

全国高等职业教育规划教材

可编程控制器原理及应用

第2版

主编 田淑珍

参编 孙建东 王延忠

主审 李 丽



机械工业出版社

本书作为高等职业教育中可编程序控制器的教材，充分体现了高等职业教育培养技能型人才的教学特色。

本书共分为 8 章，第 1 章~第 3 章介绍可编程序控制器的基础知识、结构和编程软件的使用及实训；第 4 章~第 5 章介绍 PLC 的基本指令及应用，第 6 章介绍 PLC 的功能指令及指令向导的应用，在常用指令后配有例题、实训，由浅入深，培养读者兴趣；第 7 章通过综合实例和实训，介绍 PLC 应用系统的设计，提高读者技能；第 8 章介绍 S7-200 的通信与网络，并重点介绍了 PPI 通信及 NETR/NETW 指令及向导的应用，并配有实训。每章后都有习题，既可做课堂教学及书面练习，也可供上机实际操作用。

本书适合高职高专自动化、机电一体化、计算机控制等相关专业的教学使用，也可供 S7-200 系列 PLC 用户参考，同时还可作为从事相关专业的技术人员的自学用书。

本书配备授课电子教案，需要的教师可登录 www.cmpedu.com 免费注册、审核通过后下载，或联系编辑索取（QQ：1239258369，电话：010-88379739）。

图书在版编目（CIP）数据

可编程控制器原理及应用 / 田淑珍主编. —2 版. —北京：机械工业出版社，2014.4

全国高等职业教育规划教材

ISBN 978-7-111-46014-5

I. ①可… II. ①田… III. ①可编程序控制器—高等职业教育—教材
IV. ①TP332.3

中国版本图书馆 CIP 数据核字（2014）第 036600 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）

责任编辑：王 颖

责任印制：

印刷（ 装订）

2014 年 6 月第 2 版·第 1 次印刷

184mm×260mm·15.5 印张·381 千字

0001—3000 册

标准书号：ISBN 978-7-111-46014-5

定价：33.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务

网络服务

社服务中心：(010) 88361066

教材网：<http://www.cmpedu.com>

销售一部：(010) 68326294

机工官网：<http://www.cmpbook.com>

销售二部：(010) 88379649

机工官博：<http://weibo.com/cmp1952>

读者购书热线：(010) 88379203

封面无防伪标均为盗版

前 言

可编程序控制器的原理及应用是从事自动控制及机电一体化专业工作的技术人员不可缺少的重要技能。在许多高职院校已将其作为一门主要的实用性很强的专业课。西门子公司的可编程序控制器在我国的应用市场中占有一定的份额，特别是 S7-200 系列的 CPU21X 和 CPU22X 系列有着广泛的应用，因其结构紧凑、功能强、易于扩展以及性价比高等方面的因素，被许多高职院校作为教学用机。我们编写这本以培养综合性技能型兼顾应用型人才为目标的“讲、练、用”结合的教材，在理论够用条件下，突出实训教学环节，力图做到便于教学，突出职业教育的特点。本书在第 1 版的基础上，将编程软件升级到 STEP7 V4.0 版本，同时强化了 PID、高速计数器、高速脉冲输出和通信指令及其指令向导的应用及实训。

本书重点介绍了 S7-200 系列 PLC 的组成、原理、指令和应用，详细介绍了 PLC 的编程方法，并列举了大量应用示例。为了突出职业教育的特点，在介绍常用指令后都配有例题、实训，由浅入深地培养学生的学习兴趣，并通过综合实例和实训，介绍 PLC 应用系统的设计，提高学生的技能。全书编写体现了讲练结合、工学结合，淡化了理论和实践的界限。在内容安排上，精练理论，突出实用技能，基本概念准确，基本原理简单易懂，并以有趣实用的例子和“看得见、摸得着”的实训介绍可编程序控制器的编程和调试等应用方法，让学生学会应用 PLC 实现一定的控制任务，提高应用技能。

本书既可供少学时（如 40~50 学时）的教学使用，也可供多学时（如 70~80 学时）的教学使用。少学时教学可以将第 1~5 章作为重点详细介绍，有条件可多安排一些实训，而对第 6、8 章可作简单介绍，可有选择地重点讲解第 7 章，并安排实训，让学生实现一定的控制任务。

对第 3 章关于 STEP7 编程软件一章讲解，可以根据教学内容和实训内容的需要合理安排，最好是“现用现讲，用多少讲多少”，特别是要与实训内容交织在一起讲，通过上机练习，教学效果更好。

本书由田淑珍主编，并编写第 1、2、3、4、6、7 章，孙建东编写第 8 章，王延忠编写第 5 章和附录并进行图文处理工作。全书由田淑珍整理定稿，由李丽主审，在此表示感谢。

由于编者水平有限，书中错漏在所难免，恳请广大读者批评指正。

编 者

目 录

前言

第 1 章 可编程序控制器概述	1
1.1 可编程序控制器的发展概况	1
1.2 可编程序控制器的定义	1
1.3 可编程序控制器的基本组成	2
1.3.1 控制组件	2
1.3.2 输入/输出接口电路	4
1.3.3 编程器	6
1.4 可编程序控制器的工作原理及主要技术指标	7
1.4.1 可编程序控制器的工作原理	7
1.4.2 可编程序控制器的主要技术指标	7
1.5 可编程序控制器的分类、特点、应用及发展	8
1.5.1 可编程序控制器的分类	8
1.5.2 可编程序控制器的特点	9
1.5.3 可编程序控制器的应用	9
1.5.4 可编程序控制器的发展	10
1.6 习题	11
第 2 章 西门子 S7-200 系列可编程序控制器	12
2.1 S7-200 系列 PLC 概述	12
2.2 S7-200 系列 CPU224 型 PLC 的结构	13
2.2.1 CPU224 型 PLC 的外形、端子及接线	13
2.2.2 CPU224 型 PLC 的结构及性能指标	16
2.2.3 PLC 的 CPU 工作方式	18
2.3 扩展功能模块	18
2.3.1 扩展单元及电源模块	18
2.3.2 常用扩展模块	19
2.4 S7-200 系列 PLC 数据存储类型、编址和寻址方式、元器件功能及地址分配	22
2.4.1 数据存储类型	22
2.4.2 编址方式	23
2.4.3 寻址方式	24
2.4.4 元器件功能及地址分配	25
2.5 习题	29

第 3 章	STEP7 V4.0 编程软件	30
3.1	STEP7 V4.0 编程软件概述	30
3.1.1	通信设置	30
3.1.2	STEP7-Micro/Win V4.0 SP9 窗口组件	33
3.1.3	编程准备	39
3.2	STEP7-Micro/Win 主要编程功能	39
3.2.1	梯形图程序的输入	39
3.2.2	数据块的编辑	41
3.2.3	符号表操作	41
3.3	程序的下载和上载	42
3.4	程序的调试与监控	43
3.4.1	选择工作方式	43
3.4.2	显示程序状态	44
3.4.3	显示状态表	45
3.4.4	执行有限次扫描	46
3.5	编程软件实训	47
3.6	习题	49
第 4 章	S7-200 系列 PLC 基本指令及实训	50
4.1	可编程序控制器程序设计语言	50
4.2	基本位逻辑指令与应用	52
4.2.1	基本位操作指令	52
4.2.2	基本位逻辑指令应用举例	60
4.2.3	编程注意事项及编程技巧	64
4.2.4	电动机控制实训	66
4.3	定时器指令与应用	68
4.3.1	定时器指令	68
4.3.2	定时器指令应用举例	71
4.3.3	正次品分拣机编程实训	73
4.3.4	用定时器指令编写循环类程序	75
4.3.5	交通灯的控制编程实训	76
4.4	计数器指令与应用	77
4.4.1	计数器指令	77
4.4.2	计数器指令应用举例	79
4.4.3	轧钢机的控制实训	82
4.5	比较指令	83
4.6	程序控制类指令	85
4.6.1	跳转及标号指令	85
4.6.2	子程序调用及子程序返回指令	87

4.6.3	步进顺序控制指令	91
4.6.4	送料车控制实训	93
4.7	习题	98
第 5 章	数据处理、运算指令及应用	101
5.1	数据处理指令	101
5.1.1	数据传送指令	101
5.1.2	移位指令及应用举例	102
5.1.3	转换指令	109
5.1.4	天塔之光的模拟控制实训	113
5.2	算术运算、逻辑运算和递增/递减指令及实训	115
5.2.1	算术运算指令	115
5.2.2	逻辑运算指令	119
5.2.3	递增/递减指令	121
5.2.4	运算单位转换实训	122
5.2.5	控制小车的运行方向实训	123
5.3	字填充指令	125
5.4	习题	125
第 6 章	特殊功能指令	127
6.1	中断指令	127
6.1.1	中断源	127
6.1.2	中断指令	129
6.1.3	中断程序	129
6.1.4	程序举例	130
6.1.5	中断程序编程实训	132
6.2	高速计数器与高速脉冲输出	133
6.2.1	占用输入/输出端子	133
6.2.2	高速计数器的工作模式	134
6.2.3	高速计数器的控制字和状态字	137
6.2.4	高速计数器指令及举例	137
6.2.5	高速计数器指令向导的应用	141
6.2.6	高速脉冲输出	143
6.2.7	高速脉冲输出指令向导的应用	150
6.2.8	高速输入、高速输出指令编程实训	158
6.3	PID 控制	161
6.3.1	PID 指令	161
6.3.2	PID 控制功能的应用	163
6.3.3	PID 指令向导的应用	166
6.4	时钟指令	170

6.5	习题	172
第 7 章	PLC 应用系统设计及实例	173
7.1	PLC 应用系统设计概述	173
7.2	PLC 应用系统的设计	173
7.2.1	PLC 控制系统的设计内容及设计步骤	173
7.2.2	PLC 的硬件设计和软件设计及其调试	174
7.2.3	PLC 程序设计常用的方法	175
7.2.4	PLC 程序的设计步骤	186
7.3	PLC 应用举例	187
7.3.1	机械手的模拟控制	187
7.3.2	除尘室 PLC 控制	192
7.3.3	水塔水位的模拟控制实训	193
7.3.4	温度的检测与控制实训	195
7.4	S7-200 系列 PLC 的装配、检测和维修	196
7.4.1	PLC 的安装与配线	196
7.4.2	PLC 的自动检测功能及故障诊断	199
7.4.3	PLC 的维护与检修	199
7.5	习题	199
第 8 章	S7-200 的通信与网络	201
8.1	通信的基本知识	201
8.1.1	基本概念和术语	201
8.1.2	差错控制	202
8.1.3	传输介质	203
8.1.4	串行通信接口标准	204
8.2	工业局域网基础	205
8.2.1	局域网的拓扑结构	205
8.2.2	网络通信协议和体系结构	207
8.2.3	现场总线	207
8.3	S7-200 通信部件	208
8.3.1	通信端口	208
8.3.2	PC/PPI 电缆	208
8.3.3	网络连接器	210
8.3.4	PROFIBUS 网络电缆	210
8.3.5	网络中继器	211
8.3.6	EM277 PROFIBUS-DP 模块	211
8.4	S7-200 PLC 的通信	212
8.4.1	概述	213
8.4.2	利用 PPI 协议进行网络通信	214

8.4.3 利用 MPI 协议进行网络通信	217
8.4.4 利用 PROFIBUS 协议进行网络通信	217
8.4.5 工业以太网	221
8.5 S7-200 PLC 的通信实训	221
8.6 习题	227
附录	228
附录 A 错误代码	228
A.1 致命错误代码和信息	228
A.2 运行程序错误	229
A.3 编译规则错误	229
附录 B S7-200 故障处理指南	230
附录 C 特殊存储器位	231
附录 D S7-200 指令速查	236
参考文献	238

第 1 章 可编程序控制器概述

本章要点

- 可编程序控制器的特点、分类与发展
- 可编程序控制器的定义和基本含义
- 可编程序控制器的基本组成及各部分的作用
- 可编程序控制器的工作原理
- 可编程序控制器的技术指标

1.1 可编程序控制器的发展概况

随着计算机控制技术的不断发展，可编程序控制器的应用已广泛普及，成为自动化技术的重要组成部分。

1969 年，美国数字设备公司（DEC）研制出了世界上第一台可编程序控制器，并应用于通用汽车公司的生产线上。当时叫做可编程序逻辑控制器（Programmable Logic Controller, PLC），目的是用来取代继电器，以执行逻辑判断、计时、计数等顺序控制功能。

随着半导体技术、尤其是微处理器和微型计算机技术的发展，PLC 已广泛地使用 16 位甚至 32 位微处理器作为中央处理器，输入、输出模块和外围电路也都采用了中、大规模甚至超大规模的集成电路，使 PLC 在概念、设计、性能价格比以及应用方面都有了新的突破。这时的 PLC 已不仅仅具有逻辑判断功能，还同时具有数据处理、PID 调节和数据通信功能，称为可编程序控制器（Programmable Controller）更为合适，简称为 PC，但为了与个人计算机（Personal Computer）的简称 PC 相区别，一般将它简称为 PLC（Programmable Logic Controller）。

我国从 1974 年起也开始研制可编程序控制器，于 1977 年开始工业应用。目前它已经大量地被应用在楼宇自动化、家庭自动化、商业、公用事业、测试设备和农业等领域，并涌现出大批应用可编程序控制器的新型设备。掌握可编程序控制器的工作原理，具备设计、调试和维护可编程序控制器控制系统的能力，已经成为现代工业对电气技术人员的基本要求。

1.2 可编程序控制器的定义

国际电工委员会（IEC）曾于 1982 年 11 月颁发了可编程序控制器标准草案第一稿，于 1985 年 1 月又发表了第二稿，1987 年 2 月颁发了第三稿。该草案中对可编程序控制器的定义是：

“可编程序控制器是一种数字运算操作的电子系统，专为在工业环境下应用而设计。它采用了可编程序的存储器，用来在其内部存储和执行逻辑运算、顺序控制、定时、计数和算术运算等操作命令，并通过数字式和模拟式的输入和输出，控制各种类型的机械或生产过程。可编程序控制器及其有关外围设备，都按易于与工业系统联成一个整体、易于扩充其功能的原则设计。”

该定义强调了可编程序控制器是“数字运算操作的电子系统”，是一种计算机。它是“专为在工业环境下应用而设计”的工业计算机，是一种用程序来改变控制功能的工业控制计算机，除了能完成各种各样的控制功能外，还有与其他计算机通信联网的功能。

这种工业计算机采用“面向用户的指令”，因此编程方便。它能完成逻辑运算、顺序控制、定时、计数和算术操作，它还具有“数字量和模拟量输入和输出控制”的能力，并且非常容易与“工业控制系统联成一体”，易于“扩充”。

定义还强调了可编程序控制器能直接应用于工业环境，它须具有很强的抗干扰能力、广泛的适应能力和应用范围。这也是区别于一般微型计算机控制系统的一个重要特征。

应该强调的是，可编程序控制器与以往所讲的顺序控制器在“可编程”方面有质的区别。PLC 中引入了微处理器及半导体存储器等新一代电子器件，并用规定的指令进行编程，能灵活地修改，即用软件方式来实现“可编程序”的目的。

可编程序控制器是应用面最广、功能强大、使用方便的通用工业控制装置。它已经成为当代自动化的主要支柱之一。

1.3 可编程序控制器的基本组成

1.3.1 控制组件

可编程序控制器主要由 CPU、存储器、基本输入/输出 (I/O) 接口电路、外设接口、编程装置和电源等组成。

可编程序控制器的结构多种多样，但其组成的一般原理基本相同，都是以微处理器为核心的结构。可编程序控制器系统的结构如图 1-1 所示。编程装置将用户程序送入可编程序控制器，在可编程序控制器运行状态下，输入单元接收到外部元件发出的输入信号，可编程序控制器执行程序，并根据程序运行后的结果，由输出单元驱动外部设备。

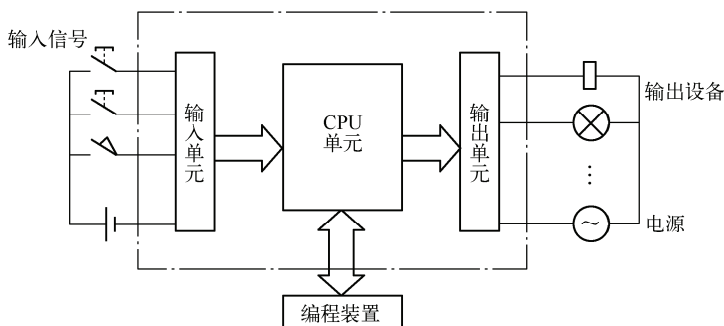


图 1-1 可编程序控制器系统的结构

1. CPU 单元

CPU 是可编程序控制器的控制中枢，相当于人的大脑。CPU 一般由控制电路、运算器和寄存器组成。这些电路通常都被封装在一个集成的芯片上。CPU 通过地址总线、数据总线、控制总线与存储单元、输入/输出接口电路连接。CPU 的功能有：它在系统监控程序的控制下工作，通过扫描方式，将外部输入信号的状态写入输入映像寄存区域，PLC 进入运行状态后，从存储器逐条读取用户指令，按指令规定的任务进行数据的传送、逻辑运算、算术运算等，然后将结果送到输出映像寄存区域。简单地说，CPU 的功能就是读输入、执行程序、写输出。

2. 存储器

可编程序控制器的存储器由只读存储器（ROM）、随机存储器（RAM）和可电擦写的存储器（E²PROM）3 大部分构成，主要用于存放系统程序、用户程序及工作数据。

只读存储器（ROM）用于存放系统程序，可编程序控制器在生产过程中将系统程序固化在 ROM 中，用户是不可改变的。用户程序和中间运算数据存放在随机存储器（RAM）中，RAM 存储器是一种高密度、低功耗、价格便宜的半导体存储器，可用锂电池作为备用电源。它存储的内容是易失的，掉电后内容丢失；当系统掉电时，用户程序可以保存在只读存储器 E²PROM 或由高能电池支持的 RAM 中。E²PROM 兼有 ROM 的非易失性和 RAM 的随机存取优点，用来存放需要长期保存的重要数据。

3. I/O 单元及 I/O 扩展接口

1) I/O 单元（输入/输出接口电路）。PLC 内部输入电路的作用是将 PLC 外部电路（如行程开关、按钮和传感器等）提供的符合 PLC 输入电路要求的电压信号，通过光耦合电路送至 PLC 内部电路中。输入电路通常以光电隔离和阻容滤波的方式提高抗干扰能力，输入响应时间一般在 0.1~15ms。根据输入信号形式的不同，可将 I/O 单元分为模拟量 I/O 单元、数字量 I/O 单元两大类。根据输入单元形式的不同，可将 I/O 单元分为基本 I/O 单元、扩展 I/O 单元两大类。PLC 内部输出电路作用是，将输出映像寄存器的结果通过输出接口电路驱动外部的负载（如接触器线圈、电磁阀和指示灯等）。

