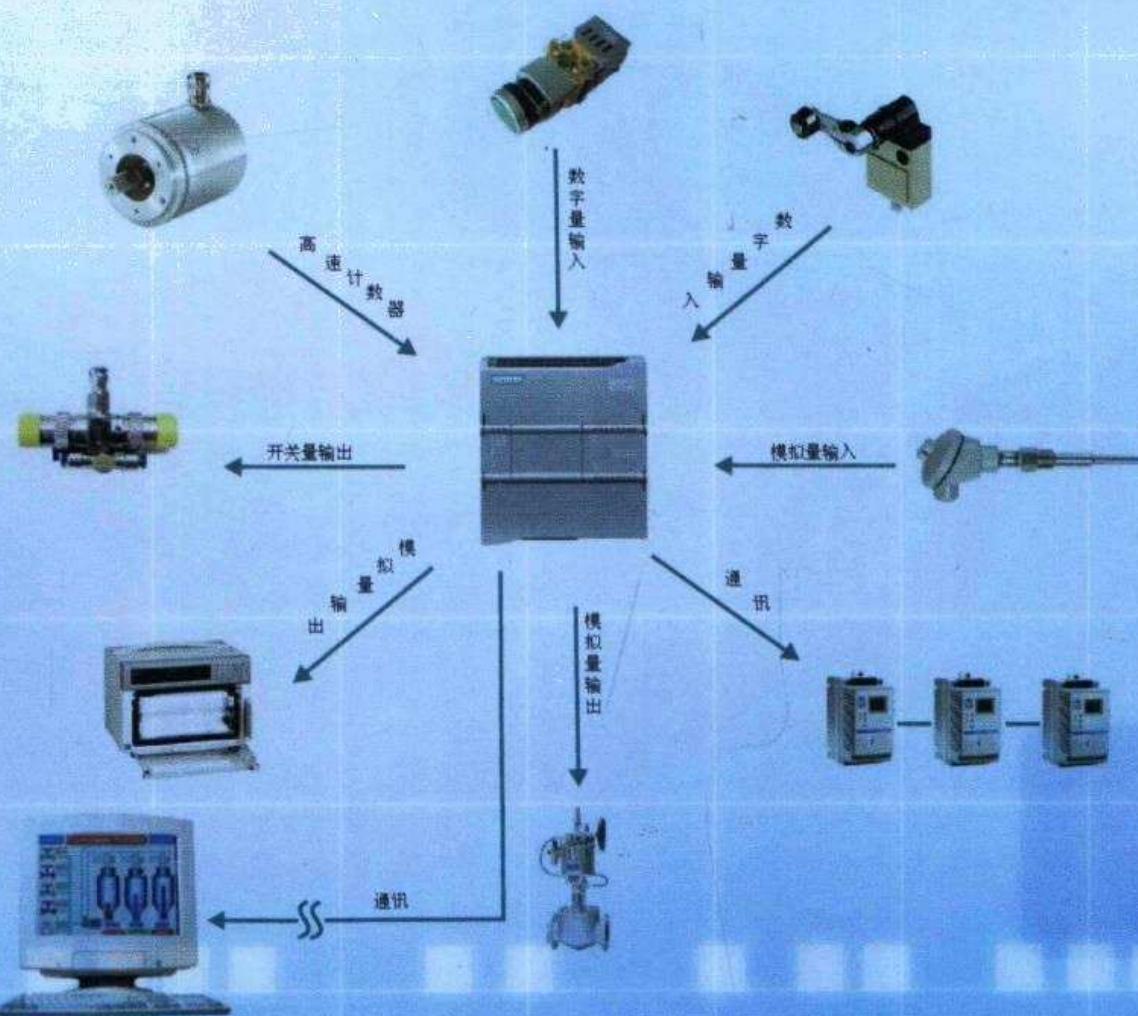


图解西门子

S7-1200 PLC入门到实践

李方园 编著





ISBN 978-7-111-31816-3

策划编辑：林春泉

封面设计：路恩中

上架指导：工业技术／自动化技术

地址：北京市百万庄大街22号

电话服务

社服务中心：(010)88361066

销售一部：(010)68326294

销售二部：(010)88379649

读者服务部：(010)68993821

邮政编码：100037

网络服务

门户网：<http://www.cmpbook.com>教材网：<http://www.cmpedu.com>

封面无防伪标均为盗版

ISBN 978-7-111-31816-3

定价：50.00元



9 787111 318163 >

图解西门子S7-1200 PLC

入门到实践

李方园 编著



本书是西门子公司推出的创新系统解决方案——模块化控制器 SIMATIC S7-1200 PLC 项目工具书，本书不仅可以锻炼读者的编程技巧，更是以创新性的写作方式叙述了项目从简单到复杂、从入门到实践的过程。全书共分为 9 章，涵盖了 S7-1200 PLC 应用的大部分场合，包括电动机控制、生产线流程控制、TIA 集成控制、PID、PWM、HSC、运动控制、串口通信和以太网 OPC 组态。这些案例通过在 OEM 用户近一年的使用，已经具有推广的价值，通过作者创造性地归纳和总结，使读者只需要一个 CPU1214C 和 KTP 触摸屏就能完整地进行模拟和使用书中介绍的所有项目。

本书深入浅出、图文并茂，既适合自动化技术人员作为工程应用案例，也适合高职院校的电气自动化、机电一体化、应用电子技术等相关专业的学生作为教材使用。

免费提供全部源程序，请登录机械工业出版社门户网：<http://www.cmpbook.com>。

图书在版编目(CIP)数据

图解西门子 S7-1200 PLC 入门到实践 / 李方园编著. —北京：机械工业出版社，2010.9

ISBN 978 - 7 - 111 - 31816 - 3

I. ①图… II. ①李… III. ①可编程序控制器—基本知识
IV. ①TM571.6

中国版本图书馆 CIP 数据核字 (2010) 第 177707 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策划编辑：林春泉 责任编辑：赵 任

版式设计：霍永明 责任校对：程俊巧

封面设计：路恩中 责任印制：杨 曜

北京蓝海印刷有限公司印刷

2011 年 1 月第 1 版第 1 次印刷

184mm × 260mm · 22.5 印张 · 554 千字

0001—3000 册

标准书号：ISBN 978 - 7 - 111 - 31816 - 3

定价：50.00 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

门户网：<http://www.cmpbook.com>

销售一部：(010) 68326294

教材网：<http://www.cmpedu.com>

销售二部：(010) 88379649

封面无防伪标均为盗版

读者服务部：(010) 68993821



前言

S7-1200 PLC 产品推向市场已有一年的时间，为了使广大读者尽快地掌握新产品的使用技巧，进而增强对新产品和新技术的应用能力，作者通过近一年时间在工程上应用该产品的经验，推出了《图解西门子 S7-1200 PLC 入门到实践》一书。本书涵盖了 S7-1200 PLC 应用的大部分场合，包括电动机控制、生产线流程控制、TIA 集成控制、PID、PWM、HSC、运动控制、串口通信和以太网 OPC 组态。

同时，本书进一步阐述了 S7-1200 PLC 产品的亮点及如何使用 STEP 7 Basic 软件对其进行编程、S7-1200 PLC 硬件组态及网络配置、S7-1200 PLC 的通信、如何使用 PLC 实现运动控制等。

本书从西门子 S7-1200 PLC 初学者的角度出发，按照项目导入、任务驱动的原则对 9 章内容进行讲解，并对每章中的项目给出了具体的硬件接线、程序清单与注释。

本书通俗易懂，对于每一个项目，都从项目背景、知识讲座、技能实训到最后的工程案例一一展开。因此，通过本书的学习，不仅能了解一般电气自动化控制系统的设计过程、设计要求、应完成的工作内容和具体设计方法，同时也有助于复习、巩固以往所学的 PLC 知识，达到在工程设计中灵活应用的目的。

在本书的编写过程中，不仅得到了张永惠教授的大力支持，而且得到了西门子（中国）有限公司、宁波中华纸业有限公司、宁波钢铁有限公司、常州米高电子科技有限公司等厂家相关人员的帮助，提供了相当多的典型案例和维护经验。陈亚玲、叶明、陈贤富、沈阿宝、陈亚珠、李伟庄、章富科、方定桂、刘军毅、戴琴、王永行、刘伟红等参与了编写工作。同时，在本书的编写过程中参考和引用了国内外许多专家、学者最新发表的论文和著作等资料，作者在此一并致谢。

希望本书能成为广大 S7-1200 PLC 用户的实用工具书，帮助大家全面、快速地掌握 SIMATIC S7-1200 PLC 的应用，并且能为众多的工业用户提供有力的支持和有效的解决方案，同时也为工业 PLC 的进一步发展做出一份贡献。

作者
2010 年 7 月 8 日

目 录

前言

| | |
|---------------------------------|----|
| 第1章 S7-1200 PLC入门 | 1 |
| 1.1 背景知识介绍：PLC的产生与原理 | 2 |
| 1.1.1 PLC的产生 | 2 |
| 1.1.2 PLC的控制原理 | 2 |
| 1.1.3 PLC的定义 | 4 |
| 1.1.4 PLC开关量控制的应用 | 4 |
| 1.2 知识讲座：S7-1200 PLC的组成 | 5 |
| 1.2.1 S7-1200 PLC的硬件组成 | 5 |
| 1.2.2 以S7-1200 PLC为核心的控制系统 | 7 |
| 1.2.3 IEC 61131-3 标准 | 8 |
| 1.3 技能训练一：STEP 7 Basic软件的安装 | 11 |
| 1.3.1 STEP 7 Basic V10.5 软件 | 11 |
| 1.3.2 STEP 7 Basic V10.5 软件安装步骤 | 12 |
| 1.3.3 TIA 软件的界面特点 | 16 |
| 1.4 技能训练二：电动机起停 PLC控制程序的创建 | 17 |
| 1.4.1 三相电动机的直接起动控制 | 17 |
| 1.4.2 电动机正反转 PLC 控制 | 37 |
| 1.4.3 三相电动机的星-三角启动 PLC 控制 | 39 |
| 1.5 工程案例：自动分拣装置的 PLC 控制 | 46 |
| 1.5.1 案例介绍 | 46 |
| 1.5.2 硬件部分接线 | 49 |

| | |
|-------------------------------|----|
| 第2章 S7-1200 PLC扩展模块的应用 | 55 |
| 2.1 背景知识介绍：S7-1200 PLC的扩展模块 | 56 |
| 2.1.1 扩展模块介绍 | 56 |
| 2.1.2 扩展模块的变量寻址 | 57 |
| 2.1.3 数字量输入输出电路的工作原理及模块选型 | 58 |
| 2.1.4 模拟量输入输出电路的工作原理及模块选型 | 61 |
| 2.1.5 信号板选型 | 65 |
| 2.2 知识讲座：用户程序结构与数据类型 | 67 |
| 2.2.1 用户程序的执行 | 67 |
| 2.2.2 S7-1200 PLC实现控制的过程 | 70 |
| 2.2.3 S7-1200 PLC的数据类型 | 71 |
| 2.3 技能训练一：数字量扩展模块的使用 | 72 |
| 2.3.1 任务说明 | 72 |
| 2.3.2 电气接线及安装 | 73 |
| 2.3.3 PLC编程 | 74 |
| 2.4 技能训练二：模拟量扩展模块的应用 | 81 |
| 2.4.1 任务说明 | 81 |
| 2.4.2 模拟量模块选型与电气接线 | 82 |
| 2.4.3 编程软件的安装 | 84 |
| 2.5 工程案例：造纸生产线的速度自动控制 | 95 |
| 2.5.1 案例介绍 | 95 |
| 2.5.2 硬件配置、变量定义与软件编程 | 97 |

目 录



| | | | |
|----------------------------------|-----|------------|-----|
| 第3章 KTP 触摸屏编程 | 107 | 4.3.1 案例介绍 | 184 |
| 3.1 背景知识介绍：触摸屏的由来 | 108 | 4.3.2 硬件设计 | 185 |
| 3.1.1 工业触摸屏的出现 | 108 | 4.3.3 软件编程 | 188 |
| 3.1.2 触摸屏的种类 | 108 | 4.3.4 画面测试 | 196 |
| 3.1.3 基于现场总线与以太网的触摸屏 | 111 | | |
| 3.2 知识讲座：触摸屏与全集成自动化思想 | 113 | | |
| 3.2.1 触摸屏的基本功能和使用方法 | 113 | | |
| 3.2.2 触摸屏的设计原则 | 114 | | |
| 3.2.3 西门子公司的 TIA 思想 | 117 | | |
| 3.3 技能训练：KTP 600 触摸屏的使用 | 117 | | |
| 3.3.1 任务说明 | 117 | | |
| 3.3.2 电气接线 | 118 | | |
| 3.3.3 软件编程 | 124 | | |
| 3.4 工程案例：全自动定时喷淋系统 | 145 | | |
| 3.4.1 案例介绍 | 145 | | |
| 3.4.2 时钟和日历指令 | 146 | | |
| 3.4.3 变量定义与软件编程 | 147 | | |
| 第4章 S7-1200 PLC 的 PID 控制 | 163 | | |
| 4.1 背景知识介绍：PID 的基本概念 | 164 | | |
| 4.1.1 PID 的基本概念 | 164 | | |
| 4.1.2 PID 控制器 | 166 | | |
| 4.2 技能训练：液压站压力控制的 PID 构建 | 168 | | |
| 4.2.1 案例介绍 | 168 | | |
| 4.2.2 在 S7-1200 PLC 中添加 PID 工艺对象 | 169 | | |
| 4.2.3 PID 指令调用与编程 | 180 | | |
| 4.2.4 在线模式下激活 PID 控制器 | 182 | | |
| 4.3 工程案例：房间温度控制 | 184 | | |
| 第5章 S7-1200 PLC 的 PWM 控制 | 199 | | |
| 5.1 背景知识介绍：PWM 控制 | 200 | | |
| 5.1.1 PWM 控制的基本概念 | 200 | | |
| 5.1.2 S7-1200 PLC 的 PWM 应用 | 201 | | |
| 5.2 技能训练：人机界面控制 PWM | 204 | | |
| 5.2.1 控制要求 | 204 | | |
| 5.2.2 硬件配置与软件编程 | 204 | | |
| 5.2.3 实际测试画面 | 212 | | |
| 5.3 工程案例：食品生产线的喷嘴控制 | 212 | | |
| 5.3.1 案例介绍 | 212 | | |
| 5.3.2 PWM 硬件配置与软件编程 | 213 | | |
| 5.3.3 实际测试结果 | 220 | | |
| 第6章 S7-1200 PLC 的 HSC 控制 | 223 | | |
| 6.1 背景知识介绍：HSC 概念 | 224 | | |
| 6.1.1 脉冲量输入和高速计数器 | 224 | | |
| 6.1.2 S7-1200 PLC HSC 的指令与硬件 | 225 | | |
| 6.2 技能训练一：单相计数 | 229 | | |
| 6.2.1 案例介绍 | 229 | | |
| 6.2.2 硬件组态与软件编程 | 230 | | |
| 6.3 技能训练二：A/B 正交模式下的速度/频率的测量 | 237 | | |
| 6.3.1 案例介绍 | 237 | | |
| 6.3.2 硬件组态与软件编程 | 237 | | |
| 6.4 工程案例：自动定长切管机 | 242 | | |
| 6.4.1 案例介绍 | 242 | | |
| 6.4.2 硬件组态与软件编程 | 242 | | |





图解西门子 S7-1200 PLC 入门到实践

| | |
|---|------------|
| 第7章 S7-1200 PLC 的 PTO | |
| 控制..... | 255 |
| 7.1 背景知识介绍：运动控制与步进电动机 | 256 |
| 7.1.1 运动控制的基本架构 | 256 |
| 7.1.2 S7-1200 PLC 实现运动控制的基础 | 257 |
| 7.1.3 S7-1200 PLC 的 PTO 脉冲输出 | 258 |
| 7.1.4 驱动器 HB-4020M 的特点及其与 PLC 接线..... | 258 |
| 7.1.5 步进电动机的基本工作原理及选型 | 260 |
| 7.1.6 工艺对象“轴”的概念 | 262 |
| 7.2 技能训练：工艺对象“轴”的组态与调试 | 263 |
| 7.2.1 S7-1200 PTO 的硬件组态 | 263 |
| 7.2.2 组态工艺“轴” | 264 |
| 7.2.3 通过控制面板调试工艺“轴” | 270 |
| 7.2.4 诊断工艺“轴” | 272 |
| 7.3 工程案例：通过触摸屏控制工作台滑动座电动机 | 273 |
| 7.3.1 案例介绍 | 273 |
| 7.3.2 S7-1200 PLC 的硬件设计 | 274 |
| 7.3.3 硬件组态与软件编程 | 274 |
| 7.3.4 软件编程 | 278 |
| 7.3.5 调试及总结 | 286 |
| 第8章 S7-1200 PLC 的串口通信 | 291 |
| 8.1 背景知识介绍：串口通信与以太网通信 | 292 |
| 8.1.1 RS-232-C 串口和 RS-485 串口 | 292 |
| 8.1.2 CM1241 RS-232 和 RS-485 模块 | 293 |
| 8.2 技能训练一：S7-1200 PLC 与 PC 超级终端的串口通信 | 294 |
| 8.2.1 案例介绍 | 294 |
| 8.2.2 超级终端的设置 | 295 |
| 8.2.3 S7-1200 PLC 的硬件组态 | 296 |
| 8.2.4 S7-1200 PLC 的软件编程 | 300 |
| 8.3 工程案例：S7-1200 PLC 与 PAC3200 的 MODBUS 通信 | 306 |
| 8.3.1 案例介绍 | 306 |
| 8.3.2 指令介绍 | 306 |
| 8.3.3 S7-1200 PLC 与 PAC3200 进行 MODBUS RTU 的通信组态 | 311 |
| 8.3.4 PLC 的通信编程 | 312 |
| 8.3.5 测试与总结 | 316 |
| 第9章 S7-1200 PLC 的 OPC | |
| 组态 | 319 |
| 9.1 背景知识介绍：OPC 标准及其应用 | 320 |
| 9.1.1 OPC 的产生 | 320 |
| 9.1.2 OPC 的工作原理及规范 | 320 |
| 9.1.3 OPC 的服务器与客户机 | 321 |
| 9.1.4 OPC 数据访问 | 322 |
| 9.2 技能训练一：使用 SIMATIC NET 建立 OPC 数据链 | 322 |
| 9.2.1 SIMATIC NET 软件介绍 | 322 |
| 9.2.2 S7-1200 PLC 与 SIMATIC NET 通信的建立 | 322 |
| 9.3 技能训练二：WinCC 案例 | 332 |
| 9.3.1 WinCC 介绍 | 332 |
| 9.3.2 WinCC 与 S7-1200 CPU 通信 | 332 |
| 9.4 工程案例：IFIX 在变频器监控中的应用 | 338 |
| 9.4.1 案例介绍 | 338 |
| 9.4.2 S7-1200 PLC 的硬件设计与软件编程 | 338 |
| 9.4.3 IFIX 的设置与组态 | 340 |
| 9.4.4 工程案例小结 | 349 |
| 参考文献 | 351 |

第1章 S7-1200 PLC入门

可编程序控制器（Programmable Logic Controller, PLC）自20世纪60年代问世以来，很快地被应用到汽车制造、机械加工、冶金、矿业、轻工等各个领域，并大大地推进了机电一体化进程。经过长时间的发展和完善，PLC的编程概念和控制思想已为广大的自动化行业人员所熟悉，这是一个目前任何其他工业控制器（包括DCS和FCS等）都无法与之相提并论的巨大知识资源。西门子S7-1200 PLC作为中小型PLC的佼佼者，无论在硬件配置还是在软件编程上都具有强大的优势。

本章的学习目标如下：

知识目标

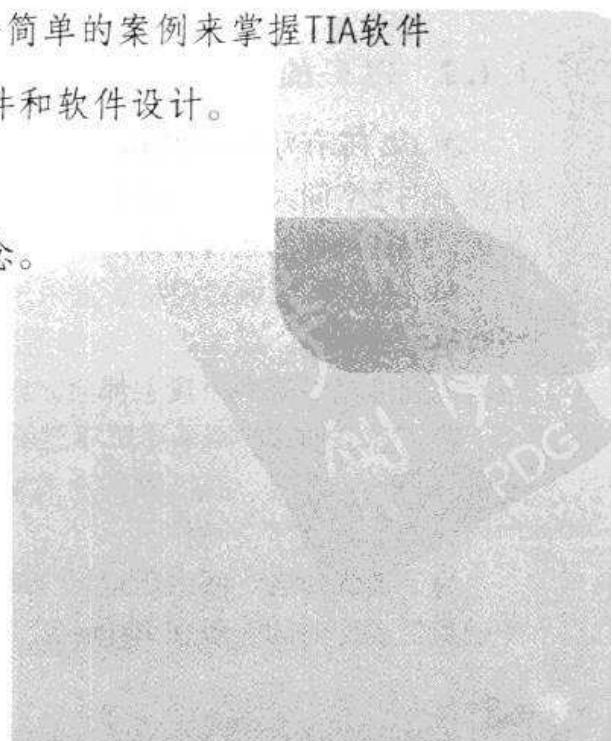
熟悉PLC产生的背景，了解PLC的定义，掌握S7-1200 PLC的硬件组成，掌握IEC 61131-3关于PLC编程语言的要点。

能力目标

能独立安装STEP 7 Basic软件，能通过一个简单的案例来掌握TIA软件的整个应用过程，能解决开关量控制的硬件和软件设计。

职业素养目标

能更新自身的知识库，掌握先进的编程理念。





1.1 背景知识介绍：PLC 的产生与原理

1.1.1 PLC 的产生

工业机器控制的传统系统采用硬连线继电器、定时器和计数器。但是，使用这种方法在全部接线完成以前，操作者不能测试控制元件。此外，如需任何改动都要花许多时间。这种系统的最大缺点在于，由于每个元件的使用寿命都很短，所以需要经常维修。

为了解决这类题，在 20 世纪 60 年代末研制出了 PLC。如图 1-1 所示为西门子公司最新的 S7-1200 系列 PLC 外观。

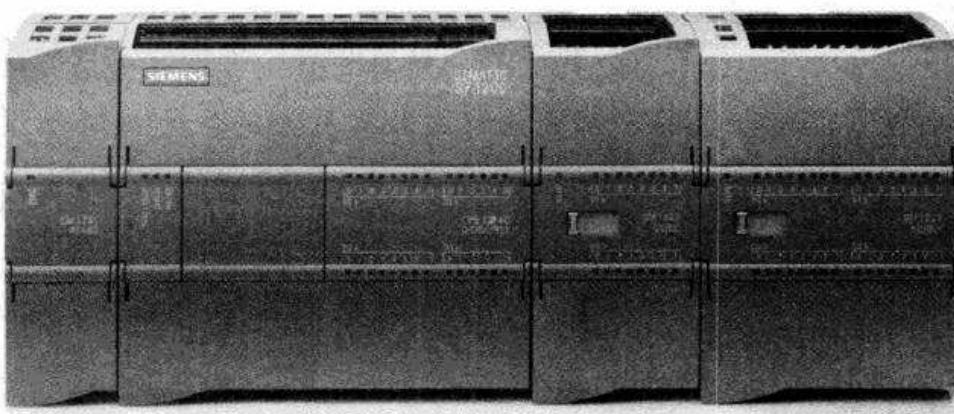


图 1-1 S7-1200 系列 PLC 外观

由于 PLC 的控制过程由程序创立，所以控制元件可以在程序内更改。PLC 还进行算术运算，控制高速生产线，这些操作使用传统的继电器、定时器和计数器的组合是不能进行的。由于近年来半导体器件和微型计算机技术的迅速发展，PLC 的使用寿命已大大延长，所以很少需要维修，因此它成为了工业控制不可或缺的重要控制元件。

1.1.2 PLC 的控制原理

图 1-2 所示为传统的继电器-接触器控制柜与 PLC 控制柜，显然前者接线复杂、所需元件多、占用空间大。最初研制生产的 PLC 主要用于代替传统的由继电器-接触器构成的控制装置，但这两者的运行方式是不相同的。

1) 继电器-接触器控制装置采用硬逻辑并行运行的方式，即如果这个继电器-接触器的线圈通电或断电，该继电器-接触器所有的触点（包括其常开或常闭触点）不论在继电器-接触器控制电路的哪个位置上都会立即同时动作。

2) PLC 的 CPU 则采用顺序逻辑扫描用户程序的运行方式，即如果一个输出线圈或逻辑线圈被接通或断开，该线圈的所有触点（包括其常开或常闭触点）不会立即动作，必须等扫描到该触点时才会动作。

为了消除两者之间由于运行方式不同而造成的差异，考虑到继电器-接触器控制装置各类触点的动作时间一般在 100ms 以上，而 PLC 扫描用户程序的时间一般均小于 100ms，因此



PLC 采用了一种不同于一般微型计算机的运行方式——扫描技术。这样在对于 I/O 响应要求不高的场合，PLC 与继电器-接触器控制装置在处理结果上就没有什么区别了。

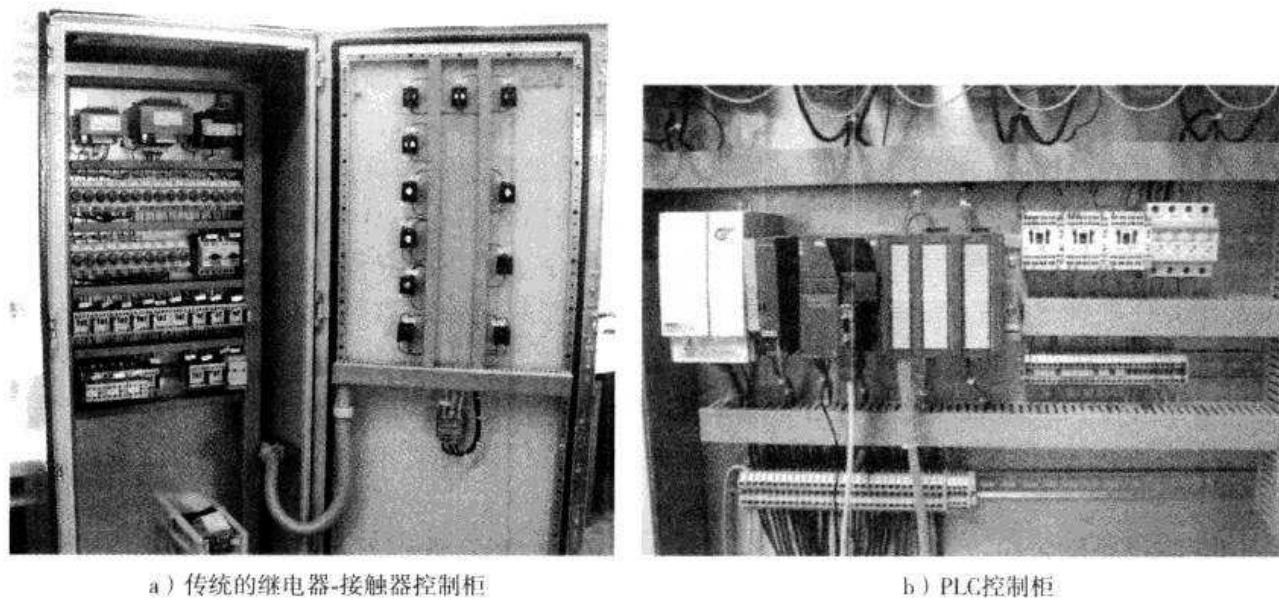


图 1-2 传统的继电器-接触器控制柜与 PLC 控制柜

当 PLC 投入运行后，其工作过程一般分为 3 个阶段，即输入采样、用户程序执行和输出刷新。完成上述 3 个阶段称作一个扫描周期。扫描技术如图 1-3 所示，在整个运行期间，PLC 的 CPU 以一定的扫描速度重复执行上述 3 个阶段。

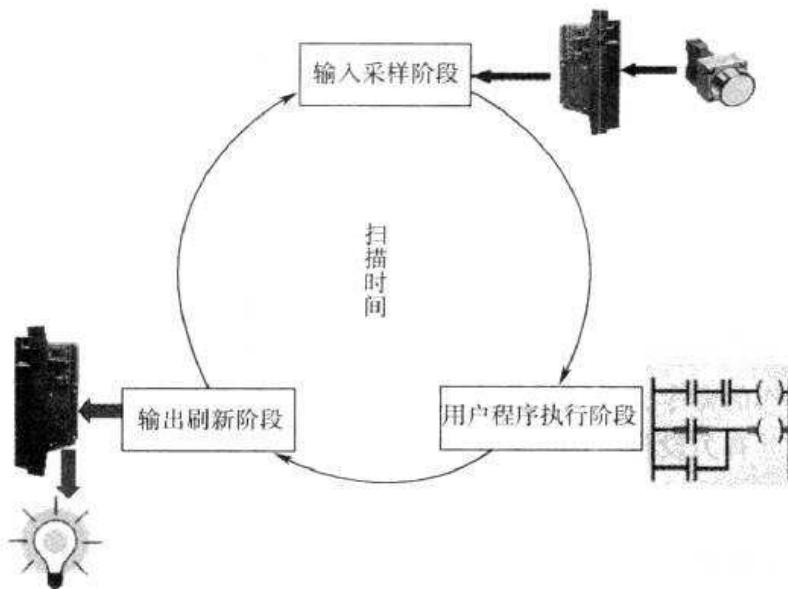


图 1-3 扫描技术

1. 输入采样阶段

在输入采样阶段，PLC 以扫描方式依次地读入所有输入状态和数据，并将它们存入 I/O 映像区中相应的单元内。输入采样结束后，转入用户程序执行和输出刷新阶段。在这两个阶



段中，即使输入状态和数据发生变化，I/O 映像区中相应单元的状态和数据也不会改变。因此，如果输入是脉冲信号，则该脉冲信号的宽度必须大于一个扫描周期才能保证在任何情况下该输入均能被读入。

2. 用户程序执行阶段

在用户程序执行阶段，PLC 总是按由上而下的顺序依次地扫描用户程序。在扫描每一条梯形图时，又总是先扫描梯形图左边的由各触点构成的控制电路，并按先左后右、先上后下的顺序对由触点构成的控制电路进行逻辑运算，然后根据逻辑运算的结果，刷新该逻辑线圈在系统 RAM 存储区中对应位的状态；或者刷新该输出线圈在 I/O 映像区中对应位的状态；或者确定是否要执行该梯形图所规定的特殊功能指令。即在用户程序执行过程中，只有输入点在 I/O 映像区内的状态和数据不会发生变化，而其他输出点和软设备在 I/O 映像区或系统 RAM 存储区内的状态和数据都有可能发生变化，而且排在上面的梯形图，其程序执行结果会对排在下面的凡是用到这些线圈或数据的梯形图起作用；相反，排在下面的梯形图，其被刷新的逻辑线圈的状态或数据只能到下一个扫描周期才能对排在其上面的程序起作用。

3. 输出刷新阶段

当扫描用户程序结束后，PLC 就进入输出刷新阶段。在此期间，CPU 按照 I/O 映像区内对应的状态和数据刷新所有的输出锁存电路，再经输出电路驱动相应的外设。这时，才是 PLC 的真正输出。

显然，图 1-3 中所示的梯形图就是输入到 PLC 的程序，又称为“用户程序”。在 PLC 操作期间，这个程序被反复执行，其扫描时间随用户程序的大小以及程序中所用的指令而不同。

输入/输出反应时间除包括扫描时间外，还包括输入/输出反应延时。这些延时是由以下原因造成的，即输入状态只能在输入操作期间读入存储器。如果输入状态在输入操作后改变，新的输入状态只能在下一个扫描时间内读入。

1.1.3 PLC 的定义

国际电工委员会 IEC 于 1982 年 11 月和 1985 年 1 月颁布了 PLC 标准的第一稿和第二稿，对 PLC 做了如下的定义：“PLC 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。它可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命令，并通过数字式、模拟式的输入和输出，控制各种类型的机械和生产过程。PLC 及其有关设备，都应易于与工业控制系统联成一个整体，易于以扩充功能的原则而设计。”

1.1.4 PLC 开关量控制的应用

用 PLC 进行开关量控制的实例很多，冶金、机械、轻工、化工、纺织等，几乎所有工业行业都需要用到它。如图 1-4 所示为包装生产线的 PLC 开关量控制。

PLC 控制开关量的能力很强，所控制的输入输出点数，少则十几点、几十点，多则几百、几千，甚至几万点。由于它能联网，点数几乎不受限制，不管多少点都能直接或间接地控制。

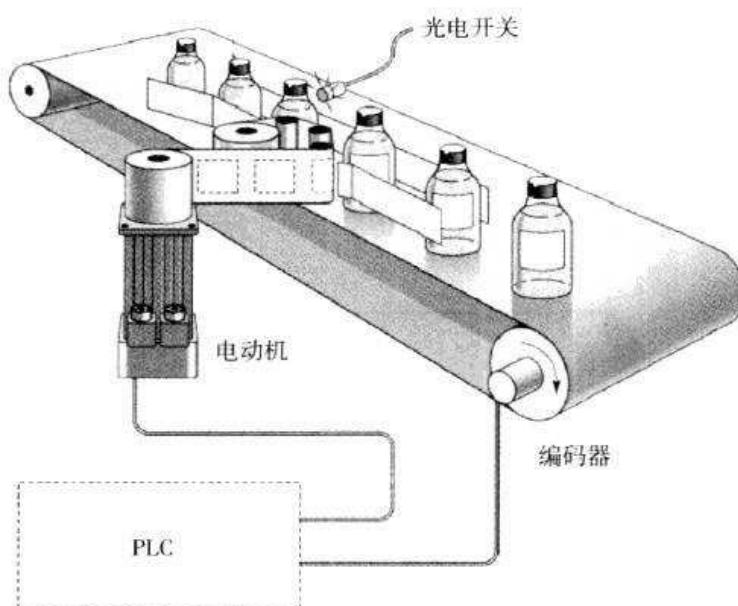


图 1-4 包装生产线的 PLC 开关量控制

在开关量控制中，PLC 所控制的逻辑问题可以是多种多样的：组合的、时序的；即时的、延时的；不需计数的、需要计数的；固定顺序的、随机工作的等。

1.2 知识讲座：S7-1200 PLC 的组成

1.2.1 S7-1200 PLC 的硬件组成

组成 PLC 的模块是 PLC 的硬件基础，只有弄清所选用的 PLC 都具有哪些模块及其特点，才能正确地选用模块，去组成一台完整的 PLC，以满足控制系统对 PLC 的要求。

1. CPU 模块

CPU 模块是 PLC 的硬件核心，PLC 的主要性能，如速度、规模等都由它的性能来体现。如图 1-5 所示，S7-1200 PLC 的 CPU 模块包括 CPU、电源、输入信号处理回路、输出信号处理回路、存储区、RJ45 端口和扩展模块接口。其本质为一台计算机，该计算机负责系统程序的调度、管理、运行和 PLC 的自诊断，负担将用户程序作出编译解释处理以及调度用户目标程序运行的任务。与之前的西门子 S7-200 系列 CPU 模块最大的区别在于它配置了以太网接口 RJ45，并可以采用一根标准网线与安装有 STEP 7 Basic 软件的 PC 进行通信，这也是它的优点之一。

目前，西门子公司提供 CPU1211C、CPU1212C 和 CPU1214C 三种类型，以 CPU1211C 为例，图 1-6 所示为其外观示意。

S7-1200 系列 PLC 不同的 CPU 模块提供了不同的特征和功能，这些特征和功能可帮助用户针对不同的应用创建有效的解决方案。

这里需要注意的是：S7-1200 PLC 的 I/O 端子与 S7-200 PLC 刚好相反，前者是输入在上，输出在下，这对已经使用过 S7-200 PLC 的用户尤其要引起关注，以防接错线。

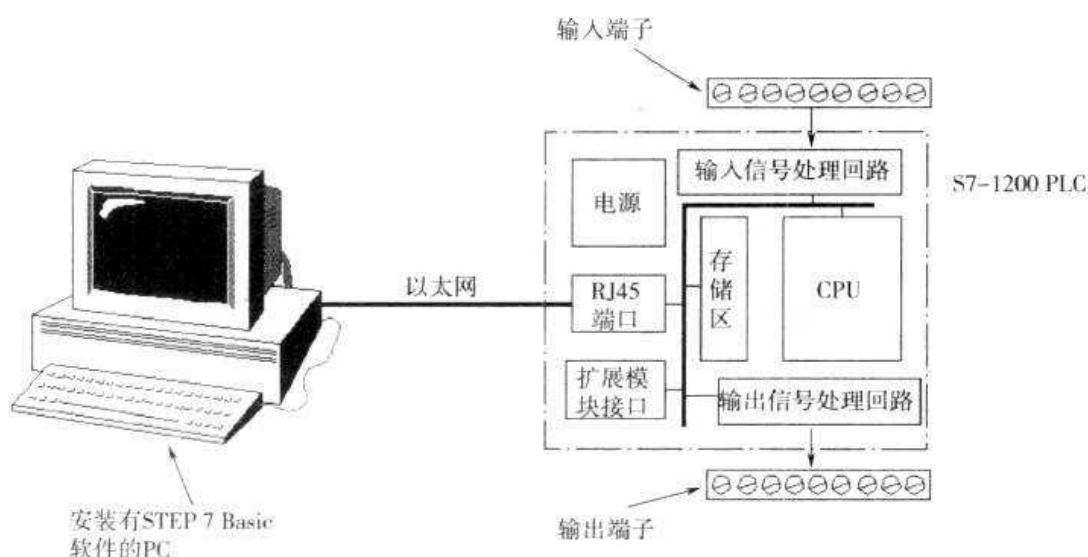


图 1-5 S7-1200 PLC 的 CPU 模块

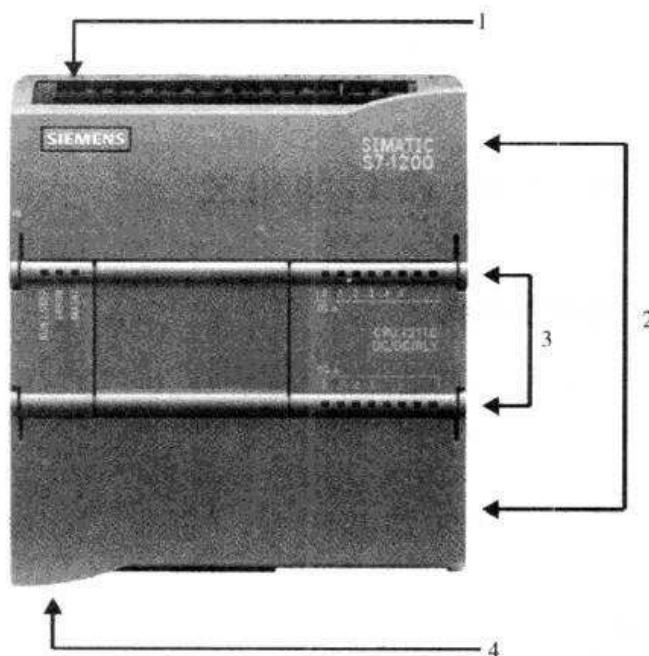


图 1-6 CPU1211C 的外观示意

1—电源接口 2—可拆卸用户接线连接器（保护盖下面）；存储卡插槽（上部保护盖下面）
3—板载 I/O 的状态 LED 4—PROFINET 连接器（CPU 的底部）

2. 信号模块 (SM)

信号模块 (Signal Module) 用于扩展控制器的输入和输出通道，可以使 CPU 增加附加功能。信号模块连接在 CPU 模块右侧 (见图 1-7)，与 S7-200 系列 PLC 不同的是它的全新安装方式。



3. 信号板 (SB)

信号板 (Signal Board) 为 S7-1200 PLC 所特有的，通过信号板 (SB) 给 CPU 模块增加 I/O。每一个 CPU 模块都可以添加一个具有数字量或模拟量 I/O 的 SB，SB 连接在 CPU 的前端。信号板如图 1-8 所示。

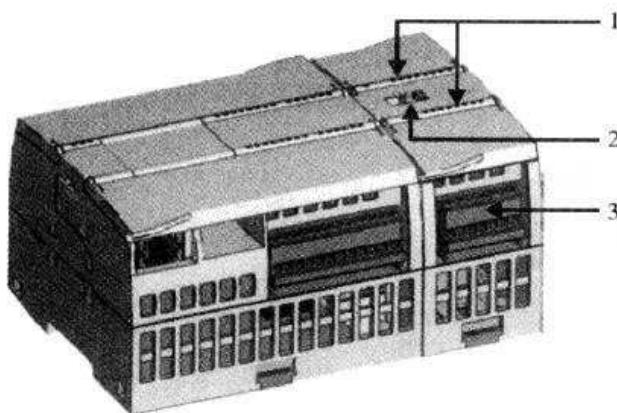


图 1-7 信号模块

1—信号模块的 I/O 的状态 LED
2—总线连接器 3—可拆卸用户接线连接器

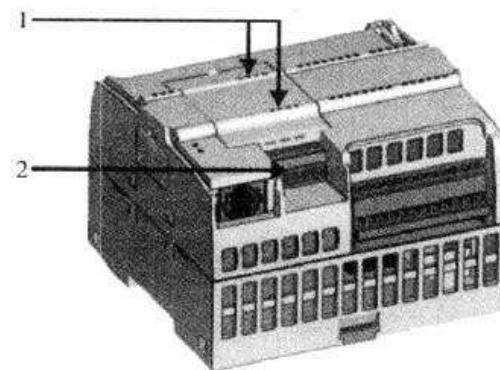


图 1-8 信号板

1—SB 上的状态 LED 2—可拆卸用户接线连接器

4. 存储卡

如果确实需要安全保护数据，可将用户程序存储在存储卡内，用这种方式可保证断电时不会丢失数据或程序。存储卡以 FLASH EPROM 提供最大 512KB 存储器。它们是直接在 CPU 内编程，因此不需要 MC 编程器。存储卡在 CPU 上的中央数据管理方面也起到重要作用，这是因为连接 I/O 模块的所有参数化数据都安全的存储在存储卡上。要插入存储卡，需打开 CPU 顶盖（见图 1-9），然后将存储卡插入到插槽中。推弹式连接器可以轻松地插入和取出。要求正确安装存储卡。

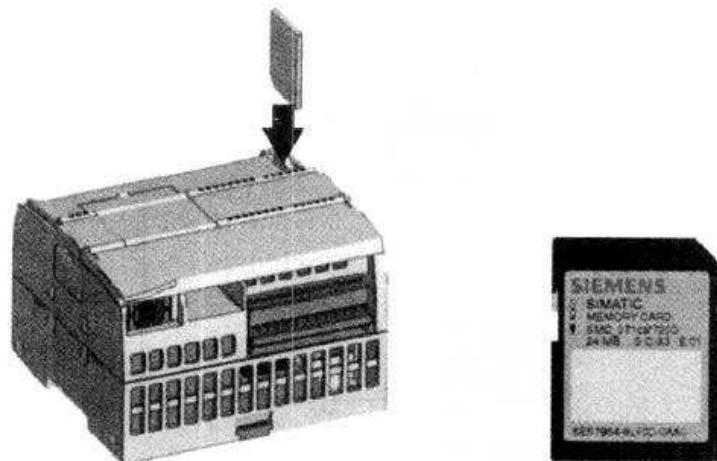


图 1-9 存储卡的插入

5. 通信模块

通信模块接入 PLC 后，可使 PLC 与计算机，或 PLC 与 PLC 进行通信，有的还可实现与其他控制部件，如与变频器、温控器通信，或组成局部网络。通信模块代表 PLC 的组网能力，是代表 PLC 性能的重要内容。

1.2.2 以 S7-1200 PLC 为核心的控制系统

在以 S7-1200 PLC 为核心的控制系统中（见图 1-10），I/O 部分用以接收信号或输出信号，便于与 PLC 进行人机对话。输入的有条码读入器、输入模拟量的电位器等。输出的有



图解西门子 S7-1200 PLC 入门到实践

打印机、编程器、监控器（虽也可对 PLC 输入信息，从 PLC 输出信息，但 I/O 设备实现人机对话更方便，可在现场条件下实现，便于使用）。随着技术的进步，这种设备将更加丰富。

PLC 与生产过程相连接的输入通道，输入部分接受来自生产现场的各种信号，如行程开关、热电偶、光电开关、按钮等。

PLC 与生产过程相连接的输出通道，输出部分接收 CPU 的处理输出，并转换成被控设备所能接受的电压、电流信号，以驱动被控设备，如继电器、电磁阀、指示灯等。

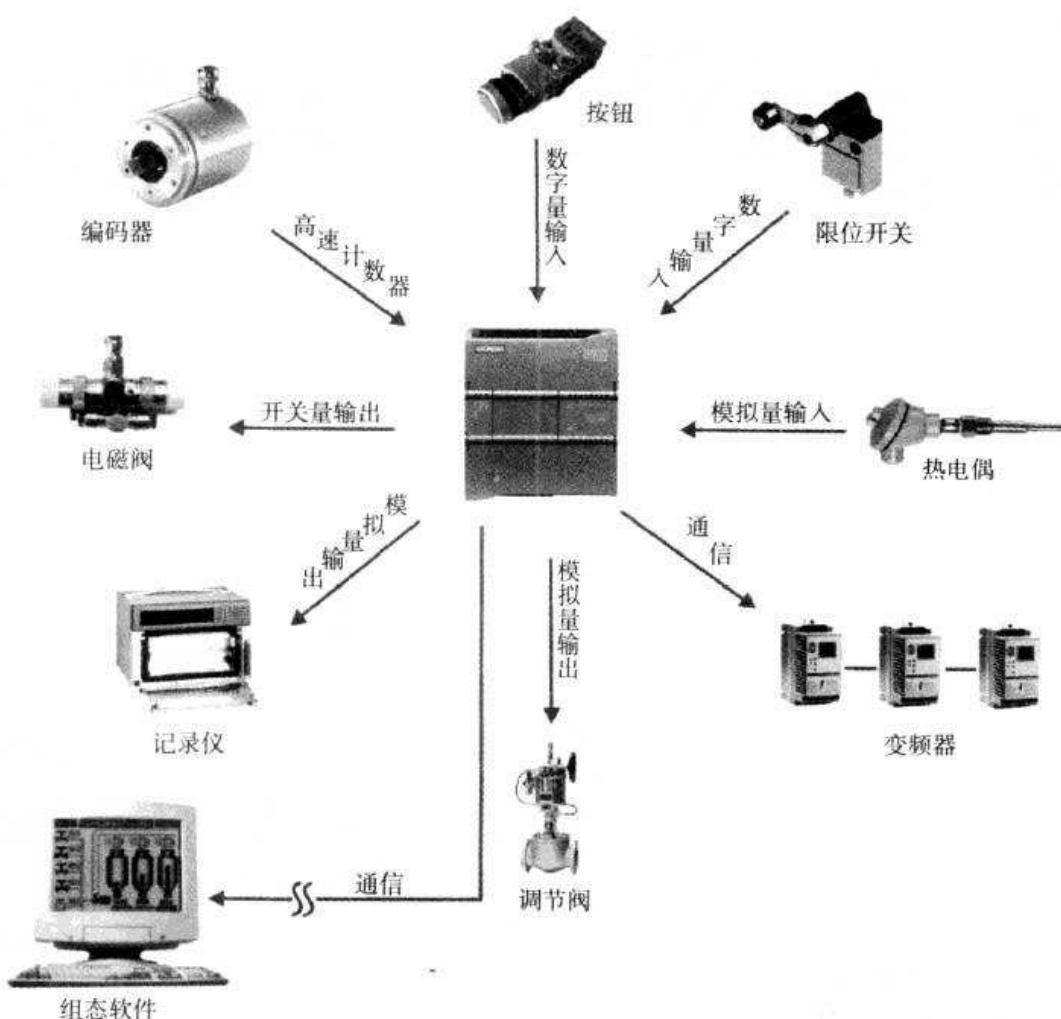


图 1-10 S7-1200 与外部设备的连接

1.2.3 IEC 61131-3 标准

S7-1200 PLC 采用 IEC 61131-3 标准，因此首先必须要了解该标准的具体内容。IEC 61131-3 国际标准得到了包括德国西门子公司、美国 AB 公司等世界知名大公司在内的众多厂家的共同推动和支持，它极大地改进了工业控制系统的编程软件质量，提高了软件开发效率；它定义的一系列图形化语言和文本语言（见图 1-11）不仅给系统集成商和系统工程师的编程带来很大的方便，而且给最终用户同样也带来了很大的方便；它在技术上的实现是高水平的，有足够的发展空间和变动余地，能很好地适应发展。

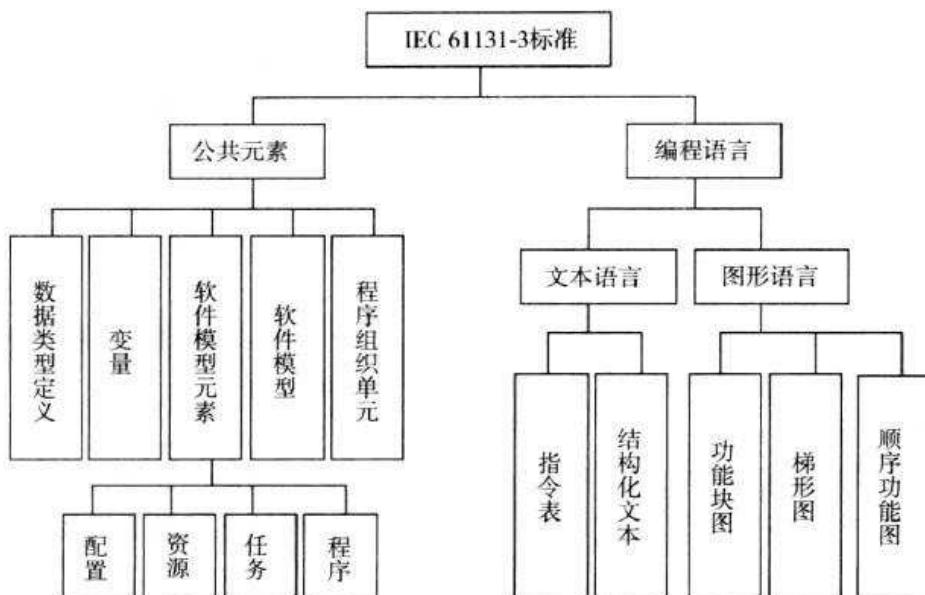


图 1-11 IEC61131-3 标准的图形化语言和文本语言

IEC 61131-3 标准为 PLC 制定了 5 种标准的编程语言，包括图形化编程语言和文本化编程语言。

图形化编程语言包括：梯形图（Ladder Diagram, LD）、功能块图（Function Block Diagram, FBD）、顺序功能图（Sequential Function Chart, SFC）。文本化编程语言包括：指令表（Instruction List, IL）和结构化文本（Strutured Text, ST）。

IEC 61131-3 标准的编程语言是 IEC 工作组在对世界范围的 PLC 厂家的编程语言合理地吸收、借鉴的基础上形成的一套针对工业控制系统的国际编程语言标准，它不但适用于 PLC 系统，而且还适用于更广泛的工业控制领域，为 PLC 编程语言的全球规范化做出了重要的贡献。

1. 梯形图语言

继电器梯形图语言是 PLC 首先采用的编程语言，也是 PLC 普遍采用的编程语言。梯形图编程语言是从继电器控制系统原理图的基础上演变而来的，与继电器控制系统梯形图的基本思想是一致的，只是在使用符号和表达方式上有一定区别。PLC 的设计初衷是为工厂车间电气技术人员而使用的，为了符合继电器控制电路的思维习惯，作为首先在 PLC 中使用的编程语言，梯形图保留了继电器电路的风格和习惯，成为广大电气技术人员最容易接受和使用的语言。

梯形图程序设计语言的特点是：

- 1) 与电气操作原理图相对应，具有直观性和对应性；
- 2) 与原有继电器逻辑控制技术相一致，对电气技术人员来说，易于掌握和学习；
- 3) 与原有的继电器逻辑控制技术的不同点是，梯形图中的能流（Power Flow）不是实际意义的电流，内部的继电器也不是实际存在的继电器，因此应用时需与原有继电器逻辑控制技术的有关概念区别对待；
- 4) 与指令表程序设计语言有一一对应关系，便于相互的转换和程序的检查。



2. 功能块图语言

功能块图采用类似于数字逻辑门电路的图形符号，逻辑直观，使用方便，它有梯形图编程中的触电和线圈等价的指令，可以解决范围广泛的逻辑问题。功能块图程序设计语言有如下特点：

- 1) 以功能模块为单位，从控制功能入手，使控制方案的分析和理解变得容易；
- 2) 功能模块是用图形化的方法描述功能，它的直观性大大方便了设计人员的编程和组态，有较好的易操作性；
- 3) 对控制规模较大、控制关系较复杂的系统，由于控制功能的关系可以较清楚地表达出来，因此，编程和组态时间可以缩短，调试时间也能减少。

3. 顺序功能图语言

顺序功能图亦称流程图或状态转移图，是一种图形化的功能性说明语言，专用于描述工业顺序控制程序，使用它可以对具有并发、选择等复杂结构的系统进行编程。顺序功能图程序设计语言有如下特点：

- 1) 以功能为主线，条理清楚，便于对程序操作的理解和沟通；
- 2) 对大型的程序，可分工设计，采用较为灵活的程序结构，可节省程序设计时间和调试时间；
- 3) 常用于系统的规模较大、程序关系较复杂的情况；
- 4) 只有在活动步的命令和操作被执行后，才对活动步后的转换进行扫描，因此整个程序的扫描时间较其他程序编制的程序扫描时间要大大缩短。

4. 指令表语言

指令表编程语言类似于计算机中的助记符汇编语言，它是 PLC 最基础的编程语言。所谓指令表编程，是用一个或几个容易记忆的字符来代表 PLC 的某种操作功能。指令表程序设计语言有如下特点：

- 1) 采用助记符来表示操作功能，具有容易记忆，便于掌握的特点；
- 2) 在编程器的键盘上采用助记符表示，具有便于操作的特点，可在无计算机的场合进行编程设计；
- 3) 与梯形图有一一对应的关系，其特点与梯形图语言基本相同。

5. 结构化文本语言

结构化文本是一种高级的文本语言，可以用来描述功能，功能块和程序的行为，还可以在顺序功能流程图中描述步、动作和转变的行为。结构化文本语言表面上与 PASCAL 语言很相似，但它是一个专门为工业控制应用开发的编程语言，具有很强的编程能力，用于对变量赋值、回调功能和功能块、创建表达式、编写条件语句和迭代程序等。结构化文本程序设计语言有如下特点：

- 1) 采用高级语言进行编程，可以完成较复杂的控制运算；
- 2) 需要有一定的计算机高级程序设计语言的知识和编程技巧，对编程人员的技能要求较高，普通电气人员无法完成；
- 3) 直观性和易操作性等性能较差；
- 4) 常被用于采用功能模块等其他语言较难实现的一些控制功能的实施。

不是所有的 PLC 都支持所有的编程语言（如功能块图、顺序功能图就有很多中低档



PLC 不支持），而大型的 PLC 控制系统一般都支持这 5 种标准编程语言或类似的编程语言。还有一些标准以外的编程语言，它们虽然没有被选择进标准语言中，但是它们是为了适合某些特殊场合的应用而开发的，在某些情况下，它们也许是较好的编程语言。

1.3 技能训练一：STEP 7 Basic 软件的安装

1.3.1 STEP 7 Basic V10.5 软件

西门子公司最新推出的小型 PLC S7-1200 系列，其编程软件为 SIMATIC STEP 7 Basic V10.5，与以往的 STEP7 MicroWin 相比有较大改观。

2009 年 5 月 18 日，西门子公司在西门子北京中心举办了全新小型可编程序控制器系列 S7-1200 的发布仪式，并推出了 STEP 7 Basic V10.5 SP1 软件。2009 年 12 月 9 日，具有更强大功能的 STEP 7 Basic V10.5 SP2 正式发布，新版本的发布为 S7-1200 PLC 在中国的推广和应用奠定了坚实的基础，该版本增加了中文编程用户界面、读取域和写入域指令等。

STEP 7 Basic 软件是在一个应用程序中集成了各种西门子 SIMATIC 全集成自动化（即 Totally Integrated Automation）产品，利用这一软件用户可以大大提高生产力和效率。所有 TIA 产品在该软件中能做到协作顺畅并一目了然，能支持用户创建的各种领域的自动化解决方案。

如图 1-12 所示是利用 STEP 7 Basic 软件来解决的典型自动化方案，它包含以下内容：PLC 使用用户程序来控制机器和用户使用人机界面设备操作和监视过程。

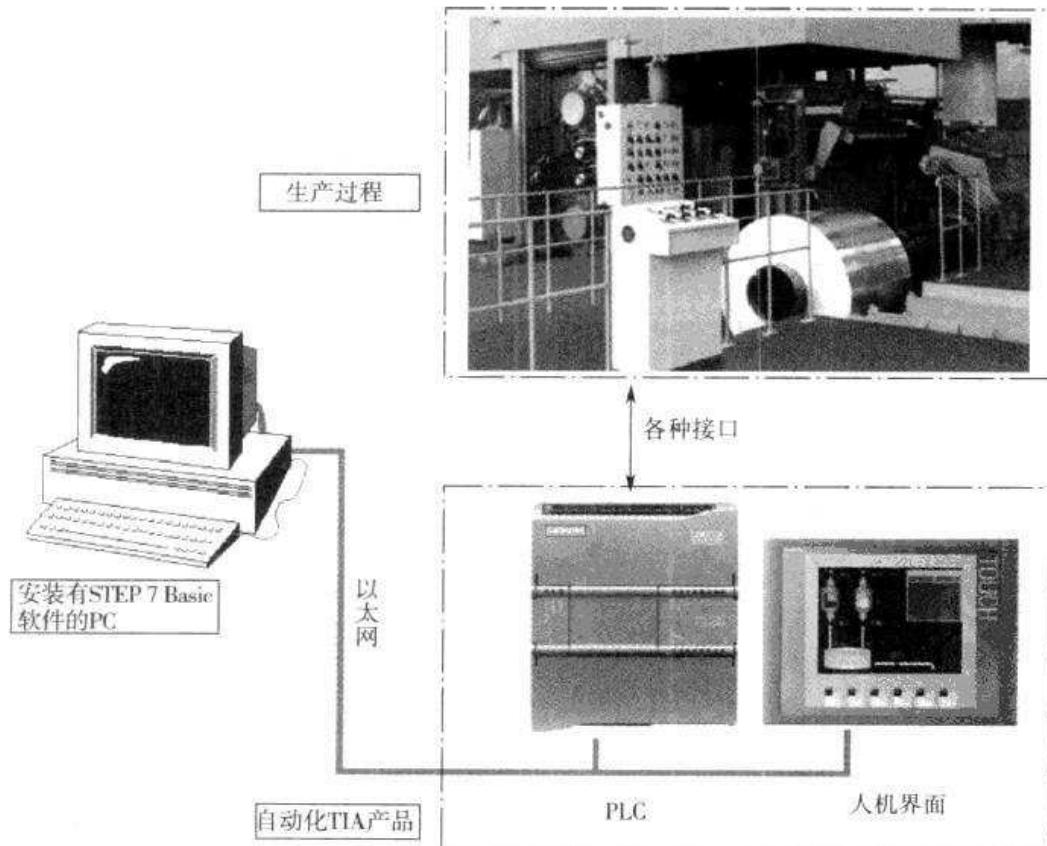


图 1-12 典型的自动化方案



1.3.2 STEP 7 Basic V10.5 软件安装步骤

软件安装的具体步骤如下：

1) 在要求选择安装语言的对话框里，选择需要安装的语言（在这里选择中文），如图 1-13 所示。然后选择在安装后的应用程序中所应用的语言，如图 1-14 所示。

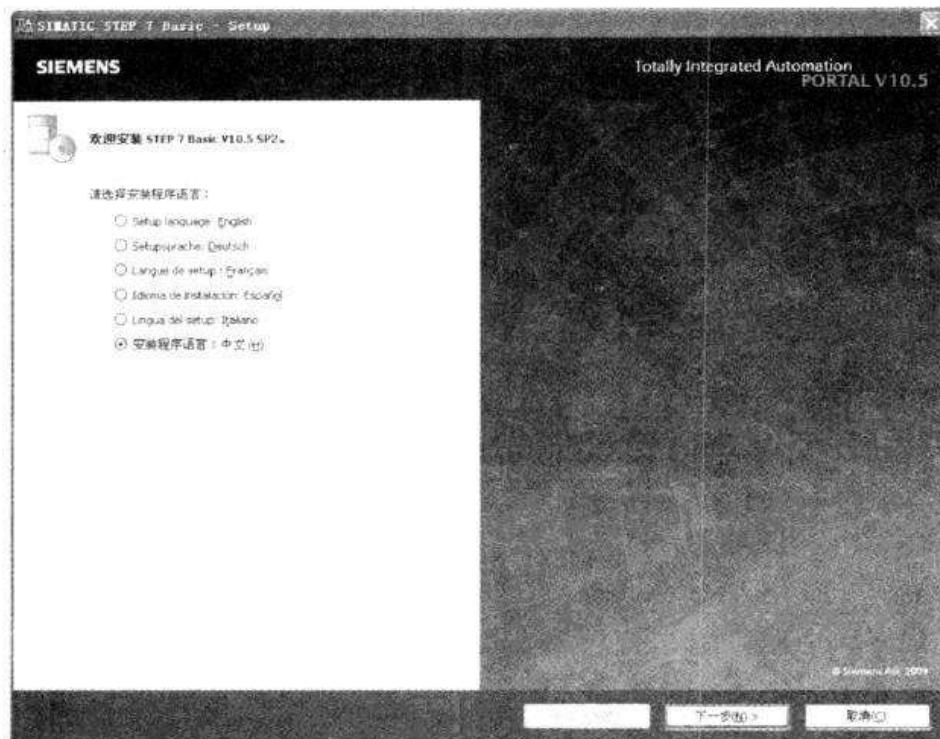


图 1-13 选择安装语言

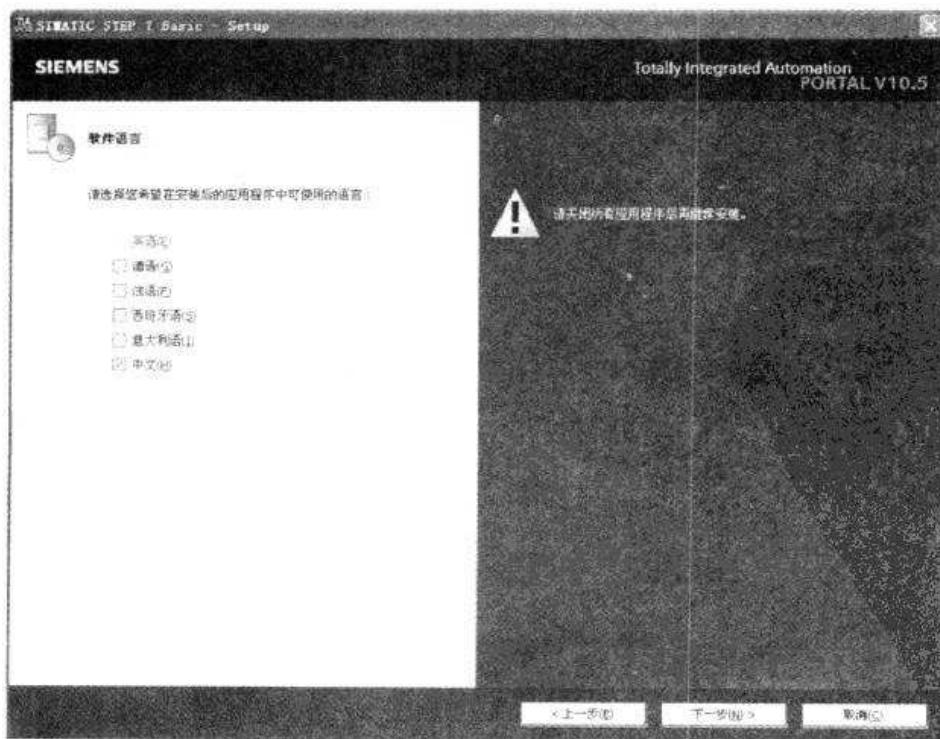


图 1-14 选择在安装后的应用程序中所应用的语言

第1章 S7-1200 PLC入门



2) 如图 1-15 所示，在产品安装的描述和路径的对话框中，选择相应的安装路径。随后便会出现本软件的授权协议，如图 1-16 所示。

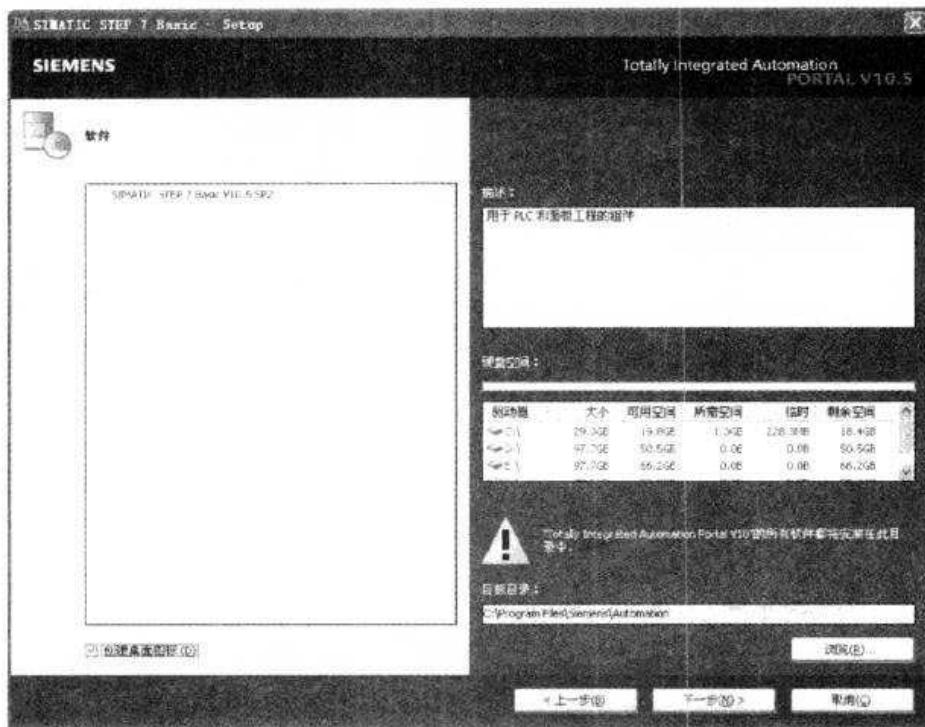


图 1-15 选择安装路径



图 1-16 软件授权协议

3) 在接受西门子公司的授权协议后，就开始了正式的软件安装（见图 1-17），其包括



图解西门子 S7-1200 PLC 入门到实践

SIMATIC STEP 7 Basic V10.5、Siemens Automation License Manager 和 SIMATIC WinCC Basic V10.5 3 个软件。

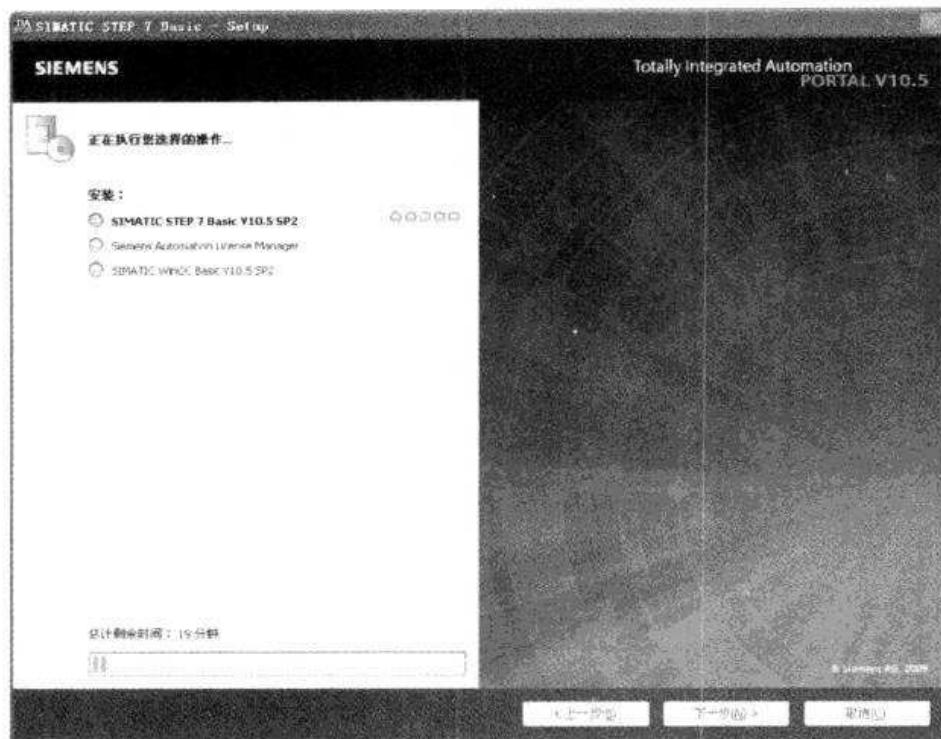


图 1-17 正式安装软件

需要注意的是，在安装软件的过程中，会提示安装 Microsoft SQL server 软件（见图 1-18）。

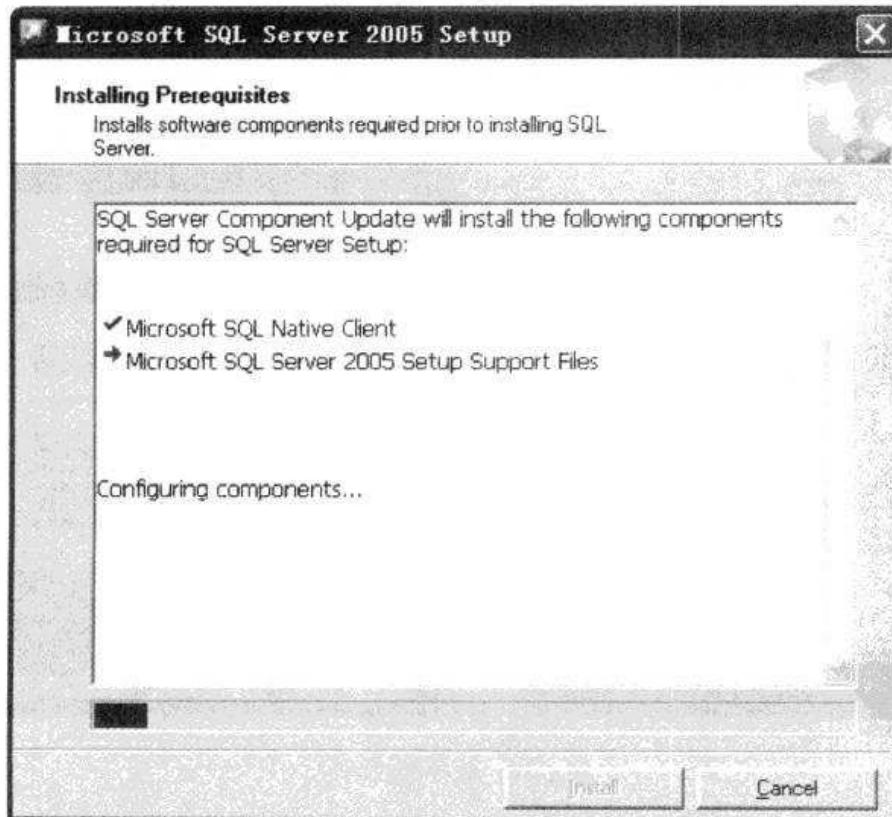


图 1-18 安装 Microsoft SQL server 软件



在 STEP 7 Basic 软件中，其数据库服务器选择采用客户机/服务器计算机模型的 Microsoft SQL Server 作为后台关系数据库服务器，它可以根据用户需要建立用户注册信息表、历史数据表、历史报警数据表、历史事件数据表、日志表、用户登陆信息表等，能够方便地对数据库进行操作，进行历史数据、历史报警、事件记录、归档和查询。

4) 如果安装成功，计算机屏幕会显示成功安装的消息（见图 1-19），会提示用户进行计算机重启。如果安装时有错误发生，将会显示错误消息，从中可知错误的类型，同时可以利用安装程序进行修改、修复或卸载（见图 1-20）。

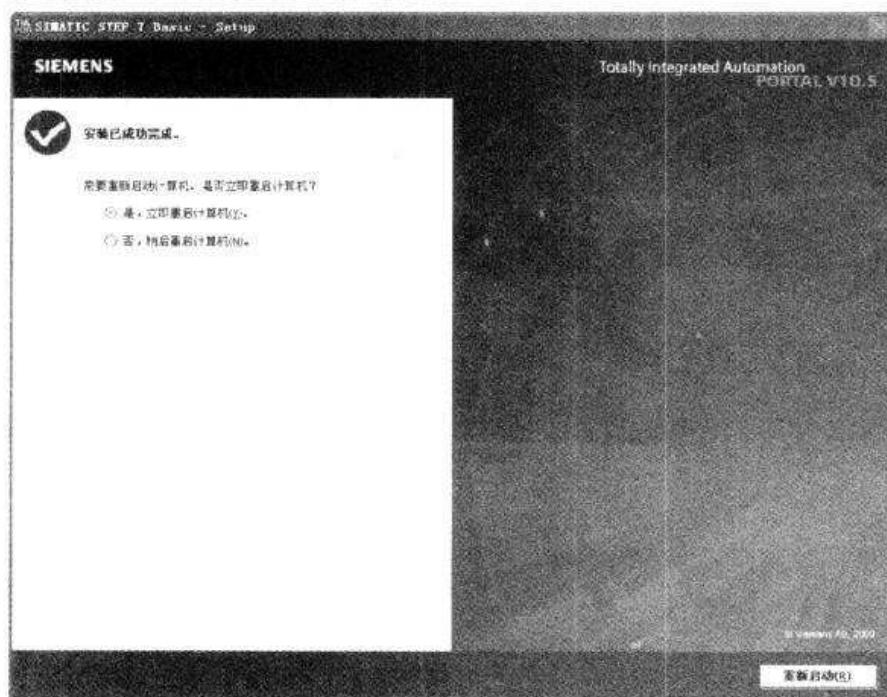


图 1-19 安装成功消息



图 1-20 修改、修复或卸载



图解西门子 S7-1200 PLC 入门到实践

5) 成功安装后，将会出现两个程序的快捷启动图标，分别是 Totally Integrated Automation 软件（简称 TIA 软件）和 Automation License Manager 软件。TIA 软件包括了 SIMATIC STEP 7 Basic 和 SIMATIC WinCC Basic。

如图 1-21 所示为 Automation License Manager 软件的界面，这意味着在本机中已经安装了无限制的 WinCC Basic 和 STEP 7 Basic 软件授权。



图 1-21 Automation License Manager 的软件界面

如图 1-22 所示为 TIA 软件的界面，点击“已安装产品”，将提示目前所安装软件的种类和版本。

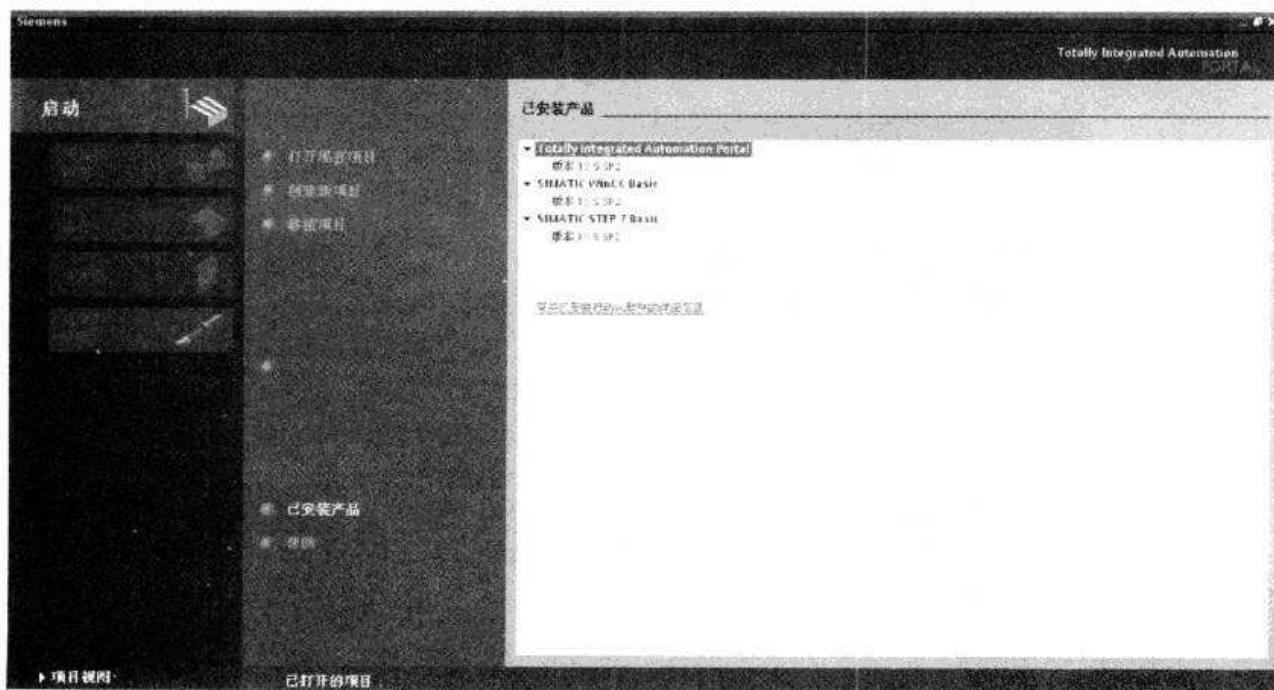


图 1-22 TIA 软件的界面

1.3.3 TIA 软件的界面特点

TIA 软件提供两种优化的视图，即起始视图（见图 1-23）和项目视图（见图 1-24）。

如图 1-23 所示，起始视图可以清晰地显示自动化项目的所有任务入口，初次使用者可以快速上手使用，通过它可以快速地找到自动化任务的正确编辑器，并且编辑器之间切换方便，直接可以在线切换。在起始视图中，“设备和网络”是定义和配置项目中的设备和它们之间的通信关系，如配置 PLC、人机界面和两者的网络，同时也可以通过使用共同变量建立必要的逻辑连接；“PLC 编程”是为项目中的每个 PLC 设备创建控制程序；“可视化”是创建人机界面的画面组态；“在线和诊断”是显示连接的设备及状态。

项目视图与起始视图不同，它可以访问所有的编辑器、参数和数据，可以进行高效的工程组态和编程，具体内容将结合案例在下一单元进行阐述。



图 1-23 起始视图

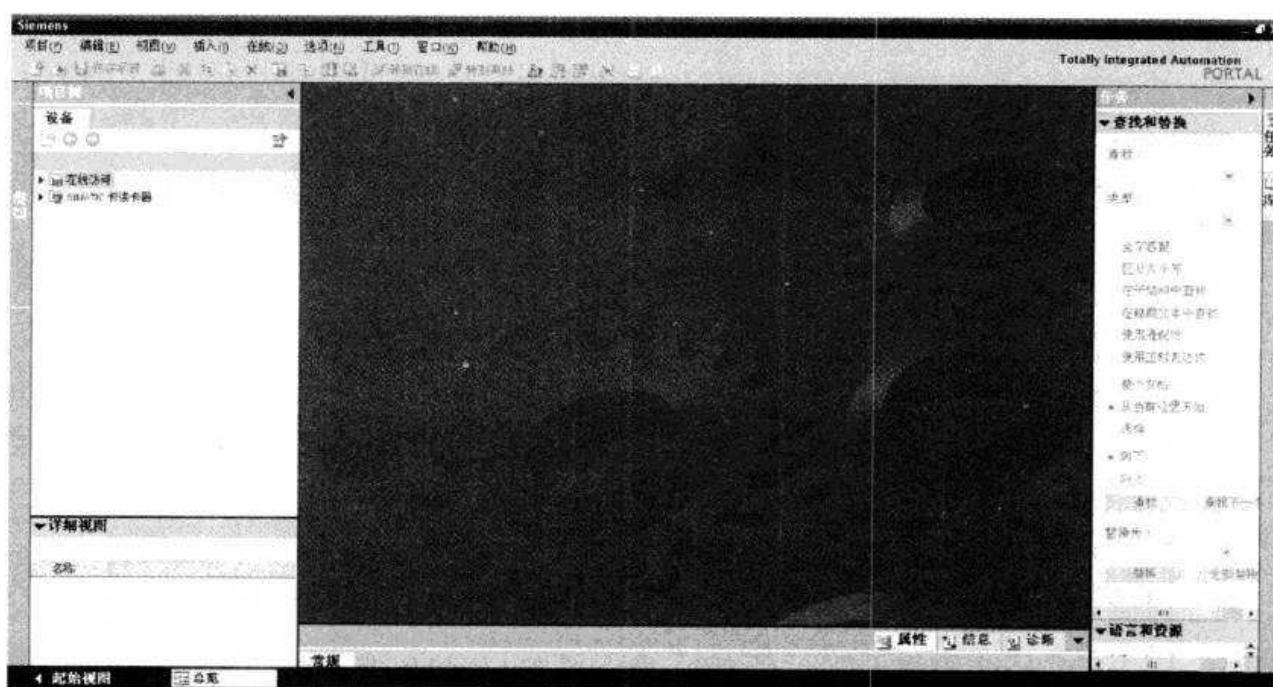


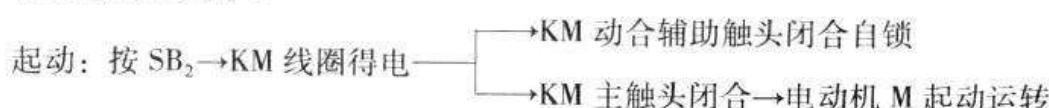
图 1-24 项目视图

1.4 技能训练二：电动机起停 PLC 控制程序的创建

1.4.1 三相电动机的直接起动控制

1. 任务说明

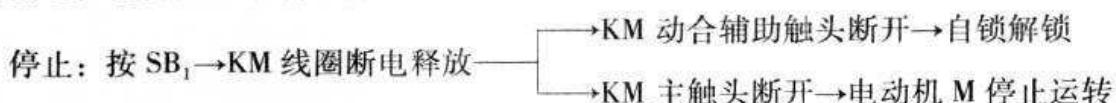
三相异步电动机的直接起动控制如图 1-25 所示，电路的工作原理如下：
合上电源开关 QS，





图解西门子 S7-1200 PLC 入门到实践

松开启动按钮 SB₂，由于接在按钮 SB₂ 两端的 KM 动合辅助触头闭合自锁，控制回路仍保持接通，电动机 M 继续运转。



现在要求对控制电路进行 S7-1200 PLC 改造，请设计合理的硬件图，并进行软件编程。

2. S7-1200 PLC 电气接线

本书所有的案例都采用 S7-1200 CPU1214C DC/DC/DC 进行接线与编程，因此首先需要了解该型 PLC 的具体接线（见图 1-26）。

从图 1-26 中可以看出，S7-1200 CPU1214C DC/DC/DC 的接线有以下几个特点：

- 1) 外部传感器的接线可以借用 PLC 的输入电源 DC 24V；
- 2) PLC 的输入电源和输出电源可以采用同一个电源，也可以采用不同的直流电源；
- 3) DC 24V 输入既可以采取 NPN 输入，也可以采用 PNP 输入。

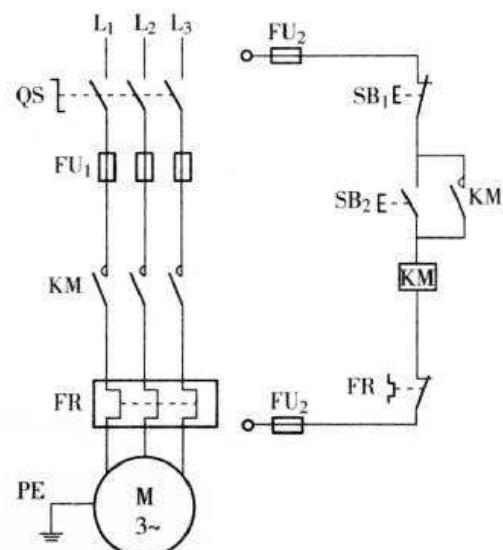


图 1-25 三相异步电动机的直接起动

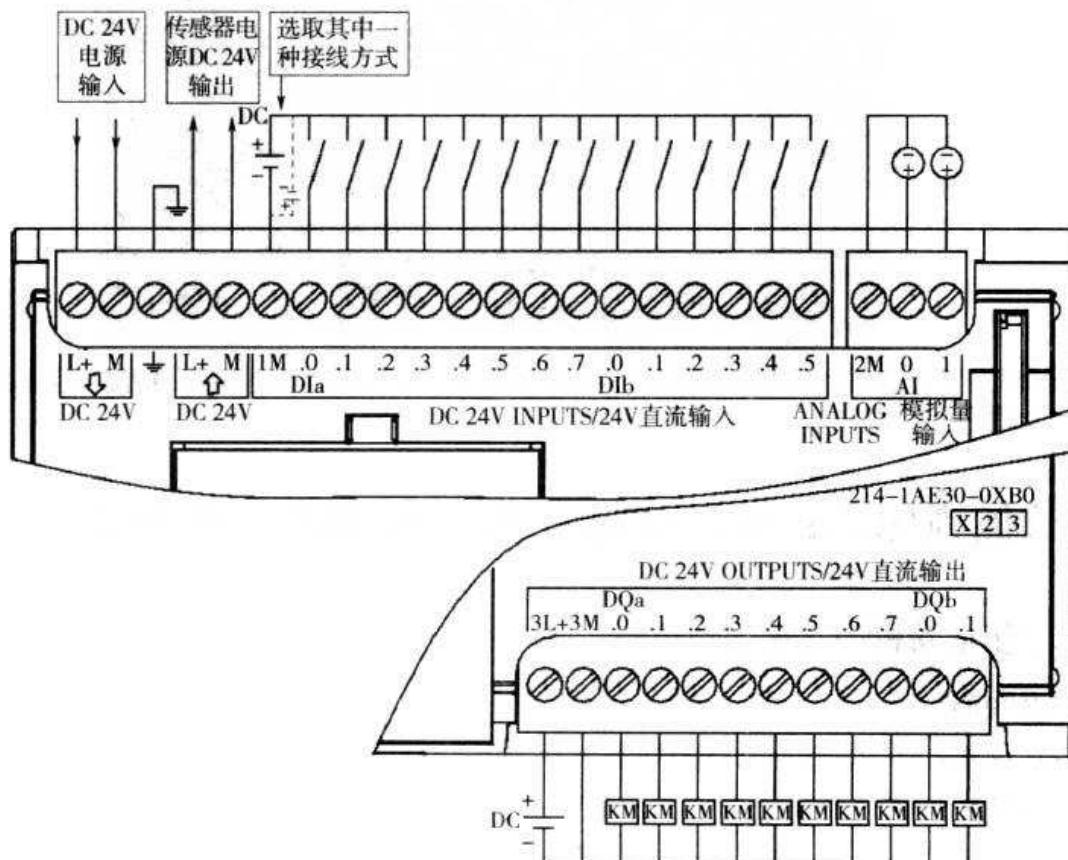


图 1-26 CPU 1214C DC/DC/DC 电气接线



根据以上原则，可以画出本方案的 PLC 原理图（见图 1-27）。

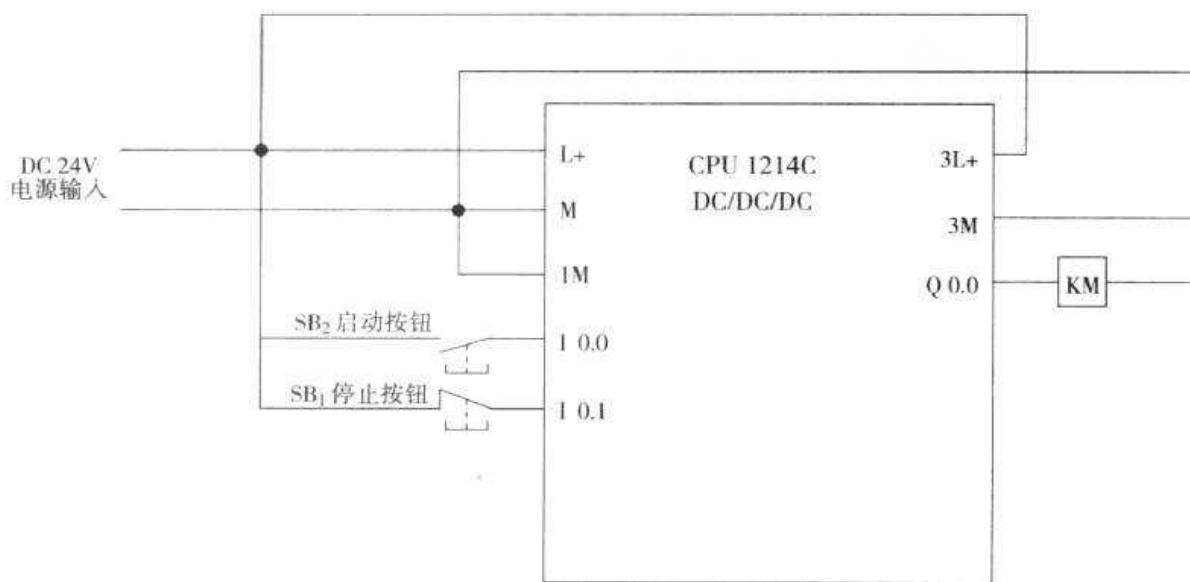


图 1-27 电动机直接起动的 PLC 原理图

3. TIA 软件的应用

TIA 软件可用来帮助用户组态自动化解决方案，其关键的组态步骤依次为：创建项目、配置硬件、设备联网、对 PLC 或人机界面编程、装载组态数据和使用在线和诊断功能。

1) 创建新项目，输入项目名称及存放路径。

对于本案例来说，首先要在起始视图中创建一个新项目（见图 1-28），然后输入项目名称，比如 Motor1，并点击 图符输入存放路径（见图 1-29 和图 1-30）。



图 1-28 创建新项目

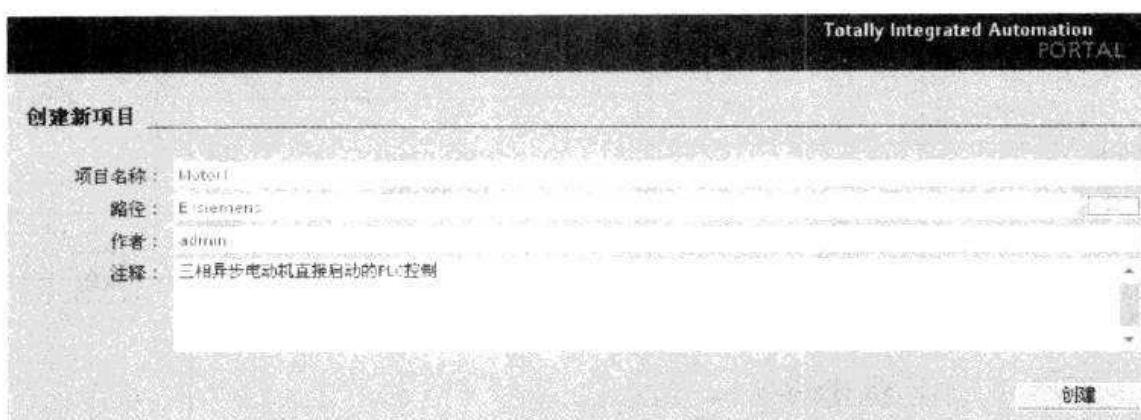


图 1-29 输入项目名称



图 1-30 选择存放路径

2) 切换到项目视图，熟悉项目树、设备和网络、硬件目录、信息窗口等。

从起始视图切换到项目视图，如图 1-31 所示。它共分为项目树、设备与网络、硬件目录、信息窗口等。

3) 硬件配置初步——添加新设备。

与 S7-200 PLC 不同，S7-1200 PLC 提供了完整的硬件配置。从项目树中，选择“添加新设备”，如图 1-32 所示，首先选择 SIMATIC PLC，并依次点开各个 PLC 的 CPU 类型，最终选择本案例所选用的 6ES7 214-1AE30-0XB0。

点击“确定”后，就会出现图 1-33 所示的完整设备视图。

第1章 S7-1200 PLC入门

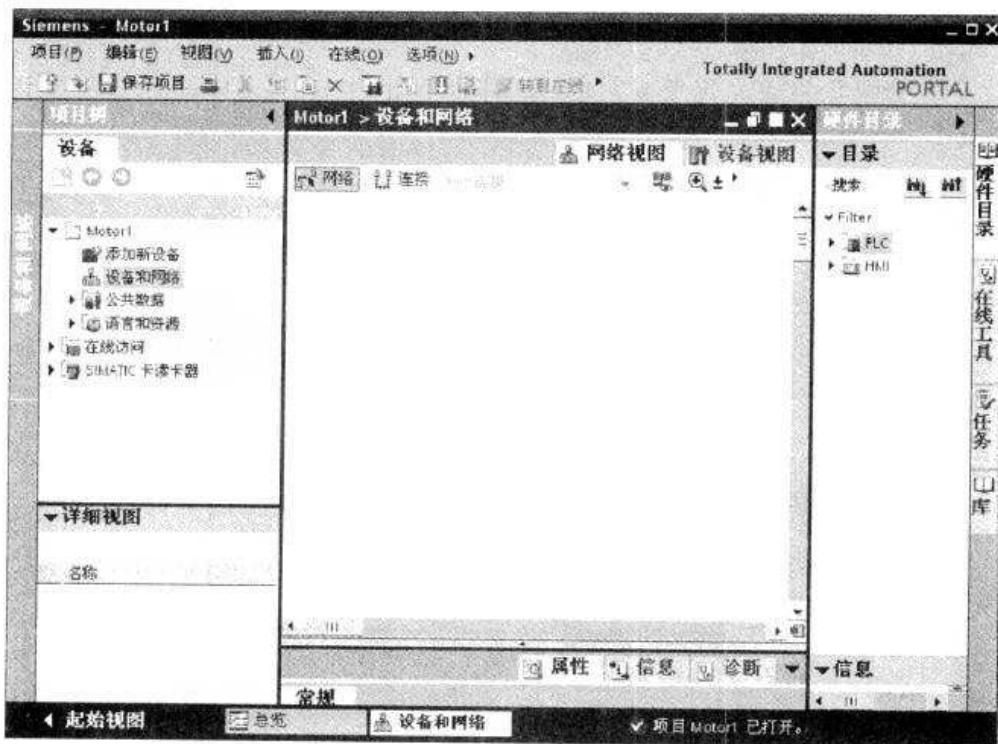


图 1-31 项目视图总览

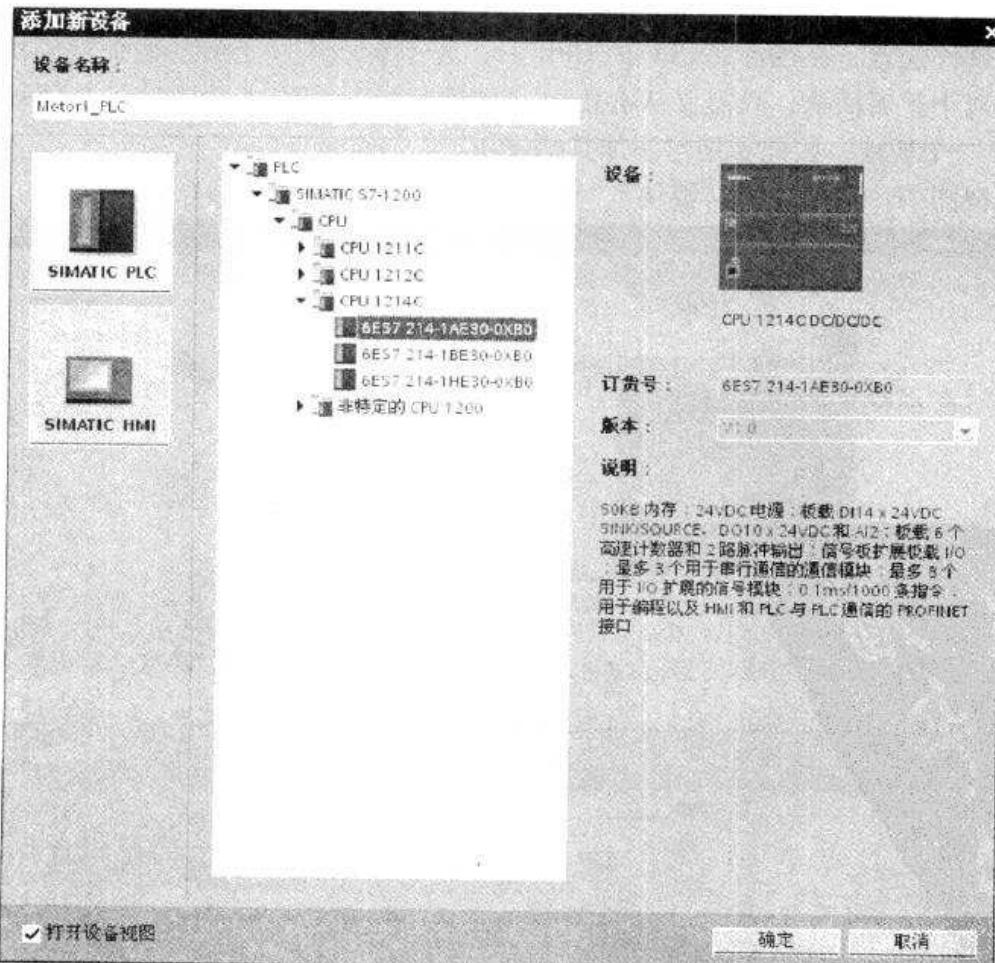


图 1-32 添加新设备



图解西门子 S7-1200 PLC 入门到实践

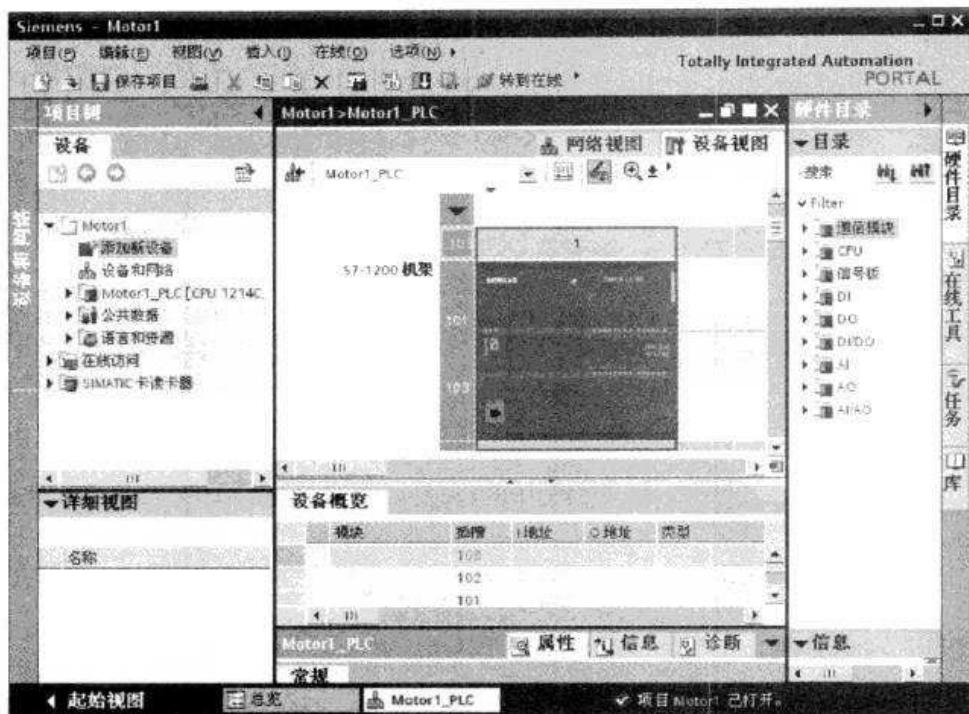


图 1-33 完整设备视图

4) 定义设备属性，完成硬件配置。

要完成硬件配置，在选择完 PLC 的 CPU 后，还需要添加和定义其他扩展模块、网络等重要信息。对于扩展模块，只需要从右边的“硬件目录”中拖入相应的模块即可。本案例只用到 CPU 一个模块，因此不用再添加其他模块。

在设备视图中，点击“CPU 模块”，就会出现 CPU 的属性窗口（见图 1-34），可以将该



图 1-34 CPU 的属性窗口

第1章 S7-1200 PLC 入门



窗口拉高，这样就会出现图 1-35 所示的 CPU 常规属性，包括项目信息和目录信息。

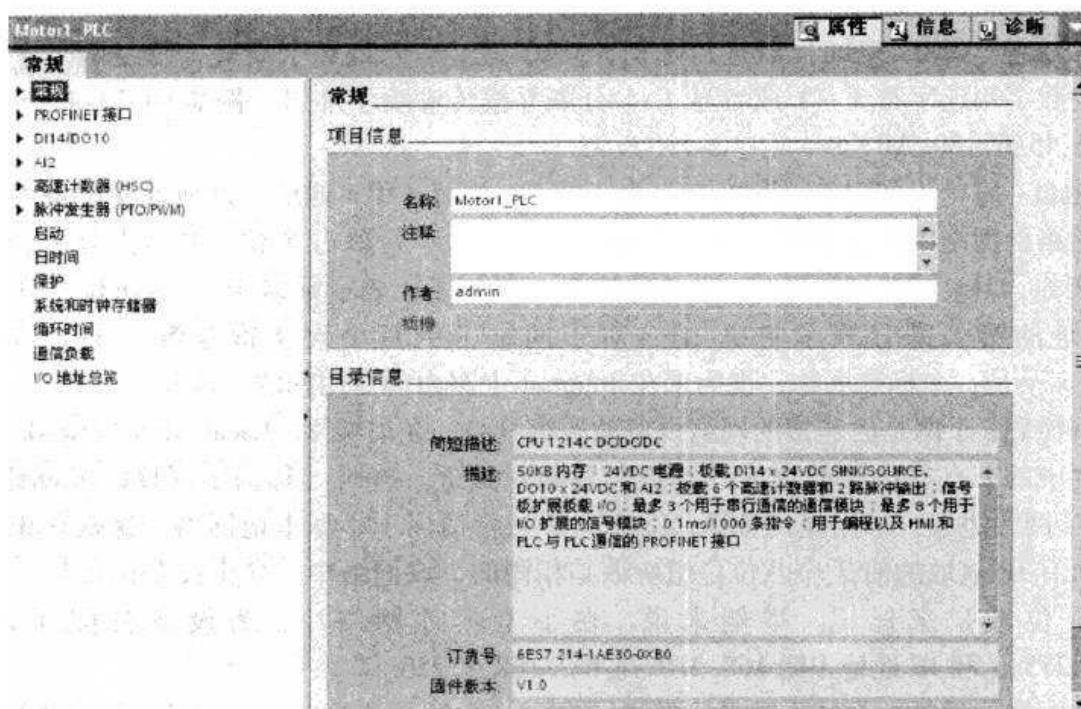


图 1-35 CPU 的常规属性

因为 CPU 没有预组态的 IP 地址，所以必须手动分配 IP 地址。如图 1-36 所示，在组态 CPU 的属性时，组态 PROFINET 接口的 IP 地址与其他参数。在 PROFINET 网络中，制造商为每个设备都分配一个惟一的“介质访问控制”地址（MAC 地址）以进行标识。每个设备也都必须具有一个 IP 地址。

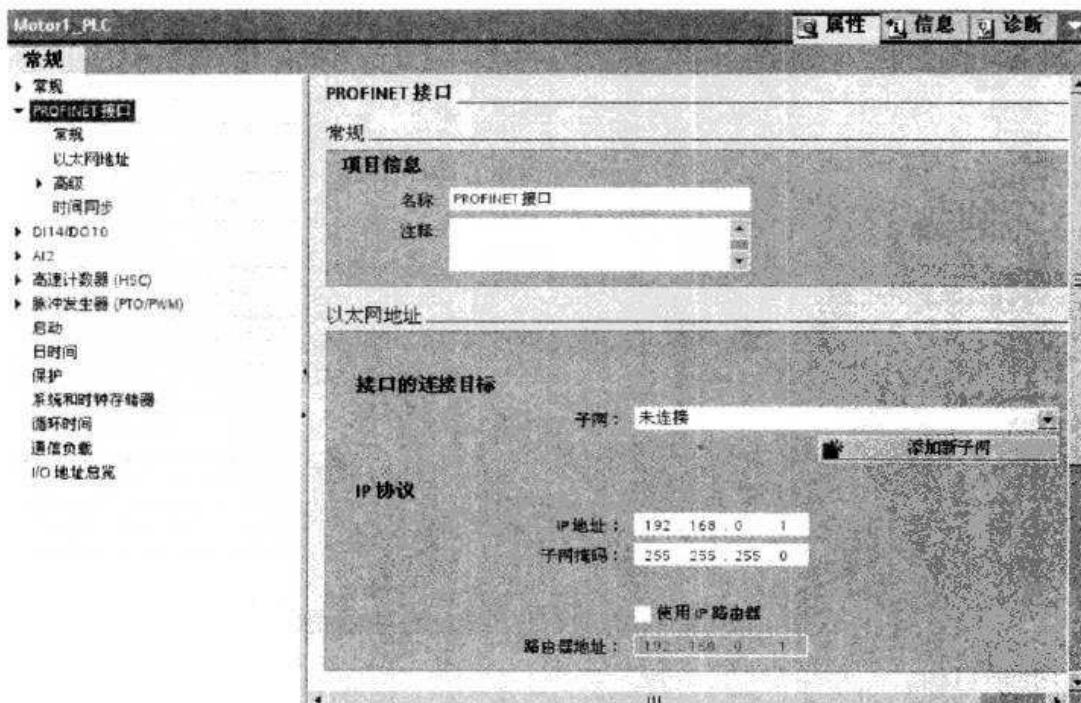


图 1-36 PROFINET 接口属性



下面对以太网地址、IP 地址和子网掩码进行介绍：

以太网（MAC）地址：在 PROFINET 网络中，制造商会为每个设备都分配一个“介质访问控制”地址（MAC 地址）以进行标识。MAC 地址由六组数字组成，每组两个十六进制数，这些数字用连字符（-）或冒号（:）分隔并按传输顺序排列（例如 01-23-45-67-89-AB 或 01:23:45:67:89:AB）。

IP 地址：每个设备也都必须具有一个 Internet 协议（IP）地址。该地址使设备可以在更加复杂的路由网络中传送数据。每个 IP 地址分为四段，每段占 8 位，并以点分十进制格式表示（例如，211.154.184.16）。IP 地址的第一部分用于表示网络 ID（您正位于什么网络中），地址的第二部分表示主机 ID（对于网络中的每个设备都是惟一的）。IP 地址 192.168.x.y 是一个标准名称，视为未在 Internet 上路由的专用网的一部分。

子网掩码：子网是已连接的网络设备的逻辑分组。在局域网（Local Area Network, LAN）中，子网中的节点往往彼此之间的物理位置相对接近。掩码（称为子网掩码或网络掩码）定义 IP 子网的边界。子网掩码 255.255.255.0 通常适用于小型本地网络。这就意味着此网络中的所有 IP 地址的前 3 个八位位组应该是相同的，该网络中的各个设备由最后一个八位位组（8 位域）来标识。举例来说，在小型本地网络中，为设备分配子网掩码 255.255.255.0 和 IP 地址 192.168.2.0 到 192.168.2.255。

硬件配置的另外一个特点就是：灵活、自由（包括寻址的自由）。在以往的 S7-200 系列 PLC 中，CPU 及扩展模块的寻址是固定的，但是 S7-1200 系列 PLC 则提供了自由地址的功能，如图 1-37 所示，它可以对 I/O 地址进行起始地址的自由选择，如 0~1023 均可以。

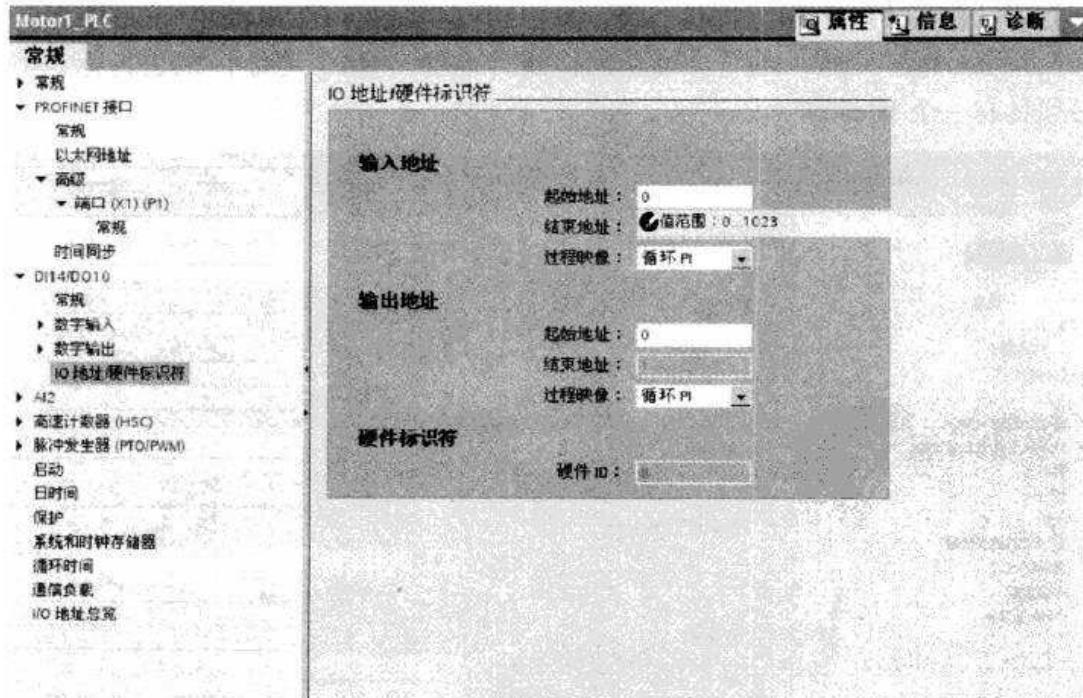


图 1-37 I/O 地址进行起始地址的自由选择

5) 打开项目树。

如图 1-38 所示为项目树全貌。对于 S7-1200 PLC 和人机界面，其项目树都是统一的。即

第1章 S7-1200 PLC入门



使在复杂的工程组态项目中，项目树仍然可以保持清晰的结构。因此，用户可以在组态自动化任务时快速访问相关设备、文件夹或特定的视图。

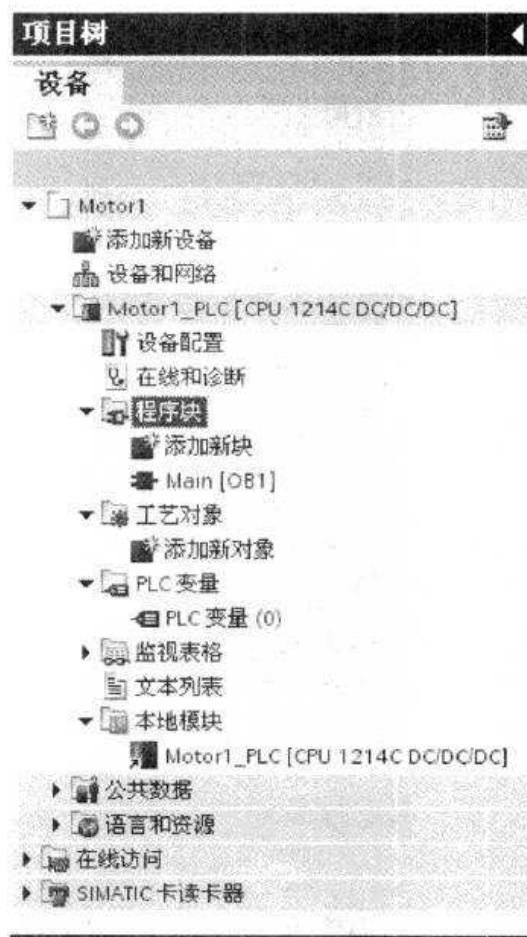


图 1-38 项目树全貌

6) 变量定义。

变量是 PLC I/O 和地址的符号名称。用户创建 PLC 变量后，TIA 软件将变量存储在变量表中。项目中的所有编辑器（例如程序编辑器、设备编辑器、可视化编辑器和监视表格编辑器）均可访问该变量表。

在项目树中，单击“PLC 变量”就可以创建本案例所需要用到的变量，具体使用 3 个变量，分别是“启动按钮”、“停止按钮”和“接触器”（见图 1-39）。需要注意的是，这里采用默认数据类型为 Bool，即布尔量（具体数据类型将在项目 2 中进行介绍）。

| PLC 变量 | | | | |
|--------|------|------|-------|-----|
| | 名称 | 数据类型 | 地址 | 保択性 |
| 1 | 启动按钮 | Bool | %M0.0 | |
| 2 | 停止按钮 | Bool | %M0.1 | |
| 3 | 接触器 | Bool | %Q0.0 | |

图 1-39 变量定义



7) 梯形图编程。

TIA 软件提供了包含各种程序指令的指令窗口（见图 1-40），共包括收藏夹、指令和扩展指令，同时这些指令按功能分组（如常规、位逻辑运算、定时器操作等）。



图 1-40 指令窗口

用户要创建程序，只需将指令从任务卡中拖动到程序段即可。比如本案例，先要使用常开触点时，从收藏夹将常开触点直接拉入程序段 1。如图 1-41 所示，程序段 1 出现 符号，

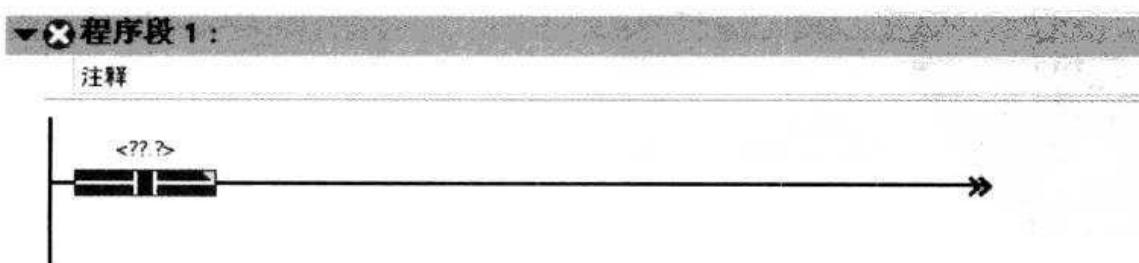


图 1-41 程序段编辑一



标识该程序段处于语法错误状态。

TIA 软件与 S7-200 PLC 的 Micro/WIN 编程环境相比，优势在于其指令编辑的可选择性，比如单击功能框指令黄色角以显示指令的下拉列表，比如常开、常闭、P 触点（上升沿）、N 触点（下降沿），向下滚动列表并选择常开指令（见图 1-42）。

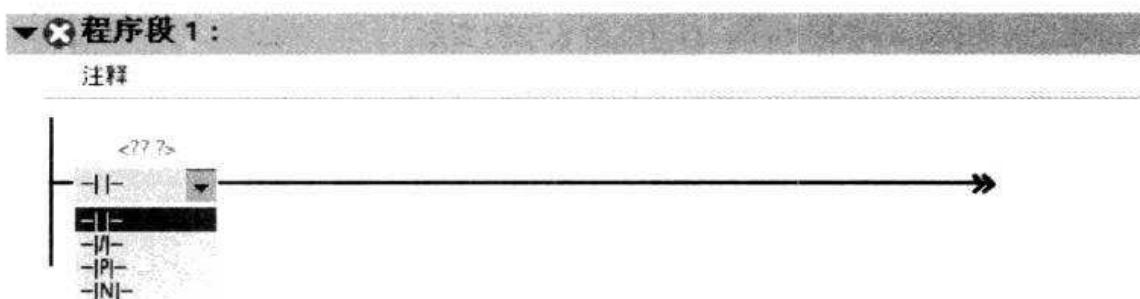


图 1-42 显示指令的下拉列表

在选择完具体的指令后，必须输入具体的变量名，最基本的方法就是：双击第一个常开触点上方的默认地址“<?? . ? >”，直接输入固定地址变量“%I0.1”，这时就会出现图 1-43 所示的“停止按钮 %I0.1”注释。

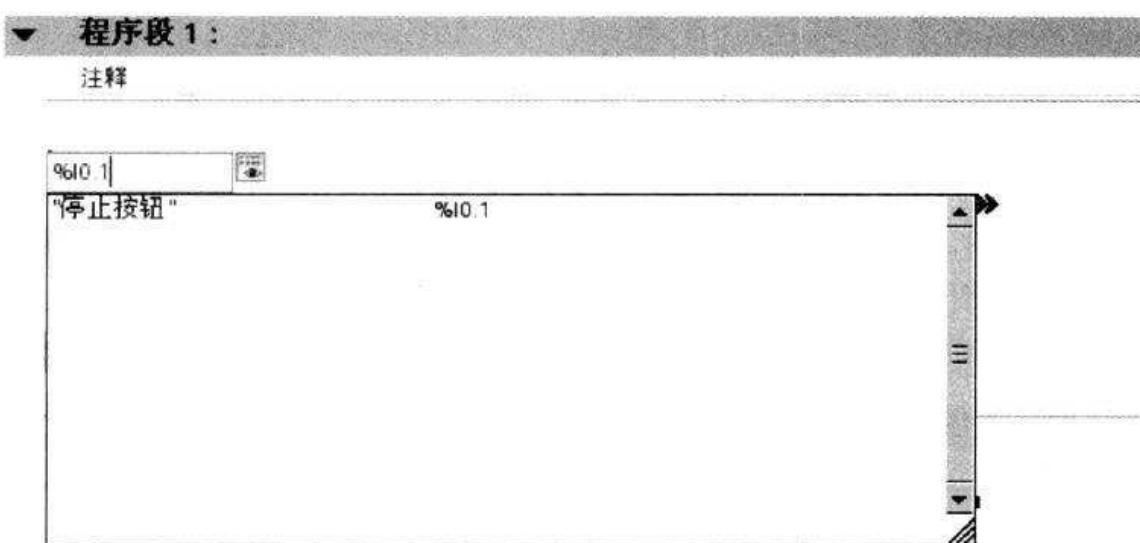


图 1-43 使用固定地址输入变量

需要引起注意的是：与 S7-200 PLC 不同，TIA 软件默认的就是 IEC 61131-3 标准，其地址用特殊字母序列来指示，字母序列的起始用%符号，跟随一个范围前缀和一个数据前缀（数据类型）表示数据长度，最后是数字序列表示存储器的位置。其中，范围前缀：I（输入）、Q（输出）、M（标志，内部存储器范围）；长度前缀：X（单个位）、B（字节，8 位）、W（字，16 位）、D（双字，32 位）。

比如：

%MB7 标志字节 7

%MW1 标志字 1



% MD3 标志双字 3

% I0.0 输入位 I0.0

除了使用固定地址外，还可以使用变量表，用户可以快速输入对应触点和线圈地址的 PLC 变量（见图 1-44）。具体步骤如下：

双击第一个常开触点上方的默认地址“<?? . ? >”；

单击地址右侧的选择器图标 ，打开变量表中的变量；

从下拉列表中，为第一个触点选择“停止按钮 % I0.1”。

▼ 程序段 1：

注释



图 1-44 使用变量表输入变量

根据以上规则，输入第二个常开触点“% I0.0”，并根据梯形图的编辑规律，使用图符 打开分支（见图 1-45），输入接触器自保触点“% Q0.0”。最后使用图符 关闭分支（见图 1-46），使用图符 选择输出触点“% Q0.0”。

▼ 程序段 1：

注释

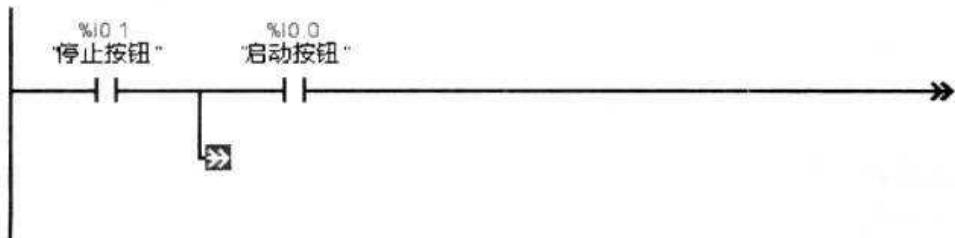


图 1-45 程序段编辑二

完成以上编辑后，就会发现程序段 1 的 符号不见了。当然，根据图 1-46 所示的梯形图，与电气控制图 1-25 相比，还是有一点疑惑：为什么停止按钮% I0.1 使用的是常开输入，而不是传统继电器控制中的常闭呢？这一点是从 PLC 的常开和常闭的定义出发，对于 PLC 输入来讲，常开不是根据用户接线是什么触点（包括常开或常闭），而是根据程序的逻辑出



程序段 1：

注释

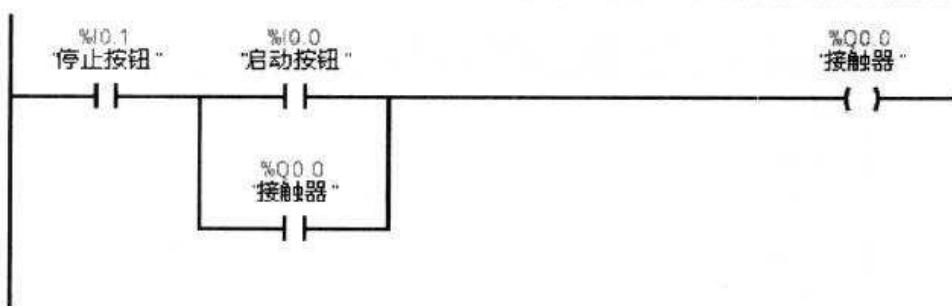


图 1-46 程序段编辑三

发，具体将在下面的相关项目中进行介绍。

8) 程序下载前的准备。

由于 S7-1200 PLC 采用常规以太网 RJ45 接口，因此必须了解程序下载前需要准备的步骤：第一，需要选择或制作一根以太网线；第二，需要在 PC 和 PLC 端设置相同频段的 IP 地址。

PLC 或 PC 的 RJ45 接口外观为 8 芯母插座，如图 1-47 所示；而网线则为 8 芯公插头（见图 1-48）

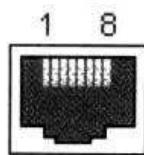


图 1-47 RJ45 母插座

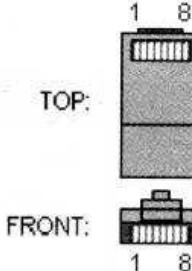


图 1-48 RJ45 公插头

在 100 兆以太网络中，实际只应用了 4 根线来传输数据，另外 4 根是备份的（见表 1-1）。传输的信号为数字信号，双绞线最大传输距离为 100m。

表 1-1 以太网线的定义

| 1 | TX+ | Transceive Data+ (发信号+) |
|---|-----|-------------------------|
| 2 | TX- | Transceive Data- (发信号-) |
| 3 | RX+ | Receive Data+ (收信号+) |
| 4 | n/c | Not connected (空脚) |
| 5 | n/c | Not connected (空脚) |
| 6 | RX- | Receive Data- (收信号-) |
| 7 | n/c | Not connected (空脚) |
| 8 | n/c | Not connected (空脚) |



图解西门子 S7-1200 PLC 入门到实践

将 IP 地址下载到 CPU 之前，必须先确保计算机的 IP 地址与 PLC 的 IP 地址相匹配。如图 1-49 所示，在计算机的本地连接属性窗口中，选择常规选项的“Internet 协议 (TCP/IP)”，将协议地址从自动获得 IP 地址改为手动设置 IP 地址，设为 192.168.0.100（见图 1-50）。

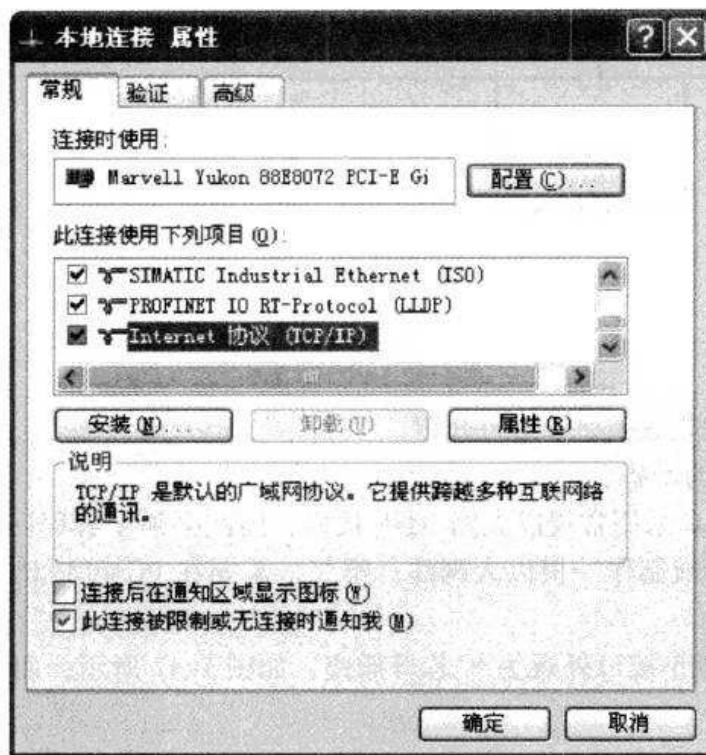


图 1-49 选择常规选项的“Internet 协议 (TCP/IP)”

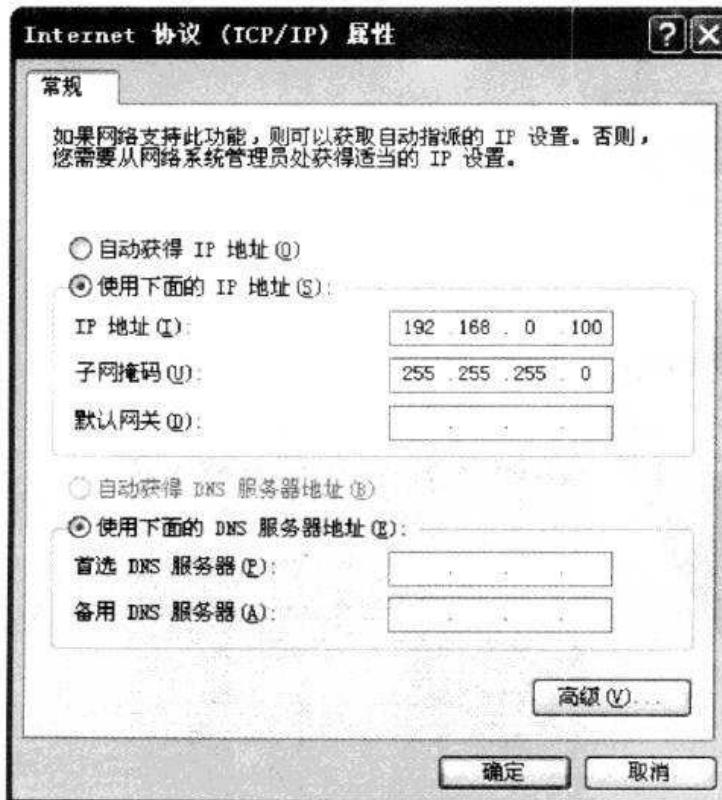


图 1-50 手动设置 IP 地址



9) 编译与下载。

在编辑阶段只是完成了基本编辑语法的输入验证，但是要完成程序的可行性还必须执行“编译”命令。在一般情况下，用户可以直接选择下载命令，TIA 软件会自动先执行编译命令。当然，也可以单独选择编译命令，如图 1-51 所示，在 TIA 软件的“编辑”菜单中选择“编译”命令，或者使用“CTRL + B”快捷键，就可获得整个程序的编译信息（见图 1-52）。



图 1-51 选择编译命令

| Main | | | | 属性 | 信息 | 诊断 |
|-----------------------------|----------------------|------|----|----|----|---------|
| 常规 | 编译 | 交叉引用 | 语法 | | | |
| 编译已完成 (错误: 0; 警告: 0) | | | | | | |
| 路径 | 描述 | | | 错误 | 警告 | 时间 |
| Motor1_PLC | 编译已完成 (错误: 0; 警告: 0) | | | 0 | 0 | 9:17:32 |
| | | | | 0 | 0 | 9:17:33 |

图 1-52 编译信息

在编译完成后，就可以将 S7-1200 PLC 的硬件配置和梯形图软件下载。可以选择两个命令下载，即“下载到设备”或“扩展的下载到设备”。

如图 1-53 所示，在“在线”菜单中选择“下载到设备”，这时 PLC 会提示：是否要停止所选的 PLC？（见图 1-54）对于调试中的程序来说，这个信息并不重要，但是对于在运行中的设备来说，这个选择将直接关系到设备的安全，因此必须在确保安全的前提下进行。

在下载过程中，还会碰到如图 1-55 所示的下载预览和图 1-56 所示的装载结果。



图解西门子 S7-1200 PLC 入门到实践



图 1-53 选择“下载到设备”命令

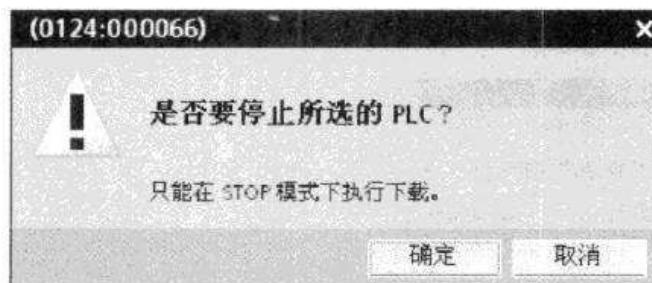


图 1-54 下载之前的提示

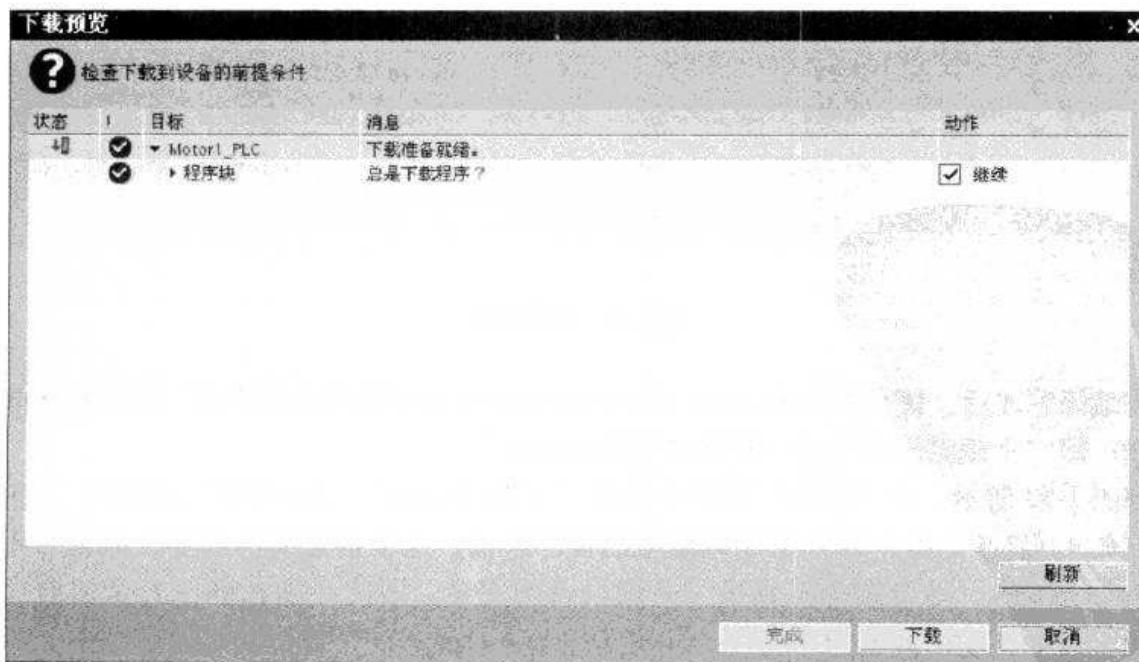


图 1-55 下载预览

第1章 S7-1200 PLC入门

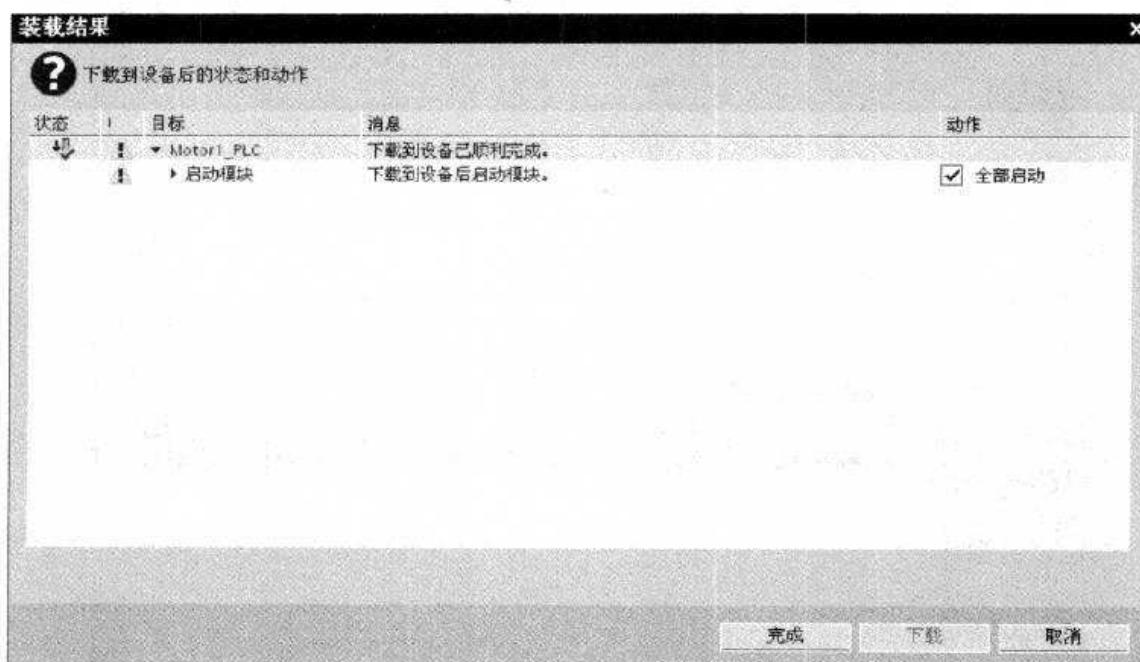


图 1-56 装载结果

为了更加透明地了解下载的过程，可以选择“扩展的下载到设备”命令（见图 1-57），这时就会出现图 1-58 所示的以太网联网情况分析，从中不仅可以看到程序中的 PLC 地址，看到用于 PC 连接的 PG/PC 接口情况（这对于多网卡用户来说非常重要），还可以看到目标子网中的所有设备。当用户选择指定的设备时，点击 闪烁 LED 图符，就会看到实际设备会黄灯闪烁，以让用户确定该设备是否需要进行配置和程序下载。



图 1-57 选择“扩展的下载到设备”命令



图解西门子 S7-1200 PLC 入门到实践

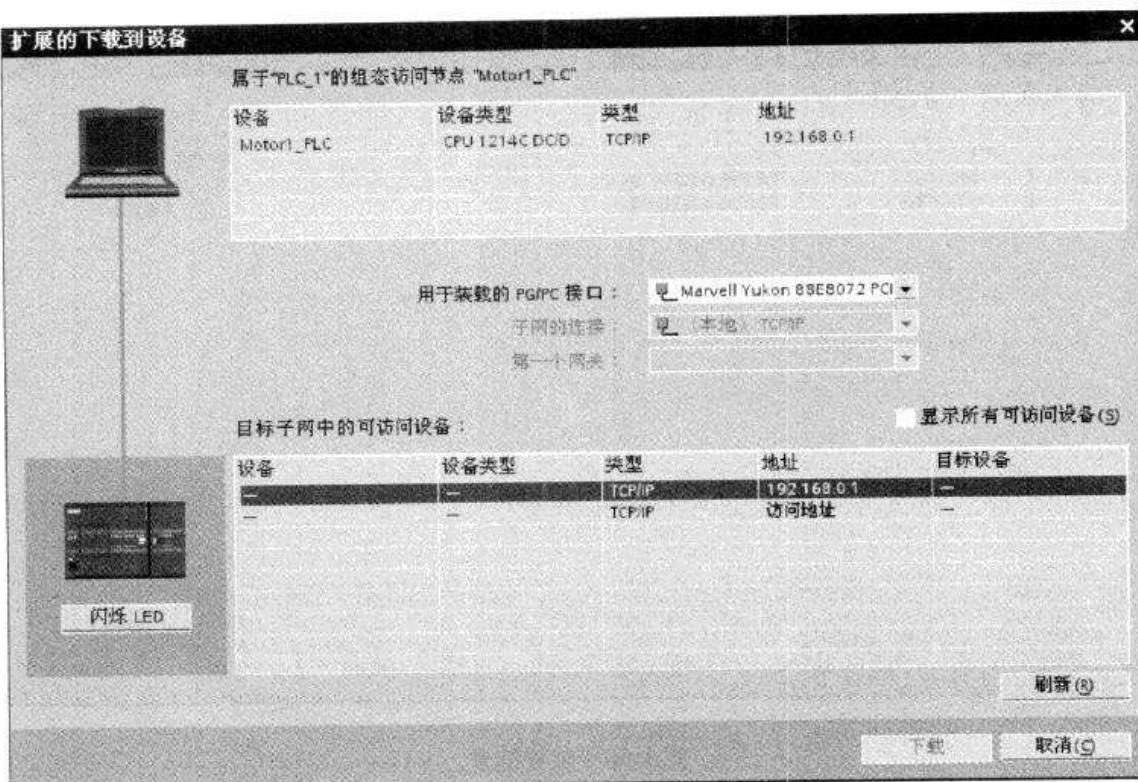


图 1-58 以太网联网情况分析

如果编程计算机的 IP 地址与下载设备的 IP 地址不一致，就会跳出“分配 IP 地址”的窗口（见图 1-59），暂时给计算机 IP 地址进行配置，并及时反馈分配 IP 地址后的信息（见图 1-60）。

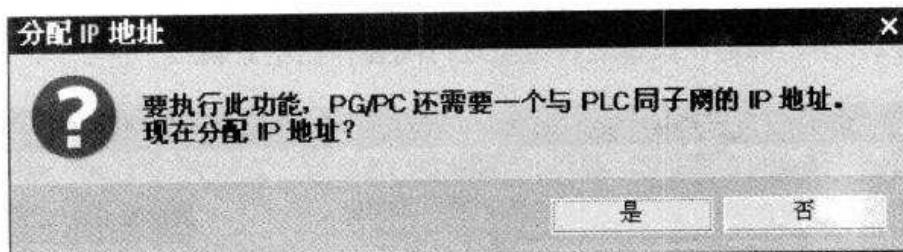


图 1-59 分配 IP 地址

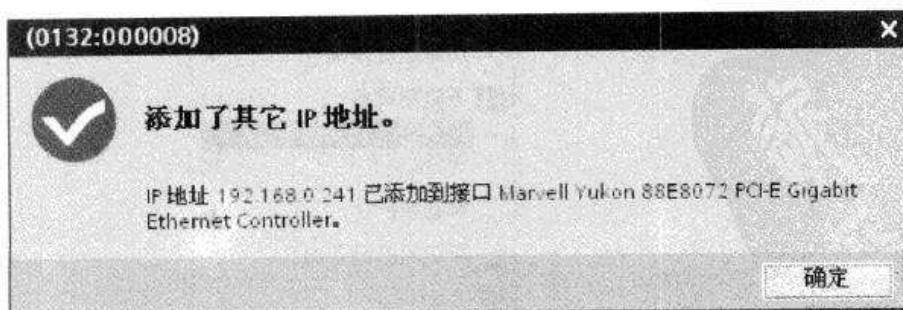


图 1-60 添加 IP 地址后的信息

10) PLC 在线与程序调试。

在 PLC 的程序与配置下载后，就可以将 PLC 切换到运行状态运行。但是，很多时候用户需要详细了解 PLC 的实际运行情况，并对程序进行一步步的调试，这就要进入“PLC 在线与程序调试”阶段。



图 1-61 所示为“转到在线”的命令选择。



图 1-61 选择“转到在线”

转到在线后，项目树就会显示黄色的■图符，动画过程就是表示在线状态，这时可以从项目树各个选项后面了解其各自的情况（见图 1-62），出现蓝色的✓和●图符表示为正



图 1-62 项目树的在线阶段



常，否则必须进行诊断或重新下载。

在本案例中，选择程序块的在线仿真（见图 1-63），选择图符，即可进入仿真阶段，分别为绿色实线表示接通，蓝色虚线表示断开。从图中可以看到停止按钮% I0.1 常开触点为接通状态，这也解释了在编辑阶段为何输入常开而不是常闭的原因。当启动按钮% I0.0 按下时，程序进入自保阶段（见图 1-64）。

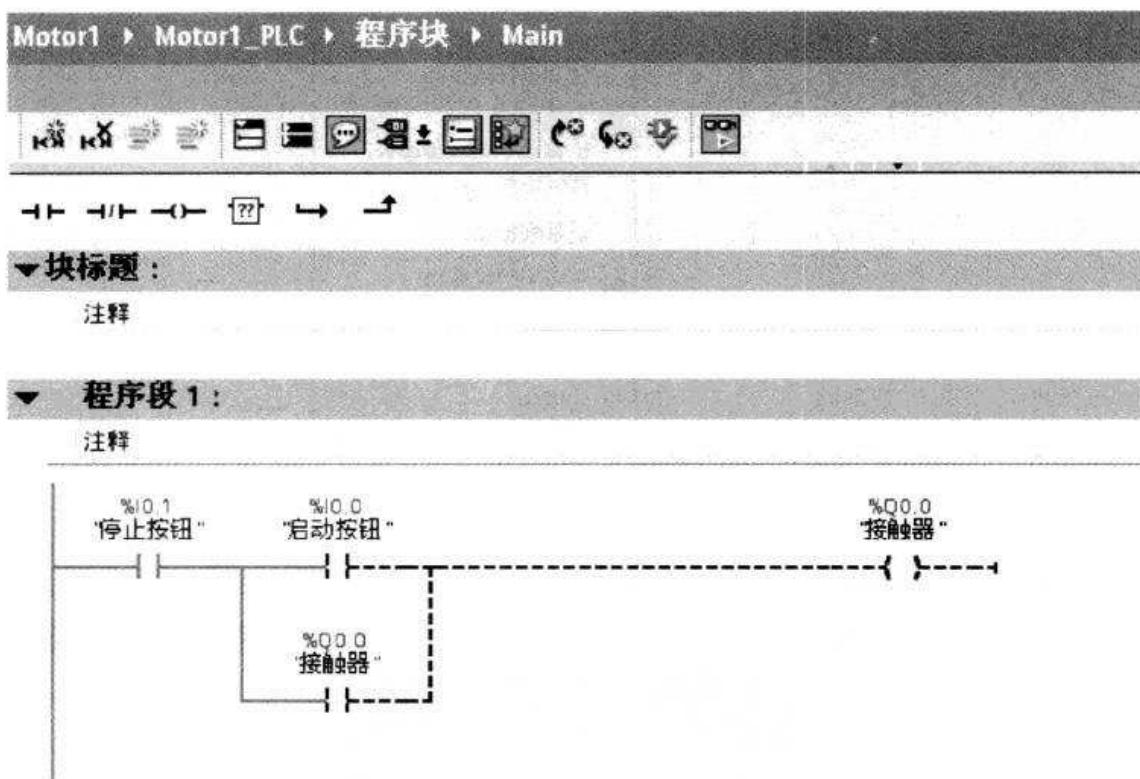


图 1-63 程序块的在线仿真一

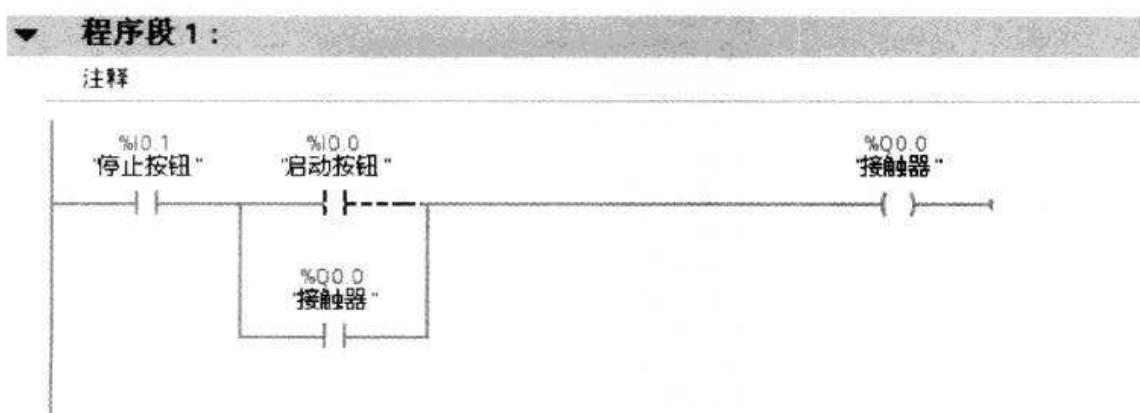


图 1-64 程序块的在线仿真二

当然，PLC 变量还可以进行在线仿真，如图 1-65 所示，选择即可看到最新的监视值。

在项目树中，选择“在线访问”即可看到图 1-66 所示的各种信息，如诊断状态、循环时间、存储器、分配 IP 地址等。



| PLC 变量 | | | | | | 常量 |
|--------|------|-------|-------------------------------------|-------|----|----|
| 名称 | 数据类型 | 地址 | 保持性 | 监视值 | 注释 | |
| 1 启动按钮 | Bool | %I0.0 | <input checked="" type="checkbox"/> | FALSE | | |
| 2 停止按钮 | Bool | %I0.1 | <input checked="" type="checkbox"/> | TRUE | | |
| 3 接触器 | Bool | %Q0.0 | <input checked="" type="checkbox"/> | FALSE | | |

图 1-65 PLC 变量的在线仿真

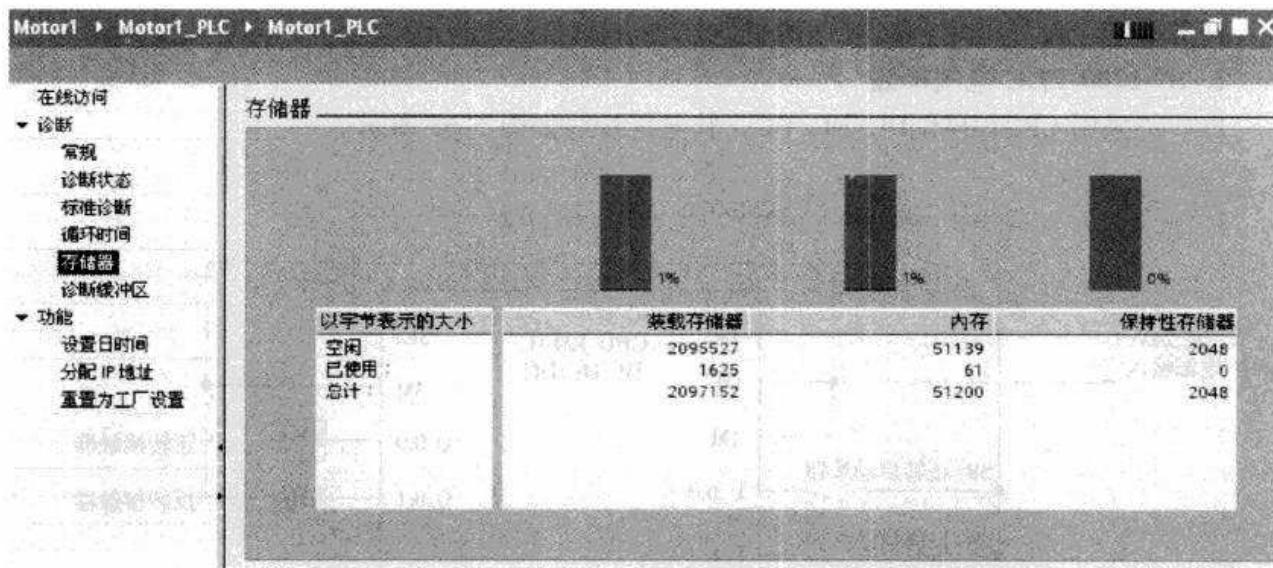


图 1-66 在线访问所示的信息

1.4.2 电动机正反转 PLC 控制

1. 任务说明

三相电动机的接触器联锁的正反转控制电路如图 1-67 所示。电路中采用 KM₁ 和 KM₂ 两

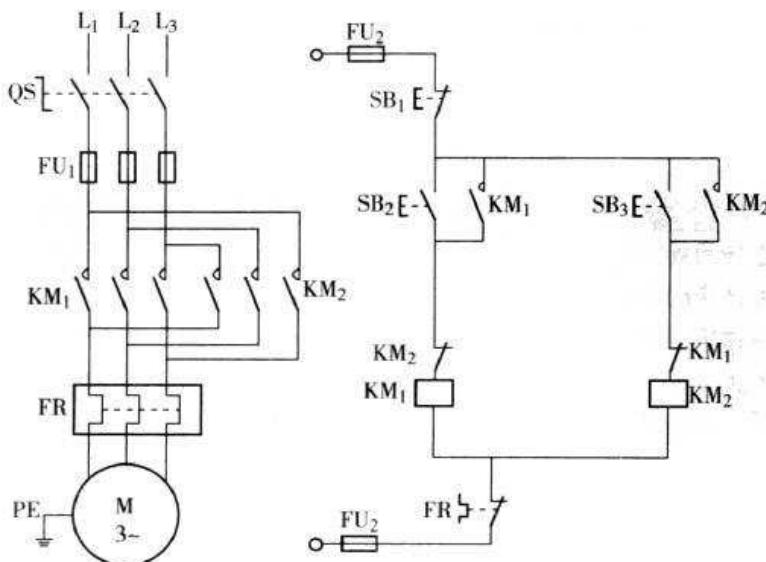


图 1-67 三相电动机的接触器联锁的正反转控制电路



个接触器，当 KM_1 接通时，三相电源的相序按 $L_1-L_2-L_3$ 接入电动机。而当 KM_2 接通时，三相电源的相序按 $L_3-L_2-L_1$ 接入电动机。所以当两个接触器分别工作时，电动机的旋转方向相反。

电路要求接触器 KM_1 和 KM_2 不能同时通电，否则它们的主触头同时闭合，将造成 L_1 、 L_3 两相电源短路，为此在 KM_1 和 KM_2 线圈各自的支路中相互串接了对方的一副动断辅助触头，以保证 KM_1 和 KM_2 不会同时通电。 KM_1 和 KM_2 这两副动断辅助触头在电路中所起的作用称为联锁（或互锁）作用。

任务要求：设计采用 PLC 控制的硬件接线与软件。

2. S7-1200 PLC 电气接线

PLC 仍采用 CPU1214C DC/DC/DC，其电气接线如图 1-68 所示。

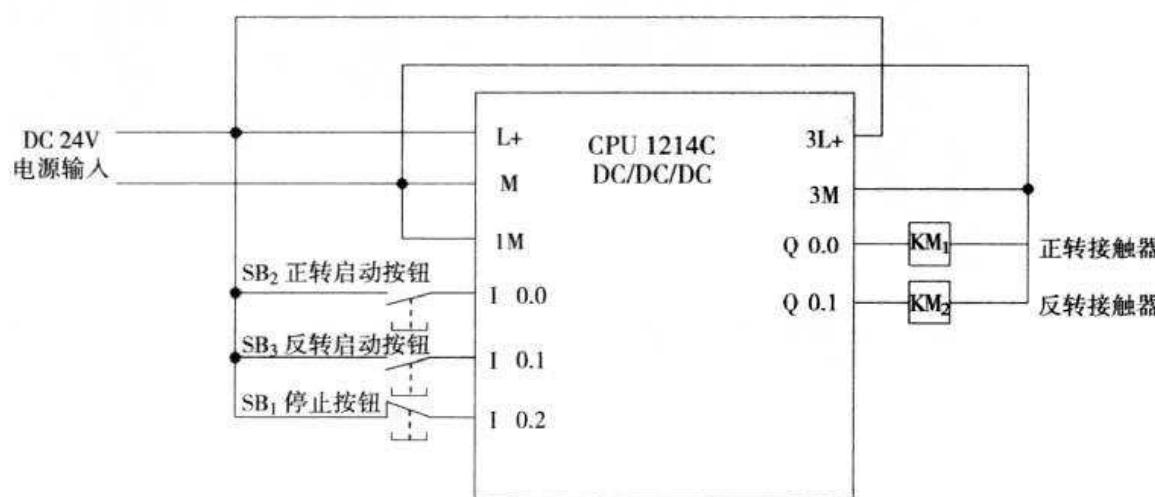


图 1-68 电动机正反转 PLC 控制电气接线

3. 硬件配置与软件编程

电动机正反转 PLC 控制的硬件配置如电动机直接起动一样，不再赘述。正反转控制 PLC 变量见表 1-2。

表 1-2 正反转控制 PLC 变量

| | 名称 | 数据类型 | 地址 |
|---|--------|------|-------|
| 1 | 正转启动按钮 | Bool | %I0.0 |
| 2 | 反转启动按钮 | Bool | %I0.1 |
| 3 | 正转接触器 | Bool | %Q0.0 |
| 4 | 停止按钮 | Bool | %I0.2 |
| 5 | 反转接触器 | Bool | %Q0.1 |

正反转 PLC 控制梯形图如图 1-69 所示。

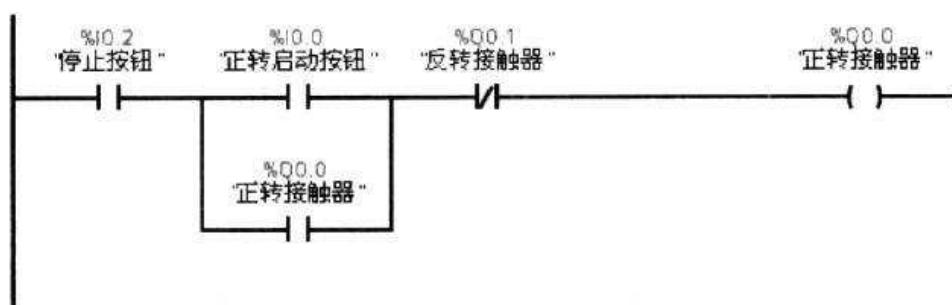


▼ 块标题：

电机正反转控制

▼ 程序段 1：

正转启动与停止



▼ 程序段 2：

反转启动与停止

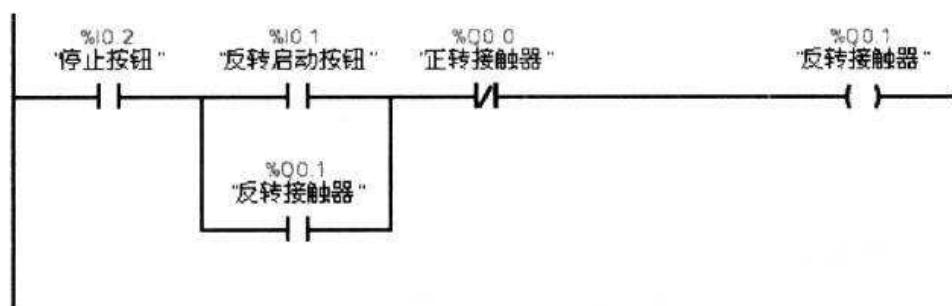


图 1-69 正反转 PLC 控制梯形图

1.4.3 三相电动机的星-三角启动 PLC 控制

1. 任务说明

在电气控制中，利用时间继电器可以实现星减压启动的自动控制，典型电路如图 1-70

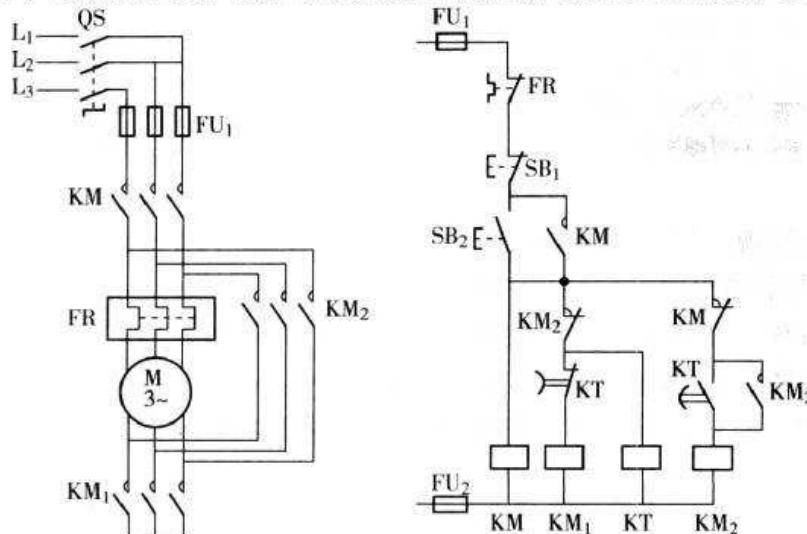


图 1-70 星-三角启动的典型电路



所示，其中应用延时继电器 KT 进行切换。

任务要求：设计采用 PLC 控制的硬件接线与软件。

2. S7-1200 PLC 电气接线

在电气控制中，利用时间继电器可以实现星减压起动的自动控制，星-三角启动 PLC 硬件接线如图 1-71 所示，其中应用延时继电器 KT 进行切换。

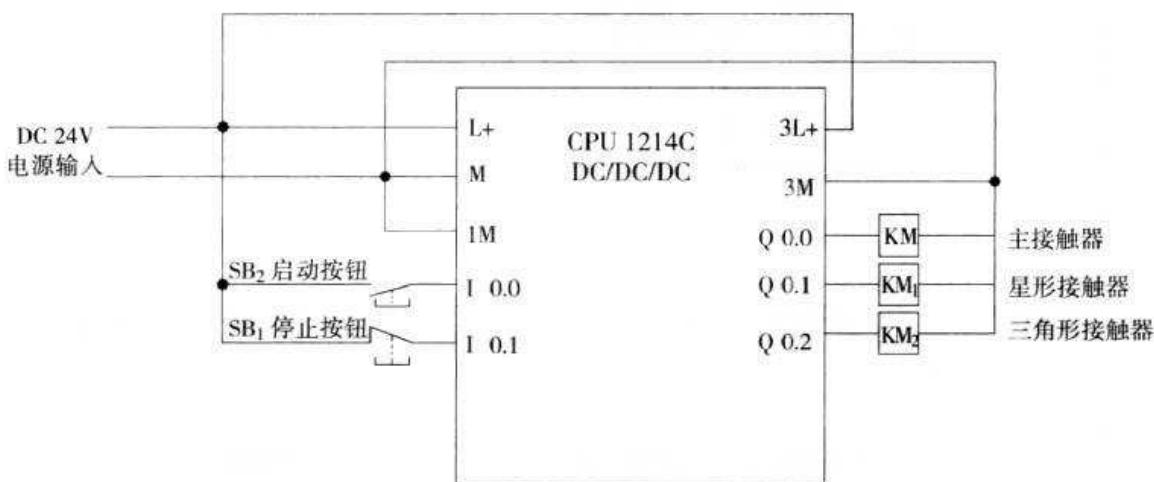


图 1-71 星-三角启动 PLC 硬件接线

3. 硬件配置与软件编程

本案例的硬件配置与以上两个案例相同。先进行 PLC 变量的编写，见表 1-3。

表 1-3 星-三角启动的 PLC 变量

| | 名称 | 数据类型 | 地址 |
|---|--------------|------|-------|
| 1 | 启动按钮 | Bool | %I0.0 |
| 2 | 停止按钮 | Bool | %I0.1 |
| 3 | 主接触器 KM | Bool | %Q0.0 |
| 4 | 星形启动接触器 KM1 | Bool | %Q0.1 |
| 5 | 三角形启动接触器 KM2 | Bool | %Q0.2 |
| 6 | 延时继电器 | Bool | %M0.0 |

