

# SIMATIC

## S7-300/400 系统软件和 标准功能 - 卷 1/2

参考手册

### 前言

组织块	1
SFC 的公共参数	2
复制功能和块功能	3
用于控制程序执行的 SFC	4
用于处理系统时钟的 SFC	5
用于处理运行时间计时器的 SFC	6
用于传送数据记录的 SFC/SFB	7
符合 PNO AK 1131 的 DPV1 SFB	8
用于处理时间中断的 SFC	9
用于处理延时中断的 SFC	10
用于处理同步错误的 SFC	11
用于处理中断和异步错误的 SFC	12
用于诊断的 SFC	13
用于更新过程映像和处理位域的 SFC 和 SFB	14
用于寻址模块的系统功能	15
用于分布式 I/O 或 PROFINET IO 的 SFC	16
PROFINet	17
用于 PROFINET CPU 的 SFC 和 SFB	18

05/2017

A5E41730563-AA

## 法律资讯

### 警告提示系统

为了您的人身安全以及避免财产损失，必须注意本手册中的提示。人身安全的提示用一个警告三角表示，仅与财产损失有关的提示不带警告三角。警告提示根据危险等级由高到低如下表示。

#### 危险

表示如果不采取适当的预防措施，**可能导致死亡或严重的人身伤害。**

#### 警告

表示如果不采取适当的预防措施，**可能导致死亡或严重的人身伤害。**

#### 注意

表示如果不采取适当的预防措施，**可能导致轻微的人身伤害。**

#### 注意

表示如果不采取适当的预防措施，**可能造成财产损失。**

当出现多个危险等级的情况下，每次总是使用最高等级的警告提示。如果在某个警告提示中带有警告可能导致人身伤害的警告三角，则可能在该警告提示中另外还附带有可能导致财产损失的警告。

### 合格的专业人员

仅允许安装和驱动与本文件相关的附属设备或系统。设备或系统的调试和运行仅允许由**合格的专业人员**进行。本文件安全技术提示中的合格专业人员是指根据安全技术标准具有从事进行设备、系统和电路的运行，接地和标识资格的人员。

### 按规定使用 Siemens 产品

注意下列各项：

#### 警告

Siemens 产品只允许用于目录和相关技术文件中规定的使用情况。如果要使用其他公司的产品和组件，必须得到 Siemens 推荐和允许。正确的运输、储存、组装、装配、安装、调试、操作和维护是产品安全、正常运行的前提。必须遵守允许的环境条件。必须注意相关文件中的提示。

### 商标

所有带有标记符号 © 的都 Siemens AG 的注册商标。标签中的其他符号可能是一些其他商标，这是出于保护所有者的权利的目地由第三方使用而特别标示的。

### 免责声明

我们已检查过本手册中的内容与所描述的硬件和软件相符。由于差错在所难免，我们不能保证完全一致。我们会定期审查本手册中的信息，并在后续版本中进行必要的更正。

# 前言

## 用途

本手册综述了在 S7-300 和 S7-400 的 CPU、诊断数据、系统状态列表(SZL)和事件的操作系统中所包含的组织块(OB)、系统功能(SFC)、系统和标准功能块(SFB)及 IEC 功能。

---

### 注意

请参考下列文件中的参考书目 - "S7-300 自动化系统 CPU 规范 : CPU 31xC 与 CPU 31x"和 "S7-300 自动化系统 CPU 规范 : CPU 312IFM - 318-2 DP"**/70/** , 或"自动化系统 S7-400 : CPU 规范"参考手册 **/101/** ,或指令列表 :*S7-400 可编程控制器***/102//102/**(适用于您的 CPU 的任意版本) ,以了解各个 CPU 上可以使用的功能和块的详细资料。特定 CPU 的 CFB 属性和 S7 发信功能在**/70/**和**/101/**中进行描述。

---

欲了解有关 CPU 操作系统、程序设计及 CPU 的通讯和诊断能力的内容 , 请参见"*配置硬件和通讯连接 STEP 7 V5.5*"手册**/234/** 关于如何在程序中调用功能和功能块的说明参见语言描述部分。

使用 STEP 7 标准软件为所有这些功能编程 , 并分配参数。关于如何使用本软件的说明请参见"*使用 STEP 7 V5.5 编程*"手册**/231/** 和 STEP 7 在线帮助。

## 目标用户

本手册的目标读者为 : 熟悉控制过程、负责为可编程逻辑控制器编写程序的程序员和工程师。

---

## 在线帮助

集成于软件中的在线帮助是对本手册第 1 卷和第 2 卷的补充。提供在线帮助的目的在于，在使用软件时提供详细的支持。

该帮助系统通过一些界面集成于软件中：

- 在**帮助**菜单中有多个菜单命令可供选择：**目录**命令打开关于 Step 7 帮助信息的索引。
- **使用帮助**提供了关于使用在线帮助的详细指示。
- 上下文相关帮助提供关于当前语境(例如，打开的对话框或激活的窗口)的信息。可以通过单击"帮助"按钮或按下 F1 键打开上下文相关的帮助。
- 状态栏提供另一种形式的上下文相关的帮助。当鼠标指针放在菜单命令上时，它显示每个菜单命令的一个简短解释。
- 当鼠标指针在图标上停留一小段时间时，还为工具栏中的每个图标显示一个简要解释。

如果喜欢按打印格式阅读在线帮助信息，则可以打印输出单个帮助主题，或整个在线帮助。

本手册是从基于 HTML 的 STEP 7 帮助中摘录下来的。由于该手册和在线帮助的结构几乎一致，因而可以很容易地在该手册和在线帮助间进行切换。

## 对文档的反馈

为了帮助我们为您和 STEP 7 未来的用户提供尽可能最好的文档，我们需要您的支持。如果您有任何关于本手册或在线帮助的批评或建议，请填写位于本手册最后的调查表，并发送到所提供的地址。请注明您本人对文档的等级评价。

## 其它手册

下列手册对各种 S7-300 CPU 和 S7-400 CPU 以及 S7-300 模块和 S7-400 模块进行了描述：

- 欲了解 S7-300 可编程逻辑控制器的内容，请参考手册："PLC S7-300，CPU 规范，CPU 312 IFM 至 CPU 318-2 DP 及 S7-300 CPU 31xC 和 CPU 31x：技术规范" /70/，"S7-300 S7-300 模块数据" /71/和指令列表/72/中。
- 欲了解 S7-400 可编程逻辑控制器的内容，请参考手册："S7-400 自动化系统：模块数据" /101/和指令列表/102/中。

---

## 如何使用本手册

本手册包含了下列主题：

- 第 1 章说明所有组织块的功能。
- 第 2 章描述公用参数 RET\_VAL、REQ 和 BUSY。
- 第 3 章至第 32 章描述 SFC、SFB 和 IEC-FC。
- 第 33 章至第 36 章包含对诊断数据结构的描述、对本手册中描述的 SZL-ID、可能的事件、SFC 列表、SFB 和 FC 的概述，以及对 SDB 的概述。
- 参考书目包含了更多其它手册的列表。
- 词汇表解释了重要的术语。
- 索引帮助您迅速地定位文本部分和主题部分。

## 约定

对其它手册和文档的引用用两个斜线/.../加数字表示。这些数字指的是参考书目中列出的手册的标题。

## 特殊注意事项

系统功能可能被中断。如果有任何适用于某些 SFC 或某些情况的限制，在特定的 SFC 的描述中有解释。

## 更多支持

如果有任何技术问题，请联系西门子代表或代理商。

您可以在下列网页中查找联系人：

<http://www.siemens.com/automation/partner>

可以在下列网址上找到单个 SIAMTIC 产品和系统的技术文档指南：

<http://www.siemens.com/simatitechdokuportal>

可以在下列网址上获得在线目录和订货系统：

<http://mall.automation.siemens.com/>

## 培训中心

西门子提供了很多培训教程，帮助您熟悉 SIMATIC S7 自动化系统。

请联系当地的培训中心，或位于德国纽伦堡(D 90026)的培训总部，以获取详细信息。

Internet : <http://sitrain.automation.siemens.com/sitrainworld/>

---

## 技术支持

您可访问"技术支持"来了解所有的工业自动化和驱动技术产品

- 通过网站请求支持

<http://www.siemens.com/automation/support-request>

关于技术支持的更多信息请参见 Internet 网页：

<http://www.siemens.com/automation/service>

## Internet 服务和支持

除文档以外，还在 Internet 上在线提供了专业技术信息，网址如下：

<http://www.siemens.com/automation/service&support>

可在其中查找下列内容：

- 公司简讯，经常提供产品的最新信息。
- 相应文档资料，可通过"服务和支持"中的搜索功能查找。
- 论坛，世界各地的用户和专家可以在此交流经验。
- 您当地的关于工业自动化和驱动技术的销售代表。
- 关于现场服务、维修、备件和查阅等信息。

## 安全提示：

西门子在工业安全功能方面提供产品和解决方案，旨在支持工厂、系统、机器和网络的安全运行。

为保护工厂、系统、机器和网络免受网络攻击威胁，必须实施并不断保持全方位的先进工业安全理念。西门子产品和解决方案仅仅是其中的一个方面。

客户应负责保护其工厂、系统、机器和网络免受未经授权的访问。系统、机器和组件仅可连接企业网络，且只能在必要时且相应安全措施(例如，使用防火墙和网络分段)到位的情况下连接互联网。

此外，还应考虑西门子在相应安全措施方面的指导。有关工业安全的更多信息，请访问

<http://www.siemens.com/industrialsecurity>。

西门子产品和解决方案经过不断发展，安全性日趋完善。西门子强烈建议您尽快应用产品更新，并始终使用最新的产品版本。如果使用不再受支持的产品版本，并且未能应用最新的更新，则会增加客户受到网络攻击的危险。

要时刻了解产品更新，请订阅西门子工业安全 RSS 信息源

<http://www.siemens.com/industrialsecurity>。

# 目录

<b>1</b>	<b>组织块.....</b>	<b>11</b>
1.1	组织块(OB)概述.....	11
1.2	程序周期组织块(OB1).....	14
1.3	时间中断组织块(OB10 至 OB17).....	16
1.4	延时中断组织块(OB20 至 OB23).....	20
1.5	循环中断组织块(OB30 至 OB38).....	22
1.6	硬件中断组织块(OB40 至 OB47).....	24
1.7	状态中断 OB (OB55).....	26
1.8	更新中断 OB (OB56).....	29
1.9	制造商特定中断 OB (OB57).....	32
1.10	多值计算中断组织块(OB60).....	35
1.11	同步周期性中断 OB (OB61 至 OB64).....	37
1.12	技术同步中断 OB (OB65).....	38
1.13	I/O 冗余错误 OB (OB70).....	39
1.14	CPU 冗余错误 OB (OB72).....	41
1.15	通信冗余错误 OB (OB73).....	44
1.16	时间错误组织块(OB80).....	45
1.17	电源错误组织块(OB81).....	49
1.18	诊断中断组织块(OB82).....	51
1.19	插入/删除模块中断组织块(OB83).....	54
1.20	CPU 硬件故障组织块(OB84).....	59
1.21	优先级错误组织块(OB85).....	61
1.22	机架故障组织块(OB86).....	65
1.23	通信错误组织块(OB87).....	70
1.24	处理中断 OB (OB88).....	72
1.25	背景组织块(OB90).....	74
1.26	启动组织块(OB100、OB101 和 OB102).....	76
1.27	编程错误组织块(OB121).....	81
1.28	I/O 访问错误组织块(OB122).....	84
<b>2</b>	<b>SFC 的公共参数.....</b>	<b>87</b>
2.1	通过输出参数 RET_VAL 判断错误.....	87
2.2	异步 SFC 的 REQ、RET_VAL 和 BUSY 参数含义.....	91
<b>3</b>	<b>复制功能和块功能.....</b>	<b>95</b>
3.1	使用 SFC 20 "BLKMOV"复制存储区.....	95
3.2	使用 SFC 81"UBLKMOV"不中断地复制变量.....	98
3.3	使用 SFC 21 "FILL"初始化存储区.....	100
3.4	使用 SFC 22 "CREAT_DB"创建数据块.....	103
3.5	使用 SFC 23 "DEL_DB"删除数据块.....	105
3.6	使用 SFC 24 "TEST_DB"测试数据块.....	107
3.7	使用 SFC 25"COMPRESS"压缩用户存储器.....	108
3.8	使用 SFC 44"REPL_VAL"将替换值传送到累加器 1 中.....	109
3.9	使用 SFC 82 "CREA_DBL"在装载存储器中生成数据块.....	110
3.10	使用 SFC 83 "READ_DBL"从装载存储器的数据块中读取数据.....	113
3.11	使用 SFC 84 "WRIT_DBL"写数据到装载存储器中的数据块.....	115
3.12	使用 SFC 85"CREA_DB"创建数据块.....	117

<b>4</b>	<b>用于控制程序执行的 SFC .....</b>	<b>119</b>
4.1	使用 SFC 43 "RE_TRIGR"再触发循环时间监控.....	119
4.2	使用 SFC 46"STP"将 CPU 切换到 STOP.....	119
4.3	使用 SFC 47"WAIT"延迟执行用户程序.....	120
4.4	使用 SFC 35"MP_ALM"触发一个多值计算中断.....	121
4.5	使用 SFC 104"CiR"控制 CiR.....	122
4.6	使用 SFC 109 "PROTECT"激活和取消激活 CPU 访问保护.....	124
4.7	用 SFB 73 "RCVREC"接收数据记录.....	127
4.8	用 SFB 74 "PRVREC"提供数据记录.....	130
<b>5</b>	<b>用于处理系统时钟的 SFC .....</b>	<b>133</b>
5.1	使用 SFC 0 "SET_CLK"设定 TOD.....	133
5.2	使用 SFC 1 "READ_CLK"读取时间.....	134
5.3	使用 SFC 48 "SNC_RTCB"同步子时钟.....	135
5.4	使用 SFC 100 "SET_CLKS"设定时间日期和 TOD 状态.....	136
<b>6</b>	<b>用于处理运行时间计时器的 SFC.....</b>	<b>139</b>
6.1	运行时间定时器.....	139
6.2	使用 SFC 101 "RTM"控制运行时间定时器.....	140
6.3	使用 SFC 2 "SET_RTM"设置运行时间定时器.....	142
6.4	使用 SFC 3 "CTRL_RTM"启动和停止运行时间定时器.....	143
6.5	使用 SFC 4 "READ_RTM"读取运行时间定时器.....	144
6.6	使用 SFC 64 "TIME_TCK"读取系统时间.....	145
<b>7</b>	<b>用于传送数据记录的 SFC/SFB.....</b>	<b>147</b>
7.1	数据记录的写和读.....	147
7.2	使用 SFC 54 "RD_DPARM"读取定义的参数.....	149
7.3	使用 SFC 102 "RD_DPARA"读取预定义参数.....	150
7.4	使用 SFC 55 "WR_PARM"写动态参数.....	151
7.5	使用 SFC 56 "WR_DPARM"写缺省参数.....	153
7.6	使用 SFC 57 "PARM_MOD"为模块分配参数.....	154
7.7	使用 SFC 58 "WR_REC"写数据记录.....	157
7.8	使用 SFC 59 "RD_REC"读数据记录.....	159
7.9	SFC 55 到 59 的进一步故障信息.....	163
7.10	使用 SFB 81 "RD_DPAR"读取预定义的参数".....	164
<b>8</b>	<b>符合 PNO AK 1131 的 DPV1 SFB.....</b>	<b>167</b>
8.1	用 SFB 52 "RDREC"读取数据记录.....	167
8.2	用 SFB53 "WRREC"写入数据记录.....	169
8.3	用 SFB 54 "RALRM"接收中断.....	171
8.4	用 SFB 75 "SALRM"向 DP 主站发送中断.....	187
<b>9</b>	<b>用于处理时间中断的 SFC .....</b>	<b>193</b>
9.1	处理时间中断.....	193
9.2	SFC 28 到 31 的特征.....	195
9.3	使用 SFC 28 "SET_TINT"设置时间中断.....	197
9.4	使用 SFC 29 "CAN_TINT"取消时间中断.....	198
9.5	使用 SFC 30 "ACT_TINT"激活时间中断.....	199
9.6	使用 SFC 31 "QRY_TINT"查询时间中断.....	200
<b>10</b>	<b>用于处理延时中断的 SFC .....</b>	<b>203</b>
10.1	处理延时中断.....	203



10.2	使用 SFC 32 "SRT_DINT"启动延时中断 .....	205
10.3	使用 SFC 34 "QRY_DINT"查询延时中断 .....	206
10.4	使用 SFC 33 "CAN_DINT"取消延时中断 .....	207
<b>11</b>	<b>用于处理同步错误的 SFC .....</b>	<b>209</b>
11.1	屏蔽同步错误 .....	209
11.2	使用 SFC 36 "MSK_FLT"屏蔽同步错误 .....	215
11.3	使用 SFC 37 "DMSK_FLT"取消屏蔽同步错误 .....	216
11.4	使用 SFC 38 "READ_ERR"读取错误寄存器 .....	217
<b>12</b>	<b>用于处理中断和异步错误的 SFC .....</b>	<b>219</b>
12.1	延迟和禁用中断和异步错误 .....	219
12.2	使用 SFC 39 "DIS_IRT"禁用新中断和异步错误的处理 .....	221
12.3	使用 SFC 40 "EN_IRT"启用新中断和异步错误的处理 .....	223
12.4	使用 SFC 41 "DIS_AIRT"延迟更高优先级中断和异步错误的处理 .....	224
12.5	使用 SFC 42 "EN_AIRT"启用更高优先级中断和异步错误的处理 .....	225
<b>13</b>	<b>用于诊断的 SFC .....</b>	<b>227</b>
13.1	系统诊断 .....	227
13.2	使用 SFC 6 "RD_SINFO"读取 OB 启动信息 .....	227
13.3	使用 SFC 51 "RDSYSST"读取系统状态列表或部分列表 .....	230
13.4	使用 SFC 52 "WR_USMSG"将自定义诊断事件写入诊断缓冲区 .....	236
13.5	使用 SFC 78 "OB_RT"确定 OB 程序运行时间 .....	240
13.6	使用 SFC 87 "C_DIAG"诊断当前连接状态 .....	244
13.7	使用 SFC 103 "DP_TOPOL"识别 DP 主站系统的总线拓扑 .....	248
<b>14</b>	<b>用于更新过程映像和处理位域的 SFC 和 SFB .....</b>	<b>251</b>
14.1	使用 SFC 26 "UPDAT_PI"更新过程映像输入表 .....	251
14.2	使用 SFC 27 "UPDAT_PO"更新过程映像输出表 .....	253
14.3	使用 SFC 126 "SYNC_PI"在同步循环中更新过程映像分区输入表 .....	254
14.4	使用 SFC 127 "SYNC_PO"在同步循环中更新过程映像分区 .....	256
14.5	使用 SFC 79 "SET"在 I/O 区域中设置位域 .....	258
14.6	使用 SFC 80 "RSET"复位 I/O 区域中的位域 .....	259
14.7	使用 SFB 32 "DRUM"实现操作序列 .....	260
<b>15</b>	<b>用于寻址模块的系统功能 .....</b>	<b>263</b>
15.1	使用 SFC 5 "GADR_LGC"查询模块的逻辑基址 .....	263
15.2	使用 SFC 49 "LGC_GADR"查询属于逻辑地址的模块插槽 .....	264
15.3	使用 SFC 50 "RD_LGADR"查询模块的所有逻辑地址 .....	267
15.4	用 SFC 70 "GEO_LOG"确定模块的起始地址 .....	268
15.5	用 SFC 71 "LOG_GEO"确定属于逻辑地址的插槽 .....	270
<b>16</b>	<b>用于分布式 I/O 或 PROFINET IO 的 SFC .....</b>	<b>273</b>
16.1	用 SFC 7 "DP_PRAL"触发 DP 主站上的硬件中断 .....	273
16.2	用 SFC 11 "DPSYC_FR"同步 DP 从站组 .....	276
16.3	用 SFC 12 "D_ACT_DP"激活和取消激活 DP 从站/PROFINET IO 设备 .....	281
16.4	用 SFC 13 "DPNRM_DG"读取 DP 从站的诊断数据(从站诊断) .....	286
16.5	用 SFC 14 "DPRD_DAT"读取 DP 标准从站//PROFINET IO 设备的连续数据 .....	289
16.6	用 SFC 15 "DPWR_DAT"向 DP 标准从站/PROFINET IO 设备写入连续数据 .....	292
<b>17</b>	<b>PROFINet .....</b>	<b>295</b>
17.1	SFC 112、113 和 114 背景信息 .....	295
17.2	使用 SFC 112 "PN_IN"更新用于 PROFINET CBA 组件的用户程序接口的输入 .....	298

17.3	使用 SFC 113 "PN_OUT"更新 PROFINET CBA 组件的 PROFINET 接口的输出 .....	299
17.4	使用 SFC 114"PN_DP"更新 DP 互连 .....	300
<b>18</b>	<b>用于 PROFINET CPU 的 SFC 和 SFB .....</b>	<b>301</b>
18.1	使用 SFC99 "WWW"启用或同步用户 Web 页面 .....	301
18.2	使用 SFB104 "IP_CONF"设置 IP 组态 .....	303

# 1 组织块

## 1.1 组织块(OB)概述

### 何为组织块？

组织块(OB)是指 CPU 的操作系统与用户程序之间的接口。OB 用于执行特定的程序段：

- 启动 CPU 时
- 在循环或定时执行过程中
- 出错时
- 发生硬件中断时。

组织块按照已分配的优先级来执行。

### 哪些 OB 可用？

并非所有的 CPU 均可处理 STEP 7 中可用的所有 OB。请参阅操作列表/72/和/102/以确定您 CPU 中所包含的 OB。

## 在何处查找详细信息？

请参阅在线帮助和下列手册以获取更多信息：

- /70/: 此手册包含描述不同 S7-300 CPU 性能的技术数据。
- /101/: 此手册包含描述不同 S7-400 CPU 性能的技术数据。

下表中给出了属于每个 OB 的启动事件和默认优先级。

OB	启动事件	默认优先级	解释
OB1	启动结束或 OB1 结束	1	空闲周期
OB10	时间中断 0	2	未指定默认时间
OB11	时间中断 1	2	
OB12	时间中断 2	2	
OB13	时间中断 3	2	
OB14	时间中断 4	2	
OB15	时间中断 5	2	
OB16	时间中断 6	2	
OB17	时间中断 7	2	
OB20	延时中断 0	3	未指定默认时间
OB21	延时中断 1	4	
OB22	延时中断 2	5	
OB23	延时中断 3	6	
OB30	循环中断 0 (默认时间间隔：5 秒)	7	循环中断
OB31	循环中断 1 (默认时间间隔：2 秒)	8	
OB32	循环中断 2 (默认时间间隔：1 秒)	9	
OB33	循环中断 3 (默认时间间隔：500 毫秒)	10	
OB34	循环中断 4 (默认时间间隔：200 毫秒)	11	
OB35	循环中断 5 (默认时间间隔：100 毫秒)	12	
OB36	循环中断 6 (默认时间间隔：50 毫秒)	13	
OB37	循环中断 7 (默认时间间隔：20 毫秒)	14	
OB38	循环中断 8 (默认时间间隔：10 毫秒)	15	
OB40	硬件中断 0	16	硬件中断
OB41	硬件中断 1	17	
OB42	硬件中断 2	18	
OB43	硬件中断 3	19	
OB44	硬件中断 4	20	
OB45	硬件中断 5	21	
OB46	硬件中断 6	22	
OB47	硬件中断 7	23	
OB55	状态中断	2	DPV1 中断
OB56	更新中断	2	
OB57	制造商特定中断	2	
OB60	SFC35 "MP_ALM"调用	25	多值计算中断
OB61	同步周期性中断 1	25	同步周期性中断
OB62	同步周期性中断 2	25	
OB63	同步周期性中断 3	25	
OB64	同步周期性中断 4	25	

OB	启动事件	默认优先级	解释
OB65	技术同步中断	25	技术同步中断
OB70	I/O 冗余错误(仅在 H CPU 中)	25	冗余错误中断
OB72	CPU 冗余错误(仅在 H CPU 中)	28	
OB73	通信冗余错误 OB (仅在 H CPU 中)	25	
OB80	时间错误	26, 28 <sup>1)</sup>	异步错误中断
OB81	电源故障	26、28 <sup>1)</sup> (对于 S7-300) , 25、28 <sup>1)</sup> (对于 S7-400 和 CPU 318)	
OB82	诊断中断	26、28 <sup>1)</sup> (对于 S7-300) , 25、28 <sup>1)</sup> (对于 S7-400 和 CPU 318)	
OB83	插入/删除模块中断	26、28 <sup>1)</sup> (对于 S7-300) , 25、28 <sup>1)</sup> (对于 S7-400 和 CPU 318)	
OB84	CPU 硬件故障	26、28 <sup>1)</sup> (对于 S7-300) , 25、28 <sup>1)</sup> (对于 S7-400 和 CPU 318)	
OB85	编程错误	26、28 <sup>1)</sup> (对于 S7-300) , 25、28 <sup>1)</sup> (对于 S7-400 和 CPU 318)	
OB86	扩展机架、DP 主站系统或用于分布式 I/O 的站故障	26、28 <sup>1)</sup> (对于 S7-300) , 25、28 <sup>1)</sup> (对于 S7-400 和 CPU 318)	
OB87	通讯错误	26、28 <sup>1)</sup> (对于 S7-300) , 25、28 <sup>1)</sup> (对于 S7-400 和 CPU 318)	
OB88	处理中断	28	
OB90	暖重启或冷重启, 或删除 OB90 中正在执行的块, 或将在 CPU 上装载 OB90 或终止 OB90	29 <sup>2)</sup>	背景周期
OB100	暖启动	27 <sup>1)</sup>	启动
OB101	热启动	27 <sup>1)</sup>	
OB102	冷启动	27 <sup>1)</sup>	
OB121	编程错误	导致错误的 OB 优先级	同步错误中断
OB122	I/O 访问错误	导致错误的 OB 优先级	

1) 优先级27和28在启动的优先级模型中有效。

2) 优先级29对应于优先级0.29。这就意味着后台循环的优先级低于自由循环。

## 1.2 程序周期组织块(OB1)

### 描述

S7 CPU 的操作系统定期执行 OB1。执行 OB1 后，操作系统将再次启动它。完成启动后，将启动 OB1 的循环执行。可调用 OB1 中的其它功能块(FB、SFB)或功能(FC、SFC)。

### 了解 OB1 的操作

在运行期受监视的所有 OB 中，OB1 的优先级最低，换言之，除 OB90 之外的其它所有 OB 均可中断 OB1 的执行。以下事件可导致操作系统调用 OB1：

- 完成启动。
- 已完成 OB1 的执行(上一循环)。

执行 OB1 后，操作系统发送全局数据。重新启动 OB1 之前，操作系统会将过程映像输出表写入输出模块中、更新过程映像输入表以及接收 CPU 的任何全局数据。

S7 监视最大的扫描时间，并确保最大的响应时间。最大扫描时间的值预置为 150ms。您可设置一个新值，也可在用户程序中的任意位置使用 SFC43 "RE\_TRIGR"来重新启动时间监视。如果用户程序超出了 OB1 的最大周期，则操作系统将调用 OB80 (时间错误 OB)；如果 OB80 没有编程，则 CPU 将转为 STOP 模式。

除了监视最大扫描时间外，还可以保证最小扫描时间。操作系统将延迟启动新循环(将过程映像输出表写入输出模块中)，直至达到最小扫描时间为止。

请参阅手册/70/和/101/以了解参数"最大"和"最小"扫描时间的范围。可使用 STEP 7 更改参数设置。

## OB1 的本地数据

下表描述了 OB1 的临时(TEMP)变量。变量名为 OB1 的默认名称。

变量	类型	描述
OB1_EV_CLASS	BYTE	事件等级和标识符：B#16#11：OB1 激活
OB1_SCAN_1	BYTE	<ul style="list-style-type: none"> <li>• B#16#01：完成暖重启</li> <li>• B#16#02：完成热重启</li> <li>• B#16#03：完成主循环</li> <li>• B#16#04：完成冷重启</li> <li>• B#16#05：主站-保留站切换和"停止"上一主站之后新主站 CPU 的首个 OB1 循环</li> </ul>
OB1_PRIORITY	BYTE	优先级 1
OB1_OB_NUMBR	BYTE	OB 编号(01)
OB1_RESERVED_1	BYTE	保留
OB1_RESERVED_2	BYTE	保留
OB1_PREV_CYCLE	INT	上一次扫描的运行时间(ms)
OB1_MIN_CYCLE	INT	自上次启动后的最小周期(ms)
OB1_MAX_CYCLE	INT	从上次启动后的最大周期(ms)
OB1_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

## 1.3 时间中断组织块(OB10 至 OB17)

### 描述

STEP 7 提供了多达八个 OB (OB10 至 OB17)，这些 OB 可单次运行，也可定期运行。可使用 SFC 或 STEP 7 为 CPU 分配参数，以便以下列时间间隔处理这些 OB：

- 单次
- 每分钟
- 每小时
- 每天
- 每周
- 每月
- 每个月末

---

### 注意

对于每月执行的时间中断 OB，只可将 1、2、...28 号作为起始日期。

---

### 了解时间中断 OB 的操作

要启动时间中断，必须先设置中断，然后再将其激活。有以下三种可能的启动方式：

- 自动启动时间中断。一旦使用 STEP 7 设置并激活了时间中断，即会发生此情况。下表说明了使用 STEP 7 激活时间中断的基本可能情况。
- 使用 STEP 7 设置时间中断，然后通过调用程序中的 SFC30 "ACT-TINT"来激活它。
- 通过调用 SFC28 "SET\_TINT"来设置时间中断，然后通过调用 SFC30 "ACT\_TINT"来激活它。

时间间隔	描述
未激活	时间中断未执行，即使将其装载到 CPU 中也是如此。可通过调用 SFC30 来激活它。
仅激活一次	时间 OB 按指定时间运行一次后，即被自动取消。 用户程序可使用 SFC28 和 SFC30 来复位并重新激活 OB。
定期激活	当发生时间中断时，CPU 将根据当前的时间和周期计算时间中断的下次启动的时间。



当前后拨动时钟时，时间中断的特性将如/234/中所述。

---

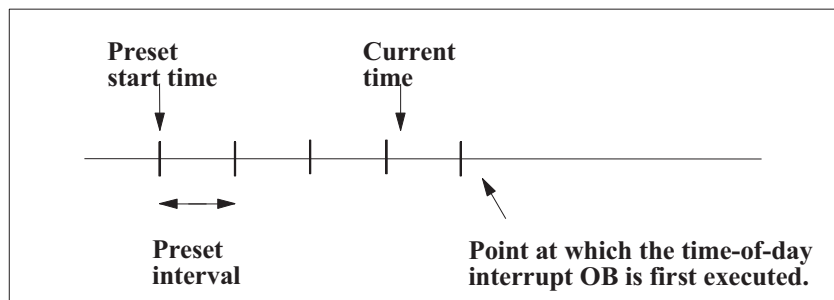
**注意**

如果以如下方式组态时间中断，即单次处理相应的 OB 要，则 DATE\_AND\_TIME 必须为过去的时间(相对于 CPU 的实际时间)。

如果以如下方式组态时间中断，即定期处理相应的 OB，而启动 DATE\_AND\_TIME 为过去的时间，随后将在下次预期的时间处理该时间中断。下图对此进行了说明。

---

可以使用 SFC 39 至 42 禁用或延迟和重新启用时间中断。



### 影响时间中断 OB 的条件

由于时间中断仅以指定的时间间隔发生，因此在执行用户程序期间，某些条件可能会影响 OB 的操作。下表列出了其中的一些条件，并说明了该条件对执行时间中断 OB 的影响。

条件	结果
用户程序调用 SFC29 (CAN_TINT)并取消时间中断。	操作系统清除了时间中断的启动事件(DATE_AND_TIME)。必须再次设置启动事件并在再次调用 OB 之前激活它。
用户程序试图激活时间中断 OB，但未将 OB 加载到 CPU 中。	操作系统将调用 OB85。如果 OB85 尚未编程(装载到 CPU 中)，则 CPU 将转为 STOP 模式。
当同步或更正 CPU 的系统时钟时，您提前设置了时间并跳过时间 OB 的启动事件日期或时间。	操作系统调用 OB80 并对时间 OB 的编号和 OB80 中的启动事件信息进行编码。 随后操作系统将运行一次时间 OB，而不管本应执行此 OB 的次数。OB80 的启动事件信息给出了第一次跳过时间 OB 的 DATE_AND_TIME。
当同步或更正 CPU 的系统时钟时，推后设置了时间以使 OB 的启动事件、日期或时间得以重复。	S7-400-CPU 和 CPU 318： 如果在推后设置时钟之前已激活了时间 OB，则不会再次调用它。 S7-300-CPU：执行时间 OB。
CPU 通过暖重启或冷重启运行。	由 SFC 组态的所有时间 OB 会被改回在 STEP 7 中指定的组态。 如果已为相应 OB 的单个启动组态了时间中断，并使用 STEP 7 对其进行了设置，并将其激活，则当所组态的启动时间为已过去的时间(相对于 CPU 的实时时钟)时，会在暖重启或冷重启操作系统后调用一次 OB。
当发生下一时间间隔的启动事件时，仍执行时间 OB。	操作系统将调用 OB80。如果 OB80 没有编程，则 CPU 转为 STOP 模式。 如果装载了 OB80，则会首先执行 OB80 和时间中断 OB，然后再执行请求的中断。

## 时间中断 OB 的本地数据

下表给出了时间中断 OB 的临时(TEMP)变量。变量名为 OB10 的默认名称。

变量	类型	描述
OB10_EV_CLASS	BYTE	事件等级和标识符：B#16#11 = 中断处于激活状态
OB10_STRT_INFO	BYTE	B#16#11：OB10 的启动请求 (B#16#12：OB11 的启动请求) ： ： (B#16#18：OB17 的启动请求)
OB10_PRIORITY	BYTE	分配的优先级；默认值为 2
OB10_OB_NUMBR	BYTE	OB 编号(10 至 17)
OB10_RESERVED_1	BYTE	保留
OB10_RESERVED_2	BYTE	保留
OB10_PERIOD_EXE	WORD	OB 以指定的时间间隔执行： W#16#0000：单次 W#16#0201：每分钟一次 W#16#0401：每小时一次 W#16#1001：每天一次 W#16#1201：每周一次 W#16#1401：每月一次 W#16#1801：每年一次 W#16#2001：月末
OB10_RESERVED_3	INT	保留
OB10_RESERVED_4	INT	保留
OB10_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

## 1.4 延时中断组织块(OB20 至 OB23)

### 描述

S7 提供多达四个在指定延迟后执行的 OB (OB20 至 OB23)。每个延时 OB 均可通过调用 SFC32 (SRT\_DINT)来启动。延迟时间是 SFC 的一个输入参数。

当程序调用 SFC32 (SRT\_DINT)时，需要提供 OB 编号、延迟时间和用户专用的标识符。经过指定的延迟后，OB 将会启动。还可取消尚未启动的延时中断。

### 了解延时中断 OB 的操作

延迟时间(同 OB 编号一起传送给 SFC32 的值，单位为 ms)到期后，操作系统将启动相应的 OB。

要使用延时中断，必须执行以下任务：

- 必须调用 SFC32 (SRT\_DINT)。
- 必须将延时中断 OB 作为用户程序的一部分下载到 CPU。

只有当 CPU 处于 RUN 模式下时才会执行延时 OB。暖重启或冷重启将清除延时 OB 的所有启动事件。如果延时中断还未启动，则可使用 SFC33 (CAN\_DINT)取消执行。

延迟时间的分辨率为 1ms。已到期的延迟时间可立即再次启动。可使用 SFC34 (QRY\_DINT)查询延时中断的状态。

如果发生以下事件之一，操作系统将调用异步错误 OB：

- 如果操作系统试图启动一个尚未装载的 OB，并且您在调用 SFC32 "SRT\_DINT"时指定了其编号。
- 如果在完全执行延时 OB 之前发生延时中断的下一个启动事件。

可使用 SFC 39 至 42 来禁用或延迟并重新使能延迟中断。

### 延时中断 OB 的本地数据

下表描述了延时中断 OB 的临时(TEMP)变量。变量名为 OB20 的默认名称。

变量	类型	描述
OB20_EV_CLASS	BYTE	事件等级和标识符： B#16#11：中断处于激活状态
OB20_STRT_INF	BYTE	B#16#21：OB20 的启动请求 (B#16#22：OB21 的启动请求) (B#16#23：OB22 的启动请求) (B#16#24：OB23 的启动请求)
OB20_PRIORITY	BYTE	分配的优先级：默认值 3 (OB20)至 6 (OB23)
OB20_OB_NUMBR	BYTE	OB 编号(20 至 23)
OB20_RESERVED_1	BYTE	保留
OB20_RESERVED_2	BYTE	保留
OB20_SIGN	WORD	用户 ID：来自调用 SFC32 (SRT_DINT)的输入参数 SIGN
OB20_DTIME	TIME	已组态的延迟时间(单位为 ms)
OB20_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

## 1.5 循环中断组织块(OB30 至 OB38)

### 描述

S7 提供了多达九个的循环中断 OB (OB30 至 OB38)，它们以固定的时间间隔来中断用户程序。下表给出了循环中断 OB 的默认时间间隔和优先级。

OB 编号	默认时间间隔	默认优先级
OB30	5 秒	7
OB31	2 秒	8
OB32	1 秒	9
OB33	500 毫秒	10
OB34	200 毫秒	11
OB35	100 毫秒	12
OB36	50 毫秒	13
OB37	20 毫秒	14
OB38	10 毫秒	15

### 了解循环中断 OB 的操作

循环中断 OB 的等距启动时间是由时间间隔和相位偏移量决定的。请参阅/234/，以了解 OB 的启动时间、周期以及相位偏移量之间的关系。

#### 注意

必须确保每个循环中断 OB 的运行时间远远小于其时间间隔。如果因时间间隔已到期，如果在预期的再次执行前未完全执行循环中断 OB，则启动时间错误 OB (OB80)。稍后将执行导致错误的循环中断。

可使用 SFC 39 至 42 来禁用或延迟，并重新启用循环中断

请参阅用户特定 CPU 的说明书，了解参数时间间隔、优先级和相位偏移量的范围。可使用 STEP 7 来更改参数设置。

## 循环中断 OB 的本地数据

下表给出了循环中断 OB 的临时(TEMP)变量。

变量	类型	描述
OB3x_EV_CLASS	BYTE	事件等级和标识符 B#16#11：中断处于激活状态
OB3x_STRT_INF	BYTE	<ul style="list-style-type: none"> <li>B#16#30：H 系统中循环中断组织块的特殊启动请求(选择了特殊处理，用于切换到"冗余"系统状态)</li> <li>B#16#31：OB30 的启动请求</li> <li>：</li> <li>B#16#36：OB35 的启动请求</li> <li>：</li> <li>B#16#39：OB38 的启动请求</li> <li>B#16#3A：循环中断 OB 的启动请求(OB30 至 OB38)，对于循环中断时钟频率低于 1 毫秒</li> </ul>
OB3x_PRIORITY	BYTE	分配的优先级：默认值为 7 (OB30)至 15 (OB38)
OB3x_OB_NUMBR	BYTE	OB 编号(30 至 38)
OB3x_RESERVED_1	BYTE	保留
OB3x_RESERVED_2	BYTE	保留
OB3x_PHS_OFFSET	INT	用于 OB 30 到 34 以及 36 到 38： <ul style="list-style-type: none"> <li>如果 OB3x_STRT_INF=B#16#3A：以<math>\mu</math>s 为单位表示的相位偏移</li> </ul> 在其它情况下：相位偏移以毫秒为单位
OB3x_PHASE_OFFSET	WORD	对于 OB35 <ul style="list-style-type: none"> <li>如果 OB35_STRT_INF=B#16#3A：以<math>\mu</math>s 为单位表示的相位偏移</li> <li>在其它情况下：相位偏移以毫秒为单位</li> </ul>
OB3x_RESERVED_3	INT	保留
OB3x_EXC_FREQ	INT	<ul style="list-style-type: none"> <li>如果 OB35_STRT_INF=B#16#3A：以<math>\mu</math>s 为单位表示的循环时间</li> <li>在其它情况下：以毫秒为单位表示的时钟时间</li> </ul>
OB3x_DATE_TIME	DATE_AND_TIME	调用 OB 时的日期和时间

## 1.6 硬件中断组织块(OB40 至 OB47)

### 描述

S7 提供了多达八个的独立硬件中断，每一中断都具有自己的 OB。

通过 STEP 7 分配参数，可以为将要触发硬件中断的每个信号模块指定以下内容：

- 在何种条件下触发硬件中断的通道。
- 要分配给个别通道组的硬件中断 OB(默认情况下，将由 OB40 来处理所有的硬件中断)。

借助于 CP 和 FM，可使用其各带软件来分配这些参数。

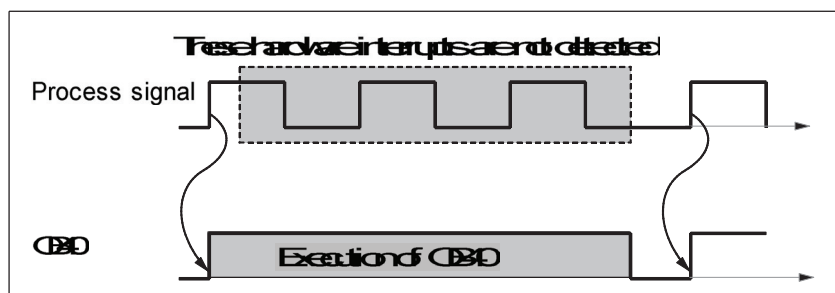
可使用 STEP 7 选择各个硬件中断 OB 的优先级。

### 了解硬件中断 OB 的操作

在模块触发硬件中断后，操作系统将标识插槽以及相应的硬件中断 OB。如果此 OB 的优先级高于当前激活的优先级，则将启动该 OB。当执行完此硬件中断 OB 后，将发送通道指定的确认。

如果在对硬件中断进行标识和确认的这段时间内，在同一模块中发生了触发硬件中断的另一事件，则应用以下规则：

- 如果该事件发生在先前触发硬件中断的通道中，则新中断丢失。在下图中，将根据数字输入模块的通道实例对此进行说明。触发事件是上升沿。硬件中断 OB 是 OB40。



- 如果该事件发生在同一模块的另一通道中，通常不会触发任何硬件中断。然而此中断不会丢失，而是在确认当前激活的硬件中断后被触发。

如果因来自另一模块中的硬件中断而使某一硬件中断被触发，并且其 OB 当前处于激活状态，则将记录新请求并且在 OB 空闲时对其进行处理。

可使用 SFC 39 至 42 来禁用或延迟，并重新启用硬件中断。

既可以使用 STEP 7，也可以使用 SFC 55 至 57 为模块的硬件中断分配参数。



## 硬件中断 OB 的本地数据

下表给出了硬件中断 OB 的临时(TEMP)变量。变量名为 OB40 的默认名称。

变量	类型	描述
OB40_EV_CLASS	BYTE	事件等级和标识符： B#16#11：中断处于激活状态
OB40_STRT_INF	BYTE	<ul style="list-style-type: none"> <li>B#16#41：通过中断线 1 中断</li> <li>B#16#42：通过中断线 2 中断(仅限 S7-400)</li> <li>B#16#43：通过中断线 3 中断(仅限 S7-400)</li> <li>B#16#44：通过中断线 4 中断(仅限 S7-400)</li> <li>B#16#45：WinAC：通过 PC 触发中断</li> </ul> 注意：将中断行 1, ... 4 分配给多值计算模式中的 CPU 1, ... 4。
OB40_PRIORITY	BYTE	分配的优先级：默认值为 16 (OB40)至 23 (OB47)
OB40_OB_NUMBR	BYTE	OB 编号(40 至 47)
OB40_RESERVED_1	BYTE	保留
OB40_IO_FLAG	BYTE	输入模块：B#16#54 输出模块：B#16#55
OB40_MDL_ADDR	WORD	触发中断的模块的逻辑基址
OB40_POINT_ADDR	DWORD	<ul style="list-style-type: none"> <li>对于数字模块： 触发硬件中断的模块上带输入的位域 OB40_POINT_ADDR 的哪一位分配给模块的哪个通道可在相关模块的说明中找到。</li> <li>对于模拟模块： 位域中包含的信息指示哪个通道超过了哪种限制(有关结构的精确描述，请参考相关模块的说明)。</li> <li>对于 CP 或 IM： 模块的中断状态(与用户无关)</li> </ul>
OB40_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

### 注意

如果使用呢具有 DPV1 功能的 CPU，可借助于 SFB54 "RALRM"获取有关中断的其它信息，这些信息不止包含 OB 的起始信息。在 S7 兼容模式下操作 DP 主站时，这一点也同样适用。

## 1.7 状态中断 OB (OB55)

### 注意

状态中断 OB (OB55)仅可用于具有 DPV1 和 PROFINET 功能的 CPU。

### 描述

如果状态中断是通过 DPV1 从站或 IO 设备的插槽触发，则 CPU 操作系统调用 OB55。如果 DPV1 从站或 IO 设备的组件(模块或机架)更改了其工作模式(例如由 RUN 改为 STOP)，则可能会发生这种情况。有关可触发状态中断事件的详细信息，请参阅 DPV1 从站或 IO 设备制造商的文档。

### 状态中断 OB 的本地数据

下表给出了状态中断 OB 的临时(TEMP)变量。所选变量的名称为 OB55 的默认名称。

变量	数据类型	描述
OB55_EV_CLASS	BYTE	事件等级和标识符： B#16#11 (即将发生的事件)
OB55_STRT_INF	BYTE	<ul style="list-style-type: none"> <li>B#16#55：DP 的状态中断</li> <li>B#16#58：PROFINET IO 的状态中断</li> </ul>
OB55_PRIORITY	BYTE	已组态的优先级，默认值为 2
OB55_OB_NUMBR	BYTE	OB 编号(55)
OB55_RESERVED_1	BYTE	保留
OB55_IO_FLAG	BYTE	输入模块：B#16#54 输出模块：B#16#55
OB55_MDL_ADDR	WORD	中断触发组件(模块)的逻辑基址
OB55_LEN	BYTE	中断所提供的数据块长度
OB55_TYPE	BYTE	中断类型"状态中断"的 ID
OB55_SLOT	BYTE	中断触发组件(模块)的插槽号
OB55_SPEC	BYTE	说明符 <ul style="list-style-type: none"> <li>0 到 1 位：中断说明符</li> <li>位 2：Add_Ack</li> <li>位 3 到 7：顺序号</li> </ul>
OB55_DATE_TIME	DATE_AND_TIME	调用 OB 时的日期和时间

### 注意

上表中所示的 OB55\_LEN、OB55\_TYPE、OB55\_SLOT、OB55\_SPEC 的含义仅适用于 DP 中的状态中断。如果有 PROFINET IO 中的状态中断，将需要根据下表所示组织本地变量。

**注意**

可通过在 OB55 中调用 SFB54 "RALRM"获取来自帧的中断的完整附加信息。

如果要对依赖于启动事件的 OB55 进行编程，建议按如下方式组织局部变量：

变量	数据类型
OB55_EV_CLASS	BYTE
OB55_STRT_INF	BYTE
OB55_PRIORITY	BYTE
OB55_OB_NUMBR	BYTE
OB55_RESERVED_1	BYTE_
OB55_IO_FLAG	BYTE
OB55_MDL_ADR	WORD
OB55_Z2	WORD
OB55_Z3	WORD
OB55_DATE_TIME	DATE_AND_TIME

根据启动事件，变量 OB55\_Z2 和 OB55\_Z3 包含不同的信息。详细说明如下。

**OB55\_Z2 的含义**

OB55_STRT_INF	OB55_Z2 的含义
B#16#55	<ul style="list-style-type: none"> <li>低字节：中断类型"状态中断"的 ID</li> <li>高字节：由中断提供的数据域的长度</li> </ul>
B#16#58	中断类型的 ID： <ul style="list-style-type: none"> <li>W#16#0005：状态中断</li> </ul>

OB55\_Z3 的含义

OB55_STRT_INF	OB55_Z3 的含义
B#16#55	<ul style="list-style-type: none"> <li>• 低字节：说明符                             <ul style="list-style-type: none"> <li>- 0 到 1 位：中断说明符</li> <li>- 位 2：Add_Ack</li> <li>- 位 3 到 7：序号</li> </ul> </li> <li>• 高字节：触发中断的组件插槽(模块或子模块)</li> </ul>
B#16#58	<p>中断说明符：</p> <ul style="list-style-type: none"> <li>• 位 0 到 10：序列号(值范围 0 至 2047)</li> <li>• 位 11：通道诊断                             <ul style="list-style-type: none"> <li>- 0: 无通道诊断信息</li> <li>- 1: 存在通道诊断信息</li> </ul> </li> <li>• 位 12：制造商特定诊断的状态：                             <ul style="list-style-type: none"> <li>- 0: 无可用的制造商特定状态信息</li> <li>- 1: 有可用的制造商特定状态信息</li> </ul> </li> <li>• 位 13：子模块的诊断状态：                             <ul style="list-style-type: none"> <li>- 0: 无可用的状态信息，已消除所有错误</li> <li>- 1: 至少通道诊断和/或状态信息的一项可用</li> </ul> </li> <li>• 位 14：保留</li> <li>• 位 15：应用程序相关诊断状态：                             <ul style="list-style-type: none"> <li>- 0: 此 AR 中没有已组态的模块报告诊断信息</li> <li>- 1: 此 AR 中至少有一个已组态的模块正报告诊断信息</li> </ul> </li> </ul>

## 1.8 更新中断 OB (OB56)

### 注意

更新中断 OB (OB56)仅可用于具有 DPV1 和 PROFINET 功能的 CPU。

### 描述

如果更新中断是通过 DPV1 从站或 IO 设备的插槽触发，则 CPU 操作系统调用 OB56。如果您更改了 DPV1 从站或 IO 设备插槽的参数(通过本地或远程访问)，则可能会发生这种情况。有关可触发更新中断的事件的详细信息，请参阅 DPV1 从站或 IO 设备制造商的文档。

### 更新中断 OB 的本地数据

下表给出了更新中断 OB 的临时(TEMP)变量。所选变量名为 OB56 的默认名称。

变量	数据类型	描述
OB56_EV_CLASS	BYTE	事件等级和标识符： B#16#11 (即将发生的事件)
OB56_STRT_INF	BYTE	<ul style="list-style-type: none"> <li>B#16#56：DP 的更新中断</li> <li>B#16#59：PROFINET IO 的更新中断</li> </ul>
OB56_PRIORITY	BYTE	已组态的优先级，默认值为 2
OB56_OB_NUMBR	BYTE	OB 编号(56)
OB56_RESERVED_1	BYTE	保留
OB56_IO_FLAG	BYTE	输入模块：B#16#54 输出模块：B#16#55
OB56_MDL_ADDR	WORD	中断触发组件(模块)的逻辑基址
OB56_LEN	BYTE	中断所提供的数据块长度
OB56_TYPE	BYTE	中断类型"更新中断"的 ID
OB56_SLOT	BYTE	中断触发组件(模块)的插槽号
OB56_SPEC	BYTE	说明符 <ul style="list-style-type: none"> <li>0 到 1 位：中断说明符</li> <li>位 2：Add_Ack</li> <li>位 3 到 7：顺序号</li> </ul>
OB56_DATE_TIME	DATE_AND_TIME	调用 OB 时的日期和时间

### 注意

上表中所示的 OB56\_LEN、OB56\_TYPE、OB56\_SLOT、OB56\_SPEC 的含义仅适用于 DP 中的更新中断。如果有 PROFINET IO 中的更新中断，将根据下表所示组织本地变量。

**注意**

可通过在 OB56 中调用 SFB54 "RALRM"获取来自帧的中断的完整附加信息。

如果要对依赖于启动事件的 OB56 进行编程，建议按如下方式组织局部变量：

变量	数据类型
OB56_EV_CLASS	BYTE
OB56_STRT_INF	BYTE
OB56_PRIORITY	BYTE
OB56_OB_NUMBR	BYTE
OB56_RESERVED_1	BYTE_
OB56_IO_FLAG	BYTE
OB56_MDL_ADR	WORD
OB56_Z2	WORD
OB56_Z3	WORD
OB56_DATE_TIME	DATE_AND_TIME

根据启动事件，变量 OB56\_Z2 和 OB56\_Z3 包含不同的信息。详细说明如下。

**OB56\_Z2 的含义**

OB56_STRT_INF	OB56_Z2 的含义
B#16#56	<ul style="list-style-type: none"> <li>低字节：中断类型"更新中断"的 ID</li> <li>高字节：由中断提供的数据域的长度</li> </ul>
B#16#59	中断类型的 ID： <ul style="list-style-type: none"> <li>W#16#0006：更新中断</li> </ul>

## OB56\_Z3 的含义

OB56_STRT_INF	OB56_Z3 的含义
B#16#56	<ul style="list-style-type: none"> <li>• 低字节：说明符               <ul style="list-style-type: none"> <li>- 0 到 1 位：中断说明符</li> <li>- 位 2：Add_Ack</li> <li>- 位 3 到 7：序号</li> </ul> </li> <li>• 高字节：触发中断的组件插槽(模块或子模块)</li> </ul>
B#16#59	中断说明符： <ul style="list-style-type: none"> <li>• 位 0 到 10：序列号(值范围 0 至 2047)</li> <li>• 位 11：通道诊断               <ul style="list-style-type: none"> <li>- 0: 无通道诊断信息</li> <li>- 1: 存在通道诊断信息</li> </ul> </li> <li>• 位 12：制造商特定诊断的状态：               <ul style="list-style-type: none"> <li>- 0: 无可用的制造商特定状态信息</li> <li>- 1: 有可用的制造商特定状态信息</li> </ul> </li> <li>• 位 13：子模块的诊断状态：               <ul style="list-style-type: none"> <li>- 0: 无可用的状态信息，已消除所有错误</li> <li>- 1: 至少通道诊断和/或状态信息的一项可用</li> </ul> </li> <li>• 位 14：保留</li> <li>• 位 15：应用程序相关诊断状态：               <ul style="list-style-type: none"> <li>- 0: 此 AR 中没有已组态的模块报告诊断信息</li> <li>- 1: 此 AR 中至少有一个已组态的模块正报告诊断信息</li> </ul> </li> </ul>

## 1.9 制造商特定中断 OB (OB57)

### 注意

制造商特定中断的 OB (OB57)仅对具有 DPV1 和 PROFINET 功能的 CPU 可用。

### 描述

如果制造商特定中断是通过 DPV1 从站或 IO 设备的插槽触发，则 CPU 操作系统调用 OB57。

### 制造商特定中断的 OB 的本地数据

下表给出制造商特定中断的 OB 的临时(TEMP)变量。所选变量名为 OB57 的默认名称。

变量	数据类型	描述
OB57_EV_CLASS	BYTE	事件等级和标识符： B#16#11 (即将发生的事件)
OB57_STRT_INF	BYTE	<ul style="list-style-type: none"> <li>B#16#57：DP 的制造商中断</li> <li>B#16#5A：PROFINET IO 的生产商中断</li> <li>B#16#5B：IO：配置文件特定的中断</li> </ul>
OB57_PRIORITY	BYTE	已组态的优先级，默认值为 2
OB57_OB_NUMBR	BYTE	OB 编号(57)
OB57_RESERVED_1	BYTE	保留
OB57_IO_FLAG	BYTE	输入模块：B#16#54 输出模块：B#16#55
OB57_MDL_ADDR	WORD	中断触发组件(模块)的逻辑基址
OB57_LEN	BYTE	中断所提供的数据块长度
OB57_TYPE	BYTE	中断类型"制造商特定中断"的 ID
OB57_SLOT	BYTE	中断触发组件(模块)的插槽号
OB57_SPEC	BYTE	说明符 <ul style="list-style-type: none"> <li>0 到 1 位：中断说明符</li> <li>位 2：Add_Ack</li> <li>位 3 到 7：顺序号</li> </ul>
OB57_DATE_TIME	DATE_AND_TIME	调用 OB 时的日期和时间

### 注意

上表中所示的 OB57\_LEN、OB57\_TYPE、OB57\_SLOT、OB57\_SPEC 的含义仅适用于 DP 中生产商相关的中断。如果有 PROFINET IO 中的生产商相关的中断，将根据下表所示组织本地变量。



**注意**

可通过在 OB57 中调用 SFB54 "RALRM" 获取来自帧的中断的完整附加信息。

如果要对依赖于启动事件的 OB57 进行编程，建议按如下方式组织局部变量：

变量	数据类型
OB57_EV_CLASS	BYTE
OB57_STRT_INF	BYTE
OB57_PRIORITY	BYTE
OB57_OB_NUMBR	BYTE
OB57_RESERVED_1	BYTE_
OB57_IO_FLAG	BYTE
OB57_MDL_ADR	WORD
OB57_Z2	WORD
OB57_Z3	WORD
OB57_DATE_TIME	DATE_AND_TIME

根据启动事件，变量 OB57\_Z2 和 OB57\_Z3 包含不同的信息。详细说明如下。

**OB57\_Z2 的含义**

OB57_STRT_INF	OB57_Z2 的含义
B#16#57	<ul style="list-style-type: none"> <li>低字节：中断类型"制造商特定中断"的 ID</li> <li>高字节：由中断提供的数据域的长度</li> </ul>
B#16#5A	中断类型的 ID： <ul style="list-style-type: none"> <li>W#16#0020 至 007F：制造商特定中断</li> </ul>

OB57\_Z3 的含义

OB57_STRT_INF	OB57_Z3 的含义
B#16#57	<ul style="list-style-type: none"> <li>• 低字节：说明符                             <ul style="list-style-type: none"> <li>- 0 到 1 位：中断说明符</li> <li>- 位 2：Add_Ack</li> <li>- 位 3 到 7：序号</li> </ul> </li> <li>• 高字节：触发中断的组件插槽(模块或子模块)</li> </ul>
B#16#5A	<p>中断说明符：</p> <ul style="list-style-type: none"> <li>• 位 0 到 10：序列号(值范围 0 至 2047)</li> <li>• 位 11：通道诊断                             <ul style="list-style-type: none"> <li>- 0: 无通道诊断信息</li> <li>- 1: 存在通道诊断信息</li> </ul> </li> <li>• 位 12：制造商特定诊断的状态：                             <ul style="list-style-type: none"> <li>- 0: 无可用的制造商特定状态信息</li> <li>- 1: 有可用的制造商特定状态信息</li> </ul> </li> <li>• 位 13：子模块的诊断状态：                             <ul style="list-style-type: none"> <li>- 0: 无可用的状态信息，已消除所有错误</li> <li>- 1: 至少通道诊断和/或状态信息的一项可用</li> </ul> </li> <li>• 位 14：保留</li> <li>• 位 15：应用程序相关诊断状态：                             <ul style="list-style-type: none"> <li>- 0: 此 AR 中没有已组态的模块报告诊断信息</li> <li>- 1: 此 AR 中至少有一个已组态的模块正报告诊断信息</li> </ul> </li> </ul>

## 1.10 多值计算中断组织块(OB60)

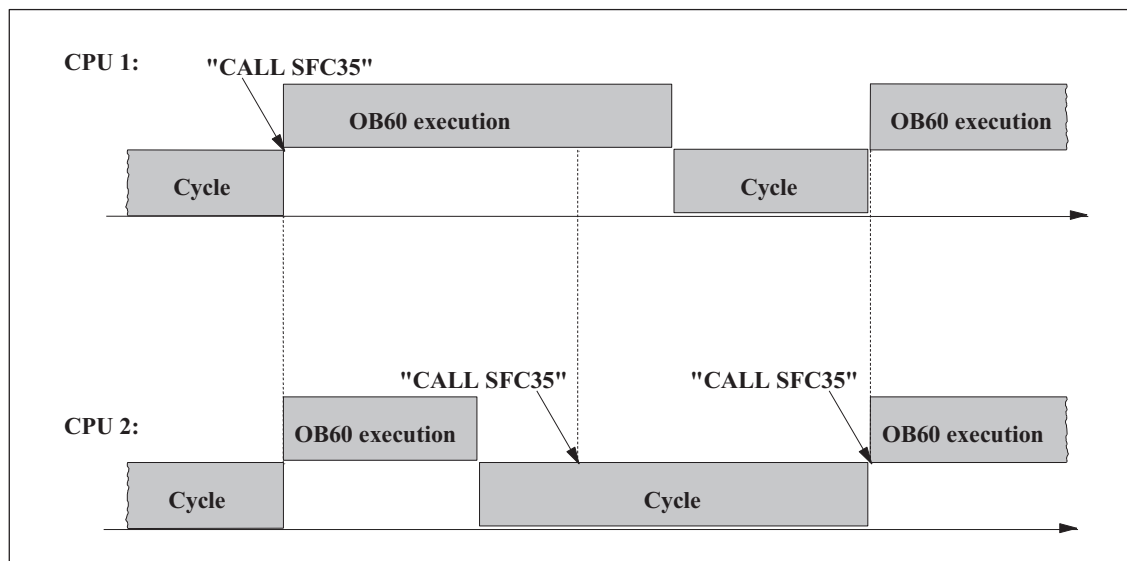
### 描述

使用多值计算中断，可确保 CPU 的反应在多值计算过程中与事件同步。与信号模块所触发的硬件中断比较而言，多值计算中断只能由 CPU 输出。

### 了解多值计算中断 OB 的操作

通过调用 SFC35 "MP\_ALM"触发多值计算中断。在多值计算期间，导致总线段上所有 CPU 上的同步的 OB60 启动，除非禁用(通过 SFC39 "DIS\_IRT")或延迟(通过 SFC41 "DIS\_AIRT")了 OB60。如果尚未将 OB60 装载到 CPU 上，则该 CPU 会在中断前返回上一优先级，并继续在此执行程序。在单处理器操作中以及在使用分段机架时，仅在您调用 SFC35 "MP\_ALM"的 CPU 上启动 OB60。

当用户程序调用 SFC35 "MP\_ALM"时，应为所有的 CPU 提供一个作业 ID。这就允许您对特定事件作出反应。如果在不同的 CPU 上以不同方式对 OB60 进行编程，将导致 OB 的执行时间不同。在这种情况下，CPU 将在不同时间返回到被中断的优先级。如果下一个多值计算中断由 CPU 输出，而另一个 CPU 仍在忙于执行上一个多值计算中断的 OB60，则 OB60 既不会在请求时启动，也不会属于该总线段的任何其它 CPU 上启动。下图以两个 CPU 为例对此进行了说明。所调用 SFC35 的函数值将通知您具体结果。



## 多值计算中断 OB 的本地数据

下表说明了多值计算中断 OB 的临时(TEMP)变量。变量名为 OB60 的默认名称。

变量	数据类型	描述
OB60_EV_CLASS	BYTE	事件等级和 ID : B#16#11 : 中断处于激活状态
OB60_STRT_INF	BYTE	B#16#61 : 由自身的 CPU 触发的多值计算中断 B#16#62 : 由其它 CPU 触发的多值计算中断
OB60_PRIORITY	BYTE	分配的优先级 : 默认值为 25
OB60_OB_NUMBR	BYTE	OB 编号 : 60
OB60_RESERVED_1	BYTE	保留
OB60_RESERVED_2	BYTE	保留
OB60_JOB	INT	作业标识符 : SFC35 "MP_ALM"的输入变量 JOB
OB60_RESERVED_3	INT	保留
OB60_RESERVED_4	INT	保留
OB60_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME。

## 1.11 同步周期性中断 OB (OB61 至 OB64)

### 描述

同步周期性中断让您可以选择以带 DP 周期 OR PN 发送时钟的同步周期来启动程序。OB61 作为一个接口 OB ,连接到同步周期性中断 TSAL1。可将 OB61 的优先级设为 0 (取消选择 OB)及 2 至 26 之间。

- 当心：在使用 L 或 T 命令(例如 L PIB、T PQB)直接访问，以及使用 SFC 14 "DPRD\_DAT"和 15 "DPWR\_DAT"时,请避免访问已为其过程映像分区分配了到 OB6x (同步周期性中断)的连接的 I/O 区域

### 同步周期性中断 OB 的本地数据

下表说明了同步周期性中断 OB 的临时(TEMP)变量。变量名为 OB61 的默认名称。

变量	数据类型	描述
OB61_EV_CLASS	BYTE	事件等级和 ID : B#16#11 : 中断处于激活状态
OB61_STRT_INF	BYTE	B#16#64 : OB61 的启动请求 B#16#67 : OB64 的启动请求
OB61_PRIORITY	BYTE	分配的优先级 ; 默认 : 25
OB61_OB_NUMBR	BYTE	OB 编号 : 61 ... 64
OB61_RESERVED_1	BYTE	保留
OB61_RESERVED_2	BYTE	保留
OB61_GC_VIOL	BOOL	PROFIBUS DP 中的 GC 违规
OB61_FIRST	BOOL	启动或停止状态后的首次使用
OB61_MISSED_EXEC	BYTE	自上次执行 OB61 以来启动 OB61 失败的次数
OB61_DP_ID	BYTE	处于同步模式的 DP 主站系统的 DP 主站系统 ID (1 至 32)或处于同步模式 PNIO 系统的 PROFINET IO 系统 ID(100 至 115)
OB61_RESERVED_3	BYTE	保留
OB61_RESERVED_4	WORD	保留
OB61_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME。

## 1.12 技术同步中断 OB (OB65)

---

### 注意

技术同步中断 OB (OB65)仅适用于技术 CPU。

---

### 描述

通过技术同步中断可选择在更新技术块的同时启动程序。技术同步中断 OB 在更新技术块后启动。技术同步中断 OB 的优先级固定设置为 25，不能对其进行更改。

---

### 注意

当启动技术同步中断 OB 时，仍然没有技术功能的最新背景数据可用。

---

### 技术同步中断 OB 的本地数据

下表说明了技术同步中断 OB65 的临时(TEMP)变量。变量名为 OB65 的默认名称。

变量	数据类型	描述
OB65_EV_CLASS	BYTE	事件等级和 ID : B#16#11 : 进入事件，事件等级 1
OB65_STRT_INF	BYTE	B#16#6A : OB65 的启动请求
OB65_PRIORITY	BYTE	优先级等级 25 (固定设置)
OB65_OB_NUMBR	BYTE	OB 编号(65)
OB65_RESERVED_1	BYTE	保留
OB65_RESERVED_2	BYTE	保留
OB65_RESERVED_3	BOOL	保留
OB65_FIRST	BOOL	启动后首次使用 OB65
OB65_MISSED_EXEC	BYTE	自上次执行 OB65 以来启动 OB65 失败的次数
OB65_RESERVED_4	BYTE	保留
OB65_RESERVED_5	BYTE	保留
OB65_RESERVED_6	WORD	保留
OB65_DATE_TIME	DATE_AND_TIME	调用 OB 时的日期和时间。

## 1.13 I/O 冗余错误 OB (OB70)

### 注意

I/O 冗余错误 OB (OB70)仅可与 H CPU 结合使用。

### 描述

当在 PROFINET IO 系统/DP 主站系统的至少一个以上的 IO 设备/DP 从站上发生丢失冗余或冗余返回时，H CPU 的操作系统调用 OB 70

OB 70 的启动事件实例：

- IO 设备/DP 从站的接口出错
- 导致在一个以上 DP 从站上发生冗余丢失的活动 DP 主站上的总线故障。  
注意：如果 DP 主站系统发生故障或返回，且所有 DP 从站的冗余状态在过程期间保持相同，则调用 OB 86。

如果发生了启动事件而 OB70 没有编程，则 CPU 不会转为 STOP 模式。如果装载了 OB70 且 H 系统处于冗余模式，则在两个 CPU 上执行 OB70。H 系统仍处于冗余模式。

### I/O 冗余 OB 的本地数据

下表给出了 I/O 冗余错误 OB 的临时(TEMP)变量。所选变量名为 OB70 的默认名称。

变量	类型	描述
OB70_EV_CLASS	BYTE	事件等级和 ID： <ul style="list-style-type: none"> <li>• B#16#72：离开事件</li> <li>• B#16#73：进入事件</li> </ul>
OB70_FLT_ID	BYTE	错误代码(可能值： B#16#A2、B#16#A3、B#16#A5)
OB70_PRIORITY	BYTE	优先级; 可通过 STEP 7 分配(硬件配置)
OB70_OB_NUMBR	BYTE	OB 编号(70)
OB70_RESERVED_1	WORD	保留
OB70_INFO_1	WORD	取决于错误代码
OB70_INFO_2	WORD	取决于错误代码
OB70_INFO_3	WORD	取决于错误代码
OB70_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

下表给出了启动 OB70 的事件。

OB70_EV_CLASS	OB70_FLT_ID	OB70 的启动事件
B#16#73	B#16#A2	DP 主站/DP 主站系统故障，至少一个 DP 从站发生冗余丢失；或者 IO 控制器/PROFINET IO 系统故障，至少一台 IO 设备发生冗余丢失
B#16#73/B#16#72	B#16#A3	冗余丢失/返回 DP 从站的冗余
B#16#73/B#16#72	B#16#A5	IO 设备的冗余丢失/冗余返回

取决于错误代码的变量具有如下含义：

错误代码：	位	含义
<ul style="list-style-type: none"> <li>• B#16#A2                             <ul style="list-style-type: none"> <li>- OB70_INFO_1 :</li> <li>- OB70_INFO_2 :</li> <li>- OB70_INFO_3 :</li> </ul> </li> <li>• B#16#A2                             <ul style="list-style-type: none"> <li>- OB70_INFO_1 :</li> <li>- OB70_INFO_2 :</li> <li>- OB70_INFO_3 :</li> </ul> </li> </ul>	0 到 7 : 8 到 15	(如果 DP 主站系统故障) 受影响 DP 主站的逻辑基址 保留 保留 受影响 DP 主站的 DP 主站系统 ID
<ul style="list-style-type: none"> <li>• B#16#A3                             <ul style="list-style-type: none"> <li>- OB70_INFO_1 :</li> <li>- OB70_INFO_2 :</li> <li>- OB70_INFO_3 :</li> </ul> </li> </ul>	0 到 10 : 11 到 14 : 15:  0 到 7 : 8 到 15 :	(如果 PROFINET IO 系统故障) 受影响的 IO 控制器的逻辑基址 保留 保留 PROFINET IO 系统 ID (偏移量为十进制 100) 1  DP 主站的逻辑基址 受影响的 DP 从站 : 逻辑基址(使用 S7 从站时)或诊断地址(使用 DP 标准从站时)。 I/O 标识 受影响的 DP 从站 : DP 站编号 DP 主站系统 ID
B#16#A5 OB70_INFO_1 : OB70_INFO_2 :  OB70_INFO_3	0 到 14 : 15:  0 到 10 : 11 到 14 : 15:	IO 控制器的逻辑基址 受影响的 IO 设备 逻辑基址 I/O 标识符 受影响的 IO 设备 站号 PROFINET IO 系统 ID (偏移量为十进制 100)

**注意**

如果使用呢具有 DPV1 功能的 CPU，可借助于 SFB54 "RALRM"获取有关中断的其它信息，这些信息不止包含 OB 的起始信息。在 S7 兼容模式下操作 DP 主站时，这一点也同样适用。



## 1.14 CPU 冗余错误 OB (OB72)

### 注意

只在 H CPU 中存在 CPU 冗余错误 OB (OB72)。

### 描述

如果发生下列事件之一，则 H CPU 的操作系统将调用 OB72：

- CPU 冗余丢失
- 保留-主站切换
- 同步错误
- SYNC 模块中的错误
- 更新中止
- 比较错误(例如，RAM、PIQ)

在适当的启动事件后，OB72 由处于 RUN 或 STARTUP 模式下的所有 CPU 执行。

### CPU 冗余错误 OB 的本地数据

下表给出了 CPU 冗余错误 OB 的临时(TEMP)变量。OB72 的默认名称已被用作变量名。

变量	类型	描述
OB72_EV_CLASS	BYTE	事件等级和 ID： B#16#78：离开事件 B#16#73、B#16#75、B#16#79：进入事件
OB72_FLT_ID	BYTE	错误代码(可能值：B#16#01、B#16#02、 B#16#03、B#16#20、B#16#21、B#16#22、 B#16#23、B#16#31、B#16#33、B#16#34、 B#16#35、B#16#40、B#16#41、B#16#42、 B#16#43、B#16#44、B#16#50、B#16#51、 B#16#52、B#16#53、B#16#54、B#16#55、 B#16#56、B#16#C1、B#16#C2)
OB72_PRIORITY	BYTE	优先级；可通过 STEP 7 分配(硬件配置)
OB72_OB_NUMBR	BYTE	OB 编号(72)

变量	类型	描述
OB72_RESERVED_1	WORD	<p>仅适用于错误代码 B#16#03 :</p> <ul style="list-style-type: none"> <li>• 高字节 : OB72_INFO_2 和 OB72_INFO_3 的内容 ID                             <ul style="list-style-type: none"> <li>- 0: OB72_INFO-2 和 OB72_INFO_3 无任何意义</li> <li>- B#16#C4 : 在故障检测模式执行后, 通过待机-主站切换(对于 OB72_INFO_3=W#16#0001)或不通过待机-主站切换(对于 OB72_INFO_3=W#16#0002)跳转到冗余模式。OB72_INFO_2 被保留。</li> <li>- B#16#CD : OB72_INFO_2 和 OB72_INFO_3 中包含优先级的实际锁定时间 &gt; 15</li> </ul> </li> <li>• 低字节 : 保留</li> </ul>
OB72_INFO_1	WORD	<p>仅适用于错误代码 B#16#C2 :</p> <ul style="list-style-type: none"> <li>• 高字节 : 超出监视时间的标识符 :                             <ul style="list-style-type: none"> <li>- 1: 周期延长</li> <li>- 2: I/O 死区时间</li> <li>- 4: 通讯延迟</li> </ul> </li> <li>• 低字节 : 当前的更新尝试</li> </ul>
OB72_INFO_2	WORD	<p>仅适用于错误代码 B#16#03 和 OB72_RESERVED_1=B#16#CD : 优先级的实际锁定时间 &gt; 15ms 的高位字</p>
OB72_INFO_3	WORD	<p>仅适用于错误代码 B#16#03 :</p> <ul style="list-style-type: none"> <li>• OB72_RESERVED_1=B#16#C4 :                             <ul style="list-style-type: none"> <li>- W#16#0001 : 在执行故障检测模式后, <b>通过</b>待机-主站切换跳转到冗余操作</li> <li>- W#16#0002 : 在执行故障检测模式后, <b>不通过</b>待机-主站切换跳转到冗余操作</li> </ul> </li> <li>• OB72_RESERVED_1=B#16#CD : 优先级的实际锁定时间 &gt; 15ms 的高位字</li> </ul>
OB82_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

下表给出了启动 OB72 的事件。

OB72_EV_CLASS	OB72_FLT_ID	OB72 的启动事件
B#16#73	B#16#01	因 CPU 故障导致冗余丢失(2 个中的 1 个)
B#16#73	B#16#02	由用户触发的保留设备上的 STOP 模式而导致冗余丢失(2 个中的 1 个)
B#16#73	B#16#03	H 系统(2 个中的 1 个)变为冗余模式
B#16#73	B#16#20	RAM 比较出错
B#16#73	B#16#21	比较过程映像输出值时出错
B#16#73	B#16#22	比较存储器位、定时器或计数器时出错
B#16#73	B#16#23	识别出不同操作系统数据
B#16#73	B#16#31	因主站故障而发生待机主站切换
B#16#73	B#16#33	因 RUN 模式中的组态而发生待机主站切换
B#16#73	B#16#34	因同步模块连接问题而导致待机-主站切换
B#16#73	B#16#35	由 SFC90 "H_CTRL"触发待机-主站切换
B#16#73	B#16#40	因等待时间已到期而在用户程序中发生同步错误
B#16#73	B#16#41	因在不同的同步点等待而在用户程序中发生同步错误
B#16#73	B#16#42	因在不同的同步点等待而在操作系统中发生同步错误
B#16#73	B#16#43	因等待时间已到期而在操作系统中发生同步错误
B#16#73	B#16#44	因数据错误而导致操作系统出现同步错误
B#16#79	B#16#50	无 SYNC 模块
B#16#79	B#16#51	在未上电的情况下, 修改 SYNC 模块
B#16#79/B#16#78	B#16#52	SYNC 模块已删除/已插入
B#16#79	B#16#53	在未复位的情况下, 修改 SYNC 模块
B#16#79	B#16#54	SYNC 模块: 机架号被分配了两次
B#16#79/B#16#78	B#16#55	SYNC 模块出错/已排除
B#16#79	B#16#56	在 SYNC 模块上设置了非法机架号
B#16#73	B#16#C1	更新中止
B#16#73	B#16#C2	因在第 n (1 ≤ n ≤ 在因超出监视时间而导致的中止后所进行的更新尝试的最大可能次数)次尝试期间超出了监视时间而导致更新尝试中止。

## 1.15 通信冗余错误 OB (OB73)

### 注意

通信冗余错误 OB (OB73)仅在 CPU 417-4H 的固件版本 V2.0.x 中可用。

### 描述

当容错 S7 连接中发生首次冗余丢失时，H CPU 的操作系统将调用 OB73(只有在 S7 通信中才会有容错 S7 连接。更多信息，请参见"S7-400 H 可编程控制器，容错系统。")。如果其它容错 S7 连接发生了冗余丢失，则不会再有 OB73 启动。

直到为具有容错功能的所有 S7 连接恢复冗余后，才会出现另一个 OB73 启动。

如果发生了启动事件且 OB73 没有编程，CPU 不会转为 STOP 模式。

### 通信冗余错误 OB 的本地数据

下表给出了通信冗余错误 OB 的临时(TEMP)变量。OB73 的默认名称已被用作变量名。

变量	类型	描述
OB73_EV_CLASS	BYTE	事件等级和 ID： B#16#73、B#16#72
OB73_FLT_ID	BYTE	错误代码(可能值：B#16#E0)
OB73_PRIORITY	BYTE	分配的优先级：默认值为 25
OB73_OB_NUMBR	BYTE	OB 编号(73)
OB73_RESERVED_1	WORD	保留
OB73_INFO_1	WORD	• (与用户无关)
OB73_INFO_2	WORD	(与用户无关)
OB73_INFO_3	WORD	(与用户无关)
OB73_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

下表给出了启动 OB73 的事件。

OB73_FLT_ID	OB73 的启动事件
B#16#E0	通讯中冗余丢失/问题已排除

## 1.16 时间错误组织块(OB80)

### 描述

无论何时执行 OB 时出错，S7-300 CPU 的操作系统将调用 OB80。此类错误包括：超出周期时间、执行 OB 时出现确认错误、提前了时间而使 OB 的启动时间被跳过、时间同步失败或返回、在 CiR 后恢复 RUN 模式。例如，如果在上一次调用之后发生了某一循环中断 OB 的启动事件，而同一 OB 此时仍在执行中，则操作系统将调用 OB80。

如果 OB80 尚未编程，则 CPU 将转为 STOP 模式。

可以使用 SFC 39 至 42 禁用或延迟和重新启用时间错误 OB。

---

### 注意

如果因超出了扫描时间而导致在同一扫描周期内调用了两次 OB80，则 CPU 转为 STOP 模式。通过在程序中的适当的点调用 SFC43 "RE\_TRIGR"，可防止这种情况。

---

### 关于使用"报告系统错误"的注意事项

"报告系统错误"延迟处理在其运行期间发生的中断。

如果满足条件"报告系统错误"的运行时间 > 周期时间的 50%，则该影响是混合的(此处，"报告系统错误"的运行时间指周期时间内"报告系统错误"的所有处理时间之和)。这可能导致调用时间错误 OB，并可能导致 CPU 进入 STOP 状态(如果没有加载时间错误 OB)。

以下措施可用于防止发生该情况：

- 如果使用定时中断 OB (OB61-OB64)，则必须确保每单位时间内这些 OB 的调用频率应比以前低。可通过增大每个定时中断 OB 的相应参数"应用周期系数"来实现该目的(在 HW Config 的 CPU 属性：应用周期的"定时周期中断"标签 > "IO 系统编号"下拉列表 > "详细资料"按钮 > "系数"下拉列表中)。
- 增大比值(周期时间 - "报告系统错误"的运行时间)/("报告系统错误"的运行时间)。为此，有两种方法可供选择：
  - 当在循环中断 OB (OB30-OB38)中调用"报告系统错误"时，增大时间周期。
  - 在 OB1 中，增大与"报告系统错误"无关的程序段的运行时间。例如，可使用 SFC47 "WAIT"来执行该功能。

**时间错误 OB 的本地数据**

下表说明了时间错误 OB 的临时(TEMP)变量。变量名为 OB80 的默认名称。

变量	类型	描述
OB80_EV_CLASS	BYTE	事件等级和标识符：B#16#35、B#16#38 (仅对于错误代码 B#16#D5)、B#16#39 (仅对于错误代码 B#16#D5)
OB80_FLT_ID	BYTE	错误代码：(可能值： B#16#01、B#16#02、B#16#05、B#16#06、B#16#07、 B#16#08、B#16#09、B#16#0A、B#16#0B、B#16#D5)
OB80_PRIORITY	BYTE	优先级：OB 80 以优先级 26 在 RUN 模式下运行，如果发生 OB 请求缓冲区溢出，则以优先级 28 运行
OB80_OB_NUMBR	BYTE	OB 编号(80)
OB80_RESERVED_1	BYTE	保留
OB80_RESERVED_2	BYTE	保留
OB80_ERROR_INFO	WORD	错误信息：取决于错误代码
OB80_ERR_EV_CLASS	BYTE	导致错误的启动事件的事件等级
OB80_ERR_EV_NUM	BYTE	导致错误的启动事件的事件编号
OB80_OB_PRIORITY	BYTE	错误信息：取决于错误代码
OB80_OB_NUM	BYTE	错误信息：取决于错误代码
OB80_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

取决于错误代码的变量具有如下：

错误代码：	位	含义
<ul style="list-style-type: none"> <li>• B#16#01                             <ul style="list-style-type: none"> <li>- OB80_ERROR_INFO：</li> <li>- OB80_ERR_EV_CLASS：</li> <li>- OB80_ERR_EV_NUM：</li> <li>- OB80_OB_PRIORITY：</li> <li>- OB80_OB_NUM</li> </ul> </li> </ul>		超过周期时间。 上一扫描周期的运行时间(ms)。 触发中断的事件的等级。 触发中断的事件的编号。 出错时执行的 OB 的优先级。 出错时执行的 OB 的编号。

错误代码：	位	含义
<ul style="list-style-type: none"> <li>• B#16#02</li> <li>- OB80_ERROR_INFO :</li>   <li>- OB80_ERR_EV_CLASS :</li> <li>- OB80_ERR_EV_NUM :</li> <li>- OB80_OB_PRIORITY</li>   <li>- OB80_OB_NUM :</li> </ul>		<p>仍在执行调用的 OB。</p> <p>被调用块的各个临时变量取决于下列各项</p> <ul style="list-style-type: none"> <li>• OB80_ERR_EV_CLASS 和</li> <li>• OB80_ERR_EV_NUM。</li> <li>• 触发中断的事件的等级。</li> <li>• 触发中断的事件的编号。</li> <li>• 导致错误的 OB 的优先级(例如：对于本应启动，但却未能启动的 OB30/优先级 7，其级别为"7")。</li> <li>• 导致错误的 OB 的编号(例如：对于本应启动，但却未能启动的 OB30，其编号为"30")。</li> </ul>
<ul style="list-style-type: none"> <li>• B#16#05 和</li> <li>• B#16#06</li> <li>- OB80_ERROR_INFO :</li>   <li>- OB80_ERR_EV_CLASS :</li> <li>- OB80_ERR_EV_NUM :</li> <li>- OB80_OB_PRIORITY :</li> <li>- OB80_OB_NUM :</li> </ul>	置位位 0 : 置位位 7 : 位 8 到 15 :	<p>因向前拨动时钟而导致时间中断已流逝。</p> <p>在 HOLD 模式后返回 RUN 模式时时间中断已流逝。</p> <p>时间中断 0 的启动时间为过去的时间。</p> <p>时间中断 7 的启动时间为过去的时间。</p> <p>未用</p> <p>未用</p> <p>未用</p> <p>未用</p> <p>未用</p>
<ul style="list-style-type: none"> <li>• B#16#07</li> </ul> <p>有关参数的含义，请参见错误代码 B#16#02。</p>		<p>当前优先级的 OB 请求缓冲区溢出</p> <p>(优先级的每个 OB 启动请求均会被输入到相应 OB 请求缓冲区中；在完成 OB 之后，将会删除该条目。如果优先级的 OB 启动请求多于相应 OB 请求缓冲区中所允许的最大条目数，则通过错误代码 B#16#07 调用 OB80)。</p>
<ul style="list-style-type: none"> <li>• B#16#08</li> </ul> <p>有关参数的含义，请参见错误代码 B#16#02。</p>		同步周期性中断时间错误
<ul style="list-style-type: none"> <li>• B#16#09</li> </ul> <p>有关参数的信息，请参考错误代码 B#16#02。</p>		因中断负载过高而导致中断丢失
B#16#0A <ul style="list-style-type: none"> <li>• OB80_ERROR_INFO :</li> </ul>		<p>在 CiR 之后恢复 RUN 模式</p> <p>CiR 同步时间(单位为 ms)</p>
B#16#0B <ul style="list-style-type: none"> <li>• OB80_ERR_EV_NUM :</li> <li>• OB80_OB_PRIORITY :</li> <li>• OB80_OB_NUM</li> </ul>		<p>技术同步中断</p> <p>触发中断的事件编号：W#16#116A</p> <p>出错时所处理 OB 的优先级。</p> <p>出错时所处理 OB 的编号：65</p>





## 1.17 电源错误组织块(OB81)

### 描述

只要发生由错误或故障所触发的事件，而此错误或故障又与电源(仅在 S7-400 上)或备用电池(当事件进入和离开时)有关，则 CPU 的操作系统调用 OB81。

在 S7-400 中，如果已使用 BATT.INDIC 开关激活了电池测试功能，则只有在出现电池故障时才会调用 OB81。

如果 OB81 没有编程，则 CPU 不会转为 STOP 模式。

可以使用 SFC 39 至 42 禁用或延迟，并重新启用电源错误 OB。

### 电源错误 OB 的本地数据

下表说明了电源错误 OB 的临时(TEMP)变量。变量名为 OB81 的默认名称。

变量	类型	描述
OB81_EV_CLASS	BYTE	事件等级和标识符： B#16#38：离开事件 B#16#39：进入事件
OB81_FLT_ID	BYTE	错误代码：(可能值) B#16#21、B#16#22、B#16#23、B#16#25、B#16#26、 B#16#27、B#16#31、B#16#32、B#16#33)
OB81_PRIORITY	BYTE	优先级；可通过 STEP 7 分配(硬件组态) 例如，RUN 模式的可能值为：2-26
OB81_OB_NUMBR	BYTE	OB 编号(81)
OB81_RESERVED_1	BYTE	保留
OB81_RESERVED_2	BYTE	保留
OB81_RACK_CPU	WORD	<ul style="list-style-type: none"> <li>• 位 0 到 7：B#16#00</li> <li>• 位 8 到 15： <ul style="list-style-type: none"> <li>- 对于标准 CPU：B#16#00</li> <li>- 对于 H-CPU：8 到 10 位：机架号；11 位：0 = 保留 CPU，1 = 主站 CPU；位 12 到 15：1111</li> </ul> </li> </ul>
OB81_RESERVED_3	BYTE	仅与错误代码 B#16#31、B#16#32 以及 B#16#33 相关
OB81_RESERVED_4	BYTE	仅与错误代码 B#16#31、B#16#32 以及 B#16#33 相关
OB81_RESERVED_5	BYTE	仅与错误代码 B#16#31、B#16#32 以及 B#16#33 相关
OB81_RESERVED_6	BYTE	仅与错误代码 B#16#31、B#16#32 以及 B#16#33 相关
OB81_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

变量 OB81\_RESERVED\_i,  $3 \leq i \leq 6$  指示其中的备用电池(错误代码 B#16#31)、备用电压(错误代码 B#16#32)或 24V 电源(错误代码 B#16#33)出现故障或返回的扩展机架。下表给出了为扩展机架分配变量 OB81\_RESERVED\_i,  $3 \leq i \leq 6$  中各个位的情况。

	OB81_RESERVED_6	OB81_RESERVED_5	OB81_RESERVED_4	OB81_RESERVED_3
第0位	保留	扩展机架 8	扩展机架 16	保留
第1位	扩展机架 1	扩展机架 9	扩展机架 17	保留
第2位	扩展机架 2	扩展机架 10	扩展机架 18	保留
第3位	扩展机架 3	扩展机架 11	扩展机架 19	保留
第4位	扩展机架 4	扩展机架 12	扩展机架 20	保留
第5位	扩展机架 5	扩展机架 13	扩展机架 21	保留
第6位	扩展机架 6	扩展机架 14	保留	保留
第7位	扩展机架 7	扩展机架 15	保留	保留

变量 OB81\_RESERVED\_i 中各个位的含义(对于相关的扩展机架)如下：

当事件发生时，将对其中至少有一个电池或备用电压或 24 伏电源出现故障的扩展机架进行标记(置位相应的位)。对于其中至少有一个电池或备用电压或 24 伏电源在先前发生过故障的扩展机架，将不再对其进行标记。

当排除了事件并且至少在一个扩展机架上恢复备份后，将就此发出信号(置位相应的位)。

下表给出了启动 OB81 的事件：

OB81_EV_CLASS	OB81_FLT_ID	含义
B#16#39/B#16#38	B#16#21 :	至少一个中央机架的备用电池电量耗尽/问题已排除(BATTF) 注意：只要两个电池中有一个电池出现故障，就会发生此事件(如果存在冗余的备用电池)。如果第二个电池恰巧也发生了故障，则该事件不会再次出现。
B#16#39/B#16#38	B#16#22 :	中央机架中的备用电压出现故障/问题已排除(BAF)
B#16#39/B#16#38	B#16#23 :	中央机架中的 24 伏电源出现故障/问题已排除。
B#16#39/B#16#38	B#16#25 :	至少在一个冗余中央机架中的至少有一个备用电池的电量耗尽/问题已排除(BATTF)
B#16#39/B#16#38	B#16#26 :	至少一个冗余中央机架中的备用电压出现故障/问题已排除(BAF)
B#16#39/B#16#38	B#16#27 :	至少一个冗余中央机架中的 24 伏电源出现故障
B#16#39/B#16#38	B#16#31 :	至少在一个扩展机架中的至少有一个备用电池的电量耗尽/问题已排除(BATTF)。
B#16#39/B#16#38	B#16#32 :	至少一个扩展机架中的备用电压出现故障/问题已排除(BAF)
B#16#39/B#16#38	B#16#33 :	至少一个扩展机架中的 24 伏电源出现故障/问题已排除。

## 1.18 诊断中断组织块(OB82)

### 描述

如果一个模块具有诊断能力，并且已为它启用了诊断中断，那么当它检测出其诊断状态发生变化时，它将发送一个诊断中断请求到 CPU。

- 存在一个问题或一个组件要求维护或两者皆有(事件输入状态)。
- 不存在问题且没有其它组件需要维护(事件退出状态)。

则操作系统调用 OB82。

OB82 的局部变量包含逻辑基址和四字节的故障模块的诊断数据(请参见下表)。

如果 OB82 尚未编程，则 CPU 转为 STOP 模式。

可以使用 SFC 39 至 42 禁用或延迟，并重新启用诊断中断 OB。

---

### 注意

对于 PROFINET IO 控制器，可在组态期间指定通信中断是否导致诊断中断 OB 调用。相关中断为 PROFINET 接口的诊断事件。

这些事件是预置的，因而它们不会导致 OB 82 调用。

---

## 诊断中断 OB 的本地数据

下表说明了诊断中断 OB 的临时(TEMP)变量。变量名为 OB82 的默认名称。

变量	类型	描述
OB82_EV_CLASS	BYTE	事件等级和标识符： <ul style="list-style-type: none"> <li>B#16#38：离开的事件(无模块错误- 清除了所有错误)</li> <li>B#16#39：到达的事件(模块现在有错误或仍有错误- 待处理错误数目发生变化)</li> </ul>
OB82_EV_CLASS	BYTE	事件等级和标识符： <ul style="list-style-type: none"> <li>B#16#38：离开事件</li> <li>B#16#39：进入事件</li> </ul>
OB82_FLT_ID	BYTE	错误代码(B#16#42)
OB82_PRIORITY	BYTE	• 优先级；可通过 STEP 7 分配(硬件组态)
OB82_OB_NUMBR	BYTE	OB 编号(82)
OB82_RESERVED_1	BYTE	保留
OB82_IO_FLAG	BYTE	<ul style="list-style-type: none"> <li>输入模块： B#16#54</li> <li>输出模块： B#16#55</li> </ul>
OB82_MDL_ADDR	WORD	发生故障的模块的逻辑基址
OB82_MDL_DEFECT	BOOL	模块发生故障
OB82_INT_FAULT	BOOL	内部故障
OB82_EXT_FAULT	BOOL	外部故障
OB82_PNT_INFO	BOOL	通道故障
OB82_EXT_VOLTAGE	BOOL	外部电压故障
OB82_FLD_CONNCTR	BOOL	未插入前面板连接器
OB82_NO_CONFIG	BOOL	未组态模块
OB82_CONFIG_ERR	BOOL	模块中的参数不正确
OB82_MDL_TYPE	BYTE	<ul style="list-style-type: none"> <li>0 到 3 位：模块等级</li> <li>位 4：通道信息存在</li> <li>位 5：用户信息存在</li> <li>位 6：诊断中断来自替换件</li> <li>位 7：需要维护</li> </ul>
OB82_SUB_MDL_ERR	BOOL	子模块丢失或存在错误
OB82_COMM_FAULT	BOOL	通信问题
OB82_MDL_STOP	BOOL	工作模式(0：RUN，1：STOP)
OB82_WTCH_DOG_FLT	BOOL	看门狗定时器响应
OB82_INT_PS_FLT	BOOL	内部电源故障
OB82_PRIM_BATT_FLT	BOOL	电池耗尽
OB82_BCKUP_BATT_FLT	BOOL	整个备份失败
OB82_RESERVED_2	BOOL	急需维护
OB82_RACK_FLT	BOOL	扩展机架故障

变量	类型	描述
OB82_PROC_FLT	BOOL	处理器故障
OB82_EPROM_FLT	BOOL	EPROM 故障的帮助信息
OB82_RAM_FLT	BOOL	RAM 故障的帮助信息
OB82_ADU_FLT	BOOL	ADC/DAC 错误
OB82_FUSE_FLT	BOOL	保险丝断开
OB82_HW_INTR_FLT	BOOL	硬件中断丢失
OB82_RESERVED_3	BOOL	保留
OB82_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

**注意**

如果使用呢具有 DPV1 功能的 CPU，可借助于 SFB54 "RALRM"获取有关中断的其它信息，这些信息不止包含 OB 的起始信息。在 S7 兼容模式下操作 DP 主站时，这一点也同样适用。

## 1.19 插入/删除模块中断组织块(OB83)

### 描述

在下列情况下，CPU 操作系统会调用 OB83：

- 插入/删除已组态模块后
  - 在 STEP 7 下作为 RUN 模式中的组态的一部分修改模块参数以及在运行期间将更改下载至 CPU 后如果尚未编程 OB83，则 CPU 将切换到 STOP 模式
- 可借助 SFC 39 至 42 禁用/延迟/启用插入/删除中断 OB。

### 模块插入和删除

每次在 RUN、STOP 和 STARTUP 模式期间删除或插入已组态模块时，都会生成插入/删除中断(在这些模式下一定不能删除电源模块、CPU、适配器模块以及 IM)。此中断会导致在诊断缓冲区和所涉及 CPU 的系统状态列表中生成一个条目。如果 CPU 处于 RUN 模式，则还会启动插入/删除 OB。如果此 OB 尚未编程，CPU 将转为 STOP 模式。

随后，系统会以数秒钟的时间间隔来轮询 S7-400 模块以检测插入或删除。为使 CPU 能够检测到 S7-400 模块的删除和插入，在删除和插入操作之间必须至少有 2s 的时间间隔。对于其它模块而言，此最短时间值会稍大一些。

如果在 RUN 模式下删除了已组态的模块，则启动 OB83。由于仅以 1s 为时间间隔来监视模块的存在，因此当直接访问模块或更新过程映像时，可能会首先检测到访问错误。

如果在 RUN 模式下将一个模块插入到已组态的插槽中，则操作系统将检查所插入模块的类型是否与所记录的组态相符。如果模块类型匹配，则启动 OB83 并分配参数。

### S7-300 的特别注意事项

- 不能通过 S7-300 来插入和删除中央 IO 设备。
- 至于分布式 IO 设备，CPU 318 的运作方式与 S7-400-CPU 类似。对于所有其它 S7-300 CPU，只有 31x PN/DP CPU 具有插入/删除中断，但仅适用于 PROFINET IO 组件。
- 对于 IM151/CPU (具有 ET 200S 的 CPU)，只有中央 IO 设备具有插入/删除中断。

## 重新组态模块

当在运行(CiR)期间修改系统组态时，可为现有模块重新分配参数。参数的这种重新分配通过将所需参数数据记录传送至模块实现。

步骤如下：

1. OB83 将在使用 STEP 7 为模块分配新的参数并将该组态下载到处于 RUN 模式的 CPU 中后启动(启动事件 W#16#3367)。相关的 OB 启动信息为逻辑基址(OB83\_MDL\_ADDR)和模块类型(OB83\_MDL\_TYPE)。从现在起，模块 I/O 数据可能不正确，这就意味着没有任何 SFC 忙于将数据记录发送到此模块。
2. 模块参数将在执行 OB83 后重新分配。
3. OB83 将在分配参数后重新启动(启动事件 W#16#3267，前提是参数分配成功，如果失败，则为 W#16#3968)。模块的 I/O 数据响应与其在插入中断后的响应相同，也就是说，当前它们可能不正确。现在可再次调用 SFC 以便将数据记录发送到模块。

## 重新组态子模块

当在运行(CiR)期间修改系统组态时，可为现有子模块重新分配参数。参数的这种重新分配通过将所需参数数据记录传送至子模块实现。

步骤如下：

1. OB83 将在使用 STEP 7 为子模块分配新的参数并将该组态下载到处于 RUN 模式的 CPU 中后启动(启动事件 W#16#335A)。相关的 OB 启动信息为逻辑基址(OB83\_MDL\_ADDR)和模块类型(OB83\_MDL\_TYPE)。如果要重新配置的子模块是 CombinedObjectContainer 组的一部分，则在 OB 83 的起始信息中输入 COC 组的替代物(此处为接口)的逻辑基址、模块类型。  
注意：一个插槽的多个 PDev 子模块(接口、端口)组合成一个 CombinedObjectContainer 组 (COC 组)。在本例中，接口子模块是 COC 组的替代物。  
从现在起，子模块 I/O 数据可能不正确，这就意味着没有任何 SFC 忙于将数据记录发送到此子模块。
2. 子模块参数将在执行 OB83 后重新分配。
3. OB83 将在分配参数后重新启动(启动事件 W#16#325A，前提是参数分配成功，如果失败，则为#16#395B)。子模块的 I/O 数据响应与其在插入中断后的响应相同，也就是说，当前它们可能不正确。现在可再次调用 SFC 以便将数据记录发送到子模块。

用于插入/删除 OB 的本地数据

下表说明了插入/删除模块中断 OB 的临时(TEMP)变量。变量名为 OB83 的默认名称。

变量	类型	描述
OB83_EV_CLASS	BYTE	事件等级和标识符： <ul style="list-style-type: none"> <li>• B#16#32：重新分配模块或子模块参数结束</li> <li>• B#16#33：重新分配模块或子模块参数开始</li> <li>• B#16#38：模块已插入</li> <li>• B#16#39：模块被删除或未响应，或者参数分配结束时出错</li> </ul>
OB83_FLT_ID	BYTE	错误代码：(可能值 B#16#51、B#16#54、B#16#55、B#16#56、B#16#57、B#16#58、B#16#5A、B#16#5B、B#16#61、B#16#63、B#16#64、B#16#65、B#16#66、B#16#67、B#16#68)
OB83_PRIORITY	BYTE	<ul style="list-style-type: none"> <li>• 优先级；可通过 STEP 7 分配(硬件配置)</li> </ul>
OB83_OB_NUMBR	BYTE	OB 编号(83)
OB83_RESERVED_1	BYTE	模块或子模块/接口模块的标识
OB83_MDL_TD	BYTE	范围： <ul style="list-style-type: none"> <li>• B#16#54：外围输入(PI)</li> <li>• B#16#55：外围输出(PO)</li> </ul>
OB83_MDL_ADDR	WORD	<ul style="list-style-type: none"> <li>• 中央或分布式 PROFIBUS DP：受影响模块的逻辑基址。如果它是混合模块，则该地址为模块中所使用的最小逻辑地址。如果混合块中的 I 地址和 O 地址相等，则逻辑基址为接收事件标识符的地址。</li> <li>• 分布式 PROFINET IO：模块/子模块的逻辑基址</li> </ul>
OB83_RACK_NUM	WORD	<ul style="list-style-type: none"> <li>• 如果 OB83_RESERVED_1 = B#16#A0：子模块/接口子模块的编号(低字节)</li> <li>• 如果 OB83_RESERVED_1 = B#16#C4：                             <ul style="list-style-type: none"> <li>- 集中式：机架号</li> <li>- 分布式 PROFIBUS DP：DP 站的编号(低字节)和 DP 主站系统 ID (高字节)</li> <li>- 分布式 PROFINET IO：物理地址：标识符位(位 15、1 = PROFINET IO)，IO 系统 ID (位 11 至 14)和设备编号(位 0 至 10)</li> </ul> </li> </ul>



变量	类型	描述
OB83_MDL_TYPE	WORD	<ul style="list-style-type: none"> <li>• 中央或分布式 PROFIBUS DP : 受影响模块的类型 (X : 与用户无关) : <ul style="list-style-type: none"> <li>- W#16#09CA : ET200SP F-PM-E 电源模块</li> <li>- W#16#09E0: ET200SP F-DQ 模块</li> <li>- W#16#X5XX : 模拟模块</li> <li>- W#16#X6XX : 网关</li> <li>- W#16#X8XX : 功能模块</li> <li>- W#16#XCXX : CP</li> <li>- W#16#XFXX : 数字模块</li> <li>- W#16#8340 : 输入模块的替换类型 ID</li> <li>- W#16#9340 : 输出模块的替换类型 ID</li> <li>- W#16#A340 : 混合模块(I/O)的替换类型 ID</li> <li>- W#16#F340 : 空模块或不能唯一识别的模块(例如 , 带压缩地址)的替换类型 ID</li> <li>- 此处未列出的模块类型在对应模块的手册中叙明。</li> </ul> </li> <li>• 分布式 PROFINET IO : <ul style="list-style-type: none"> <li>- W#16#8101 : 插入的模块类型与拔出的模块类型相同</li> <li>- W#16#8102 : 插入的模块类型与拔出的模块类型不相同</li> </ul> </li> </ul>
OB83_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

下表给出了启动 OB83 的事件：

OB83_EV_CLASS	OB83_FLT_ID	含义
B#16#39	B#16#51	PROFINET IO 模块已删除
B#16#39	B#16#54	PROFINET IO 子模块被删除
B#16#38	B#16#54	PROFINET IO 子模块已插入且与已组态的子模块匹配
B#16#38	B#16#55	PROFINET IO 子模块已插入，但与已组态的子模块不匹配
B#16#38	B#16#56	PROFINET IO 子模块已插入，但模块参数存在错误
B#16#38	B#16#57	PROFINET IO 子模块或模块已插入，但出错或需要维护
B#16#38	B#16#58	PROFINET IO 子模块访问错误已更正
B#16#33	B#16#5A	启动重新组态子模块
B#16#32	B#16#5A	结束重新组态子模块
B#16#39	B#16#5B	结束重新组态子模块时出错
B#16#39	B#16#61	模块已删除或未响应 OB83_MDL_TYPE：实际模块类型
B#16#38	B#16#61	模块已插入。模块类型正确 OB83_MDL_TYPE：实际模块类型
B#16#38	B#16#63	模块已插入，但模块类型不正确 OB83_MDL_TYPE：实际模块类型
B#16#38	B#16#64	模块已插入，但存在问题(无法读取模块 ID) OB83_MDL_TYPE：已组态的模块类型
B#16#38	B#16#65	模块已插入，但模块参数分配有错误 OB83_MDL_TYPE：实际模块类型
B#16#39	B#16#66	模块未响应，负载电压错误
B#16#38	B#16#66	模块再次响应，已更正负载电压错误
B#16#33	B#16#67	模块重组态开始
B#16#32	B#16#67	模块重组态结束
B#16#39	B#16#68	模块重新组态因错误而终止

### 注意

如果使用了具有 DPV1 或 PROFINET 功能的 CPU，可借助于 SFB54 "RALRM"获取有关中断的其它信息，这些信息不止包含 OB 的起始信息。在 S7 兼容模式下操作 DP 主站时，这一点也同样适用。

## 1.20 CPU 硬件故障组织块(OB84)

### 描述

在下列情况下，CPU 中的 OS 将调用 OB84：

- 已检测到并更正了内存错误之后
- 组态最高(包括) V4.5 版本的 H CPU：如果两个 CPU 之间的冗余链接的性能下降  
注意：当组态从 V6.0 版本起的 H CPU 时，在 CPU 之间的冗余耦合性能下降时调用诊断中断 OB (OB82)。
- 对于 S7-400H：如果两个 CPU 之间的冗余链接的性能下降
- 对于 WinAC RTX：操作系统出错(例如"蓝屏")

如果未编程 OB84，则 CPU 将不切换到 STOP 模式

可以使用 SFC 39 至 42 禁用或延迟 CPU 硬件错误 OB，然后再次启用它。

### CPU 硬件故障 OB 的本地数据

下表包括了 CPU 硬件故障的临时(TEMP)变量。变量名为 OB84 的默认名称。

变量	类型	描述
OB84_EV_CLASS	BYTE	事件等级和标识符： <ul style="list-style-type: none"> <li>• B#16#38：离开事件</li> <li>• B#16#35、B#16#39：进入事件</li> </ul>
OB84_FLT_ID	BYTE	错误代码(B#16#82、B#16#83、B#16#85、B#16#86、B#16#87、B#16#88、B#16#92)
OB84_PRIORITY	BYTE	优先级；可通过 STEP 7 分配(硬件配置)
OB84_OB_NUMBR	BYTE	OB 编号(84)
OB84_RESERVED_1	BYTE	保留
OB84_RESERVED_2	BYTE	保留
OB84_RESERVED_3	WORD	保留
OB84_RESERVED_4	DWORD	保留
OB84_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

如果 OB84\_FLT\_ID = B#16#88，应在下一次有机会时更换受影响的 CPU。CPU 可按照如下所示由 OB84\_RESERVED\_3 的位 8 和 9 标识：

- 位 8 = 1：机架 0 中出错
- 位 9 = 1：机架 1 中出错

如果 OB84\_FLT\_ID = B#16#92，则应在有机会时更换受影响的 CPU。此时，OB84\_RESERVED\_4 变量的值为 DW#16#000xFCAC。CPU 可按照如下所示使用 OB84\_RESERVED\_4 的位 16 至 19 标识：

- 位 16 = 1：机架 0 模块 1 中出错
- 位 17 = 1：机架 0 模块 2 中出错
- 位 18 = 1：机架 1 模块 1 中出错
- 位 19 = 1：机架 1 模块 2 中出错

下表给出了启动 OB84 的事件：

OB84_EV_CLASS	OB84_FLT_ID	OB84 的启动事件
B#16#35	B#16#82	在操作系统中检测到内存错误并已更正
B#16#35	B#16#83	已检测到并更正的内存错误的累积
B#16#35	B#16#85	PC 操作系统中的错误
B#16#35	B#16#86	两个 CPU 之间的冗余耦合的性能下降(仅用于组态最高(包括) V4.5 版本的 H-CPU)
B#16#35	B#16#87	检测到并更正了多位存储器错误
B#16#35	B#16#92	检测到 CPU 外部组件而导致的故障
B#16#35	B#16#88	保持的集成装载存储器中检测到限制
B#16#35	B#16#92	检测到 CPU 外部组件而导致的故障

## 1.21 优先级错误组织块(OB85)

### 描述

只要发生下列事件之一，CPU 的操作系统即调用 OB85：

- 尚未装载的 OB (OB8、OB81、OB82、OB83 和 OB86 除外)的启动事件。
- 操作系统访问模块时出错。
- 在系统更新过程映像期间出现 I/O 访问错误(如果由于组态原因，未禁止 OB85 的调用)。

### 注意

如果 OB85 尚未编程，则当检测到这些事件其中之一时，CPU 会转为 STOP 模式。

可以使用 SFC 39 至 42 禁用或延迟，并重新启用优先级错误 OB。

### 优先级错误 OB 的本地数据

下表说明了优先级错误 OB 的临时(TEMP)变量。变量名为 OB85 的默认名称。

变量	类型	描述
OB85_EV_CLASS	BYTE	事件等级和标识符：B#16#35 B#16#38 (仅限于错误代码 B#16#B3 和 B#16#B4) B#16#39 (仅限于错误代码 B#16#B1、B#16#B2、B#16#B3 和 B#16#B4)
OB85_FLT_ID	BYTE	错误代码(可能值：B#16#A1、B#16#A2、B#16#A3、B#16#A4、B#16#B1、B#16#B2、B#16#B3、B#16#B4)
OB85_PRIORITY	BYTE	• 优先级；可通过 STEP 7 分配(硬件组态)
OB85_OB_NUMBR	BYTE	OB 编号(85)
OB85_RESERVED_1	BYTE	保留
OB85_RESERVED_2	BYTE	保留
OB85_RESERVED_3	INT	保留
OB85_ERR_EV_CLASS	BYTE	引起错误的事件的等级
OB85_ERR_EV_NUM	BYTE	引起错误的事件的编号
OB85_OB_PRIOR	BYTE	在出错时处于活动状态的 OB 的优先级(仅限某些错误代码，欲知详情，请参见下文)
OB85_OB_NUM	BYTE	在出错时处于活动状态的 OB 的编号(仅限某些错误代码，欲知详情，请参见下文)
OB85_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

如果要对依赖于可能错误代码的 OB85 进行编程，建议按如下方式组织局部变量：

变量	类型
OB85_EV_CLASS	BYTE
OB85_FLT_ID	BYTE
OB85_PRIORITY	BYTE
OB85_OB_NUMBR	BYTE
OB85_DKZ23	BYTE
OB85_RESERVED_2	BYTE
OB85_Z1	WORD
OB85_Z23	DWORD
OB85_DATE_TIME	DATE_AND_TIME

下表给出了启动 OB85 的事件及如何占用与错误代码相关的变量：

OB85_EV_CLASS	OB85_FLT_ID	含义
B#16#35	B#16#A1	<p>由于使用 STEP 7 创建了组态，您的程序或操作系统将为未装载到 CPU 上的 OB 创建一个启动事件。</p> <ul style="list-style-type: none"> <li>OB85_Z1：由 OB85_Z23 确定的被调用 OB 的各个局部变量。</li> <li>OB85_Z23：                             <ul style="list-style-type: none"> <li>高位字：导致 OB 调用的事件的等级和编号</li> <li>低位字，高位字节：程序级别以及出错时处于活动状态的 OB</li> <li>低位字，低位字节：激活的 OB</li> </ul> </li> </ul>
B#16#35	B#16#A2	<p>由于使用 STEP 7 创建了组态，您的程序或操作系统将为未装载到 CPU 上的 OB 创建一个启动事件。</p> <p>OB85_Z1 和 OB85_Z23(对于 OB85_FLT_ID=B#16#A1)</p>
B#16#35	B#16#A3	<p>操作系统访问模块时出错</p> <ul style="list-style-type: none"> <li>OB85_Z1：操作系统的错误 ID                             <ul style="list-style-type: none"> <li>高字节：1= 集成的功能，2=IEC 定时器</li> <li>低字节：0= 无错误分辨率，1= 未装载块，2= 区域长度错误，3= 写保护错误</li> </ul> </li> <li>OB85_Z23：                             <ul style="list-style-type: none"> <li>高位字：块编号</li> <li>低位字：导致错误的 MC7 命令的相对地址。块类型必须取自 OB85_DKZ23 (B#16#88：OB，B#16#8C：FC，B#16#8E：FB，B#16#8A：DB)。</li> </ul> </li> </ul>
B#16#35	B#16#A4	不能寻址 PROFINET 接口 DB
B#16#34	B#16#A4	不能再次寻址 PROFINET 接口 DB

OB85_EV_CLASS	OB85_FLT_ID	含义
B#16#39	B#16#B1	<p>更新输入的过程映像时出现 I/O 访问错误</p> <ul style="list-style-type: none"> <li>• OB85_DKZ23 : 过程映像传送(在此期间发生 I/O 访问错误)类型的 ID <ul style="list-style-type: none"> <li>- B#16#10 : 字节访问</li> <li>- B#16#20 : 字访问</li> <li>- B#16#30 : 双字访问</li> <li>- B#16#56 或 B#16#57 : 传送组态的一致性范围</li> </ul> </li> <li>• OB85_Z1 : 留作 CPU 内部使用 如果 OB85_RESERVED_2 的值为 B#16#76 , 则 OB85_Z1 将接收受影响的 SFC (SFC 14、15、26 或 27)的返回值。</li> <li>• OB85_Z23 : <ul style="list-style-type: none"> <li>- 字节 0 : 如果 OB85_DKZ23=B#16#57 : 部分过程映像编号</li> <li>- 字节 1 : 如果 OB85_DKZ23=B#16#10、20 或 30 , 则不相关 ; 如果 OB85_DKZ23=B#16#57 , 则为以字节为单位的一致性范围的长度</li> <li>- 字节 0 和 1 , 如果 OB85_DKZ23=B#16#56 : 一致性范围的字节长度</li> <li>- 字节 2 和 3 : 如果 OB85_DKZ23=B#16#10、20 或 30 , 则为导致 PAE 的 I/O 地址 ; 如果 OB85_DKZ23=B#16#57 , 则为一致性范围的逻辑起始地址</li> </ul> </li> </ul>
B#16#39	B#16#B2	<p>当将输出过程映像传送到输出模块时出现 I/O 访问错误</p> <p>OB85_DKZ23、OB85_Z1 和 OB85_Z23(对于 OB85_FLT_ID=B#16#B1)</p>
<p>如果已为系统过程映像表更新组态了 I/O 访问错误的重复 OB85 调用 , 则会获得错误代码 B#16#B1 和 B#16#B2。</p>		
B#16#39/B#16#38	B#16#B3	<p>当更新输入的过程映像、进入/离开事件时出现 I/O 访问错误</p> <ul style="list-style-type: none"> <li>• OB85_DKZ23 : 过程映像传送(在此期间发生 I/O 访问错误)类型的 ID <ul style="list-style-type: none"> <li>- B#16#10 : 字节访问</li> <li>- B#16#20 : 字访问</li> <li>- B#16#30 : 双字访问</li> <li>- B#16#57 或 B#16#57 : 传送组态的一致性范围</li> </ul> </li> <li>• OB85_Z1 : 留作 CPU 内部使用 如果 OB85_RESERVED_2 的值为 B#16#76 , 则 OB85_Z1 将接收受影响的 SFC (SFC 14、15、26 或 27)的返回值。</li> <li>• OB85_Z23 : <ul style="list-style-type: none"> <li>- 字节 0 : 如果 OB85_DKZ23=B#16#57 : 部分过程映像编号</li> <li>- 字节 1 : 如果 OB85_DKZ23=B#16#10、20 或 30 , 则不相关 ; 如果 OB85_DKZ23=B#16#57 , 则为以字节为单位的一致性范围的长度</li> <li>- 字节 0 和字节 1 , 如果 OB85_DKZ23=B#16#56 : 一致性范围的字节长度</li> <li>- 字节 2 和 3 : 如果 OB85_DKZ23=B#16#10、20 或 30 , 则为导致 PAE 的 I/O 地址 ; 如果 OB85_DKZ23=B#16#57 , 则为一致性范围的逻辑起始地址</li> </ul> </li> </ul>

OB85_EV_CLASS	OB85_FLT_ID	含义
B#16#39/B#16#38	B#16#B4	当更新输出的过程映像时出现 I/O 访问错误，进入/离开事件 OB85_DKZ23、OB85_Z1、OB85_Z23(对于 OB85_FLT_ID=B#16#B3)
<p>如果已为由系统进行的过程映像表更新组态了 I/O 访问错误进入和离开事件的 OB85 调用，则会获得错误代码 B#16#B3 和 B#16#B4。冷重启或暖重启后，在下次更新过程映像表期间，对非现有输入和输出的所有访问均报告为 I/O 访问错误。</p>		



## 1.22 机架故障组织块(OB86)

### 描述

在以下情况下，CPU 的操作系统调用 OB86：

- 中央扩展单元(不带 S7-300)的故障已经删除(事件进入和退出状态)。
- DP 主站系统的故障已删除(事件进入和退出状态)。
- 使用分布式 I/O (PROFIBUS DP 或 PROFINET IO)时检测到站故障(事件进入和退出状态)。
- 使用分布式 I/O (PROFIBUS DP 或 PROFINET IO)时禁用了一个带 SFC 12 "D\_ACT\_DP"和设置 MODE=4 的站。
- 使用分布式 I/O (PROFIBUS DP 或 PROFINET IO)时启用了带 SFC 12 "D\_ACT\_DP"和设置 MODE=3 的站。
- 检测到 PROFINET IO 系统或 PROFINET IO 站或 PROFINET I 设备的某些子模块发生故障。

如果 OB86 尚未编程，当检测到此种类型的错误时，CPU 将转为 STOP 模式。

可使用 SFC 39 至 42 禁用或延迟，并重新启用 OB86。

### 机架故障 OB 的本地数据

下表说明了机架故障 OB 的临时(TEMP)变量。变量名为 OB86 的默认名称。

变量	类型	描述
OB86_EV_CLASS	BYTE	事件等级和标识符： <ul style="list-style-type: none"> <li>• B#16#32：启用一个带 SFC 12 "D_ACT_DP"和 MODE=3 的站</li> <li>• B#16#33：禁用一个带 SFC 12 "D_ACT_DP"和 MODE=3 的站</li> <li>• B#16#38：事件退出状态</li> <li>• B#16#39：事件输入状态</li> </ul>
OB86_FLT_ID	BYTE	错误代码：(可能值：B##16#C1、B#16#C2、B#16#C3、B#16#C4、B#16#C5、B#16#C6、B#16#C7、B#16#C8、B#16#C9、B#16#CA、B#16#CB、B#16#CC、B#16#CD、B#16#CE、B#16#CF、B#16#F8、B#16#F9))
OB86_PRIORITY	BYTE	优先级；可通过 STEP 7 分配(硬件配置)
OB86_OB_NUMBR	BYTE	OB 编号(86)
OB86_RESERVED_1	BYTE	保留
OB86_RESERVED_2	BYTE	保留
OB86_MDL_ADDR	WORD	取决于错误代码
OB86_RACKS_FLTD	数组[0..31] BOOL 型	取决于错误代码
OB86_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

如果要对依赖于可能错误代码的 OB86 进行编程，建议按如下方式组织局部变量：

变量	类型
OB86_EV_CLASS	BYTE
OB86_FLT_ID	BYTE
OB86_PRIORITY	BYTE
OB86_OB_NUMBR	BYTE
OB86_RESERVED_1	BYTE
OB86_RESERVED_2	BYTE
OB86_MDL_ADDR	WORD
OB86_Z23	DWORD
OB86_DATE_TIME	DATE_AND_TIME

下表给出了启动 OB86 的事件：

OB86_EV_CLASS	OB86_FLT_ID	含义
B#16#39	B#16#C1	<p>扩展机架故障</p> <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR IM 的逻辑基址</li> <li>• OB86_Z23：针对于每个可能的扩展机架包含一个位：导致调用 OB86 的每个扩展机架均报告为已出现故障(置位各个位)。不再显示之前出现故障的扩展机架。                             <ul style="list-style-type: none"> <li>- 位 0：始终为 0</li> <li>- 位 1：扩展机架 1</li> <li>- :</li> <li>- 位 21：扩展机架 21</li> <li>- 22 到 29 位：始终为 0</li> <li>- 位 30：在 SIMATIC S5 区域中至少有一个扩展机架出现了故障</li> <li>- 位 31：始终为 0</li> </ul> </li> </ul>
B#16#38	B#16#C1	<p>扩展机架再次可运行</p> <p>OB86_MDL_ADDR(对于 OB86_FLT_ID=B#16#C1)。再次可运行的扩展机架在 OB86_Z23 中进行报告(置位各个位)。</p>
B#16#38	B#16#C2	<p>扩展机架再次可运行(扩展机架因预期组态与实际组态不符而出现故障)</p> <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR IM 的逻辑基址</li> <li>• OB86_Z23：针对每个可能扩展机架包含一个位，请参见 OB86_FLT_ID B#16#C1。 已置位的含义：在受影响机架中：                             <ul style="list-style-type: none"> <li>- 存在具有错误类型 ID 的模块</li> <li>- 已组态的模块丢失</li> <li>- 至少有一个模块故障。</li> </ul> </li> </ul>

OB86_EV_CLASS	OB86_FLT_ID	含义
B#16#39	B#16#C3	<p>分布式 I/O 设备：DP 主站系统故障 仅进入的事件导致 OB86 启动，错误代码为 B#16#C3。离开的事件使用错误代码 B#16#C4 和事件等级 B#16#38 启动 OB86： 每个 DP 从站的返回会启动 OB86</p> <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR DP 主站的逻辑基址</li> <li>• OB86_Z23：DP 主站 ID： <ul style="list-style-type: none"> <li>- 位 0 到 7：保留</li> <li>- 位 8 到 15：DP 主站系统 ID</li> <li>- 位 16 到 31：保留</li> </ul> </li> </ul>
B#16#39/B#16#38	B#16#C4	<p>DP 站故障</p> <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR DP 主站的逻辑基址</li> <li>• OB86_Z23：受影响 DP 从站的地址： <ul style="list-style-type: none"> <li>- 位 0 到 7：DP 站编号</li> <li>- 位 8 到 15：DP 主站系统 ID</li> <li>- 位 16 到 30：S7 从站的逻辑基址或者标准 DP 从站的诊断地址</li> <li>- 位 31：I/O 标识符</li> </ul> </li> </ul>
B#16#39/B#16#38	B#16#C5	<p>返回站，但站发生故障 OB86_MDL_ADDR 和 OB86_Z23(对于 FLT_ID=B#16#C4)</p>
B#16#38	B#16#C6	<p>扩展机架可再次运行，但模块参数仍存在错误。</p> <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR IM 的逻辑基址</li> <li>• OB86_Z23：包含针对于每个可能的扩展机架的一个位： <ul style="list-style-type: none"> <li>- 位 0：始终为 0</li> <li>- 位 1：扩展机架 1</li> <li>- ：</li> <li>- 位 21：扩展机架 21</li> <li>- 位 22 到 30：保留</li> <li>- 位 31：始终为 0</li> </ul> </li> <li>• 置位时的含义(在所涉及的扩展机架中)： <ul style="list-style-type: none"> <li>- 存在具有错误类型标识符的模块</li> <li>- 存在缺少参数或具有错误参数的模块。</li> </ul> </li> </ul>
B#16#38	B#16#C7	<p>返回 DP 站，但模块参数分配存在错误</p> <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR DP 主站的逻辑基址</li> <li>• 受影响 DP 从站的地址： <ul style="list-style-type: none"> <li>- 位 0 到 7：DP 站编号</li> <li>- 位 8 到 15：DP 主站系统 ID</li> <li>- 位 16 到 30：DP 从站的逻辑基址</li> <li>- 位 31：I/O 标识符</li> </ul> </li> </ul>
B#16#38	B#16#C8	<p>返回 DP 站，但所配置的组态和实际组态不符</p> <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR DP 主站的逻辑基址</li> <li>• OB86_Z23：受影响 DP 从站的地址： <ul style="list-style-type: none"> <li>- 位 0 到 7：DP 站编号</li> <li>- 位 8 到 15：DP 主站系统 ID</li> <li>- 位 16 到 30：DP 从站的逻辑基址</li> <li>- 位 31：I/O 标识符</li> </ul> </li> </ul>

OB86_EV_CLASS	OB86_FLT_ID	含义
B#16#32/B#16#33	B#16#C9	启用/禁用一个带 SFC 12 "D_ACT_DP"和 MODE=3/MODE=4 的 DP 从站 <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR DP 主站的逻辑基址</li> <li>• OB86_Z23 : 涉及的 DP 从站的地址 :                             <ul style="list-style-type: none"> <li>- 位 0 到 7 : DP 站编号</li> <li>- 位 8 到 15 : DP 主站系统 ID</li> <li>- 位 16 到 30 : S7 从站的逻辑基址或 DP 标准从站的诊断地址</li> <li>- 位 31 : I/O 标识符</li> </ul> </li> </ul>
B#16#39	B#16#CA	PROFINET IO 系统故障 <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR IO 系统的逻辑地址</li> <li>• OB86_Z23 :                             <ul style="list-style-type: none"> <li>- 位 0 到 10 : 0 (站号)</li> <li>- 位 11 到 14 : IO 系统 ID</li> <li>- 位 15 : 1</li> <li>- 位 16 到 31 : 0</li> </ul> </li> </ul>
B#16#39/38	B#16#CB	PROFINET IO 站故障/站返回 <ul style="list-style-type: none"> <li>• OB86_RESERVED_1 : B#16#C4</li> <li>• OB86_MDL_ADDR IO 系统的逻辑地址</li> <li>• OB86_Z23 :                             <ul style="list-style-type: none"> <li>- 位 0 到 10 : 站号</li> <li>- 位 11 到 14 : IO 系统 ID</li> <li>- 位 15 : 1</li> <li>- 位 16 到 30 : 站的逻辑基址</li> <li>- 位 31 : I/O 标识符</li> </ul> </li> </ul> 注意 : 在程序中检查 OB 86 启动的原因是否是 PROFINET IO 站故障之前, 应检查是否存在 PROFINET IO 系统故障。
B#16#38	B#16#CC	1. PROFINET IO 站因出错或需要维护而返回 2. OB86_RESERVED_1 : B#16#C4 3. OB86_MDL_ADDR IO 控制器的逻辑基址 4. OB86_Z23 : 5. 位 0 到 10 : 站号 6. 位 11 到 14 : IO 系统 ID 7. 位 15 : 1 8. 位 16 到 30 : 站的逻辑基址 9. 位 31 : I/O ID
B#16#38	B#16#CD	PROFINET IO 站返回, 预期和实际组态不同 <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR IO 控制器的逻辑基址</li> <li>• OB86_Z23 :                             <ul style="list-style-type: none"> <li>- 位 0 到 10 : 站号</li> <li>- 位 11 到 14 : IO 系统 ID</li> <li>- 位 15 : 1</li> <li>- 位 16 到 30 : 站的逻辑基址</li> <li>- 位 31 : I/O ID</li> </ul> </li> </ul>

OB86_EV_CLASS	OB86_FLT_ID	含义
B#16#38	B#16#CE	PROFINET IO 站返回，模块组态参数存在错误 <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR IO 控制器的逻辑基址</li> <li>• OB86_Z23 : <ul style="list-style-type: none"> <li>- 位 0 到 10 : 设备编号</li> <li>- 位 11 到 14 : IO 系统 ID</li> <li>- 位 15 : 1</li> <li>- 位 16 到 30 : 站的逻辑基址</li> <li>- 位 31 : I/O 标识符</li> </ul> </li> </ul>
B#16#32/B#16#33	B#16#CF	启用/禁用带 SFC 12 "D_ACT_DP"和 MODE=3/MODE=4 的 PROFINET IO 设备 <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR IO 控制器的逻辑基址</li> <li>• OB86_Z23 : <ul style="list-style-type: none"> <li>- 位 0 到 10 : 站号</li> <li>- 位 11 到 14 : IO 系统 ID</li> <li>- 位 15 : 1</li> <li>- 位 16 到 30 : 站的逻辑基址</li> <li>- 位 31 : I/O 标识符</li> </ul> </li> </ul>
B#16#39/B#16#38	B#16#F8	PROFINET I 设备的某些子模块发生故障/返回 <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR IO 控制器的逻辑基址</li> <li>• OB86_Z23 : <ul style="list-style-type: none"> <li>- 位 0 到 10 : 站号</li> <li>- 位 11 到 14 : IO 系统 ID</li> <li>- 位 15 : 1</li> <li>- 位 16 到 30 : 站的逻辑基址</li> <li>- 位 31 : I/O ID</li> </ul> </li> </ul>
B#16#38	B#16#F9	PROFINET I 设备的某些子模块因设备配置不同而发生故障/返回 <ul style="list-style-type: none"> <li>• OB86_MDL_ADDR IO 控制器的逻辑基址</li> <li>• OB86_Z23 : <ul style="list-style-type: none"> <li>- 位 0 到 10 : 站号</li> <li>- 位 11 到 14 : IO 系统 ID</li> <li>- 位 15 : 1</li> <li>- 位 16 到 30 : 站的逻辑基址</li> <li>- 位 31 : I/O ID</li> </ul> </li> </ul>

**注意**

如果使用 DPV1 兼容的 CPU，可借助于 SFB54 "RALRM"获取有关中断的其它信息，这些信息超出了 OB 的起始信息。在 S7 兼容模式下操作 DP 主站时，这一点也同样适用。

## 1.23 通信错误组织块(OB87)

### 描述

当发生由通信错误导致的事件，CPU 的操作系统就会调用 OB87。

如果 OB87 未编程，且发生 OB87 的启动事件，则 CPU 将作如下反应：

- S7-300 CPU 切换至 STOP 模式。
- S7-400 CPU 未切换至 STOP 模式。

可以使用 SFC 39 至 42 禁用或延迟，并重新启用通信错误 OB。

### OB87 的本地数据

下表说明了通信错误 OB 的临时(TEMP)变量。变量名为 OB87 的默认名称。

变量	类型	描述
OB87_EV_CLASS	BYTE	事件等级和标识符：B#16#35
OB87_FLT_ID	BYTE	错误代码： (可能值：B#16#D2、B#16#D3、B#16#D4、B#16#D5、 B#16#E1、B#16#E2、B#16#E3、B#16#E4、B#16#E5、 B#16#E6)
OB87_PRIORITY	BYTE	• 优先级; 可通过 STEP 7 分配(硬件配置)
OB87_OB_NUMBR	BYTE	OB 编号(87)
OB87_RESERVED_1	BYTE	保留
OB87_RESERVED_2	BYTE	保留
OB87_RESERVED_3	WORD	取决于错误代码
OB87_RESERVED_4	DWORD	取决于错误代码
OB87_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

取决于错误代码的变量具有如下：

错误代码：	字节/字	含义
<ul style="list-style-type: none"> <li>• B#16#D2：</li> <li>• B#16#D3</li> <li>• B#16#D4：</li> <li>• B#16#D5</li> </ul>		当前无法传送诊断条目。 无法传送同步消息(主站)。 因时钟同步而导致非法时间跳转。 接收同步时间时出错(从站)。
<ul style="list-style-type: none"> <li>- OB87_RESERVED_3：</li> <li>- OB87_RESERVED_4：</li> </ul>		未包含附加信息。 未包含附加信息。

错误代码：	字节/字	含义
<ul style="list-style-type: none"> <li>• B#16#E1：</li> <li>• B#16#E3：</li> <li>• B#16#E4：</li> <li>- OB87_RESERVED_3：</li> <li>- OB87_RESERVED_4：</li> </ul>	高字节： 低字节：	在全局数据通信期间帧 ID 不正确。 在全局数据通信期间帧长度出错。 收到非法的 GD 信息包编号。 接口 ID (0：K 总线，1：MPI) GD 电路编号 未包含附加信息。
<ul style="list-style-type: none"> <li>• B#16#E2：</li> <li>- OB87_RESERVED_3：</li> <li>- OB87_RESERVED_4：</li> </ul>	高位字： 低位字：	无法将 GD 信息包状态输入到 DB 中 DB 编号 未包含附加信息。 <ul style="list-style-type: none"> <li>• GD 回路数目(高位字节)，</li> <li>• GD 信息包编号(低位字节)</li> </ul>
<ul style="list-style-type: none"> <li>• B#16#E5：</li> <li>- OB87_RESERVED_3：</li> <li>- OB87_RESERVED_4：</li> </ul>	高位字： 低位字：	通过通信功能块进行数据交换期间出现 DB 访问错误 留作 CPU 内部使用。 包含导致错误的 MC7 命令的块编号。 导致错误的 MC7 命令的相对地址。

块类型可从 OB\_87\_RESERVED\_1 读取(B#16#88：OB，B#16#8A：DB，B#16#8C：FC，B#16#8E：FB)。

错误代码：	含义
<ul style="list-style-type: none"> <li>• B#16#E6：</li> <li>OB87_RESERVED_3：</li> <li>OB87_RESERVED_4：</li> </ul>	无法将 GD 组状态输入到 DB 中。 DB 编号。 未包含附加信息。

## 1.24 处理中断 OB (OB88)

### 描述

CPU 操作系统在执行程序块被中止后调用 OB88。导致此中断的原因可能是：

- 同步错误的嵌套深度过大
- 块调用(I 堆栈)的嵌套深度过大
- 分配本地数据时出错

如果未编程 OB88，而程序块执行被中止，则 CPU 进入 STOP 模式(事件 ID：W#16#4570)。

如果在优先级 28 下中止了程序块执行，则 CPU 进入 STOP 模式。

可借助于 SFC 39 至 42 禁用、延迟和启用处理中断 OB。

### 处理中断 OB 的本地数据

下表给出了处理中断 OB 的临时(TEMP)变量。OB88 的默认名称选择作为变量名称。

变量	数据类型	描述
OB88_EV_CLASS	BYTE	事件等级和 ID：B#16#35
OB88_SW_FLT	BYTE	错误代码 可能值：
		• B#16#71：同步错误的嵌套深度过大
		• B#16#72：同步错误的嵌套深度过大
		• B#16#73：同步错误的嵌套深度过大
		• B#16#74：块调用(I 堆栈)的嵌套深度过大
		• B#16#75：块调用(B 堆栈)的嵌套深度过大
		• B#16#76：分配本地数据时出错
		• B#16#78：未知指令
		• B#16#7A：跳转指令指向到块外部的目标
		请参考您 CPU 的操作列表以确定哪些错误代码适用于您的 CPU。
OB88_PRIORITY	BYTE	• 优先级：28
OB88_OB_NUMBR	BYTE	OB 编号(88)



变量	数据类型	描述
OB88_BLK_TYPE	BYTE	出现错误的块的类型： <ul style="list-style-type: none"> <li>• B#16#88 : OB·</li> <li>• B#16#8C : FC·</li> <li>• B#16#8E : FB·</li> <li>• B#16##00 : 无法确定中断源</li> </ul>
OB88_RESERVED_1	BYTE	保留
OB88_FLT_PRIORITY	BYTE	导致错误的 OB 的优先级
OB88_FLT_OB_NUMBR	BYTE	导致错误的 OB 的编号
OB88_BLK_NUM	WORD	具有导致错误的 MC7 指令的块编号
OB88_PRG_ADDR	WORD	导致错误的 MC7 指令的相对地址
OB88_DATE_TIME	DATE_AND_TIME	OB 调用的日期和 TOD

## 1.25 背景组织块(OB90)

### 描述

使用 STEP 7，可以监视最大扫描周期并确保最小扫描周期。如果包含所有嵌套中断和系统活动在 OB1 的执行时间少于指定的最小扫描周期时间，则操作系统将作出如下反应：

- 调用后台 OB (如果它存在于 CPU 中)。
- 延迟下一次 OB1 启动(如果 OB90 在 CPU 中不存在)。

### 了解后台 OB 的操作

在所有 OB 中，OB90 的优先级最低。任何系统活动和中断都会将其中断(甚至在最小周期到期后由 OB1 中断)，并且只有在尚未达到所选最小扫描周期时间的情况下才会恢复。对此有一个例外，SFC 和 SFB 的执行在 OB90 中启动。这两项以 OB1 的优先级执行，因此不会被 OB1 中断。在此没有 OB90 的时间监视。

在下列情况下，将从第一条指令开始处理 OB90 中的用户程序：

- 暖重启、冷重启或热重启后
- 装载或删除块后(使用 STEP 7)
- 在 RUN 模式下将 OB90 装载到 CPU 之后
- 终止后台循环后

---

### 注意

对于最小扫描周期时间与周期监视时间差别不大的组态，后台 OB 中的 SFC 和 SFB 调用有可能意外地超出周期时间。

---

## 后台 OB 的本地数据

下表介绍了 OB90 的临时(TEMP)变量。变量名为 OB90 的默认名称。

变量	数据类型	描述
OB90_EV_CLASS	BYTE	事件等级和标识符：B#16#11：激活
OB90_STRT_INF	BYTE	<ul style="list-style-type: none"> <li>• B#16#91：暖重启/冷重启/热重启</li> <li>• B#16#92：已删除块</li> <li>• B#16#93：在 RUN 模式下将 OB90 下载到 CPU</li> <li>• B#16#95：终止后台循环</li> </ul>
OB90_PRIORITY	BYTE	优先级：29 (对应于优先级 0.29)
OB90_OB_NUMBR	BYTE	OB 编号(90)
OB90_RESERVED_1	BYTE	保留
OB90_RESERVED_2	BYTE	保留
OB90_RESERVED_3	INT	保留
OB90_RESERVED_4	INT	保留
OB90_RESERVED_5	INT	保留
OB90_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

## 1.26 启动组织块(OB100、OB101 和 OB102)

### 启动类型

以下几种启动类型有所区别

- 热重启(不使用 S7-300 和 S7-400H)
- 暖启动
- 冷启动

下表中列出了启动过程中操作系统将调用哪个 OB。

启动类型	对应的 OB
热启动	OB101
暖启动	OB100
冷启动	OB102

有关启动类型的更多详细信息，请参阅手册"使用 STEP 7 编程"、"使用 STEP 7 配置硬件"和"连接 S7-400H PLC"。

### 启动事件

CPU 按如下所述执行启动：

- 上电后
- 只要将模式选择器由 STOP 切换为 RUN-P 时
- 使用通信功能(编程设备中的菜单命令或者通过调用不同 CPU 上的通信功能块 19 "START"或 21 "RESUME")发出请求后。
- 多值计算的同步
- 在链接之后的 H 系统中(仅适用于待机的 CPU)

根据启动事件、特定的 CPU 及其参数，将调用适当的启动 OB(OB100、OB101 或 OB102)。通过匹配的编程，可针对循环程序进行某些设置(例外：在 H 系统中，当待机 CPU 已链接时，在该待机 CPU 上存在启动，但却不会调用启动 OB)。

## 启动 OB 的本地数据

下表介绍了启动 OB 的临时(TEMP)变量。变量名为 OB100 的默认名称。

变量	类型	描述
OB10x_EV_CLASS	BYTE	事件等级和标识符：B#16#13：激活
OB10x_STRTUP	BYTE	启动请求： <ul style="list-style-type: none"> <li>• B#16#81：手动暖重启</li> <li>• B#16#82：自动暖重启</li> <li>• B#16#83：要求手动热重启</li> <li>• B#16#84：要求自动热重启</li> <li>• B#16#85：要求手动冷重启</li> <li>• B#16#86：要求自动冷重启</li> <li>• B#16#87：主站：要求手动冷重启</li> <li>• B#16#88：主站：要求自动冷重启</li> <li>• B#16#8A：主站：要求手动暖重启</li> <li>• B#16#8B：主站：要求自动暖重启</li> </ul>
OB10x_PRIORITY	BYTE	优先级：27
OB10x_OB_NUMBR	BYTE	OB 编号(100、101 或 102)
OB10x_RESERVED_1	BYTE	保留
OB10x_RESERVED_2	BYTE	保留
OB10x_STOP	WORD	导致 CPU 停止的事件的编号
OB10x_STRT_INFO	DWORD	有关当前启动的辅助信息
OB10x_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

下表给出了变量 OB100\_STR\_INFO 和 OB101\_STR\_INFO。

位号	含义	可能的二进制值	解释
31 - 24	启动信息	0000 xxxx	机架号 0 (仅限 H CPU)
		0100 xxxx	机架号 1 (仅限 H CPU)
		1000 xxxx	机架号 2 (仅限 H CPU)
		0001 xxxx	多值计算(仅限 S7-400)
		0010 xxxx	分段机架中多个 CPU 的操作 (仅限 S7-400)
		xxxx xxx0	预期组态与实际组态之间无任何差异 (仅限 S7-300)
		xxxx xxx1	预期组态与实际组态之间存在差异 (仅限 S7-300)
		xxxx xx0x	预期组态与实际组态之间无任何差异
		xxxx xx1x	预期组态与实际组态之间存在差异
		xxxx x0xx	非 H CPU
		xxxx x1xx	H CPU
		xxxx 0xxx	最后一次上电时用于时间戳的时钟 未由电池供电
		xxxx 1xxx	最后一次上电时用于时间戳的时钟 由电池供电
23 - 16	刚完成启动	0000 0001	多值计算中的暖重启, 未根据参数 分配来更改 CPU 上的设置(仅限 S7-400)
		0000 0011	利用模式选择器触发重新启动 (暖重启)
		0000 0100	经由 MPI 通过命令触发重新启动 (暖重启)
		0000 0101	多值计算中的冷重启, 未根据 参数分配来更改 CPU 中的设置 (仅限 S7-400)
		0000 0011	使用模式选择器触发冷重启
		0000 1000	经由 MPI 通过命令触发冷重启
		0000 1010	多值计算中的热重启, 未根据参数 分配来更改 CPU 上的设置(仅限 S7-400)
		0000 1011	使用模式选择器触发热重启 (仅限 S7-400)
		0000 1100	经由 MPI 通过命令触发热重启 (仅限 S7-400)

位号	含义	可能的二进制值	解释
		0001 0000	在由电池支持的上电后，自动重新启动(暖重启)
		0001 0001	根据参数分配在由电池支持的上电后进行冷重启
		0001 0011	利用模式选择器触发重新启动(暖重启)；最后一次由电池支持上电
		0001 0100	经由 MPI 通过命令触发重新启动(暖重启)；最后一次由电池支持上电
		0010 0000	在经过由电池支持的上电后，自动重新启动(暖重启)(由系统复位存储器)
		0010 0001	在经过由电池支持的上电后，执行冷重启(由系统复位存储器)
		0010 0011	利用模式选择器触发重新启动(暖重启)；最后一次上电非电池支持
		0010 0100	经由 MPI 通过命令触发重新启动(暖重启)；最后一次上电非电池支持
		1010 0000	根据参数分配在经过由电池支持的上电后自动热重启(仅限 S7-400)
15 - 12	自动启动的许可	0000	自动启动非法，请求存储器复位
		0001	自动启动非法，有必要进行参数修改等
		0111	允许自动重新启动(暖重启)
		1111	允许自动重新启动(暖重启/热重启)(仅限 S7-400)
11 - 8	手动启动的许可	0000	自动启动非法，请求存储器复位
		0001	启动非法，有必要进行参数修改等
		0111	允许重新启动(暖重启)
		1111	允许重新启动(暖重启/热重启)(仅限 S7-400)

位号	含义	可能的二进制值	解释
7 - 0	上一次上电时对自动启动的有效干预或设置	0000 0000	无启动
		0000 0001	多值计算中的暖重启, 未根据参数分配来更改 CPU 上的设置 (仅限 S7-400)
		0000 0011	由模式选择器触发的重新启动 (暖重启)
		0000 0100	经由 MPI 通过命令触发重新启动 (暖重启)
		0000 0101	多值计算中的热重启, 未根据参数分配来更改 CPU 上的设置 (仅限 S7-400)
		0000 0111	使用模式选择器触发冷重启
		0000 1000	经由 MPI 通过命令触发冷重启
		0000 1010	多值计算中的热重启, 未根据参数分配来更改 CPU 上的设置 (仅限 S7-400)
		0000 1011	使用模式选择器触发热重启 (仅限 S7-400)
		0000 1100	经由 MPI 通过命令触发热重启 (仅限 S7-400)
		0001 0000	在由电池支持的上电后, 自动重新启动(暖重启)
		0001 0001	根据参数分配在由电池支持的上电后进行冷重启
		0001 0011	利用模式选择器触发重新启动(暖重启); 最后一次由电池支持上电
		0001 0100	经由 MPI 通过命令触发重新启动(暖重启); 最后一次由电池支持上电
		0010 0000	在经过由电池支持的上电后, 自动重新启动(暖重启) (由系统复位存储器)
		0010 0001	根据参数分配在由电池支持的上电后进行冷重启
		0010 0011	利用模式选择器触发重新启动(暖重启); 最后一次上电非电池支持
		0010 0100	经由 MPI 通过命令触发重新启动(暖重启); 最后一次上电非电池支持
		1010 0000	根据参数分配在经过由电池支持的上电后自动热重启(仅限 S7-400)



## 1.27 编程错误组织块(OB121)

### 描述

只要发生同程序处理相关的错误所导致的事件，CPU 的操作系统即调用 OB121。例如，如果用户程序调用了尚未装载到 CPU 中的块，将会调用 OB121。

### 了解编程错误 OB 的操作

执行 OB121 时所处优先级与中断块的优先级相同。

如果 OB121 未编程，则 CPU 将由 RUN 模式改变为 STOP 模式。

S7 提供了下列 SFC，用于在执行程序期间屏蔽和取消屏蔽 OB121 的启动事件：

- SFC36 (MSK\_FLT)：屏蔽特定的错误代码
- SFC37 (DMSK\_FLT)：取消屏蔽由 SFC36 屏蔽的错误代码
- SFC38 (READ\_ERR)：读取错误寄存器

### 编程错误 OB 的本地数据

下表介绍了编程错误 OB 的临时(TEMP)变量。变量名为 OB121 的默认名称。

变量	类型	描述
OB121_EV_CLASS	BYTE	事件等级和标识符：B#16#25
OB121_SW_FLT	BYTE	错误代码： (可能值：B#16#21、B#16#22、B#16#23、B#16#24、B#16#25、B#16#26、B#16#27、B#16#28、B#16#29、B#16#30、B#16#31、B#16#32、B#16#33、B#16#34、B#16#35、B#16#3A、B#16#3C、B#16#3D、B#16#3E、B#16#3F)
OB121_PRIORITY	BYTE	优先级 = 出错 OB 的优先级
OB121_OB_NUMBR	BYTE	OB 编号(121)
OB121_BLK_TYPE	BYTE	错误所在块的类型(使用 S7-300 时，未在此处输入有效值)： B#16#88：OB，B#16#8A：DB，B#16#8C：FC， B#16#8E：FB
OB121_RESERVED_1	BYTE	保留
OB121_FLT_REG	WORD	错误源(取决于错误代码)。 例如： <ul style="list-style-type: none"> <li>• 发生转换错误的寄存器</li> <li>• 不正确的地址(读/写错误)</li> <li>• 不正确的定时器/计数器/块编号</li> <li>• 不正确的存储区</li> </ul>
OB121_BLK_NUM	WORD	具有导致错误的 MC7 命令的块编号(未在此处为 S7-300 输入有效的编号)
OB121_PRG_ADDR	WORD	导致错误的 MC7 命令的相对地址 (未在此处为 S7-300 输入有效值)
OB121_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

取决于错误代码的变量具有如下：

错误代码：	含义
B#16#21： OB121_FLT_REG：	BCD 码转换出错 所涉及寄存器的 ID (W#16#0000：累加器 1)
B#16#22： B#16#23： B#16#28： B#16#29：  OB121_RESERVED_1：	读取数据时发生的区域长度错误 写入数据时发生的区域长度错误 利用位地址不为 0 的指针对字节、字或双字进行读取访问。 利用位地址不为 0 的指针对字节、字或双字进行写访问。 字节地址不正确。可从 OB121_RESERVED_1 读取数据区和访问类型。 <ul style="list-style-type: none"> <li>• 位 7 至 4 访问类型。                             <ul style="list-style-type: none"> <li>- 0: 位访问，</li> <li>- 1: 字节访问，</li> <li>- 2: 字访问，</li> <li>- 3: 双字访问</li> </ul> </li> <li>• 位 3 至 0 存储区：                             <ul style="list-style-type: none"> <li>- 0: I/O 区域</li> <li>- 1: 过程映像输入表</li> <li>- 2: 过程映像输出表</li> <li>- 3: 位存储器</li> <li>- 4: 全局 DB</li> <li>- 5: 背景数据块</li> <li>- 6: 自己的本地数据</li> <li>- 7: 发话人的本地数据</li> </ul> </li> </ul>
B#16#24： B#16#25： OB121_FLT_REG：	读取时发生范围错误 写入时发生范围错误 在低字节中包含非法区域的 ID (自己的本地数据区的 B#16#86)
B#16#26： B#16#27： OB121_FLT_REG：	定时器编号错误 计数器编号错误 编号非法
B#16#30： B#16#31： B#16#32： B#16#33： OB121_FLT_REG：	对写保护全局 DB 的写访问 对写保护背景数据块的写访问 访问全局 DB 时 DB 编号出错 访问背景数据块时 DB 编号出错 DB 编号非法

错误代码：	含义
B#16#34：	调用 FC 时 FC 编号出错
B#16#35：	调用 FB 时 FB 编号出错
B#16#3A：	访问了尚未装载的 DB；DB 编号位于允许的范围内
OB121_FLT_REG：	DB 编号
B#16#3C：	访问了尚未装载的 FC；FC 编号位于允许的范围内
OB121_FLT_REG：	FC 编号
B#16#3D：	访问了不可用的 SFC；SFC 编号位于允许的范围内
OB121_FLT_REG：	SFC 编号
B#16#3E：	访问了尚未装载的 FB；FB 编号位于允许的范围内
OB121_FLT_REG：	FB 编号
B#16#3F：	访问了不可用的 SFB；SFB 编号位于允许的范围内
OB121_FLT_REG：	SFB 编号

## 1.28 I/O 访问错误组织块(OB122)

### 描述

只要在访问模块上的数据时出错，CPU 的操作系统即调用 OB122。例如，如果在访问 I/O 模块上的数据时，CPU 检测到读取错误，操作系统将调用 OB122。

