

电气与控制实验科学技术

王科俊 主编

可编程控制器实验技术

KE BIAN CHENG KONG ZHI QI SHI YAN JI SHU

孙 蓉 李 冰 编著

黑龙江人民出版社

· 电气与控制实验科学技术 · 王科俊 主编

江南大学图书馆



91142454

可编程控制器实验技术

孙 蓉 李 冰 编著



黑龙江人民出版社

图书在版编目(CIP)数据

电气与控制实验科学技术/王科俊 主编. —哈尔滨:黑龙江人民出版社,2008.7
ISBN 978-7-207-07885-8

I. 电… II. 王… III. 电气控制—实验—研究 IV. TM921.5

中国版本图书馆 CIP 数据字(2008)第 110906 号

责任编辑:李荣焕

装帧设计:李若聃

可编程控制器实验技术

孙 蓉 李 冰 编著

出版发行 黑龙江人民出版社
通讯地址 哈尔滨市南岗区宣庆小区 1 号楼
邮 编 150008
网 址 www.longpress.com
电子邮箱 hljrmcbs@yeah.net
印 刷 哈尔滨太平洋彩印有限公司
开 本 16 开
印 张 135 印张
字 数 355 万字
印 数 2000 套
版 次 2008 年 8 月第 1 版 2008 年 8 月第 1 次印刷
书 号 ISBN 978-7-207-07885-8/TP·14
定 价 350.00 元(全套 14 本)

(如发现本书有印制质量问题,印刷厂负责调换)

本社常年法律顾问:北京市大成律师事务所哈尔滨分所律师赵学利、赵景波

前言

可编程控制器(PLC)是应用十分广泛的微机控制装置,是自动控制系统的关键设备。专为工业现场应用而设计的,它采用可编程序的存储器,用来在其内部存储执行逻辑运算、顺序控制、定时/计数和算术运算等操作的指令,并通过数字式或模拟式的输入和输出,控制各种类型的机械或生产过程。目前 PLC 已广泛应用于冶金、矿业、机械、轻工等领域,为工业自动化提供了有力的工具;为此,各高校的电气自动化、机电一体化等相关专业相继开设了有关可编程控制器原理及应用的课程。可编程控制器课程是一门实践性很强的课程,要学好可编程控制器,除了在课堂上的书本中作基本的传授外,通过实验手段进行自动控制系统的模拟设计与程序调试,进一步验证、巩固和深化控制器原理知识与软硬件设计知识是必不可少的;通过实验还可以加强对常见工控设备的认识 and 了解。

本书就是基于这样一个出发点,以目前用得较普遍的西门子 S7-200/300 中小型 PLC 为实训样机,结合清华科教开发制作的材料分拣教学模型、五层电梯教学模型,从工程实践出发,由易到难,循序渐进,在典型应用的基础上,逐步解决实际问题。

本书共分两部分,第一部分为基础知识,从可编程控制器的基础知识、S7-200/300 的编程语言与常用指令、编程软件的使用方法、梯形图编程规则和基本编程方法四个方面简要介绍了可编程控制器的基础理论,第二部分为实例,以材料分拣教学模型控制系统、电梯教学模型控制系统二个 PLC 控制系统为主线,帮助读者深入了解 PLC 控制系统。

本书中,孙蓉撰写了第 1 章、第 3 章和第 4 章,李冰编写了第 2 章和第 5 章。

本书的编写得到了哈尔滨工程大学自动化学院控制工程实验教学中心的资助,控制工程实验教学中心的吕淑萍教授、彭秀艳教授、于占东副教授和清华科教的薛磊工程师、陈凯工程师对本书给予了大力支持,在此,笔者表示深切的谢意;本书参考、引用了一些文献资料,在本书问世之际,向这些文献资料的作者表示衷心的感谢。

因作者水平有限,书中难免有错漏之处,恳请读者批评指正。

内 容 简 介

本书作为可编程控制器(PLC)及其应用技术的入门读物,从一般性原理和方法出发,系统地介绍了可编程控制器应用软件设计的基本方法和一般步骤,并结合实例详细地介绍了西门子 S7-200/300PLC 程序设计的方法和技巧。

本书包括可编程控制器的基础知识、编程语言与常用指令、编程软件的使用方法和梯形图编程规则、基本编程方法和应用实例等内容。

本书可供开发应用 PLC 的工程技术人员使用,也可以作为大中专院校电子、机电、自动化类专业学生实践教学的参考书。

目 录

第 1 章 可编程控制器基础知识.....	(1)
1.1 可编程控制器基本概念和工作原理	(1)
1.2 S7-200/300 硬件结构与性能指标	(8)
1.3 计算机通信网络及 S7-200/300 通信功能	(16)
第 2 章 S7-200/300 编程语言与常用指令	(22)
2.1 S7-200/300 编程语言和存储区	(22)
2.2 位逻辑指令	(29)
2.3 定时器与计数器指令	(41)
2.4 传送指令	(60)
第 3 章 编程软件使用方法	(62)
3.1 编程软件简介	(62)
3.2 硬件组态和编程功能	(69)
第 4 章 梯形图编程规则和基本编程方法	(83)
4.1 梯形图编程规则	(83)
4.2 逻辑控制系统梯形图设计方法	(85)
第 5 章 应用实例	(110)
5.1 材料分拣教学模型控制系统实例	(110)
5.2 电梯教学模型控制系统实例	(122)

第1章 可编程控制器基础知识

随着微处理器、计算机和数字通信技术的飞速发展,计算机控制已经广泛地应用在不同的工业领域。现代社会要求制造业对市场需求作出迅速的反应,生产出小批量、多品种、多规格、低成本和高品质的产品。为了满足这一要求,生产设备和自动生产线的控制系统必须具有极高的可靠性和灵活性。可编程控制器正是顺应这一要求出现的,它是以微处理器为基础的通用工业控制装置,已经成为当代工业自动化的主要支柱之一。

