

# SIEMENS

## SIMATIC

### S7-300 CPU 31xC 技术功能

操作说明

前言

---

技术功能概述

---

1

定位

---

2

使用模拟输出定位

---

3

使用数字输出定位

---

4

计数、频率测量和脉冲宽度  
调制

---

5

点对点通讯

---

6

控制

---




7

2007 年 2 月版

A5E00432666-04

## 安全技术提示

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。

 <b>危险</b>
表示如果不采取相应的小心措施， <b>将会</b> 导致死亡或者严重的人身伤害。
 <b>警告</b>
表示如果不采取相应的小心措施， <b>可能</b> 导致死亡或者严重的人身伤害。
 <b>小心</b>
带有警告三角，表示如果不采取相应的小心措施，可能导致轻微的人身伤害。
<b>小心</b>
不带警告三角，表示如果不采取相应的小心措施，可能导致财产损失。
<b>注意</b>
表示如果不注意相应的提示，可能会出现不希望的结果或状态。


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

请注意下列说明：

 <b>警告</b>
设备仅允许用在目录和技术说明中规定的使用情况下，并且仅允许使用西门子股份有限公司推荐的或指定的其他制造商生产的设备和部件。设备的正常和安全运行必须依赖于恰当的运输，合适的存储、安放和安装以及小心的操作和维修。

## 商标

所有带有标记符号®的都是西门子股份有限公司的注册商标。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权利的目地由第三方使用而特别标示的。

## 责任免除

我们已对印刷品中所述内容与硬件和软件的一致性作过检查。然而不排除存在偏差的可能性，因此我们不保证印刷品中所述内容与硬件和软件完全一致。印刷品中的数据都按规定经过检测，必要的修正值包含在下一版本中。

# 前言

## 本手册用途

本手册全面概述了 CPU 31xC 集成的技术功能。

本手册是为使用基于 SIMATIC 过程控制系统的技术功能实现控制任务的系统和控制工程师而编写的。

## 经验要求

为了理解本手册的内容，您应具备一些自动化工程方面的经验。

## 本手册的有效性

本手册对具有下列硬件和软件版本的 CPU 有效：

CPU	约定： 本手册介绍了下列 CPU：	订货号	版本	
			固件	硬件
CPU 312C	CPU 31xC	6ES7312-5BD01-0AB0	V2.0.0	01
CPU 313C		6ES7313-5BE01-0AB0	V2.0.0	01
CPU 313C-2 PtP		6ES7313-6BE01-0AB0	V2.0.0	01
CPU 313C-2 DP		6ES7313-6CE01-0AB0	V2.0.0	01
CPU 314C-2 PtP		6ES7314-6BF01-0AB0	V2.0.0	01
CPU 314C-2 DP		6ES7314-6CF01-0AB0	V2.0.0	01

---

## 说明

本文档包含截止到出版时所有可用模块的说明。

我们保留另行增加关于新模块和现有模块新版本的最新产品信息的权利。

---

## 相对于先前版本的变更

在上述版本中，CPU 31xC 加入了下列新功能：

### 计数器模块：

- 在计数操作模式下，当前的计数值可通过直接访问计数器模块的输入地址读取。
- 在频率测量操作模式下，当前的频率值可通过直接访问计数器模块的输入地址读取。
- 在计数和频率测量操作模式下，可分步设置轨迹 A/脉冲、轨迹 B/方向和硬件门信号的最大频率。
- 可在脉冲宽度调制操作模式下分步设置硬件门信号的最高频率。
- 可在计数操作模式下分步设置锁存信号的最高频率。

### 定位子模块：

- 使用数字输出和模拟输出进行定位时，可分步设置位置反馈信号（编码器信号 A、B、N）的最高计数频率。
- 使用数字输出和模拟输出进行定位时，可设置长度测量和参考点转换信号的最大频率。
- 使用模拟输出进行定位时，除了使用  $\pm 10\text{ V}$  ( $\pm 20\text{ mA}$ ) 的驱动控制外，还要使用  $0\text{...}10\text{ V}$  ( $0\text{...}20\text{ mA}$ ) 的第二种控制模式以及一个附加的方向信号。

## 本手册的构成

本手册是 CPU 31xC 文档包的组成部分。

<b>参考手册/手册</b> <ul style="list-style-type: none"> <li>• PLC S7-300, CPU 规范 CPU 312 IFM 到 CPU 318-2 DP</li> <li>• S7-300 CPU 31xC 和 CPU 31x, 技术规范</li> </ul>	CPU 的功能、安装和技术规范的说明。
<b>手册</b> <ul style="list-style-type: none"> <li>• S7-300 自动化系统 CPU 31xC 技术功能 (→ 您当前正在阅读的手册)</li> <li>• CD-ROM: “实例”</li> </ul>	特定技术功能的描述: <ul style="list-style-type: none"> <li>• 定位</li> <li>• 计数</li> <li>• 点对点连接</li> <li>• 控制</li> </ul> CD 中包含了技术功能的实例。
<b>软件安装手册/操作说明</b> <ul style="list-style-type: none"> <li>• S7-300 自动化系统 硬件和安装: CPU 312 IFM -- 318-2 DP</li> <li>• S7-300 自动化系统 硬件和安装: CPU 31xC 和 CPU 31 x</li> </ul>	S7-300 的组态、安装、接线、寻址和调试的说明。
<b>参考手册</b> <ul style="list-style-type: none"> <li>• S7-300 自动化系统 模块数据</li> </ul>	信号模块、电源模块和接口模块的功能说明和技术数据。
<b>指令列表</b> <ul style="list-style-type: none"> <li>• PLC S7-300, CPU 规范 CPU 312 IFM 到 CPU 318-2 DP</li> <li>• S7-300 指令列表 CPU 31xC、CPU 31x、IM 151-7 CPU、BM 147-1 CPU、 BM 147-2 CPU</li> </ul>	CPU 指令及其执行时间的列表。 可执行块 (OB/SFC/SFB) 及其执行时间的列表。
<b>入门指南</b> <ul style="list-style-type: none"> <li>• 自动化系统 S7-300 入门指南集 提供以下“入门指南”文档:               <ul style="list-style-type: none"> <li>• CPU 31x: 调试</li> <li>• CPU 31xC: 调试</li> <li>• CPU 314C: 使用模拟输出定位</li> <li>• CPU 314C: 使用数字输出定位</li> <li>• CPU 31xC: 计数</li> <li>• CPU 31xC: 点对点连接</li> <li>• CPU 31xC: 控制</li> </ul> </li> <li>• CPU 315-2 PN/DP、CPU 317-2 PN/DP 和 CPU 319-3 PN/DP: 组态 PROFINET 接口</li> </ul>	“入门指南”文档通过运行具体实例, 引领您完成从每个调试步骤到功能应用的整个过程。

## 更多支持

如果您有任何技术问题，请联系您的西门子代表或负责的代理商。

可在以下网站上找到有关联系人的信息：

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

可在以下网站上获得各种 SIMATIC 产品和系统的技术文档向导：

<http://www.siemens.de/simatic-tech-doku-portal>

在线目录和订购系统位于：

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

## 培训中心

西门子提供许多培训课程，帮助您熟悉 SIMATIC S7 自动化系统。预知详情，请联系当地培训中心或位于 D 90327 Nuremberg, Germany 的培训中心总部。

电话：+49 (911) 895-3200

网址：<http://www.sitrain.com>

## 技术支持

您可通过以下方式联系所有 A&D 产品的技术支持部门

- 通过提交支持请求 (Support Request) Web 表单，网址为：  
<http://www.siemens.de/automation/support-request>
- 电话：+ 49 180 5050 222
- 传真：+ 49 180 5050 223

有关 Siemens 技术支持的详细信息，请访问我们的网站

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

## 在线服务与支持

除文档之外，我们在 Internet 上提供了一个综合的在线知识库，网址为：

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

在此处您将找到：

- 不断为您提供产品最新信息的新闻快递。
- 在“服务和支持”中通过“搜索”功能找到所需文档。
- 论坛，世界各地的用户和专家在此交流想法。
- 您当地的自动化与驱动产品代表。
- 有关现场服务、维修和备件的信息。在“服务”部分还提供了更多可用信息。

# 目录

前言 .....	3
<b>1 技术功能概述 .....</b>	<b>15</b>
<b>2 定位 .....</b>	<b>17</b>
2.1 受控定位器所支持的模式 .....	17
2.1.1 用模拟输出进行定位控制 .....	17
2.1.2 使用数字输出来控制定位 .....	18
2.2 定位概述 .....	19
2.3 功能范围 .....	20
2.4 用于定位控制的组件 .....	21
<b>3 使用模拟输出定位 .....</b>	<b>23</b>
3.1 接线 .....	23
3.1.1 重要安全规则 .....	23
3.1.2 接线规则 .....	24
3.1.3 用于使用模拟输出定位的端子 .....	25
3.1.4 连接组件 .....	28
3.2 参数组态 .....	29
3.2.1 参数组态的基本信息 .....	29
3.2.2 使用参数分配窗口组态参数 .....	31
3.2.3 基本参数 .....	31
3.2.4 驱动器参数 .....	32
3.2.5 轴参数 .....	35
3.2.6 编码器参数 .....	38
3.2.7 组态诊断 .....	39
3.3 集成到用户程序中 .....	40
3.4 使用模拟输出定位的功能 .....	41
3.4.1 使用模拟输出定位 - 步骤 .....	41
3.4.2 SFB ANALOG (SFB 44) 的基本组态 .....	47
3.4.3 点动模式 .....	50
3.4.4 参考点逼近 .....	52
3.4.5 参考点逼近 - 步骤 .....	56
3.4.6 相对增量逼近模式 .....	58
3.4.7 绝对增量逼近模式 .....	61
3.4.8 指定参考点 .....	64
3.4.9 删除剩余行程 .....	67
3.4.10 长度测量 .....	68

3.5	调整参数 .....	70
3.5.1	重要安全规则 .....	70
3.5.2	确定模块参数及其作用 .....	70
3.5.3	SFB 参数的作用 .....	72
3.5.4	检查监视时间 .....	73
3.6	错误处理和中断 .....	75
3.6.1	系统功能块 (SFB) 中的错误消息 .....	75
3.6.2	在用户程序中判断出错 .....	76
3.6.3	组态和判断诊断中断 .....	78
3.7	安装实例 .....	80
3.8	规范 .....	81
3.8.1	增量编码器 .....	81
3.8.2	错误列表 .....	83
3.8.3	参数分配窗口的模块参数 - 概述 .....	87
3.8.4	SFB ANALOG (SFB44) 的背景数据块的参数 .....	89
<b>4</b>	<b>使用数字输出定位 .....</b>	<b>93</b>
4.1	接线 .....	93
4.1.1	重要安全规则 .....	93
4.1.2	接线规则 .....	94
4.1.3	用于使用数字输出定位的端子 .....	95
4.1.4	连接组件 .....	97
4.1.5	用于数字输出的断路器 .....	98
4.2	参数组态 .....	100
4.2.1	参数组态的基本信息 .....	100
4.2.2	使用参数分配窗口进行组态 .....	101
4.2.3	基本参数 .....	101
4.2.4	驱动器参数 .....	102
4.2.5	轴参数 .....	107
4.2.6	编码器参数 .....	110
4.2.7	组态诊断 .....	111
4.3	集成到用户程序中 .....	112
4.4	用于使用数字输出定位的功能 .....	113
4.4.1	使用数字输出定位 (快行/爬行速度) .....	113
4.4.2	SFB DIGITAL (SFB 46) 的基本组态 .....	117
4.4.3	点动模式 .....	120
4.4.4	参考点逼近 .....	122
4.4.4.1	参考点逼近 - 工作原理 .....	122
4.4.4.2	参考点逼近 - 步骤 .....	126
4.4.5	相对增量逼近模式 .....	128
4.4.6	绝对增量逼近模式 .....	131
4.4.7	指定参考点 .....	134
4.4.8	删除剩余行程 .....	137
4.4.9	长度测量 .....	138



4.5	调整参数 .....	140
4.5.1	重要安全规则 .....	140
4.5.2	确定模块参数及其作用 .....	140
4.5.3	SFB 参数的作用 .....	141
4.5.4	检查监视时间 .....	142
4.6	错误处理和中断 .....	143
4.6.1	系统功能块 (SFB) 中的错误消息 .....	143
4.6.2	在用户程序中判断出错 .....	144
4.6.3	组态和判断诊断中断 .....	146
4.7	安装实例 .....	148
4.8	规范 .....	149
4.8.1	增量编码器 .....	149
4.8.2	错误列表 .....	151
4.8.3	参数分配窗口的模块参数 - 概述 .....	155
4.8.4	SFB DIGITAL (SFB46) 的背景数据块的参数 .....	157
<b>5</b>	<b>计数、频率测量和脉冲宽度调制 .....</b>	<b>159</b>
5.1	概述 .....	159
5.1.1	操作模式和特性 - 概述 .....	159
5.1.2	功能范围 - 概述 .....	160
5.1.3	计数器组件应用 - 概述 .....	161
5.2	接线 .....	162
5.2.1	接线规则 .....	162
5.2.2	针脚分配 .....	163
5.2.3	连接组件 .....	167
5.3	参数组态 .....	168
5.3.1	使用参数分配窗口进行组态 .....	168
5.3.2	基本参数 .....	169
5.3.3	连续、单独和周期计数参数 .....	170
5.3.4	频率测量 .....	172
5.3.5	脉冲宽度调制 .....	174
5.4	在用户程序中实现功能 .....	175
5.5	计数 - 功能说明 .....	177
5.5.1	基本计数术语 .....	177
5.5.2	连续计数 .....	179
5.5.3	单循环计数 .....	180
5.5.4	周期性计数 .....	183
5.5.5	通过用户程序控制计数器 .....	186
5.5.6	通过请求接口读写计数器 .....	189
5.5.7	计数器 FB .....	192
5.5.8	计数器输入 .....	193
5.5.9	计数器的门功能 .....	194
5.5.10	计数器输出的响应 .....	196
5.5.11	滞后对计数器模式的影响 .....	198
5.5.12	计数运行期间的硬件中断 .....	201

5.6	频率测量功能描述 .....	202
5.6.1	频率测量, 步骤 .....	202
5.6.2	通过用户程序控制频率计数器 .....	204
5.6.3	为频率测量读写请求接口 .....	207
5.6.4	频率计数器的功能块 .....	209
5.6.5	频率计数器输入 .....	210
5.6.6	频率测量的门功能 .....	210
5.6.7	频率测量输出的反应 .....	211
5.6.8	频率测量和硬件中断 .....	212
5.7	脉冲宽度调制功能说明 .....	213
5.7.1	脉冲宽度调制的步骤 .....	213
5.7.2	通过用户程序控制脉冲宽度调制 .....	214
5.7.3	为脉冲宽度调制读写请求接口 .....	216
5.7.4	脉冲宽度调制功能块 .....	218
5.7.5	脉冲宽度调制的门功能 .....	219
5.7.6	设置脉冲串的参数 .....	220
5.7.7	脉冲宽度调制输出的反应 .....	222
5.7.8	脉冲宽度调制和硬件中断 .....	222
5.8	错误处理和中断 .....	223
5.8.1	错误显示 .....	223
5.8.2	系统功能块 (SFB) 的错误消息 .....	223
5.8.3	组态诊断中断 .....	225
5.8.4	组态硬件中断 .....	227
5.9	安装实例 .....	230
5.10	规范 .....	231
5.10.1	功能 .....	231
5.10.2	增量编码器 .....	233
5.10.3	错误列表 .....	235
5.10.4	模块参数 (概述) .....	237
5.10.5	SFB 的背景 DB .....	241
<b>6</b>	<b>点对点通讯 .....</b>	<b>247</b>
6.1	概述 .....	247
6.1.1	产品描述 .....	247
6.1.2	通讯伙伴 .....	248
6.1.3	PtP 通讯的组件 .....	248
6.1.4	(RS422/485) 接口的属性 .....	249
6.1.5	串行传输字符 .....	250
6.2	接线 .....	252
6.2.1	接线规则 .....	252
6.2.2	连接串行电缆 .....	253
6.3	参数组态 .....	254
6.3.1	参数类型 .....	254
6.3.2	使用参数分配窗口进行组态 .....	255
6.3.3	基本参数 .....	256

6.3.4	ASCII 驱动程序的参数分配数据 .....	257
6.3.5	3964(R) 程序的参数分配数据 .....	264
6.3.6	用于 RK 512 通讯的参数分配数据 .....	267
6.4	在用户程序中实现连接 .....	268
6.5	通讯功能 .....	269
6.5.1	ASCII/3964(R) 的通讯功能 .....	269
6.5.1.1	ASCII/3964(R) 的通讯功能 - 基本功能 .....	269
6.5.1.2	用 SFB 60 “SEND_PTP” 发送数据 .....	269
6.5.1.3	用 SFB 61 “RCV_PTP” 接收数据 .....	272
6.5.1.4	用 SFB 62 “RES_RCVB” 清除接收缓冲区 .....	274
6.5.2	RK 512 通讯功能 .....	276
6.5.2.1	RK 512 计算机连接的通讯功能 - 基本功能 .....	276
6.5.2.2	用 SFB 63 “SEND_RK” 发送数据 .....	277
6.5.2.3	用 SFB 64 “FETCH_RK” 获取数据 .....	281
6.5.2.4	用 SFB 65 “SERVE_RK” 接收/提供数据 .....	285
6.5.2.5	实例：使用处理器间通讯标志 .....	288
6.5.2.6	实例：带处理器间通讯标志的 SEND_RK .....	289
6.5.3	有关系统功能块编程的信息 .....	290
6.6	调试 .....	292
6.6.1	调试接口硬件 .....	292
6.7	错误处理和中断 .....	293
6.7.1	错误定位和诊断 .....	293
6.7.2	系统功能块 (SFB) 的错误消息 .....	293
6.7.3	响应消息帧中的错误 ID .....	294
6.7.4	组态和判断诊断中断 .....	295
6.8	安装实例 .....	296
6.9	协议说明 .....	297
6.9.1	使用 ASCII 驱动程序传输数据 .....	297
6.9.1.1	使用 ASCII 驱动程序传输数据 - 基本信息 .....	297
6.9.1.2	使用 ASCII 驱动程序发送数据 .....	298
6.9.1.3	使用 ASCII 驱动程序接收数据 .....	300
6.9.1.4	数据流控制/握手 .....	306
6.9.2	用 3964(R) 程序传输数据 .....	307
6.9.2.1	用 3964(R) 程序传输数据 - 基本信息 .....	307
6.9.2.2	用 3964(R) 发送数据 .....	309
6.9.2.3	用 3964(R) 接收数据 .....	310
6.9.2.4	用 3964 (R) 程序发送和接收时的错误处理 .....	312
6.9.2.5	3964(R) 程序启动顺序 .....	314
6.9.2.6	用 3964(R) 程序发送 .....	315
6.9.2.7	用 3964(R) 程序接收 .....	316
6.9.3	用 RK 512 计算机连接传输数据 .....	319
6.9.3.1	使用 RK 512 计算机连接传输数据 — 基本信息 .....	319
6.9.3.2	使用 RK 512 发送数据 .....	322
6.9.3.3	用 RK 512 获取数据 .....	325
6.9.3.4	RK 512 处理请求的顺序 .....	329

6.10	规范 .....	331
6.10.1	常规规范 .....	331
6.10.2	ASCII 驱动程序的规范 .....	332
6.10.3	3964(R) 程序的规范 .....	333
6.10.4	RK 512 计算机连接的规范 .....	334
6.10.5	最小 CPU 周期数 .....	334
6.10.6	传输时间 .....	335
6.10.7	电缆 .....	336
6.10.8	错误消息 .....	340
6.10.9	SFB 的参数 .....	348
<b>7</b>	<b>控制 .....</b>	<b>353</b>
7.1	概述 .....	353
7.1.1	集成控制的原理 .....	353
7.1.2	基本信息 .....	355
7.2	接线 .....	357
7.2.1	接线规则 .....	357
7.3	参数组态 .....	358
7.3.1	用参数分配窗口组态 SFB .....	358
7.4	在用户程序中实现控制 .....	359
7.5	功能说明 .....	360
7.5.1	用 SFB 41 “CONT_C” 实现连续控制 .....	360
7.5.2	使用 SFB 42 “CONT_S” 进行步控制 .....	366
7.5.3	用 SFB 43 “PULSEGEN” 生成脉冲 .....	371
7.6	诊断/错误处理 .....	380
7.7	安装实例 .....	380
	<b>索引 .....</b>	<b>381</b>

表格

表格 3-1	连接器 X1 的针脚分配.....	26
表格 3-2	连接器 X2 的针脚分配.....	27
表格 4-1	连接器 X2 的针脚分配.....	96
表格 4-2	控制模式参数.....	155
表格 4-3	控制模式的含义.....	155
表格 4-4	其它驱动器参数.....	155
表格 5-1	输入参数.....	215
表格 5-2	输出参数.....	215
表格 5-3	频率范围.....	231
表格 5-4	可能出现错误指示的测量范围.....	231
表格 5-5	事件类别 01 (01H): “计数, SFB 参数 (SFB 47) 中的参数分配错误”.....	235
表格 5-6	事件类别 02 (02HH): “频率测量, SFB 参数 (SFB 48) 中的组态错误”.....	235
表格 5-7	事件类别 04 (04H): “脉冲宽度调制, SFB 参数 (SFB 49) 中的组态错误”.....	236
表格 5-8	事件类别 128 (80H): “全局 SFB 参数中的组态错误”.....	236



## 技术功能概述

### 概述

根据 CPU 类型，支持下列技术功能：

CPU	定位	计数	点对点通讯	控制
CPU 312C	-	2 个通道分别用于 计数、频率测量 (最大 10 kHz) 或脉冲宽度调制 (2.5 kHz)	-	-
CPU 313C	-	3 个通道分别用于 计数、频率测量 (最大 30 kHz) 或脉冲宽度调制 (2.5 kHz)	-	是
CPU 313C-2 DP	-	3 个通道分别用于 计数、频率测量 (最大 30 kHz) 或脉冲宽度调制 (2.5 kHz)	-	是
CPU 313C-2 PtP	-	3 个通道分别用于 计数、频率测量 (最大 30 kHz) 或脉冲宽度调制 (2.5 kHz)	ASCII (19.2 kbps 全双工, 38.4 kbps 半双工) 3964R (38.4 kbps)	是
CPU 314C-2 DP	有 1 个通道可用于 模拟或数字输出	4 个通道 <sup>1</sup> 分别用于 计数、频率测量 (最大 60 kHz) 或脉冲宽度调制 (2.5 kHz)	-	是
CPU 314C-2 PtP	有 1 个通道可用于 模拟或数字输出	4 个通道 <sup>1</sup> 分别用于 计数、频率测量 (最大 60 kHz) 或脉冲宽度调制 (2.5 kHz)	ASCII (19.2 kbps 全双工, 38.4 kbps 半双工) 3964R (38.4 kbps) RK512 (38.4 kbps)	是

<sup>1</sup> 使用定位通道时，仅两个通道可用

### 访问技术功能使用的 I/O

通过数字输入 I/O 的输入地址，始终都可以访问技术功能所使用的输入。

在内部锁定了对技术功能使用的输出的写访问。





## 定位

### 2.1 受控定位器所支持的模式

#### 2.1.1 用模拟输出进行定位控制

##### 引言

CPU 支持使用模拟输出控制定位。

##### 特性

用模拟输出控制的定位有以下特点：

- 驱动器是通过永久分配的**模拟输出**，电压为  $\pm 10\text{ V}$ （针脚 16）或电流为  $\pm 20\text{ mA}$ （针脚 17），或者电压为 0 到 10 V（针脚 16）或电流为 0 到 20 mA（针脚 17），和一个附加的用于方向信号的 24 V 数字输出（X2，针脚 29）共同控制的。
- 通过一个永久分配的 24V 数字输出（X2，针脚 28）控制制动器或启用驱动器。
- 例如，通过转换器可连接伺服驱动电机，或通过频率转换器可连接异步电机。
- 24 V 增量编码器用于位置反馈。
- 可组态加速度和减速度来运行。
- 首先，轴加速到指定速度。在达到离目标的指定距离内后，减速到较低速度（蠕行速度）。就在轴将要达到目标时，驱动器在指定距离处断电。CPU 可以监视此过程中的目标逼近。
- 在参数中指定速度、减速和目标逼近误差。

## 2.1.2 使用数字输出来控制定位

### 引言

CPU 支持用数字输出来控制定位（快行/爬行速度控制）。

### 特性

用数字输出控制的定位（快行/爬行速度控制）有以下特点：

- 驱动器通过 4 个永久分配的 **24 V 数字输出** 控制。这些数字输出根据组态的控制类型，来控制方向和速度阶段（快行/爬行速度）。
- 可通过接触器组合来连接多速度、电极切换电机，或通过具有固定速度设置的频率转换器连接多个异步电机。
- 24-V 增量编码器用于位置反馈。
- 首先，以指定速度（快行速度）逼近目标。在达到离目标的指定的距离内后，速度减到较低速度（爬行速度）。就在轴将要达到目标时，驱动器在指定距离处断电。CPU 可以监视此过程中的目标逼近。
- 在参数中声明速度、减速和目标逼近误差。

## 2.2 定位概述

### 概述

- 轴数
  - CPU 314C-2 DP/PtP: 1 个轴

---

### 说明

使用定位功能时，仅可使用两个额外的计数通道（通道 2 和 3）。

---

- 轴类型
  - 线性轴
  - 旋转轴
- 通常使用的驱动器/电机
  - 通过接触器组合进行电极切换的异步电机
  - 带频率转换器的异步电机
  - 带转换器的伺服电机
- 距离测量系统：
  - 24 V 增量编码器，非对称，相差为 90 度的轨迹（带或不带零标记）
- 监视功能（可分别激活）
  - 缺少脉冲（零标记）
  - 行程范围
  - 工作范围
  - 实际值
  - 目标逼近
  - 目标范围
- 单位系统
  - 所有值均以脉冲指定
- 项目设计
  - 通过参数分配窗口

## 2.3 功能范围

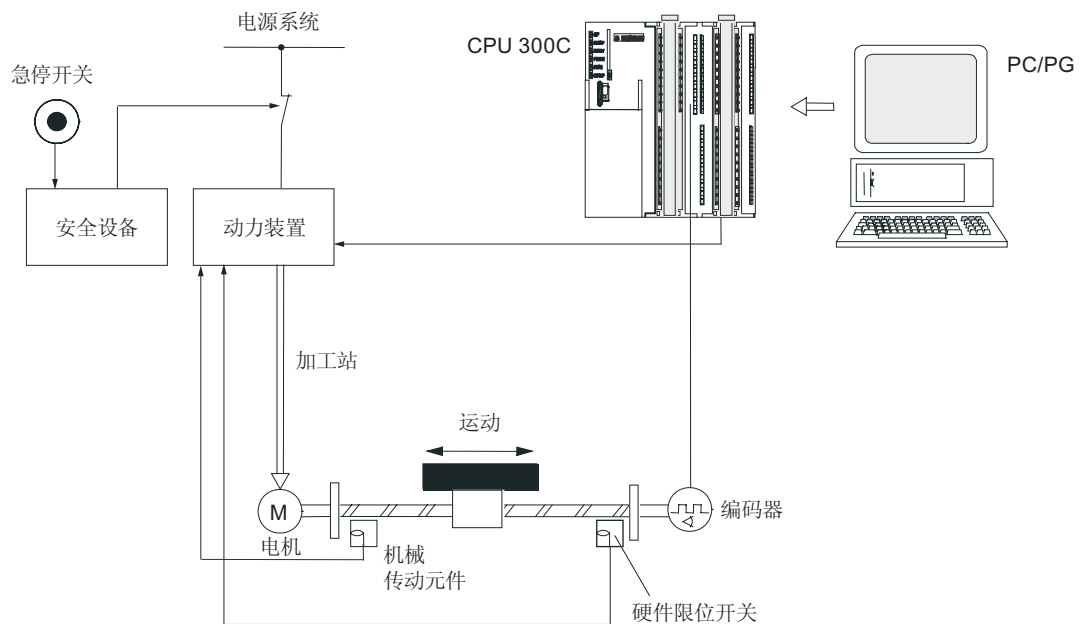
### 概述

- **操作模式:**
  - 点动模式
  - 参考点逼近
  - 相对增量逼近
  - 绝对增量逼近
- **其它功能:**
  - 设置参考点
  - 删除剩余行程
  - 长度测量

## 2.4 用于定位控制的组件

### 基本设计

下图显示了控制定位需要的组件：



- CPU 使用输出来控制转换器。
- 转换器处理定位信号并控制电机。
- 起动安全设备（紧急切断开关或硬件限位开关）时，转换器关闭电机。
- 电机在转换器控制下驱动轴。
- 编码器反馈位置和方向信息。
- 可将旋转轴或线性轴作为机械传输元件来控制。
- 使用 PG/PC
  - 在参数分配屏幕中组态 CPU，以实现 CPU 的技术功能。
  - 编程 CPU SFB，这些功能块可直接在用户程序中实现。
  - 启动 CPU 并通过标准 STEP 7 用户接口（监视功能和变量表）进行测试。





## 使用模拟输出定位

### 3.1 接线

#### 3.1.1 重要安全规则

##### 遵守安全规则

 <b>危险</b>
为符合系统的安全概念，必须安装下面提及的开关设备并使其适用于您的系统： <ul style="list-style-type: none"><li>• <b>紧急切断开关</b>。可用它关闭整个系统。</li><li>• <b>硬件限位开关</b>。直接影响所有驱动器电源装置。</li><li>• <b>电机保护</b></li></ul>

 <b>警告</b>
如果不切断电源，可能会危及人身安全和导致财产损失： 如果在带电状态下连接 CPU 的前插头，会有触电危险！ 请始终在断电状态下连接 CPU！ 缺少安全设备可能会危及人身安全和导致财产损失： 如果未安装“紧急切断开关”，连接在一起的机组可能会对设备造成损害。 安装“紧急切断开关”，使您能够切断连接的所有驱动器。

---

##### 说明

可直接连接感性负载（例如，继电器和接触器），而无需辅助电路。

如果可通过另外安装的辅助接触器（例如继电器触点）切断 SIMATIC 输出电路，则必须在感性负载线圈上另外安装浪涌电压抑制元件。

---

### 3.1 接线

#### 3.1.2 接线规则

##### 连接电缆/屏蔽

- 必须屏蔽模拟输出和 24-V 编码器的电缆。
- 连接数字 I/O 的电缆如果长度超过 100m，则必须屏蔽。
- 在电缆屏蔽的两端，必须对屏蔽层进行端接。
- 软线，横截面积为 0.25 到 1.5 mm<sup>2</sup>
- 不需要电缆套。如果坚持使用电缆套，请使用不带绝缘环的电缆套（DIN 46228，A 形，短型）。

##### 屏蔽端接元件

使用此屏蔽端接元件能够轻松完成屏蔽电缆的接地连接，因为屏蔽端接元件直接接触固定轨。

##### 其它信息

有关其它信息，请参见 *CPU 数据手册* 和 *CPU 安装说明*。

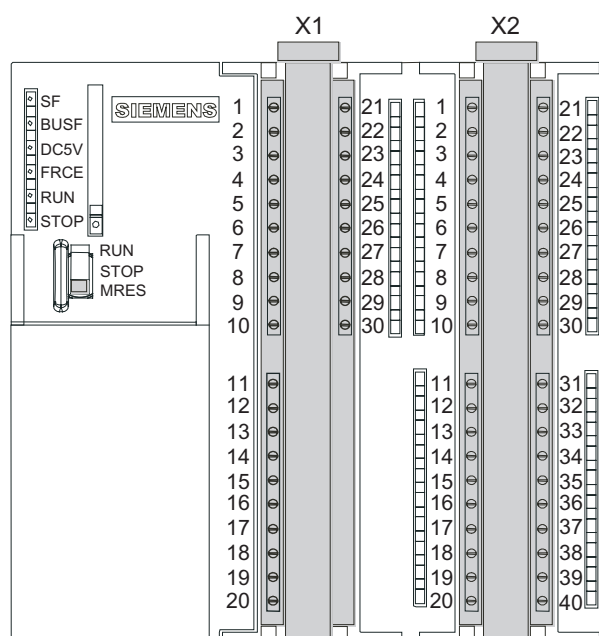


### 3.1.3 用于使用模拟输出定位的端子

#### 引言

使用 314C-2 DP/PtP CPU 的前连接器 X1 和 X2 可连接下列组件：

- 24 V 编码器
- 长度测量开关
- 参考点开关
- 动力装置



## 3.1 接线

## 针脚分配说明

下面的连接器针脚分配仅涉及与定位模式相关的连接。

## 说明

由于这些连接部分地使用相同的输入，因此在使用定位功能时不能使用计数器 0 和 1。

表格 3-1 连接器 X1 的针脚分配

连接	名称/地址	功能
1	-	未连接
2	AI 0 (V)	-
3	AI 0 (I)	-
4	AI 0 (C)	-
5	AI 1 (V)	-
6	AI 1 (I)	-
7	AI 1 (C)	-
8	AI 2 (V)	-
9	AI 2 (I)	-
10	AI 2 (C)	-
11	AI 3 (V)	-
12	AI 3 (I)	-
13	AI 3 (C)	-
14	AI R_P	-
15	AI R_N	-
16	AO 0 (V)	动力装置的电压输出
17	AO 0 (I)	动力装置的电流输出
18	AO 1 (V)	-
19	AO 1 (I)	-
20	Mana	模拟量接地
21	-	未连接
22	DI+2.0	-
23	DI+2.1	-
24	DI+2.2	-
25	DI+2.3	-
26	DI+2.4	-
27	DI+2.5	-
28	DI+2.6	-
29	DI+2.7	-
30	4 M	外壳接地

V: 电压输入/输出  
I: 电流输入/输出  
C: 公共输入

表格 3-2 连接器 X2 的针脚分配

连接	名称/地址	功能
1	1 L+	输入的 24 V 电源
2	DI+0.0	编码器信号 A
3	DI+0.1	编码器信号 B
4	DI+0.2	编码器信号 N
5	DI+0.3	长度测量
6	DI+0.4	参考点开关
7	DI+0.5	-
8	DI+0.6	-
9	DI+0.7	-
10	-	未连接
11	-	未连接
12	DI+1.0	-
13	DI+1.1	-
14	DI+1.2	-
15	DI+1.3	-
16	DI+1.4	-
17	DI+1.5	-
18	DI+1.6	-
19	DI+1.7	-
20	1 M	外壳接地
21	2 L+	输出的 24 V 电源
22	DO+0.0	-
23	DO+0.1	-
24	DO+0.2	-
25	DO+0.3	-
26	DO+0.4	-
27	DO+0.5	-
28	DO+0.6	CONV_EN: 启用动力装置
29	DO+0.7	CONV_DIR: 方向信号 *
30	2 M	外壳接地
31	3 L+	输出的 24 V 电源
32	DO+1.0	-
33	DO+1.1	-
34	DO+1.2	-
35	DO+1.3	-
36	DO+1.4	-
37	DO+1.5	-
38	DO+1.6	-
39	DO+1.7	-
40	3 M	外壳接地