从表 1-3 中会发现，只有延时继电器，而没有定时器变量，这是因为 S7-1200 PLC 与其他的小型 PLC 编程不尽相同（包括 S7-200 PLC）。

在本案例编程中，还需要引入新的概念，即数据块（DB）。在 S7-1200 PLC 编程中，用户程序中创建 DB 是用来存储代码块的数据，它分为全局 DB 和背景 DB 两种，其中用户程序中的所有程序块都可访问全局 DB 中的数据，而背景 DB 仅存储特定功能块（FB）的数据。

为什么没有引入 DB 块呢？这是因为在 S7-1200 PLC 中，定时器是以 FB 的形式出现，而 FB 则必须为其定义 DB。至于 FB 的具体介绍也将在项目 2 中详细介绍。在本案例只是需



要重点了解定时器是如何调用 DB 的。

如图 1-72 所示，在指令窗口中选择“定时器操作”中的 TON 指令，并将它拖入到程序段中，这时就会跳出一个“调用数据块”窗口，选择自动编号，则会直接生成 DB1 数据库，如图 1-73 所示。



图 1-72 定时器指令

在项目树的“程序块”中，可以看到自动生成的 IEC_Timer_0 [DB1] 数据块（见图 1-74），双击进入，即可读取到 DB1 的定时器的各个数据，变量的数据类型为 IEC_Timer。DB1 的 IEC_Timer_0 内容见表 1-4。



图解西门子 S7-1200 PLC 入门到实践

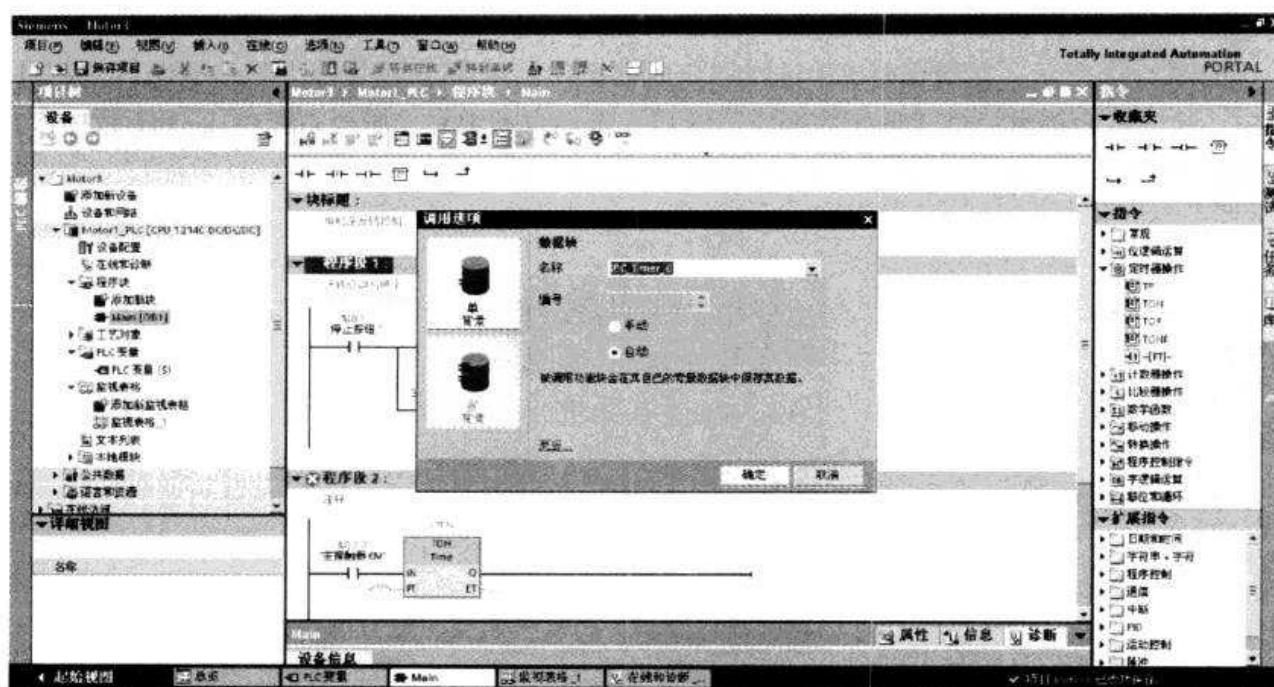


图 1-73 TON 指令调用数据块

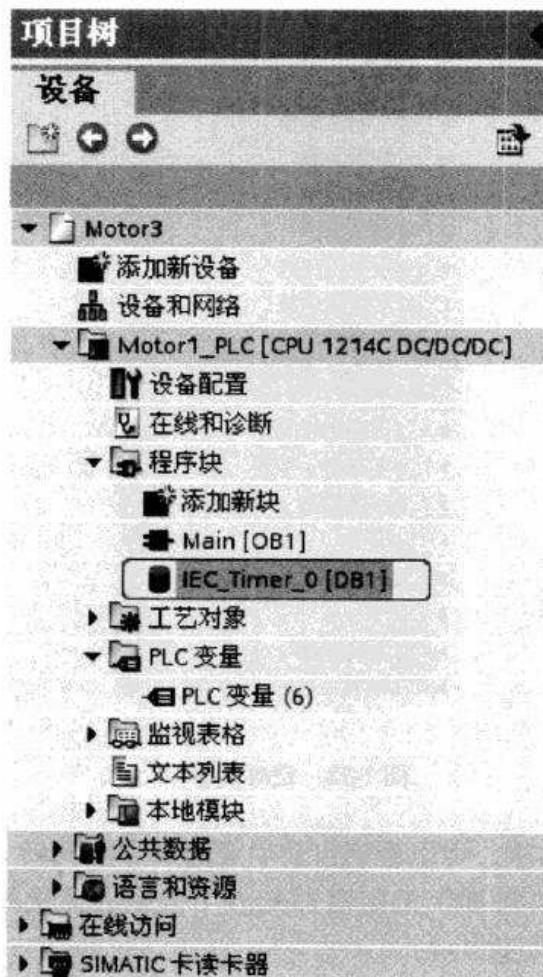


图 1-74 DB1 块的位置



表 1-4 DB1 的 IEC Timer 0 内容

| 名称 | 数据类型 | 初始值 |
|------------|------|---------|
| 1 ▼ Static | | |
| 2 START | Time | T#0ms |
| 3 PRESET | Time | T#0ms |
| 4 ELAPSED | Time | T#0ms |
| 5 RUNNING | Bool | false |
| 6 IN | Bool | false |
| 7 Q | Bool | false |
| 8 PAD | Byte | B#16#00 |
| 9 PAD_1 | Byte | B#16#00 |
| 10 PAD_2 | Byte | B#16#00 |

TON 指令就是接通延迟定时器输出 Q 在预设的延时过后设置为 ON，其指令形式如图 1-75 所示，参数及其数据类型见表 1-5。在表 1-5 中，R 参数一般用于 TONR 等指令。参数 IN 从 0 跳变为 1 将启动定时器 TON。

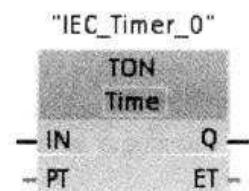


图 1-75 TON 指令形式

表 1-5 TON 参数及数据类型

| 参数 | 数据类型 | 说明 |
|--------|------|-------------------|
| IN | Bool | 启用定时器输入 |
| R | Bool | 将 TONR 经过的时间重置为零 |
| PT | Bool | 预设的时间值输入 |
| Q | Bool | 定时器输出 |
| ET | Time | 经过的时间值输出 |
| 定时器数据块 | DB | 指定要使用 RT 指令复位的定时器 |

PT（预设时间）和 ET（经过的时间）值以表示毫秒时间的有符号双精度整数形式存储在存储器中（见表 1-6）。TIME 数据使用 T# 标识符，可以简化时间单元“T#200ms”或复合时间单元“T#2s_200ms”的形式输入。

表 1-6 TIME 数据类型

| 数据类型 | 大小 | 有效数值范围 |
|------|--------------|--|
| TIME | 32 位 存储形式 | T# -24d_20h_31m_23s_648ms 到 T#24d_20h_31m_23s_647ms -2, 147, 483, 648ms 到 +2, 147, 483, 647ms |

TON 指令的时序图如图 1-76 所示。

有了以上 TON 指令的基础后，就可以进行星-三角形的定时器控制编程了，详细的梯形图如图 1-77 所示。

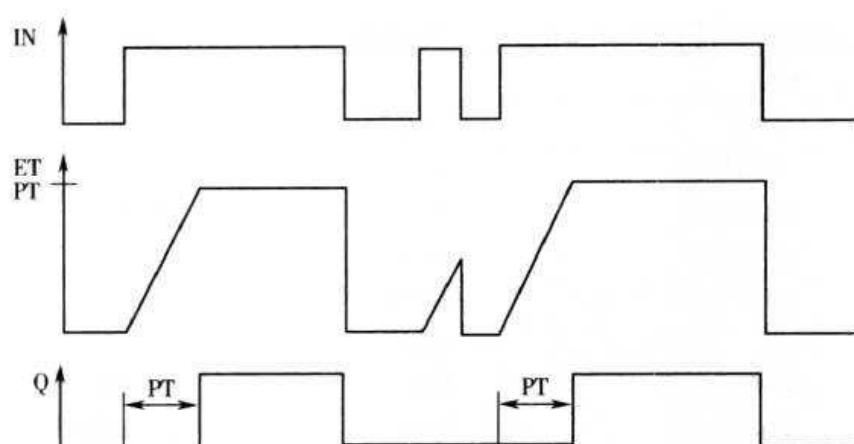


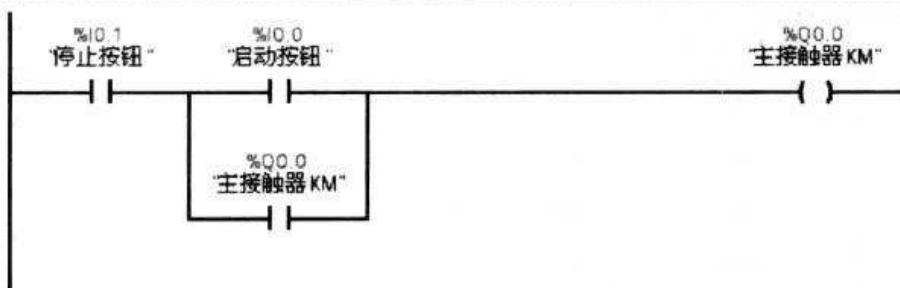
图 1-76 TON 指令的时序图

▼ 块标题：

电动机星-三角控制

▼ 程序段 1：

主接触器启动与停止



▼ 程序段 2：

三角形启动延时

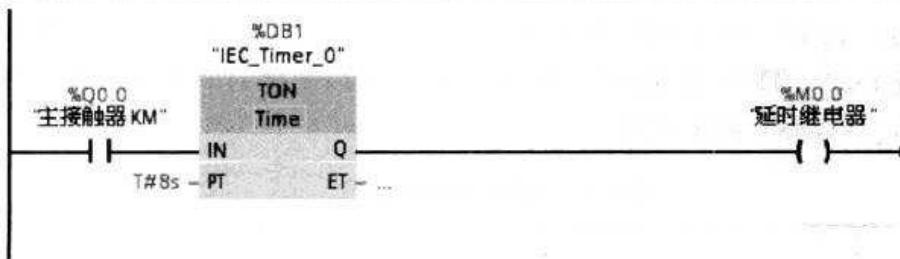


图 1-77 星-三角起动的 PLC 梯形图

将硬件配置和程序下载到 PLC 后，对定时器的调试如图 1-78 所示，用户可以实时地看到定时器的变化。



▼ 程序段 2：

三角形启动延时

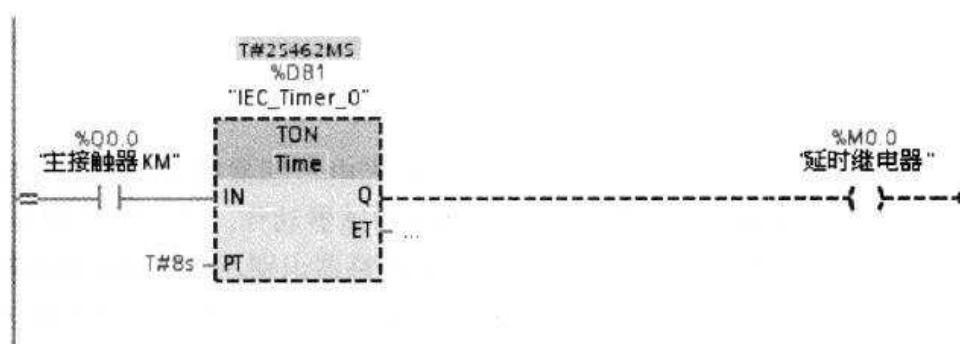


图 1-78 定时器调试

4. 定时器编程的拓展

从上述编程中已经了解到关于定时器的基本应用，即用户直接为定时器指定单一背景数据块，此数据块仅包括一个 IEC_Timer 类型的变量，优点是易于用户区分多个定时器；缺点是当使用多个定时器时，会导致出现多个独立的数据块，程序结构显得零散。

为了解决这个问题，可以在全局数据块中定义一个 IEC_Timer 类型的变量，以供定时器使用，其优点是不会因为使用多个定时器而增加多个数据块。表 1-7 为本案例增加的一个全局数据块（DB1）的内容，并可以在原来的程序中（程序段 2）修改为如图 1-79 所示的程序。

表 1-7 全局数据块（DB1）的内容

| 名称 | 数据类型 | 初始值 |
|-----------------|-----------|-----|
| 1 ▼ Static | | |
| 2 ▶ Motor_Time1 | IEC_Timer | ▼ |
| 3 ▶ Motor_Time2 | IEC_Timer | |
| 4 ▶ Motor_Time3 | IEC_Timer | |

▼ 程序段 2：

三角形启动延时

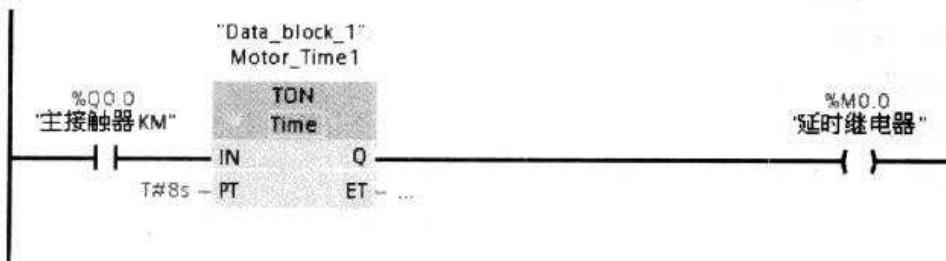


图 1-79 使用全局数据块的 TON 指令



1.5 工程案例：自动分拣装置的 PLC 控制

1.5.1 案例介绍

1. 组成部分

金属与非金属自动分拣装置如图 1-80 所示，当落料光电传感器检测到有物料后，马上起动输送电动机；当物料经过推料一位置，如果电感式传感器动作，则说明该物料为金属，气缸动作将物料推入到金属料槽中；当物料未被电感式传感器识别，而被输送到推料二位置时，此时如果电容式传感器动作，则说明物料必定为非金属物料，气缸动作将其推入到塑料料槽中。

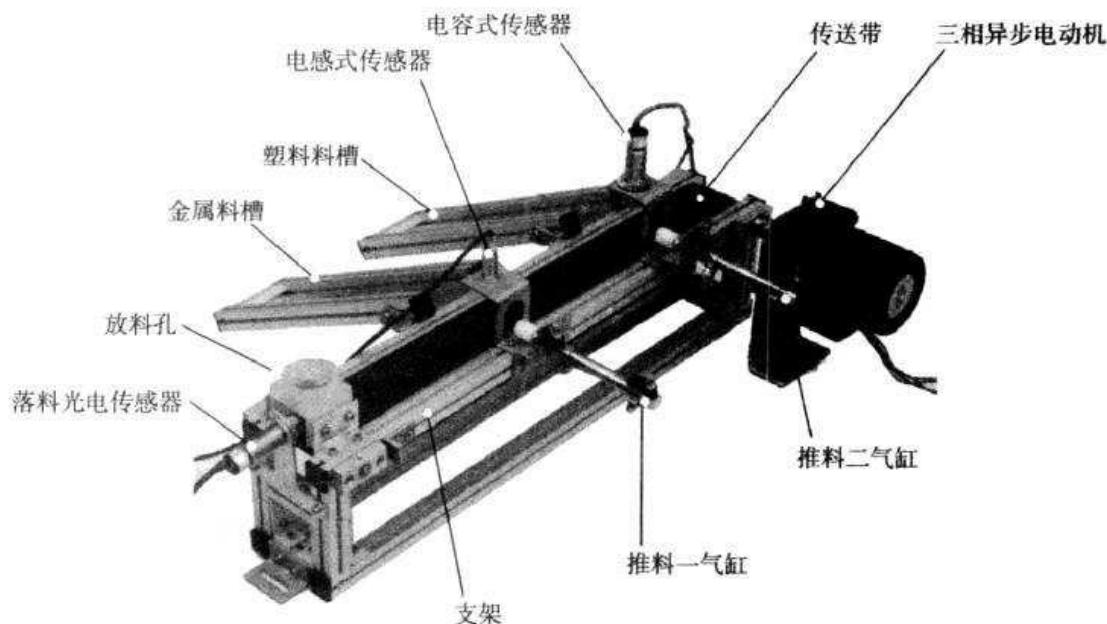


图 1-80 自动分拣装置

自动分拣装置具体包括以下几部分：

- 1) 落料光电传感器：检测是否有物料到传送带上，并给 PLC 一个输入信号；
- 2) 放料孔：物料落料位置定位；
- 3) 金属料槽：放置金属物料；
- 4) 塑料料槽：放置非金属物料；
- 5) 电感式传感器：检测金属材料；
- 6) 电容式传感器：检测非金属材料；
- 7) 三相异步电动机：驱动传送带转动，低速运行；
- 8) 推料气缸：将物料推入料槽，由双向电控气阀控制。

2. 物料自动分拣用的检测开关

(1) 金属分拣用的电感式传感器

电感式传感器由三大部分组成：振荡器、开关电路及放大输出电路。振荡器产生一个交



变磁场。当金属目标接近这一磁场，并达到感应距离时，在金属目标内产生涡流，从而导致振荡衰减，以至停振。振荡器振荡及停振的变化被后级放大电路处理并转换成开关信号，触发驱动控制器件，从而达到非接触式的检测目的（见图 1-81）。

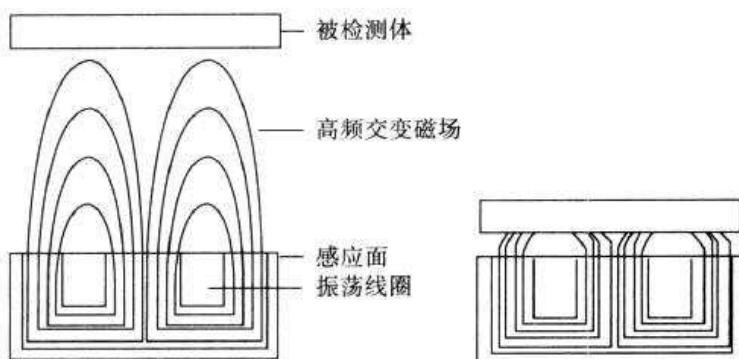


图 1-81 电感式传感器的工作原理

在物料分拣中，电感式传感器的工作流程如图 1-82 所示，金属物体在接近电感式传感器的振荡感应头时，会在物体内部产生涡流，这个涡流反作用于传感器，使传感器振荡能力衰减，内部电路的参数发生变化，由此识别出有无金属物体接近，进而控制开关的通或断。这种电感式传感器所能检测的物体必须是金属物体。

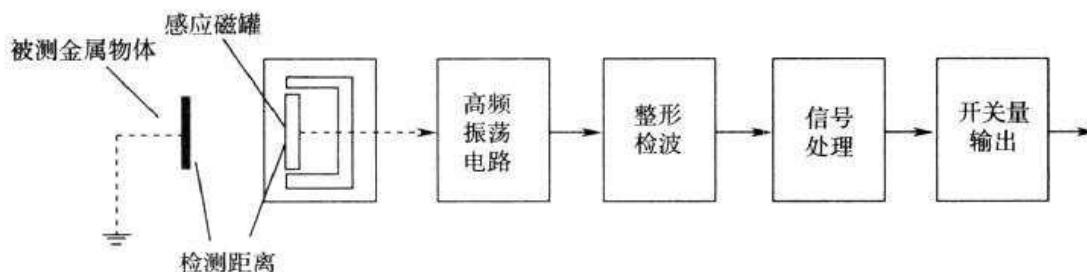


图 1-82 电感式传感器的工作流程

(2) 物料分拣用的电容式传感器

电容式传感器的感应面由两个同轴金属电极构成，很像“打开的”电容器电极，这两个电极构成一个电容，串接在 RC 振荡回路内。电源接通时，RC 振荡器不振荡，当一目标朝着电容器靠近时，电容器的容量增加，振荡器开始振荡。通过后级电路的处理，将振和振荡两种信号转换成开关信号，从而起到了检测有无物体存在的目的。该传感器能检测金属物体，也能检测非金属物体，对金属物体可以获得最大的动作距离，对非金属物体动作距离决定于材料的介电常数，材料的介电常数越大，可获得的动作距离越大。

图 1-83 所示为电容式传感器的工作流程。电容式传感器亦属于一种具有开关量输出的位置传感器，它的测量头通常是构成电容器的一个极板，而另一个极板是物体的本身，当物体移向接近开关时，物体和接近开关的介电常数发生变化，使得和测量头相连的电路状态也随之发生变化，由此便可控制开关的接通和关断。这种传感器的检测物体，并不限于金属导体，也可以是绝缘的液体或粉状物体，在检测较低介电常数 ϵ 的物体时，可以调节传感器的内部电位器来增加感应灵敏度。

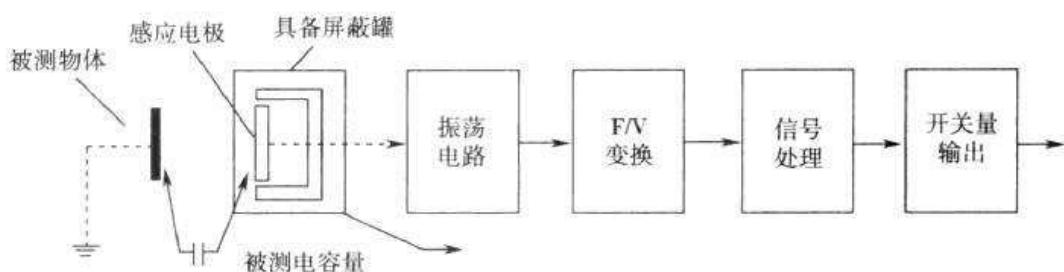


图 1-83 电容式传感器的工作流程

(3) 提升气缸的到位检测

为了确保气缸动作的正确性，通常还要在气缸上安装传感器，用于检测气缸伸出和缩回是否到位。由于提升气缸的动作形式为开关形式，因此只需要在气缸的前点和后点上各安装一个传感器，当检测到气缸准确到位后，就给 PLC 或上位机发出一个信号。

由于气缸运动部件是处于金属壳体内部，这时就无法使用光电开关、电感开关等常用的接近开关来进行检测，在这种情况下可以考虑采用能测量永久变化的磁感应式接近开关。图 1-84 所示是利用磁性开关来测量气缸活塞运动位置的示意图。

永久磁铁固定在非导磁性材料制作的活塞体内，磁性开关固定在不导磁材料制作的气缸壁上。磁性开关壳体内设置一只绕有线圈的 U 型铁心，如图 1-85 所示。当磁铁运动到铁心正上方时，铁心饱和，线圈电感量大大减小，通过转换电路，使输出端（OC 门）跳变为低电平（NPN 型）；当磁铁远离铁心时，OC 门恢复为高阻态。

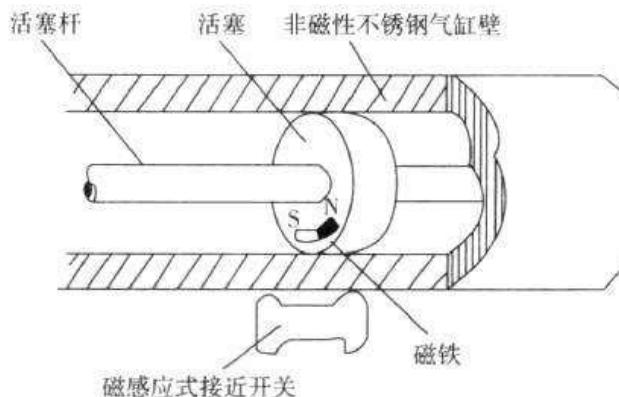


图 1-84 利用磁性开关来测量气缸活塞运动位置

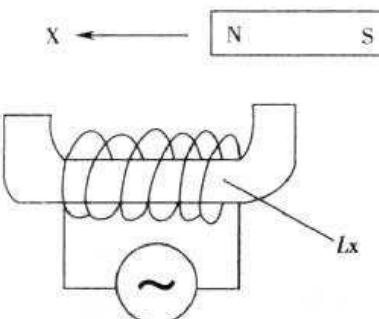


图 1-85 磁性开关的工作原理

图 1-86 所示为日本 SMC 公司生产的 D-C73 型磁性开关外观。

用于气缸到位检测的磁性开关有不同的安装方式，主要有带式、导轨式、拉杆式和直接式四种，如图 1-87 所示。



图 1-86 D-C73 型磁性开关外观

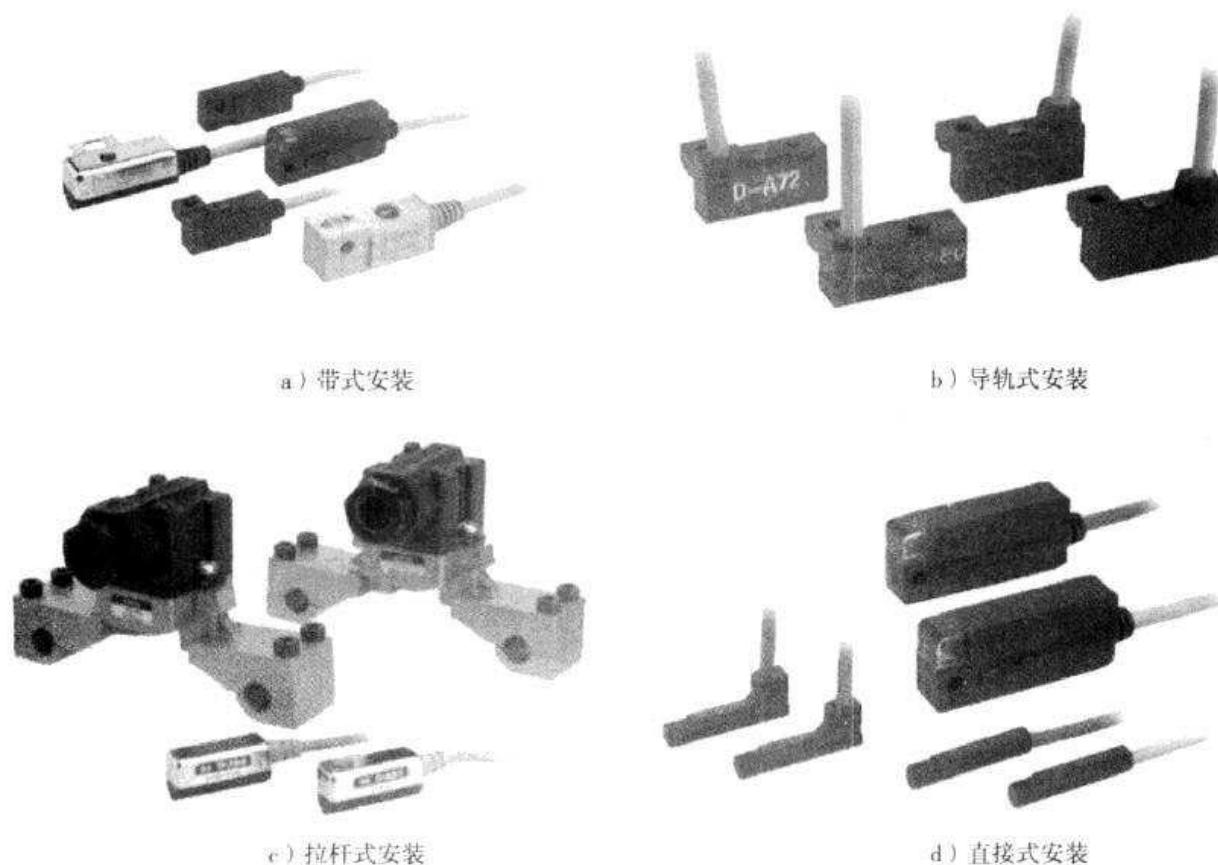


图 1-87 气缸用磁性开关的安装方式

1.5.2 硬件部分接线

1. 列出 I/O 资源配置

根据自动分拣控制工艺要求，列出物料传送和分拣机构的 I/O 资源配置，见表 1-8。

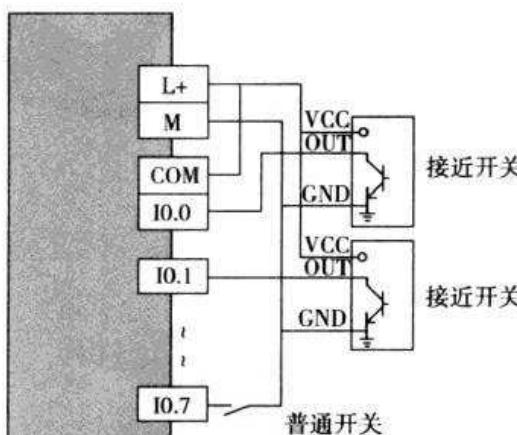
表 1-8 自动分拣装置的 I/O 资源配置

| 序号 | 输入地址 | 说 明 | 序号 | 输入地址 | 说 明 |
|----|------|---------------|----|------|-----------|
| 1 | I0.0 | 推料一气缸后限位 | 1 | Q0.0 | 推料一气缸（缩回） |
| 2 | I0.1 | 推料一气缸前限位 | 2 | Q0.1 | 推料一气缸（推出） |
| 3 | I0.2 | 推料二气缸前限位 | 3 | Q0.2 | 推料二气缸（缩回） |
| 4 | I0.3 | 推料二气缸后限位 | 4 | Q0.3 | 推料二气缸（伸出） |
| 5 | I0.4 | 电感式传感器（推料1气缸） | 5 | Q0.4 | 传动带启停 |
| 6 | I0.5 | 电容式传感器（推料2气缸） | | | |
| 7 | I0.6 | 传送带物料检测光电传感器 | | | |

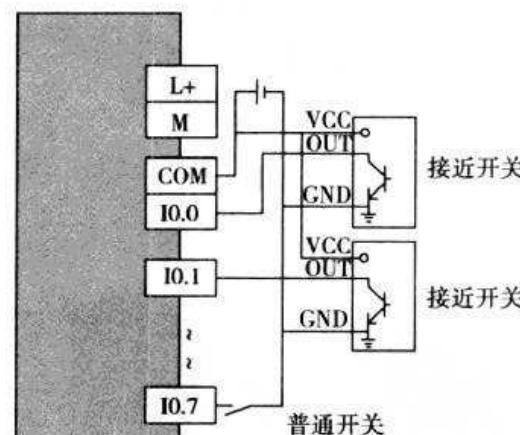


2. PLC 外部接线原理

无论是光电开关还是磁性开关，都有 NPN 型和 PNP 型输出两种接线方式。对于 PLC 来说，也有对应的两种输入接线方式，如图 1-88 和图 1-89 所示。

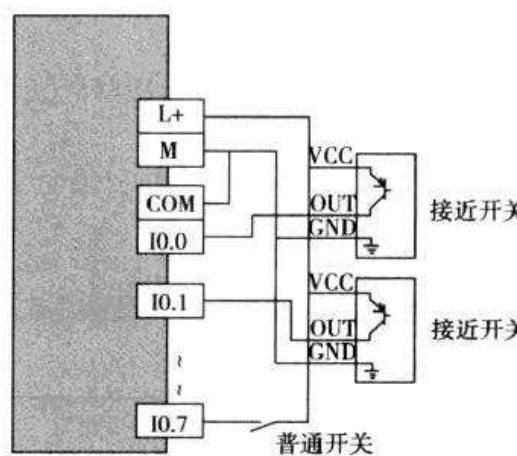


NPN型接近开关（内部供电）

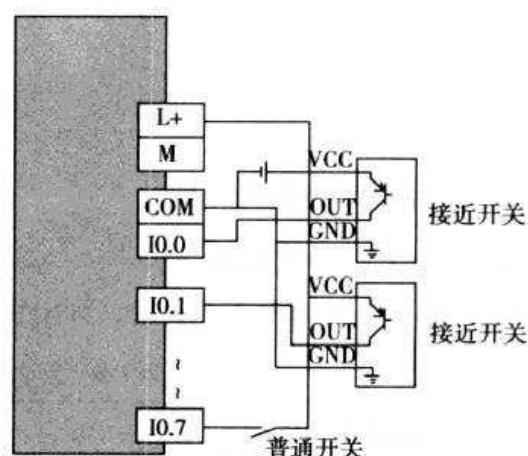


NPN型接近开关（外部供电）

图 1-88 NPN 型开关的 PLC 输入接线方式



PNP型接近开关（内部供电）



PNP型接近开关（外部供电）

图 1-89 PNP 型开关的 PLC 输入接线方式

根据自动分拣要求，画出 PLC 外部接线（见图 1-90），其中接近开关和光电开关等均采用 PNP 型接线，而输出电磁阀或接触器等采用直流 24V 线圈。

3. 软件编程

自动分拣控制的 PLC 变量表见表 1-9，除输入信号和输出信号外，新增了进口物料检测上升沿% M0.0 布尔变量和未检测到物料定时器% MD4 时间变量。

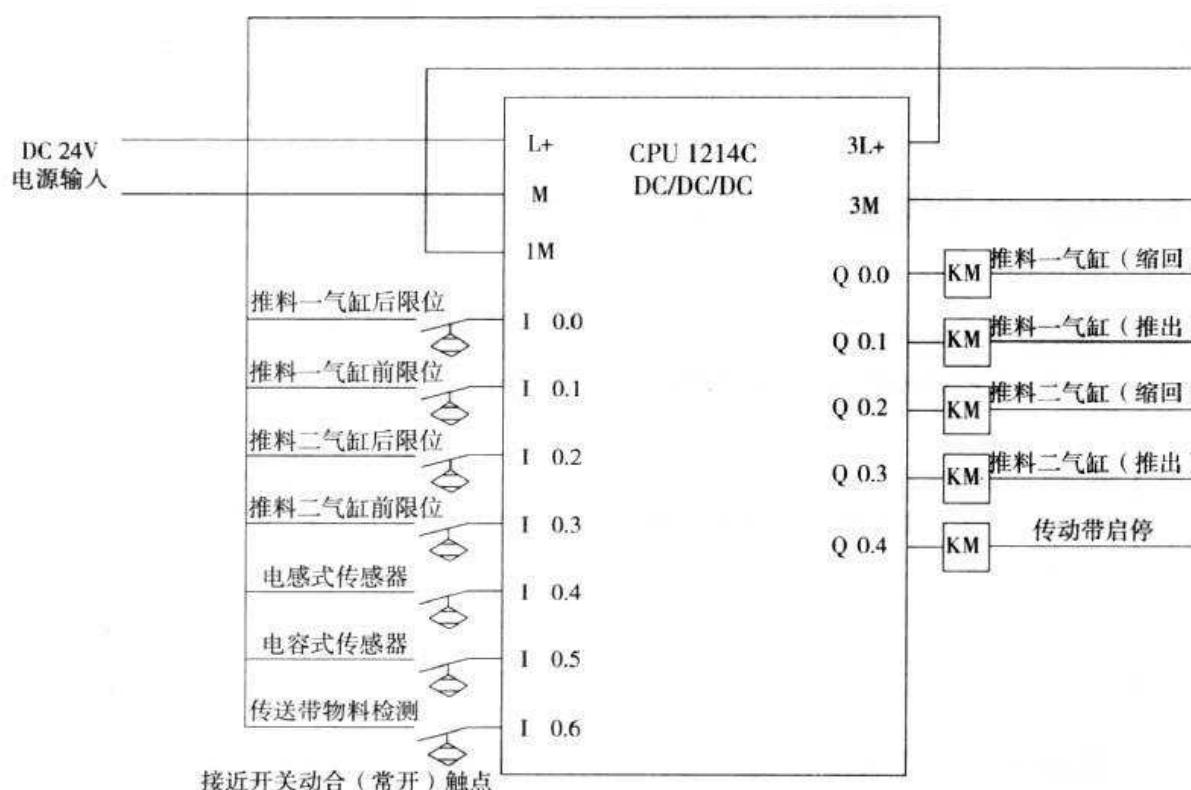


图 1-90 自动分拣装置的 PLC 外部接线

表 1-9 自动分拣控制的 PLC 变量表

| 序号 | 名称 | 数据类型 | 地址 |
|----|-----------|------|-------|
| 1 | 推料一气缸后限位 | Bool | %I0.0 |
| 2 | 推料一气缸前限位 | Bool | %I0.1 |
| 3 | 推料二气缸后限位 | Bool | %I0.2 |
| 4 | 推料二气缸前限位 | Bool | %I0.3 |
| 5 | 电感式传感器 | Bool | %I0.4 |
| 6 | 电容式传感器 | Bool | %I0.5 |
| 7 | 进口物料检测 | Bool | %I0.6 |
| 8 | 推料一气缸缩回 | Bool | %Q0.0 |
| 9 | 推料一气缸推出 | Bool | %Q0.1 |
| 10 | 推料二气缸缩回 | Bool | %Q0.2 |
| 11 | 推料二气缸推出 | Bool | %Q0.3 |
| 12 | 传送带启动 | Bool | %Q0.4 |
| 13 | 进口物料检测上升沿 | Bool | %M0.0 |
| 14 | 未检测到物料定时器 | Time | %MD4 |

本程序中需要注意的编程要点是：如果在设定时间（本案例为 60s）内没有检测到新的物料，则传送带停止；如果在设定时间内有新的物料检测后，则定时器复位。因此，需要应用到复位定时器的指令“----[RT]----”，它可重置指定定时器的定时器数据。

同时，由于需要检测上升沿信号，需要用到 P 触点（见图 1-91）。

它在分配的“IN”位上检测到正跳变（关到开）时，该触点的状

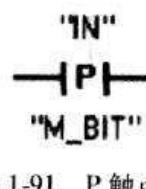


图 1-91 P 触点



图解西门子 S7-1200 PLC 入门到实践

态为 TRUE；该触点逻辑状态随后与能流输入状态组合以设置能流输出状态。P 触点可以放置在程序段中除分支结尾外的任何位置。

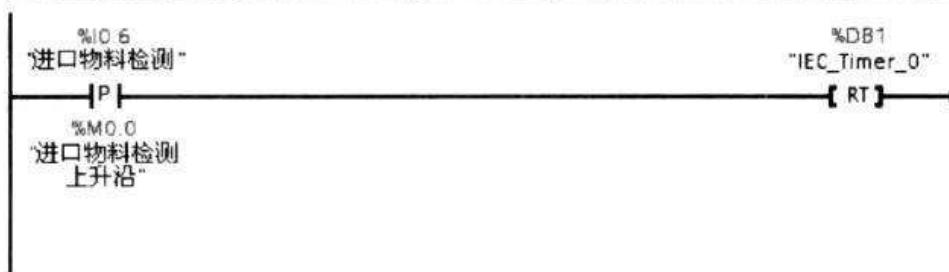
图 1-92 所示是自动分拣装置的详细梯形图程序，本程序中，定时器采用了单一背景数据块 (DB1)。

▼ 块标题： "Main Program Sweep (Cycle)"

主扫描程序

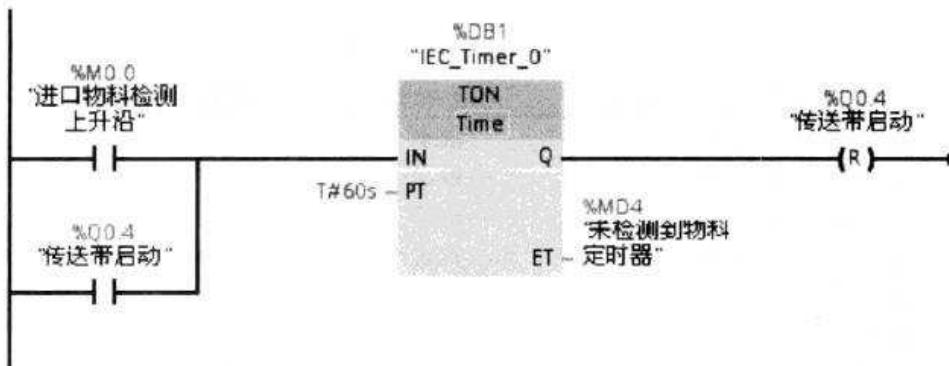
▼ 程序段 1：

进口物料检测到信号时，复位“未检测物料定时器”IEC_Timer_0。



▼ 程序段 2：

满足“进口物料检测上升沿”或“传送带启动”任一条件时，开始计时；当定时到时，传送带电动机停机。



▼ 程序段 3：

当未检测到物料定时器为 (0, 最大值) 区间时，传送带启动；在该区间外，则传送带停止。

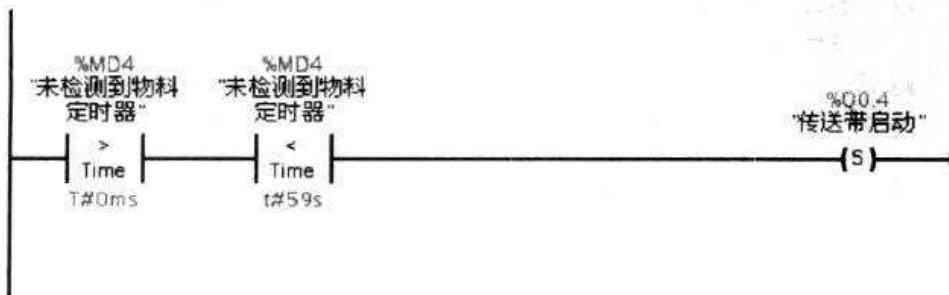
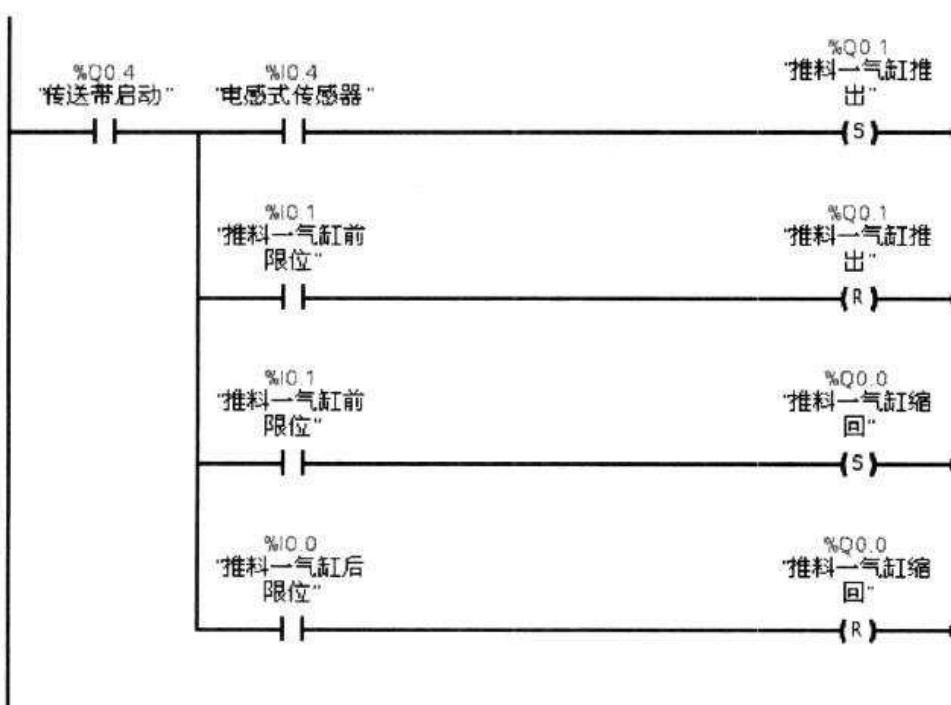


图 1-92 自动分拣装置的详细梯形图程序

▼ 程序段 4：

推料一气缸位置电感传感器检测到有金属物品时,用气缸将金属推出;当前限位动作时,气缸缩回。



▼ 程序段 5：

推料二气缸位置电容传感器检测到有非金属物品时,用气缸将非金属推出;当前限位动作时,气缸缩回。

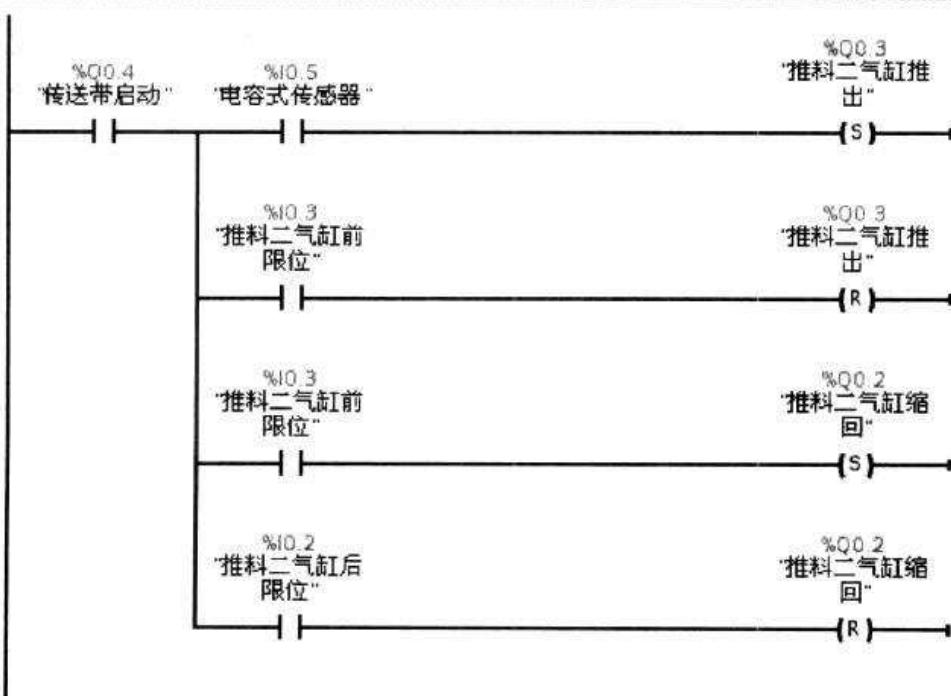


图 1-92 自动分拣装置的详细梯形图程序 (续)



调试定时器的在线数据如图 1-93 所示。

▼ 程序段 2：

满足“进口物料检测上升沿”或“传送带启动”任一条件时，开始计时；当定时到时，传送带电动机停机。

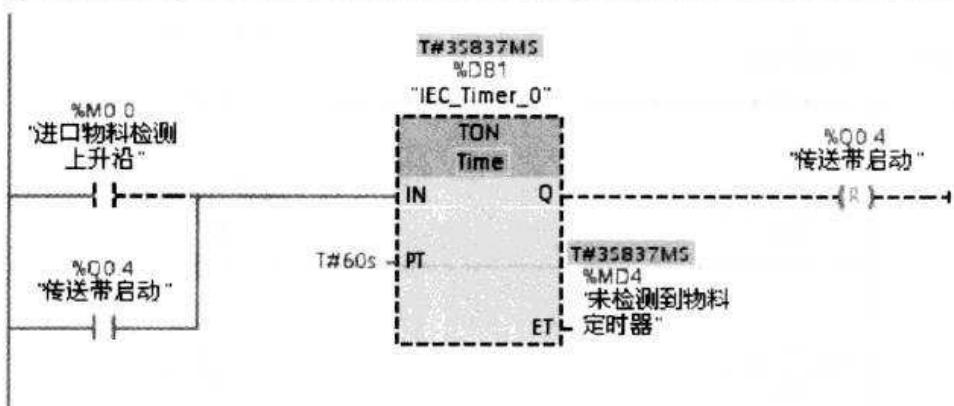
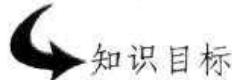


图 1-93 调试定时器的在线数据

第2章 S7-1200 PLC扩展模块的应用

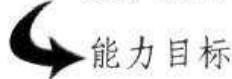
S7-1200 PLC的扩展模块分输出模块和输入模块，在PLC主机的输入或输出点不够用的情况下就要用到扩展模块，主要是扩展PLC的输出或输入点数。当然采用继电器扩展模块还可以在外围设备反馈电压高于PLC的反馈电压时用到。S7-1200 PLC的扩展模块主要包括安装在CPU前端的信号板(SB)、连接在CPU右侧的信号模块(SM)和连接在CPU左侧的通信模块(CM)。

本章的学习目标如下：



知识目标

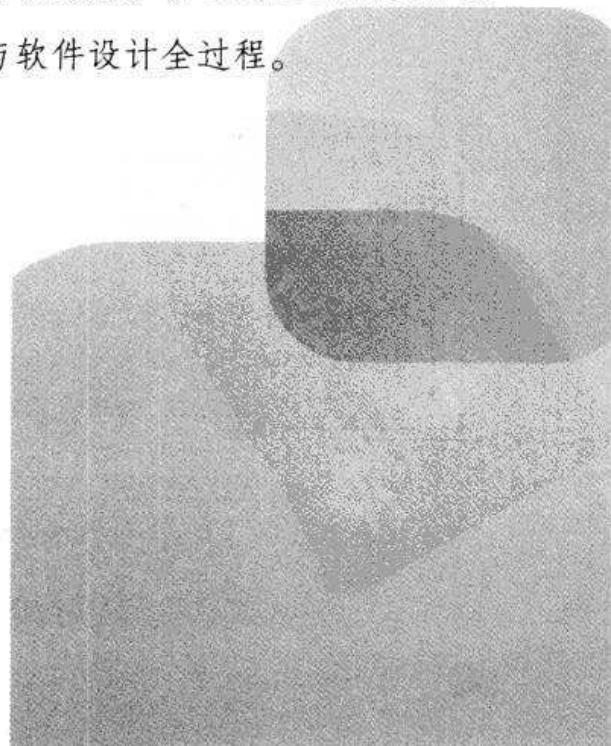
了解S7-1200 PLC扩展模块的类型及其安装方式、熟悉扩展模块的变量寻址、熟悉模拟量模块的指令。



能正确地选择S7-1200 PLC数字量或模拟量扩展模块，并进行正确的硬件配置，实现一个小型自动化控制系统的硬件与软件设计全过程。



能正确地树立全局自动化系统观。





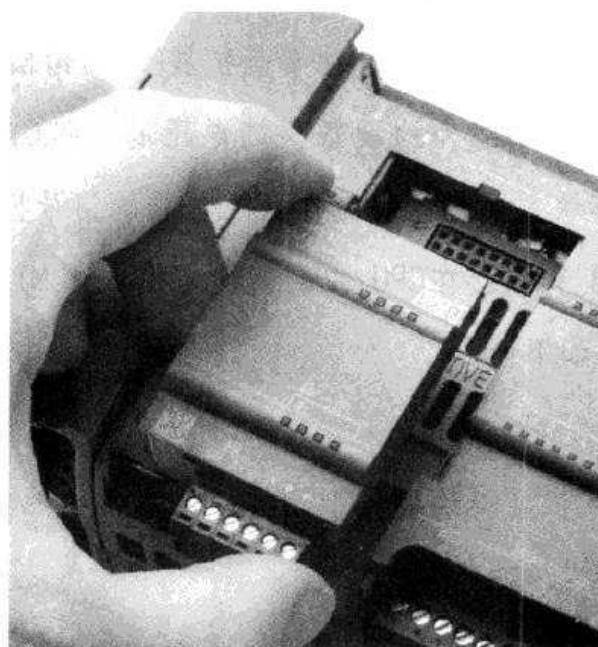
2.1 背景知识介绍：S7-1200 PLC 的扩展模块

2.1.1 扩展模块介绍

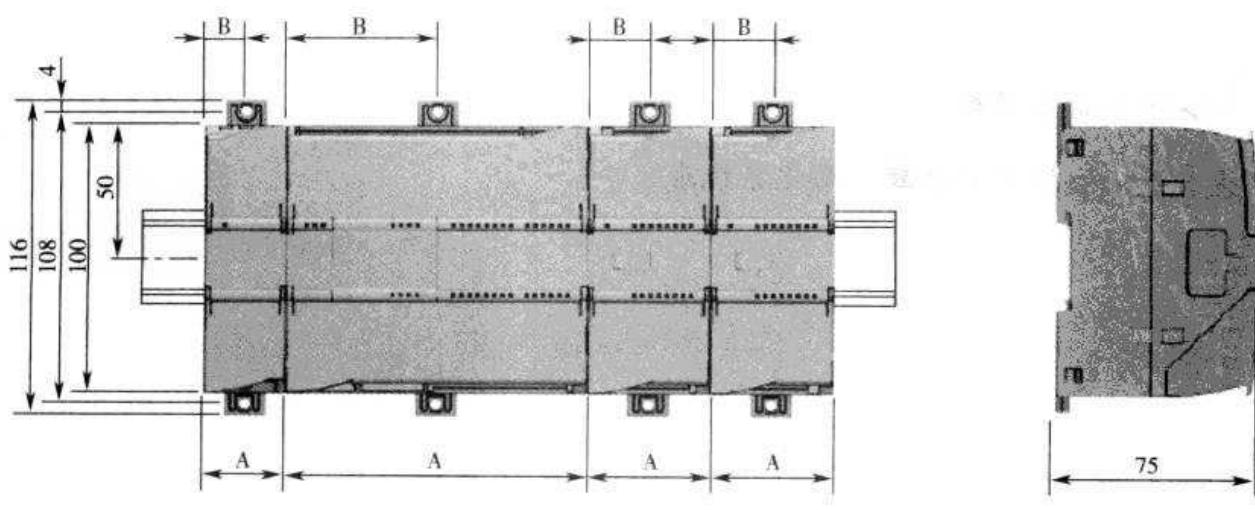
PLC 的扩展模块分输出模块和输入模块，在 PLC 主机的输入或输出点不够用的情况下就要用到扩展模块，主要是扩展 PLC 的输出或输入点数，还有比如在外围设备反馈电压高于 PLC 的反馈电压时就要用到继电器。另外，有的 I/O 信号采用中继隔离是为了避免损坏 PLC 的 I/O 端口。

S7-1200 PLC 有三种类型的模块：

- 1) 信号板 (SB)：仅为 CPU 提供几个附加的 I/O 点。SB 安装在 CPU 的前端（见图 2-1a）。



a) 信号板



b) 信号模块

图 2-1 扩展模块的安装位置



- 2) 信号模块 (SM): 提供附加的数字或模拟 I/O 点。这些模块连接在 CPU 右侧。
- 3) 通信模块 (CM): 为 CPU 提供附加的通信端口 (RS-232 或 RS-485)。这些模块连接在 CPU 左侧。

S7-1200 PLC 的扩展模块设计方便并易于安装，无论安装在面板上还是标准 DIN 导轨上，其紧凑型设计都有利于有效利用空间。CPU、SM 和 CM 支持 DIN 导轨安装和面板安装。使用模块上的 DIN 导轨卡夹将设备固定到导轨上 (见图 2-1b)。这些卡夹还能掰到一个伸出位置以提供将设备直接安装到面板上的螺钉安装位置。一般而言，设备上 DIN 卡夹的孔内部尺寸是 4.3mm。

表 2-1 为常见 S7-1200 PLC 的扩展模块类型与安装尺寸，用户可以参照它来为自动化系统选配合理而正确的控制箱、柜。

表 2-1 常见 S7-1200 PLC 的扩展模块的类型与安装尺寸

| S7-1200 设备 | | 宽度 A/mm | 宽度 B/mm |
|------------|---|---------|---------|
| CPU | CPU 1211C 和 CPU 1212C | 90 | 45 |
| | CPU 1214C | 110 | 55 |
| 信号模块(SM) | 8 和 16 点 DC 和继电器型 (8I、16I、8Q、16Q、8I/8Q) 模拟量 (4AI、8AI、4AI/4AQ、2AQ、4AQ) | 45 | 22.5 |
| | 16I/16Q 继电器型 (16I/16Q) | 70 | 35 |
| 通信模块(CM) | CM 1241 RS-232 和 CM 1241 RS-485 | 30 | 15 |

规划安装时，还需要注意以下指导原则：将设备与热辐射、高压和电噪声隔离开；留出足够的空隙以便冷却和接线；必须在设备的上方和下方留出 25mm 的发热区以便空气自由流通。

2.1.2 扩展模块的变量寻址

在 S7-1200 PLC 中，每个存储单元都有惟一的地址，用户程序利用这些地址访问存储单元中的信息。对输入 (I) 或输出 (Q) 存储区 (例如 I0.3 或 Q1.7) 的引用会访问过程映像。要立即访问物理输入或输出，请在引用后面添加 “: P” (例如，I0.3:P、Q1.7:P 或 "Stop: P")。仅向输入 (I) 或输出 (Q) 强制写入值，要强制输入或输出，请在 PLC 变量或地址后面添加 “:P”。扩展模块的存储区见表 2-2。

表 2-2 扩展模块的存储区

| 存 储 区 | 说 明 | 强 制 | 保 持 性 |
|--------------------------------|--------------------------|-----|-------|
| I 过程映像输入 I_-:P (物理输入) | 在扫描周期开始时从物理输入复制 | 否 | 否 |
| | 立即读取 CPU、SB 和 SM 上的物理输入点 | 是 | 否 |
| Q 过程映像输出 Q_-:P (物理输出) | 在扫描周期开始时复制到物理输出 | 否 | 否 |
| | 立即写入 CPU、SB 和 SM 上的物理输出点 | 是 | 否 |



图解西门子 S7-1200 PLC 入门到实践

图 2-2 说明了如何访问一个位（也称为“字节.位”寻址）。在本例中，存储区和字节地址（Q 代表位存储区，3 代表字节 3）通过后面的句点（“.”）与位地址（位 4）分隔。

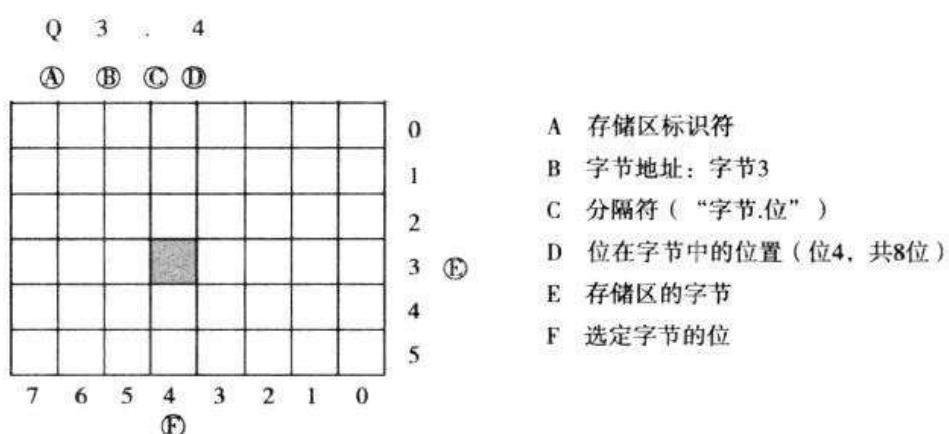


图 2-2 扩展模块的存储区寻址

2.1.3 数字量输入输出电路的工作原理及模块选型

1. 数字量输入电路的工作原理

数字量输入电路按以下方式响应输入信号（见图 2-3）：

- 1) 输入滤波滤除因触点抖动或电气干扰而产生的错误信号；
- 2) 光电隔离将逻辑电路与输入信号隔离，以保护输入电路和背板电路；
- 3) 逻辑电路处理信号；
- 4) 用输入指示灯 LED 的亮或不亮，指示输入设备的状态。

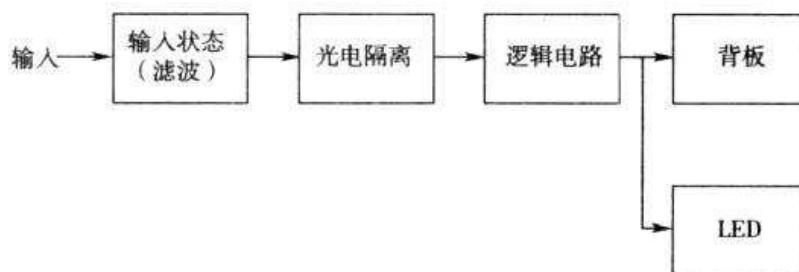


图 2-3 数字量输入电路的工作原理

2. 数字量输出电路的工作原理

数字量输出电路按以下方式控制输出信号（见图 2-4）：

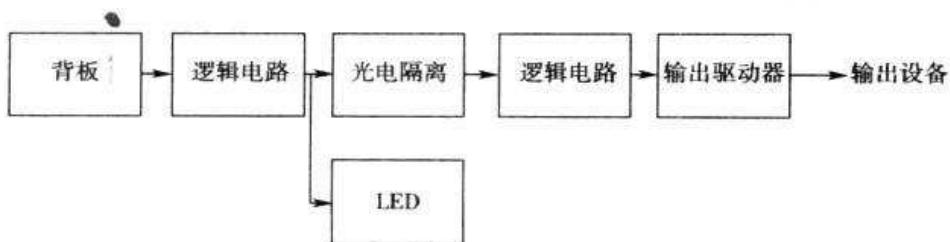


图 2-4 数字量输出电路的工作原理



- 1) 逻辑电路决定输出状态；
- 2) 输出指示灯 LED 指示输出信号状态；
- 3) 光电隔离将现场信号与输出逻辑电路、背板电路隔离；
- 4) 输出驱动器使相应的输出设备接通或断开。

3. 数字量输入模块的选型

对于 S7-1200 PLC 来说，主要有 6ES7221-1BF30-0XB0 (SM1221 8 输入)、6ES7221-1BH30-0XB0 (SM1221 16 输入) 等，具体输入规范见表 2-3。

表 2-3 数字量输入模块的规范

| 类 型 | 漏型/源型 (IEC 1 类漏型) |
|--------------|---------------------|
| 额定电压 | 4mA 时为 DC 24V，额定值 |
| 允许的连续电压 | 最大为 DC 30V |
| 浪涌电压 | DC 35V，持续 0.5s |
| 逻辑 1 信号 (最小) | 2.5mA 时为 DC 15V |
| 逻辑 0 信号 (最大) | 1mA 时为 DC 5V |
| 隔离 (现场侧与逻辑侧) | AC 500V，持续 1min |
| 隔离组 | 2 |
| 同时接通的输入数 | 8 或 16 个 (具体根据订货型号) |
| 电缆长度 (m) | 500 (屏蔽); 300 (非屏蔽) |

以 SM1221 DI8 * DC 24V 为例，数字量输入模块的电气接线如图 2-5 所示。

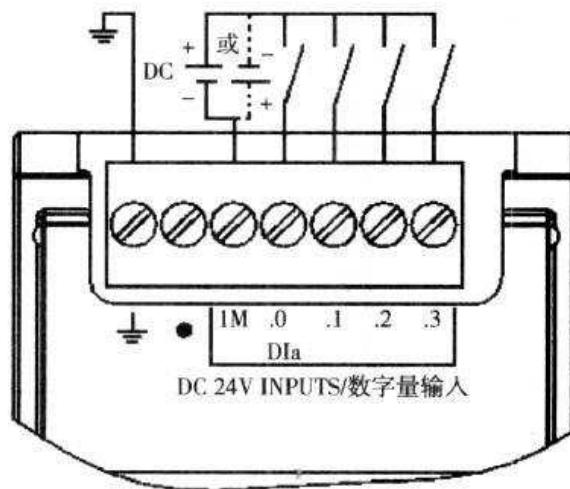


图 2-5 数字量输入模块的电气接线

4. 数字量输出模块的选型

对于 S7-1200 PLC 来说，主要有 6ES7221-1BF30-0XB0 (SM1221 8 输入)、6ES7221-1BH30-0XB0 (SM1221 16 输入) 等，具体输出规范见表 2-4。



表 2-4 数字量输出模块的规范

| 类 型 | 继电器，干触点 | 固态-MOSFET |
|---------------------|---|---------------------------------|
| 电压范围 | DC 5 ~ 30V 或 AC 5 ~ 250V | DC 20.4 ~ 28.8V |
| 最大电流时的逻辑 1 信号 | — | 最小为 DC 20V |
| 具有 10kΩ 负载时的逻辑 0 信号 | — | 最大为 DC 0.1V |
| 电流 (最大) | 2.0A | 0.5A |
| 灯负载 | DC 30W/AC 200W | 5W |
| 通态触点电阻 | 新设备最大为 0.2Ω | 最大为 0.6Ω |
| 每点的漏泄电流 | — | 最大为 10μA |
| 浪涌电流 | 触点闭合时为 7A | 8A, 最长持续 100ms |
| 隔离 (现场侧与逻辑侧) | AC 1500V, 持续 1min (线圈与触点) 无 (线圈与逻辑侧) | AC 500V, 持续 1min |
| 开关延迟 | 最长为 10ms | 断开到接通最长为 50μs 接通到断开最长为 200μs |
| 机械寿命 (无负载) | 10,000,000 个断开/闭合周期 | — |
| 额定负载下的触点寿命 | 100,000 个断开/闭合周期 | — |

以 SM1222 DI8 * DC 24V 为例, 数字量输出模块的电气接线如图 2-6 所示。

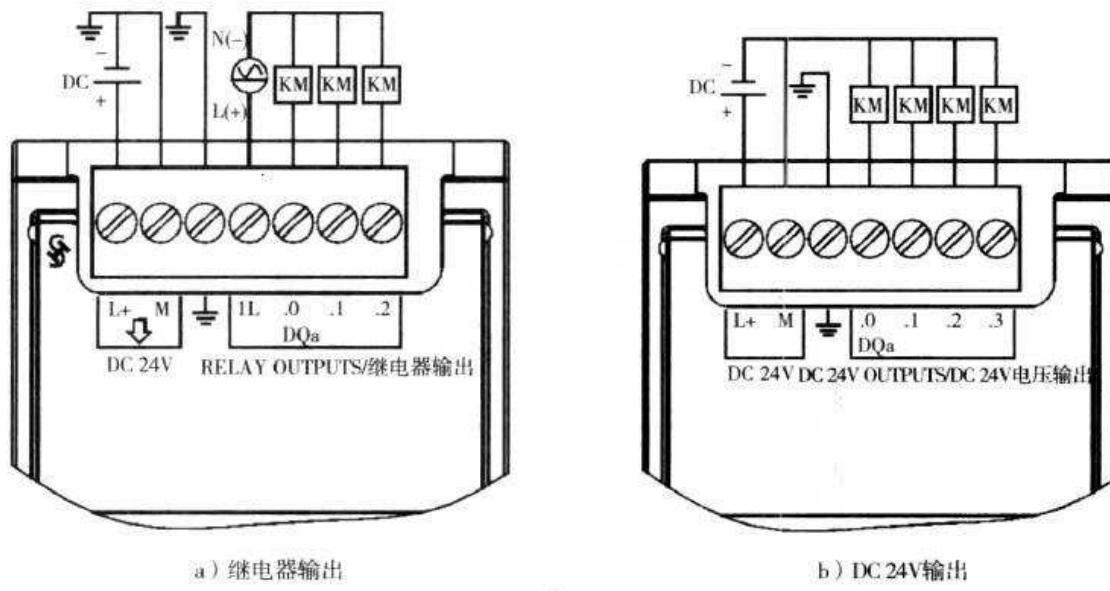


图 2-6 SM1222 输出模块的电气接线

5. 数字量输入输出混合模块选型

对于 S7-1200 PLC 来说, 主要有 6ES7221-1BF30-0XB0 (SM1221 8 输入)、6ES7221-1BH30-0XB0 (SM1221 16 输入) 等, 具体规范见表 2-5。



表 2-5 数字量输入输出混合模块的规范

| 型 号 | SM 1223 DI 8x24 VDC, DQ 8x 继电器 | SM 1223 DI 16x24 VDC, DQ 16x 继电器 | SM 1223 DI 8x24 VDC, DQ 8x24 VDC | SM 1223 DI 16x24 VDC, DQ16x24 VDC |
|------------------|-----------------------------------|-------------------------------------|-------------------------------------|--------------------------------------|
| 订货号 (MLFB) | 6ES7 223-1PH30-0XB0 | 6ES7 223-1PL30-0XB0 | 6ES7 223-1BH30-0XB0 | 6ES7 223-1BL30-0XB0 |
| 电流消耗 (SM 总线) /mA | 145 | 180 | 145 | 185 |
| 输入点数 | 8 | 16 | 8 | 16 |
| 输入类型 | 漏型/源型 (IEC 1 类漏型) | | | |
| 输出点数 | 8 | 16 | 8 | 16 |
| 输出类型 | 继电器, 干触点 | | 固态-MOSFET | |

2.1.4 模拟量输入输出电路的工作原理及模块选型

1. PLC 处理模拟量的过程

在生产过程中，存在大量的物理量，如压力、温度、速度、旋转速度、pH 值、粘度等。为了实现自动控制，这些模拟信号都需要被 PLC 来处理。图 2-7 所示为 PLC 处理模拟量的过程。

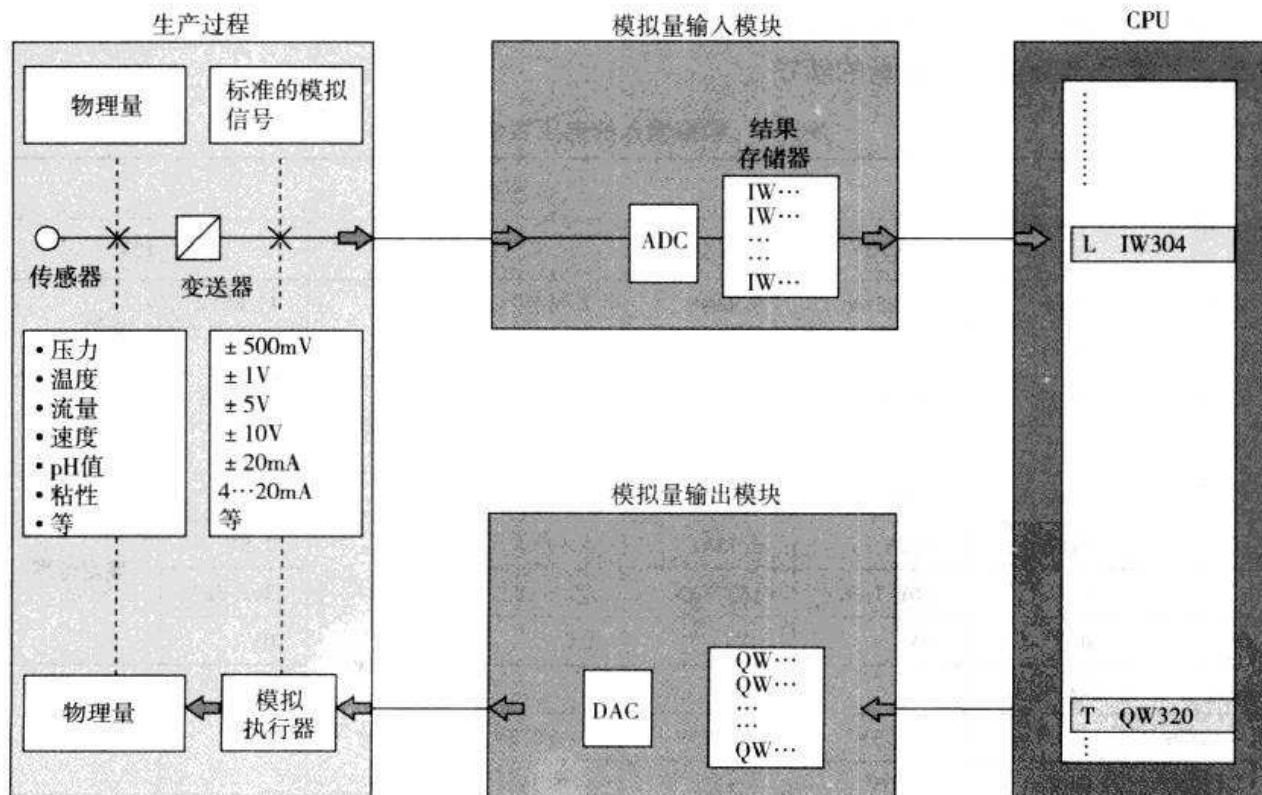


图 2-7 PLC 处理模拟量的过程

图 2-7 中，测量传感器利用线性膨胀、角度扭转或电导率变化等原理来测量物理量的变化；测量变送器将传感器检测到的变化量转换为标准的模拟信号，如：±500mV，±10V，



$\pm 20\text{mA}$, $4\ldots 20\text{mA}$, 这些标准的模拟信号将接到模拟输入模块上。

由于 PLC 的 CPU 只能处理数字量信号, 因此模拟量输入模块中的 ADC (模/数转换器) 就是用来实现转换功能的。模数转换是顺序执行的, 也就是说每个模拟通道上的输入信号是轮流被转换的。模数转换的结果存在结果存储器中, 并一直保持到被一个新的转换值所覆盖。在 S7-1200 PLC 中, 可用“MOVE”指令来访问模数转换的结果。

如果要进行模拟量输出, 也可以使用“MOVE”向模拟量输出模块中写模拟量的数值 (由用户程序计算所得), 该数值由模块中的 DAC (数/模转换器) 变换为标准的模拟信号。采用标准模拟输入信号的模拟执行器可以直接连接到模拟量输出模块上。

2. 模拟量输入模块的选型

图 2-8 所示为模拟量输入模块的电气接线, 对于电压输入或电流输入来说, 接线都是相同的, 对于信号的区别只需要在硬件组态中加以选择即可。

模拟输入的电压表示法见表 2-6, 模拟输入的电流表示法见表 2-7, 只有了解两种表示方法, 才能正确地进行工程量的转换。

表 2-6 模拟输入的电压表示法

| 十进制 | 十六进制 | 电压测量范围 | | | | | |
|--------|------|--------------------|--------------------|-------------------|--------------------|--------------------|------|
| | | $\pm 10\text{V}$ | $\pm 5\text{V}$ | $\pm 2.5\text{V}$ | $0\sim 10\text{V}$ | | |
| 32767 | 7FFF | 11.851V | 5.926V | 2.963V | 上溢 | 11.851V | 上溢 |
| 32512 | 7F00 | | | | | | |
| 32511 | 7EFF | 11.759V | 5.879V | 2.940V | 过冲范围 | 11.759V | 过冲范围 |
| 27649 | 6C01 | | | | | | |
| 27648 | 6C00 | 10V | 5V | 2.5V | 额定范围 | 10V | 额定范围 |
| 20736 | 5100 | 7.5V | 3.75V | 1.875V | | 7.5V | |
| 1 | 1 | $361.7\mu\text{V}$ | $180.8\mu\text{V}$ | $90.4\mu\text{V}$ | | $361.7\mu\text{V}$ | |
| 0 | 0 | 0V | 0V | 0V | | 0V | |
| -1 | FFFF | | | | 不支持负值 | | |
| -20736 | AF00 | -7.5V | -3.75V | -1.875V | | | |
| -27648 | 9400 | -10V | -5V | -2.5V | | | |
| -27649 | 93FF | | | | | | |
| -32512 | 8100 | -11.759V | -5.879V | -2.940V | 下溢 | | |
| -32513 | 80FF | | | | | | |
| -32768 | 8000 | -11.851V | -5.926V | -2.963V | | | |

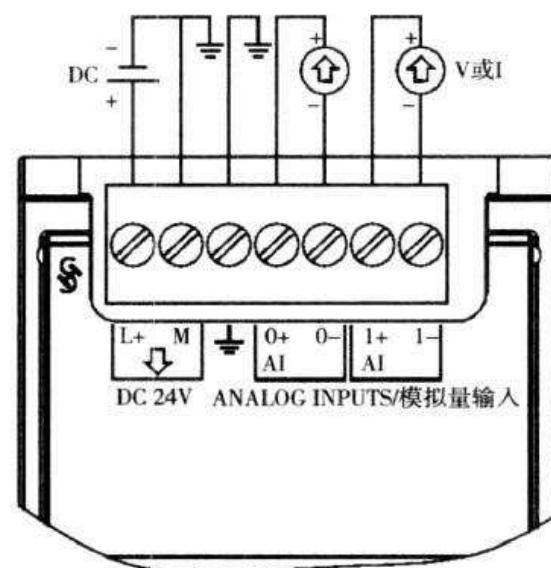


图 2-8 模拟量输入模块的电气接线



表 2-7 模拟输入的电流表示法

| 十进制 | 十六进制 | 电流测量范围 | |
|--------|------|----------|------|
| | | 0 ~ 20mA | |
| 32767 | 7FFF | 23.70mA | 上溢 |
| 32512 | 7F00 | | |
| 32511 | 7EFF | 23.52mA | |
| 27649 | 6C01 | | |
| 27648 | 6C00 | 20mA | 额定范围 |
| 20736 | 5100 | 15mA | |
| 1 | 1 | 723.4nA | |
| 0 | 0 | 0mA | |
| -1 | FFFF | | 下冲范围 |
| -4864 | ED00 | -3.52mA | |
| -4865 | ECFF | | |
| -32768 | 8000 | | 下溢 |

3. 模拟量输出模块的选型

图 2-9 所示为模拟量输出模块的电气接线，对于电压输入或电流输出来说，只需要硬件组态即可，接线不需要更改。

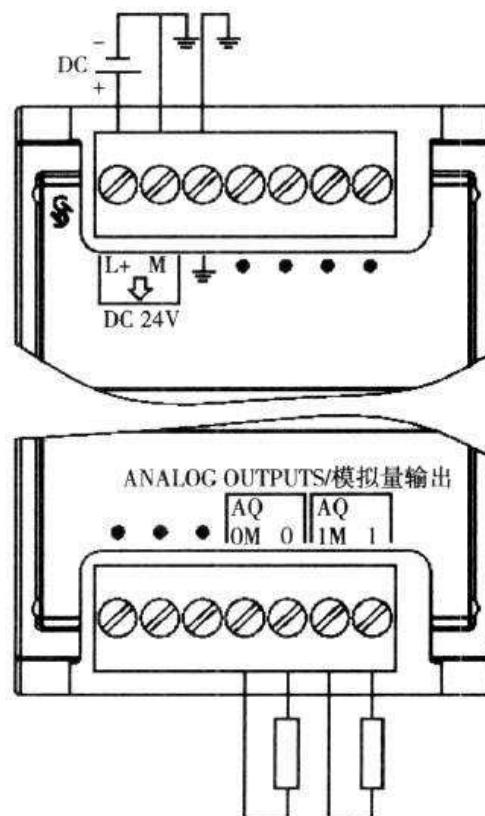


图 2-9 模拟量输出模块的电气接线



图解西门子 S7-1200 PLC 入门到实践

模拟输出的电压表示法见表 2-8，模拟输出的电流表示法见表 2-9。在下溢或上溢情况下，模拟量输出将根据为模拟量信号模块设置的设备配置属性动作。在“对 CPU STOP 的响应”(Reaction to CPU STOP) 参数中，任选“使用替换值”(Usesubstitute value) 或“保持上一个值”(Keep last value)。

表 2-8 模拟输出的电压表示法

| 十进制 | 十六进制 | 电压输出范围 | |
|--------|------|----------|------|
| | | ±10V | 注释 |
| 32767 | 7FFF | | 上溢 |
| 32512 | 7F00 | | |
| 32511 | 7EFF | 11.76V | 过冲范围 |
| 27649 | 6C01 | | |
| 27648 | 6C00 | 10V | |
| 20736 | 5100 | 7.5V | |
| 1 | 1 | 361.7μV | 额定范围 |
| 0 | 0 | 0V | |
| -1 | FFFF | -361.7μV | |
| -20736 | AF00 | -7.5V | |
| -27648 | 9400 | -10V | |
| -27649 | 93FF | | |
| -32512 | 8100 | -11.76V | 下冲范围 |
| -32513 | 80FF | | |
| -32768 | 8000 | | 下溢 |

表 2-9 模拟输出的电流表示法

| 十进制 | 十六进制 | 电流输出范围 | |
|--------|------|---------|------|
| | | ±20mA | 注释 |
| 32767 | 7FFF | | 上溢 |
| 32512 | 7F00 | | |
| 32511 | 7EFF | 23.52mA | 过冲范围 |
| 27649 | 6C01 | | |
| 27648 | 6C00 | 20mA | |
| 20736 | 5100 | 15mA | |
| 1 | 1 | 723.4nA | 额定范围 |
| 0 | 0 | 0mA | |
| -1 | FFFF | | |
| -32512 | 8100 | | |
| -32513 | 80FF | | |
| -32768 | 8000 | | 下溢 |



4. 模拟量输入输出混合模块的选型

S7-1200 PLC 具有模拟量输入输出混合模块 SM1234 AI4 * 13 位/AQ2 * 14 位，其具体接线如图 2-10 所示。

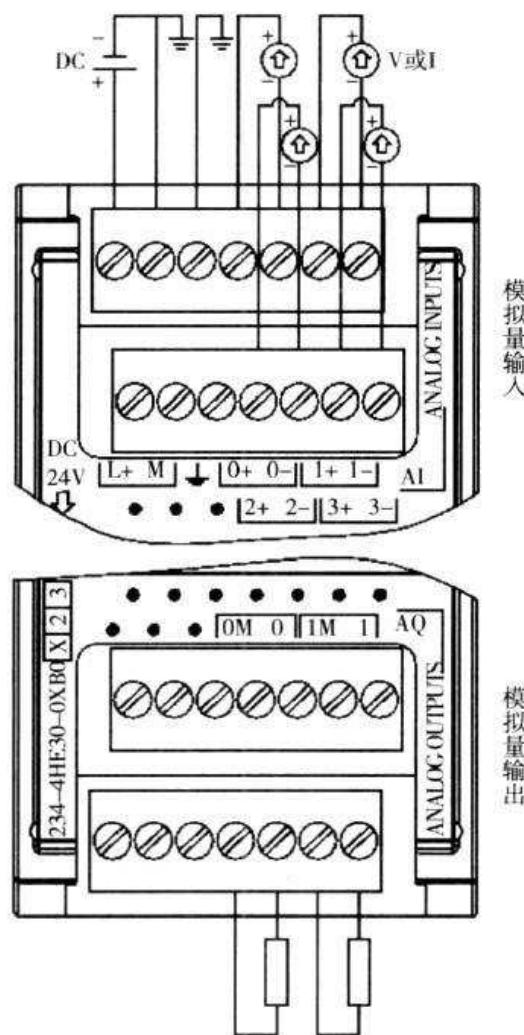


图 2-10 模拟量输入输出混合模块的电气接线

2.1.5 信号板选型

S7-1200 PLC 独有的信号扩展板 SB1223 DC 2 * 24V 输入/DC 2 * 24V 输出技术特性见表 2-10，电气接线如图 2-11 所示。

表 2-10 信号扩展板 SB1223 DC 2 * 24V 输入/DC 2 * 24V 输出技术特性

| | |
|------------------------|--|
| 输入的点数和类型 | IEC 1 类漏型：2 点输入 |
| 额定电压 | 4mA 时为 DC 24V，额定值 |
| 允许的连续电压 | 最大为 DC 30V |
| 浪涌电压 | DC 35V，持续 0.5s |
| 逻辑 1 信号（最小）逻辑 0 信号（最大） | 2.5mA 时为 DC 15V 1mA 时为 DC 5V |
| HSC 时钟输入频率（最大） | 20kHz (DC 15 ~ 30V); 30kHz (DC 15 ~ 26V) |



(续)

| | |
|---------------------|--|
| 隔离（现场侧与逻辑侧） | AC 500V，持续 1min |
| 滤波时间 | 0.2ms、0.4ms、0.8ms、1.6ms、3.2ms、6.4ms 和 12.8ms 可选择，两个为一组 |
| 电缆长度（m） | 500（屏蔽）；300（非屏蔽） |
| 输出的点数和类型 | 固态-MOSFET：2 点输出 |
| 电压范围 | DC 20.4 ~ 28.8V |
| 最大电流时的逻辑 1 信号 | 最小为 DC 20V |
| 具有 10kΩ 负载时的逻辑 0 信号 | 最大为 DC 0.1V |
| 电流（最大） | 0.5A |
| 灯负载 | 5W |
| 通态触点电阻 | 最大为 0.6Ω |
| 每点的漏泄电流 | 最大为 10μA |
| 脉冲串输出频率 | 最大为 20kHz，最小为 2Hz |
| 浪涌电流 | 5A，最长持续 100ms |
| 过载保护 | 无 |

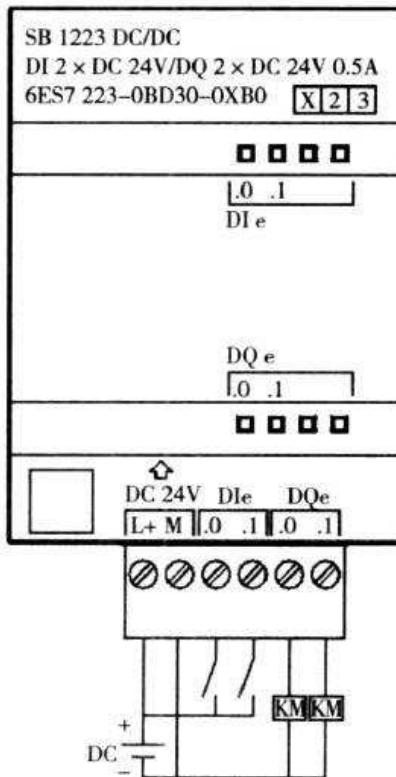


图 2-11 SB1223 信号板的电气接线

SB1232 模拟量信号板的技术指标见表 2-11，其电气接线如图 2-12 所示。

第2章 S7-1200 PLC 扩展模块的应用



表 2-11 SB1232 模拟量信号板的技术指标

| | | |
|--|---------------------------------------|----------------------------------|
| 输出的路数和类型 | 1 路（电压或电流） | |
| 范围 | $\pm 10V$ 或 $0 \sim 20mA$ | |
| 精度 | 电压：12 位 | 电流：11 位 |
| 满量程范围（数据字） | 电压： $-27,648 \sim 27,648$ | 电流： $0 \sim 27,648$ |
| 精度（ $25^{\circ}C / 0 \sim 55^{\circ}C$ ） | 满量程的 $\pm 0.5\% / \pm 1\%$ | |
| 稳定时间（新值的 95%） | 电压： $300\mu s (R), 750\mu s (1\mu F)$ | 电流： $600\mu s (1mH), 2ms (10mH)$ |
| 负载阻抗 | 电压： $\geq 1000\Omega$ | 电流： $\leq 600\Omega$ |
| RUN 到 STOP 时的行为 | 上一个值或替换值（默认值为 0） | |
| 隔离（现场侧与逻辑侧） | 无 | |
| 电缆长度（m） | 100m，屏蔽双绞线 | |
| 诊断 | 上溢/下溢 | 有 |
| | 对地短路(仅限电压模式) | 有 |
| | 断路(仅限电流模式) | 有 |

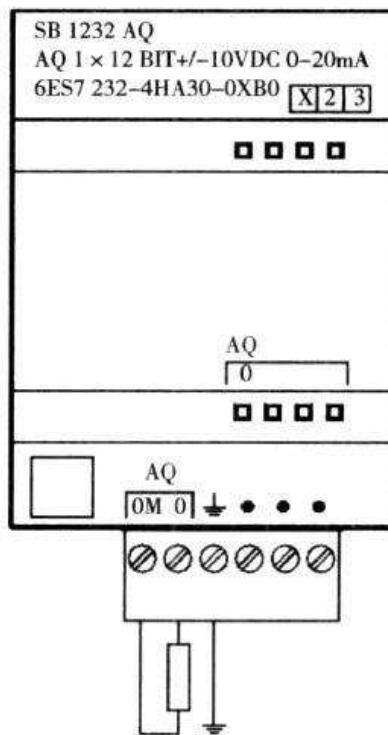


图 2-12 SB1232 模拟量信号板的电气接线

2.2 知识讲座：用户程序结构与数据类型

2.2.1 用户程序的执行

1. 代码块种类

在 S7-1200 PLC 中，CPU 支持 OB、FC、FB、DB 代码块，使用它们可以创建有效的用



户程序结构，具体介绍如下：

1) 组织块 (OB) 定义程序的结构。有些 OB 具有预定义的行为和启动事件，但用户也可以创建具有自定义启动事件的 OB。

2) 功能 (FC) 和功能块 (FB) 包含与特定任务或参数组合相对应的程序代码。每个 FC 或 FB 都提供一组输入和输出参数，用于与调用块共享数据。FB 还使用相关联的数据块 (称为背景数据块) 来保存执行期间的值状态，程序中的其他块可以使用这些值状态。

3) 数据块 (DB) 存储程序块可以使用的数据。

用户程序的执行顺序是：从一个或多个在进入 RUN 模式时运行一次的可选启动 OB 开始，然后执行一个或多个循环执行的程序循环 OB。OB 也可以与中断事件（可以是标准事件或错误事件）相关联，并在相应的标准或错误事件发生时执行。

2. 用户程序的结构

创建用于自动化任务的用户程序时，需要将程序的指令插入代码块中：

1) OB 对应于 CPU 中的特定事件，并可中断用户程序的执行。用于循环执行用户的默认组织块 (OB1) 为用户提供基本结构，是惟一一个用户必需的代码块。如果程序中包括其他 OB，这些 OB 会中断 OB1 的执行。其他 OB 可执行特定功能，如用于启动任务、用于处理中断和错误或者用于按特定的时间间隔执行特定的程序代码。

2) FB 是从另一个代码块 (OB、FB 或 FC) 进行调用时执行的子例程。调用块将参数传递到 FB，并标识可存储特定调用数据或该 FB 实例的特定 DB。更改背景 DB 可使通用 FB 控制一组设备的运行。例如，借助包含每个泵或阀门的特定运行参数的不同背景 DB，一个 FB 可控制多个泵或阀。

3) FC 是从另一个代码块 (OB、FB 或 FC) 进行调用时执行的子例程。FC 不具有相关的背景 DB。调用块将参数传递给 FC。FC 中的输出值必须写入存储器地址或全局 DB 中。

根据实际应用要求，可选择线性结构或模块化结构用于创建用户程序（见图 2-13）。

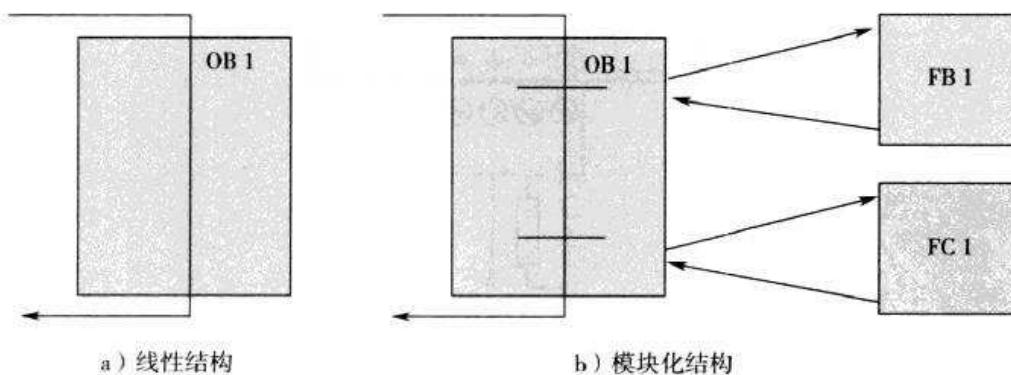


图 2-13 用户程序的结构

线性程序按顺序逐条执行用于自动化任务的所有指令。通常，线性程序将所有程序指令都放入用于循环执行程序的 OB (如 OB1) 中。

模块化程序调用可执行特定任务的特定代码块。要创建模块化结构，需要将复杂的自动化任务划分为与过程的工艺功能相对应的更小的次级任务，每个代码块都为每个次级任务提



供程序段，通过从另一个块中调用其中一个代码块来构建程序。

通过创建可在用户程序中重复使用的通用代码块，可简化用户程序的设计和实现。使用通用代码块具有许多优点：

- 1) 可为标准任务创建能够重复使用的代码块，如用于控制泵或电动机。也可以将这些通用代码块存储在可由不同的应用或解决方案使用的库中。

- 2) 将用户程序构建到与功能任务相关的模块化组件中，可使程序的设计更易于理解和管理。模块化组件不仅有助于标准化程序设计，也有助于使更新或修改程序代码更加快速和容易。

- 3) 创建模块化组件可简化程序的调试。通过将整个程序构建为一组模块化程序段，可在开发每个代码块时测试其功能。

- 4) 创建与特定工艺功能相关的模块化组件，有助于简化对已完成应用程序的调试，并减少调试过程中所用的时间。

3. 使用块来构建程序

通过设计 FB 和 FC 执行通用任务，可创建模块化代码块，然后可通过由其他代码块调用这些可重复使用的模块来构建程序，调用块将设备特定的参数传递给被调用块，块调用示意如图 2-14 所示。当一个代码块调用另一个代码块时，CPU 会执行被调用块中的程序代码。执行完被调用块后，CPU 会继续执行该块调用之后的指令。

调用可嵌套块以实现更加模块化的结构，如图 2-15 所示。

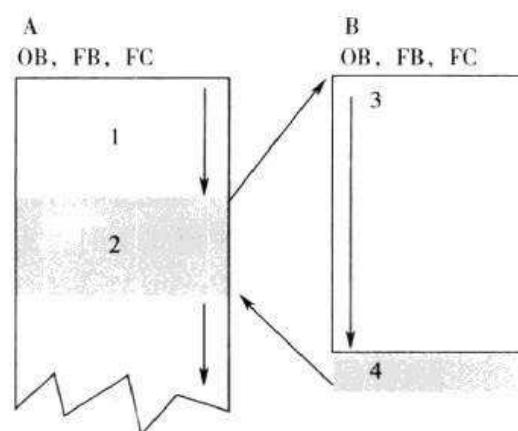


图 2-14 块调用示意

A—调用块 B—被调用（或中断）块
1—程序执行 2—可调用其他块的操作
3—程序执行 4—块结束（返回到调用块）

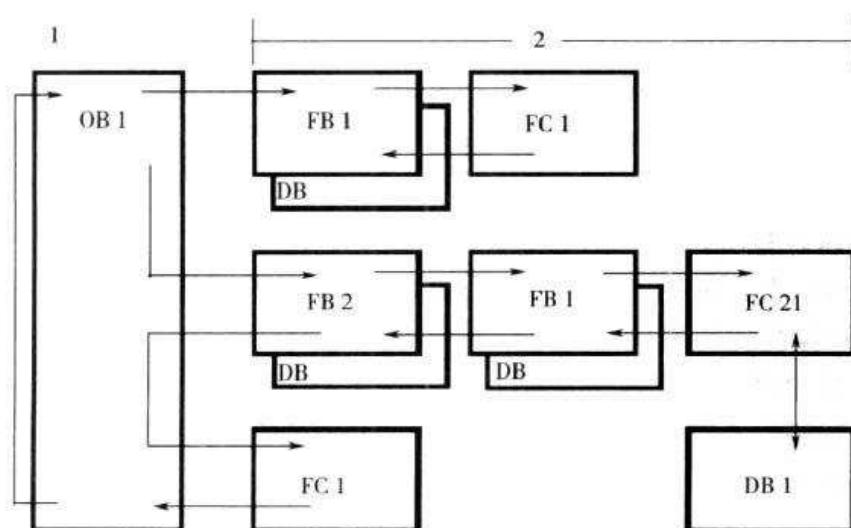


图 2-15 可嵌套块

1—循环开始 2—嵌套深度



2.2.2 S7-1200 PLC 实现控制的过程

1. CPU 的三种工作模式

S7-1200 CPU 有以下三种工作模式：STOP 模式、STARTUP 模式和 RUN 模式。CPU 前面的状态 LED 指示当前的工作模式。

- 1) 在 STOP 模式下，CPU 不执行任何程序，而用户可以下载项目。
- 2) 在 STARTUP 模式下，执行一次启动 OB（如果存在）。在 RUN 模式的启动阶段，不处理任何中断事件。

STARTUP 过程具体描述如下：只要工作状态从 STOP 切换到 RUN，CPU 就会清除过程映像输入，初始化过程映像输出并处理启动 OB。启动 OB 中的指令对过程映像输入进行任何读访问时，读取到的都只有零，而不是当前物理输入值。因此，要在启动模式下读取物理输入的当前状态，必须执行立即读取操作。接着再执行启动 OB 以及任何相关的 FC 和 FB。如果存在多个启动 OB，则按照 OB 编号依次执行各启动 OB，OB 编号最小的先执行。

3) 在 RUN 模式下，重复执行扫描周期。中断事件可能会在程序循环阶段的任何点发生并进行处理。处于 RUN 模式下时，无法下载任何项目。在 RUN 模式下，CPU 执行图 2-16 所示的任务。

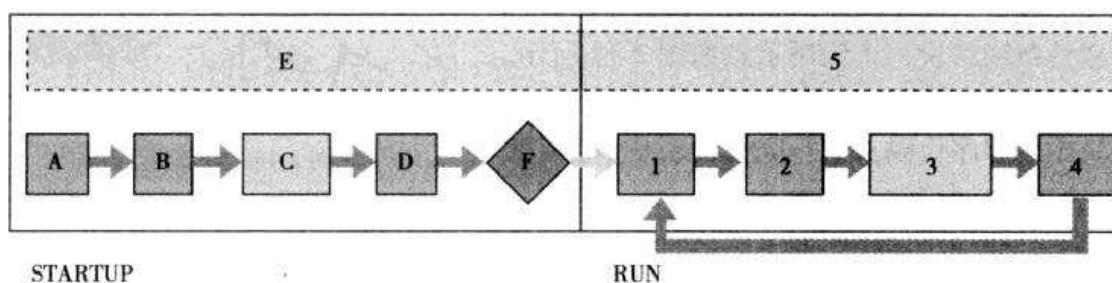


图 2-16 RUN 模式下 CPU 执行的任务

A—清除 I 存储区 B—使用上一个值或替换值对输出执行初始化 C—执行启动 OB D—将物理输入的状态复制到 I 存储器
E—将所有中断事件存储到要在 RUN 模式下处理的队列中 F—启用 Q 存储器到物理输出的写入操作

1—将 Q 存储器写入物理输出 2—将物理输入的状态复制到 I 存储器 3—执行程序循环 OB
4—执行自检诊断 5—在扫描周期的任何阶段处理中断和通信

2. OB 的实现功能

在 S7-1200 PLC 中，OB 控制用户程序的执行，每个 OB 的编号必须唯一，200 以下的一些默认 OB 编号被保留。其他 OB 编号必须大于或等于 200。

CPU 中的特定事件将触发组织块的执行。OB 无法互相调用或通过 FC 或 FB 调用。只有启动事件（例如诊断中断或时间间隔）可以启动 OB 的执行。CPU 按优先等级处理 OB，即先执行优先级较高的 OB，然后执行优先级较低的 OB。最低优先等级为 1（对应主程序循环），最高优先等级为 27（对应时间错误中断）。

OB 控制以下操作：

- 1) 程序循环 OB 在 CPU 处于 RUN 模式时循环执行。主程序块是程序循环 OB。用户在其中放置控制程序的指令以及调用其他用户块。允许使用多个程序循环 OB，它们按编号顺



序执行。OB1 是默认循环 OB。其他程序循环 OB 必须标识为 OB200 或更大。

2) 启动 OB 在 CPU 的工作模式从 STOP 切换到 RUN 时执行一次，包括处于 RUN 模式时和执行 STOP 到 RUN 切换命令时上电。之后将开始执行主“程序循环”OB。允许有多个启动 OB。OB100 是默认启动 OB。其他启动 OB 必须是 OB200 或更大。

3) 通过启动中断 (SRT_DINT) 指令组态事件后，时间延迟 OB 将以指定的时间间隔执行。延迟时间在扩展指令 SRT_DINT 的输入参数中指定。指定的延迟时间结束时，时间延迟 OB 将中断正常的循环程序执行。对任何给定的时间最多可以组态 4 个时间延迟事件，每个组态的时间延迟事件只允许对应一个 OB。时间延迟 OB 必须是 OB200 或更大。

4) 循环中断 OB 以指定的时间间隔执行。循环中断 OB 将按用户定义的时间间隔（例如每隔 2s）中断循环程序执行。最多可以组态 4 个循环中断事件，每个组态的循环中断事件只允许对应一个 OB。该 OB 必须是 OB200 或更大。

5) 硬件中断 OB 在发生相关硬件事件时执行，包括内置数字输入端的上升沿和下降沿事件以及 HSC 事件。硬件中断 OB 将中断正常的循环程序执行来响应硬件事件信号。可以在硬件配置的属性中定义事件。每个组态的硬件事件只允许对应一个 OB。该 OB 必须是 OB200 或更大。

6) 时间错误中断 OB 在检测到时间错误时执行。如果超出最大循环时间，时间错误中断 OB 将中断正常的循环程序执行。最大循环时间在 PLC 的属性中定义。OB80 是唯一支持时间错误事件的 OB。可以组态没有 OB80 时的动作：忽略错误或切换到 STOP 模式。

7) 诊断错误中断 OB 在检测到和报告诊断错误时执行。如果具有诊断功能的模块发现错误（如果模块已启用诊断错误中断），诊断 OB 将中断正常的循环程序执行。OB82 是唯一支持诊断错误事件的 OB。如果程序中没有诊断 OB，则可以组态 CPU 使其忽略错误或切换到 STOP 模式。

2.2.3 S7-1200 PLC 的数据类型

数据类型用于指定数据元素的大小以及如何解释数据。每个指令参数至少支持一种数据类型，而有些参数支持多种数据类型。常见的 S7-1200 PLC 常用数据类型及常量输入事例，见表 2-12。

表 2-12 常见的 S7-1200 PLC 常用数据类型及常量输入事例

| 数据类型 | 大小/位 | 范 围 | 常量输入实例 |
|-------|------|--------------------------------|-------------------|
| Bool | 1 | 0 ~ 1 | TRUE, FALSE, 0, 1 |
| Byte | 8 | 16#00 ~ 16#FF | 16#12, 16#AB |
| Word | 16 | 16#0000 ~ 16#FFFF | 16#ABCD, 16#0001 |
| DWord | 32 | 16#00000000 ~ 16#FFFFFFFF | 16#02468ACE |
| Char | 8 | 16#00 ~ 16#FF | 'A', 'T', '@' |
| Sint | 8 | -128 ~ 127 | 123, -123 |
| Int | 16 | -32,768 ~ 32,767 | 123, -123 |
| Dint | 32 | -2,147,483,648 ~ 2,147,483,647 | 123, -123 |
| USInt | 8 | 0 ~ 255 | 123 |



(续)

| 数据类型 | 大小/位 | 范 围 | 常量输入实例 |
|--------|------|--|--|
| UInt | 16 | 0 ~ 65,535 | 123 |
| UDInt | 32 | 0 ~ 4,294,967,295 | 123 |
| Real | 32 | +/- 1.18 × 10 ⁻³⁸ ~ +/ - 3.40 × 10 ³⁸ | 123.456, -3.4, -1.2E+12, 3.4E-3 |
| LReal | 64 | +/- 2.23 × 10 ⁻³⁰⁸ ~ +/ - 1.79 × 10 ³⁰⁸ | 12345.123456789 -1.2E+40 |
| Time | 32 | T# -24d_20h_31m_23s_648ms ~ T#24d_20h_31m_23s_647ms 存储形式: -2,147,483,648ms ~ +2,147,483,647ms | T#5m_30s 5# -2d T#1d_2h_15m_30x_45ms |
| String | 可变 | 0 ~ 254 字节字符 | 'ABC' |

2.3 技能训练一：数字量扩展模块的使用

2.3.1 任务说明

图 2-17 所示为一典型的半自动计数包装生产线，它是由送料盘和输送带组成。该生产线一般适用于五金、塑料、食品等行业中形状较规则、尺寸较小的产品（如球形、圆柱形，直径或长度小于 50mm）的半自动计数包装，其中计数采用光电开关。

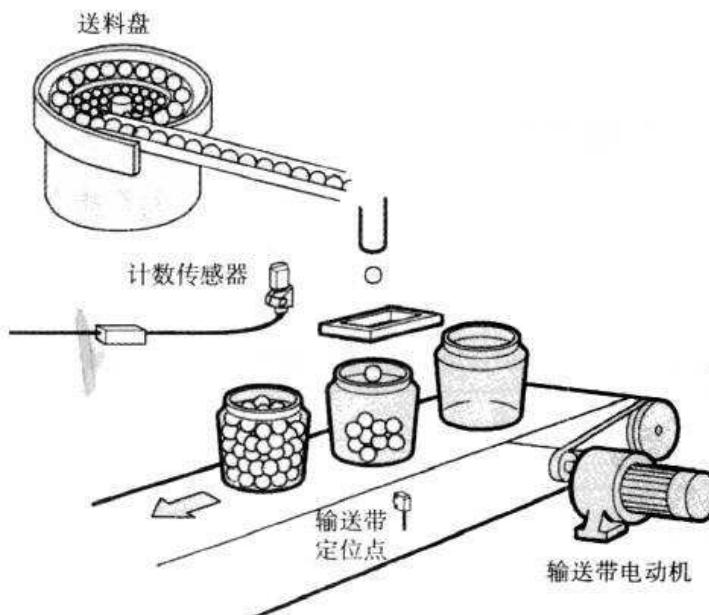


图 2-17 半自动计数包装生产线

现在要求对控制电路进行 S7-1200 PLC 改造设计，要求如下：1) 原来的系统是采用 CPU 1214C DC/DC/DC 主模块，但是要求采用数字量扩展模块进行改造，其 I/O 的起始地址都是从 100 开始；2) 按启动按钮，送料盘电动机起动，开始送料；3) 当包装盒装箱达到

第2章 S7-1200 PLC 扩展模块的应用



设定值时，送料电动机自动停止，计数到指示灯亮；4) 按复位按钮，计数指示灯灭，可以重复2、3、4步，按停止按钮，送料盘电动机可以停止。

2.3.2 电气接线及安装

1. 电气接线

本案例由于新增加的按钮、指示灯、光电开关数量不是很多，采用一个扩展模块SM1223 DI8/DO8 * 24V即可，具体电气接线如图2-18所示。

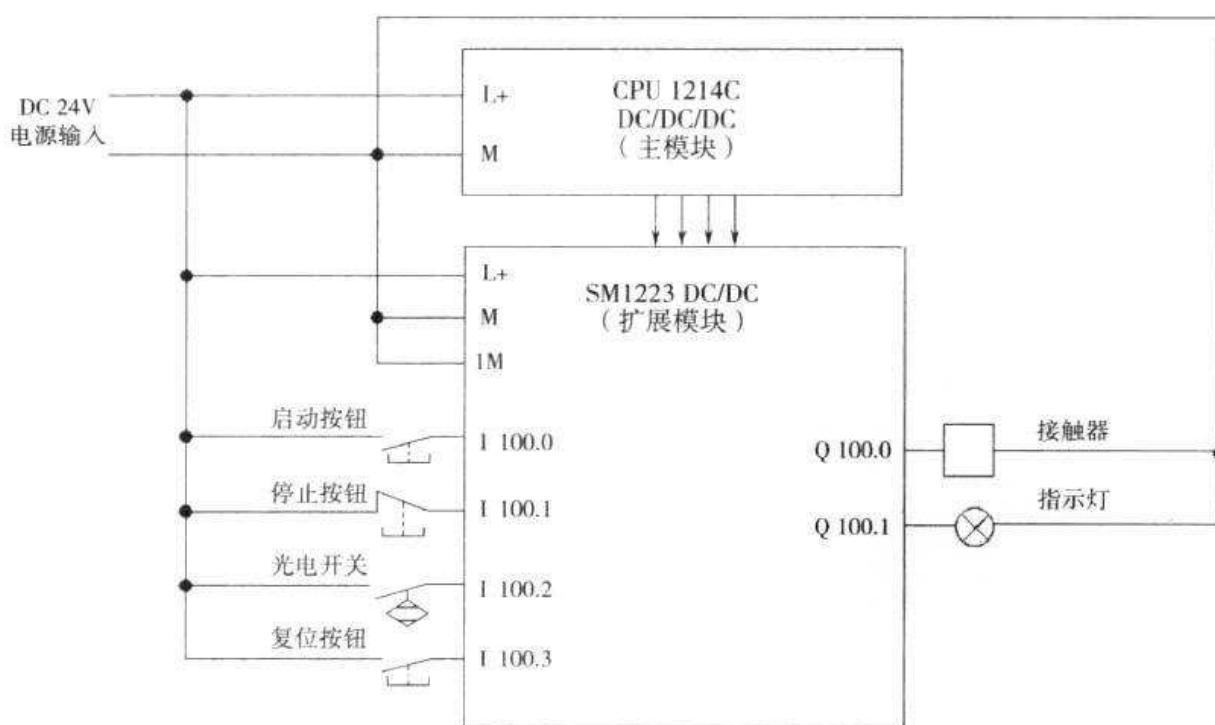


图 2-18 电气接线图

2. 扩展模块(SM)的安装

一般在安装CPU之后再安装扩展模块SM，其安装步骤如下(见图2-19)：

- 1) 卸下CPU右侧的连接器盖，将螺钉旋具插入盖上方的插槽中，将其上方的盖轻轻撬出并卸下盖，收好盖以备再次使用。

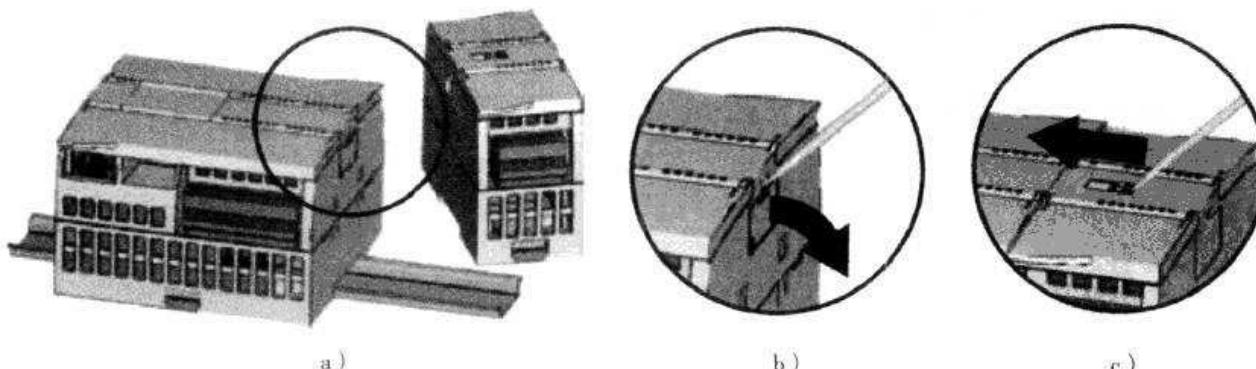


图 2-19 扩展模块的安装



- 2) 将 SM 挂到 DIN 导轨上方，拉出下方的 DIN 导轨卡夹以便将 SM 安装到导轨上。向下转动 CPU 旁的 SM 使其就位并推入下方的卡夹将 SM 锁定到导轨上。
- 3) 将螺钉旋具放到 SM 上方的小接头旁，将小接头滑到最左侧，使总线连接器伸到 CPU 中。伸出总线连接器即为 SM 建立了机械和电气连接。

2.3.3 PLC 编程

1. 设备配置

如图 2-20 所示，首先根据项目 1 的步骤添加 S7-1200 CPU1214C DC/DC/DC，在 CPU 右侧可以看到有 8 个虚框，这说明可以添加 8 个扩展模块；而在 CPU 左边可以添加 101 ~ 103 共计 3 个通信模块（该内容将在后续项目中进行介绍）。

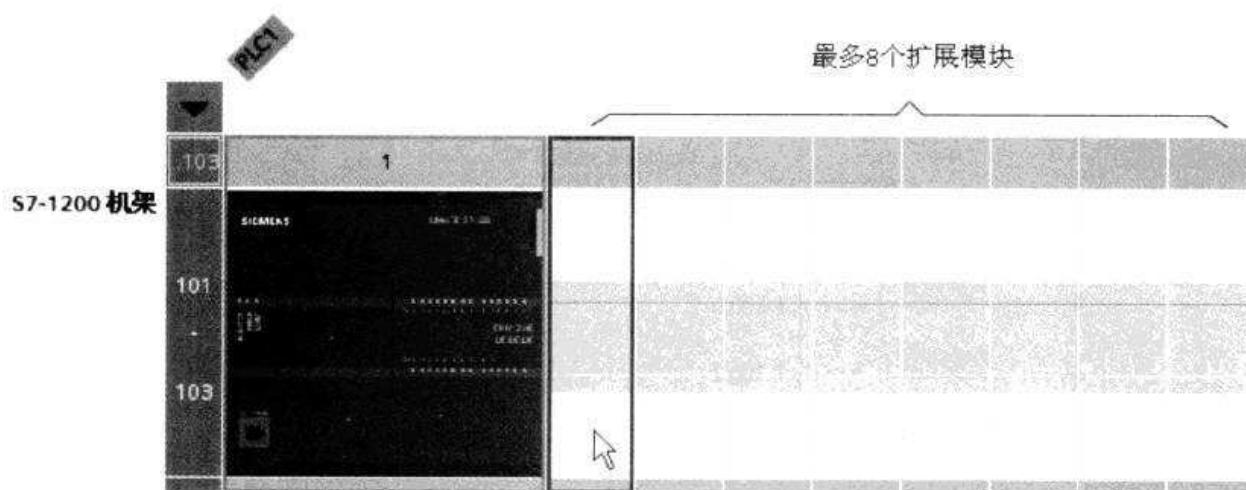


图 2-20 S7-1200 的扩展

在设备视图中有 3 种方法可将扩展模块添加到 S7-1200 机架：1) 如果有可用的有效插槽，则在硬件目录中双击模块；2) 通过拖放操作将该模块从硬件目录移动到图形或表格区域内可用的有效插槽中；3) 选择硬件目录中相应模块的快捷菜单中的“复制”，然后在图形或表格区域中可用的有效插槽上选择相应快捷菜单中的“粘贴”。

图 2-21 所示为本案例中需要选择的 DI/DO 模块，即 6ES7223-1BH30-0XB0，然后将它按照上述 3 种方法中的任何一种添加到 S7-1200 机架中（见图 2-22）。

待扩展模块添加完毕之后，就可以对该模块进行属性设置，比如将图 2-23 所示的 DI/DO 模块的常规属性中的 I/O 起始地址从“8”改为图 2-24 所示的“100”。与其他小 PLC 不同，S7-



图 2-21 选择合适的 DI/DO 扩展模块

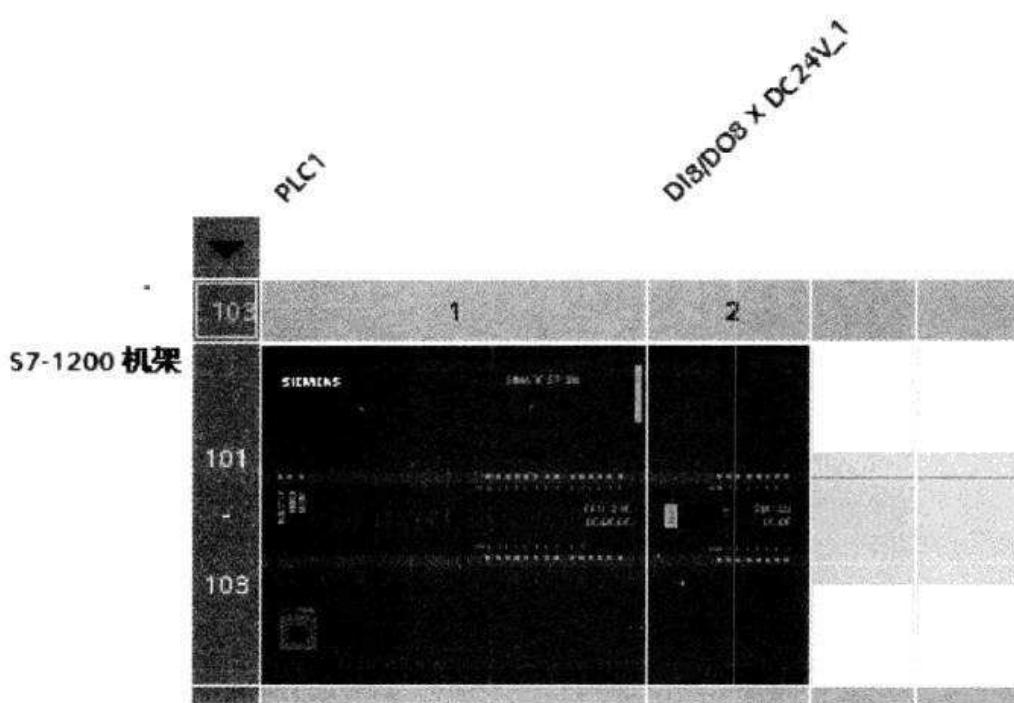


图 2-22 将 DI/DO 扩展模块添加到 S7-1200 机架中

1200 PLC 具有任意组态 I/O 地址的特性，这对于扩展模块来说尤其重要。

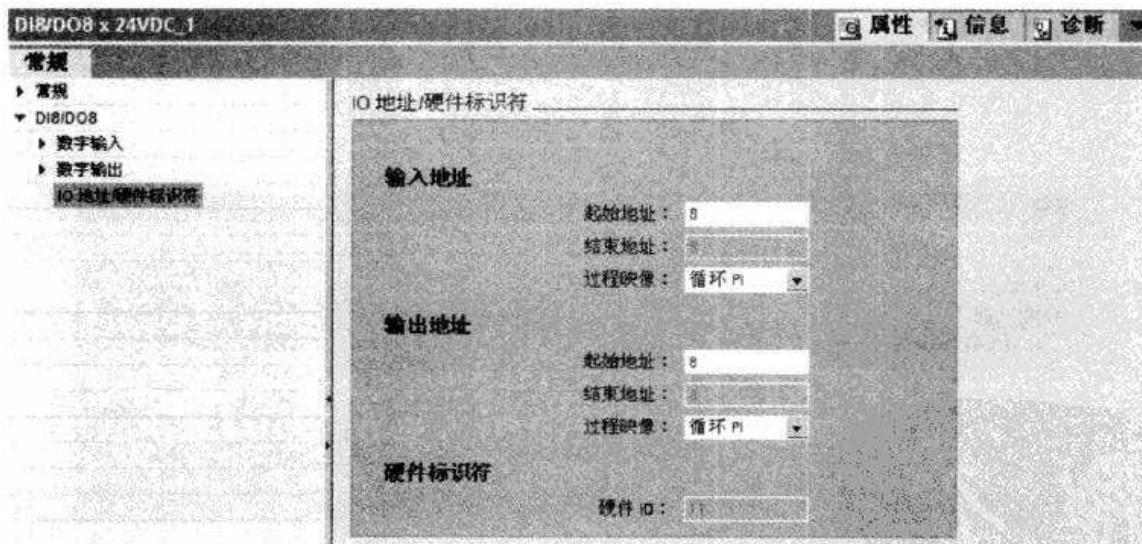


图 2-23 DI/DO 模块的常规属性设置

在 DI/DO 模块中，还可以以图 2-25 所示设置输入过滤器属性，即输入合适的滤波时间，比如该模块 8 个输入点，可以分成两组分别进行设置。

图 2-26 所示为 DI/DO 模块的数字输出属性设置，如 CPU 停止时，数字输出可以选择保持上一个值还是使用替换值，这对于工业生产中的稳定性和安全性都是一个重要的因素。当使用替换值时，它可以设置通道的属性，即从 RUN 切换到 STOP 时，替换值为 1。

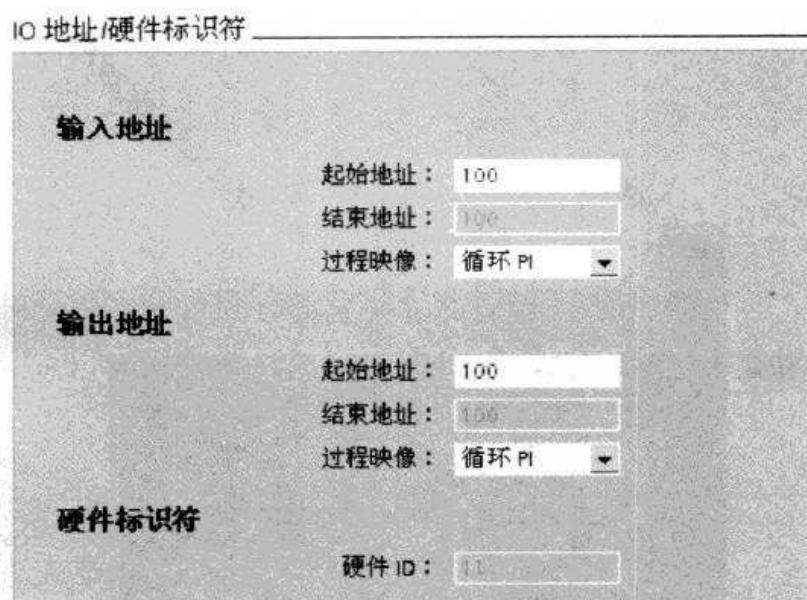


图 2-24 I/O 地址的更改

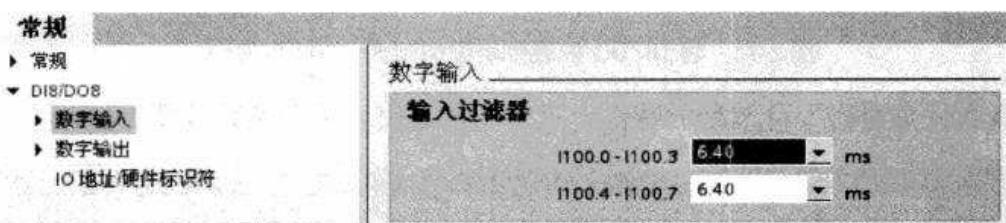


图 2-25 输入过滤器属性

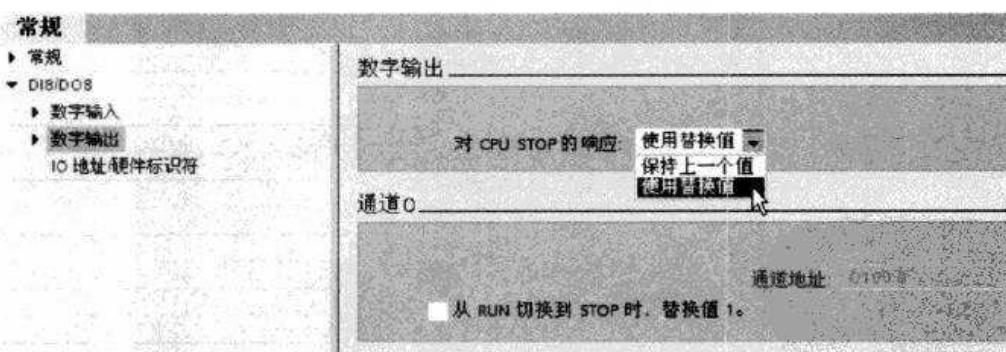


图 2-26 数字输出属性设置

2. 计数器指令的应用

可使用计数器指令对内部程序事件和外部过程事件进行计数：CTU 是加计数器；CTD 是减计数器；CTUD 是加减计数器。图 2-27 所示为选择计数器指令。

每个计数器都使用数据块中存储的结构来保存计数器数据。用户在编辑器中放置计数器指令时分配相应的数据块。这些指令使用软件计数器，软件计数器的最大计数速率受其所在 OB 的执行速率限制。指令所在 OB 的执行频率必须足够高，以检测 CU 或 CD 输入的所有跳

第2章 S7-1200 PLC 扩展模块的应用



图 2-27 选择计数器指令

变。在功能块中放置计数器指令后，可以选择单背景或多重背景数据块选项（见图 2-28）。计数器指令的变量说明见表 2-13。

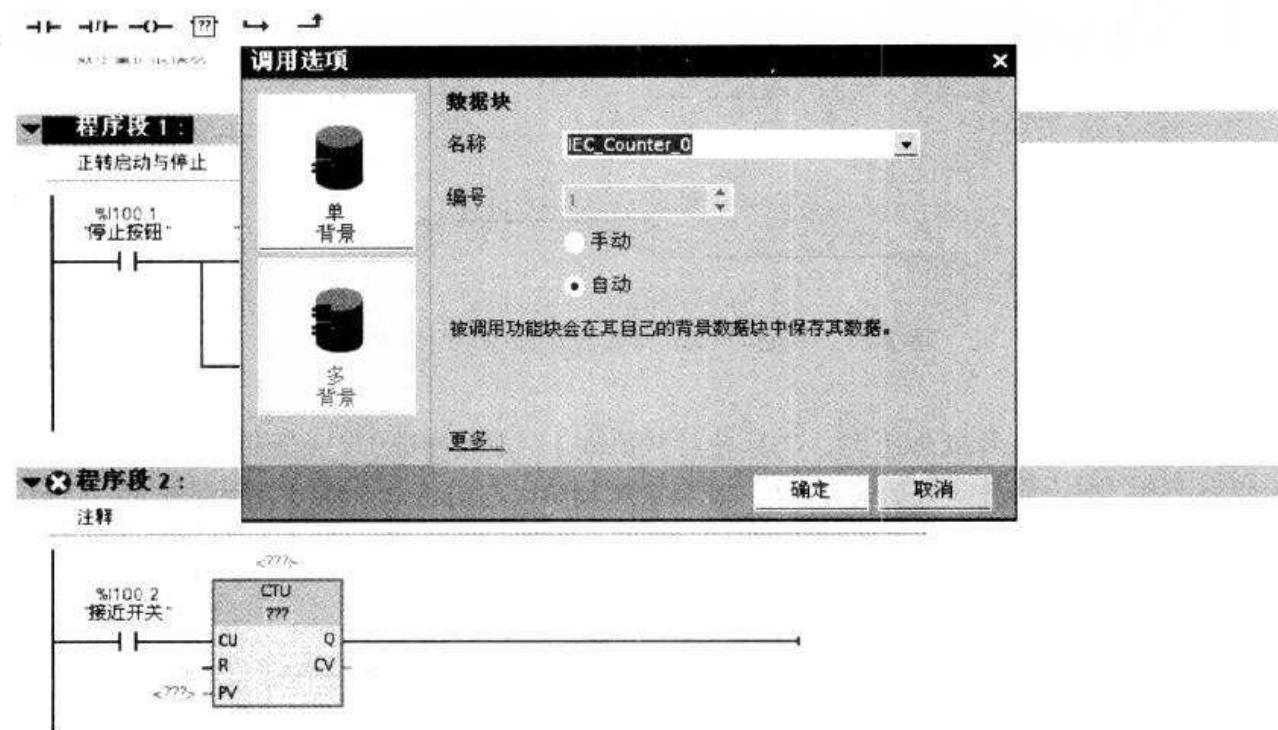


图 2-28 选择单背景或多重背景数据块选项

表 2-13 计数器指令的变量说明

| 参 数 | 数据类型 | 说 明 |
|-----------------|--------------------------------|-----------------|
| CU、CD | Bool | 加计数或减计数，按加或减一计数 |
| R (CTU、CTUD) | Bool | 将计数值重置为零 |
| LOAD (CTD、CTUD) | Bool | 预设值的装载控制 |
| PV | SInt、Int、DInt、USInt、UInt、UDInt | 预设计数值 |
| Q、QU | Bool | CV ≥ PV 时为真 |
| QD | Bool | CV ≤ 0 时为真 |
| CV | SInt、Int、DInt、USInt、UInt、UDInt | 当前计数值 |



计数值的数值范围取决于所选的数据类型。如果计数值是无符号整型数，则可以减计数到零或加计数到范围限值。如果计数值是有符号整数，则可以减计数到负整数限值或加计数到正整数限值。

(1) CTU 指令

参数 CU 的值从 0 变为 1 时，CTU 使计数值加 1。如果参数 CV（当前计数值）的值大于或等于参数 PV（预设计数值）的值，则计数器输出参数 Q = 1。如果复位参数 R 的值从 0 变为 1，则当前计数值复位为 0。图 2-29 显示了计数值是无符号整数时的 CTU 时序图（其中，PV = 3）。

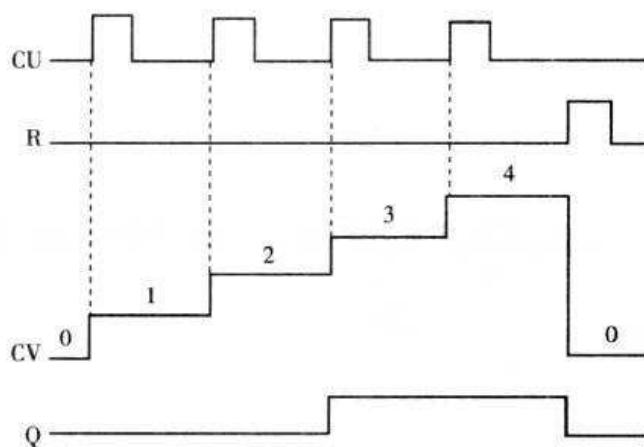


图 2-29 计数值是无符号整数时的 CTU 时序图

(2) CTD 指令

参数 CD 的值从 0 变为 1 时，CTD 使计数值减 1。如果参数 CV（当前计数值）的值等于或小于 0，则计数器输出参数 Q = 1。如果参数 LOAD 的值从 0 变为 1，则参数 PV（预设值）的值将作为新的 CV（当前计数值）装载到计数器。图 2-30 显示了计数值是无符号整数时的 CTD 时序图（其中，PV = 3）。

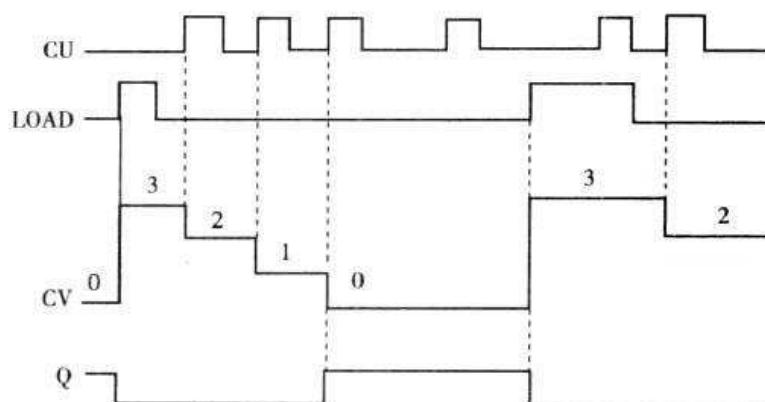


图 2-30 计数值是无符号整数时的 CTD 时序图

(3) CTUD 指令

加计数（Count Up, CU）或减计数（Count Down, CD）输入的值从 0 跳变为 1 时，CTUD 会使计数值加 1 或减 1。如果参数 CV（当前计数值）的值大于或等于参数 PV（预设

第2章 S7-1200 PLC 扩展模块的应用



值) 的值, 则计数器输出参数 QU = 1。如果参数 CV 的值小于或等于 0, 则计数器输出参数 QD = 1。如果参数 LOAD 的值从 0 变为 1, 则参数 PV (预设值) 的值将作为新的 CV (当前计数值) 装载到计数器。如果复位参数 R 的值从 0 变为 1, 则当前计数值复位为 0。图 2-31 显示了计数值是无符号整数时的 CTUD 时序图 (其中, PV = 4)。

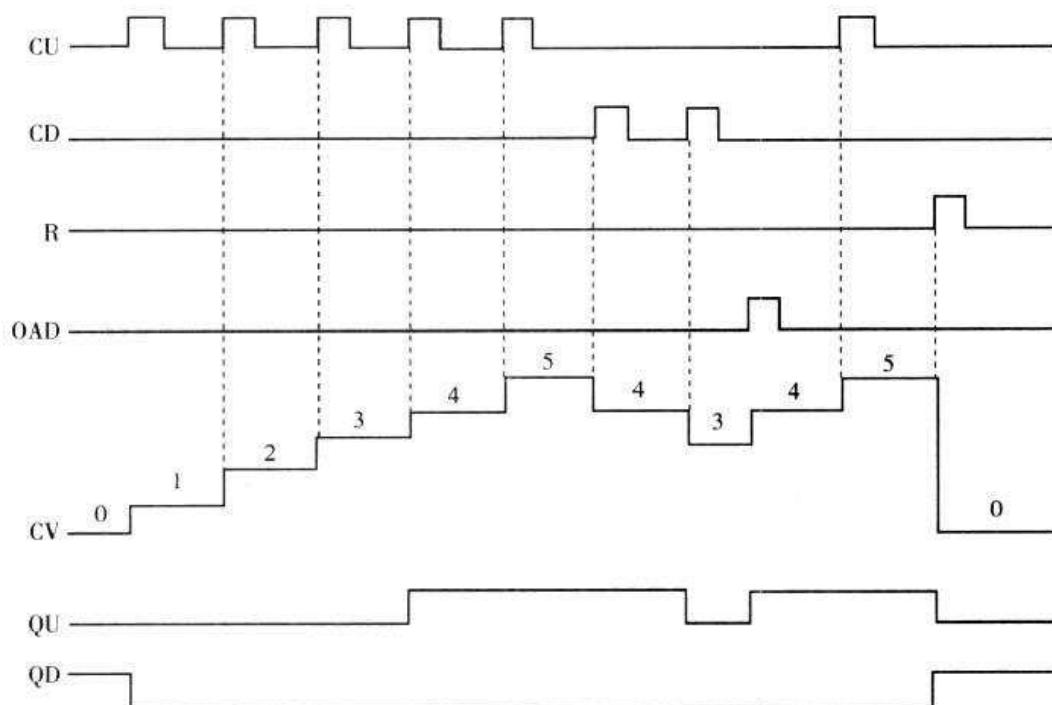


图 2-31 计数值是无符号整数时的 CTUD 时序图

图 2-32 所示为 CTUD 指令的数据块变量说明。

| 名称 | 数据类型 | 初始值 |
|----------------|------|---------|
| 1 ▼ Static | | |
| 2 COUNT_UP | Bool | false |
| 3 COUNT_DOWN | Bool | false |
| 4 RESET | Bool | false |
| 5 LOAD | Bool | false |
| 6 Q_UP | Bool | false |
| 7 Q_DOWN | Bool | false |
| 8 PAD | Byte | B#16#00 |
| 9 PRESET_VALUE | Int | 0 |
| 10 COUNT_VALUE | Int | 0 |

图 2-32 CTUD 的数据块变量说明

3. 变量定义

对于半自动计数包装生产线, 根据电气接线图和计数器的指令的要求, 可以定义变量, 见表 2-14。



表 2-14 变量定义

| | 名称 | 数据类型 | 地址 |
|---|--------|------|---------|
| 1 | 启动按钮 | Bool | %I100.0 |
| 2 | 接触器 | Bool | %Q100.0 |
| 3 | 停止按钮 | Bool | %I100.1 |
| 4 | 接近开关 | Bool | %I100.2 |
| 5 | 实时计数值 | Int | %MW0 |
| 6 | 计数到 | Bool | %M100.0 |
| 7 | 复位按钮 | Bool | %I100.3 |
| 8 | 计数到上升沿 | Bool | %M100.1 |
| 9 | 指示灯 | Bool | %Q100.1 |

4. 梯形图 PLC 编程

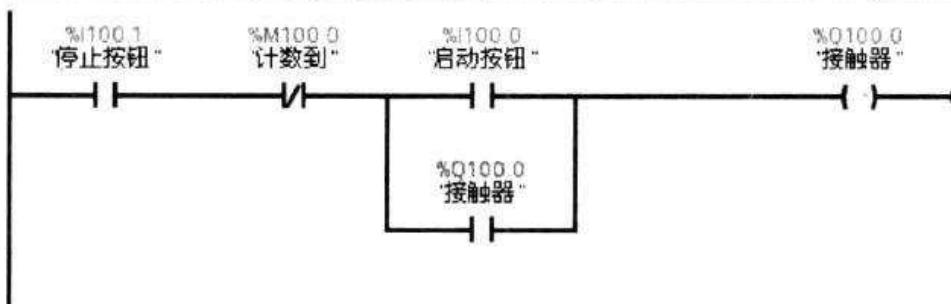
图 2-33 所示为半自动计数包装生产线应用数字量扩展模块的 PLC 梯形图。

▼ 块标题：

数字量扩展模块的应用

▼ 程序段 1：

正转启动与停止



▼ 程序段 2：

计数器功能

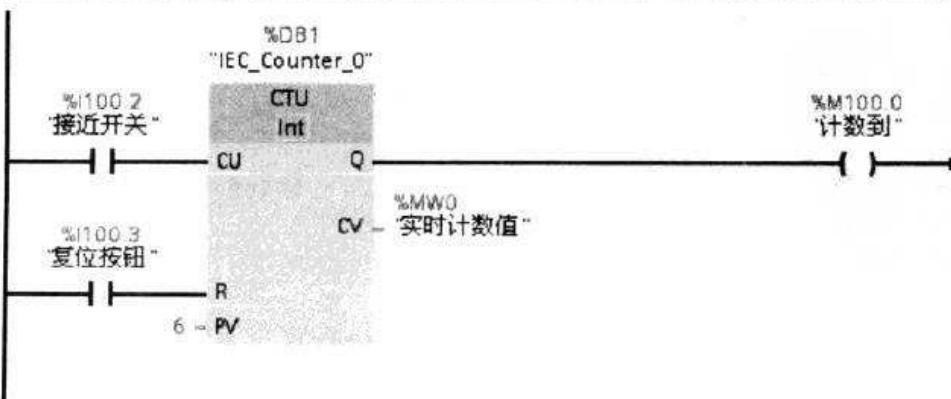


图 2-33 半自动计数包装生产线应用数字量扩展模块的 PLC 梯形图



▼ 程序段 3：

计数到指示灯亮



▼ 程序段 4：

计数到指示灯灭



图 2-33 半自动计数包装生产线应用数字量扩展模块的 PLC 梯形图（续）

2.4 技能训练二：模拟量扩展模块的应用

2.4.1 任务说明

图 2-34 所示的食品机械中，传输系统被大量地使用，比如蛋糕烘烤前必须由传输带进行送料，并按照烘烤工艺匀速地通过烘烤箱。以前这样的设备调速基本都采用手动机械式有级变速（比如更换带轮大小或者齿轮箱变速比等），但如今采用变频调速后能大大扩展调速范围，且能实现无级调速。

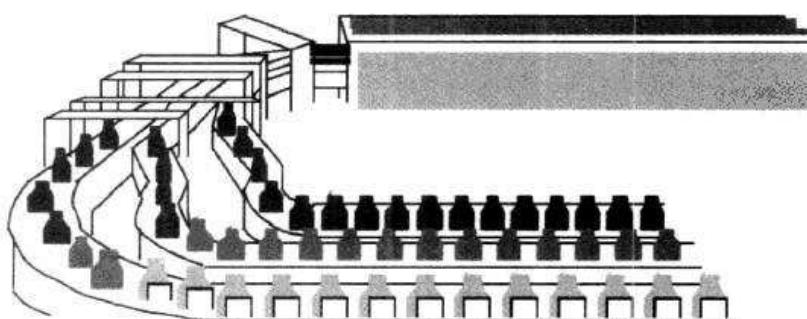


图 2-34 食品输送带传动

现在要求对该输送带进行控制系统设计：1) 传动采用变频器控制，其启动与停止通过与 PLC 连接的启动与停止按钮来进行；2) 变频器的速度控制分为本地和远程两种，以选择开关来进行切换；3) 当选择开关置于“本地”时，其速度分别由 3 个速度开关来设定；4) 当选择开关置于“远程”时，其速度来自于上位机的电压 DC 0 ~ 10V 信号，并要求对该信



号进行“增益”（范围在 0.5 ~ 2 之间）。

2.4.2 模拟量模块选型与电气接线

1. 模拟量输入和输出模块的特性说明

食品输送带传动的 PLC 控制中，主要包括本地操作单元、上位机、PLC 与变频器，这四者之间的联系如图 2-35 所示。

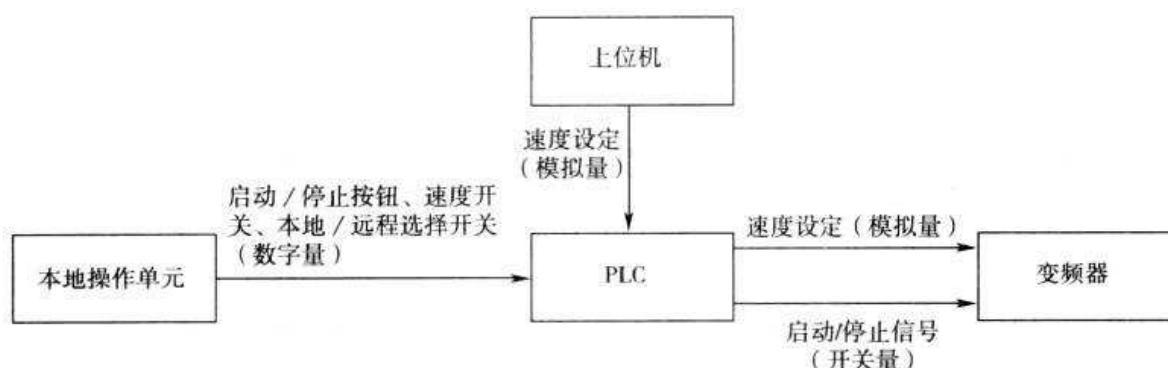


图 2-35 食品输送带传动的 PLC 控制示意

根据 S7-1200 PLC 的特点，该款 PLC 内置了 2 点模拟量输入，但是没有模拟量输出，需要增加模拟量扩展模块，常见的模拟量输入模块和输出模块的特性见表 2-15 和表 2-16。

表 2-15 模拟量输入模块的特性

| 型 号 | SM 1231 AI 4 × 13 位 | SM 1231 AI 8 × 13 位 | SM 1234 AI 4 × 13 位 AQ 2 × 14 位 |
|----------------------|--|---------------------|---------------------------------|
| 订货号 (MLFB) | 6ES7 231-4HD30-0XB0 | 6ES7 231-4HF30-0XB0 | 6ES7 234-4HE30-0XB0 |
| 输入路数 | 4 | 8 | 4 |
| 输入类型 | 电压或电流 (差动): 可两个选为一组 | | |
| 输入范围 | ±10V、±5V、±2.5V 或 0 ~ 20mA | | |
| 输入满量程范围 (数据字) | -27, 648 ~ 27, 648 | | |
| 输入过冲/下冲范围 (数据字) | 电压: 32, 511 ~ 27, 649 / -27, 649 ~ -32, 512 电流: 32, 511 ~ 27, 649 / 0 ~ -4864 | | |
| 输入上溢/下溢 (数据字) | 电压: 32, 767 ~ 32, 512 / -32, 513 ~ -32, 768 电流 32, 767 ~ 32, 512 / -4865 ~ -32, 768 | | |
| 精度 | 12 位 + 符号位 | | |
| 最大耐压/耐流 | ±35V / ±40mA | | |
| 平滑 | 无、弱、中或强 | | |
| 噪声抑制 | 400、60、50 或 10Hz | | |
| 阻抗 | ≥9MΩ (电压) / 250Ω (电流) | | |
| 精度 (25°C / 0 ~ 55°C) | 满量程的 ±0.1% / ±0.2% | | |
| 模数转换时间 | 625μs (400Hz 抑制) | | |
| 共模抑制 | 40dB, DC 到 60Hz | | |
| 工作信号范围 | 信号加共模电压必须小于 +12V 且大于 -12V | | |
| 电缆长度 (m) | 100m, 屏蔽双绞线 | | |

第2章 S7-1200 PLC 扩展模块的应用



表 2-16 模拟量输出模块的特性

| 型 号 | SM 1232 AQ 2×14 位 | SM 1232 AQ 4×14 位 | SM 1234 AI 4×13 位 AQ 2×14 位 |
|---|---|---------------------|-----------------------------|
| 订货号 (MLFB) | 6ES7 232-4HB30-0XB0 | 6ES7 232-4HD30-0XB0 | 6ES7 234-4HE30-0XB0 |
| 输出路数 | 2 | 4 | 2 |
| 类型 | 电压或电流 | | |
| 范围 | $\pm 10V$ 或 $0 \sim 20mA$ | | |
| 精度 | 电压: 14 位; 电流: 13 位 | | |
| 满量程范围 (数据字) | 电压: $-27, 648 \sim 27, 648$; 电流: $0 \sim 27, 648$ | | |
| 精度 ($25^{\circ}C / 0 \sim 55^{\circ}C$) | 满量程的 $\pm 0.3\% / \pm 0.6\%$ | | |
| 稳定时间 (新值的 95%) | 电压: $300\mu s$ (R)、 $750\mu s$ ($1\mu F$); 电流: $600\mu s$ ($1mH$)、 $2ms$ ($10mH$) | | |
| 负载阻抗 | 电压: $\geq 1000\Omega$; 电流: $\leq 600\Omega$ | | |
| RUN 到 STOP 时的行为 | 上一个值或替换值 (默认值为 0) | | |
| 电缆长度 (m) | 100m, 屏蔽双绞线 | | |

为了确保该食品设备的后续升级和改造, 本方案选用了具有 4 点模拟量输入和 2 点模拟量输出的 SM1234 4 * AI/2 * AQ 模块。

2. 电气接线

图 2-36 所示为食品输送带传动的 PLC 电气接线。需要注意的是, SM1234 模块的输入和输出接线跟电压或电流信号类型无关, 只需要在硬件配置中进行相应设定即可。

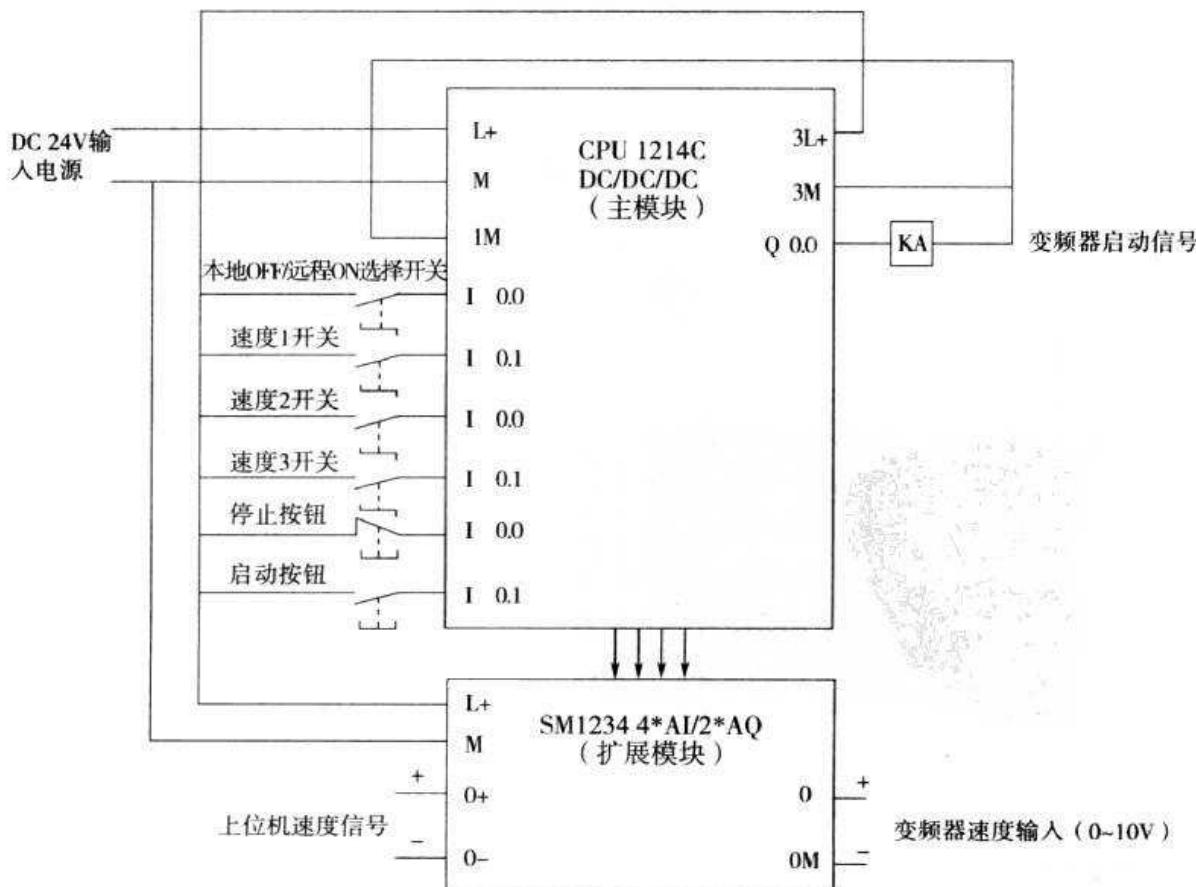


图 2-36 食品输送带传动的 PLC 电气接线



2.4.3 编程软件的安装

1. 硬件配置

在 CPU1214C DC/DC/DC 的基础上，从硬件目录中选择 SM1234 模块（见图 2-37），添加后的结果如图 2-38 所示。

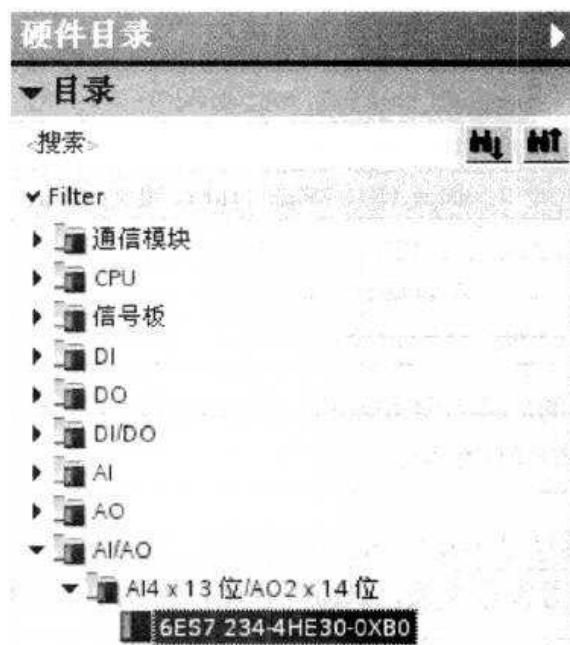


图 2-37 选择 SM1234 模块

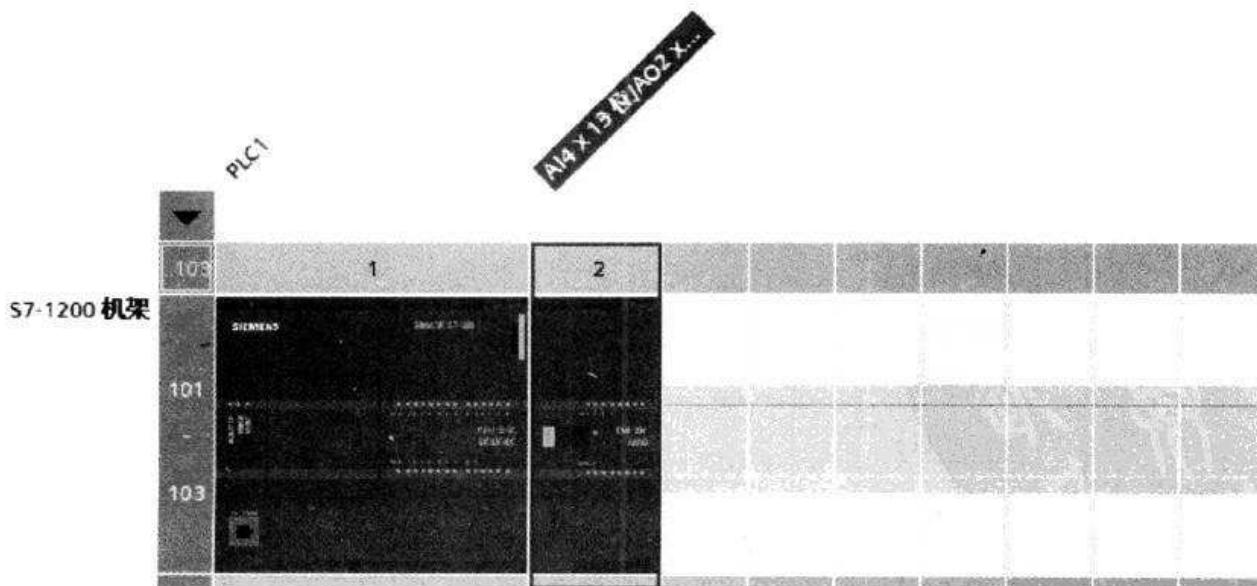


图 2-38 将模拟量模块添加到 S7-1200 机架

如图 2-39 所示，用户可以在硬件组态设置中定义 SM1234 模块的 I/O 地址，地址的范围为 0 ~ 1023。

第2章 S7-1200 PLC 扩展模块的应用

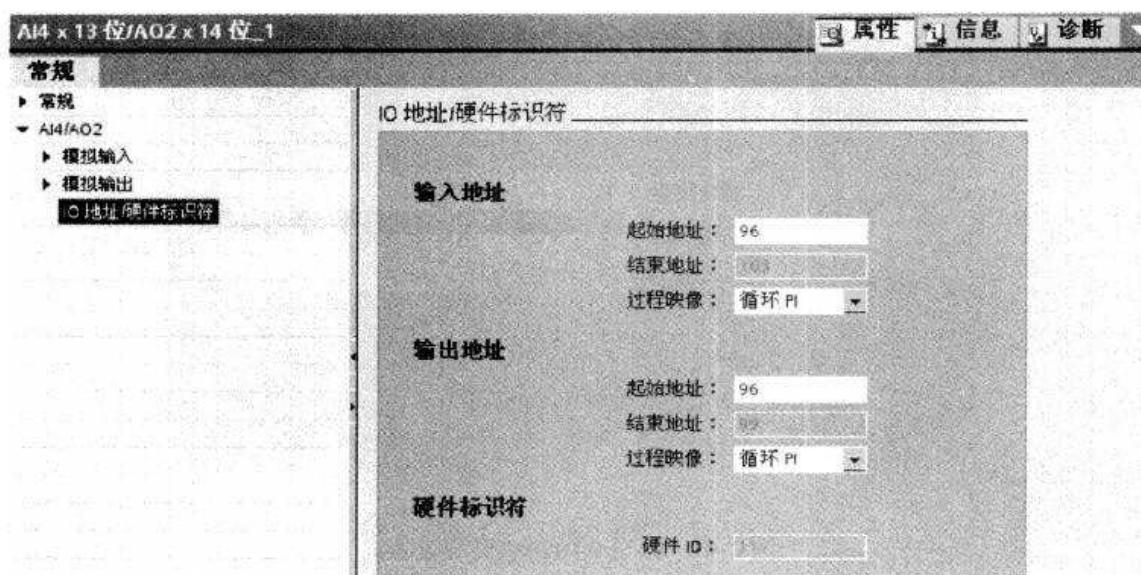


图 2-39 定义 SM1234 模块的 I/O 地址

由于现场电磁环境的影响，模拟量模块会出现数据失真或漂移，这时可以设置滤波属性，如图 2-40 所示，选择使用 10Hz/50Hz/ 60Hz/ 400Hz 滤波，以抗现场的干扰。

