

17. OPEN CONTROLLERS

Topics:

- Open systems
- IEC 61131 standards
- Open architecture controllers

Objectives:

- To understand the decision between choosing proprietary and public standards.
- To understand the basic concepts behind the IEC 61131 standards.

17.1 INTRODUCTION

In previous decades (and now) PLC manufacturers favored “proprietary” or “closed” designs. This gave them control over the technology and customers. Essentially, a proprietary architecture kept some of the details of a system secret. This tended to limit customer choices and options. It was quite common to spend great sums of money to install a control system, and then be unable to perform some simple task because the manufacturer did not sell that type of solution. In these situations customers often had two choices; wait for the next release of the hardware/software and hope for a solution, or pay exorbitant fees to have custom work done by the manufacturer.

“Open” systems have been around for decades, but only recently has their value been recognized. The most significant step occurred in 1981 when IBM broke from its corporate tradition and released a personal computer that could use hardware and software from other companies. Since that time IBM lost control of its child, but it has now adopted the open system philosophy as a core business strategy. All of the details of an open system are available for users and developers to use and modify. This has produced very stable, flexible and inexpensive solutions. Controls manufacturers are also moving toward open systems. One such effort involves Devicenet, which is discussed in a later chapter.

A troubling trend that you should be aware of is that many manufacturers are mislabeling closed and semi-closed systems as open. An easy acid test for this type of system is the question “does the system allow me to choose alternate suppliers for all of the components?” If even one component can only be purchased from a single source, the system is not open. When you have a choice you should avoid “not-so-open” solutions.

17.2 IEC 61131

The IEC 1131 standards were developed to be a common and open framework for PLC architecture, agreed to by many standards groups and manufacturers. They were initially approved in 1992, and since then they have been reviewed as the IEC-61131 standards. The main components of the standard are;

- IEC 61131-1 Overview
- IEC 61131-2 Requirements and Test Procedures
- IEC 61131-3 Data types and programming
- IEC 61131-4 User Guidelines
- IEC 61131-5 Communications
- IEC 61131-7 Fuzzy control

This standard is defined loosely enough so that each manufacturer will be able to keep their own look-and-feel, but the core data representations should become similar. The programming models (IEC 61131-3) have the greatest impact on the user.

- IL (Instruction List) - This is effectively mnemonic programming
- ST (Structured Text) - A BASIC like programming language
- LD (Ladder Diagram) - Relay logic diagram based programming
- FBD (Function Block Diagram) - A graphical dataflow programming method
- SFC (Sequential Function Charts) - A graphical method for structuring programs

Most manufacturers already support most of these models, except Function Block programming. The programming model also describes standard functions and models. Most of the functions in the models are similar to the functions described in this book. The standard data types are shown in Figure 17.1.

Name	Type	Bits	Range
BOOL	boolean	1	0 to 1
SINT	short integer	8	-128 to 127
INT	integer	16	-32768 to 32767
DINT	double integer	32	-2.1e9 to 2.1e9
LINT	long integer	64	-9.2e19 to 9.2e19
USINT	unsigned short integer	8	0 to 255
UINT	unsigned integer	16	0 to 65536
UDINT	unsigned double integer	32	0 to 4.3e9
ULINT	unsigned long integer	64	0 to 1.8e20
REAL	real numbers	32	
LREAL	long reals	64	
TIME	duration	not fixed	not fixed
DATE	date	not fixed	not fixed
TIME_OF_DAY, TOD	time	not fixed	not fixed
DATE_AND_TIME, DT	date and time	not fixed	not fixed
STRING	string	variable	variable
BYTE	8 bits	8	NA
WORD	16 bits	16	NA
DWORD	32 bits	32	NA
LWORD	64 bits	64	NA

Figure 17.1 IEC 61131-3 Data Types

Previous chapters have described Ladder Logic (LD) programming in detail, and Sequential Function Chart (SFC) programming briefly. Following chapters will discuss Instruction List (IL), Structured Text (ST) and Function Block Diagram (FBD) programming in greater detail.

17.3 OPEN ARCHITECTURE CONTROLLERS

Personal computers have been driving the open architecture revolution. A personal computer is capable of replacing a PLC, given the right input and output components. As a result there have been many companies developing products to do control using the personal computer architecture. Most of these devices use two basic variations;

- a standard personal computer with a normal operating system, such as Windows NT, runs a virtual PLC.

- the computer is connected to a normal PLC rack
- I/O cards are used in the computer to control input/output functions
- the computer is networked to various sensors
- a miniaturized personal computer is put into a PLC rack running a virtual PLC.

In all cases the system is running a standard operating system, with some connection to rugged input and output cards. The PLC functions are performed by a virtual PLC that interprets the ladder logic and simulates a PLC. These can be fast, and more capable than a stand alone PLC, but also prone to the reliability problems of normal computers. For example, if an employee installs and runs a game on the control computer, the controller may act erratically, or stop working completely. Solutions to these problems are being developed, and the stability problem should be solved in the near future.

17.4 SUMMARY

- Open systems can be replaced with software or hardware from a third party.
- Some companies call products open incorrectly.
- The IEC 61131 standard encourages interchangeable systems.
- Open architecture controllers replace a PLC with a computer.

17.5 PRACTICE PROBLEMS

1. Describe why traditional PLC racks are not 'open'.
2. Discuss why the IEC 61131 standards should lead to open architecture control systems.

17.6 PRACTICE PROBLEM SOLUTIONS

1. The hardware and software are only sold by Allen Bradley, and users are not given details to modify or change the hardware and software.
2. The IEC standards are a first step to make programming methods between PLCs the same. The standard does not make programming uniform across all programming platforms, so it is not yet ready to develop completely portable controller programs and hardware.

17.7 ASSIGNMENT PROBLEMS

1. Write a ladder logic program to perform the function outlined below. (Hint: use a structured

technique.)

- i) when the input 'part' turns on, the value 'weight' should be added to an array in memory.
- ii) if any 'weight' value is greater than 15, and output 'halt' should be turned on, and the process should stop. A 'reset' input will be turned on to clear the array and start the process again.
- iii) when 'part' has been activated 10 times the median of the part weights should be found. If it is greater than 14 the process should be stopped as described in step ii).
- iv) if the median is less than or equal to 14, then a 'dump' output should be turned on for 2 seconds. After that the matrix should be reset and the process should begin again.