### 了解 I/O 访问错误 OB 的操作

执行 OB122 时所处优先级与中断 OB 的优先级相同。如果 OB122 未编程，则 CPU 将由 RUN 模式改变为 STOP 模式。

S7 提供了下列 SFC，用于在执行程序期间屏蔽和取消屏蔽 OB122 的启动事件：

- SFC36 (MSK\_FLT)：屏蔽特定的错误代码
- SFC37 (DMSK\_FLT)：取消屏蔽由 SFC36 屏蔽的错误代码
- SFC38 (READ\_ERR)：读取错误寄存器

### I/O 访问错误 OB 的本地数据

下表说明了 I/O 访问错误 OB 的临时(TEMP)变量。变量名为 OB122 的默认名称。

变量	类型	描述
OB122_EV_CLASS	BYTE	事件等级和标识符：B#16#29
OB122_SW_FLT	BYTE	错误代码： <ul style="list-style-type: none"> <li>• B#16#42：I/O 访问错误，读取</li> <li>• B#16#43：I/O 访问错误，写入</li> </ul>
OB122_PRIORITY	BYTE	优先级： <ul style="list-style-type: none"> <li>• 出错 OB 的优先级</li> </ul>
OB122_OB_NUMBR	BYTE	OB 编号(122)
OB122_BLK_TYPE	BYTE	出错块的类型(B#16#88：OB， B#16#8C：FC，B#16#8E：FB) (未在此处为 S7-300 输入有效编号)
OB122_MEM_AREA	BYTE	存储器区域和访问类型： <ul style="list-style-type: none"> <li>• 7 到 4 位：访问类型 <ul style="list-style-type: none"> <li>- 0: 位访问</li> <li>- 1: 字节访问</li> <li>- 2: 字访问</li> <li>- 3: 双字访问</li> </ul> </li> <li>• 3 到 0 位：存储区 <ul style="list-style-type: none"> <li>- 0: I/O 区域</li> <li>- 1: 输入的过程映像</li> <li>- 2: 输出的过程映像</li> </ul> </li> </ul>
OB122_MEM_ADDR	WORD	出错的内存地址

变量	类型	描述
OB122_BLK_NUM	WORD	具有导致错误的 MC7 命令的块编号(未在此处为 S7-300 输入有效的编号)
OB122_PRG_ADDR	WORD	导致错误的 MC7 命令的相对地址(未在此处为 S7-300 输入有效编号)
OB122_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME



## 2 SFC 的公共参数

### 2.1 通过输出参数 RET\_VAL 判断错误

#### 错误信息的类型

在用户程序中执行的系统功能(SFC)将指示 CPU 是否能够成功执行 SFC 的功能。您可获取有关已出现的任何错误的信息，这些信息将以下列两种方式显示：

- 在状态字的 BR 位中
- 在输出参数 RET\_VAL (返回值)中

---

#### 注意

在判断某个特定 SFC 的输出参数前，您应始终执行以下步骤：

- 首先，判断状态字的 BR 位。
- 然后，检查输出参数 RET\_VAL。

如果 BR 位指示出现了错误或者如果 RET\_VAL 包含一个常规错误代码，则您不得判断该 SFC 的输出参数！

---

#### 返回值中的错误信息

系统功能(SFC)通过在状态字的二进制结果位(BR)中输入值"0"来指示执行时出现了一个错误。某些系统功能会在称为返回值(RET\_VAL)输出的输出处提供附加错误代码。如果在输出参数 RET\_VAL 中输入一个常规错误(相关解释请参见下文)，则只能通过状态字 BR 位中的值"0"指示。

返回值的数据类型为整型(INT)。返回值与值"0"之间的关系指示出在执行该功能时是否出错。

在 CPU 上执行 SFC	BR	返回值	整数符号
出现错误	0	小于"0"	负 (符号位为"1")
未出现错误	1	大于或等于"0"	正 (符号位为"0")

## 对错误信息的响应

RET\_VAL 中有如下两种不同类型的错误代码：

- 所有系统功能均可输出的常规错误代码以及
- 系统功能可输出的特定错误代码，该代码与其特定功能相关。

您可以编写程序，使其对系统功能执行期间出现的错误作出响应。这样，您便可防止由第一个错误而导致的更多错误的出现。

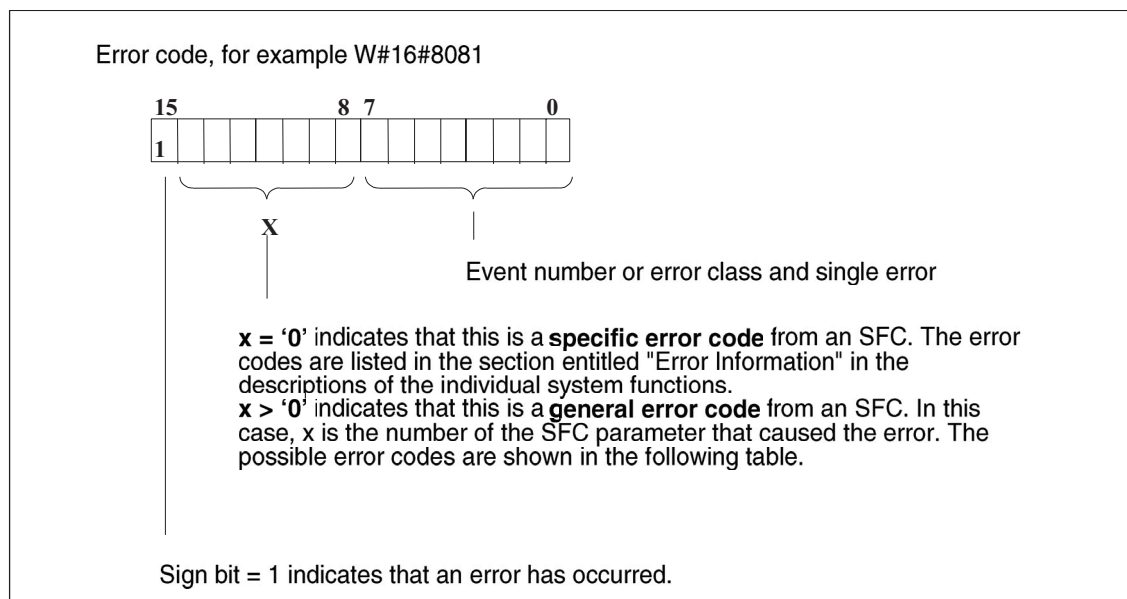
## 常规和特定错误信息

系统功能的返回值(RET\_VAL)将提供下列两种错误代码类型之一：

- 与可在任何系统功能中出现的错误相关的常规错误代码。
- 仅与特定系统功能相关的特定错误代码。

虽然输出参数 RET\_VAL 的数据类型为整型(INT)，但会根据十六进制值对系统功能的错误代码进行编组。如果您要检查某个返回值并将该值与本手册中列出的错误代码进行比较，则请用十六进制格式显示错误代码。

下图给出了十六进制格式的系统功能错误代码结构。



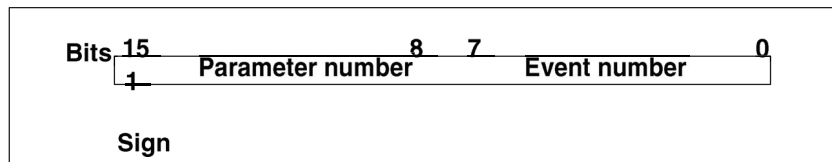


## 常规错误信息

常规错误代码指示可在任何系统功能中出现的错误。常规错误代码由下列两种数字组成：

- 介于 1 到 111 之间的参数编号，其中 1 指示 SFC 的第一个参数，2 指示第二个参数，等等。
- 事件编号 0 到 127。事件编号指示发生的同步错误。

下表列出了常规错误的代码以及每个错误的解释。



### 注意

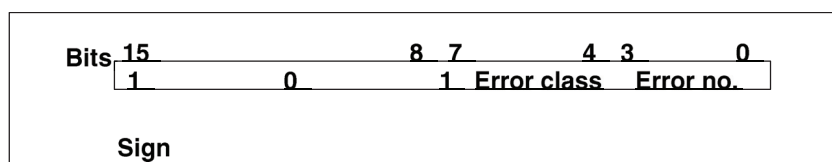
如果在 RET\_VAL 中输入了一个常规错误代码，则可能会出现下列情况：

- 与 SFC 相关的操作可能已开始或者已经完成。
- 可能在执行该操作时已经出现了特定 SFC 错误。但由于之后出现了常规错误而可能不再指示该特定错误。

## 特定错误信息

某些系统功能(SFC)有一个可提供特定错误代码的返回值。此错误代码指示在某一特定系统功能执行期间出现了一个与该功能相关的错误(参见下图)。特定错误代码由下列两种数字组成：

- 介于 0 到 7 之间的错误类别。
- 介于 0 到 15 之间的错误编号。



## 常规错误代码

下表解释了返回值的常规错误代码。该错误代码用十六进制格式显示。每个代码编号中的字母 x 仅仅是一个占位符，它代表引起该错误的系统功能参数的编号。

错误代码 (W#16#...)	解释
8x7F	内部错误 此错误代码指示参数 x 处的内部错误。
8x01	ANY 参数非法语法 ID
8x22	读参数时出现范围长度错误。
8x23	写参数时出现范围长度错误。 该错误代码指示对于 ANY 参数，参数 x 整体或部分位于地址范围之外或位范围长度不是 8 的倍数。
8x24	读参数时发生范围错误。
8x25	写参数时发生范围错误。 此错误代码指示参数 x 位于一个对于系统功能非法的范围之内。有关非法范围的信息，请参见各个功能的描述。
8x24	读参数时发生范围错误。
8x25	写参数时发生范围错误。 此错误代码指示参数 x 位于一个对于系统功能非法的范围之内。有关非法范围的信息，请参见各个功能的描述。
8x26	该参数包含一个过大的定时器编号。 此错误代码指示在参数 x 中指定的定时器不存在。
8x27	该参数包含一个过大的计数器编号(计数器编号错误)。 此错误代码指示在参数 x 中指定的计数器不存在。
8x28	读参数时发生地址对齐错误。
8x29	写参数时发生地址对齐错误。 此错误代码指示参数 x 的参考是一个不等于 0 的位地址。
8x30	该参数位于一个只读的全局 DB 中。
8x31	该参数位于一个只读的实例 DB 中。 此错误代码指示参数 x 位于一个只读数据块中。如果该数据块是用系统功能本身打开的，则系统功能将始终返回值 W#16#8x30。
8x32	该参数包含一个过大的 DB 编号(DB 编号错误)。
8x34	该参数包含一个过大的 FC 编号(FC 编号错误)。
8x35	该参数包含一个过大的 FB 编号(FB 编号错误)。 此错误代码指示参数 x 包含一个大于允许最大编号的块编号。
8x3A	该参数包含一个未装载的 DB 编号。
8x3C	该参数包含一个未装载的 FC 编号。
8x3E	该参数包含一个未装载的 FB 编号。
8x42	在系统尝试从外设输入区读参数时出现访问错误。
8x43	在系统尝试将参数写入外设输出区时出现访问错误。
8x44	在出现一个错误后，在第 n (n > 1)个读访问中出现错误。
8x45	在出现一个错误后，在第 n (n > 1)个写访问中出现错误。 此错误代码指示对所需参数的访问被拒绝。

## 2.2 异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义

### 异步 SFC

异步运行的 SFC 是这样一种 SFC：在完成其功能前，它们将被调用一次以上。下列 SFC 始终异步执行，或者在某些情况下才会异步执行：

- SFC 7 "DP\_PRAL"
- SFC 11 "DPSYC\_FR"
- SFC 12 "D\_ACT\_DP"
- SFC 13 "DPNRM\_DG"
- SFC 51 "RDSYSST"
- SFC 55 "WR\_PARM"
- SFC 56 "WR\_DPARM"
- SFC 57 "PARM\_MOD"
- SFC 58 "WR\_REC"
- SFC 59 "RD\_REC"
- SFC 65 "X\_SEND"
- SFC 67 "X\_GET"
- SFC 68 "X\_PUT"
- SFC 69 "X\_ABORT"
- SFC 72 "I\_GET"
- SFC 73 "I\_PUT"
- SFC 74 "I\_ABORT"
- SFC 82 "CREA\_DBL"
- SFC 83 "READ\_DBL"
- SFC 84 "WRIT\_DBL"
- SFC 90 "H\_CTRL"
- SFC 102 "RD\_DPARA"
- SFC 103 "DP\_TOPOL"
- SFC 114 "PN\_DP"

## 标识作业

如果您用以上所列的某个 SFC 触发一个硬件中断、将控制命令输出到 DP 从站、启动数据传送或者中止未组态连接，然后在当前作业完成之前再次调用同一 SFC，则 SFC 的执行结果取决于第二次调用是否涉及相同的作业。

下表解释了哪些输入参数可为这些 SFC 指定该作业。如果这些参数与一个尚未完成的作业的参数相匹配，则 SFC 调用将被视为后续调用。

SFC	作业的标识符为...
7 "DP_PRAL"	IOID、LADDR
11 "DPSYC_FR"	LADDR, GROUP, MODE
12 "D_ACT_DP"	LADDR
13 "DPNRM_DG"	LADDR
51 "RDSYSST"	SSL_ID, INDEX
55 "WR_PARM"	IOID, LADDR, RECNUM
56 "WR_DPARM"	IOID, LADDR, RECNUM
57 "PARM_MOD"	IOID、LADDR
58 "WR_REC"	IOID, LADDR, RECNUM
59 "RD_REC"	IOID, LADDR, RECNUM
65 "X_SEND"	DEST_ID、REQ_ID
67 "X_GET"	DEST_ID、VAR_ADDR
68 "X_PUT"	DEST_ID、VAR_ADDR
69 "X_ABORT"	DEST_ID
72 "I_GET"	IOID、LADDR、VAR_ADDR
73 "I_PUT"	IOID、LADDR、VAR_ADDR
74 "I_ABORT"	IOID、LADDR
82 "CREA_DBL"	LOW_LIMIT, UP_LIMIT, COUNT, ATTRIB, SRCBLK
83 "READ_DBL"	SRCBLK, DSTBLK
84 "WRIT_DBL"	SRCBLK, DSTBLK
90 "H_CTRL"	MODE, SUBMODE
102 "RD_DPARA"	LADDR, RECNUM
103 "DP_TOPOL"	DP_ID
114 "PN_DP"	-

## 输入参数 REQ

REQ (请求)输入参数只用于启动作业：

- 如果您为一个当前未激活的作业调用 SFC，则该作业将用 REQ = 1 进行启动(情况 1)。
- 如果某特定作业已启动但尚未完成，并且您再次调用 SFC 来执行相同的作业(例如，在一个循环中断 OB 中)，则 SFC 不会判断 REQ (情况 2)。

## 输出参数 RET\_VAL 和 BUSY

作业执行状态由输出参数 RET\_VAL 和 BUSY 指示。

另请参考使用输出参数 RET\_VAL 判断错误中的“注意”。

- 在第 1 种情况中(第一次用 REQ=1 进行调用), 如果系统资源可用且输入参数正确, 则会将 W#16#7001 输入到 RET\_VAL 中。然后将 BUSY 置位。

如果所需要的系统资源当前正在使用中或者输入参数有错误, 则会将相应的错误代码输入到 RET\_VAL 中, 且 BUSY 具有值 0。

- 在第 2 种情况中(在相同的作业已激活时进行调用), 会将 W#16#7002 输入到 RET\_VAL 中(这是一个说明该作业仍在处理中的警告), 并将 BUSY 置位。
- 下列内容适用于作业的最后一次调用:
  - 对于 SFC 13 "DPNRM\_DG"、SFC 67 "X\_GET"和 SFC 72 "I\_GET", 如果未出现任何错误, 则会将提供的数据的数目以正的字节数的形式输入到 RET\_VAL 中。BUSY 值将为 0。如果出错, 则 RET\_VAL 将提供错误信息。这种情况下, 不得使用 BUSY 进行判断。
  - 对于 SFC 59 "RD\_REC", 如果没有出错, 以字节计的数据记录数将被输入到 RET\_VAL 中, 或输入 0 值(参考使用 SFC 59 "RD\_REC"读取数据记录!)。在这种情况下, BUSY 的值为 0。如果出错, 错误代码将被输入到 RET\_VAL 中, BUSY 的值为 0。
  - 对于所有其它 SFC, 如果作业无错执行, 0 将被输入到 RET\_VAL 中, BUSY 的值为 0。如果出错, 错误代码将被输入到 RET\_VAL 中, BUSY 的值为 0。

---

### 注意

如果第一次和最后一次调用一起执行, 则 RET\_VAL 和 BUSY 的执行结果与最后一次调用之描述相同。

---

## 作业占用的资源

每项作业都会占用 CPU 资源。例如, 使用 SFC 59 "RD\_REC"会占用目标范围的内存空间。这些资源以独占方式分配给该作业; 只要 SFC 处于活跃状态, 即只要 BUSY 的值是 1, 便会分配资源。

您无法将被占用的资源用于其他目的或更改它们。例如, 使用 SFC 59 "RD\_REC", 您无法更改目标范围的长度或内容。

## 概述

下表概括了上面所述的关系。特别是，其给出了在已调用 SFC 但未执行完作业的情况下输出参数的可能值。

### 注意

在每次调用之后，您必须评估您程序中相关的输出参数。

在作业执行期间，调用类型、REQ、RET、RET\_VAL 和 BUSY 之间的关系。

调用编号	调用类型	REQ	RET_VAL	BUSY
1	第一次调用	1	W#16#7001	1
			错误代码：	0
2 到(n - 1)	中间调用	无关	W#16#7002	1
N	最后一次调用	无关	如果未出现任何错误，则为 W#16#0000 (例外：SFC 59 "RD_REC" (如果目标区域大于传送的数据记录)及 SFC 13 "DPNRM_DG"、SFC 67 "X_GET"和 SFC 72 "I_GET")	0
			如果出现了错误，则为错误代码	0

## 3 复制功能和块功能

### 3.1 使用 SFC 20 "BLKMOV"复制存储区

#### 描述

使用 SFC 20 "BLKMOV"(块移动)可将存储器区(= 源区域)的内容复制到另一个存储器区(= 目标区域)。

以下为允许的源区域：

- 部分数据块
- 存储位
- 输入的过程映像分区(部分过程映像)
- 输出的过程映像分区(部分过程映像)

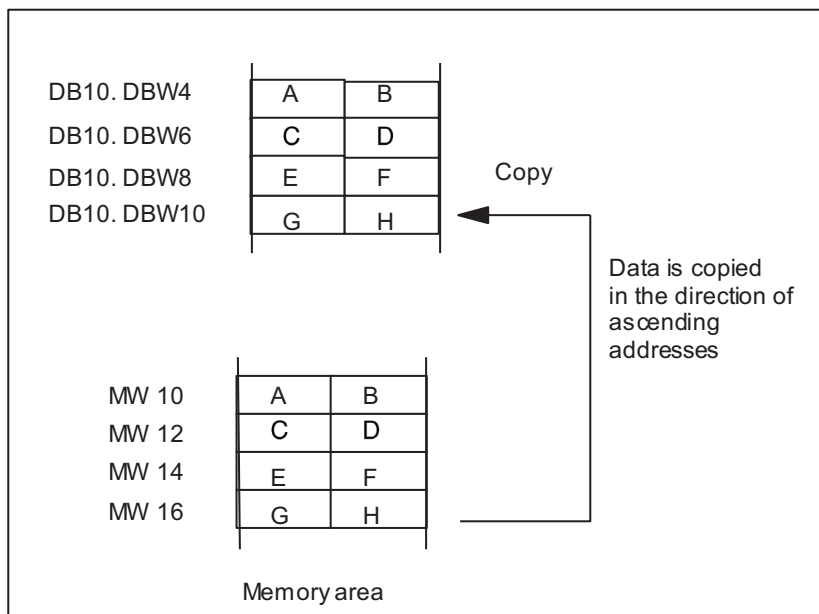
源参数可以是与程序执行无关的装载存储器中数据块(用关键字 UNLINKED 编译的数据块)的一部分。

---

#### 注意

如果 CPU 中包含 SFC 83，则必须使用 SFC 83 读取与程序执行无关的数据块。如果使用 SFC 20，会产生故障信息 W#16#8092。

---



### 源数据和目标数据的一致性

注意，在 SFC20 "BLKMOV"执行期间，源数据应保持不变。否则，无法保证目标数据的一致性。

### 中断功能

只要源区域不是只存在于装载存储器中的数据块的一部分，就没有嵌套深度的限制。

但是，如果在从与程序执行无关的数据块复制时 SFC 20 被中断，则 SFC 20 不能再嵌套执行。

参数	声明	数据类型	存储器区	描述
SRCBLK	INPUT	ANY	I、Q、M、D、L	指定要复制的存储区(源区域)。不允许指定 STRING 类型的数组。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	在功能执行时如有故障发生，此参数值中包含故障代码。
DSTBLK	OUTPUT	ANY	I、Q、M、D、L	指定要将数据复制到的存储区(目标区域)。不允许指定 STRING 类型的数组。

### 源与目标区域的条件

源与目标区域一定不可交叉。如果指定的目标区域大于源区域，则仅将源区域实际的数据量复制到目标区域。

如果指定的目标区域(DSTBLK 参数)小于源区域(SRCBLK 参数)，则仅复制目标区域所能容纳的数据量。

如果实际存在的目标或源区域小于参数(SRCBLK、DSTBLK)中设置的源或目标区域的存储区大小，操作将取决于 CPU 类型：

- S7-300 CPU：无数据传送。RET\_VAL 的值为 W#16#837F。
- 以下适用于 S7-400 V4 CPU (标准 CPU V4、H-CPU V4 和 V4.5)：无数据传送。  
以下适用于所有其它 S7-400CPU：传送数据。  
对于所有的 S7-400 CPU，将在 RET\_VAL 中显示一个错误，错误代码为 W#16#8122 或 W#16#8323。确保始终在用户程序中对这些 RET\_VAL 值进行判断。

如果 ANY 指针(源或目标)的类型为 BOOL，指定的长度必须能够被 8 整除，否则不能进行复制。

如果为 SRCBLK 和 DSTBLK 参数使用 STRING 数据类型的变量，则 STEP 7 将这些变量解析为字节类型的 ARRAY (包括具有最大和当前长度的两个管理字节)。SFC20 中的结果以字节为单位复制：即，复制具有最大长度的字节、具有当前长度的字节和实际字符串。

如果参数 SRCBLK 或 DSTBLK 中的一个或全部两个使用数据类型 STRING，则需要自行设置相应的 ANY 指针。如果源数据位字符串，则最多仅复制字符串中当前所包含的字符。如果源和目标都是字符串，则将目标当前长度作为要复制字符数。

#### 注意

如果使用 SFC20 BLKMOV 将一个非运行相关的 DB 复制到工作存储器，同时重新加载(例如从 PG 上)，则 SFC 可能会延迟数毫秒。这会导致 OB 周期延长，可能会触发扫描时间看门狗。避免在 CPU 正在使用 SFC20 复制块时重新加载该块。



## 错误信息

错误代码 (W#16#...)	解释
0000	无错误
8091	嵌套深度超出。
8092	无法执行 SFC 20 "BLKMOV", 因为访问了不可使用(写保护、不可运行或不可载入)的数据块。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误

## 3.2 使用 SFC 81 "UBLKMOV" 不中断地复制变量

### 描述

使用 SFC 81 "UBLKMOV"(不间断的块移动)，可以将存储区(= 源区域)的内容原封不动地复制到另一个存储区(=目标区域)。此复制操作不能被其它操作系统任务打断。

使用 SFC 81 "UBLKMOV"，可以复制除下列之外的所有存储器区：

- 下列块类型：FB、SFB、FC、SFC、OB、SDB
- 计数器
- 定时器
- 外设 I/O 区存储区
- 与运行无关的块

可以复制最大 512 字节的数据。请注意不同的 CPU 此数值不同；可以参阅指令列表。

源参数也可以加入未链接数据块(用关键字 UNLINKED 编译的数据块)的装载存储器中！

### 中断能力，中断响应时间

复制不可以被中断。注意如果使用 SFC 81 "UBLKMOV"，可以增加 CPU 的中断响应时间。

参数	声明	类型	存储器区	描述
SRCBLK	INPUT	ANY	I、Q、M、D、L	指定要复制的存储区(源区域)。不允许指定 STRING 类型的数组。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	在功能执行时如有故障发生，此参数值中包含故障代码。
DSTBLK	OUTPUT	ANY	I、Q、M、D、L	指定要将数据复制到的存储区(目标区域)。不允许指定 STRING 类型的数组。

## 源与目标区域的条件

源与目标区域一定不可交叉。如果指定的目标区域大于源区域，则仅将源区域实际的数据量复制到目标区域。

如果指定的目标区域小于源区域，则仅复制目标区域能够容纳的数据量。

如果实际存在的目标或源区域小于参数(SRCBLK、DSTBLK)中设置的源或目标区域的存储区大小，操作将取决于 CPU 类型：

- S7-300 CPU：无数据传送。RET\_VAL 的值为 W#16#837F。
- 以下适用于 S7-400 V4 CPU (标准 CPU V4、H-CPU V4 和 V4.5)：无数据传送。  
以下适用于所有其它 S7-400CPU：传送数据。  
对于所有的 S7-400 CPU，将在 RET\_VAL 中显示一个错误，错误代码为 W#16#8122 或 W#16#8323。确保始终在用户程序中对这些 RET\_VAL 值进行判断。

如果 ANY 指针(源或目标)的类型为 BOOL，指定的长度必须能够被 8 整除，否则，不执行 SFC。

如果 ANY 指针为 STRING 类型，则指定长度必须等于 1。

## 错误信息

错误代码 (W#16#...)	描述
0000	无错误
8091	源区域在未链接数据块中。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

### 3.3 使用 SFC 21 "FILL"初始化存储区

#### 描述

使用 SFC 21 "FILL", 可以初始化一个存储区(目标区域), 数据采用另一个存储区(源区域)的内容。此 SFC 拷贝数据直到充满整个存储区。

#### 源与目标区域的条件

源与目标区域一定不可交叉。

如果预分配的目标区域不是输入参数 BVAL 长度的整数倍, 则将写入目标区域直到最后一个字节。

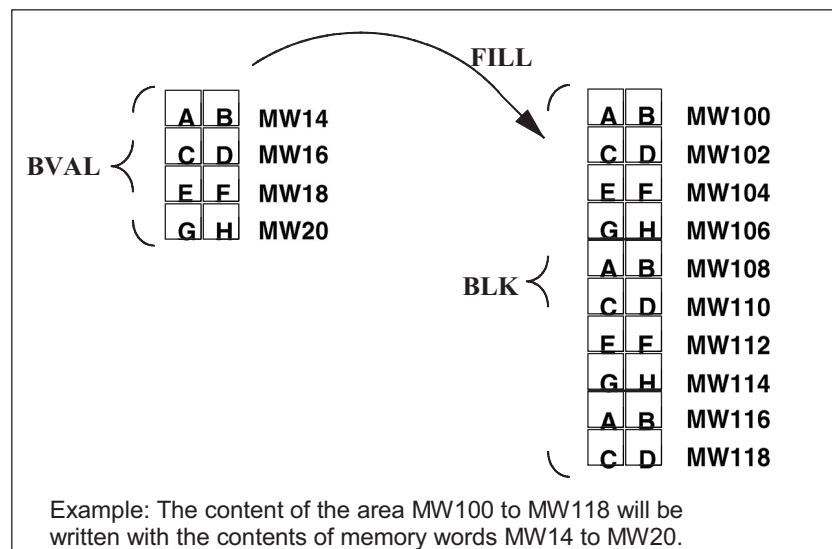
如果指定的目标区域小于源区域, 则仅复制目标区域所能输入的数据量。

如果实际存在的目标或源区域小于参数(BVAL、BLK)中设置的源或目标区域的存储区大小, 操作将取决于 CPU 类型:

- S7-300 CPU: 无数据传送。RET\_VAL 的值为 W#16#837F。
- 以下适用于 S7-400 V4 CPU (标准 CPU V4、H-CPU V4 和 V4.5): 无数据传送。  
以下适用于所有其它 S7-400CPU: 传送数据。  
对于所有的 S7-400 CPU, 将在 RET\_VAL 中显示一个错误, 错误代码为 W#16#8122 或 W#16#8323。确保始终在用户程序中对这些 RET\_VAL 值进行判断。

如果 ANY 指针(源或目标)的类型为 BOOL, 指定的长度必须能够被 8 整除, 否则, 不执行 SFC。

如果目标区域的数据类型为 STRING, 则 SFC21 "FILL"将写入整个字符串, 包括管理信息。



## 源数据和目标数据的一致性

注意，在 SFC21 "FILL"执行期间，源数据应保持不变。否则，无法保证目标数据的一致性。

## 例外

下列项目不能用作源区域：

- 计数器
- 定时器

使用 SFC 21 不能将数据写到下列单元中：

- 下列块类型：FB、SFB、FC、SFC、SDB，
- 计数器、
- 定时器，
- 外设 I/O 存储区。

参数	声明	数据类型	存储器区	描述
BVAL	INPUT	ANY	I、Q、M、D、L	参数 BVAL 包含用于初始化目标区域的区域(源区域)的值或说明。不允许指定 STRING 类型的数组。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	在功能执行时发生故障，则此参数中包含一个故障代码。
BLK	OUTPUT	ANY	I、Q、M、D、L	参数 BLK 包含被初始化的区域(目标区域)的说明。不允许指定 STRING 类型的数组。

### 数据以结构形式输入

如果将一个结构作为输入参数传送，注意以下事项：

STEP 7 总是把一个结构的长度定义为偶数个字节数。因此，如果声明一个包含奇数个字节数的结构，需要另加一个字节的存储空间。

#### 实例

声明一个结构如下：

```
TYP_5_BYTE_STRUCTURE : STRUCT
    BYTE_1_2 : WORD
    BYTE_3_4 : WORD
    BYTE_5 : BYTE
END_STRUCT
```

声明的结构"TYP\_5\_BYTE\_STRUCTURE"需要 6 字节存储空间。

### 错误信息

SFC 21 "FILL"只提供常规(非特定的)故障信息。请参见通过输出参数 RET\_VAL 判断错误。

### 3.4 使用 SFC 22 "CREAT\_DB"创建数据块

#### 描述

使用 SFC 22 "CREAT\_DB"(创建数据块)，可以创建不包含初始化数据的数据块。同时，该数据块中会包含随机数据。此 SFC 可以创建一个可选择长度的数据块，数据块号在一个特定范围内。SFC 将给数据块分配一个可能的最低编号。如果要创建特定编号的数据块，只需选择将上限和下限指定为同一值的范围。在用户程序中已存在的数据块编号不可以再使用。数据块的长度必须为偶数字节。

#### 中断功能

SFC 22 "CREAT\_DB"可被更高优先级的组织块中断。如果 SFC 22 "CREAT\_DB"再次被更高优先级的组织块调用，则出现 W#16#8091 故障。

参数	声明	数据类型	存储器区	描述
LOW_LIMIT	INPUT	WORD	I、Q、M、D、L、 常数	下限值是可分配给数据块的编号范围内的最小编号。
UP_LIMIT	INPUT	WORD	I、Q、M、D、L、 常数	上限值是可分配给数据块号的最大编号。
COUNT	INPUT	WORD	I、Q、M、D、L、 常数	计数值指定要为数据块保留的数据字节数。此处必须指定偶数个字节(最大 65534)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在执行功能时出错，则返回值将包含故障代码。
DB_NUMBER	OUTPUT	WORD	I、Q、M、D、L	创建的数据块的编号。如果出错，(置位 RET_VAL 的位 15)在 DB_NUMBER 中输入数值 0。

## 错误信息

故障代码 (W#16#...)	解释
0000	未产生故障。
8091	调用了嵌套的 SFC 22。
8092	当前不能执行"创建数据块"功能，可能因为 <ul style="list-style-type: none"> <li>• "压缩用户存储器"功能正在执行</li> <li>• CPU 中的数据块数量已经达到最大数量。</li> <li>• H CPU 正在执行链接或者升级功能。</li> <li>• 安装了 WinAC 软件的 CPU 探测到一个计算机操作系统故障。</li> <li>• 先前的删除操作尚未完成。</li> </ul>
80A1	数据块编号出错： <ul style="list-style-type: none"> <li>• 编号为 0。</li> <li>• 编号超出本 CPU 规定的数据库编号值。</li> <li>• 参数下限 &gt; 上限。</li> </ul>
80A2	数据块长度出错： <ul style="list-style-type: none"> <li>• 长度为 0。</li> <li>• 长度指定为奇数。</li> <li>• 长度超出 CPU 中的允许值。</li> </ul>
80B1	没有可用的数据块编号。
80B2	存储空间不够。
80B3	连续存储空间不够(纠正方法：压缩存储器！)
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误



## 3.5 使用 SFC 23 "DEL\_DB"删除数据块

### 描述

使用 SFC 23 "DEL\_DB"(删除数据块)，可删除存在于 CPU 的工作存储器以及装载存储器(如果存在)中的数据块。此数据块必须没有在当前或任何更低的优先级中打开，换言之，此数据一定不是位于两个数据块寄存器中的任意一个或 B 堆栈中。否则，CPU 在调用 SFC 23 时将启动 OB°121。如果 OB°121 不存在，则 CPU 停机。对于 S7-300(例外：CPU 318)，删除数据块时不调用 OB121。

### 注意

不建议使用 SFC 23 "DEL\_DB"删除实例 DB。此操作总会导致程序错误。请避免使用 SFC 23 删除实例 DB！

下表说明何时可以使用 SFC 23 "DEL\_DB"删除数据块。

如果...	那么...
数据块是由 SFC 22 "CREAT_DB"创建的，	SFC 23 可以删除。
DB 通过 STEP 7 传送给 CPU，不是以关键词 UNLINKED 生成的，	SFC 23 可以删除。
数据块是用关键字 UNLINKED 创建的。	<ul style="list-style-type: none"> <li>• SFC 23 可以在 S7-300 上删除它</li> <li>• SFC 23 无法在 S7-400 上删除它</li> </ul>
DB 是通过 SFC 82 "CREA_DBL"调用创建的	SFC 23 可以删除。
DB 保存在闪存卡上。	SFC 23 不能删除。

### 中断功能

SFC 23 "DEL\_DB"可被具有更高优先级的优先级中断。此时如再次调用此 SFC，则调用失败并且故障 W#16#8091 写入参数 RET\_VAL 中。

参数	声明	数据类型	存储器区	描述
DB_NUMBER	INPUT	WORD	I、Q、M、D、L、常数	要删除的数据块的编号
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息

## 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8091	SFC 23 被嵌套调用，且超出了 CPU 的最大嵌套深度。
8092	当前不能执行"删除数据块"功能，可能因为 <ul style="list-style-type: none"> <li>• 当前激活了"压缩用户存储器"功能。</li> <li>• 正在上传将要删除的数据块。</li> <li>• H CPU 正在执行链接或者升级功能。</li> <li>• 在装有 WinAC 软件的计算机操作系统中 CPU 检测到一个故障</li> </ul>
80A1	输入参数 DB_NUMBER 出错：实际选择的参数 <ul style="list-style-type: none"> <li>• 为 0。</li> <li>• 大于所用 CPU 允许的数据块编号最大值。</li> </ul>
80B1	带有特定编号的 DB 不在 CPU 的工作存储器中。
80B2	指定编号的数据块是由关键字 UNLINKED 创建的。 (仅限 S7-400)
80B3	数据块位于闪存卡上。
80B4	数据块不能被删除。可能原因： <ul style="list-style-type: none"> <li>• 该数据块属于 F 程序。</li> <li>• 该数据块是用于 S7 通讯的块的实例 DB(仅限 S7-400)。</li> <li>• 该数据块是技术数据块。</li> </ul>
80C1	因为暂时资源紧张，当前不能执行"删除数据块"功能。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 3.6 使用 SFC 24 "TEST\_DB"测试数据块

### 描述

对于 S7-300，SFC24 "TEST\_DB" (测试数据块)提供关于 CPU 工作和装载存储器中数据块的信息；对于 S7-400，它提供关于 CPU 工作存储器的数据块的信息。SFC 查询所选 DB 中的数据字节数，检查 DB 是否写保护。

参数	声明	数据类型	存储器区	描述
DB_NUMBER	INPUT	WORD	I、Q、M、D、L、 常数	被测试的数据块编号
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
DB_LENGTH	OUTPUT	WORD	I、Q、M、D、L	所选数据块包含的数据字节数。
WRITE_PROT	OUTPUT	BOOL	I、Q、M、D、L	关于该数据块写保护标识符的信息 (1 表示只读)。

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
80A1	输入参数 DB_NUMBER 出错：实际选择的参数 <ul style="list-style-type: none"> <li>• 为 0</li> <li>• 大于所用 CPU 允许的最大数据块编号。</li> </ul>
80B1	指定编号的数据块编号在 CPU 中不存在。
80B2	数据块是用关键字 UNLINKED 创建的。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 3.7 使用 SFC 25"COMPRESS"压缩用户存储器

### 存储器中的空隙

当数据块被删除或重新装载几次后，在装载存储器和工作存储器中就会产生空隙。这些空隙会减少存储区的可用空间。

### 描述

使用 SFC 25 "COMPRESS"可以压缩装载存储器和工作存储器的 RAM 区。压缩功能的启用与在 RUN-P 模式(模式选择)下外部启动 CPU 类似。

如果压缩功能由外部启动且一直激活，则调用 SFC 25 将产生一个故障信息。

参数	声明	数据类型	存储器区	描述
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
BUSY	OUTPUT	BOOL	I、Q、M、D、L	指示调用 SFC 25 开始的压缩功能是否仍在执行。(1 表示仍在执行。)
DONE	OUTPUT	BOOL	I、Q、M、D、L	指示调用 SFC 25 开始的压缩功能是否已经成功完成。(1 表示成功完成。)

### 检查压缩功能

一旦调用 SFC 25 "COMPRESS"，压缩功能就开始执行了。但是，不能检查是否已成功压缩存储器空间。

如果要检查压缩功能，需遵循以下步骤：

循环调用 SFC 25。每次调用后先查看参数 RET\_VAL。如果其值为 0，则可查看参数 BUSY 和 DONE。如果 BUSY = 1 且 DONE = 0，则表示压缩功能仍在进行。当 BUSY 值变为 0 且 DONE 的值变为 1 时，表示压缩功能已成功完成。此后再调用 SFC 25，压缩功能会再次启动。

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。压缩功能已通过调用 SFC 25 启动。 在这种情况下，通过检查用户程序中的输出参数 BUSY 和 DONE(见上文)仅能提供有用的信息。
8091	压缩功能由外部启动而且仍在执行。
8092	当前不能执行"压缩用户存储器"功能，因为 <ul style="list-style-type: none"> <li>• "删除数据块"功能由 STEP 7 从外部启动而且仍在执行</li> <li>• 测试和启动功能当前需要一个特定的块(例如，状态)</li> <li>• "复制块"功能已由外部触发而且仍在执行。</li> <li>• H-CPU 正在执行链接或升级功能。</li> </ul>
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 3.8 使用 SFC 44 "REPL\_VAL" 将替换值传送到累加器 1 中

### 描述

使用 SFC 44 "REPL\_VAL"(替换值), 可以传送一个值到累加器 1 中, 这个值存在于可以引发故障的优先级中。

### 限定：只能是位于同步故障组织块中

只能在同步故障组织块(OB121、OB122)中调用 SFC 44 "REPL\_VAL"。

### 应用示例

如果一个输入模块损坏到不能读取任何值的程度, 则每次扫描此模块时 OB 122 都会启动。使用 SFC 44 "REPL\_VAL", 可将 OB 122 中适当的值传送到中断优先级的累加器 1 中, 这样程序就可以使用此替换值。选择此替换值的信息(例如, 出错的块或受到影响的地址)位于 OB 122 的局部变量中。

参数	声明	数据类型	存储器区	描述
VAL	INPUT	DWORD	I、Q、M、D、L、常数	替换值
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在执行功能时出错, 则返回值将包含故障代码。

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。替换值被输入。
8080	SFC 44 不是被同步故障组织块 (OB 121、OB 122)调用。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误

## 3.9 使用 SFC 82 "CREA\_DBL"在装载存储器中生成数据块

### 描述

使用 SFC 82 "CREA\_DBL"(在装载存储器中创建数据块),可在装载存储器(微型存储卡)中创建新的数据块。SFC 82 使用指定范围中的一个编号生成一个缺省值大小的数据块。该数据块的编号为指定范围内的最小可能值。可以通过分配指定编号范围的上限和下限值来生成一个特定编号的数据块。不能把已在用户程序中存在的数据块编号分配给新生成的数据块。如果相同编号的数据块已存在于工作存储器和/或装载存储器中,或者该数据块为复制得到的,则 SFC 终止运行,并生成一条故障消息。

---

### 注意

使用 SFC 24 "TEST\_DB",可以确定相同编号的数据块是否已经存在。

---

参数 SRCBLK (源块)指向的数据区的内容被写入到数据块中。此数据区必须是数据块或数据块中的一个区域。为了保持一致,当 SFC 82 被执行时(即,只要 BUSY 参数值为 TRUE 时)不能改变此数据区。

具有 READ\_ONLY 属性的数据块只能由 SFC 82 创建和初始化。

SFC 82 不改变用户程序的校验和。

## 工作原理

SFC 82 "CREA\_DBL"异步运行，也就是说，包含多次 SFC 调用。在 REQ = 1 时，调用 SFC 82 启动创建数据块功能。

作业状态可通过输出参数 RET\_VAL 和 BUSY 显示。

请参见 REQ、RET\_VAL 和 BUSY 对于异步工作的 SFC 的意义

参数	声明	数据类型	存储区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	REQ=1：请求生成数据块
LOW_LIMIT	INPUT	WORD	I、Q、M、D、L	SFC 分配给数据块编号的范围的下限值
UP_LIMIT	INPUT	WORD	I、Q、M、D、L	SFC 分配给数据块编号的范围的上限值
COUNT	INPUT	WORD	I、Q、M、D、L	计数值指定了数据块的字节数。此处，必须指定偶数个字节。
ATTRIB	INPUT	BYTE	I、Q、M、D、L	数据块的属性：
				位 0 = 1：UNLINKED： 该数据块只存在于装载存储器中。
				位 1 = 1：READ_ONLY： 该数据块具有写保护。
				位 2 = 1：NON_RETAIN： 该数据块没有掉电保持。
				位 3 至 7：保留
SRCBLK	INPUT	ANY	D	初始化当前数据块的数据区的指针
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：工作还未结束。
DB_NUM	OUTPUT	WORD	I、Q、M、D、L	生成的数据块的编号

## 错误信息

错误代码 (W#16#...):	描述
0000	无错误
0081	目标区域大于源区域。 将源区域完全写入目标区域。剩余空间用 0 填满。
7000	首次调用 REQ=0：没有数据传输；BUSY 的值为 0。
7001	首次调用 REQ=1：没有数据传输；BUSY 的值为 1。
7002	中间调用(与 REQ 无关)：数据传送已经激活；BUSY 的值为 1。
8081	源区域大于目标区域。 <b>目标区域充满。忽略源区域的其余字节。</b>
8091	调用了嵌套的 SFC 82。
8092	"生成数据块"功能当前无法执行因为 <ul style="list-style-type: none"> <li>• "压缩应用存储器"操作正在进行</li> <li>• 已经达到 CPU 的最大块数量。</li> </ul>
8093	没有数据块或者一个不在工作存储器中的数据块指定给了参数 SRCBLK。
8094	一个不被支持的属性被指定给参数 ATTRIB
80A1	数据块编号故障： <ul style="list-style-type: none"> <li>• 编号为 0</li> <li>• 下限 &gt; 上限</li> </ul>
80A2	数据块长度故障： <ul style="list-style-type: none"> <li>• 长度为 0</li> <li>• 长度为奇数</li> <li>• 长度超出 CPU 允许的值</li> </ul>
80B1	没有可用的数据块编号
80B2	存储器不够用
80BB	装载存储器不够用
80C0	目标区域正在被另外一个 SFC 或通讯操作占用。
80C3	当前达到了可同时激活的 SFC 82 的最大数量。
8xyy	一般故障的代码，例如： <ul style="list-style-type: none"> <li>• 源数据块不存在，或为复制版</li> <li>• 数据块所在的源区域不存在</li> </ul> 请参见通过输出参数 RET_VAL 判断错误



## 3.10 使用 SFC 83 "READ\_DBL"从装载存储器的数据块中读取数据

### 描述

使用 SFC 83"READ\_DBL"(读取装载存储器中的数据块)可将装载存储器(微型存储卡)中的数据块或数据块的一个数据区复制到目标数据块的数据区中。目标数据块必须与运行有关；即不能用属性 UNLINKED 创建。在复制期间，装载存储器的内容不变。

为了保持一致，在 SFC83 执行时(即只要 BUSY 参数值为 TRUE 时)切勿改变目标区域。

下列限制条件适用于参数 SRCBLK(源块)和 DSTBLK(目标块)：

- 对于 BOOLEAN 类型的 ANY 指针，长度必须可被 8 整除。
- 对于 BOOLEAN 类型的 ANY 指针，长度必须可被 1 整除。

如果需要，可以使用 SFC 24 "TEST\_DB"确定源区域的长度。

---

### 注意

SFC 83 异步执行。所以不适合从装载存储器中频繁(或反复)读取变量。

---

---

### 注意

一旦开始，作业就必须被完成。如果达到了同时激活的 SFC 83 的最大数量，而此时具有更高优先级的优先级也调用 SFC 83，将返回故障代码 W#16#80C3。因此，并不马上重启高优先级的任务。

---

## 工作方法

SFC 83 "READ\_DBL"是异步运行的，也就是说，处理过程包含此 SFC 的多次调用。REQ = 1 时调用 SFC 83 功能启动。

作业状态可通过输出参数 RET\_VAL 和 BUSY 显示。

参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义

参数	声明	数据类型	存储区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	REQ=1：读请求
SRCBLK	INPUT	ANY	D	装载存储器中读取的数据块的数据区的指针
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：读操作尚未结束。
DSTBLK	OUTPUT	ANY	D	指向要写入的工作存储器数据块的指针

## 错误信息

错误代码 (W#16#...):	描述
0000	无错误
8081	源区域大于目标区域。 目标区域充满。忽略源区域的其余字节。
7000	首次调用 REQ=0：没有数据传输；BUSY 的值为 0。
7001	首次调用 REQ=1：没有数据传输；BUSY 的值为 1。
7002	中间调用(与 REQ 无关)：数据传送已经激活；BUSY 的值为 1。
0081	目标区域大于源区域。 <b>将源区域完全写入目标区域。目标区域的其余字节不变。</b>
80C0	目标区域正在被另外一个 SFC 或通讯操作占用。
8093	没有任何数据块或者一个不存在于工作存储器中的数据块指定给了参数 DSTBLK。
80B1	没有任何数据块或者一个不是装载存储器中对象(例如，由 SFC 22 生成的数据块)指定给了参数 SRCBLK。
80B4	不能读取具有 F 属性的数据块。
80C0	目标数据块正在被另外一个 SFC 或通讯操作占用。
80C3	当前达到了同时激活的 SFC 83 的最大数目。
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误

## 3.11 使用 SFC 84 "WRIT\_DBL"写数据到装载存储器中的数据块

### 描述

使用 SFC 84 "WRIT\_DBL" (在装载存储器中写数据块), 可以将工作存储器中某一数据块或数据块区域中的内容传送到"微存储卡"中的数据块或数据块的数据区中。此源数据必须与过程相关; 即它一定不是用属性 UNLINKED 创建的。因此, 它也可以使用 SFC 22 "CREAT\_DB"生成。

为了保持一致, 在 SFC 84 执行时一定不要改变源区域(也就是说, 只要 BUSY 参数的值为 TRUE)。

下列限制条件适用于参数 SRCBLK(源块)和 DSTBLK(目标块):

- 对于 BOOLEAN 类型的 ANY 指针, 长度必须可被 8 整除。
- 对于 STRING 类型的 ANY 指针, 则长度必须等于 1。

如果需要, 可以使用 SFC 24 "TEST\_DB"确定目标数据块的长度。

如果数据块是通过此 SFC 生成的, SFC 84 不改变用户程序的校验和。但是第一次进入此数据块写数据时就改变用户程序的校验和。

### 注意

SFC 84 不适合向装载存储器频繁(或循环)写入数据。请注意由于技术原因进入微存储卡的写操作有一定的局限性。有关详细信息, 请参见参考手册 "SIMATIC S7-300 可编程控制器 CPU 数据: CPU 31xC 和 CPU 31x"。

### 工作原理

SFC 84 "WRIT\_DBL"异步运行, 也就是说, 处理过程包含此 SFC 的多次调用。REQ = 1 时调用 SFC 84 功能启动。

作业状态可通过输出参数 RET\_VAL 和 BUSY 显示。

请参见 REQ、RET\_VAL 和 BUSY 对于异步工作的 SFC 的意义

参数	声明	数据类型	存储区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	REQ=1: 写请求
SRCBLK	INPUT	ANY	D	指向要从中读取内容的工作存储器中数据块的指针
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1: 写操作尚未结束。
DSTBLK	OUTPUT	ANY	D	指向要向其中写入的装载存储器中数据块的指针

## 错误信息

错误代码 (W#16#...):	描述
0000	无故障
0081	目标区域大于源区域。 将源区域完全写入目标区域。目标区域的其余字节不变。
7000	首次调用 REQ=0：没有数据传输；BUSY 的值为 0。
7001	首次调用 REQ=1：没有数据传输；BUSY 的值为 1。
7002	中间调用(与 REQ 无关)：数据传送已经激活；BUSY 的值为 1。
8081	源区域大于目标区域。 <b>目标区域充满。忽略源区域的其余字节。</b>
8092	不正确的工作模式：当 SFC 84 处于激活状态时，CPU 进入 STOP 模式。在下次切换到 RUN 的位置会提供此故障代码。再次调用 SFC 84。
8093	没有数据块或者一个不在工作存储器中的数据块指定给了参数 SRCBLK。
80B1	没有任何数据块或者不是装载存储器中的对象的数据块分配给参数 DSTBLK (例如，由 SFC 22 生成的数据块)。
80B4	不得更改具有 F 属性的数据块
80C0	目标区域正在被另外一个 SFC 或通讯操作占用。实例：您从 CPU 向编程设备(PG)上传数据块。您要使用 SFC 84 更改此数据块的内容。
80C3	当前达到了同时激活的 SFC 84 的最大数目。
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误

## 3.12 使用 SFC 85 "CREA\_DB" 创建数据块

### 描述

使用 SFC 85 "CREA\_DB" (创建数据块), 可在用户程序中创建不带任何缺省值的数据块。相反, 此数据块包含随机数据。SFC 将以指定区域中的编号和缺省大小创建一个数据块。SFC 给该数据块分配一个可能的最低编号。要创建具有特定编号的数据块, 请为缺省区域的上限和下限输入同样的编号。用户程序中已经存在的数据块编号不可以再使用。数据块的长度必须是偶数个字节。

根据为 ATTRIB 参数所做的选择, 创建的数据块具有属性 RETAIN 或 NON\_RETAIN :

- RETAIN 表示数据块创建于工作存储器的掉电保持区域。也就是说数据块的当前值在每次断电/通电转换以及每次重新启动(热启动)之后会被保留。
- NON\_RETAIN 表示数据块创建于工作存储器的非掉电保持区域。也就是说数据块的当前值在每次断电/通电转换以及每次重新启动(热启动)之后不确定。

如果不区分掉电保持与非掉电保持工作存储器, 则 ATTRIB 参数会被忽略。也就是说数据块的值在每次断电/通电转换以及每次重新启动(热启动)之后会被保留。

### 中断功能

SFC 85 "CREA\_DB" 可由更高优先级的组织块中断。如果 SFC 85 "CREA\_DB" 再次被更高优先级的组织块调用, 则该调用将被拒绝, 并出现 W#16#8091 故障。

参数	声明	数据类型	存储区	描述
LOW_LIMIT	INPUT	WORD	I、Q、M、D、L 常数	下限值是可分配给数据块号的最小编号。
UP_LIMIT	INPUT	WORD	I、Q、M、D、L 常数	上限值是可分配给数据块号的最大编号。
COUNT	INPUT	WORD	I、Q、M、D、L 常数	计数值为数据块的数据字节数。必须指定偶数个字节(最大 65534)。
ATTRIB	INPUT	BYTE	I、Q、M、D、L 常数	数据块属性 : <ul style="list-style-type: none"> <li>• B#16#00 : RETAIN</li> <li>• B#16#04 : NON_RETAIN</li> </ul>
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果功能执行时出错, 则返回值包含故障代码。
DB_NUMBER	OUTPUT	WORD	I、Q、M、D、L	创建的数据块的编号。在出错的情况下, (RET_VAL 的第 15 位置 1) DB_NUMBER 置 0。

## 错误信息

错误代码 (W#16#...)	描述
0000	无错误
8091	调用了嵌套的 SFC 85。
8092	"创建数据块"功能当前不能被执行，因为 <ul style="list-style-type: none"> <li>• "压缩用户存储区"功能正在执行</li> <li>• 安装了 WinAC 软件的 CPU 探测到一个计算机操作系统故障。</li> </ul>
8094	ATTRIB 中的值无效
80A1	数据块编号错误： <ul style="list-style-type: none"> <li>• 编号为 0</li> <li>• 编号超出本 CPU 规定的数据块的编号值</li> <li>• 下限 &gt; 上限</li> </ul>
80A2	数据块长度错误： <ul style="list-style-type: none"> <li>• 长度为 0</li> <li>• 指定了一个奇数字节长度</li> <li>• 长度超出 CPU 中的允许值</li> </ul>
80B1	没有可用的数据块编号。
80B2	存储空间不够。
80B3	连续存储空间不够(压缩存储器！)
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 4 用于控制程序执行的 SFC

### 4.1 使用 SFC 43 "RE\_TRIGR"再触发循环时间监控

#### 描述

使用 SFC 43 "RE\_TRIGR"(再触发监视狗)，再触发循环时间监控。

#### 参数

SFC 43 "RE\_TRIGR"没有参数。

#### 错误信息

SFC 43 "RE\_TRIGR"没有提供任何故障信息。

### 4.2 使用 SFC 46"STP"将 CPU 切换到 STOP

#### 描述

使用 SFC 46 "STP"(停止)，可使 CPU 进入停机状态。

#### 参数

SFC 46 "STP"没有任何参数。

#### 错误信息

SFC 46 "STP"没有提供任何故障信息。

## 4.3 使用 SFC 47 "WAIT" 延迟执行用户程序

### 描述

使用 SFC 47 "WAIT", 可在用户程序中设定延迟或等待时间。最长可设定 32767  $\mu$ s. 最短时间依 CPU 型号而定, 并与 SFC 47 的执行时间相同。

### 中断能力

SFC 47 "WAIT" 可被优先级更高的组织块 OB 中断。

#### 注意

(仅限 S7-300, 且不包括 CPU 318)

使用 SFC 47 所编写的等待时间是最短时间。由于嵌套优先级的执行时间以及系统负荷, 该时间会有所延长。

参数	声明	数据类型	存储器区	描述
WT	INPUT	INT	I、Q、M、D、L、常数	参数 WT 包含以 $\mu$ s 为单位的等待时间。

### 错误信息

SFC 47 "WAIT" 没有提供任何故障信息。



## 4.4 使用 SFC 35"MP\_ALM"触发一个多值计算中断

### 描述

调用 SFC 35 "MP\_ALM"将触发多处理器中断功能。这时同步启动所有 CPU 的 OB 60 程序块。某机架上的单一处理器操作模式下，它的 OB 60 只会被 SFC 35 启动。

可以使用 JOB 输入参数指明多处理器中断的原因。此工作标识符同时传送到所有的 CPU，再由多处理器中断程序(OB 60)识别(请参见"STEP 7 编程"在线文档)。

可在用户程序中任何地方调用 SFC 35 "MP\_ALM"。但是只能在 RUN 状态下才有效，如果在 STARTUP 状态调用，多处理器中断功能无效。在这种情况下函数值将提示您。

参数	声明	数据类型	存储器区	描述
JOB	INPUT	BYTE	I、Q、M、D、L、常数	作业标识符：可能的值：1 到 15
RET_VAL	OUTPUT	INT	I、Q、M、D、L	功能执行时出错，返回值将包含故障代码。

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	输入参数 JOB 中的值非法。
80A0	OB60 执行的时候本 CPU 或下一个 CPU 中的下一个中断没有完成。
80A1	状态出错(STARTUP 状态代替了 RUN)状态。
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误

## 4.5 使用 SFC 104 "CiR" 控制 CiR

### 描述

RUN 状态下用 SFC 104 "CiR" 执行再组态功能：

- 可以完全禁用 CiR。此时，会始终拒绝将修改的组态从 PG 下载到 CPU。在使用 SFC 104 "CiR" 进行复位之前，会维持锁定状态。
- 可以设定 CiR 同步到 CiR 禁止的时间上限值。在这种情况下，仅当 CPU 用于评估已修改组态所需的时间小于此指定时间时，修改后的组态才能从 PG 下载到 CPU。
- 可以确定是否使能 CiR。如果使能或者条件使能，参数 A\_FT 显示 CiR 同步时间值的实际上限值。

### 注意

在 CiR 同步时输出被禁止，输入值无效。

### 参数

参数	声明	数据类型	存储区	描述
MODE	INPUT	BYTE	I、Q、M、D、L、 常数	作业 ID 号 可能的值： <ul style="list-style-type: none"> <li>• 0: 信息功能</li> <li>• 1: 使能 CiR 功能(在此，CiR 同步时间上限值已经预置)</li> <li>• 2: 完全禁用 CiR</li> <li>• 3: 有条件禁用 CiR。参数 FRZ_TIME 设定 CiR 同步时间上限值。</li> </ul>
FRZ_TIME	INPUT	TIME	I、Q、M、D、L、 常数	"冻结时间" CiR 同步时间上限值(单位为 ms) 允许的取值范围：60 ... 2500 ms (默认值：1000 ms) 注意：只有 MODE=3 时 FRZ_TIME 才有效。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果运行时出错，则返回值将包含故障代码。 当 MODE=0 时，RET_VAL 显示 CiR 是否使能。
A_FT	OUTPUT	TIME	I、Q、M、D、L	当前 CiR 同步时间的上限值

## 错误信息

错误代码 (W#16#...)	解释
0000	作业执行过程中无故障。(此故障代码仅当 MODE=1 或 MODE=2 或 MODE=3 时才有效。)
0001	CiR 使能。(此故障代码仅当 MODE=0 时才有效。)
0002	CiR 已完全被禁用。(此故障代码仅当 MODE=0 时才有效。)
0003	CiR 被有条件禁用。(此故障代码仅当 MODE=0 时才有效。)
8001	CPU 没有准备 CiR 操作。您正在使用 H 系统(单一操作)中的 H CPU，或者正在多处理器模式下使用标准 CPU。
8002	MODE 中的值非法
8003	FRZ_TIME 中的值非法
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误

## SFC 104 的应用示例

例如，SFC 104 "CiR"可用于确保没有初始化 CiR 操作，在此过程中要求 CPU 处于最高性能或适当的状态。

为此，请在增加处理过程之前执行下列 CPU 程序段：

- 通过 MODE = 2 调用 SFC 104(完全禁用 CiR 操作)

在增加处理过程之后，执行下列 CPU 程序段：

- 通过 MODE = 1 调用 SFC 104(重新启用 CiR 操作)或者通过 MODE = 3 调用 SFC 104(有条件禁用 CiR)

## 4.6 使用 SFC 109 "PROTECT"激活和取消激活 CPU 访问保护

### 描述

通过 SFC 109，可以在 CPU 上设置以下保护等级：

- 通过 MODE=0 调用 SFC 109：设置保护等级 1  
如果禁用了口令授权，则通过 MODE=0 调用 SFC 109 来启用。
- 通过 MODE=1 调用 SFC 109：通过口令授权设置保护等级 2。这表示具有有效口令的用户可以去除由 SFC 109 设置的写保护。  
如果启用了口令授权，则通过 MODE=1 调用 SFC 109 来禁用。
- 通过 MODE=12 调用 SFC 109：设置无口令授权的保护等级 3。这表示如果设置了读写保护，则不能禁用；即使通过有效口令也不能禁用。如果在建立授权连接时通过 MODE=12 调用 SFC 109，则 SFC 109 调用对该连接没有任何影响。

可以从手册《S7-400 自动化系统，CPU 规范》中找到关于保护等级含义的信息。

---

### 注意

通过 SFC 109 "PROTECT"，可以设置比在 STEP 7 "配置硬件"中组态的更低的保护等级。

实例：

已经组态了保护等级 3，然后通过 MODE=12 调用了 SFC 109。这禁用了口令授权。

当通过 MODE=0 或 MODE=1 再次调用 SFC 109 时，保持保护等级 3，并通过一个有效的口令去除读写保护。

---

一旦通过 SSL ID W#16#0232 和索引 W#16#0004 的 SSL 部分列表中的 SFC 109 设置了保护等级 2 或 3，就可以检查该级别是否有效。

---

### 注意

CPU 的当前保护等级显示在 SIMATIC Manager 的"操作模式"对话框中。

---

## 通过 SFC 109 设置的保护等级上的操作状态转换效果

动作	通过 SFC 109 设置保护等级时
<ul style="list-style-type: none"> <li>操作模式切换为 STOP</li> <li>无缓冲上电</li> <li>操作模式切换时存储器复位</li> </ul>	设为保护等级 1 (允许使用所有编程设备功能)
<ul style="list-style-type: none"> <li>冷启动</li> <li>暖启动</li> </ul>	由操作系统设为保护等级 1 (允许使用所有编程设备功能)。如有必要, 可以在程序中调用 SFC 109 将保护等级设为 2 或 3。
带缓冲上电	保持不变
RUN/STARTUP/HOLD -> STOP 的操作状态切换(通过调用 SFC 46, 不带相应错误 OB 的程序错误, 或者通过对设备操作员的操作进行编程来实现)	保持不变
S7-400: 重新启动(通过对设备操作员的操作进行编程或自动上电来实现)	保持不变

## 参数

参数	声明	数据类型	存储区	描述
MODE	INPUT	WORD	I、Q、M、D、L、常数	作业 ID 号 可能的值： <ul style="list-style-type: none"> <li>W#16#0000：设为保护等级 1 (允许使用所有编程设备功能)</li> <li>W#16#0001：设为保护等级 2 (CPU 的程序和组态无法进行更改。CPU 中的程序可通过编程设备进行读取。)</li> <li>W#16#000C：设为无口令授权的保护等级 3 (读写保护)。</li> </ul>
RET_VAL	OUTPUT	INT	I、Q、M、D、L	返回值 如果在功能执行期间发生错误, 返回值将包含错误信息。

## 错误信息

错误代码 (W#16#...)	解释
0000	作业执行过程中无错误。
8090	MODE 参数值无效。
80C3	所需资源当前正在使用。

## 应用实例

- 希望保护自己的 CPU 在 RUN 模式下不进行写访问。应在操作模式切换为 STOP 时解除保护。这可以通过在 OB 100 中使用 MODE = W#16#0001 调用 SFC 109 来实现。这样，即使是在重新启动(热启动)后仍将保持保护等级 2 的设置。
- 根据由 HMI 系统控制的输入值或变量的状态，希望在将程序下载到 CPU 时不必将操作模式切换成 STOP。  
为此，可以通过周期性地查询有关的输入值或变量来实现。通过 MODE = W#16#0000 调用 SFC 109 来获取输入或变量的已定义值。然后便可以将您的程序下载到 CPU 了。下载完毕后，通过 MODE = W#16#0001 调用 SFC 109 来重设保护等级 2。
- 希望独立于输入状态或操作员控制和监视系统强制的变量来设置 CPU 的读写保护。  
为此，可通过周期性地查询受影响的输入或变量来实现。通过 MODE=W#16#000C 调用 SFC 109 来获取输入或变量的已定义值。之后，CPU 具有读写访问保护。如果希望去除这种保护，则通过 MODE = W#16#0000 或 W#16#0001 调用 SFC 109 来恢复原始的保护等级。

## 4.7 用 SFB 73 "RCVREC"接收数据记录

### 描述

I 设备可从更高级别控制器那里接收数据记录。接收过程发生使用 SFB 73 "RCVREC" (接收记录)在应用程序中进行。

SFB 73 具有下列工作模式：

- 检查是否有 I 设备数据接收请求。
- 提供数据记录到输出参数。
- 它将答复发送到更高级别控制器。

MODE 输入参数(见下文)确定 SFB 73 使用哪种模式。

I 设备必须处于 RUN 或 START 模式。

可使用 MLEN 指定要接收的最大字节数。因而，数据记录目标区域至少应设为 MLEN 字节长度。

当接收数据记录后(MODE = 1 或 MODE = 2),NEW 输出参数指示数据记录已被保存到 RECORD 中。仔细检查 RECORD 长度是否足够。输出参数 LEN 包含所接收的数据记录的实际长度(字节)。

对于更高级别控制器的肯定响应，将 CODE1 和 CODE 2 描述为零。如果所接收的数据记录被拒绝，则在 CODE1 中使用错误代码 1，在 CODE2 中使用错误代码 2，来指示对更高级别控制器的否定响应。

---

### 注意

如果 I 设备接收到接收数据记录请求，必须在指定时间段内检测该请求。检测到之后，必须在相同时间段内给更高级别控制器发送答复。否则，I 设备将发生超时错误，这会导致 I 设备的操作系统向更高级别控制器发送否定答复。关于时间段的值，请参考 CPU 规范。

---

如果出错，则输出参数 STATUS 将提供错误信息。

### SFB 73 的操作模式

请使用 MODE 输入参数设置 SFB 73 "RCVREC"的操作模式。下表对此进行了说明。

MODE	含义
0	检查是否有接收数据记录请求 如果 I 设备上有来自更高级别控制器的数据记录，SFB 73 仅写入 NEW、SLOT、INDEX 和 LEN 输出参数。如果使用 MODE = 0 连续多次调用 SFB 73，输出参数始终指向同一个请求。
1	从 I 设备的任意子插槽接收数据记录 如果 I 设备上任意子插槽的 I 设备上有来自更高级别控制器的数据记录，则 SFB 73 将写入输出参数，并将数据记录传送到 RECORD 参数。
2	针对 I 设备的特定子插槽接收数据记录 如果 I 设备上特定子插槽的 I 设备上有来自更高级别控制器的数据记录，则 SFB 73 将写入输出参数，并将数据记录传送到 RECORD 参数。
3	向更高级别控制器发送肯定答复 SFB 73 检查更高级别控制器接收数据记录请求，接受可用的数据记录，向更高级别控制器发送肯定确认。
4	向更高级别控制器发送否定答复 SFB 73 检查更高级别控制器接收数据记录请求，拒绝可用的数据记录，向更高级别控制器发送否定确认。拒绝原因在输入参数 CODE1 和 CODE2 中给出。

#### 注意

数据记录到达后(NEW = 1)，必须调用 SFB 73 两次，确保其完整执行。需要符合下列序列：

- 第一次使用 MODE=1 或 MODE=2 调用
- 第二次使用 MODE=3 或 MODE=4 调用

参数	声明	数据类型	存储区	描述
MODE	INPUT	INT	I、Q、M、D、L、常数	工作模式
F_ID	INPUT	DWORD	I、Q、M、D、L、常数	用于要接收的数据的 I 设备传送区中的子插槽(仅与 MODE=2 有关)(始终将高位字设为零)
MLEN	INPUT	INT	I、Q、M、D、L、常数	要接收数据的最大长度,以字节为单位
CODE1	INPUT	BYTE	I、Q、M、D、L、常数	零(当 MODE=3 时)或错误代码 1 (当 MODE=4 时)
CODE2	INPUT	BYTE	I、Q、M、D、L、常数	零(当 MODE=3 时)或错误代码 2 (当 MODE=4 时)



参数	声明	数据类型	存储区	描述
NEW	OUTPUT	BOOL	I、Q、M、D、L	MODE=0：接收到新的数据记录  MODE=1 或 2：数据记录传送到 RECORD
STATUS	OUTPUT	DWORD	I、Q、M、D、L	错误信息
SLOT	OUTPUT	INT	I、Q、M、D、L、常数	与 F_ID 相同
SUBSLOT	OUTPUT	INT	I、Q、M、D、L、常数	与 F_ID 相同
INDEX	OUTPUT	INT	I、Q、M、D、L、常数	接收到的数据记录的数目
LEN	OUTPUT	INT	I、Q、M、D、L	接收到的数据记录的长度
RECORD	IN_OUT	ANY	I、Q、M、D、L	用于接收到的数据记录的目标区域  注意：S7-300 CPU 的 RECORD 参数总是需要被完整指定的 DB 参数 (例如：P#DB13.DBX0.0 Byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成出错消息。

## 错误信息

关于 STATUS 参数的解释，请参见用 SFB 54 "RALRM"接收中断

## 4.8 用 SFB 74 "PRVREC"提供数据记录

### 描述

I 设备可接收来自更高级别控制器要求提供数据记录的请求。该任务使用 SFB 74 "PRVREC" (提供记录)在应用程序中执行。

SFB 74 具有下列模式：

- 检查是否有为 I 设备提供数据记录的请求。
- 它将所请求的数据记录发送到更高级别控制器。
- 它将答复发送到更高级别控制器。

MODE 输入参数(见下文)确定 SFB 74 使用哪种模式。

I 设备必须处于 RUN 或 START 模式。

可使用 LEN 指定要接收的最大字节数。因而，数据记录目标区域至少应设为 LEN 字节长度。

当有提供数据记录的请求未决时(MODE = 0)，NEW 输出参数将被设为 TRUE。

如果已接受提供数据记录的请求，则向更高级别控制器发送肯定响应，并将所请求的记录写入到 RECORD，CODE1 和 CODE 2 设为零。如果提供数据记录的请求将被拒绝，则在 CODE1 中使用错误代码 1，在 CODE2 中使用错误代码 2，来指示对更高级别控制器的否定响应。

---

### 注意

如果 I 设备接收到提供数据记录的请求，必须在指定时间段内检测该请求。检测到之后，必须在相同时间段内给更高级别控制器发送答复。否则，I 设备将发生超时错误，这会导致 I 设备的操作系统向更高级别控制器发送否定答复。关于时间段的价值，请参考 CPU 规范。

---

如果出错，则输出参数 STATUS 将提供错误信息。

## SFB 74 的操作模式

可使用 MODE 输入参数设置 SFB 74 "PRVREC"的操作模式。下表对此进行了说明。

MODE	含义
0	检查是否有提供数据记录的请求 如果 I 设备上有来自更高级别控制器的提供数据记录的请求，SFB 74 仅写入 NEW、SLOT、INDEX 和 RLEN 输出参数。如果使用 MODE = 0 连续多次调用 SFB 74，输出参数始终指向同一个请求。
1	接收为 I 设备任意子插槽提供数据记录的请求 如果在 I 设备上有所对于任意子插槽的来自更高级别控制器的此类请求，SFB 74 写入输出参数。
2	接收为 I 设备特定子插槽提供数据记录的请求 如果在 I 设备上有所对于特定子插槽的来自更高级别控制器的此类请求，SFB 74 写入输出参数。
3	提供数据记录，向更高级别控制器发送肯定答复 SFB 74 检查更高级别控制器提供数据记录的请求，在 RECORD 中准备好所请求的记录，向更高级别控制器发送肯定确认。
4	向更高级别控制器发送否定答复 SFB 74 检查更高级别控制器提供数据记录的请求，拒绝该请求，向更高级别控制器发送否定确认。拒绝原因在输入参数 CODE1 和 CODE2 中给出。

### 注意

请求到达后(NEW = 1)，必须调用 SFB 74 两次，确保其完整执行。需要符合下列序列：

- 第一次使用 MODE=1 或 MODE=2 调用
- 第二次使用 MODE=3 或 MODE=4 调用

参数	声明	数据类型	存储区	描述
MODE	INPUT	INT	I、Q、M、D、L、常数	工作模式
F_ID	INPUT	DWORD	I、Q、M、D、L、常数	用于要发送的数据的 I 设备传送区中的子插槽 (仅与 MODE=2 有关) (始终将高位字设为零)
CODE1	INPUT	BYTE	I、Q、M、D、L、常数	零(当 MODE=3 时)或错误代码 1 (当 MODE=4 时)
CODE2	INPUT	BYTE	I、Q、M、D、L、常数	零(当 MODE=3 时)或错误代码 2 (当 MODE=4 时)
LEN	INPUT	INT	I、Q、M、D、L、常数	要发送数据的最大长度，以字节为单位
NEW	OUTPUT	BOOL	I、Q、M、D、L	更高级别控制器请求新数据
STATUS	OUTPUT	DWORD	I、Q、M、D、L	错误信息
SLOT	OUTPUT	INT	I、Q、M、D、L、常数	与 F_ID 相同

参数	声明	数据类型	存储区	描述
SUBSLOT	OUTPUT	INT	I、Q、M、D、L、常数	与 F_ID 相同
INDEX	OUTPUT	INT	I、Q、M、D、L、常数	要发送的数据记录编号
RLEN	OUTPUT	INT	I、Q、M、D、L	要发送的数据记录的长度
RECORD	IN_OUT	ANY	I、Q、M、D、L	提供的数据记录 注意：S7-300 CPU 的 RECORD 参数总是需要被完整指定的 DB 参数(例如：P#DB13.DBX0.0 Byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成出错消息。

### 错误信息

关于 STATUS 参数的解释，请参见用 SFB 54 "RALRM"接收中断

## 5 用于处理系统时钟的 SFC

### 5.1 使用 SFC 0 "SET\_CLK"设定 TOD

#### 描述

使用 SFC 0 "SET\_CLK" (设定系统时钟) ,可以设定 CPU 时钟的时间和日期。调用 SFC 0 会启动时钟。时钟将从设定时间和设定日期开始运行。

如果这个时钟是主时钟，则在调用 SFC 0 的同时，CPU 启动同步时间的功能。可使用 STEP 7 设定同步周期。

参数	声明	数据类型	存储器区	描述
PDT	INPUT	DT	D、L	在输入 PDT 时，输入要设定的日期和时间。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能执行期间出错，返回值会包含故障代码。

#### 日期和时间

以数据类型 DT 输入日期和时间。例如：1995 年 1 月 15 日，上午 10 点 30 分 30 秒，输入为：DT#1995-01-15-10:30:30。输入时间值精确到秒级。由 SFC 0 "SET\_CLK"从日期推算周的天数。

可以使用 FC 3 "D\_TOD\_DT"创建数据类型 DT(请参见时间功能：FC 1、FC 3、FC 6、FC 7、FC 8、FC 33、FC 34、FC 35、FC 40)。

#### 错误信息

错误代码 (W#16#...)	解释
0000	无错误
8080	日期出错
8081	时间出错
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误

## 5.2 使用 SFC 1 "READ\_CLK"读取时间

### 描述

使用 SFC 1 "READ\_CLK" (读取系统时钟), 可以读取当前 CPU 系统时钟的日期和时间。

参数	声明	数据类型	存储器区	描述
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能执行期间出错, 返回值会包含故障代码。
CDT	OUTPUT	DT	D、L	CDT 输出当前日期和时间值。

### 错误信息

请参见通过输出参数 RET\_VAL 判断错误

## 5.3 使用 SFC 48 "SNC\_RTCB"同步子时钟

### 定义：子时钟的同步

子时钟的同步是指将日期和时间从一个总线段的主时钟(例如，S7-400 K 总线、MPI 或 S7 背板总线)传送到该总线段的所有子时钟。

### 描述

使用 SFC 48 "SNC\_RTCB" (实时时钟同步)，可以同步总线程序段上的所有子时钟，并且与组态同步时间间隔无关。只有调用 SFC 48 的 CPU 设置为该总线段上的主时钟时，同步功能才有效。使用 STEP 7 设定相关参数。

参数	声明	数据类型	存储器区	描述
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能执行期间出错，返回值会包含故障代码。

### 错误信息

错误代码 (W#16#...)	解释
0000	在同步期间未产生故障。
0001	对任何总线而言，现时钟都没有设成主时钟。
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误

## 5.4 使用 SFC 100 "SET\_CLKS"设定时间日期和 TOD 状态

### 描述

使用 SFC 100 "SET\_CLKS"设定 CPU 的 TOD 和 TOD 状态。

#### 当心

只有使用 SFC 100，CPU 的时间日期才能同步。否则，每次同步，主站的 TOD 状态都被调用。这就使得每个 SFC 的设定值被刷新。

通过 MODE 参数，可以指定只更改 TOD、只更改 TOD 状态还是同时更改二者。下表对此进行了解释：

MODE (B#16#...)	含义
01	设定 TOD SFC 的调用相当于 SFC 0 "SET_CLK"的调用。 输入参数 CORR、SUMMER 和 ANN_1 无效。
02	设定 TOD 状态 输入参数 PDT 无效。保留的输入参数形成下列 TOD 状态： <ul style="list-style-type: none"> <li>• 带符号的修正值</li> <li>• 时间制式</li> <li>• 夏令时/冬令时标志</li> </ul> CPU 决定 TOD 的分辨率。FALSE 显示 TOD 状态位同步失败。 TOD 保持不变。
03	设定 TOD 与 TOD 状态

#### 注意

通过 SFC 51 "RDSYSST"读取 SSL ID W#16#0132 标志 W#16#0008，可以确定 CPU 的当前 TOD 状态。



参数	声明	数据类型	存储器区	描述
MODE	INPUT	BYTE	I、Q、M、D、L、 常数	操作模式 允许值：
				B#16#01： 设定 TOD
				B#16#02： 设定 TOD 状态
				B#16#03： 设定 TOD 与 TOD 状态
PDT	INPUT	DT	D、L	TOD 预置
CORR	INPUT	INT	I、Q、M、D、L、 常数	修正值(0.5h 模式) 允许值范围：-24 - +26
SUMMER	INPUT	BOOL	I、Q、M、D、L、 常数	夏令时/冬令时标识符： <ul style="list-style-type: none"> <li>• 0 = 冬令时</li> <li>• 1 = 夏令时</li> </ul>
ANN_1	INPUT	BOOL	I、Q、M、D、L、 常数	时间制式 1：从下一个小时起夏令时转换成冬令时(或反之)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误代码：

## 错误信息

错误代码 (W#16#...):	解释
0000	无故障
8080	MODE 超出允许值范围
8081	CORR 超出允许值范围 (仅限 MODE = B#16#02 或 MODE = B#16#03 有效)
8082	PDT 超出允许值范围：非法日期或 TOD
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误



## 6 用于处理运行时间计时器的 SFC

### 6.1 运行时间定时器

#### 引言

CPU 具有若干运行时间定时器(请参照有关 CPU 技术资料)。

- SFC 2、3 和 4 用于设置、启动、停止或读取 CPU 16 位运行时间定时器。
- SFC 101 "RTM"用于设置、启动、停止或读取 CPU 32 位运行时间定时器。

---

#### 注意

SFC 2、3 和 4 也可用于 32 位运行时间定时器。但是，在这种情况下这个运行时间定时器将以 16 位模式工作(取值范围：0 到 32767 小时)。

参见：SSL-ID 为 W#16#0132、索引为 W#16#000B 的部分列表摘录的数据记录。

---

#### 应用

运行时间定时器有各种应用方式：

- 测试 CPU 的运行时间
- 测试控制器或相关设备的运行时间。

#### 运行时间定时器的特征

启动时，运行时间定时器开始记录最后一个数值。如果要求在不同的初始值启动，则这个值必须自己定义(SFC 2 或 SFC 101，MODE=4)。如果 CPU 进入 STOP 状态，或者您停止运行时间定时器，则 CPU 将记录运行时间定时器的当前值。在执行 CPU 热启动或冷启动后，必须重新启动运行时间定时器(SFC 3 或 SFC 101，MODE=1)。更新操作系统以及将 CPU 复位到发货状态后，运行时间定时器复位到数值 0。

#### 取值范围：

- 带有 16 位运行时间定时器的 CPU：0 到 32767 小时
- 带有 32 位运行时间定时器的 CPU：0 到(2E31) -1 小时 = 2.147.483.647 小时

## 6.2 使用 SFC 101 "RTM"控制运行时间定时器

### 描述

调用 SFC 101 "RTM" (运行时间定时器)可设置、启动、停止和读取 CPU 的 32 位运行时间定时器。

调用 SFC 51 "RDSYSST", SZL\_ID=W#16#0132, INDEX=W#16#000B(运行时间定时器 0 ... 7)或 INDEX=W#16#000C(运行时间定时器 8 ... 15) 读取 CPU 32 位运行时间定时器的值。(参见: SSL-ID 为 W#16#0132、索引为 W#16#000B 的部分列表摘录的数据记录。)

参数	声明	数据类型	存储区	描述
NR	INPUT	BYTE	I、Q、M、D、L、 常数	运行时间定时器的编号 编号从 0 开始。 可以在技术规范中找到 CPU 的运行时间定时器的编号。
MODE	INPUT	BYTE	I、Q、M、D、L、 常数	作业标识符： <ul style="list-style-type: none"> <li>• 0: 读取(状态写入 CQ, 当前值写入 CV)。运行时间定时器达到(2E31) -1 小时, 它会停止运行并显示这个最高值, 并输出"溢出"出错消息。</li> <li>• 1: 启动(从上一计数值开始)</li> <li>• 2: 停止</li> <li>• 4: 设置(在 PV 中的设定值)</li> <li>• 5: 设置(在 PV 中的设定值)并启动</li> <li>• 6: 设置(在 PV 中的设定值)并停止</li> </ul>
PV	INPUT	DINT	I、Q、M、D、L、 常数	运行时间定时器的新数值
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果该功能运行时出错, 则返回值将包含故障代码。
CQ	OUTPUT	BOOL	I、Q、M、D、L	运行时间定时器状态(1: 运行)
CV	OUTPUT	DINT	I、Q、M、D、L	运行时间定时器当前值

### 带有 16 位运行时间定时器的程序兼容性

SFC 2 "SET\_RTM"、SFC 3 "CTRL\_RTM"和 SFC 4 "READ\_RTM"可用于操作 32 位运行时间定时器。但此时, 32 位运行时间定时器只能当作 16 位使用 (取值范围: 0 到 32767 小时)。

具有 SSL ID W#16#0132 和索引 W#16#0008 的部分列表摘录以 16 位模式显示 32 位运行时间定时器 0 到 7。也就是说, 您可以继续使用为具有 16 位运行时间定时器的 CPU 开发的程序, 这些程序使用具有 SSL ID W#16#0132 和索引 W#16#0008 的部分列表摘录。

## 错误信息

故障代码 (W#16#...)	解释
0000	无错误
8080	运行时间定时器编号错误
8081	负值被传送到参数 PV。
8082	运行时间定时器溢出。
8091	输入参数 MODE 的值非法。
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误

## 6.3 使用 SFC 2 "SET\_RTM"设置运行时间定时器

### 描述

使用 SFC 2"SET\_RTM" (设置运行时间定时器), 可将 CPU 的运行时间定时器设置为所选值。可以设置的运行时间定时器数量取决于所使用的 CPU。

参数	声明	数据类型	存储器区	描述
NR	INPUT	BYTE	I、Q、M、D、L、常数	输入 NR 包含您要设置的运行时间定时器的编号。编号从 0 开始。 可以在技术规范中找到 CPU 的运行时间定时器的编号。
PV	INPUT	INT	I、Q、M、D、L、常数	参数 PV 用于运行时间定时器设置(缺省值)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在执行功能时出错, 则返回值将包含故障代码。

### 错误信息

错误代码 (W#16#...)	解释
0000	无错误
8080	运行时间定时器的编号错误
8081	负值被传送到参数 PV。
8xyy	一般故障代码, 请参见通过输出参数 RET_VAL 判断错误

## 6.4 使用 SFC 3 "CTRL\_RTM"启动和停止运行时间定时器

### 描述

SFC 3 "CTRL\_RTM" (控制运行时间定时器)用于启动或停止 CPU 的运行时间定时器。

参数	声明	数据类型	存储器区	描述
NR	INPUT	BYTE	I、Q、M、D、L、常数	输入 NR 包含您要启动或停止的运行时间定时器的编号。 编号从 0 开始。 可以在技术规范中找到 CPU 的运行时间定时器的编号。
S	INPUT	BOOL	I、Q、M、D、L、常数	输入 S 可启动或停止运行时间定时器。当您想要停止计数器时，请将信号状态设置为"0"。当您想要启动计数器时，请将信号状态设置为"1"。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在执行功能时出错，则返回值将包含故障代码。

### 错误信息

错误代码 (W#16#...)	解释
0000	无错误
8080	运行时间定时器的编号错误
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误

## 6.5 使用 SFC 4 "READ\_RTM"读取运行时间定时器

### 描述

使用 SFC 4 "READ\_RTM" (读取运行时间定时器), 可以读取运行时间定时器。SFC 4 以输出数据的形式提供当前的运行时间, 以及计数器的状态, 例如, "停止"或"计数"。

如果运行时间定时器运行超过 32767 小时, 它将停在计数值 32767 并输出"溢出"的故障消息。

参数	声明	数据类型	存储器区	描述
NR	INPUT	BYTE	I、Q、M、D、L、常数	输入 NR 包含您要读取的运行时间定时器的编号。编号从 0 开始。 可以在技术规范中找到 CPU 的运行时间定时器的编号。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能运行时出错, 则返回值包含故障代码。
CQ	OUTPUT	BOOL	I、Q、M、D、L	CQ 显示运行时间定时器是否运行。信号状态"0"表示运行时间定时器已被停止。信号状态"1"表示运行时间定时器正在运行。
CV	OUTPUT	INT	I、Q、M、D、L	CV 显示运行时间定时器的当前值。

### 错误信息

错误代码 (W#16#...)	解释
0000	无错误
8080	运行时间定时器的编号错误
8081	运行时间定时器溢出
8xyy	一般故障代码, 请参见通过输出参数 RET_VAL 判断错误



## 6.6 使用 SFC 64 "TIME\_TCK"读取系统时间

### 描述

使用 SFC 64 "TIME\_TCK" (报时信号)，可以读取 CPU 的系统时间。系统时间是循环时间计数器，技术范围 0 ~ 2147483647 毫秒。如果出现溢出，系统时间将从 0 开始重新计数。分辨率和系统时间精度为 1 ms。只有 CPU 的操作模式才影响系统时间。

### 应用

例如，比较 SFC 64 两次执行的结果，测试执行周期。

### 系统时间和模式

模式	系统时间...
启动	... 不断更新
RUN	
STOP	... 停止并保持当前值
热启动 (不包括 S7-300 和 S7-400 H)	... 以切换到 STOP 模式时所保存的值继续
暖启动	... 删除并从"0"重新开始
冷启动	

参数	声明	数据类型	存储器区	描述
RET_VAL	OUTPUT	TIME	I、Q、M、D、L	RET_VAL 参数读取的系统时间 取值范围：0 到 $2^{31}-1$ 毫秒。

### 错误信息

SFC 64 "TIME\_TCK"不提供任何故障信息。



## 7 用于传送数据记录的 SFC/SFB

### 7.1 数据记录的写和读

#### 原理

某些模块具有只写的系统数据区，可由程序向其传送数据记录。此区域包含编号从 0 到最大为 240 的数据记录。并不是每个模块都包含所有数据记录(参见下表)。

另外，一些模块具有只读的系统数据区，可由程序从中读取数据记录。此区域包含编号从 0 到最大为 240 的数据记录。并不是每个模块都包含所有数据记录(参见下表)。

---

#### 注意

有的模块同时具有这两种系统数据区。在物理上这些区域是分开的，在逻辑结构上它们是相同的。

---

#### 只写的系统数据区

下表显示只写的系统数据区的结构。此表还显示数据记录的允许长度以及利用哪些 SFC 写这些数据记录。

数据记录号	目录	大小	限制	可用于写操作的 SFC
0	参数	S7-300 : 2 到 14 个字节	只能由 S7-400 写	56 "WR_DPARAM" 57 "PARAM_MOD"
1	参数	S7-300 : 2 到 14 个字节 数据记录 0 和 1 的总和正好是 16 字节。	-	55 "WR_PARM" 56 "WR_DPARAM" 57 "PARAM_MOD"
2 - 127	用户数据	每个 = 240 个字节	-	55 "WR_PARM" 56 "WR_DPARAM" 57 "PARAM_MOD" 58 "WR_REC"
128 - 240	参数	每个 = 240 个字节	-	55 "WR_PARM" 56 "WR_DPARAM" 57 "PARAM_MOD" 58 "WR_REC"

## 只读的系统数据区

下表显示只读的系统数据区的结构。此表还显示数据记录的允许长度以及利用哪些 SFC 读取这些数据记录。

数据记录号	目录	大小	可用于读操作的 SFC
0	模块专用诊断数据 (标准设定用于整个系统)	4 字节	51 "RDSYSST" (SSL_ID 00B1H) 59 "RD_REC"
1	通道专用诊断数据 (包括数据记录 0)	<ul style="list-style-type: none"> <li>• S7-300 : 16 字节</li> <li>• S7-400 : 4 - 220 字节</li> </ul>	51 "RDSYSST" (SSL_ID 00B2H 和 00B3H) 59 "RD_REC"
2 - 127	用户数据	每个 ≤ 240 个字节	59 "RD_REC"
128 - 240	诊断数据	每个 ≤ 240 个字节	59 "RD_REC"

## 系统资源

如果有若干个异步数据记录传送接连被启动，而它们之间只有很短的时间间隔，则操作系统对系统资源的分配可保证所有的任务都被执行且相互之间不发生冲突。

如果所有可用的系统资源都在使用，则在参数 RET\_VAL 中会有显示。这种临时出错状况可通过简单的任务重复加以补救。

可"同时"激活的 SFC 任务的最大数量依 CPU 而定。详细信息，请参考/72/、/101/和/102/。

## 7.2 使用 SFC 54 "RD\_DPARM"读取定义的参数

### 描述

通过 SFC 54 "RD\_DPARM" (读取定义的参数), 可从用 STEP 7 组态的系统数据中读取指定地址模块的、编号为 RECNUM 的数据记录。读取的数据记录被传送到由参数 RECORD 开辟的目标区域。

参数	声明	数据类型	存储器区	描述
IOID	INPUT	BYTE	I、Q、M、D、L、 常数	地址区域的 ID : B#16#54 =外设输入(PI) B#16#55 =外设输出(PQ) 若为混合模块, 则指定最低地址的区域 ID。如果 两个地址相同, 则指定 B#16#54。
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的逻辑基址。对于混合模块, 则指定两个地址 中较低的一个。
RECNUM	INPUT	BYTE	I、Q、M、D、L、 常数	数据记录号(允许值: 0 - 240)
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果读取的数据记录长度与目标区域匹配, 且传送 时未出错, 返回值是被读取的数据记录的字节数。 如果在功能激活时出错, 则返回值包含故障代码。
RECORD	OUTPUT	ANY	I、Q、M、D、L	被读取数据记录的目标区域。仅允许使用数据类型 BYTE。

### 错误信息

使用 SFC 57 "PARM\_MOD"为模块分配参数

## 7.3 使用 SFC 102 "RD\_DPARA"读取预定义参数

### 描述

通过 SFC 102 "RD\_DPARA" ,可从从 STEP 7 组态的系统数据中读取所选模块的编号为 RECNUM 的数据记录。读取的数据记录被传送到由参数 RECORD 开辟的目标区域。

### 工作原理

SFC 102 "RD\_DPARA"以异步方式操作，也就是说，处理过程包含 SFC 的多次调用。在 REQ = 1 时调用 SFC 102 以启动任务。

作业状态可通过输出参数 RET\_VAL 和 BUSY 显示。

请参见 REQ、RET\_VAL 和 BUSY 对于异步工作的 SFC 的意义

参数	声明	数据类型	存储区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	REQ=1：读请求
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的任何地址。对于输出地址，必须置位最高位。
RECNUM	INPUT	BYTE	I、Q、M、D、L、 常数	数据记录号(允许值：0 - 240)
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错，则返回值包含故障代码。 如果传送期间未出错，可区分下列两种情况： <ul style="list-style-type: none"> <li>如果目标区域大于读取的数据记录，RET_VAL 包含实际读取数据记录的长度(字节)。</li> <li>如果读取的数据记录的长度等于目标区域长度，RET_VAL 为 0。</li> </ul>
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：任务尚未结束。
RECORD	OUTPUT	ANY	I、Q、M、D、L	被读取数据记录的目标区域。只允许数据类型 BYTE。 注意：注意，S7-300 CPU 的 RECORD 参数总是需要完整指定 DB 参数(例如：P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成一条出错消息。

### 错误信息

请参见使用 SFC 57 "PARM\_MOD"为模块分配参数

## 7.4 使用 SFC 55 "WR\_PARM"写动态参数

### 描述

通过 SFC 55 "WR\_PARM" (写参数)，可将数据记录 RECORD 传送到指定地址的模块。传送到模块去的参数不会被 STEP 7 组态的参数覆盖。

### 要求

- 要传送的数据记录不允许是静态的。  
(请参考/71/和/101/ 有关静态数据记录的详细信息，)

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、 常数	REQ=1：写请求
IOID	INPUT	BYTE	I、Q、M、D、L、 常数	地址区域的 ID： B#16#54 =外设输入(PI) B#16#55 =外设输出(PQ) 若为混合模块，则指定最低地址的区域 ID。如果两个地址相同，则指定 B#16#54。
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的逻辑基址。对于混合模块，则指定两个地址中较低的一个。
RECNUM	INPUT	BYTE	I、Q、M、D、L、 常数	数据记录号
RECORD	INPUT	ANY	I、Q、M、D、L	数据记录 注意：注意，S7-300 CPU 的 RECORD 参数总是需要完整指定 DB 参数(例如：P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成一条出错消息。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错，则返回值包含故障代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：写操作尚未完成。

### 输入参数 RECORD

要传送的数据是在第一个 SFC 调用时从参数 RECORD 读取的。如果数据记录的传送占用的时间长于一次 SFC 调用的持续时间，则参数 RECORD 的内容不再与后续的 SFC 调用(同一任务)相关。

### 错误信息

使用 SFC 57 "PARM\_MOD"为模块分配参数

---

**注意**  
(只对 S7-400)

如果出现常规错误 W#16#8544，只是表明对于至少一个字节的包含数据记录的 I/O 存储区访问被拒绝。数据传送继续进行。

---



## 7.5 使用 SFC 56 "WR\_DPARM"写缺省参数

### 描述

通过 SFC 56 "WR\_DPARM" (写缺省参数), 可将 STEP 7 组态数据中编号为 RECNUM 的数据记录传送到指定地址的模块。使用本功能, 数据记录是静态还是动态都没有关系。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、 常数	REQ=1 : 写请求
IOID	INPUT	BYTE	I、Q、M、D、L、 常数	地址区域的 ID : B#16#54 =外设输入(PI) B#16#55 =外设输出(PO) 若是混合模块, 指定最低地址的区域 ID。如果两个地址相同, 则指定 B#16#54。
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的逻辑基址。对于混合模块, 则指定两个地址中较低的一个。
RECNUM	INPUT	BYTE	I、Q、M、D、L、 常数	数据记录号
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错, 则返回值包含故障代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1 : 写操作尚未完成。

### 错误信息

使用 SFC 57 "PARM\_MOD"为模块分配参数

## 7.6 使用 SFC 57 "PARM\_MOD"为模块分配参数

### 描述

通过 SFC 57 "PARM\_MOD" (为模块分配参数), 可将由 STEP 7 组态生成的某一模块中的全部数据记录传送到该模块。使用本功能, 数据记录是静态还是动态都没有关系。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	<u>BOOL</u>	I、Q、M、D、L、 常数	REQ=1 : 写请求
IOID	INPUT	<u>BYTE</u>	I、Q、M、D、L、 常数	地址区域的 ID : B#16#54 = 外设输入(PI) B#16#55 = 外设输出(PQ) 若是混合模块, 指定最低地址的区域 ID。 如果两个地址相同, 则指定 B#16#54。
LADDR	INPUT	<u>WORD</u>	I、Q、M、D、L、 常数	模块的逻辑基址。对于混合模块, 则指定两个地址中较低的一个。
RET_VAL	OUTPUT	<u>INT</u>	I、Q、M、D、L	如果在功能激活时出错, 则返回值包含故障代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1 : 写操作尚未完成。

### 错误信息

"真实的"故障信息(故障代码 W#16#8xyz)可分为两类 :

- 暂时故障(故障代码 W#16#80A2 到 80A4、80Cx) :  
对于此类故障, 可能在不采取任何措施的情况下就会消除, 换句话说, 重新调用该 SFC (必要时多次调用)是可行的。  
例如, 所需资源当前正在使用(W#16#80C3)时, 即会出现暂时故障。
- 永久故障(故障代码 W#16#809x、80A1、80Bx、80Dx) :  
如果不采取措施, 此类故障就不会消除。只有消除故障后, 再次调用 SFC 才能成功。  
例如, 在 RECORD 中输入长度错误(W#16#80B1)即为一个永久故障。

### 注意

如果使用 SFC 55、56 或 57 向 DPV1 从站传送数据记录且该从站运行于 DPV1 模式, DP 主站会按如下原则判断它从此从站接收的故障信息 :

如果错误信息位于从 W#16#8000 到 W#16#80FF 或从 W#16#F000 到 W#16#FFFF 的范围内, DP 主站将错误信息传送到 SFC。如果它超出此范围, 则 CPU 将值 W#16#80A2 传送到 SFC, 并将从站挂起。

SFC 54 "RD\_DPARM"、SFC 55 "WR\_PARM"、SFC 56 "WR\_DPARM"和 SFC 57 "PARM\_MOD"的故障信息。

错误代码 (W#16#...)	解释	限制
0000	无错误	-
7000	首次调用 REQ=0：没有数据传输；BUSY 的值为 0。	-
7001	首次调用 REQ=1：已启动数据传送；BUSY 的值为 1。	分布式 I/O
7002	中间调用(REQ 无关联)：数据传送激活；BUSY 的值为 1。	分布式 I/O
8090	指定的逻辑基址无效：SDB1/SDB2x 中没有赋值，或者没有基址。	-
8092	在 ANY 参考中指定的类型不是 BYTE。	仅适用于 S7-400 的 SFC 54 "RD_PARM"和 SFC 55 "WR_PARM"
8093	本 SFC 不允许用于由 LADDR 和 IOID 所指定的模块(允许下列模块：S7-300 模块用于 S7-300、S7-400 模块用于 S7-400、S7-DP 模块用于 S7-300 和 S7-400)。	-
80A1	当向模块发送数据记录时，收到否定的确认(模块已拆除或在传送期间损坏)。	1)
80A2	DP 协议在第 2 层出错，可能 DP 从站的硬件/接口故障	分布式 I/O 1)
80A3	DP 协议与用户接口/用户出错。	分布式 I/O 1)
80A4	通讯总线上的通讯问题。	CPU 与外部 DP 接口模块之间出现故障 1)
80B0	SFC 不能用于此类型模块，模块不识别数据记录。	1)
80B1	传送的数据记录的长度不正确。SFC 54 "RD_PARM"：由 RECORD 打开的目标区域太短。	-
80B2	组态的插槽未占用。	1)
80B3	实际模块类型与 SDB1 中要求的模块类型不匹配。	1)
80C1	对于模块的同一数据记录，上一个写任务的数据还未被模块处理。	1)
80C2	模块正在处理对一个 CPU 所可能的最大数量的任务。	1)
80C3	需要的资源(存储区等)当前被占用。	1)
80C4	内部暂时故障。无法处理任务。 • 重复执行该作业。如果经常发生此故障，请检查您的系统是否存在电干扰源。	1)
80C5	分布式 I/O 不存在。	分布式 I/O 1)
80C6	由于优先级的中止(热启动或后台操作)，数据记录传送已停止	分布式 I/O 1)
80D0	在对应模块的 SDB 中无条目。	-
80D1	数据记录在对应模块的 SDB 中未组态(数据记录号 = 241 被 STEP 7 拒绝)。	-
80D2	根据模块类型标识符，该模块不能被分配参数。	-
80D3	不能访问该 SDB，因为它不存在。	-

错误代码 (W#16#...)	解释	限制
80D4	SDB 结构错误：SDB 内部指针指向 SDB 以外的值。	仅对 S7-300
80D5	数据记录是静态的。	仅对 SFC 55 "WR_PARM"
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误。	

在 SFC 54 RD\_DPARAM"中不会发生

## 7.7 使用 SFC 58 "WR\_REC"写数据记录

### 描述

通过 SFC 58 "WR\_REC" (写记录), 可将包含在 RECORD 中的数据记录传送到指定地址的模块。调用 SFC 58 时, 给输入参数 REQ 赋值 1, 可启动写任务。如果写任务可以立即执行, 此 SFC 将在输出参数 BUSY 中返回值 0。如果 BUSY 值为 1, 说明写任务尚未完成。

### 注意

如果使用 GSD 文件(版本 3 以上的 GSD)对 DPV1 从站进行组态, 且 DP 主站的 DP 接口设置为"**S7 兼容**", 则一定不能在用户程序中通过 SFC 58 将数据记录写入 I/O 模块。其原因就是此时的 DP 主站将寻址错误的插槽(已组态的插槽+3)。

纠正方法: 将 DP 主站的接口设置为"DPV1"。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、 常数	REQ=1: 写请求
IOID	INPUT	BYTE	I、Q、M、D、L、 常数	地址区域的 ID: B#16#54 = 外设输入(PI) B#16#55 = 外设输出(PQ) 若是混合模块, 指定最低地址的区域 ID。 若地址相同, 指定 B#16#54。
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的逻辑基本地址。对于混合模块, 则指定两个地址中较低的一个。
RECNUM	INPUT	BYTE	I、Q、M、D、L、 常数	数据记录号(允许值: 2 - 240)
RECORD	INPUT	ANY	I、Q、M、D、L	数据记录。只允许数据类型 BYTE。 注意: 注意, S7-300 CPU 的 RECORD 参数总是需要完整指定 DB 参数(例如: P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的, 这将会导致在用户程序中生成一条出错消息。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错, 则返回值包含故障代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1: 写操作尚未完成。

## 输入参数 RECORD

要传送的数据是在第一个 SFC 调用时从参数 RECORD 读取的。如果数据记录的传送占用的时间长于一次 SFC 调用的持续时间，则参数 RECORD 的内容不再与后续的 SFC 调用(同一任务)相关。

## 错误信息

使用 SFC 59 "RD\_REC"读数据记录

---

### 注意

如果出现常规错误 W#16#8544，只是表明对于至少一个字节的包含数据记录的 I/O 存储区访问被拒绝。数据传送继续进行。

---

## 7.8 使用 SFC 59 "RD\_REC"读数据记录

### 描述

通过 SFC 59 "RD\_REC" (读记录)，可从指定地址的模块中读取编号为 RECNUM 的数据记录。通过调用 SFC 59 且将输入参数 REQ 置 1 启动读任务。如果读任务可以立即执行，此 SFC 在输出参数 BUSY 中返回值 0。如果 BUSY 的值为 1，则读任务尚未完成。如果数据传送没有错误，读取的数据记录将被传送到由 RECORD 参数指定的目标区域中。

#### 注意

如果从 1997 年 2 月以前购买的 FM 或 CP (下文称"旧式模块")中读取编号 > 1 的数据记录，SFC 59 的响应会有所不同。这一特殊状况在章节"使用旧的 S7-300 FM 和 CP 数据记录编号 >1"中有说明。

#### 注意

如果使用 GSD 文件(版本 3 以上的 GSD)对 DPV1 从站进行组态，且 DP 主站的 DP 接口设置为 "S7 兼容"，则一定不能在用户程序中通过 SFC 59 从 I/O 模块读取数据记录。其原因就是此时的 DP 主站将寻址错误的插槽(已组态的插槽+3)。

纠正方法：将 DP 主站的接口设置为"DPV1"。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、 常数	REQ=1：读请求
IOID	INPUT	BYTE	I、Q、M、D、L、 常数	地址区域的 ID： B#16#54 = 外设输入(PI) B#16#55 = 外设输出(PQ) 若是混合模块，指定最低地址的区域 ID。如果两个地址相同，则指定 B#16#54。
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的逻辑基本地址。对于混合模块，则指定两个地址中较低的一个。
RECNUM	INPUT	BYTE	I、Q、M、D、L、 常数	数据记录号(允许值 0 - 240)
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错，则返回值包含故障代码。如果目标区域大于传送的数据记录且传送中未出现错误，则返回值是被传送的数据记录实际字节数 (可能值： +1 - +240)。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：读操作尚未完成。
RECORD	OUTPUT	ANY	I、Q、M、D、L	被读取数据记录的目标区域。在异步执行 SFC59 时，确保在所有调用中 RECORD 的实际参数的长度信息相同。只允许数据类型 BYTE。 注意：注意，S7-300 CPU 的 RECORD 参数总是需要完整指定 DB 参数(例如：P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成一条出错消息。

### 输出参数 RET\_VAL

- 如果在执行此功能时出错，则返回值包含故障代码。
- 如果未出错，RET\_VAL 包含下列内容：
  - 0: 整个目标区域都被所选数据记录填充(数据记录也有可能不完整)。
  - 如果目标区域大于传送的数据记录且传送中未出现错误，则返回值是被传送的数据记录实际字节数 (可能值：+1 - + 240) (如果目标区域大于传送的数据记录)。

---

### 注意

如果出现常规错误 W#16#8545，这只表明对于至少一个字节包含数据记录的 I/O 存储区的访问遭到拒绝。数据记录已被模块正确读取并写入 I/O 存储区。

---



## 设定 RECORD

---

### 注意

如果要确保每次都读取完整的数据记录，可将目标区长度选为 241 字节。如果数据传送无错，RET\_VAL 包含被传送数据记录的实际字节数。

---

## 使用旧的 S7-300 FM 和 CP 数据记录号 > 1

如果要使用 SFC 59 "RD\_REC"从旧的 S7-300 FM 和 S7-300 CP 中读取编号大于 1 的数据记录，请切记下列要点：

- 如果目标区域长度大于数据记录所需的实际长度，则数据不传入 RECORD。  
RET\_VAL 的值为 W#16#80B1。
- 如果目标区域长度小于数据记录所需的实际长度，CPU 将从数据记录的起始读取 RECORD 长度信息中所定义的字节，并将这些字节传送到 RECORD。  
RET\_VAL 值为 0。
- 如果 RECORD 中指定的长度与数据记录的所需实际长度相同，CPU 读取数据记录并将其输入 RECORD 中。  
RET\_VAL 值为 0。

## 错误信息

下表中的"真实的"故障信息(故障代码 W#16#8xyz)可分为两类：

- 暂时故障(故障代码 W#16#80A2 到 80A4、80Cx)：  
对于此类故障，有可能在不采取任何措施的情况下故障即会消除，换句话说，再次调用该 SFC (必要时多次调用)是可行的。  
例如，所需资源当前正在使用(W#16#80C3)时，即会出现暂时故障。
- 永久故障(故障代码 W#16#809x、80A1、80Bx、80Dx)：  
如果不采取措施，此类故障就不会消除。只有消除故障后，再次调用 SFC 才能成功。例如，在 RECORD 中输入长度错误(W#16#80B1)即为一个永久故障。

---

### 注意

如果利用 SFC 58 "WR\_REC"向 DPV1 从站传送数据记录或利用 SFC 59 "RD\_REC"从 DPV1 从站读取数据记录且此从站正以 DPV1 模式运行，则 DP 主站会按如下原则判断所接收的来自从站的故障信息：

如果错误信息位于从 W#16#8000 到 W#16#80FF 或从 W#16#F000 到 W#16#FFFF 的范围内，DP 主站将错误信息传送到 SFC。如果它超出此范围，则 CPU 将值 W#16#80A2 传送到 SFC，并将从站挂起。

对于来自 DPV1 从站故障信息的说明，请参见用 SFB 54 "RALRM"接收中断。

---

针对 SFC 58 "WR\_REC"和 SFC 59 "RD\_REC"的故障信息。

错误代码 (W#16#...)	解释	限制
0000	无错误	-
7000	首次调用 REQ=0：没有数据传输；BUSY 的值为 0。	-
7001	首次调用 REQ=1：没有数据传输；BUSY 的值为 1。	分布式 I/O
7002	中间调用(REQ 无关联)：数据传送已经激活；BUSY 的值为 1。	分布式 I/O
8090	指定的逻辑基址无效：SDB1/SDB2x 中没有赋值，或者没有基址。	-
8092	在 ANY 参考中指定的类型不是 BYTE。	仅限 S7-400
8093	本 SFC 不允许用于由 LADDR 和 IOID 所指定的模块(允许下列模块：S7-300 模块用于 S7-300、S7-400 模块用于 S7-400、S7-DP 模块用于 S7-300 和 S7-400)。	-
80A0	当从模块读取数据时，收到消极的应答 <ul style="list-style-type: none"> <li>模块在读任务期间被拆除或损坏</li> <li>另外，对于 H 系统：备用 CPU 一侧的 I/O 不可用(例如，CPU 处于 STOP 状态)</li> </ul>	仅限 SFC 59 "RD_REC"
80A1	向模块发送数据记录时收到消极的应答 <ul style="list-style-type: none"> <li>该模块已在传送期间拆除或损坏</li> <li>另外，对于 H 系统：备用 CPU 一侧的 I/O 不可用(例如，CPU 处于 STOP 状态)</li> </ul>	仅限 SFC 58 "WR_REC"
80A2	<ul style="list-style-type: none"> <li>第 2 层上的 DP 协议错误(例如，从站故障或总线问题)</li> <li>对于 ET200S，无法在 DPV0 模式下读取数据记录。</li> </ul>	分布式 I/O
80A3	DP 协议与用户接口/用户出错	分布式 I/O
80A4	通讯总线上的通讯问题	错误发生在 CPU 与外部 DP 接口模块之间。
80B0	<ul style="list-style-type: none"> <li>SFC 不能用于此类型模块。</li> <li>模块不能识别数据记录。</li> <li>不允许使用数据记录号 241。</li> <li>对于 SFC 58 (WR_REC)，不允许数据记录号 0 和 1。</li> </ul>	-
80B1	RECORD 参数中定义的长度不正确。	<ul style="list-style-type: none"> <li>SFC 58 "WR_REC"：长度不正确</li> <li>SFC 59 "RD_REC" (仅在使用旧 S7-300 FM 和 S7-300 CP 时)：指定长度 &gt; 记录长度</li> <li>对于 SFC 13 "DPNRM_DG"：指定长度 &gt; 记录长度</li> </ul>
80B2	组态的插槽未占用。	-
80B3	实际模块类型与 SDB1 中要求的模块类型不匹配	-
80B7	DP 从站或模块发出信号表示这是一个参数或数值的禁止使用的范围	仅对 SFC 59 "RD_REC"

错误代码 (W#16#...)	解释	限制
80C0	<p>对于 SFC 58 "WR_REC"：只有在 CPU 处于 STOP 模式时才能写入数据。注意：这使得无法通过用户程序写入。只能通过 PG/PC 在线写入数据。</p> <p>对于 SFC 59 "RD_REC"：模块具有数据记录，但不能读取数据或只能在 CPU 处于 STOP 模式时才能读取数据。注意：若只有在 CPU 处于 STOP 模式时才能读取数据，则无法由用户程序进行判断。这种情况下，只能通过 PG/PC 在线读取数据。</p> <p>对于 SFC 13 "DPNRM_DG"：没有诊断信息。</p>	对于 SFC 58 "WR_REC"或 SFC 59 "RD_REC"或 SFC 13 "DPNRM_DG"
80C1	对于模块的同一数据记录，上一个写任务的数据还未被模块处理。	-
80C2	模块正在处理对一个 CPU 所可能的最大数量的任务。	-
80C3	需要的资源(存储区等)当前被占用。	-
80C4	<p>内部暂时故障。无法处理任务。</p> <ul style="list-style-type: none"> <li>重复执行该作业。如果经常发生此故障，请检查您的系统是否存在电干扰源。</li> </ul>	-
80C5	分布式 I/O 不存在。	分布式 I/O
80C6	由于优先级的中止(热启动或后台)，数据记录传送已停止	分布式 I/O
8xyy	一般故障代码，请参见通过输出参数 RET_VAL 判断错误	

## 7.9 SFC 55 到 59 的进一步故障信息

### 仅限 S7-400

使用 S7-400 时 SFC 55 到 59 也可返回故障信息 W#16#80Fx。在此情况下，将产生无法定位的故障。此时，请联系维护部门。

## 7.10 使用 SFB 81 "RD\_DPAR"读取预定义的参数"

### 描述

SFB 81 "RD\_DPAR" (读取设备参数)允许您通过组件(可从用 STEP 7 组态的系统数据中寻址该组件) INDEX 号读取数据记录。这可以是集中插入的模块或分布式组件(PROFIBUS DP 或 PROFINET IO)。

VALID 输出参数的值 TRUE 表明数据记录已成功传送到 RECORD 目标地址。这种情况下，LEN 输出参数包含读取数据的长度(单位为字节)。

如果传送数据记录期间出错，则会通过输出参数 ERROR 加以指示。如果发生错误，则输出参数 STATUS 包含错误信息。

### 功能

SFB 81 "RD\_DPAR"以异步方式运行；即，处理过程涉及多个 SFB 调用。要启动数据传送，调用 REQ = 1 的 SFB 81。

输出参数 BUSY 和输出参数 STATUS 的字节 2 和 3 显示作业状态。STATUS 的字节 2 和 3 对应于异步操作 SFC 的输出参数 RET\_VAL(另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)

当输出参数 BUSY 值为 FALSE 时，即说明数据记录的传送已完成。

### 参数

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	E、A、M、D、L	REQ=1：读请求
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的任意逻辑地址 第 15 位指示显示的是输入地址(位 15 = 0) 还是输出地址(位 15 = 1)。
INDEX	INPUT	INT	I、Q、M、D、L、 常数	数据记录号
VALID	OUTPUT	BOOL	I、Q、M、D、L	新数据记录已被接收且有效
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：作业未终止。
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR = 1：读取过程中出错。
STATUS	OUTPUT	DWORD	I、Q、M、D、L	<ul style="list-style-type: none"> <li>调用标识符(字节 2 和 3)或错误代码</li> <li>字节 1：B#16#00 (如果无错误)。DPV1-PDU 上的其它功能标识符：如果在读取数据记录 B#16#DE 时出现一个错误，如果在写入数据记录 B#16#DF 时出现一个错误。如果未使用 DPV1 协议单元：B#16#C0。</li> <li>字节 4：错误 ID 的供应商特定扩展名</li> </ul>

参数	声明	数据类型	存储器区	描述
LEN	OUTPUT	INT	I、Q、M、D、L	读取的数据记录信息的长度
RECORD	IN_OUT	ANY	I、Q、M、D、L	用于读取数据记录的目标区域 注意：切记：S7-300 CPU 的 RECORD 参数始终要求完整的 DB 参数说明(例如：P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成一条出错消息。

## 错误信息

请参见使用 SFC 57 "PARM\_MOD"为模块分配参数



## 8 符合 PNO AK 1131 的 DPV1 SFB

### 8.1 用 SFB 52 "RDREC"读取数据记录

---

#### 注意

SFB52 "RDREC"接口与"符合 IEC 61131-3 标准的 PROFIBUS 指南 PROFIBUS 通讯和代理功能块"标准中定义的 FB "RDREC"完全相同。

---

#### 描述

使用 SFB52 "RDREC" (读取记录), 可从组件读取编号为 INDEX 的数据记录。这可能是中央机架或分布式组件(PROFIBUS DP 或 PROFINET IO)中的一个模块。

在 MLEN 中指定要读取的最多字节数。目标区域 RECORD 的选定长度至少应等于 MLEN 字节的长度。

输出参数 VALID 如为 TRUE, 则表明已将数据记录成功传送到目标区域 RECORD 中。此时, 输出参数 LEN 包含所取得的数据的长度(以字节计)。

输出参数 ERROR 用以指示是否发生数据记录传送错误。如果发生错误, 则输出参数 STATUS 包含错误信息。

---

#### 注意

如果使用 GSD 文件(版本 3 以上的 GSD)对 DPV1 从站进行组态, 且 DP 主站的 DP 接口设置为 "S7 兼容", 则一定不能在用户程序中通过 SFC 52 从 I/O 模块读取数据记录。原因就是此时的 DP 主站将寻址错误的插槽(已组态的插槽+3)。

纠正方法: 将 DP 主站的接口设置为"DPV1"。

---

## 工作原理

SFC 52 "RDREC"以异步方式工作，即处理过程涉及多个 SFC 调用。通过使用 REQ = 1 调用 SFC52 来启动作业。

作业状态通过输出参数 BUSY 和输出参数 STATUS 的中间两个字节显示。此处，STATUS 的中间两个字节与异步 SFC 的输出参数 RET\_VAL 对应(参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

如果输出参数 BUSY = FALSE，则表明数据记录传送完成。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	REQ=1：传送数据记录
ID	INPUT	DWORD	I、Q、M、D、L、常数	DP 从站/PROFINET IO 组件(模块或子模块)的逻辑地址 对于输出模块，必须设置位 15(例如对于地址 5：ID:=DW#16#8005)。 对于组合模块，应确定两个地址中的较小地址。
INDEX	INPUT	INT	I、Q、M、D、L、常数	数据记录号。
MLEN	INPUT	INT	I、Q、M、D、L、常数	要获取的数据记录信息的最大长度(以字节计)
VALID	OUTPUT	BOOL	I、Q、M、D、L	新数据记录已被接收且有效
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：读操作尚未结束。
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR = 1：发生读错误。
STATUS	OUTPUT	DWORD	I、Q、M、D、L	块状态或错误信息
LEN	OUTPUT	INT	I、Q、M、D、L	所获取数据记录信息的长度
RECORD	IN_OUT	ANY	I、Q、M、D、L	被读取数据记录的目标区域 注意：注意，S7-300 CPU 的 RECORD 参数总是需要完整指定 DB 参数(例如：P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成一条出错消息。

### 注意

如果使用 SFB 52 读取 PROFINET IO 的数据记录，INDEX、MLEN 和 LEN 参数中的负值将被解释为不带前缀符号的 16 位整数。

## 错误信息

关于 STATUS 参数的解释，请参见用 SFB 54 "RALRM"接收中断



## 8.2 用 SFB53 "WRREC"写入数据记录

---

### 注意

SFB53 "WRREC"接口与在符合 IEC 61131-3 标准的 PROFIBUS 指南 PROFIBUS 通讯和代理功能块"标准中定义的 FB "WRREC"完全相同。

---

### 描述

使用 SFB53 "WRREC" (写入记录) ,将编号为 INDEX 的数据记录传送到由 ID 定义的 DP 从站/PROFINET IO 设备组件。这可能是中央机架或分布式组件(PROFIBUS DP 或 PROFINET IO)中的一个模块。

指定要传送的数据记录的字节长度。因此 ,源区域 RECORD 的选定长度至少应等于 LEN 字节的长度。

输出参数 DONE 如为 TRUE , 则表明已将数据记录成功传送到 DP 从站。

输出参数 ERROR 用以指示是否发生数据记录传送错误。如果发生错误 , 则输出参数 STATUS 包含错误信息。

---

### 注意

如果使用 GSD 文件(版本 3 以上的 GSD)对 DPV1 从站进行组态 , 且 DP 主站的 DP 接口设置为 "S7 兼容" , 则一定不能在用户程序中通过 SFB53 将数据记录写入 I/O 模块。其原因就是此时的 DP 主站将寻址错误的插槽(已组态的插槽+3)。

纠正方法 : 将 DP 主站的接口设置为"DPV1"。

---

## 工作原理

SFB53 "WRREC" 以异步方式工作，即处理过程涉及多个 SFC 调用。通过使用 REQ = 1 调用 SFB53 来启动作业。

作业状态通过输出参数 BUSY 和输出参数 STATUS 的中间两个字节显示。此处，STATUS 的中间两个字节与异步运行 SFC 的输出参数 RET\_VAL 对应(参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

如果输出参数 BUSY = FALSE，则表明数据记录传送完成。

只要数据传输正在运行，目标范围就无法改变，无论是其长度还是内容。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	REQ=1：传送数据记录
ID	INPUT	DWORD	I、Q、M、D、L、常数	DP 从站/PROFINET IO 组件(模块或子模块)的逻辑地址 对于输出模块，必须设置位 15(例如对于地址 5：ID:=DW#16#8005)。 对于组合模块，应确定两个地址中的较小地址。
INDEX	INPUT	INT	I、Q、M、D、L、常数	数据记录号。
LEN	INPUT	INT	I、Q、M、D、L、常数	要传送的数据记录的最大字节长度
DONE	OUTPUT	BOOL	I、Q、M、D、L	数据记录已传送
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：写操作尚未结束。
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR = 1：发生写错误。
STATUS	OUTPUT	DWORD	I、Q、M、D、L	块状态或错误信息 若要解释 STATUS 参数，请参见使用 SFB 54 "RALRM"接收中断
RECORD	IN_OUT	ANY	I、Q、M、D、L	数据记录 注意：注意，S7-300 CPU 的 RECORD 参数总是需要完整指定 DB 参数(例如：P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成一条出错消息。

### 注意

如果使用 SFB 53 将数据记录写入 PROFINET IO，则 INDEX、MLEN 和 LEN 参数中的负值被解释为不带前缀符号的 16 位整数。

## 错误信息

关于 STATUS 参数的解释，请参见用 SFB 54 "RALRM"接收中断

## 8.3 用 SFB 54 "RALRM"接收中断

---

### 注意

SFB54 "RALRM"接口与"符合 IEC 61131-3 标准的 PROFIBUS 指南 PROFIBUS 通讯和代理功能块"标准中定义的 FB "RALRM"完全相同。

---

### 描述

SFB "RALRM"从 I/O 模块(集中式结构)或 DP 从站或 PROFINET IO 设备组件接收中断及所有相应的信息。然后将此信息提供给输出参数。

输出参数中的信息包括被调用 OB 的启动信息以及关于中断源的信息。

仅在由 CPU 操作系统启动中断 OB 中调用 SFB54，以便检查 I/O 中断。

---

### 注意

如果在 OB 中调用 SFB54 "RALRM"，而它的启动事件未由 I/O 触发，则 SFB 将相应地减少提供关于其输出的信息。

确保在不同的 OB 中调用 SFB54 "RALRM"时使用不同的背景数据块。如果要在相关中断 OB 外部计算 SFB54 调用结果数据，则对每个 OB 启动事件应该使用单独的实例 DB。

---

## SFB 54 调用

可以在三种工作模式(MODE)下调用 SFB54 "RALRM"。下表予以说明。

MODE	SFB54 ...
0	... 在输出参数 ID 中显示触发中断的组件并将输出参数 NEW 设置为 TRUE。
1	... 描述所有输出参数，与中断触发组件无关。
2	... 检查是否是输入参数 F_ID 中指定的组件触发了中断。 <ul style="list-style-type: none"> <li>• 如果未触发中断，则 NEW = FALSE</li> <li>• 如果已触发中断，则 NEW = TRUE，且描述所有其它输出参数</li> </ul>

参数	声明	数据类型	存储器区	描述
MODE	INPUT	INT	I、Q、M、D、L、 常数	工作模式
F_ID	INPUT	DWORD	I、Q、M、D、L、 常数	要从中接收中断的组件(模块)的逻辑起始地址 注意：如果地址是输出地址，则必须设置位 15。
MLEN	INPUT	INT	I、Q、M、D、L、 常数	要接收数据中断信息的最大长度(以字节计)
NEW	OUTPUT	BOOL	I、Q、M、D、L	已接收新中断。
STATUS	OUTPUT	DWORD	I、Q、M、D、L	SFB 或 DP 主站的出错代码
ID	OUTPUT	DWORD	I、Q、M、D、L	已从中接收中断的组件(模块)的逻辑起始地址。 位 15 包含 I/O ID：0 代表输入地址；1 代表输出地址
LEN	OUTPUT	INT	I、Q、M、D、L	已接收中断信息的长度
TINFO	IN_OUT	ANY	I、Q、M、D、L	(任务信息) OB 启动和管理信息的目标范围 注意：对于 S7-300 CPU，请注意，参数 TINFO 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。
AINFO	IN OUT	ANY	I、Q、M、D、L	(中断信息) 标题信息和附加中断信息的目标区域。 至少应为 AINFO 提供 MLEN 个字节。 注意：对于 S7-300 CPU，请注意，参数 TINFO 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。

---

**当心**

如果选择的目标区域 TINFO 或 AINFO 能容纳的字节数过少，则 SFB54 不能输入完整信息。

---

**目标区域 TINFO 的数据结构**

字节	含义
0 到 19	当前已调用 SFB54 的 OB 的启动信息： <ul style="list-style-type: none"><li>• 字节 0 到 11：结构类似于 SFC 6 "RD_SINFO"中的参数 TOP_SI</li><li>• 字节 12 到 19：请求 OB 的日期和时间</li></ul>
20 和 21	地址，有关确切描述，请参见下文
22 到 31	管理信息，有关确切描述，请参见下文



管理信息的结构(字节 20 到 25)

TINFO 的字节号	数据类型	含义						
20	BYTE	集中式 :	0					
		分布式 :	PROFIBUS DP : DP 主站系统 ID (可能的值为 1 到 32)					
			PROFINET IO : 参见上文					
21	BYTE	集中式 :	模块机架号(可能的值 : 0 至 31)					
		分布式 :	DP 站号(可能的值 : 0 到 127)					
			PROFINET IO : 参见上文					
22	BYTE	集中式 :	0: 数据记录 0 或数据记录 1					
		分布式	• 位 0 到 3 :	从站类型	0000:	DP (结构数据记录 0)		
					0001:	DPS7 (结构数据记录 0 或数据记录 1)		
					0010:	DPS7 V1 (结构数据记录 0 或数据记录 1)		
					0011:	DPV1 (结构符合 PROFIBUS DP 标准)		
					0100 - 0111:	保留		
					1000:	PROFINET IO (结构符合 PROFINET IO 标准)		
					自 1001 起 :	保留		
					• 4 到 7 位 :	配置文件类型		保留
23	BYTE	集中式 :	0					
		分布式 :	• 位 0 到 3 :	中断信息类型	0000:	透明, 对于 PROFINET IO 始终是这种情况 (中断源于已组态的分布式模块)		
					0001:	哑元 (中断源于一个非 DPV1 从站/非 IO 设备或未组态的插槽)		
					0010:	生成 (中断在 CPU 中生成)		
					自 0011 起 :	保留		
					• 位 4 到 7 :	结构版本	0000:	初始
	自 0001 起 :	保留						

TINFO 的字节号	数据类型	含义	
24	BYTE	集中式：	0
		分布式：	PROFIBUS DP 主站接口模块/PROFINET IO 控制器主站接口模块的标记
		• 位 0 = 0：	中断源于集成 DP 接口 (PROFINET IO 或 PROFIBUS DP)
		• 位 0 = 1：	中断源于外部 DP 接口模块 (PROFINET IO 或 PROFIBUS DP)
		• 位 1 到 7：	保留
25	BYTE	集中式：	0
		分布式：	PROFIBUS DP 从站接口模块的标记
		• 位 0：	诊断消息帧的 EXT_DIAG_FLAG 位或为 0 (如果中断中不存在此位) 如果 DP 从站出现故障，则该位为 1。
		• 位 1 到 7：	保留
			PROFINET IO 控制器接口模块的标记
		• 位 0：	ARDiagnosisstate 位或为 0 (如果中断中无信息) 如果 IO 设备出现故障，则该位为 1。
		• 位 1 到 7：	保留

### 管理信息的结构(字节 26 到 27，带有 PROFIBUS 和集中式组态)

TINFO 的字节号	数据类型	含义	
26 和 27	WORD	集中式：	0
	WORD	分布式：	作为 PROFIBUS DP 从站唯一标识符的 PROFIBUS ID 号
28 和 29	WORD	0	(字节 28 和 29 可以省略)
30 和 31	WORD	0	(字节 30 和 31 可以省略)

此为带有 PROFIBUS 和集中式组态的 TINFO 目标区域的管理信息的结尾。



### 管理信息的结构(字节 26 到 31 , 带有 PROFINET IO)

TINFO 的字节号	数据类型	含义	
26 和 27	WORD	分布式 :	作为 PROFINET IO 设备唯一标识符的 PROFINET IO 设备的 ID 号
28 和 29	WORD	分布式 :	制造商 ID
30 和 31	WORD	分布式 :	实例的 ID 号

### 目标区域 AINFO 的数据结构(带有来自 PROFIBUS DP 或集中式 IO 设备的中断)

下文提供有关 PROFINET IO 的更多信息。

字节	含义	
0 至 3	标题信息, 有关确切描述, 请参见下文	
4 到 199	附加中断信息 : 相应中断的数据 :	
	集中式 :	ARRAY[0]到 ARRAY[195]
	分布式 :	ARRAY[0]到 ARRAY[59]

### 标题信息的结构(带有来自 PROFIBUS DP 或集中式 IO 设备的中断)

字节	数据类型	含义	
0	BYTE	已接收中断信息的长度(以字节计)	
		集中式 :	4 到 224
		分布式 :	4 到 63
1	BYTE	本地 :	保留
		分布式 :	中断类型的 ID
			1: 诊断中断 2: 硬件中断 3: 删除中断 4: 插入中断 5: 状态中断 6: 更新中断 31 扩展设备、DP 主站系统或 DP 站发生故障
			32 到 126 制造商特定中断
2	BYTE	中断触发组件的插槽号	
3	BYTE	集中式 :	保留
		分布式 :	标识符
			位 0 和 1 0: 无更多信息 1: 即将发生的事件, 插槽已损坏 2: 离开的事件, 插槽不再损坏 3: 离开的事件, 插槽仍损坏
			位 2 : Add_Ack
			位 3 到 7 : 序号

## 目标区域 AINFO 的数据结构 (带有来自 PROFINET IO 的中断)

字节	含义
0 到 25	标题信息, 有关确切描述, 请参见下文
26 到 1431	附加中断信息: 各中断的标准诊断数据: ARRAY[0]到 ARRAY[1405] 注意: 也可以忽略该附加中断信息。

## 标题信息的结构 (带有来自 PROFINET IO 的中断)

字节	数据类型	含义
0 和 1	WORD	<ul style="list-style-type: none"> <li>位 0 到 7: 块类型</li> <li>位 8 到 15: 保留</li> </ul>
2 和 3	WORD	块长度
4 和 5	WORD	版本: <ul style="list-style-type: none"> <li>位 0 到 7: 低字节</li> <li>位 8 到 15: 高字节</li> </ul>
6 和 7	WORD	中断类型的标识符: <ul style="list-style-type: none"> <li>1: 诊断中断(进入)</li> <li>2: 硬件中断</li> <li>3: 删除中断</li> <li>4: 插入中断</li> <li>5: 状态中断</li> <li>6: 更新中断</li> <li>7: 冗余中断</li> <li>8: 由监控程序控制</li> <li>9: 由监控程序释放</li> <li>10: 未插入已组态模块</li> <li>11: 返回子模块</li> <li>12: 诊断中断(退出状态)</li> <li>13: 直接数据交换连接消息</li> <li>14: 邻近区域改变消息</li> <li>15: 时钟同步消息(来自总线)</li> <li>16: 时钟同步消息(来自设备)</li> <li>17: 网络组件消息</li> <li>18: 时间同步消息(来自总线)</li> <li>19 到 31: 保留</li> <li>32 到 127: 供应商特定中断</li> <li>128 到 65535: 保留</li> </ul>

字节	数据类型	含义
8 到 11	DWORD	API (应用程序进程标识符)
12 到 13	WORD	触发中断的组件的插槽号(取值范围为 0 到 65535)
14 到 15	WORD	触发中断的组件的子模块插槽号(取值范围为 0 到 65535)
16 到 19	DWORD	模块标识; 关于中断源的特定信息
20 到 23	DWORD	子模块标识符 ; 关于中断源的特定信息
24 到 25	WORD	中断说明符 : <ul style="list-style-type: none"> <li>• 位 0 到 10 : 序号(取值范围 : 0 到 2047)</li> <li>• 位 11 : 通道诊断 :                             <ul style="list-style-type: none"> <li>0: 无可用通道诊断</li> <li>1 : 有可用通道诊断</li> </ul> </li> <li>• 位 12 : 制造商特定诊断的状态 :                             <ul style="list-style-type: none"> <li>0: 无可用的制造商特定状态信息</li> <li>1 : 有可用的制造商特定状态信息</li> </ul> </li> <li>• 位 13 : 接口模块的诊断状态 :                             <ul style="list-style-type: none"> <li>0: 无可用状态信息 ; 已更正所有错误</li> <li>1 : 至少有一个通道的诊断和/或状态信息可用</li> </ul> </li> <li>• 位 14 : 保留</li> <li>• 位 15 : 应用程序关系诊断状态                             <ul style="list-style-type: none"> <li>- 0: 此 AR 中的任何一个已组态模块都未报告诊断</li> <li>- 1: 此 AR 中至少有一个已组态模块在报告诊断</li> </ul> </li> </ul>

### 来自 PROFINET IO 中断的附加中断信息的结构

PROFINET IO 的附加中断信息取决于格式标识符。它可由具有相同或不同格式 ID 的多个数据块组成。以下格式标识符可用 :

- W#16#0000 到 W#16#7FFF : 制造商特定的诊断。

字节	数据类型	含义
0 - 1	WORD	以下用作附加中断信息数据的结构的格式标识符 W#16#0000 到 W#16#7FFF : 制造商特定诊断
2 到 n	BYTE	参见制造商手册

- W#16#8000 : 通道诊断

通道诊断通过块输出 , 其中每个块包含 6 个字节。仅对于有故障的通道才输出附加中断信息(无格式标识符)。

字节	数据类型	含义
0 - 1	WORD	以下用作附加中断信息数据的结构的格式标识符 W#16#8000 : 通道诊断

字节	数据类型	含义	
2 到 3	WORD	触发中断的组件的通道号(取值范围：0 到 65535)： <ul style="list-style-type: none"> <li>W#16#0000 到 W#16#7FFF：接口模块/子模块的通道号</li> <li>W#16#8000：通常用于替代整个接口模块的通道号</li> <li>W#16#8001 到 W#16#FFFF：保留</li> </ul>	
4	BYTE	位 0 到 2	保留
		位 3 到 4	错误类型： <ul style="list-style-type: none"> <li>0: 保留</li> <li>1: 进入的错误</li> <li>2: 离开的错误</li> <li>3: 离开的错误，存在的其它错误</li> </ul>
		位 5 到 7：	通道类型： <ul style="list-style-type: none"> <li>0: 保留</li> <li>1: 输入通道</li> <li>2: 输出通道</li> <li>3: 输入/输出通道</li> </ul>
5	BYTE	数据格式： <ul style="list-style-type: none"> <li>B#16#00：自由数据格式</li> <li>B#16#01：位</li> <li>B#16#02：2 位</li> <li>B#16#03：4 位</li> <li>B#16#04：字节</li> <li>B#16#05：字</li> <li>B#16#06：双字</li> <li>B#16#07：2 个双字</li> </ul> B#16#08 到 B#16#FF：保留	
6 到 7	WORD	错误类型： <ul style="list-style-type: none"> <li>W#16#0000：保留</li> <li>W#16#0001：短路</li> <li>W#16#0002：欠电压</li> <li>W#16#0003：过电压</li> <li>W#16#0004：超载</li> <li>W#16#0005：温度过高</li> <li>W#16#0006：线路中断</li> <li>W#16#0007：高于上限值</li> <li>W#16#0008：低于下限值</li> <li>W#16#0009：错误</li> <li>W#16#000A 到 W#16#000F：保留</li> <li>W#16#0010 到 W#16#001F：制造商特定</li> <li>W#16#0020 到 W#16#00FF：保留</li> <li>W#16#0100 到 W#16#7FFF：制造商特定</li> <li>W#16#8000：可用设备诊断</li> <li>W#16#8001 到 W#16#FFFF：保留</li> </ul> 并非所有通道都支持每种错误类型。有关详细信息，请参见特定设备的诊断数据描述。	

**注意**

从"通道号"到"错误类型"这一部分可能会出现 0 到 n 次。

- W#16#8001 : MULTIPLE (传送不同类型的诊断信息)。  
这种情况下，将附加中断信息作为长度可变的块传送。

字节	数据类型	含义
0 - 1	WORD	以下用作附加中断信息数据的结构的格式标识符 W#16#8001 : 制造商特定诊断和/或通道诊断
2 到 3	WORD	块类型
4 到 5	WORD	块长度
6	BYTE	版本 : 高字节
7	BYTE	版本 : 低字节
8 到 11	DWORD	API (仅当版本低字节= 1 时)
12 到 13	WORD	插槽号
14 到 15	WORD	子插槽号
16 到 17	WORD	通道号
18 - 19	WORD	通道属性
20 - 21	WORD	格式标识符 : <ul style="list-style-type: none"> <li>• W#16#0000 到 W#16#7FFF : 制造商特定诊断</li> <li>• W#16#8000 : 通道诊断</li> <li>• W#16#8002 : 扩充的通道诊断</li> <li>• W#16#8003 : 步进式的扩充通道诊断</li> <li>• W#16#8004 到 W#16#80FF : 保留</li> </ul>
22 到 n	BYTE	数据取决于格式标识符

**注意**

从"块类型"开始的部分可能出现 0 到 n 次。

- W#16#8002 : 扩充的通道诊断

字节	含义
0 - 1	格式 ID W#16#8002
2 到 3	通道号
4 到 5	通道属性
6 到 7	错误类型
8 到 9	附加错误值
10 - 13	附加错误信息

- W#16#8003 : 步进式的扩充通道诊断

字节	含义
0 - 1	格式 ID W#16#8003
2 到 3	通道号
4 到 5	通道属性
6 到 7	错误类型
8 到 9	附加错误值
10 - 13	附加错误信息
14 - 17	合格的通道限定符

- W#16#8100 : 维护信息

字节	含义
0 - 1	格式 ID W#16#8100
2 到 3	块类型
4 到 5	块长度
6 到 7	块版本
8 到 9	保留
10 - 13	维护状态

### 注意

可以在从 *PROFIBUS DP 到 PROFINET IO 的 SIMATIC PROFINET IO 编程手册* 以及当前版本的 IEC 61158-6-10-1 标准中找到更多关于组态附加报警信息的详细信息。

## 目标区域 TINFO 和 AINFO

根据调用 SFB54 的各个 OB 的不同，可能仅会部分改写目标区域 TINFO 和 AINFO。关于分别输入哪些信息，请参考下表。

中断类型	OB	TINFO OB 状态信息	TINFO 管理信息	AINFO 标题信息	AINFO 附加中断信息	
硬件中断	4x	是	是	是	集中式：	否
					分布式：	与 PROFIBUS DP 从站 /PROFINET IO 设备传 送的内容相同
状态中断	55	是	是	是	是	是
更新中断	56	是	是	是	是	是

中断类型	OB	TINFO OB 状态信息	TINFO 管理信息	AINFO 标题信息	AINFO 附加中断信息	
制造商特定中断	57	是	是	是	是	是
I/O 冗余出错	70	是	是	否	否	否
诊断中断	82	是	是	是	集中式：	数据记录 1
					分布式：	与 PROFIBUS DP 从站 /PROFINET IO 设备传 送的内容相同
删除/插入中断	83	是	是	是	集中式：	否
					分布式：	与 PROFIBUS DP 从站 /PROFINET IO 设备提 供的内容相同
删除中断的特殊 格式： 由监控程序控制	83	是	是	是	仅适用于 PROFINET IO	
插入中断的特殊 格式： 由监控程序控制	83	是	是	是	仅适用于 PROFINET IO	
插入未组态模块	83	是	是	是	仅适用于 PROFINET IO	
模块机架/站故障	86	是	是	否	否	
... 所有其它 OB		是	否	否	否	

## 错误信息

输出参数 STATUS 包含一些信息。它被解释为 ARRAY[1...4] OF BYTE，错误信息的结构如下：

域元素	名称：	含义
STATUS[1]	Function_Num	<ul style="list-style-type: none"> <li>B#16#00(如果无错误)</li> <li>DPV1-PDU 的功能 ID：如果在读取数据记录 B#16#DE 时出现一个错误，如果在写入数据记录 B#16#DF 时出现一个错误。如果未使用 DPV1 协议单元：B#16#C0。</li> </ul>
STATUS[2]	Error Decode	错误 ID 的位置
STATUS[3]	Error_Code_1	错误 ID
STATUS[4]	Error_Code_2	制造商特定错误 ID 扩展

STATUS[2]可具有以下值：

Error Decode (B#16#...)	源	含义
00 到 7F	CPU	无错误或无警告
80	DPV1	因不符合 IEC 61158-6 而出错
81 到 8F	CPU	B#16#8x 显示在 SFB 的第 n 个调用参数中存在错误。
FE、FF	DP 配置文件	配置文件特定错误

STATUS[3]可具有以下值：

Error Decode (B#16#...)	Error_Code_1 (B#16#...)	按照 DVP1 的解释	含义
00	00		无错误，无警告
70	00	保留，拒绝	初始调用；未传送活动数据记录
	01	保留，拒绝	初始调用；已开始传送数据记录
	02	保留，拒绝	中间调用；已激活数据记录传送
80	90	保留，通过	逻辑起始地址无效
	92	保留，通过	ANY 指针的非法类型
	93	保留，通过	通过 ID 或 F_ID 寻址的 DP 组件未组态。
	95		获取附加中断信息时在 H 系统中出错(通过外部 DP 接口在本地或分布式 I/O 获取附加中断信息时，此错误输出为"组错误"。) 注意：连接或更新时，附加中断信息可能会暂时不可用。



Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	按照 DVP1 的解释	含义
	96		H 系统中发生主-备机切换，先前的主 CPU 已进入 STOP 模式。当时正在处理一个 OB。SFB 54 不能提供 OB 启动信息、管理信息、标题信息或附加中断信息。 可使用 SFC 6 "RD_SINFO"读出 OB 启动信息。此外，还可以使用 SFC 13 "DPNRM_DG"同步读取 OB 4x、55、56、57、82 和 83 的受影响 DP 从站的当前诊断帧(地址信息来自 OB 启动信息)。
	A0	读错误	读模块时出现否定应答。
	A1	写错误	写模块时出现否定应答
	A2	模块故障	层面 2 上的 DP 协议错误(例如，从站故障或总线问题)
	A3	保留，通过	<ul style="list-style-type: none"> <li>PROFIBUS DP：直接数据链路映射或用户接口/用户出现 DP 协议错误</li> <li>PROFINET IO：常规 CM 错误</li> </ul>
	A4	保留，通过	总线通讯中断
	A5	保留，通过	-
	A7	保留，通过	DP 从站或模块已被占用(临时性错误)
	A8	版本冲突	DP 从站或模块报告出现不兼容版本
	A9	功能不被支持	DP 从站或模块不支持该功能
	AA 到 AF	用户特定	DP 从站或模块报告其应用程序中出现制造商特定错误。请检查制造商提供的 DP 从站或模块文档。
	B0	索引无效	数据记录在模块中为未知 数据记录号 ≥256，非法
	B1	写长度错误	在参数 RECORD 中指定的长度错误； 对于 SFB54：AINFO 中出现长度错误 对于 SFB52 和 53：MLEN 中出现长度错误
	B2	插槽无效	未使用已组态的插槽。
	B3	类型冲突	实际模块类型与指定模块类型不同
	B4	区域无效	DP 从站或模块报告访问的是无效区域
	B5	状态冲突	DP 从站或模块未就绪
	B6	访问被拒绝	DP 从站或模块拒绝访问
	B7	范围无效	DP 从站或模块报告参数或值的范围无效
	B8	参数无效	DP 从站或模块报告参数无效
	B9	类型无效	DP 从站或模块报告类型无效 对于 SFB52：缓冲区太小(无法读取子集) 对于 SFB53：缓冲区太小(无法写入子集)

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	按照 DVP1 的解释	含义
	BA 到 BF	用户特定	DP 从站或模块报告在访问时出现制造商特定错误。请检查制造商提供的 DP 从站或模块文档。 关于值 B#16# BA 的注意事项：对于容错系统中的 PROFINET，适用于以下情况：如果返回值 W#16#80BA 的数据记录作业遭拒绝，则必须重复作业。
	C0	读限制冲突	对于 SFB 53 "WRREC"：只有在 CPU 处于 STOP 模式时才能写入数据。注意：也就是说无法由用户程序写入数据。只能通过 PG/PC 在线写入数据。 对于 SFB 52 "RDREC"：模块具有数据记录，但数据不可用或只能在 CPU 处于 STOP 模式才能读取数据。注意：若只有在 CPU 处于 STOP 模式时才能读取数据，则无法由用户程序进行判断。这种情况下，只能通过 PG/PC 在线读取数据。
	C1	写限制冲突	针对模块的相同数据记录的上一次写请求数据尚未被该模块处理。
	C2	资源忙	模块当前正在处理 CPU 允许的最大可能作业数。
	C3	资源不可用	所需操作资源当前已被占用。
	C4		内部暂时故障。无法执行作业。 重复执行该作业。如果此错误频繁发生，请检查设备的电气干扰源。
	C5		DP 从站或模块不可用。
	C6		由于取消优先级而使数据记录传送被取消
	C7		由于重新启动(热启动)或冷启动 DP 主站而使作业被取消
	C8 到 CF		DP 从站或模块报告制造商特定资源错误。请检查制造商提供的 DP 从站或模块文档。
	Dx	用户特定	DP 从站特定。参考 DP 从站的描述。
8x (x = 1, ... 9, A, B, C, D, E, F)	00 到 FF		在第 y 次调用参数(y = 1, ...15)时发生错误 • Error_Code_1 = 00：非法模式 对于 Error_Code_1 的所有其它数值，请参考"常规错误代码"，见使用输出参数 RET_VAL 评估错误
FE、FF	00 到 FF		配置文件特定错误

出现 DPV1 错误时，DP 主站会将 STATUS[4]传递给 CPU 和 SFB。如果不出现 DPV1 错误，则此值被设置为 0，但对于 SFB 52 和 53 则会出现以下例外情况：

- STATUS[4]包含 RECORD 中的目标区域长度(如果 LEN > RECORD 中的目标区域长度)
- STATUS[4]=LEN(如果实际数据记录长度 < LEN < RECORD 中的目标区域长度)
- STATUS[4]=0(如果必须设置 STATUS[4] > 255)

出现 DPV1 或 PNIO 错误时，传递协议单元(0xDE, xx, xx, xx)或(0xDF, xx, xx, xx)。出现内部创建的错误时(0xC0, xx, xx, xx)，即不存在协议单元时，最后一个字节等于零。

## 8.4 用 SFB 75 "SALRM"向 DP 主站发送中断

### 注意

SFB 75 "SALRM"接口与在标准"符合 IEC 61131-3 标准的 PROFIBUS 指南 PROFIBUS 通讯和代理功能块"中定义的 FB "SALRM"完全相同。

### 描述

SFB 75 "SALRM"从智能从站的用户程序向相关 DP 主站发送一个插槽中断,该 DP 主站在过渡区域(虚拟插槽)中访问该中断。这将启动 DP 主站中的相关 OB。

可以与中断一起发送附加中断特定信息。在 DP 主站中,可以使用 SFB 54 "RALRM"读取所有附加信息。

SFB 75 只能用在 S7 兼容模式中。

- DP : 主站使用 GSD 文件集成 I 从站。
- S7 兼容 : 在 STEP 7 中,通过 HW 组态将 I 从站与一个主站链接在一起。

### 工作原理

SFB 75 "SALRM"是异步 SFB,这表示它的处理涉及多个 SFB 调用。通过使用 REQ = 1 调用 SFB 75 来启动中断的传送。

在 DP 主站确认或取消中断处理之前,发送过程始终保持活动状态。

输出参数 BUSY 与输出参数 STATUS 的字节 2 和 3 指示作业状态,其中 STATUS 字节 2 和 3 对应于异步工作 SFC 的 RET\_VAL 输出参数(参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)

如果输出参数 BUSY = FALSE,则中断传送完成。

## 识别作业

当使用 SFB 75 向 DP 主站启动中断传送后，如果在完成当前作业前再次调用此 SFB，则此 SFB 的进一步响应将取决于新调用是否与前一个调用完全相同。

如果参数 ID 和 TYPE 与尚未完成的作业相匹配，则将会按顺序处理对 SFB 的新调用。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	REQ=1：传送中断
ID	INPUT	DWORD	I、Q、M、D、L、常数	DP 主站的过渡区域(虚拟插槽)的所有逻辑地址，与在 DP 从站中看到的相同(站的诊断地址和插槽 2 的逻辑地址除外)。请在低字中查找相关信息。在高字中输入零。位 15 包含 I/O ID：0 = 输入地址，1 = 输出地址。
ATYPE	INPUT	INT	I、Q、M、D、L、常数	中断类型 中断类型的 ID。可能的值： <ul style="list-style-type: none"> <li>• 1: 诊断中断</li> <li>• 2: 硬件中断</li> </ul>
ASPEC	INPUT	INT	I、Q、M、D、L、常数	中断标识符： <ul style="list-style-type: none"> <li>• 0: 无更多信息</li> <li>• 1: 进入的事件，插槽有故障</li> <li>• 2: 离开的事件，插槽不再有故障故障</li> <li>• 3: 离开的事件，插槽仍存在故障</li> </ul>
LEN	INPUT	INT	I、Q、M、D、L、常数	要发送的附加中断信息的长度(以字节计) 最大值：16
DONE	OUTPUT	BOOL	I、Q、M、D、L、	DONE=1：中断已传送
BUSY	OUTPUT	BOOL	I、Q、M、D、L、	BUSY = 1：中断传送仍忙
ERROR	OUTPUT	BOOL	I、Q、M、D、L、	ERROR = 1：发生错误。
STATUS	OUTPUT	DWORD	I、Q、M、D、L、	错误信息
AINFO	IN_OUT	ANY	I、Q、M、D、L、	中断信息 附加中断信息的源区域 注意：对于 S7-300 CPU，请注意，参数 TINFO 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。

## ATYPE 参数

下表显示 ATYPE 的所有允许值，其中 OB 是在相关主站中启动的，且在 DP 模式中，允许使用给定的中断类型。

ATYPE	DPV1 标准中的含义	S7 DP 主站中的相关 OB	DP 模式	
			DP	S7 兼容
1	诊断中断	诊断中断 OB (OB82)	-	是
2	硬件中断	硬件中断 OB (OB 40 到 47)	-	是