\*: 此输出仅适用于控制模式“电压 0 到 10 V, 或电流 0 到 20 mA 且带方向信号”。

### 3.1 接线

#### 3.1.4 连接组件

##### 步骤

1. 关断所有组件的电源。
2. 连接输入和输出的电源：
  - 24 V 在 X2，针脚 1、21 和 31
  - 在 X1（针脚 30）和 X2（针脚 20、30 和 40）处接地
3. 将 24 V 编码器和开关连接到 24 V 电源。
4. 连接编码器信号和所需的开关（X2，针脚 2 到 6 以及针脚 20）。

可以将无反跳开关（24 V P 开关）或非接触传感器/BERO（2 或 3 线制接近开关）连接到数字输入“长度测量”和“参考点开关”。
5. 将动力装置连接到电源。
6. 使用屏蔽电缆连接动力装置的信号电缆（X1，针脚 16 或 17 及针脚 20 和 X2，针脚 28）。

如果您正使用 0 到 10 V 的电压（针脚 16）或 0 到 20 mA（针脚 17）的电流以及用于方向信号的附加 24 V 数字输出来控制动力装置，则还要使用 24 V 数字输出 CONV\_DIR（X2，针脚 29）连接相应的动力装置输入。
7. 剥掉屏蔽电缆的绝缘材料，并将电缆屏蔽连接到屏蔽连接元件。请使用屏蔽端子元件进行连接。

---

##### 说明

CPU 不会检测数字输入是否有故障。可通过激活实际值监视来检测编码器故障（请参阅『驱动器参数（页码 32）』）。

造成此类故障的原因有以下几种：

- 数字输入故障
  - 断线
  - 编码器有故障
  - 动力装置有故障
-

## 3.2 参数组态

### 3.2.1 参数组态的基本信息

#### 基本信息

可调整用于定位功能的参数，使其适应您的具体应用。可以为参数分配两种参数类型：

- **模块参数**

有一些基本设置只指定一次，在过程运行时不再更改。本节将介绍这些参数。

- 您可以在参数分配窗口（在 HW Config 中）中分配这些参数。
- 它们存储在 CPU 的系统存储器中。
- 当 CPU 处于 RUN 模式时，不能修改这些参数。

- **SFB 参数**

运行期间需要更改的参数位于系统功能块 (SFB) 的背景数据块中。在『使用模拟输出定位 - 步骤』一节中对 SFB 参数进行介绍。

- 可以在 DB 编辑器中离线编辑或在用户程序中在线编辑这些参数。
- 它们存储在 CPU 的工作存储器中。
- 可以在 CPU 处于 RUN 状态时在用户程序中修改这些参数。

### 3.2 参数组态

#### 参数分配窗口

可以在以下**参数分配窗口**中分配模块参数：

- 常规
- 地址
- 基本参数
- 驱动器
- 轴
- 编码器
- 诊断

参数分配窗口是自说明的。可以在以下各节和参数分配窗口的集成帮助中找到参数说明。

---

#### 说明

如果已为计数技术分配通道 0 或通道 1，则无法为定位技术分配参数。

---

---

#### 说明

在子模块 AI5/AO2 中禁用了输出 0 后，您只能使用模拟输出模式来组态定位。此时，您不能通过用户程序直接访问此输出。

---

#### 也参见

使用模拟输出定位 - 步骤（页码 41）

### 3.2.2 使用参数分配窗口组态参数

#### 要求

调用参数分配窗口的先决条件是已经创建了可以保存参数的项目。

#### 步骤

1. 启动 SIMATIC 管理器，在项目中调用 HW Config。
2. 双击 CPU 的“AI 5/AO 2”子模块。将模拟输出 AO 0 的输出状态设置为“禁用”。
3. 双击 CPU 的“定位”子模块。“特性”对话框打开。
4. 为“定位”子模块分配参数，然后“确定”(OK)退出参数分配窗口。
5. 使用“站 > 保存并编译”将组态保存在 HW Config 中。
6. 当 CPU 处于 STOP 模式时，使用“PLC > 下载到模块...”将参数数据下载到 CPU。现在数据即存储在 CPU 的系统数据存储器中。
7. 将 CPU 切换至 RUN 模式。

#### 在线帮助

分配参数时，参数分配窗口中的在线帮助可为您提供支持。可通过如下几种方式来调用在线帮助：

- 在相应的视图中，按下 **F1** 键。
- 在不同的参数分配窗口中单击**帮助按钮**。

### 3.2.3 基本参数

#### 中断选择参数

参数	值范围	缺省
中断选择	<ul style="list-style-type: none"> <li>• 无</li> <li>• 诊断</li> </ul>	无

在此可以指定是否触发诊断中断。在『组态和判断诊断中断（页码 78）』一节中对诊断中断进行了介绍。

### 3.2.4 驱动器参数

#### 目标范围参数

参数	值范围	缺省
目标范围	0 到 200,000,000 个脉冲 CPU 将向上舍入奇数值。	50

目标范围在目标周围对称排列。

当值为 0 时，POS\_RCD 不会置位为 TRUE，直到超出目标或达到一个脉冲的精度。

目标范围限制为：

- 旋转轴的旋转轴范围
- 线性轴的工作范围

#### 监视时间参数

参数	值范围	缺省
时间监视	<ul style="list-style-type: none"> <li>• 0 到 100 000 ms</li> <li>• 0 = 无监视</li> </ul> 舍入操作由 CPU 在 4 ms 的步骤内完成。	2000

CPU 使用此监视时间来监视

- 位置的实际值
- 目标逼近

当该值设置为“0”时，会关闭实际值和目标逼近监视。

#### 最高转速参数

参数	值范围	缺省
最高转速	10 到 1,000,000 个脉冲/秒	1000

此参数用于设置模拟输出电平和速度之间的比例关系。此处指定的最高转速与模拟输出的 10 V 或 20 mA 的电平成比例。



### 蠕行速度/参考转速参数

参数	值范围	缺省
蠕行速度/参考转速	10 到组态的最大转速	100

达到制动位置后，转速降为蠕行速度。

驱动器达到参考点开关后，转速降至参考点逼近转速。

### 关断延迟参数

参数	值范围	缺省
关断延迟	0 到 100 000 ms 舍入操作由 CPU 在 4 ms 内完成。	1000

取消运行和禁用转换器（数字输出 CONV\_EN）之间的关断延迟。

通过数字输出 CONV\_EN 控制制动器时，可使用此延迟以确保轴的转速足够慢，以便使制动器能吸收动能。

### 最大频率参数：位置反馈

参数	值范围	缺省
最大频率：位置反馈	<ul style="list-style-type: none"> <li>• 60kHz</li> <li>• 30kHz</li> <li>• 10kHz</li> <li>• 5kHz</li> <li>• 2kHz</li> <li>• 1kHz</li> </ul>	60 kHz

可用固定步长设置位置反馈信号（编码器信号 A、B、N）的最大频率。

### 最大频率参数：伴随信号

参数	值范围	缺省
最大频率：伴随信号	<ul style="list-style-type: none"> <li>• 60 kHz</li> <li>• 30 kHz</li> <li>• 10 kHz</li> <li>• 5 kHz</li> <li>• 2 kHz</li> <li>• 1 kHz</li> </ul>	10 kHz

可用固定步长设置长度测量和参考点开关信号的最大频率。

## 控制模式参数

参数	值范围	缺省
控制模式	<ul style="list-style-type: none"> <li>电压 +/-10 V 或电流 +/-20 mA</li> <li>电压 0 到 10 V 或电流 0 到 20 mA, 和方向信号</li> </ul>	电压 +/-10 V 或 电流 +/-20 mA

控制模式说明如何控制连接的转换器。

- 电压 +/-10 V 或电流 +/-20 mA:  
在正方向（向前）为运行输出正电压或电流。在负方向（反向）为运行输出负电压或电流。
- 电压 0 到 10 V 或电流 0 到 20 mA 和方向信号:  
在正方向（向前）为运行输出正电压或电流，并关闭数字输出 CONV\_DIR。  
在负方向（反向）为运行输出负电压或电流，并关闭数字输出 CONV\_DIR。

## 实际值参数

参数	值范围	缺省
实际值监视	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	是

在监视时间内，轴必须在指定方向上移动至少一个脉冲宽度。

运行开始时，实际值监视是打开的。在达到关断位置前，它一直处于激活状态。

当监视时间设置为“0”时，会关闭实际值监视。

当监视装置响应时，运行将取消。

CPU 不会检测数字输入是否有故障。可启用实际值监视来间接检测编码器或驱动器故障。

## 目标逼近监视参数

参数	值范围	缺省
目标逼近监视	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否

达到关断位置后，轴必须在监视时间内达到目标范围。

当监视时间设置为“0”时，会关闭目标逼近监视。

## 目标范围监视参数

参数	值范围	缺省
目标范围监视	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否

达到目标范围后，会对驱动器进行监视，以检查它是停留在已逼近的目标位置还是与之偏离。

当监视装置响应时，将生成一个外部出错消息。这将取消激活监视。在启动新运行前，不会重新打开监视。

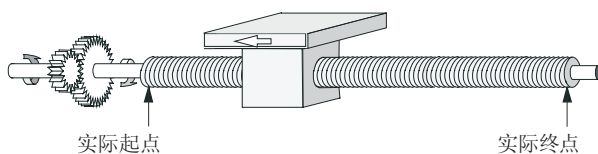
### 3.2.5 轴参数

#### 轴类型参数

参数	值范围	缺省
轴类型	<ul style="list-style-type: none"> <li>• 线性轴</li> <li>• 旋转轴</li> </ul>	线性轴

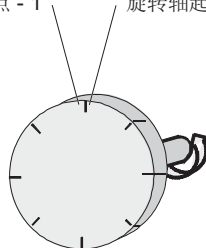
可以控制线性轴和旋转轴。

**线性轴**的最大行程范围受到机械限制。



**旋转轴**不受机械停止的限制。

可显示的最大值 = 旋转轴终点 - 1      旋转轴起点 (坐标 0) = 旋转轴终点



旋转轴的旋转从“零”坐标开始，到坐标“旋转轴终点 - 1”处结束。“零”坐标实际上与“旋转轴终点”坐标相同（都等于 0）。实际位置值显示在该点进行切换。它总是以正值形式显示。

## 软件限位开关开始/结束参数

参数	值范围	缺省
软件限位开关开始/结束	软件限位开关开始 软件限位开关结束 -5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	-100 000 000 +100 000 000

软件限位开关仅适用于线性轴。

由这些软件限位开关来限制工作范围。

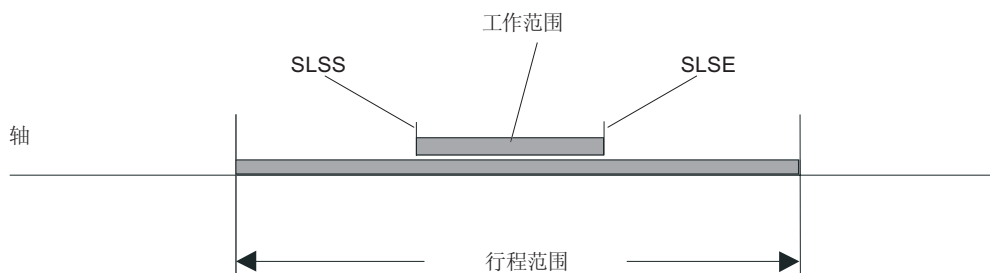
软件限位开关属于工作范围之内。

如果轴同步且已开启工作范围监视，将监视软件限位开关。

每次 CPU 进行 STOP-RUN 转换后，轴最初是不同步的。

软件限位开关开始 (SLSS) 值必须始终小于软件限位开关结束 (SLSE) 值。

工作范围必须处于行程范围之内。此行程范围表示 CPU 能够处理的值范围。



SLSS = 软件限位开关开始  
SLSE = 软件限位开关结束

## 旋转轴终点参数

参数	值范围	缺省
旋转轴终点	1 到 10 <sup>9</sup> 个脉冲	100 000

“旋转轴终点”理论上可能是最大的实际值。它的实际位置与旋转轴的起点相同（都等于“0”）。

显示的最大旋转轴值为“旋转轴终端值 - 1”。

实例：旋转轴终点 = 1,000

显示的切换情况是：

- 顺时针方向，从 999 到 0
- 逆时针方向，从 0 到 999

### 长度测量和参考点坐标参数

参数	值范围	缺省
长度测量	<ul style="list-style-type: none"> <li>关</li> <li>在正跳沿 DI 处启动/结束</li> <li>在负跳沿 DI 处启动/结束</li> <li>通过正跳沿启动，跳过负跳沿结束</li> <li>通过负跳沿启动，通过正跳沿结束</li> </ul>	关
参考点坐标	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0

CPU 进行 STOP-RUN 转换后，实际值被设置成和参考点坐标值相同。

在某个参考点逼近后，将参考点坐标值分配给该参考点。

参考点坐标值必须位于线性轴的工作范围（含软件限位开关）内。

旋转轴的参考点坐标值必须位于 0 到“旋转轴终点 - 1”范围内。

### 参考点开关参数的参考点位置

参数	值范围	缺省
参考点开关的参考点位置	<ul style="list-style-type: none"> <li>正方向（实际值增加）</li> <li>负方向（实际值减少）</li> </ul>	正方向

此参数根据参考点开关定义参考点位置。

### 行程范围监视参数

参数	值范围	缺省
行程范围监视	是（设置成固定值）	是

使用行程范围监视检查是否超出了允许的行程范围 -5 x 10<sup>8</sup> 至 +5 x 10<sup>8</sup>。不能关闭此监视功能（在“监视”参数中永久打开）。

当此监视响应时，会取消同步并中止运行。

### 工作范围监视参数

参数	值范围	缺省
工作范围监视 （仅用于线性轴）	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	是

在此可以指定是否监视线性轴的工作范围。此时，将监视实际位置值以检查它是否超出了软件限位开关的范围。此监视只影响同步的轴。

软件限位开关自身的坐标属于工作范围。

当监视功能响应时，将取消运行。

### 3.2.6 编码器参数

#### 编码器每转增量参数

参数	值范围	缺省
编码器每转增量	1 到 2 <sup>23</sup> 个脉冲	1000

“编码器每转增量”参数指定编码器每旋转一周输出的增量。有关数值的信息，请参见编码器说明。

CPU 将判断增量四次（一个增量对应着四个脉冲，请参阅『增量编码器（页码 81）』一节）。

#### 计数方向参数

参数	值范围	缺省
计数方向	<ul style="list-style-type: none"> <li>• 常规</li> <li>• 反转</li> </ul>	常规

使用“计数方向”参数可将路径监视的方向调整为线性轴的运动方向。还应考虑所有传动元件（例如，联接器和齿轮）的旋转方向。

- 常规 = 增加计数脉冲 = 减少实际值
- 反转 = 增加计数脉冲 = 减少实际值

#### 缺少脉冲（零标记）监视参数

参数	值范围	缺省
缺少脉冲（零标记）监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否

启用零标记监视时，CPU 将监视两个连续零标记信号（编码器信号 N）之间脉冲差异的一致性。

如果已组态的编码器旋转一周产生的脉冲数不能被 10 或 16 整除，则无论在参数分配窗口中的设置如何，都会自动关闭零标记监视。

#### 说明

零标记信号的最小脉冲宽度为 8.33 μs（对应于最大频率 60 kHz）。

如果使用的编码器的零标记信号通过“AND”操作与编码器信号 A 和 B 合并，脉度将减半至周期时间的 25%。这会将零标记监视的最大频率减小为 30kHz。

不能识别以下内容：

- 分配了不正确的编码器每转增量数。
- 零标记信号失效。

当此监视响应时，会取消同步并中止运行。

### 3.2.7 组态诊断

#### 为监视启用诊断中断

响应监视功能可触发诊断中断。

要求：在“基本参数”屏幕启用了诊断中断，并在“驱动器”、“轴”和“编码器”屏幕激活了相应的监视。

参数	值范围	缺省
缺少脉冲（零标记）	<ul style="list-style-type: none"><li>• 是</li><li>• 否</li></ul>	否
行程范围	<ul style="list-style-type: none"><li>• 是</li><li>• 否</li></ul>	否
工作范围（对于线性轴）	<ul style="list-style-type: none"><li>• 是</li><li>• 否</li></ul>	否
实际值	<ul style="list-style-type: none"><li>• 是</li><li>• 否</li></ul>	否
目标逼近	<ul style="list-style-type: none"><li>• 是</li><li>• 否</li></ul>	否
目标范围	<ul style="list-style-type: none"><li>• 是</li><li>• 否</li></ul>	否

### 3.3 集成到用户程序中

#### 步骤

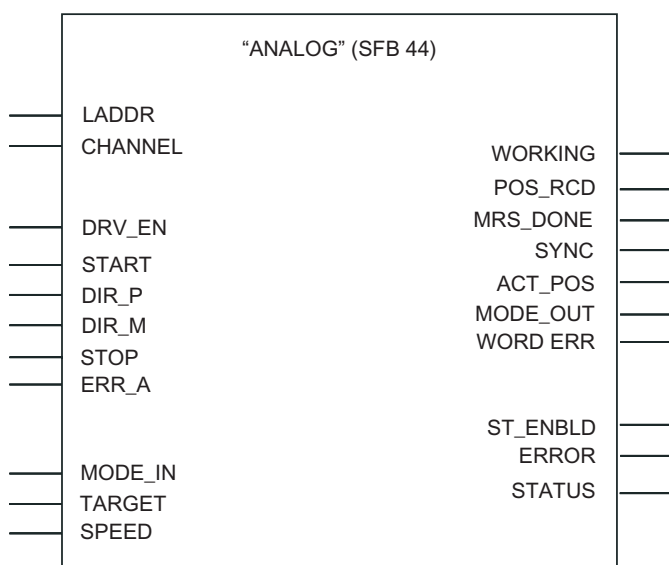
在用户程序中控制定位功能。为此，可调用系统功能块 **SFB ANALOG (SFB 44)**。SFB 在“标准库”的“系统功能块 > 块”下。

以下各节可帮助您为自己的应用设计用户程序。

#### 调用 SFB

通过相应的背景 DB 调用 SFB

实例：CALL SFB 44, DB20



#### 说明

因为 SFB 不能中断本身，所以不能在其它优先级下在另一程序段中调用已在程序中组态的 SFB。

实例：不允许在 OB 1 和中断 OB 中调用同一 SFB。

#### 背景 DB

SFB 参数存储在背景数据块中。在『SFB ANALOG (SFB 44) 的基本组态（页码 47）』一节中介绍这些参数。可以通过以下方式访问这些参数：

- DB 号和 DB 中的绝对地址
- DB 号和 DB 中的符号地址

该功能的主要参数也将与该块互联。可在 SFB 中直接分配输入参数值，也可以判断输出参数。



## 3.4 使用模拟输出定位的功能

### 3.4.1 使用模拟输出定位 - 步骤

#### 概述

永久分配的模拟输出（模拟输出 0）通过 -10 V 至 +10 V 或 0 V 到 10 V 的电压（电压信号）外加一个数字输出 **CONV\_DIR**，或者 -20 mA 至 +20 mA 或 0 mA 至 20 mA 的电流（电流信号）外加一个数字输出 **CONV\_DIR** 来控制驱动器。

定位反馈通过一个非对称 24 V 增量编码器实现，该编码器输出两个相位偏移量为 90° 的信号。

数字输出 **CONV\_EN** 用于启用和关闭动力装置和/或控制制动器。

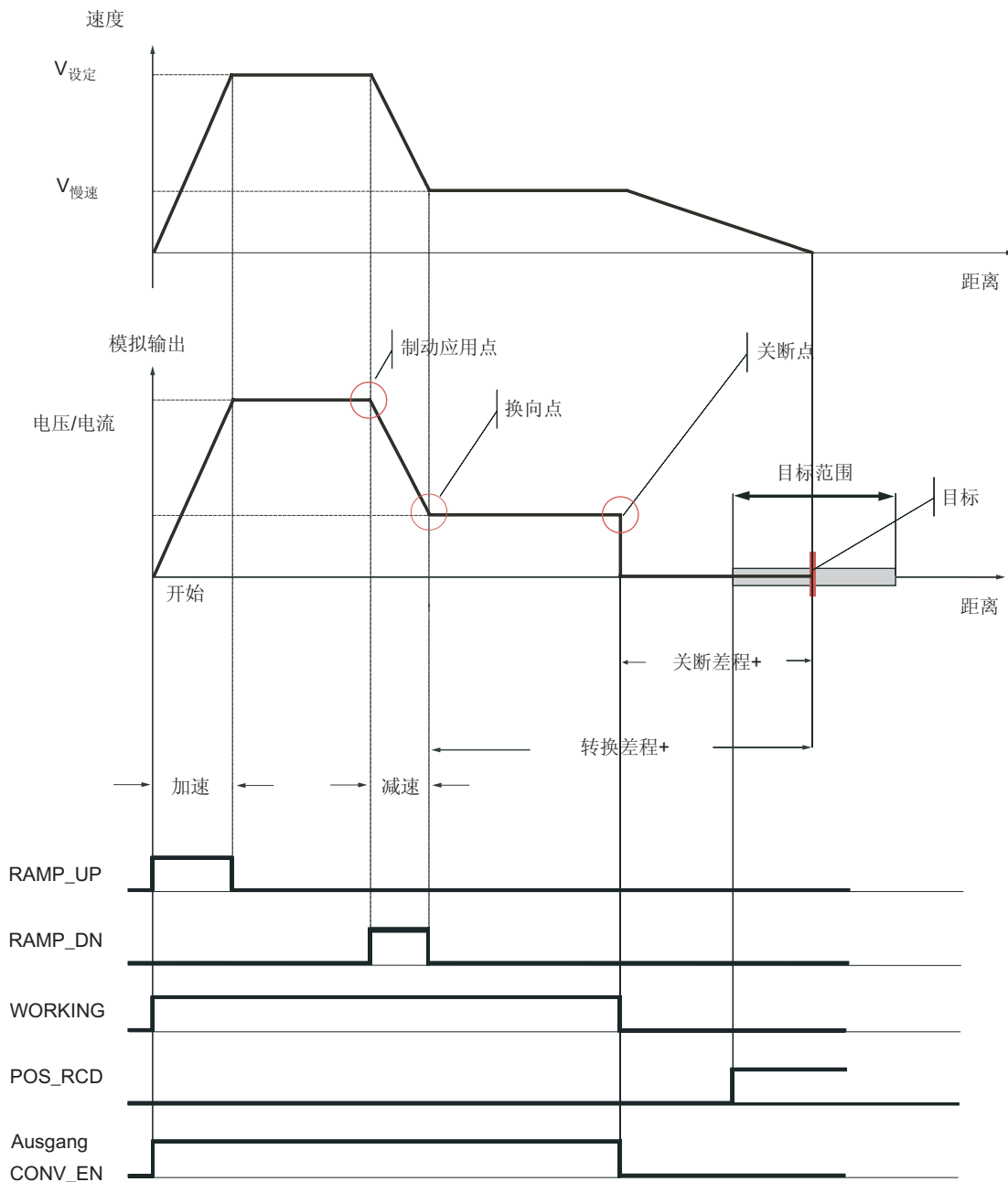
#### 启动运行

根据操作模式，使用 **START**、**DIR\_P** 或 **DIR\_M** 启动运行。

### 使用模拟输出定位

下图的上部分显示了运行曲线。我们仅假定在移动过程中实际速度只作线性变化。

该图的下部显示模拟输出的相应电压/电流曲线。



- 在加速阶段 (RAMP\_UP) 之后, 最初以速度  $V_{\text{设定}}$  逼近目标。
- 在由 CPU 计算的**制动点**处, 将激活减速阶段 (RAMP\_DN), 直到达到换向点。
- 在达到**换向点**后, 会立即以蠕行速度 ( $V_{\text{慢速}}$ ) 继续运行。
- 驱动器在**关断点**被关闭。
- 要接近的每个目标的转换点和关断点由在参数中指定的**转换差程**和**关断差程**的值决定。对于前向 (正方向) 和反向 (负方向) 运行, 可指定不同的转换差程和关断差程。
- 达到关断点时, 会终止运行 (WORKING= FALSE)。此时, 可启动一个新运行。
- 当实际位置值达到**目标区域**时, 即达到了指定目标 (POS\_RCD = TRUE)。如果在没有启动新运行的情况下实际位置值偏离目标区域, 则不会复位“已到位”信号。
- 如果转换差程小于关断差程, 则会从制动点开始, 线性减速, 直至速度设定值 0。

### 启用动力装置 (CONV\_EN)

数字输出 CONV\_EN 用于启用和禁用动力装置或控制制动器。运行开始时置位输出, 运行结束 (在关断点或速度设定值 = 0) 时复位输出。

使用数字输出控制制动器时, 必须考虑在复位输出 (在关断点或速度设定值 = 0) 时, 制动器必然会吸收驱动器的动能。

### 方向信号 (CONV\_DIR)

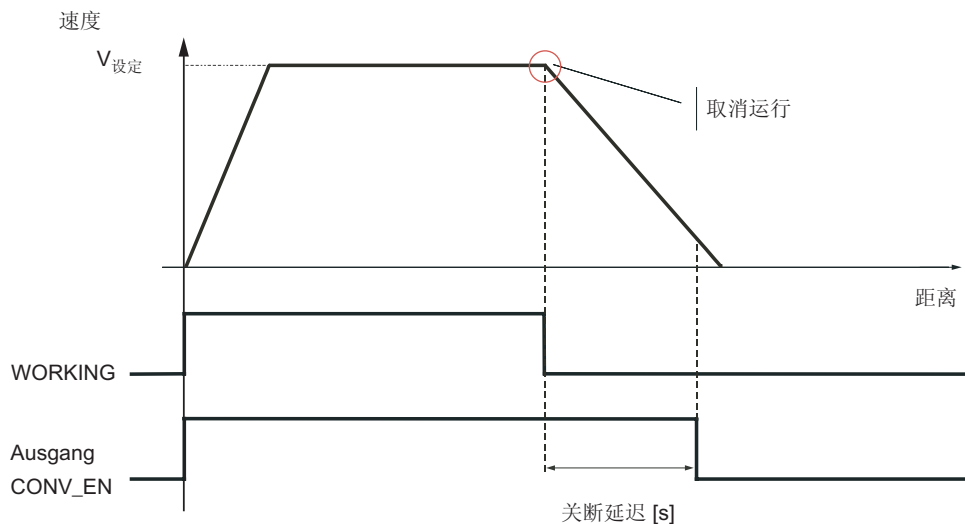
在控制模式“电压 0 到 10 V 或电流 0 到 20 mA 且带方向信号”下, 24 V 数字输出 CONV\_DIR 提供方向信息。

- 正方向运行 (向前): DO 关闭
- 负方向运行 (反向): DO 打开

### 取消运行时关断延迟

在参数分配窗口的“关断延迟”参数中，可将延迟时间（仅当运行中止时激活）组态为在取消运行和复位数字输出 CONV\_EN 之间激活。

这可确保复位输出时轴的速度足够慢，使制动器能够吸收轴的动能。

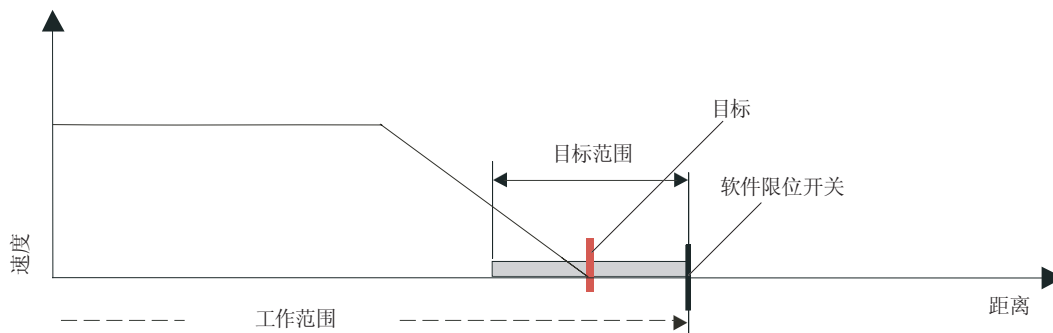


### 工作范围

借助软件限位开关坐标来确定工作范围。运行不得超过同步线性轴的工作范围。

必须始终根据工作范围指定运行目标。

当轴超出工作范围之后，只能以点动模式返回。



## 监视功能

参数分配窗口可帮助您分别启用各个监视功能。当其中一个监视功能响应时，运行会取消且发生外部错误（使用 ERR\_A 确认）。

监视	说明
缺少脉冲（零标记）	<p>启用零标记监视时，CPU 将检查两个连续零标记信号之间脉冲差异的一致性。</p> <p>如果已组态的编码器每转的脉冲数不能被 10 或 16 整除，则无论在参数分配窗口中的设置如何，都会自动关闭零标记监视。</p> <p>零标记信号的最小脉冲宽度为 8.33 <math>\mu</math>s（对应于最大频率 60 kHz）。</p> <p>如果使用的编码器的零标记信号通过“AND”操作与编码器信号 A 和 B 合并，脉度将减半至周期时间的 25%。这会将零标记监视的最大频率减小为 30 kHz。</p> <p>不能识别以下内容：</p> <ul style="list-style-type: none"> <li>• 分配了不正确的编码器每转增量数。</li> <li>• 零标记信号出错。</li> </ul> <p>CPU 对错误的响应：取消同步，取消运行。</p>
行程范围	<p>CPU 使用行程范围监视检查是否超出了允许的行程范围 <math>-5 \times 10^8</math> 至 <math>+5 \times 10^8</math>。不能关闭此监视功能（在“监视”参数中永久打开）。</p> <p>CPU 对错误的响应：取消同步，取消运行。</p>
工作范围	<p>CPU 使用行程范围监视检查实际值是否超出了软件限位开关的范围。</p> <p>对于旋转轴定位监视，不能打开此装置。此监视只影响同步的轴。软件限位开关自身的坐标属于工作范围。</p> <p>CPU 对错误的响应：运行被取消。</p>
实际值	<p>在监视时间内，移动轴必须在指定方向经过至少一个脉冲的距离。运行开始时，实际值监视是打开的。在达到关断点前，它一直处于激活状态。</p> <p>当监视时间设置为“0”时，会关闭实际值监视。</p> <p>当监视功能响应时，将取消运行。</p> <p>CPU 对错误的响应：运行被取消。</p>
目标逼近	<p>达到关断点后，轴必须在监视时间内达到目标范围。当监视时间设置为“0”时，会关闭目标逼近监视。</p> <p>CPU 对错误的响应：运行被取消。</p>
目标范围	<p>达到目标范围后，CPU 会对驱动器进行监视，以检查它是停留在已逼近的目标位置还是与之偏离。</p> <p>当监视功能响应时，将生成一个外部错误消息。当应答外部错误 ERR_A（肯定应答）时，将关闭监视。在启动新运行前，不会重新打开监视。</p> <p>CPU 对错误的响应：运行被取消。</p>

## 终止运行

共有以下三种不同的方法可以终止运行：

- 目标逼近
- 取消激活
- 取消

### 目标逼近：

目标逼近表示达到指定目标后自动终止运行。

为达到指定目标，会以“相对和绝对增量逼近”操作模式执行目标逼近。

### 取消激活：

在以下情况下将取消激活驱动器：

- 在  $STOP = TRUE$ （达到目标前）时的所有操作模式中
- 在点动模式下，当停止和反向运行时
- 在参考点逼近模式下，当检测到同步位置或反向运行时

取消激活时，将线性减速到速度设定值 0。

### 取消：

无论转换/关断差程如何，都会立即终止运行。模拟输出直接切换到速度设定值 0。

可以随时取消，也可以在停止时取消。

在以下情况下将取消运行：

- 删除驱动器使能信号 ( $DRV\_EN = FALSE$ )
- CPU 切换为 STOP 模式时
- 发生外部错误时（例外：目标逼近/目标范围监视）

### 取消的结果：

- 立即终止当前或停止的运行 ( $WORKING = FALSE$ )
- 最后一个目标 ( $LAST\_TRG$ ) 被设置为实际值 ( $ACT\_POS$ )
- 删除剩余行程，即不能恢复“相对增量逼近”
- 不会置位“已到位” ( $POS\_RCD$ )
- 使用切断延迟复位数字输出  $CONV\_EN$ （启用动力装置）

### 3.4.2 SFB ANALOG (SFB 44) 的基本组态

#### 基本参数概述:

本部分将介绍对于所有操作模式都相同的参数。操作模式特定的参数在特定模式下介绍。

根据您的应用分配以下 SFB 输入参数。

输入参数:

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
LADDR	WORD	0	在“HW Config”中指定的子模块 I/O 地址。 如果输入和输出地址不相同, 您必须指定两者中较低的地址。	CPU 特定	310 (十六进制)
CHANNEL	INT	2	通道号	0	0
STOP	BOOL	4.4	停止运行 STOP = TRUE 可用于提前停止/中断运行。	TRUE/FALSE	FALSE
ERR_A	BOOL	4.5	组错误确认 ERR_A 用于确认外部错误 (正跳沿)。	TRUE/FALSE	FALSE
SPEED	DINT	12	将轴的速度增加到 $V_{\text{设定值}}$ 。 在运行过程中速度不能有变化。	蠕行速度最高为 1 000 000 个脉冲/秒 达到最大速度	1000

未分配给块的输入参数 (静态局部数据)

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
ACCEL	DINT	30	加速度 在运行过程中不能有变化。	1 到 100 000 个脉冲/秒 <sup>2</sup>	100
DECEL	DINT	34	减速度 在运行过程中不能有变化。	1 到 100 000 个脉冲/秒 <sup>2</sup>	100
CHGDIFF_P	DINT	38	正方向转换差程: “正方向转换差程”定义了转换点, 从该点开始, 驱动器将继续以蠕行速度向前运行。	0 到 $+10^8$ 个脉冲	1000
CUTOFF-DIFF_P	DINT	42	正方向关断差程: “正方向关断差程”定义了关断点, 在正方向以蠕行速度运行时, 驱动器将在该点处关闭。	0 到 $+10^8$ 个脉冲	100
CHGDIFF_M	DINT	46	负方向转换差程: “负方向转换差程”定义转换点, 从该点开始, 驱动器将继续以蠕行速度反向运行。	0 到 $+10^8$ 个脉冲	1000
CUTOFF-DIFF_M	DINT	50	负方向关断差程: “正方向关断差程”定义了关断点, 在负方向以蠕行速度运行时, 驱动器将在该点处关闭。	0 到 $+10^8$ 个脉冲	100