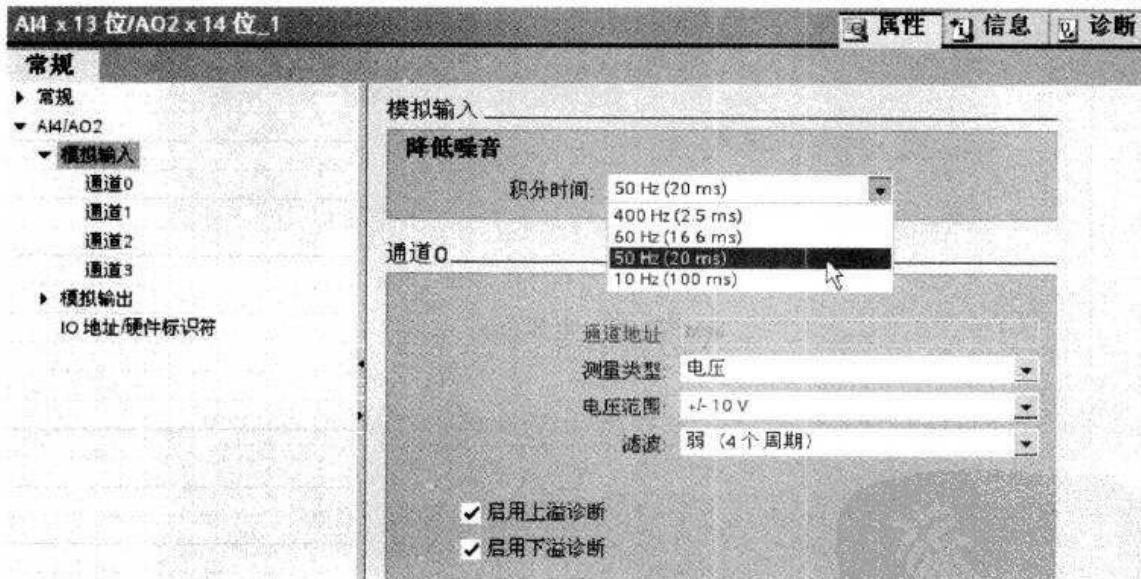


图 2-40 模拟输入的积分时间等属性

对于模拟量输入信号是电压或是电流则可以通过图 2-41 所示的测量类型进行设置。如选择电压类型，则可以选择相应的电压范围值（见图 2-42）。

同时可以根据输入动态响应的高低，选择输入平滑的弱与强（见图 2-43）。

对于模拟输出而言，图 2-44 所示显示了模拟输出的一些属性，如对 CPU STOP 的响应，还有图 2-45 所示的模拟输出类型等。

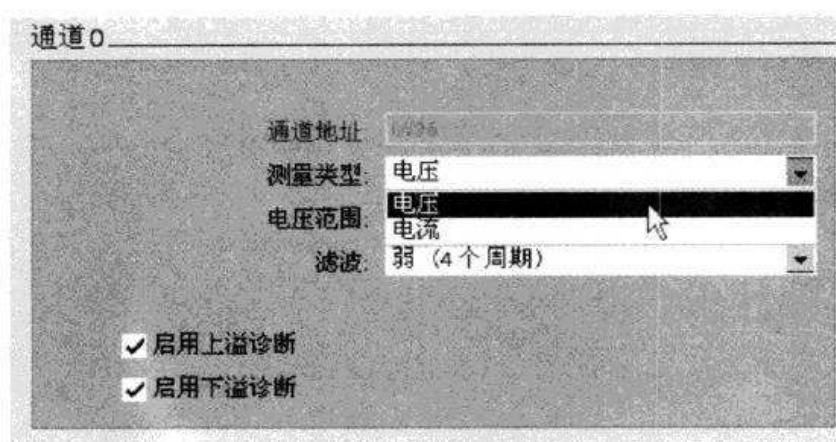


图 2-41 通道 0 的测量类型设置

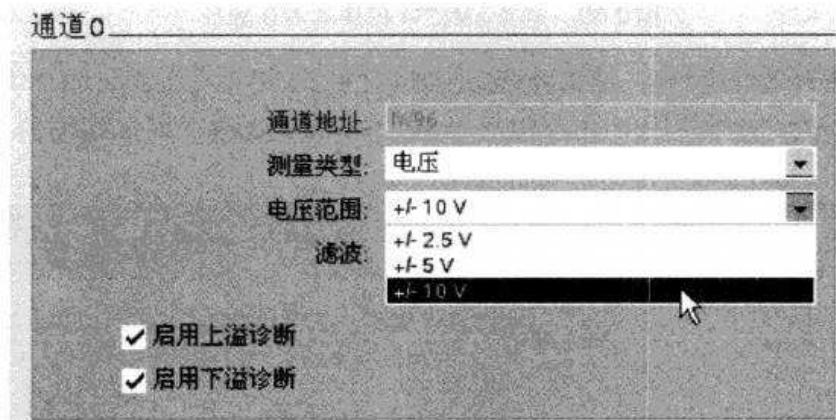


图 2-42 电压范围

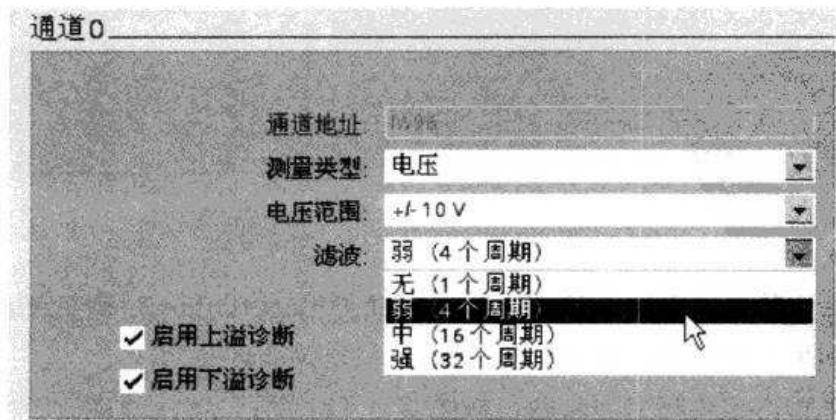


图 2-43 滤波属性的设置

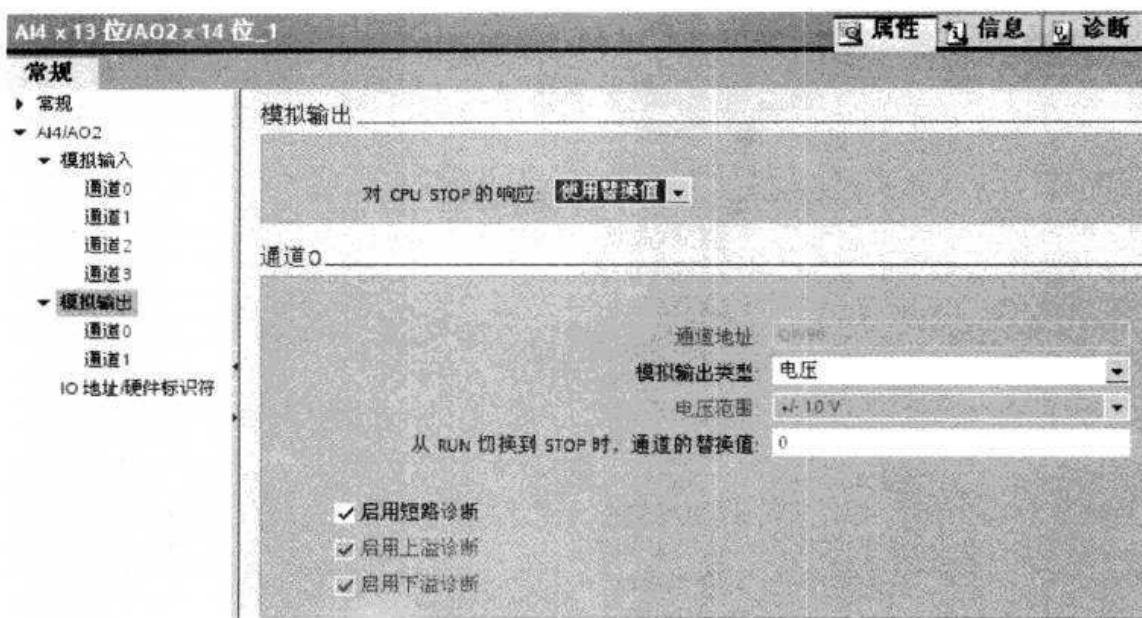


图 2-44 模拟输出属性

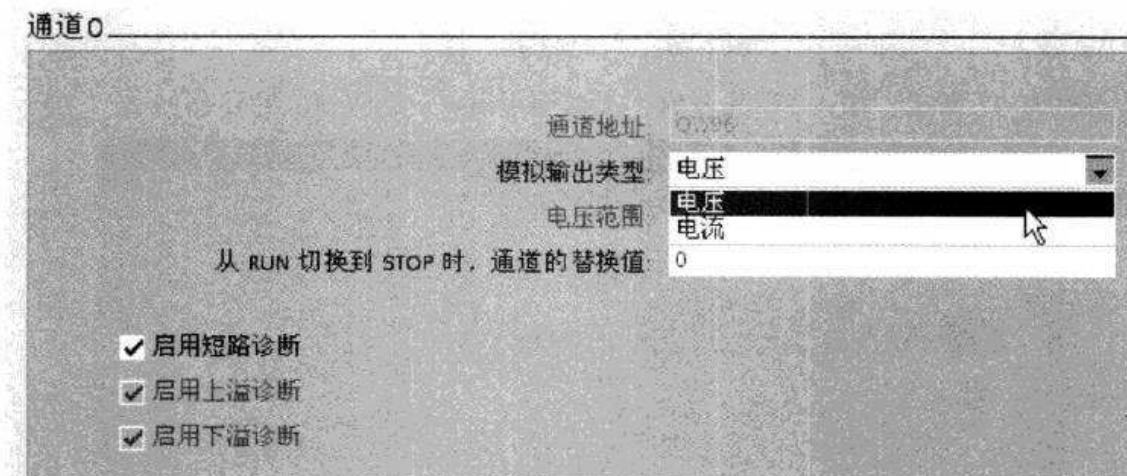


图 2-45 通道 0 模拟输出类型

2. 添加 FC

在食品输送带传动的 PLC 控制中，对于“远程功能”使用了 FC。FC 通常用于对一组输入值执行特定运算的代码块。FC 将此运算结果存储在存储器位置。

使用 FC 可执行以下任务：

- 执行标准和可重复使用的运算，如数学计算。
- 执行工艺功能，如通过使用位逻辑运算进行单独控制。

FC 也可以在程序中的不同位置多次调用。如此重复使用简化了对经常重复发生的任务的编程。FC 不具有相关的背景数据块 (DB)。对于用于计算该运算的临时数据，FC 采用了局部数据堆栈。不保存临时数据。要长期存储数据，可将输出值赋给全局存储器，如 M 存储器或全局 DB。



添加 FC 的步骤：

1) 如图 2-46 所示，选择“程序块”下的“添加新块”命令。

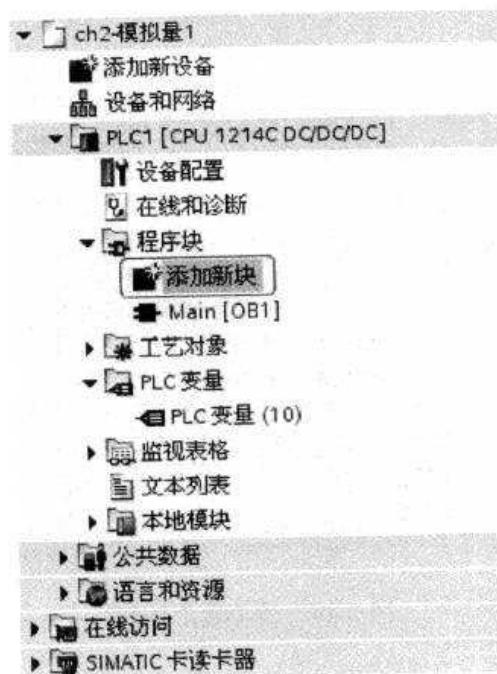


图 2-46 “添加新块”命令

2) 如图 2-47 所示，输入 FC 的名称，选择编程语言，以及 FC 编号等。

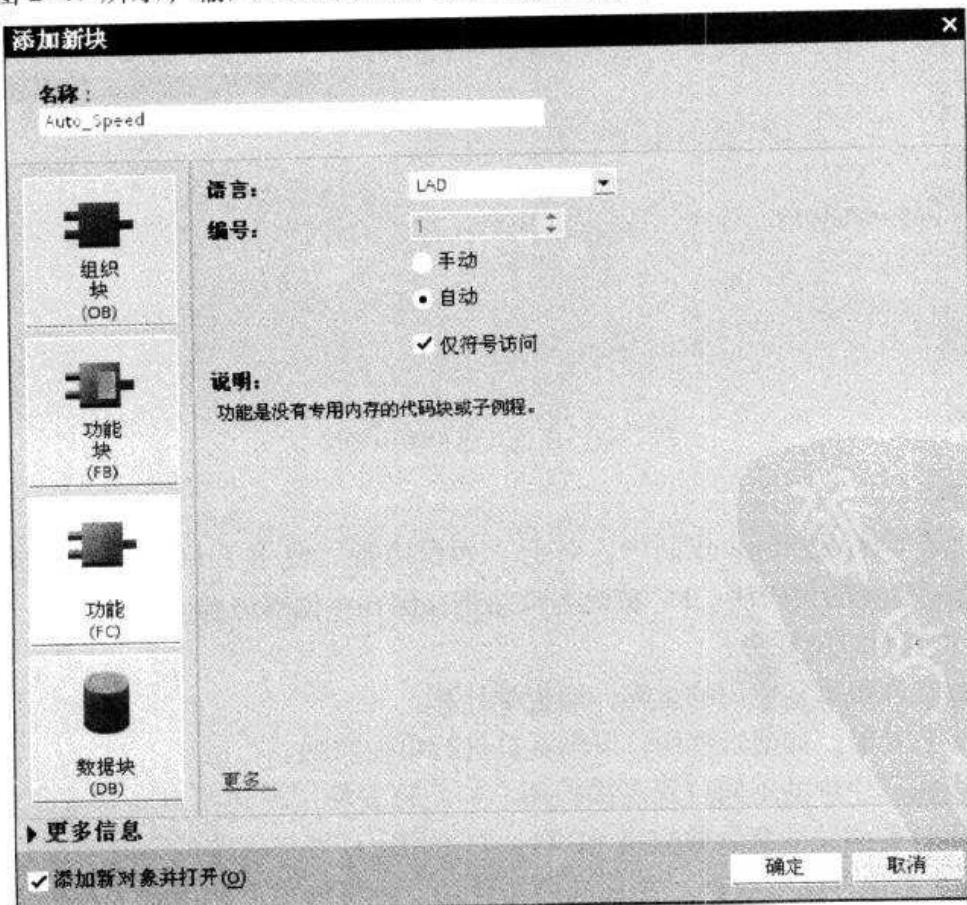


图 2-47 FC 块的功能选择

第2章 S7-1200 PLC 扩展模块的应用



3) 定义 FC 的接口参数见表 2-17, 包括输入、输出、中间变量等。需要注意的是, FC 的返回值和输出参数是不同的: 因为 FC 的返回值只有一个, 并且主要用于程序和用户交换数据; FC 的输出参数可以有多个, 并且主要用于程序与用户交换数据。

表 2-17 定义 FC 的接口参数

| | | 数据类型 | 注释 |
|---|---------|------|-----------|
| 1 | Input | | |
| 2 | Ana_in | Word | 模拟量(整数)输入 |
| 3 | Kp | Real | 增益值 |
| 4 | Output | | |
| 5 | Ana_out | Word | 模拟量(整数)输出 |
| 6 | InOut | | |
| 7 | Temp | | |
| 8 | Temp1 | Real | 中间变量(实数) |

4) 如图 2-48 所示为主程序 OB1 梯形图。

▼ 块标题:

远程功能

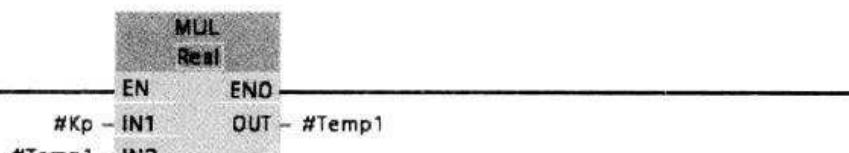
▼ 程序段 1:

将输入整数类型转换为实数类型



▼ 程序段 2:

进行增益相乘



▼ 程序段 3:

将实数类型转为整数类型

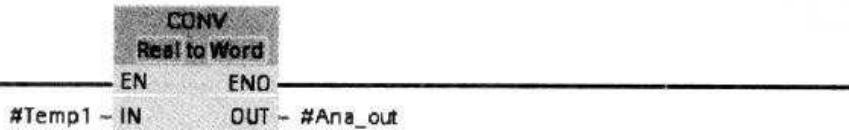


图 2-48 主程序 OB1 梯形图



3. 添加 FB

在食品输送带传动的 PLC 控制中，对于“本地功能”使用了 FB。FB 是使用背景数据块保存其参数和静态数据的代码块。FB 具有位于数据块或背景 DB 中的变量存储器。背景 DB 提供与 FB 的实例（或调用）关联的一块存储区并在 FB 完成后存储数据。可将不同的背景 DB 与 FB 的不同调用进行关联。通过背景 DB 可使用一个通用 FB 控制多个设备。通过使一个代码块对 FB 和背景 DB 进行调用，来构建程序。然后，CPU 执行该 FB 中的程序代码，并将块参数和静态局部数据存储在背景 DB 中。FB 执行完成后，CPU 会返回到调用该 FB 的代码块中。背景 DB 保留该 FB 实例的值。随后在同一扫描周期或其他扫描周期中调用该功能块时可使用这些值。

具体步骤如下：

- 1) 如图 2-49 所示，添加新块“Local_Speed”。

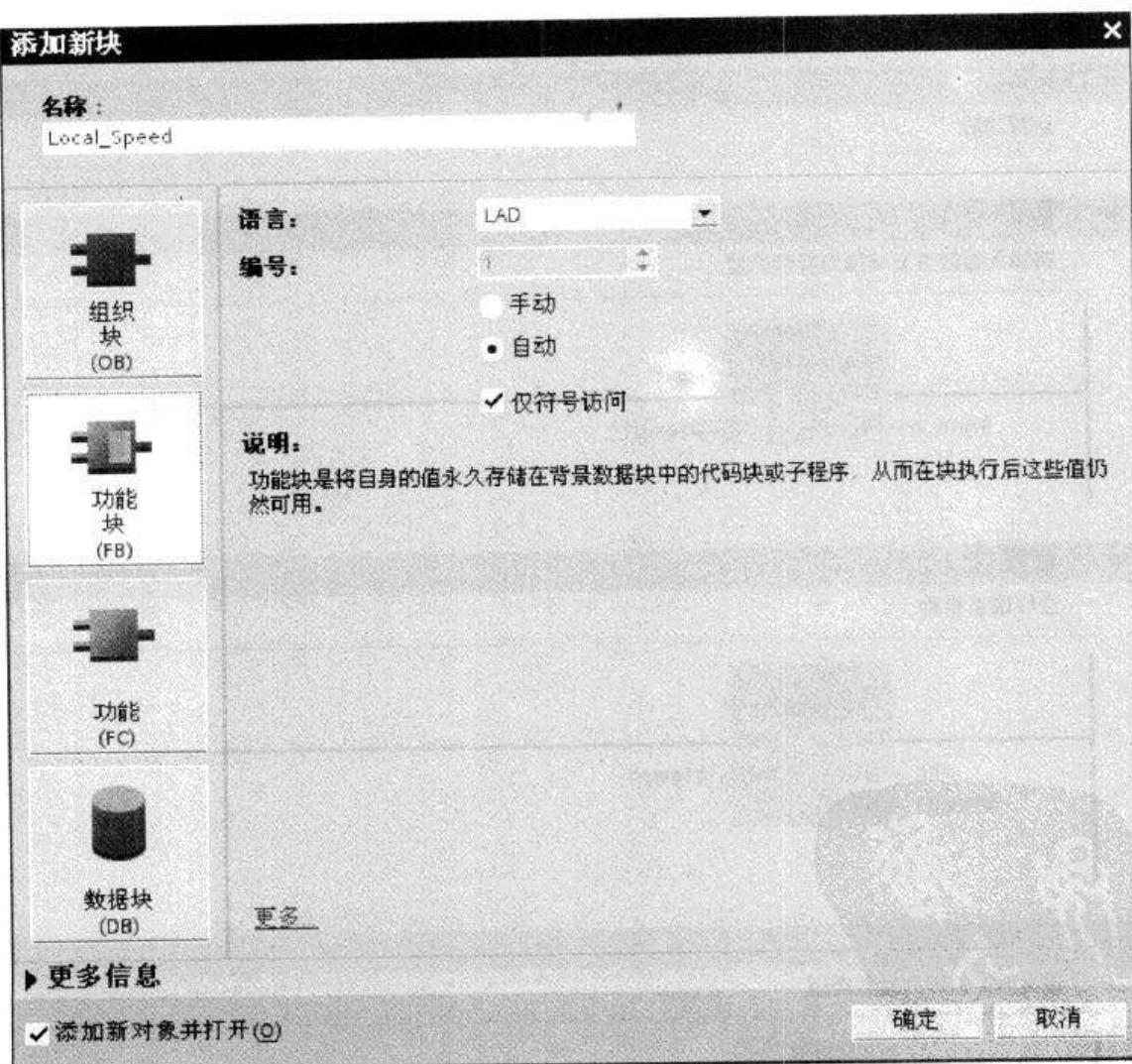


图 2-49 添加新块

- 2) 定义“Local_Speed”的接口参数见表 2-18。

用户通常使用 FB 控制在一个扫描周期内其未完成的运行任务或设备的运行。要存储运行参数以便从一个扫描快速访问到下一个扫描，用户程序中的每一个 FB 都具有一个或多个

第2章 S7-1200 PLC 扩展模块的应用



背景 DB。调用 FB 时，也需要指定包含块参数以及用于该调用或 FB “实例”的静态局部数据的背景 DB。FB 完成执行后，背景 DB 将保留这些值。

表 2-18 定义“Local_Speed”的接口参数

| 名称 | 数据类型 | 默认值 | 保持性 | 注释 |
|------------|------|-----|------------|-------------|
| 1 ▼ Input | | | | |
| 2 Select_1 | Bool | 0 | Non-Retain | 选择开关1 (速度1) |
| 3 Select_2 | Bool | 0 | Non-Retain | 选择开关2 (速度2) |
| 4 Select_3 | Bool | 0 | Non-Retain | 选择开关3 (速度3) |
| 5 Speed_1 | Word | 0 | Non-Retain | 速度1 (整数值) |
| 6 Speed_2 | Word | 0 | Non-Retain | 速度2 (整数值) |
| 7 Speed_3 | Word | 0 | Non-Retain | 速度3 (整数值) |
| 8 ▼ Output | | | | |
| 9 Ana_out | Word | 0 | Non-Retain | 模拟量输出整数值 |

通过设计用于通用控制任务的 FB，可对多个设备重复使用 FB，方法是：为 FB 的不同调用选择不同的背景 DB。FB 将输入 (IN)、输出 (OUT) 和输入/输出 (IN/OUT) 参数存储在背景 DB 中。

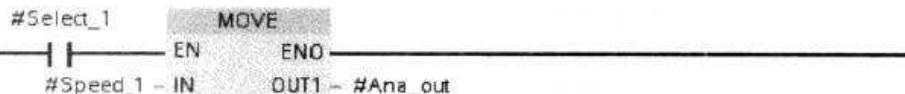
3) 如图 2-50 所示为主程序 OB1 梯形图。

▼ 块标题：

远程功能块

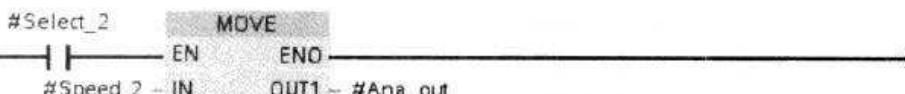
▼ 程序段 1：

选择速度1



▼ 程序段 2：

选择速度2



▼ 程序段 3：

选择速度3

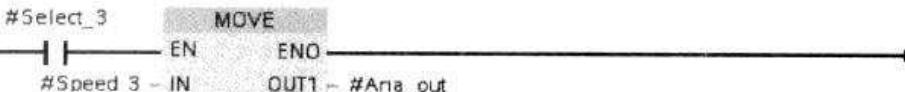


图 2-50 主程序 OB1 梯形图



4. 变量分配

变量表格见表 2-19，具体包括本地/远程选择、本地速度 1、本地速度 2、本地速度 3、启动与停止按钮、远程速度信号（模拟量）、变频器输入信号（模拟量）。

表 2-19 变量表格

| | 名称 | 数据类型 | 地址 |
|---|---------|------|-------|
| 1 | 本地/远程选择 | Bool | %I0.0 |
| 2 | 本地速度1 | Bool | %I0.1 |
| 3 | 本地速度2 | Bool | %I0.2 |
| 4 | 本地速度3 | Bool | %I0.3 |
| 5 | 停止按钮 | Bool | %I0.4 |
| 6 | 启动按钮 | Bool | %I0.5 |
| 7 | 远程速度信号 | Word | %IW96 |
| 8 | 变频器启动 | Bool | %Q0.0 |
| 9 | 变频器速度输入 | Word | %QW96 |

5. PLC 程序编程

主程序的编程，需要调用 FB 和 FC，具体如图 2-51 所示。

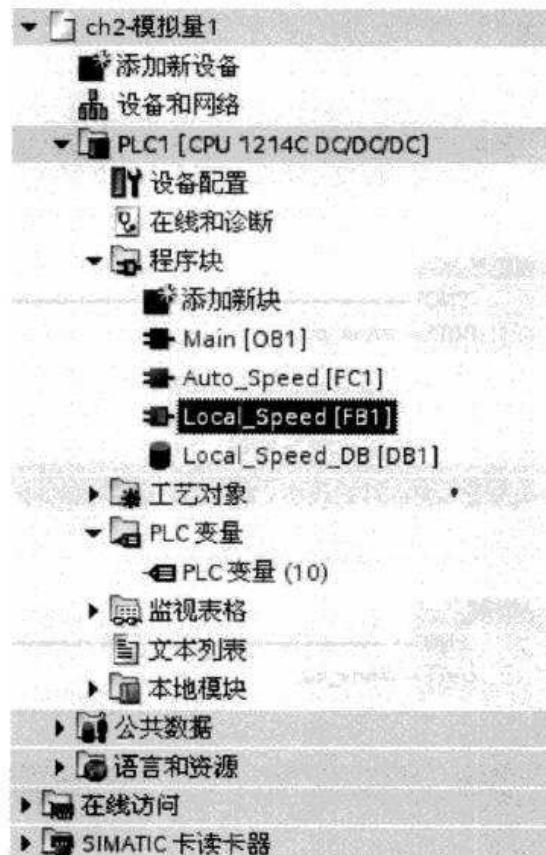


图 2-51 选择调用块

将 FB 或 FC 直接拉入到主程序中即可，但是由于 FB 需要调用 DB，因此会出现如图 2-52 所示的“调用选项”。

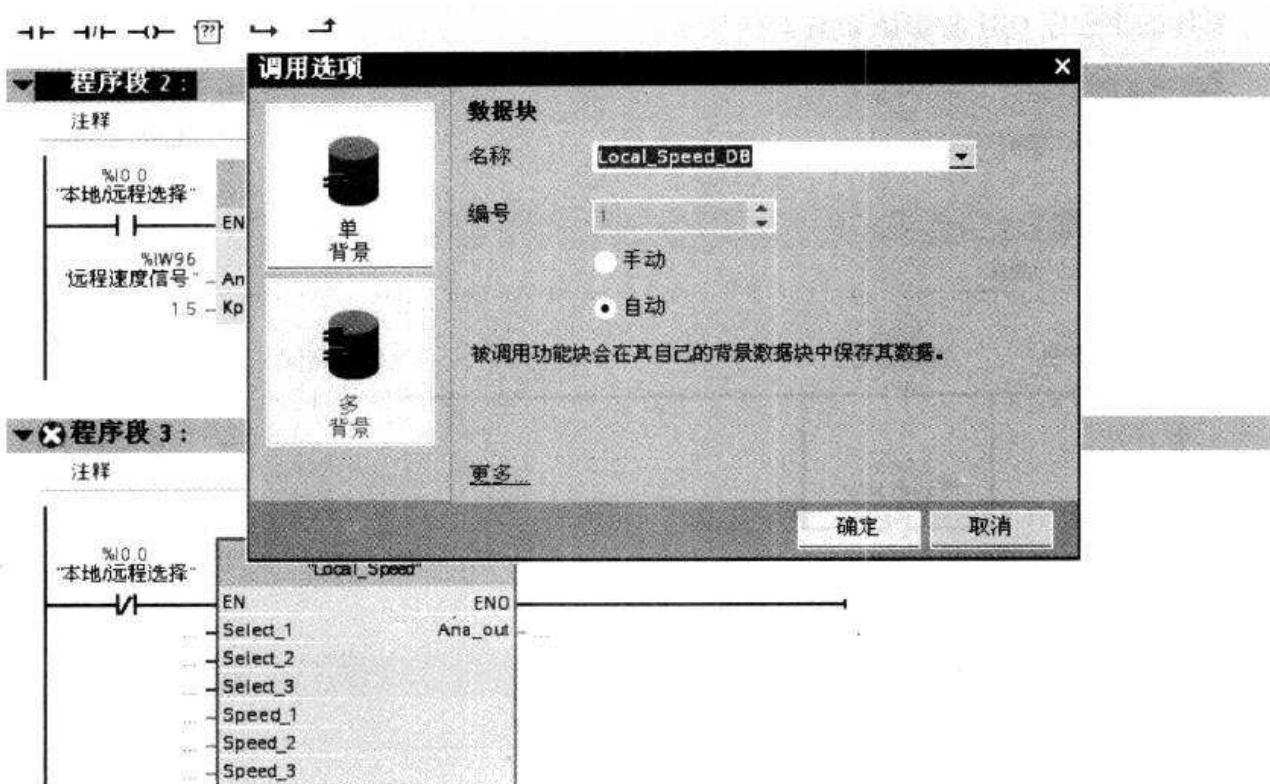


图 2-52 调用选项

图 2-53 所示为三次调用同一个 FB 的 OB，方法是针对每次调用使用一个不同的 DB。该结构使一个通用 FB 可以控制多个相似的设备（如电动机、变频器等），方法是在每次调用时为各设备分配不同的背景 DB。每个背景 DB 存储单个设备的数据（如速度、加速时间和总运行时间）。在此实例中，FB 22 控制 3 个独立的设备，其中 DB 201 用于存储第一个设备的运行数据，DB 202 用于存储第二个设备的运行数据，DB 203 用于存储第三个设备的运行数据。

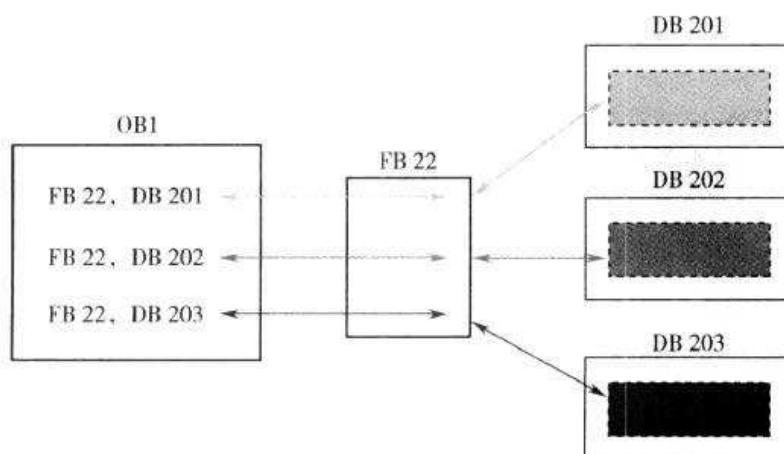


图 2-53 三次调用同一个 FB 的 OB



6. 主程序

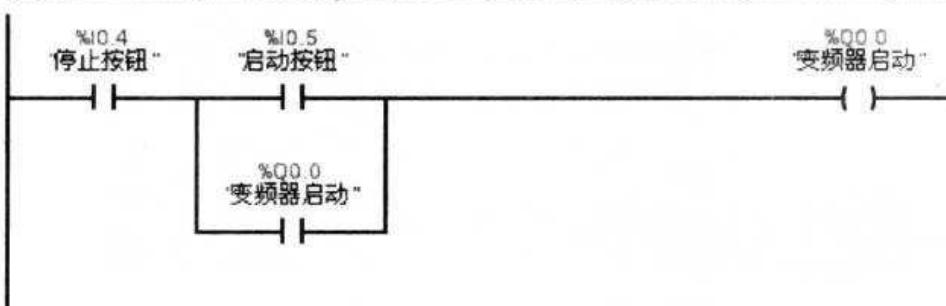
具体的主程序 OB1 梯形图如图 2-54 所示。

▼ 块标题：

模拟量扩展模块的应用

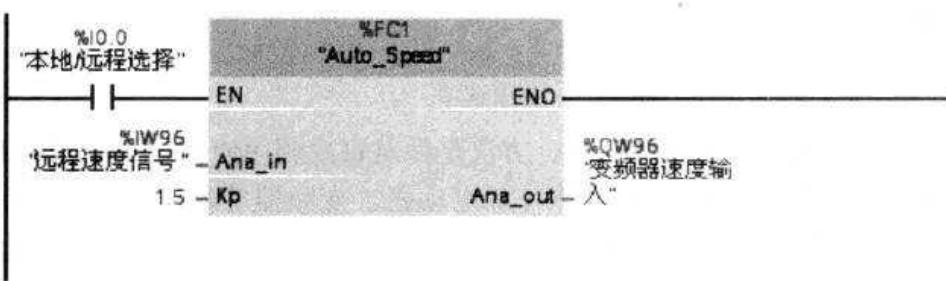
▼ 程序段 1：

正转启动与停止



▼ 程序段 2：

调用远程功能



▼ 程序段 3：

调用本地功能块

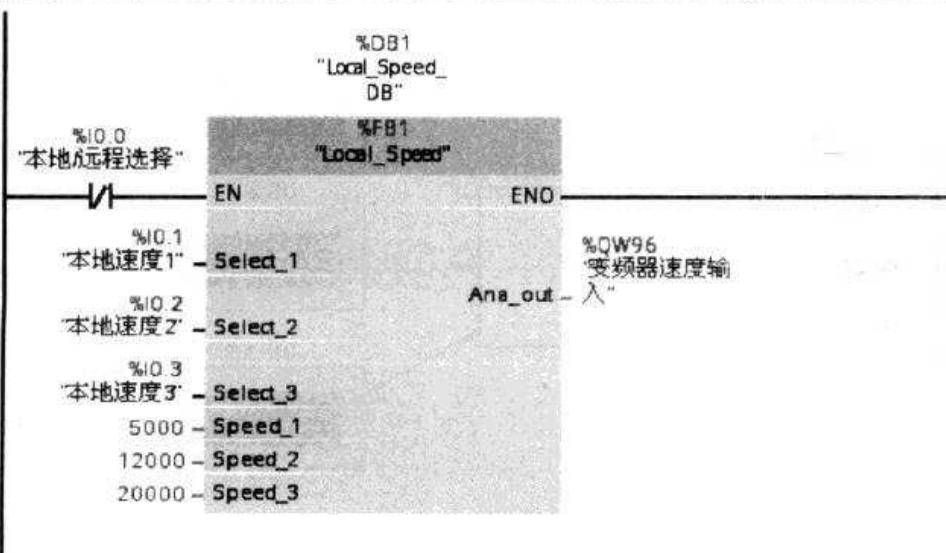


图 2-54 主程序 OB1 梯形图



2.5 工程案例：造纸生产线的速度自动控制

2.5.1 案例介绍

1. 造纸生产线概述

图 2-55 所示为造纸生产线现场，其自动化控制突出的表现在纸机的传动控制方面。因为纸种变化，纸机的速比变化较大，浆料配比、不同纸张特性等，使纸机车速需要有较大范围的调整；同时电网电压、频率和负荷的波动，也会引起纸机车速的变化，烘干部各段不同的加热温度，都会使纸页产生纵向的收缩。因此，要求纸机传动控制系统是一个稳定、精确、同步、协调的变速传动系统。由于纸机传动控制目前几乎毫无例外的采用多点分部传动的形式，客观上对自动控制系统提出了快速、精确、稳定和可靠等较高的要求；又因纸机传动在生产中的特殊重要地位，纸机电传动控制的一般配置由 PLC 控制单元、变频器单元等组成。

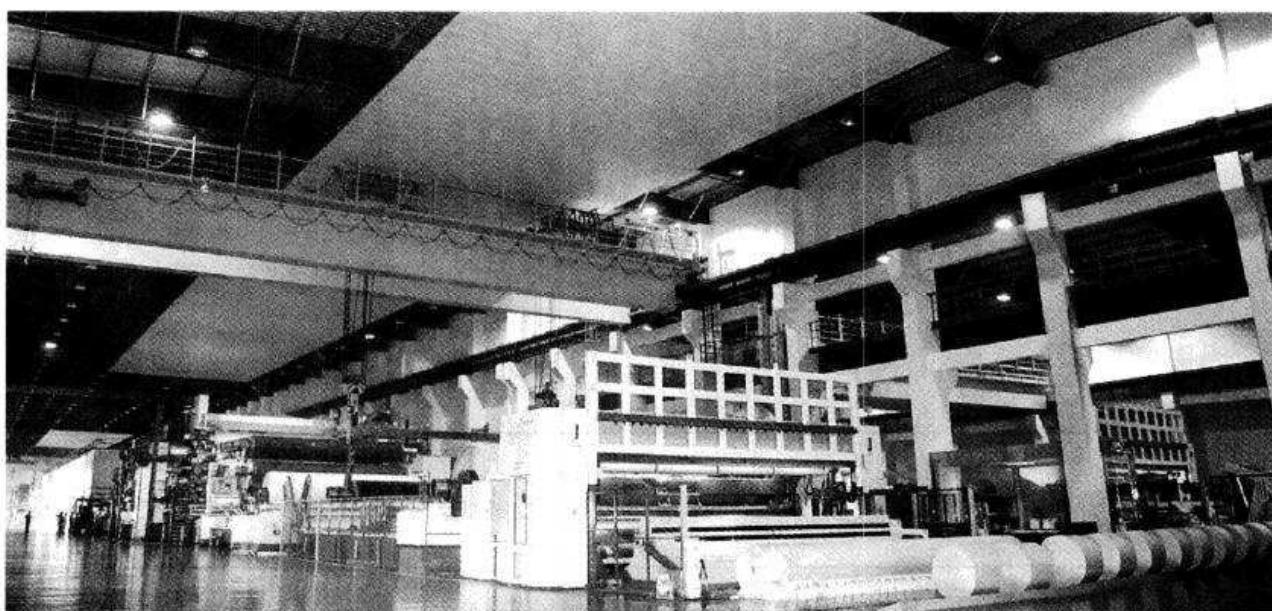


图 2-55 造纸生产线现场

由于造纸生产线的产品细薄、脆弱，为了防止纸张出现断裂、卷曲、褶皱、压痕，必须对各传动部分进行高精度的速度控制，以达到高质量的延展特性。造纸机各个部分之间必须采用拉力控制，以使造纸的生产线对应于脱水及干燥度等工艺参数所决定的伸缩性，从而保证纸张按成纸方向所限定的伸展率进行延展。也就是说，从上浆到上卷的整个过程，要保持纸幅一定的速度级联（这样才有拉力）。比如网部传动设定速度保持在 100m/min ，那么压榨部的速度保持在 104m/min 、前干燥为 108m/min 、后压榨为 110m/min 、后干燥为 112m/min 、压光机为 115m/min 、卷纸机为 116m/min 。为描述上的方便和控制的精度，这里引入速差分步控制的概念。

如传动 M_1 的速度为 v_1 ，传动 M_2 的速度为 v_2 ， M_2 的速差为 D_2 ，则三者的关系为



$$v_2 = v_1 \times (1 + D_2 \%)$$

依次类推，每个传动对应于前面一个传动都有一个速差值。因此，对于各传动点的设定只要求第一传动点的速度值（这里称为基速）和从第二传动点以后各传动点的速差值，就可由此计算出每一个传动的速度设定值，从而形成速度链。

2. 控制要求

某生产瓦楞原纸、挂面牛皮纸、箱板纸等的造纸生产线传动部的 PLC 控制示意如图 2-56 所示，其中 PLC 负责变频器的速度设定和速度反馈部分。传动部采用交流变频分部传动形式，分别由真空回头辊、压榨主传动、烘干传动、光泽缸、压光机及卷纸机等 6 个传动点组成，共计 363kW，每一台交流变频专用电动机通过硬齿面齿轮减速箱直接传动各主动辊或主动齿轮。该生产线设计车速为 150m/min。

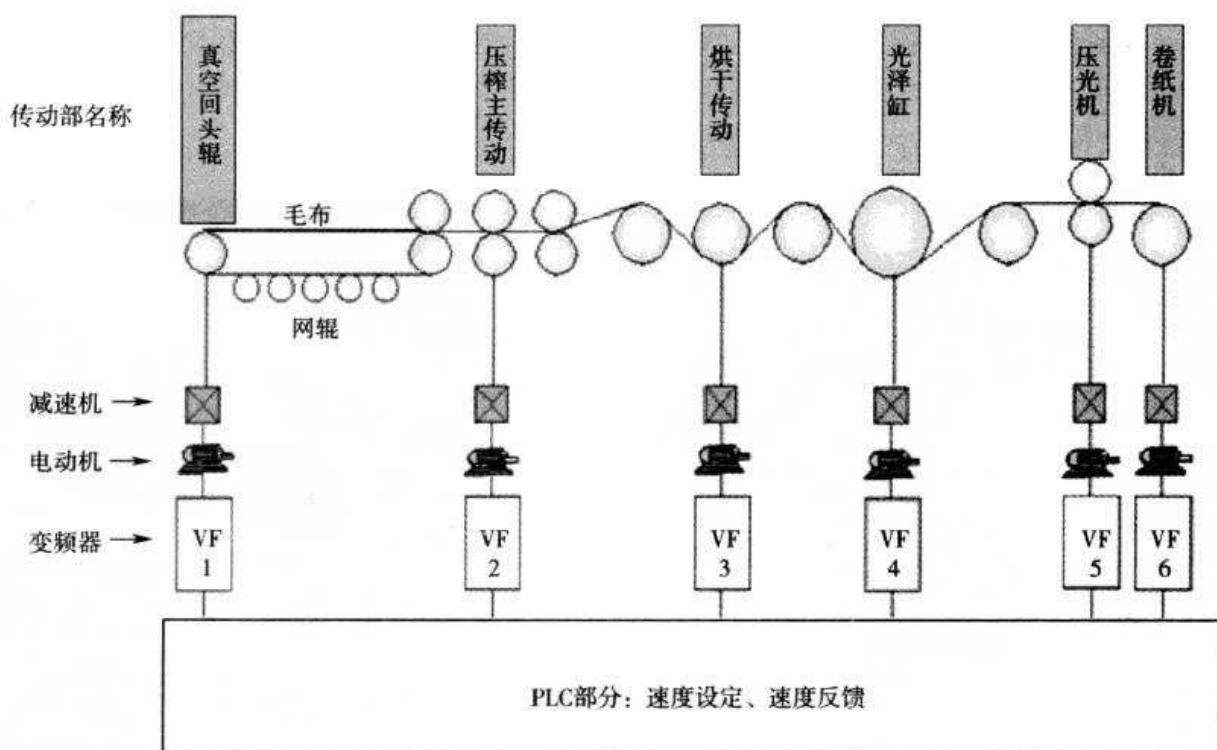


图 2-56 某造纸生产线传动部的 PLC 控制示意

控制要求如下：

- 1) 以 VF2 为总车速的设定点，其他各个传动点进行同步跟随，其主要提升、下降要平稳。
- 2) 每一个传动根据预先设置的速差，速差控制范围在 $-2\% \sim 2\%$ 以内。
- 3) VF2 的总车速设置 6 个主要的速度段，通过万能开关进行选择，分别为 25m/min、50m/min、75m/min、100m/min、125m/min、150m/min。
- 4) 每个传动都设置了加速和减速按钮，其中 VF2 主传动为 $\pm 25\text{m}/\text{min}$ 之内，每次按钮的动作按 $1\text{m}/\text{min}$ 单位变化，VF1、VF3 等其余 5 个传动为改变速差，变化单位为 0.01% 单位变化，不能超出上下限。



2.5.2 硬件配置、变量定义与软件编程

1. 硬件配置

根据控制要求，可以进行如图 2-57 所示的硬件配置，它共包括 CPU1214C DC/DC/DC、SM1234 4 * AI/2 * AQ、SM1222 8DI/8DO 24VDC、SM1232 2 * AQ 两块。

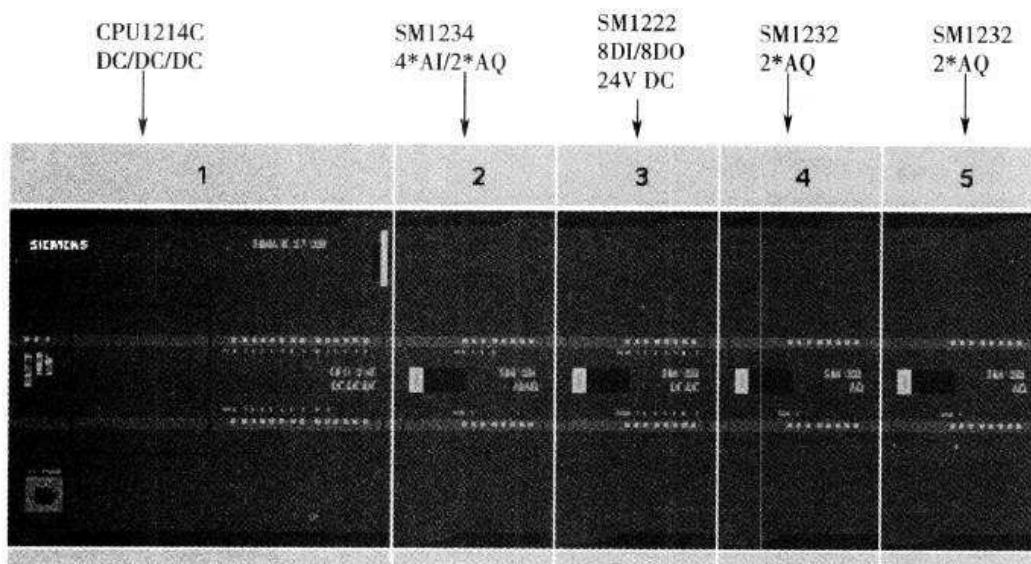


图 2-57 硬件配置

2. 变量分配

根据要求可以定义变量，见表 2-20。具体说明如下：

- 1) 模块 1 为 CPU，包括主速度设定开关、VF2 和 VF6 调整速度信号、VF5 和 VF6 的频率反馈输入。
- 2) 模块 2 为模拟量混合模块，包括 4 个模拟量输入为 VF1 和 VF4 的频率反馈输入，以及 VF1 和 VF2 的模拟量输出。
- 3) 模块 3 为数字量混合模块，包括 VF1、VF3、VF4、VF5 的速差信号。
- 4) 模块 4 和模块 5 为模拟量输出，为 VF3、VF4、VF5、VF6 的模拟量输出。

表 2-20 变量定义

| | 名称 | 数据类型 | 地址 |
|----|------------|------|--------|
| 1 | 输入地址 I80 | Byte | %I80 |
| 2 | 主速度设定开关1 | Bool | %I0.4 |
| 3 | 主速度设定开关2 | Bool | %I0.5 |
| 4 | 主速度设定开关3 | Bool | %I0.6 |
| 5 | VF2调整速度加信号 | Bool | %I0.7 |
| 6 | VF2调整速度减信号 | Bool | %I1.0 |
| 7 | VF6速差加信号 | Bool | %I1.1 |
| 8 | VF6速差减信号 | Bool | %I1.2 |
| 9 | VF1速差加信号 | Bool | %I12.0 |
| 10 | VF1速差减信号 | Bool | %I12.1 |
| 11 | VF3速差加信号 | Bool | %I12.2 |
| 12 | VF3速差减信号 | Bool | %I12.3 |
| 13 | VF4速差加信号 | Bool | %I12.4 |



(续)

| | 名称 | 数据类型 | 地址 |
|----|--------------|------|---------|
| 14 | VF4速差减信号 | Bool | %I12.5 |
| 15 | VF5速差加信号 | Bool | %I12.6 |
| 16 | VF5速差减信号 | Bool | %I12.7 |
| 17 | VF5频率反馈输入 | Int | %IW64 |
| 18 | VF6频率反馈输入 | Int | %IW66 |
| 19 | VF1频率反馈输入 | Int | %IW96 |
| 20 | VF2频率反馈输入 | Int | %IW98 |
| 21 | VF3频率反馈输入 | Int | %IW100 |
| 22 | VF4频率反馈输入 | Int | %IW102 |
| 23 | VF1模拟量输出 | Int | %QW96 |
| 24 | VF2模拟量输出 | Int | %QW98 |
| 25 | VF3模拟量输出 | Int | %QW128 |
| 26 | VF4模拟量输出 | Int | %QW130 |
| 27 | VF5模拟量输出 | Int | %QW144 |
| 28 | VF6模拟量输出 | Int | %QW146 |
| 29 | VF1频率显示 | Real | %MD0 |
| 30 | VF2频率显示 | Real | %MD4 |
| 31 | VF3频率显示 | Real | %MD8 |
| 32 | VF4频率显示 | Real | %MD12 |
| 33 | VF5频率显示 | Real | %MD16 |
| 34 | VF6频率显示 | Real | %MD20 |
| 35 | 主速度档位字 | Int | %MW30 |
| 36 | 主速度档位中间变量 | Byte | %MB30 |
| 37 | 主速度档位字节 | Byte | %MB31 |
| 38 | 主速度档位实数(0-1) | Real | %MD32 |
| 39 | 主速度档位设定值 | Real | %MD36 |
| 40 | 主速度调整值 | Real | %MD40 |
| 41 | 主速度设定值 | Real | %MD44 |
| 42 | 主速度输出值 | Real | %MD48 |
| 43 | 主速度设定中间值 | Real | %MD52 |
| 44 | 速度设定一上升沿 | Bool | %M100.4 |
| 45 | 速度设定二上升沿 | Bool | %M100.5 |
| 46 | 速度设定三上升沿 | Bool | %M100.6 |
| 47 | VF2速度调整加上升沿 | Bool | %M100.7 |
| 48 | VF2速度调整减上升沿 | Bool | %M101.0 |
| 49 | VF6速差调整加上升沿 | Bool | %M101.1 |
| 50 | VF6速差调整减上升沿 | Bool | %M101.2 |
| 51 | VF1速差调整加上升沿 | Bool | %M112.0 |
| 52 | VF1速差调整减上升沿 | Bool | %M112.1 |
| 53 | VF3速差调整加上升沿 | Bool | %M112.2 |
| 54 | VF3速差调整减上升沿 | Bool | %M112.3 |
| 55 | VF4速差调整加上升沿 | Bool | %M112.4 |
| 56 | VF4速差调整减上升沿 | Bool | %M112.5 |
| 57 | VF5速差调整加上升沿 | Bool | %M112.6 |
| 58 | VF5速差调整减上升沿 | Bool | %M112.7 |



3. 梯形图编程

本案例的程序主要分为两部分，一为上电初始化程序，一为主程序。在 S7-1200 PLC 中，上电初始化块的添加如图 2-58 所示。图 2-59 所示为 OB100 梯形图，主要设置主传动的档位值为 0、主传动的调整值为 0、老的主速度输出值为 0。

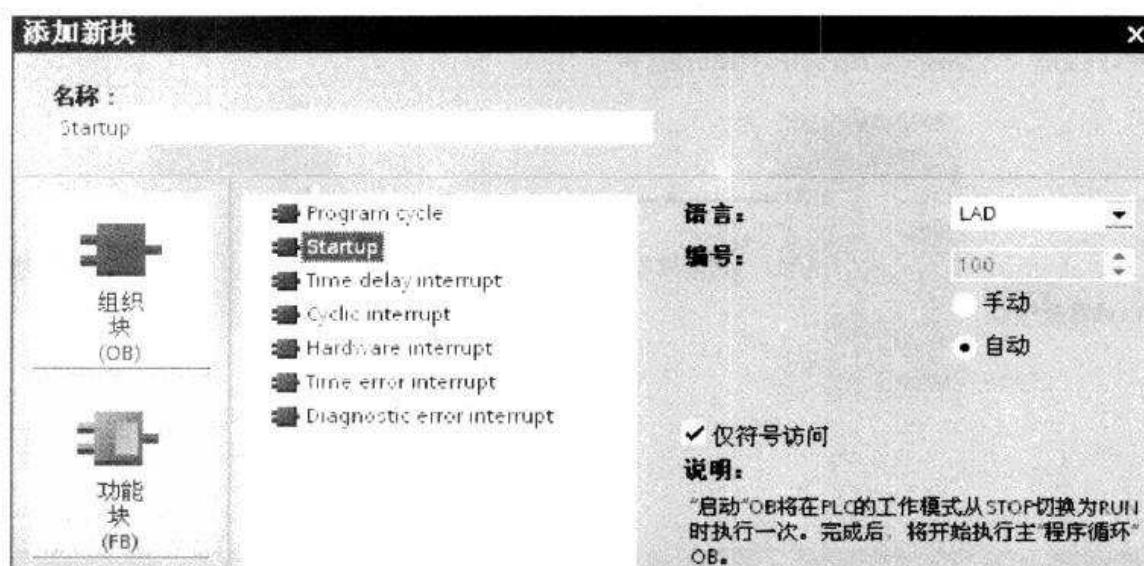


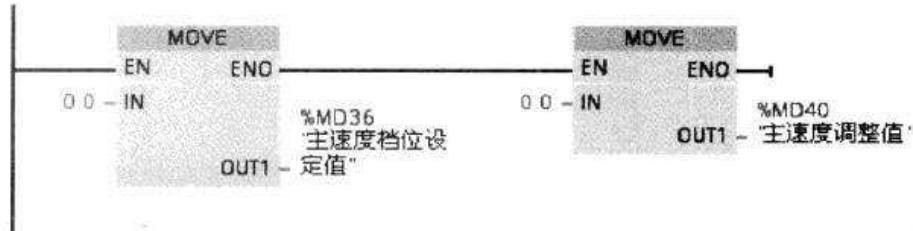
图 2-58 添加上电初始化块 OB100

块标题：

上电初始化

程序段 1：

设主传动的档位值为 0，主传动的调整值为 0



程序段 2：

设老的主速度输出值为 0



图 2-59 OB100 梯形图



图解西门子 S7-1200 PLC 入门到实践

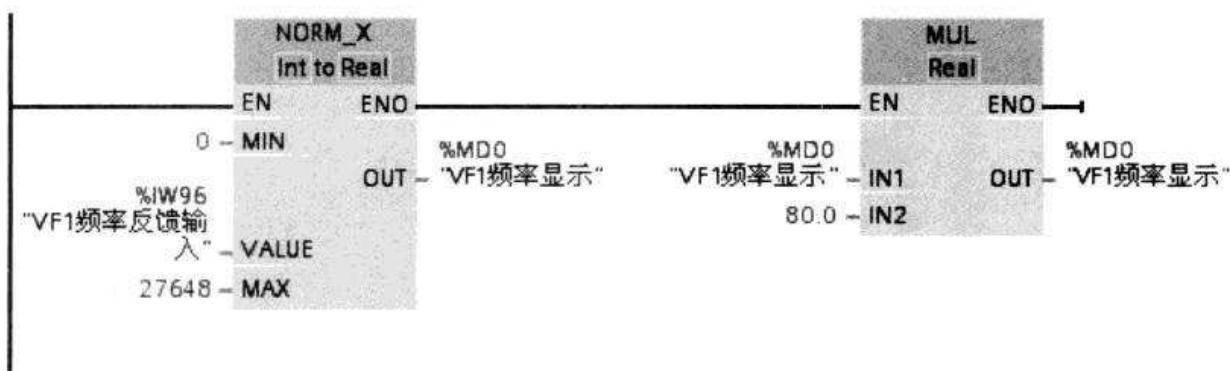
图 2-60 为主程序 OB1 梯形图。

块标题： "Main Program Sweep (Cycle)"

主扫描程序

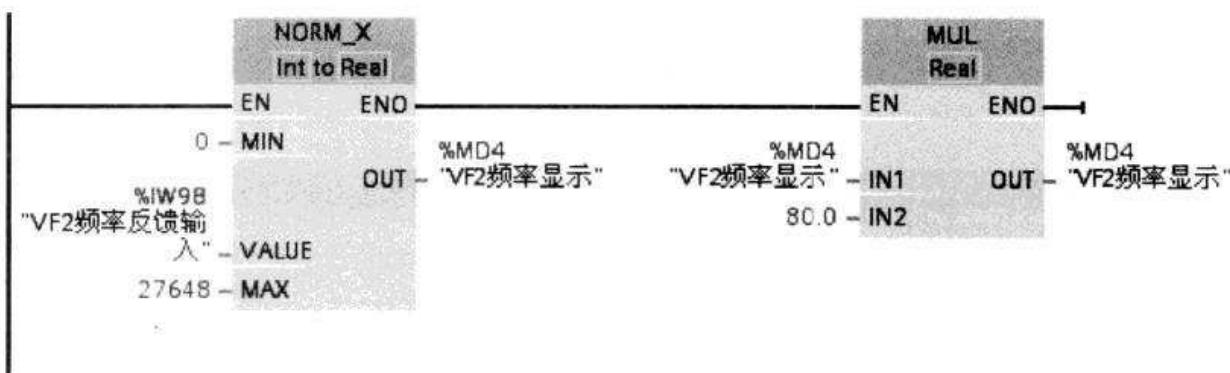
程序段 1：

将VF1的速度反馈信号0-10V转化为0-80Hz显示



程序段 2：

将VF2的速度反馈信号0-10V转化为0-80Hz显示



程序段 3：

将VF3的速度反馈信号0-10V转化为0-80Hz显示

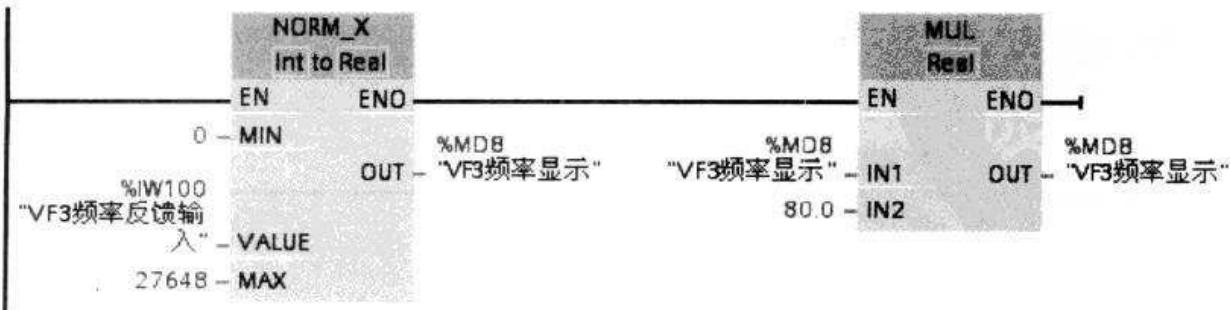
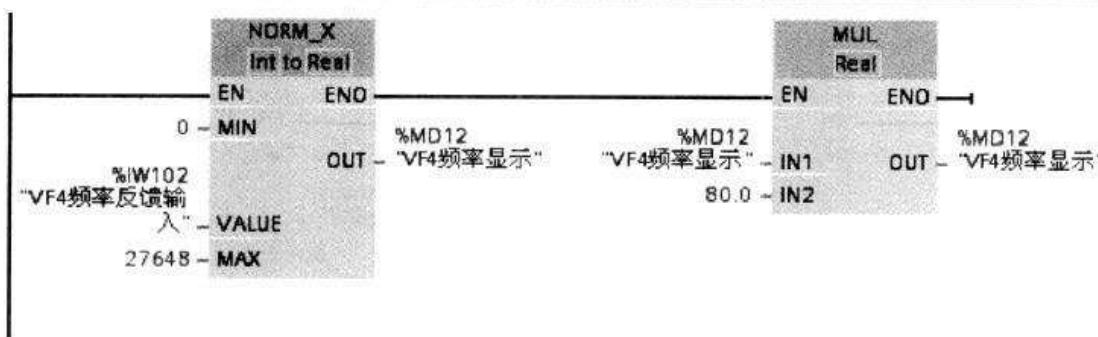


图 2-60 OB1 梯形图

第2章 S7-1200 PLC 扩展模块的应用

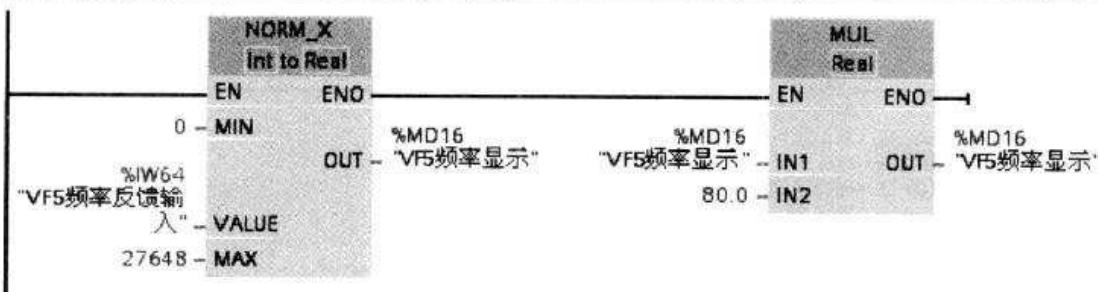
▼ 程序段 4：

将VF4的速度反馈信号0-10V转化为0-80Hz显示



▼ 程序段 5：

将VF5的速度反馈信号0-10V转化为0-80Hz显示



▼ 程序段 6：

将VF6的速度反馈信号0-10V转化为0-80Hz显示



▼ 程序段 7：

主设定速度根据选择开关进行读取 (I0.4、I0.5和I0.6)

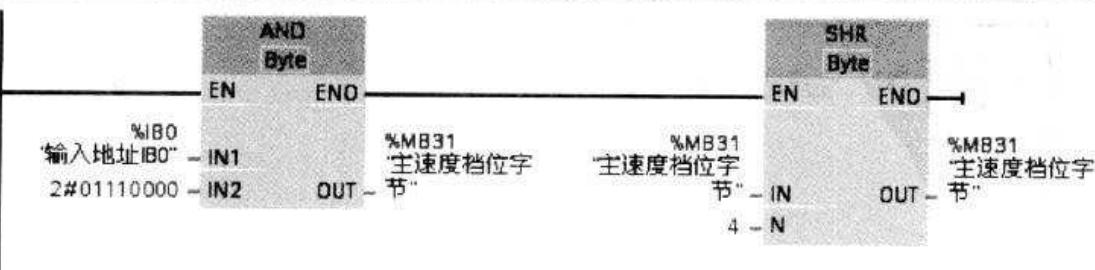


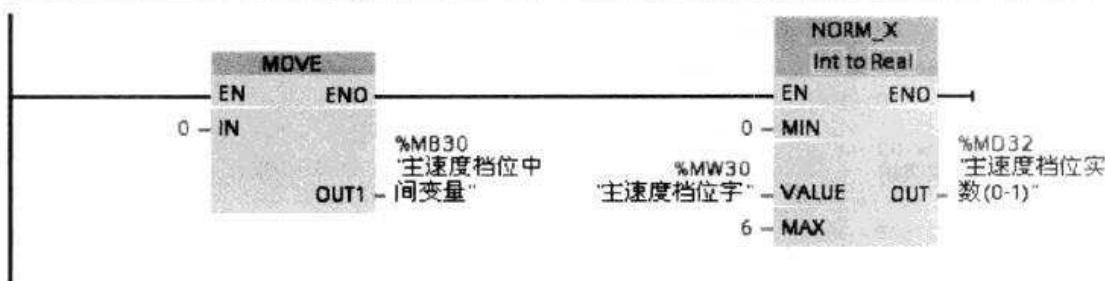
图 2-60 OB1 梯形图 (续一)



图解西门子 S7-1200 PLC 入门到实践

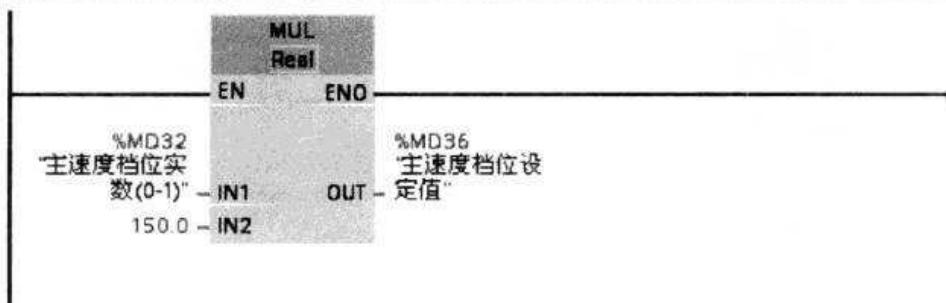
▼ 程序段 8：

将主设定速度的值直接转化为0-6档



▼ 程序段 9：

将主设定速度0-6直接转化为0-150m/min



▼ 程序段 10：

主传动速度设定值的调整

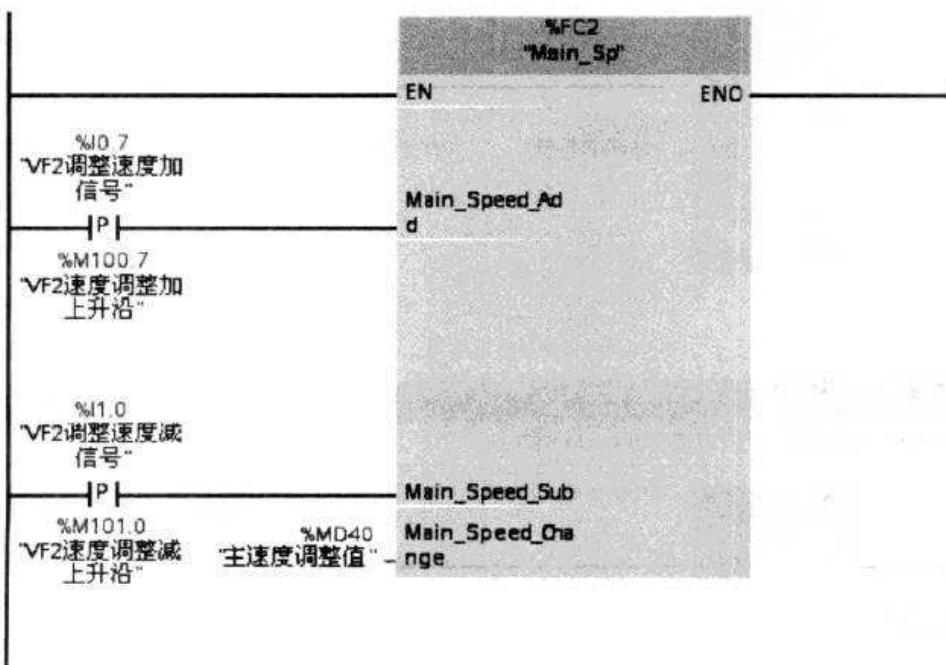
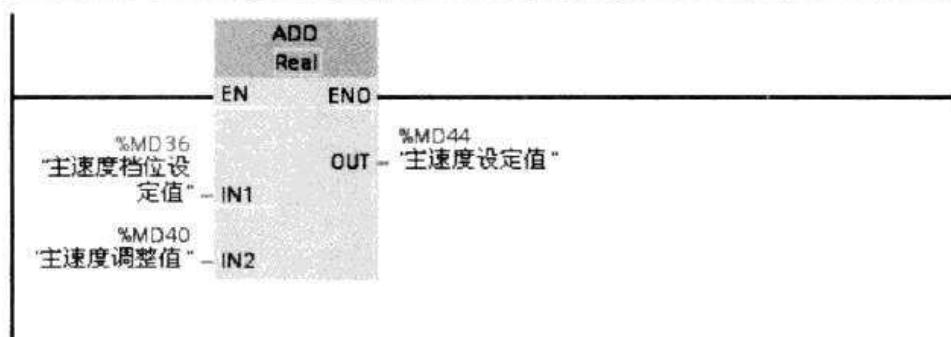


图 2-60 OBI 梯形图（续二）



▼ 程序段 11：

将主传动的设定档位值与调整值相加得到新的主传动设定值



▼ 程序段 12：

注释

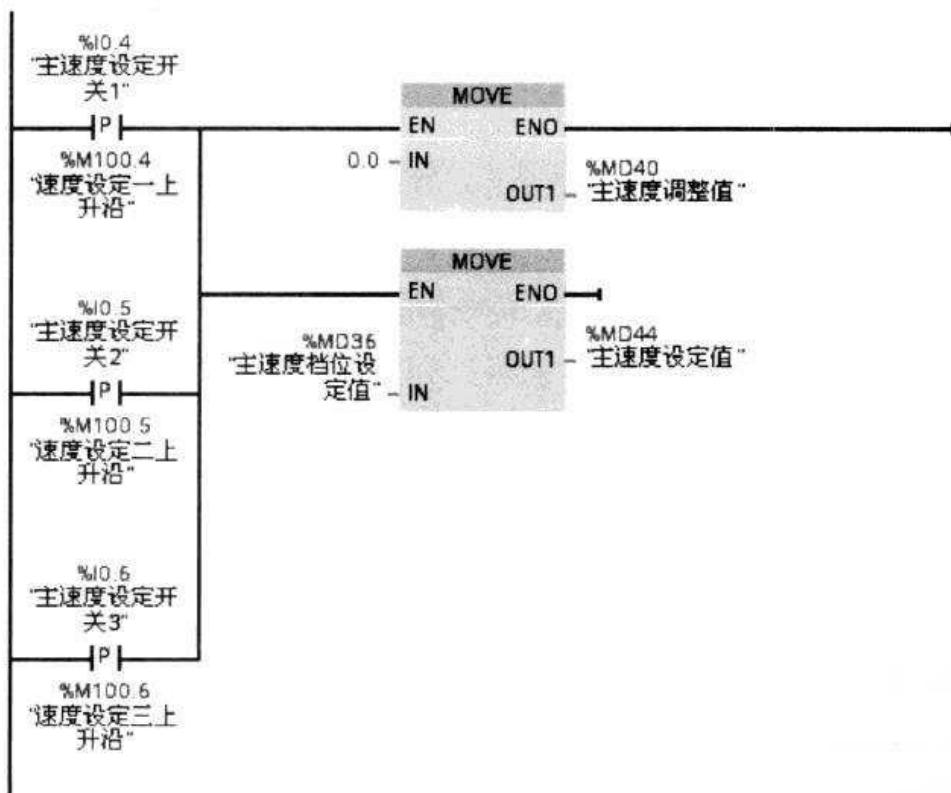
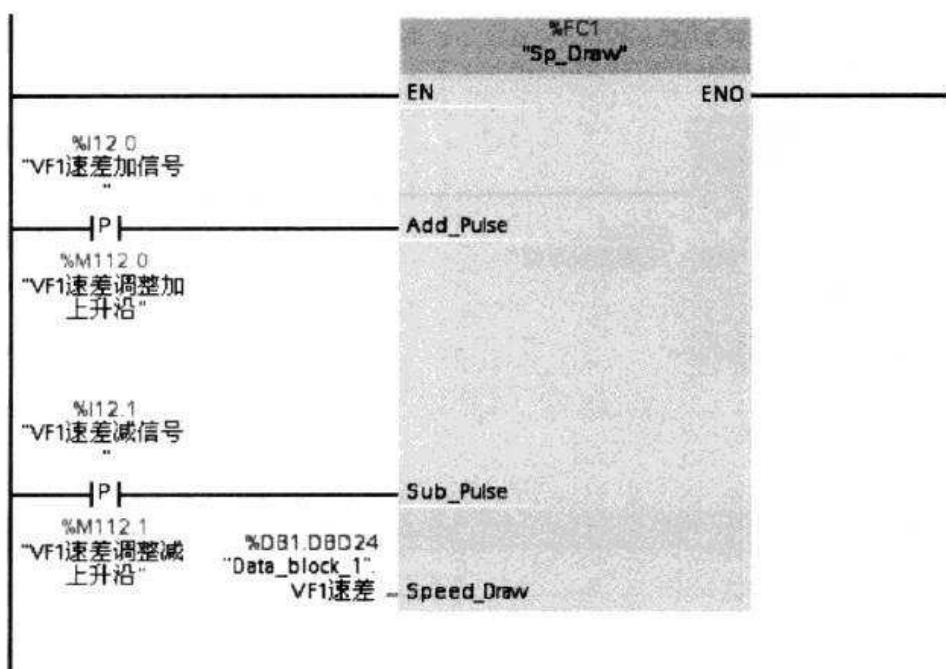


图 2-60 OBI 梯形图（续三）



▼ 程序段 13：

变频器VF1的速差控制



▼ 程序段 14：

变频器VF3的速差控制

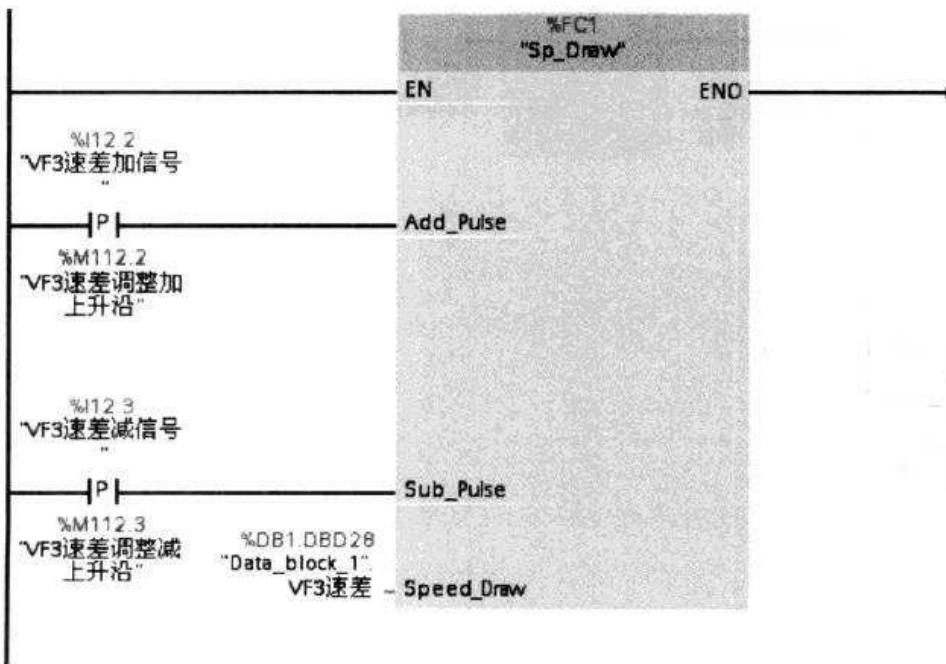
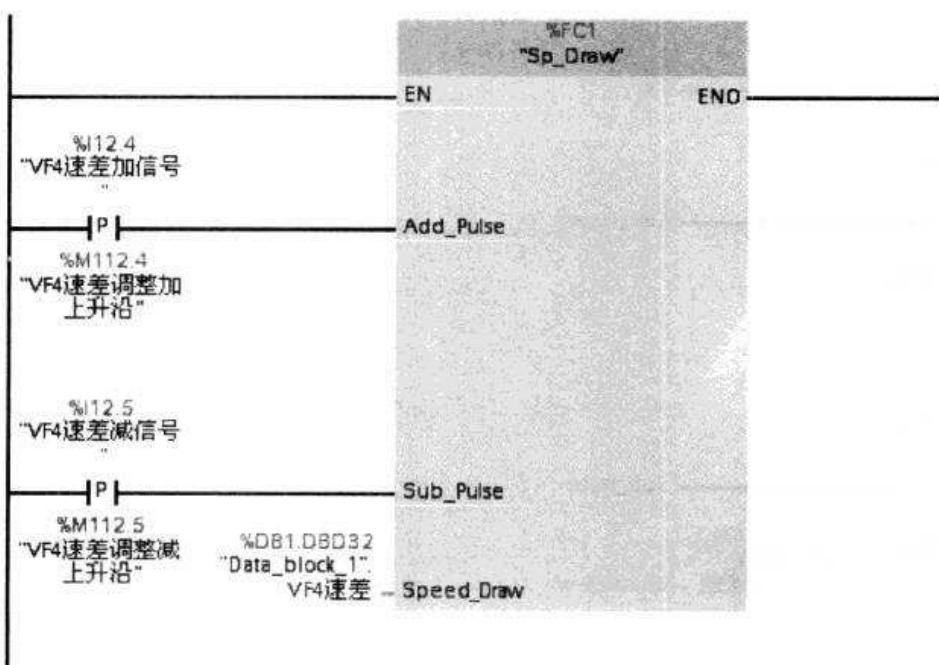


图 2-60 OBI 梯形图（续四）



▼ 程序段 15：

变频器VF4的速差控制



▼ 程序段 16：

变频器VF5的速差控制

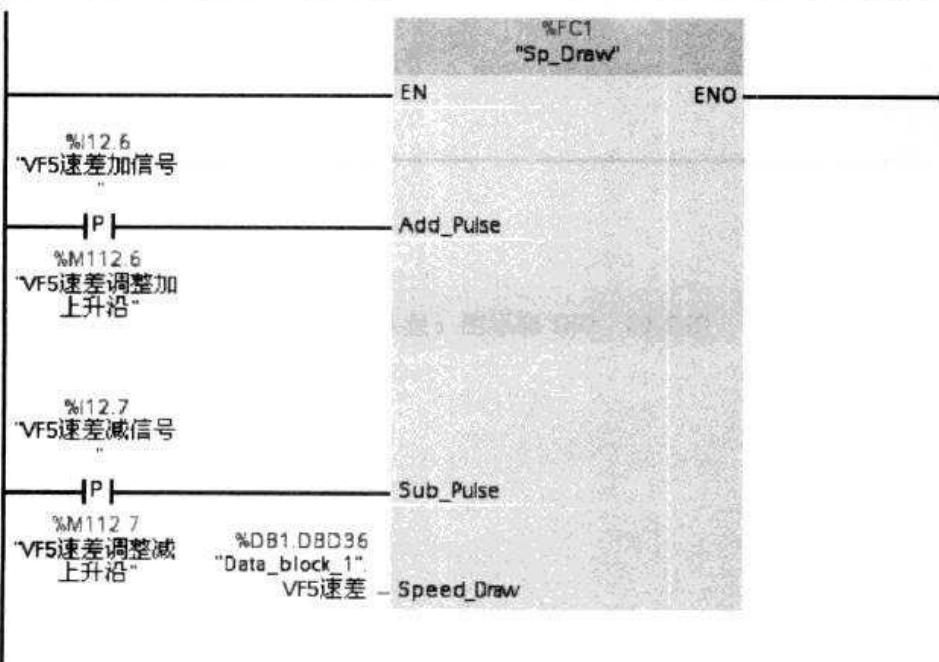
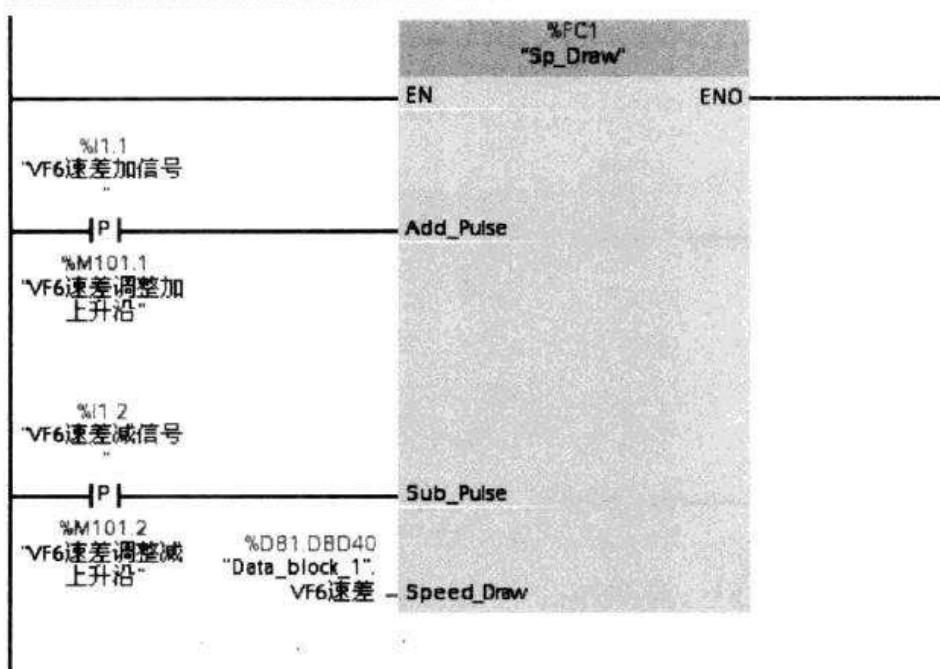


图 2-60 OBI 梯形图 (续五)



▼ 程序段 17：

变频器VF6的速差控制



▼ 程序段 18：

调用FB1块计算变频器VF1、VF3-VF6的实时设定速度

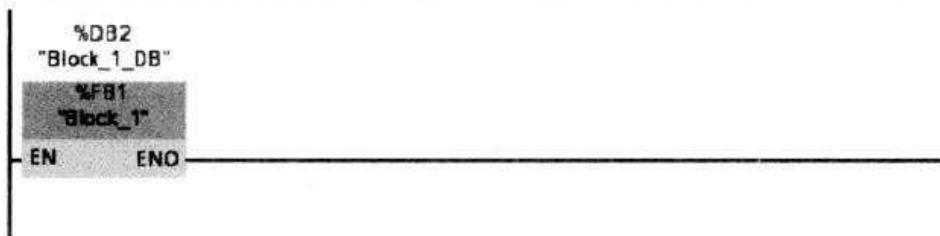
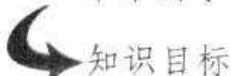


图 2-60 OBI 梯形图（续六）

第3章 KTP触摸屏编程

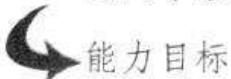
传统的工业控制系统一般使用按钮和指示灯来操作和监视系统，但很难实现系统工艺参数的现场设置和修改，也不方便对整个系统的集中监控。触摸屏的主要功能就是取代传统的控制面板和显示仪表，通过控制单元（如PLC）通信，实现人与控制系统的信息交换，更方便地实现对整个系统的操作和监视。触摸屏由于操作简便、界面美观、人机交互好等优点，将在控制领域得到广泛的应用。

本章的学习目标如下：

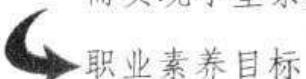


知识目标

了解触摸屏发展的历史过程、应用特点；熟悉触摸屏的设计原则；熟悉KTP西门子触摸屏的画面组态特点。



能对KTP触摸屏进行画面组态，并与西门子S7-1200 PLC进行通信连接，从而实现小型系统的可视化控制。



通过综合使用各种自动化产品从而形成全集成自动化思想（TIA）。





3.1 背景知识介绍：触摸屏的由来

3.1.1 工业触摸屏的出现

20世纪60年代末PLC的出现使工业控制向前迈进了一大步，随着PLC的应用和发展，工程师们渐渐发现：仅仅用开关、按钮和指示灯来控制PLC，并不能完全发挥PLC的潜在功能，为了实现更高层次的工业自动化，人们开始研发一种新的控制界面，即触摸屏。触摸屏集成了液晶显示屏、触摸面板、控制单元及数据存储单元，并且可以在显示屏上模拟开关、按钮、指示灯。它可以基本代替那些电气元件，从而使工业控制再次向前迈进了一步，操作工人得以面对更加友好的操作界面。

液晶显示工业触摸屏是人机交互系统中一颗耀眼的明星。由于高可靠、长寿命、高性能等特点使得触摸屏越来越受到自动化系统集成商、自动化设备制造商的青睐。

传统的工业控制系统一般使用按钮和指示灯来操作和监视系统，但很难实现系统工艺参数的现场设置和修改，也不方便对整个系统的集中监控。触摸屏的主要功能就是取代传统的控制面板和显示仪表（见图3-1），通过控制单元（如PLC）通信，实现人与控制系统的信息交换，更方便地实现对整个系统的操作和监视。触摸屏由于操作简便、界面美观、人机交互好等优点，将在控制领域得到广泛的应用。

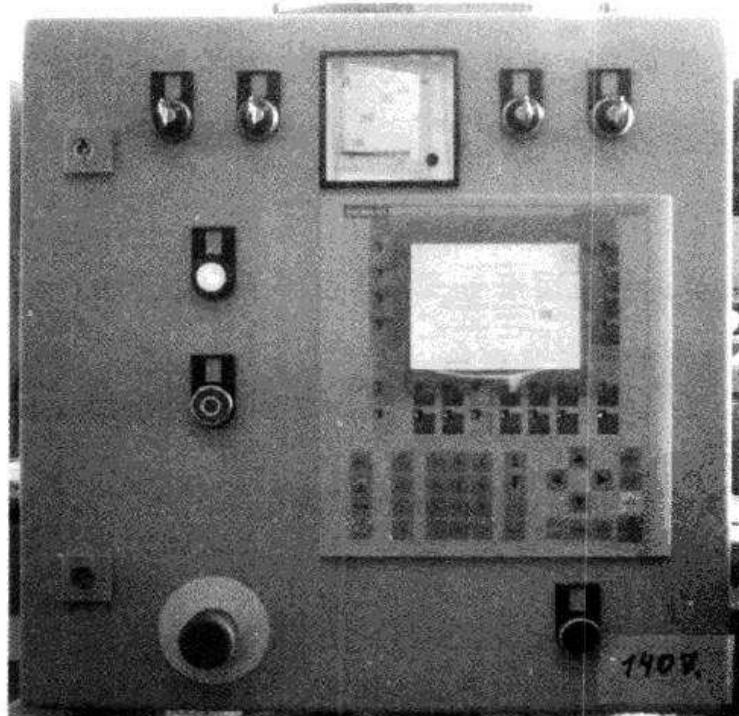


图3-1 触摸屏画面

3.1.2 触摸屏的种类

为了操作上的方便，人们用触摸屏来代替鼠标或键盘。工作时，必须首先用手指或其他



物体触摸安装在显示器前端的触摸屏，然后系统根据手指触摸的图标或菜单位置来定位选择信息输入。触摸屏由触摸检测部件和触摸屏控制器组成；触摸检测部件安装在显示器屏幕前面，用于检测用户触摸位置，接受后送触摸屏控制器；而触摸屏控制器的主要作用是从触摸点检测装置上接收触摸信息，并将它转换成触点坐标，再送给CPU，它同时能接收CPU发来的命令并加以执行。

按照触摸屏的工作原理和传输信息的介质，可以把触摸屏分为四种，它们分别为电阻式、红外线式、电容式以及表面声波式。

1. 电阻式触摸屏

电阻式触摸屏的屏体部分是一块与显示器表面非常配合的多层复合薄膜，由一层玻璃或有机玻璃作为基层，表面涂有一层透明的导电层（OTI，氧化铟），上面再盖有一层外表面硬化处理、光滑防刮的塑料层，它的内表面也涂有一层OTI，在两层导电层之间有许多细小的透明隔离点把它们隔开绝缘。当手指接触屏幕，两层OTI导电层出现一个接触点，因其中一面导电层接通Y轴方向的5V均匀电压场，使得侦测层的电压由零变为非零，控制器侦测到这个接通后，进行A/D转换，并将得到的电压值与5V相比，即可得触摸点的Y轴坐标，同理得出X轴的坐标，这就是电阻技术触摸屏的最基本原理。电阻屏根据引出线数的多少，分为四线、五线等多线电阻触摸屏。图3-2所示为电阻式触摸屏。

2. 电容式触摸屏

电容式触摸屏的构造主要是在玻璃屏幕上镀一层透明的薄膜体层，再在导体层外加上一块保护玻璃，双玻璃设计能彻底保护导体层及感应器。图3-3所示为电容式触摸屏。

电容式触摸屏在触摸屏四边均镀上狭长的电极，在导电体内形成一个低电压交流电场。用户触摸屏幕时，由于人体电场，手指与导体层间会形成一个耦合电容，四边电极发出的电流会流向触点，而电流强弱与手指到电极的距离成正比，位于触摸屏幕后的控制器会计算电流的比例及强弱，准确算出触摸点的位置。电容式触摸屏的双玻璃不但能保护导体及感应器，更能有效地防止外在环境因素对触摸屏造成的影响，就算屏幕沾有污秽、尘埃或油渍，电容式触摸屏依然能准确算出触摸位置。

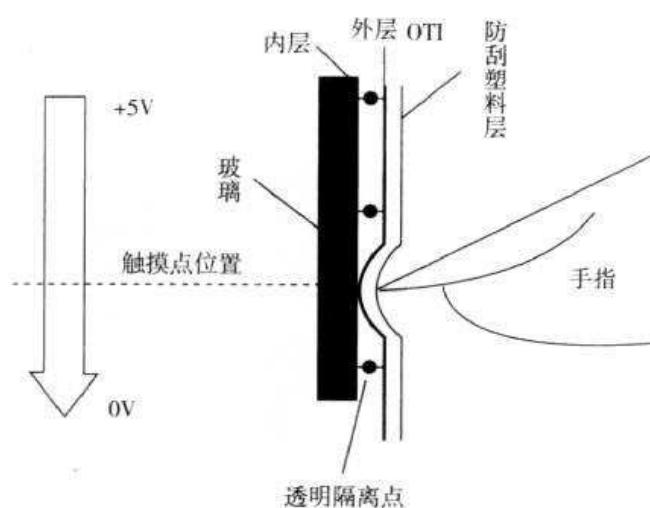


图 3-2 电阻式触摸屏

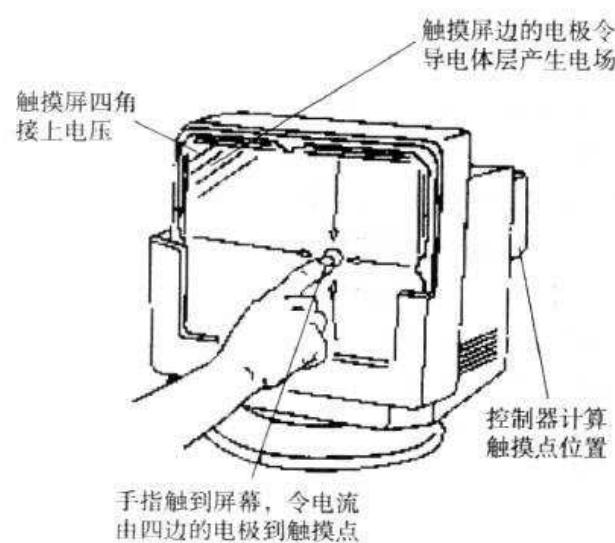


图 3-3 电容式触摸屏



3. 红外线式触摸屏

该触摸屏由装在触摸屏外框上的红外线发射与接收感测元件构成，在屏幕表面上，形成红外线探测网，任何触摸物体可改变触点上的红外线而实现触摸屏操作。红外触摸屏不受电流、电压和静电干扰，适宜某些恶劣的环境条件。其主要优点是价格低廉、安装方便、不需要卡或其他任何控制器，可以用在各档次的计算机上。此外，由于没有电容充放电过程，红外线式触摸屏的响应速度比电容式快，但分辨率较低。图 3-4 所示为红外线式触摸屏。

红外线式触摸屏的原理很简单，只是在显示器上加上光点距架框，无需在屏幕表面加上涂层或接驳控制器。光点距架框的四边排列了红外线发射管及接收管，在屏幕表面形成一个红外线网。用户以手指触摸屏幕某一点，便会挡住经过该位置的横竖两条红外线，计算机便可即时算出触摸点位置。因为红外线式触摸屏不受电流、电压和静电干扰，所以适宜某些恶劣的环境条件。不过，由于只是在普通屏幕增加了框架，在使用过程中架框四周的红外线发射管及接收管很容易损坏。

光点距离框四边排列了红外线发射管

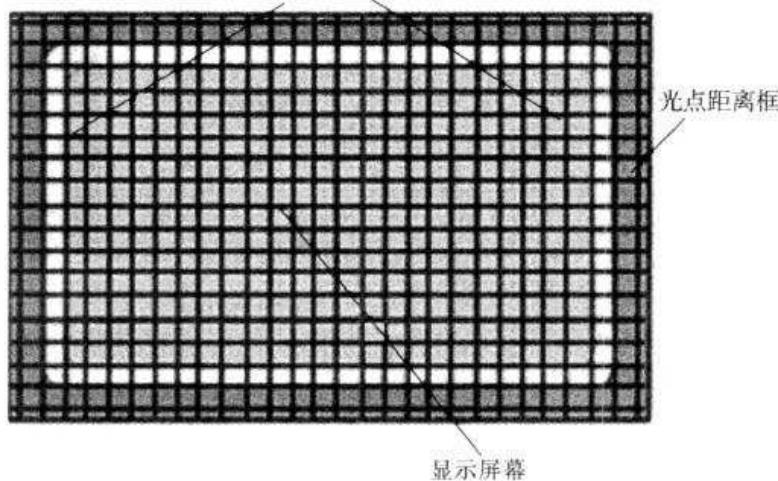


图 3-4 红外线式触摸屏

4. 表面声波触摸屏

表面声波是一种沿介质表面传播的机械波。该种触摸屏由触摸屏、声波发生器、反射器和声波接受器组成，其中声波发生器能发送一种高频声波跨越屏幕表面，当手指触及屏幕时，触点上的声波即被阻止，由此确定坐标位置。表面声波触摸屏不受温度、湿度等环境因素影响，分辨率极高，有极好的防刮性，寿命长（5000 万次无故障）；透光率高（92%），能保持清晰透亮的图像质量；没有漂移，只需安装时进行一次校正；有第三轴（即压力轴）响应，最适合公共场所使用。表面声波触摸屏的触摸屏部分可以是一块平面、球面或是柱面的玻璃平板，安装在 CRT、LED、LCD 或是等离子显示器屏幕的前面。这块玻璃平板只是一块纯粹的强化玻璃，区别于其他触摸屏技术是没有任何贴膜和覆盖层。玻璃屏的左上角和右下角各固定了竖直和水平方向的超声波发射换能器，右上角则固定了两个相应的超声波接收换能器。玻璃屏的 4 个周边则刻有 45° 角由疏到密间隔非常精密的反射条纹。图 3-5 所示为表面声波触摸屏。

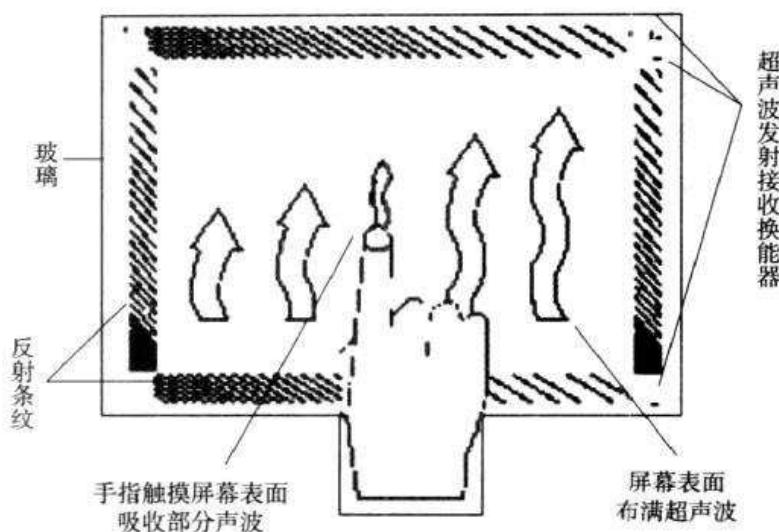


图 3-5 表面声波触摸屏

3.1.3 基于现场总线与以太网的触摸屏

在随后的几十年里，触摸屏产品发展迅速，它已经不仅仅是按钮和指示灯的替代品，更集成了操作控制、状态监控、数据存储、报表打印、网络通信、视频监控等众多高端功能，并在纺织、印刷、制药、化工、机械等诸多行业中得到广泛的应用。

但是人们不希望触摸屏仅仅应用在单机设备的控制上，在网络化的大潮中，触摸屏应该具有更广泛的应用和更强大的网络功能。现在有很多种工业局域网络：例如由西门子公司首先提出并且推广应用的 ProfiBus，它针对不同的网络构成又细分为 3 个层次：ProfiBus-DP、ProfiBus-PA、ProfiBus-FMS，它已成为欧洲的工业总线标准，在全世界范围内有广泛的应用；由三菱公司倡导的 CC-Link 总线，它的研发时间较晚，所以在技术上比较完善；欧洲的 Device-Net 总线，它支持的可连接设备非常多；还有 Modbus、Interbus 等不下十余种总线协议。如图 3-6 所示为触摸屏与现场总线连接方式。

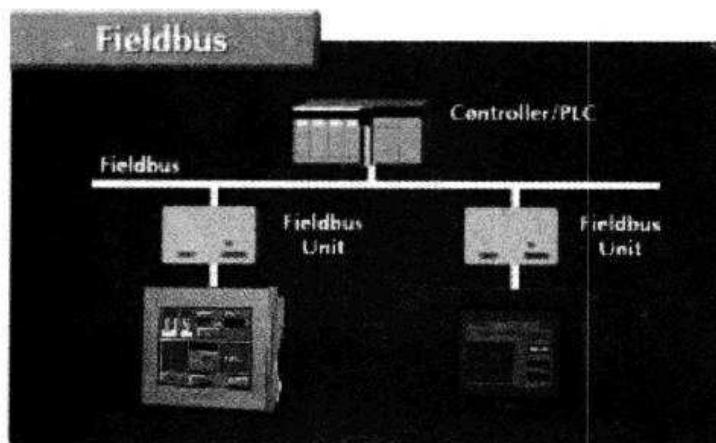


图 3-6 触摸屏与现场总线连接方式



虽然它们在工业网络的构成上有各自的优势和特点，但是它们却采用各自的通信协议，相互之间并不兼容，只能在一定的网络范围内使用，缺乏一致性和通用性。这使得触摸屏在接入这些网络实现控制集成时遇到了一定的障碍。触摸屏厂商就不得不专门针对不同的网络通信协议设计配备不同的通信模块，从而增加了整个设备的制造组网成本，缩小了应用范围，降低了通用性。在这方面工业计算机（IPC）实现起来相对简单一些，但是也要增加相应的软件成本，而且作为一台完整的计算机，在工业环境的适应上也存在一定的问题，相对于触摸屏、PLC 来说，其稳定性也比较弱。

目前，在触摸屏市场上已经出现了集成以太网接口的触摸屏，如西门子公司的 KTP 600 Basic Color PN（见图 3-7）。

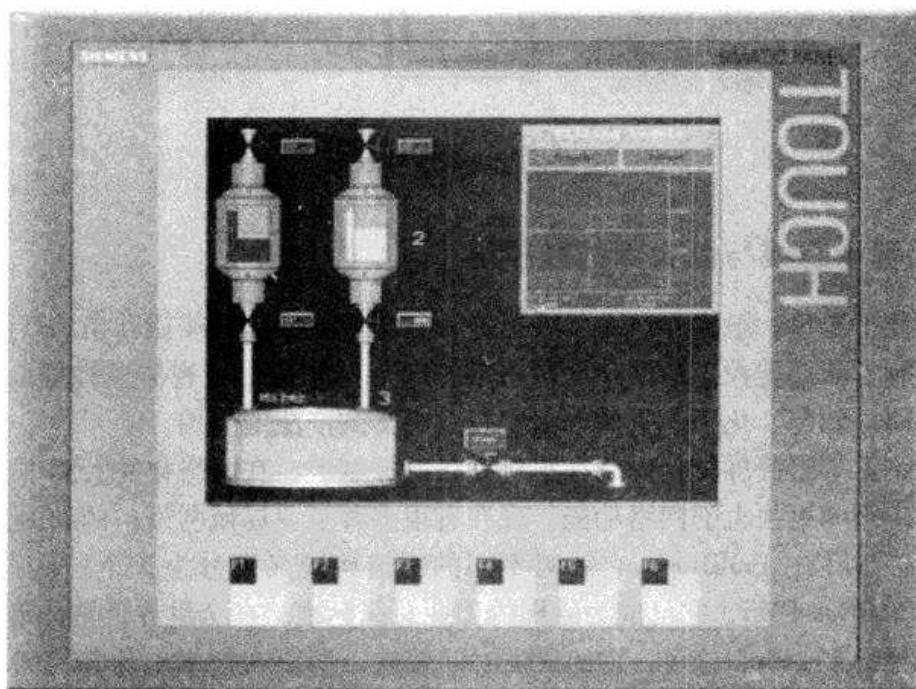


图 3-7 西门子公司的 KTP600 系列触摸屏

它所集成的是一块基于 TCP/IP 协议的 10M 标准以太网芯片。TCP/IP 协议即现在已广泛应用于国际互联网络（World Wide Web）的标准开放式通信协议。目前，个人计算机上网时基本都是基于该协议，但它在工业现场的应用还未普及。随着新一代触摸屏的诞生使其在工业现场的作用得以推广，并使整个设备接入网络成为可能。不仅仅是同一物理空间（局域网络）内的几台设备之间相互连接，更可连接到 Internet（广域网络），使不同物理空间内的设备之间实现网络通信、数据传输（见图 3-8）。

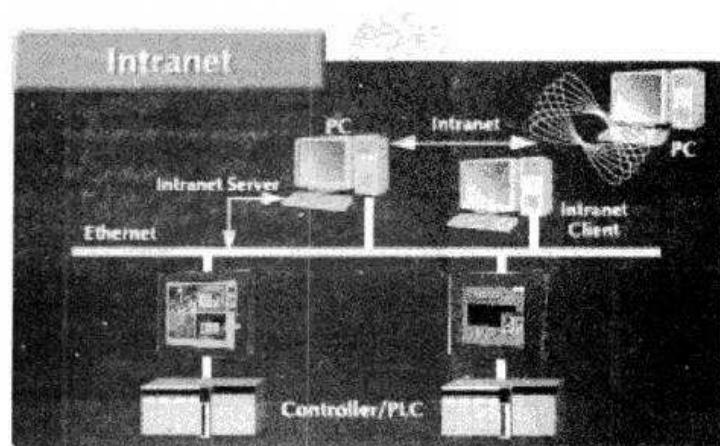


图 3-8 触摸屏与以太网

它的好处还在于：

1) TCP/IP 协议是开放的，只要有正确的 IP 地址就可以接入网络，不需要其他额外的授权或接入设备；这将大大降低设备组网的费用，扩大应用范围。

2) 以太网可以提供更快的通信速度，目前在计算机中成品化的网卡中已经出现了百兆网卡甚至千兆网卡。

3) 更可以提供无线网络的支持，摆脱物理线路的束缚。

目前，这种继承 10M/100M 网卡的触摸屏已经实现了单机通过以太网和上位监控计算机之间的通信，并可以进行状态监控、数据传输以及设备控制。

上述的只是触摸屏网络功能的最基本应用。我们有理由相信，随着技术的不断发展，触摸屏在开放网络中还会有更大的发展和更广泛的应用。

3.2 知识讲座：触摸屏与全集成自动化思想

3.2.1 触摸屏的基本功能和使用方法

1. 基本功能

触摸屏是用户和计算机之间实现互动的最简单、最直接的方式，在所有行业和应用中，触摸屏技术都带来了很大的实质益处，主要表现为以下几点：

- 1) 触摸屏使用户无需经过任何培训就能马上使用计算机；
- 2) 触摸屏几乎完全消除了操作员误操作的可能性，因为供用户选择的菜单设置非常明确；
- 3) 触摸屏结实耐用，可以承受键盘和鼠标易受损坏的恶劣环境。

图 3-9 所示为触摸屏在供水控制中的一个典型应用，由图可知，触摸屏画面可以进行：

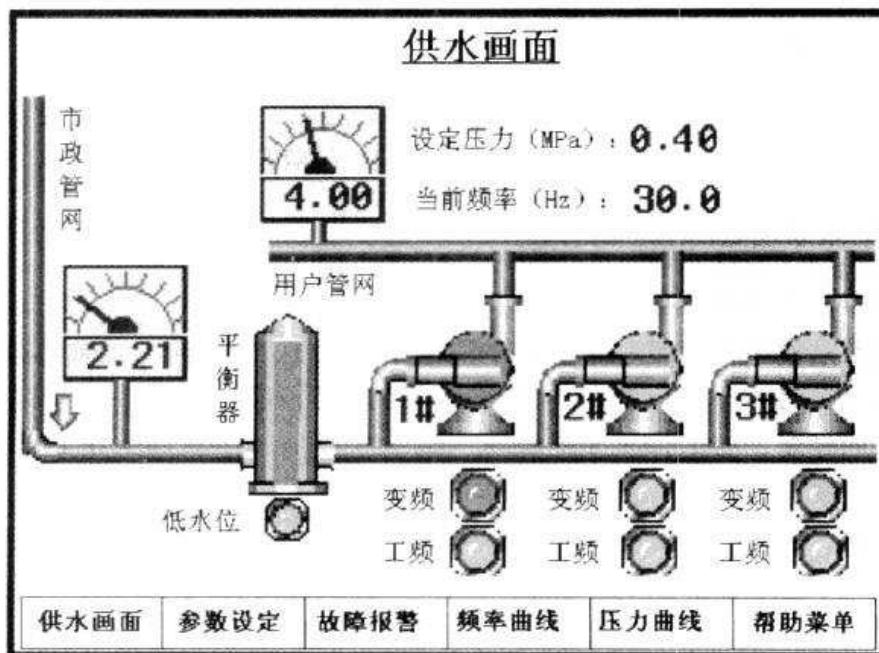


图 3-9 触摸屏在供水控制中的应用



a) 设备工作状态显示，如指示灯、按钮、文字、图形、曲线等；b) 数据、文字输入操作，打印输出；c) 生产配方存储，设备生产数据记录；d) 简单的逻辑和数值运算。

2. 触摸屏的使用方法

触摸屏的使用方法如图 3-10 所示，包括以下步骤：

- 1) 明确监控任务要求，选择适合的触摸屏产品；
- 2) 在 PC 上用画面组态软件编辑“工程文件”；
- 3) 测试并保存已编辑好的“工程文件”；
- 4) PC 连接触摸屏硬件，下载“工程文件”到触摸屏中；
- 5) 连接触摸屏和工业控制器（如 PLC、仪表等），实现人机交互。

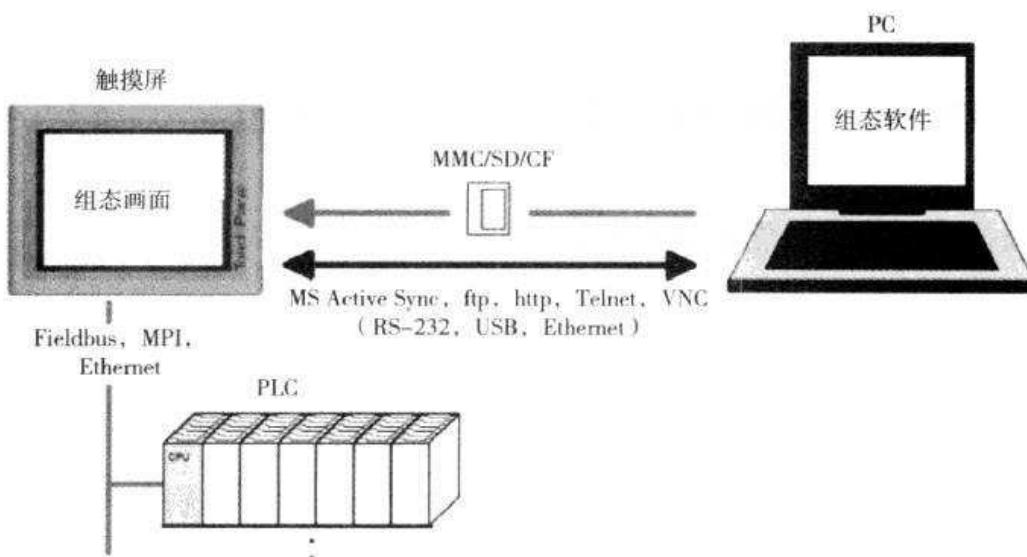


图 3-10 触摸屏的使用方法

3.2.2 触摸屏的设计原则

一个优良的触摸屏界面设计，需要符合操作者的需求，考虑到操作者的身、心理状况，一个友善便利的触摸屏界面不但可增加操作的方便性，亦可减少操作错误率，使机器、设备和器材发挥最大的功能，达到造福用户的目的。

触摸屏的画面设计应该遵循以下法则：

(1) 以通信功能作为界面设计的核心

触摸屏的关键是使人与机器之间能够准确地交流信息：一方面，人向机器输入时应当尽量采取自然的方式；另一方面，机器向人传递的信息必须准确，不致引起误解或混乱。另外，不要把内部的处理、加工与触摸屏混在一起（触摸屏程序只是通信），以免互相干扰，影响速度。

设计触摸屏时，针对每一个功能，都要按照“I-P-O”的模块化思想，使输入、处理与输出“泾渭分明”，充分体现触摸屏的通信功能。这样设计出来的触摸屏不但不易出错，而且易于维护，即使有了错误也很容易加以改正。图 3-11 所示为触摸屏的通信过程。

通过图 3-11 可以对人机交互进行如下分析：

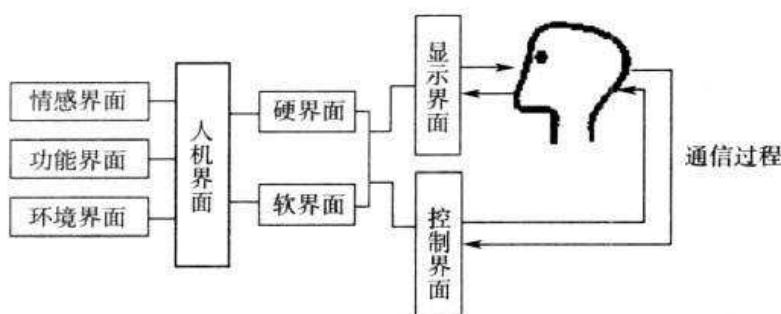


图 3-11 通信过程

1) 使用环境的分析：环境条件是影响人机关系的外界因素，如产品的使用场所、气候、季节、时间等。因为使用环境不同，使用的条件就不同。例如，冶金工业和公用事业的触摸屏因为使用环境不同，设计要求也不一样。

2) 使用者的分析：使用者的分析主要包括使用对象、使用者生理状态和条件以及使用者行为方式等的分析。因为使用对象不同，操作的尺寸、用力大小、对色彩的喜好等就不一样。在人机分工时，要根据使用者的生理状态和工作时的状态确定任务分析。使用行为是由于年龄、性别、地区、种族、职业、生活习惯、受教育程度等原因形成动作习惯。

3) 使用过程的分析：使用过程分析是一项深入细致的工作。一些产品中，人机问题不是靠常识可以发现的，甚至短时间使用也体会不到。因此必须对使用过程进行认真分析。

通过用户通信模型的建立，我们可以明确人机系统是否合理、人机分工是否恰当、是否体现了人本主义思想等。

(2) 设计合理的文字区规划

有些设备或器材的显示屏看起来像个棋盘，毫无秩序可言。当然有时候别出心裁的界面设计或许会带给操作者眼睛一亮、赏心悦目的视觉感，但无论如何，井然有序的文字区规划，才易达成最有效、最舒适的效果。

试着让说明具有顺畅地阅读连贯性。举个例来说，一个向中对齐与一个向左对齐的文字区块，明显地，向左对齐的区块读起来顺畅多了，因为使用者在读完一行字后，自然而然地往左下方继续读下去，不需要重新找寻开头，相较之下，向中对齐的区块看起来就混乱多了。因此，有规矩地文字区规划可大幅度提增使用者界面的质量。

(3) 色彩和字型的合理应用

一个触摸屏中（包括背景和操作组件），最好不要超过3~5种颜色，最好大区域使用浅灰阶色调，如浅灰、白色等；重要操作组件则以红、蓝、绿等彩度较高的颜色来表达，但颜色不宜过多，且区域不可以太大，否则看起来过度混乱，反而失去当初欲强调重点的用意了。

对管道或化学工程师，如果显示是红色的，它是关闭的，而显示绿色表明其在运行。但对电力工程师，显示红色表明其是带电的，它是火线！不要碰它！如果是绿色，表明它是关闭且安全的。那是我们使用硬性规定的编码颜色来显示对象。没有标准化，就没有办法不作



修改即从化工应用进入到电力应用。

一个界面中，最好不要有太多的字型，更不宜选用字型太复杂或软弱无力的字体，如斜体字等，越简洁清楚辨识性越佳。

(4) 减少显示器的视觉密度

一般机器、设备和器材所需表达的信息很多，易造成显示和操作界面看起来非常拥挤，所有信息挤在一团，别说是最终用户，连专业的电气设计人员都未必一眼就能获得所需的信息或知道如何正确操作之。

因此，设计触摸屏时，务必要考虑到内容的配置与空间裕度的保留，适当的留白将使整个界面看起来清楚而美观。有几点适当保留空间的方法可供参考：

1) 把次要信息用选项功能或连接功能将其置于选项中，需要时再选取即可；

2) 降低品牌识别图像的大小，像品牌 logo、名称等，无需为了彰显品牌形象而刻意放大；

3) 使用简单的图形，尽量 2-D 平面化，不要使用过度复杂的图像；

4) 使用空白空间，而非线条去区分文字内容，尽量使整个界面简易化；

5) 使用简洁的语句表达信息，避免内容过多。

(5) 使用简明扼要的词句和精简图像

操作界面中，按钮、图标以及其他功能键的设计要符合用户的认知习惯。当用户使用产品时，尽量不需要通过太多的学习，通过按钮的形态、色彩以及上面的图标就可以进行操作。言简意赅地点出重点，叙述性的文字只会让使用者失去耐性，并让整体界面看起来复杂难懂。

对于触摸屏来说，需要避免使用复杂的图像，以象形符码的表示为最佳，只要能引起使用者的共识，越简单越好。而对于操作面板来说，在增加功能时，要尽量采用复用按键，加强在屏显示提示等方法。一般来说，按键多，可能有些消费者会认为产品档次高，但太多的按键会增加操作的难度。

无论是以色彩或字体来表示某项含意，一定要前后一致，红色代表危险、粗体代表强调，决不可随意应用，否则会造成使用者的混淆。

一致的触摸屏不致增加用户的负担，让用户始终用同一种方式思考与操作。最忌讳的是每换一个屏幕用户就要换一套操作命令与操作方法。就如同 Windows 下的应用软件之所以倍受青睐，与其界面的一致性不无关系。因此，在触摸屏画面设计中也不妨用问号图标表示帮助、以磁盘图标表示存盘、以打印机图标表示打印，等等。

(6) 安全提示、报警和故障

机器、设备和器材的使用安全性也与触摸屏设计的好坏有关，因为设计上的缺失将会直接或间接的造成操作时的失误，严重时还可能造成使用者的伤害或死亡。

比如在输入过程控制中的工艺数据时，由于误操作、按键连击等，均有可能导致数据误录。巧妙地进行程序设计，可以避免此类状况造成的错误。例如，输入电动机转速时，我们可以对其范围进行限定，使用户无法输入 0 ~ 1500r/m 以外的数据，这样就可以确保电动机的正常运行。

还有一个经常会被忽略的人机工程学领域，即如何设置报警，这是人机工程学的一个部分，人们应给予足够的注意。相同的报警不能用于每一个故障事件，声音必需引起操作人员



的警觉，但不能到达干扰人们反应能力的程度。不少触摸屏给出太多的脉冲和闪烁信号，这都不符合人机工程学的设计。

当使用触摸屏产品，其功效如同任何计算机时，应对相同的警告和制约进行检验。需将重复的动作压缩到最小。触摸屏对任何的修改是开放的，从而能适应不同的需要。

由此可知，一项产品，尤其与操作者人身安全有着密切关系时，在设计时绝对不可不谨慎。

3.2.3 西门子公司的 TIA 思想

TIA 思想，即全集成自动化思想，它是用一种系统完成原来由多种系统搭配起来才能完成所有功能的系统。应用这种解决方案，可以大大简化系统的结构，减少了大量的接口部件，应用全集成自动化可以克服上位机和工业控制器之间、连续控制和逻辑控制之间、集中与分散之间的界限。同时，全集成自动化解决方案还可以为所有的自动化应用提供统一的技术环境，主要包括：统一的数据管理、统一的通信、统一的组态和编辑软件等。基于这种环境，不同的技术可以在一个用户接口下集成在一个有全局数据库的系统中。

总而言之，在西门子公司的理念中，TIA 思想 = 产品 + 集成化 + 开放性。

3.3 技能训练：KTP 600 触摸屏的使用

3.3.1 任务说明

如图 3-12 所示为西门子公司的 KTP 600 Basic Color PN 触摸屏（以下简称 KTP 600 触摸屏），请对它进行电气接线，并与 S7-1200 PLC 进行画面组态、下载与调试。具体的控制对象为电动机的起动与停止的触摸屏监控，设 I0.0 为触摸屏控制与现场按钮控制的选择开关、I0.1 为现场启动按钮、I0.2 为现场停止按钮、Q0.0 为电动机接触器。

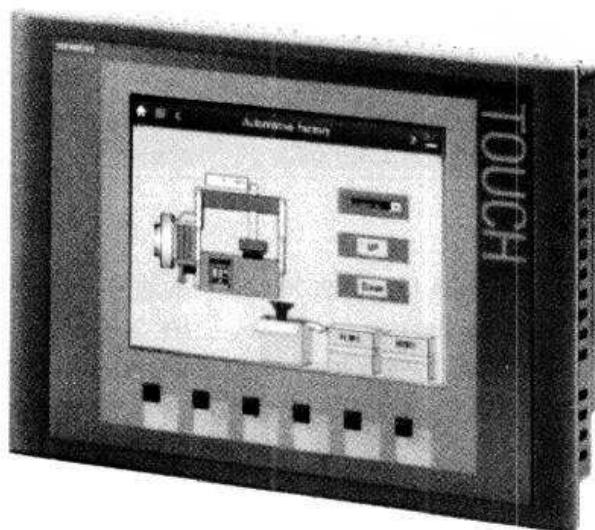


图 3-12 KTP 600 触摸屏外观



3.3.2 电气接线

1. 熟悉 KTP600 的端子 (见图 3-13)

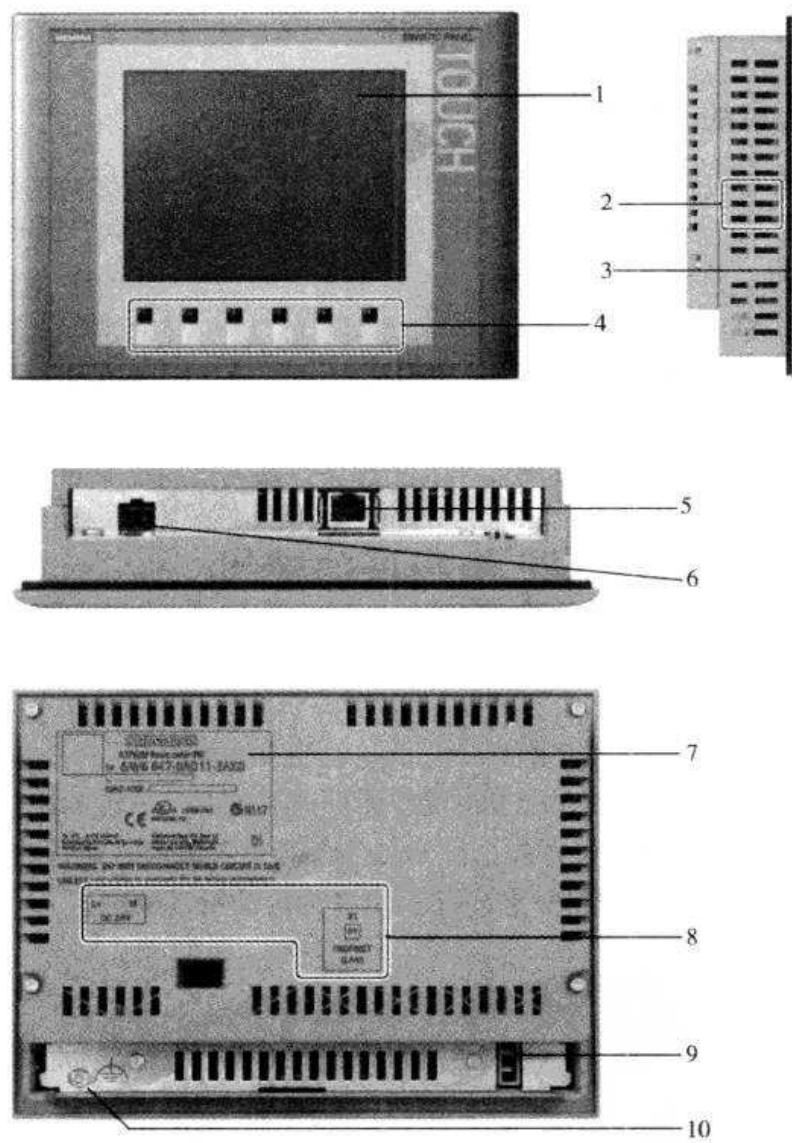


图 3-13 KTP600 触摸屏的端子

1—显示屏/触摸屏 2—夹紧端子的开口 3—嵌入式密封件 4—功能键 5—PROFINET 接口
6—电源接口 7—铭牌 8—接口名称 9—记录带导槽 10—功能接地的接口

2. 选择合适的安装方式

对于 KTP600 来说，既可以选择横向安装，也可以选择纵向安装，如图 3-14 所示。触摸屏自行通风，可垂直或倾斜安装于操作箱、电控箱等。如果超出了操作设备运行允许的最高环境温度，则需要使用外部通风设备。

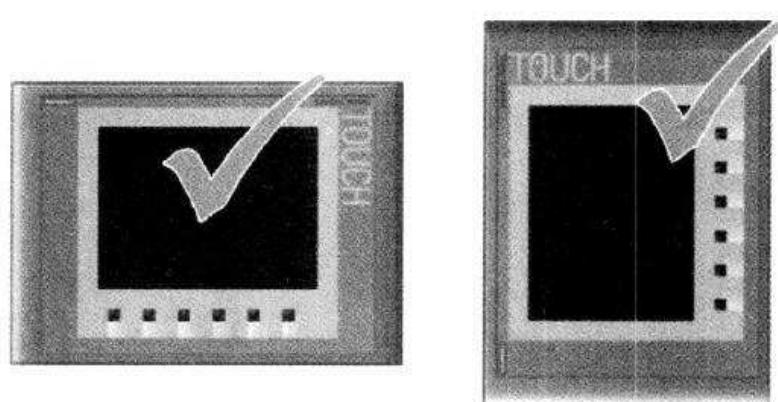


图 3-14 横向和纵向安装

3. 将触摸屏安装到电控箱表面

如果需要，将嵌入式密封件插入操作设备背面的槽内。确定嵌入式密封件没有扭转。要达到防护等级 IP65 的要求，需要正确放入嵌入式密封件，然后将 KTP600 从前面装入电控箱表面（见图 3-15）。

如图 3-16 所示，将第一个夹紧端子放在触摸屏 KTP600 背面开口的第一个位置上。用螺钉旋具固定夹紧端子，紧固螺钉允许的最大扭矩为 $0.2 \text{ N} \cdot \text{m}$ 。

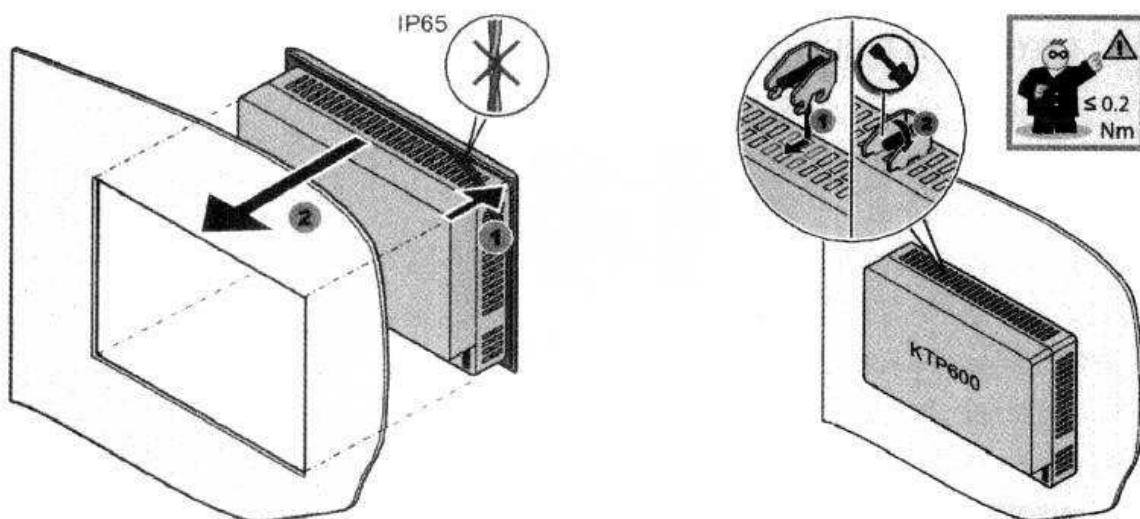


图 3-15 KTP600 安装到电控箱表面

图 3-16 螺钉紧固

4. 连接触摸屏的 DC 24V 直流电源

连接触摸屏的 DC 24V 直流电源如图 3-17 所示：1) 将两根电源电缆接入电源接线柱，并用开槽螺钉旋具固定电缆；2) 将电源接线柱与操作设备相连；3) 关闭电源；4) 将余下的电缆两端接入电源的接口，并用开槽螺钉旋具固定，此时注意极性是否正确。

5. 使用以太网连接触摸屏与 PC、PLC

KTP600 与 PC 相连如图 3-18 所示：1) 关闭操作设备；2) 将 LAN 电缆的一个 RJ45 插头与触摸屏相连；3) 将 LAN 电缆的一个 RJ45 插头与组态 PC 相连。



图解西门子 S7-1200 PLC 入门到实践

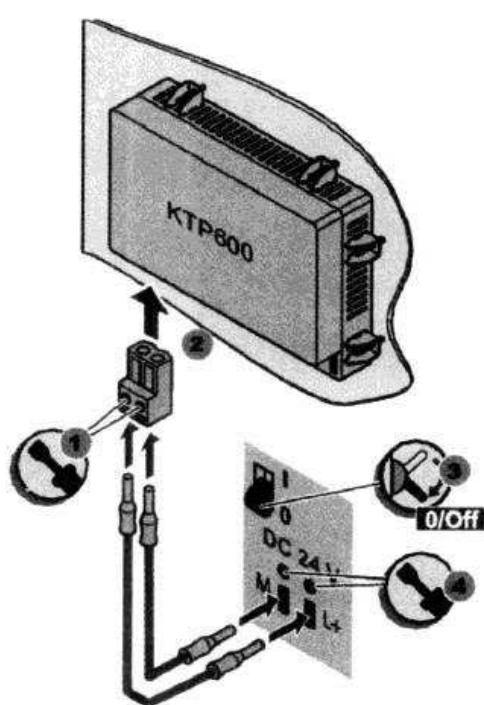


图 3-17 连接触摸屏的 DC 24V 直流电源

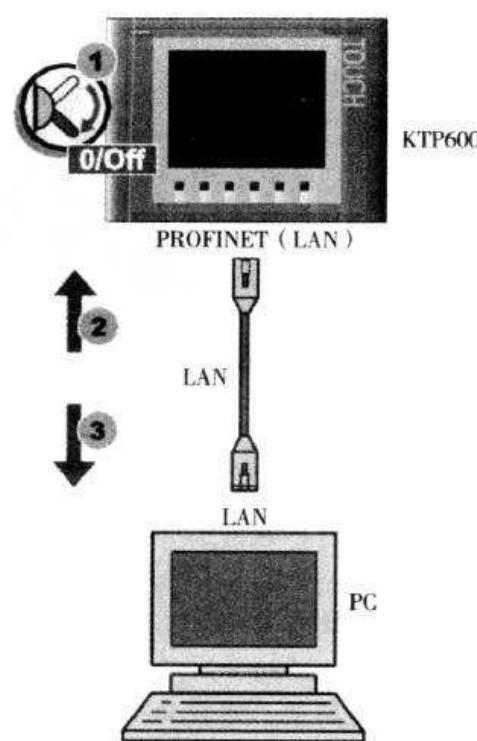


图 3-18 KTP600 与 PC 相连

同理，KTP600 与 PLC 相连如图 3-19 所示。

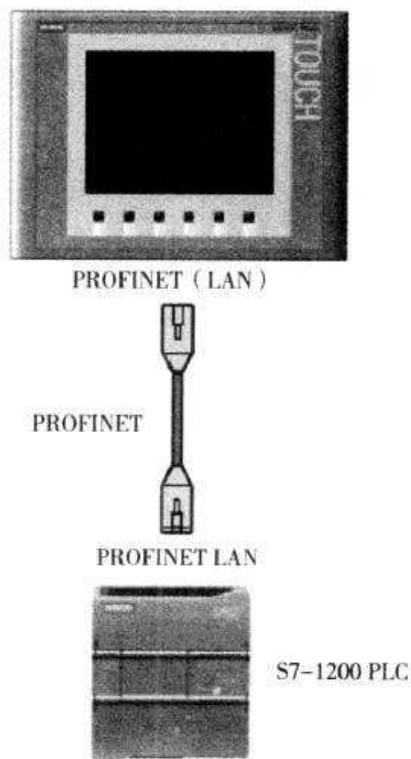


图 3-19 KTP600 与 PLC 相连

在本书中，建议采用如图 3-20 所示的方式进行连接，即通过集线器来连接 PC、PLC 与



触摸屏 KTP600。

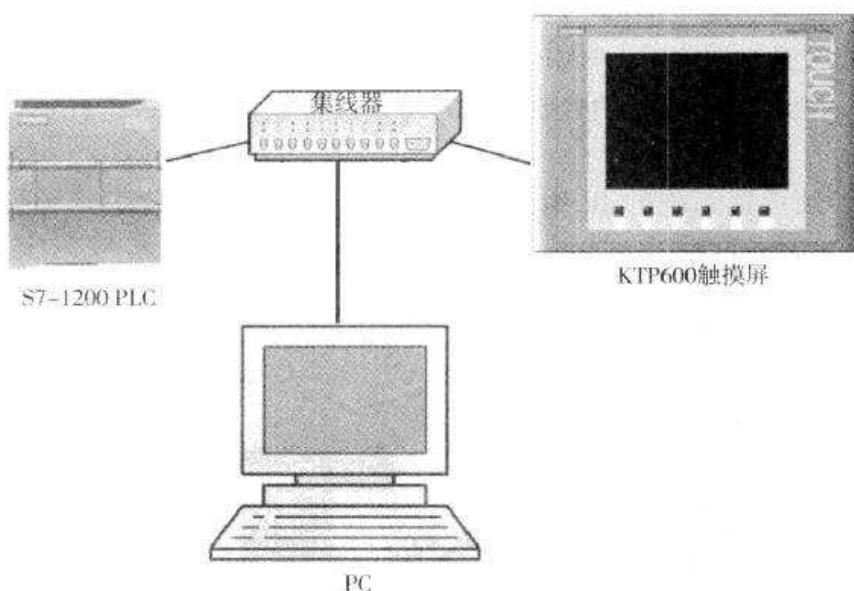


图 3-20 通过集线器连接 PC、KTP600 与 PLC

集线器是属于数据通信系统中的基础设备，它和双绞线等传输介质一样，是一种不需任何软件支持或只需很少管理软件管理的硬件设备。它被广泛应用在各种场合。集线器工作在局域网（LAN）环境，像网卡一样，应用于 OSI 参考模型第一层，因此又被称为物理层设备。集线器内部采用了电器互联，当维护 LAN 的环境是逻辑总线或环型结构时，完全可以用集线器建立一个物理上的星形或树形网络结构。在这方面，集线器所起的作用相当于多端口的中继器。其实，集线器实际上就是中继器的一种，其区别仅在于集线器能够提供更多的端口服务，所以集线器又叫多口中继器。

依据 IEEE 802.3 协议，集线器的功能是随机选出某一端口的设备，并让它独占全部带宽，与集线器的上联设备（交换机、路由器或服务器等）进行通信。集线器主要用于共享网络的组建，是解决从服务器直接到桌面最经济的方案。在交换式网络中，Hub 直接与交换机相连，将交换机端口的数据送到桌面。使用集线器组网灵活，它处于网络的一个星形结点，对结点相连的工作站进行集中管理，不让出问题的工作站影响整个网络的正常运行，并且用户的加入和退出也很自由。

6. 接通并测试 HMI 设备

接通电源，在电源接通之后屏幕会亮起，启动期间会显示进度条。如果 HMI 设备无法启动，可能是将电源端子上的电线接反了。

操作系统启动后，装载程序将打开（见图 3-21）。

在图 3-21 所示的菜单中，按“传送（Trans-

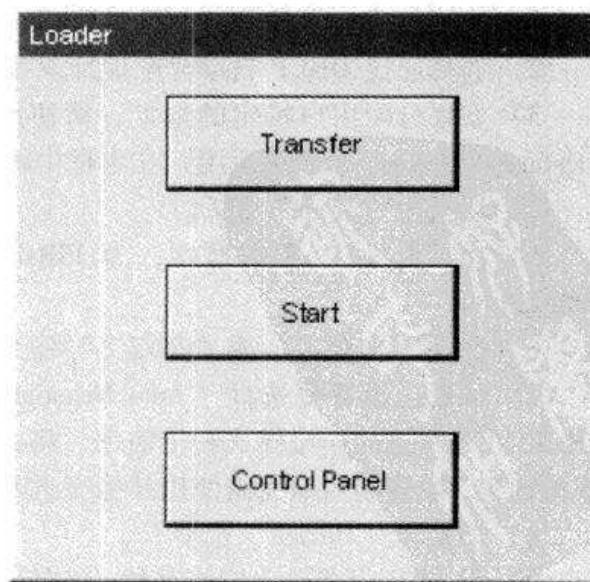


图 3-21 装载程序打开



图解西门子 S7-1200 PLC 入门到实践

fer)" 按钮可以将 HMI 设备设置为“传送”模式，仅当至少启用了一个数据通道用于传送时，才能激活传送模式；按“启动（Start）”按钮可以启动 HMI 设备上的项目，如果不执行操作，则在经过了延迟时间后，HMI 设备上的项目会自动启动；按“控制面板（Control Panel）”按钮可以打开 HMI 设备的控制面板，可以在控制面板中进行各种设置，如传送设置等。

由于本系统必须设置正确的 IP 地址才能确保从 PC 的组态传送到触摸屏，因此，按“控制面板（Control Panel）”进入如图 3-22 所示的界面。

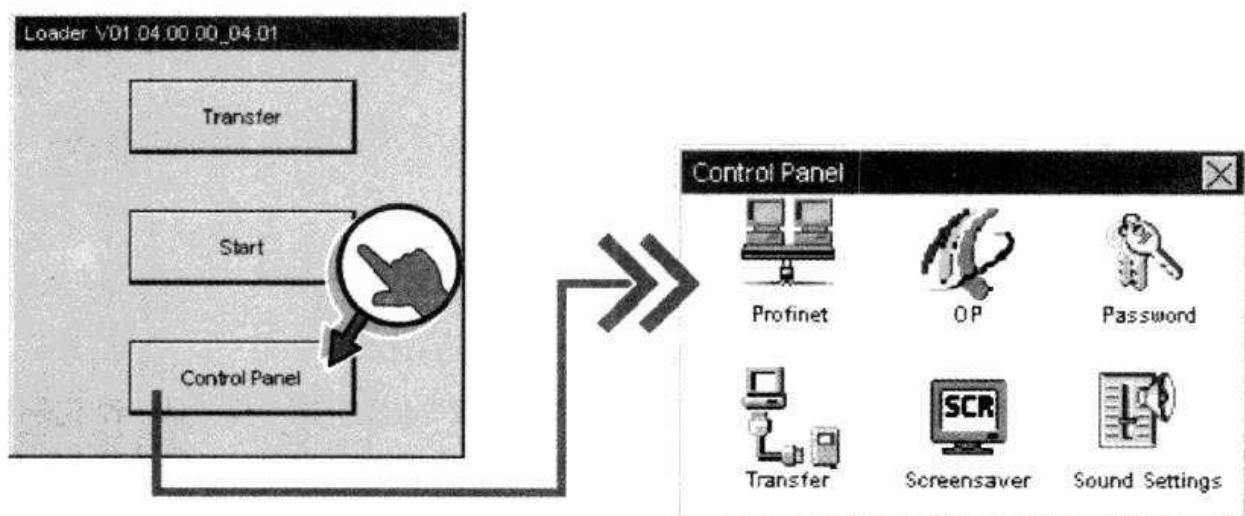


图 3-22 进入控制面板

在控制面板中组态 HMI 设备，可进行以下设置：通信设置；操作设置；密码保护；传送设置；屏幕保护程序；声音信号。

由于 KTP600 触摸屏采用以太网通信，因此必须设置正确的通信模式，具体步骤如图 3-23 所示：

- 1) 按“Profinet”按钮，打开“Profinet Settings”对话框。
- 2) 选择通过 DHCP 自动分配地址或者执行用户特定的地址分配。
- 3) 如果分配用户特定的地址，请使用屏幕键盘在“IP Address”、“Subnet Mask”和“Default Gateway”（如果适用）文本框中输入有效 IP 地址。
- 4) 切换至“Mode”选项卡。
- 5) 在“Speed”文本框中，为 PROFINET 网络输入传输率，有效值为 10Mbit/s 和 100Mbit/s。
- 6) 选择“半双工”或“全双工”作为连接模式。
- 7) 如果已设置复选框“Auto Negotiation”，则会激活以下功能：将会自动检测和设置 PROFINET 网络的连接模式和传输率、将会激活“自动交叉”功能，这意味着 HMI 设备可连接到 PC 或控制器而无需使用其他交叉电缆。
- 8) 切换至“Device”选项卡。
- 9) 为 HMI 设备输入网络名称，该名称必须满足以下条件：
 - 最大长度：240 个字符；

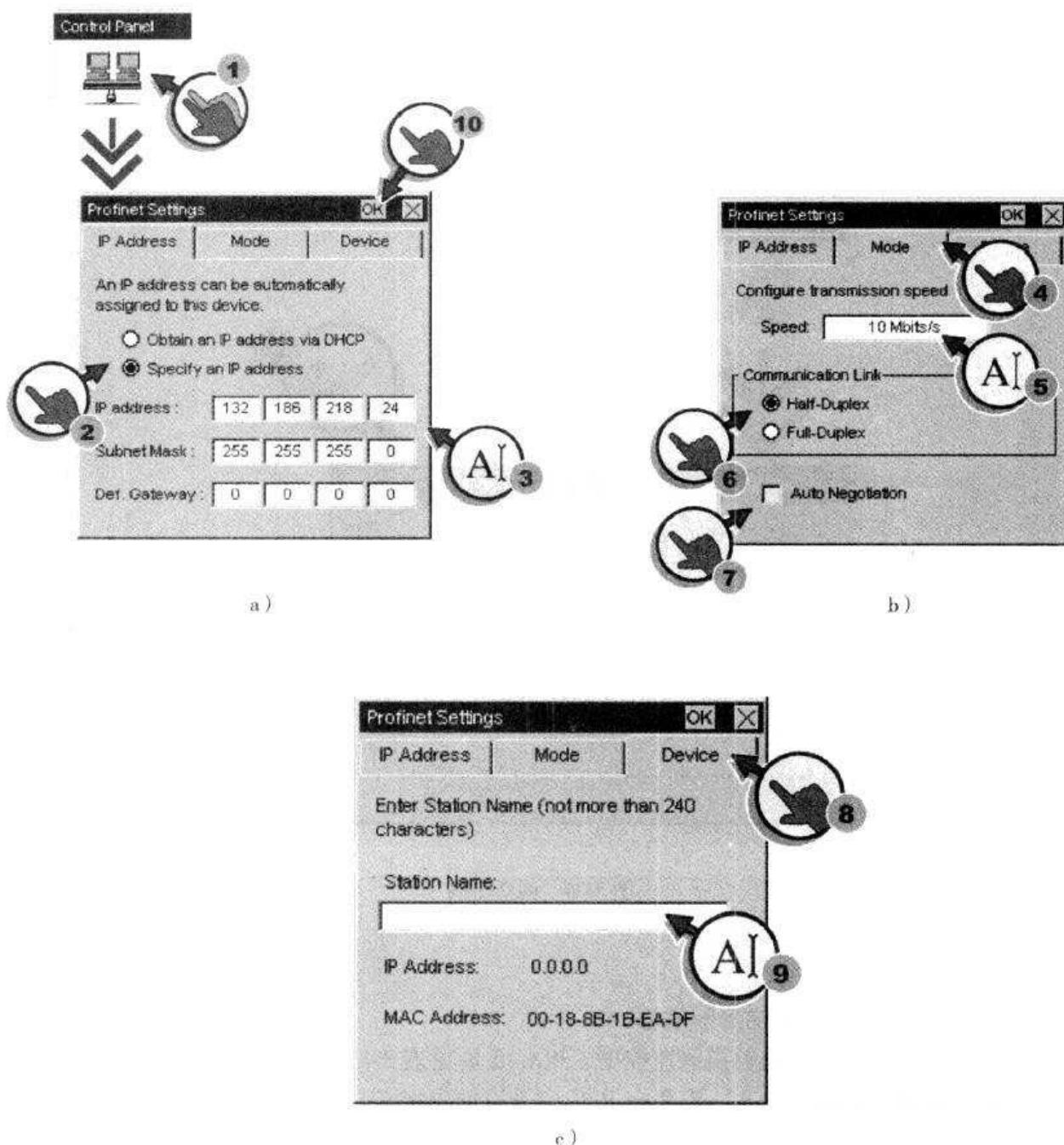


图 3-23 Profinet Settings 菜单

- 特殊字符：仅限“-”和“.”；
- 无效语法：“n.n.n.n”（n=0 到 999）以及“端口-yxz”（x、y、z=0 ~ 9）。
- 10) 点击“OK”关闭对话框并保存输入内容。

7. 设置触摸屏的密码

密码保护可以防止对控制面板进行未经授权的访问，密码不能包含空格或特殊字符 *、?、.、%、/、\、'、"。如果忘记控制面板的密码，则必须先更新操作系统，之后才能在控制面板中进行更改。在更新操作系统时，HMI 设备上的所有数据都将被覆盖。



如图 3-24 所示，将激活密码设置。具体步骤如下：

- 1) 点击“Password”按钮，打开“Password Properties”对话框。
- 2) 在“Password”文本框中输入密码。触摸文本框。字母数字屏幕键盘将显示。
- 3) 在“Confirm Password”文本框中确认密码。
- 4) 点击“OK”关闭对话框并保存输入内容。

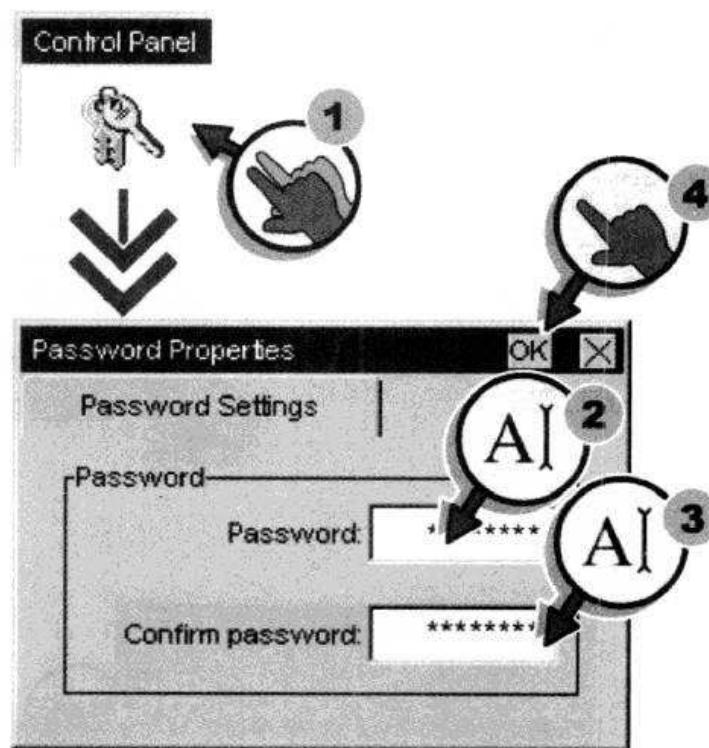


图 3-24 激活密码

3.3.3 软件编程

1. PLC 的变量定义与梯形图编程

在电动机起动与停止的触摸屏监控中，PLC 变量见表 3-1，可以看出触摸屏上按钮的启动与停止（%M0.0 和 %M0.1）需要在 PLC 中进行预先定义。

表 3-1 PLC 变量

| | 名称 | 数据类型 | 地址 |
|---|-----------|------|-------|
| 1 | ● 转换开关 | Bool | %I0.0 |
| 2 | ● 启动 | Bool | %I0.1 |
| 3 | ● 停止 | Bool | %I0.2 |
| 4 | ● 电机接触器 | Bool | %Q0.0 |
| 5 | ● 触摸屏启动按钮 | Bool | %M0.0 |
| 6 | ● 触摸屏停止按钮 | Bool | %M0.1 |
| 7 | ● 中间变量1 | Bool | %M0.2 |
| 8 | ● 中间变量2 | Bool | %M0.3 |



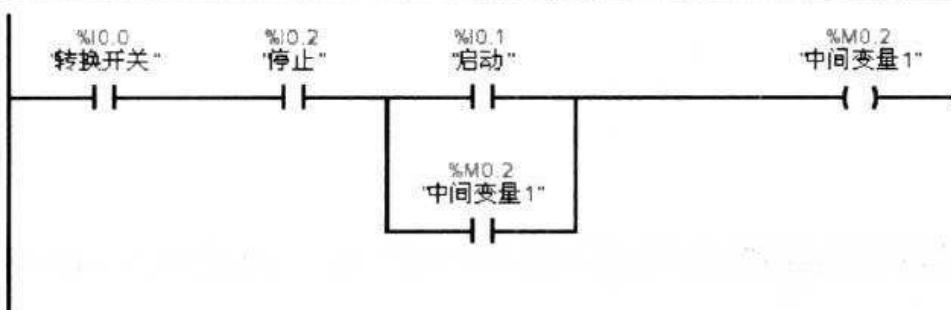
主程序的 PLC 梯形图如图 3-25 所示。

▼ 块标题：

电动机起动停止的触摸屏监控

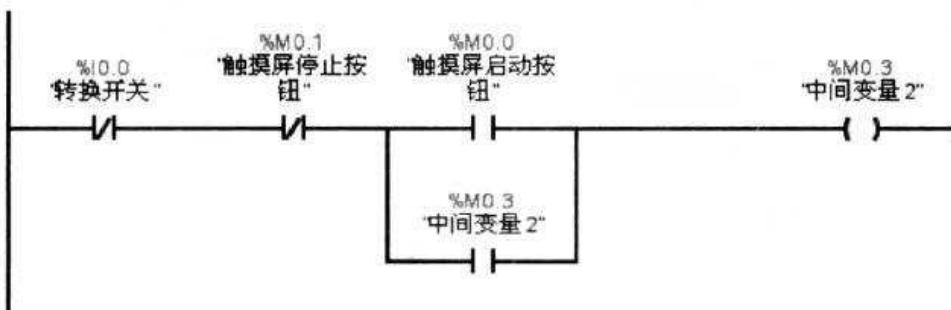
▼ 程序段 1：

当转换开关切换到本地时，本地按钮起动与停止电动机



▼ 程序段 2：

当转换开关切换到触摸屏时，触摸屏按钮起动与停止电动机



▼ 程序段 3：

两种情况下的中间变量均触发电动机接触器

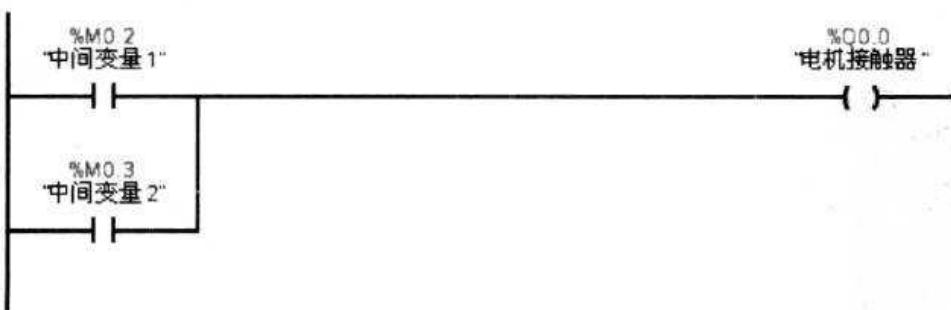


图 3-25 主程序的 PLC 梯形图

2. 触摸屏的设备向导

在 TIA 软件中新增的触摸屏一般建议使用触摸屏的设备向导。如图 3-26 所示，在项目树中选择“添加新设备”。

图 3-27 所示的“添加新设备”窗口中将会出现 SIMATIC PLC 或 SIMATIC HMI 两种，选择 SIMATIC HMI，并按照用户的 KTP 触摸屏型号选择，比如本案例选择 KTP600PN。

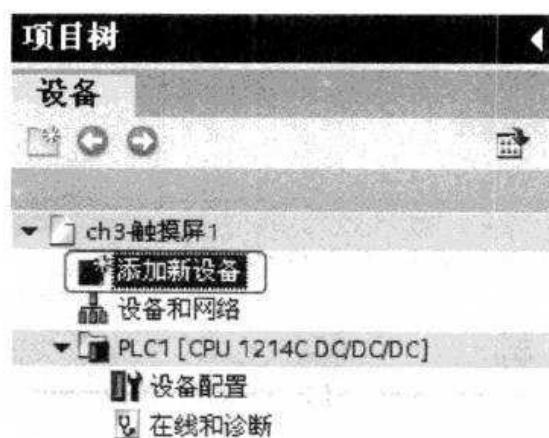


图 3-26 添加新设备

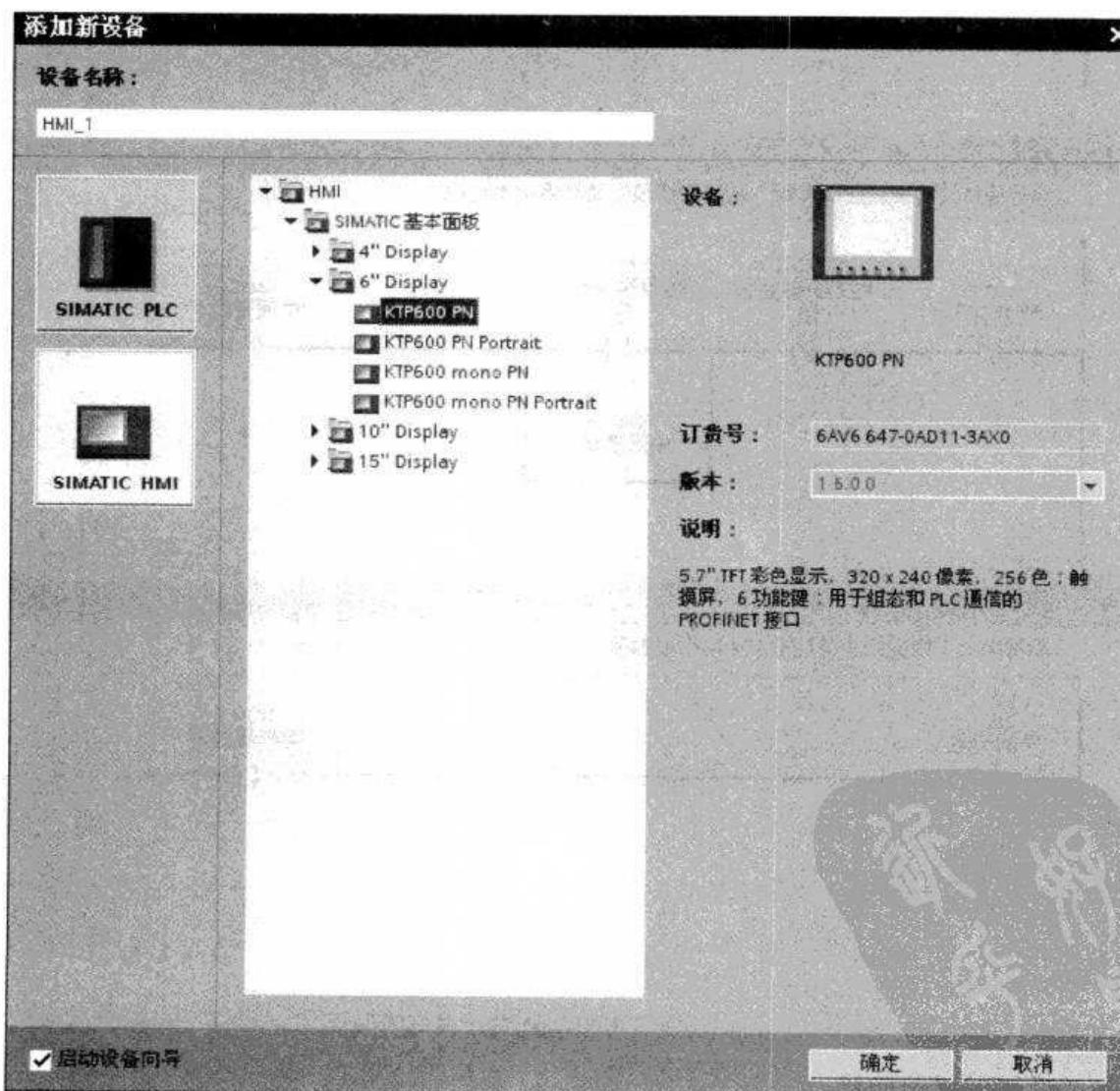


图 3-27 选择 KTP600PN

在图 3-27 中最左下角的“启动设备向导”中打钩，则进入图 3-28 所示的 HMI 设备向导 (KTP600PN)。

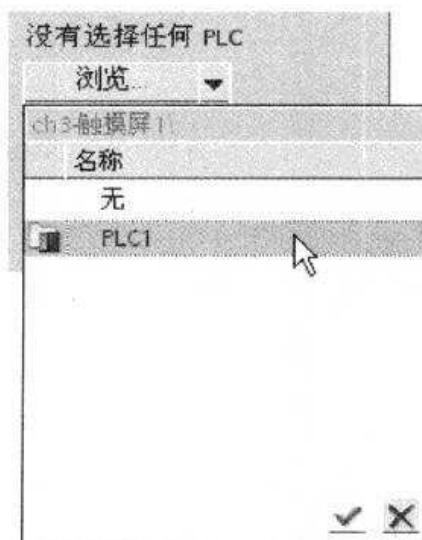
第3章 KTP 触摸屏编程



图 3-28 进入设备向导

在图 3-28 所示的设备向导中出现通信驱动程序为 SIMATIC S7-1200，接口为 PROFINET，但此时没有选择任何 PLC，点击“浏览”按钮，进行 PLC 选择（见图 3-29a 的 PLC1），此处的 PLC1 即在 TIA 软件建立的 PLC 名称。

一旦选择了 PLC，就会看到 HMI 设备向导的画面发生了变化，HMI 与 PLC 建立了连接（见图 3-29b）。

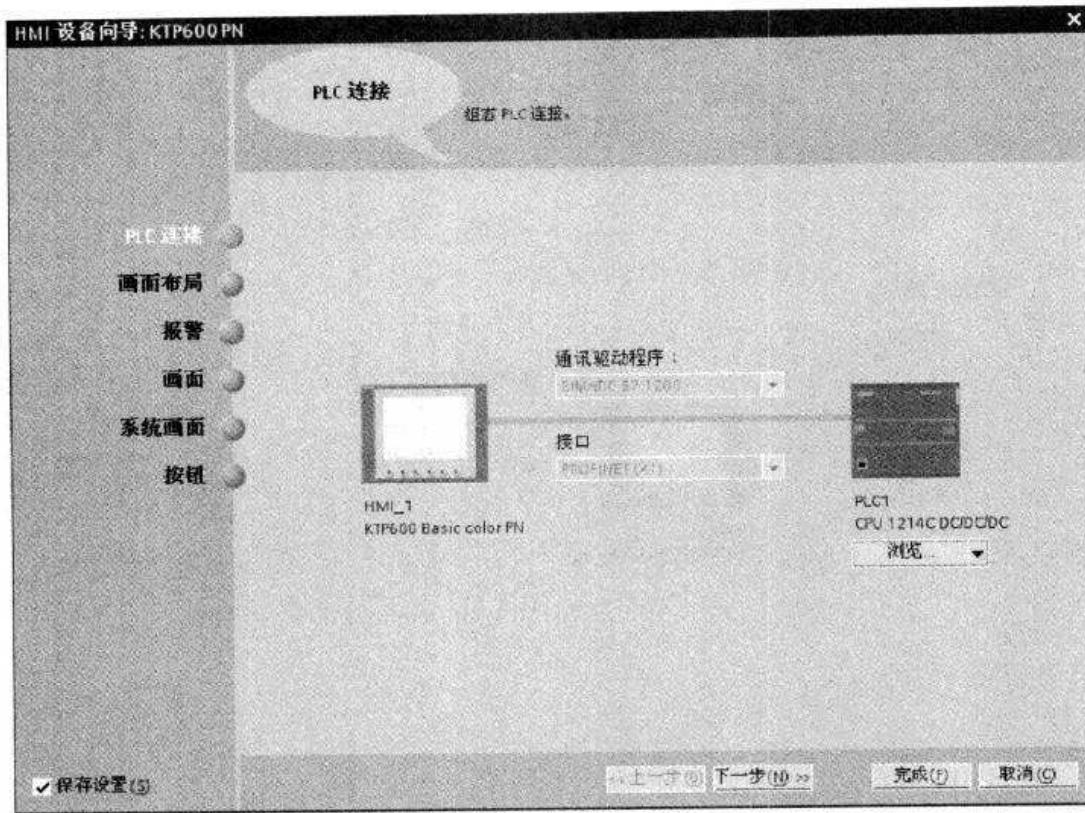


a)

