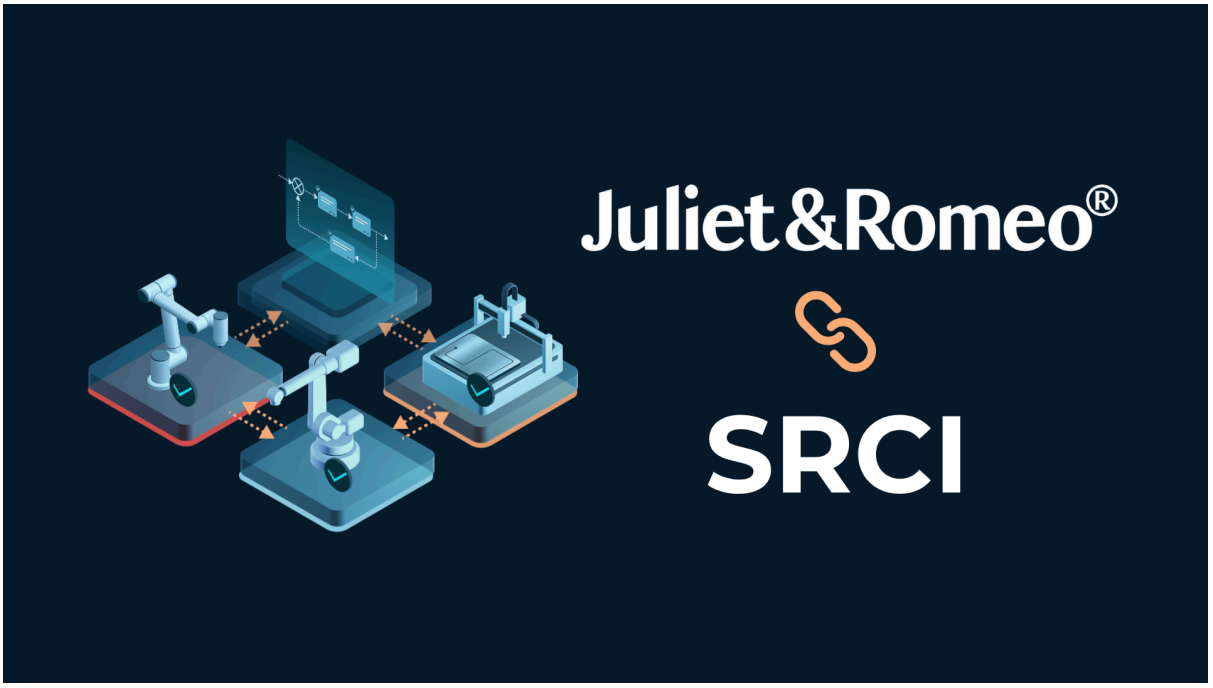
An explanation of the relation between Juliet & Romeo® and Standard Robot Command Interface, SRCI

This article tries to explain the relation between Juliet & Romeo® and the Standard Robot Command Interface, outlining their respective roles, design principles, and how they can be used together in modern robotics and automation projects. The important thing here is that the two technologies are not mutually exclusive but rather overlapping and can easily coexist and complement each other across the automation stack.



At a Glance: SRCI vs Juliet & Romeo

- **SRCI** defines a standardized interface between PLCs and robot controllers, using vendor-independent function blocks and communication structures. It's ideal for allowing robots to be commanded from PLC logic in a unified, hardware-agnostic way.

- **Juliet & Romeo** offers a modern, expressive programming language for deterministic multitasking, motion programming, logic, AI, and more.

Juliet & Romeo can implement SRCI servers to expose motion control to external PLCs, or act as SRCI clients to control robots that expose an SRCI-compatible interface.