## 转换/关断差程的规则

- 正值和负值可以不同。
- 当转换差程小于关断差程时，驱动器速度会从制动位置开始，按线性减小到速度设定值 0。
- 关断差程必须大于或等于目标范围的一半。
- 转换差程必须大于或等于目标范围的一半。
- 在转向点和关断点之间选择足够的距离，以确保驱动器速度可减小到蠕行速度。
- 在转向点和目标之间选择足够的距离，以确保驱动器达到目标区域并在那里停止。
- 要移动的距离必须至少等于关断差程。
- 转换/关断差程被限制为行程范围 (+10%) 的 1/10。

输出参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
WORKING	BOOL	16.0	正在运行	TRUE/FALSE	FALSE
ACT_POS	DINT	18	当前实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	22	激活/设置操作模式	0、1、3、4、5	0
ERR	WORD	24	外部错误： <ul style="list-style-type: none"> <li>• 位 2：缺少脉冲监视</li> <li>• 位 11：行程范围监视（始终为 1）</li> <li>• 位 12：工作范围监视</li> <li>• 位 13：实际值监视</li> <li>• 位 14：目标逼近监视</li> <li>• 位 15：目标范围监视</li> <li>• 其它位将保留</li> </ul>	每个位 0 或 1	0
ST_ENBLD	BOOL	26.0	如果符合以下所有条件，CPU 将置位“启动使能”： <ul style="list-style-type: none"> <li>• 无错参数分配 (PARA = TRUE)</li> <li>• 无 STOP 挂起 (STOP = FALSE)</li> <li>• 没有发生外部错误 (ERR = 0)</li> <li>• 置位了“驱动器使能” (DRV_EN = TRUE)</li> <li>• 无激活的定位运行 (WORKING = FALSE) 例外：点动模式</li> </ul>	TRUE/FALSE	TRUE
ERROR	BOOL	26.1	运行启动/恢复错误	TRUE/FALSE	FALSE
STATUS	WORD	28	错误 ID	0 到 FFFF (十六进制)	0



未分配给块的输出参数（静态局部数据）

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
PARA	BOOL	54.0	轴已组态	TRUE/FALSE	FALSE
DIR	BOOL	54.1	当前/上一个方向的含义 FALSE = 向前（正方向） TRUE = 反向（负方向）	TRUE/FALSE	FALSE
CUTOFF	BOOL	54.2	驱动器位于关断范围内 （从关断点到下一运行开始）	TRUE/FALSE	FALSE
CHGOVER	BOOL	54.3	驱动器位于转换范围内 （从达到换向点后，直到下一运行开始）	TRUE/FALSE	FALSE
RAMP_DN	BOOL	54.4	驱动器减速（从制动点开始，直到换向点）	TRUE/FALSE	FALSE
RAMP_UP	BOOL	54.5	驱动器加速（从开始到达到最终速度）	TRUE/FALSE	FALSE
DIST_TO_GO	DINT	56	实际剩余行程	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
LAST_TRG	DINT	60	上一/实际目标 <ul style="list-style-type: none"> <li>绝对增量逼近： 运行启动 LAST_TRG = 实际绝对目标 (TARGET) 时</li> <li>相对增量逼近： 运行启动时的距离为 LAST_TRG = 前一运行的 LAST_TRG +/- (TARGET) 所指定的距离。</li> </ul>	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0

### 3.4.3 点动模式

#### 说明

在“点动”模式下，可在正方向或负方向上运行驱动器。未指定目标。

#### 要求

- 您已通过参数分配窗口分配模块参数，并将它们下载到了 CPU (PARA = TRUE)。
- 您已按说明分配 SFB 的基本参数。
- 未产生外部错误 ERR。您必须通过 ERR\_A（正跳沿）来确认排队的外部错误。
- 启动已使能 (ST\_ENBLD = TRUE)。
- 在点动模式下，既可操作同步 (SYNC = TRUE) 轴也可操作非同步 (SYNC = FALSE) 轴。

#### 启动/停止运行

通过置位控制位 DIR\_P 或 DIR\_M 启动驱动器。

- 每次调用 SFB 时，都会对 DIR\_P 和 DIR\_M 这两个控制位进行评估以便检查逻辑级的更改。
- 如果这两个控制位均为 FALSE，则运行将减速。
- 如果这两个控制位均为 TRUE，则运行也将减速。
- 当其中一个控制位置位为 TRUE 时，轴将在相应方向上移动。

步骤

- 按“设置”列中所指定的内容分配以下 SFB 输入参数：

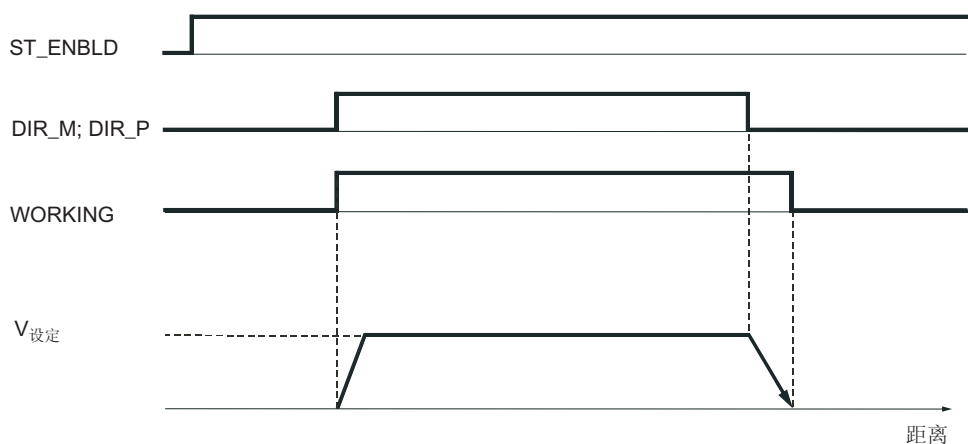
参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
DRV_EN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE	TRUE
DIR_P	BOOL	4.2	点动模式，正方向（正跳沿）	TRUE/FALSE	FALSE	DIR_P 或 DIR_M = TRUE
DIR_M	BOOL	4.3	点动模式，负方向（正跳沿）	TRUE/FALSE	FALSE	
MODE_IN	INT	6	操作模式，1 = 点动模式	0、1、3、4、5	1	1

- 调用 SFB。

SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
WORKING	BOOL	16.0	正在运行	TRUE/FALSE	FALSE
ACT_POS	DINT	18	当前实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	22	激活/设置操作模式	0、1、3、4、5	0

- 启动运行后将立即置位 WORKING = TRUE。当复位方向位 DIR\_P 或 DIR\_M 或置位 STOP = TRUE 时，运行将被终止 (WORKING = FALSE)。
- 如果在解释 SFB 调用时发生错误，则 WORKING = FALSE 且 ERROR 将置位为 TRUE。然后将使用 STATUS 参数指出准确的错误原因（请参阅『错误列表（页码 83）』一节）。
- 在点动模式下，ST\_ENBLD 始终置位为 TRUE。
- 不会置位“已到位” (POS\_RCD)。



### 3.4.4 参考点逼近

#### 说明

CPU 打开后，位置值 `ACT_POS` 并不是指向轴的机械位置。

为了向物理位置分配一个可复写的编码器值，必须在轴位置和编码器值之间建立一个参考（同步）。可通过为轴的某个已知位置（参考点）分配一个位置值来实现同步。

#### 参考点开关和参考点

轴上需要有参考点开关和参考点，这样才能执行参考点逼近。

- 需要**参考点开关**以确保参考信号始终具有相同的参考点（零标记），以及用于更改为参考点逼近速度。例如，可使用 `BERO` 开关。为使轴在超出参考点开关的范围之前达到参考点逼近速度，参考点开关的信号长度必须足够高。
- **参考点**是离开参考点开关之后的下一个编码器零标记。轴在参考点处同步，反馈信号 `SYNC` 将被置位为 `TRUE`。将在参数分配窗口中指定的参考坐标分配给参考点。

必须始终在参考点开关的方向上启动参考点逼近。否则，轴将进入范围限制，由于它未同步，因而没有软件限位开关。

通过在参考点开关处启动参考点逼近，可始终确保在开关的方向上启动轴（请参见实例 3）。

---

#### 说明

对于旋转轴：因为参考点要求的再现性，编码器的相应零标记必须始终位于相同的物理位置。因此，“旋转轴终点”和“编码器每转增量”必须为比例积分。实例：编码器转数与旋转轴终点的转数比为 4:1。在此情况下，零标记位于 90、180、270 和 360 度处。

零标记信号的最小脉冲宽度为  $8.33 \mu\text{s}$ （对应于最大频率 60 kHz）。

如果使用的编码器的零标记信号通过“AND”运算与编码器信号 A 和 B 合并，脉冲宽度将减半至周期时间的 25%。从而使最大计数频率在参考期间减小为 30 kHz。

---

## 参考点位置

关于在参考点逼近期间的参考点位置（零标记信号），必须区分以下两种情况：

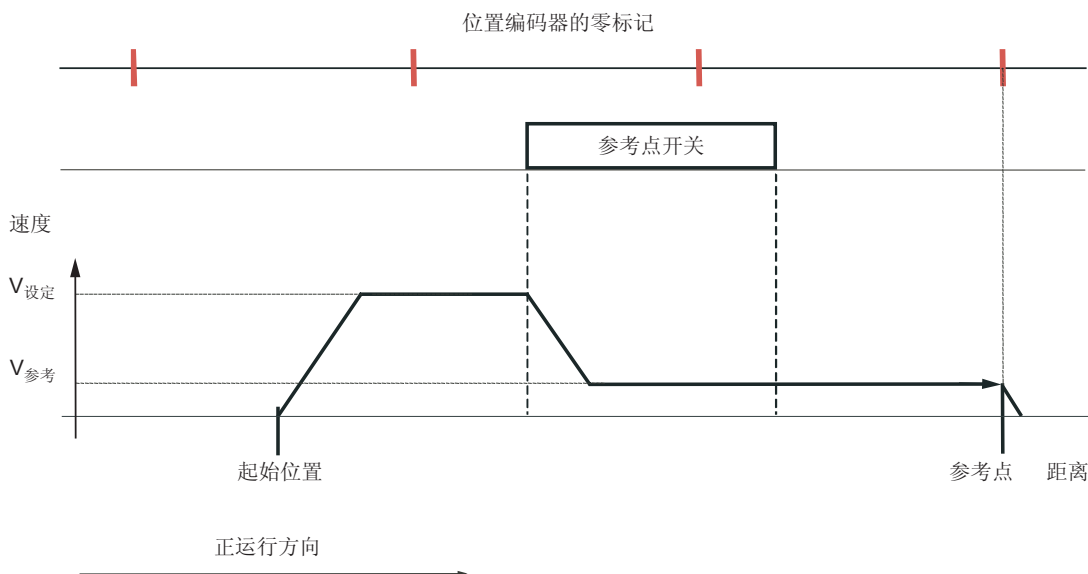
- 在正方向参考点位置和参考点开关之间的关系。
- 在负方向参考点位置和参考点开关之间的关系。

请通过参数分配窗口在参数“参考点开关的参考点位置”中进行此设置。

参考点逼近的不同情况由运行启动的方向和参考点位置来确定：

### 实例 1

- 启动方向为正方向
- 参考点在正方向上逼近参考点开关



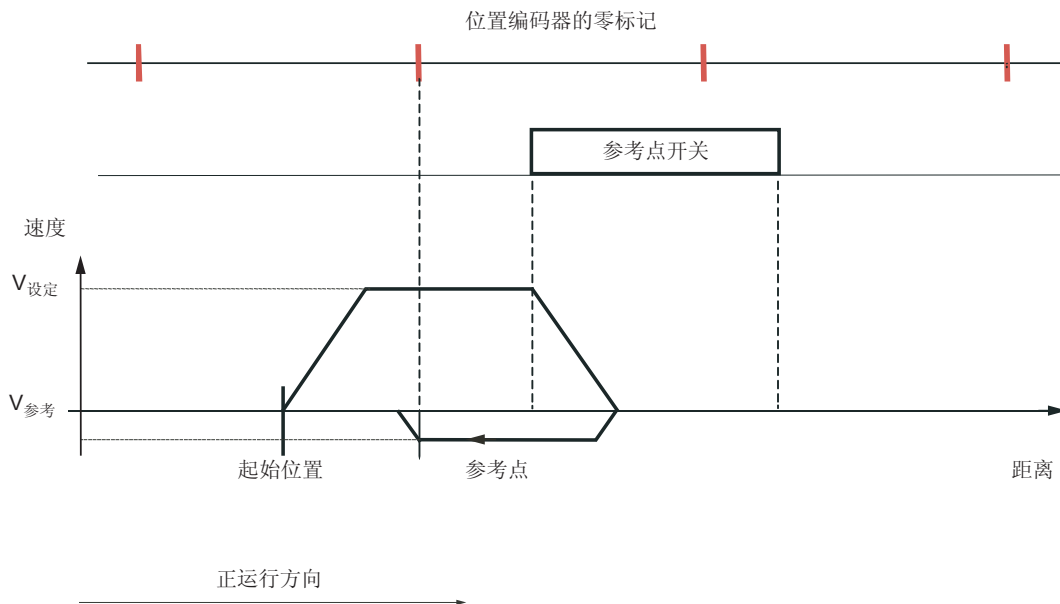
按照在 SPEED 参数中指定的速度  $V_{\text{设定}}$ ，执行朝向参考点开关的运行。

然后，速度降为参考运行速度  $V_{\text{参考}}$ 。

离开参考点开关之后，在编码器的下一个零标记处，速度将切换为零。

实例 2

- 启动方向为正方向
- 参考点在负方向上逼近参考点开关



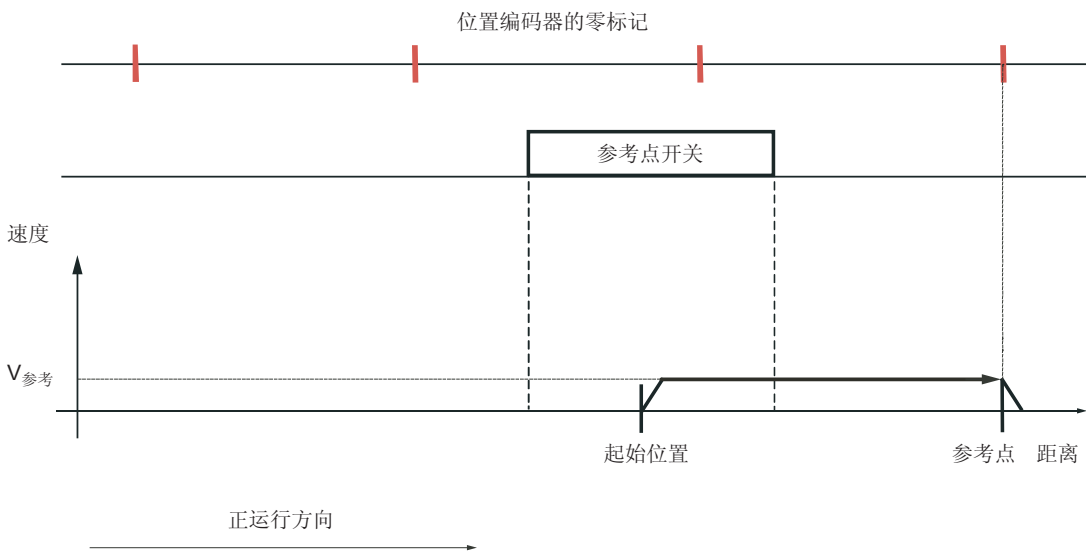
按照在 SPEED 参数中指定的速度  $V_{\text{设定}}$  执行朝向参考点开关的运行。

速度减小为零，方向反向，且以参考速度  $V_{\text{参考}}$  继续运动。

离开参考点开关之后，在编码器的下一个零标记处，速度将切换为零。

### 实例 3

- 启动位置位于参考点开关处
- 启动方向为负方向
- 参考点在正方向上逼近参考点开关



运行以参考运行速度  $V_{\text{参考}}$  执行。

运行将朝着您在参数分配窗口中使用参数“参考点开关的参考点位置”中指定的方向执行，而与 SFB 中指定的方向无关。

离开参考点开关之后，在编码器的下一个零标记处，速度将切换为零。

## 3.4.5 参考点逼近 - 步骤

## 参考点逼近的先决条件

- 具有零标记的编码器，或参考点信号开关（使用不具有零标记的编码器时）。
- 已连接参考点开关（连接器 X2，针脚 6）。
- 您已通过参数分配窗口分配模块参数，并将它们下载到了 CPU (PARA = TRUE)。
- 您已按说明分配 SFB 的基本参数。
- 未发生外部错误 ERR。您必须通过 ERR\_A（正跳沿）来确认排队的外部错误。
- 启动已使能 (ST\_ENBLD = TRUE)。

## 步骤

- 按“设置”列中所指定的内容分配以下 SFB 输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
DRV_EN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE	TRUE
DIR_P	BOOL	4.2	参考点逼近，正方向（正跳沿）	TRUE/FALSE	FALSE	DIR_P 或
DIR_M	BOOL	4.3	参考点逼近，负方向（正跳沿）	TRUE/FALSE	FALSE	DIR_M = TRUE
MODE_IN	INT	6	操作模式，3 = 参考点逼近	0、1、3、4、5	1	3

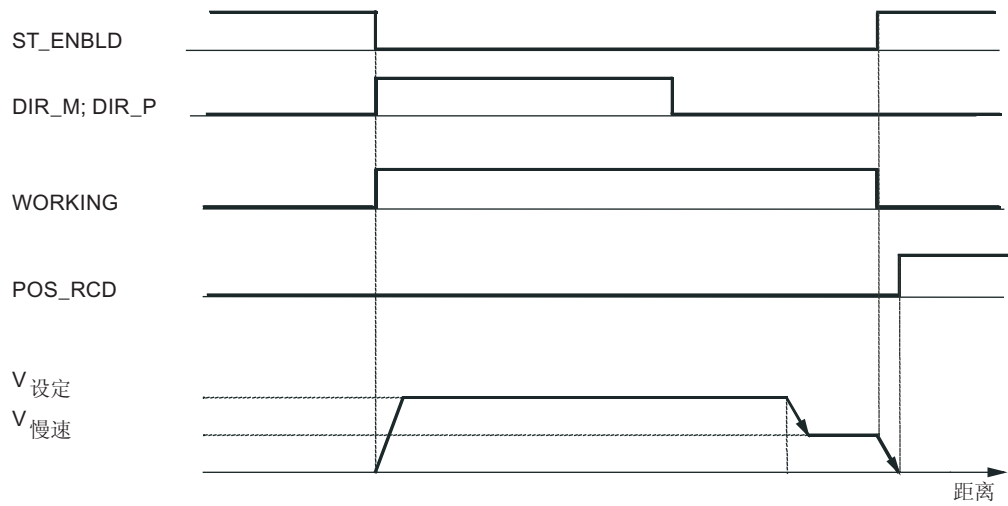
- 调用 SFB。

SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
WORKING	BOOL	16.0	正在运行	TRUE/FALSE	FALSE
SYNC	BOOL	16.3	SYNC = TRUE: 轴同步	TRUE/FALSE	FALSE
ACT_POS	DINT	18	当前实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	22	激活/设置操作模式	0、1、3、4、5	0



- 启动运行后立即置位 **WORKING = TRUE** 以及 **SYNC = FALSE**。达到参考点后，**WORKING** 的状态将复位为 **FALSE**。如果执行时未产生错误，则 **SYNC = TRUE**。
- 在启动下一个运行前，必须复位方向位 (**DIR\_P** 或 **DIR\_M**)。
- 如果在解释 **SFB** 调用时发生错误，则 **WORKING = FALSE** 且 **ERROR** 将置位为 **TRUE**。然后将使用 **STATUS** 参数指出确切的出错原因。
- “已到位” (**POS\_RCD**) 不会置位。



### 操作模式的影响有哪些

- 启动参考点逼近时，将清除可能存在的同步 (**SYNC = FALSE**)。
- 将在参考点（零标记）的正跳沿将实际位置设置为参考点坐标值，且将置位反馈信号 **SYNC**。
- 将在轴上确定工作范围。
- 位于工作范围内的所有点将保持其各自的原始坐标，但具有新的物理位置。

### 3.4.6 相对增量逼近模式

#### 说明

在“相对增量逼近”模式下，驱动器将自上一个目标 (LAST\_TRG) 开始，在指定方向上移动一段相对距离。

起始点不是实际位置，而是上一个指定目标 (LAST\_TRG)。因此，定位精度不会累加。启动定位后，由参数 LAST\_TRG 来指示实际目标。

#### 要求

- 您已通过参数分配窗口分配模块参数，并将它们下载到了 CPU (PARA = TRUE)。
- 您已根据『SFB ANALOG (SFB 44) 的基本组态 (页码 47)』一节所述分配了 SFB 的基本参数。
- 未发生外部错误 ERR。您必须通过 ERR\_A (正跳沿) 来确认排队的外部错误。
- 启动已使能 (ST\_ENBLD = TRUE)。
- 同步 (SYNC = TRUE) 和非同步 (SYNC = FALSE) 轴均可实现“相对增量逼近”。

#### 指定行程

指定线性轴行程时，请注意以下事项：

- 行程必须大于等于关断差程。
- 如果行程小于等于目标范围的一半，将不会启动新运行。此模式将立刻终止且不会产生错误。
- 目标范围必须处于工作范围之内。

## 步骤

- 按“设置”列中所指定的内容分配以下 SFB 输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
DRV_EN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE	TRUE
DIR_P	BOOL	4.2	正方向运行（正跳沿）	TRUE/FALSE	FALSE	DIR_P 或 DIR_M = TRUE
DIR_M	BOOL	4.3	负方向运行（正跳沿）	TRUE/FALSE	FALSE	
MODE_IN	INT	6	操作模式，4 = 相对增量逼近	0、1、3、4、5	1	4
TARGET	DINT	8	以脉冲表示的距离（仅允许正值）	0 到 10 <sup>9</sup> 脉冲	1000	xxxx

- 调用 SFB。

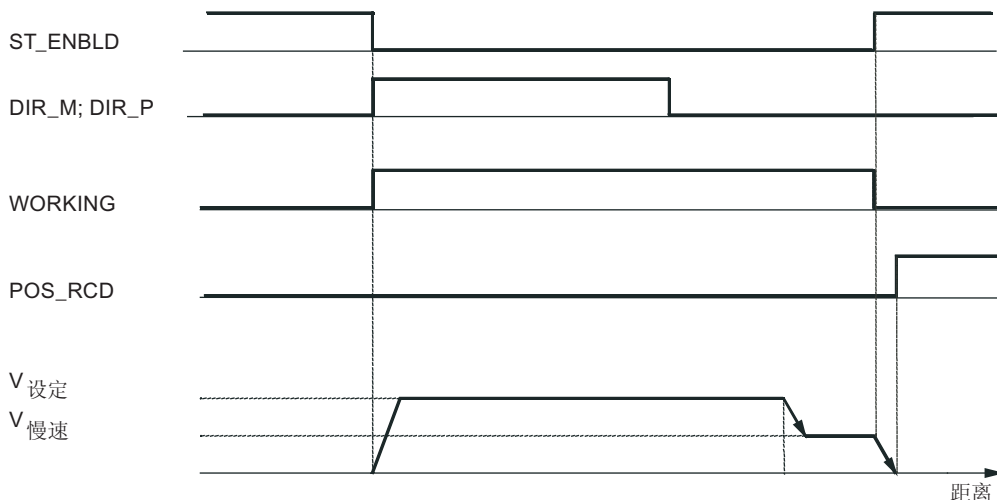
SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
WORKING	BOOL	16.0	正在运行	TRUE/FALSE	FALSE
POS_RCD	BOOL	16.1	已到位	TRUE/FALSE	FALSE
ACT_POS	DINT	18	当前实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	22	激活/设置操作模式	0、1、3、4、5	0

- 启动运行后将立即设置 WORKING = TRUE。在关断点处将 WORKING 复位为 FALSE。达到指定目标时，将设置 POS\_RCD = TRUE。
- 启动下一个运行之前，必须复位方向位（DIR\_P 或 DIR\_M）。

3.4 使用模拟输出定位的功能

- 如果在解释 SFB 调用时发生错误，则 WORKING = FALSE 且 ERROR 将设置为 TRUE。然后将使用 STATUS 参数指出准确的错误原因（请参阅『错误列表（页码 83）』一节）。



中断运行，并且目标范围未达到

使用 STOP = TRUE 停止运行时，如果尚未达到关断范围（剩余行程大于关断差程），则根据后续操作模式/作业的不同，有以下几种选项。

选项	响应
在同一方向继续该运行	将不解释运行参数。轴移动到停止的运行的目标点 (LAST_TRG)。
在相反方向继续该运行	将不解释运行参数。轴移动到停止的运行的起始点。
启动新的“绝对增量逼近”	轴移动到指定的绝对目标。
作业“删除剩余行程”	剩余行程（目标值与实际值之差）将被删除。在启动“相对增量逼近”时再次解释运行参数，然后轴将移动到当前实际位置值。

### 3.4.7 绝对增量逼近模式

#### 说明

在“绝对增量逼近”模式下，将逼近绝对目标位置。

#### 要求

- 您已通过参数分配窗口分配模块参数，并将它们下载到了 CPU (PARA = TRUE)。
- 您已根据『SFB ANALOG (SFB 44) 的基本组态 (页码 47)』一节所述分配了 SFB 的基本参数。
- 未发生外部错误 ERR。您必须通过 ERR\_A (正跳沿) 来确认排队的外部错误。
- 启动已使能 (ST\_ENBLD = TRUE)。
- 轴是同步的 (SYNC = TRUE)。

#### 指定目标

指定目标时，请注意以下事项：

- 行程必须大于等于关断差程。
- 如果行程小于等于目标范围的一半，将不会启动新运行。此模式将立刻终止且不产生错误。
- 线性轴的目标范围必须处于工作范围之内，旋转轴必须位于“0”和“旋转轴终点”-1 之间。

#### 运行启动

- 线性轴始终在 START = TRUE 时启动。
- 必须指定旋转轴方向的意义：  
DIR\_P = TRUE：正方向运行  
DIR\_M = TRUE：负方向运行  
START = TRUE：轴沿可能的最短距离逼近目标。

CPU 计算方向意义时会考虑剩余行程、实际值和目标。

如果最短距离小于等于关断差程且大于等于目标范围的一半，则将反向启动运行。

如果两个方向上的行程差相等，则轴将向正方向上移动。

## 步骤

- 按“设置”列中所指定的内容组态以下 SFB 输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
DRV_EN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE	TRUE
START	BOOL	4.1	运行启动（正跳沿）	TRUE/FALSE	FALSE	START 或 DIR_P 或 DIR_M = TRUE
DIR_P	BOOL	4.2	正方向运行（正跳沿）	TRUE/FALSE	FALSE	
DIR_M	BOOL	4.3	负方向运行（正跳沿）	TRUE/FALSE	FALSE	
MODE_IN	INT	6	操作模式，5 = 绝对增量逼近	0、1、3、4、5	1	5
TARGET	DINT	8	以脉冲表示的目标	线性轴： -5 x 10 <sup>8</sup> 到 +5 x 10 <sup>8</sup> 旋转轴： 0 到旋转轴终点 - 1	1000	xxxx

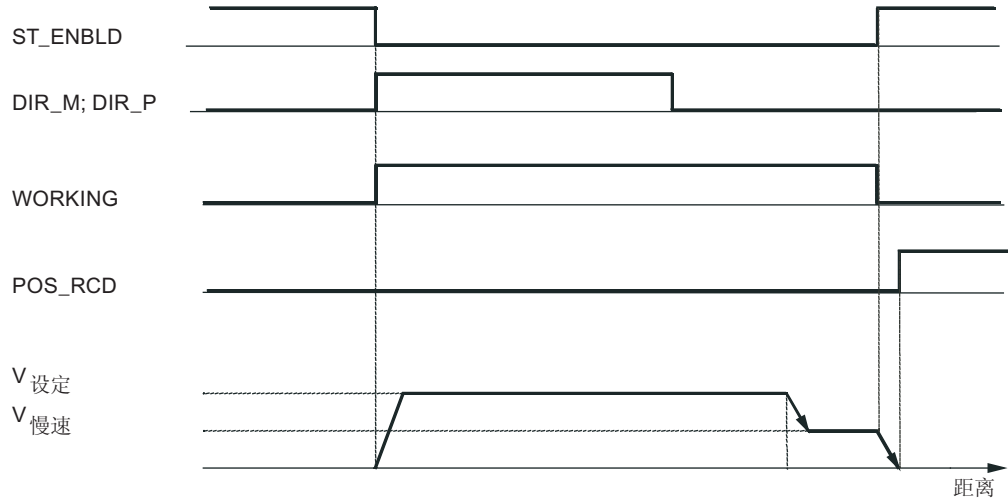
- 调用 SFB。

SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
WORKING	BOOL	16.0	正在运行	TRUE/FALSE	FALSE
POS_RCD	BOOL	16.1	已到位	TRUE/FALSE	FALSE
ACT_POS	DINT	18	当前实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	22	激活/设置操作模式	0、1、3、4、5	0

- 启动运行后将立即置位 WORKING = TRUE。在关断点处将 WORKING 复位为 FALSE。达到指定目标时，会将 POS\_RCD 置位为 TRUE。
- 启动下一个运行之前，必须复位方向位（DIR\_P 或 DIR\_M）。

- 如果在解释 SFB 调用时发生错误，则 WORKING = FALSE 且 ERROR 将置位为 TRUE。  
然后将使用 STATUS 参数指出准确的错误原因（请参阅『错误列表（页码 83）』一节）。



### 中断运行，并且未达到目标范围

用 STOP = TRUE 中断运行时，如果尚未达到关断范围（剩余行程大于关断差程），则依据后续操作模式/作业的不同，有以下几种选择。

选项	响应
启动新的“绝对增量逼近”	轴移动到指定的绝对目标。
以“相对增量逼近”模式在同一方向继续运行	将不解释运行参数。轴执行运行，直到被中断运行的目标 (LAST_TRG)。
以“相对增量逼近”模式在相反方向继续运行	将不解释运行参数。轴移动到被中断运行的起始点。
作业“删除剩余行程”	将删除剩余行程（目标值与实际值之差）。在启动“相对增量逼近”时再次解释运行参数，然后轴将移动到当前实际位置值。

### 3.4.8 指定参考点

#### 说明

也可使用“设置参考点”请求来同步轴，而不执行参考点逼近。

作业执行完毕后，实际位置的坐标值为 JOB\_VAL 的参数值。

- 线性轴：参考点坐标必须位于工作范围（包含软件限位开关）内。
- 旋转轴：参考点坐标必须位于 0 到“旋转轴终点” - 1 的范围内。

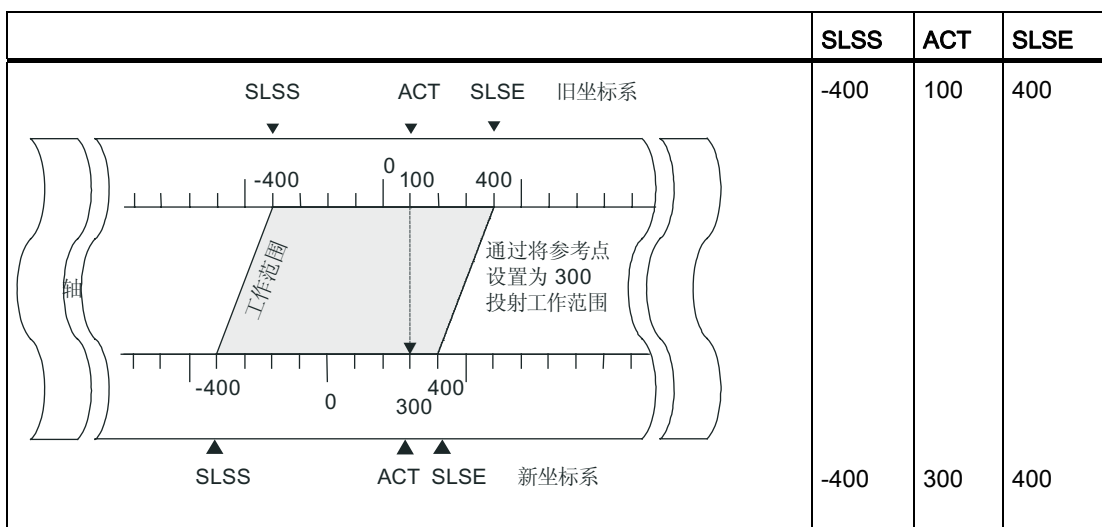
这不会更改在参数分配窗口中所输入的参考点坐标。

#### 设置参考点的实例

以下是如何设置参考点的实例。

- 实际位置值是 100。软件限位开关 (SLSS,SLSE) 位于位置 -400 和 400（工作范围）。
- 使用数值 JOB\_VAL = 300 来执行“设置参考点”请求后。

这样，实际值为坐标 300。软件限位开关和工作范围的坐标与作业之前的坐标相同。但是，此时它们实际上向左移动了 200。





## 要求

- 您已通过参数分配窗口分配模块参数，并将它们下载到了 CPU (PARA = TRUE)。
- 您已根据『SFB ANALOG (SFB 44) 的基本组态 (页码 47)』一节所述分配了 SFB 的基本参数。
- 最后一个作业必须完成 (JOB\_DONE = TRUE)。
- 最后一个定位操作必须完成 (WORKING = FALSE)。

## 步骤

- 按“设置”列中所指定的内容分配以下**输入参数**（通过背景数据块访问）：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
JOB_REQ	BOOL	76.0	作业触发（正跳沿）	TRUE/FALSE	FALSE	TRUE
JOB_ID	INT	78	作业，1 = “设置参考点”	1, 2	0	1
JOB_VAL	DINT	82	参考点坐标的作业参数	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	xxxx