### 注意

除上表提供的信息外，中断类型的使用范围还受到 DP 主站的限制。

## 中断类型与 S7 主站工作模式的依存关系

对于 S7 兼容模式下的从站(在 S7 主站上工作)，仅当相关 DP 主站处于 RUN 模式(DP：工作)下时才能发送硬件和诊断中断。如果 DP 主站处于 STOP 模式(DP：清除)，中断将被禁用，SFB 75 将返回错误信息 W#16#80C8。这种情况下，由用户负责在以后发送中断。

## ASPEC 参数

按照适用标准，此参数将显示虚拟插槽的诊断状态。因此，仅当在发送诊断中断时，才能为 ASPEC 分配非零值。

由于诊断中断的 S7 附加中断信息(数据记录 0)包含进入/离开信息(参见诊断数据字节 0 中的位 0)，必须如下描述附加中断信息字节 0 中的位 0 (模块故障)：

ASPEC	AINFO 中的"模块故障"位
0	0
1	1
2	0
3	1

## LEN 参数

LEN 参数用于确定要发送的附加中断信息的长度(以字节计)。允许的最大取值范围是 0 到 16。

对于所有可能的中断类型，下表显示在智能从站各种模式中，LEN 的取值。

中断类型	DP	S7 兼容
诊断中断	-	4 到 16
硬件中断	-	4

下表显示为 LEN 分配一个不等于 AINFO 长度的值(以字节计)后，SFB 75 的响应。

LEN 的值	SFB 75 响应
<= 为 AINFO 指定的长度	SFB 75 向 DP 主站发送一个中断。传送的附加中断信息的字节数与在 LEN 中指定的字节数相同。
超出允许的取值范围(< 0 或 > 16)	SFB 75 不发送中断。 错误信息：W#16#80B1，STATUS[4]=B#16#FF
> 为 AINFO 指定的长度	SFB 75 向 DP 主站发送一个中断。传送的附加中断信息的字节数与在 AINFO 长度信息中指定的字节数相同。 错误信息：W#16#00B1，STATUS[4]= AINFO 中的长度信息

## AINFO 参数

AINFO 是附加中断信息的源区域。就智能从站而言，可以用想要的任何值填充此区域。但是，如果您使用的是 S7 系列的 DP 主站，则随中断一起发送的附加信息必须符合 S7 惯例。

如果要发送**诊断中断**(ATYPE=1)，则必须在数据记录 0 中输入正确的相关值，如必要还要在数据记录 1 中输入正确的相关值。

下表是向您建议的一组 S7 兼容条目。此时已设置"模块故障"位(参见上文)。在实际命名该位之前，此建议与缺省分配(恢复的智能从站或站在加电或者从 STOP 转换到 RUN 后的现有分配)保持一致。

数据记录编号	分配
0	B#16#01、0B、00、00
1	对于 S7 兼容模式：数据记录 0 + 12 个字节(含零)

(参见：诊断数据结构概述)

## SFB75 调用对模块状态信息和组错误 LED (SF)的影响

对于所有 CPU，智能从站的插槽属性都存储在模块状态信息中(参见 SSL-ID W#16#xy91 - 模块状态信息)

使用 SFB 75 发送诊断中断时，智能从站的操作系统会对基于 AINFO 字节 0 中的位 0(此位已作为"模块故障"添加到状态信息中)的从站本地模块状态信息和 SF LED 产生影响。但 I 从站中未创建任何诊断缓冲区条目，并且未启动诊断中断 OB。

## DP 主站与 I 从站之间模块状态信息的一致性

下文介绍了各种情况，并论述了它们对模块状态信息的影响：

- 站恢复(导致在 DP 主站和 I 从站中启动 OB 86)  
这将对 I 从站和 S7 主站中的模块状态信息产生影响(将复位"模块故障")。如果在站恢复后，从用户角度来看 I 从站中存在故障，则必须通过 SFB 75 调用将这些故障报告给 DP 主站。
- DP 主站中 STOP 到 RUN 工作模式的过渡(导致 I 从站中启动 OB 82)  
I 从站中的模块状态信息保持不变。DP 主站将复位相关模块状态信息中的"模块故障"位。

为确保在 S7 兼容模式下 DP 主站和 I 从站之间的模块状态信息能保持一致，必须按如下所述对 I 从站重新操作：

- 对于每个无错的虚拟插槽，使用 SFB 75 向 DP 主站发送一个离开的诊断中断。
  - 对于每个有故障的虚拟插槽，使用 SFB 75 向 DP 主站发送一个进入的诊断中断。
- 
- I 从站中 STOP 到 RUN 工作模式的过渡(导致 DP 主站中启动 OB 82)  
DP 主站中的模块状态信息保持不变；将复位 I 从站中的信息(复位"模块故障")。  
为确保在 S7 兼容模式下 DP 主站和 I 从站之间的模块状态信息能保持一致，必须按如下所述对 I 从站重新操作：

- 对于每个无错的虚拟插槽，使用 SFB 75 向 DP 主站发送一个离开的诊断中断。
- 对于每个有故障的虚拟插槽，使用 SFB 75 向 DP 主站发送一个进入的诊断中断。

---

### 注意

由于 SFB 75 功能异步运行，无法在启动 OB 过程中结束 SFB 75 调用。换言之，在完成调用之前，这些调用必须在循环程序中运行。

---

---

### 注意

上文提到的主站和 I 从站模块状态信息之间的所有差别只能发生在通过 SFB 75 接收诊断中断的那些插槽上。也就是说上文论述的纠正方法仅适用于此类插槽。

---

## 错误信息

STATUS 输出参数包含错误信息。如果被解释为 ARRAY[1 ...4] OF BYTE，则此错误信息的结构将如下表所示：

域元素	含义
STATUS[1]	<ul style="list-style-type: none"> <li>B#16#00：无错误</li> <li>B#16#C0：I 从站检测到的错误</li> </ul>
STATUS[2]、STATUS[3]	对应于 SFC 的 RET_VAL 输出参数
STATUS[4]	B#16#00，LEN 和 AINFO 之间的一些长度冲突除外。这些例外显示在下表中。

下表显示 SFB 75 的所有特定错误信息。

ERROR	STATUS[2,3] (W#16# ...)	解释
0	0000	作业已完成，且没有出错。如果 LEN < AINFO 的长度，则只能传送附加中断信息中的 LEN 个字节。
0	00B1	LEN > AINFO 的长度。作业已完成。已传送 AINFO 中的附加中断信息。STATUS[4] 将接收 AINFO 的长度。
0	7000	REQ=0 的初始调用(空循环)。未发送中断。BUSY 的值为 0。
0	7001	REQ=1 的第一次调用。作业已启动。BUSY 的值为 1。
0	7002	中间调用(REQ 不相关)。发送的中断尚未得到 DP 主站的确认。BUSY 的值为 1。
1	8090	在 ID 中指定的地址超出允许的地址范围或者未对其进行组态。
1	8091	<ul style="list-style-type: none"> <li>您已在组态中禁用该中断。</li> <li>此类从站不允许使用该中断。</li> </ul>
1	8092	AINFO 中的数据类型无效(BYTE 和 BLOCK-DB 是有效类型)
1	8093	ID 属于虚拟插槽。不能从其中请求任何中断。
1	80B0	ASPEC <ul style="list-style-type: none"> <li>与 AINFO 的字节 0 中的位 0 不匹配</li> <li>使用的中断类型的值必须为 0</li> <li>超出允许的取值范围</li> </ul>
1	80B1	LEN 超出允许的取值范围。STATUS[4]接收 B#16#FF。
1	80B5	不允许在 DP 主站调用 SFB 75。
	80C3	所需资源(内存等)此时已被占用。
1	80C5	分布式 I/O 设备此时不可用(即站出现故障)
1	80C8	在当前 DP 主站工作模式下不允许使用该功能(DP 主站是 S7 主站，并且处于 STOP 模式下)。



## 9 用于处理时间中断的 SFC

### 9.1 处理时间中断

#### 定义

时间中断将导致操作系统调用一个时间中断 OB (OB10 到 OB17)。

#### 调用条件

必须满足下列条件，操作系统才能够调用时间中断 OB：

- 必须使用下列工具为时间中断 OB 分配了参数(启动日期和时间、执行)
  - STEP 7 或
  - SFC 28 "SET\_TINT" (在用户程序中)。
- 必须使用下列工具激活时间中断 OB
  - STEP 7 或
  - SFC 30 "ACT\_TINT" (在用户程序中)。
- 不得使用 STEP 7 取消选择时间中断 OB。
- 时间中断 OB 必须存在于 CPU 中。
- 如果通过 SFC 30 "ACT\_TINT"设置中断，且指定 OB 执行**仅一次**，则当前时间必须尚未超过启动日期和时间。如果选择**定期**执行，则将在下一个周期结束(启动时间 + 指定周期时间的倍数)时调用时间中断 OB。

#### 提示

可以使用 STEP 7 为时间中断分配参数，然后在用户程序(SFC 30 "ACT\_TINT")中激活中断。

### SFC 28 到 SFC 31 的用途

以下各部分所介绍的系统功能 SFC 28 到 SFC 31 的用途如下：

- 设置时间中断(SFC 28 "SET\_TINT")
- 取消时间中断(SFC 29 "CAN\_TINT")
- 激活时间中断(SFC 30 "ACT\_TINT")
- 查询时间中断(SFC 31 "QRY\_TINT")

## 9.2 SFC 28 到 31 的特征

### 状况与影响...

下表列出一些不同的状况，并说明它们对时间中断的影响。

如果...	那么...
设置时间中断(通过调用 SFC 28; SET_TINT)	将取消当前的时间中断。
取消时间中断(通过调用 SFC 29; CAN_TINT)	将清除启动日期和时间。必须重新设置时间中断，然后才能将其激活。
调用不存在的时间中断 OB。	将自动生成优先级错误，这意味着操作系统将调用 OB85。 如果 OB85 不存在，CPU 将切换到 STOP 模式。
如果将实时时钟同步或向前调整时钟	如果因为向前调整了时间而跳过启动日期/时间： <ul style="list-style-type: none"> <li>• 操作系统将调用 OB80<sup>1</sup>。</li> <li>• 调用 OB80 之后，操作系统还将调用未在 OB80<sup>2</sup> 中操作的每个被跳过的时间中断 OB(调用一次，无论跳过多少个周期)。</li> </ul> 如果 OB80 不存在，CPU 将切换到 STOP 模式。
如果将实时时钟同步或向后调整时钟	S7-400-CPU 和 CPU 318： 如果在时钟向后调整的这段时间里已调用时间中断 OB，则在第二次到达该时间时不会再调用它们。 S7-300-CPU：执行所有受影响的时间中断 OB。

- 1) OB80 包含编码的启动事件信息，用以指示哪些时间中断 OB 因向前调整了时钟而无法调用。启动事件信息中的时间对应于向前调整的时间。
- 2) 在跳过后被激活的时间中断的启动事件信息中的时间对应于第一个被跳过的时间中断的启动时间。

### 暖启动或冷启动

暖启动或冷启动过程中，用户程序中通过 SFC 指定的所有时间中断设置均将被清除。

然后，使用 STEP 7 设置的参数将生效。

### 执行时间中断 OB

下表列出了"执行"参数的不同影响。可以通过 STEP 7 或 SFC 28 "SET\_TINT"设置此参数(输入参数 PERIOD)。

时间中断 OB 的执行	响应
无 (只能通过 STEP 7 设置)	即使 CPU 中存在时间中断 OB，也会不执行。 可以在用户程序中使用 SFC 28 "SET_TINT"重新分配参数(设置时间中断)。
单次	调用时间中断 OB 后，将取消时间中断。然后，可以重新设置和激活此中断。
定期 (每分钟、每小时、每天、每周、每月、每年)	如果激活中断时启动日期和时间已过，则时间中断 OB 将在下一可能点 ("启动日期/时间 + 所选周期的倍数")上中断循环程序。 在极少数情况下，当再次调用时间中断 OB 时，其处理可能尚未完成。 结果： <ul style="list-style-type: none"> <li>• 时间错误(操作系统将调用 OB80；如果 OB80 不存在，则 CPU 将切换到 STOP 模式)。</li> <li>• 将在以后执行时间中断 OB。</li> </ul>

## 9.3 使用 SFC 28 "SET\_TINT"设置时间中断

### 描述

通过 SFC 28 "SET\_TINT" (设置时间中断)，可以设置时间中断组织块的启动日期和时间。将忽略指定的启动时间的秒和毫秒值，并将其设置为 0。

参数	声明	数据类型	存储器区	描述
OB_NR	INPUT	INT	I、Q、M、D、L、常数	在时间 SDT + PERIOD 的倍数处启动的 OB 的编号(OB10 到 OB17)。
SDT	INPUT	DT	D、L	启动日期和时间：将忽略指定的启动时间的秒和毫秒值，并将其设置为 0。 如果要设置每月启动时间中断 OB，则只能使用日期 1、2、...28 号作为启动日期。
PERIOD	INPUT	WORD	I、Q、M、D、L、常数	从启动点 SDT 开始的周期： W#16#0000 = 一次 W#16#0201 = 每分钟 W#16#0401 = 每小时 W#16#1001 = 每日 W#16#1201 = 每周 W#16#1401 = 每月 W#16#1801 = 每年 W#16#2001 = 月末
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能处于激活状态时出错，则 RET_VAL 的实际参数将包含错误代码。

### 错误信息

错误代码 (W#16#...)	解释
0000	未出错
8090	不正确的参数 OB_NR
8091	不正确的参数 SDT
8092	不正确的参数 PERIOD
80A1	设置的启动时间已过。(只有在 PERIOD= W#16#0000 时才出现该错误代码。)
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 9.4 使用 SFC 29 "CAN\_TINT"取消时间中断

### 描述

使用 SFC 29 "CAN\_TINT" (取消时间中断)，可以取消一个已激活的时间中断组织块

参数	声明	数据类型	存储器区	描述
OB_NR	INPUT	INT	I、Q、M、D、L、常数	OB 的编号，将在其中取消启动日期和时间 (OB10 到 OB17)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能处于激活状态时出错，则 RET_VAL 的实际参数将包含错误代码。

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	不正确的参数 OB_NR
80A0	没有为时间中断 OB 指定启动日期/时间
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 9.5 使用 SFC 30 "ACT\_TINT"激活时间中断

### 描述

使用 SFC 30 "ACT\_TINT" (激活时间中断), 可以激活一个时间中断组织块。

参数	声明	数据类型	存储器区	描述
OB_NR	INPUT	INT	I、Q、M、D、L、常数	将激活的 OB 的编号(OB10 到 OB17)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能处于激活状态时出错, 则 RET_VAL 的实际参数将包含错误代码。

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	不正确的参数 OB_NR。
80A0	没有为相应的时间中断 OB 设置启动日期/时间。
80A1	激活的时间已过。只有当选择"执行 = 一次"时才会出现此错误。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误

## 9.6 使用 SFC 31 "QRY\_TINT"查询时间中断

### 描述

使用系统功能 SFC 31 "QRY\_TINT" (查询时间中断), 可以在输出参数 STATUS 上显示时间中断组织块的状态。

参数	声明	数据类型	存储器区	描述
OB_NR	INPUT	INT	I、Q、M、D、L、常数	将被查询状态的 OB 的编号(OB10 到 OB17)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能处于激活状态时出错, 则 RET_VAL 的实际参数将包含错误代码。
STATUS	OUTPUT	WORD	I、Q、M、D、L	时间中断的状态; 参见下表

### 输出参数 STATUS

#### 注意

如果 RET\_VAL 包含非零值, 则不得判断 STATUS 状态。

0	<ul style="list-style-type: none"> <li>=0 : CPU 处于 RUN 模式。</li> <li>=1 : CPU 正在启动。</li> </ul>
1	<ul style="list-style-type: none"> <li>=0 : 中断已启用。</li> <li>=1 : 中断已通过调用 SFC39 "DIS_IRT"被禁用。</li> </ul>
2	<ul style="list-style-type: none"> <li>=0 : 中断未激活或已过期。</li> <li>=1 : 中断已激活。</li> </ul>
3	始终为 0
4	<ul style="list-style-type: none"> <li>=0 : 编号为 OB_NR 的 OB 不存在。</li> <li>=1 : 已加载编号为 OB_NR 的 OB。</li> </ul>
其它	始终为 0



**错误信息**

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	不正确的参数 OB_NR
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误



# 10 用于处理延时中断的 SFC

## 10.1 处理延时中断

### 定义

调用 SFC 32 "SRT\_DINT"后，操作系统将在指定的延迟时间过去后生成一个中断，也就是说，调用所选择的延时中断 OB。此中断被称为延时中断。

### 调用条件

必须满足下列条件，操作系统才能调用延时中断 OB：

- 必须通过 SFC 32 "SRT\_DINT"启动延时中断 OB。
- 不得使用 STEP 7 取消选择延时中断 OB。
- 延时中断 OB 必须存在于 CPU 中。

### SFC 32 到 SFC 34 的用途

以下各部分所介绍的系统功能 SFC 32 到 SFC 34 的用途如下：

- 启动延时中断(SFC 32 "SRT\_DINT")
- 取消延时中断(SFC 33 "CAN\_DINT")
- 查询延时中断(SFC 34 "QRY\_DINT")。

### 状况与影响...

下表列出了一些不同状况，并说明了它们对延时中断的影响。

如果...	且...	那么...
延时中断被启动(通过调用 SFC 32 "SRT_DINT")。	该延时中断已经启动。	将重写延迟时间，并重新启动延时中断。
	在调用时延时中断 OB 不存在。	操作系统将生成优先级错误(调用 OB85)。如果 OB85 不存在，CPU 将切换到 STOP 模式。
	CPU 切换为 RUN 模式前，已在启动 OB 中启动中断且延迟时间已过。	将延迟对延时中断 OB 的调用，直到 CPU 处于 RUN 模式。
延迟时间已过。	先前启动的延时中断 OB 仍在执行。	操作系统将生成时间错误(调用 OB80)。如果 OB80 不存在，CPU 将切换到 STOP 模式。

### 暖启动和冷启动

暖启动或冷启动期间，用户程序中使用 SFC 指定的所有延时中断设置均将被清除。

### 在启动 OB 中启动

可以在启动 OB 中启动延时中断。要调用延时中断 OB，必须满足下列两个条件：

- 延迟时间必须已过。
- CPU 必须处于 RUN 模式。

如果延迟时间已过而 CPU 未处于 RUN 模式，则延时中断 OB 调用将被延迟，直到 CPU 处于 RUN 模式。然后，将在执行 OB1 中的第一条指令前调用延时中断 OB。

## 10.2 使用 SFC 32 "SRT\_DINT"启动延时中断

### 描述

通过 SFC 32 "SRT\_DINT" (启动延时中断),可以在延迟时间过去(参数 DTIME)后立即启动调用延时中断组织块的延时中断。

使用 SIGN 参数,可以输入用于标识延时中断开始的标识符。执行指定的 OB 时,DTIME 和 SIGN 值将再次显示在该 OB 的启动事件信息中。

参数	声明	数据类型	存储器区	描述
OB_NR	INPUT	INT	I、Q、M、D、L、常数	将在延时后启动的 OB 的编号 (OB20 到 OB23)。
DTIME	INPUT	TIME	I、Q、M、D、L、常数	延时值(1 到 60000 毫秒) 要实现更长时间的延时,例如,可以通过在延时中断 OB 中使用计数器来实现。
SIGN	INPUT	WORD	I、Q、M、D、L、常数	调用延时中断 OB 时将显示在启动事件信息中的标识符。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在系统功能处于激活状态时出错,则 RET_VAL 的实际参数将包含错误代码。

### 精度

只要没有中断事件延迟该调用,调用 SFC 32 "SRT\_DINT"和启动延时中断 OB 之间的时间最多比所选时间少一毫秒。

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	不正确的参数 OB_NR
8091	不正确的参数 DTIME
8xyy	常见出错信息,请参见通过输出参数 RET_VAL 判断错误

## 10.3 使用 SFC 34 "QRY\_DINT"查询延时中断

### 描述

通过 SFC 34 "QRY\_DINT" (查询延时中断), 可以查询延时中断 OB 的状态。延时中断由组织块 OB20 到 OB23 管理。

参数	声明	数据类型	存储器区	描述
OB_NR	INPUT	INT	I、Q、M、D、L、常数	将查询其 STATUS 的 OB 的编号(OB20 到 OB23)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在处理功能期间出错, 则 RET_VAL 的实际参数将包含错误代码。
STATUS	OUTPUT	WORD	I、Q、M、D、L	有关延时中断的状态, 参见下表。

### 输出参数 STATUS

#### 注意

如果 RET\_VAL 包含非零值, 则不得判断 STATUS 状态。

0	<ul style="list-style-type: none"> <li>=0 : CPU 处于 RUN 模式。</li> <li>=1 : CPU 正在启动。</li> </ul>
1	<ul style="list-style-type: none"> <li>=0 : 中断已启用。</li> <li>=1 : 中断已通过调用 SFC39 "DIS_IRT"被禁用。</li> </ul>
2	<ul style="list-style-type: none"> <li>=0 : 中断未激活或已过期。</li> <li>=1 : 中断已激活。</li> </ul>
3	始终为 0
4	<ul style="list-style-type: none"> <li>=0 : 编号为 OB_NR 的 OB 不存在。</li> <li>=1 : 已加载编号为 OB_NR 的 OB。</li> </ul>
其它	始终为 0

### 错误信息

错误代码 (W#16#...)	解释
0000	未出错
8090	不正确的参数 OB_NR
8xyy	常规错误信息, 通过输出参数 RET_VAL 判断错误

## 10.4 使用 SFC 33 "CAN\_DINT"取消延时中断

### 描述

通过 SFC 33 "CAN\_DINT" (取消延时中断), 可以取消已启动的延时中断(参见使用 SFC 32 "SRT\_DINT"启动延时中断)。这样, 将不调用延时中断 OB。

参数	声明	数据类型	存储器区	描述
OB_NR	INPUT	INT	I、Q、M、D、L、常数	将取消的 OB 的编号(OB20 到 OB23)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能处于激活状态时出错, 则 RET_VAL 的实际参数将包含错误代码。

### 错误信息

错误代码 (W#16#...)	解释
0000	未出错。
8090	不正确的参数 OB_NR
80A0	未启动延时中断。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误





# 11 用于处理同步错误的 SFC

## 11.1 屏蔽同步错误

### 引言

同步错误是编程和访问错误。使用不正确的地址区域、编号或不正确的地址进行编程将导致此类错误的发生。**屏蔽**这些同步错误意味着：

- 屏蔽掉的同步错误不会触发错误 OB 调用，且不会导致所编程的备选处理。
- CPU 将在错误寄存器中"记录"被屏蔽掉的已发生错误。

通过调用 SFC 36 "MSK\_FLT"执行屏蔽。

**取消屏蔽**错误意味着取消先前设置的屏蔽，并清除当前优先级的事件状态寄存器中的相应位。使用如下操作取消屏蔽：

- 调用 SFC 37 "DMSK\_FLT"
- 在完成当前优先级时取消屏蔽(仅限 S7-400)。

如果一个错误在取消屏蔽后又发生，则操作系统将启动相应的错误 OB。可以编程 OB121 实现对编程错误的响应处理，编程 OB122 实现对访问错误的响应处理。

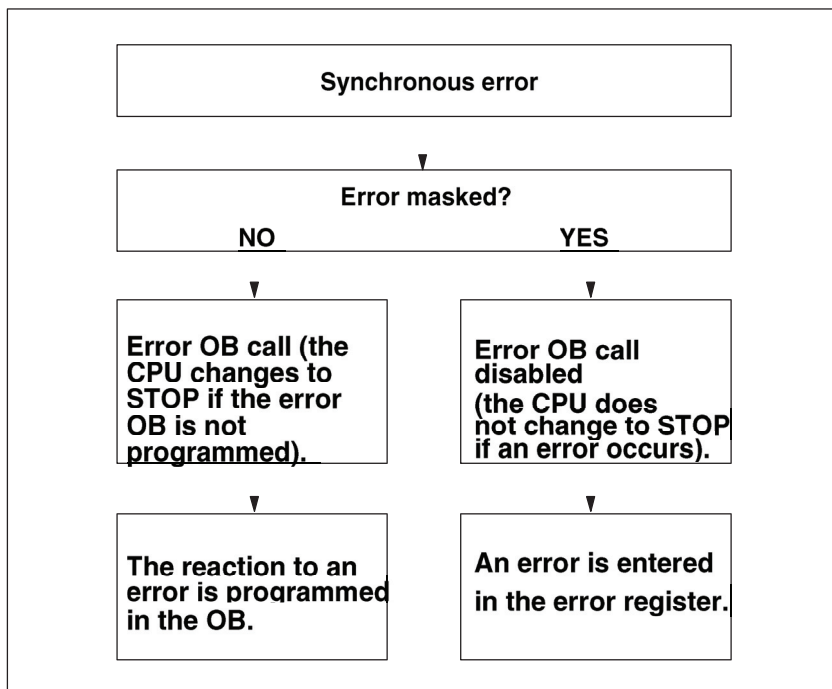
可以使用 SFC 38 "READ\_ERR"读出被屏蔽掉的已发生错误。

注意：对于 S7-300 (CPU 318 除外)，无论是否屏蔽错误，错误都将进入诊断缓冲区，CPU 的组错误 LED 都将亮起。

### 错误的常规处理

如果用户程序中出现编程和访问错误，可以不同方式做出响应：

- 可以对出现错误时操作系统调用的相应错误 OB 进行编程。
- 可以针对每个优先级单独禁用错误 OB 调用。这种情况下，当特定优先级中发生此类错误时，CPU 将不切换到 STOP 模式。CPU 将错误输入到错误寄存器中。但是，从此条目中无法识别错误的发生时间或频率。



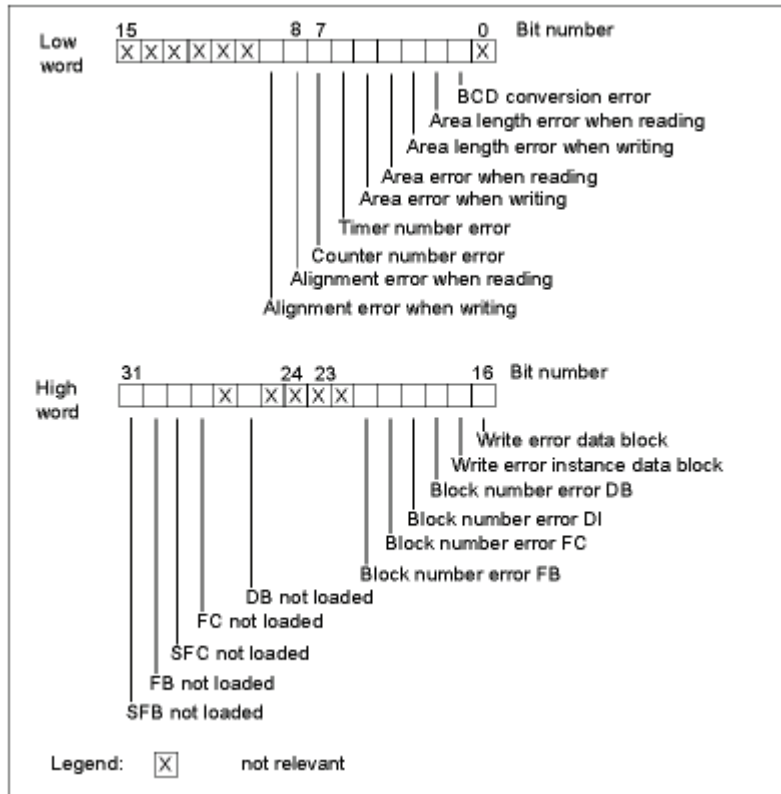
### 过滤器

同步错误被分配给特殊的位模式(被称为**错误过滤器(屏蔽)**)。此错误过滤器也在 SFC 36、37 和 38 的输入和输出参数中。

同步错误分为可以使用两种错误过滤器屏蔽的编程错误和访问错误。这些错误过滤器的图解如下。

## 编程错误过滤器

下图展示了编程错误的错误过滤器的位模式。编程错误的错误过滤器位于参数 PRGFLT\_... 中。  
请参考"编程错误的可能错误原因，低位字"或"编程错误的可能错误原因，高位字"



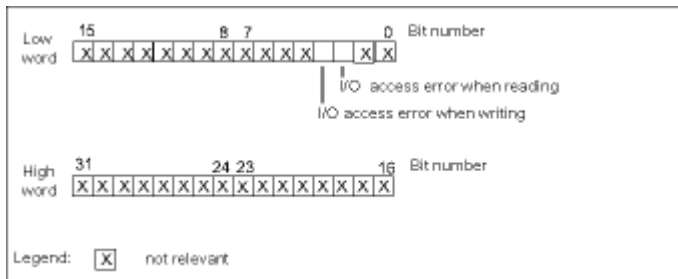
## 不相关位

在上图中，x 表示...

• ... 输入参数	用于 SFC 36、37、38	= "0"
• ... 输出参数	用于 SFC 36、37	= "1"，用于 S7-300 = "0"，用于 S7-400
	用于 SFC 38	= "0"

### 所有 CPU 的访问错误过滤器

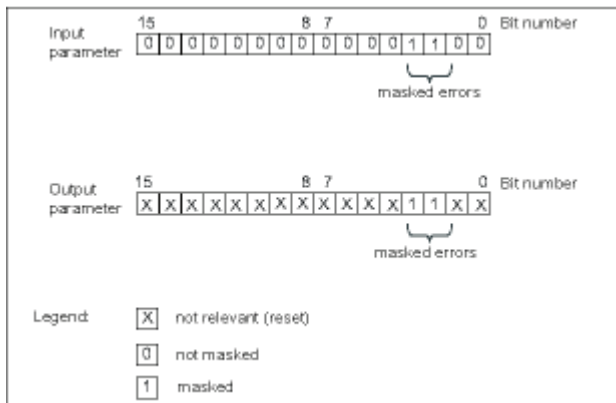
下图显示了除 CPU 417 和 CPU 417H 外所有 CPU 的访问错误的错误过滤器的位模式。访问错误的错误过滤器在参数 ACCFLT\_... 中。有关访问错误的解释，请参考表格“除 CPU 318 外所有 CPU 31x 的可能错误原因”或“所有 CPU 41x 和 CPU 318 的可能错误原因”。



### 实例

下图显示了用于访问错误的错误过滤器的低位字，带有所有 CPU 的全部被屏蔽错误。

- 作为 SFC 36 的输入参数
- 作为 SFC 36 的输出参数



## 编程错误低位字

下表列出了分配给用于编程错误的错误过滤器的低位字的错误。此表还列出了错误的可能原因。

编程错误的可能原因，低位字

错误	事件标识符 (W#16#...)	发生错误...
BCD 码转换出错	2521	... 当将转换的值不是 BCD 数字时(例如, 5E8)
读取数据时发生的区域长度错误	2522	... 当使用的地址不全在可能的地址区域内时。 实例：虽然存储器区域仅有 256 个字节长, 但必须读取 MW 320。
写入数据时发生的区域长度错误	2523	... 当使用的地址不全在可能的地址区域内时。 实例：虽然存储器区域仅有 256 个字节长, 但必须将值写入 MW 320 中。
读取时区域出错	2524	... 当使用间接跨区域寻址时, 为地址指定了不正确的区域标识符。 实例：        正确：        LAR1 P#E 12.0 L W[AR1, P#0.0] 不正确：   LAR1 P#12.0 L W[AR1, P#0.0] 对于此操作, 将发送区域长度错误信号。
写入时区域出错	2525	... 当使用间接跨区域寻址时, 为地址指定了不正确的区域标识符。 实例：        正确：        LAR1 P#E 12.0 T W[AR1, P#0.0] 不正确：   LAR1 P#12.0 T W[AR1, P#0.0] 对于此操作, 将发送区域长度错误信号。
定时器编号错误	2526	... 当访问不存在的定时器时。 实例：SP T [MW 0], 其中 MW 0 = 129; 虽然只有 128 个定时器可用, 但必须启动定时器 129。
计数器编号错误	2527	... 当访问不存在的计数器时。 实例：CU C [MW 0], 其中 MW 0 = 600; 虽然只有 512 个计数器可用, 但必须启动计数器 600(CPU 416-D)。
读取时对齐出错	2528	... 当寻址字节、字或双字地址, 而位地址 $\neq$ 0 时。 实例：        正确：        LAR1 P#M12.0 L B[AR1, P#0.0] 不正确：   LAR1 P#M12.4 L B[AR1, P#0.0]
写入时对齐出错	2529	... 当寻址字节、字或双字地址, 而位地址 $\neq$ 0 时。 实例：        正确：        LAR1 P#M12.0 T B[AR1, P#0.0] 不正确：   LAR1 P#M12.4 T B[AR1, P#0.0]

### 编程错误高位字

下表列出了分配给用于编程错误的错误过滤器的高位字的错误。同时列出了可能的错误原因。  
编程错误的可能原因，高位字

错误	事件标识符 (W#16#...)	发生错误...
写入错误数据块	2530	... 要写入的数据块为只读。
写入错误实例数据块	2531	... 要写入的背景数据块为只读。
块编号错误 DB	2532	... 必须打开的数据块的编号高于允许的最大编号。
块编号错误 DI	2533	... 必须打开的背景数据块的编号高于允许的最大编号。
块编号错误 FC	2534	... 调用的功能的编号高于允许的最大编号。
块编号错误 FB	2535	... 调用的功能块的编号高于允许的最大编号。
未装载 DB	253A	... 要打开的数据块未加载。
未装载 FC	253C	... 调用的功能未加载。
SFC 不存在	253D	... 调用的系统功能不存在。
未装载 FB	253E	... 要调用的功能块未加载。
SFB 不存在	253F	... 调用的系统/标准功能块不存在。

### 访问错误

下表列出了分配给所有 CPU 的访问错误的错误过滤器的错误。同时列出了可能的错误原因。

错误	事件标识符(W#16#...)	发生错误...
读取时的 I/O 访问错误	2942	... 未将任何信号模块分配给 I/O 区域中的地址。 或 ... 未在所选模块看门狗时间内确认对此 I/O 区域的访问(超时)。
写入时的 I/O 访问错误	2943	... 未将任何信号模块分配给 I/O 区域中的地址。 或 ... 未在所选模块看门狗时间内确认对此 I/O 区域的访问(超时)。

## 11.2 使用 SFC 36 "MSK\_FLT"屏蔽同步错误

### 描述

通过 SFC 36 "MSK\_FLT" (屏蔽同步错误), 可以控制 CPU 对同步错误的响应处理。通过此 SFC, 可以使用错误过滤器屏蔽同步错误(参见屏蔽同步错误)。当调用 SFC 36 时, 必须在当前优先级中屏蔽同步错误。

如果在输入参数中将同步错误过滤器的单个位设置为"1", 则先前设置的其它位将保留其值"1"。这样, 就可以获得可以使用输出参数读取的新错误过滤器。已屏蔽的同步错误将不调用 OB, 而只是输入到错误寄存器中。可以通过 SFC 38 "READ\_ERR"读取错误寄存器。

参数	声明	数据类型	存储器区	描述
PRGFLT_SET_MASK	INPUT	DWORD	I、Q、M、D、L、 常数	将屏蔽编程错误
ACCFLT_SET_MASK	INPUT	DWORD	I、Q、M、D、L、 常数	将屏蔽访问错误
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
PRGFLT_MASKED	OUTPUT	DWORD	I、Q、M、D、L	被屏蔽编程错误
ACCFLT_MASKED	OUTPUT	DWORD	I、Q、M、D、L	被屏蔽访问错误

### 错误信息

错误代码 (W#16#...)	解释
0000	尚未屏蔽任何错误。
0001	至少已屏蔽了一项错误。 不过, 其它错误也将屏蔽。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误

## 11.3 使用 SFC 37 "DMSK\_FLT"取消屏蔽同步错误

### 描述

通过 SFC 37 "DMSK\_FLT" (取消屏蔽同步错误), 可以取消屏蔽通过 SFC 36 "MSK\_FLT"屏蔽的错误。为此, 必须在输入参数中将错误过滤器的相应位设置为"1"。通过 SFC 37 调用, 可以取消屏蔽当前优先级的相应同步错误。同时, 在错误寄存器中清除这些条目。使用输出参数可以读取新的错误过滤器。

参数	声明	数据类型	存储器区	描述
PRGFLT_RESET_MASK	INPUT	DWORD	I、Q、M、D、L、 常数	将被取消屏蔽编程错误
ACCFLT_RESET_MASK	INPUT	DWORD	I、Q、M、D、L、 常数	将被取消屏蔽访问错误
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
PRGFLT_MASKED	OUTPUT	DWORD	I、Q、M、D、L	仍被屏蔽的编程错误
ACCFLT_MASKED	OUTPUT	DWORD	I、Q、M、D、L	仍被屏蔽的编程错误

### 错误信息

错误代码 (W#16#...)	解释
0000	已取消屏蔽所有指定的错误。
0001	至少有一项错误未屏蔽。 不过, 其它错误将被取消屏蔽。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误



## 11.4 使用 SFC 38 "READ\_ERR"读取错误寄存器

### 描述

使用 SFC 38 "READ\_ERR" (读取错误寄存器), 可以读取错误寄存器。错误寄存器的结构对应于编程和访问错误过滤器的结构, 这些结构可以通过 SFC 36 和 SFC 37 作为输入参数编程。

在输入参数中, 输入要从错误寄存器读取的同步错误。调用 SFC 38 时, 将从错误寄存器读取所需的条目, 同时将这些条目清除。

错误寄存器包含的信息可以让您了解, 在当前优先级中哪些被屏蔽的同步错误至少发生了一次。如果相应位被置位, 则意味着相应的屏蔽同步错误至少发生了一次。

参数	声明	数据类型	存储区	描述
PRGFLT_QUERY	INPUT	DWORD	I、Q、M、D、L、 常数	查询编程错误
ACCFLT_QUERY	INPUT	DWORD	I、Q、M、D、L、 常数	查询访问错误
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
PRGFLT_CLR	OUTPUT	DWORD	I、Q、M、D、L	发生的编程错误
ACCFLT_CLR	OUTPUT	DWORD	I、Q、M、D、L	发生的访问错误

### 错误信息

错误代码 (W#16#...)	解释
0000	已屏蔽所有被查询的错误。
0001	至少有一项被查询的错误未屏蔽。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误



## 12 用于处理中断和异步错误的 SFC

### 12.1 延迟和禁用中断和异步错误

#### SFC 39 到 SFC 42 的用途

通过这些 SFC，可以达到下列目的：

- 通过 SFC 39 "DIS\_IRT"，对所有后续 CPU 周期禁用中断和异步错误。
- 通过 SFC 41 "DIS\_AIRT"，延迟更高的优先级直到 OB 结束。
- 通过 SFC 40 "EN\_IRT"或 SFC 42 "EN\_AIRT"，启用中断和异步错误。

您在用户程序中编程中断和异步错误的处理。同时还必须编程相应的 OB。

#### SFC 41 和 SFC 42 的优点

通过 SFC 41 "DIS\_AIRT"禁用更高优先级的中断和异步错误，然后再通过 SFC 42 "EN\_AIRT"重新启用它们，从而延迟更高优先级的中断和异步错误，这样做具有以下优点：

被延迟的中断数由 CPU 进行计数。如果已延迟中断和异步错误，而且也在标准 FC 自身中禁用并重新启用了这些中断和异步错误，则不能通过标准 FC 取消延迟。

## 中断等级

中断分为多个等级。下表列出了所有中断等级和相应的 OB。

中断等级	OB
时间中断	OB10 到 OB17
延时中断	OB20 到 OB23
循环中断	OB30 到 OB38
硬件中断	OB40 到 OB47
DPV1 中断	OB55 到 OB57
多值计算中断	OB60
冗余错误中断	OB70、OB72
异步错误中断	OB80 到 OB87 (参见下文)
同步错误中断	OB121、OB122 (可以通过 SFC 36 到 SFC 38 屏蔽或取消屏蔽同步错误中断的处理)

## 异步错误

下表列出了可以在用户程序中通过 OB 做出响应的所有异步错误。

异步错误	OB
时间错误(例如, 超出周期时间)	OB80
电源错误(例如, 电池故障)	OB81
诊断中断(例如, 信号模块上有故障的保险丝)	OB82
删除/插入模块中断	OB83
CPU 硬件故障(例如, 存储卡已拆除)	OB84
程序顺序错误	OB85
机架故障	OB86
通讯错误	OB87

## 12.2 使用 SFC 39 "DIS\_IRT"禁用新中断和异步错误的处理

### 描述

通过 SFC 39 "DIS\_IRT" (禁用中断), 可以禁用新中断和异步错误的处理。也就是说, 如果发生中断, 则 CPU 操作系统的处理如下:

- 既不调用中断 OB 或异步错误 OB,
- 也不触发常规响应处理(如果未编程中断 OB 或异步错误 OB 的话)。

如果禁用中断和异步错误, 则这对于所有优先级都有效。只能通过调用 SFC 40 "EN\_IRT"或者通过暖启动或冷启动重新取消"DIS\_IRT"的影响。

发生中断和异步错误时, 操作系统是否将其写入诊断缓冲区取决于为 MODE 选择的输入参数设置。

### 注意

请记住, 如果在程序中使用了 SFC 39 "DIS\_IRT", 则将丢失发生的所有中断!

参数	声明	数据类型	存储器区	描述
MODE	INPUT	BYTE	I、Q、M、D、L、 常数	指定禁用哪些中断和异步错误。
OB_NR	INPUT	INT	I、Q、M、D、L、 常数	OB 编号
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错, 则返回值包含故障代码。

## MODE

MODE (B#16#...)	含义
00	将禁用所有新发生的中断和异步错误。 (未禁用同步错误。) 将值 0 赋值给 OB_NR 参数。继续在诊断缓冲区中输入条目。
01	禁用所有新发生的、属于指定中断等级的事件。通过如下指定来标识中断等级： <ul style="list-style-type: none"> <li>• 时间中断：10</li> <li>• 延时中断：20</li> <li>• 周期性中断：30</li> <li>• 硬件中断：40</li> <li>• 用于 DPV1 的中断：50</li> <li>• 多值计算中断：60</li> <li>• 冗余错误中断：70</li> <li>• 异步错误中断：80</li> </ul> 继续在诊断缓冲区中输入条目。
02	禁用所有新发生的指定中断。 使用 OB 编号指定中断。继续在诊断缓冲区中输入条目。
80	禁用所有新发生的中断和异步错误，不再将它们输入到诊断缓冲区中。将数值 0 分配给 OB_NR 参数。操作系统在诊断缓冲区中输入事件 W#16#5380。
81	禁用所有新发生的、属于指定中断等级的事件，且不再将它们输入到诊断缓冲区中。操作系统在诊断缓冲区中输入事件 W#16#5380。
82	禁用所有新发生的、属于指定中断的事件，且不再将它们输入到诊断缓冲区中。操作系统在诊断缓冲区中输入事件 W#16#5380。

## 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	输入参数 OB_NR 包含非法值。
8091	输入参数 MODE 包含非法值。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 12.3 使用 SFC 40 "EN\_IRT" 启用新中断和异步错误的处理

### 描述

使用 SFC 40 "EN\_IRT" (启用中断), 可以启用先前通过 SFC 39 "DIS\_IRT" 禁用的新中断和异步错误的处理。也就是说, 如果发生中断事件, 则 CPU 操作系统将以下列方式中的一种进行处理:

- 调用中断 OB 或异步错误 OB。
- 如果未编程中断 OB 或异步错误 OB, 则触发标准响应处理。

参数	声明	数据类型	存储器区	描述
MODE	INPUT	BYTE	I、Q、M、D、L、常数	指定将启用哪些中断和异步错误。
OB_NR	INPUT	INT	I、Q、M、D、L、常数	OB 编号
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能处于激活状态时出错, 则返回值将包含错误代码。

### MODE

MODE (B#16#...)	含义
0	启用所有新发生的中断和异步错误。
1	启用所有新发生的、属于指定中断等级的事件。通过如下指定来标识中断等级: <ul style="list-style-type: none"> <li>• 时间中断: 10</li> <li>• 延时中断: 20</li> <li>• 周期性中断: 30</li> <li>• 硬件中断: 40</li> <li>• 用于 DPV1 的中断: 50</li> <li>• 多值计算中断: 60</li> <li>• 冗余错误中断: 70</li> <li>• 异步错误中断: 80</li> </ul>
2	启用所指定中断的所有新发生的事件。使用 OB 编号指定中断。

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	输入参数 OB_NR 包含非法值。
8091	输入参数 MODE 包含非法值。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误

## 12.4 使用 SFC 41 "DIS\_AIRT"延迟更高优先级中断和异步错误的处理

### 描述

使用 SFC 41 "DIS\_AIRT" (禁用报警中断), 可以延迟优先级高于当前 OB 的中断 OB 和异步错误 OB 的处理。可以在一个 OB 中多次调用 SFC 41。SFC 41 调用将由操作系统进行计数。这些调用中的每一个都将保持有效, 直至明确地通过 SFC 42 "EN\_AIRT"调用再次取消或直至完全处理完当前 OB 为止。

一旦重新启用后, 只要通过 SFC 42 "EN\_AIRT"重新启用或当前 OB 已经执行完毕, 则 SFC 41 有效期间发生的中断和异步错误就将被处理。

参数	声明	数据类型	存储器区	描述
RET_VAL	OUTPUT	INT	I、Q、M、D、L	延迟数(= SFC 41 调用数)

### 返回值

下表列出通过 RET\_VAL 参数输出的 SFC 41 返回值。

返回值	描述
n	"n"表示处理被禁用的次数, 即 SFC 41 调用的次数(只有 n = 0 时, 才会重新启用中断处理; 请参见使用 SFC 42 "EN_AIRT"启用更高优先级中断和异步错误的处理)。



## 12.5 使用 SFC 42 "EN\_AIRT"启用更高优先级中断和异步错误的处理

### 描述

通过 SFC 42 "EN\_AIRT" (启用报警中断), 可以启用先前通过 SFC 41 "DIS\_AIRT"禁用的更高优先级中断和异步错误的处理。必须通过 SFC 42 调用来取消每个 SFC 41 调用。

### 实例

例如, 如果通过五次 SFC 41 调用禁用了五次中断, 则必须通过五次 SFC 42 调用来取消这些调用。

参数	声明	数据类型	存储器区	描述
RET_VAL	OUTPUT	INT	I、Q、M、D、L	完成 SFC 42 或发生错误消息时仍编程的延迟数。

### 返回值和错误信息

请参考使用输出参数 RET\_VAL 评估错误

"使用输出参数 RET\_VAL 评估错误"一章介绍了如何评估 RET\_VAL 参数的错误信息。此章还包含了 SFC 的常规错误信息。下表包含了可以通过 RET\_VAL 参数输出的、SFC 42 特定的所有错误信息。

返回值和错误信息	描述
N	"n"表示尚未通过 SFC 42 调用取消的 SFC 41 调用次数(只有当"n"= 0 时, 才会重新启用中断处理)。
W#16#8080	虽然已启用中断处理, 但已再次调用了此功能。



## 13 用于诊断的 SFC

### 13.1 系统诊断

CPU 将保留有关可编程逻辑控制器状态的内部数据。通过系统诊断功能，可以读取最重要的数据。可以使用 STEP 7 在编程设备上显示某些数据。

还可以使用 SFC "RD\_SINFO"和"RDSYSST"访问系统诊断所需的数据。

### 13.2 使用 SFC 6 "RD\_SINFO"读取 OB 启动信息

#### 描述

通过 SFC 6 "RD\_SINFO" (读取启动信息)，可以读取有关下列各项的启动信息：

- 尚未完全执行的、被调用的最近一个 OB，  
和
- 将要启动的最近一个启动 OB。

任一情况下均没有时间标志。如果在 OB100 或 OB101 或 OB102 中调用，将返回两条相同的启动信息消息。

参数	声明	数据类型	存储器区	描述
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
TOP_SI	OUTPUT	STRUCT	D、L	当前 OB 的启动信息
START_UP_SI	OUTPUT	STRUCT	D、L	上次启动的启动 OB 的启动信息

**TOP\_SI 和 START\_UP\_SI**

输出参数 TOP\_SI 和 START\_UP\_SI 是有着相同单元的两个结构(参见下表)。

结构单元	数据类型	描述
EV_CLASS	BYTE	<ul style="list-style-type: none"> <li>0 到 3 位：事件 ID</li> <li>4 到 7 位：事件等级</li> </ul>
EV_NUM	BYTE	事件编号
PRIORITY	BYTE	优先级编号(B#16#FE 的含义：OB 不可用或已锁定，或者不能在当前工作模式下启动)
NUM	BYTE	OB 编号
TYP2_3	BYTE	数据标识符 2_3：标识在 ZI2_3 中输入的信息
TYP1	BYTE	数据标识符 1：标识在 ZI1 中输入的信息
ZI1	WORD	附加信息 1
ZI2_3	DWORD	附加信息 2_3

**注意**

表格中列出的结构单元和 OB 的临时变量有相同内容。

但请注意，各个 OB 的临时变量可以有不同的名称和不同的数据类型。同时，请注意每个 OB 的调用接口包括 OB 请求的日期和时间等附加信息。

EV\_CLASS 结构单元的 4 到 7 位包含事件等级。此处可以为下列值：

- 1: 来自标准 OB 的启动事件
- 2: 来自同步错误 OB 的启动事件
- 3: 来自异步错误 OB 的启动事件

PRIORITY 结构单元提供属于当前 OB 的优先级。

除这两个单元外，NUM 也是相关的。NUM 包含当前 OB 或上次启动的启动 OB 的编号。

## 实例

上次启用且未完全处理的 OB 用作 OB80。上次启动的启动 OB 用作 OB100。

下表显示了 SFC 6 "RD\_SINFO"的参数 TOP\_SI 的结构单元及 OB80 的各局部变量的分配。

TOP_SI		OB80	
结构单元	数据类型	局部变量	数据类型
EV_CLASS	BYTE	OB80_EV_CLASS	BYTE
EV_NUM	BYTE	OB80_FLT_ID	BYTE
PRIORITY	BYTE	OB80_PRIORITY	BYTE
NUM	BYTE	OB80_OB_NUMBR	BYTE
TYP2_3	BYTE	OB80_RESERVED_1	BYTE
TYP1	BYTE	OB80_RESERVED_2	BYTE
ZI1	WORD	OB80_ERROR_INFO	WORD
ZI2_3	DWORD	OB80_ERR_EV_CLASS	BYTE
		OB80_ERR_EV_NUM	BYTE
		OB80_OB_PRIORITY	BYTE
		OB80_OB_NUM	BYTE

下表显示了 SFC 6 "RD\_SINFO"的参数 START\_UP\_SI 的结构单元及 OB100 的各局部变量的分配。

START_UP_SI		OB100	
结构单元	数据类型	局部变量	数据类型
EV_CLASS	BYTE	OB100_EV_CLASS	BYTE
EV_NUM	BYTE	OB100_STRTUP	BYTE
PRIORITY	BYTE	OB100_PRIORITY	BYTE
NUM	BYTE	OB100_OB_NUMBR	BYTE
TYP2_3	BYTE	OB100_RESERVED_1	BYTE
TYP1	BYTE	OB100_RESERVED_2	BYTE
ZI1	WORD	OB100_STOP	WORD
ZI2_3	DWORD	OB100_STRT_INFO	DWORD

## 错误信息

SFC 6 "RD\_SINFO"不提供任何特定错误信息，而只提供常规错误信息。在标题为"SFC 的常规参数"部分的使用输出参数 RET\_VAL 评估错误中详细介绍了常见错误代码及如何评估这些错误。

## 13.3 使用 SFC 51 "RDSYSST"读取系统状态列表或部分列表

### 描述

通过系统功能 SFC 51 "RDSYSST" (读取系统状态), 可以读取系统状态列表或部分系统状态列表。

调用 SFC 51 时, 通过将值"1"赋给输入参数 REQ 来启动读取。如果可以立即读取系统状态, 则 SFC 将在 BUSY 输出参数中返回值 0。如果 BUSY 包含值 1, 则尚未完成读取功能。

### 注意

如果使用 SSL-ID W#16#00B1 或 W#16#00B2 或 W#16#00B3, 在诊断中断 OB 中调用 SFC 51 "RDSYSST", 并访问启动该诊断中断的模块, 则将会立即读取系统状态。

通过 SFC51 "RDSYSST", 将仅传送完整的数据记录。

### 系统资源

如果以较短间隔一个接一个地启动几个异步读取功能(使用 SSL\_ID W#16#00B4、W#16#4C91、W#16#4092、W#16#4292 和 W#16#4692 及可能的 W#16#00B1 和 W#16#00B3 启动的作业), 则操作系统将确保执行所有读取作业且彼此不互相干扰。如果达到系统资源的极限, 则将在 RET\_VAL 中加以指示。可以通过重复执行作业来纠正这种临时错误状况。

"同时"激活的 SFC 51 作业的最大数目取决于 CPU。您将在/70/和/101/中找到此信息。

### 注意

对于 S7-400 CPU, 通过 SFC 51 最多可以将 432 个字节传输到目标区域

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	REQ=1: 启动处理
SSL_ID	INPUT	WORD	I、Q、M、D、L、常数	将要读取的系统状态列表或部分列表的 SSL-ID。
INDEX	INPUT	WORD	I、Q、M、D、L、常数	部分列表中对象的类型或编号。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行 SFC 时出错, 则 RET_VAL 参数将包含错误代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	TRUE: 尚未完成读取。
SSL_HEADER	OUTPUT	STRUCT	D、L	参见下文。

参数	声明	数据类型	存储器区	描述
DR	OUTPUT	ANY	I、Q、M、L、D	SSL 列表读取或 SSL 部分列表读取的目标区域： <ul style="list-style-type: none"> <li>如果仅读取了 SSL 列表的标题信息，则不能评估 DR 的值，而只能评估 SSL_HEADER 的值。</li> <li>否则，LENTHDR 和 N_DR 的乘积将指示已在 DR 中输入了多少字节。</li> </ul>

## SSL\_HEADER

SSL\_HEADER 参数是一个如下定义的结构：

```
SSL_HEADER:   STRUCT
  LENTHDR:    WORD
  N_DR:       WORD
END_STRUCT
```

LENTHDR 是 SSL 列表或 SSL 部分列表的数据记录的长度。

- 如果仅读取了 SSL 列表的标题信息，则 N\_DR 包含属于它的数据记录数。
- 否则，N\_DR 包含传送到目标区域的数据记录数。

## 错误信息

错误代码 (W#16#...)	描述
0000	无错。
0081	结果域过短。(但是，仍然将尽可能多地提供数据记录。SSL 标题指示此数值。)
7000	首次调用 REQ=0：没有数据传输；BUSY 的值为 0。
7001	首次调用 REQ=1：已启动数据传送；BUSY 的值为 1。
7002	中间调用(REQ 无关联)：数据传送已经激活；BUSY 的值为 1。
8081	结果域过短(没有可容纳一个数据记录的足够空间)。
8082	SSL_ID 错误，或在 CPU 或 SFC 中未知。
8083	INDEX 错误或不允许。
8085	因为系统中存在问题，信息目前不可用(例如，因为资源不足)。
8086	因为出现系统错误，无法读取数据记录(总线、模块、操作系统)。
8087	因为模块不存在或不能进行应答，无法读取数据记录。
8088	因为实际的模块标识符与预期的模块标识符不同，无法读取数据记录。
8089	因为模块无法进行诊断或不支持该数据记录，无法读取数据记录。
80A2	DP 协议错误(第 2 层错误)(临时错误)
80A3	关于用户接口/用户的 DP 协议错误(临时错误)
80A4	与通讯总线有关的通讯问题(CPU 和外部 DP 接口模块间出错)
80C5	分布式 I/O 不可用(临时错误)。
80C6	因为优先级中止(重新启动或后台操作)，数据记录传送已停止
80D2	因为模块不能进行诊断，无法读取数据记录。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

SSL\_ID

**注意**

有关可以通过 SFC 51 "RDSYSST"读取的部分列表的信息，参见

- /72/ (对于 S7-300)
- 下表(对于 S7-400)。

SSL_ID (W#16#...)	部分列表	INDEX (W#16#...)
	<b>模块标识符</b>	
0111	一个标识数据记录	
	模块的标识	0001
	系统扩展卡的标识	0004
	基本硬件的标识	0006
	基本固件的标识	0007
	<b>CPU 特征</b>	
0012	所有特征	无关
0112	一个组的特征	
	MC7 处理单元	0000
	时间系统	0100
	系统特性	0200
	MC7 语言描述	0300
	SFC 87 和 SFC 88 的可用性	0400
0F12	仅限 SSL 部分列表报头信息	无关
	<b>用户存储区</b>	
0113	指定的存储器区域的一个数据记录	
	工作存储器	0001
	<b>系统区域</b>	
0014	所有系统区域的数据记录	无关
0F14	仅限 SSL 部分列表报头信息	无关
	<b>块类型</b>	
0015	所有模块类型的数据记录	无关
	<b>一个组件的标识</b>	
001C	所有组件的标识	无关
011C	一个组件的标识	
	自动化系统的名称	0001
	模块名称	0002
	模块的系统标识符	0003
	版权条目	0004
	模块的序列号	0005
	模块类型名称	0007



SSL_ID (W#16#...)	部分列表	INDEX (W#16#...)
	存储卡的序列号	0008
	CPU 模块的制造商和配置文件	0009
	模块的位置指定	000B
	同步模块 1 的序列号	000C
	同步模块 2 的序列号	000D
	系统扩展卡的序列号	000E
021C	H 系统的一个 CPU 中的所有组件的标识	机架号
031C	H 系统中所有冗余 CPU 中的一个组件的标识	索引
0F1C	仅限 SSL 部分列表报头信息	无关
	<b>中断状态</b>	
0222	所指示的中断的数据记录	OB 编号
	<b>过程映像分区和 CPU 的分配</b>	
0025	将所有过程映像分区分配到 OB	无关
0125	过程映像分区到相应 OB 的分配	过程映像分区编号。
0225	OB 到相应过程映像分区的分配	OB 编号。
0F25	仅 SSL 部分列表标题的信息	无关
	<b>通讯状态数据</b>	
0132	一个通讯单元的状态数据	
	诊断	0005
	时间系统	0008
0232	一个通讯单元的状态数据	
	CPU 保护级别和操作员控制设置	0004
	<b>H CPU 组信息</b>	
0071	有关 H 系统当前状态的信息	无关
0F71	仅限 SSL 部分列表报头信息	无关
	<b>模块 LED 的状态(不能从所有 CPU 中读取, 参见/102/)。</b>	
0174	LED 的状态	LED 标识符
	<b>在 H 系统中切换的 DP 从站</b>	
0C75	H 系统与切换的 DP 从站间的通讯状态	DP 从站接口的诊断地址
	<b>DP 主站系统信息</b>	
0090	CPU 已知的 DP 主站系统信息	0000
0190	有关 DP 主站系统的信息	DP 主站系统 ID
0F90	仅限 SSL 部分列表报头信息	0000
	<b>模块状态信息(最多提供 27 个数据记录)</b>	
0091	所有插入的模块/子模块的状态信息	无关
0191	所有带不正确类型标识符的、未取消激活的模块/机架的模块状态信息	无关
0291	所有存在故障且未取消激活的模块的模块状态信息	无关
0391	所有无法获得的模块的模块状态信息	无关
0591	主机模块的所有子模块的模块状态信息	无关
0991	DP 主站系统的模块状态信息	DP 主站系统 ID

SSL_ID (W#16#...)	部分列表	INDEX (W#16#...)
0C91	在中央组态中的或与集成 DP 通信处理器连接的或在 PROFINET 接口模块(集成的或外部的)上的模块的模块状态信息	逻辑基址
4C91	连接到外部 DP 通讯处理器的模块的模块状态信息	逻辑基址
0D91	指定机架/DP 站(DP 或 PROFINET)中的所有模块的模块状态信息	机架或 DP 主站系统 ID 和站号或站号和 PNIO 子系统 ID 的最后两个位置
0E91	所有已分配模块的模块状态信息	无关
0F91	仅限 SSL 部分列表报头信息	不相关
	<b>机架/站的状态信息</b>	
0092	中央组态的机架/DP 主站系统的站的预期状态	0 / DP 主站系统 ID
4092	连接到外部 DP 接口的 DP 主站系统的站的预期状态	DP 主站系统 ID
0192	通过集成 DP 接口连接的 DP 主站系统中的站的激活状态	DP 主站系统 ID
0292	中央组态中的机架的当前状态/DP 主站系统的站的当前状态	0 / DP 主站系统 ID
4292	通过外部 DP 接口模块连接的 DP 主站系统中的站的当前状态。	DP 主站系统 ID
0392	至少一块电池出现故障后，CPU 的机架/模块机架中备用电池的状态	0
0492	CPU 的所有机架/模块机架的总电池备用状态	0
0592	CPU 的所有机架/模块机架的 24V 电源的状态	0
0692	中央组态中扩展机架的诊断状态/通过集成 DP 接口模块连接的 DP 主站系统中的站的诊断状态	0 / DP 主站系统 ID
4692	通过外部 DP 接口模块连接的 DP 主站系统中的站的诊断状态	DP 主站系统 ID
	<b>机架/站的状态信息</b>	
0094	IO 控制器系统的中央机架/站中的机架的期望状态	0 / PNIO 子系统 ID
0194	已组态和取消激活的 IO 控制器系统的激活状态	PNIO 子系统 ID
0294	IO 控制器系统的中央机架/站中的机架的实际状态	0 / PNIO 子系统 ID
0694	IO 控制器系统的中央机架/站中的扩展单元的诊断状态	0 / PNIO 子系统 ID
0794	IO 控制系统的中央机架/站中的机架的维护状态	0 / PNIO 子系统 ID
0F94	仅限报头信息	-
	<b>DP 主站系统/PROFINET IO 系统的扩展信息</b>	
0195	有关 DP 主站系统的扩充信息	DP 主站系统 ID
0F95	仅限 SSL 部分列表报头信息	0000
	<b>模块状态信息、PROFINET IO 和 PROFIBUS DP</b>	
0696	指定模块的所有子模块的模块状态信息(仅对集成接口模块上的 PROFINET IO)	具有 I/O ID 的地址
0C96	在中央组态中的或在集成 PROFIBUS DP 接口模块上的或 PROFINET 接口模块(集成或外部)上的模块/子模块的模块状态信息	具有 I/O ID 的起始地址

SSL_ID (W#16#...)	部分列表	INDEX (W#16#...)
<b>工具变换装置信息(PROFINET IO)</b>		
009C	关于 PROFINET IO 系统中所有工具变换装置及其工具的信息	PROFINET IO 系统 ID
019C	关于 PROFINET IO 系统中所有工具变换装置的信息	PROFINET IO 系统 ID
029C	关于一个工具变换装置及其工具的信息	工具变换装置的逻辑地址
039C	关于一个工具及其 IO 设备的信息	工具的一个 IO 设备的逻辑地址
0F9C	仅限 SSL 部分列表报头信息	不相关
<b>诊断缓冲区(最多提供 21 个数据记录)</b>		
00A0	当前活动工作模式中可以提供的所有条目	无关
01A0	最新条目, 编号在索引中指定	数量
0FA0	仅限 SSL 部分列表报头信息	无关
<b>关于模块的诊断数据</b>		
00B1	一个模块的前四个诊断字节(数据记录 0)	逻辑基址
00B2	一个模块的所有诊断数据(≤ 220 字节, 数据记录 1) (无 DP 模块)	机架、插槽
00B3	一个模块的所有诊断数据(≤ 220 字节, 数据记录 1)	逻辑基址
00B4	DP 从站的诊断数据	组态的诊断地址

## 13.4 使用 SFC 52 "WR\_USMSG"将自定义诊断事件写入诊断缓冲区

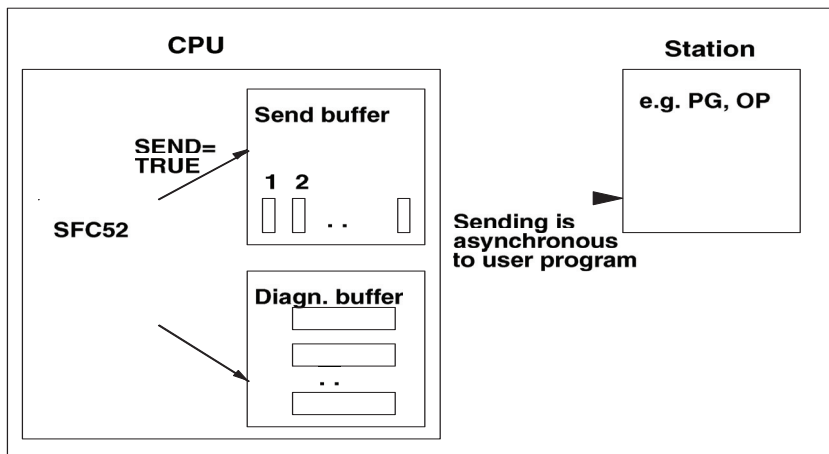
### 描述

通过 SFC 52 "WR\_USMSG" (将用户元素写入诊断缓冲区), 可以将自定义诊断事件写入诊断缓冲区。还可以将相应的诊断信息发送到为此目的而登录的所有站(通过设置输入参数 SEND = TRUE)。如果出错, 则输出参数 RET\_VAL 将提供错误信息。

### 发送自定义诊断消息

SFC 52 将自定义诊断事件写入诊断缓冲区。然后, 还可以将相应的诊断消息发送到为此目的而登录的任何站(通过设置输入参数 SEND = TRUE)。接下来, 自定义诊断消息将写入发送缓冲区并自动发送到登录的站。

可以检查当前是否能够发送自定义诊断消息。为此, 可以通过参数 SSL\_ID = W#16#0132 和 INDEX = W#16#0005 调用 SFC 51 "RDSYSST"。所获取的数据记录的第四个字指示当前能(1)否(0)发送用户元素。



### 发送缓冲区满

只有在发送缓冲区未滿时, 才能在发送缓冲区中输入诊断消息。可以在发送缓冲区中输入的条目数取决于所用 CPU 的类型。

如果发送缓冲区已滿, 则:

- 诊断事件仍输入诊断缓冲区中,
- 参数 RET\_VAL 指示发送缓冲区已滿(RET\_VAL = W#16#8092)。

### 未登录站

如果要发送自定义诊断消息(SEND = TRUE), 而未登录任何站, 则

- 自定义诊断事件将输入诊断缓冲区中,
- 参数 RET\_VAL 指示未登录任何站(RET\_VAL = W#16#8091 或 W#16#8091)。如果使用较早版本的 CPU, 将显示值 W#16#8091)。

## 常规结构

诊断缓冲区中元素的内部结构如下：

字节	目录
1 和 2	事件 ID
3	优先级等级
4	OB 编号
5 和 6	保留
7 和 8	附加信息 1
9、10、11 和 12	附加信息 2
13 - 20	时间标志

## 事件 ID

每个事件都分配有事件 ID。

## 附加信息

这是有关事件的附加信息。每个事件的附加信息可以不同。创建诊断事件时，可以自己决定这些条目的内容。

发送自定义诊断消息时，可在(事件标识符指定的)消息文本中将附加信息合并为关联值。

## 时间标志

时间戳的类型为 Date\_and\_Time。

参数	声明	数据类型	存储器区	描述
SEND	INPUT	BOOL	I、Q、M、D、L、常数	启用发送自定义诊断消息到所有已登录的站
EVENTN	INPUT	WORD	I、Q、M、D、L、常数	事件标识符 - 由您分配事件标识符。而不是由消息服务器进行分配。
INFO1	INPUT	ANY	I、Q、M、D、L	1 个字长的附加信息
INFO2	INPUT	ANY	I、Q、M、D、L	2 个字长的附加信息
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息

## SEND

如果 SEND = TRUE，则自定义诊断消息将发送到所有已登录的站。只有当站已经登录且发送缓冲区未滿时，才会发送消息。元素的发送将与用户程序异步。

## EVENTN

EVENTN 参数包含用户事件的事件标识符。可输入类型为 W#16#8xyz、W#16#9xyz、W#16#Axyz、W#16#Bxyz 的事件标识符。

W#16#8xyz 和 W#16#9xyz 格式的标识符属于预定义事件，W#16#Axyz 和 W#16#Bxyz 格式的标识符属于自由定义的事件。

进入状态的事件由 x = 1 指示，离开状态的事件由 x = 0 指示。对于 A 和 B 等级的事件，"yz"为在消息组态中分配给消息的十六进制格式的消息编号。

事件标识符的结构在第 26.1 部分介绍。

## INFO1

INFO1 参数包含一个字长的信息。INFO1 允许使用下列数据类型：

- WORD
- INT
- ARRAY [0 - 1] OF CHAR

可以在消息文本中将参数 INFO1 合并为关联值，从而向该消息添加最新信息。

## INFO2

INFO2 参数包含两个字长的信息。INFO2 允许使用下列数据类型：

- DWORD
- DINT
- REAL
- TIME
- ARRAY [0 - 3] OF CHAR

可以在消息文本中将参数 INFO2 合并为关联值，从而向该消息添加最新信息。

## 错误信息

错误代码 (W#16#...)	解释
0000	无错误
0091	未登录任何站(诊断事件输入诊断缓冲区中)
8083	INFO1 的数据类型不允许使用
8084	INFO2 的数据类型不允许使用
8085	EVENTN 不允许使用
8086	INFO1 的长度不允许 S7-300 CPU：将最大值作为附加信息 1 的长度，且诊断缓冲区条目在运行。 S7-400 CPU：没有诊断缓冲区条目。
8087	INFO2 的长度不允许 S7-300 CPU：将最大值作为附加信息 2 的长度，且诊断缓冲区条目在运行。 S7-400 CPU：没有诊断缓冲区条目。
8091	(只有当使用较早版本的 CPU 时，才会显示此错误代码。) 未登录任何站(诊断事件输入诊断缓冲区中)。
8092	目前不可以发送，发送缓冲区已满(诊断事件输入诊断缓冲区中)。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 13.5 使用 SFC 78 "OB\_RT"确定 OB 程序运行时间

### 描述

通过 SFC 78 "OB\_RT"，可以确定不同时间段各 OB 的运行时间。

### 注意

对于所需 OB，无论它当前是否加载，SFC 78 都将返回上一次为它记录的时间值。SFC 78 数据不会被删除或覆盖，而是在重新启动(暖启动)后被复位。

### 参数

参数	声明	数据类型	存储区	含义
OB_NR	INPUT	INT	I、Q、M、D、L	要查询上次计算的的时间的 OB。有效的编号为 CPU 的 OB 组态中的所有编号(OB 121 和 OB 122 除外)。同步错误处理时间包括在处理导致该错误的 OB 所需的时间中。如果声明 OB 121 和 122，或者声明未在 CPU 中组态的 OB，将导致错误消息。OB_NR=0 时，将传送已调用 SFC 的 OB 的数据。在 OB 121 或 OB 122 中调用 SFC 78 且 OB_NR=0 时，将输出触发 OB 的中断的所有时间数据，包括 OB 12x 中的时间数据。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果处理此功能时出错，则返回值将包含错误代码。否则，RET_VAL 将包含这些数据对应的 OB 的编号。
PRIO	OUTPUT	INT	I、Q、M、D、L	所查询 OB 的优先级在 PRIO 中输出



参数	声明	数据类型	存储区	含义
LAST_RT	OUTPUT	DINT	I、Q、M、D、L	<p>指定 OB 的最近一次执行的运行时间(微秒)。</p> <p>如果您请求其运行时间的 OB 当前正在处理：</p> <p>在所需 OB 的当前执行期间，第一次调用 SFC 78 后，LAST_RT 将输出该 OB 的上次总执行运行时间。</p> <p>在所需 OB 的当前执行期间，每次随后调用 SFC 78 时，Last_RT 将</p> <ul style="list-style-type: none"> <li>• 指示 DW#16#FFFF FFFF (如果所需 OB 已在 OB_NR=0 时调用 SFC 78)。</li> <li>• 指示所需 OB 的上一次执行的运行时间(如果此 OB 未在 OB_NR=0 时调用 SFC 78)。</li> </ul> <p>注意：LAST_RT 中不包括具有更高优先级的 OB 导致的中断时间。OB 相关的操作系统服务(如生成和提供 OB 启动信息、更新过程映像、更新过程映像分区等)包含在 LAST_RT 中</p>
LAST_ET	OUTPUT	<u>DINT</u>	I、Q、M、D、L	<p>指定 OB 从调用到执行结束之间的时间间隔(微秒)，即上一次完成的指定 OB 的处理所使用的时间。</p> <p>如果您请求其运行时间的 OB 当前正在处理：</p> <p>在所需 OB 处理期间，第一次调用 SFC 78 后，LAST_ET 指示从上次完成 OB 调用到指定 OB 执行结束之间的时间间隔。</p> <p>在所需 OB 的当前执行期间，每次随后调用 SFC 78 时，Last_ET 将</p> <ul style="list-style-type: none"> <li>• 指示 DW#16#FFFF FFFF (如果所需 OB 已在 OB_NR=0 时调用 SFC 78)。</li> <li>• 指示从上次完成 OB 调用到指定 OB 执行结束之间的时间间隔(如果 OB_NR=0 时此 OB 尚未调用 SFC 78 的话)。</li> </ul> <p>注意：LAST_RT 中包括具有更高优先级的 OB 导致的中断时间。</p>
CUR_T	OUTPUT	DINT	I、Q、M、D、L	<p>指定的、当前处理的 OB 的 OB 请求的时间(以微秒为单位的相对时间值)。如果指定的 OB 不是正被处理，则 CUR_T = 0。</p> <p>注意：系统时间是从 0 到 2.147.483.647 (微秒)计数的计数器。计数器溢出后将从 0 重新开始计数。</p>
CUR_RT	OUTPUT	DINT	I、Q、M、D、L	<p>指定 OB 已用的执行时间(微秒)。如果指定的 OB 不是正被处理或尚未处理，则 CUR_RT = 0。处理后，运行时间数据输入到 LAST_RT 中，CUR_RT 将设置为零。</p> <p>注意：CUR_RT 中不包括具有更高优先级的 OB 导致的中断时间。CUR_RT 中包括调用 SFC 78 之前执行的 OB 特定的操作系统服务。</p>

参数	声明	数据类型	存储区	含义
CUR_ET	OUTPUT	DINT	I、Q、M、D、L	从调用当前处理的指定 OB 起所经过的时间(以微秒为单位的运行时间)。如果指定的 OB 不是正被处理，则 CUR_ET = 0。处理后，运行时间数据将输入到 LAST_RT 中，CUR_ET 将设置为零。 注意：CUR_ET 中包括具有更高优先级的 OB 导致的中断时间。
NEXT_ET	OUTPUT	DINT	I、Q、M、D、L	如果在完成当前请求前，队列中还有更多指定 OB 的调用，则 NEXT_ET 指示实际时间与下一请求的执行时间之间的时间间隔(微秒)。如果对此 OB 来讲，除当前正排队或正处理的启动事件外，不存在其它启动事件，则 NEXT_ET = 0。 WinLC RTX 和 S7-400-CPU 中不使用此参数。它们的 NEXT_ET 值为 DW#16#FFFFFFFF。 注意：NEXT_ET 中包括具有更高优先级的 OB 导致的中断时间。

这些时间还包括任何嵌套的同步错误中断(OB 121、OB 122)处理的运行时间。

**注意**

如果在 OB\_NR 中声明在 CPU 的动态项目数据中存在的 OB 编号，而 OS 并没有调用相应的 OB，或者未将其下载到 CPU 中，则 RET\_VAL 包含指定的 OB 编号；PRIO 包含指定 OB 的组态优先级(默认值，如果需要的话)。运行系统参数(CUR\_RT、CUR\_ET、LAST\_RT、LAST\_ET、NEXT\_ET)返回初始值 DW#16#FFFF FFFF。

在下列启动类型或模式更改时，将运行系统参数设为它们的初始值：

- 标准 CPU：暖启动、冷启动、热启动
- H-CPU：暖重新启动、冷重新启动、从 RUN-Solo 更改为 RUN-Redundant、从 RUN-Redundant 更改为 RUN-Solo

**错误信息**

事件等级 错误代码	解释
1 - 102	要传送信息到其中的 OB 的编号。
W#16#8080	OB_NR 参数包含非法值。
W#16#8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 13.6 使用 SFC 87 "C\_DIAG"诊断当前连接状态

### 描述

可以使用 SFC 87 "C\_DIAG"诊断所有 S7 连接和所有容错 S7 连接(或其局部连接)的当前状态。

恰当地评估这些连接数据能够发现 S7 连接故障及当前 S7 连接故障,并在合适的时机报告给可视化操作系统。监视的连接可以是自动化系统间的连接,也可以是自动化系统与可视化操作系统间的连接。

---

### 注意

CPU 工作状态的改变 (RUN -> STOP ->RUN)不会影响所组态的连接的状态。例外: H 站从"冗余"系统状态变为"停止"系统状态时,与待机 CPU 的所有容错连接中的局部连接将断开。

另一方面,在掉电后,所有组态的连接都将重新建立,这将改变连接状态。

启动期间或启动后第一次调用 SFC 87 连接信息将有所不同,这取决于 CPU 的上一工作模式是 STOP 还是 POWER OFF。

---

### 工作模式

SFC 87 "C\_DIAG"是异步 SFC 操作,也就是说,处理可以扩展到跨多个 SFC 调用。

通过使用 REQ=1 调用 SFC 87 来启动作业。

如果可以立即执行作业,SFC 将在输出参数 BUSY 中返回值 0。如果 BUSY 的值为 1,则作业仍处于激活状态。

### 何时调用 SFC 87 ?

要识别 S7 连接的故障和当前 S7 连接,可以在已启动的周期性中断 OB 中调用 SFC87,例如,由操作系统每 10 秒调用一次。

因为连接状态通常不发生改变,所以仅当它们自上次调用(MODE=B#16#02 时的调用,参见下文)后发生改变时,才应通过这些周期性调用将连接数据复制到用户程序中。

**如何调用 SFC 87 ?**

SFC 87 "C\_DIAG"提供了四种可能的工作模式，这在下表中给予介绍。

MODE (B#16#...)	SFC 将连接数据复制到用户程序	SFC 将确认信息传送到操作系统
00	否	是
01	是	是
02	<ul style="list-style-type: none"> <li>• 是，如果连接数据已改变</li> <li>• 否，如果连接数据未改变</li> </ul>	是
03	是	否

通过将确认消息传送到操作系统来确认自上次调用 SFC 87 (通过 MODE=B#16#00、01 或 02)后连接数据的状态改变。

**注意**

如果在处于"有条件复制"模式(MODE=B#16#02)下的周期性中断OB中操作SFC 87 则必须确保CPU冷启动后，目标区域中不包含初始化值。可以通过在"带确认的无条件复制"模式(MODE=B#16#01)下调用一次 SFC 87，以便在 OB 102 中实现这一点。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数。	要激活的控制参数请求 REQ=1：初始化该作业(如果尚未启动)
MODE	INPUT	BYTE	I、Q、M、D、L、常数。	作业指定 可能的值： <ul style="list-style-type: none"> <li>• B#16#00：SFC 不复制连接数据，仅将确认信息传送到操作系统。</li> <li>• B#16#01：无论状态如何改变，SFC 均将所有连接数据复制到用户程序中，并将确认信息传送到操作系统。</li> <li>• B#16#02：如果连接数据已改变，则 SFC 会将其复制到用户程序中。如未改变，则不复制。在两种情况下，SFC 均将确认信息传送到操作系统中。</li> <li>• B#16#03：SFC 将连接数据复制到用户程序中(无论改变后的状态如何)。而不将确认消息传送到操作系统中。</li> </ul>
RET_VAL	OUTPUT	INT	I、Q、M、D、L	返回值(错误代码或作业状态)
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：作业未完成。
N_CON	OUTPUT	INT	I、Q、M、D、L	.DIS_PCON 或.DIS_CON 的值为 TRUE 的 CON_ARR 中最后一个结构的索引。因此，在用户程序中只需检查 CON_ARR 的第一个 N_CON 元素。 注意：CON_ARR 域中的第一个结构具有索引 1。
CON_ARR	OUTPUT	ANY	I、Q、M、D、L	用于接收的连接数据的目标区域。 只允许使用数据类型 BYTE。 将给每个连接分配一个结构。 选择一个目标区域大小，使之在连接了 CPU 允许的最大数目的连接时仍然可以接收所有结构。尤其是在更换 CPU 后，而新的 CPU 拥有的连接资源多于旧 CPU 时要注意这一点。

### 组织目标区域 CON\_ARR

读取目标区域是一个结构域。将给每个连接分配一个结构。

最初，该域无需填入有效条目，域的两个有效条目间可以包含无效条目。

连接不按连接引用排序。

#### 注意

如果将连接数据从操作系统复制到所选择的目标区域，则可以确保连接的日期一致性

## 结构组织

参数	数据类型	描述
CON_ID	WORD	在 NETPRO 中为此连接分配的连接引用 W#16#FFFF：无效指定，也就是说，尚未组态连接。如果设置 CON_ARR[i].DIS_PCON 或 CON_ARR[i].DIS_CON (参见下文)，则自上次调用 SFC 87 后已重新组态或删除了此连接。
STAT_CON	BYTE	S7 连接或冗余 S7 连接的当前状态 可能的值： <ul style="list-style-type: none"> <li>• B#16#00：S7 连接未建立</li> <li>• B#16#10：冗余 S7 连接未建立</li> <li>• B#16#01：当前正在建立 S7 连接</li> <li>• B#16#11：当前正在建立冗余 S7 连接</li> <li>• B#16#02：S7 连接已建立</li> <li>• B#16#12：冗余 S7 连接已建立，但不是冗余状态</li> <li>• B#16#13：冗余 S7 连接已建立</li> </ul>
PROD_CON	BYTE	运行连接的局部连接号。 可能的值：0, 1, 2, 3
STBY_CON	BYTE	备用连接的局部连接号(B#16#FF：无备用连接) 可能的值：0, 1, 2, 3 注意：只有冗余 S7 连接才可以有备用连接。
DIS_PCON	BOOL	上次调用 SFC 后，如果 CON_ARR[i].STAT_CON 发生 W#16#12 -> W#16#13 和 W#16#13 -> W#16#12 跳转，则 CON_ARR[i].DIS_PCON 将被设为 1。所有其它连接状态的变化不会影响 CON_ARR[i].DIS_PCON。 注意： <ul style="list-style-type: none"> <li>• 如果 MODE=B#16#01 和 02，则将连接数据复制到目标区域时，将复位与 DIS_PCON 对应的操作系统位。</li> <li>• MODE=B#16#03 时，与 DIS_PCON 对应的操作系统位将保持不变。</li> </ul>
RES0	BYTE	保留(B#16#00)
RES1	BYTE	保留(B#16#00)

## 错误信息

错误代码 (W#16#...)	描述
0000	<ul style="list-style-type: none"> <li>MODE=B#16#00、01 或 02：自上次调用后无连接状态变化(结构元素 STAT_CON)。已执行调用，未出错。</li> <li>MODE=B#16#03：已执行复制过程，未出错。</li> </ul>
0001	<ul style="list-style-type: none"> <li>MODE=B#16#00、01 或 02：自上次调用后至少有一个连接发生连接状态变化(结构元素 STAT_CON)。已执行作业，未出错。</li> <li>MODE=B#16#03：RET_VAL W#16#0001 不可能。</li> </ul>
7000	REQ=0 的第一次调用。不能处理在 MODE 中指定的作业。BUSY 值为 0。
7001	REQ=1 的第一次调用。已初始化 MODE 中指定的作业。BUSY 值为 1
7002	中间调用(REQ 不相关)。作业仍在运行。BUSY 值为 1。
8080	MODE 参数中的非法值。
8081	CON_ARR 参数中的非法数据类型。
8082	CON_ARR 参数中的长度描述过短。SFC 未将任何数据复制到目标区域中。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 13.7 使用 SFC 103 "DP\_TOPOL"识别 DP 主站系统的总线拓扑

## 描述

调用具有诊断中继器的 SFC 103 "DP\_TOPOL"，以启动对所选 DP 主站系统的总线拓扑的识别。调用 SFC 103 时将寻址连接到 DP 主站系统的所有诊断中继器。

**注意**

在某个给定时间，只能识别一个 DP 主站系统的拓扑。

拓扑识别是线路出错时显示详细出错位置的先决条件。安装或更改 DP 主站系统的物理结构后，请通过 SFC 103 再次识别拓扑。

对物理结构的更改包括：

- 改变电缆长度
- 添加或删除节点或具有中继器功能的组件
- 改变节点地址

每个诊断中继器在拓扑表中输入结果(所有节点的 PROFIBUS 地址，及其到诊断中继器的绝对距离)。可以在 STEP 7 下读取这些值。

建议在 OB1 或监视狗中断 OB 中调用 SFC 103。S7-300 不允许在一个启动 OB 中进行此调用。



当执行 SFC 时，SFC 将由诊断中继器返回的出错消息写入到输出 DPR 和 DPRI 中。如果所选 DP 主站系统中有多个诊断中继器报告出错，则 SFC 只在 DPR 和 DPRI 中输入与第一个生成错误消息的诊断中继器相关的信息。调用 SFC 13 "DPNRM\_DG"或 STEP 7，可以读取完整的诊断信息。如果诊断中继器未报告任何错误，则输出 DPR 和 DPRI 的值为 NULL。

如果希望在出错后触发另一个拓扑确定，必须首先复位 SFC103。为此，通过设置 REQ=0 和 R=1 调用 SFC。

## 工作方式

SFC 103 "DP\_TOPOL"以异步方式工作，也就是说，它的执行需要调用几个 SFC。通过 REQ = 1 调用 SFC 103，可以启动总线拓扑的识别，通过 R=1 调用 SFC 103，可以中止这一过程。

输出参数 RET\_VAL 和 BUSY 中指示了作业状态，另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义。

### 注意

识别拓扑可能要花费几分钟的时间。

## 作业标识

输入参数 DP\_ID 确定一个请求。

如果在完成拓扑识别前再次调用 SFC 103 "DP\_TOPOL"，SFC 的进一步响应将取决于新请求是否与先前的请求相同：如果参数 DP\_ID 与尚未完成的作业相对应，则 SFC 调用将被解释为调用序列的一部分，将在 RET\_VAL 中输入值 W#16#7002。但如果是新作业，则会被 CPU 拒绝。

参数	声明	数据 - 类型	存储区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、 常数	REQ=1：启动总线拓扑的识别
R	INPUT	BOOL	I、Q、M、D、L、 常数	R=1：中止拓扑识别
DP_ID	INPUT	INT	I、Q、M、D、L、 常数	将要识别的拓扑所属的 DP 主站系统的 DP 主站系统标识符
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果该功能运行时出错，则返回值将包含故障代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：仍在进行总线拓扑识别。
DPR	OUTPUT	BYTE	I、Q、M、D、L	生成错误消息的诊断中继器的 PROFIBUS 地址
DPRI	OUTPUT	BYTE	I、Q、M、D、L	生成错误消息的诊断中继器的测量段： <ul style="list-style-type: none"> <li>位 0 = 1：段 DP2 有临时中断</li> <li>位 1 = 1：段 DP2 有永久中断</li> <li>位 4 = 1：段 DP3 有临时中断</li> <li>位 5 = 1：段 DP3 有永久中断</li> </ul>
<p><b>永久错误：</b>系统检测到阻止成功进行拓扑识别的永久网络错误。可以通过 SFC 13 "DPNRM_DG"或 STEP 7 获取诊断信息，以查看细节。</p> <p><b>临时错误：</b>系统检测到防止成功进行拓扑识别的临时网络错误。这可能是由于不良接触或反复错误造成的。这些中断的特性使得无法确定错误源的准确位置。</p>				

### 错误信息

看到下表中的"真实"错误信息(错误代码 W#16#8xyz)时，我们应区分两类事件：

- 临时错误(错误代码 W#16#80A2 - 80A4、80C3、80C5)：  
此类错误可能无需用户操作即可消除，这意味着再次调用 SFC 是有帮助的(如必要，则多次调用)。  
临时错误的实例：所需资源当前正在使用中(W#16#80C3)。
- 永久错误(错误代码 W#16#8082、80B0、80B2)：  
此类型的错误不能自行更正。只有在消除此错误后，SFC 的新调用才有意义。永久错误实例：  
DP 主站/CPU 不支持此服务。(W#16#80B0)。

错误代码 (W#16#...)	解释
0000	作业已完成，未出错。
7000	REQ=0 的第一次调用。未启动总线拓扑识别。BUSY = 0。
7001	REQ=1 的第一次调用。已请求总线拓扑识别。BUSY = 1。
7002	中间调用(与 REQ 无关)：仍在进行拓扑识别。BUSY = 1。
7010	已请求取消拓扑识别。但没有与指定的 DP_ID 匹配的正在处理中的作业。BUSY = 0。
7011	初始调用，R=1。已启动取消拓扑识别的请求。BUSY = 1。
7012	中间调用：尚未完成拓扑识别。BUSY = 1。
7013	最终调用：已取消拓扑识别。BUSY = 0。
8082	没有以指定 DP_ID 组态的 DP 主站系统。
80A2	拓扑识别期间出错；要了解更多信息，请参考输出参数 DPR 和 DPRI。
80A3	拓扑识别期间出错：监视狗超时。
80A4	am K 总线上的通讯错误
80B0	DP 主站/CPU 不支持此服务。
80B2	拓扑识别期间出错：在所选 DP 主站系统中未找到诊断中继器。
80C3	所需资源当前正在使用。可能原因：您启动了第二个拓扑识别周期(在某给定时间只允许一个拓扑识别周期)或 H CPU 正在执行连接或更新操作。
80C5	DP 主站系统当前不可用。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

# 14 用于更新过程映像和处理位域的 SFC 和 SFB

## 14.1 使用 SFC 26 "UPDAT\_PI"更新过程映像输入表

### 描述

通过 SFC 26 "UPDAT\_PI" (更新过程映像), 可以更新 OB1 过程映像输入表(=过程映像段 0)或通过 STEP 7 定义的过程映像输入段。

如果您指定在出现 I/O 访问错误时重复发出信号以更新系统过程映像表, 则 SFC 26 将不断地更新所选的过程映像表。

否则, SFC 26 将只在系统不更新所选过程映像段时才更新过程映像表, 换句话说, 只在以下情况下进行更新:

- 未将此过程映像段分配给一个中断 OB 时, 或者
- 选择了过程映像段 0, 并在组态中禁用了 OB1 过程映像段的更新时。

---

### 注意

通过 STEP 7 分配给输入过程映像分区的每个逻辑地址不再属于 OB 1 的过程映像输入表。通过 SFC 26 更新输入过程映像分区时, 不能也通过 SFC 126 "SYNC\_PI"对其进行更新。

---

OB1 过程映像输入表及分配给某个中断 OB 的过程映像输入段的更新不受 SFC 26 调用的影响。

参数	声明	数据类型	存储器区	描述
PART	INPUT	BYTE	I、Q、M、D、L、 常数	要更新的过程映像输入段的编号。最大数值范围(取决于 CPU): 0 - 15 (0 表示 OB1 过程映像, n(其中, $1 \leq n \leq 15$ )表示过程映像段 n)
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
FLADDR	OUTPUT	WORD	I、Q、M、D、L	出现访问错误时引起错误的第一个字节的地址。

## 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	输入参数 PART 的值非法。
8091	指定的过程映像段未定义或不在 CPU 上允许的过程映像表区域中。
8092	系统通过 OB 更新了过程映像段，而您尚未组态所有 I/O 访问错误的重复发送信号。 SFC 26 "UPDAT_PI"未更新过程映像
80A0	更新过程中检测到访问错误。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

### 注意

对于定义了大于 32 个字节的一致性区域的 DP 标准从站，如果将 SFC 26 "UPDAT\_PI"用于其过程映像分区，则也可以获得来自 SFC 14 "DPRD\_DAT"的错误代码。

## 14.2 使用 SFC 27 "UPDAT\_PO"更新过程映像输出表

### 描述

通过 SFC 27 "UPDAT\_PO" (更新过程输出), 可以将 OB1 过程映像输出表(=过程映像段 0)或通过 STEP 7 定义的过程映像段的信号状态传送到输出模块。

如果已为该部分过程映像指定一致性范围, 则相应数据将一致地传送到相应外围模块。

### 注意

通过 STEP 7 分配给输出过程映像分区的每个逻辑地址不再属于 OB 1 的过程映像输出表。通过 SFC 27 更新输出过程映像分区时, 不能也通过 SFC 127 "SYNC\_PO"对其进行更新。

OB1 过程映像输出表及分配给中断 OB 的过程映像输出段的传送不受 SFC 27 调用的影响。

参数	声明	数据类型	存储器区	描述
PART	INPUT	BYTE	I、Q、M、D、L、 常数	要更新的过程映像输出段的编号。最大数值范围(取决于 CPU): 0 - 15 (0 表示 OB1 过程映像, n(其中, $1 \leq n \leq 15$ )表示过程映像段 n)
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
FLADDR	OUTPUT	WORD	I、Q、M、D、L	出现访问错误时引起错误的第一个字节的地址。

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	输入参数 PART 的值非法。
8091	指定的过程映像段未定义或不在 CPU 上允许的过程映像区中。
80A0	更新过程中检测到访问错误。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误

### 注意

对于定义了大于 32 个字节的一致性区域的 DP 标准从站, 如果将 SFC 27 "UPDAT\_PO"用于其过程映像分区, 则也可以获得来自 SFC 15 "DPWR\_DAT"的错误代码。

## 14.3 使用 SFC 126 "SYNC\_PI"在同步循环中更新过程映像分区输入表

### 描述

通过 SFC 126 "SYNC\_PI",可以在同步循环中更新过程映像分区输入表。链接到 DP 周期或 PN 发送时钟的用户程序可使用该 SFC 来一致性地同步更新过程映像分区中的输入数据。

SFC 126 可以被中断,且只能在 OB 61、62、63 和 64 中调用。

### 注意

只有当在 HW Config 中将受影响的过程映像分区分配给相关 OB 后,才允许在 OB 61 到 64 中调用 SFC 126 "SYNC\_PI"。通过 SFC 126 更新过程映像分区输入表时,不能也通过 SFC 26 "UPDAT\_PI"对其进行更新。

### 参数

参数	声明	数据类型	值范围	默认	含义
PART	INPUT	BYTE	1 - 30		要在同步循环中更新的过程映像分区输入表的编号。
RET_VAL	OUTPUT	INT			错误信息
FLADDR	OUTPUT	WORD			出现访问错误时引起错误的第一个字节的地址。

### 错误信息

事件等级错误代码	解释
W#16#0001	一致性警告。过程映像分区的更新分布在两个 DP 或 PN 周期上。但从站或 IO 设备中的数据将始终如一地被传送。
W#16#8090	参数 PART 中包含非法值和/或在此 OB 中不允许更新指定的过程映像分区输入表。未更新过程映像分区输入表。
W#16#8091	指定的过程映像分区尚未定义或不在 CPU 的可访问过程映像区域中。未更新过程映像分区输入表。
W#16#80A0	更新过程中检测到访问错误。受影响的输入被设置为"0"。
W#16#80A1	更新时间晚于允许的存取窗口。未更新输入的过程映像分区。 DP 或 PN 周期太短,不能确保正确执行 SFC 所需的时间。需要在 STEP 7 中增加时间 TDP (也被称为 T_DC)、Ti 和 To。

事件等级错误代码	解释
W#16#80A2	带一致性警告的访问错误 在更新指定的输入过程映像输出分区时检测到访问错误，同时发出一个一致性警告。 <ul style="list-style-type: none"> <li>• 未从 IO 读取故障输入的数据。在输入的过程映像分区中，受影响的输入被设为零。</li> <li>• 不受访问错误影响的输入错误更新过程跨越了两个 DP 周期。</li> </ul>
W#16#80C1	更新时间早于允许的存取窗口。未更新过程映像分区输入表。
W#16#8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

**注意**

对于定义了大于 32 个字节的一致性区域的 DP 标准从站，如果将 SFC 126 "SYNC\_PI"用于其过程映像分区，则也可以获得来自 SFC 14 "DPRD\_DAT"的错误代码。

## 14.4 使用 SFC 127 "SYNC\_PO"在同步循环中更新过程映像分区

### 描述

通过 SFC 127 "SYNC\_PO"，可以在同步循环中更新过程映像分区输出表。链接到 DP 周期或 PN 发送时钟的用户程序可使用该 SFC 来同步更新过程映像分区中的输出数据，并将它们一致性地传送到 I/O 设备。

SFC 127 可以被中断，且只能在 OB 61、62、63 和 64 中调用。

为确保稳定可靠地执行 SFC 127，应满足下列条件(仅限 S7-400)：

- DP 时钟周期 > 大约 5.0 毫秒
- 从站更新时间 < DP 时钟周期 - 4.0 毫秒

### 注意

只有当在 HW Config 中将受影响的过程映像分区分配给相关 OB 后，才允许在 OB 61 到 64 中调用 SFC 127 "SYNC\_PO"。通过 SFC 127 更新过程映像分区。通过 SFC 27 更新输出过程映像分区时，不能也通过 SFC 27 "UPDAT\_PO"对其进行更新。

### 参数

参数	声明	数据类型	值范围	默认	含义
PART	INPUT	BYTE	1 - 30		要在同步循环中更新的过程映像分区输出表的编号。
RET_VAL	OUTPUT	INT			如果处理此功能时出错，则返回值将包含错误代码。
FLADDR	OUTPUT	WORD			引起错误的第一个字节的地址。



## 错误信息

事件等级错误代码	解释
W#16#0001	一致性警告。过程映像分区的更新分布在两个 DP 或 PN 周期上。但从站或 IO 设备中的数据将始终如一地被传送。
W#16#8090	参数 PART 中包含非法值和/或在此 OB 中不允许更新指定的过程映像分区输出表。未将输出传送到 I/O 设备中。未更改过程映像分区输出表。
W#16#8091	指定的过程映像分区尚未定义或不在 CPU 的可访问过程映像区域中。未将输出传送到 I/O 设备中。未更改过程映像分区输出表。
W#16#80A0	在更新指定的过程映像输出分区时，删除了一个访问错误。未将错误输出传送到 I/O 设备中。这些输出在过程映像输出分区中保持不变。
W#16#80A1	带一致性警告的访问错误 访问错误已删除，同时，在更新指定的过程映像输出分区时，发出一个一致性警告。 <ul style="list-style-type: none"> <li>• 错误输出的数据未传送给 I/O。受影响的输出在过程映像输出分区中保持不变。</li> <li>• 不受访问错误影响的输出错误更新过程跨越了两个 DP 周期。</li> </ul>
W#16#80A2	更新时间晚于允许的存取窗口或者 DP 主站未更新输出数据。未将输出传送到 I/O 设备中。未更改过程映像分区输出表。
W#16#80C1	更新时间早于允许的存取窗口。未将输出传送到 I/O 设备中。过程映像分区输出表保持原样。
W#16#8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

**注意**

对于定义了大于 32 个字节的一致性区域的 DP 标准从站，如果将 SFC 127 "SYNC\_PO"用于其过程映像分区，则也可以获得来自 SFC 15 "DPWR\_DAT"的错误代码。

## 14.5 使用 SFC 79 "SET"在 I/O 区域中设置位域

### 描述

调用 SFC 79 "SET" (设置输出范围)具有以下作用：

- 置位使用参数 N 和 SA 选择的外围 I/O 区域中的位域。
- 同时也置位过程映像输出表中的相应位，而无论它们是否在过程映像段中。

位域必须是分配给过程映像的外围 I/O 区域的组成部分。

对于所选位域部分，如果没有插入模块，则 SFC 79 仍将尝试置位整个位域。然后在 RET\_VAL 中返回相应的错误信息。

---

### 注意

执行 SFC 79 时，总是将全部字节写入 I/O 区域中。

---

如果以参数 N 和 SA 选择的位域不以字节界限起始，则调用 SFC 79 具有以下作用：

- 要传送到外围 I/O 区域的第一个和最后一个字节中的，并且不属于所选位域中的位，包含了过程映像输出表中相应位的值。这可以导致意外的响应，如启动电机或关闭冷却系统。
- 属于所选位域的位按上述说明置位。

如果将 0 值赋给参数 N，则调用 SFC 79 将不会产生任何作用。如果未置位主控继电器，则调用 SFC 79 将不会产生任何作用。

参数	声明	数据类型	存储器区	描述
N	INPUT	INT	I、Q、M、D、L、常数	要置位的位的编号
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
SA	OUTPUT	POINTER	P	指向要置位的第一个位的指针

### 错误信息

评估参数 RET\_VAL 的错误信息的方法在使用输出参数 RET\_VAL 评估错误中进行了介绍。此章还包括 SFC 的常规错误信息。SFC 79 不通过 RET\_VAL 参数提供任何特定错误信息。

## 14.6 使用 SFC 80 "RSET"复位 I/O 区域中的位域

### 描述

调用 SFC 80 "RSET" (复位输出范围)具有以下作用：

- 复位通过参数 N 和 SA 复位选择的外围 I/O 区域中的位域。
- 同时也复位过程映像输出表中的相应位，而无论它们是否在过程映像段中。

位域必须位于分配给过程映像的外围 I/O 区域中。

对于所选位域部分，如果没有插入模块，则 SFC 80 仍将尝试复位整个位域。然后在 RET\_VAL 中返回相应的错误信息。

---

### 注意

执行 SFC 80 时，总是将全部字节写入外围 I/O 区域中。

---

如果以参数 N 和 SA 选择的位域不是以字节界限起始，则调用 SFC 80 具有以下作用：

- 要传送到外围 I/O 区域的第一个和最后一个字节中的，并且不属于所选位域中的位，包含了过程映像输出表中相应位的值。这可以导致意外的响应，如启动电机或关闭冷却系统。
- 属于所选位域的位按上述说明置位。

如果将 0 值赋给参数 N，则调用 SFC 80 将不会产生任何作用。如果未设置主控继电器，则调用 SFC 80 将不会产生任何作用。

参数	声明	数据类型	存储器区	描述
N	INPUT	INT	I、Q、M、D、L、常数	要复位的位的编号
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
SA	OUTPUT	POINTER	P	指向要复位的第一个位的指针

### 错误信息

评估参数 RET\_VAL 的错误信息的方法在使用输出参数 RET\_VAL 评估错误中进行了介绍。此章还包括 SFC 的常规错误信息。SFC 80 不通过 RET\_VAL 参数提供任何特定错误信息。

## 14.7 使用 SFB 32 "DRUM"实现操作序列

### 描述

SFB 32 "DRUM"可以实现一个最多 16 步的操作序列。通过参数 DSP 指定第一步的编号，通过参数 LST\_STEP 指定最后一步的编号。

在每个操作步骤中，将写入所有 16 个输出位 OUT0 到 OUT15 和输出参数 OUT\_WORD (在其中将输出位汇集在一起)。将用指定的 OUT\_VAL 数组的相应位或前一步骤的相应输出位的值为输出位赋值。赋哪个值取决于 S\_MASK 参数中屏蔽位的设置(参见下表)。

---

#### 注意

屏蔽位的默认值为 0。如果要更改一个或多个屏蔽位的当前设置，可以在背景数据块中实施改动。

---

当与先前的 SFB 调用相比，JOG 输入出现上升沿时，SFB 32 "DRUM"将切换到下一步骤。如果 SFB 已到达最后一步，则 JOG 的上升沿将置位变量 Q 和 EOD；DCC 的值为 0；SFB 将停留在最后的步骤中，直至 RESET 输入上设置 1。

也可以通过参数赋值，使系统根据时间切换到下一步。为此，必须将 DRUM\_EN 参数设置为 1。然后，操作序列在下列情况下切换到下一步骤：

- 用于当前步骤的事件位 EVENT 被置位，和
- 为当前步骤设定的时间已过。

此时间是对当前步骤有效的 DTBP 时间基准和时间因数(来自 S\_PRESET 数组)的乘积

---

#### 注意

只有当置位相应的事件位 EVENT 后，当前步骤(DCC)中剩余的执行时间才会减少。

---

如果在调用 SFB 时在 RESET 输入端设置 1，则操作序列将转到分配给 DSP 输入的步骤。

---

#### 注意

如果将 DRUM\_EN 设置为 1，则可以实现以下特殊情况：

- 完全根据时间启用各步骤(通过选择 EVENT = 1，其中  $DSP \leq i \leq LST\_STEP$ )。
- 完全根据事件启用各步骤(通过将 DTBP 的值设置为 0，使用事件位 EVENT 来选择)。

也可以通过 JOG 输入，随时(即使 DRUM\_EN=1)移动到操作序列中的下一个步骤。

---

第一次调用块时，必须在 RESET 输入中设置 1。

操作序列处于最后一个步骤(DSC 的值为 LST\_STEP)且此步骤的执行时间已过时，将置位输出 Q 和 EOD，并且 SFB 将停留在最后一步，直到在 RESET 输入中设置 1。

DRUM 定时器只在 STARTUP 和 RUN 模式下运行。

在冷启动过程中，操作系统将复位 SFB 32 "DRUM"，但在暖启动过程中则不复位。如果要在暖启动后初始化 SFB 32 "DRUM"，可以在 OB100 中通过 RESET = 1 来调用它。

参数	声明	数据类型	存储器区	描述
RESET	INPUT	BOOL	I、Q、M、D、L、常数	信号电平 1 将复位操作序列。第一次调用块时，必须将 RESET 设置为 1。
JOG	INPUT	BOOL	I、Q、M、D、L、常数	如果操作序列尚未进入最后一步，则上升沿(与上一次 SFB 调用相比)会将其切换到下一个步骤。将根据您赋给 DRUM_EN 的值启用下一个步骤。
DRUM_EN	INPUT	BOOL	I、Q、M、D、L、常数	控制参数，指定是否能够根据时间切换到下一个步骤(1：能够根据时间切换)
LST_STEP	INPUT	BYTE	I、Q、M、D、L、常数	最后一步的编号；可能的数值：1 - 16
EVENT, 1 < i < 16	INPUT	BOOL	I、Q、M、D、L、常数	事件位号 i (属于步骤 i)
OUT <sub>j</sub> , 0 < j < 15	OUTPUT	BOOL	I、Q、M、D、L	输出位号 j (与 OUT_WORD 的位号 j 相同)
Q	OUTPUT	BOOL	I、Q、M、D、L	状态参数，指定为最后一个步骤选择的执行时间是否已过去。
OUT_WORD	OUTPUT	WORD	I、Q、M、D、L、P	将输出位汇集到一起形成的变量
ERR_CODE	OUTPUT	WORD	I、Q、M、D、L、P	如果执行 SFB 期间出错，则 ERR_CODE 将包含错误信息。
JOG_HIS	VAR	BOOL	I、Q、M、D、L、常数	(与用户无关：上一个 SFB 调用的 JOG 输入参数)
EOD	VAR	BOOL	I、Q、M、D、L、常数	与输出参数 Q 相同
DSP	VAR	BYTE	I、Q、M、D、L、P、常数	第一步的编号；可能的数值：1 - 16
DSC	VAR	BYTE	I、Q、M、D、L、P、常数	当前步骤的编号
DCC	VAR	DWORD	I、Q、M、D、L、P、常数	当前步骤中仍剩余的执行时间(毫秒)(仅当 DRUM_EN = 1 且将相应事件位设置为 = 1 时才相关)
DTBP	VAR	WORD	I、Q、M、D、L、P、常数	对所有步骤有效的基准时间(以毫秒为单位)
PREV_TIME	VAR	DWORD	I、Q、M、D、L、常数	(与用户无关：上一个 SFB 调用的系统时间)
S_PRESET	VAR	ARRAY of WORD	I、Q、M、D、L、常数	包含每个步骤的时间因数的一维数组。有效下标为：[1 - 16]。这种情况下，S_PRESET [x]包含步骤 x 的时间因数。

参数	声明	数据类型	存储器区	描述
OUT_VAL	VAR	ARRAY of BOOL	I、Q、M、D、L、常数	包含在每个步骤中输出的数值的二维数组(如果尚未使用 S_MASK 进行屏蔽)。有效下标为：[1 - 16、0 - 15]。这种情况下，OUT_VAL [x, y]包含在步骤 x 中赋给输出位 OUTy 的值。
S_MASK	VAR	ARRAY of BOOL	I、Q、M、D、L、常数	包含每个步骤的屏蔽位的二维数组。有效下标为：[1 - 16、0 - 15]。这种情况下，S_MASK [x, y]包含将在步骤 x 中输出的第 y 个值的屏蔽位。 屏蔽位的含义： <ul style="list-style-type: none"> <li>• 0: 上一步骤的值已赋给相应的输出位。</li> <li>• 1: 来自 OUT_VAL 的相应值已赋给相应的输出位。</li> </ul>

### 错误信息

如果出现下表所列情况中的一种，SFB 32 "DRUM"将保持其当前状态不变，并将置位 ERR\_CODE 输出。

错误代码 (W#16#...)	解释
0000	无错误
8081	LST_STEP 的值非法
8082	DSC 的值非法
8083	DSP 的值非法
8084	乘积 $DCC = DTBP * S\_PRESET[DSC]$ 超出值 $2^{**}32-1$ (大约 24.86 天)

## 15 用于寻址模块的系统功能

### 15.1 使用 SFC 5 "GADR\_LGC"查询模块的逻辑基址

#### 描述

基于信号模块的通道，可以知道相应模块的插槽和用户数据地址区域中的偏移量。通过 SFC 5 "GADR\_LGC" (将地理地址转换为逻辑地址)，可以获得模块的相应逻辑地址，即最小输入和输出地址。若在电源模块或带压缩地址的模块(ET 200S)上使用 SFC 5，则返回诊断地址。

参数	声明	数据类型	存储器区	描述
SUBNETID	INPUT	BYTE	I、Q、M、D、L、常数	区域标识符： <ul style="list-style-type: none"> <li>0，如果插槽位于机架 0(中央机架)或 1 到 21(扩展机架)中(其中之一)。</li> <li>相应的分布式 I/O 系统的 DP 主站系统 ID (若插槽位于分布式 I/O 设备中)。</li> </ul>
RACK	INPUT	WORD	I、Q、M、D、L、常数	<ul style="list-style-type: none"> <li>机架号(如果区域标识符为 0)。</li> <li>分布式 I/O 设备的站号(如果区域标识符 &gt; 0)。</li> </ul>
SLOT	INPUT	WORD	I、Q、M、D、L、常数	插槽号
SUBSLOT	INPUT	BYTE	I、Q、M、D、L、常数	子模块插槽(如果没有可以插入的子模块，则必须在此指定 0)
SUBADDR	INPUT	WORD	I、Q、M、D、L、常数	模块的用户数据地址区域中的偏移量
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
IOID	OUTPUT	BYTE	I、Q、M、D、L	区域标识符： <ul style="list-style-type: none"> <li>B#16#54：外围输入(PI)</li> <li>B#16#55：外围输出(PO)</li> </ul> 如果为混合模块，则 SFC 将提供较小地址的区域标识符。 如果地址相同，则 SFC 将提供标识符 B#16#54。
LADDR	OUTPUT	WORD	I、Q、M、D、L	模块的逻辑基址

## 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8093	SUBNETID 参数值非法(SFC 5 不允许用于 PROFINET IO)。
8094	未通过指定的 SUBNETID 组态任何子网。
8095	RACK 参数的值非法。
8096	SLOT 参数的值非法。
8097	SUBSLOT 参数的值非法。
8098	SUBADDR 参数的值非法。
8099	未组态插槽。
809A	未组态所选插槽的子地址(仅对于 CPU 的中央 IO 设备和 IM 有效)。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 15.2 使用 SFC 49 "LGC\_GADR"查询属于逻辑地址的模块插槽

## 描述

通过 SFC 49 "LGC\_GADR" (将逻辑地址转换为地理地址)，可以获得属于逻辑地址的模块插槽及模块的用户数据地址区域中的偏移量。

## 注意

不能将 SFC 49 "LGC\_GADR"用于使用组合地址(ET 200S)的子模块。

参数	声明	数据类型	存储器区	描述
IOID	INPUT	BYTE	I、Q、M、D、L、常数	地址区域的 ID : <ul style="list-style-type: none"> <li>B#16#00 : LADDR 的第 15 位指定输入(位 15=0)或输出地址(位 15=1)是否存在。</li> <li>B#16#54 = 外设输入(PI)</li> <li>B#16#55 = 外围输出(PO)</li> </ul> 若是混合模块，指定最低地址的区域 ID。如果两个地址相同，则指定 B#16#54。
LADDR	INPUT	WORD	I、Q、M、D、L、常数	逻辑地址。对于混合模块，则指定两个地址中较低的一个。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
AREA	OUTPUT	BYTE	I、Q、M、D、L	区域标识符：这将指定必须如何解释剩余的其它输出参数。
RACK	OUTPUT	WORD	I、Q、M、D、L	机架号



参数	声明	数据类型	存储器区	描述
SLOT	OUTPUT	WORD	I、Q、M、D、L	插槽号
SUBADDR	OUTPUT	WORD	I、Q、M、D、L	相应模块的用户数据地址区域中的偏移量。

**注意**

如果在故障安全模式中使用 S7-400H PLC，并在以参数 LADDR 调用 SFC 49 时指定连接的 DP 从站中的模块的逻辑地址，则将在 RACK 参数的高位字节中提供活动通道的 DP 主站系统标识符。如果不存在活动通道，则将相关 DP 主站系统的 DP 主站系统标识符输出到主 CPU。

**输出参数 AREA**

输出参数 AREA 指定必须采用何种方式解释输出参数 RACK、SLOT 和 SUBADDR(参见下表)。

AREA 的值	系统	RACK、SLOT 和 SUBADDR 的含义
0	S7-400	RACK：模块机架号。 SLOT：插槽号。 SUBADDR：逻辑地址与逻辑基址间的差值
1	S7-300	RACK：模块机架号。 SLOT：插槽号。 SUBADDR：逻辑地址与逻辑基址间的差值
2	DP	RACK：(低字节)站号 RACK：(高字节) DP 主站系统标识符 SLOT：站中的插槽号 SUBADDR：相应模块的用户数据地址区域中的偏移量。
3	S5 P 区域	RACK：模块机架号。 SLOT：适配室的插槽号 SUBADDR：S5 x 范围中的地址
4	S5 O 区域	RACK：模块机架号。 SLOT：适配室的插槽号 SUBADDR：S5 x 范围中的地址
5	S5 IM3 区域	RACK：模块机架号。 SLOT：适配室的插槽号 SUBADDR：S5 x 范围中的地址
6	S5 IM4 区域	RACK：模块机架号。 SLOT：适配室的插槽号 SUBADDR：S5 x 范围中的地址

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	指定的逻辑地址无效或者 IOID 参数的值非法
8093	此 SFC 不适用于通过 IOID 和 LADDR 选择的模块。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 15.3 使用 SFC 50 "RD\_LGADR"查询模块的所有逻辑地址

### 描述

从模块、中央子模块或 PNIO 中的子模块的逻辑地址开始。使用 SFC 50 "RD\_LGADR" (读模块逻辑地址)，可获取所有该模块或子模块所有已声明的逻辑地址。您先前已使用 STEP 7 将地址分配给模块或子模块。SFC 50 按升序将获取的逻辑地址输入到 PEADDR 域或 PAADDR 域中。

参数	声明	数据类型	存储器区	描述
IOID	INPUT	BYTE	I、Q、M、D、L、常数	区域标识符： <ul style="list-style-type: none"> <li>B#16#00：LADDR 的第 15 位指定输入 (位 15=0)或输出地址(位 15=1)是否存在。</li> <li>B#16#54 = 外设输入(PI)</li> <li>B#16#55 = 外设输出(PQ)</li> </ul>
LADDR	INPUT	WORD	I、Q、M、D、L、常数	一个逻辑地址
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
PEADDR	OUTPUT	ANY	I、Q、M、D、L	用于 PI 地址的域，域元素的数据类型必须为 WORD。
PECOUNT	OUTPUT	INT	I、Q、M、D、L	返回的 PI 地址数
PAADDR	OUTPUT	ANY	I、Q、M、D、L	用于 PO 地址的域，域的数据类型必须为 WORD。
PACOUNT	OUTPUT	INT	I、Q、M、D、L	返回的 PQ 地址数

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	指定的逻辑地址无效或者 IOID 参数的值非法。
80A0	输出参数 PEADDR 中的错误：域元素的数据类型不是 WORD。 (此错误代码仅对于 S7-400 和 CPU 318 存在。)
80A1	输出参数 PAADDR 中的错误：域元素的数据类型不是 WORD。 (此错误代码仅对于 S7-400 和 CPU 318 存在。)
80A2	输出参数 PEADDR 中的错误：指定的域不能容纳所有逻辑地址。
80A3	输出参数 PAADDR 中的错误：指定的域不能容纳所有逻辑地址。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 15.4 用 SFC 70 "GEO\_LOG"确定模块的起始地址

### 描述

假设：模块的相关模块插槽是通过信号模块通道获知的。使用 SFC 70"GEO\_LOG" (将物理地址转换成逻辑地址)，可以确定模块的相关起始地址，即最小的 I 地址或 Q 地址。

若在电源模块或带压缩地址的模块(ET 200S)上使用 SFC 70，则返回诊断地址。

### 参数

参数	声明	数据类型	存储器区	描述
MASTER	INPUT	INT	I、Q、M、D、L、 常数	区域标识符： <ul style="list-style-type: none"> <li>• 0，如果插槽位于机架 0-3 之一 (中央单元)</li> <li>• 1 到 32：相关现场设备的 DP 主站系统 ID (如果插槽位于 PROFIBUS 现场设备中)</li> <li>• 100 到 115：相关现场设备的 PROFINET IO 系统 ID (如果插槽位于 PROFINET 现场设备中)</li> </ul>
STATION	INPUT	INT	I、Q、M、D、L、 常数。	机架编号，如果区域 ID= 0 现场设备的站号(如果区域 ID> 0)
SLOT	INPUT	INT	I、Q、M、D、L、 常数	插槽号
SUBSLOT	INPUT	INT	I、Q、M、D、L、 常数	接口模块插槽(如果不能插入任何接口模块，请在此输入 0)
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
LADDR	OUTPUT	WORD	I、Q、M、D、L	模块的起始地址 LADDR 的第 15 位指示显示的是输入地址(位 15 = 0)还是输出地址(位 15 = 1)

## 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8094	未通过指定的 SUBNETID 组态任何子网。
8095	STATION 参数的值无效
8096	SLOT 参数的值无效
8097	SUBSLOT 参数的值无效
8099	未组态插槽。
809A	没有为选中插槽组态接口模块地址。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 15.5 用 SFC 71 "LOG\_GEO"确定属于逻辑地址的插槽

### 描述

SFC 71 "LOG\_GEO" (将逻辑地址转换成物理地址)允许您确定属于逻辑地址的模块插槽以及模块的用户数据区域中的偏移量。

### 参数

参数	声明	数据类型	存储器区	描述
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的任意逻辑地址 第 15 位指示显示的是输入地址 (位 15 = 0)还是输出地址(位 15 = 1)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
AREA	OUTPUT	INT	I、Q、M、D、L	区域标识符：指出如何解释其余参数。
MASTER	OUTPUT	INT	I、Q、M、D、L、 常数	区域标识符： <ul style="list-style-type: none"> <li>• 0，如果插槽位于机架 0-3 之一 (中央单元)</li> <li>• 1 到 32：相关现场设备的 DP 主站系统 ID (如果插槽位于 PROFIBUS 现场设备 中)</li> <li>• 100 到 115：相关现场设备的 PROFINET IO 系统 ID (如果插槽位于 PROFINET 现场设备中)</li> </ul>
STATION	OUTPUT	INT	I、Q、M、D、L	机架编号，如果区域 ID= 0 现场设备的站号(如果区域 ID> 0)
SLOT	OUTPUT	INT	I、Q、M、D、L	插槽号
SUBSLOT	OUTPUT	INT	I、Q、M、D、L	接口模块编号
OFFSET	OUTPUT	INT	I、Q、M、D、L	相关模块的用户数据区域中的偏移量

## AREA 输出参数

AREA 的值	系统	RACK、SLOT 和 SUBADDR 的含义
0	S7-400	MASTER : 0 STATION : 机架号 SLOT : 插槽号 SUBSLOT : 0 OFFSET : 逻辑地址和逻辑基址之间的差异。
1	S7-300	MASTER : 0 STATION : 机架号 SLOT : 插槽号 SUBSLOT : 0 OFFSET : 逻辑地址和逻辑基址之间的差异。
2	PROFIBUS DP	MASTER : DP 主站系统 ID STATION : 站号 SLOT : 站中的插槽号 SUBSLOT : 0 OFFSET : 相关模块的用户数据地址区域中的偏移量
	PROFINET IO	MASTER : PROFINET IO 系统 ID STATION : 站号 SLOT : 站中的插槽号 SUBSLOT : 接口模块编号 OFFSET : 相关模块的用户数据地址区域中的偏移量
3	S5-P 区域	MASTER : 0 STATION : 机架号 SLOT : 适配器模块的插槽号 SUBSLOT : 0 OFFSET : S5 x 区域中的地址
4	S5-Q 区域	MASTER : 0 STATION : 机架号 SLOT : 适配器模块的插槽号 SUBSLOT : 0 OFFSET : S5 x 区域中的地址
5	S5-IM3 区域	MASTER : 0 STATION : 机架号 SLOT : 适配器模块的插槽号 SUBSLOT : 0 OFFSET : S5 x 区域中的地址
6	S5-IM4 区域	MASTER : 0 STATION : 机架号 SLOT : 适配器模块的插槽号 SUBSLOT : 0 OFFSET : S5 x 区域中的地址

## 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	指定的逻辑地址无效
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误





## 16 用于分布式 I/O 或 PROFINET IO 的 SFC

### 16.1 用 SFC 7 "DP\_PRAL"触发 DP 主站上的硬件中断

#### 描述

利用 SFC 7 "DP\_PRAL"，可以从智能从站的用户程序触发 DP 主站上的硬件中断。此中断将启动 DP 主站上的 OB40。

使用输入参数 AL\_INFO 能够识别硬件中断的原因。此中断标识符被传送至 DP 主站，您可以判断 OB40 (变量 OB40\_POINT\_ADDR)中的标识符。

被请求的硬件中断将由输入参数 IOID 和 LADDR 唯一确定。对于传送存储器中的每个已组态地址区域，无论何时都能准确地触发一个硬件中断。

#### SFC 的工作方式

SFC 7 "DP\_PRAL"以异步方式运行，换言之，它通过多个 SFC 调用来执行。通过使用 REQ=1 调用 SFC 7 启动硬件中断请求。

作业状态由输出参数 RET\_VAL 和 BUSY 指示，参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义。在 DP 主站上执行完 OB40 时，该作业便完成。

---

#### 注意

如果将 DP 从站作为标准从站运行，则当 DP 主站获得诊断帧后，作业便会立即完成。

---

## 识别作业

输入参数 IOID 和 LADDR 可唯一确定作业。

如果已在 DP 从站中调用了 SFC 7 "DP\_PRAL"，并在主站确认已请求的硬件中断之前再次调用此 SFC，则 SFC 的反应方式主要取决于新调用是否涉及同一个作业：如果参数 IOID 和 LADDR 与某个尚未完成的作业相匹配，便将 SFC 调用视为继续调用，而不考虑参数 AL\_INFO 的值，并将值 W#16#7002 输入到 RET\_VAL 中。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	REQ=1：DP 主站上属于从站的硬件中断
IOID	INPUT	BYTE	I、Q、M、D、L、常数	<p>传送存储器中地址区的标识符(用于 DP 从站的观察)：</p> <ul style="list-style-type: none"> <li>B#16#00：LADDR 的第 15 位指定输入(位 15=0)或输出地址(位 15=1)是否存在。</li> <li>B#16#54 = 外设输入(PI)</li> <li>B#16#55 = 外设输出(PQ)</li> </ul> <p>如果为混合模块，则必须指定较小地址的区域标识符。如果两个地址相同，则指定 B#16#54。</p>
LADDR	INPUT	WORD	I、Q、M、D、L、常数	<p>传送存储器中地址范围的起始地址(以 DP 从站的角度)。</p> <p>如果是属于混合模块的范围，则指定两个地址中的较低地址。</p>
AL_INFO	INPUT	DWORD	I、Q、M、D、L、常数	<p>中断 ID</p> <p>将被传送至要在 DP 主站上启动的 OB40 (变量 OB40_POINT_ADDR)。</p> <p>如果要运行带有远程主站的智能从站，必须对主站上的诊断帧进行判断。 (参见/70/)</p>
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在执行功能时出错，则返回值将包含故障代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	<p>BUSY = 1：</p> <p>触发的硬件中断尚未得到 DP 主站的确认。</p>

## 错误信息

错误代码 (W#16#...)	解释
0000	作业已执行且未出错。
7000	REQ=0 的第一次调用。硬件中断请求未激活；BUSY 的值为 0。
7001	REQ=1 的第一次调用。已将硬件中断请求发送至 DP 主站；BUSY 的值为 1。
7002	中间调用(REQ 无关联)：触发的硬件中断尚未得到 DP 主站的确认。BUSY 的值为 1。
8090	传送存储器中地址范围的起始地址不正确。
8091	中断被阻塞(由用户组态阻塞)
8093	参数 IOID 和 LADDR 将对不具备处理硬件中断请求功能的模块寻址。
80B5	不允许在 DP 主站中调用。
80C3	所需资源(内存等)此时已被占用。
80C5	分布式 I/O 设备此时不可用(即站出现故障)。
80C8	在当前 DP 主站工作模式下不允许使用该功能。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 16.2 用 SFC 11 "DPSYC\_FR"同步 DP 从站组

### 描述

利用 SFC 11 "DPSYC\_FR"，可以同步一组或多组 DP 从站。

该功能可将以下控制命令之一或这些命令的组合发送到相关组中：

- SYNC (同步输出并冻结 DP 从站的输出状态)
- UNSYNC (取消 SYNC 控制命令)
- FREEZE (冻结 DP 从站中从已冻结输入中读取的输入状态)
- UNFREEZE (取消 FREEZE 控制命令)

---

### 注意

注意，执行重新启动/暖启动时，控制命令 SYNC 和 FREEZE 仍保持有效。

另外请注意，在给定时间内只能发起一个 SYNC-/UNSYNC 请求或一个 FREEZE/UNFREEZE 请求。

---

### 要求

在发送上文列出的控制命令之前，必须使用 STEP 7 将 DP 从站分配给各个组(参见/231/)。必须了解将哪个 DP 从站分配给哪个组以及该组的编号，还要知道各个组对 SYNC/FREEZE 的反应。

### SFC 的工作方式

SFC 11 "DPSYC\_FR"是异步 SFC；换言之，它的执行要涉及多个 SFC 调用。通过使用 REQ=1 调用 SFC 11 启动作业。

作业状态由输出参数 RET\_VAL 和 BUSY 指示，另请参考异步 SFC 中参数 REQ、RET\_VAL 和 BUSY 的含义。

### 识别作业

如果已触发一个 SYNC/FREEZE 作业，且在第一个作业完成之前再次调用了 SFC 11，则 SFC 的响应取决于新调用是否针对同一个作业。如果输入参数 LADDR、GROUP 和 MODE 匹配，则 SFC 调用将被视为后续调用。

## 写 DP 模块输出

对 DP 模块输出的写操作可通过下列方式触发：

- 通过向 DP I/O 传送命令，
- 通过将过程映像输出表写入模块中(在 OB1 结束处通过操作系统或通过调用 SFC 27 "UPDAT\_PO")，
- 通过调用 SFC 15 "DPWR\_DAT"。

正常操作下，DP 主站会将输出字节周期性(在 PROFIBUS DP 总线周期内)传送给 DP 从站的输出。

如果希望同时将某些输出数据(可能分布于多个从站)应用于过程输出，可以使用 SFC 11 "DPSYC\_FR"向相关 DP 主站发送 SYNC 命令。

## SYNC 的作用

利用 SYNC 控制命令，选定组的 DP 从站将切换至"同步"模式。换言之，DP 主站传送当前输出数据并指示相关 DP 从站冻结各自输出。利用随后的输出帧，DP 从站将输出数据输入到内部缓冲区中，输出状态保持不变。

执行完每一条 SYNC 控制命令后，选定组的 DP 从站都会将其内部缓冲区中的输出数据应用到过程的输出中。

使用 SFC 11 "DPSYC\_FR"发送 UNSYNC 控制命令后，输出只是再次周期性更新。

---

### 注意

如果选定组的 DP 从站当前未连接到网络上或者在发送控制命令时出现故障，那么这些从站将不会切换至 SYNC 模式。SFC 的返回值中不会传达此信息。

---

## 读取 DP 模块的输入数据

按照下列方式读取 DP 模块的输入数据：

- 对 DP I/O 使用装载命令，
- 更新过程映像输入表时(启动 OB1 时通过操作系统或通过调用 SFC 26 "UPDAT\_PI")，
- 通过调用 SFC 14 "DPRD\_DAT"。

正常操作下，DP 主站将从其 DP 从站周期性(在 PROFIBUS DP 总线周期内)接收此输入数据，并向 CPU 提供这些数据。

如果希望同时从过程读取某些输入数据(可能分布于多个从站)，可以使用 SFC 11 "DPSYC\_FR"向相关 DP 主站发送 FREEZE 命令。

## FREEZE 的作用

利用 FREEZE 控制命令，可将 DP 从站切换到"冻结"模式，换言之，DP 主站指示 DP 从站冻结当前输入状态。然后，将冻结的数据传送到 CPU 的输入区域。

执行每条 FREEZE 控制命令后，DP 从站都会再次冻结各自的输入状态。

仅当用 SFC 11 "DPSYC\_FR"发送 UNFREEZE 控制命令后，DP 主站才再次开始周期性接收当前输入状态。

---

### 注意

如果选定组的 DP 从站当前未连接到网络上或者在发送控制命令时出现故障，那么这些从站将不会切换至 FREEZE 模式。SFC 的返回值中不会传达此信息。

---

## 数据一致性

由于 SFC 11 "DPSYC\_FR"功能是非周期性的且能被更高优先级中断，因而在使用 SFC 11 "DPSYC\_FR"时应确保过程映像与实际输入输出保持一致。

如果您遵循下列一致性规则，这一点将得到保证：

- 为"SYNC 输出"和"FREEZE 输入"定义适当的过程映像区域(仅适用于 S7-400)。首次调用 SYNC 作业之前调用 SFC 27 "UPDAT\_PO"。在最后一次调用 FREEZE 作业之后立即调用 SFC 26 "UPDAT\_PI"。
- 或者：对于与 SYNC 作业有关的输出和与 FREEZE 作业有关的输入，仅使用直接 I/O 访问。在 SYNC 作业处于激活状态时不得对这些输出进行写操作，在 FREEZE 作业处于激活状态时不得读取这些输入。

## 使用 SFC 15 和 SFC 14

如果使用 SFC 15 "DPWR\_DAT"，则必须在向相关输出发送 SYNC 作业之前完成此 SFC。

如果使用 SFC 14 "DPRD\_DAT"，则必须在向相关输入发送 FREEZE 作业之前完成此 SFC。

## SFC 11 "DPSYC\_FR"和启动

用户必须自行负责在启动 OB 中发送 SYNC 和 FREEZE 控制命令。

如果希望在启动用户程序后，一组或多组输出处于"同步"模式，则必须在启动期间初始化这些输出，并用 SYNC 控制命令完整执行 SFC 11 "DPSYC\_FR"。

如果希望在启动用户程序后，一组或多组输入处于 FREEZE 模式，则必须在启动期间用 FREEZE 控制命令对这些输入完整执行 SFC 11 "DPSYC\_FR"。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	电平触发的控制参数 REQ=1：触发 SYNC/FREEZE 作业
LADDR	INPUT	WORD	I、Q、M、D、L、常数	DP 主站的逻辑地址
GROUP	INPUT	BYTE	I、Q、M、D、L、常数	组选择 位 0 = 1：选择了组 1 位 1 = 1：选择了组 2 ： 位 7 = 1：选择了组 8 可以为每个作业选择多个组。 值 B#16#0 无效。
MODE	INPUT	BYTE	I、Q、M、D、L、常数	作业 ID (编码符合 EN 50 170 Volume 2 , PROFIBUS) 位 0：保留(值 0) 位 1：保留(值 0) 位 2 = 1：执行 UNFREEZE = 0：无意义 位 3 = 1：执行 FREEZE = 0：无意义 位 4 = 1：执行 UNSYNC = 0：无意义 位 5 = 1：执行 SYNC = 0：无意义 位 6：保留(值 0) 位 7：保留(值 0)  可能的值： <ul style="list-style-type: none"> <li>• 每个作业只有一个 ID： <ul style="list-style-type: none"> <li>- B#16#04 (UNFREEZE)</li> <li>- B#16#08 (FREEZE)</li> <li>- B#16#10 (UNSYNC)</li> <li>- B#16#20 (SYNC)</li> </ul> </li> <li>• 每个作业有多个 ID： <ul style="list-style-type: none"> <li>- B#16#14 (UNSYNC, UNFREEZE)</li> <li>- B#16#18 (UNSYNC, FREEZE)</li> <li>- B#16#24 (SYNC, UNFREEZE)</li> <li>- B#16#28 (SYNC, FREEZE)</li> </ul> </li> </ul>

参数	声明	数据类型	存储器区	描述
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错，则返回值将包含一个错误代码。 每次执行完块后都必须对 RET_VAL 进行判断。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1 : SYNC/FREEZE 作业尚未完成。

## 错误信息

### 注意

如果访问 DPV1 从站，这些从站的错误信息可从 DP 主站转发到 SFC。有关此错误信息的详细信息，请参见用 SFB 54 "RALRM"接收中断

错误代码 (W#16#...)	解释
0000	作业已完成且没有错误。
7000	REQ=0 的第一次调用。用 LADDR、GROUP 和 MODE 指定的作业未激活；BUSY 的值为 0。
7001	REQ=1 的第一次调用。用 LADDR、GROUP 和 MODE 指定的作业已被触发；BUSY 的值为 1。
7002	中间调用(与 REQ 无关)。已激活的 SYNC /FREEZE 作业仍处于激活状态；BUSY 的值为 1。
8090	用 LADDR 选定的模块不是 DP 主站。
8093	对于用 LADDR(DP 主站的组态或版本)选择的模块，不允许使用此 SFC。
8094	GROUP 参数无效
8095	MODE 参数无效
80B0	用 GROUP 选定的组未组态。
80B1	用 GROUP 选定的组未分配给此 CPU。
80B2	用 MODE 指定的 SYNC 作业不允许在用 GROUP 选定的组中使用。
80B3	用 MODE 指定的 FREEZE 作业不允许在用 GROUP 选定的组中使用。
80C2	DP 主站上出现临时资源短缺：DP 主站当前正在处理 CPU 所允许的最多作业数。
80C3	目前不能激活此 SYNC/UNSYNC 作业，原因是任何时候都只能触发一个 SYNC/UNSYNC 作业。检查用户程序。
80C4	目前不能激活此 FREEZE/UNFREEZE 作业，原因是任何时候都只能触发一个 FREEZE-/UNFREEZE 作业。检查用户程序。
80C5	在 DP 接口处直接发生短路
80C6	由于 CPU 断开 I/O 连接，作业被中止
80C7	由于 DP 主站上的暖启动或冷启动，作业被中止
8xyy	一般错误信息，请参见使用输出参数 RET_VAL 判断故障



## 16.3 用 SFC 12 "D\_ACT\_DP"激活和取消激活 DP 从站/PROFINET IO 设备

### 描述

利用 SFC 12 "D\_ACT\_DP"，可以明确地取消激活及再次激活已组态的 DP 从站/PROFINET IO 设备。此外，还可以确定当前是激活还是取消激活每个已分配的 DP 从站或 PROFINET IO 设备。

如果使用 SFC 12 取消激活 IE/PB Link PN IO 类型的网关，则连接的所有 PROFIBUS DP 从站也将停止运行。这些故障将被报告。

SFC 12 不能用在通过 DP/PA 链接与 DP 主站系统相连的 PROFIBUS PA 现场设备上。

---

### 注意

只要有一个或多个 SFC 12 作业在忙，便不能将已修改的组态从 PG 下载到 CPU 中(在 CiR 处理期间)。在 CiR 处理期间，CPU 在接收已修改组态的下载时，会拒绝 SFC 12 的启动请求。

执行禁用或启用作业需要多次经过循环控制点。因此，在一个编程周期中不可能等到此类作业的结束。

---

### 用途

如果在实际并不存在或非当前所需的 CPU 中组态 DP 从站/PROFINET IO 设备，则该 CPU 仍会继续以正常时间间隔访问这些 DP 从站/PROFINET IO 设备。而在取消激活这些从站后，则其它 CPU 访问也将停止。使用 PROFIBUS DP，有可能达到最快的 DP 总线周期，同时不再发生相应的错误事件。

### 应用实例

从机器制造厂商的角度看，在连续生产机器的过程中，有大量的设备选件可供使用。但交付的每台机器仅是选定组件的一个组合。

制造商将所有这些可能的机器选件均组态为 DP 从站/PROFINET IO 设备，以便创建并维护包含所有可能选件的通用用户程序。利用 SFC 12，可以取消激活所有 DP 从站/PROFINET IO 设备，这些从站在机器启动时并不存在。

与此类似的情况是机床，可以使用大量加工选件，但实际上常常用到的仅是其中的一小部分。这些工具被作为 DP 从站/PROFINET IO 设备来实现。利用 SFC 12，用户程序可以激活当前需要的工具而取消激活那些以后才需要的工具。

### SFC 的工作方式

SFC 12 "D\_ACT\_DP"以异步方式运行，换言之，它通过多个 SFC 调用来执行。通过使用 REQ=1 调用 SFC 12 启动请求。

作业状态由输出参数 RET\_VAL 和 BUSY 指示；(参见：异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

### 识别作业

如果已启动一个取消激活或激活的作业，并在该作业完成前再次调用 SFC 12，则 SFC 的反应方式主要取决于新调用是否涉及同一个作业：如果参数 LADDR 匹配，则 SFC 调用被解释为继续调用。

### 取消激活 DP 从站/PROFINET IO 设备

用 SFC 12 取消激活 DP 从站/PROFINET IO 设备后，其过程输出将被设置为已组态的替换值或 0 (安全状态)。所分配的 DP 主站/PROFINET IO 控制器不继续寻址该组件。DP 主站/PROFINET IO 控制器或 CPU 上的错误 LED 不会将取消激活的 DP 从站/PROFINET IO 设备识别为出现故障或丢失。

而取消激活的 DP 从站/PROFINET IO 设备输入的过程映像将用 0 更新，即将其作为有故障的 DP 从站/PROFINET IO 设备处理。

如果要使用程序直接访问先前取消激活的 DP 从站/PROFINET IO 设备的用户数据，将调用 I/O 访问错误 OB (OB 122)，并在诊断缓冲区中输入相应的启动事件。如果尝试使用 SFC (即 SFC 59 "RD\_REC")访问取消激活的 DP 从站/PROFINET IO 设备，则会在 RET\_VAL 中收到一条出错信息(与访问不可用的 DP 从站/PROFINET IO 设备相同)。

取消激活 DP 从站/PROFINET IO 设备不会启动程序错误 OB (OB 85)，即使其输入或输出属于要更新的系统范围的过程映像。诊断缓冲区中不输入任何条目。

取消激活 DP 从站/PROFINET IO 设备是否会启动导致在诊断缓冲区中输入一个条目的机架故障 OB (OB 86)，这取决于 MODE 参数。

MODE	OB 86 启动	诊断缓冲区条目
2	否	否
4	是	是

如果用 SFC 12 取消激活 DP 站/PNIO 站后该站出现故障，则操作系统不对该故障进行检测。因而不会后续启动 OB86 或诊断缓冲区条目。

适用于 PROFIBUS DP：如果希望取消激活作为交叉通讯中发送器的 DP 从站，建议您首先取消激活用于检测发送器要将哪些输入数据传送到 DP 主站的接收器(监听器)。仅当执行此步骤后才取消激活发送器。

## 激活 DP 从站/PROFINET IO 设备

当使用 SFC 12 重新激活 DP 从站/PROFINET IO 设备时，该组件由指定的 DP 主站/PROFINET IO 控制器进行组态和分配参数(作为有故障的 DP 站/PROFINET IO 站的返回)。组件能够传送用户数据时，激活操作便完成。

激活 DP 从站/PROFINET IO 设备不启动程序错误 OB (OB85)，即使其输入或输出属于要更新的系统范围的过程映像。并且不会向诊断缓冲区输入条目。

激活 DP 从站/PROFINET IO 设备是否会启动导致在诊断缓冲区中输入一个条目的机架故障 OB (OB 86)，这取决于 MODE 参数。

MODE	OB 86 启动	诊断缓冲区条目
1	否	否
3	是	是

如果尝试使用 SFC 12 激活一个曾被取消激活并已与 DP 总线物理断开的 DP 从站，则大约 1 分钟后，SFC 将返回错误代码 W#16#80A2。DP 从站仍保持取消激活状态。如果以后从站重新连接到 DP 总线，则必须用 SFC 12 重新激活它。

如果尝试激活一个已从 PROFINET 总线物理断开的 PROFINET IO 设备，则 SFC 12 保持激活状态。与 DP 从站一样，在指定时间后不会自动取消。您需要手动取消正在运行的作业。

### 注意

激活 DP 从站/PROFINET IO 设备可能需要花费一定时间。因此，如果希望取消当前的激活作业，可对 LADDR 使用同一个数值和 MODE=2 再次启动 SFC 12。重复调用 SFC 12，直至 RET\_VAL = 0 指示成功取消激活。

如果希望激活参与交叉通讯的 DP 从站，建议您首先激活发送器，然后激活接收器(监听器)。

## CPU 启动

根据启动模式的不同，CPU 操作系统对 DP 从站/PROFINET IO 设备的作用如下：

- 在冷启动和暖启动模式下，从站/设备被自动激活。
  - S7-400：激活 DP 从站/PROFINET IO 设备可能需要花费一些时间。这种情况下，CPU 会继续运行，但在激活完成之前将一直出现 I/O 访问错误。CPU 的反应取决于 OB 85 中的组态(OB 85 调用 I/O 访问错误)和程序。您无法制止这些 I/O 访问错误的发生。
  - S7-300：激活 DP 从站/PROFINET IO 设备可能需要花费一些时间。这种情况下，CPU 将处于等待状态，直到 DP 从站/PROFINET IO 设备被激活。激活 DP 从站/PROFINET IO 设备的最长等待时间大约为 1 分钟。在此时间内，不发生 I/O 访问错误。
- 在热启动模式下，从站/设备的激活状态保持不变，即已激活的从站/设备保持已激活状态，取消激活的从站/设备保持取消激活状态。

CPU 启动之后，将定期尝试联系所有已组态但未取消激活的从站/设备，这些从站或者不存在或者未响应。

### 注意

启动 OB 不支持对 SFC 12 的调用。

参数

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数。	电平触发的控制参数 REQ= 1 : 执行激活或取消激活
MODE	INPUT	BYTE	I、Q、M、D、L、常数	作业 ID。可能的值 : <ul style="list-style-type: none"> <li>• 0: 是激活还是取消激活已寻址组件的请求信息</li> <li>• 1: 激活 DP 从站/PROFINET IO 设备</li> <li>• 2: 取消激活 DP 从站/PROFINET IO 设备</li> <li>• 3: 激活 DP 从站/PROFINET IO 设备，并在激活状态改变后调用 OB 86。</li> <li>• 4: 取消激活 DP 从站/PROFINET IO 设备，并在激活状态改变后调用 OB 86。</li> </ul>
LADDR	INPUT	WORD	I、Q、M、D、L、常数	DP 从站的所有逻辑地址/PROFINET IO 设备 注意：如果地址是输出地址，则必须设置位 15
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在处理功能时出现错误，返回值将包含一个错误代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	激活代码 : <ul style="list-style-type: none"> <li>• BUSY = 1 : 作业仍处于激活状态。</li> <li>• BUSY=0 : 作业已被终止。</li> </ul>

## 错误信息

错误代码(W#16#...)	解释
0000	作业已完成且没有错误。
0001	DP 从站/PROFINET IO 设备已激活。(仅当 MODE = 0 时才可能出现此错误代码。)
0002	DP 从站/PROFINET IO 设备已被取消激活(仅当 MODE = 0 时才可能出现此错误代码)
7000	REQ=0 的第一次调用。用 LADDR 指定的作业未激活；BUSY 的值为 0。
7001	REQ=1 的第一次调用。用 LADDR 指定的作业已被触发；BUSY 的值为 1。
7002	中间调用(与 REQ 无关)。已激活的作业仍处于激活状态；BUSY 的值为 1。
8090	<ul style="list-style-type: none"> <li>尚未用 LADDR 中指定的地址组态模块。</li> <li>将 CPU 作为 I 从站运行，并且已在 LADDR 中为此 I 从站指定了一个地址。</li> </ul>
8092	要取消激活当前寻址的 DP 从站/PROFINET IO 设备(MODE=2)，不能通过将其激活(MODE=1)来取消。以后再激活组件。
8093	没有将任何 DP 从站/PROFINET IO 设备分配给 LADDR 中指定的地址(无可用组态)，或者 MODE 参数未知。
8094	尝试激活一台设备，它可能是工具切换端口的伙伴。此时，在该工具切换端口上已经激活了另外一台设备。已经激活的设备保持激活。
80A1	不能为所寻址的组件分配参数(仅当 MODE = 1 时才可能出现该错误代码)。 注意： 仅当已激活的从站在参数分配期间再次出现故障时，CPU 才提供此信息。如果单个模块的参数分配没有成功，则 SFC 将返回错误信息 W#16#0000。
80A2	已寻址 DP 从站未返回确认。(此错误信息对于 PROFINET IO 设备不可用。)不在 PROFINET 中对激活作业进行时间监视。
80A3	相关的 DP 主站/PROFINET IO 控制器不支持此功能。
80A4	CPU 不支持外部 DP 主站/PROFINET IO 控制器的这一功能。
80A6	DP 从站/PROFINET IO 设备中的插槽错误；并非所有用户数据均可访问(仅当 MODE=1 时才出现该错误代码)。 注意： 仅当激活的组件在参数分配之后 SFC 结束之前出现故障时，SFC 才会返回此错误信息。如果仅是一个模块不可用，则 SFC 将返回错误信息 W#16#0000。
80C1	SFC12 已启动，并用另一逻辑地址继续(仅当 MODE=1 和 MODE=2 时才可能出现此错误代码)。
80C3	<ul style="list-style-type: none"> <li>临时资源错误：CPU 当前正在处理允许的最多数目的激活和取消激活作业。(仅当 MODE = 1 和 MODE = 2 时才可能出现此错误代码。)</li> <li>CPU 正在忙于接收已修改的组态。当前无法启用/禁用 DP 从站/PROFINET IO 设备/PROFINET IO 设备。</li> </ul>
80C5	DP：用户尚未取得的作业将在重启期间放弃。
80C6	PROFINET：用户尚未取得的作业将在重启期间放弃。
8xyy	错误信息，参见通过输出参数 RET_VAL 判断错误

## 16.4 用 SFC 13 "DPNRM\_DG"读取 DP 从站的诊断数据(从站诊断)

### 从站诊断

每个 DP 从站都会提供一些结构符合"EN 50 170 Volume 2, PROFIBUS"的从站诊断数据。要读出此诊断数据，需要使用 SFC 13 "DPNRM\_DG"。

有关从站诊断数据的基本结构，请参考下表；更多信息，则请参考 DP 从站手册。

字节	含义
0	站状态 1
1	站状态 2
2	站状态 3
3	主站号
4	供应商 ID (高字节)
5	供应商 ID (低字节)
6 ...	更多从站特定诊断信息

### 描述

利用 SFC 13 "DPNRM\_DG"(读取 DP 从站的诊断数据)，可读取 DP 从站中格式符合"EN 50 170 Volume 2, PROFIBUS"的当前诊断数据。在经过无错数据传送之后，已读取的数据被输入到由 RECORD 指示的目标区域中。

通过将 1 分配给 SFC 13 调用中的输入参数 REQ 可开始读取作业。

## 功能

读取作业异步执行，换言之，它需要多个 SFC 13 调用。作业状态由输出参数 RET\_VAL 和 BUSY 指示，另请参考异步 SFC 中参数 REQ、RET\_VAL 和 BUSY 的含义。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	REQ=1：读请求
LADDR	INPUT	WORD	I、Q、M、D、L、常数	DP 从站的已组态诊断地址。 <b>注意：</b> 必须以十六进制格式输入地址。例如，诊断地址 1022 表示：LADDR:=W#16#3FE。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错，则返回值将包含一个错误代码。如果未出现错误，则实际传送的数据长度将输入到 RET_VAL 中。
RECORD	OUTPUT	ANY	I、Q、M、D、L	被读取诊断数据的目标区域。只允许数据类型 BYTE。要读取的数据记录的最小长度或目标区域为 6。要读取的数据区的最大长度为 240。对于其中标准诊断数据的数目大于 240 字节和最大 244 字节的标准从站，前面的 240 字节将被传送到目标区域，并在数据中设置相应溢出位。 <b>注意：</b> 注意，S7-300 CPU 的 RECORD 参数总是需要完整指定 DB 参数(例如：P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成一条出错消息。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1： 读取作业尚未完成。

### 输入参数 RECORD使用 SFC 59 "RD\_REC"读数据记录

CPU 按照以下方式判断已读诊断数据的实际长度：

- 如果为 RECORD 指定的长度小于提供的数据字节数，则数据将被放弃并在 RET\_VAL 中输入相应错误代码。
- 如果为 RECORD 指定的长度大于或等于提供的数据字节数，则目标区域将接受该数据并以正数形式在 RET\_VAL 中输入实际长度。

---

#### 注意

必须确保 RECORD 的实际参数在属于一个作业的所有调用中均匹配。  
作业由 LADDR 输入参数唯一识别。

---

### 诊断数据超出 240 个字节的标准从站

对于标准诊断数据字节数介于 241 和 244 之间的标准从站，请注意以下几点：

- 如果为 RECORD 指定的长度小于 240 个字节，则数据将被放弃，并在 RET\_VAL 中输入相应错误信息。
- 如果为 RECORD 指定的长度大于或等于 240 个字节，则标准诊断数据中的前 240 个字节将被传送到目标区域，并在数据中设置溢出位。