1.1 可编程控制器基本概念和工作原理

1.1.1 可编程控制器的基本概念

1. 可编程控制器的产生

在20世纪60年代,汽车生产流水线的自动控制系统基本上都是由继电器控制装置构成的。当时汽车的每一次改型都直接导致继电器控制装置的重新设计和安装。随着生产的发展,汽车型号更新的周期愈来愈短,这样,继电器控制装置就需要经常地重新设计和安装,十分费时、费工、费料,甚至阻碍了更新周期的缩短。为了改变这一现状,美国通用汽车公司在1969年公开招标,要求用新的控制装置取代继电器控制装置,并提出了10项招标指标,即:

- (1) 编程方便,现场可修改程序;
- (2) 维修方便,采用模块化结构;
- (3) 可靠性高于继电器控制装置;
- (4) 体积小于继电器控制装置;
- (5) 数据可直接送入管理计算机;
- (6) 成本可与继电器控制装置竞争;
- (7) 输入可以是交流115V;
- (8) 输出为交流115V,2A以上,能直接驱动电磁阀、接触器等;
- (9) 在扩展时,原系统只要很小变化;
- (10) 用户程序存储器容量至少能扩展到4KB。

1969年,美国数字设备公司(DEC)研制出第一台PLC(可编程控制器),在美国通用汽车自动装配线上试用,获得了成功。这种新型的工业控制装置以其简单易懂、操作方便、可靠性高、通用灵活、体积小、使用寿命长等一系列优点,很快地在美国其他工业领域推广应用。到1971年,已经成功地应用于食品、饮料、冶金、造纸等工业。

这一新型工业控制装置的出现,也受到了世界其他国家的高度重视。1971年,日本从美国引进了这项新技术,很快研制出了日本第一台PLC。1973年,西欧国家也研制出它们的第一台PLC。我国从1974年开始研制,1977年开始应用于工业。

2. 可编程控制器的发展

早期的PLC一般称为可编程逻辑控制器。这时的PLC多少有点继电器控制装置的替代物的含义,其主要功能只是执行原先由继电器完成的顺序控制、定时等。它在硬件上以准计算机的形式出现,在I/O接口电路上做了改进以适应工业控制现场的要求。装置中的器件主要采用分立元件和小规模集成电路,存储器采用磁芯存储器。另外,还采取了一些措施,以提高其抗干扰的能力。在软件编程上,采用广大电气工程技术人员所熟悉的继电器控制线路的方式——梯形图。因此,早期的

PLC 的性能要优于继电器控制装置,其优点是简单易懂、便于安装、体积小、能耗低、有故障显示、能重复使用等。其中 PLC 特有的编程语言——梯形图一直沿用至今。

20 世纪 70 年代,微处理器的出现使 PLC 发生了巨大的变化。美国、日本、德国等一些厂家先后开始采用微处理器作为 PLC 的中央处理单元(CPU),这样使 PLC 的功能大大增强。在软件方面,除了保持其原有的逻辑运算、计时、计数等功能以外,还增加了算术运算、数据处理和传送、通信、自诊断等功能。在硬件方面,除了保持其原有的开关模块以外,还增加了模拟量模块、远程 I/O 模块、各种特殊功能模块并扩大了存储器的容量,使各种逻辑线圈的数量增加。除此以外,还提供了一定数量的数据寄存器。

进入 20 世纪 80 年代中、后期,由于超大规模集成电路技术的迅速发展,微处理器的市场价格大幅度下跌,使得各种类型的 PLC 所采用的微处理器的档次普遍提高。而且,为了进一步提高 PLC 的处理速度,各制造厂商还纷纷研制开发出专用逻辑处理芯片,这样使得 PLC 软、硬件功能发生了巨大变化。

3. 可编程控制器的概念

可编程控制器(Programmable Controller)本来简称为 PC,为了与个人计算机(Personal Computer)的简称 PC 相区别,将它简称为 PLC(Programmable Logical Controller)。

国际电工委员会(IEC)在 1985 年的可编程控制器标准草案第 3 稿中,对可编程控制器作了如下定义:“可编程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。它采用可编程序的存储器,用来在其内部存储执行逻辑运算、顺序控制、定时、计数和算术运算等操作的指令,并通过数字式、模拟式的输入和输出,控制各种类型的机械或生产过程。可编程控制器及其有关设备,都应按易于使工业控制系统形成一个整体,易于扩充其功能的原则设计。”从上述定义看出,可编程控制器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。

4. 可编程控制器的特点

(1) 编程方法简单易学

梯形图是使用得最多的 PLC 的编程语言,其电路符号和表达方式与继电器电路原理图相似,梯形图语言形象直观,易学易用,熟悉继电器电路图的电气技术人员只需花几天时间就可以熟悉梯形图语言,并用来编制用户程序。

(2) 功能强,性能价格比高

一台小型 PLC 内有成百上千个可供用户使用的编程元件,可以实现非常复杂的控制功能。与相同功能的继电器系统相比,具有很高的性能价格比。PLC 可以通过通信联网,实现分散控制,集中管理。

(3) 硬件配套齐全,用户使用方便,适应性强

PLC 产品已经标准化、系列化、模块化,配备有品种齐全的硬件装置供用户选用,用户能灵活方便地进行系统配置,组成不同功能、不同规模的系统。PLC 的安装接线也很方便,一般用接线端子连接外部接线。当控制要求改变,需要变更控制系统的功能时,只要改变存储器中的控制程序即可。PLC 的输入、输出可直接与交流 220V、直流 24V 等强电相连,并具有较强的带负载能力可以直接驱动一般的电磁阀和中小型交流接触器。

(4) 可靠性高,抗干扰能力强

PLC 是专为工业控制设计的,能适应工业现场的恶劣环境。绝大多数用户都将可靠性作为选取控制装置的首要条件,因此,PLC 在硬件和软件方面均采取了一系列的抗干扰措施。PLC 使用了一系列硬件和软件抗干扰措施,具有很强的抗干扰能力,平均无故障时间通常在 20000 小时以上,可以直接用于有强烈干扰的工业生产现场,PLC 已被广大用户公认为最可靠的工业控制设备之一。