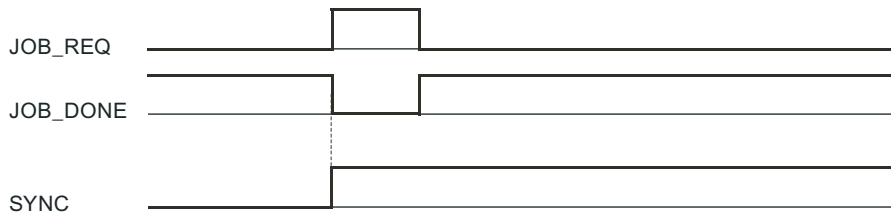
- 调用 SFB。

SFB 的**输出参数**（可通过背景数据块访问的 JOB\_DONE、JOB\_ERR、JOB\_STAT）提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
SYNC	BOOL	16.3	轴同步	TRUE/FALSE	FALSE
JOB_DONE	BOOL	76.1	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	BOOL	76.2	错误作业	TRUE/FALSE	FALSE
JOB_STAT	WORD	80	作业错误编号（请参阅『错误列表（页码 83）』一节）。	0 到 FFFF （十六进制）	0

3.4 使用模拟输出定位的功能

- 调用 SFB 之后将立即处理作业。在一个 SFB 循环时间内 JOB\_DONE 被设置为 FALSE。
- 必须复位作业请求 (JOB\_REQ)。
- 如果处理作业时未出现错误，则 SYNC = TRUE。
- 如果出现错误，则 JOB\_ERR = TRUE。随后将在 JOB\_STAT 中指出确切的出错原因。
- 可以通过 JOB\_DONE = TRUE 来启动新作业。



请求结果

- 实际位置值将被设置成参考点坐标值，并将置位状态信号 SYNC。
- 工作范围将沿着轴物理移动。
- 位于工作范围内的所有点将保持其各自的原始坐标，但具有新的物理位置。

同时调用作业和定位操作

同时启动定位操作和作业时，将先执行作业。如果作业已结束且出现错误，则将不执行定位。如果在执行某个运行时启动了一个作业，则该作业将结束，且出现错误。

### 3.4.9 删除剩余行程

#### 说明

在目标运行（绝对或相对增量逼近）后，可删除作业中未完成的剩余行程 (DIST\_TO\_GO)。

#### 要求

- 您已通过参数分配窗口分配模块参数，并将它们下载到了 CPU (PARA = TRUE)。
- 您已根据『SFB ANALOG (SFB 44) 的基本组态（页码 47）』一节所述分配了 SFB 的基本参数。
- 最后一个作业必须完成 (JOB\_DONE = TRUE)。
- 最后一个定位操作必须完成 (WORKING = FALSE)。

#### 步骤

- 按“设置”列中所指定的内容分配以下**输入参数**（通过背景数据块访问）：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
JOB_REQ	BOOL	76.0	作业触发（正跳沿）	TRUE/FALSE	FALSE	TRUE
JOB_ID	INT	78	作业，2 = 删除剩余行程	1, 2	0	2
JOB_VAL	DINT	82	无	-	0	任意

- 调用 SFB。

SFB（可通过背景数据块访问）的**输出参数**提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_DONE	BOOL	76.1	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	BOOL	76.2	错误作业	TRUE/FALSE	FALSE
JOB_STAT	WORD	80	作业错误 ID	0 到 FFFF（十六进制）	0

- 调用 SFB 之后将立即处理作业。在一个 SFB 周期时间内 JOB\_DONE 被设置为 FALSE。
- 必须复位作业请求 (JOB\_REQ)。
- 如果出现错误，则 JOB\_ERR = TRUE。然后将在 JOB\_STAT 中指出确切的错误原因。
- 可以使用 JOB\_DONE = TRUE 来启动新作业。

#### 同时调用作业和定位操作

同时启动定位操作和作业时，将先执行作业。如果作业已结束且出现错误，则将不执行定位。如果在执行某个运行时启动了一个作业，则该作业将结束，且出现错误。

### 3.4.10 长度测量

#### 说明

可通过“长度测量”来确定工件长度。在数字输入“长度测量”处通过边沿触发长度测量的启动和停止。

在 SFB 中给出了长度测量的启动和结束坐标以及测量的长度。

可通过参数分配窗口（参数“长度测量”）来启动和关闭长度测量，也可确定边沿的类型：

- 关
- 在正跳沿处启动/结束
- 在负跳沿处启动/结束
- 正跳沿启动，负跳沿结束
- 负跳沿启动，正跳沿结束

#### 要求

- 您已通过参数分配窗口分配模块参数，并将它们下载到了 CPU (PARA = TRUE)。
- 您已根据『SFB ANALOG (SFB 44) 的基本组态（页码 47）』一节所述分配了 SFB 的基本参数。
- 已将无反跳开关连接到了数字输入“长度测量”（连接器 X2，针脚 5）。
- 同步 (SYNC = TRUE) 和非同步 (SYNC = FALSE) 轴均可实现“长度测量”。

## 步骤

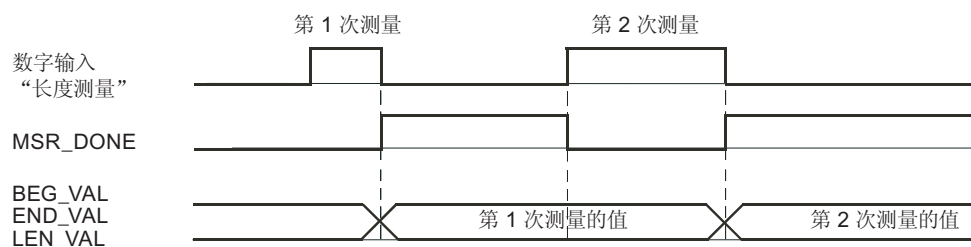
- 数字输入的信号边沿启动长度测量。
- 启动长度测量时将复位 MSR\_DONE。
- 长度测量结束时 MSR\_DONE 状态将被设置为 TRUE。
- 然后 SFB 将输出下列值：
  - 启动长度测量：BEG\_VAL
  - 结束长度测量：END\_VAL
  - 测量的长度：LEN\_VAL

从一个长度测量结束到下一个长度测量结束，这些值在块中可用。

- SFB（可通过背景数据块访问 BEG\_VAL、END\_VAL、LEN\_VAL）的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
MSR_DONE	BOOL	16.2	结束长度测量	TRUE/FALSE	FALSE
BEG_VAL	DINT	64	实际位置值，启动长度测量	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
END_VAL	DINT	68	实际位置值，结束长度测量	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
LEN_VAL	DINT	72	测量长度	0 到 10 <sup>9</sup> 个脉冲	0

下图显示了某个长度测量的信号曲线，该长度测量的类型为：在正/负跳沿启动/结束长度测量。



## 说明

在长度测量期间参考时，将以如下方式将实际值更改考虑在内：


实例：在距离为 100 个脉冲的两个点之间执行长度测量。在长度测量期间参考时，坐标将移动 + 20。从而会导致测量的长度为 120。

### 3.5 调整参数

#### 3.5.1 重要安全规则

##### 重要注意事项

请注意以下警告内容：

 <b>警告</b>
<p>可能会造成人身伤害或财产损失。</p> <p>为避免伤害和财产损坏，请注意以下内容：</p> <ul style="list-style-type: none"> <li>在控制系统区域内安装<b>紧急停车开关</b>。这是当控制系统发生故障时安全关闭系统的唯一可行方法。</li> <li>安装<b>硬件限位开关</b>，它可直接影响所有驱动器的驱动器转换器。</li> <li>确保<b>没有人进入系统区域</b>，该区域中存在运动部件。</li> <li>通过程序和 STEP7 接口进行<b>并行控制和监视</b>可能会导致冲突，从而会造成不确定的影响。</li> </ul>

#### 3.5.2 确定模块参数及其作用

##### 编码器每转增量

在所连接的增量编码器的铭牌或规格说明书上可以找到其**编码器每转增量**参数。该技术以四倍模式判断编码器信号。四个脉冲表示一个编码器增量。所有距离规范都以脉冲为单位。

##### 最高转速

必须计算**最高转速**参数。先决条件是要知道驱动器的额定转速（模拟输出处使用 +/-10 V）。可以在您的驱动器技术规范中找到该信息。如果通过齿轮将编码器安装到电机上，还必须考虑传动比，因为最高转速是参考编码器获得的。

$$\text{最高转速 [ 个脉冲/秒 ]} = \text{驱动器的额定转速 [ 转/秒 ]} \times \text{传动比} \times \text{编码器每转增量 [ 增量/转 ]} \times 4$$

实例：

驱动器的额定转速：	3000 [rpm]
传动比：	1:1（无齿轮）
编码器每转增量：	500 [ 增量/转 ]

$$3000 \text{ [ 转/分 ]} = 50 \text{ [ 转/秒 ]}$$

$$500 \text{ [ 增量/转 ]} = 2000 \text{ [ 脉冲/转 ]}$$

$$\text{最高转速} = 50 \frac{\text{转}}{\text{秒}} \times 1 \times 2000 \frac{\text{个脉冲}}{\text{转}} = 100000 \frac{\text{个脉冲}}{\text{秒}}$$

为获得良好的可重复的定位结果，必须正确地确定并指定最高转速。

## 爬行速度/参考转速

**爬行速度/参考转速**参数也依编码器而定。此时，指定的最高转速将转换为模拟电压。

例如，如果最高转速为 10000 个脉冲/秒，爬行速度/参考转速为 1000 个脉冲/秒，则模拟输出处输出的电压为 1 V。

爬行速度/参考转速必须高到足以维持驱动器运转。

## 监视时间

必须在**监视时间**参数中选择一段足够长的时间，以确保驱动器能够在指定时间内代替轴的启动保持转矩。

实例：

驱动器在模拟电压为 0.5 V 时开始运转。

最高转速：	10,000 [ 个脉冲/秒 ] = 10 V
加速度：	1,000 [ 个脉冲/秒 <sup>2</sup> ]

=> 转速 = 500 个脉冲/秒 = 0.5 V

=> T = 转速/加速度 = 500 个脉冲/秒/1,000 个脉冲/秒<sup>2</sup> = 0.5 秒

也就是说，驱动器将在 0.5 秒之后才开始运转。在这种情况下，必须将监视时间设置为大于 0.5 秒。

也可使用此监视时间来监视目标逼近。即，驱动器在达到关断位置后，必须在此时间内达到目标范围。

## 计数方向

使用**计数方向**参数可将路径监视的方向调整为线性轴的运动方向。还应考虑所有传动元件（例如，联接器和齿轮）的旋转方向。

- “标准”表示计数脉冲数的增加将对应着实际位置值的增加。
- “反转”表示计数脉冲数的增加将对应着实际位置值的减小。

### 3.5.3 SFB 参数的作用

#### ACCEL 和 DECEL

在 **ACCEL**（加速度）和 **DECEL**（减速度）参数中，声明驱动器的加速度/减速度值。

实例：

如果所需的移动速度为 10,000 个脉冲/秒，加速度为 1,000 个脉冲/秒<sup>2</sup>，则需要 10 秒才能达到速度设定值 10,000 个脉冲/秒。

#### CHGDIFF\_P 和 CHGDIFF\_M

**CHGDIFF\_P**（正方向转换差程）和 **CHGDIFF\_M**（负方向转换差程）参数定义驱动器开始以蠕行速度运转时的转换位置。

如果差值设置得过大，则不会随时间对定位进行优化，因为这样会不必要地延长了蠕行速度运行的时间。

#### CUTOFFDIFF\_P 和 CUTOFFDIFF\_M

参数 **CUTOFFDIFF\_P**（正方向的关断差程）和 **CUTOFFDIFF\_M**（负方向的关断差程）指定在目标逼近时关闭驱动器之前要经过的脉冲数。

请注意，该距离将随着驱动器上负载的不同而有所不同。

如果为转换/关断差程选择的差程不够高，则驱动器关闭时的转速将高于组态的蠕行速度。结果会导致定位不准确。

各个方向的转换/关断差程的差应至少与驱动器为达到蠕行速度而实际需要的距离成比例。此时，要求的移动速度构成基准速度。还要考虑驱动器的负载情况。



### 3.5.4 检查监视时间

#### 要求

- 系统接线正确。
- 已组态了定位子模块，已分配了参数且已载入了项目。
- 例如，已装载了随供货一起提供的实例程序“Analog 1, Getting Started”。
- CPU 处于 RUN 状态。

#### 核对清单

步骤	要执行的操作	✓
1	<b>校验接线</b>	
	<ul style="list-style-type: none"> <li>• 检查输出接线是否正确（转换器的模拟输出和使能输出“CONV_EN”）</li> <li>• 检查编码器输入接线是否正确</li> </ul>	<input type="checkbox"/> <input type="checkbox"/>
2	<b>检查轴运动</b>	
	<ul style="list-style-type: none"> <li>• 在“步进”模式下，爬行速度正方向和反向移动（请参见模块参数）。 方向 DIR 的实际意义必须与指定的方向相符。 如果事实并非如此，请更改模块参数“计数方向”。</li> </ul>	<input type="checkbox"/>
3	<b>同步轴</b>	
	<ul style="list-style-type: none"> <li>• 选择作业“设置参考”(JOB_ID = 1)。 通过 JOB_VAL 输入所需要的实际轴位置的坐标（例如 0 脉冲）。 通过将 JOB_REQ 设置为 TRUE 来执行同步。 指定的坐标将显示为实际位置值，并且置位同步位 SYNC。 判断 (JOB_STAT) 报告的错误 (JOB_ERR = TRUE)。 如果需要，请修正指定的坐标并重复用于设置参考的作业。</li> </ul>	<input type="checkbox"/>

3.5 调整参数

步骤	要执行的操作	✓
4	<p><b>检查转换/关断差程</b></p> <ul style="list-style-type: none"> <li>• 在“绝对/相对增量逼近”模式下，逼近指定目标 (TARGET)，该目标与实际位置的距离要比组态的转换差程更远。 在此处，选择一个大于爬行速度的适当转速 (SPEED)。 爬行速度 &lt;= SPEED &lt;= 最高转速。</li> <li>• 请注意各个定位阶段（加速、恒速运行、减速、目标逼近）。 增加转换差程，使驱动器以明显的爬行速度运行到关断点。 如果未达到组态的目标范围，请减小转换差程并重复该运行直到达到目标范围为止。 如果超出了组态的目标范围，请增加转换差程并重复该运行直到不再超出目标范围为止。</li> <li>• 现在优化转换差程。 在不更改关断差程的情况下减小转换差程，并重复该运行。 可以将转换差程减小为某个值，驱动器在达到该值时几乎觉察不到其以爬行速度移动，即驱动器在关断位置处实际上已达到爬行速度并且在该位置关断。 只要驱动器在爬行速度时关闭，定位精度就会保持不变。 进一步减小关断差程没有任何意义。</li> </ul>	<p style="text-align: right;">✓</p> <p style="text-align: right;"><input type="checkbox"/></p> <p style="text-align: right;"><input type="checkbox"/></p> <p style="text-align: right;"><input type="checkbox"/></p> <p style="text-align: right;"><input type="checkbox"/></p> <p style="text-align: right;"><input type="checkbox"/></p>
5	<p><b>检查最高转速（如果定位效果不好）</b></p> <ul style="list-style-type: none"> <li>• 在“步进”模式下，以最高转速正方向和反向移动（请参见模块参数）。 测量编码器信号 A 或 B 的频率（例如使用计数器子模块），单位为 [1/s]。将测量频率字节乘 4，然后将乘积作为最高转速输入到模块参数。</li> </ul>	<p style="text-align: right;"><input type="checkbox"/></p>

## 3.6 错误处理和中断

### 3.6.1 系统功能块 (SFB) 中的错误消息

#### 错误类型

SFB 指出了下表中列出的错误。

除系统错误外的所有错误都将通过一个错误编号更详细地进行指定，错误编号可作为 SFB 中的输出参数。

错误类型	错误显示于 SFB 参数	错误编号显示于 SFB 参数
操作模式错误	ERROR = TRUE	STATUS
作业错误	JOB_ERR = TRUE	JOB_STAT
外部错误	ERR > 0	ERR
系统错误	BIE = FALSE	-

#### 操作模式错误 (ERROR = TRUE)

此类错误出现在

- SFB 上发生常规参数分配错误时（例如使用了错误的 SFB）。
- 运行启动/恢复时。这些错误在解释操作模式参数期间出现。

检测到错误时，输出参数 ERROR 将被置位为 TRUE。

可在『错误列表（页码 83）』一节中找到可能的错误编号。

参数 STATUS 指示出错原因。

#### 作业错误 (JOB\_ERR = TRUE)

只有在解释/执行作业期间会出现作业错误。

检测到错误时，输出参数 JOB\_ERR 将被置位为 TRUE。

可在『错误列表（页码 83）』一节中找到可能的错误编号。

出错原因会在 JOB\_STAT 参数中指出。

### 外部错误 (ERR)

系统将监视运行、行程范围和所连接的 I/O。先决条件是需要将“驱动器”、“轴”和“编码器”参数分配窗口中打开监视。

监视功能响应时，将发送外部错误信号。

外部错误可能随时出现，与启动的功能无关。

您必须通过 ERR\_A 处的正跳沿确认队列中等待的外部错误。

通过在 SFB 参数 ERR (WORD) 中置位一个位来指示外部错误。

监视	ERR	ERR-WORD 中的位
缺少脉冲（零标记）	0004（十六进制）	2
行程范围	0800（十六进制）	11
工作范围	1000（十六进制）	12
实际值	2000（十六进制）	13
目标逼近	4000（十六进制）	14
目标范围	8000（十六进制）	15

检测到外部错误（“进入”和“离开”）还会触发诊断中断（请参阅『组态和判断诊断中断（页码 78）』一节）。

### 系统错误

用 BIE = FALSE 来指示系统错误。

下列情况下会触发系统错误：

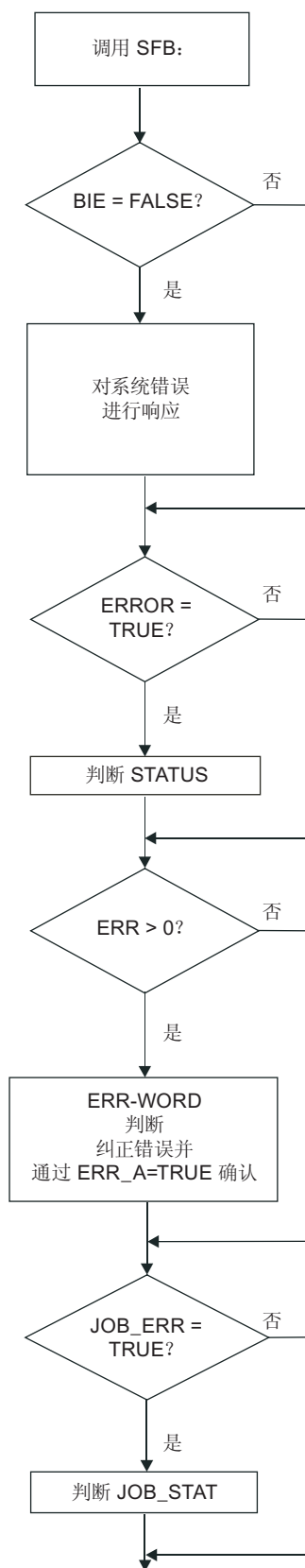
- 背景数据块的读/写访问错误
- 多重调用 SFB

### 3.6.2 在用户程序中判断出错

#### 步骤

1. 调用错误处理例程“错误判断”（参见下图）。
2. 按顺序查询特定的错误类型。
3. 如果需要，可跳转到特别适合您的应用的错误响应方法。

错误判断:



### 3.6.3 组态和判断诊断中断

#### 基本信息

出现以下错误时，可以触发诊断中断：

- 参数分配错误（模块数据）
- 外部错误（监视）

在出现进入错误和离开错误事件时，将会显示诊断中断。

在用户程序中，可借助诊断中断立即对错误作出响应。

#### 步骤

1. 在参数分配窗口的“基本参数”对话框中启用诊断中断。
2. 在“驱动器”、“轴”和“编码器”参数分配窗口中，分别打开出错时会触发诊断中断的监视功能。
3. 在参数分配窗口“诊断”中，分别为每个监视对象启用诊断中断。
4. 可将诊断中断 OB (OB 82) 合并到用户程序中。

#### 通过诊断中断对错误作出响应

- 定位被取消。
- CPU 操作系统将调用用户程序中的 OB 82。

---

#### 说明

如果未装载相应的 OB，则在触发中断后，CPU 会切换为 STOP。

---

- CPU 将打开 SF LED 指示灯。
- 在 CPU 的诊断缓冲区中将错误报告为“进入”。清除所有未决错误前，不会将错误指示为“离开”。

## 如何在用户程序中判断诊断中断

触发诊断中断后，可判断 OB 82 以检查哪个诊断中断还未处理。

- 如果在 OB 82 的字节 6 + 7 (OB 82\_MDL\_ADDR) 中输入了“定位”子模块的模块地址，则诊断中断是由 CPU 的定位功能触发的。
- 只要队列中有错误，就会置位 OB 82 中字节 8 的位 0（故障模块）。
- 所有错误都已报告“离开”后，将复位 OB 82 中字节 8 的位 0。
- 通过判断数据记录 1（字节 8 和 9），可以确定确切的出错原因。为此，必须调用 SFC 59（读取数据记录）。
- 用 ERR\_A 确认错误。

数据记录 1, 字节 8	说明	JOB_STAT	ERR
位 0	没有使用	-	-
位 1	没有使用	-	-
位 2	缺少脉冲 *	-	X
位 3	没有使用	-	-
位 4	没有使用	-	-
位 5	没有使用	-	-
位 6	没有使用	-	-
位 7	没有使用	-	-
*: 随后出现的错误将触发一个进入中断，然后自动触发离开中断。			

数据记录 1, 字节 9	说明	JOB_STAT	ERR
位 0	组态错误	X	-
位 1	没有使用	-	-
位 2	没有使用	-	-
位 3	行程范围监视	X	X
位 4	工作范围监视	X	X
位 5	实际值监视 *	X	X
位 6	目标逼近监视 *	X	X
位 7	目标范围监视 *	X	X
*: 随后出现的错误将触发进入中断，然后自动触发离开中断。			

## 3.7 安装实例

### 使用实例

可在文档附带的 CD-ROM 中找到实例（程序和说明）。也可从 Internet 下载。项目由具有不同复杂度和针对性的多个带注释的 S7 程序组成。

CD 上的 Readme.wri 说明了安装实例的方法。安装后，实例即存储在目录 `...\STEP7\EXAMPLES\ZDI26_03_TF____31xC_Pos` 中。



## 3.8 规范

### 3.8.1 增量编码器

#### 可连接的增量编码器

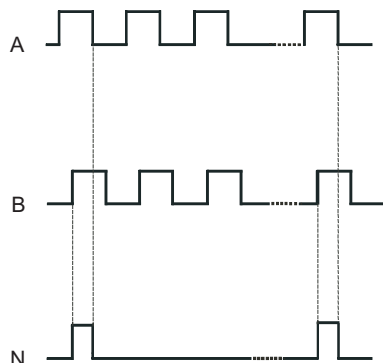
支持非对称 24 V 增量编码器，该编码器具有两个电相差为 90° 的脉冲轨迹（有/无零标记）。

编码器输入	最小脉冲宽度/最小脉冲间宽度	最大输入频率	最大电缆长度 (最大输入频率)
编码器信号 A, B	8 μs	60 kHz	50 m
编码器信号 N (零标记信号)	8 μs	60 kHz / 30 kHz <sup>1</sup>	50 m

1: 如果使用的编码器的零标记信号通过“AND”运算与编码器信号 A 和 B 合并，脉冲宽度将减半至周期时间的 25%。为了维持最小脉冲宽度，必须将最大计数频率降低到 30 kHz。

#### 信号判断

下图显示具有非对称输出信号的编码器的信号曲线：



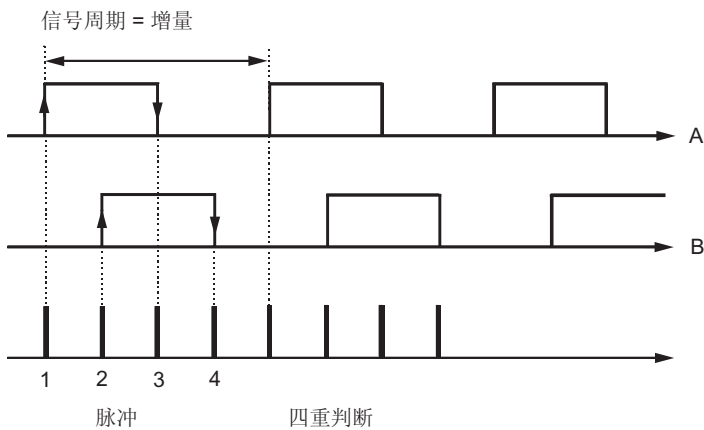
CPU 内部生成零标记信号与 A 和 B 轨迹信号的逻辑 AND 链接。

为了参照，CPU 在零标记处使用正跳沿。

如果信号 A 转换超前信号 B，则 CPU 正方向计数。

### 增量

一个增量标识两个编码器轨迹信号 A 和 B 的信号周期。该值在编码器的铭牌和/或技术规范中指定。

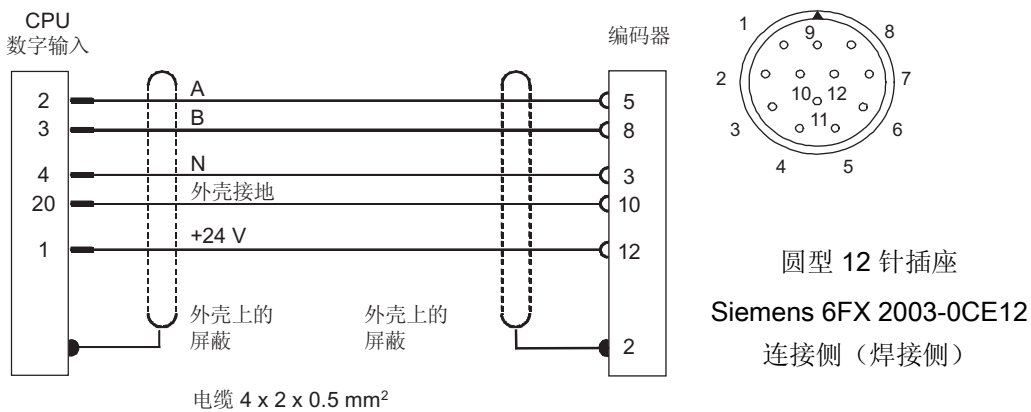


### 脉冲

CPU 使用每个增量（四重判断）评估轨迹信号 A 和 B 的全部 4 个跳沿（参见上图）。也就是说，一个编码器增量对应四个脉冲。

### 增量编码器 Siemens 6FX 2001-4 (Up = 24 V; HTL) 的接线图

下面的视图显示了增量编码器 Siemens 6FX 2001-4xxxx (Up = 24 V; HTL) 的接线图：



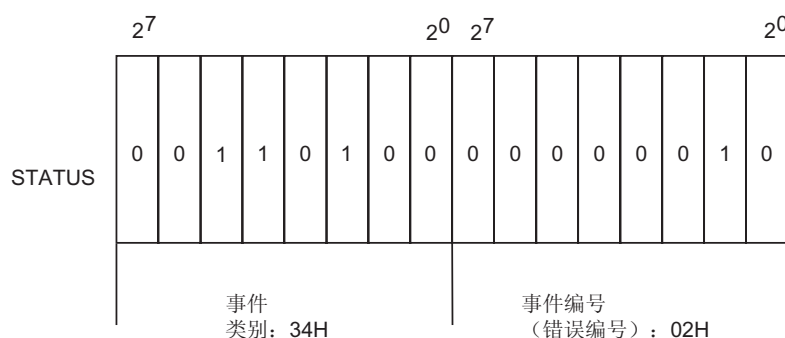
## 3.8.2 错误列表

### 基本信息

如果出现错误，会在 SFB 参数 STATUS 或 JOB\_STAT 中输出错误 ID。错误 ID 由事件类别和编号组成。

### 错误列表实例

下图显示了事件“指定的目标错误”（事件类别：34H，事件编号：02H）的 STATUS 参数内容。



### SFB 参数“状态”的错误 ID

事件类别 32 (20H): “SFB 错误”		
事件编号	事件文本	纠正方法
(20)02H	SFB 错误	使用 SFB 44
(20)04H	通道号 (CHANNEL) 错误	设置通道号 “0”

事件类别 48 (30H): “常规运行启动错误”		
事件编号	事件文本	纠正方法
(30)01H	运行作业被拒绝，因为同一 SFB 调用中存在错误作业	更正相应 JOB 参数
(30)02H	驱动器仍在运行时不允许修改 MODE_IN	等待，直至完成当前的定位操作
(30)03H	未知操作模式 (MODE_IN)	允许的模式有：1（点动模式）、3（参考点逼近）、4（相对增量逼近）和 5（绝对增量逼近）
(30)04H	一次只能设置一个启动请求	允许的启动请求为 DIR_P、DIR_M 或 START
(30)05H	仅在“绝对增量逼近”模式下允许 START	使用 DIR_P 或 DIR_M 启动运行
(30)06H	线性轴和“绝对增量逼近”模式不允许 DIR_P 或 DIR_M	使用 START 启动运行
(30)07H	轴未同步	仅当轴同步时才能使用“绝对增量逼近”。
(30)08H	移出工作范围	只允许向进入工作范围的方向运行。

事件类别 49 (31H): “运行启动错误 (启用启动)”		
事件编号	事件文本	纠正方法
(31)01H	未启用启动, 因为未组态轴。	通过 HW Config 组态 “定位” 子模块
(31)02H	未启用启动, 因为未设置驱动器使能。	在 SFB (DRV_EN = TRUE) 中置位 “启动使能”
(31)03H	启动未使能, 因为 STOP 置位。	在 SFB (STOP = FALSE) 中将 STOP 清零
(31)04H	启动未使能, 因轴当前在执行定位运行 (WORKING = TRUE)。	等待, 直至完成当前的定位操作
(31)05H	启动未使能, 因为至少有一个未决错误未得到确认。	首先, 排除并确认所有外部错误, 然后重新启动运行

事件类别 50 (32H): “运行启动错误 (速度/加速度)”		
事件编号	事件文本	纠正方法
(32)02H	速度规范 SPEED 错误	指定的速度超出了爬行速度允许的最大范围 1000000 个脉冲/秒, 但未超过指定的最大速度。
(32)03H	加速度规范 ACCEL 错误	指定的加速度值超出了允许的范围, 即 1 到 100000 个脉冲/秒 <sup>2</sup> 。
(32)04H	减速度规范 DECEL 错误	指定的减速度值超出了允许的范围, 即 1 到 100000 个脉冲/秒 <sup>2</sup> 。
(32)06H	速度规范 SPEED 错误	指定的速度值必须大于/等于指定的参考频率。

事件类别 51 (33H): “运行启动错误 (转换/关断差程)”		
事件编号	事件文本	纠正方法
(33)01H	转换/关断差程不允许大于 10 <sup>8</sup>	指定最大值为 10 <sup>8</sup> 的转换/关断差程。
(33)04H	关断差程过短	关断差程必须至少为目标范围长度的一半。
(33)05H	转换差程过短	转换差程必须至少为目标范围长度的一半。

事件类别 52 (34H): “运行启动错误 (缺省目标/距离)”		
事件编号	事件文本	纠正方法
(34)01H	缺省目标超出工作范围	对于线性轴和绝对增量逼近, 缺省目标必须位于软件限位开关范围内 (含此范围)。
(34)02H	目标规范错误	对于旋转轴, 指定的目标必须大于 0 并小于旋转轴终点。
(34)03H	距离规范错误	对于相对增量逼近, 行程必须为正值。
(34)04H	距离规范错误	作为结果的绝对目标坐标必须大于 $-5 \times 10^8$ 。
(34)05H	距离规范错误	作为结果的绝对目标坐标必须小于 $-5 \times 10^8$ 。
(34)06H	距离规范错误	作为结果的绝对目标坐标必须位于工作范围 (+/- 目标范围的一半) 内。

事件类别 53 (35H): “运行启动错误 (行程)”		
事件编号	事件	纠正方法
(35)01H	行程过长	目标坐标 + 实际剩余行程必须大于等于 $-5 \times 10^8$
(35)02H	行程过长	目标坐标 + 实际剩余行程必须小于等于 $5 \times 10^8$
(35)03H	行程过短	正方向的行程必须大于正方向的指定关断差程
(35)04H	行程过短	负方向的行程必须大于负方向的指定关断差程
(35)05H	行程过短或已在正方向上超出限位开关	正方向的上一可逼近目标 (工作范围/行程范围限制) 距离实际位置太近
(35)06H	行程过短或已在负方向上超出限位开关	负方向的上一可逼近目标 (工作范围/行程范围限制) 距离实际位置太近

## SFB 参数 JOB\_STAT 上的错误编号

事件类别 64 (40H): “常规作业执行错误”		
事件编号	事件	纠正方法
(40)01H	轴未组态	使用 HW Config 组态“定位”子模块。
(40)02H	作业无法执行, 因为定位仍在运行	仅当没有处于活动状态的定位运行时, 才能执行作业。等待, 直到 WORKING = FALSE, 然后重复该作业。
(40)04H	未知作业	检查作业 ID 并重复该作业。

事件类别 65 (41H): “执行设置参考点请求时出错”		
事件编号	事件	纠正方法
(41)01H	参考点坐标超出工作范围	对于线性轴, 参考点坐标不得超出工作范围限制。
(41)02H	参考点坐标不正确	对于线性轴, 指定的参考点坐标 + 实际剩余行程必须仍大于等于 $-5 \times 10^8$ 。
(41)03H	参考点坐标不正确	对于线性轴, 指定的参考点坐标 + 实际剩余行程必须仍小于等于 $5 \times 10^8$ 。
(41)04H	参考点坐标不正确	对于线性轴, 指定的参考点坐标 + 距运行起始点的实际距离必须仍大于等于 $-5 \times 10^8$ 。
(41)05H	参考点坐标不正确	对于线性轴, 指定的参考点坐标 + 距运行起始点的实际距离必须仍小于等于 $5 \times 10^8$ 。
(41)06H	参考点坐标超出旋转轴范围	对于旋转轴, 参考点坐标不得小于 0 和大于等于旋转轴终点的值。

## 外部错误 (ERR)

通过在 SFB 参数 ERR (WORD) 中置位一个位来指示外部错误。

监视	ERR	ERR-WORD 中的位
缺少脉冲 (零标记)	0004 (十六进制)	2
行程范围	0800 (十六进制)	11
工作范围	1000 (十六进制)	12
实际值	2000 (十六进制)	13
目标逼近	4000 (十六进制)	14
目标范围	8000 (十六进制)	15

### 3.8.3 参数分配窗口的模块参数 - 概述

#### 基本参数

参数	值范围	缺省
中断选择	<ul style="list-style-type: none"> <li>• 无</li> <li>• 诊断</li> </ul>	无

