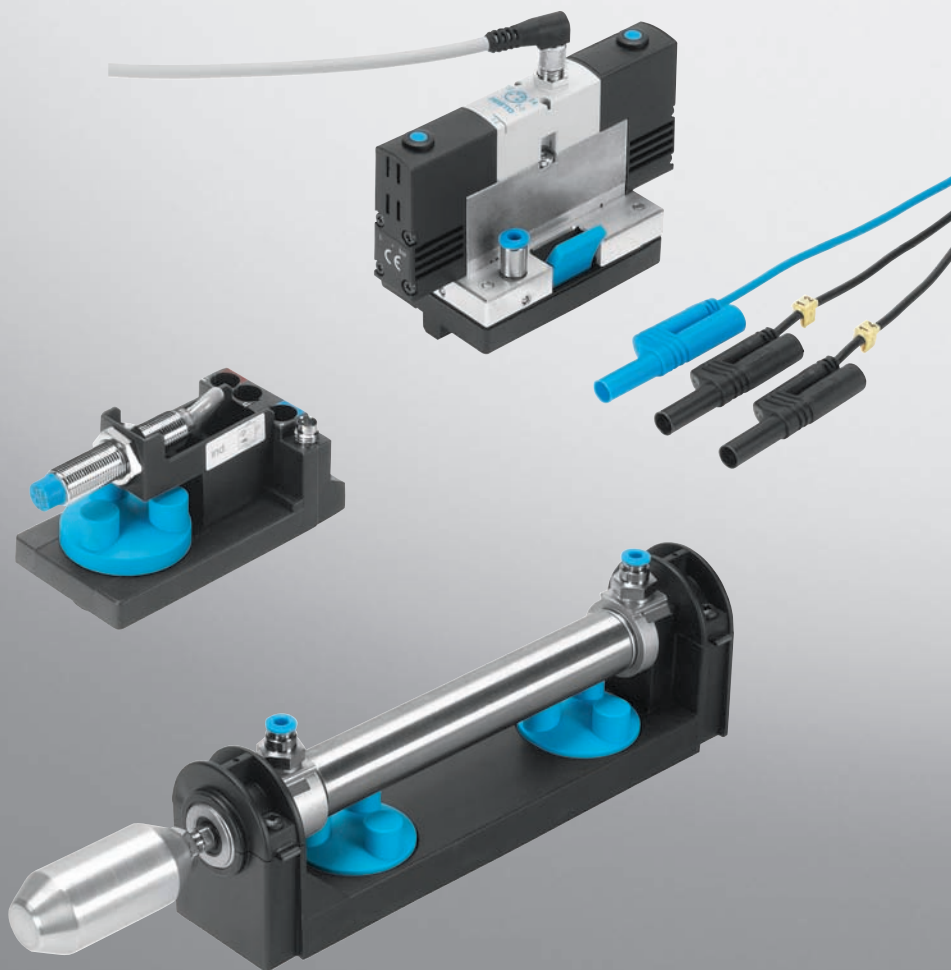


# Programmable Logic Controllers Basic Level

**FESTO**

Textbook TP 301



#### Authorised applications and liability

The Learning System for Automation and Technology has been developed and prepared exclusively for training in the field of automation. The training organization and/or trainee shall ensure that the safety precautions described in the accompanying Technical documentation are fully observed.

Festo Didactic hereby excludes any liability for injury to trainees, to the training organization and/or to third parties occurring as a result of the use or application of the station outside of a pure training situation, unless caused by premeditation or gross negligence on the part of Festo Didactic.

Order No.:	093311
Description:	SPS LB GS
Designation:	D.LB-TP301-1-GB
Edition:	08/2002
Layout:	28.08.2002, OCKER Ingenieurbüro
Graphics:	D. Schwarzenberger, OCKER Ingenieurbüro
Authors:	R. Bliesener, F.Ebel, C.Löffler, B. Plagemann, H.Regber, E.v.Terzi, A. Winter

© Copyright by Festo Didactic GmbH & Co., D-73770 Denkendorf 2002

The copying, distribution and utilization of this document as well as the communication of its contents to others without expressed authorization is prohibited. Offenders will be held liable for the payment of damages. All rights reserved, in particular the right to carry out patent, utility model or ornamental design registrations.