在硬件方面,PLC 采取的抗干扰措施主要是隔离和滤波技术。PLC 的输入和输出电路一般都用

光电耦合器传递信号,使 CPU 与外部电路完全切断电的联系,有效地抑制外部干扰源对 PLC 的影响。在 PLC 的电源电路和 I/O 接口中,还设置了多种滤波电路,以抑制高频干扰信号。在软件方面,PLC 设置了故障检测及自诊断程序用来检测系统硬件是否正常,用户程序是否正确,便于自动地做出相应的处理,如报警、封锁输出、保护数据等。PLC 还用软件代替继电器控制系统中大量的中间继电器和时间继电器,接线可减少到继电器控制系统的十分之一以下,大大减少了因触点接触不良造成的故障。

(5) 系统的设计、安装、调试工作量少

用 PLC 完成一项控制工程时,由于其硬、软件齐全,设计和施工可同时进行。由于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,实现控制功能,使控制柜的设计、安装、接线工作量大大减少,缩短了施工周期。PLC 的梯形图程序可以用顺序控制设计法来设计。这种设计方法很有规律,很容易掌握。用这种方法设计梯形图的时间比设计继电器系统电路图的时间要少得多。同时,可以先在实验室模拟调试 PLC 的用户程序,用小开关来模拟输入信号,通过各输出点对应的发光二极管的状态来观察输出信号的状态,然后再将 PLC 控制系统在生产现场进行联机调试,使得调试方便、快速、安全,因此大大缩短了设计和投运周期。系统的调试时间比继电器系统少得多。

(6) 维修工作量小,维修方便

PLC 的控制程序可通过其专用的编程器输入到 PLC 的用户程序存储器中。编程器不仅能对 PLC 控制程序进行写入、读出、检测、修改等操作,还能对 PLC 的工作进行监控,使得 PLC 的操作及维护都很方便。PLC 还具有很强的自诊断能力,能随时检查出自身的故障,并显示给操作人员,使操作人员能迅速检查、判断故障原因。由于 PLC 的故障率很低,并且有完善的诊断和显示能力,当 PLC 或外部的输入装置及执行机构发生故障时,如果是 PLC 本身的原因,在维修时只需要更换插入式模块及其他易损坏部件即可迅速地排除故障,既方便又减少影响生产的时间。

(7) 体积小,能耗低

PLC 控制系统与继电器控制系统相比,减少了大量的中间继电器和时间继电器,配线用量少,安装接线工时短,加上开关柜体积的缩小,因此可以节省大量的费用。

1.1.2 可编程控制器的逻辑运算

在数字量(或称开关量)控制系统中,变量仅有两种相反的工作状态,例如高电平和低电平、继电器线圈的通电和断电、触点的接通和断开,可以分别用逻辑代数中的 1 和 0 来表示这些状态,在波形图中,用高电平表示 1 状态,用低电平表示 0 状态。

使用 PLC 的梯形图都可以实现数字量的逻辑运算。图 1-1 的上面是 PLC 的梯形图,下面是对应的数字门电路。

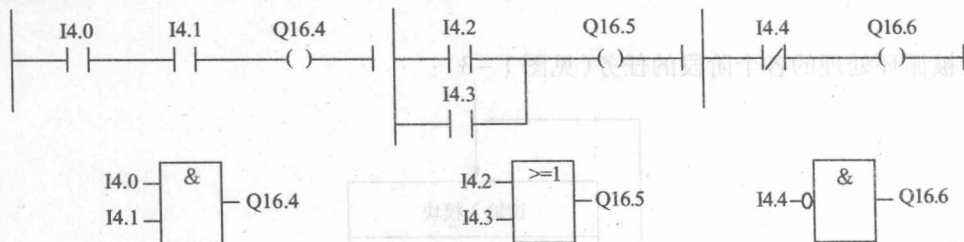


图 1-1 基本逻辑运算

图 1-1 中的 I4.0 ~ I4.4 为数字输入变量, Q16.4 ~ Q16.6 为数字输出变量,它们之间的“与”、“或”、“非”逻辑运算关系如表 1-1 所示,逻辑时序图如图 1-2 所示。用梯形图可以实现基本的逻辑运算,触点的串联可以实现“与”运算,触点的并联可以实现“或”运算,用常闭触点控制线圈可以实现“非”运算。多个触点的串、并联电路可以实现复杂的逻辑运算。

表 1-1 逻辑运算关系表

与			或			非	
$Q16.4 = I4.0 \cdot I4.1$			$Q16.5 = I4.2 + I4.3$			$Q16.6 = \overline{I4.4}$	
I4.0	I4.1	Q16.4	I4.2	I4.3	Q16.5	I4.4	Q16.6
0	0	0	0	0	0	0	1
0	1	0	0	1	1	1	0
1	0	0	1	0	1		
1	1	1	1	1	1		

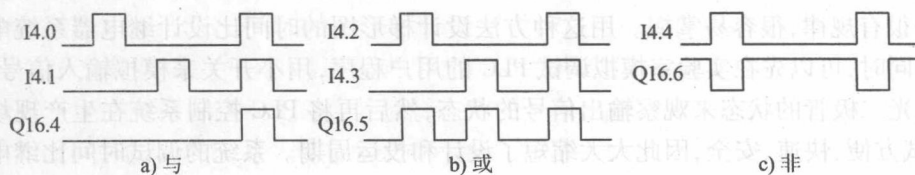


图 1-2 逻辑时序图

1.1.3 可编程控制器的循环处理过程

1. PLC 的循环处理过程

CPU 中的程序分为操作系统和用户程序。操作系统用来处理 PLC 的起动、刷新输入/输出过程映像区、调用用户程序、处理中断和错误、管理存储区和通信等任务。用户程序由用户生成,用来实现用户要求的自动化任务。程序创建软件 STEP 7 将用户编写的程序和程序所需的数据放置在块中,功能块 FB 和功能 FC 相当于用户编写的子程序,系统功能 SFC 和系统功能块 SFB 是操作系统提供给用户使用的标准子程序,这些块统称为逻辑块。