### 输出参数 RET\_VAL

- 如果在执行功能时出错，则返回值将包含故障代码。
- 如果在数据传送期间未出现错误，则 RET\_VAL 中将包含读取数据的长度(以正数表示的字节数)。

---

#### 注意

在 DP 从站中读取数据的数量取决于它的诊断状态。

---

### 错误信息

关于如何判断 RET\_VAL 参数的错误信息，请参见通过输出参数 RET\_VAL 判断错误中的解释。本章还包含了 SFC 的常规错误信息。SFC 13 特定的错误信息是 SFC 59 "RD\_REC"错误信息的子集，参见使用 SFC 59 "RD\_REC"读数据记录

### S7-400 的系统资源

如果某个当前未进行处理的作业调用 SFC 13 "DPNRM\_DG"，则将使用 S7-400 中的 CPU 资源(内存空间)。可以为多个 DP 从站连续调用 SFC 13，前提是未超过 CPU 允许的"同时"激活 SFC 13 作业的最大数目。在 I101 中可以找到此类作业的最大数目。

如果"同时"激活多个作业，则将执行所有这些作业，并且彼此不会干扰。

如果达到系统资源的限值，则将在 RET\_VAL 中指出。此时请重新执行作业。



## 16.5 用 SFC 14 "DPRD\_DAT"读取 DP 标准从站//PROFINET IO 设备的连续数据

### 数据一致性

请参考"S7 通信和 S7 基本通信概述 - 数据连续性"一节。

### SFC 14 的用途

需要 SFC 14 "DPRD\_DAT"的原因是使用用于访问 I/O 或过程映像输入表的装载指令最多只能读出四个连续字节。

---

#### 注意

如果需要，还可以通过输入的过程映像读取连续数据。要确定 CPU 是否支持此功能，请参考 "*S7-300 自动化系统，硬件与安装*"手册。

所有 S7-400 CPU 均支持此功能。

---

#### 当心

在使用 SFC 14 "DPRD\_DAT"时，应避免访问已分配了带有 OB6x 连接(时钟中断)的过程映像分区 I/O 区域。

---



### 描述

利用 SFC 14 "DPRD\_DAT" (读取 DP 标准从站的连续数据)，能读取 DP 标准从站/PROFINET IO 设备的连续数据。

要查找适用于 S7-300 CPU 的最大长度，请参考以下手册：*S7-300 自动化系统、硬件与安装*、*ET 200S 接口子模块*、*IM151-7 CPU*或*基本子模块 BM147CPU*。关于 S7-400 CPU，请参考手册 *S7-400 自动化系统的 CPU 数据*。如果在数据传送期间未出现错误，则已读取的数据将被输入到由 RECORD 标识的目标区域中。

目标区域的长度必须与用 STEP 7 为选定模块组态的长度相同。

如果从具有模块化设计或具有多个 DP 标识符的 DP 标准从站读取数据，则通过指定组态的起始地址，每个 SFC 14 调用只能访问一个模块/DP 标识符的数据。

参数	声明	数据类型	存储器区	描述
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的 I 区域中已组态的起始地址，将从该处读取数据。 <b>注意：</b> 必须以十六进制格式输入地址。例如，诊断地址 100 表示：LADDR:=W#16#64。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错，则返回值将包含一个错误代码。
RECORD	OUTPUT	ANY	I、Q、M、D、L	被读取用户数据的目标区域。必须与用 STEP 7 为选定模块组态的长度完全相同。只允许数据类型 BYTE。 <b>注意：</b> 注意，S7-300 CPU 的 RECORD 参数总是需要完整指定 DB 参数(例如：P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成一条出错消息。

## 错误信息

### 注意

如果访问 DPV1 从站，这些从站的错误信息可从 DP 主站转发到 SFC。有关此错误信息的详细信息，请参见用 SFB 54 "RALRM" 接收中断。

错误代码 (W#16#...)	解释
0000	未产生故障。
8090	<ul style="list-style-type: none"> <li>• 尚未为指定的逻辑基址组态模块或者</li> <li>• 忽略了关于连续数据长度的限制或者</li> <li>• 没有以十六进制格式在 LADDR 参数中输入起始地址。</li> </ul>
8092	在 ANY 引用中，指定了一个非 BYTE 的类型。
8093	在 LADDR 中指定的逻辑地址处没有可从中读取连续数据的 DP 模块/PROFINET IO 设备。
80A0	访问 I/O 设备时检测到访问错误。
80B0	外部 DP 接口模块从站出现故障。
80B1	指定目标区域的长度不等于用 STEP 7 组态的用户数据的长度。
80B2	外部 DP 接口模块出现系统错误。
80B3	外部 DP 接口模块出现系统错误。
80C0	模块尚未读取数据。
80C2	外部 DP 接口模块出现系统错误。
80Fx	外部 DP 接口模块出现系统错误。
87xy	外部 DP 接口模块出现系统错误。
808x	外部 DP 接口模块出现系统错误。
8xyy	一般错误信息，请参见使用输出参数 RET_VAL 判断故障

## 16.6 用 SFC 15 "DPWR\_DAT"向 DP 标准从站/PROFINET IO 设备写入连续数据

### 数据一致性

参考 "S7 通信和 S7 基本通信概述 - 数据连续性"一节。

### SFC 15 的用途

需要 SFC 15 "DPWR\_DAT"的原因是使用用于访问 I/O 或过程映像输入表的传送指令最多只能写入四个连续字节。

---

#### 注意

如果需要，还可以通过输入的过程映像读取连续数据。要确定 CPU 是否支持此功能，请参考 "S7-300 自动化系统，硬件与安装"手册。

所有 S7-400 CPU 均支持此功能。

当写入一致性数据时，不得同时使用两个选项：使用 SFC 15 或通过过程输出映像写入。

---

---

#### 当心

在使用 SFC 15 "DPWR\_DAT"时，应避免访问已分配了带有 OB6x 连接(时钟中断)的过程映像分区的 I/O 区域。

---



### 描述

利用 SFC 15 "DPWR\_DAT" (向 DP 标准从站写入连续数据)，可将 RECORD 中的数据连续地传送至已寻址的 DP 标准从站/PROFINET IO 设备，如必要，还可以传送至过程映像(即如果已将 DP 标准从站的各个地址区域组态为过程映像中的连续地址范围)。要查找 S7-300 CPU 允许的传输数据的最大长度，请参考以下手册：*S7-300 自动化系统、硬件与安装*、*ET 200S 接口子模块 IM151-7 CPU*或*基本子模块 BM147CPU*。关于 S7-400 CPU，请参考手册 *S7-400 自动化系统的 CPU 数据*。数据是同步传送的，即完成 SFC 时也完成了写作业。

源区域的长度必须与用 STEP 7 为选定模块组态的长度相同。

如果 DP 标准从站是模块化设计，则只能访问 DP 从站的一个模块。

参数	声明	数据类型	存储器区	描述
LADDR	INPUT	WORD	I、Q、M、D、L、 常数	模块的过程映像输出区域中已组态的起始地址，数据将被写入该地址。 <b>注意：</b> 必须以十六进制格式输入地址。例如，诊断地址 100 表示：LADDR:=W#16#64。
RECORD	INPUT	ANY	I、Q、M、D、L	要写入用户数据的源区域。必须与用 STEP 7 为选定模块组态的长度完全相同。只允许数据类型 BYTE。 <b>注意：</b> 注意，S7-300 CPU 的 RECORD 参数总是需要完整指定 DB 参数(例如：P#DB13.DBX0.0 byte 100)。忽略明确的 DB 编号对于 S7-300 CPU 是不允许的，这将会导致在用户程序中生成一条出错消息。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在功能激活时出错，则返回值将包含一个错误代码。

## 错误信息

**注意**

如果访问 DPV1 从站，这些从站的错误信息可从 DP 主站转发到 SFC。有关此错误信息的详细信息，请参见用 SFB 54 "RALRM"接收中断 STATUS[3]。

错误代码 (W#16#...)	解释
0000	未产生故障。
808x	外部 DP 接口模块出现系统错误。
8090	<ul style="list-style-type: none"> <li>• 尚未为指定的逻辑基址组态模块或者</li> <li>• 忽略了关于连续数据长度的限制或者</li> <li>• 没有以十六进制格式在 LADDR 参数中输入起始地址。</li> </ul>
8092	在 ANY 引用中，指定了一个非 BYTE 的类型。
8093	在 LADDR 中指定的逻辑地址处没有要写入连续数据的 DP 模块/PROFINET IO 设备。
80A1	访问 I/O 设备时检测到访问错误。
80B0	外部 DP 接口模块从站出现故障。
80B1	指定源区域的长度不等于用 STEP 7 组态的用户数据的长度。
80B2	外部 DP 接口模块出现系统错误。
80B3	外部 DP 接口模块出现系统错误。
80C1	模块中先前写作业中的数据尚未被模块处理。
80C2	外部 DP 接口模块出现系统错误。
80Fx	外部 DP 接口模块出现系统错误。
85xy	外部 DP 接口模块出现系统错误。
8xyy	一般错误信息，请参见使用输出参数 RET_VAL 判断故障

# 17 PROFINet

## 17.1 SFC 112、113 和 114 背景信息

---

### 注意

默认情况下，操作系统会更新循环检验点上的 PROFINET CBA 组件和 DP 互连。但是，如果在组态中禁用了这些自动更新(例如，为了提高 CPU 的时间响应)，您就必须自己执行更新。您还可以为此在合适的时间调用 SFC 112 至 114。

这些更新只能在组态中成组禁用，而不能单独禁用。

---

### 影子存储器

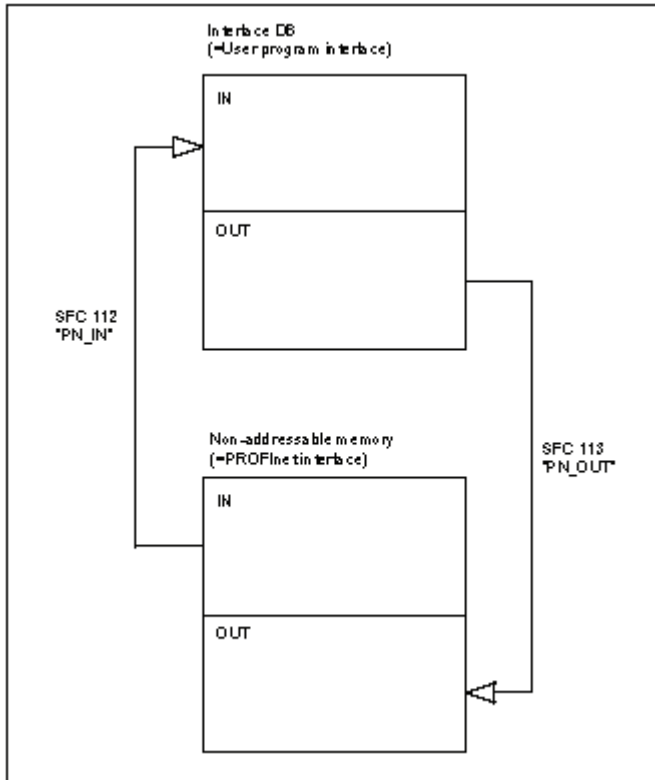
接口数据块是 PROFINET CBA 组件的用户程序接口。为确保关联程序部分运行时输入与输出一致，各个接口数据块都有一个由操作系统进行管理的结构完全相同的存储器区。该区叫做"影子存储器"。在程序中，您只能访问接口数据块，而其它(外部) PROFINET CBA 组件则只能访问影子存储器。这样安排可以防止在访问接口数据块的输入和输出时发生冲突。

### 更新 PROFINET CBA 组件

通过使用影子存储器实现的数据一致性意味着更新 PROFINET CBA 组件分以下两步进行：

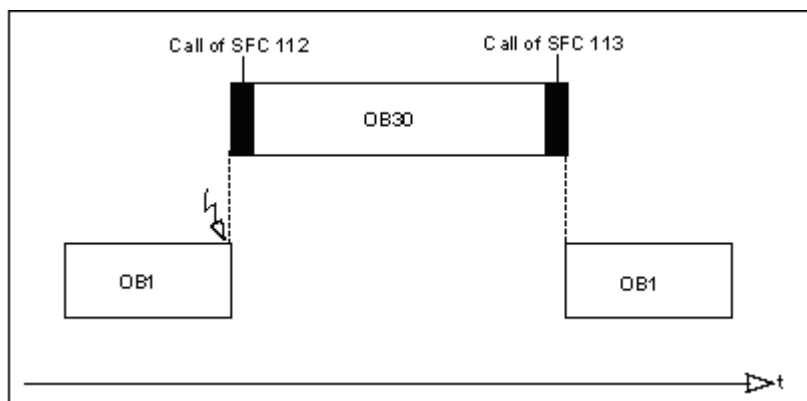
- 接口数据块的输入将在 PROFINET CBA 组件程序启动前被复制到影子存储器的输入
- 接口数据块的输出将在 PROFINET CBA 组件程序运行后被复制到影子存储器的输出

通过操作系统或 SFC 112 和 113，根据 SIMATIC Manager 中的 PROFINET CBA 组件的组态方式，对带有集成 PROFINET 接口的 CPU 上的 PROFINET CBA 组件进行更新。下图所示为通过 SFC 112 和 113 完成更新的方式。



系统范围更新始终发生在扫描周期检查点。

如果通过 SFC 112 和 113 执行更新，则在包含 PROFINET CBA 组件关联程序的 OB 的开始处调用 SFC 112，在此 OB 的结束处调用 SFC 113。下图以 OB 30 为实例说明这一过程。





**注意多功能组件(在一个设备上有一个以上的 PROFINET 接口数据块)：**

对于多功能组件，可以通过一个 SFC 作业(DBNO=0)更新所有接口数据块 - 类似于循环检验点上的更新，但由 SFC 触发。如果在更新接口数据块时出现错误，则继续更新其它接口数据块。一个负 RET\_VAL 关联一个更新接口数据块。

---

**更新 DP 互连**

根据在 SIMATIC Manager 中对 PROFINET CBA 组件的组态方式，确定是由操作系统还是通过 SFC 114 来更新 DP 互连。

## 17.2 使用 SFC 112 "PN\_IN"更新用于 PROFINET CBA 组件的用户程序接口的输入

### 描述

使用 SFC 112 "PN\_IN"可以将 PROFINET CBA 中的输入数据从 PROFINET CBA 组件的影子存储器复制到关联的接口数据块。完成 SFC 后，应用程序即可以使用当前这些输入数据。

参数	声明	数据类型	存储器区	描述
DBNO	INPUT	WORD	I、Q、M、D、L、 常数	接口数据块 DB 号(DBNO=0：更新所有 PROFINET CBA 接口数据块)
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8001	PROFINET CBA 组态不可用或无效。
8002	DB 编号与组件组态中的编号不符。
8004	DB 编号与组件组态中的编号相符，但尚未装载该 DB。
8005	接口 DB 编译了关键字 UNLINKED，也就是说，接口 DB 位于装载存储区而非工作存储区。
8006	CPU 中的接口数据块有写保护。
80B1	读取或写入时发生长度错误。组件组态不适合载入的 DB。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 17.3 使用 SFC 113 "PN\_OUT"更新 PROFINET CBA 组件的 PROFINET 接口的输出

### 描述

使用 SFC 113 "PN\_OUT"可以将应用程序中生成的输出数据从 PROFINET CBA 组件的接口数据块复制到关联的影子存储器。完成 SFC 后，其它 PROFINET CBA 组件可以使用当前这些输入数据。

参数	声明	数据类型	存储器区	描述
DBNO	INPUT	WORD	I、Q、M、D、L、 常数	接口数据块 DB 号(DBNO=0：更新所有 PROFINET 接口数据块)
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8001	PROFINET CBA 组态不可用或无效。
8002	DB 编号与组件组态中的编号不符。
8004	DB 编号与组件组态中的编号相符，但尚未装载该 DB。
8005	接口 DB 编译了关键字 UNLINKED，也就是说，接口 DB 位于装载存储区而非工作存储区。
8006	CPU 中的接口数据块有写保护。
80B1	读取或写入时发生长度错误。组件组态不适合载入的 DB。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 17.4 使用 SFC 114"PN\_DP"更新 DP 互连

### 描述

使用 SFC 114 "PN\_DP"可以更新下列所有互连：

- 本地 PROFIBUS 上 PROFINET CBA 组件间的互连
- 本地 PROFIBUS 上 PROFINET CBA 组件与外部 PROFINET CBA 组件间的周期性传输互连。这种互连应用于网络之间(Industrial Ethernet 与 PROFIBUS DP 间)。

### 功能

SFC 114 "PN\_DP"以异步方式工作。这意味着其处理过程覆盖若干个 SFC 调用。要开始更新 DP 互连，请使用 REQ=1 调用 SFC 114。

输出参数 RET\_VAL 和 BUSY 中指示了作业状态，另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、 常数	REQ=1：启动 DP 互连更新
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行功能时发生错误，返回值中将包含相应的错误代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：DP 互连更新尚未完成。

### 错误信息

错误代码 (W#16#...)	解释
0000	作业已无错执行
7000	初次调用时 REQ = 0：未启动 DP 互连更新。BUSY 的值为 0。
7001	(仅针对 S7-400：) REQ=1 的第一次调用。BUSY 的值为 1。
7002	(仅针对 S7-400：) 中间调用(REQ 不相关)。DP 互连更新尚未完成。BUSY 的值为 1。
8001	PROFINET CBA 组态不可用或无效。
8095	在较高优先级启动了另一个 DP 互连更新。不过，较低优先级更新(由操作系统或 SFC 114 执行)仍在进行。
8xyy	一般错误信息，请参见使用输出参数 RET_VAL 判断故障

# 18 用于 PROFINET CPU 的 SFC 和 SFB

## 18.1 使用 SFC99 "WWW"启用或同步用户 Web 页面

### 引言

使用组态工具 S7-Web2PLC ,可将您自己的 Web 页面集成到 STEP 7 程序和 S7 PROFINET CPU 中。CPU 中的每个 Web 应用程序都具有以下可通过 S7-Web2PLC 创建的 DB :

- 一个 Web 控制 DB
- 多片段 DB

Web 控制 DB 包含 Web 应用程序的所有结构信息。片段 DB 包含编码的用户 Web 页面。使用 SFC 99 , 可启用或同步 CPU 上的 Web 应用程序。

页面内容自动或由用户程序协调访问 CPU 数据。可在 S7-Web2PLC 组态工具的在线帮助中找到该工作方式的精确描述(见下文)。

### 描述

- 在 CPU 启动后 ,Web 服务器不能识别任何用户自定义 Web 页面。因而需要调用 SFC99 "WWW" , 这样 Web 页面才能为 Web 服务器所知。例如可在 OB 100 中进行此调用。

其它 SFC99 调用取决于 Web 页面的更新方式 :

- 自动更新的 Web 页面 :  
调用 SFC99 "WWW" , 直到在 Web 控制 DB 的"commandstate"结构中设置了变量"init"或变量"error" (当发生错误时) 为止。当发生错误时 , 错误原因显示在变量"last\_error"中 ,
- 用户程序控制的 Web 页面更新 :  
SFC99 "WWW"协调用户程序与 Web 页面。对于每个 Web 页面 , 基本程序如下 :
- 调用 SFC 99。
- 评估 Web 控制 DB 中的"requesttab"结构。
- 处理用户程序中的手动请求。
- 通过再次调用 SFC 99 对请求进行回应。

参数	声明	数据类型	存储区	描述
CTRL_DB	INPUT	BLOCK_DB	D	Web 控制 DB
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息

## 错误信息

错误代码(W#16#...)	解释
0000	未产生故障。
00xy	x = 1 : DB Web 控制初始化失败。发生的错误将在"last_error" (位于 Web 控制 DB 中)中详细描述。其中一个此类错误为"片段 DB 与 DB Web 控制不匹配"。 y 指示在用户程序中 4 个通道的哪一个应用于接收作业。
803A	指定的 Web 控制 DB 在 CPU 中不存在。
8081	版本错误或 Web 控制 DB 格式错误
80C1	没有可用于初始化 Web 应用程序的资源。此处所说的"资源"是同时使用的用户程序(不同的 Web 控制 DB)的最大数目。

## 更多信息

可在 S7-Web2PLC 组态工具的在线帮助中找到 Web 控制 DB 的 SFC99 的详细描述。它位于 STEP 7 DVD 光盘上的 DVD\_Root\Optional Components\S7 Web2PLC 目录下的文件"Web2PLCApx.chm"中(x=a 表示德语, x=b 表示英语)。

## 18.2 使用 SFB104 "IP\_CONF"设置 IP 组态

### 描述

SFB104 "IP\_CONF"用于 CPU 集成 PROFINET 接口的程序控制的组态。以前有效的组态数据将被覆盖。

可进行以下接口组态设置：

- IP 参数：IP 地址、子网掩码、路由器地址
- PROFINET IO 设备名称(如果 CPU 作为 PROFINET IO 设备操作)

需要将组态数据保存在组态 DB 中。

可使用 SFB104 选择程序控制的 IP 组态设置，作为使用 STEP 7 组态的备用。然而，它仅当在 HW Config 中明确指定 IP 参数将"使用其它方式" (而非通过 HW Config)分配时才有效。

### 块如何工作

SFB104 "IP\_CONF"是一个同步工作的 SFB，也就是说，处理过程跨越多个 SFB 调用。可使用 REQ = 1 调用 SFC104 启动传送。

每次只能激活一个作业。

作业状态通过输出参数 BUSY 和 STATUS 来显示。STATUS 对应于异步运行的 SFC 的 RET\_VAL 输出参数(参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

下表显示了 BUSY、DONE 和 ERROR 之间的关系。基于这些值，可在组态数据传送完成后查看 SFB104 的当前状态。

BUSY	DONE	ERROR	描述
TRUE	不相关	不相关	作业正在处理。
FALSE	TRUE	FALSE	该作业成功完成。
FALSE	FALSE	TRUE	该作业以出错而结束。出错原因可在 STATUS 参数中找到。
FALSE	FALSE	FALSE	无(新)作业发送到 SFB。

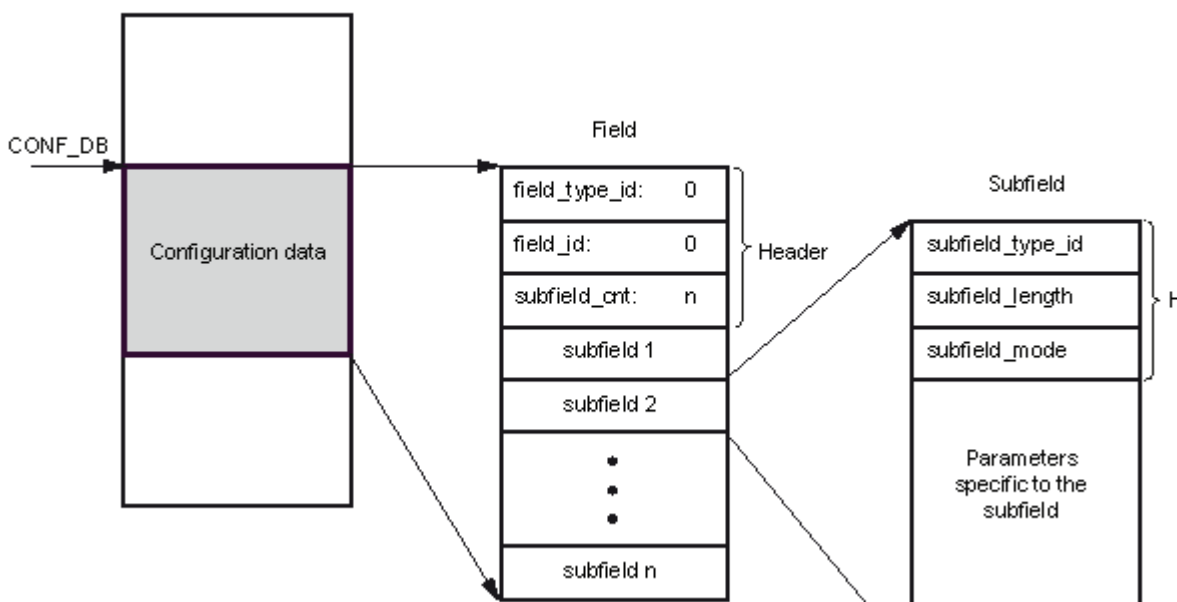
### 参数

参数	声明	数据类型	存储区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数 REQUEST 在上升沿开始传送。
LADDR	INPUT	WORD	M、D、常数	PROFINET 接口的诊断地址
CONF_DB	INPUT	ANY	D	指向组态数据的指针(允许的数据类型：BYTE、WORD、Block_DB) 注意：对于 S7-300 CPU，请注意，参数 CONF_DB 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。
DONE	OUTPUT	BOOL	I、Q、M、D、L	状态参数 DONE： 0: 作业尚未开始或仍在执行 1: 无错执行作业。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	状态参数 BUSY： 0: 作业已完成。 1: 作业仍在处理。
ERROR	OUTPUT	BOOL	I、Q、M、D、L	状态参数 ERROR： ERROR = 1：处理作业期间出现错误。STATUS 返回有关错误类型的详细信息。
STATUS	OUTPUT	DWORD	I、Q、M、D、L	状态参数 STATUS：错误信息
ERR_LOC	OUTPUT	DWORD	I、Q、M、D、L	错误源(导致错误的参数块的 field_id 和 subfield_type_id)



## 组态 DB

下图说明了要传送的组态数据是如何保存在组态 DB 中的。



因而，组态数据包含一个域和多个子域：

- 域由一个报头和多个子域组成。而报头包括以下成分
  - field\_type\_id (数据类型 INT)：零
  - field\_id (数据类型 INT)：零
  - subfield\_cnt (数据类型 INT)：子域数
- 每个子域则包含一个报头(subfield\_type\_id、subfield\_length、subfield\_mode)和专属于子域的参数。必须以偶数字节填充每个子域。

subfield\_mode 允许以下值：

- 1: 组态数据的永久有效性
- 2: 组态数据的临时有效性，包括删除现有的永久组态数据

### 注意

当前仅允许一个域。其参数"field\_type\_id"和"field\_id"的值必须为零。带有其它值的更多域将为 field\_type\_id 和 field\_id 保留以用于将来扩展。

### 允许的子域

子域类型 id	子域名称	解释
30	SUB_IP_SUITE_IPV4	IP 参数：IP 地址、子网掩码、路由器地址
40	SUB_NOS	PROFINET IO 设备名称(站名称)

### 允许的子域的类型定义

- "SUB\_IP\_SUITE\_IPV4"子域

TYPE "SUB_IP_SUITE_V4"	
STRUCT	
id: int := 30;	// 子域类型 id
len: int := 18;	// subfield_length
mode: int := 1;	// subfield_mode (1 : 永久 , // 2 : 临时)
ipaddr_3: BYTE := b#16#c8;	// IP 地址高字节 : 200
ipaddr_2: BYTE := b#16#0c;	// IP 地址高字节 : 12
ipaddr_1: BYTE := b#16#01;	// IP 地址低字节 : 1
ipaddr_0: BYTE := b#16#90;	// IP 地址低字节 : 144
snmask_3: BYTE := b#16#FF;	//子网掩码高字节 : 255
snmask_2: BYTE := b#16#FF;	//子网掩码高字节 : 255
snmask_1: BYTE := b#16#FF;	//子网掩码低字节 : 255
snmask_0: BYTE := b#16#00;	//子网掩码低字节 : 0
router_3: BYTE := b#16#c8;	//路由器高字节 : 200
router_2: BYTE := b#16#0c;	//路由器高字节 : 12
router_1: BYTE := b#16#01;	//路由器低字节 : 1
router_0: BYTE := b#16#01;	//路由器低字节 : 1
END_STRUCT	
END_TYPE	

- "SUB\_NOS"子域

TYPE "SUB_NOS"	
STRUCT	
id: int := 40;	// 子域类型 id
len: int := 246;	// subfield_length
mode: int := 1;	// subfield_mode (1 : 永久 , // 2 : 临时)
nos: BYTE 的 ARRAY[1..240] := 0;	设备名称 : ARRAY 必须从第一个字节开始复制。如果 ARRAY 长于要分配的设备名称, 必须在实际设备名称(符合 IEC 61158-6-10)后输入一个空(null)字节。否则, 将拒绝 nos, SFB 104 将在 STATUS 中接收到错误代码 DW#16#C0809400。如果分配 NULL 作为第一个字节, 设备名称将被删除。
END_STRUCT	
END_TYPE	

设备名称有以下限制 :

- 总长度限定为 240 个字符(小写字母、数字、短划线或句点)
- 设备名称中的部分名称, 即, 介于两个句点之间的字符串, 最多不得超过 63 个字符。部分名称的开头或结尾不允许使用"."字符。
- 不得使用特殊字符, 例如元音变音(ä、ö 等)、括号、下划线、斜线、空格等。短划线是唯一允许使用的特殊字符。
- 设备名称不得以 "-" 或 "." 字符开头或结束。
- 设备名称不得以数字开始。
- 设备名称结构不得为 n.n.n.n (n = 0...999)。
- 设备名称不得以字符串 "port-xyz" 或 "port-xyz-abcde" (a、b、c、d、e、x、y、z = 0, ... 9) 开始。
- 设备名称不允许使用字符串 ".."。

### 注意

也可以创建少于 240 个字符的 "nos" ARRAY, 但长度最小不得少于 2 个字节。在这种情况下, 需要相应调整 "len" 变量(子域长度)。

## 错误信息

ERROR	STATUS (DW#16#...)	ERR_LOC	解释
0	0x00000000	0	作业处理无错完成
0	0x00700000	0	没有正在处理的作业
0	0x00700100	0	启动作业处理
0	0x00700200	0	中间调用(REQ 不相关)
1	0xC08xyy00	0	关于常规错误信息，请参考"使用输出参数 RET_VAL 判断错误"
1	0xC0808000	0	LADDR 无效
1	0xC0808100	0	LADDR 未分配给所支持的 PROFINET 接口
1	0xC0808200	0	参数 CONF_DB 中的错误：不支持数据类型
1	0xC0808300	0	参数 CONF_DB 中的错误：指针指向不受支持的区域
1	0xC0808400	0	参数 CONF_DB 中的错误：错误的 ANY 指针长度
1	0xC0808700	0	参数 CONF_DB 中的错误：配置数据中的长度规定不一致
1	0xC0808800	f、0	field_type_id 的值非法
1	0xC0808900	f、0	field_id 的值非法
1	0xC0808A00	f、0	subfield_cnt 的数值错误
1	0xC0808B00	f、t	subfield_id 的值非法
1	0xC0808C00	f、t	子域使用超过一次
1	0xC0808D00	f、t	subfield_len 的值不正确或非法
1	0xC0808E00	f、t	subfield_mode 的值非法
1	0xC0808F00	f、t	子域包含与早期子域的冲突。
1	0xC0809000	f、t	子域的参数为写保护。例如，参数分配在组态中进行，或启用了 PNIO 模式。
1	0xC0809400	f、t	子域中的参数值未定义或无效
1	0xC080C200	0	无法进行传送(例如，由于接口不可用)
1	0xC080C300	0	资源不足(例如，使用不同参数多次调用 SFB104)
1	0xC080C400	0	临时通信错误

在上表中，f 和 t 分别是导致错误的参数块的 field\_id 和 subfield\_type\_id。

## SIMATIC

### System Software for S7-300/400 System and Standard Functions - Volume 2/2

参考手册

用于根据 PNO 循环访问用户数据的 FB	19
用于全局数据通讯的 SFC	20
S7 通讯和 S7 基本通讯的概述	21
S7 通讯	22
用于未组态的 S7 连接的通讯 SFC	23
通过 Industrial Ethernet 打开通讯	24
产生与块有关的消息	25
IEC 定时器和 IEC 计数器	26
IEC 功能	27
用于集成控制的 SFB	28
用于紧凑型 CPU 的 SFB	29
用于 H CPU 的 SFC	30
集成功能(对于带集成 I/O 的 CPU)	31
塑料技术	32
诊断数据	33
系统状态列表(SSL)	34
事件	35
SFC 和 SFB 列表	36

## 法律资讯

### 警告提示系统

为了您的人身安全以及避免财产损失，必须注意本手册中的提示。人身安全的提示用一个警告三角表示，仅与财产损失有关的提示不带警告三角。警告提示根据危险等级由高到低如下表示。

#### 危险

表示如果不采取适当的预防措施，可能导致死亡或严重的人身伤害。

#### 警告

表示如果不采取适当的预防措施，可能导致死亡或严重的人身伤害。

#### 注意

表示如果不采取适当的预防措施，可能导致轻微的人身伤害。

#### 注意

表示如果不采取适当的预防措施，可能造成财产损失。

当出现多个危险等级的情况下，每次总是使用最高等级的警告提示。如果在某个警告提示中带有警告可能导致人身伤害的警告三角，则可能在该警告提示中另外还附带有可能导致财产损失的警告。

### 合格的专业人员

仅允许安装和驱动与本文件相关的附属设备或系统。设备或系统的调试和运行仅允许由**合格的专业人员**进行。本文件安全技术提示中的合格专业人员是指根据安全技术标准具有从事进行设备、系统和电路的运行，接地和标识资格的人员。

### 按规定使用 Siemens 产品

注意下列各项：

#### 警告

Siemens 产品只允许用于目录和相关技术文件中规定的使用情况。如果要使用其他公司的产品和组件，必须得到 Siemens 推荐和允许。正确的运输、储存、组装、装配、安装、调试、操作和维护是产品安全、正常运行的前提。必须遵守允许的环境条件。必须注意相关文件中的提示。

### 商标

所有带有标记符号 © 的都 Siemens AG 的注册商标。标签中的其他符号可能是一些其他商标，这是出于保护所有者的权利的目地由第三方使用而特别标示的。

### 免责声明

我们已检查过本手册中的内容与所描述的硬件和软件相符。由于差错在所难免，我们不能保证完全一致。我们会定期审查本手册中的信息，并在后续版本中进行必要的更正。

# 目录

<b>19</b>	<b>用于根据 PNO 循环访问用户数据的 FB</b> .....	<b>315</b>
19.1	用于根据 PNO 循环访问用户数据的 FB 的介绍 .....	315
19.2	使用 FB 20 "GETIO"读取 DP 标准从站/PROFINET IO 设备的所有输入 .....	316
19.3	使用 FB 21 "SETIO"写入 DP 标准从站/PROFINET IO 设备的所有输出 .....	317
19.4	使用 FB 22 "GETIO_PART"读取 DP 标准从站/PROFINET IO 设备的一部分输入 .....	318
19.5	使用 FB 23 "SETIO_PART"写入 DP 标准从站/PROFINET IO 设备的一部分输出 .....	320
<b>20</b>	<b>用于全局数据通讯的 SFC</b> .....	<b>323</b>
20.1	通过 SFC 60"GD_SND"发送 GD 信息包 .....	323
20.2	通过 SFC 61 "GD_RCV"编程接受已接收到的 GD 信息包 .....	326
<b>21</b>	<b>S7 通讯和 S7 基本通讯的概述</b> .....	<b>329</b>
21.1	S7 通讯块和 S7 基本通讯块之间的差别 .....	329
21.2	数据一致性 .....	331
21.3	S7 通讯块概述 .....	332
21.4	S7 通讯块概述 .....	334
<b>22</b>	<b>S7 通讯</b> .....	<b>337</b>
22.1	用于 S7 通讯的 SFB/FB 和 SFC/FC 的公用参数 .....	337
22.2	用于组态的 S7 连接的 SFB 启动例行程序 .....	341
22.3	SFB 对故障如何反应 .....	343
22.4	通过 SFB 8/FB 8"USEND"进行无协调的数据发送 .....	345
22.5	通过 SFB/FB 9 "URCV"进行无协调的数据接收 .....	349
22.6	通过 SFB/FB 12 "BSEND"发送分段数据 .....	352
22.7	通过 SFB/FB 13 "BRCV"接收分段数据 .....	356
22.8	通过 SFB/FB 15 "PUT"向远程 CPU 写入数据 .....	359
22.9	通过 SFB/FB 14 "GET"从远程 CPU 中读取数据 .....	363
22.10	通过 SFB 16 "PRINT"将数据发送到打印机 .....	366
22.11	通过 SFB 19 "START"在远程设备上启动暖启动或冷启动 .....	372
22.12	通过 SFB 20 "STOP"将远程设备切换到 STOP 状态 .....	375
22.13	通过 SFB 21 "RESUME"在远程设备上启动热启动 .....	377
22.14	通过 SFB 22"STATUS"查询远程伙伴的状态 .....	379
22.15	通过 SFB 23 "USTATUS"接收远程设备的状态改变 .....	381
22.16	通过 SFC 62"CONTROL"查询属于 SFB 实例的连接状态 .....	383
<b>23</b>	<b>用于未组态的 S7 连接的通讯 SFC</b> .....	<b>385</b>
23.1	通讯 SFC 的公用参数 .....	385
23.2	用于未组态 S7 连接的通讯 SFC 的出错信息 .....	387
23.3	通过 SFC 65"X_SEND"发送数据到本地 S7 站以外的通讯伙伴 .....	390
23.4	通过 SFC 66 "X_RCV"从本地 S7 站以外的通讯伙伴中接收数据 .....	391
23.5	通过 SFC 68 "X_PUT"将数据写入本地 S7 站以外的通讯伙伴 .....	395
23.6	通过 SFC 67 "X_GET"从本地 S7 站以外的通讯伙伴中读取数据 .....	397
23.7	通过 SFC 69"X_ABORT"中止已存在的、到本地 S7 站以外的通讯伙伴的连接 .....	399

23.8	通过 SFC 72 "I_GET"从本地 S7 站内的通讯伙伴中读取数据 .....	400
23.9	通过 SFC 73 "I_PUT"将数据写入本地 S7 站内的通讯伙伴 .....	402
23.10	通过 SFC 74 "I_ABORT"中止已存在的、到本地 S7 站内的通讯伙伴的连接 .....	404
<b>24</b>	<b>通过 Industrial Ethernet 打开通讯 .....</b>	<b>405</b>
24.1	概述 .....	405
24.2	用于工业以太网上开放通讯的 FB 的功能 .....	406
24.3	为 TCP 和 ISO-on-TCP 通信连接分配参数 .....	410
24.4	为使用 UDP 的本地通讯接入点分配参数 .....	415
24.5	使用 UDP 的远程通讯伙伴地址信息的结构 .....	417
24.6	CPU 和所用的协议变量(connection_type)以及可传送数据长度之间的关系 .....	418
24.7	使用 FB 65 "TCON"建立连接 .....	419
24.8	使用 FB 66 "TDISCON"终止连接 .....	422
24.9	使用 FB 63 "TSEND"通过 TCP 和 ISO on TCP 发送数据 .....	424
24.10	使用 FB 64 "TRCV"通过 TCP 和 ISO on TCP 接收数据 .....	427
24.11	使用 FB 67 "TUSEND"通过 UDP 发送数据 .....	431
24.12	使用 FB 68 "TURCV"通过 UDP 接收数据 .....	434
24.13	通过 FB 210 "FW_TCP"使用通过 TCP 的 FETCH 和 WRITE 服务链接第三方系统 .....	437
24.14	通过 FB 220 "FW_IOT"使用通过 ISO-on-TCP 的 FETCH 和 WRITE 服务链接第三方系统 .....	438
<b>25</b>	<b>产生与块有关的消息 .....</b>	<b>439</b>
25.1	关于使用 SFB 生成块相关消息的介绍 .....	439
25.2	使用 SFB 36 "NOTIFY"生成无需确认的块相关消息 .....	443
25.3	使用 SFB 31 "NOTIFY_8P"生成无确认显示的块相关消息 .....	445
25.4	使用 SFB 33 "ALARM"生成需要确认的块相关消息 .....	447
25.5	使用 SFB 35 "ALARM_8P"生成针对八个信号的带有关联值的块相关消息 .....	450
25.6	使用 SFB 34 "ALARM_8"生成针对八个信号的不附带关联值的块相关消息 .....	453
25.7	使用 SFB 37 "AR_SEND"发送归档数据 .....	455
25.8	使用 SFC 10 "DIS_MSG"禁用块相关、符号相关和组状态消息 .....	457
25.9	使用 SFC 9 "EN_MSG"启用块相关、符号相关和组状态消息 .....	459
25.10	用于生成块相关消息的 SFB 的启动特性 .....	461
25.11	用于生成块相关消息的 SFB 如何对问题做出反应 .....	462
25.12	关于使用 SFC 生成块相关消息的介绍 .....	463
25.13	使用 SFC 17 "ALARM_SQ"生成可确认的块相关消息及使用 SFC 18 "ALARM_S"生成永久确认的块相关消息 .....	466
25.14	使用 SFC 19 "ALARM_SC"查询上一 ALARM_SQ/ALARM_DQ 进入事件消息的确认状态 .....	469
25.15	使用 SFC 107 "ALARM_DQ"和 108 "ALARM_D"生成可确认和永久确认的块相关消息 .....	470
25.16	使用 SFC 105 "READ_SI"读取动态系统资源 .....	472
25.17	使用 SFC 106 "DEL_SI"读取动态系统资源 .....	475
<b>26</b>	<b>IEC 定时器和 IEC 计数器 .....</b>	<b>477</b>
26.1	使用 SFB 3 "TP"生成脉冲 .....	477
26.2	使用 SFB 4 "TON"生成接通延迟 .....	479
26.3	使用 SFB 5 "TOF"生成断开延迟 .....	480
26.4	使用 SFB 0 "CTU"递增计数 .....	481
26.5	使用 SFB 1 "CTD"递减计数 .....	482
26.6	使用 SFB 2 "CTUD"递增/递减计数 .....	483
<b>27</b>	<b>IEC 功能 .....</b>	<b>485</b>
27.1	概述 .....	485
27.2	IEC 功能的技术数据 .....	487
27.3	日期和时间作为复杂数据类型 .....	489
27.4	Time-of-Day 功能 .....	490
27.5	比较 DATE_AND_TIME 变量 .....	493



27.6	编辑 STRING 变量 .....	495
27.7	比较 STRING 变量 .....	499
27.8	转换数据类型格式 .....	502
27.9	编辑数值 .....	504
<b>28</b>	<b>用于集成控制的 SFB .....</b>	<b>507</b>
28.1	使用 SFB 41/FB 41 "CONT_C"进行连续控制 .....	507
28.2	使用 SFB 42/FB 42 "CONT_S"进行步控制 .....	514
28.3	使用 SFB 43/FB 43 "PULSEGEN"生成脉冲 .....	520
28.4	PULSEGEN 块的实例 .....	532
<b>29</b>	<b>用于紧凑型 CPU 的 SFB .....</b>	<b>535</b>
29.1	通过 SFB 44 "Analog"使用模拟量输出进行定位 .....	535
29.2	通过 SFB 46 "DIGITAL"使用数字量输出进行定位 .....	546
29.3	使用 SFB 47 "COUNT"控制计数器 .....	557
29.4	使用 SFB 48 "FREQUENC"控制频率测量 .....	562
29.5	使用 SFB 49 "PULSE"控制脉宽调制 .....	566
29.6	使用 SFB 60 "SEND_PTP"发送数据(ASCII , 3964(R)) .....	569
29.7	使用 SFB 61 "RCV_PTP"接收数据(ASCII , 3964(R)) .....	572
29.8	使用 SFB 62 "RES_RCVB"删除接收缓冲区(ASCII , 3964(R)) .....	575
29.9	使用 SFB 63 "SEND_RK"发送数据(512(R)) .....	577
29.10	使用 SFB 64 "FETCH_RK"获取数据(RK 512) .....	581
29.11	使用 SFB 65 "SERVE_RK"接收和提供数据(RK 512) .....	586
29.12	SFB 60 至 65 的补充错误信息 .....	590
<b>30</b>	<b>用于 H CPU 的 SFC .....</b>	<b>597</b>
30.1	在 H 系统中使用 SFC 90 "H_CTRL"控制操作 .....	597
<b>31</b>	<b>集成功能(对于带集成 I/O 的 CPU) .....</b>	<b>601</b>
31.1	SFB 29 (HS_COUNT) .....	601
31.2	SFB 30 (FREQ_MES) .....	603
31.3	SFB 38 (HSC_A_B) .....	604
31.4	SFB 39 (POS) .....	605
<b>32</b>	<b>塑料技术 .....</b>	<b>607</b>
32.1	SFC 63 (AB_CALL) .....	607
<b>33</b>	<b>诊断数据 .....</b>	<b>609</b>
33.1	诊断数据结构概述 .....	609
33.2	诊断数据 .....	610
33.3	通道专有诊断数据的结构 .....	612
<b>34</b>	<b>系统状态列表(SSL) .....</b>	<b>615</b>
34.1	系统状态列表(SSL)概述 .....	615
34.2	部分 SSL 列表的结构 .....	616
34.3	SSL-ID .....	617
34.4	可能的部分系统状态列表 .....	618
34.5	SSL-ID W#16#xy11 - 模块标识 .....	619
34.6	SSL-ID W#16#xy12 - CPU 特征 .....	620
34.7	SSL-ID W#16#xy13 - 存储器区 .....	623
34.8	SSL-ID W#16#xy14 - 系统区域 .....	624
34.9	SSL-ID W#16#xy15 - 块类型 .....	626

34.10	SSL-ID W#16#xy74 - 模块 LED 的状态 .....	627
34.11	SSL-ID W#16#xy1C - 组件标识 .....	629
34.12	SSL-ID W#16#xy22 - 中断状态 .....	635
34.13	SSL ID W#16#xy25 - 将过程映像分区分配到 OB .....	637
34.14	SSL-ID W#16#xy32 - 通讯状态数据 .....	640
34.15	SSL-ID 为 W#16#0132、索引为 W#16#0005 的部分列表摘录的数据记录 .....	641
34.16	SSL-ID 为 W#16#0132、索引为 W#16#0008 的部分列表摘录的数据记录 .....	642
34.17	SSL-ID 为 W#16#0132、索引为 W#16#000B 的部分列表摘录的数据记录 .....	644
34.18	具有 SSL ID W#16#0132 索引 W#16#000C 的部分列表摘录的数据记录 .....	645
34.19	SSL-ID 为 W#16#0232、索引为 W#16#0004 的部分列表摘录的数据记录 .....	646
34.20	SSL-ID W#16#xy71 - H CPU 组信息 .....	648
34.21	SSL-ID W#16#xy75 - H 系统中切换的 DP 从站/IO 设备 .....	651
34.22	SSL-ID W#16#xy90 - DP 主站系统信息 .....	653
34.23	SSL-ID W#16#xy91 - 模块状态信息 .....	655
34.24	SSL-ID W#16#xy92 - 机架/站状态信息 .....	661
34.25	SSL-ID W#16#xy95 - 扩展的 DP 主站系统/PROFINET IO 系统信息 .....	665
34.26	SSL-ID W#16#xyA0 - 诊断缓冲区 .....	667
34.27	SSL-ID W#16#00B1 - 模块诊断信息 .....	668
34.28	SSL-ID W#16#00B2 - 对应物理地址的诊断数据记录 1 .....	670
34.29	SSL-ID W#16#00B3 - 对应逻辑基址的模块诊断数据 .....	671
34.30	SSL-ID W#16#00B4 - DP 从站的诊断数据 .....	672
<b>35</b>	<b>事件 .....</b>	<b>673</b>
35.1	事件和事件 ID .....	673
35.2	事件等级 1 - 标准 OB 事件 .....	674
35.3	事件等级 2 - 同步错误 .....	675
35.4	事件等级 3 - 异步错误 .....	676
35.5	事件等级 4 - 停止事件和其它模式更改 .....	679
35.6	事件等级 5 - 模式运行期事件 .....	682
35.7	事件等级 6 - 通讯事件 .....	683
35.8	事件等级 7 - H/F 事件 .....	684
35.9	事件等级 8 - 模块的诊断事件 .....	687
35.10	事件等级 9 - 标准用户事件 .....	689
35.11	事件等级 A 和 B - 自由用户事件 .....	691
35.12	保留的事件等级 .....	691
<b>36</b>	<b>SFC 和 SFB 列表 .....</b>	<b>693</b>
36.1	按编号排序的 SFC 列表 .....	693
36.2	按字母排序的 SFC 列表 .....	696
36.3	按数字排序的 SFB 列表 .....	699
36.4	按字母排序的 SFB 列表 .....	701
	<b>参考书目 .....</b>	<b>703</b>
	<b>词汇表 .....</b>	<b>705</b>
	<b>索引 .....</b>	<b>719</b>

# 19 用于根据 PNO 循环访问用户数据的 FB

## 19.1 用于根据 PNO 循环访问用户数据的 FB 的介绍

### 概述

根据 PNO (PROFIBUS Nutzerorganisation e.V./德国 PROFIBUS 用户组织) 用于循环访问用户数据的 FB 是下列块(括号中提供了含义)：

- FB 20 "GETIO" (读取 DP 标准从站/PROFINET IO 设备的所有输入)
- FB 21 "SETIO" (写入 DP 标准从站/PROFINET IO 设备的所有输出)
- FB 22 "GETIO\_PART" (读取 DP 标准从站/PROFINET IO 设备的一部分输入)
- FB 23 "SETIO\_PART" (写入 DP 标准从站/PROFINET IO 设备的一部分输出)

### 用于根据 PNO 来循环访问用户数据的 FB 的接口

上述命名的 FB 接口与那些在标准"PROFIBUS DP 和 PROFINET IO 上的 PROFIBUS 和 PROFINET 通信功能块"中定义的具有相同名称的 FB 完全相同。

## 19.2 使用 FB 20 "GETIO"读取 DP 标准从站/PROFINET IO 设备的所有输入

### 描述

使用 FB 20 "GETIO"一致读取 DP 标准从站/PROFINET IO 设备的所有输入。为此，FB 20 调用 SFC 14 "DPRD\_DAT"。如果在数据发送期间没有错误，则读取的数据将在由 INPUTS 指示的目标区域中输入。

目标区域的长度必须与用 STEP 7 为选定组件所组态的长度相同。

对于具有模块化结构或带有多个 DP 标识的 DP 标准从站，每次只能使用 FB 20 调用在组态的起始地址访问一个组件/DP 标识的数据。

参数	声明	数据类型	存储器区	描述
ID	INPUT	DWORD	I、Q、M、D、L 常数	<ul style="list-style-type: none"> <li>低位字：DP 从站/PROFINET IO 组件 (模块或子模块)的逻辑地址</li> <li>高位字：不相关</li> </ul>
STATUS	OUTPUT	DWORD	I、Q、M、D、L	包含 SFC 14 "DPRD_DAT"的错误信息，格式为 DW#16#40xxxx00
LEN	OUTPUT	INT	I、Q、M、D、L	按字节读取的数据量
INPUTS	IN_OUT	ANY	I、Q、M、D	用于读取数据的目标区域。必须与用 STEP 7 为选定的 DP 从站/ PROFINET IO 组件组态的长度相同。仅允许使用数据类型 BYTE。

### 错误信息

请参见 用 SFC 14 "DPRD\_DAT"读取 DP 标准从站//PROFINET IO 设备的连续数据

## 19.3 使用 FB 21 "SETIO"写入 DP 标准从站/PROFINET IO 设备的所有输出

### 描述

使用 FB 21 "SETIO", 可将数据从由 OUTPUTS 指示的源区域一致性地传送到所寻址的 DP 标准从站/PROFINET IO 设备, 如有必要, 还可传送到过程映像(如果已将用于 DP 标准从站的受影响的地址区域组态为过程映像中的一致性区域)。为此, FB 21 调用 SFC 15 "DPRD\_DAT"。

源区域的长度必须与用 STEP 7 为选定组件所组态的长度相同。

对于具有模块化结构或带有多个 DP 标识的 DP 标准从站, 每次只能使用 FB 20 调用在组态的起始地址访问一个组件/DP 标识的数据。

参数	声明	数据类型	存储器区	描述
ID	INPUT	DWORD	I、Q、M、D、L、 常数	<ul style="list-style-type: none"> <li>低位字：DP 从站/PROFINET IO 组件 (模块或子模块)的逻辑地址</li> <li>高位字：不相关</li> </ul>
LEN	INPUT	INT	I、Q、M、D、L	无关
STATUS	OUTPUT	DWORD	I、Q、M、D、L	包含 SFC 15 "DPRD_DAT"的错误信息, 格式为 DW#16#40xxxx00
OUTPUTS	IN_OUT	ANY	I、Q、M、D	要读取数据的源区域。必须与用 STEP 7 为选定的 DP 从站/ PROFINET IO 组件组态的长度相同。仅允许使用数据类型 BYTE。

### 错误信息

请参见用 SFC 15 "DPWR\_DAT"向 DP 标准从站/PROFINET IO 设备写入连续数据。

## 19.4 使用 FB 22 "GETIO\_PART"读取 DP 标准从站/PROFINET IO 设备的一部分输入

### 描述

使用 FB 22 "GETIO\_PART"，一致性地读取属于 DP 标准从站/PROFINET IO 设备的过程映像区域的一部分。为此，FB 22 调用 SFC 81 "UBLKMOV"。

#### 注意

必须为调用 FB 22 "GETIO\_PART"的 OB 分配过程映像分区，以进行输入。而且，在调用 FB 22 前，必须将相关的 DP 标准从站或相关的 PROFINET IO 设备添加到该过程映像分区，以进行输入。如果 CPU 不能识别任何过程映像分区或想要在 OB 1 中调用 FB 22，则必须在调用 FB 22 之前将相关的 DP 标准从站或相关的 PROFINET IO 设备添加到该过程映像分区，以进行输入。

使用 OFFSET 和 LEN 参数指定过程映像区域的一部分，该过程映像将通过其标识寻址的组件所读取。

如果在数据发送期间没有错误，则 ERROR 接收数值 FALSE，并且已读取的数据在通过 INPUTS 指示的目标区域中输入。

如果在数据发送期间出现一个错误，则 ERROR 接收数值 TRUE，STATUS 接收 SFC 81 错误信息 "UBLKMOV"。

如果目标区域(INPUTS 参数)小于 LEN，则将尽可能多的传送 INPUTS 可以接受的字节数。ERROR 接收数值 FALSE。如果目标区域大于 LEN，则目标区域中的第一个 LEN 字节将写入。ERROR 接收数值 FALSE。

#### 注意

FB 22 "GETIO\_PART"不会在属于不同的 PROFIBUS DP 或 PROFINET IO 组件的数据之间检查所输入的过程映像的分隔符。正因为如此，您必须自己确保通过 OFFSET 和 LEN 指定的过程映像区域属于同一个组件。将来的系统不能保证可以读取多个组件的数据并传送到来自其它制造商的系统。

参数	声明	数据类型	存储器区	描述
ID	INPUT	DWORD	I、Q、M、D、L 常数	<ul style="list-style-type: none"> <li>低位字：DP 从站/PROFINET IO 组件(模块或子模块)的逻辑地址</li> <li>高位字：不相关</li> </ul>
OFFSET	INPUT	INT	I、Q、M、D、L 常数	在过程映像中要为组件读取的第一个字节数(最小可能值：0)
LEN	INPUT	INT	I、Q、M、D、L 常数	要读取的字节数
STATUS	OUTPUT	DWORD	I、Q、M、D、L	包含 SFC 81 "UBLKMOV"的错误信息，格式为 DW#16#40xxxx00，如果 ERROR = TRUE
ERROR	OUTPUT	BOOL	I、Q、M、D、L	出错显示： ERROR = TRUE，如果在调用 SFC 81 "UBLKMOV"时出错。

参数	声明	数据类型	存储器区	描述
INPUTS	IN_OUT	ANY	I、Q、M、D	用于读取数据的目标区域： <ul style="list-style-type: none"> <li>• 如果目标区域小于 LEN，则将尽可能多的传送 INPUTS 可以接受的字节数。ERROR 接收数值 FALSE。</li> <li>• 如果目标区域大于 LEN，则目标区域中的第一个 LEN 字节将写入。ERROR 接收数值 FALSE。</li> </ul>

## 错误信息

请参见使用 SFC 81"UBLKMOV"不中断地复制变量。

## 19.5 使用 FB 23 "SETIO\_PART"写入 DP 标准从站/PROFINET IO 设备的一部分输出

### 描述

使用 FB 23 "SETIO\_PART"，可将数据从由 OUTPUTS 指示的源区域传送到属于 DP 标准从站/PROFINET IO 设备的过程映像区域的一部分。为此，FB 23 调用 SFC 81 "UBLKMOV"。

#### 注意

必须为调用 FB 23 "SETIO\_PART"的 OB 分配过程映像分区，以进行输出。而且，在调用 FB 23 前，必须将相关的 DP 标准从站或相关的 PROFINET IO 设备添加到该过程映像分区，以进行输出。如果 CPU 不能识别任何过程映像分区或想要在 OB 1 中调用 FB 23，则必须在调用 FB 23 之前将相关的 DP 标准从站或相关的 PROFINET IO 设备添加到该过程映像分区，以进行输出。

使用 OFFSET 和 LEN 参数指定要为通过其标识寻址的组件写入的过程映像区域的一部分。

如果在数据发送期间没有错误，则 ERROR 接收数值 FALSE。

如果在数据发送期间出现一个错误，则 ERROR 接收数值 TRUE，STATUS 接收 SFC 81 错误信息 "UBLKMOV"。

如果源区域(OUTPUTS 参数)小于 LEN，则将尽可能多的传送 OUTPUTS 所包含的字节数。ERROR 接收数值 FALSE。如果源区域大于 LEN，则第一个 LEN 字节将从 OUTPUTS 传送。ERROR 接收数值 FALSE。

#### 注意

FB 23 "SETIO\_PART"不会在属于不同的 PROFIBUS DP 或 PROFINET IO 组件的数据之间检查所输入的过程映像的分隔符。正因为如此，您必须自己确保通过 OFFSET 和 LEN 指定的过程映像区域属于同一个组件。将来的系统不能保证可以写入多个组件的数据并传送到来自其它制造商的系统。

参数	声明	数据类型	存储器区	描述
ID	INPUT	DWORD	I、Q、M、D、L、 常数	<ul style="list-style-type: none"> <li>低位字：DP 从站/PROFINET IO 组件(模块或子模块)的逻辑地址</li> <li>高位字：不相关</li> </ul>
OFFSET	INPUT	INT	I、Q、M、D、L、 常数	在过程映像中要为组件写入的第一个字节数(最小可能值：0)
LEN	INPUT	INT	I、Q、M、D、L、 常数	要写入的字节数
STATUS	OUTPUT	DWORD	I、Q、M、D、L	包含 SFC 81 "UBLKMOV"的错误信息，格式为 DW#16#40xxxx00，如果 ERROR = TRUE
ERROR	OUTPUT	BOOL	I、Q、M、D、L	出错显示： ERROR = TRUE，如果在调用 SFC 81 "UBLKMOV"时出错。



参数	声明	数据类型	存储器区	描述
OUTPUTS	IN_OUT	ANY	I、Q、M、D	要写入数据的源区域： <ul style="list-style-type: none"> <li>• 如果源区域小于 LEN，则将尽可能多的传送 OUTPUTS 所包含的字节数。ERROR 接收数值 FALSE。</li> <li>• 如果源区域大于 LEN，则第一个 LEN 字节将从 OUTPUTS 传送。ERROR 接收数值 FALSE。</li> </ul>

## 错误信息

请参见使用 SFC 81"UBLKMOV"不中断地复制变量。



## 20 用于全局数据通讯的 SFC

### 20.1 通过 SFC 60"GD\_SND"发送 GD 信息包

#### 描述

通过 SFC 60"GD\_SND"(全局数据发送)，采集 GD 信息包的数据，并通过在 GD 信息包中指定的路径发送。必须已经使用 STEP 7 对 GD 信息包进行组态。

可以在用户程序的任何地方调用 SFC 60"GD\_SND"。

扫描速率以及系统在循环检测点处所进行的数据采集和发送并不受 SFC 60 调用的影响。

#### 中断能力

SFC 60"GD\_SND"可以被具有更高优先级的程序中中断。还可以在更高的优先级中为同一个 GD 信息包再次调用 SFC 60。

然后，以更高的优先级收集、发送数据。当程序返回到被中断的 SFC 时，将立即终止执行此程序块，并丢弃已经接收到的数据。

此过程意味着，在具有更高优先级的处理过程中，传送的是一致性数据(从全局数据的意义上定义一致性)。

#### GD 的数据一致性

下列规则适用于从各种存储区域采集到的和发送的数据的一致性。

下列数据是一致的：

- 简单数据类型(位、字节、字和双字)
- 数据类型为字节、字和双字的数组，最大数据长度取决于特定的 CPU 类型。

### 确保整个 GD 信息包的一致性

CPU 上发送数据的 GD 信息包所具有的结构不会自动保证所采集到的数据是一致的。例如，当信息包由字节数组组成，而字节的数目超过了指定 CPU 所允许的最大长度时，就会发生这种情况。

然而，如果需要整个 GD 信息包具有一致性，请在程序中遵守下列操作步骤：

- 通过调用 SFC 39"DIS\_IRT"或 SFC 41"DIS\_AIRT"，禁止或延迟出现具有更高优先级的中断和异步错误。
- 调用 SFC 60"GD\_SND"。
- 通过调用 SFC 40"EN\_IRT"或 SFC 42"EN\_AIRT"，重新允许具有更高优先级的中断和异步错误。

参数	声明	数据类型	存储器区	描述
CIRCLE_ID	INPUT	BYTE	I、Q、M、D、L、 常数	待发送 GD 信息包所在的 GD 环编号。在使用 STEP 7 组态全局数据时指定该编号。 允许使用的数值：1 到 16。可以在 CPU 的技术数据中找到允许使用的 GD 环的最大数目。
BLOCK_ID	INPUT	BYTE	I、Q、M、D、L、 常数	要在选定 GD 环中发送的 GD 信息包的编号。在通过 STEP 7 组态全局数据时设置该编号。 允许使用的数值：1 到 3。可以在 CPU 的技术数据中找到允许使用的 GD 环的最大数目。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息

**错误信息**

错误代码 (W#16#...)	解释
0000	未产生故障。
8081	没有组态使用参数 CIRCLE_ID 和 BLOCK_ID 选定的 GD 信息包。
8082	参数 CIRCLE_ID 或 BLOCK_ID 的数值非法或这两个参数的数值都非法。
8083	执行 SFC 时出错。在为状态信息组态的变量中输入该错误类型。这可通程序进行判断。
8084	由于为较高优先级的同一个 GD 信息包再次调用 SFC 60，终止 SFC 的执行(参见"中断能力")。
8085	在将状态信息输入到所组态的变量中时出错。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

**注意**

在每次调用 SFC 60 之后，都应该评估相应 GD 信息包的状态，并且，如果有必要，则复位该状态。

## 20.2 通过 SFC 61 "GD\_RCV"编程接受已接收到的 GD 信息包

### 描述

通过 SFC 61"GD\_RCV"(全局数据接收),从进入的 GD 帧中为单个 GD 信息包提取数据,然后输入接收到的 GD 信息包中。必须已经通过 STEP 7 对此进行了组态。

可以在用户程序的任何地方调用 SFC 61"GD\_SND"。

扫描速率以及系统在循环检测点处所进行的数据提取并不受 SFC 61 调用的影响。

### 中断能力

SFC 61 可以被具有更高优先级的程序中中断,然而,其前提是必须保证为全局数据定义的数据一致性。如果函数的处理被中断,则可能是:为同一个 GD 信息包以更高优先级再次调用了 SFC 61。

然后,将数据输入到在更高优先级中接收到的 GD 信息包中。当程序返回到中断的 SFC 时,将立即终止该过程。

### GD 的数据一致性

下列规则适用于在各种存储区域输入的数据的一致性。

下列数据是一致的:

- 简单数据类型(位、字节、字和双字)
- 数据类型为字节、字和双字的数组,最大数据长度取决于接收 CPU。

### 确保整个 GD 信息包的一致性

接收 CPU 上的 GD 信息包所具有的结构不会自动保证它的数据来自一个相同的帧。例如，当信息包由三个 GD 单元组成时就是这种情况。

然而，如果需要整个 GD 信息包具有一致性，请在程序中遵守下列操作步骤：

- 通过调用 SFC 39"DIS\_IRT"或 SFC 41"DIS\_AIRT"，禁止或延迟出现具有更高优先级的中断和异步错误。
- 调用 SFC 60"GD\_SND"。
- 通过调用 SFC 40"EN\_IRT"或 SFC 42"EN\_AIRT"，重新允许具有更高优先级的中断和异步错误。

参数	声明	数据类型	存储器区	描述
CIRCLE_ID	INPUT	BYTE	I、Q、M、D、L、常数	将输入进入的 GD 信息包的 GD 环编号。在通过 STEP 7 组态全局数据期间指定该编号。 允许使用的数值：1 到 16。可以在 CPU 的技术数据中找到允许使用的 GD 环的最大数目。
BLOCK_ID	INPUT	BYTE	I、Q、M、D、L、常数	将输入进来数据的选定 GD 环中的 GD 信息包的编号。在通过 STEP 7 组态全局数据期间指定该编号。 允许使用的数值：1 到 3。可以在 CPU 的技术数据中找到允许使用的 GD 环的最大数目。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8081	没有组态使用参数 CIRCLE_ID 和 BLOCK_ID 选定的 GD 信息包。
8082	参数 CIRCLE_ID 或 BLOCK_ID 的数值非法或这两个参数的数值都非法。
8083	执行 SFC 时出错。在为状态信息组态的变量中输入该错误类型。这可通程序进行判断。
8084	由于为较高优先级的同一个 GD 信息包再次调用 SFC 61，终止 SFC 的执行(参见"中断能力")。
8085	在将状态信息输入到所组态的变量中时出错。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

### 注意

在每次调用 SFC 61 之后，都应该评估相应 GD 信息包的状态，并且，如果有必要，则复位该状态。



# 21 S7 通讯和 S7 基本通讯的概述

## 21.1 S7 通讯块和 S7 基本通讯块之间的差别

### 选择标准

除了全局数据通讯以外，SIMATIC S7 可编程控制器的 CPU/FM 之间还有其它两种数据交换方式：

- 对于未组态的 S7 连接，使用通讯 SFC 进行数据交换
- 对于已组态的 S7 连接，使用通讯 SFB 进行数据交换

所选择的方法取决于所使用的 SIMATIC S7 可编程控制器(S7-300、S7-400)和用于数据交换的其它参数。可以基于下表包含的标准列表。

标准	用于未组态的 S7 连接的通讯 SFC	用于已组态的 S7 连接的通讯 SFB
程序块的可用性	S7-300 和 S7-400 作为 SFC	S7-300 作为 FB 和 FC S7-400 作为 SFB 和 SFC
通讯连接	连接没有组态。当激活 SFC 时连接建立。在数据已经传送完毕之后，连接或者保持建立，或者由参数控制而终止。 如果暂时无法建立连接，则相应作业将无法发送。	在系统配置中永久配置连接。
切换到 STOP 模式	如果启动数据传送的 CPU 切换到 STOP 模式，则将终止它所建立的所有连接。	此连接将保持在 STOP 模式。
到一个伙伴的几个连接	在任何时刻，一个连接最多只能连接到一个通讯伙伴。	可以建立到同一个伙伴的几个连接。
地址范围	可以在本地站点或在 MPI 子网中寻址的模块	可以在 MPI 网络、PROFIBUS 或工业以太网上寻址的模块。
通讯伙伴数	可依次到达的通讯伙伴的数目不受可用的连接资源的限制(参见/70/、/101/)。(当程序正在运行时，可以重新建立和终止连接。)	可以同时获得的通讯伙伴数受可用的连接资源数限制。它还取决于所使用的 CPU(参见 /70/、/101/)。
最大用户数据长度	保证 76 个字节的用户数据长度。	可传送的最大用户数据长度取决于块类型(USEND/URCV、GET 等)和通讯伙伴(S7-300、S7-400 或 M7)。
每个块调用所传送的变量数	只能传送一个变量。	<ul style="list-style-type: none"> <li>• S7-300：一个变量</li> <li>• S7-400：最多四个变量</li> </ul>
功能块分类	用于 S7 基本通讯的 SFC 是系统函数。因此，它们并不需要用户存储区。	用于 S7 基本通讯的 SFB/FB 是系统函数块。因此，它们需要实例 DB 来提供实际参数和静态数据。

标准	用于未组态的 S7 连接的通讯 SFC	用于已组态的 S7 连接的通讯 SFB
地址参数的动态修改	可以进行地址参数的动态修改：在激活的作业结束时，可以寻址其它通讯伙伴。	<p>S7-300：可以在功能块激活时重新组态寻址参数。当前一个作业关闭时，新参数开始有效。</p> <p>S7-400：不能进行地址参数的动态修改：连接由第一次块调用指定并固定，且在下一次暖启动或冷启动之前保持不变。</p>

### 使用 S7-400 实现 IEC 61131-5

通过下列功能块来实现 IEC 标准 61131-5：

- USEND(SFB 8)/URCV(SFB 9)
- BSEND(SFB 12)/BRCV(SFB 13)
- PUT(SFB 15)/GET(SFB 14)，相当于 READ/WRITE
- STATUS(SFB 22)/USTATUS(SFB 23)
- ALARM(SFB 33)
- NOTIFY(SFB 36)
- START (SFB 19)、STOP (SFB 20)和 RESUME (SFB 21)实现用于程序控制函数的调用接口。

### 使用 S7-300 实现 IEC 61131-5

通过下列功能块来实现 IEC 标准 61131-5：

- USEND(SFB 8)/URCV(SFB 9)
- BSEND(SFB 12)/BRCV(SFB 13)
- PUT(SFB 15)/GET(SFB 14)，相当于 READ/WRITE
- USEND\_E(FB28)/URCV\_E(FB29)
- PUT\_E(FB35)/GET\_E(FB34)对应于 READ/WRITE

## 21.2 数据一致性

### 定义

可以被一致性过程同时修改的数据区的大小被称为一致数据区。大于一致数据区的数据区将被认为整体失真。

也就是说，属于一组且大于一致数据区的数据区可以同时包含新的一致数据块部分和旧的一致数据块部分。

### 实例

如果通讯块被中断，例如，被具有更高优先级的硬件中断 OB 所中断，则可能会引起数据不一致。如果此 OB 中的用户程序现在改变数据，而一部分数据已经由通讯块处理过，则已传送的数据：

- 部分来自处理硬件中断之前的时间
- 部分来自处理硬件中断之后的时间
- 也就是说，这些数据将不一致(不连贯)。

### 影响

如果数据的较大数据包以一致格式传送，则传送可能会被中断。这可能会，例如，增加 CPU 中的中断反应时间。

这表示：需要绝对一致传送的数据量越大，系统的中断反应时间越长。

### SIMATIC 上的数据一致性

如果用户程序包含访问公用数据的通信函数，例如 BSEND/BRCV，则可以通过使用参数"DONE"来协调对此数据区的访问。这样，便可以在用户程序中确保通过通讯块本地传送的通讯区的数据一致性。

然而，当使用 S7 通讯函数时，例如 PUT/GET 或通过 OP 通讯进行写/读时，在编程或组态期间必须考虑一致数据区的大小，因为在目标设备(服务器)的用户程序中没有通讯块可用于向用户程序发送同步通讯数据。

在 S7-300 和 C7-300 中(例外：CPU 318-2 DP)，在操作系统的周期检测点上，通讯数据以 32 个字节的数据块为单位，一致地复制到用户存储区。对于较大数据区，不能保证数据一致性。如果要求定义数据的一致性，则用户程序中的通讯数据不可以超出 32 个字节(根据版本的不同，最大为 8 个字节)。

与此相反，在 S7-400 中，通讯数据不是在周期检测点上处理，而是在程序循环周期的固定时间段上进行处理。变量的一致性由系统来确保。

随后，可使用 PUT/GET 函数或变量的读/写来一致访问这些通讯区，例如通过 OS 上的 OP。

### 建议

关于数据一致性的更多信息，请参考各个块的信息描述以及 *通过 SIMATIC 进行通讯手册*。

## 21.3 S7 通讯块概述

### 分类

对于 S7 通讯，需要进行连接组态。通过 SFB/FB 或 SFC/FC，在应用程序中调用集成的通讯功能。

可以将这些块分成下列几个类别：

- 用于数据交换的 SFB/FB
- 用于改变工作状态的 SFB
- 用于查询工作状态的 SFB
- 用于查询连接的 SFC/FC

### 可用的 S7-300/400 通讯块

- 用于 S7-400 的块位于"标准库"中。
- S7-300 的可装载块位于"SIMATIC\_NET\_CP"库或"通讯块"的"标准库"中...

若 S7-300 CPU 没有一个 Ethernet 接口，则要求使用来自 S7-300 系列的一个 SIMATIC NET CP 来执行 S7-300 块。欲知详情，请参见相关文档。

### 用于数据交换的 SFB/FB

用于数据交换的通讯 SFB/FB 被用于在两个通讯伙伴之间交换数据。如果只在本地模块上存在 SFB，则称为单边数据交换。如果在本地模块和远程模块上都存在 SFB/FB，则称为双边数据交换。

块 S7-400	块 S7-300	描述	简要描述
SFB 8	FB -8	USEND	无确认的快速数据交换，不考虑通讯伙伴上通讯函数 (URCV)的相应执行情况(例如，操作和维护消息)。也就是说，通讯伙伴上的数据可以被更多最新的数据所覆盖。
SFB 9	FB 9	URCV	
-	FB 28 FB 29	USEND_E URCV_E	
SFB 12	FB 12	BSEND/	到通讯伙伴的固定数据块传送。也就是说，在通讯伙伴中的接收函数(BRCV)接受该数据之前，数据传送不会结束。
SFB 13	FB 13	BRCV	
SFB 14	FB -14	GET	通讯伙伴用户程序中，由程序控制的变量读功能，无更多的通讯功能。
-	FB 34	GET_E	
SFB 15	FB 15	PUT	通讯伙伴用户程序中，由程序控制的变量写功能，无更多的通讯功能。
-	FB 35	PUT_E	
SFB 16	FB 16	PRINT	向打印机发送数据(只用于 S7-400)

### 用于改变工作状态的 SFB

通过用于改变工作状态的 SFB，可以控制远程设备的工作状态。

通过用于改变工作状态的 SFB 进行的数据交换是单边数据交换。

S7-400 中使用的块		简要描述
SFB 19	START	如果 S7/M7-300/400 或 C7-300 CPU 处于 STOP 工作模式，则该功能块将触发这些模块的重启动(RESTART)功能。
SFB 20	STOP	如果 S7/M7-300/400 或 C7-300 CPU 处于 RUN、HALT 工作模式或正在启动，则该功能块将触发这些模块的停机(STOP)功能。
SFB 21	RESUME	如果 S7-400 CPU 处于 STOP 模式，则该功能块将触发此模块，使其继续运行。

### 用于查询工作状态的 SFB

通过用于查询工作状态的 SFB，可以获得远程设备的工作状态信息。

使用 SFB"STATUS"时，数据交换是单边的；而使用 SFB"USTATUS"时，数据交换是双边的。

S7-400 中使用的块		简要描述
SFB 22	STATUS	在用户请求时，提供通讯伙伴的工作状态(S7-400-CPU、M7-300/400)。
SFB 23	USTATUS	如果已经设置了相应的连接属性(发送工作状态消息)，则当 CPU 的工作状态发生改变时，将接收 S7-400-CPU 的工作状态。

### 用于查询连接的 SFC/FC

S7-400 中使用的块	S7-300 中使用的块	简要描述
SFC 62 CONTROL		查询连接的状态，该连接属于某个 SFB/FB 的一个实例。
	FC 62 C CNTRL	通过连接标识号查询连接的状态

#### 提示：

还可使用 SFC 87 C\_DIAG 来执行实际连接状态的诊断(仅限 S7-400)。

### 实例程序

STEP 7 提供了一个 S7-400 的示例程序，该程序给出了如何使用 SFB 进行 S7 通讯。这个示例程序的名称是 step7\examples\com\_sfb。它位于路径文件夹"..\STEP7\Examples\ZDT01\_10"中的示例程序中。

## 21.4 S7 通讯块概述

### 用于 S7 基本通讯的 SFC 的分类

S7 基本通讯不需要连接组态。通过用户程序中的 SFC 调用集成的通讯函数。

SFC 分为两类：

- 用于在 S7 CPU 和其它有通信功能的模块之间交换数据的 SFC，条件是通信伙伴属于相同的 S7 站(开头以"I"标识，表示**内部**)。
- 用于在 S7 CPU 和其它有通信功能的模块之间交换数据的 SFC，条件是通信伙伴连接到公共 MPI 子网(开头以"X"标识，表示**外部**)。

对于与其它子网中的站的通讯，不能使用用于 S7 基本通讯的 SFC。

用于基本通讯的 SFC 可以运行在 S7-300 和 S7-400 系列的所有 CPU 上。通过这些 CPU，还可以向 S7-200 的 CPU 中写入变量，以及从 S7-200 的 CPU 中读取变量。

### 用于外部通讯的 SFC

块		简要描述
SFC 65/ SFC 66	X_SEND/ X_RCV	确保将数据块传送给通讯伙伴。也就是说，在通讯伙伴中的接收函数(X_RCV)接受该数据之前，数据传送不会结束。
SFC 67	X_GET	读取通讯伙伴中的变量，而不需要在通讯伙伴上运行相应的 SFC。在通讯伙伴的操作系统中实现此功能。
SFC 68	X_PUT	向通讯伙伴写入变量，而不需要在通讯伙伴上运行相应的 SFC。在通讯伙伴的操作系统中实现此功能。
SFC 69	X_ABORT	中止一个已存在的、无正在传送数据的连接。然后再在通讯连接的两端释放相应的连接资源。

### 用于内部通讯的 SFC

块		简要描述
SFC 72	I_GET	读取通讯伙伴中的变量，而不需要在通讯伙伴上运行相应的 SFC。在通讯伙伴的操作系统中实现此功能。
SFC 73	I_PUT	向通讯伙伴写入变量，而不需要在通讯伙伴上运行相应的 SFC。在通讯伙伴的操作系统中实现此功能。
SFC 74	I_ABORT	中止一个已存在的、无正在传送数据的连接。然后再在通讯连接的两端释放相应的连接资源。

## 实例程序

随 STEP 7 提供了两个用于 S7 基本通讯的 SFC 的实例程序。这两个实例程序包含在目录 step7\examples\com\_SFC1 和 step7\examples\com\_SFC2 中。

## 最大用户数据长度

在 S7-300 和 S7-400 的所有 CPU 上都集成了用于未组态的 S7 连接的通讯 SFC。

所有 SFC 都保证可以传送 76 个字节的用户数据(参数 SD 或 RD)。可以在各个块上找到准确的字节数。

## 到通讯伙伴的连接

通过用于未组态的 S7 连接的通讯 SFC，在 SFC 执行时建立连接。根据分配给 CONT 输入参数的数值，或者保持建立连接，或者在数据交换结束时终止连接。也就是说，通讯具有下列特征：

- 可依次到达的通讯伙伴的数目高于可同时到达的通讯伙伴的数目(数目取决于特定的 CPU，请参见 /70/、/101/)。
- 如果由于所有连接资源(在本地 CPU 或在通讯伙伴上)都正被使用，当前不能建立任何与通讯伙伴的连接，则通过 RET\_VAL 来指示。必须在随后的某个合适的时间点上重新触发此作业。然而，并不能保证随后可以成功建立连接。如必要，检查程序中连接资源的使用情况，并使用带有更多资源的 CPU。

已存在的、用于已组态的 S7 连接的通讯 SFB 的连接，不能被用于未组态的 S7 连接的通讯 SFC 使用。

一旦触发了一个作业，则为此作业建立的连接只能用于此特定作业。只有在当前作业结束之后，才能执行其它包含相同通讯伙伴的作业。

---

### 注意

如果程序中包含了几个涉及到同一个通讯伙伴的作业，则必须确保在随后的某个合适的时间点上重新调用为其 RET\_VAL 输入了 W#16#80C0 的 SFC。

---

## 识别作业

如果已经通过一个用于未组态 S7 连接的通讯 SFC 触发了一次数据传送或一个连接中止，并在当前传送结束之前重新调用这个 SFC，则 SFC 的反应取决于新调用是否包含了相同的作业。下表解释了哪个输入参数为每个 SFC 指定一个作业。如果参数与还没有结束的作业相匹配，则将此 SFC 调用算作一个后继调用。

块		用于识别作业的标识符
SFC 65	X_SEND	DEST_ID、REQ_ID
SFC 67	X_GET	DEST_ID、VAR_ADDR
SFC 68	X_PUT	DEST_ID、VAR_ADDR
SFC 69	X_ABORT	DEST_ID
SFC 72	I_GET	IOID、LADDR、VAR_ADDR
SFC 73	I_PUT	IOID、LADDR、VAR_ADDR
SFC 74	I_ABORT	IOID、LADDR

## 响应中断

用于未组态 S7 连接的通讯 SFC 可以被具有更高优先级的 OB 中断。如果带有完全相同作业的不同 SFC 被中断 OB 重新调用，则中止此第二次调用，并在 RET\_VAL 中输入一个相应的条目。然后继续执行中断 SFC。

## 访问 CPU 的工作存储器

不管要传送的用户数据量有多大，操作系统的通讯函数以最大长度的数据域来访问 CPU 的工作存储器，因而不会因为通讯函数的使用而延长中断反应时间。

根据通过 STEP 7 设置的通讯引起的最大周期性负载，在作业的执行期间，操作系统的通讯函数可以多次访问工作存储器。

## 客户机切换到 STOP

如果启动作业(并因此建立连接)的 CPU 在数据传送期间切换到 STOP，则将终止由它所建立的所有连接。

## 更改程序

对于程序中所有可以立即对用于未组态 S7 连接的通讯 SFC 调用产生影响的部分，只能在处于 STOP 模式时对此进行修改。特别地，这包括删除包含用于未组态 S7 连接的通讯 SFC 调用的 FC、FB 或 OB。

在修改程序之后，必须执行暖启动或冷启动。

不遵守这些规则，可能会使资源保持已分配状态，以及可编程控制器随后进入未定义的状态。



## 22 S7 通讯

### 22.1 用于 S7 通讯的 SFB/FB 和 SFC/FC 的公用参数

#### 分类

根据功能，用于组态 S7 连接的通讯 SFB/FB 的参数可分为以下五个类别：

1. 控制参数，用于激活块。
2. 寻址参数，用于寻址远程通讯伙伴。
3. 发送参数，指向将要发送到远程伙伴的数据区。
4. 接收参数，指向用于输入从远程伙伴接收到的数据的数据区。
5. 状态参数，用于监视块是否已经无错地完成了它的任务，或用于分析任何已经发生的错误。

#### 控制参数

只有当调用 SFB/FB 时，相应的控制参数具有已定义的信号状态(例如被置位)，或自从前一次 SFB/FB 调用之后，信号状态发生了特定变化(例如，上升沿)时，数据交换才会被激活。

---

#### S7-300 的注意事项

对于第一次调用，需要将参数 REQ 设置为 FALSE。

---

#### 寻址参数

参数	描述
ID	指向本地连接描述的引用(由 STEP 7 连接组态定义)。注意：ID W # 16 # EEEE 不允许用于 S7 通讯块。
R_ID	使用 R_ID 参数来指定一对发送 SFB 和接收 SFB：发送端 SFB/FB 的 R_ID 参数必须和接收端 SFB 的 R_ID 参数相同。 这允许几对 SFB/FB 通过相同的逻辑连接进行通讯。 <ul style="list-style-type: none"><li>• R_ID 必须以 DW#16#wxyzWXYZ 的形式指定。</li><li>• 通过 R_ID 指定的一个逻辑连接的几个程序块对，对于这个连接来说必须是唯一的。</li></ul>

参数 PI\_NAME 只在相关 SFB 中进行描述(只用于 S7-400)。

---

**注意**

S7-300：当寻址参数 ID 和 R ID 已激活时，可以改变这些参数。在前一个作业关闭之后，当新建一个作业时新参数开始生效。在此，可以在一个实例中链接多个 FB 对。

**提示：**下列一些方法可以用于保存实例 DB，进而保存工作存储器：

1st 通过变量 ID，可以通过一个数据实例块来使用多个连接。

2nd 通过变量 R\_ID，可以为一个作业定义多个发送和接收 FB 对的标识符。

3rd 方法 1 和方法 2 可以同时使用。

4th 请注意，新参数在最后一个作业执行完毕之后才开始生效。如果激活了发送作业，则发送和接收 FB 的 R\_ID 参数必须相互匹配。

**S7-400：**寻址参数 ID 和 R ID 只在第一次调用块时计算(实际参数或来自实例的预定义值)。因此，第一次调用块时所指定的与远程伙伴的通讯关系(连接)会持续到下一次暖启动或冷启动。

---

## 状态参数

通过状态参数，可以监视块是否已经正确地完成了它的任务，或块是否还处于工作状态。状态参数还可以用于指示是否出错。

---

**注意**

状态参数只在一个周期内有效，即从 SFB/FB 调用之后的第一个指令开始到下一个 SFB/FB 调用为止。所以，必须在每个块周期之后评估这些参数。

---

## 发送和接收参数

如果不使用 SFB/FB 的所有发送或接收参数，则第一个未使用的参数必须为一个 NIL 指针(参见/232/)，且所使用的参数必须依次相邻，中间不能出现间隙。

---

**S7-400 的注意事项**

在第一次调用期间，ANY 指针指定可为作业传送的最大用户数据量。即，在 CPU 的工作存储器中创建一个通讯缓冲区，以确保数据的一致性。这个缓冲区最多占用 480 个字节的工作存储器。如果当 CPU 处于 RUN 模式时，没有通过 SFB 调用重新装载块，则建议在暖重启或冷重启 OB 中运行第一次调用。

在后来的调用中可以发送/接收任意数量的数据，但是不能超过第一次调用所发送/接收的数据量。

本规则的例外情况是 BSEND 和 BRCV 系统功能块。通过这两个功能块，每个作业可以发送多达 64 K 字节的数据(参见通过 SFB/FB 12 "BSEND"发送分段数据和通过 SFB/FB 13 "BRCV"接收分段数据)

---

对于用于双边通讯的 SFB/FB：

- 在发送端和接收端，所使用的 SD\_i 和 RD\_i 参数的数目必须匹配。
  - 在发送端和接收端，属于一对的 SD\_i 和 DR\_i 参数的数据类型必须匹配。
  - 根据 SD\_i 参数发送的数据量不得超出相应 RD\_i 参数指定的可用区域(不适用于 BSEND/BRCV)。
- ERROR = 1 和 STATUS = 4 表示用户违反了上面的规则。

此最大用户数据长度取决于远程伙伴是 S7-300 还是 S7-400。

## 用户数据大小

对于 SFB/FB USEND、URCV、GET 和 PUT 以及 FB USEND\_E、URCV\_E、GET\_E 和 PUT\_E，要传送的数据量不得超出最大用户数据长度。最大用户数据大小取决于：

- 所使用的块类型和
- 通讯伙伴。

下面的表格列出了使用带 1- 4 个变量的 SFB/FB 时，可保证的最小用户数据大小：

块	伙伴：S7-300	伙伴：S7-400
PUT/GET	160 个字节	400 个字节
USEND/URCV	160 个字节	440 个字节
BSEND/BRCV	32768/65534 个字节	65534 个字节
PUT_E / GET_E	160 个字节	不存在
USEND_E / URCV_E	160 个字节	不存在

关于用户数据大小的更多信息，请参见各 CPU 技术数据。

### 精确的用户数据大小

如果上面指定的用户数据大小不够，则可以如下确定用户数据的最大字节长度：

1. 首先，从下面的表格中读取有效的通讯数据块大小：

本地 CPU	远程 CPU	数据块大小，以字节为单位
S7-300	Any	240 (S7-300)
S7-400	S7-300/C7-300	240 (S7-400)
S7-400	S7-400 或 CPU 318	480

2. 使用下表中的该值来读取最大可能的用户数据长度(字节)作为要使用的参数总数。此表适用于 SD<sub>i</sub>、RD<sub>i</sub> 和 ADDR<sub>i</sub> 区域的偶数长度。在任意奇数长度的区域中，最大可能的用户数据长度减少一个字节。

		所使用的参数 SD <sub>i</sub> 、RD <sub>i</sub> 和 ADDR <sub>i</sub> 的数目			
数据块大小	SFB/FB	1	2	3	4
240 (S7-300)	PUT/GET/ USEND	160	-	-	-
240 (S7-300 , 通过集成接口)	PUT	212	-	-	-
	GET	222	-	-	-
	USEND	212	-	-	-
	PUT_E	212	196	180	164
240 (S7-400)	GET_E	222	218	214	210
	USEND_E	212	208	204	200
	PUT	212	196	180	164
	GET	222	218	214	210
480	USEND	212	-	-	-
	PUT	452	436	420	404
	GET	462	458	454	450
960	USEND	452	448	444	440
	PUT	932	916	900	884
	GET	942	938	934	930
	USEND	932	928	924	920

## 22.2 用于组态的 S7 连接的 SFB 启动例行程序

### 要求

在 S7-400 的下列描述中，假设下列条件成立：

- 在模块上存在连接描述(SDB)。
- 所组态的连接已经建立。
- ID 的实际参数与为每个 SFB 组态的连接 ID 匹配。

### 暖启动和冷启动

在暖启动和冷启动期间，所有 SFB 都设置为 NO\_INIT 状态。存储在实例 DB 中的实际参数不会改变。

### 通过 SFB 执行暖启动和冷启动，以实现双边数据交换

通常，用于双边数据交换的带有 SFB 的两个模块不会同时暖启动或冷启动。SFB 的反应受下列规则的控制：

接收块(SFB URVCV、BRVCV)的反应如下：

- 如果在暖启动或冷启动时，SFB 已经接收了一个作业，但是还没有确认这个作业，则 SFB 将产生一个序列中止帧(CFB、BRVCV)，并立即跳转到 NO\_INIT 状态。
- 通过 SFB BRVCV，即使已经发送了序列中止，仍然可以接收其他数据段。将在本地丢弃该数据帧。
- SFB URVCV 立即切换到 NO\_INIT 状态。

发送块(SFB USEND、BSEND)的反应如下：

- 如果 SFB BSEND 已经启动了一个还没有完成的作业序列，则当启动暖启动或冷启动时，它将发送一个序列中止。然后立即跳转到 NO\_INIT 状态。将在本地丢弃此后到达的应答。
- 如果当请求暖启动或冷启动时，SFB BSEND 已经发送或接收了一个序列中止，则它立即切换到 NO\_INIT 状态。
- 在所有其它情形中，只要 SFB 只发送消息(例如，SFB USEND)，则将中止本地处理，且 SFB 立即跳转到 NO\_INIT 状态。

### 通过 SFB 执行暖启动和冷启动，以实现单边数据交换

可以认为，在连接建立之后，通讯伙伴上的服务器是运行的，即，服务器可以在任何时刻处理作业或输出消息。

发送作业并期待应答的 SFB 对完全重新启动的反应如下：

中止当前处理，然后 CFB 立即跳转到 NO\_INIT 状态。如果在暖启动或冷启动之前发送的作业应答随后到达，则将在本地丢弃该应答。

在前面作业的应答接收到之前，新作业可能已经发送。

输出或接收消息的 SFB 反应如下：

- 中止当前处理，然后 CFB 立即跳转到 NO\_INIT 状态。
- 通过 SFB USTATUS，在本地丢弃在 NO\_INIT 和 DISABLED 状态期间到达的消息。

### 对热启动的反应

只有在暖启动或冷启动期间，用于 S7 通讯的 SFB 才被设置为 NO\_INIT 状态。也就是说，它们的反应和可以在热启动之后重新恢复运行的用户功能块一样。

### 对存储器复位的反应

存储器复位将终止所有连接。由于在存储器复位之后，对于用户程序来说，暖启动或冷启动是唯一可以运行的启动类型，所以所有用于 S7 通讯的 SFB (如果仍然存在)都被设置为 NO\_INIT 状态并被初始化。作为对连接终止的反应，存储器没有复位的模块中的伙伴块将切换到 IDLE、ENABLED 或 DISABLED 状态。

## 22.3 SFB 对故障如何反应

下面讲述了 S7-400 中用于 S7 通讯的 SFB 对故障如何反应。

### 连接被终止

监视分配给 SFB 实例的连接。

如果终止连接，则 SFB 的反应取决于它的内部状态。

如果在块处于 IDLE 或 ENABLED 状态时检测到连接断开，则 SFB 反应如下：

- SFB 跳转到 ERROR 状态，并在 ERROR 和 STATUS 输出参数上输出出错 ID"通讯故障"。
- 当下一次调用时，功能块返回到其初始状态，并重新检查此连接。

不处于 IDLE 或 DISABLED 状态的通讯 SFB 反应如下：

- SFB 中止处理，立即或在下一个块调用时切换到 ERROR 状态，并在 ERROR 和 STATUS 输出参数上输出出错 ID"通讯故障"。
- 当下一次调用 SFB 时，功能块切换到 IDLE、DISABLED 或 ENABLED 状态。在 IDLE 和 ENABLED 状态中重新检查连接。

如果在此期间重新建立连接，则也会执行此过程。

### 电源掉电

带备用电池的电源掉电，随后重新启动，这将终止所有已建立的连接。因此，上面讲到的各点适用于所有有关的功能块。

如果发生带备用电池的电源掉电，随后自动暖启动或冷启动，则终止连接和暖启动或冷启动的各个相关事项都适用。

在无备用电池的自动暖启动或冷启动特例中，电源恢复之后自动执行存储器复位，用于 S7 通讯的 SFB 的反应可参见"用于 S7 通讯的 SFB 启动例行程序"一节中的描述。

### 对操作模式切换的反应

如果操作模式在 STOP、START、RUN 和 HOLD 状态之间切换，则通讯 SFB 保持其当前状态(例外：在暖启动或冷启动期间，它将切换到 NO\_INIT 状态)。这既适用于单边通讯 SFB，也适用于双边通讯 SFB。

### 到用户程序的出错接口

如果在通讯 SFB 的处理期间出错，则它将切换到 ERROR 状态。在 ERROR 输出参数置位为 1 的同时，将相应出错 ID 输入到 STATUS 输出参数。可以在用户程序中判断此出错信息。

可能发生的错误实例：

- 收集发送数据时出错。
- 复制接收数据到接收区时出错(例如，尝试访问一个并不存在的 DB)。
- 要发送的数据区的长度和在伙伴 SFB 中指定的接收区的长度不匹配。



## 22.4 通过 SFB 8/FB 8"USEND"进行无协调的数据发送

---

### 注意

S7-300 CPU 有下列两个 FB8 "USEND"版本：

- 一个版本位于标准库中。如果到远程伙伴的连接是通过 PN 接口的，则使用该版本。
  - 另一个版本位于 SIMATIC\_NET\_CP 库中。如果到远程伙伴的连接是通过 CP 接口的，则使用该版本。
- 

### 描述

SFB/FB 8"USEND"向类型为"URCV"的远程伙伴 SFB/FB 发送数据。执行发送过程而不需要和 SFB/FB 伙伴进行协调。也就是说，在进行数据传送时不需要伙伴 SFB/FB 进行确认。

**S7-300**：在 REQ 的上升沿处发送数据。在 REQ 的每个上升沿处传送参数 R\_ID、ID 和 SD\_1。在每个作业结束之后，可以给 R\_ID、ID 和 SD\_1 参数分配新数值。

**S7-400**：在控制输入 REQ 的上升沿处发送数据。通过参数 SD\_1 到 SD\_4 来指向要发送的数据，但并非都需要用到所有四个发送参数。

然而，必须确保参数 SD\_1 到 SD\_4/SD\_1 和 RD\_1 到 RD\_4/RD\_1 (在相应通讯伙伴 SFB/FB"URCV"上)所定义的区域在以下几个方面保持一致：

- 编号
- 长度，和
- 数据类型。

参数 R\_ID 必须在两个 SFB 中完全相同。

如果传送成功完成，则通过状态参数 DONE 来表示，此时其逻辑数值为 1。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求，在上升沿激活数据交换
ID	INPUT	WORD	M、D、常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
R_ID	INPUT	DWORD	I、Q、M、D、L、常数	寻址参数 R_ID，请参见用于 S7 通讯的 SFB 和 SFC 的公共参数
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
S7-300 : SD_1  S7-400 : SD_i (1 ≤ i ≤ 4)	IN_OUT	ANY	M、D、T、Z  I、Q、M、D、 T、C	指针，指向第 i 个发送数据区。 仅允许使用下列数据类型：BOOL (不允许使用：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME、COUNTER、TIMER。 <b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须要指定该数据块(例如： P# DB10.DBX5.0 Byte 10)。

## 错误信息

ERROR	STATUS (十进制)	解释
0	11	警告： <ul style="list-style-type: none"> <li>• 由于前一个作业还没有结束，所以不能执行新作业。</li> <li>• 正在以较低优先级处理此作业。</li> </ul>
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如： <ul style="list-style-type: none"> <li>• 没有装载连接描述(本地或远程)</li> <li>• 连接被中断(例如：电缆断线、CPU 关闭、或 CP 处于 STOP 模式)</li> <li>• 没有建立到通讯伙伴的连接</li> <li>• FB 无法在 S7-400 CPU 上运行</li> <li>• 此外对于 S7-300：  <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> </ul> </li> </ul>
1	4	发送数据区指针 SD_i 中的出错，包括数据长度或数据类型。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)。
1	12	当调用 SFB 时 <ul style="list-style-type: none"> <li>• 指定了一个不属于 SFB"USEND"的实例 DB</li> <li>• 没有指定任何实例 DB，而是指定了全局数据块。</li> <li>• 没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	18	<ul style="list-style-type: none"> <li>• 在连接 ID 中已经存在 R_ID。</li> <li>• 此外对于 S7-300：  <ul style="list-style-type: none"> <li>- CPU 在 RUN 模式(要求 CPU 或 CP 进行从 STOP 到 RUN 的切换)时，实例超载。</li> </ul> </li> </ul>
1	20	<ul style="list-style-type: none"> <li>• S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>• H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> <li>• S7-300：  <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> <li>- CPU 在 RUN 模式(要求 CPU 或 CP 进行从 STOP 到 RUN 的切换)时，实例超载。</li> <li>- 当第一次调用时可以</li> </ul> </li> </ul>
1	27	CPU 中没有用于此功能块的功能代码(仅对于 S7-300)

### 数据一致性

S7-300 : 为确保数据一致性,只能在当前发送操作结束之后,重新向发送区 SD\_1 写入数据。状态参数 DONE 的数值变为 1 时就是这种情形。

S7-400 和 S7-300 通过集成接口 : 当激活一个发送操作时(在 REQ 的上升沿),从用户程序中复制要从发送区 SD\_i 发送的数据。在块调用之后,可以在不破坏当前发送数据的前提下向这些区域写入数据。

---

#### 注意

只有当 DONE 状态参数数值为 1 时,发送操作才完成。

---

## 22.5 通过 SFB/FB 9 "URCV"进行无协调的数据接收

### 描述

SFB/FB 9"URCV"从类型为"USEND"的远程伙伴 SFB/FB 中异步接收数据，并把接收到的数据复制到组态的接收区域内。

当 EN\_R 输入为逻辑 1 时，表明块已准备好接收。可以通过 EN\_R=0 来取消已激活的作业。

**S7-300** : 在 EN\_R 的每个上升沿处应用参数 R\_ID、ID 和 RD\_1。在每个作业结束之后，可以给 R\_ID、ID 和 RD\_1 参数分配新数值。

**S7-400** : 通过参数 RD\_1 到 RD\_4 来指向接收数据区。

但是，必须确保参数 RD\_i/RD\_1 和 SD\_i/SD\_1 (在相应通讯伙伴 SFB/FB"USEND"上)所定义的区域在以下几个方面保持一致：

- 编号
- 长度，和
- 数据类型。

通过 NDR 状态参数逻辑数值为 1 来指示已经成功完成复制处理过程。

参数 R\_ID 必须在两个 SFB/FB 上完全相同。

参数	声明	数据类型	存储器区	描述
EN_R	INPUT	BOOL	I、Q、M、D、L、常数	经激活可进行接收的控制参数在输入置位时发出信号，通知该对象准备接收。
ID	INPUT	WORD	M、D、常数	寻址参数 ID，请参见 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
R_ID	INPUT	DWORD	I、Q、M、D、L、常数	寻址参数 R_ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
NDR	OUTPUT	BOOL	I、Q、M、D、L	NDR 状态参数： 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。

参数	声明	数据类型	存储器区	描述
S7-300 : RD_1  S7-400 : RD_i (1 ≤ i ≤ 4)	IN_OUT	ANY	M、D、T、Z  I、Q、M、D、 T、Z	指针，指向第 i 个接收数据区： 只允许使用下列数据类型：BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME、COUNTER、TIMER。 <b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须要指定该数据块(例如： P# DB10.DBX5.0 Byte 10)。

### 错误信息

ERROR	STATUS (十进制)	解释
0	9	超限警告：较旧的接收数据被新的接收数据覆盖。
0	11	警告：正在以较低优先级处理接收数据。
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如： <ul style="list-style-type: none"> <li>没有装载连接描述(本地或远程)</li> <li>连接被中断(例如：电缆断线、CPU 关闭、或 CP 处于 STOP 模式)</li> <li>没有建立到通讯伙伴的连接</li> <li>FB 无法在 S7-400 CPU 上运行</li> <li>此外对于 S7-300： <ul style="list-style-type: none"> <li>超出并行作业/实例的最大数目</li> </ul> </li> </ul>
1	4	接收区指针 RD_i 中出错，包括数据长度或数据类型。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)
1	12	当调用 CFB 时 <ul style="list-style-type: none"> <li>指定了一个不属于 SFB"URCV"的实例 DB</li> <li>没有指定任何实例 DB，而指定了一个全局数据块。</li> <li>没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	18	<ul style="list-style-type: none"> <li>在连接 ID 中已经存在 R_ID。</li> <li>此外对于 S7-300： <ul style="list-style-type: none"> <li>CPU 在 RUN 模式(要求 CPU 或 CP 进行从 STOP 到 RUN 的切换)时，实例超载。</li> </ul> </li> </ul>
1	19	相应的 SFB/FB"USEND"正在发送数据，其速度快于 SFB/FB"URCV"能够将它们复制到接收区的速度。

ERROR	STATUS (十进制)	解释
1	20	<ul style="list-style-type: none"> <li>• S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>• H系统：在更新期间不能进行SFB的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> <li>• S7-300： <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> <li>- CPU在RUN模式(要求CPU或CP进行从STOP到RUN的切换)时，实例超载。</li> <li>- 当第一次调用时可以</li> </ul> </li> </ul>
1	27	CPU中没有用于此功能块的功能代码(仅对于S7-300)

## 数据一致性

如果遵守下列要点，则可以保证数据接收的一致性：

- S7-300：在状态参数NOR变为数值1后，必须立即使用EN\_R处的数值0重新调用FB9"URCV"。这可以保证在计算完接收到的数据之前，接收区不会被覆盖。
- 在调用块(控制输入EN\_R取值1)之前，完成评估接收区(RD\_1)。

S7-400：在状态参数NDR数值变化为1之后，在接收区(RD\_i)中有新的接收数据。新的块调用可能引起这些数据被新的接收数据覆盖。如果要防止这种情况发生，则必须调用SFB9"URCV"(例如通过周期性的块处理)(EN\_R的数值为0)，一直到完成对接收数据的处理为止。

## 22.6 通过 SFB/FB 12 "BSEND"发送分段数据

### 描述

SFB/FB 12"BSEND"向类型为"BRCV"的远程伙伴 SFB/FB 发送数据。通过该数据传送类型，通讯伙伴之间可传送的数据量多于使用已组态 S7 连接的所有其它通讯 SFB/FB 传送的数据量：

- 通过 SIMATIC-Net CP 为 S7-300 传送 32768 字节
- 通过一个集成的接口为 S7-400 和 S7-300 传送 65534 字节

要发送的数据区是分段的。各个分段单独发送给通讯伙伴。通讯伙伴在接收到最后一个分段时对此分段进行确认，该过程与相应 SFB/FB"BRCV"的调用无关。

**S7-300**：在 REQ 的上升沿处激活发送作业。在 REQ 的每个上升沿处传送参数 R\_ID、ID、SD\_1 和 LEN。在一个作业结束之后，可以给 R\_ID、ID、SD\_1 和 LEN 参数分配新的数值。为了进行分段数据的传送，必须在用户程序中周期性地调用块。

由 SD\_1 指定起始地址和要发送数据的最大长度。可以通过 LEN 来确定数据域的作业指定长度。

**S7-400 和 S7-300 通过集成接口**：在调用块之后，当在控制输入 REQ 上有上升沿时，发送作业被激活。发送用户存储区中的数据与处理用户程序是异步执行的。

由 SD\_1 指定起始地址和要发送数据的最大长度。可以通过 LEN 来确定数据域的作业指定长度。在这种情况下，LEN 替换 SD\_1 的长度区域。

参数 R\_ID 必须在相应的两个 SFB/FB 上完全相同。

如果在控制输入 R 处有上升沿，则当前数据传送将被取消。

如果传送成功完成，则通过将状态参数 DONE 的数值设置为 1 来进行指示。

如果状态参数 DONE 或 ERROR 的数值为 1，则在前一个发送处理结束之前，不能处理新的发送作业。

由于是异步数据传送，所以只有在通过调用伙伴 SFB/FB 而检索到前一个数据时，新传送才能启动。在数据被检索到之前，当调用 SFB/FB"BSEND"时，将给出状态值 7 (参见下表)。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求，在上升沿激活数据交换。
R	INPUT	BOOL	I、Q、M、D、L、常数	控制参数复位，当仍然进行数据交换时，将在上升沿处激活中止的数据交换。
ID	INPUT	WORD	M、D、常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
R_ID	INPUT	DWORD	I、Q、M、D、L、常数	寻址参数 R_ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数 使用通过 CP 441 到 S5 或辅助设备的连接，R_ID 包含了远程设备的地址信息。要了解更多信息，请参考 CP 441 的描述。
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕



参数	声明	数据类型	存储器区	描述
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
SD_1	IN_OUT	ANY	S7-300：M、D S7-400：I、Q、 M、D、T、Z	指针，指向发送区。 只允许使用下列数据类型：BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME、COUNTER、TIMER。 <b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须要指定该数据块(例如：P# DB10.DBX5.0 Byte 10)。
LEN	IN_OUT	WORD	I、Q、M、D、L	要发送的数据字段的长度(字节)。

## 错误信息

下表包含了所有 SFB/FB 12 特有的出错信息, 这些信息可以通过参数 ERROR 和 STATUS 输出表示。

ERROR	STATUS (十进制)	解释
0	11	警告： <ul style="list-style-type: none"> <li>• 由于前一个作业还没有结束，所以不能执行新作业。</li> <li>• 正在以较低优先级处理此作业。</li> </ul>
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如： <ul style="list-style-type: none"> <li>• 没有装载连接描述(本地或远程)</li> <li>• 连接被中断(例如：电缆断线、CPU 关闭、或 CP 处于 STOP 模式)</li> <li>• 没有建立到通讯伙伴的连接</li> <li>• FB 无法在 S7-400 CPU 上运行</li> <li>• 此外对于 S7-300：  <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> </ul> </li> </ul>
1	2	来自伙伴 SFB/FB 的否定应答。无法执行该函数。
1	3	R_ID 在 ID 指定的连接上未知，或接收块还未调用。
1	4	发送区指针 SD_1 中出错，包括数据长度或数据类型，或者通过 LEN 传送数值 0。
1	5	复位请求已执行。
1	6	伙伴 SFB/FB 处于 DISABLED 状态(EN_R 数值为 0)。同时还检查 BRCV 块的输入参数与 BSEND 块的一致性。
1	7	伙伴 SFB/FB 处于错误状态。 在最后一次数据传送之后，没有重新调用过接收块。
1	8	访问用户存储区中的远程对象被拒绝：用于相应 SFB/FB 13"BRCV"的目标区域太小。 相应的 SFB/FB 13"BRCV"报告 ERROR = 1、STATUS = 4 或 ERROR = 1、STATUS = 10。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)。
1	12	当调用 SFB 时 <ul style="list-style-type: none"> <li>• 指定了一个不属于 SFB 12 的实例 DB</li> <li>• 没有指定任何实例 DB，而是指定了全局数据块。</li> <li>• 没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	18	<ul style="list-style-type: none"> <li>• 在连接 ID 中已经存在 R_ID。</li> <li>• 此外对于 S7-300：  <ul style="list-style-type: none"> <li>- CPU 在 RUN 模式(要求 CPU 或 CP 进行从 STOP 到 RUN 的切换)时，实例超载。</li> </ul> </li> </ul>

ERROR	STATUS (十进制)	解释
1	20	<ul style="list-style-type: none"> <li>• S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>• H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> <li>• S7-300： <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> <li>- CPU 在 RUN 模式(要求 CPU 或 CP 进行从 STOP 到 RUN 的切换)时，实例超载。</li> <li>- 当第一次调用时可以</li> <li>- CP 中的内存瓶颈</li> </ul> </li> </ul>
1	27	CPU 中没有用于此功能块的功能代码(仅对于 S7-300)

### 数据一致性

为确保数据一致性，只能在当前发送操作结束之后，重新向发送区 SD\_i 写入数据。状态参数 DONE 的数值变为 1 时就是这种情形。

## 22.7 通过 SFB/FB 13 "BRCV"接收分段数据

### 描述

SFB/FB 13"BRCV"接收来自类型为"BSEND"的远程伙伴 SFB/FB 的数据。在收到每个数据段后，向伙伴 SFB/FB 发送一个确认帧，同时更新 LEN 参数。

在块调用完毕，并且在控制输入 EN\_R 数值为 1 之后，块准备接收数据。可以通过 EN\_R=0 来取消一个已激活的作业。

由 RD\_1 指定起始地址和接收区的最大长度。由 LEN 指示已接收数据域的长度。

**S7-300** : 在 EN\_R 的每个上升沿处应用参数 R\_ID、ID 和 RD\_1。在每个作业结束之后，可以给 R\_ID、ID 和 RD\_1 参数分配新数值。为了进行分段数据的传送，必须在用户程序中周期性地调用块。

**S7-400 和 S7-300 通过集成接口** : 从用户存储区中接收数据与处理用户程序是异步执行的。

参数 R\_ID 必须在相应的两个 SFB/FB 上完全相同。

通过状态参数 NDR 的数值为 1 来指示所有数据段的无错接收。接收到的数据保持不变，直到通过 EN\_R=1 来重新调用 SFB/FB 13 为止。

如果在数据的异步接收期间调用块，则将引发一个警告，该警告通过 STATUS 参数输出；如果当控制输入 EN\_R 数值为 0 时进行调用，则接收将被终止，并且 SFB/FB 将返回到它的初始状态。

参数	声明	数据类型	存储器区	描述
EN_R	INPUT	BOOL	I、Q、M、D、L、常数	经激活可进行接收的控制参数在输入置位时发出信号，通知该对象准备接收。
ID	INPUT	WORD	M、D、常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
R_ID	INPUT	DWORD	I、Q、M、D、L、常数	寻址参数 R_ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数 使用通过 CP 441 到 S5 或辅助设备的连接，R_ID 包含了远程设备的地址信息。要了解更多信息，请参考 CP 441 的描述。
NDR	OUTPUT	BOOL	I、Q、M、D、L	NDR 状态参数： 0: 作业还没有启动，或仍然处于激活状态 1: 作业已成功完成。
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。

参数	声明	数据类型	存储器区	描述
RD_1	IN_OUT	ANY	S7-300 : M、D S7-400 : I、Q、 M、D、T、C	指针，指向接收区。长度信息指定要接收块的最大长度。 仅允许使用下列数据类型：BOOL (不允许使用：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME、COUNTER、TIMER。 <b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须要指定该数据块(例如：P# DB10.DBX5.0 Byte 10)。
LEN	IN_OUT	WORD	I、Q、M、D、L	已接收的数据长度，以字节为单位。

## 错误信息

下表包含了所有 SFB/FB 13 特有的出错信息，这些信息可以通过参数 ERROR 和 STATUS 输出表示。

ERROR	STATUS (十进制)	解释
0	11	警告：正在以较低优先级处理接收数据。
0	17	警告：块异步接收数据。LEN 参数显示已接收的数据量，以字节为单位。
0	25	S7-300：通讯已触发。作业正在处理。
1	1	通讯故障，例如： <ul style="list-style-type: none"> <li>没有装载连接描述(本地或远程)</li> <li>连接被中断(例如：电缆断线、CPU 关闭、或 CP 处于 STOP 模式)</li> <li>没有建立到通讯伙伴的连接</li> <li>FB 无法在 S7-400 CPU 上运行</li> <li>此外对于 S7-300： <ul style="list-style-type: none"> <li>超出并行作业/实例的最大数目</li> </ul> </li> </ul>
1	2	函数无法执行(协议错误)。
1	4	接收区指针 RD_1 中出错，包括数据长度或数据类型。发送数据域长于接收区。
1	5	复位已接收的请求，传送未完成。
1	8	相应 SFB/FB 12"BSEND"中发生访问出错。在发送完最后一个有效数据段之后，报告 ERROR = 1 和 STATUS = 4 或 ERROR = 1 和 STATUS = 10。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)。
1	12	当调用 SFB 时 <ul style="list-style-type: none"> <li>指定了一个不属于 SFB 13 的实例 DB</li> <li>没有指定任何实例 DB，而是指定了全局数据块。</li> <li>没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	18	<ul style="list-style-type: none"> <li>在连接 ID 中已经存在 R_ID。</li> <li>此外对于 S7-300： <ul style="list-style-type: none"> <li>CPU 在 RUN 模式(要求 CPU 或 CP 进行从 STOP 到 RUN 的切换)时，实例超载。</li> </ul> </li> </ul>

ERROR	STATUS (十进制)	解释
1	20	<ul style="list-style-type: none"> <li>• S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>• H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> <li>• S7-300： <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> <li>- CPU 在 RUN 模式(要求 CPU 或 CP 进行从 STOP 到 RUN 的切换)时，实例超载。</li> <li>- 当第一次调用时可以</li> <li>- CP 中的内存瓶颈</li> </ul> </li> </ul>
1	27	CPU 中没有用于此功能块的功能代码(仅对于 S7-300)

### 数据一致性

如果遵守下列要点，则可以保证数据接收的一致性：

- 在再次调用块(控制输入 EN\_R 数值为 1)之前，完成计算上次使用的接收区(RD\_1)的数值。

### 接收数据的特例(只用于 S7-400)

如果在相应发送块发送第一个数据段之前，准备接受数据的带有 BRCV 块的接收 CPU (即，已经在 EN\_R = 1 时进行了块调用)切换到 STOP 模式，则将发生下列事件：

- 接收 CPU 切换到 STOP 模式之后的第一个作业内的数据全部输入到接收区内。
- 伙伴 SFB"BSEND"接收到一个肯定应答。
- 处于 STOP 模式的接收 CPU 不再能接受任何附加的 BSEND 作业。
- 只要 CPU 保持在 STOP 模式，NDR 和 LEN 数值就都为 0。

为了防止丢失关于接收数据的信息，必须在接收 CPU 上执行热启动，并以 EN\_R = 1 来调用 SFB 13"BRCV"。

## 22.8 通过 SFB/FB 15 "PUT"向远程 CPU 写入数据

### 描述

通过使用 SFB/FB 15"PUT"，可以将数据写入到远程 CPU。

**S7-300**：在 REQ 的上升沿处发送数据。在 REQ 的每个上升沿处传送参数 ID、ADDR\_1 和 SD\_1。在每个作业结束之后，可以给 ID、ADDR\_1 和 SD\_1 参数分配新数值。

**S7-400**：在控制输入 REQ 的上升沿处启动 SFB。指向即将写入的区域和数据(SD\_i)的指针(ADDR\_i)将被发送到伙伴 CPU。

远程伙伴将所需要的数据保存在随数据一起提供的地址下面，并返回一个执行确认。

必须要确保通过参数 ADDR\_i 和 SD\_i 定义的区域在编号、长度和数据类型方面相互匹配。

如果没有发生错误，则在下一次 SFB/FB 调用时通过状态参数 DONE 设为值 1 来指示。

只有在最后一个作业完成之后，才能再次激活写作业。

远程 CPU 可以处于 RUN 或 STOP 模式。

如果正在写入数据时发生访问故障，或如果执行检查过程中出错，则出错和警告信息将通过 ERROR 和 STATUS 输出表示。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求，在上升沿激活数据交换。
ID	INPUT	WORD	M、D、常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息。 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
S7-300 : ADDR_1  S7-400 : ADDR_i (1 ≤ i ≤ 4)	IN OUT	ANY	M、D  I、Q、M、D、 T、C	指针，指向伙伴 CPU 中要用于写入数据的区域。 <b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须要指定该数据块(例如：P# DB10.DBX5.0 Byte 10)

参数	声明	数据类型	存储器区	描述
S7-300 : SD_1  S7-400 : SD_i (1 ≤ i ≤ 4)	IN_OUT	ANY	S7-300 : M、D S7-400 I、Q、 M、D、T、C	<p>指针，指向本地 CPU 中包含了要发送的数据的区域。</p> <p>只允许使用下列数据类型：</p> <p>BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、COUNTER、TIMER。</p> <p><b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须要指定该数据块(例如：P# DB10.DBX5.0 Byte 10)。</p>



## 错误信息

下表包含了所有 SFB/FB 15 特定的出错信息, 这些信息可以通过参数 ERROR 和 STATUS 输出表示。

ERROR	STATUS (十进制)	解释
0	11	警告： <ul style="list-style-type: none"> <li>• 由于前一个作业还没有结束，所以不能执行新作业。</li> <li>• 正在以较低优先级处理此作业。</li> </ul>
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如： <ul style="list-style-type: none"> <li>• 没有装载连接描述(本地或远程)</li> <li>• 连接被中断(例如：电缆断线、CPU 关闭、或 CP 处于 STOP 模式)</li> <li>• 没有建立到通讯伙伴的连接</li> <li>• FB 无法在 S7-400 CPU 上运行</li> <li>• 此外对于 S7-300： <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> </ul> </li> </ul>
1	2	来自伙伴设备的否定应答。无法执行该函数。
1	4	发送区指针 SD_i 中出错，包括数据长度或数据类型。
1	8	在伙伴 CPU 上发生访问出错。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)
1	12	当调用 SFB 时， <ul style="list-style-type: none"> <li>• 指定了一个不属于 SFB 15 的实例 DB。</li> <li>• 没有指定任何实例 DB，而是指定了一个共享数据块。</li> <li>• 没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	20	<ul style="list-style-type: none"> <li>• S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>• H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> <li>• S7-300： <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> <li>- CPU 在 RUN 模式(要求 CPU 或 CP 进行从 STOP 到 RUN 的切换)时，实例超载。</li> <li>- 当第一次调用时可以</li> </ul> </li> </ul>
1	27	CPU 中没有用于此功能块的功能代码(仅对于 S7-300)

### S7-300 的数据一致性

为了确保数据一致性,在当前发送处理结束之前,不能重新向发送区 SD\_I 写入数据。状态参数 DONE 数值为 1 时便是这种情况。

### S7-400 和 S7-300 通过集成接口的数据一致性 :

当激活一个发送操作时(在 REQ 的上升沿),从用户程序中复制要从发送区 SD\_i 发送的数据。在块调用之后,可以在不破坏当前发送数据的前提下向这些区域写入数据。

---

#### 注意

只有当 DONE 状态参数数值为 1 时,发送操作才完成。

---

## 22.9 通过 SFB/FB 14 "GET"从远程 CPU 中读取数据

### 描述

可以通过 SFB/FB 14"GET"，从远程 CPU 中读取数据。

**S7-300**：在 REQ 的上升沿处读取数据。在 REQ 的每个上升沿处传送参数 ID、ADDR\_1 和 RD\_1。在每个作业结束之后，可以分配新数值给 ID、ADDR\_1 和 RD\_1 参数。

**S7-400**：在控制输入 REQ 的上升沿处启动 SFB。指向要读取区域的相关指针(ADDR\_i)将被发送到伙伴 CPU。

远程伙伴返回此数据。

在下一个 SFB/FB 调用处，已接收的数据被复制到组态的接收区(RD\_i)中。

必须要确保通过参数 ADDR\_i 和 RD\_i 定义的区域在长度和数据类型方面要相互匹配。

通过状态参数 NDR 数值为 1 来指示此作业已完成。

只有在前一个作业已经完成之后，才能重新激活读作业。

远程 CPU 可以处于 RUN 或 STOP 工作状态。

如果正在读取数据时发生访问故障，或如果数据类型检查过程中出错，则出错和警告信息将通过 ERROR 和 STATUS 输出表示。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求，在上升沿激活数据交换。
ID	INPUT	WORD	IM、D、常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
NDR	OUTPUT	BOOL	I、Q、M、D、L	NDR 状态参数： 0: 作业还未启动或仍然处于激活状态。 1: 作业已成功完成。
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0  STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供 详细信息。  ERROR = 1  产生一个错误。STATUS 给出了关于 错误类型的详细信息。关于错误类型 的详细信息。
S7-300 : ADDR_1  S7-400 : ADDR_i (1 ≤ i ≤ 4)	IN_OUT	ANY	S7-300 : M、D  S7-400 : I、Q、M、D、 T、C	指针，指向伙伴 CPU 中要被读取的区域。 <b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须 要指定该数据块(例如：P# DB10.DBX5.0 Byte 10)

参数	声明	数据类型	存储器区	描述
S7-300 : RD_1	IN_OUT	ANY	S7-300 : M、D	指针，指向本地 CPU 中用于输入读取数据的区域。
S7-400 : RD_i (1 ≤ i ≤ 4)			S7-400 : I、Q、M、D、 T、C	只允许使用下列数据类型： BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、COUNTER、TIMER。 <b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须要指定该数据块(例如：P# DB10.DBX5.0 Byte 10)。

## 错误信息

下表包含了所有 SFB/FB 14 特有的出错信息, 这些信息可以通过参数 ERROR 和 STATUS 输出表示。

ERROR	STATUS (十进制)	解释
0	11	警告： <ul style="list-style-type: none"> <li>• 由于前一个作业还没有结束，所以不能执行新作业。</li> <li>• 正在以较低优先级处理此作业。</li> </ul>
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如： <ul style="list-style-type: none"> <li>• 没有装载连接描述(本地或远程)</li> <li>• 连接被中断(例如：电缆断线、CPU 关闭、或 CP 处于 STOP 模式)</li> <li>• 没有建立到通讯伙伴的连接</li> <li>• FB 无法在 S7-400 CPU 上运行</li> <li>• 此外对于 S7-300： <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> </ul> </li> </ul>
1	2	来自伙伴设备的否定应答。无法执行该函数。
1	4	接收区指针 RD_i 中出错，包括数据长度或数据类型。
1	8	在伙伴 CPU 上发生访问出错。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)
1	12	当调用 SFB 时， <ul style="list-style-type: none"> <li>• 指定了一个不属于 SFB 14 的实例 DB。</li> <li>• 没有指定任何实例 DB，而是指定了一个共享数据块。</li> <li>• 没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	20	<ul style="list-style-type: none"> <li>• S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>• H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> <li>• S7-300： <ul style="list-style-type: none"> <li>- 超出并行作业/实例的最大数目</li> <li>- CPU 在 RUN 模式(要求 CPU 或 CP 进行从 STOP 到 RUN 的切换)时，实例超载。</li> <li>- 当第一次调用时可以</li> </ul> </li> </ul>
1	27	CPU 中没有用于此功能块的功能代码(仅对于 S7-300)

## 数据一致性

如果遵守了下面的要点，则能保证数据接收处于一致性状态：

在启动其他作业之前，评估当前正在使用的接收区 RD\_i 部分。

## 22.10 通过 SFB 16 "PRINT"将数据发送到打印机

### 描述

SFB 16"PRINT"将数据和格式指令发送到远程打印机，例如，通过 CP 441。

当在控制输入 REQ 上有上升沿时，格式描述(FORMAT)和数据(SD\_i)被发送到通过 ID 和 PRN\_NR 选择的打印机上。

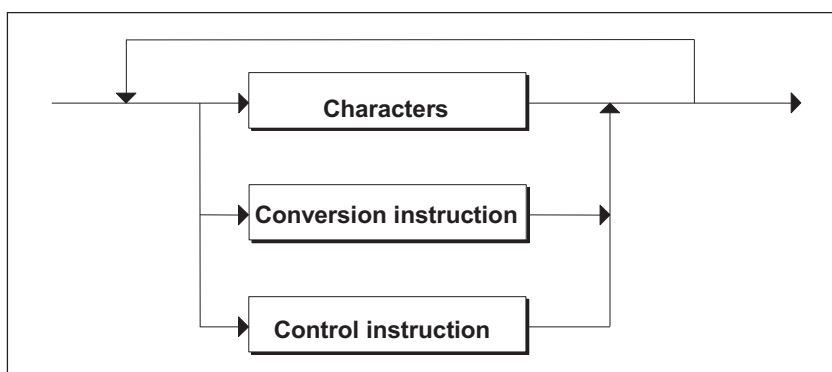
如果并未使用所有四个发送区，则必须确保用 SD\_1 参数描述第一个区域，用 SD\_2 参数描述第二个区域(如果存在)，用 SD\_3 描述第三个区域(如果存在)。

通过 DONE 状态参数来指示作业的成功执行，通过 ERROR 和 STATUS 参数来指示发生的错误。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求，在上升沿激活数据交换。
ID	INPUT	WORD	M、D、常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕
ERROR STATE	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 和 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
PRN_NR	IN_OUT	BYTE	I、Q、M、D、L	打印机数目
格式	IN_OUT	STRING	I、Q、M、D、L	格式描述
SD_i (1 ≤ i ≤ 4)	IN_OUT	ANY	M、D、T、C	指针，指向"第 i 个"发送数据区。 只允许使用下列数据类型：BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME、COUNTER、TIMER。 <b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须要指定该数据块(例如：P# DB10.DBX5.0 Byte 10)。

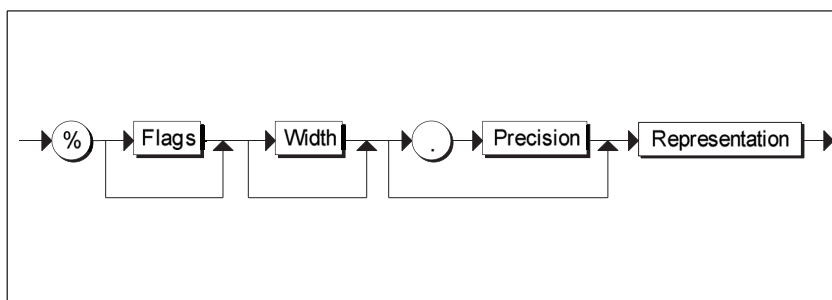
## In\_out 参数 FORMAT

FORMAT 字符串包含了可打印的字符和格式元素。它具有下列结构：



对于每个要打印的发送区(SD\_1 到 SD\_4)，在 FORMAT 中必须要有一个转换指令。转换指令依据发送区格式化的顺序，依次应用于各个发送区(SD\_i)。字符和指令可以按照任意顺序一个接一个地排列。

- 字符  
可以使用下列字符：
  - 所有可打印的字符
  - \$\$ (美元字符)、\$' (单引号)、\$L 和\$I (换行)、\$P 和\$p (页面)、\$R 和\$r (回车)、\$T 和\$t (制表键)



转换指令的语法图

转换指令的元素	含义
标记	<ul style="list-style-type: none"> <li>• 无：右对齐输出</li> <li>• -: 左对齐输出</li> </ul>
宽度	<ul style="list-style-type: none"> <li>• 无：以标准表达式输出</li> <li>• n：精确地输出 n 个字符。如果输出是右对齐，则字符前面可能会有空格，而对于左对齐，则在字符后面输出空格。</li> </ul>
精度	<p>精度只与表达式 A、D、F 和 R 相关(参见下表)。</p> <ul style="list-style-type: none"> <li>• 无：以标准表达式输出</li> <li>• 0：在 F 和 R 表达式中没有小数点或小数位输出 F 和 R 表达式</li> <li>• n：- 对于 F 和 R：小数点和 n 个小数位的输出 n 个小数位</li> <li>• - 对于 A 和 D (日期)：年的位数：可以使用的数值为 2 和 4。</li> </ul>

转换指令的元素	含义
表示法	下表包含了： <ul style="list-style-type: none"> <li>• 可以使用的表达式</li> <li>• 可以用于每个表达式的数据类型</li> <li>• 用于每个表达式的标准格式(如果没有在 FORMAT 参数中指定宽度和精度，则打印输出使用标准表达式)，以及它们的最大长度</li> </ul>

下表给出了在 FORMAT 参数的转换指令中可以使用的表达式模式。

表示法	可用的数据类型	实例	长度	注释
A、a	DATE	25.07.1996	10	-
	DWORD			
C、c	CHAR	K	1	-
	BYTE	M	1	
	WORD	KL	2	
	DWORD	KLMN	4	
	CHAR 的 ARRAY ARRAY of BYTE	KLMNOP	字符数	
D、d	DATE	1996-07-25	10	-
	DWORD			
F、f	REAL	0.345678	8	-
	DWORD			
H、h	所有数据类型，包括 ARRAY of BYTE	根据数据类型	根据数据类型	十六进制表达式
I、i	INT	- 32 768	最大 6	-
	WORD	- 2 147 483 648	最大 11	
N、n	WORD	文本输出	-	相应发送区 SD_i 包含了一个指向要打印文本的引用(编号)。文本在创建可打印字符串的模块(例如，CP 441)上。如果在指定的编号下找不到任何文本，则输出 "*****"。
R、r	REAL	0.12E-04	8	-
	DWORD			
S、s	STRING	文本输出		-
T、t	TIME	2d_3h_10m_5s_ 250ms	最大 21	如果出错，则输出 "*****"。



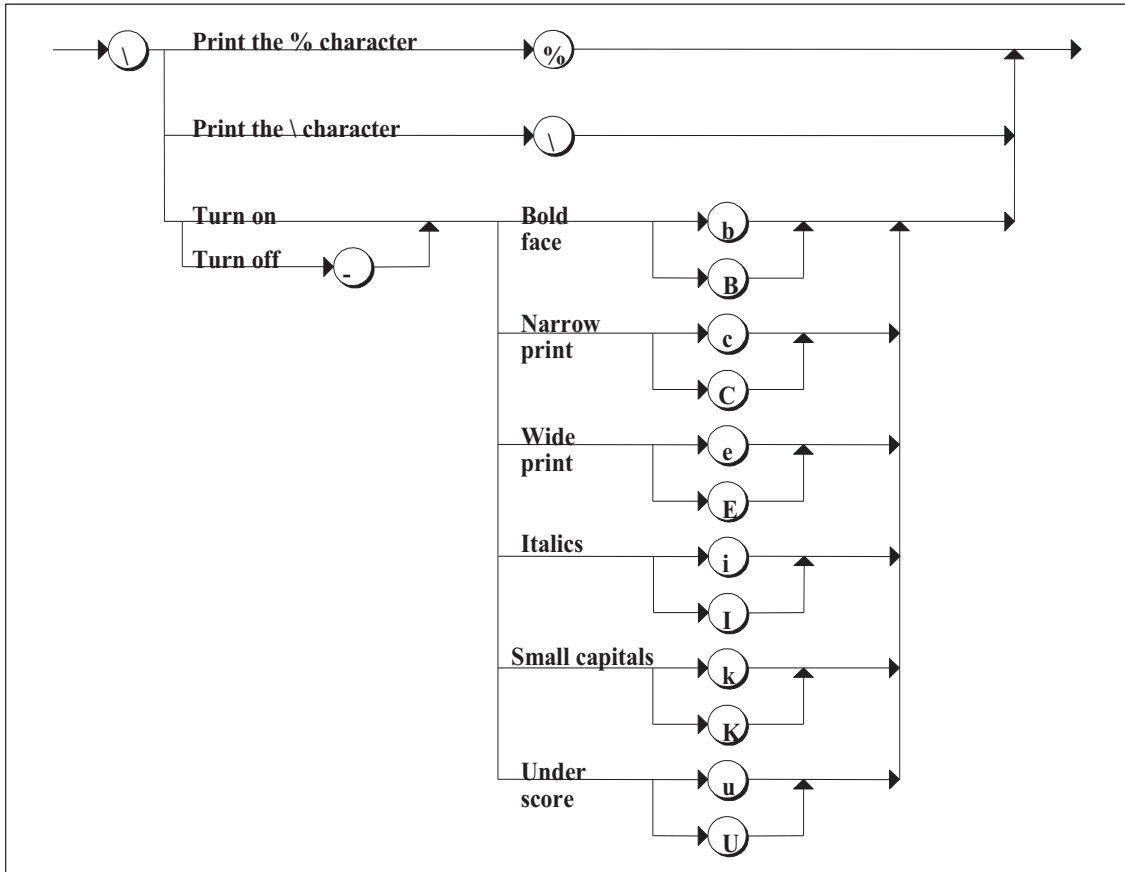
表示法	可用的数据类型	实例	长度	注释
	DWORD			
U、u	BYTE	255	最大 3	-
	WORD	65 535	最大 5	
	DWORD	4 294 967 295	最大 10	
X、x	BOOL	1	1	-
	BYTE	101 ..	8	
	WORD	101 ..	16	
	DWORD	101 ..	32	
Z、z	TIME_OF_DAY (TOD)	15:38:59.874	12	-

此表中的某些数据，是为表达式指定的最大长度，所以实际长度可以更短一些。

#### 注意

对于数据类型 C 和 S，以下几点取决于所使用的打印机：

- 可以打印哪些字符
  - 对于不能打印的字符，打印机将打印什么，除非打印机驱动程序对于这些字符有转换表。
- 
- 控制指令  
通过使用控制指令，可执行下列任务：
    - 打印字符%和\
    - 改变打印机设置。



控制指令的语法图

如果尝试进行某些工作，例如，禁止一种还没有启用的字体，或执行打印机无法识别的一个函数，则控制指令将被忽略。下表包含了使用 FORMAT 输入/输出参数时可能发生的错误。

错误	打印机输出
转换指令无法执行	根据缺省表达式的(最大)长度或指定宽度输出*字符。
指定的宽度太小	在表达式 A、C、D、N、S、T 和 Z 中，打印和所选择的宽度指定的字符数一样多的字符。 对于所有其它表达式，打印*字符来补足指定的宽度。
转换指令太多	没有发送区指针 SD_i 的转换指令将被忽略。
转换指令太少	不打印没有转换指令的发送区。
未定义或不支持的转换指令	打印输出*****。
不完整的转换指令	打印输出*****。
未定义或不支持的转换指令	不符合上图中显示的语法的控制指令将被忽略。

## 错误信息

下表包含了所有 SFB 16"PRINT"特有的出错信息，这些信息可以通过参数 ERROR 和 STATUS 打印输出。

ERROR	STATUS (十进制)	解释
0	11	警告： <ul style="list-style-type: none"> <li>由于前一个作业还没有结束，所以不能执行新作业。</li> <li>正在以较低优先级处理此作业。</li> </ul>
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如 <ul style="list-style-type: none"> <li>没有装载连接描述(本地或远程)</li> <li>连接中断(例如，电缆断线、CPU 关闭、CP 处于 STOP 模式)</li> </ul>
1	2	来自打印机的否定应答。无法执行该函数。
1	3	在 ID 指定的通讯链接上不能识别 PRN_NR。
1	4	在 FORMAT 输入/输出参数或在发送区指针 SD_j 中，在数据长度或数据类型方面出错。
1	6	远程打印机处于 OFFLINE 状态。
1	7	远程打印机状态不正确(例如，缺纸)。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)。
1	13	在 FORMAT 输入/输出参数中出错
1	20	<ul style="list-style-type: none"> <li>S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> </ul>

## 可传送的数据量

可以传送到远程打印机的数据量不得超出最大长度。

此最大数据长度按照下面的公式来计算：

最大长度 = 420 - 格式

格式是 FORMAT 参数的当前长度，以字节为单位。要打印的数据可以分布在一个或者多个发送区。

## 22.11 通过 SFB 19 "START"在远程设备上启动暖启动或冷启动

### 描述

如果在 SFB 19"START"的控制输入 REQ 上有上升沿，则这将激活由 ID 寻址的远程设备上的暖启动或冷启动。如果远程系统是一个容错系统，则重新启动请求的结果就取决于参数 PI\_NAME：启动请求或者仅对此系统中的一个 CPU 有效，或者对此系统中的所有 CPU 有效。如果远程设备是一个 CPU，则必须满足下列条件：

- CPU 必须处于 STOP 模式。
- CPU 的钥匙开关必须设置为"RUN"或"RUN-P"。

一旦完成暖启动或冷启动，设备切换到 RUN 模式，并发送一个肯定执行应答。接收到肯定确认之后，状态参数 DONE 置位为 1。如果发生任何错误，则通过状态参数 ERROR 和 STATUS 来指示这些出错。

只有在最近一次完全重新启动完成之后，才能在相同的远程设备上再次激活暖启动或冷启动。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求，在上升沿激活 SFB
ID	INPUT	WORD	I、Q、M、D、 常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 和 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息  ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
PI_NAME	IN_OUT	ANY	I、Q、M、D、 T、C	指针，指向存储要启动的程序的名称(ASCII 代码)的存储区域。此名称所包含的字符不能超过 32 个。 对于 S7 PLC，名称必须是 P_PROGRAM。 对于 H 系统，可以使用下列名称： <ul style="list-style-type: none"> <li>• P_PROGRAM (启动作业对 H 系统中的所有 CPU 都有效)。</li> <li>• P_PROG_0 (启动作业对 H 系统中机架 0 中的 CPU 有效)。</li> <li>• P_PROG_1 (启动作业对 H 系统中机架 1 中的 CPU 有效)。</li> </ul>

参数	声明	数据类型	存储器区	描述
ARG	IN_OUT	ANY	I、Q、M、D、 T、C	<p>执行参数。</p> <ul style="list-style-type: none"> <li>如果没有给 ARG 分配数值，则在远程设备上执行暖启动。</li> <li>如果分配数值"C"，则在远程设备上执行冷启动(如果远程设备支持这种类型的启动)。</li> </ul>
IO_STATE	IN_OUT	BYTE	I、Q、M、D、L	当前不相关。如果通讯伙伴是 S7 可编程控制器，则不给这个参数分配任何数值。

## 错误信息

下表包含了所有 SFB 19 特有的出错信息，这些信息可以通过参数 ERROR 和 STATUS 输出表示。

ERROR	STATUS (十进制)	解释
0	11	警告： <ul style="list-style-type: none"> <li>由于前一个作业还没有结束，所以不能执行新作业。</li> <li>正在以较低优先级处理此作业。</li> </ul>
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如： <ul style="list-style-type: none"> <li>没有装载连接描述(本地或远程)</li> <li>连接中断(例如，电缆断线、CPU 关闭、CP 处于 STOP 模式)</li> </ul>
1	2	来自伙伴设备的否定应答。无法执行该函数。
1	3	无法识别在 PI_NAME 中输入的程序名称。
1	4	指针 PI_NAME 或 ARG 中出错，包括数据长度或数据类型。
1	7	无法在伙伴设备上执行完全重启动。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)
1	12	当调用 SFB 时， <ul style="list-style-type: none"> <li>指定了一个不属于 SFB 19 的实例 DB。</li> <li>没有指定任何实例 DB，而是指定了一个共享数据块。</li> <li>没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	20	<ul style="list-style-type: none"> <li>S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> </ul>

## 22.12 通过 SFB 20 "STOP"将远程设备切换到 STOP 状态

### 描述

如果在 SFB 20"STOP"的控制输入 REQ 上有上升沿，则将由 ID 寻址的远程设备切换到 STOP 模式。当设备处于 RUN、HOLD 或 STARTUP 模式时，可以进行模式切换。

如果远程系统是一个容错系统，则重新启动请求的结果就取决于参数 PI\_NAME：启动请求或者仅对此系统中的一个 CPU 有效，或者对此系统中的所有 CPU 有效。

通过状态参数 DONE 的数值为 1 来指示作业的成功执行。如果发生任何错误，则通过状态参数 ERROR 和 STATUS 来指示这些出错。

只有在前一个 SFB 20 调用结束之后，才能再次在相同远程设备上启动模式切换。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求，在上升沿激活 SFB
ID	INPUT	WORD	I、Q、M、D、 常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 和 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息。 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
PI_NAME	IN_OUT	ANY	I、Q、M、D	指针，指向存储要启动的程序的名称(ASCII 代码)的存储区域。此名称所包含的字符不能超过 32 个。 对于 S7 PLC，名称必须是 P_PROGRAM。 对于 H 系统，可以使用下列名称： <ul style="list-style-type: none"> <li>• P_PROGRAM (启动作业对 H 系统中的所有 CPU 都有效)。</li> <li>• P_PROGR_0 (启动作业对 H 系统中机架 0 中的 CPU 有效)。</li> <li>• P_PROGR_1 (启动作业对 H 系统中机架 1 中的 CPU 有效)。</li> </ul>
IO_STATE	IN_OUT	BYTE	I、Q、M、D、L	当前不相关。如果通讯伙伴是 S7 可编程控制器，则不给这个参数分配任何数值。

## 错误信息

下表包含了所有 SFB 20 特有的出错信息，这些信息可以通过参数 ERROR 和 STATUS 输出表示。

ERROR	STATUS (十进制)	解释
0	11	警告： <ul style="list-style-type: none"> <li>• 由于前一个作业还没有结束，所以不能执行新作业。</li> <li>• 正在以较低优先级处理此作业。</li> </ul>
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如 <ul style="list-style-type: none"> <li>• 没有装载连接描述(本地或远程)</li> <li>• 连接中断(例如，电缆断线、CPU 关闭、CP 处于 STOP 模式)</li> </ul>
1	2	来自伙伴设备的否定应答。无法执行该函数。
1	3	无法识别在 PI_NAME 中输入的程序名称。
1	4	指针 PI_NAME 中出错，包括数据长度或数据类型。
1	7	伙伴设备已经处于 STOP 状态。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)
1	12	当调用 SFB 时， <ul style="list-style-type: none"> <li>• 指定了一个不属于 SFB 20 的实例 DB。</li> <li>• 没有指定任何实例 DB，而是指定了一个共享数据块。</li> <li>• 没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	20	<ul style="list-style-type: none"> <li>• S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>• H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> </ul>



## 22.13 通过 SFB 21 "RESUME"在远程设备上启动热启动

### 描述

如果在 SFB 21"RESUME"的控制输入 REQ 上有上升沿，则将在通过 ID 选择的远程设备上激活一次热启动。

如果远程设备是一个 CPU，则必须满足下列条件：

- CPU 必须处于 STOP 模式。
- CPU 的钥匙开关必须设置为"RUN"或"RUN-P"。
- 当通过 STEP 7 创建组态时，允许进行手动热启动。
- 必须没有阻止热启动的其他条件。

一旦完成热启动，设备将切换到 RUN 模式，并发送一个肯定执行应答。接收到肯定确认之后，状态参数 DONE 置位为 1。所发生的错误由状态参数 ERROR 和 STATUS 指出。

只有在前一个热启动结束之后，才能在同一个远程设备上重新激活重启动。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求，在上升沿激活 SFB
ID	INPUT	WORD	I、Q、M、D、 常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 和 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
PI_NAME	IN_OUT	ANY	I、Q、M、D	指针，指向存储要启动的程序的名称(ASCII 代码)的存储区域。此名称所包含的字符不能超过 32 个。对于 S7 PLC，名称必须是 P_PROGRAM。
ARG	IN_OUT	ANY	I、Q、M、D、 T、C	执行参数。当前不相关。如果通讯伙伴是 S7 可编程控制器，则不给这个参数分配任何数值。
IO_STATE	IN_OUT	BYTE	I、Q、M、D、L	当前不相关。如果通讯伙伴是 S7 可编程控制器，则不给这个参数分配任何数值。

## 错误信息

下表包含了所有 SFB 21 特有的出错信息，这些信息可以通过参数 ERROR 和 STATUS 输出表示。

ERROR	STATUS (十进制)	解释
0	11	警告： <ul style="list-style-type: none"> <li>• 由于前一个作业还没有结束，所以不能执行新作业。</li> <li>• 正在以较低优先级处理此作业。</li> </ul>
0	25	通讯已经启动。作业正在处理。
1	1	<ul style="list-style-type: none"> <li>• 通讯故障，例如，没有装载连接描述(本地或远程)</li> <li>• 连接中断(例如，电缆断线、CPU 关闭、CP 处于 STOP 模式)</li> </ul>
1	2	来自伙伴设备的否定应答。无法执行该函数。
1	3	无法识别在 PI_NAME 中输入的程序名称。
1	4	指针 PI_NAME 或 ARG 中出错，包括数据长度或数据类型。
1	7	不能执行热启动
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)
1	12	当调用 SFB 时， <ul style="list-style-type: none"> <li>• 指定了一个不属于 SFB 21 的实例 DB。</li> <li>• 没有指定任何实例 DB，而是指定了一个共享数据块。</li> <li>• 没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	20	<ul style="list-style-type: none"> <li>• S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>• H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> </ul>

## 22.14 通过 SFB 22"STATUS"查询远程伙伴的状态

### 描述

通过使用 SFB 22"STATUS", 可以查询远程通讯伙伴的状态。

如果在控制输入 REQ 上有上升沿, 则将作业发送到远程伙伴。判断回复, 以确定是否产生了故障。如果没有产生任何错误, 则通过下一个 SFB 调用将接收到的状态复制到变量 PHYS、LOG 和 LOCAL。通过状态参数 NDR 的数值为 1 来指示作业完成。

只有最后一次查询结束之后, 才能再次查询同一个通讯伙伴的状态。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求, 在上升沿激活 SFB
ID	INPUT	WORD	I、Q、M、D、 常数	寻址参数 ID, 请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
NDR	OUTPUT	BOOL	I、Q、M、D、L	NDR 状态参数: 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数, 出错显示: ERROR=0 和 STATUS 的数值: 0000H: 既不是警告也不是出错 不等于 0000H: 警告, STATUS 提供详细信息。 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
PHYS	IN_OUT	ANY	I、Q、M、D	物理状态(最小长度: 一个字节)。 可能的值: • 10H 执行功能 • 13H 服务请求
LOG	IN_OUT	ANY	I、Q、M、D	逻辑状态(最小长度: 一个字节)。 可能的数值: • 00H 允许状态改变
LOCAL	IN_OUT	ANY	I、Q、M、D	如果伙伴设备是一个 S7 CPU 时的状态 (最小长度: 两个字节)

### 输入/输出参数 LOCAL

如果通讯伙伴是一个 S7 CPU, 则输入/输出参数 LOCAL 包含它的当前状态。第一个字节保留, 第二个字节包含状态的 ID。

工作模式	相应标识符
STOP	00H
暖启动	01H
RUN	02H
热启动	03H
HOLD	04H
冷启动	06H
RUN_R	09H
LINK-UP	0BH
UPDATE	0CH

### 错误信息

下表包含了所有 SFB 22 特有的出错信息，这些信息可以通过参数 ERROR 和 STATUS 输出表示。

ERROR	STATUS (十进制)	解释
0	11	警告： <ul style="list-style-type: none"> <li>由于前一个作业还没有结束，所以不能执行新作业。</li> <li>正在以较低优先级处理此作业。</li> </ul>
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如 <ul style="list-style-type: none"> <li>没有装载连接描述(本地或远程)</li> <li>连接中断(例如，电缆断线、CPU 关闭、CP 处于 STOP 模式)</li> </ul>
1	2	来自伙伴设备的否定应答。无法执行该函数。
1	4	PHYS、LOG 或 LOCAL 中出错，包括数据长度或数据类型。
1	8	访问远程对象被拒绝。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)。
1	12	当调用 SFB 时， <ul style="list-style-type: none"> <li>指定了一个不属于 SFB 22 的实例 DB。</li> <li>没有指定任何实例 DB，而是指定了一个共享数据块。</li> <li>没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	20	<ul style="list-style-type: none"> <li>S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> </ul>

## 22.15 通过 SFB 23 "USTATUS"接收远程设备的状态改变

### 描述

SFB 23"USTATUS"接收远程通讯伙伴的设备状态改变。如果在 STEP 7 中进行了相应组态，当发生状态改变时，通讯伙伴主动发送它的状态。

当调用 CFB 并且有来自通讯伙伴的数据帧时，如果控制输入 EN\_R 数值为 1，则在下一次调用 SFB 时，将状态信息输入到变量 PHYS、LOG 和 LOCAL 中。通过状态参数 NDR 的数值为 1 来指示作业完成。

必须在 USTATUS 所使用的连接上允许进行工作状态消息传送。

### 注意

每个连接只能使用一个 SFB 23 实例。

参数	声明	数据类型	存储器区	描述
EN_R	INPUT	BOOL	I、Q、M、D、L	经激活可进行接收的控制参数在输入置位时发出信号，通知该对象准备接收。
ID	INPUT	WORD	I、Q、M、D、 常数	寻址参数 ID，请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公共参数
NDR	OUTPUT	BOOL	I、Q、M、D、L	NDR 状态参数： 0: 作业还未启动或仍然在运行 1: 作业已经无错地执行完毕
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 和 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息。 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
PHYS	IN_OUT	ANY	I、Q、M、D	物理状态(最小长度：一个字节)。 可能的值： • 10H 执行功能 • 13H 服务请求
LOG	IN_OUT	ANY	I、Q、M、D	逻辑状态(最小长度：一个字节) 可能的数值： • 00H 允许状态改变
LOCAL	IN_OUT	ANY	I、Q、M、D	如果伙伴设备是一个 S7 CPU 时的状态 (最小长度：一个字节)

## 输入/输出参数 LOCAL

如果通讯伙伴是一个 S7 CPU，则输入/输出参数 LOCAL 包含它的当前状态。第一个字节保留，第二个字节包含状态的 ID。

工作模式	相应标识符
STOP	00H
暖启动	01H
RUN	02H
热启动	03H
HOLD	04H
冷启动	06H
RUN (H 系统状态：冗余)	09H
LINK-UP	0BH
UPDATE	0CH