图 3-29 建立了 PLC 与 HMI 通信的 HMI 设备向导



图解西门子 S7-1200 PLC 入门到实践



b)

图 3-29 建立了 PLC 与 HMI 通信的 HMI 设备向导（续）

点击“下一步”，出现了如图 3-30 所示的“画面布局”，包括画面的分辨率、背景色、

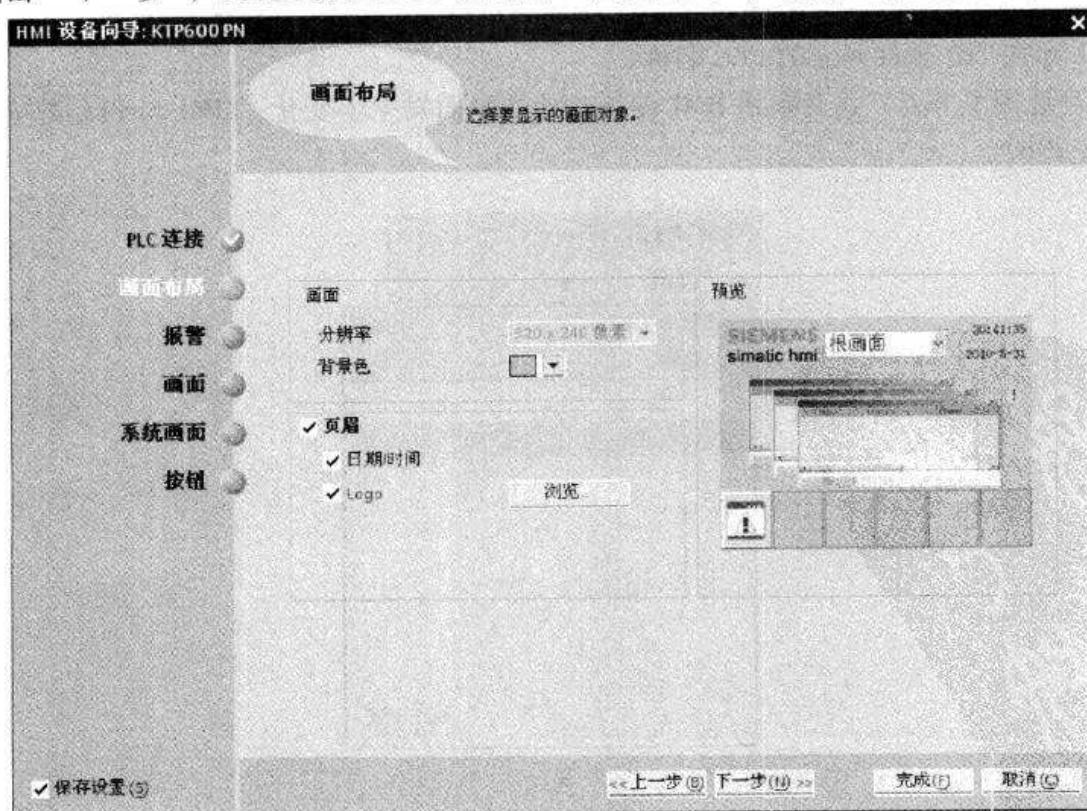


图 3-30 设置画面布局

第3章 KTP 触摸屏编程



页眉等。

点击“下一步”，则出现了如图 3-31 所示的报警设置。

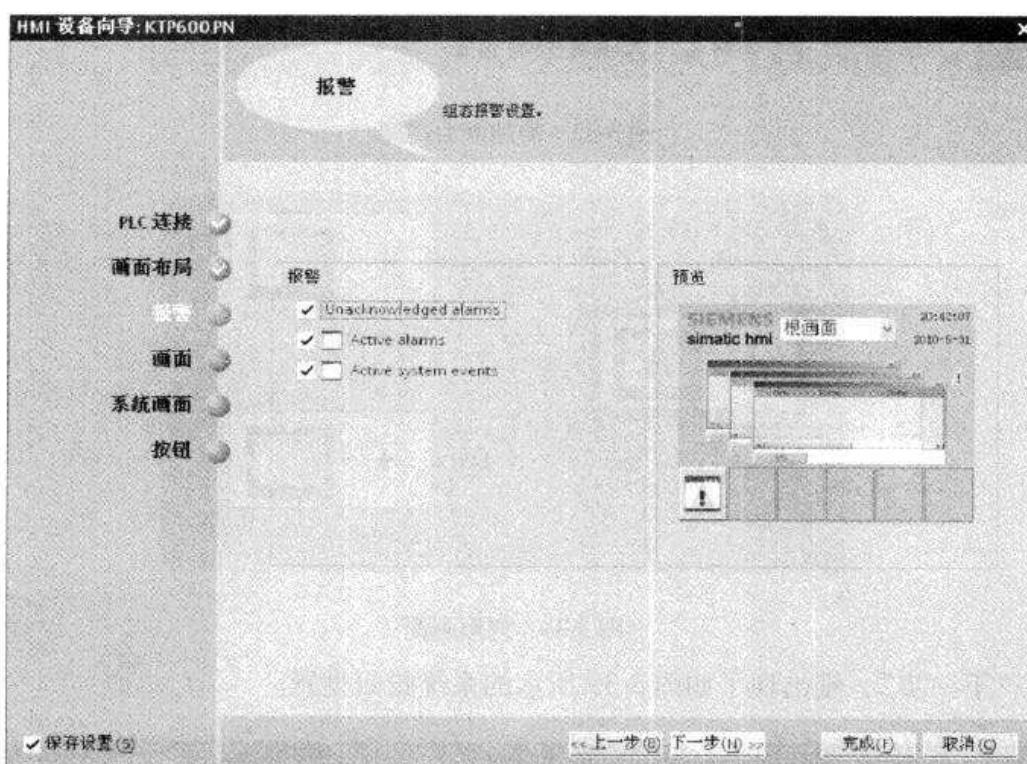


图 3-31 报警设置

点击“下一步”，则看到如图 3-32 所示的画面设置，包括添加画面、删除画面、重命名等功能。

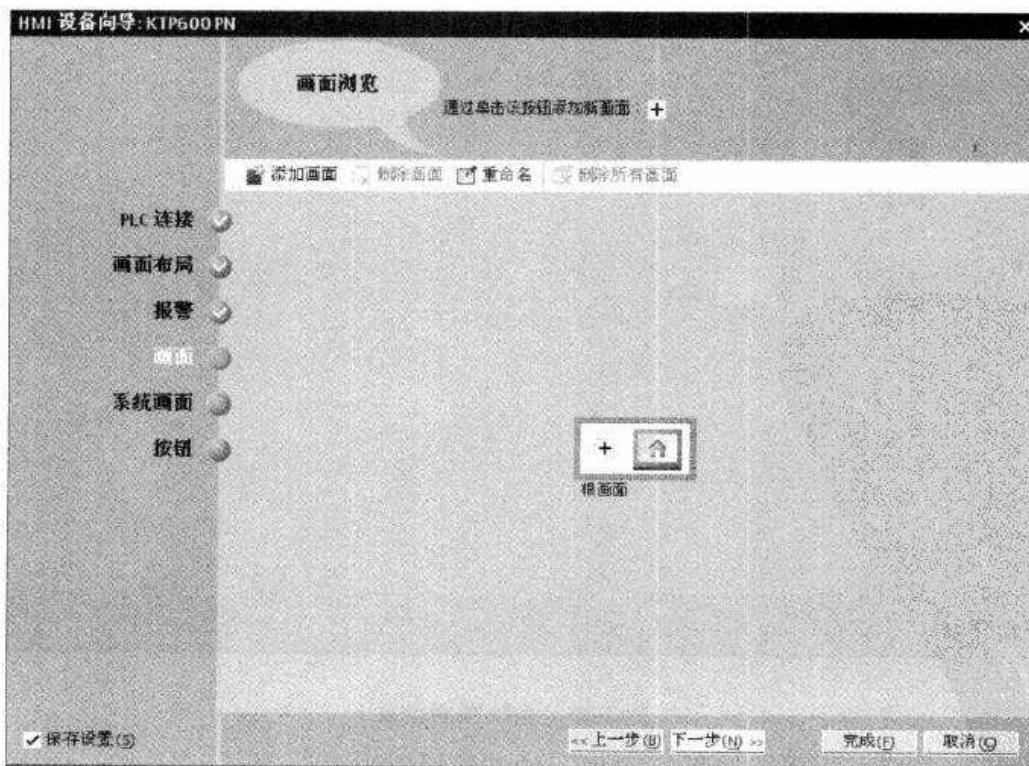


图 3-32 画面设置



图解西门子 S7-1200 PLC 入门到实践

在图 3-32 所示的画面设置中，选择添加画面功能（见图 3-33），则会跳出如图 3-34 所示的树形画面，包括根画面、Screen0 和 Screen1。



图 3-33 添加画面功能

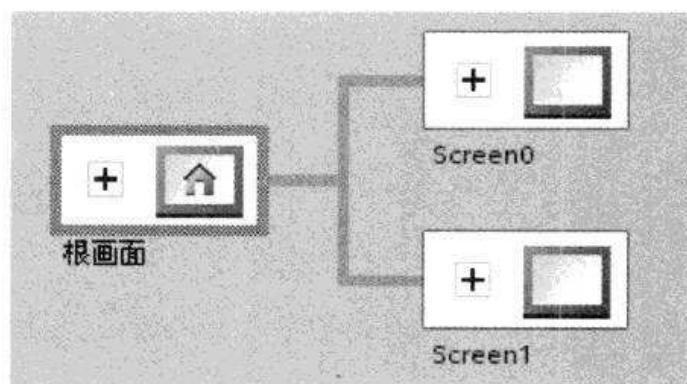


图 3-34 树形画面

点击“下一步”，则出现了如图 3-35 所示的系统画面设置。

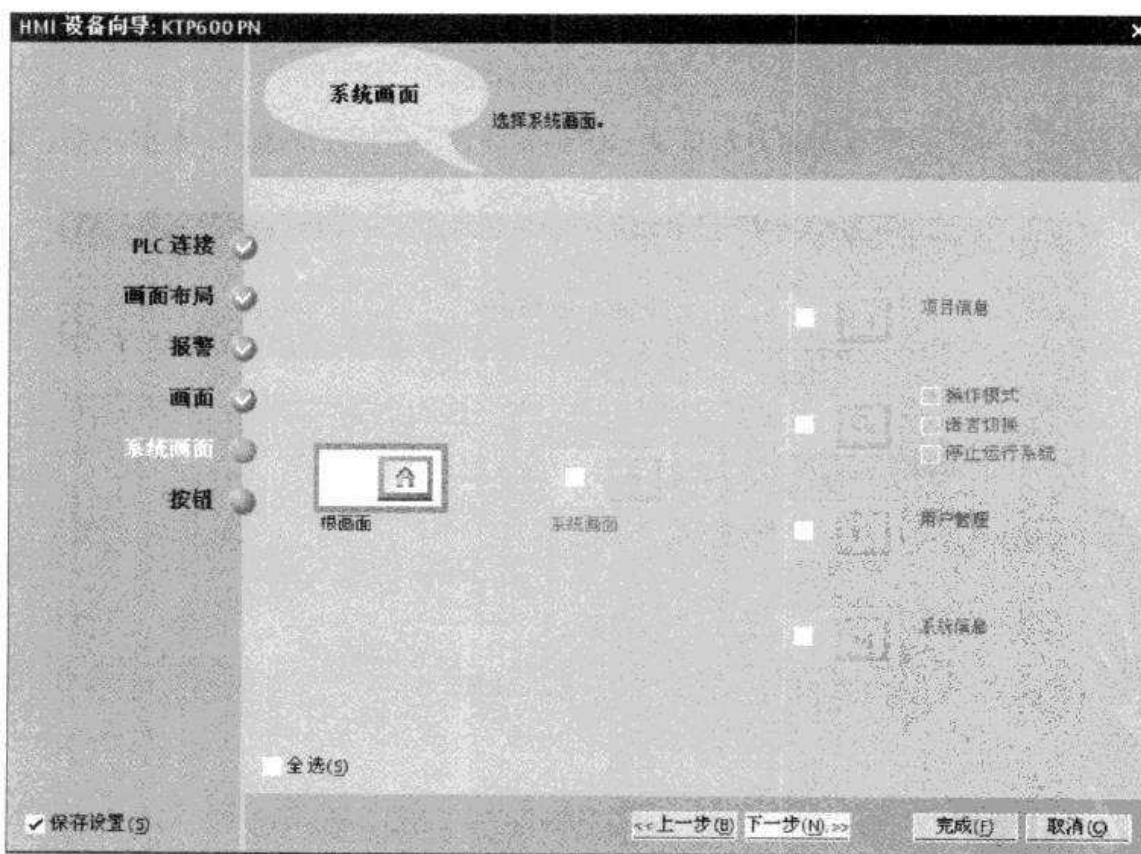


图 3-35 系统画面设置

点击“下一步”，出现了如图 3-36 所示的按钮设置画面。

第3章 KTP 触摸屏编程

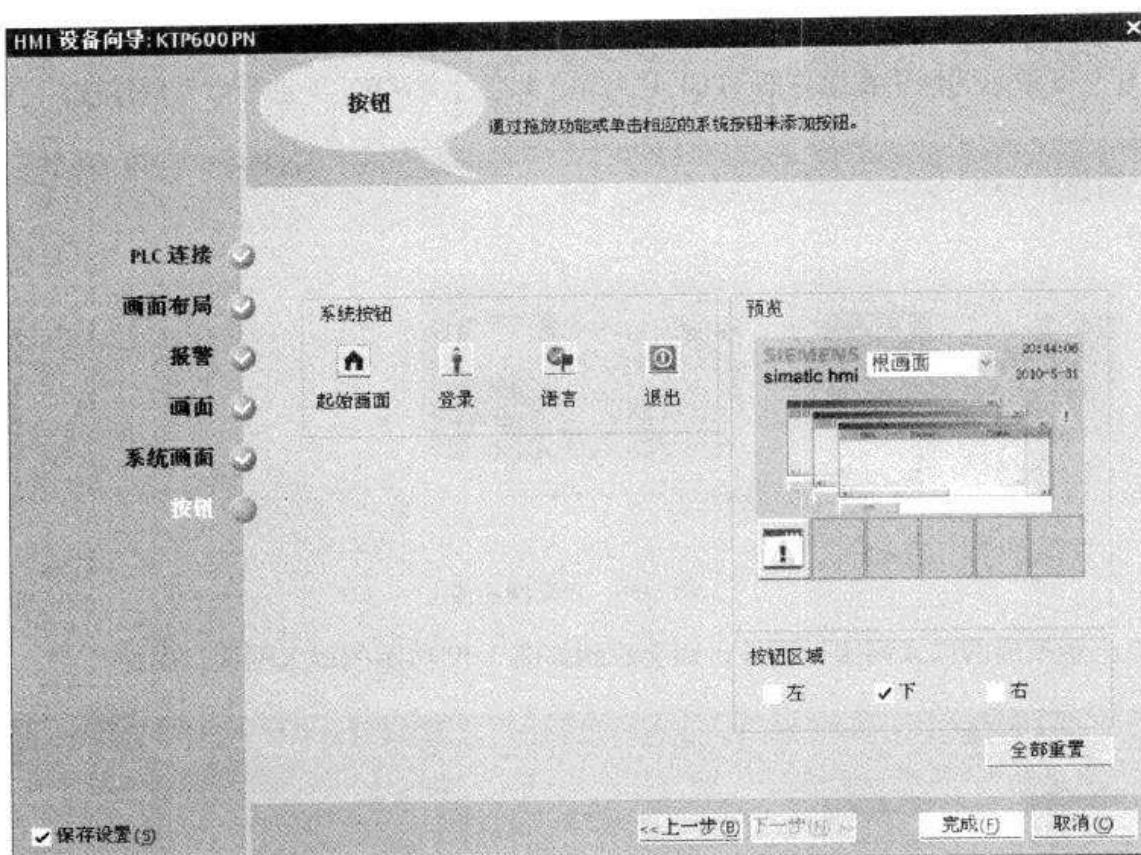


图 3-36 按钮设置画面

此时，在图 3-36 中就会在右下角出现“完成”按钮，点击“完成”，则出现如图 3-37

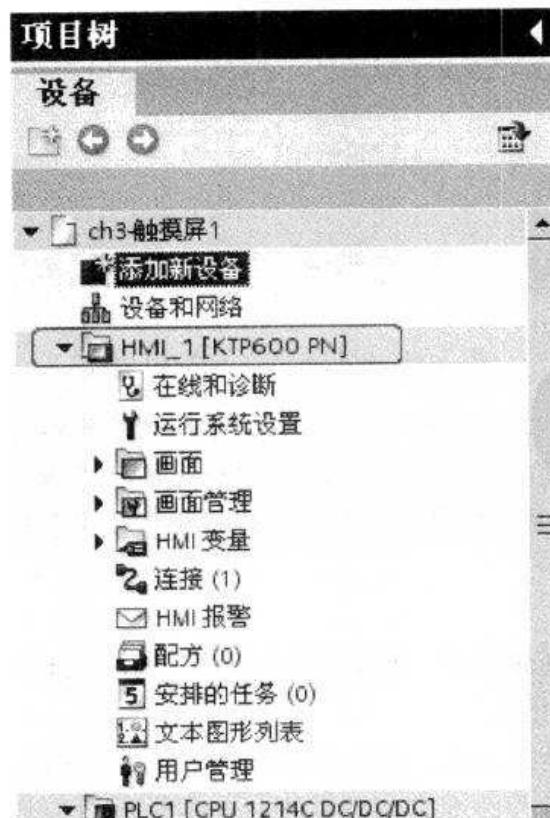


图 3-37 完成后的 HMI



图解西门子 S7-1200 PLC 入门到实践

所示的项目树中的“HMI_1 [KTP600PN]”。

图 3-38 所示为网络视图中的 PLC1 与 HMI_1 通过 PN/IE_1 进行以太网连接。

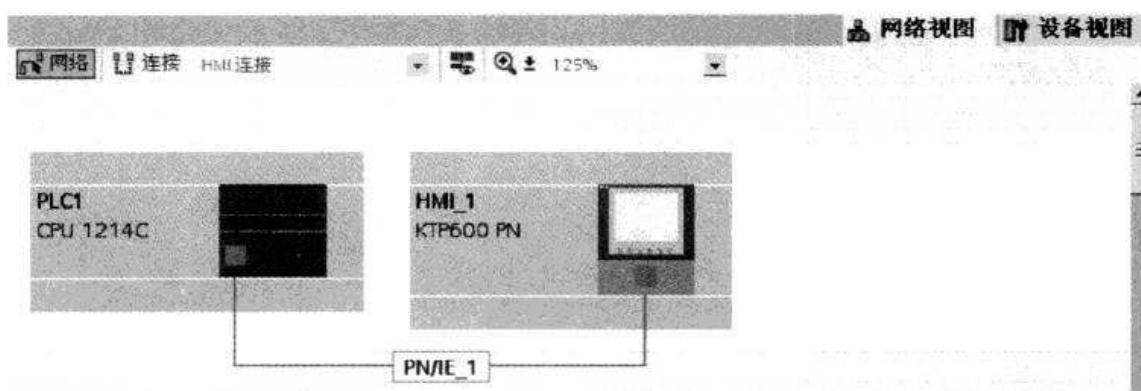


图 3-38 以太网连接

点击触摸屏的以太网接口属性，则会看到如图 3-39 所示的以太网接口属性。

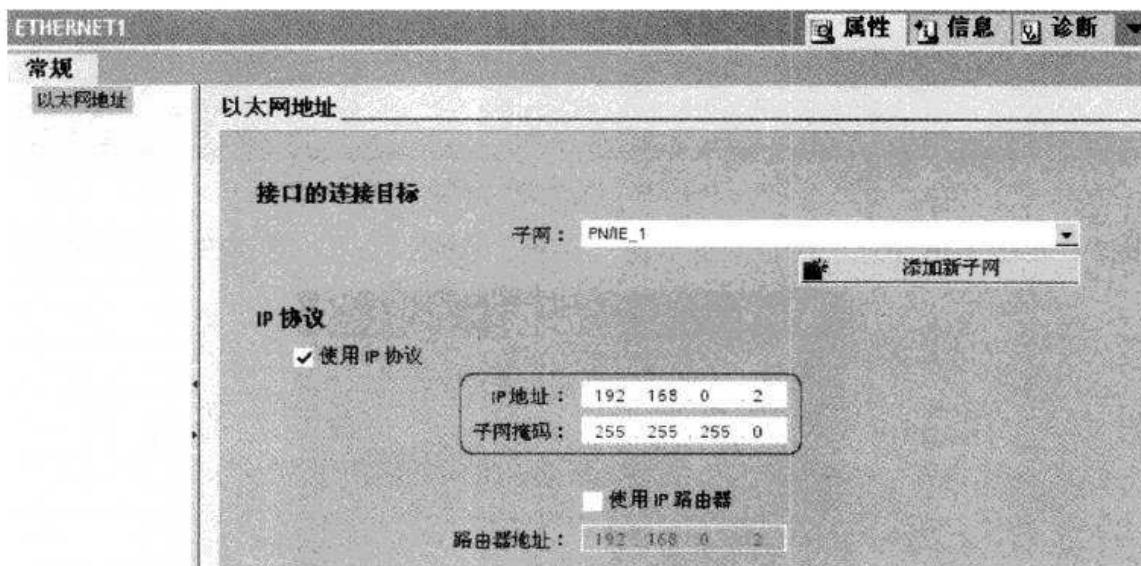


图 3-39 触摸屏的以太网接口属性

需要注意的是，触摸屏也有 MAC 地址，即介质访问控制地址，它是识别 LAN 节点的标识。网卡的物理地址通常是由网卡生产厂家烧入网卡的 EPROM（一种闪存芯片，通常可以通过程序擦写），它存储的是传输数据时真正赖以标识发出数据的电脑和接收数据的主机的地址。

3. 触摸屏的画面组态

触摸屏的组态主要是画面组态，就是将需要用其表示过程的对象插入到画面，并对该对象进行组态使之符合过程要求。

画面可以包含静态和动态元素。静态元素（例如文本或图形对象）在运行时不改变它们的状态。动态元素根据过程改变它们的状态，一般情况可以通过下列方式显示当前过程值：显示从 PLC 的存储器中输出；以字母数字、趋势图和棒图的形式显示 HMI 设备存储器



中输出的过程值；HMI 设备上的输入域也认作是动态对象。通过变量可以在 PLC 控制器和 HMI 设备之间交换过程值和操作员输入值。图 3-40 所示为工具箱（包括触摸屏常见的基本对象、元素、控制等）。

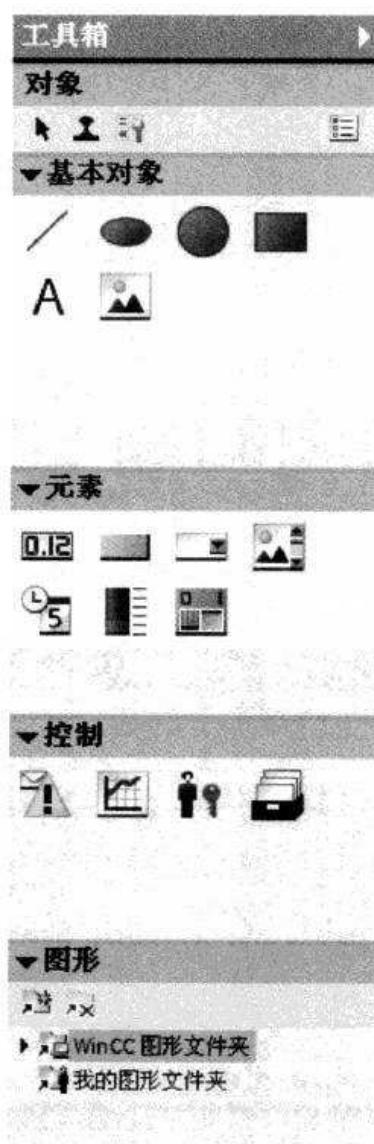


图 3-40 工具箱

(1) 添加椭圆指示灯

图 3-41 所示为在触摸屏上进行添加电动机接触器指示。

众所周知，对于触摸屏上的指示一般采用颜色变化，比如信号接通为红色，信号不接通为绿色等。如图 3-42 所示为新建指示灯椭圆“外观”动画，即会出现如图 3-43 所示的外观属性画面。

在图 3-43 所示的外观属性画面中点击 按钮，立即跳出图 3-44 所示的变量选择窗口，在这个窗口中可以选择 HMI 变量、程序块、工艺对象和 PLC 变量，如本例中选择“电机接触器% Q0.0”。

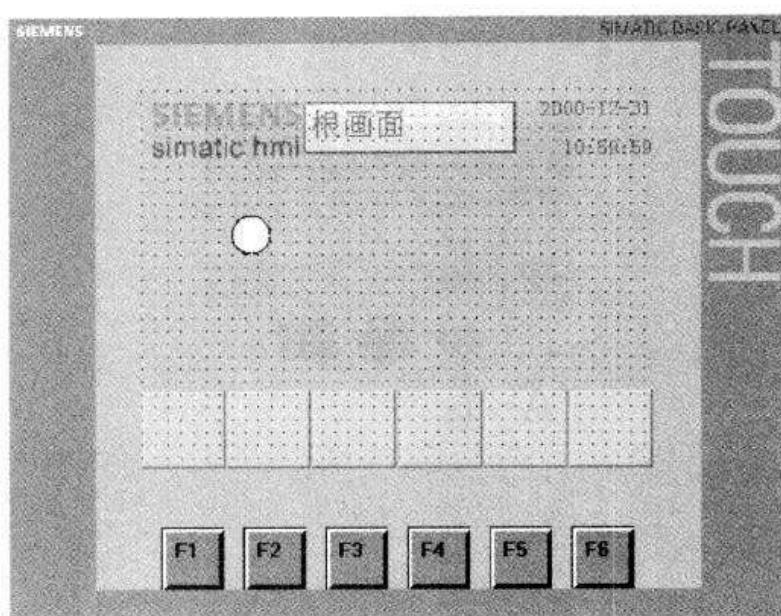


图 3-41 添加电动机接触器指示

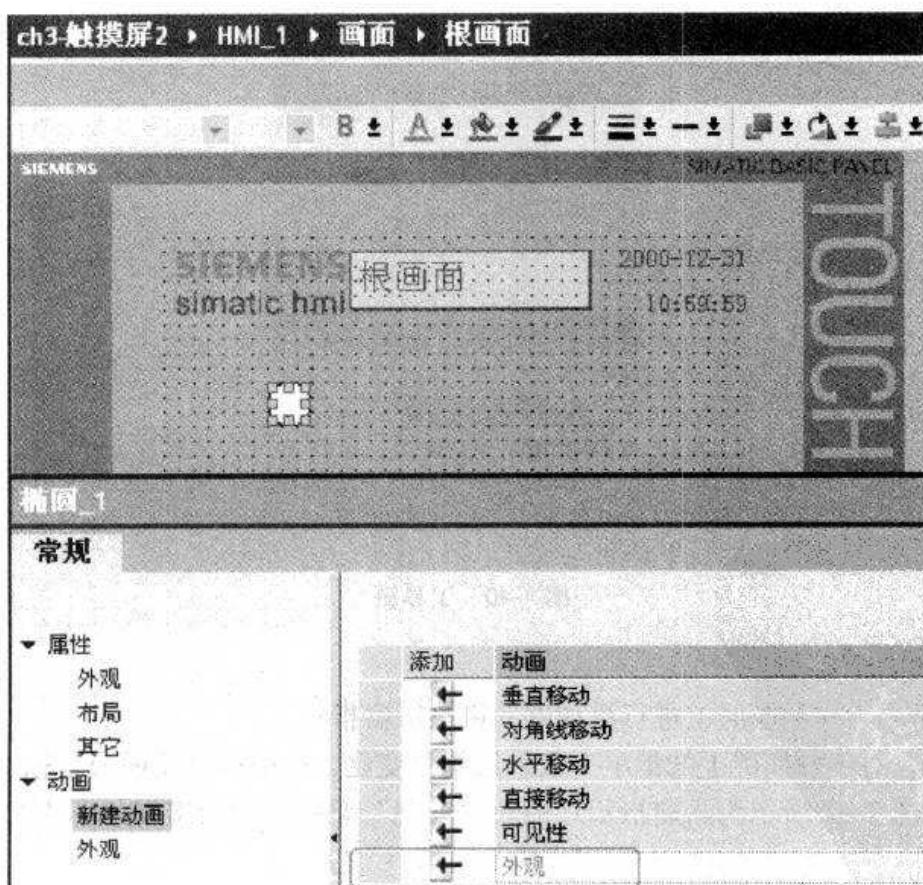


图 3-42 新建指示灯椭圆“外观”动画

第3章 KTP 触摸屏编程

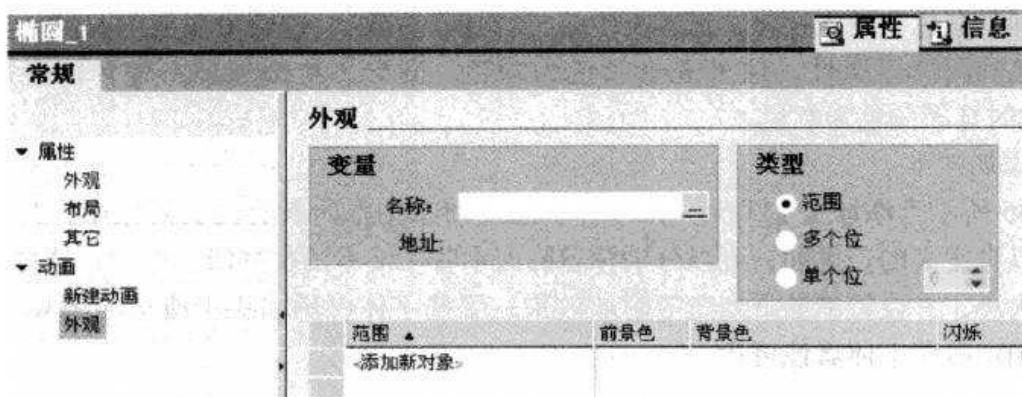


图 3-43 外观属性画面

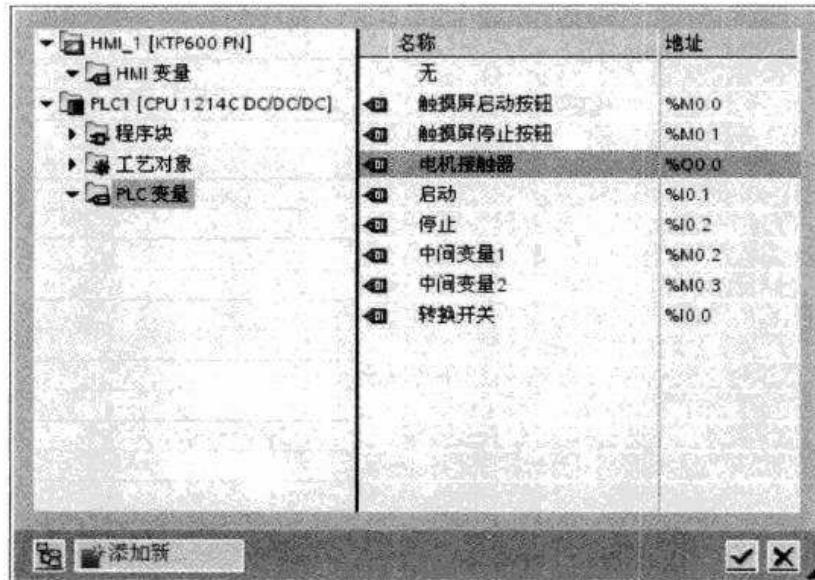


图 3-44 变量选择窗口

一旦变量选择结束，即可出现如图 3-45 所示的添加前景色和背景色画面。显然，不同

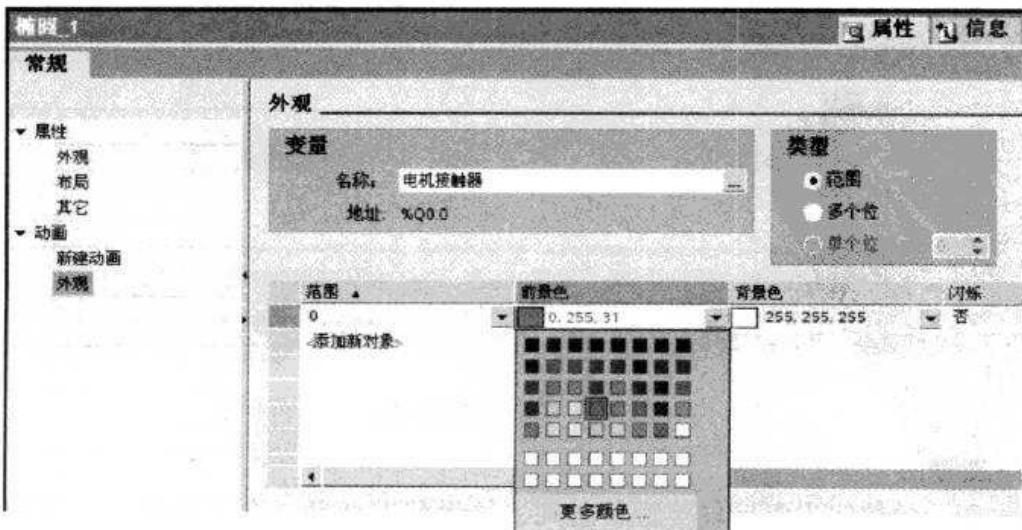


图 3-45 添加前景色和背景色画面



的变量范围其颜色会发生变化，选择图中的“添加新对象”，即会出现范围“0”，选择前景色、背景色和闪烁等属性，这里选择颜色为绿色。同样，再添加新对象，即会出现范围“1”，在此时选择颜色为红色。

(2) 添加文本

图 3-46 所示为添加文本 Text。在包含静态或动态文本的所有画面对象中，其文本的外观都是可以自定义的。例如，可以分别标识画面中各个文本的优先级。例如，选择字体、字体样式和大小，并设置诸如下划线等附加效果；哪些字体可用取决于所选的 HMI 设备；所选的字体确定哪些字体属性可用。

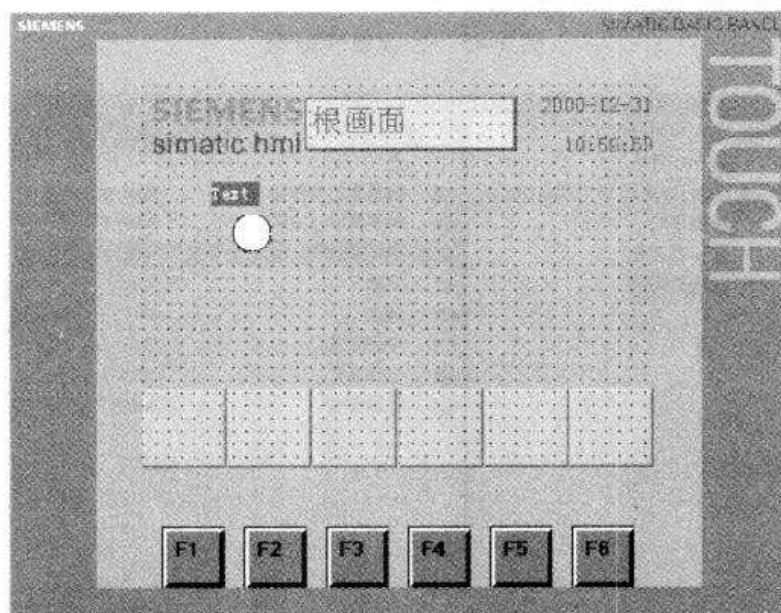


图 3-46 添加文本 Text

图 3-47 所示为文本常规属性，图 3-48 所示则是输入“电动机接触器”文本字符。

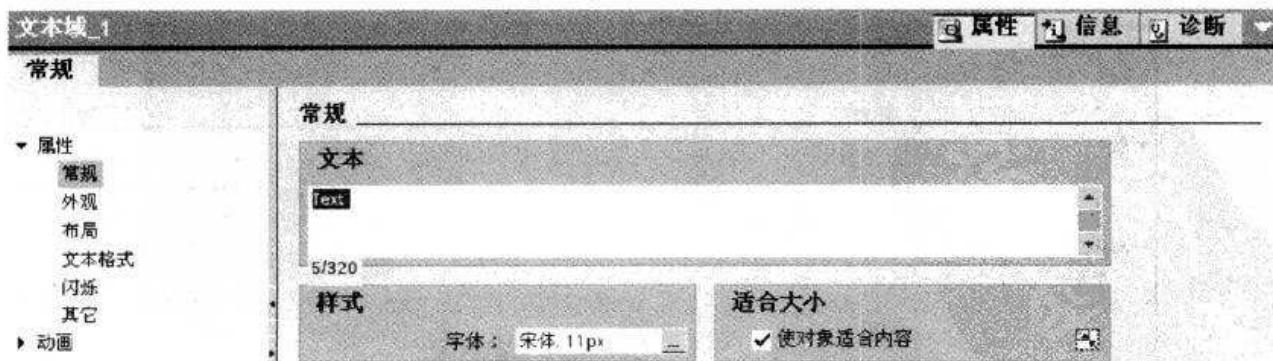


图 3-47 文本常规属性

(3) 添加按钮

图 3-49 所示的按钮主要呈现接触状态和未接触状态两种。在本案例中，选择按钮作为触摸屏对电动机的起动与停止之用。

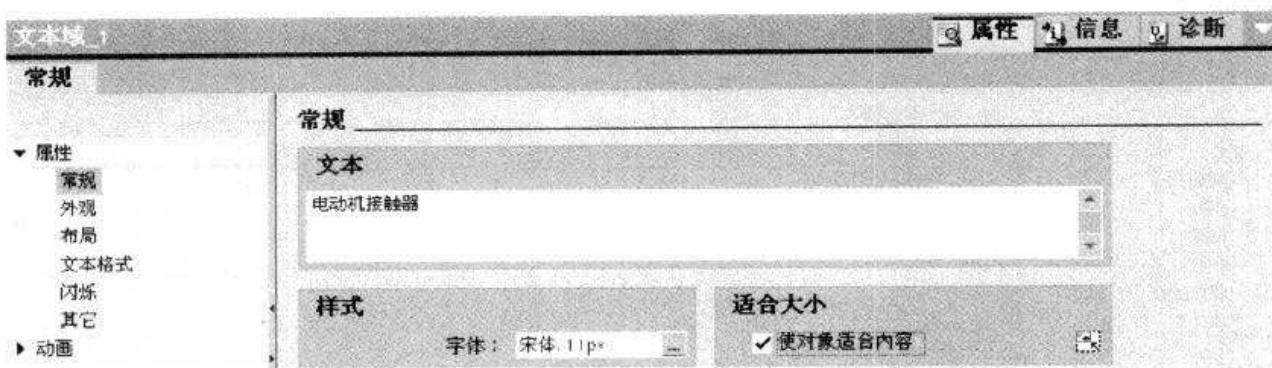


图 3-48 输入“电动机接触器”文本字符

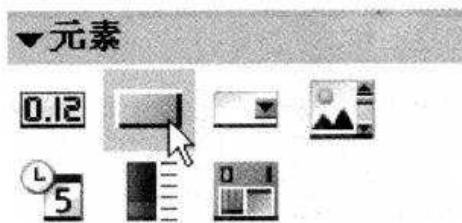


图 3-49 选择按钮

在将按钮放置到触摸屏画面中的某一个位置后，可以设置该按钮的相关属性（见图 3-50），比如文本标签，比如“开”字符，表示该按钮可以起动现场电动机。

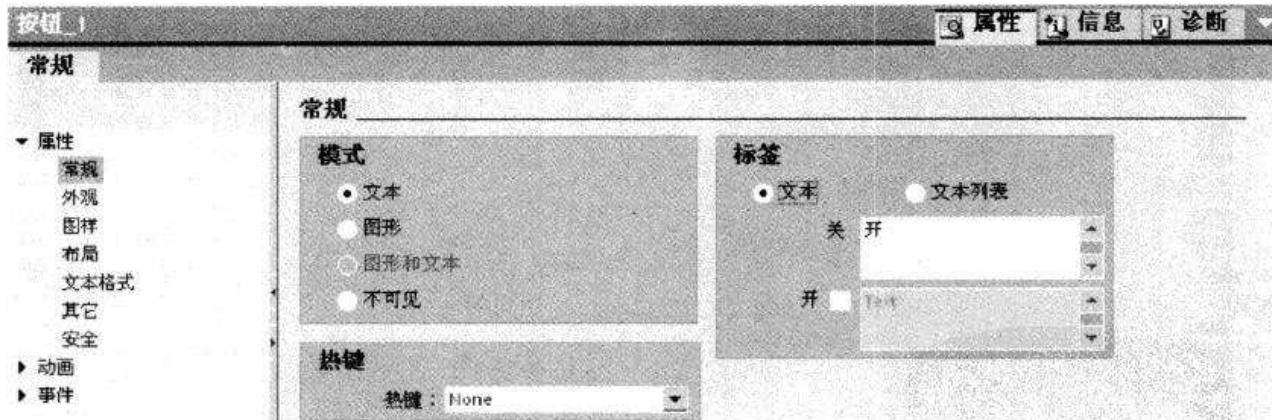


图 3-50 设置按钮的文本标签

在触摸屏中，如何模拟跟真实按钮所一致的特性呢？这就需要对该“按钮”元素进行相关“事件”定义（见图 3-51）。从图 3-51 中可以看出，按钮元素的事件包括单击、按下、释放、启用、禁用、更改，显然，前面 3 个事件跟本案例的动作比较相关。比如，我们在此定义这个按钮的属性为：当按下按钮时，将 PLC 的相关变量置位（即该变量处于 ON 状态）；当释放按钮时，将 PLC 的相关变量复位（即该变量处于 OFF 状态）。

具体操作如下：

第一步，选择“事件”下的“按下”，这时会出现右边的“添加函数”。



图解西门子 S7-1200 PLC 入门到实践

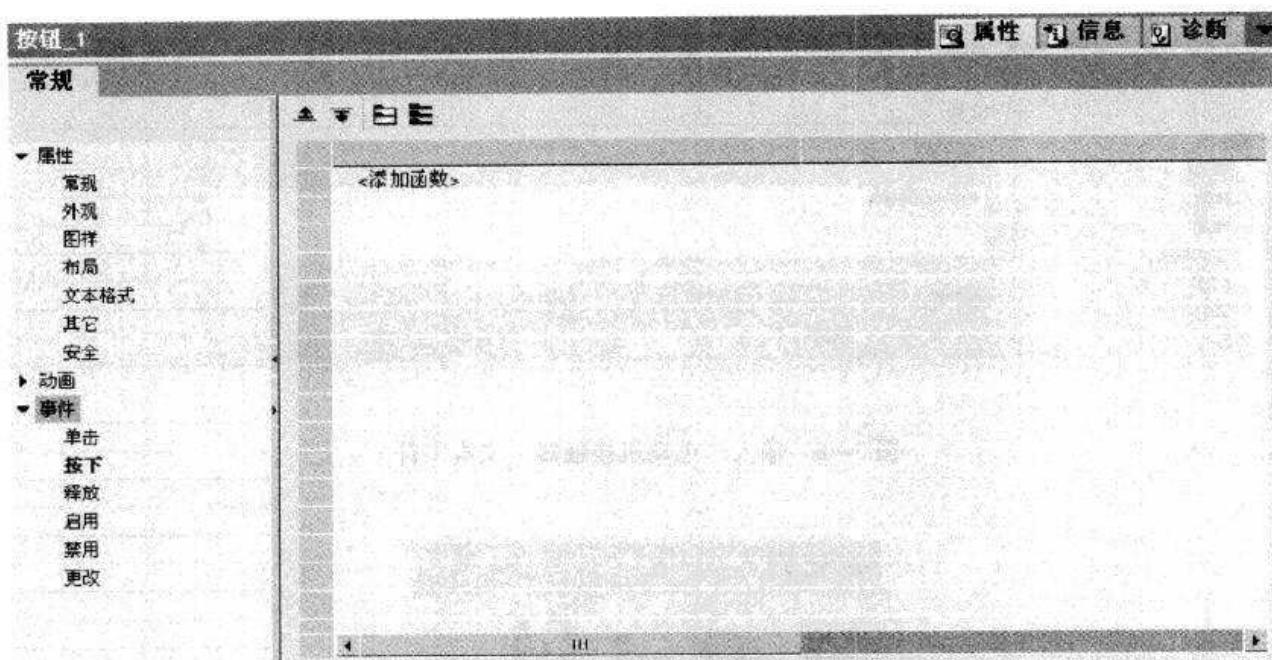


图 3-51 选择“按钮”的“事件”

第二步，添加“系统函数”，并定位“编辑位”（见图 3-52）。

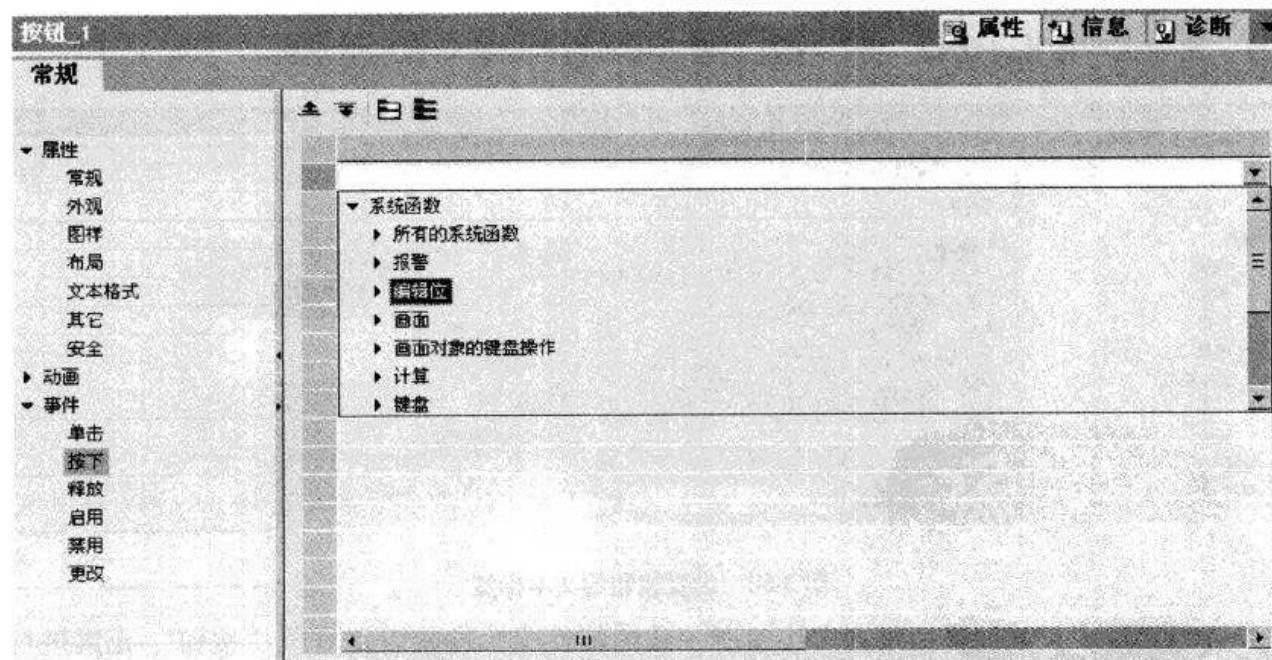


图 3-52 定位“系统函数”下的“编辑位”

第三步，选择“编辑位”下的 SetBit 属性（见图 3-53），即置位属性。

第四步，如图 3-54 所示，在按钮置位操作时，选择变量来源为“PLC 变量”中的“触摸屏启动按钮 (%M0.0)”。

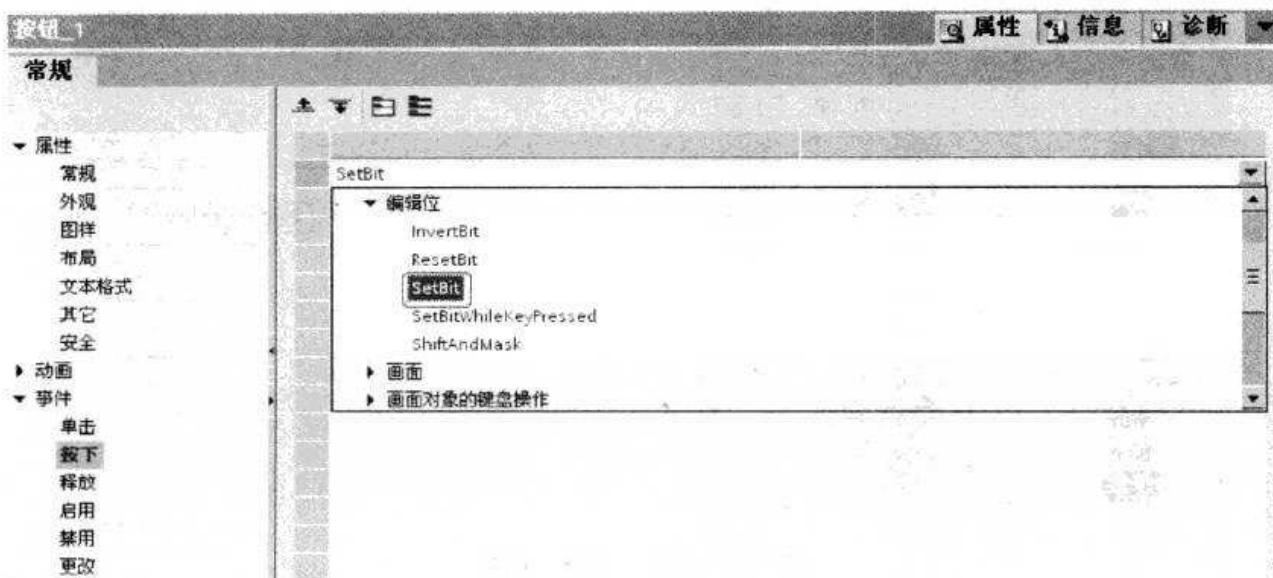


图 3-53 选择“编辑位”下的 SetBit 属性

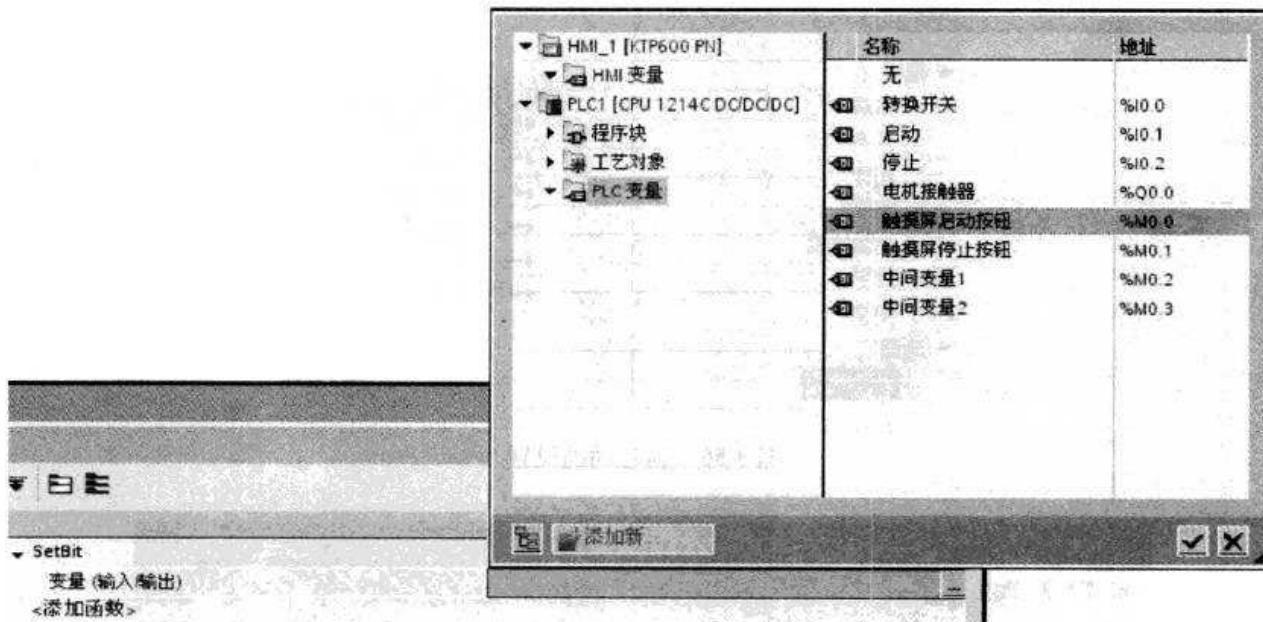


图 3-54 选择按钮的变量来源

第五步，如图 3-55 所示，在按钮“释放”操作时，选择变量来源为“PLC 变量”中的“触摸屏启动按钮（% M0.0）”，其函数为“复位（ResetBit）”。这样一来，% M0.0 就能实现按钮“瞬动”功能。

第六步，在本案例中，由于触摸屏按钮只能在转换开关% I0.0 = OFF 时才出现，因此还需要进行动画组态。图 3-56 所示为“新建动画”的可选内容，包括垂直移动、对角线移动、水平移动、直接移动、可见性和外观。

第七步，对按钮的“可见性”进行添加（见图 3-57）。



图解西门子 S7-1200 PLC 入门到实践

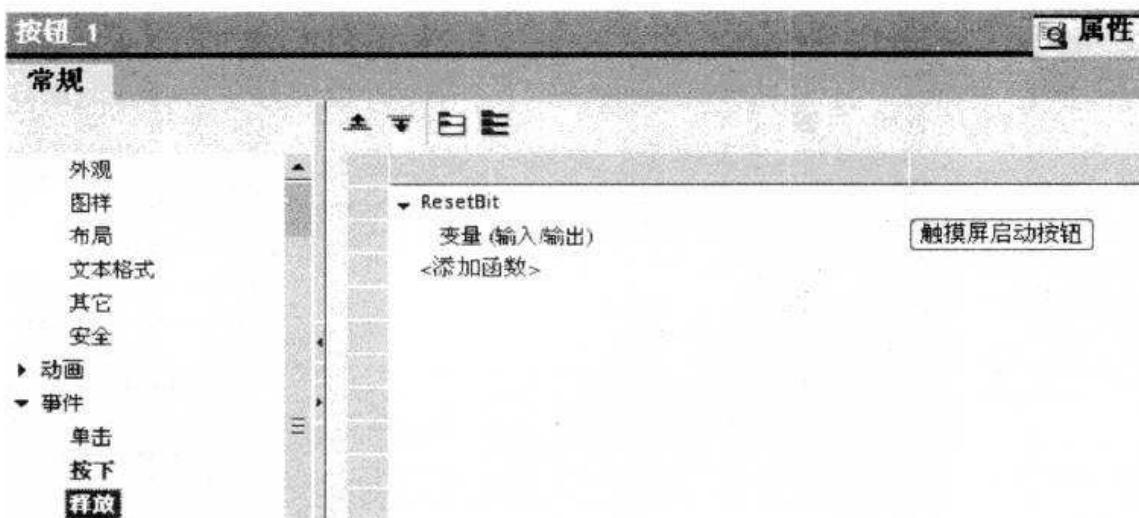


图 3-55 设置“释放”函数

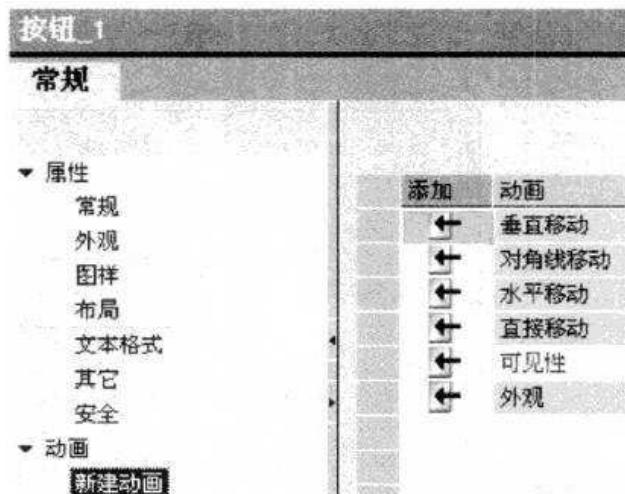


图 3-56 新建动画设置

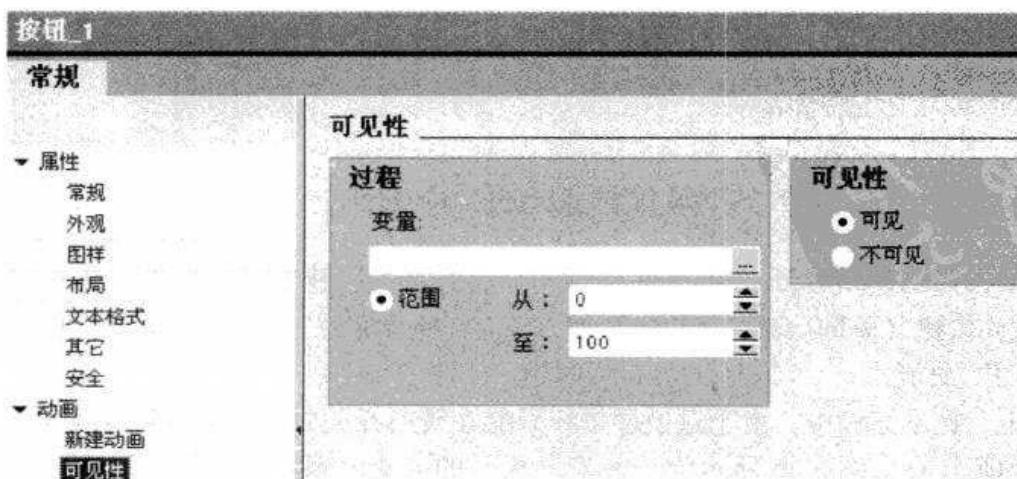


图 3-57 添加可见性动画



第八步，点击 **—** 按钮，选择变量为“PLC 变量”中的“转换开关 (%I0.0)”，即当%I0.0 = OFF 时“开”“关”两个按钮同时消失（见图 3-58）。

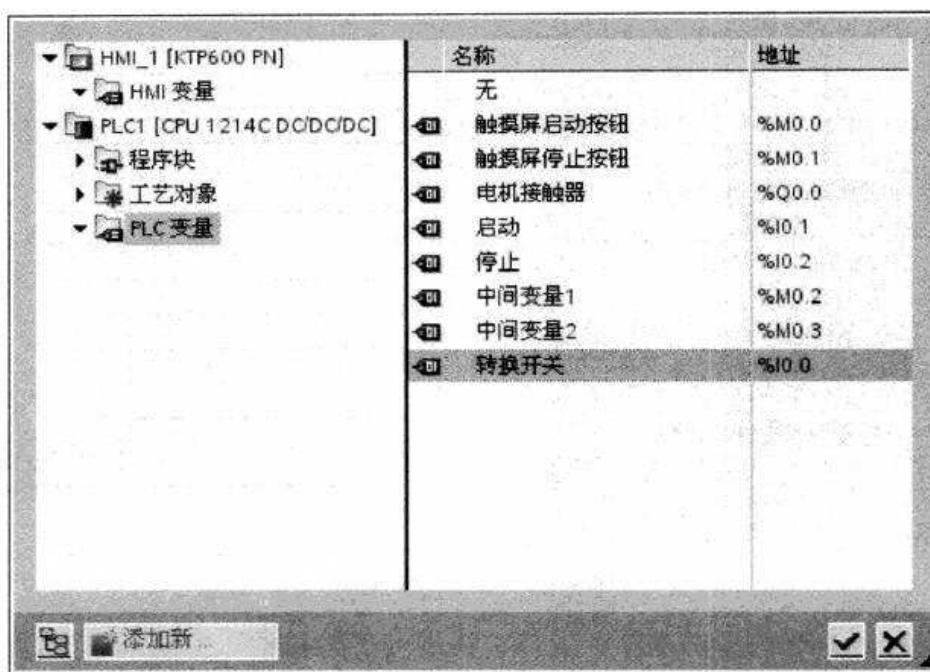


图 3-58 选择 PLC 变量中的“转换开关”

第九步，确认变量后，选择“转换开关”变量 = 1 时，即范围从“1”到“1”时为“不可见”（见图 3-59）。当然，也可以选择“转换开关”变量 = 0 时为“可见”，从此也可以看出，设置变量的方式有很多种。

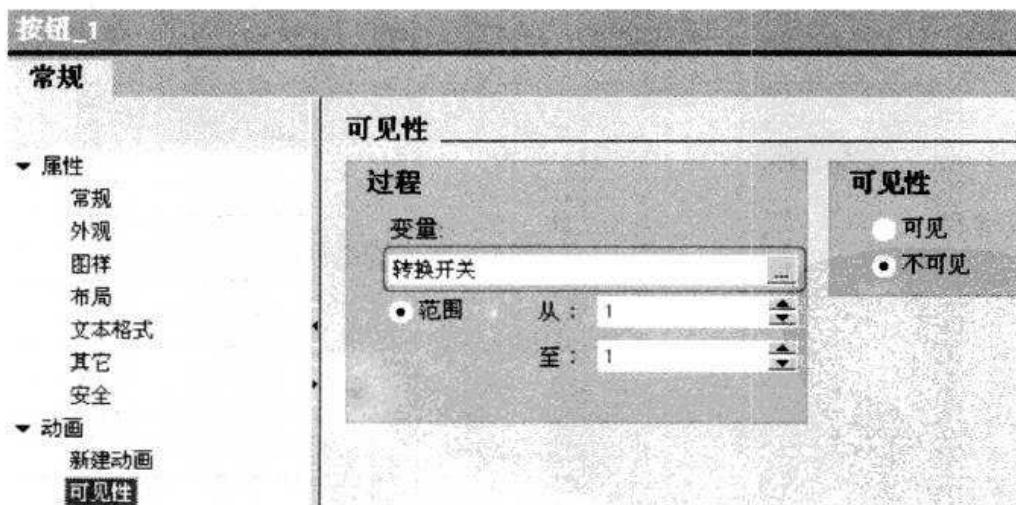


图 3-59 设置“转换开关”为“1”时按钮为不可见

4. 触摸屏的通信设置

图 3-60 所示为通信连接属性，表明 KTP600 Basic color PN 的接口为 PROFINET，地址为 192.168.0.2，其访问点为 S7ONLINE；工作站 PLC 的地址为 192.168.0.1。

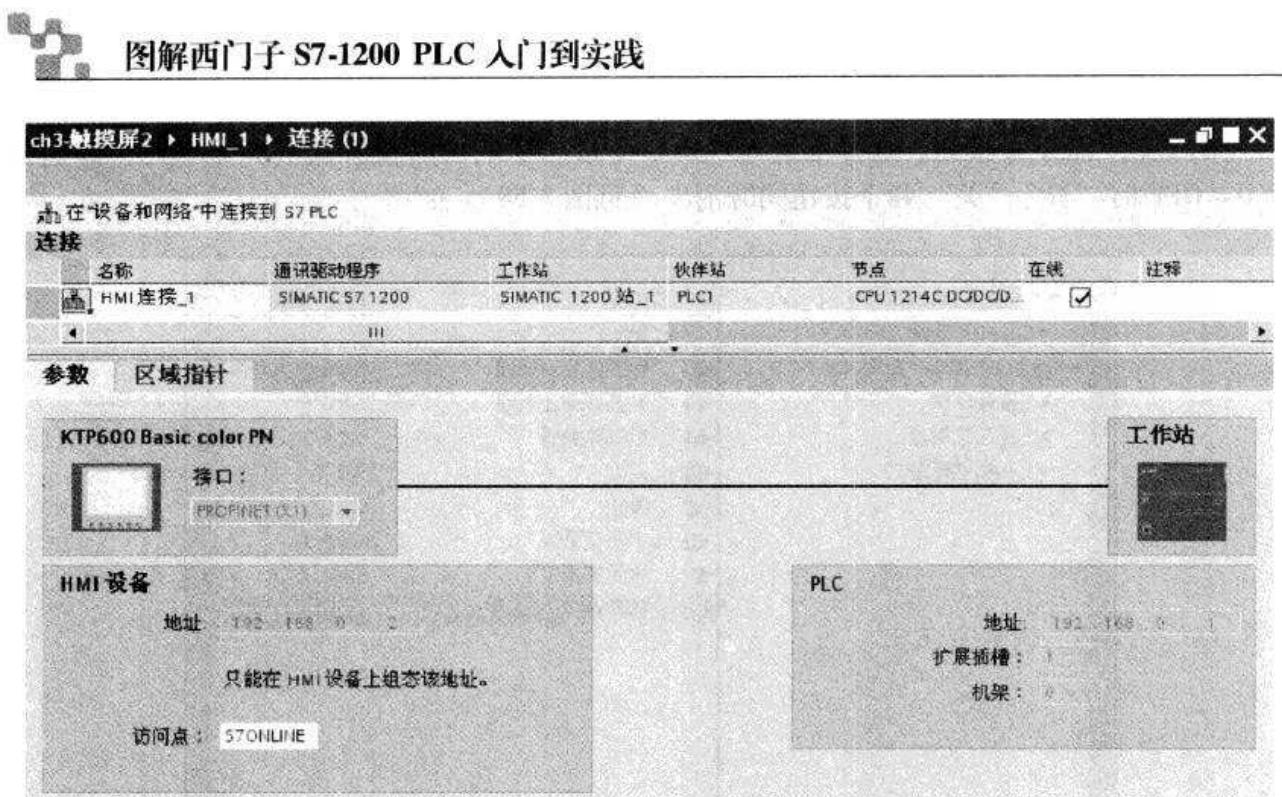


图 3-60 通信连接属性

图 3-61 所示为 HMI 变量的名称、连接、数据类型等属性，尤其是采集周期非常重要，当触摸屏上显示的变量与 PLC 之间的通信要非常快速时，比如实时显示快速变化的数据，则可以修改采集周期（见图 3-62）。

| 名称 | 连接 | 数据类型 | PLC 变量 | 地址 | 数组元素 | 采集周期 |
|---------|----------|------|---------|--------|------|------|
| 触摸屏启动按钮 | HMI 连接_1 | Bool | 触摸屏启动按钮 | <符号访问> | 1 | 1 s |
| 触摸屏停止按钮 | HMI 连接_1 | Bool | 触摸屏停止按钮 | <符号访问> | 1 | 1 s |
| 电机接触器 | HMI 连接_1 | Bool | 电机接触器 | <符号访问> | 1 | 1 s |
| 转换开关 | HMI 连接_1 | Bool | 转换开关 | <符号访问> | 1 | 1 s |

图 3-61 HMI 变量属性

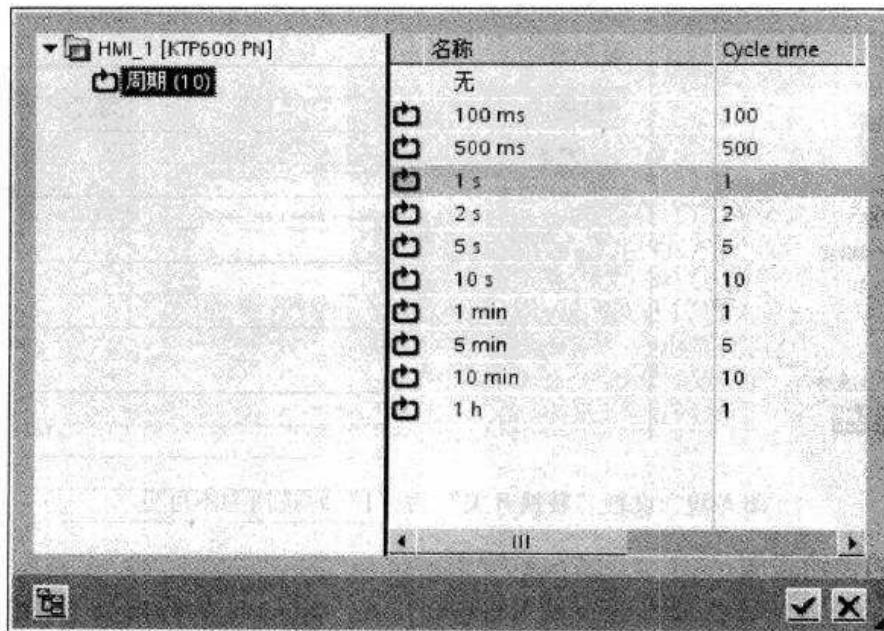


图 3-62 设置采集周期



5. 触摸屏的编译与下载

当触摸屏的画面组态结束时，需要对创建的软件进行编译。编译会生成可在相应 HMI 设备上运行的文件。如果编译期间出现错误，则 TIA 软件会提供支持来查找错误并进行更正。更正所有问题之后，将编译后的项目装载到要运行该项目的 HMI 设备中。

在启动项目的生产操作之前，使用 HMI 设备快捷菜单中的“编译”→“软件（全部重建）”命令对项目进行完全编译。如果在项目中使用连接到控制变量的 HMI 变量，则使用快捷菜单中的“编译”→“软件”命令对所有修改的块进行编译，然后再编译 HMI 设备。为了减少当前组态下编译更新内容的时间，建议时常使用“编译”→“软件（全部重建）”命令（见图 3-63）。



图 3-63 触摸屏组态的编译

如果提示编译信息正确，可以按照图 3-64 所示进行下载，即将组态软件从 PC 下载至 PLC 中去。在此过程中，将更新触摸屏所有的现有项目，如果在下载期间出现错误或警告，则相应的报警会显示在信息窗口中。成功下载项目后，用户即可在 HMI 设备上执行相关的工艺操作。

6. 触摸屏组态后的实际测试画面

根据下载后的 PLC 和触摸屏程序，可以依次实现电动机的触摸屏控制：

- 1) 当 I0.0 选择开关置于 OFF 时为触摸屏控制，此时出现的画面如图 3-65 所示；

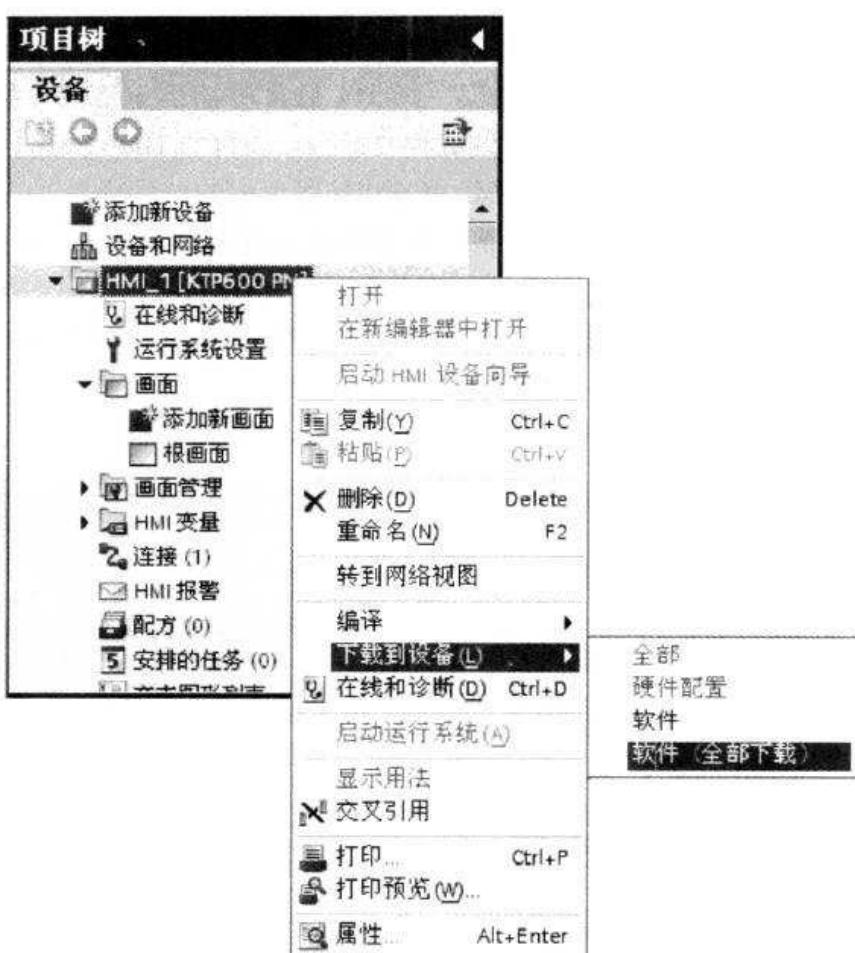


图 3-64 触摸屏组态下载

2) 在触摸屏控制时, 点击启动按钮, 触摸屏的电动机接触器指示灯为红色, 如图 3-66 所示;



图 3-65 触摸屏操作电动机停止



图 3-66 触摸屏操作电动机运行

3) 当 I0.0 选择开关置于 ON 时为现场控制, 触摸屏的按钮均消隐, 但是指示灯仍旧可以显示停止或运行状态 (见图 3-67)。

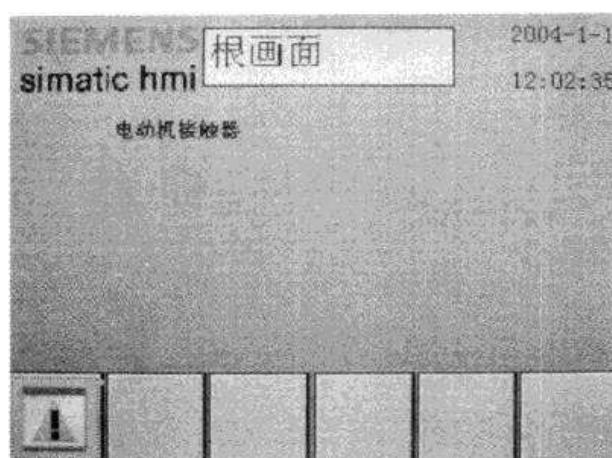


图 3-67 现场操作触摸屏按钮消隐

3.4 工程案例：全自动定时喷淋系统

3.4.1 案例介绍

在全自动豆芽生产中，开机生产豆芽时应注意喷水间隔时间控制旋钮及喷水延续时间，并在每次供水时，观察排水管口出水有无异常情况，发现有异常情况时，应打开箱门检查，及时排除故障。由此可以看出，全自动定时喷淋系统是豆芽生产的关键。

图 3-68 所示为全自动喷淋系统示意，图 3-69 所示为喷淋行车。



图 3-68 全自动喷淋系统示意

任务要求如下：1) S7-1200 PLC 连接现场的行车左右限位和热继电器信号等输入信号，同时连接水泵、行车左行和右行、故障指示灯、运行指示灯；2) 在触摸屏可以进行全自动



图 3-69 喷淋行车

喷淋系统的启动与停止，进行各个输入输出状态显示，同时能在触摸屏上进行一天内 6 次喷淋时间的设置；3) 设置的喷淋时间一旦到达，则延时 5s 后开启喷淋水泵，再延时 5s 行车左行，待左限位触发时，行车右行，待行车右限位触发时，行车左行离开限位一段时间后停止，这就是一次喷淋过程；4) 能在触摸屏上显示和设置当前时间。

3.4.2 时钟和日历指令

日期和时间指令可以用于设计日历和时间计算，TIME 数据类型与 DTL 数据结构的比较见表 3-2：

- 1) T_CONV 用于转换时间值的数据类型：(Time 转换为 DInt) 或 (DInt 转换为 Time)；
- 2) T_ADD 用于将 Time 与 DTL 值相加：(Time + Time = Time) 或 (DTL + Time = DTL)；
- 3) T_SUB 用于将 Time 与 DTL 值相减：(Time - Time = Time) 或 (DTL - Time = DTL)；
- 4) T_DIFF 提供两个 DTL 值的差作为 Time 值：DTL - DTL = Time。

表 3-2 TIME 数据类型与 DTL 数据结构的比较

| 数据类型 | 大小(位) | 有效范围 |
|-----------------|---------|---|
| Time | 32 存储形式 | T#-24d_20h_31m_23s_648ms ~ T#24d_20h_31m_23s_647ms -2,147,483,648ms ~ +2,147,483,647ms |
| DTL 数据结构 | | |
| 年：UInt | 16 | 1970 ~ 2554 |
| 月：USInt | 8 | 1 ~ 12 |
| 日：USInt | 8 | 1 ~ 31 |
| 工作日：USInt | 8 | 1 = 周日到 7 = 周六 |
| 小时：USInt | 8 | 0 ~ 23 |
| 分钟：USInt | 8 | 0 ~ 59 |
| 秒：USInt | 8 | 0 ~ 59 |
| 纳秒：UDInt | 32 | 0 ~ 999,999,999 |



时钟指令用于设置和读取 PLC 系统时钟，使用数据类型 DTL 提供日期和时间值。需要注意的是，S7-1200 PLC 存储的是世界标准时间，即格林威治时间。

1) WR_SYS_T (写入系统时间) 使用参数 IN 中的 DTL 值设置 PLC 日时钟，如图 3-70 所示。该时间值不包括本地时区或夏令时偏移量，即为世界标准时间。

2) RD_SYS_T (读取系统时间) 从 PLC 读取当前系统时间，如图 3-71 所示，为世界标准时间。

3) RD_LOC_T (读取本地时间) 以 DTL 数据类型提供 PLC 的当前本地时间，如图 3-72 所示。

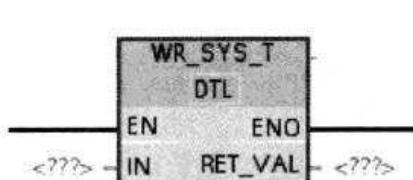


图 3-70 WR_SYS_T 指令

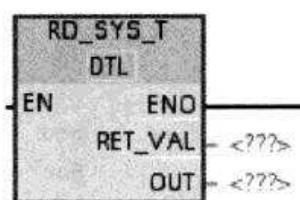


图 3-71 RD_SYS_T 指令

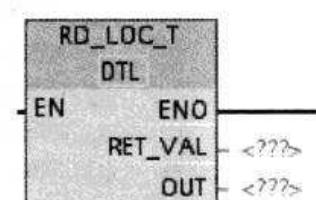


图 3-72 RD_LOC_T 指令

通过使用用户在 CPU 时钟设备配置中设置的时区和夏令时偏移量计算本地时间。其时区组态是相对于协调世界时系统时间的偏移量。夏令时组态指定夏令时开始时的月份、星期、日期和小时。标准时间组态也会指定标准时间开始时的月份、星期、日期和小时。时区偏移量始终会应用到系统时间值。只有在夏令时有效时才会应用夏令时偏移量。

关于上述 3 个指令的 RET_VAL 值的说明见表 3-3。

表 3-3 RET_VAL 值的说明

| RET_VAL(W#16#...) | 说 明 | RET_VAL(W#16#...) | 说 明 |
|-------------------|---------|-------------------|-----------|
| 0000 | 无错误 | 8084 | 非法小时值 |
| 8080 | 本地时间不可用 | 8085 | 非法分钟值 |
| 8081 | 非法年份值 | 8086 | 非法秒数值 |
| 8082 | 非法月份值 | 8087 | 非法纳秒值 |
| 8083 | 非法日期值 | 80B0 | 实时时钟发生了故障 |

3.4.3 变量定义与软件编程

1. 变量定义

根据工程要求，对自动喷淋系统的输入、输出、中间变量和其他相关数据进行定义，具体的变量表格见表 3-4。其中 %MB110 ~ %MB121 为触摸屏进行设置的 6 个自动喷淋时间，数据类型为 Byte。



表 3-4 变量表格

| | 名称 | 数据类型 | 地址 |
|----|-----------|-------|--------|
| 1 | 左限位 | Bool | %I0.0 |
| 2 | 右限位 | Bool | %I0.1 |
| 3 | 热继电器故障 | Bool | %I0.2 |
| 4 | 水泵运行 | Bool | %Q0.0 |
| 5 | 行车左行 | Bool | %Q0.1 |
| 6 | 行车右行 | Bool | %Q0.2 |
| 7 | 故障指示灯 | Bool | %Q0.3 |
| 8 | 运行指示灯 | Bool | %Q0.4 |
| 9 | 读取时间状态值 | Int | %MW0 |
| 10 | 修改时间状态值 | Int | %MW2 |
| 11 | 行车状态值 | Word | %MW4 |
| 12 | 时间修改按钮 | Bool | %M10.0 |
| 13 | 时间修改上升沿 | Bool | %M10.1 |
| 14 | 定时到状态 | Bool | %M10.2 |
| 15 | 行车左行上升沿 | Bool | %M10.4 |
| 16 | 左限位上升沿 | Bool | %M10.5 |
| 17 | 右限位上升沿 | Bool | %M10.6 |
| 18 | 定时到上升沿 | Bool | %M10.7 |
| 19 | 报警字 | Word | %MW12 |
| 20 | 热继电器报警位_1 | Bool | %M12.0 |
| 21 | 定时器值1 | Time | %MD20 |
| 22 | 定时器值2 | Time | %MD24 |
| 23 | 修改后年份 | Word | %MW100 |
| 24 | 修改后月份 | USInt | %MB102 |
| 25 | 修改后日期 | USInt | %MB103 |
| 26 | 修改后小时 | USInt | %MB104 |
| 27 | 修改后分钟 | USInt | %MB105 |
| 28 | 修改后秒数 | USInt | %MB106 |
| 29 | 设定小时1 | Byte | %MB110 |
| 30 | 设定分钟1 | Byte | %MB111 |
| 31 | 设定小时2 | Byte | %MB112 |
| 32 | 设定分钟2 | Byte | %MB113 |
| 33 | 设定小时3 | Byte | %MB114 |
| 34 | 设定分钟3 | Byte | %MB115 |
| 35 | 设定小时4 | Byte | %MB116 |
| 36 | 设定分钟4 | Byte | %MB117 |
| 37 | 设定小时5 | Byte | %MB118 |
| 38 | 设定分钟5 | Byte | %MB119 |
| 39 | 设定小时6 | Byte | %MB120 |
| 40 | 设定分钟6 | Byte | %MB121 |

除了以上变量外，还定义了全局数据块 DB1，其包含的数据结构见表 3-5。从表中可以看出，共包括 time_set1（即时间设置）、Time_now（即现在时间），数据类型为 DTL 类型。



表 3-5 DB1 的数据结构

| | 名称 | 数据类型 | 偏移量 | 初始值 |
|----|-------------|-------|------|-------------------|
| 1 | ▼ Static | | | |
| 2 | ▼ time_set1 | DTL | 0.0 | DTL#1970-1-1-0... |
| 3 | YEAR | UInt | 0.0 | 1970 |
| 4 | MONTH | USInt | 2.0 | 1 |
| 5 | DAY | USInt | 3.0 | 1 |
| 6 | WEEKDAY | USInt | 4.0 | 5 |
| 7 | HOUR | USInt | 5.0 | 0 |
| 8 | MINUTE | USInt | 6.0 | 0 |
| 9 | SECOND | USInt | 7.0 | 0 |
| 10 | NANOSECOND | UDInt | 8.0 | 0 |
| 11 | ▼ Time_now | DTL | 12.0 | DTL#1970-1-1-0... |
| 12 | YEAR | UInt | 0.0 | 1970 |
| 13 | MONTH | USInt | 2.0 | 1 |
| 14 | DAY | USInt | 3.0 | 1 |
| 15 | WEEKDAY | USInt | 4.0 | 5 |
| 16 | HOUR | USInt | 5.0 | 0 |
| 17 | MINUTE | USInt | 6.0 | 0 |
| 18 | SECOND | USInt | 7.0 | 0 |
| 19 | NANOSECOND | UDInt | 8.0 | 0 |

2. PLC 编程

全自动喷淋的 PLC 程序块结构如图 3-73 所示，共包括上电初始化 OB100、主程序 OB1、全局数据块 DB1 和定时器背景数据块 DB2、DB3。



图 3-73 PLC 程序块结构

(1) 上电初始化 OB (OB100)

图 3-74 所示为上电初始化 OB100，它主要进行数据的复位，包括将“定时到状态”变量复位、“行车状态值”设为 4、故障报警字设为 0。

(2) 主程序 OB (OB1)

图 3-75 所示为主程序 OB1，它主要进行时钟的读写、自动喷淋的时间控制、故障报警等。



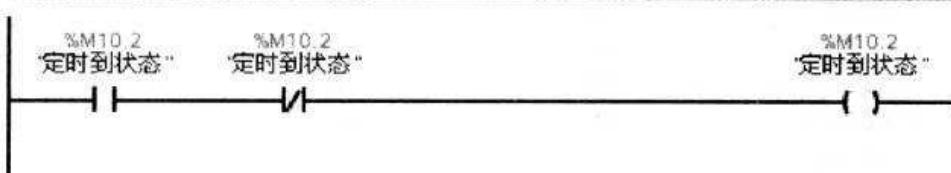
图解西门子 S7-1200 PLC 入门到实践

▼ 块标题：

上电初始化OB

▼ 程序段 1：

上电初始化时,将“定时到状态”变量复位.



▼ 程序段 2：

将“行车状态值”设置为“4”



▼ 程序段 3：

将“故障报警字”设置为“0”



图 3-74 上电初始化 OB100

▼ 块标题：

自动喷淋时间设置

▼ 程序段 1：

读取系统当地时间

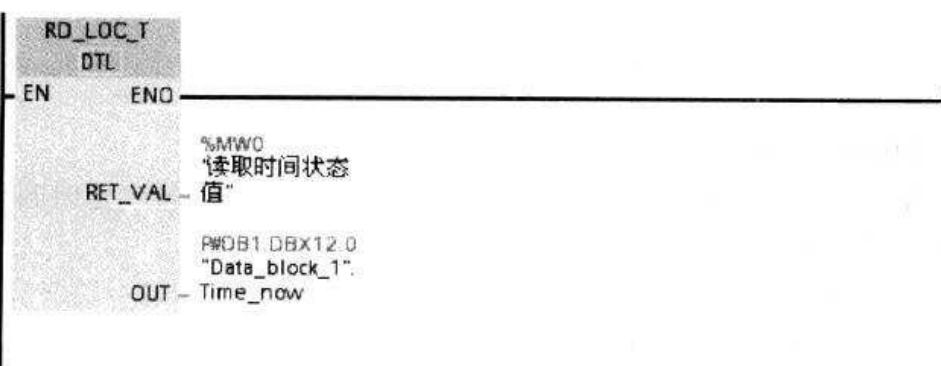
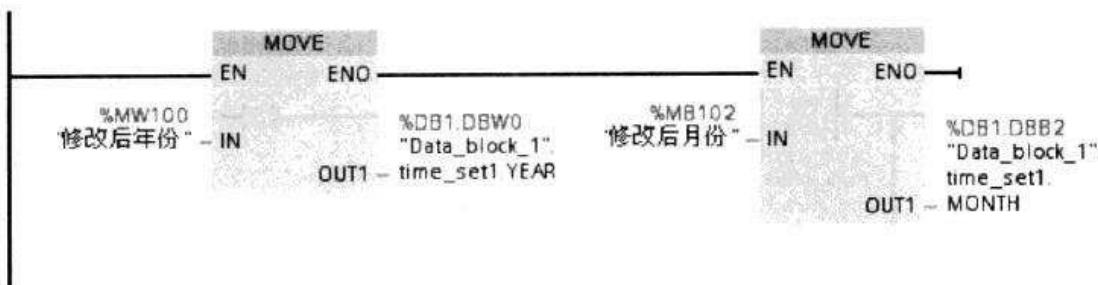


图 3-75 主程序 OB1

第3章 KTP 触摸屏编程

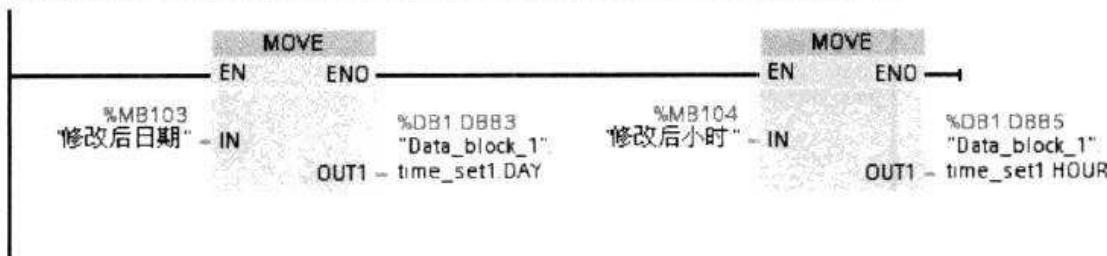
▼ 程序段 2：

从触摸屏读取修改后年份、月份



▼ 程序段 3：

从触摸屏读取修改后日期、小时



▼ 程序段 4：

从触摸屏读取修改后分钟、秒



▼ 程序段 5：

修改时间（注意：修改的是世界标准时间）

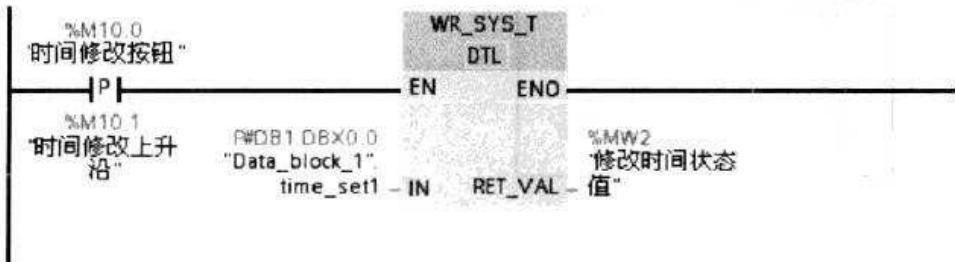


图 3-75 主程序 OB1 (续一)



▼ 程序段 6：

判断“当定时到状态”未激活时，是否要到达设定的 6 个定时值

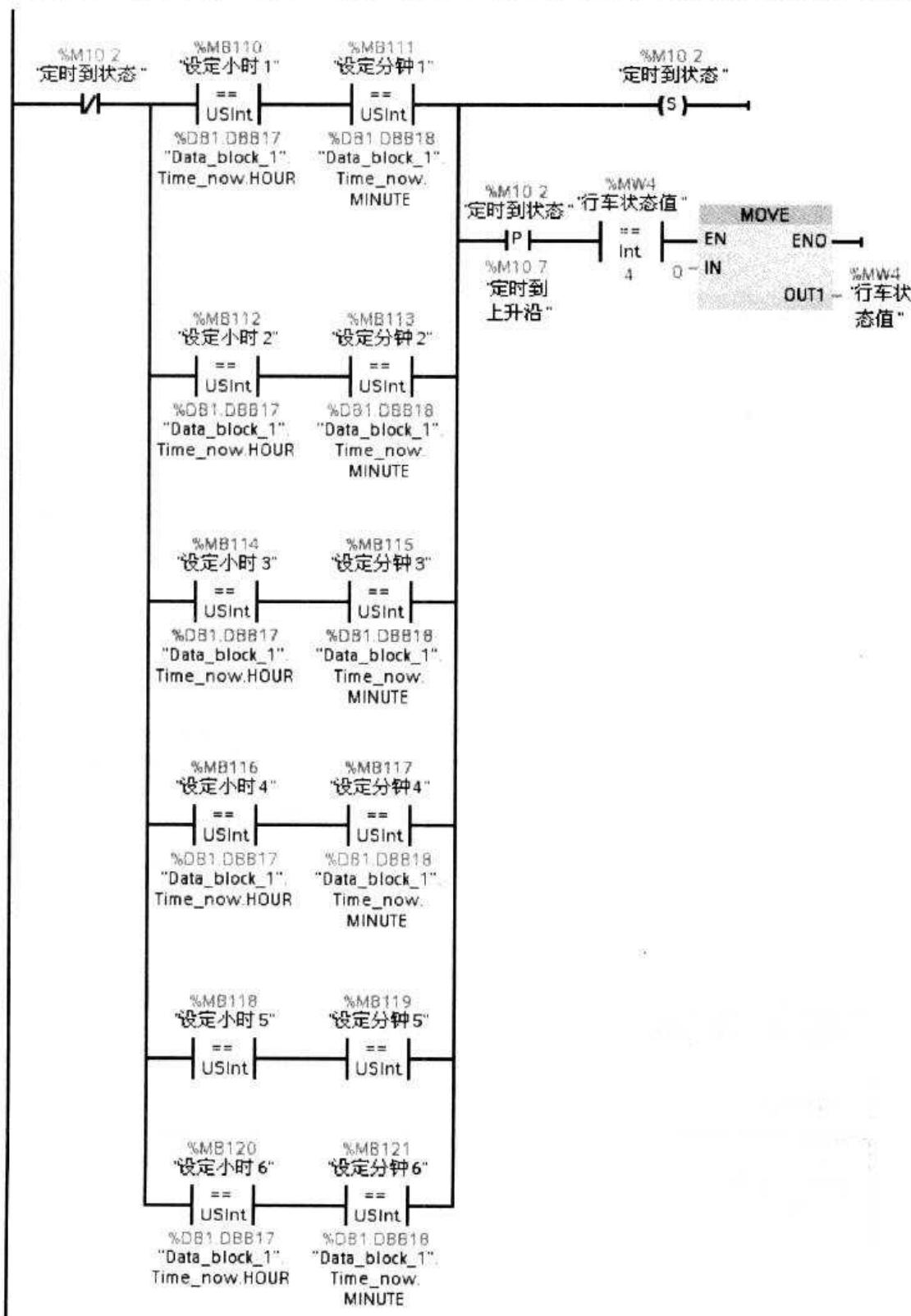
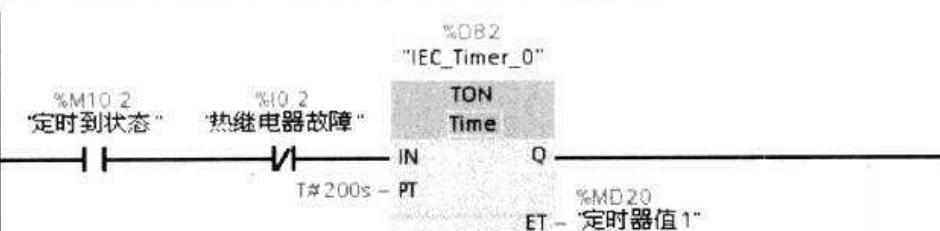


图 3-75 主程序 OB1（续二）



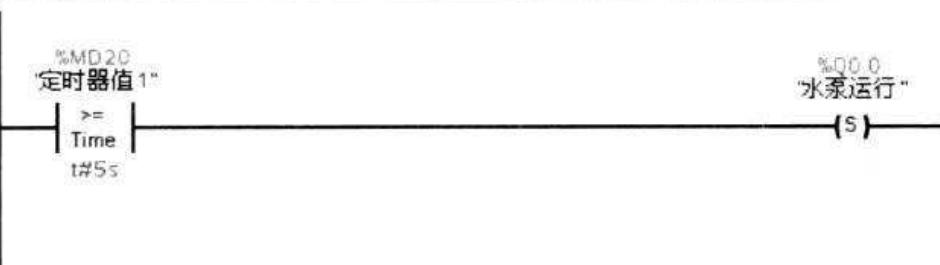
程序段 7：

定时到状态时，且未热继电器故障，开始进行计时



程序段 8：

计时5S后，开始水泵运行，并置行车状态位为0



程序段 9：

步序控制

- (1) 计时10S 后，行车状态位为0时，行车左行；(2) 行车左行，其状态位为1；(3) 当行车左限位到时，行车状态位为2，行车右行；(4) 行车右限位到达时，行车状态位为3；(5) 由于行车不能停留在右限位位置，需要向左再行走3S，其状态位变为4，整个过程结束。

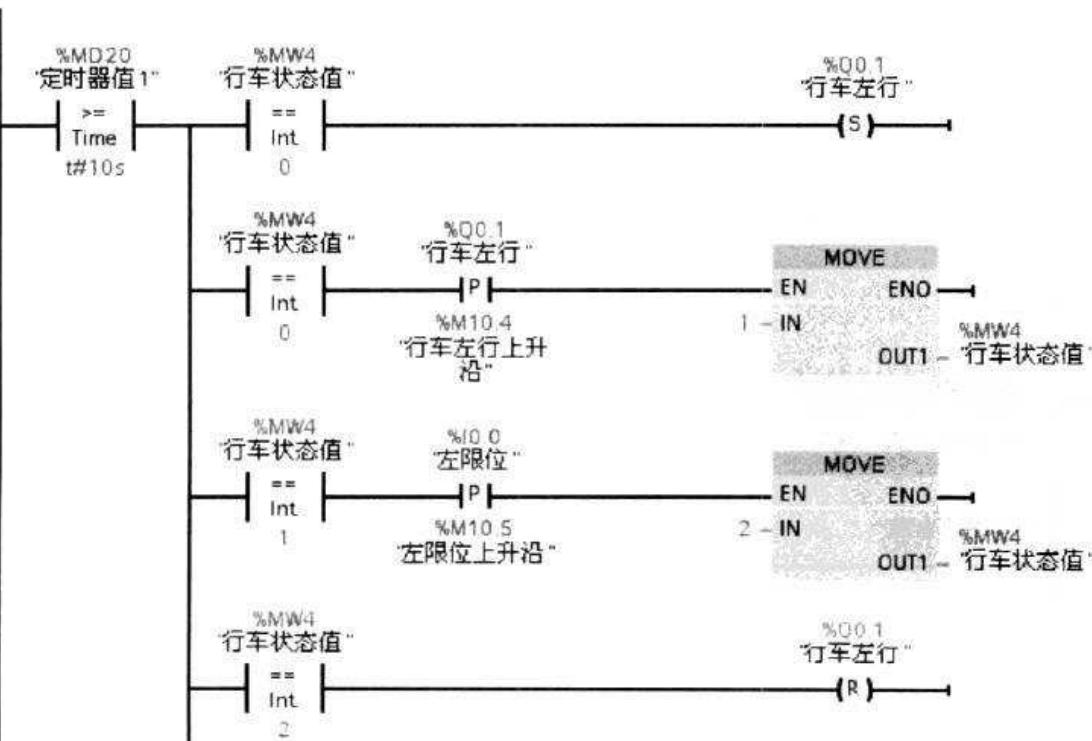


图 3-75 主程序 OB1 (续三)



图解西门子 S7-1200 PLC 入门到实践

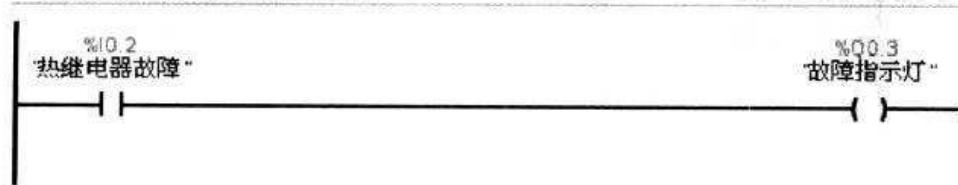
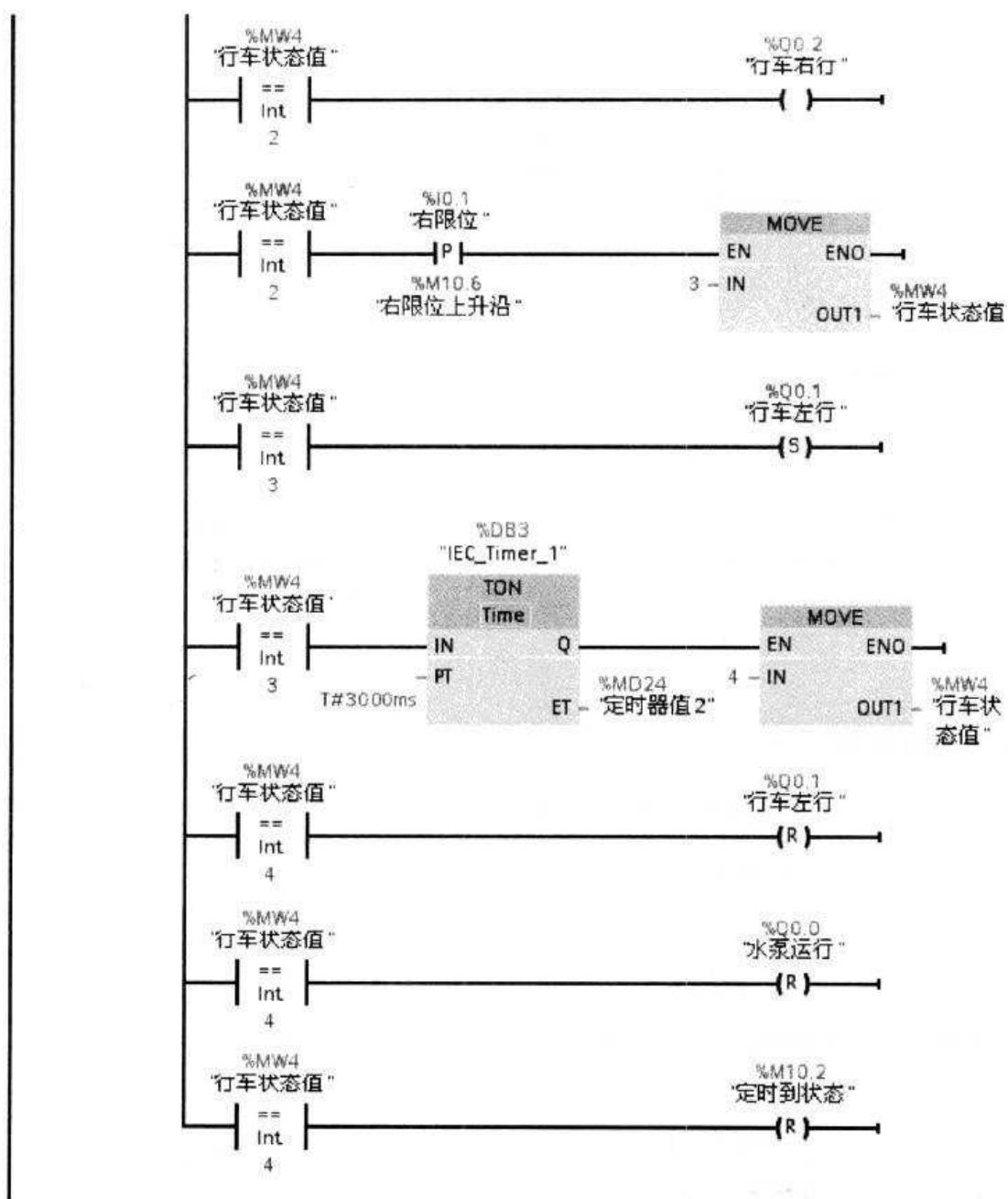


图 3-75 主程序 OB1 (续四)



程序段 11：

热继电器故障激活HMI报警位（即MW12的第8位）



图 3-75 主程序 OB1 (续五)

3. 触摸屏的画面管理与报警组态

(1) 建立不同的触摸屏画面

根据触摸屏的画面向导建立如图 3-76 所示的画面，它包括根画面、PLC 时间等。

在本案例中，最重要的两个画面为主画面（见图 3-77）、时间设置画面（见图 3-78）。在主画面中，可以实时显示当前的时间，显示泵和行车的运行状态，也可以对每天的 6 次喷淋时间进行设定。一般喷淋时间只需要“时”和“分”两个单位即可。

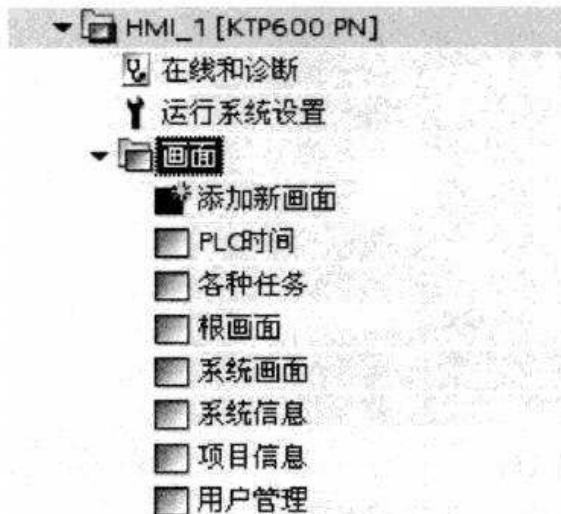


图 3-76 触摸屏画面

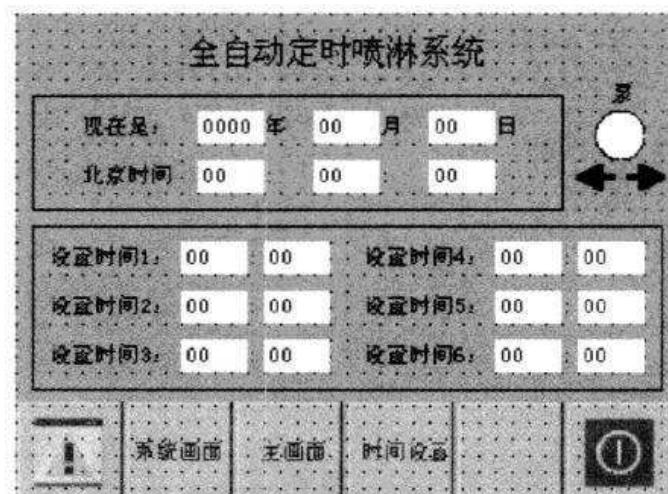


图 3-77 主画面



图 3-78 时间设置画面



图解西门子 S7-1200 PLC 入门到实践

(2) 触摸屏画面的运行系统设置

如图 3-79 所示为触摸屏画面的运行系统设置。

运行系统的设置最重要的是画面设置。它主要可以设置起始画面，如本案例中选择“根画面”，这是为了确保触摸屏在重新上电后出现的首页是用户工艺所需要的（见图 3-80）。通过点击 就会跳出图 3-81 所示的画面，这样用户就可以自由选择了。

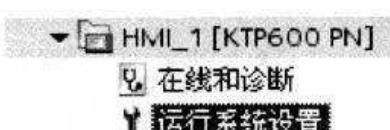


图 3-79 运行系统设置

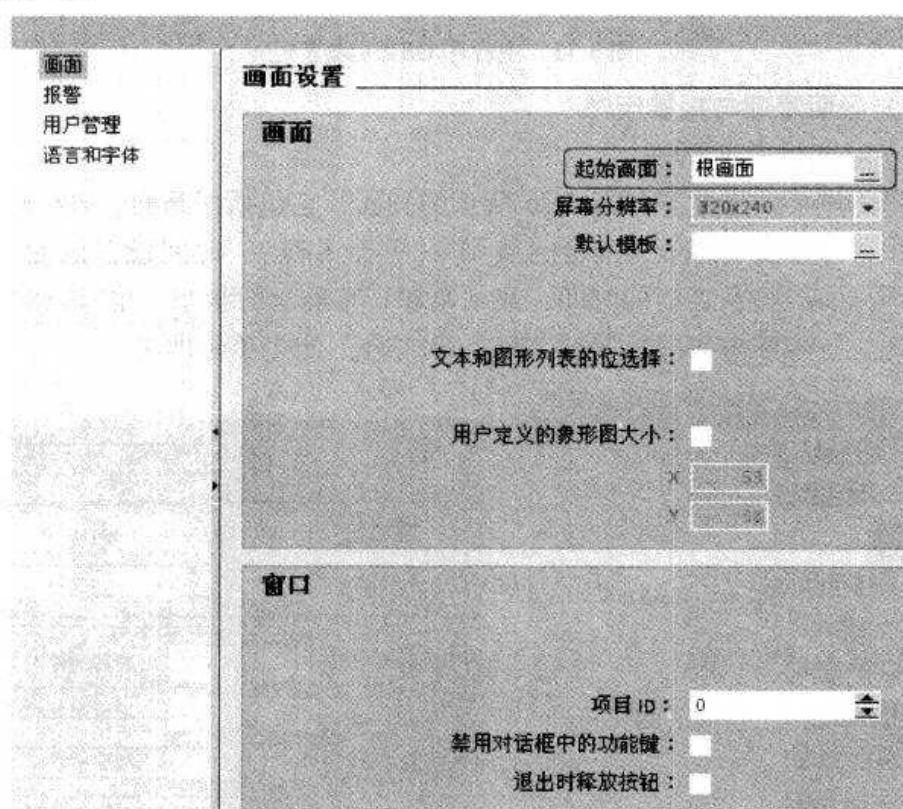


图 3-80 设置起始画面

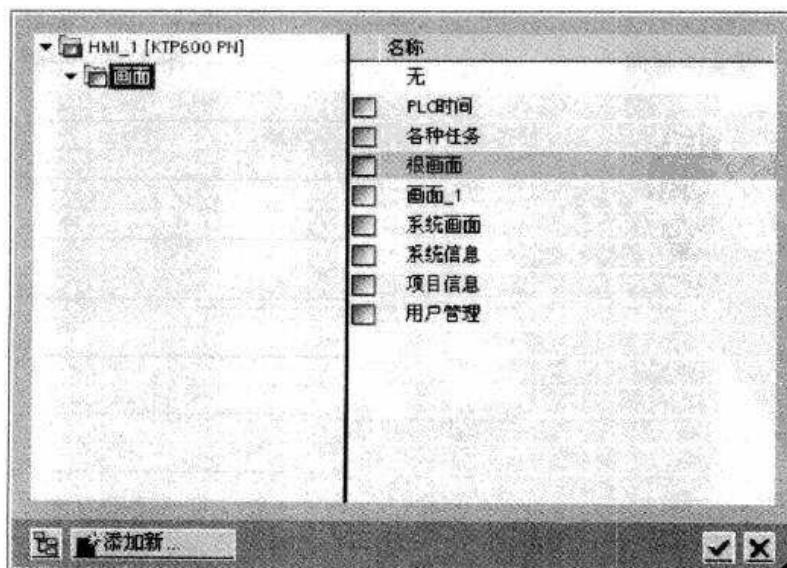


图 3-81 选择起始画面



(3) 触摸屏画面的模板设置

从图 3-77 和图 3-78 中可以看到触摸屏画面的最下角有“**!**”、“系统画面”、“主画面”、“时间设置”等文字或符号，具体如图 3-82 所示为画面切换的功能区域，显然在任何画面中都可以通过点击相应的按钮来跳转到相应的画面。



图 3-82 画面切换的功能区域

在 KTP 触摸屏中，有一个画面管理功能，如图 3-83 所示，选择“Template_1”（即模板）菜单，然后即可进入图 3-84 所示的模板及其属性画面，进行文本输入。



图 3-83 画面管理功能

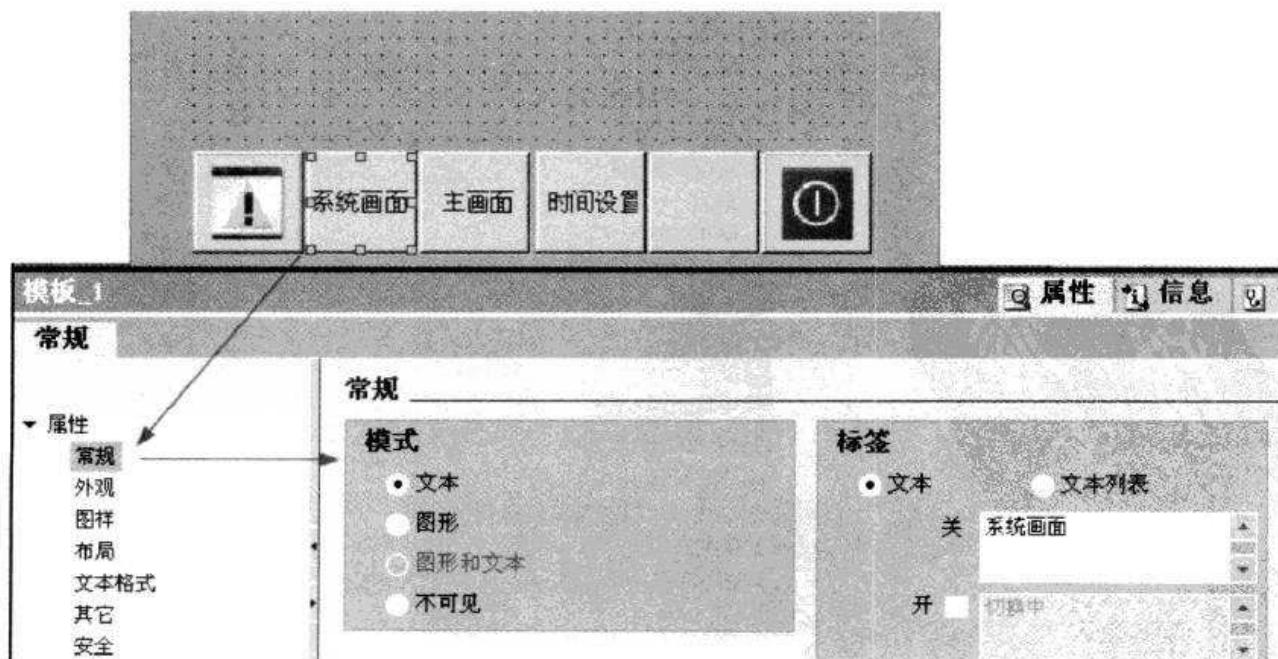


图 3-84 模板的文本输入



图解西门子 S7-1200 PLC 入门到实践

在模板中定义好文本后，即可进行事件设置，如本案例进行画面切换的事件为“单击”，其添加函数为“ActivateScreen”（即激活画面），选择“系统画面”即可（见图 3-85）。

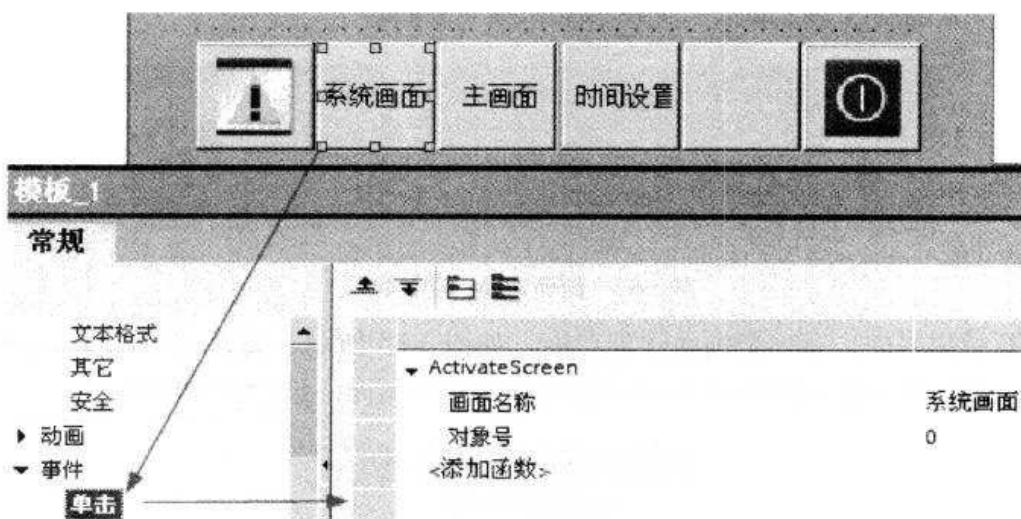


图 3-85 模板的单击事件

对于主画面和时间设置可以照此进行文本输入和事件函数定义。

(4) 触摸屏画面的报警组态

HMI 报警是 KTP 触摸屏的一个方便而有效的报警组态方式，它可以确保当故障到来时，即在当前页面中进行报警显示，以进行第一时间的处理，避免用户损失。

图 3-86 所示为 HMI 报警组态入口。



图 3-86 HMI 报警组态入口



HMI 的报警可以包括离散量报警、模拟量报警、报警类别和报警组，如图 3-87 所示，添加新对象，即可以将“报警字”触发变量的第“8”触发位设置为“错误”报警类别，其事件文本为“电动机热继电器过载”。注意，报警位% M12.0 为报警字% MW12 的第 8 位，因为 MB12 为高位，MB13 为低位，这个需要读者进行认真思考。

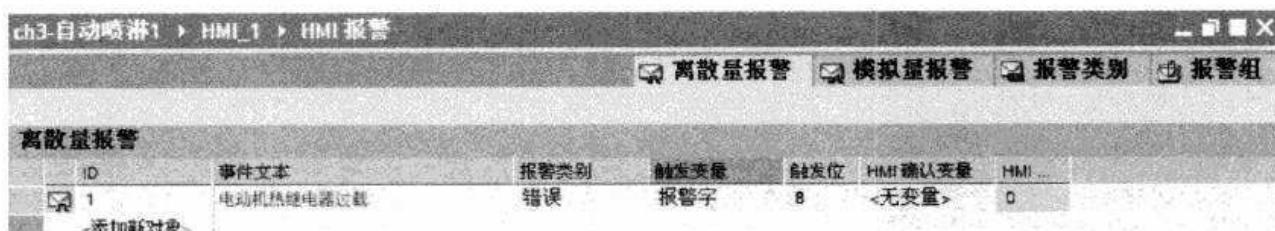


图 3-87 HMI 报警设置

图 3-88 所示为实际主画面中的报警提示，在右上角跳出▲符号，点击进入后会看到图 3-89 所示的故障报警画面。当故障消失后，可以通过点击■符号，清理历史故障。



图 3-88 实际主画面中的报警提示

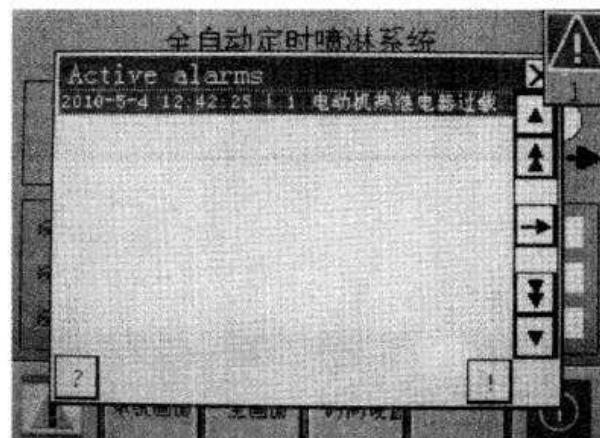


图 3-89 故障报警画面

4. 模拟触摸屏

对于触摸屏的组态，TIA 软件还提供了模拟功能，从 SIMATIC 主菜单进入到 WinCC flexible 2008 下的 WinCC flexible Runtime Simulator（见图 3-90）。



图 3-90 进入触摸屏模拟功能



图解西门子 S7-1200 PLC 入门到实践

图 3-91 所示为触摸屏模拟开始画面，通过选择打开命令，即可从图 3-92 所示的文件夹中找到 pdata 文件。

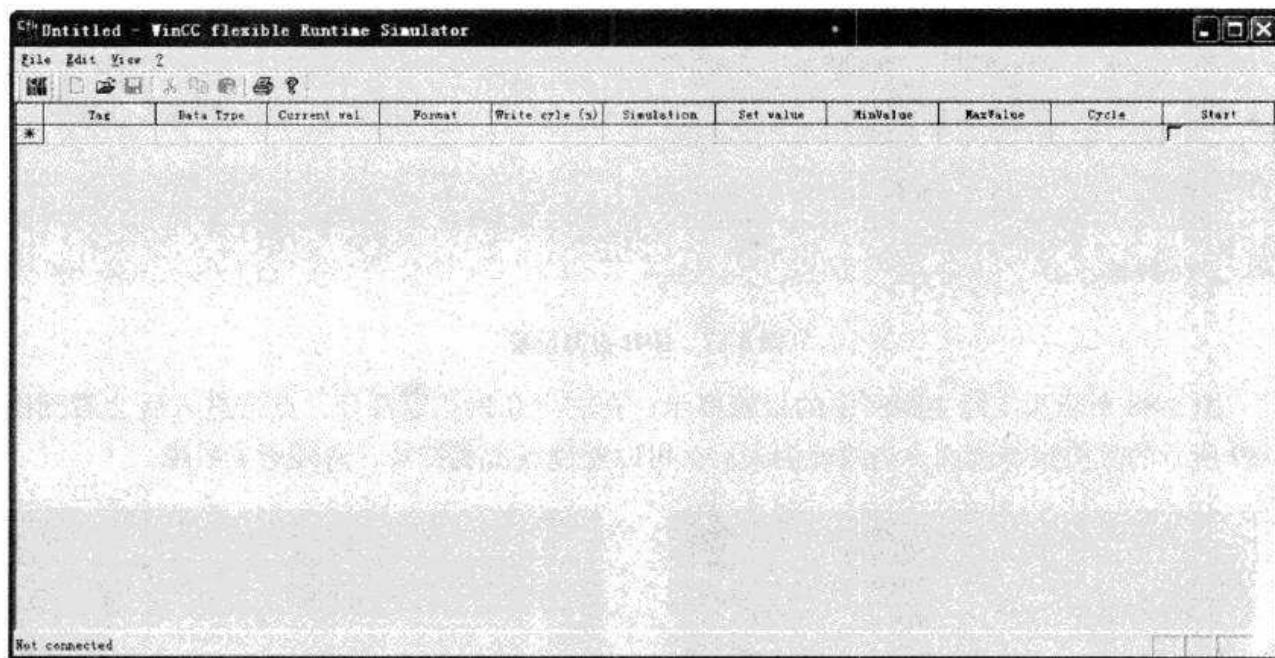


图 3-91 触摸屏模拟开始画面

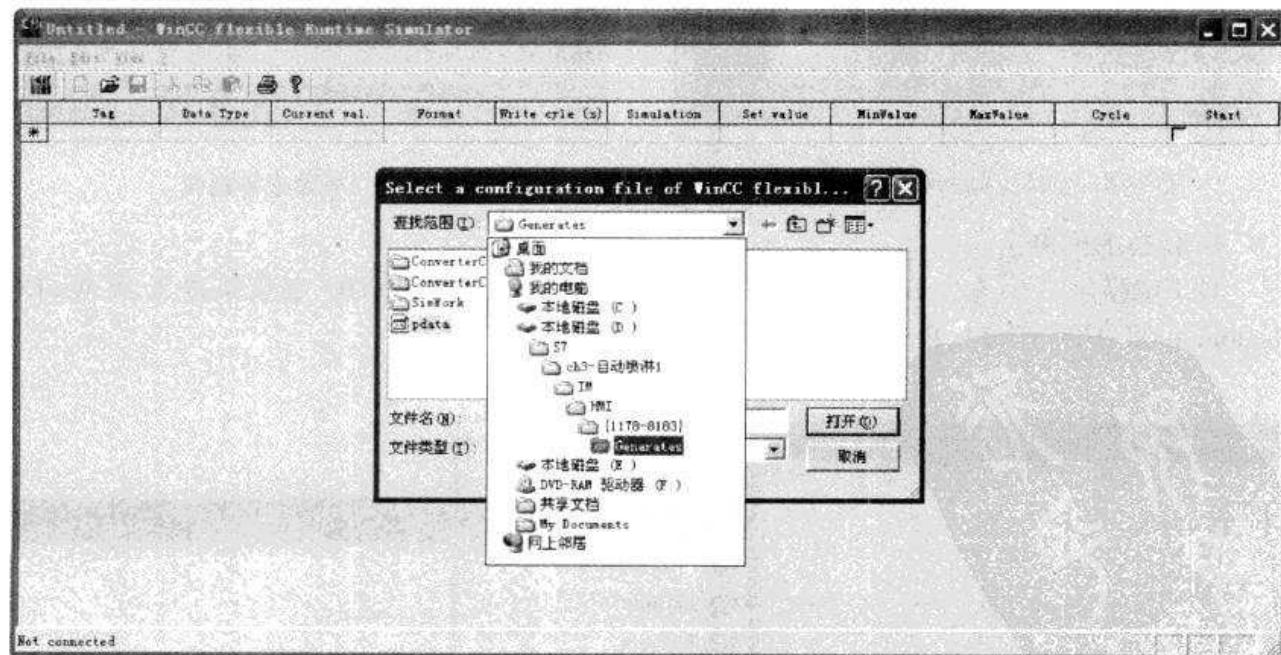


图 3-92 选择 pdata 文件

经确定后，就可以看到图 3-93 ~ 图 3-96 所示的模拟画面。

第3章 KTP 触摸屏编程

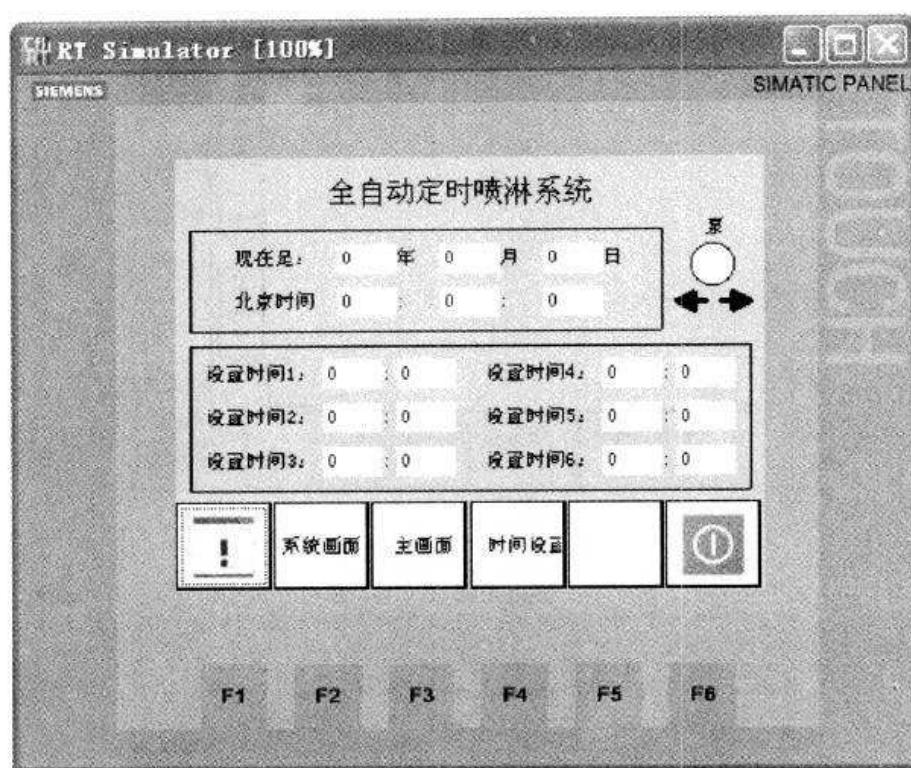


图 3-93 模拟画面一

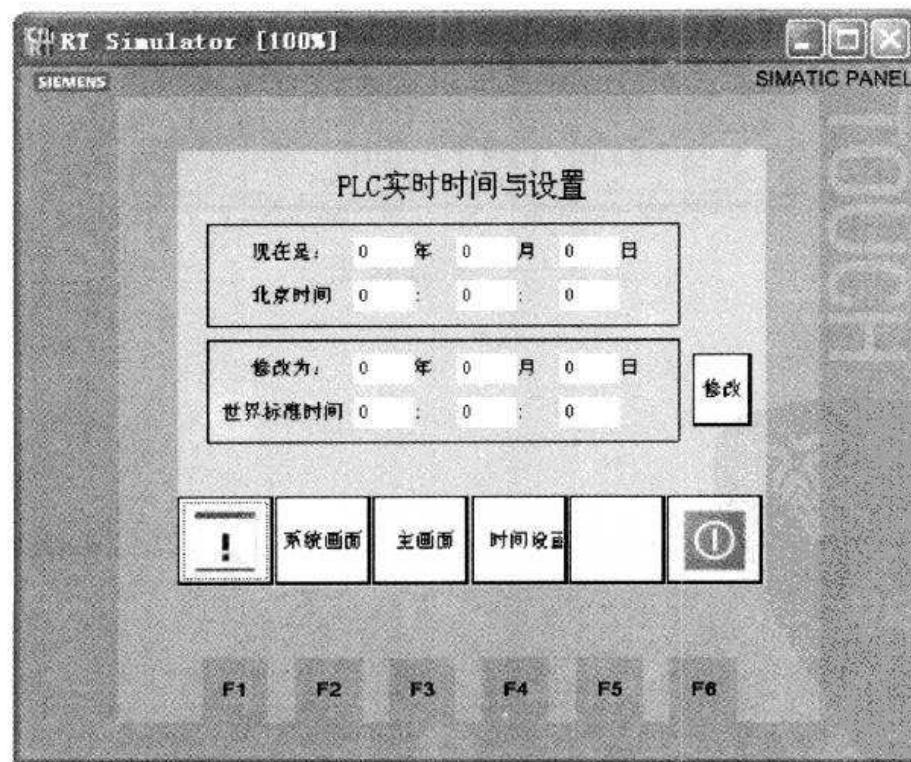


图 3-94 模拟画面二



图解西门子 S7-1200 PLC 入门到实践

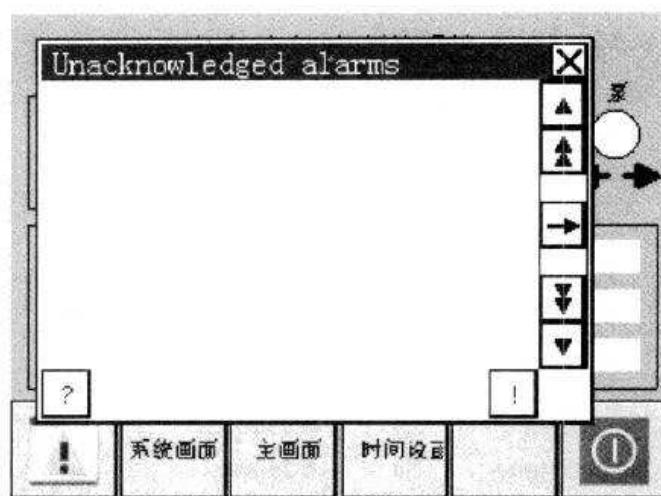


图 3-95 模拟画面三

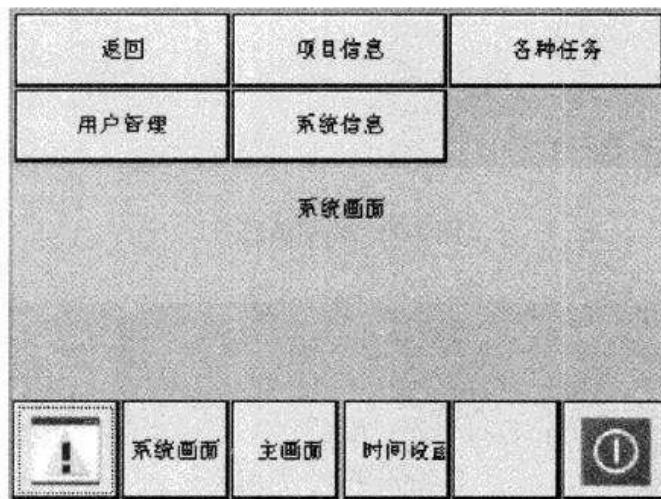
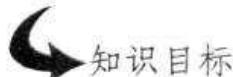


图 3-96 模拟画面四

第4章 S7-1200 PLC的PID控制

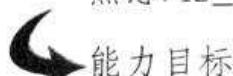
自动控制是指在无人直接参与的情况下，利用控制装置操纵受控对象，使被控量等于给定值或按给定信号的变化规律去变化的过程。在工程实际中，应用最为广泛的调节器控制规律为比例（P）、积分（I）、微分（D）控制，简称PID控制或调节。PID控制器问世至今已有近80年历史，它以其结构简单、稳定性好、工作可靠、调整方便的特点而成为工业控制的主要技术之一。S7-1200 PLC创新性地引入了工艺对象这个概念，使得PID控制的编程更加简洁、方便。

本章的学习目标如下：

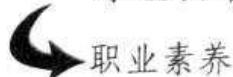


了解PID的基本概念、PID控制器及其应用；熟悉S7-1200 PLC的PID工艺对象；

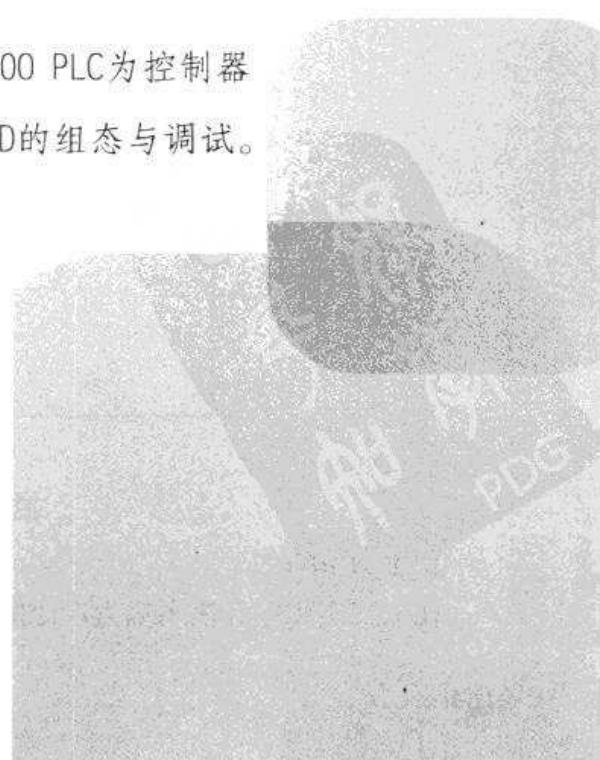
熟悉PID_Compact指令及编程特点。



能用常见的自动化元器件组成一个以S7-1200 PLC为控制器的PID回路，并通过增加工艺对象来实现PID的组态与调试。



培养自动控制系统解决方案的创新精神。





4.1 背景知识介绍：PID 的基本概念

4.1.1 PID 的基本概念

1. 自动控制

自动控制是指在无人直接参与的情况下，利用控制装置操纵受控对象，使被控量等于给定值或按给定信号的变化规律去变化的过程。

如图 4-1a 和图 4-1b 所示为液位控制中的两种方式，即手动液位控制和自动液位控制。

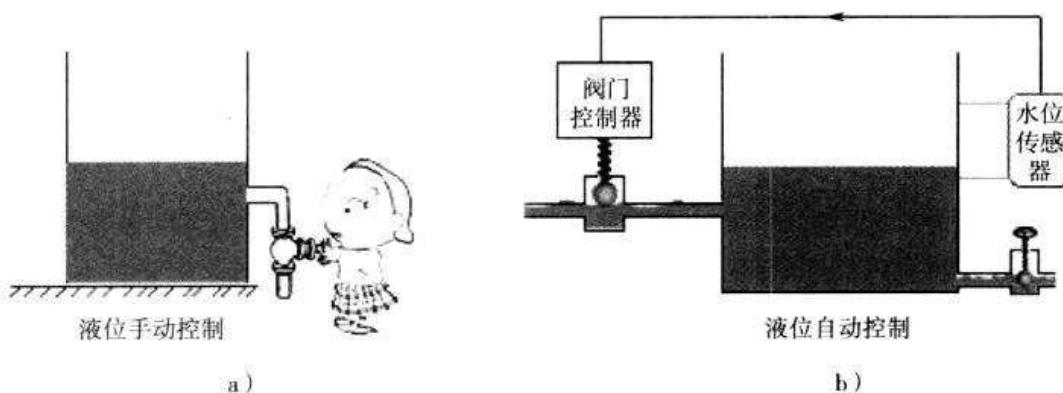


图 4-1 液位手动和自动控制

液位手动控制是根据眼来观察、脑来判断、手来操作的一种方式，其目的就是为了减少或消除液位差 Δh ，以保证恒液位控制。

在自动控制中，则是要建立一个受控对象（水池）、一个输出量（实际水位）、一个输入量（要求水位）、一个检测装置（水位传感器）、一个执行机构（阀门），根据图 4-2 所示的自动控制示意图进行控制。通过给定量和实际检测得到的实际值，得出一个偏差量，再由控制器进行控制。

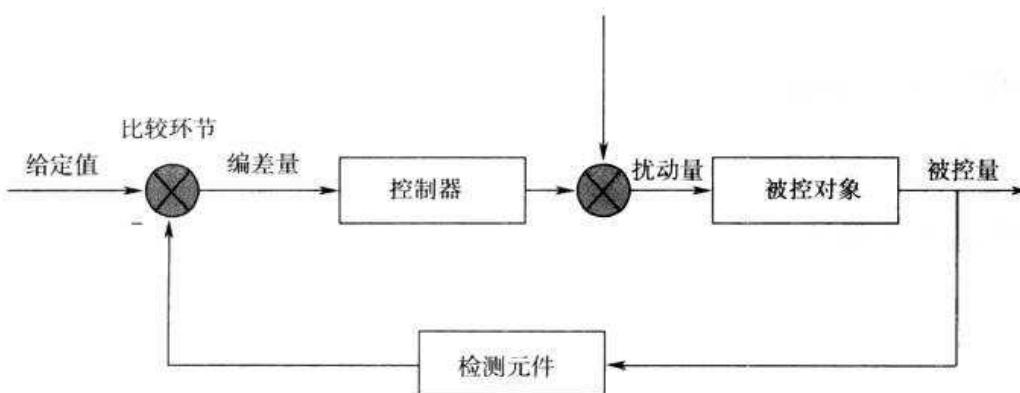


图 4-2 液位自动控制示意图

2. PID 控制

在工程实际中，应用最为广泛的调节器控制规律为比例、积分、微分控制，简称 PID 控



制或调节。PID 控制器问世至今已有近 80 年历史，它以其结构简单、稳定性好、工作可靠、调整方便的特点而成为工业控制的主要技术之一。当被控对象的结构和参数不能完全掌握，或得不到精确的数学模型，控制理论的其他技术难以采用时，系统控制器的结构和参数必须依靠经验和现场调试来确定，这时应用 PID 控制技术最为方便。即当我们不完全了解一个系统和被控对象，或不能通过有效的测量手段来获得系统参数时，最适合用 PID 控制技术。

在实际中也有 PI 和 PD 控制。PID 控制器就是根据系统的误差，利用比例、积分、微分计算出控制量进行控制的。

(1) 比例控制

比例控制是一种最简单的控制方式。比例控制器的输出与输入误差信号成比例关系。当仅有比例控制时，系统输出存在稳态误差。

(2) 积分控制

在积分控制中，控制器的输出与输入误差信号的积分成正比关系。对一个自动控制系统，如果在进入稳态后存在稳态误差，则称这个控制系统是有稳态误差的或简称有差系统。为了消除稳态误差，在控制器中必须引入“积分项”。积分项对误差取决于时间的积分，随着时间的增加，积分项会增大。这样，即便误差很小，积分项也会随着时间的增加而加大，它推动控制器的输出增大使稳态误差进一步减小，直到等于零。因此，比例 + 积分 (PI) 控制器可以使系统在进入稳态后无稳态误差。

(3) 微分控制

在微分控制中，控制器的输出与输入误差信号的微分（即误差的变化率）成正比关系。自动控制系统在克服误差的调节过程中可能会出现振荡甚至失稳。其原因是由于存在有较大惯性组件（环节）或有滞后组件，它们具有抑制误差的作用，其变化总是落后于误差的变化。解决的办法是使抑制误差的作用的变化“超前”，即在误差接近零时，抑制误差的作用应该是零。这就是说，在控制器中仅引入“比例”项往往是不够的，比例项的作用仅是放大误差的幅值，而目前需要增加的是“微分项”，它能预测误差变化的趋势，这样，具有比例 + 微分的控制器就能够提前使抑制误差的控制作用等于零，甚至为负值，从而避免了被控量的严重超调。所以对有较大惯性或滞后的被控对象，比例 + 微分 (PD) 控制器能改善系统在调节过程中的动态特性。

3. PID 算法

在连续控制系统中，模拟 PID 的控制规律形式为

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (4-1)$$

式中， $e(t)$ 是偏差输入函数； $u(t)$ 是调节器输出函数； K_p 是比例系数； T_i 是积分时间常数； T_d 是微分时间常数。

由于式 (4-1) 为模拟量表达式，而 PLC 程序只能处理离散数字量，为此，必须将连续形式的微分方程化成离散形式的差分方程。式 (4-1) 经离散化后的差分方程为

$$u(k) = K_p \left[e(k) + \frac{1}{T_i} \sum_{i=0}^k Te(k-i) + T_d \frac{e(k) - e(k-1)}{T} \right] \quad (4-2)$$

式中， T 是采样周期； k 是采样序号， $k=0, 1, 2 \cdots i$ ； $u(k)$ 是采样时刻 k 时的输出值； $e(k)$ 是采样时刻 k 时的偏差值； $e(k-1)$ 是采样时刻 $k-1$ 时的偏差值。



为了减小计算量和节省内存开销，将式（4-2）化为递推关系式形式：

$$\begin{aligned} u(k) &= u(k-1) + K_p \left(1 + \frac{T}{T_1} + \frac{T_D}{T} \right) e(k) - K_p \left(1 + \frac{2T_D}{T} \right) e(k-1) + K_p \frac{T_D}{T} e(k-2) \\ &= u(k-1) + r_{0e}(k) - r_{0e}(k-1) + r_{2e}(k-2) \\ &= u(k-1) - r_0 f(k) + r_1 f(k-1) - r_2 f(k-2) + S_p(r_0 - r_1 + r_2) \end{aligned} \quad (4-3)$$

式中， S_p 是调节器设定值； $f(k)$ 是采样时刻 k 时的反馈值； $f(k-1)$ 是采样时刻 $k-1$ 时的反馈值； $f(k-2)$ 是采样时刻 $k-2$ 时的反馈值。至此，式（4-3）已可以用作编程算法使用了。

4.1.2 PID 控制器

1. 定义

PID 控制器是由比例、积分和微分单元组成。它在控制回路中连续检测受控变量的实际测量值，并将其与期望设定值进行比较。PID 使用所生成的控制偏差来计算控制器的输出，以便尽可能快速平稳地将受控变量调整到设定值。S7-1200 PLC 的 PID 控制回路是由受控对象、控制器、测量元件（传感器）和控制元件组成。

2. PID 控制器的简单应用

图 4-3 所示是使用具有 PID 控制的 SIMATIC S7-1200 作为控制器的室内温度自动控制系统，其中测量元件是传感器，用于测量加热室内的温度；控制元件是由 PLC 直接控制的加热器。

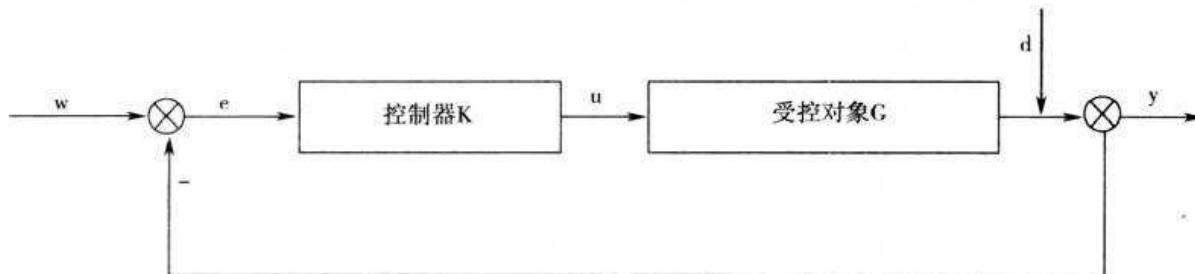


图 4-3 PID 控制器的简单应用

设定值 “w” 已预先定义，设定值是加热室中的期望温度 75°C。可通过设定值 (w) 和实际值 (y) 来计算控制偏差 (e)。控制器 (K) 可将控制偏差转换为受控变量 (u)。受控变量通过受控对象 (G) 来更改实际值 (y)。受控对象 (G) 为加热室中的温度调节，可以通过增加或减少能量输入进行控制。

除受控对象 (G) 外，也可以通过干扰变量 (d) 改变实际值 (y)。本应用中的干扰变量可能是加热室中意外的温度变化。例如，由室外温度变化引起的温度变化。使用 PID 控制器能尽可能快地达到所需的 75°C 并能尽可能地保持设定值不变。

在本例中，由于加热元件在关闭后继续发热，因此将超出设定值。该效应称为“过调”。如果实际值的控制和测量之间存在延时，则会发生过调。图 4-4 所示显示了 PID 控制的实际温度曲线图。

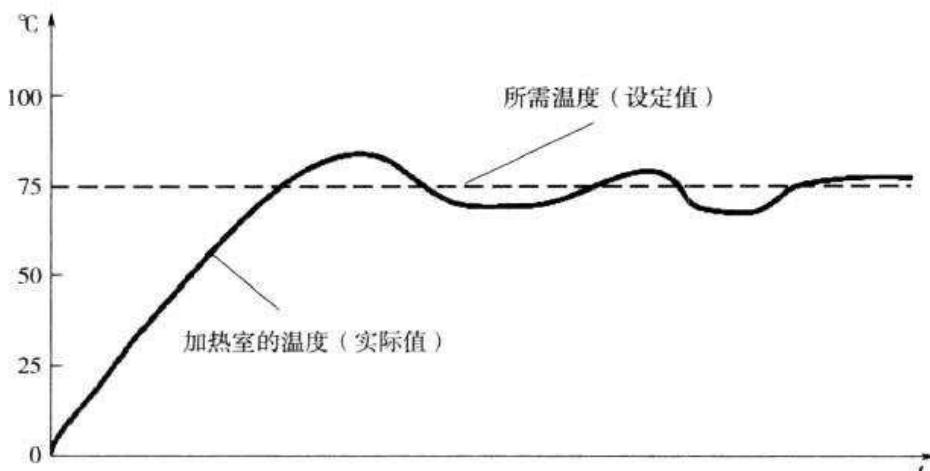


图 4-4 PID 控制的实际温度曲线图

3. PID 工艺对象与 PID 指令

“PID _ Compact”工艺对象是用于实现自动和手动模式下都可自我优化调节的 PID 控制器 PID _ Compact _ DB。在控制回路中，PID 控制器连续采集受控变量的实际测量值，并将其与期望设定值进行比较。

PID 控制器基于所生成的系统偏差计算控制器输出，尽可能快速稳定地将受控变量调整到设定值。在 PID 控制器中，控制器输出值通过以下 3 个分量进行计算：比例分量计算的控制器输出值与系统偏差成比例；积分分量计算的控制器输出值随着控制器输出的持续时间而增加，最终补偿控制器输出；PID 控制器的微分分量随着系统偏差变化率的增加而增加。受控变量将尽快调整到设定值。系统偏差的变化率减小时，微分分量也会随之减小。

工艺对象在“初始启动时自调节”期间自行计算 PID 控制器的比例、积分和微分分量。可通过“运行中自调节”对这些参数进行进一步优化。

一般来说，要在新的组织块中创建 PID 控制器的块。当前所创建的循环中断组织块将用作新的组织块。循环中断组织块可用于以周期性时间间隔启动程序，而与循环程序执行情况无关。循环中断 OB 将中断循环程序的执行，并会在中断结束后继续执行。图 4-5 所示为循环组织块、循环中断与 PID 控制器。

从图 4-5 中可以看出，PID 控制器的工作原理为：

- 1) 程序从 Main [OB1] 开始执行。
- 2) 循环中断每 100ms 触发一次，它会在任何时间（例如，在执行 Main [OB1] 期间）中断程序并执行循环中断 OB 中的程序。在本例中，程序包含功能块 PID _ Compact。
- 3) 执行 PID _ Compact 并将值写入数据块 PID _ Compact (DB)。
- 4) 执行循环中断 OB 后，Main [OB1] 将从中断点继续执行，相关值将保留不变。
- 5) Main [OB1] 操作完成。
- 6) 将重新开始该程序循环。

图解西门子 S7-1200 PLC 入门到实践

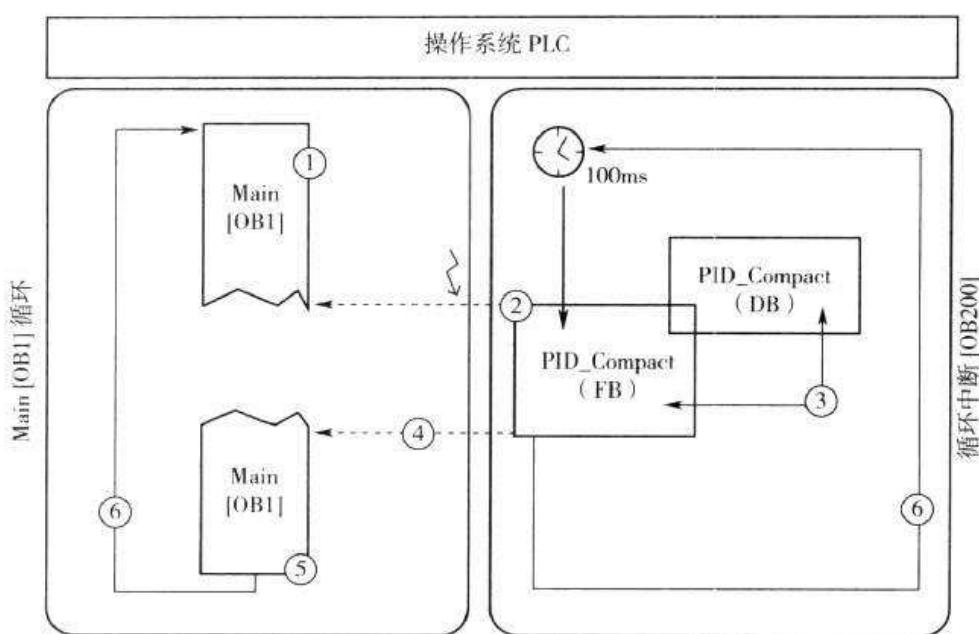


图 4-5 循环组织块、循环中断与 PID 控制器

4.2 技能训练：液压站压力控制的 PID 构建

4.2.1 案例介绍

液压站又称液压泵站，是独立的液压装置（见图 4-6）。它按逐级要求供油，并控制液压油流的方向、压力和流量，适用于主机与液压装置可分离的各种液压机械上。用户只要将液压站与主机上的执行机构（油缸或油电动机）用油管相连，液压机械即可实现各种规定的动作和循环工作。

液压站是由泵装置、集成块或阀组合、油箱、电气盒组合而成。各部的功能为：泵装置——装有电动机和油泵，是液压站的动力源，将机械能转化为液压油的压力能；集成块——由液压阀及通道体组装而成，对液压油实行方向、压力和流量调节；阀组合——板式阀装在立板上，板后管连接，与集成块功能相同；油箱——板焊的半封闭容器，上面还装有滤油网、空气滤清器等，用来储油、油的冷却及过滤；电气盒——分两种型式，一种设置外接引线的端子板；一种配置了全套控制电器。

液压站的工作原理为：电动机带动油泵转动，泵从油箱中吸油供油，将机械能转化为液压站的压力能，液压油通过集成块（或阀组合）实现了方向、压力、流量调节后经外接管路并至液压机械的油缸或油电动机中，从而控制液动机方向的变换、力量的大小及速度的快

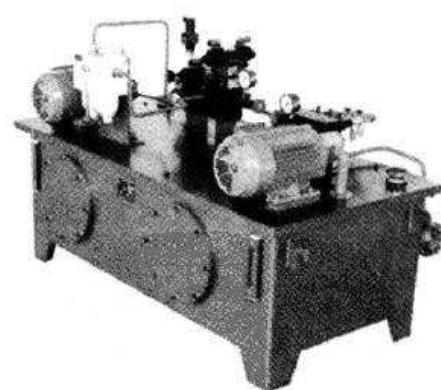


图 4-6 液压站



慢，推动各种液压机械做功。

本案例为通过液压泵电动机的转速来控制可调输出的油压压力，控制器采用 PLC 中的 PID 回路，具体工作示意如图 4-7 所示。请选择合适的方案并进行编程、调试。

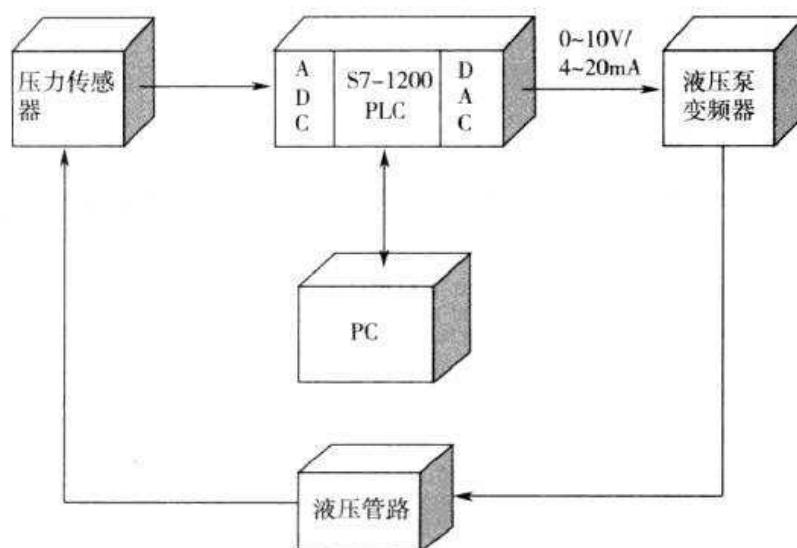


图 4-7 液压站压力工作示意

4.2.2 在 S7-1200 PLC 中添加 PID 工艺对象

在 S7-1200 PLC 中添加 PID 工艺对象的方式有很多种，下面介绍其中比较简单的一种方式。

- 1) 在现有 PLC 的项目树中打开“工艺对象”，并点击“添加新对象”（见图 4-8）。

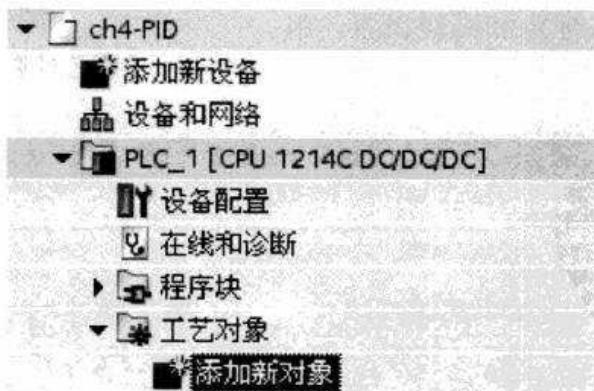


图 4-8 添加新的工艺对象

- 2) 进入如图 4-9 所示的“添加新的 PID 对象窗口”，选择“PID 控制器”，这时会出现类型为 **PID_CMPT [FB1130]** 的默认选项。注意，PID_CMPT 就是 PID_Compact 的简称。编号为 DB 的序号，可以手动，也可以自动。



图 4-9 添加新的 PID 对象窗口

3) 从项目树中进入如图 4-10 所示的“工艺对象” PID_Compact_1 [DB1]，这时会出现组态和调试两个功能。

选择组态功能，则会出现图 4-11 所示的菜单，包括基本参数、输入标定和高级设置。组态完成情况示意见表 4-1。

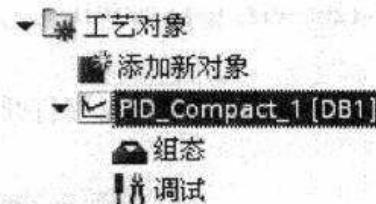


图 4-10 “工艺对象” PID_Compact_1 [DB1]

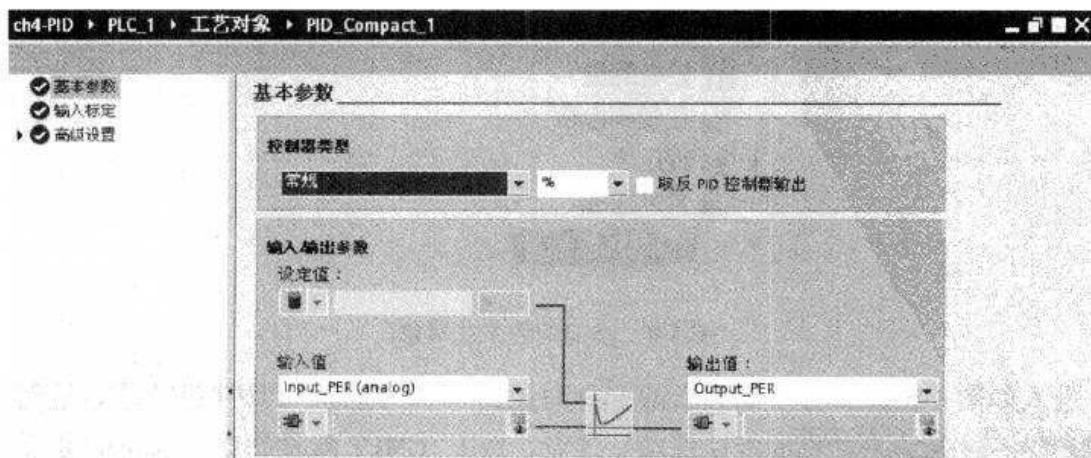


图 4-11 PID 的组态菜单



表 4-1 组态完成情况示意

| | |
|--|---|
| <input checked="" type="checkbox"/> 蓝色 | 组态包含默认值且已完成 组态仅包含默认值。通过这些默认值即可使用工艺对象，而无需进一步更改 |
| <input checked="" type="checkbox"/> 绿色 | 组态包含用户定义的值且已完成 组态的所有输入域中均包含有效值，而且至少更改了一个默认值 |
| <input checked="" type="checkbox"/> 红色 | 组态不完整或有缺陷 至少一个输入域或下拉列表框不包含任何值或者包含的值无效。相应域或下拉列表框的背景为红色。单击这些域或下拉列表框时，弹出的错误消息便会指出错误原因 |

(1) 控制器类型

控制器类型用于预先选择需控制值的单位。在本例中，将单位为“Bar”的“压力”用作控制器类型，如图 4-12 所示。常见的控制器类型包括速度控制、压力控制、流量控制、温度控制等，默认是以百分比为单位的“常规”控制器。