PLC 的 CPU 是以分时操作方式处理各项任务的。由于运算速度高,从 PLC 的外部输入、输出关系来看,处理过程几乎是瞬时完成的。PLC 的用户程序由若干条指令组成,CPU 从第一条指令开始,在无中断或跳转控制的情况下,按程序储存顺序的先后逐条执行用户程序,直到程序结束。然后,程序返回第一条指令开始新一轮的扫描,周而复始地重复上述的扫描循环。这种采用循环执行用户程序的方式也称为扫描工作方式,OB1 是用于循环处理的组织块,相当于用户程序中的主程序,它可以调用别的逻辑块,或被中断程序(组织块)中断。

PLC 得电或由 STOP 模式切换到 RUN 模式时,CPU 执行启动操作,清除没有保持功能的位存储器、定时器和计数器,清除中断堆栈和块堆栈的内容,复位保存的硬件中断等。以后将进入周期性的循环运行。

下面是被循环处理的各个阶段的任务(见图 1-3):

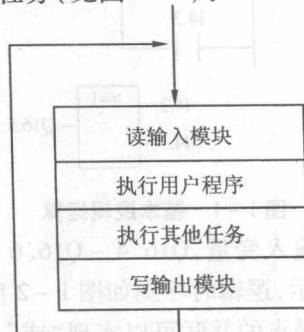


图 1-3 扫描过程

- (1) CPU 读输入模块的输入状态,并存入输入过程映像区;
- (2) CPU 处理用户程序,执行用户程序中的指令;
- (3) CPU 将输出过程映像区的数据写到输出模块;
- (4) CPU 返回第一阶段,开始新一轮的循环。

在启动完成后,不断地循环调用 OB1,在 OB1 中可以调用其他逻辑块(FB、SFB、FC 或 SFC)。OB1 具有很低的优先级,除了 OB90 外,所有的组织块都能中断 OB1。

循环程序处理过程可以被某些事件中断。如果有中断事件出现,当前正在执行的块被暂停执行,并调用分配给该事件的组织块。该组织块被执行完后,被暂停执行的块将从被中断的地方开始继续执行。

PLC 的扫描工作方式简单直观,简化了程序的设计,并为 PLC 的可靠运行提供了保证。当 PLC 扫描到的指令被执行后,其结果马上就可以被将要扫描到的指令所利用,而且还可以通过 CPU 内部设置的监视定时器来监视每次扫描是否超过规定时间,避免由于 CPU 内部故障使程序执行进入死循环。

2. 扫描周期

一个循环扫描过程称为扫描周期。扫描过程分为三个阶段进行:即输入采样(输入处理)阶段,程序执行(程序处理)阶段,输出刷新(输出处理)阶段,见图 1-4。

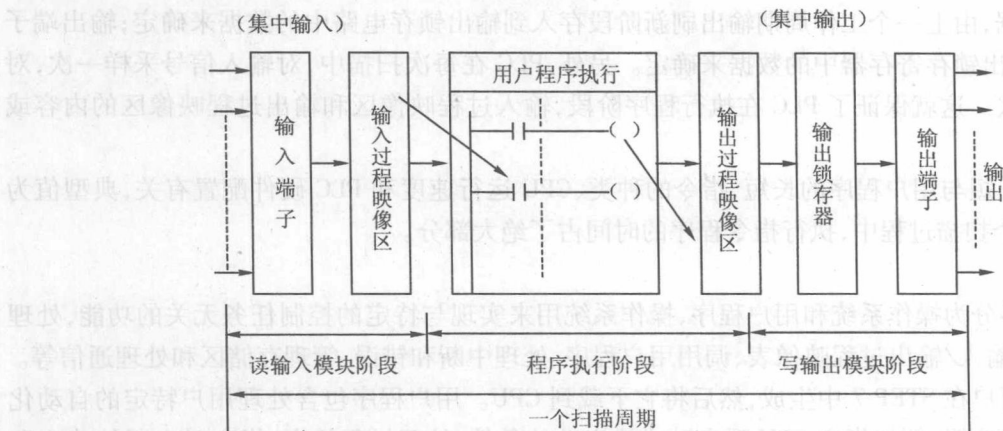


图 1-4 扫描周期的三个阶段

在循环程序处理过程中,CPU 并不直接访问 I/O 模块中的输入地址区和输出地址区,而是访问 CPU 内部的过程映像区。在 PLC 的存储器中,设置了一片区域用来存放输入信号和输出信号的状态,它们分别称为输入过程映像区和输出过程映像区。PLC 梯形图中的其他编程元件也有对应的映像存储区。

(1) 输入采样阶段

在输入采样阶段,PLC 以扫描工作方式按顺序将所有输入端的接通/断开状态读入到输入过程映像区中,称为对输入信号的采样,或称输入刷新,此时输入过程映像区被刷新。接着进入程序执行阶段,在程序执行阶段或其他阶段,即使输入状态发生变化,输入过程映像区的内容也不会改变,输入状态的变化只有在下一个扫描周期的输入采样阶段才被重新读入。

例如,外部输入电路接通时,对应的输入过程映像位为“1”状态,梯形图中对应的输入位的常开触点接通,常闭触点断开。外部输入触点电路断开时,对应的输入过程映像位为“0”状态,梯形图中对应的输入位的常开触点断开,常闭触点接通。

(2) 程序执行阶段

在程序执行阶段,PLC 对程序按顺序进行扫描执行。若程序用梯形图来表示,则总是按先上后下、先左后右的顺序进行。当遇到程序跳转指令时,则根据跳转条件是否满足来决定程序是否跳转。当指令中涉及输入、输出状态时,PLC 从输入过程映像区和输出过程映像区中读出,根据用户程序进

