PLC WORLD

第一章 S7-300/400 的基本结构

1、 S7-300/400 属于模块式 PLC，主要由机架、CPU 模块、信号模块、功能模块、接口模块、通信处理器、电源模块和编程设备（工程师、操作员站和操作屏）组成。

图 1-1 PLC 控制系统示意图

PLC 的主要生产厂家：德国的西门子(Siemens)公司，美国 Rockwell 公司所属的 AB 公司，GE-Fanuc 公司，法国的施耐德(Schneider)公司，日本的三菱和欧姆龙(OMRON)公司。

PLC 的工作过程

表 1-1 逻辑运算关系表

与 或 非

Q4.0=I0.0*I0.1 Q4.1 = I0.2+I0.3 Q4.2 =/I0.4

I0.0	I0.1	Q4.0	I0.2	I0.3	Q4.1	I0.4	Q4.2
0	0	0	0	0	0	0	1
0	1	0	0	1	1	1	0
1	0	0	1	0	1		
1	1	1	1	1	1		

在 CPU 模块上有存储器(用来存放系统程序、用户程序、逻辑变量和其它一些信息)，包括 ROM 和 RAM。可通过扩展槽扩展用户 RAM。

l RAM：主程序区 OB1+子程序区（FB、FCB、定时中断块等）断电时由锂电池供电（几年）以免 RAM 中信息丢失。锂电池电压< 规定值，灯报警，换电池（期间靠电容充电几分钟）。

l PLC 采用循环执行用户程序的方式。

OB1 是用于循环处理的组织块（主程序），它可以调用别的逻辑块，或被中断程序（组织块）中断。在起动完成后，不断地循环调用 OB1，在 OB1 中可以调用其它逻辑块(FB, SFB, FC 或 SFC)。

循环程序处理过程可以被某些事件中断。

在循环程序处理过程中，CPU 并不直接访问 I/O 模块中的输入地址区和输出地址区，而是访问 CPU 内部的输入/输出过程映像区。批量输入、批量输出。

梯形图中 Q4.0 的线圈（称为内部线圈）“通电”时，对应的输出过程映像位为 1 状态。信号经输出模块隔离和功率放大后，继电器型输出模块中对应的硬件继电器的线圈（外部线圈）通电，其常开触点闭合，使外部负载通电工作。

外部输入电路接通时，对应的输入过程映像位(例如 I0.0)为 1 状态，梯形图中对应的输入位的常开触点接通，常闭触点断开。

某一编程元件对应的过程映像位为 1 状态时，称该编程元件为 ON，过程映像位为 0 状态时，称该编程元件为 OFF。

循环时间（Cycle time）：是指操作系统执行一次图 1-4 所示的循环操作所需的时间，又称为扫描循环时间（Scan Cycle Time）或扫描周期。如 0.7ms、1.7ms 等

l 性能指标：

I/O 点数、扫描周期、指令数目、功能模块多少、

PLC WORLD

```
A(
      O      I0.1   // 接在左侧母线上的 I0.1 的常开触点
      O    Q4.0   // 与 I0.1 的常开触点并联的 Q4.0 的常开触点
      )
      AN   I0.2      // 与并联电路串联的 I0.2 的常闭触点
      =       Q4.0       // Q4.0 的线圈
```

梯形图对应的逻辑表达式： Q4.0 = (I0.1+Q4.0)

I PLC 性能指标：

第二章 西门子 PLC 的分类

1．S7 系列：传统意义的 PLC 产品， S7-200 是针对低性能要求的小型 PLC。S7-300 是模块式中小型 PLC，最多可以扩展 32 个模块。S7-400 是大型 PLC，可以扩展 300 多个模块。S7-300/400 可以组成 MPI、PROFIBUS 和工业以太网等。

2．M7-300/400：采用与 S7-300/400 相同的结构，它可以作为 CPU 或功能模块使用。具有 AT 兼容计算机的功能，可以用 C，C＋＋或 CFC 等语言来编程。

3．C7 由 S7-300 PLC，HMI（人机接口）操作面板、I/O、通信和过程监控系统组成。

4．WinAC 基于 Windows 和标准的接口(ActiveX，OPC)，提供软件 PLC 或插槽 PLC。

S7-300 系列 PLC 简介

S7-300 的 CPU 模块(简称为 CPU)都有一个编程用的 RS-485 接口，有的有 PROFIBUS-DP 接口或 PtP 串行通信接口，可以建立一个 MPI（多点接口）网络或 DP 网络。

1.电源模块 2.后备电池 3. 24V DC 连接器 4.模式开关 5.状态和故障指示灯
6.存储器卡(CPU 313 以上) 7. MPI 多点接口 8.前连接器 9.前盖

图 2-2 S7-300 PLC

功能最强的 CPU 的 RAM 为 512KB，最大 8192 个存储器位，512 个定时器和 512 个计数器，数字量最大 65536 I/O 点，模拟量通道最大为 4096。有 350 多条指令。一个数字量为 1 点，一个模拟量为 16 点。

计数器的计数范围为 1～999，定时器的定时范围为 10ms～9990s。

图 2-3 S7-300 的安装

图 2-4 多机架的 S7-300 PLC

只需要扩展一个机架，可以使用价格便宜的 IM 365 接口模块对。

数字量模块：从 0 号机架的 4 号槽开始，每个槽位分配 4 个字节的地址，32 个 I/O 点。

模拟量模块：一个通道占一个字地址。从 I B256 开始，给每一个模拟量模块分配 8 个字。

1．模块诊断功能

可以诊断出以下故障：失压，熔断器熔断，看门狗故障，EPROM、RAM 故障。

模拟量模块共模故障、组态/参数错误、断线、上下溢出。

2．过程中断

数字量输入信号上升沿、下降沿中断，模拟量输入超限，CPU 暂停当前程序，处理 OB40。

3．状态与故障显示 LED

SF（系统出错/故障显示，红色）：CPU 硬件故障或软件错误时亮。

BATF（电池故障，红色）：电池电压低或没有电池时亮。

PLC WORLD

DC 5V（＋5V 电源指示，绿色）： 5V 电源正常时亮。
FRCE（强制，黄色）：至少有一个 I/O 被强制时亮。
RUN（运行方式，绿色）：CPU 处于 RUN 状态时亮；重新启动时以 2 Hz 的频率闪亮；HOLD（单步、断点）状态时以 0.5Hz 的频率闪亮。
STOP（停止方式，黄色）：CPU 处于 STOP，HOLD 状态或重新启动时常亮。
BUSF（总线错误，红色）。

图 2-5 CPU 318-2 的面板
4．模式选择开关
（1）RUN-P(运行-编程)位置：运行时还可以读出和修改用户程序，改变运行方式。
（2）RUN (运行)位置：CPU 执行、读出用户程序，但是不能修改用户程序。
（3）STOP（停止）位置：不执行用户程序，可以读出和修改用户程序。
（4）MRES（清除存储器）：不能保持。将钥匙开关从 STOP 状态搬到 MRES 位置，可复位存储器，使 CPU 回到初始状态。
复位存储器操作：通电后从 STOP 位置扳到 MRES 位置，“STOP”LED 熄灭 1s，亮 1s，再熄灭 1s 后保持亮。放开开关，使它回到 STOP 位置，然后又回到 MRES，“STOP”LED 以 2Hz 的频率至少闪动 3s，表示正在执行复位，最后“STOP”LED 一直亮。
某些 CPU 模块上有集成 I/O。
PLC 使用的物理存储器：RAM，ROM，快闪存储器（Flash EPROM）和 EEPROM。
S7-300 CPU 的分类
1．紧凑型 CPU：CPU 312C，313C，313C-PtP，313C-2DP，314C-PtP 和 314C-2DP。各 CPU 均有计数、频率测量和脉冲宽度调制功能。有的有定位功能，有的带有 I/O。
2．标准型 CPU：CPU 312，CPU 313，314，315，315-2DP 和 316-2DP。
3．户外型 CPU：CPU 312 IFM，314 IFM，314 户外型和 315-2DP。在恶劣的环境下使用。
4．高端 CPU：317-2DP 和 CPU 318-2DP。
5．故障安全型 CPU：CPU 315F。
S7-300 的输入/输出模块
输入/输出模块统称为信号模块(SM)。
前连接器插在前盖后面的凹槽内。一个编码元件与之啮合，该连接器只能插入同类模块。

两线式接近开关的漏电流小于输入模块允许的静态电流，汇点输入的电流流进输入模块，反之为源输入电路。

SM323 是 S7-300 的数字量输入输出模块，8DI/8DO，16DI/16DO。

表 2-13 SM331 模拟量输入模块的模拟值
范围 双极性
百分比 十进制 十六进制 ±5V ±10 V ±20 mA
上溢出 118.515% 32767 7FFFH 5.926 V 11.851V 23.70 mA
超出范围 117.589% 32511 7EFFH 5.879 V 11.759V 23.52 mA
正常范围 100.000% 27648 6C00H 5V 10 V 20 mA
0 % 0 0H 0V 0 V 0mA
-100.000% - 27648 9400H -5V - 10 V -20 mA

PLC WORLD

低于范围　-117.593%　- 32512　8100H　-5.879 V　- 11.759 V　-23.52 mA
下溢出　-118.519%　- 32768　8000H　-5.926 V　- 11.851 V　-23.70 mA

范围　单极性
　百分比　十进制　十六进制　0~10 V　0~20 mA　4~20 mA
上溢出　118.515%　32767　7FFFH　11.852 V　23.70 mA　22.96mA
超出范围　117.589%　32511　7EFFH　11.759 V　23.52 mA　22.81mA
正常范围　100.000%　27648　6C00H　10 V　20 mA　20 mA
　0 %　0　0H　0 V　0 mA　4 mA
低于范围　- 17.593 %　- 4864　ED00H　- 3.52 mA　1.185mA

模拟值的精度小于 15 位，则模拟值左移，左对齐。
【例 2-2】压力变送器的量程为 0～10MPa，输出信号为 4～20mA，模拟量输入模块的量程为 4～20mA，转换后的数字量为 0～27 648，设转换后得到的数字为 N，试求以 kPa 为单位的压力值。
　解：0～10MPa(0～10 000kPa)对应于转换后的数字 0～27 648，转换公式为
P = 10 000 ´ N / 27 648　（kPa）
注意：在运算时一定要先乘后除，否则可能会损失原始数据的精度。

图 2-13　量程卡

【例 2-3】某发电机的电压互感器的变比为 10kV/100V（线电压），电流互感器的变比为 1000A/5A，功率变送器的额定输入电压和额定输入电流分别为 AC 100V 和 5A，额定输出电压为 DC ±10V，模拟量输入模块将 DC ±10V 输入信号转换为数字+27648 和-27649。设转换后得到的数字为 N，求以 kW 为单位的有功功率值。
解：根据互感器额定值计算的原边有功功率额定值为

由以上关系不难推算出互感器原边的有功功率与转换后的数字之间的关系为
17321/ 27648 = 0.62648 kW / 字。转换后的数字为 N 时，对应的有功功率为 0.6265 N (kW)，如果以 kW 为单位显示功率 P，使用定点数运算时的计算公式为
　P = N ´ 6 265 / 10 000　（kW）
【例 2-4】用于测量锅炉炉膛压力（−60 Pa～60 Pa）的变送器的输出信号为 4～20mA，模拟量输入模块将 0～20mA 转换为数字 0～27 648，设转换后得到的数字为 N，试求以 0.1Pa 为单位的压力值。
　解：4～20mA 的模拟量对应于数字量 5530～27648，即−600～600（0.1Pa）对应于数字量 5 530～27 648，压力的计算公式应为

1．EX 系列数字量模拟量输入/输出模块
EX 模块在化工等行业使用。将外部的本质-安全设备（用于有爆炸危险区域的传感器和执行器）与 PLC 非本质-安全内部回路隔离。
2．F 系列数字量模拟量输入/输出模块

PLC WORLD

这些模块具有故障安全运行的集成安全功能，在 ET 200M 分布式 I/O 或 S7-300F 中使用。用于连接有爆炸危险区域的设备。

S7-300 的功能模块

1．计数器模块

模块的计数器均为 0～32 位或 31 位加减计数器，可以判断脉冲的方向，模块给编码器供电。达到比较值时发出中断。可以 2 倍频和 4 倍频计数。有集成的 DI/DO。

FM 350-1 是单通道计数器模块，可以检测最高达 500kHz 的脉冲，有连续计数、单向计数、循环计数 3 种工作模式。FM 350-2 和 CM 35 都是 8 通道智能型计数器模块。

2．位置控制与位置检测模块

FM 351 双通道定位模块用于控制变级调速电动机或变频器。FM 353 是步进电机定位模块。FM 354 伺服电机定位模块。FM 357 可以用于最多 4 个插补轴的协同定位。FM 352 高速电子凸轮控制器，它有 32 个凸轮轨迹，13 个集成的 DO，采用增量式编码器或绝对式编码器。SM 338 超声波传感器检测位置，无磨损、保护等级高、精度稳定不变。

3．闭环控制模块

FM 355 闭环控制模块有 4 个闭环控制通道，有自优化温度控制算法和 PID 算法。

4．称重模块

SIWAREX U 称重模块是紧凑型电子称，测定料仓和贮斗的料位，对吊车载荷进行监控，对传送带载荷进行测量或对工业提升机、轧机超载进行安全防护等。

SIWAREX M 称重模块是有校验能力的电子称重和配料单元，可以组成多料称系统，安装在易爆区域。

5．电源模块

PS 307 电源模块将 120/230 伏交流电压转换为 24V 直流电压，为 S7-300/400、传感器和执行器供电。输出电流有 2A、5A 或 10A 3 种。电源模块安装在 DIN 导轨上的插槽 1。

图 2-17 S7-300 的浮动参考电位

某些大型工厂（例如化工厂和发电厂）为了监视对地的短路电流，可能采用浮动参考电位，可以将 M 点与接地点之间的短接片去掉。

2.5 S7-400 系列 PLC 的硬件组成

模块的尺寸为 25(宽)×290(高)×210(深)mm。高 炉 喷 煤 工 程构成图。

集中式扩展方式适用于小型配置或一个控制柜中的系统。CC 和 EU 的最大距离为 1.5m（带 5V 电源）或 3m（不带 5V 电源）。

分布式扩展适用于分布范围广的场合，CC 与最后一个 EU 的最大距离为 100m（S7 EU）或 600m（S5 EU）。

用 ET 200 分布式 I/O 可以进行远程扩展，用于分布范围很广的系统。通过 CPU 中的

PLC WORLD

PROFIBUS-DP 接口，最多连接 125 个总线节点。使用光缆时 CC 和最后一个节点的距离为 23km。

2．S7-400 的特点

（1）运行速度高，S7 416 执行一条二进制指令只要 0.08ms。

（2）存储器容量大，例如 CPU 417-4 的 RAM 可以扩展到 16MB，装载存储器（FEPROM 或 RAM）可以扩展到 64MB。

（3）I/O 扩展功能强，可以扩展 21 个机架，S7 417-4 最多可以扩展 262144 个数字量 I/O 点和 16384 个模拟量 I/O。

（4）有极强的通信能力，集成的 MPI 能建立最多 32 个站的简单网络。大多数 CPU 集成有 PROFIBUS-DP 主站接口，用来建立高速的分布式系统, 通信速率最高 12M bit/s。锅炉系统构成图

2.5.2 机架与接口模块

（1）通用机架 UR1/UR2

（2）中央机架，CR2 是 18 槽,一个电源模块和两个 CPU 模块。

CR3 是 4 槽的中央机架，有 I/O 总线和通信总线。

图 2-17 机架与总线

（3）扩展机架 ER1/ER2

ER1 和 ER2 是扩展机架，分别有 18 槽和 9 槽，只有 I/O 总线。

（4）UR2-H 机架

UR2-H 机架用于在一个机架上配置一个完整的 S7-400H 冗余系统，每个均有自己的 I/O。两个电源模块和两个冗余 CPU 模块。

2.5.3 S7-400 的通信功能

MPI、PROFIBUS-DP、工业以太网或 AS-i 现场总线，周期性自动交换 I/O 模块的数据。或基于事件驱动，由用户程序块调用。

2.5.4 冗余设计的容错自动化系统 S7-400H

S7 Software Redundancy（软件冗余性）可选软件在 S7-300 和 S7-400 标准系统上运行。生产过程出现故障时，在几秒内切换到替代系统。