2) I/O 扩展接口。可编程序控制器利用 I/O 扩展接口实现 I/O 扩展单元与 PLC 基本单元的连接，当基本 I/O 单元的输入或输出点数不够使用时，可以用 I/O 扩展单元来扩充开关量 I/O 点数和增加模拟量的 I/O 端子。

4. 外设接口

外设接口电路用于连接编程器、文本显示器、触摸屏、变频器等，并能通过外设接口组成 PLC 的控制网络。PLC 通过 PC/PPI 电缆或使用 MPI 卡通过 RS-485 接口与计算机连接，可以实现编程、监控和联网等功能。

5. 电源

电源单元的作用是把外部电源（220V 的交流电源）转换成内部工作电压。外部连接的电源通过 PLC 内部配有的一个专用开关式稳压电源，将交流/直流供电电源转化为 PLC 内部电路需要的工作电源（直流 5V、±12V、24V），并为外部输入元件（如接近开关）提供 24V 直流电源（仅供输入端点使用），而驱动 PLC 负载的电源由用户提供。

1.3.2 输入/输出接口电路

输入/输出接口电路实际上是 PLC 与被控对象间传递输入、输出信号的接口部件。输入/输出接口电路要有良好的电隔离和滤波作用。

1. 输入接口电路

在生产过程中使用的各种开关、按钮和传感器等输入器件直接被接到 PLC 输入接口电路上，为防止因触点抖动或干扰脉冲引起错误的输入信号，输入接口电路必须具有很强的抗干扰能力。

可编程程序控制器输入电路如图 1-2 所示。用输入接口电路提高抗干扰能力的方法主要有：

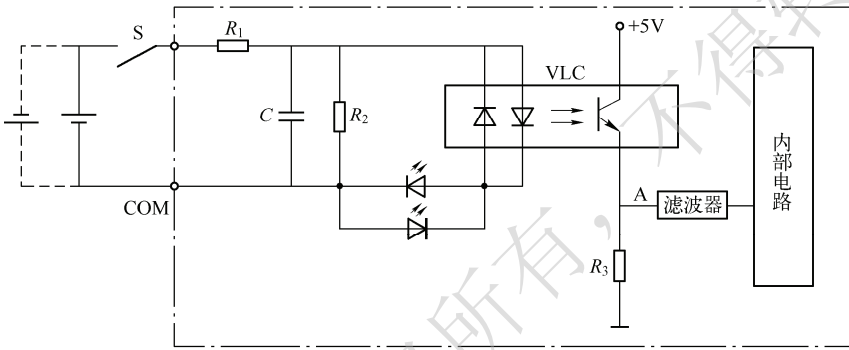


图 1-2 可编程程序控制器输入电路

1) 利用光耦合器提高抗干扰能力。光耦合器的工作原理是，当发光二极管有驱动电流流过时，导通发光，光敏晶体管接收到光线，由截止变为导通，将输入信号送入 PLC 内部。光耦合器中的发光二极管是电流驱动器件，要有足够的能量才能驱动，而干扰信号虽然有的电压值很高，但能量较小，不能使发光二极管导通发光，所以不能进入 PLC 内，实现了电隔离。

2) 利用滤波电路提高抗干扰能力。最常用的滤波电路是电阻电容滤波，见图 1-2 中的 R_1 、 C 。

在图 1-2 中， S 为输入开关，当 S 闭合时，LED 点亮，显示输入开关 S 处于接通状态，光耦合器导通，将高电平经滤波器送到 PLC 内部电路中。当 CPU 在循环的输入阶段锁入该信号时，将该输入点对应的映像寄存器状态置 1；当 S 断开时，则将对应的映像寄存器状态置 0。

根据常用输入电路的电压类型及电路形式不同，可将输入接口电路分为干接点式、直流输入式和交流输入式。输入电路的电源可由外部提供，也可由 PLC 内部提供。

2. 输出接口电路

根据驱动负载元件的不同，可将输出接口电路分为以下 3 种形式。

1) 小型继电器输出形式，其电路如图 1-3 所示。这种输出形式既可驱动交流负载，又可驱动直流负载，驱动负载的能力在 2A 左右。它的优点是适用电压范围比较宽，导通压降小，承受瞬时过电压和过电流的能力强。缺点是动作速度较慢，动作次数（寿命）有一定的

限制。建议在输出量变化不频繁时优先选用。不能用于高速脉冲的输出。

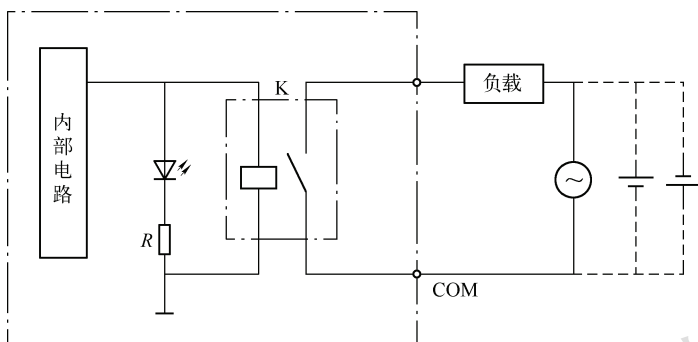


图 1-3 小型继电器输出形式电路

图 1-3 所示电路的工作原理是，当内部电路的状态为 1 时，使继电器 K 的线圈通电，产生电磁吸力，触点闭合，则负载得电，同时点亮 LED，表示该路输出点有输出。当内部电路的状态为 0 时，使继电器 K 的线圈无电流，触点断开，则负载断电，同时 LED 熄灭，表示该路输出点无输出。

2) 大功率晶体管或场效应晶体管输出形式，其电路如图 1-4 所示。这种输出形式只可驱动直流负载。驱动负载的能力：每一个输出点为零点几安培左右。它的优点是可靠性强，执行速度快，寿命长；缺点是过载能力差。适合在直流供电、输出量变化快的场合使用。

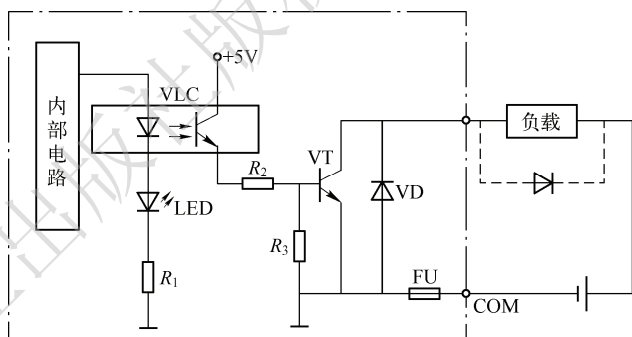


图 1-4 大功率晶体管输出形式电路

图 1-4 所示电路工作原理是，当内部电路的状态为 1 时，光耦合器 VLC 导通，使大功率晶体管 VT 饱和导通，负载得电，同时点亮 LED，表示该路输出点有输出。当内部电路的状态为 0 时，光耦合器 VLC 断开，大功率晶体管 VT 截止，则负载失电，LED 熄灭，表示该路输出点无输出。VD 为保护二极管，可防止负载电压极性接反或高电压、交流电压损坏晶体管。FU 的作用是，防止负载短路时损坏 PLC。由于当负载为电感性负载、VT 关断时，会产生较高的反电势，所以必须给负载并联续流二极管，为其提供放电回路，以避免 VT 承受过电压。

3) 双向晶闸管输出形式, 其电路如图 1-5 所示。这种输出形式适合驱动交流负载。由于双向晶闸管和大功率晶体管同属于半导体材料器件, 所以其优缺点与大功率晶体管或场效应晶体管输出形式的相似, 适合在交流供电、输出量变化快的场合选用。这种输出接口电路驱动负载的能力为 1A 左右。

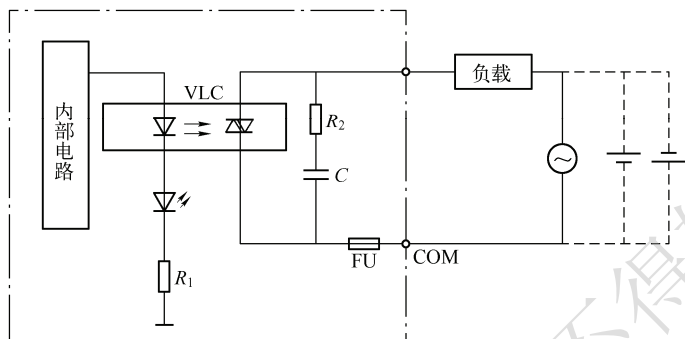


图 1-5 双向晶闸管输出形式电路

图 1-5 所示电路工作原理是, 当内部电路的状态为 1 时, 发光二极管导通发光, 相当于给双向晶闸管施加了触发信号, 无论外接电源极性如何, 双向晶闸管均导通, 负载得电, 同时输出指示灯 LED 点亮, 表示该输出点接通; 当内部电路的状态为 0 时, 双向晶闸管无触发信号, 双向晶闸管关断, 此时 LED 不亮, 负载失电。

3. I/O 电路的常见问题

1) 用晶体管等有源器件作为无触点开关的输出设备与 PLC 输入单元连接时, 由于晶体管自身有漏电流存在, 或者电路不能保证晶体管可靠截止而处于放大状态, 使得即使在截止时, 仍会有一个小漏电流流过, 当该电流值大于 1.3mA 时, 就可能引起 PLC 输入电路发生误动作。可在 PLC 输入端并联一个旁路电阻来分流, 使流入 PLC 的电流小于 1.3mA。

2) 应在输出回路串联熔丝, 避免负载电流过大, 损坏输出元器件或电路板。

3) 由于晶体管、双向晶闸管型输出端子漏电流和残余电压的存在, 当驱动不同类型的负载时, 需要考虑电平匹配和误动等问题。

4) 当感性负载断电时, 会产生很高的反电势, 对输出单元电路产生冲击。对于大电感或频繁关断的感性负载, 应使用外部抑制电路, 一般采用阻容吸收电路或二极管吸收电路。

1.3.3 编程器

编程器是 PLC 的重要外围设备。利用编程器, 可将用户程序送入 PLC 的存储器中, 还可用编程器检查程序和修改程序, 监视 PLC 的工作状态。

目前, 可编程序控制器厂商或经销商向用户提供编程软件, 在个人计算机上添加适当的硬件接口和软件包, 即可用个人计算机对 PLC 编程。利用微型计算机作为编程器, 可以直接编制并显示梯形图, 程序可以存盘、打印、调试, 对于查找故障非常有利。

1.4 可编程序控制器的工作原理及主要技术指标

1.4.1 可编程序控制器的工作原理

结合 PLC 的组成和结构分析 PLC 的工作原理更容易理解。PLC 采用周期循环扫描的工作方式，CPU 连续执行用户程序和任务的循环序列称为扫描。CPU 对用户程序的执行过程是通过 CPU 的循环扫描，并用周期性地集中采样、集中输出的方式来完成的一个扫描周期主要可分为：

1) 读输入阶段。每次扫描周期的开始，先读取输入点的当前值，然后写到输入映像寄存器区域中。在用户程序执行的过程中，CPU 访问输入映像寄存器区域，而并非读取输入端口的状态，输入信号的变化并不会影响到输入映像寄存器的状态，通常要求输入信号有足够的脉冲宽度，才能被响应。

2) 执行程序阶段。用户程序执行阶段，PLC 按照梯形图的顺序，自左而右、自上而下的逐行扫描，在这一阶段 CPU 从用户程序的第一条指令开始执行，直到最后一条指令结束，程序运行结果放入输出映像寄存器区域。在此阶段，允许对数字量 I/O 指令和不设置数字滤波的模拟量 I/O 指令进行处理，在扫描周期的各个部分，均可对中断事件进行响应。

3) 处理通信请求阶段。扫描周期的信息处理阶段，CPU 处理从通信端口接收到的信息。

4) 执行 CPU 自诊断测试阶段。在此阶段，CPU 检查其硬件、用户程序存储器 and 所有 I/O 模块的状态。

5) 写输出阶段。在每个扫描周期的结尾，CPU 把存在输出映像寄存器中的数据输出给数字量输出端点（写入输出锁存器中），更新输出状态。然后 PLC 进入下一个循环周期，重新执行输入采样阶段，周而复始。

如果程序中使用了中断，中断事件出现，CPU 立即执行中断程序，中断程序可以在扫描周期的任意点被执行。

如果程序中使用了立即 I/O 指令，可以直接存取 I/O 点。用立即 I/O 指令读输入点值时，相应的输入映像寄存器的值未被修改，用立即 I/O 指令写输出点值时，相应的输出映像寄存器的值被修改。

1.4.2 可编程序控制器的主要技术指标

可编程序控制器的种类很多，用户可以根据控制系统的具体要求选择不同技术性能指标的 PLC。可编程序控制器的技术性能指标主要有以下几点：

1. 输入/输出点数

可编程序控制器的 I/O 点数指外部输入、输出端子数量的总和。它是 PLC 的一个重要参数。

2. 存储容量

PLC 的存储器由系统程序存储器、用户程序存储器和数据存储器 3 部分组成。PLC 的存储容量通常指用户程序存储器和数据存储器容量之和，表征系统提供给用户的可用资源，是

系统性能的一项重要技术指标。

3. 扫描速度

可编程序控制器采用循环扫描方式工作，完成一次扫描所需的时间叫做扫描周期。影响扫描速度的主要因素有用户程序的长度和 PLC 产品的类型。PLC 中 CPU 的类型、机器字长等直接影响 PLC 的运算精度和运行速度。

4. 指令系统

指令系统是指 PLC 所有指令的总和。可编程序控制器的编程指令越多，软件功能就越强，但其应用也相对较复杂。用户应根据实际控制要求，选择合适指令功能的可编程序控制器。

5. 通信功能

通信有 PLC 之间的通信和 PLC 与其他设备之间的通信。通信主要涉及通信模块、通信接口、通信协议和通信指令等内容。PLC 的组网和通信能力也已成为 PLC 产品水平的重要衡量指标之一。

厂家的产品手册上还提供 PLC 的负载能力、外形尺寸、重量、保护等级、适用的安装和使用环境（如温度、湿度等性能指标）等参数，供用户参考。

1.5 可编程序控制器的分类、特点、应用及发展

1.5.1 可编程序控制器的分类

1. 按 I/O 点数和功能分类

可编程序控制器用于对外部设备的控制，外部信号的输入、PLC 运算结果的输出都要通过 PLC 的输入、输出端子来进行连接，输入、输出端子的数目之和被称做 PLC 的输入、输出点数，简称为 I/O 点数。

由 I/O 点数的多少，可将 PLC 分成小型、中型和大型。

小型 PLC 的 I/O 点数小于 256 点，以开关量控制为主，它具有体积小、价格低的优点。可用于开关量的控制、定时/计数的控制、顺序控制及少量模拟量的控制，代替继电器-接触器控制，用于单机或小规模生产过程中。

中型 PLC 的 I/O 点数在 256~1024 点，功能比较丰富，兼有开关量和模拟量的控制能力，适用于较复杂系统的逻辑控制和闭环过程的控制。

大型 PLC 的 I/O 点数在 1024 点以上，用于大规模过程控制、集散式控制和工厂自动化网络。

2. 按结构形式分类

PLC 可分为整体式结构和模块式结构两大类。

整体式 PLC 是将 CPU、存储器、I/O 部件等组成部分集中于一体，安装在印制电路板上，并连同电源一起装在一个机壳内，形成一个整体，通常称为主机或基本单元。整体式结构的 PLC 具有结构紧凑、体积小、重量轻和价格低的优点。一般小型或超小型 PLC 多采用这种结构。

模块式 PLC 是把各个组成部分制成独立的模块，如 CPU 模块、输入模块、输出模块和电

源模块等。将各模块制成插件式，组装在一个具有标准尺寸并带有若干插槽的机架内。模块式结构的 PLC 配置灵活，装配和维修方便，易于扩展。一般大中型的 PLC 都采用这种结构。

1.5.2 可编程序控制器的特点

1) 编程简单，使用方便。梯形图是用得最多的可编程序控制器的编程语言，其符号与继电器电路原理图相似。有继电器电路基础的电气技术人员只要很短的时间就可以熟悉梯形图语言，并用它来编制用户程序。梯形图语言形象直观，易学易懂。

2) 控制灵活，程序可变，具有很好的柔性。可编程序控制器产品采用模块化形式，配备有品种齐全的各种硬件装置供用户选用，用户能灵活方便地进行系统配置，将其组成不同功能、不同规模的系统。可编程序控制器用软件功能取代了继电器控制系统中大量的中间继电器、时间继电器、计数器等器件，硬件配置确定后，可以修改用户程序而不用改变硬件，方便快速地适应工艺条件的变化，具有很好的柔性。

3) 功能强，扩充方便，性能价格比高。可编程序控制器内有成百上千个可供用户使用的编程元件，有很强的逻辑判断、数据处理、PID 调节和数据通信功能，可以实现非常复杂的控制功能。如果元件不够，只要加上需要的扩展单元即可，扩充非常方便。与相同功能的继电器控制系统相比，具有很高的性能价格比。

4) 控制系统设计及施工的工作量少，维修方便。可编程序控制器的配线与其他控制系统的配线比较少得多，故可以省下大量的配线，减少大量的安装接线时间，开关柜体积缩小，节省大量的费用。可编程序控制器有较强的带负载能力，可以直接驱动一般的电磁阀和交流接触器。一般可用接线端子连接外部线路。可编程序控制器的故障率很低，且有完善的自诊断和显示功能，便于迅速地排除故障。

5) 可靠性高，抗干扰能力强。可编程序控制器是为现场工作设计的，采取了一系列硬件和软件抗干扰措施。硬件措施如屏蔽、滤波、电源调整与保护、隔离、后备电池等。例如，在西门子公司 S7-200 系列 PLC 内部 E²PROM 中，储存了用户源程序和预设值，在一个较长时间段（190h），所有中间数据可以通过一个超级电容器保持，如果选配电池模块，就可以确保停电后中间数据保存 200 天。软件措施如故障检测、信息保护和恢复、警戒时钟，加强了对程序的检测和校验，从而提高了系统抗干扰能力，平均无故障时间达到数万小时以上，可以直接用于有强烈干扰的工业生产现场。可编程序控制器已被广大用户公认为是最可靠的工业控制设备之一。