行运算,运算的结果再存入输出过程映像区中。对于输出过程映像区来说,其内容会随程序执行的过程而动态变化。

在程序执行阶段,即使外部输入信号的状态发生了变化,输入过程映像区的状态也不会随之而变,输入信号变化了的状态只能在下一个循环扫描周期的读输入模块阶段被读入。

(3) 输出刷新阶段

当所有程序执行完毕后,进入输出刷新阶段。在这一阶段里,PLC 将输出过程映像区的状态转存到输出,并通过一定方式输出,驱动外部负载,这就是 PLC 的实际输出。

例如,输出线圈“通电”时,对应的输出过程映像位为“1”状态。信号经输出模块隔离和功率放大后,继电器型输出模块中对应的硬件继电器的线圈通电,其常开触点闭合,使外部负载通电工作。

若梯形图中输出位的线圈“断电”,对应的输出过程映像位为“0”状态,在写输出模块阶段之后,继电器型输出模块中对应的硬件继电器的线圈断电,其常开触点断开,外部负载断电,停止工作。

PLC 重复执行上述三个过程,每重复一次的时间就是一个扫描周期。在一个扫描周期内,PLC 对输入状态的采样只在输入采样阶段进行,当 PLC 进入程序执行阶段后输入端将被封锁,直到下一个扫描周期的输入采样阶段才对输入状态进行重新采样。因此,输入过程映像区的数据,取决于输入端子在输入采样阶段所刷新的状态;输出过程映像区的状态,由程序中输出指令的执行结果决定;输出锁存寄存器中的数据,由上一个工作周期输出刷新阶段存入到输出锁存电路中的数据来确定;输出端子的输出状态,由输出锁存寄存器中的数据来确定。另外,PLC 在每次扫描中,对输入信号采样一次,对输出信号刷新一次。这就保证了 PLC 在执行程序阶段,输入过程映像区和输出过程映像区的内容或数据保持不变。

扫描周期的长短与用户程序的长短、指令的种类、CPU 运行速度和 PLC 硬件配置有关,典型值为 $1\text{ms} \sim 100\text{ms}$ 。一个扫描过程中,执行指令程序的时间占了绝大部分。

3. 块结构

PLC 中的程序分为操作系统和用户程序,操作系统用来实现与特定的控制任务无关的功能,处理 PLC 的启动、刷新输入/输出过程映像表、调用用户程序、处理中断和错误、管理存储区和处理通信等。

用户程序由用户在 STEP 7 中生成,然后将它下载到 CPU。用户程序包含处理用户特定的自动化任务所需要的所有功能,例如指定 CPU 暖启动或热启动的条件,处理过程数据,指定对中断的响应和处理程序正常运行中的干扰等。

STEP 7 将用户编写的程序和程序所需的数据放置在块中,使单个的程序部件标准化。通过在块内或块之间类似子程序的调用,使用户程序结构化,可以简化程序组织,使程序易于修改、查错和调试。块结构显著地增加了 PLC 程序的组织透明性、可理解性和易维护性。

(1) 组织块(OB)

组织块是操作系统与用户程序的接口,由操作系统调用,用于控制扫描循环和中断程序的执行、PLC 的启动和错误处理等,有的 CPU 只能使用部分组织块。

● OB1

OB1 的功能已在第 1 章中作了介绍,OB1 用于循环处理,是用户程序中的主程序。操作系统在每一次循环中调用一次组织块 OB1。一个循环周期分为输入、程序执行、输出和其他任务,例如下载、删除块、接收和发送全局数据等。

● 事件中断处理

如果出现一个中断事件,例如时间日期中断、硬件中断和错误处理中断等,当前正在执行的块在当前语句执行完后被停止执行(被中断),操作系统将会调用一个分配给该事件的组织块。该组织块执行完后,被中断的块将从断点处继续执行。

这意味着部分用户程序可以不必在每次循环中处理,而是在需要时才被及时地处理。用户程序可以分解为分布在不同的组织块中的“子程序”。如果用户程序是对一个重要事件的响应,并且这个

事件出现的次数相对较少,例如液位达到了最大上限,处理中断事件的程序放在该事件驱动的 OB 中。

● 中断的优先级

OB 按触发事件分成几个级别,这些级别有不同的优先级,高优先级的 OB 可以中断低优先级的 OB。当 OB 启动时,提供触发它的初始化启动事件的详细信息,这些信息可以在用户程序中使用。

(3) 功能(FC)

功能是用用户编写的没有固定的存储区的块,其临时变量存储在局域数据堆栈中,功能执行结束后,这些数据就丢失了。可以用共享数据区来存储那些在功能执行结束后需要保存的数据,不能为功能的局域数据分配初始值。

调用功能和功能块时用实参(实际参数)代替形参(形式参数),例如将实参“11.0”赋值给形参“Start”。形参是实参在逻辑块中的名称,功能不需要背景数据块。功能和功能块用输入(IN)、输出(OUT)和输入/输出(IN_OUT)参数做指针,指向调用它的逻辑块提供的实参。功能被调用后,可以为调用它的块提供一个数据类型为 RETURN 的返回值。

(4) 功能块(FB)

功能块是用用户编写的有自己的存储区(背景数据块)的块,每次调用功能块时需要提供各种类型的数据给功能块,功能块也要返回变量给调用它的块。这些数据以静态变量(STAT)的形式存放在指定的背景数据块(DI)中,临时变量存储在局域数据堆栈中。功能块执行完后,背景数据块中的数据不会丢失,但是不会保存局域数据堆栈中的数据。

在编写调用 FB 或系统功能块(SFB)的程序时,必须指定 DI 的编号,调用时 DI 被自动打开。在编译 FB 或 SFB 时自动生成背景数据块中的数据。可以在用户程序中或通过 HMI(人机接口)访问这些背景数据。

一个功能块可以有多个背景数据块,使功能块用于不同的被控对象。

可以在 FB 的变量声明表中给形参赋初值,它们被自动写入相应的背景数据块中。在调用块时,CPU 将实参分配给形参的值存储在 DI 中。如果调用块时没有提供实参,将使用上一次存储在背景数据块中的参数。