Cognibotics AB | Ideon Science Park | Scheelevägen 15 | SE-22370 | Lund | SWEDEN | +46 (0) 46 286 59 90 | info@cognibotics.com

This two-way compatibility allows seamless integration into SRCI-based PLC environments or broader automation architectures, letting Juliet applications benefit from the SRCI interface just as PLC systems does. Where SRCI is limited to the robot interface, Juliet & Romeo offers a complete suite for building modern robot applications.

Brief introduction to Juliet & Romeo and SRCI

Juliet & Romeo

- Juliet is a robot and automation programming language (from Cognibotics), designed to support real-time motion, process logic, sensor handling, user interfaces, AI integration, etc. It is intended to enable developers to model, reuse and evolve complex automation systems with clarity.

- Romeo is the real-time virtual machine/runtime that executes Juliet (and potentially other languages that follow its bytecode format). It offers execution safety, deterministic behavior, real-time constraints, automatic memory management, etc.

- It's positioned as a "future-proof robotics and automation foundation" that unifies programming, execution, and system evolution; meant to reduce setup time, support for modular updates, enable AI readiness, etc.

SRCI

- SRCI is an open standard/protocol/interface that defines how PLCs (Programmable Logic Controllers) can send standardized robot commands to robot controllers (servers), regardless of manufacturer. It is being defined and hosted by groups like Profibus & Profinet International.

- It standardizes not just communication but also function-blocks / libraries in the PLC environment, so you can program robot motion, logic, etc., via PLC using SRCI blocks rather than proprietary robot interfaces.

- It has (or will have) *profiles* (Core, Extended, optional) that define a set of commands (move, path, force control, etc.), and vendors can support subsets depending on hardware or version.

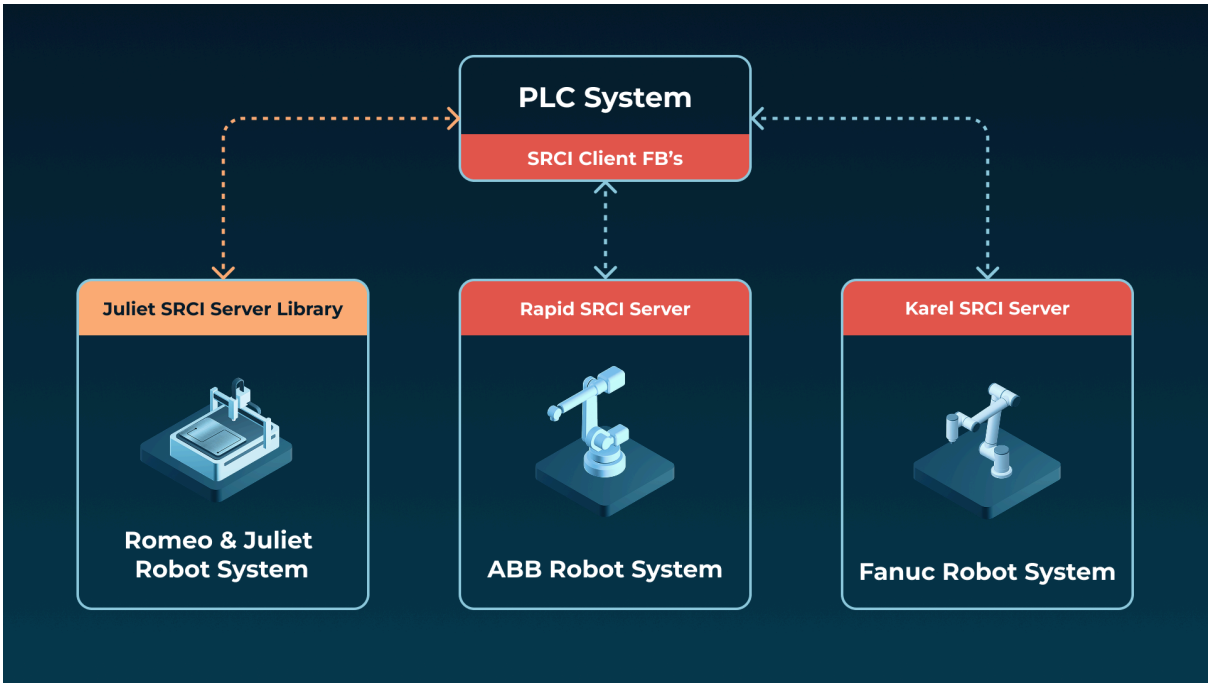
Compatibility between SRCI and Juliet & Romeo