6) 体积小、重量轻、能耗低，是“机电一体化”特有的产品。

1.5.3 可编程序控制器的应用

目前，可编程序控制器已经广泛地应用在各个工业部门。随着其性能价格比的不断提高，应用范围还在不断扩大，主要有以下几个方面：

1. 逻辑控制

可编程序控制器具有“与”、“或”、“非”等逻辑运算的能力，可以实现逻辑运算，用触点和电路的串、并联，代替继电器进行组合逻辑控制、定时控制与顺序逻辑控制。数字量逻辑控制可以用于单台设备，也可以用于自动生产线，其应用领域最为普及，包括微电子、家电行业。

2. 运动控制

可编程序控制器使用专用的运动控制模块，灵活运用指令，使运动控制与顺序控制功能有机地结合在一起。随着变频器、电动机起动器的普遍使用，可编程序控制器可以与变频器结合，运动控制功能更为强大，并广泛地用于各种机械（如金属切削机床、装配机械、机器人和电梯等）中。

3. 过程控制

可编程序控制器可以接收温度、压力、流量等连续变化的模拟量，通过模拟量 I/O 模块，实现模拟量（Analog）和数字量（Digital）之间的模/数（A/D）转换和数/模（D/A）转换，并对被控模拟量实行闭环 PID（比例-积分-微分）控制。现代的大中型可编程序控制器一般都有 PID 闭环控制功能，此功能已经广泛地应用于加热炉、锅炉等设备以及轻工、化工、机械、冶金、电力和建材等行业。

4. 数据处理

可编程序控制器具有数学运算、数据传送、转换、排序和查表、位操作等功能，可以完成数据的采集、分析和处理。这些数据可以是运算的中间参考值，也可以通过通信功能传送到别的智能装置，或者将它们保存、打印。数据处理一般用于大型控制系统，如无人柔性制造系统，也可以用于过程控制系统，如造纸、冶金、食品工业中的一些大型控制系统。

5. 构建网络控制

可编程序控制器的通信包括主机与远程 I/O 之间的通信、多台可编程序控制器之间的通信、可编程序控制器和其他智能控制设备（如计算机、变频器）之间的通信。可编程序控制器与其他智能控制设备一起，可以组成“集中管理、分散控制”的分布式控制系统。

当然，并非所有的可编程序控制器都具有上述功能，用户应根据系统的需要选择可编程序控制器，这样既能完成控制任务，又可节省资金。

1.5.4 可编程序控制器的发展

1. 向高集成、高性能、高速度和大容量发展

微处理器技术、存储技术的发展十分迅猛，为可编程序控制器的发展提供了良好的环境。大型可编程序控制器大多采用多 CPU 结构，不断地向高性能、高速度和大容量方向发展。

在模拟量控制方面，除了专门用于模拟量闭环控制的 PID 指令和智能 PID 模块外，一些可编程序控制器还具有模糊控制、自适应、参数自整定功能，使调试时间减少，控制精度提高。

2. 向普及化方向发展

由于微型可编程序控制器的价格便宜，体积小、重量轻、能耗低，适合单机自动化，而且其外部接线简单，容易实现或组成控制系统，在很多控制领域中将得到广泛应用。

3. 向模块化和智能化发展

可编程序控制器采用模块化的结构，方便了用户的使用和维护。智能 I/O 模块主要有模拟量 I/O、高速计数输入、中断输入、机械运动控制、热电偶输入、热电阻输入、条形码阅读器、多路 BCD 码输入/输出、模糊控制器、PID 控制、通信等模块。智能 I/O 模块本身就是一个小的微型计算机系统，有很强的信息处理能力和控制功能，有的模块甚至可以自成系

统，单独工作。它们可以完成可编程序控制器的主 CPU 难以兼顾的功能，简化了某些控制领域的系统设计和编程，提高了可编程序控制器的适应性和可靠性。

4. 向软件化发展

编程软件可以控制可编程序控制器系统的硬件组态，即设置硬件的结构和参数，例如设置各框架各个插槽上模块的型号、模块的参数、各串行通信接口的参数等。在屏幕上可以直接生成和编辑梯形图、指令表、功能块图和顺序功能图程序，并可以实现不同编程语言的相互转换。可编程序控制器的编程软件有调试和监控功能，可以在梯形图中显示触点的通断和线圈的通电情况，使查找复杂电路的故障非常方便。对于历史数据，可以存盘或打印，通过网络或 Modem 卡还可以实现远程编程和传送。

个人计算机（PC）的价格便宜，有很强的数学运算、数据处理、通信和人机交互的功能。目前已有多家厂商推出了在 PC 上运行的实现可编程序控制器功能的软件包，如亚控公司的 KingPLC。“软 PLC”在很多方面比传统的“硬 PLC”有优势，有的场合“软 PLC”可能是理想的选择。

5. 向通信网络化发展

伴随科技发展，很多工业控制产品都加设了智能控制和通信功能，如变频器、软起动器等，可以和现代的可编程序控制器通信联网，实现更强大的控制功能。通过双绞线、同轴电缆或光纤联网，信息可以传送到几十千米远的地方，通过 Modem 和互联网可以与世界上其他地方的计算机装置进行通信。

相当多的大中型控制系统都采用上位计算机加可编程序控制器的方案，通过串行通信接口或网络通信模块，实现上位计算机与可编程序控制器交换数据信息。组态软件引发的上位计算机编程革命，很容易实现两者的通信，降低了系统集成的难度，节约了大量的设计时间，提高了系统的可靠性。国际上比较著名的组态软件有 Intouch、Fix 等，国内也涌现出了组态王、力控等一批组态软件。有的可编程序控制器厂商也推出了自己的组态软件，如西门子的 WinCC。

1.6 习题

1. 简述可编程序控制器的定义。
2. 可编程序控制器有哪些基本组成？
3. 输入接口电路有哪几种形式？输出接口电路有哪几种形式？各有何特点？
4. PLC 的工作原理是什么？工作过程分为哪几个阶段？
5. 可编程序控制器有哪些主要特点？
6. 可编程序控制器可以在哪些领域应用？

第 2 章 西门子 S7-200 系列可编程序控制器

本章要点

- 西门子 S7-200 CPU224 可编程序控制器的结构、性能指标
- 西门子 S7-200 CPU224 可编程序控制器的工作方式
- 扩展模块介绍
- S7-200 系列可编程序控制器的编址、寻址方式
- 可编程序控制器元器件功能及地址分配

2.1 S7-200 系列 PLC 概述

西门子 S7 系列可编程序控制器分为 S7-400、S7-300、S7-200 三个系列，分别为 S7 系列的大、中、小型可编程序控制器系统。S7-200 系列可编程序控制器有 CPU21X 系列和 CPU22X 系列，其中 CPU22X 型可编程序控制器提供了 4 个不同的基本型号，分别为 CPU221、CPU222、CPU224 和 CPU226。

在小型 PLC 中，CPU221 价格低廉，能满足多种集成功能的需要。CPU222 是 S7-200 家族中低成本的单元，通过可连接的扩展模块即可处理模拟量。CPU224 具有更多的输入输出点及更大的存储器。CPU226 和 226XM 是功能最强的单元，可完全满足一些中小型复杂控制系统的要求。4 种型号的 PLC 具有下列特点。

- 1) 集成的 24V 电源。可直接连接到传感器和变送器执行器，CPU221 和 CPU222 具有 180mA 输出。CPU224 输出 280mA。CPU226、CPU226XM 输出 400mA，可用做负载电源。
- 2) 高速脉冲输出。具有两路高速脉冲输出端，输出脉冲频率可达 20kHz，用于控制步进电动机或伺服电动机，实现定位任务。
- 3) 通信口。CPU221、CPU222 和 CPU224 具有一个 RS-485 通信口。CPU226、CPU226XM 具有两个 RS-485 通信口。支持 PPI、MPI 通信协议，有自由口通信能力。
- 4) 模拟电位器。CPU221/222 有一个模拟电位器，CPU224/226/226XM 有两个模拟电位器。模拟电位器用来改变特殊寄存器（SMB28，SMB29）中的数值，以改变程序运行时的参数，如定时器、计数器的预置值，过程量的控制参数。
- 5) 中断输入允许以极快的速度对过程信号的上升沿作出响应。
- 6) E²PROM 存储器模块（选件）。可作为修改与复制程序的快速工具，无需编程器，并可进行辅助软件归档工作。
- 7) 电池模块。用户数据（如标志位状态、数据块、定时器和计数器）可通过内部的超级电容存储大约 5 天。选用电池模块能延长存储时间到 200 天（10 年寿命）。电池模块插在存储器模块的卡槽中。

8) 不同的设备类型。CPU221~226 各有两种类型 CPU，具有不同的电源电压和控制电压。

9) 数字量输入/输出点。CPU221 具有 6 个输入点和 4 个输出点；CPU222 具有 8 个输入点和 6 个输出点；CPU224 具有 14 个输入点和 10 个输出点；CPU226/226XM 具有 24 个输入点和 16 个输出点。CPU22X 主机的输入点为 24V 直流双向光耦合输入电路，输出有继电器和直流（MOS 型）两种类型。

10) 高速计数器。CPU221/222 有 4 个 30kHz 高速计数器，CPU224/226/226XM 有 6 个 30kHz 的高速计数器，用于捕捉比 CPU 扫描频率更快的脉冲信号。

CPU22X 模块主要技术指标见表 2-1。

表 2-1 CPU22X 模块主要技术指标

型 号	CPU221	CPU222	CPU224	CPU226	CPU226MX
用户数据存储类型	E ² PROM	E ² PROM	E ² PROM	E ² PROM	E ² PROM
程序空间（永久保存）/字	2048	2048	4096	4096	8192
用户数据存储/字	1024	1024	2560	2560	5120
数据后备（超级电容）典型值/H	50	50	190	190	190
主机 I/O 点数	6/4	8/6	14/10	24/16	24/16
可扩展模块	无	2	7	7	7
24V 传感器电源最大电流/电流限制/mA	180/600	180/600	280/600	400/约 1500	400/约 1500
最大模拟量输入/输出	无	16/16	28/7 或 14	32/32	32/32
AC 240V 电源 CPU 输入电流/最大负载电流/mA	25/180	25/180	35/220	40/160	40/160
DC 24V 电源 CPU 输入电流/最大负载/mA	70/600	70/600	120/900	150/1050	150/1050
为扩展模块提供的 DC5V 电源的输出电流/mA	-	最大 340	最大 660	最大 1000	最大 1000
内置高速计数器	4 (30kHz)	4 (30kHz)	6 (30kHz)	6 (30kHz)	6 (30kHz)
高速脉冲输出	2 (20kHz)	2 (20kHz)	2 (20kHz)	2 (20kHz)	2 (20kHz)
模拟量调节电位器/个	1	1	2	2	2
实时时钟	有（时钟卡）	有（时钟卡）	有（内置）	有（内置）	有（内置）
RS-485 通信口	1	1	1	1	1
各组输入点数	4,2	4,4	8,6	13,11	13,11
各组输出点数	4 (DC 电源) 1,3 (AC 电源)	6 (DC 电源) 3,3 (AC 电源)	5,5 (DC 电源) 4,3,3 (AC 电源)	8,8 (DC 电源) 4,5,7 (AC 电源)	8,8 (DC 电源) 4,5,7 (AC 电源)

2.2 S7-200 系列 CPU224 型 PLC 的结构

2.2.1 CPU224 型 PLC 的外形、端子及接线

1. CPU224 型 PLC 的外形

S7-200 PLC 外形如图 2-1 所示。其输入、输出、CPU、电源模块均被装设在一个基本单元的机壳内，是典型的整体式结构。当系统需要扩展时，选用需要的扩展模块与基本单元连接。

2) 基本输出端子。PLC 晶体管输出端子及外部接线如图 2-3 所示。Q0.0~Q0.4 共用 1M 和 1L 公共端, Q0.5~Q1.1 共用 2M 和 2L 公共端, 在公共端上需要用户连接适当的电源, 为 PLC 的负载服务。

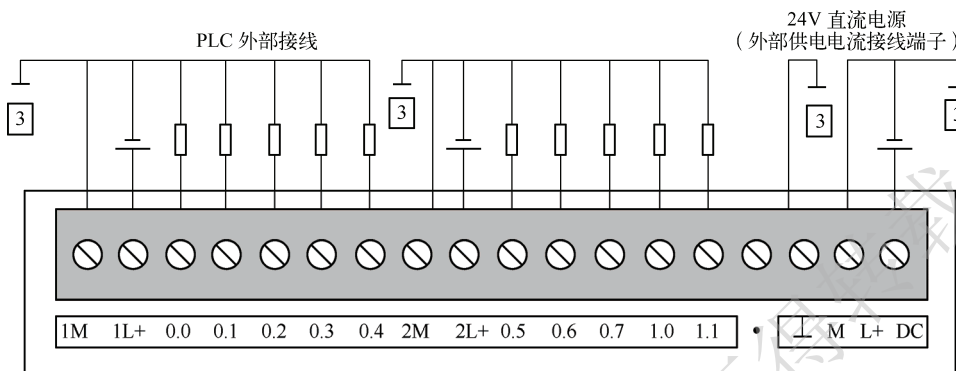


图 2-3 PLC 晶体管输出端子及外部接线图

CPU224 的输出电路有晶体管输出电路和继电器输出两种供用户选用。在晶体管输出电路(型号为 6ES7 214-1AD21-0XB0)中, PLC 由 24V 直流供电, 负载采用了 MOSFET 功率驱动器件, 所以只能用直流为负载供电。输出端将数字量输出分为两组, 每组有一个公共端, 共有 1L、2L 两个公共端, 可接入不同电压等级的负载电源。在继电器输出电路中(型号为 6ES7 212-1BB21-0XB0), PLC 由 220V 交流电源供电, 负载采用了继电器驱动, 所以既可以选用直流为负载供电, 也可以采用交流为负载供电。在继电器输出电路中, 数字量输出被分为 3 组, 每组的公共端为本组的电源供给端, Q0.0~Q0.3 共用 1L, Q0.4~Q0.6 共用 2L, Q0.7~Q1.1 共用 3L, 各组之间可接入不同电压等级、不同电压性质的负载电源。继电器输出形成 PLC 输出端子及外部接线如图 2-4 所示。

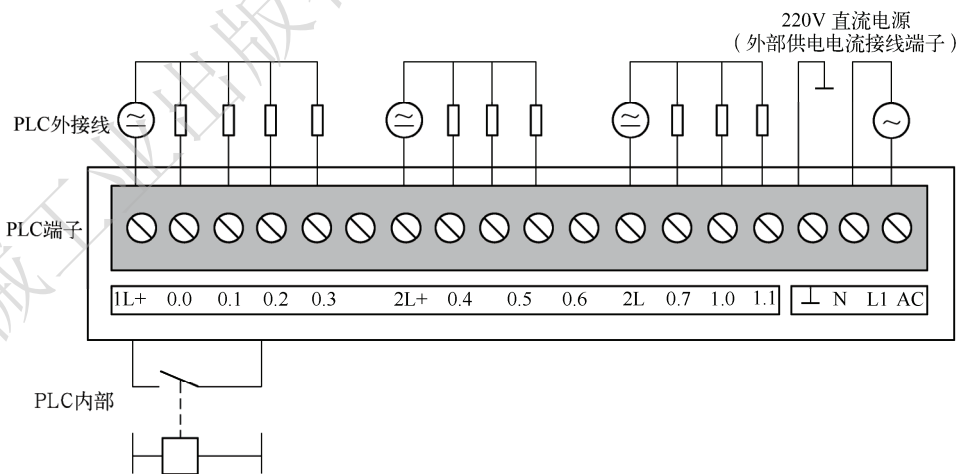


图 2-4 继电器输出形式 PLC 输出端子及外部接线图

3) 高速反应性。CPU224 PLC 有 6 个高速计数脉冲输入端 (I0.0~I0.5), 最快的响应速度为 30kHz, 用于捕捉比 CPU 扫描周期更短的脉冲信号。

CPU224 PLC 有两个高速脉冲输出端 (Q0.0, Q0.1), 输出频率可达 20kHz, 用于 PTO (高速脉冲束) 和 PWM (宽度可变脉冲输出) 高速脉冲输出。

4) 模拟电位器。模拟电位器用来改变特殊寄存器 (SM28, SM29) 中的数值, 以改变程序运行时的参数, 如定时器、计数器的预置值和过程量的控制参数。

5) 存储卡。该卡位可以选择安装扩展卡。扩展卡有 EEPROM 存储卡、电池和时钟卡等模块。存储卡用于用户程序的复制。在 PLC 通电后插此卡, 通过操作可将 PLC 中的程序装载到存储卡中。若卡已被插在基本单元上, 则 PLC 通电后不需任何操作, 卡上的用户程序数据会自动复制在 PLC 中。利用这一功能, 可对无数台实现同样控制功能的 CPU22X 系列进行程序写入。

注意: 每次通电就写入一次, 所以在 PLC 运行时, 不要插入此卡。

电池模块用于长时间保存数据, 使用 CPU224 内部存储电容数据的存储时间达 190h, 而使用电池模块数据的存储时间可达 200 天。

3. S7-224XP CN 介绍

图 2-5 所示为 S7-224CN XP 外形图。在图中有两个通信端口, 有电源端子、输入端子、输出端子、模拟量 AI/AO 端子、24V 直流电源输出端子、拨码开关、用于连接扩展电缆的接口等。

S7-224 XP CN 集成数字量输入 14 点, 输出 10 点, 最多可连续扩展模块 7 个, 高速计数器可计 200kHz 的高速脉冲, 可输出 100kHz 的高速脉冲, 有两个串行通信端口, 并且在本机体上自带有两路模拟量输入和一路模拟量输出 (2AI/1AO), 不用配置模拟量模块即可进行单回路模拟量控制, 具有良好的性价比。编程软件需使用 Step7.Micro/Win4.0SP3 及以上版本。S7-224 XP CN 模拟量通道接线如图 2-6 所示, A+、B+ 为模拟量输入单端, M 为公共端; 输入电压范围为 $\pm 10V$; 分辨率为 11 位, 加 1 符号位; 数据字格式对应的满量程范围为 $-32\ 000 \sim +32\ 000$, 对应的模拟量输入映像寄存器分别为 AIW0、AIW2。图 2-6 中有一路单极性模拟量输出, 可以选择是电流输出或电压输出, I 为电流负载输出, V 为电压负载输出, 输出电流的范围为 $0 \sim 20mA$, 输出电压的范围为 $0 \sim 10V$, 分辨率为 12 位, 数据格式对应的量程范围为 $0 \sim 32\ 767$, 对应的模拟量输出映像寄存器为 AQW0。