S7-400H 主要器件都是双重的：CPU、电源模块以及连接两个 CPU 的硬件……

3．S7-400H 冗余控制 PLC 的工作原理

S7-400H 采用“热备用”模式的主动冗余原理，在发生故障时，无扰动地自动切换。

两个控制器使用相同的用户程序，接收相同数据，两个控制器同步地更新内容，任意一个子系统有故障时，另一个承担全部控制任务。

2.5.5 安全型自动化系统 S7-400F/FH

S7-400F/FH 使用标准模块和安全型模块，整个工厂用相同的标准工具软件来配置和编程。

2.5.6 多 CPU 处理

S7-400 中央机架上最多 4 个具有多 CPU 处理能力的 CPU 同时运行。这些 CPU 自动地、同步地变换其运行模式。

适用场合：程序太长，存储空间不够，系统可以分。通过通信总线，CPU 彼此互连。

2.5.9 输入/输出模块

S7-400 的信号模块地址在 STEP 7 中自动生成。用户可以修改。

S7-400 的模拟量模块起始地址从 512 开始，同类模块的地址按顺序连续排列。

表 2-5 模块地址举例

PLC WORLD

0 号机架　1 号机架
槽号　模块种类　地址　槽号　模块种类　地址
1　PS 417 10A 电源模块　　1　32 点 DI　IB4～IB7
2　　2　16 点 DO　QB2，QB3
3　CPU 412-2DP　　3　16 点 DO　QB4，QB5
4　16 点 DO　QB0，QB1　4　8 点 AO　QW528～QW543
5　16 点 DI　IB0，IB1　5　8 点 AI　IW544～IW559
6　8 点 AO 2 个字节　QW512～QW527　6　16 点 DO　QB6，QB7
7　16 点 AI2 个字节　IW512～IW543　7　8 点 AI　IW560～IW575
8　16 点 DI　IB2，IB3　8　32 点 DI　IB8～IB11
9　IM460-1　4093　9　IM461-0　4092
表 2-6 S7-300 与 S7-400 性能比较接近的功能模块
功能模块　S7-300 系列　S7-400 系列
计数器模块　FM 350-1　FM 450-1
定位模块　FM 351，双通道　FM 451，3 通道
定位模块　FM 353，双通道　FM 453，3 通道
电子凸轮控制器　FM 352，13 个数字量输出　FM 452，16 个数字量输出
闭环控制模块　FM 355，4 通道　FM 455，16 通道
2.6 ET 200 分布式 I/O
基于 PROFIBUS-DP 现场总线的分布式 I/O。I/O 传送信号到 CPU 只需 ms 级。
只需要很小的空间，能在非常严酷的环境（例如酷热、严寒、强压、潮湿或多粉尘）中使用。
（1）电机启动器：异步电机的单向或可逆启动，7.5kW，最大电流 40A，一个站可以带 6 个电机启动器。
（2）气动系统：ET 200X 用于阀门控制。
（3）变频器
（4）智能传感器：光电式编码器或光电开关等与使用 ET 200S 进行通信。
（5）安全技术：在冗余设计的容错控制系统或安全自动化系统中使用。包括紧急断开开关，安全门的监控以及众多与安全有关的电路。有 ET 200S 故障防止模块、故障防止 CPU 和 PROFISafe 协议。

2.6.2 ET 200 的分类
(1) ET 200S 是分布式 I/O 系统。
(2) ET 200M 是模块化的分布式 I/O，采用 S7-300 全系列模块，最多 8 个模块。
ET 200M 户外型温度范围-25°C 到+60°C。
(3) ET 200is 是本质安全系统，适用于有爆炸危险的区域。
(4) ET 200X：IP65/67 的分布式 I/O，相当于 CPU 314，可用于有粉末和水流喷溅的场合。
(5) ET 200eco 是经济实用的 I/O，IP67。
(6) ET 200R 适用于机器人，能抗焊接火花的飞溅。
(7) ET 200L 是小巧经济的分布式 I/O，像明信片大小的 I/O 模块。
(8) ET 200B：整体式的一体化分布式 I/O。

3.1 S7-300/400 的编程语言
3.1.1 PLC 编程语言的国际标准
IEC 61131 是 PLC 的国际标准，1992～1995 年发布了 IEC 61131 标准中的 1～4 部分，我

PLC WORLD

国在 1995 年 11 月发布了 GB/T 15969-1/2/3/4(等同于 IEC 61131-1/2/3/4)。
IEC 61131-3 广泛地应用 PLC、DCS 和工控机、“软件 PLC”、数控系统、RTU 等产品。
定义了 5 种编程语言
1) 指令表 IL(Instruction list)：西门子称为语句表 STL。
2) 结构文本 ST(Structured text)：西门子称为结构化控制语言（SCL）。
3) 梯形图 LD(Ladder diagram)：西门子简称为 LAD。
4) 功能块图 FBD (Function block diagram)：标准中称为功能方框图语言。
5) 顺序功能图 SFC(Sequential function chart)：对应于西门子的 S7 Graph。

3.1.2 STEP 7 中的编程语言
梯形图、语句表和功能块图是 3 种基本编程语言，可以相互转换。
1．顺序功能图(SFC)：STEP 7 中的 S7 Graph
2．梯形图(LAD)
直观易懂，适合于数字量逻辑控制。“能流”(Power flow)与程序执行的方向。
3. 语句表(STL)：功能比梯形图或功能块图强。

4．功能块图(FBD)：“LOGO！”系列微型 PLC 使用功能块图编程。
5．结构文本(ST)：STEP 7 的 S7 SCL（结构化控制语言）符合 EN 61131-3 标准。
SCL 适合于复杂的公式计算、复杂的计算任务和最优化算法，或管理大量的数据等。
6．S7 HiGraph 编程语言
图形编程语言 S7 HiGraph 属于可选软件包，它用状态图（state graphs）来描述异步、非顺序过程的编程语言。
7．S7 CFC 编程语言
可选软件包 CFC（Continuous Function Chart，连续功能图）用图形方式连接程序库中以块的形式提供的各种功能。
8．编程语言的相互转换与选用
在 STEP 7 编程软件中，如果程序块没有错误，并且被正确地划分为网络，在梯形图、功能块图和语句表之间可以转换。如果部分网络不能转换，则用语句表表示。
语句表可供喜欢用汇编语言编程的用户使用。语句表的输入快，可以在每条语句后面加上注释。设计高级应用程序时建议使用语句表。
梯形图适合于熟悉继电器电路的人员使用。设计复杂的触点电路时最好用梯形图。
功能块图适合于熟悉数字电路的人使用。
S7 SCL 编程语言适合于熟悉高级编程语言（例如 PASCAL 或 C 语言）的人使用。
S7 Graph，HiGraph 和 CFC 可供有技术背景，但是没有 PLC 编程经验的用户使用。
S7 Graph 对顺序控制过程的编程非常方便，HiGraph 适合于异步非顺序过程的编程，CFC 适合于连续过程控制的编程。
3.2 S7-300/400 CPU 的存储区
3.2.1 数制
1．二进制数
二进制数的 1 位（bit）只能取 0 和 1 这两个不同的值，用来表示开关量的两种不同的状态。该位的值与线圈、触点的关系。ON/OFF，TURE/FALSE。二进制常数：
2#1111_0110_1001_0001。
2．十六进制数
十六进制的 16 个数字是 0～9 和 A～F, 每个占二进制数的 4 位。B#16#，W#16#，

PLC WORLD

DW#16#， W#16#13AF（13AFH）。逢 16 进 1，例如 B#16#3C＝3×16＋12＝60。
3．BCD 码
BCD 码用 4 位二进制数表示一位十进制数，十进制数 9 对应的二进制数为 1001。
最高 4 位用来表示符号，16/32 位 BCD 码的范围。BCD 码实际上是十六进制数，但是各位之间逢十进一。296 对应的 BCD 码为 W#16#296，或 2#0000 0010 1001 0110。
2#0000 0001 0010 1000 对应的十进制数也是 296，对应的十进制数为
。
3.2.2 基本数据类型

1．位（bit）：位数据的数据类型为 BOOL（布尔）型。I3.2 的意义。
2．字节（Byte）
3．字(Word)表示无符号数。取值范围为 W#16#0000～W#16#FFFF。
4．双字（Double Word）表示无符号数。范围 DW#16#0000_0000～DW#16#FFFF_FFFF。
5．16 位整数（INT，Integer）是有符号数，补码。最高位为符号位，为 0 时为正数，取值范围为-32 768～32 767。
6．32 位整数（DINT，Double Integer）最高位为符号位，取值范围为 MB100 M 表示内部存储区
-2 147 483 648～2 147 483 647。

图 3-6 字节、字和双字

3.2.3 复合数据类型与参数类型
1．复合数据类型
通过组合基本数据类型和复合数据类型可以生成下面的数据类型：
(1) 数组（ARRAY）
(2) 结构（STRUCT）
(3) 字符串（STRING）是最多有 254 个字符（CHAR）的一维数组。
(4) 日期和时间（DATE_AND_TIME）用于存储年、月、日、时、分、秒、毫秒和星期，占用 8 个字节，用 BCD 格式保存。星期天的代码为 1，星期一～星期六的代码为 2～7。
例如 DT#2004-07-15-12:30:15.200 为 2004 年 7 月 15 日 12 时 30 分 15.2 秒。
(5) 用户定义的数据类型 UDT (user-defined data types)。
在数据块 DB 和逻辑块的变量声明表中定义复合数据类型。
2．参数类型
为在逻辑块之间传递参数的形参（formal parameter，形式参数）定义的数据类型：
(1) TIMER（定时器）和 COUNTER（计数器）：对应的实参（actual parameter，实际参数）应为定时器或计数器的编号，例如 T3，C21。
(2) BLOCK（块）：指定一个块用作输入和输出，实参应为同类型的块。
3.2.5 系统存储器 （存储器 RAM = 用户 RAM + 系统 RAM）
1．过程映像输入/输出（I/Q）
在扫描循环开始时，CPU 读取数字量输入模块的输入信号的状态，并将它们存入 RAM 中过程映像输入（process image input，PII）中。
在扫描循环中，用户程序计算输出值，并将它们存入过程映像输出表
（process image output，PIQ）。在循环扫描结束时将过程映像输出表的内容写入数字量输出模块。

PLC WORLD

I 和 Q 均以按位、字节、字和双字来存取，例如 I0.0，Q4.0、IB0, IW0 和 ID0。
与直接访问 I/O 模块相比的优缺点。

2．内部存储器标志位（M）存储器区　MB1
3．定时器（T）存储器区
时间值可以用二进制或 BCD 码方式读取。
4．计数器（C）存储器区
计数值（0～999）可以用二进制或 BCD 码方式读取。
5．共享数据块（DB）与背景数据块（DI）
DB 为共享数据块，DBX2.3，DBB5，DBW10 和 DBD12。
DI 为背景数据块，DIX, DIB，DIW 和 DID。
6．外设 I/O 区（PI/PO）
　外设输入（PI）和外设输出（PQ）区允许直接访问本地的和分布式的输入模块和输出模块。可以按字节（PIB 或 PQB）、字（PIW 或 PQW）或双字（PID 或 PQD）存取，不能以位为单位存取 PI 和 PO。
3.2.6 CPU 中的寄存器
1．累加器（ACCUx）
累加器用于处理字节、字或双字的寄存器。S7-300 有两个 32 位累加器（ACCU1 和 ACCU2），S7-400 有 4 个累加器（ACCU1～ACCU4）。数据放在累加器的低端（右对齐）。
2．状态字寄存器（16 位）
首次检测位/FC, 逻辑运算结果（RLO）；
状态位 STA 不能用指令检测；
OR 位暂存逻辑“与”的操作结果（先与后或）；
算术运算或比较指令执行时出现错误，溢出位 OV 被置 1。
OV 位被置 1 时溢出状态保持位 OS 位也被置 1，OV 位被清 0 时 OS 仍保持为 1，用于指明前面的指令执行过程中是否产生过错误。
条件码 1（CC1）和条件码 0（CCO）综合起来用于表示在累加器 1 中产生的算术运算或逻辑运算的结果与 0 的大小关系、比较指令的执行结果或移位指令的移出位状态。
二进制结果位（BR）在一段既有位操作又有字操作的程序中，用于表示字操作结果是否正确。在梯形图的方框指令中，BR 位与 ENO 有对应关系，用于表明方框指令是否被正确执行：如果执行出现了错误，BR 位为 0，ENO 也为 0；如果功能被正确执行，BR 位为 1，ENO 也为 1。

图 3-9 状态字的结构

3．数据块寄存器：DB 和 DI 寄存器分别用来保存打开的共享数据块和背景数据块的编号。
3.3 位逻辑指令
　位逻辑指令用于二进制数的逻辑运算。位逻辑运算的结果简称为 RLO。
3.3.1 触点指令
1．　触点与线圈
A（And，与）指令来表示串联的常开触点。
O (Or，或)指令来表示并联的常开触点。
AN (And Not，与非)来表示串联的常闭触点，
ON (Or Not)来表示并联的常闭触点。

PLC WORLD

输出指令“=”将 RLO 写入地址位，与线圈相对应。L20.0 是局域变量。将梯形图转换为语句表时，局域变量 L20.0 是自动分配的。

```
A(
      A      I 0.0
      AN     I 0.1
      O      I 0.2
      )
      A      I 0.3
      ON     C 5
      =      L 20.0
      A      L 20.0
      =      Q 4.3
      A      L 20.0
      =      Q 4.4
      A      L 20.0
      AN     I 3.4
=     Q 4.6
```

2. 取反触点

3. 电路块的串联和并联

4. 中线输出指令 下面是图 3-14(b)中第一行对应的语句表。

```
    A     I0.0
   AN     I0.1
   =      M0.1
   A      M0.1
   A      I0.3
   =      Q4.3
```

```
Network 1：
A     I0.3
A     I0.0
FP
=     Q4.5
Network 2：
A     I0.3
```

PLC WORLD

```
A       I0.0
FN
=       Q4.3

A       I0.3
A(
A       I0.4
BLD     100
FN      M0.1
     )
  =       Q4.5
```

【例 3-1】设计故障信息显示电路，故障信号 I0.0 为 1 使 Q4.0 控制的指示灯以 1Hz 的频率闪烁。操作人员按复位按钮 I0.1 后，如果故障已经消失，指示灯熄灭。如果没有消失，指示灯转为常亮，直至故障消失。

设置 CPU 的属性时，在“Cycle/Clock Memory”标签页令 M1 为时钟存储器字节，其中的 M1.5 提供周期为 1s 的时钟脉冲。

SET 与 CLR（Clear）指令将 RLO（逻辑运算结果）置位或复位，紧接在它们后面的赋值语句中的地址将变为 1 状态或 0 状态。

```
SET         //将 RLO 置位
= M0.2      //M0.2 的线圈“通电”
CLR         //将 RLO 复位
= Q4.7      //Q4.7 的线圈“断电”
```

3.4.1 定时器指令

在 CPU 内部，时间值以二进制格式存放，占定时器字的 0～9 位。

可以按下列的形式将时间预置值装入累加器的低位字：

(1) 十六进制数 W#16#wxyz，其中的 w 是时间基准，xyz 是 BCD 码形式的时间值。

(2) S5T#aH_bM_cS_Dms，例如 S5T#18S。

时基代码为二进制数 00，01，10 和 11 时，对应的时基分别为 10ms，100ms，1s 和 10s。

6．脉冲定时器

类似于上升沿触发的单稳态电路。

S5 脉冲定时器(Pulse S5 Timer)，S 为设置输入端，TV 为预置值输入端，R 为复位输入端；Q 为定时器位输出端，BI 输出不带时基的十六进制格式，BCD 输出 BCD 格式的当前时间值和时基。

定时器中的 S，R，Q 为 BOOL（位）变量，BI 和 BCD 为 WORD（字）变量，TV 为 S5TIME 量。各变量均可以使用 I, Q, M, L, D 存储区, TV 也可以使用定时时间常数 S5T#。

```
A       I 1.2
FR      T0      //允许定时器 T1 再起动
A       I 0.0
```

PLC WORLD

```
L       S5T#2s    //预置值 2s 送入累加器 1
SP       T0       //启动 T0
A       I 0.1
R       T0       //复位 T0
L       T0       //将 T0 的十六进制时间当前值装入累加器 1
T       MW10     //将累加器 1 的    内容传送到 MW10
LC       T0      //将 T0 的 BCD 时间当前值装入累加器 1.
T       MW12     //将累加器 1 的内容传送到 MW12
A       T0       //检查 T0 的信号状态
=       Q 4.0    //T0 的定时器位为 1 时，Q4.0 的线圈通电
```

仅在语句表中使用的 FR 指令允许定时器再起动，即控制 FR 的 RLO（I1.2）由 0 变为 1 状态时，重新装入定时时间，定时器又从预置值开始定时。再起动只是在定时器的起动条件满足(图 3-28 中的 I0.1=1)时起作用。该指令可以用于所有的定时器，但是它不是起动定时器定时的必要条件。

8．扩展的脉冲定时器

10．接通延时定时器

12．保持型接通延时定时器

14．断开延时定时器线圈

3.4.2 计数器指令

1．计数器的存储器区

每个计数器有一个 16 位的字和一个二进制位。

计数器字的 0～11 位是计数值的 BCD 码，计数值的范围为 0～999。二进制格式的计数值只占用计数器字的 0～9 位。

下面是图 3-44 中左边的电路对应的语句表：

```
A       I0.0      //在 I0.0 的上升沿
CU       C10      //加计数器 C10 的当前值加 1
BLD     101
A       I0.2      //在 I0.2 的上升沿
L       C#6       //计数器的预置值 6 被装入累加器的低字
S       C10       //将预置值装入计数器 C10
A       I0.3      //如果 I0.3 为 1
R       C10       //复位 C10
```