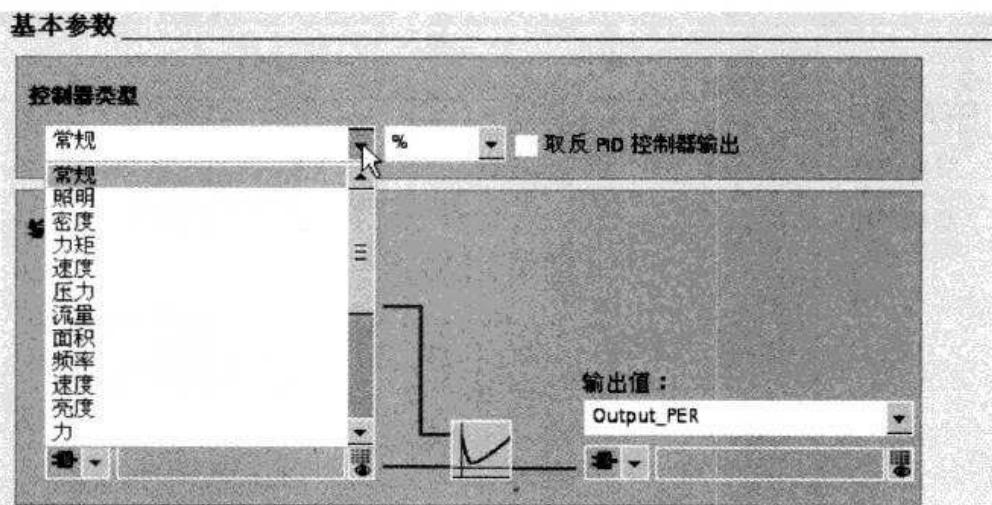


图 4-12 控制器类型

如果受控值的增加会引起实际值的减小（例如，由于阀位开度增加而使水位下降，或者由于冷却性能增加而使温度降低），请选中“取反 PID 控制器输出”复选框。

(2) 输入/输出参数

在该区域中，为设定值、实际值和工艺对象“PID _ Compact”的受控变量提供输入和输出参数。输入值为 Input，输出值为 Output _ PER (analog)，如图 4-13 所示。在输入值选项中，Input 表示使用从用户程序而来的反馈值；Input _ PER (analog) 表示使用外设输入。在输出值选项中，Output 表示输出至用户程序，外设输出；Output _ PER 表示外设输出；Output _ PWM 表示使用 PWM 输出。

(3) 输入标定

如图 4-14 所示为反馈值输入标定。其中，标定上限值和上限为一组，标定下限值和下限为一组，而上限和下限为用户设置的高低限值，当反馈值达到高限或低限时，系统将停止 PID 的输出。



基本参数

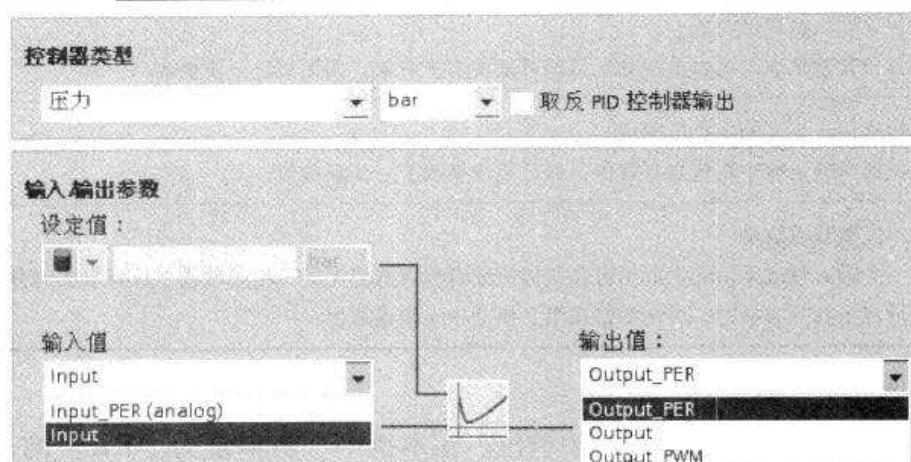


图 4-13 输入值选择

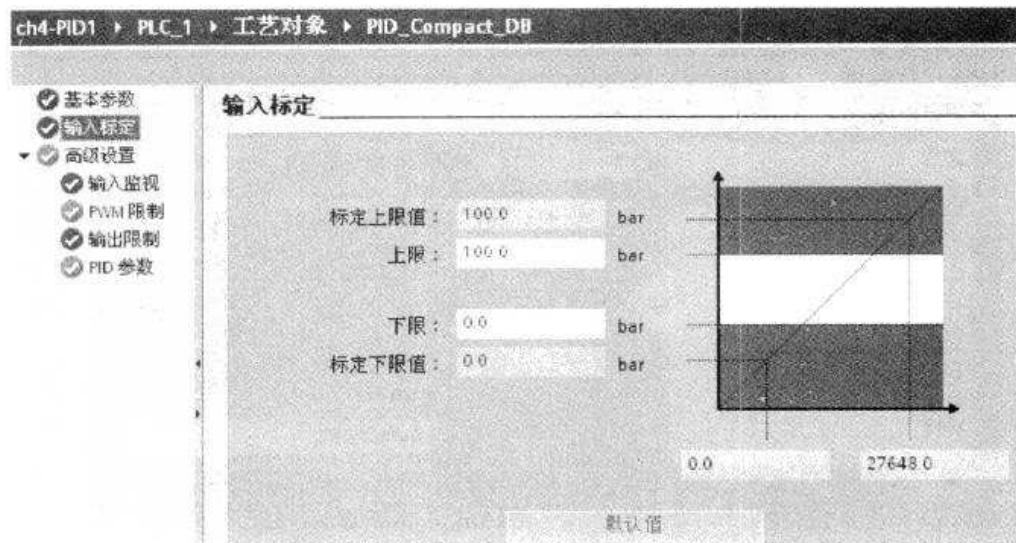


图 4-14 反馈值输入标定

(4) 高级设置

如图 4-15 所示为高级设置中的输入监视。当反馈值达到高限或低限时，PID 指令块会

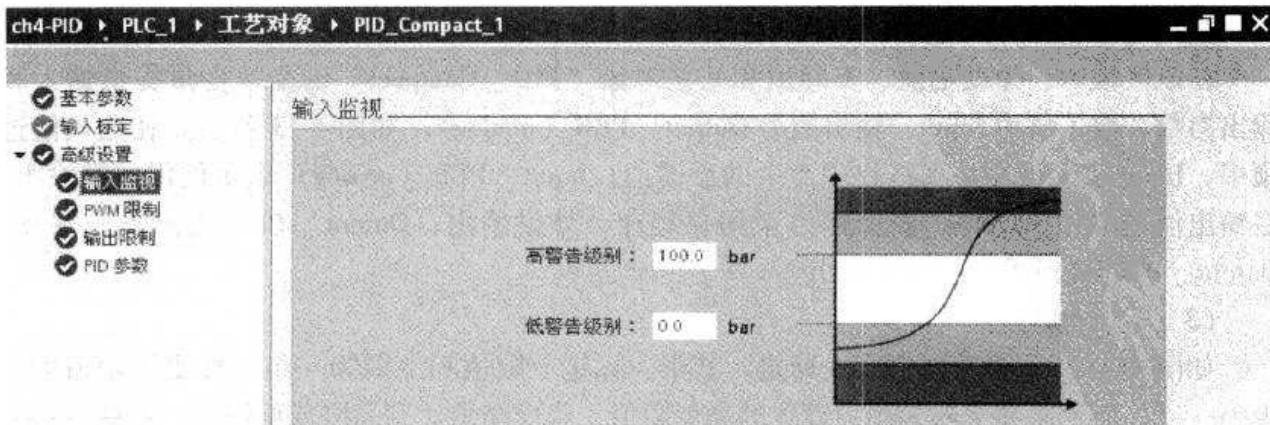


图 4-15 输入监视

第4章 S7-1200 PLC 的 PID 控制



给出相应的报警位。

当输出是 PWM 而非模拟量时，则需要定义如图 4-16 所示的 PWM 限制，即最小接通时间和最小关闭时间（关于 PWM 的概念将在后续项目中介绍）。

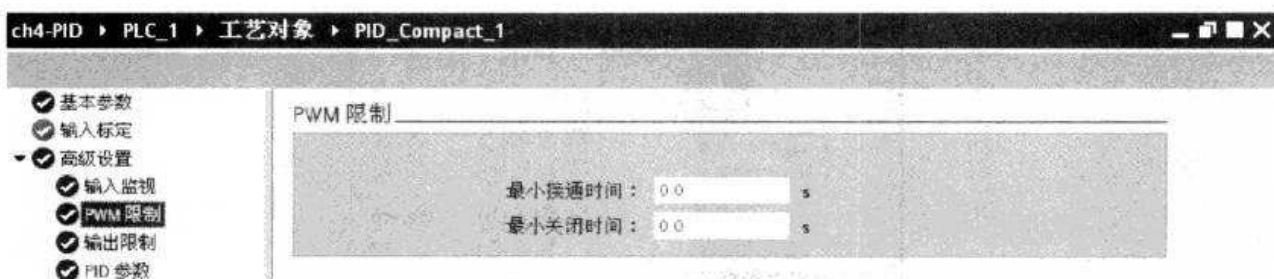


图 4-16 PWM 限制

当输出是 PWM 而非模拟量时，还需要定义如图 4-17 所示的 PID 参数。

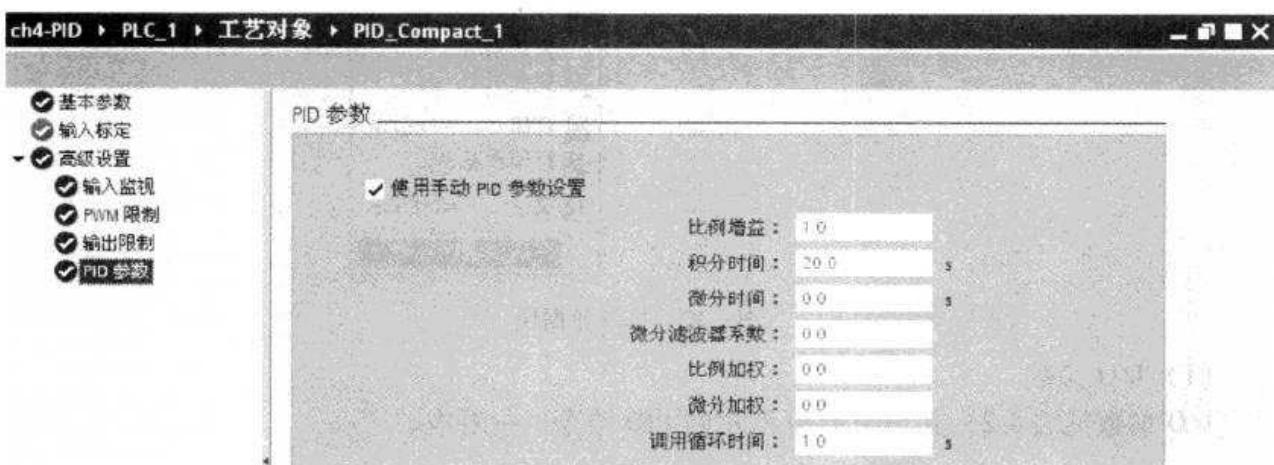


图 4-17 PID 参数

在某些场合，为了确保输出可控的模拟量，可以如图 4-18 所示进行输出限制的定义，包括上限和下限。

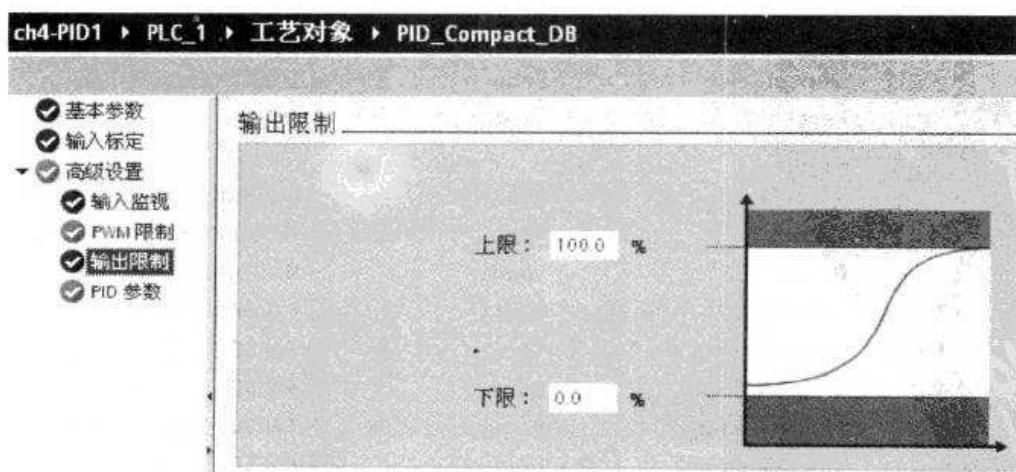


图 4-18 输出限制的定义



4) 完成以上的组态之后，就可以在项目树中的 PID_Compact_1 [DB1] 按右键进入“在编辑器中打开”（见图 4-19），进入背景数据块参数表。



图 4-19 打开数据块

(1) I/O 参数

I/O 参数见表 4-2，其与图 4-20 所示的 PID 指令一一对应。

表 4-2 I/O 参数

| | 名称 | 数据类型 | 初始值 |
|----|-----------------|-------|---------------|
| 1 | ▼ Input | | |
| 2 | Setpoint | Real | 0.000000e+000 |
| 3 | Input | Real | 0.000000e+000 |
| 4 | Input_PER | Word | W#16#0000 |
| 5 | ManualEnable | Bool | FALSE |
| 6 | ManualValue | Real | 0.000000e+000 |
| 7 | Reset | Bool | FALSE |
| 8 | ▼ Output | | |
| 9 | ScaledInput | Real | 0.000000e+000 |
| 10 | Output | Real | 0.000000e+000 |
| 11 | Output_PER | Word | W#16#0000 |
| 12 | Output_PWM | Bool | FALSE |
| 13 | SetpointLimit_H | Bool | FALSE |
| 14 | SetpointLimit_L | Bool | FALSE |
| 15 | InputWarning_H | Bool | FALSE |
| 16 | InputWarning_L | Bool | FALSE |
| 17 | State | Int | 0 |
| 18 | Error | Dword | DW#16#0000 |

第4章 S7-1200 PLC 的 PID 控制

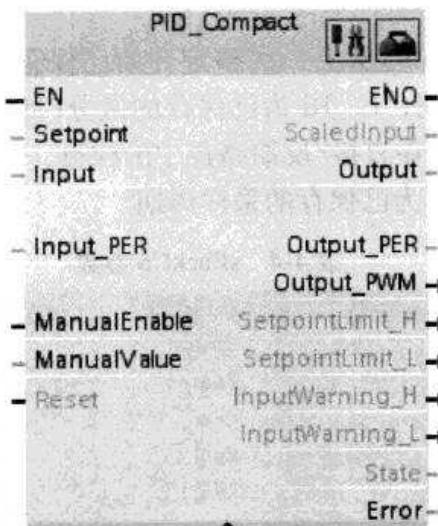


图 4-20 PID 指令

(2) Static 参数

Static 参数（见表 4-3）表示固定值，如 sd_VersionID 为控制器版本（如表中的版本为 1.0.0.9）；sb_GetCycleTime 为开始自动语句采样时间；sb_EnCyclEstimation 为使能预估采样时间；sb_EnCyclMonitoring 为使能监视采样时间；sb_RunModeByStartup 为在上电或复位后保持上一次状态或保持未激活状态；si_Unit 为反馈量单位；si_Type 为控制器类型；sd_Warning 为警告信息。

表 4-3 Static 参数

| | 名称 | 数据类型 | 初始值 |
|----|---------------------|-------|----------------|
| 19 | ▼ InOut | | |
| 20 | ▼ Static | | |
| 21 | sd_VersionID | DWord | DW#16#01000009 |
| 22 | sb_ResOld | Bool | FALSE |
| 23 | sb_TMBeginExec | Bool | FALSE |
| 24 | sb_GetCycleTime | Bool | TRUE |
| 25 | sb_EnCyclEstimation | Bool | TRUE |
| 26 | sb_EnCyclMonitoring | Bool | TRUE |
| 27 | sb_Startup | Bool | FALSE |
| 28 | sb_RunModeByStartup | Bool | TRUE |
| 29 | sby_ESData_1 | Byte | B#16#00 |
| 30 | sby_ESData_2 | Byte | B#16#00 |
| 31 | si_TMCnt | Int | 0 |
| 32 | si_Unit | Int | 0 |
| 33 | si_Type | Int | 5 |
| 34 | si_SveModeByRes | Int | 0 |
| 35 | sd_Warning | DWord | DW#16#0000 |
| 36 | st_TMEnd | Time | T#0MS |
| 37 | sr_TMDiff | Real | 0.000000e+000 |
| 38 | sr_TMDiffMax | Real | 0.000000e+000 |
| 39 | sr_TMDiffMaxMed | Real | 0.000000e+000 |
| 40 | sr_TMDiffSum | Real | 0.000000e+000 |



(3) sBackUp 参数

sBackUp 参数（见表 4-4）为从上一次整定开始已保存的参数。r_Gain 为已保存的增益，r_Ti 为已保存的积分时间，r_Td 为已保存的微分时间，r_A 为已保存的微分滤波系数，r_B 为已保存的比例部分在直接/反馈路径中的权重，r_C 为已保存的微分部分在直接/反馈路径中的权重，r_Cycle 为已保存的采样周期。

表 4-4 sBackUp 参数

| | 名称 | 数据类型 | 初始值 |
|----|-----------|--------|---------------|
| 41 | ▼ sBackUp | Struct | |
| 42 | r_Gain | Real | 1.000000e+000 |
| 43 | r_Ti | Real | 2.000000e+001 |
| 44 | r_Td | Real | 0.000000e+000 |
| 45 | r_A | Real | 0.000000e+000 |
| 46 | r_B | Real | 0.000000e+000 |
| 47 | r_C | Real | 0.000000e+000 |
| 48 | r_Cycle | Real | 1.000000e+000 |

(4) sPid_Calc 参数

sPid_Calc 参数（见表 4-5）表示为 PID 计算过程值与计算命令。主要的参数包括：r_Cycle 为采样时间；b_RunIn 为强制在设定点运行；b_CalcParamSUT 为重新计算启动整定参数；b_CalcParamTIR 为重新计算运行整定参数。

表 4-5 sPid_Calc 参数

| | 名称 | 数据类型 | 初始值 |
|----|----------------|--------|---------------|
| 49 | ▼ sPid_Calc | Struct | |
| 50 | r_Cycle | Real | 1.000000e-001 |
| 51 | r_Resol | Real | 1.000000e+000 |
| 52 | b_Ctrln | Bool | FALSE |
| 53 | b_Switch_On | Bool | FALSE |
| 54 | b_Switch_Off | Bool | FALSE |
| 55 | b_Jump_On | Bool | FALSE |
| 56 | b_Jump_Off | Bool | FALSE |
| 57 | b_Jump | Bool | FALSE |
| 58 | b_CalcOn | Bool | FALSE |
| 59 | b_SpCrossed | Bool | FALSE |
| 60 | b_PvFilAdapt | Bool | FALSE |
| 61 | b_2StepDeadTm | Bool | FALSE |
| 62 | b_LastPeriod | Bool | FALSE |
| 63 | b_Part1 | Bool | FALSE |
| 64 | b_Part2 | Bool | FALSE |
| 65 | b_RunIn | Bool | FALSE |
| 66 | b_FrstDerivRdy | Bool | FALSE |
| 67 | b_SpOffLtd | Bool | FALSE |
| 68 | b_TimeAdapt | Bool | FALSE |
| 69 | b_CalcParamSUT | Bool | FALSE |
| 70 | b_CalcParamTIR | Bool | FALSE |
| 71 | i_CtrlTypeSUT | Int | 0 |
| 72 | i_CtrlTypeTIR | Int | 0 |
| 73 | i_WPcyclMax | Int | 0 |

第4章 S7-1200 PLC 的 PID 控制



(续)

| | 名称 | 数据类型 | 初始值 |
|-----|----------------|------|---------------|
| 74 | i_WPCycl | Int | 0 |
| 75 | i_FilCyc | Int | 0 |
| 76 | i_MaxPeriod | Int | 0 |
| 77 | i_Ev4Step | Int | 0 |
| 78 | i_RepeatProc | Int | 0 |
| 79 | i_CntPeriod | Int | 0 |
| 80 | i_CtrlValAdapt | Int | 0 |
| 81 | i_Counter4 | Int | 0 |
| 82 | d_Counter1 | DInt | L#0 |
| 83 | d_Counter2 | DInt | L#0 |
| 84 | d_Counter3 | DInt | L#0 |
| 85 | d_CycCounter | DInt | L#0 |
| 86 | d_CycCountEnd | DInt | L#0 |
| 87 | d_TOn | DInt | L#0 |
| 88 | d_TOff | DInt | L#0 |
| 89 | d_TSUm | DInt | L#0 |
| 90 | d_TPnt1 | DInt | L#0 |
| 91 | d_THys | DInt | L#0 |
| 92 | d_THysAlt | DInt | L#0 |
| 93 | r_Medi | Real | 0.000000e+000 |
| 94 | r_Pv0 | Real | 0.000000e+000 |
| 95 | r_PvAlt | Real | 0.000000e+000 |
| 96 | r_PvAlt2 | Real | 0.000000e+000 |
| 97 | r_PvAltSUT | Real | 0.000000e+000 |
| 98 | r_PvMedi | Real | 0.000000e+000 |
| 99 | r_LmnFilOld1 | Real | 0.000000e+000 |
| 100 | r_LmnFilOld2 | Real | 0.000000e+000 |
| 101 | r_Stabw_Pv_1 | Real | 0.000000e+000 |
| 102 | r_Stabw_Pv_2 | Real | 0.000000e+000 |
| 103 | r_SpAlt | Real | 0.000000e+000 |
| 104 | r_Time | Real | 0.000000e+000 |
| 105 | r_PvDxMax | Real | 0.000000e+000 |
| 106 | r_TDxMax | Real | 0.000000e+000 |
| 107 | r_Noise | Real | 0.000000e+000 |
| 108 | r_Noise2 | Real | 0.000000e+000 |
| 109 | r_Dx0 | Real | 0.000000e+000 |
| 110 | r_Dx | Real | 0.000000e+000 |
| 111 | r_Dx2 | Real | 0.000000e+000 |
| 112 | r_DxMax | Real | 0.000000e+000 |
| 113 | r_DxMax2 | Real | 0.000000e+000 |
| 114 | r_DiffDx | Real | 0.000000e+000 |
| 115 | r_DiffDx2 | Real | 0.000000e+000 |
| 116 | r_Break | Real | 1.000000e+002 |
| 117 | r_BreakTm | Real | 0.000000e+000 |
| 118 | r_X1 | Real | 0.000000e+000 |



(续)

| | 名称 | 数据类型 | 初始值 |
|-----|-------------|------|---------------|
| 119 | r_X2 | Real | 0.000000e+000 |
| 120 | r_Up | Real | 0.000000e+000 |
| 121 | r_Down | Real | 0.000000e+000 |
| 122 | r_Switch | Real | 0.000000e+000 |
| 123 | r_AKrit | Real | 0.000000e+000 |
| 124 | r_DKrit | Real | 0.000000e+000 |
| 125 | r_LmnOpt | Real | 0.000000e+000 |
| 126 | r_ErOld | Real | 0.000000e+000 |
| 127 | r_TRelOld | Real | 1.000000e+000 |
| 128 | r_StdAbw | Real | 0.000000e+000 |
| 129 | r_Lmn0 | Real | 0.000000e+000 |
| 130 | r_LmnFacLim | Real | 0.000000e+000 |
| 131 | r_Lmn1 | Real | 0.000000e+000 |
| 132 | r_Sp1 | Real | 0.000000e+000 |
| 133 | r_SpOff | Real | 1.000000e+000 |
| 134 | r_SHys | Real | 0.000000e+000 |
| 135 | r_Progress | Real | 0.000000e+000 |

(5) sPid_Cmpt 参数

sPid_Cmpt 参数（见表 4-6）主要描述的是 PID_Compact 控制的组态值。其中包括：r_Sp_Hlm、r_Sp_Llm 为设定值高限、设定值低限；r_Pv_Norm_IN_1、r_Pv_Norm_IN_2、r_Pv_Norm_OUT_1、r_Pv_Norm_OUT_2 为输入量程化低限、输入量程化高限、输出量程化低限、输出量程化高限；r_Pv_Hlm、r_Pv_Llm、r_Pv_HWm、r_Pv_LWm 为反馈高限、反馈低限、反馈报警高限、反馈报警低限；r_Lmn_Hlm、r_Lmn_Llm 为输出高限、输出低限。

表 4-6 sPid_Cmpt 参数

| | 名称 | 数据类型 | 初始值 |
|-----|-----------------|--------|----------------|
| 136 | ▼ sPid_Cmpt | Struct | |
| 137 | r_Sp_Hlm | Real | 3.402822e+038 |
| 138 | r_Sp_Llm | Real | -3.402822e+038 |
| 139 | r_Pv_Norm_IN_1 | Real | 0.000000e+000 |
| 140 | r_Pv_Norm_IN_2 | Real | 2.764800e+004 |
| 141 | r_Pv_Norm_OUT_1 | Real | 0.000000e+000 |
| 142 | r_Pv_Norm_OUT_2 | Real | 1.000000e+002 |
| 143 | r_Lmn_Hlm | Real | 1.000000e+002 |
| 144 | r_Lmn_Llm | Real | 0.000000e+000 |
| 145 | b_Input_PER_On | Bool | True |
| 146 | b_LoadBackUp | Bool | False |
| 147 | b_InvCtrl | Bool | False |
| 148 | r_Lmn_Pwrn_PPTm | Real | 0.000000e+000 |
| 149 | r_Lmn_Pwrn_PBTm | Real | 0.000000e+000 |
| 150 | r_Man | Real | 0.000000e+000 |
| 151 | b_ManOn | Bool | False |

第4章 S7-1200 PLC 的 PID 控制

(续)

| | 名称 | 数据类型 | 初始值 |
|-----|-----------------|------|----------------|
| 152 | b_Sync | Bool | FALSE |
| 153 | b_PIDInit | Bool | FALSE |
| 154 | d_Per | DInt | L#0 |
| 155 | d_Per2 | DInt | L#0 |
| 156 | r_PvSum | Real | 0.000000e+000 |
| 157 | r_Pv_Hlm | Real | 5.0 |
| 158 | r_Pv_Llm | Real | 0.000000e+000 |
| 159 | r_Pv_HWrn | Real | 1.000000e+002 |
| 160 | r_Pv_LWrn | Real | 0.000000e+000 |
| 161 | r_Ctrl_IntHlm | Real | 3.402822e+038 |
| 162 | r_Ctrl_IntLlm | Real | -3.402822e+038 |
| 163 | r_Ctrl_IOutv | Real | 0.000000e+000 |
| 164 | r_Ctrl_IRest | Real | 0.000000e+000 |
| 165 | r_Ctrl_DRest | Real | 0.000000e+000 |
| 166 | r_Ctrl_DRueck | Real | 0.000000e+000 |
| 167 | d_Lmn_Pwrn_Ptm | DInt | L#0 |
| 168 | r_Lmn_Pwrn_Rest | Real | 0.000000e+000 |
| 169 | r_Lmn_Outv | Real | 0.000000e+000 |
| 170 | d_Lmn_Per_Outv | DInt | L#0 |

(6) sRet 参数

sRet 参数（见表 4-7）主要是设置控制器模式和返回当前 PID 参数值。其中最重要的是 i_Mode 为设置参数，0 为未激活，1 为启动整定模式，2 为运行整定模式，3 为自动模式，4 为手动模式。r_Ctrl_Gain、r_Ctrl_Ti、r_Ctrl_Td、r_Ctrl_Cycle 分别表示当前增益、当前积分时间、当前微分时间、当前采样时间。

表 4-7 sRet 参数

| | 名称 | 数据类型 | 初始值 |
|-----|------------------|--------|---------------|
| 181 | • sRet | Struct | |
| 182 | b_EnableManOld | Bool | FALSE |
| 183 | i_Mode | Int | 0 |
| 184 | i_ModeOld | Int | 0 |
| 185 | i_SveModeByEnMan | Int | 0 |
| 186 | i_StateOld | Int | 0 |
| 187 | r_Ctrl_Gain | Real | 1.000000e+000 |
| 188 | r_Ctrl_Ti | Real | 2.000000e+001 |
| 189 | r_Ctrl_Td | Real | 0.000000e+000 |
| 190 | r_Ctrl_A | Real | 0.000000e+000 |
| 191 | r_Ctrl_B | Real | 0.000000e+000 |
| 192 | r_Ctrl_C | Real | 0.000000e+000 |
| 193 | r_Ctrl_Cycle | Real | 1.000000e+000 |



4.2.3 PID 指令调用与编程

为了让 PID 运算以预想的采样频率工作，PID 指令必须用在定时发生的中断程序中或者用在主程序中被定时器所控制，以一定频率执行。

PID 指令调用与编程的步骤如下所示：

第一步，定义相关变量（见表 4-8），包括模拟量输入% IW96（即实际压力）、模拟量输出% QW96（变频泵输出频率）、手动/自动选择% I0.0。

表 4-8 变量定义

| | 名称 | 数据类型 | 地址 |
|---|---------|------|-------|
| 1 | 模拟量输入1 | Word | %IW96 |
| 2 | 模拟量输出1 | Word | %QW96 |
| 3 | 手动/自动选择 | Bool | %I0.0 |

第二步，如图 4-21 所示添加时间中断组织块 OB200，定义扫描时间为 100ms。

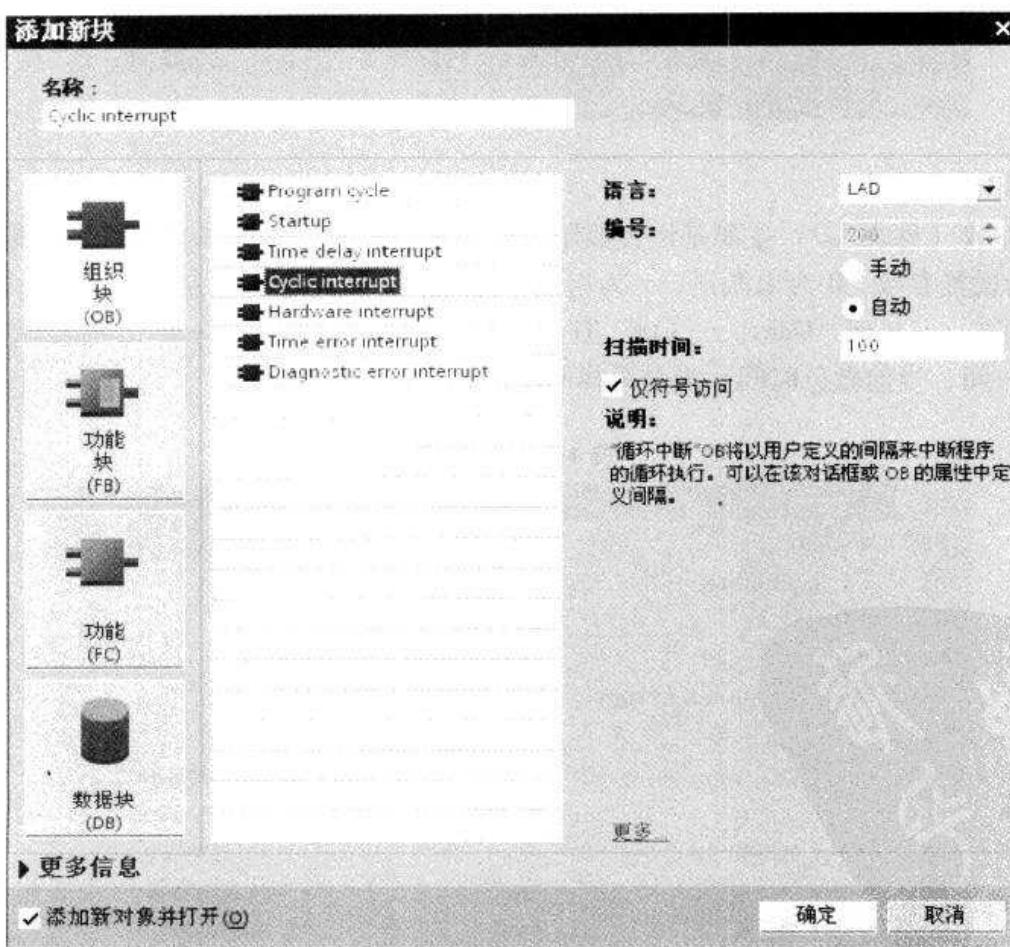


图 4-21 添加时间中断组织块 OB200

第三步，如图 4-22 所示从指令树中扩展指令处找到相应的 PID _ Compact 指令。



图 4-22 PID_Compact 指令

第四步，如图 4-23 所示将 PID_Compact 指令放置于 OB200 时间中断块中，这里面用到了 PID_Compact [DB1] 背景数据块。

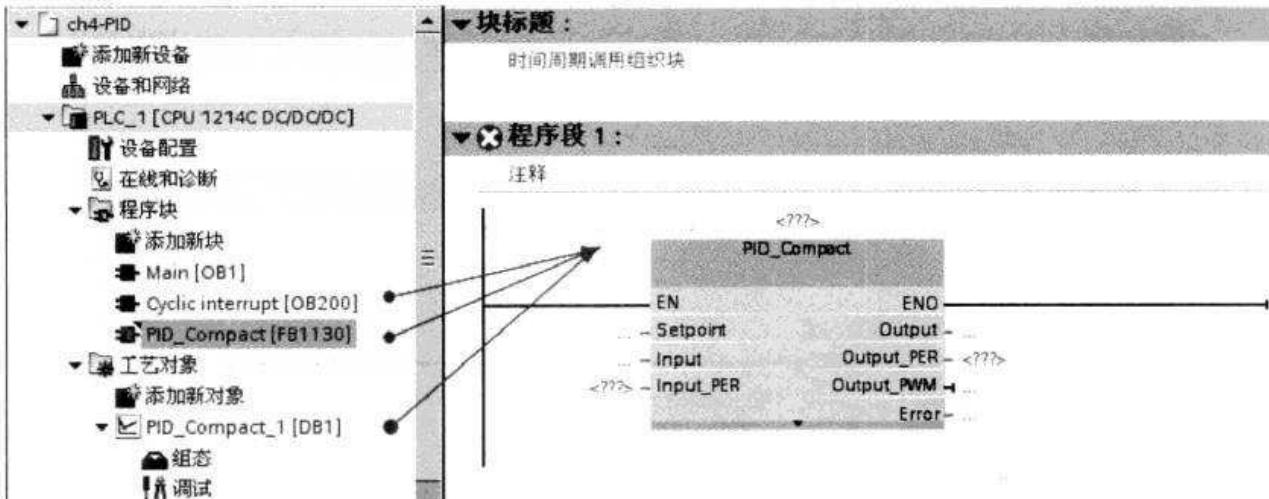


图 4-23 放在程序中的 PID 指令与程序块、工艺对象的关系

第五步，如图 4-24 所示完成完整的 OB200 块中关于 PID 指令的调用。从图 4-24 中可以看出，设定值为 25%，手动情况下模拟量输出为 10%。

第六步，如图 4-25 所示在主程序块 OB1 中对 PID 控制器的模式进行修改，即当 %I0.0 = OFF 时（选择开关为自动），将 PID 控制器模式 sRet.i_Mode 修改为 3（自动模式，见表 4-7）。



▼ 块标题：

时间周期调用组织块

▼ 程序段 1：

调用 PID 功能块

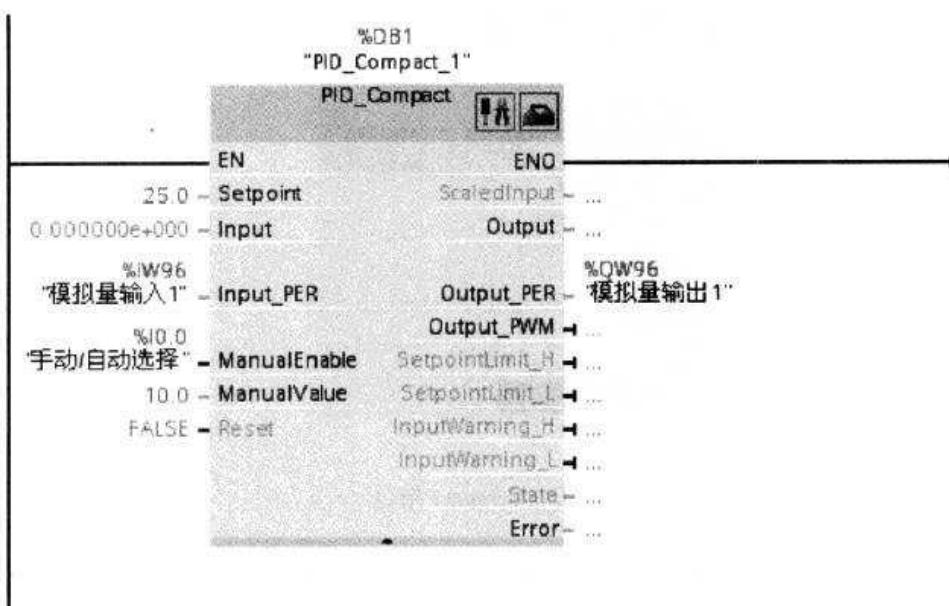


图 4-24 完整的 PID 程序 (OB200)

▼ 块标题：

主程序块

▼ 程序段 1：

当选择开关切换到自动时，将 PID 控制器模式改为 3，即自动模式

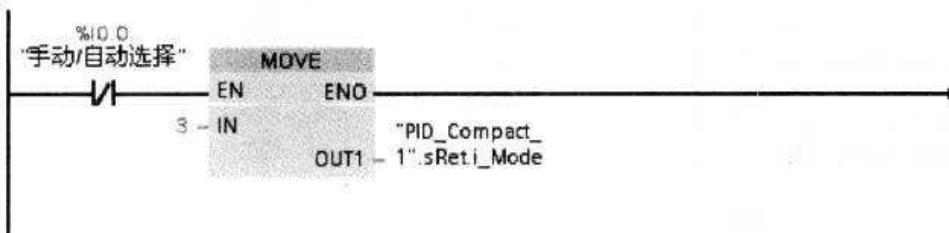


图 4-25 主程序块

4.2.4 在线模式下激活 PID 控制器

在 S7-1200 PLC 中可以使用优化步骤调整 PID 控制器以适应受控对象，一般情况下可选择以下两种方式来优化控制器：

(1) 启动调节

在启动调节中，将应用转折正切定理。该定理用于确定阶跃响应的时间常数。受控对象



的阶跃响应中存在一个转折点。对该转折点应用切线，使用该切线可确定过程参数延迟时间 (T_u) 和恢复时间 (T_g)。根据这些过程参数将确定优化的控制器参数。设定值与实际值之间必须至少相差 30% 才能使用转折正切定理确定参数。否则，将通过振荡过程和功能“运行中调节”自动确定控制器参数。

(2) 运行中调节

运行中调节使用振荡过程来优化控制器参数。通过该过程可间接确定受控对象的行为。增益因子将增大，直到其达到稳定限制且受控变量均匀振荡。控制器参数将基于振荡周期进行计算。

图 4-26 所示显示了分别采用转折正切定理（启动调节）和振荡过程（运行中调节）时受控对象的阶跃响应。

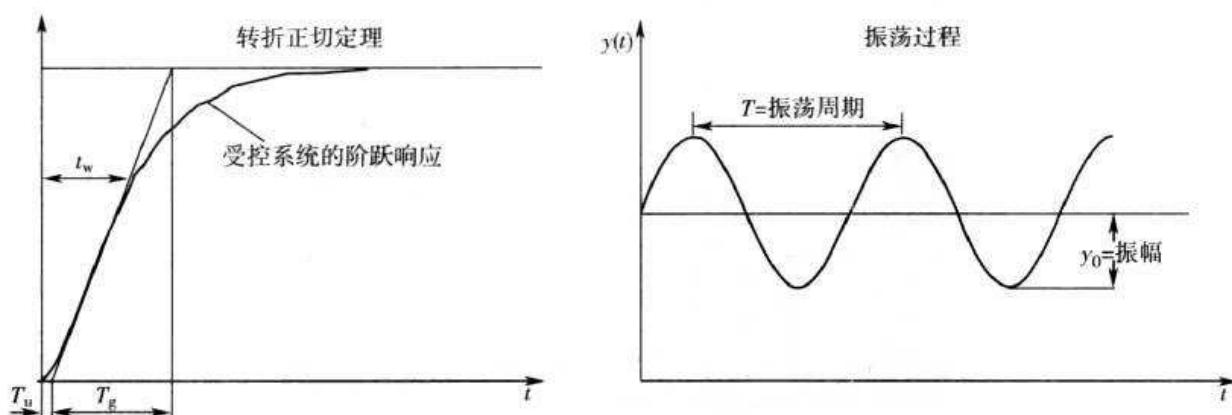


图 4-26 采用转折正切定理（启动调节）和振荡过程（运行中调节）时受控对象的阶跃响应

S7-1200 PLC 的调试面板主要分图 4-27 所示的调试面板控制区和趋势显示窗口、图 4-28

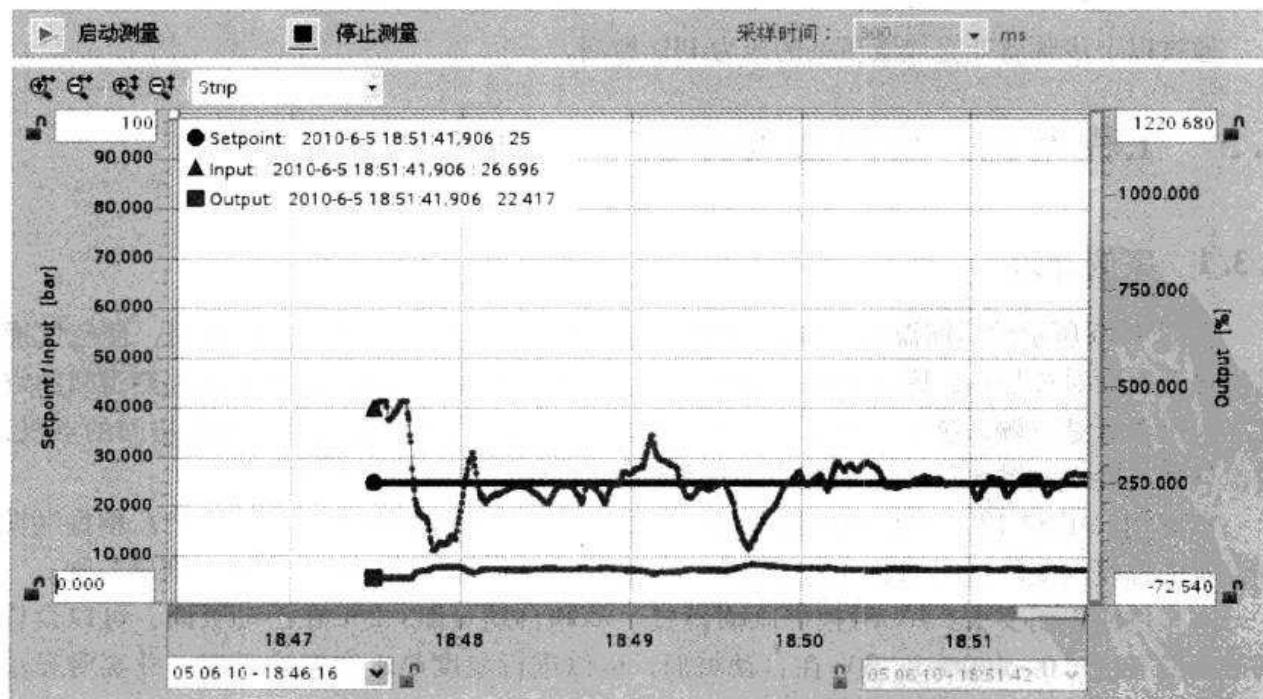


图 4-27 调试面板控制区和趋势显示窗口



图解西门子 S7-1200 PLC 入门到实践

所示的优化区和图 4-29 所示的当前值显示（即液压站设定压力、实际压力和变频器输出频率百分比）。

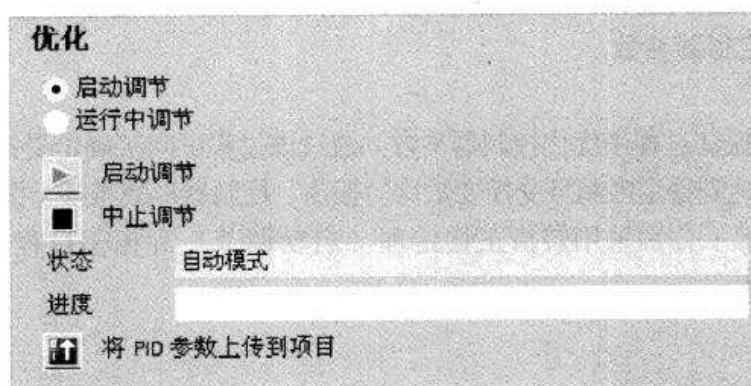


图 4-28 优化区

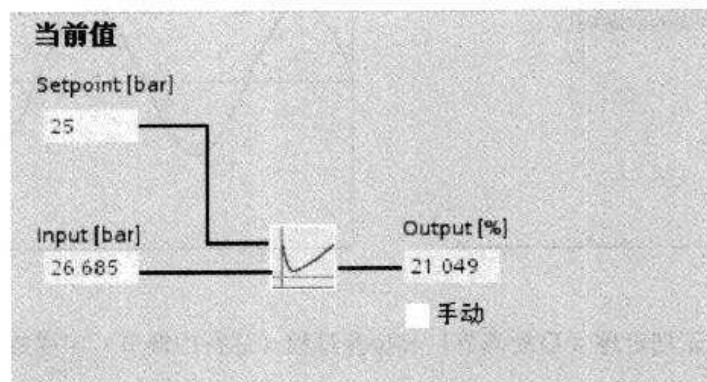


图 4-29 当前值显示（液压站设定压力、实际压力和变频器输出频率百分比）

通过以上步骤就能完成液压站的压力 PID 控制。

4.3 工程案例：房间温度控制

4.3.1 案例介绍

在图 4-30 所示的房间温度控制示意中，风机盘管主要依靠风机的强制作用，使空气通过加热器表面时被加热，因而强化了散热器与空气间的对流换热器，能够迅速加热房间的空气。风机盘管是空调系统的末端装置，其工作原理是机组内不断的再循环所在房间的空气，使空气通过冷水（热水）盘管后被冷却（加热），以保持房间温度的恒定。

本案例采用 S7-1200 PLC 和触摸屏进行控制，其工作原理如图 4-31 所示，PLC 根据触摸屏设置的温度与实际房间的温度进行比较，并进行 PID 运算，将运算结果输出到调节阀。

本案例的控制要求：1）触摸屏画面设置手动和自动页面；2）在手动页面，可以设置调节阀的开度（0 ~ 100%）；3）在自动页面，可以进行温度显示和温度设置，并实时显示调节阀的开度值；4）能进行缺水报警。

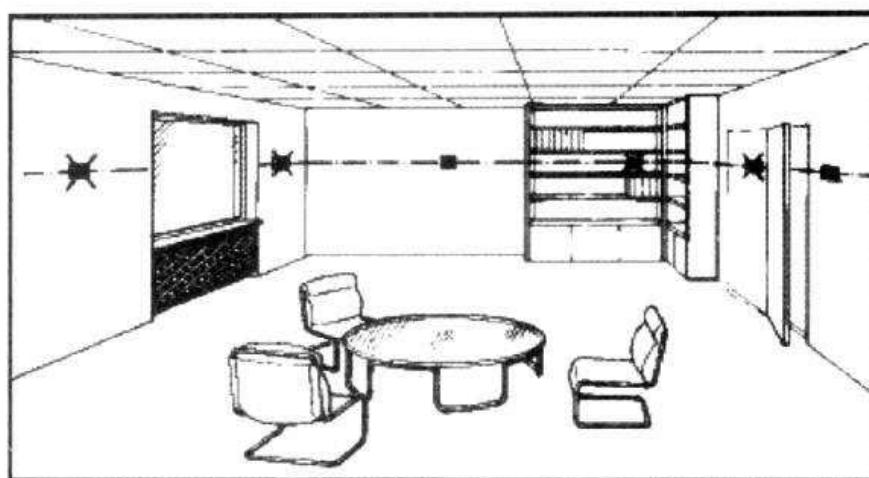


图 4-30 房间温度控制示意

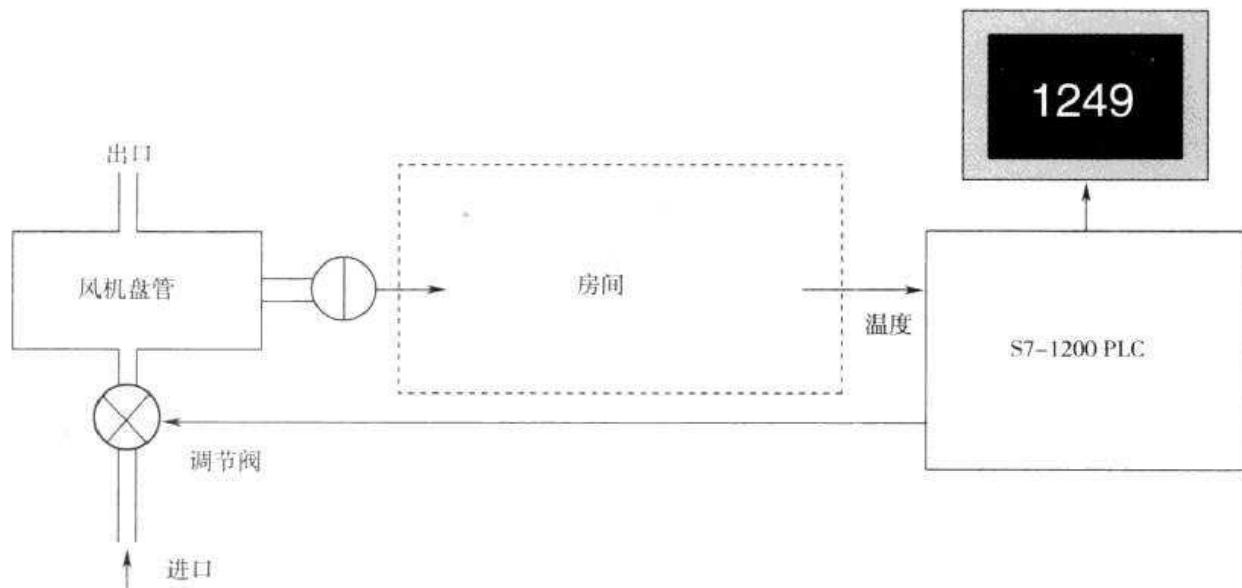


图 4-31 房间温度控制的工作原理

4.3.2 硬件设计

1. PLC 的设备与网络

首先根据案例要求，进行相应设备与网络定义，如图 4-32 所示。它具体包括 PLC_1

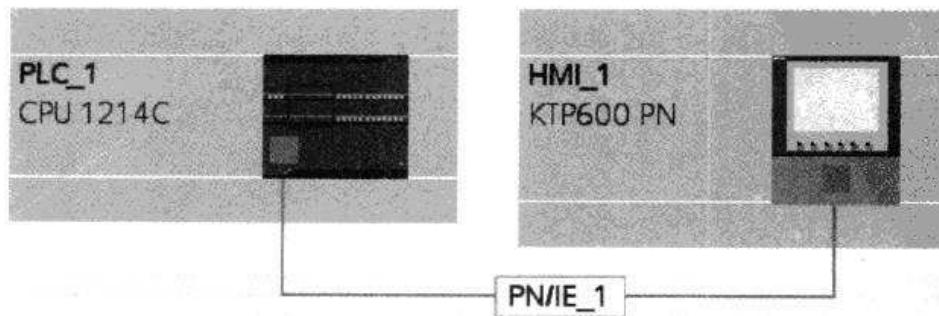


图 4-32 设备与网络定义



(CPU1214C) 和 HMI_1 (KTP600PN)，其网络连接为 PN/IE_1。

从图 4-31 所示的房间温度控制的工作原理中可以知道，S7-1200 PLC 的硬件组态包括 CPU 模块和模拟量模块（见图 4-33），其中模拟量模块为 SM1234，即 AI4/AQ2（4 模拟量输入/2 模拟量输出），以连接输入温度传感器、输出调节阀模拟量等。

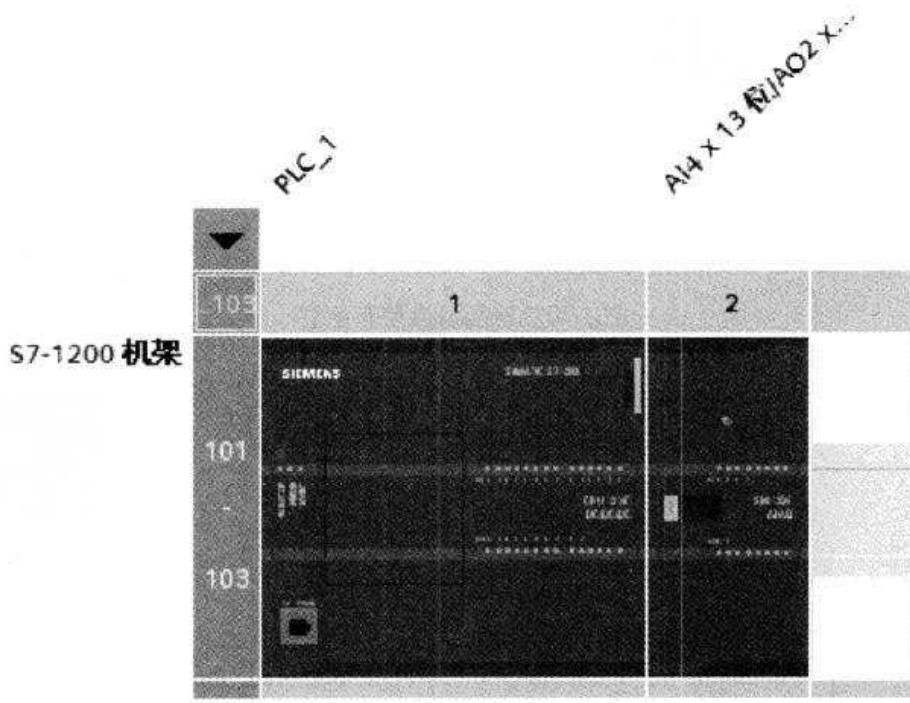


图 4-33 PLC 硬件组态

模拟量模块的 I/O 地址定义如图 4-34 所示。

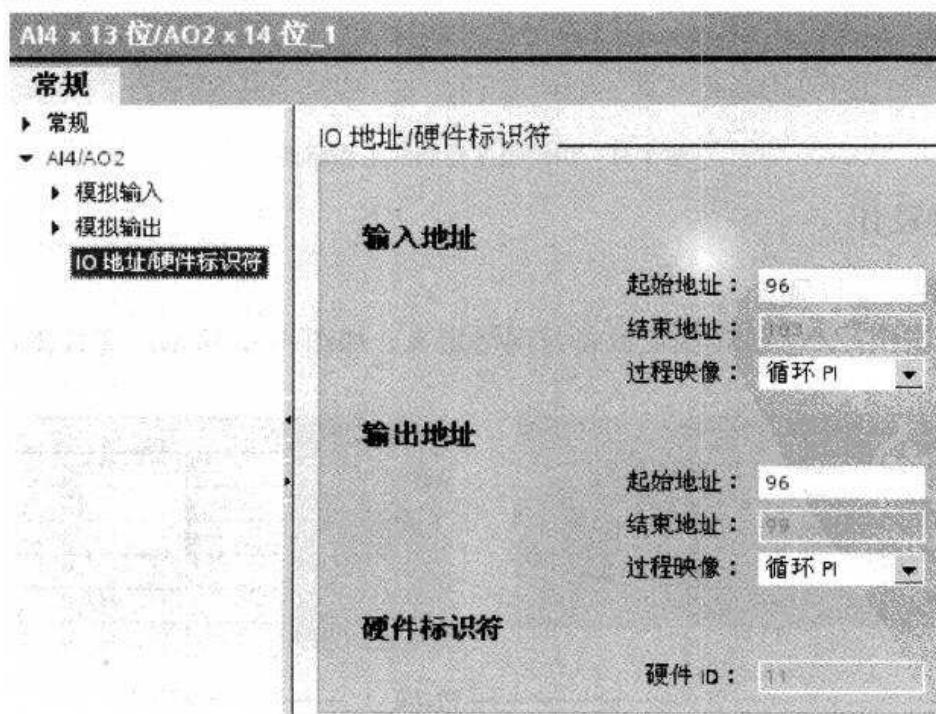


图 4-34 I/O 地址定义



2. 温度传感器的选型与 PLC 模拟量模块的配置

由于 S7-1200 PLC 没有专用的温度扩展模块，因此往往都会采用一体化的温度传感器，即包含温度传感部分和变送部分，一般为 24V 供电、二线制、连接热电阻或热电偶、输出 4~20mA，如图 4-35 所示为一体化温度传感器的外观。

电阻式一体化温度传感器的工作原理是，它是先通过确认阻值的不同计算出当前的温度，再根据热电阻的量程变送输出对应的标准信号值（4~20mA），即：温度变化→热电阻阻值变化→一体化温度传感器进行计算→输出 4~20mA 信号。

举个例子：Pt100 的量程为 -199.9~600.0℃。一体化温度传感器把这个转化为标准信号后，对应关系为：4mA 就是 -199.9℃、20mA 就是 600.0℃。因此，通过确认变送器输出的电流大小就可以知道当前的温度值。

图 4-36 所示为温度传感器接入到模拟量通道 0 的硬件组态。在本案例中，为了控制上的便利，定义该一体化温度传感器的量程为 0~100.0℃，即 4mA 就是 0℃、20mA 就是 100.0℃。根据 PLC 读入的数据可以知道，0mA 对应 0、20mA 对应 27648，那么 4mA 对应 5529，这个具体会在编程时用到。



图 4-35 一体化温度传感器的外观

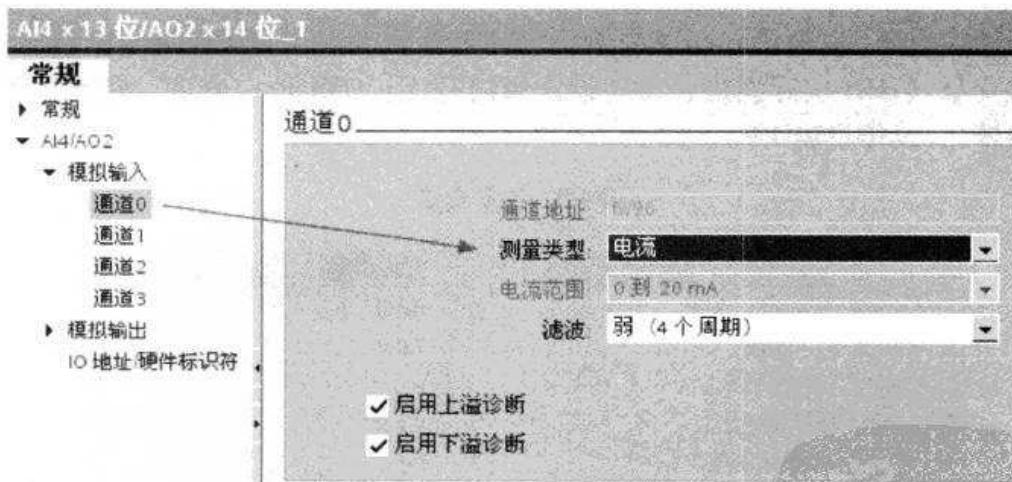


图 4-36 温度传感器接入到模拟量通道 0 的硬件组态

3. 调节阀的选型与 PLC 模拟量模块的配置

图 4-37 所示为调节阀外观，该调节阀能够接收 0~10V 信号来进行开度调节，其中 10V 代表 100% 开度，0V 表示 0% 开度。

与调节阀相对应的模拟量输出的硬件组态如图 4-38 所示，其选用通道 0，模拟量输出类型为电压，范围为 +/ -10V，并启用短路诊断。

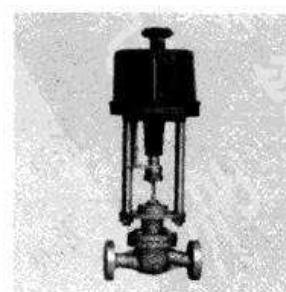


图 4-37 调节阀外观

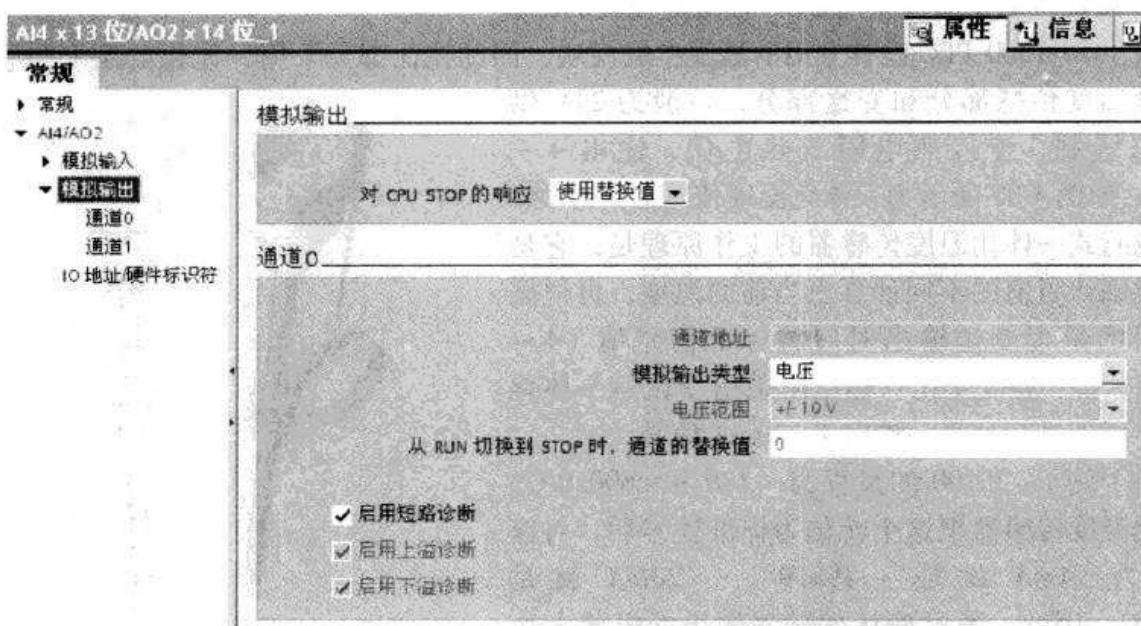


图 4-38 模拟量输出的硬件组态

4.3.3 软件编程

1. PLC 的变量定义

房间温度控制案例会用到模拟量输入 1、模拟量输出 1、手动/自动切换（现场开关）、缺水故障等硬件 I/O，也会用到程序中所要求的 PID 块变量（PID 设定温度和阀门开度）、触摸屏设置变量（HMI 实际温度、HMI 设定阀门开度、HMI 设定温度、缺水故障报警字、缺水故障报警），具体见表 4-9。

表 4-9 变量表格

| | 名称 | 数据类型 | 地址 |
|----|-----------|------|--------|
| 1 | 手动/自动选择 | Bool | %IO.0 |
| 2 | 缺水故障 | Bool | %IO.1 |
| 3 | 模拟量输入1 | Word | %IW96 |
| 4 | 模拟量输出1 | Word | %QW96 |
| 5 | PID设定温度 | Real | %MD0 |
| 6 | 阀门开度 | Real | %MD4 |
| 7 | HMI实际温度 | Real | %MD8 |
| 8 | HMI设定阀门开度 | Real | %MD12 |
| 9 | HMI设定温度 | Word | %MW20 |
| 10 | 缺水故障报警字 | Word | %MW22 |
| 11 | 缺水故障报警 | Bool | %M23.0 |

2. 添加 PID 工艺对象

按照上节内容添加房间温度控制 PID 工艺对象，需要注意的是，控制器类型为“温度”，同时要注意控制温度会有两种情况，即夏天降温为反作用、冬天加热为正作用，所以需要恰当地选择“取反 PID 控制器输出”选项（见图 4-39）。

第4章 S7-1200 PLC 的 PID 控制



基本参数

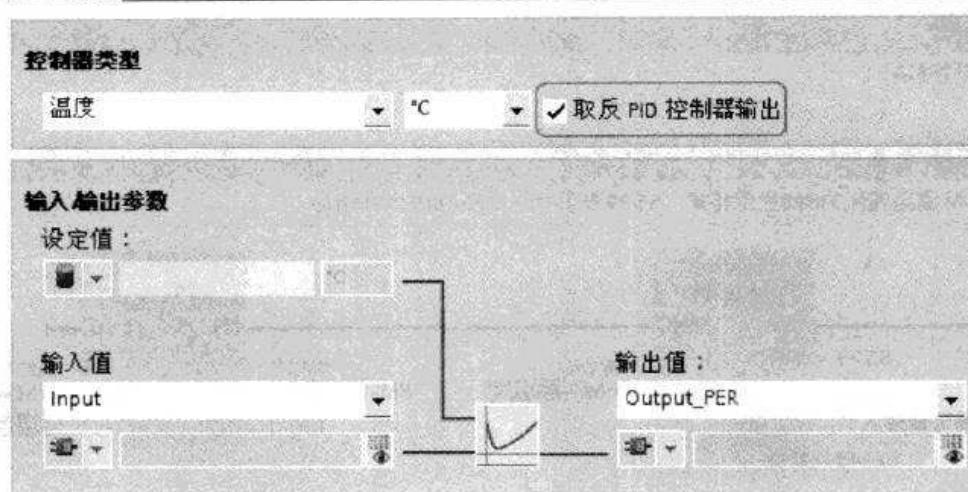


图 4-39 添加 PID 工艺对象

由于本案例选用的模拟量输入为 0 ~ 100°C，因此需要完成图 4-40 所示的输入标定。

输入标定

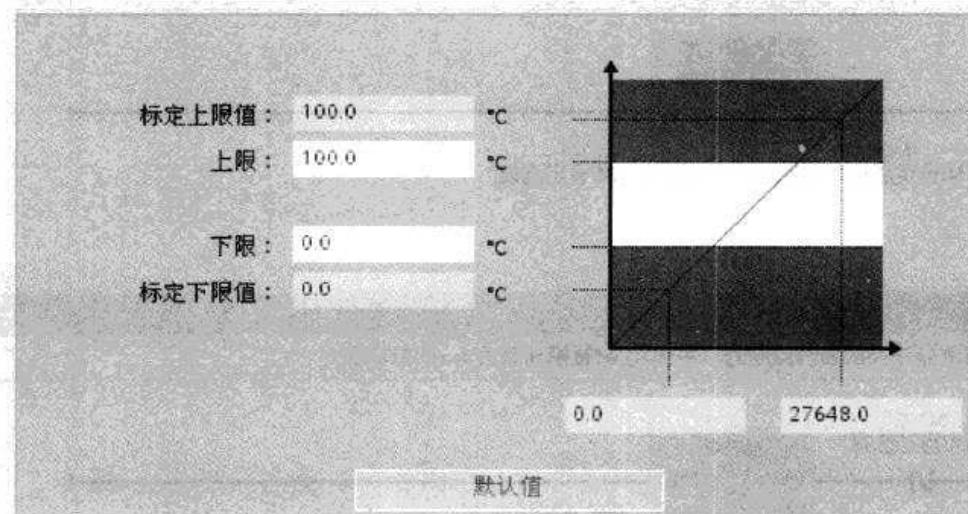


图 4-40 PID 对象的输入标定

3. PLC 程序编程

房间温度控制的 PLC 程序结构如图 4-41 所示，具体包括 OB1（主程序）、OB200（循环程序模块）、FB1130（PID 块）、DB1（PID 工艺对象数据块）。

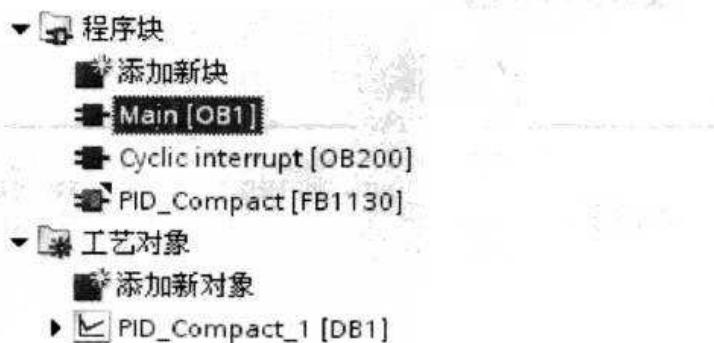


图 4-41 PLC 程序结构



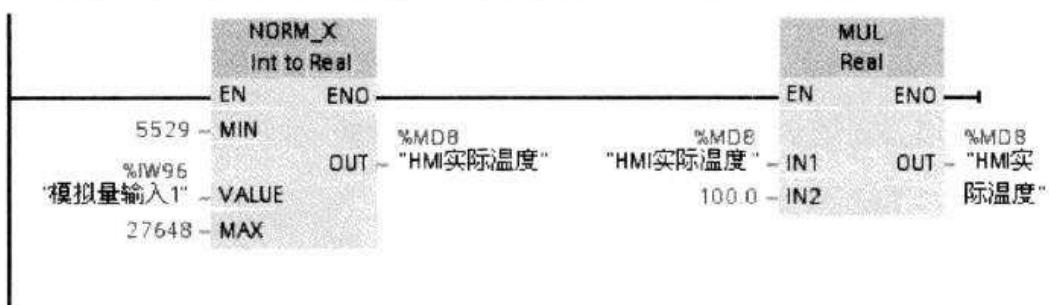
OB1 主程序如图 4-42 所示。

▼ 块标题：

主程序块

▼ 程序段 1：

实际温度转化为HMI显示格式（5529表示4mA;27648表示20mA）



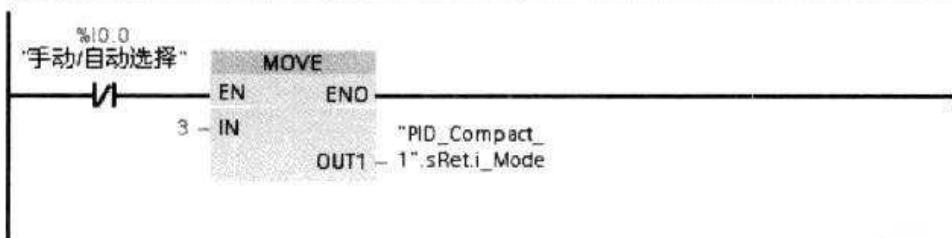
▼ 程序段 2：

设定温度值转为PID设定温度值格式



▼ 程序段 3：

当选择开关切换到自动时，将PID控制器模式改为3，即自动模式



▼ 程序段 4：

将调节阀开度转为百分比

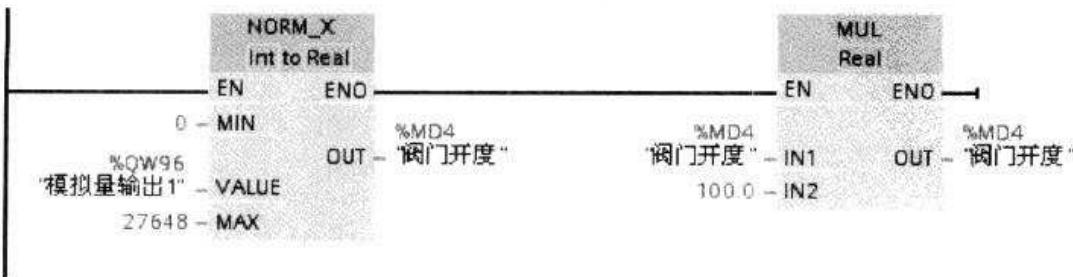


图 4-42 OB1 主程序

▼ 程序段 5：

缺水故障

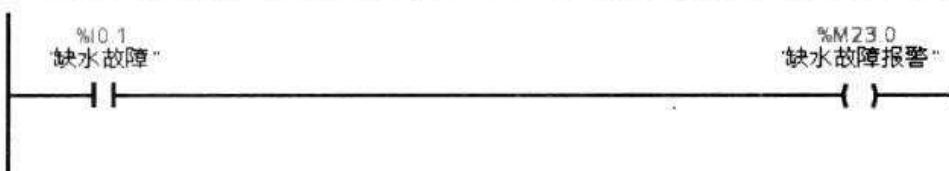


图 4-42 OB1 主程序（续）

OB200 为时间循环中断，其属性如图 4-43 所示，梯形图编程如图 4-44 所示。

循环中断

循环时间 (ms) : 100

相移 (ms) : 0

图 4-43 OB200 的循环中断属性

▼ 块标题：

时间周期调用组织块

▼ 程序段 1：

调用PID功能块

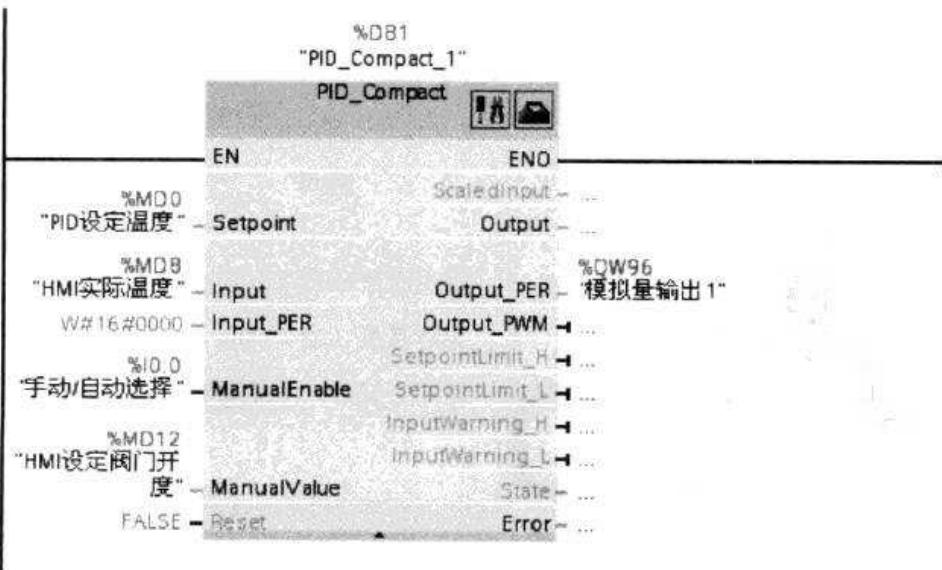


图 4-44 OB200 的梯形图编程



4. 触摸屏界面设计

房间温度控制的触摸屏共分 5 个画面（见图 4-45），通过建立模板及其事件定义（见图 4-46），可以在 5 个画面之间相互切换。

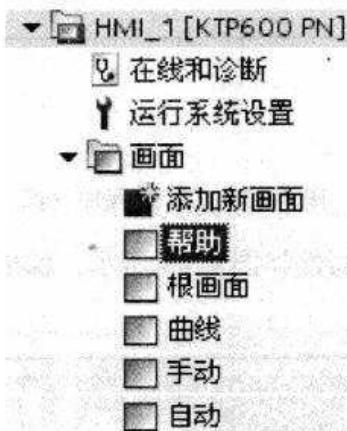


图 4-45 触摸屏画面



图 4-46 触摸屏的模板

(1) 主画面

主画面的画面设计如图 4-47 所示。

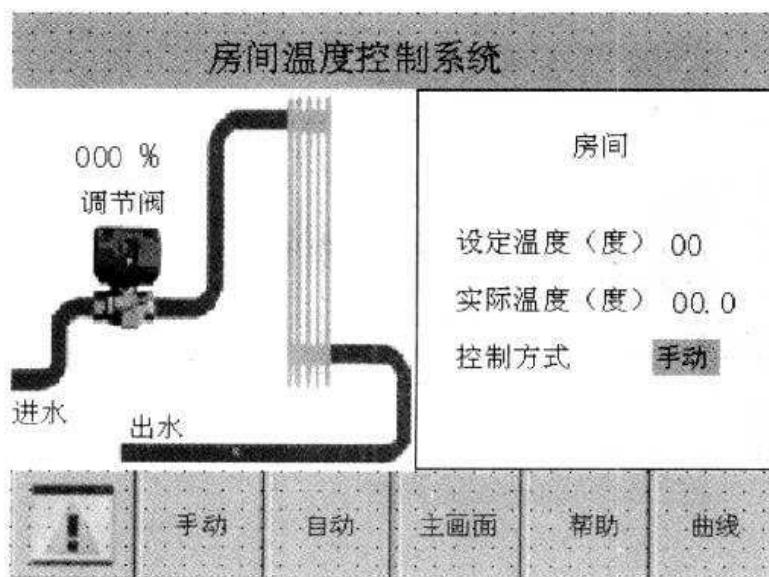


图 4-47 主画面的画面设计

第4章 S7-1200 PLC 的 PID 控制



图 4-48 所示为调节阀变量显示常规设置，能实时显示在自动情况下 PID 输出的调节阀值，也能显示在手动情况下的调节阀设定值。

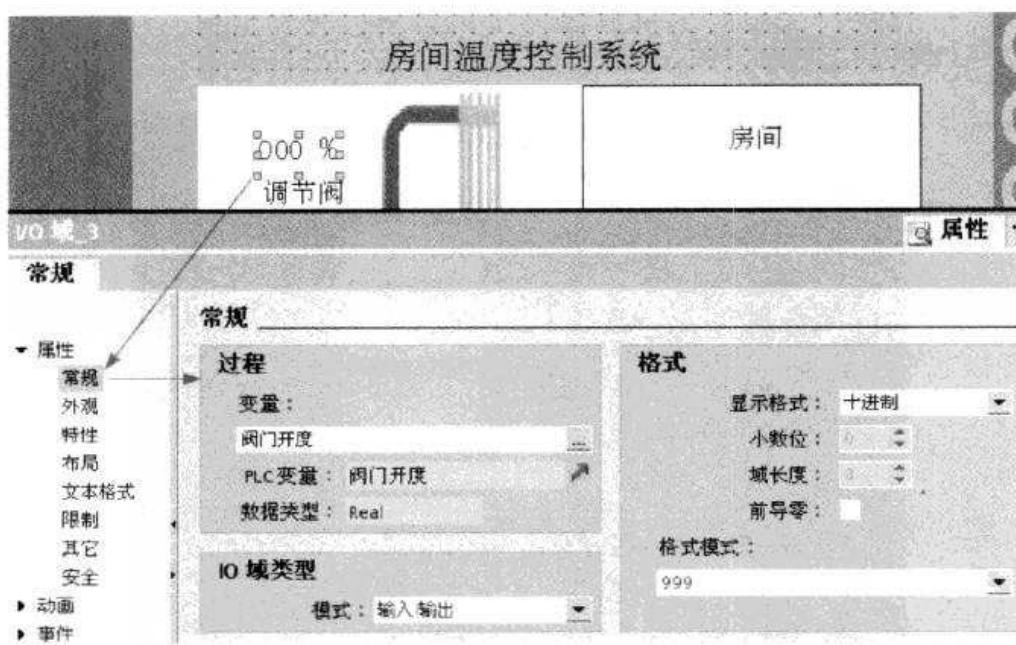


图 4-48 调节阀变量显示常规设置

在图 4-47 中，“设定温度（度） 00 ”显示的变量为“HMI 设定温度”，而“实际温度（度） 00.0 ”显示的变量为“HMI 实际温度”。

而在主画面 4-47 中所显示的“控制方式”控制方式“手动”中带阴影部分实际上是“手动”和“自动”两个显示变量的叠加，并采用“可见性”动画进行组态，即当% I0.0 = ON 时显示“手动”，相反显示“自动”，具体动画组态如图 4-49 所示。图 4-49 中显示了两个文本，当分别定义好动态后，需要将两个文本放置在同一个位置中。

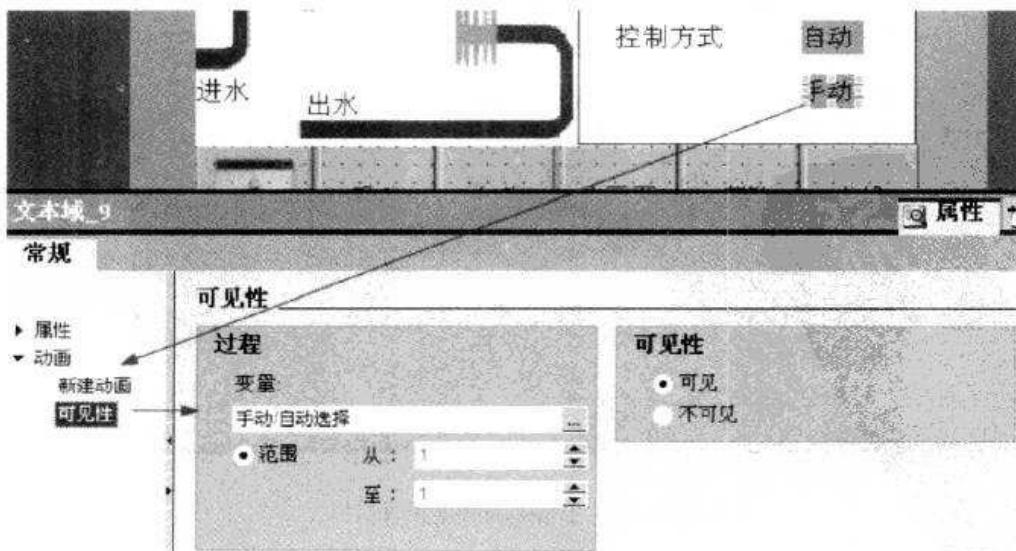


图 4-49 动画组态



(2) 自动画面

图 4-50 所示为自动画面的画面设计，图 4-50 中的 KP 增益设定等 3 个值分别对应为 DB1 中的 PID _ Compact _ 1 _ sRet _ r _ Ctrl _ Gain、PID _ Compact _ 1 _ sRet _ r _ Ctrl _ Ti、PID _ Compact _ 1 _ sRet _ r _ Ctrl _ Td。

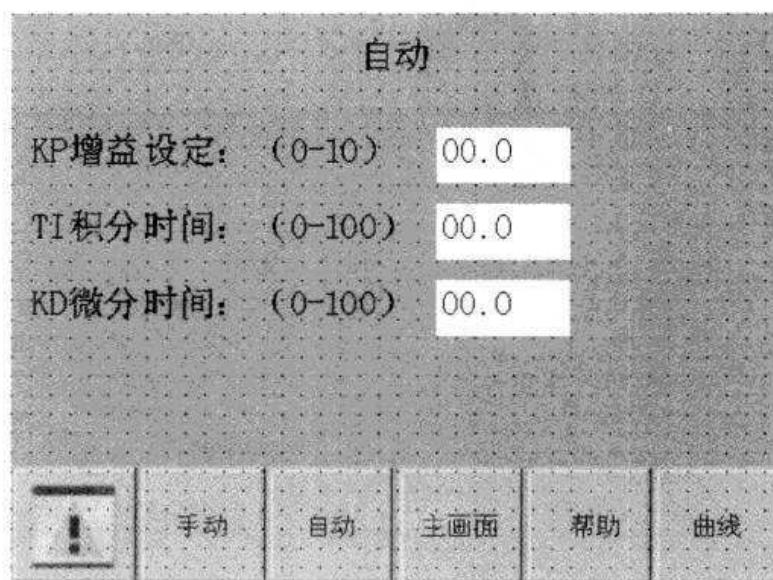


图 4-50 自动画面的画面设计

(3) 手动画面

图 4-51 所示为手动画面的画面设计，图 4-51 中的调节阀开度为“HMI 设定阀门开度”变量。

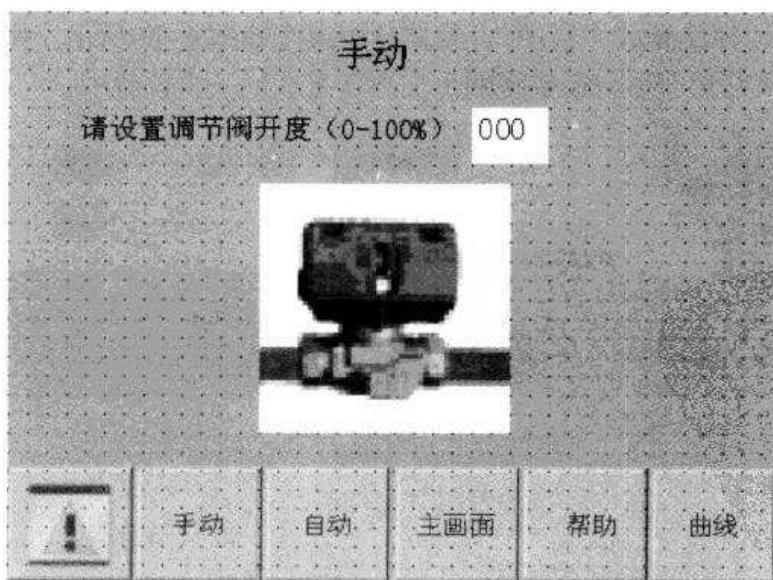


图 4-51 手动画面的画面设计

(4) 曲线画面

图 4-52 所示为曲线画面的画面设计。

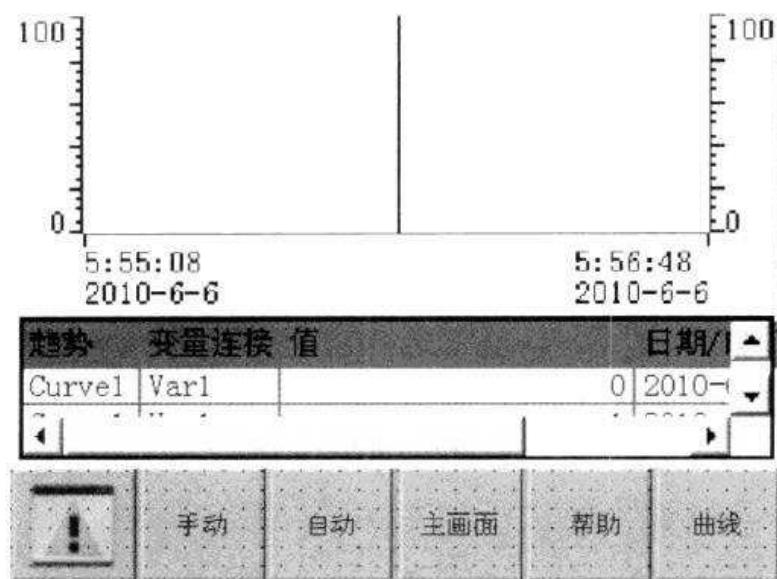


图 4-52 曲线画面的画面设计

曲线的趋势组态如图 4-53 所示，设置了三条曲线，分别是实际温度、阀门开度和设定温度，并用不同的颜色来进行表示，非常直观而形象地表达了 PID 房间温度控制的过程和效果。

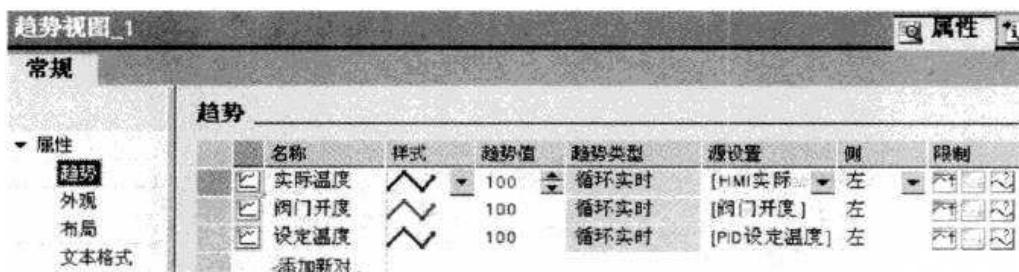


图 4-53 曲线趋势的组态

对于趋势的定义，包括时间轴（见图 4-54）、左侧数值轴（见图 4-55）和右侧数值轴。

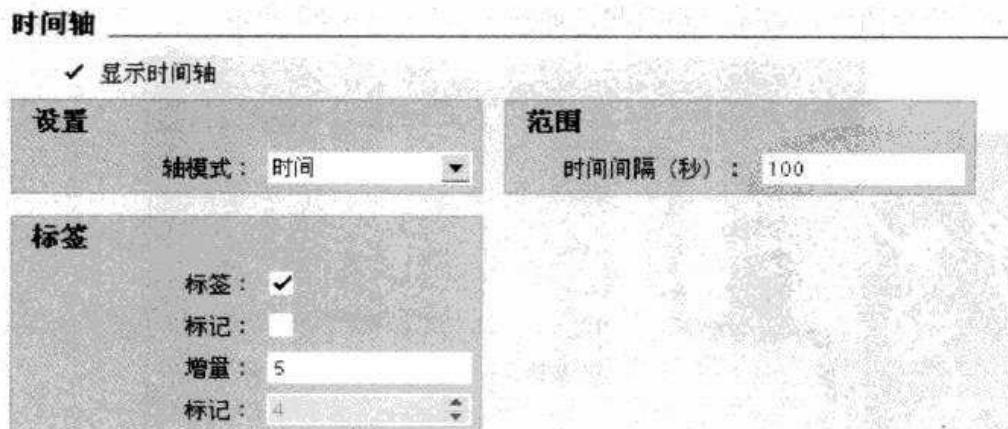


图 4-54 趋势时间轴的组态

(5) 帮助画面

图 4-56 所示为帮助画面的画面设计。



左侧数值轴

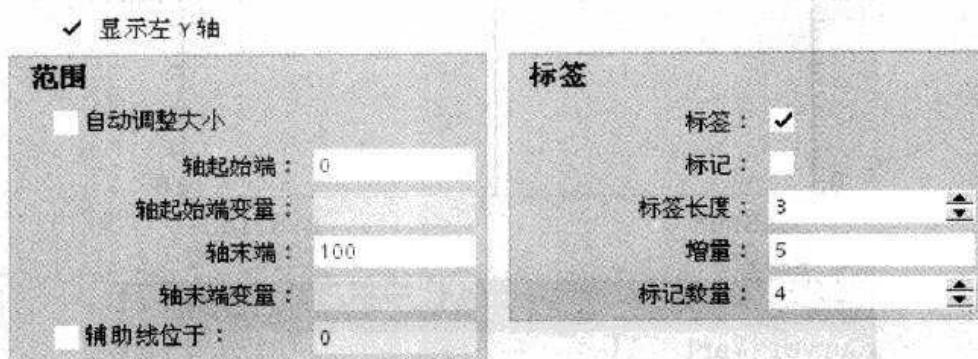


图 4-55 趋势左侧数值轴的组态

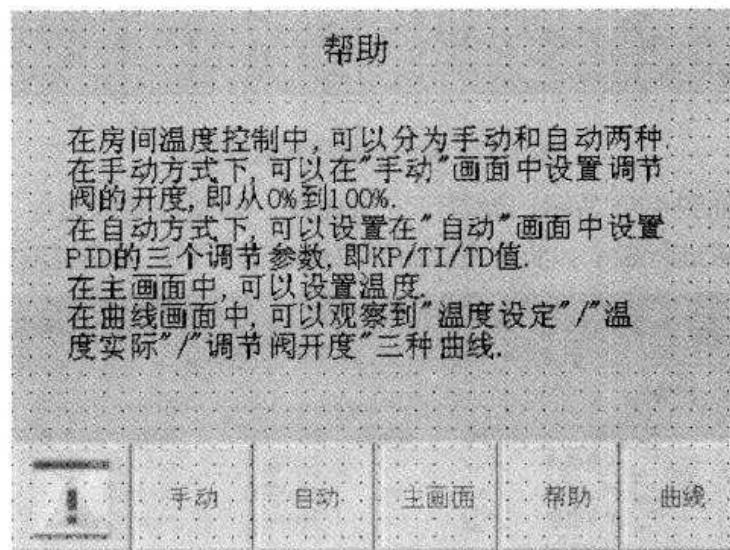


图 4-56 帮助画面的画面设计

4.3.4 画面测试

经实际安装与调试，实际的画面测试如图 4-57 ~ 图 4-62 所示。

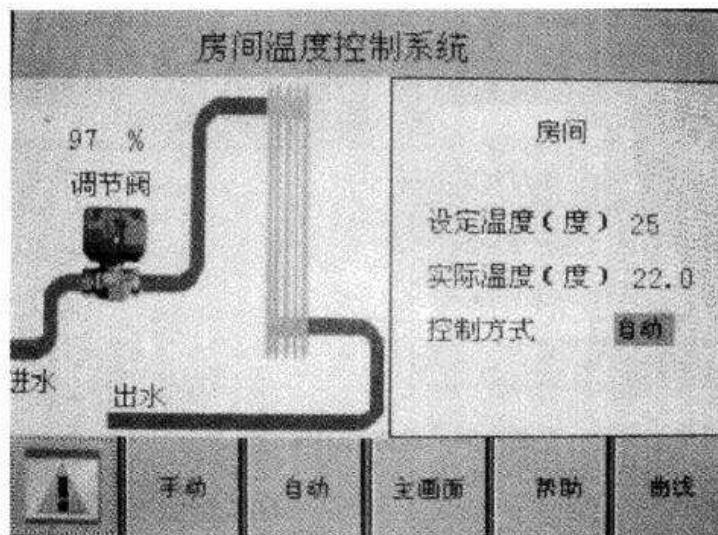


图 4-57 实际显示的主要画面

第4章 S7-1200 PLC 的 PID 控制

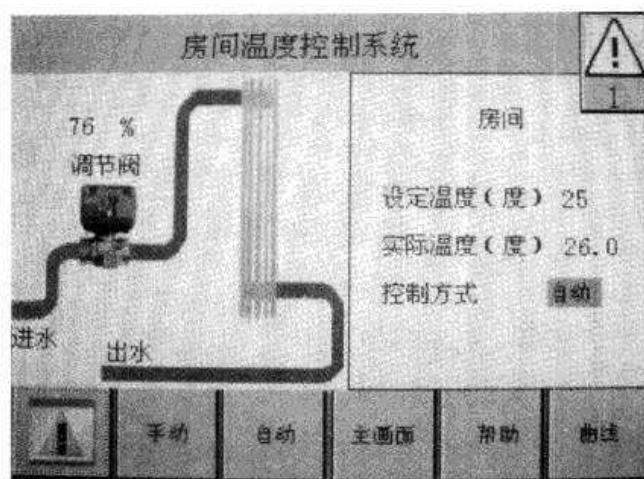


图 4-58 实际显示的主画面（有报警）

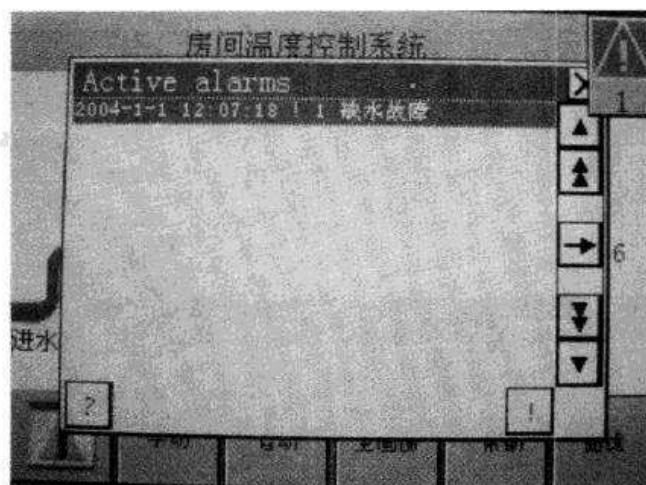


图 4-59 报警画面

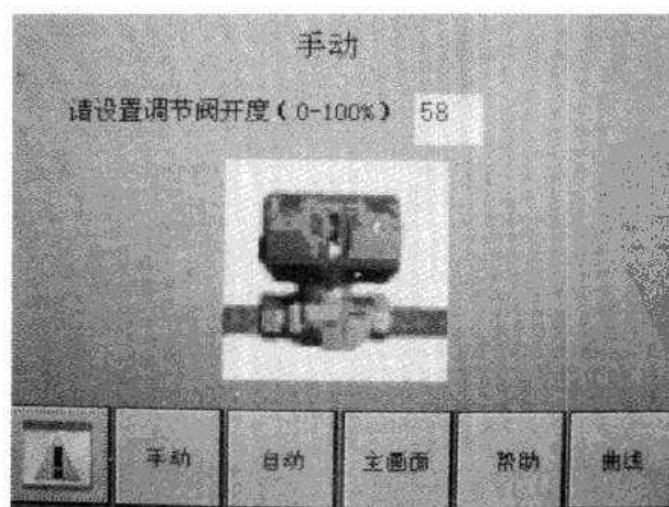


图 4-60 实际显示的手动画面

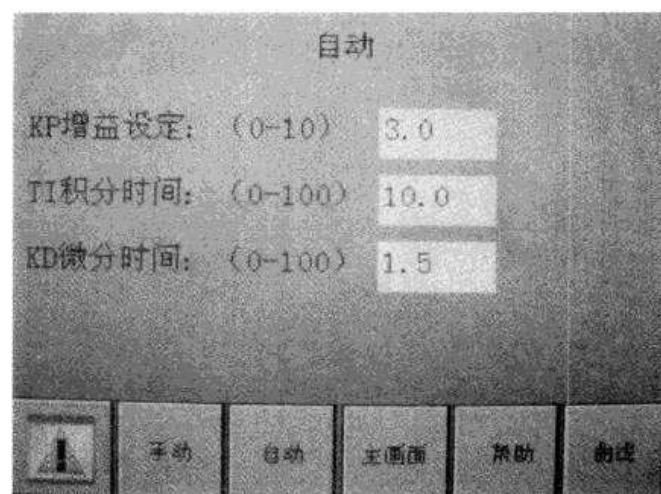


图 4-61 实际显示的自动画面

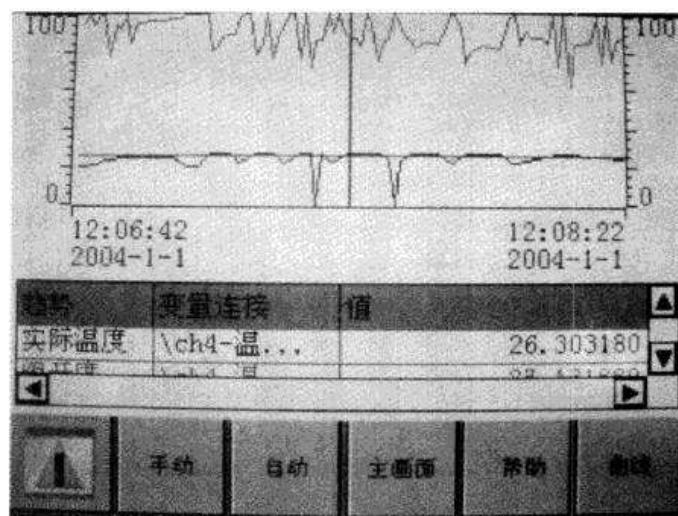
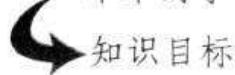


图 4-62 实际显示的趋势画面

第5章 S7-1200 PLC的 PWM控制

PWM是Pulse Width Modulation的简称。PWM控制是一种脉冲宽度调制技术，通过对一系列脉冲的宽度进行调制，来等效地获得所需要的波形（含形状和幅值）。PWM控制技术在逆变电路中应用比较广，应用的逆变电路绝大部分是PWM型。除此之外，PWM控制技术还可以应用在类似模拟量控制的场合中，比如它可以控制电动机的转速、阀门的位置等。

本章的学习目标如下：



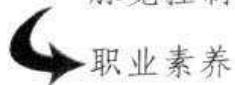
知识目标

了解PWM控制的基本原理；熟悉S7-1200 PLC中PWM控制的硬件基础；熟悉PWM与PTO的配置差异。



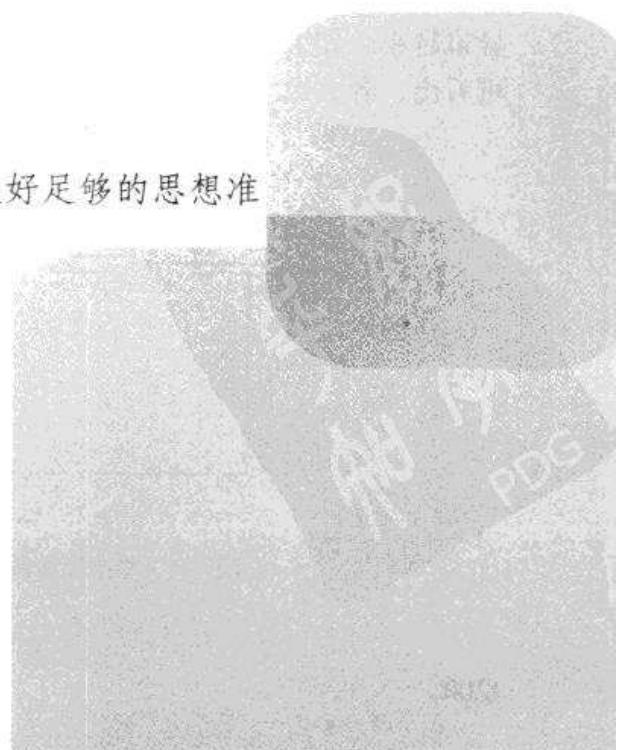
能力目标

能在TIA软件中对PWM输出进行硬件配置，并能通过触摸屏对它进行使能和脉宽控制。



职业素养目标

能了解PLC应用的行业特点，为服务行业做好足够的思想准备。





5.1 背景知识介绍：PWM 控制

5.1.1 PWM 控制的基本概念

PWM 是 Pulse Width Modulation 的简称。PWM 控制是一种脉冲宽度调制技术，通过对一系列脉冲的宽度进行调制，来等效地获得所需要的波形（含形状和幅值）。PWM 控制技术在逆变电路中应用比较广，应用的逆变电路绝大部分是 PWM 型。除此之外，PWM 控制技术还可以应用在类似模拟量控制的场合中，比如它可以控制电动机的转速、阀门的位置等。

1. PWM 控制的基本原理

冲量相等而形状不同的窄脉冲加在具有惯性的环节上时，其效果基本相同。冲量指窄脉冲的面积。效果基本相同，是指环节的输出响应波形基本相同。低频段非常接近，仅在高频段略有差异（见图 5-1）。

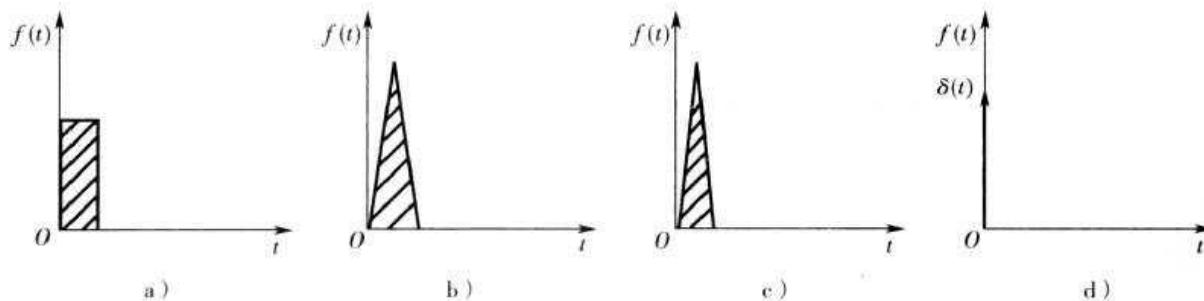


图 5-1 形状不同而冲量相同的各种窄脉冲

2. 面积等效原理

分别将图 5-1 所示的电压窄脉冲加在一阶惯性环节（R-L 电路）上，如图 5-2a 所示。其输出电流 $i(t)$ 对不同窄脉冲时的响应波形如图 5-2b 所示。从波形可以看出，在 $i(t)$ 的上升段， $i(t)$ 的形状也略有不同，但其下降段则几乎完全相同。脉冲越窄，各 $i(t)$ 响应波形的差异也越小。如果周期性地施加上述脉冲，则响应 $i(t)$ 也是周期性的。用傅里叶级数分解后可看出，各 $i(t)$ 在低频段的特性将非常接近，仅在高频段有所不同。

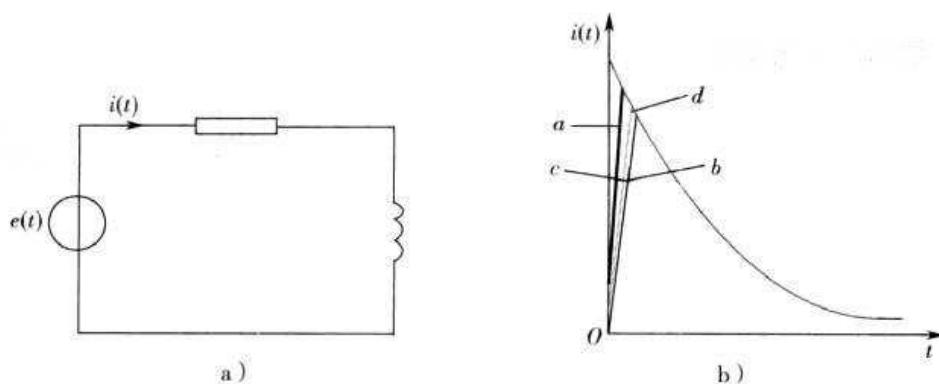


图 5-2 冲量相同的各种窄脉冲加在一阶惯性环节上的响应波形

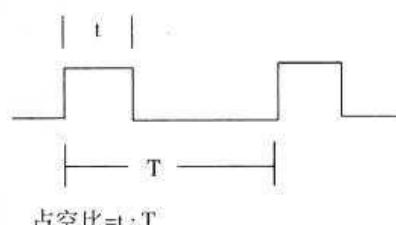


3. 占空比的定义

占空比就是 Duty Cycle (即负载周期)，其含义就是在一串理想的脉冲序列中（如方波），正脉冲的持续时间与脉冲总周期的比值。

例如：脉冲宽度为 $1\mu s$ ，信号周期为 $4\mu s$ 的脉冲序列占空比为 0.25。

占空比也可以理解为高电平所占周期时间与整个周期时间的比值，如图 5-3 所示。因此，方波的占空比为 50%，占空比为 0.1，说明正电平所占时间为 0.1 个周期。



5.1.2 S7-1200 PLC 的 PWM 应用

1. 硬件配置

PWM 是一种周期固定、脉冲宽度可以调节的脉冲输出，是数字量输出。在 S7-1200 PLC 中，它提供两个输出通道用于高速脉冲输出，分别可以组态为 PTO 或 PWM，关于 PTO 的功能将在后续项目中进行介绍。需要注意的是，当一个通道被组态为 PTO 时，就不能再作为 PWM 使用，反之亦然。

脉冲功能输出点占用情况见表 5-1。可以使用板载 CPU 输出，也可以使用可选的信号板输出。表 5-1 中列出了输出点编号（假定使用默认输出组态），如果更改了输出点编号，则输出点编号将为用户指定的编号。无论是在 CPU 上还是在连接的信号板上，PTO1/PWM1 都使用前两个数字输出，PTO2/PWM2 使用接下来的两个数字输出。请注意，PWM 仅需要一个输出，而 PTO 每个通道可选择使用两个输出。如果脉冲功能不需要输出，则相应的输出可用于其他用途。

表 5-1 脉冲功能输出点占用情况表

| 说 明 | 默认输出分配 | | |
|------|--------|------|------|
| | | 脉冲 | 方向 |
| PTO1 | 板载 CPU | Q0.0 | Q0.1 |
| | 信号板 | Q4.0 | Q4.1 |
| PWM1 | 板载 CPU | Q0.0 | — |
| | 信号板 | Q4.0 | — |
| PTO2 | 板载 CPU | Q0.2 | Q0.3 |
| | 信号板 | Q4.2 | Q4.3 |
| PWM2 | 板载 CPU | Q0.2 | — |
| | 信号板 | Q4.2 | — |

2. CTRL_PWM 指令调用

在 TIA 软件中使用 PWM 指令，可以直接从图 5-4 所示的“扩展指令”下的“脉冲”获得，即 CTRL_PWM 指令，其控制脉冲宽度可以表示为脉冲周期的百分之几到万分之几或 S7 analog 模拟量形式。

与其他指令相同，CTRL_PWM 指令调用也需要背景数据库的支持，如图 5-5 所示。

背景数据块中的 PWM 参数见表 5-2。

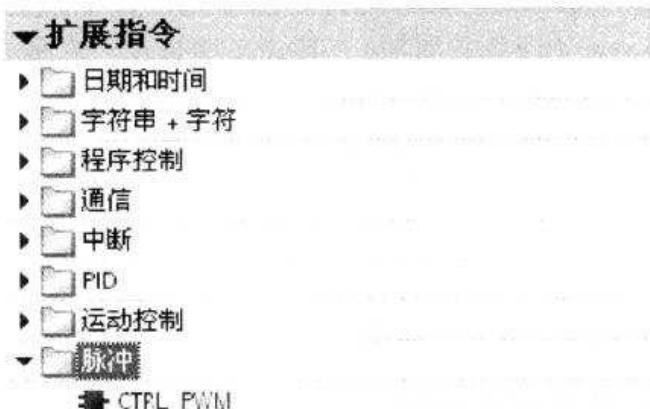


图 5-4 扩展指令下的 CTRL_PWM 指令

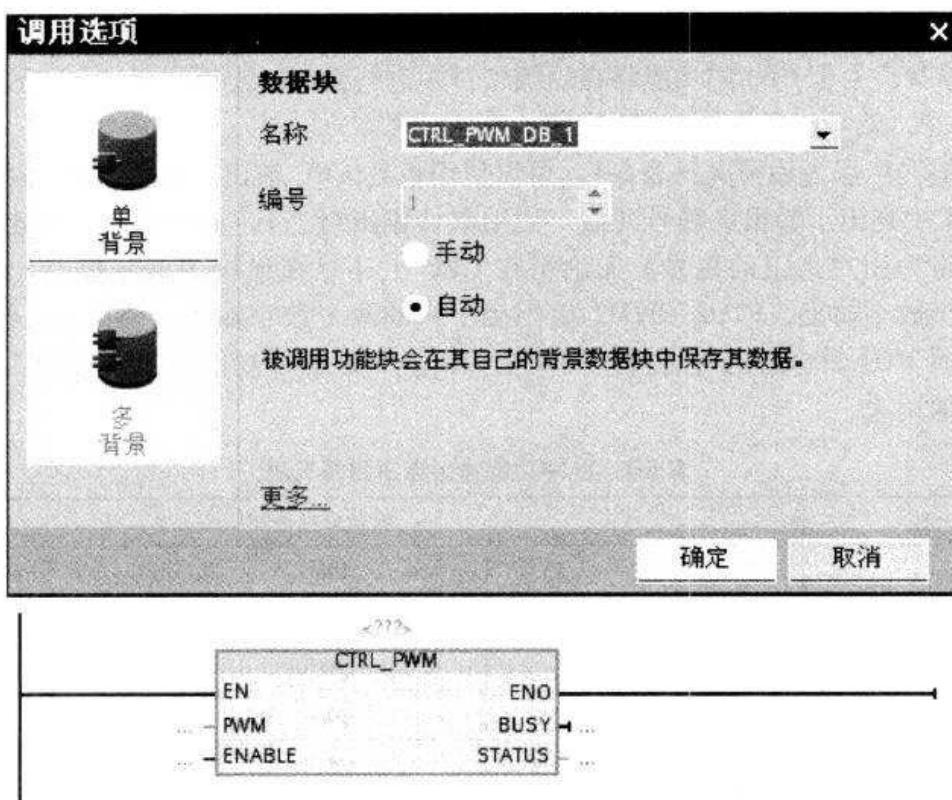


图 5-5 指令调用

表 5-2 PWM 参数

| 名称 | 数据类型 | 初始值 |
|------------|--------|-----------|
| 1 ▼ Input | | |
| 2 PWM | HW_PWM | W#16#0 |
| 3 ENABLE | Bool | False |
| 4 ▼ Output | | |
| 5 BUSY | Bool | False |
| 6 STATUS | Word | W#16#0000 |

第5章 S7-1200 PLC 的 PWM 控制



从图 5-5 中可以看出，当 EN 端变为 1 时，指令块通过 ENABLE 端使能或禁止脉冲输出，脉冲宽度通过组态好的 QW（输出字）来调节。对于 PWM1，默认位置是 QW1000；而对于 PWM2，默认位置是 QW1002。该位置的值控制脉冲宽度，并且在每次 CPU 从 STOP 切换到 RUN 模式时都会初始化为上面指定的“初始脉冲宽度”值。在运行期间更改该 QW 字值会引起脉冲宽度变化。当 CTRL_PWM 指令正在运行时，BUSY 位一直为 0。有错误发生时，ENO 端输出为 0，同时 STATUS 显示错误状态“80A1”（即硬件识别号非法）。

表 5-3 归纳了 CTRL_PWM 指令的 I/O 参数类型、数据类型和说明。

表 5-3 I/O 参数类型、数据类型和说明

| 参数 | 参数类型 | 数据类型 | 初 始 值 | 说 明 |
|--------|------|------|-------|--|
| PWM | IN | Word | 0 | PWM 标识符： 已启用的脉冲发生器的名称将变为“常量”（constant）变量表中的变量，并可用作 PWM 参数 |
| ENABLE | IN | Bool | | 1 = 启动脉冲发生器 0 = 停止脉冲发生器 |
| BUSY | OUT | Bool | 0 | 功能忙 |
| STATUS | OUT | Word | 0 | 执行条件代码 |

3. PWM 端口故障原因

在 PWM 功能硬件配置和软件编程中经常会出现如图 5-6 所示的错误。

| 设备概览 | | | | | |
|-----------|------|--------|---------|--------------------|--|
| 模块 | 插槽 | I 地址 | Q 地址 | 类型 | |
| 101 | | | | | |
| PLC_1 | 1 | | | CPU 1214C DC/DC/DC | |
| DI14/DO10 | 1.1 | 0..1 | 0..1 | DI14/DO10 | |
| AI2 | 1.2 | 64..67 | | AI2 | |
| | 1.3 | | | | |
| HSC_1 | 1.16 | | | 高速计数器 (HSC) | |
| HSC_2 | 1.17 | | | 高速计数器 (HSC) | |
| HSC_3 | 1.18 | | | 高速计数器 (HSC) | |
| HSC_4 | 1.19 | | | 高速计数器 (HSC) | |
| HSC_5 | 1.20 | | | 高速计数器 (HSC) | |
| HSC_6 | 1.21 | | | 高速计数器 (HSC) | |
| Pulse_1 | 1.32 | | 1000..1 | 脉冲发生器 (PTO/P) | |
| Pulse_2 | 1.33 | | 1002..1 | 脉冲发生器 (PTO/P) | |

图 5-6 硬件配置出错

出现该故障的原因有以下几种：

- 1) 在该程序中出现 Q0.0 或 Q0.2 的双重定义；
- 2) 端口号没有匹配；
- 3) 硬件配置没有完全正确。



5.2 技能训练：人机界面控制 PWM

5.2.1 控制要求

本案例需要在 HMI 上对 PWM 的端口输出进行控制，具体包括（见图 5-7）：

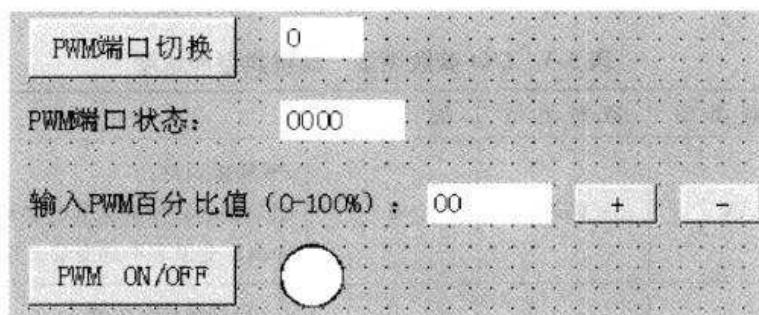


图 5-7 控制要求

- 1) 能通过按钮进行端口切换，并在触摸屏显示端口值（1 或 2）；
- 2) 能实时显示 PWM 端口状态（或者错误代码）；
- 3) 能通过键盘输入或者加减按钮输入 PWM 占空比的百分比值（范围为 0 ~ 100%）；
- 4) 能 ON/OFF PWM 功能。

5.2.2 硬件配置与软件编程

1. 硬件配置

在 TIA 软件中进入设备组态界面，选中 CPU，单击“属性”按钮，可以看到如图 5-8 所示的 PLC 常规属性。根据需要可以选择两个脉冲输出口，即 Pulse_1 和 Pulse_2。



图 5-8 PLC 的常规属性

第5章 S7-1200 PLC 的 PWM 控制



图 5-9 所示的脉冲输出常规属性，选择“启用”，即允许使用该脉冲发生器。

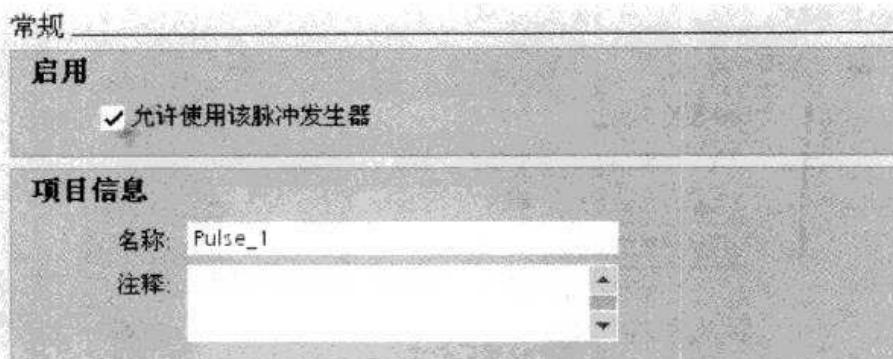


图 5-9 选择启用脉冲发生器

对于启用的脉冲发生器，可以定义如图 5-10 所示的脉冲选项参数，并首先定义脉冲发生器用作“PTO”或“PWM”，本案例选择“PWM”。

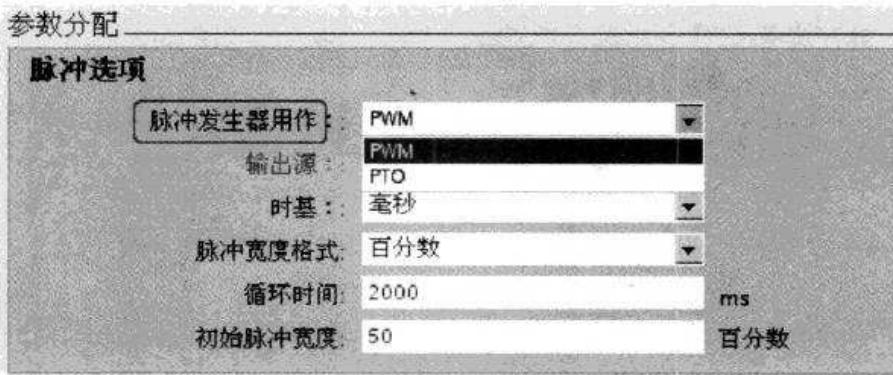


图 5-10 脉冲选项参数

接下来就会看到默认的输出源“板载 CPU 输出”，然后定义图 5-11 所示的时基，可以选择毫秒或微秒，本案例选择毫秒。

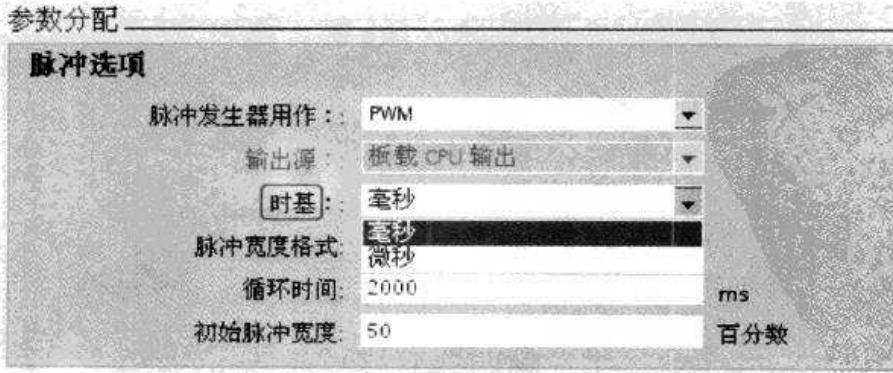


图 5-11 时基

脉冲宽度格式（见图 5-12）也是非常重要的参数，它可以选择为百分数、千分数、万



分数或 S7 模拟量格式，本案例选择百分数格式。

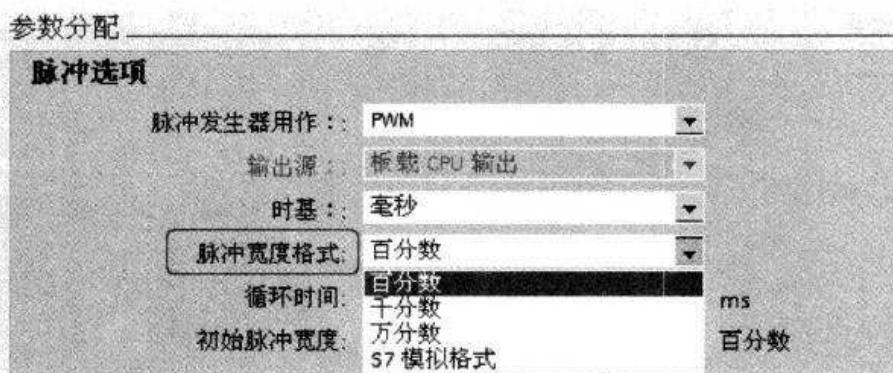


图 5-12 脉冲宽度格式

图 5-13 所示的循环时间表示脉冲的周期值，其单位即“时基”单位。

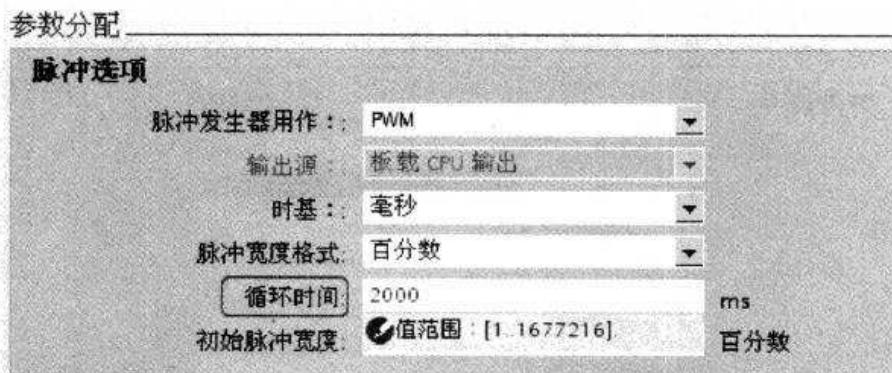


图 5-13 循环时间

图 5-14 所示为初始脉宽，如本案例选用百分比，则值范围为 0 ~ 100；如果选择 S7 模拟量格式，则 **值范围：[0..27648]**；依次类推。

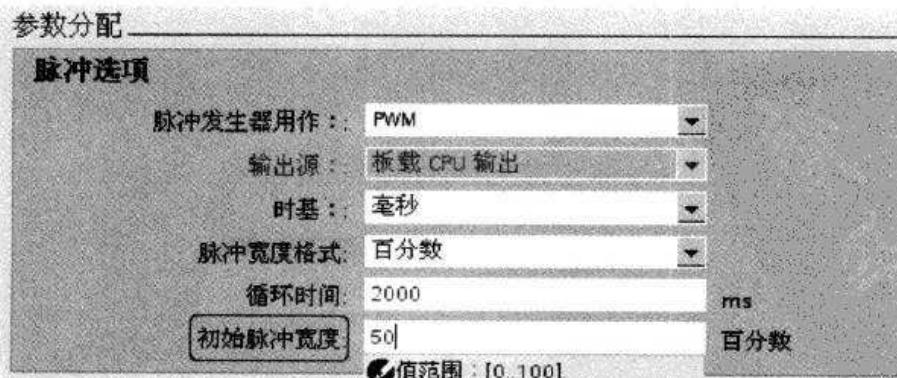


图 5-14 初始脉宽

对于 Pulse_1 来说，其硬件输出就是图 5-15 所示的默认脉冲输出 Q0.0。



图 5-15 硬件输出

最后就是 IO 地址/硬件标识符，如图 5-16 所示，可以采用默认地址，也可以进行修改，

值范围：0..1023

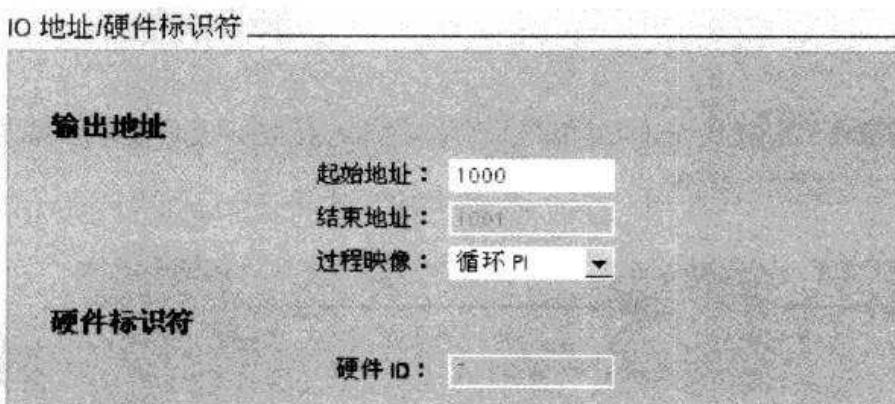


图 5-16 IO 地址/硬件标识符

对于 Pulse_2，可以按照以上相同步骤进行。

2. PLC 程序编程

根据训练要求，定义变量见表 5-4，其中包括脉冲 1 输出为 %QW1000、脉冲 2 输出为 %QW1002，触摸屏的按钮加、减信号为 %M0.1、%M0.3 等。

表 5-4 变量定义

| | 名称 | 数据类型 | 地址 |
|----|------------|------|---------|
| 1 | 脉冲1输出 | Word | %QW1000 |
| 2 | 脉冲2输出 | Word | %QW1002 |
| 3 | PWM控制忙 | Bool | %M0.0 |
| 4 | 按钮加 | Bool | %M0.1 |
| 5 | 按钮加信号 | Bool | %M0.2 |
| 6 | 按钮减 | Bool | %M0.3 |
| 7 | 按钮减信号 | Bool | %M0.4 |
| 8 | PWM控制使能 | Bool | %M1.0 |
| 9 | PWM端口选择 | Bool | %M1.1 |
| 10 | PWM脉冲宽度百分比 | Word | %MW10 |
| 11 | PWM控制状态 | Word | %MW12 |
| 12 | PWM端口 | Int | %MW14 |
| 13 | PWM端口号显示 | Word | %MW16 |



图解西门子 S7-1200 PLC 入门到实践

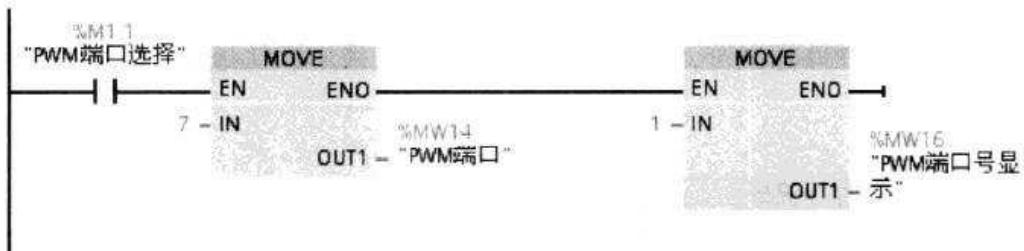
OB1 主程序如图 5-17 所示。

块标题：

通过触摸屏的加减按钮来调节输出 PWM 的宽度

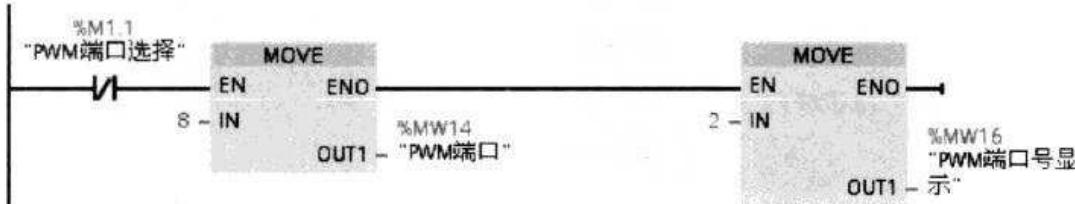
程序段 1：

通过 HMI 的按钮进行选择端口 1



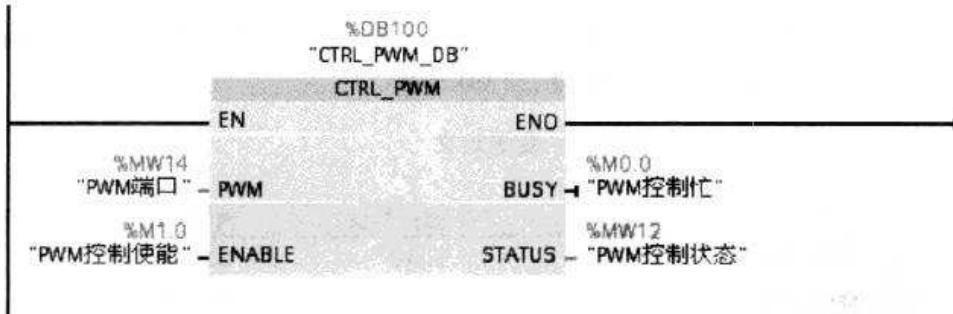
程序段 2：

通过 HMI 的按钮进行选择端口 2



程序段 3：

根据 HMI 上的端口选择按钮进行端口分配：输入使能来自于触摸屏按钮信号 %M1.0；状态变量 %



程序段 4：

输入百分比 %MW 的值来控制脉冲输出宽度 %Q0.0

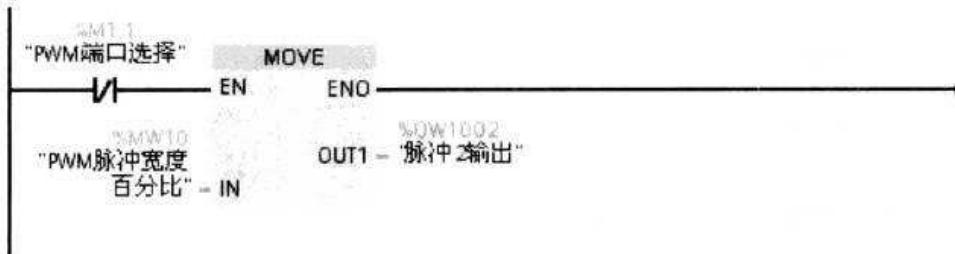


图 5-17 OB1 主程序



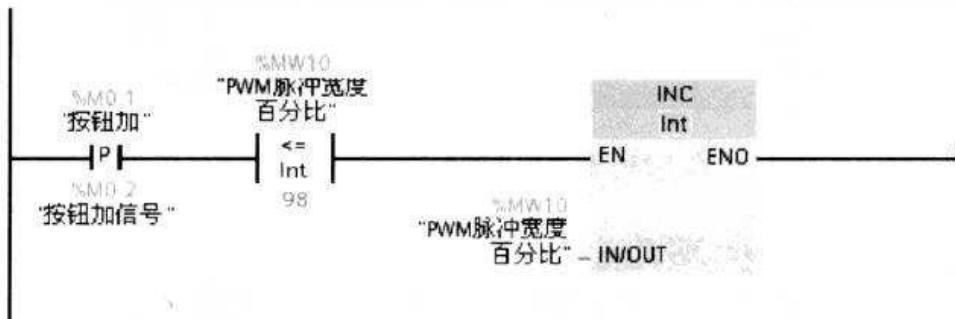
▼ 程序段 5：

输入百分比%MW的值来控制脉冲输出宽度%Q0.2



▼ 程序段 6：

判断小于等于98时，加“1”



▼ 程序段 7：

判断大于等于2时，减“1”

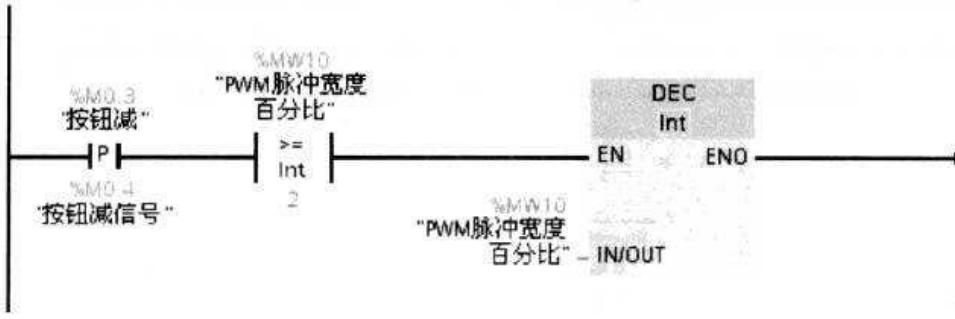


图 5-17 OB1 主程序（续）

3. 触摸屏界面设计

根据控制要求，进行触摸屏的主画面设计如图 5-18 所示。

图 5-19 所示为 PWM 端口切换按钮% M1.1 的事件组态，即释放时进行 InvertBit（位切换）。

图 5-20 所示为加减按钮的事件组态，即按下时进行 SetBitWhileKeyPressed（按键时置位）。从以上事件定义，结合项目 3、项目 4 的案例来看，对于按钮的定义不仅函数种类多，而且事件定义也可以自由发挥，这也是 KTP 系列触摸屏的优点所在。

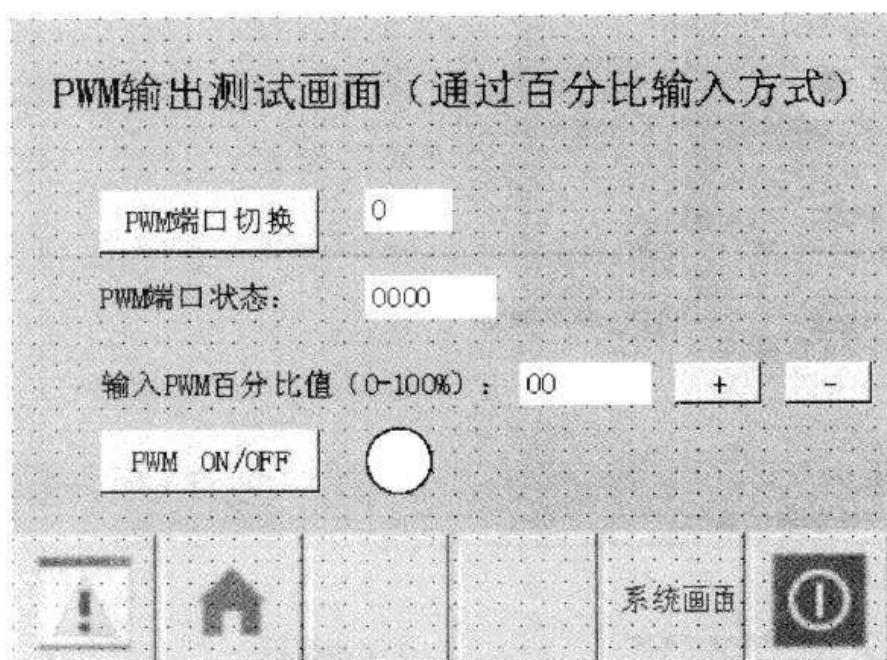


图 5-18 触摸屏主画面设计

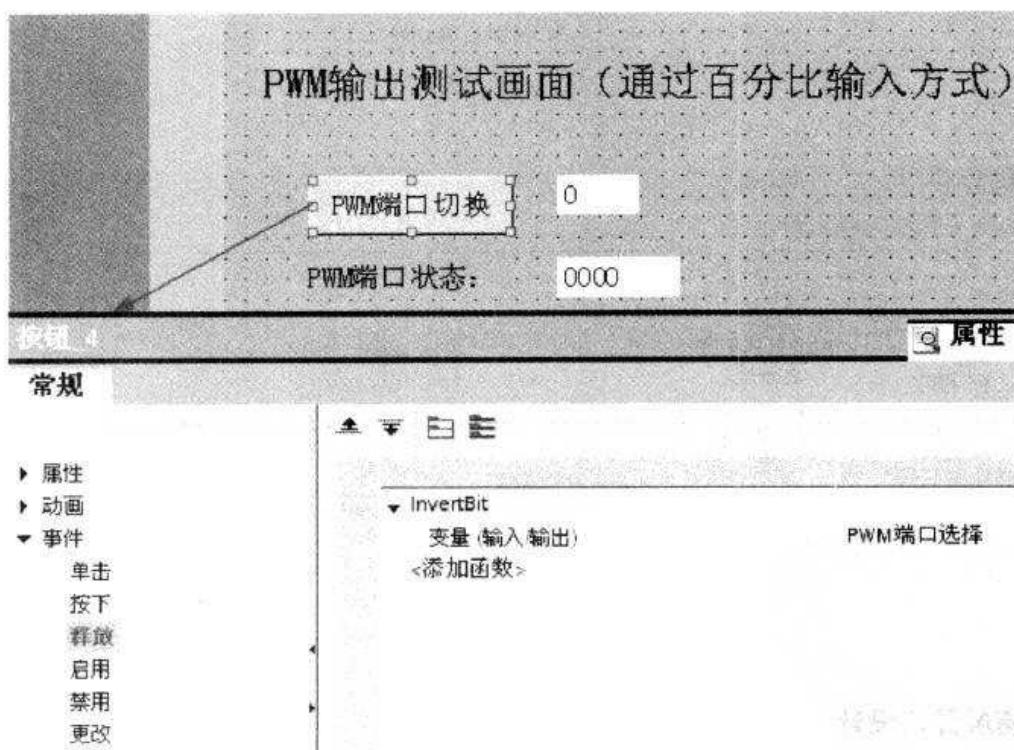


图 5-19 PWM 端口切换按钮% M1.1 的事件组态

图 5-21 所示为 PWM 端口显示“1”或“2”的属性设置，数据类型为 Word，显示格式为十进制，域长度为 1，小数位为 0。

图 5-22 所示为 PWM 端口状态显示，数据类型为 Word，域长度为 4，显示格式为十六进制。

第5章 S7-1200 PLC 的 PWM 控制



图 5-20 加减按钮的事件组态

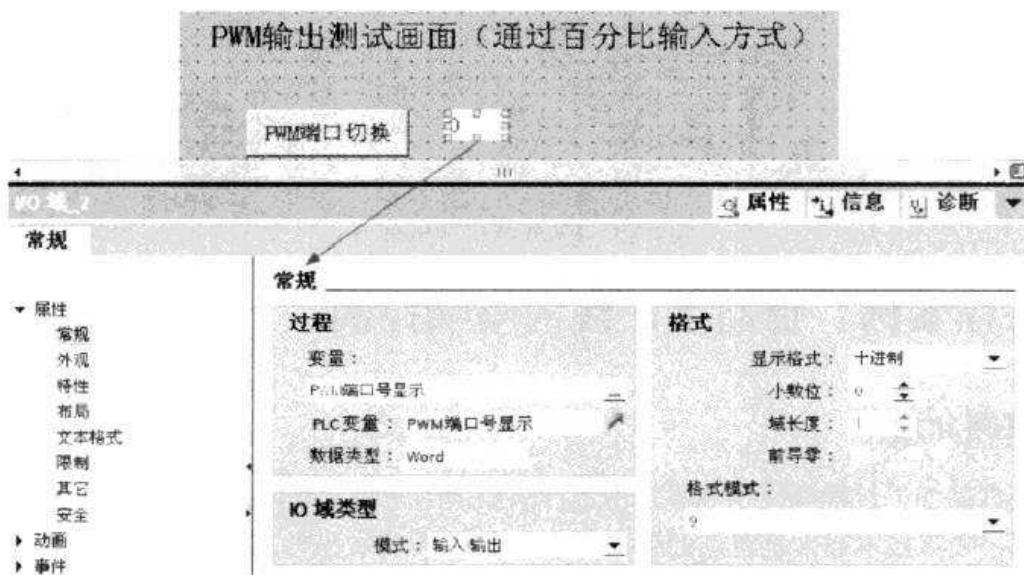


图 5-21 PWM 端口显示“1”或“2”的属性设置

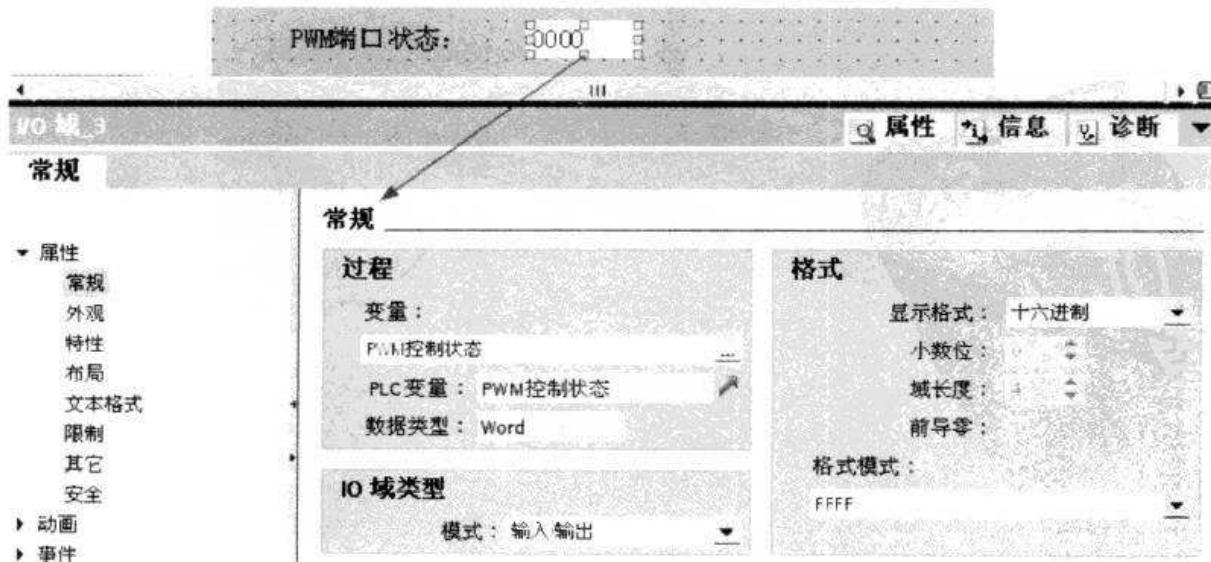


图 5-22 PWM 端口状态显示

图解西门子 S7-1200 PLC 入门到实践

5.2.3 实际测试画面

图 5-23 所示为 PWM 输出测试画面，从图 5-23 中可以看出，现在端口为“2”，端口状态为“0000”（状态正常），输入 PWM 百分比值为 50%，PWM 为 ON 状态（红色为 ON）。

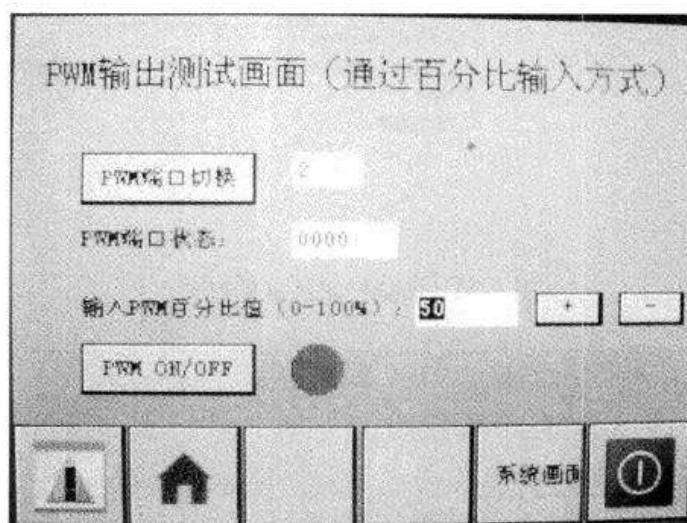


图 5-23 PWM 输出测试画面

5.3 工程案例：食品生产线的喷嘴控制

5.3.1 案例介绍

有许多食品加工过程都使用喷雾技术进行添加剂的配料。此外，食品工厂具有严格的清洗和卫生要求，喷雾技术通常是首选的解决方案。图 5-24 所示为食品生产线的喷嘴控制示意。

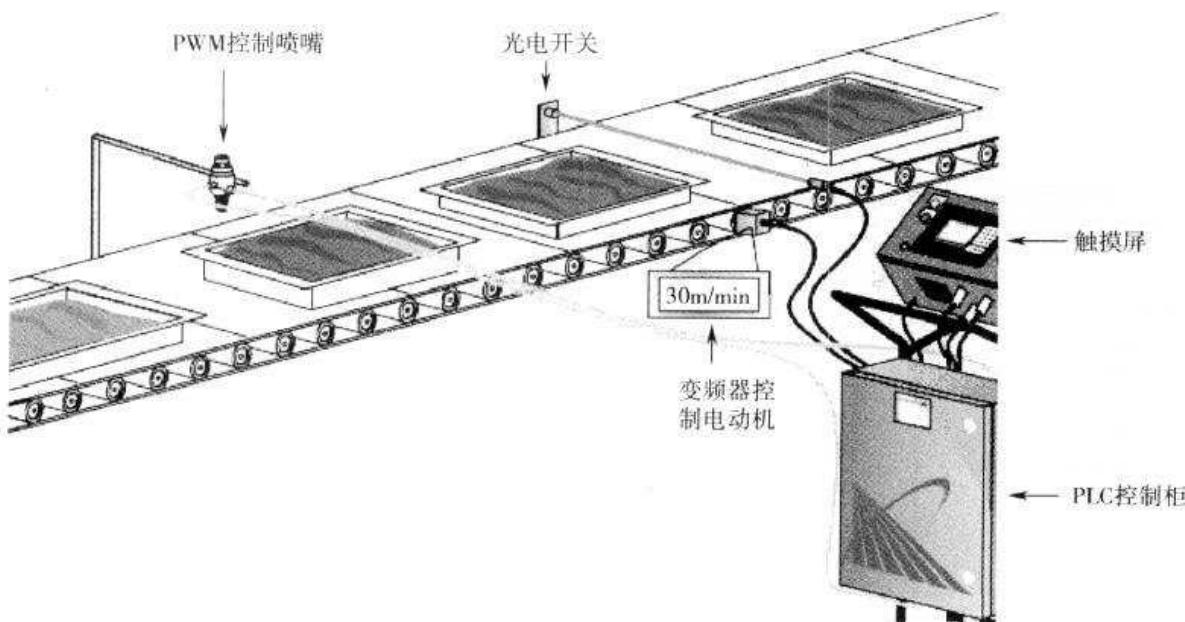


图 5-24 食品生产线的喷嘴控制



为了实现自动化控制，目前喷嘴通常采用 PWM 控制，比如按照一定的占空比进行 ON/OFF（见图 5-25）。现在要求对生产线进行自动控制，要求如下：1) 喷嘴控制分手动和自动两种；2) 在手动情况下，可以在触摸屏上设置占空比；3) 在自动情况下，喷嘴的占空比能自动跟随车速运行，以确保均匀喷洒；4) 能进行故障报警，比如输送带电动机故障或添加剂缺料等。

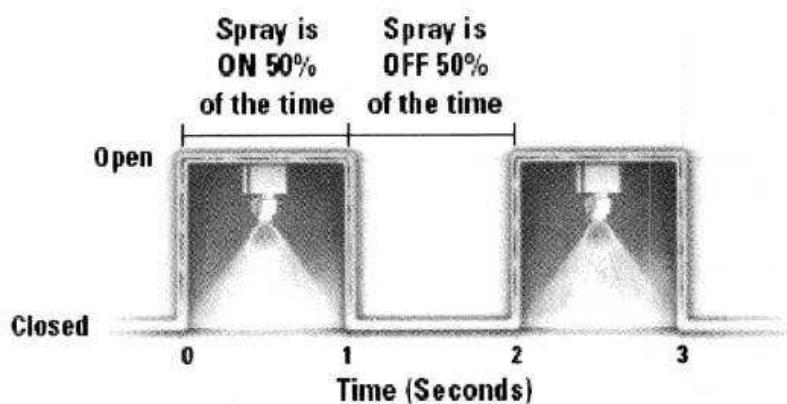


图 5-25 喷嘴 PWM 控制

5.3.2 PWM 硬件配置与软件编程

1. 硬件配置

由于本案例采用了一个 PWM 端口进行控制，以 PWM1 为例，进行设置。按图 5-26 所示进行脉冲选项设置，并采用 S7 模拟量格式。

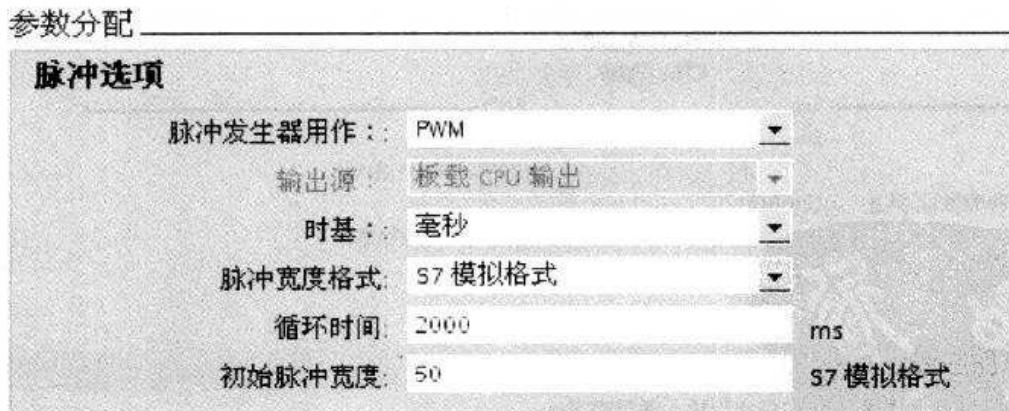


图 5-26 脉冲选项设置

2. 变量定义

本案例的变量定义见表 5-5，其中包括输送带电机故障输入% I0.0、添加剂缺料故障输入% I0.1、输送带速度% IW96、PWM 喷嘴输出% QW1000 等 I/O 变量，还包括 PWM 使能/禁止% M0.0、PWM BUSY% M0.1 等中间变量。



表 5-5 变量定义

| | 名称 | 数据类型 | 地址 |
|----|---------------|------|---------|
| 1 | ● 输送带电机故障输入 | Bool | %I0.0 |
| 2 | ● 添加剂缺料故障输入 | Bool | %I0.1 |
| 3 | ● 输送带速度 | Word | %IW96 |
| 4 | ● PWM喷嘴输出 | Word | %QW1000 |
| 5 | ● PWM使能/禁止 | Bool | %M0.0 |
| 6 | ● PWM BUSY | Bool | %M0.1 |
| 7 | ● 喷嘴手动/自动 | Bool | %M0.2 |
| 8 | ● PWM状态 | Word | %MW2 |
| 9 | ● 触摸屏 PWM 设定 | Int | %MW4 |
| 10 | ● PWM 设定值转为实数 | Real | %MD6 |
| 11 | ● 速度(米/分) | Real | %MD10 |
| 12 | ● 占空比 | Real | %MD14 |
| 13 | ● 故障字 | Int | %MW18 |
| 14 | ● PWM出错 | Bool | %M19.0 |
| 15 | ● 输送带电机故障 | Bool | %M19.1 |
| 16 | ● 添加剂缺料故障 | Bool | %M19.2 |

3. PLC 梯形图程序

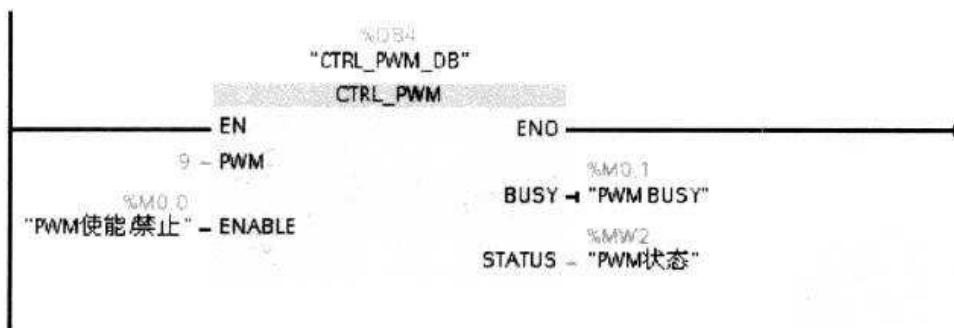
主程序 OB1 如图 5-27 所示。

块标题：

食品加工厂添加剂 PWM 控制

程序段 1：

调用 PWM 功能



程序段 2：

喷嘴自动情况下，其占空比与输入速度成正比



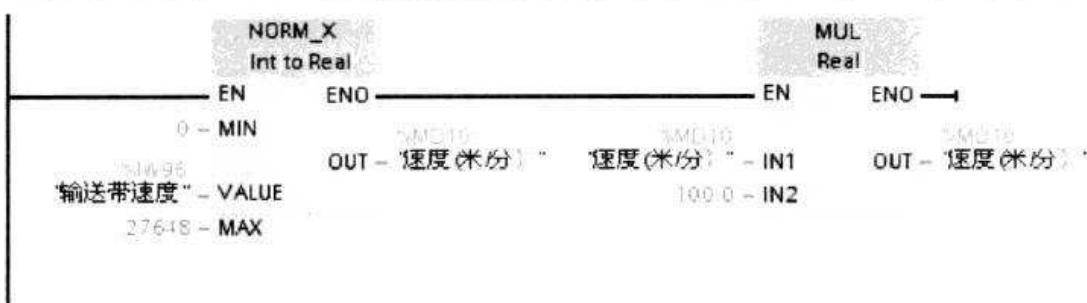
图 5-27 主程序 OB1

第5章 S7-1200 PLC 的 PWM 控制



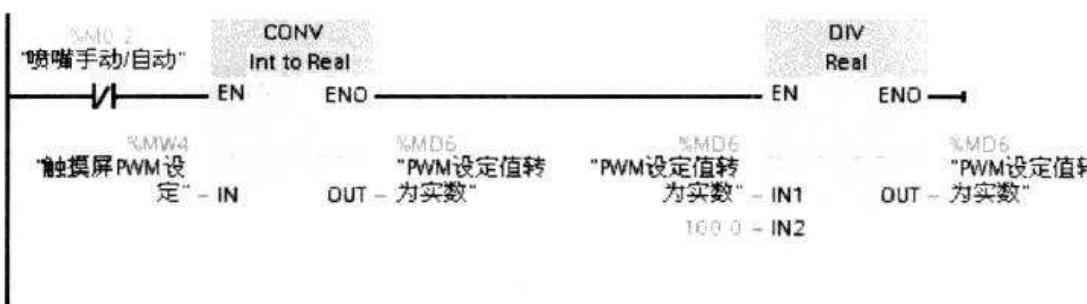
▼ 程序段 3：

速度比例



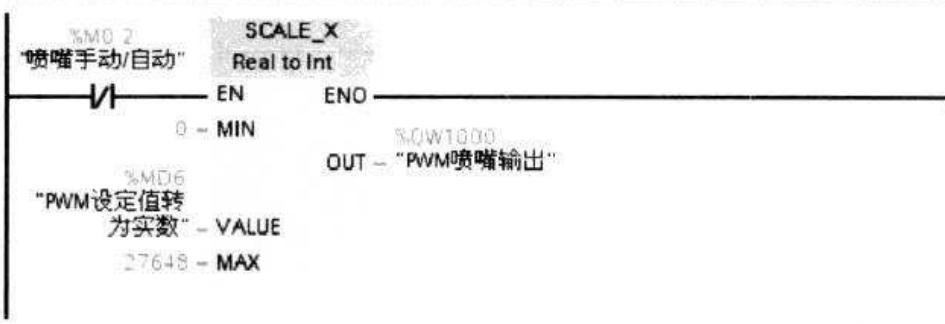
▼ 程序段 4：

喷嘴手动情况下,可以自由设定占空比



▼ 程序段 5：

将手动占空比设定值输出到PWM1口



▼ 程序段 6：

将手动或自动情况下的占空比输出%QW1000转为百分比值（触摸屏显示用）

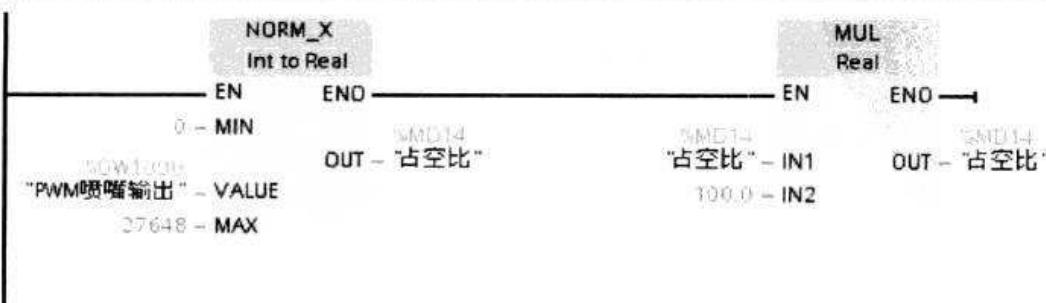
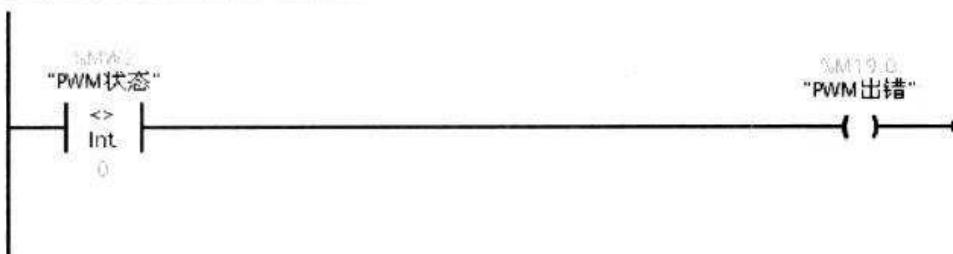


