



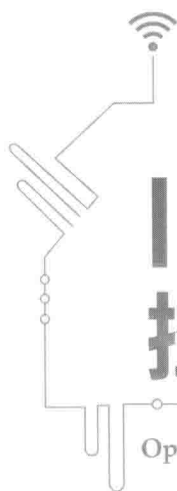
开放式 IEC 61131 控制系统设计

Open Architecture IEC 61131 Control System Design

任向阳 编著



机械工业出版社
China Machine Press



开放式 IEC 61131 控制系统设计

Open Architecture IEC 61131 Control System Design

任向阳 编著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

开放式 IEC 61131 控制系统设计 / 任向阳编著. —北京: 机械工业出版社, 2016.9
(工业控制与智能制造丛书)

ISBN 978-7-111-54945-1

I. 开… II. 任… III. 工业控制系统 - 控制系统设计 IV. TB4

中国版本图书馆 CIP 数据核字 (2016) 第 229012 号

开放式 IEC 61131 控制系统设计

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 张梦玲

责任校对: 董纪丽

印 刷: 三河市宏图印务有限公司

版 次: 2016 年 10 月第 1 版第 1 次印刷

开 本: 186mm × 240mm 1/16

印 张: 14.25

书 号: ISBN 978-7-111-54945-1

定 价: 59.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿邮箱: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

工业自动化作为国民经济的支柱之一，在提高工业企业生产效率、保证产品质量等方面发挥了重要作用。随着现代信息技术和电子技术的飞速发展，其在传统自动化领域的应用极大地提高了控制系统的性能和应用范围，具体表现在以下几个方面：

- 可编程控制器（PLC）和 HMI 面板的处理速度大幅度提高，这大大改进了工业生产自动检测水平、执行精确度与速度，达到了优化装置和过程的效果。
- 以实时以太网为基础的现场总线技术，如 PROFINET、EtherCAT、Ethernet/IP 等使得控制系统与现场设备之间的通信更加可靠、高效。
- 现代网络通信技术可以把整个企业的资金、物流、生产装置状态、生产效率和能力等信息准确、全面、系统地提供给企业，供企业决策者和管理者做实时和准确的决策，给用户 户提供管理和控制一体化设备的系统和服务。

信息化和工业化的结合必然为工业自动化产品制造和应用带来很大的发展空间，同时，随着经济的发展，社会生产组织形式越来越倾向于小批量、定制化的柔性生产方式。基于以上两个因素，德国为了保持其在制造领域的优势，在政府层面提出了“工业 4.0”战略；美国则希望由强势的信息产业优势向传统制造领域回归，提出了“工业互联网”的概念；中国则相应地提出了自己的“中国制造 2025”战略。但是这些宏观的战略都需要得到现场级自动化技术的强力支撑，如现代 PLC 技术、现代现场总线技术和智能机器人等，否则就如同在沙滩上建立大厦一般不稳固。

自 1968 年美国 GM（通用汽车）公司提出取代继电器控制装置的要求以来，随着集成电路技术和计算机技术的发展，现在已出现第五代 PLC 产品了。20 世纪 80 年代至 90 年代中期是 PLC 发展最快的时期，年增长率一直保持为 30% ~ 40%。PLC 由于在模拟处理能力和网络方面的进步，挤占了一部分 DCS（过程控制）的市场，并逐渐垄断了污水处理等行业，但是由

于工业 PC (IPC) 的出现,特别是近年来现场总线技术的发展,IPC 和 DCS 也挤占了一部分 PLC 市场,总体来说 DCS 和 PLC 有相互融合的趋势。目前,全世界有 200 多家工厂生产 300 多个品种的 PLC 产品,主要应用在汽车、粮食加工、化学/制药、金属/矿山、纸浆/造纸等行业。

由于各 PLC 厂商的产品在指令系统上的差异以及在编程方法上对用户的要求不同,近年来为了规范化,IEC 针对 PLC 规定了一系列标准,即 IEC 61131 标准。1995 年,与国际标准等效的国家标准相继颁布,2006 年,IEC 61131 最新国际标准的中文对照版 GB/T 15969 出版。其中第一部分规定了 PLC 系统的定义。IEC 61131-3 是该系列标准中的第三部分,主要涉及 PLC 编程语言的语法和语义定义。它规定了指令表、梯形图、顺序功能图、功能块图、结构化文本 5 种编程语言。这包括文本化编程和图形编程两个方面,SFC 在这两类编程语言中均可使用。

