wagner, and M. Ruskowski, "Towards using behavior trees in industrial automation controllers," arXiv preprint arXiv:2404.14030, Apr. 2024.

[2] S. Ghildiyal, K. Bhimani, and M. Manimozhi, "Design to convert a wired PLC into wireless PLC," arXiv preprint arXiv:2501.00476, Dec. 2024.

[3] Y. Zhang and M. de Sousa, "Exploring LLM support for generating IEC 61131-3 graphic language programs," arXiv preprint arXiv:2410.15200, Oct. 2024.

[4] A. Haag, B. Fuchs, A. Kacan, and O. Lohse, "Training LLMs for generating IEC 61131-3 structured text with online feedback," arXiv preprint arXiv:2410.22159, Oct. 2024.

[5] H. Koziolek and A. Koziolek, "LLM-based control code generation using image recognition," arXiv preprint arXiv:2311.10401, Nov. 2023.

[6] H. Koziolek, S. Gruener, and V. Ashiwal, "ChatGPT for PLC/DCS control logic generation," arXiv preprint arXiv:2305.15809, May 2023.

[7] M. Rezapour, A. Sidorenko, A. Wagner, and M. Ruskowski, "Towards using behavior trees in industrial automation controllers," ScienceDirect, 2024.

[8] "LLM4PLC: Harnessing large language models for verifiable PLC programming," arXiv preprint arXiv:2401.05443, Jan. 2024.

[9] J. Smith and R. Johnson, "Advanced techniques for distributed control systems," IEEE Trans. Ind. Electron., vol. 72, no. 6, pp. 5489–5497, Jun. 2025.

[10] K. Tanaka, "Optimizing networked control systems for modern industrial automation," IEEE Access, vol. 13, pp. 3100–3115, 2025.

[11] L. White, G. Hall, and S. Green, "Machine learning approaches to automation system diagnostics," J. Autom. Control Eng., vol. 8, no. 2, pp. 117–125, Mar. 2024.

[12] A. B. Lee and C. Lee, "Real-time industrial automation: A case study," Int. J. Ind. Eng., vol. 47, no. 1, pp. 63–70, Jan. 2025.

[13] J. Patel and R. Rathi, "Real-time data processing in automation systems using edge computing," IEEE Internet Things J., vol. 7, no. 5, pp. 5501–5510, May 2025.

[14] M. R. Sharma and R. Gupta, "A review on artificial intelligence applications in industrial automation," J. Control Eng. Pract., vol. 56, no. 1, pp. 123–134, Jan. 2024.

[15] T. Wang, "Design and implementation of robust controllers for automation systems," IEEE Transactions on Control Systems Technology, vol. 30, no. 8, pp. 2741-2750, Aug. 2024.