图 5-27 主程序 OB1 (续一)



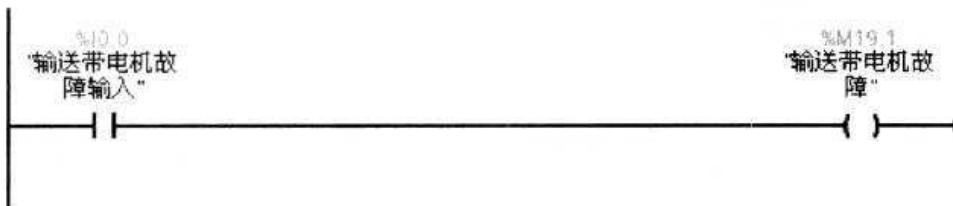
▼ 程序段 7：

故障字定义1: PWM端口故障



▼ 程序段 8：

故障字定义2: 输送带电机故障



▼ 程序段 9：

故障字定义3: 添加剂缺料故障

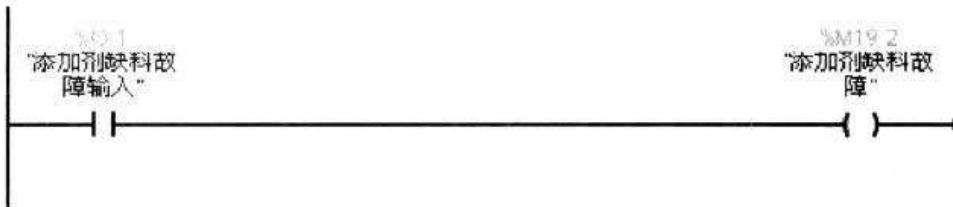


图 5-27 主程序 OBI (续二)

4. 触摸屏的画面组态

触摸屏的画面组态步骤如下：

- 1) 点击图 5-28 所示的 Template_1 模板设置进入图 5-29 所示的模板画面，新增“主画面”、“喷嘴设置”、“帮助”3 个主画面，并将事件定义为“跳转相应页面”。

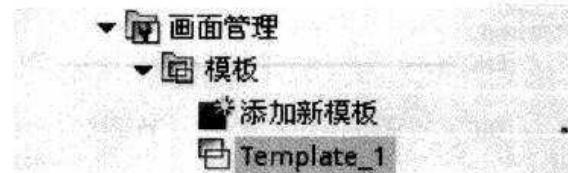


图 5-28 模板设置

- 2) 在图 5-30 所示的主画面进行画面设计。

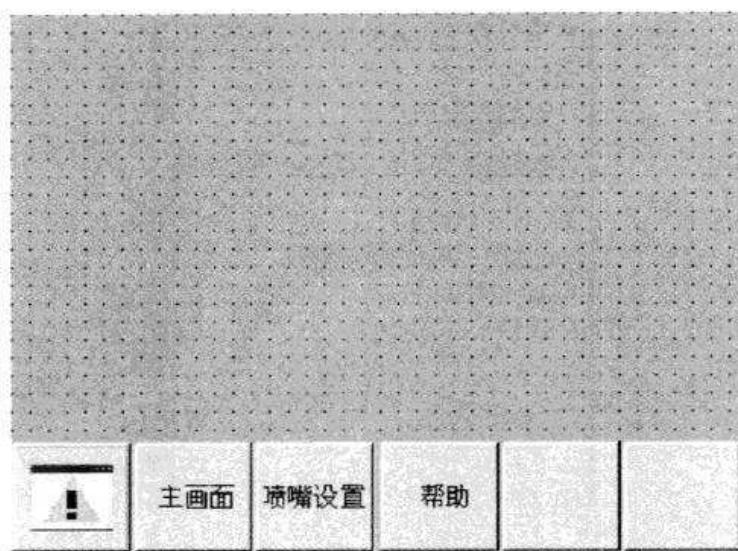


图 5-29 模板画面

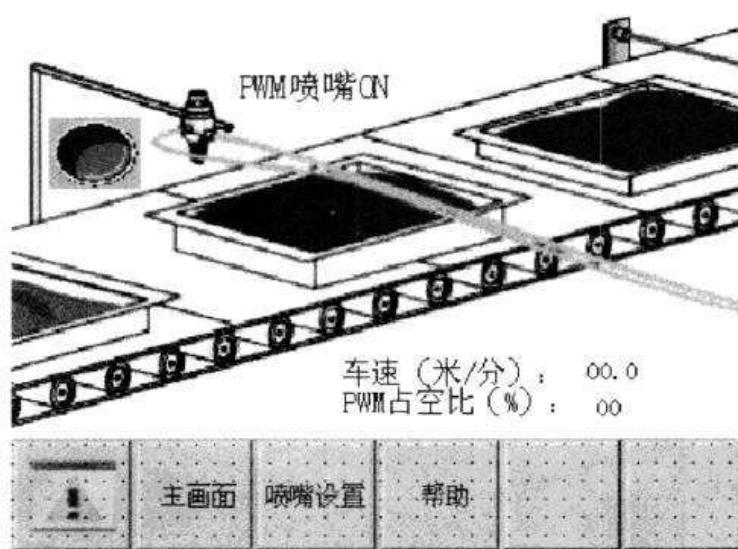


图 5-30 主画面

从图 5-30 中可以看出，PWM 喷嘴 ON 的按钮是采用了 HMI 内置的“库”，这样一来画面相对就会丰富多彩。

那么如何建立库呢？图 5-31 所示为库中的“全局库”，打开“HMI Buttons & Switches”就会发现各种各样的按钮开关等现成的图库（见图 5-32）。

PWM 喷嘴 ON 的按钮的常规属性如图 5-33 所示，其事件的打开和关闭组态如图 5-34 和图 5-35 所示。

3) 在图 5-36 所示的喷嘴设置中进行画面设计。其中也是一个图符库元件，其变量名称为“喷嘴手动/自动”。

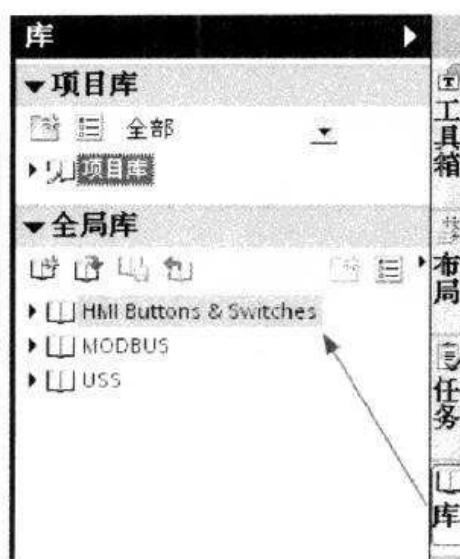


图 5-31 全局库中的 HMI Buttons & Switches



图 5-32 各种各样的按钮开关等现成的图库

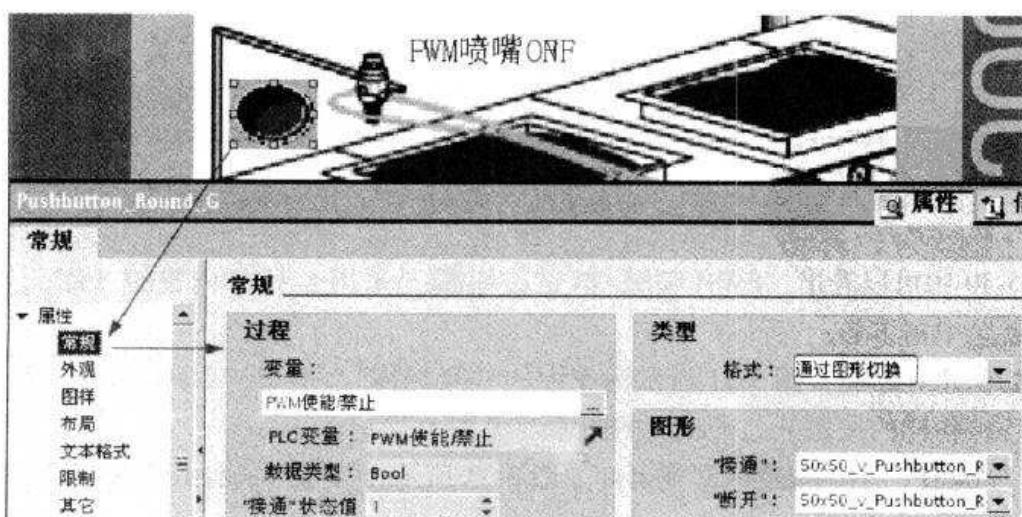


图 5-33 常规属性

第5章 S7-1200 PLC 的 PWM 控制

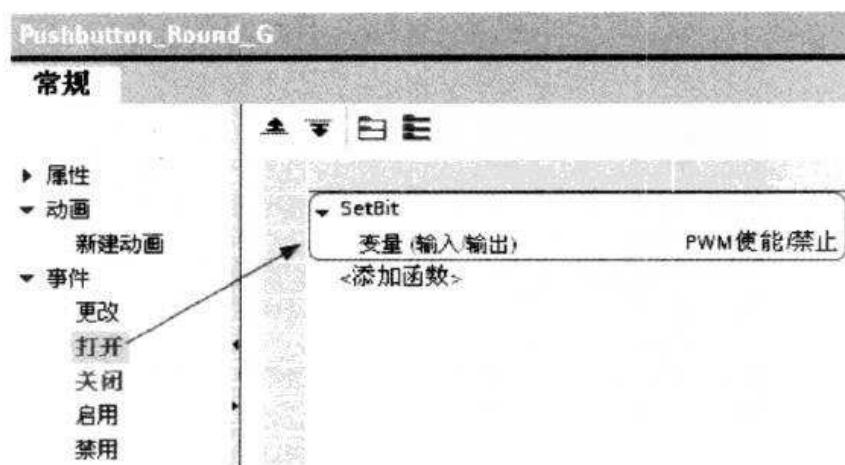


图 5-34 打开事件组态

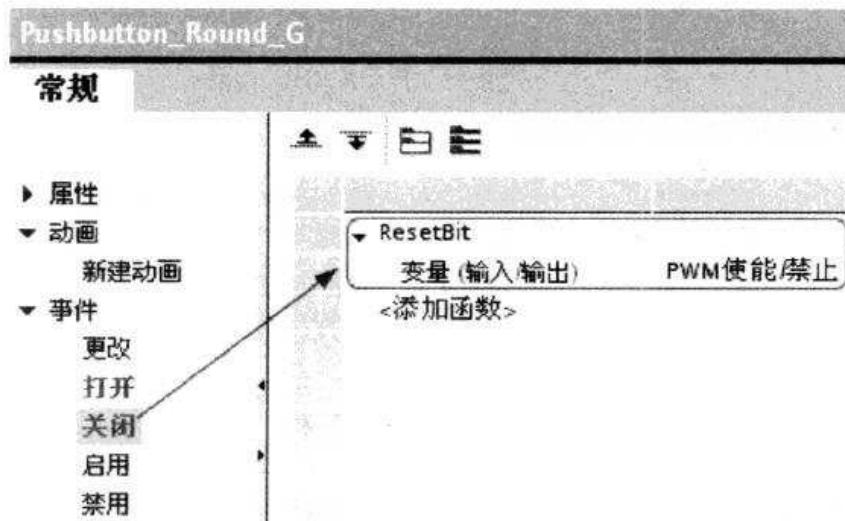


图 5-35 关闭事件组态

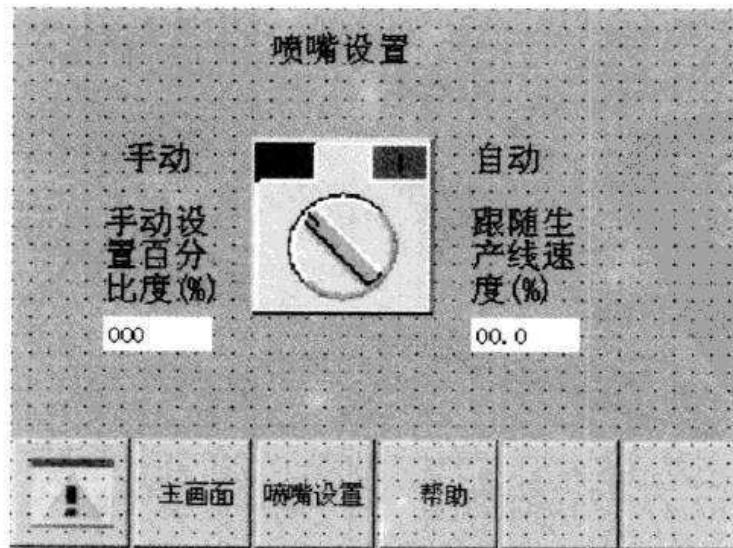


图 5-36 喷嘴设置



4) 在图 5-37 所示的帮助画面中进行画面设计。

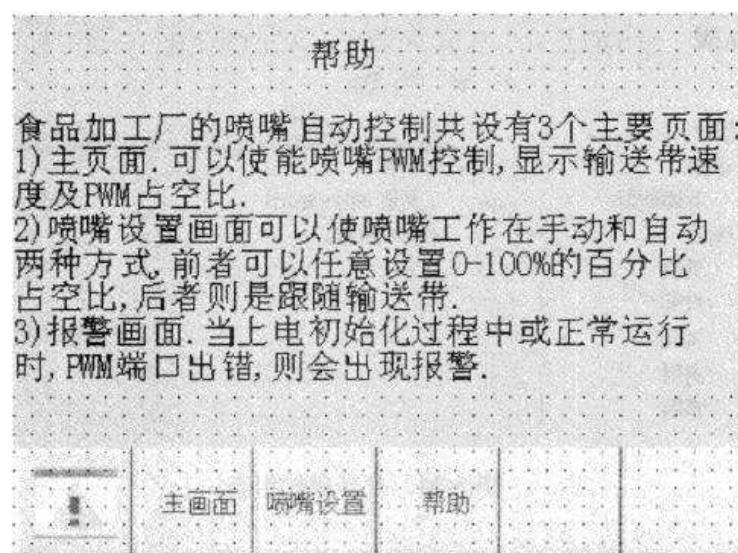


图 5-37 帮助画面

5) 在 HMI 报警中增加 3 条故障报警 (见图 5-38)。

| ID | 事件文本 | 报警类别 | 触发变量 | 触... | HMI 确认变量 | HMI... |
|----|---------|------|------|------|----------|--------|
| 1 | PWM端口故障 | 错误 | 故障字 | 0 | <无变量> | 0 |
| 2 | 输送带电机故障 | 错误 | 故障字 | 1 | <无变量> | 0 |
| 3 | 添加剂缺料故障 | 错误 | 故障字 | 2 | <无变量> | 0 |

图 5-38 故障报警画面

5.3.3 实际测试结果

实际运行的主画面如图 5-39 所示，当 PWM 喷嘴 ON 时，车速达到 39.1 米/分，PWM 占

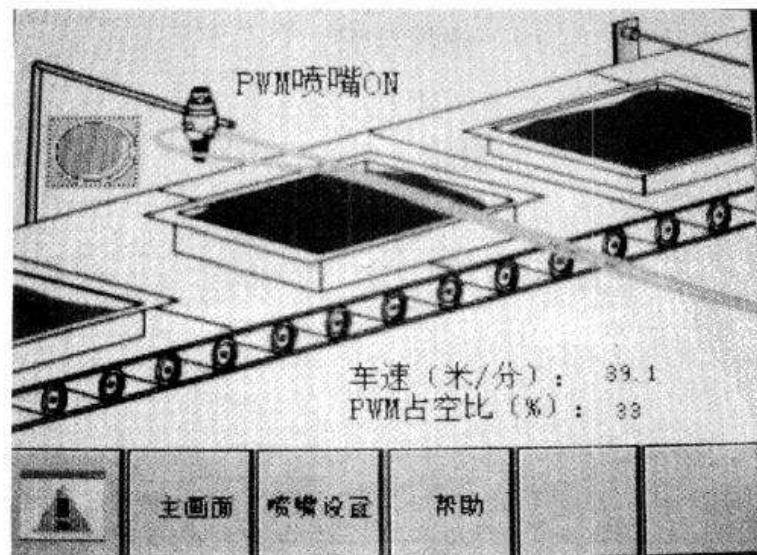


图 5-39 实际运行的主画面

第5章 S7-1200 PLC 的 PWM 控制



空比相应回应为 33%（其比例可以在 PLC 程序段第三段中进行修改）

图 5-40 所示为实际的喷嘴设置画面，此时为自动状态，跟随生产线速度为 38.7%。



图 5-40 实际的喷嘴设置画面

图 5-41 所示为故障画面，共有两条故障，即 %M19.1 和 %M19.2 动作。

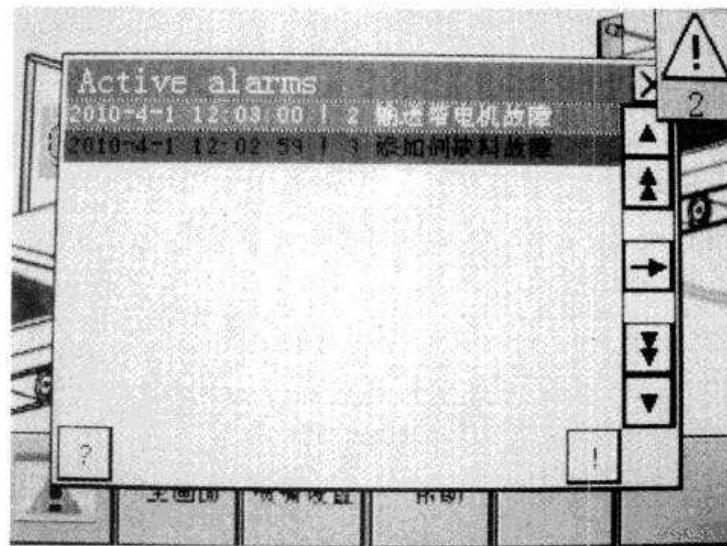
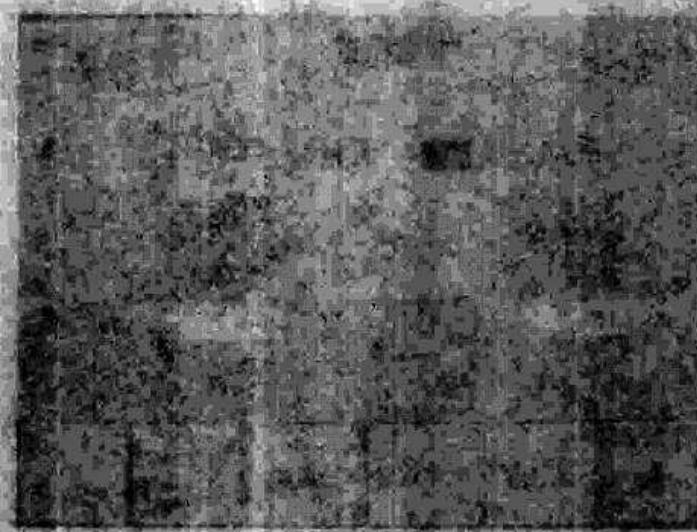


图 5-41 故障画面

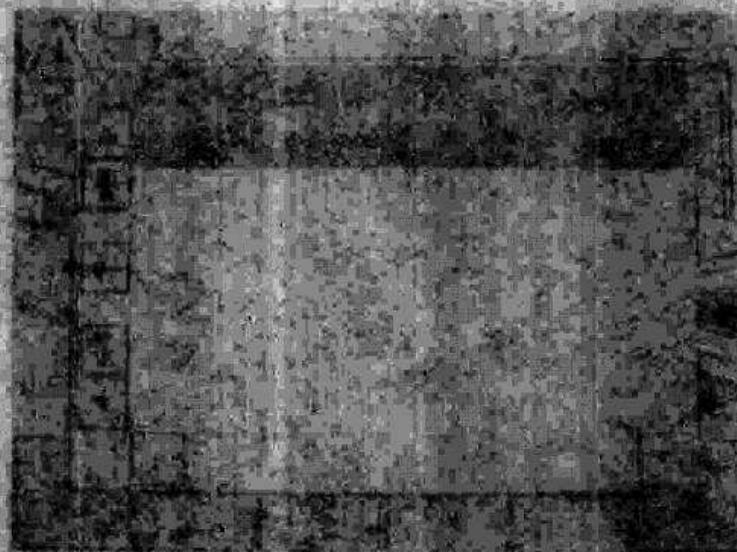
图 5 MW1#0.1#001-12 第 2 页

(如图所示中显示为逻辑块 123 未从存储器读取，将生成故障消息
，表示 800 级别的故障。本例中白衣 101#，而由多线与则由硬接线连接 0#-1#)



（图 5 多线与输出端子 0#-1#）

（图 5 RIM 制作：双行双列，两个系列的开关，通过接触式水银开关）

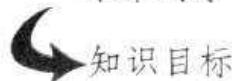


（图 5 第 2 页）

第6章 S7-1200 PLC的HSC控制

在工业控制中的有些场合，输入的是一些高速脉冲信号，如编码器信号，这时候PLC可以使用高速计数器功能对这些特定的脉冲量进行加减计数，来最终获取所需要的工艺数据（转速、角度、位移等）。从硬件角度来讲，小型PLC都会内置一些端口用于高速脉冲输入，其结构与普通的数字量不同。从软件角度来讲，小型PLC都会采用特殊的高速计数器指令来进行中断处理。

本章的学习目标如下：



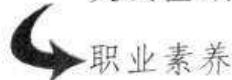
知识目标

了解脉冲量输入和高速计数器；熟悉高速计数器的4种模式；熟悉S7-1200 PLC HSC的指令与硬件。



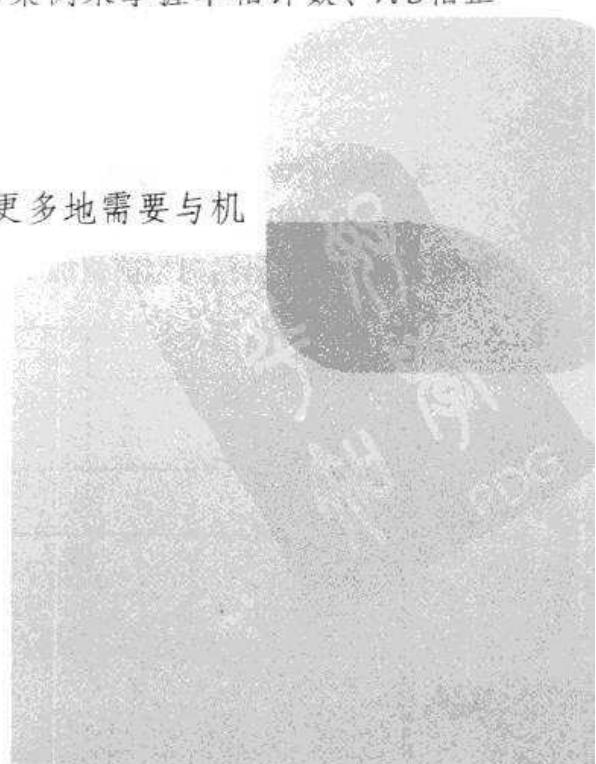
能力目标

能对HSC进行硬件组态，能通过一个简单的案例来掌握单相计数、A/B相正交测量频率和定长控制。



职业素养目标

解决问题不单单靠电气自动化技术，而是更多地需要与机械、工艺进行配合，才能完成一个综合的解决方案。





6.1 背景知识介绍：HSC 概念

6.1.1 脉冲量输入和高速计数器

在工业控制中的有些场合，输入的是一些高速脉冲信号，如编码器信号，这时候 PLC 可以使用高速计数器功能对这些特定的脉冲量进行加减计数，来最终获取所需要的工艺数据（转速、角度、位移等）。从硬件角度来讲，中小型 PLC 都会内置一些端口用于高速脉冲输入，其结构与普通的数字量不同。从软件角度来讲，小型 PLC 都会采用特殊的高速计数器指令来进行中断处理。

高速计数器的模式一般有以下四种：

1. 单相运行模式

单相运行模式即在输入脉冲的上升沿时当前值加 1，如图 6-1 所示。

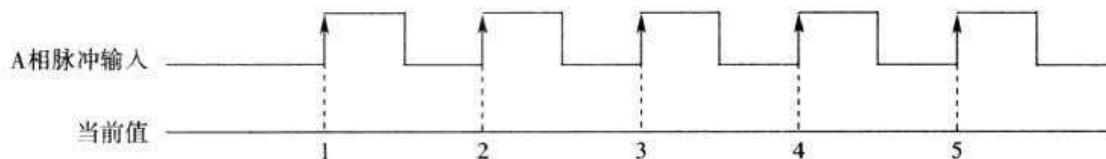


图 6-1 单相运行模式

2. 单相脉冲 + 方向模式

当 B 相在低电平时，在 A 相脉冲的上升沿时当前值加 1。当 A 相在高电平时，在 A 相脉冲的上升沿时当前值加 1，如图 6-2 所示。

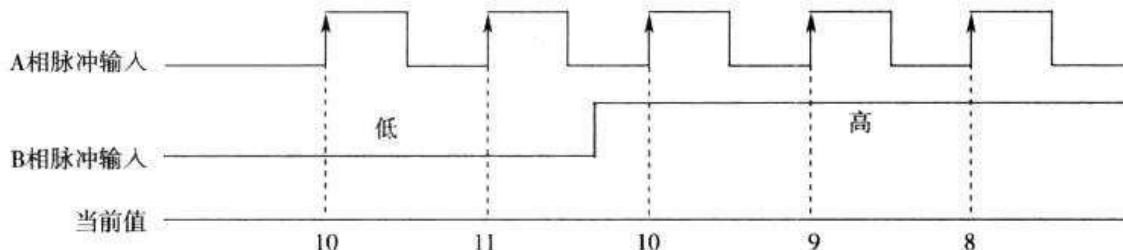


图 6-2 单相脉冲 + 方向模式

3. 双相 CW/CCW 模式

当 B 相在低电平时，在 A 相输入脉冲的上升沿时当前值加 1。当 A 相在低电平时，在 B 相输入脉冲的上升沿时当前值加 1，如图 6-3 所示。

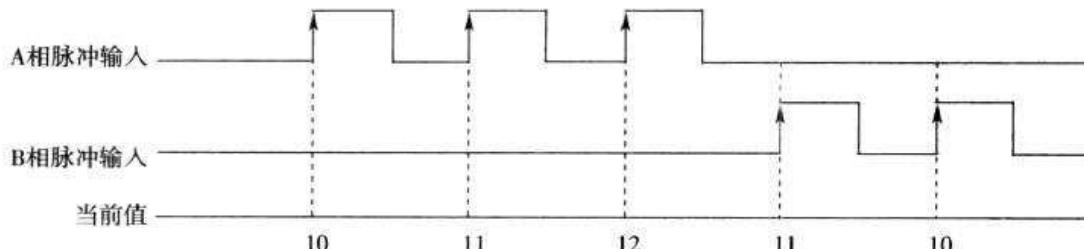


图 6-3 双相 CW/CCW 模式



4. A/B 相正交脉冲模式

Up 或 Down 通过 A 和 B 相的不同自动设定，如图 6-4 所示。

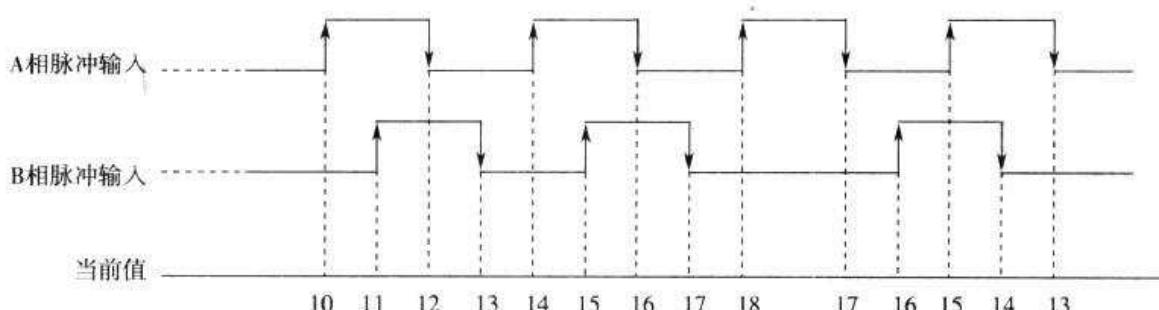


图 6-4 A/B 相正交脉冲模式

(1) Up 计数器

- 当 B 相在低电平时，在 A 相脉冲输入的上升沿动作。
- 当 B 相在高电平时，在 A 相脉冲输入的下降沿动作。
- 当 A 相在高电平时，在 B 相脉冲输入的上升沿动作。
- 当 A 相在低电平时，在 B 相脉冲输入的下降沿动作。

(2) Down 计数器

- 当 B 相在高电平时，在 A 相脉冲输入的上升沿动作。
- 当 B 相在低电平时，在 A 相脉冲输入的下降沿动作。
- 当 A 相在低电平时，在 B 相脉冲输入的上升沿动作。
- 当 A 相在高电平时，在 B 相脉冲输入的下降沿动作。

图 6-4 所示为正交 4 倍，还有一种是正交 1 倍的模式。

6.1.2 S7-1200 PLC HSC 的指令与硬件

1. HSC 的接线

图 6-5 所示为 S7-1200 PLC 与高速 PG 编码器的连接。HSC 具体连接的 PLC 的输入点与功能见表 6-1。

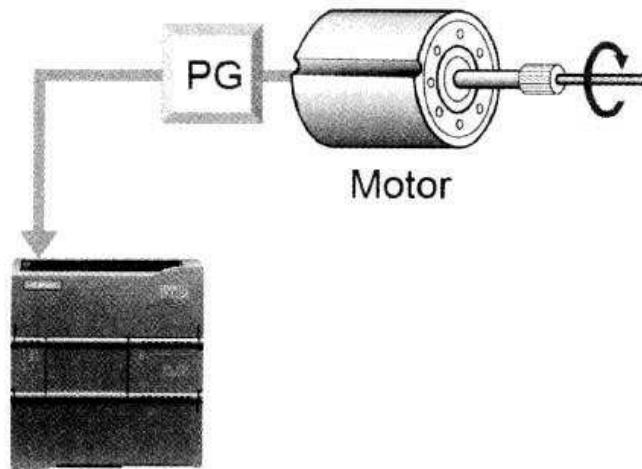


图 6-5 S7-1200 PLC 与高速 PG 编码器的连接



表 6-1 HSC 的具体连接的 PLC 的输入点与功能

| 说 明 | | | 默认输入分配 | | | 功 能 |
|-----|-------------------|------------------------|---------------------------|---------------------------|---------------------|-------|
| HSC | HSC1 | 内置 或信号板 或监视 PTO0 | I0. 0 I4. 0 PTO0 脉冲 | I0. 1 I4. 1 PTO0 方向 | I0. 3 I4. 3 — | |
| | HSC | 内置 或信号板 或监视 PTO1 | I0. 2 I4. 2 PTO1 脉冲 | I0. 3 I4. 3 PTO1 方向 | I0. 1 I4. 1 — | |
| | HSC3 ² | 内置 | I0. 4 | I0. 5 | I0. 7 | |
| | HSC4 ³ | 内置 | I0. 6 | I0. 7 | I0. 5 | |
| | HSC5 ⁴ | 内置 或信号板 | I1. 0 I4. 0 | I1. 1 I4. 1 | I1. 2 I4. 3 | |
| | HSC6 ⁴ | 内置 或信号板 | I1. 3 I4. 2 | I1. 4 I4. 3 | I1. 5 I4. 1 | |
| 模式 | 具有内部方向控制的单相计数器 | 时钟 | — | — | — | 计数或频率 |
| | | | | | 复位 | 计数 |
| | 具有外部方向控制的单相计数器 | 时钟 | 方向 | — | — | 计数或频率 |
| | | | | | 复位 | 计数 |
| | 具有两个时钟输入的双相计数器 | 加时钟 | 减时钟 | — | — | 计数或频率 |
| | A/B 相正交计数器 | A 相 | | | 复位 | 计数 |
| | 监视脉冲串输出 (PTO) | 时钟 | 方向 | — | — | 计数 |

脉冲串输出监视功能始终使用时钟和方向。如果仅为脉冲组态了相应的 PTO 输出，则通常应将方向输出设置为正计数。对于仅支持 6 个内置输入的 CPU 1211C，不能使用带复位输入的 HSC3。仅当安装信号板时，CPU 1211C 和 CPU 1212C 才支持 HSC5 和 HSC6。

2. 访问 HSC 的当前值

CPU 将每个 HSC 的当前值存储在一个输入 (I) 地址中。表 6-2 列出了为每个 HSC 的当前值分配的默认地址。可以通过在设备配置中修改 CPU 的属性来更改当前值的 I 地址。

表 6-2 高速计数器的数据类型及默认地址

| 高速计数器 | 数据类型 | 默认地址 |
|-------|------|--------|
| HSC1 | DInt | ID1000 |
| HSC2 | DInt | ID1004 |
| HSC3 | DInt | ID1008 |
| HSC4 | DInt | ID1012 |
| HSC5 | DInt | ID1016 |
| HSC6 | DInt | ID1020 |

第6章 S7-1200 PLC 的 HSC 控制



在设备配置期间分配高速计数器设备使用数字量 I/O 点。将数字量 I/O 点分配给这些设备之后，无法通过监视表格强制功能修改所分配的 I/O 点的地址值。

3. HSC 调用指令

图 6-6 所示为 HSC 调用指令。HSC 指令 I/O 参数说明见表 6-3。

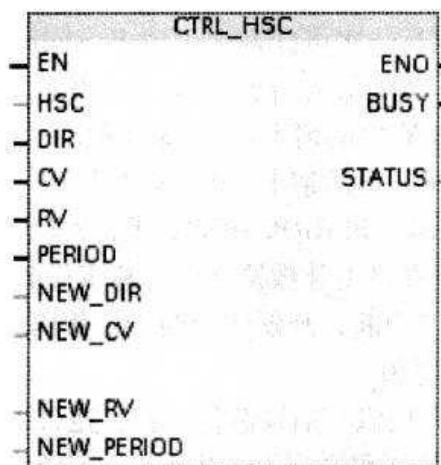


图 6-6 HSC 调用指令

表 6-3 HSC 指令 I/O 参数说明

| 参数 | 参数类型 | 数据类型 | 说 明 |
|------------|------|--------|---------------------------------------|
| HSC | IN | HW_HSC | HSC 标识符 |
| DIR | IN | Bool | 1 = 请求新方向 |
| CV | IN | Bool | 1 = 请求设置新的计数器值 |
| RV | IN | Bool | 1 = 请求设置新的参考值 |
| PERIOD | IN | Bool | 1 = 请求设置新的周期值 (仅限频率测量模式) |
| NEW_DIR | IN | Int | 新方向： 1 = 向上 -1 = 向下 |
| NEW_CV | IN | DInt | 新计数器值 |
| NEW_RV | IN | DInt | 新参考值 |
| NEW_PERIOD | IN | Int | 以秒为单位的新周期值：0.01、0.1 或 1 (仅限频率测量模式) |
| BUSY | OUT | Bool | 功能忙 |
| STATUS | OUT | Word | 执行条件代码 |

必须先在项目设置 PLC 设备配置中组态高速计数器，然后才能在程序中使用高速计数器。HSC 设备配置设置包括选择计数模式、I/O 连接、中断分配以及是作为高速计数器还是设备来测量脉冲频率。无论是否采用程序控制，均可操作高速计数器。

许多高速计数器组态参数只在项目设备配置中进行设置。有些高速计数器参数在项目设备配置中初始化，但以后可以通过程序控制进行修改。

CTRL_HSC 指令参数提供了计数过程的程序控制：



- 1) 将计数方向设置为 NEW_DIR 值；
- 2) 将当前计数值设置为 NEW_CV 值；
- 3) 将参考值设置为 NEW_RV 值；
- 4) 将周期值（仅限频率测量模式）设置为 NEW_PERIOD 值。

如果执行 CTRL_HSC 指令后以下布尔标记值置位为 1，则相应的 NEW_xxx 值将装载到计数器。CTRL_HSC 指令执行一次可处理多个请求（同时设置多个标记）。

- 1) DIR = 1 是装载 NEW_DIR 值的请求，0 = 无变化。
- 2) CV = 1 是装载 NEW_CV 值的请求，0 = 无变化。
- 3) RV = 1 是装载 NEW_RV 值的请求，0 = 无变化。
- 4) PERIOD = 1 是装载 NEW_PERIOD 值的请求，0 = 无变化。

CTRL_HSC 指令通常放置在触发计数器硬件中断事件时执行的硬件中断 OB 中。例如，如果 CV = RV，事件触发计数器中断，则硬件中断 OB 代码块执行 CTRL_HSC 指令并且可通过装载 NEW_RV 值更改参考值。

在 CTRL_HSC 参数中没有提供当前计数值。在高速计数器硬件配置期间分配存储当前计数值的过程映像地址。可以使用程序逻辑直接读取该计数值，并且返回到程序的值将是读取计数器瞬间的正确计数。但计数器仍将继续对高速事件计数。因此，程序使用旧的计数值完成处理前，实际计数值可能会更改。

CTRL_HSC 参数的详细信息：

- 1) 如果不请求更新参数值，则会忽略相应的输入值。
- 2) 仅当组态的计数方向设置为“用户程序（内部方向控制）”（User program (internal-direction control)）时，DIR 参数才有效。用户在 HSC 设备配置中确定如何使用该参数。
- 3) 对于 CPU 或信号板上的 S7-1200 HSC，BUSY 参数的值始终为 0。

条件代码：发生错误时，ENO 设置为 0，并且 STATUS 输出包含条件代码（见表 6-4）。

表 6-4 STATUS 故障代码

| STATUS 值 (W#16#…) | 说 明 |
|-------------------|-------------------|
| 0 | 无错误 |
| 80A1 | HSC 标识符没有对 HSC 寻址 |
| 80B1 | NEW_DIR 的值非法 |
| 80B2 | NEW_CV 的值非法 |
| 80B3 | NEW_RV 的值非法 |
| 80B4 | NEW_PERIOD 的值非法 |

4. 编码器 PG

编码器 PG 是把角位移或直线位移转换成电信号的一种装置，前者称为码盘，后者称为码尺，如图 6-7 所示为编码器外观。

编码器有以下分类方式：

- 1) 按照读出方式，编码器可以分为接触式和非接触式两种。接触式采用电刷输出，以电刷接触导电区或绝缘区来表示代码的状态是“1”还是“0”；非接触式的接受敏感元件是光敏元件或磁敏元件，采用光敏元件时，以透光区和不透光区来表示代码的状态是“1”还是“0”。
- 2) 按照工作原理，编码器可分为增量式和绝对式两类。增量式编码器是将位移转换成



周期性的电信号，再把这个电信号转变成计数脉冲，用脉冲的个数表示位移的大小。绝对式编码器的每一个位置对应一个确定的数字码，因此它的示值只与测量的起始和终止位置有关，而与测量的中间过程无关。

绝对式编码器可以分单圈绝对式编码器和多圈绝对式编码器。旋转单圈绝对式编码器在转动中测量光码盘各道刻线，以获取惟一的编码，当转动超过 360° 时，编码又回到原点，这样就不符合绝对编码惟一的原则，这样的编码器只能用于旋转范围在 360° 以内的测量，称为单圈绝对式编码器。如果要测量旋转超过 360° 的范围，就要用到多圈绝对式编码器。



图 6-7 编码器外观

旋转编码器的机械安装有高速端安装、低速端安装、辅助机械装置安装等多种形式。

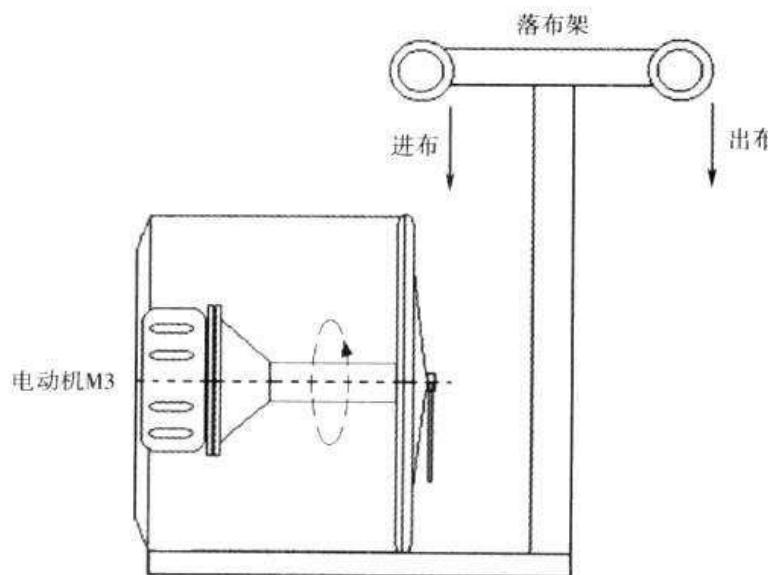
高速端安装：安装于传动转轴端（或齿轮连接），此方法的优点是分辨率高，由于多圈编码器有 4096 圈甚至更多，转动圈数在此量程范围内，可充分用足量程而提高分辨率，缺点是运动物体通过减速齿轮后，来回程有齿轮间隙误差，因此这种安装方法一般用于单向高精度控制定位。编码器直接安装于高速端，传动抖动须较小，不然易损坏编码器。

低速端安装：安装于减速齿轮后，如卷扬钢丝绳卷筒的轴端或最后一节减速齿轮轴端，此方法已无齿轮来回程间隙，测量较直接，精度较高。

6.2 技能训练一：单相计数

6.2.1 案例介绍

在某离心机中，为了直观地了解离心水洗机进料或出料的长度，需要在落布架的转动轴处安装一个带齿轮的码盘，并配接一个电感式传感器，来获取齿轮变化的规律（见图 6-8）。



a) 离心机

图 6-8 落布长度的测量

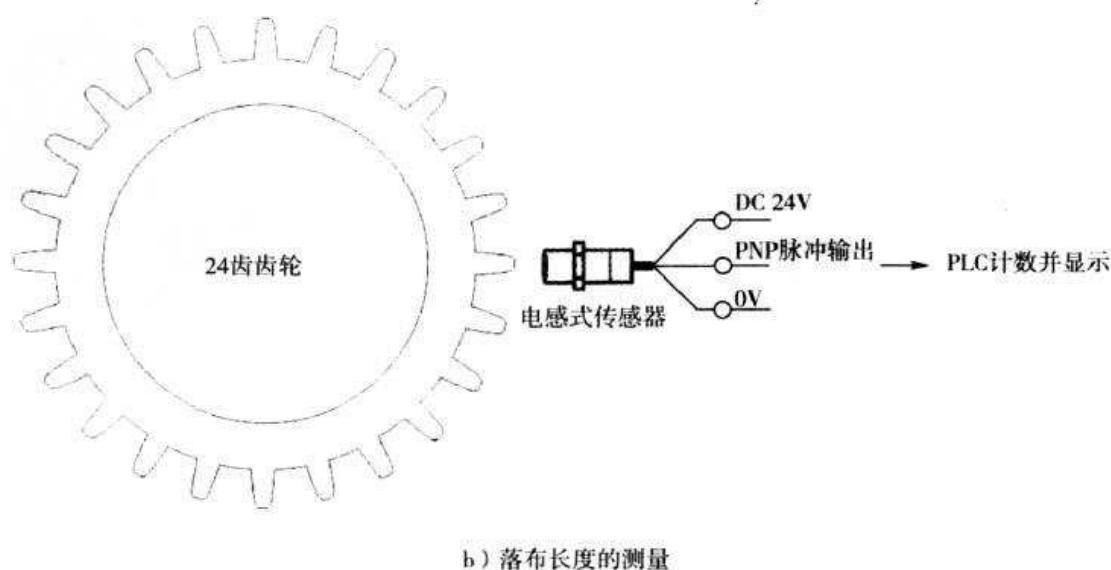


图 6-8 落布长度的测量 (续)

控制要求如下：利用 S7-1200 PLC 进行计数控制，进入离心机的长度为 100 脉冲、200 脉冲、100 脉冲、200 脉冲……（间隔进行），请进行硬件接线、配置与软件编程。

6.2.2 硬件组态与软件编程

1. 硬件接线

本案例采用 PNP 型齿轮传感器时，PLC 的高速计数器的硬件接线如图 6-9 所示。

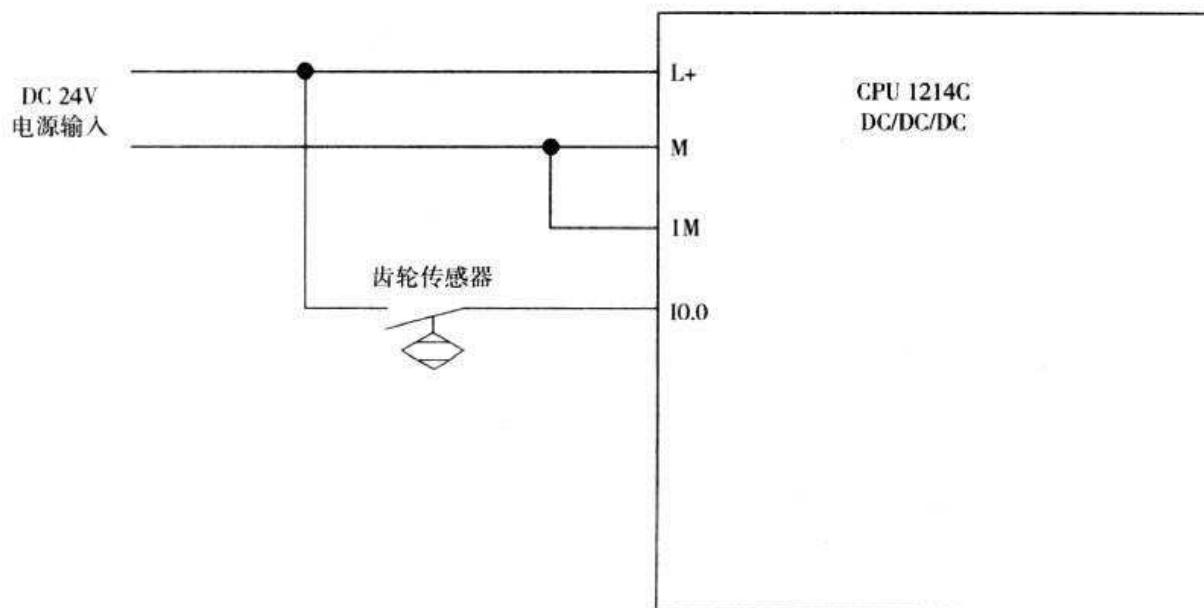


图 6-9 PLC 的高速计数器的硬件接线

当然，如果采用 NPN 型传感器时，则需要按照图 6-10 所示进行接入，图 6-10 中输入端子为 %I0.0。

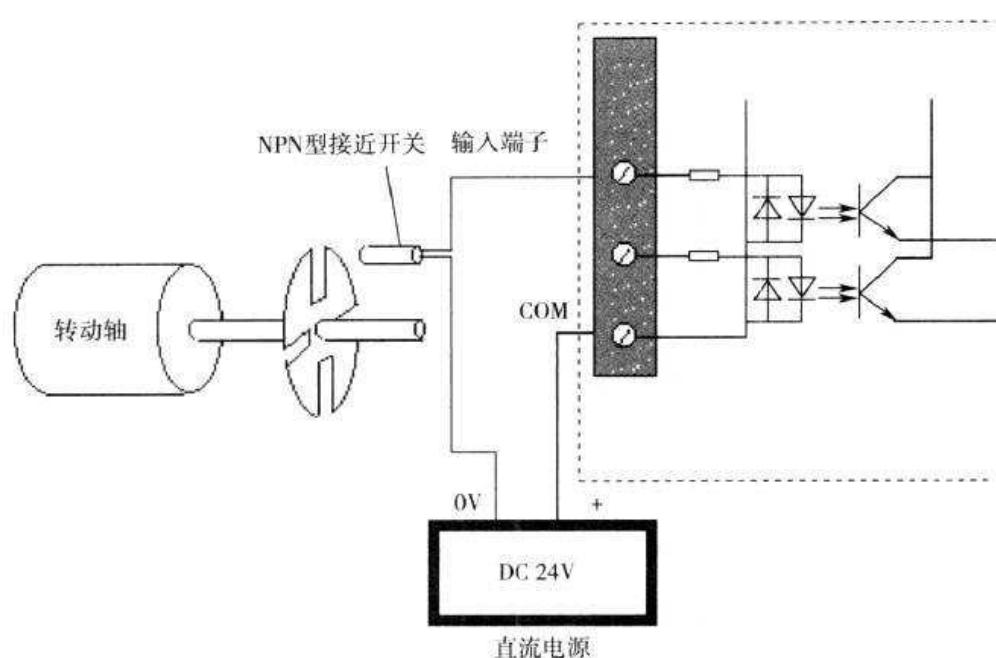


图 6-10 NPN 型传感器的接入方式

2. HSC 硬件组态

图 6-11 所示为 PLC 的高速计数器的硬件组态，它包括常规、功能、重置为初始值、事件组态、硬件输入、IO 地址/硬件标识符。

调用 HSC 的硬件组态具体步骤如下：

第一步，按图 6-12 所示进行功能的设置，包括计数类型可以选择“计数”、“频率”、“运动轴”；运行阶段为“单相”、“双相”、“正交 1X”和“正交 4X”；计数方向为“内部方向控制”、“外部方向控制”；初始计数方向为“加计数”、“减计数”。

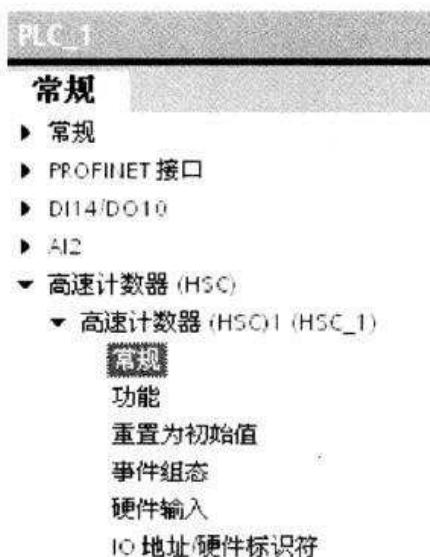


图 6-11 高速计数器硬件组态

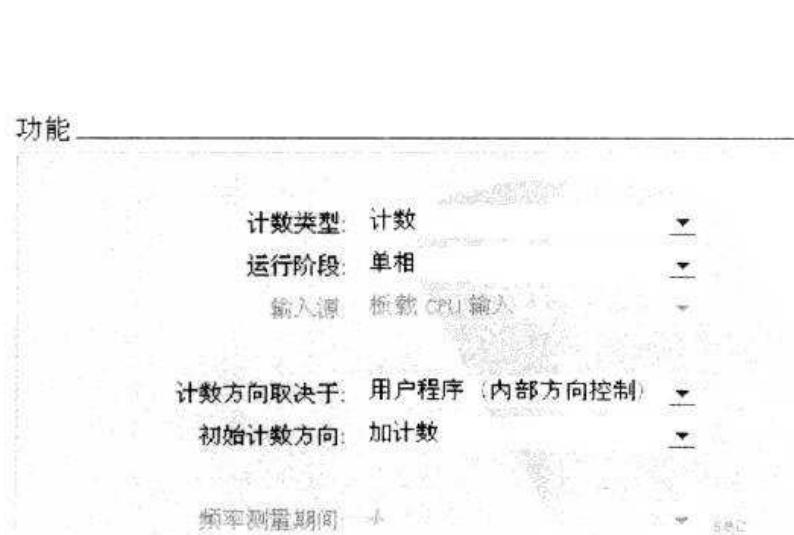


图 6-12 功能设置



图解西门子 S7-1200 PLC 入门到实践

第二步，按图 6-13 所示进行重置为初始值的设置，包括初始值计数器值、初始参考值、复位选项。

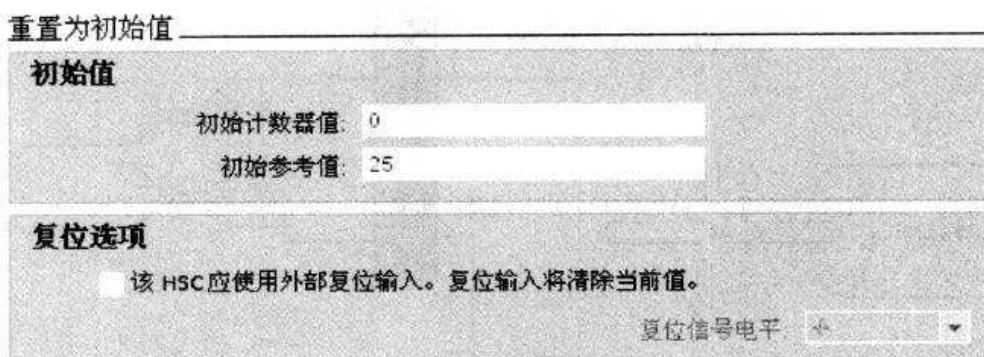


图 6-13 重置为初始值设置

第三步，按图 6-14 所示进行事件组态，共三种，包括为计数器值等于参考值这一事件生成中断、为外部复位事件生成中断、为方向变化事件生成中断。在本案例中，选择第一种事件。

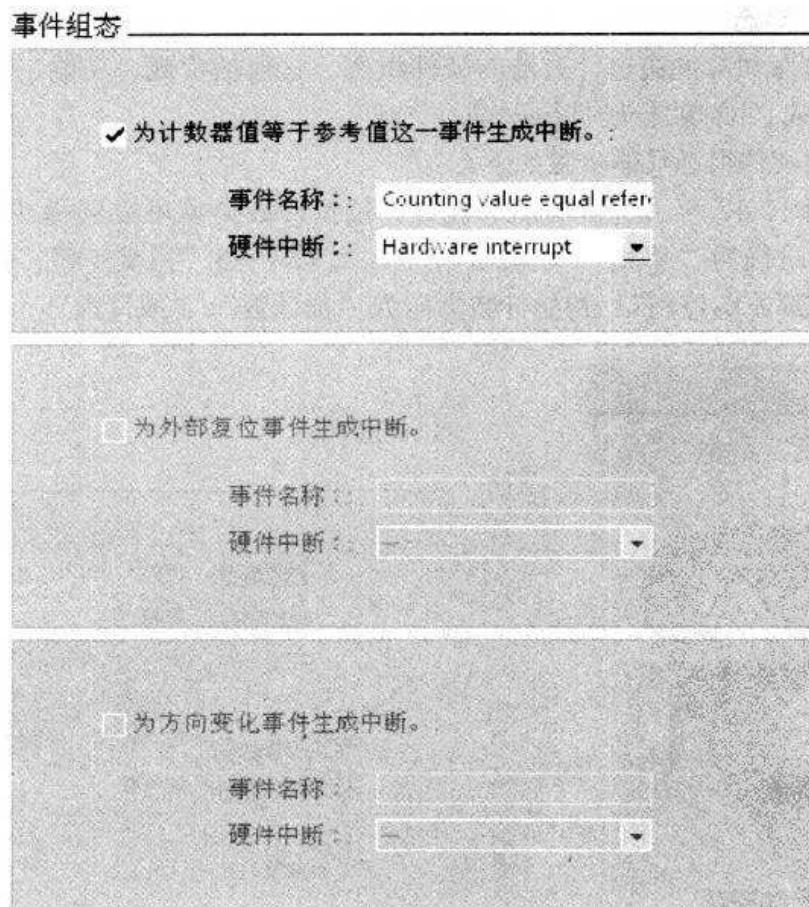


图 6-14 事件组态



第四步，查看图 6-15 所示生成的硬件输入是否与本案例相同，其中时钟输入为 %I0.0，方向为程序内部控制，无复位输入。

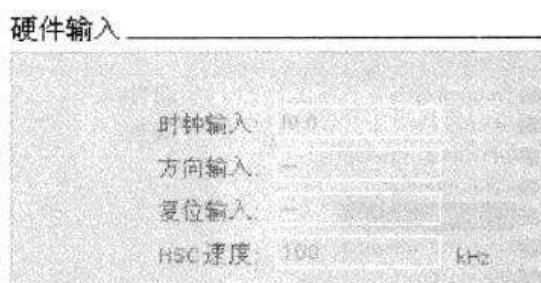


图 6-15 硬件输入

第五步，按图 6-16 所示进行 IO 地址/硬件标识符设置，其中输入地址为默认地址 1000、过程映像为循环 PI、硬件 ID 为 1。

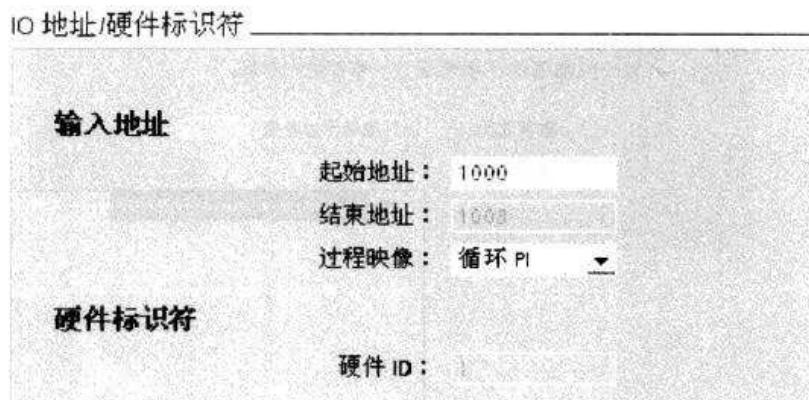


图 6-16 IO 地址/硬件标识符设置

3. 软件编程

根据本案例可以定义的变量见表 6-5。

表 6-5 变量定义

| | 名称 | 数据类型 | 地址 |
|---|-----------|-------|---------|
| 1 | HSC1值 | DWord | %ID1000 |
| 2 | 新参考值 | DInt | %MD10 |
| 3 | 读取HSC1当前值 | DWord | %MD6 |
| 4 | HSC状态 | Word | %MW2 |

由于本案例需要对高速计数器 HSC1 产生中断，即当前值与参考值相同时，需要进行调用中断。图 6-17 所示为添加一个硬件中断（即 Hardware interrupt）组织块 OB200。

即使是 OB200 还是空块的时候，还是可以在图 6-18 所示的硬件组态下的“事件组态”中进行中断定义及选用 OB200。

本案例共分 3 个程序块，具体如图 6-19 所示。

调用 CTRL_HSC 指令所生成的背景数据块 CTRL_HSC_0 的参数值见表 6-6。



图解西门子 S7-1200 PLC 入门到实践

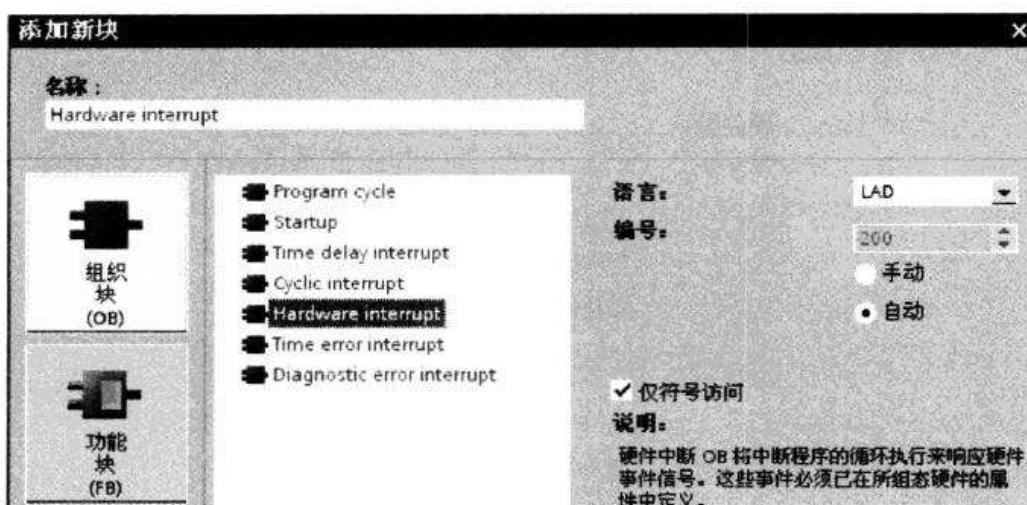


图 6-17 添加硬件中断组织块

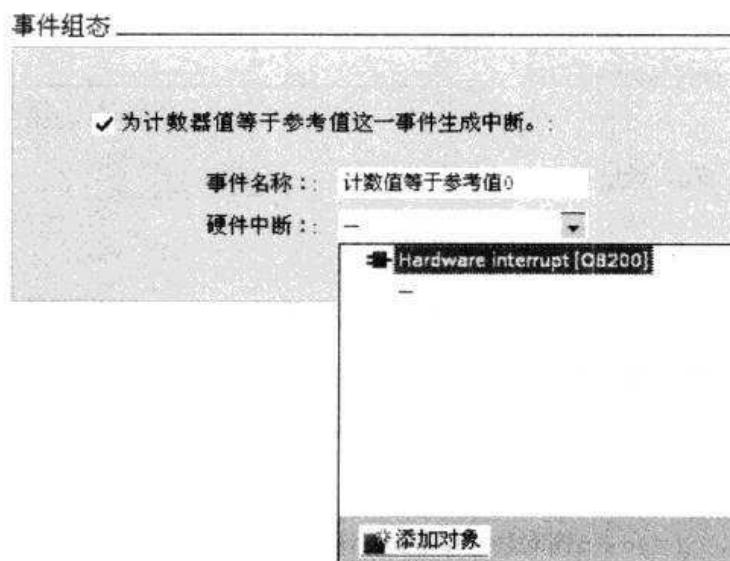


图 6-18 在事件组态中选择 OB200



图 6-19 程序块

第6章 S7-1200 PLC 的 HSC 控制

表 6-6 CTRL_HSC_0 数据块的参数

| | 名称 | 数据类型 | 初始值 |
|----|------------|--------|--------|
| 1 | ▼ Input | | |
| 2 | HSC | HW_HSC | W#16#0 |
| 3 | DIR | Bool | False |
| 4 | CV | Bool | False |
| 5 | RV | Bool | False |
| 6 | PERIOD | Bool | False |
| 7 | NEW_DIR | Int | 0 |
| 8 | NEW_CV | DInt | L#0 |
| 9 | NEW_RV | DInt | L#0 |
| 10 | NEW_PERIOD | Int | 0 |
| 11 | ▼ Output | | |
| 12 | BUSY | Bool | False |
| 13 | STATUS | Word | W#16#0 |

图 6-20 所示为 OB100 上电初始化，即先需要将 HSC1 的新初始值设为 0，并将 HSC1 的参考值设为 200。

▼ 块标题：

上电初始化

▼ 程序段 1：

将HSC1的新初始值设为0

```

      MOVE
      EN      ENO
      0 - IN      "CTRL_HSC_0".
      OUT1 - NEW_CV
  
```

▼ 程序段 2：

将HSC1的新参考值设为200

```

      MOVE
      EN      ENO
      200 - IN      "CTRL_HSC_0".
      OUT1 - NEW_RV
  
```

图 6-20 OB100 上电初始化

图 6-21 所示为 OBI 主程序块，即读取当前计数器值。从该块可以看出，读取 HSC 的值



图解西门子 S7-1200 PLC 入门到实践

不一定就调用 CTRL_HSC 指令，而是直接可以从%ID1000 读取，也就是说，高速计数指令块不是使能高速计数的必要条件。

▼ 块标题：

主程序

▼ 程序段 1：

读取当前计数器HSC1的值

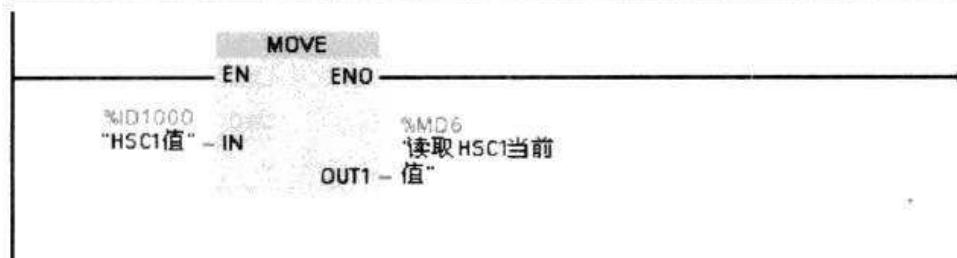


图 6-21 OB1 主程序块

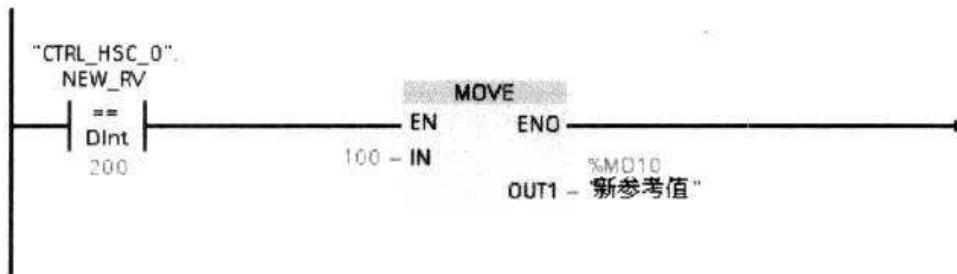
图 6-22 所示为硬件中断组织块 OB200，即将高速计数器的参考值在 200 与 100 之间进行变化。

▼ 块标题：

硬件中断

▼ 程序段 1：

当前参考值为 200 时，则将新参考值设为 100



▼ 程序段 2：

当前参考值为 100 时，则将新参考值设为 200

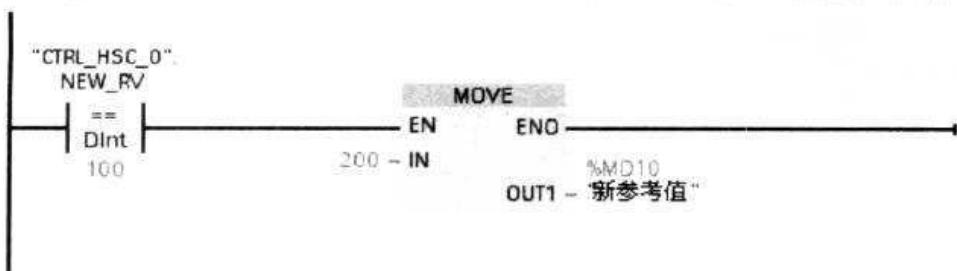


图 6-22 硬件中断组织块 OB200



▼ 程序段 3：

调用 CTRL_HSC 语句

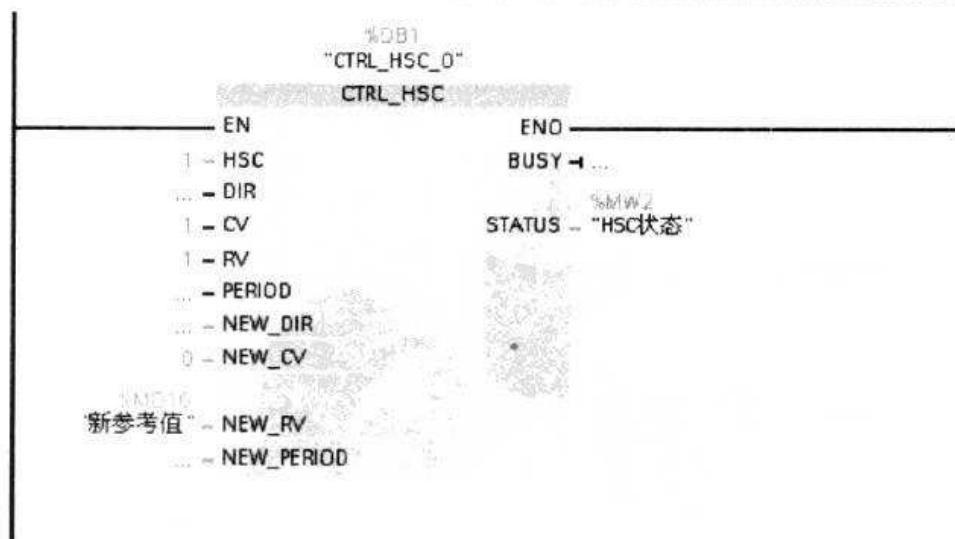


图 6-22 硬件中断组织块 OB200（续）

6.3 技能训练二：A/B 正交模式下的速度/频率的测量

6.3.1 案例介绍

某卷材测速辊的编码器安装方式可以采用低速端安装，如图 6-23 所示。请设计合理的硬件接线方式并编程。

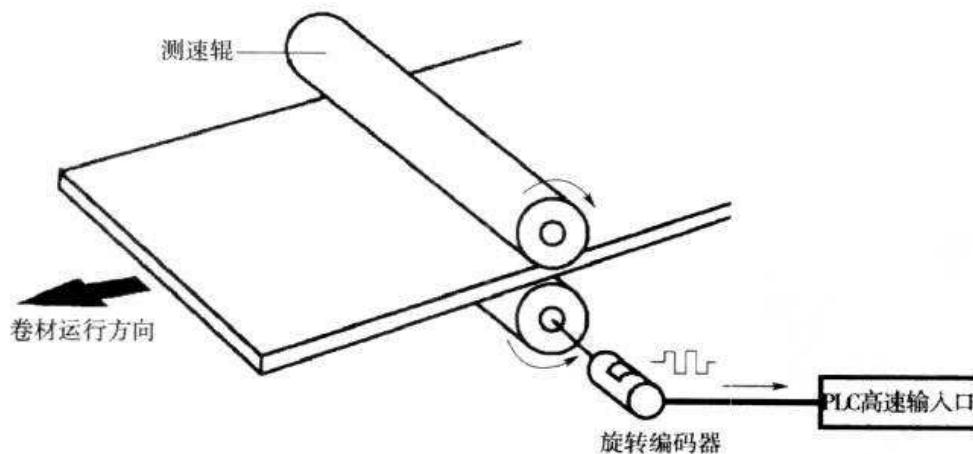


图 6-23 卷材测速辊编码器的安装

6.3.2 硬件组态与软件编程

1. HSC 的频率测量

有些 HSC 模式允许 HSC 被组态（计数类型）为报告频率而非当前脉冲计数值。有三种



图解西门子 S7-1200 PLC 入门到实践

可用的频率测量周期：0.01s、0.1s 或 1.0s。频率测量周期决定 HSC 计算并报告新频率值的频率。报告频率是通过上一测量周期内总计数值确定的平均值。如果该频率在快速变化，则报告值将是介于测量周期内出现的最高频率和最低频率之间的一个中间值。无论频率测量周期的设置是什么，总是会以赫兹为单位来报告频率（每秒脉冲个数）。

以 OMRON E6B2 的编码器为例，其外观如图 6-24 所示，接线见表 6-7。



图 6-24 OMRON E6B2 编码器的外观

表 6-7 编码器与 S7-1200 PLC 的接线

| 序号 | 编码器线号 | S7-1200 PLC 的端子 | 功 能 |
|----|-------|-----------------|---------|
| 1 | 棕色 | 24V | 编码器电源 + |
| 2 | 蓝色 | 0V | 编码器电源 - |
| 3 | 黑色 | %IO.0 | A 相 |
| 4 | 白色 | %IO.1 | B 相 |

需要注意的是，编码器的 NPN 型接线与 PNP 型接线必须与 S7-1200 PLC 的接线相匹配，即 PNP 型时，公共点 M 接的是 0V；NPN 型时，公共点 M 接的是 24V。

2. 硬件组态

由于频率测量与计数测量的硬件组态不一样，必须重新进行组态（见图 6-25）。



图 6-25 功能组态



第一步，选择计数类型为“频率”（见图 6-26）。



图 6-26 计数类型

第二步，选择运行阶段为“AB 相正交计数器 1X”（见图 6-27）。

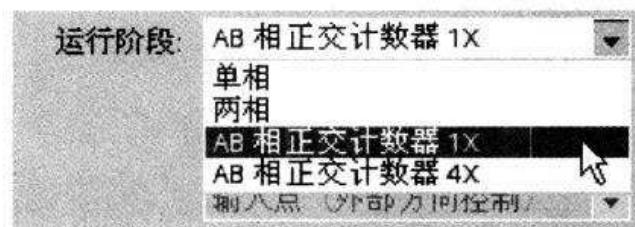


图 6-27 运行阶段

第三步，选择频率测量周期为“1.0sec”（见图 6-28）。

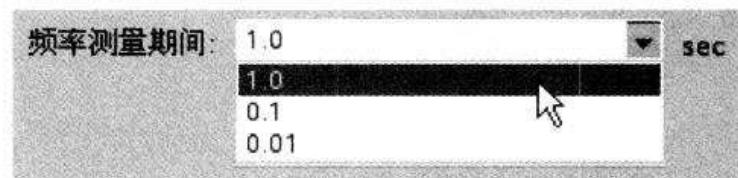


图 6-28 频率测量期间

3. 变量定义与 PLC 梯形图编程

卷材测速系统的变量定义见表 6-8，其中主机启动和停止按钮为% I0.6、% I0.7，主机输出为% Q0.0，HSC1 的地址为% ID1000，实时速度为% MD18。

表 6-8 变量定义

| | 名称 | 数据类型 | 地址 |
|---|-----------|-------|---------|
| 1 | HSC1值 | DWord | %ID1000 |
| 2 | 读取HSC1当前值 | DWord | %MD6 |
| 3 | 停止按钮 | Bool | %I0.7 |
| 4 | 启动按钮 | Bool | %I0.6 |
| 5 | 主机启动 | Bool | %Q0.0 |
| 6 | 中间变量1 | Real | %MD10 |
| 7 | 中间变量2 | Real | %MD14 |
| 8 | 实时速度 | Real | %MD18 |

如图 6-29 所示为主程序。由于频率测量不需要用到 CTRL_HSC 指令，因此其编程相对简单，只需要读取% ID1000 数据，并对其进行数据转换即可。由于本案例的采样周期为 1s，因此只需要将测算的脉冲数除以编码器每转脉冲数再乘以 60s，就可得到转速值。

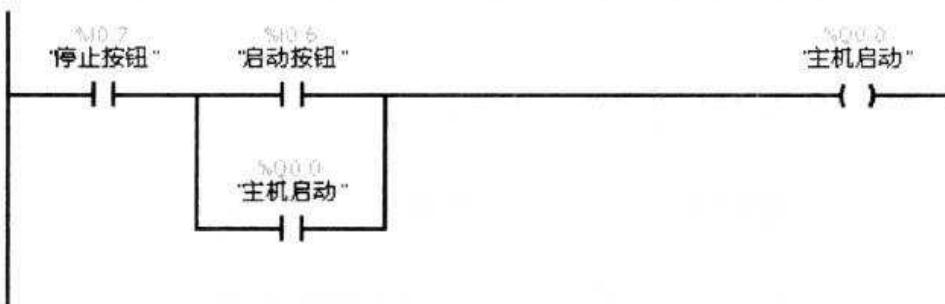


▼ 块标题：

主程序

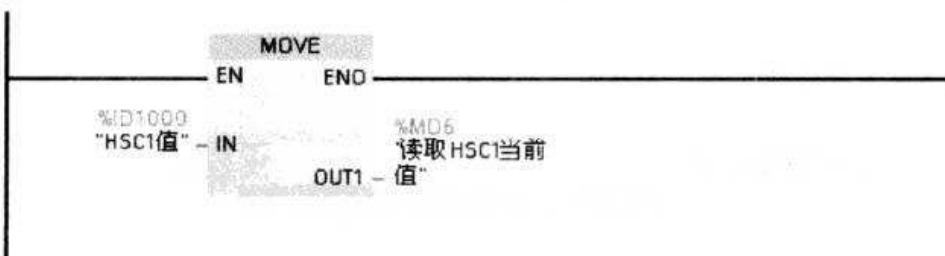
▼ 程序段 1：

通过现场按钮控制主机启动与停止



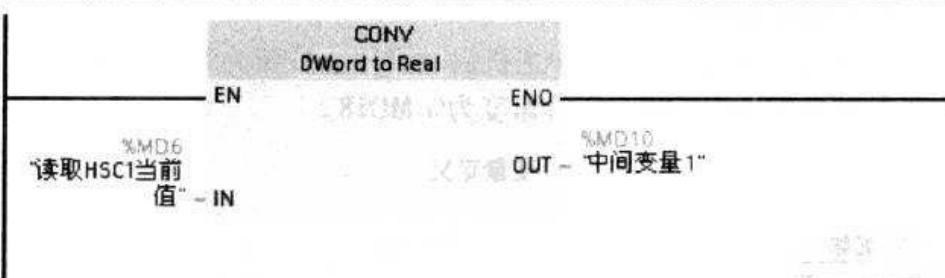
▼ 程序段 2：

直接读取HSC1的频率当前值



▼ 程序段 3：

进行数据数据转换



▼ 程序段 4：

进行换算求出收卷电动机实际运行转速 (r/min)

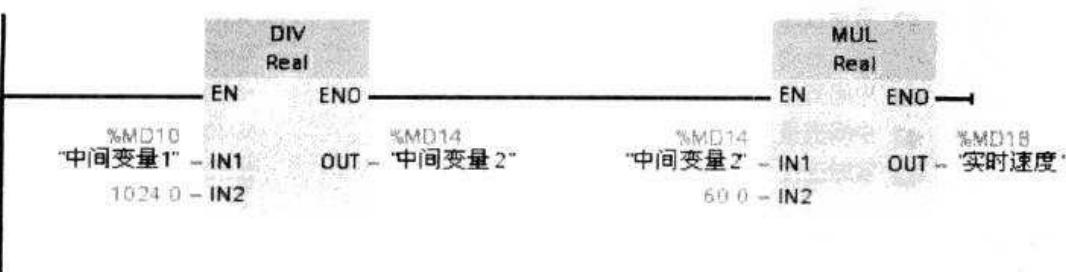


图 6-29 主程序

第6章 S7-1200 PLC 的 HSC 控制



4. 画面设计与组态

图 6-30 所示为卷材测速主画面，其速度值的常规设置如图 6-31 所示，实际测速画面如图 6-32 所示。

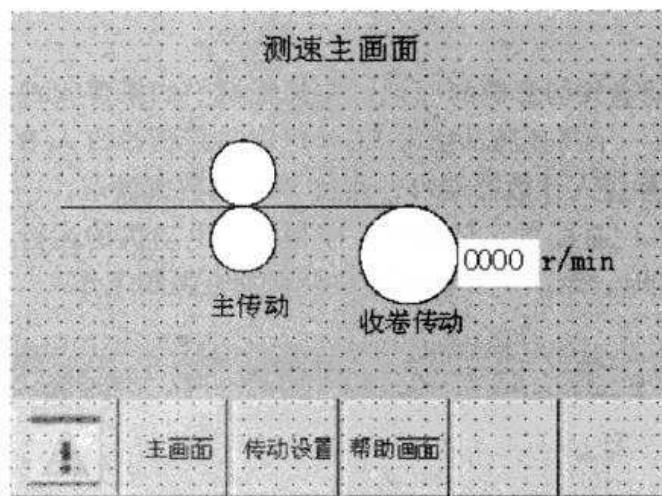


图 6-30 卷材测速主画面

常规

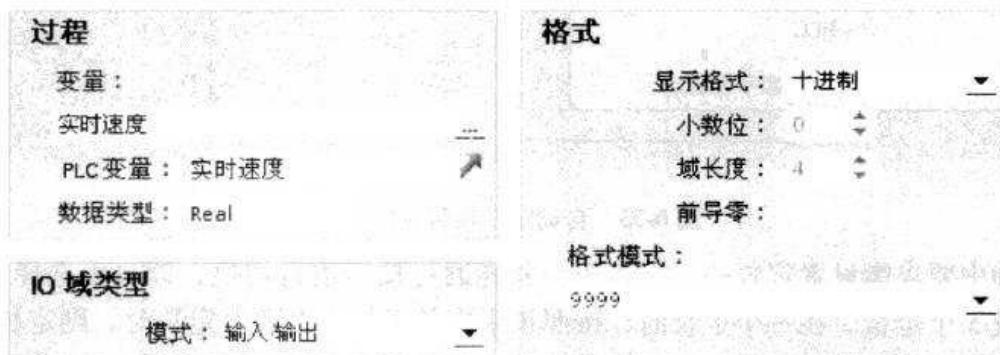


图 6-31 速度值的常规设置

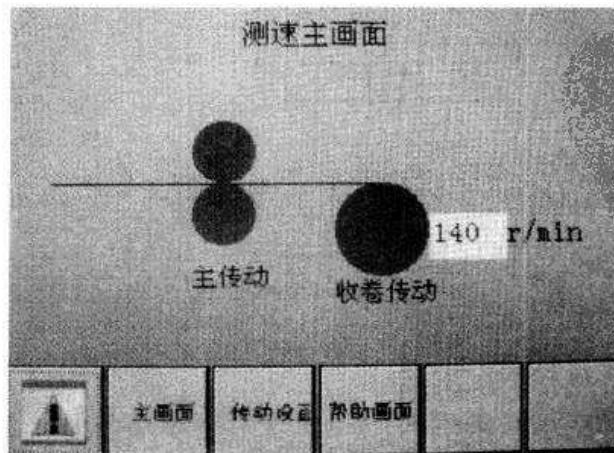


图 6-32 实际测速画面



6.4 工程案例：自动定长切管机

6.4.1 案例介绍

图 6-33 所示为自动定长切管机的原理，在某管材中如果要实现定长切割，则可以在主传动后的计长辊处安装一个高速脉冲信号 EC（可以是编码器的 A/B 正交信号，也可以是接近开关信号单脉冲），并接入计数器端口，通过 PLC 的数据设定，一旦 PLC 接收到启动命令（数字量输入信号），PLC 通过数字量输出，控制牵引电动机的运行，此时计数器指令开始工作，在长度脉冲达到时，牵引电动机停止并使切割电磁阀工作。

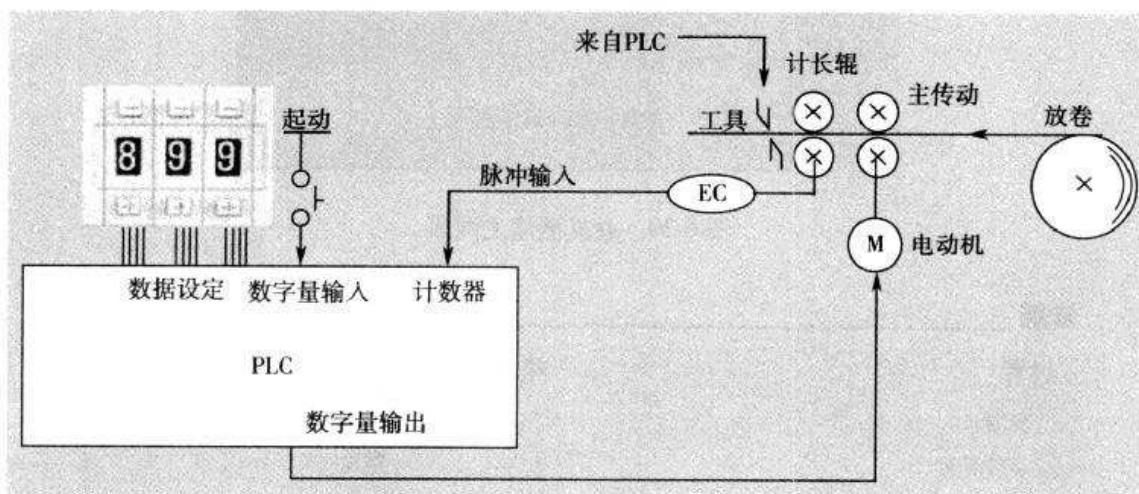


图 6-33 自动定长切管机的原理

在控制中要求能设置定长一、二、三，并能通过按钮进行切换，即自由选择一个定长、两个定长或 3 个定长。在一个定长时，按照正常定长运行；在两个定长时，则定长计数按照定长一、定长二间隔进行运行；在 3 个定长时，也按照 3 个定长循环进行，如图 6-34 所示。

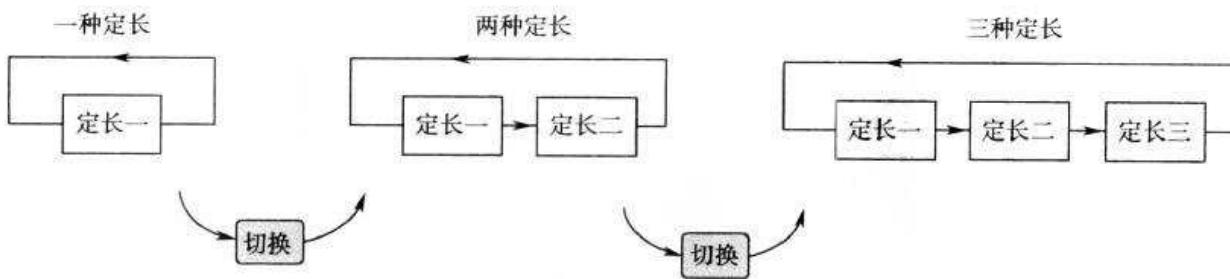


图 6-34 切换原理

6.4.2 硬件组态与软件编程

1. 硬件组态

本案例采用 A/B 相 Z 正交高速脉冲，分别接入到 %I0.0 和 %I0.1，且采用实际值与参



考值相等的中断事件进行。具体硬件设置如图 6-35 ~ 图 6-37 所示。

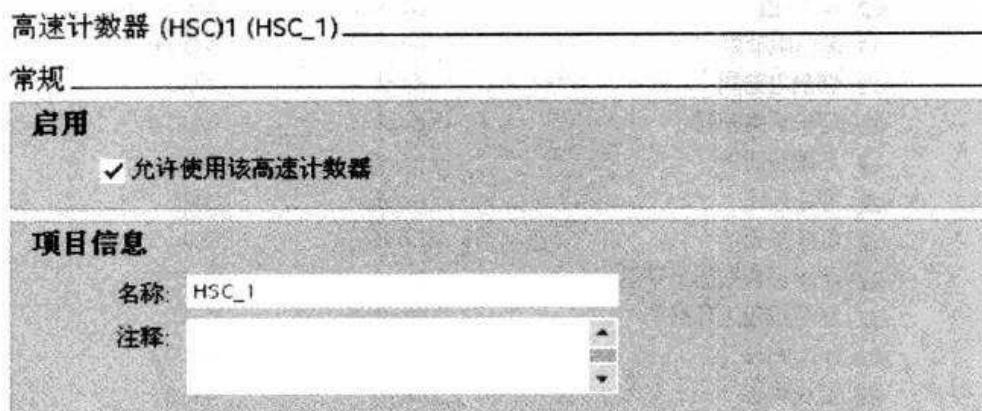


图 6-35 HSC_1 的常规设置

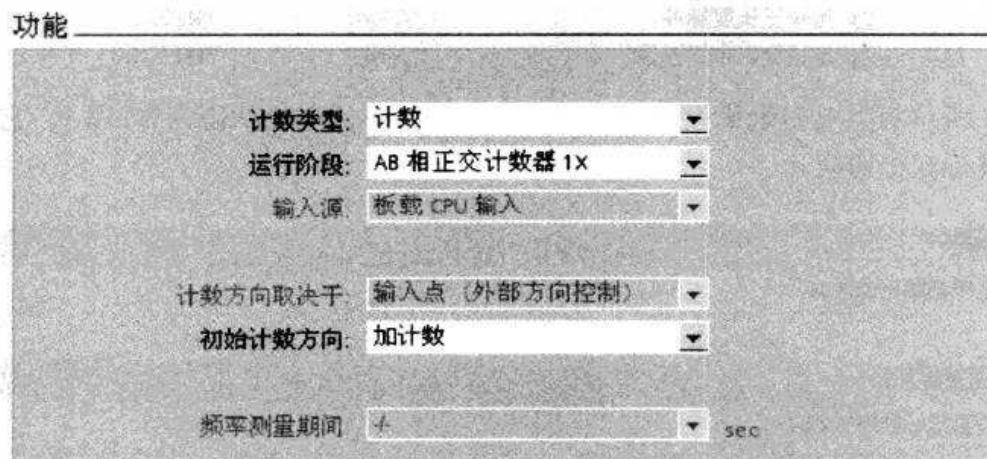


图 6-36 功能设置

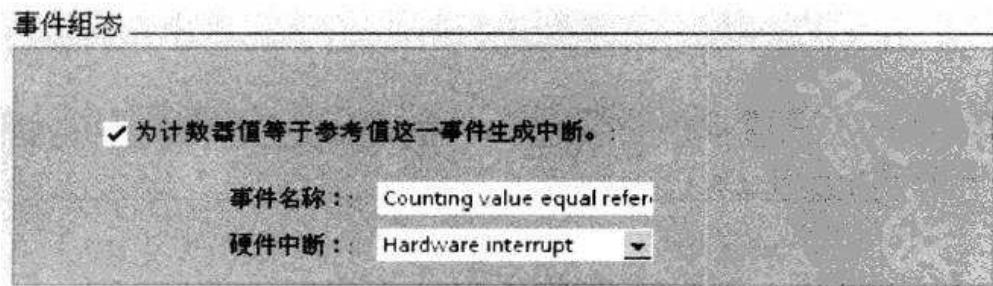


图 6-37 事件组态

2. 变量定义

定长控制的变量定义见表 6-9，包括计长脉冲信号 HSC1% ID1000、牵引电动机% Q0.0 和切割电磁阀输出% Q0.1、触摸屏各种按钮、定长一二三的设定值等。



表 6-9 变量定义

| | 名称 | 数据类型 | 地址 ▲ |
|----|--------------|-------|---------|
| 1 | →HSC1值 | DWord | %ID1000 |
| 2 | →牵引电动机 | Bool | %Q0.0 |
| 3 | →切割电磁阀 | Bool | %Q0.1 |
| 4 | →定长切换按钮 | Bool | %M0.0 |
| 5 | →启动按钮 | Bool | %M0.1 |
| 6 | →停止按钮 | Bool | %M0.2 |
| 7 | →复位按钮 | Bool | %M0.3 |
| 8 | →定长切换按钮上升沿 | Bool | %M1.0 |
| 9 | →复位按钮上升沿 | Bool | %M1.1 |
| 10 | →HSC状态 | Word | %MW2 |
| 11 | →定长状态 | Word | %MW4 |
| 12 | →读取 HSC1 当前值 | DWord | %MD6 |
| 13 | →新参考值 | DInt | %MD10 |
| 14 | →定长一长度脉冲 | DWord | %MD14 |
| 15 | →定长二长度脉冲 | DWord | %MD18 |
| 16 | →定长三长度脉冲 | DWord | %MD22 |
| 17 | →定长状态中间变量 | Word | %MW26 |

图 6-38 所示为上电初始化程序 OB100。在本案例中，定长状态为 %MW4，它可以在 1、2、3 之间进行互相切换。

▼ 块标题：

上电初始化OB100

▼ 程序段 1：

设置定长状态 %MW4=1



▼ 程序段 2：

设置当前 CV=0

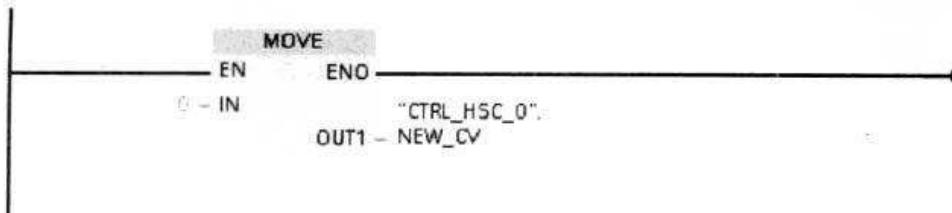


图 6-38 上电初始化程序 OB100



▼ 程序段 3：

设置当前RV=HMI定义的定长一数值

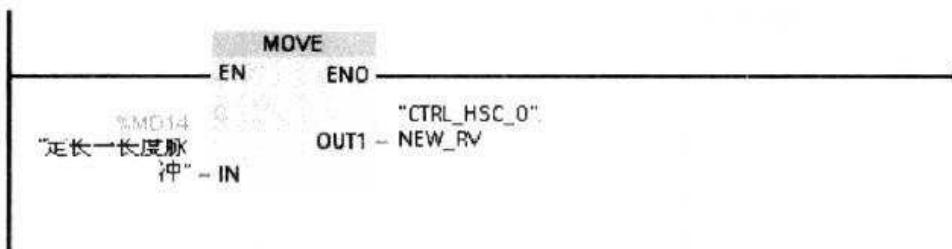


图 6-38 上电初始化程序 OB100（续）

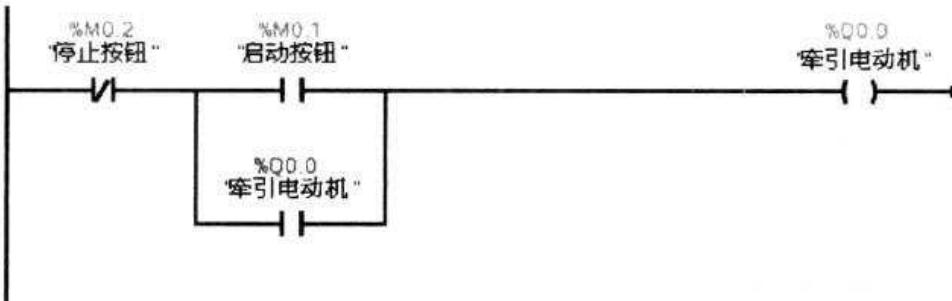
图 6-39 所示为主程序 OB1。需要注意的是，定长状态切换只能在牵引电动机不在运行的时候才可以。

▼ 块标题：

主程序OB1

▼ 程序段 1：

牵引电动机启动与停止



▼ 程序段 2：

定长状态处理

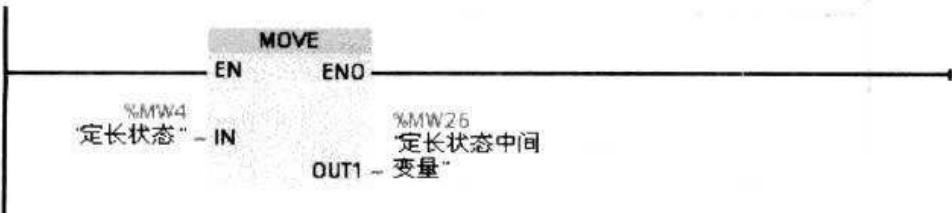
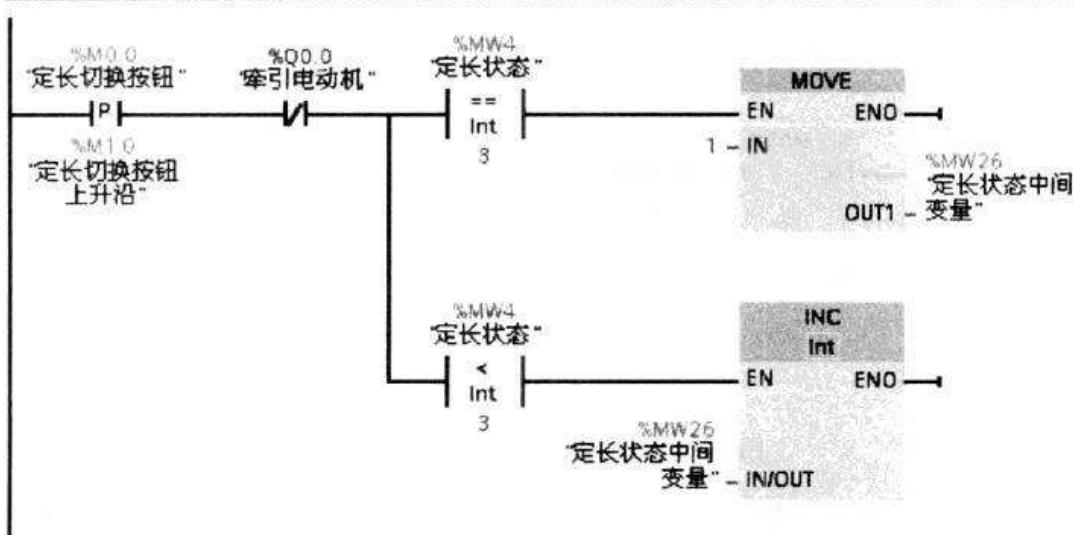


图 6-39 主程序 OB1



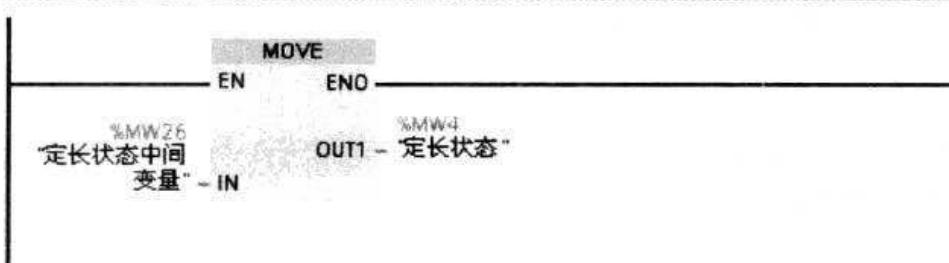
▼ 程序段 3：

定长切换按钮的动作



▼ 程序段 4：

定长状态处理



▼ 程序段 5：

复位按钮动作处理，即类似上电初始化

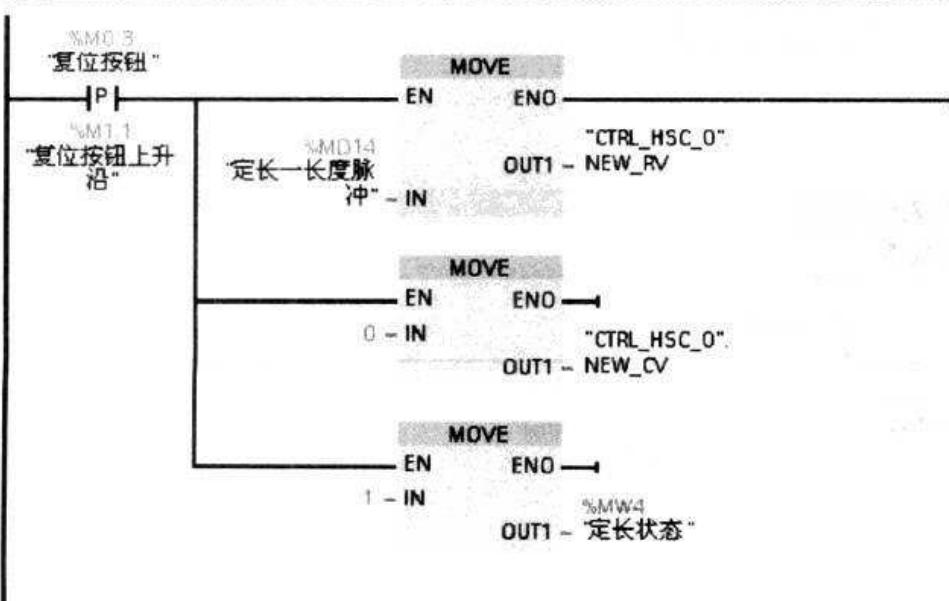
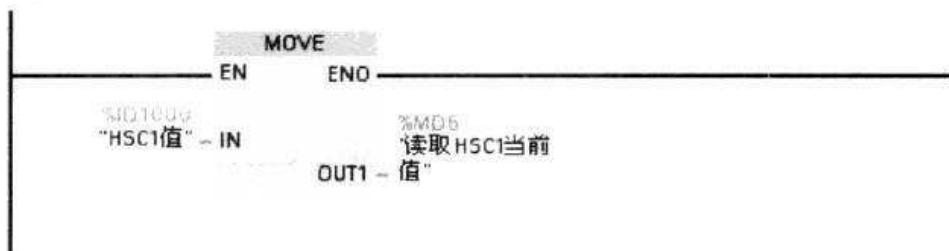


图 6-39 主程序 OB1（续一）

第6章 S7-1200 PLC 的 HSC 控制

▼ 程序段 6：

读取当前HSC脉冲数



▼ 程序段 7：

注释

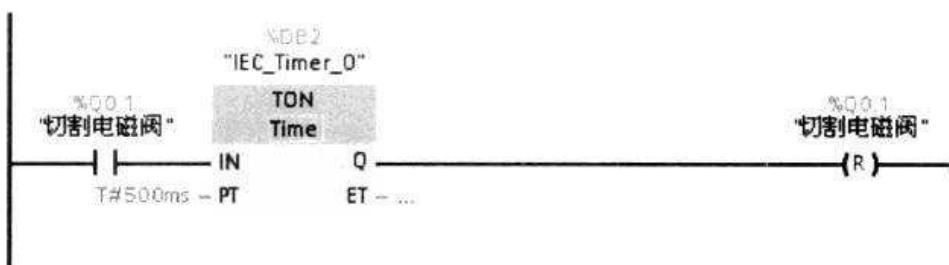


图 6-39 主程序 OBI (续二)

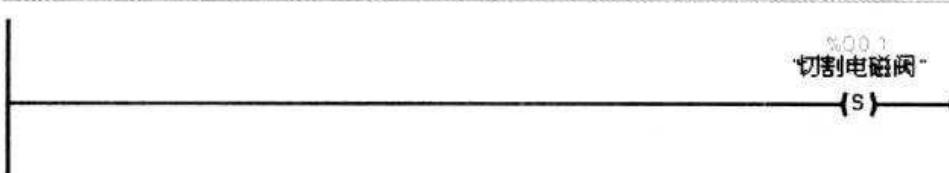
图 6-40 所示为中断程序。对于切割电磁阀的处理是：当硬件中断 CV = RV 时，开始置

▼ 块标题：

硬件中断CV=RV

▼ 程序段 1：

切割电磁阀动作



▼ 程序段 2：

1种定长状态

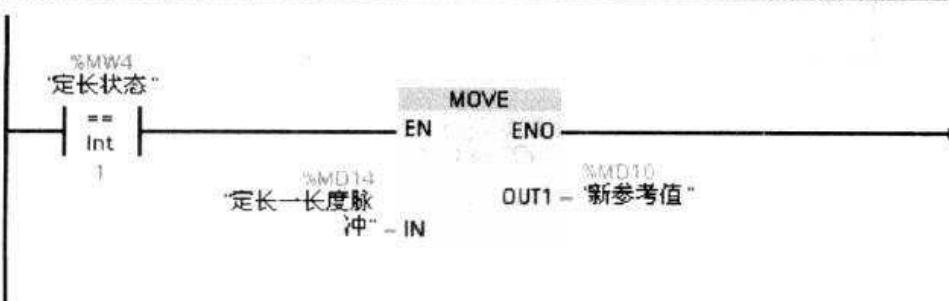


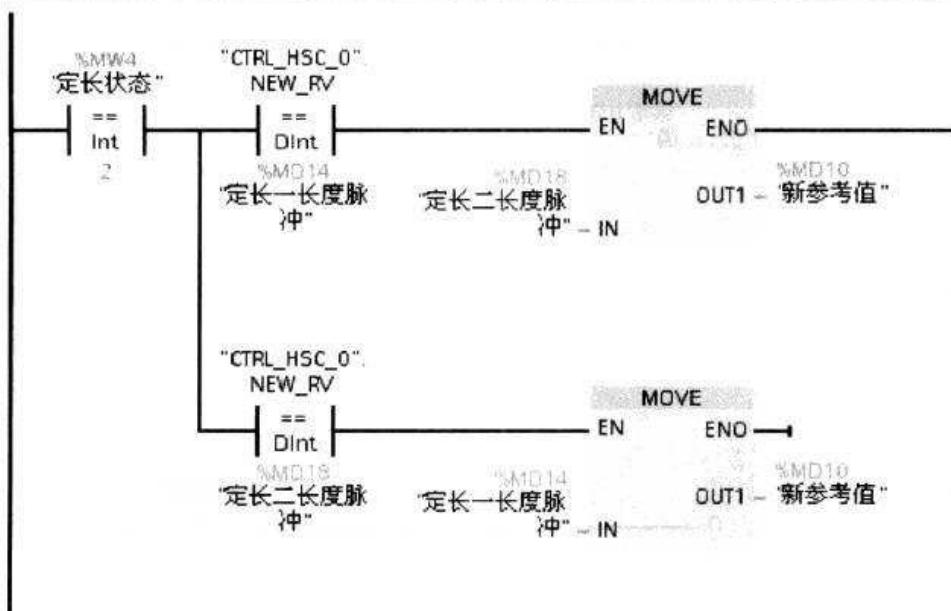
图 6-40 中断程序



图解西门子 S7-1200 PLC 入门到实践

▼ 程序段 3：

2种定长状态



▼ 程序段 4：

3种定长状态

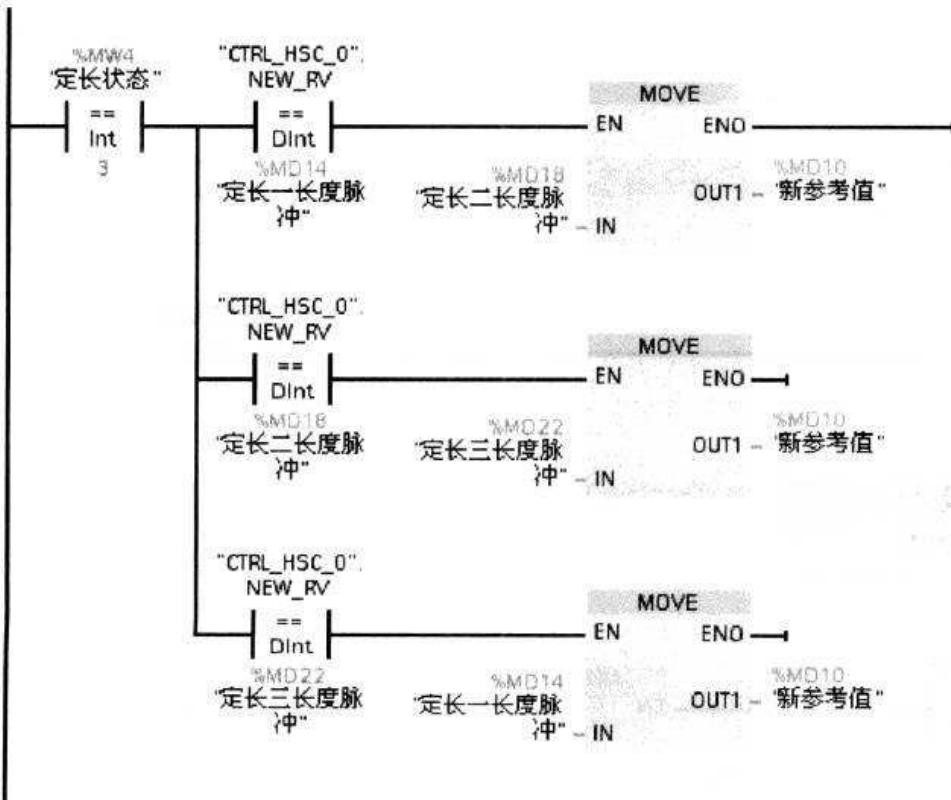


图 6-40 中断程序（续一）



▼ 程序段 5：

调用 CTRL_HSC 指令

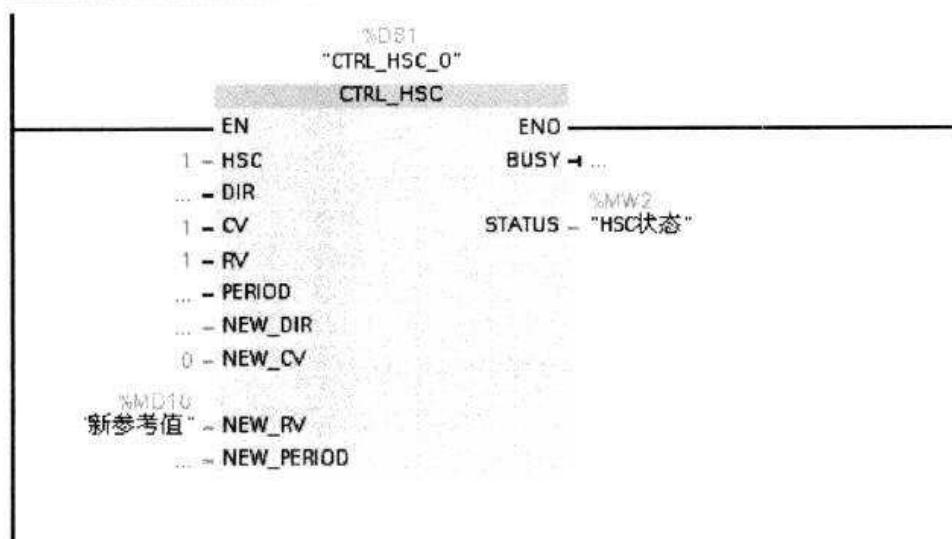


图 6-40 中断程序（续二）

位电磁阀，等延时 0.5s 在主程序中将切割电磁阀复位。三种定长状态的切换则是根据不同状态下定长状态脉冲的值来进行切换。

3. 画面组态

图 6-41 所示为主画面。

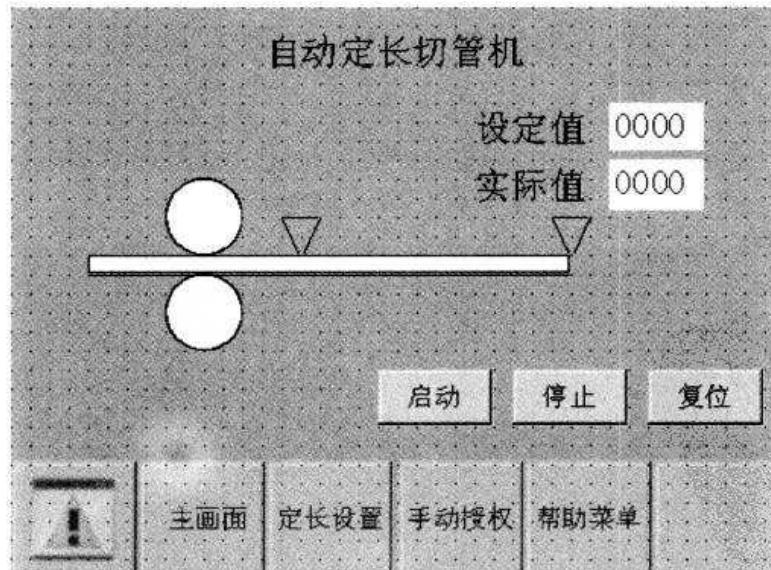


图 6-41 主画面

图 6-42 所示为定长设置画面。

图 6-43 所示为手动授权画面。

图 6-44 所示为帮助画面。

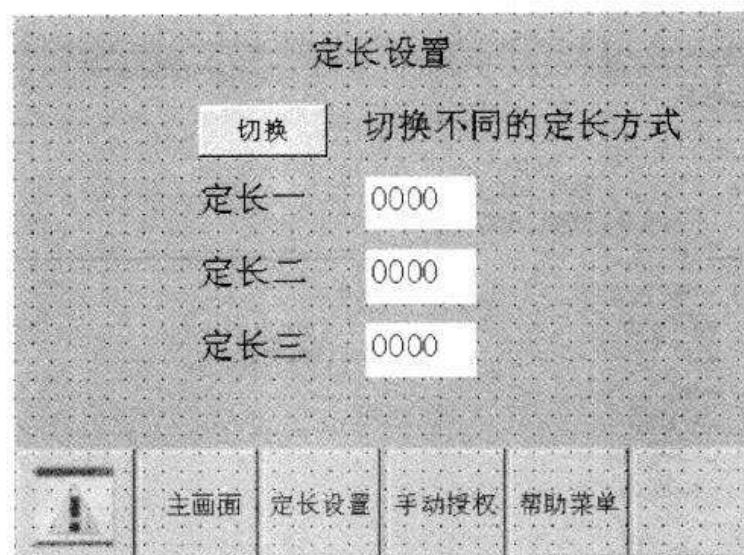


图 6-42 定长设置画面

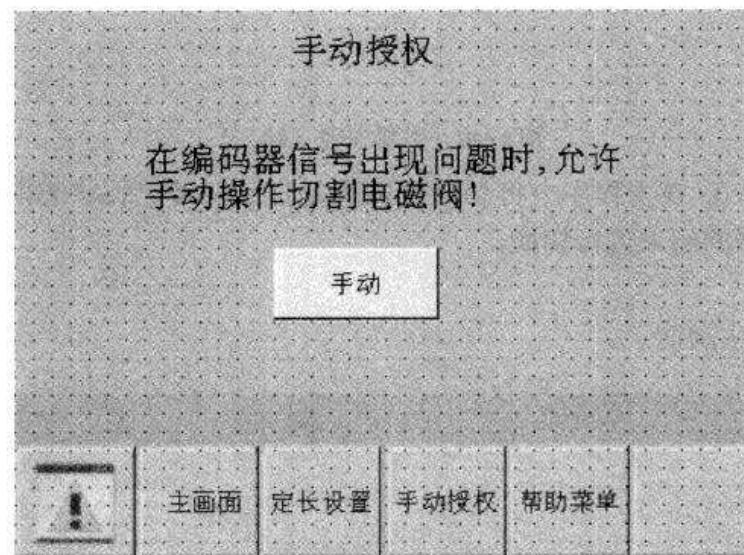


图 6-43 手动授权画面

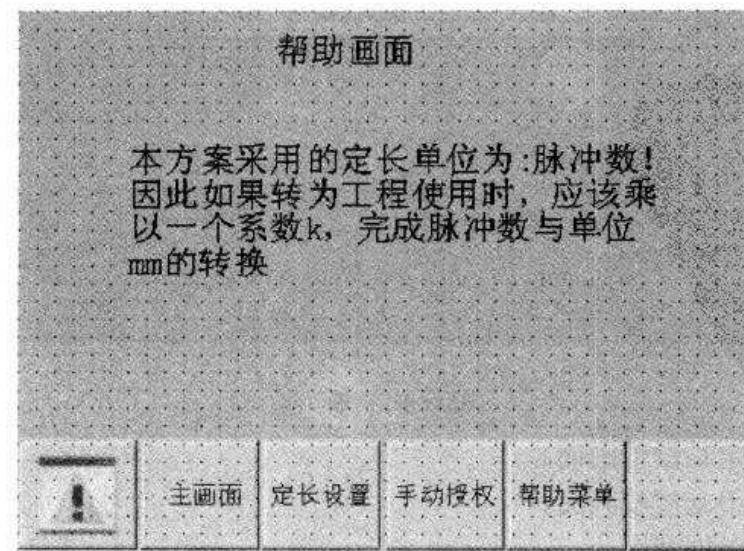


图 6-44 帮助画面

第6章 S7-1200 PLC 的 HSC 控制



为了形象地表达复卷机的实际长度，可以在主画面中对复卷的长度进行动作设置，如图 6-45 所示的“水平移动”，即从起始位置到目标位置，按照比例进行移动。

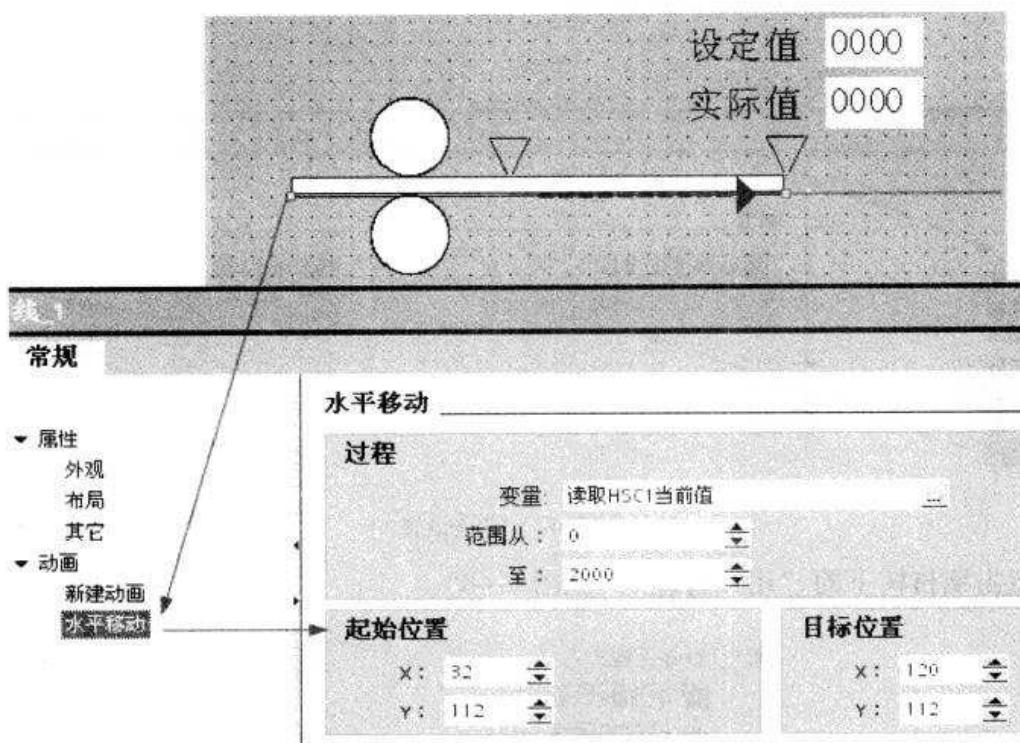


图 6-45 实际长度脉冲的动画

在定长设置画面中，定长一数据的常规设置如图 6-46 所示。

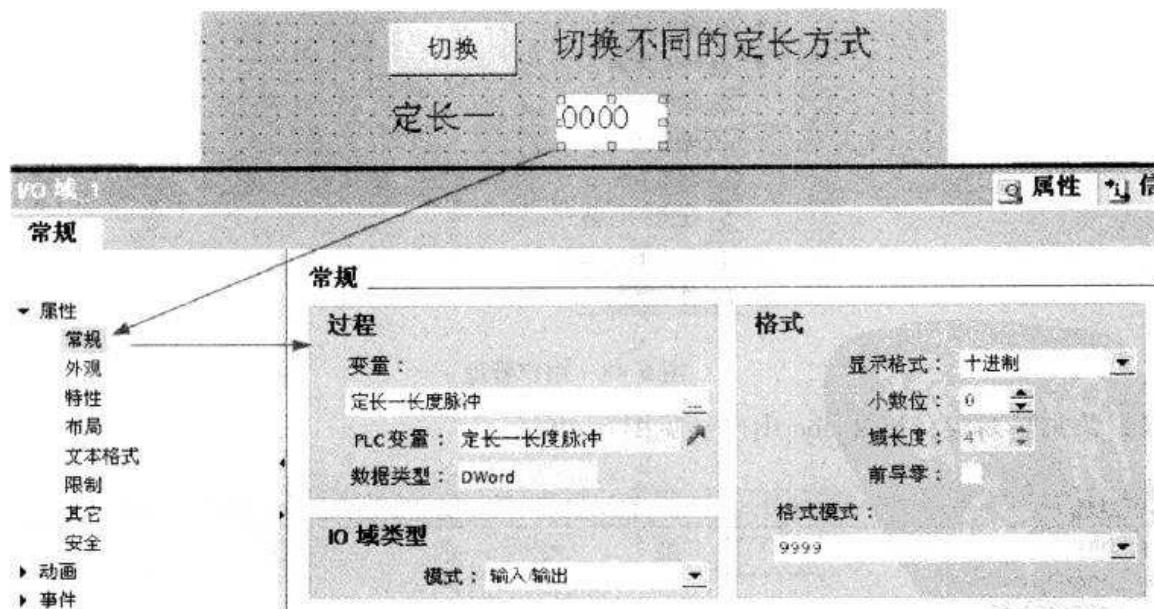


图 6-46 定长一数据的常规设置

在图 6-47 中，为确保手动切割电磁阀能被正常操作，需要输入密码，这就需要对其进行“安全”设置，其权限为“用户管理”。

安全设置的操作步骤如下：

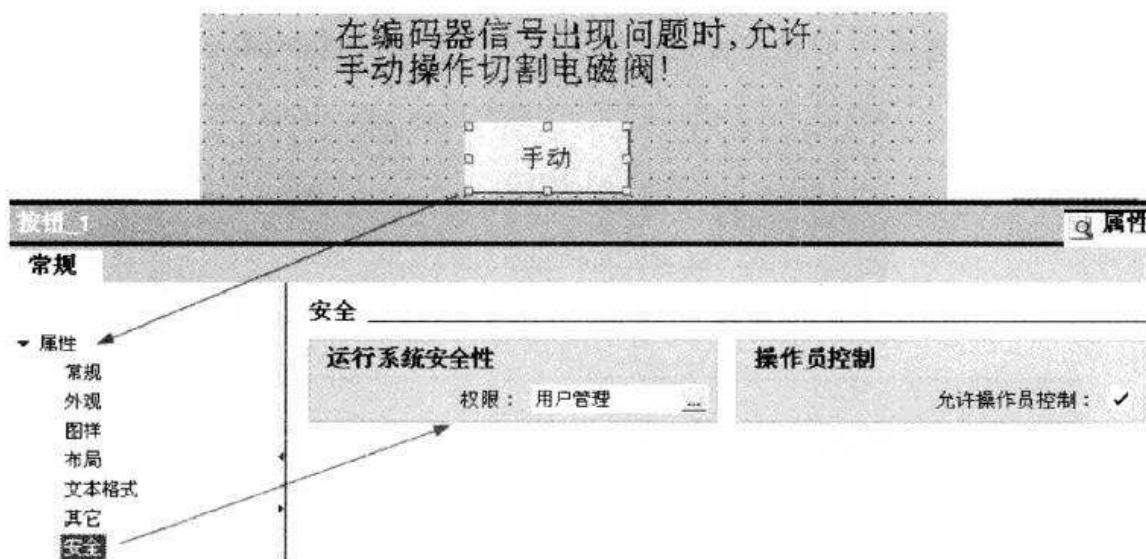


图 6-47 安全设置

1) 点击项目树下的“用户管理”(见图 6-48)。



图 6-48 用户管理

2) 添加新对象，增加 abc 用户(见图 6-49)。

| 名称 | 密码 | 自动注销 | 退出登录时间 | 编号 | 注释 |
|---------------|-------|-------------------------------------|--------|----|----------------------|
| abc | ***** | <input checked="" type="checkbox"/> | 5 | 2 | |
| Administrator | ***** | <input checked="" type="checkbox"/> | 5 | 1 | 将用户 'Administrat...' |
| <添加新对象> | | | | | |

图 6-49 增加 abc 用户

3) 对新增用户进行密码输入和确认(见图 6-50)。

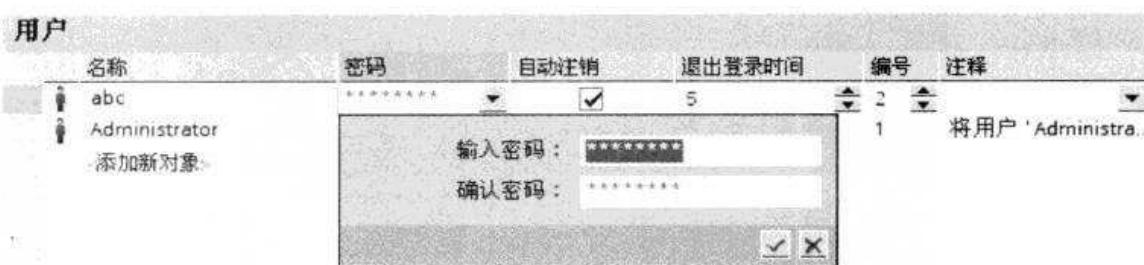


图 6-50 abc 用户的密码确认

4. 实际测试画面

实际测试的画面如图 6-51 ~ 图 6-54 所示。

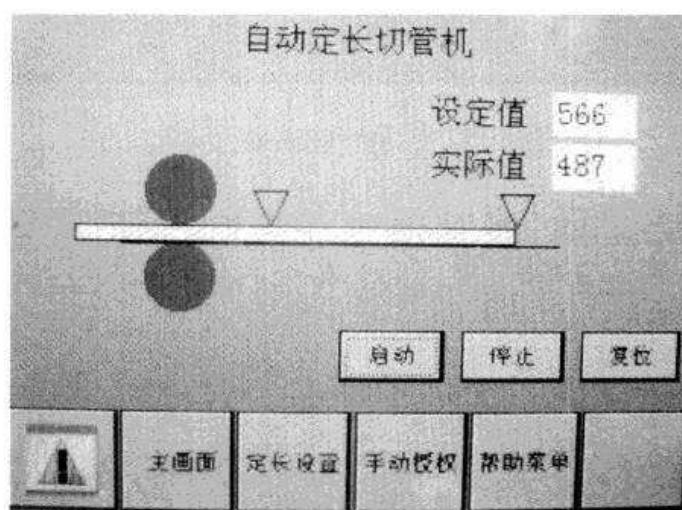


图 6-51 自动定长切管机设置画面

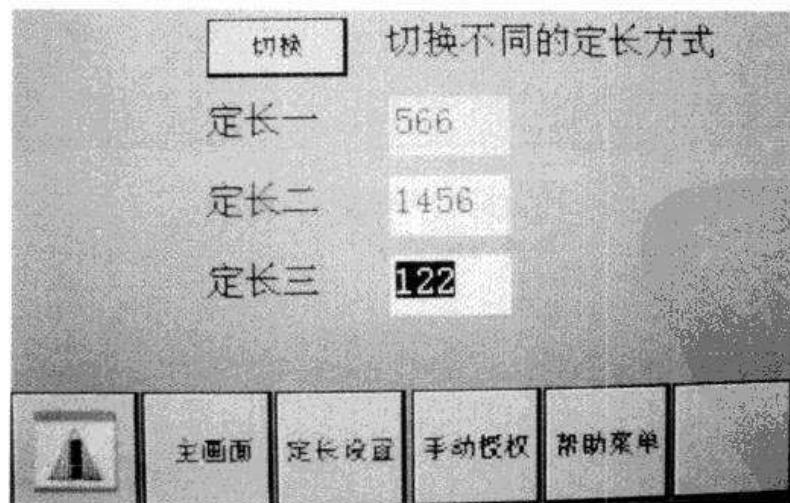


图 6-52 定长设置画面

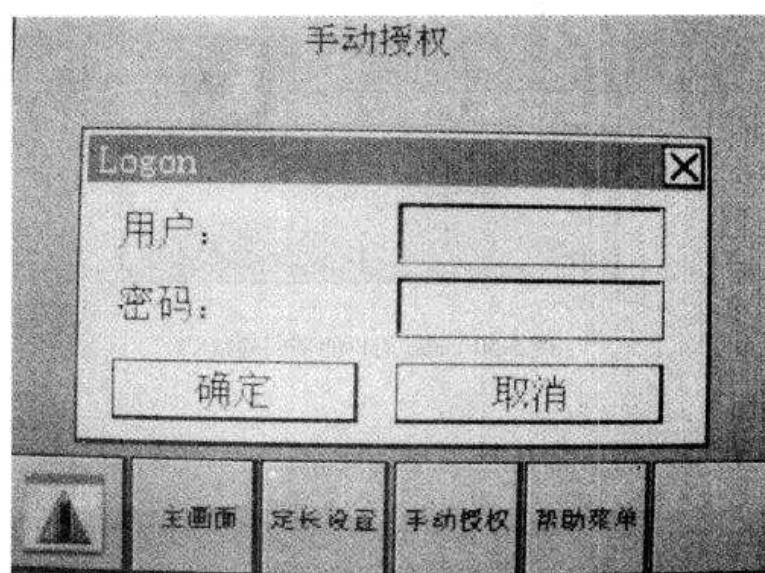


图 6-53 手动密码授权输入画面

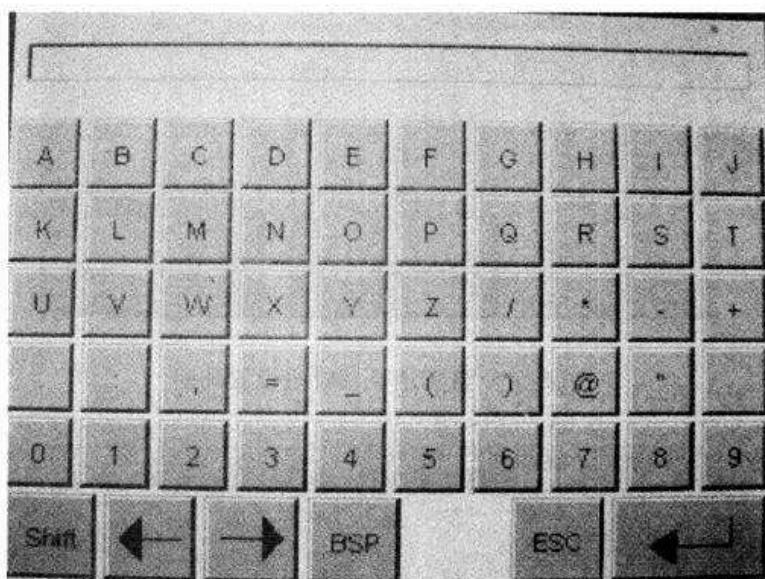
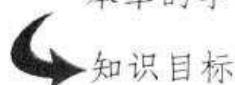


图 6-54 密码输入窗口

第7章 S7-1200 PLC的PTO控制

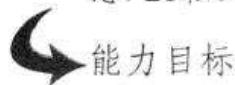
运动控制是电气控制的一个分支，它使用通称为伺服机构的一些设备，如液压泵、线性执行机或者是电动机来控制机器的位置和/或速度。运动控制在机器人和数控机床的领域内的应用要比在专用机器中的应用更复杂，因为后者的运动形式更简单，通常被称为通用运动控制。运动控制被广泛应用在包装、印刷、纺织和装配工业中。S7-1200 PLC独有轴工艺对象，通过硬件组态、面板调试和诊断功能即可实现在线手动控制，最后通过调用相关指令实现自动控制。

本章的学习目标如下：



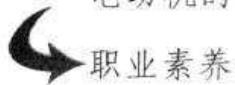
知识目标

了解S7-1200 PLC的PTO功能的硬件配置；了解运动控制指令及其应用；熟悉PLC轴工艺对象的概念及其应用。



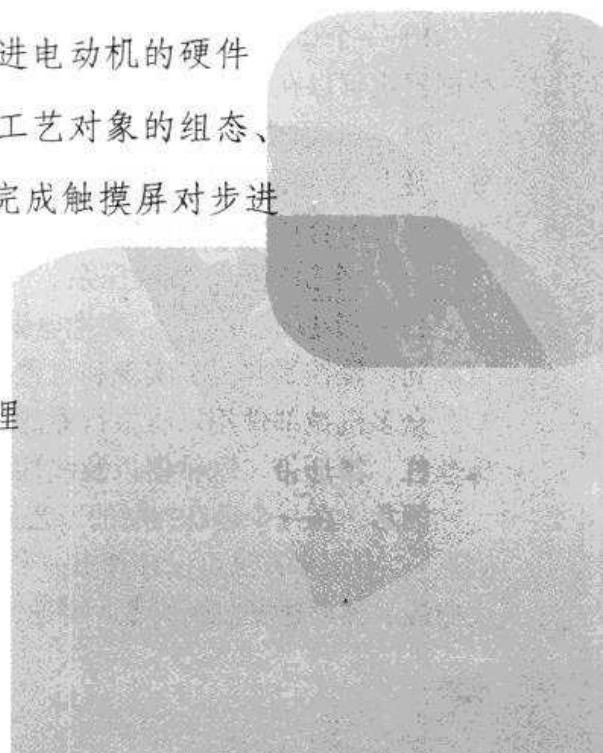
能力目标

能独立进行S7-1200 PLC与步进驱动器、步进电动机的硬件接线，并在此基础上对S7-1200 PLC进行轴工艺对象的组态、调试和诊断，同时调用相关运动控制指令完成触摸屏对步进电动机的控制。



职业素养目标

熟悉工艺对象，构建自动化为工艺服务的理念。





7.1 背景知识介绍：运动控制与步进电动机

7.1.1 运动控制的基本架构

运动控制是电气控制的一个分支，它使用通称为伺服机构的一些设备，如液压泵、线性执行机或者是电动机来控制机器的位置和/或速度。运动控制在机器人和数控机床的领域内的应用要比在专用机器中的应用更复杂，因为后者的运动形式更简单，通常被称为通用运动控制。运动控制被广泛应用在包装、印刷、纺织和装配工业中。

一个运动控制系统的基本架构（见图 7-1）组成包括：

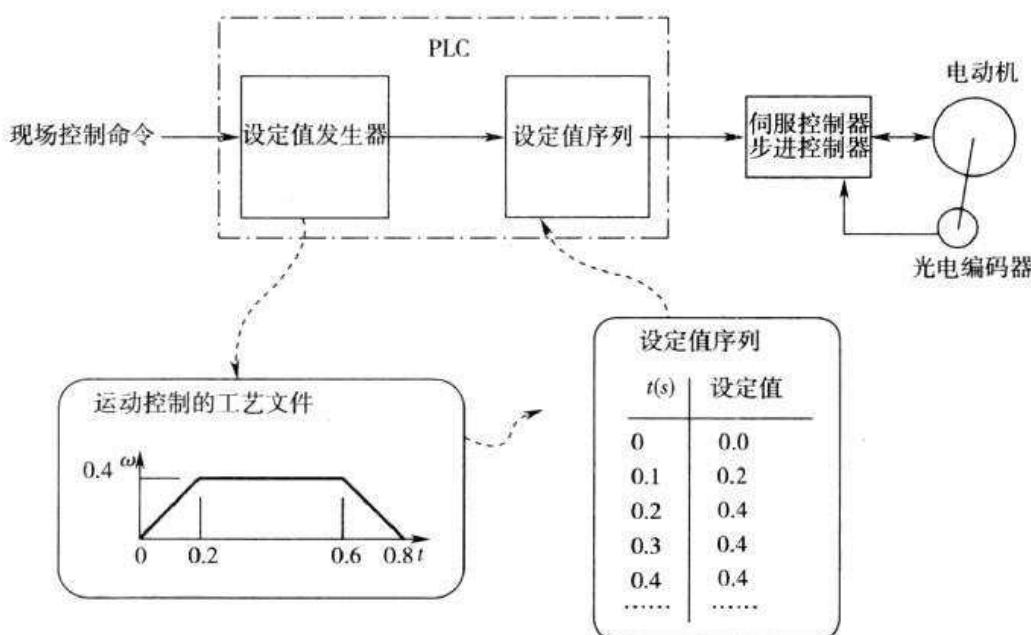


图 7-1 运动控制系统的基本架构

1) 一个运动控制器（如 PLC）用以生成轨迹点（期望输出）和闭合位置反馈环。许多控制器也可以在内部闭合一个速度环。

2) 一个驱动或放大器用（如伺服控制器和步进控制器）以将来自运动控制器的控制信号（通常是速度或扭矩信号）转换为更高功率的电流或电压信号。更为先进的智能化驱动可以自身闭合位置环和速度环，以获得更精确的控制。

3) 一个执行器，如液压泵、气缸、线性执行机或电动机用以输出运动。

4) 一个反馈传感器，如光电编码器、旋转变压器或霍尔效应设备等用以反馈执行器的位置到位置控制器，以实现和位置控制环的闭合。

众多机械部件用以将执行器的运动形式转换为期望的运动形式，它包括齿轮箱、轴、滚珠丝杠、齿形带、联轴器以及线性和旋转轴承。

通常，在一个运动控制的工艺文件中，其功能主要包括：1) 速度控制；2) 点位控制（点到点）；有很多方法可以计算出一个运动轨迹，它们通常基于一个运动的速度曲线，如三角速度曲线、梯形速度曲线或者 S 型速度曲线；3) 电子齿轮（或电子凸轮），也就是从动轴的位置

第7章 S7-1200 PLC 的 PTO 控制



在机械上跟随一个主动轴的位置变化。举一个简单的例子就是，一个系统包含两个转盘，它们按照一个给定的相对角度关系转动。电子凸轮较之电子齿轮更复杂一些，它使得主动轴和从动轴之间的随动关系曲线是一个函数。这个曲线可以是非线性的，但必须是一个函数关系。

从运动控制的基本架构可以看到，PLC 作为一种典型的运动控制核心，起到了非常重要的作用，这主要归因于 PLC 具有高速脉冲输入、高速脉冲输出和运动控制模块等软硬件功能。

7.1.2 S7-1200 PLC 实现运动控制的基础

S7-1200 PLC 可以实现运动控制的基础在于集成了高速计数口、高速脉冲输出口等硬件和相应的软件功能。尤其是 S7-1200 PLC 在运动控制中使用了轴的概念，通过对轴的组态，包括硬件接口、位置定义、动态特性、机械特性等，与相关的指令块（符合 PLCoPerm 规范）组合使用，可实现绝对位置、相对位置、点动、转速控制及自动寻找参考点的功能。

如图 7-2 所示为 S7-1200 PLC 的运动控制应用，即 CPU 输出脉冲（即脉冲串输出，Pulse Train Output, PTO）和方向到驱动器（步进或伺服），驱动器再将从 CPU 输入的给定值进行处理，然后输出到步进电动机或伺服电动机，控制电动机加速、减速和移动到指定位置。需要注意的是，S7-1200 PLC 内部的高速计数器是测量 CPU 上的脉冲输出（类似于编码器信号）来计算速度和当前位置，并非实际电动机编码器所反馈的实际速度和位置。

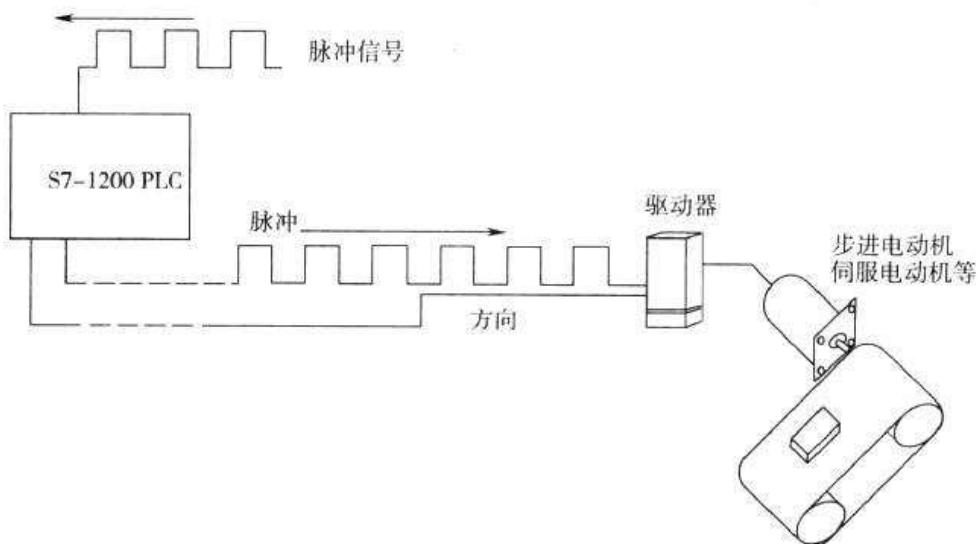


图 7-2 S7-1200 PLC 的运动控制应用

如图 7-3 所示，S7-1200 PLC 实现运动控制的途径主要包括以下 4 个部分：1) 程序指令



图 7-3 S7-1200 PLC 实现运动控制的途径



块；2) 定义工艺对象“轴”；3) CPU PTO 硬件输出；4) 定义相关执行设备，比如机床。

7.1.3 S7-1200 PLC 的 PTO 脉冲输出

S7-1200 PLC 的高速脉冲输出包括脉冲串输出（PTO）和脉冲调制输出（PWM），前者可以输出一串脉冲（占空比为 50%），用户可以控制脉冲的周期和个数（见图 7-4a）；后者可以输出连续的、占空比可以调制的脉冲串，用户可以控制脉冲的周期和脉宽（见图 7-4b）。由于 PWM 已经在前面讲述过，因此不在此赘述。

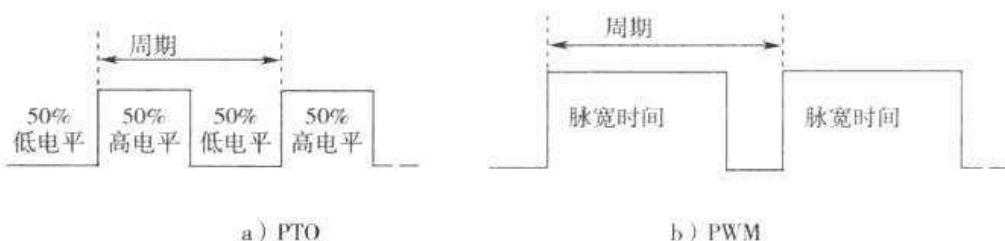


图 7-4 高速脉冲 PTO 和 PWM

S7-1200 PLC 的集成 PTO 输出的最高频率为 100kHz，信号板输出的最高频率为 20kHz。CPU 在使用 PTO 功能时，将占用集成点 Qa.0、Qa.2 或信号板的 Q4.0 作为脉冲输出点，而 Qa.1、Qa.3 或信号板的 Q4.1 作为方向信号输出点。虽然使用了过程映像驱动地址，但这些点会被 PTO 功能独立使用，不会受扫描周期的影响，其作为普通输出点的功能将被禁止。

需要注意的是：目前 S7-1200 PLC 的 CPU 输出类型只支持 PNP 型输出、电压为 DC 24V 的脉冲信号，继电器的点不能用于 PTO 功能，因此在与驱动器连接的过程中尤其要关注。

7.1.4 驱动器 HB-4020M 的特点及其与 PLC 接线

由于 S7-1200 PLC 使用的运动控制还是属于“开环”控制的范围，使用在定位精度一般的情况下，比如机床的进刀、丝杠的定位等，因此在实际使用中采用“PLC + 步进”控制的场合会相比“PLC + 伺服”的场合来得多。本项目将主要介绍 PLC 在步进控制中的应用，其中，步进驱动器采用 HB-4020M 系列、步进电动机采用 57 两相系列。

1. HB-4020M 的特点

HB-4020M 细分型步进电动机驱动器的驱动电压为 DC 12~32V，适配 4、6 或 8 出线、电流在 2.0A 以下、外径为 39-57MM 型号的二相混合式步进电动机，可运用在对细分精度有一定要求的设备上。

图 7-5 所示为 HB-4020M 的外观。

HB-4020M 的电气规格见表 7-1。



图 7-5 HB-4020M 的外观

第7章 S7-1200 PLC 的 PTO 控制

表 7-1 HB-4020M 的电气规格

| 说 明 | 最 小 值 | 推 荐 值 | 最 大 值 |
|---------------|-------|-------|-------|
| 供电电压/V (2A) | 12 | 24 | 32 |
| 输出相电流 (峰值) /A | 0.0 | — | 2.0 |
| 逻辑控制输入电流/mA | 5 | 10 | 30 |
| 步进脉冲响应频率/kHz | 0 | — | 100 |

2. 驱动器与 PLC 的电气接线

HB-4020M 的接线端子功能说明见表 7-2。

表 7-2 HB-4020M 的接线端子功能

| 序号 | 标 示 | 说 明 |
|----|-------|--|
| 1 | GND | 电源 DC 12 ~ 32V |
| 2 | + V | 电源 DC 12 ~ 32V，用户可根据各自的需要选择，一般来说，较高的电压有利于提高电动机的高速力矩，但会加大驱动器和电动机的损耗和发热 |
| 3 | A + | 电动机 A 相，A +、A - 互调，可更改一次电动机运转方向 |
| 4 | A - | 电动机 A 相 |
| 5 | B + | 电动机 B 相，B +、B - 互调，可更改一次电动机运转方向 |
| 6 | B - | 电动机 B 相 |
| 7 | (+5V) | 光电隔离电源，控制信号在 +5V ~ +24V 均可驱动，需注意限流，一般情况下，12V 串接 1kΩ 电阻，24V 串接 2kΩ 电阻，驱动器内部电阻为 330Ω |
| 8 | PUL | 脉冲信号：上升沿有效 |
| 9 | DIR | 方向信号：低电平有效 |
| 10 | ENA | 使能信号：低电平有效 |

HB-4020M 与 PLC 的接线如图 7-6 所示，由于 PLC 的脉冲信号为 PNP 和 24V 两种，因此需要考虑串接 2kΩ 电阻。

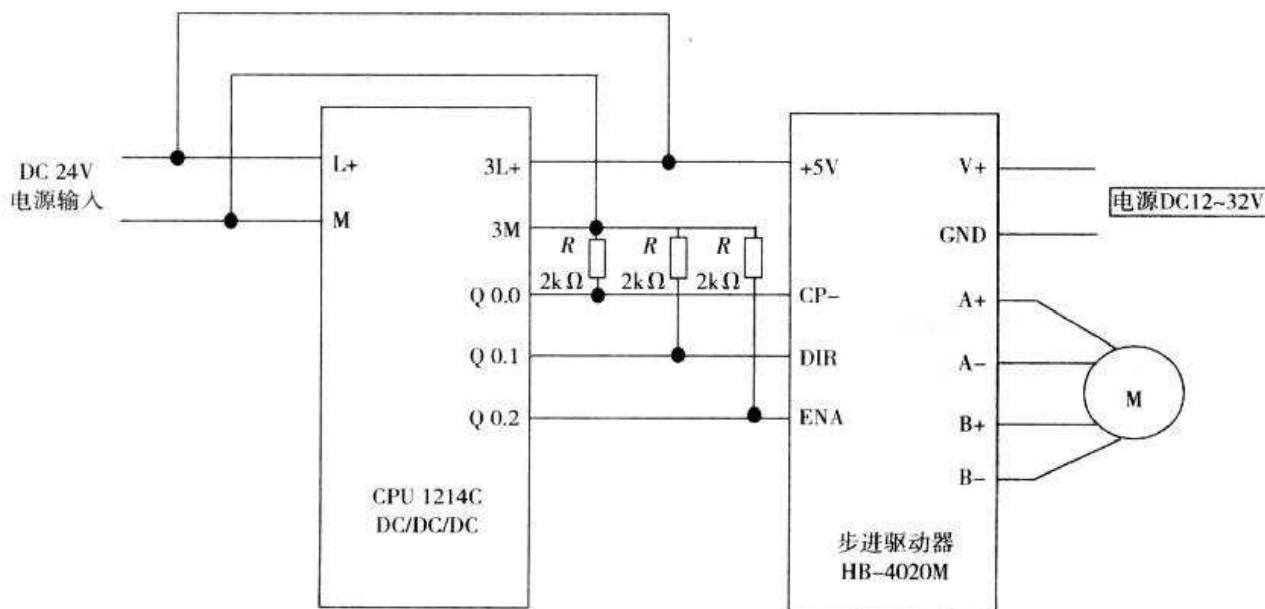


图 7-6 PLC 与 HB-4020M 的接线

图解西门子 S7-1200 PLC 入门到实践

3. 驱动器供电电压

供电电压越高，电动机高速时的力矩越大，但另一方面，电压太高会导致过电压保护，甚至可能损坏驱动器，而且在高压下工作时，低速运动振动较大。所以电压设定在一般情况下，电动机转速小于 150r/min 时，尽量使用低电压（小于等于 24V）。转速越高，可相应提高电压，但不要超过驱动器的最大电压（DC 32V）。

4. 驱动器上电动机电流的设置

如图 7-7 所示为在步进驱动器上进行步进电动机的电流设置。电流设定值越大，电动机输出力矩越大，但电流大时，电动机和驱动器的发热也比较严重。所以，一般情况是把电流设成电动机的额定电流，在保证力矩足够的情况下尽量减小电流，这样长时间工作可以提供驱动器和电动机工作的稳定性。高速状态工作时可以提高电流值（不要超过 30%）。

具体设置规则如下：

- 1) 四线电动机和六线电动机高速度模式：输出电流设成等于或小于电动机额定电流值；
- 2) 六线电动机高力矩模式：输出电流设成电动机额定电流的 70%；
- 3) 八线电动机串联接法：由于串联时电阻增大，输出电流应设成电动机额定电流的 70%；
- 4) 八线电动机并联接法：输出电流可设成电动机额定电流的 1.4 倍。注意：电流设定后，电动机温升太高，则应降低电流设定值。

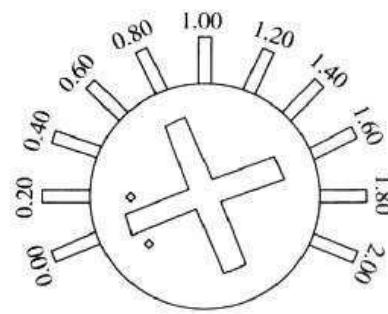


图 7-7 步进电动机的电流设置

7.1.5 步进电动机的基本工作原理及选型

1. 基本工作原理

步进电动机是利用电磁铁原理，将脉冲信号转换成线位移或角位移的电动机。每来一个电脉冲，电动机转动一个角度，带动机械移动一小段距离。

步进电动机的特点主要包括：1) 来一个脉冲，转一个步距角；2) 控制脉冲频率，可控制电动机转速；3) 改变脉冲顺序，改变转动方向；4) 角位移量或线位移量与电脉冲数成正比。

通常按励磁方式可以将步进电动机分为三类：

1) 反应式：转子无绕组，定子开小齿、步距小。这种方式的应用最广。

2) 永磁式：转子的极数 = 每相定子极数，不开小齿，步距角较大，力矩较大。

3) 感应子式（混合式）：开小齿，混合反应式与永磁式的优点是转矩大、动态性能好、步距角小。

如图 7-8 所示，步进电动机主要由两部分构

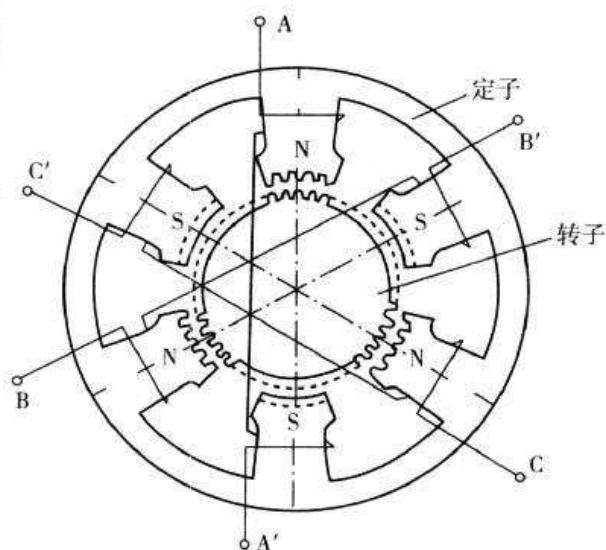


图 7-8 步进电动机的结构



成，即定子和转子。它们均由磁性材料构成。定、转子铁心由软磁材料或硅钢片叠成凸极结构，定、转子磁极上均有小齿，定、转子的齿数相等。其中，定子有 6 个磁极，定子磁极上套有星形联结的三相控制绕组，每两个相对的磁极为一相，组成一相控制绕组，转子上没有绕组。转子上相邻两齿间的夹角 $\theta_t = \frac{360^\circ}{Z_t}$ ，称为齿距角。

2. 步进电动机选型

虽然步进电动机已被广泛地应用，但步进电动机并不能像普通的直流电动机、交流电动机一样在常规下使用。它必须由双环形脉冲信号、功率驱动电路等组成控制系统方可使用。因此，用好步进电动机却非易事，它涉及到机械、电动机、电子及计算机等许多专业知识。

步进电动机一经定型，其性能取决于电动机的驱动电源。步进电动机的转速越高，力矩越大，则要求电动机的电流越大，驱动电源的电压越高。电压对力矩的影响如图 7-9 所示。

在步进电动机的步距角不能满足使用的条件下，可采用细分驱动器来驱动步进电动机，细分驱动器的原理是通过改变相邻 (A, B) 电流的大小来改变合成磁场的夹角，从而控制步进电动机运转的。细分驱动器的原理如图 7-10 所示。

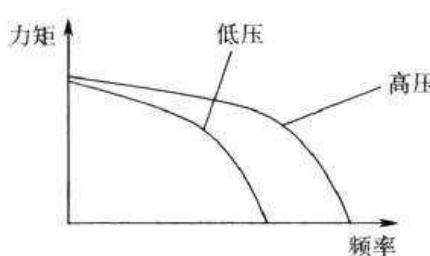


图 7-9 电压对力矩影响

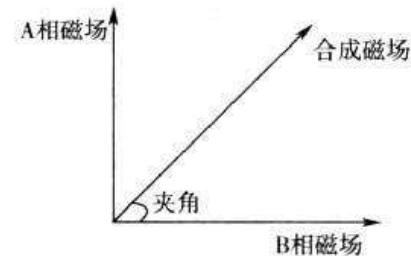
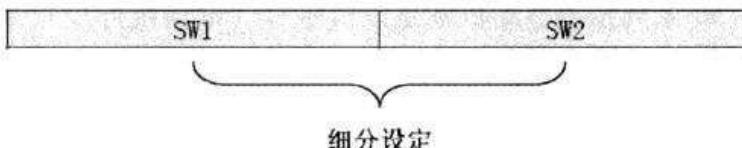


图 7-10 细分驱动器的原理

驱动器 HB-4020M 可以对拨码开关 DIP-SW 进行细分设定，具体如图 7-11 所示。



| 细分倍数 | SW1 | SW2 |
|------|-----|-----|
| 1 | on | on |
| 2 | off | on |
| 4 | on | off |
| 8 | off | off |

图 7-11 拨码开关的细分设定

一般而言，步进电动机的型号由步距角（涉及到相数）、静力矩及电流三大要素组成，一旦三大要素确定，步进电动机的型号便确定了下来。

(1) 步距角的选择

电动机的步距角取决于负载精度的要求，将负载的最小分辨率（当量）换算到电动机轴上，每个当量电动机应走多少角度（包括减速）。电动机的步距角应等于或小于此角度。



图解西门子 S7-1200 PLC 入门到实践

目前，市场上步进电动机的步距角一般有 $0.36^\circ/0.72^\circ$ （五相电动机）、 $0.9^\circ/1.8^\circ$ （二、四相电动机）、 $1.5^\circ/3^\circ$ （三相电动机）等。

（2）静力矩的选择

步进电动机的动态力矩很难确定，我们往往先确定电动机的静力矩。静力矩选择的依据是电动机工作的负载，而负载可分为惯性负载和摩擦负载两种。单一的惯性负载和单一的摩擦负载是不存在的。直接起动时（一般由低速），两种负载均要考虑，加速起动时，主要考虑惯性负载，恒速运行只要考虑摩擦负载。一般情况下，静力矩应为摩擦负载的 $2 \sim 3$ 倍，静力矩一旦选定，电动机的机座及长度便能确定下来（几何尺寸）。

（3）电流的选择

静力矩一样的电动机，由于电流参数不同，其运行特性差别很大，可依据矩频特性曲线，判断电动机的电流（参考驱动电源及驱动电压）。

（4）力矩与功率的换算

步进电动机一般在较大范围内调速使用，其功率是变化的，一般只用力矩来衡量，力矩与功率的换算如下：

$$P = \Omega \cdot M$$

$$\Omega = \frac{2\pi \cdot n}{60}$$

$$P = \frac{2\pi n M}{60}$$

其中， P 为功率，单位为 W； Ω 为每秒角速度，单位为 rad/s； n 为每分钟转速； M 为力矩，单位为 N·m

$$P = \frac{2\pi f M}{400} \text{ (半步工作)}$$

其中， f 为每秒脉冲数（简称 PPS）。

至于本项目所选用的 57 系列步进电动机则是以机座号（电动机外径）来划分的。根据机座号可分为：42BYG（BYG 为感应子式步进电动机代号）、57BYG、86BYG、110BYG 等国际标准，而像 70BYG、90BYG、130BYG 等均为国内标准。

图 7-12 所示为 57 步进电动机外观及其接线端子。

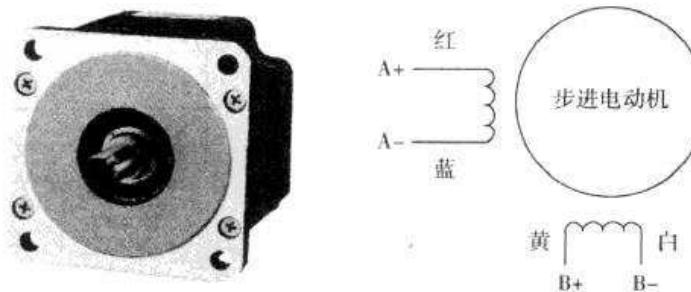


图 7-12 57 步进电动机外观及其接线端子

7.1.6 工艺对象“轴”的概念

在 S7-1200 PLC 中，术语“轴”特指用“轴”工艺对象表示的驱动器工艺映像。“轴”

第7章 S7-1200 PLC 的 PTO 控制



工艺对象是用户程序与驱动器之间的接口。该工艺对象接收用户程序中的运动控制命令，执行这些命令并监视其运行情况。运动控制命令在用户程序中通过运动控制语句启动。

术语“驱动器”特指由步进电动机与动力部分或伺服驱动器与具有脉冲接口的转换器组成的机电装置。驱动器由“轴”工艺对象通过 S7-1200 CPU 的脉冲发生器控制。

7.2 技能训练：工艺对象“轴”的组态与调试

7.2.1 S7-1200 PTO 的硬件组态

S7-1200 PLC 对于运动控制需要先进行硬件配置，具体步骤包括：1) 选择设备组态；2) 选择合适的 PLC；3) 定义脉冲发生器为 PTO。

如图 7-13 所示为选择 PTO1 或 PTO2。

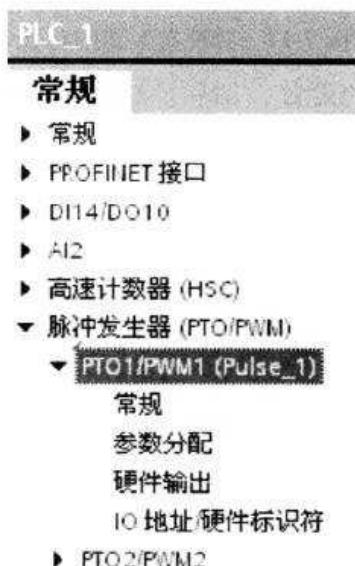


图 7-13 选择 PTO

如图 7-14 所示为使能 PTO 功能，即允许使用该脉冲发生器。

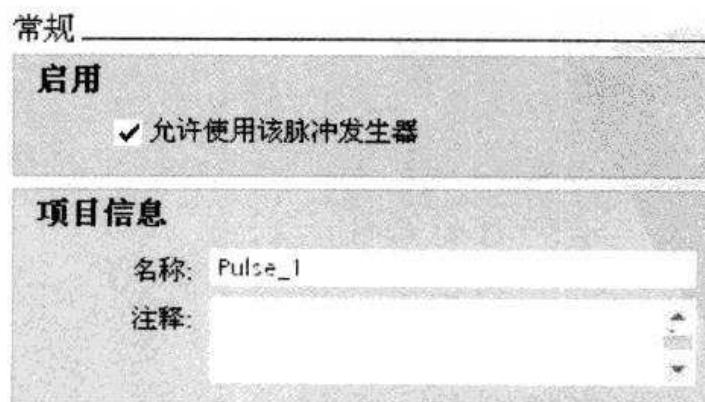


图 7-14 使能 PTO 功能



图解西门子 S7-1200 PLC 入门到实践

如图 7-15 所示为脉冲发生器的参数分配，包括设置 PTO 功能或 PWM 功能，还有输出源为集成输出还是板载 CPU 输出（如果使用具有继电器输出的 PLC，则必须将信号板用于 PTO 的输出）。