IEC 61131 标准是不同的 PLC 厂商相互协调、讨论并妥协的产物,因此该标准中也有很多与具体实现相关的功能定义。对于这些功能,不同的厂商可采用不同的策略来实现。这既增加了标准的灵活性,又减弱了标准的非二义性。

本书主要目的不在于详细阐述 IEC 61131 标准的相关内容,而在于以德国菲尼克斯电气软件公司的 MULTIPROG/ProConOS eCLR 控制软件为例,展示如何构建一个符合 IEC 61131 标准的实际控制系统。至于 IEC 61131 的详细内容,用户可以直接参考国际标准或者等效国标 GB/T 15969。至于如何在生产实际中使用 IEC 61131-3 所定义的编程语言对 PLC 进行编程控制,用户可以参考其他的一些书籍或文章,如彭瑜先生所著《IEC 61131-3 编程语言及应用基础》^[1]、Karl-Heinz John 先生参与撰写的《IEC 61131-3 Programming Industrial Automation System》^[2]等。在本书中,各章内容安排如下:

- 第 1 章简要介绍 IEC 61131 标准所定义的各种模型以及标准编程语言。
- 第 2 章主要涉及 PLC 编程环境 MULTIPROG 的操作使用,即如何在 MULTIPROG 中实现 IEC 61131 所定义的各项要素。
- 第 3 章重点介绍基于 ProConOS eCLR 的二次开发。ProConOS eCLR 只是实现了一个基本的 PLC 运行时内核,要构建一个完整的 PLC 系统,用户在这个基础上还需要添加额外的模块,比如能完成特定功能的自定义功能块和 I/O 模块的相应驱动等。
- 第 4 章主要描述了 eCLR 与不同现场总线的协同工作,如 PROFINET、EtherCAT、Modbus TCP、CANopen 和 Powerlink 等。

- 第 5 章介绍 MULTIPROG/ProConOS eCLR 与 PLCopen 运动控制的集成。

谨此机会，感谢我的家人在本书的撰写过程中给予的长期支持。另外，假如没有 PLCopen 中国组织名誉主席彭瑜老师和菲尼克斯电气软件公司的同事及好友梁恩泉的大力协助，本书恐怕也难以写成；同时还要感谢菲尼克斯电气软件、acontis technologies GmbH、ISG Industrielle Steuerungstechnik GmbH 以及 Tenesys 等公司在写作过程中对我提供的帮助。

任向阳

2016 年 8 月于德国比勒菲尔德

声明 | Statement

IEC 61131 是国际电工委员会 (IEC) 颁布的可编程控制器 (PLC) 国际标准, 用于规范可编程控制器编程工具和应用控制程序的开发。对应的等效国标为 GB/T 15969。

PLCopen MC 运动规范为 PLCopen 国际组织以 IEC 61131-3 为基础建立的标准运动控制应用功能块库。

本书所涉及的 MULTIPROG/ProConOS eCLR 为德国菲尼克斯电气软件公司的注册商标, 与其相关的信息、图片和资料的版权均归菲尼克斯电气软件公司所有。

EcMaster 和 EC-Engineer 为 acontis technologies GmbH 公司的 EtherCAT 主站和总线配置器产品, 及本书中所涉及的 EcMaster 和 EC-Engineer 相关资料版权均均为 acontis technologies GmbH 公司所有。

INtime 为 Tenasys 公司注册商标, 本文中所有与 INtime 相关的信息版权均归 Tenasys 公司所有。

VirtuOS 和 ISG 运动内核为 ISG Industrielle Steuerungstechnik GmbH 产品, ISG Industrielle Steuerungstechnik GmbH 拥有本书涉及 VirtuOS 和 ISG 运动内核部分所有信息的全部版权。

若本书对上述内容的描述与国际标准 / 公司产品说明有冲突, 请以国际标准 / 公司产品说明为准。

前 言
声 明

第 1 章 IEC 61131 标准简介	1
1.1 PLC 中的一些通用模型	2
1.1.1 功能模型	2
1.1.2 硬件模型	3
1.1.3 网络通信模型	3
1.1.4 软件模型	4
1.1.5 通信模型	5
1.1.6 编程模型	6
1.2 数据类型	7
1.2.1 基本数据类型	7
1.2.2 派生数据类型	8
1.3 变量	10
1.3.1 变量的表示	10
1.3.2 变量的初始化	11
1.3.3 变量的声明	11
1.4 程序组织单元	12
1.4.1 功能	13