(5) 数据块(DB)

数据块(DB)是用于存放执行用户程序时所需的变量数据的数据区。与逻辑块不同,在数据块中没有 STEP 7 的指令,STEP 7 按数据生成的顺序自动地为数据块中的变量分配地址。数据块分为共享数据块和背景数据块。数据块的最大允许容量与 CPU 的型号有关。

数据块中基本的数据类型有 BOOL(二进制位),REAL(实数或浮点数)和 INTEGER(整数,简称为 INT)等。结构化数据类型(数组和结构)由基本数据类型组成。可以用符号表中定义的符号来代替数据块中的数据的地址,这样更便于程序的编写和阅读。

● 共享数据块(Share Block)

共享数据块存储的是全局数据,所有的 FB、FC 或 OB(统称为逻辑块)都可以从共享数据块中读取数据,或将数据写入共享数据块。CPU 可以同时打开一个共享数据块和一个背景数据块。如果某个逻辑块被调用,它可以使用它的临时局域数据区(即 L 堆栈)。逻辑块执行结束后,其局域数据区中的数据丢失,但是共享数据块中的数据不会被删除。

● 背景数据块(Instance Data Block)

背景数据块中的数据是自动生成的,它们是功能块的变量声明表中的数据(不包括临时变量 TEMP)。背景数据块用于传递参数,FB 的实参和静态数据存储在背景数据块中。调用功能块时,应同时指定背景数据块的编号或符号,背景数据块只能被指定的功能块访问。

应首先生成功能块,然后生成它的背景数据块。在生成背景数据块时,应指明它的类型为背景数据块(Instance),并指明它的功能块的编号,例如 FB2。

背景数据块的功能块被执行完后,背景数据块中存储的数据不会丢失。

(6) 系统功能块(SFB)

系统功能块和系统功能是为用户提供的已经编好程序的块,可以在用户程序中调用这些块,但是用户不能修改它们。它们作为操作系统的一部分,不占用程序空间。SFB 有存储功能,其变量保存在指定给它的背景数据块中。

(7) 系统功能(SFC)

系统功能是集成在 S7 CPU 的操作系统中预先编好程序的逻辑块,例如时间功能和块传送功能等。SFC 属于操作系统的一部分,可以在用户程序中调用。与 SFB 相比,SFC 没有存储功能。S7 CPU 提供以下的 SFC:复制及块功能,检查程序,处理时钟和运行时间计数器,数据传送,在多 CPU 模式的 CPU 之间传送事件,处理日期时间中断和延时中断,处理同步错误、中断错误和异步错误,有关静态和动态系统数据的信息,过程映像刷新和位域处理,模块寻址,分布式 I/O,全局数据通信,非组态连接的通信,生成与块相关的信息等。

(8) 系统数据块(SDB)

系统数据块是由 STEP 7 产生的程序存储区,包含系统组态数据,例如硬件模块参数和通信连接参数等用于 CPU 操作系统的数据。

1.2 S7-200/300 硬件结构与性能指标

西门子公司的 PLC 产品有 SIMATIC S7、M7 和 C7 等几大系列。S7 系列是传统意义的 PLC 产品,其中的 S7-200 是针对低性能要求的整体式小型 PLC,S7-300 是针对低性能要求的模块式中型 PLC。

S7-200 是在美国德州仪器公司的小型 PLC 的基础上发展起来的,其编程软件为 STEP7-Micro/IN 32。而 S7-300 的前身是西门子公司的 S5 系列 PLC,其编程软件为 STEP 7。所以,S7-200 和 S7-300 虽然有许多共同之处,但是在指令系统、程序结构和编程软件等方面均有相当大的差异。

1.2.1 S7-200 的硬件结构与性能指标

S7-200 是西门子自动化与驱动集团开发、生产的整体式小型 PLC。S7-200 PLC 除了能够进行传统的继电逻辑控制、计数和计时控制,还能进行复杂的数学运算、处理模拟量信号,并可支持多种协议和形式与其他智能设备进行数据通讯。

S7-200 的核心部件是 CPU(即中央处理单元),实际的工艺计算就在 CPU 中进行。开发人员必须根据实际的工艺要求,在软件开发环境中选择合适的指令,把它们编辑、组合成能够完成上述功能的程序并下载到 CPU 中执行。这些程序称为用户程序。

整体式 PLC 将 CPU 模块、I/O 模块和电源装在一个箱型机壳内,S7-200 称为 CPU 模块。图 1-5 中的前盖下面有 RUN/STOP 开关、模拟量电位器和扩展 I/O 连接器。S7-200 系列 PLC 提供多种具有不同 I/O 点数的 CPU 模块和数字量、模拟量 I/O 扩展模块供用户选用,CPU 模块和扩展模块用扁平电缆连接。

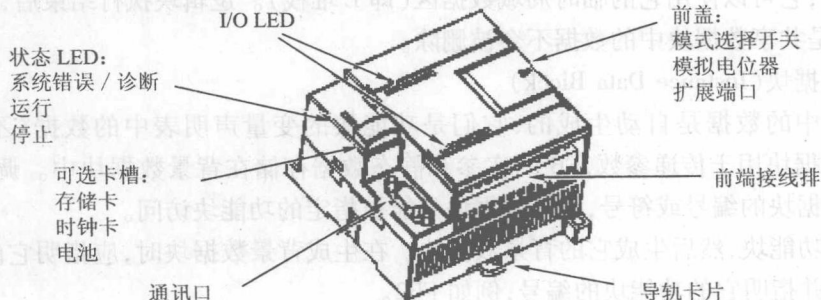


图 1-5 S7-200 的 CPU 外形

S7-200 的指令丰富,指令功能强,易于掌握,操作方便,内置有高速计数器、高速输出、PID 控制器、RS-485 通信/编程接口、PPI 通信协议、MPI 通信协议和自由方式通信功能。最多可以扩展到 248 点数字量 I/O 或 35 路模拟量 I/O,最多有 26 KB 的程序和数据存储空间。