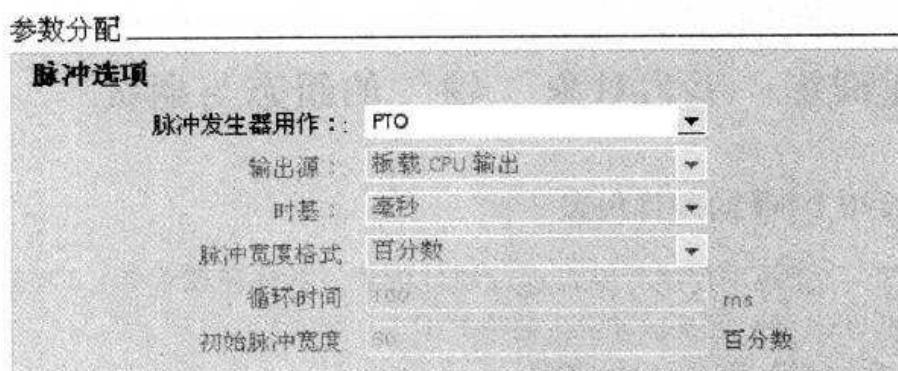


图 7-15 脉冲发生器的参数分配

如图 7-16 所示为硬件输出的设置，即脉冲输出为 Q0.0、方向输出为 Q0.1、分配的计数器为 HSC_1（内置）。

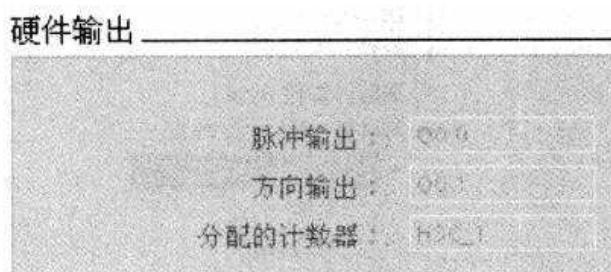


图 7-16 硬件输出的设置

如图 7-17 所示为硬件标识符，即 ID = 7。

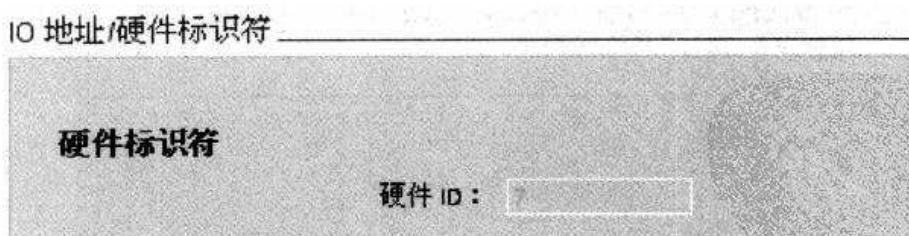


图 7-17 硬件标识符

7.2.2 组态工艺“轴”

图 7-18 所示为运动控制指令生成脉冲的顺序。一般而言，用于控制步进电动机的脉冲通过 S7-1200 PLC 输出，并生成送到步进电动机的驱动器后，这些脉冲将由步进电动机的驱动器转化为轴向运动，以最终进行定位、定长等位置动作。

第7章 S7-1200 PLC的PTO控制

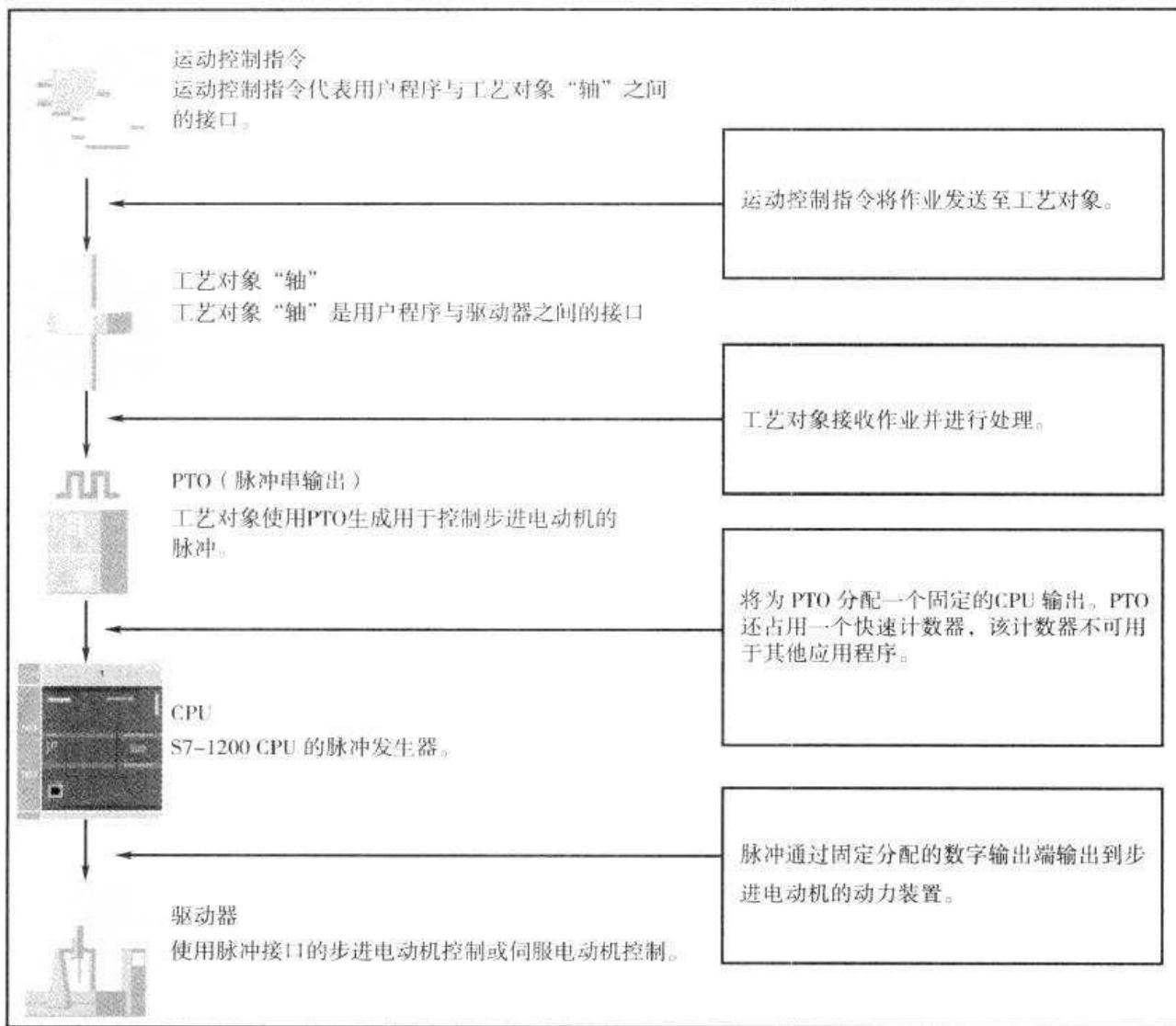


图 7-18 运动控制指令生成脉冲的顺序

如图 7-19 所示为在项目树中创建“轴”对象。

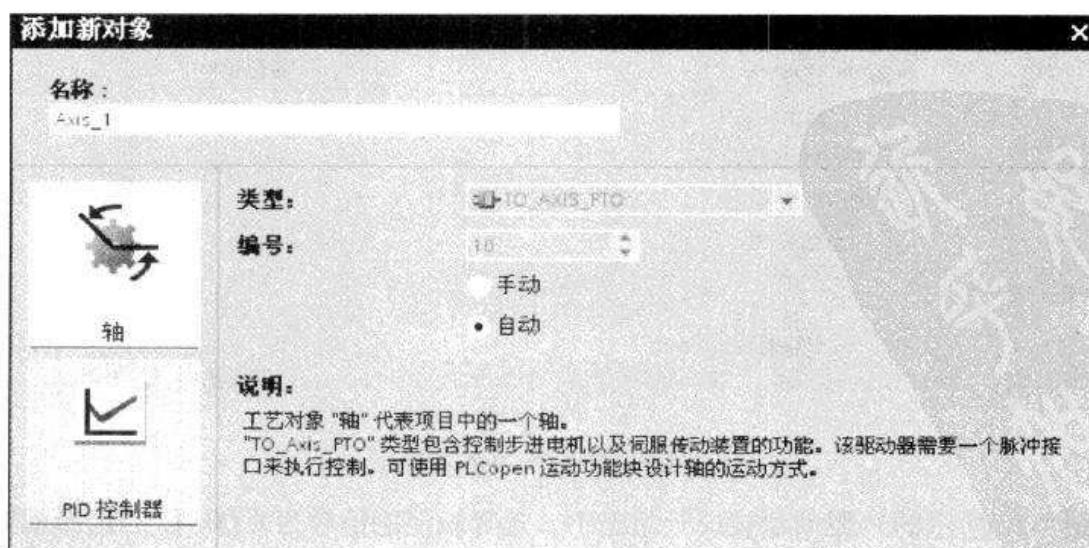


图 7-19 创建“轴”对象



图解西门子 S7-1200 PLC 入门到实践

在创建了“轴”对象后，即可在项目树的“工艺对象中”找到“Axis_1”，并选择“组态”菜单（见图 7-20）。

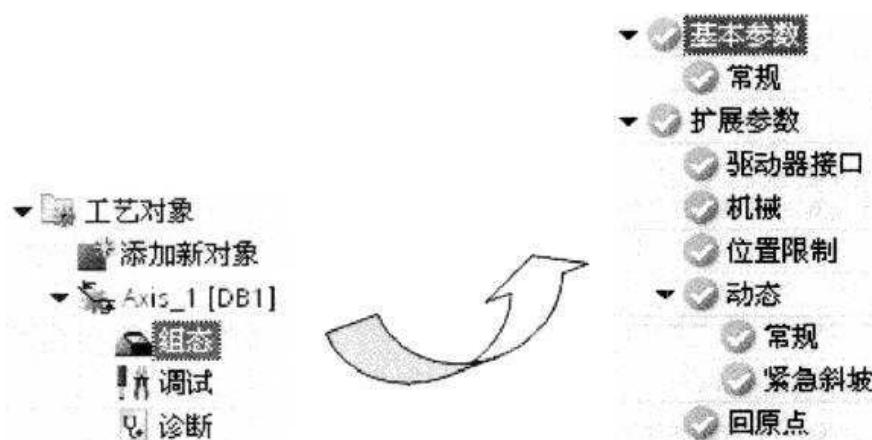


图 7-20 工艺“轴”组态菜单

在图 7-21 所示的常规选项中，对硬件接口的“为轴控制选择 PTO”选择“Pulse_1”，这时可以切换到“设备配置”进行硬件组态。如果这个步骤此前已经完成，也可以不做。

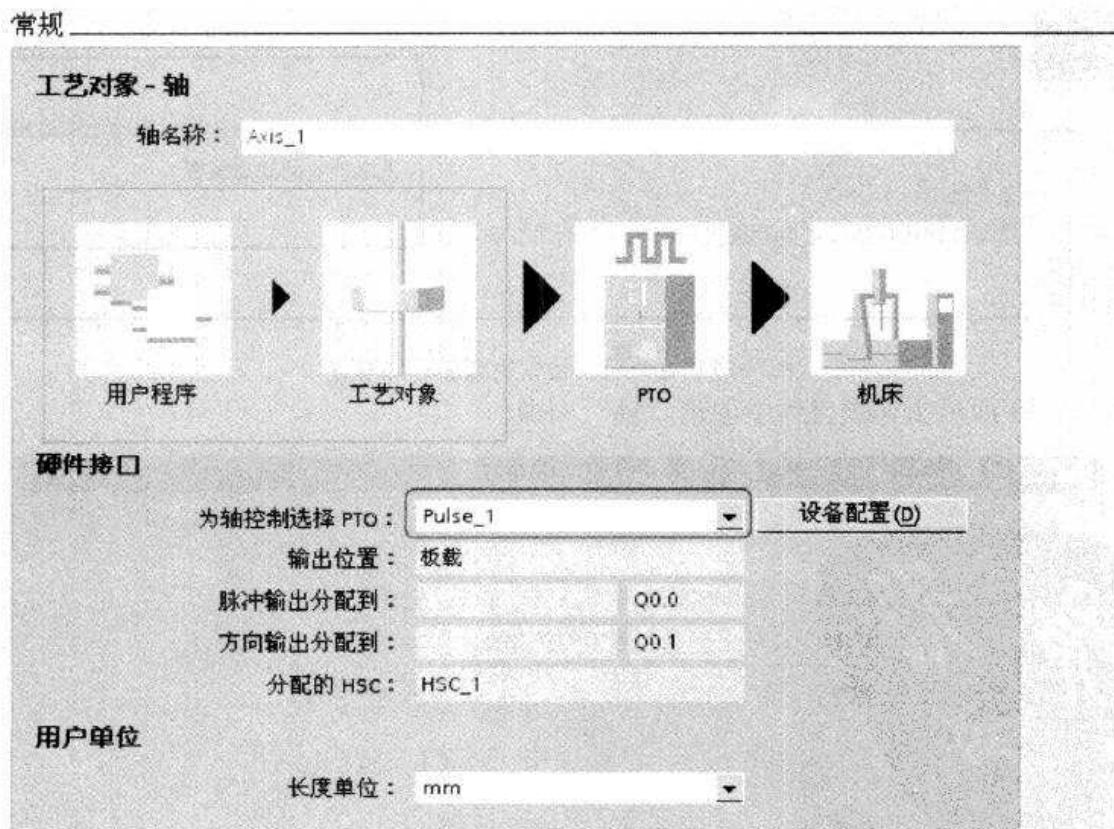


图 7-21 “轴”对象的组态

在图 7-22 所示的“驱动器接口”组态中，选择轴使能信号为% Q0.4。“选择就绪输入”是设置驱动系统正常输入点，当驱动设备正常时，会给出一个开关量输出，此信号可以接到

第7章 S7-1200 PLC 的 PTO 控制



CPU 中，告知运动控制器驱动器正常与否。本案例中，驱动器 HB-4020M 不提供这种接口，可将此参数设为“TRUE”。

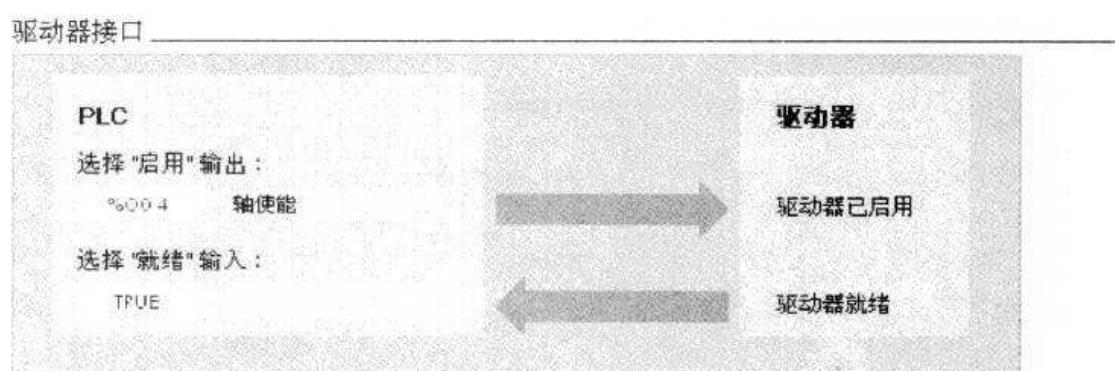


图 7-22 驱动器接口组态

机械组态如图 7-23 所示，选项“电机每转的增量”为电动机旋转一周所产生的脉冲个数；选项“电机每转的运载距离”为电动机旋转一周后生产机械所产生的位移，这里的单位与图 7-21 所选择的单位一致。

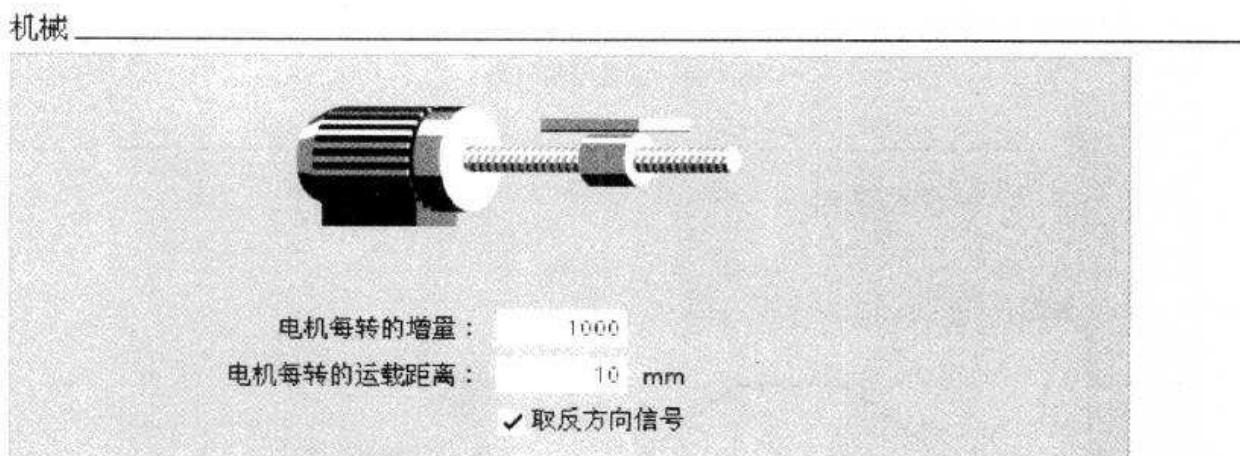


图 7-23 机械组态

图 7-24 所示为位置限制组态，在 S7-1200 PLC 运动控制中，它可以设置两种限位，即软件限位和硬件限位，如两者都启用，则必须输入硬件下限开关输入、硬件上限开关输入、激活方式（高电平）、软件下限和软件上限。在达到硬件限位时，“轴”将使用急停减速斜坡停车；在达到软件限位时，激活的“运动”将停止，工艺对象报故障，在故障被确认后，“轴”可以恢复在原工作范围内运动。

图 7-25 所示为动态常规参数组态，它包括速度限值的单位、最大速度、启动和停止速度、加速度、加速与减速时间。加/减速度与加/减速时间这两组数据，只要定义其中任意一组，系统就会自动计算另外一组数据，这里的加/减速度和加/减速时间需要用户根据实际工艺要求和系统本身特性调试得出。运动控制功能所支持的最高频率根据所使用的硬件点决定，目前最低频率为 2Hz，最小加/减速度为 0.28Hz/s，最大加/减速度为 9500MHz/s。



图解西门子 S7-1200 PLC 入门到实践

位置限制

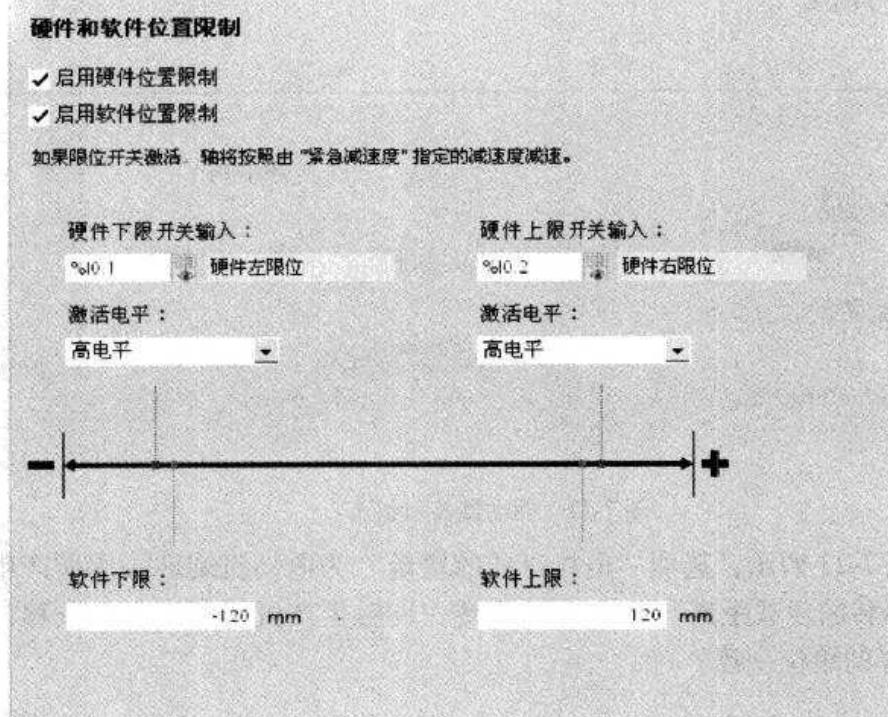


图 7-24 位置限制组态

常规

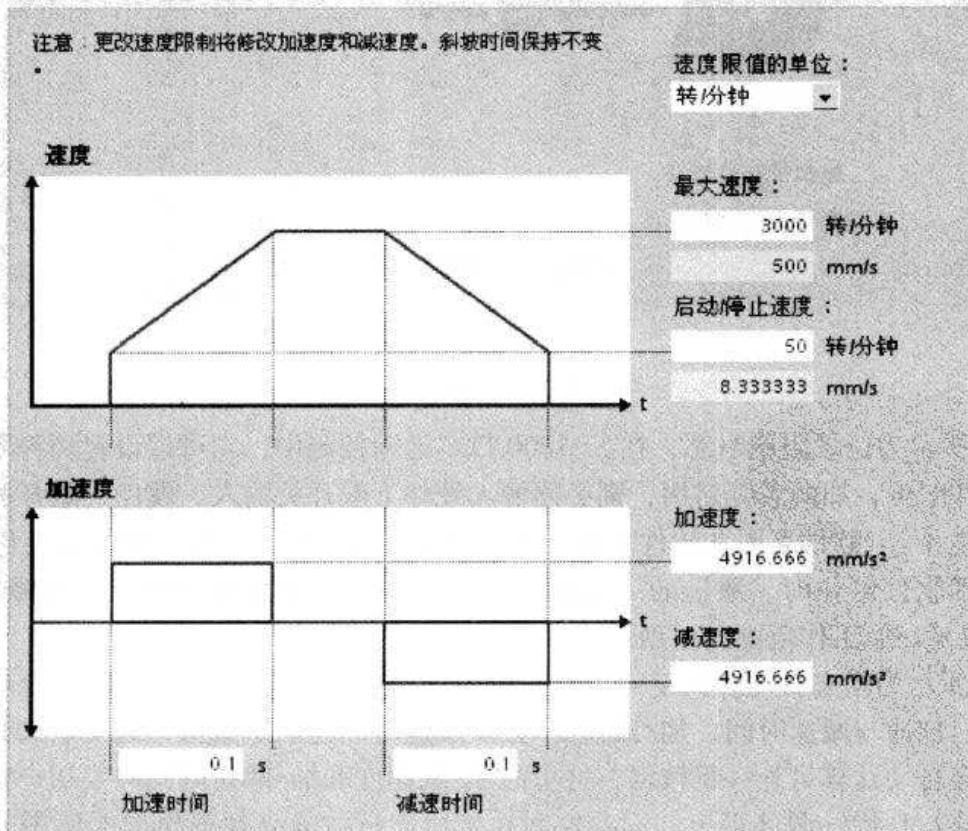


图 7-25 动态常规参数组态

第7章 S7-1200 PLC 的 PTO 控制



图 7-26 所示为“动态紧急斜坡”组态，它需要定义一组从最大速度急停减速到启动/停止速度的减速度。

在图 7-27 所示的回原点组态中，需要输入参考点开关，一般使用数字量输入作为参考

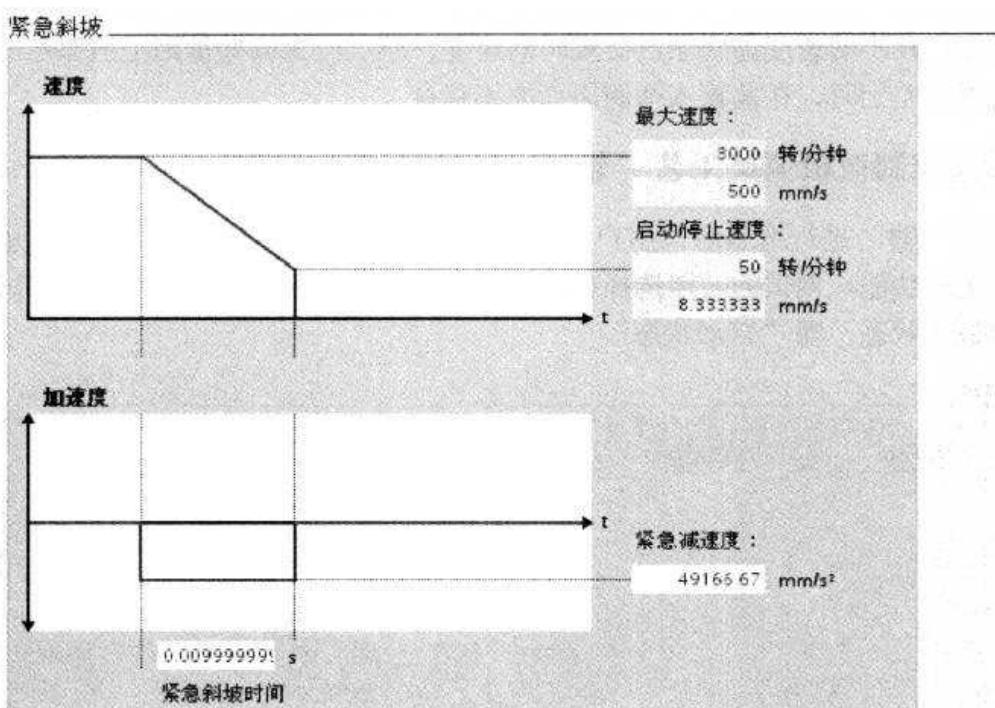


图 7-26 动态紧急斜坡组态

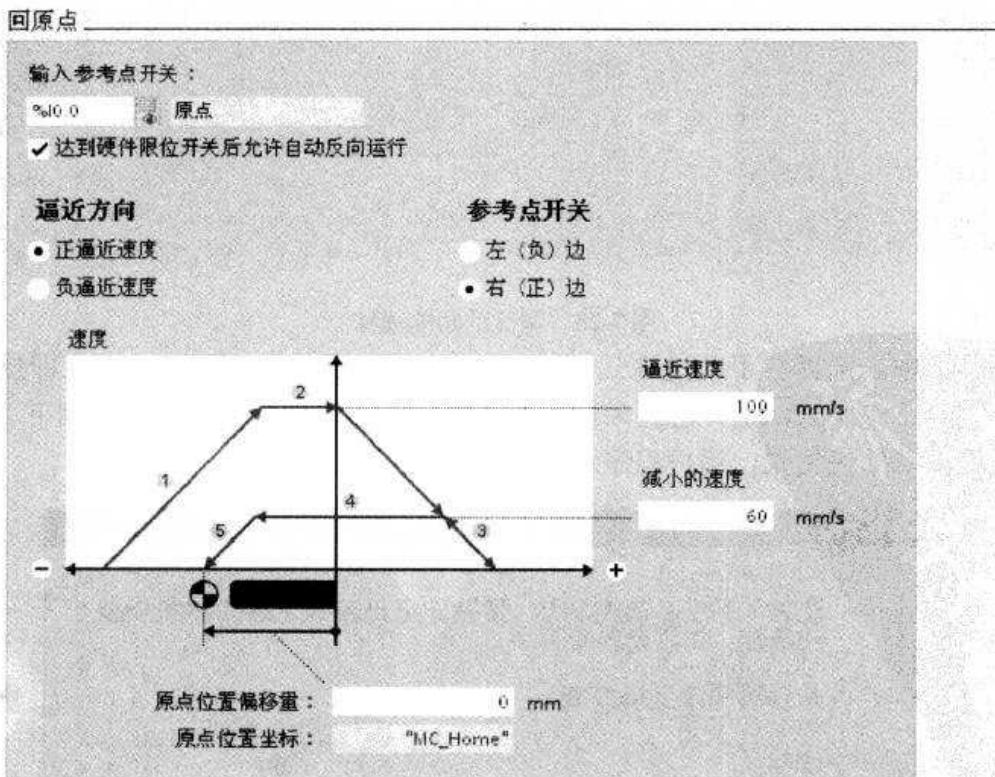


图 7-27 回原点组态



图解西门子 S7-1200 PLC 入门到实践

点开关。“达到硬件限位开关允许自动反向运行”选项使能后，在轴碰到原点之前碰到了硬件限位点，此时系统认为原点在反方向，会按组态好的斜坡减速曲线停车并反转，若该功能没有被激活并碰到硬件限位，则回原点过程会因为错误被取消，并紧急停止。逼近方向定义了在执行原点过程中的初始方向，包括正逼近速度和负逼近速度两种。逼近速度为进入原点区域时的速度；减小的速度为到原点位置时的速度；原点位置偏移量则是当原点开关位置和原点实际位置有差别时，在此输入举例原点的偏移量。

7.2.3 通过控制面板调试工艺“轴”

在对工艺“轴”进行组态后，用户可以使用控制面板调试步进电动机及驱动器，以进行轴的实际运行功能。调试的功能选择如图 7-28 所示，图 7-28 中显示了选择调试功能后的控制面板的最初状态，即“自动状态”。

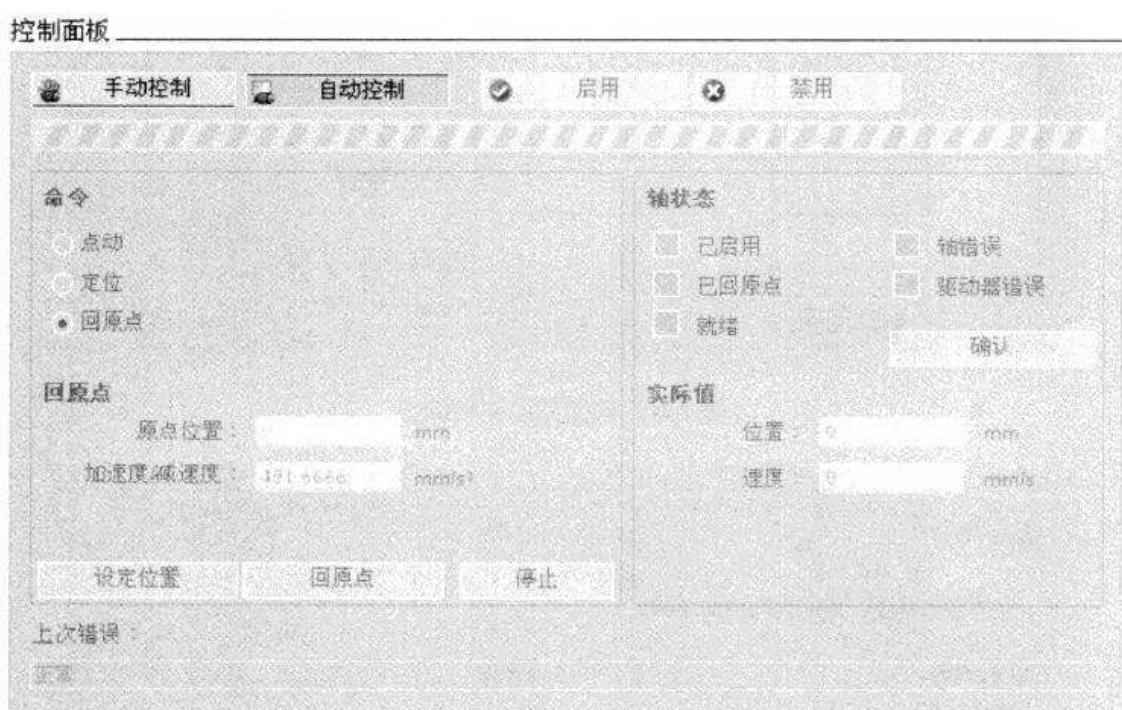


图 7-28 调试的功能选择

在控制面板中，选择“手动控制”，此时会跳出提示窗口（见图 7-29），即提醒用户在采用手动控制前，先要确认是否已经采取了适当的安全预防措施。在手动控制下，控制面板具有控制轴和驱动的优先权，用户程序对轴不起作用。

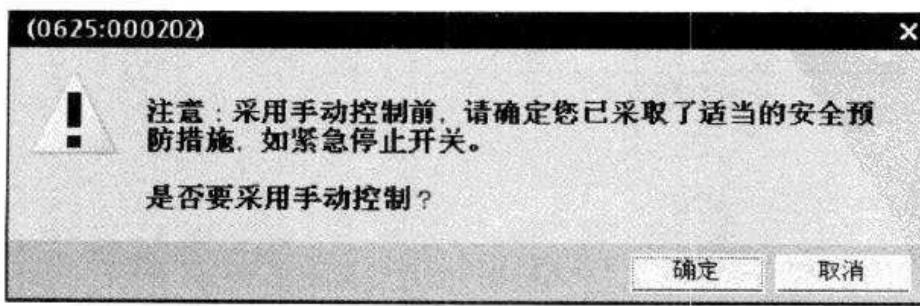


图 7-29 手动控制的提示窗口

第7章 S7-1200 PLC 的 PTO 控制



选择“手动控制”并选择“启用”，这时可以观察到如图 7-30 所示的轴状态“已启用”和“就绪”，首先可进行“回原点”命令。

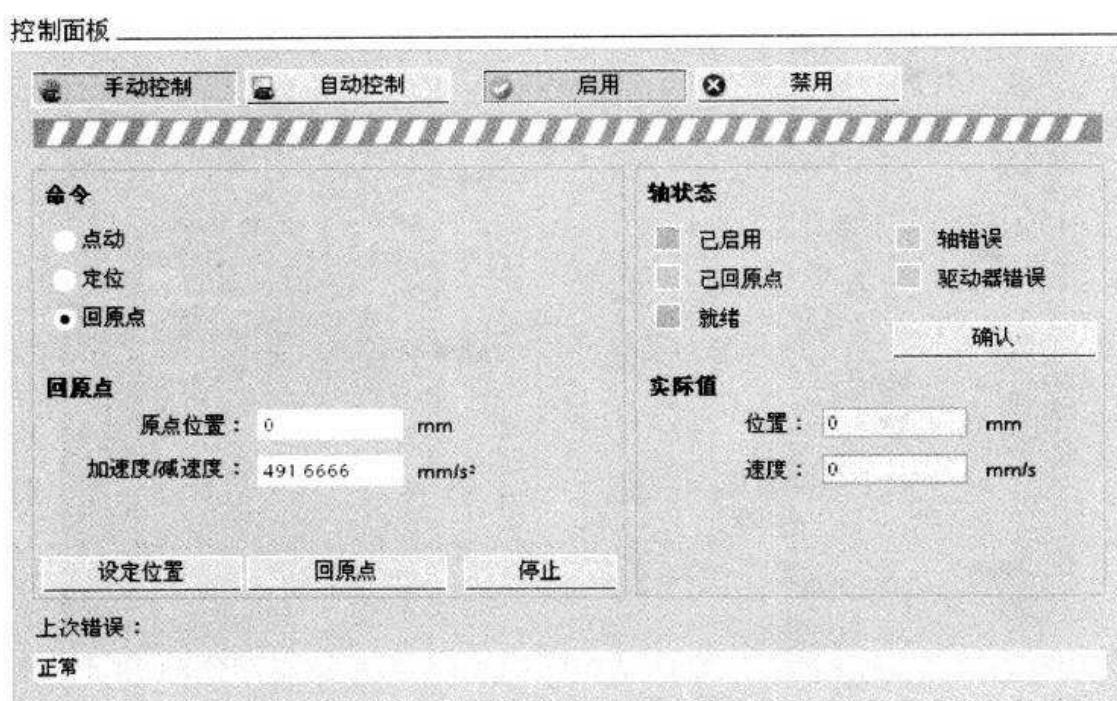


图 7-30 轴状态和回原点命令

图 7-31 所示为点动命令，轴将按照组态好的加速度/减速度、速度进行向前点动或向后点动。

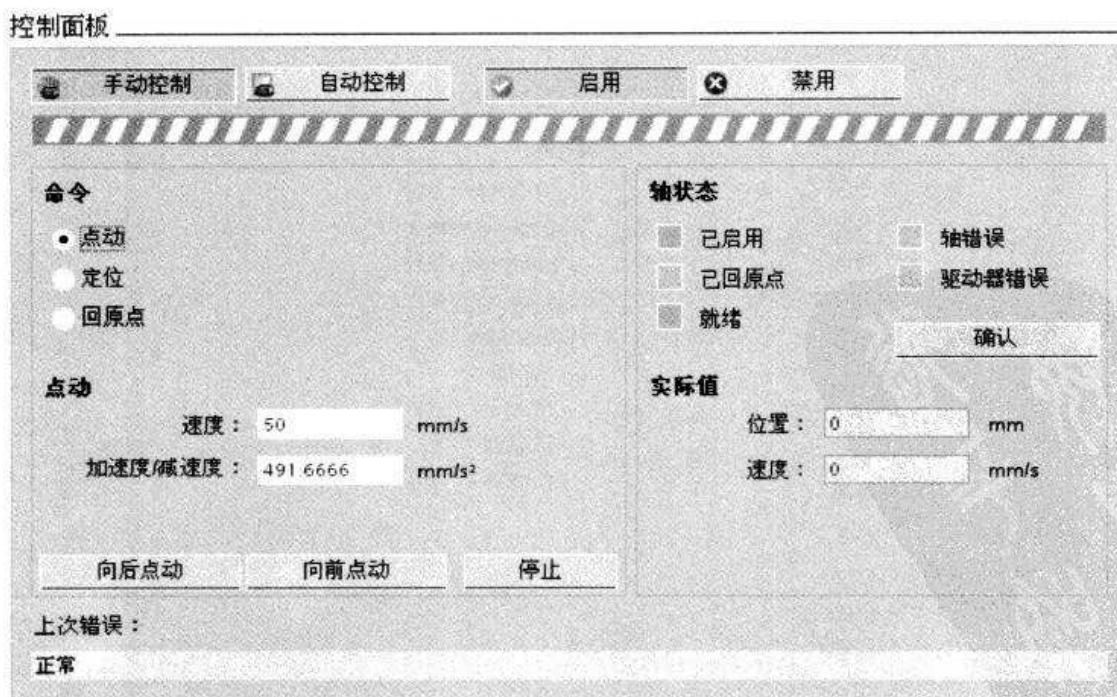


图 7-31 点动命令



在点动过程中，如果出现软件限位动作，则会报故障（见图 7-32），这时候轴被停止，并只有在复位故障后才能进行下一步调试。

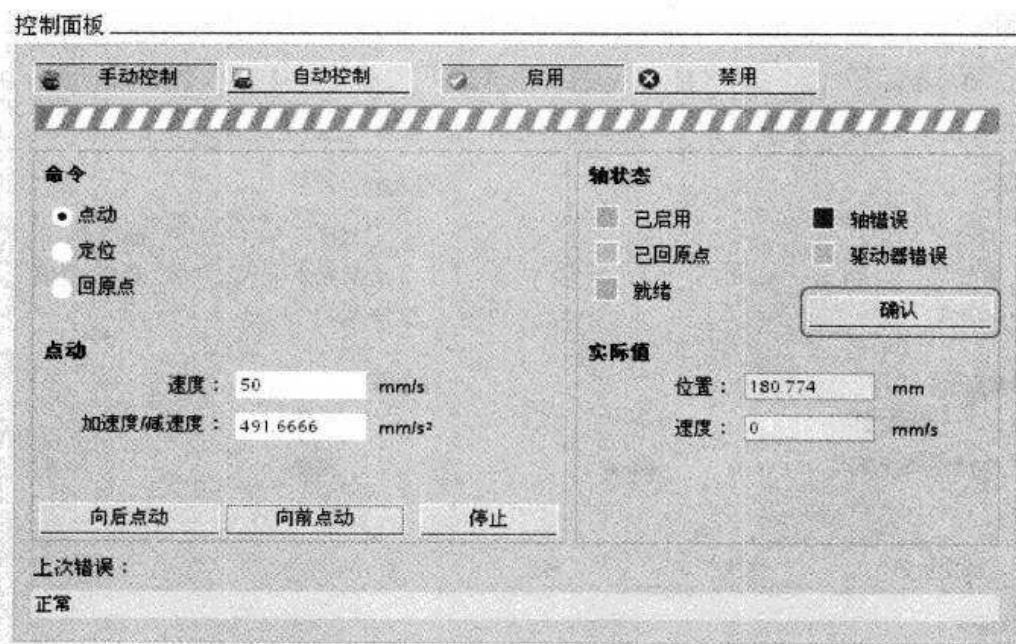


图 7-32 点动命令轴出错

7.2.4 诊断工艺“轴”

在对“轴”组态和调试后，还可以点击项目树下工艺对象 Axis_1 的 图符对运动控制轴进行诊断，如图 7-33 所示为状态位和错误位诊断，如图 7-34 所示为运动状态诊断。

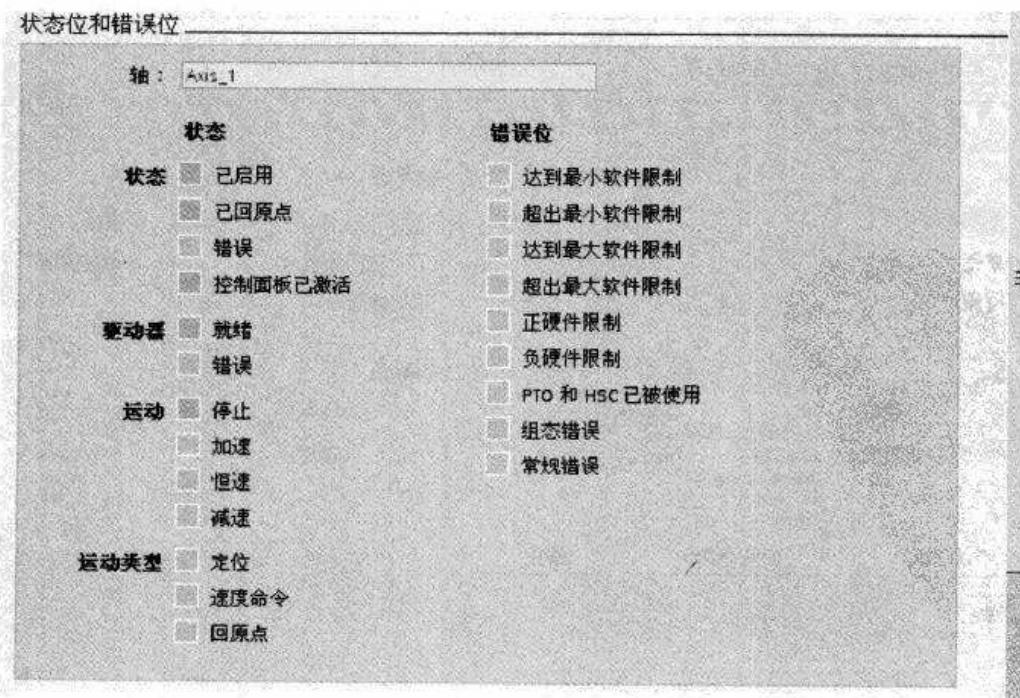


图 7-33 状态位和错误位诊断



运动状态

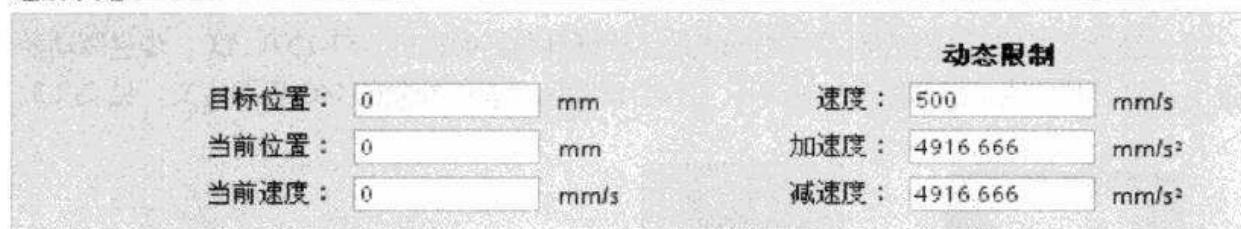


图 7-34 运动状态诊断

7.3 工程案例：通过触摸屏控制工作台滑动座电动机

7.3.1 案例介绍

现有一工程，需要通过触摸屏对工作台的滑动座步进电动机进行控制（见图 7-35），具体要求如下：

- 1) 滑动座③由步进电动机②带动丝杠①在轨道上左右滑行；
- 2) 磁性限位开关⑧分别代表左极限、外部参考点、右极限，直接输入到 S7-1200 PLC 的输入点；
- 3) 该滑动座的最大行程为 240mm，如超出，则电动机停止，并同时在触摸屏上报警显示；
- 4) 该滑动座的移动方式为两种，一种为 HOME（即回参考原点）；一种是位置移动（分绝对位置和相对位置两种）。

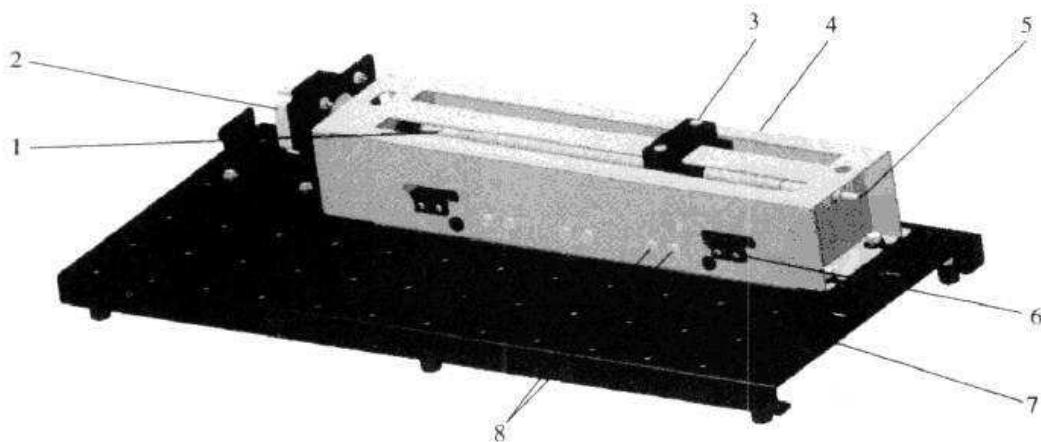


图 7-35 工作台滑动座电动机

1—丝杠 2—步进电动机 3—滑动座 4—机盖 5—杆端
6—左右机械限位 7—工作台底座 8—磁性限位开关（分别是左极限、外部参考点、右极限）



7.3.2 S7-1200 PLC 的硬件设计

通过触摸屏来控制工作台滑动座电动机的主要硬件包括触摸屏、S7-1200 PLC、步进驱动器和步进电动机，其接线示意如图 7-36 所示。工作台滑动座电动机控制的 I/O 资源定义。见表 7-3。

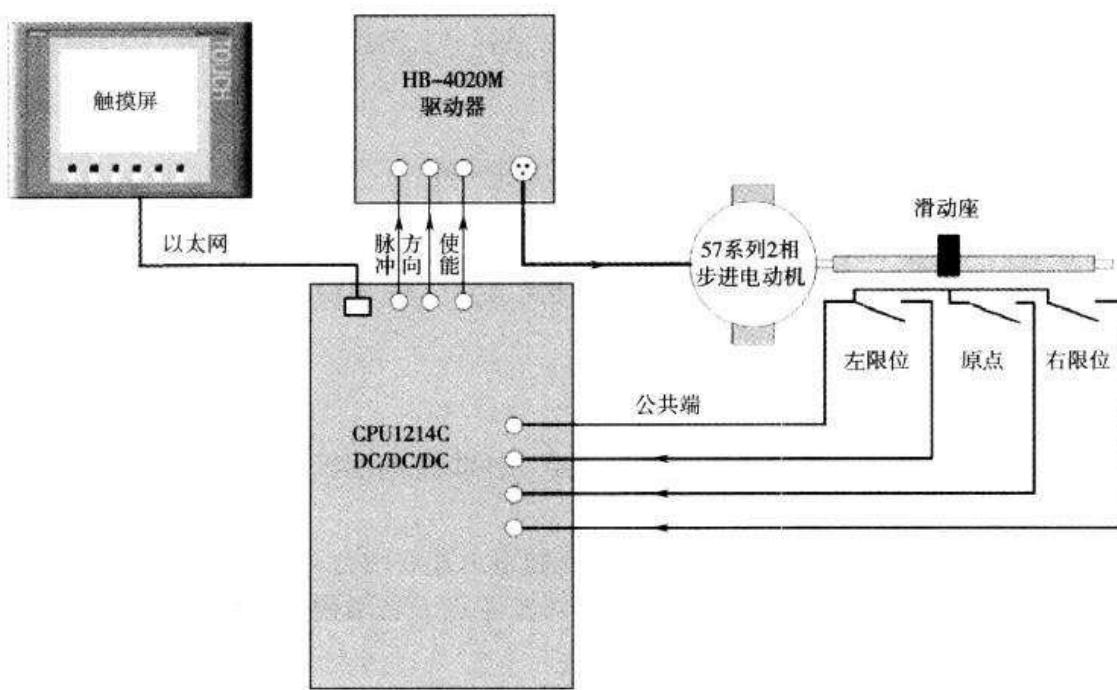


图 7-36 工作台滑动座电动机控制示意

表 7-3 工作台滑动座电动机控制的 I/O 资源定义

| 序号 | 名称 | 地址 |
|----|----------|-------|
| 1 | 原点 | %I0.0 |
| 2 | 硬件左限位 | %I0.1 |
| 3 | 硬件右限位 | %I0.2 |
| 4 | 紧急停止按钮 | %I0.3 |
| 5 | 输出脉冲 PTO | %Q0.0 |
| 6 | 输出脉冲方向 | %Q0.1 |
| 7 | 轴使能 | %Q0.4 |
| 8 | 紧急停止输出 | %Q0.5 |

7.3.3 硬件组态与软件编程

1. 硬件组态

本案例是使用 Q0.0，因此可以激活脉冲发生器 PTO1，系统默认指定 Q0.0 为脉冲输出、Q0.1 为方向输出，HSC1 为该 PTO 功能的高速计数器。

2. 指令介绍

通过指令库的扩展指令，可以获得如图 7-37 所示的一系列运动控制指令，具体为：MC_Power 启用/禁用轴；MC_Reset 确认错误；MC_Home 使轴回原点，设置参考点；MC_

第7章 S7-1200 PLC 的 PTO 控制



Halt 停止轴；MC_MoveAbsolute 绝对定位轴；MC_MoveRelative 相对定位轴；MC_MoveVelocity 以速度预设值移动轴；MC_MoveJog 在点动模式下移动轴。

(1) MC_Power 指令

轴在运动之前必须先被使能，使用运动控制指令“MC_Power”可集中启用或禁用轴。如果启用了轴，则分配给此轴的所有运动控制指令都将被启用。如果禁用了轴，则用于此轴的所有运动控制指令都将无效，并将中断当前的所有作业。图 7-38 所示为 MC_Power 指令，当 Enable 端变为高电平后，CPU 就按照工艺对象中组态好的方式使能外部驱动器；当 Enable 端变为低电平后，CPU 就按照 StopMode 中定义的模式进行停车。



图 7-37 一系列的运动控制指令

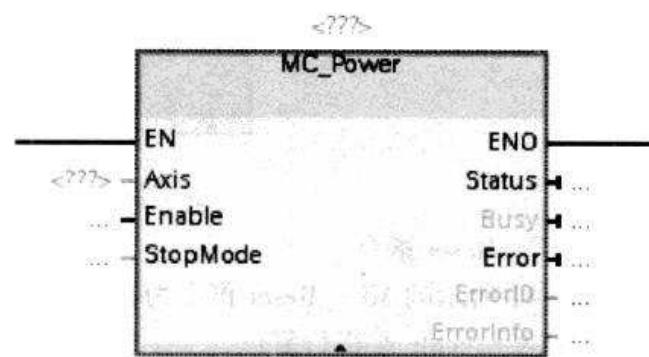


图 7-38 MC_Power 指令

(2) MC_Home 指令

轴回原点由运动控制语句“MC_Home”启动（见图 7-39）。回原点期间，参考点坐标设置在定义的轴机械位置处。回原点模式共有 4 种模式：

1) Mode = 3 主动回原点。在主动回原点模式下，运动控制语句“MC_Home”执行所需要的参考点逼近。将取消其他所有激活的运动。

2) Mode = 2 被动回原点。在被动回原点模式下，运动控制语句“MC_Home”不执行参考点逼近。不取消其他激活的运动。逼近参考点开关必须由用户通过运动控制语句或由机械运动执行。

3) Mode = 0 绝对式直接回原点。无论参考凸轮位置为何，都设置轴位置。不取消其他激活的运动。立即激活“MC_Home”语句中的“Position”参数的值作为轴的参考点和位置值，轴必须处于停止状态才能将参考点准确分配到机械位置。

4) Mode = 1 相对式直接回原点。无论参考凸轮位置为何，都设置轴位置。不取消其他激活的运动。适用于参考点和轴位置的规则：新的轴位置 = 当前轴位置 + “Position”参数的值。

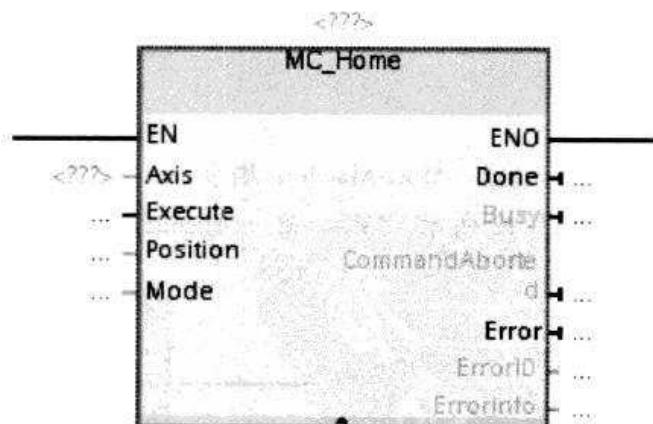


图 7-39 MC_Home 指令



图解西门子 S7-1200 PLC 入门到实践

(3) MC_Halt 指令

如图 7-40 所示的 MC_Halt 指令为停止轴的运动，每个被激活的运动指令都可由此块停止，上升沿使能 Execute 后，轴会立即按照组态好的的减速曲线停车。

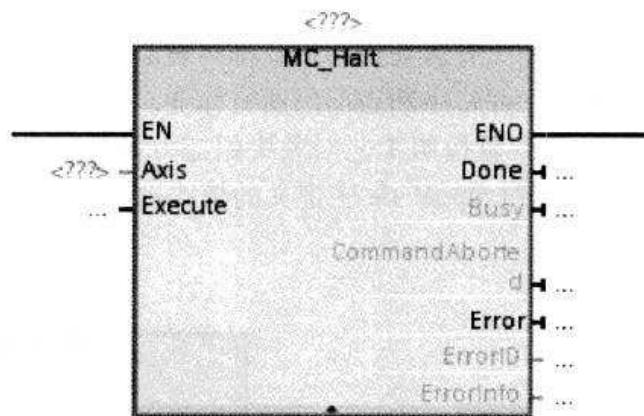


图 7-40 MC_Halt 指令

(4) MC_Reset 指令

如图 7-41 所示的 MC_Reset 指令为错误确认，即如果存在一个需要确认的错误，可通过上升沿激活 Execute 端进行复位。

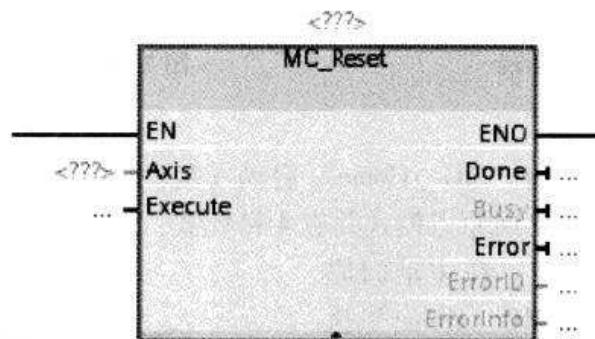


图 7-41 MC_Reset 指令

(5) MC_MoveAbsolute 指令

图 7-42 所示的 MC_MoveAbsolute 指令为绝对位置移动，它需要在定义好参考点、建立

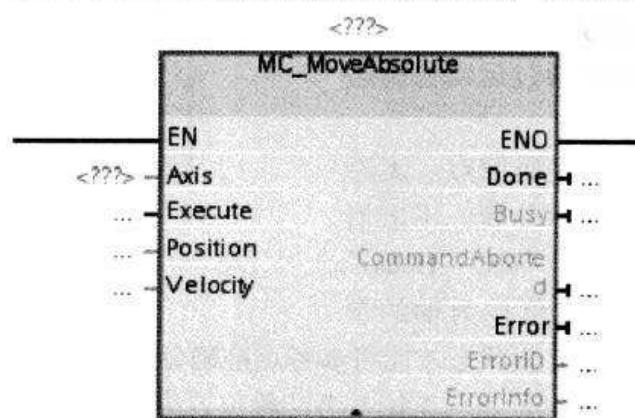


图 7-42 MC_MoveAbsolute 指令



起坐标系后才能使用。通过指定参数 Position 和 Velocity 可到达机械限位内的任意一点，当上升沿使能 Execute 选项后，系统会自动计算当前位置与目标位置之间的脉冲数，并加速到指定速度，在到达目标位置时减速到启动/停止速度。

(6) MC_MoveRelative 指令

图 7-43 所示的 MC_MoveRelative 语句表示相对位置移动，它的执行不需要建立参考点，只需定义运行距离、方向及速度。当上升沿使能 Execute 端后，轴按照设置好的距离与速度运行，其方向根据距离值的符号决定。

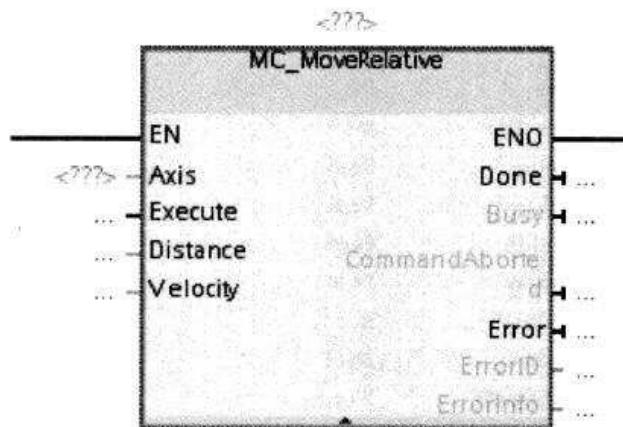


图 7-43 MC_MoveRelative 指令

绝对位置移动指令与相对位置移动的主要区别在于：是否需要建立起坐标系统（即是否需要参考点）。绝对位置移动指令需要知道目标位置在坐标系中的坐标，并根据坐标自动决定运动方向而不需要定义参考点；而相对位置移动只需要知道当前点与目标位置的距离（Distance），由用户给定方向，无需建立坐标系。

3. 变量定义与数据块分析

工作台滑动座电动机控制程序的变量定义见表 7-4。根据案例控制要求而建立的全局数据块 DB10 见表 7-5。

表 7-4 工作台滑动座电动机控制程序的变量定义

| | 名称 | 数据类型 | 地址 |
|----|---------|------|--------|
| 1 | 硬件左限位 | Bool | %I0.1 |
| 2 | 硬件右限位 | Bool | %I0.2 |
| 3 | 原点 | Bool | %I0.0 |
| 4 | 紧急停止按钮 | Bool | %I0.3 |
| 5 | 轴使能 | Bool | %Q0.4 |
| 6 | 紧急停止输出 | Bool | %Q0.5 |
| 7 | 输出脉冲PTO | Bool | %Q0.0 |
| 8 | 输出脉冲方向 | Bool | %Q0.1 |
| 9 | 实际位置 | Int | %MW126 |
| 10 | 设定位置 | Real | %MD128 |
| 11 | 设定速度 | Real | %MD132 |
| 12 | 故障字 | Word | %MW20 |
| 13 | MC故障 | Bool | %M21.0 |



表 7-5 根据案例控制要求而建立的全局数据块 DB10

| 名称 | 数据类型 | 偏移量 | 初始值 |
|-----------------------|------|------|-------|
| 1 ▼ Static | | | |
| 2 MC_enable | Bool | 0.0 | false |
| 3 Home_Active | Bool | 0.1 | false |
| 4 Halt | Bool | 0.2 | false |
| 5 Absolute_active | Bool | 0.3 | false |
| 6 Relative_active | Bool | 0.4 | false |
| 7 Velocity_active | Bool | 0.5 | false |
| 8 Reset_active | Bool | 0.6 | false |
| 9 Home_Mode | Int | 2.0 | 0 |
| 10 Velocity_direction | Int | 4.0 | 0 |
| 11 Velocity_value | Real | 6.0 | 0.0 |
| 12 Relative_value | Real | 10.0 | 0.0 |
| 13 Absolute_value | Real | 14.0 | 0.0 |
| 14 Home_Position | Real | 18.0 | 0.0 |
| 15 MC_Power_Busy | Bool | 22.0 | false |
| 16 MC_Power_Error | Bool | 22.1 | false |
| 17 MC_Power_ErrorID | Word | 24.0 | 0 |
| 18 MC_Power_ErrorInfo | Word | 26.0 | 0 |
| 19 Home_Done | Bool | 28.0 | false |
| 20 Home_Error | Bool | 28.1 | false |
| 21 Halt_Done | Bool | 28.2 | false |
| 22 Absolute_Done | Bool | 28.3 | false |
| 23 Relative_Done | Bool | 28.4 | false |
| 24 Reset_Done | Bool | 28.5 | false |

7.3.4 软件编程

图 7-44 所示为 OB1 主程序，其中，程序段 2 将轴 Axis_1 背景数据块的运动状态位置值直接显示在触摸屏上（即变量%MW126）。

▼ 块标题：

主程序

▼ 程序段 1：

调用FC1块

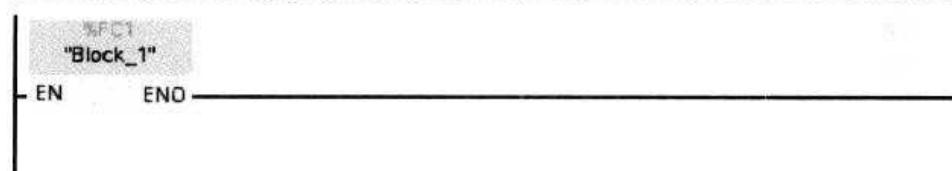
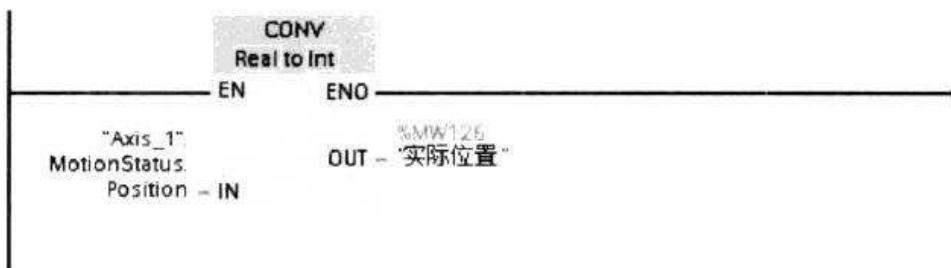


图 7-44 OB1 主程序



▼ 程序段 2：

将实际位置转为HMI动画显示



▼ 程序段 3：

MC故障

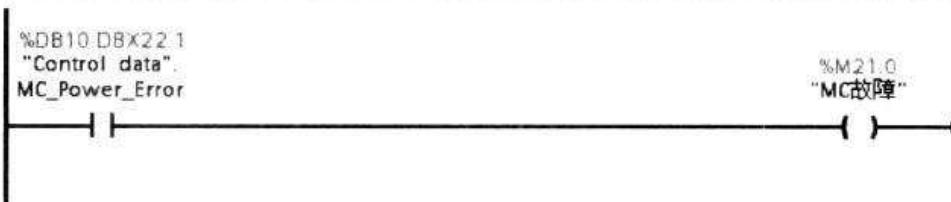


图 7-44 OBI 主程序 (续)

图 7-45 所示为 FC1 的 MC 功能调用块程序，分别调用 MC_Power、MC_Reset、MC_Home、MC_Halt、MC_MoveAbsolute 和 MC_MoveRelative，以实现触摸屏对步进电动机的操作。在程序编制过程中，采用 DB10 数据块来直接控制这些功能，具有形式一致、方便理解的特点。

▼ 块标题：

MC功能调用块

▼ 程序段 1：

紧急停止

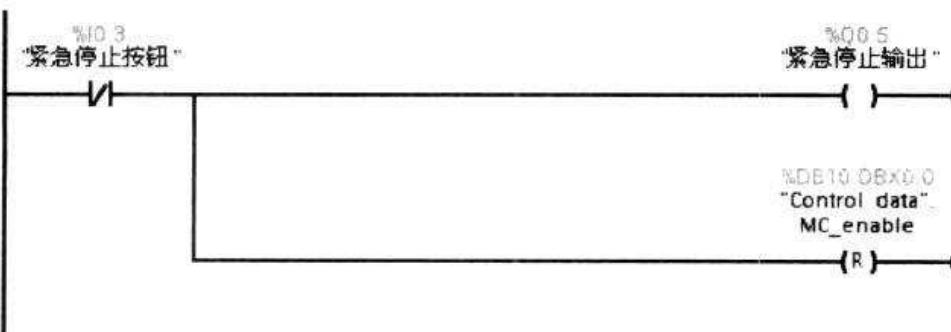


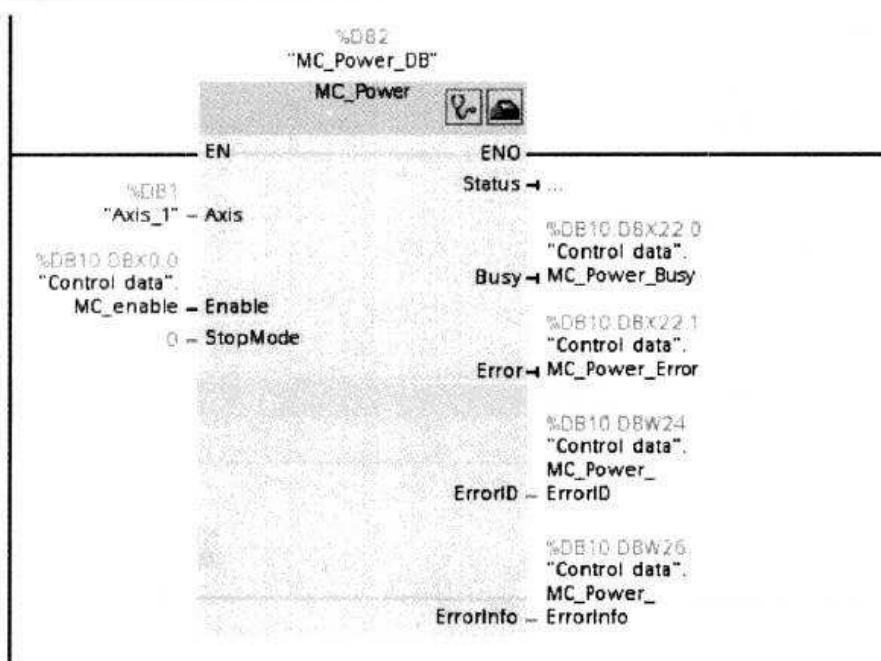
图 7-45 FC1 的 MC 功能调用块程序



图解西门子 S7-1200 PLC 入门到实践

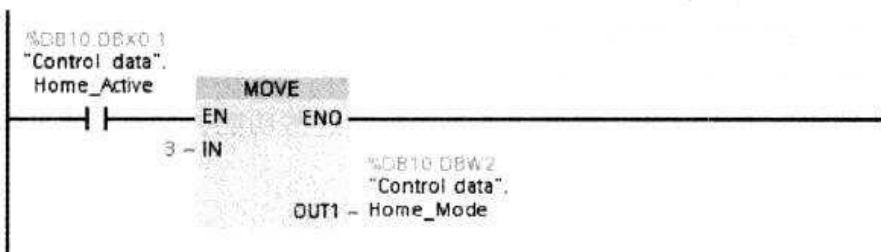
▼ 程序段 2：

使能MC功能



▼ 程序段 3：

当HOME功能激活时，定义HOME模式为3



▼ 程序段 4：

调用MC_Home指令

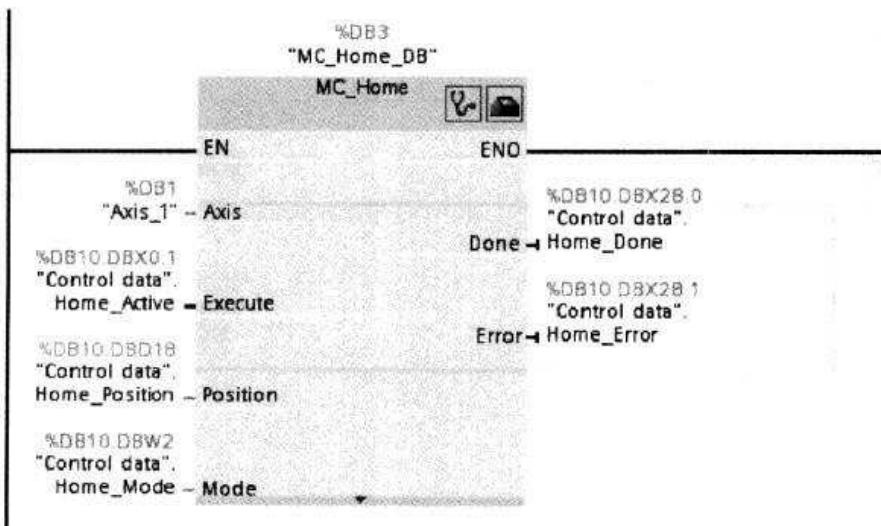


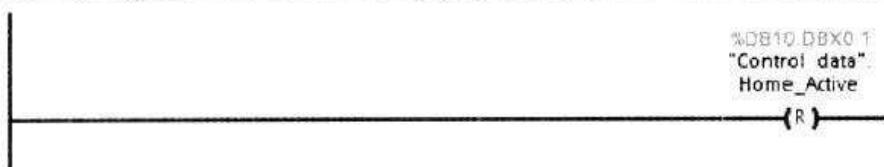
图 7-45 FCI 的 MC 功能调用块程序（续一）

第7章 S7-1200 PLC 的 PTO 控制



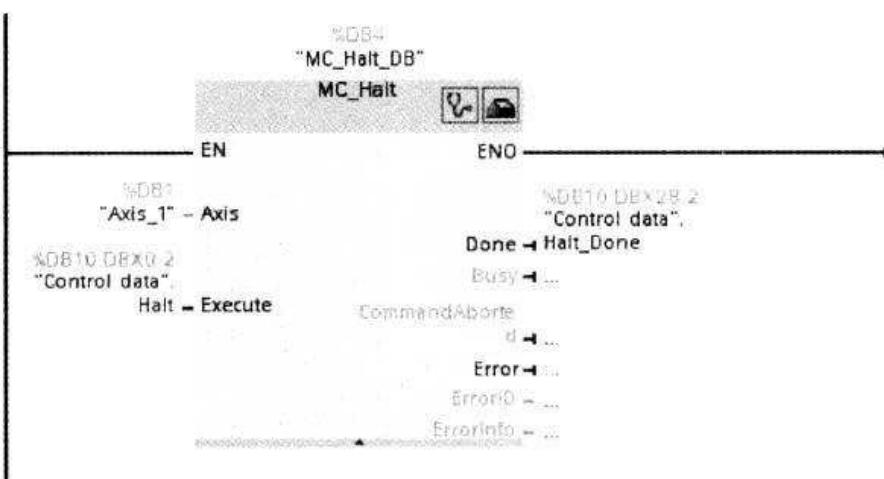
▼ 程序段 5：

复位HOME激活命令



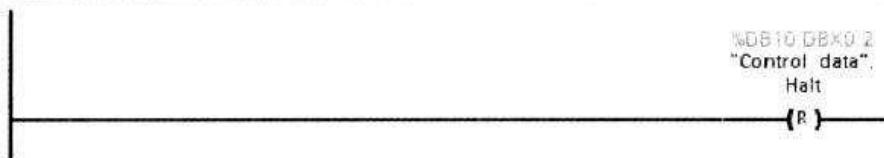
▼ 程序段 6：

调用Halt停止指令



▼ 程序段 7：

复位Halt激活命令



▼ 程序段 8：

当绝对位置移动时，定义HMI设定位置和设定速度

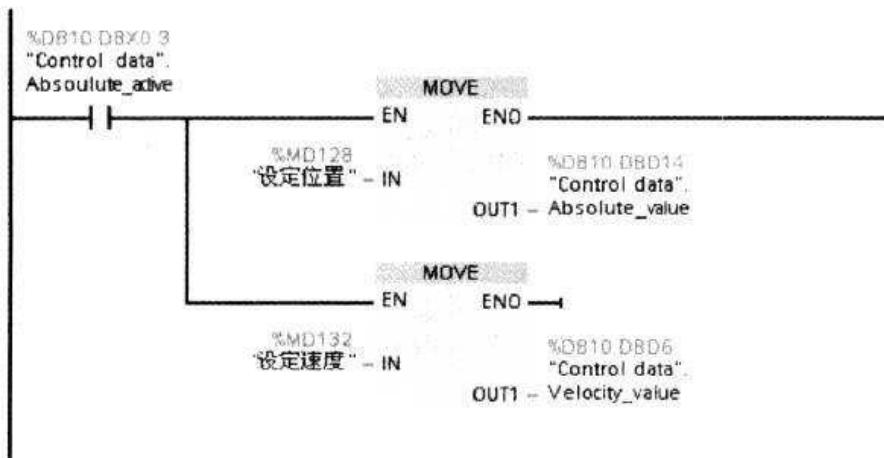


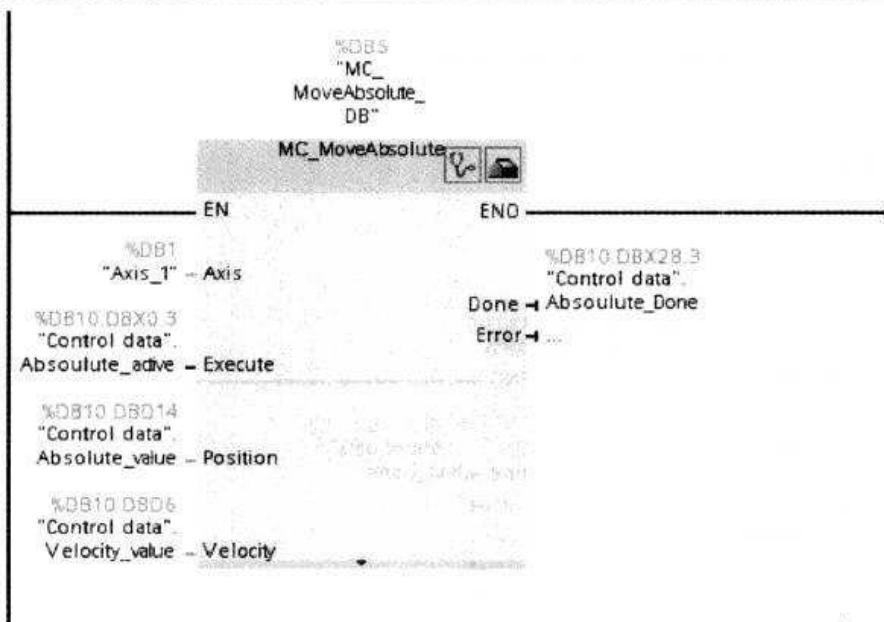
图 7-45 FCI 的 MC 功能调用块程序（续二）



图解西门子 S7-1200 PLC 入门到实践

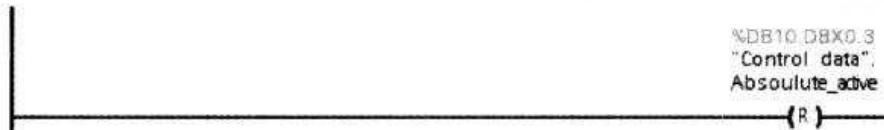
▼ 程序段 9：

调用绝对位置移动指令



▼ 程序段 10：

复位绝对位置移动激活命令



▼ 程序段 11：

相对位置移动

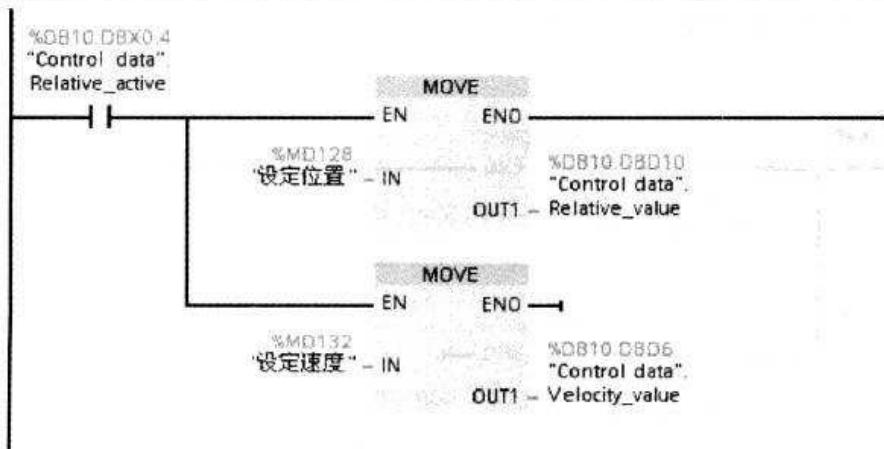
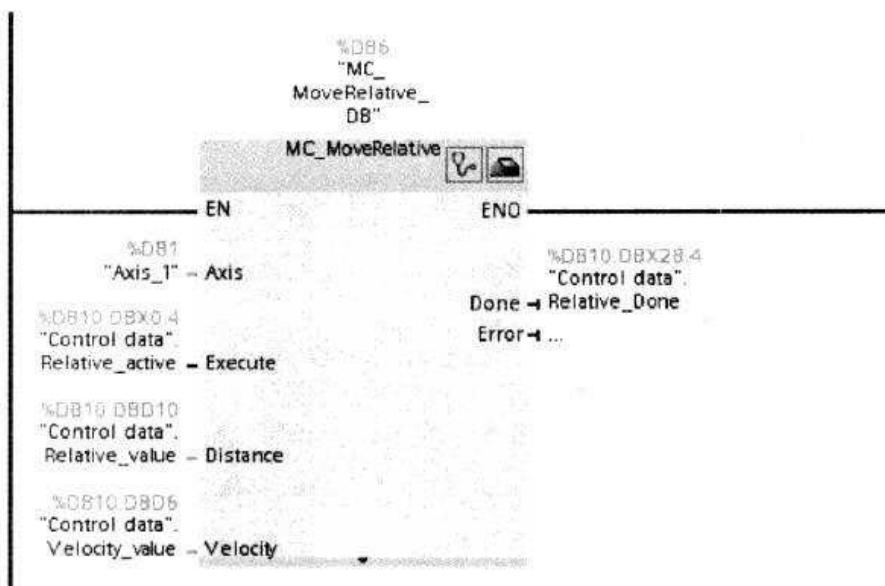


图 7-45 FCI 的 MC 功能调用块程序（续三）

第7章 S7-1200 PLC 的 PTO 控制

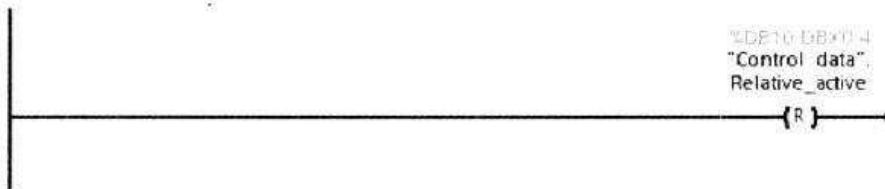
▼ 程序段 12 :

调用相对位置移动指令



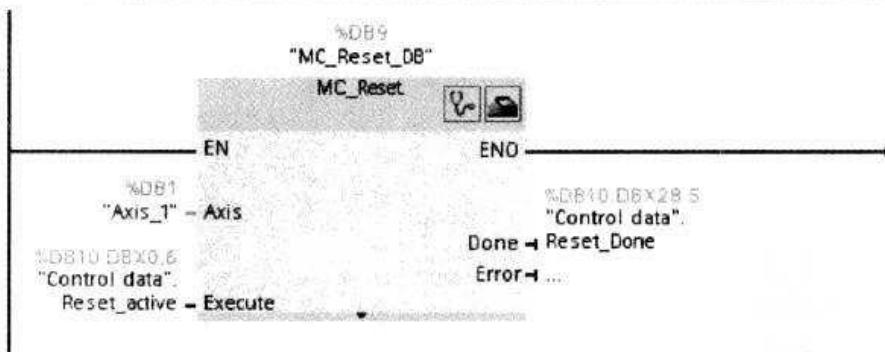
▼ 程序段 13 :

复位相对位置移动激活命令



▼ 程序段 14 :

调用MC复位指令



▼ 程序段 15 :

复位MC复位激活命令

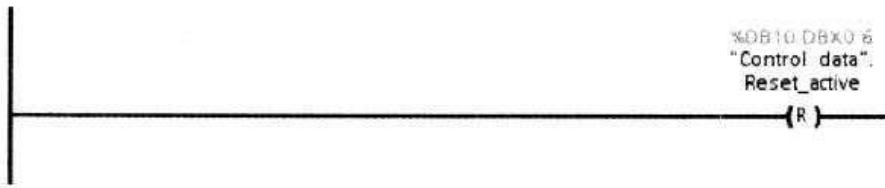


图 7-45 FC1 的 MC 功能调用块程序 (续四)



触摸屏画面组态

触摸屏的主要作用是通过设置参数来控制工作台滑动座电动机的原点定位、位置移动等功能，同时显示工作台滑动座电动机的实时位置、实时状态。

图 7-46 和图 7-47 所示为本案例的两个主要画面，即运动控制 HOME 画面（即回原点画面）和运动控制位置画面。需要注意的是，步进电动机运行之前，需要先进行回原点操作。

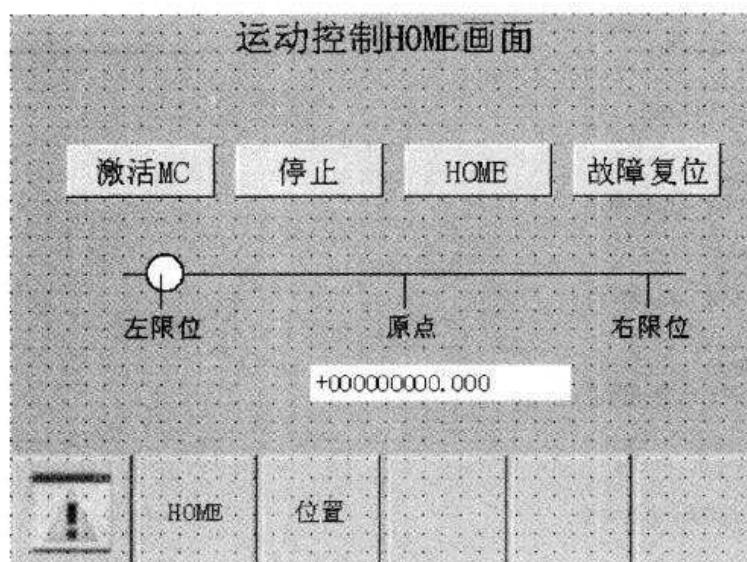


图 7-46 运动控制 HOME 画面

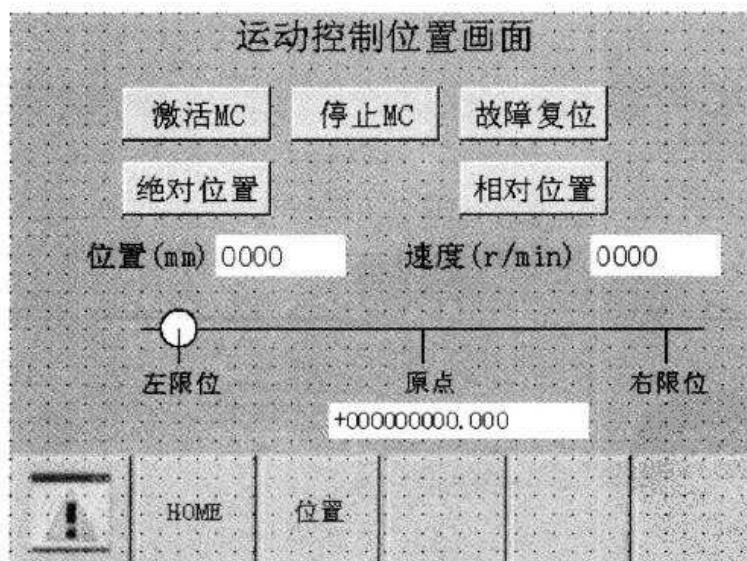


图 7-47 运动控制位置画面

图 7-48 所示详细介绍了运动控制 HOME 画面各个按钮的基本定义，比如“激活 MC”，即定义“按下”事件为 SetBit 操作。

图 7-49 所示详细介绍了运动控制 HOME 画面或位置画面中工作台滑动座电动机的“圆”动画，即定义指示灯“水平移动”动画，具体属性包括过程的变量值（这里为“实际位置”）、范围（这里为 -120 ~ 120）、起始位置 X 和 Y、目标位置 X 和 Y。

第7章 S7-1200 PLC 的 PTO 控制

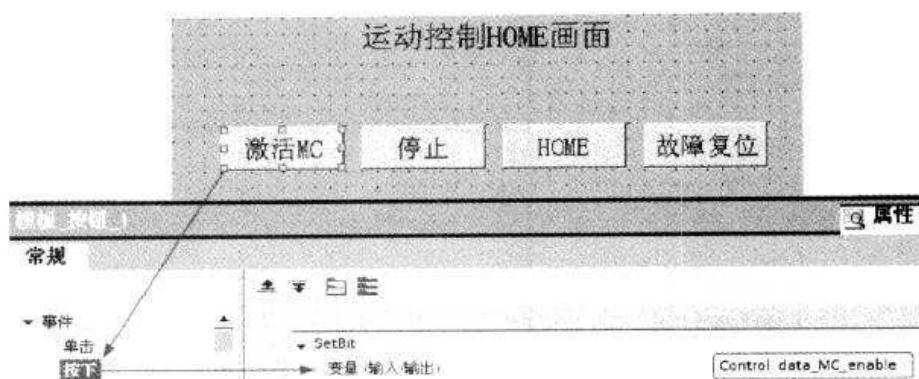


图 7-48 定义“激活 MC”按钮

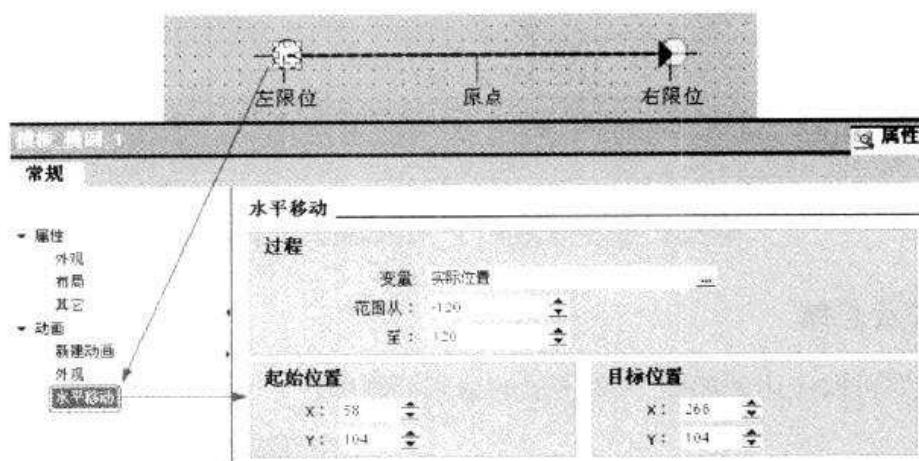


图 7-49 定义指示灯“水平移动”动画

图 7-50 所示详细介绍了运动控制 HOME 画面或位置画面中工作台滑动座电动机的位置数值显示属性，即定义为 Axis_1_MotionStatus_Position（见图 7-51），格式为十进制。

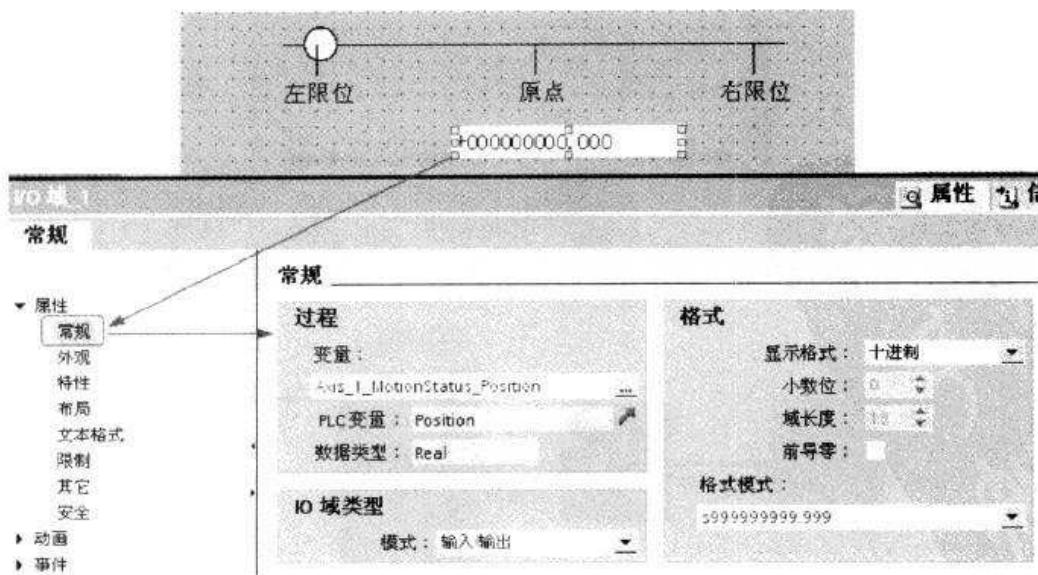


图 7-50 定义位置显示属性

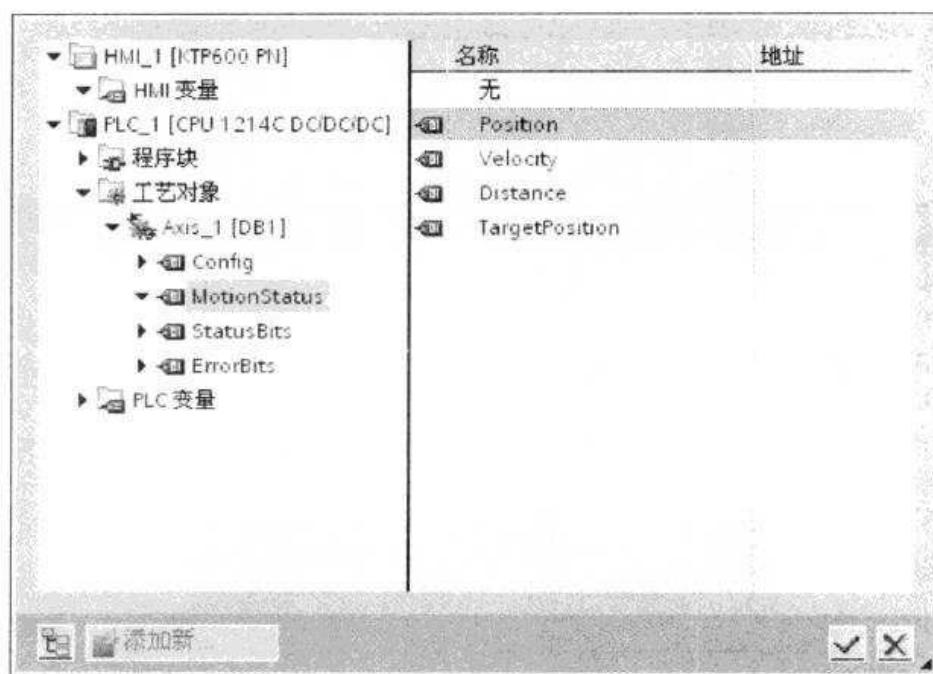


图 7-51 定义为 Axis_1_MotionStatus_Postion

7.3.5 调试及总结

本案例主要可以应用到如图 7-52 所示的钻床控制中。

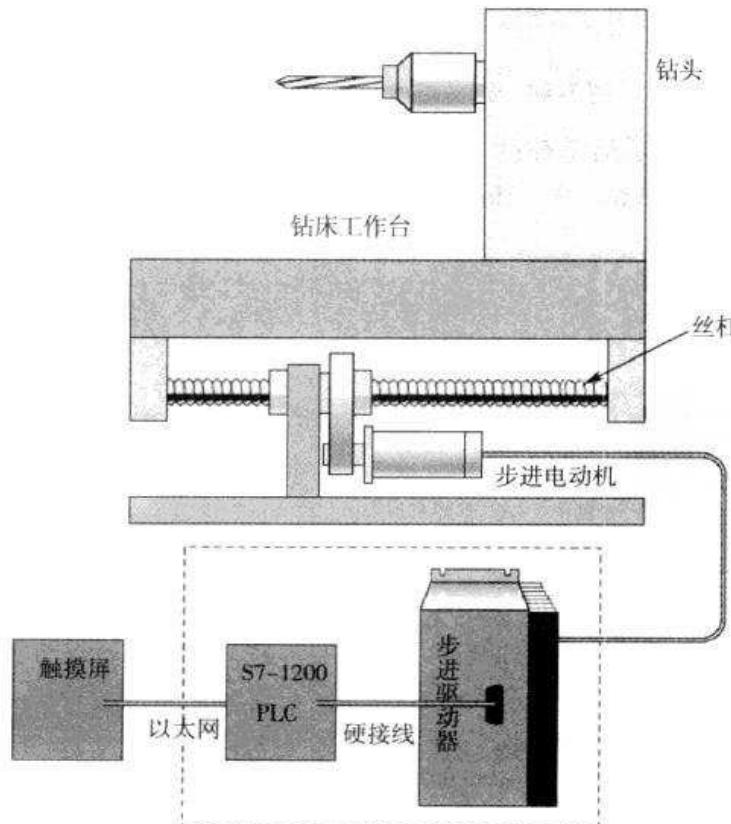


图 7-52 钻床控制

第7章 S7-1200 PLC 的 PTO 控制



图 7-53 所示为钻床运动控制主画面，执行 HOME 后，工作台滑动座回到原点，并显示位置值 +0.000。

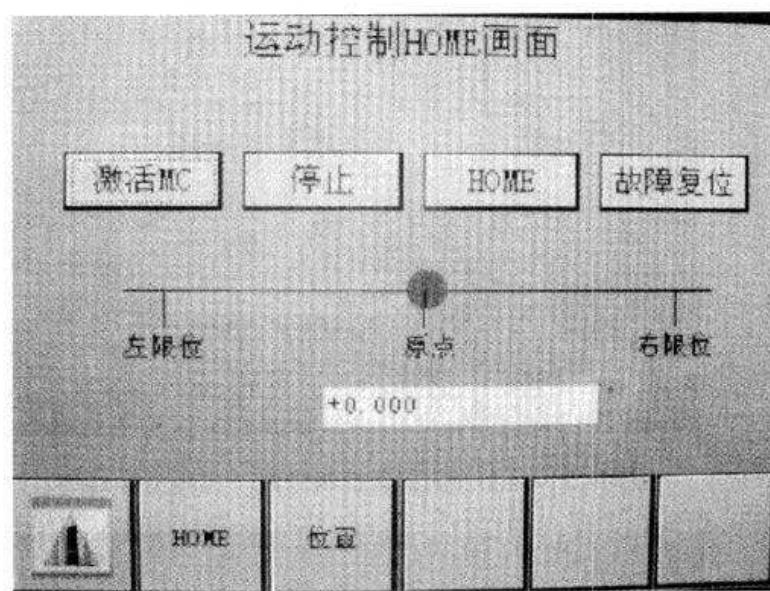


图 7-53 运动控制主画面

图 7-54 所示为运动控制位置控制画面（绝对位置），在输入位置值“45mm”，速度值“20r/min”后，工作台滑动座从原点向右以相对低速移动到“+45.000”。



图 7-54 运动控制位置控制画面（绝对位置）

图 7-55 所示为运动控制位置控制画面（相对位置），在输入位置值“45mm”，速度值“20r/min”后，工作台滑动座从原点向右以相对低速移动到“+90.000”。

图 7-56 所示为运动控制位置画面位置移动时的故障报警画面，如果当前位置值超过 +120.000mm 时，即达到软限位，认为 MC 故障，同时报警显示。

图 7-57 所示为运动控制位置画面报警显示窗口。



图 7-55 运动控制位置控制画面（相对位置）

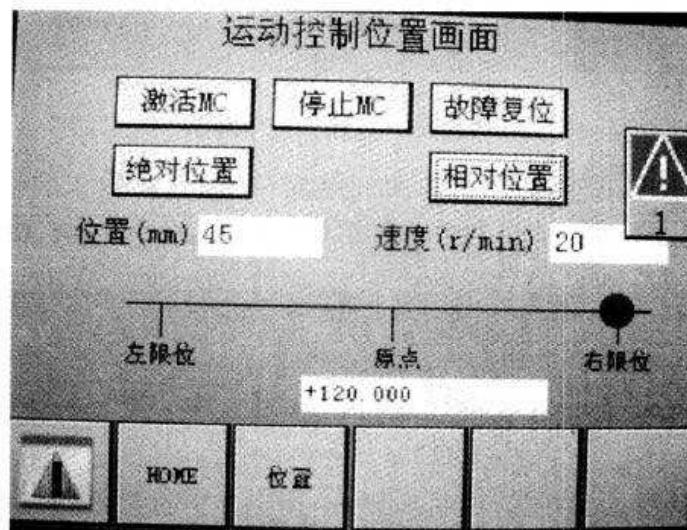


图 7-56 故障报警画面

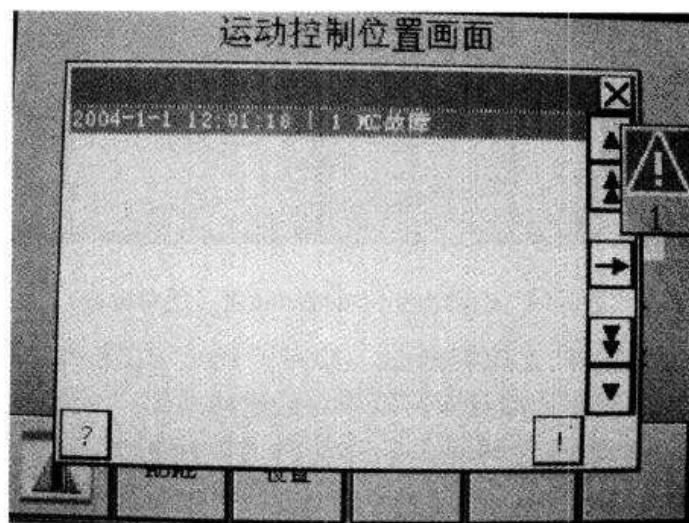


图 7-57 运动控制位置画面报警显示窗口



根据操作手册，MC 故障包括激活软件限位等一些常见的错误信息，主要分为以下 5 类：

- 1) 不造成使轴停止的错误，如错误类型 8200 为轴没有使能、8201 为轴被多个 MC_Power 指令使能、8204 为轴未回参考点等。
- 2) 使轴停止的错误，如 8001 为软件低限位触发、8002 为软件限位高限位触发、8003 为硬件限位低限位触发、8004 为硬件限位高限位触发等。
- 3) 指令块参数错误，如 8400 为运动控制语句的 Postion（位置）值参数不正确等。
- 4) 配置错误，如图 7-58 所示为故障代码 8613 配置错误，表示尽管硬件限位禁止，激活轴在硬件限位反转。

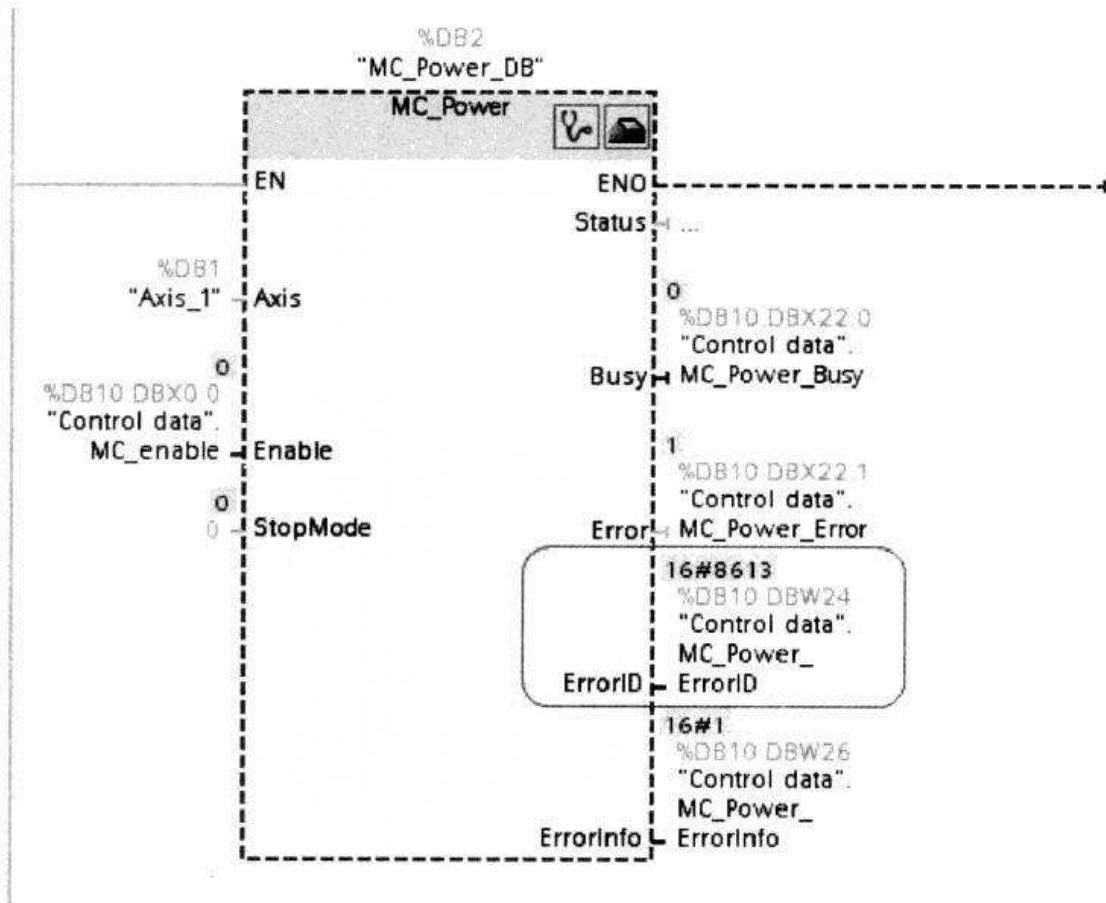
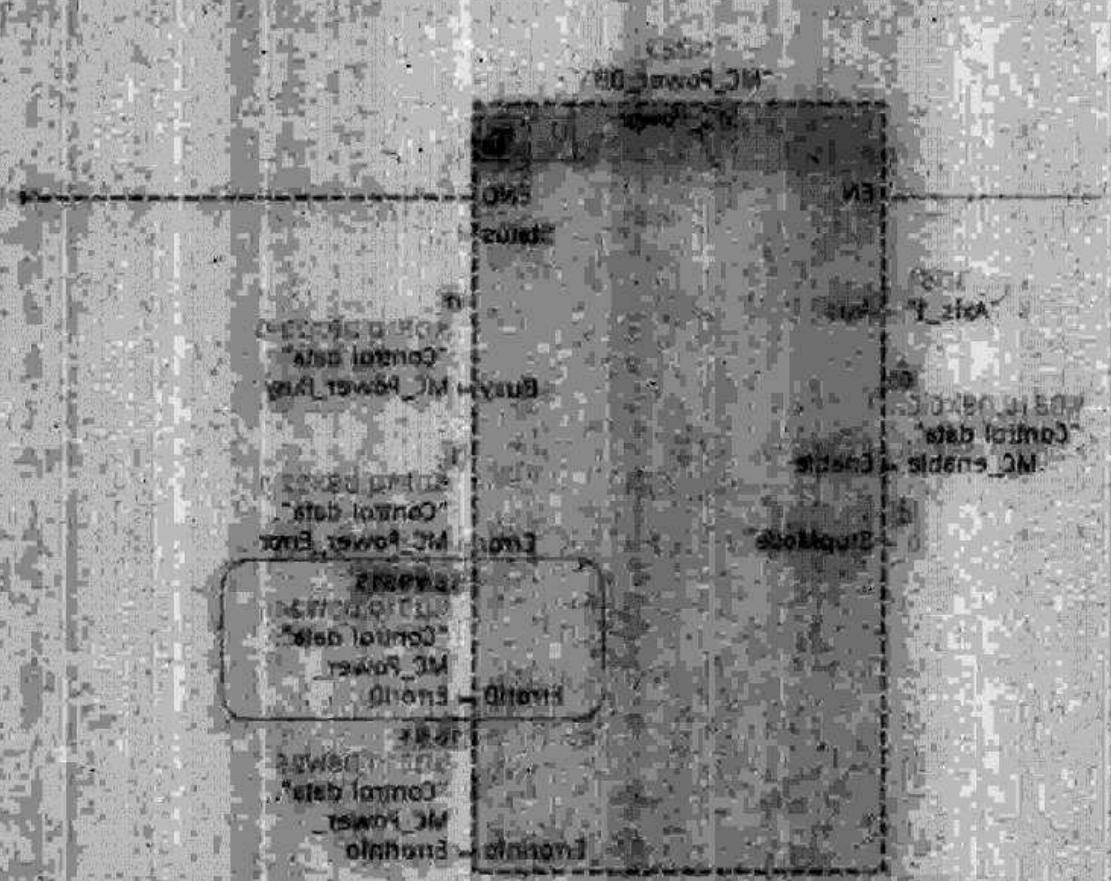


图 7-58 故障代码 8613 配置错误

- 5) 内部错误，代码为 8FFF。

图 8-109 MC980 PLC 第二层

已可以完成要求，但最好增加限位开关一节的接线并用热继电器 KM，但不能将限位开关直接接至 I008，而应通过端子 0008 采集集选信号，同时将 I008、I009 分别接至 E008、E009，这样过流保护接线为 S008，欠电压保护接线为 I008，启动停止按钮为 S008，停止按钮为常闭型，常闭点不接至 I008，而是接至 E008，同时接至常开型的常闭点，这样当常闭点接通时，常闭点接至 I008，常开点接至 E008，从而实现启停控制。



启停普通的 I008 与 I009 接线 82-5 图

1938 端钮外，另前端内 (2)

第8章 S7-1200 PLC的串口通信

西门子公司的S7-1200 紧凑型PLC 在当前的市场中有着广泛的应用，由于其性价比高，所以常被用作小型自动化控制设备的控制器，这也使得它经常与第三方的设备（扫描枪、打印机等设备）进行通信。本项目主要以超级终端为例介绍自由口通信，同时以PAC3200仪表为例介绍MODBUS的通信协议及其编程。

本章的学习目标如下：

知识目标

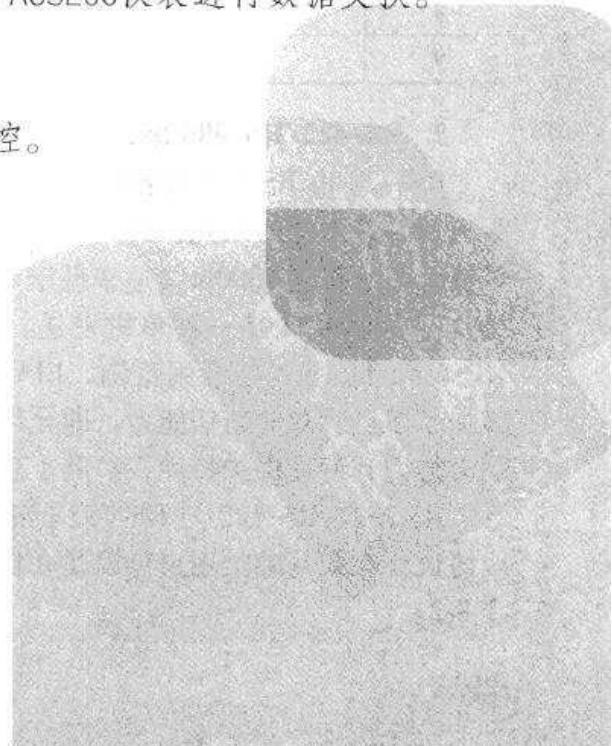
了解RS-232-C串口和RS-485串口的应用特点；熟悉S7-1200 PLC的通信扩展模块；熟悉S7-1200 PLC与其他设备进行串口通信的基本工作原理。

能力目标

能安装CM1241通信模块，并进行硬件组态和通信设置，并能通过自由口协议与超级终端进行通信、以MODBUS协议与PAC3200仪表进行数据交换。

职业素养目标

能了解不同设备的控制弱点，并能全面掌控。





8.1 背景知识介绍：串口通信与以太网通信

8.1.1 RS-232-C 串口和 RS-485 串口

1. RS-232C 串口

串行通信接口标准经过使用和发展，目前已经有几种，但都是在 RS-232 标准的基础上经过改进而形成的。所以，以 RS-232-C 为主来讨论。RS-232-C 标准是美国 EIA（电子工业联合会）与 BELL 等公司一起开发的于 1969 年公布的通信协议。它适合于数据传输速率在 0 ~ 20000bit/s 范围内的通信。这个标准对串行通信接口的有关问题，如信号线功能、电器特性等都做了明确规定。由于通信设备厂商都生产与 RS-232-C 制式兼容的通信设备，因此它作为一种标准，目前已在微机和 PLC 等通信接口中广泛采用。

对于 RS-232-C 来说，最常见的是 DB9 和 DB25，DB9 的外观如图 8-1 所示，其引脚名称及功能描述见表 8-1。

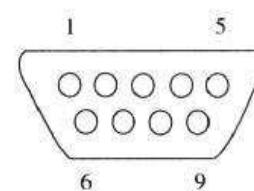


图 8-1 DB9 的外观

表 8-1 RS-232-C 的引脚名称及功能描述

| 针号 | 功能说明 | 缩写 |
|----|---------|------|
| 1 | 数据载波检测 | DCD |
| 2 | 接收数据 | RXD |
| 3 | 发送数据 | TXD |
| 4 | 数据终端准备 | DTR |
| 5 | 信号地 | GND |
| 6 | 数据设备准备好 | DSR |
| 7 | 请求发送 | RTS |
| 8 | 清除发送 | CTS |
| 9 | 振铃指示 | DELL |

2. RS-422/RS-485 串口

RS-422 由 RS-232 发展而来，它是为弥补 RS-232 的不足而提出的。为改进 RS-232 通信距离短、速率低的缺点，RS-422 定义了一种平衡通信接口，将传输速率提高到 10Mbit/s，传输距离延长到 4000in（速率低于 100kbit/s 时），并允许在一条平衡总线上连接最多 10 个接收器。RS-422 是一种单机发送、多机接收的单向、平衡传输规范，被命名为 TIA/EIA-422-A 标准。为扩展应用范围，EIA 又于 1983 年在 RS-422 的基础上制定了 RS-485 标准，增加了多点、双向通信能力，即允许多个发送器连接到同一条总线上，同时增加了发送器的驱动能力和冲突保护特性，扩展了总线共模范围。

RS-232、RS-422 与 RS-485 标准只对接口的电气特性做出规定，而不涉及接插件、电缆或协议，在此基础上用户可以建立自己的高层通信协议。三种不同的串口的电气参数比较见表 8-2。



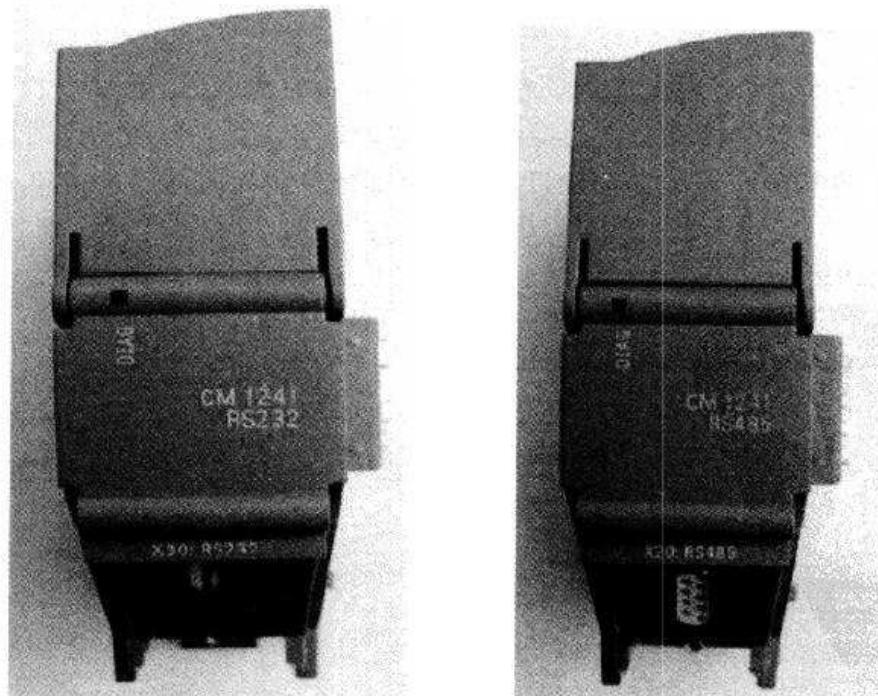
表 8-2 三种不同的串口的电气参数比较

| 规 定 | RS - 232 | RS - 422 | RS - 485 |
|--------------------|----------|--------------|-----------|
| 工作方式 | 单端 | 差分 | 差分 |
| 节点数 | 1 收、1 发 | 1 发 10 收 | 1 发 32 收 |
| 最大传输电缆长度/in | 50 | 400 | 400 |
| 最大传输速率 | 20kbit/s | 10Mbit/s | 10Mbit/s |
| 最大驱动输出电压/V | +/-25 | -0.25 ~ +6 | -7 ~ +12 |
| 驱动器输出信号电平(负载最小值)/V | 负载 | +/-5 ~ +/-15 | +/-2.0 |
| 驱动器输出信号电平(空载最大值)/V | 空载 | +/-25 | +/-6 |
| 驱动器负载阻抗/Ω | | 3k ~ 7k | 100 |
| 接收器输入电压范围/V | | +/-15 | -10 ~ +10 |
| 接收器输入门限 | | +/-3V | +/-200mV |
| 接收器输入电阻/Ω | | 3k ~ 7k | 4k (最小) |
| 驱动器共模电压/V | | | -3 ~ +3 |
| 接收器共模电压/V | | | -7 ~ +7 |
| | | | -7 ~ +12 |

注：1in = 0.0254m。

8.1.2 CM1241 RS-232 和 RS-485 模块

S7-1200 PLC 的串口通信模块共有两种型号，如图 8-2 所示，分别为 CM1241 RS-232 接口模块 (6ES7241-1AH30-0XB0) 和 CM1241 RS-485 接口模块 (6ES7241-1CH30-0XB0)，前者支持基于字符的自由口协议 (ASCII) 和 MODBUS RTU 主从协议；后者支持基于字符的自



a) CM1241 RS-232模块

b) CM1241 RS-485模块

图 8-2 CM1241 模块



由口协议（ASCII）、MODBUS RTU 主从协议和 USS 协议。

图 8-3 所示为 CM1241 模块与外部设备的通信连接示意，它为执行点对点通信提供电气连接，其特点包括：独立的 9 针 D-sub 端口；通过 LED 方式动态显示发送和接收；有一个诊断 LED；由 CPU 提供电源，不需要额外连接电源。

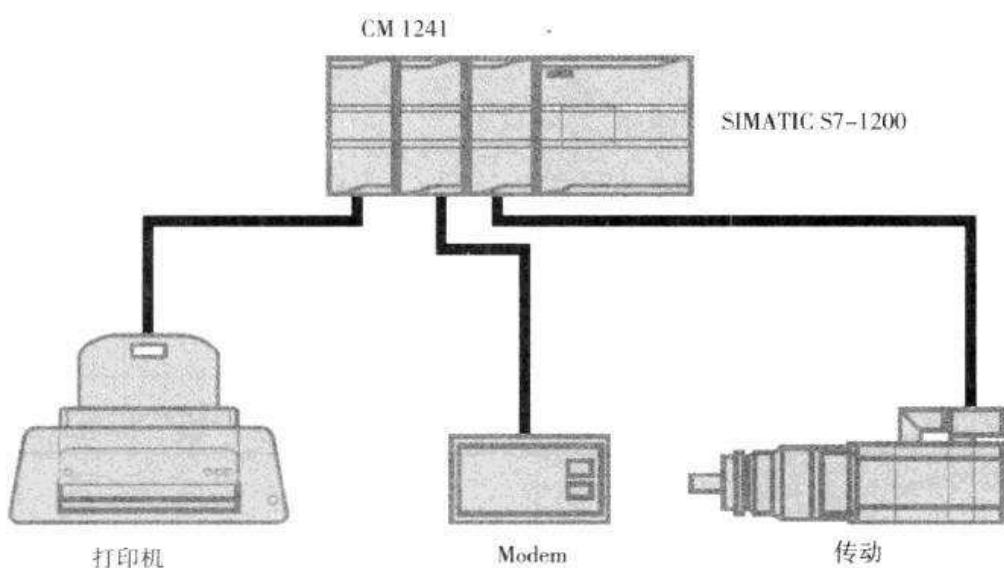


图 8-3 CM1241 模块与外部设备的通信连接示意

8.2 技能训练一：S7-1200 PLC 与 PC 超级终端的串口通信

8.2.1 案例介绍

西门子公司的 S7-1200 紧凑型 PLC 在当前的市场中有着广泛的应用，由于其性价比高，所以常被用作小型自动化控制设备的控制器，这也使得它经常与第三方的设备（扫描枪、打印机等设备）进行通信。现在要求以 PC 上的超级终端为例通过自由口通信建立与 S7-1200 PLC 的通信（见图 8-4），即 S7-1200 PLC 能发送信息到超级终端，而超级终端也能发送信息到 S7-1200 PLC。

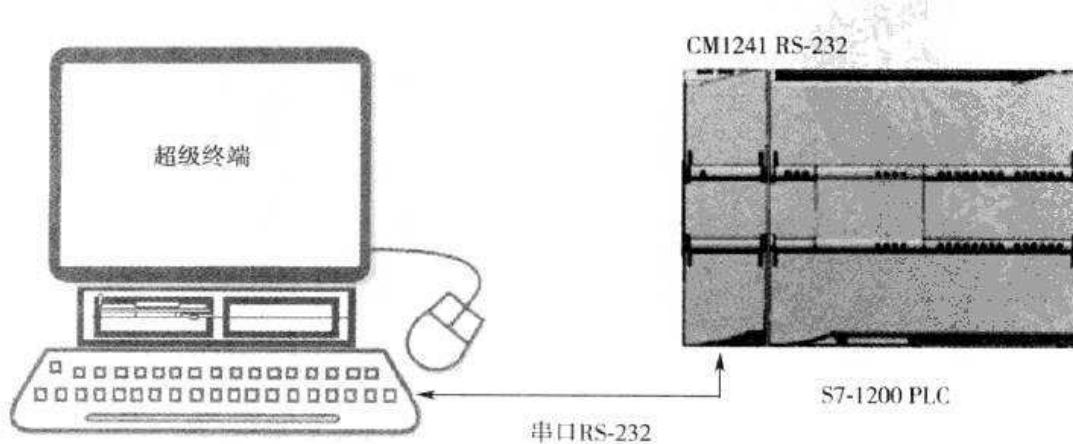


图 8-4 超级终端与 S7-1200 PLC 的通信



8.2.2 超级终端的设置

第一步，在Windows桌面上，单击“开始→程序→附件→通讯→超级终端”，为要建立的连接输入名称，如图8-5所示。



图8-5 打开超级终端

第二步，建立连接（见图8-6）。



图8-6 建立连接

第三步：选择连接时要使用的串口，如图8-7所示。

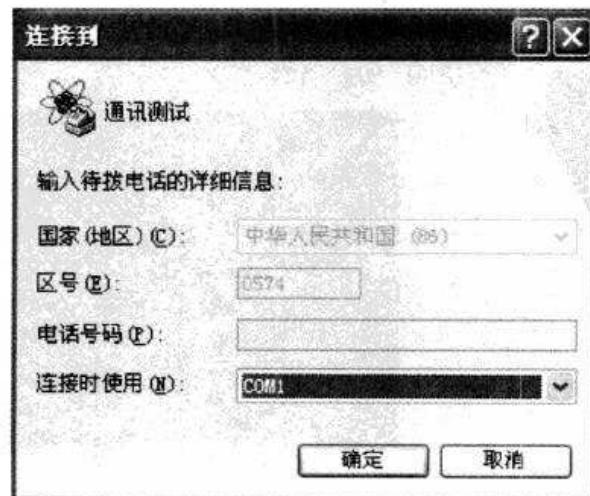


图8-7 选择串口



图解西门子 S7-1200 PLC 入门到实践

第四步：设置串口通信参数并保存连接，注意此处设置要与 PLC 程序中对应，如图 8-8 所示。

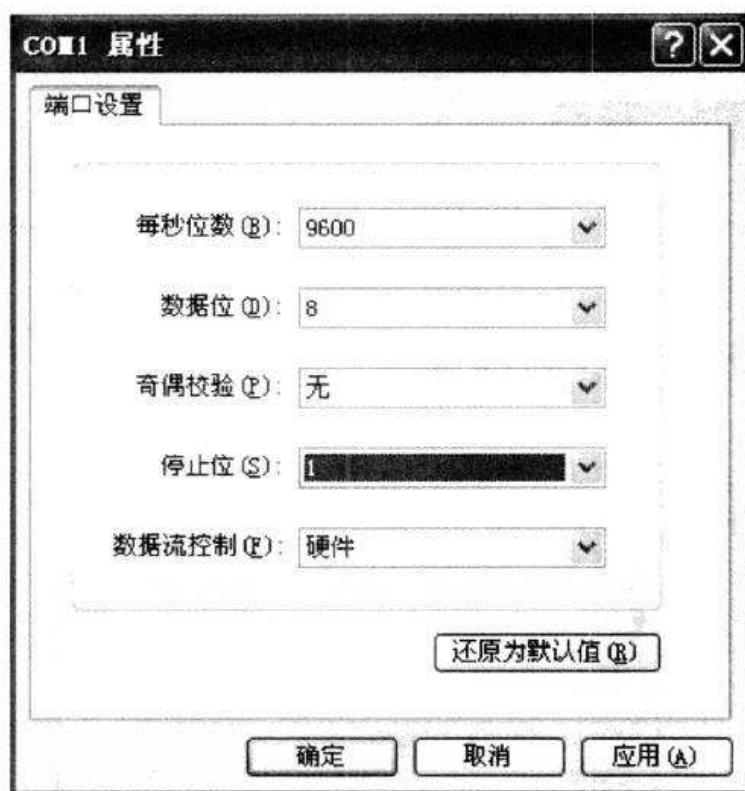


图 8-8 COM1 属性设置

8.2.3 S7-1200 PLC 的硬件组态

如图 8-9 所示，首先在设备列表里选择 CPU 的类型，插入 CPU 后，点击 CPU 左边的空

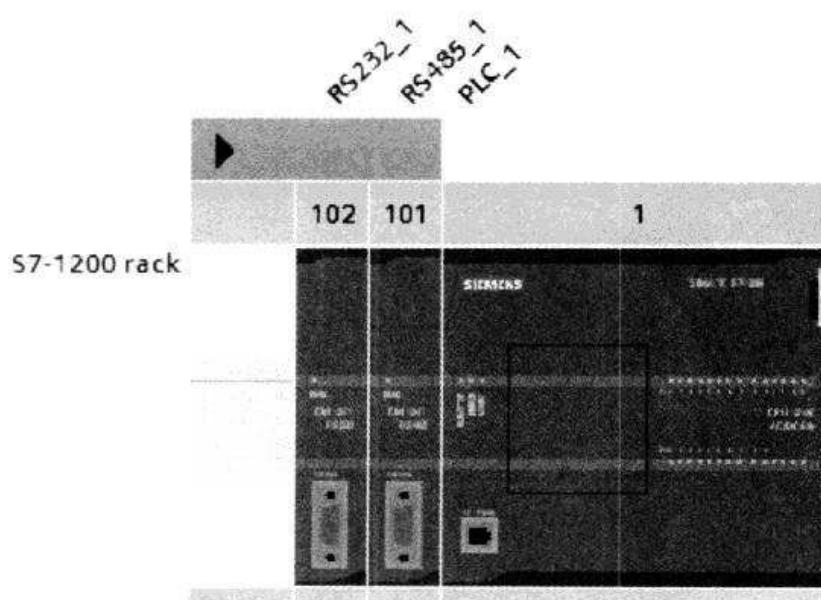


图 8-9 串口通信硬件组态

第8章 S7-1200 PLC 的串口通信



槽，在右边的“硬件目录”里找到“通信模块”下的 RS-232 模块或 RS-485 模块（见图 8-10），拖拽或双击此模块，这样就把串口模块插入到硬件配置里。



图 8-10 通信模块硬件目录

接下来就需要配置此 RS-232 模块硬件接口参数，选择 RS-232 模块，在其下方会出现该模块的硬件属性配置窗口（见图 8-11），在属性窗口里有两个选项，一个是“常规”；一个是“RS232 接口”。

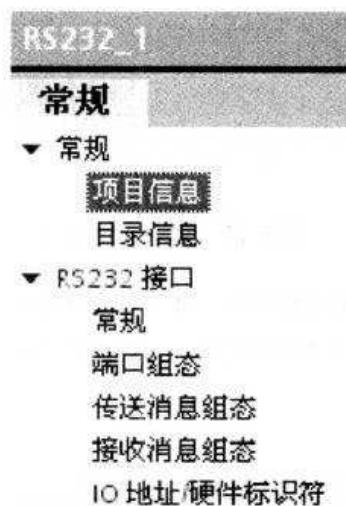


图 8-11 硬件属性配置窗口

在“常规”里包括了此模块的“项目信息”（见图 8-12）和“目录信息”（见图 8-13）；而在“RS232 接口”里包括“常规”、“端口组态”、“传送消息组态”、“接收消息组态”和

项目信息



图 8-12 项目信息窗口



图解西门子 S7-1200 PLC 入门到实践

“IO 地址/硬件识别符”。

目录信息

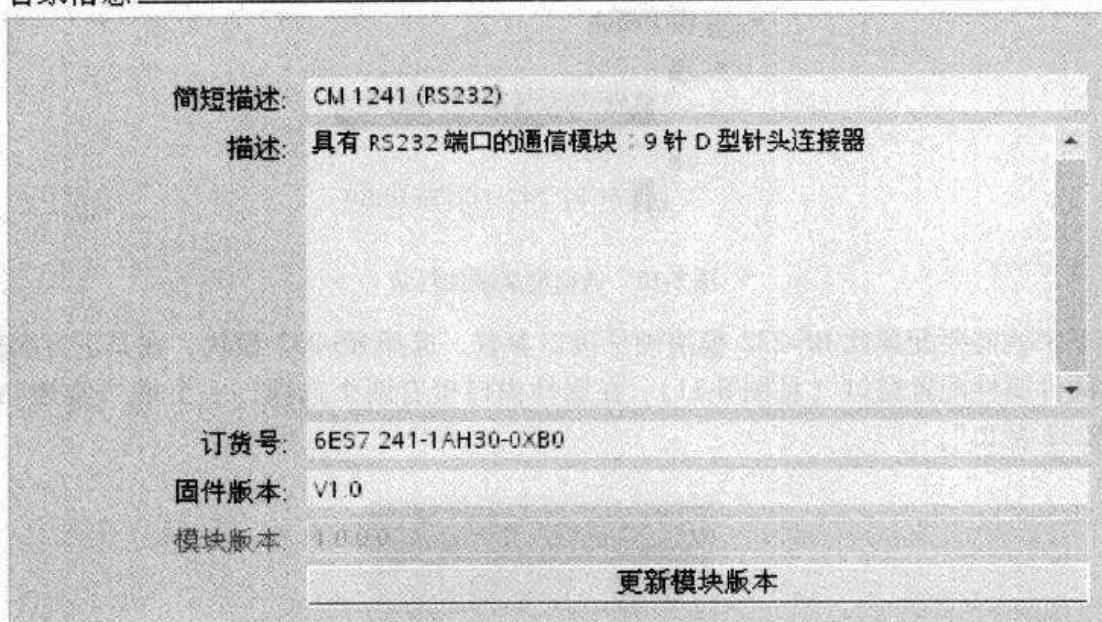


图 8-13 目录信息窗口

如图 8-14 所示，选择“RS232 接口”，在“端口组态”的选项里，进行端口的参数配置，具体为：波特率为 9.6kbit；校验方式无；数据位为 8；停止位为 1；硬件流控制无；等待时间为 1ms。

端口组态

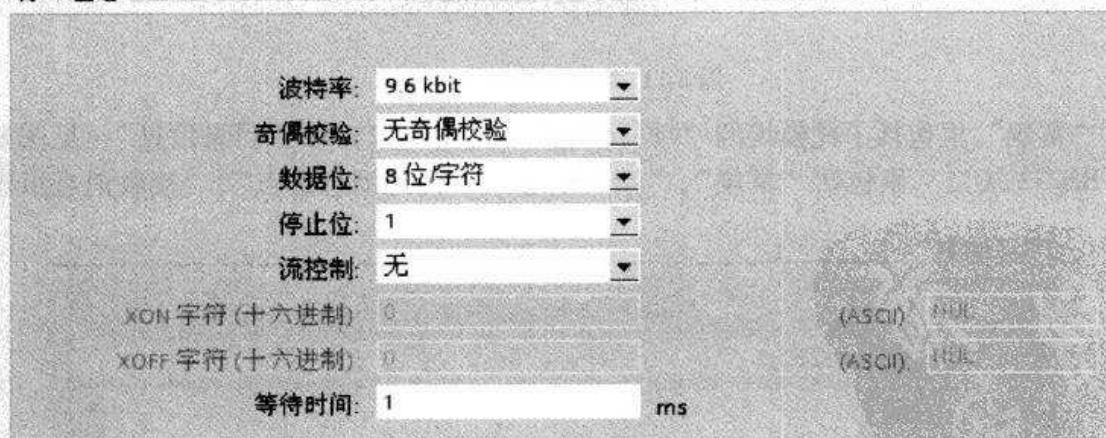


图 8-14 端口组态

图 8-15 所示为发送消息组态。

图 8-16 和图 8-17 所示为接收消息组态的消息开始部分和消息结束部分。

第8章 S7-1200 PLC 的串口通信

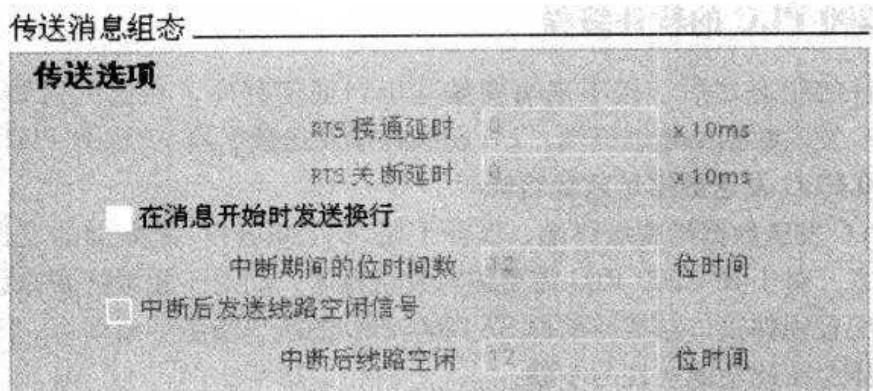


图 8-15 传送消息组态

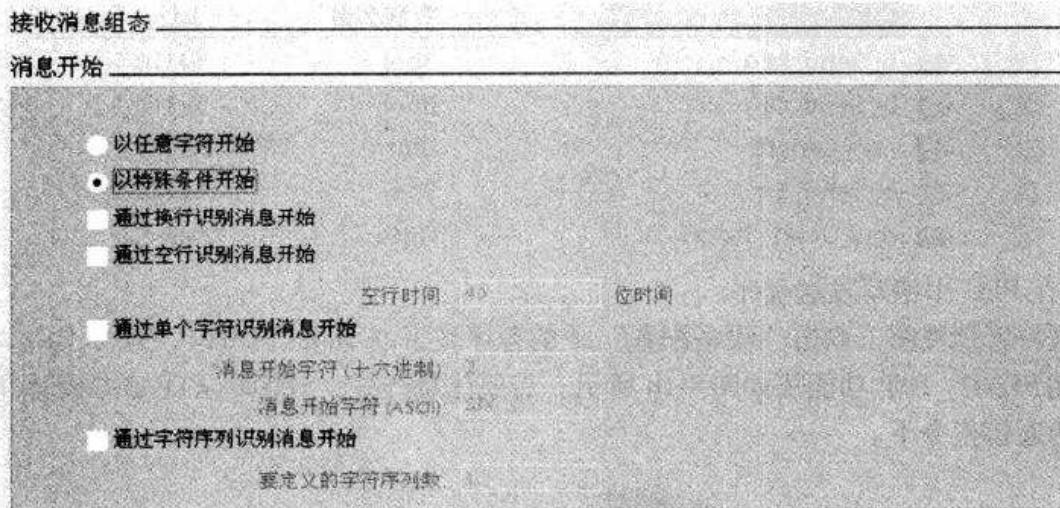


图 8-16 接收消息组态 (消息开始)

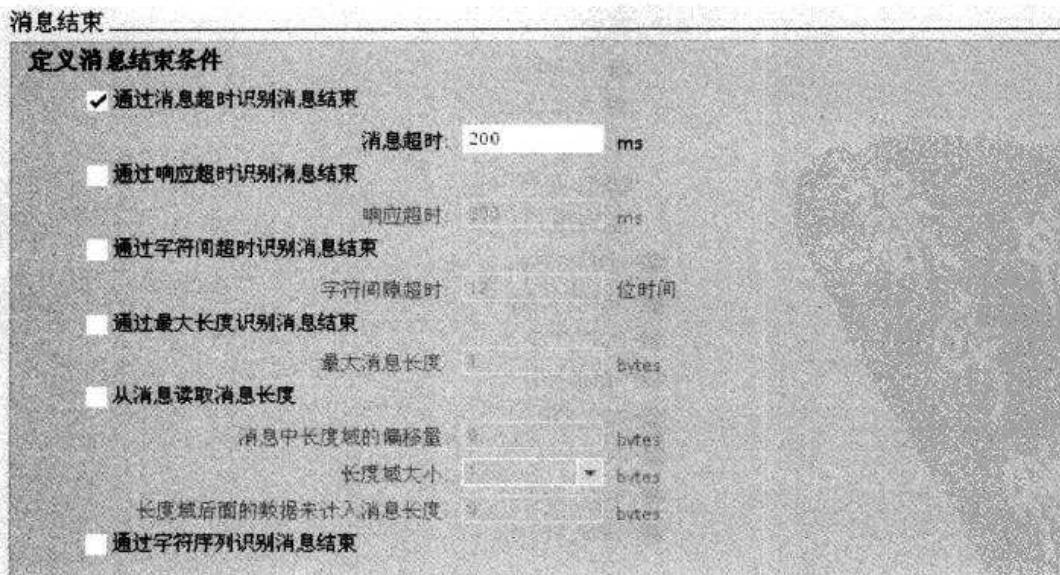


图 8-17 接收消息组态 (消息结束)



8.2.4 S7-1200 PLC 的软件编程

完成了硬件的组态之后，接下来需要编写串口通信程序，在这里需要实现两个功能：

- S7-1200 PLC 发送数据给超级终端；2) 超级终端发送数据给 S7-1200 PLC。

1. S7-1200 PLC 发送数据给超级终端

S7-1200 PLC 发送数据给超级终端，实际上是 S7-1200 PLC 是数据的发送方，超级终端是数据的接收方，对于 S7-1200 PLC 需要编写发送程序；而对于超级终端来说，只要打开超级终端程序，配置硬件接口参数与前面 S7-1200 PLC 的端口参数一致即可。

下面将具体介绍此功能实现的步骤：

- 变量定义见表 8-3。

表 8-3 变量定义

| | 名称 | 数据类型 | 地址 |
|---|----------------|------|--------|
| 1 | b_Send_Req | Bool | %M10.0 |
| 2 | b_Send_OK | Bool | %M10.1 |
| 3 | mw_Length | UInt | %MW12 |
| 4 | b_Send_Err | Bool | %M10.2 |
| 5 | mw_Send_Status | Word | %MW14 |

- 在 PLC 中编写发送程序。

在项目管理视图下双击“设备配置”下的程序块下的 Main (OB1)，打开 OB1，在主程序中调用 SEND _PTP 功能块如图 8-18 所示。需要注意的是：SEND _PTP 在指令库下的扩展指令中的通信指令下。

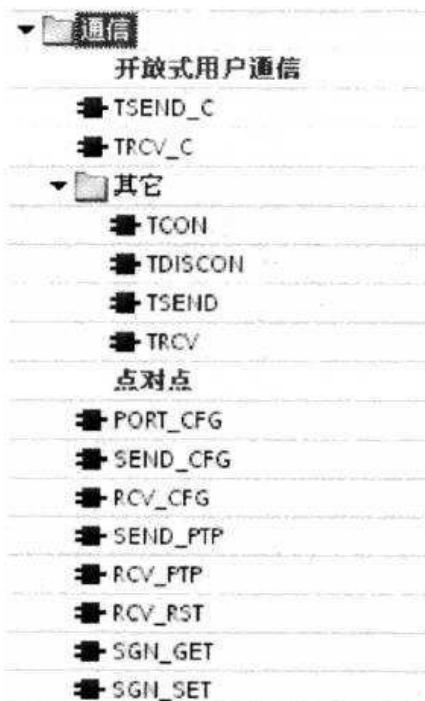


图 8-18 调用发送功能块

第8章 S7-1200 PLC 的串口通信



图 8-19 为调用 SEND _ PTP 的主程序。

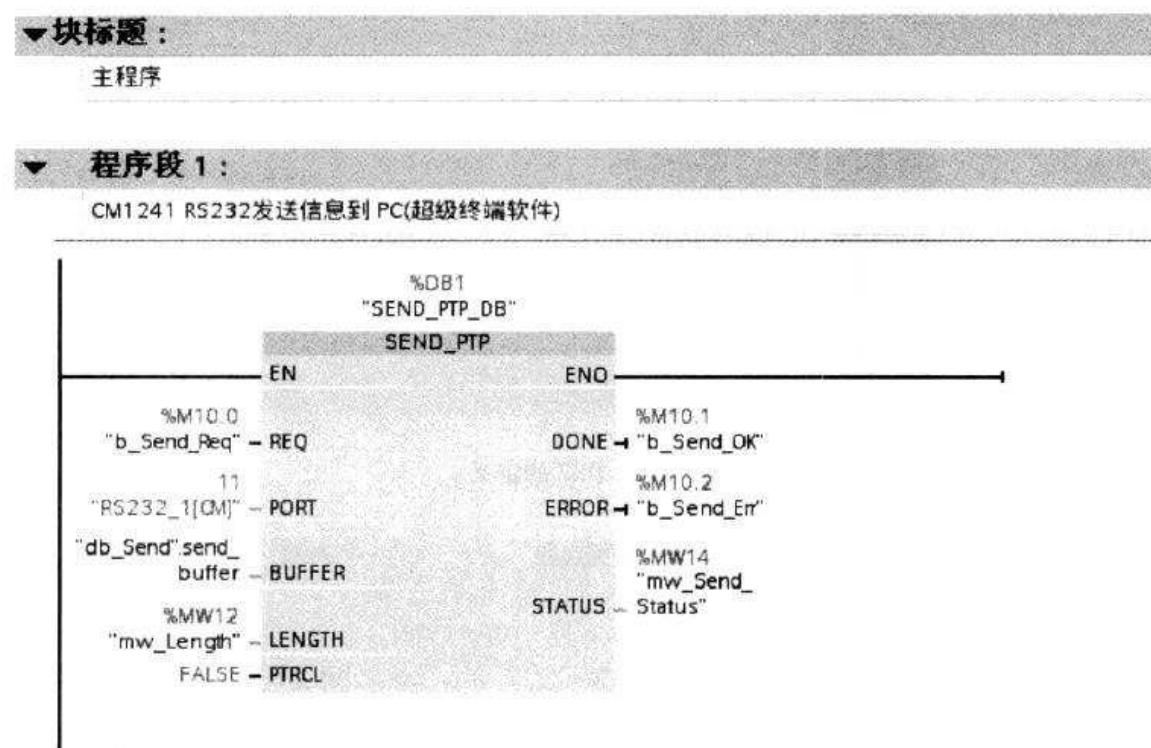


图 8-19 调用 SEND _ PTP 的主程序

显然，在本程序中用到了如图 8-20 所示的两块 DB，其中 DB1 为 SEND _ PTP 的背景 DB (见图 8-21)，DB2 为发送缓冲 DB。



图 8-20 程序块结构

| 名称 | 数据类型 | 初始值 |
|------------|---------|--------|
| 1 ▼ Input | | |
| 2 REQ | Bool | False |
| 3 PORT | PORT | 0 |
| 4 BUFFER | VARIANT | |
| 5 LENGTH | UInt | 0 |
| 6 PTRCL | Bool | False |
| 7 ▼ Output | | |
| 8 DONE | Bool | False |
| 9 ERROR | Bool | False |
| 10 STATUS | Word | W#16#0 |

图 8-21 DB1 (SEND _ PTP 指令的 I/O 参数)



图解西门子 S7-1200 PLC 入门到实践

如图 8-22 所示，需要选择 DB2 类型为“全局 DB”，并去掉“仅符号访问”选项勾，这样可以对该 DB 进行直接地址访问。图 8-23 所示为 DB2 (SEND _ PTP 发送缓冲区)。



图 8-22 DB2 的定义

| | 名称 | 数据类型 | 初始值 |
|----|-----------------|-------------------------|---------|
| 1 | ▼ Static | | |
| 2 | ▼ send_buffer | Array [1 .. 10] of BYTE | |
| 3 | send_buffer[1] | Byte | B#16#00 |
| 4 | send_buffer[2] | Byte | B#16#00 |
| 5 | send_buffer[3] | Byte | B#16#00 |
| 6 | send_buffer[4] | Byte | B#16#00 |
| 7 | send_buffer[5] | Byte | B#16#00 |
| 8 | send_buffer[6] | Byte | B#16#00 |
| 9 | send_buffer[7] | Byte | B#16#00 |
| 10 | send_buffer[8] | Byte | B#16#00 |
| 11 | send_buffer[9] | Byte | B#16#00 |
| 12 | send_buffer[10] | Byte | B#16#00 |

图 8-23 DB2 (SEND _ PTP 发送缓冲区)

建好这两个 DB 后，打开如图 8-24 所示的监视表格，双击打开 DB _ SEND _ BUFF 预先定义好要发送的数据 (见图 8-25)。

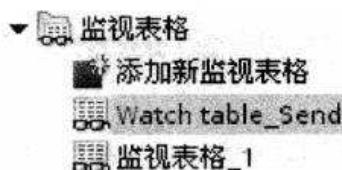


图 8-24 打开监视表格

然后使用超级终端接收 S7-1200 PLC 发送的信息，如图 8-26 所示。

2. 超级终端发送数据给 S7-1200 PLC

超级终端发送数据给 S7-1200 PLC，实际上是 S7-1200 PLC 是数据的接收方，超级终端是数据的发送方，对于 S7-1200 PLC 需要编写接收程序；而对于超级终端来说，只要打开超级终端程序，配置硬件接口参数与前面 S7-1200 PLC 的端口参数一致，在界面上输入发送内



| | 名称 | 地址 | 显示格式 | 监视值 | 修改值 |
|----|---------------------------|--------|------|------|-----|
| 1 | "b_Send_Req" | %M10.0 | 布尔型 | TRUE | |
| 2 | "db_Send".send_buffer[1] | | 字符 | 'H' | |
| 3 | "db_Send".send_buffer[2] | | 字符 | 'o' | |
| 4 | "db_Send".send_buffer[3] | | 字符 | 'w' | |
| 5 | "db_Send".send_buffer[4] | | 字符 | '' | |
| 6 | "db_Send".send_buffer[5] | | 字符 | 'a' | |
| 7 | "db_Send".send_buffer[6] | | 字符 | 'r' | |
| 8 | "db_Send".send_buffer[7] | | 字符 | 'e' | |
| 9 | "db_Send".send_buffer[8] | | 字符 | '' | |
| 10 | "db_Send".send_buffer[9] | | 字符 | 'u' | |
| 11 | "db_Send".send_buffer[10] | | 字符 | '?' | |
| 12 | "mw_Length" | %MW12 | 十进制 | 10 | |
| 13 | "b_Send_OK" | %M10.1 | 布尔型 | | |
| 14 | "b_Send_Err" | %M10.2 | 布尔型 | | |
| 15 | "mw_Send_Status" | %MW14 | 十六进制 | | |

图 8-25 预定义值

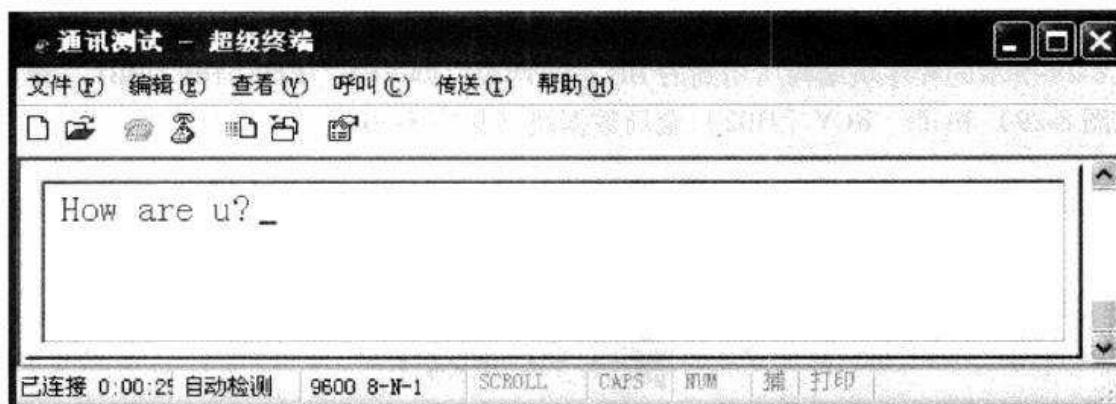


图 8-26 超级终端接收窗口

容即可。

下面将具体介绍此功能实现的步骤：

1) 定义变量见表 8-4。

表 8-4 变量定义

| | 名称 | 数据类型 | 地址 |
|---|--------------|------|--------|
| 1 | b_RCV_Req | Bool | %M20.0 |
| 2 | b_NDP | Bool | %M20.1 |
| 3 | b_RCV_Err | Bool | %M20.2 |
| 4 | mw_RCVStatus | Word | %MW22 |
| 5 | mw_Length | UInt | %MW24 |

2) 在 PLC 中编写接收程序。

在项目管理视图下双击“Device”下的程序块下的 Main (OB1)，打开 OB1，在主程序



中调用 RCV _ PTP 功能块如图 8-27 所示（注：RCV _ PTP 在指令库下的扩展指令中的通信指令下）。

▼ 程序段 1：

PC超级终端软件) 发送信息给CM1241 RS232

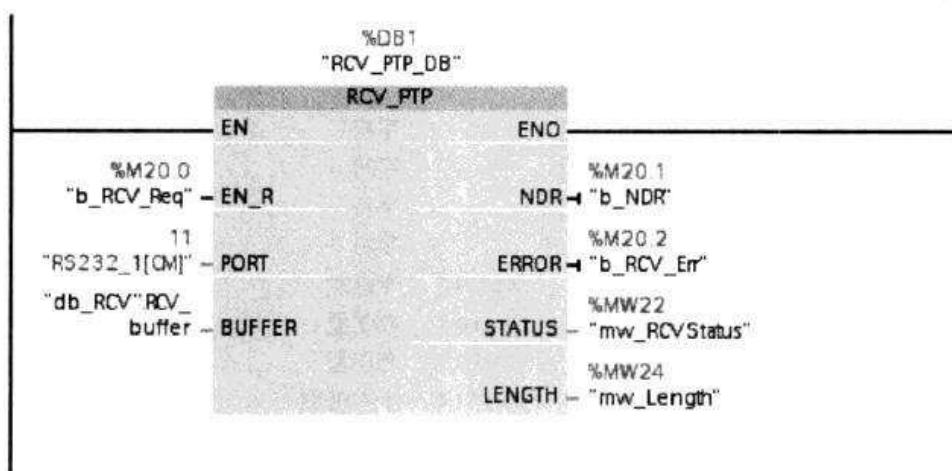


图 8-27 在主程序中调用 RCV _ PTP 功能块

图 8-28 所示的程序块结构所用到的 DB 也是两个，即 RCV _ PTP _ DB [DB1] 背景数据块（见图 8-29）和 db _ RCV [DB2] 全局数据块（见图 8-30）。



图 8-28 程序块结构

| | 名称 | 数据类型 | 初始值 |
|---|----------|---------|--------|
| 1 | ▼ Input | | |
| 2 | EN_R | Bool | False |
| 3 | PORT | PORT | 0 |
| 4 | BUFFER | VARIANT | |
| 5 | ▼ Output | | |
| 6 | NDR | Bool | False |
| 7 | ERROR | Bool | False |
| 8 | STATUS | Word | W#16#0 |
| 9 | LENGTH | UInt | 0 |

图 8-29 DB1 变量



| | 名称 | 数据类型 | 初始值 |
|----|----------------|-------------------------|---------|
| 1 | ▼ Static | | |
| 2 | ▼ RCV_buffer | Array [1 .. 15] of BYTE | |
| 3 | RCV_buffer[1] | Byte | B#16#00 |
| 4 | RCV_buffer[2] | Byte | B#16#00 |
| 5 | RCV_buffer[3] | Byte | B#16#00 |
| 6 | RCV_buffer[4] | Byte | B#16#00 |
| 7 | RCV_buffer[5] | Byte | B#16#00 |
| 8 | RCV_buffer[6] | Byte | B#16#00 |
| 9 | RCV_buffer[7] | Byte | B#16#00 |
| 10 | RCV_buffer[8] | Byte | B#16#00 |
| 11 | RCV_buffer[9] | Byte | B#16#00 |
| 12 | RCV_buffer[10] | Byte | B#16#00 |
| 13 | RCV_buffer[11] | Byte | B#16#00 |
| 14 | RCV_buffer[12] | Byte | B#16#00 |
| 15 | RCV_buffer[13] | Byte | B#16#00 |
| 16 | RCV_buffer[14] | Byte | B#16#00 |
| 17 | RCV_buffer[15] | Byte | B#16#00 |

图 8-30 DB2 变量

3) 在超级终端中输入字符串“PCSENDTOCPU1214”，即可看到 PLC 中的监视值（见图 8-31）。如果输入字符串位数不足，则看到的监视值为‘\$00’。

| | 名称 | 地址 | 显示格式 | 监视值 |
|----|-------------------------|--------|------|------|
| 1 | "b_RCV_Req" | %M20.0 | 布尔型 | TRUE |
| 2 | "db_RCV".RCV_buffer[1] | | 字符 | 'P' |
| 3 | "db_RCV".RCV_buffer[2] | | 字符 | 'C' |
| 4 | "db_RCV".RCV_buffer[3] | | 字符 | 'S' |
| 5 | "db_RCV".RCV_buffer[4] | | 字符 | 'E' |
| 6 | "db_RCV".RCV_buffer[5] | | 字符 | 'H' |
| 7 | "db_RCV".RCV_buffer[6] | | 字符 | 'D' |
| 8 | "db_RCV".RCV_buffer[7] | | 字符 | 'T' |
| 9 | "db_RCV".RCV_buffer[8] | | 字符 | 'O' |
| 10 | "db_RCV".RCV_buffer[9] | | 字符 | 'C' |
| 11 | "db_RCV".RCV_buffer[10] | | 字符 | 'P' |
| 12 | "db_RCV".RCV_buffer[11] | | 字符 | 'U' |
| 13 | "db_RCV".RCV_buffer[12] | | 字符 | 'I' |
| 14 | "db_RCV".RCV_buffer[13] | | 字符 | '2' |
| 15 | "db_RCV".RCV_buffer[14] | | 字符 | '1' |
| 16 | "db_RCV".RCV_buffer[15] | | 字符 | '4' |

图 8-31 监视值



8.3 工程案例：S7-1200 PLC 与 PAC3200 的 MODBUS 通信

8.3.1 案例介绍

西门子公司的 SENTRON PAC3200 多功能电力仪表是一种用于面板安装的仪表，可用来计量、显示配电系统多达 50 个测量变量，例如电压、电流、功率、有功功率、频率以及最大值、最小值和平均值。中文大屏幕图形液晶显示使用户可远距离读表。PAC3200 仪表如图 8-32 所示。

控制要求：设计 PAC3200 与 S7-1200 PLC 的通信系统。

8.3.2 指令介绍

1. PAC3200 与 S7-1200 PLC 的通信连接

PAC3200 多功能仪表的本体没有 MODBUS RTU 通信的功能，如果希望将 PAC3200 作为从站连接到 MODBUS RTU 网络与主站进行数据交换，必须选用外部扩展通信模块——SENTRONPAC RS-485 模块。如果 PAC RS-485 扩展模块使用错误的固件版本将不能工作，SENTRON PAC3200 电力监测设备的固件版本最低应为 FWV2.0X，因为较早的版本不支持 PAC RS-485 扩展模块。

该扩展模块具有下列性能特点：

- 1) 可通过设备正面设置参数；
- 2) 即插即用；
- 3) 支持 4.8/9.6/19.2 以及 38.4kbit/s 通信传输速率；
- 4) 通过 6 针螺钉端子接线；
- 5) 不需要外接辅助电源；
- 6) 通过模块上的 LED 显示状态。

PAC3200 MODBUS RTU 通信扩展模块如图 8-33 所示。

SENTRON PAC3200 MODBUS RTU 通信扩展模块的接线如图 8-34 所示。具体连接步骤如下：1) 将电缆连接到端子排上相应的螺栓端子；2) 将电缆屏蔽层的一端连接到保护性接地 PE；3) 将信号公共端连接到保护性接地，这样也使得扩展模块接地；4) 在第一个和最后一个通信节点上，在正信号和负信号之间接入总线端接电阻器。为此，PAC RS-485 扩展模块中集成了一个 120Ω 的总线端接电阻器。如果需要其他电阻值，请使用外部总线端接电阻器。将它连接到第一个和最后一个通信节点。

PAC3200 与 S7-1200 PLC 的 MODBUS 通信连接方式如图 8-35 和图 8-36 所示。例中使用的 PAC3200 仪表硬件为：1) PAC3200 (7KM2112-0BA00-3AA0)；2) MODBUS RTU 模块