#### 驱动器参数

参数	值范围	缺省
目标范围	0 到 200,000,000 个脉冲 CPU 将向上舍入奇数值	50
时间监视	<ul style="list-style-type: none"> <li>• 0 到 100 000 ms</li> <li>• 0 = 无监视</li> </ul> 舍入操作由 CPU 在 4 ms 内完成	2000
最高转速	10 到 1,000,000 个脉冲/秒	1000
蠕行速度/参考转速	10 到组态的最大转速	100
切断延迟	0 到 100 000 ms 舍入操作在 4 ms 的处理周期内完成	1000
最大频率: 位置反馈	60、30、10、5、2、1 kHz	60kHz
最大频率: 伴随信号	60、30、10、5、2、1 kHz	10kHz
控制模式	<ul style="list-style-type: none"> <li>• 电压 +/-10 V 或电流 +/-20 mA</li> <li>• 电压 0 到 10 V 或电流 0 到 20 mA, 和方向信号</li> </ul>	电压 +/-10 V 或 电流 +/-20 mA
实际值监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	是
目标逼近监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
目标范围监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否

## 轴参数

参数	值范围	缺省
轴类型	<ul style="list-style-type: none"> <li>线性轴</li> <li>旋转轴</li> </ul>	线性轴
软件限位开关开始/结束	软件限位开关开始/软件限位开关结束 -5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	-100 000 000 +100 000 000
旋转轴终点	1 到 10 <sup>9</sup> 个脉冲	100 000
长度测量	<ul style="list-style-type: none"> <li>关</li> <li>在正跳沿 DI 处启动/结束</li> <li>在负跳沿 DI 处启动/结束</li> <li>通过正跳沿启动, 通过负跳沿结束</li> <li>通过负跳沿启动, 通过正跳沿结束</li> </ul>	关
参考点坐标	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
参考点开关的参考点位置	<ul style="list-style-type: none"> <li>正方向 (实际值增加)</li> <li>负方向 (实际值减少)</li> </ul>	正方向
行程范围监视	是 (设置成固定值)	是
工作范围监视	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	是

## 编码器参数

参数	值范围	缺省
编码器每转增量	1 到 2 <sup>23</sup> 个脉冲	1000
计数方向	<ul style="list-style-type: none"> <li>标准</li> <li>反转</li> </ul>	标准
缺少脉冲 (零标记) 监视	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否

## 诊断参数

参数	值范围	缺省
缺少脉冲 (零标记)	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
行程范围	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
工作范围 (对于线性轴)	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
实际值	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
目标逼近	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
目标范围	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否



## 3.8.4 SFB ANALOG (SFB44) 的背景数据块的参数

## 概述

参数	声明	数据类型	地址（背景数据块）	说明	值范围	缺省
LADDR	IN	WORD	0	在“HW Config”中指定的子模块 I/O 地址。 如果 I/O 地址不相同，必须指定两者中的较低一个。	CPU 特定	310 (十六进制)
CHANNEL	IN	INT	2	通道号	0	0
DRV_EN	IN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE
START	IN	BOOL	4.1	启动运行 (正跳沿)	TRUE/FALSE	FALSE
DIR_P	IN	BOOL	4.2	正方向运行（正跳沿）	TRUE/FALSE	FALSE
DIR_M	IN	BOOL	4.3	负方向运行（正跳沿）	TRUE/FALSE	FALSE
STOP	IN	BOOL	4.4	停止运行	TRUE/FALSE	FALSE FALSE
ERR_A	IN	BOOL	4.5	组错误确认 ERR_A 用于确认外部错误 (正跳沿)	TRUE/FALSE	
MODE_IN	IN	INT	6	操作模式	0、1、3、4、5	1
TARGET	IN	DINT	8	相对增量逼近： 以脉冲表示的距离（仅允许正值）  绝对增量逼近： 以脉冲表示的目标	0 到 $10^9$  线性轴： $-5 \times 10^8$ 到 $+5 \times 10^8$ 旋转轴： 旋转轴： 0 到旋转轴终点 - 1	1000
SPEED	IN	DINT	12	轴被加速到“V <sub>设定值</sub> ”	10 到 1,000,000 个脉冲/秒 可达到组态的最大 转速	1000
WORKING	OUT	BOOL	16.0	正在运行	TRUE/FALSE	FALSE
POS_RCD	OUT	BOOL	16.1	位置已达到	TRUE/FALSE	FALSE
MSR_DONE	OUT	BOOL	16.2	结束长度测量	TRUE/FALSE	FALSE
SYNC	OUT	BOOL	16.3	轴同步	TRUE/FALSE	FALSE
ACT_POS	OUT	DINT	18	实际位置值	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0
MODE_OUT	OUT	INT	22	启用/设置操作模式	0、1、3、4、5	0

参数	声明	数据类型	地址（背景数据块）	说明	值范围	缺省
ERR	OUT	WORD	24	外部错误 位 2: 缺少脉冲监视 位 11: 行程范围监视（始终为 1） 位 12: 工作范围监视 位 13: 实际值监视 位 14: 目标逼近监视 位 15: 目标范围监视 其它位将保留	每个位 0 或 1	0
ST_ENBLD	OUT	BOOL	26.0	启动使能	TRUE/FALSE	TRUE
ERROR	OUT	BOOL	26.1	运行启动/恢复错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	28.0	错误 ID	0 到 FFFF （十六进制）	0
ACCEL	STAT	DINT	30	加速	1 到 100,000 个脉冲/秒 <sup>2</sup>	100
DECEL	STAT	DINT	34	减速度	1 到 100,000 个脉冲/秒 <sup>2</sup>	100
CHGDIFF_P	STAT	DINT	38	正方向转换差程	0 到 +10 <sup>8</sup> 脉冲	1000
CUTOFF-DIFF_P	STAT	DINT	42	正方向关断差程	0 到 +10 <sup>8</sup> 脉冲	100
CHGDIFF_M	STAT	DINT	46	负方向转换差程	0 到 +10 <sup>8</sup> 脉冲	1000
CUTOFF-DIFF_M	STAT	DINT	50	负方向关断差程	0 到 +10 <sup>8</sup> 脉冲	100
PARA	STAT	BOOL	54.0	轴已组态	TRUE/FALSE	FALSE
DIR	STAT	BOOL	54.1	当前/上一方向的意义 FALSE = 向前（正方向） TRUE = 反向（负方向）	TRUE/FALSE	FALSE
CUTOFF	STAT	BOOL	54.2	驱动器位于关断范围内 （从关断位置到下一运行开始）	TRUE/FALSE	FALSE
CHGOVER	STAT	BOOL	54.3	驱动器位于转换范围内 （从达到转换位置到下一运行开始）	TRUE/FALSE	FALSE
RAMP_DN	STAT	BOOL	54.4	驱动器减速 （从制动点到转换位置）	TRUE/FALSE	FALSE
RAMP_UP	STAT	BOOL	54.5	驱动器加速 （从开始到达到最终速度）	TRUE/FALSE	FALSE
DIST_TO_GO	STAT	DINT	56	实际剩余行程	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
LAST_TRG	STAT	DINT	60	上一/实际目标	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
BEG_VAL	STAT	DINT	64	实际位置值，启动长度测量	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0

参数	声明	数据类型	地址（背景数据块）	说明	值范围	缺省
END_VAL	STAT	DINT	68	实际位置值，结束长度测量	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
LEN_VAL	STAT	DINT	72	测量长度	0 到 10 <sup>9</sup> 个脉冲	0
JOB_REQ	STAT	BOOL	76.0	启动作业（正跳沿）	TRUE/FALSE	FALSE
JOB_DONE	STAT	BOOL	76.1	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	STAT	BOOL	76.2	错误作业	TRUE/FALSE	FALSE
JOB_ID	STAT	INT	78	作业 ID	1, 2	0
JOB_STAT	STAT	WORD	80	作业错误 ID	0 到 FFFF (十六进制)	0
JOB_VAL	STAT	DINT	82	参考点坐标的作业参数	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0





## 使用数字输出定位

### 4.1 接线

#### 4.1.1 重要安全规则

##### 遵守安全规则

 <b>危险</b>
为符合系统的安全概念，必须安装下面提及的开关设备并使其适用于您的系统： <ul style="list-style-type: none"><li>• <b>紧急停车开关</b>，用于关闭整个系统。</li><li>• <b>硬件限位开关</b>，直接影响所有驱动器电源装置。</li><li>• <b>电机保护</b></li></ul>

 <b>警告</b>
如果不切断电源，可能会危及人身安全和导致财产损失： 如果在带电状态下连接 CPU 的前插头，会有触电危险！ 请始终在断电状态下连接 CPU！ 缺少安全设备可能会危及人身安全和导致财产损失： 如果未安装“紧急停车开关”，连接在一起的机组可能会对设备造成损害。 安装“紧急停车开关”，使您能够切断连接的所有驱动器。

##### 说明

可在无辅助电路的情况下直接连接感性负载（例如，继电器和接触器）。  
如果可通过另外安装的辅助触点（例如继电器触点）切断 SIMATIC 输出电流电路，则必须在感性负载线圈中安装附加的浪涌电压抑制元件。

## 4.1 接线

### 4.1.2 接线规则

#### 连接电缆/屏蔽

- 必须屏蔽模拟输出和 24-V 编码器的电缆。
- 连接数字 I/O 的电缆如果长度超过 100m，则必须屏蔽。
- 在电缆屏蔽的两端，必须对屏蔽层进行端接。
- 软线，横截面积为 0.25 到 1.5 mm<sup>2</sup>
- 不需要电缆套。如果坚持使用电缆套，请使用不带绝缘环的电缆套（DIN 46228，A 形，短型）。

#### 屏蔽端接元件

使用此屏蔽端接元件能够轻松完成屏蔽电缆的接地连接，因为屏蔽端接元件直接接触固定轨。

#### 其它信息

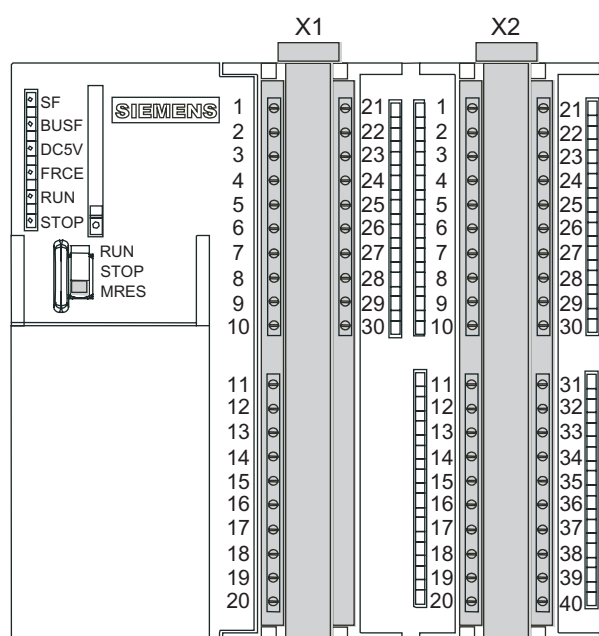
有关其它信息，请参见 *CPU 数据手册* 和 *CPU 安装说明*。

### 4.1.3 用于使用数字输出定位的端子

#### 概述

使用 CPU 314C-2 DP/PtP 的前连接器 X2 可连接下列组件：

- 24-V 编码器
- 长度测量开关
- 参考点开关
- 转换器（断路器）



#### 引脚分配说明

下面的引脚分配仅设计与位置模式相关的连接。

#### 说明

由于这些组件有时会使用相同的输入，所以在使用定位功能时不能使用计数器 0 和 1。

## 4.1 接线

表格 4-1 连接器 X2 的针脚分配

连接	名称/地址	功能
1	1 L+	输入的 24-V 电源
2	DI+0.0	编码器信号 A
3	DI+0.1	编码器信号 B
4	DI+0.2	编码器信号 N
5	DI+0.3	长度测量
6	DI+0.4	参考点开关
7	DI+0.5	-
8	DI+0.6	-
9	DI+0.7	-
10	-	n.c.
11	-	n.c.
12	DI+1.0	-
13	DI+1.1	-
14	DI+1.2	-
15	DI+1.3	-
16	DI+1.4	-
17	DI+1.5	-
18	DI+1.6	-
19	DI+1.7	-
20	1 M	外壳接地
21	2 L+	输出的 24-V 电源
22	DO+0.0	-
23	DO+0.1	-
24	DO+0.2	-
25	DO+0.3	-
26	DO+0.4	-
27	DO+0.5	-
28	DO+0.6	-
29	DO+0.7	-
30	2 M	外壳接地
31	3 L+	输出的 24-V 电源
32	DO+1.0	数字输出 Q0
33	DO+1.1	数字输出 Q1
34	DO+1.2	数字输出 Q2
35	DO+1.3	数字输出 Q3
36	DO+1.4	-
37	DO+1.5	-
38	DO+1.6	-
39	DO+1.7	-
40	3 M	外壳接地



## 4.1.4 连接组件

### 步骤

1. 关闭所有组件的电源。
2. 连接数字输入和输出的电源：
  - 24 V 在 X2，针脚 1、21 和 31
  - 接地到 X2，针脚 20、30 和 40
3. 将 24 V 编码器和开关连接到 24 V 电源。
4. 连接编码器信号和所需的开关（X2，针脚 2 到 6 以及针脚 20）。可将无反跳开关（24 V P 操作）或非接触传感器/BERO（2 或 3 线制接近开关）连接到数字输入“长度测量”和“参考点开关”。
5. 将动力装置连接到电源。
6. 连接动力装置电缆（X2，针脚 32 到 35 和针脚 40）。
7. 剥掉屏蔽电缆的绝缘材料并将电缆屏蔽连接到屏蔽连接元件。请使用屏蔽端子元件进行连接。

---

### 说明

CPU 不会检测数字输入是否有故障。可通过激活实际值监视来检测编码器故障；参见『驱动器参数（页码 102）』一节。

造成此类故障的原因有以下几种：

- 数字输入故障
  - 断线
  - 编码器有故障
  - 动力装置有故障
-

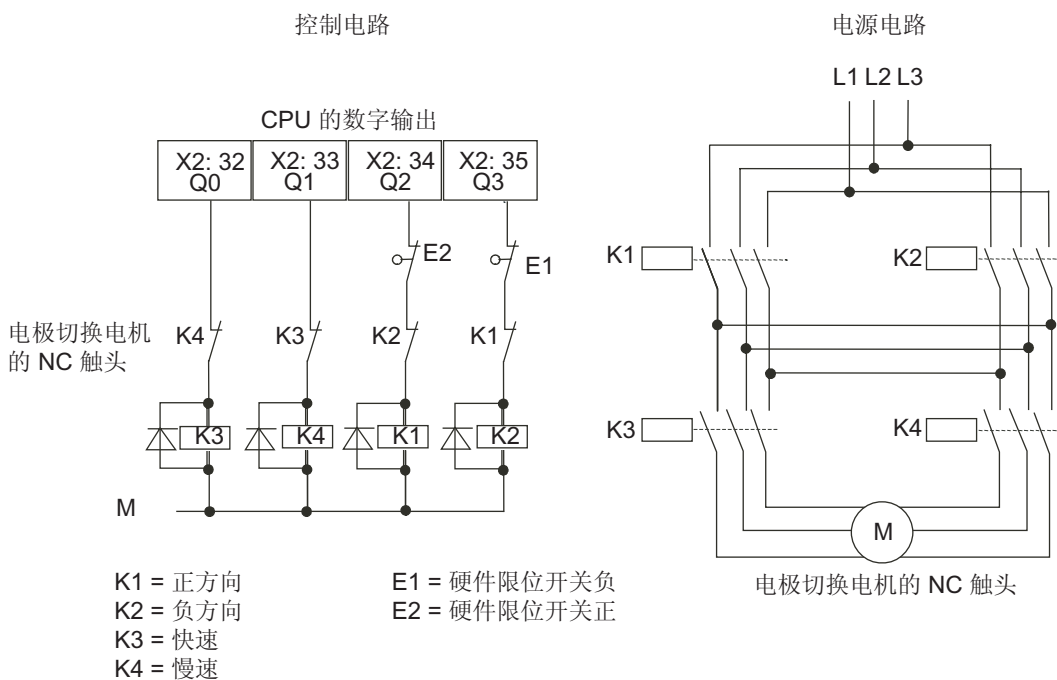
### 4.1.5 用于数字输出的断路器

#### 说明

CPU 314C-2 DP/PtP 有 4 个用于定位模式的数字输出。可通过数字输出控制动力装置。数字输出的功能取决于使用的控制模式（请参阅『驱动器参数（页码 102）』一节）。请在组态软件中选择控制模式。

输出	控制模式			
	1	2	3	4
Q0	快行速度	快行/爬行速度	快行速度	正方向快行速度
Q1	爬行速度	位置已达到	爬行速度	正方向爬行速度
Q2	正方向运行	正方向运行	正方向运行	负方向快行速度
Q3	负方向运行	负方向运行	负方向运行	负方向爬行速度

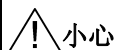
下图显示动力装置的控制和电源电路。数字输出的功能与控制模式 1 相对应。



## 断路器的工作原理

接触器 **K1** 和 **K2** 用于控制电机的运转方向。它们分别被 **NC** 触点 **K2** 和 **K1** 互锁。硬件限位开关 **E1** 和 **E2** 代表正方向/负方向限位开关。当电机超出其中一个限位开关时，电机将关闭。

接触器 **K3**（快行速度）和 **K4**（爬行速度）用于切换电机速度。它们分别被 **NC** 触点 **K4** 和 **K3** 互锁。



以下情况可造成财产损失：

如果电源接触器未被互锁，电力网中可能会出现短路。上图显示了电源接触器的互锁。

## 4.2 参数组态

### 4.2.1 参数组态的基本信息

#### 基本信息

可调整用于定位功能的参数，使其适应您的具体应用。可以为参数分配两种参数类型：

- **模块参数**

有一些基本设置只指定一次，在过程运行时不再更改。本节将介绍这些参数。

- 您可以在参数分配窗口（在 HW Config 中）中分配这些参数。
- 它们存储在 CPU 的系统存储器中。
- 当 CPU 处于 RUN 模式时，不能修改这些参数。

- **SFB 参数**

运行期间需要更改的参数位于系统功能块 (SFB) 的背景数据块中。在『使用数字输出定位（快行/爬行速度）（页码 113）』一节中对 SFB 参数进行介绍。

- 可以在 DB 编辑器中离线编辑或在用户程序中在线编辑这些参数。
- 它们存储在 CPU 的工作存储器中。
- 可以在 CPU 处于 RUN 状态时在用户程序中修改这些参数。

#### 参数分配窗口

可以在以下**参数分配窗口**中分配模块参数：

- 常规
- 地址
- 基本参数
- 驱动器
- 轴
- 编码器
- 诊断

参数分配窗口是自说明的。可以在以下各节和参数分配窗口的集成帮助中找到参数说明。

---

#### 说明

如果已为计数技术分配通道 0 或通道 1，则无法为定位技术分配参数。

---

## 4.2.2 使用参数分配窗口进行组态

### 要求

调用参数分配窗口的先决条件是已经创建了可以保存参数的项目。

### 步骤

1. 启动 SIMATIC 管理器，在项目中调用 HW Config。
2. 双击 CPU 的“定位”子模块。“特性”对话框打开。
3. 为“定位”子模块分配参数，然后使用“确定”退出参数分配窗口。
4. 使用“站 > 保存并编译”将组态保存在 HW Config 中。
5. 当 CPU 处于 STOP 模式时，使用“PLC > 下载到模块...”将参数数据下载到 CPU。现在数据即存储在 CPU 的系统数据存储器中。
6. 将 CPU 切换至 RUN 模式。

### 在线帮助

在您分配参数时，参数分配窗口的在线帮助可为您提供支持。可通过如下几种方式来调用在线帮助：

- 在相应的视图中，按下 **F1** 键
- 在不同的参数分配窗口中单击**帮助按钮**。

## 4.2.3 基本参数

### 中断选择参数

参数	值范围	缺省
中断选择	<ul style="list-style-type: none"> <li>• 无</li> <li>• 诊断</li> </ul>	无

在此可以指定是否触发诊断中断。诊断中断在『组态和判断诊断中断（页码 146）』一节中介绍。

### 4.2.4 驱动器参数

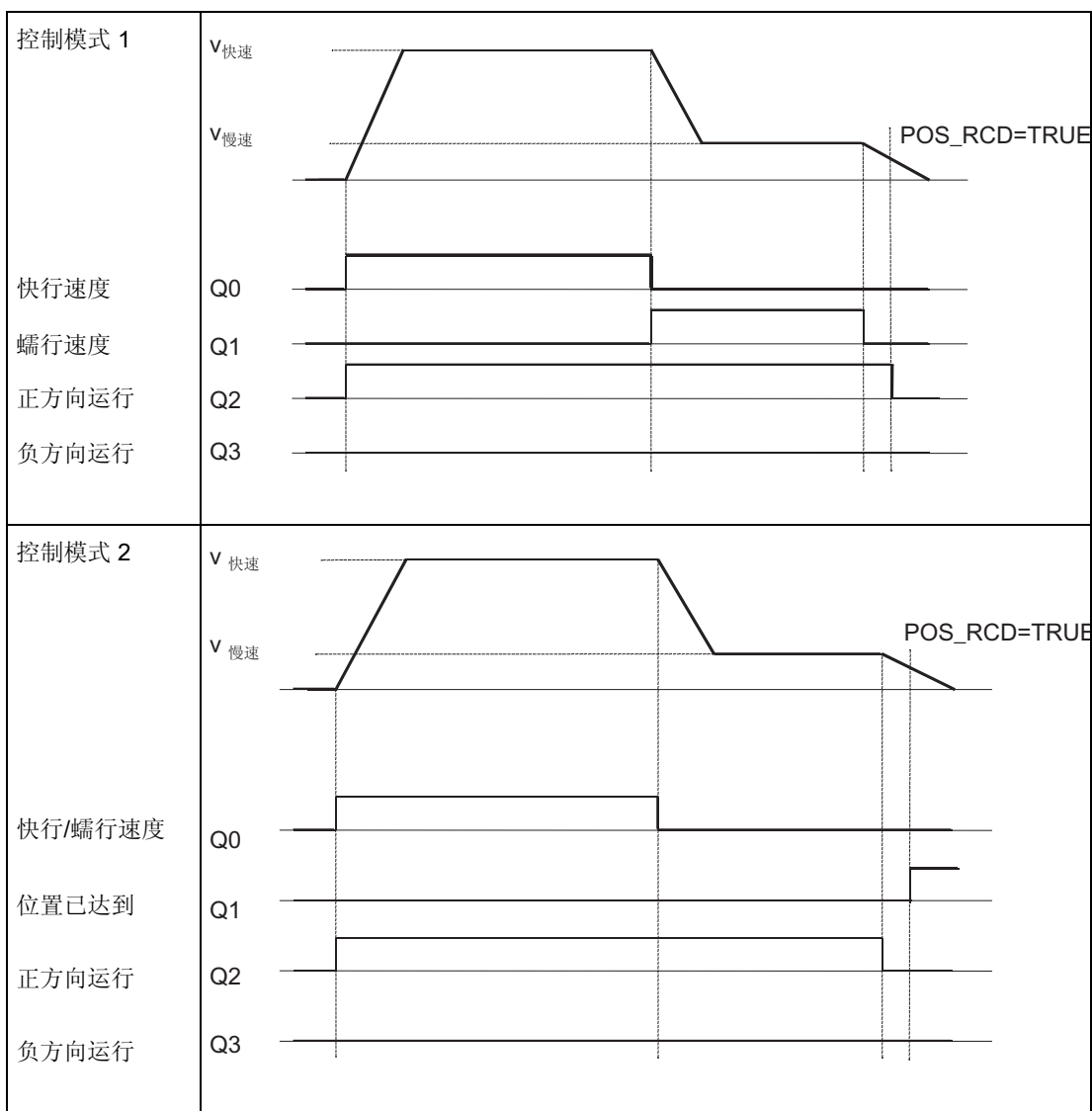
#### 控制模式参数

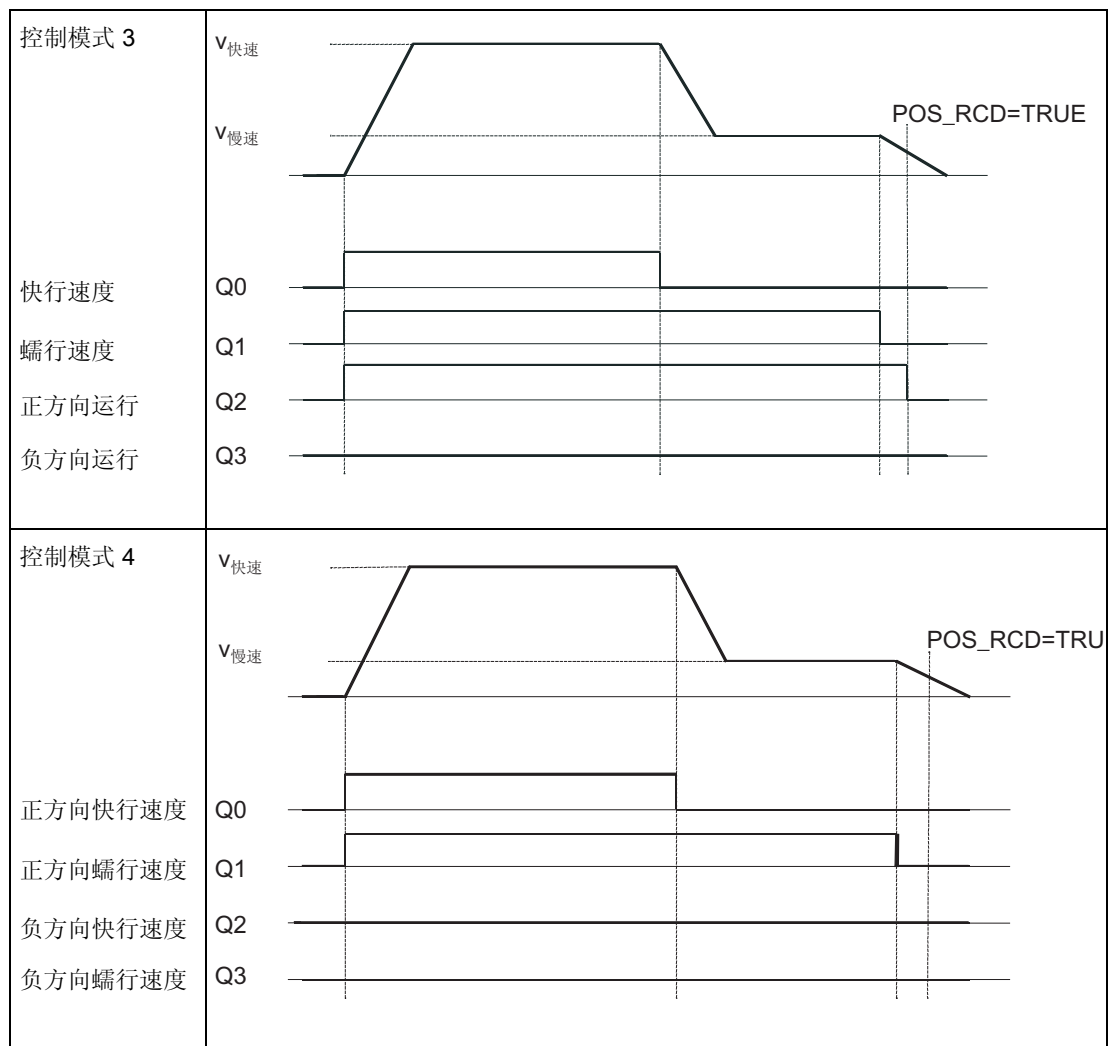
参数	值范围	缺省
控制模式	1-4	1

控制模式说明 4 种数字输出（Q0 到 Q3）如何通过转换器控制来操作连接的电机。

可选择 4 种不同的控制模式：下图显示了这四种控制模式。

以下各图分别显示以正方向逼近的情形（POS\_RCD = 反馈信号）。





4.2 参数组态

控制模式 1

控制模式 1	快行速度		爬行速度		位置已达到 (POS_RCD)
	正方向	负方向	正方向	负方向	
Q0	1	1	0	0	-
Q1	0	0	1	1	-
Q2	1	0	1	0	-
Q3	0	1	0	1	-

控制模式 2

控制模式 2	快行速度		爬行速度		位置已达到 (POS_RCD)
	正方向	负方向	正方向	负方向	
Q0	1	1	0	0	0
Q1	0	0	0	0	1
Q2	1	0	1	0	0
Q3	0	1	0	1	0

控制模式 3

控制模式 3	快行速度		爬行速度		位置已达到 (POS_RCD)
	正方向	负方向	正方向	负方向	
Q0	1	1	0	0	-
Q1	1	1	1	1	-
Q2	1	0	1	0	-
Q3	0	1	0	1	-

控制模式 4

控制模式 4	快行速度		爬行速度		位置已达到 (POS_RCD)
	正方向	负方向	正方向	负方向	
Q0	1	0	0	0	-
Q1	1	0	1	0	-
Q2	0	1	0	0	-
Q3	0	1	0	1	-



## 目标范围参数

参数	值范围	缺省
目标范围	0 到 200,000,000 个脉冲 CPU 将向上舍入奇数值。	50

目标范围在目标周围对称排列。

当值为 0 时，POS\_RCD 不会置位为 TRUE，直到超出目标或达到一个脉冲的精度。

目标范围限制为：

- 旋转轴的旋转轴范围
- 线性轴的工作范围

## 监视时间参数

参数	值范围	缺省
时间监视	<ul style="list-style-type: none"> <li>• 0 到 100 000 ms</li> <li>• 0 = 无监视</li> </ul> 舍入操作由 CPU 在 4 ms 内完成。	2000

CPU 使用此监视时间来监视

- 位置的实际值
- 目标逼近

当该值设置为“0”时，会关闭实际值和目标逼近监视。

## 实际值参数

参数	值范围	缺省
实际值监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	是

在监视时间内，移动轴必须在指定方向经过至少一个脉冲的距离。

运行开始时，实际值监视是打开的。在达到关断位置前，它一直处于激活状态。

当监视时间设置为“0”时，会关闭实际值监视。

当监视装置响应时，将取消运行。

CPU 不会检测数字输入是否有故障。可启用实际值监视来间接检测编码器或驱动器故障。

## 目标逼近监视参数

参数	值范围	缺省
目标逼近监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否

达到关断位置后，轴必须在监视时间内达到目标范围。

当监视时间设置为“0”时，会关闭目标逼近监视。

## 目标范围监视参数

参数	值范围	缺省
目标范围监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否

达到目标范围后，会对驱动器进行监视，以检查它是停留在已逼近的目标位置还是与之偏离。

当监视装置响应时，将生成一个外部出错消息。这将取消激活监视。在启动新运行前，不会重新打开监视。

## 最大频率参数：位置反馈

参数	值范围	缺省
最大频率： 位置反馈	<ul style="list-style-type: none"> <li>• 60 kHz</li> <li>• 30kHz</li> <li>• 10 kHz</li> <li>• 5kHz</li> <li>• 2kHz</li> <li>• 1kHz</li> </ul>	60 kHz

可用固定步长设置位置反馈信号（编码器信号 A、B、N）的最大频率。

## 最大频率参数：伴随信号

参数	值范围	缺省
最大频率： 伴随信号	<ul style="list-style-type: none"> <li>• 60 kHz</li> <li>• 30 kHz</li> <li>• 10 kHz</li> <li>• 5 kHz</li> <li>• 2 kHz</li> <li>• 1 kHz</li> </ul>	10 kHz

可用固定步长设置长度测量和参考点开关信号的最大频率。

## 4.2.5 轴参数

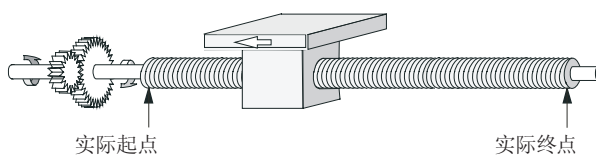
### 轴类型参数

参数	值范围	缺省
轴类型	<ul style="list-style-type: none"> <li>• 线性轴</li> <li>• 旋转轴</li> </ul>	线性轴

可以控制线性轴和旋转轴。

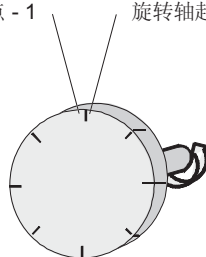
### 说明

**线性轴**的最大行程范围受到机械限制。



**旋转轴**不受机械停止的限制。

可显示的最大值 = 旋转轴终点 - 1      旋转轴起点 (坐标 0) = 旋转轴终点



旋转轴的旋转从“零”坐标开始，到坐标“旋转轴终点 - 1”处结束。“零”坐标实际上与“旋转轴终点”坐标相同（都等于 0）。实际位置值显示在该点进行切换。它总是以正值形式显示。

软件限位开关开始/结束

参数	值范围	缺省
软件限位开关开始/结束	软件限位开关开始 软件限位开关结束 • $-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	-100 000 000 +100 000 000

软件限位开关仅适用于线性轴。

由这些软件限位开关来限制工作范围。

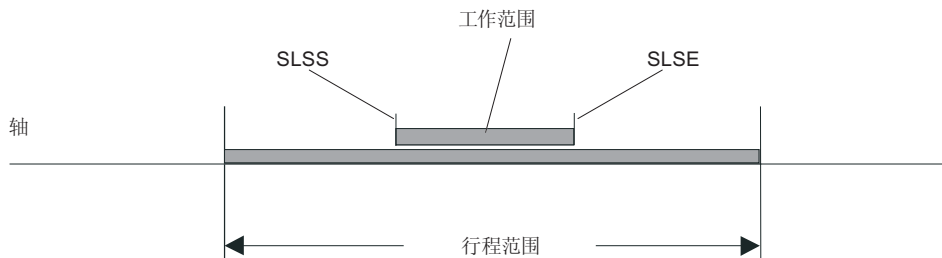
软件限位开关属于工作范围之内。

如果轴同步且已开启工作范围监视，将监视软件限位开关。

每次 CPU 进行 STOP-RUN 转换后，轴最初是不同步的。

软件限位开关开始 (SLSS) 值必须始终小于软件限位开关结束 (SLSE) 值。

工作范围必须处于行程范围之内。此行程范围代表 CPU 能够处理的值范围。



SLSS = 软件限位开关开始  
SLSE = 软件限位开关结束

旋转轴终点参数

参数	值范围	缺省
旋转轴终点	• 1 到 $10^9$ 个脉冲	100 000

“旋转轴终点”的值是理论上可能的最高实际值。它的实际位置与旋转轴的起始点相同（都等于“0”）。

显示的最大旋转轴值为“旋转轴终点 - 1”。

实例：旋转轴终点 = 1,000

显示的切换情况是：

- 正旋转方向，从 999 到 0
- 负旋转方向，从 0 到 999

长度测量参数

参数	值范围	缺省
长度测量	<ul style="list-style-type: none"> <li>• 关</li> <li>• 在正跳沿 DI 处启动/结束</li> <li>• 在负跳沿 DI 处启动/结束</li> <li>• 正跳沿启动，负跳沿结束</li> <li>• 负跳沿启动，正跳沿结束</li> </ul>	关