1.4.2	功能块	16
1.4.3	程序	19
1.5	顺序功能图元素	19
1.5.1	步	20
1.5.2	转换	20
1.5.3	动作	20
1.6	配置元素	21
1.6.1	配置、资源和全局变量	21
1.6.2	任务	22
1.7	IEC 61131-3 编程语言	23
1.7.1	指令表	23
1.7.2	结构化文本	24
1.7.3	功能块图	26
1.7.4	梯形图	26
1.7.5	顺序功能图	28
第 2 章	IEC 61131 编程环境 MULTIPROG	31
2.1	MULTIPROG 5.50 Express 及其安装	32
2.2	MULTIPROG 工程管理	36
2.2.1	新建并编辑工程	36
2.2.2	I/O 配置	40
2.2.3	工程编译和下装	42
2.3	自定义用户库开发	46
2.3.1	自定义复杂数据类型	46
2.3.2	用户库开发	48
2.4	断点设置和在线调试	48
2.4.1	在线监视	49

2.4.2	断点调试	50
2.4.3	强制和覆盖变量	52
2.5	逻辑分析仪	53
2.5.1	变量添加	53
2.5.2	连续和限定周期记录	55
2.5.3	触发条件配置	56
2.5.4	所记录数据的导出和加载	59
2.6	交叉参考	60
2.7	MULTIPROG 编程示例	61
2.7.1	三相异步电动机正反转控制程序	61
2.7.2	PID 控制器和系统仿真	63
第 3 章	IEC 61131 运行时系统 ProConOS eCLR 及二次开发	66
3.1	eCLR 简介	67
3.2	ProConOS eCLR 架构	68
3.3	eCLR 应用程序映像	71
3.4	简单的 eCLR 及 CIL 代码示例	72
3.5	eCLR 开发包	74
3.5.1	多核处理器支持	75
3.5.2	任务优先级	76
3.5.3	eCLR 启动模块	78
3.6	eCLR 在不同平台上的移植	82
3.6.1	环境接口	83
3.6.2	文件接口	84
3.6.3	操作系统接口	85
3.7	用户自定义功能和功能块	87

3.7.1	托管固件库开发	89
3.7.2	原生固件库开发	94
3.7.3	复杂数据类型	98
3.8	用户自定义 I/O 驱动	109
3.8.1	I/O 驱动工作原理	109
3.8.2	I/O 配置和 I/O 分组	110
3.8.3	I/O 变量	111
3.8.4	I/O 驱动接口	112
3.8.5	I/O 驱动的时序	116
3.8.6	I/O 驱动开发步骤	117
3.9	后台任务	121
3.10	系统标志区和共享内存区	122
3.10.1	系统标志区	122
3.10.2	共享内存区	125
3.11	回调函数的接口	128
3.11.1	状态转换回调函数和异常回调函数的接口	128
3.11.2	任务回调函数	131
3.12	事件驱动型任务	134
3.12.1	MULTIPROG 的事件驱动型任务设置	135
3.12.2	运行期中事件任务安装	136
3.12.3	运行期中系统任务安装	136
第 4 章	IEC 61131 和工业现场总线	138
4.1	PLC 与 PROFINET	139
4.1.1	PROFINET 简介	139
4.1.2	PROFINET 解决方案	145
4.1.3	eCLR PROFINET I/O 驱动	145
4.1.4	PROFINET 总线配置	146

4.2	PLC 与 EtherCAT	154
4.2.1	Acontis EtherCAT 主站简介	154
4.2.2	eCLR ECAT I/O 驱动	157
4.2.3	总线配置	158
4.2.4	基于功能块的总线诊断	165
4.3	PLC 与 Modbus TCP	166
4.3.1	Modbus TCP 简介	166
4.3.2	OSCAT 库	169
4.3.3	基于 IP 功能块和 OSCAT 库的 Modbus TCP 解决方案	170
4.4	PLC 与其他现场总线	173
4.4.1	PLC 和 CANopen 简介	173
4.4.2	PLC 和 Powerlink	175
第 5 章	PLCopen 运动控制	176
5.1	PLCopen Part1/Part2 简单运动控制	178
5.1.1	PLCopen MC 状态机	178
5.1.2	异常和错误处理	179
5.1.3	功能块接口	179
5.1.4	Part1/Part2 运动控制功能块	180
5.2	PLCopen Part4 组合运动控制	182
5.2.1	坐标系和运动学	183
5.2.2	运动混成	185
5.2.3	状态机	186
5.2.4	功能块定义	187
5.3	基于 ProConOS eCLR 和 ISG 内核的运动控制方案	189
5.3.1	实时操作系统 INtime	189
5.3.2	INtime 下 Acontis EtherCAT 主站安装和配置	195
5.3.3	PLC 运行期 ProConOS eCLR	197