PLC WORLD

```
L       C10       //将 C10 的二进制计数当前值装入累加器 1
T       MW0       //将累加器 1 的内容传送到 MW0
LC      C10       //将 C10 的 BCD 计数当前值装入累加器 1
T       MW8       //将累加器 1 的内容传送到 MW8
A       C10       //如果 C10 的当前值非 0
=       Q 5.0     //Q 5.0 为 1 状态
```

设置计数值线圈 SC(Set Counter Value)用来设置计数值，在 RLO 的上升沿预置值被送入指定的计数器。CU 的线圈为加计数器线圈。在 I0.0 的上升沿，如果计数值小于 999，计数值加 1。复位输入 I0.3 为 1 时，计数器被复位，计数值被清 0。

计数值大于 0 时计数器位（即输出 Q）为 1；计数值为 0 时，计数器位亦为 0。

在减计数输入信号 CD 的上升沿，如果计数值大于 0，计数值减 1。

3.5.1 装入指令与传送指令

1．装入指令与传送指令

装入（L，Load）指令将源操作数装入累加器 1，而累加器 1 原有的数据移入累加器 2。

装入指令可以对字节（8 位）、字（16 位）、双字（32 位）数据进行操作。

传送（T，Transfer）指令将累加器 1 中的内容写入目的存储区中，累加器 1 的内容不变。

2．立即寻址的装入与传送指令

立即寻址的操作数直接在指令中，下面是使用立即寻址的例子。

```
L   -35             //将 16 位十进制常数-35 装入累加器 1 的低字 ACCU1-L
L   L#5             //将 32 位常数 5 装入累加器 1
L   B#16#5A         //将 8 位十六进制常数装入累加器 1 最低字节 ACCU1-LL
L   W#16#3E4F       //将 16 位十六进制常数装入累加器 1 的低字 ACCU1-L
L   DW#16#567A3DC8  //将 32 位十六进制常数装入累加器 1
L   2#0001_1001_1110_0010  //将 16 位二进制常数装入累加器 1 的低字 ACCU1-L
L   25.38           //将 32 位浮点数常数(25.38)装入累加器 1
L   'ABCD'          //将 4 个字符装入累加器 1
L   TOD#12:30:3.0   //将 32 位实时时间常数装入累加器 1
L   D#2004-2-3      //将 16 位日期常数装入累加器 1 的低字 ACCU1-L
L   C#50            //将 16 位计数器常数装入累加器 1 的低字 ACCU1-L
L   T#1M20S         //将 16 位定时器常数装入累加器 1 的低字 ACCU1-L
L   S5T#2S          //将 16 位定时器常数装入累加器 1 的低字 ACCU1-L
L   P#M5.6          //将指向 M5.6 的指针装入累加器 1
AW  W#16#3A12       //常数与累加器 1 的低字相"与"，运算结果在累加器 1 的低字中
L   B#(100,12,50,8) //装入 4 字节无符号常数
```

3．直接寻址的装入与传送指令

直接寻址在指令中直接给出存储器或寄存器的区域、长度和位置，例如用 MW200 指定位存储区中的字，地址为 200；下面是直接寻址的程序实例：

```
A       I0.0        //输入位 I0.0 的"与"（AND）操作
    L       MB10    //将 8 位存储器字节装入累加器 1 最低的字节 ACCU1-LL
L       DIW15       //将 16 位背景数据字装入累加器 1 的低字 ACCU1-L
L       LD22        //将 32 位局域数据双字装入累加器 1
T       QB10        //将 ACCU1-LL 中的数据传送到过程映像输出字节 QB10
T       MW14        //将 ACCU1-L 中的数据传送到存储器字 MW14
```

PLC WORLD

T DBD2 //将 ACCU1 中的数据传送到数据双字 DBD2

3．存储器间接寻址

在存储器间接寻址指令中，给出一个作地址指针的存储器，该存储器的内容是操作数所在存储单元的地址。在循环程序中经常使用存储器间接寻址。

地址指针可以是字或双字，定时器（T）、计数器（C）、数据块（DB）、功能块（FB）和功能（FC）的编号范围小于 65 535，使用字指针就够了。

其它地址则要使用双字指针，如果要用双字格式的指针访问一个字、字节或双字存储器，必须保证指针的位编号为 0，例如 P#Q20.0。

L QB[DBD 10] //将输出字节装入累加器 1，输出字节的地址指针在数据双字 DBD10 中

//如果 DBD10 的值为 2＃0000 0000 0000 0000 0000 0000 0010 0000，装入的是 QB4

A M[LD 4] //对存储器位作“与”运算，地址指针在数据双字 LD4 中

//如果 LD4 的值为 2＃0000 0000 0000 0000 0000 0000 0010 0011，则是对 M4.3 进行操作

4．寄存器间接寻址

地址寄存器 AR1 和 AR2，的内容加上偏移量形成地址指针，指向数值所在的存储单元。

其中第 0～2 位（xxx）为被寻址地址中位的编号（0～7），第 3～18 位为被寻址地址的字节的编号（0～65535）。第 24～26 位（rrr）为被寻址地址的区域标识号，第 31 位 x = 0 为区域内的间接寻址，第 31 位 x = 1 为区域间的间接寻址。

第一种地址指针格式存储区的类型在指令中给出，例如 L DBB[AR1, P#6.0]。在某一存储区内寻址。第 24～26 位（rrr）应为 0。

第二种地址指针格式的第 24～26 位还包含存储区域标识符 rrr，区域间寄存器间接寻址。

如果要用寄存器指针访问一个字节、字或双字，必须保证指针中的位地址编号为 0。

指针常数＃P5.0 对应的二进制数为 2＃0000 0000 0000 0000 0000 0000 0010 1000。下面是区内间接寻址的例子：

L P#5.0 //将间接寻址的指针装入累加器 1

LAR1 //将累加器 1 中的内容送到地址寄存器 1

A M[AR1, P#2.3] //AR1 中的 P#5.0 加偏移量 P#2.3, 实际上是对 M7.3 进行操作

= Q[AR1, P#0.2] //逻辑运算的结果送 Q5.2

L DBW[AR1, P#18.0] //将 DBW23 装入累加器 1

下面是区域间间接寻址的例子：

L P#M6.0 //将存储器位 M6.0 的双字指针装入累加器 1

LAR1 //将累加器 1 中的内容送到地址寄存器 1

T W[AR1, P#50.0] //将累加器 1 的内容传送到存储器字 MW56

P#M6.0 对应的二进制数为 2#1000 0011 0000 0000 0000 0000 0011 0000。因为地址指针 P#M6.0 中已经包含有区域信息，使用间接寻址的指令 T W[AR1, P#50]中没有必要再用地址标识符 M。

表 3-6 寄存器间接寻址的区域标识位

区域标识符	存储区	位 26～24
P	外设输入输出	000
I	输入过程映像	001
Q	输出过程映像	010

PLC WORLD

M　位存储区　011
DBX　共享数据块　100
DIX　背景数据块　101
L　块的局域数据　111
5．装入时间值或计数值

```
L       T5          //将定时器 T5 中的二进制时间值装入累加器 1 的低字中
LC      T5          //将定时器 T5 中的 BCD 码格式的时间值装入累加器 1 低字中
L       C3          //将计数器 C3 中的二进制计数值装入累加器 1 的低字中
LC      C16         //将计数器 C16 中的 BCD 码格式的值装入累加器 1 的低字中
```

6．地址寄存器的装入与传送指令
可以不经过累加器 1，与地址寄存器 AR1 和 AR2 交换数据。下面是应用实例：

```
LAR1    DBD20       //将数据双字 DBD20 中的指针装入 AR1
LAR2    LD180       //将局域数据双字 LD180 中的指针装入 AR2
LAR1    P#M10.2     //将带存储区标识符的 32 位指针常数装入 AR1
LAR2    P#24.0      //将不带存储区标识符 32 位指针常数装入 AR2
TAR1    DBD20       //AR1 中的内容传送到数据双字 DBD20
TAR2    MD24        //AR2 中的内容传送到存储器双字 MD24
```

梯形图中的传送指令：

```
A       I1.0
    JNB     _001    //如果 I1.0 = 0，则跳转到标号_001 处
    L       MW2     //MW2 的值装入累加器 1 的低字
    T       MW4     //累加器 1 低字的内容传送到 MW4
    SET             //将 RLO 置为 1
    SAVE            //将 RLO 保存到 BR 位
    CLR             //将 RLO 置为 0
_001:  A        BR
    ……
```

如果功能被正确执行，BR 位为 1，ENO 也为 1。
3.5.2 比较指令
比较指令用于比较累加器 1 与累加器 2 中的数据大小，被比较的两个数的数据类型应该相同。如果比较的条件满足，则 RLO 为 1，否则为 0。状态字中的 CC0 和 CC1 位用来表示两个数的大于、小于和等于关系（见表 3-7）。
表 3-7　指令执行后的 CC1 和 CC0

CC1	CC0	比较指令	移位和循环移位指令	字逻辑指令
0	0	累加器 2＝累加器 1	移出位为 0	结果为 0
0	1	累加器 2<累加器 1	—	—
1	0	累加器 2>累加器 1	—	结果不为 0
1	1	非法的浮点数	移出位为 1	—

表 3-8　比较指令

语句表指令	梯形图中的符号	说明
? I? D? R	CMP ? I CMP ? D CMP ? R	比较累加器 2 和累加器 1 低字中的整数，如果条件满足，RLO=1 比较累加器 2 和累加器 1 中的双整数，如果条件满足，RLO=1 比较累加器

PLC WORLD

2 和累加器 1 中的浮点数，如果条件满足，RLO=1
？可以是==, <>, >, <, >=, <=。
下面是比较两个浮点数的例子：

```
L     MD4          //MD4 中的浮点数装入累加器 1
L     2.345E+02    //浮点数常数装入累加器 1，MD4 装入累加器 2
>R                 //比较累加器 1 和累加器 2 的值
=     Q4.2         //如果 MD4 > 2.345E+02，则 Q4.2 为 1
```

梯形图中的方框比较指令可以比较整数（I）、双整数（D）和浮点数（R）。方框比较指令在梯形图中相当于一个常开触点，可以与其他触点串连和并联。

表 3-9 数据转换指令

语句表	梯形图	说明
BTI	IT_BCD	将累加器 1 中的 3 位 BCD 码转换成整数
BTD	DT_BCD	将累加器 1 中的整数转换成 3 位 BCD 码
DTB	DI_BCD	将累加器 1 中的 7 位 BCD 码转换成双整数
DTR	DI_R	将累加器 1 中的双整数转换成 7 位 BCD 码
ITD	DI_DI	将累加器 1 中的双整数转换成浮点数
RND	ROUND	将累加器 1 中的整数转换成双整数
RND+	CEIL	将浮点数转换为四舍五入的双整数
RND-	FLOOR	将浮点数转换为大于等于它的最小双整数
TRUNC	TRUNC	将浮点数转换为小于等于它的最大双整数
CAW	CAW	将浮点数转换为截位取整的双整数
CAD	——	交换累加器 1 低字中两个字节的位置
		交换累加器 1 中 4 个字节的顺序

下面是双整数转换为 BCD 码的例子：

```
A     I0.2          //如果 I0.2 为 1
L     MD10          //将 MD10 中的双整数装入累加器 1
DTB                 //将累加器 1 中的数据转换为 BCD 码，结果仍在累加器 1 中
JO    OVER          //运算结果超出允许范围（OV＝1）则跳转到标号 OVER 处
T     MD20          //将转换结果传送到 MD20
A     M4.0
R     M4.0          //复位溢出标志
JU    NEXT          //无条件跳转到标号 NEXT 处
OVER:   AN    M4.0
        S     M4.0  //置位溢出标志
NEXT:   ……
```

【例 3-5】将 101 英寸转换为以厘米为单位的整数，送到 MW0 中。

```
L       101         //将 16 位常数 101(65H)装入累加器 1
ITD                 //转换为 32 位双整数
DTR                 //转换为浮点数 101.0
L       2.54        //浮点数常数 2.54 装入累加器 1，累加器 1 的内容装入累加器 2
*R                  //101.0 乘以 2.54，转换为 256.54 厘米
RND                 //四舍五入转换为整数 257(101H)
T       MW30
```

7．取反与求补指令

表 3-12 取反与求补指令

PLC WORLD

语句表指令　梯形图指令　说明
INVIINVDNEGINEGDNEGR　INV_IINV_DINEG_INEG_DINEG_R　求累加器 1 低字中的 16 位整数的反码求累加器 1 中双整数的反码求累加器 1 低字中的 16 位整数的补码求累加器 1 中双整数的补码将累加器 1 中的浮点数的符号位取反

```
L     MD20      //将 32 位双整数装入累加器 1
NEGD            //求补
T     MD30      //运算结果传送到 MD30
```

表 3-13 取反与求补
内容　累加器 1 的低字
变换前的数　0101 1101 0011 1000
取反的结果　1010 0010 1100 0111
求补的结果　1010 0010 1100 1000
3.6.1 整数数学运算指令

```
L    IW10       //IW10 的内容装入累加器 1 的低字
L    MW14       //累加器 1 的内容装入累加器 2，MW14 的值装入累加器 1 的低字
/I              //累加器 2 低字的值除以累加器 1 低字的值，结果在累加器 1 的低字
T    DB1.DBW2   //累加器 1 低字中的运算结果传送到数据块 DB1 的 DBW2 中
```

表 3-16 整数数学运算指令
语句表　梯形图　描　述
+I -I *I / I ++D -D *D / D MOD　ADD_ISUB_IMUL_IDIV_I——ADD_DISUB_DIMUL_DIDIV_DIMOD_DI　将累加器 1，2 低字中的整数相加，运算结果在累加器 1 的低字中累加器 2 中的整数减去累加器 1 中的整数，运算结果在累加器 1 的低字将累加器 1，2 低字中的整数相乘，32 位双整数运算结果在累加器 1 中累加器 2 的整数除以累加器 1 的整数，商在累加器 1 的低字，余数在累加器 1 的高字累加器的内容与 16 位或 32 位常数相加，运算结果在累加器 1 中将累加器 1，2 中的双整数相加，双整数运算结果在累加器 1 中累加器 2 中的双整数减去累加器 1 中的双整数运算结果在累加器 1 中将累加器 1，2 中的双整数相乘，32 位双整数运算结果在累加器 1 中累加器 2 中的双整数除以累加器 1 中的双整数，32 位商在累加器 1 中， 累加器 2 中的双整数除以累加器 1 中的双整数，32 位余数在累加器 1 中
3.6.2 浮点数数学运算指令
表 3-17 浮点数数学指令
语句表　梯形图　描述
+R-R*R/RABSSQRSQRTEXPLNSINCOSTANASINACOSATAN　ADD_RSUB_RMUL_RDIV_RABSSQRSQRTEXPLNSINCOSTANASINACOSATAN　将累加器 1，2 中的浮点数相加，浮点数运算结果在累加器 1 中累加器 2 中的浮点数减去累加器 1 中的浮点数，运算结果在累加器 1 中将累加器 1，2 中的浮点数相乘，浮点数乘积在累加器 1 中累加器 2 中的浮点数除以累加器 1 中的浮点数，商在累加器 1，余数丢掉取累加器 1 中的浮点数的绝对值求浮点数的平方求浮点数的平方根求浮点数的自然指数求浮点数的自然对数求浮点数的正弦函数求浮点数的余弦函数求浮点数的正切函数求浮点数的反正弦函数求浮点数的反余弦函数求浮点数的反正切函数

```
OPN   DB17      //打开数据块 DB17
```

PLC WORLD

```
L       DBD0 //数据块 DB17 的 DBD0 中的浮点数装入累加器 1
SQR//求累加器 1 中的浮点数的平方，运算结果在累加器 1 中
AN      OV          //如果运算时没有出错
JC      OK          //跳转到标号 OK 处
BEU                 //如果运算时出错，功能块无条件结束
OK:   T       DBD4      //累加器 1 中的运算结果传送到数据块 DB17 的 DBD4 中
```

求以 10 为底的对数时，应将自然对数值除以 2.302585(10 的自然对数值)。例如
lg100＝ln100/2.302585＝4.605170/2.302585=2
【例 3-6】用浮点数对数指令和指数指令求 5 的立方。计算公式为：

```
L       L#5
DTR
LN
L       3.0
*R
EXP
RND
T       MW40
```