图 2-5 S7-224CN XP 外形图

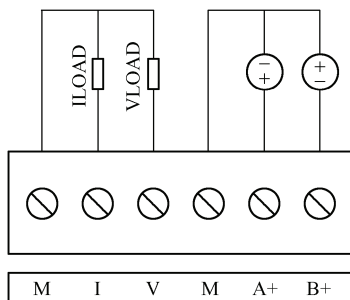


图 2-6 S7-224 XP CN 模拟量通道接线图

2.2.2 CPU224 型 PLC 的结构及性能指标

CPU224 型可编程程序控制器主要由 CPU、存储器、基本 I/O 接口电路、外设接口、编程

器通信网络和电源等组成，其结构框图如图 2-7 所示。

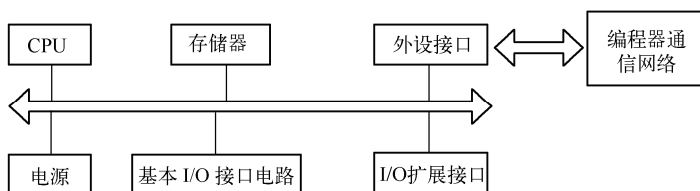


图 2-7 CPU224 型可编程序控制器结构框图

CPU224 型可编程序控制器有两种，一种是 CPU224 AC/DC/继电器，交流输入电源，提供 24V 直流给外部元器件（如传感器等），继电器方式输出，14 点输入，10 点输出；另一种是 CPU 224 DC/DC/DC，直流 24V 输入电源，提供 24V 直流给外部元器件（如传感器等），半导体元器件直流方式输出，14 点输入，10 点输出。用户可根据需要选用。它们的主要技术指标见表 2-2~表 2-4。

表 2-2 电源的主要技术指标

特 性	24V 电源	AC 电源
电压允许范围	20.4~28.8V	85~264V, 47~63Hz
冲击电流	10A, 28.8V	20A, 254V
内部熔断器（用户不能更换）	3A, 250V 慢速熔断	2A, 250V 慢速熔断

表 2-3 数字量输入的主要技术指标

项 目	指 标
输入类型	漏型/源型
输入电压额定值	DC 24V
“1” 信号	15~35V, 最大 4mA
“0” 信号	0~5V
光电隔离	AC 500V, 1min
非屏蔽电缆长度	300m
屏蔽电缆长度	500m

表 2-4 数字量输出的主要技术指标

特 性	DC 24V 输出	继电器型输出
电压允许范围	20.4~28.8V	—
逻辑 1 信号最大电流	0.75A（电阻负载）	2A（电阻负载）
逻辑 0 信号最大电流	10 μ A	0
灯负载	5W	DC 30W / AC 200W
非屏蔽电缆长度	150m	150m
屏蔽电缆长度	500m	500m
触点机械寿命	—	10 000 000 次
额定负载时触点寿命	—	100 000 次

2.2.3 PLC 的 CPU 工作方式

1. CPU 的工作方式

CPU 前面板上用两个发光二极管显示当前工作方式，绿色指示灯亮，表示为运行状态，红色指示灯亮，表示为停止状态，在标有 SF 的指示灯亮时表示系统出现故障，PLC 停止工作。

1) STOP (停止)。CPU 在停止工作方式时，不执行程序，此时可以通过编程装置向 PLC 装载程序或进行系统设置。在程序编辑、上下载等处理过程中，必须把 CPU 置于 STOP 方式。

2) RUN (运行)。CPU 在 RUN 工作方式下，PLC 按照自己的工作方式运行用户程序。

2. 改变工作方式的方法

(1) 用工作方式开关改变工作方式。

工作方式开关有 3 个档位：STOP、TERM (Terminal)、RUN。

1) 把方式开关切到 STOP 位，可以停止程序的执行。

2) 把方式开关切到 RUN 位，可以启动程序的执行。

3) 把方式开关切到 TERM (暂态) 或 RUN 位，允许 STEP 7-Micro/Win 32 软件设置 CPU 工作状态。

如果工作方式开关设为 STOP 或 TERM，电源上电时，CPU 自动进入 STOP 工作状态。设置为 RUN 时，电源上电，CPU 自动进入 RUN 工作状态。

(2) 用编程软件改变工作方式

把方式开关切换到 TERM (暂态)，可以使用 STEP7-Micro/Win 32 编程软件设置工作方式。

(3) 在程序中用指令改变工作方式

在程序中插入一个 STOP 指令，CPU 可由 RUN 方式进入 STOP 工作方式。

2.3 扩展功能模块

2.3.1 扩展单元及电源模块

1. 扩展单元

扩展单元没有 CPU，作为基本单元输入/输出点数的扩充，只能与基本单元连接使用，而不能单独使用。S7-200 的扩展单元包括数字量扩展单元、模拟量扩展单元、热电偶/热电阻扩展模块和 PROFIBUS-DP 通信模块。

用户选用具有不同功能的扩展模块，可以满足不同的控制需要，节约投资费用。连接时将 CPU 模块放在最左侧，扩展模块用扁平电缆与左侧的模块相连，如图 2-8 所示。CPU222 最多连接两个扩展模块，CPU224/CPU226 最多连接 7 个扩展模块。

2. 电源模块

外部提供给 PLC 的电源有 DC 24V、AC 220V 两种，根据型号不同有所变化。S7-200 的 CPU 单元有一个内部电源模块，S7-200 小型 PLC 的电源模块与 CPU 封装在一起，通过连接总线为 CPU 模块、扩展模块提供 5V 的直流电源，如果容量许可，还可提供给外部

DC 24V 的电源，供本机输入点和扩展模块继电器线圈使用。应根据下面的原则来确定 I/O 电源的配置。

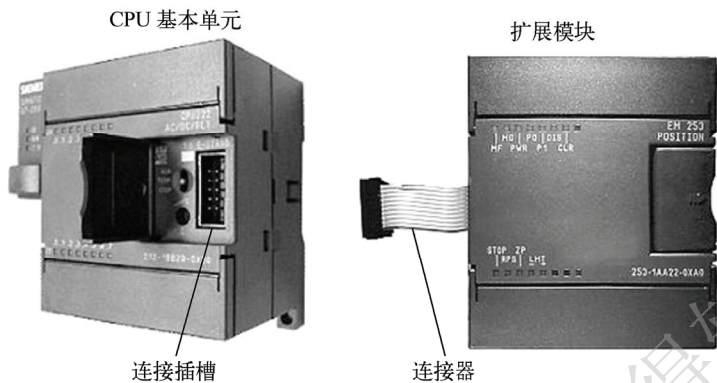


图 2-8 CPU 基本单元和扩展模块的连接

1) 有扩展模块连接时，如果扩展模块对 DC 5V 电源的需求超过 CPU 的 5V 电源模块的容量，则必须减少扩展模块的数量。

2) 当 DC +24V 电源的容量不满足要求时，可以增加一个外部 DC 24V 电源给扩展模块供电。此时，外部电源不能与 S7-200 的传感器电源并联使用，但应将两个电源的公共端 (M) 连接在一起。

I/O 电源的具体参数可以参看表 2-1~表 2-4。

2.3.2 常用扩展模块

1. 数字量扩展模块

当需要本机集成的数字量输入/输出点外更多的数字量的输入/输出时，可选用数字量扩展模块。用户选择具有不同 IO 点数的数字量扩展模块，可以满足应用的实际要求，同时节约不必要的投资费用。可选择 8、16 和 32 点输入/输出模块。

S7-200 PLC 系列目前总共可以提供 3 大类共 9 种数字量输入/输出扩展模块，其数据如表 2-5 所示。

表 2-5 数字量输入/输出扩展模块的数据表

类 型	型 号	各组输入点数	各组输出点数
输入扩展模块 EM221	EM221 24V DC 输入	4, 4	—
	EM221 230V AC 输入	8 点相互独立	—
输出扩展模块 EM222	EM222 24V DC 输出	—	4, 4
	EM222 继电器输出	—	4, 4
	EM222 230V AC 双向晶闸管输出	—	8 点相互独立
输入/输出 扩展模块 EM223	EM223 24V DC 输入/继电器输出	4	4
	EM223 24V DC 输入/24V DC 输出	4, 4	4, 4
	EM223 24V DC 输入/24V DC 输出	8, 8	4, 4, 8
	EM223 24V DC 输入/继电器输出	8, 8	4, 4, 4, 4

2. 模拟量扩展模块

模拟量扩展模块提供了模拟量输入/输出的功能。在工业控制中，被控对象常常是模拟量，如温度、压力、流量等。PLC 内部执行的是数字量，模拟量扩展模块可以将 PLC 外部的模拟量转换为数字量送入 PLC 内，经 PLC 处理后，再由模拟量扩展模块将 PLC 输出的数字量转换为模拟量送给控制对象。模拟量扩展模块优点如下。

- 1) 最佳适应性。可适用于复杂的控制场合，直接与传感器和执行器相连。例如，可将 EM235 模块直接与 PT100 热电阻相连。
- 2) 灵活性。当实际应用变化时，可以相应地将 PLC 进行扩展，并可非常容易的调整用户程序。

模拟量扩展模块的数据如表 2-6 所示。

表 2-6 模拟量扩展模块的数据表

模 块	EM231	EM232	EM235
点数	4 路模拟量输入	2 路模拟量输出	4 路输入，1 路输出

EM235 模块的面板及接线如图 2-9 所示。EM235 具有 4 路模拟量输入和 1 路模拟量输出，它的输入信号可以是不同量程的电压或电流。其电压、电流的量程由配置设定开关 SW1~SW6 设定。EM235 有 1 路模拟量输出，其输出电压、或电流。EM235 有 4 路模拟量输入，为 RA、A+、A-；RB、B+、B-；RC、C+、C-；RD、D+、D- 共 4 路模拟量输入通道，每 3 个点为一组。接线如图 2-9 上部所示，当输入信号为电压信号时，只用两个端子（如图 2-9 中的 A+、A-），电流信号需用 3 个端子（见图 2-9 中的 RC、C+、C-），其中 RC 与 C+ 端子

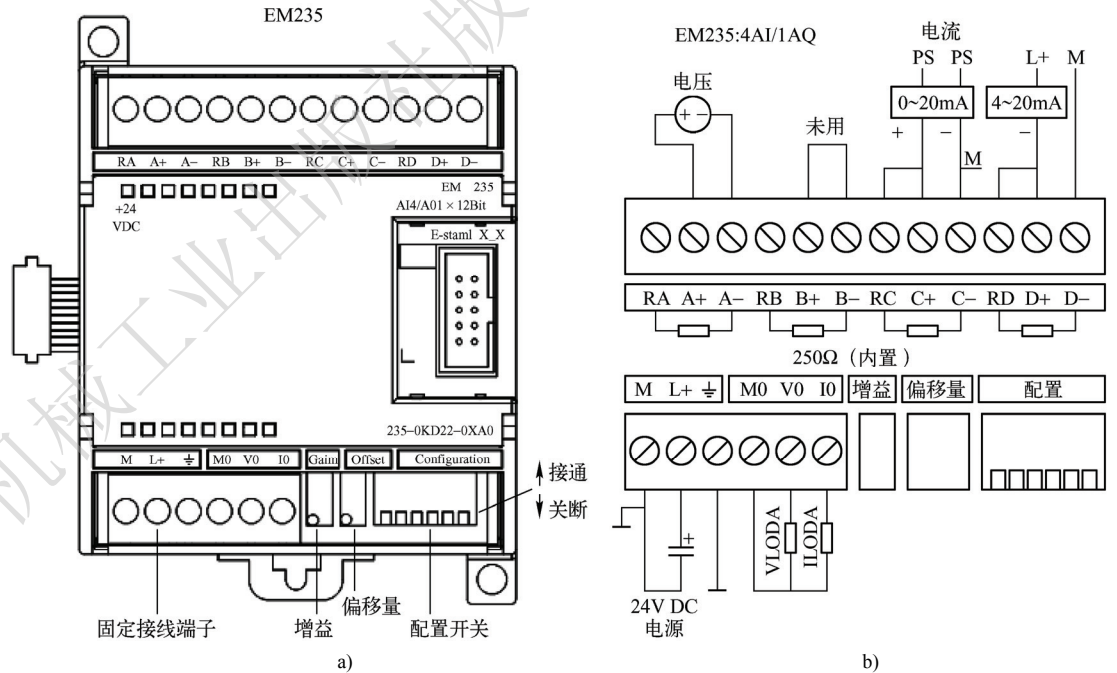


图 2-9 EM235 模块的面板及接线图

a) 面板 b) 接线图

短接。对于未使用的输入通道应短接（见图 2-9 中的 B+、B-）。EM235 模拟量输出端子为 M0、V0、I0，电压输出时，“V0”为电压正端、“M0”为电压负端；电流输出时，“I0”为电流的流入端，“M0”为电流的流出端。在模块下部左端 M、L+两端，应接入 DC24V 电源，M 为 DC24V 电源负极端，L+为电源正极端。

EM235 配置设定开关 SW1~SW6 设置如表 2-7 所示。

表 2-7 EM235 配置设定开关 SW1~SW6 设置表

SW1	SW2	SW3	SW4	SW5	SW6	输入类型及范围
1	0	0	1	0	1	0~50mV
0	1	0	1	0	1	0~100mV
1	0	0	0	1	1	0~500mV
0	1	0	0	1	1	0~1V
1	0	0	0	0	1	0~5V
1	0	0	0	0	1	0~20mA
0	1	0	0	0	1	0~10V
1	0	0	1	0	0	±25mV
0	1	0	1	0	0	±50mV
0	0	1	1	0	0	±100mV
1	0	0	0	1	0	±250mV
0	1	0	0	1	0	±500mV
0	0	1	0	1	0	±1V
1	0	0	0	0	0	±2.5V
0	1	0	0	0	0	±5V
0	0	1	0	0	0	±10V

3. 热电偶、热电阻扩展模块

EM231 热电偶、热电阻扩展模块是为 S7-200 CPU222、CPU224、CPU226、CPU226XM 设计的模拟量扩展模块。EM231 热电偶模块具有特殊的冷端补偿电路，该电路测量模块连接器上的温度，并适当改变测量值，以补偿参考温度与模块温度之间的温度差，如果在 EM231 热电偶模块安装区域的环境温度迅速地变化，就会产生额外的误差，要想达到最大的精度和重复性，热电阻和热电偶模块应被安装在稳定的环境温度中。

EM231 热电偶模块用于 J、K、E、N、S、T 和 R 7 种热电偶类型。用户必须用 DIP 开关来选择热电偶的类型，连到同模块上的热电偶必须为相同类型。热电偶、热电阻扩展模块的外形如图 2-10 所示。

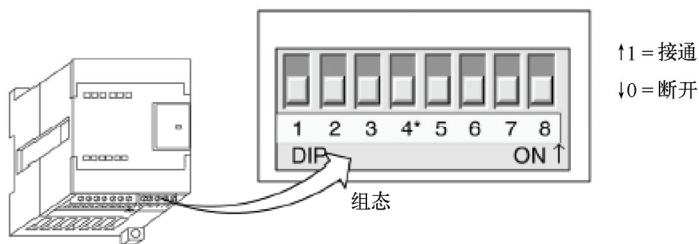


图 2-10 热电偶、热电阻扩展模块的外形图

4. PROFIBUS-DP 通信模块

通过 EM277 PROFIBUS-DP 扩展从站模块，可将 S7-200CPU 连接到 PROFIBUS-DP 网络，如图 2-11 所示。EM277 经过串行 I/O 总线连接到 S7-200 CPU，PROFIBUS 网络经过其 DP 通信端口，连接到 EM277 PROFIBUS-DP 模块。可将 EM277 PROFIBUS-DP 模块的 DP 端口连接到网络上的一个 DP 主站上，但它仍能作为一个 MPI 从站，与同一网络上（如 SIMATIC 编程器或 S7-300/S7-400 CPU 等）其他主站进行通信。

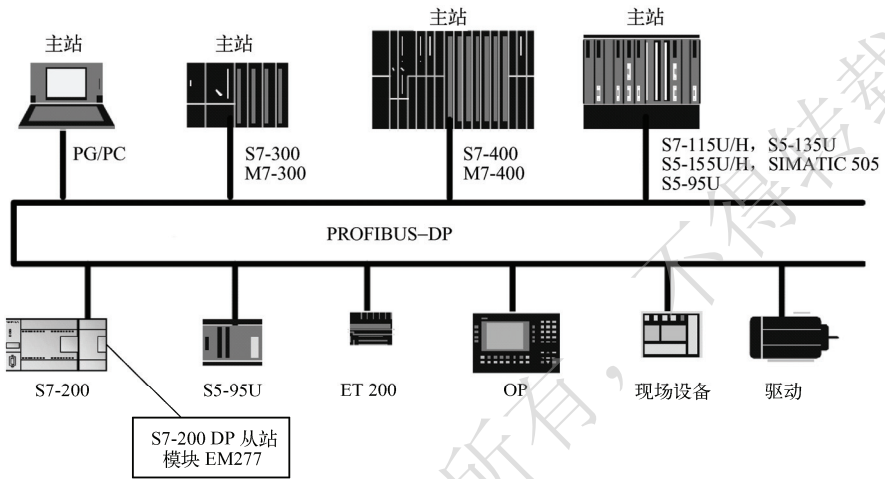


图 2-11 通过 EM 277 PROFIBUS-DP 扩展从站模块将 S7-200CPU 连接到 PROFIBUS-DP 网络

2.4 S7-200 系列 PLC 数据存储类型、编址和寻址方式、元器件功能及地址分配

2.4.1 数据存储类型

1. 数据的长度

在计算机中使用的都是二进制数，其最基本的存储单位是位 (bit)，8 位二进制数组成 1 个字节 (Byte)，其中的第 0 位为最低位 (LSB)，第 7 位为最高位 (MSB)。两个字节 (16 位) 组成 1 个字 (Word)，两个字 (32 位) 组成 1 个双字 (Double word)。位、字节、字和双字如图 2-12 所示。把位、字节、字和双字占用的连续位数称为长度。