The two technologies are complementary and overlapping, not competing. This compatibility offers two primary integration paths:

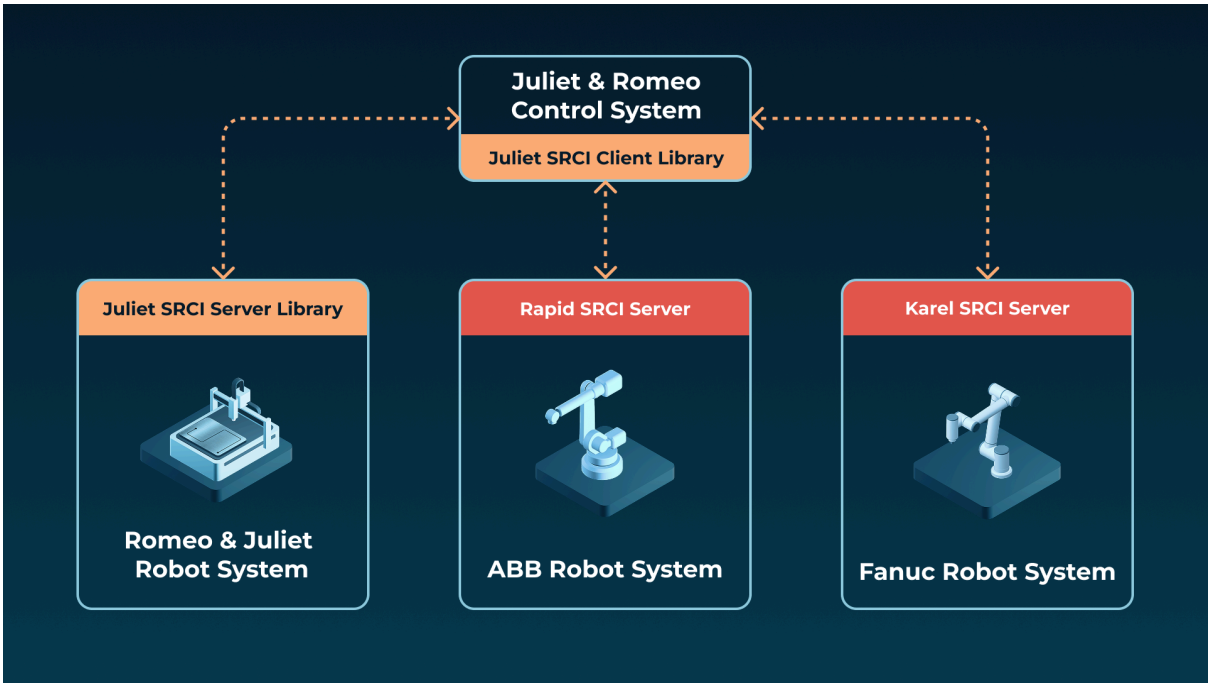
Juliet & Romeo as an SRCI Server: When used to develop a robot control system (as done by

Cognibotics AB | Ideon Science Park | Scheelevägen 15 | SE-22370 | Lund | SWEDEN | +46 (0) 46 286 59 90 | info@cognibotics.com

Estun Automation), Juliet & Romeo can incorporate an SRCI library. This allows external SRCI clients, such as PLC systems, to seamlessly access and command the Juliet & Romeo-based robot control system.



Juliet & Romeo as an SRCI Client: For line or cell control systems, a Juliet & Romeo application can utilize an SRCI client library to send standardized commands to a robot system that exposes an SRCI-compatible interface.