浮点数三角函数指令的输入值为弧度，角度值乘以 p/180，可转换为弧度值。

【例 3-7】压力变送器的量程为 0～10MPa，输出信号为 4～20mA，S7-300 的模拟量输入模块的量程为 4～20mA，转换后的数字量为 0～27 648，设转换后的数字为 N，试求以 kPa 为单位的压力值。
　解：0～10MPa(0～10 000kPa)对应于转换后的数字 0～27 648，转换公式为
P =（10 000 ´ N）/ 27 648 （kPa）　　(3-1)
值得注意的是在运算时一定要先乘后除，否则会损失原始数据的精度。假设 A/D 转换后的数据 N 在 MD6 中，以 kPa 为单位的运算结果在 MW10 中。图 3-58 是实现式（3-1）中的运算的梯形图程序。

图 3-58 算术运算指令
语句表中“*I”指令的运算结果为 32 位整数，梯形图中 MUL_I 指令的运算结果为 16 位整数。A/D 转换后的最大数字为 27 648，所以要使用 MUL_DI。双字除法指令 DIV_DI 的运算结果为双字，运算结果不会超过 16 位正整数的最大值（32 767）。
3.6.3 移位与循环移位指令
表 3-20 移位指令（对累加器 1 中的数操作，结果在累加器 1 中）
名称　语句表　梯形图　描述
有符号整数右移有符号双整数右移 16 位字左移 16 位字右移 16 位双字左移 16 位双字右移　SSISSDSLWSRWSLDSRD　SHR_ISHR_DISHL_WSHR_W SHL_DWSHR_DW
整数逐位右移，空出的位添上符号位双整数逐位右移，空出的位添上符号位字逐位左移，空出的位添 0 字逐位右移，空出的位添 0 双字逐位左移，空出的位添 0 双字逐位右移，空出的位添 0
双字循环左移双字循环右移双字＋CC1 循环左移双字＋CC1 循环右移　RLDRRDRLDARRDA　ROL_DWROR_DW－ －　双字循环左移双字循环右移双字

PLC WORLD

通过 CC1（一共 33 位）循环左移双字通过 CC1（一共 33 位）循环右移
（1）用指令中的参数<number>来指定移位位数，16 位移位指令为 0～15，32 位移位指令为 0～32。如果<number>等于 0，移位指令被当作 NOP（空操作）指令来处理。
（2）指令没有参数<number>，移位位数放在累加器 2 的最低字节中（0～255）。如果移位位数等于 0，移位指令被当作 NOP（空操作）指令来处理。
有符号字的移位位数>16 时，移位后被移位的数的各位全部变成了符号位。

```
L       MW4     //将 MW4 的内容装入累加器 1 的低字
SSI     6       //累加器 1 低字中的有符号数右移 6 位，结果仍在累加器 1 的低字中
T       MW8     //累加器 1 低字中的运算结果传送到 MW8 中
```

表 3-21 整数右移 6 位前后的数据

内容	累加器 1 的高字	累加器 1 的低字
移位前	0101 1111 0110 0100	1001 1101 0011 1011
右移 6 位后	0101 1111 0110 0100	1111 1110 0111 0100

```
L       +3      //将＋3 装入累加器 1
L       MW20    //将累加器 1 的内容装入累加器 2，MW20 的内容装入累加器 1
SSI             //累加器 1 低字中的有符号数右移 3 位
JP      NEXT    //如果最后移入 CC1 的位为 1，跳转到标号 NEXT 处
```

表 3-23 字右移 6 位移位前后的数据

内容	累加器 1 的高字	累加器 1 的低字
移位前	0101 1111 0110 0100	0101 1101 0011 1011
右移 6 位后	0101 1111 0110 0100	0000 0001 0111 0100

表 3-24 双字循环左移 4 位前后累加器中的数据

内容	累加器 1 的高字	累加器 1 的低字
移位前	0101 1111 0110 0100	0101 1101 0011 1011
右移 4 位后	1111 0110 0100 0101	1101 0011 1011 0101

表 3-25 双字通过 CC1 循环左移 1 位前后累加器中的数据

内容	CC1	累加器 1 的高字	累加器 1 的低字
移位前	X	0101 1111 0110 0100	0101 1101 0011 1011
左移后	0	1011 1110 1100 1000	1011 1010 0111 011X

图 3-60 有符号数右移指令

3.6.4 字逻辑运算指令

表 3-26 字逻辑运算指令

语句表	梯形图	描 述
AW	WAND_W	字与字
OW	WOR_W	字或字
XOW	WXOR_W	字异或
AD	WAND_DW	双字与双字
OD	WOR_DW	双字或双字
XOD	WXOR_DW	双字异或

表 3-27 字逻辑运算的结果

位	15	0
逻辑运算前累加器 1 的低字	0101 1001 0011 1011	
逻辑运算前累加器 2 的低字或常数	1111 0110 1011 0101	
“与”运算后累加器 1 的低字	0101 0000 0011 0001	
“或”运算后累加器 1 的低字	1111 1111 1011 1111	
“异或”运算后累加器 1 低字	1010 1111 1000 1110	

```
L     QW10      //QW10 的内容装入累加器 1 的低字
```

PLC WORLD

L　W#16#000F　//累加器 1 的内容装入累加器 2，W#16#000F 装入累加器 1 的低字
OW　　//累加器 1 低字与 W#16#000F 逐位相或，结果在累加器 1 的低字中
T　QW10　　//累加器 1 低字中的运算结果传送到 QW10 中

MB9 是 MW8 中的低字节，M9.1 和 M9.2 对应于输入信号 I0.1 和 I0.2。
3.6.5 累加器指令
表 3-28　累加器指令
语句表　描　述
TAKPUSHPOPENTLEAVEINCDEC+AR1+AR2BLDNOP 0NOP 1　交换累加器 1，2 的内容入栈出栈进入 ACCU 堆栈离开 ACCU 堆栈累加器 1 最低字节加上 8 位常数累加器 1 最低字节减去 8 位常数 AR1 的内容加上地址偏移量 AR2 的内容加上地址偏移量程序显示指令（空指令）空操作指令空操作指令

【例 3-9】用语句表程序实现浮点数运算（DBD0＋DBD4）/（DBD8–DBD12）。
L　DBD0　　//DBD0 中的浮点数装入累加器 1
L　DBD4　　//累加器 1 的内容装入累加器 2，DBD4 中的浮点数装入累加器 1
+R　　//累加器 1，2 中的浮点数相加，结果保存在累加器 1 中
L　DBD8　　//累加器 1 的内容装入累加器 2，DBD8 中的浮点数装入累加器 1
ENT　　//累加器 3 的内容装入累加器 4，累加器 2 的中间结果装入累加器 3
L　DBD12　　//累加器 1 的内容装入累加器 2，DBD12 中的浮点数装入累加器 1
–R　　//累加器 2 的内容减去累加器 1 的内容，结果保存在累加器 1 中
LEAVE　　//累加器 3 的内容装入累加器 2，累加器 4 的中间结果装入累加器 3
/R　　//累加器 2 的（DBD0＋DBD4）除以累加器 1 的（DBD8–DBD12）
T　DBD16　　//累加器 1 中的运算结果传送到 DBD16
3．加、减 8 位整数指令
L　MB4　　//MB4 的内容装入累加器 1 的最低字节
INC 1　　//累加器 1 最低字节的内容加 1，结果存放在累加器 1 的最低字节
T　MB4　　//运算结果传回 MB4
4．地址寄存器指令
+AR1 (Add to AR1) 指令将 AR1 的内容加上累加器 1 中低字的内容，或加上指令中的 16 位常数，结果在 AR1 中。地址寄存器中的存储区域标识符（第 24～26 位）保持不变。

3.7 逻辑控制指令
表 3-29 逻辑控制指令与状态位触点指令
语句表中的逻辑控制指令　梯形图中的状态位触点指令　说明
JUJLJCJCNJCBJNBJBIJNBIJOJOSJZJNJPJMJPZJMZJUOLOOP　－－－－－－BR－OVOS＝＝0<> 0> 0< 0>= 0<= 0UO－　无条件跳转多分支跳转 RLO=1 时跳转 RLO=0 时跳转 RLO=1 且 BR＝1 时跳转 RLO=0 且 BR＝1 时跳转 BR=1 时跳转 BR=0 时跳转 OV=1 时跳转 OS=1 时跳转运算结果为 0 时跳转运算结果非 0 时跳转运算结果为正时跳转运算结果为负时跳转运算结果大于等于 0 时跳转运算结果小于等于 0 时跳转指令出错时跳转循环指令
只能在同一逻辑块内跳转。同一个跳转目的地址只能出现一次。跳转或循环指令的操作数为地址标号，标号由最多 4 个字符组成，第一个字符必须是字母，其余的可以是字母或数字。在梯形图中，目标标号必须是一个网络的开始。

PLC WORLD

【例 3-10】IW8 与 MW12 的异或结果如果为 0，将 M4.0 复位，非 0 则将 M4.0 置位。

```
      L       IW8          //IW8 的内容装入累加器 1 的低字
L       MW12       //累加器 1 的内容装入累加器 2，MW12 的内容装入累加器 1
XOW             //累加器 1，2 低字的内容逐位异或
JN      NOZE       //如果累加器 1 的内容非 0，则跳转到标号 NOZE 处
R       M4.0
JU      NEXT
NOZE:   AN      M4.0
      S       M4.0
NEXT:   NOP   0
```

3.7.3 循环指令

循环指令 LOOP <jump label>用 ACCU 1-L 作循环计数器，每次执行 LOOP 指令时 ACCU 1-L 的值减 1，若减 1 后 ACCU 1-L 非 0，将跳转到<jump label>指定的标号处。

【例 3-11】用循环指令求 5！(5 的阶乘)。

```
L       L#1        //32 位整数常数装入累加器 1，置阶乘的初值
T       MD20    //累加器 1 的内容传送到 MD20，保存阶乘的初值
L       5          //循环次数装入累加器的低字
BACK:   T       MW10       //累加器 1 低字的内容保存到循环计数器 MW10
L       MD20       //取阶乘值
*D              //MD20 与 MW10 的内容相乘
T       MD20       //乘积送 MD20
L       MW10       //循环计数器内容装入累加器 1
LOOP    BACK       //累加器 1 低字的内容减 1，减 1 后非 0，跳到标号 BACK
      ……           //循环结束后，恢复线性扫描
```

3.8 程序控制指令

表 3-30 程序控制指令

语句表指令　梯形图指令　描述

BEBEUBECCALL FCnCALL SFCn CALL FBn1，DBn2CALL SFBn1，DBn2CC FCn 或 SFCnUC FCn 或 SFCnRETMCRAMCRDMCR()MCR　－－－－－－－－ CALLCALLRETMCRAMCRDMCR<MCR>　块结束块无条件结束块条件结束调用功能调用系统功能调用功能块调用系统功能块 RLO＝1 时条件调用无条件调用条件返回起动主控继电器功能取消主控继电器功能打开主控继电器区关闭主控继电器区

```
OPN       DB10    //打开数据块 DB10 作为共享数据块
L       DBW35    //将打开的 DB10 中的数据字 DBW35 装入累加器 1 的低字
T       MW12    //累加器 1 低字的内容装入 MW12
OPN    DI20    //打开作为背景数据块的数据块 DB20
L       DIB35    // DB20.DIB35 装入累加器 1 的最低字节
T       DBB27    //累加器 1 最低字节传送到 DB10.DBB27
```

PLC WORLD

表 3-31 数据块指令

指令 描述

OPNCDBL DBLGL DBNOL DILGL DINO 打开数据块交换共享数据块和背景数据共享数据块的长度装入累加器 1 共享数据块的编号装入累加器 1 背景数据块的长度装入累加器 1 背景数据块的编号装入累加器 1

第四章 STEP 7 编程软件的使用方法

4.1.1 STEP 7 概述

STEP 7 用于 S7，M7，C7，WinAC 的编程、监控和参数设置，基于 STEP 7 V5.2 版。

STEP 7 具有以下功能：硬件配置和参数设置、通信组态、编程、测试、启动和维护、文件建档、运行和诊断功能等。

4.1.2 STEP 7 的硬件接口

PC./MPI 适配器＋RS-232C 通信电缆。

计算机的通信卡 CP 5611（PCI 卡）、CP 5511 或 CP 5512（PCMCIA 卡）将计算机连接到 MPI 或 PROFIBUS 网络。计算机的工业以太网通信卡 CP 1512(PCMCIA 卡)或 CP 1612（PCI 卡），通过工业以太网实现计算机与 PLC 的通信。

STEP 7 的授权在软盘中。STEP 7 光盘上的程序 AuthorsW 用于显示、安装和取出授权。

4.1.4 STEP 7 的编程功能

1．编程语言

3 种基本的编程语言：梯形图(LAD)、功能块图(FBD) 和语句表(STL)。

S7-SCL (结构化控制语言)，S7-GRAPH（顺序功能图语言），S7 HiGraph 和 CFC。

2．符号表编辑器

3．增强的测试和服务功能

设置断点、强制输入和输出、多 CPU 运行(仅限于 S7-400)，重新布线、显示交叉参考表、状态功能、直接下载和调试块、同时监测几个块的状态等。

程序中的特殊点可以通过输入符号名或地址快速查找。

4．STEP 7 的帮助功能

按 F1 键便可以得到与它们有关的在线帮助。菜单命令“Help→contents”进入帮助窗口。

4.1.5 STEP 7 的硬件组态与诊断功能

1．硬件组态

（1）系统组态：选择硬件机架，模块分配给机架中希望的插槽。

（2）CPU 的参数设置。

（3）模块的参数设置。可以防止输入错误的数据。

2．通信组态

（1）网络连接的组态和显示；

（2）设置用 MPI 或 PROFIBUS-DP 连接的设备之间的周期性数据传送的参数。

（3）设置用 MPI、PROFIBUS 或工业以太网实现的事件驱动的数据传输，用通信块编程。

3．系统诊断

（1）快速浏览 CPU 的数据和用户程序在运行中的故障原因。

（2）用图形方式显示硬件配置、模块故障；显示诊断缓冲区的信息等。

4.2 硬件组态与参数设置

4.2.1 项目的创建与项目的结构

PLC WORLD

插入新的对象的方法。

4.2.2 硬件组态

图 4-2 S7-300 的硬件组态窗口

4.2.3 CPU 模块的参数设置

图 4-3 CPU 属性设置对话框

表 4-1 时钟存储器各位对应的时钟脉冲周期与频率

位	7	6	5	4	3	2	1	0
周期（s）	2	1.6	1	0.8	0.5	0.4	0.2	0.1
频率（Hz）	0.5	0.625	1	1.25	2	2.5	5	10

4.2.4 数字量输入模块的参数设置

在 CPU 处于 STOP 模式下进行。设置完后下载到 CPU 中。当 CPU 从 STOP 模式转换为 RUN 模式时，CPU 将参数传送到每个模块。

图 4-4 数字量输入模块的参数设置

4.2.5 数字量输出模块的参数设置

图 4-5 数字量输出模块的参数设置

4.2.6 模拟量输入模块的参数设置

1．模块诊断与中断的设置

8 通道 12 位模拟量输入模块（订货号为 6ES7 331-7KF02-0AB0）的参数设置。

图 4-6 模拟量输入模块的参数设置

2．模块测量范围的选择

“4DMU”是 4 线式传感器电流测量，“R-4L”是 4 线式热电阻，“TC-I”是热电偶，“E”表示测量种类为电压。

未使用某一组的通道应选择测量种类中的“Deactivated”（禁止使用）。

3．模块测量精度与转换时间的设置

SM 331 采用积分式 A/D 转换器，积分时间直接影响到 A/D 转换时间、转换精度和干扰抑制频率。为了抑制工频频率，一般选用 20ms 的积分时间。

表 4-2 6ES7 331-7KF02 模拟量输入模块的参数关系

积分时间（ms）	2.5	16.7	20	100
基本转换时间（ms，包括积分时间）	3	17	22	102
附加测量电阻转换时间（ms）	1	1	1	1
附加开路监控转换时间（ms）	10	10	10	10
附加测量电阻和开路监控转换时间（ms）	16	16	16	16
精度（位，包括符号位）	9	12	12	14
干扰抑制频率（Hz）	400	60	50	10
模块的基本响应时间（ms，所有通道使能）	24	136	176	816

4．设置模拟值的平滑等级

在平滑参数的四个等级（无，低，平均，高）中进行选择。

4.2.7 模拟量输出模块的参数设置

PLC WORLD

CPU 进入 STOP 时的响应：不输出电流电压（0CV）、保持最后的输出值（KLV）和采用替代值（SV）。

4.3.1 符号表

共享符号（全局符号）在符号表中定义，可供程序中所有的块使用。

在程序编辑器中用 “View→Display with→Symbolic Representation”选择显示方式。

2．生成与编辑符号表

CPU 将自动地为程序中的全局符号加双引号，在局部变量的前面自动加“#”号。生成符号表和块的局域变量表时不用为变量添加引号和#号。

图 4-7 符号表

数据块中的地址（DBD，DBW，DBB 和 DBX）不能在符号表中定义。应在数据块的声明表中定义。

用菜单命令“View→Columns R, O, M, C, CC”可以选择是否显示表中的“R, O, M, C, CC”列，它们分别表示监视属性、在 WinCC 里是否被控制和监视、信息属性、通信属性和触点控制。可以用菜单命令“View→Sort”选择符号表中变量的排序方法。