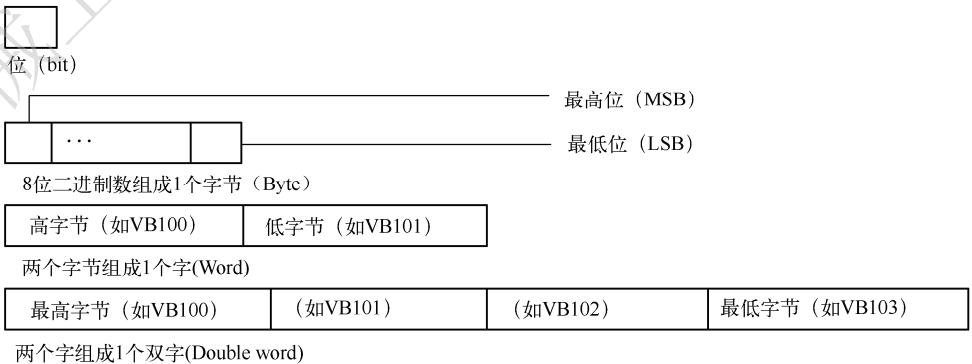


图 2-12 位、字节、字和双字

二进制数的“位”只有 0 和 1 两种取值。开关量（或数字量）也只有两种不同的状态，如触点的断开和接通，线圈的失电和得电等。在 S7-200 梯形图中，可用“位”描述它们，如果该位为 1，则表示对应的线圈为得电状态，触点为转换状态（常开触点闭合、常闭触点断开）；如果该位为 0，则表示对应线圈、触点的状态与前者相反。

在数据长度为字或双字时，起始字节均放在高位上。

2. 数据类型及数据范围

S7-200 系列 PLC 的数据类型可以是字符串、布尔型（0 或 1）、整数型和实数型（浮点数）。布尔型数据指字节型无符号整数；整数型数据包括 16 位符号整数（INT）和 32 位符号整数（DINT）。实数型数据采用 32 位单精度数来表示。数据类型、长度及数据范围如表 2-8 所示。

表 2-8 数据类型、长度及数据范围表

数据的长度、类型	无符号整数范围		符号整数范围	
	十进制	十六进制	十进制	十六进制
字节 B (8 位)	0~255	0~FF	-128~127	80~7F
字 W (16 位)	0~65 535	0~FFFF	-32 768~32 767	8000~7FFF
双字 D (32 位)	0~4 294 967 295	0~FFFFFFFF	-2 147 483 648~2 147 483 647	80000000~7FFFFFFF
位 (BOOL)	0、1			
实数	$-10^{38} \sim 10^{38}$			
字符串	每个字符串以字节形式存储，最大长度为 255 个字节，第一个字节中定义该字符串的长度			

3. 常数

S7-200 的许多指令中常会使用常数。常数的数据长度可以是字节、字和双字。CPU 以二进制的形式存储常数，书写常数可以用二进制、十进制、十六进制、ASCII 码或实数等多种形式。书写格式如下。

十进制常数：1234；十六进制常数：16#3AC6；二进制常数：2#1010 0001 1110 0000；ASCII 码：“Show”；实数（浮点数）：+1.175495E-38（正数），-1.175495E-38（负数）。

2.4.2 编址方式

可编程序控制器的编址就是对 PLC 内部的元器件进行编码，以便程序执行时可以唯一地识别每个元器件。PLC 内部在数据存储区为每一种元器件分配一个存储区域，并用字母作为区域标志符，同时表示元器件的类型。如：数字量输入写入输入映像寄存器（区标志符为 I），数字量输出写入输出映像寄存器（区标志符为 Q），模拟量输入写入模拟量输入映像寄存器（区标志符为 AI），模拟量输出写入模拟量输出映像寄存器（区标志符为 AQ）。PLC 除了输入输出外，还有其他元器件，V 表示变量存储器，M 表示内部标志位存储器，SM 表示特殊标志位存储器，L 表示局部存储器，T 表示定时器，C 表示计数器，HC 表示高速计数器，S 表示顺序控制存储器，AC 表示累加器。PLC 内部元器件如图 2-13 所示。掌握各元器件的功能和使用方法是编程的基础。下面介绍元器件的编址方式。

存储器的单位可以是位（bit）、字节（Byte）、字（Word）、双字（Double Word），也可以将编址方式分为位、字节、字、双字编址。

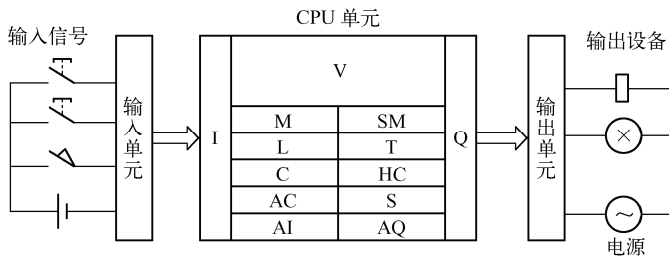


图 2-13 PLC 的内部元器件

1. 位编址

位编址的指定方式为（区域标志符）字节号·位号，如 I0.0；Q0.0；I1.2。

2. 字节编址

字节编址的指定方式为（区域标志符）B（字节号），如 IB0 表示由 I0.0~I0.7 这 8 位组成的字节。

3. 字编址

字编址的指定方式为（区域标志符）W（起始字节号），且最高有效字节为起始字节。如 VW0 表示由 VB0 和 VB1 这 2 字节组成的字。

4. 双字编址

双字编址的指定方式为（区域标志符）D（起始字节号），且最高有效字节为起始字节。如 VD0 表示由 VB0 到 VB3 这 4 字节组成的双字。

2.4.3 寻址方式

1. 直接寻址

直接寻址是在指令中直接使用存储器或寄存器的元器件名称（区域标志）和地址编号，直接到指定的区域读取或写入数据。按位、字节、字和双字的寻址方式示意图如图 2-14 所示。

2. 间接寻址

间接寻址时操作数并不提供直接数据位置，而是通过使用地址指针来存取存储器中的数据。在 S7-200 中允许使用指针对 I、Q、M、V、S、T、C（仅当前值）存储区进行间接寻址。

1) 使用间接寻址前，要先创建一指向该位置的指针。指针为双字（32 位），存放的是另一存储器的地址，只能用 V、L 或累加器 AC 作指针。生成指针时，要使用双字传送指令（MOVD），将数据所在单元的内存地址送入指针，双字传送指令的输入操作数开始处加&符号，表示某存储器的地址，而不是存储器内部的值。指令输出操作数是指针地址。例如，MOVD &VB200, AC1 指令就是将 VB200 的地址送入累加器 AC1 中。

2) 指针建立好后，利用指针存取数据。在使用地址指针存取数据的指令中，操作数前加“*”号表示该操作数为地址指针。例如，MOVW *AC1 AC0 //MOVW 表示字传送指令，指令将 AC1 中的内容为起始地址的一个字长的数据（即 VB200, VB201 内部数据）送入 AC0 中。间接寻址示意图如图 2-15 所示。

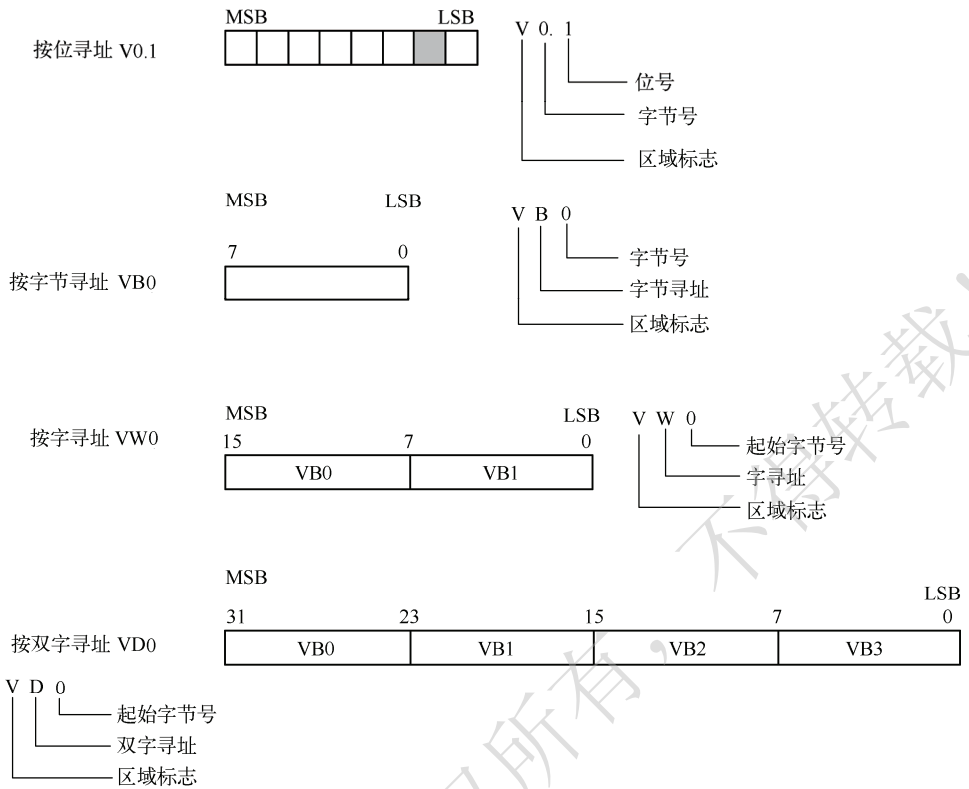


图 2-14 按位、字节、字、双字的寻址方式示意图

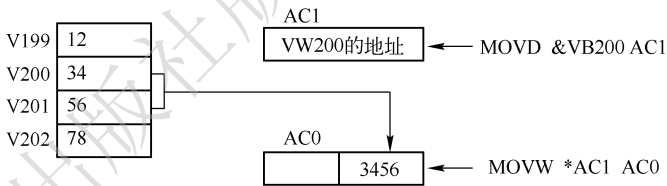


图 2-15 间接寻址示意图

2.4.4 元器件功能及地址分配

1. 输入映像寄存器（输入继电器）

1) 输入映像寄存器的工作原理。在每次扫描周期的开始，CPU 对 PLC 的实际输入端进行采样，并将采样值写入输入映像寄存器中。可以形象的将输入映像寄存器比做输入继电器，每一个“输入继电器”线圈都与相应的 PLC 输入端相连（如“输入继电器”I0.0 的线圈与 PLC 的输入端子 0.0 相连），当外部开关信号闭合时，则“输入继电器的线圈”得电，将“1”写入对应的输入映像寄存器的位中，在程序中其对应的常开触点闭合，常闭触点断开。由于存储单元可以无限次的读取，所以有无数对常开、常闭触点供编程时使用。编程时应注意，“输入继电器”的线圈只能由外部信号来驱动，即输入映像寄存器的值只能由外部的输入信号来改写，不能在程序内部用指令来驱动，因此，在用户编制的梯形图中只应出现“输

入继电器”的触点，而不应出现“输入继电器”的线圈。

2) 输入映像寄存器的地址分配。S7-200 输入映像寄存器区域有 IB0~IB15 共 16 个字节的存储单元。系统对输入映像寄存器是以字节（8 位）为单位进行地址分配的。输入映像寄存器可以按位进行操作，每一位对应一个数字量的输入点。如 CPU224 的基本单元输入为 14 点，需占用 $2 \times 8 = 16$ 位，即占用 IB0 和 IB1 两个字节。而 I1.6、I1.7 因没有实际输入而未使用，在用户程序中不可使用。但如果整个字节未使用（如 IB3~IB15），就可作为内部标志位（M）使用。

输入继电器可采用位、字节、字或双字来存取。输入继电器位存取的地址编号范围为 I0.0~I15.7。

2. 输出映像寄存器（输出继电器）

1) 输出映像寄存器的工作原理。在每次扫描周期的结尾，CPU 用输出映像寄存器中的数值驱动 PLC 输出点上的负载。可以将输出映像寄存器形象的比做输出继电器，每一个“输出继电器”线圈都与相应的 PLC 输出相连，并有无数对常开和常闭触点供编程时使用。除此之外，还有一对常开触点与相应 PLC 输出端相连（如输出继电器 Q0.0 有一对常开触点与 PLC 输出端子 0.0 相连），用于驱动负载。输出继电器线圈的通断状态只能在程序内部用指令驱动。

2) 输出映像寄存器的地址分配。S7-200 输出映像寄存器区域有 QB0~QB15 共 16 个字节的存储单元。系统对输出映像寄存器也是以字节（8 位）为单位进行地址分配的。输出映像寄存器可以按位进行操作，每一位对应一个数字量的输出点。如 CPU224 的基本单元输出为 10 点，需占用 $2 \times 8 = 16$ 位，即占用 QB0 和 QB1 两个字节。但未使用的位和字节均可在用户程序中作为内部标志位使用。

输出继电器可采用位、字节、字或双字来存取。输出继电器位存取的地址编号范围为 Q0.0~Q15.7。

以上介绍的输入映像寄存器、输出映像寄存器与输入、输出设备是有联系的，因而是 PLC 与外部联系的窗口。下面所介绍的存储器则是与外部设备没有联系的。它们既不能用来接收输入信号，也不能用来驱动外部负载，只是在编程时使用。

3. 变量存储器 V

变量存储器主要用于存储变量。可以存放数据运算的中间运算结果或设置参数，在进行数据处理时，变量存储器会被经常使用。变量存储器可以是位寻址，也可按字节、字、双字为单位寻址，其位存取的编号范围根据 CPU 的型号有所不同，CPU221/222 为 V0.0~V2047.7 共 2KB 存储容量，CPU224/226 为 V0.0~V5119.7，共 5KB 存储容量。

4. 内部标志位存储器（中间继电器）M

内部标志位存储器，用来保存控制继电器的中间操作状态，其作用相当于继电器控制中的中间继电器。内部标志位存储器在 PLC 中没有输入/输出端与之对应，其线圈的通断状态只能在程序内部用指令驱动，其触点不能直接驱动外部负载，只能在程序内部驱动输出继电器的线圈，再用输出继电器的触点去驱动外部负载。

内部标志位存储器可采用位、字节、字或双字来存取。其位存取的地址编号范围为 M0.0~M31.7，共 32 个字节。

5. 特殊标志位存储器 SM

PLC 中还有若干特殊标志位存储器，其位提供大量的状态和控制功能，用来在 CPU 和用户程序之间交换信息，它能以位、字节、字或双字来存取，CPU224 的 SM 的位地址编号范围为 SM0.0~SM179.7，共 180 个字节。其中 SM0.0~SM29.7 的 30 个字节为只读型区域。

常用的特殊存储器的用途如下。

SM0.0: 运行监视。SM0.0 始终为 1 状态。当 PLC 运行时，可以利用其触点驱动输出继电器，在外部显示程序是否处于运行状态。

SM0.1: 初始化脉冲。每当 PLC 的程序开始运行时，SM0.1 线圈接通一个扫描周期，因此 SM0.1 的触点常用于调用初始化程序等。

SM0.3: 开机进入 RUN 时，接通一个扫描周期，可用在启动操作之前，给设备提前预热。

SM0.4、SM0.5: 占空比为 50% 的时钟脉冲。当 PLC 处于运行状态时，SM0.4 产生周期为 1min 的时钟脉冲，SM0.5 产生周期为 1s 的时钟脉冲。若将时钟脉冲信号送入计数器作为计数信号，可起到定时器的作用。

SM0.6: 扫描时钟，一个扫描周期闭合，另一个为 OFF，循环交替。

SM0.7: 工作方式开关位置指示。开关放置在 RUN 位置时为 1。

SM1.0: 零标志位。运算结果等于 0 时，该位置 1。

SM1.1: 溢出标志位。结果溢出或非法值时，该位置 1。

SM1.2: 负数标志位。运算结果为负数时，该位置 1。

SM1.3: 被 0 除标志位。

其他特殊存储器的用途可查阅相关手册。

6. 局部变量存储器 L

局部变量存储器 L 用来存放局部变量，它和变量存储器 V 十分相似，主要区别在于全局变量是全局有效，即同一个变量可以被任何程序（主程序、子程序和中断程序）访问；而局部变量只是局部有效，即变量只和特定的程序相关联。

S7-200 有 64 个字节的局部变量存储器，其中 60 个字节可以作为暂时存储器，或给子程序传递参数。后 4 个字节作为系统的保留字节。PLC 在运行时，根据需要动态地分配局部变量存储器，在执行主程序时，64 个字节的局部变量存储器分配给主程序，当调用子程序或出现中断时，局部变量存储器分配给子程序或中断程序。

局部存储器可以按位、字节、字和双字直接寻址，其位存取的地址编号范围为 L0.0~L63.7。

L 可以作为地址指针。

7. 定时器 T

PLC 所提供的定时器，其作用相当于继电器控制系统中的时间继电器。每个定时器可提供无数对常开和常闭触点供编程使用。其设定时间由程序设置。

每个定时器有一个 16 位的当前值寄存器，用于存储定时器累计的时基增量值（1~32767），另有一个状态位表示定时器的状态。若当前值寄存器累计的时基增量值大于等于设定值时，定时器的状态位被置“1”，该定时器的常开触点闭合。

定时器的定时精度分别为 1ms、10ms 和 100ms 三种，CPU222、CPU224 及 CPU226 的定时器地址编号范围为 T0~T255，它们的分辨率和定时范围并不相同，用户应根据所用 CPU 型号及时基，正确选用定时器的编号。

8. 计数器 C

计数器用于累计计数输入端接收到的由断开到接通的脉冲个数。计数器可提供无数对常开和常闭触点供编程使用，其设定值由程序赋予。

计数器的结构与定时器基本相同，每个计数器有一个 16 位的当前值寄存器用于存储计数器累计的脉冲数，另有一个状态位表示计数器的状态，若当前值寄存器累计的脉冲数大于等于设定值时，计数器的状态位被置“1”，该计数器的常开触点闭合。计数器的地址编号范围为 C0~C255。

9. 高速计数器 HC

一般计数器的计数频率受扫描周期的影响，不能太高。而高速计数器可用来累计比 CPU 的扫描速度更快的事件。高速计数器的当前值是一个双字长（32 位）的整数，且为只读值。

高速计数器的地址编号范围根据 CPU 的型号有所不同，CPU221/222 各有 4 个高速计数器，CPU224/226 各有 6 个高速计数器，编号为 HC0~HC5。

10. 累加器 AC

累加器是用来暂存数据的寄存器，它可以用来存放运算数据、中间数据和结果。CPU 提供了 4 个 32 位的累加器，其地址编号为 AC0~AC3。累加器的可用长度为 32 位，可采用字节、字、双字的存取方式，按字节、字只能存取累加器的低 8 位或低 16 位，双字可以存取累加器全部的 32 位。

11. 顺序控制继电器 S（状态元器件）

顺序控制继电器是使用步进顺序控制指令编程时的重要状态元器件，通常与步进指令一起使用，以实现顺序功能流程图的编程。

顺序控制继电器的地址编号范围为 S0.0~S31.7。

12. 模拟量输入/输出映像寄存器（AI/AQ）

S7-200 的模拟量输入电路是将外部输入的模拟量信号转换成一个字长的数字量存入模拟量输入映像寄存器区域，区域标志符为 AI。

模拟量输出电路是将模拟量输出映像寄存器区域的一个字长（16 位）数值转换为模拟电流或电压输出，区域标志符为 AQ。

在 PLC 内的数字量字长为 16 位，即两个字节，故其地址均以偶数表示，如 AIW0、AIW2…；AQW0、AQW2…。

对模拟量输入/输出是以 2 个字（W）为单位分配地址，每路模拟量输入/输出占用一个字（2 个字节）。如有 3 路模拟量输入，需分配 4 个字（AIW0、AIW2、AIW4、AIW6），其中没有被使用的字 AIW6，不可被占用或分配给后续模块。如果有 1 路模拟量输出，就需分配 2 个字（AQW0、AQW2），其中没有被使用的字 AQW2，不可被占用或分配给后续模块。