Parts of this training documentation may be duplicated, solely for training purposes, by persons authorised in this sense.

## **Preface**

The programmable logic controller represents a key factor in industrial automation. Its use permits flexible adaptation to varying processes as well as rapid fault finding and error elimination.

This textbook explains the design of a programmable logic controller and its interaction with peripherals.

One of the main focal points of the textbook deals with the new international standard for PLC programming, the EN 61131-3 (IEC-61131-3). This standard takes into account expansions and developments, for which no standardised language elements existed hitherto.

The aim of this new standard is to standardise the design, functionality and the programming of a PLC in such a way as to enable the user to easily operate with different systems.

In the interest of continual further improvement, all readers of this book are invited to make contributions by way suggestions, ideas and constructive criticism.

August 2002

The authors

*B-IV*

---

## **Table of Contents**

<b>Chapter 1 Automating with a PLC</b>		<b>B-1</b>
1.1	Introduction	B-1
1.2	Areas of application of a PLC	B-2
1.3	Basic design of a PLC	B-5
1.4	The new PLC standard EN 61131 (IEC 61131)	B-8
<b>Chapter 2 Fundamentals</b>		<b>B-11</b>
2.1	The decimal number system	B-11
2.2	The binary number system	B-11
2.3	The BCD code	B-13
2.4	The hexadecimal number system	B-13
2.5	Signed binary numbers	B-14
2.6	Real numbers	B-14
2.7	Generation of binary and digital signals	B-15
<b>Chapter 3 Boolean operations</b>		<b>B-19</b>
3.1	Basic logic functions	B-19
3.2	Further logic operations	B-23
3.3	Establishing switching functions	B-25
3.4	Simplification of logic functions	B-28
3.5	Karnaugh-Veitch diagram	B-30

<b>Chapter 4</b>	<b>Design and mode of operation of a PLC</b>	<b>B-33</b>
4.1	Structure of a PLC	B-33
4.2	Central control unit of a PLC	B-35
4.3	Function mode of a PLC	B-37
4.4	Application program memory	B-39
4.5	Input module	B-41
4.6	Output module	B-43
4.7	Programming device/Personal computer	B-45
<b>Chapter 5</b>	<b>Programming of a PLC</b>	<b>B-47</b>
5.1	Systematic solution finding	B-47
5.2	EN 61131-3 (IEC 61131-3) structuring resources	B-50
5.3	Programming languages	B-54
<b>Chapter 6</b>	<b>Common elements of programming languages</b>	<b>B-57</b>
6.1	Resources of a PLC	B-57
6.2	Variables and data types	B-60
6.3	Program	B-70
<b>Chapter 7</b>	<b>Function block diagram</b>	<b>B-85</b>
7.1	Elements of function block diagram	B-85
7.2	Evaluation of networks	B-85
7.3	Loop structures	B-87
<b>Chapter 8</b>	<b>Ladder diagram</b>	<b>B-89</b>
8.1	Elements of ladder diagram	B-89
8.2	Functions and function blocks	B-92
8.3	Evaluation of current rungs	B-93

**Chapter 9 Instruction list** **B-95**

9.1	Instructions	B-95
9.2	Operators	B-96
9.3	Functions and function blocks	B-97

**Chapter 10 Structured text** **B-99**

10.1	Expressions	B-99
10.2	Statements	B-101
10.3	Selection statements	B-103
10.4	Iteration statements	B-106

**Chapter 11 Sequential function chart** **B-111**

11.1	Introduction	B-111
11.2	Elements of sequential function chart	B-111
11.3	Transitions	B-120
11.4	Steps	B-123
11.5	Example	B-135

**Chapter 12 Logic control systems** **B-139**

12.1	What is a logic control system	B-139
12.2	Logic control systems without latching properties	B-139
12.3	Logic control systems with memory function	B-145
12.4	Edge evaluation	B-148