### 参考点坐标参数

参数	值范围	缺省
参考点坐标	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0

CPU 进行 STOP-RUN 转换后，实际值被设置成和参考点坐标值相同。

在某个参考点逼近后，将参考点坐标值分配给该参考点。

参考点坐标值必须位于线性轴的工作范围（含软件限位开关）内。

旋转轴的参考点坐标值必须位于 0 到“旋转轴终点 - 1”范围内。

### 参考点开关的参考点位置

参数	值范围	缺省
参考点开关的参考点位置	<ul style="list-style-type: none"> <li>• 正方向（实际值增加）</li> <li>• 负方向（实际值减少）</li> </ul>	正方向

此参数根据参考点开关定义参考点位置。

### 行程范围监视参数

参数	值范围	缺省
行程范围监视	是（设置为固定值）	是

使用行程范围监视检查是否超出了允许的行程范围 -5 x 10<sup>8</sup> 至 +5 x 10<sup>8</sup>。不能关闭此监视功能（在“监视”参数中永久打开）。

当此监视响应时，会取消同步并中止运行。

### 参数工作范围监视（仅适用于线性轴）

参数	值范围	缺省
工作范围监视（仅适用于线性轴）	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	是

在此可以指定是否监视线性轴的工作范围。此时，将监视实际位置值以检查它是否超出了软件限位开关的范围。此监视只影响同步的轴。

软件限位开关自身的坐标属于工作范围。

当监视响应时，将取消运行。

## 4.2.6 编码器参数

### 概述

参数	值范围	缺省
编码器每转增量	1 到 $2^{23}$ 个脉冲	1000

“编码器每转增量”参数指定编码器每旋转一周输出的增量。有关数值的信息，请参见编码器说明。

CPU 将判断增量四次（一个增量对应着四个脉冲，请参阅『增量编码器（页码 149）』一节）。

### 计数方向参数

参数	值范围	缺省
计数方向	<ul style="list-style-type: none"> <li>• 常规</li> <li>• 反转</li> </ul>	常规

使用“计数方向”参数可将路径监视的方向调整为线性轴的运动方向。还应考虑所有传动元件（例如，联接器和齿轮）的旋转方向。

- 标准 = 增加计数脉冲 = 减少实际值
- 反转 = 增加计数脉冲 = 减少实际值

### 缺少脉冲（零标记）监视参数

参数	值范围	缺省
缺少脉冲（零标记）监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否

启用零标记监视时，CPU 将监视两个连续零标记信号（编码器信号 N）之间脉冲差异的一致性。

如果已组态的编码器旋转一周产生的脉冲数不能被 10 或 16 整除，则无论在参数分配窗口中的设置如何，都会自动关闭零标记监视。

### 说明

零标记信号的最小脉冲宽度为  $8.33 \mu\text{s}$ （等于最大频率 60 kHz）。

如果使用的编码器的零标记信号通过“AND”运算与编码器信号 A 和 B 合并，脉冲宽度将减半至周期时间的 25%。这会将零标记监视的最大频率减小为 30kHz。

不能识别以下内容：

- 错误地分配了编码器的每转增量数。
- 零标记信号失效。

当此监视响应时，会取消同步并中止运行。

## 4.2.7 组态诊断

### 用于监视的诊断中断

响应监视功能可触发诊断中断。

### 启用诊断中断

要求：在“基本参数”屏幕中启用诊断中断，并在“驱动器”、“轴”和“编码器”屏幕中打开相应的监视装置。

参数	值范围	缺省
缺少脉冲（零标记）	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
行程范围	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
工作范围 （对于线性轴）	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
实际值	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
目标逼近	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
目标范围	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否

## 4.3 集成到用户程序中

### 步骤

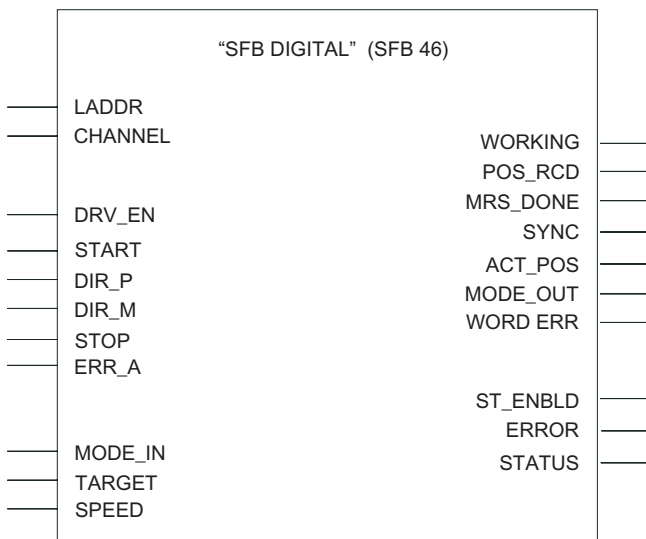
在用户程序中控制定位功能。调用系统功能块 **SFB DIGITAL (SFB 46)**。SFB 在“标准库”的“系统功能块” > “块”下。

下面的章节帮助您为您的应用设计用户程序。

### 调用 SFB

通过对应的背景数据块调用 SFB

实例：CALL SFB 46、DB22



### 说明

因为 SFB 不能中断本身，所以不能在其它优先级下的另一程序段中调用已在程序中组态的 SFB。

实例：不允许在 OB 1 和中断 OB 中调用 SFB。

### 背景数据块

SFB 参数存储在背景数据块中。在『使用数字输出定位（快行/蠕行速度）（页码 113）』一节中介绍这些参数。

可以通过以下方式访问这些参数：

- DB 号和 DB 中的绝对地址
- DB 号和 DB 中的符号地址

基本的功能参数也将分配给该块。可在 SFB 中直接声明输入参数值，或者可以判断输出参数。



## 4.4 用于使用数字输出定位的功能

### 4.4.1 使用数字输出定位（快行/蠕行速度）

#### 概述

四个永久分配的 24 V 数字输出 (Q0-Q3) 控制驱动器。这些数字输出根据组态的控制类型，来控制方向和速度阶段（快行/蠕行速度）。

位置反馈通过一个非对称 24 V 增量编码器实现，该编码器配有相位差为 90° 的两个信号轨迹。

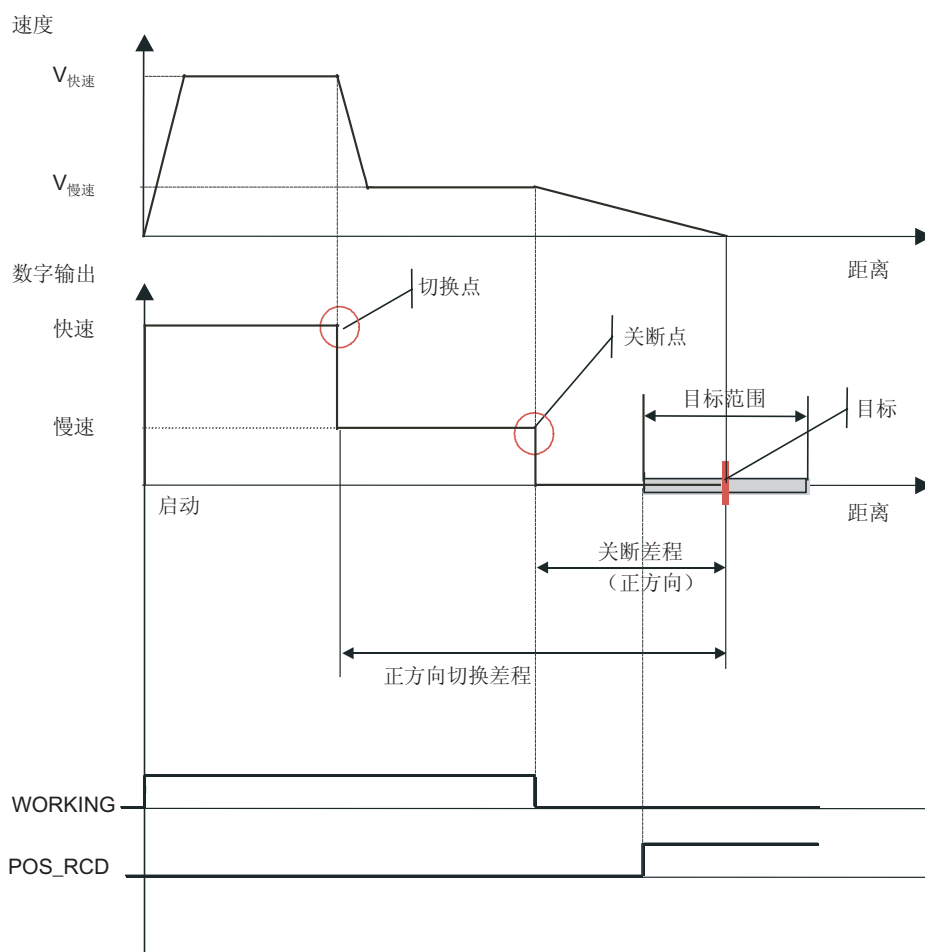
#### 启动运行

根据操作模式，使用 START、DIR\_P 或 DIR\_M 启动运行。

#### 使用数字输出定位

下图的上部分显示了运行曲线。我们只简单地假定在行程上实际速度只做线性变化。

图的下半部分显示相应的数字输出曲线。快行速度和蠕行速度由数字输出 0 和 1 的组合来确定（请参阅『驱动器参数（页码 102）』一节）。



4.4 用于使用数字输出定位的功能

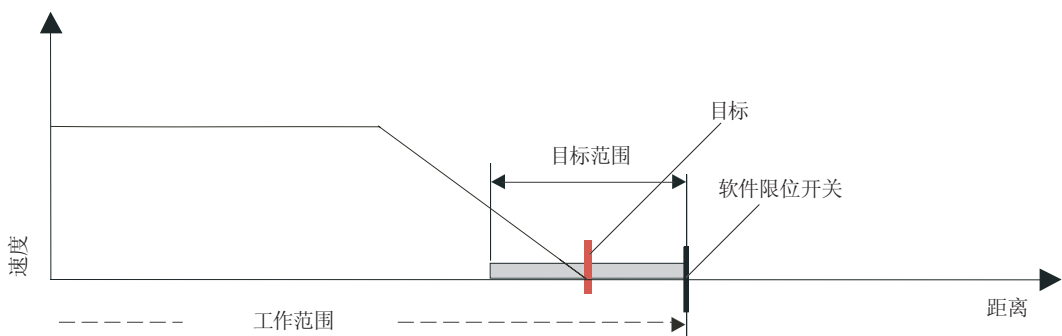
- 首先，以  $V_{快速}$  速度逼近目标。
- 在**换向点**处，驱动器将切换为爬行速度 ( $V_{慢速}$ )。
- 驱动器在**关断位置**关闭。
- 通过**转换差程**和**关断差程**参数中指定的值来确定每个目标逼近的换向点和关断点。对于前向（正方向）和反向（负方向）运行，可指定不同的转换差程和关断差程。
- 达到关断位置时，会终止运行 ( $WORKING= FALSE$ )。此时，可启动一个新运行。
- 当实际位置值达到**目标区域**时，即达到了指定目标 ( $POS\_RCD = TRUE$ )。如果在没有启动新运行的情况下实际位置值偏离目标区域，则不会复位“位置已达到”信号。

工作范围

借助软件限位开关坐标来确定工作范围。运行绝不可以超过同步线性轴的工作范围。

必须始终根据工作范围指定运行目标。

当轴超出工作范围之后，只能以点动模式返回。



## 监视功能

参数分配窗口可帮助您分别启用各个监视功能。当其中一个监视功能响应时，运行会取消并发生外部错误（使用 ERR\_A 确认）。

监视	说明
缺少脉冲 (零标记)	<p>启用零标记监视时，CPU 将检查两个连续零标记信号之间脉冲差异的一致性。</p> <p>如果已组态的编码器每转的脉冲数不能被 10 或 16 整除，则无论在参数分配窗口中的设置如何，都会自动关闭零标记监视。</p> <p>零标记信号的最小脉冲宽度为 8.33 <math>\mu</math>s（对应于最大频率 60 kHz）。</p> <p>如果使用的编码器的零标记信号通过“AND”运算与编码器信号 A 和 B 合并，脉冲宽度将减半至周期时间的 25%。这会将零标记监视的最大频率减小为 30 kHz。</p> <p>不能识别以下内容：</p> <ul style="list-style-type: none"> <li>• 错误地分配了编码器的每转增量数。</li> <li>• 零标记信号出错。</li> </ul> <p>CPU 对错误的响应：取消同步，取消运行。</p>
行程范围	<p>CPU 使用行程范围监视检查是否超出了允许的行程范围 <math>-5 \times 10^8</math> 至 <math>+5 \times 10^8</math>。不能关闭此监视功能（在“监视”参数中永久打开）。</p>
工作范围	<p>CPU 使用行程范围监视检查实际值是否超出了软件限位开关的范围。</p> <p>对于旋转轴定位监视，不能打开此装置。</p> <p>此监视只影响同步的轴。</p> <p>软件限位开关自身的坐标属于工作范围。</p> <p>CPU 对错误的响应：运行被取消。</p>
实际值	<p>在监视时间内，轴必须在指定方向至少移动一个脉冲。</p> <p>运行开始时，实际值监视是打开的，并在达到关断位置前一直处于激活状态。</p> <p>当监视时间设置为“0”时，会关闭实际值监视。</p> <p>当监视响应时，将取消运行。</p> <p>CPU 对错误的响应：运行被取消。</p>
目标逼近	<p>达到关断位置后，轴必须在监视时间内达到目标范围。</p> <p>当设置的监视时间为“0”时，会关闭目标逼近监视。</p> <p>CPU 对错误的响应：取消运行并关闭输出。</p>
目标范围	<p>达到目标范围后，CPU 会对驱动器进行监视，以检查它是停留在已逼近的目标位置还是与之偏离。</p> <p>当监视功能响应时，将生成一个外部错误消息。使用 ERR_A（正跳沿）确认外部错误时，将关闭监视。</p> <p>在启动新运行前，不会重新打开监视。</p> <p>CPU 对错误的响应：运行被取消。</p>

## 终止运行

共有以下三种不同的方法可以终止运行：

- 目标逼近
- 取消激活
- 取消

## 目标逼近

运行归位表示达到指定目标后自动终止运行。

为达到指定目标，会以“相对和绝对增量逼近”操作模式执行运行归位。

## 取消激活

在下列情况下，会逐渐关闭驱动器：

- 在 **STOP = TRUE**（达到目标前）时的所有操作模式中
- 当停止和运行方向被反转时，在“点动模式”中
- 当检测到同步位置和方向反转时，在“参考点逼近”模式中  
顺序类似于目标逼近。

## 取消

无论转换/关断差程如何，都会立即终止运行。该控制模式的所有相关输出均被立即取消激活。

可以随时取消，也可以在停止时取消。

在以下情况下将取消运行：

- 取消驱动器使能信号 (**DRV\_EN = FALSE**)
- CPU 切换为 **STOP** 模式时
- 发生外部错误时（例外：目标逼近/目标范围监视）

反应：

- 立即终止当前或中断的运行 (**WORKING = FALSE**)
- 最后一个目标 (**LAST\_TRG**) 被设置为实际值 (**ACT\_POS**)
- 删除剩余行程，即不能恢复“相对增量逼近”
- 不会置位“位置已达到” (**POS\_RCD**)

## 4.4.2 SFB DIGITAL (SFB 46) 的基本组态

### 基本参数概述

本部分将介绍对于所有操作模式都相同的参数。操作模式特定的参数在特定模式下介绍。

根据您的应用组态以下 SFB 输入参数。

#### 输入参数

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
LADDR	WORD	0	在“HW Config”中指定的子模块 I/O 地址。 如果 I/O 地址不相同，必须指定两者中的较低一个。	CPU 特定	310 (十六进制)
CHANNEL	INT	2	通道号	0	0
STOP	BOOL	4.4	停止运行 STOP = TRUE 可用于提前停止/中断运行。	TRUE/FALSE	FALSE
ERR_A	BOOL	4.5	组错误确认 使用 ERR_A（正跳沿）确认外部错误。	TRUE/FALSE	FALSE
SPEED	BOOL	12.0	快行/爬行速度两个速度阶段 TRUE = 快行速度 FALSE = 爬行速度 在运行过程中速度不能有变化。	TRUE/FALSE	FALSE

#### 未分配给块的输入参数（静态局部数据）

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
CHGDIFF_P	DINT	28	正方向转换差程： “正方向转换差程”定义转换位置，从该位置开始，以正方向移动的驱动器将从快行速度转换为爬行速度。	0 到 +10 <sup>8</sup> 个脉冲	1000
CUTOFF-DIFF_P	DINT	32	正方向关断差程： “正方向关断差程”定义关断位置，在该位置以正方向爬行速度操作的驱动器将关闭。	0 到 +10 <sup>8</sup> 个脉冲	100
CHGDIFF_M	DINT	36	负方向转换差程： “负方向转换差程”定义转换位置，在该位置以负方向移动的驱动器将从快行速度转换为爬行速度。	0 到 +10 <sup>8</sup> 个脉冲	1000
CUTOFF-DIFF_M	DINT	40	负方向关断差程： “负方向关断差程”定义关断位置，在该位置以负方向爬行速度运行的驱动器将关闭。	0 到 +10 <sup>8</sup> 个脉冲	100

## 转换/关断差程的规则

- 正值和负值可以不同。
- 转换差程必须大于/等于关断差程。
- 关断差程必须大于或等于目标范围的一半。
- 在转换位置和关断位置之间选择足够的距离，以确保驱动器的速度可减小为爬行速度。
- 在转换位置和目标之间选择足够的距离，以确保驱动器达到目标区域并在那里停止。
- 要移动的距离至少要等于关断差程
- 转换/关断差程被限制为行程范围 (+10<sup>8</sup>) 的 1/10。

## 输出参数

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
WORKING	BOOL	14.0	正在运行	TRUE/FALSE	FALSE
ACT_POS	DINT	16	实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	20	启用/设置操作模式	0、1、3、4、5	0
ERR	WORD	22	外部错误 <ul style="list-style-type: none"> <li>• 位 2: 缺少脉冲监视</li> <li>• 位 11: 行程范围监视（始终为 1）</li> <li>• 位 12: 工作范围监视</li> <li>• 位 13: 实际值监视</li> <li>• 位 14: 目标逼近监视</li> <li>• 位 15: 目标范围监视</li> <li>• 其它位将保留</li> </ul>	每个位 0 或 1	0
ST_ENBLD	BOOL	24.0	如果符合以下所有条件，CPU 将置位“启动使能”： <ul style="list-style-type: none"> <li>• 无错参数分配 (PARA = TRUE)</li> <li>• 无 STOP 挂起 (STOP = FALSE)</li> <li>• 没有发生外部错误 (ERR = 0)</li> <li>• 置位了“驱动器使能” (DRV_EN = TRUE)</li> <li>• 无激活的定位运行 (WORKING = FALSE)</li> </ul> 例外：点动模式	TRUE/FALSE	TRUE
ERROR	BOOL	24.1	运行启动/恢复错误	TRUE/FALSE	FALSE
STATUS	WORD	26	错误 ID	0 到 FFFF (十六进制)	0

## 未分配给块的输出参数（静态局部数据）

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
PARA	BOOL	44.0	轴已组态	TRUE/FALSE	FALSE
DIR	BOOL	44.1	当前/上一方向的意义 FALSE = 向前（正方向） TRUE = 反向（负方向）	TRUE/FALSE	FALSE
CUTOFF	BOOL	44.2	驱动器位于关断范围内 （从关断位置到下一运行开始）	TRUE/FALSE	FALSE
CHGOVER	BOOL	44.3	驱动器位于转换范围内 （从达到转换位置到下一运行开始）	TRUE/FALSE	FALSE
DIST_TO_GO	DINT	46	实际剩余行程	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
LAST_TRG	DINT	50	上一/实际目标 <ul style="list-style-type: none"> <li>绝对增量逼近： 运行启动 LAST_TRG = 实际绝对目标 (TARGET)</li> <li>相对增量逼近： 运行启动时的距离为 LAST_TRG = 前一运行的 LAST_TRG +/- (TARGET) 所指定的距离。</li> </ul>	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0

### 4.4.3 点动模式

#### 说明

在“点动”模式下，可在正方向或负方向上运行驱动器。未指定目标。

#### 要求

- 您已通过参数分配窗口分配模块参数并将它们下载到了 CPU (PARA = TRUE)。
- 您已按照『SFB DIGITAL (SFB 46) 的基本组态 (页码 117)』一节中所述分配了 SFB 的基本参数。
- 未产生外部错误 ERR。您必须使用 ERR\_A (正跳沿) 来确认排队的外部错误。
- 启动使能 ST\_ENBLD = TRUE。
- 在点动模式下，既可操作同步 (SYNC = TRUE) 轴也可操作非同步 (SYNC = FALSE) 轴。

#### 启动/停止运行

通过置位控制位 DIR\_P 或 DIR\_M 启动驱动器。

- 每次调用 SFB 时，都会对 DIR\_P 和 DIR\_M 这两个控制位进行评估以便检查逻辑级的更改。
- 如果这两个控制位均为 FALSE，则运行将减速。
- 如果这两个控制位均为 TRUE，则运行也将减速。
- 当其中一个控制位置位为 TRUE 时，轴将在相应方向上移动。



## 步骤

- 按“设置”列中所指定的内容组态以下 SFB 输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
DRV_EN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE	TRUE
DIR_P	BOOL	4.2	点动模式，正方向（正跳沿）	TRUE/FALSE	FALSE	DIR_P 或 DIR_M = TRUE
DIR_M	BOOL	4.3	点动模式，负方向（正跳沿）	TRUE/FALSE	FALSE	
MODE_IN	INT	6	操作模式，1 = “点动模式”	0、1、3、4、5	1	1

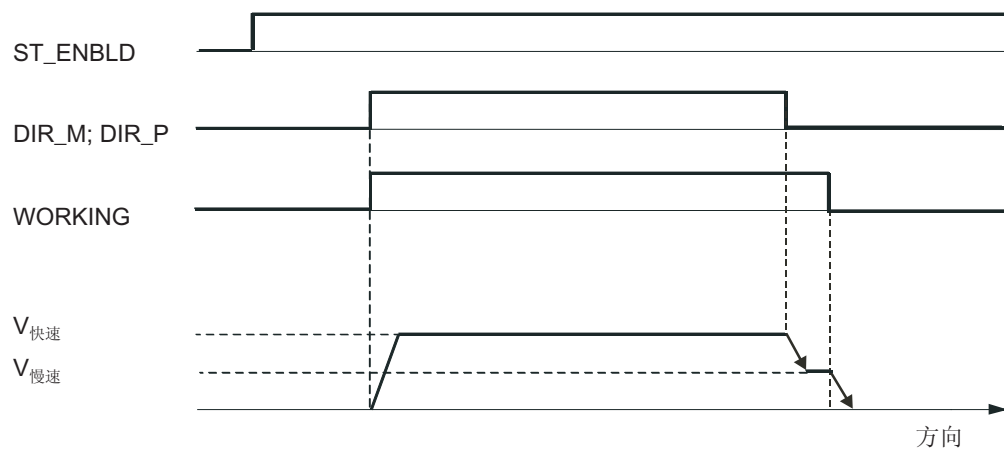
- 调用 SFB。

SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
WORKING	BOOL	14.0	正在运行	TRUE/FALSE	FALSE
ACT_POS	DINT	16	当前的实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	20	激活/设置操作模式	0、1、3、4、5	0

## 结果

- 启动运行后将立即置位 WORKING = TRUE。当复位方向位 DIR\_P 或 DIR\_M 或置位 STOP = TRUE 时，运行将被终止 (WORKING = FALSE)。
- 如果在解释 SFB 调用时发生错误，则 WORKING = FALSE 且 ERROR 将被置位为 TRUE。然后将使用 STATUS 参数指出确切的出错原因（请参阅『错误列表（页码 151）』一节）。
- 在点动模式下，ST\_ENBLD 始终置位为 TRUE。
- 不会置位“已到位” (POS\_RCD)。



#### 4.4.4 参考点逼近

##### 4.4.4.1 参考点逼近 - 工作原理

###### 说明

CPU 打开后，位置值 `ACT_POS` 并不是指向轴的机械位置。

为了向物理位置分配一个可复写的编码器值，必须在轴位置和编码器值之间建立一个参考（同步）。可通过为轴的某个已知位置（参考点）分配一个位置值来实现同步。

###### 参考点开关和参考点

轴上需要有参考点开关和参考点，这样才能执行参考点逼近。

- 需要**参考点开关**以确保参考信号始终具有相同的参考点（零标记），以及用于更改为参考点逼近速度。例如，可使用 `BERO` 开关。为使轴在超出参考点开关的范围之前达到参考点逼近速度，参考点开关的信号长度必须足够高。
- **参考点**是离开参考点开关之后的下一个编码器零标记。轴在参考点处同步，反馈信号 `SYNC` 将被置位为 `TRUE`。将在参数分配窗口中指定的参考坐标分配给参考点。

必须始终在参考点开关的方向上启动参考点逼近。否则，轴将进入范围限制，由于它未同步，因而没有软件限位开关。

通过在参考点开关处启动参考点逼近，可始终确保在开关的方向上启动轴（请参见实例 3）。

---

###### 说明

对于旋转轴：因为参考点要求的再现性，编码器的相应零标记必须始终位于相同的物理位置。因此，“旋转轴终点”和“编码器每转增量”必须为比例积分。实例：编码器转数与旋转轴终点的转数比为 4:1。在此情况下，零标记位于 90、180、270 和 360 度处。

零标记信号的最小脉冲宽度为 `8.33 μs`（相当于最大频率 `60 kHz`）。

如果使用的编码器的零标记信号通过“AND”运算与编码器信号 `A` 和 `B` 合并，脉冲宽度将减半至周期时间的 `25%`。从而使最大计数频率在参考期间减小为 `30 kHz`。

---

## 参考点位置

关于在参考点逼近期间的参考点位置（零标记信号），必须区分以下两种情况：

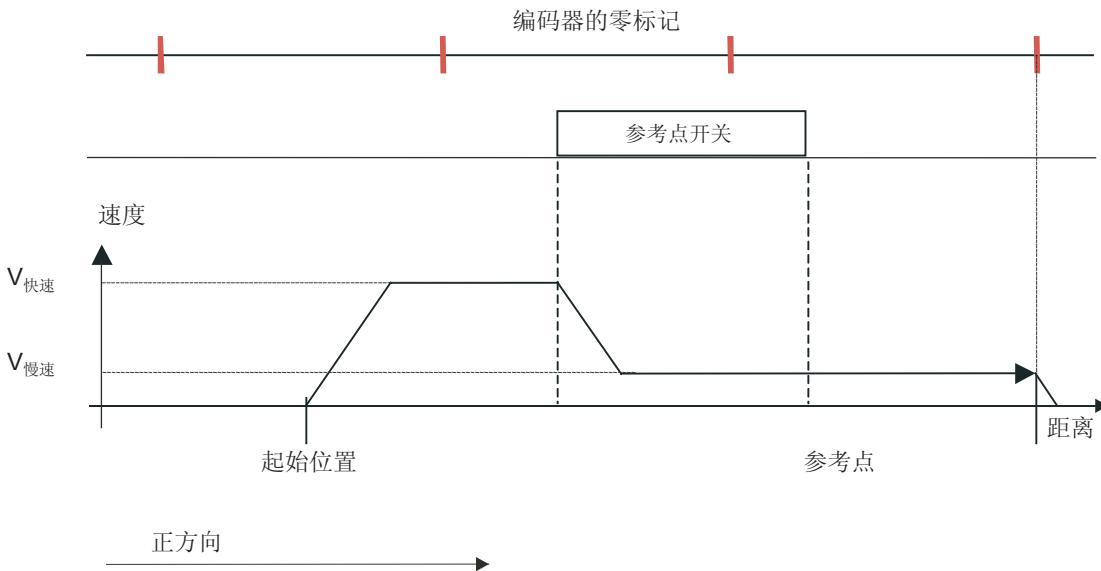
- 参考点位置从正方向指向参考点开关。
- 参考点位置从负方向指向参考点开关。

请通过参数分配窗口在参数“参考点开关的参考点位置”中进行此设置。

参考点逼近的不同情况由运行启动的方向和参考点位置来确定：

### 实例 1：

- 启动方向为正方向
- 参考点开关的参考点位置在正方向上



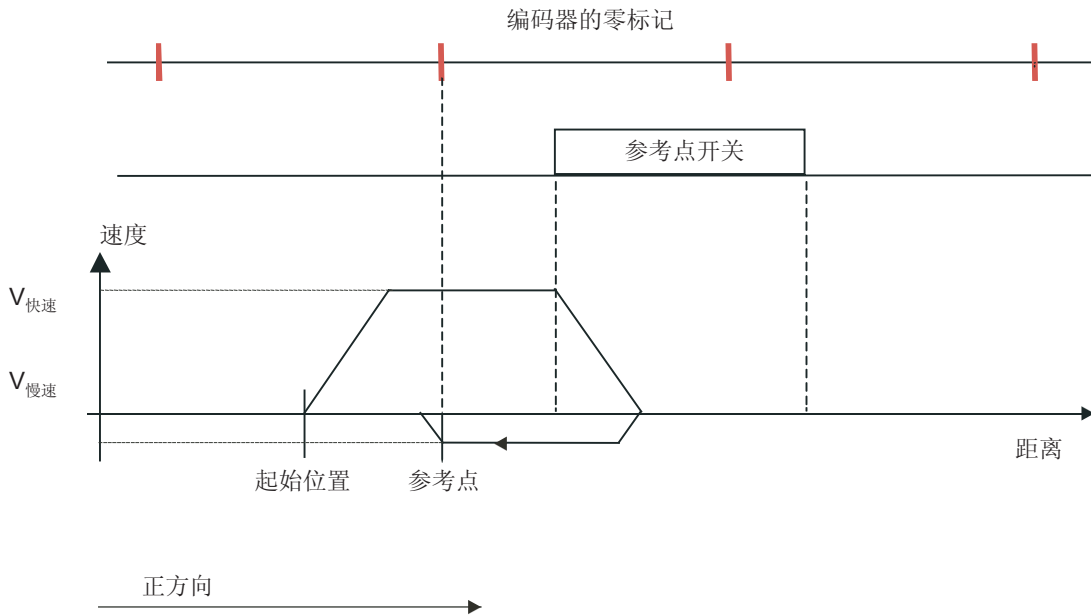
通过快速往返逼近参考点开关。

随后驱动器被切换为蠕行速度。

离开参考点开关后，驱动器将在下一个编码器零标记处被关闭。

实例 2:

- 启动方向为正方向
- 参考点在负方向上逼近参考点开关



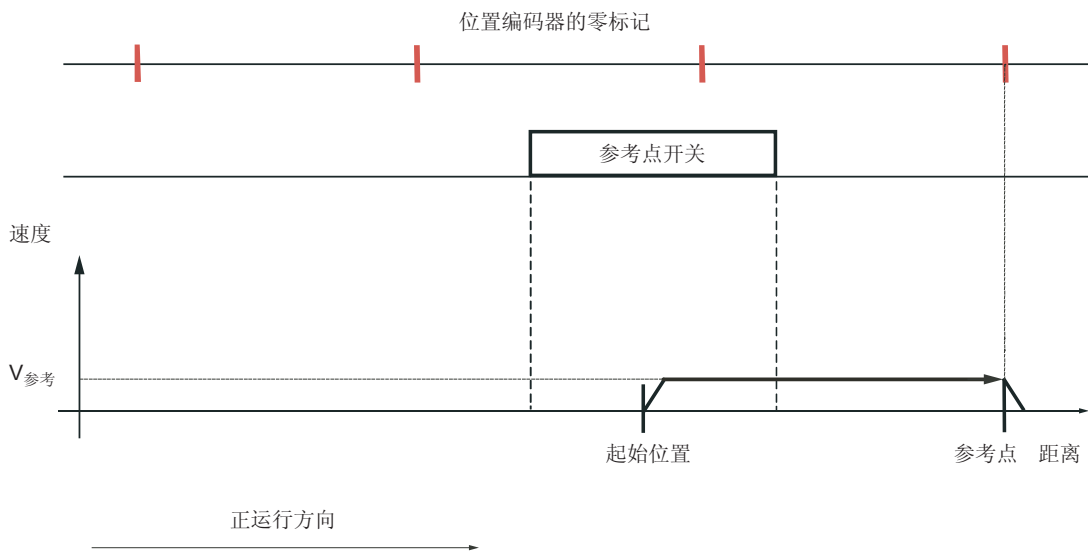
通过快速往返逼近参考点开关。

随后驱动器被转换为蠕行速度，且方向反转。

离开参考点开关后，驱动器将在下一个编码器零标记处被关闭。

## 实例 3:

- 启动位置位于参考点开关处
- 启动方向为负方向
- 参考点在正方向上逼近参考点开关



以蠕行速度执行操作。

运行将朝着您在参数分配窗口中通过参数“参考点开关的参考点位置”指定的方向执行，而与 **SFB** 中指定的方向无关。

离开参考点开关后，驱动器将在下一个编码器零标记处被关闭。

## 4.4.4.2 参考点逼近 - 步骤

## 参考点逼近的先决条件

- 具有零标记的编码器，或参考点信号开关（使用不具有零标记的编码器时）。
- 已连接参考点开关（连接器 X2，针脚 6）。
- 您已通过参数分配窗口分配模块参数并将它们下载到了 CPU (PARA = TRUE)。
- 您已按照『SFB DIGITAL (SFB 46) 的基本组态（页码 117）』一节中所述分配了 SFB 的基本参数。
- 未发生外部错误 ERR。您必须使用 ERR\_A（正跳沿）来确认排队的外部错误。
- 启动使能 ST\_ENBLD = TRUE。

## 步骤

- 按“设置”列中所指定的内容分配 SFB 的以下输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
DRV_EN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE	TRUE
DIR_P	BOOL	4.2	参考点逼近，正方向（正跳沿）	TRUE/FALSE	FALSE	DIR_P 或 DIR_M = TRUE
DIR_M	BOOL	4.3	参考点逼近，负方向（正跳沿）	TRUE/FALSE	FALSE	
MODE_IN	INT	6	操作模式， 3 = “参考点逼近”	0、1、3、4、5	1	3