5.3.4 IEC 61131 编程环境配置	197
5.3.5 ISG 运动内核	198
5.3.6 启动顺序	199
5.3.7 PLCopen Part1/Part2 运动方案	199
5.3.8 Part4 和机器人方案	203
5.4 ProConOS eCLR 与第三方运动内核的集成	208
5.4.1 PLCopen 运动功能块定义	208
5.4.2 PLC 和运动内核的数据交换	210
5.4.3 PLC、运动内核和电动机驱动器的数据交换	211
5.4.4 系统优先级设定	213
参考文献	214

IEC 61131 标准简介



1.1 PLC 中的一些通用模型

本章主要涉及 IEC 61131-3 中定义的一些通用模型。对这些模型的讨论有助于读者将 PLC 中不同的硬件模块与 IEC 标准中抽象的定义相联系,以获得一个感性的认识。读者同时还了解到 PLC 内部以及 PLC 和 PLC 之间如何相互通信和交换数据。本节所引用的图表大多来自 IEC 61131-3 标准第 2 版^[3]和 MULTIPROG 在线帮助文档^[4]。至于标准第 3 版中所提出的一些新概念,如面向对象编程等,目前由于缺乏编程系统的支持,本书不涉及。

1.1.1 功能模型

一个标准 PLC 设备应当具备一些特定的功能。如图 1-1 所示描述了一个典型 PLC 所必须拥有的功能子系统。

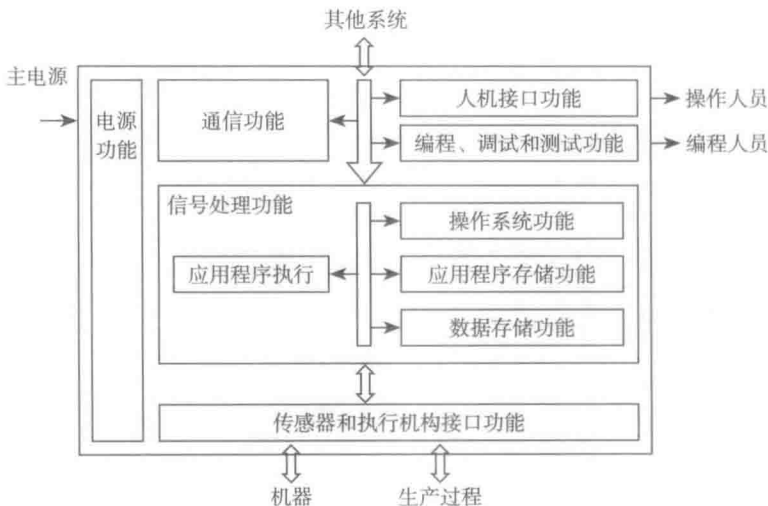


图 1-1 PLC 的功能模型

这些功能可以分为两类：需要与外界交互的功能和内部功能。前者包括电源功能、通信功能、人机接口功能，程序编制、调试和测试功能，以及到传感器和执行器的接口功能；后者包括信号处理功能、操作系统功能、应用程序存储功能和数据存储功能。电源功能将外部主电源转化为系统内所需要的电压或电流。PLC 设备操作员通过人机接口功能实现设备的起停和参数设定等交互。应用程序编制人员则利用程序编制、调试和测试功能进行 PLC 应用程序的编制、下装、调试和测试。传感器和执行器的接口功能可以从机器或生产过程中读取必要的信息，或向其执行器输出必要的动作，其 I/O 可以是本地的也可以是远程的。信号处理功能是 PLC 系统的核心功能，它决定了 PLC 系统所能处理的 I/O 点数和处理周期的快慢。

操作系统功能、应用程序存储功能和数据存储功能则为信号处理功能的辅助功能，分别用于访问操作系统服务、存储应用程序、数据存储。

除此以外，PLC 系统还有一个编程和调试工具的功能（PADT, Programming And Debugging Tool），该功能对于 PLC 系统来说是一个外部的功能，它通过通信功能实现与 PLC 系统的交互。