Cognibotics AB | Ideon Science Park | Scheelevägen 15 | SE-22370 | Lund | SWEDEN | +46 (0) 46 286 59 90 | info@cognibotics.com

Detailed Comparison: Goals and Strengths

Aspect	Juliet & Romeo	SRCI
Abstraction / Expressiveness	High: Juliet is a full programming language for robotics and automation. It handles motion, logic, AI, sensor fusion, etc. Romeo provides a VM with deterministic real-time behavior. Supports tasks that go beyond simple motion commands but also to make it simple to do simple motion applications	More focused: provides a defined set of commands to control robot behavior via PLCs. Less about full custom language features; more about interoperability and standard command semantics.
Real-Time / Safety Guarantees	Designed for real-time. Romeo offers a real-time virtual machine, deterministic execution, automatic memory management, etc.	Supports real-time motion commands insofar as robot controllers and PLCs allow; but standard is more about ensuring consistent command semantics rather than providing its own full language or runtime. Safety aspects are partly in vendors' implementations.
Vendor / Hardware Independence	Designed to be used on "open platforms" (Linux, containerization, etc.), aiming to integrate new and legacy tech. Part of Cognibotics' and its partners' strategy is openness and scalability.	Vendor-agnostic interface by definition. One of the core strengths of SRCI is that you can switch robot brands or PLC brands without rewriting PLC code (if both support SRCI).
Ease of Integration / Commissioning	Emphasizes fast commissioning, modular updates, code reuse, ability for software developers (not just robot specialists) to contribute. More powerful but also more to learn.	Easier for PLC programmers who already know their environment: SRCI provides function blocks/libraries in existing PLC tools. Integration is more straightforward when hardware supports SRCI already. Less ground to cover than introducing new language & runtime. Suffers from interoperability problems as FBs are differently implemented on different control platforms

Cognibotics AB | Ideon Science Park | Scheelevägen 15 | SE-22370 | Lund | SWEDEN | +46 (0) 46 286 59 90 | info@cognibotics.com

Aspect	Juliet & Romeo	SRCI
Scope & Capabilities	Very broad: not just motion, but process logic, AI, sensors, UIs. Built-in language features, possibly richer dynamics. Romeo's runtime handles memory safety, multitasking.	More limited / structured: SRCI defines a fixed (though extensible) command set (Core + optional), focused on robot control, not a general purpose programming environment.

Key Differences and Trade-offs

- **Learning Curve vs Standardization**: Juliet & Romeo likely requires familiarity with the language paradigm, real-time systems, etc. SRCI works with existing PLC programming paradigms, so PLC engineers may adopt it more easily.
- **Flexibility vs Simplicity**: Juliet & Romeo gives more flexibility (expressiveness, real-time tasks, combining motion + logic + AI + sensors + UIs) and applicability at the cost of more complexity. SRCI keeps things simpler with standard commands, less freedom but possibly lower risk.
- **Deployment & Ecosystem Maturity**: SRCI is already adopted by various robot vendors (Yaskawa, Universal Robots, etc.) and PLC ecosystems. Juliet & Romeo is new, launching markets 2025, early partners (e.g. with KEBA, ctrlX OS, etc.).
- **Real-Time Guarantees & Runtime Safety**: Juliet & Romeo emphasizes deterministic and safe real-time execution and automatic memory management. SRCI depends on the underlying robot controller and PLC to provide real-time behavior; it's a protocol/interface, not a runtime language / VM.
- **Ownership / IP / Customization**: With Juliet & Romeo, Cognibotics likely owns the language, the runtime, and the associated artifacts; vendors/users may need to license tools etc. SRCI is a standard, so multiple vendors implement it; IP is distributed according to standard terms, often more open in terms of interface.