3．共享符号与局域符号，后者不能用汉字。

4．过滤器（Filter）

在符号表中执行菜单命令“View→Filter”，“I*”表示显示所有的输入，“I*.*”表示所有的输入位，“I2.*”表示 IB2 中的位等。

4.3.2 逻辑块

逻辑块包括组织块 OB、功能块 FB 和功能 FC。

1．程序的输入方式：增量输入方式或源代码方式（或称文本方式、自由编辑方式）。

2．生成逻辑块

图 4-8 梯形图编辑器

6．网络

执行菜单命令“Insert→Network”，或点击工具条中相应的图标，在当前网络的下面生成一个新的网络。菜单命令“View→Display→Comments”用来激活或取消块注释和网络注释。

可以用剪贴板在块内部和块之间复制和粘贴网络，可用 Ctrl 键。

7．打开和编辑块的属性

菜单命令“File→Properties”来查看和编辑块属性。

8．程序编辑器的设置

进入程序编辑器后用菜单命令“Option→Customize”打开对话框，可以进行下列设置：

（1）在“General”标签页的“Font”设置编辑器使用的字体和字符的大小。

（2）在“STL”和“LAD/FDB”标签页中选择这些程序编辑器的显示特性。

（3）在“Block”（块）标签页中，可以选择生成功能块时是否同时生成背景数据块、功能块是否有多重背景功能。

（4）在“View”选项卡中的“View after Open Block”区，选择在块打开时显示的方式。

9．显示方式的设置

执行 View 菜单中命令，放大、缩小梯形图或功能块图的显示比例。

菜单命令“View→Display→Symbolic Representation”，切换绝对地址和符号地址方式。

菜单命令“View→Display→Symbol information”用来打开或关闭符号信息。

图 4-9 符号信息

PLC WORLD

4.4 S7-PLCSIM 仿真软件在程序调试中的应用

4.4.1 S7-PLCSIM 的主要功能

在计算机上对 S7-300/400 PLC 的用户程序进行离线仿真与调试。
模拟 PLC 的输入/输出存储器区，来控制程序的运行，观察有关输出变量的状态。
在运行仿真 PLC 时可以使用变量表和程序状态等方法来监视和修改变量。
可以对大部分组织块（OB）、系统功能块（SFB）和系统功能（SFC）仿真。

4.4.2 使用 S7-PLCSIM 仿真软件调试程序的步骤

（1）在 STEP 7 编程软件中生成项目，编写用户程序。
（2）打开 S7-PLCSIM 窗口，自动建立了 STEP 7 与仿真 CPU 的连接。
仿真 PLC 的电源处于接通状态，CPU 处于 STOP 模式，扫描方式为连续扫描。
（3）在管理器中打开要仿真的项目，选中“Blocks”对象，将所有的块下载到仿真 PLC。
（4）生成视图对象。
（5）用视图对象来模拟实际 PLC 的输入/输出信号，检查下载的用户程序是否正确。

4.4.3 应用举例

电动机串电阻降压起动。速度监视。

图 4-11 S7-PLCSIM 仿真窗口

4.4.4 视图对象与仿真软件的设置与存档

1．CPU 视图对象
2．其他视图对象
通用变量（Generic Variable）视图对象用于访问仿真 PLC 所有的存储区（包括数据块）。
垂直位（Vertical Bits）视图对象可以用绝对地址或符号地址来监视和修改 I，Q，M 等存储区。
累加器与状态字视图对象用来监视 CPU 中的累加器、状态字和地址寄存器 AR1 和 AR2。
块寄存器视图对象用来监视数据块地址寄存器的内容，当前和上一次打开的逻辑块的编号，以及块中的步地址计数器 SAC 的值。
嵌套堆栈（Nesting Stacks）视图对象用来监视嵌套堆栈和 MCR（主控继电器）堆栈。
定时器视图对象标有“T=0”的按钮用来复位指定的定时器。
3．设置扫描方式
用“Execute”菜单中的命令选择单次扫描或连续扫描。
4．设置 MPI 地址
菜单命令“PLC→MPI Address…”设置仿真 PLC 在指定的网络中的节点地址。
5．LAY 文件和 PLC 文件
LAY 文件用于保存仿真时各视图对象的信息；PLC 文件用于保存上次仿真运行时设置的数据和动作等。退出仿真软件时将会询问是否保存 LAY 文件或 PLC 文件。一般选择不保存。

4.5 STEP 7 与 PLC 的在线连接与在线操作

4.5.1 装载存储器与工作存储器

系统数据（System Data）包括硬件组态、网络组态和连接表，也应下载到 CPU。
下载的用户程序保存在装载存储器的快闪存储器（FEPROM）中。CPU 电源掉电又重新恢复时，FEPROM 中的内容被重新复制到 CPU 存储器的 RAM 区。

4.5.2 在线连接的建立与在线操作

1．建立在线连接

PLC WORLD

通过硬件接口连接计算机和 PLC 必须，然后通过在线的项目窗口访问 PLC。
管理器中执行菜单命令“View→Online”、“View→Offline”进入离线状态。
在线窗口显示的是 PLC 中的内容，离线窗口显示的是计算机中的内容。
如果 PLC 与 STEP 7 中的程序和组态数据是一致的，在线窗口显示的是 PLC 与 STEP 7 中的数据的组合。
2．处理模式与测试模式
在设置 CPU 属性的对话框中的“Protection”（保护）标签页选择处理（Process）模式或测试（Test）模式。
3．在线操作
进入在线状态后，执行菜单命令“PLC →Diagnostics/Settings”中不同的子命令。
进入在线状态后，“PLC”主菜单中的命令功能。
设置了口令后，执行在线功能时，会显示出“Enter Password”对话框。若输入的口令正确，就可以访问该模块。用菜单命令“PLC→Access Rights→ Setup”输入口令。
4.5.3 下载与上载
1．下载的准备工作
计算机与 CPU 之间必须建立起连接，要下载的程序已编译好；在 RUN-P 模式一次只能下载一个块，建议在 STOP 模式下载。
在保存块或下载块时，STEP 7 首先进行语法检查，应改正检查出来的错误。下载前应将 CPU 中的用户存储器复位。可以用模式选择开关复位，CPU 进入 STOP 模式，再用菜单命令“PLC→Clear/Reset”复位存储器。
2．下载的方法
（1）在离线模式下载
在管理器的块工作区选择块，可用 Ctrl 键和 Shift 键选择多个块，用菜单命令“PLC→Download”将被选择的块下载到 CPU。在管理器左边的目录窗口中选择 Blocks 对象，下载所有的块和系统数据。
对块编程或组态硬件和网络时，在当时主窗口，用菜单命令“PLC→Download”下载当前正在编辑的对象。
(4) 上载程序
可以用“PLC→Upload”命令从 CPU 的 RAM 装载存储器中，把块的当前内容上载到计算机打开的项目中。
4.6 用变量表调试程序
4.6.1 系统调试的基本步骤
首先进行硬件调试，可以用变量表来测试硬件，通过观察 CPU 模块上的故障指示灯，或使用 4.8 节介绍的故障诊断工具来诊断故障。
下载程序之前应将 CPU 的存储器复位，将 CPU 切换到 STOP 模式，下载用户程序时应同时下载硬件组态数据。

可以在 OB1 中逐一调用各程序块，一步一步地调试程序。
最先调试起动组织块 OB100，然后调试 FB 和 FC。应先调试嵌套调用最深的块，例如首先调试图 4-13 中的 FB1。调试时可以在完整的 OB1 的中间临时插入 BEU（块无条件结束）指令，只执行 BUE 指令之前的部分，调试好后将它删除掉。
最后调试不影响 OB1 的循环执行的中断处理程序，或者在调试 OB1 时调试它们。
4.6.2 变量表的基本功能
变量表可以在一个画面中同时监视、修改和强制用户感兴趣的全部变量。一个项目可以生成

PLC WORLD

多个变量表。变量表的功能：
监视（Monitor）变量、修改（Modify）变量、对外设输出赋值、强制变量、定义变量被监视或赋予新值的触发点和触发条件。

4.6.3 变量表的生成

1．生成变量表的几种方法
（1）在管理器中用生成新的变量表。
（3）在变量表编辑器中，可以用主菜单“Table”生成一个新的变量表。
2．在变量表中输入变量
可以从符号表中拷贝地址，将它粘贴到变量表。
IW2 用二进制数（BIN）可以同时显示和分别修改 I 2.0～I 3.7 这十六点数字量输入变量。

图 4-14 变量表

4.6.4 变量表的使用

1．建立与 CPU 的连接
2．定义变量表的触发方式

图 4-15 定义变量表的触发方式

用菜单命令“Variable→Trigger”打开图 4-15 中的对话框选择触发方式。
3．监视变量
用菜单命令“Variable→Update Monitor Values”对所选变量的数值作一次立即刷新。
4．修改变量
在 STOP 模式修改变量时，各变量的状态不会互相影响，并且有保持功能。
在 RUN 模式修改变量时，各变量同时又受到用户程序的控制。
5．强制变量
强制变量操作给用户程序中的变量赋一个固定的值，不会因为用户程序的执行而改变。

图 4-16 强制数值窗口

强制作业只能用菜单命令“Variable→Stop Forcing”来删除或终止。

4.7 用程序状态功能调试程序

4.7.1 程序状态功能的起动与显示

1．起动程序状态
进入程序状态的条件：经过编译的程序下载到 CPU；打开逻辑块，用菜单命令“Debug→Monitor”进入在线监控状态；将 CPU 切换到 RUN 或 RUN-P 模式。
2．语句表程序状态的显示

图 4-17 用程序状态监视语句表程序

从光标选择的网络开始监视程序状态。右边窗口显示每条指令执行后的逻辑运算结果（RLO）和状态位 STA(Status)、累加器 1（STANDARD）、累加器 2(ACCU 2)和状态字（STATUS…）。用菜单命令“Options→Customize”打开的对话框分 STL 标签页选择需要监视的内容，用 LAD/FBD 标签页可以设置梯形图(LAD)和功能块图(SFB)程序状态的显示方式。
3．梯形图程序状态的显示
LAD 和 FBD 中用绿色连续线来表示状态满足，即有“能流”流过，见图 4-18 左边较粗较浅的线；用兰色点状线细表示状态不满足，没有能流流过；用黑色连续线表示状态未知。

PLC WORLD

图 4-18 梯形图程序状态的显示

梯形图中加粗的字体显示的参数值是当前值，细体字显示的参数值来自以前的循环。

4．使用程序状态功能监视数据块

4.7.2 单步与断点功能的使用

进入 RUN 或 RUN-P 模式后将停留在第一个断点处。单步模式一次只执行一条指令。

程序编辑器的“Debug（调试）”菜单中的命令用来设置、激活或删除断点。执行菜单命令“View＞Breakpoint Bar”后，在工具条中将出现一组与断点有关的图标。

1．设置断点与进入单步模式的条件

（1）只能在语句表中使用单步和断点功能。

（2） 执行菜单命令“Options → Customize”，在对话框中选择 STL 标签页，激活“Activate new breakpoints immediately（立即激活新断点）”选项。

（3）必须用菜单命令“Debug＞Operation”使 CPU 工作在测试（Test）模式。

（4）在 SIMATIC 管理器中进入在线模式，在线打开被调试的块。

（5）设置断点时不能起动程序状态（Monitor）功能。

（6）STL 程序中有断点的行、调用块的参数所在的行、空的行或注释行不能设置断点。

2．设置断点与单步操作

在菜单命令“Debug →Breakpoints Active”前有一个“√”（默认的状态），表示断点的小圆是实心的。执行该菜单命令后“√”消失，表示断点的小圆变为空心的。要使断点起作用，应执行该命令来激活断点。

图 4-19 断点与断点处 CPU 寄存器和状态字的内容

将 CPU 切换到 RUN 或 RUN-P 模式，将在第一个表示断点的紫色圆球内出现一个向右的黄色的箭头（见图 4-19），表示程序的执行在该点中断，同时小窗口中出现断点处的状态字等。执行菜单命令“Debug→Execute Next Statement”，黄色箭头移动到下一条语句，表示用单步功能执行下一条语句。执行菜单命令“Debug→ Execute Call（执行调用）”将进入调用的块。块结束时将返回块调用语句的下一条语句。

为使程序继续运行至下一个断点，执行菜单命令“Debug → Resume（继续）”。

菜单命令“Debug→Delete Breakpoint”删除一个断点，菜单命令“Debug→Delete All Breakpoint”删除所有的断点。执行菜单命令“Show Next Breakpoint”，光标跳到下一个断点。

4.8 故障诊断

4.8.1 故障诊断的基本方法

图 4-20 诊断符号

在管理器中用 “View→Online”打开在线窗口。查看是否有 CPU 显示诊断符号。

4.8.2 模块信息在故障诊断中的应用

1．打开模块信息窗口

建立在线连接后，在管理器中选择要检查的站，执行菜单命令“PLC→Diagnostics/ Settings→Module Information”，显示该站中 CPU 模块的信息。诊断缓冲区（Diagnostic Buffer）标签页中，给出了 CPU 中发生的事件一览表。

图 4-21 CPU 模块的在线模块信息窗口

PLC WORLD

最上面的事件是最近发生的事件。因编程错误造成 CPU 进入 STOP 模式，选择该事件，并点击“Open Block”按钮，将在程序编辑器中打开与错误有关的块，显示出错的程序段。

4.8.3 用快速视窗和诊断视窗诊断故障

1．用快速视窗诊断故障

管理器中选择要检查的站，用命令“PLC→Diagnostics/Settings→Hardware Diagnose”打开 CPU 的硬件诊断快速视窗（Quick View），显示该站中的故障模块。用命令“Option→Customize”，在打开的对话框的“View”标签页中，应激活“诊断时显示快速视窗”。

图 4-22 快速视窗

2．打开诊断视窗

诊断视窗实际上就是在线的硬件组态窗口。在快速视窗中点击“Open Station Online”（在线打开站）按键，打开硬件组态的在线诊断视窗。

在管理器中与 PLC 建立在线连接。打开一个站的“Hardware”对象，可以打开诊断视窗。

3．诊断视窗的信息功能

诊断视窗显示整个站在线的组态。用命令“PLC＞Module Information”查看其模块状态。

第五章 数字量控制系统梯形图设计方法

5.1.2 用经验法设计梯形图

1．起动、保持与停止电路

经验设计法。

2．三相异步电动机的正反转控制

```
Network 1:
A    I 1.0
=    L20.0
A    L20.0
A    I 1.1
=    Q4.3
A    L20.0
A    I 1.2
=    Q4.4
//图 5-5 (b)中的程序
Network 1
A    I 1.0
A    I 1.1
=    Q4.3
Network 2
A    I 1.0
A    I 1.2
=    Q4.4
```

PLC WORLD

3．常闭触点输入信号的处理
4．小车控制程序的设计

按下右行起动按钮 SB2，小车右行。暂停，左行，停止。

5.1.2 根据继电器电路图设计梯形图

液压动力滑台开始停在最左边，在自动模式开关 SA 闭合。按下起动按钮 SB1(I0.0)， YV11 和 YV2 的线圈通电，快进；碰到中限位开关变为工进， YV2 的线圈断电；碰到右限位开关暂停 8sYV11 的线圈断电；时间到时快退， YV12 的线圈通电；返回初始位置时 YV12 的线圈断电，停止运动。

5.2 顺序控制设计法与顺序功能图

5.2.1 顺序控制设计法

顺序控制设计法将系统的一个工作周期划分为若干个顺序相连的阶段（步，Step)，用编程元件(例如 M)来代表各步。在任何一步内输出量的状态不变，相邻两步输出量总的状态是不同的，步与各输出量有着极为简单的逻辑关系。