1.1.2 硬件模型

PLC 系统的硬件主要包括 3 个部分：主处理单元、远程 I/O 从站和外围设备，如图 1-2 所示。其中主处理单元又可以细分为不同的模块，比如内存和处理单元模块、输入模块、输出模块、通信模块、电源模块以及厂商自定义子系统模块。这些模块和远程 I/O 从站和外围设备相互交互，从而实现系统的控制功能。

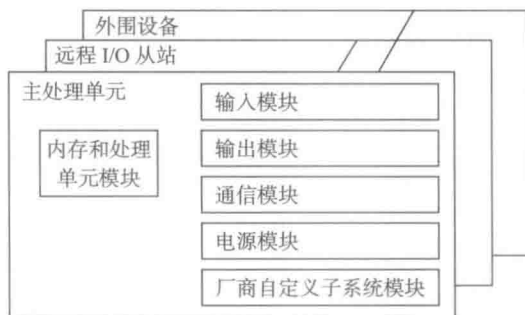


图 1-2 PLC 硬件模型

1.1.3 网络通信模型

一个 PLC 还为控制系统的其余部分提供了一些特定的应用层功能。同时它也可能要求系统中其他控制器提供某些功能。这些信息的交换都是通过 PLC 的网络通信系统来实现的。

图 1-3 描述了一个 PLC 通信网络的例子。该网络中有一个监测控制器，一个非控制器终端系统和两个 PLC：控制器 1 和控制器 2。其中控制器 2 需要实现服务器功能，而其他的设备则需要实现客户端功能，它们之间通过通信系统来实现信息的交换。值得注意的是在这个例子中，从通信的角度来说，监测控制器和非控制器终端系统都在向服务器（控制器 2）发送请求。

PLC 可以与任何设备通过客户端功能通信，只要该设备的行为与 PLC 相类似。

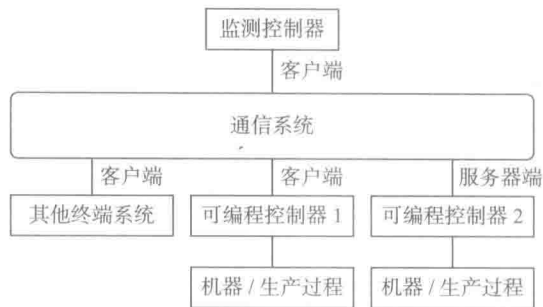


图 1-3 PLC 通信模型

1.1.4 软件模型

根据 IEC 61131 标准，一个 PLC 的软件模型（见图 1-4）大致可以分为 3 个部分：控制序列部分、配置部分和实例相关初始化部分。控制序列包括采用 IEC 61131 标准程序设计语言编写的程序（Program）和功能块（Function Block）等。配置部分则由配置（Configuration）、资源（Resource）、任务（Task）、全局变量（Global Variable）和访问路径（Access Path）组成。实例相关的初始化部分主要负责将编写好的 PLC 控制程序下载到 PLC 系统中运行。

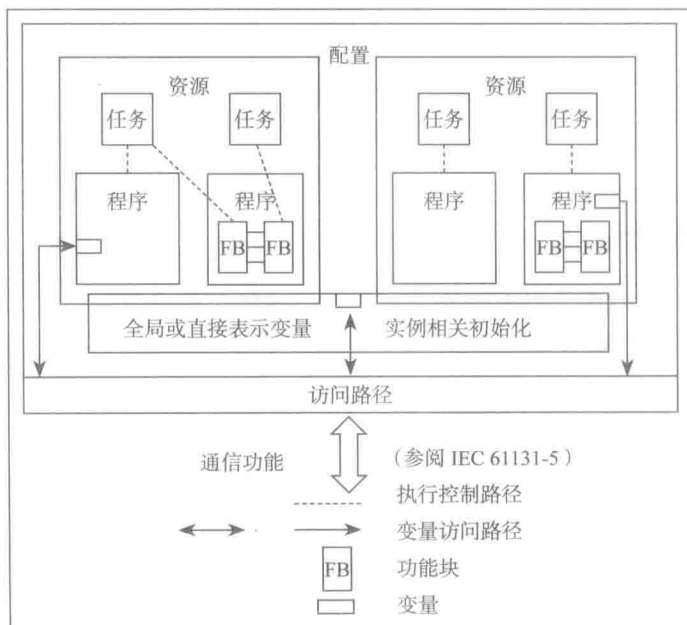


图 1-4 PLC 软件模型