图 8-32 PAC3200 仪表

第8章 S7-1200 PLC 的串口通信

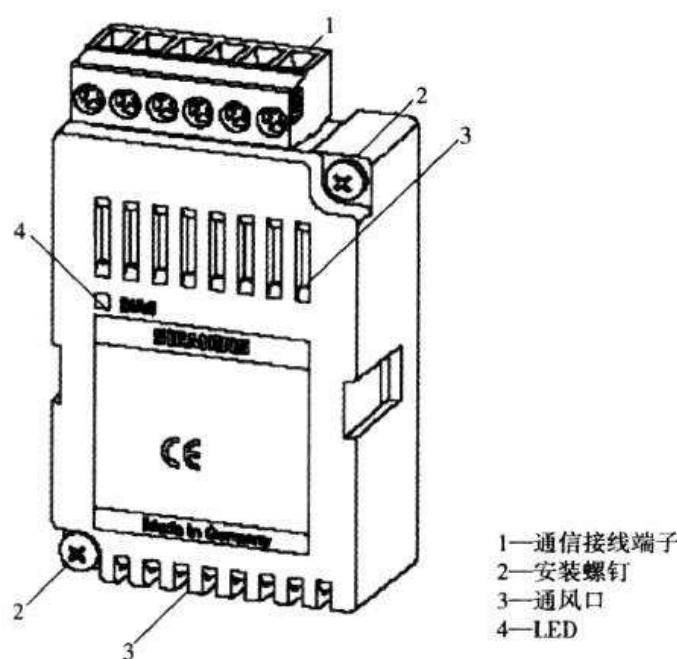


图 8-33 PAC3200 MODBUS RTU 通信扩展模块

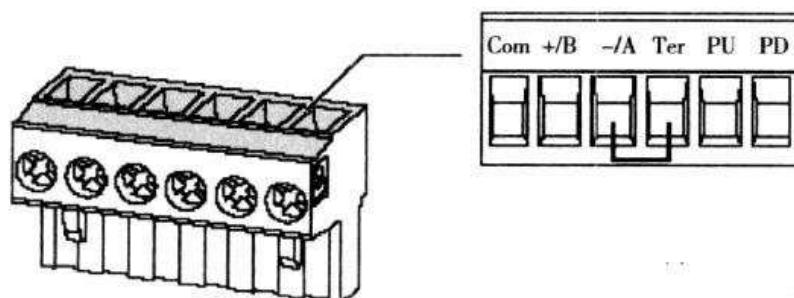


图 8-34 通信扩展模块的接线

(7KM9300-0AB00-0AA0); 3) MODBUS 通信电缆 (6XV1830-0EH10)。

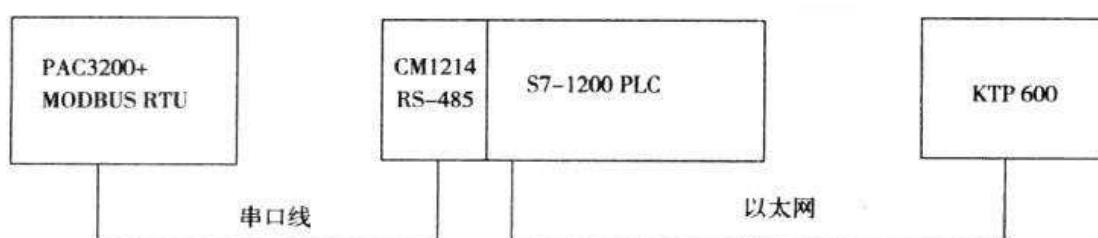


图 8-35 PAC3200 与 S7-1200 PLC 的硬件连线

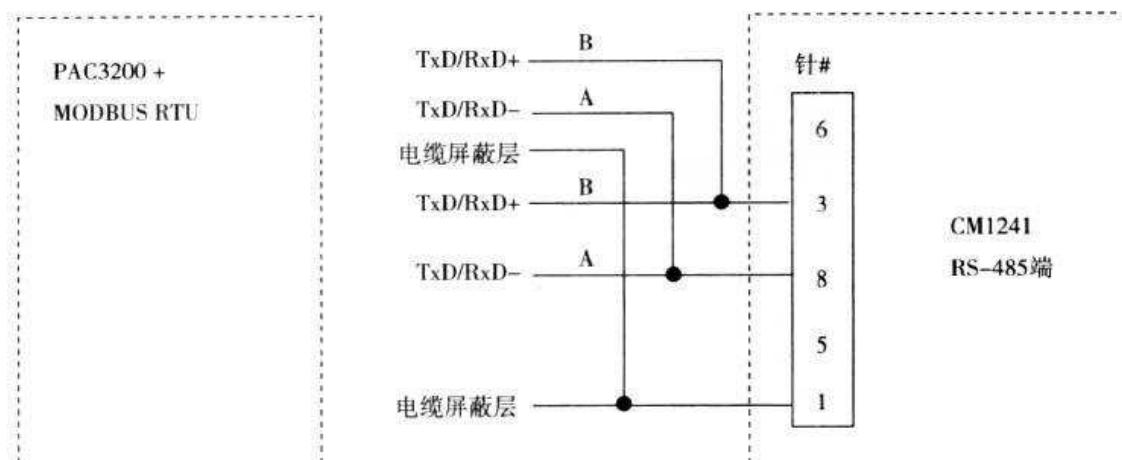


图 8-36 具体接线

2. SENTRON PAC3200 MODBUS RTU 通信的方式

SENTRON PAC3200 设备支持的功能码见表 8-5。其中，R—可读，RW—可读写。

表 8-5 SENTRON PAC3200 设备支持的功能码

| FC | 功 能 码 | 数 �据 类 型 | | 访 问 权 限 |
|----|---------|----------|----|---------|
| 02 | 输入的状态 | 位 | 输入 | R |
| 03 | 输出寄存器 | 寄存器 | 输出 | R |
| 04 | 输入寄存器 | 寄存器 | 输入 | R |
| 06 | 单一输出寄存器 | 寄存器 | 输出 | RW |
| 10 | 多个输出寄存器 | 寄存器 | — | RW |
| 2B | 设备识别 | — | — | R |

S7-1200 PLC 可以通过功能代码 0x03 和 0x04 访问仪表 PAC3200 的被测量数据。表 8-6 是一些 PAC3200 被测量的数据。

表 8-6 PAC3200 被测量的数据

| 偏 移 量 | 寄 存 器 数 | 名 称 | 格 式 | 单 位 | 数 值 范 围 | 访 问 权 限 |
|-------|---------|------------|-------|-----|---------|---------|
| 1 | 2 | a 相相电压 | Float | V | — | R |
| 3 | 2 | b 相相电压 | Float | V | — | R |
| 5 | 2 | c 相相电压 | Float | V | — | R |
| 7 | 2 | a-b 相间的线电压 | Float | V | — | R |
| 9 | 2 | b-c 相间的线电压 | Float | V | — | R |
| 11 | 2 | c-a 相间的线电压 | Float | V | — | R |
| 13 | 2 | a 相电流 | Float | A | — | R |
| 15 | 2 | b 相电流 | Float | A | — | R |
| 17 | 2 | c 相电流 | Float | A | — | R |
| 19 | 2 | a 相视在功率 | Float | VA | — | R |
| 21 | 2 | b 相视在功率 | Float | VA | — | R |



(续)

| 偏移量 | 寄存器数 | 名称 | 格式 | 单位 | 数值范围 | 访问权限 |
|-----|------|--------|-------|-----|------|------|
| 23 | 2 | c相视在功率 | Float | VA | — | R |
| 25 | 2 | a相有功功率 | Float | W | — | R |
| 27 | 2 | b相有功功率 | Float | W | — | R |
| 29 | 2 | c相有功功率 | Float | W | — | R |
| 31 | 2 | a相无功功率 | Float | Var | — | R |
| 33 | 2 | b相无功功率 | Float | Var | — | R |
| 35 | 2 | c相无功功率 | Float | Var | — | R |
| 37 | 2 | a相功率因数 | Float | — | 0…1 | R |
| 39 | 2 | b相功率因数 | Float | — | 0…1 | R |
| 41 | 2 | c相功率因数 | Float | — | 0…1 | R |

3. S7-1200 PLC MODBUS 的库指令

图 8-37 所示为全局库中的 MODBUS 指令，共有三条指令：MB_COMM_LOAD、MB_MASTER、MB_SLAVE。

MB_COMM_LOAD 指令用于组态 CM 模块上的端口，以进行 MODBUS RTU 协议通信。可以使用 RS-232 或 RS-485 CM 模块。用户程序必须先执行 MB_COMM_LOAD 来组态端口，之后 MB_SLAVE 或 MB_MASTER 指令才能与该端口进行通信。

MB_MASTER 指令允许用户程序作为 MODBUS 主站进行通信。可访问一个或多个 MODBUS 从站设备中的数据。插入 MB_MASTER 指令可创建背景数据块。使用此 DB 名称作为 MB_COMM_LOAD 指令中的 MB_DB 参数。从同一个 OB（或 OB 优先等级）执行指定端口的所有 MB_MASTER 指令。

MB_SLAVE 指令允许用户程序作为 MODBUS 从站进行通信。MODBUS RTU 主站可以发出请求，然后程序通过执行 MB_SLAVE 来响应。插入 MB_SLAVE 指令可创建背景数据块。使用此 DB 名称作为 MB_COMM_LOAD 指令中的 MB_DB 参数。从循环中断 OB 执行所有的 MB_SLAVE 指令。

MODBUS 指令不使用通信中断事件来控制通信过程。用户程序必须轮询 MB_MASTER 或 MB_SLAVE 指令以了解传送和接收的完成情况。如果某个端口作为从站响应 MODBUS 主站，则 MB_MASTER 无法使用该端口。对于给定端口，只能使用一个 MB_SLAVE 执行实例。同样，如果要将某个端口用于初始化 MODBUS 主站的请求，则 MB_SLAVE 将不能使用该端口。MB_MASTER 执行的一个或多个实例可使用该端口。

如果用户程序操作 MODBUS 从站，则对 MB_SLAVE 的轮询（周期性执行）速度有要求，即必须使该指令能及时响应来自 MODBUS 主站的进入请求。如果用户程序操作 MODBUS 主站并使用 MB_MASTER 向从站发送请求，则用户必须继续轮询（执行 MB_MASTER）直到返回从站的响应。



图 8-37 全局库中的 MODBUS 指令



4. S7-1200 PLC MODBUS RTU 的通信方式

S7-1200 PLC 作为 MODBUS RTU 主站的通信方式是由 DATA _ ADDR 和 MODE 参数来选择 MODBUS 功能类型。DATA _ ADDR (从站中的起始 MODBUS 地址)：指定要在 MODBUS 从站中访问的数据的起始地址。MB _ MASTER 使用 MODE 输入而非功能代码输入。MODE 和 MODBUS 地址范围一起确定实际 MODBUS 消息中使用的功能代码。

表 8-7 列出了 MB _ MASTER MODBUS 功能。

表 8-7 MB _ MASTER MODBUS 功能

| MB _ MASTER MODBUS 功能 | | | | |
|--|----------------------------|-------|------------------------------|-----------|
| | MODBUS 地址参数 DATA _ ADDR | 地址类型 | MODBUS 数据长度 参数 DATA _ LEN | MODBUS 功能 |
| 模式 0 | | | | |
| 读取 | 00001 ~ 09999 | 输出位 | 1 ~ 2000 | 01H |
| | 10001 ~ 19999 | 输入位 | 1 ~ 2000 | 02H |
| | 30001 ~ 39999 | 输入寄存器 | 1 ~ 125 | 04H |
| | 40001 ~ 49999 | 保持寄存器 | 1 ~ 125 | 03H |
| | 400001 ~ 465536 (扩展) | | | |
| 模式 1 | | | | |
| 写入 | 00001 ~ 09999 | 输出位 | 1 (单个位) | 05H |
| | 40001 ~ 49999 | 保持寄存器 | 1 (单个字) | 06H |
| | 400001 ~ 465536 (扩展) | | | |
| | 00001 ~ 09999 | 输出位 | 2 ~ 1968 | 15H |
| | 40001 ~ 49999 | 保持寄存器 | 2 ~ 123 | 16H |
| 模式 2 | | | | |
| 有些 MODBUS 从站下支持使用 MODBUS 功能 05H 和 06H 写入单个位或字。在这些情况下，可通过模式 2 强制使用 MODBUS 函数 15H 和 16H 写入单个位和字 | | | | |
| 写入 | 00001 ~ 09999 | 输出位 | 1 ~ 1968 | 15H |
| | 40001 ~ 49999 | 保持寄存器 | 1 ~ 123 | 16H |
| | 400001 ~ 465536 (扩展) | | | |
| 模式 11 | | | | |
| <ul style="list-style-type: none"> • 从 MB _ ADDR 输入引用的 MODBUS 从站中读取事件计数器字 • 在 Siemens S7-1200 PLC MODBUS 从站中，从站每次从 MODBUS 主站收到一个有效的读或写请求（非广播）时，该计数器就会递增 • 返回的值存储在 DATA _ PTR 输入指定的字位置 • 此模式不需要有效的 DATA _ LEN | | | | |
| 模式 80 | | | | |
| <ul style="list-style-type: none"> • 检查 MB _ ADDR 输入引用的 MODBUS 从站的通信状态 • MB _ MASTER 指令的 NDR 输出位置位时，表示所寻址的 MODBUS 从站使用适当的响应数据进行了响应 • 没有数据返回到用户程序中 • 此模式不需要有效的 DATA _ LEN | | | | |
| 模式 81 | | | | |
| <ul style="list-style-type: none"> • 重置 MB _ ADDR 输入引用的 MODBUS 从站中的事件计数器（即模式 11 返回的值） • MB _ MASTER 指令的 NDR 输出位置位时，表示所寻址的 MODBUS 从站使用适当的响应数据进行了响应 • 没有数据返回到用户程序中 • 此模式不需要有效的 DATA _ LEN | | | | |



8.3.3 S7-1200 PLC 与 PAC3200 进行 MODBUS RTU 的通信组态

1. S7-1200 PLC 的硬件组态

在硬件配置中，添加 CPU1214C 和通信模块 CM1241 RS-485，如图 8-38 所示。

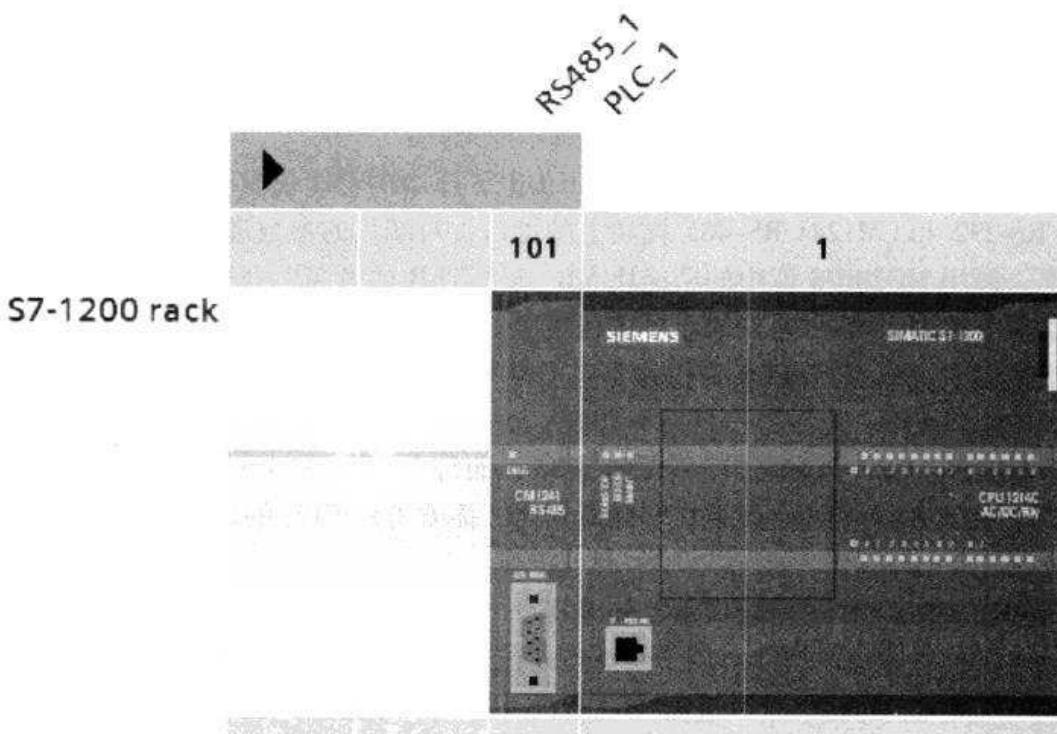


图 8-38 添加设备

2. PAC3200 参数设置

在 SENTRON PAC 电力监测设备的主菜单中，调用“设置”→“RS485 模块”，出现下面的设置画面（见图 8-39）。

- 1) 地址的设置范围：1~247。本例中设为 8。
- 2) 波特率的设置范围：4800, 9600, 19200, 38400。本例中设为 38400。
- 3) 设置外部通信的数据位、奇偶校验位及停止位：
 - ① 8E1 = 8 个数据位，奇偶校验位为 even, 1 个停止位；
 - ② 8O1 = 8 个数据位，奇偶校验位为 odd, 1 个停止位；
 - ③ 8N2 = 8 个数据位，无奇偶校验位，两个停止位；
 - ④ 8N1 = 8 个数据位，无奇偶校验位，1 个停止位。

本例根据 S7-1200 PLC MODBUS MASTER 的参数设置为 8N1。

- 4) 协议的设置：可选项为 SEABUS, MODBUS RTU。

本例中设为 MODBUS RTU。

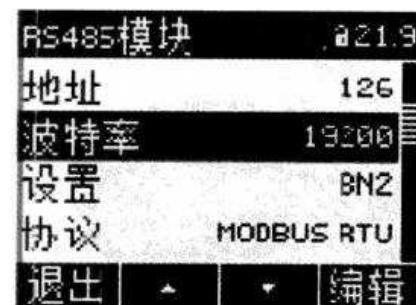


图 8-39 PAC3200 MODBUS RTU
通信参数的设置



5) 响应时间的设置：注意与波特率的设置相匹配，本例中设为 10ms。

8.3.4 PLC 的通信编程

1. RTU 通信原理

西门子公司的 S7-1200 PLC 的模块 CM1241 RS-232 和 CM1241 RS-485 都可以实现 MODBUS RTU 的通信，本例中采用 CM1241 RS-485 模块来实现与仪表 PAC3200 的 MODBUS RTU 通信。

S7-1200 PLC 的 MODBUS RTU 通信的基本原理是：

首先，S7-1200 PLC 的程序调用一次 MODBUS 库中的功能块 MB_COMM_LOAD 来组态 CM1241 RS-232 和 CM1241 RS-485 模块上的端口，对端口的参数进行配置。

其次，调用 MODBUS 库中的功能块 MB_MASTER 或者 MB_SLAVE 作为 MODBUS 主站或者从站与支持 MODBUS 协议的设备进行通信。

在本案例中，S7-1200 PLC 是作为 MODUBUS 主站与 PAC3200 进行 MODBUS RTU 通信控制的。

每个 S7-1200 CPU 最多可带 3 个通信模块，而每个 CM1241 RS-485 通信模块理论上最多支持 247 个 MODBUS 子站。但是在实际应用时，需要考虑 CPU 的性能以及轮询 MODBUS 子站的时间。

2. MODBUS RTU 通信接口参数的编程

MB_COMM_LOAD 功能块用于组态点对点 (Point-to-Point, PtP) CM1241 RS-485 或 CM1241 RS-232 模块上的端口，以进行 MODBUS RTU 协议通信。程序开始运行时，调用一次 MB_COMM_LOAD 功能块来实现对 MODBUS RTU 模块的初始化组态。MB_COMM_LOAD 执行一次的编程时钟位的设置如图 8-40 所示采用时钟位 M10.0 来完成。

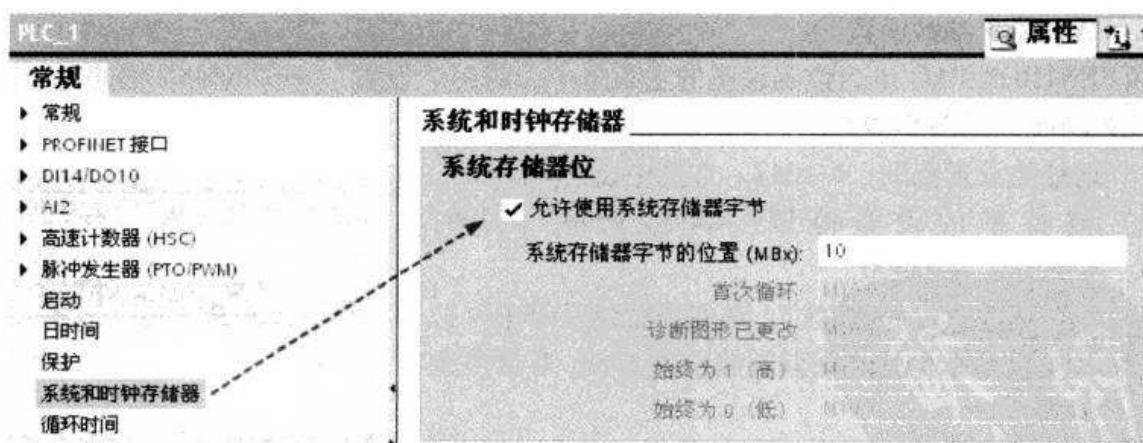


图 8-40 MB_COMM_LOAD 执行一次的编程时钟位的设置

MB_COMM_LOAD 功能块的编程如图 8-41 所示。

PORT：通过哪个通信模块进行 MODBUS RTU 通信。

BAUD：和 MODBUS 子站进行通信的速率。

通信端口的波特率的取值范围为 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200。



▼ Network 1:

Comment

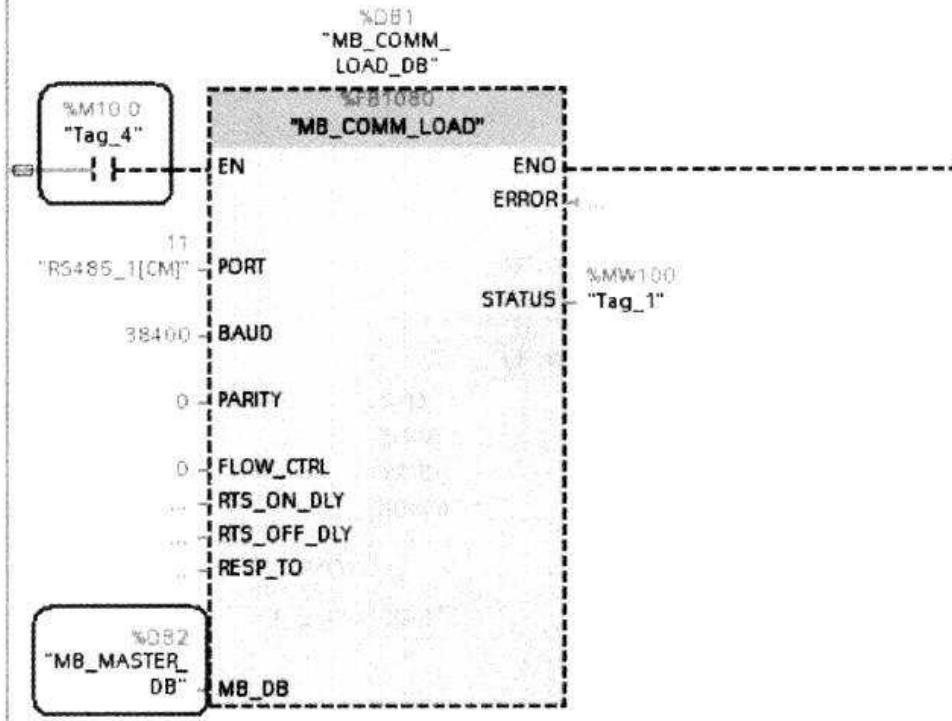


图 8-41 MB_COMM_LOAD 功能块的编程

注意：仪表 PAC3200 的波特率的设置范围是 4800，9600，19200，38400。因此，S7-1200 PLC 的波特率的设置一定要和仪表 PAC3200 的波特率的设置相一致。

MB_DB：对 MB_MASTER 或 MB_SLAVE 指令所使用的背景数据块的引用。在用户程序中放置 MB_SLAVE 或 MB_MASTER 后，DB 标识符会出现在 MB_DB 功能框连接的助手下拉列表中，如“MB_MASTER_DB”或“MB_SLAVE_DB”。

STATUS：端口状态代码。具体含义见表 8-8。

表 8-8 MB_COMM_LOAD 组态端口的状态代码

| STATUS 值 (W#16#…) | 说 明 |
|-------------------|--|
| 0000 | 无错误 |
| 8180 | 端口 ID 值无效 |
| 8181 | 波特率值无效 |
| 8182 | 奇偶校验值无效 |
| 8183 | 流控制值无效 |
| 8184 | 响应超时值无效 |
| 8185 | 指向 MB_MASTER 或 MB_SLAVE 的背景数据块的 MB_DS 指针错误 |



3. MODBUS_MASTER 功能块的编程

MB_MASTER 功能块允许程序作为 MODBUS 主站使用点对点 CM1241 RS-485 或 CM1241 RS-232 模块上的端口进行通信。可访问一个或多个 MODBUS 从站设备中的数据。

MB_MASTER 功能块的编程如图 8-42 所示。

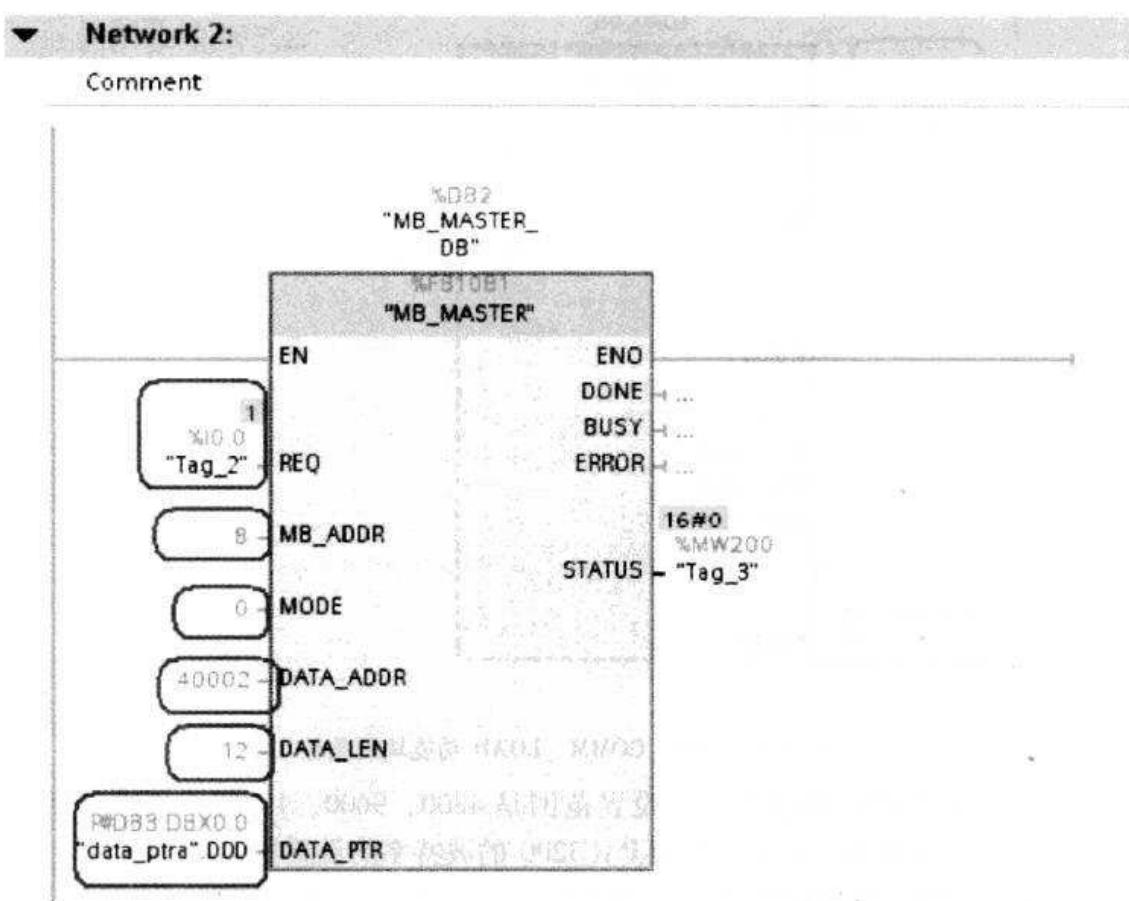


图 8-42 MB_MASTER 功能块的编程

REQ：数据发送请求信号。0：无请求；1：请求将数据传送到 MODBUS 从站。

MB_ADDR：通信对象 MODBUS 从站的地址。有效地址范围为 0 ~ 247。值 0 被保留用于将消息广播到所有 MODBUS 从站。只有 MODBUS 功能代码 05、06、15 和 16 是可用于广播的功能代码。注意：此处 MODBUS 从站的地址一定要与仪表 PAC3200 的 MODBUS 地址相一致。

MODE：模式选择。选择范围为：读、写、诊断。

DATA_ADDR：从站中的起始地址，指定要在 MODBUS 从站中访问的数据的起始地址。

特别需要注意的是：由于仪表 PAC3200 的寄存器与 S7-1200 PLC MODBUS RTU 寄存器的不一致性，读取仪表 PAC3200 的 DATA_ADDR 的地址必须从 40002 开始。

S7-1200 PLC 的 MODBUS RTU 通信功能是通过使用“DATA_ADDR”和“MODE”的组合来选择 MODBUS 功能码，ST_1200 PLC 的 MODBUS RTU 通信功能码见表 8-9。

第8章 S7-1200 PLC 的串口通信



表 8-9 S7-1200 PLC 的 MODBUS RTU 通信功能码

| MB_MASTER MODBUS 功能 | | | | |
|---------------------|---------------------------------------|-------|----------------------------|-----------|
| | MODBUS 地址参数 DATA_ADDR | 地址类型 | MODBUS 数据长度 参数 DATA_LEN | MODBUS 功能 |
| 模式 0 | | | | |
| 读取 | 00001 ~ 09999 | 输出位 | 1 ~ 2000 | 01H |
| | 10001 ~ 19999 | 输入位 | 1 ~ 2000 | 02H |
| | 30001 ~ 39999 | 输入寄存器 | 1 ~ 125 | 04H |
| | 40001 ~ 49999 400001 ~ 465536 (扩展) | 保持寄存器 | 1 ~ 125 | 03H |
| 模式 1 | | | | |
| 写入 | 00001 ~ 09999 | 输出位 | 1 (单个位) | 05H |
| | 40001 ~ 49999 400001 ~ 465536 (扩展) | 保持寄存器 | 1 (单个字) | 06H |

而仪表 PAC3200 的 MODBUS RTU 通信功能则是通过功能码来实现的，见表 8-10。

表 8-10 仪表 PAC3200 的 MODBUS RTU 通信功能码

| FC | 功 能 码 | 数据 类型 | | 访问 权限 |
|----|---------|-------|----|-------|
| 02 | 输入的状态 | 位 | 输入 | R |
| 03 | 输出寄存器 | 寄存器 | 输出 | R |
| 04 | 输入寄存器 | 寄存器 | 输入 | R |
| 06 | 单一输出寄存器 | 寄存器 | 输入 | RW |
| 10 | 多个输出寄存器 | 寄存器 | — | RW |
| 2B | 设备识别 | — | — | R |

因此可以得出，如果需要读取输出寄存器的值，需要使用模式 0 的 03H 功能，即从寄存器 40001 ~ 49999 来读取仪表的数据，但是由于仪表 PAC3200 的寄存器与 S7-1200 PLC-MODBUS RTU 寄存器的不一致性，读取仪表 PAC3200 的 DATA_ADDR 的地址必须从 40002 开始。

DATA_LEN：请求访问数据的长度、位数或字节数。

DATA_PTR：数据指针，指向要写入或读取的数据的 CPU DB 地址。该 DB 必须为“非仅符号访问”DB 类型。

NDR：新数据就绪：

- 0-事务未完成；
- 1-表示 MB_MASTER 指令已完成所请求的有关 MODBUS 从站的事务。

BUSY：忙：

- 0-无正再进行的 MB_MASTER 事务；
- 1-MB_MASTER 事务正在进行。

ERROR：错误：



- 0-未检测到错误；
- 1-表示检测到错误并且参数 STATUS 提供的错误代码有效。

STATUS：状态代码，见表 8-11。

表 8-11 MB_MASTER 进行 MODBUS RTU 通信的状态代码

| STATUS 值 (W#16#…) | 说 明 |
|-------------------|--|
| 0000 | 无错误 |
| 80C8 | 指定的响应超时时间（指 RCVTIME 或 MSGTIME）为 0 |
| 80D1 | 接收方发出了暂停主动传输的流控制请求并且在指定的等待时间内未重新激活该传输 在硬件流控制期间，如果接收方在指定的等待时间内没有声明 CTS，也会产生该错误 |
| 80D2 | 传送请求中止，因为没有从 DCE 收到任何 DSR 信号 |
| 80E0 | 因接收缓冲区已满，消息被终止 |
| 80E1 | 因出现奇偶校验错误，消息被终止 |
| 80E2 | 因组帧错误，消息被终止 |
| 80E3 | 因出现超限错误，消息被终止 |
| 80E4 | 因指定长度超出总缓冲区大小，消息被终止 |
| 8180 | 端口 ID 值无效 |
| 8186 | MODBUS 站地址无效 |
| 8188 | 模式值无效或只读从站地址区的写模式无效 |
| 8189 | 数据地址值无效 |
| 818A | 数据长度值无效 |
| 818B | 指向本地数据源/目标的指针无效：大小不正确 |
| 818C | 指向安全 DB 类型的 DATA_PTR（必须为典型 DB 类型）的指针 |
| 8200 | 端口正忙于处理传送请求 |

8.3.5 测试与总结

在成功地编译下载到在 S7-1200 PLC 中后，可以从变量表中看到仪表 PAC3200 的三相相电压数据，如图 8-43 所示。

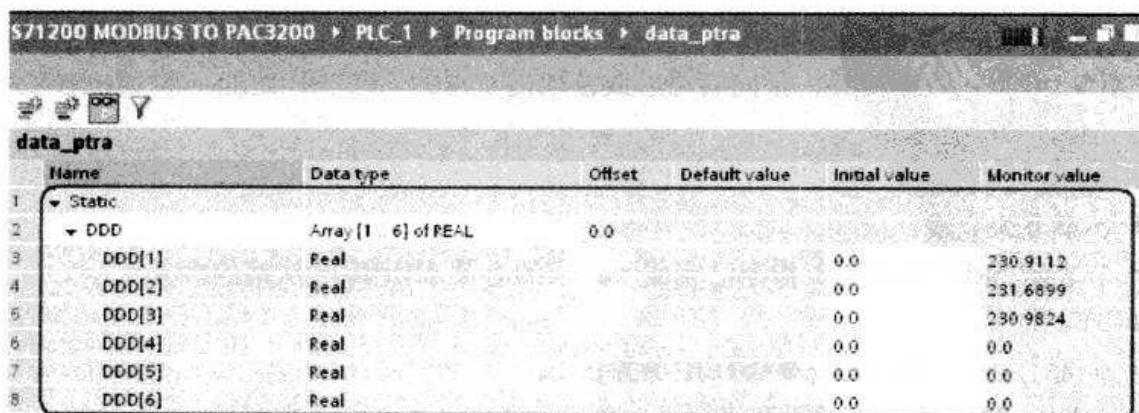


图 8-43 在 S7-1200 PLC 中通过 MODBUS 通信得到 PAC3200 的三相相电压数据



通过本案例可以知道，MODBUS 主站通信必须遵循以下规则：

- 1) 必须先执行 MB_COMM_LOAD 组态端口，然后 MB_MASTER 指令才能与该端口通信。
- 2) 如果要将某个端口用于初始化 MODBUS 主站的请求，则 MB_SLAVE 将不能使用该端口。MB_MASTER 执行的一个或多个实例可使用该端口。
- 3) MODBUS 指令不使用通信中断事件来控制通信过程。用户程序必须轮询 MB_MASTER 指令以了解传送和接收的完成情况。
- 4) 如果用户程序操作 MODBUS 主站并使用 MB_MASTER 向从站发送请求，则用户必须继续轮询（执行 MB_MASTER）直到返回从站的响应。
- 5) 从同一个 OB（或 OB 优先等级）执行指定端口的所有 MB_MASTER 指令。

S7-300 PLC 基本机架

：測量不同帶頭數的齒輪級數。MOTOR。直聯皮帶與皮帶拉緊
：口輪轉子旋長令游標卡尺。游標卡尺：WESTER_EASY_MM_GEAR_MM_游標卡尺（1）

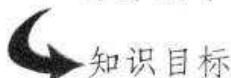
游標卡尺不拆 EASY_MM 標。米游標卡尺 2000S 游標卡尺頭部某處要緊固。（2）
：口輪轉子測量皮帶輪外緣尺寸。游標卡尺 WESTER_MM。口輪
：2MM_MM 游標卡尺測量皮帶輪外緣尺寸。唇游標卡尺游標卡尺頭部某處要緊固不令游標卡尺
：游標卡尺的游標亦要拆下以令游標卡尺（3）

：各齒輪頭部拆去皮帶。游標卡尺 WESTER_MM 游標卡尺頭部某處要緊固皮帶頭部（4）
：測量游標卡尺頭部某處（WESTER_MM 游標卡尺）頭部要拆下
：令游標卡尺測量皮帶輪外緣尺寸。游標卡尺游標卡尺頭部某處要緊固皮帶頭部（5）

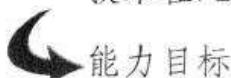
第9章 S7-1200 PLC的 OPC组态

OPC是随过程控制的发展产生的一种工业标准，它既可以用于将现场PLC设备的数据采集到DCS或SCADA中来，也可以用于将数据从一种DCS或SCADA系统采集到另一种DCS系统、SCADA系统或其他上位应用程序中。在OPC的客户端/服务器结构中，同一个程序可以访问多个服务器，同一个服务器也可以同时为多个客户端服务。西门子公司的S7-1200 PLC作为优秀的下位机PLC，它可以通过OPC技术与WinCC、IFIX等优秀组态软件进行数据通信。

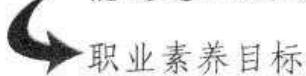
本章的学习目标如下：



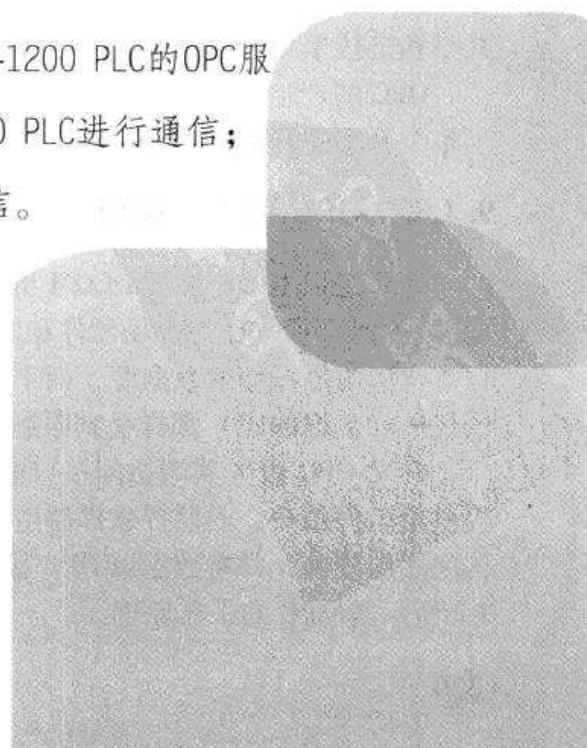
了解OPC的产生及工作原理；熟悉SIMATIC OPC服务器的工作原理；掌握OPC技术在PLC与上位机数据交换中的特点。



能通过STEP 7和SIMATIC OPC软件来构建S7-1200 PLC的OPC服务器功能；能通过WinCC组态软件与S7-1200 PLC进行通信；能通过IFIX组态软件与S7-1200 PLC进行通信。



通过了解工控软件的应用范围来掌握新技术对自动化专业所提出的新要求。





9.1 背景知识介绍：OPC 标准及其应用

9.1.1 OPC 的产生

现阶段的 SCADA 软件（即数据采集与监视控制系统）已经由集中控制逐步转换成集散式控制，所以监控系统、人机接口等应用程序需要通过硬件驱动程序的方式和现场的硬件设备通信（见图 9-1）。

驱动程序一般是由监控软件的开发商根据硬件厂商的产品来制作的，不同的厂商，不同的设备需要不同的驱动程序，同一个硬件设备对于不同的监控软件也需要不同的驱动程序，这样对软件监控工程的顺利完成造成了很多麻烦，使现场硬件设备、远程监控软件和驱动程序的兼容成为一个棘手的问题。虽然现场总线协议的产生解决了一部分问题，但是并不是所有的现场设备都适用现场总线协议，就算是勉强兼容也不能 100% 的发挥远程监控软件对现场设备的控制功能。

由于实际需要，软件制造商和系统集成商以及硬件开发商急切需要一种高效、可靠、开放、互操作性强的驱动程序模式，于是 OPC 标准产生了。OPC 全称为 OLE for Process Control，是随过程控制的发展产生的一种工业标准。而 OLE 技术，则是 Object Linking and Embedding（对象连接与嵌入）的简称，它不仅是桌面应用程序集成，而且还定义和实现了一种允许应用程序作为软件“对象”（数据集合和操作数据的函数）彼此进行“连接”的机制，这种连接机制和协议称为部件对象模型。OLE 是在客户应用程序间传输和共享信息的一组综合标准，它允许创建带有指向应用程序的链接的混合文档以使用户修改时不必在应用程序间切换。OLE 基于组件对象模型（COM）并允许开发可在多个应用程序间互操作的可复用即插即用对象，该协议可广泛用于电子表格和其他应用程序，可以通过客户/服务器体系共享和链接单独的信息。

OPC 的产生解决了硬件、软件厂商的矛盾，满足了系统集成中无缝访问的要求，提高了整个系统的开放性和互交换的操作性，使组件多样化的集散式监控系统更为简单。

9.1.2 OPC 的工作原理及规范

OPC 是以 OLE/COM/DCOM 机制作为应用程序级的通信标准，采用 Client/Server（客户机/服务器）模式作为现场硬件和监控软件之间的一个中间环节，使不同的客户端软件可以任意访问服务器端的数据源，而不会像 DLL 驱动程序（即一个包含可由多个程序同时使用的代码和数据的库）那样受到限制。

开发 OPC 服务器端访问接口的任务由硬件生产厂家来完成，以 OPC Server 软件的形式提供给软件用户，而监控软件制造商只要在软件中加入 OPC Client 驱动就可以。OPC 服务器访问底层硬件，并将数据以 OPC 接口的方式提供到任何具有 OPC 客户端接口的软件。如图 9-2 所示为 OPC 的工作原理。

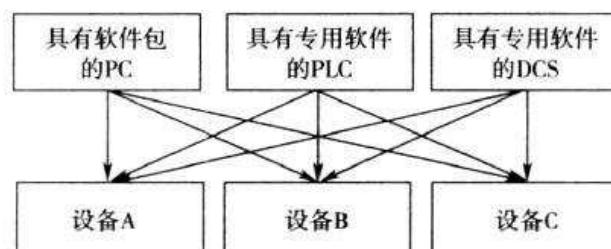


图 9-1 通过硬件驱动程序进行通信

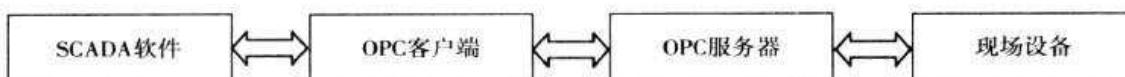


图 9-2 OPC 的工作原理

OPC 规范要求硬件生产商在提供新的硬件设备的同时也提供一款根据新设备的协议开发的可以和新设备通信的符合 OPC 标准的服务器端软件，为设备提供基本的协议接口，并具有 OLE/COM 连接，一般用来完成数据交互和数据前置处理。这样任何标准的 OPC 客户端都可以以标准方式访问该 OPC 服务器。

OPC 规范还要求监控软件制造商必须提供符合 OPC 标准的客户端驱动，客户端驱动负责按标准的 OPC 接口函数对 OPC 服务器进行数据访问，这样不管现场设备是哪种形式，客户端都可以用统一的方式去访问，方便了硬件和监控软件数据间的交互。当 OPC 服务器里的数据发生改变时，服务器端将向客户端发出信息，客户端得到信息后再进行必要的处理；同样当客户端数据改变时，客户向服务器端发出信息，服务器也会相应的处理数据，这样就不需要花费大量时间来互相查询数据。

9.1.3 OPC 的服务器与客户机

OPC 技术对于工业控制系统的集成是革命性的，使不同的现场设备和系统软件都可以互相兼容。OPC 作为一种连接件，按标准的 COM 对象、方法属性，提供了更为简单的数据交互。无论是公司管理系统还是现场监控系统，无论是 DCS 还是 FCS，无论是 PLC 还是智能仪表，都可以通过 OPC 快速交换数据。简单来说，OPC 技术增强了整个监控系统的功能，提高了系统内各个组件的兼容性。

OPC 服务器和客户机的概念与超级市场相似，是存放各种供选择商品的通道代表服务器。

供选择的商品构成服务器读取和写入的所有进程数据位置。客户机就如同沿着通道移动并选择需要的物品的购物车。

OPC 数据项是 OPC 服务器与数据来源的连接。所有与 OPC 数据项的读写、存取均通过包含 OPC 项目的 OPC 群组目标进行。同一个 OPC 项目可包含在几个群组中。当某个变量被查询时，对应的数值会从最新进程数据中获取并被返回，这些数值可以是传感器、控制参数、状态信息或网络连接状态的数值。OPC 的结构由 3 类对象组成：服务器、组和数据项。

提供数据的 OPC 元件被称为 OPC 服务器。OPC 服务器向下对设备数据进行采集，向上与 OPC 客户应用程序通信完成数据交换。使用 OPC 服务器作为数据源的 OPC 元件称为 OPC 客户端。

如图 9-3 所示为 OPC 客户机和服务 器组成系统，它具有如下特点：

- 1) 任何客户机都可以与服务器连接；
- 2) 接纳所有类型的客户机及服务 器；
- 3) 与 DDE 技术相比，OPC 能够



图 9-3 OPC 客户机和服务器组成系统



快速、可靠的传输数据；

- 4) 可以支持所有编程语言（C++，VB，Java）；
- 5) 提供了 XML 语言和 Web Service，支持所有平台分布式环境下软件部件的交互使用和分享数据。

9.1.4 OPC 数据访问

OPC 服务器支持两种类型的数据读取：同步读写（Synchronous read/write）和异步读写（Asynchronous read/write）。同步读写就是指 OPC 的客户端向服务器发出一个读/写请求，然后不再继续执行，一直等待，直到收到服务器发给客户机的返回值，OPC 客户端才会继续执行下去。而异步读写则是 OPC 的客户端向服务器发出一个读/写请求，在等待返回值的过程中，可以继续执行下面的程序，直到服务器数据准备好后，向客户机发出一个返回值，在回调函数中客户端处理返回数值，然后结束此次读/写过程。

同步读/写数据存取速度快，编程简单，无需回调，但需要等待返回结果。异步读写不需等待返回值，可以同时处理多个请求。

9.2 技能训练一：使用 SIMATIC NET 建立 OPC 数据链

9.2.1 SIMATIC NET 软件介绍

目前，西门子公司的上位计算机组态软件 WinCC 等均不能直接访问 S7-1200 PLC，需要用软件 SIMATIC NET 的 OPC 功能来解决这一问题。为了实现 OPC 通信，需要安装西门子公司的通信软件 SIMATIC NET。

SIMATIC NET 是西门子公司在工业控制层面上提供的一个开放的、多元的通信系统。它意味着可以将工业现场的 PLC、主机、工作站和个人电脑联网通信，为了适应自动化工程中的种类多样性，SIMATIC NET 推出了多种不同的通信网络以因地制宜，这些通信网络符合德国或国际标准，他们包括工业以太网、PROFIBUS、AS-I、MPI。

SIMATIC NET 系统包括：1) 传输介质、网络配件和相应的传输设备及传输技术；2) 数据传输的协议和服务；3) 连接 PLC 和电脑到 LAN 网上的通信处理器（CP 模块）。

高级 PC Station 组态（即 Advanced PC Configuration）是随 SIMATIC NET V6.0 以上提供的，它代表一个 PC 站的全新、简单、一致和经济的调试和诊断解决方案，一台 PC 可以和 PLC 一样，在 SIMATIC S7 中进行组态，并通过网络装入。PC Station 包含了 SIMATIC NET 通信模块和应用软件，SIMATIC NET OPC server 就是允许和其他应用通信的一个典型应用软件。

9.2.2 S7-1200 PLC 与 SIMATIC NET 通信的建立

1. 硬件与软件要求

如图 9-4 所示为 S7-1200 PLC 与 SIMATIC NET 建立通信的示意图，它包括 S7-1200 CPU、含以太网卡的 PC 和以太网线硬件设备，也包括预装了 STEP 7 Basic V10.5、STEP 7 V5.4 和 SIMATIC NET V7.1 系统的软件部分。

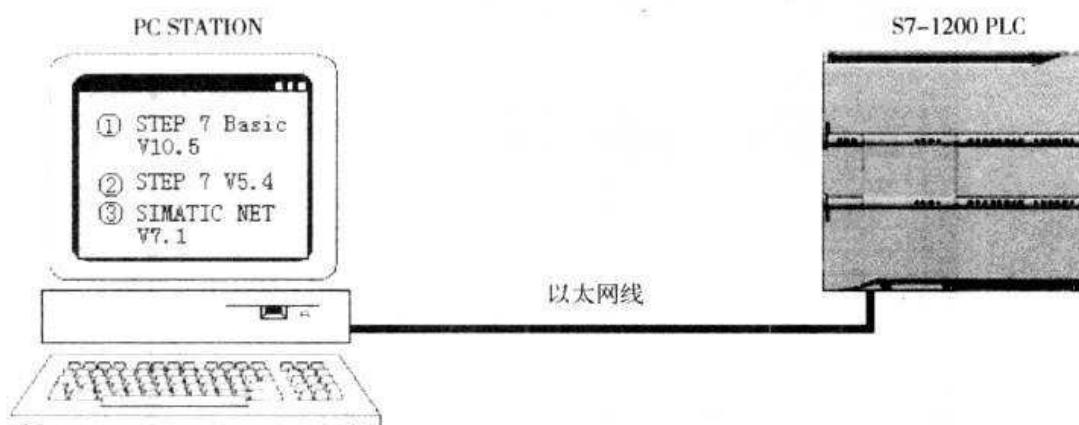


图 9-4 S7-1200 PLC 与 SIMATIC NET 建立通信

2. 在 STEP 7 中组态 PC Station

在 STEP 7 中组态 PC Station 的操作步骤如下：

1) 在 STEP7 中新建项目，组态 PC Station。

打开 STEP7 软件，并新建一个项目：“S7-1200 _ OPC”，通过“Insert”→“Station”→“SIMATIC PC Station”插入一个 PC 站，PC 站的名字为：“SIMATIC PC Station (1)”，如图 9-5 所示。需要注意的是，STEP7 中 PC Station 的名字“SIMATIC PC Station (1)”要与 SIMATIC NET 中“Station Configuration Editor”的“Station Name”完全一致，才能保证下载成功。

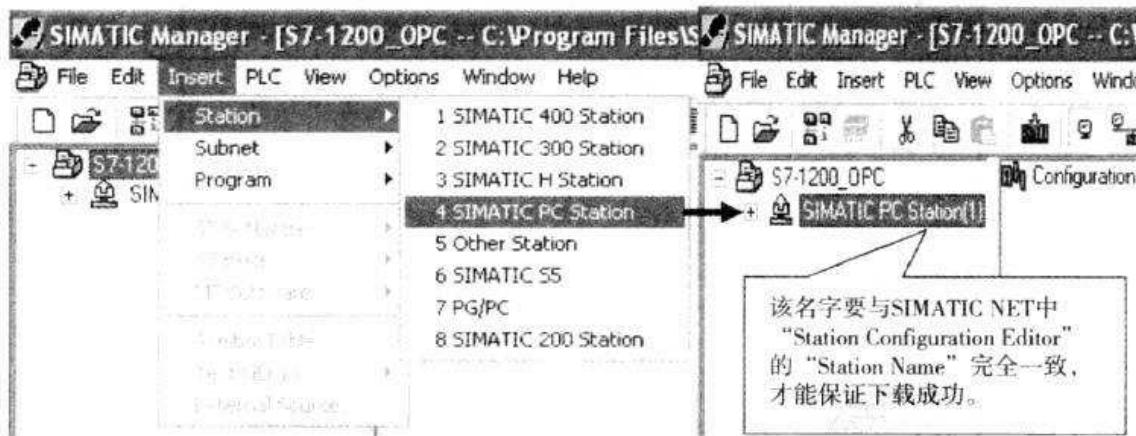


图 9-5 插入并组态 PC 站

2) 双击 Configuration 即可进入 PC Station 硬件组态界面。

在第一个槽中，从“SIMATIC PC Station”→“User Application”→“OPC Server”选择版本“SW V6.2 SP1”添加一个 OPC Sever 的应用。在第三个槽中，从“SIMATIC PC Station”→“CP Industrial Ethernet”→“IEGeneral”选择版本“SW V6.2 SP1”添加一个 IE General，并设置 IP 地址，如图 9-6 所示。因为使用的是普通以太网卡，所以要选择添加“IE General”。

然后配置网卡地址并新建一个以太网，如图 9-7 所示。

完成 PC 站硬件组件设置后，按下编译存盘按钮，确定且存储当前组态配置。



图解西门子 S7-1200 PLC 入门到实践

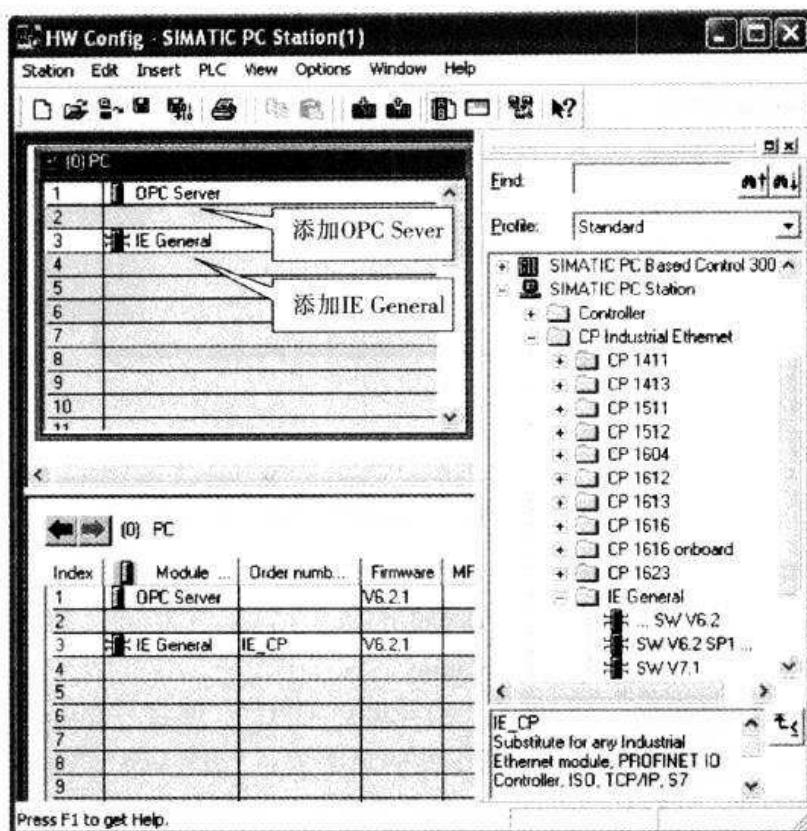


图 9-6 PC Station 的硬件组态

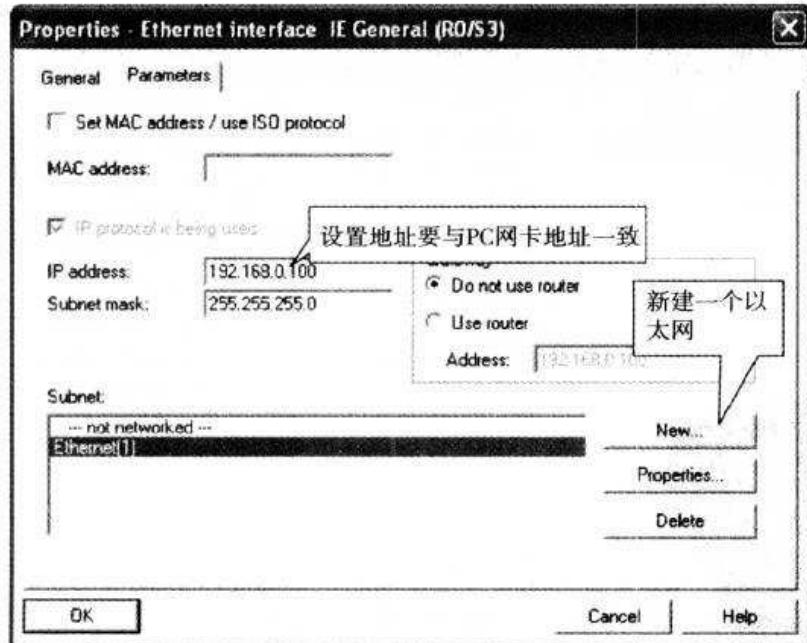


图 9-7 设置以太网地址

3) 配置网络连接。

通过点击工具栏右上角网络配置的图标 进入网络配置，然后在 NetPro 网络配置中，用鼠标选择 OPC Server 后，在连接表第一行鼠标右键插入一个新的连接或通过“Insert”→“New Connection”也可建立一个新连接然后定义连接属性，如图 9-8、图 9-9 所示。

第9章 S7-1200 PLC 的 OPC 组态

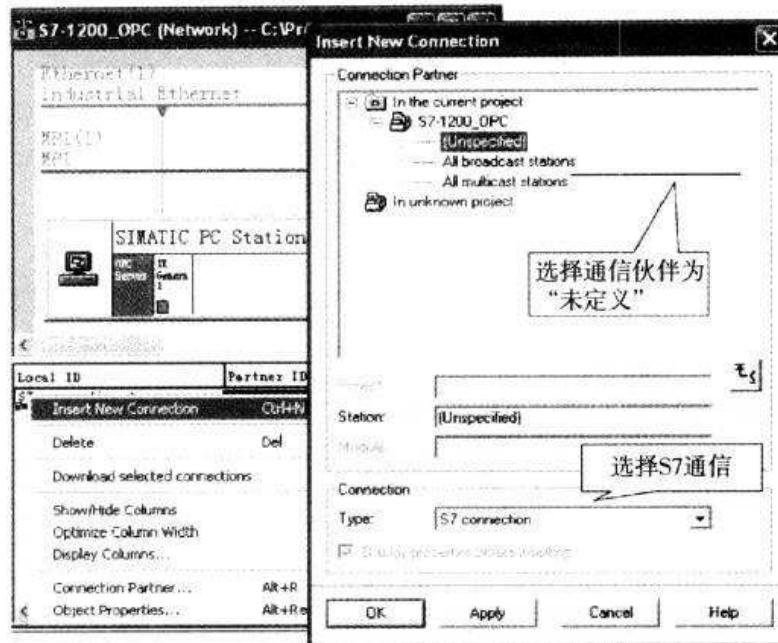


图 9-8 建立连接及定义连接属性

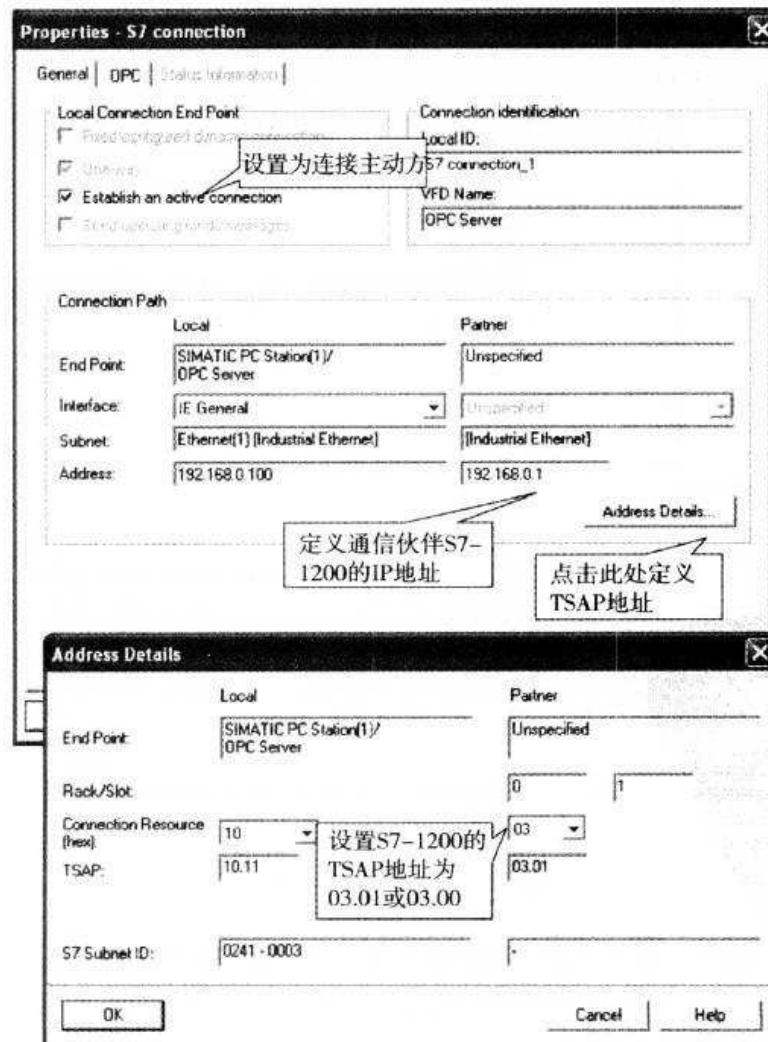


图 9-9 设置 S7 连接属性及 TSAP 地址



确认完成所有配置后，已建好的 S7 连接会显示在连接列表中。点击编译存盘按钮或选择“Network”→“Save and Compile”，如得到 No error 的编译结果，则表示组态完成。这里的编译结果信息非常重要，如果有错误信息（error Message），说明组态不正确，是不能下载到 PC Station 中的。编译完成后，在 STEP7 中的所有 PC Station 的硬件组态就完成了。

3. 创建一个虚拟的 PC Station 硬件机架

通过“Station Configuration Editor”创建一个虚拟的 PC Station 硬件机架，以便在 STEP7 中将组态的 PC Station 下载到这个虚拟的 PC Station 硬件机架中去。

- 1) 点击右下角的图标 ，进入 PC Station 硬件机架组态界面。
- 2) 选择第一号插槽，点击 Add 按钮或鼠标右键选择添加，在添加组件窗口中选择 OPCServer，如图 9-10 所示。

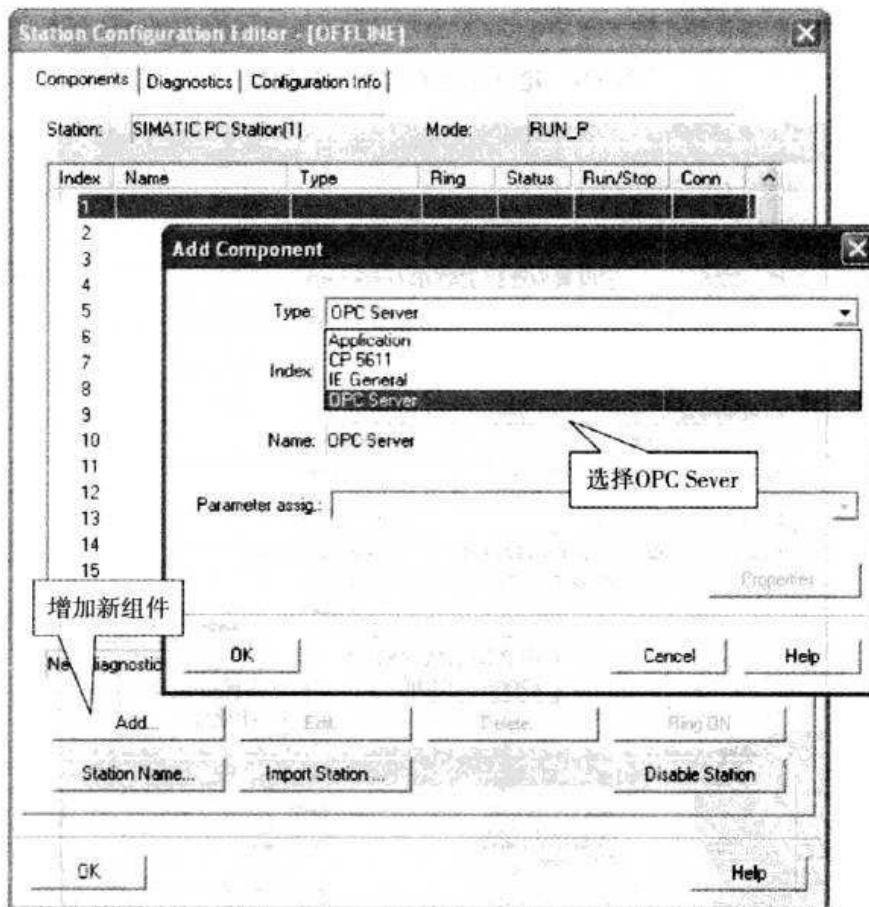


图 9-10 插入 OPC Server

- 3) 选择第三号插槽，点击 Add 按钮或鼠标右键选择添加，在添加组件窗口中选择 IE General，如图 9-11 所示。STEP7 中的 PC Station 硬件组态与虚拟 PC Station 硬件机架的名字、组件及“Index”必须完全一致。
- 4) 插入 IE General 后，随即会弹出组件属性对话框。点击“Network Properties”，进行

第9章 S7-1200 PLC 的 OPC 组态

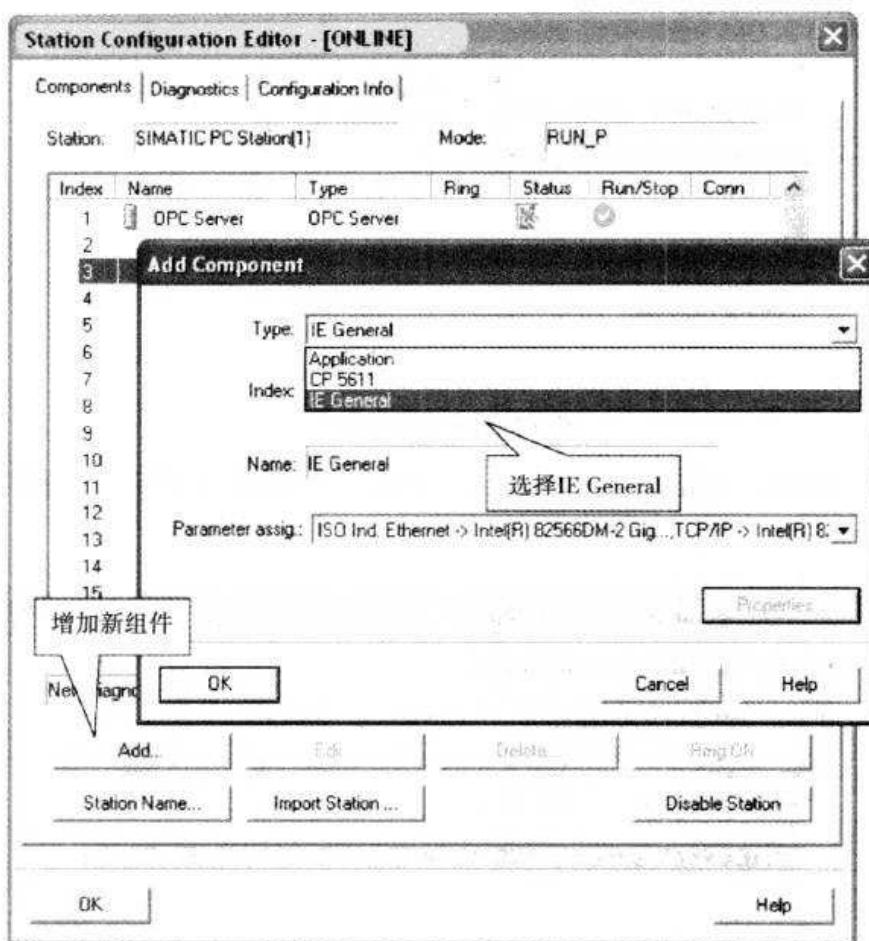


图 9-11 插入 OPC Server

网卡参数配置，如图 9-12 所示。

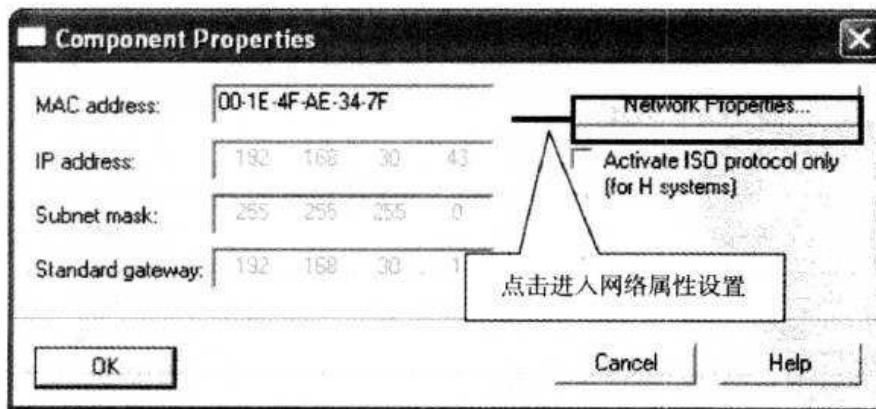


图 9-12 网卡参数配置

选择本地连接，通过右键点击菜单进入属性设置画面，选择 TCP/IP 属性，定义网卡地



图解西门子 S7-1200 PLC 入门到实践

址，如图 9-13 所示。

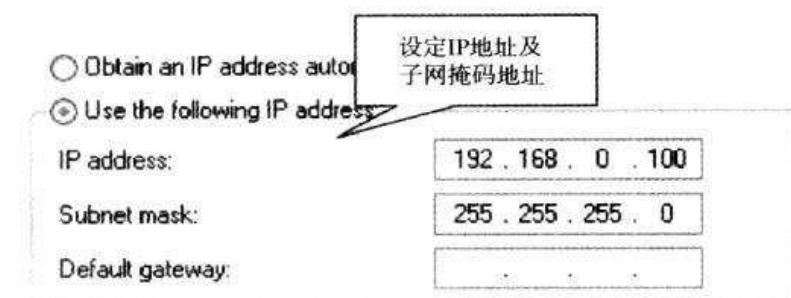


图 9-13 定义网卡地址

5) 命名 PC Station。这里的“PC Station”的名字一定要与 STEP 7 硬件组态中的“PC Station”的名字一致，如图 9-14 所示。

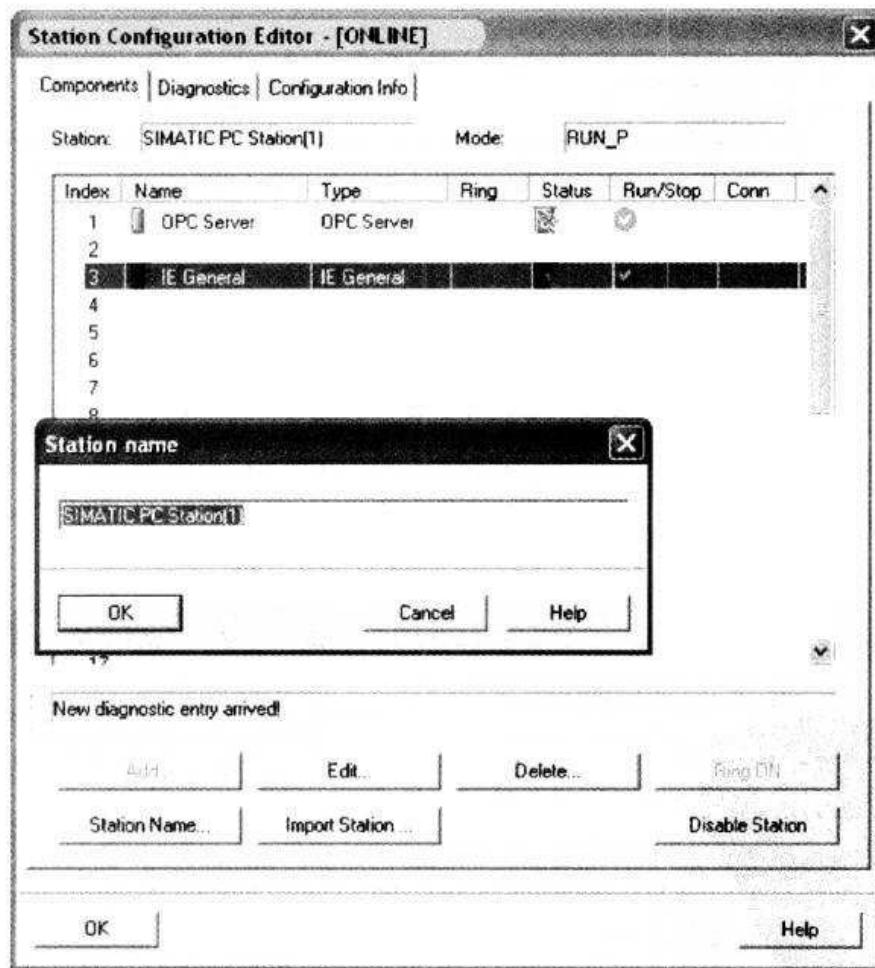


图 9-14 命名 PC STATION

4. 下载 PC Station 硬件组态及网络连接

1) 首先设置 PG/PC 接口，在 STEP7 软件中，通过“Options”→“Set PG/PCInterface”进入设置界面，如图 9-15 所示。

第9章 S7-1200 PLC 的 OPC 组态

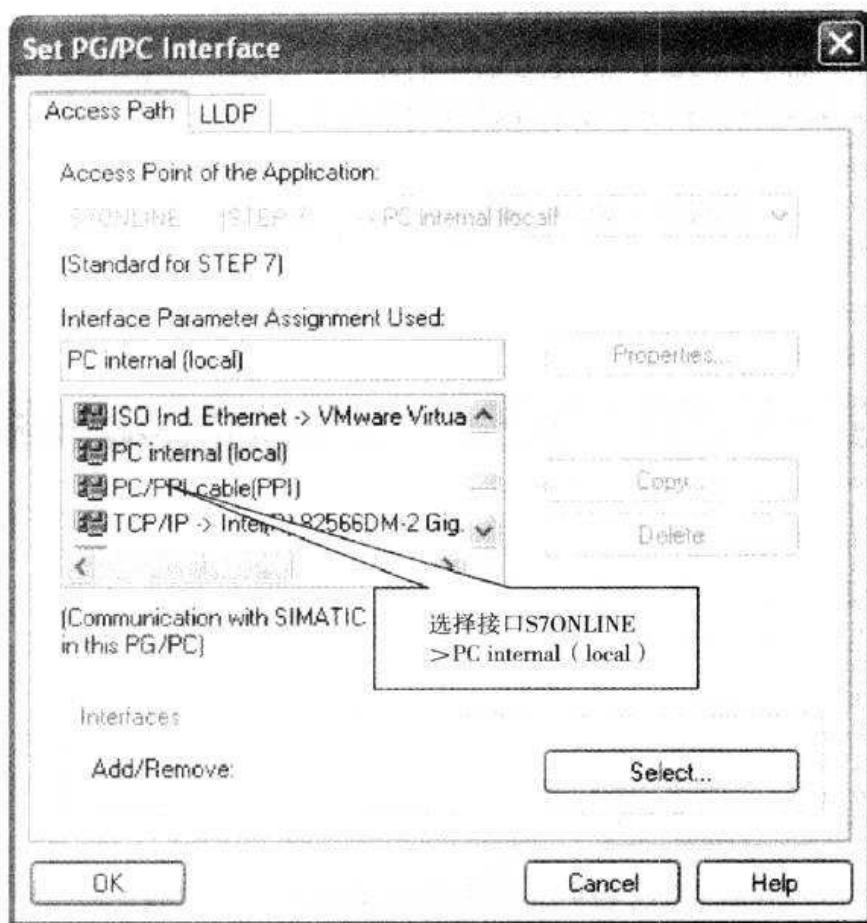


图 9-15 设置 PG/PC 接口

2) 检查配置控制台, 如图 9-16 所示。

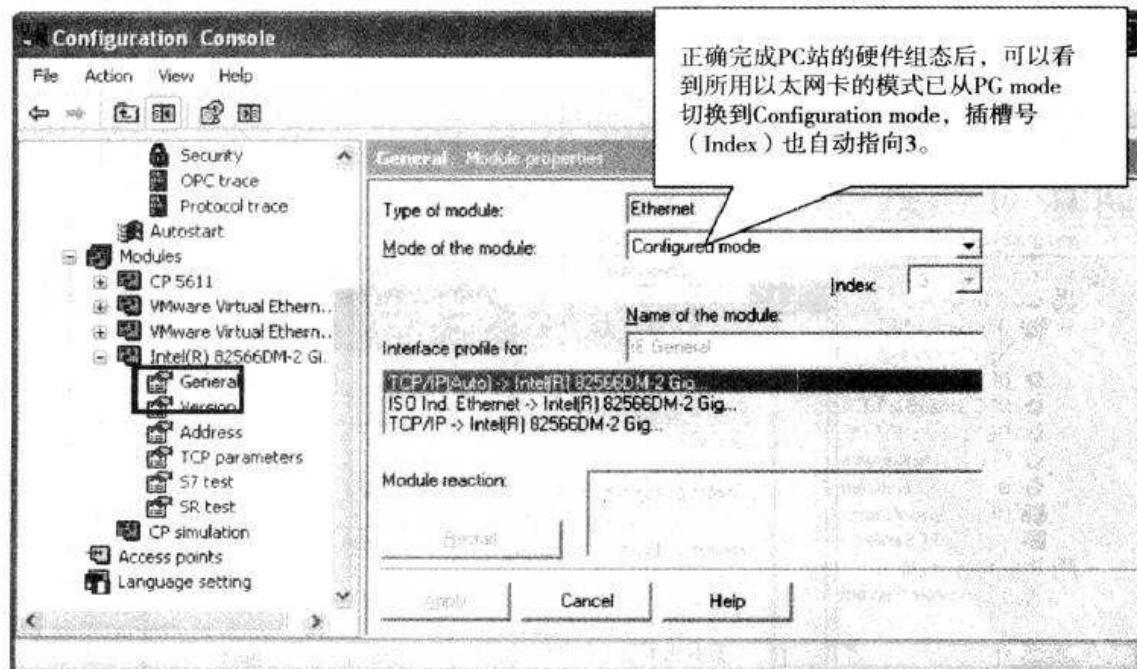


图 9-16 控制台



通过“start”→“Simatic”→“SIMATIC NET”→“Configuration Console”进入配置控制台检查。对于 Simatic Net V6.1 或 V6.0 版本的软件，需要在上面的窗口中，手动将模块模式（Mode of the module）从 PG 模式切换到组态模式（Configured mode），并设置 Index 号，然后再在 Station Configuration Editor 中添加硬件。

- 3) 在 STEP7 的硬件配置中下载 PC Station 组态。
- 4) 再在网络配置中将配置好的连接下载到 PC Station 中。

下载完成后，在“Station Configuration Editor”中状态显示，如图 9-17 所示。在编程过程中，可以根据这些状态显示判断组态是否正确。



图 9-17 下载完成后的状态

5. 使用 OPC Scout 测试 S7 OPC Sever

SIMATIC NET 自带 OPC Client 端软件 OPC Scout，可以使用这个软件测试所组态的 OPC Server。通过点击左下角的“Start”→“Simatic”→“SIMATIC NET”→“OPCScout”进行测试。

- 1) 双击“OPC. SimaticNET”，新建一个组并输入变量组的名称，例如 S7-1200，如图 9-18 所示。

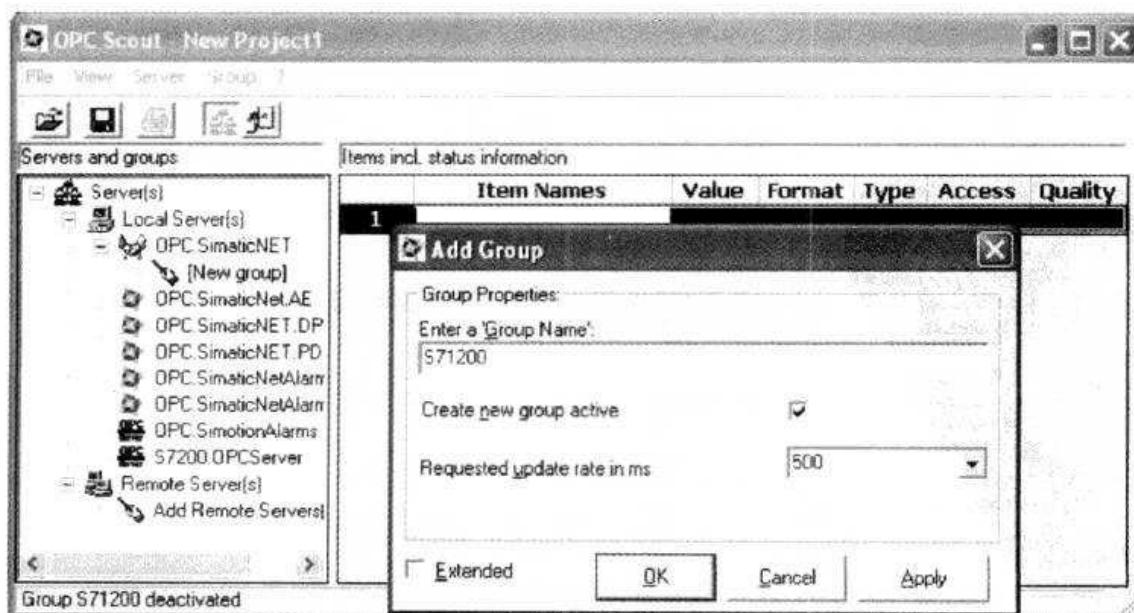


图 9-18 创建一个新的变量组

第9章 S7-1200 PLC 的 OPC 组态



2) 选择一个数据，点击“S7:”→“S7 connection _ 1”→“objects”→“M”→“NewDefinition”来添加一个变量，并为变量选择数据类型、起始地址、数据长度，并添加到右侧窗口中，如图 9-19 所示。如果使用 OPC 与 S7-1200 PLC 的 DB 通信，则建立的 DB 不能勾选“Symbolic access only (仅符号寻址)”，因为 OPC 不能访问只支持符号寻址的 DB。

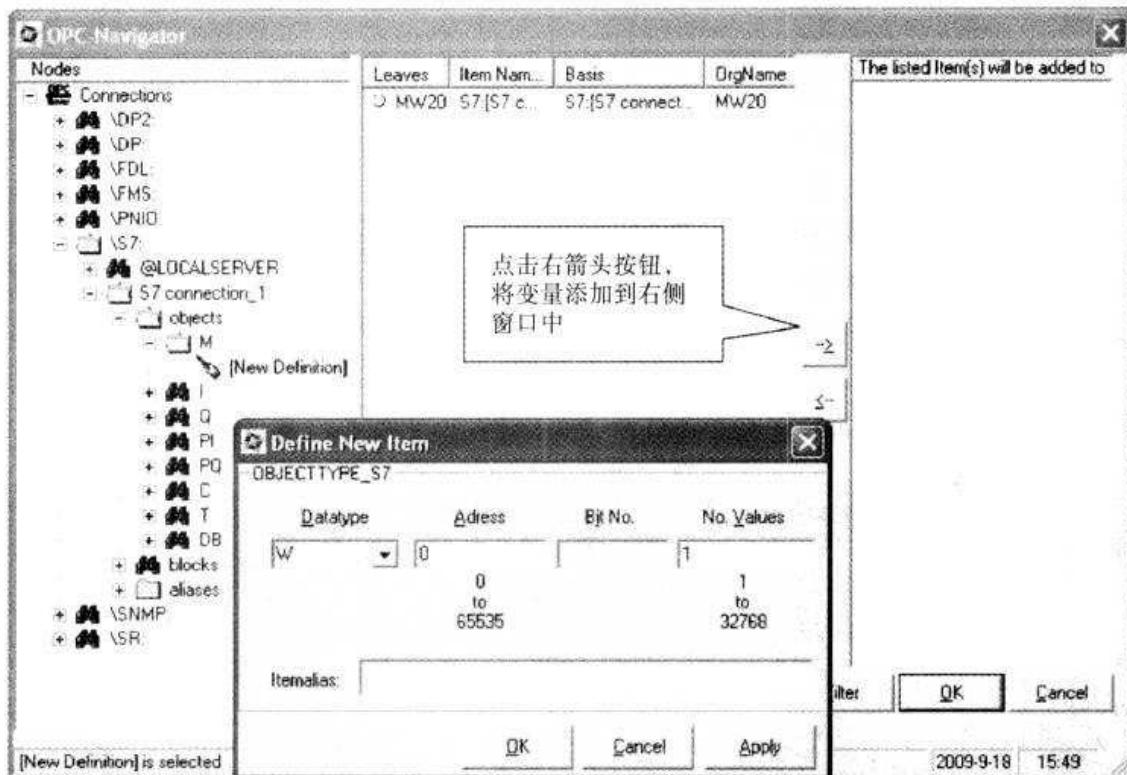


图 9-19 添加变量

3) 检查通信结果，如图 9-20 所示。如果通信质量为“bad”，则说明通信失败，需要检查软件组态及硬件连接是否正确。

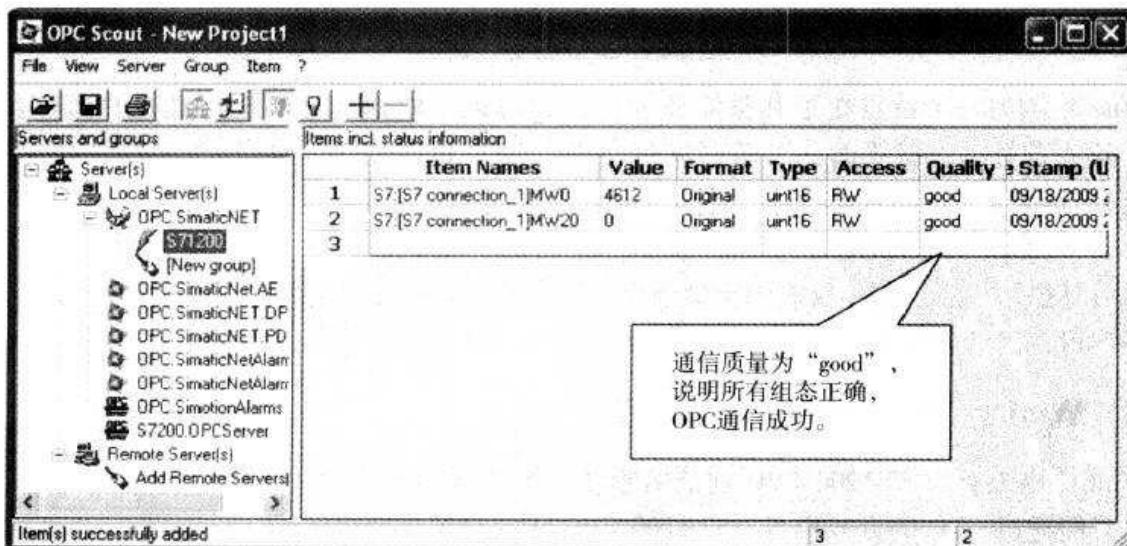


图 9-20 检查通信结果



9.3 技能训练二：WinCC 案例

9.3.1 WinCC 介绍

西门子公司的 WinCC 是 WindowsControlCenter（视窗控制中心）的简称。它集成了 SCA-DA、组态、脚本（Script）语言和 OPC 等先进技术，为用户提供了在 Windows 操作系统（Windows 2000 或 XP）环境下使用各种通用软件的功能。WinCC 继承了西门子公司的全集成自动化（TIA）产品的技术先进和无缝集成的特点。

WinCC 运行于个人计算机环境，可以与多种自动化设备及控制软件集成，具有丰富的设置项目、可视窗口和菜单选项，使用方式灵活，功能齐全。用户在其友好的界面下进行组态、编程和数据管理，可形成所需的操作画面、监视画面、控制画面、报警画面、实时趋势曲线、历史趋势曲线和打印报表等。它为操作者提供了图文并茂、形象直观的操作环境，不仅缩短了软件设计周期，而且提高了工作效率。如图 9-21 所示为 WinCC 监控画面。

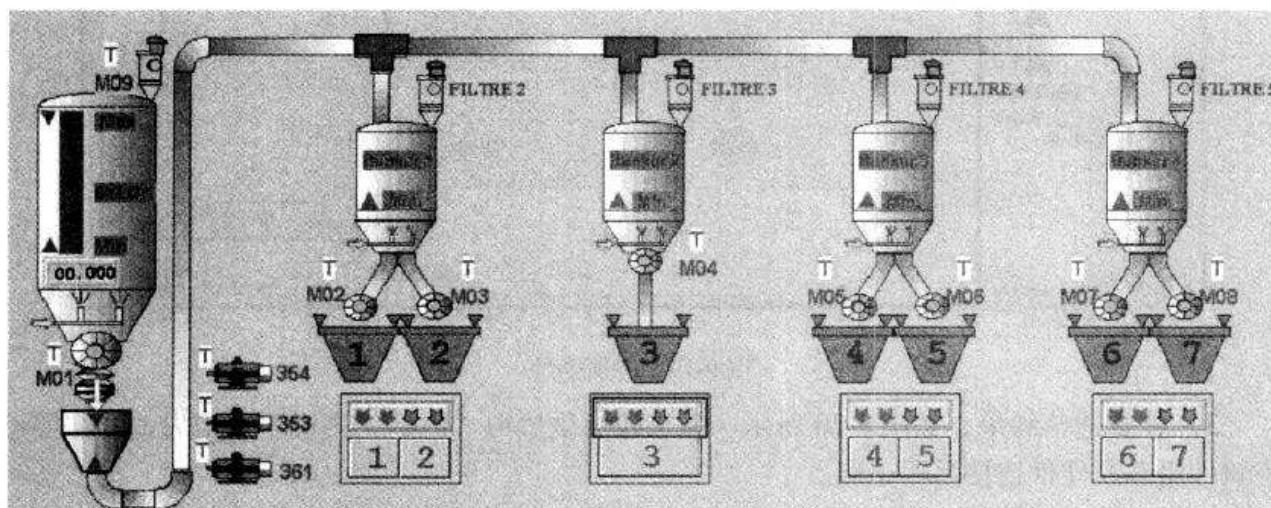


图 9-21 WinCC 监控画面

WinCC 的另一个特点在于其整体开放性，它可以方便地与各种软件和用户程序组合在一起，建立友好的人机界面，以满足实际需要。用户也可将 WinCC 作为系统扩展的基础，通过开放式接口，开发其自身需要的应用系统。

WinCC 因其具有独特的设计思想而具有广阔的应用前景。借助于模块化的设计，能以灵活的方式对其加以扩展。它不仅能用于单用户系统，而且能构成多用户系统，甚至包括多个服务器和客户机在内的分布式系统。WinCC 集生产过程和自动化于一体，实现了相互间的集成。

9.3.2 WinCC 与 S7-1200 CPU 通信

WinCC 中没有与 S7-1200 CPU 通信的驱动，所以 WinCC 与 S7-1200 CPU 之间通过以太网通信，只能通过 OPC 的方式实现。S7-1200 CPU 作为 OPC 的 Server 端，只需设置 IP 地址即可。上位机作为 OPC 的 Client 端，通过 SIMATIC NET 软件建立 PC Station 来与 S7-1200 CPU 通信。

具体的通信步骤如下：

第9章 S7-1200 PLC 的 OPC 组态



1) 如图 9-22 所示, 在 STEP 7 中生成一个 PC 站点, 将 OPC 服务器插入 1 号槽, 将通用以太网卡插入 3 号槽。

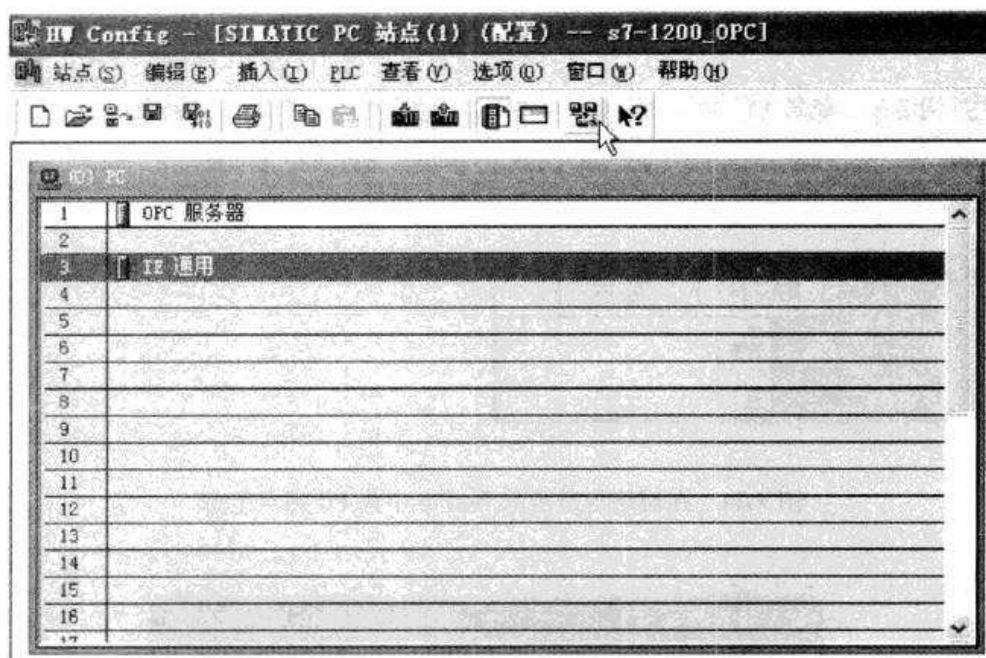


图 9-22 生成一个 PC 站点

2) 如图 9-23 所示, 用 SIMATIC NET 的站组态编辑器组态 PC 站。

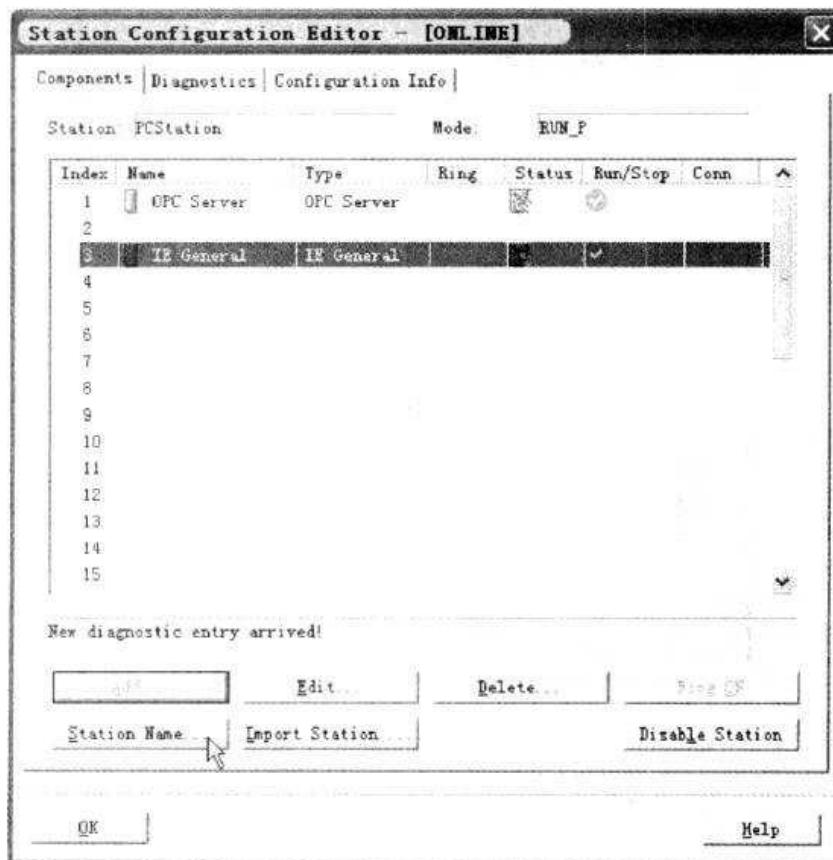


图 9-23 SIMATIC NET 的站组态编辑器组态 PC 站



图解西门子 S7-1200 PLC 入门到实践

3) 如图 9-24 所示, 在 STEP 7 的网络组态工具 NetPro 中为 PC 站点组态一个单向 S7 连接, 通信伙伴为“未指定”, 采用 S7-1200 PLC 的 IP 地址, 如图 9-25、图 9-26 所示。

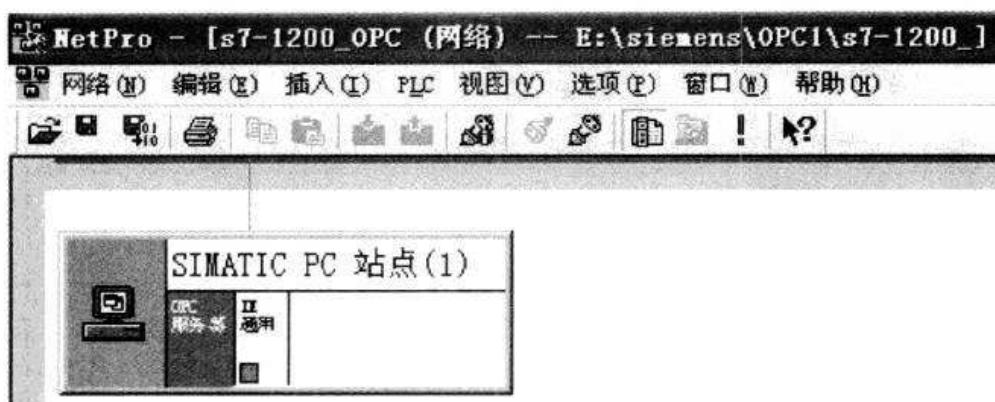


图 9-24 在网络组态工具 NetPro 中为 PC 站点组态

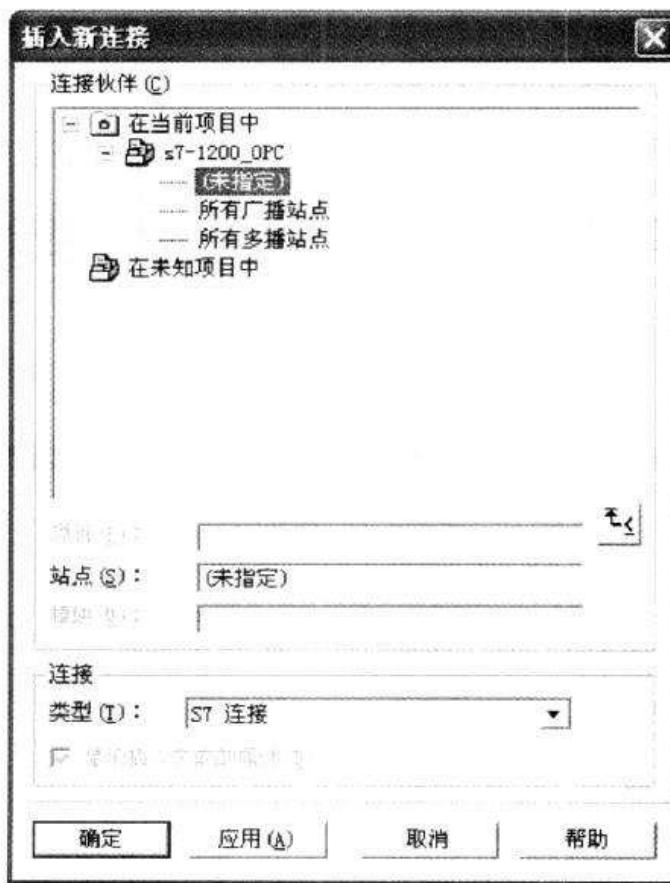


图 9-25 组态一个单向 S7 连接

4) 用 SIMATIC NET 的 OPC Scout 工具生成 OPC 的条目, 即 S7-1200 中的变量。

5) 在 WinCC V6.2 中添加 OPC 驱动程序, 在变量表中添加 OPC 的条目。

如图 9-27 所示, 打开 WinCC 软件新建一个项目, 用鼠标右键点击“变量管理”, 在快

第9章 S7-1200 PLC 的 OPC 组态

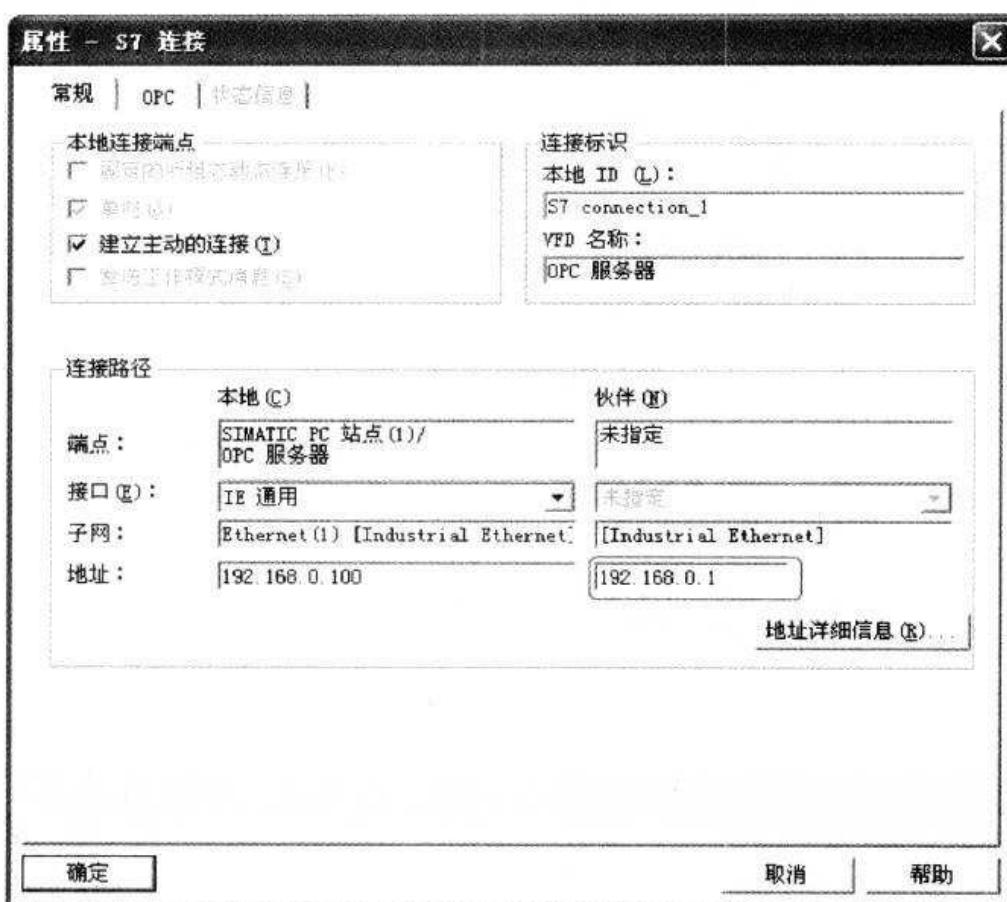


图 9-26 S7 连接属性定义

在快捷菜单中点击“添加新的驱动程序”，添加新的驱动：OPC.chn。

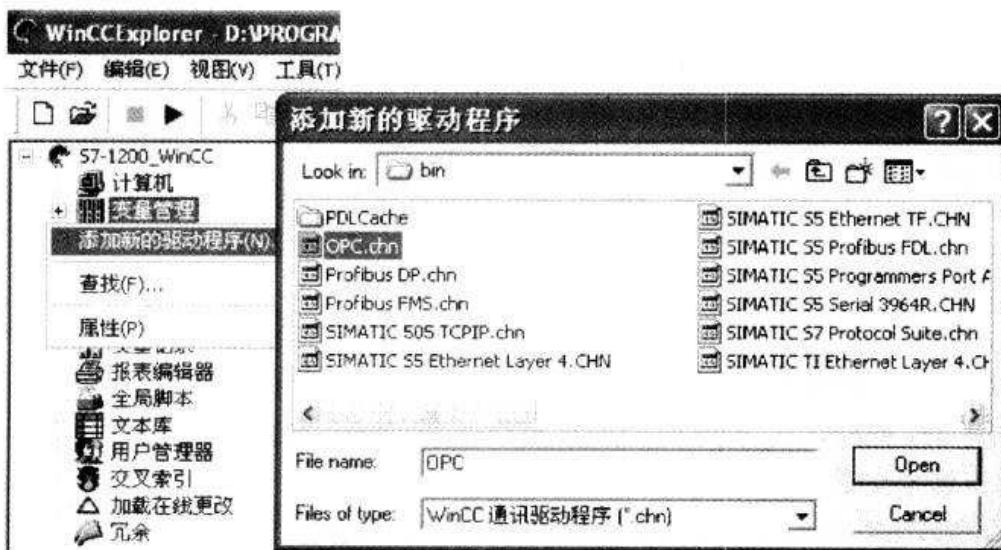


图 9-27 在 WinCC 添加一个新的驱动

在 WinCC 中搜索及添加 OPC Scout 中定义的变量，即用鼠标右键点击 OPC Groups，在快捷菜单中点击“系统参数”，如图 9-28 所示。

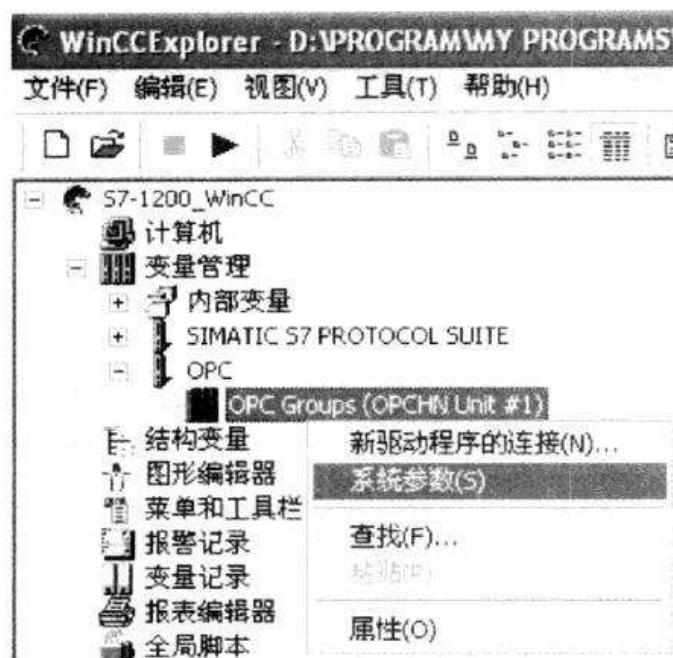


图 9-28 进入系统参数设置

然后选中 OPC. SimaticNET，点击“浏览服务器”按钮进行搜索，如图 9-29 所示。

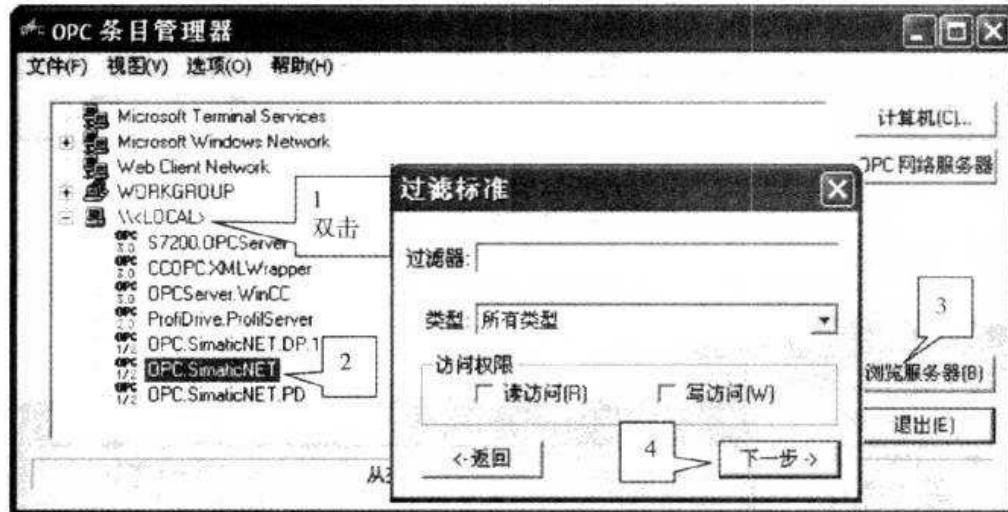


图 9-29 选择浏览服务器

在变量列表中选择所需要的变量，点“添加条目”按钮添加所需变量，此时会自动要求建立一个新连接，并将变量添加到这个连接中，如图 9-30 所示。

成功添加完变量后，在 WinCC 中即可进行变量显示。完成以上所有配置，就可以在 WinCC 里监控这些变量了。

6) 组态画面上的指示灯、按钮和输出域通过 OPC 的条目与 S7-1200 中的变量连接起来（见图 9-31）。

7) 在系统运行时，画面上的按钮将起动和停止信号发送到 S7-1200，PLC 程序运行的结果用画面上的指示灯显示。PLC 中动态变化的变量用输出域显示。实验表明 OPC 通信的响应速度很快。

第9章 S7-1200 PLC 的 OPC 组态

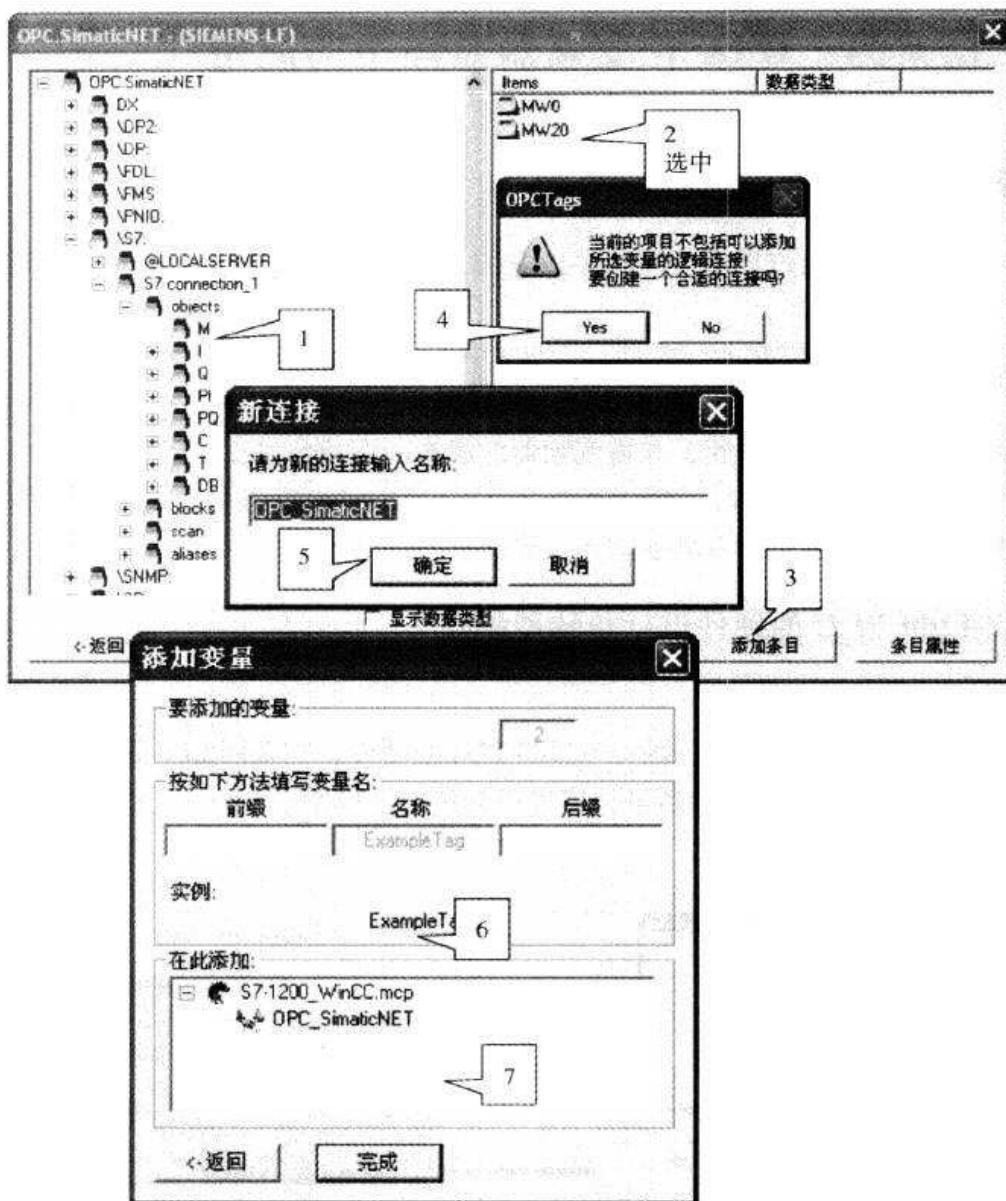


图 9-30 添加变量并建立连接



图 9-31 从 OPC Scout 中成功添加变量



9.4 工程案例：IFIX 在变频器监控中的应用

9.4.1 案例介绍

现有一工程，需要通过现场按钮和中控室的 IFIX 组态软件对变频器进行操作，具体要求如下：

- 1) 切换按钮通过中控室进行控制，并能在中控室显示当前的状态；
- 2) 当切换到现场操作时，按钮能通过 S7-1200 PLC 对该变频器进行启动和停止操作，同时能利用现场电位器进行调速，并将变频器的状态和调速信号反映到中控室；
- 3) 当切换到中控室操作时，现场操作无效，中控室的组态软件进行启动和停止变频器，同时利用数值输入框进行变频器的速度设定。

9.4.2 S7-1200 PLC 的硬件设计与软件编程

1. 硬件设计

根据要求设计了如图 9-32 所示的 S7-1200 PLC 的硬件，它采用 CPU1214C DC/DC/DC 和 SM1234 模拟量扩展模块，对变频器进行启停控制和速度控制。

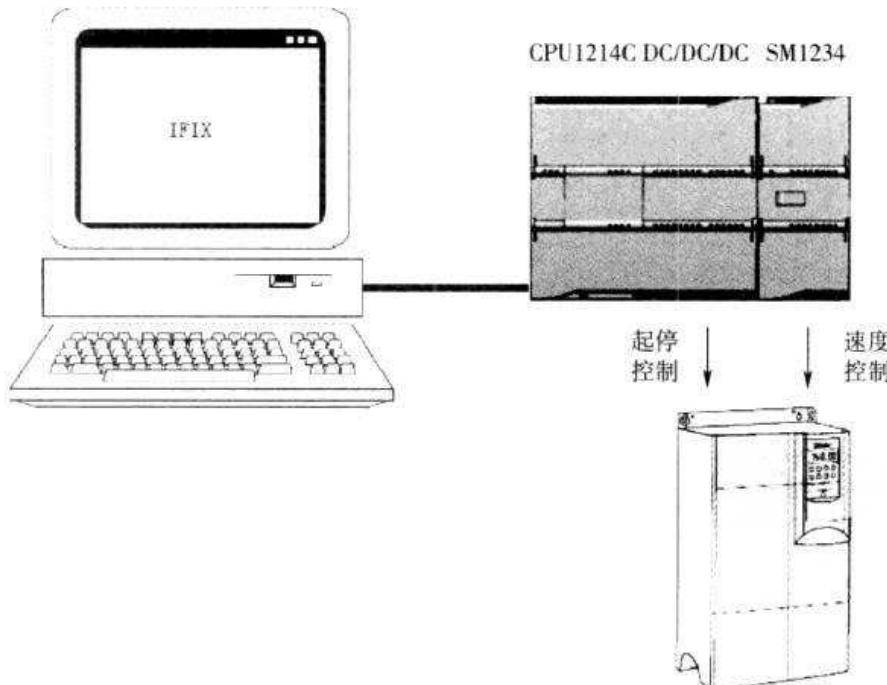


图 9-32 S7-1200 PLC 的硬件设计

2. 变量分配

“变频器监控”的 I/O 变量见表 9-1。



表 9-1 “变频器监控”的 I/O 变量

| | 名称 | 数据类型 | 地址 |
|----|----------|------|--------|
| 1 | 现场停止按钮 | Bool | %I0.0 |
| 2 | 现场启动按钮 | Bool | %I0.1 |
| 3 | 现场电位器输入 | Word | %IW96 |
| 4 | 变频器启动 | Bool | %Q0.0 |
| 5 | 变频器速度信号 | Word | %QW96 |
| 6 | IFIX停止 | Bool | %M0.0 |
| 7 | IFIX启动 | Bool | %M0.1 |
| 8 | 切换 | Bool | %M0.2 |
| 9 | 现场启动输出 | Bool | %M1.0 |
| 10 | IFIX启动输出 | Bool | %M1.1 |
| 11 | IFIX速度输入 | Word | %MW90 |
| 12 | 速度值 | Word | %MW96 |
| 13 | 中间转换变量1 | Real | %MD100 |
| 14 | 中间转换变量2 | Real | %MD104 |

3. 程序编写

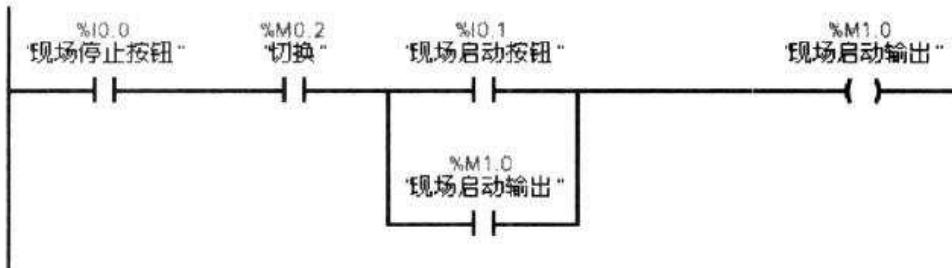
根据要求，变频器监控的主程序如图 9-33 所示。

块标题：

主程序

程序段 1：

当切换至现场控制时，进行按钮启动与停止控制



程序段 2：

当切换至IFIX控制时，由组态软件进行控制

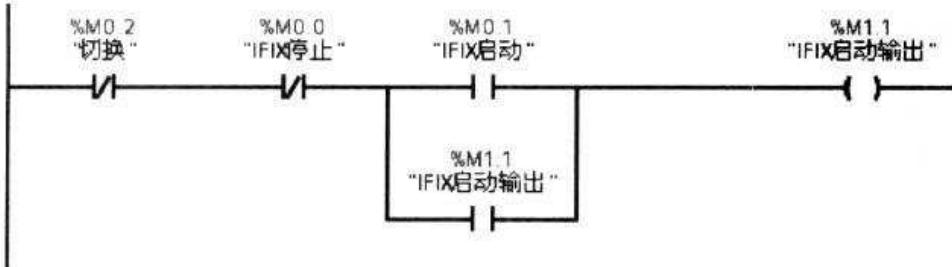
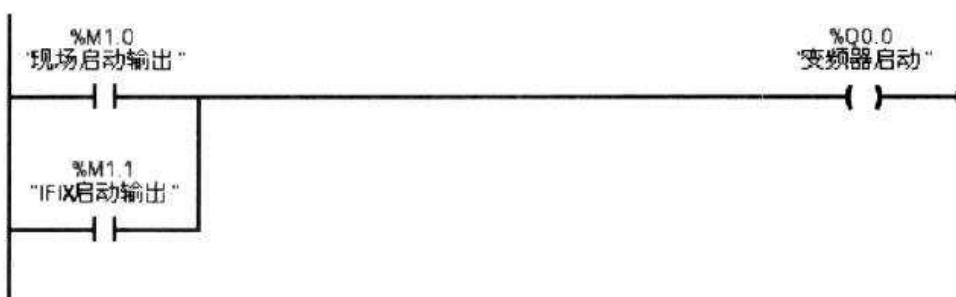


图 9-33 变频器控制主程序



▼ 程序段 3：

启动或停止变频器



▼ 程序段 4：

当现场控制时，进行速度值转换

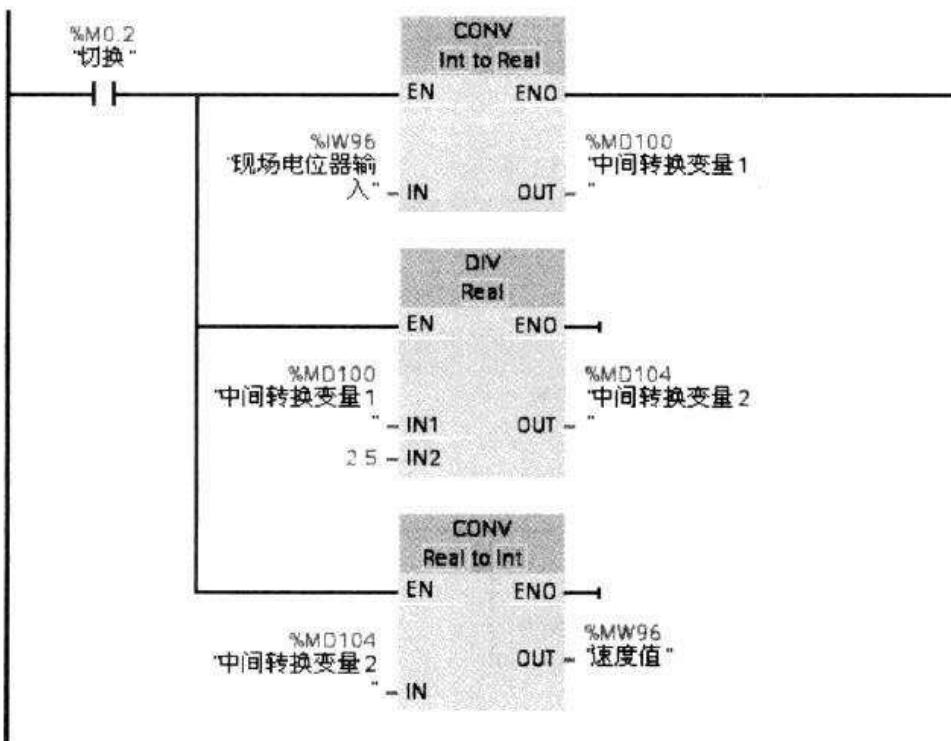


图 9-33 变频器控制主程序（续）

9.4.3 IFIX 的设置与组态

IFIX 软件在国内非常通用，其性能稳定，功能强大，界面友好，最重要的是其一直都支持 OPC 功能，即可以作为 OPC 服务器，也可以作为 OPC 客户端。

IFIX 软件是一套工业自动化软件，为用户提供一个“过程化的窗口”，能提供实时数据给操作员及软件应用。IFIX 的基本功能数据采集和数据管理，前者与工厂的 I/O 设备直接通信或通过 I/O 驱动程序与 I/O 设备接口，后者负责处理、使用所取数据，并以图形显示进行过程监视，完成监视控制、报警、报表、数据存档等功能。



1. 用 SIMATIC NET 建立 OPC 服务器 (见图 9-34)

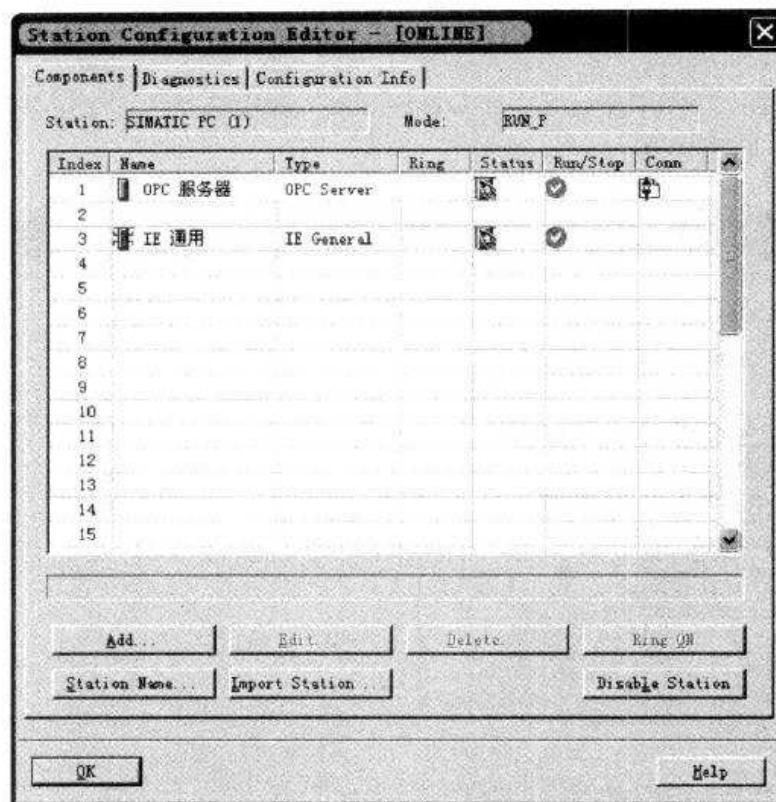


图 9-34 用 SIMATIC NET 建立 OPC 服务器

2. IFIX OPC Client 的配置和使用

安装完 IFIX，桌面上出现 IFIX 图标，其启动画面如图 9-35 所示。



图 9-35 启动 IFIX 组态软件



图解西门子 S7-1200 PLC 入门到实践

然后安装“OPCDrv OLE for Process Control Client, Version 7.32c”驱动，如果没有该驱动软件，可以向软件销售商索要。进入 SCU 系统配置，在 SCADA 组态的 I/O 驱动器定义中加载 OPC 驱动，安装完成后，SCU 里就出现了 OPC 这一项，双击进入“I/O Driver Server Connection”界面。在“I/O Driver Server Connection”界面选择“使用本地服务器”，服务器名称是“Server1”（见图 9-36）。

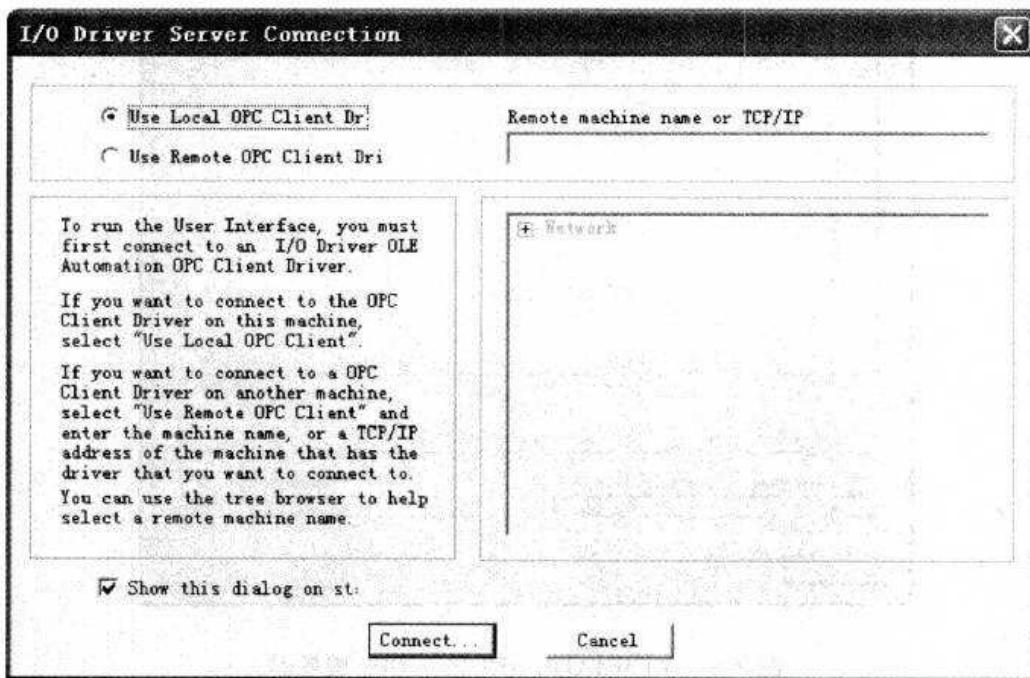


图 9-36 “I/O Driver Server Connection”界面

然后点“下一步”进入 PowerTool。PowerTool 是 IFIX 的 OPC 客户端软件，所有配置工作都在这里完成，它主要用来连接 OPC 服务器，其基本设置如图 9-37 所示。

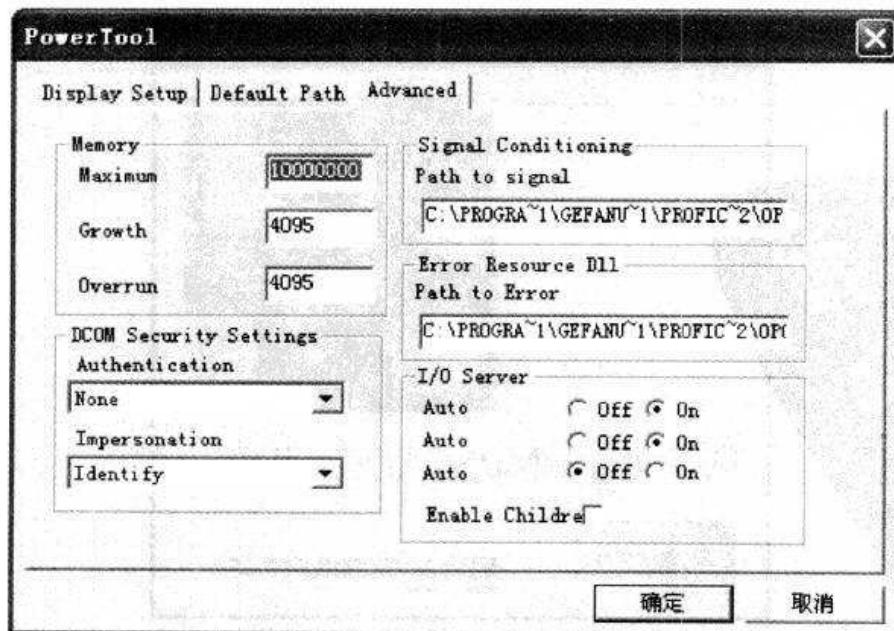


图 9-37 PowerTool 的基本属性设置



在数据组界面，I/O 分异步（Asynchronous）和同步（Synchronous）信号。异步为数据有变化时才更新，或看门狗超时才访问 OPC 服务器；同步为定期访问服务器。一般模拟量用异步，开关量用同步。一般参数对通信影响不大可以默认。如图 9-38 所示为输入 Item 的设置画面。

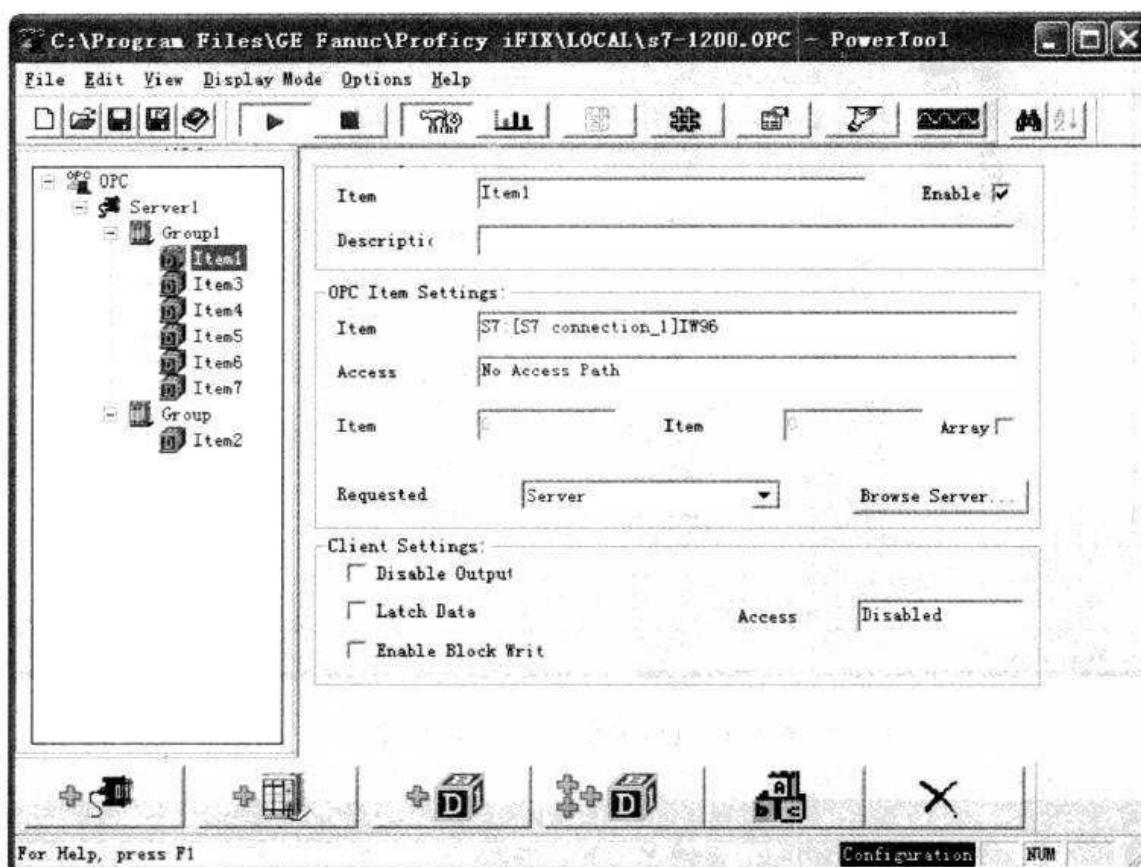


图 9-38 输入 Item 的设置界面

此时，可以点击“Browse Server...”按钮进入服务器数据项的选择画面，选择完成后“OPC Item Settings”会相应的改变。在以上设置全部正确的情况下，先在左面的树形结构中选择一个数据项，然后点击菜单“Display Mode”下的“数据状态”按钮，接着点击菜单栏上的绿色箭头，运行 OPC Client。图 9-39 中显示的就是 OPC Client 与 OPC Server 通信正常的情况，通信正常的标志是“Data”项的显示为 GOOD。

3. IFIX 数据库与 IFIX OPC Client 的连接

如图 9-40 所示，在数据库管理器中建立一批模拟量输入/输出、数字量输入/输出变量，I/O 地址是不可以选择的，必须按照默认格式逐个填写〔Server；Group；Item；Access〕四项，缺一不可，否则报错。

图 9-41 和图 9-42 所示为数字量和模拟量数据输入设置，点击“保存”按钮后变量添加到列表中，接着再次点击存盘按钮，保存列表，然后在表格上右击选择刷新，该变量的当前值就显示出来了，如果当前量位置显示“????”，说明连接或设置有错。

上面已经说明了数据传送的过程，SIMATIC NET OPC 服务器软件直接与 S7-1200 硬件通信，IFIX OPC 客户端 PowerTool 从 OPC 服务器上读取数据，提供给 IFIX 数据库管理器使用。

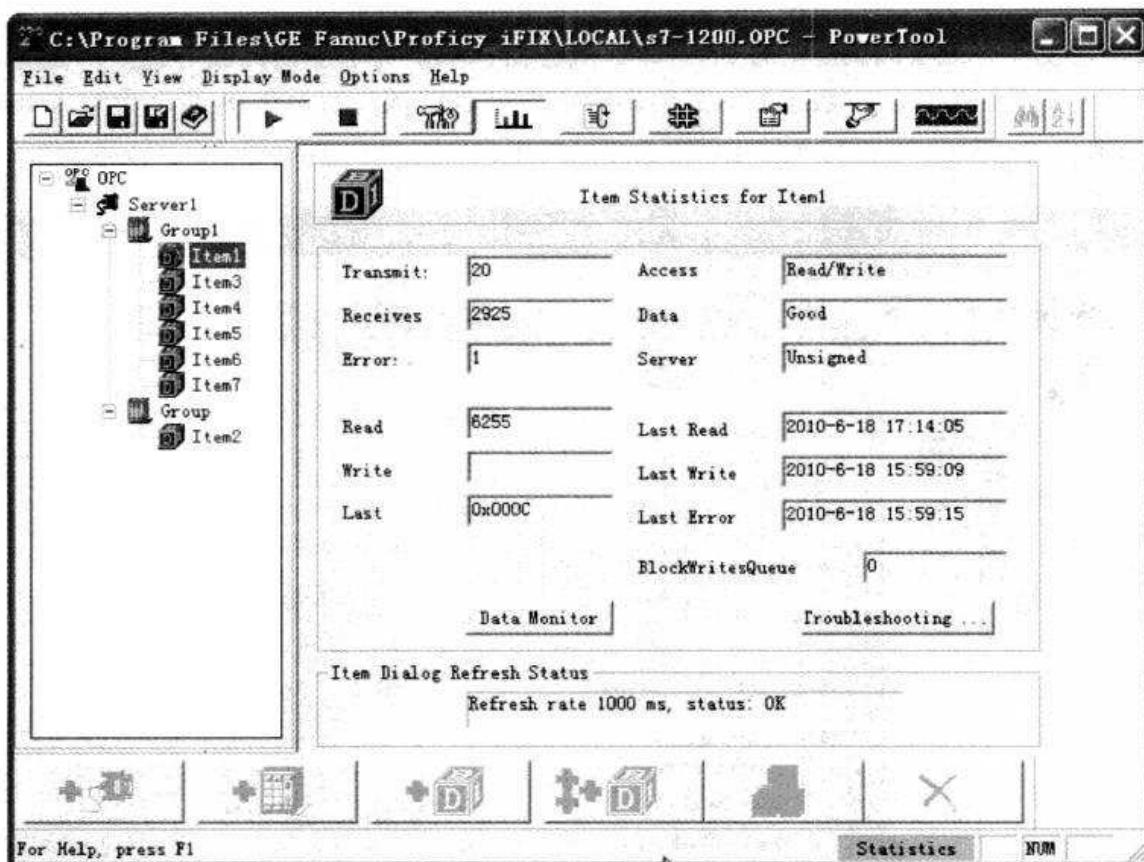


图 9-39 通信成功或失败的信息画面

| GE Proficy iFIX 数据库管理器 - [FIX : 9 列] | | | | | | | |
|--|----|----|------|-------|---------------|-----------|--|
| 数据库(E) 编辑(E) 显示(V) 数据块(B) 驱动器(D) 工具(T) 帮助(H) | | | | | | | |
| + 新建 + 打开 + 保存 + 导入 + 导出 + 剪切 + 复制 ? | | | | | | | |
| 标签名 | 类型 | 描述 | 扫描时间 | I/O设备 | I/O地址 | 当前值 | |
| 1 Q00 | DI | | 1 | OPC | Server1.Group | OPEN | |
| 2 MW96 | AI | | 1 | OPC | Server1.Group | ???? | |
| 3 MW90 | AI | | 1 | OPC | Server1.Group | 30.00 | |
| 4 M02 | DI | | 1 | OPC | Server1.Group | CLOSE | |
| 5 M01 | DI | | 1 | OPC | Server1.Group | OPEN | |
| 6 M00 | DI | | 1 | OPC | Server1.Group | OPEN | |
| 7 I01 | DI | | 1 | OPC | Server1.Group | OPEN | |
| 8 I00 | DI | | 1 | OPC | Server1.Group | CLOSE | |
| 9 IW96 | AI | | 1 | OPC | Server1.Group | 13,159.00 | |

图 9-40 IFIX 数据库管理器

因此，数据库的启动顺序如下：

1) 启动 OPC 服务器。在启动过程中，要连接控制器。

2) 启动 OPC 客户端 PowerTool。打开相应的导入文件，点击绿三角，进入运行状态。从停止到运行要等一段时间。如果很快进入了运行状态，要看一下 Server 的设置是否为 LOCAL。OPC 数据结构是树形的，依次为 Server、Group、Item，这三层都有使能端 ENABLE，要将 ENABLE 勾上，对应的数据项才会进行通信。所以要保证所有的这三层的 ENABLE 都勾上。

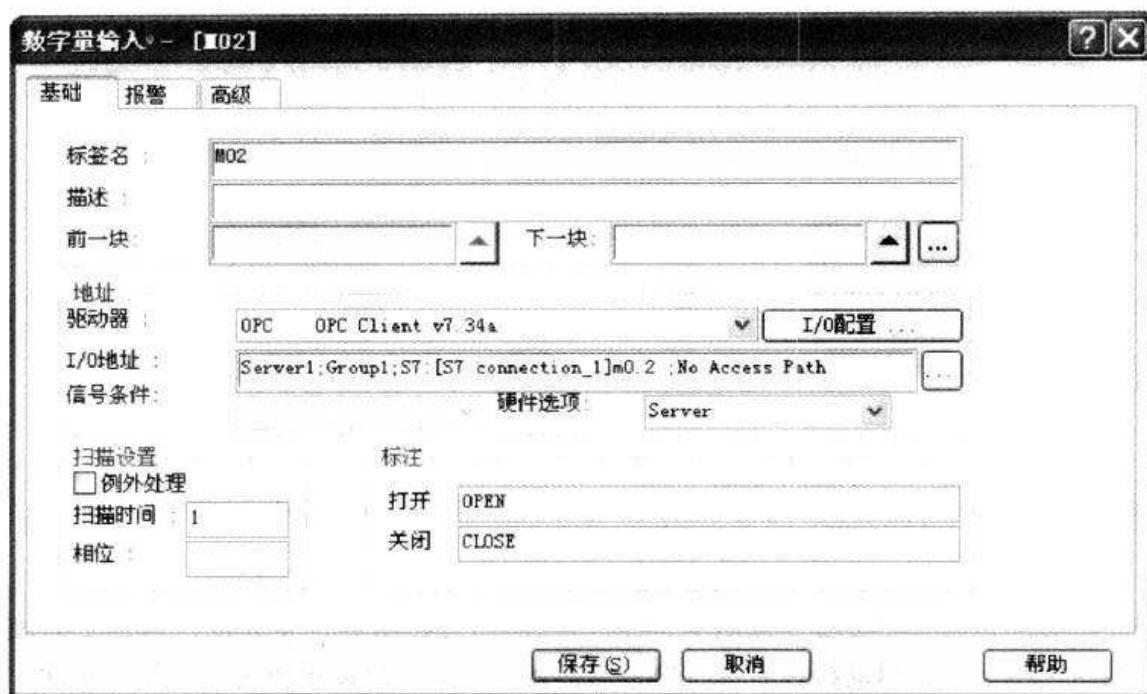


图 9-41 输入数字量变量的 I/O 地址

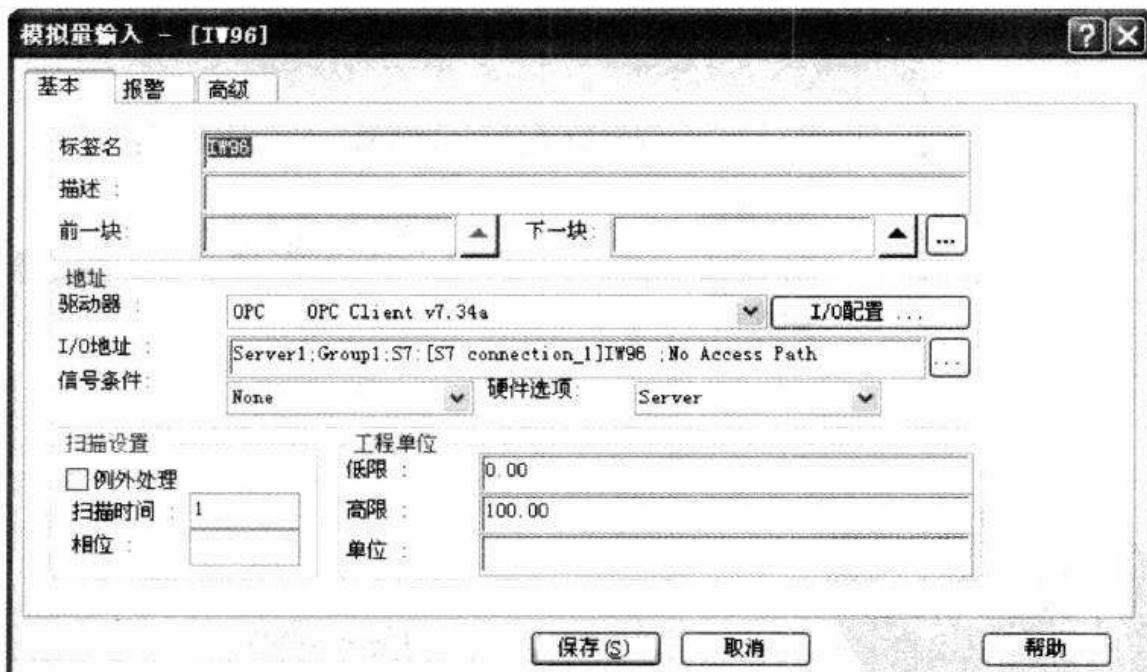


图 9-42 输入模拟量变量的 I/O 地址

4. IFIX 软件画面组态

如图 9-43 所示，进行“变频器监控主画面”的画面组态工作，它具体包括：按钮、指示灯、显示文本、数值输入和显示。

(1) 按钮动作的脚本定义

在 IFIX 组态软件中，脚本是采用 VB 来编写，尽管可以设置相应的“功能”选项，但

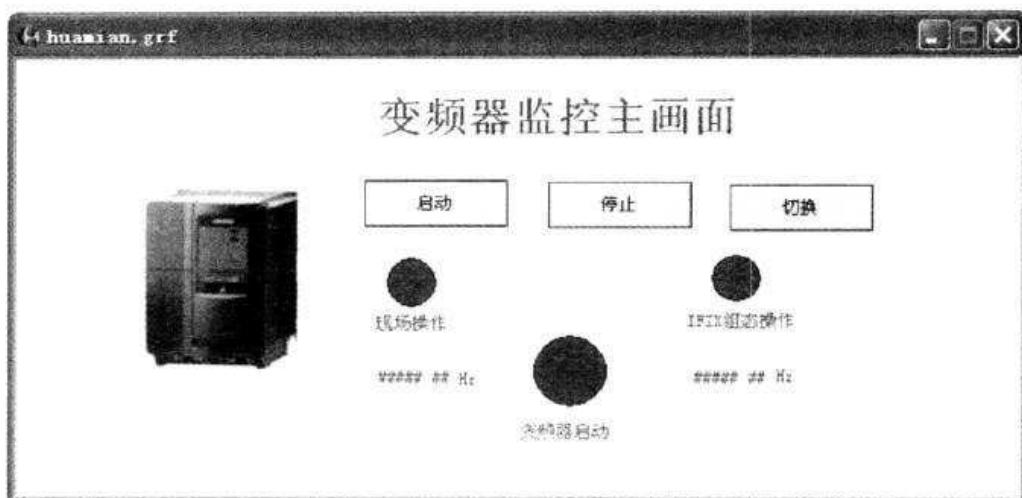


图 9-43 画面组态

有时候还不如脚本来得更直接。以图 9-44 所示的启动按钮组态定义为例，其定义的脚本含义为：当按钮按下时，启动按钮变量置位；当按钮放开时，启动按钮变量复位。因此，可以直接编写如图 9-45 所示的脚本。



图 9-44 启动按钮组态定义

在该脚本语句中，需要正确了解 IFIX 的数据源。其标准的数据信息标识句法为： SERVER. NODE. TAG. FIELD。其中， SERVER 为 OPC 数据服务器的名称； NODE 为数据库所在的节点名； TAG 为数据库中的标签名称； FIELD 为标签的特殊参数信息（域名）。例如，FIX32. SCADA1. FLOW_IN. F_CV 为显示 FLOW_IN 的当前值（F_CV）的含义，而变量 FLOW_IN 是驻留在 SCADA1 节点的 PDB 里，SCADA1 的数据来自 OPC 服务器 FIX32。

第9章 S7-1200 PLC 的 OPC 组态



```

Private Sub CommandButton1_MouseDown(ByVal Button As Integer, ByVal Shift As Integer, ByVal X As Single, ByVal Y As Single)
closeDigitalPoint "Fix32 FIX_M00_F_CV"
End Sub

Private Sub CommandButton1_MouseUp(ByVal Button As Integer, ByVal Shift As Integer, ByVal X As Single, ByVal Y As Single)
openDigitalPoint "Fix32 FIX_M00_F_CV"
End Sub

Private Sub CommandButton2_MouseDown(ByVal Button As Integer, ByVal Shift As Integer, ByVal X As Single, ByVal Y As Single)
closeDigitalPoint "Fix32 FIX_M01_F_CV"
End Sub

Private Sub CommandButton2_MouseUp(ByVal Button As Integer, ByVal Shift As Integer, ByVal X As Single, ByVal Y As Single)
openDigitalPoint "Fix32 FIX_M01_F_CV"
End Sub

Private Sub CommandButton4_Click()
    '下面的注释已自动加入。
    '任何修改都可能对该功能造成不利的影响
    '脚本编写专家。
    'Win32Name=切换数字量点
    'Property1=Fix32 FIX_M02_F_CV
    'Property2=False
    ToggleDigitalPoint "Fix32 FIX_M02_F_CV"
End Sub

```

图 9-45 启动按钮等的脚本

(2) 指示灯和文本的消隐组态

在 IFIX 组态软件中，消隐组态直接可以采用表达式来进行。在本案例中，当“切换”到现场或 IFIX 组态操作时，其相对应的指示灯和文本要消隐或变换颜色。

以图 9-46 所示的“现场操作”指示灯为例基本动画对话框，打开基本动画对话框，选择前景颜色，并进行组态，如图 9-47 所示的前景颜色专家。



图 9-46 指示灯基本动画对话框



图解西门子 S7-1200 PLC 入门到实践

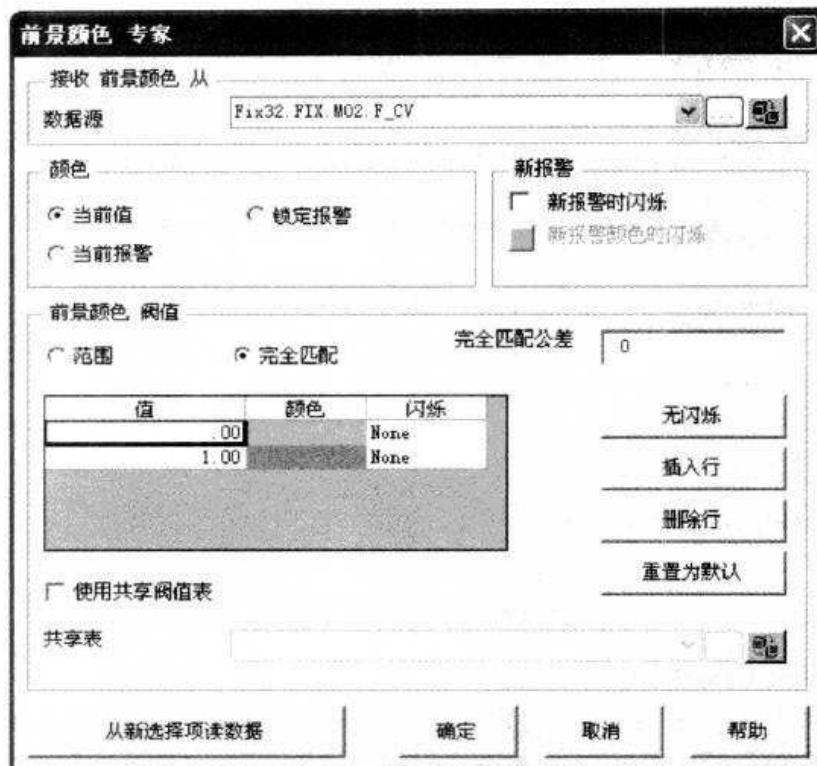


图 9-47 指示灯的前景颜色专家

在图 9-47 所示的前景颜色专家中，其数据源编辑如图 9-48 所示，其前景颜色值则根据要求进行选择。



图 9-48 指示灯颜色表达式编辑器



5. IFIX 运行

点击 IFIX 软件的运行按钮，出现如图 9-49 和图 9-50 所示的画面。

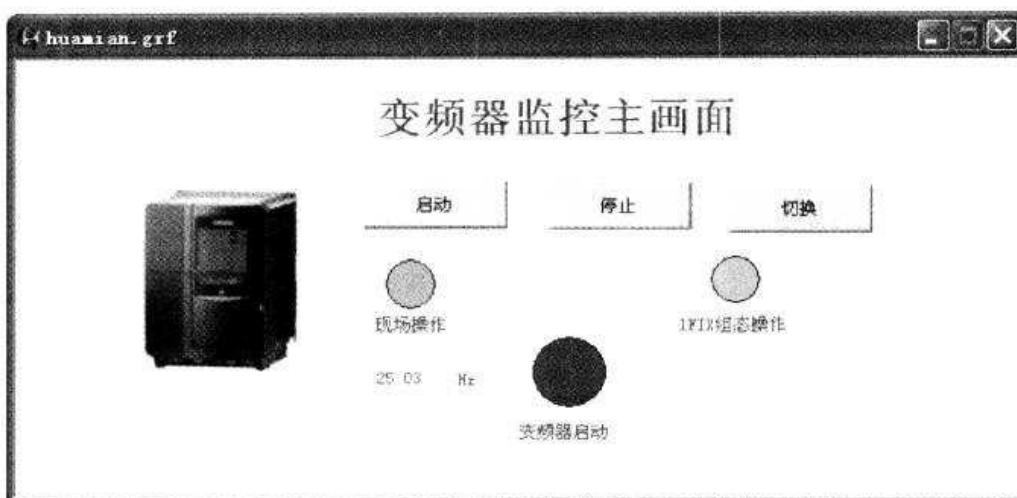


图 9-49 变频器控制的运行画面（现场操作）

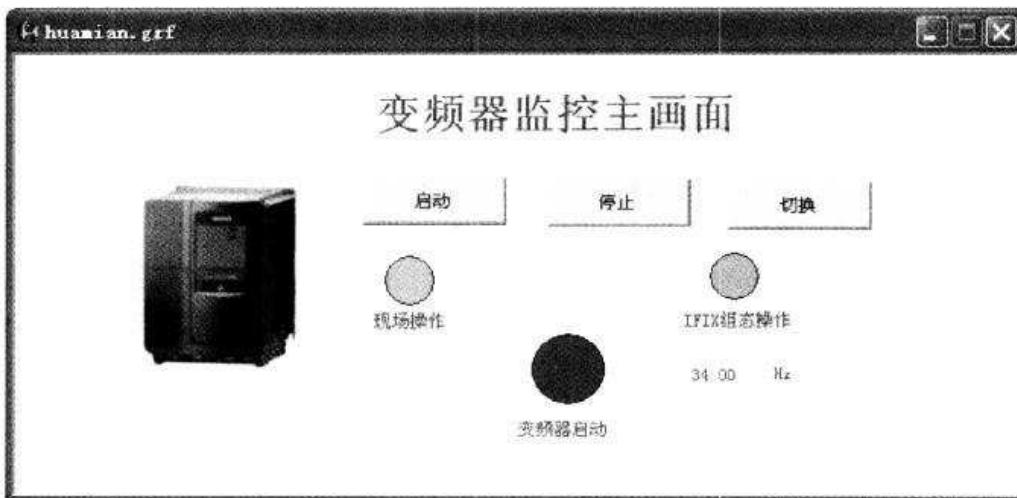


图 9-50 变频器控制的运行画面（组态软件操作）

9.4.4 工程案例小结

1. 工程中容易出现的问题

如果在 IFIX 画面上显示问号，则在 IFIX 数据库中此数据点也会显示问号，这表示数据还没有读上来，如果是刚启动，则要等一会儿。如果问号一直存在，就要在 PowerTool 中找原因。对于某些输出的点，如果一定时间（默认为 5min）没有数据存取操作，则将数据状态置为 BAD。可以先将数据项 Item 置为 Disable，然后再 ENABLE，如果 OPC 中的确存在这个点，则数据状态等一会儿就会变为 GOOD。

另外一种办法是将数据项 Item 的访问时间设为 Disabled，则 PowerTool 会一直更新这个数据项，不超时。如果数据项仍为 BAD，则要打开 OPC 服务器，手动查找数据库中是否真的存在这个点。路径设错会出现这种情况。



IFIX 数据库对 AI、DI 会一直读取数据，一般不存在访问超时的情况。如果是 AO、DO 点，画面不对这些点进行写操作，IFIX 数据库会让这些点超时，这时数据库里显示问号。例如，某个设备在画面上的联锁投入和切除按钮，它们的写值与读值连的是同一个变量，如果不将数据项 Item 的访问时间设为 Disabled，则一段时间之后，画面上这两个按钮会显示红色，点任何一个按钮则颜色恢复正常。这是因为点击这个操作将这个变量激活，系统重新对这个点进行计时，直到下一次超时再将此变量退出活动状态。

此外，当控制器发生故障时，也会出现这种情况。

2. 有前景的画面对象显示红色

红色是 IFIX 的系统报警色，如果此前景对应的数据点通信故障，会出现这种情况。比如上一段提到的情况。

3. 数据显示不正确

数据显示不正确时，需要检查 IFIX 画面数据连接是否连到了相应的数据库点和检查通道信号。在数据库管理器中，变量的 I/O 地址格式必须正确写入。

4. IFIX OPC 与 S7-1200 通信的发展趋势

OPC 既可以用于将现场 PLC 设备的下位机数据采集到 DCS 或 SCADA 中来，也可以用于将数据从一种 DCS 或 SCADA 系统采集到另一种 DCS 系统、SCADA 系统或其他上位应用程序中。在 OPC 的客户端/服务器结构中，同一个程序可以访问多个服务器，同一个服务器也可以同时为多个客户端服务。

事实上，带有 OPC 驱动的应用软件既可以是客户端，又可以是服务器。IFIX OPC 驱动同样也可以作为 OPC 服务器，提供数据给其他应用程序，甚至可以将 OPC 结合网络计算机来采集数据，允许客户从网络上运行的 OPC 服务器中获取需要的数据。

总之，把这一技术引进过程控制系统，使得 OPC 成为自动化系统、现场设备与工厂办公管理应用程序之间的有效联络工具，相互之间的数据交换得以简捷化和标准化，极大的方便了硬件开发商、软件开发商和最终用户。

参 考 文 献

- [1] 李方园. 自动化综合实践 [M]. 北京: 中国电力出版社, 2009.
- [2] 李方园. 维修电工技能实训 [M]. 北京: 中国电力出版社, 2009.
- [3] 李方园. PLC 行业应用实践 [M]. 北京: 中国电力出版社, 2007.
- [4] 李方园. 西门子 S7-1200 PLC 从入门到实践 [M]. 北京: 电子工业出版社, 2010.
- [5] 西门子 (中国) 有限公司. 深入浅出西门子 S7-1200 PLC [M]. 北京: 北京航空航天大学出版社, 2009.
- [6] 西门子自动化与驱动集团网站 (www.ad.siemens.com.cn) .