**Chapter 13 Timers** **B-153**

13.1	Introduction	B-153
13.2	Pulse timer	B-154
13.3	Switch-on signal delay	B-156
13.4	Switch-off signal delay	B-158

<b><i>Chapter 14 Counter</i></b>		<b><i>B-161</i></b>
14.1	Counter functions	B-161
14.2	Incremental counter	B-161
14.3	Decremental counter	B-165
14.4	Incremental/decremental counter	B-167
<b><i>Chapter 15 Sequence control systems</i></b>		<b><i>B-169</i></b>
15.1	What is a sequence control system	B-169
15.2	Function chart to IEC 60848	B-169
<b><i>Chapter 16 Commissioning and operational safety of a PLC</i></b>		<b><i>B-175</i></b>
16.1	Commissioning	B-175
16.2	Operational safety of a PLC	B-177
<b><i>Chapter 17 Communication</i></b>		<b><i>B-183</i></b>
17.1	The need for communication	B-183
17.2	Data transmission	B-183
17.3	Interfaces	B-184
17.4	Communication in the field area	B-185
<b><i>Appendix</i></b>		
A	Bibliography of illustrations	B-187
B	Bibliography of literature	B-189
C	Guidelines and standards	B-191
D	Glossary	B-193
E	Index	B-199

## ***The PLC in automation technology***

### **1.1 Introduction**

The first Programmable Logic Controller (PLC) was developed by a group of engineers at General Motors in 1968, when the company were looking for an alternative to replace complex relay control systems.

The new control system had to meet the following requirements:

- Simple programming
- Program changes without system intervention (no internal rewiring)
- Smaller, cheaper and more reliable than corresponding relay control systems
- Simple, low cost maintenance

Subsequent development resulted in a system, which enabled the simple connection of binary signals. The requirements as to how these signals were to be connected were specified in the control program. With the new systems it became possible for the first time to plot signals on a screen and to file these in electronic memories.

Since then, three decades have passed, during which the enormous progress made in the development of microelectronics did not stop short of programmable logic controllers. For instance, even if program optimisation and thus a reduction of required memory capacity initially still represented an important key task for the programmer, nowadays this is hardly of any significance.

Moreover, the range of functions has grown considerably. 15 years ago, process visualisation, analogue processing or even the use of a PLC as a controller, were considered as Utopian. Nowadays, the support of these functions forms an integral part of many PLCs.

The following pages in this introductory chapter outline the basic design of a PLC together with the currently most important tasks and applications.

## 1.2 Areas of application of a PLC

Every system or machine has a controller. Depending on the type of technology used, controllers can be divided into pneumatic, hydraulic, electrical and electronic controllers. Frequently, a combination of different technologies is used. Furthermore, differentiation is made between hard-wired programmable (e.g. wiring of electro-mechanical or electronic components) and programmable logic controllers. The first is used primarily in cases, where any reprogramming by the user is out of the question and the job size warrants the development of a special controller. Typical applications for such controllers can be found in automatic washing machines, video cameras, and cars.

However, if the job size does not warrant the development of a special controller or if the user is to have the facility of making simple or independent program changes, or of setting timers and counters, then the use of a universal controller, where the program is written to an electronic memory, is the preferred option. The PLC represents such a universal controller. It can be used for different applications and, via the program installed in its memory, provides the user with a simple means of changing, extending and optimising control processes.



Fig. B1.1:  
Example of a  
PLC application

The original task of a PLC involved the interconnection of input signals according to a specified program and, if "true", to switch the corresponding output. Boolean algebra forms the mathematical basis for this operation, which recognises precisely two defined statuses of one variable: "0" and "1" (see also chapter 3). Accordingly, an output can only assume these two statuses. For instance, a connected motor could therefore be either switched on or off, i.e. controlled.

This function has coined the name PLC: **Programmable logic controller**, i.e. the input/output behaviour is similar to that of an electromagnetic relay or pneumatic switching valve controller; the program is stored in an electronic memory.

However, the tasks of a PLC have rapidly multiplied: Timer and counter functions, memory setting and resetting, mathematical computing operations all represent functions, which can be executed by practically any of today's PLCs.