## 错误信息

下表包含了所有 SFB 23 特有的出错信息，这些信息可以通过参数 ERROR 和 STATUS 输出表示。

ERROR	STATUS (十进制)	解释
0	9	超限警告： 使用新设备状态所覆盖较旧的设备状态。
0	11	警告：正在以较低优先级处理接收数据。
0	25	通讯已经启动。作业正在处理。
1	1	通讯故障，例如 <ul style="list-style-type: none"> <li>没有装载连接描述(本地或远程)</li> <li>连接中断(例如，电缆断线、CPU 关闭、CP 处于 STOP 模式)</li> </ul>
1	4	PHYS、LOG 或 LOCAL 中出错，包括数据长度或数据类型。
1	10	不能访问本地用户存储区(例如，访问一个已经删除的数据块)。
1	12	当调用 SFB 时， <ul style="list-style-type: none"> <li>指定了一个不属于 SFB 23 的实例 DB。</li> <li>没有指定任何实例 DB，而是指定了一个共享数据块。</li> <li>没有找到任何实例 DB(从 PG 中装载新实例 DB)。</li> </ul>
1	18	对于此 ID 所标识的连接，已经有一个用于 SFB 23"USTATUS"的实例。
1	19	远程 CPU 发送数据的速度快于用户程序中 SFB 可以接收的速度。
1	20	<ul style="list-style-type: none"> <li>S7-400：没有足够的工作存储器可用。如果仍然有足够的总工作存储器可用，则解决方案是压缩工作存储器。</li> <li>H 系统：在更新期间不能进行 SFB 的第一次调用，因为正在创建工作存储器中的通讯缓冲区。同样地，在更新期间不能改变工作存储器。</li> </ul>

## 22.16 通过 SFC 62"CONTROL"查询属于 SFB 实例的连接状态

### 描述

通过 SFC 62"CONTROL"，可以为 S7-400 查询属于本地通讯 SFB 实例的连接状态

在通过将控制输入 EN\_R 置 1 来调用系统函数之后，查询通讯 SFB 实例的连接的当前状态，通过 I\_DB 选择此实例。

参数	声明	数据类型	存储器区	描述
EN_R	INPUT	BOOL	I、Q、M、D、L、常数	经激活可进行接收的控制参数在输入置位时发出信号，通知该对象准备接收。
I_DB	INPUT	BLOCK_DB	I、Q、M、D、L、常数	实例 DB 的数目
OFFSET	INPUT	WORD	I、Q、M、D、L、常数	多重背景数据块中数据记录的编号(若没有多重背景数据块，则必须在此输入 0)。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I、Q、M、D、L I、Q、M、D、L	ERROR 和 STATUS 状态参数，出错显示： ERROR=0 和 STATUS 的数值： 0000H：既不是警告也不是出错 不等于 0000H：警告，STATUS 提供详细信息。 ERROR = 1 出错。STATUS 提供关于错误类型的详细信息。
I_TYP	OUTPUT	BYTE	I、Q、M、D、L	属于选定背景的块类型的标识符
I_STATE	OUTPUT	BYTE	I、Q、M、D、L	<ul style="list-style-type: none"> <li>=0：从最后一次冷启动/暖启动或装载之后，还没有调用相应的 SFB 实例。</li> <li>&lt;&gt;0：从最后一次冷启动/暖启动或装载之后，至少已经调用了一次相应的 SFB 实例。</li> </ul>
I_CONN	OUTPUT	BOOL	I、Q、M、D、L	相应连接的状态，可能的数值： <ul style="list-style-type: none"> <li>0: 连接断开或没有建立</li> <li>1: 连接存在</li> </ul>
I_STATUS	OUTPUT	WORD	I、Q、M、D、L	所查询的通讯 SFB 实例的状态参数 STATUS

## 输出参数 I\_TYP

下表列出了不同 SFB 类型和相应的标识符

SFB 类型	标识符(W#16#...)
USEND	00
URCV	01
BSEND	04
BRCV	05
GET	06
PUT	07
PRINT	08
START	0B
STOP	0C
RESUME	0D
STATUS	0E
USTATUS	0F
ALARM	15
ALARM_8	16
ALARM_8P	17
NOTIFY	18
AR_SEND	19
NOTIFY_8P	1A
(不存在 SFB ; I_DB 或 OFFSET 错误)	FF

## 错误信息

SFC 62"CONTROL"的输出参数 RET\_VAL 可能会为下列两个数值：

- 0000H : 在 SFC 的执行期间没有出错。
- 8000H : 在 SFC 的执行过程中出错。

### 注意

即使输出参数 RET\_VAL 指示了数值 0000H，还是应该评估输出参数 ERROR 和 STATUS。

ERROR	STATUS (十进制)	解释
1	10	无法访问本地用户存储区(例如，将存储器字节指定为 I_TYP 的实际参数，但是在正在使用的 CPU 中此存储器字节并不存在)。
1	12	对于通过 I_DB 指定的数字， <ul style="list-style-type: none"> <li>• 没有任何实例 DB，而有一个共享数据块。</li> <li>• 没有任何数据块，或实例已经损坏。</li> </ul>



## 23 用于未组态的 S7 连接的通讯 SFC

### 23.1 通讯 SFC 的公用参数

#### 输入参数 REQ

输入参数 REQ (请求激活)是电平触发控制参数。其用于触发作业(数据传送或连接中止)：

- 如果为当前没有激活的作业调用 SFC 则通过 REQ=1 来触发该作业。如果当第一次调用通讯 SFC 时，没有到通讯伙伴的连接，则在数据传送开始之前首先建立连接。
- 如果触发一个作业，并且当为同样的作业重新调用 SFC 时该作业还没有结束，则 SFC 不能使用 REQ。

#### 输入参数 REQ\_ID (只适用于 SFC 65 和 SFC 66)

输入参数 REQ\_ID 用于识别发送数据。该参数被发送 CPU 的操作系统传送到通讯伙伴的 CPU 的 SFC 66"X\_RCV"中。

对于下列两种情况，在接收端上需要 REQ\_ID 参数：

- 如果在一个发送 CPU 上通过不同参数 REQ\_ID 调用几个 SFC 65"X\_SEND"，并将数据传送到一个通讯伙伴。
- 如果使用 SFC 65"X\_SEND"，从几个发送 CPU 发送数据到一个通讯伙伴。

通过计算 REQ\_ID，可以将接收到的数据保存到不同的存储区域中。

#### 输出参数 RET\_VAL 和 BUSY

通讯 SFC 是异步执行的，也就是说作业的执行会在多个 SFC 调用中持续。输出参数 RET\_VAL 和 BUSY 指示作业的状态。参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义

### 输入参数 CONT

输入参数 CONT (继续)是一个控制参数。通过此参数可以决定：作业结束之后是否保持建立与通讯伙伴的连接。

- 如果在第一次调用时选择 CONT=0，则在数据传送结束之后将再次终止连接。然后，连接可重新用于与新通讯伙伴进行数据交换。

这种方法确保了只在实际使用时才占用连接资源。

- 如果在第一次调用时选择 CONT=1，则在数据传送结束时保持建立连接。

这种方法用于，例如，在两个站之间周期性地交换数据。

---

#### 注意

通过 CONT=1 建立的连接可以明确地通过 SFC 69"X\_ABORT"或 SFC 74"I\_ABORT"来终止。

---

## 23.2 用于未组态 S7 连接的通讯 SFC 的出错信息

### 错误信息

"SFC 65 到 SFC 74 的特有出错信息"表给出的用于 SFC 65 到 SFC 74 的"实际"出错信息，可以分类如下：

错误代码(W#16#...)	解释
809x	执行 SFC 的 CPU 上产生的出错
80Ax	永久通讯出错
80Bx	通讯伙伴上出错
80Cx	暂时出错

SFC 65 到 SFC 74 的特有出错信息

错误代码 (W#16#...)	解释(常规)	解释(用于特定 SFC)
0000	执行无错完成。	SFC 69"X_ABORT"和 SFC 74"I_ABORT"：REQ=1，并且没有建立指定的连接。 SFC 66"X_RCV"：EN_DT=1 和 RD=NIL
00xy	-	SFC 66"X_RCV"，NDA=1 和 RD<>NIL：RET_VAL 包含已接收数据的长度(EN_DT=0)，或复制到 RD 的数据长度(EN_DT=1)。 SFC 67"X_GET"：RET_VAL 包含已接收的数据块长度。 SFC 72"I_GET"：RET_VAL 包含已接收的数据块长度。
7000	-	SFC 65"X_SEND"、SFC 67"X_GET"、 SFC 68"X_PUT"、SFC 69"X_ABORT"、 SFC 72"I_GET"、SFC 73"I_PUT"和 SFC 74"I_ABORT"： 通过 REQ = 0 调用(调用而不执行)，BUSY 数值为 0，没有数据传送被激活。 SFC 66"X_RCV"：EN_DT=0/1，NDA=0
7001	首次调用 REQ=1： 数据传送被触发；BUSY 的值为 1。	-
7002	中间调用(REQ 无关联)： 数据传送已经激活； BUSY 的值为 1。	SFC 69"X_ABORT"和 SFC 74"I_ABORT"：中间调用，使用 REQ=1
8090	指定的通讯伙伴目标地址 无效，例如： <ul style="list-style-type: none"> <li>• 错误的 IOID</li> <li>• 存在错误的基址</li> <li>• 错误的 MPI 地址 (&gt; 126)</li> </ul>	-

错误代码 (W#16#...)	解释(常规)	解释(用于特定 SFC)
8092	SD 或 RD 中出错，例如： 不允许寻址本地数据区。	SFC 65"X_SEND"，例如 <ul style="list-style-type: none"> <li>• SD 长度非法</li> <li>• SD=NIL 是非法的</li> </ul>
		SFC 66"X_RCV"，例如 <ul style="list-style-type: none"> <li>• 已接收的数据超过 RD 指定的区域可以容纳的数据量。</li> <li>• RD 的数据类型是 BOOL，然而，所接收的数据长度大于 1 个字节。</li> </ul>
		SFC 67"X_GET"和 SFC 72"I_GET"，例如 <ul style="list-style-type: none"> <li>• RD 非法长度</li> <li>• RD 的长度或数据类型和已接收的数据不匹配。</li> <li>• 不允许 RD=NIL。</li> </ul>
		SFC 68"X_PUT"和 SFC 73"I_PUT"，例如 <ul style="list-style-type: none"> <li>• SD 长度非法</li> <li>• SD=NIL 是非法的</li> </ul>
8095	已经以较低优先级执行此功能块。	-
80A0	在已接收的应答中出错	SFC 68"X_PUT"和 SFC 73"I_PUT"：通讯伙伴不支持在发送 CPU 的 SD 中指定的数据类型。
80A1	通讯故障：在一个已存在的连接终止之后再调用 SFC	-
80B0	不能获得对象，例如，没有装载数据块	使用 SFC 67"X_GET"、SFC 68"X_PUT"、SFC 72"I_GET"和 SFC 73"I_PUT"时可能会发生这种错误
80B1	ANY 指针出错。 要发送的数据区的长度不正确。	-
80B2	硬件出错：模块不存在 <ul style="list-style-type: none"> <li>• 组态的插槽未占用。</li> <li>• 实际模块类型和希望的数据类型不匹配</li> <li>• 分布式外围 I/O 不可用。</li> <li>• 在相应 SDB 中没有用于此模块的条目。</li> </ul>	使用 SFC 67"X_GET"、SFC 68"X_PUT"、SFC 72"I_GET"和 SFC 73"I_PUT"时可能会发生这种错误
80B3	数据可能是只读或只写，例如，写保护的数据块	使用 SFC 67"X_GET"、SFC 68"X_PUT"、SFC 72"I_GET"和 SFC 73"I_PUT"时可能会发生这种错误
80B4	ANY 指针中的数据类型出错，或不允许使用指定类型的 ARRAY。	SFC 67"X_GET"、SFC 68"X_PUT"、SFC 72"I_GET"和 SFC 73"I_PUT"：通讯伙伴不支持在 VAR_ADDR 中指定的数据类型。
80B5	由于使用了非法模式，执行被拒绝	使用 SFC 65"X_SEND"时可能会发生这种错误
80B6	已接收的应答包含不能识别的错误代码。	-
80B7	已传送的数据的数据类型和/或长度与伙伴 CPU 上用于写入数据的区域不相符。	使用 SFC 68"X_PUT"和 SFC 73"I_PUT"时可能会发生这种错误

错误代码 (W#16#...)	解释(常规)	解释(用于特定 SFC)
80B8	-	SFC 65"X_SEND" : 通讯伙伴的 SFC 66"X_RCV"不允许数据接收(RD=NIL)。
80B9	-	SFC 65"X_SEND" : 通讯伙伴已经识别出数据块(SFC 66"X_RCV"调用, EN_DT=0), 但是还没有输入到用户程序, 因为通讯伙伴处于 STOP 模式。
80BA	通讯伙伴的响应和通讯帧 不相符。	-
80C0	指定的连接正在被其他作 业使用。	-
80C1	执行 SFC 的 CPU 上资源不 足, 例如 : 正在模块上执行的不同发 送作业数已经达到允许的 最大数目。 连接资源正在使用, 例 如, 正用于接收数据。	-
80C2	在通讯伙伴上, 资源暂时 不足, 例如 : <ul style="list-style-type: none"> <li>• 通讯伙伴当前正在处 理最大数目的作业。</li> <li>• 所需要的资源、存储 器等正被使用。</li> <li>• 没有足够的工作存储 器可用。(压缩存储 器)。</li> </ul>	-
80C3	在连接建立中出错, 例如 : <ul style="list-style-type: none"> <li>• 本地 S7 站没有连接到 MPI 子网。</li> <li>• 已经在 MPI 子网上对 站点进行了编址。</li> <li>• 无法访问通讯伙伴。</li> <li>• 在通讯伙伴上, 资源 暂时不足</li> </ul>	-

## 23.3 通过 SFC 65"X\_SEND"发送数据到本地 S7 站以外的通讯伙伴

### 描述

通过 SFC 65"X\_SEND", 发送数据到本地 S7 站以外的通讯伙伴。

在通讯伙伴上使用 SFC 66"X\_RCV"接收数据。

在通过 REQ=1 调用 SFC 之后再发送数据。

必须要确保由参数 SD (在发送 CPU 上)定义的发送区小于或等于由参数 RD (在通讯伙伴上)定义的接收区。若 SD 是 BOOL 数据类型, 则 RD 必须也是 BOOL 类型。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"请求激活", 请参见用于 S7 基本通讯的 SFC 公共参数
CONT	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"继续", 请参见用于 S7 基本通讯的 SFC 公共参数
DEST_ID	INPUT	WORD	I、Q、M、D、L、常数	地址参数"目标 ID"。它包含了通讯伙伴的 MPI 地址。通过 STEP 7 组态了此参数。
REQ_ID	INPUT	DWORD	I、Q、M、D、L、常数	作业标识符。此参数用于识别通讯伙伴上的数据。
SD	INPUT	ANY	I、Q、M、D	指向发送区。允许使用下列数据类型: BOOL、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME 和这些数据类型的数组, 除 BOOL 外。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行功能时发生错误, 返回值中将包含相应的错误代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1: 发送还没有结束。 BUSY=0: 发送已经结束, 或没有发送函数被激活。

### 数据一致性

数据以一致状态发送。

### 错误信息

请参见用于未组态 S7 连接的通讯 SFC 的出错信息和通过输出参数 RET\_VAL 判断错误

## 23.4 通过 SFC 66 "X\_RCV"从本地 S7 站以外的通讯伙伴中接收数据

### 描述

通过 SFC 66"X\_RCV" ,接收本地 S7 站以外的一个或多个通讯伙伴通过 SFC 65"X\_SEND"发送的数据。

通过 SFC 66"X\_RCV" ,

- 可以检查数据是已经发送还是正在等待复制。数据被操作系统输入到内部队列。
- 可以将队列中最早的数据块复制到所选择的接收区。

参数	声明	数据类型	存储器区	描述
EN_DT	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"激活数据传送"。通过数值 0, 可以检查是否至少有一个数据块正在等待被输入到接收区。数值 1 复制队列中最早的数据块到 RD 指定的工作存储区域。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行功能时发生错误, 返回值中将包含相应的错误代码。如果没有出错, 则 RET_VAL 包含下列内容: <ul style="list-style-type: none"> <li>• W#16#7000, 如果 EN_DT=0/1 且 NDA=0。在这种情况下, 队列中没有数据块。</li> <li>• 如果 EN_DT=0 且 NDA=1, 在队列中输入的最早的数据块的长度是一个以字节为单位的正数。</li> <li>• 如果 EN_DT=1 和 NDA=1, 复制到 RD 接收区的数据块长度是一个以字节为单位的正数。</li> </ul>
REQ_ID	OUTPUT	DWORD	I、Q、M、D、L	SFC"X_SEND"的作业标识符, 它的数据位于队列的开头, 即, 是队列中最早的数据。如果队列中没有数据块, 则 REQ_ID 数值为 0。
NDA	OUTPUT	BOOL	I、Q、M、D、L	状态参数"已到达的新数据"。 NDA=0 : <ul style="list-style-type: none"> <li>• 队列中没有数据块。</li> </ul> NDA=1 : <ul style="list-style-type: none"> <li>• 队列至少包含了一个数据块。(调用 SFC 66, EN_DT=0)。</li> <li>• 队列中最早的数据块被复制到用户程序中。(调用 SFC 66, EN_DT=1)。</li> </ul>
RD	OUTPUT	ANY	I、Q、M、D	指向接收数据区域。允许使用下列数据类型: BOOL、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5_TIME、DATE_AND_TIME 和这些数据类型的数组, 除 BOOL 外。 如果要丢弃队列中最早的数据块, 则将数值 NIL 分配给 RD。 接收区的最大长度是 76 个字节。

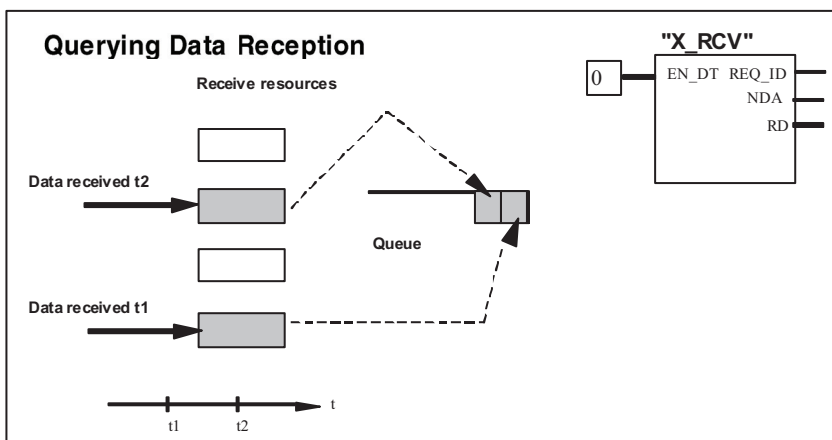
### 通过 EN\_DT=0 指示数据接收

只要来自通讯伙伴的数据到达，操作系统就按照接收顺序将这些数据输入到队列中。

如果要检查队列中是否至少有一个数据块，则通过 EN\_DT=0 调用 SFC 66，并按如下方式计算输出参数 NDA：

- NDA=0 意味着队列并未包含数据块。REQ\_ID 与此不相关。RET\_VAL 数值为 W#16#7000。
- NDA=1 意味着队列中至少有一个数据块可以被读取。

这种情况下，应该判断输出参数 RET\_VAL 和 REQ\_ID (如果可用)。RET\_VAL 包含以字节为单位的数据块长度，REQ\_ID 包含发送块的作业标识符。如果队列中有多个数据块，则 REQ\_ID 和 RET\_VAL 属于队列中最早的数据块。

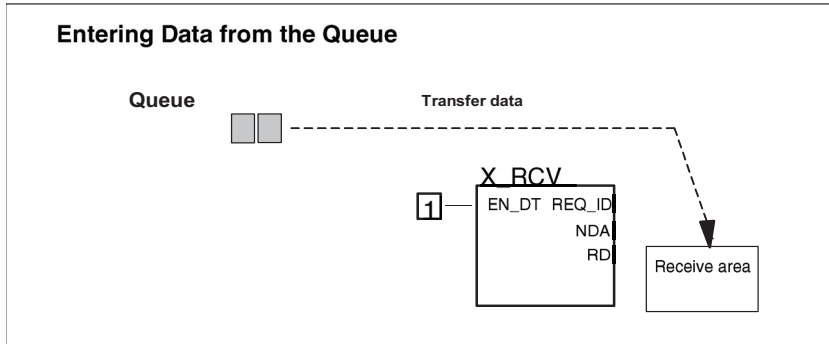


数据接收



## 通过 EN\_DT=1 在接收区中输入数据

当通过 EN\_DT=1 调用 SFC 66"X\_RCV"时，队列中最早的数据块被复制到 RD 指定的工作存储器区域。RD 必须大于或等于由 SD 参数定义的相应 SFC 65"X\_SEND"的发送区。如果输入参数 SD 是 BOOL 数据类型，则 RD 必须也是 BOOL 数据类型。如果要在不同区域内输入接收到的数据，则可以查询 REQ\_ID (SFC 调用，EN\_DT = 0)，并在继续调用中(EN\_DT = 1)选择一个合适的 RD。如果在复制数据时没有出错，则 RET\_VAL 包含以字节为单位的所复制的数据块的长度，并给发送方发送一个肯定应答。



数据接收

## 放弃数据

若不希望从队列中输入数据，则将数值 NIL 分配给 RD (参见/232/)。在这种情况下，发送方接收一个否定应答(相应 SFC 65"X\_SEND"的 RET\_VAL，数值为 W#1680B8)。SFC 66"X\_RCV"的 RET\_VAL 数值为 0。

## 数据一致性

在通过 EN\_DT=1 和 RETVAL=W#16#00xy 进行调用之后，接收区 RD 包含了新数据。这些数据可能会被随后的功能块调用所覆盖。为了防止发生这种情况，在评估完接收数据之前，请不要使用相同的接收区 RD 来调用 SFC 66"X\_RCV"。

## 切换到 STOP 模式

如果 CPU 切换到 STOP 模式，

- 则否定应答所有新到达的作业。
- 否定应答所有已经到达并存储在队列中的作业。
  - 如果在 STOP 之后暖启动或冷启动，则所有数据块都将被丢弃。
  - 如果在 STOP 之后重新启动且在切换到 STOP 模式之前查询队列(通过调用 SFC 66 "X\_RCV"，EN\_DT=0)，则属于最早作业的数据块被输入用户程序(S7-300 和 S7-400H 系统不可以)。否则数据块将被丢弃。

所有其他数据块都将被丢弃。

### 连接中止

如果连接被终止，则将放弃已经在队列中、属于此连接的作业。

例外：如果此作业是队列中最早的作业，并且已经通过调用 SFC 66"X\_RCV"(EN\_DT=0)检测到它的存在，则可以通过 EN\_DT=1 将它输入到接收区。

### 错误信息

请参见用于未组态 S7 连接的通讯 SFC 的出错信息和通过输出参数 RET\_VAL 判断错误

## 23.5 通过 SFC 68 "X\_PUT"将数据写入本地 S7 站以外的通讯伙伴

### 描述

通过 SFC 68"X\_PUT" ,将数据写入不在同一个本地 S7 站中的通讯伙伴。在通讯伙伴上没有相应 SFC。在通过 REQ=1 调用 SFC 之后, 激活写作业。此后, 可以继续调用 SFC, 直到 BUSY=0 指示接收到应答为止。

必须要确保由 SD 参数(在发送 CPU 上)定义的发送区和由 VAR\_ADDR 参数(在通讯伙伴上)定义的接收区长度相同。SD 的数据类型还必须和 VAR\_ADDR 的数据类型相匹配。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"请求激活", 参见 用于 S7 基本通讯的 SFC 的公用参数
CONT	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"继续", 请参见控制参数"请求激活", 通讯 SFC 的公用参数
DEST_ID	INPUT	WORD	I、Q、M、D、L、常数	地址参数"目标 ID"。它包含了通讯伙伴的 MPI 地址。通过 STEP 7 组态了此参数。
VAR_ADDR	INPUT	ANY	I、Q、M、D	指向伙伴 CPU 上要写入数据的区域。必须选择通讯伙伴支持的数据类型。
SD	INPUT	ANY	I、Q、M、D	指向本地 CPU 中包含要发送数据的区域。允许使用下列数据类型: BOOL、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5_TIME、DATE_AND_TIME 和这些类型的数组, 除 BOOL 外。 SD 必须与通讯伙伴上的 VAR_ADDR 长度相同。SD 的数据类型还必须和 VAR_ADDR 的数据类型相匹配。发送区的最大长度是 76 个字节
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行功能时发生错误, 返回值中将包含相应的错误代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1: 发送还没有结束。 BUSY=0: 发送已经结束, 或没有发送函数被激活。

### 切换到 STOP 模式

如果 CPU 切换到 STOP 模式, 则将终止通过 SFC 68"X\_PUT"建立的连接。不再继续发送数据。如果当 CPU 改变工作模式时发送数据已经复制到内部缓冲区, 则缓冲区的内容将被丢弃。

### 通讯伙伴切换到 STOP 模式

如果通讯伙伴的 CPU 切换到 STOP 模式, 不会影响通过 SFC 68"X\_PUT"进行的数据传送。处于 STOP 模式的通讯伙伴也可以写数据。

### 数据一致性

数据以一致状态发送。

### 错误信息

请参见用于未组态 S7 连接的通讯 SFC 的出错信息和通过输出参数 RET\_VAL 判断错误

## 23.6 通过 SFC 67 "X\_GET"从本地 S7 站以外的通讯伙伴中读取数据

### 描述

通过 SFC 67"X\_GET"，可以从本地 S7 站以外的通讯伙伴中读取数据。在通讯伙伴上没有相应 SFC。在通过 REQ=1 调用 SFC 之后，激活读作业。此后，可以继续调用 SFC，直到 BUSY=0 指示数据接收为止。然后，RET\_VAL 便包含了以字节为单位的、已接收的数据块的长度。

必须要确保由 RD 参数定义接收区(在接收 CPU 上)至少和由 VAR\_ADDR 参数定义的要读取的区域(在通讯伙伴上)一样大。RD 的数据类型还必须和 VAR\_ADDR 的数据类型相匹配。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"请求激活"，参见通讯 SFC 的公用参数
CONT	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"继续"，参见通讯 SFC 的公用参数
DEST_ID	INPUT	WORD	I、Q、M、D、L、常数	地址参数"目标 ID"。它包含了通讯伙伴的 MPI 地址。通过 STEP 7 组态了此参数。
VAR_ADDR	INPUT	ANY	I、Q、M、D	指向伙伴 CPU 上要从中读取数据的区域。必须选择通讯伙伴支持的数据类型。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行功能时发生错误，返回值中将包含相应的错误代码。 如果没有出错，RET_VAL 包含复制到接收区 RD 的数据块的长度，它是一个以字节为单位的正数。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：接收还没有结束。 BUSY=0：接收已经结束，或当前没有激活的接收作业。
RD	OUTPUT	ANY	I、Q、M、D	指向接收区(接收数据区)。允许使用下列数据类型：BOOL、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5_TIME、DATE_AND_TIME 和这些数据类型的数组，除 BOOL 外。 接收区 RD 必须至少和通讯伙伴上要读取的数据区域 VAR_ADDR 一样长。RD 的数据类型还必须和 VAR_ADDR 的数据类型相匹配。 接收区的最大长度是 76 个字节。

### 切换到 STOP 模式

如果 CPU 切换到 STOP 模式，则将终止通过 SFC 67"X\_GET"建立的连接。位于操作系统缓冲区中的、已接收到的数据是否会丢失，取决于所执行的重启动类型：

- 在热启动之后(S7-300 和 S7-400H 系统不可以)，数据被复制到由 RD 定义的区域中。
- 在暖启动或冷启动之后，数据被丢弃。

### 通讯伙伴切换到 STOP 模式

如果通讯伙伴的 CPU 切换到 STOP 模式，这不会影响通过 SFC 67"X\_GET"进行的数据传送。处于 STOP 模式的通讯伙伴也可以读取数据。

### 数据一致性

数据以一致状态接收。

### 错误信息

请参见用于未组态 S7 连接的通讯 SFC 的出错信息和通过输出参数 RET\_VAL 判断错误

## 23.7 通过 SFC 69"X\_ABORT"中止已存在的、到本地 S7 站以外的通讯伙伴的连接

### 描述

通过 SFC 69"X\_ABORT"，终止一个通过 SFC X\_SEND、X\_GET 或 X\_PUT 建立的、到不在同一个本地 S7 站的通讯伙伴的连接。如果属于 X\_SEND、X\_GET 或 X\_PUT 的作业已结束(BUSY = 0)，则在调用 SFC 69"X\_ABORT"之后，将释放在通讯两端使用的连接资源。如果属于 X\_SEND、X\_GET 或 X\_PUT 的作业还没有结束(BUSY = 1)，则在连接中止之后重新通过 REQ = 0 和 CONT = 0 调用相关的 SFC，然后等待 BUSY = 0。只有这样才能重新释放所有连接资源。只能在有 SFC"X\_SEND"、"X\_PUT"或"X\_GET"的通讯端点上才可以调用 SFC 69 "X\_ABORT"。通过 REQ=1 来调用 SFC，激活中止的连接。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"请求激活"，参见通讯 SFC 的公用参数
DEST_ID	INPUT	WORD	I、Q、M、D、L、常数	地址参数"目标 ID"。它包含了通讯伙伴的 MPI 地址。通过 STEP 7 组态了此参数。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行功能时发生错误，返回值中将包含相应的错误代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：连接中止还没有完成。 BUSY=0：连接中止已经完成。

### 切换到 STOP 模式

如果 CPU 切换到 STOP 模式，则结束通过 SFC 69"X\_ABORT"启动的连接中止。

### 通讯伙伴切换到 STOP 模式

如果通讯伙伴的 CPU 切换到 STOP 模式，不会影响通过 SFC 69"X\_ABORT"进行的连接中止。连接被终止。

### 错误信息

请参见用于未组态 S7 连接的通讯 SFC 的出错信息和通过输出参数 RET\_VAL 判断错误

## 23.8 通过 SFC 72 "I\_GET"从本地 S7 站内的通讯伙伴中读取数据

### 描述

通过 SFC 72"I\_GET", 可以从同一个本地 S7 站内的通讯伙伴中读取数据。通讯伙伴可以位于中央机架, 也可以位于扩展机架或分布式机架中。一定要通过 STEP 7 分配分布式通讯伙伴到本地 CPU。在通讯伙伴上没有相应 SFC。

在通过 REQ=1 调用 SFC 之后, 激活接收作业。此后, 可以继续调用 SFC, 直到 BUSY=0 指示数据接收为止。然后, RET\_VAL 便包含了以字节为单位的、已接收的数据块的长度。

必须要确保由 RD 参数定义的接收区(在接收 CPU 上)至少和由 VAR\_ADDR 参数定义的要读取的区域(在通讯伙伴上)一样大。RD 的数据类型还必须和 VAR\_ADDR 的数据类型相匹配。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"请求激活", 参见 通讯 SFC 的公用参数.
CONT	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"继续", 参见通讯 SFC 的公用参数.
IOID	INPUT	BYTE	I、Q、M、D、L、常数	通讯伙伴模块的地址范围标识符: B#16#54= 外设输入(PI) B#16#55= 外设输出(PQ) 混合模块的范围标识符是两个地址中较小的一个。如果两个地址相同, 则指定 B#16#54。
LADDR	INPUT	WORD	I、Q、M、D、L、常数	通讯伙伴模块的逻辑地址。如果是一个混合模块, 则指定两个地址中较小的一个。
VAR_ADDR	INPUT	ANY	I、Q、M、D	指向伙伴 CPU 上要从中读取数据的区域。选择通讯伙伴支持的数据类型。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行功能时发生错误, 返回值中将包含相应的错误代码。 如果没有出错, RET_VAL 包含复制到接收区 RD 的数据块的长度, 它是一个以字节为单位的正数。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1 : 接收还没有结束。 BUSY=0 : 接收已经结束, 或当前没有激活的接收作业。
RD	OUTPUT	ANY	I、Q、M、D	指向接收区(接收数据区)。允许使用下列数据类型: BOOL、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5_TIME、DATE_AND_TIME 和这些数据类型的数组, 除 BOOL 外。 接收区 RD 必须至少和通讯伙伴上要读取的数据区域 VAR_ADDR 一样长。RD 的数据类型还必须和 VAR_ADDR 的数据类型相匹配。 接收区的最大长度是 76 个字节。



### 切换到 STOP 模式

如果 CPU 切换到 STOP 模式,则将终止通过 SFC 72"L\_GET"建立的连接。位于操作系统缓冲区中的、已接收到的数据是否会丢失,取决于所执行的重启动类型:

- 在热启动之后(S7-300 和 S7-400H 系统不可以),数据被复制到由 RD 定义的区域中。
- 在暖启动或冷启动之后,数据被丢弃。

### 通讯伙伴切换到 STOP 模式

如果通讯伙伴的 CPU 切换到 STOP 模式,不会影响通过 SFC 72"L\_GET"进行的数据传送。处于 STOP 模式的通讯伙伴也可以读取数据。

### 数据一致性

数据以一致状态接收。

### 错误信息

请参见用于未组态 S7 连接的通讯 SFC 的出错信息和通过输出参数 RET\_VAL 判断错误

## 23.9 通过 SFC 73 "I\_PUT"将数据写入本地 S7 站内的通讯伙伴

### 描述

通过 SFC 73"I\_PUT", 将数据写入相同本地 S7 站内的通讯伙伴。通讯伙伴可以位于中央机架, 也可以位于扩展机架或分布式机架中。一定要通过 STEP 7 分配分布式通讯伙伴到本地 CPU。在通讯伙伴上没有相应 SFC。

在使用 REQ 控制输入上的信号电平为 1 调用 SFC 之后, 激活发送作业。

必须要确保由 SD 参数(在发送 CPU 上)定义的发送区和由 VAR\_ADDR 参数(在通讯伙伴上)定义的接收区长度相同。SD 的数据类型还必须和 VAR\_ADDR 的数据类型相匹配。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"请求激活", 参见通讯 SFC 的公用参数.
CONT	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"继续", 参见通讯 SFC 的公用参数.
IOID	INPUT	BYTE	I、Q、M、D、L、常数	通讯伙伴模块的地址范围标识符: B#16#54= 外设输入(PI) B#16#55= 外设输出(PQ) 混合模块的范围标识符是两个地址中较小的一个。如果两个地址相同, 则指定 B#16#54。
LADDR	INPUT	WORD	I、Q、M、D、L、常数	通讯伙伴模块的逻辑地址。如果是一个混合模块, 则指定两个地址中较小的一个。
VAR_ADDR	INPUT	ANY	I、Q、M、D、L	指向通讯伙伴上要写入数据的区域。选择通讯伙伴支持的数据类型。
SD	INPUT	ANY	I、Q、M、D	指向本地 CPU 上包含要发送数据的区域。允许使用下列数据类型: BOOL、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME 和这些类型的数组, 除 BOOL 外。 SD 必须和通讯伙伴上的参数 VAR_ADDR 长度相同。SD 的数据类型还必须和 VAR_ADDR 的数据类型相匹配。 发送区的最大长度是 76 个字节
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行功能时发生错误, 返回值中将包含相应的错误代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1: 发送还没有结束。 BUSY=0: 发送已经结束, 或没有发送函数被激活。

### 切换到 STOP 模式

如果 CPU 切换到 STOP 模式，则将终止使用 SFC 73"I\_PUT"建立的连接。不再继续发送数据。如果当 CPU 改变工作模式时发送数据已经复制到内部缓冲区，则缓冲区的内容将被丢弃。

### 通讯伙伴切换到 STOP 模式

如果通讯伙伴的 CPU 切换到 STOP 模式，不会影响通过 SFC 73"I\_PUT"进行的数据传送。处于 STOP 模式的通讯伙伴也可以写数据。

### 数据一致性

数据以一致状态发送。

### 错误信息

请参见用于未组态 S7 连接的通讯 SFC 的出错信息和通过输出参数 RET\_VAL 判断错误

## 23.10 通过 SFC 74 "I\_ABORT"中止已存在的、到本地 S7 站内的通讯伙伴的连接

### 描述

通过 SFC 74 "I\_ABORT"，终止通过 SFC 72 "I\_GET"或 SFC 73 "I\_PUT"建立的、到同一个本地 S7 站内的通讯伙伴的连接。如果属于 I\_GET 或 I\_PUT 的作业已结束(BUSY = 0) ,则在调用 SFC 74 "I\_ABORT"之后，将释放在通讯两端使用的连接资源。

如果属于 I\_GET 或 I\_PUT 的作业还没有结束(BUSY = 1)，则在连接中止之后重新通过 REQ = 0 和 CONT = 0 调用相关的 SFC，然后等待 BUSY = 0。只有这样才能重新释放所有连接资源。

只能在有 SFC "I\_PUT"或 "I\_GET"的通讯端上(即在客户端上)调用 SFC 74 "I\_ABORT"。

通过 REQ=1 来调用 SFC，激活中止的连接。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	控制参数"请求激活"，参见通讯 SFC 的公用参数。
IOID	INPUT	BYTE	I、Q、M、D、L、常数	通讯伙伴模块的地址范围标识符： B#16#54= 外设输入(PI) B#16#55= 外设输出(PQ) 混合模块的范围标识符是两个地址中较小的一个。如果两个地址相同，则指定 B#16#54。
LADDR	INPUT	WORD	I、Q、M、D、L、常数	通讯伙伴模块的逻辑地址。如果是一个混合模块，则指定两个地址中较小的一个。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果执行功能时发生错误，返回值中将包含相应的错误代码。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：连接中止还没有完成。 BUSY=0：连接中止已经完成。

### 切换到 STOP 模式

如果 CPU 切换到 STOP 模式，则结束通过 SFC 74 "I\_ABORT"启动的连接中止。

### 通讯伙伴切换到 STOP 模式

如果通讯伙伴的 CPU 切换到 STOP 模式，不会影响通过 SFC 74 "I\_ABORT"进行的连接中止。连接被终止。

### 错误信息

请参见用于未组态 S7 连接的通讯 SFC 的出错信息和通过输出参数 RET\_VAL 判断错误

## 24 通过 Industrial Ethernet 打开通讯

### 24.1 概述

#### 通过 Industrial Ethernet 打开通讯

STEP 7 在位于"通信块"下的"标准库"中提供了下列 FB 和 UDT ,用于通过 Ethernet(TCP、ISO-on-TCP、UDP 协议) ,使用用户程序与其它通信伙伴交换数据 :

- 连接导向协议 : 依据 RFC 793 的 TCP , 依据 RFC 1006 的 ISO on TCP :
  - 用于分配连接参数的 UDT 65 "TCON\_PAR"及其数据结构
  - 具有协议特定默认值的 UDT 651 至 UDT 656
  - 用于建立连接的 FB 65 "TCON"
  - 用于终止连接的 FB 66 "TDISCON"
  - 用于发送数据的 FB 63 "TSEND"
  - 用于接收数据的 FB 64 "TRCV"
- 无连接协议 : 依据 RFC 768 的 UDP
  - UDT 65"TCON\_PAR" , 具有分配本地通讯接入点参数的数据结构
  - 具有协议特定默认值的 UDT 657
  - UDT 66"TCON\_ADR" , 具有分配远程伙伴寻址参数的数据结构
  - 具有协议特定默认值的 UDT 661
  - FB 65 "TCON" , 用于组态本地通讯接入点
  - FB 66 "TDISCON" , 用于关闭本地通讯接入点
  - 用于发送数据的 FB 67 "TUSEND"
  - 用于接收数据的 FB 68 "TURCV"
- 使用 FETCH 和 WRITE 服务链接到第三方系统
  - FB 210 "FW\_TCP" : 用于使用 TCP 的第三方链接
  - FB 220 "FW\_IOT" : 用于使用 ISO-on-TCP 的第三方链接

## 24.2 用于工业以太网上开放通讯的 FB 的功能

### 连接导向和无连接协议

在数据通信中主要有下列协议类型：

- 连接导向协议类型：  
用于在数据传送开始之前建立一个到通讯伙伴的逻辑连接。如必要，将在数据发送完成之后终止连接。连接导向协议用于对数据发送的可靠和安全性要求特别高的场合。通常，一条物理线路上可以存在许多逻辑连接。

用于工业以太网上开放通讯的 FB 支持下列连接导向协议：

- 依据 RFC 793 的 TCP/IP
- 按照 RFC 1006 的 ISO-on-TCP
- 无连接协议类型：  
它们工作时不需要连接。因此，也就没有与远程伙伴间的连接建立和终止发生。无连接协议传送未确认数据，向远程伙伴不可靠、不保证地传送，换句话说，数据可能在块不指示的情况下丢失。

用于 Industrial Ethernet 上开放通信的 FB 支持下列连接导向协议：

- 依据 RFC 768 的 UDP

功能块实际如何工作取决于所使用的协议变量。这将在下一节中详细论述。

### 接收区

本术语将在下列部分反复使用。它指的是功能块输入所接收数据的区域。

接收区由下列两个变量指定：

- 指向区域起始地址的指针
- 区域长度

区域长度由 LEN 参数使用的协议类型指定(如果 LEN <> 0)或 DATA 参数长度信息指定(如果 LEN = 0)。

在 ANY 指针中允许使用下列数据类型：BOOL、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TIME\_OF\_DAY、TIME、S5TIME、DATE\_AND\_TIME。

## TCP

在数据传输期间，不会传输关于消息的长度信息或有关消息的起始和结束信息。这在发送过程中不算什么问题，因为发送器知道它将发送多少数据字节。但是接收器无法检测数据流中的某条消息在何处结束以及下一条消息在何处开始。因此，建议在通信伙伴上将接收区 FB64 "TRCV"精确地设置为 LEN 参数 FB63 "TSEND" (要发送的字节数)。

- 在 ad-hoc 模式中接收数据：  
接受区与使用 FB64 "TRCV"的 DATA 参数指定的区域完全相同

接收到一个数据块后，FB 64 "TRCV"立即将其传送到接收区，并将 NDR 设为 1。最大长度为 8192 字节。

若接收区选定的长度大于发送数据的长度，则 FB 64 "TRCV"在接收区内输入所有接收的数据。然后将 NDR 设置为 TRUE，并将所接收的数据的长度写入 RCVD\_LEN。

如果您选择了一个接收区，但接收区的长度小于所发送数据的长度，FB64 将尽量复制数据，直到接收区满。然后将 NDR 设置为 TRUE，并将接收区的长度写入 RCVD\_LEN。每次进一步调用时，都将收到另一个发送数据块。

- 接收具有预设长度的数据：  
接收区由 DATA 参数(接收区的开始地址)和 FB64 的 LEN 参数(接收区的长度)定义。

如果所接收的数据无法完全填满接收区，该数据一开始时无法使用。只有当后续数据完全填满接收区时，该数据才可用。请注意，在这种情况下，来自两个不同发送作业的数据将位于同一个接收区内。如果您无法识别第一个消息的末尾或第二个消息的起始，您将无法检测到第一个或第二个消息。

如果您选择了一个接收区，但接收区的长度小于所发送数据的长度，FB64 将尽量复制字节，直到接收区满。

当 FB64 完全填满接收区后，会将 NDR 设置为 TRUE，并将数值 LEN 写入 RCVD\_LEN。每次进一步调用时，都将收到另一个发送数据块。

## ISO-on-TCP

在数据传输期间，同时传送长度和消息末尾信息。

如果您已经选择了长于发送数据的接收区，则 FB64 "TRCV"输入所有接收区中的数据。然后将 NDR 设置为 TRUE，并将发送数据的长度写入 RCVD\_LEN。

如果选择了比发送数据短的接收区，则 FB64 不向接收区复制任何数据，并返回下列错误信息：  
ERROR=1，STATUS=W#16#8088。

## UDP

没有按照 TCP 和 ISO-on-TCP 协议建立连接。因此当调用发送块 FB 67 "TUSEND" 时，必须指定接收器(IP 地址和端口编号)地址参数的参考值。同样地，在接收块 FB 68 "TURCV"结束之后，将会接收到发送器(IP 地址和端口号)地址参数的参考值。

为了使用 FB 67 "TUSEND"和 68 "TURCV"，首先需要同时在发送端和接收端调用 FB 65 "TCON"，以设置本地通信接入点。

在每次调用 FB 67 "TUSEND"时，可以通过指定 IP 地址和端口号重新引用远程伙伴。

在数据传输期间，同时传送长度和消息末尾信息。

如果您已经选择了长于发送数据的接收区，则 FB64 "TURCV"将所有数据输入接收区中。然后将 NDR 设置为 TRUE，并将发送数据的长度写入 RCVD\_LEN。

如果选择了比发送数据短的接收区，则 FB64 不向接收区复制任何数据，并返回下列错误信息：ERROR=1，STATUS=W#16#8088。

## TCP 和 UDP 的端口号

按如下寻址连接：

- 通过 TCP：
  - 连接：通过远程 IP 地址、远程端口和通过本地端口(可以显式设置本地端口的端口号或由操作系统对此进行分配)。
  - 被动连接：通过本地端口。如果想要多次使用本地端口，还需要指定 IP 地址或远程 IP 地址和远程端口。
- 通过 UDP：通过远程 IP 地址、远程端口和本地端口(显式设置本地端口的端口号)。

下面列出的范围适用于本地端口。只有在远程伙伴是 S7 CPU 时，这些范围才适用于远程端口。否则，远程端口没有任何限制。

- 对于固件版本最高为 V2.6 的 CPU 31x-2 PN/DP (含)和固件版本最高为 V5.1 的 CPU 41x-3 PN/DP (含)：
  - 参数分配中允许的端口号(UDT 65)：2000 - 5000
  - 若参数分配不包括一个显式设置的端口号，则操作系统从 49152 - 65534 的动态编号范围内分配一个端口。
- 对于固件版本 V2.7 起的 CPU 31x-2 PN/DP 和固件版本 V5.2 起的 CPU 41x-3 PN/DP：
  - 参数分配中允许的端口号(UDT 65)：1 - 49151，由于西门子为系统保留 1-1999 和 5001-49151 之间的编号，因此建议将端口号限制为 2000 - 5000。
  - 若参数分配不包括一个显式设置的端口号，则操作系统从 49152 - 65534 的动态编号范围内分配一个端口。
- 对于固件版本从 V3.2 起的 CPU 31x-2 PN/DP、固件版本从 V6.0 起的 CPU 41x-3 PN/DP、固件版本从 V6.0 起的 CPU 41x-5H PN/DP 和固件版本从 V6.0 版本起的 CPU 41x-5H PN/DP
  - 可多次使用一个端口。



下列列表包含为系统保留的端口号：

协议	端口号	服务
-	0	(该端口具有一个固定分配。禁止使用。)
TCP	20 和 21	FTP
TCP	25	SMTP
TCP	80	HTTP
TCP	102	RFC 1006
UDP	135	RPC-DCOM
UDP	161	SNMP_REQUEST
TCP、UDP	34962	PNIO
TCP、UDP	34963	PNIO
TCP、UDP	34964	PNIO
UDP	65532	NTP
UDP	65533	NTP
UDP	65534	NTP
UDP	65535	NTP

参见 <http://www.iana.org/assignments/port-numbers>。

---

**注意**

不得使用为系统保留的任何端口号。

---

## 24.3 为 TCP 和 ISO-on-TCP 通信连接分配参数

### 分配参数的数据块

为了给用于通信连接的 TCP 和 ISO on TSP 分配参数,创建包含了 UDT 65 "TCON\_PAR"数据结构的 DB。此数据结构包含组态连接所必需的参数。每个连接都需要这样一个数据结构。您可以在全局数据块中对该结构进行汇编。

FB 65 " TCON" 的 CONNECT 连接参数地址包含一个关联连接描述的引用(例如, P#DB100.DBX0.0 byte 64)。

### 连接结构描述(UDT 65)

字节	参数	数据类型	起始值	描述
0 - 1	block_length	WORD	W#16#40	UDT 65 的长度: 64 字节(固定)
2 到 3	id	WORD	W#16#0001	对连接的引用(取值范围: W#16#0001 至 W#16#0FFF) 您必须在各个带有标识号的块中指定参数的值。
4	connection_type	BYTE	B#16#11	协议类型: <ul style="list-style-type: none"> <li>B#16#11: TCP</li> <li>B#16#12: ISO-on-TCP</li> <li>B#16#01: TCP (兼容模式)</li> </ul>
5	active_est	BOOL	FALSE	建立连接的方式的标识号: <ul style="list-style-type: none"> <li>FALSE: 被动建立</li> <li>TRUE: 主动建立</li> </ul>

字节	参数	数据类型	起始值	描述
6	local_device_id	BYTE	B#16#02	<ul style="list-style-type: none"> <li>B#16#00：通过 CP 443-1 进行通信(只用于 S7-400，且 connection_type = B#16#12)。允许的 CP：CP443-1EX4x、CP443-1EX20、CP443-1GX20、CP443-1EX30、CP443-1GX30</li> <li>B#16#01：通过接口插槽 1 (IF1)中的 IE 接口与 WinAC RTX (仅 TCP)进行通信</li> <li>B#16#02：通过 CPU 315-2 PN/DP 和 317-2 PN/DP 的集成 IE 接口进行通讯</li> <li>B#16#03：通过集成 IE 接口与 CPU 315T-3 PN/DP、317T-3 PN/DP 和 319-3 PN/DP 进行通信</li> <li>B#16#05：通过用于 CPU 41x 与 41H (机架 0)的集成 IE 接口 X5 进行通信</li> <li>B#16#06：通过接口插槽 2 (IF2)中的 IE 接口与 WinAC RTX (仅 TCP)进行通信</li> <li>B#16#08：通过用于 CPU 41x 与 41H (机架 0)的集成 IE 接口 X8 进行通信</li> <li>B#16#0B。通过接口插槽 3 (IF3)中的 IE 接口与 WinAC RTX (仅 TCP)进行通信</li> <li>B#16#0F：通过接口插槽 4 (IF4)中的 IE 接口与 WinAC RTX (仅 TCP)进行通信</li> <li>B#16#10：如果将 CP 插入到机架 1 或相关扩展机架，则通过 CP 443-1 (仅用于 S7-400H 且连接类型 = B#16#12)进行通信</li> <li>B#16#15：通过用于 CPU 41xH (机架 1)的集成 IE 接口 X5 进行通信</li> <li>B#16#18：通过用于 CPU 41xH (机架 1)的集成 IE 接口 X8 进行通信</li> </ul>
7	local_tsap_id_len	BYTE	B#16#02	<p>所使用的参数 local_tsap_id 的长度；可能的数值：</p> <ul style="list-style-type: none"> <li>0 或 2，用于 connection_type = B#16#01 (主动端：0、被动端：2)</li> <li>0 或 2，用于 connection_type = B#16#11 (主动端：0 或 2，被动端：2)</li> <li>2 至 16，用于 connection_type = B#16#12</li> </ul>
8	rem_subnet_id_len	BYTE	B#16#00	该参数当前未使用。必须将 B#16#00 分配给它。
9	rem_staddr_len	BYTE	B#16#00	<p>远程连接终端的地址长度：</p> <ul style="list-style-type: none"> <li>0: 未指定，即参数 rem_staddr 不相关。</li> <li>4: 参数 rem_staddr 中有效的 IP 地址</li> </ul>

字节	参数	数据类型	起始值	描述
10	rem_tsap_id_len	BYTE	B#16#00	所使用的 rem_tsap_id 参数的长度；可能的值： <ul style="list-style-type: none"> <li>• 0 或 2，如果 connection_type = B#16#01 对于被动端，仅允许值 B#16#00。</li> <li>• 0 或 2，如果 connection_type = B#16#11</li> <li>• 0 或 2 至 16，如果 connection_type = B#16#12 (主动端：2 至 16；被动端：0 或 2 至 16，0 代表未指定)</li> </ul>
11	next_staddr_len	BYTE	B#16#00	使用了参数 next_staddr 的长度(仅与 ISO on TCP 相关)
12 到 27	local_tsap_id	BYTE 的 ARRAY [1..16]	B#16#07 B#16#D0 B#16#00 ...	如果 connection_type = <ul style="list-style-type: none"> <li>• B#16#11：本地端口号(可能的数值：请参见用于工业以太网上开放通讯的 FB 的功能)，local_tsap_id[1] = 以十六进制表示的端口号的低字节；local_tsap_id[2] = 以十六进制表示的端口号的高字节；local_tsap_id[3-16] = B#16#00</li> <li>• B#16#12：本地 TSAP 标识号：参见下文。</li> <li>• B#16#01：本地端口号(可能的数值：请参见用于工业以太网上开放通讯的 FB 的功能)，local_tsap_id[1] = 以十六进制表示的端口号的低字节；local_tsap_id[2] = 以十六进制表示的端口号的高字节；local_tsap_id[3-16] = B#16#00</li> </ul> 注意：若同一个通信伙伴存在多个连接，则确保所使用的每个 local_tsap_id 的数值在 CPU 上唯一。
28 到 33	rem_subnet_id	BYTE 的 ARRAY [1..6]	B#16#00 ...	该参数当前未使用。您必须将"0"分配给它。
34 到 39	rem_staddr	BYTE 的 ARRAY [1..6]	B#16#00 ...	远程连接端点的 IP 地址，192.168.002.003： <p>对于 connection_type =</p> <ul style="list-style-type: none"> <li>• B#16#1x：                             <ul style="list-style-type: none"> <li>rem_staddr[1] = B#16#C0 (192)，</li> <li>rem_staddr[2] = B#16#A8 (168)，</li> <li>rem_staddr[3] = B#16#02 (002)，</li> <li>rem_staddr[4] = B#16#03 (003)，</li> <li>rem_staddr[5-6]= B#16#00 (保留)</li> </ul> </li> <li>• B#16#01：                             <ul style="list-style-type: none"> <li>rem_staddr[1] = B#16#03 (003)，</li> <li>rem_staddr[2] = B#16#02 (002)，</li> <li>rem_staddr[3] = B#16#A8 (168)，</li> <li>rem_staddr[4] = B#16#C0 (192)，</li> <li>rem_staddr[5-6]= B#16#00 (保留)</li> </ul> </li> </ul>

字节	参数	数据类型	起始值	描述
40 到 55	rem_tsap_id	BYTE 的 ARRAY [1..16]	B#16#00 ...	<p>如果 connection_type =</p> <ul style="list-style-type: none"> <li>B#16#11 : 本地端口号可能的数值 : 请参见用于工业以太网上开放通讯的 FB 的功能), rem_tsap_id[1] = 以十六进制表示的端口号的高字节 ; rem_tsap_id[2] = 以十六进制表示的端口号的低字节 ; rem_tsap_id[3-16] = B#16#00</li> <li>B#16#12 : 远程 TSAP 标识号 : 参见下文</li> <li>B#16#01 : 本地端口号可能的数值 : 请参见用于工业以太网上开放通讯的 FB 的功能), local_tsap_id[1] = 以十六进制表示的端口号的低字节 ; local_tsap_id[2] = 以十六进制表示的端口号的高字节 ; local_tsap_id[3-16] = B#16#00</li> </ul>
56 到 61	next_staddr	BYTE 的 ARRAY [1..6]	B#16#00 ...	<p>如果 local_device_id =</p> <ul style="list-style-type: none"> <li>B#16#00 : next_staddr[1] : 相应(本地) CP 的机架和插槽 (位 0 至 4 : 插槽 , 位 5 至 7 : 机架号) next_staddr[2-6] : B#16#00</li> <li>B#16#02、B#16#03、B#16#05 : next_staddr[1-6] : B#16#00</li> <li>B#16#01、B#16#06、B#16#0B、B#16#0F : next_staddr[1-6] : B#16#00 (这种情况下 , next_staddr 为 len=B#16#00。)</li> </ul>
62 到 63	spare	WORD	W#16#0000	保留 : 分配 0 给该参数。

### 用于 ISO-on-TCP 的本地和远程 TSAP 的长度和结构

- 主动连接建立：
  - 本地 TSAP 标识号：
 

通过一个集成的 IE 接口，允许使用长度为 1-16 字节的任何字符串。每个字符串可使用位于 B#16#00 到 B#16#FF 之间的一个数值。
  - 远程 TSAP 标识号：
 

允许使用长度为 1-16 字节的任何字符串。每个字符串可使用位于 B#16#00 到 B#16#FF 之间的一个数值。
- 被动连接建立：
  - 本地 TSAP 标识号：

loc_tsap_id_len	local_tsap_id[1]	local_tsap_id[2]	local_tsap_id[3 - 16]
2	B#16#E0 (连接类型 T 型连接)	0 (仅限于集成的 IE 接口) 或本地 CPU 的机架和插槽 (位 0 - 4 插槽，位 5 - 7 机架号)	不存在
>2	B#16#E0 (连接类型 T 型连接)	0 (仅限于集成的 IE 接口) 或本地 CPU 的机架和插槽 (位 0 - 4 插槽，位 5 - 7 机架号)	TSAP 扩展
	仅限于集成的 IE 接口：一个 ASCII 字符(B#16#20 到 B#16#7E)	不相关	TSAP 扩展

- 远程 TSAP 标识号：
  - 允许使用长度为 0-16 字节的任何字符串。每个字符串可使用位于 B#16#00 到 B#16#FF 之间的一个数值。

### 协议类型 TCP 和 ISO on TCP 的 CPU 从属关系

哪个 CP 用哪种 CP 和 ISO-on-TCP 的说明参见 CPU 和所使用的协议类型(connection\_type)和可传送数据长度的关系。

欲了解关于可能建立的连接数的信息，请参考 CPU 的技术数据。

### 建立连接

主动连接的建立必须通过一个通信伙伴 A 启动。被动连接的建立必须通过一个通信伙伴 B 启动。如果两个通信伙伴都启动了连接建立，那么操作系统能够完整地建立一个连接。

在连接参数中，您需指定哪个通信伙伴激活建立连接、哪个通信伙伴根据通信伙伴的请求被动建立连接

在 UDP 中，两个伙伴必须启动建立被动连接。

## 24.4 为使用 UDP 的本地通讯接入点分配参数

### 用于为本地通信接入点分配参数的数据结构

要为本地通讯接入点分配参数，需要创建一个包含 UDT 65"TCON\_PAR"数据结构的 DB。此数据结构包含在用户程序和操作系统通讯层之间组态连接时所必需的参数。

FB 65 " TCON" 的 CONNECT 参数包含一个对关联连接描述的地址的引用(例如，P#DB100.DBX0.0 byte 64)。

### UDP (UDT 65)的连接结构描述

字节	参数	数据类型	起始值	描述
0 - 1	block_length	WORD	W#16#40	UDT 65 的长度：64 字节(固定)
2 到 3	id	WORD	W#16#0001	用户程序和操作系统通讯层之间该连接的标识号 (数值范围：W#16#0001 至 W#16#0FFF) 您必须在各个带有标识号的块中指定参数的值。
4	connection_type	BYTE	B#16#13	协议类型： • B#16#13：UDP
5	active_est	BOOL	FALSE	建立连接的方式的标识号：必须将 FALSE 分配给该参数。
6	local_device_id	BYTE	B#16#02	<ul style="list-style-type: none"> <li>B#16#00：通过 CP 443-1EX4x 进行通信(只用于 S7-400，且 connection_type = B#16#12)。允许的 CP：CP443-1EX4x、CP443-1EX20、CP443-1GX20、CP443-1EX30、CP443-1GX30)</li> <li>B#16#01：通过接口插槽 1 (IF1)中的 IE 接口与 WinAC RTX (仅 TCP)进行通信</li> <li>B#16#02：通过 CPU 315-2 PN/DP 和 317-2 PN/DP 的集成 IE 接口进行通信</li> <li>B#16#03：通过 CPU 319-3 PN/DP 上的集成 IE 接口进行通信</li> <li>B#16#05：通过集成 IE 接口与 CPU 414-3 PN/DP、416-3 PN/DP、416-3F PN/DP 和 41x-5H PN/DP (机架 0)进行通信</li> <li>B#16#06：通过接口插槽 2 (IF2)中的 IE 接口与 WinAC RTX (仅 TCP)进行通信</li> <li>B#16#0B。通过接口插槽 3 (IF3)中的 IE 接口与 WinAC RTX (仅 TCP)进行通信</li> <li>B#16#0F：通过接口插槽 4 (IF4)中的 IE 接口与 WinAC RTX (仅 TCP)进行通信</li> <li>B#16#15：通过 IE 接口与 CPU 41x-5H PN/DP (机架 1)进行通信</li> </ul>
7	local_tsap_id_len	BYTE	B#16#02	所使用的参数 local_tsap_id 的长度；2 个字节
8	rem_subnet_id_len	BYTE	B#16#00	该参数当前未使用。必须将 B#16#00 分配给它。

字节	参数	数据类型	起始值	描述
9	rem_staddr_len	BYTE	B#16#00	该参数当前未使用。必须将 B#16#00 分配给它。
10	rem_tsap_id_len	BYTE	B#16#00	该参数当前未使用。必须将 B#16#00 分配给它。
11	next_staddr_len	BYTE	B#16#00	该参数当前未使用。必须将 B#16#00 分配给它。
12 到 27	local_tsap_id	BYTE 的 ARRAY [1..16]	B#16#07 B#16#D0 B#16#00 ...	本地端口号(可能值：请参见用于工业以太网上开放通讯的 FB 的功能)， local_tsap_id[1] = 以十六进制表示的端口号的高字节； local_tsap_id[2] = 以十六进制表示的端口号的低字节； local_tsap_id[3-16] = B#16#00 (保留) 注意：请确保您在 CPU 中使用的每个 local_tsap_id 值都是唯一的。
28 到 33	rem_subnet_id	BYTE 的 ARRAY [1..6]	B#16#00 ...	该参数当前未使用。您必须将"0"分配给它。
34 到 39	rem_staddr	BYTE 的 ARRAY [1..6]	B#16#00 ...	该参数当前未使用。您必须将"0"分配给它。
40 到 55	rem_tsap_id	BYTE 的 ARRAY [1..16]	B#16#00 ...	该参数当前未使用。您必须将"0"分配给它。
56 到 61	next_staddr	BYTE 的 ARRAY [1..6]	B#16#00 ...	该参数当前未使用。您必须将"0"分配给它。
62 到 63	spare	WORD	W#16#0000	保留：分配 0 给该参数

### CPU 与 UDP 协议类型的关联

预知能使用 UDP 协议类型的 CPU ,请参阅 CPU 和所用的协议类型(connection\_type)和可传送数据长度之间的关系

关于用户程序和操作系统通讯层之间可能的连接数目，请参考 CPU 的技术数据。

### 组态本地通讯接入点

必须为每个通讯伙伴组态独立于其它伙伴的本地通讯点。这适用于在用户程序和操作系统通讯层之间建立连接。

在 UDP 中，两个连接伙伴必须触发被动连接建立。



## 24.5 使用 UDP 的远程通讯伙伴地址信息的结构

### 概述

对于 FB 67 "TUSEND", 在参数 ADDR 中传送接收器的地址信息。该地址信息必须是下文指定的结构。

对于 FB 68 "TURCV", 在参数 ADDR 中, 可以从接收到的数据中获取发送器的地址。该地址信息必须是下文指定的结构。

### 用于远程伙伴地址信息的数据块

必须创建一个 DB, 其中包含一个或多个依据 UDT 66 "TADDR\_PAR"建立的数据结构。

一个指向关联远程伙伴地址的指针将在 FB 67 "TUSEND"的参数 ADDR 中发送, 在 FB 68 "TURCV"的参数 ADDR 中接收(例如, P#DB100.DBX0.0 byte 8)。

### 远程通讯伙伴地址信息的结构(UDP 66)

字节	参数	数据类型	起始值	描述
0 至 3	rem_ip_addr	BYTE 的 ARRAY [1..4]	B#16#00 ...	远程伙伴的 IP 地址, 例如, 192.168.002.003 : <ul style="list-style-type: none"> <li>• rem_ip_addr[1] = B#16#C0 (192)</li> <li>• rem_ip_addr[2] = B#16#A8 (168)</li> <li>• rem_ip_addr[3] = B#16#02 (002)</li> <li>• rem_ip_addr[4] = B#16#03 (003)</li> </ul>
4 到 5	rem_port_nr	BYTE 的 ARRAY [1..2]	B#16#00 ...	远程端口号(可能值: 请参见用于工业以太网上开放通讯的 FB 的功能), <ul style="list-style-type: none"> <li>• rem_port_nr[1] = 端口编号(以十六进制表示)的高字节</li> <li>• rem_port_nr[2] = 端口编号(以十六进制表示)的低字节</li> </ul>
6 到 7	spare	BYTE 的 ARRAY [1..2]	B#16#00 ...	保留: 分配 0 给该参数

## 24.6 CPU 和所用的协议变量(connection\_type)以及可传送数据长度之间的关系

下表显示 CPU 可以使用的协议类型以及能够传送的数据长度。

协议类型	UDT 65 中的参数 "connection_type"	CPU	用于在集成式 IE 接口上通信的数据长度，以字节表示	用于在 CP 上进行通信的数据长度，以字节表示
TCP	B#16#11	CPU 31x-2 PN/DP，从固件版本 V2.4 开始	1 至 8192	-
		CPU 31x-2 PN/DP，从固件版本 V3.1 开始	1 至 32767	
		CPU 31x-3 PN/DP，从固件版本 V3.2 开始	1 至 32767	
		CPU 41x-3 PN/DP	1 至 32767	-
		CPU 41x-5H PN/DP	1 至 32767	
		CPU 41x-5H PN/DP	1 至 32767	
ISO-on-TCP	B#16#12	CPU 31x-2 PN/DP	1 至 1460	-
		CPU 31x-2 PN/DP，从固件版本 V2.4 开始	1 至 8192	-
		CPU 31x-2 PN/DP，从固件版本 V3.1 开始	1 至 32767	
		CPU 31x-3 PN/DP，从固件版本 V3.2 开始	1 至 32767	
		从固件版本 V4.1 起的 CPU 41x (无 CPU 41x-4H)	-	1 至 1452
		CPU 41x-3 PN/DP	1 至 32767	1 至 1452
		CPU 41x-3 PN/DP	1 至 32767	
		CPU 41x-5H PN/DP	1 至 32767	1 至 1452
		CPU 41x-5H PN/DP	1 至 32767	
UDP	B#16#13	CPU 31x-2 PN/DP，从固件版本 V2.4 开始，CPU 41x-3 PN/DP 和 41x-5H PN/DP	1 至 1472	-

## 24.7 使用 FB 65 "TCON"建立连接

欲了解关于可能建立的连接数的信息，请参考 CPU 的技术数据。

### 通过 TCP 和 ISO on TCP 使用

两个通信伙伴都调用 FB 65 "TCON"来设置和建立通信连接。在这些参数中，您要指定哪个伙伴是主动通信传输点、哪个伙伴是被动通信传输点。

设置并建立连接后，CPU 会自动对连接进行维护和监视。

如果连接中断，如由于线路中断或远程通信伙伴而造成的中断，主动伙伴会尝试重新建立已组态的连接。在这种情况下，不必再次调用 FB 65 "TCON"。

当调用 FB 66 "TDISCON"或 CPU 进入 STOP 模式时，现有连接将被终止并删除已组态的连接。要重新设置并再次建立该连接，必须再次调用 FB 65 "TCON"。

### 通过 UDP

通信伙伴的双方都调用 FB 65 "TCON"，用以组态其自身的本地通信接入点(本地端口)。将在用户程序和操作系统通讯层之间组态一个连接。不会与远程伙伴建立连接。

本地接入点用来发送和接收 UDP 消息帧。

当分配本地接入点(UDT65)的参数或将参数分配给远程伙伴(UDT66)时，下列参数必须相同。

- 对于 UDP：引用连接和本地端口号

### 功能

FB 65 "TCON"是一种以异步方式工作的 FB，这意味着它的作业处理要覆盖若干个 FB 调用。要启动作业来设置并建立连接，可用 REQ = 1 调用 FB 65。

输出参数 RET\_VAL 和 BUSY 中指示作业状态。STATUS 对应于异步功能 SFC 的输出参数 RET\_VAL (另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

下表显示了 BUSY (忙碌)、DONE (完成)和 ERROR (出错)之间的关系。使用该表格，可以确定 FB 65 的当前状态或是否(为 TCP 和 ISO on TCP)设置并建立了连接。

BUSY	DONE	ERROR	描述
TRUE	不相关	不相关	作业正在处理。
FALSE	TRUE	FALSE	该作业成功完成。
FALSE	FALSE	TRUE	该作业以出错而结束。出错原因可在 STATUS 参数中找到。
FALSE	FALSE	FALSE	没有给 FB 分配(新)作业。

## 参数

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数 REQUEST 启动建立由 ID 指定的连接的作业。作业在上升沿启动。
ID	INPUT	WORD	M、D、常数	与远程伙伴之间建立的连接或用户程序和操作系统通讯层之间建立的连接的标识号。标识号必须与本地连接描述中的相关参数标识号相同。 取值范围：W#16#0001 至 W#16#0FFF
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： <ul style="list-style-type: none"> <li>0: 作业尚未开始或仍在运行。</li> <li>1: 无错执行作业。</li> </ul>
BUSY	OUTPUT	BOOL	I、Q、M、D、L	<ul style="list-style-type: none"> <li>BUSY = 1：作业尚未完成。</li> <li>BUSY=0：作业完成。</li> </ul>
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数： <ul style="list-style-type: none"> <li>ERROR = 1：处理作业期间出现错误。STATUS 返回有关错误类型的详细信息</li> </ul>
STATUS	OUTPUT	WORD	M、D	STATUS 状态参数：错误信息
CONNECT	IN_OUT	ANY	D	指向相关联的连接描述(UDT 65)的指针，参见为 TCP 和 ISO-on-TCP 通信连接分配参数和为使用 UDP 的本地通讯接入点分配参数 注意：对于 S7-300 CPU，请注意，参数 CONNECT 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。

错误信息

ERROR	STATUS (W#16#...)	解释
0	0000	成功建立连接
0	7000	没有正在处理的作业
0	7001	启动作业处理，建立连接
0	7002	后续调用(与 REQ 不相关)，正在建立连接
1	8086	ID 参数超出允许的范围。
1	8087	已达最大连接数，无法再建立更多连接
1	8089	CONNECT 参数未指向数据块。
1	809A	CONNECT 参数指向的区域与连接描述(UDT65)的长度不匹配。
1	809B	连接说明中的 local_device_id 与 CPU 不匹配或相关接口锁定。
1	80A0	用于错误代码 W#16#80A1 和 W#16#80A2 的组错误
1	80A1	连接或端口已经由用户占用
1	80A2	本地或远程端口已经由系统占用
1	80A3	正尝试重新建立现有连接
1	80A4	远程连接端点的 IP 地址无效，它可能与本地 IP 地址相匹配
1	80A7	通讯错误：TCON 尚未完成便调用了 TDISCON。TDISCON 必须先完全终止 ID 引用的连接。
1	80B2	CONNECT 参数指向的数据块是通过关键字 UNLINKED 生成的。
1	80B3	不一致的参数分配：用于错误代码 W#16#80A0 到 W#16#80A2、W#16#80A4、W#16#80B4 到 W#16#80B9 的组错误
1	80B4	当使用协议变量 ISO on TCP (connection_type = B#16#12)进行被动连接建立 (active_est = FALSE)时，违反了下列一个或两个条件： <ul style="list-style-type: none"> <li>• local_tsap_id_len &gt;= B#16#02</li> <li>• local_tsap_id[1] = B#16#E0</li> <li>• 当 local_tsap_id_len &gt;= B#16#03 时，local_tsap_id[1]是一个 ASCII 字符 local_tsap_id[1]是一个 ASCII 字符，且 local_tsap_id_len &gt;= B#16#03</li> </ul>
1	80B5	UDP 协议变量中的 active_est 参数(UDT 65)出错
1	80B6	与 connection_type 参数(UDT 65)相关的参数分配错误
1	80B7	UDT 65 的下列其中一个参数出错：block_length、local_tsap_id_len、rem_subnet_id_len、rem_staddr_len、rem_tsap_id_len、next_staddr_len
1	80B8	本地连接描述(UDT 65)中的参数 ID 和参数 ID 不同
1	80C3	正在使用所有连接资源。
1	80C4	临时通讯错误： <ul style="list-style-type: none"> <li>• 此时无法建立连接。</li> <li>• 接口正在接收新参数。</li> <li>• TDISCON 当前正在删除已组态的连接。</li> <li>• H 系统正在连接和更新。</li> </ul>
1	8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 24.8 使用 FB 66 "TDISCON"终止连接

### 通过 TCP 和 ISO on TCP 使用

FB 66 "TDISCON"终止 CPU 与通讯伙伴的通讯连接。

### 通过 UDP

FB 66 "TDISCON"关闭本地通讯接入点。用户程序和操作系统通讯层之间的连接被终止。

### 功能

FB 66 "TDISCON"是一种以异步方式工作的 FB，这意味着它的作业处理要覆盖若干个 FB 调用。为了开始终止连接作业，用 REQ = 1 调用 FB 66。

在成功调用 FB 66 "TDISCON"之后，为 FB 65"TCON"指定的 ID 不再有效，从而不能将其用于发送或接收。

输出参数 RET\_VAL 和 BUSY 中指示作业状态。STATUS 对应于异步功能 SFC 的输出参数 RET\_VAL (另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

下表显示了 BUSY (忙碌)、DONE (完成)和 ERROR (出错)之间的关系。通过该表可以确定 FB 66 的当前状态，或何时完成建立连接。

BUSY	DONE	ERROR	描述
TRUE	不相关	不相关	作业正在处理。
FALSE	TRUE	FALSE	该作业成功完成。
FALSE	FALSE	TRUE	该作业以出错而结束。出错原因可在 STATUS 参数中找到。
FALSE	FALSE	FALSE	没有给 FB 分配(新)作业。

## 参数

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数 REQUEST 启动终止由 ID 指定的连接的作业。作业在上升沿上启动。
ID	INPUT	WORD	M、D、常数	将与远程伙伴终止的连接或用户程序和操作系统通讯层之间的连接的标识号。标识号必须与本地连接描述中的相关参数标识号相同。 取值范围：W#16#0001 至 W#16#0FFF
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： <ul style="list-style-type: none"> <li>0: 作业尚未开始或仍在运行。</li> <li>1: 无错执行作业。</li> </ul>
BUSY	OUTPUT	BOOL	I、Q、M、D、L	<ul style="list-style-type: none"> <li>BUSY = 1：作业尚未完成。</li> <li>BUSY=0：作业完成。</li> </ul>
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数： <ul style="list-style-type: none"> <li>ERROR = 1：处理时出错。STATUS 提供有关错误类型的详细信息</li> </ul>
STATUS	OUTPUT	WORD	M、D	STATUS 状态参数：错误信息

## 错误信息

ERROR	STATUS (W#16#...)	解释
0	0000	成功终止连接
0	7000	没有正在处理的作业
0	7001	开始处理作业，正在终止连接
0	7002	后续调用(与 REQ 不相关)，正在终止连接
1	8086	ID 参数不在允许的地址范围内
1	80A3	尝试终止的连接不存在
1	80C4	临时通讯错误：接口正在接收新参数，或当前正在建立连接，或 H 系统正在连接和更新..

## 24.9 使用 FB 63 "TSEND"通过 TCP 和 ISO on TCP 发送数据

### 描述

FB 63 "TSEND"通过现有的通讯连接发送数据。

### 功能

FB 63 "TSEND"异步工作；换句话说，它的处理需要多个 FB 调用。要启动发送数据，请使用 REQ = 1 调用 FB 63。

输出参数 BUSY 和 STATUS 中指示作业状态。STATUS 对应于异步功能 SFC 的输出参数 RET\_VAL (另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

如果待发送的数据已全部传输至本地接口，则设置输出参数 DONE。

下表显示了 BUSY (忙碌)、DONE (完成)和 ERROR (出错)之间的关系。通过该表可以确定 FB 63 的当前状态。

BUSY	DONE	ERROR	描述
TRUE	不相关	不相关	作业正在处理。
FALSE	TRUE	FALSE	该作业成功完成。
FALSE	FALSE	TRUE	该作业以出错而结束。出错原因可在 STATUS 参数中找到。
FALSE	FALSE	FALSE	没有给 FB 分配(新)作业。

### 注意

由于 FB 63 "TSEND"的异步功能，在 DONE 参数或 ERROR 参数的值变为 TRUE 之前，必须保持发送器区域中数据不变。



## 参数

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数 REQUEST 在上升沿开始发送作业。 数据从由 DATA 和 LEN 指定的区域传送。
ID	INPUT	WORD	M、D、常数	将终止对连接的引用。标识号必须与本地连接描述中的相关参数标识号相同。 取值范围：W#16#0001 至 W#16#0FFF
LEN	INPUT	INT	I、Q、M、D、L	要使用作业发送的最大字节数 参见CPU 和所用的协议变量(connection_type) 以及可传送数据长度之间的关系
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： <ul style="list-style-type: none"> <li>0: 作业尚未开始或仍在运行。</li> <li>1: 待发送的数据已全部传输至本地接口。</li> </ul>
BUSY	OUTPUT	BOOL	I、Q、M、D、L	<ul style="list-style-type: none"> <li>BUSY = 1：作业尚未完成。无法触发新作业。</li> <li>BUSY=0：作业完成。</li> </ul>
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数： <ul style="list-style-type: none"> <li>ERROR = 1：处理时出错。STATUS 提供有关错误类型的详细信息</li> </ul>
STATUS	OUTPUT	WORD	M、D	STATUS 状态参数：错误信息
DATA	IN_OUT	ANY	I、Q、M、D	发送区域包含地址和长度地址指的是： <ul style="list-style-type: none"> <li>输入过程映像</li> <li>输出过程映像</li> <li>存储器位</li> <li>数据块</li> </ul> 注意：不要使用 BOOL 类型的 ARRAY 作为发送区域。 注意：对于 S7-300 CPU，请注意，参数 DATA 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。

## 错误信息

ERROR	STATUS (W#16#...)	解释
0	0000	发送作业无错完成
0	7000	没有正在处理的作业
0	7001	开始处理作业，正在发送数据 注意：在处理期间，操作系统将访问 DATA 发送区中的数据。
0	7002	后续调用(与 REQ 不相关)，正在处理作业 注意：在处理期间，操作系统将访问 DATA 发送区中的数据。
1	8085	LEN 参数的值为"0"或比最大的允许值大
1	8086	ID 参数不在允许的地址范围内
1	8088	LEN 参数大于在 DATA 中指定的存储器区
1	8092	仅限 S7-300: DATA 中的无效数据类型
1	80A1	通讯错误： <ul style="list-style-type: none"> <li>• 尚未建立指定的连接</li> <li>• 当前正在终止指定的连接。无法通过此连接进行传送。</li> <li>• 正在重新初始化接口。</li> </ul>
1	80B7	长度错误：在 ADDR 参数中长度指定为小于 8 字节。
1	80B3	连接类型参数(连接描述中的 connection_type 参数)被设置为 UDP。请使用 FB 67 "TUSEND"。
1	80C3	<ul style="list-style-type: none"> <li>• 具有该 ID 的块正在以不同的优先级进行处理。</li> <li>• 内部资源缺乏</li> </ul>
1	80C4	临时通讯错误： <ul style="list-style-type: none"> <li>• 此时无法建立与通信伙伴的连接。</li> <li>• 接口正在接收新参数，或目前正在建立连接</li> </ul>
1	8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 24.10 使用 FB 64 "TRCV"通过 TCP 和 ISO on TCP 接收数据

### 描述

FB 64 "TRCV"通过现有通讯连接接收数据。

### 接收区

它指的是功能块输入所接收数据的区域。

接收区由下列两个变量指定：

- 指向区域起始地址的指针
- 区域长度

区域长度由 LEN 参数使用的协议类型指定(如果 LEN  $\neq$  0)或 DATA 参数长度信息指定(如果 LEN = 0)。

### FB 64 "TRCV"的接收模式

下表显示出 FB 64 如何在接收区中输入所接收的数据。

协议类型	在接收区中输入数据	参数"connection_type"	LEN 参数的数值
TCP	Ad-hoc 模式	B#16#01、B#16#11	0
TCP	接收具有预设长度的数据	B#16#01、B#16#11	$\neq$ 0
ISO-on-TCP	协议受控	B#16#12	0 (建议)或 $\neq$ 0

### TCP / Ad-hoc 模式

ad-hoc 模式仅对于 TCP 协议类型存在。通过将 0 分配给 LEN 参数设置 ad-hoc 模式。

接收区与由 DATA 形成的区域相同。最多可接收 8192 个字节。

一接收到数据块，FB64 就在接收区中输入数据并将 NDR 设置为 1。

### TCP / 接收具有预设长度的数据

通过将非 0 数值赋值给 LEN 参数，可设置接收具有预设长度的数据。

接收区由 LEN 和 DATA 参数定义。

一收到 LEN 字节，FB64 就将它们传送给接收区，并将 NDR 设置为 1。

### ISO-on-TCP / 协议受控的数据传输

使用 ISO-on-TCP 协议类型，数据传送时将受协议控制。

接收区由 LEN 和 DATA 参数定义。

—收到作业数据，FB64 就将它传送给接收区，并将 NDR 设置为 1。

### 替换 STRING 数据类型的数据

如果想要使用 FB "TSEND"和"TRCV"替换 STRING 数据类型的数据，仅可以使用 Ad-hoc 模式 LEN=0。还必须将 FB "TRCV"的字符串长度至少选为与 FB "TSEND"的字符串长度一样长。如果未遵循这一条规则，可能会导致进一步处理字符串时发生错误。

如果使用 BYTE 类型的 ARRAY 作为要发送字符串的接收区域，第一个字节包含由发送方设置的最大长度，第二个字节包含字符串的当前长度。

### 功能

FB 64 "TRCV"是一种以异步方式工作的 FB，这意味着它的处理要覆盖若干个 FB 调用。为了启动接收数据作业，通过 EN\_R = 1 调用 FB 64。

输出参数 BUSY 和 STATUS 中指示作业状态。STATUS 对应于异步功能 SFC 的输出参数 RET\_VAL (另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

下表显示了 BUSY (忙碌)、DONE (完成)和 ERROR (出错)之间的关系。使用该表，可以确定 FB 64 的当前状态，或者接收过程何时完成。

BUSY	DONE	ERROR	描述
TRUE	不相关	不相关	作业正在处理。
FALSE	TRUE	FALSE	该作业成功完成。
FALSE	FALSE	TRUE	该作业以出错而结束。出错原因可在 STATUS 参数中找到。
FALSE	FALSE	FALSE	没有给 FB 分配(新)作业。

### 注意

由于 FB 64 "TRCV"的异步功能，只有当 NDR 参数的值变为 TRUE 时，接收器区域中的数据才会变得一致。

## 参数

参数	声明	数据类型	存储器区	描述
EN_R	INPUT	BOOL	I、Q、M、D、L	使能接收的控制参数：EN_R = 1 时，FB 64 "TRCV"准备接收。正在处理接收作业。
ID	INPUT	WORD	M、D、常数	将终止对连接的引用。标识号必须与本地连接描述中的相关参数标识号相同。 取值范围：W#16#0001 至 W#16#0FFF
LEN	INPUT	INT	I、Q、M、D、L	接收区域的长度(以字节为单位) 关于 LEN = 0 或 LEN <> 0 的含义，请参见上文 (FB64 "TRCV"的接收模式)。 对于值的范围，请参见CPU 和所用的协议变量 (connection_type)以及可传送数据长度之间的关系。
NDR	OUTPUT	BOOL	I、Q、M、D、L	NDR 状态参数： <ul style="list-style-type: none"> <li>NDR = 0：作业尚未开始或仍在运行。</li> <li>NDR = 1：作业成功完成</li> </ul>
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数： <ul style="list-style-type: none"> <li>ERROR = 1：处理时出错。STATUS 提供有关错误类型的详细信息</li> </ul>
BUSY	OUTPUT	BOOL	I、Q、M、D、L	<ul style="list-style-type: none"> <li>BUSY = 1：作业尚未完成。无法触发新作业。</li> <li>BUSY=0：作业完成。</li> </ul>
STATUS	OUTPUT	WORD	M、D	STATUS 状态参数：错误信息
RCVD_LEN	OUTPUT	INT	I、Q、M、D、L	实际接收到的数据量(字节)
DATA	IN_OUT	ANY	I、Q、M、D	接收区域(定义见上文)包含地址和长度 地址指的是： <ul style="list-style-type: none"> <li>输入过程映像</li> <li>输出过程映像</li> <li>存储器位</li> <li>数据块</li> </ul> 注意：不要使用 BOOL 类型的 ARRAY 作为接收区域。 注意：对于 S7-300 CPU，请注意，参数 DATA 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。

## 错误信息

ERROR	STATUS (W#16#...)	解释
0	0000	接受了新数据。RCVD_LEN 中显示接收到的数据的当前长度。
0	7000	块未准备接收
0	7001	块准备接收，接收作业已经启动。
0	7002	后续的调用，正在处理的接收作业 注意：在该处理期间，FB 将数据写入接收区。因此，错误将导致接收区中的数据不一致。
1	8085	LEN 参数大于最大允许值，或从第一次调用之后已更改 LEN 或 DATA 参数。
1	8086	ID 参数不在允许的地址范围内
0	8088	<ul style="list-style-type: none"> <li>接收区太小</li> <li>LEN 中的数值大于由 DATA 指定的接收区。</li> </ul>
1	8092	<ul style="list-style-type: none"> <li>仅限 S7-300: DATA 中的无效数据类型</li> </ul>
1	80A1	通讯错误： <ul style="list-style-type: none"> <li>尚未建立指定的连接</li> <li>当前正在终止指定的连接。通过该连接无法执行接收作业。</li> <li>接口正在接收新参数。</li> </ul>
1	80B3	协议类型参数(连接描述中的 connection_type 参数)被设置为 UDP。请使用 FB 68 "TRCV"。
1	80C3	<ul style="list-style-type: none"> <li>具有该 ID 的块正在以不同的优先级进行处理。</li> <li>缺乏内部资源。</li> </ul>
1	80C4	临时通讯错误： <ul style="list-style-type: none"> <li>此时无法建立与伙伴的连接。</li> <li>接口正在接收新参数设置，或目前正在建立连接。</li> </ul>
1	8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 24.11 使用 FB 67 "TUSEND"通过 UDP 发送数据

### 描述

FB 67 "TUSEND"通过 UDP 将数据发送到参数 ADDR 指定的远程伙伴。



#### 警告

在使用 UDP 根据 RFC 768 传送数据时，不经过确认就将数据传送给远程伙伴，因此是不可靠传送。这意味着数据会丢失，且块不会进行指示。

#### 注意

在依次将单独数据发送到不同的伙伴时，只需要在调用 FB 67 "TUSEND"时调整参数 ADDR。不必重新调用 FB 65 "TCON"和 66 "TDISCON"。

### 功能

FB 67 "TUSEND"是一种以异步方式工作的 FB，这意味着它的作业处理要覆盖若干个 FB 调用。为了开始发送数据作业，用 REQ = 1 调用 FB 67。

输出参数 BUSY 和 STATUS 中指示作业状态。STATUS 对应于异步功能 SFC 的输出参数 RET\_VAL (另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

下表显示了 BUSY (忙碌)、DONE (完成)和 ERROR (出错)之间的关系。使用该表，可以确定 FB 67 的当前状态，或者发送过程何时完成。

BUSY	DONE	ERROR	描述
TRUE	不相关	不相关	作业正在处理。
FALSE	TRUE	FALSE	该作业成功完成。
FALSE	FALSE	TRUE	该作业以出错而结束。出错原因可在 STATUS 参数中找到。
FALSE	FALSE	FALSE	没有给 FB 分配(新)作业。

#### 注意

由于 FB 67 "TUSEND"的异步功能，在 DONE 参数或 ERROR 参数的值变为 TRUE 之前，必须保持发送器区域中数据不变。

## 参数

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数 REQUEST 在上升沿开始发送作业。 数据从由 DATA 和 LEN 指定的区域传送。
ID	INPUT	WORD	M、D、常数	用户程序和操作系统通讯层之间的关联连接的标识号。标识号必须与本地连接描述中的相关参数标识号相同。 取值范围：W#16#0001 至 W#16#0FFF
LEN	INPUT	INT	I、Q、M、D、L	要通过作业发送的字节数 取值范围：1 至 1472
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数： <ul style="list-style-type: none"> <li>0: 作业尚未开始或仍在运行。</li> <li>1: 无错执行作业。</li> </ul>
BUSY	OUTPUT	BOOL	I、Q、M、D、L	<ul style="list-style-type: none"> <li>BUSY = 1：作业尚未完成。无法触发新作业。</li> <li>BUSY=0：作业完成。</li> </ul>
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数： <ul style="list-style-type: none"> <li>ERROR = 1：处理时出错。STATUS 提供有关错误类型的详细信息</li> </ul>
STATUS	OUTPUT	WORD	M、D	STATUS 状态参数：错误信息
DATA	IN_OUT	ANY	I、Q、M、D	发送器区，包含地址和长度 地址指的是： <ul style="list-style-type: none"> <li>过程映像输入表</li> <li>过程映像输出表</li> <li>位存储器</li> <li>数据块</li> </ul> 注意：对于 S7-300 CPU，请注意，参数 DATA 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。
ADDR	IN_OUT	ANY	D	指向接收器地址的指针(例如，P#DB100.DBX0.0 byte 8)，参见使用 UDP 的远程通讯伙伴地址信息的结构 注意：对于 S7-300 CPU，请注意，参数 ADDR 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。



## 错误信息

ERROR	STATUS (W#16#...)	解释
0	0000	发送作业无错完成
0	7000	没有正在处理的作业
0	7001	开始处理作业，正在发送数据 注意：在处理期间，操作系统将访问 DATA 发送区中的数据。
0	7002	后续调用(与 REQ 不相关)，正在处理作业 注意：在处理期间，操作系统将访问 DATA 发送区中的数据。
1	8085	LEN 参数的值为"0"或比最大的允许值大
1	8086	ID 参数不在允许的地址范围内
0	8088	LEN 参数大于在 DATA 中指定的存储区
1	8089	ADDR 参数未指向数据块
1	80A1	通讯错误： <ul style="list-style-type: none"> <li>所指定的用户程序和操作系统通信层之间的连接尚未建立。</li> <li>用户程序和操作系统通讯层之间的指定连接当前已被终止。无法通过此连接进行传送。</li> <li>正在重新初始化接口。</li> </ul>
1	80A4	远程连接端点的 IP 地址无效；它可能与本地 IP 地址相匹配。
1	80B3	<ul style="list-style-type: none"> <li>设定的协议类型(连接描述中的 connection_type 参数)不是 UDP。请使用 FB 63 "TSEND"。</li> <li>ADDR 参数：端口号的无效设置</li> </ul>
1	80C3	<ul style="list-style-type: none"> <li>具有该 ID 的块正在以不同的优先级进行处理。</li> <li>缺乏内部资源。</li> </ul>
1	80C4	临时通讯错误： <ul style="list-style-type: none"> <li>此刻无法建立用户程序和操作系统通讯层之间的连接。</li> <li>接口正在接收新参数。</li> </ul>
1	8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 24.12 使用 FB 68 "TURCV"通过 UDP 接收数据

### 描述

FB 68 "TURCV"通过 UDP 接收数据。在成功完成 FB 68 "TURCV"之后，参数 ADDR 将显示远程伙伴 (发送器)的地址。



#### 警告

在使用 UDP 根据 RFC 768 传送数据时，不经过确认就将数据传送给远程伙伴，因此是不可靠传送。这意味着数据会丢失，且块不会进行指示。

### 功能

FB 68 "TURCV"是一种以异步方式工作的 FB，这意味着它的作业处理要覆盖若干个 FB 调用。为了启动接收数据作业，通过 EN\_R = 1 调用 FB 68。

输出参数 RET\_VAL 和 BUSY 中指示作业状态。STATUS 对应于异步功能 SFC 的输出参数 RET\_VAL (另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

下表显示了 BUSY (忙碌)、DONE (完成)和 ERROR (出错)之间的关系。使用该表，可以确定 FB 68 的当前状态，或者接收过程何时完成。

BUSY	DONE	ERROR	描述
TRUE	不相关	不相关	作业正在处理。
FALSE	TRUE	FALSE	该作业成功完成。
FALSE	FALSE	TRUE	该作业以出错而结束。出错原因可在 STATUS 参数中找到。
FALSE	FALSE	FALSE	没有给 FB 分配(新)作业。

#### 注意

由于 FB 68 "TURCV"的异步功能，只有当 NDR 参数的值变为 TRUE 时，接收器区域中的数据才会变得一致。

## 参数

参数	声明	数据类型	存储器区	描述
EN_R	INPUT	BOOL	I、Q、M、D、L	使能接收的控制参数：EN_R = 1 时，FB 68 "TURCV"准备接收。正在处理接收作业。
ID	INPUT	WORD	M、D、常数	用户程序和操作系统通讯层之间的关联连接的标识号。标识号必须与本地连接描述中的相关参数标识号相同。 取值范围：W#16#0001 至 W#16#0FFF
LEN	INPUT	INT	I、Q、M、D、L	关于接收区域的长度(以字节为单位)，请参见用于工业以太网上开放通讯的 FB 的功能：1 至 1472
NDR	OUTPUT	BOOL	I、Q、M、D、L	NDR 状态参数： <ul style="list-style-type: none"> <li>NDR = 0：作业尚未开始或仍在运行。</li> <li>NDR = 1：作业成功完成</li> </ul>
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数： <ul style="list-style-type: none"> <li>ERROR = 1：处理时出错。STATUS 提供有关错误类型的详细信息</li> </ul>
BUSY	OUTPUT	BOOL	I、Q、M、D、L	<ul style="list-style-type: none"> <li>BUSY = 1：作业尚未完成。无法触发新作业。</li> <li>BUSY=0：作业完成。</li> </ul>
STATUS	OUTPUT	WORD	M、D	STATUS 状态参数：错误信息
RCVD_LEN	OUTPUT	INT	I、Q、M、D、L	实际接收到的数据量(字节)
DATA	IN_OUT	ANY	I、Q、M、D	接收定义，参见用于工业以太网上开放通讯的 FB 的功能 地址参照： <ul style="list-style-type: none"> <li>过程映像输入表</li> <li>过程映像输出表</li> <li>存储器位</li> <li>数据块</li> </ul> 注意：对于 S7-300 CPU，请注意，参数 DATA 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。
ADDR	IN_OUT	ANY	D	指向发送器地址的指针(例如，P#DB100.DBX0.0 byte 8)，参见使用 UDP 的远程通讯伙伴地址信息的结构 注意：对于 S7-300 CPU，请注意，参数 ADDR 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。

## 错误信息

ERROR	STATUS (W#16#...)	解释
0	0000	接受了新数据。RCVD_LEN 中显示接收到的数据的当前长度。
0	7000	块未准备接收
0	7001	块准备接收，接收作业已经启动
0	7002	后续的调用，正在处理的作业 注意：在该处理期间，FB68 将数据写入接收区。因此，错误将导致接收区中的数据不一致。
1	8085	LEN 参数大于最大允许值，或从第一次调用起已更改 LEN 或 DATA 参数的值。
1	8086	ID 参数不在允许的地址范围内
1	8088	<ul style="list-style-type: none"> <li>接收区太小。</li> <li>LEN 中的数值大于由 DATA 指定的接收区。</li> </ul>
1	8089	ADDR 参数未指向数据块
1	80A1	通讯错误： <ul style="list-style-type: none"> <li>所指定的用户程序和操作系统通信层之间的连接尚未建立。</li> <li>用户程序和操作系统通信层之间的指定连接当前已被终止。通过该连接无法执行接收作业。</li> <li>正在重新初始化接口。</li> </ul>
1	80B3	设定的协议类型(连接描述中的 connection_type 参数)不是 UDP。请使用 FB 64 "TRCV"。
1	80C3	<ul style="list-style-type: none"> <li>具有该 ID 的块正在以不同的优先级进行处理。</li> <li>缺乏内部资源。</li> </ul>
1	80C4	临时通讯错误：当前正在重新启动连接。
1	8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 24.13 通过 FB 210 "FW\_TCP"使用通过 TCP 的 FETCH 和 WRITE 服务链接第三方系统

### 描述

FB 210 "FW\_TCP"提供像服务器一样的 FETCH 和 WRITE 服务。需要为 FETCH 服务和 WRITE 服务分别建立连接。这表示需要为每种服务单独调用 FB 210 ,并在 CONNECT 参数中指定相应的被动 TCP 连接的描述。

### 参数

参数	声明	数据类型	存储区	描述
ENABLE	INPUT	BOOL	I、Q、M、D、L	该参数启动连接的建立或终止 : <ul style="list-style-type: none"> <li>0 = 正在终止连接</li> <li>1 = 正在建立连接</li> </ul>
CONNECT	INPUT	ANY	D	指向相应的被动 TCP 连接描述的指针(UDT 65)
ADDRMODE	INPUT	INT	I、Q、M、D、L	通过该参数，可以指定如何解释数据请求的结构 : <ul style="list-style-type: none"> <li>0 = S7 地址模式(面向字节)</li> <li>1 = S5 地址模式(面向字)</li> </ul>
NDR	OUTPUT	BOOL	I、Q、M、D、L	该参数表示成功接受了 WRITE 作业的数据，生成并发送了确认帧。
ERROR	OUTPUT	BOOL	I、Q、M、D、L	在建立或终止连接，或在发送或接收数据时，如果出现错误，则该参数置位。
MODE	OUTPUT	BYTE	I、Q、M、D、L	该参数显示是否正在执行 FETCH 或 WRITE 作业 : <ul style="list-style-type: none"> <li>0 = 无激活的作业</li> <li>1 = WRITE 作业</li> <li>2 = FETCH 作业</li> </ul>
状态	OUTPUT	WORD	M、D	内部通信状态位在 STATUS 参数中输出。

### 更多信息

关于 FB 210 "FW\_TCP"的详细信息，请参见 STEP 7 DVD 光盘的 DVD\_Root\Optional Components\Communication Blocks，文件"FetchWrite\_PNCPU\_x.pdf" (x=a 为德语，x=b 为英语)。

## 24.14 通过 FB FB 220 "FW\_IOT"使用通过 ISO-on-TCP 的 FETCH 和 WRITE 服务链接第三方系统

### 描述

FB 220 "FW\_IOT"提供像服务器一样的 FETCH 和 WRITE 服务。需要为 FETCH 服务和 WRITE 服务分别建立连接。这表示需要为每种服务单独调用 FB 220，并在 CONNECT 参数中指定相应的被动 ISO-on-TCP 连接的描述。

### 参数

参数	声明	数据类型	存储区	描述
ENABLE	INPUT	BOOL	I、Q、M、D、L	该参数启动连接的建立或终止： <ul style="list-style-type: none"> <li>• 0 = 正在终止连接</li> <li>• 1 = 正在建立连接</li> </ul>
CONNECT	INPUT	ANY	D	指向相应的被动 ISO-on-TCP 连接描述的指针 (UDT 65)
ADDRMODE	INPUT	INT	I、Q、M、D、L	通过该参数，可以指定如何解释数据请求的结构： <ul style="list-style-type: none"> <li>• 0 = S7 地址模式(面向字节)</li> <li>• 1 = S5 地址模式(面向字)</li> </ul>
NDR	OUTPUT	BOOL	I、Q、M、D、L	该参数表示成功接受了 WRITE 作业的数据，生成并发送了确认帧。
ERROR	OUTPUT	BOOL	I、Q、M、D、L	在建立或终止连接，或在发送或接收数据时，如果出现错误，则该参数置位。
MODE	OUTPUT	BYTE	I、Q、M、D、L	该参数显示是否正在执行 FETCH 或 WRITE 作业： <ul style="list-style-type: none"> <li>• 0 = 无激活的作业</li> <li>• 1 = WRITE 作业</li> <li>• 2 = FETCH 作业</li> </ul>
状态	OUTPUT	WORD	M、D	内部通信状态位在 STATUS 参数中输出。

### 更多信息

关于 FB 220 "FW\_IOT"的详细信息，请参见 STEP 7 DVD 光盘的 DVD\_Root\Optional Components\Communication Blocks，文件"FetchWrite\_PNCPU\_x.pdf" (x=a 为德语，x=b 为英语)。

## 25 产生与块有关的消息

### 25.1 关于使用 SFB 生成块相关消息的介绍

#### 用于生成块相关消息的 SFB

可以在程序中通过调用以下 SFB 中的一个生成块相关消息：

- SFB 36 "NOTIFY"
- SFB 31 "NOTIFY\_8P"
- SFB 33 "ALARM"
- SFB 34 "ALARM\_8"
- SFB 35 "ALARM\_8P"

上述 SFB 具有以下属性：

- 除 SFB 36 "NOTIFY"和 31 "NOTIFY\_8P"以外，调用块时所有检测到的 0 -> 1 或 1 -> 0 信号转变都会引起消息传送。
- 如果设置了缺省报告过程(禁用了确认触发报告)，则在调用块时如果检测到任何信号转变，SFB 33 "ALARM"、34 "ALARM\_8"和 35 "ALARM\_8P"也将生成一条消息。  
相反，如果启用了确认触发报告，则并非所有信号转变都会触发消息传送(有关详细信息，请参见下文)。
- 块执行之后，将完整读取关联值(输入 SD\_i)，并将其分配给消息(请参见用于 S7 通讯的 SFB/FB 和 SFC/FC 的公用参数)。  
在一致性方面，与高优先级类相比，以下关联值是一致的：

每个关联值 SD\_i 都具有固有的一致性。

可以通过状态参数 DONE、ERROR 和 STATUS 监视块的处理状态(请参见 SFB/FB 的公用参数及 S7 通讯的 SFC/FC 的公用参数中的"状态参数")。

---

#### 注意

只有在初次调用块(实例的实际参数或已定义值)时才会对参数 ID 和 EV\_ID 进行判断。

---

#### 登录显示设备

要使用于生成块相关消息的 SFB 能够在检测到信号转变时发送消息，事先必须至少登录一台用于显示块相关消息的显示设备(例如，WinCC)。如果没有登录，参数 STATUS = 1。

## 信号转变检测

消息块的每个实例可以使用一个包含两个存储器块的消息存储器。

首先将对消息存储器清零。将在输入端 SIG 或 SIG\_1 至 SIG\_8 的其中一个输入端的信号转变输入到第一个存储器块中。该存储器块将保持占用状态，直到相应消息传送完毕。

然后将在输入端 SIG 或 SIG\_1 至 SIG\_8 的其中一个输入端检测到的信号转变输入到第二个存储器块中。如果第一个块仍处于占用状态，在任何情况下均将覆盖第二个存储器块。

输出参数 ERROR 和 STATUS 将指示这一消息丢失情况。该信息还会与下一个可以传送的消息一并传递给在线显示设备。

清空第一个存储器块后，将把第二个存储器块的数据传递给第一个块。这样，第二个存储器块再次被清空。

## 确认触发报告

将 SFB 33 "ALARM"、34 "ALARM\_8"和 35 "ALARM\_8P"与确认触发报告一起使用，可以减少系统的消息通讯量。

也就是说，在初次生成信号的进入消息(信号由 0 转变为 1)之后，只有当用户在显示设备上确认第一个消息后，才会继续生成后续消息。确认标识离开消息(信号由 1 转变为 0)后，下一个消息将显示在显示设备上。出现必须进行确认的进入消息(信号由 0 转变为 1)时，消息周期将重启。可以通过此方法使用显示设备来控制信号转变消息(离开消息除外)。

以全局方式为 STEP 7 组态中的 CPU 设置 SFB 33 到 35 的报告方法(启用或禁用确认触发报告)。缺省情况下将禁用确认触发报告。

为确保系统内消息评估的一致性，应确认是否所有显示设备都能够处理确认触发报告。

---

### 关于不能处理确认触发报告的显示设备的操作注意事项

已启用确认触发报告的 CPU 只会将消息分配给能够处理此报告方法的显示设备。如果没有显示设备能够处理确认触发报告，CPU 将不会发送任何消息。此种情况将通过 ERROR=1 和 STATUS=1 指示出来。

---

## 使用 SFB 33 "ALARM"、34 "ALARM\_8"和 35 "ALARM\_8P"确认消息

采用集中确认原则。在显示设备上确认消息后，确认信息将首先发送给生成该消息的 CPU。然后从 CPU 将确认信息分配到为确认而登录的所有站。

确认的将是信号而不是单个消息。例如，如果指示了信号的若干个上升沿，并且确认了进入状态的事件，将把具有相同消息编号的所有先前事件视为已确认的事件。



## 确认显示

SFB 36 "NOTIFY"和 SFB 31 "NOTIFY\_8P"没有确认指示。可以检查 SFB 33 "ALARM"的输出参数 ACK\_UP 和 ACK\_DN 及 SFB 35 "ALARM\_8P"和 34 "ALARM\_8"的输出参数 ACK\_STATE。如果控制参数 EN\_R 的值为 1，将在调用块时更新这些输出。

## 通过 SFC 或显示设备(WinCC)启用和禁用消息

在某些情况下，隐藏消息会有帮助(例如，重新组态系统时)。因此，可以在显示设备或程序中禁用和启用消息。禁用/启用应用于所有为特定消息而登录的站。禁用的消息将保持禁用状态，直到再次将其启用。

通过 ERROR 和 STATUS 输出参数(ERROR = 1，STATUS = 21)发送禁用消息通知。

## SFB 生成块相关消息所需的存储器资源

为了使 SFB 能够正常运行，必须在 CPU 通讯缓冲区中为其提供中介存储器空间，具体大小取决于关联值数据的长度。有关占用的存储器大小的信息，请参考下表。

块类型	所需的 CPU 工作存储器空间(字节)
NOTIFY	200 + 2 * 初次调用时在 SD_1、...SD_10 指定的关联值的长度
NOTIFY_8P	200 + 2 * 初次调用时在 SD_1、...SD_10 指定的关联值的长度
ALARM	200 + 2 * 初次调用时在 SD_1、...SD_10 指定的关联值的长度
ALARM_8	100
ALARM_8P	200 + 2 * 初次调用时在 SD_1、...SD_10 指定的关联值的长度
AR_SEND	54

## 可以传送的数据量

与 NOTIFY、NOTIFY\_8P、ALARM 和 ALARM\_8P SFB 的关联值 SD\_i 一并传送的数据不得超过最大长度。最大数据长度的计算方法如下：

$$\text{maxleng} = \min(\text{pdu\_local}, \text{pdu\_remote}) - \text{diff} - 4 * \text{使用的 SD}_i \text{ 参数的数量}$$

此处：

- $\min(\text{pdu\_lokal}, \text{pdu\_remote})$ 是数据字段 pdu\_lokal 和 pdu\_remote 的最小值
- pdu\_local 是本地 CPU 的数据字段的最大长度(请参见 CPU 的技术数据)
- pdu\_remote 是显示设备的数据字段的最大长度
- diff = 48(如果启用确认触发报告)；diff = 44(如果禁用确认触发报告)。

**实例：**

CPU 414-2 通过工业以太网向 WinCC 发送消息。确认触发报告处于禁用状态。

使用的关联值为 SD\_1、SD\_2 和 SD\_3。

pdu\_local = 480 字节，pdu\_remote = 480 字节

使用的 SD\_i 参数的数量：3

**因此：**

$\text{maxleng} = \min(480, 480) - 44 - 4 * 3 = 480 - 44 - 12 = 424$

每个 SFB 可以传送的数据的最大长度为 424 个字节。

## 25.2 使用 SFB 36 "NOTIFY"生成无需确认的块相关消息

### 描述

SFB 36 "NOTIFY"对信号进行监视。它在上升沿(进入状态的事件)和下降沿(离开状态的事件)均会生成消息。最多可以将十个关联值与消息一起发送。消息将发送到为此目的登录的所有站。初次调用 SFB 时，将发送具有当前信号状态的消息。

检测到沿时将查询关联值，并将其分配给消息。

每个 SFB 36 "NOTIFY"的实例可以使用一个包含两个存储器块的存储器。有关将信号转变保存到中介存储器的信息，请参考关于使用 SFB 生成块相关消息的介绍中的"信号转变检测"一节。

SFB 36 "NOTIFY"符合 IEC 1131-5 标准。

参数	声明	数据类型	存储器区	描述
SIG	INPUT	BOOL	I、Q、M、D、L	要监视的信号
ID	INPUT	WORD	I、Q、M、D、L、常数	消息的数据通道： 仅在初次调用时判断 W#16#EEEE ID。
EV_ID	INPUT	DWORD	I、Q、M、D、L、常数	仅在初次调用时判断消息号(不允许为 0) EV_ID。此后每次使用相应的实例 DB 调用 SFB 36 时，都将应用初次调用所使用的消息号。 STEP 7 会自动分配消息号以确保消息号的一致性。用户程序内的消息号必须唯一。
SEVERITY	INPUT	WORD	I、Q、M、D、L、常数	事件评价：可能的值：0 至 127 (0 表示最高权重)此参数与消息处理无关。
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数：消息生成已完成
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数 ERROR=TRUE 表示处理过程中出错。有关详细信息，请参考参数 STATUS。
STATUS	OUTPUT	WORD	I、Q、M、D、L	STATUS 参数：显示出错信息
SD_i , 1 i 10	IN_OUT	ANY	I、Q、M、D、T、C	i-th 关联值 只允许使用下列数据类型：BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME。 <b>注意：</b> 如果 ANY 指针访问某个 DB，则务必指定该 DB (例如：P# DB10.DBX5.0 Byte 10)。

## 错误信息

下表包含所有可通过 ERROR 和 STATUS 参数输出的 SFB 36 特有的出错信息。

ERROR	STATUS (十进制)	解释
0	11	消息丢失，上一次的信号改变或上一次的无法发送，将被当前消息取代。
0	22	<ul style="list-style-type: none"> <li>• 指向关联值 SD_i 的指针出错： <ul style="list-style-type: none"> <li>- 涉及数据长度或数据类型</li> <li>- 用户存储器中的关联值无法访问（例如，由于 DB 被删除或区域长度出错）</li> </ul> </li> <li>• 激活的消息没有或最终与有效数量的关联值一起发送</li> <li>• 所选的 SEVERITY 实际参数超出了允许的范围。激活的消息将在 SEVERITY=127 下发送。</li> </ul>
0	25	启动了通讯。正在处理消息。
1	1	通讯问题：连接中止或未登录
1	4	初次调用时： <ul style="list-style-type: none"> <li>• 指定的 EV_ID 超出了允许的范围或</li> <li>• ANY 指针 SD_i 有形式错误</li> <li>• 超出了每个 SFB 36 可以为 CPU 发送的最大存储器区</li> </ul>
1	10	无法访问本地用户存储器(例如，访问的 DB 已被删除)
1	12	调用 SFB 时： <ul style="list-style-type: none"> <li>• 指定了不属于 SFB 36 的实例 DB</li> <li>• 指定的是共享数据块而不是实例 DB</li> </ul>
1	18	EV_ID 已由 SFB 31 或 33 至 36 之一使用。
1	20	工作存储器不足。 H 系统：更新过程中调用了此 SFB
1	21	禁用了带有指定 EV_ID 的消息

## 25.3 使用 SFB 31 "NOTIFY\_8P"生成无确认显示的块相关消息

### 描述

SFB 31 "NOTIFY\_8P"表示将 SFB 36 "NOTIFY"扩展到八个信号。

在至少检测到一个信号转变时生成消息。初次调用 SFB 31 时将始终生成消息。为所有八个信号分配同一个公用消息号，它将在显示设备上拆分成八个子消息。

为所有八个信号分配一个公用消息号，该消息号在显示设备上拆分成八个子消息。有关将信号转变保存到中介存储器的信息，请参考关于使用 SFB 生成块相关消息的介绍中的"信号转变检测"一节。

### 注意

无论消息是否丢失，显示设备均显示最近的两次信号转变。

### 警告



在 CPU 中调用 SFB 31 "NOTIFY\_8P"前，必须确保所有连接的显示设备均可以识别此块。PLC 必须至少采用以下的一个软件版本，才能确保这一点：STEP 7 V5.1 Service Pack 3 以上版本、WinCC V5.1 Hot Fix 1 以上版本、PCS7 V5.2 Service Pack 2 以上版本、SIMATIC Device Driver V5.6 以上版本。否则，PLC 和所连接的显示设备之间的通讯将关闭。在这种情况下，将无法再使用系统连接的显示设备访问系统。

参数	声明	数据类型	存储区	描述
SIG_i, 1 i 8	INPUT	BOOL	I、Q、M、D、L。	要监视的 i-th 信号
ID	INPUT	WORD	常数(I、Q、M、D、L)	消息的数据通道：仅在初次调用时判断 W#16#EEEEID。
EV_ID	INPUT	DWORD	常数(I、Q、M、D、L)	消息号(不允许：0) 仅在初次调用时判断 EV_ID。此后每次使用相应的实例 DB 调用 SFB 31 时，都将应用初次调用所使用的消息号。 STEP 7 会自动分配消息号以确保消息号的一致性。用户程序内的消息号必须唯一。
SEVERITY	INPUT	WORD	常数(I、Q、M、D、L)	事件权重 有效值：0 至 127(值为 0 = 最高权重)； 默认值: 64 此参数与消息处理无关。
DONE	OUTPUT	BOOL	I、Q、M、D、L	状态参数 DONE：消息生成已完成。
ERROR	OUTPUT	BOOL	I、Q、M、D、L	状态参数 ERROR ERROR=TRUE 表示处理过程中出错。 有关详细信息，请参考参数 STATUS。
STATUS	OUTPUT	WORD	I、Q、M、D、L	状态参数 STATUS：出错信息显示

参数	声明	数据类型	存储区	描述
SD_i , 1 ≤ i ≤ 10	IN_OUT	ANY	I、Q、M、D、T、Z	<p>i-th 关联值</p> <p>只允许使用以下类型的数据：BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME。</p> <p><b>注意：</b> 如果 ANY 指针访问某个 DB，则务必指定该 DB。 (例如：P# DB10.DBX5.0 Byte 10)</p>

## 错误信息

下表包含所有可通过 ERROR 和 STATUS 参数输出的 SFB 31 特有的出错信息。

ERROR	STATUS (十进制)	解释
0	11	消息丢失: 上一次的信号改变或上一次的无法发送, 将被当前消息取代。
0	22	<ul style="list-style-type: none"> <li>指向关联值 SD_i 的指针出错： <ul style="list-style-type: none"> <li>与数据长度/类型有关</li> <li>无法访问用户存储器中的关联值(例如，由于 DB 已被删除或区域长度出错)</li> </ul> </li> <li>激活的消息未与或被要求与最大允许数量的关联值一起传送。</li> <li>所选择的 SEVERITY 实际参数超出了上限。激活的消息将在 SEVERITY=127 下发送。</li> </ul>
0	25	启动了通讯。正在处理消息。
1	1	通讯出错：通讯关闭或无登录
1	4	<p>初次调用时</p> <ul style="list-style-type: none"> <li>指定的 EV_ID 超出了允许的范围</li> <li>ANY 指针 SD_j 中有形式错误</li> <li>已超过 CPU 每个 SFB 31 可以传送的最大存储器区长度</li> </ul>
1	10	无法访问本地用户存储器(例如，尝试访问的 DB 已被删除)
1	12	<p>调用 SFB 时</p> <ul style="list-style-type: none"> <li>指定了不属于 SFB 31 的实例 DB</li> <li>指定的是全局数据块而不是实例 DB</li> </ul>
1	18	EV_ID 已由 SFB 31 或 33 至 36 之一使用。
1	20	<p>工作存储器空间不足。</p> <p>H 系统：更新过程中调用了此 SFB</p>
1	21	锁定了带有指定 EV_ID 的消息。

## 25.4 使用 SFB 33 "ALARM"生成需要确认的块相关消息

### 描述

SFB 33 "ALARM"对信号进行监视。

缺省模式(即,禁用确认触发报告):该块在上升沿(进入状态的事件)和下降沿(离开状态的事件)均会生成消息。最多可以将十个关联值与消息一起发送。

启用了确认触发报告:为信号生成进入消息后,在显示设备上确认该进入消息时,该块才会继续生成消息。

另请参见关于使用 SFB 生成块相关消息的介绍。

消息将发送到为此目的登录的所有站。

初次调用 SFB 时,将发送具有当前信号状态的消息。

将在上升沿重置 ACK\_UP 输出。收到登录的显示设备对进入状态的事件的确认时,将对其进行设置。

ACK\_DN 输出的情况与此类似:将在下降沿对其进行重置。从登录的显示设备收到对离开状态的事件的确认时,将对其进行设置。从登录的显示设备收到确认后,即将确认信息传递给为此目的登录的所有其它站。

每个 SFB33 "ALARM"的实例可以使用一个包含 2 个存储器块的消息存储器。有关将信号转变保存到中介存储器的信息,请参考关于使用 SFB 生成块相关消息的介绍中的"信号转变检测"一节。

SFB 33 "ALARM"符合 IEC 1131-5 标准。

参数	声明	数据类型	存储器区	描述
EN_R	INPUT	BOOL	I、Q、M、D、L、常数	为接收该项而启用的控制参数 决定初次调用块时更新(EN_R=1)还是不更新(EN_R=0)输出 ACK_UP 和 ACK_DN。如果 EN_R=0, 输出参数 ACK_UP 和 ACK_DN 将保持不变。
SIG	INPUT	BOOL	I、Q、M、D、L	要监视的信号
ID	INPUT	WORD	I、Q、M、D、L、常数	消息的数据通道: 仅在初次调用时判断 W#16#EEEE ID。
EV_ID	INPUT	DWORD	I、Q、M、D、L、常数	仅在初次调用时判断消息号(不允许为 0) EV_ID。此后每次使用相应的实例 DB 调用 SFB 33 时,都将应用初次调用所使用的消息号。 STEP 7 会自动分配消息号以确保消息号的一致性。用户程序内的消息号必须唯一。
SEVERITY	INPUT	WORD	I、Q、M、D、L、常数	事件权重 有效值:0 至 127 (值为 0 表示最高权重) 此参数与此消息的处理无关。
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数:消息生成已完成
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数 ERROR=TRUE 表示处理过程中出错。 有关详细信息,请参考参数 STATUS。
STATUS	OUTPUT	WORD	I、Q、M、D、L	STATUS 参数:显示出错信息

参数	声明	数据类型	存储器区	描述
ACK_DN	OUTPUT	BOOL	I、Q、M、D、L	在显示设备上确认了离开状态的事件初始化状态：1
ACK_UP	OUTPUT	BOOL	I、Q、M、D、L	在显示设备上确认了进入状态的事件初始化状态：1
SD_i , 1≤i≤10	IN_OUT	ANY	I、Q、M、D、 T、C	i-th 关联值 只允许使用下列数据类型：BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME。 <b>注意：</b> 如果 ANY 指针访问某个 DB，则务必指定该 DB (例如：P# DB10.DBX5.0 Byte 10)。



## 错误信息

下表包含所有可通过 ERROR 和 STATUS 参数输出的 SFB 33 特有的出错信息。

ERROR	STATUS (十进制)	解释
0	11	消息丢失: 上一次的信号改变或上一次的无法发送, 将被当前消息取代。
0	22	<ul style="list-style-type: none"> <li>• 指向关联值 SD_i 的指针出错 :               <ul style="list-style-type: none"> <li>- 涉及数据长度或数据类型</li> <li>- 无法访问用户存储器中的关联值 (例如, 由于 DB 已被删除或区域长度出错)</li> <li>- 激活的消息未与关联值一起发送</li> </ul> </li> <li>• 所选的 SEVERITY 实际参数超出了允许的范围。激活的消息将在 SEVERITY=127 下发送。</li> </ul>
0	25	启动了通讯。正在处理消息。
1	1	<p>通讯问题: 连接中止或未登录</p> <p>启用了确认触发报告时: 临时显示, 如果没有显示设备可以支持确认触发报告</p>
1	4	<p>初次调用时:</p> <ul style="list-style-type: none"> <li>• 指定的 EV_ID 超出了允许的范围或</li> <li>• ANY 指针 SD_i 有形式错误</li> <li>• 超出了每个 SFB 33 可以为 CPU 发送的最大存储器区</li> </ul>
1	10	无法访问本地用户存储器(例如, 访问的 DB 已被删除)
1	12	<p>调用 SFB 时:</p> <ul style="list-style-type: none"> <li>• 指定了不属于 SFB 33 的实例 DB</li> <li>• 指定的是共享数据块而不是实例 DB</li> </ul>
1	18	EV_ID 已由 SFB 31 或 33 至 36 之一使用。
1	20	<p>工作存储器不足。</p> <p>H 系统: 更新过程中调用了 FSB</p>
1	21	禁用了带有指定 EV_ID 的消息

### 注意

初次调用块后, ACK\_UP 和 ACK\_DN 输出的值将是 1, 并假定 SIG 输入的先前值为 0。

## 25.5 使用 SFB 35 "ALARM\_8P"生成针对八个信号的带有关联值的块相关消息

### 描述

SFB 35 "ALARM\_8P"表示将 SFB 33 "ALARM"线性扩展至八个信号。

只要未启用确认触发报告，在一个或多个信号处检测到信号转变时即会生成消息(例外：初次调用块时将始终发送消息)。所有八个信号拥有公用消息 ID，它在显示设备上被拆分成八个独立的消息。可以分别确认各个消息或确认一组消息。

可以使用 ACK\_STATE 输出参数在程序中处理单个消息的确认状态。如果禁用或启用 ALARM\_8P 块的消息，在任何情况下均会影响整个 ALARM\_8P 块。无法禁用或启用单个信号。

每个 SFB35 "ALARM\_8P"的实例可以使用一个带有两个存储器块的存储器。

有关将信号转变保存到中介存储器的信息，请参考关于使用 SFB 生成块相关消息的介绍中的"信号转变检测"一节。

参数	声明	数据类型	存储器区	描述
EN_R	INPUT	BOOL	I、Q、M、D、L、常数	为接收该项而启用的控制参数决定调用块时更新(EN_R=1)还是不更新(EN_R=0)输出 ACK_STATE。
SIG_i, 1 = i = 8	INPUT	BOOL	I、Q、M、D、L	要监视的 i(th)信号
ID	INPUT	WORD	I、Q、M、D、L、常数	消息的数据通道： 仅在初次调用时判断 W#16#EEEE ID。
EV_ID	INPUT	DWORD	I、Q、M、D、L、常数	仅在初次调用时判断消息号(不允许为 0) EV_ID。此后每次使用相应的实例 DB 调用 SFB 35 时，都将应用初次调用所使用的消息号。 STEP 7 会自动分配消息号以确保消息号的一致性。用户程序内的消息号必须唯一。
SEVERITY	INPUT	WORD	I、Q、M、D、L、常数	事件权重 有效值：0 至 127 (值为 0 表示最高权重) 此参数与此消息的处理无关。
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数：消息生成已完成
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数 ERROR=TRUE 表示处理过程中出错。 有关详细信息，请参考参数 STATUS。
STATUS	OUTPUT	WORD	I、Q、M、D、L	STATUS 参数：显示出错信息
ACK_STATE	OUTPUT	WORD	I、Q、M、D、L	包含所有八个消息的当前确认状态的位字段：(1: 事件已确认, 0: 事件未确认)： <ul style="list-style-type: none"> <li>0 到 7 位分别对应 SIG_1 至 SIG_7 的进入事件</li> <li>8 到 15 位分别对应 SIG_1 至 SIG_7 的离开事件</li> </ul> 初始化状态：W#16#FFFF，即已确认所有进入和离开事件

参数	声明	数据类型	存储器区	描述
SD_j, 1 ≤ j ≤ 10	IN_OUT	ANY	I、Q、M、D、 T、C	<p>j-th 关联值</p> <p>关联值将应用于所有消息。只允许使用下列数据类型：BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME。</p> <p><b>注意：</b> 如果 ANY 指针访问某个 DB，则务必指定该 DB (例如：P# DB10.DBX5.0 Byte 10)。</p>

## 错误信息

下表包含所有可通过 ERROR 和 STATUS 参数输出的 SFB 35 特有的出错信息。

ERROR	STATUS (十进制)	解释
0	11	消息丢失: 上一次的信号改变或上一次的无法发送, 将被当前消息取代。
0	22	<ul style="list-style-type: none"> <li>指向关联值 SD_j 的指针出错： <ul style="list-style-type: none"> <li>与数据长度或数据类型有关</li> <li>无法访问用户存储器中的关联值(例如，由于 DB 已被删除或区域长度出错)</li> </ul> </li> </ul> <p>激活的消息未与关联值一起发送</p> <ul style="list-style-type: none"> <li>所选的 SEVERITY 实际参数超出了允许的范围。激活的消息将在 SEVERITY=127 下发送。</li> </ul>
0	25	启动了通讯。正在处理消息。
1	1	<p>通讯问题：连接中止或未登录</p> <p>启用了确认触发报告时：临时显示，如果没有显示设备可以支持确认触发报告</p>
1	4	<p>初次调用时：</p> <ul style="list-style-type: none"> <li>指定的 EV_ID 超出了允许的范围或</li> <li>ANY 指针 SD_i 有形式错误</li> <li>超出了每个 SFB 35 可以为 CPU 发送的最大存储器区</li> </ul>
1	10	无法访问本地用户存储器(例如，访问的 DB 已被删除)
1	12	<p>调用 SFB 时：</p> <ul style="list-style-type: none"> <li>指定了不属于 SFB 35 的实例 DB</li> <li>指定的是共享数据块而不是实例 DB</li> </ul>
1	18	EV_ID 已由 SFB 31 或 33 至 36 之一使用。
1	20	<p>工作存储器不足。</p> <p>H 系统：更新过程中调用了此 SFB</p>
1	21	禁用了带有指定 EV_ID 的消息。

---

**注意**

初次调用块后，将设置 ACK\_STATE 输出的所有位，并假定输入 SIG\_i ( $1 \leq i \leq 8$ )的先前值均为 0。

---

## 25.6 使用 SFB 34 "ALARM\_8"生成针对八个信号的不附带关联值的块相关消息

### 描述

除了没有关联值 SD\_1 至 SD\_10 外，SFB 34 "ALARM\_8"与 SFB 35 "ALARM\_8P"完全相同。

参数	声明	数据类型	存储器区	描述
EN_R	INPUT	BOOL	I、Q、M、D、L、常数	为接收该项而启用的控制参数决定调用块时更新(EN_R=1)还是不更新(EN_R=0)输出 ACK_STATE。
SIG_i, 1 ≤ i ≤ 8	INPUT	BOOL	I、Q、M、D、L	要监视的 i(th)信号
ID	INPUT	WORD	I、Q、M、D、L、常数	消息的数据通道：仅在初次调用时判断 W#16#EEEE ID。
EV_ID	INPUT	DWORD	I、Q、M、D、L、常数	仅在初次调用时判断消息号(不允许为 0) EV_ID。此后每次使用相应的实例 DB 调用 SFB 34 时，都将应用初次调用所使用的消息号。 STEP 7 会自动分配消息号以确保消息号的一致性。用户程序内的消息号必须唯一。
SEVERITY	INPUT	WORD	I、Q、M、D、L、常数	事件权重 有效值：0 至 127 (值为 0 表示最高权重) 此参数与该消息的处理无关。
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数：消息生成已完成
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数 ERROR=TRUE 表示处理过程中出错。有关详细信息，请参考参数 STATUS。
STATUS	OUTPUT	WORD	I、Q、M、D、L	STATUS 参数：显示出错信息
ACK_STATE	OUTPUT	WORD	I、Q、M、D、L	包含所有八个消息的当前确认状态的位字段：(1: 事件已确认，0: 事件未确认)： <ul style="list-style-type: none"> <li>0 到 7 位分别对应 SIG_1 至 SIG_8 的进入事件</li> <li>8 到 15 位分别对应 SIG_1 至 SIG_8 的离开事件</li> </ul> 初始化状态：W#16#FFFF，即已确认所有进入和离开事件

**错误信息**

下表包含所有可通过 ERROR 和 STATUS 参数输出的 SFB 34 特有的出错信息。

ERROR	STATUS (十进制)	解释
0	11	消息丢失: 上一次的信号改变或上一次的无法发送, 将被当前消息取代。
0	22	所选的 SEVERITY 实际参数超出了允许的范围。激活的消息将在 SEVERITY=127 下发送。
0	25	启动了通讯。正在处理消息。
1	1	通讯问题: 连接中止或未登录 启用了确认触发报告时: 临时显示, 如果没有显示设备可以支持确认触发报告
1	4	初次调用时, 指定的 EV_ID 超出了允许的范围。
1	10	无法访问本地用户存储器(例如, 访问的 DB 已被删除)
1	12	<ul style="list-style-type: none"> <li>• 调用 SFB 时:</li> <li>• 指定了不属于 SFB 34 的实例 DB</li> <li>• 指定的是共享数据块而不是实例 DB</li> </ul>
1	18	EV_ID 已由 SFB 31 或 33 至 36 之一使用。
1	20	工作存储器不足。 H 系统: 更新过程中调用了此 SFB
1	21	禁用了带有指定 EV_ID 的消息

**注意**

初次调用块后, 将设置 ACK\_STATE 输出的所有位, 并假定输入 SIG\_i (1 ≤ i ≤ 8) 的先前值均为 0。

## 25.7 使用 SFB 37 "AR\_SEND"发送归档数据

### 描述

SFB 37 "AR\_SEND"将归档数据发送给为此目的登录的操作员界面系统。这些系统把登录消息中的相关归档号通知 CPU。视 CPU 可用的存储空间和已用地址区，归档数据最大长度可达 65534 个字节。在归档数据的结构中，必须考虑所使用的操作员界面系统的缺省值。

数据的发送在调用块后由控制输入 REQ 的上升沿激活。归档数据的起始地址由 SD\_1 指定，数据字段的长度由 LEN 指定。数据传送与用户程序的执行异步。如果传送成功完成，则通过将 DONE 状态参数的数值设置为 1 来进行指示。在控制输入 R 的上升沿处中止数据的传送。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	控制参数请求
R	INPUT	BOOL	I、Q、M、D、L、常数	控制参数重置：当前作业已中止
ID	INPUT	WORD	I、Q、M、D、L、常数	消息的数据通道：仅在初次调用时判断 W#16#EEEE ID。
AR_ID	INPUT	DWORD	I、Q、M、D、L、常数	归档号(不允许为 0) 仅在初次调用时判断 AR_ID。此后，每次使用相应实例 DB 调用 SFB 37 时，都将使用初次调用时的归档号。 STEP 7 会自动分配消息号以确保消息号的一致性。
DONE	OUTPUT	BOOL	I、Q、M、D、L	DONE 状态参数：发送已完成
ERROR	OUTPUT	BOOL	I、Q、M、D、L	ERROR 状态参数 ERROR=TRUE 表示处理过程中出错。有关详细信息，请参考参数 STATUS。
STATUS	OUTPUT	WORD	I、Q、M、D、L	STATUS 参数：显示出错信息
SD_1	IN_OUT	ANY	I、Q、M、D、T、C	指向归档数据的指针。不对长度规范进行判断。 只允许使用下列数据类型：BOOL (不允许：位域)、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME 归档数据必须具有 PLC 特有的结构。 <b>注意：</b> 如果使用 ANY 指针访问一个数据块，则必须要指定该数据块(例如：P# DB10.DBX5.0 Byte 10)。
LEN	IN_OUT	WORD	I、Q、M、D、L	要发送的数据字段的长度(字节)

## 错误信息

下表包含所有可通过 ERROR 和 STATUS 参数输出的 SFB 37 特有的出错信息。

ERROR	STATUS (十进制)	解释
0	11	警告：新作业未激活，因为前一作业仍在进行。
0	25	通讯已经启动。作业正在处理。
1	1	通讯问题
1	2	否定确认，无法执行功能
1	3	没有针对指定 AR_ID 的登录。
1	4	<ul style="list-style-type: none"> <li>归档数据指针 SD_1 中存在涉及数据长度或数据类型的错误。</li> <li>初次调用时，指定的 AR_ID 超出了允许的范围。</li> </ul>
1	5	所请求的重置已执行。
1	7	由于当前功能已完成或未激活(块处于不正确的状态)，因此与 RESET 作业无关。
1	10	无法访问本地用户存储器(例如，访问的 DB 已被删除)。
1	12	调用 SFB 时： <ul style="list-style-type: none"> <li>指定了不属于 SFB 37 的实例 DB</li> <li>指定的是共享数据块而不是实例 DB</li> </ul>
1	18	AR_ID 已由 SFB 37 使用。
1	20	工作存储器空间不足。 H 系统：更新过程中调用了此 SFB

## 数据一致性

为确保数据一致性，只有在完成当前发送过程后，才能写入当前使用的发送区域部分。当状态参数 DONE 的值为 1 时，便是这种情况。



## 25.8 使用 SFC 10 "DIS\_MSG"禁用块相关、符号相关和组状态消息

### 描述

可以使用 SFC 10 "DIS\_MSG"(禁用消息)禁任由 SFB 生成的块相关消息及禁用符号相关消息(SCAN)和组状态消息。使用输入参数 MODE 和 MESGN 来选择要禁用的消息。只有在未使用 SFC 10 禁用消息时，才能调用 SFC 10 "DIS\_MSG"并成功禁用消息。

对于准备在调用 SFC 10 时发送但仍然位于内部缓冲区中的消息，将无法再行禁用，而会发送出去。禁用的消息在"NOTIFY"、"NOTIFY\_8P"、"ALARM"、"ALARM\_8P"和"ALARM\_8" SFB 的 ERROR 和 STATUS 输出中给予指示。

通过在调用 SFC 10 时为 REQ 输入参数赋值 1 来启动对消息的禁用。

### SFC 10 的工作方式

禁用以异步方式执行，换言之，它可以在若干个 SFC 10 调用过程中一直保持激活状态。另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义：

- 初次调用时(REQ = 1)，SFC 10 检查输入参数，并尝试占据所需的系统资源。如果尝试成功，将在 RET\_VAL 中输入值 W#16#7001、设置 BUSY 并启动对消息的禁用。如果不成功，将在 RET\_VAL 中输入出错信息并终止作业。在这种情况下不会评价 BUSY。
- 如果期间进行了更多调用，将在 RET\_VAL 中输入值 W#16#7002(仍然由 CPU 执行作业)并设置 BUSY。进行更多调用不会影响当前作业。
- 最后一次调用 SFB 时，如果没有出错，将在 RET\_VAL 中输入值 W#16#0000。BUSY 值将为 0。如果出错，错误信息将被输入到 RET\_VAL 中，不会评价 BUSY。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	REQ=1：触发禁用
MODE	INPUT	BYTE	I、Q、M、D、L、常数	有关用于选择要禁用消息的参数，请参见下表
MESGN	INPUT	DWORD	I、Q、M、D、L、常数	只有 MODE 设置为 5、6、7 时消息号才有效。这使得单个消息禁用。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	有关出错信息，请参见"出错信息"表
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：尚未取消禁用。

**MODE 输入参数**

下表列出了允许 MODE 输入参数使用的值：

值	含义
0	使用 SFB 生成 CPU 的所有块相关、符号相关和组状态消息
1	使用 SFB 生成 CPU 的所有块相关消息，即所有由"NOTIFY"、"NOTIFY_8P"、"ALARM"、"ALARM_8P"和"ALARM_8" SFB 生成的消息
2	CPU 的所有组状态消息
3	CPU 的所有符号相关消息(SCAN)
5	"符号相关消息"类的单个消息
6	"使用 SFB 生成的块相关消息"类的单个消息
7	"组状态消息"类的单个消息

**错误信息**

错误代码 (W#16#...)	解释
0000	禁用终止而没有出错。
7000	初次调用时 REQ = 0：未激活禁用。
7001	初次调用时 REQ = 1：触发了禁用。
7002	再次调用：禁用已激活。
8081	访问参数时出错
8082	MODE 含有违规值。
8083	消息号超出了允许的值范围。
8084	没有登录使用 MODE(及可能的情况下使用 MESGN)指定的消息。
80C3	无法禁用要在 MODE(及可能的情况下在 MESGN)中禁用的消息，因为目前 SFC 10 已在禁用消息。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 25.9 使用 SFC 9 "EN\_MSG"启用块相关、符号相关和组状态消息

### 描述

使用 SFC 9 "EN\_MSG" (启用消息)可以启用先前禁用的块相关、符号相关和组状态消息。或者在显示设备上禁用消息，或者使用 SFC 10 "DIS\_MSG"禁用。

使用 MODE 和 MESGN 输入参数指定要启用的消息。只有在 SFC 9 尚未有效地启用消息时，才能使用 SFC 9 "EN\_MSG"成功地启用消息。

通过为 SFC 9 的 REQ 输入参数赋值 1 来启动启用功能。

### SFC 9 的工作方式

启用以异步方式执行，换言之，它可以在若干个 SFC 9 调用过程中一直保持激活状态。另请参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义：

- 初次调用时(REQ = 1)，SFC 9 会检查输入参数，并尝试占用所需的系统资源。如果尝试成功，将在 RET\_VAL 中输入值 W#16#7001、设置 BUSY 并启动对消息的启用。  
如果不成功，将在 RET\_VAL 中输入出错信息并终止作业。在这种情况下不会评价 BUSY。
- 如果期间进行了更多调用，将在 RET\_VAL 中输入值 W#16#7002(仍然由 CPU 执行作业)并设置 BUSY。进行更多调用不会影响当前作业。
- 最后一次调用 SFB 时，如果没有出错，将在 RET\_VAL 中输入值 W#16#0000。BUSY 值将为 0。  
如果出错，错误信息将被输入到 RET\_VAL 中，不会评价 BUSY。

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L	REQ=1：触发启用
MODE	INPUT	BYTE	I、Q、M、D、L、常数	用于选择要启用消息的参数
MESGN	INPUT	DWORD	I、Q、M、D、L、常数	只有 MODE 设置为 5、6、7 时消息号才有效。这使得单个消息禁用。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	出错信息，请参见表格。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1：尚未取消启用。

**MODE 输入参数**

下表列出了允许 MODE 输入参数使用的值。

值	含义
0	使用 SFB 生成 CPU 的所有块相关、符号相关和组状态消息
1	使用 SFB 生成 CPU 的所有块相关消息，即所有由"NOTIFY"、"NOTIFY_8P"、"ALARM"、"ALARM_8P"和"ALARM_8" SFB 生成的消息
2	CPU 的所有组状态消息
3	CPU 的所有符号相关消息(SCAN)
5	"符号相关消息"类的单个消息
6	"使用 SFB 生成的块相关消息"类的单个消息
7	"组状态消息"类的单个消息

**错误信息**

错误代码 (W#16#...)	解释
0000	禁用终止而没有出错。
7000	初次调用时 REQ = 0：未激活启用。
7001	初次调用时 REQ = 1：已触发启用。
7002	再次调用：启用已激活。
8081	访问参数时出错
8082	MODE 含有违规值。
8083	消息号超出了允许的值范围。
8084	没有登录使用 MODE(及可能的情况下使用 MESGN)指定的消息。
80C3	目前无法启用要在 MODE(及可能的情况下在 MESGN)中启用的消息，因为 SFC 9 已在启用消息。
8xyy	常规错误信息，通过输出参数 RET_VAL 判断错误

## 25.10 用于生成块相关消息的 SFB 的启动特性

### 重启(暖启动)

在暖启动过程中，将把用于生成块相关消息的 SFB 的实例 DB 设置为 NO\_INIT(未初始化)状态。存储在实例 DB 中的实际参数未发生变化。将在下次调用块时判断参数 ID 和 EV\_ID。

### 完全重启

在完全重启过程中，将把用于生成块相关消息的 SFB 的实例 DB 内容重置为其初始值。

### 热启动

在热启动过程中，用于生成块相关消息的 SFB 的特性，类似于具有继续执行功能的用户功能块。它们从中断点继续执行。

### 存储器复位

只要复位存储器，就将使所有连接终止，从而也没有为消息而登录的站。将删除用户程序。如果插入了闪存卡，与执行相关的程序段将再次从该卡加载到 CPU 上，随后 CPU 执行暖启动或冷启动(在任何情况下均隐式执行冷启动，因为清空存储器后将初始化所有用户数据)。

## 25.11 用于生成块相关消息的 SFB 如何对问题做出反应

### 连接故障

对分配给 SFB 实例的连接进行故障监视。如果某个连接发生故障，将从为块相关消息而登录的站内部 CPU 列表中删除所涉及的站。删除这些站的所有未决消息。

如果在连接发生故障后其它站仍然处于登录状态，它们将继续接收消息。只有在没有任何登录到站的连接时，SFB 才会停止发送消息。ERROR 和 STATUS 输出参数会指示这种情况(ERROR = 1，STATUS = 1)。

### 到用户程序的出错接口

如果在执行用于生成块相关消息的 SFB 的过程中发生错误，将把 ERROR 输出参数设置为 1，STATUS 输出参数将具有相应的错误标识符。可以在用户程序中判断此出错信息。

可能发生的错误实例：

- 由于资源不足而无法发送
- 访问要监视的信号中的一个时出错。

## 25.12 关于使用 SFC 生成块相关消息的介绍

### 用于生成块相关消息的 SFC

可以使用下列 SFC 生成块相关消息：

- SFC 17 "ALARM\_SQ"
- SFC 18 "ALARM\_S"
- SFC 107 "ALARM\_DQ"
- SFC 108 "ALARM\_D"

以上 SFC 具有下列属性：

- 可以在登录的显示设备上确认信号状态为 1 时由 SFC 17 "ALARM\_SQ"和 SFC 107 "ALARM\_DQ"发送的消息。始终隐式确认 SFC 18 "ALARM\_S"和 SFC 108 "ALARM\_D"的消息。生成消息的不是检测到的边沿变化,而是每个 SFC 调用。有关更多详细信息,请参考使用 SFC 17 "ALARM\_SQ"生成可确认的块相关消息及使用 SFC 18 "ALARM\_S"生成永久确认的块相关消息
- "ALARM\_S"和使用 SFC 17 "ALARM\_DQ"和 108 "ALARM\_D"生成可确认的块相关消息一节。
- 执行块后,将完整读取关联值 SD\_1,并将其分配给消息。在一致性方面,与高优先级类相比,以下关联值是一致的:
  - 简单数据类型(位、字节、字及双字)
  - 数据类型字节的数组,其最大长度取决于具体 CPU 特定的最大长度(请参见/71/、/101/)。

### SFC 19 "ALARM\_SC"

使用 SFC 19 "ALARM\_SC"可以查询下列内容：

- 上一"进入状态消息"的确认状态和上一 SFC 17/SFC 107 调用的信号状态,或
- 上一 SFC 18/SFC 108 调用的信号状态。

### 登录显示设备

只有在至少为块相关消息登录了一台显示设备时,用于生成块相关消息的 SFC 才会在其被调用时发送消息。

### 消息存储

为避免在通讯系统的通讯量大时丢失消息,SFC 17、18、107 和 108 都可以缓存两个消息。不过,如果丢失了消息,会在 RET\_VAL 中通知用户。下次可以发送消息时,会将此情况通知登录的显示设备。

### 使用 SFC 17 "ALARM\_SQ"和 107 "ALARM\_DQ"确认消息

如果在显示设备上确认了"进入事件消息",将先把此确认信息发送给生成消息的 CPU。随后由 CPU 将该确认信息分配给为此目的登录的所有站。

### 禁用和启用消息

对于使用 SFC 17 "ALARM\_SQ"、SFC 18 "ALARM\_S"、SFC 107 "ALARM\_DQ"或 SFC 108 "ALARM\_D"生成的块相关消息，无法在禁用后再启用。

### 包含 SFC 17/SFC 18 调用的程序的更改

---

#### 注意

使用 SFC 17/SFC 18 调用下载一个已在 CPU 中的块时，可能会出现的情况是：前一个块已发送了进入状态消息，但新块未发送相应的离开状态消息。这意味着消息将保留在 CPU 的内部消息存储器中。这种情况还会发生在使用 SFC 17/SFC 18 删除块时。

将 CPU 转入 STOP 状态，然后执行暖启动或冷启动，即可从 CPU 的内部消息存储器中删除此类消息。

---

### 包含 SFC 17/SFC 18 调用的程序的更改

即使程序可能包含 SFC 107 和/或 SFC 108 调用，所述程序修改仍可能使消息驻留在内部消息存储器中，并因此永久性地占用系统资源。

与由 SFC 17/SFC 18 调用占用的系统资源相反，可以在不将 CPU 切换到 STOP 模式的情况下，释放由 SFC 107/SFC 108 调用占用的系统资源。这是通过使用 SFC 106 "DEL\_SI"来执行的，请参见使用 SFC 106 "DEL\_SI"读取动态系统资源。在通过调用 SFC 106 "DEL\_SI"释放动态占用的系统资源前，可能需要借助 SFC 105 "READ\_SI"获取有关 CPU 当前动态占用的系统资源的信息。请参见使用 SFC 105 "READ\_SI"读取动态系统资源。



## 可以传送的数据的量

使用 SFC ALARM\_S、ALARM\_SQ、ALARM\_D 和 ALARM\_DQ 的伴随值 SD 可以传送的数据量不能超过最大长度。数据长度的计算方法如下：

$$\text{maxleng} = \min(\text{pdu\_local}, \text{pdu\_remote}) - 48$$

定义：

- pdu\_local : CPU 数据块(SZL\_ID W#16#0131、INDEX 1、变量 pdu)的最大长度
- pdu\_remote : 显示设备数据块的最大长度

实例：

一个 CPU 414-1 将向一个编程设备 PG 760 (通过 MPI)发送消息。

pdu\_local = 480 字节，pdu\_remote = 480 字节，

结果：

$$\text{maxleng} = \min(480, 480) - 48 = 480 - 48 = 432$$

因此，每个 SFC 可以传送的最大数据长度为 432 个字节。

## 25.13 使用 SFC 17 "ALARM\_SQ"生成可确认的块相关消息及使用 SFC 18 "ALARM\_S"生成永久确认的块相关消息

---

### 注意

新创建的程序只能使用 SFC 107 和 108 (如果您的 CPU 支持 SFC 107 和 108) ,因为它们提供了用于管理系统资源的增强选项。

---

### 描述

每次调用 SFC 17 "ALARM\_SQ"和 SFC 18 "ALARM\_S"时,它们都会生成一个可以向其添加关联值的消息。将该消息发送给为此而登录的所有站。SFC 17 和 SFC 18 提供了简单的消息发送机制。必须确保当触发信号 SIG 的值是上一调用的反值时,只调用 SFC 17 或 SFC 18。如果不是这种情况,将在 RET\_VAL 中给予指示,且不会发送消息。初次调用 SFC 17 或 SFC 18 时,必须确保 SIG 输入的值为 1。否则 RET\_VAL 包含错误信息,将不会发送消息。

---

### 注意

请务必在先前已分配了适当系统属性的 FB 中调用 SFC 17 和 SFC 18 !有关向块分配系统属性的更多详细信息,请参考/234/一节。

---

### 系统资源的使用

使用 SFC 17 "ALARM\_SQ"和 18 "ALARM\_S"生成消息时,操作系统在信号周期过程中使用一个系统资源。

对于 SFC 18 "ALARM\_S",信号周期从 SFC 调用 SIG=1 开始持续到 SIG=0 的另一个调用。对于 SFC 17 "ALARM\_SQ",此时期还包括直到其中一个已报告的显示设备确认进入信号的时间(必要时)。

如果在信号周期过程中,生成消息的块超载或被删除,相关系统资源将保持占用状态,直到下次重启(暖启动)。

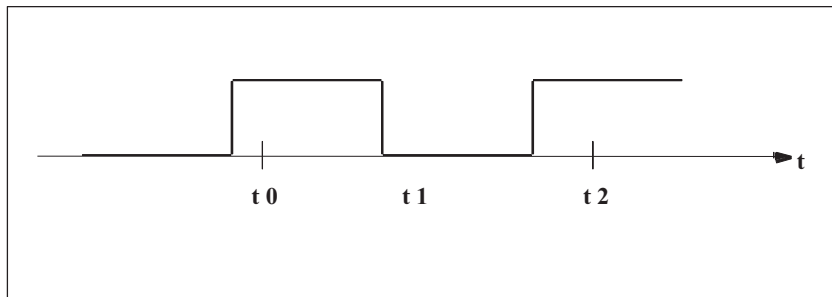
### 确认消息

当所监视的信号的状态为 1 时,可确认 SFC 17 "ALARM\_SQ"发送的消息。可查询上一"进入的事件消息"的确认状态和使用 SFC 19 "ALARM\_SC"进行的上一 SFC 调用的信号状态。始终隐式确认使用 SFC 18 "ALARM\_S"发送的消息。使用 SFC 19 "ALARM\_SC"可以查询上次 SFC 18 调用的信号状态。

## 信号状态的临时存储

SFC 17 "ALARM\_SQ"和 SFC 18 "ALARM\_S"会临时占用系统资源。它们在此处输入上两个信号状态(包括时间标志和关联值)等信息。如果在未发送上两个"有效"SFC 调用的信号状态(信号溢出)时调用 SFC 17 或 SFC 18，将放弃当前和上一信号状态，并在缓冲区中设置溢出 ID。在下一个可能的时机，将发送倒数第二个信号和溢出标识符。

实例：



t0、t1 和 t2 是调用 SFC 17 或 SFC 18 所在的点。如果 t0 和 t1 的信号状态未在时间 t2 发送，将放弃 t1 和 t2 的信号状态，并会为 t0 的信号状态设置溢出标识符。

## 实例溢出

如果 SFC 17 或 SFC 18 调用的数量大于 CPU 系统资源的最大数量，可能会导致资源不足(实例溢出)。将通过 RET\_VAL 中的信息和登录的显示设备指明这种情况。

SFC 17 或 SFC 18 调用的最大数量取决于具体的 CPU。您将在/70/和/101/中找到此信息。

参数	声明	数据类型	存储器区	描述
SIG	INPUT	BOOL	I、Q、M、D、L	触发消息的信号
ID	INPUT	WORD	I、Q、M、D、L、常数	消息的数据通道：W#16#EEEE
EV_ID	INPUT	DWORD	I、Q、M、D、L、常数	消息号(不允许为 0)
SD	INPUT	ANY	I、Q、M、D、T、C	相关值 最大长度：12 字节 允许使用以下数据类型 BOOL (不允许：位域)、BYTE、CHAR、WORD、INT、DWORD、DINT、REAL、DATE、TOD、TIME、S5TIME、DATE_AND_TIME、COUNTER、TIMER。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息

