uPLC Structured Text Compiler

Introduction to the ST to IL Converter

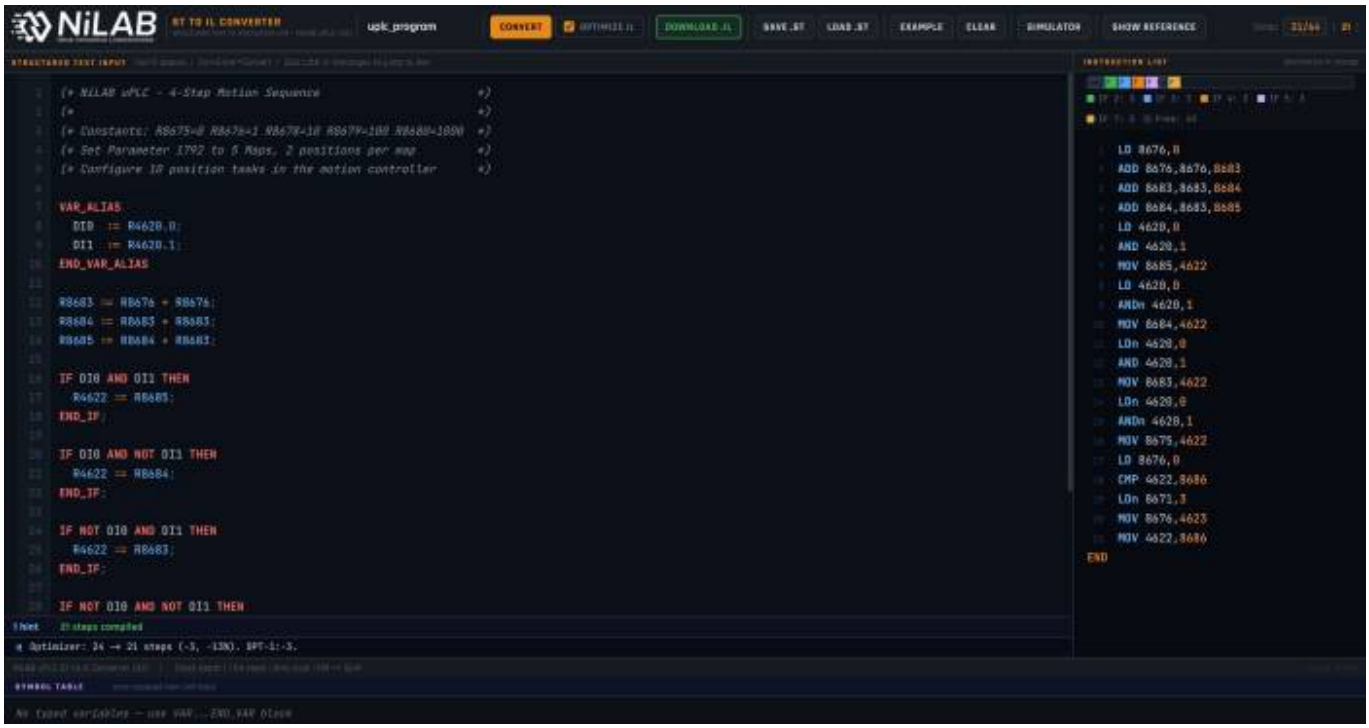
The **ST to IL Converter** is a specialized utility designed to bridge the gap between high-level programming and drive-level execution. This tool allows developers to write control logic using **Structured Text (ST)**—a high-level, Pascal-like language—and automatically convert it into the **Instruction List (IL)** format required by our integrated drives.

This tool is ideal for users who need to implement complex mathematical algorithms, data handling, or conditional loops that are often more efficiently managed via text-based coding than graphical environments.

You can use this tool here: https://www.ni-lab.online/new_websmart/nilab_st_converter.php

How it Works with NiLAB Starter

The converter is fully integrated into the NiLAB development workflow, providing a flexible alternative to the Ladder Editor:



- **Code Development:** Write your control algorithms in the ST editor using standard syntax (IF-THEN-ELSE, CASE, FOR/WHILE loops, etc.).
- **Conversion:** The tool parses the Structured Text and generates an optimized **Instruction List (IL)** file.
- **Integration:** The resulting IL file is loaded into **NiLAB Starter**, which compiles the code specifically for the integrated drive's architecture.
- **Deployment:** Using NiLAB Starter, the compiled program is downloaded directly into the

motor, ready for execution.

Key Advantages

- **Flexible Programming:** Choose the best language for the task—use ST for complex logic and Ladder for sequential control.
- **Seamless Translation:** Automated conversion eliminates manual coding errors when moving from ST to IL.
- **Optimized Performance:** The converter generates clean Instruction List code to ensure efficient execution within the drive's internal uPLC.
- **Unified Workflow:** Both the ST Converter and the Ladder Editor share the same deployment path via NiLAB Starter, providing a consistent user experience.