Cognibotics AB | Ideon Science Park | Scheelevägen 15 | SE-22370 | Lund | SWEDEN | +46 (0) 46 286 59 90 | info@cognibotics.com

Feature Mapping and Scope

Area	SRCI	Juliet & Romeo	Mapping / Gap
Core motion control	Provides standardized commands: jog, point-to-point, linear moves, stop, pause, resume. Implemented as function blocks in PLCs.	Juliet supports real-time motion control (trajectory, blending, interpolation) executed by Romeo VM.	Overlap: Juliet could *generate* or *wrap* SRCI commands where supported.
Path & trajectory	Optional SRCI profile: predefined paths, path blending, Cartesian space commands.	Juliet: fully programmable paths, with flexible trajectory generation & modification at runtime.	Partial overlap. Juliet offers more expressive path programming (custom logic + AI).
State machine / execution states	SRCI defines a standard robot state machine (idle, running, error, stop, reset, etc.).	Romeo VM enforces deterministic state handling for running Juliet tasks, incl. error handling.	Compatible: Romeo's runtime state model can be aligned with SRCI's state machine.
Error handling	SRCI defines standardized error states and recovery commands.	Romeo ensures safe task termination, exception handling, memory safety.	Similar goals, but Juliet adds stronger runtime safety.
Force / compliance control	SRCI extended profile: force/torque mode, impedance control.	Juliet can integrate sensor feedback (force, vision, AI).	Overlap; Juliet expands flexibility in combining multi-sensor data.
Multi-robot / multi-task coordination	SRCI is single-robot centric (PLC can coordinate multiple robots, but each via SRCI separately).	Juliet/Romeo natively supports parallel tasks and coordination logic across robots/devices.	Juliet adds value beyond SRCI, especially in cell-level orchestration.
PLC integration	SRCI's main strength: standardized FBs in PLC programming environment.	Juliet: standalone language/runtime; integrates with PLCs through APIs, middleware, or potentially via SRCI bridges.	Difference: Juliet is a new programming paradigm, not just PLC FBs.

Cognibotics AB | Ideon Science Park | Scheelevägen 15 | SE-22370 | Lund | SWEDEN | +46 (0) 46 286 59 90 | info@cognibotics.com

Area	SRCI	Juliet & Romeo	Mapping / Gap
Vendor independence	Core mission: all SRCI robots can be controlled the same way from PLCs.	Juliet: vendor-agnostic runtime, runs on Linux/containers, integrates legacy + new systems.	Shared philosophy of openness, but Juliet is broader (not only robot vendors).
Process logic	Minimal: PLC handles process logic, SRCI handles robot motion.	Juliet integrates process logic directly into the same program.	Big gap: Juliet is “full automation programming,” SRCI only robot command.
AI / data handling	Not in SRCI scope.	Juliet explicitly supports AI/ML, data handling, decision-making.	Juliet far beyond SRCI here.
UI / HMI	Outside SRCI scope.	Juliet can define HMIs/UIs alongside logic.	Juliet beyond SRCI.
Extensibility / ecosystem	SRCI profiles extend functionality but remain standardized.	Juliet/Romeo designed as future-proof, extensible language & runtime, container-based.	Both support extension, but Juliet is broader.

Conclusion: The Future of Unified Automation

The relationship between SRCI and Juliet & Romeo is not one of competition, but of strategic integration. SRCI achieves its core mission of providing a reliable, vendor-agnostic interface for motion control from the PLC. However, modern automation systems demand more than simple robot commands. Juliet & Romeo steps into this gap, offering a complete, future-proof programming foundation that encompasses real-time motion, complex process logic, multi-robot coordination, AI integration, and HMI development. By offering two-way compatibility, Juliet & Romeo ensures that users can both serve and benefit from the SRCI standard while gaining a robust and flexible platform for all advanced automation challenges.

Cognibotics AB | Ideon Science Park | Scheelevägen 15 | SE-22370 | Lund | SWEDEN | +46 (0) 46 286 59 90 | info@cognibotics.com