模拟量输入/输出的地址编号范围根据 CPU 的型号的不同有所不同，CPU222 为 AIW0~AIW30/AQW0~AQW30；CPU224/226 为 AIW0~AIW62/AQW0~AQW62。

【例】 给表 2-9 所示的硬件组态配置 I/O 地址。

表 2-9 硬件组态及 I/O 地址

基本 I/O		扩展 I/O							
主机 CPU224		EM223 4DI/4DQ		EM221 8DI	EM235 4AI/1AQ		EM222 8DQ	EM235 4AI/1AQ	
I0.0	Q0.0	I2.0	Q2.0	I3.0	AIW0	AQW0	Q3.0	AIW8	AQW4
I0.1	Q0.1	I2.1	Q2.1	I3.1	AIW2		Q3.1	AIW10	
I0.2	Q0.2	I2.2	Q2.2	I3.2	AIW4		Q3.2	AIW12	
I0.3	Q0.3	I2.3	Q2.3	I3.3	AIW6		Q3.3	AIW14	
I0.4	Q0.4			I3.4			Q3.4		
I0.5	Q0.5			I3.5			Q3.5		
I0.5	Q0.6			I3.6			Q3.6		
I0.7	Q0.7			I3.7			Q3.7		
I1.0	Q1.0								
I1.1	Q1.1								
I1.2									
I1.3									
I1.4									
I1.5									

2.5 习题

1. S7-200 系列 PLC 有哪些编址方式？
2. S7-200 系列 CPU224 PLC 有哪些寻址方式？
3. S7-200 系列 PLC 的结构是什么？
4. CPU224 PLC 有哪几种工作方式？改变工作方式的方法有几种？
5. CPU224 PLC 有哪些元器件，它们的作用是什么？
6. CPU224 PLC 的累加器有几个？其长度是多少？
7. S7-200 系列 PLC 的数据类型有几种？各类型的数据长度是多少？
8. SM0.0、SM0.1、SM0.4 和 SM0.5 各有何作用？
9. 常见的扩展模块有几类？扩展模块的具体作用是什么？
10. PLC 外部供电电源有几种？
11. 给表 2-10 所示的硬件组态配置 I/O 地址。

表 2-10 题 11 表

基本 I/O		扩展 I/O					
主机 CPU224	EM221 8DI	EM223 8DI/8DQ	EM235 4AI/1AQ	EM223 4DI/4DQ	EM222 8DQ	EM235 4AI/1AQ	EM232 2AI

第3章 STEP7 V4.0 编程软件

本章要点

- STEP7-Micro/Win V4.0 SP9 编程软件的通信设置及窗口组件
- STEP7 编程软件的主要编程功能
- 程序的调试与监控

3.1 STEP7 V4.0 编程软件概述

S7-200 可编程序控制器使用 STEP7-Micro/Win V4.0 编程软件进行编程。STEP7-Micro/Win 编程软件是基于 Windows 的应用软件，功能强大，主要用于开发程序，也可用于适时监控用户程序的执行状态。可在全汉化的界面下进行操作。

按下列操作将英文操作界面转换成中文操作界面。打开 STEP7-Micro/Win 编程软件，在菜单栏中选中“Tools”→“Options”→“General”命令，在语言选择栏中选择“Chinese”，单击“确定”按钮，关闭软件，然后重新打开后系统即为中文界面。对于 CN 的 S7-200 PLC，STEP7 编程软件必须设置为中文界面，才能下载 PLC 程序。

3.1.1 通信设置

1. 建立 S7-200 CPU 的通信

可以采用 PC/PPI 电缆建立 PC 与 PLC 之间的通信。这是典型的单主机与 PC 的连接，不需要其他的硬件设备。PLC 与计算机的连接如图 3-1 所示。PC/PPI 电缆的两端分别为 RS-232 和 RS-485 接口。RS-232 端连接到个人计算机 RS-232 通信口的 COM1 或 COM2 接口上；RS-485 端接到 S7-200 PLC 的 CPU 通信口上。PC/PPI 电缆中间有通信模块，模块外部设有波特率设置开关，有 5 种支持 PPI 协议的波特率可以选择，分别为 1.2kbit/s，2.4kbit/s，9.6kbit/s，19.2kbit/s，38.4kbit/s。系统的默认值为 9.6kbit/s。PC/PPI 电缆波特率设置开关（DIP 开关）的位置应与软件系统设置的通信波特率相一致。DIP 开关的设置如图 3-2 所示，DIP 开关上有 5 个扳键，1、2、3 号键用于设置波特率，4 号和 5 号键用于设置通信方式。通信速率的默认值为 9 600bit/s。1、2、3 号键设置为 010，未使用调制解调器时，4、5 号键均应设置为 0。

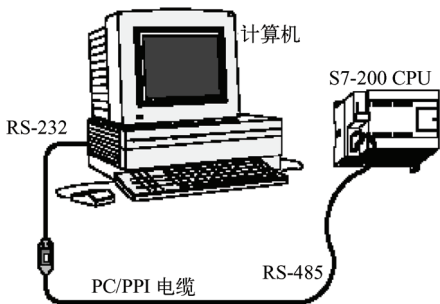
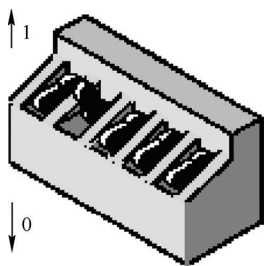


图 3-1 PLC 与计算机的连接



a)

SIEMENS		隔离的 PC/PPI电缆		PC	
PPI	波特率	123 开关	4	1 = 10 BIT	
	38.4K	000		0 = 11 BIT	
	19.2K	001			
	9.6K	010 开关	5	1 = DTE	
	2.4K	100		0 = DCE	
	1.2K	101			

b)

图 3-2 DIP 开关的设置

a) DIP 开关设置 (下=0, 上=1) b) 面板

2. 通信参数的设置

硬件设置好后, 按下面的步骤设置通信参数。

1) 在 STEP7-Micro/Win 运行时, 单击“设置 PG/PC 接口”图标, 会出现“设置 PG/PC 接口”对话框, 如图 3-3 所示。

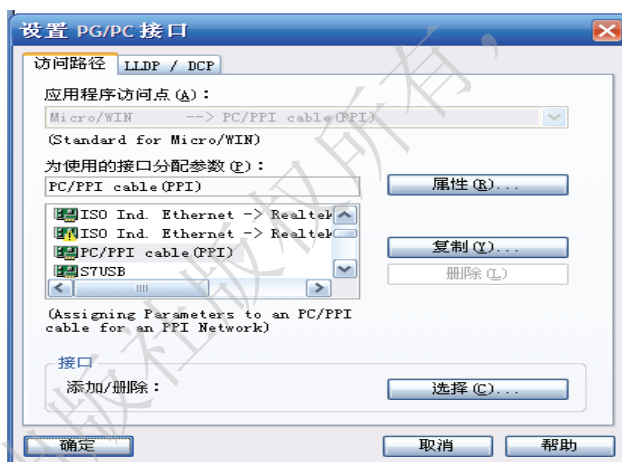


图 3-3 “设置 PG/PC 接口”对话框

2) 在“为使用的接口分配参数”中选择“PC/PPI cable (PPI)”, 然后单击“属性”按钮, 出现图 3-4 所示“属性”PPI 选项卡对话框。在“传输速率”中选择 9.6kbit/s (默认值)。然后单击“本地连接”选项卡, 出现图 3-5 所示的“属性”本地连接选项卡对话框, 如果使用的是 USB 接口的 PC/PPI 电缆, 就选择连接到 USB; 如果使用的是 COM 接口的 PC/PPI 电缆, 就选择连接到 COM1。然后单击“确定”按钮回到初始界面。

3. 建立在线连接

在前几步顺利完成后, 可以建立与 S7-200 CPU 的在线联系, 步骤如下。

在 STEP7-Micro/Win 运行时单击“通信”图标, 出现一个“通信”对话框, 如图 3-6 所示。选中“搜索所有波特率”, 双击对话框中的“双击刷新”图标, STEP7-Micro/Win 编程软件将检查所连接的所有 S7-200 CPU 站。在对话框中显示已建立起连接的每个站的 CPU 图标、CPU 型号和站地址。PC 与 PLC 的通信连接成功对话框如图 3-7 所示。能够刷新到 PLC 的地址, 说明 PC 与 PLC 的通信连接成功。



图 3-4 “属性” PPI 选项卡对话框

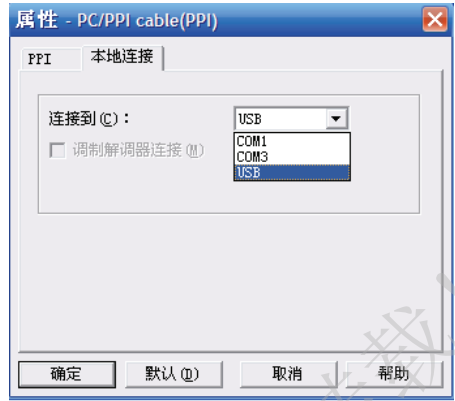


图 3-5 “属性”本地连接选项卡对话框

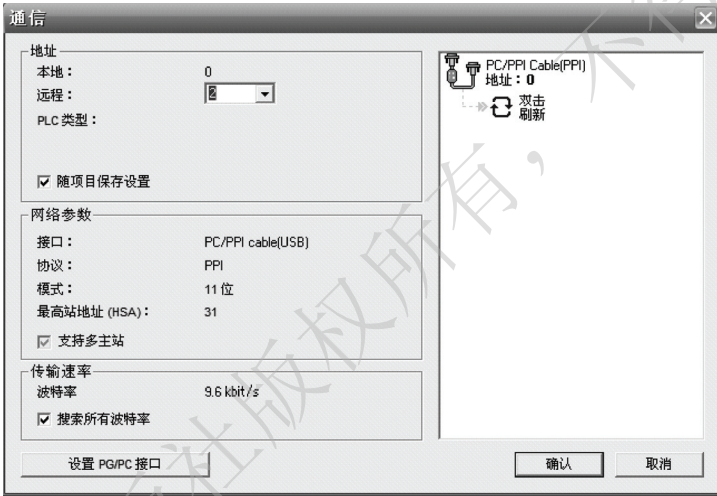


图 3-6 “通信”对话框

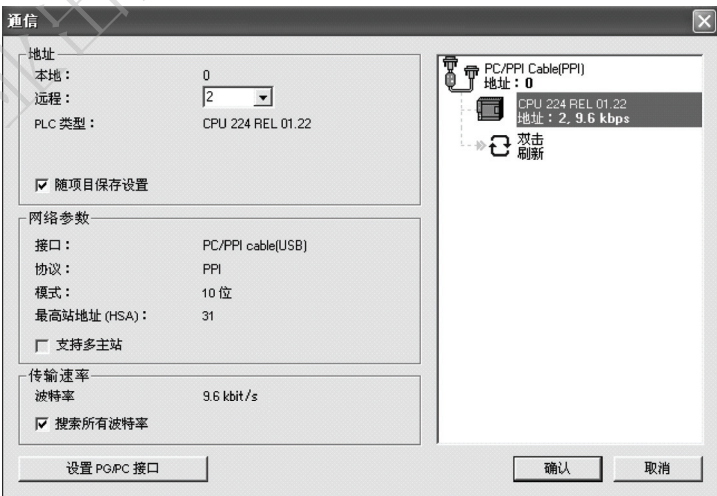


图 3-7 PC 与 PLC 的通信连接成功对话框

4. 修改 PLC 的通信参数

计算机与可编程序控制器建立起在线连接后，即可以利用软件检查、设置和修改 PLC 的通信参数。步骤如下。

1) 单击浏览条中的系统块图标，将出现“系统块”对话框，如图 3-8 所示。

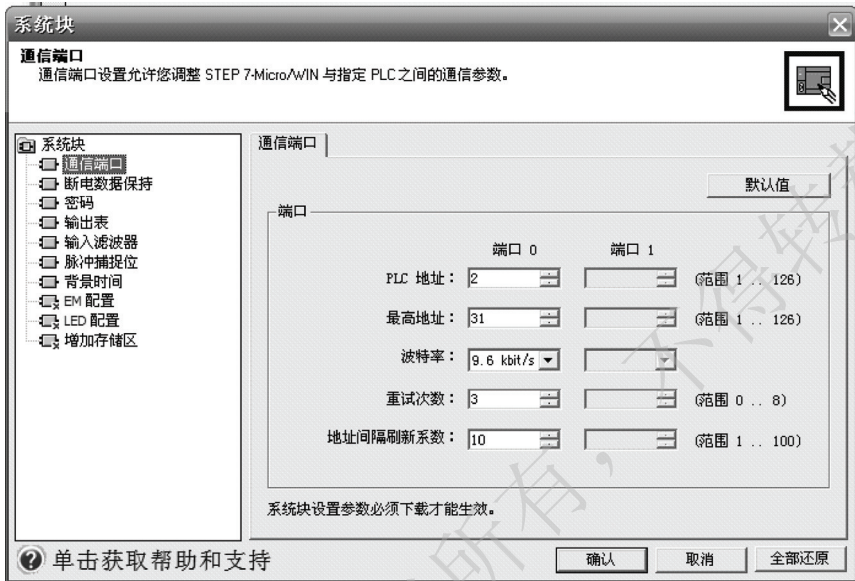



图 3-8 “系统块”对话框

2) 单击“通信口”选项卡，检查各参数，确认无误后单击“确定”按钮。若需修改某些参数，可以先进行有关修改，再单击“确认”按钮。

3) 单击工具条的下载按钮，将修改后的参数下载到可编程序控制器中，设置的参数才会起作用。

5. 可编程序控制器信息的读取

选择菜单命令“PLC”，找“信息”，将显示出可编程序控制器 RUN/STOP 的状态、扫描速率、CPU 的型号错误的情况和各模块的信息。

3.1.2 STEP7-Micro/Win V4.0 SP9 窗口组件

STEP7-Micro/Win V4.0 SP9 的主界面如图 3-9 所示。主界面一般可以分为以下几个部分：菜单条、工具条、浏览条、指令树、用户窗口、输出窗口和状态条。除菜单条外，用户还可以根据需要通过查看菜单和窗口菜单决定其他窗口的取舍和样式的设置。

1. 主菜单

主菜单包括文件、编辑、查看、PLC、调试、工具、窗口和帮助 8 个主菜单项。各主菜单项的功能如下。

1) 文件 (File)。文件的操作有新建 (New)、打开 (Open)、关闭 (Close)、保存 (Save)、另存 (Save As)、导入 (Import)、导出 (Export)、上载 (Upload)、下载 (Download)、页面设置 (Page Setup)、打印 (Print)、预览、最近使用文件和退出。



图 3-9 STEP7-Micro/Win V4.0 SP9 的主界面

① 导入。若从 STEP 7-Micro/Win 32 编辑器之外导入程序，可使用“导入”命令导入 ASCII 文本文件。

② 导出。使用“导出”命令创建程序的 ASCII 文本文件，并导出至 STEP7-Micro/Win32 外部的编辑器。

③ 上载。在运行 STEP7-Micro/Win 32 的个人计算机与 PLC 之间建立通信后，从 PLC 将程序上载至运行 STEP 7-Micro/Win 32 的个人计算机。

④ 下载。在运行 STEP 7-Micro/Win32 的个人计算机与 PLC 之间建立通信后，将程序下载至该 PLC 中。下载之前，PLC 应位于“停止”模式。

2) 编辑 (Edit)。编辑菜单提供程序的编辑工具有撤销 (Undo)、剪切 (Cut)、复制 (Copy)、粘贴 (Paste)、全选 (Select All)、插入 (Insert)、删除 (Delete)、查找 (Find)、替换 (Replace) 和转至 (Go To) 等项目。

3) 查看 (View)。

① 通过查看菜单可以选择不同的程序编辑器，即 LAD (梯形图)，STL (语句表)，FBD (功能块)。

② 通过查看菜单可以进行项目组件的设置，如数据块 (Data Block)、符号表 (Symbol Table)、状态表 (Chart Status)、系统块 (System Block)、交叉引用 (Cross Reference) 和通信 (Communications) 参数的设置。

③ 通过查看菜单可以选择注解、网络注解 (POU Comments) 显示与否等。

④ 通过查看菜单的框架栏区可以选择浏览栏 (Navigation Bar)、指令树 (Instruction Tree) 及输出视窗 (Output Window) 的显示与否。

⑤ 通过查看菜单的工具栏区可以选择标准、调试、公用和指令等快捷工具显示与否。

⑥ 通过查看菜单可以对程序块的属性进行设置。

4) PLC。PLC 菜单用于与 PLC 联机时的操作。如用软件改变 PLC 的运行方式（运行、停止），对用户程序进行编译，清除 PLC 程序、上电复位、查看 PLC 的信息、时钟、存储卡的操作、程序比较和 PLC 类型选择等操作。其中，对用户程序进行编译可以离线进行。

联机方式（在线方式）：有编程软件的计算机与 PLC 连接，两者之间可以直接通信。

离线方式：有编程软件的计算机与 PLC 断开连接。此时可进行编程、编译。

PLC 有两种操作模式：STOP（停止）和 RUN（运行）模式。在 STOP（停止）模式中
可以建立/编辑程序，在 RUN（运行）模式中监控程序操作和数据，进行动态调试。

若使用 STEP 7-Micro/Win 软件控制 RUN/STOP（运行/停止）模式，则在 STEP 7-Micro/Win 和 PLC 之间必须建立通信。另外，PLC 硬件模式开关必须设为 TERM（终端）或 RUN（运行）。

编译（Compile）。用来检查用户程序语法错误。用户程序编辑完成后，通过编译在显示器下方的输出窗口显示编译结果，明确指出错误的网络段，可以根据错误提示对程序进行修改，然后再编译，直至无错误。

全部编译（Compile All）。编译全部项目元件（程序块、数据块和系统块）。

信息（Information）。可以查看 PLC 信息，例如，PLC 型号和版本号码、操作模式、扫描速率、I/O 模块配置以及 CPU 和 I/O 模块错误等。

上电复位（Power-Up Reset）。从 PLC 清除严重错误，并返回 RUN（运行）模式。如果操作 PLC 存在严重错误，SF（系统错误）指示灯就亮，程序停止执行。必须将 PLC 模式重设为 STOP（停止），然后再设置为 RUN（运行），才能清除错误，或使用“PLC”→“上电复位”。