使系统由当前步进入下一步的信号称为转换条件。顺序控制设计法用转换条件控制代表各步的编程元件，让它们的状态按一定的顺序变化，然后用代表各步的编程元件去控制输出。

当系统正处于某一步所在的阶段时，该步处于活动状态，称该步为“活动步”。
非存储型动作与存储型动作。

5.2.4 顺序功能图的基本结构

4．复杂的顺序功能图举例

5.2.5 顺序功能图中转换实现的基本规则

1．转换实现的条件

在顺序功能图中，步的活动状态的进展是由转换的实现来完成的。转换实现必须同时满足两个条件：

(1) 该转换所有的前级步都是活动步；
(2) 相应的转换条件得到满足。

如果转换的前级步或后续步不止一个，转换的实现称为同步实现(见图 5-18)。为了强调同步实现，有向连线的水平部分用双线表示。

2. 转换实现应完成的操作

转换实现时应完成以下两个操作：

（1）使所有由有向连线与相应转换符号相连的后续步都变为活动步；
（2）使所有由有向连线与相应转换符号相连的前级步都变为不活动步。

PLC WORLD

5.2.6 绘制顺序功能图的注意事项

(1) 两个步绝对不能直接相连，必须用一个转换将它们隔开。

(2) 两个转换也不能直接相连，必须用一个步将它们隔开。

(3) 顺序功能图中的初始步对应于系统等待起动的初始状态，初始步是必不可少的。

(4) 顺序功能图中一般应有由步和有向连线组成的闭环。

5.2.7 顺序控制设计法的本质

5.3 使用起保停电路的顺序控制梯形图编程方法

5.3.1 设计顺序控制梯形图的一些基本问题

1．程序的基本结构

2．执行自动程序的初始状态

3．双线圈问题

4．设计顺序控制程序的基本方法

用存储器位 M 来代表步。顺序控制程序分为控制电路和输出电路两部分。

5.3.2 单序列的编程方法

1．控制电路的编程方法

起保停电路的起动电路只能接通一个扫描周期，必须用有记忆功能的电路来控制 M。

2．输出电路的编程方法

5.3.3 选择序列的编程方法

5.3.4 并行序列的编程方法

5.3.5 仅有两步的闭环的处理

5.3.6 应用举例

图 5-25 中的物料混合装置用来将粉末状的固体物料（粉料）和液体物料（液料）按一定的比例混合在一起，经过一定时间的搅拌后便得到成品。粉料和液料都用电子称来计量。

初始状态时粉料称料斗、液料称料斗和搅拌器都是空的，它们底部的排料阀关闭；液料仓的放料阀关闭，粉料仓下部的螺旋输送机的电机和搅拌机的电机停转；Q4.0～Q4.4 均为 0 状态。

PLC 开机后用 OB100 将初始步对应的 M0.0 置为 1 状态，将其余各步对应的存储器位复位为 0 状态，并将 MW10 和 MW12 中的计数预置值分别送给减计数器 C0 和 C1。

按下起动按钮 I0.0，Q4.0, Q4.1 变为 1 状态，开始进料。电子称的光电码盘输出与称斗内物料重量成正比的脉冲信号。减计数器 C0 和 C1 分别对粉料称和液料称产生的脉冲计数。脉冲计数值减至 0 时，其常闭触点闭合，称斗内的物料等于预置值。Q4.0, Q4.1 变为 0 状态，停止进料。进入等待步后预置计数器。

PLC WORLD

5.4 使用置位复位指令的顺序控制梯形图编程方法

5.4.1 单序列的编程方法

5.4.2 选择序列的编程方法

5.4.3 并行序列的编程方法

图 5-32 组合钻床控制系统的梯形图

值得注意的是标有“CD”的C0的减计数线圈必须“紧跟”在图5-32中使M0.7置位的指令后面。这是因为如果 M0.4 先变为活动步，M0.7 的“生存周期”非常短，M0.7 变为活动步后，在本次循环扫描周期内的下一个网络就被复位了。如果将 C0 的减计数线圈放在使 M0.7 复位的指令的后面，C0 还没有计数 M0.7 就被复位了，将不能执行计数操作。

5.5 具有多种工作方式的系统的顺序控制梯形图编程方法

5.6 顺序功能图语言 S7 Graph 的应用

5.6.1 S7 Graph 语言概述

S7 Graph 语言是 S7-300/400 的顺序功能图语言，遵从 IEC 61131-3 标准的规定。

1．顺序控制程序的结构

一个顺序控制项目至少需要一个调用 S7 Graph FB 的块，一个 S7 Graph FB 和它的背景数

PLC WORLD

据块。

图 5-45 顺序控制系统中的块

图 5-46 S7 Graph 编辑器

图 5-49 顺序控制器工具条与移动的图形

3．S7 Graph 的显示模式

在 View 菜单中选择显示顺序控制器（Sequencer）、单步和永久性指令。

（1）在顺序控制器显示方式，执行菜单命令“View>Display with”，可以选择：

Symbols：显示符号表中的符号地址；

Comments：显示块和步的注释；

Conditions and Actions：显示转换条件和动作；

Symbol List：在输入地址时显示下拉式符号地址表。

（2）单步显示模式

只显示一个步和转换的组合，还可以显示 Supervision：监控被显示的步的条件；Interlock：对被显示的步互锁的条件；执行命令“View>Display with> comments”显示和编辑步的注释。

用“↑”键或“↓”键可以显示上一个或下一个步与转换的组合。

（3）在“permanent instructions”（永久性指令）显示方式，可以对顺序控制器之前或之后的永久性指令编程。每个扫描循环执行一次永久性指令。可以调用块。

图 5-50 运输带控制系统示意图与顺序功能图

1． 创建使用 S7 Graph 语言的功能块 FB

执行菜单命令“Insert → Direct”将进入“Direct”编辑模式。

执行菜单命令“Insert → Drag-and-Drop”，进入“Drag and Drop（拖放）”编辑模式。

执行菜单命令“View→Display with→Conditions and Actions”，显示或关闭各步的动作和转换条件。

图 5-51 运输带控制系统的顺序功能图

（1）命令 S：当步为活动步时，使输出置位为 1 状态并保持。

（2）命令 R：当步为活动步时，使输出复位为 0 状态并保持。

（3）命令 N：当步为活动步时，输出被置为 1；该步变为不活动步时，输出被复位为 0。

（4）命令 L：用来产生宽度受限的脉冲，相当于脉冲定时器。

（5）命令 CALL：用来调用块，当该步为活动步时，调用命令中指定的块。

（6）命令 D：使某一动作的执行延时，延时时间在该命令右下方的方框中设置。

在“直接”模式用鼠标右键点击动作框，在弹出的菜单中选择插入动作行。

6．对监控功能编程

双击步 S3 后，切换到单步视图，选中 Supervision（监控）线圈左边的水平线的缺口处，插入比较器图标，设置的监视时间为 2 小时。

8．在主程序中调用 S7 Graph FB

9．用 S7-PLCSIM 仿真软件调试 S7 Graph 程序

图 5-52 单步显示模式中的监控与互锁条件

PLC WORLD

5.6.3 顺序控制器的运行模式与监控操作

执行菜单命令“Debug→Control Sequencer”，对顺序控制器进行各种监控操作。

图 5-54 顺序控制器监控对话框

1．自动模式

“Acknowledge”按钮确认被挂起的错误信息。

点击“初始化（Initialize）”按钮，将重新起动顺序控制器，使之返回初始步。

点击“禁止（Disable）”按钮，使顺序控制器中所有的步变为不活动步。

2．手动模式

选择“Manual”模式后，用“Disable”按钮关闭当前的活动步。在“Step Number”输入框中输入希望控制的步的编号，用激活（Activate）按钮或去活（Unactivate）按钮来使该步变为活动步或不活动步。同时只能有 1 步是活动步。

3．单步（Inching）模式

在单步模式转换条件满足时，需要点击“Continue”按钮，才能使转换到下一步。

4．Automatic or switch to next 模式

转换条件未满足，用“Continue”按钮也能转换到后续步。转换条件满足将自动转换。

5．错误显示

有互锁（Interlock）错误或监控（Supervision）错误时，相应的检查框为红色。

5.6.4 顺序控制器中的动作

1．标准动作中的命令：S, R, N, L, D, CALL

标准动作可以设置互锁（在命令的后面加“C”），仅在步处于活动状态和互锁条件满足时，有互锁的动作才被执行。没有互锁的动作在步处于活动状态时就会被执行。

2．与事件有关的动作

表 5-2 控制动作的事件

名称	事件意义
S1	步变为活动步
S0	步变为不活动步
V1	发生监控错误（有干扰）
V0	监控错误消失（无干扰）
L1	互锁条件解除
L0	互锁条件变为 1
A1	报文被确认
R1	注册信号被置位，在输入信号 REG_EF/REG_S 的上升沿

ON 命令或 OFF 命令分别使命令所在的步之外的其他步变为活动步或不活动步。

如果命令 OFF 的地址标识符为 S _ALL，将除了命令“S1(V1, L1) OFF”所在的步之外其他的步变为不活动步。

一旦 S3 变为活动步和互锁条件满足，指令“S1 RC”使输出 Q2.1 复位为 0 并保持为 0。

一旦监控错误发生（出现 V1 事件），除了动作中的命令“V1 OFF”所在的步 S3，其他的活动步变为不活动步。

PLC WORLD

S3 变为不活动步时（出现事件 S0），将步 S7 变为活动步。
只要互锁条件满足（出现 L0 事件），就调用指定的功能块 FB 2。
4．动作中的计数器
有互锁功能的计数器在互锁条件满足和指定的事件出现时，动作中的计数器才会计数。
事件发生时，计数器指令 CS 将初值装入计数器。CS 指令下面一行是要装入的初值。
事件发生时，CU，CD，CR 指令使计数值分别加 1、减 1 或将计数值复位为 0。
5．动作中的定时器
事件出现时定时器被执行。互锁功能也可以用于定时器。
TL 为扩展的脉冲定时器命令，一旦事件发生，定时器被起动。
TD 命令用来实现定时器位有闭锁功能的延迟。一旦事件发生，定时器被起动。互锁条件 C 仅仅在定时器被起动的那一时刻起作用。

（4）TR 是复位定时器命令，一旦事件发生，定时器位与定时值被复位为 0。
当图 5-57 中的步 S4 变为活动步，事件 S1 使计数器 C4 的值加 1。C4 可以用来计步 S4 变为活动步的次数。只要步 S4 变为活动步，事件 S1 使 A 的值加 1。
S4 变为活动步后，T3 开始定时，4s 后 T3 的定时器位变为 1 状态。
5．动作中的算术运算
在动作中可以使用：A:=B；A:=函数(B)；A:=B<运算符号>C。A:=函数(B)；
5.6.5 顺序控制器中的条件
1．转换条件
2． 互锁条件：如果互锁条件的逻辑满足，受互锁控制的动作被执行。
3．监控条件：如果监控条件的逻辑运算满足，表示有干扰事件 V1 发生。顺序控制器不会转换到下一步，保持当前步为活动步。如果监控条件的逻辑运算不满足，表示没有干扰，如果转换条件满足，转换到下一步。只有活动步被监控。
4．S7 Graph 地址在条件中的应用
可以在转换、监控、互锁、动作和永久性的指令中，以地址的方式使用关于步的系统信息(见表 5-3)。
表 5-3 S7 Graph 地址

地址	意义	应用于
Si.T	步 i 当前或前一次处于活动状态的时间	比较器，设置
Si.U	步 i 处于活动状态的总时间，不包括干扰的时间	比较器，设置
Si.X	指示步 i 是否是活动的	常开触点、常闭触点
Transi.TT	检查转换 i 所有的条件是否满足	常开触点、常闭触点

表 5-4 FB 的参数集

名称	任务
Minimum	最小参数集，只用于自动模式，不需要其他控制和监视功能
Standard	标准参数集，有多种操作方式，需要反馈信息，可选择确认报文
Definable/Maximum(V5)	可定义最大参数集，需要更多的操作员控制和用于服务和调试的监视功能，它们由 V5 的块提供

PLC WORLD

5.6.7 用 S7 Graph 编写具有多种工作方式的控制程序

1．初始化程序、手动程序与自动回原点程序

OB100 中的初始化程序与 5.5 节中的图 5-37 完全相同。手动程序 FC 2 与 5.5 节中的图 5-39 完全相同。自动返回原点的梯形图程序 FC 3 与 5.5 节图 5-42(b)中的相同。

图 5-61 主程序 OB1

S7 Graph FB 的参数有好几十个，图 5-61 中的 FB1 使用的是标准参数级，下面介绍图中使用的参数：

连续、单周期或单步时“自动方式”M0.3 为 1，调用 FB1。

参数 INIT_SQ（“自动允许”M0.0）为 1：原点条件满足，激活初始步，复位顺序控制器。

参数 OFF_SQ 为 1（“自动允许”M0.0=0）：复位顺序控制器，所有的步变为不活动步。

参数 ACK_EF（“确认故障”I1.3）为 1：确认错误和故障，强制切换到下一步。

参数 SW_AUTO（“单周连续”M0.2）为 1：切换到自动模式。

参数 SW_TAP（“单步”I2.2）为 1：切换到 Inching(单步)模式。

参数 T_PUSH（“起动按钮”I2.6）：条件满足并且在 T_PUSH 的上升沿时，转换实现。

参数 ERR_FLT（“错误报警”Q4.5）为 1：组故障。

表 5-9 符号表

符号	地址	符号	地址	符号	地址	符号	地址	符号	地址
自动数据块	DB1	松开按钮	I0.7	单步	I2.2	自动方式	M0.3	下降阀	Q4.0
下限位	I0.1	下降按钮	I1.0	单周期	I2.3	原点条件	M0.5	夹紧阀	Q4.1
上限位	I0.2	右行按钮	I1.1	连续	I2.4	转换允许	M0.6	上升阀	Q4.2
右限位	I0.3	夹紧按钮	I1.2	起动按钮	I2.6	连续标志	M0.7	右行阀	Q4.3
左限位	I0.4	确认故障	I1.3	停止按钮	I2.7	回原点上升	M1.0	左行阀	Q4.4
上升按钮	I0.5	手动	I2.0	自动允许	M0.0	回原点左行	M1.1	错误报警	Q4.5
左行按钮	I0.6	回原点	I2.1	单周连续	M0.2	夹紧延时	M1.2		

图 5-62 公用程序

连续标志 M0.7 的控制电路放在 FB1 的顺序控制器之前的永久性指令中。

图 5-63 顺序控制器之前的永久性指令

2．初始化程序、手动程序与自动回原点程序

OB100 中的初始化程序与 5.5 节中的图 5-37 完全相同。手动程序 FC 2 与 5.5 节中的图 5-39 完全相同。自动返回原点的梯形图程序 FC 3 与 5.5 节图 5-42(b)中的相同。

FB1 是自动程序（单步、单周期、连续）。

单步 I2.2＝SW_TAP＝1 时有单步功能。

单周连续 M0.2=SW_AUTO=1 时顺序控制器正常运行。

在顺序控制器中，用永久性指令中的 M0.7（连续标志）区分单周期和连续模式。

第六章 S7-300/400 的用户程序结构

6.1 用户程序的基本结构

6.1.1 用户程序中的块

操作系统处理起动、刷新过程映像表、调用用户程序、处理中断和错误、管理存储区和处理

PLC WORLD

通信等。用户程序包含处理用户特定的自动化任务所需要的所有功能。
用户程序和所需的数据放置在块中，使程序部件标准化，用户程序结构化，可以简化程序组织，使程序易于修改、查错和调试。块结构显著地增加了 PLC 程序的组织透明性、可理解性和易维护性。
表 6-1 用户程序中的块
块　简要描述
组织块（OB）　操作系统与用户程序的接口，决定用户程序的结构
系统功能块（SFB）　集成在 CPU 模块中，通过 SFB 调用一些重要的系统功能，有存储区
系统功能（SFC）　集成在 CPU 模块中，通过 SFC 调用一些重要的系统功能，无存储区
功能块（FB）　用户编写的包含经常使用的功能的子程序，有存储区
功能（FC）　用户编写的包含经常使用的功能的子程序，无存储区
背景数据块（DI）　调用 FB 和 SFB 时用于传递参数的数据块，在编译过程中自动生成数据
共享数据块（DB）　存储用户数据的数据区域，供所有的块共享
1．组织块(OB)
控制扫描循环和中断程序的执行、PLC 的启动和错误处理等。
（1）OB1 用于循环处理，用户程序中的主程序。
（2）事件中断处理，需要时才被及时地处理。
（3）中断的优先级，高优先级的 OB 可以中断低优先级的 OB。
2．临时局域数据
生成逻辑块（OB、FC、FB）时可以声明临时局域数据。这些数据是临时的，局域（Local）数据，只能在生成它们的逻辑块内使用。所有的逻辑块都可以使用共享数据块中的共享数据。
3．功能（FC）
没有固定的存储区的块，其临时变量存储在局域数据堆栈中，功能执行结束后，这些数据就丢失了。用共享数据区来存储那些在功能执行结束后需要保存的数据。
调用功能和功能块时用实参（实际参数）代替形参（形式参数）。形参是实参在逻辑块中的名称，功能不需要背景数据块。功能和功能块用 IN、OUT 和 IN_OUT 参数做指针，指向调用它的逻辑块提供的实参。功能可以为调用它的块提供数据类型为 RETURN 的返回值。
4．功能块（FB）
功能块是用户编写的有自己的存储区（背景数据块）的块，每次调用功能块时需要提供各种类型的数据给功能块，功能块也要返回变量给调用它的块。这些数据以静态变量（STAT）的形式存放在指定的背景数据块 (DI) 中，临时变量 TEMP 存储在局域数据堆栈中。
调用 FB 或 SFB 时，必须指定 DI 的编号。在编译 FB 或 SFB 时自动生成背景数据块中的数据。一个功能块可以有多个背景数据块，用于不同的被控对象。
可以在 FB 的变量声明表中给形参赋初值。如果调用块时没有提供实参，将使用上一次存储在 DI 中的参数。
5．数据块
数据块中没有 STEP 7 的指令，STEP 7 按数据生成的顺序自动地为数据块中的变量分配地址。数据块分为共享数据块和背景数据块。
应首先生成功能块，然后生成它的背景数据块。在生成背景数据块时指明它的类型为背景数据块（Instance）和它的功能块的编号。