The demands to be met by PLC's continued to grow in line with their rapidly spreading usage and the development in automation technology. Visualisation, i.e. the representation of machine statuses such as the control program being executed, via display or monitor. Also controlling, i.e. the facility to intervene in control processes or, alternatively, to make such intervention by unauthorised persons impossible. Very soon, it also became necessary to interconnect and harmonise individual systems controlled via PLC by means of automation technology. Hence a master computer facilitates the means to issue higher-level commands for program processing to several PLC systems

The networking of several PLCs as well as that of a PLC and master computer is effected via special communication interfaces. To this effect, many of the more recent PLCs are compatible with open, standardised bus systems, such as Profibus to EN 50170. Thanks to the enormously increased performance capacity of advanced PLCs, these can even directly assume the function of a master computer.

At the end of the seventies, binary inputs and outputs were finally expanded with the addition of analogue inputs and outputs, since many of today's technical applications require analogue processing (force measurement, speed setting, servo-pneumatic positioning systems). At the same time, the acquisition or output of analogue signals permits an actual/setpoint value comparison and as a result the realisation of automatic control engineering functions, a task, which widely exceeds the scope suggested by the name (programmable logic controller).

The PLCs currently on offer in the market place have been adapted to customer requirements to such an extent that it has become possible to purchase an eminently suitable PLC for virtually any application. As such, miniature PLCs are now available with a minimum number of inputs/outputs starting from just a few hundred Pounds. Also available are larger PLCs with 28 or 256 inputs/outputs.

Many PLCs can be expanded by means of additional input/output, analogue, positioning and communication modules. Special PLCs are available for safety technology, shipping or mining tasks. Yet further PLCs are able to process several programs simultaneously – (multitasking). Finally, PLCs are coupled with other automation components, thus creating considerably wider areas of application.

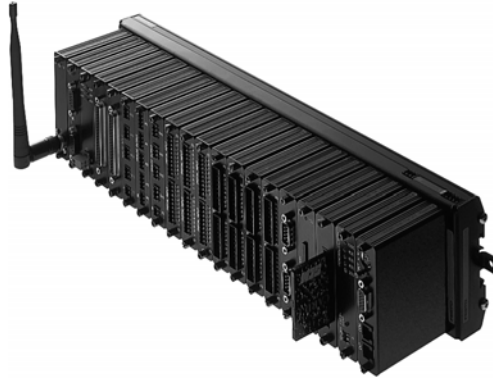


Fig. B1.2:  
Example of a PLC:  
Festo IPC PS1 Professional

### 1.3 Basic design of a PLC

The term 'programmable logic controller' is defined as follows by EN 61131-1 (IEC 61131-1):

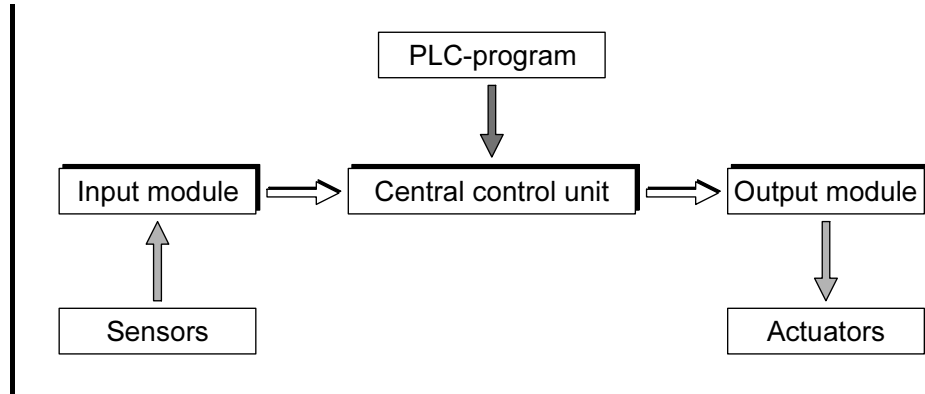
*" A digitally operating electronic system, designed for use in an industrial environment, which uses a programmable memory for the internal storage of user-oriented instructions for implementing specific functions such as logic, sequencing, timing, counting and arithmetic, to control, through digital or analogue inputs and outputs, various types of machines or processes.*

*Both the PC and its associated peripherals are designed so that they can be easily integrated into an industrial control system and easily used in all their intended functions."*

A programmable logic controller is therefore nothing more than a computer, tailored specifically for certain control tasks.

