Automation! v0.0

Prolog

My name is Sandor Vamos. I was born in Hungary, have lived in Germany for approximately 20 years, and now hold German citizenship. I speak Hungarian, German, and English. I am over 50 years old and have about 25 years of experience in programming and designing industrial systems. I mainly design with the market leader in Europe, **Simatic systems**, but I have also worked on many other systems (*AB Rockwell, Bosch, Wago, Codesys, Mitsubishi, …*).

At first, I worked on production lines at Bosch; later, I founded my own company and worked in power plant technology. The Budapest (Rákospalota) Waste Incineration Plant (4 boilers, turbine, district heating) has been operating with my software for more than 20 years.

In 2006, my family and I moved to Germany, where I initially worked on programming trains; for example, I wrote the door control software module for the ICE4 (intercity train). For several years, I have been working on shore power supply for container vessels and cruise ships. In this area, my current references include the Hamburg CTB, the Toll Ports of Melbourne and Burnie, and the Thialf and Sleipnir ships in Rotterdam.

In addition to Simatic systems, I also work extensively with Arduino and ESP32 microcontrollers. I use these primarily for IoT projects, often as an economical complement to Simatic systems. Simatic is an expensive system, and in many cases it is not cost-effective to use; in those cases, its *“little brother”*, Arduino, comes in handy. In addition to control, I implement visualization (SCADA/HMI), typically with WinCC for Simatic systems. For data collection and exceptional communication, I develop IPC solutions in LabVIEW, which, of course, connect to Simatic and Arduino.

I use a wide range of industrial communication systems to service and communicate with the systems: ProfiNet, ProfiBus, (Industrial) Ethernet protocols, Modbus, IEC61850, … - check out my site.

The idea for the online book stems from the fact that I have many notes on *“real”* programming, i.e., the programming I use in practice. These are tricks and procedures that official books rarely present, or present with a significantly different emphasis. I felt that, amid the flood of technical books, a description focused on the practical application of PLCs might have its place.

The book **Automation!** is an ever-expanding online documentation project based primarily on my experiences. I highlight the knowledge I consider essential for programming. I also adjust the language so it reads as if I were explaining to a friend which things are unimportant and which are essential. The online documentation will also be available for download as a book, but I will provide it with a version number, precisely because of the continuous expansion. For now, the version starts with 0, since I have only just started writing this.

Sándor Vámos; lamaPLC.com

To quickly review the content, use the “**Table of contents**” function in the upper-right corner ☝ (on PC).

You can save the entire page content as a **PDF** by clicking the PDF icon in the right menu (Export to PDF).

A few important notes regarding the document (Automation!):

- The document (Automation!) is freely available, but I hold the rights to publish it. Redistribution – even partially – requires my approval.
- The document's content reflects my personal judgment. Clearly, I cannot provide a comprehensive and detailed Simatic documentation, partly due to length constraints.
- The example programmes are written in SCL because it is the language I use for programming.
- Although I have taken great care, there might still be errors in this document. If you find any, please let me know at: info at lamaplc.com.

2026/04/23 21:51

PLC

The first question that comes to mind is: What exactly is a PLC? Is it just a PC, a microcontroller, or a trendy but temporary device?

It’s unlikely to categorize PLCs as modern, short-lived devices. The term PLC stands for programmable logic controller, and in German, it’s called SPS, short for speicherprogrammierbare Steuerung. In 1968, GM Hydramatic, the automatic transmission division of General Motors, sought a solution to replace fixed-wiring relay systems. Bedford Associates of Bedford, Massachusetts, submitted the winning bid. The result of this project was the first PLC, built in 1969 and designated 084, named after Bedford Associates' eighty-fourth project. Interestingly, the first industrial communication protocol, Modbus, was developed simultaneously with the first PLC. Despite being over half a century old, it remains one of the most widely used protocols today (see Modbus RTU/TCP).

Instead of fixed relays, the PLC can perform control dynamically through software, and it must load the following tasks to do so:

- The processor (CPU) interprets inputs, executes the control program stored in memory, and sends output signals
- Power supply
- Memory unit (and data carrier), which stores the contents of inputs and outputs, as well as the programs to be executed by the processor
- Input and output interfaces, where the controller receives and sends data from and to external devices
- Communication interface for receiving and transmitting data over communication networks

Simatic

SIMATIC is a series of programmable logic controllers and automation systems created by Siemens. First launched in 1958, the series has evolved through four major generations, with the most recent being the SIMATIC S7 series. It is designed for industrial automation and manufacturing processes.

The name SIMATIC is a registered trademark of Siemens. It combines the words *“Siemens”* and *“Automatic.”*

In 1959, the Simatic G was introduced, a system designed to replace relay logic with a hard-wired programmable logic controller that used Germanium transistors. It was marketed as the *“Building-Block System for Solid-State Controls.”*

In 1964, the Simatic N was launched, featuring a hard-wired programmable logic controller based on Silicon transistors. 1973 saw the release of the Simatic S3, the first system with a microprocessor that enabled programmable logic.

In 1979, the Simatic S5 series was introduced and remained in use until 2000, with systems still operational worldwide. Programming was done using STEP 5 software. The S5 U (universal) controllers were introduced in 1984.

In 1994, the Simatic S7 series was launched, mainly consisting of the S7-200 for simple automation, S7-300 for general use, and S7-400 for large-scale projects, all supporting networking via Profibus or Industrial Ethernet.

2026/04/23 21:51

2026/01/06 16:16

Automation! - S7 index

URL_name	Name	Description	Readiness status
automation:s7_var	Simatic Variables/Types	Variables, types, addressing, pointer	90 %
automation:s7_hw	Simatic HW basic / PLC Types	HW config, SW structure	2 %
automation:s7_modbus	Simatic and Modbus	Simatic S7 and Modbus communication	1 %

URL_name	Name	Description	Readiness status
automation:s7_scl_commands	Simatic Functions	Standard and system functions	70 %
automation:s7_com	Simatic Communication	Communication	0 %
automation:s7_opc	Simatic OPC UA	Using and operating OPC UA	0 %
automation:s7_ard	Simatic & Arduino	Simatic & Arduino	1 %
automation:s7_iot	Simatic IoT	Simatic & Internet of Things	0 %

2026/03/23 15:40

Simatic, programming, automation!, plc, TIA portal, S7, S7 1500, S7 1200

This page has been accessed for: Today: 2, Until now: 135