6．系统功能块 SFB 和系统功能 SFC

PLC WORLD

系统功能块和系统功能是为用户提供的已经编好程序的块，可以调用不能修改。操作系统的一部分，不占用户程序空间。SFB 有存储功能，其变量保存在指定给它的背景数据块中。
7．系统数据块(SDB)包含系统组态数据，例如硬件模块参数和通信连接参数等。
CALL、CU（无条件调用）和 CC（RLO = 1 时调用）指令调用没有参数的 FC 和 FB。
6.1.2 用户程序使用的堆栈

堆栈采用“先入后出”的规则存入和取出数据。最上面的存储单元称为栈顶。
1．局域数据堆栈（L）
存储块的局域数据区的临时变量、组织块的启动信息、块传递参数的信息和梯形图程序的中间结果。可以按位、字节、字和双字来存取，例如 L 0.0，LB9，LW4 和 LD52。各逻辑块均有自己的局域变量表，局域变量仅在它被创建的逻辑块中有效。
2．块堆栈（B 堆栈）
存储被中断的块的类型、编号和返回地址；从 DB 和 DI 寄存器中获得的块被中断时打开的共享数据块和背景数据块的编号；局域数据堆栈的指针。
3．中断堆栈（I 堆栈）
当前的累加器和地址寄存器的内容、数据块寄存器 DB 和 DI 的内容、局域数据的指针、状态字、MCR（主控继电器）寄存器和 B 堆栈的指针。

6.1.3 线性化编程与结构化编程
1．线性化编程：整个用户程序放在循环控制组织块 OB1（主程序）中。
2．模块化编程：程序被分为不同的逻辑块，每个块包含完成某些任务的逻辑指令。
3．结构化编程：将复杂的自动化任务分解为小任务，这些任务由相应的逻辑块来表示，程序运行时所需的大量数据和变量存储在数据块中。调用时将“实参”赋值给形参。

创建顺序：FC1→FB1 及其背景数据块→OB1，被调用的块应该是已经存在的。
6.2.1 发动机控制系统的用户程序结构

3．局域变量的类型
（1）IN(输入变量)：由调用它的块提供的输入参数。
（2）OUT(输出变量)：返回给调用它的块的输出参数。
（3）IN_OUT：初值由调用它的块提供，被子程序修改后返回给调用它的块。
（4）TEMP (临时变量)：暂时保存在局域数据区中的变量。
（5）STAT（静态变量）：在功能块的背景数据块中使用。关闭功能块后，其静态数据保持不变。功能（FC）没有静态变量。
表 6-3 FB1 的变量声明表

Name	Data Type	Address	Declare	Initial Value	Comment
Switch_On	Bool	0.0	IN	FALSE	起动按钮
Switch_Off	Bool	0.1	IN	FALSE	停车按钮
Failure	Bool	0.2	IN	FALSE	故障信号
Actual_Speed	Int	2.0	IN	0	实际转速

PLC WORLD

Engine_On Bool 4.0 OUT FALSE 控制发动机的输出信号
Preset_Speed_Reached Bool 4.1 OUT FALSE 达到预置转速
Preset_Speed Int 6.0 STAT 1500 预置转速

5．程序库
6.2.3 功能块与功能
表 6-4 FC1 的变量声明表
Name Data Type Declare Comment
Engine_On Bool IN 输入信号，发动机运行
Timer_Function Timer IN 停机延时的定时器功能
Fan_On Bool OUT 控制风扇的输出信号

6.2.4 功能块与功能的调用

为了能全部转换为图 6-10 中的梯形图，下面的语句表还需要增加一些语句。

```
Network1：自动手动切换
A       "自动"
        S       "自动模式"
        A       "手动"
        R       "自动模式"
Network2：汽油机控制
CALL  "发动机控制" , "汽油机数据"
  Switch_On          := "起动汽油机"
  Switch_Off         := "关闭汽油机"
  Failure            := "汽油机故障"
  Actual_Speed       := "汽油机转速"
  Engine_On          := "汽油机运行"
  Preset_Speed_Reached := "汽油机到达设置转速"
Network3：汽油机风扇控制
        CALL  "风扇控制"
          Engine_On       := "汽油机运行"
          Timer_Function  := "汽油机风扇延时"
          Fan_On          := "汽油机风扇运行"
```

6.3 数据块
6.3.1 数据块中的数据类型
1．基本数据类型
基本数据类型包括位（Bool），字节（Byte）、字（Word）、双字（Dword）、整数（INT）、双整数（DINT）和浮点数（Float，或称实数 Real）等。
2．复合数据类型
日期和时间用 8 个字节的 BCD 码来存储。第 0～5 号字节分别存储年、月、日、时、分和秒，毫秒存储在字节 6 和字节 7 的高 4 位，星期存放在字节 7 的低 4 位。例如 2004 年 7 月 27 日 12 点 30 分 25.123 秒可以表示为 DT#04-07-27-12:30:25.123。
字符串（STRING）由最多 254 个字符（CHAR）和 2 字节的头部组成。字符串的默认长度

PLC WORLD

为 254，通过定义字符串的长度可以减少它占用的存储空间。

3．数组

数组（ARRAY）是同一类型的数据组合而成的一个单元。ARRAY[1..2,1..3]是一个二维数组，共有 6 个整数元素。最多为 6 维。

数组元素"TANK".PRESS[2,1]：TANK 是数据块的符号名，PRESS 是数组的名称。方括号中是数组元素的下标。如果在块的变量声明表中声明形参的类型为 ARRAY，可以将整个数组而不是某些元素作为参数来传递。

4．结构

结构（STRUCT）是不同类型的数据的组合。可以用基本数据类型、复杂数据类型，和，UDT 作为结构中的元素，可以嵌套 8 层。

数据块 TANK 内结构 STACK 的元素 AMOUNT 应表示为"TANK".STACK.AMOUNT。

将结构作为参数传递时，作为形参和实参的两个结构必须有相同的数据结构，即相同数据类型的结构元素和相同的排列顺序。

5．用户定义数据类型

用户定义数据类型（UDT）是一种特殊的数据结构，由用户自己生成，定义好后可以在用户程序中多次使用。

例如可以生成用于颜料混合配方的 UDT，然后用它生成用于不同颜色配方的数据组合。

6.3.2 数据块的生成与使用

菜单命令"View→Declaration View"和"View→Data View"分别指定声明表显示方式和数据显示方式。声明表显示状态用于定义和修改共享数据块中的变量。

6.4 多重背景

6.4.1 多重背景功能块

生成 FB10 时应激活"Multiple Instance FB"（多重背景功能块）选项。应首先生成 FB1。为调用 FB1，在 FB10 的变量声明表中声明了两个名为"Petrol_Engine（汽油机）"和"Diesel_Engine（柴油机）"的静态变量（STAT），其数据类型为 FB1。生成 FB10 后，"Petrol_Engine"和"Diesel_Engine"将出现在管理器编程元件目录的"Multiple Instances（多重背景）"文件夹内。可以将它们"拖放"到 FB 10 中，然后指定它们的输入参数和输出参数。

6.4.2 多重背景数据块 其中的数据自动产生。

6.4.3 在 OB1 中调用多重背景

图 6-21 中调用 FB10（符号名为"发动机"）的语句表为：

```
Network4：调用多重背景
CALL  "发动机" , "多重背景数据块"
        Preset_Speed_Reached :="两台都达到设置转速"
```

图 6-17 FB10 的变量声明表

图 6-18 多重背景功能块 FB10

使用多重背景时应注意以下问题：

（1）首先应生成需要多次调用的功能块（例如上例中的 FB1）。

（2）管理多重背景的功能块（例如上例中的 FB10）必须设置为有多重背景功能。

（3）在管理多重背景的功能块的变量声明表中，为被调用的功能块的每一次调用定义一个静态（STAT）变量，以被调用的功能块的名称（例如 FB1）作为静态变量的数据类型。

PLC WORLD

（4）必须有一个背景数据块（例如上例中的 DB10）分配给管理多重背景的功能块。背景数据块中的数据是自动生成的。
（5）多重背景只能声明为静态变量（声明类型为“STAT”）。

6.5 组织块与中断处理

组织块是操作系统与用户程序之间的接口。用组织块可以响应延时中断、外部硬件中断和错误处理等。

6.5.1 中断的基本概念

1．中断过程

中断处理用来实现对特殊内部事件或外部事件的快速响应。CPU 检测到中断请求时，立即响应中断，调用中断源对应的中断程序（OB）。执行完中断程序后，返回被中断的程序。
中断源：I/O 模块的硬件中断，软件中断，例如日期时间中断、延时中断、循环中断和编程错误引起的中断。
中断源的中断优先级与中断程序的嵌套调用。操作系统对现场进行保护。被中断的 OB 的局域数据压入 L 堆栈、I 堆栈（中断堆栈）、B 堆栈（块堆栈）。

2．组织块的分类

组织块只能由操作系统起动，它由变量声明表和用户编写的控制程序组成。
（1）起动组织块 OB100～OB102
（2）循环执行的组织块
（3）定期执行的组织块
（4）事件驱动的组织块
延时中断、硬件中断、异步错误中断 OB80～OB87，同步错误中断 OB121 和 OB122。

3．中断的优先级

下面是优先级的顺序（后面的比前面的优先）：背景循环、主程序扫描循环、日期时间中断、时间延时中断、循环中断、硬件中断、多处理器中断、I/O 冗余错误、异步故障(OB80~87)、启动和 CPU 冗余，背景循环的优先级最低。

4．对中断的控制

日期时间中断和延时中断有专用的允许处理中断和禁止中断的系统功能（SFC）。
SFC 39“DIS_INT”用来禁止所有的中断、某些优先级范围的中断、或指定的某个中断。
SFC 40“EN_INT”用来激活（使能）新的中断和异步错误处理。如果用户希望忽略中断，可以下载一个只有块结束指令 BEU 的空的 OB。
SFC 41“DIS_AIRT”延迟处理比当前优先级高的中断和异步错误。SFC 42“EN_ AIRT”允许立即处理被 SFC 41 暂时禁止的中断和异步错误。

6.5.2 组织块的变量声明表

OB 没有背景数据块和静态变量，只有 20 个字节组成的包含 OB 的起动信息的变量声明表（临时变量）。
表 6-6 OB 的变量声明表

字节地址	内容
0	事件级别与标识符，例如 OB40 为 B#16#11，表示硬件中断被激活
1	用代码表示与起动 OB 的事件有关的信息
2	优先级，例如 OB40 的优先级为 16
3	OB 块号，例如 OB40 的块号为 40
4～11	附加信息，例如 OB40 的第 5 字节为产生中断的模块的类型，16#54 为输入模块，16#55 为输出模块；第 6，7 字节组成的字为产生中断的模块的起始地址；第 8～11 字节组

PLC WORLD

成的双字为产生中断的通道号
12～19　OB 被起动的日期和时间（年、月、日、时、分、秒、毫秒与星期）
6.5.3 日期时间中断组织块（OB10～OB17）
CPU 可以使用的日期时间中断 OB 的个数与 CPU 的型号有关。S7-300 只能用 OB10。
可以在某一特定的日期和时间执行一次，也可以从设定的日期时间开始，周期性地重复执行，例如每分钟、每小时、每天、甚至每年执行一次。可以用 SFC 28～SFC 30 取消、重新设置或激活日期时间中断。
1．设置和起动日期时间中断
（1）用 SFC 28“SET_TINT”和 SFC 30“ACT_TINT”设置和激活日期时间中断。
（2）在硬件组态工具中设置和激活。
（3）在硬件组态工具中设置，用 SFC 30“ACT_TINT”激活日期时间中断。
2．调用 SFC 31 “QRY_TINT”查询日期时间中断
3．禁止与激活日期时间中断
用 SFC 29“CAN_TINT”取消（禁止）日期时间中断，用 SFC 28“SET_TINT”重新设置那些被禁止的日期时间中断，用 SFC 30“ACT_TINT”重新激活日期时间中断。
在调用 SFC 28 时，如果参数“OB10_PERIOD_EXE”为十六进制数 W#16#0000，W#16#0201，W#16#0401, W#16#1001, W#16#1201, W#16#1401, W#16#1801 和 W#16#2001，分别表示执行一次、每分钟、每小时、每天、每周、每月、每年和月末执行一次。
6.5.4 延时中断组织块
延时中断以 ms 为单位定时。CPU 可以使用的延时中断 OB 的个数与 CPU 的型号有关。
用 SFC 32“SRT_DINT”起动，经过设定的时间触发中断，调用 SFC 32 指定的 OB。延时中断可以用 SFC 33“CAN_DINT”取消。用 SFC 34“QRY_DINT”查询延时中断的状态。
6.5.5 循环中断组织块
CPU 可以使用的日期时间中断 OB 的个数与 CPU 的型号有关。
设 OB38 和 OB37 的时间间隔分别为 10ms 和 20ms，它们的相位偏移分别为 0ms 和 3ms。OB38 分别在 10 ms，20ms，……，60ms 时产生中断，而 OB37 分别在 t = 23ms，43ms，63ms 时产生中断。可以用 SFC 40 和 SFC 39 来激活和禁止循环中断。
表 6-7 循环 OB 默认参数
OB 号　时间间隔　优先级　OB 号　时间间隔　优先级
OB30　5s　7　OB35　100ms　12
OB31　2s　8　OB36　50ms　13
OB32　1s　9　OB37　20ms　14
OB33　500ms　10　OB38　10ms　15
OB34　200ns　11
6.5.6 硬件中断组织块
硬件中断组织块（OB40～OB47）用于快速响应信号模块（SM，即输入/输出模块）、通信处理器（CP）和功能模块（FM）的信号变化。
硬件中断被模块触发后，操作系统将自动识别是哪一个槽的模块和模块中哪一个通道产生的硬件中断。硬件中断 OB 执行完后，将发送通道确认信号。

如果正在处理某一中断事件，又出现了同一模块同一通道产生的完全相同的中断事件，新的中断事件将丢失。
如果正在处理某一中断信号时同一模块中其他通道或其它模块产生了中断事件，当前已激活

PLC WORLD

的硬件中断执行完后，再处理暂存的中断。

用 PLCSIM 的菜单命令“Execute→Trigger Error OB→Hardware Interrupt (OB40-OB47)…”打开“Hardware Interrupt (OB40-OB47)”对话框，输入模块的起始地址和位地址 0。按“Apply”键触发指定的硬件中断，按“OK”键将执行与“Apply”键同样的操作，同时退出对话框。

6.5.7 起动时使用的组织块

1．CPU 模块的启动方式

（1）暖启动(Warm Restart)

S7-300 CPU（不包括 CPU 318）只有暖起动。过程映像数据以及非保持的 M/T/C。有保持功能的 M/T/C/DB 将保留原数值。模式开关扳由 STOP 板到 RUN 位置。

（2）热起动(Hot Restart 仅 S7-400 有)

在 RUN 状态时如果电源突然丢失，然后又重新上电，从上次 RUN 模式结束时程序被中断之处继续执行，不对计数器等复位。

（3）冷启动(Cold Restart，CPU 417 和 CPU 417H)

冷启动时，过程数据区的 I, Q, M, T, C, DB 等被复位为零。模式开关扳到 MRES 位置。

2．启动组织块（OB100～OB102）

在暖起动、热起动或冷起动时，操作系统分别调用 OB100，OB101 或 OB102。

6.5.8 异步错误组织块

1．错误处理概述

S7-300/400 有很强的错误（或称故障）检测和处理能力。PLC 内部的功能性错误或编程错误，而不是外部设备的故障。CPU 检测到错误后，操作系统调用对应的组织块，用户可以在组织块中编程，对发生的错误采取相应的措施。对于大多数错误，如果没有给组织块编程，出现错误时 CPU 将进入 STOP 模式。

表 6-8 错误处理组织块

OB 号	错误类型	优先级
OB 70	I/O 冗余错误（仅 H 系列 CPU）	25
OB 72	CPU 冗余错误（仅 H 系列 CPU）	28
OB 73	通信冗余错误（仅 H 系列 CPU）	25
OB 80	时间错误	26
OB 81	电源故障	26/28
OB 82	诊断中断	
OB 83	插入/取出模块中断	
OB 84	CPU 硬件故障	
OB 85	优先级错误	
OB 86	机架故障或分布式 I/O 的站故障	
OB 87	通信错误	
OB 121	编程错误	引起错误的 OB 的优先级
OB 122	I/O 访问错误	

为避免发生某种错误时 CPU 进入停机状态，可以在 CPU 中建立一个对应的空的组织块。

2．错误的分类

被 S7 CPU 检测到并且用户可以通过组织块对其进行处理的错误分为两个基本类型：

（1）异步错误

异步错误是与 PLC 的硬件或操作系统密切相关的错误，与程序执行无关。后果严重。异步错误 OB 具有最高等级的优先级，其他 OB 不能中断它们。同时有多个相同优先级的异步错