Fig. B1.3 illustrates the system components of a PLC.

Fig. B1.3:  
System components  
of a PLC



The function of an input module is to convert incoming signals into signals, which can be processed by the PLC, and to pass these to the central control unit. The reverse task is performed by an output module. This converts the PLC signal into signals suitable for the actuators.

The actual processing of the signals is effected in the central control unit in accordance with the program stored in the memory.

The program of a PLC can be created in various ways: via assembler-type commands in 'statement list', in higher-level, problem-oriented languages such as structured text or in the form of a flow chart such as represented by a sequential function chart. In Europe, the use of function block diagrams based on function charts with graphic symbols for logic gates is widely used. In America, the 'ladder diagram' is the preferred language by users.

Depending on how the central control unit is connected to the input and output modules, differentiation can be made between compact PLCs (input module, central control unit and output module in one housing) or modular PLCs.

Fig. B1.4 shows the FX0 controller by Mitsubishi and the IPC FEC Standard controller by Festo as an Example



Fig. B1.4:  
Compact-PLC  
(Mitsubishi FX0,  
Festo IPC FEC Standard),  
modular PLC  
(Siemens S7-300)

Modular PLCs may be configured individually. The modules required for the practical application – apart from digital input/output modules, which can, for instance, include analogue, positioning and communication modules – are inserted in a rack, where individual modules are linked via a bus system. This type of design is also known as series technology. Two examples of modular PLCs are shown in figs. B1.2 and B1.4. These represent the modular system IPC PS1 Professional by Festo and the new S7-300 series by Siemens.

A wide range of variants exists, particularly in the case of more recent PLCs. These include both modular as well as compact characteristics and important features such as spacing saving, flexibility and scope for expansion.

The card format PLC is a special type of modular PLC, developed during the last few years. With this type, individual or a number of printed circuit board modules are in a standardised housing.

The hardware design for a programmable logic controller is such that it is able to withstand typical industrial environments as regard signal levels, heat, humidity, and fluctuations in current supply and mechanical impact.

#### **1.4 The new PLC standard EN 61131 (IEC 61131)**

Previously valid PLC standards focussing mainly on PLC programming were generally geared to current state of the art technology in Europe at the end of the seventies. This took into account non-networked PLC systems, which primarily execute logic operations on binary signals.

Previously, no equivalent, standardised language elements existed for the PLC developments and system expansions made in the eighties, such as processing of analogue signals, interconnection of intelligent modules, networked PLC systems etc. Consequently, PLC systems by different manufacturers required entirely different programming.

Since 1992, an international standard now exists for programmable logic controllers and associated peripheral devices (programming and diagnostic tools, testing equipment, man-to-machine interfaces etc.). In this context, a device configured by the user and consisting of the above components is known as a PLC system.

The new EN 61131 (IEC 61131) standard consists of five parts:

- Part 1: General information
- Part 2: Equipment requirements and tests
- Part 3: Programming languages
- Part 4: User guidelines (in preparation with IEC)
- Part 5: Messaging service specification (in preparation with IEC)

Parts 1 to 3 of this standard were adopted unamended as European Standard EN 61 131, Parts 1 to 3.

The purpose of the new standard was to define and standardise the design and functionality of a PLC and the languages required for programming to the extent where users were able to operate using different PLC systems without any particular difficulties.

The next chapters will be dealing with this standard in greater detail. However, for the moment the following information should suffice:

- The new standard takes into account as many aspects as possible regarding the design, application and use of PLC systems.
- The extensive specifications serve to define open, standardised PLC systems.
- Manufacturers must conform to the specifications of this standard both with regard to purely technical requirements for the PLC as well as the programming of controllers.
- Any variations must be fully documented for the user.

After initial reservations, a large group of interested people (PLCopen) has been formed to support this standard. A large number of major PLC suppliers are members of the association, i.e. ABB, GE Fanuc, Mitsubishi Electric, Moeller, OMRON, Schneider Electric, Siemens.

A large number of the members of the association offer control and programming systems conforming to EN 61131 (IEC 61131).

In the future, languages in accordance with IEC 61131 will not only dominate PLC programming, but rather industrial automation in its entirety.

# B-10

---

## Chapter 1