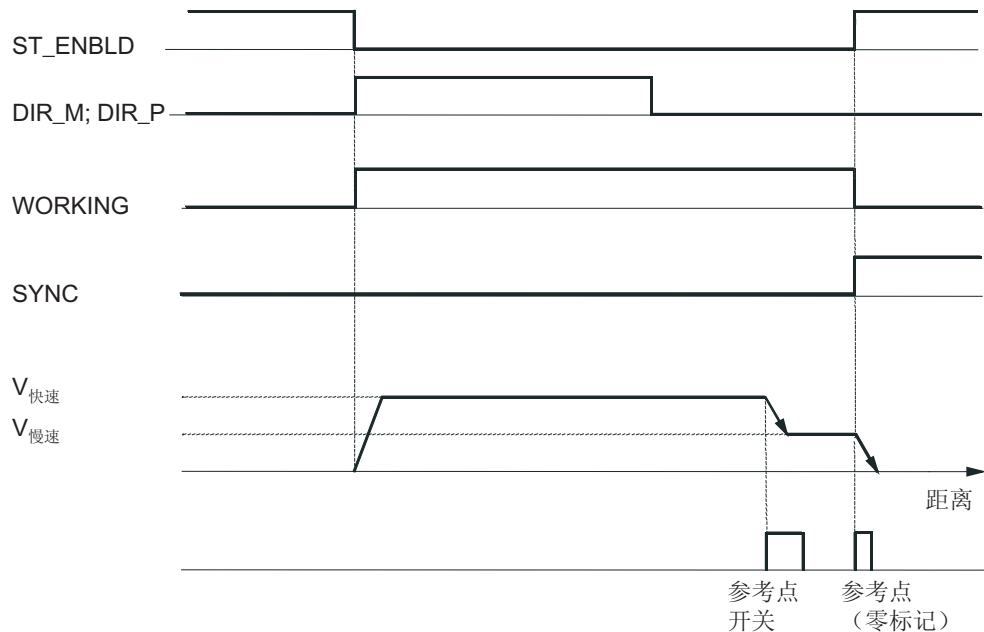
- 调用 SFB。

SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
WORKING	BOOL	14.0	正在运行	TRUE/FALSE	FALSE
SYNC	BOOL	14.3	SYNC = TRUE：轴同步	TRUE/FALSE	FALSE
ACT_POS	DINT	16	当前实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	20	激活/设置操作模式	0、1、3、4、5	0

## 结果

- 启动运行后立即置位  $WORKING = TRUE$  且  $SYNC = FALSE$ 。达到参考点后， $WORKING$  的状态将复位为  $FALSE$ 。如果执行时未发生错误，则  $SYNC = TRUE$ 。
- 在启动下一个运行前，必须复位方向位 ( $DIR\_P$  或  $DIR\_M$ )。
- 如果在解释 SFB 调用时发生错误，则  $WORKING = FALSE$  且  $ERROR$  将被置位为  $TRUE$ 。然后将使用  $STATUS$  参数指出确切的出错原因（请参阅『错误列表（页码 151）』一节）。
- 不会置位“已到位” ( $POS\_RCD$ )。



## 操作模式的影响有哪些

- 启动参考点逼近时，将清除可能存在的同步 ( $SYNC = FALSE$ )。
- 将在参考点（零标记）的正跳沿将实际位置设置为参考点坐标值，且将置位反馈信号  $SYNC$ 。
- 将在轴上确定工作范围。
- 位于工作范围内的所有点将保持其各自的原始坐标，但具有新的物理位置。

### 4.4.5 相对增量逼近模式

#### 说明

在“相对增量逼近”模式下，驱动器将自上一个目标 (LAST\_TRG) 开始，在指定方向上移动一段相对距离。

起始点不是实际位置，而是上一个指定目标 (LAST\_TRG)。这样可防止累积定位误差。启动定位后，由参数 LAST\_TRG 来指示实际目标。

#### 要求

- 您已通过参数分配窗口分配模块参数，并将它们下载到了 CPU (PARA = TRUE)。
- 您已按照『SFB DIGITAL (SFB 46) 的基本组态 (页码 117)』一节中所述分配了 SFB 的基本参数。
- 未发生外部错误 ERR。您必须使用 ERR\_A (正跳沿) 来确认排队的外部错误。
- 启动使能 ST\_ENBLD = TRUE。
- 同步 (SYNC = TRUE) 和非同步 (SYNC = FALSE) 轴均可实现“相对增量逼近”。

#### 指定行程

指定线性轴行程时，请注意以下事项：

- 行程必须大于等于关断差程。
- 如果行程小于等于目标范围的一半，将不会启动新运行。该模式将立即终止且不产生错误。
- 目标范围必须处于工作范围之内。



## 步骤

- 按“设置”列中所指定的内容分配 SFB 的以下输入参数:

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省	设置
DRV_EN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE	TRUE
DIR_P	BOOL	4.2	正方向运行 (正跳沿)	TRUE/FALSE	FALSE	DIR_P 或 DIR_M = TRUE
DIR_M	BOOL	4.3	负方向运行 (正跳沿)	TRUE/FALSE	FALSE	
MODE_IN	INT	6	操作模式, 4 = “相对增量逼近”	0、1、3、4、5	1	4
TARGET	DINT	8	以脉冲表示的距离 (仅允许正值)	0 到 109	1000	xxxx

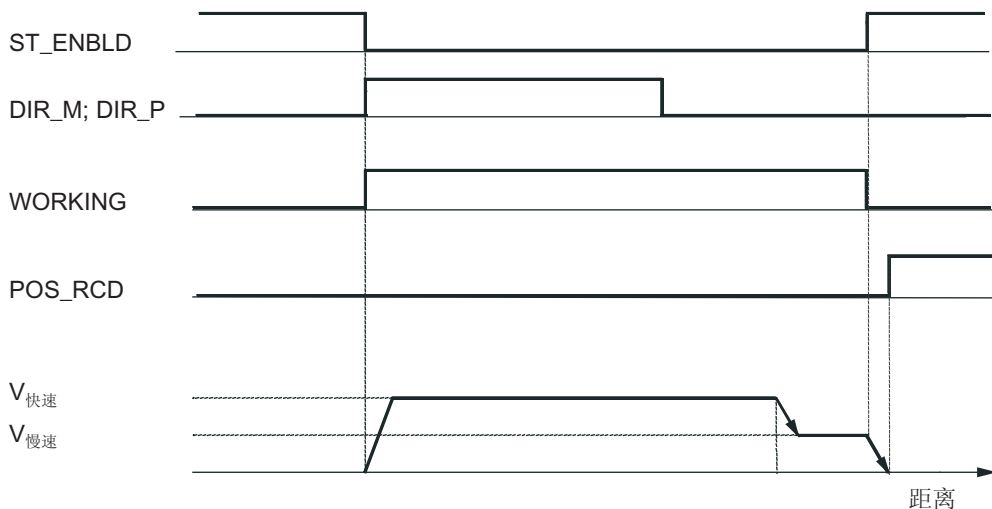
- 调用 SFB。

SFB 的输出参数提供了以下信息:

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
WORKING	BOOL	14.0	正在运行	TRUE/FALSE	FALSE
POS_RCD	BOOL	14.1	已到位	TRUE/FALSE	FALSE
ACT_POS	DINT	16	当前实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	20	激活/设置操作模式	0、1、3、4、5	0

结果

- 启动运行后将立即置位 WORKING = TRUE。在关断点处将 WORKING 复位为 FALSE。达到指定目标后，将置位 POS\_RCD = TRUE。
- 在启动下一个运行前，必须复位方向位 (DIR\_P 或 DIR\_M)。
- 如果在解释 SFB 调用时发生错误，则 WORKING = FALSE 且 ERROR 将被置位为 TRUE。然后将使用 STATUS 参数指出确切的出错原因 (请参阅『错误列表 (页码 151)』一节)。



停止运行/目标范围未达到

使用 STOP = TRUE 停止运行时，如果尚未达到关断范围 (剩余行程大于关断差程)，则依据后续操作模式/作业的不同，有以下几种选项。

选项	响应
在同一方向继续该运行	将不解释运行参数。轴移动到被停止的运行的目标点 (LAST_TRG)。
在相反方向继续该运行	将不解释运行参数。轴移动到被停止的运行的起始点。
启动新的“绝对增量逼近”	轴移动到指定的绝对目标。
作业“删除剩余行程”	将删除剩余行程 (目标值与实际值之差)。在启动“相对增量逼近”时再次解释运行参数，然后轴将移动到当前实际位置值。

## 4.4.6 绝对增量逼近模式

### 说明

在“绝对增量逼近”模式下，将逼近绝对目标位置。

### 要求

- 您已通过参数分配窗口分配模块参数并将它们下载到了 CPU (PARA = TRUE)。
- 您已按照『SFB DIGITAL (SFB 46) 的基本组态 (页码 117)』一节中所述分配了 SFB 的基本参数。
- 未发生外部错误 ERR。您必须使用 ERR\_A (正跳沿) 来确认排队的外部错误。
- 启动使能 ST\_ENBLD = TRUE。
- 轴是同步的 (SYNC = TRUE)。

### 指定目标

指定目标时，请注意以下事项：

- 行程必须大于等于关断差程。
- 如果行程小于等于目标范围的一半，将不会启动新运行。该模式将立即终止且不产生错误。
- 线性轴的目标范围必须处于工作范围之内，旋转轴必须位于“0”和“旋转轴终点”-1 之间。

### 运行启动

- 线性轴始终在 START = TRUE 时启动。
- 必须指定旋转轴方向的意义：
  - DIR\_P = TRUE：正方向运行
  - DIR\_M = TRUE：负方向运行
  - START = TRUE：轴沿可能的最短距离逼近目标

CPU 计算方向意义时会考虑实际剩余行程、实际值和目标。

如果最短距离小于等于关断差程且大于等于目标范围的一半，则将反向启动运行。

如果在两个方向上的行程相等，则轴将在正方向上移动。

## 步骤

- 按“设置”列中所指定的内容分配 SFB 的以下输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
DRV_EN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE	TRUE
START	BOOL	4.1	运行启动（正跳沿）	TRUE/FALSE	FALSE	START 或 DIR_P 或 DIR_M = TRUE
DIR_P	BOOL	4.2	正方向运行（正跳沿）	TRUE/FALSE	FALSE	
DIR_M	BOOL	4.3	负方向运行（正跳沿）	TRUE/FALSE	FALSE	
MODE_IN	INT	6	操作模式，5 = “绝对增量逼近”	0、1、3、4、5	1	5
TARGET	DINT	8	以脉冲表示的目标	线性轴： -5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲 旋转轴： 0 到旋转轴终点 - 1	1000	xxxx

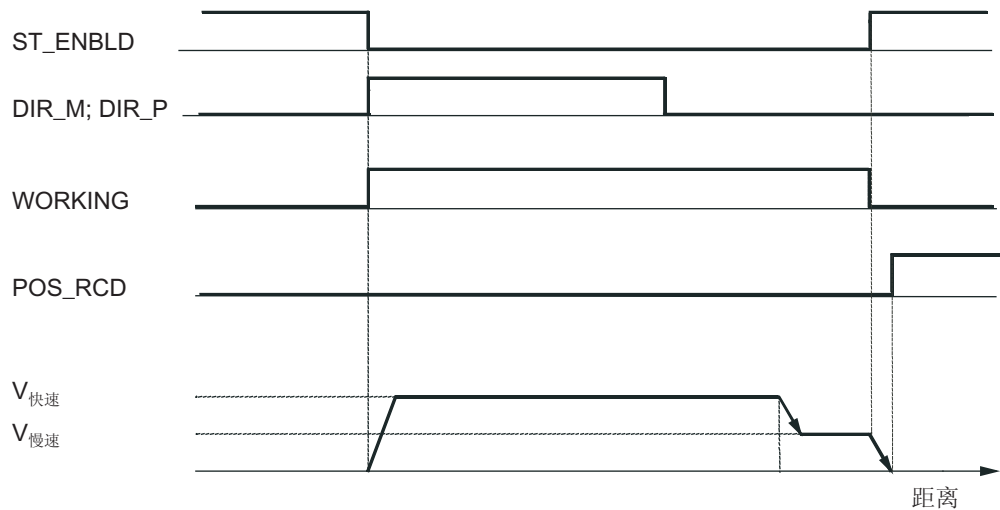
- 调用 SFB。

SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
WORKING	BOOL	14.0	正在运行	TRUE/FALSE	FALSE
POS_RCD	BOOL	14.1	已到位	TRUE/FALSE	FALSE
ACT_POS	DINT	16	当前实际位置值	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
MODE_OUT	INT	20	激活/设置操作模式	0、1、3、4、5	0

## 结果

- 启动运行后将立即置位 `WORKING = TRUE`。在关断点处将 `WORKING` 复位为 `FALSE`。达到指定目标后，将置位 `POS_RCD = TRUE`。
- 在启动下一个运行前，必须复位方向位 (`DIR_P` 或 `DIR_M`)。
- 如果在解释 SFB 调用时发生错误，则 `WORKING = FALSE` 且 `ERROR` 将被置位为 `TRUE`。然后将使用 `STATUS` 参数指出确切的出错原因（请参阅『错误列表（页码 151）』一节）。



## 停止运行且目标范围未达到

使用 `STOP = TRUE` 停止运行时，如果尚未达到关断范围（剩余行程大于关断差程），则依据后续操作模式/作业的不同，有以下几种选项。

选项	响应
启动新的“绝对增量逼近”	轴移动到指定的绝对目标。
以“相对增量逼近”模式在同一方向继续运行	将不解释运行参数。轴移动到被停止的运行的目标点 ( <code>LAST_TRG</code> )。
以“相对增量逼近”模式在相反方向继续运行	将不解释运行参数。轴移动到被停止的运行的起始点。
作业“删除剩余行程”	将删除剩余行程（目标值与实际值之差）。在启动“相对增量逼近”时再次解释运行参数，然后轴将移动到当前实际位置值。

### 4.4.7 指定参考点

#### 说明

也可使用“设置参考点”请求来同步轴，而不执行参考点逼近。

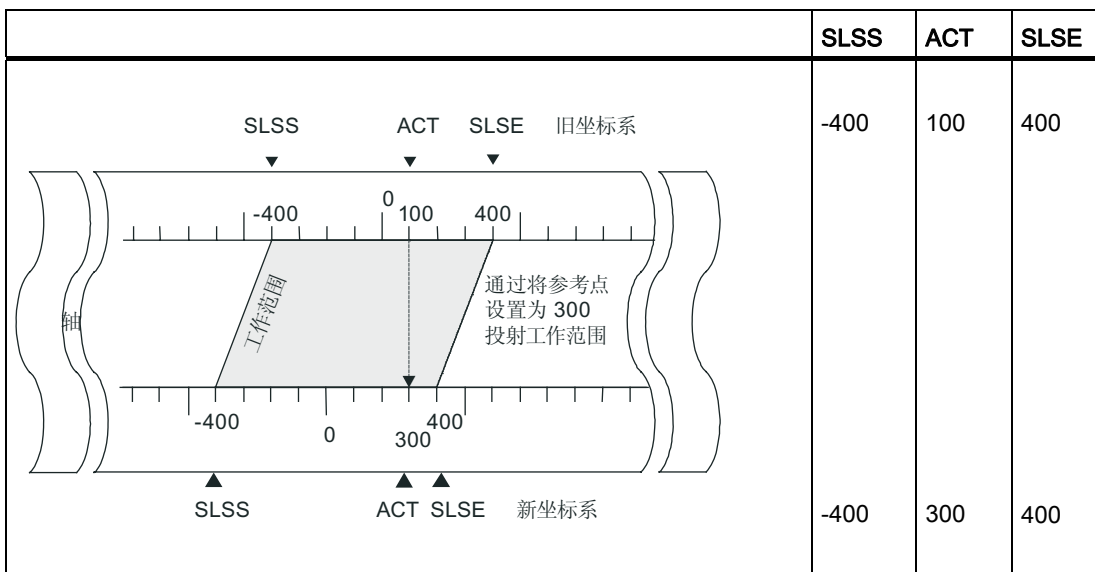
作业执行完毕后，实际位置的坐标值为您通过 JOB\_VAL 参数所分配的值。

- 线性轴：参考点坐标必须位于工作范围（包含软件限位开关）内。
- 旋转轴：参考点坐标必须位于 0 到“旋转轴终点” - 1 范围内。

这不会更改在参数分配窗口中所输入的参考点坐标。

#### 设置参考点的实例

- 实际位置值是 100。软件限位开关 (SLSS,SLSE) 位于位置 -400 和 400（工作范围）处。
- 使用数值 JOB\_VAL = 300 来执行“设置参考点”请求后。
- 这样，实际值为坐标 300。软件限位开关和工作范围的坐标与作业之前的坐标相同。不过，此时它们实际上向左移动了 200。



## 要求

- 您已通过参数分配窗口分配模块参数并将它们下载到了 CPU (PARAM = TRUE)。
- 您已按照『SFB DIGITAL (SFB 46) 的基本组态 (页码 117)』一节中所述分配了 SFB 的基本参数。
- 最后一个作业必须完成 (JOB\_DONE = TRUE)。
- 最后一个定位操作必须完成 (WORKING = FALSE)。

## 步骤

- 按“设置”列中所指定的内容分配以下**输入参数**（通过背景数据块访问）：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
JOB_REQ	BOOL	66.0	作业请求（正跳沿）	TRUE/FALSE	FALSE	TRUE
JOB_ID	INT	68	作业，1 = “设置参考点”	1, 2	0	1
JOB_VAL	DINT	72	参考点坐标的作业参数	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0	xxxx

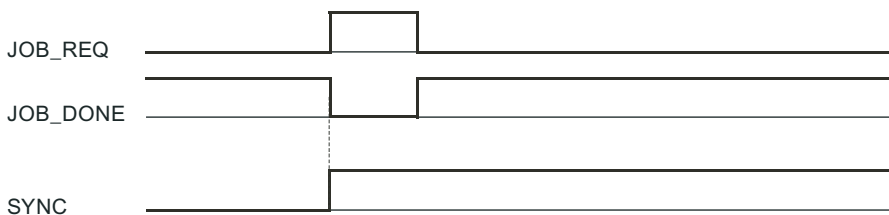
- 调用 SFB。

SFB 的**输出参数**（可通过背景数据块访问的 JOB\_DONE、JOB\_ERR、JOB\_STAT）提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
SYNC	BOOL	14.3	轴同步	TRUE/FALSE	FALSE
JOB_DONE	BOOL	66.1	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	BOOL	66.2	错误作业	TRUE/FALSE	FALSE
JOB_STAT	WORD	70	作业错误编号 （请参阅『错误列表（页码 151）』一节）	0 到 FFFF（十六进制）	0

### 结果

- 调用 SFB 之后将立即处理作业。在一个 SFB 循环时间内 JOB\_DONE 被置位为 FALSE。
- 必须复位作业请求 (JOB\_REQ)。
- 如果处理作业时未出现错误，则 SYNC = TRUE。
- 如果出现错误，则 JOB\_ERR = TRUE。随后将在 JOB\_STAT 中指出确切的出错原因。
- 可以通过 JOB\_DONE = TRUE 来启动新作业。



### 请求结果

- 实际位置值将被设置成参考点坐标值，并将置位状态信号 SYNC。
- 工作范围将沿着轴物理移动。
- 位于工作范围内的所有点将保持其各自的原始坐标，但具有新的物理位置。

### 同时调用作业和定位操作

同时启动定位操作和作业时，将先执行作业。如果作业已结束且出现错误，则不执行定位。如果在执行某个运行时启动了一个作业，则该作业将结束，且出现错误。



### 4.4.8 删除剩余行程

#### 说明

在目标运行（绝对或相对增量逼近）后，可删除作业中未完成的剩余行程 (DIST\_TO\_GO)。

#### 要求

- 您已通过参数分配窗口分配模块参数并将它们下载到了 CPU (PARAM = TRUE)。
- 您已按照『SFB DIGITAL (SFB 46) 的基本组态 (页码 117)』一节中所述分配了 SFB 的基本参数。
- 最后一个作业必须完成 (JOB\_DONE = TRUE)。
- 最后一个定位操作必须完成 (WORKING = FALSE)。

#### 步骤

- 按“设置”列中所指定的内容分配以下 SFB 输入参数（通过背景数据块访问）：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省	设置
JOB_REQ	BOOL	66.0	作业请求（正跳沿）	TRUE/FALSE	FALSE	TRUE
JOB_ID	INT	68	作业，2 = “删除剩余行程”	1, 2	0	2
JOB_VAL	DINT	72	无	-	0	任意

- 调用 SFB。

SFB（可通过背景数据块访问）的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_DONE	BOOL	66.1	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	BOOL	66.2	错误作业	TRUE/FALSE	FALSE
JOB_STAT	WORD	70	作业错误编号 （请参阅『错误列表（页码 151）』一节）	0 到 FFFF（十六进制）	0

#### 结果

- 调用 SFB 之后将立即处理作业。  
在一个 SFB 循环时间内 JOB\_DONE 被置位为 FALSE。
- 必须复位作业请求 (JOB\_REQ)。
- 如果出现错误，则 JOB\_ERR = TRUE。然后将在 JOB\_STAT 中指出确切的出错原因。
- 可以通过 JOB\_DONE = TRUE 来启动新作业。

#### 同时调用作业和定位操作

同时启动定位操作和作业时，将先执行作业。如果作业已结束且出现错误，则不执行定位。

如果正在执行某个运行时启动了一个作业，则该作业将完成，且出现错误。

### 4.4.9 长度测量

#### 说明

可通过“长度测量”来确定工件长度。在数字输入“长度测量”处通过边沿触发长度测量的启动和停止。

在 SFB 中给出了长度测量的启动和结束坐标以及测量的长度。

可通过参数分配窗口（参数“长度测量”）来开启和关闭长度测量，也可确定边沿的类型：

- 关
- 在正跳沿处启动/结束
- 在负跳沿处启动/结束
- 通过正跳沿启动，通过负跳沿结束
- 通过负跳沿启动，通过正跳沿结束

#### 要求

- 您已通过参数分配窗口分配模块参数并将它们下载到了 CPU (PARA = TRUE)。
- 您已按照『SFB DIGITAL (SFB 46) 的基本组态 (页码 117)』一节中所述分配了 SFB 的基本参数。
- 已将无反跳开关连接到了数字输入“长度测量”（连接器 X2，针脚 5）。
- 同步 (SYNC = TRUE) 和非同步 (SYNC = FALSE) 轴均可实现“长度测量”。

#### 步骤

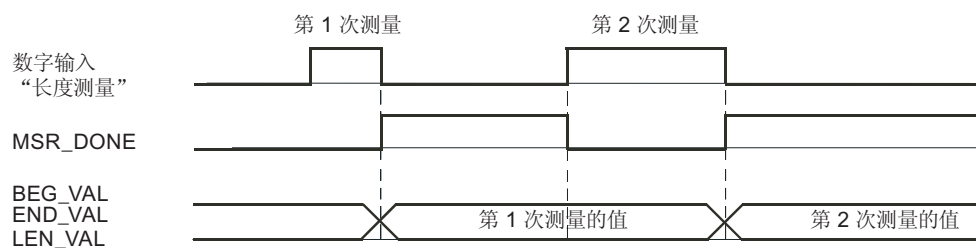
- 数字输入的信号边沿启动长度测量。
- 启动长度测量时将复位 MSR\_DONE。
- 长度测量结束时，将置位 MSR\_DONE = TRUE。
- 然后 SFB 将输出下列值：
  - 启动长度测量：BEG\_VAL
  - 结束长度测量：END\_VAL
  - 测量的长度：LEN\_VAL

从一个长度测量结束到下一个长度测量结束，这些值在块中可用。

SFB（可通过背景数据块访问 BEG\_VAL、END\_VAL、LEN\_VAL）的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
MSR_DONE	BOOL	14.2	结束长度测量	TRUE/FALSE	FALSE
BEG_VAL	DINT	54	实际位置值，启动长度测量	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
END_VAL	DINT	58	实际位置值，结束长度测量	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
LEN_VAL	DINT	62	测量长度	0 到 10 <sup>9</sup> 个脉冲	0

下图显示了某个长度测量的信号曲线，该长度测量的类型为：在正/负跳沿启动/结束长度测量。



#### 说明

在长度测量期间参考时，将以如下方式将实际值更改考虑在内：


实例：在距离为 100 个脉冲的两个点之间执行长度测量。在长度测量期间参考时，坐标将移动 + 20。从而会导致测量的长度为 120。

## 4.5 调整参数

### 4.5.1 重要安全规则

#### 重要注意事项

请注意以下警告内容：

 <b>警告</b>
可能会造成人身伤害或财产损失。 为避免伤害和财产损坏，请注意以下内容： <ul style="list-style-type: none"><li>• 在控制系统的区域内安装<b>紧急停车开关</b>。这是可确保当控制系统发生故障时能够安全关闭系统的唯一可能方法。</li><li>• 安装<b>硬件限位开关</b>，它可直接影响所有驱动器的驱动器转换器。</li><li>• 确保<b>没有人进入系统区域</b>，该区域中存在运动部件。</li><li>• 通过程序和 STEP7 接口进行<b>并行控制和监视</b>可能会导致冲突，从而会造成不确定的影响。</li></ul>

### 4.5.2 确定模块参数及其作用

#### 编码器每转增量

在其铭牌或规格说明书上可以找到所连接的增量编码器的“**编码器每转增量**”参数。该技术以四倍模式判断编码器信号。四个脉冲表示一个编码器增量。所有距离规范都以脉冲为单位。

#### 控制模式

“**控制模式**”参数描述了控制驱动器的四个数字输出信号。必须根据物理控制电路（断路器）指定此参数。控制模式的描述可以在『驱动器参数（页码 102）』一节中找到。

#### 监视时间

必须在“**监视时间**”参数中选择一段足够长的时间，以确保驱动器在指定时间内可代替轴的启动保持转距。

也可使用此监视时间来监视目标逼近。即，驱动器在达到关断点后，必须在此时间内达到目标范围。

#### 计数方向

使用“**计数方向**”参数可将路径监视的方向调整为线性轴的运动方向。还应考虑所有传动元件（例如，联接器和齿轮）的旋转方向。

- “**常规**”表示计数脉冲数的增加将对应着实际位置值的增加。
- “**反转**”表示计数脉冲数的增加将对应着实际位置值的减小。

### 4.5.3 SFB 参数的作用

#### CHGDIFF\_P 和 CHGDIFF\_M

参数“**CHGDIFF\_P**”（转换差程，正方向）和“**CHGDIFF\_M**”（转换差程，负方向）定义驱动器由快行速度转换为爬行速度时的位置。

如果差值设置得过大，则不会随时间对定位进行优化，因为这样会不必要地延长了爬行速度运行的时间。

#### CUTOFFDIFF\_P 和 CUTOFFDIFF\_M

参数 **CUTOFFDIFF\_P**（正方向的关断差程）和 **CUTOFFDIFF\_M**（负方向的关断差程）指定在目标逼近时关闭驱动器之前要行进的脉冲数。

请注意，该距离将随着驱动器上负载的不同而有所不同。

如果转换/关断差程过小，则驱动器会在速度高于爬行速度时关闭。结果会导致定位不准确。

各个方向的转换/关断差程的差应至少与驱动器为达到爬行速度而实际需要的距离成比例。此时还必须考虑驱动器上的负载。

## 4.5.4 检查监视时间

## 要求

- 系统接线正确。
- 已组态了定位子模块，已分配了参数且已载入了项目。
- 例如，已载入了所提供的示例程序“数字 1，入门指南”。
- CPU 处于 RUN 状态

## 核对清单

步骤	要执行的操作	✓
1	<b>校验接线</b>	
	<ul style="list-style-type: none"> <li>• 校验输出接线是否正确。</li> <li>• 校验编码器输入接线是否正确。</li> </ul>	<input type="checkbox"/> <input type="checkbox"/>
2	<ul style="list-style-type: none"> <li>• <b>检查轴运动</b></li> <li>• 在正方向或负方向上爬行速度“点动”驱动器。 方向 DIR 的实际意义必须与指定方向相符。 如果事实并非如此，请更改模块参数“计数方向”。</li> </ul>	<input type="checkbox"/>
3	<ul style="list-style-type: none"> <li>• <b>同步轴</b></li> <li>• 选择作业“设置参考”(JOB_ID = 1)。 通过 JOB_VAL 输入所需要的实际轴位置的坐标（例如 0 脉冲）。 通过将 JOB_REQ 置位为 TRUE 来执行同步。 指定的坐标将显示为实际位置值，并置位同步位 SYNC。 判断 (JOB_STAT) 报告的错误 (JOB_ERR = TRUE)。 如果需要，请修正指定的坐标并重复用于设置参考的作业。</li> </ul>	<input type="checkbox"/>
4	<ul style="list-style-type: none"> <li>• <b>检查转换/关断差程</b></li> <li>• 爬行速度向指定目标 (TARGET) 执行“相对或绝对增量逼近”运行，该目标与当前位置的距离要比在转换差程中指定的距离远。</li> <li>• 请注意各个定位阶段（加速、恒速运行、减速、目标逼近）。 增加转换差程，直到发现驱动器爬行速度向关断位置移动为止。 如果未达到组态的目标范围，请减小转换差程并重复该运行直到达到目标范围为止。</li> <li>• 现在优化转换差程。 在不更改关断差程的情况下减小转换差程，并重复该运行。 可将转换差程减小为某个值（在该值下，您几乎注意不到驱动器的爬行速度移动），即驱动器在达到关断点时必然爬行速度移动。 只要驱动器在爬行速度时关闭，定位精度就会保持不变。 进一步减小关断差程没有任何意义。</li> </ul>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

## 4.6 错误处理和中断

### 4.6.1 系统功能块 (SFB) 中的错误消息

#### 概述

SFB 指出了下表中列出的错误。

除系统错误外的所有错误都将通过错误编号更详细地指出，错误编号可作为 SFB 中的输出参数。

错误类型	通过 SFB 参数显示错误	在 SFB 参数中显示错误编号
操作模式错误	ERROR = TRUE	STATUS
作业错误	JOB_ERR = TRUE	JOB_STAT
外部错误	ERR > 0	ERR
系统错误	BIE = FALSE	-

#### 操作模式错误 (ERROR = TRUE)

此类错误出现在

- SFB 中的常规参数分配错误（例如使用了错误的 SFB）时。
- 运行启动/恢复时。这些错误会在解释操作模式参数期间出现。

检测到错误时，输出参数 ERROR 将被置位为 TRUE。

参数 STATUS 指明出错原因。可能的错误编号可在『错误列表（页码 151）』一节中找到。

#### 作业错误 (JOB\_ERR = TRUE)

只有在解释/执行作业期间会出现作业错误。

检测到错误时，输出参数 JOB\_ERR 将被置位为 TRUE。

出错原因会在 JOB\_STAT 参数中指出。可能的错误编号可在『错误列表（页码 151）』一节中找到。

### 外部错误 (ERR)

系统将监视运行、行程范围和连接的 I/O。先决条件是需要在“驱动器”、“轴”和“编码器”参数分配窗口中启用监视功能。

监视功能响应时，将发送外部错误信号。

外部错误可能随时出现，与启动的功能无关。

您必须通过 ERR\_A（正跳沿）来确认外部错误。

通过在 SFB 参数 ERR (WORD) 中置位一个位来指示外部错误。

监视	ERR	ERR-WORD 中的位
缺少脉冲（零标记）	0004（十六进制）	2
行程范围	0800（十六进制）	11
工作范围	1000（十六进制）	12
实际值	2000（十六进制）	13
目标逼近	4000（十六进制）	14
目标范围	8000（十六进制）	15

检测到外部错误（“进入”和“离开”）还会触发诊断中断（请参阅『组态和判断诊断中断（页码 146）』一节）。

### 系统错误

用 BIE = FALSE 来指示系统错误。

下列情况下会触发系统错误：

- 背景数据块的读/写访问错误
- 多重调用 SFB

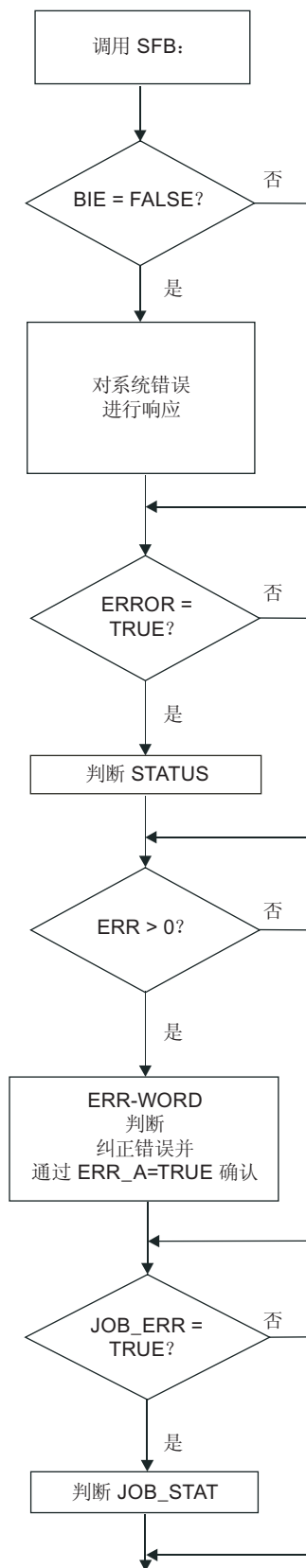
## 4.6.2 在用户程序中判断出错

### 步骤

1. 调用错误处理例程“错误判断”（参见下图）。
2. 按顺序查询特定的错误类型。
3. 如果需要，可跳转到特别适合您的应用的错误响应方法。



错误判断:



### 4.6.3 组态和判断诊断中断

#### 基本信息

出现以下错误时，可以触发诊断中断：

- 参数分配错误（模块数据）
- 外部错误（监视）

在出现进入错误和离开错误事件时，将会显示诊断中断。

在用户程序中，可借助诊断中断立即对错误作出响应。

#### 步骤

1. 在参数分配窗口的“基本参数”对话框中启用诊断中断。
2. 在“驱动器”、“轴”和“编码器”参数分配窗口中，分别打开出错时会触发诊断中断的监视功能。
3. 在参数分配窗口“诊断”中，分别为每个监视对象启用诊断中断。
4. 可将诊断中断 OB (OB 82) 合并到用户程序中。