1. S7-200 的 CPU 模块

S7-200 有 5 种 CPU 模块,本书中所介绍的是 CPU 224,它具有较强控制功能。CPU 224 XP 相比其他型号具有更高的硬件指标,如高达 100 kHz 的脉冲输出、100/200 kHz 的双相/单相高速脉冲输入,CPU 本体上的模拟量 I/O 等。它的 CPU 将一个微处理器、一个集成的电源和若干数字量 I/O 点集成在一个紧凑的封装中。

对于每个型号,西门子提供直流(24V)和交流(120~220V)两种电源供电的 CPU。如 CPU 224 DC/DC/DC 和 CPU 224 AC/DC/Relay。每个类型都有各自的订货号,可以单独订货。

- DC/DC/DC:说明 CPU 是直流供电,直流数字量输入,数字量输出点是晶体管直流电路的类型。
- AC/DC/Relay:说明 CPU 是交流供电,直流数字量输入,数字量输出点是继电器触点的类型。

2. S7-200 的扩展模块

西门子生产几种型号的 S7-200 CPU,每种 CPU 拥有不同的 I/O 点数和特殊功能,但是它们的核心处理芯片的运算能力相同。CPU 必须通过硬件取得实际过程信号和操作指令,这些硬件电路就是数字量(二进制的开关信号)和模拟量信号的输入点(通道);CPU 发出的控制指令也要通过硬件才能驱动系统中的执行机构,即数字量和模拟量信号的输出通道。输入/输出的硬件点(通道)称为 I/O (Input/Output)。如果控制系统需要更多的 I/O 点数,可以通过附加 I/O 扩展模块的方式实现。为此西门子生产了一系列输入/输出扩展模块。所有的扩展模块都是用自身配有的总线扩展电缆方便地连接到前面的 CPU 或其他扩展模块。为获得一些特殊功能或更多的通讯能力,S7-200 系统中还有很多通讯模块和工艺控制模块,用户可以根据自己的实际需要选用。

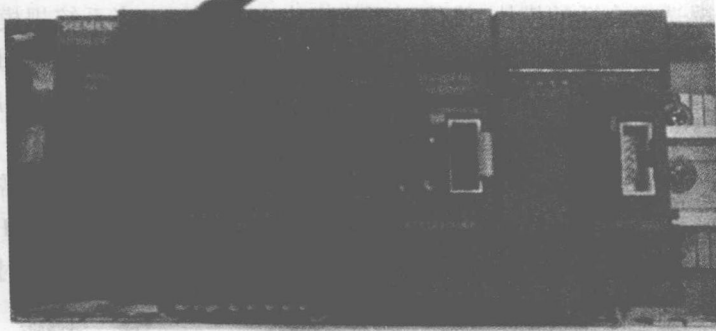


图 1-6 S7-200 的扩展模块

(1) S7-200 CPU 为了扩展 I/O 点和执行特殊的功能,可以连接扩展模块(CPU 221 除外)。扩展模块主要有如下几类:

- 数字量 I/O 模块
- 模拟量 I/O 模块
- 通讯模块
- 特殊功能模块

(2) EM 221:数字量输入扩展模块

EM 221 包括 3 种类型:

- 8 点 24 V DC 输入
- 8 点 120/230 V AC 输入
- 16 点 24 V DC 输入

(3) EM 222:数字量输出扩展模块

EM 222 包括 5 种类型:

- 8 点 24 V DC(晶体管)输出,每点 0.75 A
- 8 点继电器输出,每点 2 A
- 8 点 120/230 V AC 输出
- 4 点 24 V DC 输出,每点 5 A
- 4 点继电器输出,每点 10 A

(4) EM 223:数字量输入/输出扩展模块

EM 223 共有 6 种类型:

- 4 点 24 V DC 输入/4 点 24 V DC 输出
- 4 点 24 V DC 输入/4 点继电器输出
- 8 点 24 V DC 输入/8 点 24 V DC 输出
- 8 点 24 V DC 输入/8 点继电器输出
- 16 点 24 V DC 输入/16 点 24 V DC 输出

除了特别为 S7-200 系列开发设计的相关产品外,西门子还提供范围宽广、种类繁多的自动化与驱动类产品。作为西门子自动化与驱动产品系列的一个组成部分,S7-200 可以很容易地与西门子的其他系列产品搭配和联网,获得最优性能。

1.2.2 S7-300 的硬件结构与性能指标

大、中型 PLC(例如西门子的 S7-300 和 S7-400 系列)一般采用模块式结构,用搭积木的方式来组成系统,模块式 PLC 由机架和模块组成。S7-300(见图 1-7)是模块化的中型 PLC,适用于中等性能的控制要求。品种繁多的 CPU 模块、信号模块和功能模块能满足各种领域的自动控制任务,用户可以根据系统的具体情况选择合适的模块,维修时更换模块也很方便。当系统规模扩大和更为复杂时,可以增加模块,对 PLC 进行扩展。简单实用的分布式结构和强大的通信联网能力,使其应用十分灵活。

S7-300 的 CPU 模块(简称为 CPU)继承了过程控制功能,用于执行用户程序。S7-300 的每个 CPU 都有一个编程用的 RS-485 接口,使用西门子的 MPI(多点接口)通信协议。有的 CPU 还带有集成的现场总线 PROFIBUS-DP 接口或 PpP(点对点)串行通信接口。S7-300 不需要附加任何硬件、软件和编程,就可以建立一个 MPI 网络,通过 PROFIBUS-DP 接口可以建立一个 DP 网络。

功能最强的 CPU 的 RAM 存储容量为 512KB,有 8192 个存储器位,512 个定时器和 512 个计数器,数字量通道最大为 65536 点,模拟量通道最大为 4096 个。计数器的计数范围为 1~999,定时器的定时范围为 10ms~9990s。由于使用 Flash EPROM,CPU 断电后无需后备电池也可以长时间保持动态数据,使 S7-300 成为完全无维护的控制设备。