## 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
0001	<ul style="list-style-type: none"> <li>关联值超过了允许的最大长度或</li> <li>无法访问用户存储器(例如, 访问的 DB 已被删除)。发送消息。</li> <li>关联值指向本地数据区中的某个值。发送消息。(仅限 S7-400)</li> </ul>
0002	警告: 已经使用了最后一部分可用的消息确认存储器。 (仅限 S7-400)
8081	指定的 EV_ID 超出了允许的范围。
8082	由于 CPU 不再有可以供 SFC 生成块相关消息的资源, 消息丢失。
8083	由于已存在相同的信号更改但尚无法发送(信号溢出), 消息丢失。
8084	当前 SFC 17 或 SFC 18 调用中, 触发消息的信号(SIG)的值与上次调用时的值相同。
8085	没有针对指定 EV_ID 的登录
8086	已在较低优先级类执行了针对指定 EV_ID 的 SFC 调用。
8087	初次调用 SFC 17 或 SFC 18 时, 消息触发信号的值为 0。
8088	指定的 EV_ID 已由另一个系统资源(分配给 SFC 17、18、107、108)使用。
8xyy	常见出错信息, 请参见通过输出参数 RET_VAL 判断错误

## 25.14 使用 SFC 19 "ALARM\_SC"查询上一 ALARM\_SQ/ALARM\_DQ 进入事件消息的确认状态

### 描述

使用 SFC 19 "ALARM\_SC"可以查询下列内容：

- 上一 ALARM\_SQ/ALARM\_DQ 进入状态消息的确认状态和上次调用 SFC 17 "ALARM\_SQ"/SFC 107 "ALARM\_DQ"时触发消息的信号状态，或
- 上次调用 SFC 18 "ALARM\_S"/SFC 108 "ALARM\_D"时触发消息的信号状态。

假定在消息组态期间分配了消息号，将通过 SFC 19 "ALARM\_SC"访问 SFC 17 或 SFC 18/SFC 107/SFC 108 临时占用存储器的唯一消息号引用消息或信号。

参数	声明	数据类型	存储器区	描述
EV_ID	INPUT	DWORD	I、Q、M、D、L、常数	上次 SFC 调用信号状态的消息号或想要查询的上一进入状态消息(仅限使用 SFC 17 和 SFC 107!)的确认状态。
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息
STATE	OUTPUT	BOOL	I、Q、M、D、L	上次 SFC 调用触发消息的信号状态
Q_STATE	OUTPUT	BOOL	I、Q、M、D、L	如果指定的 EV_ID 参数属于 SFC 18/SFC 108 调用：1
				如果指定的 EV_ID 参数属于 SFC 17/SFC 107 调用：上一进入状态消息的确认状态： 0: 未确认 1: 已确认

### 错误信息

错误代码 (W#16#...)	解释
0000	未产生故障。
8081	指定的 EV_ID 超出了允许的范围。
8082	当前没有针对此 EV_ID 而占据存储器空间(可能的原因：相应的信号状态尚不是 1 或信号状态已恢复为 0)。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 25.15 使用 SFC 107 "ALARM\_DQ"和 108 "ALARM\_D"生成可确认和永久确认的块相关消息

### 描述

每次调用，SFC 107 "ALARM\_DQ"和 108 "ALARM\_D"都会生成一个可向其附加关联值的消息。从而与 SFC 17 "ALARM\_SQ"和 18 "ALARM\_S"保持一致。

使用 SFC 107 "ALARM\_DQ"和 108 "ALARM\_D"生成消息时，操作系统会在信号周期过程中临时占用系统资源。

SFC 108 "ALARM\_D"的信号周期始于 SIG=1 的 SFC 调用，终于 SIG=0 的新调用。对于 SFC 107 "ALARM\_DQ"，它的周期可能因时间超时而延长，一直到在登录的显示设备上确认进入信号为止。

对于 SFC 108 "ALARM\_D"，信号周期从 SFC 调用 SIG=1 开始持续到 SIG=0 的另一个调用。对于 SFC 107 "ALARM\_DQ"，此时期还包括直到其中一个已报告的显示设备确认进入信号的时间(如有必要)。

如果在信号周期过程中，生成消息的块超载或被删除，相关系统资源将保持占用状态，直到下次重启(暖启动)。

与 SFC 17 和 18 相比，SFC 107 "ALARM\_DQ"和 108 "ALARM\_D"所增加的功能是：可以对这些占用的系统资源进行管理：

- 借助 SFC 105 "READ\_SI"，可以获取与占用的系统资源有关的信息。
- 可以使用 SFC 106 "DEL\_SI"再次释放占用的系统资源。这对于永久占用的系统资源具有特殊意义。例如，如果在程序更改过程中删除了包含 SFC107 或 SFC108 调用的 FB，当前占用的系统资源将保持占用状态，直到下次重启(暖启动)。更改程序并重新加载包含 SFC 107 或 SFC 108 调用的 FB 时，可能会发生 SFC 107 和 108 不再生成消息的情况。

SFC 107 和 108 包含的参数比 SFC 17 和 18 多一个，即输入 CMP\_ID。使用此输入可以将由 SFC 107 和 108 生成的消息分配到逻辑区，例如，系统的各个部分。。如果调用 FB 中的 SFC 107/SFC 108，显然需要将相应的实例 DB 号分配到 CMP\_ID。

参数	声明	数据类型	存储器区	描述
SIG	INPUT	BOOL	I、Q、M、D、L	消息触发信号
ID	INPUT	WORD	I、Q、M、D、L、常数	消息的数据通道：W#16#EEEE
EV_ID	INPUT	DWORD	I、Q、M、D、L、常数	消息号(不允许：0)
CMP_ID	INPUT	DWORD	I、Q、M、D、L、常数	组件标识符(不允许：0) 将相应消息分配到的部分系统的 ID 建议值： <ul style="list-style-type: none"> <li>• 低位字：1 至 65535</li> <li>• 高位字：0</li> </ul> 如果使用上述建议值，将不会遇到与 SIEMENS 程序数据包冲突的情况。

参数	声明	数据类型	存储器区	描述
SD	INPUT	ANY	I、Q、M、D、 T、C	相关值 最大长度：12 字节 只允许使用 BOOL 型数据(不允许：位域)、 BYTE、CHAR、WORD、INT、DWORD、 DINT、REAL、DATE、TOD、TIME、 S5TIME、DATE_AND_TIME
RET_VAL	OUTPUT	INT	E、A、M、D、L	错误信息

## 错误信息

错误代码 (W#16#...):	解释
0000	未产生故障。
0001	<ul style="list-style-type: none"> <li>关联值的长度超出了允许的最大长度，或</li> <li>无法访问用户存储器(例如，访问的 DB 已被删除)。发送激活的消息。</li> <li>关联值指向本地数据区中的某个值。发送消息。(仅限 S7-400)</li> </ul>
0002	警告：最后一部分可用的消息确认存储器已被占用。 (仅限 S7-400)
8081	指定的 EV_ID 超出了有效范围。
8082	由于 CPU 不再有可以供 SFC 生成块相关消息的资源，消息丢失。
8083	由于已存在相同的信号转变但尚无法发送(信号溢出)，消息丢失。
8084	对于当前和以前的 SFC 107-/SFC-108 调用，消息触发信号 SIG 具有相同的值。
8085	没有针对指定 EV_ID 的登录。
8086	已在较低优先级类处理了针对指定 EV_ID 的 SFC 调用。
8087	初次调用 SFC 107/SFC 108 时，消息触发信号的值为 0。
8088	指定的 EV_ID 已由另一个系统资源(分配给 SFC 17、18、107、108)使用。
8089	已将 CMP_ID 赋值为 0。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 25.16 使用 SFC 105 "READ\_SI"读取动态系统资源

### 使用 SFC 107 和 108 生成消息时，动态占用的系统资源有何变化

使用 SFC 107 "ALARM\_DQ"和 108 "ALARM\_D"生成消息时，操作系统会临时占用系统存储器空间。

例如，如果不删除 CPU 中存在的包含 SFC 107 或 SFC 108 调用的 FB，可能会发生相应系统资源保持永久占用状态的情况。如果重新加载包含 SFC 107 或 SFC 108 调用的 FB，则可能发生 SFC 107 和 108 无法再得到正确处理的情况。

### 描述

使用 SFC 105 "READ\_SI"可以读取生成消息时由 SFC 107 和 108 占用的系统资源。这是通过在此处使用 EV\_ID 和 CMP\_ID 值来实现的。这些值通过 SI\_ID 参数传给 SFC 105 "READ\_SI"。

SFC 105 "READ\_SI"具有下表说明的四种可能的工作模式。通过 MODE 参数设置所需的工作模式。

MODE	读取 SFC 107/SFC 108 占用的哪些系统资源？
1	全部(SI_ID:=0 的 SFC 105 调用)
2	EV_ID:=ev_id 的 SFC 107-/SFC 108 调用(SI_ID:=ev_id 的 SFC 105 调用)占用的系统资源
3	CMP_ID:=cmp_id 的 SFC 107-/SFC 108 调用(SI_ID:=ev_id 的 SFC 105 调用)占用的系统资源
0	由于指定的目标域 SYS_INST 过小，不能通过上一调用在 MODE=1 或 MODE=3 下读取的其它系统资源



## 工作原理

如果在 MODE=1 或 MODE=3 下调用 SFC 105 时未选择足够大的 SYS\_INST 目标区，该区将包含通过 MODE 参数选择的所有当前占用的系统资源的内容。

系统资源负载大将导致相应的 SFC 运行时间较长。也就是说，高 CPU 性能负载可能导致超过最大可组态周期监视时间。

可以使用下列方法解决此运行时间问题：选择一个相对较小的 SYS\_INST 目标区。如果 SFC 无法在 SYS\_INST 中输入要读取的所有系统资源，RET\_VAL=W#16#0001 会发送通知。在这种情况下，请在 MODE=0 下并使用与前一调用相同的 SI\_ID 调用 SFC 105，直到 RET\_VAL 的值为 W#16#0000。

## 注意

由于操作系统不会组合属于读取作业的 SFC 105 调用，因此应以相同优先级执行所有 SFC 105 调用。

## 目标区 SYS\_INST 的结构

获取的已占用系统资源的目标区必须位于 DB 内。应将目标区适当定义为结构域，结构的构成如下：

结构单元	数据类型	描述
SFC_NO	WORD	占用系统资源的 SFC 的编号
LEN	BYTE	结构的长度(字节)，包括 SFC_NO 和 LEN : B#16#0C
SIG_STAT	BOOL	信号状态
ACK_STAT	BOOL	进入事件的确认状态(上升沿)
EV_ID	DWORD	消息号
CMP_ID	DWORD	部分系统 ID

参数	声明	数据类型	存储器区	描述
MODE	INPUT	INT	I、Q、M、D、L、 常数	作业标识符 允许值 <ul style="list-style-type: none"> <li>1: 读取所有系统资源</li> <li>2: 读取调用 SFC 107-/SFC 108 时使用 EV_ID = ev_id 占用的系统资源</li> <li>3: 读取调用 SFC 107-/SFC 108 时使用 CMP_ID = cmp_id 占用的系统资源</li> <li>0: 后续调用</li> </ul>
SI_ID	INPUT	DWORD	I、Q、M、D、L、 常数	要读取的系统资源的 ID 允许值 <ul style="list-style-type: none"> <li>0，如果 MODE=1</li> <li>消息号 ev_id，如果 MODE=2</li> <li>用于标识系统部分的 ID cmp_id，如果 MODE=3</li> </ul>
RET_VAL	OUTPUT	INT	I、Q、M、D、L	返回值(出错信息或作业状态)
N_SI	OUTPUT	INT	I、Q、M、D、L	包含 SYS_INT 的输出系统资源数
SYS_INT	OUTPUT	ANY	D	已获取系统资源的目标区。

## 错误信息

错误代码 (W#16#...):	解释
0000	未产生故障。
0001	由于所选的 SYS_INT 目标范围过窄，因此无法读取所有系统资源。
8081	(仅限 MODE=2 或 3)已将 SI_ID 赋值为 0。
8082	(仅限 MODE=1)已将一个 0 不同值赋值到 SI_ID。
8083	(仅限 MODE=0)已将 SI_ID 赋值为与前一 SFC 调用(MODE=1 或 3)SI_ID 不同的值。
8084	为 MODE 分配了非法值。
8085	SFC 105 正在另一个 OB 中接受处理。
8086	系统资源的目标区 SYS_INST 过小。
8087 或 8092	DB 中不存在目标区域 SYS_INST 或者 ANY 指针中出现错误。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 25.17 使用 SFC 106 "DEL\_SI"读取动态系统资源

### 使用 SFC 107 和 108 生成消息时，动态系统资源有何变化

使用 SFC 107 "ALARM\_DQ"和 108 "ALARM\_D"生成消息时，操作系统会临时占用系统存储器空间。

例如，如果不删除 CPU 中存在的包含 SFC 107 或 SFC 108 调用的 FB，可能会发生相应系统资源保持永久占用状态的情况。如果重新加载包含 SFC 107 或 SFC 108 调用的 FB，则可能发生 SFC 107 和 108 无法再得到正确处理的情况。

### 描述

使用 SFC 106 "DEL\_SI"可以删除当前使用的系统资源。

SFC 106 "DEL\_SI"具有下表说明的三种可能的工作模式。通过 MODE 参数设置所需的工作模式。

MODE	删除 SFC 107/SFC 108 占用的哪些系统资源？
1	全部(SI_ID:=0 的 SFC 106 调用)
2	EV_ID:=ev_id 的 SFC 107-/SFC 108 调用(SI_ID:=ev_id 的 SFC 106 调用)占用的系统资源
3	CMP_ID:=cmp_id 的 SFC 107-/SFC 108 调用(SI_ID:=ev_id 的 SFC 106 调用)占用的系统资源

参数	声明	数据类型	存储器区	描述
MODE	INPUT	INT	I、Q、M、D、L、 常数	作业标识符 允许值 <ul style="list-style-type: none"> <li>1: 删除所有系统资源</li> <li>2: 删除调用 SFC 107-/SFC 108 时使用 EV_ID = ev_id 占用的系统资源</li> <li>3: 删除调用 SFC 107-/SFC 108 时使用 CMP_ID = cmp_id 占用的系统资源</li> </ul>
SI_ID	INPUT	DWORD	I、Q、M、D、L、 常数	要删除的系统资源的 ID 允许值 <ul style="list-style-type: none"> <li>0, 如果 MODE=1</li> <li>消息号 ev_id, 如果 MODE=2</li> <li>用于标识系统部分的 ID cmp_id, 如果 MODE=3</li> </ul>
RET_VAL	OUTPUT	INT	I、Q、M、D、L	错误信息

## 错误信息

错误代码 (W#16#...):	解释
0000	未产生故障。
8081	(仅限 MODE=2 或 3)已将 SI_ID 赋值为 0。
8082	(仅限 MODE=1)已将一个 0 不同值赋值到 SI_ID。
8084	为 MODE 分配了非法值。
8085	当前正在处理 SFC 106。
8086	由于在调用 SFC 106 时至少有一个所选择的系统资源正在接受处理，因此无法删除全部所选择的系统资源。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

## 26 IEC 定时器和 IEC 计数器

### 26.1 使用 SFB 3 "TP"生成脉冲

#### 描述

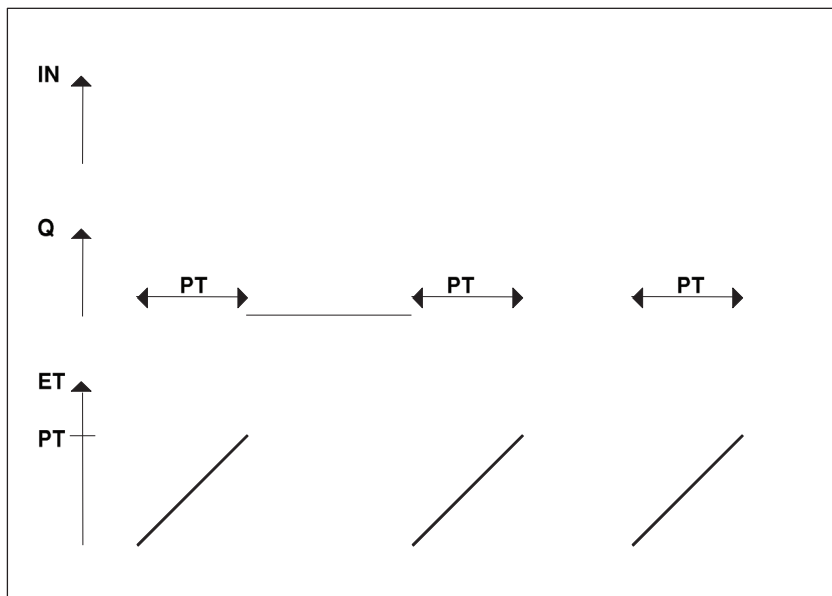
SFB 3 "TP"生成了长度为 PT 的脉冲。定时器只在 STARTUP 和 RUN 模式下运行。

输入 IN 的信号上升沿启动脉冲。在时间 PT 内输出 Q 将保持置位状态，无论输入信号是否有变化(换言之，即使 IN 输入在时间 PT 用完前从 0 变回 1)。ET 输出提供输出 Q 处于置位状态的时间。ET 输出的最大值是 PT 输入的值。输出 ET 在输入 IN 变为 0 时复位，但复位不会在时间 PT 用完前进行。

SFB 3 "TP"符合 IEC 1131-3 标准。

操作系统会在冷启动期间重置 SFB 3 "TP"的实例。如果希望在暖启动后初始化此 SFB 的实例，必须在 OB100 中使用 PT = 0 ms 调用 SFB 3。如果此 SFB 的实例位于另一个块中，可以，例如通过初始化更高级别的块，重置这些实例。

#### 时序图



参数	声明	数据类型	存储器区	描述
IN	INPUT	BOOL	I、Q、M、D、L、常数	使能输入
PT	INPUT	TIME	I、Q、M、D、L、常数	脉冲的持续时间。PT 必须为正值。(注意：值范围由 TIME 数据类型确定)
Q	OUTPUT	BOOL	I、Q、M、D、L	时间的状态
ET	OUTPUT	TIME	I、Q、M、D、L	用完的时间

## 26.2 使用 SFB 4 "TON"生成接通延迟

### 描述

SFB 4 "TON"使信号上升沿延迟时间 PT。定时器只在 STARTUP 和 RUN 模式下运行。

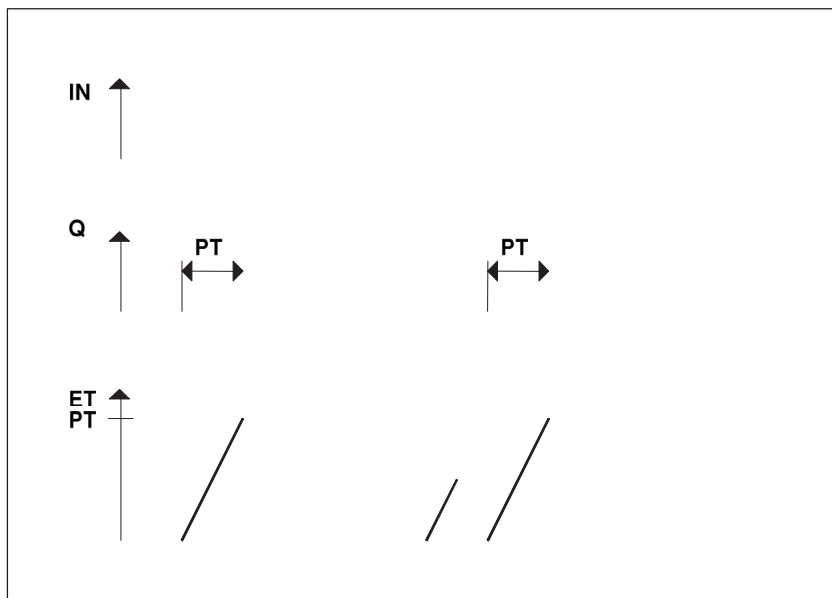
IN 输入的上升沿会导致时间 PT 用完后输出 Q 出现上升沿。随后 Q 将保持置位状态，直到 IN 输入再次变为 0。如果 IN 输入在时间 PT 用完前变为 0，输出 Q 将保持其设置 0 不变。

ET 输出提供 IN 输入的上一上升沿后过去的时间。其最大值为 PT 输入的值。IN 输入变为 0 时将复位 ET。

SFB 4 "TON"符合 IEC 1131-3 标准。

操作系统会在冷启动期间重置 SFB 4 "TON"的实例。如果希望在暖启动后初始化此 SFB 的实例，必须在 OB100 中使用 PT = 0 ms 调用 SFB 4。如果此 SFB 的实例位于另一个块中，可以，例如通过初始化更高级别的块，重置这些实例。

### 时序图



参数	声明	数据类型	存储器区	描述
IN	INPUT	BOOL	I、Q、M、D、L、 常数	使能输入
PT	INPUT	TIME	I、Q、M、D、L、 常数	IN 输入上升沿延迟的时间。PT 必须为正值。 (注意：值范围由 TIME 数据类型确定)
Q	OUTPUT	BOOL	I、Q、M、D、L	时间的状态
ET	OUTPUT	TIME	I、Q、M、D、L	用完的时间

## 26.3 使用 SFB 5 "TOF"生成断开延迟

### 描述

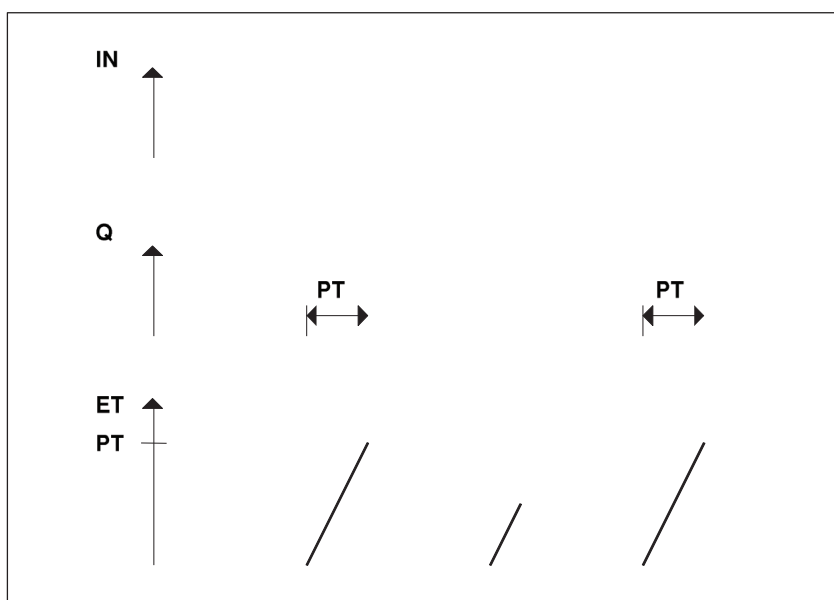
SFB 5 "TOF"使下降沿延迟时间 PT。定时器只在 STARTUP 和 RUN 模式下运行。

IN 输入的上升沿会导致输出 Q 出现上升沿。IN 输入的下降沿会导致输出 Q 的下降沿延迟时间 PT。如果 IN 输入在时间 PT 用完前变为 1，输出 Q 将保持其设置 1 不变。ET 输出提供 IN 输入的上一下降沿后过去的时间。但其最大值为 PT 输入的值。IN 输入变为 1 时将重置 ET。

SFB 5 "TOF"符合 IEC 1131-3 标准。

操作系统会在冷启动期间重置 SFB 5 "TOF"的实例。如果希望在暖启动后初始化此 SFB 的实例，必须在 OB100 中使用 PT = 0 ms 调用 SFB 5。如果此 SFB 的实例位于另一个块中，可以，例如通过初始化更高级别的块，重置这些实例。

### 时序图



参数	声明	数据类型	存储器区	描述
IN	INPUT	BOOL	I、Q、M、D、L、常数	使能输入
PT	INPUT	TIME	I、Q、M、D、L、常数	IN 输入下降沿延迟的时间。PT 必须为正值。 (注意：值范围由 TIME 数据类型确定)
Q	OUTPUT	BOOL	I、Q、M、D、L	时间的状态
ET	OUTPUT	TIME	I、Q、M、D、L	用完的时间



## 26.4 使用 SFB 0 "CTU" 递增计数

### 描述

可以使用 SFB 0 "CTU" 递增计数。CU 输入出现上升沿将使计数器递增 1(与上一 SFB 调用相比)。如果计数值达到上限 32767，将不再递增。CU 输入出现任何后续上升沿都将不再起作用。

不管 CU 输入的当前值是什么，R 输入信号级别为 1 时将把计数器重置为 0。

Q 输出指示当前计数值是否大于或等于预设值 PV。

SFB 0 "CTU" 符合 IEC 1131-3 标准。

操作系统会在冷启动期间重置 SFB 0 "CTU" 的实例。如果希望在暖启动后初始化此 SFB 的实例，必须在 OB100 中使用 R = 1 调用 SFB 0。如果此 SFB 的实例位于另一个块中，可以，例如通过初始化更高级别的块，重置这些实例。

参数	声明	数据类型	存储器区	描述
CU	INPUT	BOOL	I、Q、M、D、L、常数	计数器输入
R	INPUT	BOOL	I、Q、M、D、L、常数	重置输入 R 优先于 CU。
PV	INPUT	INT	I、Q、M、D、L、常数	预设值。有关 PV 的作用，请参考参数 Q。
Q	OUTPUT	BOOL	I、Q、M、D、L	计数器的状态：Q 具有下列值 <ul style="list-style-type: none"> <li>• 1，如果 <math>CV \geq PV</math></li> <li>• 0，其它情况</li> </ul>
CV	OUTPUT	INT	I、Q、M、D、L	当前计数值 (允许值：0 至 32767)

## 26.5 使用 SFB 1 "CTD"递减计数

### 描述

可以使用 SFB 1 "CTD"递减计数。CD 输入出现上升沿将使计数器递减(与上一 SFB 调用相比)。如果计数值达到下限 -32768，将不再递减。CD 输入再出现任何上升沿将不再起作用。

不管 CD 输入的值是多少，LOAD 输入信号级别为 1 时将把计数器设置为预设值 PV。

Q 输出指示当前计数值是否小于或等于 0。

SFB 1 "CTD"符合 IEC 1131-3 标准。

操作系统会在冷启动期间重置 SFB 1 "CTD"的实例。如果希望在暖启动后初始化此 SFB 的实例，必须在 OB100 中使用 LOAD = 1 和 PV = 所需的 CV 初始值调用 SFB 1。如果此 SFB 的实例位于另一个块中，可以，例如通过初始化更高级别的块，重置这些实例。

参数	声明	数据类型	存储器区	描述
CD	INPUT	BOOL	I、Q、M、D、L、常数	计数输入
LOAD	INPUT	BOOL	I、Q、M、D、L、常数	加载输入。LOAD 优先于 CD。
PV	INPUT	INT	I、Q、M、D、L、常数	预设值。LOAD 输入的信号级别为 1 时，将把计数器预置为 PV。
Q	OUTPUT	BOOL	I、Q、M、D、L	计数器的状态：Q 具有下列值： <ul style="list-style-type: none"> <li>• 1，如果 <math>CV \leq 0</math></li> <li>• 0，其它情况</li> </ul>
CV	OUTPUT	INT	I、Q、M、D、L	当前计数值(允许值：-32768 至 32767)

## 26.6 使用 SFB 2 "CTUD" 递增/递减计数

### 描述

可以使用 SFB 2 "CTUD" 递增和递减计数。出现上升沿时计数值按以下方式变化(与上一 SFB 调用相比)：

- 出现在输入 CU 时递增 1
- 出现在输入 CD 时递减 1。

如果计数值达到极限值，计数器将做如下反应：

- 达到下限 -32768 时将不再递减
- 达到上限 32767 时将不再递增。

如果在一个周期内输入 CU 和输入 CD 都有上升沿，计数器将保持其当前值不变。此反应不符合 IEC 1131-3 标准。在该标准中，如果两个信号同时处于激活状态，CU 输入优先。已向 IEC 提交有关对此更改的提议。

不管 CU 和 CD 输入的值是什么，LOAD 输入的信号级别为 1 时将把计数器预置为值 PV。

不管 CU、CD 和 LOAD 输入的值是什么，R 输入的信号级别为 1 时将把计数器重置为 0。QU 输出指示当前计数值是否大于或等于预设值 PV；QD 输出指示该值是否小于或等于 0。

操作系统会在冷启动期间重置 SFB 2 "CTUD"。如果希望在暖启动后初始化 SFB 2 "CTUD"，必须在 OB100 中使用下列参数调用 SFB 2：

- 使用 R = 1 (使用块递增计数时)
- 使用 R = 0 和 LOAD = 1 及 PV = 必需的 CV 初始值(使用块递减计数时)
- 如果此 SFB 的实例位于另一个块中，可以，例如通过初始化更高级别的块，重置这些实例。

参数	声明	数据类型	存储器区	描述
CU	INPUT	BOOL	I、Q、M、D、L、常数	递增计数输入。
CD	INPUT	BOOL	I、Q、M、D、L、常数	降值计数输入
R	INPUT	BOOL	I、Q、M、D、L、常数	重置输入。R 优先于 LOAD。
LOAD	INPUT	BOOL	I、Q、M、D、L、常数	加载输入。LOAD 优先于 CU 和 CD。
PV	INPUT	INT	I、Q、M、D、L、常数	预设值。LOAD 输入的信号级别为 1 时，将把计数器设置为预设值 PV。
QU	OUTPUT	BOOL	I、Q、M、D、L	递增计数器的状态： QU 具有下列值 <ul style="list-style-type: none"> <li>• 1，如果 <math>CV \geq PV</math></li> <li>• 0，其它情况</li> </ul>
QD	OUTPUT	BOOL	I、Q、M、D、L	递减计数器的状态： QD 具有下列值 <ul style="list-style-type: none"> <li>• 1，如果 <math>CV \leq 0</math></li> <li>• 0，其它情况</li> </ul>
CV	OUTPUT	INT	I、Q、M、D、L	当前计数值 (允许值：-32768 至 32767)



## 27 IEC 功能

### 27.1 概述

您可以将以下国际电工委员会(IEC)功能从 STEP 7 库"S7libs\Stdlib30"复制到程序目录。

名称	IEC 块系列	功能
FC3 D_TOD_DT	转换	将 DATE 和 TIME_OF_DAY 组合为 DT
FC6 DT_DATE	转换	从 DT 提取 DATE
FC7 DT_DAY	转换	从 DT 提取星期
FC8 DT_TOD	转换	从 DT 提取 TIME_OF_DAY
FC33 S5TI_TIM	转换	数据类型转换, 从 S5TIME 到 TIME
FC40 TIM_S5TI	转换	数据类型转换, 从 TIME 到 S5TIME
FC16 I_STRNG	转换	数据类型转换, 从 INT 到 STRING
FC5 DI_STRNG	转换	数据类型转换, 从 DINT 到 STRING
FC30 R_STRNG	转换	数据类型转换, 从 REAL 到 STRING
FC38 STRNG_I	转换	数据类型转换, 从 STRING 到 INT
FC37 STRNG_DI	转换	数据类型转换, 从 STRING 到 DINT
FC39 STRNG_R	转换	数据类型转换, 从 STRING 到 REAL
FC9 EQ_DT	DT	比较 DT, 求等于
FC12 GE_DT	DT	比较 DT, 求大于或等于
FC14 GT_DT	DT	比较 DT, 求大于
FC18 LE_DT	DT	比较 DT, 求小于或等于
FC23 LT_DT	DT	比较 DT, 求小于
FC28 NE_DT	DT	比较 DT, 求不等于
FC10 EQ_STRNG	字符串	比较 STRING, 求等于
FC13 GE_STRNG	字符串	比较 STRING, 求大于或等于
FC15 GT_STRNG	字符串	比较 STRING, 求大于
FC19 LE_STRNG	字符串	比较 STRING, 求小于或等于
FC24 LT_STRNG	字符串	比较 STRING, 求小于
FC29 NE_STRNG	字符串	比较 STRING, 求不等于
FC21 LEN	字符串	STRING 变量的长度
FC20 LEFT	字符串	STRING 变量的左侧部分
FC32 RIGHT	字符串	STRING 变量的右侧部分
FC26 MID	字符串	STRING 变量的中间部分
FC2 CONCAT	字符串	组合两个 STRING 变量
FC17 INSERT	字符串	在 STRING 变量中插入
FC4 DELETE	字符串	在 STRING 变量中删除
FC31 REPLACE	字符串	在 STRING 变量中替换
FC11 FIND	字符串	在 STRING 变量中查找
FC1 AD_DT_TM	浮点数数学运算	时间加持续时间

名称	IEC 块系列	功能
FC35 SB_DT_TM	浮点数数学运算	时间减持续时间
FC34 SB_DT_DT	浮点数数学运算	两个时间值相减
FC22 LIMIT	浮点数数学运算	限定值
FC25 MAX	浮点数数学运算	选择最大值
FC27 MIN	浮点数数学运算	选择最小值
FC36 SEL	浮点数数学运算	二进制选择

有关符合 IEC 的通信块的更多信息，请参考系统功能块(SFB/SFC)的上下文相关帮助中的通信功能 (参见：S7通信块和 S7 基本通信块之间的差别)

## 27.2 IEC 功能的技术数据

### 存储器要求

下表显示每个国际电工技术委员会(IEC)功能所需要的工作存储器字节数、装入存储器字节数和本地数据所需的字节数。

功能号	名称	大小(字节数)		本地数据(字节数)
		工作存储器	装载存储器	
FC3	D_TOD_DT	634	810	12
FC6	DT_DATE	340	466	10
FC7	DT_DAY	346	472	10
FC8	DT_TOD	114	210	6
FC33	S5TI_TIM	94	208	2
FC40	TIM_S5TI	104	208	6
FC16	I_STRNG	226	340	10
FC5	DI_STRNG	314	440	18
FC30	R_STRNG	528	684	28
FC38	STRNG_I	292	420	12
FC37	STRNG_DI	310	442	12
FC39	STRNG_R	828	1038	30
FC9	EQ_DT	96	194	2
FC12	GE_DT	174	288	4
FC14	GT_DT	192	310	4
FC18	LE_DT	168	280	4
FC23	LT_DT	192	310	4
FC28	NE_DT	96	194	2
FC10	EQ_STRNG	114	220	4
FC13	GE_STRNG	162	282	8
FC15	GT_STRNG	158	278	8
FC19	LE_STRNG	162	282	8
FC24	LT_STRNG	158	278	8
FC29	NE_STRNG	150	266	8
FC21	LEN	38	132	2
FC20	LEFT	200	320	8
FC32	RIGHT	230	350	8
FC26	MID	302	390	8
FC2	CONCAT	358	452	14
FC17	INSERT	488	644	20
FC4	DELETE	376	512	8
FC31	REPLACE	562	726	20
FC11	FIND	236	360	14

功能号	名称	大小(字节数)		本地数据(字节数)
		工作存储器	装载存储器	
FC1	AD_DT_TM	1350	1590	22
FC35	SB_DT_TM	1356	1596	22
FC34	SB_DT_DT	992	1178	30
FC22	LIMIT	426	600	12
FC25	MAX	374	532	8
FC27	MIN	374	532	8
FC36	SEL	374	560	8



## 27.3 日期和时间作为复杂数据类型

### DATE\_AND\_TIME 的实际参数

DATE\_AND\_TIME 数据类型和 ARRAY、STRING 和 STRUCT 一样均属于复杂数据类型类别。复杂数据类型的允许存储区域是数据块(D)区域和本地数据(L)区域。

因为 DATE\_AND\_TIME 是复杂数据类型，在语句中使用 DATE\_AND\_TIME 作为形式参数时只能以下列形式提供实际参数：

- 指定块的变量声明表中的特定于块的符号
- 数据块的符号名称，例如"DB\_sys\_info.System\_Time"，由以下部分组成：
  - 在符号表中为数据块号定义的名称(例如 DB5 的名称为"DB\_sys\_info")
  - 在数据块中为 DATE\_AND\_TIME 元素定义的名称(例如 DB5 中含有的 DATE\_AND\_TIME 数据类型变量的名称为"System\_Time")

不能将常数作为实际参数传送给复杂数据类型(包括 DATE\_AND\_TIME)的形式参数。也不能将绝对地址作为实际参数传送给 DATE\_AND\_TIME。

## 27.4 Time-of-Day 功能

### FC1 AD\_DT\_TM

功能 FC1 向一个时间(格式 DT)值加上一段时间(格式 TIME),产生一个新的时间值(格式 DT)。时间(参数 T)必须设在 DT#1990-01-01-00:00:00.000 到 DT#2089-12-31-23:59:59.999 范围之内。功能不运行输入检查。如果加法结果超出了有效范围,则将结果限制为相应值,并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
T	INPUT	DATE_AND_TIME	D、L	DT 格式的时间
D	INPUT	TIME	I、Q、M、D、L、常数	TIME 格式的持续时间
RET_VAL	OUTPUT	DATE_AND_TIME	D、L	总和, 格式为 DT

只能为输入参数 T 和输出参数分配一个符号定义的变量。

### 描述 FC3 D\_TOD\_DT

功能 FC3 将数据格式 DATE 和 TIME\_OF\_DAY (TOD)组合在一起,并将这些格式转换为数据类型格式 DATE\_AND\_TIME (DT)。输入值 IN1 必须在限定值 DATE#1990-01-01 和 DATE#2089-12-31 之间。(不检查此值。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
IN1	INPUT	DATE	I、Q、M、D、L、常数	输入变量, 格式为 DATE
IN2	INPUT	TIME_OF_DAY	I、Q、M、D、L、常数	输入变量, 格式为 TOD
RET_VAL	OUTPUT	DATE_AND_TIME	D、L	DT 格式的返回值

只能为返回值分配一个符号定义的变量。

### 描述 FC6 DT\_DATE

功能 FC6 从格式 DATE\_AND\_TIME 提取数据类型格式 DATE。DATE 值必须在限定值 DATE#1990-1-1 和 DATE#2089-12-31 之间。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
IN	INPUT	DATE_AND_TIME	D、L	输入变量, 格式为 DT
RET_VAL	OUTPUT	DATE	I、Q、M、D、L	返回值, 格式为 DATE

只能为输入分配一个符号定义的变量。

**描述 FC7 DT\_DAY**

功能 FC7 从格式 DATE\_AND\_TIME 提取周时。以数据类型格式 INTEGER 提供周时：

- 1 = 星期日
- 2 = 星期一
- 3 = 星期二
- 4 = 星期三
- 5 = 星期四
- 6 = 星期五
- 7 = 星期六

此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
IN	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
RET_VAL	OUTPUT	INT	I、Q、M、D、L	返回值，格式为 INT

只能为输入分配一个符号定义的变量。

**描述 FC8 DT\_TOD**

功能 FC8 从格式 DATE\_AND\_TIME 提取数据类型格式 TIME\_OF\_DAY。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
IN	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
RET_VAL	OUTPUT	TIME_OF_DAY	I、Q、M、D、L	返回值的格式为 TOD

只能为输入分配一个符号定义的变量。

**描述 FC33 S5TI\_TIM**

功能 FC33 将数据类型格式 S5TIME 转换为格式 TIME。如果转换结果超出 TIME 范围，则将结果限制为相应值，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
IN	INPUT	S5TIME	I、Q、M、D、L、常数	输入变量，格式为 S5TIME
RET_VAL	OUTPUT	TIME	I、Q、M、D、L	返回值，格式为 TIME

**描述 FC34 SB\_DT\_DT**

功能 FC34 将两个时间值(格式 DT)相减，结果为持续时间(格式 TIME)。时间必须设在 DT#1990-01-01-00:00:00.000 到 DT#2089-12-31-23:59:59.999 范围之内。功能不运行输入检查。如果第一个时间(参数 T1)大于(较晚于)第二个时间(参数 DT2)，结果为正；如果第一个时间小于(较早于)第二个时间，结果为负。如果减法结果超出 TIME 范围，则将结果限制到相应值，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
DT1	INPUT	DATE_AND_TIME	D、L	第一个时间，格式为 DT
DT2	INPUT	DATE_AND_TIME	D、L	第二个时间，格式为 DT
RET_VAL	OUTPUT	TIME	I、Q、M、D、L	差，格式为 TIME

只能为输入参数分配一个符号定义的变量。

**描述 FC35 SB\_DT\_TM**

功能 FC35 从一个时间值中(格式 DT)减去持续时间(格式 TIME)，结果产生一个新的时间值(格式 DT)。时间(参数 T)必须设在 DT#1990-01-01-00:00:00.000 到 DT#2089-12-31-23:59:59.999 范围之内。功能不运行输入检查。如果减法结果超出了有效范围，则将结果限制为相应值，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
T	INPUT	DATE_AND_TIME	D、L	DT 格式的时间
D	INPUT	TIME	I、Q、M、D、L、常数	TIME 格式的持续时间
RET_VAL	OUTPUT	DATE_AND_TIME	D、L	差，格式为 DT

只能为输入参数 T 和输出参数分配一个符号定义的变量。

**描述 FC40 TIM\_S5TI**

功能 FC40 将数据类型格式 TIME 转换为格式 S5TIME。在转换中进行四舍五入。如果输入参数大于 S5TIME 格式的允许范围(大于 TIME#02:46:30.000)，则输出结果为 S5TIME#999.3，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
IN	INPUT	TIME	I、Q、M、D、L、常数	输入变量，格式为 TIME
RET_VAL	OUTPUT	S5TIME	I、Q、M、D、L	返回值，格式为 S5TIME

## 27.5 比较 DATE\_AND\_TIME 变量

### 描述 FC9 EQ\_DT

功能 FC9 比较两个 DATE\_AND\_TIME 数据类型格式变量的内容，检查它们是否相等，并将比较结果输出为返回值。如果参数 DT1 的时间与参数 DT2 的时间相等，则返回值的信号状态为"1"。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
DT1	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
DT2	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

### 描述 FC12 GE\_DT

功能 FC12 比较两个 DATE\_AND\_TIME 数据类型格式变量的内容，检查其中一个是否大于或等于另一个，并将比较结果作为返回值。如果参数 DT1 的时间大于(晚于)参数 DT2 的时间，或者两个时间相同，则返回值的信号状态为"1"。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
DT1	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
DT2	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

### 描述 FC14 GT\_DT

功能 FC14 比较两个 DATE\_AND\_TIME 数据类型格式变量的内容，检查其中一个是否大于另一个，并将比较结果作为返回值。如果参数 DT1 的时间大于(晚于)参数 DT2 的时间，则返回值的信号状态为"1"。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
DT1	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
DT2	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

**描述 FC18 LE\_DT**

功能 FC18 比较两个 DATE\_AND\_TIME 数据类型格式变量的内容，检查其中一个是否小于或等于另一个，并将比较结果作为返回值。如果参数 DT1 的时间小于(早于)参数 DT2 的时间，或者两个时间相同，则返回值的信号状态为"1"。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
DT1	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
DT2	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

**描述 FC23 LT\_DT**

功能 FC23 比较两个 DATE\_AND\_TIME 数据类型格式变量的内容，检查其中一个是否小于另一个，并将比较结果作为返回值。如果参数 DT1 的时间小于(早于)参数 DT2 的时间，则返回值的信号状态为"1"。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
DT1	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
DT2	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

**描述 FC28 NE\_DT**

功能 FC28 比较两个 DATE\_AND\_TIME 数据类型格式变量的内容，检查它们是否不相等，并将比较结果作为返回值。如果参数 DT1 的时间不等于参数 DT2 的时间，则返回值的信号状态为"1"。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
DT1	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
DT2	INPUT	DATE_AND_TIME	D、L	输入变量，格式为 DT
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

## 27.6 编辑 STRING 变量

### 描述 FC2 CONCAT

功能 FC2 将两个 STRING 变量连接在一起，组成一个字符串。如果结果字符串比在输出参数中提供的变量长，则将结果字符串限制到设置的最大长度，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
IN1	INPUT	STRING	D、L	输入变量，格式为 STRING
IN2	INPUT	STRING	D、L	输入变量，格式为 STRING
RET_VAL	OUTPUT	STRING	D、L	连接字符串

只能为参数分配一个符号定义的变量。

### 描述 FC4 DELETE

功能 FC4 从字符串中位置 P 的字符(含此字符)开始删除多个(L)字符。如果 L 和/或 P 等于零，或者 P 大于输入字符串的当前长度，则返回输入字符串。如果 L 加 P 的和大于输入字符串，则一直删除到输入字符串的末尾。如果 L 和/或 P 为负数，则返回空字符串，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
IN	INPUT	STRING	D、L	要从其中进行删除操作的 STRING 变量
L	INPUT	INT	I、Q、M、D、L、常数	要删除的字符数
P	INPUT	INT	I、Q、M、D、L、常数	要删除的第一个字符的位置
RET_VAL	OUTPUT	STRING	D、L	结果字符串

只能为输入参数 IN 和输出参数分配一个符号定义的变量。

### 描述 FC11 FIND

功能 FC11 提供第二个字符串(IN2)在第一个字符串(IN1)中的位置。搜索从左侧开始；报告字符串第一次出现的位置。如果在第一个字符串中未找到第二个字符串，则返回零。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
IN1	INPUT	STRING	D、L	要在其中进行查找的 STRING 变量
IN2	INPUT	STRING	D、L	要查找的 STRING 变量
RET_VAL	OUTPUT	INT	I、Q、M、D、L	找到的字符串位置

只能为输入参数 IN1 和 IN2 分配一个符号定义的变量。

**描述 FC17 INSERT**

功能 FC17 将参数 IN2 的字符串插入到参数 IN1 的字符串中位置 P 的字符之后。如果 P 等于零，则将第二个字符串插入到第一个字符串之前。如果 P 大于第一个字符串的当前长度，则将第二个字符串附加到第一个字符串。如果 P 为负，则输出空字符串，并将状态字的二进制结果(BR)位设置为"0"。如果结果字符串大于输出参数所给定的变量，二进制结果位也将被设为"0"。在这种情况下，结果字符串将被限定为最大设定长度。

参数	声明	数据类型	存储器区	描述
IN1	INPUT	STRING	D、L	要向其中进行插入操作的 STRING 变量
IN2	INPUT	STRING	D、L	要插入的 STRING 变量
P	INPUT	INT	I、Q、M、D、L、常数	插入位置
RET_VAL	OUTPUT	STRING	D、L	结果字符串

只能为输入参数 IN1 和 IN2 及输出参数分配一个符号定义的变量。

**描述 FC20 LEFT**

功能 FC20 提供字符串的前 L 个字符(其中 L 代表数字)。如果 L 大于 STRING 变量的当前长度，则返回输入值。如果 L = 0 并且输入值为空字符串，则返回空字符串。如果 L 为负数，则返回空字符串，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
IN	INPUT	STRING	D、L	输入变量，格式为 STRING
L	INPUT	INT	I、Q、M、D、L、常数	左侧字符串的长度
RET_VAL	OUTPUT	STRING	D、L	输出变量，格式为 STRING

只能为参数 IN 和返回值分配一个符号定义的变量。

**描述 FC21 LEN**

STRING 变量包含两个长度：最大长度(定义变量时在方括号中给出)和当前长度(当前有效字符数)。当前长度必须小于或等于最大长度。字符串占用的字节数为最大长度加 2。

功能 FC21 输出字符串的当前长度(有效字符数)作为返回值。空字符串(' ')的长度为零。最大长度为 254。此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
S	INPUT	STRING	D、L	输入变量，格式为 STRING
RET_VAL	OUTPUT	INT	I、Q、M、D、L	有效字符数

只能为输入参数分配一个符号定义的变量。



**描述 FC26 MID**

功能 FC26 提供字符串的中间部分(从第 P 个字符(含第 P 个字符)开始的 L 个字符)。如果 L 加 P 的和超出 STRING 变量的当前长度,则返回的字符串为从第 P 个字符到输入值末尾。在所有其它情况下(P 超出当前长度, P 和/或 L 等于零或为负),返回空字符串,并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
IN	INPUT	STRING	D、L	输入变量, 格式为 STRING
L	INPUT	INT	I、Q、M、D、L、常数	中间字符串的长度
P	INPUT	INT	I、Q、M、D、L、常数	第一个字符的位置
RET_VAL	OUTPUT	STRING	D、L	输出变量, 格式为 STRING

只能为参数 IN 和返回值分配一个符号定义的变量。

**描述 FC31 REPLACE**

功能 FC31 使用整个第二个字符串(IN2)替换第一个字符串(IN1)中从位置 P (包含该位置)开始的多个字符(L)。若 L 等于零, P 不等于零, 则返回第一个字符串。若 L 等于零, P 等于零, 则在第一个字符串之前添加第二个字符串。若 L 不等于零, P 等于零或一, 则从第一个字符(含此字符)开始代替该字符串。如果 P 超出第一个字符串, 则将第二个字符串附加到第一个字符串。如果 L 和/或 P 为负数, 则返回空字符串, 并将状态字的二进制结果(BR)位设置为"0"。如果结果字符串大于输出参数所给定的变量, 二进制结果位也将被设为"0"。在这种情况下, 结果字符串将被限定为最大设定长度。

参数	声明	数据类型	存储器区	描述
IN1	INPUT	STRING	D、L	要在其中进行替换的 STRING 变量
IN2	INPUT	STRING	D、L	要插入的 STRING 变量
L	INPUT	INT	I、Q、M、D、L、常数	要替换的字符数
P	INPUT	INT	I、Q、M、D、L、常数	要替换的第一个字符的位置
RET_VAL	OUTPUT	STRING	D、L	结果字符串

只能为输入参数 IN1 和 IN2 及输出参数分配一个符号定义的变量。

**描述 FC32 RIGHT**

功能 FC32 提供字符串的后 L 个字符(其中 L 代表数字)。如果 L 大于 STRING 变量的当前长度，则返回输入值。如果 L = 0 并且输入值为空字符串，则返回空字符串。如果 L 为负数，则返回空字符串，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
IN	INPUT	STRING	D、L	输入变量，格式为 STRING
L	INPUT	INT	I、Q、M、D、L、常数	右侧字符串的长度
RET_VAL	OUTPUT	STRING	D、L	输出变量，格式为 STRING

只能为参数 IN 和返回值分配一个符号定义的变量。

## 27.7 比较 STRING 变量

### 描述 FC10 EQ\_STRNG

功能 FC10 比较两个 STRING 数据类型格式变量的内容，检查它们是否相等，并将比较结果作为返回值。如果参数 S1 的字符串与参数 S2 的字符串相同，则返回值的信号状态为"1"。

此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
S1	INPUT	STRING	D、L	输入变量，格式为 STRING
S2	INPUT	STRING	D、L	输入变量，格式为 STRING
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

### 描述 FC13 GE\_STRNG

功能 FC13 比较两个 STRING 数据类型格式变量的内容，检查第一个是否大于或等于另一个，并将比较结果作为返回值。如果参数 S1 的字符串大于或等于参数 S2 的字符串，则返回值的信号状态为"1"。

使用字符的 ASCII 码比较字符(例如"a"大于"A")。从左开始比较。第一个不同的字符决定比较结果。如果开始字符都相同，则较长的字符串较大。

此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
S1	INPUT	STRING	D、L	输入变量，格式为 STRING
S2	INPUT	STRING	D、L	输入变量，格式为 STRING
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

**描述 FC15 GT\_STRNG**

功能 FC15 比较两个 STRING 数据类型格式变量的内容，检查第一个是否大于另一个，并将比较结果作为返回值。如果参数 S1 的字符串大于参数 S2 的字符串，则返回值的信号状态为"1"。

使用字符的 ASCII 码比较字符(例如"a"大于"A")。从左开始比较。第一个不同的字符决定比较结果。如果开始字符都相同，则较长的字符串较大。

此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
S1	INPUT	STRING	D、L	输入变量，格式为 STRING
S2	INPUT	STRING	D、L	输入变量，格式为 STRING
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

**描述 FC19 LE\_STRNG**

功能 FC19 比较两个 STRING 数据类型格式变量的内容，检查第一个是否小于或等于另一个，并将比较结果作为返回值。如果参数 S1 的字符串小于或等于参数 S2 的字符串，则返回值的信号状态为"1"。

使用字符的 ASCII 码比较字符(例如"a"小于"A")。从左开始比较。第一个不同的字符决定比较结果。如果开始字符都相同，则较短的字符串较小。

此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
S1	INPUT	STRING	D、L	输入变量，格式为 STRING
S2	INPUT	STRING	D、L	输入变量，格式为 STRING
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

**描述 FC24 LT\_STRNG**

功能 FC24 比较两个 STRING 数据类型格式变量的内容，检查第一个是否小于另一个，并将比较结果作为返回值。如果参数 S1 的字符串小于参数 S2 的字符串，则返回值的信号状态为"1"。

使用字符的 ASCII 码比较字符(例如"a"小于"A")。从左开始比较。第一个不同的字符决定比较结果。如果较长字符串的左侧部分和较短字符串相同,则较短的字符串较小。

此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
S1	INPUT	STRING	D、L	输入变量，格式为 STRING
S2	INPUT	STRING	D、L	输入变量，格式为 STRING
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

**描述 FC29 NE\_STRNG**

功能 FC29 比较两个 STRING 数据类型格式变量的内容，检查它们是否不同，并将比较结果作为返回值。如果参数 S1 的字符串与参数 S2 的字符串不同，则返回值的信号状态为"1"。

此功能不报告任何错误。

参数	声明	数据类型	存储器区	描述
S1	INPUT	STRING	D、L	输入变量，格式为 STRING
S2	INPUT	STRING	D、L	输入变量，格式为 STRING
RET_VAL	OUTPUT	BOOL	I、Q、M、D、L	比较结果

只能为输入参数分配一个符号定义的变量。

## 27.8 转换数据类型格式

### 描述 FC5 DI\_STRNG

功能 FC5 将 DINT 数据类型格式变量转换为字符串。显示的字符串以符号开头。如果在返回参数中给出的变量太短，则不进行转换，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
I	INPUT	DINT	I、Q、M、D、L、常数	输入值
RET_VAL	OUTPUT	STRING	D、L	结果字符串

只能为输出参数分配一个符号定义的变量。

### 描述 FC16 I\_STRNG

功能 FC16 将 INT 数据类型格式变量转换为字符串。显示的字符串以符号开头。如果在返回参数中给出的变量太短，则不进行转换，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
I	INPUT	INT	I、Q、M、D、L、常数	输入值
RET_VAL	OUTPUT	STRING	D、L	结果字符串

只能为输出参数分配一个符号定义的变量。

### 描述 FC30 R\_STRNG

功能 FC30 将 REAL 数据类型格式变量转换为字符串。显示的字符串有 14 位：

±v.nnnnnnnE±xx	±	符号
	v	小数点前的 1 位数字
	n	小数点后的 7 位数字
	x	2 位指数数字

如果在返回参数中给出的变量太短，或者参数 IN 未提供有效的浮点数，则不进行转换，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
IN	INPUT	REAL	I、Q、M、D、L、常数	输入值
RET_VAL	OUTPUT	STRING	D、L	结果字符串

只能为输出参数分配一个符号定义的变量。

**描述 FC37 STRNG\_DI**

功能 FC37 将字符串转换为 DINT 数据类型格式变量。字符串中的第一个字符可以是符号或数字，随后的字符必须是数字。如果字符串的长度等于零或大于 11，或者字符串中有无效字符，则不进行转换，并将状态字的二进制结果(BR)位设置为"0"。如果转换结果超出 DINT 范围，则将结果限制到相应值，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
S	INPUT	STRING	D、L	输入字符串
RET_VAL	OUTPUT	DINT	I、Q、M、D、L	结果

只能为输入参数分配一个符号定义的变量。

**描述 FC38 STRNG\_I**

功能 FC38 将字符串转换为 INT 数据类型格式变量。字符串中的第一个字符可以是符号或数字，随后的字符必须是数字。如果字符串的长度等于零或大于 6，或者字符串中有无效字符，则不进行转换，并将状态字的二进制结果(BR)位设置为"0"。如果转换结果超出 INT 范围，则将结果限制到相应值，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
S	INPUT	STRING	D、L	输入字符串
RET_VAL	OUTPUT	INT	I、Q、M、D、L	结果

只能为输入参数分配一个符号定义的变量。

**描述 FC39 STRNG\_R**

功能 FC39 将字符串转换为 REAL 数据类型格式变量。字符串必须为以下格式：

±v.nnnnnnnE±xx	±	符号
	v	小数点前的 1 位数字
	n	小数点后的 7 位数字
	x	2 位指数数字

如果字符串长度小于 14，或者不是如上所示的结构，则不进行任何转换，并将状态字的二进制结果(BR)位设置为"0"。如果转换结果超出 REAL 范围，则将结果限制到相应值，并将状态字的二进制结果(BR)位设置为"0"。

参数	声明	数据类型	存储器区	描述
S	INPUT	STRING	D、L	输入字符串
RET_VAL	OUTPUT	REAL	I、Q、M、D、L	结果

只能为输入参数分配一个符号定义的变量。

## 27.9 编辑数值

### 描述 FC22 LIMIT

功能 FC22 将变量的数值限制到可分配有参数的限制值。INT、DINT 和 REAL 数据类型变量可作为输入值。所有分配有参数的变量都必须为相同的数据类型。使用 ANY 指针标识变量类型。下限值(参数 MN)必须小于/等于上限值(参数 MX)。

如果下列任一条件为真，则输出值保持不变，并将状态字的二进制结果(BR)位设置为"0"：

- 已分配有参数的变量的数据类型无效
- 所有分配参数的变量不具有相同的数据类型
- 下限值大于上限值
- REAL 变量不能代表有效浮点数。

参数	声明	数据类型	存储器区	描述
MN	INPUT	ANY	I、Q、M、D、L	下限
IN	INPUT	ANY	I、Q、M、D、L	输入变量
MX	INPUT	ANY	I、Q、M、D、L	上限
RET_VAL	OUTPUT	ANY	I、Q、M、D、L	限定的输出变量

### 描述 FC25 MAX

功能 FC25 选择三个数值型变量值中的最大值。INT、DINT 和 REAL 数据类型变量可作为输入值。所有分配有参数的变量都必须为相同的数据类型。使用 ANY 指针标识变量类型。

如果下列任一条件为真，则输出值保持不变，并将状态字的二进制结果(BR)位设置为"0"：

- 已分配有参数的变量的数据类型无效
- 所有分配参数的变量不具有相同的数据类型
- REAL 变量不能代表有效浮点数。

参数	声明	数据类型	存储器区	描述
IN1	INPUT	ANY	I、Q、M、D、L	第一个输入值
IN2	INPUT	ANY	I、Q、M、D、L	第二个输入值
IN3	INPUT	ANY	I、Q、M、D、L	第三个输入值
RET_VAL	OUTPUT	ANY	I、Q、M、D、L	输入值的最大值



## STL 示例

```

CALL FC 25
  IN1           := P#M 10.0 DINT 1
  IN2           := MD20
  IN3           := P#DB1.DBX 0.0 DINT 1
  RET_VAL      := P#M 40.0 DINT 1
=
M 0.0

```

### 注意

必须使用 ANY 指针输入允许的数据类型 INT、DINT 和 REAL。也允许输入如"MD20"这样的参数，但必须在"Symbol"中定义"MD20"的相应数据类型。

## 描述 FC27 MIN

功能 FC27 选择三个数值型变量值中的最小值。INT、DINT 和 REAL 数据类型变量可作为输入值。所有分配有参数的变量都必须为相同的数据类型。使用 ANY 指针识别变量类型。

如果下列任一条件为真，则输出值保持不变，并将状态字的二进制结果(BR)位设置为"0"：

- 已分配有参数的变量的数据类型无效
- 所有分配参数的变量不具有相同的数据类型
- REAL 变量不能代表有效浮点数。

参数	声明	数据类型	存储器区	描述
IN1	INPUT	ANY	I、Q、M、D、L	第一个输入值
IN2	INPUT	ANY	I、Q、M、D、L	第二个输入值
IN3	INPUT	ANY	I、Q、M、D、L	第三个输入值
RET_VAL	OUTPUT	ANY	I、Q、M、D、L	输入值的最小值

## STL 示例

```

CALL FC 27
  IN1           := P#M 10.0 DINT 1
  IN2           := MD20
  IN3           := P#DB1.DBX 0.0 DINT 1
  RET_VAL      := P#M 40.0 DINT 1
=              M 0.0

```

**注意**

必须使用 ANY 指针输入允许的数据类型 INT、DINT 和 REAL。也允许输入如"MD20"这样的参数，但必须在"Symbol"中定义"MD20"的相应数据类型。

## 描述 FC36 SEL

功能 FC36 根据一个开关(参数 G)在两个变量值中选择其一。符合数据位宽度要求的所有数据类型的变量均可作为参数 IN0 和 IN1 的输入值。例如字节、字和双字(不包括数据类型 DT 和 STRING)。输入变量和输出变量必须为相同的数据类型。

如果下列任一条件为真，则输出值保持不变，并将状态字的二进制结果(BR)位设置为"0"：

- 已分配有参数的变量的数据类型无效
- 所有分配参数的变量不具有相同的数据类型
- REAL 变量不能代表有效浮点数。

参数	声明	数据类型	存储器区	描述
G	INPUT	BOOL	I、Q、M、D、L	选择开关
IN0	INPUT	ANY	I、Q、M、D、L	第一个输入值
IN1	INPUT	ANY	I、Q、M、D、L	第二个输入值
RET_VAL	OUTPUT	ANY	I、Q、M、D、L	所选择的输入值

## 28 用于集成控制的 SFB

### 28.1 使用 SFB 41/FB 41 "CONT\_C"进行连续控制

#### 引言

SFB/FB "CONT\_C" (连续控制器)在 SIMATIC S7 可编程逻辑控制器上使用，通过持续的输入和输出变量来控制工艺过程。在参数分配期间，可以通过激活或取消激活 PID 控制器的子功能使控制器适应过程的需要。使用参数分配工具可以轻松完成分配(菜单路径：**开始 > Simatic > Step7 > 分配 PID 控制参数**)。以 Windows XP 为例，**开始 > Simatic > Step7 > 分配 PID 控制(英文)**中提供了在线电子手册。

#### 应用

可以使用该控制器作为 PID 固定设定值控制器或在多循环控制中作为层叠、混料或比率控制器。该控制器的功能基于使用模拟信号的采样控制器的 PID 控制算法，必要时可以通过加入脉冲发生器阶段进行扩展，为使用成比例执行机构的两个或三个步骤控制器生成脉冲持续时间调制输出信号。

---

#### 注意

只有在以固定时间间隔调用块时，在控制块中计算的才是正确的。为此，应该在周期性中断 OB (OB30 至 OB38)中调用控制块。在 CYCLE 参数中输入采样时间。

---

## 描述

除了设定值和过程值分支中的功能，SFB/FB 还通过持续操作变量输出和手动影响操作值的选项实现了完整的 PID 控制器。

下文提供了对这些子功能的详细说明：

### 设定值分支

以浮点格式在 SP\_INT 输入键入设定值。

### 过程变量分支

可以外设(I/O)或以浮点格式输入过程变量。CRP\_IN 功能根据以下公式将 PV\_PER 外设值转换为介于 -100 和 +100 %间的浮点格式值：

$$\text{Output of CPR\_IN} = \text{PV\_PER} * \frac{100}{27648}$$

PV\_NORM 功能根据以下公式统一 CRP\_IN 输出的格式：

$$\text{PV\_NORM 的输出} = (\text{CPR\_IN 的输出}) * \text{PV\_FAC} + \text{PV\_OFF}$$

PV\_FAC 的缺省值为 1，PV\_OFF 的缺省值为 0。

### 出错信号

设定值和过程变量间的差异就是出错信号。为消除由于操作变量量化导致的小幅恒定振荡(例如，在使用 PULSEGEN 进行脉宽调制时)，将死区应用于出错信号(DEADBAND)。如果 DEADB\_W = 0，将关闭死区。

### PID 算法

PID 算法用于定位计算。比例、积分(INT)和微分(DIF)操作以并联方式连接，因而可以分别激活或取消激活。这使对 P、PI、PD 和 PID 控制器进行组态成为可能。还可以对纯 I 和 D 控制器进行组态。

### 手动值

可以在手动和自动模式间进行切换。在手动模式下，使用手动选择的值更正操作变量。积分器(INT)内部设置为 LMN - LMN\_P - DISV，微分单元(DIF)设置为 0 并在内部进行匹配。这意味着切换到自动模式不会导致操作值发生任何突变。

### 操纵值

使用 LMNLIMIT 功能可以将操作值限制为所选择的值。输入变量超过限制时，信号位会给予指示。LMN\_NORM 功能根据以下公式统一 LMNLIMIT 输出的格式：

$$\text{LMN} = (\text{LMNLIMIT 的输出}) * \text{LMN\_FAC} + \text{LMN\_OFF}$$

LMN\_FAC 的缺省值为 1，LMN\_OFF 的缺省值为 0。

也可以得到外设格式的操作值。CPR\_OUT 功能根据以下公式将浮点值 LMN 转换为外设值：

$$\text{LMN\_PER} = \text{LMN} * \frac{27648}{100}$$

### 前馈控制

可以在 DISV 输入前馈干扰变量。

### 初始化

SFB 41 "CONT\_C"有一个在输入参数 COM\_RST = TRUE 时自动运行的初始化程序。

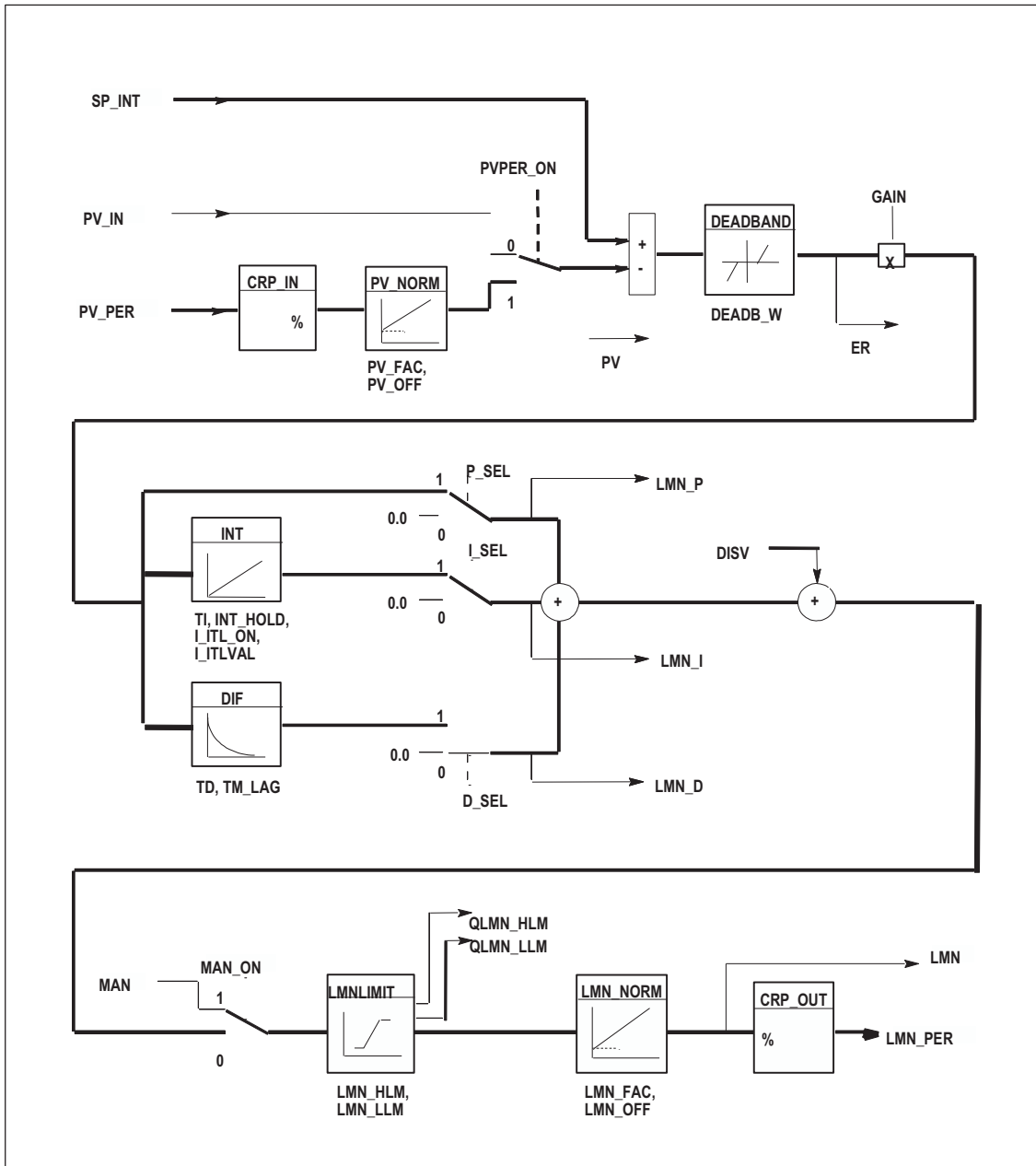
在初始化过程中，将把积分器内部设置为初始化值 I\_ITVAL。以周期性中断优先级调用它时，它会从此值开始继续工作。

将所有其它输出设置为它们各自的缺省值。

### 错误信息

不使用出错输出参数 RET\_VAL。

CONT\_C 方框图



## 输入参数

下表包含对 SFB 41/FB 41 "CONT\_C"输入参数的说明。

参数	数据类型	取值范围：	默认	描述
COM_RST	BOOL		FALSE	COMPLETE RESTART 该块有一个在设置输入 COM_RST 时自动执行的初始化程序。
MAN_ON	BOOL		TRUE	MANUAL VALUE ON 如果设置输入"启用手动值"，将中断控制回路。手动值作为操作值进行设置。
PVPER_ON	BOOL		FALSE	PROCESS VARIABLE PERIPHERAL ON 如果从 I/O 读取过程变量，必须将输入 PV_PER 连接到 I/O，且必须设置输入"启用过程变量外设"。
P_SEL	BOOL		TRUE	PROPORTIONAL ACTION ON 可以在 PID 算法中单独激活或取消激活 PID 操作。如果设置输入"启用比例作用"，将启用 P 操作。
I_SEL	BOOL		TRUE	INTEGRAL ACTION ON 可以在 PID 算法中单独激活或取消激活 PID 操作。如果设置输入"启用积分作用"，将启用 I 操作。
INT_HOLD	BOOL		FALSE	INTEGRAL ACTION HOLD 设置输入"积分作用暂停"可以"冻结"积分器的输出。
I_ITL_ON	BOOL		FALSE	INITIALIZATION OF THE INTEGRAL ACTION ON 设置输入"积分作用初始化打开"可以将积分器的输出连接到输入 I_ITL_VAL。
D_SEL	BOOL		FALSE	DERIVATIVE ACTION ON 可以在 PID 算法中单独激活或取消激活 PID 操作。如果设置输入"启用微分作用"，将启用 D 操作。
CYCLE	TIME	>= 1 毫秒	T#1s	SAMPLING TIME 块调用之间的时间必须为常数。"采样时间"输入指定块调用之间的时间。
SP_INT	REAL	-100.0 至 +100.0 (%)或物理值 1)	0.0	INTERNAL SETPOINT "内部设定值"输入用于指定设定值。
PV_IN	REAL	-100.0 至 +100.0 (%)或物理值 1)	0.0	PROCESS VARIABLE IN 可以在"过程变量输入"输入中设置初始化值，也可以连接浮点格式的外部过程变量。
PV_PER	WORD		W#16#0000	PROCESS VARIABLE PERIPHERAL 将 I/O 格式的过程变量连接到"过程变量外设"输入处的控制器。
MAN	REAL	-100.0 至 +100.0 (%)或物理值 2)	0.0	MANUAL VALUE "手动值"输入用于通过操作员界面功能设置手动值。
GAIN	REAL		2.0	PROPORTIONAL GAIN "比例值"输入指定控制器增益。

参数	数据类型	取值范围：	默认	描述
TI	TIME	>= CYCLE	T#20s	RESET TIME "复位时间"输入决定积分器的时间响应。
TD	TIME	>= CYCLE	T#10s	DERIVATIVE TIME "微分时间"输入决定微分单元的时间响应。
TM_LAG	TIME	>= CYCLE/2	T#2s	TIME LAG OF THE DERIVATIVE ACTION D 操作的算法包括可以在"微分作用的时间延迟"输入分配的时间延迟。
DEADB_W	REAL	>= 0.0 (%) 或物理值 1)	0.0	DEAD BAND WIDTH 将死区应用于出错。"死区带宽"输入决定死区的大小。
LMN_HLM	REAL	LMN_LLM ... 100.0 (%) 或物理值 2)	100.0	MANIPULATED VALUE HIGH LIMIT 操作值始终受上限和下限的限制。"操作值上限"输入指定上限。
LMN_LLM	REAL	-100.0... LMN_HLM (%) 或物理值 2)	0.0	MANIPULATED VALUE LOW LIMIT 操作值始终受上限和下限的限制。"操作值下限"输入指定下限。
PV_FAC	REAL		1.0	PROCESS VARIABLE FACTOR "过程变量因子"输入与过程变量相乘。该输入用于调整过程变量范围。
PV_OFF	REAL		0.0	PROCESS VARIABLE OFFSET 将"过程变量偏移量"输入与过程变量相加。该输入用于调整过程变量范围。
LMN_FAC	REAL		1.0	MANIPULATED VALUE FACTOR 将"操作值因子"输入与操作值相乘。该输入用于调整操作值范围。
LMN_OFF	REAL		0.0	MANIPULATED VALUE OFFSET 将"操作值偏移量"与操作值相加。该输入用于调整操作值范围。
I_ITLVAL	REAL	-100.0 至 +100.0 (%) 或物理值 2)	0.0	INITIALIZATION VALUE OF THE INTEGRAL ACTION 可以在输入 I_ITL_ON 设置积分器的输出。将初始化值应用于输入"积分作用的初始化值"。
DISV	REAL	-100.0 至 +100.0 (%) 或物理值 2)	0.0	DISTURBANCE VARIABLE 为进行前馈控制，将干扰变量连接到输入"干扰变量"。

- 1) 同一单元中设定值和过程变量分支中的参数
- 2) 同一单元中操作值分支中的参数



## 输出参数

下表包含对 SFB 41/FB41 "CONT\_C"输出参数的说明。

参数	数据类型	取值范围：	默认	描述
LMN	REAL		0.0	MANIPULATED VALUE 有效的操作值为"操作值"输出处的浮点格式输出。
LMN_PER	WORD		W#16#0000	MANIPULATED VALUE PERIPHERAL 将 I/O 格式的操作值连接到"操作值外设"输出的控制器。
QLMN_HLM	BOOL		FALSE	HIGH LIMIT OF MANIPULATED VALUE REACHED 操作值始终受上限和下限的限制。如果输出为"达到操作值上限",则表明已超过上限。
QLMN_LLM	BOOL		FALSE	LOW LIMIT OF MANIPULATED VALUE REACHED 操作值始终受上限和下限的限制。如果输出为"达到操作值下限",则表明已超过下限。
LMN_P	REAL		0.0	PROPORTIONAL COMPONENT "比例组件"输出包含操作变量的比例组件。
LMN_I	REAL		0.0	INTEGRAL COMPONENT "积分组件"输出包含操作值的积分组件。
LMN_D	REAL		0.0	DERIVATIVE COMPONENT "微分组件"输出包含操作值的微分组件。
PV	REAL		0.0	PROCESS VARIABLE 有效过程变量为"过程变量"输出处的输出。
ER	REAL		0.0	ERROR SIGNAL 有效出错为"出错信号"输出处的输出。

## 28.2 使用 SFB 42/FB 42 "CONT\_S"进行步控制

### 引言

SFB/FB "CONT\_S" (步控制器)在 SIMATIC S7 可编程逻辑控制器上使用，通过集成执行机构的数字操作值输出信号来控制工艺过程。在参数分配期间，可以通过激活或取消激活 PI 步控制器的子功能使控制器适应过程的需要。使用参数分配工具可以轻松完成分配(菜单路径：**开始 > Simatic > Step7 > 分配 PID 控制参数**)。**开始 > Simatic > Step7 > 分配 PID 控制(英文)**中提供了在线电子手册。

### 应用

可以将该控制器作为 PI 固定设定值控制器使用，或在层叠、混料或比率控制器的辅助控制回路中使用，但不能作为主控制器使用。该控制器的功能基于采样控制器的 PI 控制算法，补充了使用模拟激励信号生成二进制输出信号的功能。

从 CPU 314 IFM 的 FB V1.5 或 V1.1.0 开始增加了以下功能：

TI = T#0ms 时，可以禁用控制器的积分组件，从而可以将块作为比例控制器使用。

由于该控制器在没有任何位置反馈信号的情况下工作，内部计算的操作变量将不会与信号控制元素位置完全相符。如果操作变量(ER \* GAIN)为负值，将进行调整。随后该控制器会设置输出 QLMNDN (操作值信号下限)，直到设置了 LMNR\_LS (位置反馈信号下限)为止。

该控制器还可作为控制器层叠中的辅助执行机构使用。设定值输入 SP\_INT 用于分配控制元素位置。在这种情况下，必须将实际值输入和参数 TI (集成时间)设置为零。一个应用实例是采用热量输出控制方法对温度进行调节，该方法通过阀进行脉冲暂停控制和冷却能力控制。在这种情况下，要完全关闭阀，应将操作变量(ER \* GAIN)设置为负值。

---

### 注意

只有在以固定时间间隔调用块时，在控制块中计算的才是正确的。为此，应该在周期性中断 OB (OB30 至 OB38)中调用控制块。在 CYCLE 参数中输入采样时间。

---

## 描述

除了过程值分支中的功能，SFB 还通过数字操作值输出和手动影响操作值选项实现了完整的 PI 控制器。步控制器在没有位置反馈信号的情况下工作。

下文提供了对部分功能的说明：

### 设定值分支

以浮点格式在 SP\_INT 输入键入设定值。

### 过程变量分支

可以外设(I/O)或以浮点格式输入过程变量。CRP\_IN 功能根据以下公式将 PV\_PER 外设值转换为介于 -100 和 +100 %间的浮点格式值：

$$\text{Output of CPR\_IN} = \text{PV\_PER} \quad * \quad \frac{100}{27648}$$

PV\_NORM 功能根据以下公式统一 CRP\_IN 输出的格式：

PV\_NORM 的输出 = (CPR\_IN 的输出) \* PV\_FAC + PV\_OFF

PV\_FAC 的缺省值为 1，PV\_OFF 的缺省值为 0。

### 出错信号

设定值和过程变量间的差异就是出错信号。为消除由于操作变量量化(例如，由于制动器阀操作值的精度有限)导致的小幅恒定振荡，将死区应用于出错信号(DEADBAND)。如果 DEADB\_W = 0，将关闭死区。

### PI 步算法

SFB/FB 在没有位置反馈信号的情况下工作。PI 算法的 I 操作和假定的位置反馈信号在一个积分器(INT)中计算，并作为反馈值与其余 P 操作进行比较。将差异应用于三步元素(THREE\_ST)和创建执行机构脉冲的脉冲发生器(PULSEOUT)。调整三步元素的阈值可以降低控制器的切换频率。

### 前馈控制

可以在 DISV 输入前馈干扰变量。

## 初始化

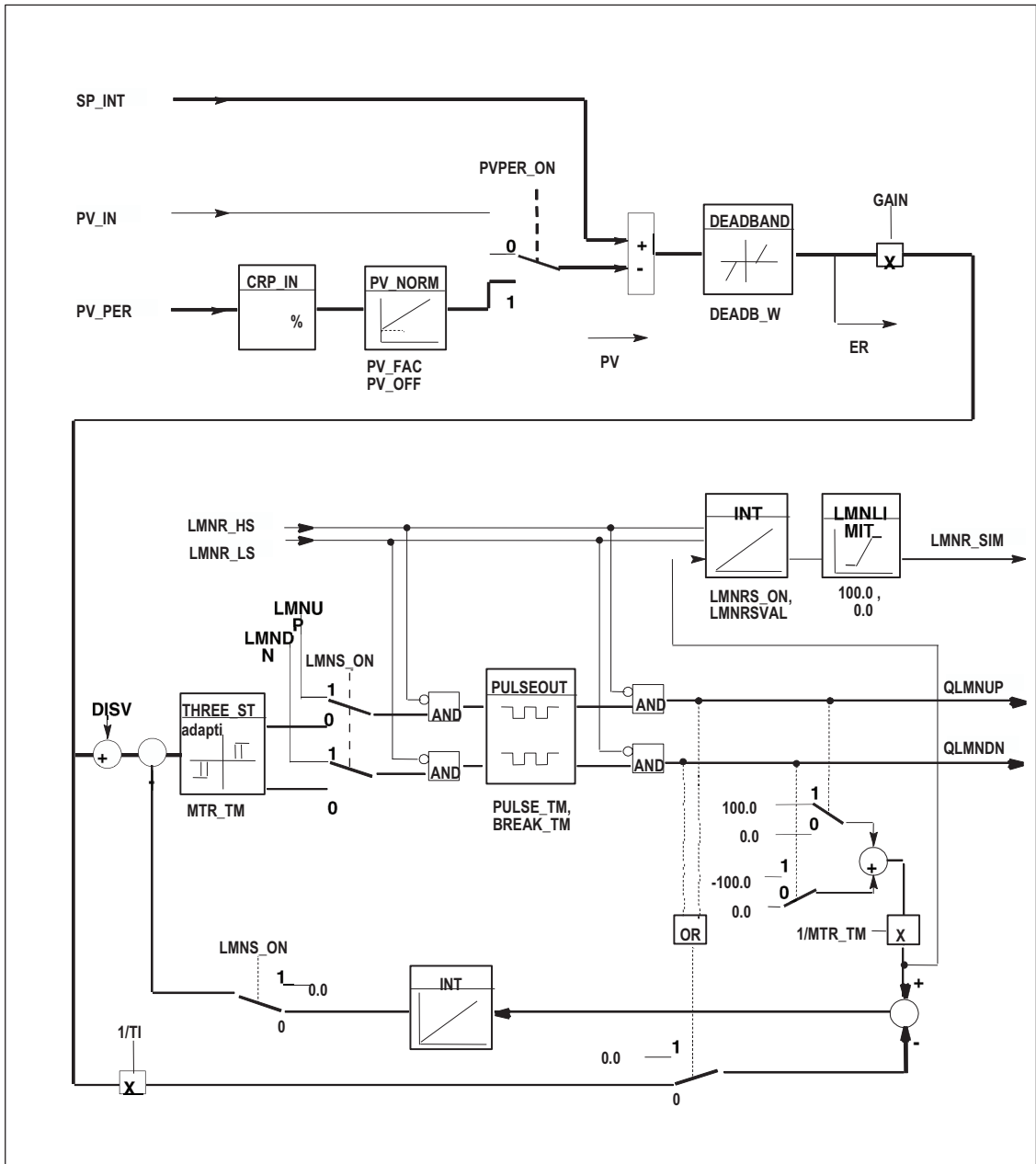
SFB/FB42 "CONT\_S"有一个在输入参数 COM\_RST = TRUE 时自动运行的初始化程序。

将所有其它输出设置为它们各自的缺省值。

## 错误信息

不使用出错输出参数 RET\_VAL。

方框图



## 输入参数

下表包含对 SFB 42/FB 42 "CONT\_S"输入参数的说明。

参数	数据类型	值	默认	描述
COM_RST	BOOL		FALSE	COMPLETE RESTART 该块有一个在设置输入 COM_RST 时自动执行的初始化程序。
LMNR_HS	BOOL		FALSE	HIGH LIMIT OF POSITION FEEDBACK SIGNAL 将"执行机构位于上限止点"信号连接到"位置反馈信号的上限"输入。LMNR_HS=TRUE 表示执行机构位于上限止点。
LMNR_LS	BOOL		FALSE	LOW LIMIT OF POSITION FEEDBACK SIGNAL 将"执行机构位于下限止点"信号连接到"位置反馈信号的下限"输入。LMNR_LS=TRUE 表示执行机构位于下限置点。
LMNS_ON	BOOL		FALSE	MANUAL ACTUATING SIGNALS ON 激励信号处理在"启用手动激励信号"输入中切换为手动。
LMNUP	BOOL		FALSE	ACTUATING SIGNALS UP 使用手动激励值信号时，在输入"激励信号增加"设置输出信号 QLMNUP。
LMNDN	BOOL		FALSE	ACTUATING SIGNALS DOWN 使用手动激励值信号时，在输入"激励信号降低"设置输出信号 QLMNDN。
PVPER_ON	BOOL		FALSE	PROCESS VARIABLE PERIPHERAL ON 如果从 I/O 读入过程变量，必须将输入 PV_PER 连接到 I/O，且必须设置输入"启用过程变量外设"。
CYCLE	TIME	>= 1 毫秒	T#1s	SAMPLING TIME 块调用之间的时间必须为常数。"采样时间"输入指定块调用之间的时间。
SP_INT	REAL	-100.0 ... +100.0 (%) 或物理值 1)	0.0	INTERNAL SETPOINT "内部设定值"输入用于指定设定值。
PV_IN	REAL	-100.0 ... +100.0 (%) 或物理值 1)	0.0	PROCESS VARIABLE IN 可以在"过程变量输入"输入设置初始化值，也可以连接浮点格式的外部过程变量。
PV_PER	WORD		W#16#0000	PROCESS VARIABLE PERIPHERAL 将 I/O 格式的过程变量连接到"过程变量外设"输入处的控制器。
GAIN	REAL		2.0	PROPORTIONAL GAIN "比例增益"输入设置控制器增益。
TI	TIME	>= CYCLE	T#20s	RESET TIME "复位时间"输入决定积分器的时间响应。
DEADB_W	REAL	0.0... 100.0 (%) 或物理值 1)	1.0	DEAD BAND WIDTH 将死区应用于出错。"死区带宽"输入决定死区的大小。
PV_FAC	REAL		1.0	PROCESS VARIABLE FACTOR "过程变量因子"输入与过程变量相乘。该输入用于调整过程变量范围。

参数	数据类型	值	默认	描述
PV_OFF	REAL		0.0	PROCESS VARIABLE OFFSET 将"过程变量偏移量"输入与过程变量相加。该输入用于调整过程变量范围。
PULSE_TM	TIME	>= CYCLE	T#3s	MINIMUM PULSE TIME 使用参数"最小脉冲时间"可以分配最小脉冲宽度。
BREAK_TM	TIME	>= CYCLE	T#3s	MINIMUM BREAK TIME 使用参数"最小中断时间"可以分配最小中断宽度。
MTR_TM	TIME	>= CYCLE	T#30s	MOTOR ACTUATING TIME 在"电机动作时间"参数中输入执行机构在挡块间移动所需的时间。
DISV	REAL	-100.0... 100.0 (%) 或物理值 2)	0.0	DISTURBANCE VARIABLE 为进行前馈控制，将干扰变量连接到输入"干扰变量"。

- 1) 同一单元中设定值和过程变量分支中的参数
- 2) 同一单元中操作值分支中的参数

**输出参数**

下表包含对 SFB 42/FB 42 "CONT\_S"输出参数的说明。

参数	数据类型	值	默认	描述
QLMNUP	BOOL		FALSE	ACTUATING SIGNAL UP 如果设置输出"激励信号增加", 将打开激励阀。
QLMNDN	BOOL		FALSE	ACTUATING SIGNAL DOWN 如果设置输出"激励信号降低", 将打开激励阀。
PV	REAL		0.0	PROCESS VARIABLE 有效过程变量为"过程变量"输出处的输出。
ER	REAL		0.0	ERROR SIGNAL 有效出错为"出错信号"输出处的输出。

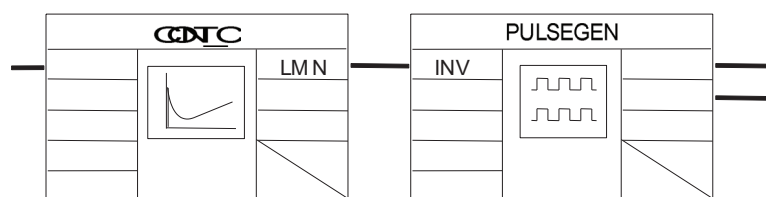
## 28.3 使用 SFB 43/FB 43 "PULSEGEN"生成脉冲

### 引言

SFB 43 "PULSEGEN" (脉冲发生器)用于构建具有比例执行机构脉冲输出的 PID 控制器。文档 > 英文 > STEP7 PID 控制中提供了电子手册。

### 应用

使用 SFB/FB "PULSEGEN"可以组态具有脉宽调制功能的 PID 两/三步控制器。该功能通常与连续控制器"CONT\_C"结合使用。



### 注意

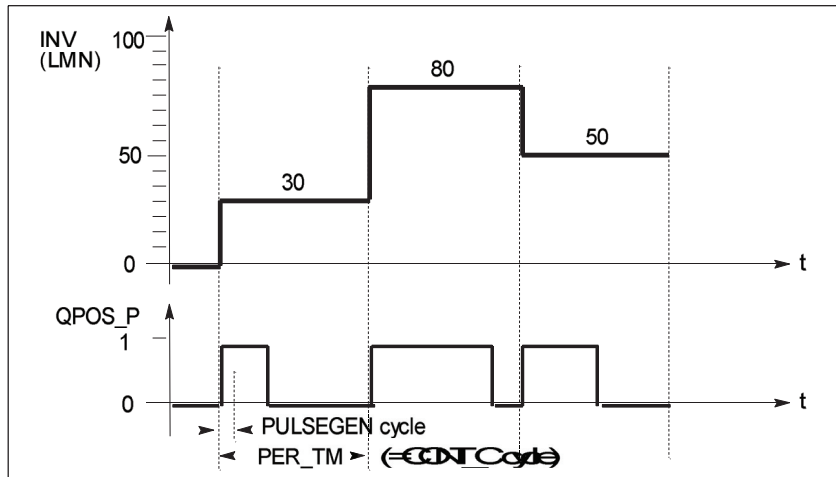
只有在以固定时间间隔调用块时，在控制块中计算的才是正确的。为此，应该在周期性中断 OB (OB30 至 OB38)中调用控制块。在 CYCLE 参数中输入采样时间。



## 描述

PULSEGEN 功能通过脉宽调制将输入变量 INV (= PID 控制器的操作值)转换为具有恒定周期的脉冲列，从而与输入变量的更新周期及必须在 PER\_TM 中分配的周期相应。

每周期脉冲宽度与输入变量成比例。分配给 PER\_TM 的周期与 SFB/FB "PULSEGEN"的处理周期不完全相同。PER\_TM 周期由若干个 SFB/FB "PULSEGEN"处理周期组成，因此可以将每个 PER\_TM 周期 SFB/FB "PULSEGEN"调用的数量作为脉宽调制精度的衡量标准。

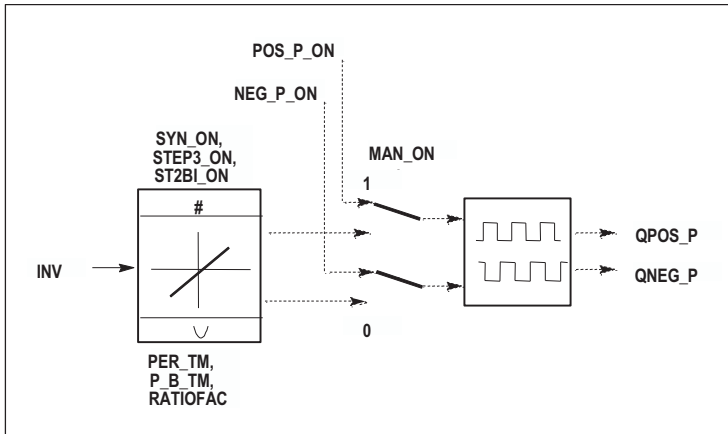


## 脉宽调制

输入变量为 30%、每个 PER\_TM 有 10 个 SFB/FB "PULSEGEN"调用的含义如下：

- 前三次 SFB/FB "PULSEGEN"调用(10 次调用的 30%)在 QPOS 输出为"一"
- 后七次 SFB/FB "PULSEGEN"调用(10 次调用的 70%)在 QPOS 输出为"零"

方框图



操作值的精度

"采样比率"为 1:10 (CONT\_C 调用与 PULSEGEN 调用之比)时,此实例中的操作值精度将限制在 10%,换言之,设置的输入值 INV 只能在 QPOS 输出通过以 10%为步长的脉冲宽度进行模拟。

精度随每个 CONT\_C 调用的 SFB/FB "PULSEGEN"调用数的增加而增加。

例如,如果调用 PULSEGEN 的频率是调用 CONT\_C 频率的 100 倍,则获得的操作值范围的精度为 1%。

**注意**

调用频率必须由用户进行设置。

## 自动同步

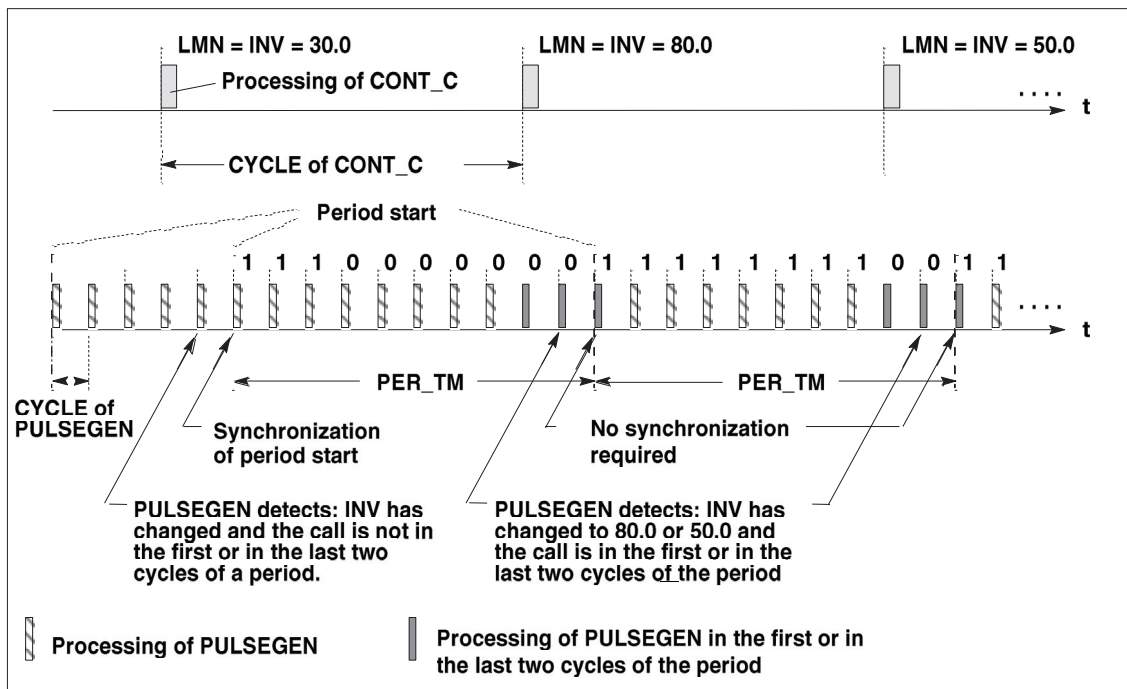
可以将脉冲输出与更新输入变量 INV (例如, CONT\_C)的块同步。这样可以确保输入变量更新的输出速度尽可能达到脉冲速度。

脉冲发生器以与 PER\_TM 周期相应的间隔判断输入值 INV, 并将该值转换为相应长度的脉冲信号。

不过, 由于 INV 通常是在较慢的周期性中断级别计算, 脉冲发生器应在更新 INV 后尽快开始将离散值转换为脉冲信号。

为此, 该块可以执行下列步骤来同步周期的开始:

如果 INV 发生变化且块调用不是发生在时期的前两个或后两个调用周期, 则执行同步。将重新计算脉冲宽度, 并在下一周期与新时期一起输出。



可以在"SYN\_ON"输入禁用自动同步  
(= FALSE)。

### 注意

随着新的周期的开始, 将在同步后于脉冲信号中近似精确地模拟 INV 的原值(即 LMN 值)。

## 模式

根据分配给脉冲发生器的参数，可以组态具有三步输出或双极或单极两步输出的 PID 控制器。下表列出了可能模式的开关组合设置。

模式	切换		
	MAN_ON	STEP3_ON	ST2BI_ON
三步控制	FALSE	TRUE	Any
具有双极控制范围的两步控制(-100%至 +100%)	FALSE	FALSE	TRUE
具有单极控制范围的两步控制(0%至 100%)	FALSE	FALSE	FALSE
手动模式	TRUE	Any	Any

## 三步控制

在"三步控制"模式下，激励信号可以具有三种状态。将二进制输出信号 QPOS\_P 和 QNEG\_P 的值分配给执行机构的状态。

下表提供了一个温度控制实例：

输出信号	执行机构		
	烘炉	关闭	冷却
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

基于输入变量，采用某一特性曲线计算脉冲宽度。特性曲线的形状由最小脉冲或最小中断时间及比率因子定义。

比率因子的正常值为 1。

曲线中的"转折"由最小脉冲或最小中断时间引起。

**最小脉冲或最小中断时间**

正确指定最小脉冲或最小中断时间 P\_B\_TM 可以防止频繁开/关，频繁开/关会缩短开关元件和执行机构的使用寿命。

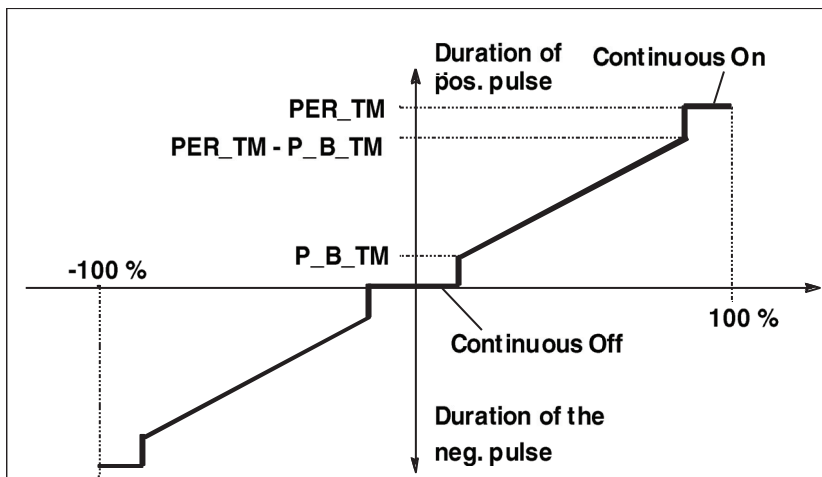
**注意**

输入变量 LMN 处那些小绝对值会受到抑制，否则它们会生成较 P\_B\_TM 更短的脉冲宽度。将生成比 (PER\_TM - P\_B\_TM) 长的脉冲宽度的大输入值设置为 100% 或 -100%。

通过将输入变量(%)与周期时间相乘获得正脉冲或负脉冲的宽度。

$$I\text{Period time} = \frac{\text{INV}}{100} * \text{PER\_TM}$$

下图显示的是三步控制器的对称曲线(比率因子 = 1)。



### 非对称三步控制

使用比率因子 RATIOFAC 可以更改正脉冲与负脉冲宽度的比率。例如，在热处理中，这使得加热和冷却可以有不同的系统时间常数。

比率因子还影响最小脉冲或最小中断时间。比率因子 < 1 意味着负脉冲的阈值与比率因子相乘。

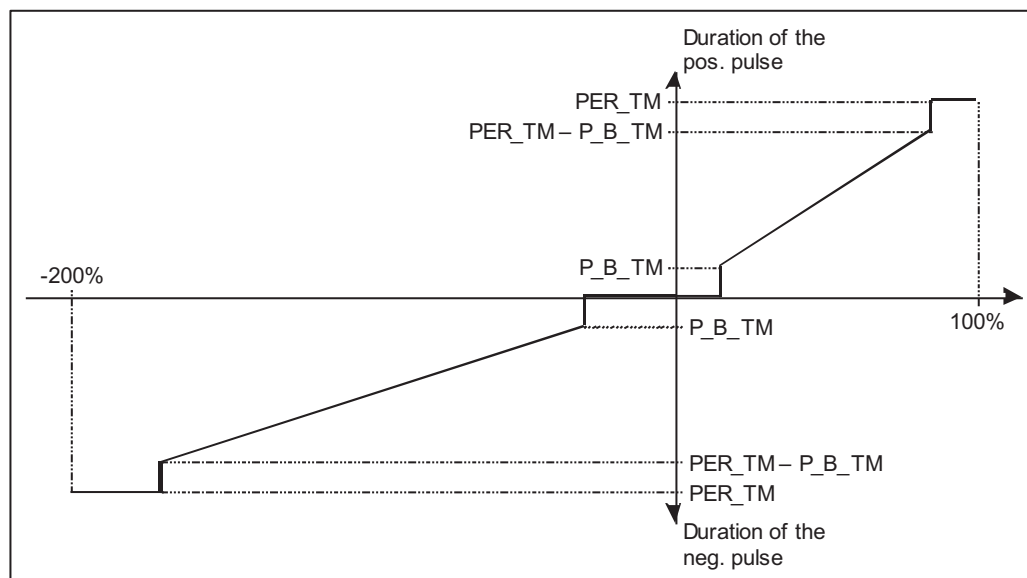
#### 比率因子 < 1

比率因子会缩短将输入变量与周期时间相乘所得到的负脉冲输出脉冲宽度。

$$\text{Dur. of positive pulse} = \frac{\text{INV}}{100} * \text{PER\_TM}$$

$$\text{Dur. of negative pulse} = \frac{\text{INV}}{100} * \text{PER\_TM} * \text{RATIOFAC}$$

下图显示三步控制器的非对称曲线(比率因子 = 0.5) :



**比率因子 > 1**

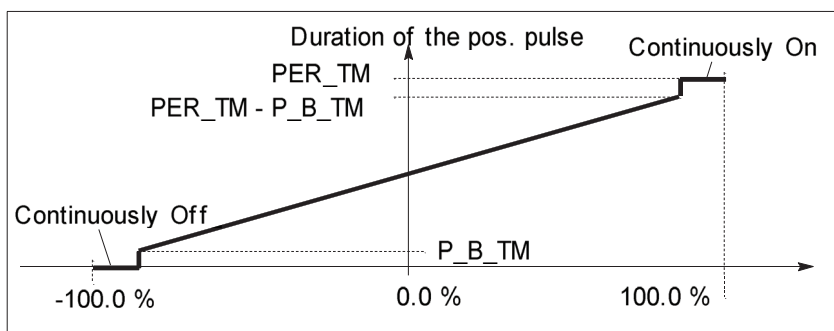
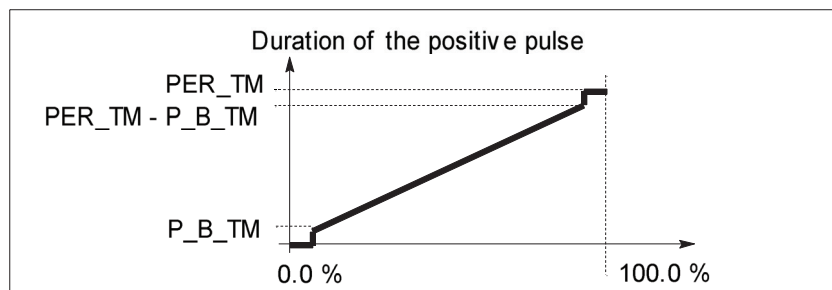
比率因子会缩短将输入变量与周期时间相乘所得到的正脉冲输出脉冲宽度。

$$\text{Dur. of negative pulse} = \frac{\text{INV}}{100} * \text{PER\_TM}$$

$$\text{Dur. of positive pulse} = \frac{\text{INV}}{100} * \frac{\text{PER\_TM}}{\text{RATIOFAC}}$$

**两步控制**

在两步控制中，只会将 PULSEGEN 的正脉冲输出 QPOS\_P 连接到开/关执行机构。视所使用的操作值范围，两步控制器将具有双极或单极操作值范围。

**具有双极操作变量范围的两步控制  
(-100%至 100%)****具有单极操作变量范围的两步控制  
(0%至 100%)**

如果控制回路中的两步控制器的连接需要使用逻辑取反的二进制信号来激励脉冲，可在 QNEG\_P 获得取反输出信号。

脉冲	执行机构	
	接通	关闭
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

### 两步/三步控制的手动模式

在手动模式(MAN\_ON = TRUE)下，无论 INV 为何值，均可使用信号 POS\_P\_ON 和 NEG\_P\_ON 设置三步或两步控制器的二进制输出。

控制	POS_P_ON	NEG_P_ON	QPOS_P	QNEG_P
三步控制	FALSE	FALSE	FALSE	FALSE
	TRUE	FALSE	TRUE	FALSE
	FALSE	TRUE	FALSE	TRUE
	TRUE	TRUE	FALSE	FALSE
两步控制	FALSE	Any	FALSE	TRUE
	TRUE	Any	TRUE	FALSE

### 初始化

SFB/FB "PULSGEN"有一个在输入参数 COM\_RST = TRUE 时自动运行的初始化程序。

所有信号输出都设置为 0。

### 错误信息

不使用出错输出参数 RET\_VAL。



## 输入参数

参数	数据类型	取值范围：	默认	描述
INV	REAL	-100.0...100.0 (%)	0.0	INPUT VARIABLE 将模拟操作值连接到输入参数"输入变量"。
PER_TM	TIME	$\geq 20 \cdot \text{CYCLE}$	T#1s	PERIOD TIME 使用"周期"输入参数输入脉宽调制的恒定周期。它对应于控制器的采样时间。脉冲发生器采样时间与控制器采样时间的比率决定脉宽调制的精度。
P_B_TM	TIME	$\geq \text{CYCLE}$	T#0ms	MINIMUM PULSE/BREAK TIME 可以在输入参数"最小脉冲或最小中断时间"分配最小脉冲或最小中断时间。
RATIOFAC	REAL	0.1 ...10.0	1.0	RATIO FACTOR 可以使用输入参数"比率因子"来改变负脉冲宽度与正脉冲宽度的比率。在热处理中，这使(例如)为加热和冷却补偿不同的时间常数(例如，在使用电加热和水冷却的工艺中)成为可能。
STEP3_ON	BOOL		TRUE	THREE STEP CONTROL ON "启用三步控制"输入参数可以激活此模式。在三步控制中，两个输出信号都处于激活状态。
ST2BI_ON	BOOL		FALSE	TWO STEP CONTROL FOR BIPOLAR MANIPULATED VALUE RANGE ON 使用输入参数"启用双极操作值范围的两步控制"可以在"双极操作值的两步控制"模式和"单极操作值范围的两步控制"模式间进行选择。必须设置参数 STEP3_ON = FALSE。
MAN_ON	BOOL		FALSE	MANUAL MODE ON 通过设置输入参数"启用手动模式"可以手动设置输出信号。
POS_P_ON	BOOL		FALSE	POSITIVE PULSE ON 在采用三步控制的手动模式下，可以在输入参数"启用正脉冲"设置输出信号 QPOS_P。在采用两步控制的手动模式下，QNEG_P 始终设置为与 QPOS_P 反向。
NEG_P_ON	BOOL		FALSE	NEGATIVE PULSE ON 在采用三步控制的手动模式下，可以在输入参数"启用负脉冲"设置输出信号 QNEG_P。在采用两步控制的手动模式下，QNEG_P 始终设置为与 QPOS_P 反向。
SYN_ON	BOOL		TRUE	SYNCHRONIZATION ON 通过设置输入参数"启用同步"，可以自动与更新输入变量 INV 的块同步。这样可以确保变化的输入变量的输出速度尽可能达到脉冲速度。
COM_RST	BOOL		FALSE	COMPLETE RESTART 该块有一个在设置了 COM_RST 输入时自动执行的初始化程序
CYCLE	TIME	$\geq 1 \text{ 毫秒}$	T#10ms	SAMPLING TIME 块调用间的时间必须为常数。"采样时间"输入指定块调用之间的时间。

---

**注意**

输入参数的值在块中不受限制。不进行参数检查。

---

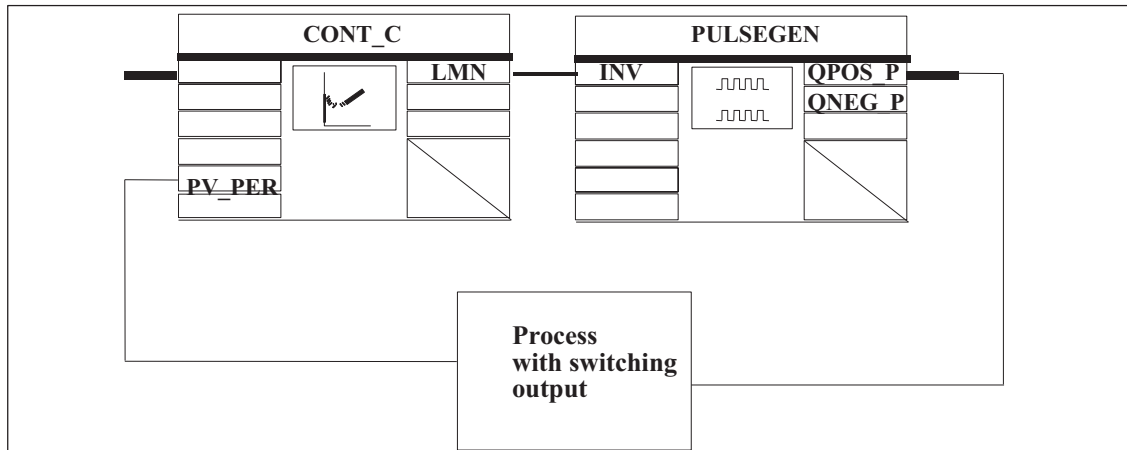
## 输出参数

参数	数据类型	值	默认	描述
QPOS_P	BOOL		FALSE	OUTPUT POSITIVE PULSE 将在要输出脉冲时设置输出参数"输出正脉冲"。在三步控制中，此项始终为正脉冲。在两步控制中，QNEG_P 始终设置为与 QPOS_P 反向。
QNEG_P	BOOL		FALSE	OUTPUT NEGATIVE PULSE 将在要输出脉冲时设置输出参数"输出负脉冲"。在三步控制中，此项始终是负脉冲。在两步控制中，QNEG_P 始终设置为与 QPOS_P 反向。

## 28.4 PULSEGEN 块的实例

### 控制回路

使用连续控制器 CONT\_C 和脉冲发生器 PULSEGEN , 用户可以实现具有比例执行机构切换输出的固定设定值控制器。下图显示控制回路的信号流。

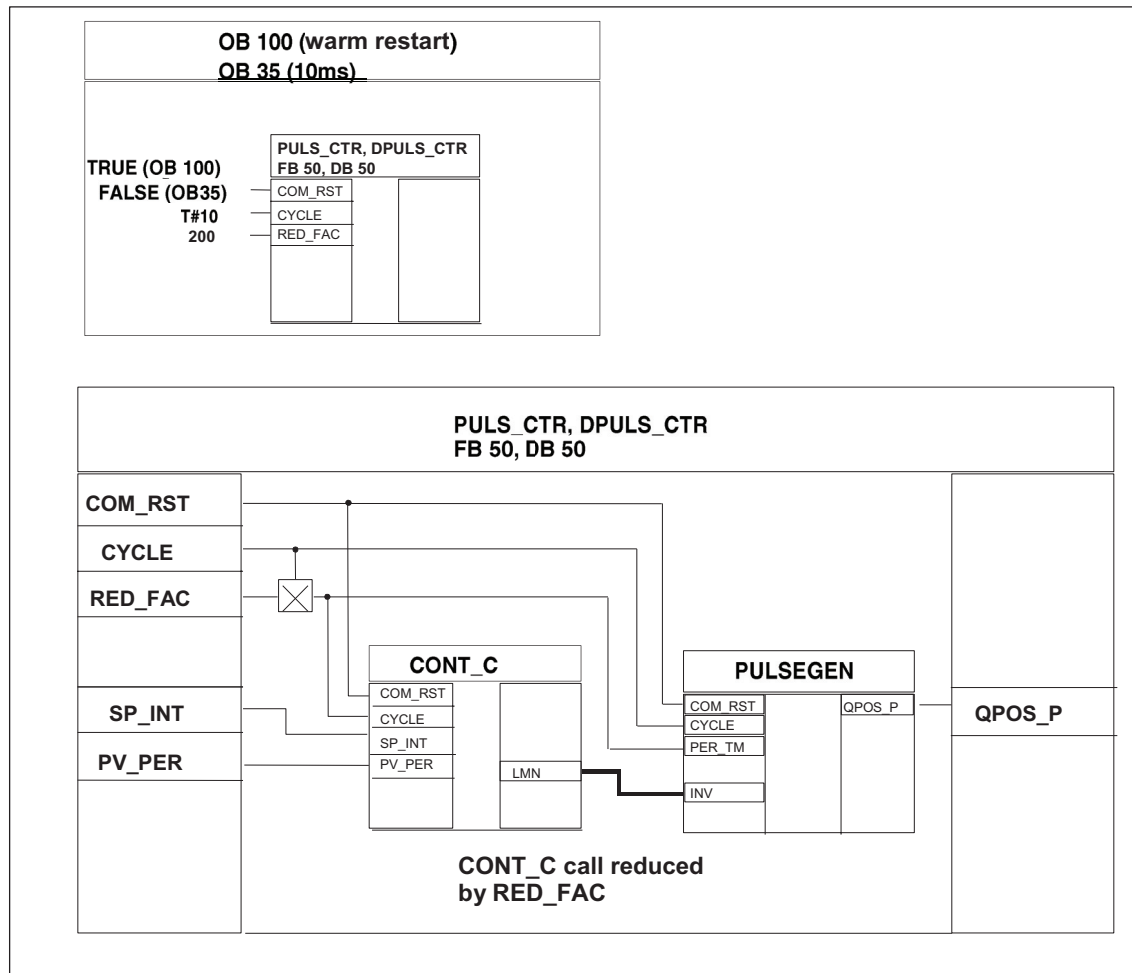


连续控制器 CONT\_C 形成操作值 LMN , 该操作值由脉冲发生器 PULSEGEN 转换为脉冲/中断信号 QPOS\_P 或 QNEG\_P。

### 调用块并连接块

具有比例执行机构 PULS\_CTRL 切换输出的固定设定值控制器由块 CONT\_C 和 PULSEGEN 组成。按以下方式实现块调用：每 2 秒(=CYCLE\*RED\_FAC)调用一次 CONT\_C，每 10 毫秒(=CYCLE)调用一次 PULSEGEN。下图显示了互连情况。

在暖启动过程中，将在 OB100 中调用块 PULS\_CTRL 并将输入 COM\_RST 设置为 TRUE。



FB PULS\_CTR 的 STL 程序

地址	声明	名称	类型	注释
0.0	in	SP_INT	REAL	设定值
4.0	in	PV_PER	WORD	过程变量外设
6.0	in	RED_FAC	INT	调用换算系数
8.0	in	COM_RST	BOOL	完全重启
10.0	in	CYCLE	TIME	采样时间
14.0	out	QPOS_P	BOOL	激励信号
16.0	stat	DI_CONT_C	FB-CONT_C	计数器
142.0	stat	DI_PULSEGEN	FB-PULSEGEN	计数器
176.0	stat	SCount	INT	计数器
0.0	temp	TCycCtr	TIME	控制器采样时间

STL	描述
<pre> A      #COM_RST JCN    M001 L      0 T      #sCount                     </pre>	//初始化程序
<pre> M001: L      #CYCLE L      #RED_FAC *D T      #tCycCtr                     </pre>	//计算控制器采样时间
<pre> L      #sCount L      1 -I T      #sCount L      0 &lt;=I                     </pre>	//递减计数器并与零比较
<pre> JCN    M002 CALL   #DI_CONT_C COM_RST :  =#COM_RST CYCLE :   =#tCycCtr SP_INT :  =#SP_INT PV_PER :  =#PV_PER L      #RED_FAC T      #sCount M002: L      #DI_CONT_C.LMN T      #DI_PULSEGEN.INV CALL   #DI_PULSEGEN PER_TM :  =#tCycCtr COM_RST :  =#COM_RST CYCLE :   =#CYCLE QPOS_P :  =#QPOS_P BE                     </pre>	//条件块调用和设置计数器

## 29 用于紧凑型 CPU 的 SFB

### 29.1 通过 SFB 44 "Analog"使用模拟量输出进行定位

#### 描述

要通过用户程序控制定位功能，请使用 **SFB ANALOG (SFB 44)**。

固定分配的模拟输出通过±10 V 的电压(**电压信号**)或±20mA 的电流(**电流信号**)控制功率级。

- 加速阶段(**RAM\_UP** 过后，驱动器以速度(**VSetpoint**)接近目标。
- 到达由 CPU 计算出的**制动点**时，将开始减速(**RAMP\_DN**)，一直到转换点。
- 一旦到达**转换点**，将继续以慢速(**VCreep**)运行。
- 驱动器在到达**截止点**时关闭。
- 每个 Step Approach 的转换点和截止点由已指定的**转换差值**和**截止差值**的参数值确定。可以用不同方式确定前进运动(正方向)和反向运动(负方向)的转换差值和截止差值。
- 到达截止点时，运行即告结束(**WORKING = FALSE**)。随后即可开始新的运行。
- 在当前位置值达到**目标范围**时，即达到指定的目标(**POS\_RCD = TRUE**)。如果当前位置值逐渐偏移，而又未启动新的运行，则"到达位置"信号将不会再次复位。

当转换差值小于截止差值时，驱动器将从制动点减速到速度设定值 0。

#### 基本参数：

在此描述适用于所有操作模式的 SFB 参数。各操作模式的专用参数将随该操作模式一并描述。

参数：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
LADDR	INPUT	WORD	0	CPU 专用	W#16#0310	子模块的 I/O 地址，由用户在"HW 配置"中指定。 如果 E 和 A 地址不相等，则必须指定二者中较低的一个。
CHANNEL	INPUT	INT	2	0	0	通道号
STOP	INPUT	BOOL	4.4	TRUE/FALSE	FALSE	停止运行 使得 STOP = TRUE，可以提前停止/中断运行。
ERR_A	INPUT	BOOL	4.5	TRUE/FALSE	FALSE	收集对外部错误的确认 经 ERR_A = TRUE 确认后，外部错误即被清除。
SPEED	INPUT	DINT	12	慢速最高为 1,000,000 脉冲/s 不高于在参数 中声明的最大 速度	1000	轴加速到速度为"VSetpoint"。 运行中无法改变速度。
WORKING	OUTPUT	BOOL	16.0	TRUE/FALSE	FALSE	正在运行
ACT_POS	OUTPUT	DINT	18	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	实际位置值
MODE_OUT	OUTPUT	INT	22	0, 1, 3, 4, 5	0	激活的/已组态的操作模式
ERR	OUTPUT	WORD	24	每位 "0"或"1"：	0	外部错误： 位 2：零点监视 位 11：行程监视(始终为 1) 位 12：操作范围监视 位 13：当前值监视 位 14：目标归位监视 位 15：目标范围监视 剩余的其它位保留备用
ST_ENBLD	OUTPUT	BOOL	26.0	TRUE/FALSE	TRUE	如果满足下列所有条件， CPU 会设置启动使能： <ul style="list-style-type: none"> <li>没有未决 STOP (STOP = FALSE)</li> <li>没有未决外部错误 (ERR = 0)</li> <li>已设置驱动器使能 (DRV_EN = TRUE)</li> <li>没有活动的定位运行 (WORKING = FALSE)</li> </ul>
ERROR	OUTPUT	BOOL	26.1	TRUE/FALSE	FALSE	开始/恢复运行时出错
STATUS	OUTPUT	WORD	28.0	W#16#0000 至 W#16#FFFF	W#16#0000	错误编号



未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
ACCEL	STATIC	DINT	30	1 到 100,000 脉冲/s <sup>2</sup>	100	加速度 运行中无法更改。
DECEL	STATIC	DINT	34	1 到 100,000 脉冲/s <sup>2</sup>	100	减速度 运行中无法更改。
CHGDIFF_P	STATIC	DINT	38	0 到 +10 <sup>8</sup> 个 脉冲	1000	正转换差值： 正"转换差值"定义驱动器 以慢速继续向前运行的转 换点。
CUTOFF- DIFF_P	STATIC	DINT	42	0 到 +10 <sup>8</sup> 个 脉冲	100	正截止差值： "正截止差值"定义关闭驱 动器慢速向前运行时的截 止点。
CHGDIFF_M	STATIC	DINT	46	0 到 +10 <sup>8</sup> 个 脉冲	1000	负转换差值： "负转换差值"定义驱动器 以慢速反向继续运行的转 换点。
CUTOFF- DIFF_P	STATIC	DINT	50	0 到 +10 <sup>8</sup> 个 脉冲	100	负截止差值： "负截止差值"定义关闭驱 动器慢速反向运行时的截 止点。
PARA	STATIC	BOOL	54.0	TRUE/FALSE	FALSE	参数已分配给轴
DIR	STATIC	BOOL	54.1	TRUE/FALSE	FALSE	实际的/上一方向 FALSE = 前进(正方向) TRUE = 反向(负方向)
CUTOFF	STATIC	BOOL	54.2	TRUE/FALSE	FALSE	驱动器在截止范围内(从截 止点到下一运行的起点)
CHGOVER	STATIC	BOOL	54.3	TRUE/FALSE	FALSE	驱动器在转换范围内(到达 慢速和下一运行起点之间)
RAMP_DN	STATIC	BOOL	54.4	TRUE/FALSE	FALSE	驱动器减速(从制动点到转 换点)
RAMP_UP	STATIC	BOOL	54.5	TRUE/FALSE	FALSE	驱动器加速(从起点到达到 速度 SPEED (VSetpoint))
DIST_TO_ GO	STATIC	DINT	56	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	实际行进距离
LAST_TRG	STATIC	DINT	60	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	上一个/当前目标 <ul style="list-style-type: none"> <li>绝对步骤法： 运行启动时 LST_TRG = 当前绝对目标 (TARGET)。</li> <li>相对步骤法： 运行启动时 LST_TRG = LAST_TRG 为前一 运行指定的 +/- 距离 (TARGET)。</li> </ul>

**"点动"模式的参数**

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	驱动器使能
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	正向点动(上升沿)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	负向点动(上升沿)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	操作模式, 1 = 点动
WORKING	OUTPUT	BOOL	16.0	TRUE/FALSE	FALSE	正在运行
ACT_POS	OUTPUT	DINT	18	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	实际位置值
MODE_OUT	OUTPUT	INT	22	0, 1, 3, 4, 5	0	激活的/已组态的操作模式

**"参考运行"模式的参数**

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	驱动器使能
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	正向参考运行(上升沿)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	负向参考运行(上升沿)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	操作模式, 3 ="参考运行"
WORKING	OUTPUT	BOOL	16.0	TRUE/FALSE	FALSE	正在运行
SYNC	OUTPUT	BOOL	16.3	TRUE/FALSE	FALSE	SYNC = TRUE : 轴已同步
ACT_POS	OUTPUT	DINT	18	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	实际位置值
MODE_OUT	OUTPUT	INT	22	0, 1, 3, 4, 5	0	激活的/已组态的操作模式

## "相对法"模式的参数

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	驱动器使能
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	正向运行(上升沿)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	负向运行(上升沿)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	操作模式, 4 = 相对步骤法
TARGET	INPUT	DINT	8	0 到 $10^9$ 个脉冲	1000	以脉冲计算的距离(只允许正值)
WORKING	OUTPUT	BOOL	16.0	TRUE/FALSE	FALSE	正在运行
POS_RCD	OUTPUT	BOOL	16.1	TRUE/FALSE	FALSE	到达位置
ACT_POS	OUTPUT	DINT	18	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0	实际位置值
MODE_OUT	OUTPUT	INT	22	0, 1, 3, 4, 5	0	激活的/已组态的操作模式

## "绝对法"的参数

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	驱动器使能
START	INPUT	BOOL	4.1	TRUE/FALSE	FALSE	开始运行(上升沿)
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	正向运行(上升沿)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	负向运行(上升沿)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	操作模式, 5 = 绝对法
TARGET	INPUT	DINT	8	线性轴 -5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 旋转轴 0 到旋转轴 终点 -1	1000	以脉冲计算的目标
WORKING	OUTPUT	BOOL	16.0	TRUE/FALSE	FALSE	正在运行
POS_RCD	OUTPUT	BOOL	16.1	TRUE/FALSE	FALSE	到达位置
ACT_POS	OUTPUT	DINT	18	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	实际位置值
MODE_OUT	OUTPUT	INT	22	0, 1, 3, 4, 5	0	激活的/已组态的操作 模式

**"设置参考点"作业的参数**

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
SYNC	OUTPUT	BOOL	16.3	TRUE/FALSE	FALSE	轴已同步

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
JOB_REQ	STATIC	BOOL	76.0	TRUE/FALSE	FALSE	作业初始化(上升沿)
JOB_DONE	STATIC	BOOL	76.1	TRUE/FALSE	TRUE	可以启动新作业
JOB_ERR	STATIC	BOOL	76.2	TRUE/FALSE	FALSE	故障作业
JOB_ID	STATIC	INT	78	1, 2	0	作业，1 = "设置参考点"
JOB_STAT	STATIC	WORD	80	W#16#0000 至 W#16#FFFF	W#16# 0000	作业错误号
JOB_VAL	STATIC	DINT	82	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	参考点坐标的作业 参数

**"清除剩余距离"作业的参数**

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
JOB_REQ	STATIC	BOOL	76.0	TRUE/FALSE	FALSE	作业初始化(上升沿)
JOB_DONE	STATIC	BOOL	76.1	TRUE/FALSE	TRUE	可以启动新作业
JOB_ERR	STATIC	BOOL	76.2	TRUE/FALSE	FALSE	故障作业
JOB_ID	STATIC	INT	78	1, 2	0	作业，2 = "清除剩余 距离"
JOB_STAT	STATIC	WORD	80	W#16#0000 至 W#16#FFFF	W#16# 0000	作业错误号
JOB_VAL	STATIC	DINT	82	-	0	任意设置

**"长度测量"操作的参数**

此操作开始于数字输入的上升沿。没有特定的输入参数。

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
MSR_DONE	OUTPUT	BOOL	16.2	TRUE/FALSE	FALSE	长度测量完成

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
BEG_VAL	STATIC	DINT	64	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	开始长度测量的实际位置值
END_VAL	STATIC	DINT	68	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	完成长度测量的实际位置值
LEN_VAL	STATIC	DINT	72	0 到 10 <sup>9</sup> 个脉冲	0	测量的长度

## 错误信息

## 操作模式错误(ERROR = TRUE)

如果检测到错误，输出参数 ERROR 将置为 TRUE。STATUS 参数显示发生错误的原因。

事件等级 错误代码	解释
W#16#2002	SFB 错误，请使用 SFB 44
W#16#2004	通道号(CHANNEL)错误。将通道设置为"0"
W#16#3001	由于调用同一 SFB 时发生作业错误而拒绝运行作业。更正各自的 JOB 参数
W#16#3002	驱动器工作时不允许更改 MODE_IN。请等待当前定位运行结束。
W#16#3003	未知操作模式(MODE_IN)。允许模式为 1(点动)、3(参考运行)、4(相对法)和 5(绝对法)。
W#16#3004	同一时间只允许一个启动请求。有效的启动请求为 DIR_P、DIR_M 或 START。
W#16#3005	只有"绝对法"操作模式允许 START。使用 DIR_P 或 DIR_M 启动运行
W#16#3006	线性轴和"绝对法"操作模式不允许使用 DIR_P 或 DIR_M。使用 START 启动运行
W#16#3007	轴未同步。只有同步轴时才能使用"绝对法"。
W#16#3008	清除工作范围。只有在点动模式下才允许将运行返回工作位置。
W#16#3101	由于没有为轴指定参数，因此禁止启动。通过 HWConfig 为"定位"子模块指定参数
W#16#3102	由于未使能驱动器，因此禁止启动。在 SFB 上设置"驱动器使能"(DRV_EN=TRUE)
W#16#3103	由于设置了 STOP，因此禁止启动。在 SFB 中清除 STOP (STOP=FALSE)
W#16#3104	由于驱动器当前正在执行定位运行(WORKING=TRUE)，禁止启动。请等待当前定位运行结束。
W#16#3105	由于至少有一个未决错误尚未清除，因此禁止启动。首先，排除并清除所有外部错误，然后重新启动运行。
W#16#3202	SPEED 中的速度设定值错误。速度设定值超出最高为 1000000 脉冲/s 的慢速允许范围，尽管未超出参数指定的最大速度。
W#16#3203	ACCEL 的加速设定值超出 1 到 100,000 脉冲/s <sup>2</sup> 的范围。
W#16#3204	DECEL 的减速设定值超出 1 到 100,000 脉冲/s <sup>2</sup> 的范围。
W#16#3206	SPEED 的速度设定值必须高于/等于参数化的参考频率。
W#16#3301	转换/截止差值过高。将最大转换/截止差值设置为 10 <sup>8</sup>
W#16#3304	截止差值过低。截止差值大小必须至少为目标范围的一半。
W#16#3305	转换差值过低。转换差值大小至少必须为目标范围的一半。
W#16#3401	目标设置超出工作范围。对于线性轴和 Step Approach，目标设置必须在软件限位开关范围内(含此限值)。
W#16#3402	目标设置错误。对于旋转轴，目标设置必须大于 0 并小于旋转轴终点值。
W#16#3403	距离设置错误。"相对法"的行进距离设定值必须为正值。
W#16#3404	距离设置错误。所得结果(绝对目标坐标)必须大于 -5x10 <sup>8</sup> 。
W#16#3405	距离设置错误。所得结果(绝对目标坐标)必须小于 5x10 <sup>8</sup> 。
W#16#3406	距离设置错误。所得结果(绝对目标坐标)必须在工作范围(目标范围的 +/- 一半)内
W#16#3501	行进距离过长。目标坐标 + 当前剩余距离必须大于/等于 -5x10 <sup>8</sup>
W#16#3502	行进距离过长。目标坐标 + 当前剩余距离必须小于/等于 5x10 <sup>8</sup>
W#16#3503	行进距离过短。正向的行进距离必须大于指定的正向截止差值
W#16#3504	行进距离过短。负向的行进距离必须大于指定的负向截止差值

事件等级 错误代码	解释
W#16#3505	行进距离过短，或正方向的限位开关已经启动。正向最后可接近的目标(工作范围或行进距离限制)与当前位置过近
W#16#3506	行进距离过短，或负方向的限位开关已经启动。负向最后可接近的目标(工作范围或行进距离限制)与当前位置过近



**作业错误(JOB\_ERR = TRUE)**

如果检测到错误，输出参数 JOB\_ERROR 将置为 TRUE。参数 JOB\_ERROR 给出了错误原因。

事件等级 错误代码	解释
W#16#4001	未设置轴参数。通过 HWConfig 为"定位"子模块指定参数
W#16#4002	由于仍在定位，无法执行作业。等到 WORKING = FALSE，再重复执行作业。
W#16#4004	未知作业。检查作业标识号，然后重复执行作业。
W#16#4101	对于线性轴，参考点坐标不能超出工作范围的限制。
W#16#4102	对于线性轴，设置的参考点坐标 + 当前剩余距离必须大于/等于 $-5 \times 10^8$ 。
W#16#4103	对于线性轴，设置的参考点坐标 + 当前剩余距离必须小于/等于 $5 \times 10^8$ 。
W#16#4104	对于线性轴，设置的参考点坐标 + 距起始点的当前剩余差值必须大于/等于 $-5 \times 10^8$ 。
W#16#4105	对于线性轴，设置的参考点坐标 + 距起始点的当前剩余差值必须小于/等于 $-5 \times 10^8$ 。
W#16#4106	对于旋转轴，参考点坐标不能小于 0，也不能大于/等于旋转轴终点。

**外部错误(ERR)**

技术电路会监视运行、行进距离和连接的外围设备。首先必须已在"驱动器"、"轴"和"编码器"参数画面窗体中打开监视功能。

触发监视单元时将报告外部故障。外部错误的发生与已启动的操作无关。必须使用 ERR\_A = TRUE 清除外部错误。

SFB 参数 ERR (WORD)的置位对应外部错误。

监视	错误代码：	ERR-WORD 中的位
零脉冲(零标记)	W#16#0004	2
行进距离	W#16#0800	11
工作范围	W#16#1000	12
实际值	W#16#2000	13
目标位置	W#16#4000	14
目标范围	W#16#8000	15

**系统错误**

系统错误由 BIE = FALSE 指示。系统错误由读取/写入实例 DB 引起或由多次调用 SFB 引起。

## 29.2 通过 SFB 46 "DIGITAL"使用数字量输出进行定位

### 描述

通过用户程序使用 SFB DIGITAL (SFB 46)控制定位功能。

已为驱动器固定地分配了四个 24V 数字输出。它们用于控制功率级。数字输出根据控制模式组态控制方向和速度级(快速/慢速)。

通过两个相位偏移为 90 度的非对称 24V 增量传感器测量距离。

- 首先，以(V<sub>Rapid</sub>)速度接近目标。
- 到达**转换点**时速度切换为慢速(V<sub>Creep</sub>)。
- 驱动器在到达**关闭点**时关闭。
- 每个 Step Approach 的切换点和关闭点由为**转换差值**和**截止差值**声明的参数值确定。前进运动(正方向)和反向运动(负方向)的转换差值和截止差值可以用不同方式确定。
- 到达截止点时，运行即告结束(WORKING = FALSE)。随后即可开始新的运行。
- 在当前位置值达到**归位目标**时，即达到指定的目标(POS\_RCD = TRUE)。如果当前位置值逐渐偏移，而又未启动新的运行，则"到达位置"信号将不会再次复位。

### 基本参数：

在此将说明 SFB 参数。这些参数适用于所有操作模式。各操作模式的专用参数将随该操作模式一并描述。

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
LADDR	INPUT	WORD	0	CPU 专用	W#16#0310	子模块的 I/O 地址，由用户在"HW 配置"中指定。 如果 E 和 A 地址不相等，则必须指定二者中较低的一个。
CHANNEL	INPUT	INT	2	0	0	通道号：
STOP	INPUT	BOOL	4.4	TRUE/FALSE	FALSE	停止运行 使得 STOP = TRUE，可以提前停止/中断运行。
ERR_A	INPUT	BOOL	4.5	TRUE/FALSE	FALSE	收集对外部错误的确认 经 ERR_A = TRUE 确认后，外部错误即被清除。
SPEED	INPUT	BOOL	12.0	TRUE/FALSE	FALSE	用于"快速/慢速"模式的两个速度级 TRUE = 快速模式 FALSE = 慢速模式
WORKING	OUTPUT	BOOL	14.0	TRUE/FALSE	FALSE	正在运行
ACT_POS	OUTPUT	DINT	16	-5x10 <sup>8</sup> 到 5x10 <sup>8</sup> 个脉冲	0	实际位置值

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
MODE_OUT	OUTPUT	INT	20	0, 1, 3, 4, 5	0	激活的/已组态的操作模式
ERR	OUTPUT	WORD	22	每位 "0"或"1"：	0	外部错误： 第 2 位：零标记监视 第 11 位：行程监视 (始终为 1) 第 12 位：操作范围监视 第 12 位：实际值监视 第 12 位：目标位置监视 第 15 位：目标位置监视。 剩余的其它位保留备用
ST_ENBLD	OUTPUT	BOOL	24.0	TRUE/FALSE	TRUE	如果满足下列所有条件，则使能 CPU： <ul style="list-style-type: none"> <li>• 没有未决 STOP (STOP = FALSE)</li> <li>• 没有未决外部错误 (ERR = 0)</li> <li>• 已设置驱动器使能 (DRV_EN = TRUE)</li> <li>• 没有活动的定位运行 (WORKING = FALSE)</li> </ul>
ERROR	OUTPUT	BOOL	24.1	TRUE/FALSE	FALSE	开始/恢复运行时出错
STATUS	OUTPUT	WORD	26.0	W#16#0000 至 W#16#FFFF	W#16#0000	错误编号

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
CHGDIFF_P	STATIC	DINT	28	0 到 $+10^8$ 个脉冲	1000	正切换差值： "正切换差值"定义驱动器以慢速继续向前运行的点。
CUTOFF-DIFF_P	STATIC	DINT	32	0 到 $+10^8$ 个脉冲	100	正截止差值： "正截止差值"定义关闭驱动器慢速向前运行时的截止点。
CHGDIFF_M	STATIC	DINT	36	0 到 $+10^8$ 个脉冲	1000	负切换差值： "负切换差值"定义驱动器以慢速继续向前运行的点。
CUTOFF-DIFF_P	STATIC	DINT	40	0 到 $+10^8$ 个脉冲	100	负截止差值： "正截止差值"定义关闭驱动器慢速反向运行时的截止点。
PARA	STATIC	BOOL	44.0	TRUE/FALSE	FALSE	参数已分配给轴
DIR	STATIC	BOOL	44.1	TRUE/FALSE	FALSE	实际的/上一方向 FALSE = 前进(正方向) TRUE = 反向(负方向)
CUTOFF	STATIC	BOOL	44.2	TRUE/FALSE	FALSE	驱动器在截止范围内 (从截止位置到下一运行的起点)
CHGOVER	STATIC	BOOL	44.3	TRUE/FALSE	FALSE	驱动器在转换范围内 (从到达慢速的点到下一运行的起点)
DIST_TO_GO	STATIC	DINT	46	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0	实际剩余距离
LAST_TRG	STATIC	DINT	50	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0	上一个/当前目标 <ul style="list-style-type: none"> <li>• 绝对步骤法： 运行启动时 LAST_TRG = 当前绝对目标 (TARGET)。</li> <li>- 相对步骤法： 运行启动时 LAST_TRG = LAST_TRG 为 前一运行指定的 +/- 距离 (TARGET)。</li> </ul>

## "点动"模式的参数

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	驱动器使能
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	正向点动(上升沿)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	负向点动(上升沿)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	操作模式，1 = 点动
WORKING	OUTPUT	BOOL	14.0	TRUE/FALSE	FALSE	正在运行
ACT_POS	OUTPUT	DINT	16	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	实际位置值
MODE_OUT	OUTPUT	INT	20	0, 1, 3, 4, 5	0	激活的/已组态的操作模式

**"参考运行"模式的参数**

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	驱动器使能
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	正向参考运行 (上升沿)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	负向参考运行 (上升沿)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	操作模式， 3 = "参考运行"
WORKING	OUTPUT	BOOL	14.0	TRUE/FALSE	FALSE	正在运行
SYNC	OUTPUT	BOOL	14.3	TRUE/FALSE	FALSE	SYNC = TRUE： 轴已同步
ACT_POS	OUTPUT	DINT	16	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	实际位置值
MODE_OUT	OUTPUT	INT	20	0, 1, 3, 4, 5	0	激活的/已组态的 操作模式

**"相对法"模式的参数**

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	驱动器使能
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	正向运行(上升沿)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	负向运行(上升沿)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	操作模式，4 = 相对步 骤法
TARGET	INPUT	DINT	8	0 到 10 <sup>9</sup> 个 脉冲	1,000	以脉冲计算的距离 (只允许正值)
WORKING	OUTPUT	BOOL	14.0	TRUE/FALSE	FALSE	正在运行
POS_RCD	OUTPUT	BOOL	14.1	TRUE/FALSE	FALSE	到达位置
ACT_POS	OUTPUT	DINT	16	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	实际位置值
MODE_OUT	OUTPUT	INT	20	0, 1, 3, 4, 5	0	激活的/已组态的操作 模式

## "绝对法"的参数

参数	声明	数据类型	地址 (实例 DB)	取值范围 :	默认	描述
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	驱动器使能
START	INPUT	BOOL	4.1	TRUE/FALSE	FALSE	开始运行(上升沿)
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	正向运行(上升沿)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	负向运行(上升沿)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	操作模式, 5 = 绝对法
TARGET	INPUT	DINT	8	线性轴 -5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 旋转轴 : 0 到旋转轴 终点 -1	1,000	以脉冲计算的目标
WORKING	OUTPUT	BOOL	14.0	TRUE/FALSE	FALSE	正在运行
POS_RCD	OUTPUT	BOOL	14.1	TRUE/FALSE	FALSE	到达位置
ACT_POS	OUTPUT	DINT	16	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	实际位置值
MODE_OUT	OUTPUT	INT	20	0, 1, 3, 4, 5	0	激活的/已组态的操作 模式

**"设置参考点"作业的参数**

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
SYNC	OUTPUT	BOOL	14.3	TRUE/FALSE	FALSE	轴已同步

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
JOB_REQ	STATIC	BOOL	66.0	TRUE/FALSE	FALSE	作业初始化 (上升沿)
JOB_DONE	STATIC	BOOL	66.1	TRUE/FALSE	TRUE	可以启动新作业
JOB_ERR	STATIC	BOOL	66.2	TRUE/FALSE	FALSE	故障作业
JOB_ID	STATIC	INT	68	1, 2	0	作业， 1 ="设置参考点"
JOB_STAT	STATIC	WORD	70	W#16#0000 至 W#16#FFFF	W#16#0000	作业错误号
JOB_VAL	STATIC	DINT	72	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	参考点坐标的 作业参数

**"清除剩余距离"作业的参数**

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
JOB_REQ	STATIC	BOOL	66.0	TRUE/FALSE	FALSE	作业初始化(上升沿)
JOB_DONE	STATIC	BOOL	66.1	TRUE/FALSE	TRUE	可以启动新作业
JOB_ERR	STATIC	BOOL	66.2	TRUE/FALSE	FALSE	故障作业
JOB_ID	STATIC	IINT	68	1, 2	0	作业， 2 ="清除剩余距离"
JOB_STAT	STATIC	WORD	70	0 到 FFFF 十六进制	0	作业错误号
JOB_VAL	STATIC	DINT	72	-	0	无。



**"长度测量"功能的参数**

此操作开始于数字输入的上升沿。没有特定的输入参数。

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
MSR_DONE	OUTPUT	BOOL	14.2	TRUE/FALSE	FALSE	长度测量完成

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
BEG_VAL	STATIC	DINT	54	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	开始长度测量的实际位置值
END_VAL	STATIC	DINT	58	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	完成长度测量的实际位置值
LEN_VAL	STATIC	DINT	62	0 到 10 <sup>9</sup> 个脉冲	0	测量的长度

## 错误信息

## 操作模式错误(ERROR = TRUE)

如果检测到错误，输出参数 ERROR 将置为 TRUE。STATUS 参数显示发生错误的原因。

事件等级 错误代码	解释
W#16#2001	SFB 错误，请使用 SFB 46
W#16#2004	通道号(CHANNEL)错误。将通道设置为"0"
W#16#3001	由于调用同一 SFB 时发生作业错误而拒绝运行作业。更正各自的 JOB 参数
W#16#3002	驱动器工作时不允许更改 MODE_IN。请等待当前定位运行结束。
W#16#3003	未知操作模式(MODE_IN)。允许模式为 1 (点动)、3 (参考运行)、4 (相对法)和 5 (绝对法)。
W#16#3004	同一时间只允许一个启动请求。有效的启动请求为 DIR_P、DIR_M 或 START。
W#16#3005	只有"绝对法"操作模式允许 START。使用 DIR_P 或 DIR_M 启动运行
W#16#3006	线性轴和"绝对法"操作模式不允许使用 DIR_P 或 DIR_M。使用 START 启动运行
W#16#3007	轴未同步。只有同步轴时才能使用"绝对法"。
W#16#3008	清除工作范围。只有在点动模式下才允许将运行返回工作位置。
W#16#3101	由于未设置轴参数，因此禁止启动。通过 HWConfig 为"定位"子模块指定参数
W#16#3102	由于未使能驱动器，因此禁止启动。在 SFB 上设置"驱动器使能"(DRV_EN=TRUE)
W#16#3103	由于设置了 STOP，因此禁止启动。在 SFB 中清除 STOP (STOP=FALSE)
W#16#3104	由于驱动器当前正在执行定位运行(WORKING=TRUE)，因此禁止启动。请等待当前定位运行结束。
W#16#3105	由于至少有一个未决错误尚未清除，因此禁止启动。首先，排除并清除所有外部错误，然后重新启动运行。
W#16#3201	SPEED 的速度设定值错误。使用数字量输出进行定位时，只能使用"慢速"(0)和"快速"(1)。
W#16#3301	转换/截止差值过高。将最大转换/截止差值设置为 $10^8$
W#16#3303	转换差值过低。转换差值必须高于/等于截止差值。
W#16#3304	截止差值过低。截止差值大小必须至少为目标范围的一半。
W#16#3401	目标设置超出工作范围。对于线性轴和 Step Approach，目标设置必须在软件限位开关范围内(含此限值)。
W#16#3402	目标设置错误。对于旋转轴，目标设置必须大于 0 并小于旋转轴终点值。
W#16#3403	距离设置错误。"相对法"的行进距离设定值必须为正值。
W#16#3404	距离设置错误。所得结果(绝对目标坐标)必须大于 $-5 \times 10^8$ 。
W#16#3405	距离设置错误。所得结果(绝对目标坐标)必须小于 $5 \times 10^8$ 。
W#16#3406	距离设置错误。所得结果(绝对目标坐标)必须在工作范围(目标范围的 +/- 一半)内
W#16#3501	行进距离过长。目标坐标 + 当前剩余距离必须大于/等于 $-5 \times 10^8$
W#16#3502	行进距离过长。目标坐标 + 当前剩余距离必须小于/等于 $5 \times 10^8$
W#16#3503	行进距离过短。正方向的行进距离必须大于指定的正向截止差值
W#16#3504	行进距离过短。负方向的行进距离必须大于指定的负向截止差值
W#16#3505	行进距离过短，或正方向的限位开关已经启动。正向最后可接近的目标(工作范围或行进距离限制)与当前位置过近。
W#16#3506	行进距离过短，或负方向的限位开关已经启动。负向最后可接近的目标(工作范围或行进距离限制)与当前位置过近。

**作业错误(JOB\_ERR = TRUE)**

如果检测到错误，输出参数 JOB\_ERROR 将置为 TRUE。参数 JOB\_ERROR 给出了错误原因。

事件等级错误代码	解释
W#16#4001	未设置轴参数。通过 HWConfig 为"定位"子模块指定参数
W#16#4002	由于仍在定位，无法执行作业。只有定位未运行时才能执行作业。等到 WORKING = FALSE，再重复执行作业。
W#16#4004	未知作业。检查作业标识号，然后重复执行作业。
W#16#4101	对于线性轴，参考点坐标不能超出工作范围的限制。
W#16#4102	对于线性轴，指定的参考点坐标 + 当前剩余距离必须大于/等于 $-5 \times 10^8$ 。
W#16#4103	对于线性轴，指定的参考点坐标 + 当前剩余距离必须小于/等于 $5 \times 10^8$ 。
W#16#4104	对于线性轴，指定的参考点坐标 + 距起始点的当前剩余差值必须大于/等于 $-5 \times 10^8$ 。
W#16#4105	对于线性轴，指定的参考点坐标 + 距起始点的当前剩余差值必须小于/等于 $-5 \times 10^8$ 。
W#16#4106	对于旋转轴，参考点坐标不能小于 0，也不能大于/等于旋转轴终点。

## 外部错误(ERR)

技术电路会监视运行、行进距离和连接的外围设备。首先必须已在"驱动器"、"轴"和"编码器"参数画面窗体中打开监视功能。

触发监视单元时将报告外部故障。外部错误的发生与已启动的操作无关。必须使用 ERR\_A = TRUE 清除外部错误。

SFB 参数 ERR (WORD)的置位对应外部错误。

监视	错误代码：	ERR-WORD 中的位
零脉冲(零标记)	W#16#0004	2
行进距离	W#16#0800	11
工作范围	W#16#1000	12
实际值	W#16#2000	13
进入目标	W#16#4000	14
目标范围	W#16#8000	15

## 系统错误

系统错误由 BIE = FALSE 指示。系统错误由读取/写入实例 DB 引用或多次调用 SFB 引起。

## 29.3 使用 SFB 47 "COUNT"控制计数器

### 描述

要通过用户程序控制定位功能，请使用 SFB COUNT (SFB 47)。

可以使用下列操作：

- 通过软件门 SW\_GATE 启动/停止计数器
- 使能/控制输出 DO
- 检索状态位 STS\_CMP、STS\_OFLW、STS\_UFLW 和 STS\_ZP
- 检索当前的计数器值 COUNTVAL
- 读取/写入内部计数器寄存器的作业
- 检索当前的持续时间 TIMEVAL

### 参数

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
LADDR	INPUT	WORD	0	CPU 专用	W#16#0300	子模块的 I/O 地址，由用户在"HW 配置"中指定。 如果E和A地址不相等，则必须指定二者中较低的一个。
CHANNEL	INPUT	INT	2	CPU 312C : 0 到 1 CPU 313C : 0 到 2 CPU 314C : 0 至 3	0	通道号：
SW_GATE	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	用于启动/停止计数器的软件门
CTRL_DO	INPUT	BOOL	4.1	TRUE/FALSE	FALSE	启用输出
SET_DO	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	控制输出
JOB_REQ	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	作业初始化 (上升沿)

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
JOB_ID	INPUT	WORD	6	W#16#0000 无功能作业 W#16#0001 写计数值 W#16#0002 写载入值 W#16#0004 写比较值 W#16#0008 写滞后 W#16#0010 写脉冲周期 W#16#0082 读载入值 W#16#0084 读比较值 W#16#0088 读滞后 W#16#0090 读脉冲周期	W#16#0000	作业号
JOB_VAL	INPUT	DINT	8	$-2^{31}$ 至 $+2^{31}-1$	0	写作业的值。
STS_GATE	OUTPUT	BOOL	12.0	TRUE/FALSE	FALSE	内部门的状态
STS_STRT	OUTPUT	BOOL	12.1	TRUE/FALSE	FALSE	硬件门的状态 (开始输入)
STS_LTCH	OUTPUT	BOOL	12.2	TRUE/FALSE	FALSE	锁定输入的状态
STS_DO	OUTPUT	BOOL	12.3	TRUE/FALSE	FALSE	输出状态
STS_C_DN	OUTPUT	BOOL	12.4	TRUE/FALSE	FALSE	反方向状态。 显示始终为计数的最后方向。第一次调用 SFB 后，STS_C_DN 值为 FALSE。
STS_C_UP	OUTPUT	BOOL	12.5	TRUE/FALSE	FALSE	正方向状态 显示始终为计数的最后方向。第一次调用 SFB 后，STS_C_DN 的值为 TRUE。
COUNTVAL	OUTPUT	DINT	14	$-2^{31}$ 至 $+2^{31}-1$	0	实际计数值
LATCHVAL	OUTPUT	DINT	18	$-2^{31}$ 至 $+2^{31}-1$	0	实际锁定值
JOB_DONE	OUTPUT	BOOL	22.0	TRUE/FALSE	TRUE	可以启动新作业
JOB_ERR	OUTPUT	BOOL	22.1	TRUE/FALSE	FALSE	故障作业
JOB_STAT	OUTPUT	WORD	24	0 至 W#16#FFFF	0	作业错误号