PLC WORLD

误 OB 出现，将按出现的顺序处理。
（2）同步错误（OB121 和 OB122）
同步错误是与程序执行有关的错误， 其 OB 的优先级与出现错误时被中断的块的优先级相同，即同步错误 OB 中的程序可以访问块被中断时累加器和状态寄存器中的内容。对错误进行处理后，可以将处理结果返回被中断的块。
3．电源故障处理组织块（OB81）
电源故障包括后备电池失效或未安装，S7-400 的 CPU 机架或扩展机架上的 DC 24V 电源故障。电源故障出现和消失时操作系统都要调用 OB81。
4．时间错误处理组织块（OB80）
循环监控时间的默认值为 150ms，时间错误包括实际循环时间超过设置的循环时间、因为向前修改时间而跳过日期时间中断、处理优先级时延迟太多等。
5．诊断中断处理组织块（OB82）
OB82 在下列情况时被调用：有诊断功能的模块的断线故障，模拟量输入模块的电源故障，输入信号超过模拟量模块的测量范围等。错误出现和消失时，操作系统都会调用 OB82。用 SFC 51“RDSYSST”可以读出模块的诊断数据。
6．插入/拔出模块中断组织块（OB83）
S7-400 可以在 RUN，STOP 或 STARTUP 模式下带电拔出和插入模块，但是不包括 CPU 模块、电源模块、接口模块和带适配器的 S5 模块，上述操作将会产生插入/拔出模块中断。
7．CPU 硬件故障处理组织块（OB84）
当 CPU 检测到 MPI 网络的接口故障、通信总线的接口故障或分布式 I/O 网卡的接口故障时，操作系统调用 OB84。故障消除时也会调用该 OB 块。
8．优先级错误处理组织块（OB85）
在以下情况下将会触发优先级错误中断：
（1）产生了一个中断事件，但是对应的 OB 块没有下载到 CPU；
（2）访问一个系统功能块的背景数据块时出错。
（3）刷新过程映像表时 I/O 访问出错，模块不存在或有故障。
9．机架故障组织块（OB86）
（1）机架故障，例如找不到接口模块或接口模块损坏，或者连接电缆断线；
（2）机架上的分布式电源故障；
（3）在 SINEC L2-DP 总线系统的主系统中有一个 DP 从站有故障。
10．通信错误组织块（OB87）
（1）接收全局数据时，检测到不正确的帧标识符（ID）；
（2）全局数据通信的状态信息数据块不存在或太短；
（3）接收到非法的全局数据包编号。
6.5.9 同步错误组织块
1．同步错误
同步错误是与执行用户程序有关的错误，OB121 用于对程序错误的处理；OB122 用于处理模块访问错误。
同步错误 OB 的优先级与检测到出错的块的优先级一致。
同步错误可以用 SFC 36“MASK_FLT”来屏蔽，用错误过滤器中的一位用来表示某种同步错误是否被屏蔽。错误过滤器分为程序错误过滤器和访问错误过滤器，分别占一个双字。屏蔽后的错误过滤器可以读出。
表 6-9 SFC 36“MSK_FLT”的局域变量表
参数　声明　数据类型　存储区　描述

PLC WORLD

PRGFLT_SET_MASK INPUT DWORD I,Q,M,D,L,常数 要屏蔽的程序错误
ACCFLT_SET_MASK INPUT DWORD I,Q,M,D,L,常数 要屏蔽的访问错误
RET_VAL OUTPUT INT I,Q,M,D,L 错误信息返回值
PRGFLT_MASKED OUTPUT DWORD I,Q,M,D,L 被屏蔽的程序错误
ACCFLT_MASKED OUTPUT DWORD I,Q,M,D,L 被屏蔽的访问错误
调用 SFC 37“DMSK_FLT”并且在当前优先级被执行完后，将解除被屏蔽的错误。
可以用 SFC 38“READ_ERR”读出已经发生的被屏蔽的错误。
2．编程错误组织块（OB121）
出现编程错误时，CPU 的操作系统将调用 OB121。局域变量 OB121_SW_FLT 给出了错误代码，可以查看《S7-300/400 的系统软件和标准功能》中 OB121 部分的错误代码表。
3．I/O 访问错误组织块（OB122）
STEP 7 指令访问有故障的模块，例如直接访问 I/O 错误（模块损坏或找不到），或者访问了一个 CPU 不能识别的 I/O 地址，此时 CPU 的操作系统将会调用 OB122。
6.5.10 背景组织块
CPU 可以保证设置的最小扫描循环时间，如果它比实际的扫描循环时间长，在循环程序结束后 CPU 处于空闲的时间内可以执行背景组织块（OB90）。背景 OB 的优先级为 29（最低）。

第七章 计算机通信网络与 S7-300/400 的通信功能
7.1 计算机通信方式与串行通信接口
7.1.1 计算机的通信方式

1．并行通信与串行通信
2．异步通信与同步通信

某字符中包含以下 8 个数据位：
1 0 1 0 0 0 1 1
如果选择了偶校验，奇偶校验位将是 0。如果选择了奇校验，奇偶校验位将是 1。如果选择不进行奇偶校验，传输时没有校验位，也不进行奇偶校验检测。
同步通信以字节为单位，每次传送 1～2 个同步字符、若干个数据字节和校验字符。通过调制解调的方式在数据流中提取出同步信号，使接收方得到与发送方同步的接收时钟信号。
单工通信方式只能沿单一方向传输数据，双工通信方式的信息可以沿两个方向传送，每一个站既可以发送数据，也可以接收数据。双工方式又分为全双工和半双工。

4．传输速率
传输速率（又称波特率）的单位是波特，其符号为 bit/s 或 bps。
7.1.2 串行通信接口的标准
1．RS-232C

最大通信距离为 15m，最高传输速度速率为 20kbit/s，只能进行一对一的通信。
2．RS-422A 与 RS-485
RS-422A 采用平衡驱动、差分接收电路(见图 7-6)，共模信号可以互相抵消。
RS-422A 在最大传输速率 (10 Mbit/s) 时，允许的最大通信距离为 12m。传输速率为 100 kbit/s 时，最大通信距离为 1 200m，一台驱动器可以连接 10 台接收器。

PLC WORLD

3．RS-485
RS-485 为半双工，只有一对平衡差分信号线，最多可以有 128 个站。

7.2 计算机通信的国际标准
7.2.1 开放系统互连模型

1．物理层
物理层为用户提供建立、保持和断开物理连接的功能，RS-232C，RS-422A / RS-485。
2．数据链路层
数据以帧（frame）为单位传送，每一帧包含一定数量的数据和必要的控制信息，例如同步信息、地址信息、差错控制和流量控制信息。数据链路层负责在两个相邻节点间的链路上，实现差错控制、数据成帧、同步控制等。
7．应用层
应用层作为 OSI 的最高层，为用户的应用服务提供信息交换，为应用接口提供操作标准。
7.2.2 IEEE 802 通信标准
数据链路层分解为逻辑链路控制层(LLC)、媒体访问层(MAC)，数据链路层是一条链路(Link)两端的两台设备进行通信时所共同遵守的规则和约定。
1．CSMA/CD
竞争发送、广播式传送、载体监听、冲突检测、冲突后退和再试发送。
2．令牌总线
3．令牌环
7.2.3 现场总线及其国际标准
IEC (国际电工委员会) 对现场总线(Fieldbus)的定义是“安装在制造和过程区域的现场装置与控制室内的自动控制装置之间的数字式、串行、多点通信的数据总线称为现场总线”。
IEC 61158 是迄今为止制订时间最长、意见分歧最大的国际标准之一。制订时间超过 12 年，先后经过 9 次投票，在 1999 年底获得通过。IEC 61158 最后容纳了 8 种互不兼容的协议：
类型 1：原 IEC61158 技术报告，即现场总线基金会（FF）的 H1；
类型 2：Control Net（美国 Rockwell 公司支持）；
类型 3：PROFIBUS（德国西门子公司支持）；
类型 4：P-Net（丹麦 Process Data 公司支持）；
类型 5：FF 的 HSE（原 FF 的 H2，高速以太网，美国 Fisher Rosemount 公司支持）；
类型 6：Swift Net（美国波音公司支持）；
类型 7：WorldFIP（法国 Alstom 公司支持）；
类型 8：Interbus（德国 Phoenix contact 公司支持）。
各类型将自己的行规纳入 IEC 61158，且遵循两个原则：
(1) 不改变 IEC 61158 技术报告的内容。
(2)八种类型都是平等的，类型 2～8 都对类型 1 提供接口，标准并不要求类型 2～8 之间提供接口。
IEC 62026 是供低压开关设备与控制设备使用的控制器电气接口标准，于 2000 年 6 月通过。它包括：
(1) IEC 62026-1：一般要求；
(2) IEC 62026-2：执行器传感器接口 AS-i (Actuator Sensor Interface)；
(3) IEC 62026-3：设备网络 DN (Device Network)；
(4) IEC 62026-4：Lonworks（Local Operating Networks）总线的通信协议 LonTalk；

PLC WORLD

(5) IEC 62026-5：灵巧配电（智能分布式）系统 SDS (Smart Distributed System)；
(6) IEC 62026-6：串行多路控制总线 SMCB(Serial Multiplexed Control Bus)。

7.3 S7-300/400 的通信功能

7.3.1 工厂自动化网络结构

1．现场设备层

主要功能是连接现场设备，例如分布式 I/O、传感器、驱动器、执行机构和开关设备等，完成现场设备控制及设备间连锁控制。

2．车间监控层

车间监控层又称为单元层，用来完成车间主生产设备之间的连接，包括生产设备状态的在线监控、设备故障报警及维护等。还有生产统计、生产调度等功能。传输速度不是最重要的，但是应能传送大容量的信息。

3．工厂管理层

车间操作员工作站通过集线器与车间办公管理网连接，将车间生产数据送到车间管理层。车间管理网作为工厂主网的一个子网，连接到厂区骨干网，将车间数据集成到工厂管理层。

7.3.2 S7-300/400 的通信网络

1．通过多点接口(MPI)协议的数据通信

MPI 是多点接口（Multi Point Interface）的简称，MPI 的物理层是 RS-485，通过 MPI 能同时连接运行 STEP 7 的编程器、计算机、人机界面(HMI)及其他 SIMATIC S7，M7 和 C7。通过 MPI 接口实现全局数据(GD)服务，周期性地相互进行数据交换。

2．PROFIBUS

用于车间级监控和现场层的通信系统，开放性。PROFIBUS-DP 与分布式 I/O。最多可以与 127 个网络上的节点进行数据交换。网络中最多可以串接 10 个中继器来延长通信距离。使用光纤作通信介质，通信距离可达 90 km。

3．工业以太网

西门子的工业以太网符合 IEEE 802.3 国际标准，通过网关来连接远程网络。
10M/100M bit/s，最多 1024 个网络节点，网络的最大范围为 150km。
采用交换式局域网，每个网段都能达到网络的整体性能和数据传输速率，电气交换模块与光纤交换模块将网络划分为若干个网段，在多个网段中可以同时传输多个报文。本地数据通信在本网段进行，只有指定的数据包可以超出本地网段的范围。
全双工模式使一个站能同时发送和接收数据，不会发生冲突。传输速率到 20 Mbit/s 和 200 Mbit/s。可以构建环形冗余工业以太网。最大的网络重构时间为 0.3 秒。
自适应功能自动检测出信号传输速率（10 M 或 100 Mbit/s）。
自协商是高速以太网的配置协议，通过协商确定数据传输速率和工作方式。
使用 SNMP-OPC 服务器对支持 SNMP 协议的网络设备进行远程管理。

4．点对点连接

点对点连接（Point-to-Point Connections）可以连接S7 PLC 和其他串口设备。使用 CP 340，CP 341、CP 440、CP 441 通信处理模块，或 CPU 31xC-2PtP 集成的通信接口。
接口有 20mA（TTY），RS-232C 和 RS-422A/RS-485。通信协议有 ASCII 驱动器、3964（R）和 RK 512（只适用于部分 CPU）。
使用通信软件 PRODAVE 和编程用的 PC/MPI 适配器，通过 PLC 的 MPI 编程接口，可以实现计算机与 S7-300/400 的通信。

5．通过 AS-i 网络的过程通信

PLC WORLD

AS-i 是执行器-传感器接口（Actuator Sensor Interface）的简称，位于最底层。
AS-i 每个网段只能有一个主站。AS-i 所有分支电路的最大总长度为 100m，可以用中继器延长。可以用屏蔽的或非屏蔽的两芯电缆，支持总线供电。
DP/AS-i 网关（Gateway）用来连接 PROFIBUS-DP 和 AS-i 网络。
CP 342-2 最多可以连接 62 个数字量或 31 个模拟量 AS-i 从站。最多可以访问 248 个 DI 和 186 个 DO。可以处理模拟量值。
西门子的“LOGO！”微型控制器可以接入 AS-i 网络，西门子提供各种各样的 AS-i 产品。

7.4 MPI 网络与全局数据通信

7.4.1 MPI 网络

周期性地相互交换少量的数据，最多 15 个 CPU。编程设备、人机接口和 CPU 的默认地址分别为 0，1，2。
MPI 默认的传输速率为 187.5 k bit/s 或 1.5 M bit/s，与 S7-200 通信 19.2 k bit/s。相邻节点间的最大传送距离为 50m，加中继器后为 1000m，使用光纤和星形连接时为 23.8 km。

7.4.2 全局数据包

参与全局数据包交换的 CPU 构成了全局数据环（GD circle）。可以建立多个 GD 环。
具有相同的发送者和接收者的全局数据集合成一个全局数据包。数据包中的变量有变量号。例如 GD1.2.3 是 1 号 GD 环、2 号 GD 包中的 3 号数据。
S7-300 CPU 可以建立 4 个全局数据环，每个环中一个 CPU 只能发送和接收一个数据包，每个数据包最多包含 22 个数据字节。
S7-400 CPU 可以建立的全局数据环个数与 CPU 的型号有关（16～64 个），每个环中一个 CPU 只能发送一个数据包和接收两个数据包，每个数据包最多包含 54 个数据字节。

7.4.3 MPI 网络的组态

在 SMATIC 管理器中生成 3 个站，它们的 CPU 分别为 CPU 413-1, CPU 313C 和 CPU 312C。
双击 MPI 图标，打开 NetPro 工具，打开 CPU 的属性设置对话框，设置 MPI 站地址。
将 CPU 就被连接到 MPI（1）子网上。
保存 CPU 的配置参数，用点对点的方式将它们分别下载到各 CPU 中。
用 PROFIBUS 电缆连接 MPI 节点可以用管理器的“Accessible Nodes”功能来测试可以访问的节点。

图 7-13 MPI 网络的组态

7.4.4 全局数据表

1．生成和填写 GD 表
在“NetPro”窗口中用右键点击 MPI 网络线，在弹出的窗口中执行执行菜单命令“Options →Define Global Data（定义全局数据）”同样的命令。
在表的第一行输入 3 个 CPU 的名称。
鼠标右键点击 CPU 413-1 下面的单元（方格），在出现的菜单中选择“Sender”（发送者），输入要发送的全局数据的地址 MW0。在每一行中只能有一个 CPU 发送方。同一行中各个单元的字节数应相同。
点击 CPU 313C 下面的单元，输入 QW0，该格的背景为白色，表示 CPU 313C 是接收站。

图 7-14 全局数据表

MB20:4 表示 MB20 开始的 4 个字节。如果 GD 包由若干个连续的数据区组成，一个连续的数据区占用的空间为数据区内的字节数加上两个头部说明字节。一个单独的双字占 6 个字

PLC WORLD

节，一个单独的字占 4 个字节，一个单独的字节占 3 个字节，一个单独的位也占 3 个字节，例如 DB2.DBB0:10 和 QW0:5 一共占用 22 个字节。

发送方 CPU 自动地周期性地将指定地址中的数据发送到接收方指定的地址区中。完成全局数据表的输入后，应执行菜单命令“GD Table→Compile…”，对它进行第一次编译。

2．设置扫描速率和状态双字的地址

执行菜单命令“View→Scan Rates”，每个数据包将增加标有“SR”的行，扫描速率单位是 CPU 的循环扫描周期，S7-300 默认的扫描速率为 8，S7-400 的为 22，用户可以修改。S7-400 的扫描速率为 0，表示是事件驱动的 GD 发送和接收。

图 7-15 第一次编译后的全局数据表

GD 数据传输的状态双字用来检查数据是否被正确地传送，执行菜单命令“View→Status”，在出现的 GDS 行中可以给每个数据包指定一个用于状态双字的地址。最上面一行的全局状态双字 GST 是各 GDS 行中的状态双字相“与”的结果。

设置好扫描速率和状态字的地址后，应对全局数据表进行第二次编译。将配置数据下载到 CPU 中，以后可以自动交换数据。