5) 调试（Debug）。调试菜单用于联机时的动态调试，有首次扫描（First Scan）、多次扫描（Multiple Scans）、开始程序状态监控（Start Program Status）、暂停程序状态监控（Pause Program Status）、状态表监控（Start Chart Status）、暂停趋势图监控（Pause Trend Chart）、用程序状态模拟运行条件（读取、强制、取消强制和全部取消强制）等功能。

调试时，可以指定 PLC 对程序执行有限次数扫描（1~65 535 次扫描）。通过选择 PLC 运行的扫描次数，可以在程序改变过程变量时对其进行监控。第一次扫描时，SM0.1 数值为 1（打开）。

首次扫描：可编程序控制器从 STOP 方式进入 RUN 方式，执行一次扫描后，回到 STOP 方式，可以观察到首次扫描后的状态。

PLC 必须位于 STOP（停止）模式，通过菜单“调试”→“单次扫描”命令进行操作。

多次扫描：调试时可以指定 PLC 对程序执行有限次数扫描（从 1 次扫描到 65 535 次扫描）。通过选择 PLC 运行的扫描次数，可以在程序过程变量改变时对其进行监控。

PLC 必须位于 STOP（停止）模式，通过菜单“调试”→“多次扫描”设置扫描次数。

6) 工具。

① 工具菜单提供复杂指令向导（PID、HSC、NETR/NETW 指令），简化复杂指令编程的工作。

② 工具菜单提供文本显示器 TD200 设置向导。

③ 工具菜单的定制子菜单可以更改 STEP 7-Micro/Win 32 工具条的外观或内容，以及在

“工具”菜单中增加常用工具。

④ 工具菜单的选项子菜单可以设置 3 种编辑器的风格，如字体、指令盒的大小等样式。

7) 窗口。窗口菜单可以设置窗口的排放形式，如层叠、水平、垂直。

8) 帮助。帮助菜单可以提供 S7-200 的指令系统及编程软件的所有信息，并提供在线帮助、网上查询、访问等功能。

2. 工具条

1) 标准工具条如图 3-10 所示。各快捷按钮从左到右分别为：新建项目、打开现有项目、保存当前项目、打印、打印预览、剪切选项并复制至剪贴板、将选项复制至剪贴板、在光标位置粘贴剪贴板内容、撤销最后一个条目、编译程序块或数据块（任意一个现用窗口）、全部编译（程序块、数据块和系统块）、将项目从 PLC 上载至 STEP 7-Micro/Win、从 STEP 7-Micro/Win 下载至 PLC、符号表名称列按照 A~Z 从小至大排序、符号表名称列按照 Z~A 从大至小排序、选项（配置程序编辑器窗口）。



图 3-10 标准工具条

2) 调试工具条如图 3-11 所示。各快捷按钮从左到右分别是，将 PLC 设为运行模式、将 PLC 设为停止模式、在程序状态打开/关闭之间切换、在触发暂停打开/停止之间切换（只用于语句表）、在状态表监控打开/关闭之间切换、状态表单次读取、状态表全部写入、强制 PLC 数据、取消强制 PLC 数据、状态表全部取消强制、状态表全部读取强制数值、趋势图。



图 3-11 调试工具条

3) 公用工具条如图 3-12 所示。公用工具条各快捷按钮从左到右的含义分别为：

① 插入网络。单击该按钮，在 LAD 或 FBD 程序中插入一个空网络。

② 删除网络。单击该按钮，删除 LAD 或 FBD 程序中的整个网络。

③ 程序注解。单击该按钮在程序注解打开（可视）或关闭（隐藏）之间切换。可视时，始终位于第一个网络之前显示。POU 注解如图 3-13 所示。

④ 网络注解。单击该按钮，在光标所在的网络标号下方出现灰色方框中，输入网络注解。再单击该按钮，网络注解关闭。网络注解如图 3-14 所示。



图 3-12 公用工具条

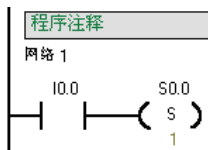


图 3-13 POU 注解

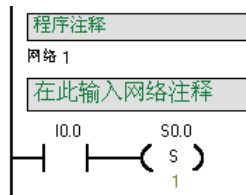


图 3-14 网络注解

⑤ 查看/隐藏每个网络的符号信息表。单击该按钮，在符号信息表打开和关闭之间切换。网络的符号信息表如图 3-15 所示。

⑥ 切换书签。设置或移除书签。网络设置书签如图 3-16 所示。

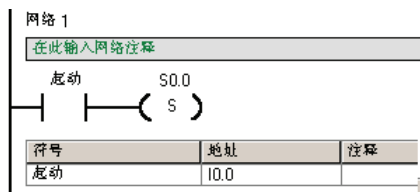


图 3-15 网络的符号信息表

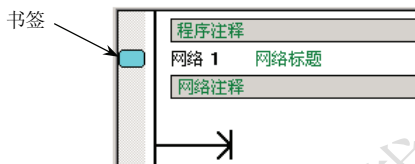


图 3-16 网络设置书签

⑦ 下一个书签。单击该按钮，向下移至程序的下一个带书签的网络。

⑧ 前一个书签。单击该按钮，向上移至程序的前一个带书签的网络。

⑨ 清除全部书签。单击该按钮，移除程序中的所有当前书签。

在项目中应用所有的符号。单击该按钮，用所有新、旧和修改的符号名更新项目，并在符号信息表打开和关闭之间切换。

建立表格未定义符号。单击该按钮，从程序编辑器将不带指定地址的符号名传输至指定地址的新符号表标记。

常量说明符。在 SIMATIC 类型说明符打开 / 关闭之间切换，单击“常量描述符”按钮，使常量描述符可视或隐藏。对许多指令参数可直接输入常量。仅被指定为 100 的常量具有不确定的大小，因为常量 100 可以表示为字节、字或双字大小。当输入常量参数时，程序编辑器根据每条指令的要求指定或更改常量描述符。

4) LAD 指令工具条如图 3-17 所示。从左到右分别是，插入向下直线，插入向上直线，插入左行，插入右行，插入接点，插入线圈，插入指令盒。



图 3-17 LAD 指令工具条

3. 浏览条 (Navigation Bar)

浏览条为编程提供按钮控制，可以实现窗口的快速切换，即对编程工具执行直接按钮存取，包括程序块 (Program Block)、符号表 (Symbol Table)、状态表表 (Status Chart)、数据块 (Data Block)、系统块 (System Block)、交叉引用 (Cross Reference)、通信 (Communication) 和设置 PG/PC 接口。单击上述任意按钮，则主窗口切换成此按钮对应的窗口。

4. 指令树 (Instuction Tree)

指令树提供编程时用到的所有快捷操作命令和 PLC 指令。可分为项目分支和指令分支。

1) 项目分支用于组织程序项目。

① 用鼠标右键单击“程序块”文件夹，插入新子程序和中断程序。

② 打开“程序块”文件夹，并用鼠标右键单击 POU 图标，可以打开 POU、编辑 POU 属性、用密码保护 POU 或为子程序和中断程序重新命名。

③ 用鼠标右键单击“状态图”或“符号表”文件夹，插入新图或表。

④ 打开“状态图”或“符号表”文件夹，在指令树中用鼠标右键单击图或表图标，或

双击适当的 POU 标记, 执行打开、重新命名或删除操作。

2) 指令分支用于输入程序, 打开指令文件夹并选择指令。

① 拖放或双击指令, 可在程序中插入指令。

② 用鼠标右键单击指令, 并从弹出菜单中选择“帮助”, 获得有关该指令的信息。

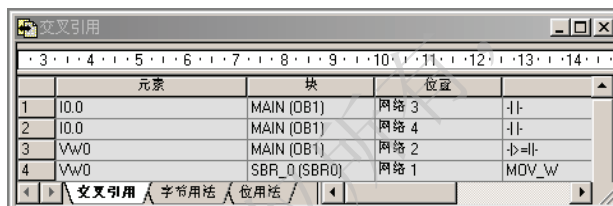
③ 将常用指令可拖放至“偏好项目”文件夹。

④ 若项目指定了 PLC 类型, 指令树中红色标记 x 是表示对该 PLC 无效的指令。

5. 用户窗口

可同时或分别打开图 3-9 中的 6 个用户窗口, 分别为交叉引用、数据块、状态表、符号表、程序编辑器和局部变量表。


1) 交叉引用 (Cross Reference)。在程序编译成功后, 才能打开交叉引用表。如图 3-18 所示, “交叉引用”表列出在程序中使用的各操作数所在的位置以及每次使用各操作数的指令。通过交叉引用表, 还可以查看哪些内存区域已经被使用, 作为位还是作为字节使用。交叉引用表不下载到可编程控制器, 在交叉引用表中双击某操作数, 可以显示出包含该操作数的那一部分程序。





	元素	块	位置	指令
1	I0.0	MAIN (OB1)	网络 3	↑↑
2	I0.0	MAIN (OB1)	网络 4	↑↑
3	VW0	MAIN (OB1)	网络 2	↓=↑↑
4	VW0	SBR_0 (SBR0)	网络 1	MOV_W


图 3-18 交叉引用表

2) 数据块。“数据块”窗口可以设置和修改变量存储器的初始值和常数值, 并加注必要的注释说明。单击浏览条上的“数据块”按钮, 可打开“数据块”窗口。

3) 状态表 (Status Chart)。将程序下载至 PLC 之后, 可以建立一个或多个状态表, 在联机调试时, 打开状态表, 监视各变量的值和状态。状态表并不被下载到可编程控制器中, 它只是监视用户程序运行的一种工具。单击浏览条上的“状态表”按钮, 可打开状态表。

若在项目中有一个以上的状态表, 使用位于“状态表”窗口底部的“表”标签在状态表之间移动。可在状态表的地址列输入需监视的程序变量地址。在 PLC 运行时, 打开状态表窗口, 在程序扫描执行时, 连续、自动地更新状态表的数值。

4) 符号表 (Symbol Table)。用有实际含义的自定义符号名作为编程元件的操作数, 这样可使程序更容易理解。符号表则建立了自定义符号名与直接地址编号之间的关系。单击浏览条中的“符号表”按钮, 可打开符号表。

5) 程序编辑器。用菜单命令“文件”→“新建”“文件”→“打开”或“文件”→“导入”命令, 可打开一个项目。然后单击浏览条中的“程序块”按钮, 打开“程序编辑器”窗口, 建立或修改程序。可用菜单命令“查看”→STL、LAD、FBD, 更改编辑器类型。

6) 局部变量表。每个程序块都有自己的局部变量表。局部变量只在建立该局部变量的程序块中才有效。在带参数的子程序调用中, 参数的传递是通过局部变量表传递的。

6. 输出窗口

输出窗口用来显示程序编译的结果，如编译结果有无错误、错误编码和位置等。

7. 状态条

状态条提供有关在 STEP 7-Micro/Win 中操作的信息。

3.1.3 编程准备

1. 指令集和编辑器的选择

写程序之前，用户必须选择指令集和编辑器。

S7-200 系列 PLC 支持的指令集有 SIMATIC 和 IEC1131-3 两种。SIMATIC 是专为 S7-200PLC 设计的，专用性强，采用 SIMATIC 指令编写的程序执行时间短，可以使用 LAD、STL、FBD 3 种程序编辑器。IEC1131-3 指令集是按国际电工委员会（IEC）PLC 编程标准提供的指令系统，作为不同 PLC 厂商的指令标准，集中指令较少。有些 SIMATIC 所包含的指令，在 IEC 1131-3 中不是标准指令。IEC1131-3 标准指令集适用于不同厂家 PLC，可以使用 LAD 和 FBD 两种编辑器。本书主要用 SIMATIC 编程模式。

菜单命令“工具”→“选项”→“常规”选项卡→“编程模式”→选 SIMATIC。

对 LAD、STL、FBD 这 3 种程序编辑器的比较在下一章介绍。本书主要介绍 LAD 和 STL。选择编辑器的方法如下：① 用菜单命令“查看”→梯形图或 STL；② 菜单命令“工具”→“选项”→“常规”选项卡→“梯形图编辑器”。

2. 根据 PLC 类型进行参数检查

在 PLC 和运行 STEP7-Micro/Win 的 PC 连线后，应根据 PLC 的类型进行范围检查。必须保证 STEP7-Micro/Win 中 PLC 类型选择与实际 PLC 类型相符。具体方法如下：菜单命令“PLC”→“类型”→“读取 PLC”。“PLC 类型”对话框如图 3-19 所示。

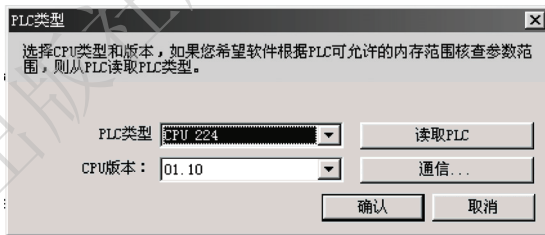


图 3-19 “PLC 类型”对话框

3.2 STEP7-Micro/Win 主要编程功能

3.2.1 梯形图程序的输入

1. 建立项目

1) 打开已有的项目文件。用菜单命令“文件”→“打开”，在“打开文件”对话框中，选择项目的路径及名称，单击“确定”按钮，打开现有项目。

2) 创建新项目。菜单命令“文件”→“新建”；或者单击浏览条中的程序块图标，新建一个项目。

2. 输入程序

打开项目后就可以进行编程，本书主要介绍梯形图的相关操作。

1) 输入指令。梯形图的元素主要有触点、线圈和指令盒，对梯形图的每个网络，必须从接点开始，以线圈或没有 ENO 输出的指令盒结束。不允许线圈串联使用。

要输入梯形图指令首先要进入梯形图编辑器，方法如下。

① “查看” → 单击“梯形图”选项。接着在梯形图编辑器中输入指令。输入指令可以通过指令树、工具条按钮、快捷键等方法。

② 在指令树中选择需要的指令，拖放到需要的位置上。

③ 将光标放在需要的位置，在指令树中双击需要的指令。


④ 将光标放到需要的位置，单击工具栏指令按钮，打开一个通用指令窗口，选择需要的指令。

⑤ 使用功能键。F4（触点）、F6（线圈）、F9（指令盒），打开一个通用指令窗口，选择需要的指令。

在编程元件图形出现在指定位置后，再单击编程元件符号的“???”，输入操作数。红色字样显示语法出错，当把不合法的地址或符号改变为合法值时，红色消失。若数值下面出现红色的波浪线，则表示输入的操作数超出范围或与指令的类型不匹配。


2) 上下线的操作。将光标移到要合并的接点处，单击工具栏中向上连线[↑]或向下连线[↓]按钮。

3) 输入程序注释。LAD 编辑器中共有 4 个注释级别：项目组件（POU）注释、网络标题、网络注释、项目组件属性。

① 项目组件（POU）注释。在“网络 1”上方的灰色方框中单击，输入 POU 注释。单击“切换 POU 注释”按钮或者用菜单命令“查看” → “POU 注释”选项，在 POU 注释“打开”（可视）或“关闭”（隐藏）之间切换。可视时，始终位于 POU 顶端，并在第一个网络之前显示。

② 网络标题。将光标放在网络标题行，输入一个识别便于该逻辑网络的标题。

③ 网络注释。将光标移到网络标号下方的灰色方框中，可以输入网络注释。网络注释可对网络的内容进行简单的说明，以便于程序的理解和阅读。


④ 单击“切换网络注释”按钮或者用菜单命令“查看” → “网络注释”，可在网络注释“打开”（可视）和“关闭”（隐藏）之间切换。

4) 程序的编辑。


① 剪切、复制、粘贴或删除多个网络。通过按〈Shift〉键+鼠标单击，可以选择多个相邻的网络，进行剪切、复制、粘贴或删除等操作。注意：不能选择部分网络，只能选择整个网络。

② 编辑单元格、指令、地址和网络。用光标选中需要进行编辑的单元，单击右键，弹出快捷菜单，可以进行插入或删除行、列、垂直线或水平线的操作。删除垂直线时把方框放在垂直线左边单元上，删除时选“行”，或按〈Del〉键。进行插入编辑时，先将方框移至欲插入的位置，然后选“列”。

5) 程序的编译。程序经过编译后，方可下载到 PLC 中。编译的方法如下：

① 单击“编译”按钮或选择菜单命令“PLC” → “编译”（Compile），编译当前被

激活的窗口中的程序块或数据块。

② 单击“全部编译”按钮或选择菜单命令“PLC”→“全部编译”(Compile All)，编译全部项目元件(程序块、数据块和系统块)。使用“全部编译”，与哪一个窗口是活动窗口无关。

编译结束后，输出窗口显示编译结果。

3.2.2 数据块的编辑

数据块窗口(如图 3-20 所示)用来对变量存储器 V 赋初值，可用字节、字或双字赋值。注解(前面带双斜线)是可选项目。编写的块被编译后，可下载到可编程序控制器中，注释被忽略。

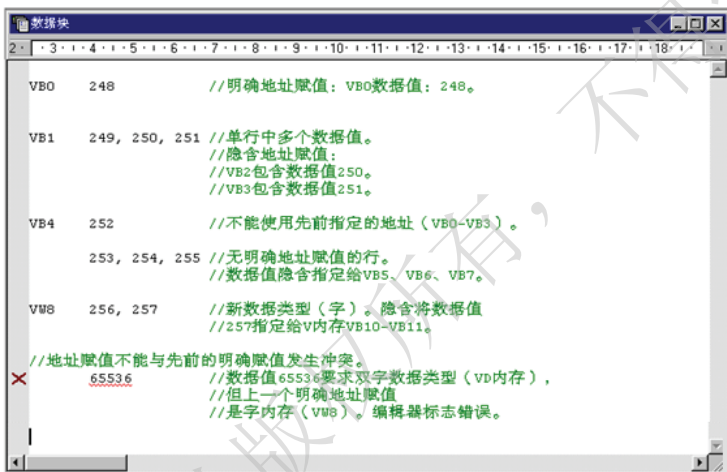


图 3-20 数据块窗口


数据块的第一行必须包含一个明确地址，以后的行可包含明确或隐含地址。在单地址后键入多个数据值或键入仅包含数据值的行时，由编辑器指定隐含地址。编辑器根据以前的地址分配及数据长度(字节、字或双字)指定适当的 V 内存数量。