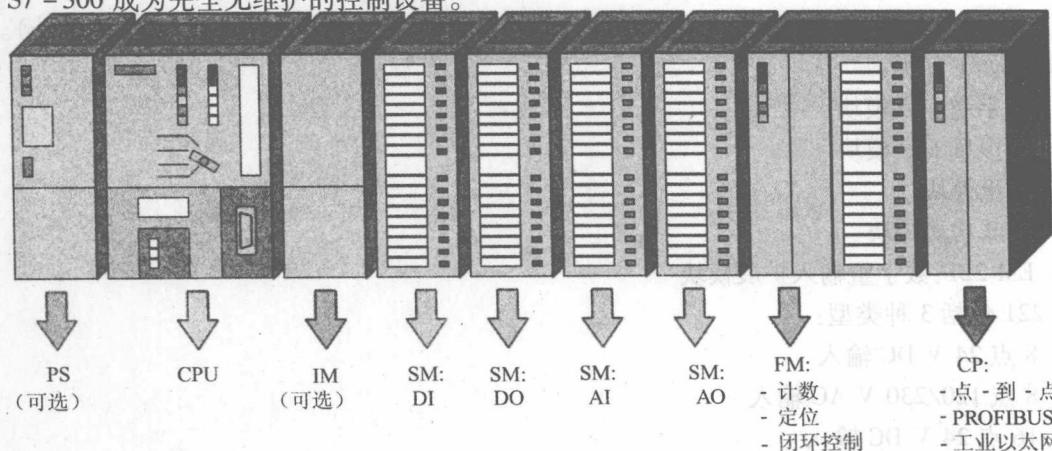


图 1-7 S7-300 PLC

S7-300 有很高的电磁兼容性和抗振动抗冲击能力。S7-300 标准型的环境温度为 $0 \sim 60^{\circ}\text{C}$ 。环境条件扩展型的温度范围为 $-25 \sim +60^{\circ}\text{C}$ ，有更强的耐振动和耐污染性能。

S7-300 是模块式的 PLC，采用紧凑的、无槽位限制的模块结构，S7-300 主要由机架、CPU 模块、信号模块、功能模块、接口模块、通信处理器、电源模块和编程设备组成（见图 1-8），各种模块安装在机架上。通过 CPU 模块或通信模块上的通信接口，PLC 被连接到通信网络上，可以与计算机、其他 PLC 或其他设备通信。

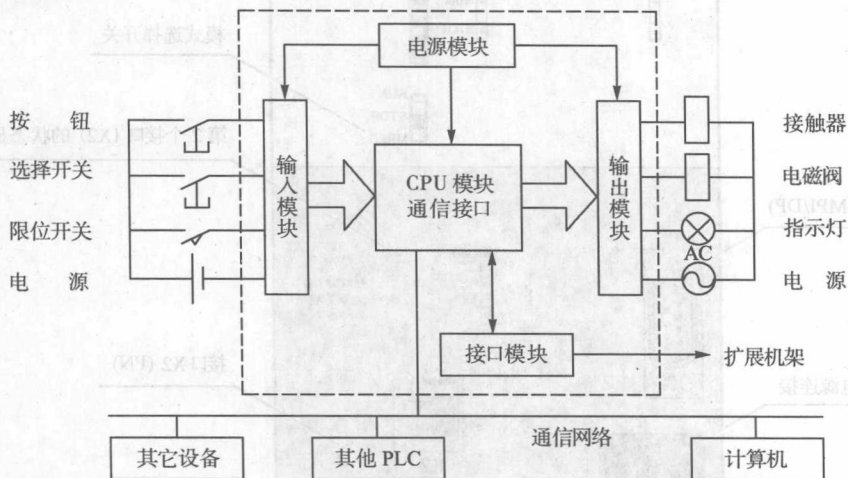


图 1-8 PLC 控制系统示意图

1. CPU 模块

CPU 模块主要由微处理器（CPU 芯片）和存储器组成，S7-300 将 CPU 模块简称为 CPU。在 PLC 控制系统中，CPU 模块相当于人的大脑和心脏，它不断地采集输入信号，执行用户程序，刷新系统的输出，模块中的存储器用来储存程序和数据。CPU 前面板上有状态故障指示灯、模式开关、24V 电源端子、电池盒与存储器模块盒。

CPU 内的元件封装在一个牢固而紧凑的塑料机壳内，面板上有状态和故障指示 LED、模式选择开关和通信接口。存储器插槽可以插入多达数兆字节的 Flash EPROM 微存储器卡（简称为 MMC），用于断电后程序和数据的保存。

图 1-9 是 CPU 315-2PN/DP 的面板图，有微存储器卡 MMC 才能运行，新面板横向的宽度只是原来的一半。此型号 CPU 没有集成的输入/输出模块。

(1) 状态与故障显示 LED

CPU 模块面板上的 LED（发光二极管）的意义如下：

- SF（系统出错/故障显示，红色）：CPU 硬件故障或软件错误时亮
- BATF（电池故障，红色）：电池电压低或没有电池时亮
- DC 5V（+5V 电源指示，绿色）：CPU 和 S7 300 总线的 5V 电源正常时亮
- FRCE（强制，黄色）：至少有一个 I/O 被强制时亮
- STOP（停止方式，黄色）：CPU 处于 STOP、HOLD 状态或重新启动时常亮；执行存储器复位时闪亮。
- BUSF（总线错误，红色）：PROFIBUS-DP 接口硬件或软件故障时亮，集成有 DP 接口的 CPU 才有此 LED。集成有两个 DP 接口的 CPU 有两个对应的 LED（BUS1F 和 BUS2F）。

(2) CPU 的运行模式

CPU 有 4 种操作模式：STOP（停机）、STARTUP（启动）、RUN（运行）和 HOLD（保持）。在所有的模式中，都可以通过 MPI 接口与其他设备通信。