**注意**

如果已通过组态界面将"输出的反应"设置为"无比较"，则以下各项有效：

- 输出将以正常输出方式切换。
- SFB 的输入参数 CTRL\_DO 和 SET\_DO 未激活。
- 状态位 STS\_DO 和 STS\_CMP (IDB 中的状态比较器)保持复位状态。

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
STS_CMP	STATIC	BOOL	26.3	TRUE/FALSE	FALSE	状态比较器。 通过 RES_STS 复位。 状态位 STS_CMP 指示满足或已满足比较器的比较条件。 STS_CMP 还指示已设置输出 (STS_DO = TRUE)
STS_OFLW	STATIC	BOOL	26.5	TRUE/FALSE	FALSE	状态溢出 通过 RES_STS 复位。
STS_UFLW	STATIC	BOOL	26.6	TRUE/FALSE	FALSE	状态下溢 通过 RES_STS 复位。
STS_ZP	STATIC	BOOL	26.7	TRUE/FALSE	FALSE	过零状态 通过 RES_STS 复位。 只为无主计数方向的计数器设置。 用于指示过零。 计数器设置为 0 或从载入值=0 开始计数时，也会设置此值。
JOB_OVAL	STATIC	DINT	28	-2 <sup>31</sup> 至 +2 <sup>31</sup> -1	0	读作业的输出值。
RES_STS	STATIC	BOOL	32.2	TRUE/FALSE	FALSE	复位状态位。 将状态位 STS_CMP、STS_OFLW、STS_UFLW 和 STS_ZP 复位。 复位状态位时需要调用两次 SFB。

---

**注意**

有关使用 SFB 47 的更多信息，请参见 *S7-300 可编程控制器 CPU 31xC* 技术功能手册。

---



## 错误信息

### 作业错误

如果发生作业错误，将置 **JOB\_ERR = TRUE**。**JOB\_STAT** 中将给出精确的错误原因。

事件等级错误代码	解释
W#16#0121	比较值过低。
W#16#0122	比较值过高。
W#16#0131	滞后过小。
W#16#0132	滞后过大。
W#16#0141	脉冲周期过短。
W#16#0142	脉冲周期过长。
W#16#0151	载入值过低。
W#16#0152	载入值过高。
W#16#0161	计数器值过低。
W#16#0162	计数器值过高。
W#16#01FF	非法的作业号。

### 系统错误

发生系统错误后，将置 **BIE = False**。

事件等级错误代码	解释
W#16#8001	操作模式错误或参数错误。在"组态硬件"中设置正确的操作模式，或使用与已设置的操作模式相匹配的 SFB
W#16#8009	通道号非法。将通道号设置为≤3 (CPU 的专用值)。

## 29.4 使用 SFB 48 "FREQUENC"控制频率测量

### 描述

可通过用户程序操作频率计数器。这种情况下，必须使用 SFB FREQUENC (SFB48)。

可以使用下列操作：

- 通过软件门 SW\_GATE 开始/停止
- 使能/控制输出 DO
- 检索状态位 STS\_CMP、STS\_OFLW 和 STS\_UFLW
- 检索当前频率值 MEAS\_VAL
- 读取/写入内部频率计数器寄存器的作业

### 参数

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
LADDR	INPUT	WORD	0	CPU 专用	W#16#0300	子模块的 I/O 地址，由用户在"HW 配置"中指定。 如果 I 和 O 地址不同，则必须指定二者中较低的一个。
CHANNEL	INPUT	INT	2	CPU 312C : 0 到 1 CPU 313C : 0 到 2 CPU 314C : 0 至 3	0	通道号：
SW_GATE	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	用于启动/停止频率测量的软件门
MAN_DO	INPUT	BOOL	4.1	TRUE/FALSE	FALSE	手动输出控制使能
SET_DO	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	控制输出
JOB_REQ	INPUT	BOOL	4.3	TRUE/FALSE	FALSE	作业初始化(上升沿)

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
JOB_ID	INPUT	WORD	6	W#16#0000 = 无功能作业 W#16#0001 = 写下限 W#16#0001 = 写上限 W#16#0004 = 写集成时间 W#16#0081 = 读下限 W#16#0081 = 读上限 W#16#0084 = 读集成时间	0	作业号
JOB_VAL	INPUT	DINT	8	-2 <sup>31</sup> 至 +2 <sup>31</sup> -1	0	写作业的值
STS_GATE	OUTPUT	BOOL	12.0	TRUE/FALSE	FALSE	内部门的状态
STS_STRT	OUTPUT	BOOL	12.1	TRUE/FALSE	FALSE	硬件门的状态(开始输入)
STS_DO	OUTPUT	BOOL	12.2	TRUE/FALSE	FALSE	输出状态
STS_C_DN	OUTPUT	BOOL	12.3	TRUE/FALSE	FALSE	反方向状态 显示始终为计数的最后方向。第一次调用 SFB 后，STS_C_DN 值为 FALSE。
STS_C_UP	OUTPUT	BOOL	12.4	TRUE/FALSE	FALSE	正方向状态 显示始终为计数的最后方向。第一次调用 SFB 后，STS_C_UP 的值为 TRUE。
MEAS_VAL	OUTPUT	DINT	14	0 至 +2 <sup>31</sup> -1	0	实际频率值
COUNTVAL	OUTPUT	DINT	18	-2 <sup>31</sup> 至 +2 <sup>31</sup> -1	0	实际计数值(每次门在 0 时打开，开始计数)
JOB_DONE	OUTPUT	BOOL	22.0	TRUE/FALSE	TRUE	可以启动新作业
JOB_ERR	OUTPUT	BOOL	22.1	TRUE/FALSE	FALSE	故障作业
JOB_STAT	OUTPUT	WORD	24	W#16#0000 至 W#16#FFFF	W#16#0000	作业错误号

**注意**

如果已通过组态界面将"输出的反应"设置为"无比较"，则以下各项有效：

- 输出将以正常输出方式切换。
- SFB 输入参数 MAN\_DO 和 SET\_DO 未激活。
- 状态位 STS\_DO 保持复位状态。

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
STS_CMP	STATIC	BOOL	26.3	TRUE/FALSE	FALSE	测量结束状态。 通过 RES_STS 复位。 每次时间间隔期满后， 将更新测量值。 在此，测量结束由状态位 STS_CMP 报告
STS_OFLW	STATIC	BOOL	26.5	TRUE/FALSE	FALSE	状态溢出 通过 RES_STS 复位。
STS_UFLW	STATIC	BOOL	26.6	TRUE/FALSE	FALSE	状态下溢 通过 RES_STS 复位。
JOB_OVAL	STATIC	DINT	28	-2 <sup>31</sup> 至 2 <sup>31</sup> -1	0	读作业的输出值。
RES_STS	STATIC	BOOL	32.2	TRUE/FALSE	FALSE	复位状态位。 复位状态位 STS_CMP、 STS_OFLW、 STS_UFLW。 复位状态位时需要调用 两次 SFB。

#### 注意

有关使用 SFB 48 的更多信息，请参见 *S7-300 可编程控制器 CPU 31xC 技术功能手册*。

#### 作业错误

如果发生作业错误，则 JOB\_ERR = TRUE。JOB\_STAT 中将给出精确的错误原因。

事件等级 错误代码	解释
W#16#0221	集成时间过低。
W#16#0222	集成时间过高。
W#16#0231	频率下限过低。
W#16#0232	频率上限过高。
W#16#0241	频率上限过低。
W#16#0242	频率上限过高。
W#16#02FF	非法的作业号。

## 系统错误

发生系统错误后，将置 **BIE** = False。

事件等级 错误代码	解释
W#16#8001	操作模式错误或参数错误。在"组态硬件"中设置正确的操作模式，或使用与已设置的操作模式相匹配的 SFB
W#16#8009	通道号非法。将通道号设置为 $\leq 3$ (CPU 的专用值)。

## 29.5 使用 SFB 49 "PULSE"控制脉宽调制

### 描述

要通过用户程序控制脉宽调制，请使用 SFB ANALOG (SFB 49)。

可以使用下列操作：

- 通过软件门 SW\_EN 开始/结束
- 使能/控制输出 DO
- 检索状态位 STS\_EN、STS\_STRT 和 STS\_DO
- 输入输出值
- 读/写寄存器的作业

### 参数

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	含义
LADDR	INPUT	WORD	0	CPU 专用	W#16#0300	子模块的 I/O 地址，由用户在"HW 配置"中指定。 如果 E 和 A 地址不相等，则必须指定二者中较低的一个。
CHANNEL	INPUT	INT	2	CPU 312C : 0 到 1 CPU 313C : 0 到 2 CPU 314C : 0 至 3	0	通道号：
SW_EN	INPUT	BOOL	4.0	TRUE/FALSE	FALSE	用于开始/停止输出的软件门
MAN_DO	INPUT	BOOL	4.1	TRUE/FALSE	FALSE	手动输出控制使能
SET_DO	INPUT	BOOL	4.2	TRUE/FALSE	FALSE	控制输出
OUTP_VAL	INPUT	INT	6.0	单位 ppm : 0 1,000 S7 模拟值 : 0 27,648	0	缺省输出值 如果输入的输出值 > 1 000 或 27648，CPU 会将其限制为 1,000 或 27,648
JOB_REQ	INPUT	BOOL	8.0	TRUE/FALSE	FALSE	作业初始化(上升沿)

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	含义
JOB_ID	INPUT	WORD	10	W#16#0000 = 无功能作业 W#16#0001 = 写周期 W#16#0001 = 写延时 W#16#0004 = 写最小脉冲周期 W#16#0081 = 读周期 W#16#0081 = 读延时 W#16#0084 = 读最小脉冲周期	W#16#0000	作业号
JOB_VAL	INPUT	DINT	12	-2 <sup>31</sup> 至 +2 <sup>31</sup> -1	0	写作业的值。
STS_EN	OUTPUT	BOOL	16.0	TRUE/FALSE	FALSE	状态使能
STS_STRT	OUTPUT	BOOL	16.1	TRUE/FALSE	FALSE	硬件门的状态(开始输入)
STS_DO	OUTPUT	BOOL	16.2	TRUE/FALSE	FALSE	输出状态
JOB_DONE	OUTPUT	BOOL	16.3	TRUE/FALSE	TRUE	可以启动新作业
JOB_ERR	OUTPUT	BOOL	16.4	TRUE/FALSE	FALSE	故障作业
JOB_STAT	OUTPUT	WORD	18	W#16#0000 至 W#16#FFFF	W#16#0000	作业错误号

未分配给块的参数(本地统计数据)：

参数	声明	数据类型	地址 (实例 DB)	取值范围：	默认	描述
JOB_OVAL	OUTPUT	DINT	20	-2 <sup>31</sup> 至 2 <sup>31</sup> -1	0	读作业的输出值。

#### 注意

有关使用 SFB 49 的更多信息，请参见 *S7-300 可编程控制器 CPU 31xC 技术功能手册*。

## 作业错误

如果发生作业错误，则 **JOB\_ERR** = TRUE。**JOB\_STAT** 中将给出精确的错误原因。

事件等级 错误代码	解释
W#16#0411	周期过短。
W#16#0412	周期过长。
W#16#0421	延时过短。
W#16#0422	延时过长。
W#16#0431	最小脉冲周期过短。
W#16#0432	最小脉冲周期过长。
W#16#04FF	非法的作业号。

## 系统错误

发生系统错误后，将置 **BIE** = False。

事件等级 错误代码	解释
W#16#8001	操作模式错误或参数错误。在"组态硬件"中设置正确的操作模式，或使用与已设置的操作模式相匹配的 SFB
W#16#8009	通道号非法。将通道号设置为 ≤ 3 (CPU 的专用值)。



## 29.6 使用 SFB 60 "SEND\_PTP"发送数据(ASCII , 3964(R))

### 描述

可通过 SFB SEND\_PTP (SFB 60)从 DB 传送数据块。

调用数据块后且控制输入 REQ 为上升沿，将执行发送操作。

要传送的数据范围由 SD\_1 (DB 编号和起始地址)确定。数据块的长度由 LEN 确定。

要使能 SFB 处理作业，必须使用 R(复位)=FALSE 调用。当控制输入 R 为上升沿时，将中止当前传送并将 SFB 复位为基本状态。被中止的作业将以错误消息(STATUS 输出)终止。

对于子模块，应在 LADDR 中声明已在"HW 配置"中指定的 I/O 地址。

如果作业无错误终止，则将 DONE 置为 TRUE，如果作业错误终止，则将 ERROR 置为 TRUE。

如果处理作业后 DONE=TRUE，则表示：

- 使用 ASCII 驱动程序时：数据已传送给通讯伙伴。不保证通讯伙伴已收到所有数据。
- 使用过程 3964(R)时：数据已传送给通讯伙伴，并且通讯伙伴已确认收到。不保证数据传送到通讯伙伴的 CPU。

在 STATUS 中，CPU 将指示错误，或出现警告时指示各自的事件 ID。

复位(R=TRUE) SFB 时，还将输出 DONE 或 ERROR/STATUS。

如果已发生错误，则复位二进制结果 BIE。如果块无错终止，二进制结果的状态为 TRUE。

### 注意

SFB 中不包括参数检查。如果参数化出现错误，CPU 可能会进入 STOP 模式。

### 背景数据块

执行 SFB SEND\_PTP 时还需要实例 DB。DB 编号通过调用传递。不允许访问实例 DB 中的数据。

## 参数

参数	声明	数据类型	取值范围：	默认	描述
REQ	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"请求"：在上升沿激活数据交换。
R	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"复位"中止作业。锁定传送。
LADDR	INPUT	WORD	CPU 专用	W#16#03FF	子模块的 I/O 地址，由用户在"HW 配置"中指定。
DONE	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数(此参数仅为调用持续时间设置)： 1. FALSE 表示作业尚未开始或仍在执行。 2. TRUE 表示作业已执行，无错误。
ERROR	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数(此参数仅为调用持续时间设置)：作业完成，无错误
STATUS	OUTPUT	WORD	W#16#0000 至 W#16#FFFF	W#16#0000	状态参数(此参数仅为调用持续时间设置)：要显示状态，应将 STATUS 复制到空闲数据区) 根据 ERROR 位的不同，STATUS 具有下列含义： <ul style="list-style-type: none"> <li>• ERROR=FALSE： STATUS 的值为 W#16#0000：既不是警告也不是错误 STATUS 的值 &lt;&gt; W#16#0000：警告，STATUS 提供详细信息。</li> <li>• ERROR=TRUE： 发生错误，STATUS 提供有关错误类型的详细信息。</li> </ul>
SD_1	INPUT/ OUTPUT	ANY	CPU 专用	0	发送参数： 在此输入下列值： 1st 要从中传送数据的 DB 的编号。 2nd 数据字节编号，数据将从此编号开始传送。 例如：从 DB 10 的第 2 字节开始 -> DB10.DBB2 注意：对于 S7-300 CPU，请注意，参数 SD_1 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。
LEN	INPUT/ OUTPUT	INT	1 1024	1	在此声明要传送的数据块的长度。(在此间接设置长度。)

## 数据一致性

数据的一致性限制为 206 字节。如果要使超过 206 个字节的数据传送保持一致性，必须注意以下的问题：  
除非传送已终止，否则请勿向发送范围 SD\_1 的当前使用部分写入数据。这种情况下，状态参数 DONE 的值为 TRUE。

## 29.7 使用 SFB 61 "RCV\_PTP"接收数据(ASCII , 3964(R))

### 描述

使用 SFB RCV\_PTP (SFB 61)可以接收数据，然后将其归档到数据块。

数据块被调用后，且控制输入 EN\_R 为 TRUE 时，数据块即准备就绪接收数据。将参数 EN\_R 的信号状态置为 FALSE，则可取消当前传送。被取消的作业将以错误消息(STATUS 输出)终止。只要参数 EN\_R 的信号状态置为 FALSE，此输入关闭。

接收区域在 RD\_1 (DB 编号和起始地址)中声明。数据块长度在 LEN 中声明。

要使能 SFB 处理作业，必须使用 R(复位)=FALSE 调用。当控制输入 R 为上升沿时，将中止当前传送并将 SFB 复位为基本状态。被取消的作业将以错误消息(STATUS 输出)终止。

对于子模块，应在 LADDR 中声明已在"HW 配置"中指定的 I/O 地址。

如果作业无错误终止，则将 NDR 置为 TRUE，如果作业错误终止，则将 ERROR 置为 TRUE。

在 STATUS 中，CPU 将指示错误，或者出现警告时，指示各自的事件 ID。

复位(R=TRUE) SFB 时，还将输出 NDR 或 ERROR/STATUS (参数 LEN = 16#00)。

如果已发生错误，则复位二进制结果 BIE。如果块无错终止，二进制结果的状态为 TRUE。

### 注意

SFB 中不包括参数检查。如果组态出现错误，CPU 可能会跳转到 STOP 模式。

### 背景数据块

执行 SFB RCV\_PTP 时还需要实例 DB。DB 编号通过调用传递。不允许访问实例 DB 中的数据。

## 参数

参数	声明	数据类型	取值范围：	默认	描述
EN_R	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"使能接收"： 接收使能。
R	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"复位"。 中止作业。
LADDR	INPUT	WORD	CPU 专用	W#16#03FF	子模块的 I/O 地址，由用户在"HW 配置"中指定。
NDR	OUTPUT	BOOL	TRUE/FALSE	FALSE	作业完成，无错误，数据被接受。 <ul style="list-style-type: none"> <li>FALSE 表示作业尚未开始或仍在执行</li> <li>TRUE 表示作业已成功完成。</li> </ul>
ERROR	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数(此参数仅为调用持续时间设置)： 作业完成，无错误
STATUS	OUTPUT	WORD	W#16#0000 至 W#16#FFFF	W#16#0000	状态参数(此参数仅为调用持续时间设置)：要 显示状态，应将 STATUS 复制到空闲数据区) 根据 ERROR 位的不同，STATUS 具有下 列含义： <ul style="list-style-type: none"> <li>ERROR=FALSE： STATUS 的值为 W#16#0000： 既不是警告也不是错误 STATUS 的值 &lt;&gt; W#16#0000： 警告，STATUS 提供详细信息。</li> <li>ERROR=TRUE： 发生错误，STATUS 提供有关错误类 型的详细信息。</li> </ul>
RD_1	INPUT/ OUTPUT	ANY	CPU 专用	0	接收参数： 在此声明： 1st 要存储已接收数据的 DB 的编号。 2nd 数据字节编号，数据将从此编号开始 存储。 例如：从 DB 20 的第 5 字节开始 -> DB10.DBB2  注意：对于 S7-300 CPU，请注意，参数 RD_1 始终要求对 DB 参数进行完整的指定 (例如 P#DB13.DBX0.0 中的第 100 字节)。 对于 S7-300 CPU，忽略明确的 DB 编号是 不允许的，这将会导致在用户程序中生成 出错消息。
LEN	INPUT/ OUTPUT	INT	0 1024	0	数据长度(字节数)输出

### 数据一致性

数据的一致性限制为 206 字节。如果要使超过 206 字节的数据传送保持一致性，必须注意如下问题：  
全部数据都已接收(NDR = TRUE)前，请勿访问接收 DB。然后，锁定接收 DB (EN\_R = FALSE)，直到完成数据处理。

## 29.8 使用 SFB 62 "RES\_RCVB"删除接收缓冲区(ASCII , 3964(R))

### 描述

使用 SFB RES\_RECV (SFB 62)可以清除模块的整个输入缓冲区。系统将放弃存储的所有报文。调用 SFB RES\_RCVB 时，会存储进入的消息帧。

调用数据块后且控制输入 **REQ** 为上升沿，将激活作业。作业可在多个调用(程序周期)间运行。

要使能 SFB 处理作业，必须使用 **R(复位)=FALSE** 调用。当控制输入 R 为上升沿时，将取消删除过程并将 SFB 复位为基本状态。被取消的作业将以错误消息(STATUS 输出)终止。

对于子模块，应在 **LADDR** 中声明已在"HW 配置"中指定的 I/O 地址。

如果作业无错误终止，则将 **DONE** 置为 TRUE，如果作业错误终止，则将 **ERROR** 置为 TRUE。

在 **STATUS** 中，CPU 将指示错误，或者出现警告时，指示各自的事件 ID。

复位(R=TRUE) SFB 时，还将输出 **DONE** 或 **ERROR/STATUS**。

如果已发生错误，则复位二进制结果 **BIE**。如果块无错终止，二进制结果的状态为 TRUE。

### 注意

SFB 中不包括参数检查。如果参数化出现错误，CPU 可能会跳转到 STOP 模式。

### 背景数据块

执行 SFB RES\_RCVB 时还需要实例 DB。DB 编号通过调用传递。不允许访问实例 DB 中的数据。

## 参数

参数	声明	数据类型	取值范围：	默认	描述
REQ	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"请求"： 在上升沿激活作业。
R	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"复位"。 中止作业。
LADDR	INPUT	WORD	CPU 专用	W#16#03FF	在"HW 配置"中设置的子模块的 I/O 地址。
DONE	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数(此参数仅为调用持续时间设置)： <ul style="list-style-type: none"> <li>FALSE 表示作业尚未开始或仍在执行。</li> <li>TRUE 表示作业已执行，无错误。</li> </ul>
ERROR	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数(此参数仅为调用持续时间设置)： 作业完成，无错误
STATUS	OUTPUT	WORD	W#16#0000 至 W#16#FFFF	W#16#0000	状态参数(此参数仅为调用持续时间设置)： 要显示状态，应将 STATUS 复制到空闲数据区) 根据 ERROR 位的不同，STATUS 具有下列含义： <ul style="list-style-type: none"> <li>ERROR=FALSE： STATUS 的值为 W#16#0000： 既不是警告也不是错误 STATUS 的值 &lt;&gt; W#16#0000： 警告，STATUS 提供详细信息。</li> <li>ERROR=TRUE： 发生错误，STATUS 提供有关错误类型的详细信息。</li> </ul>



## 29.9 使用 SFB 63 "SEND\_RK"发送数据(512(R))

### 描述

可通过 SFB SEND\_PTP (SFB 63)从数据块发送数据。

调用数据块后且控制输入 REQ 为上升沿，将执行发送操作。

要传送的数据范围由 SD\_1 (DB 编号和起始地址)确定。数据块的长度由 LEN 确定。

在 SFB 中，还可声明伙伴站的接收范围。CPU 在消息帧报头中输入此信息(请参见附录 F)，并将其传送给伙伴。

目标由以下各项指定：CPU 号 R\_CPU(仅适用于多处理器通讯)、R\_TYPE 中的数据类型(数据块(DB)和扩展数据块(DX))、R\_DBNO 中的数据块号以及 R\_OFFSET 中距写入第一个字节的偏移量。

在 R\_CF\_BYT 和 R\_CF\_BIT 中，声明伙伴 CPU 上的连接存储器字节和位。

在参数 SYNC\_DB 中，声明将在其中存储数据的 DB，该数据用于所有 SFB 在启动和同步期间的初始化。用户程序中所有 SFB 的 DB 编号必须相同。

要使能 SFB 处理作业，必须使用 R(复位)=FALSE 调用。当控制输入 R 为上升沿时，将取消当前发送操作，并将 SFB 复位为基本状态。被取消的作业将以错误消息(STATUS 输出)终止。

对于子模块，应在 LADDR 中声明已在"HW 配置"中指定的 I/O 地址。

如果作业无错误终止，则将 DONE 置为 TRUE，如果作业错误终止，则将 ERROR 置为 TRUE。

处理完作业且 DONE = TRUE 后，数据被发送到通讯伙伴，通讯伙伴确认收到，并将其传送到伙伴 CPU。

在 STATUS 中，CPU 将指示错误，或者出现警告时，指示各自的事件 ID。

复位(R=TRUE) SFB 时，还将输出 DONE 或 ERROR/STATUS。

如果已发生错误，则复位二进制结果 BIE。如果块无错终止，二进制结果的状态为 TRUE。

### 注意

SFB 中不包括参数检查。如果参数化错误，CPU 可能会跳转到 STOP 模式。

### 背景数据块

执行 SFB SEND\_RK 时还需要实例 DB。DB 编号通过调用传递。不允许访问实例 DB 中的数据。

## 发送数据的特性

"发送数据"时，要考虑下列特性：

- 使用 RK512，只能发送偶数数据。如果声明了奇数长度(LEN)的数据，将会在所发送的数据后额外追加具有值"0"的附加填充字节。
- 在 RK512 中，只能声明偶数偏移量。如果声明了奇数偏移量，数据将从下一个较小偶数偏移量开始存储在伙伴站中。

实例：偏移量是 7，则从第 6 字节开始存储数据。

## 参数

参数	声明	数据类型	取值范围：	默认	描述
SYNC_DB	INPUT	INT	CPU 专用，不允许零	0	存储用于 RK-SFB 同步的公用数据的数据块编号(最小长度 = 240 字节)。
REQ	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"请求"：在上升沿激活作业。
R	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"复位"中止作业。
LADDR	INPUT	WORD	CPU 专用	W#16#03FF	子模块的 I/O 地址，由用户在"HW 配置"中指定。
R_CPU	INPUT	INT	0 - 4	1	伙伴 CPU 的 CPU 编号(仅用于多处理器操作)
R_TYPE	INPUT	CHAR	"D"、"X"	"D"	伙伴 CPU 上的地址类型(只允许大写字母) "D"：数据块 "X"：扩展数据块
R_DBNO	INPUT	INT	0 至 255	0	伙伴 CPU 上的数据块编号
R_OFFSET	INPUT	INT	0 510 (只限偶数值)	0	伙伴 CPU 上的数据字节编号
R_CF_BYT	INPUT	INT	0 至 255	255	伙伴 CPU 上的连接存储器位(255: 表示：无连接存储器位)
R_CF_BIT	INPUT	INT	0 7	0	伙伴 CPU 上的连接存储器位
DONE	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数(此参数仅为调用持续时间设置)： 1st FALSE 表示作业尚未开始或仍在执行。 2nd TRUE 表示作业已执行，无错误。
ERROR	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数(此参数仅为调用持续时间设置)： 作业完成，无错误

参数	声明	数据类型	取值范围：	默认	描述
<b>STATUS</b>	OUTPUT	WORD	W#16#0000 至 W#16#FFFF	W#16#0000	<p>状态参数(此参数仅为调用持续时间设置)：要显示状态，应将 STATUS 复制到空闲数据区)</p> <p>根据 ERROR 位的不同，STATUS 具有下列含义：</p> <ul style="list-style-type: none"> <li>• ERROR=FALSE： STATUS 的值为 W#16#0000： 既不是警告也不是错误 STATUS 的值 &lt;&gt; W#16#0000： 警告，STATUS 提供详细信息。</li> <li>• ERROR=TRUE： 发生错误，STATUS 提供有关错误类型的详细信息。</li> </ul>
<b>SD_1</b>	INPUT/ OUTPUT	ANY	CPU 专用	0	<p>发送参数：</p> <p>在此声明：</p> <p>1st 要从中传送数据的 DB 的编号。</p> <p>2nd 数据字节编号，数据将从此编号开始传送。</p> <p>例如：从 DB 10 的第 2 字节开始 -&gt; DB10.DBB2</p> <p>注意：对于 S7-300 CPU，请注意，参数 SD_1 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。</p>
<b>LEN</b>	INPUT/ OUTPUT	INT	1 1024	1	<p>在此声明要传送的数据块的长度。(在此间接设置长度。)</p>

### 消息帧中的声明

下表显示了 RK 512 消息帧的消息帧报头中的声明。

S7 自动化系统上的源 (本地 CPU)	目标伙伴 CPU	消息帧报头，字节		
		3/4 指令类型	5/6 D-DBNR/D 偏移量	7/8 计数单位为
数据块	数据块	AD	DB/DW	字
数据块	扩展数据块	AD	DB/DW	字

所用缩写的解释：

D-DBNR	目标数据块编号
D 偏移量	目标起始地址
DW	以字计的偏移量

### 数据一致性

数据的一致性限制为 128 字节。如果要使超过 128 字节的数据传送保持一致性，必须注意如下问题：  
除非传送已终止，否则请勿向发送范围 SD\_1 的当前使用部分写入数据。这种情况下，状态参数 DONE 的值为 TRUE。

## 29.10 使用 SFB 64 "FETCH RK"获取数据(RK 512)

### 描述

**SFB FETCH\_RK (SFB 64)**用于从伙伴获取数据块并将其存储在数据块中。

调用数据块后且控制输入 **REQ** 为上升沿，将执行发送操作。

存储所获取数据的区域在 **RD\_1** (DB 编号和起始地址)中声明。数据块长度在 **LEN** 中声明。

在 SFB 中，还可指定从中获取数据的伙伴区域。CPU 在 RK512 消息帧报头中输入该信息，并将其传送给伙伴。

伙伴区域由以下各项确定：**R\_CPU** 中的 CPU 编号(只适用于多处理器通讯)、**R\_TYPE** 中的数据类型(数据块、扩展数据块、存储位、输入、输出、计数器和定时器)、**R\_DBNO** 中的数据块编号(只适用于数据块和扩展数据块)和 **R\_OFFSET** 中获取第一个字节的偏移量。

在 **R\_CF\_BYT** 和 **R\_CF\_BIT** 中，可声明伙伴 CPU 上的连接存储字节和连接存储位。

在参数 **SYNC\_DB** 中，声明将在其中存储数据的 DB，该数据用于所有 SFB 在启动和同步期间的初始化。用户程序中所有 SFB 的 DB 编号必须相同。

要使能 SFB 处理作业，必须使用 **R(复位)=FALSE** 调用。当控制输入 R 为上升沿时，将取消当前传送，并将 SFB 复位为基本状态。取消的作业将以错误消息(**STATUS** 输出)结束。

对于子模块，应在 **LADDR** 中声明已在"HW 配置"中指定的 I/O 地址。

如果作业无错误终止，则将 **DONE** 置为 TRUE，如果作业错误终止，则将 **ERROR** 置为 TRUE。

在 **STATUS** 中，CPU 将指示错误，或者出现警告时，指示各自的事件 ID。

复位(**R=TRUE**) SFB 时，还将输出 **DONE** 或 **ERROR/STATUS**。

如果已发生错误，则复位二进制结果 **BIE**。如果块无错终止，二进制结果的状态为 TRUE。

### 注意

SFB 中不包括参数检查。如果组态出现错误，CPU 可能会进入 STOP 模式。

### 注意

从 CPU 获取数据时，必须为 CPU 编程 SFB "SERVE\_RK"。

### 背景数据块

执行 SFB **FETCH\_RK** 时还需要实例 DB。DB 编号通过调用传递。不允许访问实例 DB 中的数据。

**(扩展)数据块的特性**

从数据块或扩展数据块"获取数据"时，请注意下列特性：

- 使用 RK512，只能发送偶数数据。如果输入奇数长度(LEN)，则会传送一个附加字节。但是，在目标数据块中，始终输入正确的数据数。
- 在 RK512 中，只能声明偶数偏移量。如果声明了奇数偏移量，数据将从下一个较小偶数偏移量开始存储在伙伴站中。

实例：偏移量是 7，则从第 6 字节开始存储数据。

**定时器和计数器的特性**

从通讯伙伴获取定时器或计数器时，必须注意要为每个定时器或计数器拿来两个字节。例如，如果要获取 10 个计数器，则必须声明长度为 20。

**参数**

注意：在 SFB 中，值的范围都以德国存储器惯例表示。

参数	声明	数据类型	取值范围：	默认	描述
SYNC_DB	INPUT	INT	CPU 专用，不允许零	0	存储用于 RK-SFB 同步的公用数据的数据块编号(最小长度 = 240 字节)。
REQ	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"请求"：在上升沿激活作业。
R	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"复位"中止作业。
LADDR	INPUT	WORD	CPU 专用	W#16#03FF	子模块的 I/O 地址，由用户在"HW 配置"中指定。
R_CPU	INPUT	INT	0 - 4	1	伙伴 CPU 的 CPU 编号(仅用于多处理器操作)
R_TYPE	INPUT	CHAR	"D"、"X"、"M"、"E"、"A"、"Z"、"T"	"D"	伙伴 CPU 上的地址类型 "D"：数据块 "X"：扩展数据块 "M"：存储器位 "E"：输入 "A"：输出 "Z"：计数器 "T"：定时器
R_DBNO	INPUT	INT	0 至 255	0	伙伴 CPU 上的数据块编号
R_OFFSET	INPUT	INT	参考表格： "数据源(伙伴 CPU)功能块中的参数"	0	伙伴 CPU 上的数据字节编号
R_CF_BYT	INPUT	INT	0 至 255	255	伙伴 CPU 上的连接存储器位(255: 表示：无连接存储器位)
R_CF_BIT	INPUT	INT	0 7	0	伙伴 CPU 上的连接存储器位

参数	声明	数据类型	取值范围：	默认	描述
DONE	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数 (此参数仅为调用持续时间设置)： <ul style="list-style-type: none"> <li>FALSE 表示作业尚未开始或仍在执行。</li> <li>TRUE 表示作业已执行，无错误。</li> </ul>
ERROR	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数 (此参数仅为调用持续时间设置)： 作业完成，无错误
STATUS	OUTPUT	WORD	W#16#0000 至 W#16#FFFF	W#16#0000	状态参数(此参数仅为调用持续时间设置)：要显示状态，应将 STATUS 复制到空闲数据区) 根据 ERROR 位的不同，STATUS 具有下列含义： <ul style="list-style-type: none"> <li>ERROR=FALSE： STATUS 的值为 W#16#0000： 既不是警告也不是错误 STATUS 的值 &lt;&gt; W#16#0000： 警告，STATUS 提供详细信息。</li> <li>ERROR=TRUE： 发生错误，STATUS 提供有关错误类型的详细信息。</li> </ul>
RD_1	INPUT/ OUTPUT	ANY	CPU 专用	0	接收参数： 在此声明： 1st 存储获取数据的数据块编号。 2nd 数据字节编号，获取的数据将从此编号开始存储。 例如：从 DB 10 的第 2 字节开始 -> DB10.DBB2 注意：对于 S7-300 CPU，请注意，参数 RD_1 始终要求对 DB 参数进行完整的指定(例如 P#DB13.DBX0.0 中的第 100 字节)。对于 S7-300 CPU，忽略明确的 DB 编号是不允许的，这将会导致在用户程序中生成出错消息。
LEN	INPUT/ OUTPUT	INT	1 1024	1	在此声明要获取的数据块的字节长度。 每个定时器和每个计数器必须声明两个字节。(在此间接设置长度。)

**数据源(伙伴 CPU)SFB 中的参数**

下表给出了可发送的数据类型。  
R\_OFFSET 的值由伙伴 CPU 确定。

伙伴 CPU 上的源	R_TYP	R_NO	R_OFFSET(单位：字节)
数据块	"D"	0 - 255	0 - 510 · 只适用偶数值
扩展数据块	"X" :	0 - 255	0 - 510 · 只适用偶数值
存储器位	"M"	无关	0 - 255
输入	"E"	无关	0 - 255
输出	"A"	无关	0 - 255
计数器	"Z"	无关	0 - 255
定时器	"T"	无关	0 - 255



## 消息帧中的声明

下表给出了 RK512 消息帧的消息帧报头中的声明。

伙伴 CPU 上的源	目标, S7 自动化系统 (本地 CPU)	消息帧报头, 字节		
		3/4 指令类型	5/6 S-DBNR/S 偏移量	7/8 计数单位为 偏移量
数据块	数据块	ED	DB/DW	字
扩展数据块	数据块	EX	DB/DW	字
存储器位	数据块	EM	字节地址	字节
输入	数据块	EI	字节地址	字节
输出	数据块	EO	字节地址	字节
计数器	数据块	EC	计数器编号	字
定时器	数据块	ET	定时器编号	字

所用缩写的解释：

S-DBNO	源数据块编号
S 偏移量	源起始地址

## 数据一致性

数据的一致性限制为 128 字节。如果要使超过 128 字节的数据传送保持一致性，必须注意如下问题：  
除非传送已终止，否则请勿向发送范围 SD\_1 的当前使用部分写入数据。这种情况下，状态参数 DONE 的值设置为 TRUE。

## 29.11 使用 SFB 65 "SERVE\_RK"接收和提供数据(RK 512)

### 描述

使用 SFB SERVE\_RK (SFB 65)接收和提供数据。

- 接收数据：数据存储在由伙伴在 RK512 消息帧报头中指定的区域中。通讯伙伴执行"发送数据"(SEND 作业)作业时，需要调用 SFB。
- 提供数据：数据从由伙伴在 RK512 消息帧报头中的指定的区域中获取。通讯伙伴执行"获取数据"(FETCH 作业)作业时，需要调用 SFB。

块被调用后，且控制输入 EN\_R 的值为 TRUE 时，即准备就绪接收数据。将参数 EN\_R 的信号状态置为 FALSE，则可取消当前传送。被取消的作业将以错误消息(STATUS 输出)终止。只要参数 EN\_R 的信号状态置为 FALSE，此输入关闭。

在参数 SYNC\_DB 中，声明将在其中存储数据的 DB，该数据用于所有 SFB 在启动和同步期间的初始化。用户程序中所有 SFB 的 DB 编号必须相同。

要使能 SFB 处理作业，必须使用 R(复位)=FALSE 调用。当控制输入 R 为上升沿时，将取消当前传送，并将 SFB 复位为基本状态。被取消的作业将以错误消息(STATUS 输出)终止。

对于子模块，应在 LADDR 中声明已在"HW 配置"中指定的 I/O 地址。

如果作业无错误终止，则将 NDR 置为 TRUE，如果作业错误终止，则将 ERROR 置为 TRUE。

调用 SFB 时，如果 NDR=TRUE，CPU 将在参数 L\_TYPE、L\_DBNO 和 L\_OFFSET 中指定要将数据存储到的区域或从中获取数据的区域。系统还会为调用显示相关作业的 L\_CF\_BYT 和 L\_CF\_BIT 以及长度 LEN 参数。系统还会为调用显示相关作业的 L\_CF\_BYTL\_CF\_BITLEN 参数。

在 STATUS 中，CPU 将指示错误，或者出现警告时，指示各自的事件标 ID(请参考附录)。

"复位"(R=TRUE) SFB 时，还将输出 NDR 或 ERROR/STATUS (参数 LEN = 16#00)。

如果已发生错误，则复位二进制结果 BIE。如果块无错终止，二进制结果的状态为 TRUE。

### 注意

SFB 中不包括参数检查。如果组态出现错误，CPU 可能会进入 STOP 模式。

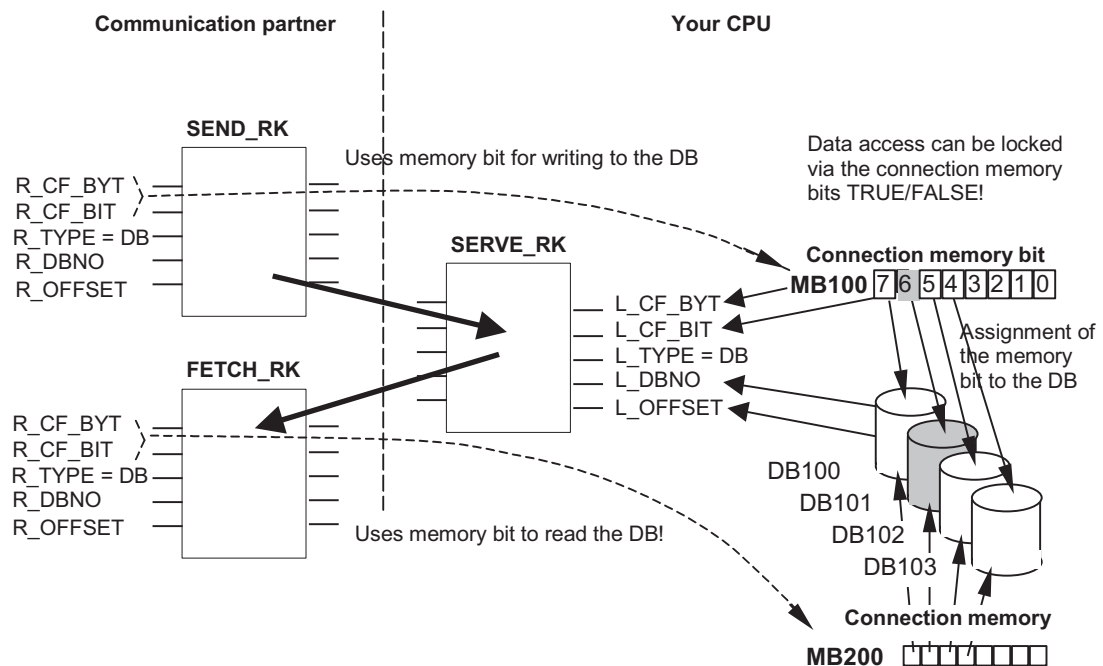
### 背景数据块

执行 SFB SERVE\_RK 时还需要实例 DB。DB 编号通过调用传递。不允许访问实例 DB 中的数据。

## 如何使用连接存储器位

通过连接存储器位可以锁定或使能通讯伙伴的 SEND 和 FETCH 作业。从而可防止覆盖或读取还未处理的数据。

可以为每个作业都指定连接存储器位。



### 实例：使用连接存储器位执行 SEND\_RK

在本例中，通讯伙伴将数据传送到用户的 CPU 上的 DB 101

在用户的 CPU 中，将连接存储器位 100.6 设置为 FALSE。

在通讯伙伴中，为 SEND 作业指定连接存储器位 100.6 (参数 R\_CF\_BYT、R\_CF\_BIT)。

连接存储器位在 RK 512 消息帧报头中传送到 CPU。

处理作业前，CPU 会验证在 RK512 消息帧报头中指定的连接存储器位。只有在 CPU 上将连接存储器位置为 FALSE 时，才会处理作业。如果连接存储器位置为 TRUE，错误消息"32 hex"将通过响应消息帧传送到通讯伙伴。

数据传送到 DB101 后，SFB SERVE 将把用户 CPU 上的连接存储器 100.6 置为 TRUE。另外，在调用的持续时间，SFB SERVE 上将输出连接存储器字节和位(如果 NDR = TRUE)。

在用户程序中判断连接存储器(连接存储器 100.6 = TRUE)时，可查看作业是否已完成以及是否可以处理所传送的数据。

在用户程序中处理完数据后，必须将连接存储器 100.6 复位到 FALSE。否则，伙伴将不能再次无错执行该作业。

## 参数

注意：在 SFB 中，值的范围都以德国存储器惯例表示。

参数	声明	数据类型	取值范围：	默认	描述
SYNC_DB	INPUT	INT	CPU 专用	0	存储用于 RK-SFB 同步的公用数据的数据块编号(最小长度 = 240 字节)。
EN_R	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"使能接收"作业使能
R	INPUT	BOOL	TRUE/FALSE	FALSE	控制参数"复位"中止作业。
LADDR	INPUT	WORD	CPU 专用	W#16#03FF	子模块的 I/O 地址，由用户在"HW 配置"中指定。
NDR	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数"新数据就绪"(此参数仅为调用持续时间设置)： 1. FALSE 表示作业尚未开始或仍在执行。 2. TRUE 表示作业已成功执行。
ERROR	OUTPUT	BOOL	TRUE/FALSE	FALSE	状态参数(此参数仅为调用持续时间设置)： 作业完成，无错误
STATUS	OUTPUT	WORD	W#16#0000 至 W#16#FFFF	W#16#0000	状态参数(此参数仅为调用持续时间设置)：要显示状态，应将 STATUS 复制到空闲数据区) 根据 ERROR 位的不同，STATUS 具有下列含义： <ul style="list-style-type: none"> <li>• ERROR=FALSE： STATUS 的值为 W#16#0000： 既不是警告也不是错误 STATUS 的值 &lt;&gt; W#16#0000： 警告，STATUS 提供详细信息。</li> <li>• ERROR=TRUE： 发生错误，STATUS 提供有关错误类型的详细信息。</li> </ul>

参数	声明	数据类型	取值范围：	默认	描述
L_TYPE	OUTPUT	CHAR	"D"  "D"、"X"、"M"、 "E"、"A"、"Z"、 "T"	' '	接收数据： 本地 CPU 上的目标区域类型 (只允许大写字母)： "D"：数据块 提供数据： 本地 CPU 上的源区域类型 (只允许大写字母)： "D"：数据块 "M"：存储器位 "E"：输入 "A"：输出 "Z"：计数器 "T"：定时器 该参数仅对调用的持续时间 设置。
L_DBNO	OUTPUT	INT	CPU 专用	0	本地 CPU 上的数据块编号。该 参数仅对调用的持续时间设置。
L_OFFSET	OUTPUT	INT	0 - 510	0	本地 CPU 上的数据字节编号。该 参数仅对调用的持续时间设置。
L_CF_BYT	OUTPUT	INT	0 至 255	0	本地 CPU 上的连接存储器字 节。该参数仅对调用的持续时 间设置。 (255: 表示：无连接存储器位)
L_CF_BIT	OUTPUT	INT	0 7	0	本地 CPU 上的连接存储器位。该 参数仅对调用的持续时间设置。
LEN	INPUT/ OUTPUT	INT	0 1024	0	消息帧的长度，以字节计算的 数目(该参数仅对调用的持续时 间设置)。

## 数据一致性

数据的一致性限制为 128 字节。如果要使超过 128 字节的数据传送保持一致性，必须注意如下问题：  
使用连接存储器功能。数据全部传送完毕前不要访问数据(对该作业指定的连接存储器的判断；如果  
NDR = TRUE，则连接存储器在调用 SFB 时处于激活状态)。完成数据处理前，不要将连接存储器复  
位为 FALSE。

## 29.12 SFB 60 至 65 的补充错误信息

### 错误信息

下表给出了不同的事件类别和事件标识号。

"SFB 参数组态中的错误"		
事件等级 错误代码	事件	纠正方法
W#16#0301	源/目标数据类型非法或不存在。 非法范围 (起始地址, 长度)。 DB 非法或不存在。 其它数据类型非法或不存在。 连接存储器字节或位标识号无效。	验证参数设置, 如有需要进行更正。 伙伴在消息帧报头中传递了非法参数。 验证参数, 如有需要则创建一个块。 有关允许的数据类型的信息, 请参考作业表。 伙伴在消息帧报头中传递了错误参数。
W#16#0303	无法访问的范围	验证参数。有关允许的起始地址和长度的信息, 请参考作业表, 或者伙伴在消息帧报头中提供了错误参数。
作业处理错误		
事件等级 错误代码	事件	纠正方法
W#16#0501	当前作业因重新启动或复位而中止。	重复执行被中止的作业。当通过 PG 重新分配参数时, 应确保写入接口前没有其它正在处理的作业。
W#16#0502	处于此操作状态时, 不允许该作业 (例如, 没有为设备接口分配任何参数)。	为设备接口分配参数。
W#16#050E	<ul style="list-style-type: none"> <li>消息帧长度无效</li> <li>在允许的最大长度内, 未出现在参数中分配的消息结束标识号。</li> </ul>	<ul style="list-style-type: none"> <li>消息帧长度 &gt; 1024 字节。选择较小的消息帧长度</li> <li>在发送缓冲区中期望的位置添加消息结束标识号。</li> </ul>
W#16#0513	数据类型错误(DB ...): 未知数据类型或非法数据类型(例如, DE) 在 SFB 中指定的源和目标数据类型不匹配。	有关允许的数据类型及其组合的信息, 请参考作业表。
W#16#0515	协调存储器中声明的位编号错误。	允许的位编号为 0 至 7
W#16#0516	指定的 CPU 编号太高。	允许的 CPU 编号为 0、1、2、3 或 4
W#16#0517	传送的数据 > 1024 字节, 过大	将作业拆分成几个较小长度的作业。
W#16#051D	发送/接收作业因以下原因中止 <ul style="list-style-type: none"> <li>通讯块复位</li> <li>重新分配了参数</li> </ul>	重复调用通讯块。
W#16#0522	已开始了新的 SEND 作业, 即使前一作业尚未完成。	前一发送作业以 DONE 或 ERROR 终止前, 不要开始新的 SEND 作业。

"SFB 参数组态中的错误		
处理伙伴作业时的错误"(仅限 RK512)		
事件等级 错误代码	事件	纠正方法
W#16#0601	第一个指令字节出错(非 00 或 FFH)	伙伴中的基本报头结构错误。如有必要,可通过将接口测试仪挂接至数据链路来检验伙伴设备的错误情况。
W#16#0602	第三个指令字节出错(非 A、0 或 E)	伙伴中的基本报头结构错误。如有必要,可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0603	后续消息帧的第三个指令字节出错(指令与第一个消息帧中的指令不同)	伙伴中的基本报头结构错误。如有必要,可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0604	第四个指令字节出错(指令字符错误)	伙伴的基本报头结构错误或请求了非法指令组合。检查允许的指令。如有必要,可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0606	第五个指令字节出错(非法数据块编号)	有关允许的数据块编号、起始地址或长度,请参考作业表。
W#16#0607	第五个或第六个指令字节出错(起始地址过高)	有关允许的数据块编号、起始地址或长度,请参考作业表。
W#16#0609	第九个或第十个指令字节出错(该数据类型不允许协调存储器位或位号太高)。	伙伴中的基本报头结构错误。有关何时允许协调存储器位的信息,请参考作业表。
W#16#060A	第十个指令字节出错(非法 CPU 编号)	伙伴中的基本报头结构错误。
发送错误		
事件等级 错误代码	事件	纠正方法
W#16#0701	<b>仅适用于 3964(R) :</b> 发送第一个重复 : <ul style="list-style-type: none"> <li>• 发送消息帧时检测出错误</li> <li>• 伙伴以否定确认字符(NCC)请求了重复。</li> </ul>	重复不代表错误。但是,可能表示数据链路中断或伙伴的错误行为。如果在允许的最大重复次数内未传送消息帧,会报告一个错误号,说明首次发生的错误。
W#16#0702	<b>仅适用于 3964(R) :</b> 建立连接时的错误 传送 STX 后,收到 NCC 或其它字符(除 DLE 或 STX 外)。	如有必要,可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0703	<b>仅适用于 3964(R) :</b> 超出确认时间(QVZ) : 传送 STX 后,伙伴在确认延迟时间内没有响应。	伙伴设备太慢或尚未作好接收准备,或数据链路已中断。如有必要,可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0704	<b>仅适用于 3964(R) :</b> 由伙伴中止 : 发送操作正在进行时,从伙伴中收到一个或多个字符。	检查伙伴是否因以下原因也指示一项错误:尚未收到所有已传送数据(例如,由于数据链路中断)、未决致命错误、或伙伴设备的错误行为。如有必要,可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0705	<b>仅适用于 3964(R) :</b> 传送正在进行时收到否定确认	检查伙伴是否因以下原因也指示一项错误:尚未收到所有已传送数据(例如,由于数据链路中断)、未决致命错误、或伙伴设备的错误行为。如有必要,可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。

"SFB 参数组态中的错误"		
W#16#0706	<p><b>仅适用于 3964(R) :</b>                      传送结束错误 :</p> <ul style="list-style-type: none"> <li>• 伙伴最终以 NCC 或任何字符 (除 DLE 外)拒绝了消息帧</li> <li>• 确认字符(DLE)接收过早。</li> </ul>	检查伙伴是否因以下原因也指示一项错误：尚未收到所有已传送数据(例如，由于数据链路中断)、未决致命错误、或伙伴设备的错误行为。如有必要，可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0707	<p><b>仅适用于 3964(R) :</b>                      超出传送结束确认延迟时间/响应监视狗时间：                      连接断开后，伙伴未在 QVZ 内以 DLE ETX 发出响应。</p>	伙伴设备太慢或中断。如有必要，可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0708	<p><b>仅限 ASCII 驱动程序 :</b>                      超出 XON 的等待时间。</p>	通讯伙伴中断、太慢或切换到离线。如有必要，检查通讯伙伴或改变参数。
W#16#0709	<p><b>仅适用于 3964(R) :</b>                      无法建立连接，超出了允许的启动尝试次数。</p>	检查接口线或传送参数。 还要在伙伴方检查 CPU 和 CP 间接收功能的参数是否已正确分配。
W#16#070A	<p><b>仅适用于 3964(R) :</b>                      无法传送数据，超出了允许的传送尝试次数。</p>	检查接口线或传送参数。
W#16#070B	<p><b>仅适用于 3964(R) :</b>                      由于两个伙伴都设置为高优先级，无法解决初始化冲突。</p>	改变参数。
W#16#070C	<p><b>仅适用于 3964(R) :</b>                      由于两个伙伴都设置为低优先级，无法解决初始化冲突。</p>	改变参数。



"SFB 参数组态中的错误		
接收错误		
事件等级 错误代码	事件	纠正方法
W#16#0801	仅适用于 3964(R) : 等待第一个重复 : 收到消息帧时, 检测到错误, 并且 CPU 以否定确认(NCC)向伙伴请求重复。	重复不代表错误。但是, 可能表示数据链路中断或伙伴的错误行为。如果在允许的最大重复次数内未传送消息帧, 会报告一个错误号, 说明首次发生的错误。
W#16#0802	仅适用于 3964(R) : 建立连接时的错误 <ul style="list-style-type: none"> <li>空闲时收到一个或多个字符(除 NCC 或 STX 外)</li> <li>收到 STX 之后, 伙伴未等待 DLE 响应即发送了其它字符。</li> </ul> 伙伴设备启动后 : <ul style="list-style-type: none"> <li>启动伙伴设备后, CPU 收到未定义字符。</li> </ul>	如有必要, 可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0805	仅适用于 3964(R) : 逻辑接收错误 : 接收 DLE 之后, 又收到了另一个字符(除 DLE 和 ETX 外)。	检查伙伴是否在消息帧报头和数据字符串中加入两个 DLE, 或是否通过 DLE ETX 建立了连接。如有必要, 可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0806	超出字符延迟时间(CDT) : <ul style="list-style-type: none"> <li>在 CDT 内未收到两个后续字符。</li> </ul> 仅适用于 3964(R) : <ul style="list-style-type: none"> <li>1. 建立连接后发送 DLE 时, 在 CDT 内未收到字符。</li> </ul>	伙伴设备太慢或中断。如有必要, 可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0807	消息帧长度非法 : 收到消息帧的长度为 0。	收到长度为 0 的消息帧并不代表错误。 检查通讯伙伴发送的消息帧中为何没有用户数据。
W#16#0808	仅适用于 3964(R) : 块检查字符错误 BCC : 内部生成的 BCC 值与伙伴在通讯链接结束时收到的 BCC 不一致。	检查通讯是否严重中断。这种情况下, 还可以见到偶发错误代码。如有必要, 可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0809	<b>仅适用于 3964(R) :</b> 块重复的延迟时间已到期	在通讯伙伴和用户模块的参数中声明相同的块延迟时间。如有必要, 可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#080A	没有空闲输入缓冲区 : 没有可用于接收的已清空的输入缓冲区。	必须增加调用 SFB RCV 的频率。
W#16#080C	传送错误 <ul style="list-style-type: none"> <li>检测到传送错误(奇偶校验/停止位/溢出错误)。</li> </ul> 仅适用于 3964(R) : <ul style="list-style-type: none"> <li>如果在空闲模式收到中断的字符, 则立即报告错误, 以便尽快发现对数据链路的干扰影响。</li> </ul> 仅适用于 3964(R) : <ul style="list-style-type: none"> <li>如果这种错误在发送和接收操作期间发生, 则启动重复。</li> </ul>	对数据链路的干扰会引起消息帧的重复, 并因此降低用户数据的吞吐量。不检测错误的危险也会增加。改变系统结构或连线。 检查通讯伙伴的数据链路或双方设备上设置的波特率、奇偶校验和停止位是否相同。
W#16#080D	中断 : 与伙伴的接收链接中断。	再次建立链接或启动伙伴设备。

"SFB 参数组态中的错误"		
W#16#080E	未启用数据流控制时，输入缓冲区溢出。	必须在用户程序中更为频繁地调用用于接收的 SFB，或必须为通讯分配数据流控制参数。
W#16#0810	校验错误	检查通讯伙伴的数据链路或双方设备上设置的波特率、奇偶校验和停止位是否相同。
W#16#0811	字符帧错误	检查通讯伙伴的数据链路或双方设备上设置的波特率、奇偶校验和停止位是否相同。 改变系统结构或连线。
W#16#0812	仅限 ASCII 驱动程序： CPU 发送 XOFF 后，又接收到其它字符。	重新分配通讯伙伴的参数或加快处理数据。
W#16#0814	仅限 ASCII 驱动程序： 由于未使用数据流控制执行操作，丢失了一个或多个消息帧。	尽可能使用数据流控制操作。利用全部输入缓冲区。在基本参数中，将"对 CPU STOP 模式的反应"参数设置为"继续运行"。
W#16#0816	收到的消息帧长度超出指定的最大长度。	需要在伙伴站中更正。

"SFB 参数组态中的错误		
从通讯伙伴收到具有错误的响应消息帧或错误消息帧		
事件等级 错误代码	事件	纠正方法
W#16#0902	<b>仅限 RK 512</b> ：伙伴站的存储器访问错误 (无存储器) 伙伴设备为 SIMATIC S5 时： <ul style="list-style-type: none"> <li>• 显示字范围错误</li> <li>• 无数据范围(DB/DX 除外)</li> <li>• 数据范围不足(DB/DX 除外)</li> </ul>	检查伙伴是否配备了所需的数据范围、该数据范围是否有足够大小，或检查所调用 SFB 的参数。 检查在 SFB 中指定的长度。
W#16#0903	<b>仅限 RK 512</b> ：伙伴站中的 DB/DX 访问错误(DB/DX 不存在或过短) 伙伴设备为 SIMATIC S5 时： <ul style="list-style-type: none"> <li>• DB/DX 不存在</li> <li>• DB/DX 过短</li> <li>• DB/DX 编号非法</li> </ul> FETCH 作业超出了允许的源范围。	检查伙伴是否配备了所需的数据范围、该数据范围是否有足够大小，或检查所调用 SFB 的参数。 检查在 SFB 中指定的长度。
W#16#0904	<b>仅限 RK 512</b> ：伙伴报告"不允许此作业类型"。	由于 CPU 从未输出系统指令，伙伴行为错误。
W#16#0905	<b>仅限 RK 512</b> ：伙伴错误或 SIMATIC S5 伙伴错误： <ul style="list-style-type: none"> <li>• 不允许的源/目标类型</li> <li>• 伙伴设备发生存储器错误</li> <li>• 伙伴 CP/CPU 通讯出错</li> <li>• 伙伴 PLC 处于 STOP 模式</li> </ul>	检查伙伴是否能够传送所需的数据类型。 检查伙伴的硬件结构。 将伙伴的 PLC 切换到 RUN 模式。
W#16#0908	<b>仅限 RK 512</b> ：伙伴设备发现同步错误：消息帧序列中断。	自身重新启动或重新启动伙伴的 PLC 时出现此错误。这是系统正常的启动操作。无需纠正。在运行期间，由于之前的错误，也可能出现此错误。否则，可以视为伙伴的错误行为。
W#16#0909	<b>仅限 RK 512</b> ：伙伴设备中，DB/DX 被协调存储器位锁定。	在伙伴程序中：处理完最后的发送数据后，复位协调存储器位！ 程序：重复作业！
W#16#090A	<b>仅限 RK 512</b> ：伙伴发现消息帧报头出错：3. 报头中的指令字节错误	检查该错误是否由干扰或错误的伙伴行为引起。通过挂接到数据链路的接口测试仪对此进行检查。
W#16#090C	<b>仅限 RK 512</b> ：伙伴检测到消息帧长度(总长度)错误。	检查该错误是否由干扰或错误的伙伴行为引起。通过挂接到数据链路的接口测试仪对此进行检查。
W#16#090D	<b>仅限 RK 512</b> ：到目前为止，伙伴设备未进行重新启动。	重新启动伙伴 PLC 或将工作模式选择开关设置为 RUN。
W#16#090E	<b>仅限 RK 512</b> ：在响应消息帧中收到未知错误号。	检查该错误是否由干扰或错误的伙伴行为引起。通过挂接到数据链路的接口测试仪对此进行检查。

"SFB 参数组态中的错误		
"CPU 在伙伴的响应消息帧中检测到错误		
事件等级 错误代码	事件	纠正方法
W#16#0A02	<b>仅限 RK 512</b> ：收到的响应消息帧的结构出错 (1. 字节不是 00 或 FF)	如有必要，可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0A03	<b>仅限 RK 512</b> ：收到的响应消息帧所含数据过多或不足。	如有必要，可通过将接口测试仪挂接至数据链路来检验伙伴的错误状况。
W#16#0A05	<b>仅限 RK 512</b> ：在监视时间内，没有来自伙伴的响应消息帧。	伙伴设备的速度很慢吗？ 此错误也经常因前一错误显示。例如，发送 FETCH 消息帧后会显示接收过程错误 (事件类别 8)。 原因：由于干扰或监视狗超时，无法接收响应消息帧。如果伙伴设备在响应最后收到的 FETCH 消息帧之前重启动，也可能产生此错误。
警告		
事件等级 错误代码	事件	纠正方法
W#16#0B01	输入缓冲区的装载量超出其容量的 2/3	增加调用接收块的频率，以避免输入缓冲区溢出。

## 30 用于 H CPU 的 SFC

### 30.1 在 H 系统中使用 SFC 90 "H\_CTRL"控制操作

#### 描述

使用 SFC 90 "H\_CTRL"，可在下列方面影响 H 系统：

- 可防止主站 CPU 中的待机链接。通过 SFC 90 "H\_CTRL"取消此设置，或者 H 系统进入 STOP 模式时会禁用此功能。  
系统会存储所有来自待机 CPU 与主站链接的请求。
- 可在主站 CPU 上禁用更新。通过 SFC 90 "H\_CTRL"取消此设置，或者 H 系统进入 STOP 模式时会禁用此功能。  
系统会存储所有来自待机 CPU 的更新请求。



---

#### 当心

如果已经禁用更新但尚未连接，硬件系统仍可如先前一样确定连接状态。请注意，主站 CPU 在连接时不处理任何删除/插入中断、站故障/返回的中断或机架故障/返回的中断。

---

- 可从循环自检中删除测试组件，再次添加该组件或立即启动。如果已经禁用了循环自检中的一个或多个测试组件，使用 SFC 90 "H\_CTRL"取消该设置或 H 系统进入 STOP 模式前，被禁用的组件始终保持禁用状态。

#### 注意：

如果在冗余系统中使用 CPU 414-4H 或 417-4H，请注意以下内容：如果禁用组件超过 24 小时，CPU 会进入 STOP 模式。对于冗余系统，适用规则规定某些测试必须在 24 小时之内完成。

- 可以执行通过编程设定的主站-待机切换。可以执行以下切换：  
当前待机 CPU 成为主站 CPU。  
机架 0 中的 CPU 成为主站 CPU。  
机架 1 中的 CPU 成为主站 CPU。

下表说明了输入参数 MODE 和 SUBMODE 允许的组合。

作业	MODE 输入	SUBMODE 输入
禁用链接	3	0
重新使能链接	4	0
禁用更新	1	0
重新使能更新	2	0
在循环自检中删除在 SUBMODE 中指定的测试组件。 测试组件只能删除一次。	20	0.1,...5
向循环自检中重新添加在 SUBMODE 中指定的测试组件。 测试组件只有先前已删除过才能再次添加。	21	0.1,...5
立即启动在 SUBMODE 中指定的测试组件。 此测试组件不能被删除。	22	0.1,...5
当前待机 CPU 成为主站 CPU。	5	0
机架 0 中的 CPU 成为主站 CPU。	30	0
机架 1 中的 CPU 成为主站 CPU。	31	0

下表给出了使用 SUBMODE 输入值对循环自检的各测试组件的分配情况。(仅与输入 MODE 的值 20、21 和 22 相关)

SUBMODE 的值	相关测试组件
0	SP7 – ASIC – Test
1	代码存储器测试
2	数据存储器测试
3	操作系统代码检验和测试
4	代码块检验和测试
5	冗余操作中编码、时间、标记和数据块的比较。

## SFC 的工作方式

SFC 90 "H\_CTRL"是异步 SFC，换言之，其执行可贯穿多个 SFC 调用。

通过用 REQ=1 调用 SFC 90 启动作业。

如果可以立即执行作业，SFC 会在 BUSY 输出参数中返回 0。即使测试包括多个周期 (MODE=22，RET\_VAL=W#16#0001)，长期测试例行程序的初始化也以第一个 SFC 调用(BUSY=0)结束。如果 BUSY 的值为 1，则作业仍处于活动状态(参见异步 SFC 的 REQ、RET\_VAL 和 BUSY 参数含义)。

## 识别作业

作业由输入参数 MODE 和 SUBMODE 指定。如果这些参数与尚未完成的作业匹配，则 SFC 调用为后续调用。

## 参数

参数	声明	数据类型	存储器区	描述
REQ	INPUT	BOOL	I、Q、M、D、L、常数	电平触发的控制参数 REQ= 1 : 触发作业
MODE	INPUT	BYTE	I、Q、M、D、L、常数	作业
SUBMODE	INPUT	WORD	I、Q、M、D、L、常数	二级作业
RET_VAL	OUTPUT	INT	I、Q、M、D、L	如果在执行功能时出错，则返回值将包含故障代码。确保每次执行块后，都判断 RET_VAL。
BUSY	OUTPUT	BOOL	I、Q、M、D、L	BUSY = 1 : 作业尚未完成。

## 错误信息

错误代码 (W#16#...)	解释
0000	作业已无错执行
7000	初次调用时 REQ = 0 : 作业未激活 ; BUSY 的值为 0。
7001	初次调用时 REQ = 1 : 作业已启动 ; BUSY 的值为 1。
7002	后续调用(与 REQ 无关)。激活的作业仍在运行 ; BUSY 的值为 1。
0001	<ul style="list-style-type: none"> <li>MODE=1 时 : 已禁用更新。</li> <li>MODE=3 时 : 已禁用链接。</li> <li>MODE=22 时 : 测试组件已运行，不能重启动。</li> <li>MODE=30 时 : 由于 CPU 0 已经为主站 CPU，因此，无法执行可编程的主站 - 待机切换。</li> <li>MODE=31 时 : 由于 CPU 1 已经为主站 CPU，因此，无法执行可编程的主站 - 待机切换。</li> </ul>
8080	由于系统不处于冗余模式，因此，不能在 H 系统的当前系统状态下执行可编程的主站-待机切换。
8082	<ul style="list-style-type: none"> <li>MODE=1 时 : 已激活更新，不能再禁用。</li> <li>MODE=3 时 : 已激活链接，不能再禁用。</li> <li>MODE=20 时 : 指示的测试组件从循环自检中删除。</li> <li>MODE=21 时 : 指示的测试组件尚未从循环自检中删除。</li> <li>MODE=22 时 : 指示的测试组件已从循环自检中删除，无法执行。</li> </ul>
8090	MODE 输入参数值无效。
8091	SUBMODE 输入参数值无效。
8xyy	常见出错信息，请参见通过输出参数 RET_VAL 判断错误

### 使用 SFC 90 的实例

通过 SFC 90 "H\_CTRL"，可以确保需要 CPU 的最大资源时未启动链接和更新。

要实现此目的，可在进程活动增加期间前，将下列程序段加入主站 CPU。

- 以 MODE = 3 和 SUBMODE = 0 调用 SFC 90 (禁用链接)
- 以 MODE = 1 和 SUBMODE = 0 调用 SFC 90 (禁用更新)

在活动增加期间结束时，将下列程序段加入主站 CPU：

- 以 MODE = 4 和 SUBMODE = 0 调用 SFC 90 (重新启用链接)
- 以 MODE = 2 和 SUBMODE = 0 调用 SFC 90 (重新启用更新)。



## 31 集成功能(对于带集成 I/O 的 CPU)

### 31.1 SFB 29 (HS\_COUNT)

#### 描述

通过 SFB 29 "HS\_COUNT" (计数器) , 可影响具有集成 I/O 的 CPU 的集成计数器功能的以下方面 :

- 设置和输入起始值。
- 选择和设置比较值。
- 使能计数器。
- 使能数字输出。
- 读取当前计数值和当前比较值。
- 查询计数值和比较值之间的关系。

#### 更多信息

SFB 29 各参数及集成功能计数器参数和 CPU 硬件输入与输出参数的含义 ,在 S7-300 可编程控制器 , 集成功能手册中有详细描述。

#### 参数

参数	声明	数据类型	存储器区	描述
PRES_COUNT	INPUT	DINT	I、Q、M、D、 L、常数	计数器的起始值
PRES_COMP_A	INPUT	DINT	I、Q、M、D、 L、常数	新比较值 COMP_A
PRES_COMP_B	INPUT	DINT	I、Q、M、D、 L、常数	新比较值 COMP_B
EN_COUNT	INPUT		I、Q、M、D、L	使能计数器
EN_DO	INPUT	BOOL	I、Q、M、D、 L、常数	使能数字输出
SET_COUNT	INPUT	BOOL	I、Q、M、D、 L、常数	为起始值 PRES_COUNT 设置输出
SET_COMP_A	INPUT	BOOL	I、Q、M、D、 L、常数	为比较值 COMP_A 设置输出
SET_COMP_B	INPUT	BOOL	I、Q、M、D、 L、常数	为比较值 COMP_B 设置输出
COUNT	OUTPUT	DINT	I、Q、M、D、L	计数器的实际值
COMP_A	OUTPUT	DINT	I、Q、M、D、L	当前比较值 COMP_A
COMP_B	OUTPUT	DINT	I、Q、M、D、L	当前比较值 COMP_B

31.1 SFB 29 (HS\_COUNT)

参数	声明	数据类型	存储器区	描述
STATUS_A	OUTPUT	BOOL	I、Q、M、D、L	状态位 STATUS_A 1 : COUNT ≥ COMP_A 0 : COUNT < COMP_A
STATUS_B	OUTPUT	BOOL	I、Q、M、D、L	状态位 STATUS_B 1 : COUNT ≥ COMP_B 0 : COUNT < COMP_B

## 31.2 SFB 30 (FREQ\_MES)

### 描述

通过 SFB 30 "FREQ\_MES" (频率计), 可影响具有集成 I/O 的 CPU 的集成频率计功能的以下方面:

- 选择和设置比较值。
- 输出测量的频率。
- 读取当前比较值。
- 查询测量的频率与比较值的关系。

### 更多信息

SFB 30 各参数及集成频率计功能参数和 CPU 硬件输入与输出参数的含义, 在 S7-300 可编程控制器, 集成功能手册中有详细描述。

### 参数

参数	声明	数据类型	存储器区	描述
PRES_U_LIMIT	INPUT	DINT	I、Q、M、D、L、常数	新(上限)比较值 U_LIMIT
PRES_L_LIMIT	INPUT	DINT	I、Q、M、D、L、常数	新(下限)比较值 L_LIMIT
SET_U_LIMIT	INPUT	BOOL	I、Q、M、D、L、常数	为新比较值 U_LIMIT 设置输入
SET_L_LIMIT	INPUT	BOOL	I、Q、M、D、L、常数	为新比较值 L_LIMIT 设置输入
FREQ	OUTPUT	DINT	I、Q、M、D、L	测得的频率(以 MHz 计)
U_LIMIT	OUTPUT	DINT	I、Q、M、D、L	当前比较值(上限)
L_LIMIT	OUTPUT	DINT	I、Q、M、D、L	当前比较值(下限)
STATUS_U	OUTPUT	BOOL	I、Q、M、D、L	状态位 "1": FREQ > U_LIMIT "0": FREQ ≤ U_LIMIT
STATUS_L	OUTPUT	BOOL	I、Q、M、D、L	状态位 "1": FREQ < L_LIMIT "0": FREQ ≥ L_LIMIT

## 31.3 SFB 38 (HSC\_A\_B)

### 描述

通过 SFB 38 (HSC\_A\_B)，可影响具有集成输入/输出的 CPU 的集成 A/B 计数器功能以下的方面：

- 指定和采用起始值
- 指定和设置比较值
- 使能计数器
- 使能数字输出
- 读取当前计数值和当前比较值
- 查询与比较值相关的计数值

SFB 38 (HSC\_A\_B)可在集成功能的实例 DB 的用户程序中读写数据。A/B 计数器由两个计数器 A 和 B 组成，这两个计数器既可同时计数又相互独立(可进行向上计数和向下计数)。

两个计数器的运行方式相同，记录的计数脉冲频率可达 10 千赫。

### 更多信息

SFB 38 参数的确切含义及集成功能 A/B 计数器的参数和 CPU 硬件输入与输出的参数，在手册 S7-300 可编程控制器，集成函数 CPU 312 IFM/314 IFM 中有详细描述。

### 参数

参数	声明	数据类型	存储器区	描述
PRES_COMP	INPUT	DINT	I、Q、M、D、L、常数	新比较值 COMP
EN_COUNT	INPUT	BOOL	I、Q、M、D、L、常数	使能计数器
EN	INPUT	BOOL	I、Q、M、D、L、常数	使能数字输出
SET_COMP	INPUT	BOOL	I、Q、M、D、L、常数	为比较值 COMP 设置输入
COUNT	OUTPUT	DINT	I、Q、M、D、L	计数器的实际值
COMP	OUTPUT	DINT	I、Q、M、D、L	当前比较值 COMP
ENO	OUTPUT	BOOL	I、Q、M、D、L	错误处理： 1：执行中无错误 0：执行时出错

## 31.4 SFB 39 (POS)

### 描述

通过 SFB 39 (POS), 可以影响具有集成输入/输出的 CPU 的集成定位功能。SFB 39 (POS)具有下列功能：

- 同步
- 执行微动模式
- 定位

集成定位功能的 SFB 39 (POS)可从用户程序中读取数据并写入具有集成功能的实例 DB。集成定位功能可从非对称的 24 V 增量式编码器获得最高频率为 10 千赫的信号。通过 CPU 314 IFM 的指定集成输出，它可控制快速/慢速模式或频率转换器(控制定位)

### 更多信息

SFB 39 参数的确切含义及集成功能 A/B 计数器的参数和 CPU 硬件输入与输出的参数,在手册 S7-300 可编程控制器，集成功能 CPU 312 IFM/314 IFM 中有详细描述。

### 参数

参数	声明	数据类型	存储器区	描述
EN	INPUT	BOOL	I、Q、M、D、L、常数	数字输入使能
DEST_VAL	INPUT	DINT	I、Q、M、D、L、常数	集成定位功能的目标位置
REF_VAL	INPUT	DINT	I、Q、M、D、L、常数	同步参考点
SWITCH_OFF_DIFF	INPUT	WORD	I、Q、M、D、L、常数	行程增量中的关闭差值 (关闭点和目标位置的差值)
PRES_COMP	INPUT	DINT	I、Q、M、D、L、常数	新比较值 COMP
BREAK	INPUT	BYTE	I、Q、M、D、L、常数	控制遍历移动的最大模拟值
POS_MODE1、 POS_MODE2	INPUT	BOOL	I、Q、M、D、L、常数	启动和执行微动模式
POS_STRT	INPUT	BOOL	I、Q、M、D、L、常数	在上升沿启动定位操作
SET_POS	INPUT	BOOL	I、Q、M、D、L、常数	出现上升沿时，输入参数 REF_VAL 的值 将作为新实际值
ENO	OUTPUT	BOOL	I、Q、M、D、L	错误处理： 1：执行中无错误 0：执行时出错
ACTUAL_POS	OUTPUT	DINT	I、Q、M、D、L	当前值

参数	声明	数据类型	存储器区	描述
POS_READY (状态消息)	OUTPUT	BOOL	I、Q、M、D、L	如果 POS_READY=1，定位/微动完成
REF_VALID (状态消息)	OUTPUT	BOOL	I、Q、M、D、L	是否到达参考点开关
POS_VALID (状态消息)	OUTPUT	BOOL	I、Q、M、D、L	轴的实际位置已与集成功能的当前位置同步

## 32 塑料技术

### 32.1 SFC 63 (AB\_CALL)

#### 描述

SFC 63 (AB\_CALL)可调用汇编码块。  
汇编码块是用"C"语言编写或在汇编器中编写、然后编译的逻辑块。

#### 应用

只能对 CPU 614 使用汇编码块。

#### 更多信息

SFC 63 各参数的含义在 CPU 614 的文档中详细说明。编写汇编码块有单独的编程指南。

#### 参数

参数	声明	数据类型	存储器区	描述
AB_NUMBER	INPUT	WORD	I、Q、M、D、L、 常数	要调用的汇编码块的位
CALL_REASON	INPUT	WORD	I、Q、M、D、L、 常数	调用 SFC 的组织块、对计算 DB 指针 (参数 DB_NUMBER), 或激活调试器
DB_NUMBER	INPUT	WORD	I、Q、M、D、L、 常数	DB 指针的编号
RET_VAL	OUTPUT	WORD	I、Q、M、D、L	SFC 的返回值





## 33 诊断数据

### 33.1 诊断数据结构概述

#### 系统数据的数据记录 0 和 1

模块的诊断数据位于系统数据区的数据记录 0 和 1 中(参见数据记录的写和读)。

- 数据记录 0 包含 4 个字节的诊断数据，用来描述信号模块的当前状态。
- 数据记录 1 包含
  - 4 个字节的诊断数据(也在数据记录 0 中)和
  - 模块专用的诊断数据。

#### 诊断数据的结构和内容

该部分描述诊断数据各字节的结构和内容。

出错时，相应位将置为"1"。

## 33.2 诊断数据

诊断数据的结构和内容：

字节	位	含义	说明	数据记录	
0	0	模块故障		0 和 1	
	1	内部错误			
	2	外部错误			
	3	通道错误			
	4	无外部辅助电压			
	5	无前置连接器			
	6	无参数分配			
	7	模块参数错误			
1	0 到 3	模块等级	0101: 模拟模块 0000 : CPU 1000 : 功能模块 1100 : CP 1111 : 数字模块 0011 : DP 标准从站 1011 : I 从站 0100 : IM	0 和 1	
	4	通道信息存在			
	5	用户信息存在			
	6	诊断中断来自替换件			
	7	维护要求(仅适用 PROFINET IO)			
	2	0	无存储卡或存储卡错误		0 和 1
		1	通信问题		
		2	模式	0: RUN 1: STOP	
3		响应周期监视			
4		内部模块电源电压故障			
5		电池耗尽			
6		整个备用电池故障			
7		维护请求(仅适用 PROFINET IO)			
3	0	扩展机架故障		0 和 1	
	1	处理器故障			
	2	EPROM 错误			
	3	RAM 错误			
	4	ADC/DAC 错误			
	5	保险丝断开			
	6	硬件中断丢失			
	7	保留			

字节	位	含义	说明	数据记录
4	0 到 6	通道类型	B#16#70 : 数字量输入 B#16#72 : 数字量输出 B#16#71 : 模拟量输入 B#16#73 : 模拟量输出 B#16#74 : FM-POS B#16#75 : FM-REG B#16#76 : FM-ZAEHL B#16#77 : FM-TECHNO B#16#78 : FM-NCU B#16#79 至 B#16#7D : 保留 B#16#7E : US300 B#16#7F : 保留	1
	7	是否还有其它通道类型 ?	0: 否 1: 是	
5	0 到 7	模块每个通道的诊断位数输出。	每个通道的诊断位数 向上取整到字节边界	1
6	0 到 7	模块每种通道类型的通道数	如果一个模块上存在不同的通道类型， 则对于每种通道类型，在数据记录 1 中 从字节 4 开始向前重复这一结构。	1
7	0	通道错误通道 0/通道组 0	通道错误向量的第一个字节(通道错误向 量的长度取决于通道数并向上取整到字 节边界。)	1
	1	通道错误通道 1/通道组 1		
	2	通道错误通道 2/通道组 2		
	3	通道错误通道 3/通道组 3		
	4	通道错误通道 4/通道组 4		
	5	通道错误通道 5/通道组 5		
	6	通道错误通道 6/通道组 6		
	7	通道错误通道 7/ 通道组 7		
...	-	通道专有错误(参见通道专有诊断数据 的结构)		1

### 33.3 通道专有诊断数据的结构

#### 通道专有错误

从紧随通道错误向量的字节开始，指示模块每个通道的通道专有错误。下表显示不同通道类型的通道专有诊断数据的结构。各个位的含义如下：

- 1 = 错误
- 0 = 无错

#### 模拟量输入通道

模拟量输入通道的诊断字节：

位	含义	说明
0	组态/参数分配错误	可通过 SFC 52 和 EVENTN = W#16#8x50 发出信号
1	共模错误	可通过 SFC 52 和 EVENTN = W#16#8x51 发出信号
2	P 短路	可通过 SFC 52 和 EVENTN = W#16#8x52 发出信号
3	M 短路	可通过 SFC 52 和 EVENTN = W#16#8x53 发出信号
4	断线	可通过 SFC 52 和 EVENTN = W#16#8x54 发出信号
5	参考通道错误	可通过 SFC 52 和 EVENTN = W#16#8x55 发出信号
6	电流低于测量范围	可通过 SFC 52 和 EVENTN = W#16#8x56 发出信号
7	电流高于测量范围	可通过 SFC 52 和 EVENTN = W#16#8x57 发出信号

**模拟量输出通道**

模拟量输出通道的诊断字节：

位	含义	说明
0	组态/参数分配错误	可通过 SFC 52 和 EVENTN = W#16#8x60 发出信号
1	共模错误	可通过 SFC 52 和 EVENTN = W#16#8x61 发出信号
2	P 短路	可通过 SFC 52 和 EVENTN = W#16#8x62 发出信号
3	M 短路	可通过 SFC 52 和 EVENTN = W#16#8x63 发出信号
4	断线	可通过 SFC 52 和 EVENTN = W#16#8x64 发出信号
5	0	保留
6	空载电压	可通过 SFC 52 和 EVENTN = W#16#8x66 发出信号
7	0	保留

**数字量输入通道**

数字量输入通道的诊断字节：

位	含义	说明
0	组态/参数分配错误	可通过 SFC 52 和 EVENTN = W#16#8x70 发出信号
1	接地错误	可通过 SFC 52 和 EVENTN = W#16#8x71 发出信号
2	P 短路(传感器)	可通过 SFC 52 和 EVENTN = W#16#8x72 发出信号
3	M 短路	可通过 SFC 52 和 EVENTN = W#16#8x73 发出信号
4	断线	可通过 SFC 52 和 EVENTN = W#16#8x74 发出信号
5	无传感器电源	可通过 SFC 52 和 EVENTN = W#16#8x75 发出信号
6	0	保留
7	0	保留

### 数字量输出通道

数字量输出通道的诊断字节：

位	含义	说明
0	组态/参数分配错误	可通过 SFC 52 和 EVENTN = W#16#8x80 发出信号
1	接地错误	可通过 SFC 52 和 EVENTN = W#16#8x81 发出信号
2	P 短路	可通过 SFC 52 和 EVENTN = W#16#8x82 发出信号
3	M 短路	可通过 SFC 52 和 EVENTN = W#16#8x83 发出信号
4	断线	可通过 SFC 52 和 EVENTN = W#16#8x84 发出信号
5	保险丝断开	可通过 SFC 52 和 EVENTN = W#16#8x86 发出信号
6	空载电压	可通过 SFC 52 和 EVENTN = W#16#8x86 发出信号
7	过热	可通过 SFC 52 和 EVENTN = W#16#8x87 发出信号

## 34 系统状态列表(SSL)

### 34.1 系统状态列表(SSL)概述

本章描述系统状态列表中与以下项相关的所有部分列表：

- CPU
  - 模块，其部分列表并非模块特有。(例如 SSL-ID W#16#00B1、W#16#00B2、W#16#00B3)。
- 模块特有的部分列表，例如 CP 和 FM 的部分列表，将在特定模块的描述中介绍。

#### 定义：系统状态列表

系统状态列表(SSL)用于描述可编程逻辑控制器的当前状态。SSL 的内容只能通过信息功能进行读取，而不能修改。换言之，部分列表是虚拟列表，只是在有特殊请求时由 CPU 的操作系统所创建。

只能通过 SFC 51 "RDSYSST"读取一个系统状态列表。

#### 目录

系统状态列表包含下列内容的有关信息：

- 系统数据
- CPU 中的模块状态数据
- 关于模块的诊断数据
- 诊断缓冲区

#### 系统数据

系统数据是 CPU 固有的或者已分配的特征数据。它们提供了以下信息：

- CPU 的组态
- 优先级的状态
- 通讯

#### 模块状态数据

模块状态数据描述由系统诊断功能监视组件的当前状态。

#### 关于模块的诊断数据

分配给 CPU 的具有诊断能力的模块的诊断数据直接存储在模块上。

#### 诊断缓冲区

诊断缓冲区包含的诊断条目按出现的顺序排列。。

## 34.2 部分 SSL 列表的结构

### 基础

可以用 SFC 51 "RDSYSST"读取部分列表和部分列表的摘录。可使用参数 SSL\_ID 和 INDEX 指定要读取的内容。

### 结构

部分列表由下列各项组成：

- 一个报头和
- 数据记录。

### 文件头

部分列表的报头由下列各项组成：

- SSL-ID
- 索引
- 部分列表数据记录的长度(以字节计)
- 部分列表中包含的数据记录数目。

### 索引

对于某些部分列表或部分列表摘录，必须指定对象类型 ID 或对象编号。索引即用于此目的。如果不需要该部分信息，则其内容是不相关的。

### 数据记录

部分列表中的数据记录有规定长度。这取决于部分列表中的信息。如何使用数据记录中的数据字也视特定的部分列表而定。



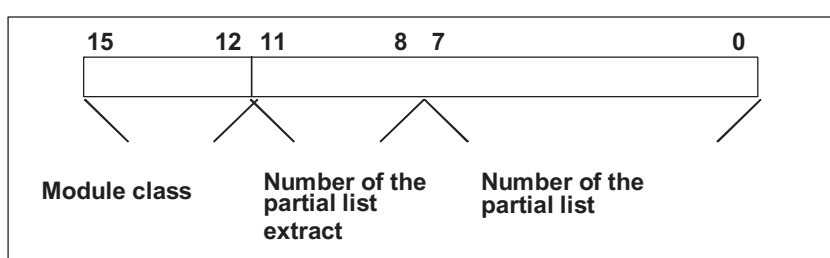
## 34.3 SSL-ID

### SSL-ID

每个部分系统状态列表都有一个编号。可以根据编号输出完整的部分列表或摘录。预定义了可能的部分列表摘录，并由一个数字标识。SSL-ID 由部分列表的编号、部分列表摘录的编号和模块等级组成。

### 结构

SSL-ID 为一个字长。SSL-ID 中的各个位的含义如下所示：



*SCL\_ID* 的结构

### 模块等级

模块等级实例：

模块等级	编码(二进制)
CPU	0000
IM	0100
FM	1000
CP	1100

### 部分列表摘录的编号

部分列表摘录的编号及其含义取决于其所属的特定系统状态列表。通过部分列表摘录的编号，可以指定要读取部分列表的哪一个子集。

### 部分列表的编号

使用部分列表的编号，可以指定要读取系统状态列表的哪一个部分列表。

## 34.4 可能的部分系统状态列表

### 子集

任何一个模块都仅具有所有可能的部分列表的一个子集。哪些部分列表可用取决于特定的模块。

### 可能的 SSL 部分列表

下表列出了所有可能的部分列表及其包含在 SSL-ID 中的编号。

部分列表	SSL-ID
模块标识	W#16#xy11
CPU 特征	W#16#xy12
用户存储区	W#16#xy13
系统区域	W#16#xy14
块类型	W#16#xy15
中断状态	W#16#xy22
过程映像分区和 OB 之间的分配	W#16#xy25
通讯状态数据	W#16#xy32
H CPU 组信息	W#16#xy71
模块 LED 的状态	W#16#xy74
在 H 系统中切换的 DP 从站	W#16#xy75
模块状态信息	W#16#xy91
机架/站的状态信息	W#16#xy92
机架/站的状态信息	W#16#0x94
DP 主站系统/PROFINET IO 系统的扩展信息	W#16#xy95
模块状态信息、PROFINET IO 和 PROFIBUS DP	W#16#xy96
工具变换装置信息(PROFINET IO)	W#16#xy9C
CPU 的诊断缓冲区	W#16#xyA0
模块诊断信息(数据记录 0)	W#16#00B1
模块诊断信息(数据记录 1)、地理地址	W#16#00B2
模块诊断信息(数据记录 1)、逻辑地址	W#16#00B3
DP 从站的诊断数据	W#16#00B4

## 34.5 SSL-ID W#16#xy11 - 模块标识

### 用途

如果读取 SSL-ID 为 W#16#xy11 的系统状态列表，则可获得该模块的模块标识。

### 文件头

系统状态列表 SSL-ID W#16#xy11 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID W#16#0111：单个标识数据记录
INDEX	特定数据记录的编号 W#16#0001：模块的标识 W#16#0004：系统扩展卡的标识 W#16#0006：基本硬件的标识 W#16#0007：基本固件的标识
LENTHDR	W#16#001C：一个数据记录为 14 个字(28 字节)
N_DR	数据记录的数目

### 数据记录

系统状态列表 SSL-ID W#16#xy11 的数据记录具有如下结构：

名称	长度	含义
索引	1 个字	标识数据记录的编号
MifB	20 字节	索引 W#16#0001：模块的订货号；字符串由 19 个字符和一个空格(20H)组成； 例如，对于 CPU 314："6ES7 314-0AE01-0AB0" 索引 W#16#0004：系统扩展卡的订货号；字符串由 19 个字符和一个空格(20H)组成； 例如"6ES7 653-2CE00-0XB0" 索引 W#16#0006：模块的订货号；字符串由 19 个字符和一个空格(20H)组成； 例如，对于 CPU 314："6ES7 314-0AE01-0AB0" 索引 W#16#0007：保留
BGTyp	1 个字	保留
Ausbg1	1 个字	索引 W#16#0001：模块版本 索引 W#16#0004：硬件版本，例如"1" 索引 W#16#0006 和 W#16#0007："V"和版本 ID 的第一个数字
Ausbg2	1 个字	索引 W#16#0001：保留 索引 W#16#0004：固件版本，例如"V1" 索引 W#16#0006 和 W#16#0007：版本 ID 的其余数字

## 34.6 SSL-ID W#16#xy12 - CPU 特征

### 用途

根据所使用的硬件，CPU 模块具有不同的特征。将为每个特征分配一个 ID。如果读取 SSL-ID 为 W#16#xy12 的部分列表，则可获得该模块的特征。

### 文件头

部分列表 SSL-ID W#16#xy12 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID： W#16#0012：所有特征 W#16#0112：组的特征 可在 INDEX 参数中指定组。 W#16#0F12：部分列表报头信息
INDEX	组 W#16#0000：MC7 处理单元 W#16#0100：时间系统 W#16#0200：系统响应 W#16#0300：CPU 的 MC7 语言描述 W#16#0400：SFC 87 和 SFC 88 的可用性
LENTHDR	W#16#0002：一个数据记录为 1 个字长(2 字节)
N_DR	数据记录的数目

### 数据记录

部分列表 SSL-ID W#16#xy12 的数据记录为一个字长。为每个特征都输入了一个标识符。一个特征标识符为一个字长。

---

#### 注意

将输出 CPU 所有相关的数据记录。它们按照次序彼此相连，没有任何间隙。

---

## 特征标识符

下表列出了所有的特征标识符。

标识符	含义
W#16#0000 - 00FF	MC7 处理单元(具有索引 0000 的组)
W#16#0001	MC7 处理生成代码
W#16#0002	MC7 翻译器
W#16#0100 - 01FF	时间系统(具有索引 0100 的组)
W#16#0101	1ms 的分辨率
W#16#0102	10ms 的分辨率
W#16#0103	无实时时钟
W#16#0104	BCD 日时钟格式
W#16#0105	所有日时钟功能(设置日时钟、设置和读取日时钟、日时钟同步：从日时钟和主日时钟)
W#16#0106	SFC 78 "OB_RT"可用
W#16#0200 - 02FF	系统响应(具有索引 0200 的组)
W#16#0201	可使用多处理器模式
W#16#202	可进行冷重启、暖重启和热重启
W#16#203	可进行冷重启和热重启
W#16#204	可进行暖重启和热重启
W#16#205	只能进行暖重启
W#16#0206	通过使用预定义资源可在运行期间进行新的分布式 I/O 组态
W#16#0207	处于单机模式的 H-CPU：通过使用预定义资源可在运行期间进行新的分布式 I/O 组态
W#16#0208	用于考虑运动控制功能
W#16#0300 - 03FF	CPU 的 MC7 语言描述(具有索引 0300 的组)
W#16#0301	保留
W#16#0302	所有 32 位的定点指令
W#16#0303	所有浮点型指令
W#16#0304	sin、asin、cos、acos、tan、atan、sqr、sqrt、ln、exp
W#16#0305	具有相应指令(ENT、PUSH、POP、LEAVE)的累加器 3/累加器 4
W#16#0306	主控继电器指令
W#16#0307	具有相应指令的地址寄存器 1 已存在
W#16#0308	具有相应指令的地址寄存器 2 已存在
W#16#0309	区域交叉寻址操作
W#16#030A	区域内寻址操作
W#16#030B	位存储器(M)的所有存储器间接寻址指令
W#16#030C	数据块(DB)的所有存储器间接寻址指令
W#16#030D	数据块(DI)的所有存储器间接寻址指令
W#16#030E	本地数据(L)的所有存储器间接寻址指令
W#16#030F	FC 中用于参数传送的所有指令
W#16#0310	过程映像输入(I)的存储器位边沿指令
W#16#0311	过程映像输出(Q)的存储器位边沿指令
W#16#0312	位存储器(M)的存储器位边沿指令

标识符	含义
W#16#0313	数据块(DB)的存储器位边沿指令
W#16#0314	数据块(DI)的存储器位边沿指令
W#16#0315	本地数据(L)的存储器位边沿指令
W#16#0316	FC 位的动态判断
W#16#0317	具有相应指令的动态本地数据区
W#16#0318	保留
W#16#0319	保留
W#16#0401	SFC 87 "C_DIAG"可用
W#16#0402	SFC 88 "C_CNTRL"可用

## 34.7 SSL-ID W#16#xy13 - 存储器区

### 用途

如果读取 SSL-ID 为 W#16#xy13 的部分列表，则可获得有关该模块存储区的信息。

### 文件头

部分列表 SSL-ID W#16#xy13 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID W#16#0113：一个存储区的数据记录 可使用 INDEX 参数指定存储区。
INDEX	指定一个存储区(仅针对 SSL-ID W#16#0113) W#16#0001：工作存储器
LENTHDR	W#16#0024：一个数据记录为 18 个字长(36 字节)
N_DR	数据记录的数目

### 数据记录

部分列表 SSL-ID W#16#xy13 的数据记录具有如下结构：

名称	长度	含义
索引	1 个字	存储区域的索引 W#16#0001：工作存储器
代码	1 个字	存储器类型： W#16#0001：易失性存储器(RAM) W#16#0002：非易失性存储器(FEPROM) W#16#0003：混合性存储器(RAM + FEPROM)
大小	2 个字	所选存储器的总容量(区域 1 和区域 2 的合计容量)
模式	1 个字	存储器的逻辑模式 位 0：易失性存储区 位 1：非易失性存储区 位 2：混合性存储区 对于工作存储器： 位 3：代码和数据分开 位 4：代码和数据在一起
Granu	1 个字	值始终为 0
Ber1	2 个字	易失性存储区的大小(以字节计)。
Belegt1	2 个字	正在使用的易失性存储区的大小
Block1	2 个字	易失性存储区中的最大空闲块 如果为 0：无信息可用或者无法确定。
Ber2	2 个字	非易失性存储区的大小(以字节计)
Belegt2	2 个字	正在使用的非易失性存储区的大小
Block2	2 个字	非易失性存储区中的最大空闲块 如果为 0：无信息可用或者无法确定。

## 34.8 SSL-ID W#16#xy14 - 系统区域

### 用途

如果读取 SSL-ID 为 W#16#xy14 的部分列表，则可获得有关该模块的系统区域信息。

### 文件头

部分列表 SSL-ID W#16#xy14 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID W#16#0014：模块的所有系统区域 W#16#0F14：仅用于部分列表报头信息
INDEX	不相关
LENTHDR	W#16#0008：一个数据记录为 4 个字长(8 字节)
N_DR	数据记录的数目 分配的数据记录数目必须至少为 9 个。如果选择的目标区域过小，则 SFC51 将不会提供数据记录。

### 数据记录

部分列表 SSL-ID W#16#xy14 的数据记录的结构如下：

名称	长度	含义
索引	1 个字	系统区域的索引 W#16#0001：PII (以字节计) W#16#0002：PIQ (以字节计) W#16#0003：存储器(以字节计) <b>注意：</b> 此索引仅由 CPU 提供，其中，标记数可在一个字内显示。 如果您的 CPU 不提供此值，则必须估计索引 W#16#0008 的值。  W#16#0004：定时器(数量) W#16#0005：计数器(数量) W#16#0006：逻辑地址区域中的字节数 W#16#0007：本地数据(CPU 的整个本地数据区(以字节计)) <b>注意：</b> 此索引仅由 CPU 提供，其中，标记数可在一个字内显示。 如果您的 CPU 不提供此值，则必须估计索引 W#16#0009 的值。  W#16#0008：存储器(以字节计) W#16#0009：本地数据(CPU 的整个本地数据区(以千字节计))



名称	长度	含义
代码	1 个字	存储器类型 W#16#0001 : 易失性存储器(RAM) W#16#0002 : 非易失性存储器(FEPROM) W#16#0003 : 混合性存储器(RAM 和 FEPROM)
数量	1 个字	系统区域的元素数
Reman	1 个字	掉电保护元素数

## 34.9 SSL-ID W#16#xy15 - 块类型

### 用途

如果读取 SSL-ID 为 W#16#xy15 的部分列表，则可获得该模块上存在的块类型。

### 文件头

部分列表 SSL-ID W#16#xy15 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID W#16#0015：模块的所有块类型的数据记录
INDEX	不相关
LENTHDR	W#16#0006：一个数据记录为 5 个字长(10 字节)
N_DR	数据记录的数目

### 数据记录

部分列表 SSL-ID W#16#xy15 的数据记录的结构如下：

名称	长度	含义
索引	1 个字	块类型编号 W#16#0800：OB W#16#0A00：DB W#16#0B00：SDB W#16#0C00：FC W#16#0E00：FB
MaxAnz	1 个字	以下类型的块的最大数目 OB：一个 CPU 可能具有的最大 OB 数 DB：DB 可能的最大数目(包括 DB0) SDB：SDB 可能的最大数目(包括 SDB2) FC 和 FB：可装载块可能的最大数目
MaxLng	1 个字	待装载对象的总量最大值(以千字节计)
Maxabl	2 个字	一个块的工作存储器部分的最大长度(以字节计)

## 34.10 SSL-ID W#16#xy74 - 模块 LED 的状态

### 用途

对于标准 CPU (如存在)和 H CPU 读取部分列表 SSL-ID W#16#xy74，将获得模块 LED 的状态。

如果 H CPU 处于非冗余 H 模式，将获得被寻址的 CPU 的 LED 状态。如果 H CPU 处于 RUN-REDUNDANT 模式，则返回所有冗余的 H CPU 的 LED 状态。

### 文件头

部分列表 SSL-ID W#16#xy74 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID W#16#0174 一个 LED 的状态。 可使用 INDEX 参数选择 LED。
INDEX	LED ID (只与 SSL-ID W#16#0174 有关) W#16#0001 : SF (组错误) W#16#0002 : INTF (内部错误) W#16#0003 : EXTf (外部错误) W#16#0004 : RUN W#16#0005 : STOP W#16#0006 : FRCE (压力) W#16#0007 : CRST (冷启动) W#16#0008 : BAF (总线上的电池故障/超载和电池电压短路) W#16#0009 : USR (用户自定义) W#16#000A : USR1 (用户自定义) W#16#000B : BUS1F (总线错误接口 1) W#16#000C : BUS2F (总线错误接口 2) 或 BUS5F (总线错误接口 5)，带 CPU 414-3 PN/DP、 416-3 PN/DP 和 416F-3 PN/DP

目录	含义
INDEX	W#16#000D : REDF (冗余错误) W#16#000E : MSTR (主站) W#16#000F : RACK0 (机架号 0) W#16#0010 : RACK1 (机架号 1) W#16#0011 : RACK2 (机架号 2) W#16#0012 : IFM1F (接口错误接口模块 1) W#16#0013 : IFM2F (接口错误接口模块 2) W#16#0014 : BUS3F (总线错误接口 3) W#16#0015 : MAINT (维护要求) W#16#0016 : DC24V W#16#0017 : BUS5F (总线错误接口 5) W#16#0018 : BUS8F (总线错误接口 8) : W#16#0080 : IF (初始化失败) W#16#0081 : UF (用户故障) W#16#0082 : MF (监视故障) W#16#0083 : CF (通信故障) W#16#0084 : TF (任务失败) : W#16#00EC : APPL_STATE_RED W#16#00ED : APPL_STATE_GREEN
ENTHDR	W#16#0004 : 一个数据记录为 2 个字长(4 字节)
N_DR	数据记录的数目

### 数据记录

H CPU 冗余运行时,为每个 CPU 返回一个数据记录:第一个数据记录包含当前主 CPU 的 LED 信息,第二个数据记录包含当前备用 CPU 的 LED 信息。

部分列表摘录 SSL-ID W#16#xy74 的数据记录具有如下结构:

名称	长度	含义
cpu_led_ID	1 个字	<ul style="list-style-type: none"> <li>• 字节 0                             <ul style="list-style-type: none"> <li>- 标准 CPU : B#16#00</li> <li>- H-CPU : 位 0 到 2 : 机架号</li> <li>位 3 : 0 = 待机 CPU , 1 = 主站 CPU</li> <li>位 4 到 7 : 1111</li> </ul> </li> </ul> 字节 1 : LED 标识符
led_on	1 字节	LED 的状态 : 0 : 关闭 1 : 打开
led_blink	1 字节	LED 的闪烁状态 : 0 : 不闪烁 1 : 正常闪烁(2 赫兹) 2 : 缓慢闪烁(0.5 赫兹)

## 34.11 SSL-ID W#16#xy1C - 组件标识

### 用途

如果读取 SSL-ID 为 W#16#xy1C 的部分列表，则可识别 CPU 或 PLC。

### 文件头

部分列表 W#16#xy1C 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID
	W#16#001C : 所有组件的标识
	W#16#011C : 一个组件的标识
	W#16#021C : H 系统中一个 CPU 的所有组件的标识
	W#16#031C : H 系统中所有冗余 CPU 的一个组件的标识
	W#16#0F1C : 仅限 SSL 部分列表报头信息
INDEX	<ul style="list-style-type: none"> <li>• SSL-ID 为 W#16#011C 和 W#16#031C 的部分列表摘录的组件标识               <ul style="list-style-type: none"> <li>- W#16#0001 : 自动化系统的名称</li> <li>- W#16#0002 : 模块名称</li> <li>- W#16#0003 : 模块的设备标识</li> <li>- W#16#0004 : 版权条目</li> <li>- W#16#0005 : 模块的序列号</li> <li>- W#16#0007 : 模块类型名称</li> <li>- W#16#0008 : 存储卡的序列号</li> <li>对于不能插入存储卡的模块，不返回任何数据记录。</li> <li>- W#16#0009 : CPU 模块的制造商和配置文件</li> <li>- W#16#000A : 模块的 OEM ID (仅限 S7-300)</li> <li>- W#16#000B : 模块的位置 ID</li> <li>- W#16#000C : 同步模块 1 的序列号</li> <li>- W#16#000D : 同步模块 2 的序列号</li> <li>- W#16#000E : 系统扩展卡的序列号</li> </ul> </li> <li>• SSL-ID 为 W#16#021C 的部分列表摘录的机架号(字节 0 : 机架号，字节 1 : B#16#00)</li> </ul>
LENTHDR	W#16#0022 : 一个数据记录为 17 个字长(34 字节)
N_DR	数据记录的数目

## 返回的数据

在包含 SSL-ID W#16#011C 和 W#16#031C 的部分列表中，返回的数据来自仅仅一个 INDEX 的一条或多条数据记录。

在包含 SSL-ID W#16#001C 和 W#16#021C 的部分列表中，返回的数据来自所有可能的数据记录。

### 注意

如果通过 SSL-ID W#16#001C 和 W#16#021C 调用 SFC 51，INDEX 调用参数具有与其它返回的数据记录不同的含义。

对于 SSL-ID W#16#001C，INDEX 调用参数无关，而对于 SSL-ID W#16#021C，可以在 INDEX 调用参数中传送机架号(0 或 1)。

## 数据记录

SSL-ID 为 W#16#xy1C 的部分列表的数据记录具有如下结构：

- INDEX = W#16#0001

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C：组件 ID：W#16#0001</li> <li>• 对于 H CPU 及部分列表摘录 W#16#021C 与 W#16#031C： 字节 0：0 到 2 位：机架号；位 3：0 = 保留 CPU，1 = 主站 CPU； 4 到 7 位：1111 字节 1：组件 ID：B#16#01</li> </ul>
名称	12 个字	PLC 的名称 (最多 24 个字符；当使用短名称时，则用 B#16#00 填补间隙)
Res	4 个字	保留

- INDEX = W#16#0002

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C：组件 ID：W#16#0002</li> <li>• 对于 H CPU 及部分列表摘录 W#16#021C 与 W#16#031C： 字节 0：0 到 2 位：机架号；位 3：0 = 保留 CPU，1 = 主站 CPU； 4 到 7 位：1111 字节 1：组件 ID：B#16#02</li> </ul>
名称	12 个字	模块名称 (最多 24 个字符；当使用短名称时，则用 B#16#00 填补间隙)
Res	4 个字	保留

- INDEX = W#16#0003

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C : 组件 ID : W#16#0003</li> <li>• 对于 H CPU 及部分列表摘录 W#16#021C 与 W#16#031C : 字节 0 : 0 到 2 位 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111 字节 1 : 组件 ID : B#16#03</li> </ul>
Tag	16 个字	模块的设备标识 (最多 32 个字符 ; 当使用短设备标识时 , 则用 B#16#00 填补间隙)

- INDEX = W#16#0004

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C : 组件 ID : W#16#0004</li> <li>• 对于 H CPU 及部分列表摘录 W#16#021C 与 W#16#031C : 字节 0 : 0 到 2 位 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111 字节 1 : 组件 ID : B#16#04</li> </ul>
版权	13 个字	连续的字符序列 "西门子原厂设备"
Res	3 字	保留

- INDEX = W#16#0005

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C : 组件 ID : W#16#0005</li> <li>• 对于 H CPU 及部分列表摘录 W#16#021C 与 W#16#031C : 字节 0 : 0 到 2 位 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111 字节 1 : 组件 ID : B#16#05</li> </ul>
serialn	12 个字	模块的序列号 ; 最大长度为 24 个字符的字符串。短编号用 B#16#00 填补。 注意 : 对于 SIMATIC 组件 , 此序列号全球唯一且与 CPU 硬件永久相关 , 即该序列号在执行固件更新后保持不变。
res	4 个字	保留

- INDEX = W#16#0007

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C : 组件 ID : W#16#0007</li> <li>• 对于 H CPU 及部分列表摘录 W#16#021C 与 W#16#031C : 字节 0 : 0 到 2 位 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111 字节 1 : 组件 ID : B#16#07</li> </ul>
Cputypname	16 个字	模块类型名称 ; 最大长度为 32 个字符的字符串。短名称用 B#16#00 填补。

- INDEX = W#16#0008

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C : 组件 ID : W#16#0008</li> <li>• 对于 H CPU 及部分列表摘录 W#16#011C、W#16#021C 与 W#16#031C : 字节 0 : 0 到 2 位 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111 字节 1 : 组件 ID : B#16#08</li> </ul>
sn_mc/mmc	16 个字	存储卡/微存储卡的序列号 ; 最大长度为 32 个字符的字符串。短名称用 B#16#00 填补。 <ul style="list-style-type: none"> <li>• 西门子序列号 : 仅序列号 , 无索引</li> <li>• S7 微存储卡的产品序列号(PSN) : "MMC"加序列号(PSN)</li> <li>• S7 存储卡的序列号 : "MC"加序列号</li> </ul> 如果未安装任何存储卡 , 则字符串在"MMC"或"MC"后即结束。

- INDEX = W#16#0009

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C : 组件 ID : W#16#0009</li> <li>• 对于 H CPU 及部分列表摘录 W#16#021C 与 W#16#031C : 字节 0 : 0 到 2 位 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111 字节 1 : 组件 ID : B#16#09</li> </ul>
manufacturer_id	1 个字	请参见"PROFIBUS 配置文件指南 , 第 1 部分 - 标识与维护功能"
profile_id	1 个字	请参见"PROFIBUS 配置文件指南 , 第 1 部分 - 标识与维护功能"
profile_specific_typ	1 个字	请参见"PROFIBUS 配置文件指南 , 第 1 部分 - 标识与维护功能"
res	13 个字	保留



- INDEX = W#16#000A

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C : 组件 ID : W#16#000A</li> <li>• 对于 H CPU 及部分列表摘录 W#16#021C 与 W#16#031C : 字节 0 : 0 到 2 位 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111 字节 1 : 组件 ID : B#16#0A</li> </ul>
oem_copyright_string	13 个字	OEM 版权 ID ; 最大长度为 20 个字符的字符串。短名称用 B#16#00 填补。
oem_id	1 个字	OEM ID ; 由西门子签发。
oem_add_id	2 个字	OEM 附加 ID ; 可由用户签发。

- INDEX = W#16#000B

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 对于一个标准 CPU 和部分列表摘录 W#16#011C : 组件 ID : W#16#000B</li> <li>• 对于 H CPU 及部分列表摘录 W#16#021C 与 W#16#031C : 字节 0 : 0 到 2 位 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111 字节 1 : 组件 ID : B#16#0B</li> </ul>
loc_id	16 个字	位置标识 ; 最大长度为 32 个字符的字符串。短名称用 B#16#00 填补。

- INDEX = W#16#000C

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 字节 0 : 位 0 到 2 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111</li> <li>• 字节 1 : 组件的标识符 : B#16#0C</li> </ul>
mlfb	5 字	10. 同步模块 1 的订货号的 10-19 个字符
注入	1 个字	四个空格(ASCII)
es	1 个字	产品版本(ASCII), 例如, "01"
注入	1 字节	两个空格(ASCII)
vendor_sn	17 个字节	序列号

**注意：**

如果同步模块 1 不存在或无法识别，则在变量"索引"后用 B#16#00 填入记录。

- INDEX = W#16#000D

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 字节 0 : 位 0 到 2 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111</li> <li>• 字节 1 : 组件的标识符 : B#16#0D</li> </ul>
mifb	5 字	10. 同步模块 2 的订货号的 10-19 个字符
注入	1 个字	四个空格(ASCII)
es	1 个字	产品版本(ASCII), 例如, "01"
注入	1 字节	两个空格(ASCII)
vendor_sn	17 个字节	序列号

**注意 :**

如果同步模块 2 不存在或无法识别, 则在变量"索引"后用 B#16#00 填入记录。

- INDEX = W#16#000E

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>• 在非冗余设置和部分列表摘录 W#16#011C 中 : 组件 ID : W#16#000E</li> <li>在冗余设置和部分列表摘录 W#16#021C 和 W#16#031C 中 : 字节 0 : 0 到 2 位 : 机架号 ; 位 3 : 0 = 保留 CPU , 1 = 主站 CPU ; 4 到 7 位 : 1111 字节 1 : 组件 ID : B#16#0E</li> </ul>
serialn_sec	9 字	系统扩展卡的序列号 : "SECSVP" + 12 个 ASCII 字符
注入	7 个字	14 乘以 B#16#00

## 34.12 SSL-ID W#16#xy22 - 中断状态

### 用途

SSL ID 为 W#16#xy22 的部分列表包含有关模块中中断处理和中断生成的当前状态信息。

### 文件头

SSL-ID 为 W#16#xy22 的部分列表的报头具有下列结构：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID： W#16#0222 有关指定中断的数据记录。该中断 (OB 号)将在 INDEX 参数中指定。
INDEX	中断等级或 OB 编号(对于 SSL-ID W#16#0222) W#16#0000 : 空闲周期 W#16#000A : 时间中断 W#16#0014 : 延时中断 W#16#001E : 循环中断 W#16#0028 : 硬件中断 W#16#0032 : DP 中断 W#16#003C : 多值计算或同步循环(等时线)中断 W#16#0048 : 冗余中断(仅针对 S7-400H 系统) W#16#0050 : 异步错误中断 W#16#005A : 背景 W#16#0064 : 启动 W#16#0078 : 同步错误中断
LENTHDR	W#16#001C : 一个数据记录为 14 个字长(28 字节)
N_DR	数据记录的数目

数据记录

部分列表摘录 SSL-ID W#16#xy22 的数据记录具有下列结构：

名称	长度	含义
信息	10 个字	<p>给定 OB 的启动信息，下列情况例外：</p> <ul style="list-style-type: none"> <li>OB 1 提供当前最小(字节 8 和 9)和最大周期(字节 10 和 11)(时间基准：ms，字节计数从 0 开始)。</li> <li>在一个延时中断的作业被激活后，字节 8 和 11(字节计数从 0 开始)会将剩余时间(以 ms 为单位的剩余延迟时间)设置为一个参数。</li> <li>OB 80 包含已组态的最小(字节 8 和 9)和最大周期(字节 10 和 11)(时间基准：ms，字节计数从 0 开始)。</li> <li>无当前信息的错误中断</li> <li>中断将包含来自中断源的当前参数设置的状态信息。</li> <li>在出现同步错误的情况下，如果尚未处理 OB，则输入的优先级为 B#16#7F；否则，为最后一次调用的优先级。</li> <li>如果一个 OB 有多个启动事件，且在信息时间这些事件尚未发生，则将返回事件编号 W#16#xyzz，其中 x：事件等级，zz：组中最小的已定义编号，y：未定义。否则，将使用已出现的上一启动事件的编号。</li> </ul>
al 1	1 个字	<p>处理标识符：</p> <p>位 0： 中断事件是由参数引起的 =0： 启用 =1： 禁用</p> <p>位 1： 中断事件(根据 SFC 39 "DIS_IRT") =0： 未锁定 =1： 锁定</p> <p>位 2 = 1： 中断源已激活(生成作业已就绪，可以启动时间中断，时间 OB 已启动，延时中断 OB 已启动，已通过 STEP 7 组态循环中断 OB)</p> <p>位 4： 中断 OB =0： 未装载 =1： 已装载</p> <p>位 5： 由 TIS 中断 OB =1： 锁定</p> <p>位 6： 诊断缓冲区中的条目 =1： 锁定</p>
al 2	1 个字	<p>未装载/锁定的 OB 的反应</p> <p>位 0 = 1： 锁定中断源</p> <p>位 1 = 1： 生成中断事件错误</p> <p>位 2 = 1： CPU 进入 STOP 模式</p> <p>位 3 = 1： 仅放弃中断</p>
al 3	2 个字	<p>由 TIS 功能放弃：</p> <p>置位位号 x 意味着：比受影响的 OB 的最小事件编号大 x 的事件编号可以由 TIS 功能放弃。</p>

## 34.13 SSL ID W#16#xy25 - 将过程映像分区分配到 OB

### 用途

SSL ID 为 W#16#xy25 的部分列表给出了将过程映像分区分配到 OB 的方法。

此列表提供有关下列内容的信息

- 已分配给指定 OB 的过程映像分区通过系统进行更新
- 已分配给指定时钟同步中断 OB(OB 61 到 64)的过程映像分区通过调用 SFC 126 "SYNC\_PI"和 127 "SYNC\_PO"更新。  
DP 主站系统和时钟同步中断 OB 之间的分配可在 SSL W#16#xy95 中找到。

### 文件头

SSL ID 为 W#16#xy25 的部分列表的报头具有如下结构：

目录	含义
SSL ID	部分列表摘录的 SSL ID <ul style="list-style-type: none"> <li>• W#16#0025：将过程映像分区分配给 CPU 中的 OB</li> <li>• W#16#0125：将过程映像分区分配给相应的 OB 在参数 INDEX 中指定过程映像分区 ID。</li> <li>• W#16#0225：将 OB 分配给过程映像分区 在参数 INDEX 中指定 OB 编号。 注意：时钟同步中断 OB (OB 61 到 64)是可以分配给多个过程映像分区的唯一 OB。</li> <li>• W#16#0F25：仅限 SSL 部分列表报头的信息</li> </ul>
INDEX	<ul style="list-style-type: none"> <li>• 对于 SSL ID W#16#0025：不相关</li> <li>• 对于 SSL ID W#16#0125：过程映像分区 ID</li> <li>• 对于 SSL ID W#16#0225：OB 编号。</li> <li>• 对于 SSL ID W#16#0F25：不相关</li> </ul>
LENTHDR	W#16#0004：一个数据记录为 2 个字长(4 字节)
N_DR	数据记录的数目

## 数据记录

部分列表摘录 SSL-ID W#16#xy25 的数据记录具有如下结构：

名称	长度	含义
tpa_nr	1 字节	部分过程映像 ID
tpa_use	1 字节	过程映像分区和 OB 之间的分配类型： <ul style="list-style-type: none"> <li>• 位 0 = 1：输入的过程映像分区被分配给指定的 OB，并由系统进行更新。</li> <li>• 位 1 = 1：输出的过程映像分区被分配给指定的 OB，并由系统进行更新。</li> <li>• 位 2 = 1：输入的过程映像分区将被分配给指定的时钟同步中断 OB。可通过调用 SFC 126 "SYNC_PI"在此 OB 中更新过程映像分区。</li> <li>• 位 3 = 1：输出的过程映像分区被分配给指定的时钟同步中断 OB。可通过调用 SFC 126 "SYNC_PO"在此 OB 中更新过程映像分区。</li> <li>• 4 到 7 位：0</li> </ul>
ob_nr	1 字节	OB 编号
res	1 字节	保留

## 部分列表摘录

- SSL ID = W#16#0025 的部分列表摘录：  
在组态中已分配给一个 OB 的所有过程映像分区的数据记录将会按升序返回。对于未分配给任何 OB 的过程映像分区，此 ob\_nr 参数值为零。这种情况下，将不会为过程映像分区返回数据记录。
- SSL ID = W#16#0125 的部分列表摘录：  
如果在组态中已将寻址的过程映像分区分配给一个 OB，则会返回数据记录。如果尚未分配 OB，则不会返回任何数据记录。

### 注意

OB 1 被永久分配给过程映像分区 0。因此，请求有关过程映像分区 0 的信息时，总会接收到数据记录。

- SSL ID = W#16#0225 的部分列表摘录：  
对于已分配给调用 OB 的所有过程映像分区将返回一个数据记录。如果尚未将过程映像分区分配给调用 OB，则不会返回任何数据记录。

### 注意

可以将多个过程映像分区分配给时钟同步中断 OB。在这种情况下，会返回多个数据记录。

- SSL ID = W#16#0F25 的部分列表摘录：  
返回的数据记录的最大数目。

## 有关数据记录含义的实例

SFC 51 的调用参数	返回的变量	解释
SZL_ID = W#16#0125 , INDEX = W#16#0008	tpa_nr = B#16#08 , tpa_use = B#16#03 , ob_nr = B#16#15	返回一个数据记录。 输入/输出过程映像 8 被分配给 OB 21 , 并由系统更新该过程映像。
SZL_ID = W#16#0125 , INDEX = W#16#0009	-	未返回任何数据记录。 因此 : 过程映像分区 9 未被分配给任何 OB。
SZL_ID = W#16#0225 , INDEX = W#16#003D	tpa_nr = B#16#0A , tpa_use = B#16#C0 , ob_nr = B#16#3D  tpa_nr = B#16#10 , tpa_use = B#16#C0 , ob_nr = B#16#3D	返回两个数据记录。 将输入和输出的过程映像 10 和 16 分配给 OB 61。可通过调用 SFC 126 和 127 在 OB 61 中将其更新。
SZL_ID = W#16#0225 , INDEX = W#16#0001	tpa_nr = B#16#00 , tpa_use = B#16#03 , ob_nr = B#16#01	返回一个数据记录。 过程映像 0 分配给 OB 1 , 用于输入和输出。它们将由系统进行更新。

## 34.14 SSL-ID W#16#xy32 - 通讯状态数据

### 用途

如果读取 SSL-ID 为 W#16#xy32 的部分列表，则可获得模块通讯的状态数据。

### 文件头

部分列表 SSL-ID W#16#xy32 的报头结构如下所示：

目录	含义										
SSL-ID	部分列表摘录的 SSL ID <ul style="list-style-type: none"> <li>W#16#0132：CPU 的一个通讯部分的状态数据(始终为一个数据记录)。可以用 INDEX 参数指定 CPU 的通讯部分。</li> <li>W#16#0232：一个通讯部分的状态数据(在一个处于 RUN-REDUNDANT 模式、单独模式或单机操作的 H 系统中,将返回 2 个数据记录。)。可以用 INDEX 参数指定 CPU 的通讯部分。</li> </ul>										
INDEX	通讯部分 <ul style="list-style-type: none"> <li>对于 SSL ID W#16#0132：               <table style="margin-left: 20px;"> <tr> <td>W#16#0005</td> <td>诊断</td> </tr> <tr> <td>W#16#0008</td> <td>时间系统</td> </tr> <tr> <td>W#16#000B</td> <td>时间系统</td> </tr> <tr> <td>W#16#000C</td> <td>时间系统</td> </tr> </table> </li> <li>对于 SSL ID W#16#0232：               <table style="margin-left: 20px;"> <tr> <td>W#16#0004</td> <td>CPU保护等级、操作员控制设置和版本ID/检验和</td> </tr> </table> </li> </ul>	W#16#0005	诊断	W#16#0008	时间系统	W#16#000B	时间系统	W#16#000C	时间系统	W#16#0004	CPU保护等级、操作员控制设置和版本ID/检验和
W#16#0005	诊断										
W#16#0008	时间系统										
W#16#000B	时间系统										
W#16#000C	时间系统										
W#16#0004	CPU保护等级、操作员控制设置和版本ID/检验和										
LENTHDR	W#16#0028：一个数据记录为 20 个字长(40 字节)										
N_DR	数据记录的数目										

### 数据记录

部分列表 SSL-ID W#16#0132 的数据记录始终为 20 个字长。数据记录具有不同的内容。具体内容取决于 INDEX 参数，换言之，取决于数据记录所属的 CPU 的通讯部分。



## 34.15 SSL-ID 为 W#16#0132、索引为 W#16#0005 的部分列表摘录的数据记录

### 目录

SSL ID 为 W#16#0132、索引为 W#16#0005 的部分列表摘录包含有关模块上的诊断状态的信息。

### 数据记录

SSL-ID 为 W#16#0132、索引为 W#16#0005 的部分列表摘录的数据记录具有如下结构：

名称	长度	含义
索引	1 个字	W#16#0005：诊断
Erw	1 个字	扩展功能 0：否 1：是
发送	1 个字	自动发送 0：否 1：是
Moeg	1 个字	当前可以发送用户自定义的诊断消息 0：否 1：是
Res	16 个字	保留

## 34.16 SSL-ID 为 W#16#0132、索引为 W#16#0008 的部分列表摘录的数据记录

### 目录

SSL-ID 为 W#16#0132、索引为 W#16#0008 的部分列表摘录包含有关模块上的时间系统状态的信息。

### 数据记录

SSL-ID 为 W#16#01032、索引 W#16#0008 的部分列表摘录的数据记录具有如下结构：

名称	长度	含义
索引	1 个字	W#16#0008：时间系统状态
Zykl	1 个字	同步帧的周期
Korr	1 个字	时间的修正因子
时钟 0	1 个字	运行系统计时器 0：以小时计
时钟 1	1 个字	运行系统计时器 1：以小时计
时钟 2	1 个字	运行系统计时器 2：以小时计
时钟 3	1 个字	运行系统计时器 3：以小时计
时钟 4	1 个字	运行系统计时器 4：以小时计
时钟 5	1 个字	运行系统计时器 5：以小时计
时钟 6	1 个字	运行系统计时器 6：以小时计
时钟 7	1 个字	运行系统计时器 7：以小时计
时间	4 个字	当前日期和时间 (格式：DATE_AND_TIME)
bszl_0 到 bszl_1	2 个字节	运行系统计时器已激活(位 = 1：运行系统计时器已激活)
bszl_0	1 字节	位 x：运行时间定时器 x， $0 < x < 7$
bszl_1	1 字节	保留
bszü_0 到 bszü_1	2 个字节	运行系统计时器溢出(位 = 1：溢出)
bszü_0	1 字节	位 x：运行时间定时器 x， $0 < x < 7$
bszü_1	1 字节	保留
状态	1 个字	时间状态(用于位分配，参见下文)
Res	3 字节	保留
status_valid	1 字节	变量状态的有效性： B#16#01：状态有效

## 状态

位	缺省值	描述
15	0	修正值的符号 (0：正号，1：负号)
14 到 10	00000	修正值 此参数允许将帧中的基本时间修正为当地时间： 当地时间 = 基本时间 修正值 * 0.5 h 此修正考虑时区及夏令时和冬令时(标准时间)的时差。
9	0	保留
8	0	保留
7	0	通知时间 此参数指示下一时间调整是否也会包括从夏令时到冬令时(标准时间)的转换(反之亦然)。 (0: 未进行任何调整，1：进行了调整)。
6	0	夏令时/冬令时(标准时间)指示符此参数指示使用修正值计算的当地时间是夏令时还是冬令时。 (0: 冬令时，1：夏令时)
5	0	参数未被 S7 使用。
4 到 3	00	时间分辨率 此参数指示传送的时钟时间的分辨率。 (00: 0.001 秒，01：0.01 秒，10：0.1 秒，11：1 秒)
2	0	参数未被 S7 使用。
1	0	参数未被 S7 使用。
0	0	同步失败 此参数指示来自外部时间主站(例如 SICLOCK)的帧中传送的时间是否同步 (0：同步失败，1：已同步) 注意： 仅当存在持续的外部时间同步时，对 CPU 中此位的判断才有意义。

## 34.17 SSL-ID 为 W#16#0132、索引为 W#16#000B 的部分列表摘录的数据记录

### 目录

SSL ID 为 W#16#0132、索引为 W#16#000B 的部分列表摘录包含有关模块的 32 位运行系统计时器 0 ... 7 的状态的信息。

#### 注意

SSL ID 为 W#16#0132、索引为 W#16#0008 的部分列表摘录将以 16 位模式显示运行系统计时器。这将允许用户使用为 CPU 开发的程序 这些程序与 16 位的运行系统计时器和 SSL ID 为 W#16#0132、索引为 W#16#0008 的部分列表摘录一起运行。

### 数据记录

SSL ID 为 W#16#0132、索引为 W#16#000B 的部分列表摘录的数据记录具有如下结构：

名称	长度	含义
索引	1 个字	W#16#000B：时间系统状态
bszl_0	1 字节	位 x：运行系统计时器 x 的状态， $0 \leq x \leq 7$ (位 = 1：运行时间定时器忙)
bszl_1	1 字节	保留
bszü_0	1 字节	位 x：运行系统计时器 x 的溢出， $0 \leq x \leq 7$ (位 = 1：溢出)
bszü_1	1 字节	保留
时钟 0	2 个字	运行系统计时器 0：以小时计
时钟 1	2 个字	运行系统计时器 1：以小时计
时钟 2	2 个字	运行系统计时器 2：以小时计
时钟 3	2 个字	运行系统计时器 3：以小时计
时钟 4	2 个字	运行系统计时器 4：以小时计
时钟 5	2 个字	运行系统计时器 5：以小时计
时钟 6	2 个字	运行系统计时器 6：以小时计
时钟 7	2 个字	运行系统计时器 7：以小时计
Res	1 个字	保留

## 34.18 具有 SSL ID W#16#0132 索引 W#16#000C 的部分列表摘录的数据记录

### 目录

SSL ID 为 W#16#0132、索引为 W#16#000C 的部分列表摘录将包含有关模块的 32 位运行系统计时器 8 ... 15 的状态信息。

### 数据记录

SSL ID 为 W#16#0132、索引为 W#16#000C 的部分列表摘录的数据记录具有下列结构：

名称	长度	含义
索引	1 个字	W#16#000C：时间系统状态
Bszl_0	1 字节	位 x：运行系统计时器(8+x)的状态， $0 \leq x \leq 7$ (位 = 1：运行时间定时器忙)
Bszl_1	1 字节	保留
Bszü_0	1 字节	位 x：运行系统计时器(8+x)的溢出， $0 \leq x \leq 7$ (位 = 1：溢出)
Bszü_1	1 字节	保留
时钟 8	2 个字	运行系统计时器 8：以小时计
时钟 9	2 个字	运行系统计时器 9：以小时计
时钟 10	2 个字	运行系统计时器 10：以小时计
时钟 11	2 个字	运行系统计时器 11：以小时计
时钟 12	2 个字	运行系统计时器 12：以小时计
时钟 13	2 个字	运行系统计时器 13：以小时计
时钟 14	2 个字	运行系统计时器 14：以小时计
时钟 15	2 个字	运行系统计时器 15：以小时计
Res	1 个字	保留

## 34.19 SSL-ID 为 W#16#0232、索引为 W#16#0004 的部分列表摘录的数据记录

### 目录

SSL-ID W#16#0232 和索引 W#16#0004 的部分列表摘录包含有关下列内容的信息：CPU 保护等级、操作员模式切换设置、硬件配置和用户程序的检验和设置。

#### 注意

在处于 RUN-REDUNDANT 模式的 H 系统中，将返回两个数据记录。其中只有"索引"字第 0 字节上的值不为零的数据记录有效。

在 solo 模式或单机操作的 H 系统中，将返回 1 个数据记录。

### 数据记录

SSL-ID 为 W#16#0232、索引为 W#16#0004 的部分列表摘录的数据记录具有如下结构：

名称	长度	含义
索引	1 个字	<ul style="list-style-type: none"> <li>字节 1： B#16#04：CPU 保护等级、操作员控制设置和版本标识/检验</li> <li>字节 0： 标准 CPU：B#16#00 H CPU：位 0 到 2：机架号 位 3：0 = 待机 CPU，1 = 主站 CPU 位 4 到 7：1111</li> </ul>
sch_schal	1 个字	以模式选择器(1、2、3)设置的保护级别
sch_par	1 个字	以参数设置的保护等级(0、1、2、3；0：无密码，保护级别无效)
sch_rel	1 个字	CPU 的有效保护级别
bart_sch	1 个字	模式选择器设置(1：RUN，2：RUN-P，3：STOP，4：MRES，0：未定义或无法确定)
anl_sch	1 个字	启动开关设置(1：CRST，2：WRST，0：未定义、不存在或无法确定)
ken_f	1 个字	保留
ken_rel	1 个字	有效检验和的 ID (0：无效)
ken_ver1_hw	1 个字	硬件配置的检验和 1 (Intel 格式)： 所有系统数据块长度的异或操作
ken_ver2_hw	1 个字	硬件配置的检验和 2 (Intel 格式)： 所有系统数据块检验和的异或操作
ken_ver1_awp	1 个字	用户程序的检验和 1 (Intel 格式)： 以下数据块长度的异或操作：OB、DB、FB、FC
ken_ver2_awp	1 个字	用户程序的检验和 2 (Intel 格式)： 以下数据块检验和的异或操作：OB、DB、FB、FC

名称	长度	含义
Res	2 个字	保留
sfc_req	1 个字	使用 SFC 109 设置保护等级 2 或 3 时需要使用(1 : 满足要求)
sfc_act	1 个字	使用 SFC 109 激活保护等级 2 或 3(1 : 已激活)
res	4 个字	保留

## 34.20 SSL-ID W#16#xy71 - H CPU 组信息

### 用途

SSL-ID 为 W#16#xy71 的部分列表摘录包含有关 H 系统当前状态的信息。

### 文件头

部分列表 SSL-ID W#16#xy71 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID： W#16#0071： 有关 H 系统当前状态的信息 W#16#0F71： 仅限 SSL 部分列表报头信息
INDEX	W#16#0000
LENTHDR	W#16#0010： 一个数据记录为 8 个字长(16 字节)
N_DR	W#16#0001： 数据记录的数目



## 数据记录

部分列表摘录 ID W#16#xy71 的数据记录具有如下结构：

目录	长度	含义
Redinf	2 个字节	关于冗余的信息 W#16#0011： 单个 H CPU W#16#0012： 2 个 H 系统中的一个
Mwstat1	1 字节	状态字节 1 位 0： 保留 位 1： 保留 位 2： 保留 位 3： 保留 位 4： 机架 0 中 CPU 的 H 状态 =0：待机 CPU =1：主站 CPU 位 5： 机架 1 中 CPU 的 H 状态 =0：待机 CPU =1：主站 CPU 位 6： 保留 位 7： 保留
Mwstat2	1 字节	状态字节 2 位 0： 同步链接 01 的状态： CPU 0 和 CPU 1 之间的同步 =0：不可能 =1：可能 位 1： 0 位 2： 0 位 3： 保留 位 4： =0：CPU 未插入机架 0 中 =1：CPU 插入到机架 0 中 (在冗余模式中：位 4 = 0) 位 5： =0：CPU 未插入机架 1 中 =1：CPU 插入到机架 1 中 (在冗余模式中：位 5 = 0) 位 6： 保留 位 7： 自上次重新启用后的待机主站转换 =0：否 =1：是
Hsfinfo	2 个字节	SFC 90 "H_CTRL"的信息字 位 0： =0：重新启用未激活 =1：重新启用已激活 位 1： =0：待机更新已启用 =1：待机更新已禁用 位 2： =0：待机链接已启用 =1：待机链接已禁用 位 3： 保留 位 4： 保留 位 5： 保留 位 6： 保留 位 7： 保留 位 8： 保留
Samfehl	2 个字节	保留

目录	长度	含义
Bz_cpu_0	2 个字节	机架 0 中 CPU 的模式 W#16#0001 : STOP (更新) W#16#0002 : STOP (复位存储器) W#16#0003 : STOP (自检) W#16#0004 : STOP (内部) W#16#0005 : STARTUP (冷重启) W#16#0006 : STARTUP (暖重启) W#16#0007 : STARTUP (热启动重启) W#16#0008 : RUN (单独模式) W#16#0009 : RUN-R (冗余模式) W#16#000A : HOLD W#16#000B : LINK-UP W#16#000C : UPDATE W#16#000D : DEFECTIVE W#16#000E : SELFTEST W#16#000F : NO POWER
Bz_cpu_1	2 个字节	机架 1 中 CPU 的模式 (与 bz_cpu_0 值相同)
Bz_cpu_2	2 个字节	保留
Cpu_valid	1 字节	变量 bz_cpu_0 和 bz_cpu_1 的有效性 B#16#01 : bz_cpu_0 有效 B#16#02 : bz_cpu_1 有效 B#16#03 : bz_cpu_0 和 bz_cpu_1 有效
hsync_f	1 字节	连接质量的状态(仅当在 mwstat2 中已置位位 0 时才有效) <ul style="list-style-type: none"> <li>• 位 0 : 上接头中同步模块的光纤连接质量受到限制。</li> <li>• 位 1 : 下接头中同步模块的光纤连接质量受到限制。</li> </ul> 位 2 到 7 : 0

## 34.21 SSL-ID W#16#xy75 - H 系统中切换的 DP 从站/IO 设备

### 用途

对于处于冗余 H 工作模式的 H 系统的 CPU，如果读取部分列表 SSL-ID W#16#xy75，将获得有关 H 系统和切换的 DP 从站/IO 设备之间通信状态的信息。

该部分列表将报告当前用于与 DP 从站/IO 设备通信的 DP 主站/IO 控制器的接口模块被插入到哪一个机架中。

### 文件头

部分列表 SSL-ID W#16#xy75 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID W#16#0C75：H 系统和切换的 DP 从站/IO 设备之间的通信状态。通过 INDEX 参数选择 DP 从站/IO 设备。
INDEX	DP 从站/IO 设备的接口模块的诊断地址
LENTHDR	W#16#0010：一个数据记录为 8 个字长(16 字节)
N_DR	W#16#0001：数据记录的数目

## 数据记录

部分列表 ID SSL-ID W#16#xy75 的数据记录具有如下结构：

名称	长度	含义
adr1_bgt0	1 个字	DP 主站/IO 控制器接口模块插入到机架 0 中的 DP 从站/IO 设备接口模块的第一个地址部分： DP：高字节：DP 主站系统 ID，低字节：站号  PNIO: 位 15=1，位 11 到 14：PROFINET IO 系统 ID 的最后两位，位 0 到 10：站号
adr2_bgt0	1 个字	DP 主站/IO 控制器接口模块插入到机架 0 中的 DP 从站/IO 设备接口模块的第二个地址部分： DP：高字节：插槽，低字节：零(无子模块)  PNIO: 高字节：插槽，低字节：子模块插槽
adr1_bgt1	1 个字	DP 主站/IO 控制器接口模块插入到机架 1 中的 DP 从站/IO 设备接口模块的第一个地址部分： DP：高字节：DP 主站系统 ID，低字节：站号  PNIO: 位 15=1，位 11 到 14：PROFINET IO 系统 ID 的最后两位，位 0 到 10：站号
adr2_bgt1	1 个字	DP 主站/IO 控制器接口模块插入到机架 1 中的 DP 从站/IO 设备接口模块的第二个地址部分： DP：高字节：插槽，低字节：零(无子模块)  PNIO: 高字节：插槽，低字节：子模块插槽
Res	2 个字	保留
Logadr	1 个字	DP 从站/IO 设备的接口模块的诊断地址： <ul style="list-style-type: none"> <li>位 0 到 14：逻辑基址</li> <li>位 15：I/O 标识符(0 = 输入，1 = 输出)</li> </ul>
Slavestatus	1 个字	通讯状态： <ul style="list-style-type: none"> <li>位 0 = 1：不能访问 DP 主站/IO 控制器接口模块插入到机架 0 中的 DP 从站/IO 设备的接口模块</li> <li>位 1 = 1：不能访问 DP 主站/IO 控制器接口模块插入到机架 1 中的 DP 从站/IO 设备的接口模块</li> <li>位 2 到 7：保留(每位 = 0)</li> <li>位 8 = 1：当前正在通过机架 0 中的 DP 主站/IO 控制器接口模块进行通讯</li> <li>位 9 = 1：当前正在通过机架 1 中的 DP 主站/IO 控制器接口模块进行通讯</li> <li>位 10 到 15：保留(每位 = 0)</li> </ul>

## 34.22 SSL-ID W#16#xy90 - DP 主站系统信息

### 用途

如果读取 SSL-ID 为 W#16#xy90 的部分列表，将获得 CPU 已知的所有 DP 主站系统的状态信息。

### 文件头

部分列表 SSL-ID W#16#xy90 的报头结构如下所示：

目录	含义	
SSL-ID	部分列表摘录的 SSL-ID	
	W#16#0090 :	CPU 已知的所有 DP 主站系统的信息
	W#16#0190 :	一个 DP 主站系统的信息
	W#16#0F90 :	仅限 SSL 部分列表报头信息
INDEX	<ul style="list-style-type: none"> <li>对于 SSL-ID 为 W#16#0190 的部分列表摘录： 低位字节：B#16#00 高位字节：DP 主站系统 ID</li> <li>对于 SSL-ID 为 W#16#0090 和 W#16#0F90 的部分列表摘录： W#16#0000</li> </ul>	
LENTHDR	W#16#000E :	一个数据记录为 7 个字长(14 字节)
N_DR	数据记录的数目 <ul style="list-style-type: none"> <li>对于 SSL-ID 为 W#16#0190 的部分列表摘录： 0 - 1</li> <li>对于 SSL-ID 为 W#16#0090 的部分列表摘录： 对于标准 CPU： 0 14 对于 H 系统： 0 - 12 (在除冗余以外的所有系统状态下) 0 - 2 x 12 (在冗余系统状态下)</li> </ul>	

## 数据记录

部分列表 ID 为 W#16#xy90 的数据记录具有下列结构：

名称	长度	含义	
dp_m_id	1 字节	DP 主站系统 ID	
rack_dp_m	1 字节	DP 主站的机架号 <ul style="list-style-type: none"> <li>• 对于标准 CPU : 0</li> <li>• 对于 H 系统 : 0 或 1</li> </ul>	
Steckpl_dp_m	1 字节	DP 主站的插槽或 CPU 的插槽(具有集成的 DP 接口)	
Subm_dp_m	1 字节	<ul style="list-style-type: none"> <li>• 具有集成的 DP 接口 : DP 主站的接口编号 : <ul style="list-style-type: none"> <li>- 1: X2</li> <li>- 2: X1</li> <li>- 3: IF1</li> <li>- 4: IF2</li> </ul> </li> <li>• 具有外部 DP 接口 : 0</li> </ul>	
Logadr	1 个字	DP 主站的逻辑起始地址	
dp_m_sys_cpu	1 个字	保留	
dp_m_sys_dpm	1 个字	保留	
dp_m_state	1 字节	DP 主站系统的其它属性	
		位 0 :	DP 模式 <ul style="list-style-type: none"> <li>• 0: S7 兼容</li> <li>• 1: DPV1</li> </ul>
		第 1 位	DP 循环 <ul style="list-style-type: none"> <li>• 0: 不等距</li> <li>1: 等距</li> </ul>
		位 2 到 6 :	保留
		位 7 :	DP 主站类型 <ul style="list-style-type: none"> <li>• 0: 集成的 DP 主站</li> <li>• 1: 外部 DP 主站</li> </ul>
保留	3 字节	保留	

## 有关多值计算的注释(仅限 S7-400)

所有部分列表仅提供被分配给一个 CPU 的模块的有关信息。因此，在多值计算模式中，必须对所有 CPU 进行采样以便获得所有连接模块的数据。

## 34.23 SSL-ID W#16#xy91 - 模块状态信息

### 用途

如果读取 SSL-ID 为 W#16#xy91 的部分列表，将获得已分配给 CPU 的模块的状态信息。

### 注意

不能为具有压缩地址(ET 200S)的子模块判断 SSL-ID W#16#xy91。

### 文件头

部分列表 SSL-ID W#16#xy91 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID
W#16#0091	所有已插入和已组态的 CPU 模块/子模块的模块状态信息 (仅 S7-400)
W#16#0191	所有具有错误模块 ID 的未被禁用的模块/机架的状态信息 (仅 S7-400)
W#16#0291	未被禁用的所有故障模块的模块状态信息(仅 S7-400)
W#16#0391	所有不可用模块的模块状态信息(仅 S7-400)
W#16#0591	主机模块的所有子模块的模块状态信息
W#16#0991	DP 主站系统的模块状态信息
W#16#0A91	所有 DP 子模块及 DP 主站系统(仅限没有 CPU 318-2 DP 的 S7-300)或 PROFINET IO 系统的状态信息
W#16#0C91	中央机架中的模块或使用逻辑基址的集成 DP 接口模块或 PROFINET 接口模块的模块状态信息
W#16#4C91	使用逻辑基址的外部 DP 接口模块的模块状态信息 如果使用四个以上的外部 DP 接口模块， RET_VAL W#16#80A4 可能产生错误结果。
W#16#0D91	与集成接口连接的指定机架 / 指定站(DP 或 PROFINET)中所有模块的模块状态信息。
W#16#0E91	所有已分配模块的模块状态信息
W#16#0F91	仅限 SSL 部分列表报头信息

目录	含义
INDEX	<ul style="list-style-type: none"> <li>• 对于 SSL-ID 为 W#16#0C91 的部分列表摘录：                             <ul style="list-style-type: none"> <li>- S7-400：位 0 到 14：模块的逻辑基址 位 15：0 = 输入，1 = 输出</li> <li>- S7-300：位 0 到 14：模块的任何逻辑基址 位 15：0 = 输入，1 = 输出</li> </ul> </li> <li>• 对于 SSL-ID 为 W#16#4C91 的部分列表摘录(仅限 S7-400)：                             <ul style="list-style-type: none"> <li>位 0 到 14：模块的逻辑基址</li> <li>位 15：0 = 输入，1 = 输出</li> </ul> </li> <li>• 对于 SSL-ID 为 W#16#0091、W#16#0191、W#16#0291、W#16#0391、W#16#0491、W#16#0591、W#16#0A91、W#16#0E91、W#16#0F91 的部分列表摘录： INDEX 不相关，所有模块(在机架和分布式 I/O 中)</li> <li>• 对于 SSL-ID 为 W#16#0991 的部分列表摘录(仅限 S7-400)                             <ul style="list-style-type: none"> <li>- W#16#xx00：DP 主站系统的所有模块(xx 包含 DP 主站系统 ID)</li> </ul> </li> <li>• 对于 SSL-ID W#16#0D91 的部分列表摘录：                             <ul style="list-style-type: none"> <li>- W#16#00xx：一个机架的所有模块和接口模块(xx 包含机架号)</li> <li>- W#16#xxyy：DP 站的所有模块或与集成接口连接的 PROFINET IO 站的所有 IO 设备 (PROFIBUS DP：xx 包含 DP 主站系统 ID，yy 包含站号；PROFINET IO：位 0 到 10：站号，位 11 到 14：PNIO 子系统 ID 的最后两位，位 15：1 (请参见下面关于 adr1 的第三个插图)</li> </ul> </li> </ul>
LENTHDR	W#16#0010：一个数据记录为 8 个字长(16 字节)
N_DR	数据记录的数目。根据产品而定，在 SFC 51 中传送的记录数可能较少

对于 W#16#0091、W#16#0191 和 W#16#0F91，每个机架将另外提供两个数据记录：

- 一个电源记录(只要它存在并已被规划)和
- 一个机架记录。
- 如果是集中式结构，则记录的顺序将为：PS、插槽 1、插槽 2、.....、插槽 18、机架。



部分列表 ID W#16#xy91 的数据记录具有如下结构：

名称	长度	含义
Adr1	1 个字	<ul style="list-style-type: none"> <li>对于中央组态：机架编号</li> <li>对于具有 PROFIBUS DP 的分布式组态：DP 主站系统 ID、站号</li> <li>对于具有 PROFIBUS IO 的分布式组态：位 15 = 1 (PROFINET IO 标识符)PROFINET IO 系统 ID 的最后两位、站号 注意：PROFINET 接口始终被作为"中央组态中的接口模块"进行处理，而不管 PROFINET IO 的使用情况。</li> </ul>
Adr2	1 个字	<ul style="list-style-type: none"> <li>对于中央组态和具有 PROFIBUS DP 的分布式组态：插槽号和接口模块插槽号</li> <li>对于具有 PROFIBUS IO 的分布式组态：插槽号 注意：PROFINET 接口始终被作为"中央组态中的接口模块"进行处理，而不管 PROFINET IO 的使用情况。</li> </ul>
Logadr	1 个字	第一个已分配的逻辑 I/O 地址(基址)
solltyp	1 个字	PROFINET IO：期望的(已组态)类型(参见下文)否则保留
isttyp	1 个字	PROFINET IO：实际类型(参见下文)否则保留
reserviert	1 个字	00xx = CPU-Nr.1-4 (仅限 S7-400) 对于 PROFINET IO： <ul style="list-style-type: none"> <li>SSL ID = W#16#0C91：实际存在的接口模块编号(无接口模块 0)</li> <li>SSL ID = W#16#0D91：接口模块编号(无接口模块 0)</li> <li>SSL ID = W#16#4C91：实际存在的接口模块编号(无接口模块 0)</li> </ul> SSL ID = W#16#4D91：实际存在的接口模块编号(无接口模块 0)
eastat	1 个字	I/O 状态 位 0 = 1： 模块错误 (由诊断中断检测) 位 1 = 1： 模块存在 位 2 = 1： 模块不可用 位 3 = 1： 模块已禁用 位 4 = 1： 站错误(仅限替代插槽) 位 5 = 1： 此模块/站中的 CiR 事件忙或尚未完成 位 6 = 1： 对于 S7-400 保留 位 7 = 1： 局部总线分段中的模块 位 8 到 15： 逻辑地址的数据 ID (输入：B#16#B4，输出：B#16#B5， 集成的 DP 接口：B#16#FE， 外部 DP 接口：B#16#FF)

名称	长度	含义
Ber_bgbr	1 个字	区域 ID/模块宽度 位 0 到 位 2 : 模块宽度 位 3 : 保留 位 4 到 位 6 : 区域 ID 0 = S7-400 1 = S7-300 2 = ET 区域 3 = P 区域 4 = Q 区域 5 = IM3 区域 6 = IM4 区域 位 7 : 保留

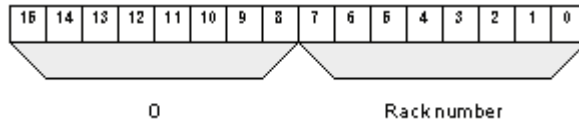
在某些模块中，下列值将显示在记录中：

名称	PS (仅限 S7-400)	CPU	IFM-CPU (S7-300)	机架 (仅限 S7-400)
Adr1	机架号	如上所述的标准信息	如上所述的标准信息	机架号
Adr2	W#16#01FF	W#16#0200 或 W#16#0200 到 W#16#1800	W#16#0200	W#16#00FF
Logadr	W#16#0000	W#16#7FFF	W#16#007C	W#16#0000
Solltyp	如上所述的标准信息	W#16#00C0 或 W#16#0081 或 W#16#0082	W#16#00C0	如上所述的标准信息
Eastat	W#16#0000	如上所述的标准信息	如上所述的标准信息	W#16#0000
Ber_bgbr	W#16#0000	W#16#0011 或 W#16#0001 或 W#16#0002	W#16#0011	W#16#0000