键入的地址和数据之间留有空格。键入一行后，按〈Enter〉键，数据块编辑器格式化行(对齐地址列、数据、注解；捕获 V 内存地址)并重新显示。

数据块需要下载至 PLC 后才起作用。

3.2.3 符号表操作

1. 在符号表中符号赋值的方法

1) 建立符号表。单击浏览条中的“符号表”按钮。符号表见图 3-21。

			符号	地址	注释
1			启动	I0.0	启动按钮SB2
2			停止	I0.1	停止按钮SB1
3			M1	Q0.0	电动机
4					
5					

图 3-21 符号表

2) 在“符号”列键入符号名称(如,启动)。注意:在给符号指定地址之前,该符号下有绿色波浪下划线;在给符号指定地址后,绿色波浪下划线自动消失。

3) 在“地址”列中键入地址(例如: I0.0)。

4) 键入注解(此为可选项)。

5) 符号表建立后,使用菜单命令“查看”→选中“符号寻址”,直接地址将转换成符号表中对应的符号名,并且可通过菜单命令“工具”→“选项”→“程序编辑器”选项卡→“符号寻址”选项,来选择操作数显示的形式。如选择“显示符号和地址”,则对应的带符号表的梯形图如图 3-22 所示。

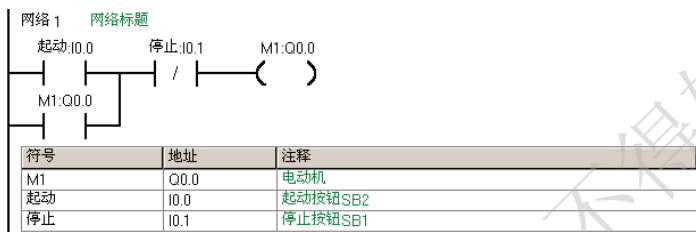


图 3-22 带符号表的梯形图

6) 使用菜单命令“查看”→“符号信息表”,可选择符号表的显示与否;使用菜单命令“查看”→“符号寻址”,可选择是否已将直接地址转换成对应的符号名。

2. 在符号表中插入行

使用下列方法之一在符号表中插入行。

1) 选择菜单命令“编辑”→“插入”→“行”,将在符号表光标的当前位置上方插入新行。

2) 用鼠标右键单击符号表中的一个单元格,选择弹出菜单中的命令“插入”→“行”,将在光标的当前位置上方插入新行。

3) 若在符号表底部插入新行,将光标放在最后一行的任意一个单元格中,按〈↓〉键即可。

3. 建立多个符号表

默认情况下,符号表窗口显示一个符号名称(USR1)的选项卡。可用下列方法建立多个符号表。

1) 从“指令树”用鼠标右键单击“符号表”文件夹,在弹出菜单命令中选择“插入符号表”。

2) 打开符号表窗口,使用“编辑”菜单或用鼠标右键单击,在弹出菜单命令中选择“插入”→“表格”。

插入新符号表后,新的符号表标签会出现在符号表窗口的底部。在打开符号表时,要选择正确的标签。双击或用鼠标右键单击标签,可为标签重新命名。

3.3 程序的下载和上载

1. 下载

如果已经成功地在运行 STEP 7-Micro/Win 的个人计算机和 PLC 之间建立了通信,就可

以将编译好的程序下载至该 PLC 中。如果 PLC 中已经有内容，就将被覆盖。下载步骤如下：

1) 下载之前，PLC 必须位于“停止”的工作方式。检查 PLC 上的工作方式指示灯，如果 PLC 没有在“停止”，那么单击工具条中的“停止”按钮，将 PLC 至于停止方式。

2) 单击工具条中的“下载”按钮，或用菜单命令“文件”→“下载”。出现“下载”对话框。

3) 根据默认值，在初次发出下载命令时，“程序块”、“数据块”和“系统块”复选框都被选中。如果不需要下载某个块，就可以清除该复选框。

4) 单击“确定”按钮，开始下载程序。如果下载成功，就将出现一个确认框显示“下载成功”。

5) 如果 STEP 7-Micro/Win 中的 CPU 类型与实际的 PLC 不匹配，就会显示以下警告信息：“为项目所选的 PLC 类型与远程 PLC 类型不匹配。继续下载吗？”

6) 此时应纠正 PLC 类型选项，选择“否”，终止下载程序。

7) 用菜单命令“PLC”→“类型”，调出“PLC 类型”对话框。单击“读取 PLC”按钮，由 STEP 7-Micro/Win 自动读取正确的数值。单击“确定”按钮，确认 PLC 类型。

8) 单击工具条中的“下载”按钮，重新开始下载程序，或用菜单命令“文件”→“下载”。

下载成功后，单击工具条中的“运行”按钮，或“PLC”→“运行”，PLC 将进入 RUN（运行）工作方式。

2. 上载

用下面的方法从 PLC 将项目元器件上载到 STEP 7-Micro/Win 程序编辑器中。

1) 单击“上载”按钮。

2) 选择菜单命令“文件”→“上载”。



3) 按〈Ctrl+U〉组合键。

4) 执行的步骤与下载基本相同，选择所需的上载块（程序块、数据块或系统块），单击“上载”按钮，上载的程序将从 PLC 复制到当前打开的项目中，随后即可保存上载的程序。

3.4 程序的调试与监控

在运行 STEP 7-Micro/Win 编程设备和 PLC 之间建立通信并向 PLC 下载程序后，便可运行程序，进行监控和调试程序。

3.4.1 选择工作方式

PLC 有运行和停止两种工作方式。在不同的工作方式下，PLC 进行调试的操作方法不同。单击工具栏中的“运行”按钮  或“停止”按钮  可以进入相应的工作方式。

1. 选择 STOP 工作方式

在 STOP（停止）工作方式中，可以创建和编辑程序，使 PLC 处于半空闲状态：停止用户程序执行；执行输入更新；用户中断条件被禁用。PLC 操作系统继续监控 PLC，将状态数据传递给 STEP 7-Micro/Win。并执行所有的“强制”或“取消强制”命令。当 PLC 位于

STOP（停止）工作方式时，可以进行下列操作。

1) 使用“状态表监控”或“程序状态监控”查看操作数的当前值。因为程序未执行，所以这个步骤等同于执行“单次读取”。

2) 可以使用“状态表监控”或“程序状态监控”强制数值。使用“状态表监控”写入数值。

3) 写入或强制输出。

4) 执行有限次扫描，并通过“状态表监控”或“程序状态监控”观察结果。

2. 选择运行工作方式

当 PLC 位于 RUN（运行）工作方式时，不能使用“首次扫描”或“多次扫描”功能。可以在“状态表”中写入和强制数值，或使用 LAD 或 FBD 程序编辑器强制数值，方法与在 STOP（停止）工作方式中强制数值相同。还可以执行下列操作（不能在 STOP 工作方式使用）。

1) 使用“状态表监控”收集 PLC 数据值的连续更新。如果希望使用单次更新，“状态表监控”必须关闭，才能使用“单次读取”命令。

2) 使用“程序状态监控”收集 PLC 数据值的连续更新。




3.4.2 显示程序状态

在程序下载至 PLC 后，可以用“程序状态监控”功能测试程序网络。

1. 启动程序状态

1) 在程序编辑器窗口，显示希望操作和测试的程序部分。


2) PLC 置于 RUN 工作方式，启动“程序状态监控”查看 PLC 数据值。方法如下：

单击“程序状态监控”按钮或用菜单命令“调试”→“程序状态监控”，在梯形图中显示出各元件的状态。在进入“程序状态监控”的梯形图中，用彩色块表示位操作数的线圈得电或接点闭合状态。如：表示接点闭合状态；表示位操作数的线圈得电。

运行中梯形图内各元件的状态将随程序执行过程连续更新变换。

2. 用“程序状态监控”模拟进程条件（读取、强制、取消强制和全部取消强制）


在程序状态监控过程中，使用从程序编辑器向操作数写入或强制新数值的方法，可以模拟进程条件。

单击“程序状态监控”按钮，开始监控数据状态，并启用调试工具。

(1) 写入操作数


直接单击操作数（不要单击指令），然后用鼠标右键直接单击操作数，并从弹出菜单选择“写入”命令。

(2) 强制单个操作数

直接单击操作数（不是指令），然后从“调试”工具条中单击“强制”图标。


直接用鼠标右键单击操作数（不是指令），并从弹出菜单选择“强制”命令。

(3) 单个操作数取消强制

直接单击操作数（不是指令），然后从“调试”工具条中单击“取消强制”图标。

直接用鼠标右键单击操作数（不是指令），并从弹出菜单选择“取消强制”命令。


(4) 全部强制数值取消强制


从“调试”工具条中单击“全部取消强制”图标。


注意：强制功能是调试程序的辅助工具，切勿为了弥补处理装置的故障而执行强制。仅限合格人员使用强制功能。在不带负载的情况下调试程序时，可以使用强制功能。

3. 识别强制图标

被强制的数据处将显示一个图标。

1) 黄色锁定图标表示显示强制。即该数值已经被直接强制为当前正在显示的数值。


2) 灰色隐去锁定图标表示隐式。该数值已经被“隐含”强制，即不对地址进行直接强制，但内存区落入另一个被明确强制的较大区域中。例如，如果 VW0 被显示强制，则 VB0 和 VB1 被隐含强制，因为它们包含在 VW0 中。

3) 半块图标表示部分强制。例如，VB1 被明确强制，则 VW0 被部分强制，因为其中的一个字节 VB1 被强制。

3.4.3 显示状态表

可以建立一个或多个状态表，用来监管和调试程序操作。

1. 打开状态表

打开状态表。单击浏览条上的“状态表”按钮。如果在项目中有多个状态表，使用“状态表”窗口底部的“表”标签，可在状态表之间移动。

2. 状态表的创建和编辑

1) 建立状态表。如果打开一个空状态表，可以输入地址或定义符号名。按以下步骤定义状态表，如图 3-23 所示。

	地址	格式	当前值	新数值
1	I0.0	位		
2	VW0	带符号		
3	M0.0	位		
4	SMW70	带符号		

图 3-23 状态表举例

① 在“地址”列输入存储器的地址（或符号名）。

② 在“格式”列选择数值的显示方式。如果操作数是位（例如，I、Q 或 M），格式中被设为位；如果操作数是字节、字或双字，浏览有效格式并选择适当的格式。定时器或计数器数值可以显示为位或字；如果将定时器或计数器地址格式设置为位，则会显示输出状态（输出打开或关闭）；如果将定时器或计数器地址格式设置为字，则使用当前值。

还可以按下面的方法更快地建立状态表。选中程序代码建立状态表如图 3-24 所示。

选中程序代码的一部分，单击鼠标右键，从弹出菜单命令中选择“创建状态表”。新状态表包含选中程序中每个操作数的一个条目。

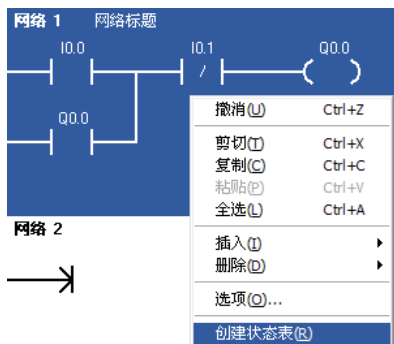


图 3-24 选中程序代码建立状态表

2) 编辑状态表。在状态表修改过程中,可采用下列方法。

① 插入新行。使用“编辑”菜单或用鼠标右键单击状态表中的一个单元格,从弹出菜单中选择“插入”→“行”命令。


② 删除一个单元格或行。选中单元格或行,用鼠标右键单击,从弹出菜单命令中选择“删除”→“选项”。

3) 建立多个状态表。用下面方法可以建立一个新状态表。


① 从指令树中,用鼠标右键单击“状态表”文件夹→弹出菜单命令→“插入”→“状态表”。

② 打开状态表窗口,使用“编辑”菜单或用鼠标右键单击,在弹出菜单中选择“插入”→“状态表”命令。

3. 状态表的启动与监视

1) 状态表启动和关闭。用菜单命令“调试”→“开始状态表监控”或使用工具条中的“状态表监控”按钮;再操作一次可关闭状态表。状态表启动后,便不能再编辑状态表。

2) 单次读取与连续状态表监控。状态表被关闭时(未启动),可以使用“单次读取”功能,方法如下。

① 用菜单命令“调试”→“单次读取”或使用工具条中的“单次读取”按钮。

单次读取可以从可编程序控制器收集当前的数据,并在表中当前值列显示出来,且在执行用户程序时并不对其更新。

状态表被启动后,使用“状态表监控”功能,将连续收集状态表信息。

② 用菜单命令“调试”→“状态表监控”或使用“状态表监控”工具条中的按钮。

3) 写入与强制数值。

① 全部写入。对状态表内的新数值改动完成后,可利用全部写入将所有改动传送至可编程序控制器中。物理输入点不能用此功能改动。

② 强制。在状态表的地址列中选中一个操作数,在新数值列写入模拟实际条件的数值,然后单击工具条中的“强制”按钮。一旦使用“强制”,每次扫描就都会将强制数值应用于该地址,直至对该地址“取消强制”为止。

③ 取消强制。与“程序状态”的操作方法相同。

3.4.4 执行有限次扫描

可以指定 PLC 对程序执行有限次数扫描(1~65 535 次扫描),通过指定 PLC 运行的扫描次数,可以监控程序过程变量的改变。在进行第一次扫描时,SM0.1 数值为 1。

1. 执行单次扫描

“单次扫描”使 PLC 从 STOP 转变成 RUN,执行单次扫描,然后再转回 STOP,因此与第一次相关的状态信息不会消失。操作步骤如下。

1) PLC 必须位于 STOP(停止)模式。如果不在 STOP(停止)模式,就将 PLC 转换成停止模式。

2) 用菜单“调试”→“首次扫描”命令。

2. 执行多次扫描

步骤如下。

1) PLC 须位于 STOP (停止) 模式。如果不在 STOP (停止) 模式, 就将 PLC 转换成停止模式。

2) 用菜单命令“调试”→“多次扫描”→出现“执行扫描”对话框, 如图 3-25 所示。

3) 输入所需的扫描次数数值, 单击“确定”按钮。



图 3-25 “执行扫描”对话框

3.5 编程软件实训

1. 实训目的

- 1) 认识 S7-200 系列可程序控制器及其与 PC 的通信。
- 2) 练习使用 STEP 7-Micro/Win V4.0 编程软件。
- 3) 学会程序的输入和编辑方法。
- 4) 初步了解程序调试的方法。

2. 实训内容及指导

1) PLC 认识。记录所使用 PLC 的型号、输入/输出点数, 观察主机面板的结构以及 PLC 和 PC 之间的连接。

2) 开机 (打开 PC 和 PLC) 并新建一个项目。用菜单命令“文件”→“新建”或用新建项目快捷按钮。

3) 检查 PLC 和运行 STEP7-Micro/Win 的 PC 连线后, 设置与读取 PLC 的型号。菜单命令“PLC”→“类型”→“读取 PLC”或者在指令树→“项目”名称→“类型”→“读取 PLC”。

4) 选择指令集和编辑器。

① 菜单命令“工具”→“选项”→“常规”选项卡→“编程模式”→SIMATIC; “助记符集”→“国际”。

② 用菜单命令“查看”→“LAD”。或者用菜单命令“工具”→“选项”→“一般”选项卡→“梯形图编辑器”。

5) 输入、编辑图 3-26 所示的梯形图, 并将其转换成语句表指令。

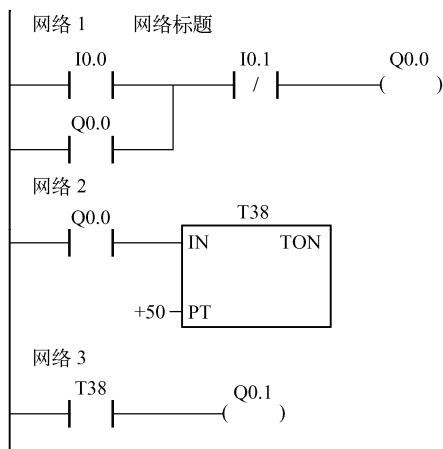



图 3-26 梯形图

6) 给梯形图加程序注释、网络标题、网络注释。

7) 编写符号表, 如图 3-27 所示。并选择操作数显示形式为: 符号和地址同时显示。


	符号	地址	注释
1	启动按钮	I0.0	
2	停止按钮	I0.1	
3	灯1	Q0.0	
4	灯2	Q0.1	
5			

图 3-27 符号表

① 建立符号表。单击浏览条中的“符号表”按钮。

② 符号和地址同时显示。“工具”→“选项”→“程序编辑器”。

8) 编译程序。并观察编译结果, 若提示错误, 则修改, 直到编译成功。

“PLC”→“编译”、“全部编译”或用快捷按钮 。


9) 将程序下载到 PLC。下载之前, PLC 必须位于“停止”的工作方式。如果 PLC 没有在“停止”, 就单击工具条中的“停止”按钮, 将 PLC 至于停止方式。


单击工具条中的“下载”按钮, 或用菜单命令“文件”→“下载”, 出现“下载”对话框。可选择是否下载“程序块”、“数据块”和“系统块”, 单击“确定”按钮, 开始下载程序。

10) 建立状态表表监视各元器件的状态, 状态图表如图 3-28 所示。选中程序代码的一部分, 单击鼠标右键→弹出菜单→“建立状态表”。


	地址	格式	当前值	新数值
1	I0.0	位		
2	I0.1	位		
3	Q0.0	位		
4	Q0.1	位		
5	T38	位		

图 3-28 状态图表

11) 运行程序。单击工具栏中的“运行”按钮。

12) 启动“状态表监控”。用菜单命令“调试”→“状态表监控”或使用“状态表监控”工具条按钮。

13) 输入强制操作。因为不带负载进行运行调试, 所以采用强制功能模拟物理条件。对 I0.0 进行强制 ON, 在对应 I0.0 的新数值列输入 1, 对 I0.1 进行强制 OFF, 在对应 I0.1 的新数值列输入 0。然后单击工具条中的“强制”按钮。

14) 在运行中显示梯形图的程序状态。单击“程序状态打开/关闭”按钮或用菜单命令“调试”→“程序状态”, 在梯形图中显示出各元器件的状态。

3. 记录结果

1) 认真观察 PLC 基本单元上的输入/输出指示灯的变化, 并记录结果。

- 2) 总结梯形图输入及修改的操作过程。
- 3) 写出梯形图添加注释的过程。

3.6 习题

1. 如何建立项目？
2. 如何在 LAD 中输入程序注解？
3. 如何下载程序？
4. 如何在程序编辑器中显示程序状态？
5. 如何建立状态表？
6. 如何执行有限次数扫描？

机械工业出版社版权所有，不得转载！