**参数 adr1**

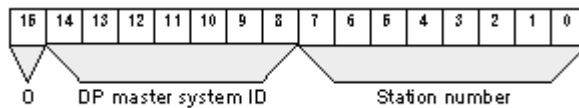
参数 adr1 包含：

- 对于中央组态，机架号(0-31)。



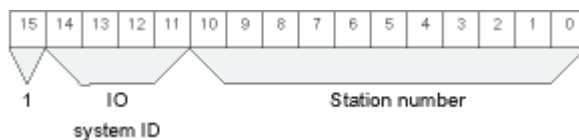
- 对于具有 PROFIBUS DP 的分布式组态：

- DP 主站系统 ID (1-32)
- 站号(0-127)。



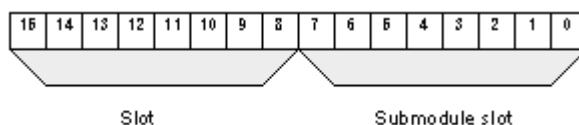
- 对于具有 PROFINET IO 的分布式组态

- PROFINET IO 的标识符位(位 15)
- PROFINET IO 系统 ID (0-15)的最后两位。要获得完整的 PROFINET IO 系统 ID，必须将 100 (十进制)添加到其中
- 站号(0-2047)。

**参数 adr2**

参数 adr2 包含：

- 对于中央组态和具有 PROFIBUS DP 的分布式组态，插槽号和子模块插槽号。



- 对于具有 PROFINET IO 的分布式组态，插槽号。

## PROFINET IO 的 Solltyp (期望的类型)和 Isttyp (实际类型)参数

类型 ID (W#16#...)	含义
8100	如果不能执行类型检查,则为期望的(已组态)类型和实际类型输入此 ID
8101	如果可以执行类型检查,则为期望的(已组态)类型输入此 ID
8101	如果期望的类型 = 实际类型,则将此项作为实际类型输入此 ID
8102	如果期望的类型 <> 实际类型,则将此项作为实际类型输入此 ID

**有关多值计算的信息(仅限 S7-400)**

所有的部分列表只提供分配给一个 CPU 的模块有关信息。因此,在多值计算模式中,必须对所有 CPU 进行采样以便获得所有连接模块的数据。

## 34.24 SSL-ID W#16#xy92 - 机架/站状态信息

### 用途

如果读取部分列表 SSL-ID W#16#xy92，将获得有关一个 DP 主站系统的集中安装的机架及站的期望和当前的硬件组态的信息。

### 通过 SFC51 "RDSYSST"用 S7-400 CPU 读取 SSL

如果通过 SFC51 读取部分列表，则必须使 SFC51 的参数 SSL\_ID 与 INDEX 相匹配。

SSL_ID	INDEX
W#16#0092 或 W#16#0192 或 W#16#0292 或 W#16#0392 或 W#16#0492 或 W#16#0592 或 W#16#0692 或	通过 <b>集成</b> DP 开关连接的 DP 主站系统的 DP 主站系统 ID。
W#16#4092 或 W#16#4292 或 W#16#4692 或	通过 <b>外部</b> DP 开关连接的 DP 主站系统的 DP 主站系统 ID。

## 文件头

部分列表 SSL-ID W#16#xy92 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID： W#16#0092：通过集成 DP 接口连接的 DP 主站系统的中央机架/站的期望状态 W#16#4092：通过外部 DP 接口连接的 DP 主站系统的站的期望状态 W#16#0292：通过集成 DP 接口连接的 DP 主站系统的中央机架/站的实际状态 W#16#0392：至少一个电池发生故障时 CPU 机架/站的电池充电缓冲状态 W#16#0492：一个 CPU 的所有机架/站的整个电池充电缓冲的状态 W#16#0592：一个 CPU 的所有机架/站的 24V 电源状态 W#16#4292：通过外部 DP 接口连接的 DP 主站系统的站的实际状态 W#16#0692：中央组态中的扩展机架/通过集成 DP 接口连接的 DP 主站系统的站的"正常"状态 W#16#4692：通过外部 DP 接口连接的 DP 主站系统的站的"正常"状态
INDEX	0 / DP 主站系统 ID
LENTHDR	W#16#0010：一个数据记录为 8 个字长(16 字节)
N_DR	数据记录的数目

## 数据记录

SSL ID 为 W#16#xy92 的部分列表的数据记录具有如下结构：

目录	长度	含义
status_0 到 status_15	16 字节	机架状态 / 站状态、备份状态或激活状态。(激活状态只与 DP 模块相关) W#16#0092 : 位 = 0 : 机架/站未组态 位 = 1 : 机架/站已组态 W#16#4092 : 位 = 0 : 站未组态 位 = 1 : 站已组态 W#16#0192 : 位 = 0 : 站未组态或已组态并已激活 位 = 1 : 配置和停用站 W#16#0292 : 位 = 0 : 机架/站故障、已取消激活或未组态 位 = 1 : 机架/站存在、已激活、尚未发生故障 W#16#4292 : 位 = 0 : 站故障、已取消激活或未组态 位 = 1 : 站存在、已激活、尚未发生故障 W#16#0692 : 位 = 0 : 扩展机架/一个站的所有模块存在、可用并且 正常, 该站已激活 位 = 1 : 扩展机架/一个站至少有 1 个模块不"正常"或者 该站已取消激活 W#16#4692 : 位 = 0 : 一个站的所有模块存在、可用并且正常, 该站已激活 位=1: 一个站至少有 1 个模块不"正常"或者该站已 取消激活
status_0	1 字节	位 0 : 中央机架(INDEX = 0)或 站 1 (INDEX tu0) 位 1 : 1. 扩展机架或站 2 : : 位 7 : 7. 扩展机架或站 8
status_1	1 字节	位 0 : 8. 扩展机架或站 9 : : 位 7 : 15. 扩展机架或站 16
status_2	1 字节	位 0 : 16. 扩展机架或站 17 : : 位 5 : 21. 扩展机架或站 22 位 6 : 0 或站 23 位 7 : 0 或站 24
status_3	1 字节	位 0 : 0 或站 25 : : 位 5 : 0 或站 30 位 6 : 扩展机架(SIMATIC S5 区域) 或站 31 位 7 : 0 或站 32
status_4	1 字节	位 0 : 0 或站 33 : : 位 7 : 0 或站 40
:		
:		

目录	长度	含义
status_15	1 字节	位 0 :            0            或站 121
		:
		:
		位 7 :            0            或站 128

---

**有关多值计算的信息(仅限 S7-400)**

所有的部分列表只提供分配给一个 CPU 的模块有关信息。因此,在多值计算模式中,必须对所有 CPU 进行采样以便获得所有连接模块的数据。

---



## 34.25 SSL-ID W#16#xy95 - 扩展的 DP 主站系统/PROFINET IO 系统信息

### 用途

SSL ID 为 W#16#xy95 的部分列表提供有关 CPU 已知的所有 DP 主站系统/PROFINET IO 系统的扩展状态信息。与 SSL ID W#16#xy90 的部分列表相比，该列表包含关于 PROFINET IO 系统的声明和关于 DP 主站系统等时模式的附加信息。

### 文件头

部分列表 SSL ID W#16#xy95 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL ID
	W#16#0195 : 有关 DP 主站系统/PROFINET IO 系统的扩展信息
	W#16#0F95 : 仅限与 SSL 部分列表的报头相关的信息
INDEX	<ul style="list-style-type: none"> <li>对于 SSL ID 为 W#16#0195 的部分列表摘录：               <ul style="list-style-type: none"> <li>低字节：B#16#00</li> <li>高字节：DP 主站系统 ID</li> </ul> </li> <li>对于 SSL ID 为 W#16#0F95 的部分列表摘录：               <ul style="list-style-type: none"> <li>W#16#0000</li> </ul> </li> </ul>
LENTHDR	W#16#0028 : 一个数据记录为 20 个字长(40 字节)
N_DR	数据记录的数目： 对于 SSL ID 为 W#16#0195 的部分列表摘录：0 - 1

## 数据记录

ID 为 W#16#xy95 的部分列表的数据记录具有下列结构：

名称	长度	含义
dp_m_id	1 字节	DP 主站系统 ID/PROFINET IO 系统 ID
rack_dp_m	1 字节	DP 主站/IO 控制器的机架号 <ul style="list-style-type: none"> <li>对于标准 CPU : 0</li> <li>对于 H 系统 : 0 或 1</li> </ul>
steckpl_dp_m	1 字节	DP 主站/IO 控制器的插槽或 CPU 的插槽(具有集成的 DP 接口)
subm_dp_m	1 字节	<ul style="list-style-type: none"> <li>具有集成的 DP 接口 : DP 主站/IO 控制器的接口 ID : <ol style="list-style-type: none"> <li>X2</li> <li>X1</li> <li>IF1</li> <li>IF2</li> </ol> </li> <li>具有外部 DP 接口 : 0</li> </ul>
logadr	2 个字节	DP 主站/IO 控制器的逻辑起始地址
dp_m_sys_cpu	2 个字节	保留
dp_m_sys_dpm	2 个字节	保留
dp_m_state	1 字节	DP 主站系统/PROFINET IO 系统的更多属性
		位 0 : DP 模式(仅 PROFIBUS DP) <ul style="list-style-type: none"> <li>0: S7 兼容</li> <li>1: DPV1</li> </ul>
		位 1 : DP 或 PN 周期 <ul style="list-style-type: none"> <li>0: 非恒定总线周期</li> <li>1: 恒定总线循环时间</li> </ul>
		位 2 到 6 : 保留
		位 7 : DP 主站/IO 控制器的类型 <ul style="list-style-type: none"> <li>0: 集成的 DP 主站/IO 控制器</li> <li>1: 外部 DP 主站/IO 控制器</li> </ul>
dp_address	1 字节	DP 节点编号(PROFIBUS 地址)
保留	2 个字节	保留
tsal_ob	1 字节	已配置的时钟同步中断 OB(仅在 DP 或 PN 总线周期恒定时相关)
保留	1 字节	保留
波特率	4 字节	DP 主站系统或 PN 系统的传输率(十六进制值)
dp_iso_takt	4 字节	恒定 DP 或 PN 周期时间( $\mu$ s)
保留	16 字节	保留

## 有关多值计算的信息(仅限 S7-400)

所有的部分列表只提供分配给一个 CPU 的模块有关信息。因此,在多值计算模式中,必须对所有 CPU 进行采样以便获得所有连接模块的数据。

## 34.26 SSL-ID W#16#xyA0 - 诊断缓冲区

### 用途

如果读取 SSL-ID 为 W#16#xyA0 的部分列表，可获得该模块诊断缓冲区中的条目。

### 注意

S7-300 CPU 最多提供与在 RUN 模式中显示的诊断缓冲区条目相同数目的数据记录(默认值：10)。S7-400 CPU 提供最多 21 个记录。

### 文件头

部分列表 SSL-ID W#16#xyA0 的报头结构如下所示：

目录	含义
SSL-ID	部分列表摘录的 SSL-ID： W#16#00A0： 当前模式下所有条目可用 W#16#01A0： 最新条目；使用 INDEX 参数指定最新条目的数目。 如果诊断缓冲区中的消息数小于已组态的最大消息数，SFC51 会通过此部分列表摘录提供无效值。因此，应避免未备份的断电损失！ W#16#0FA0： 仅限部分列表报头信息
INDEX	仅适用于 SSL-ID W#16#01A0：最新条目的数目
LENTHDR	W#16#0014：一个数据记录为 10 个字长(20 字节)
N_DR	数据记录的数目

### 数据记录

部分列表 SSL-ID W#16#xyA0 的数据记录具有下列结构：

名称	长度	含义
ID	1 个字	事件 ID
信息	5 个字	有关事件及其结果的信息
时间	4 个字	事件的时间标志

### 诊断缓冲区

使用 STEP 7 可在诊断缓冲区中获取事件的更多信息。

## 34.27 SSL-ID W#16#00B1 - 模块诊断信息

### 用途

如果读取 SSL-ID 为 W#16#00B1 部分列表，将获取具有诊断能力的模块的前 4 个诊断字节。

### 文件头

部分列表 SSL-ID W#16#00B1 的报头结构如下所示：

	含义
SSL-ID	W#16#00B1
INDEX	位 0 到位 14：逻辑基址 位 15：0 = 输入，1 = 输出
LENTHDR	W#16#0004：一个数据记录为 2 个字长(4 字节)
N_DR	1

### 数据记录

部分列表 SSL-ID W#16#00B1 的数据记录具有如下结构：

名称	长度	含义
Byte0	1 字节	位 0：模块故障/正常(组故障 ID) 位 1：内部故障 位 2：外部故障 位 3：存在通道错误 位 4：无外部辅助电压 位 5：无前置连接器 位 6：未分配参数给模块 位 7：模块的参数错误
Byte1	1 字节	位 0 到位 3：模块等级(CPU、FM、CP、IM、SM, ...) 位 4：存在通道信息 位 5：存在用户信息 位 6：来自替换的诊断中断 位 7：维护要求(仅 PROFINET IO)
Byte2	1 字节	位 0：用户模块不正确/不存在 位 1：通讯故障 位 2：RUN/STOP 模式(0 = RUN, 1 = STOP) 位 3：监视狗已响应 位 4：内部模块电源故障 位 5：电池耗尽(BFS) 位 6：整个缓冲区故障 位 7：维护要求(仅 PROFINET IO)

名称	长度	含义
Byte3	1 字节	位 0 :        扩展机架故障(由 IM 检测) 位 1 :        处理器故障 位 2 :        EPROM 错误 位 3 :        RAM 错误 位 4 :        ADC/DAC 错误 位 5 :        保险丝熔断 位 6 :        硬件错误丢失 位 7 :        保留(用 0 初始化)

#### 有关多值计算的信息(仅限 S7-400)

所有的部分列表只提供分配给一个 CPU 的模块有关信息。因此,在多值计算模式中,必须对所有 CPU 进行采样以便获得所有连接模块的数据。

## 34.28 SSL-ID W#16#00B2 - 对应物理地址的诊断数据记录 1

### 用途

如果读取 SSL-ID 为 W#16#00B2 的部分列表，将获得集中机架中的模块的诊断数据记录 1(不适用于 DP 或子模块)。使用机架和插槽号指定编号。

### 文件头

部分列表 SSL-ID W#16#00B2 的报头结构如下所示：

目录	含义
SSL-ID	W#16#00B2
INDEX	W#16#xxyy :   xx 包含机架号 yy 包含插槽号
LENTHDR	数据记录的长度取决于模块。
N_DR	1

### 数据记录

部分列表 SSL-ID W#16#00B2 的数据记录的大小及其内容取决于特定模块。有关详细信息，请参考 I70/、I101/以及描述相关模块的手册。

---

#### 有关多值计算的信息(仅限 S7-400)

所有的部分列表只提供分配给一个 CPU 的模块有关信息。因此，在多值计算模式中，必须对所有 CPU 进行采样以便获得所有连接模块的数据。

---

## 34.29 SSL-ID W#16#00B3 - 对应逻辑基址的模块诊断数据

### 用途

如果读取 SSL-ID 为 W#16#00B3 的部分列表，将获得模块的所有诊断数据。还可以获取 DP 和子模块的此信息。使用模块的逻辑基址来选择模块。

### 文件头

部分列表 SSL-ID W#16#00B3 的报头结构如下所示：

目录	含义
SSL-ID	W#16#00B3
INDEX	位 0 到位 14：逻辑基址 位 15：0 = 输入，1 = 输出
LENTHDR	数据记录的长度取决于模块。
N_DR	1

### 数据记录

部分列表 SSL-ID W#16#00B3 的数据记录的大小及其内容取决于具体模块。有关详细信息，请参考 [/70/Gloss.70.](#)、[/101/](#)以及描述相关模块的手册。

---

#### 有关多值计算的信息(仅限 S7-400)

所有的部分列表只提供分配给一个 CPU 的模块有关信息。因此，在多值计算模式中，必须对所有 CPU 进行采样以便获得所有连接模块的数据。

---



---

#### 注意

使用 SFC51，只能在 OB82 外部读取 SSL-ID 为 W#16#00B3 的部分列表。

---

## 34.30 SSL-ID W#16#00B4 - DP 从站的诊断数据

### 用途

如果读取 SSL-ID 为 W#16#00B4 的部分列表,将获得 DP 从站的诊断数据。此诊断数据按照 EN 50 170 Volume 2, PROFIBUS 标准构建。可使用已组态的诊断地址选择模块。

### 文件头

部分列表 SSL-ID W#16#00B4 的报头结构如下所示：

目录	含义
SSL-ID	W#16#00B4
INDEX	DP 从站的已组态诊断地址
LENTHDR	数据记录的长度。最大长度为 240 字节。对于具有大于 240 字节的诊断数据长度 (最多可达 244 字节)的标准从站,只读取前 240 个字节并在数据中置位溢出位。
N_DR	1

### 数据记录

部分列表 SSL-ID W#16#00B4 的数据记录具有下列结构：

名称	长度	含义
状态 1	1 字节	站状态 1
状态 2	1 字节	站状态 2
状态 3	1 字节	站状态 3
stat_nr	1 字节	主站号
ken_hi	1 字节	供应商 ID (高字节)
ken_lo	1 字节	供应商 ID (低字节)
....	....	特定从站的更多专用诊断数据

#### 有关多值计算的信息(仅限 S7-400)

所有的部分列表只提供分配给一个 CPU 的模块有关信息。因此,在多值计算模式中,必须对所有 CPU 进行采样以便获得所有连接模块的数据。



# 35 事件

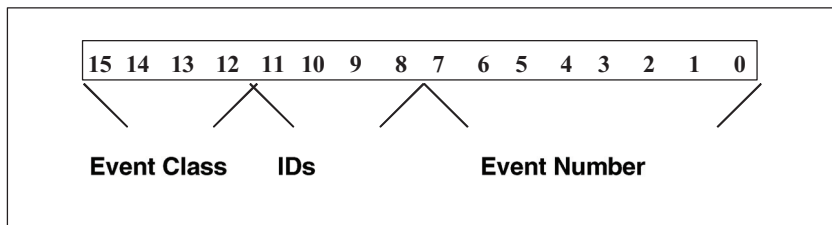
## 35.1 事件和事件 ID

### 事件

在 SIMATIC S7 可编程逻辑控制器中所有事件有编号。允许将消息文本与事件相关。

### 事件 ID

每个事件都分配有事件 ID。事件 ID 的结构如下：



事件 ID 的结构。

### 事件等级

事件等级如下所示：

编号	事件等级
1	标准 OB 事件
2	异步错误
3	异步错误
4	模式转换
5	运行期事件
6	通讯事件
7	故障安全和容错系统事件
8	模块上的标准诊断数据
9	预定义用户事件
A、B	可自由定义的事件
C、D、E	保留
F	除 CPU 以外的模块(例如，CP、FM)的事件

## 标识符

标识符用于区分事件的类型。四个位具有下列意义：

事件 ID 中的位号	含义
8	= 0 事件离开状态 = 1 事件进入状态
9	= 1 诊断缓冲区中的条目
10	= 1 内部错误
11	= 1 外部错误

## 35.2 事件等级 1 - 标准 OB 事件

事件 ID	事件
W#16#1381	要求手动暖重启
W#16#1382	要求自动暖重启
W#16#1383	要求手动热重启
W#16#1384	要求自动热重启
W#16#1385	要求手动冷重启
W#16#1386	要求自动冷重启
W#16#1387	主 CPU：要求手动冷重启
W#16#1388	主 CPU：要求自动冷重启
W#16#138A	主 CPU：要求手动暖重启
W#16#138B	主 CPU：要求自动暖重启
W#16#138C	待机 CPU：要求手动热重启
W#16#138D	待机 CPU：要求自动热重启

### 35.3 事件等级 2 - 同步错误

事件 ID	事件	OB
W#16#2521	BCD 码转换出错	OB 121
W#16#2522	读取数据时发生的区域长度错误	OB 121
W#16#2523	写入数据时发生的区域长度错误	OB 121
W#16#2524	读取时区域出错	OB 121
W#16#2525	写入时区域出错	OB 121
W#16#2526	定时器编号错误	OB 121
W#16#2527	计数器编号错误	OB 121
W#16#2528	读取时对齐出错	OB 121
W#16#2529	写入时对齐出错	OB 121
W#16#2530	访问 DB 时发生写错误	OB 121
W#16#2531	访问 DI 时发生写错误	OB 121
W#16#2532	打开 DB 时块编号错误	OB 121
W#16#2533	打开 DI 时块编号错误	OB 121
W#16#2534	调用 FC 时块编号错误	OB 121
W#16#2535	调用 FB 时块编号错误	OB 121
W#16#253A	未装载 DB	OB 121
W#16#253C	未装载 FC	OB 121
W#16#253D	未装载 SFC	OB 121
W#16#253E	未装载 FB	OB 121
W#16#253F	未装载 SFB	OB 121
W#16#2942	I/O 访问错误，读取	OB 122
W#16#2943	I/O 访问错误，写入	OB 122

## 35.4 事件等级 3 - 异步错误

事件 ID	事件	OB
W#16#3501	超过周期时间。	OB 80
W#16#3502	用户接口(OB 或 FRB)请求错误	OB 80
W#16#3505	由于新时钟的设置而跳过时间中断	OB 80
W#16#3506	在从 HOLD 变为 RUN 时跳过时间中断	OB 80
W#16#3507	多个 OB 请求错误导致内部缓冲区溢出	OB 80
W#16#3508	同步周期中断定时错误	OB 80
W#16#3509	中断负载过大导致中断丢失	OB 80
W#16#350A	在 CiR 之后恢复 RUN 模式	OB 80
W#16#350B	技术同步中断 - 定时错误	OB 80
W#16#3921/3821	BATTF：中央机架的备用电池至少有一个发生故障/问题已排除 注意：只有当其中一个备用电池出现故障(如果有冗余备用电池)时，才会出现事件进入状态。如果其它备用电池也发生故障，则不会再次出现该事件。	OB 81
W#16#3922/3822	BAF：中央机架的备用电压发生故障/问题已排除	OB 81
W#16#3923/3823	中央机架的 24V 电源故障/问题已排除	OB 81
W#16#3925/3825	BATTF：冗余中央机架的备用电池至少有一个发生故障/问题已排除	OB 81
W#16#3926/3826	BAF：冗余中央机架的备用电压发生故障/问题已排除	OB 81
W#16#3917/3827	冗余中央机架的 24V 电源发生故障/问题已排除	OB 81
W#16#3931/3831	BATTF：扩展机架的备用电池至少有一个发生故障/问题已排除	OB 81
W#16#3932/3832	BAF：扩展机架上的备用电压发生故障/问题已排除	OB 81
W#16#3933/3833	至少有一个扩展机架上的 24V 电源发生故障/问题已排除	OB 81
W#16#3942	诊断中断(模块/子模块故障或需要维护或两者)	OB 82
W#16#3842	诊断中断(模块/子模块正常)	OB 82
W#16#3951	PROFINET IO 子模块被删除	OB 83
W#16#3954	PROFINET IO 接口子模块/子模块已拆除	OB 83
W#16#3854	PROFINET IO 接口子模块/子模块与已组态的接口子模块/子模块匹配	OB 83
W#16#3855	PROFINET IO 接口子模块/子模块已插入，但是与已组态的接口子模块/子模块不匹配	OB 83
W#16#3856	PROFINET IO 接口子模块已插入，但是模块参数分配出错	OB 83
W#16#3857	PROFINET IO 子模块/模块已插入，但出错或需要维护	OB 83
W#16#3858	PROFINET IO 接口子模块访问错误已更正	OB 83
W#16#3861	模块/接口模块已插入，模块类型正确	OB 83
W#16#3961	模块/接口模块已删除，不能寻址	OB 83
W#16#3863	模块/接口模块已插入，但模块类型错误	OB 83
W#16#3864	模块/接口模块已插入，但出现问题(类型标识号不可读)	OB 83
W#16#3865	模块已插入，但模块参数分配出错	OB 83
W#16#3866	可以再次寻址模块，负载电压错误已排除	OB 83
W#16#3966	无法寻址模块，负载电压错误	OB 83

事件 ID	事件	OB
W#16#3367	模块重组态开始	OB 83
W#16#3267	模块重组态结束	OB 83
W#16#3968	模块重组态已结束，有错误	OB 83
W#16#3571	在嵌套层中嵌套深度过高	OB 88
W#16#3572	主控继电器的嵌套深度过高	OB 88
W#16#3573	同步错误后的嵌套深度过高	OB 88
W#16#3574	块调用(U 堆栈)的嵌套深度过高	OB 88
W#16#3575	块调用(B 堆栈)的嵌套深度过高	OB 88
W#16#3576	本地数据分配错误	OB 88
W#16#3578	未知指令	OB 88
W#16#357A	跳转指令指向到块外部的目标	OB 88
W#16#3981	输入状态时发生接口错误	
W#16#3881	离开状态时发生接口错误	
W#16#3582	检测到存储器错误并已由操作系统更正	OB 84
W#16#3583	已检测和更正备忘录错误的累积	OB 84
W#16#3585	PC 操作系统中的错误(仅对 Win AC 控制器)	OB 84
W#16#3986	H-Sync 链接的性能受到负面影响	OB 84
W#16#3587	检测到并更正了多位存储器错误	OB 84
W#16#35A1	未找到用户接口(OB 或 FRB)	OB 85
W#16#35A2	未装载 OB (因组态而由 SFC 或操作系统启动)	OB 85
W#16#35A3	操作系统访问块时出错	OB 85
W#16#35A4	不能寻址 PROFINET 接口 DB	OB 85
W#16#34A4	不能再次寻址 PROFINET 接口 DB	OB 85
W#16#39B1	更新过程映像输入表时发生 I/O 访问错误	OB 85
W#16#39B2	将过程映像传送到输出模块时发生 I/O 访问错误	OB 85
W#16#39B3/38B3	更新过程映像输入表时发生 I/O 访问错误	OB 85
W#16#39B4/38B4	在输出的过程映像更新时发生 I/O 访问错误	OB 85
W#16#38C1	扩展机架可再次操作(1 到 21)，离开状态	OB 86
W#16#39C1	扩展机架故障(1 到 21)，进入状态	OB 86
W#16#38C2	扩展机架可再次操作,但设定值与实际组态不匹配	OB 86
W#16#39C3	分布式 I/O：主站系统故障进入状态	OB 86
W#16#39C4	分布式 I/O：站故障，进入状态	OB 86
W#16#38C4	分布式 I/O：站故障，离开状态	OB 86
W#16#38C5	分布式 I/O：站错误返回	OB 86
W#16#38C6	扩展机架可再次操作，但模块参数分配出错	OB 86
W#16#38C7	DP：站可再次操作，但模块参数分配出错	OB 86
W#16#38C8	DP：站可再次操作，但设定值和实际组态不匹配	OB 86
W#16#39CA	PROFINET IO 系统故障	OB 86
W#16#39CB	PROFINET IO 站故障	OB 86
W#16#38CB	PROFINET IO 站可再次操作	OB 86
W#16#38CC	PROFINET IO 站因出错或需要维护而返回	OB 86
W#16#39CD	PROFINET IO 站可再次操作，但预期组态与实际组态不匹配	OB 86

事件 ID	事件	OB
W#16#39CE	PROFINET IO 站可再次操作，但模块参数分配出错	OB 86
W#16#33CF/32CF	PROFINET IO 设备通过 SFC 12 "D_ACT_DP"激活/取消激活时发生故障/返回	W#16#33CF/32CF
W#16#39F8/38F8	PROFINET I 设备的某些子模块发生故障/返回	W#16#39F8/38F8
W#16#38F9	PROFINET I 设备的某些子模块因设备配置不同而发生故障/返回	W#16#38F9
W#16#35D2	目前不能发送诊断条目	OB 87
W#16#35D3	不能发送同步帧	OB 87
W#16#35D4	因同步而产生非法时间跳跃	OB 87
W#16#35D5	采用同步时间时出错	OB 87
W#16#35E1	GD 中的帧 ID 不正确	OB 87
W#16#35E2	不能在 DB 中输入 GD 包状态	OB 87
W#16#35E3	GD 中的帧长度错误	OB 87
W#16#35E4	接收到非法 GD 包编号	OB 87
W#16#35E5	在已组态的 S7 连接的通信 SFB 中访问 DB 出错	OB 87
W#16#35E6	无法在 DB 中输入 GD 所有的状态	OB 87

## 35.5 事件等级 4 - 停止事件和其它模式更改

事件 ID	事件
W#16#4300	备用电源开启
W#16#4301	模式从 STOP 转换为 STARTUP
W#16#4302	模式从 STARTUP 转换为 RUN
W#16#4303	停止已激活的开关而导致 STOP 模式
W#16#4304	PG STOP 操作或 SFB 20"STOP"导致 STOP 模式
W#16#4305	HOLD : 到达断点
W#16#4306	HOLD : 退出断点
W#16#4307	PG 操作启动存储器复位
W#16#4308	开关设置启动存储器复位
W#16#4309	存储器复位自动启动(加电未备份)
W#16#430A	退出 HOLD , 转换为 STOP
W#16#430D	由其它多值计算的 CPU 导致 STOP 模式
W#16#430E	已执行存储器复位
W#16#430F	因一个 CPU 的 STOP 模式而导致模块的 STOP 模式
W#16#4510	CPU 数据范围发生 STOP 违例
W#16#4318	CiR 启动
W#16#4319	CiR 已完成
W#16#4520	DEFECTIVE : 无法进入 STOP 模式
W#16#4521	DEFECTIVE : 处理指令的处理器发生故障
W#16#4522	DEFECTIVE : 时钟芯片发生故障
W#16#4523	DEFECTIVE : 时钟脉冲发生器发生故障
W#16#4524	DEFECTIVE : 定时器更新功能发生故障
W#16#4525	DEFECTIVE : 多值计算同步失败
W#16#4926	DEFECTIVE : I/O 访问监视狗发生故障
W#16#4527	DEFECTIVE : I/O 访问监视发生故障
W#16#4528	DEFECTIVE : 扫描时间监视发生故障
W#16#4530	DEFECTIVE : 内部存储器的存储器测试出错
W#16#4531	STOP 或 DEFECTIVE : 存储器子模块的存储器测试出错
W#16#4532	DEFECTIVE : 核心资源发生故障
W#16#4533	检验和错误
W#16#4934	DEFECTIVE : 存储器不可用
W#16#4535	DEFECTIVE : 因监视狗/处理器例外而取消
W#16#4536	DEFECTIVE : 开关发生故障
W#16#4540	STOP : 内部工作存储器的内存扩展有间隙。第一个存储器扩展过小或丢失。
W#16#4541	因优先级系统而导致 STOP 模式
W#16#4542	因对象管理系统而导致 STOP 模式
W#16#4543	因测试功能而导致 STOP 模式
W#16#4544	因诊断系统而导致 STOP 模式

事件 ID	事件
W#16#4545	因通信系统而导致 STOP 模式
W#16#4546	因 CPU 存储器管理而导致 STOP 模式
W#16#4547	因过程映像管理而导致 STOP 模式
W#16#4548	因 I/O 管理而导致 STOP 模式
W#16#4949	因持续硬件中断而导致 STOP 模式
W#16#454A	因组态而导致 STOP 模式：在 STARTUP 过程中 CPU 装载了 STEP 7 取消选定的 OB
W#16#494D	因 I/O 错误而导致 STOP 模式
W#16#494E	因电源故障而导致 STOP 模式
W#16#494F	因组态错误而导致 STOP 模式
W#16#4550	DEFECTIVE：内部系统错误
W#16#4555	无法重新启动，监视时间已过
W#16#4556	STOP：通信系统的存储器复位请求/由于数据不一致
W#16#4357	模块监视狗启动
W#16#4358	所有模块准备就绪，可以操作
W#16#4959	一个或多个模块操作未准备就绪
W#16#4562	因编程错误而导致 STOP 模式(OB 未装载或不可用)
W#16#4563	因 I/O 访问错误而导致 STOP 模式(OB 未装载或不可用)
W#16#4567	因 H 事件而导致 STOP 模式
W#16#4568	因时间错误而导致 STOP 模式(OB 未装载或不可用)
W#16#456A	因诊断中断而导致 STOP 模式(OB 未装载或不可用)
W#16#456B	因删除/插入模块而导致 STOP 模式(OB 未装载或不可用)
W#16#456C	因 CPU 硬件错误而导致 STOP 模式(OB 未装载或不可用，或无 FRB)
W#16#456D	因程序序列错误而导致 STOP 模式(OB 未装载或不可用)
W#16#456E	因通信错误而导致 STOP 模式(OB 未装载或不可用)
W#16#456F	因机架故障 OB 而导致 STOP 模式(OB 未装载或不可用)
W#16#4570	因过程中断而导致 STOP 模式(OB 未装载或不可用)
W#16#4571	因嵌套堆栈错误而导致 STOP 模式
W#16#4572	因主控继电器堆栈错误而导致 STOP 模式
W#16#4573	因超出同步错误的嵌套深度而导致 STOP 模式
W#16#4574	因超出优先级堆栈中的中断堆栈嵌套深度而导致 STOP 模式
W#16#4575	因超出优先级堆栈中的块堆栈嵌套深度而导致 STOP 模式
W#16#4576	因分配本地数据时出错而导致 STOP 模式
W#16#4578	因未知操作码而导致 STOP 模式
W#16#457A	因代码长度错误而导致 STOP 模式
W#16#457B	因板载 I/O 上未装载 DB 而导致 STOP 模式
W#16#497C	因集成技术而导致 STOP 模式
W#16#457D	请求复位/清除，因为与集成技术的内部接口的版本已更改。
W#16#457F	因 STOP 命令导致 STOP 模式
W#16#4580	STOP：备份缓冲区内容不一致(未转换到 RUN 模式)
W#16#4590	因内部功能超载而导致 STOP 模式
W#16#49A0	因参数分配错误或不允许的设定值和实际扩展变动而导致 STOP 模式：启动被封锁。
W#16#49A1	因参数分配错误而导致 STOP 模式：请求存储器复位



事件 ID	事件
W#16#49A2	因参数修改错误而导致 STOP 模式：启动禁用
W#16#49A3	因参数修改错误而导致 STOP 模式：请求存储器复位
W#16#49A4	STOP：组态数据不一致
W#16#49A5	STOP：分布式 I/O：装载的组态信息不一致
W#16#49A6	STOP：分布式 I/O：组态信息无效
W#16#49A7	STOP：分布式 I/O：无组态信息
W#16#49A8	STOP：由分布式 I/O 接口模块指示错误
W#16#43B0	成功进行了固件更新/备份
W#16#49B1	固件更新数据不正确
W#16#49B2	固件更新：硬件版本与固件不匹配
W#16#49B3	固件更新：模块类型与固件不匹配
W#16#43B4	固件保险丝异常
W#16#43B6	固件更新被冗余模块取消
W#16#49D0	LINK-UP 因协调规则违例而中止
W#16#49D1	LINK-UP/UPDATE 序列中止
W#16#49D2	链接过程中，待机 CPU 因主 CPU 上的 STOP 模式变为 STOP 模式
W#16#43D3	待机 CPU 为 STOP 模式
W#16#49D4	主机为 STOP 模式，因为伙伴 CPU 也是主机(链接错误)
W#16#45D5	因子 PLC 的 CPU 存储器组态不匹配而拒绝 LINK-UP
W#16#45D6	因子 PLC 的系统程序不匹配而拒绝 LINK-UP
W#16#49D7	因用户程序或组态改变而拒绝 LINK-UP
W#16#45D8	DEFECTIVE：检测到因其它错误导致的硬件故障
W#16#45D9	因 SYNC 模块错误而导致 STOP 模式
W#16#45DA	因 H CPU 间同步错误而导致 STOP 模式
W#16#43DC	转换链接过程中中断
W#16#45DD	因运行测试或其它在线功能而拒绝 LINK-UP
W#16#43DE	因第 n 次尝试时超过监视时间而中断，启动新的更新尝试
W#16#43DF	更新因超出完成最大尝试数后的监视时间而最终中断，需要用户干预。
W#16#43E0	链接后从单独模式切换
W#16#43E1	更新后从链接切换
W#16#43E2	从更新变为冗余模式
W#16#43E3	主 CPU：从冗余模式变为单独模式
W#16#43E4	待机 CPU：在错误搜索模式后从冗余模式切换
W#16#43E5	待机 CPU：在链接或 STOP 后从错误搜索模式切换
W#16#43E6	待机 CPU 上中止了链接
W#16#43E7	待机 CPU 上中止了更新
W#16#43E8	待机 CPU：启动后从链接切换
W#16#43E9	待机 CPU：更新后从启动切换
W#16#43F1	保留-主站切换
W#16#43F2	不兼容 H-CPU 的连接被系统程序阻塞
W#16#42F3	检测到检验和错误，并由操作系统更正
W#16#42F4	待机 CPU：主 CPU 中已锁定通过 SFC90 的连接/更新

## 35.6 事件等级 5 - 模式运行期事件

事件 ID	事件
W#16#530D	STOP 模式下新的启动信息
W#16#510F	与 WinLC 同时发生的问题。此问题造成 CPU 进入 STOP 模式或造成 CPU 故障。
W#16#5311	忽略模块的"未就绪"消息而启动
W#16#5545	在 RUN 模式下启动系统重组态
W#16#5445	在 RUN 模式下启动系统重组态
W#16#5380	中断和异步错误的诊断缓冲器条目禁用
W#16#5395	分布式 I/O : DP 主站复位
W#16#5481	所有运行系统软件的许可证再次完成。
W#16#5498	因 CiR 而与 DP 主站系统不再不一致
W#16#5581	运行系统软件的一个或几个许可证丢失。
W#16#558A	组态和已插入 CPU 的 MLFB 间的差异
W#16#558B	组态和已插入 CPU 的固件版本的差异
W#16#5598	因 CiR 与 DP 主站系统可能的不一致性的开始
W#16#5960	切换时参数分配出错
W#16#5961	参数分配出错
W#16#5962	参数分配错误阻止启动
W#16#5963	参数分配错误并请求存储器复位
W#16#5966	切换时参数分配出错
W#16#5967	参数分配错误 : 未知的组态数据
W#16#5968/5858	启用/禁用 Ethernet 接口的紧急地址
W#16#5969	参数分配错误并阻止启动
W#16#596A	PROFINET IO : IO 设备的 IP 地址已存在
W#16#596B	以太网接口的 IP 地址已存在
W#16#596C	以太网接口名已存在
W#16#596D	现有网络组态与系统要求或组态不匹配。
W#16#5371	分布式 I/O : 与 DP 主站的同步结束
W#16#5979/5879	来自 DP 接口的诊断信息 : EXTF LED 打开/关闭
W#16#597C	DP 全局控制命令失败或被移动
W#16#597C	DP 命令"全局控制"失败或被移动
W#16#59A0	CPU 中无法关联中断
W#16#59A1	集成技术中组态出错
W#16#53A2	技术固件下载成功
W#16#59A3	下载集成技术时出错
W#16#53A4	技术 DB 下载失败
W#16#55A5	版本冲突 : 具有集成技术的内部接口
W#16#55A6	已超过技术对象的最大数。
W#16#55A7	此类型的技术 DB 已存在。
W#16#53FF	复位至出厂设置

## 35.7 事件等级 6 - 通讯事件

事件 ID	事件
W#16#6316	启动可编程控制器时接口出错
W#16#6390	微存储卡格式化完成
W#16#6500	连接 ID 在模块中出现两次
W#16#6501	连接资源不足
W#16#6502	连接说明中有错误
W#16#6510	判断 EPROM 时在实例 DB 中检测到 CFB 结构错误
W#16#6514	GD 包编号在模块中出现两次
W#16#6515	组态信息中的规定长度不一致
W#16#6521	无可用的存储器子模块和内部存储器
W#16#6522	非法存储器子模块：更换子模块并复位存储器
W#16#6523	因访问子模块出错而请求存储器复位
W#16#6524	因块报头出错而请求存储器复位
W#16#6526	因存储器更换而请求存储器复位
W#16#6527	已更换存储器，因此无法重新启动
W#16#6528	STOP/HOLD 模式下的对象处理功能，无法重新启动
W#16#6529	"装载用户程序"功能过程中无法启动
W#16#652A	因块在用户存储器中出现两次而不能启动
W#16#652B	因子模块的块过长而不能启动 - 更换子模块
W#16#652C	因子模块中有非法 OB 而不能启动
W#16#6532	因子模块中有非法组态信息而不能启动
W#16#6533	因子模块内容无效而请求存储器复位
W#16#6534	不启动：块在子模块中出现一次以上
W#16#6535	不启动：无足够内存从子模块中传送块
W#16#6536	不启动：子模块中有非法模块编号
W#16#6537	不启动：子模块中包含非法长度的块
W#16#6538	块的本地数据或写保护 ID(对于 DB)对于 CPU 非法
W#16#6539	块中有非法命令(由编译器检测到)
W#16#653A	因子模块上的本地 OB 数据过短而请求存储器复位
W#16#6543	不启动：块类型非法
W#16#6544	不启动："与处理相关的"属性非法
W#16#6545	源语言非法
W#16#6546	已达到某种块类型的块的最大数
W#16#6547	将参数分配给模块时发生参数分配错误(不在 P 总线上，取消下载)
W#16#6548	块检查过程中发生似真的错误
W#16#6549	块中有结构性错误
W#16#6550	在 CRC 中的块有错误
W#16#6551	块没有 CRC
W#16#6353	固件更新：启动通过网络的固件下载

事件 ID	事件
W#16#6253	固件更新：结束通过网络的固件下载
W#16#6560	SCAN 溢出
W#16#6881	离开状态时发生接口错误
W#16#6905/6805	组态连接中有资源问题/已排除
W#16#6981	输入状态时发生接口错误

## 35.8 事件等级 7 - H/F 事件

事件 ID	事件	OB
W#16#72A2	DP 主站或 DP 主站系统发生故障	OB 70
W#16#72A3	DP 从站上的冗余恢复	OB 70
W#16#7301	CPU 故障造成冗余丢失(2 个中的 1 个)	OB 72
W#16#7302	用户触发待机 STOP 状态造成的冗余丢失(2 个中的 1 个)	OB 72
W#16#7303	H 系统(2 个中的 1 个)变为冗余模式	OB 72
W#16#7323	在操作系统数据中发现偏差	OB 72
W#16#7331	因主站故障而发生待机主站切换	OB 72
W#16#7333	因运行期间系统修改而发生待机主站切换	OB 72
W#16#7334	因同步模块的通信错误而发生待机主站切换	OB 72
W#16#7340	因等待时间已到期而在用户程序中发生同步错误	OB 72
W#16#7341	因在不同的同步点等待而在用户程序中发生同步错误	OB 72
W#16#7342	因在不同的同步点等待而在操作系统中发生同步错误	OB 72
W#16#7343	因等待时间已到期而在操作系统中发生同步错误	OB 72
W#16#7344	因数据不正确而在操作系统中发生同步错误	OB 72
W#16#734A	因执行了 SFC 90 "H_CTRL"而触发的"重新启用"作业。	OB 72
W#16#73A3	DP 从站中冗余丢失	OB 70
W#16#73D8	安全模式禁用	
W#16#73E0/72E0	通讯中冗余丢失/问题已排除	OB 73
W#16#7520	RAM 比较出错	OB 72
W#16#7521	过程映像输出值的比较出错	OB 72
W#16#7522	存储器位、定时器或计数器的比较出错	OB 72
W#16#73C1	更新过程已取消	OB 72
W#16#73C2	因第 n 次尝试期间超出监视时间而中止更新( $1 \leq n \leq$ 因超过监视时间而终止后的最大可能更新尝试数目)	OB 72
W#16#75D1	安全程序：内部 CPU 出错	
W#16#75D2	安全程序错误：周期超时	
W#16#75D6	安全程序中的数据在输出至 F I/O 之前损坏	
W#16#75D7	安全程序中的数据在输出至伙伴 F CPU 之前损坏	
W#16#75D9	DB 中有无效实型数	
W#16#75DA	安全程序：安全数据格式出错	
W#16#73DB/72DB	安全程序：安全模式已启用/已禁用	

事件 ID	事件	OB
W#16#75DC	运行期间组、内部协议错误	
W#16#75DD/74DD	安全程序：故障安全运行组关闭已启用/已禁用	
W#16#75DE/74DE	安全程序：F 程序关闭已启用/已禁用	-
W#16#75DF/74DF	F 程序初始化开始/结束	-
W#16#75E1	安全程序：FB"F_PLK"、"F_PLK_O"、"F_CYC_CO"、"F_TEST"或"F_TESTC"中出错	
W#16#7934	因 SYNC 模块连接问题切换待机主站	OB 72
W#16#7950	同步模块丢失	OB 72
W#16#7951	未上电时 SYNC 模块发生改变	OB 72
W#16#7952/7852	SYNC 模块已删除/已插入	OB 72
W#16#7953	SYNC 模块的改变未复位	OB 72
W#16#7954	SYNC 模块：机架号被分配了两次	OB 72
W#16#7955/7855	SYNC 模块出错/已排除	OB 72
W#16#7956	SYNC 模块中设置了非法机架号	OB 72
W#16#7960	冗余 I/O：数字输入时偏差时间超时，错误尚未本地化	
W#16#7961	冗余 I/O，数字输入错误：偏差时间过期后信号变化	
W#16#7962	冗余 I/O：数字输入错误	-
W#16#7963	冗余 I/O：数字输入通道错误：	
W#16#7964	冗余 I/O：数字输入错误：差异时间用完后信号改变	
W#16#796F	冗余 I/O：全局激活了 I/O	-
W#16#7970	冗余 I/O：数字输出错误	-
W#16#7971	冗余 I/O：数字输出通道错误：	
W#16#7980	冗余 I/O：模拟输入的偏差时间超时	-
W#16#7981	冗余 I/O：模拟输入错误	-
W#16#7982	冗余 I/O：模拟输入通道错误	
W#16#7983	冗余 I/O：模拟输入通道的差异时间已用完	
W#16#7984	冗余 I/O：模拟输入通道错误	
W#16#7990	冗余 I/O：模拟输出错误	-
W#16#7991	冗余 I/O：模拟输出通道错误	
W#16#73A2	DP 主站或 DP 主站系统发生故障	
W#16#73A3/72A3	冗余丢失/返回 DP 从站上的冗余	
W#16#73A5/72A5	冗余丢失/返回 IO 设备上的冗余	OB 70
W#16#73C1	链接/更新中止	OB 72
W#16#73C2	由于第 n 次尝试(1 ≤ n ≤ 因超时而中止后的最大可能更新尝试数目)超出监视时间而中止更新	OB 72
W#16#75D1	安全程序：内部 CPU 出错	-
W#16#75D2	安全程序错误：超出周期	-
W#16#79D3/78D3	PROFIsafe 与 F I/O 间的通讯出错	-
W#16#79D4/78D4	F CPU 之间与安全相关的通讯中出错	-
W#16#79D5/78D5	F CPU 之间与安全相关的通讯中出错	-
W#16#75D6	安全程序中的数据在输出至 F I/O 之前损坏	
W#16#75D7	安全程序中的数据在输出至伙伴 F CPU 之前损坏	
W#16#73D8	安全模式取消激活	

事件 ID	事件	OB
W#16#75D9	DB 中有无效实型数	
W#16#75D9	安全程序：安全数据格式出错	
W#16#73DB/72DB	安全程序：安全模式激活/取消激活	
W#16#75DC	运行期间组、内部协议错误	
W#16#75DD/74DD	安全程序：禁用故障安全运行系统组激活/取消激活	
W#16#75DE/74DE	安全程序：完全禁用 F 程序激活/取消激活	
W#16#75DF/74DF	F 程序初始化开始/结束	
W#16#73E0/72E0	丢失通信冗余/已消除	
W#16#75E1	安全程序：FB"F_PLK"、"F_PLK_O"、"F_CYC_CO"、"F_TEST"或 "F_TESTC"中出错	
W#16#75E2	安全程序：区域长度错误	-
W#16#79E3	F-I/O 设备输入通道处于等待状态	-
W#16#78E3	F-I/O 设备输入通道处于主动状态	-
W#16#79E4	F-I/O 设备输入通道处于等待状态	-
W#16#78E4	F-I/O 设备输入通道处于主动状态	-
W#16#79E5	F-I/O 设备处于等待状态	-
W#16#78E5	F-I/O 设备处于主动状态	-
W#16#79E6	不一致的安全程序	-
W#16#79E7	模拟块(F 系统块)已载入	-
W#16#73E8	通过测试确认安全程序的一致性	-
W#16#73E9	无法检查安全程序的一致性	-

## 35.9 事件等级 8 - 模块的诊断事件

事件 ID	事件	模块类型
W#16#8x00	模块故障/正常	Any
W#16#8x01	内部错误	
W#16#8x02	外部错误	
W#16#8x03	通道错误	
W#16#8x04	无外部辅助电压	
W#16#8x05	无前置连接器	
W#16#8x06	无参数分配	
W#16#8x07	模块中的参数不正确	
W#16#8x30	用户子模块不正确/未找到	
W#16#8x31	通信问题	
W#16#8x32	工作模式：RUN/STOP (STOP：进入状态，RUN：离开状态)	
W#16#8x33	时间监视响应(监视狗)	
W#16#8x34	内部模块电源故障	
W#16#8x35	BATTF：电池用尽	
W#16#8x36	所有的备份失败	
W#16#8x40	扩展机架故障	
W#16#8x41	处理器故障	
W#16#8x42	EPROM 错误	
W#16#8x43	RAM 错误	
W#16#8x44	ADC/DAC 错误	
W#16#8x45	保险丝融化	
W#16#8x46	硬件中断丢失	
W#16#8x50	组态/参数分配错误	模拟输入
W#16#8x51	共模错误	
W#16#8x52	相位短路	
W#16#8x53	接地短路	
W#16#8x54	断线	
W#16#8x55	参考通道错误	
W#16#8x56	低于测量范围	
W#16#8x57	高于测量范围	
W#16#8x60	组态/参数分配错误	模拟输出
W#16#8x61	共模错误	
W#16#8x62	相位短路	
W#16#8x63	接地短路	
W#16#8x64	断线	
W#16#8x66	空载电压	
W#16#8x70	组态/参数分配错误	数字输入
W#16#8x71	机壳接地故障	

事件 ID	事件	模块类型
W#16#8x72	相位短路(传感器)	
W#16#8x73	接地短路(传感器)	
W#16#8x74	断线	
W#16#8x75	无传感器电源	
W#16#8x80	组态/参数分配错误	数字量输出
W#16#8x81	机壳接地故障	
W#16#8x82	相位短路	
W#16#8x83	接地短路	
W#16#8x84	断线	
W#16#8x85	保险丝断开	
W#16#8x86	空载电压	
W#16#8x87	温度过高	
W#16#8xB0	计数器模块, 信号 A 错误	FM
W#16#8xB1	计数器模块, 信号 B 错误	
W#16#8xB2	计数器模块, 信号 N 错误	
W#16#8xB3	计数器模块, 通道间传递的值不正确	
W#16#8xB4	计数器模块, 5.2V 传感器电源故障	
W#16#8xB5	计数器模块, 24V 传感器电源故障	



## 35.10 事件等级 9 - 标准用户事件

事件 ID	事件
W#16#9001	自动模式
W#16#9101	手动模式
W#16#9x02	OPEN/CLOSED、ON/OFF
W#16#9x03	手动命令启用
W#16#9x04	单元保护命令(OPEN/CLOSED)
W#16#9x05	过程启用
W#16#9x06	系统保护命令
W#16#9x07	过程值监视响应
W#16#9x08	操作变量监视响应
W#16#9x09	系统偏差大于允许值
W#16#9x0A	限制位置错误
W#16#9x0B	运行错误
W#16#9x0C	命令执行错误(序列发生器)
W#16#9x0D	工作状态运行 > OPEN
W#16#9x0E	工作状态运行 > CLOSED
W#16#9x0F	命令阻塞
W#16#9x11	过程状态 OPEN/ON
W#16#9x12	过程状态 CLOSED/OFF
W#16#9x13	过程状态的中间位置
W#16#9x14	通过 AUTO 将过程状态设置为 ON
W#16#9x15	通过手动设置过程状态为 ON
W#16#9x16	通过保护性命令将过程状态设置为 ON
W#16#9x17	通过 AUTO 将过程状态设置为 OFF
W#16#9x18	通过手动将过程状态设置为 OFF
W#16#9x19	通过保护性命令将过程状态设置为 OFF
W#16#9x21	接近时出现功能错误
W#16#9x22	离开时出现功能错误
W#16#9x31	执行器(DE/WE)限位 OPEN
W#16#9x32	执行器(DE/WE)限位未 OPEN
W#16#9x33	执行器(DE/WE)限位 CLOSED
W#16#9x34	执行器(DE/WE)限位未 CLOSED
W#16#9x41	非法状态, 容差时间耗尽
W#16#9x42	非法状态, 容差时间未耗尽
W#16#9x43	互锁错误, 容差时间 = 0
W#16#9x44	互锁错误, 容差时间 > 0
W#16#9x45	无反应
W#16#9x46	非法退出最终状态, 容差时间 = 0
W#16#9x47	非法退出最终状态, 容差时间 > 0

事件 ID	事件
W#16#9x50	信号范围上限 USR
W#16#9x51	测量范围上限 UMR
W#16#9x52	信号范围下限 LSR
W#16#9x53	测量范围下限 LMR
W#16#9x54	报警上限 UAL
W#16#9x55	警告上限 UWL
W#16#9x56	容差上限 UTL
W#16#9x57	容差下限 LTL
W#16#9x58	警告下限 LWL
W#16#9x59	报警下限 LAL
W#16#9x60	GRAPH7 步骤进入/离开
W#16#9x61	GRAPH7 互锁错误
W#16#9x62	GRAPH7 执行错误
W#16#9x63	GRAPH7 错误已通知
W#16#9x64	GRAPH7 错误已确认
W#16#9x70	趋势在正方向超出
W#16#9x71	趋势在负方向超出
W#16#9x72	无反应
W#16#9x73	非法退出最终状态
W#16#9x80	超过限制值，容差时间 = 0
W#16#9x81	超过限制值，容差时间 > 0
W#16#9x82	低于限制值，容差时间 = 0
W#16#9x83	低于限制值，容差时间 > 0
W#16#9x84	梯度超出，容差时间 = 0
W#16#9x85	梯度超出，容差时间 > 0
W#16#9x86	低于梯度，容差时间 = 0
W#16#9x87	低于梯度，容差时间 > 0
W#16#9190/9090	用户参数分配错误进入/离开
W#16#91F0	溢出
W#16#91F1	下溢
W#16#91F2	被 0 除
W#16#91F3	计算操作非法

## 35.11 事件等级 A 和 B - 自由用户事件

事件 ID	事件
W#16#Axyz	用户可用的事件
W#16#Bxyz	

## 35.12 保留的事件等级

### 保留

以下的事件等级将被保留，供将来扩展：

- C
- D
- E
- F 为集中机架外的模块保留(例如，CP 或 FM)



## 36 SFC 和 SFB 列表

### 36.1 按编号排序的 SFC 列表

编号	短名称	功能
SFC 0	SET_CLK	设置系统时钟
SFC 1	READ_CLK	读系统时钟
SFC 2	SET_RTM	设置运行系统计时器
SFC 3	CTRL_RTM	启动/停止运行系统计时器
SFC 4	READ_RTM	读运行系统计时器
SFC 5	GADR_LGC	查询通道的逻辑地址
SFC 6	RD_SINFO	读 OB 的启动信息
SFC 7	DP_PRAL	在 DP 主站上触发硬件中断
SFC 9	EN_MSG	启用与块相关、符号相关及组状态消息
SFC 10	DIS_MSG	禁用与块相关、符号相关及组状态消息
SFC 11	DPSYC_FR	同步 DP 从站组
SFC 12	D_ACT_DP	取消激活和激活 DP 从站
SFC 13	DPNRM_DG	读 DP 从站的诊断数据(从站诊断)
SFC 14	DPRD_DAT	从标准 DP 从站读取一致性数据
SFC 15	DPWR_DAT	将一致性数据写入 DP 标准从站
SFC 17	ALARM_SQ	生成可确认的与块相关的消息
SFC 18	ALARM_S	生成永久确认的与块相关的消息
SFC 19	ALARM_SC	查询上一个 ALARM_SQ 输入状态消息的确认状态
SFC 20	BLKMOV	复制变量
SFC 21	FILL	初始化存储区
SFC 22	CREAT_DB	创建数据块
SFC 23	DEL_DB	删除数据块
SFC 24	TEST_DB	测试数据块
SFC 25	COMPRESS	压缩用户存储器
SFC 26	UPDAT_PI	更新过程映像更新表
SFC 27	UPDAT_PO	更新过程映像输出表
SFC 28	SET_TINT	设置时间中断
SFC 29	CAN_TINT	取消时间中断
SFC 30	ACT_TINT	激活时间中断
SFC 31	QRY_TINT	查询时间中断
SFC 32	SRT_DINT	启动延时中断
SFC 33	CAN_DINT	取消延时中断
SFC 34	QRY_DINT	查询延时中断
SFC 35	MP_ALM	触发多值计算中断
SFC 36	MSK_FLT	屏蔽同步错误

编号	短名称	功能
SFC 37	DMSK_FLT	取消屏蔽同步错误
SFC 38	READ_ERR	读错误寄存器
SFC 39	DIS_IRT	禁用新的中断和异步错误
SFC 40	EN_IRT	启用新的中断和异步错误
SFC 41	DIS_AIRT	延迟高优先级的中断和异步错误
SFC 42	EN_AIRT	启用高优先级的中断和异步错误
SFC 43	RE_TRIGR	重新触发周期监视
SFC 44	REPL_VAL	将替换值传送到累加器 1
SFC 46	STP	将 CPU 切换到 STOP 模式
SFC 47	WAIT	延迟执行用户程序
SFC 48	SNC_RTCB	同步从站时钟
SFC 49	LGC_GADR	查询属于一个逻辑地址的模块插槽
SFC 50	RD_LGADR	查询模块的所有逻辑地址
SFC 51	RDSYSST	读系统状态列表或部分列表
SFC 52	WR_USMSG	将用户自定义的诊断事件写入诊断缓冲区
SFC 54	RD_PARM	读已定义的参数
SFC 55	WR_PARM	写动态参数
SFC 56	WR_DPARM	写缺省参数
SFC 57	PARM_MOD	为一个模块分配参数
SFC 58	WR_REC	写一条数据记录
SFC 59	RD_REC	读一个数据记录
SFC 60	GD_SND	发送 GD 信息包
SFC 61	GD_RCV	获取一个收到的 GD 信息包
SFC 62	CONTROL	查询属于一个通讯 SFB 实例的连接状态
SFC 63	AB_CALL	组合代码块
SFC 64	TIME_TCK	读系统时间
SFC 65	X_SEND	将数据发送到本地 S7 站外部的通讯伙伴
SFC 66	X_RCV	接收来自本地 S7 站外部的通讯伙伴的数据
SFC 67	X_GET	读来自本地 S7 站外部的通讯伙伴的数据
SFC 68	X_PUT	将数据写入本地 S7 站外部的通讯伙伴
SFC 69	X_ABORT	中止与本地 S7 站外部的通讯伙伴之间的现有连接
SFC 70	GEO_LOG	确定模块的起始地址
SFC 71	LOG_GEO	确定属于逻辑地址的插槽
SFC 72	I_GET	读来自本地 S7 站内的通讯伙伴的数据
SFC 73	I_PUT	将数据写入本地 S7 站内的通讯伙伴
SFC 74	I_ABORT	中止与本地 S7 站内的通讯伙伴之间的现有连接
SFC 78	OB_RT	确定 OB 程序运行时间
SFC 79	SET	设置输出范围
SFC 80	RSET	重置输出范围
SFC 81	UBLKMOV	不可中断的块移动
SFC 82	CREA_DBL	在装载存储器中创建数据块
SFC 83	READ_DBL	从装载存储器中的数据块读

编号	短名称	功能
SFC 84	WRIT_DBL	从装载存储器中的数据块写
SFC 85	CREA_DB	创建数据块
SFC 87	C_DIAG	实际连接状态的诊断
SFC 90	H_CTRL	控制 H 系统中的操作
SFC 99	WWW	启用或同步用户 Web 页面
SFC 100	SET_CLKS	设置日时钟和 TOD 状态
SFC 101	RTM	处理运行系统计时器
SFC 102	RD_DPARA	重新定义的参数
SFC 103	DP_TOPO	标识 DP 主站系统中的总线拓扑
SFC 104	CiR	控制 CiR
SFC 105	READ_SI	读动态系统资源
SFC 106	DEL_SI	删除动态系统资源
SFC 107	ALARM_DQ	生成始终可确认的与块相关的消息
SFC 108	ALARM_D	生成始终可确认的与块相关的消息
SFC 109	PROTECT	激活和取消激活 CPU 访问保护
SFC 112	PN_IN	更新 PROFINET 组件的用户程序接口中的输入
SFC 113	PN_OUT	更新 PROFINET 组件的用户程序接口中的输出
SFC 114	PN_DP	更新 DP 互连
SFC 126	SYNC_PI	以同步循环的方式更新过程映像分区输入表
SFC 127	SYNC_PO	以同步循环的方式更新过程映像分区输出表

SFC 63 "AB\_CALL"仅适用于 CPU 614。详细描述，请参考相应的手册。

## 36.2 按字母排序的 SFC 列表

短名称	编号	功能
AB_CALL	SFC 63	组合代码块
ACT_TINT	SFC 30	激活时间中断
ALARM_D	SFC 108	生成永久可确认的与块相关的消息
ALARM_DQ	SFC 107	生成永久可确认的与块相关的消息
ALARM_S	SFC 18	生成永久确认的与块相关的消息
ALARM_SC	SFC 19	查询上一个 ALARM_SQ 输入状态消息的确认状态
ALARM_SQ	SFC 17	生成可确认的与块相关的消息
BLKMOV	SFC 20	复制变量
C_DIAG	SFC 87	实际连接状态的诊断
CAN_DINT	SFC 33	取消延时中断
CAN_TINT	SFC 29	取消时间中断
CiR	SFC 104	控制 CiR
COMPRESS	SFC 25	压缩用户存储器
CONTROL	SFC 62	查询属于一个通讯 SFB 实例的连接状态
CREA_DB	SFC 85	创建数据块
CREA_DBL	SFC 82	在装载存储器中生成数据块
CREAT_DB	SFC 22	创建数据块
CTRL_RTM	SFC 3	启动/停止运行系统计时器
D_ACT_DP	SFC 12	取消激活和激活 DP 从站
DEL_DB	SFC 23	删除数据块
DEL_SI	SFC 106	删除动态占用的系统资源
DIS_AIRT	SFC 41	延迟高优先级的中断和异步错误
DIS_IRT	SFC 39	禁用新的中断和异步错误
DIS_MSG	SFC 10	禁用与块相关、符号相关及组状态消息
DMSK_FLT	SFC 37	取消屏蔽同步错误
DP_PRAL	SFC 7	在 DP 主站上触发硬件中断
DP_TOPOL	SFC 103	标识 DP 主站系统中的总线拓扑
DPNRM_DG	SFC 13	读 DP 从站的诊断数据(从站诊断)
DPRD_DAT	SFC 14	从标准 DP 从站读取一致性数据
DPSYC_FR	SFC 11	同步 DP 从站组
DPWR_DAT	SFC 15	将一致性数据写入 DP 标准从站
EN_AIRT	SFC 42	启用高优先级的中断和异步错误
EN_IRT	SFC 40	启用新的中断和异步错误
EN_MSG	SFC 9	启用与块相关、符号相关及组状态消息
FILL	SFC 21	初始化存储区
GADR_LGC	SFC 5	查询通道的逻辑地址
GD_RCV	SFC 61	获取一个收到的 GD 信息包
GEO_LOG	SFC 70	确定模块的起始地址



短名称	编号	功能
GD_SND	SFC 60	发送 GD 信息包
H_CTRL	SFC 90	控制 H 系统中的操作
I_ABORT	SFC 74	中止与本地 S7 站内的通讯伙伴之间的现有连接
I_GET	SFC 72	读来自本地 S7 站内的通讯伙伴的数据
I_PUT	SFC 73	将数据写入本地 S7 站内的通讯伙伴
LOG_GEO	SFC 71	确定属于逻辑地址的插槽
LGC_GADR	SFC 49	查询属于一个逻辑地址的模块插槽
MP_ALM	SFC 35	触发多值计算中断
MSK_FLT	SFC 36	屏蔽同步错误
PARAM_MOD	SFC 57	为一个模块分配参数
PN_DP	SFC 114	更新 DP 互连
PN_IN	SFC 112	更新 PROFINet 组件的用户程序接口中的输入
PN_OUT	SFC 113	更新 PROFINet 组件的 PROFINet 接口的输出
PROTECT	SFC 109	激活和取消激活 CPU 访问保护
QRY_DINT	SFC 34	查询延时中断
QRY_TINT	SFC 31	查询时间中断
RD_DPARA	SFC 102	重新定义的参数
RD_LGADR	SFC 50	查询模块的所有逻辑地址
RD_PARM	SFC 54	读已定义的参数
RD_REC	SFC 59	读一个数据记录
RD_SINFO	SFC 6	读 OB 的启动信息
RDSYSST	SFC 51	读系统状态列表或部分列表
RE_TRIGR	SFC 43	重新触发周期监视
READ_CLK	SFC 1	读系统时钟
READ_DBL	SFC 83	从装载存储器中的数据块读
READ_ERR	SFC 38	读错误寄存器
READ_RTM	SFC 4	读运行系统计时器
READ_SI	SFC 105	读动态占用的系统资源
REPL_VAL	SFC 44	将替换值传送到累加器 1
RSET	SFC 80	重置输出范围
RTM	SFC 101	处理运行系统计时器
SET	SFC 79	设置输出范围
SET_CLK	SFC 0	设置系统时钟
SET_CLKS	SFC 100	设置日时钟和 TOD 状态
SET_RTM	SFC 2	设置运行系统计时器
SET_TINT	SFC 28	设置时间中断
SNC_RTCB	SFC 48	同步从站时钟
SRT_DINT	SFC 32	启动延时中断
STP	SFC 46	将 CPU 切换到 STOP 模式
SYNC_PI	SFC 126	以同步循环的方式更新过程映像分区输入表
SYNC_PO	SFC 127	以同步循环的方式更新过程映像分区输出表
TEST_DB	SFC 24	测试数据块

短名称	编号	功能
TIME_TCK	SFC 64	读系统时间
UBLKMOV	SFC 81	不可中断的块移动
UPDAT_PI	SFC 26	更新过程映像更新表
UPDAT_PO	SFC 27	更新过程映像输出表
WAIT	SFC 47	延迟执行用户程序
WR_DPARM	SFC 56	写缺省参数
WR_PARM	SFC 55	写动态参数
WR_REC	SFC 58	写一条数据记录
WR_USMSG	SFC 52	将用户自定义的诊断事件写入诊断缓冲区
WRIT_DBL	SFC 84	从装载存储器中的数据块写
WWW	SFC 99	启用或同步用户 Web 页面
X_ABORT	SFC 69	中止与本地 S7 站外部的通讯伙伴之间的现有连接
X_GET	SFC 67	读来自本地 S7 站外部的通讯伙伴的数据
X_PUT	SFC 68	将数据写入本地 S7 站外部的通讯伙伴
X_RCV	SFC 66	接收来自本地 S7 站外部的通讯伙伴的数据
X_SEND	SFC 65	将数据发送到本地 S7 站外部的通讯伙伴

SFC 63 "AB\_CALL"仅适用于 CPU 614。详细描述，请参考相应的手册。

### 36.3 按数字排序的 SFB 列表

编号	短名称	功能
SFB 0	CTU	递增计数器
SFB 1	CTD	递减计数器
SFB 2	CTUD	递增/递减计数器
SFB 3	TP	生成脉冲
SFB 4	TON	生成接通延迟
SFB 5	TOF	生成断开延迟
SFB 8	USEND	数据的不对等的发送
SFB 9	URCV	数据的不对等的接收
SFB 12	BSEND	发送分段数据
SFB 13	BRCV	接收分段数据
SFB 14	GET	从远程 CPU 读数据
SFB 15	PUT	将数据写入远程 CPU
SFB 16	PRINT	将数据发送到打印机
SFB 19	START	在远程设备上开始一个暖重启或冷重启
SFB 20	STOP	将远程设备切换为 STOP 状态
SFB 21	RESUME	在远程设备上开始一个热重启
SFB 22	STATUS	查询远程伙伴的状态
SFB 23	USTATUS	接收远程设备的状态
SFB 29	HS_COUNT*	计数器(高速计数器, 集成功能)
SFB 30	FREQ_MES*	频率计(频率计, 集成功能)
SFB 31	NOTIFY_8P	生成没有确认指示的与块相关的消息
SFB 32	DRUM	执行序列发生器
SFB 33	ALARM	生成具有确认显示的与块相关的消息
SFB 34	ALARM_8	生成与块相关的消息, 其中没有用于表示 8 个信号的值
SFB 35	ALARM_8P	生成与块相关的消息, 其中带有用于表示 8 个信号的值
SFB 36	NOTIFY	生成没有确认显示的与块相关的消息
SFB 37	AR_SEND	发送归档数据
SFB 38	HSC_A_B*	计数器 A/B (集成功能)
SFB 39	POS*	定位(集成功能)
SFB 41	CONT_C <sup>1)</sup>	连续控制
SFB 42	CONT_S <sup>1)</sup>	步进控制
SFB 43	PULSEGEN <sup>1)</sup>	脉冲生成
SFB 44	ANALOG <sup>2)</sup>	使用模拟量输出定位
SFB 46	DIGITAL <sup>2)</sup>	使用数字量输出定位
SFB 47	COUNT <sup>2)</sup>	控制计数器
SFB 48	FREQUENC <sup>2)</sup>	控制频率测量
SFB 49	PULSE <sup>2)</sup>	控制脉宽调制
SFB 52	RDREC	读一条数据记录

编号	短名称	功能
SFB 53	WRREC	写一条数据记录
SFB 54	RALRM	接收中断
SFB 60	SEND_PTP <sup>2)</sup>	发送数据(ASCII、3964(R))
SFB 61	RECV_PTP <sup>2)</sup>	接收数据(ASCII、3964(R))
SFB 62	RES_RECV <sup>2)</sup>	删除接收缓冲区(ASCII、3964(R))
SFB 63	SEND_RK <sup>2)</sup>	发送数据(RK 512)
SFB 64	FETCH_RK <sup>2)</sup>	获取数据(RK 512)
SFB 65	SERVE_RK <sup>2)</sup>	接收和提供数据(RK 512)
SFB 73	RCVREC	接收数据记录
SFB 74	PRVREC	提供数据记录
SFB 75	SALRM	发送中断到 DP 主站
SFB 81	RD_DPAR	读预定义的参数
SFB 104	IP_CONF	设置 IP 组态

\* SFB 29 "HS\_COUNT"和 SFB 30 "FREQ\_MES"仅存在于 CPU 312 IFM 和 CPU 314 IFM 中。  
SFB 38 "HSC\_A\_B"和 SFB 39 "POS"仅存在于 CPU 314 IFM 上。详细描述，请参考/73/。

1) SFB 41 "CONT\_C"、42 "CONT\_S"和 43 "PULSEGEN"仅存在于 CPU 314 IFM 中。

2) FB 44 到 49 和 60 到 65 仅存在于 S7-300C CPU 中。

## 36.4 按字母排序的 SFB 列表

短名称	编号	功能
ALARM	SFB 33	生成具有确认的与块相关的消息
ALARM_8	SFB 34	生成与块相关的消息，其中没有用于表示 8 个信号的值
ALARM_8P	SFB 35	生成与块相关的消息，其中带有用于表示 8 个信号的值
ANALOG	SFB 44	使用模拟量输出定位
AR_SEND	SFB 37	发送归档数据
BRCV	SFB 13	接收分段数据
BSEND	SFB 12	发送分段数据
CONT_C <sup>1)</sup>	SFB 41	连续控制
CONT_S <sup>1)</sup>	SFB 42	步进控制
COUNT	SFB 47	控制计数器
CTD	SFB 1	递减计数器
CTU	SFB 0	递增计数器
CTUD	SFB 2	递增/递减计数器
DIGITAL	SFB 46	使用数字量输出定位
DRUM	SFB 32	执行序列发生器
FETCH_RK	SFB 64	接收数据(RK 512)
FREQ_MES <sup>*</sup>	SFB 30	频率计(频率计，集成功能)
FREQUENC	SFB 48	控制频率测量
GET	SFB 14	从远程 CPU 读数据
HSC_A_B <sup>*</sup>	SFB 38	计数器 A/B (集成功能)
HS_COUNT <sup>*</sup>	SFB 29	计数器(高速计数器，集成功能)
IP_CONF	SFB 104	设置 IP 组态
NOTIFY	SFB 36	生成没有确认显示的与块相关的消息
NOTIFY_8P	SFB 31	生成没有确认指示的与块相关的消息
POS <sup>*</sup>	SFB 39	定位(集成功能)
PRINT	SFB 16	将数据发送到打印机
PRVREC	SFB 74	提供数据记录
PULSE	SFB 49	控制脉宽调制
PULSEGEN <sup>1)</sup>	SFB 43	脉冲生成
PUT	SFB 15	将数据写入远程 CPU
RALRM	SFB 54	接收中断
RD_DPAR	SFB 81	读预定义的参数
RDREC	SFB 52	读一条数据记录
RECV_PTP	SFB 61	接收数据(ASCII、3964(R))
RCVREC	SFB 73	接收数据记录
RES_RECV	SFB 62	删除接收缓冲区(ASCII、3964(R))
RESUME	SFB 21	在远程设备上开始一个热重启
SALRM	SFB 75	发送中断到 DP 主站

短名称	编号	功能
SEND_PTP	SFB 60	发送数据(ASCII、3964(R))
SEND_RK	SFB 63	发送数据(RK 512)
SERVE_RK	SFB 65	接收和提供数据(RK 512)
START	SFB 19	在远程设备上开始一个暖重启或冷重启
STATUS	SFB 22	查询远程伙伴的状态
STOP	SFB 20	将远程设备切换为 STOP 状态
TOF	SFB 5	生成断开延迟
TON	SFB 4	生成接通延迟
TP	SFB 3	生成脉冲
URCV	SFB 9	数据的不对等的接收
USEND	SFB 8	数据的不对等的发送
USTATUS	SFB 23	接收远程设备的状态
WRREC	SFB 53	写数据记录

- 1) SFB 29 "HS\_COUNT"和 SFB 30 "FREQ\_MES"仅存在于 CPU 312 IFM 和 CPU 314 IFM 中。SFB 38 "HSC\_A\_B"和 SFB 39 "POS"仅存在于 CPU 314 IFM 上。详细信息，请参考/73/。
- 2) SFB 41 "CONT\_C"、42 "CONT\_S"和 43 "PULSEGEN"仅存在于 CPU 314 IFM 中。FC 列表

## 参考书目

- /30/使用入门：使用 STEP 7
- /70/手册：S7-300 可编程控制器，硬件与安装
- /71/参考手册：S7-300 可编程控制器，模块数据
- /72/指令列表：S7- 300 可编程控制器
- /101/参考手册：S7-400 可编程控制器，模块技术规范说明
- /102/指令列表：S7- 400 可编程控制器
- /231/手册：使用 STEP 7 配置硬件和通信连接
- /232/参考手册：S7-300 和 S7-400 的语句表(STL)
- /233/参考手册：S7-300 和 S7-400 的梯形图(LAD)
- /234/手册：使用 STEP 7 编程
- /236/参考手册：S7-300 和 S7-400 的功能块图(FBD)
- /250/手册：用于 S7-300 和 S7-400 编程的结构控制语言(SCL)
- /251/手册：用于 S7-300 和 S7-400 的 S7-GRAPH，顺序控制系统编程
- /252/手册：用于 S7-300 和 S7-400 的 S7-HiGraph，状态图编程
- /270/手册：用于 S7-300 和 S7-400 的 S7-PDIAG "组态 LAD、STL 和 FBD 的过程诊断"
- /350/用户手册：SIMATIC 7，标准控制器





# 词汇表

## 伴随值

生成消息时可随该消息一起输出以及提供与变量或/和地址状态有关信息的值。

## ACCU (累加器)

累加器是 CPU 中的寄存器，并作为缓冲区用于装载和传送操作，以及用于比较、算术和转换运算。

## 实际参数

当调用功能块(FB)或功能(FC)时，实际参数替换形式参数，例如，由实际参数"I 3.6"替换形式参数"REQ"。

## 地址

地址是存储位置或存储位置范围的标识符，例如：输入 I 12.1；位存储器 MW25；数据块 DB3。

## 寻址

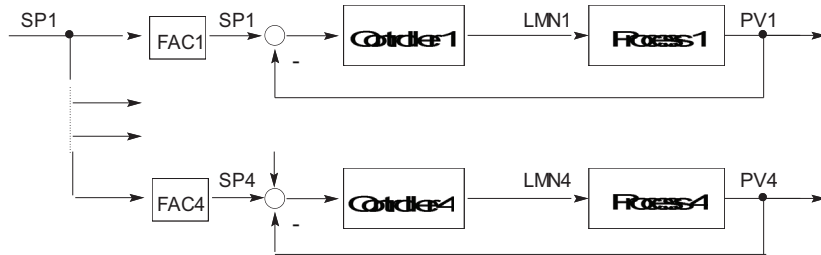
在用户程序中分配地址。可以将地址分配给存储位置或存储位置范围(例如：输入 I 12.1；位存储器 MW25)。

## 位存储器

这是 1 位的存储位置。位存储器允许通过 STEP 7 的基本操作进行读写访问(使用位、字节、字和双字进行寻址)。用户可使用位存储器的地址区来保存中间结果。

### 混合控制

混合控制涉及一个控制器结构，在该结构内，SP 总量的设定值被转换成单个组件的百分比。混合因子 FAC 之和必须为 1 (= 100 %)。



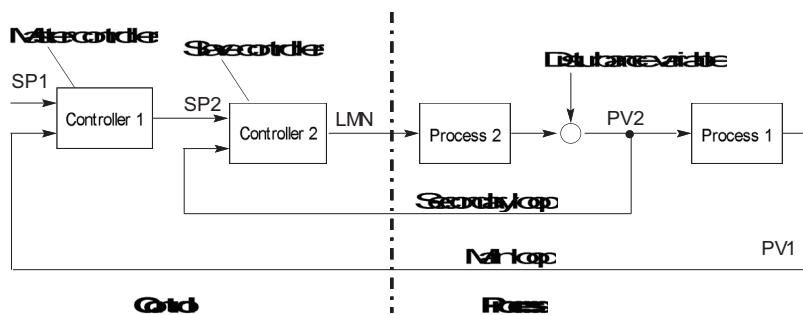
### 与块相关的消息

为具有消息能力的块(FB 或 DB)组态的消息。

### 串级控制

串级控制涉及一系列互连的控制器，在这些控制器中，主控制器根据主过程变量的瞬时误差信号调节二级(从站)控制器的设定值。

串级控制系统可通过包括附加过程变量进行改进。在适当的地点测量二级过程变量 PV2，并对其进行控制，使其接近参考设定值(主控制器的输出 SP2)。主控制器控制过程变量 PV1，使其接近固定的设定值 SP1，并设置 SP2，从而在无超调的情况下，尽快实现目标。



## 闭环控制器

闭环控制器是一种设备，在这种设备中，不断地计算误差信号，并生成一个调节信号，其目的是在不出现超调的情况下尽快消除误差信号。

## 通信，双向

使用通信 SFB 进行数据交换时，区别单向和双向通信。当本地和远程模块上都存在一个 SFB 时，为双向通信，例如，通信 SFB"USEND"和"URCV"。

## 用于已组态连接的通信 SFB

通讯 SFB 是用于数据交换和程序管理的系统功能块。

数据交换实例：SEND、RECEIVE、GET。

程序管理实例：将通信伙伴的 CPU 设置到 STOP 状态，查询通信伙伴的 CPU 状态。

## 用于未组态连接的通信 SFC

通信 SFC 是用于数据交换以及中止由通信 SFC 建立的现有连接的系统功能。

## 通信，单向

使用通信 SFB 进行数据交换时，区别单向和双向通信。当只有在本地模块上存在一个 SFB 时，为单向通信，例如 SFB"GET"。

## 完全重启

启动 CPU 时(例如，当模块选择开关从 STOP 切换到 RUN 或当接通电源时)，在开始处理循环程序之前(OB1)，首先处理组织块 OB101 (重新启动；只在 S7-400 中)或 OB100 (完全重新启动)。在完全重新启动时，读入过程映像输入表，然后从 OB1 的第一个语句开始处理 STEP 7 用户程序。

## 常数

"常数"是逻辑块中恒定数值的标记值。常数用于提高程序的可读性。例如，在功能块中输入标记值"Max\_iteration\_count"，而不是直接输入一个数值(例如，10)。然后在调用块时，输入常数的数值(例如，10)。

## 连续控制器

连续控制器是误差信号的每次改变都引起调节变量改变的控制器。这可以采用可调节变量范围内的任何数值。

## 控制设备

用于确定过程变量的全部设备。它包含一个控制器、一个受控设备(即，执行器)和一个传感器(测量设备)。

## 控制回路

控制闭环是过程输出(过程变量)和控制器输入之间、以及控制器输出(调节变量)和过程输入之间的连接，从而控制器和过程组成一个闭环。

### 控制系统的组消息

当在诊断缓冲区中输入一个标准诊断事件时，由 CPU 操作系统生成一个组消息。

### 控制器参数

控制器参数是控制器对给定环或过程特征进行响应时用于静态和动态调整的特征值。

### CPU 操作系统

CPU 操作系统组织未链接到特定控制任务的 CPU 的所有功能和过程。

### 数据块(DB)

数据块是用户程序中包含用户数据的区域。有所有逻辑块均可访问的共享数据块，还有同特定功能块 (FB)调用相关的实例数据块。

### 诊断缓冲区

诊断缓冲区是 CPU 中的一个存储器区域，所有诊断事件按照其发生的先后顺序存储在该区域中。

### 诊断数据

诊断数据是指错误消息中包含的信息(诊断事件、时间标志)。

### 诊断条目

诊断事件在诊断缓冲区中使用诊断条目进行描述。

### 诊断中断

诊断模块使用发往 CPU 的诊断中断来报告识别出的系统错误。

### 诊断消息

诊断消息由已处理的诊断事件组成，并从 CPU 发往显示单元。

### 诊断

诊断功能集成所有系统诊断，并包括在 PLC 内对错误的识别、解释和报告。

### 显示设备

用于显示过程结果的设备。

### 异步错误

异步错误是运行系统错误(例如，电源错误、超出扫描时间)，它们不会在用户程序中分配给任何特定位置。发生这些错误时，操作系统调用相应的组织块，在这些组织块中，用户可以编程响应。

## 处理 OB 时的错误

如果系统程序识别出一个特定错误(例如, S7 中的访问错误), 它将会调用指定的组织块, 在该组织块中 CPU 对错误的响应可由用户程序来设置。

## 错误 OB

错误 OB 是用户可用于编程错误响应的组织块。然而, 只有在错误不引起 PLC 停止时, 才能编程错误响应。每种类型的错误都有一个错误 OB。(例如, 用于寻址错误的错误 OB、用于 S7 中的访问错误的错误 OB。)

## 错误反应

对运行系统错误的响应, 操作系统可以以下列方式作出响应通过将 PLC 切换到 STOP 状态、通过调用一个用户可以编程响应的组织块或通过显示错误。

## 同步错误

同步错误是分配给用户程序中特定位置的运行系统错误(例如, 访问 I/O 模块错误)。发生这些错误时, 操作系统调用相应的组织块, 在这些组织块中, 用户可以编程响应。

## 系统错误

系统错误是可能在 PLC 内发生的错误(不位于过程中)。例如, 系统错误可以为 CPU 中的程序错误和模块中的故障。

## 形式参数

形式参数是逻辑块中实际参数的占位符, 可以为其分配参数。在 FB 和 FC 中, 由用户声明形式参数; 在 SFB 和 SFC 中, 已存在形式参数。调用块时, 将实际参数分配给形式参数, 从而被调用的块使用最新的数值。形式参数属于块的本地数据, 被声明为输入、输出和进入/离开参数。

## 组错误

由模块前面板上的 LED 显示所指示的错误消息(仅限在 S7-300 中)。只要所涉及的模块中存在错误(内部错误和外部错误), LED 即点亮。

## 硬件中断

硬件中断是由于过程中的特殊事件, 由具有中断能力的模块触发。将硬件中断报告给 CPU。然后根据该中断的优先级处理已分配的组织块。

## 输入参数

输入参数只位于功能和功能块中。通过输入参数, 将数据传送给被调用块, 进行处理。

## 指令

指令(STEP 5 或 STEP 7)是以文本语言创建的程序的最小组成部分。它代表处理器的命令。

## 积分组件

控制器的积分组件。

在过程变量(或误差信号)中发生一次步进变化后,输出变量按与积分动作因子  $K_I (= 1/T_I)$ 成比例的一个变化率、随时间按斜坡函数改变。闭环控制中的积分组件可以更正控制器输出变量,直到误差信号为零为止。

## 集成控制器

集成控制器是一个可在操作系统中使用的即时编程的控制器块,包含闭环控制应用的最重要功能。用户可使用软开关选择和取消选定功能。

## 完全重启

CPU 的启动模式。下面信息适用于完全重新启动模式:从工作存储器中删除由 SFC 生成的数据块,且给所有其它数据块分配它们在装载存储器中的缺省值。还将复位过程映像、定时器、计数器和存储位 - 与剩余的存储器组态无关。在完全重新启动期间,CPU 执行 OB 102、读取过程映像输入表,然后从第一个 OB 1 指令开始继续处理用户程序。

## 中断

SIMATIC S7 优先级系统识别 10 种不同的优先级,调整处理用户程序。中断属于这些优先级,例如,硬件中断。发生中断时,操作系统自动调用一个组织块,在该组织块中,用户可编程所要求的响应(例如,在一个功能块中)。

## 时间中断

时间中断属于 SIMATIC S7 程序执行中的优先级之一。在指定的日期(或天)和时间(例如,9:50,或每小时或每分钟)生成时间中断。然后执行一个相应的组织块。

## 中断,延时

延时中断属于 SIMATIC S7 程序执行中的优先级之一。当定时器在用户程序中到时后,将生成延时中断。然后执行一个相应的组织块。

## 逻辑块

在 SIMATIC S7 中,逻辑块是包含部分 STEP 7 用户程序的块。其它类型的块是只包含数据的数据块。下面的列表给出了逻辑块的类型:

- 组织块(OB)
- 功能块(FB)
- 功能(FC)
- 系统功能块(SFB)
- 系统功能(SFC)

## 消息

一个事件的发生报告。消息可以输出到一个有合适组态的显示设备，包含诸如优先级、位置和消息事件时间之类的信息，以及关于状态转变(进入状态/离开状态)的信息。

## 消息组态

消息组态指创建消息和消息模板以及编辑它们的文本和属性，包括与块相关的消息、与符号相关的消息和诊断消息之类的消息。

## 消息号

配给一条消息的唯一编号，用于识别该消息，例如用于确认。

## 模块参数

模块参数是可设置模块特性的数值。根据特定的模块，可以在用户程序中修改某些参数。

## OB1

组织块 OB1 是系统程序接口，用于循环程序处理。

## OB 优先级

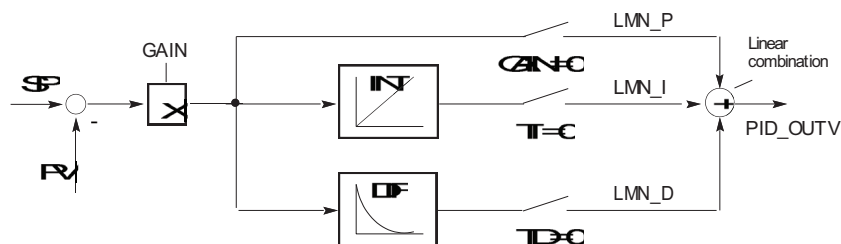
CPU 操作系统区别各种优先级，例如，循环程序处理、硬件中断受控程序处理。将组织块(OB)分配给每个优先级，在组织块中，S7 用户可以编程响应。OB 具有不同的优先级，允许在同时出现两个 OB 时按正确的顺序进行处理，并允许高优先级的 OB 中断低优先级的 OB。S7 用户可以改变标准优先级。

## 组织块(OB)

组织块可作为 CPU 操作系统和用户程序之间的接口。在组织块中指定处理用户程序的顺序。

## 并行结构

并行结构是控制器中的一种特殊类型的信号处理(数学处理)方式。P、I 和 D 组件相互并行计算，它们互不影响，最后进行汇总。



## 参数

1. 参数是 S7 逻辑块的变量

(参见块参数、实际参数、形式参数)

2. 用于设置模块特性的变量

(每个模块有一个或多个变量)

从工厂发货时，每个可组态的模块都有基本的参数设置，但可以使用 STEP 7 改变该参数设置。

(每个模块有一个或多个参数设置)

有两种参数类型：

静态和动态参数(参数，静态/参数，动态)。

## 参数，动态

与静态参数相反，模块的动态参数可在操作期间，通过调用 SFC 由用户程序改变，例如，模拟量模块的限制值。

## 参数，静态

与动态参数相反，模块的静态参数不能由用户程序改变，只能使用 STEP 7 改变，例如，数字量输入模块的输入延迟。

## P 算法

用于计算输出信号的算法，在该算法中，误差信号和调节变量改变之间存在比例关系。特征：过程中不使用稳态误差信号，包括死区时间。

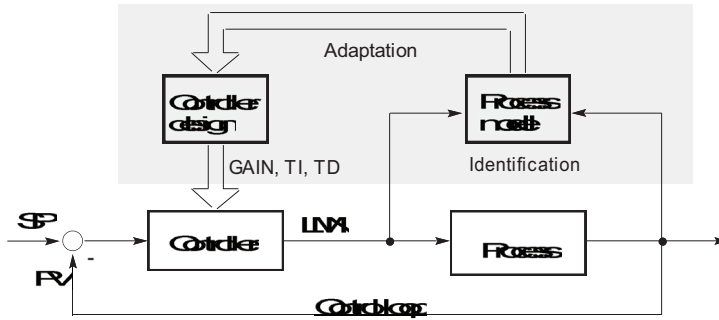
## PI 算法

用于计算输出信号的算法，在该算法中，调节变量的改变由一个与误差信号成比例的要素以及一个与误差信号和时间成比例的 I 要素组成。特征：没有稳态误差信号，比 I 算法更快速进行补偿，适用于所有过程。



### PID 算法

用于计算输出信号的算法，输出信号由误差信号的倍数、积分和微分组成。PID 算法是一个并行结构。  
特征：如果过程的死区时间不大于其它时间常数，那么可实现高质量控制。



### 优先级

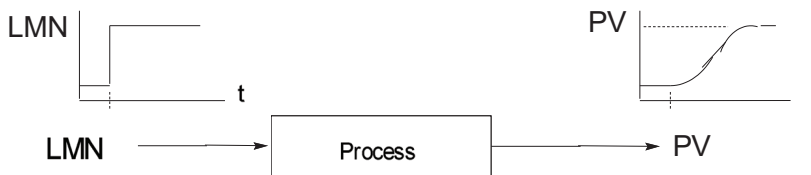
当给组织块分配优先级时，确认当前正在使用的用户程序的中断能力，从而使高优先级的事件可以中断低优先级的事件。

### 优先级

CPU 的操作系统最多具有 28 个优先级，给这些优先级分配各种组织块。优先级确认哪个 OB 可以中断其它 OB。当一个优先级包含一个以上 OB 时，它们不能相互中断，而是按顺序执行。

### 过程

过程是系统的一部分，在过程内过程变量受调节变量影响(通过改变能量级或质量)。过程分成执行器和被控制的实际过程。



### 程序执行、事件控制

对于事件控制的程序执行，由启动事件(优先级)中断周期性用户程序的运行。发生启动事件时，在下一个指令前中断当前正在执行的块，然后调用并执行一个已分配的组织块。之后，从中断点继续执行周期性程序。

### 比例执行器

脉宽调制

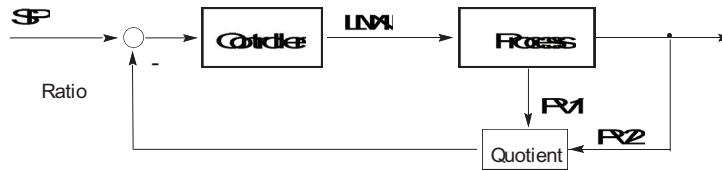
### 脉宽调制

脉宽调制是影响不连续输出的调节变量的一种方法。以百分比表示的已计算调节值在调节变量输出处被转换成一个比例信号脉冲时间  $T_p$ ，例如， $100\% T_p = T_A$  或= 周期。

## 比例控制

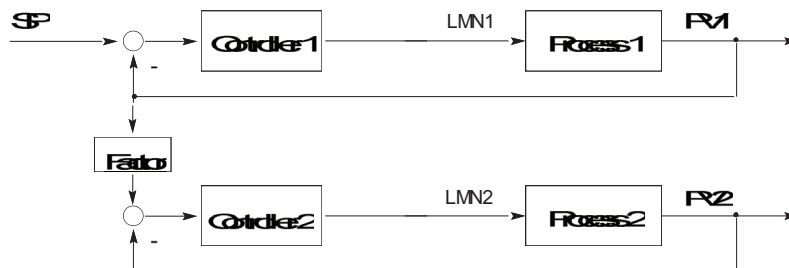
- 单环比例控制器

当两个过程变量的比率比各个变量的绝对值更重要时，使用单环比例控制器。



- 多环比例控制器

在多环比例控制器中，两个过程变量 PV1 和 PV2 的比率必须保持常数。为此，根据第 1 个控制环的过程变量计算第 2 个控制环的设定值。即使过程变量 PV1 动态变化，比率也保持不变。



## 远程设备

远程设备是可在网络上到达的设备，例如，打印机或计算机。与本地设备相反，安装时，必须给它们分配一个网络地址。

## 重新启动

启动 CPU 时(例如，当模式选择开关从 STOP 切换到 RUN 或当接通电源时)，在周期性程序处理开始之前(OB1)，首先处理组织块 OB100 (重新启动)或组织块 OB101 (重新启动；只在 S7-400 中)或组织块 OB 102 (完全重新启动)(完全重新启动)。重新启动时，将读取过程映像输入表，并且会在 STEP 7 用户程序处理被上一次停止(STOP，断电)中断的位置重新启动它。

## 逻辑运算的结果(RLO)

逻辑运算结果(RLO)是处理器中的当前信号状态，用于进一步的二进制信号处理。上一个 RLO 的信号状态决定是否执行了某些运算。

## 运行系统错误

在 PLC 中执行用户程序期间发生的错误(不位于过程中)。

## 扫描

集成在 CPU 中的一个操作系统功能，用于以设定的时间间隔扫描、检测信号，以确认信号是否发生改变。

## 标准功能

标准功能是由西门子提供的功能块，用于执行复杂任务。

## 标准功能块

标准功能块是由西门子提供的功能块，用于执行复杂任务。

## 启动事件

启动事件是已定义的事件，例如错误或中断，提示操作系统调用适当的组织块。

## 启动事件信息

启动事件信息是组织块(OB)的一部分。启动事件信息给 S7 用户提供关于触发 OB 调用的事件的详细信息。启动事件信息包含事件编号(由事件类别和事件 ID 组成)、事件的时间戳以及附加信息(例如，由中断激活的信号模块的地址)。

## 启动信息

当操作系统调用一个组织块时，操作系统传送可在用户程序中解释的启动信息。

## 启动 OB

根据启动模式选择器的设置(只对 S7-400)，启动(断电后重新接通电源、通过模式选择器或来自编程设备的命令手动从 STOP 切换到 RUN)原因是操作系统调用了"完全重新启动"启动组织块，或调用了"重新启动"启动组织块(只位于 S7-400)。在启动 OB 中，例如，SIMATIC S7 用户可以编程在断电后，如何重新启动系统。

指令(STEP 5 或 STEP 7)是以文本语言创建的程序的最小组成部分。它代表处理器的命令。

## 语句表

语句表是 STEP 7 的汇编语言。在语句表中处理程序时，单个指令与 CPU 处理程序的顺序一致。

## STEP 7

用于创建 SIMATIC S7 控制器用户程序的编程软件。

## STEP 7 编程语言

SIMATIC S7 控制器的编程语言。S7 程序可以不同的表达类型使用 STEP 7 : a) 语句表、b) 控制系统流程图、c) 梯形图逻辑。

## 步进控制器

步进控制器是一个具有不连续输出的伪连续控制器(以及带 I 动作的电机驱动的执行器)。执行器有三步响应,例如,向上 - 停止 - 向下(或打开 - 保持 - 关闭)。

(三步控制器)。

## STL

语句表。

## 子编号

当消息块可以监视一个以上信号时,待监视信号的编号。

## 与符号相关的消息

在为符号表中的符号(输入、输出、位存储器、数据块)组态消息时所使用的消息。在组态期间,必须为用于监视信号的 SCAN 功能设置时间间隔。

## 符号编程

STEP 7 编程语言允许用符号名称代替 STEP 7 地址。例如,STEP 7 地址"Q 1.1"可用"阀 17"代替。STEP 7 中的符号列表还会创建地址和已分配的符号名之间的链接。

## 系统诊断

检测和评估系统诊断事件。

## 系统诊断事件

在 CPU 的诊断缓冲区中输入的条目,用于初始化操作系统。

## 系统功能(SFC)

系统功能(SFC)是集成在 CPU 操作系统中的功能,可按要求在 STEP 7 用户程序中调用。

## 系统功能块(SFB)

系统功能块(SFB)是集成在 CPU 操作系统中的功能块,可按要求在 STEP 7 用户程序中调用。

## 三步控制器

一种只能采用三个离散状态的控制器;例如,"加热 - 关闭 - 冷却"或"右 - 停止 - 左"(步进控制器)。

## 延时中断

延时中断是进行 SIMATIC S7 程序处理时所使用的优先级之一。在指定的时间用完后，在用户程序中生成该中断，并在相关的组织块中进行处理。

## 工具

用于组态和编程的软件部件。

## 两步控制器

两步控制器是只能为调节变量设置两种状态的控制器(例如，打开 - 关闭)。

## 用户自定义诊断

检测和评估用户自定义诊断事件。

## 自定义的诊断事件

由用户检测的、可输入到诊断缓冲区(使用 SFC 52)的诊断事件。

## 自定义诊断消息

报告发生自定义诊断事件的消息。

## 用户程序

用户程序包含所有语句和声明，以及用于信号处理的数据，通过这些数据，可以控制一个系统或过程。用户程序被分配给一个可编程模块(模块，可编程)，且可使用称为块的较小单元进行构造。

## 用户程序错误

在 SIMATIC S7 PLC 中处理用户程序期间发生的错误(与过程错误相反)。操作系统使用错误 OB (优先级系统)以及系统功能的状态字和输出参数来处理错误。

## 变量

变量定义一种可在 STEP 7 用户程序中使用的、具有可变内容的数据。变量包含地址(例如，M 3.1)和数据类型(例如，BOOL)，由一个符号表示(例如，MOTOR\_ON)。

## 变量声明

变量声明包括符号名、数据类型条目，可能还会有默认值、地址和注释条目。

# 索引

## 字母数字

- "DPRD\_DAT", 289
- ACCFLT\_ESR, 217
- ACCFLT\_MASKED, 215, 216
- ACCFLT\_QUERY, 217
- ACCFLT\_RESET\_MASK, 216
- ACCFLT\_SET\_MASK, 215
- ACT\_TINT, 199
- AD\_DT\_TM, 490
- ADC/DAC 错误, 610
- ALARM, 447
- ALARM\_8, 453
- ALARM\_8P, 450
- ALARM\_D, 470
- ALARM\_DQ, 470
- ALARM\_S, 466
- ALARM\_SC, 469
- ALARM\_SQ, 466
- AR\_SEND, 455
- BCD 码转换出错, 209
- BLK, 101
- BLKMOV, 96
- BRCV, 358
- BSEND, 354
- BVAL, 100
- C\_DIAG, 244
- CAN\_DINT, 207
- CAN\_TINT, 198
- CDT, 134
- CiR, 122, 123
- CiR 操作:, 54
- COMPRESS, 108
- CONCAT, 495
- CONT\_C**, 510
- CONT\_S, 514
- CONTROL, 383
- COUNT, 103
- CPU, 119, 219, 220, 620, 621
  - 使用 SFC 46 STP 进入停机状态, 119
  - 特征, 620
  - 硬件故障, 220
- CPU 417 和 CPU 417H 的访问错误, 209
- CPU 417 和 CPU 417H 的访问错误屏蔽, 209
- CPU 和所用的协议变量(connection\_type)以及可传送数据长度之间的关系, 418
- CPU 冗余错误 OB (OB72), 41
- CPU 硬件故障 OB, 59
- CPU 硬件故障组织块(OB84), 59
- CQ, 144
- CREA\_DB, 117
- CREA\_DBL, 110
- CREATE\_DB, 103
- CTD, 482
- CTRL\_RTM, 143
- CTU, 481
- CTUD, 483
- CV, 144
- D\_TOD\_DT, 490
- DB\_NUMBER, 103
- DEL\_DB, 105
- DEL\_SI, 475
- DELETE, 495
- DI\_STRNG, 502
- DIS\_AIRT, 224
- DIS\_IRT, 221
- DIS\_MSG, 457
- DMSK\_FLT, 216
- DP\_PRAL, 273
- DP\_TOPOL, 248
- DPNRM\_DG, 286
- DPRD\_DAT, 289
- DPSYC\_FR, 276
- DPWR\_DAT, 292
- DP 从站组, 276
  - 同步, 276

- DP 主站系统信息, 653  
DRUM, 261  
DSTBLK, 96  
DT\_DATE, 490  
DT\_DAY, 491  
DT\_TOD, 491  
DTIME, 205  
EN\_AIRT, 225  
EN\_IRT, 223  
EN\_MSG, 459  
EQ\_DT, 493  
EQ\_STRNG, 499  
EVENTN, 236  
FB 12 BSEND, 352  
FB 13 BRCV, 356  
FB 15 PUT, 359  
FB 20 GETIO, 316  
FB 21 SETIO, 317  
FB 210 "FW\_TCP", 437  
FB 22 GETIO\_PART, 318  
FB 220 "FW\_IOT", 438  
FB 23 SETIO\_PART, 320  
FB 63 "TSEND", 424  
FB 64 "TRCV", 427, 428, 429  
FB 65 "TCON", 419  
FB 66 "TDISCON", 422  
FB 67 "TUSEND", 431  
FB 68 "TURCV", 434, 435  
FB 8 USEND, 345  
FB 9 URCV, 349  
FC1, 490  
FC10, 499  
FC11, 495  
FC12, 493  
FC13, 499  
FC14, 493  
FC15, 500  
FC16, 502  
FC17, 496  
FC18, 494  
FC19, 500  
FC2, 495  
FC20, 496  
FC21, 496  
FC22, 504  
FC23, 494  
FC24, 501  
FC25, 504  
FC26, 497  
FC27, 505  
FC28, 494  
FC29, 501  
FC3, 490  
FC30, 502  
FC31, 497  
FC32, 498  
FC33, 491  
FC34, 492  
FC35, 492  
FC36, 506  
FC37, 503  
FC38, 503  
FC39, 503  
FC4, 495  
FC40, 492  
FC5, 502  
FC6, 490  
FC7, 491  
FC8, 491  
FC9, 493  
FILL, 100  
FIND, 495  
FW\_IOT, 438  
FW\_TCP, 437  
GADR\_LGC, 263  
GD\_RCV, 326  
GD\_SND, 323  
GD 信息包, 323, 324, 325, 326, 327, 328  
    通过 SFC 60 发送, 323  
    通过 SFC 61 编程接受, 326  
GE\_DT, 493  
GE\_STRNG, 499  
GEO\_LOG, 268  
GET, 363  
GETIO, 316  
GETIO\_PART, 318



- GET 和 PUT SFC 的数据一致性, 331
- GT\_DT, 493
- GT\_STRNG, 500
- H/F 事件, 684
- H\_CTRL, 597
- I/O 访问错误, 209
- I/O 访问错误组织块(OB122), 84
- I/O 区域中的位域, 258  
    使用 SFC 79 设置, 258
- I/O 冗余错误 OB (OB70), 39
- I\_ABORT, 404
- I\_GET, 400
- I\_PUT, 402
- I\_STRNG, 502
- IEC 功能的技术数据, 487
- INFO1, 236
- INFO2, 238, 239
- INSERT, 496
- IOID, 151, 153, 157, 159
- IP\_CONF, 303
- LADDR, 151, 153, 157, 159
- LE\_DT, 494
- LE\_STRNG, 500
- LEFT, 496
- LEN, 496
- LGC\_GADR, 264
- LIMIT, 504
- LOG\_GEO, 270
- LOW\_LIMIT, 103
- LT\_DT, 494
- LT\_STRNG, 501
- MAX, 504
- MID, 497
- MIN, 505
- MODE, 221, 223
- MP\_ALM, 121
- MSK\_FLT, 215
- M 短路, 612, 613, 614  
    模拟量输出模块, 612  
    模拟量输入模块, 612
- NE\_DT, 494
- NE\_STRNG, 501
- NOTIFY, 443
- NOTIFY\_8P, 445
- NR, 142, 143, 144
- OB), 54
- OB\_NR, 197, 198, 199, 200, 205, 206, 207, 222, 223
- OB\_RT, 240
- OB10 到 OB17, 16
- OB100  
    OB101  
    和 OB102, 76
- OB121, 81, 84
- OB20 到 OB23, 20
- OB30 到 OB38, 22
- OB40 到 OB47, 24
- OB55, 26, 27
- OB56, 29, 30
- OB57, 32, 33
- OB60, 35, 36, 121
- OB70, 39, 40
- OB72, 41, 43
- OB73, 44
- OB80, 45
- OB81, 49
- OB82, 51
- OB83, 54, 55, 56
- OB84, 59
- OB85, 61
- OB86, 65, 66, 67
- OB87, 70
- OB 的本地数据, 627
- P sh, 612
- PERIOD, 197
- PN\_DP, 300
- PN\_IN, 298
- PN\_OUT, 299
- PRGFLT\_ESR, 217
- PRGFLT\_MASKED, 215, 216
- PRGFLT\_QUERY, 217
- PRGFLT\_RESET\_MASK, 216
- PRGFLT\_SET\_MASK, 215
- PRINT, 369
- PROFINET CBA 组件, 298, 299, 300  
    更新互连, 300  
    更新用户程序接口输入, 298

- PROFINET 接口, 296, 297
- PROTECT, 126
- PRVREC, 130
- PULSEGEN, 521, 522, 527
- PULSEGEN 块的实例, 532
- PUT, 359
- PV, 142
- P 短路, 612, 613, 614
  - 模拟量输出模块, 612
  - 模拟量输入模块, 612
- QRY\_DINT, 206
- QRY\_TINT, 200
- R\_STRNG, 502
- RALRM, 171
- RAM 错误, 610
- RCVREC, 127
- RD\_DPAR, 164
- RD\_DPARA, 150
- RD\_LGADR, 267
- RD\_REC, 159
- RD\_SINFO, 227
- RDREC, 167
- RDSYSST, 230, 615
- RE\_TRIGR, 119
- READ\_CLK, 134
- READ\_DBL, 113
- READ\_ERR, 217
- READ\_RTM, 144
- READ\_SI, 472
- Receiving segmented data, 356
  - 通过 FB 13, 356
  - 通过 SFB 13, 356
- RECNUM, 151, 153, 157, 159
- RECORD, 151, 152, 157, 158
- REPL\_VAL, 109
- REPLACE, 497
- RESUME, 377
- RIGHT, 498
- RTM, 140
- S, 143
- S5TI\_TIM, 491
- S7 通讯块 SFB 的分类和对工作存储区的要求, 332
- S7 通讯块和 S7 基本通讯块之间的差别, 329
- SALRM, 187
- SB\_DT\_DT, 492
- SB\_DT\_TM, 492
- SDT, 197
- SEL, 506
- SEND, 236
- SET\_CLK, 133
- SET\_CLKS, 136
- SET\_RTM, 142
- SET\_TINT, 197
- SETIO, 317
- SETIO\_PART, 320
- SFB, 338, 341, 342, 343, 383
  - 参数分类, 337
  - 查询属于 SFB 实例的连接的状态, 383
  - 对出错和故障的反应, 343
  - 启动反应, 341
- SFB 0 CTU, 481
- SFB 1 CTD, 482
- SFB 12 BSEND, 352
- SFB 13 BRCV, 356
- SFB 14 GET, 363
- SFB 15 PUT, 359
- SFB 16 PRINT, 366
- SFB 19 START, 372
- SFB 2 CTUD, 483
- SFB 20 STOP, 375
- SFB 21 RESUME, 377
- SFB 22 STATUS, 379
- SFB 23 "USTATUS", 381, 382
- SFB 23 USTATUS, 381
- SFB 29 (HS\_COUNT), 601
- SFB 3 TP, 477
- SFB 30 (FREQ\_MES), 603
- SFB 31 NOTIFY\_8P, 445
- SFB 32 DRUM, 260
- SFB 33 ALARM, 447
- SFB 34 ALARM\_8, 453
- SFB 35 ALARM\_8P, 450
- SFB 36 NOTIFY, 443
- SFB 37 AR\_SEND, 455
- SFB 38 (HSC\_A\_B), 604
- SFB 39 (POS), 605

- SFB 4 TON, 479
- SFB 41 CONT\_C, 507
- SFB 42 CONT\_S, 514
- SFB 43 PULSEGEN, 520
  - 非对称三步控制, 526
  - 两步控制, 520
  - 三步控制, 524, 526, 528, 529, 531
  - 自动同步, 523
- SFB 44, 535
- SFB 46, 546
- SFB 47, 560
- SFB 48, 564
- SFB 49, 567
- SFB 5 TOF, 480
- SFB 52 RDREC, 167
- SFB 53 WRREC, 169
- SFB 54 RALRM, 171
- SFB 60, 569, 590
  - 附加错误信息, 590
- SFB 60 至 65 的补充错误信息, 590
- SFB 61, 572, 590
  - 附加错误信息, 590
- SFB 62, 575, 590
  - 附加错误信息, 590
- SFB 63, 577, 590
  - 附加错误信息, 590
- SFB 64, 581, 590
  - 附加错误信息, 590
- SFB 65, 586, 590
  - 附加错误信息, 590
- SFB 73 RCVREC, 127
- SFB 74 PRVREC, 130
- SFB 75 SALRM, 187
- SFB 8 USEND, 345
- SFB 81, 164
- SFB 9 URCV, 349
- SFB ANALOG, 535
- SFB COUNT, 557
- SFB DIGITAL, 546
- SFB FETCH RK, 581
- SFB FREQUENC, 562
- SFB PULSE, 566
- SFB RCV\_PTP, 572
- SFB RES\_RCVB, 575
- SFB SEND\_PTP, 569, 577
- SFB SERVE\_RK, 586
- SFB104 IP\_CONF, 303
- SFB 列表
  - 按数字排序, 699
  - 按字母排序, 701
- SFC 0 SET\_CLK, 133
- SFC 1 READ\_CLK, 134
  - 参数, 134
- SFC 10 DIS\_MSG, 457
- SFC 100 SET\_CLKS, 136
- SFC 101 RTM, 140
- SFC 102 RD\_DPARA, 150
- SFC 103 DP\_TOPOL, 248
- SFC 104 CiR, 122
- SFC 105 READ\_SI, 472
- SFC 106 DEL\_SI, 475
- SFC 107 ALARM\_DQ, 470
- SFC 108 ALARM\_D, 470
- SFC 109 "PROTECT", 124
- SFC 11 DPSYC\_FR, 276
- SFC 112 "PN\_IN", 295, 298
  - 背景信息, 295
- SFC 112 至 114 上的背景信息, 295
- SFC 113 "PN\_OUT", 295, 299
  - 背景信息, 295
- SFC 114 "PN\_DP", 295, 300
  - 背景信息, 295
- SFC 12 D\_ACT\_DP, 281
- SFC 126 SYNC\_PI, 254
- SFC 127 SYNC\_PO, 256
- SFC 13 DPNRM\_DG, 286
- SFC 14 DPRD\_DAT, 289
- SFC 15 DPWR\_DAT, 292
- SFC 17 ALARM\_SQ, 466
- SFC 18 ALARM\_S, 466
- SFC 19 ALARM\_SC, 469
- SFC 2 SET\_RTM, 142
- SFC 20 BLKMOV, 95
- SFC 21 FILL, 100
- SFC 22 CREAT\_DB, 103
  - 错误信息, 103

- SFC 22 CREATE\_DB, 103
- SFC 23 DEL\_DB, 105  
    错误信息, 105
- SFC 25 COMPRESS, 108
- SFC 26 UPDAT\_PI, 251
- SFC 27 UPDAT\_PO, 253
- SFC 28 SET\_TINT, 197
- SFC 28 到 31 的特征, 195
- SFC 29 CAN\_TINT, 198
- SFC 3 CTRL\_RTM, 143
- SFC 30 ACT\_TINT, 199
- SFC 31 QRY\_TINT, 200
- SFC 32 SRT\_DINT, 205
- SFC 33 CAN\_DINT, 207
- SFC 34 QRY\_DINT, 206  
    错误信息, 206
- SFC 35 MP\_ALM, 121
- SFC 36 MSK\_FLT, 215
- SFC 37 DMSK\_FLT, 216
- SFC 38 READ\_ERR, 217
- SFC 39 DIS\_IRT, 221
- SFC 4 READ\_RTM, 144
- SFC 40 EN\_IRT, 223  
    错误信息, 223
- SFC 41 DIS\_AIRT, 224  
    返回值, 224
- SFC 42 EN\_AIRT, 225  
    错误信息, 225  
    **返回值**, 225
- SFC 43 RE\_TRIGR, 119
- SFC 44 REPL\_VAL, 109
- SFC 46 STP, 119
- SFC 46 TIME\_TCK, 145
- SFC 47 WAIT, 120
- SFC 48 SNC\_RTCB, 135
- SFC 49 LGC\_GADR, 264
- SFC 5 GADR\_LGC, 263
- SFC 50 RD\_LGADR, 267
- SFC 51 RDSYSST, 230, 615
- SFC 52 WR\_USRMSG, 236
- SFC 55 WR\_PARM, 151
- SFC 55 到 59 的进一步故障信息, 163
- SFC 56 WR\_DPARM, 153
- SFC 58 WR\_REC, 157  
    参数, 157
- SFC 59 RD\_REC, 159
- SFC 6 RD\_SINFO, 227
- SFC 60 GD\_SND, 323
- SFC 61 GD\_RCV, 326
- SFC 62 CONTROL, 383
- SFC 63 (AB\_CALL), 607
- SFC 64 TIME\_TICK, 145  
    参数, 145
- SFC 65 X\_SEND, 390
- SFC 66 X\_RCV, 391
- SFC 67 X\_GET, 397
- SFC 68 X\_PUT, 395
- SFC 69 X\_ABORT, 399
- SFC 7 DP\_PRAL, 273
- SFC 70, 268
- SFC 71, 270
- SFC 72 I\_GET, 400
- SFC 73 I\_PUT, 402
- SFC 74 I\_ABORT, 404
- SFC 78 "OB\_RT", 240
- SFC 79 SET, 258
- SFC 80 RSET, 259
- SFC 82 CREA\_DBL, 110
- SFC 83 READ\_DBL, 113
- SFC 84 WRIT\_DBL, 115
- SFC 85 CREA\_DB, 117
- SFC 87 C\_DIAG, 244
- SFC 9 EN\_MSG, 459
- SFC 90 H\_CTRL, 597
- SFC99, 301, 302
- SFC 列表  
    按数字排序, 693  
    按字母排序, 696
- SIGN, 205
- SNC\_RTCB, 135
- SRCBLK, 95
- SRT\_DINT, 205
- SSL ID W#16#xy25 - 将过程映像分区分配到 OB, 637
- SSL ID 为 W#16#0132、索引为 W#16#000B 的部分列表摘录的数据记录, 644

- SSL ID 为 W#16#0132、索引为 W#16#000C 的部分列表摘录的数据记录, 645
- SSL\_HEADER, 230
- SSL-ID, 617
- SSL-ID W#16#00B1 - 模块诊断信息, 668
- SSL-ID W#16#00B2 - 对应物理地址的诊断数据记录 1, 670
- SSL-ID W#16#00B3 - 对应逻辑基址的模块诊断数据, 671
- SSL-ID W#16#00B4 - DP 从站的诊断数据, 672
- SSL-ID W#16#0x75 - H 系统中切换的 DP 从站/IO 设备, 651
- SSL-ID W#16#xy11 - 模块标识, 619
- SSL-ID W#16#xy12 - CPU 特征, 620
- SSL-ID W#16#xy13 - 存储器区, 623
- SSL-ID W#16#xy14 - 系统区域, 624
- SSL-ID W#16#xy15 - 块类型, 626
- SSL-ID W#16#xy1C - 组件标识, 629
- SSL-ID W#16#xy22 - 中断状态, 635
- SSL-ID W#16#xy32 - 通讯状态数据, 640
- SSL-ID W#16#xy71 - H CPU 组信息, 648
- SSL-ID W#16#xy74 - 模块 LED 的状态, 627
- SSL-ID W#16#xy90 - DP 主站系统信息, 653
- SSL-ID W#16#xy91 - 模块状态信息, 655
- SSL-ID W#16#xy92 - 机架/站状态信息, 661
- SSL-ID W#16#xy95 - 扩展的 DP 主站系统信息, 665
- SSL-ID W#16#xyA0 - 诊断缓冲区, 667
- SSL-ID 为 W#16#0132、索引为 W#16#0005 的部分列表摘录的数据记录, 641
- SSL-ID 为 W#16#0132、索引为 W#16#0008 的部分列表摘录的数据记录, 642
- SSL-ID 为 W#16#0232、索引为 W#16#0004 的部分列表摘录的数据记录, 646
- START, 372, 373
- STATUS, 200, 206, 379, 380
- STEP 7, 45, 49, 52, 61, 62, 81, 84
- OB 的类型
  - OB121, 81
  - OB122, 84
  - OB80, 45
  - OB81, 49
  - OB82, 51
  - OB85, 61
- STOP, 375, 376
- 将远程设备切换到, 375
- STP, 119
- STRNG\_DI, 503
- STRNG\_I, 503
- STRNG\_R, 503
- SYNC\_PI, 254
- SYNC\_PO, 256
- TCON, 421
- TDISCON, 422
- TIM\_S5TI, 492
- TIME\_TCK, 145
- Time-of-Day 功能, 490, 491, 492
- TOF, 480
- TON, 479
- TP, 477
- TRCV, 427
- TSEND, 424
- TURCV, 434
- TUSEND, 431
- UP\_LIMIT, 103
- UPDAT\_PI, 251
- UPDAT\_PO, 253
- URCV, 349
- USEND, 345
- USTATUS, 381
- WAIT, 120
- WR\_DPARM, 153
- WR\_PARM, 151
- WR\_REC, 157
- WR\_USRMSG, 236
- WRIT\_DBL, 115
- WRREC, 169
- WT, 120
- WWW, 301
- X\_ABORT, 399
- X\_GET, 397
- X\_PUT, 395
- X\_RCV, 391
- X\_SEND, 390

**B**

保留的事件等级, 691  
 保险丝断开, 610  
 备用电池, 610  
   故障, 610  
 背景组织块(OB90), 74  
 比较 DATE\_AND\_TIME 变量, 493, 494  
 比较 STRING 变量, 499, 500, 501  
 编程, 45, 49, 51, 61, 81, 84  
   OB 的类型  
   OB121, 81  
   OB122, 84  
   OB80, 45  
   OB81, 49  
   OB82, 51, 52  
   OB85, 61  
 编程错误, 209, 219  
 编程错误屏蔽, 209  
 编程错误事件, 209  
 编程错误组织块(OB121), 81  
 编辑 STRING 变量, 495, 496, 497, 498  
 编辑数值, 504, 505, 506  
 变量声明表, 45, 81, 84  
   对于 OB121, 81  
   对于 OB122, 84  
   对于 OB80, 45  
 标准 OB 事件, 674  
 标准用户事件, 689  
 使用 SFC 81, 98  
 部分 SSL 列表的结构, 616

**C**

参考书目, 703  
 参考通道错误, 612  
   模拟量输入模块, 612  
 参数, 92, 93, 95, 96, 100, 101, 102, 103, 104, 120, 134,  
 142, 143, 144, 145, 147, 151, 153, 157, 158, 159, 160,  
 162, 197, 198, 199, 200, 201, 205, 206, 207, 221, 222,  
 223, 236, 238  
 ACCFLT\_ESR, 217  
 ACCFLT\_Masked, 216

ACCFLT\_MASKED, 215  
 ACCFLT\_QUERY, 217  
 ACCFLT\_RESET\_MASK, 216  
 ACCFLT\_SET\_MASK, 215  
 BLK, 100  
 BVAL, 100, 101  
 CDT, 134  
 COUNT, 103  
 CQ, 144  
 CV, 144  
 DB\_NUMBER, 103  
 DSTBLK, 95  
 DTIME, 205  
 EVENTN, 238, 239  
 INFO1, 238, 239  
 INFO2, 236  
 IOID, 151, 153, 157, 159, 162  
 LADDR, 151, 153, 157, 159, 162  
 LOW\_LIMIT, 103  
 MODE, 221, 222, 223  
 NR, 142, 143, 144  
 OB\_NR, 197, 198, 199, 200, 205, 206, 207, 221, 223  
 PERIOD, 197  
 PRGFLT\_ESR, 217  
 PRGFLT\_MASKED, 215, 216  
 PRGFLT\_QUERY, 217  
 PRGFLT\_RESET\_MASK, 216  
 PRGFLT\_SET\_MASK, 215  
 PV, 142  
 RECNUM, 151, 153, 157, 159  
 RECORD, 151, 157  
 SDT, 197  
 SEND, 236, 237, 238  
 SFC 1 READ\_CLK, 134  
 SFC 51 和 55 至 59 的 BUSY, 91  
 SFC 51 和 55 至 59 的 RET\_VAL, 91  
 SFC 58 WR\_REC, 157  
 SFC 64 TIME\_TICK, 145  
 SRCBLK, 96  
 STATUS, 200, 206  
 UP\_LIMIT, 103  
 WT, 120  
 写默认参数, 153

- 信号模块的, 147
- 异步 SFC 的 REQ, 91
- 参数分配出错, 612, 613
  - 模拟量输出模块, 612
  - 模拟量输入模块, 612
  - 数字量输入模块, 612
- 操作序列, 260, 261
  - 实现, 260
- 使用 SFC 24, 107
- 插入/删除 OB (OB83), 54
- 超出测量范围, 612
  - 模拟量输入模块, 612
- 初始化存储区, 100
  - 使用 SFC 21 FILL, 100
- 处理器故障, 610
- 处理时间中断, 193
- 处理延时中断, 203
- 传送参数, 151, 153
  - 使用 SFC 55 WR\_PARM, 151
  - 使用 SFC 56 WR\_DPARM, 153
- 创建数据块, 117
- 词汇表, 705
- 从通信伙伴接收数据并将其存储在数据块(RK 512)中, 586
- 存储卡, 610
- 存储器区, 623
- 错误, 220, 610, 611
  - ADC/DAC, 610
  - EPROM, 610
  - RAM, 610
  - 异步, 219
- 错误 OB, 45, 46, 49, 51, 61, 209
  - OB 的类型
    - OB80, 45, 46, 47
    - OB81, 49
    - OB82, 51
    - OB85, 61, 62, 63, 64
- 错误处理, 209
- 错误寄存器, 217
  - 使用 SFC 38 READ\_ERR 读, 217
- 错误检测, 45, 81, 84
  - OB 的类型
    - OB121, 81
    - OB122, 84

- OB80, 45
- 错误屏蔽, 209
  - 编程错误, 209
  - 访问出错, 209
- 错误信息, 104, 105, 106, 206, 223, 225
  - SFC 22 CREAT\_DB, 103
  - SFC 23 DEL\_DB, 105
  - SFC 34 QRY\_DINT, 206
  - SFC 40 EN\_IRT, 223
  - SFC 42 EN\_AIRT, 225
- 错误中断, 219
  - 同步, 220
  - 异步, 219, 220

## D

- 递减计数, 482
- 递增/递减计数, 483
- 递增计数, 481
- 电池耗尽, 610
- 电流低于测量范围, 612
  - 模拟量输入模块, 612
- 电源, 610
  - 故障, 610
- 电源错误, 220
- 电源错误组织块(OB81), 49
- 定时器编号错误, 209
- 读取, 230, 289
  - DP 标准从站/PROFINET IO 设备的连续数据, 289
  - DP 从站的诊断数据, 286
  - 使用 SFC 51 RDSYSST, 230
- 读取时的 I/O 访问错误, 209
- 读取时对齐出错, 209
- 读取时间, 134
  - 使用 SFC 1 READ\_CLK, 134
- 读取时区域出错, 209
- 读取系统时间, 145
  - 使用 SFC 64 TIME\_TCK, 145
- 读取系统状态, 230
  - 使用 SFC 51 RDSYSST, 230
- 断开延迟, 480
  - 生成, 480
- 断线, 612, 613, 614

模拟量输出模块, 612  
 模拟量输入模块, 612  
 多值计算中断, 219  
 多值计算中断组织块(OB60), 35

## F

发送分段数据, 352  
   通过 FB 12, 352  
   通过 SFB 12, 352  
 返回值, 224, 225  
   SFC 41 DIS\_AIRT, 224  
   SFC 42 EN\_AIRT, 225  
 访问出错, 209  
 访问错误屏蔽, 209  
 访问错误事件, 209  
 非协调发送数据, 345  
   通过 FB 8, 345  
   通过 SFB 8, 345  
 非协调接收数据, 349  
   通过 FB 9, 349  
   通过 SFB 9, 349  
 复制变量, 95  
   使用 SFC 20 BLKMOV, 95  
   使用 SFC 20, 95

## G

概述, 334, 485  
 更新 DP 互连, 300  
 更新 PROFINET CBA 组件的 PROFINET 接口的输出, 299  
 更新过程映像输出表, 253  
 更新过程映像输入表, 251  
 更新中断, 29  
 更新中断 OB (OB56), 29  
 共模错误, 612, 613  
   模拟量输出模块, 612  
   模拟量输入模块, 612  
 关于使用 SFB 生成块相关消息的介绍, 439  
 关于使用 SFC 生成块相关消息的介绍, 463

## J

机架故障, 65, 66, 219

机架故障组织块(OB86), 65  
 集成计数器功能, 601  
 集成频率计功能, 603  
 计数器(CPU 312), 601  
 计数器编号错误, 209  
 将全部或部分数据块传送给通信伙伴(RK 512), 581  
 将全部或部分数据块传送给通讯伙伴, 569, 577  
 接地错误, 613, 614  
   数字量输入模块, 612  
 接口 DB, 295, 297  
 接收来自通信伙伴的数据并将其填入数据块。 , 572  
   使用 FB 64, 427  
 接通延迟, 479  
   生成, 479

## K

开放通信, 405, 406  
   FB 功能, 406  
   分配使用 TCP 和 ISO on TCP 的通信连接参数, 410  
   概述, 405  
   使用 UDP 的远程通讯伙伴地址信息的结构, 417  
   为使用 UDP 的本地通讯接入点分配参数, 415  
 可能的部分系统状态列表, 618  
 空载电压, 613, 614  
   模拟量输出模块, 612  
 控制, 507, 511, 512, 514, 515, 518  
   使用 SFB 41 进行连续控制, 507  
   使用 SFB 42 进行步控制, 514  
 块编号错误, 209  
 块类型, 626  
 扩展机架故障, 610

## L

类型 ID, 617  
   模块的, 617  
 连接, 244, 245, 246, 247, 248  
   使用 SFC 87 诊断, 244  
 临时变量(TEMP), 84  
   OB 所需的, 84  
 逻辑地址, 263, 267  
   模块的



查询所有地址, 267  
通道的  
查询, 263

## M

脉冲, 477, 478  
生成, 477  
脉宽调制, 520, 521, 529  
模块, 610, 611, 617, 619  
标识, 619  
故障, 610  
类型 ID, 617  
模块插槽, 264  
逻辑地址的  
查询, 264  
模块诊断数据, 609, 670, 671  
模块诊断信息, 668  
模块状态信息, 655  
模式更改, 679

## N

内部错误, 610

## P

频率计(CPU 312), 603  
屏蔽, 209  
错误事件, 209  
屏蔽同步错误, 209

## Q

启动, 76, 77, 78, 79, 80  
启动组织块(OB100  
OB101 和 OB102), 76  
切换的 DP 从站, 651  
清除输入缓冲区, 575  
区域错误, 209  
取消屏蔽, 209  
错误事件, 209

## R

日期, 133  
日期和时间作为复杂数据类型, 489

## S

删除/插入模块中断, 219  
使用 SFC 23, 105  
设备状态改变, 381  
接收远程设备的状态改变, 381  
设定 TOD, 133  
使用 SFC 0 SET\_CLK, 133  
生成数据块, 110  
使用 SFC 82 "CREA\_DBL, 110  
时间, 133  
时间错误, 220  
时间错误组织块(OB80), 45  
时间中断, 193, 195, 196, 197, 198, 199, 200, 219  
OB, 193  
冷重启动, 196  
**暖启动**, 196  
使用 SFC 28 SET\_TINT 设置, 197  
使用 SFC 29 CAN\_TINT 取消, 198  
使用 SFC 30 ACT\_TINT 激活, 199  
使用 SFC 31 QRY\_TINT 查询, 200  
调用条件, 193  
执行和响应, 195  
状况影响, 195  
时间中断 OB (OB 10 至 OB 17), 16  
时钟, 133  
同步, 133  
主站, 133  
识别 DP 主站系统的总线拓扑, 248  
使用 FB 20 "GETIO"读取 DP 标准从站/PROFINET IO 设备的所有输入, 316  
使用 FB 21 "SETIO"写入 DP 标准从站/PROFINET IO 设备的所有输出, 317  
使用 FB 22 "GETIO\_PART"读取 DP 标准从站/PROFINET IO 设备的一部分输入, 318  
使用 FB 23 "SETIO\_PART"写入 DP 标准从站/PROFINET IO 设备的一部分输出, 320  
使用 FB 63 "TSEND"发送数据, 424  
使用 FB 65 "TCON"建立连接, 419

- 使用 FB 66 "TDISCON"终止连接, 422
- 使用 FB 67 "TUSEND"通过 UDP 发送数据, 431
- 使用 FB 68 "TURCV"通过 UDP 接收数据, 434
- 使用 SFB 0 "CTU"进行递增计数, 481
- 使用 SFB 1 "CTD"进行递减计数, 482
- 使用 SFB 2 "CTUD"递增/递减计数, 483
- 使用 SFB 3 "TP"生成一个脉冲, 477
- 使用 SFB 31 "NOTIFY\_8P"生成无确认显示的块相关消息, 445
- 使用 SFB 32 "DRUM"实现操作序列, 260
- 使用 SFB 33 "ALARM"生成带确认的与块相关的消息, 447
- 使用 SFB 34 "ALARM\_8"生成针对八个信号的不附带关联值的块相关消息, 453
- 使用 SFB 35 "ALARM\_8P"生成带 8 个信号伴随值的与块相关的消息, 450
- 使用 SFB 36"NOTIFY"生成不带确认的与块相关的消息, 443
- 使用 SFB 37 "AR\_SEND"发送归档数据, 455
- 使用 SFB 4 "TON"生成一个接通延迟, 479
- 使用 SFB 41/FB 41 "CONT\_C"进行连续控制, 507
- 使用 SFB 42/FB 42 "CONT\_S"进行步控制, 514
- 使用 SFB 43 "PULSEGEN"生成脉冲, 520
- 使用 SFB 5 "TOF"生成一个关闭延迟, 480
- 使用 SFB 53 "WRREC"写入数据记录, 169
- 使用 SFB 81"RD\_DPAR"读取预定义的参数, 164
- 使用 SFB104 "IP\_CONF"设置 IP 组态, 303
- 使用 SFC 0 "SET\_CLK"设定 TOD, 133
- 使用 SFC 1 "READ\_CLK"读取时间, 134
- 使用 SFC 100 "SET\_CLKS"设定时间和 TOD 状态, 136
- 使用 SFC 101 "RTM"控制运行时间定时器, 140
- 使用 SFC 102 "RD\_DPARA"读取预定义参数, 150
- 使用 SFC 104 "CiR"控制 CiR, 122
- 使用 SFC 105 "READ\_SI"读取动态系统资源, 472
- 使用 SFC 106 "DEL\_SI"释放动态系统资源, 475
- 使用 SFC 107 "ALARM\_DQ"生成可确认和永久确认的块相关消息, 470
- 使用 SFC 108 "ALARM\_D"生成可确认和永久确认的块相关消息, 470
- 使用 SFC 109 "PROTECT"激活和取消激活 CPU 访问保护, 124
- 使用 SFC 126 "SYNC\_PI"在同步循环中更新过程映像分区输入表, 254
- 使用 SFC 127 "SYNC\_PO"在同步循环中更新过程映像分区, 256
- 使用 SFC 17 "ALARM\_SQ"生成可确认的块相关消息及使用 SFC 18 "ALARM\_S"生成永久确认的块相关消息, 466
- 使用 SFC 19"ALARM\_SC"查询最后一个 ALARM\_SQ 进入事件消息的确认状态, 469
- 使用 SFC 2 "SET\_RTM"设置运行时间定时器, 142
- 使用 SFC 21 "FILL, 100
- 使用 SFC 22 "CREAT\_DB, 103
- 使用 SFC 26 "UPDAT\_PI"更新过程映像输入表, 251
- 使用 SFC 27 "UPDAT\_PO"更新过程映像输出表, 253
- 使用 SFC 28 "SET\_TINT"设置时间中断, 197
- 使用 SFC 29 "CAN\_TINT"取消时间中断, 198
- 使用 SFC 3 "CTRL\_RTM"启动和停止运行时间定时器, 143
- 使用 SFC 30 "ACT\_TINT"激活时间中断, 199
- 使用 SFC 31 "QRY\_TINT"查询时间中断, 200
- 使用 SFC 32 "SRT\_DINT"启动延时中断, 205
- 使用 SFC 33 "CAN\_DINT"取消延时中断, 207
- 使用 SFC 34 "QRY\_DINT"查询延时中断, 206
- 使用 SFC 35 "MP\_ALM"触发多处理器中断, 121
- 使用 SFC 36 "MSK\_FLT"屏蔽同步错误, 215
- 使用 SFC 37 "DMSK\_FLT"取消屏蔽同步错误, 216
- 使用 SFC 38 "READ\_ERR"读取错误寄存器, 217
- 使用 SFC 39 "DIS\_IRT"禁用新中断和异步错误的处理, 221
- 使用 SFC 4 "READ\_RTM"读取运行时间定时器, 144
- 使用 SFC 40 "EN\_IRT"启用新中断和异步错误的处理, 223
- 使用 SFC 41 "DIS\_AIRT"延迟更高优先级中断和异步错误的处理, 224
- 使用 SFC 42 "EN\_AIRT"启用更高优先级中断和异步错误的处理, 225
- 使用 SFC 43 "RE\_TRIGR"再触发循环时间监控, 119
- 使用 SFC 44 "REPL\_VAL, 109
- 使用 SFC 46 "STP"使 CPU 进入停机状态, 119
- 使用 SFC 47 "WAIT"延时用户程序执行, 120
- 使用 SFC 48 "SNC\_RTCB"同步子时钟, 135
- 使用 SFC 49 "LGC\_GADR"查询属于逻辑地址的模块插槽, 264
- 使用 SFC 5 "GADR\_LGC"查询模块的逻辑基址, 263
- 使用 SFC 50 "RD\_LGADR"查询模块的所有逻辑地址, 267
- 使用 SFC 51 "RDSYSST"读取系统状态列表或部分列表, 230
- 使用 SFC 52 "WR\_USMSG"将用户自定义诊断事件写入诊断缓冲区, 236
- 使用 SFC 54 "RD\_DPARM"读取定义的参数, 149
- 使用 SFC 55"WR\_PARM"写动态参数, 151

使用 SFC 56 "WR\_DPARM"写默认参数, 153  
 使用 SFC 58 "WR\_REC"写数据记录, 157  
 使用 SFC 59 "RD\_REC"读数据记录, 159  
 使用 SFC 6 "RD\_SINFO"读取 OB 启动信息, 227  
 使用 SFC 64 "TIME\_TCK"读取系统时间, 145  
 使用 SFC 65 "X\_SEND"将数据发送给在本地 S7 站外的一个通信伙伴, 390  
 使用 SFC 6 读取 OB 启动信息, 227  
 使用 SFC 70 "GEO\_LOG"确定模块的起始地址, 268  
 使用 SFC 71 "LOG\_GEO"确定属于一个逻辑地址的插槽, 270  
 使用 SFC 78 "OB\_RT"确定 OB 程序运行时间, 240  
 使用 SFC 79 "SET"在 I/O 区域中设置位域, 258  
 使用 SFC 80 "RSET"复位 I/O 区域中的位域, 259  
 使用 SFC 82 "CREA\_DBL"在装载存储器中生成数据块, 110  
 使用 SFC 83 "READ\_DBL"从装载存储器的数据块中读取数据, 113  
 使用 SFC 84 "WRIT\_DBL"写数据到装载存储器中的数据块, 115  
 使用 SFC 85 "CREA\_DB"创建数据块, 117  
 使用 SFC 87 "C\_DIAG"诊断当前连接状态, 244  
 使用 SFC 87 诊断, 244  
 使用 SFC 9 "EN\_MSG"启用块相关、符号相关和组状态消息, 459  
 使用 SFC99 "WWW"启用或同步用户 Web 页面, 301  
 使用输出参数 RET\_VAL 判断错误, 87  
 事件, 236, 237, 238, 239, 673, 674  
   ID, 237, 238, 673  
   等级, 673  
 事件等级 1 - 标准 OB 事件, 674  
 事件等级 2 - 同步错误, 675  
 事件等级 3 - 异步错误, 676  
 事件等级 4 - 停止事件和其它模式更改, 679  
 事件等级 5 - 模式运行时事件, 682  
 事件等级 6 - 通讯事件, 683  
 事件等级 7 - H/F 事件, 684  
 事件等级 8 - 模块的诊断事件, 687  
 事件等级 9 - 标准用户事件, 689  
 事件等级 A 和 B - 自由用户事件, 691  
 事件和事件 ID, 673  
 数据, 292, 293, 294  
   向 DP 标准从站/PROFINET IO 设备写入连续数据, 292

数据记录, 147, 148, 157, 159, 160, 161, 162, 163, 167, 168, 169  
 读取, 147  
 使用 SFC 58 WR\_REC 写, 157  
 使用 SFC 59 RD\_REC 读, 159  
 写, 147  
 用 SFB 52 RDREC 读取, 167  
 用 SFB 53 WRREC 写入, 169  
 数据记录的写和读, 147  
 数据块, 103, 105, 113, 114, 115, 116  
   使用 SFC 22 CREAT\_DB 创建, 103  
   使用 SFC 23 删除, 105  
   使用 SFC 83 "READ\_DBL"从装载存储器的数据块中读取数据, 113  
   用 SFB 74 ", 130

## T

替换值, 109  
   使用 SFC 44 REPL\_VAL 写入到 ACCU 1, 109  
 调用汇编码块, 607  
 停止事件, 679  
 通道, 610, 611  
   CPU 的, 610  
   错误, 610  
 通道专有诊断数据的结构, 612  
 通过 Industrial Ethernet 打开通讯, 405  
 通过 FB 210 "FW\_TCP"使用通过 TCP 的 FETCH 和 WRITE 服务链接第三方系统, 437  
 通过 FB FB 220 "FW\_IOT"使用通过 ISO-on-TCP 的 FETCH 和 WRITE 服务链接第三方系统, 438  
 通过 SFB 16 "PRINT"将数据发送到打印机, 366  
 通过 SFB 19 "START"在远程设备上启动暖启动或冷启动, 372  
 通过 SFB 20 "STOP"将远程设备切换到 STOP 状态, 375  
 通过 SFB 21 "RESUME"在远程设备上启动热启动, 377  
 通过 SFB 22 "STATUS"查询远程伙伴的状态, 379  
 通过 SFB 23 "USTATUS"接收远程设备的状态改变, 381  
 通过 SFB 8/FB 8 "USEND"进行无协调的数据发送, 345  
 通过 SFB/FB 12 "BSEND"发送分段数据, 352  
 通过 SFB/FB 13 "BRCV"接收分段数据, 356  
 通过 SFB/FB 14 "GET"从远程 CPU 中读取数据, 363  
 通过 SFB/FB 14 从远程 CPU 中读取数据, 363  
 通过 SFB/FB 15 "PUT"向远程 CPU 写入数据, 359

通过 SFB/FB 15 向远程 CPU 写入数据, 359  
通过 SFB/FB 9 "URCV"进行无协调的数据接收, 349  
通过 SFC 60"GD\_SND"发送 GD 信息包, 323  
通过 SFC 61 "GD\_RCV"编程接受已接收到的 GD 信息包,  
326  
通过 SFC 62 "CONTROL"查询属于 SFB 实例的连接的状态,  
383  
通过 SFC 66 "X\_RCV"从本地 S7 站以外的通信伙伴中接收  
数据, 391  
通过 SFC 67 "X\_GET"从本地 S7 站以外的通信伙伴中读取  
数据, 397  
通过 SFC 68 "X\_PUT"将数据写入本地 S7 站以外的通信伙  
伴, 395  
通过 SFC 69 "X\_ABORT"中止已存在的、到本地 S7 站以外  
的通信伙伴的连接, 399  
通过 SFC 72 "I\_GET"从本地 S7 站内的通信伙伴中读取数  
据, 400  
通过 SFC 73 "I\_PUT"将数据写入本地 S7 站内的通信伙伴,  
402  
通过 SFC 74 "I\_ABORT"中止已存在的、到本地 S7 站内的  
通信伙伴的连接, 404  
通过用户程序控制计数器, 557  
通过用户程序控制脉宽调制, 566  
通过用户程序控制频率测量, 562  
通过用户程序使用模拟量输出控制定位, 535  
通过用户程序使用数字量输出控制定位, 546  
通信 SFC, 335, 336  
通信错误组织块(OB87), 70  
通信冗余错误 OB (OB73), 44  
通讯, 220, 640  
    错误, 219  
    **中断**, 220  
    状态数据, 640  
通讯 SFC 的公用参数, 385  
通讯事件, 683  
同步, 133  
    时钟, 133  
同步错误, 209  
同步错误事件  
    屏蔽, 209  
    取消屏蔽, 209

## W

外部错误, 610  
完全重启, 76, 372, 374  
无参数分配, 610  
无辅助电压, 610  
无前置连接器, 610

## X

系统区域, 624  
系统数据, 615  
系统诊断, 227  
系统状态列表, 615, 618  
    部分列表, 618  
系统状态列表(SSL)概述, 615  
校准错误, 209  
写错误, 209  
    数据块, 209  
写入错误实例数据块, 209  
写入时的 I/O 访问错误, 209  
写入时对齐出错, 209  
写入时区域出错, 209  
循环中断, 22, 219  
循环中断 OB (OB30 至 OB38), 22  
    使用 SFC 25, 108

## Y

延迟和禁用中断和异步错误, 219  
延迟时间, 203  
延时用户程序, 120  
    使用 SFC 47 WAIT, 120  
延时中断, 20, 203, 204, 205, 206, 207, 219  
    使用 SFC 32 SRTL\_DINT 启动, 205  
    使用 SFC 33 CAN\_DINT 取消, 207  
    使用 SFC 34 QRY\_DINT 查询, 206  
    调用条件, 203  
    在启动 OB 中启动, 203  
    状况影响, 203  
延时中断 OB (OB20 至 OB23), 20  
异步 SFC 的 REQ、  
    RET\_VAL 和 BUSY 参数含义, 91

- 异步错误, 45, 81, 84, 215, 216, 219, 221, 222, 223, 224, 225, 675, 676
    - OB121, 81
    - OB122, 84
    - OB80, 45
    - 使用 SFC 36 MSK\_FLT 屏蔽, 215
    - 使用 SFC 37 DMSK\_FLT 取消屏蔽, 216
    - 使用 SFC 39 DIS\_IRT 禁用, 221
    - 使用 SFC 40 EN\_IRT 启用, 223
    - 使用 SFC 41 DIS\_AIRT 延迟, 224
    - 使用 SFC 42 EN\_AIRT 启用, 225
  - 影子存储器, 295
  - 硬件中断, 219, 610
    - 丢失, 610
  - 硬件中断 OB, 24
  - 硬件中断组织块(OB40 至 OB47), 24
  - 用 SFB 52 "RDREC"读取数据记录, 167
  - 用 SFB 54 "RALRM"从 DP 从站接收中断, 171
  - 用 SFB 73 "RCVREC"接收数据记录, 127
  - 用 SFB 75 "SALRM"向 DP 主站发送中断, 187
  - 用 SFC 11 "DPSYC\_FR"同步 DP 从站组, 276
  - 用 SFC 12 "D\_ACT\_DP"激活和取消激活 DP 从站 /PROFINET IO 设备, 281
  - 用 SFC 13 "DPNRM\_DG"读取 DP 从站的诊断数据(从站诊断), 286
  - 用 SFC 15 "DPWR\_DAT"向 DP 标准从站/PROFINET IO 设备写入连续数据, 292
  - 用 SFC 7 "DP\_PRAL"触发 DP 主站上的硬件中断, 273
  - 用户存储器, 108
    - 使用 SFC 25 压缩, 108
  - 用户信息, 610
  - 用于 S7 通讯的 SFB/FB 和 SFC/FC 的公用参数, 337
  - 用于 S7 通讯的 SFB 对故障如何反应, 343
  - 用于生成块相关消息的 SFB 的启动特性, 461
  - 用于生成块相关消息的 SFB 如何对问题做出反应, 462
  - 用于未组态 S7 连接的通讯 SFC 的出错信息, 387
  - 用于未组态的 S7 连接的通信 SFC, 335, 336
    - 分类, 334
  - 用于循环访问用户数据的 FB :
    - 引言, 315
  - 用于组态的 S7 连接的 SFB 启动例行程序, 341
  - 优先级错误 OB, 61
  - 优先级错误组织块(OB85), 61
  - 优先级等级, 19, 20, 22, 24, 35, 39, 41, 46, 47, 49, 52, 59, 61, 65, 70, 74, 76, 81, 84, 209, 217, 227, 679
    - OB 的类型
    - OB121, 81
    - OB122, 84
    - OB80, 45
    - OB81, 49
    - OB82, 51
    - OB85, 61
  - 远程伙伴的状态, 379
    - 查询, 379
  - 运行时间定时器, 139, 142, 143, 144
    - 取值范围, 139
    - 使用 SFC 2 SET\_RTM 设置, 142
    - 使用 SFC 3 CTRL\_RTM 启动, 143
    - 使用 SFC 3 CTRL\_RTM 停止, 143
    - 使用 SFC 4 READ\_RTM 读出, 144
    - 特征, 139
- ## Z
- 再触发循环时间监控, 119
    - 使用 SFC 43 RE\_TRIGR, 119
  - 在 H 系统中使用 SFC 90 "H\_CTRL"控制操作, 597
  - 在 H 系统中使用 SFC 90 控制操作, 597
  - 诊断缓冲区, 209, 615, 667
  - 诊断事件, 687
  - 诊断数据, 148, 609, 610, 615
    - CPU 的, 615
    - 结构, 609
    - 内容, 609
    - 信号模块的, 147, 615
  - 诊断数据结构概述, 609
  - 诊断中断, 220, 610
    - 来自替换件, 610
  - 诊断中断组织块(OB82), 51
  - 制造商特定中断, 32
  - 制造商特定中断的 OB(OB57), 32
  - 中断, 16, 20, 21, 22, 24, 26, 27, 28, 29, 32, 54, 219, 221, 222, 223, 224, 225
    - DPV1 中断, 26, 29, 32
    - 插入/删除中断, 54
    - 等级, 219, 220
    - 更新中断, 29

- 使用 SFC 39 DIS\_IRT 禁用, 221
- 使用 SFC 40 EN\_IRT 启用, 223
- 使用 SFC 41 DIS\_AIRT 延迟, 224
- 使用 SFC 42 EN\_AIRT 启用, 225
- 循环中断, 22
- 延时, 20, 21
- 硬件中断 OB, 24
- 制造商特定中断, 32
- 状态中断, 26, 27
- 中断 OB, 51
  - 时间中断, 17
  - 诊断中断, 51
- 中断等级, 54
- 中断状态, 635
- 重新启动, 76, 78, 79, 80, 377, 378
  - 在远程设备上启动, 377
- 重新启动(暖启动), 372
- 重新启动(暖启动)或完全重新启动, 372
  - 在远程设备上执行, 372
- 重新组态模块**, 55
- 周期监视, 610
- 主时钟, 133
- 转换数据类型格式, 502, 503
- 状态中断, 26
- 状态中断 OB (OB55), 26
- 子时钟, 135
  - 同步, 135
- 自由用户事件, 691
- 组合代码块
  - 调用, 607
- 组件标识, 629
- 组态, 612
  - 错误
    - 模拟量输出模块, 612
    - 模拟量输入模块, 612
    - 数字量输入模块, 612
- 组织块(OB), 16, 20, 22, 26, 29, 32, 35, 45, 49, 51, 54, 59, 61, 65, 70, 74, 76, 81, 84
  - CPU 硬件故障 OB (OB84), 59
  - OB121, 81
  - OB122, 84
  - 插入/删除 OB (OB83), 54
  - 多值计算中断 OB (OB60), 35
  - 更新中断 OB (OB56), 29
  - 后台 OB (OB90), 74
  - 机架故障组织块(OB86), 65
  - 类型
    - OB80, 45
    - OB81, 49
    - OB82, 51
    - OB85, 61
  - 启动组织块(OB100 和 101), 76
  - 时间中断 OB (OB 10 至 OB 17), 16
  - 通讯错误组织块(OB87), 70
  - 完全重新启动 OB (OB100), 76
  - 循环中断 OB (OB30 至 OB38), 22
  - 延时中断 OB (OB20 至 OB23), 20
  - 优先级错误组织块(OB85), 61
  - 诊断中断 OB (OB82), 51
  - 制造商特定中断的 OB(OB57), 32
  - 重新启动 OB (OB101), 76
  - 状态中断 OB (OB55), 26
- 组织块(OB)概述, 11