#### 通过诊断中断对错误作出响应

- 定位被取消。
- CPU 操作系统将调用用户程序中的 OB 82。

---

#### 说明

如果未装载相应的 OB，则在触发中断后，CPU 会切换为 STOP。

---

- CPU 将打开 SF LED 指示灯。
- 在 CPU 的诊断缓冲区中将错误报告为“进入”。清除所有未决错误前，不会将错误指示为“离开”。

## 如何在用户程序中判断诊断中断

触发诊断中断后，可判断 OB 82 以检查哪个诊断中断还未处理。

- 如果在 OB 82 的字节 6 + 7 (OB 82\_MDL\_ADDR) 中输入了“定位”子模块的模块地址，则诊断中断是由 CPU 的定位功能触发的。
- 只要队列中有错误，就会置位 OB 82 中字节 8 的位 0（故障模块）。
- 所有错误都已报告“离开”后，将复位 OB 82 中字节 8 的位 0。
- 通过判断数据记录 1（字节 8 和 9），可以确定确切的出错原因。为此，必须调用 SFC 59（读取数据记录）。
- 用 ERR\_A 确认错误。

数据记录 1, 字节 8	说明	JOB_STAT	ERR
位 0	没有使用	-	-
位 1	没有使用	-	-
位 2	缺少脉冲 *	-	X
位 3	没有使用	-	-
位 4	没有使用	-	-
位 5	没有使用	-	-
位 6	没有使用	-	-
位 7	没有使用	-	-

数据记录 1, 字节 9	说明	JOB_STAT	ERR
位 0	组态错误	X	-
位 1	没有使用	-	-
位 2	没有使用	-	-
位 3	行程范围监视	X	X
位 4	工作范围监视	X	X
位 5	实际值监视 *	X	X
位 6	目标逼近监视 *	X	X
位 7	目标范围监视 *	X	X

\*: 随后出现的错误将触发一个进入中断，然后自动触发离开中断。

## 4.7 安装实例

### 使用实例

可在文档附带的 CD-ROM 中找到实例（程序和说明）。也可从 Internet 下载。项目由具有不同复杂度和针对性的多个带注释的 S7 程序组成。

CD 上的 Readme.wri 说明了安装实例的方法。安装后，实例即存储在目录 `...\STEP7\EXAMPLES\ZDI26_03_TF____31xC_Pos` 中。

## 4.8 规范

### 4.8.1 增量编码器

#### 可连接的增量编码器

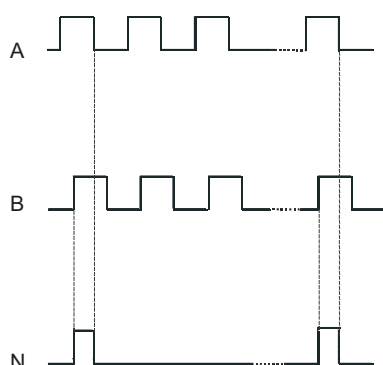
支持非对称 24 V 增量编码器，这些编码器具有两个电相差为 90° 的脉冲轨迹（有/无零标记）。

用于编码器连接的输入	最小脉冲宽度/最小脉冲间宽度	最大输入频率	最大电缆长度（最大输入频率）
编码器信号 A, B	8 μs	60 kHz	50 m
编码器信号 N （零标记信号）	8 μs	60 kHz / 30 kHz <sup>1</sup>	50 m

1: 如果使用的编码器的零标记信号通过“AND”运算与编码器信号 A 和 B 合并，脉冲宽度将减半至周期时间的 25%。为了维持最小脉冲宽度，必须将最大计数频率降低到 30 kHz。

#### 信号判断

下图显示具有非对称输出信号的编码器的信号曲线：



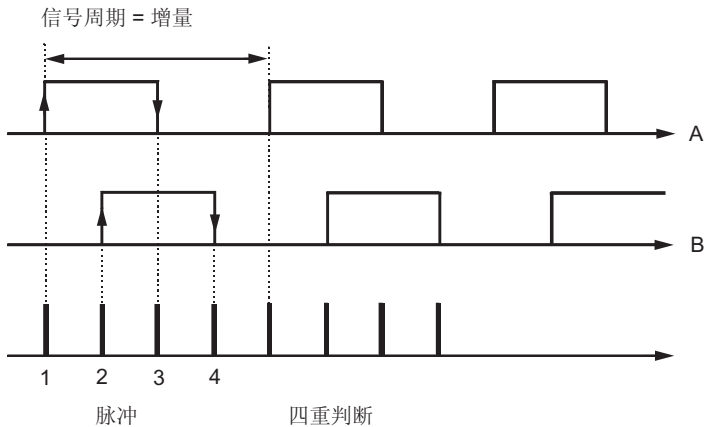
CPU 内部生成零标记信号与 A 和 B 轨迹信号的逻辑 AND 链接。

为了参照，CPU 在零标记处使用正跳沿。

如果信号 A 转换超前信号 B，则 CPU 正方向计数。

### 增量

增量标识两个编码器轨迹信号 A 和 B 的信号周期。该值在编码器的铭牌和/或技术规范中指定。

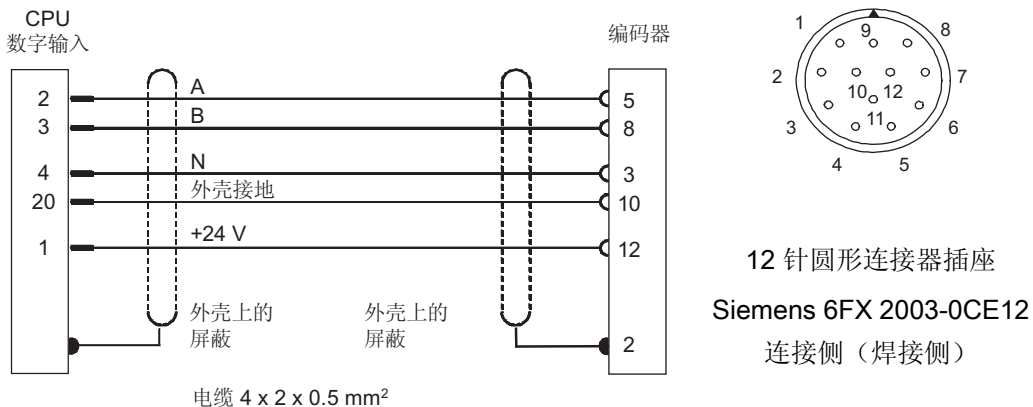


### 脉冲

CPU 使用每个增量（四重判断）计算轨迹信号 A 和 B 的全部 4 个跳沿（参见上图）。也就是说，一个编码器增量对应四个脉冲。

### 增量编码器 Siemens 6FX 2001-4 (Up = 24 V;HTL) 的接线图

下图显示了增量编码器 Siemens 6FX 2001-4xxxx (Up = 24 V; HTL) 的接线图：



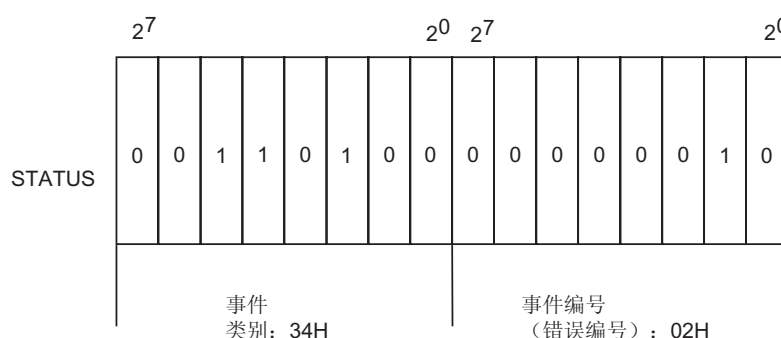
## 4.8.2 错误列表

### 基本信息

如果出现错误，会在 SFB 参数 STATUS 或 JOB\_STAT 中输出错误 ID。错误编号由事件类别和事件编号组成。

### 实例

下图显示了事件“指定了错误目标”（事件类别：34H，事件编号：02H）对应的 STATUS 参数内容。



### SFB 参数“STATUS”的错误编号

事件类别 32 (20H): “SFB 错误”

事件编号	事件文本	补救措施
(20)01H	SFB 不正确	使用 SFB 46
(20)04H	通道号 (CHANNEL) 不正确	设置通道号“0”

事件类别 48 (30H): “常规运行启动错误”

事件编号	事件文本	补救措施
(30)01H	运行作业被拒绝，因为同一 SFB 调用中存在错误作业	更正相应 JOB 参数
(30)02H	驱动器仍在运行时不允许修改 MODE_IN	等待，直至完成当前的定位操作
(30)03H	未知操作模式 (MODE_IN)	允许的模式有：1（点动模式）、3（参考点逼近）、4（相对增量逼近）和 5（绝对增量逼近）
(30)04H	一次只能置位一个启动请求	允许的启动请求为 DIR_P、DIR_M 或 START
(30)05H	仅在“绝对增量逼近”模式下允许 START	使用 DIR_P 或 DIR_M 启动运行

事件类别 48 (30H): “常规运行启动错误”

事件编号	事件文本	补救措施
(30)06H	线性轴和“绝对增量逼近”模式不允许 DIR_P 或 DIR_M	使用 START 启动运行。
(30)07H	轴未同步	仅当轴同步时才能使用“绝对增量逼近”。
(30)08H	移出工作范围	只允许向进入工作范围的方向运行。

事件类别 49 (31H): “运行启动错误（启用启动）”

事件编号	事件	补救措施
(31)01H	未启用启动，因为未组态轴。	通过 HW Config 组态“定位”子模块。
(31)02H	未启用启动，因为未置位驱动器使能。	在 SFB (DRV_EN = TRUE) 中置位“启动使能”。
(31)03H	启动未使能，因为置位了 STOP。	在 SFB (STOP = FALSE) 中清除 STOP。
(31)04H	启动未使能，因轴当前在执行定位运行 (WORKING = TRUE)。	等待，直至完成当前的定位操作。
(31)05H	启动未使能，因为至少有一个未决错误未得到确认。	首先，排除并确认所有外部错误，然后重新启动运行。

事件类别 50 (32H): “运行启动错误（速度/加速度）”

事件编号	事件	补救措施
(32)01H	速度规范 SPEED 不正确	通过数字输出定位时仅允许“爬行速度”(0)和“快速往返”(1)

事件类别 51 (33H): “运行启动错误（转换/关断差程）”

事件编号	事件	补救措施
(33)01H	转换/关断差程不允许大于 $10^8$	指定最大值为 $10^8$ 的转换/关断差程。
(33)03H	转换差程不允许小于关断差程	转换差程必须大于/等于关断差程。
(33)04H	关断差程过短	关断差程必须至少为目标范围长度的一半。



事件类别 52 (34H): “运行启动错误 (缺省目标/距离)”

事件编号	事件	补救措施
(34)01H	缺省目标超出工作范围	对于线性轴和绝对增量逼近, 缺省目标必须位于软件限位开关范围内 (含此范围)。
(34)02H	目标规范不正确	对于旋转轴, 指定目标必须大于 0 且小于旋转轴终点值。
(34)03H	距离规范不正确	对于相对增量逼近, 行程必须为正值。
(34)04H	距离规范不正确	结果 (即绝对目标坐标) 必须大于 $-5 \times 10^8$ 。
(34)05H	距离规范不正确	作为结果的绝对目标坐标必须小于 $5 \times 10^8$ 。
(34)06H	距离规范不正确	作为结果的绝对目标坐标必须位于工作范围内 (+/- 目标范围的一半)。

事件类别 53 (35H): “运行启动错误 (行程)”

事件编号	事件	补救措施
(35)01H	行程过长	目标坐标 + 实际剩余行程必须大于等于 $-5 \times 10^8$
(35)02H	行程过长	目标坐标 + 实际剩余行程必须小于等于 $5 \times 10^8$
(35)03H	行程过短	正方向的行程必须大于正方向的指定关断差程
(35)04H	行程过短	负方向的行程必须大于负方向的指定关断差程
(35)05H	行程过短或已在正方向上超出限位开关	正方向的上一可逼近目标 (工作范围/行程范围限制) 距离实际位置太近
(35)06H	行程过短或已在负方向上超出限位开关	负方向的上一可逼近目标 (工作范围/行程范围限制) 距离实际位置太近

## SFB 参数“JOB\_STAT”的错误编号

事件类别 64 (40H): “常规作业执行错误”

事件编号	事件	补救措施
(40)01H	轴未组态	通过 HW Config 组态“定位”子模块。
(40)02H	无法执行作业，因为定位仍在运行	仅当没有处于活动状态的定位运行时，才能执行作业。等待，直到 WORKING = FALSE，然后重复该作业。
(40)04H	未知作业	检查作业 ID 并重复该作业。

事件类别 65 (41H): “执行设置参考点请求时出错”

事件编号	事件	补救措施
(41)01H	参考点坐标超出工作范围	对于线性轴，参考点坐标不得超出工作范围限制。
(41)02H	参考点坐标不正确	对于线性轴，指定的参考点坐标 + 实际剩余行程必须仍大于等于 $-5 \times 10^8$ 。
(41)03H	参考点坐标不正确	对于线性轴，指定的参考点坐标 + 实际剩余行程必须仍小于等于 $5 \times 10^8$ 。
(41)04H	参考点坐标不正确	对于线性轴，指定的参考点坐标 + 距运行起始点的实际距离必须仍大于等于 $-5 \times 10^8$ 。
(41)05H	参考点坐标不正确	对于线性轴，指定的参考点坐标 + 距运行起始点的实际距离必须仍小于等于 $5 \times 10^8$ 。
(41)06H	参考点坐标超出旋转轴范围	对于旋转轴，参考点坐标不得小于 0 和大于等于旋转轴终点的值。

## 外部错误 (ERR)

通过在 SFB 参数 ERR (WORD) 中置位一个位来指示外部错误。

监视	ERR	ERR-WORD 中的位
缺少脉冲 (零标记)	0004 (十六进制)	2
行程范围	0800 (十六进制)	11
工作范围	1000 (十六进制)	12
实际值	2000 (十六进制)	13
目标逼近	4000 (十六进制)	14
目标范围	8000 (十六进制)	15

### 4.8.3 参数分配窗口的模块参数 - 概述

#### 引言

下表概述了可在参数分配窗口中设置的模块参数。

#### 基本参数

参数	值范围	缺省
中断选择	<ul style="list-style-type: none"> <li>• 无</li> <li>• 诊断</li> </ul>	无

#### 驱动器参数

表格 4-2 控制模式参数

参数	值范围	缺省
控制模式	1-4	1

表格 4-3 控制模式的含义

输出	控制模式			
	1	2	3	4
Q0	快速往返	快速往返/爬行速度	快速往返	正方向快速往返
Q1	爬行速度	已到位	爬行速度	正方向爬行速度
Q2	正方向运行	正方向运行	正方向运行	负方向快速往返
Q3	负方向运行	负方向移动	负方向移动	负方向爬行速度

表格 4-4 其它驱动器参数

参数	值范围	缺省
目标范围	0 到 200 000 000 个脉冲，且包括 200 000 000 CPU 将向上舍入奇数值。	50
监视时间	<ul style="list-style-type: none"> <li>• 0 到 100 000 ms</li> <li>• 0 = 无监视</li> </ul> 舍入操作由 CPU 在 4 ms 内完成。	2000
实际值监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	是
目标逼近监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
目标范围监视	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
最大频率：位置反馈	60、30、10、5、2、1 kHz	60 kHz
最大频率：伴随信号	60、30、10、5、2、1 kHz	10 kHz

## 轴参数

参数	值范围	缺省
轴类型	<ul style="list-style-type: none"> <li>线性轴</li> <li>旋转轴</li> </ul>	线性轴
软件限位开关开始/结束	软件限位开关开始 软件限位开关结束 -5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	-100 000 000 +100 000 000
旋转轴终点	1 到 10 <sup>9</sup> 个脉冲	100 000
长度测量	<ul style="list-style-type: none"> <li>关</li> <li>在正跳沿 DI 处启动/结束</li> <li>在负跳沿 DI 处启动/结束</li> <li>通过正跳沿启动, 通过负跳沿结束</li> <li>通过负跳沿启动, 通过正跳沿结束</li> </ul>	关
参考点坐标	-5x10 <sup>8</sup> 到 +5x10 <sup>8</sup> 个脉冲	0
参考点开关的参考点位置	<ul style="list-style-type: none"> <li>正方向 (实际值增加)</li> <li>负方向 (实际值减少)</li> </ul>	正方向
监视行程范围	是 (设置成固定值)	是
监视工作范围	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	是

## 编码器参数

参数	值范围	缺省
编码器每转增量	1 到 2 <sup>23</sup> 个脉冲	1000
计数方向	<ul style="list-style-type: none"> <li>常规</li> <li>反转</li> </ul>	常规
缺少脉冲 (零标记) 监视	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否

## 诊断参数

参数	值范围	缺省
缺少脉冲 (零标记)	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
行程范围	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
工作范围 (对于线性轴)	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
实际值	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
目标逼近	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
目标范围	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否

## 4.8.4 SFB DIGITAL (SFB46) 的背景数据块的参数

## 概述

参数	声明	数据类型	地址 (背景数据块)	说明	值范围	缺省
LADDR	IN	WORD	0	在“HW Config”中指定的子模块 I/O 地址。 如果 I/O 地址不相同, 必须指定两者中的较低一个。	CPU 特定	310 (十六进制)
CHANNEL	IN	INT	2	通道号	0	0
DRV_EN	IN	BOOL	4.0	驱动器使能	TRUE/FALSE	FALSE
START	IN	BOOL	4.1	运行启动 (正跳沿)	TRUE/FALSE	FALSE
DIR_P	IN	BOOL	4.2	正方向运行 (正跳沿)	TRUE/FALSE	FALSE
DIR_M	IN	BOOL	4.3	负方向运行 (正跳沿)	TRUE/FALSE	FALSE
STOP	IN	BOOL	4.4	停止运行	TRUE/FALSE	FALSE
ERR_A	IN	BOOL	4.5	组错误确认 ERR_A 用于确认外部错误 (正跳沿)	TRUE/FALSE	FALSE
MODE_IN	IN	INT	6	操作模式	0、1、3、4、5	1
TARGET	IN	DINT	8	相对增量逼近: 以脉冲表示的距离 (仅允许正值)	0 到 $10^9$	1000
				绝对增量逼近: 以脉冲表示的目标	线性轴: $-5 \times 10^8$ 到 $+5 \times 10^8$ 旋转轴: 旋转轴: 0 到旋转轴终点 - 1	
SPEED	BOOL	DINT	12.0	快行/爬行速度两个速度阶段 TRUE = 快行速度 FALSE = 爬行速度	TRUE/FALSE	FALSE
WORKING	OUT	BOOL	14.0	正在运行	TRUE/FALSE	FALSE
POS_RCD	OUT	BOOL	14.1	已到位	TRUE/FALSE	FALSE
MSR_DONE	OUT	BOOL	14.2	结束长度测量	TRUE/FALSE	FALSE
SYNC	OUT	BOOL	14.3	轴同步	TRUE/FALSE	FALSE
ACT_POS	OUT	DINT	16	实际位置值	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0
MODE_OUT	OUT	INT	20	启用/设置操作模式	0、1、3、4、5	0

参数	声明	数据类型	地址 (背景数据块)	说明	值范围	缺省
ERR	OUT	WORD	22	外部错误 位 2: 缺少脉冲监视 位 11: 行程范围监视 (始终为 1) 位 12: 工作范围监视 位 13: 实际值监视 位 14: 目标逼近监视 位 15: 目标范围监视 其它位将保留	每个位 0 或 1	0
ST_ENBLD	OUT	BOOL	24.0	启动使能	TRUE/FALSE	TRUE
ERROR	OUT	BOOL	24.1	运行启动错误/恢复错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	26.0	错误 ID	0 到 FFFF (十六进制)	0
CHGDIFF_P	STAT	DINT	28	正方向转换差程	0 到 $+10^8$ 脉冲	1000
CUTOFF-DIFF_P	STAT	DINT	32	正方向关断差程	0 到 $+10^8$ 脉冲	100
CHGDIFF_M	STAT	DINT	36	负方向转换差程	0 到 $+10^8$ 脉冲	1000
CUTOFF-DIFF_M	STAT	DINT	40	负方向关断差程	0 到 $+10^8$ 脉冲	100
PARA	STAT	BOOL	44.0	轴已组态	TRUE/FALSE	FALSE
DIR	STAT	BOOL	44.1	当前/上一方向的意义 FALSE = 向前 (正方向) TRUE = 反向 (负方向)	TRUE/FALSE	FALSE
CUTOFF	STAT	BOOL	44.2	驱动器位于关断范围内 (从关断位置到下一运行开始)	TRUE/FALSE	FALSE
CHGOVER	STAT	BOOL	44.3	驱动器位于转换范围内 (从达到转换位置到下一运行开始)	TRUE/FALSE	FALSE
DIST_TO_GO	STAT	DINT	46	实际剩余行程	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0
LAST_TRG	STAT	DINT	50	上一/实际目标	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0
BEG_VAL	STAT	DINT	54	实际位置值, 启动长度测量	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0
END_VAL	STAT	DINT	58	实际位置值, 结束长度测量	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0
LEN_VAL	STAT	DINT	62	测量长度	0 到 $10^9$ 个脉冲	0
JOB_REQ	STAT	BOOL	66.0	启动作业 (正跳沿)	TRUE/FALSE	FALSE
JOB_DONE	STAT	BOOL	66.1	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	STAT	BOOL	66.2	错误作业	TRUE/FALSE	FALSE
JOB_ID	STAT	INT	68	作业 ID	1, 2	0
JOB_STAT	STAT	WORD	70	作业错误 ID	0 到 FFFF (十六进制)	0
JOB_VAL	STAT	DINT	72	参考点坐标的作业参数	$-5 \times 10^8$ 到 $+5 \times 10^8$ 个脉冲	0

## 计数、频率测量和脉冲宽度调制

### 5.1 概述

#### 5.1.1 操作模式和特性 - 概述

##### CPU 的操作模式

- 计数
- 频率计数
- 脉冲宽度调制（脉冲串输出）

##### CPU 特性概述

- 通道数
  - CPU 312C: 2 个通道
  - CPU 313C、CPU 313C-2 DP/PtP: 3 个通道
  - CPU 314C-2 DP/PtP: 4 个通道

---

##### 说明

使用定位功能时只有 2 个通道（通道 2 和 3）可用。

---

- 计数频率
  - CPU 312C - 10 kHz（最大）
  - CPU 313C、CPU 313C-2 DP/PtP - 30 kHz（最大）
  - CPU 314C-2 DP/PtP - 60 kHz（最大）
- 发信号通知 CPU 计数
  - 24 V 增量编码器，带有两个相位偏移 90° 的轨迹（旋转编码器）<sup>1)</sup>
    - 具有方向电平的 24 V 脉冲编码器
    - 24 V 启动器（例如，BERO 或挡光板）
- 组态
  - 通过参数分配窗口

<sup>1)</sup>两个信号的四折判断可产生内部四折计数频率

## 5.1 概述

### 5.1.2 功能范围 - 概述

#### 计数

- 计数模式
  - 连续计数
  - 一次计数
  - 周期性计数
- 门功能  
用于开始、停止和中断计数功能。
- 锁存器功能  
可使用此功能在数字输入出现正跳沿时保存当前的内部计数值。
- 比较器  
可在 CPU 中存储比较值。根据计数值和比较值，可激活数字输出或生成硬件中断。
- 滞后  
可指定数字输出的滞后。这样，当计数值在比较值范围内时，可防止因编码器信号的每次轻微抖动而造成数字输出抖动。
- 硬件中断
- 周期测量  
您可以测量计数信号（最大计数频率为 1 kHz）的周期。

#### 频率计数

- 门功能  
使用门功能可启动/停止频率测量。
- 上/下限  
可指定频率监控的上限和下限。达到其中的一个限值时，可激活数字输入和/或生成硬件中断。
- 硬件中断

#### 脉冲宽度调制 (PWM)

- 门功能  
使用门功能可启动/停止脉冲宽度调制。
- 硬件中断



### 5.1.3 计数器组件应用 - 概述

#### 概述

计数功能（计数、频率计数和脉冲宽度调制）集成在 **CPU** 中。**编码器**，或无反跳开关，提供计数脉冲。

使用 **PG/PC**:

- 使用参数分配窗口为 **CPU** 的技术功能分配 **CPU** 参数。
- 为 **CPU SFB** 编程，这些功能块可直接集成在用户程序中。
- 借助标准 **STEP7** 接口（监视功能和变量表）调试和测试 **CPU**。

5.2 接线

## 5.2 接线


### 5.2.1 接线规则

#### 连接电缆/屏蔽

- 必须屏蔽编码器电缆。
- 连接数字 I/O 的电缆如果长度超过 100m，则必须屏蔽。
- 在电缆屏蔽的两端，必须对屏蔽层进行端接。
- 软线，横截面积为 0.25 到 1.5 mm<sup>2</sup>
- 不需要电缆套。如果坚持使用电缆套，请使用不带绝缘环的电缆套（DIN 46228，A 形，短型）。

#### 屏蔽端接元件

使用此屏蔽端接元件能够轻松完成屏蔽电缆的接地连接，因为屏蔽端接元件直接接触固定轨。

 <b>警告</b>
如果不切断电源，可能会危及人身安全和导致财产损失： 如果连接带电模块的前插头，会有触电危险！ 必须在断电状态下连接模块！

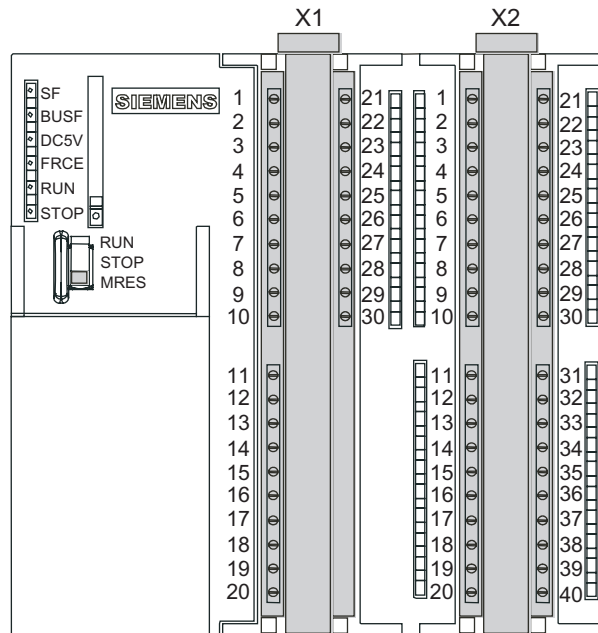
#### 其它信息

有关其它信息，请参见 *CPU 数据手册* 和 *CPU 安装说明*。

## 5.2.2 引脚分配

### 连接器的基本布置

以 CPU 314C-2 DP/PtP 为例，下图显示了有两个连接器（X1 和 X2）的 CPU 的主要插头布置情况：



### 引脚分配

以下引脚分配仅适用于与计数、频率测量和脉冲宽度调制相关的连接器。

#### 说明

因为它们需要相同的 I/O，所以使用定位功能时**不能再使用通道 0 和通道 1**。

#### 警告

进行脉冲宽度调制时，相应的通道输入“轨迹 B/方向”必须保持断开或连接到逻辑“0”。

## CPU 312C 连接器 X1 的引脚分配

连接	名称/地址	计数	频率测量	脉冲宽度调制
1	-	n.c.		
2	DI+0.0	通道 0: 轨迹 A/脉冲	通道 0: 轨迹 A/脉冲	-
3	DI+0.1	通道 0: 轨迹 B/方向	通道 0: 轨迹 B/方向	0/不使用
4	DI+0.2	通道 0: 硬件门	通道 0: 硬件门	通道 0: 硬件门
5	DI+0.3	通道 1: 轨迹 A/脉冲	通道 1: 轨迹 A/脉冲	-
6	DI+0.4	通道 1: 轨迹 B/方向	通道 1: 轨迹 B/方向	0/不使用
7	DI+0.5	通道 1: 硬件门	通道 1: 硬件门	通道 1: 硬件门
8	DI+0.6	通道 0: 锁存器	-	-
9	DI+0.7	通道 1: 锁存器	-	-
10	DI+1.0	-		
11	DI+1.1	-		
12	2 M	外壳接地		
13	1 L+	输出的 24-V 电源		
14	DO+0.0	通道 0: 输出	通道 0: 输出	通道 0: 输出
15	DO+0.1	通道 1: 输出	通道 1: 输出	通道 1: 输出
16	DO+0.2	-		
17	DO+0.3	-		
18	DO+0.4	-		
19	DO+0.5	-		
20	1 M	外壳接地		

## CPU 313C（连接器 X2）或 CPU 313C-2 DP/PtP（连接器 X1）的针脚分配

连接	名称/地址	计数	频率测量	脉冲宽度调制
1	1 L+	输入的 24-V 电源		
2	DI+0.0	通道 0: 轨迹 A/脉冲	通道 0: 轨迹 A/脉冲	-
3	DI+0.1	通道 0: 轨迹 B/方向	通道 0: 轨迹 B/方向	0/不使用
4	DI+0.2	通道 0: 硬件门	通道 0: 硬件门	通道 0: 硬件门
5	DI+0.3	通道 1: 轨迹 A/脉冲	通道 1: 轨迹 A/脉冲	-
6	DI+0.4	通道 1: 轨迹 B/方向	通道 1: 轨迹 B/方向	0/不使用
7	DI+0.5	通道 1: 硬件门	通道 1: 硬件门	通道 1: 硬件门
8	DI+0.6	通道 2: 轨迹 A/脉冲	通道 2: 轨迹 A/脉冲	-
9	DI+0.7	通道 2: 轨迹 B/方向	通道 2: 轨迹 B/方向	0/不使用
10	-	n.c.		
11	-	n.c.		
12	DI+1.0	通道 2: 硬件门	通道 2: 硬件门	通道 2: 硬件门
13	DI+1.1	-	-	-
14	DI+1.2	-	-	-
15	DI+1.3	-	-	-
16	DI+1.4	通道 0: 锁存器	-	-
17	DI+1.5	通道 1: 锁存器	-	-
18	DI+1.6	通道 2: 锁存器	-	-
19	DI+1.7	-	-	-
20	1 M	外壳接地		
21	2 L+	输出的 24-V 电源		
22	DO+0.0	通道 0: 输出	通道 0: 输出	通道 0: 输出
23	DO+0.1	通道 1: 输出	通道 1: 输出	通道 1: 输出
24	DO+0.2	通道 2: 输出	通道 2: 输出	通道 2: 输出
25	DO+0.3	-		
26	DO+0.4	-		
27	DO+0.5	-		
28	DO+0.6	-		
29	DO+0.7	-		
30	2 M	外壳接地		
31	3 L+	输出的 24-V 电源		
32	DO+1.0	-		
33	DO+1.1	-		
34	DO+1.2	-		
35	DO+1.3	-		
36	DO+1.4	-		
37	DO+1.5	-		
38	DO+1.6	-		
39	DO+1.7	-		
40	3 M	外壳接地		

5.2 接线

CPU314C-2 DP/PtP（连接器 X2）的针脚分配

接线端	名称/地址	计数	频率测量	脉冲宽度调制
1	1 L+	输入的 24-V 电源		
2	DI+0.0	通道 0: 轨迹 A/脉冲	通道 0: 轨迹 A/脉冲	-
3	DI+0.1	通道 0: 轨迹 B/方向	通道 0: 轨迹 B/方向	0/不使用
4	DI+0.2	通道 0: 硬件门	通道 0: 硬件门	通道 0: 硬件门
5	DI+0.3	通道 1: 轨迹 A/脉冲	通道 1: 轨迹 A/脉冲	-
6	DI+0.4	通道 1: 轨迹 B/方向	通道 1: 轨迹 B/方向	0/不使用
7	DI+0.5	通道 1: 硬件门	通道 1: 硬件门	通道 1: 硬件门
8	DI+0.6	通道 2: 轨迹 A/脉冲	通道 2: 轨迹 A/脉冲	-
9	DI+0.7	通道 2: 轨迹 B/方向	通道 2: 轨迹 B/方向	0/不使用
10	-	n.c.		
11	-	n.c.		
12	DI+1.0	通道 2: 硬件门	通道 2: 硬件门	通道 2: 硬件门
13	DI+1.1	通道 3: 轨迹 A/脉冲	通道 3: 轨迹 A/脉冲	-
14	DI+1.2	通道 3: 轨迹 B/方向	通道 3: 轨迹 B/方向	0/不使用
15	DI+1.3	通道 3: 硬件门	通道 3: 硬件门	通道 3: 硬件门
16	DI+1.4	通道 0: 锁存器	-	-
17	DI+1.5	通道 1: 锁存器	-	-
18	DI+1.6	通道 2: 锁存器	-	-
19	DI+1.7	通道 3: 锁存器	-	-
20	1 M	外壳接地		
21	2 L+	输出的 24-V 电源		
22	DO+0.0	通道 0: 输出	通道 0: 输出	通道 0: 输出
23	DO+0.1	通道 1: 输出	通道 1: 输出	通道 1: 输出
24	DO+0.2	通道 2: 输出	通道 2: 输出	通道 2: 输出
25	DO+0.3	通道 3: 输出	通道 3: 输出	通道 3: 输出
26	DO+0.4	-		
27	DO+0.5	-		
28	DO+0.6	-		
29	DO+0.7	-		
30	2 M	外壳接地		
31	3 L+	输出的 24-V 电源		
32	DO+1.0	-		
33	DO+1.1	-		
34	DO+1.2	-		
35	DO+1.3	-		
36	DO+1.4	-		
37	DO+1.5	-		
38	DO+1.6	-		
39	DO+1.7	-		
40	3 M	外壳接地		

### 5.2.3 连接组件

#### 步骤

1. 关闭所有组件的电源。
2. 连接输入和输出的电源：
  - CPU 312C:
    - 24 V 连接到 X1, 针脚 13
    - 接地连接到 X1, 针脚 12 和 20
  - CPU 313C-2 DP/PtP:
    - 24 V 连接到 X1, 针脚 1 和 21
    - 接地连接到 X1, 针脚 20 和 30
  - CPU 313C、CPU 314C-2 DP/PtP:
    - 24 V 连接到 X2, 针脚 1 和 21
    - 接地连接到 X2, 针脚 20 和 30
3. 将编码器和开关连接到 24 V 电源。
4. 连接编码器信号和所需的开关。可将无反跳开关（24 V P 开关）或传感器/BERO（2 或 3 线制接近开关）连接到数字输入“硬件门”和“锁存器”上。
5. 剥掉屏蔽电缆的绝缘材料并将电缆屏蔽连接到屏蔽连接元件。为此可使用屏蔽端子元件。

## 5.3 参数组态

### 5.3.1 使用参数分配窗口进行组态

#### 基本信息

可通过分配参数调整计数功能以适应特定应用：

- 通过参数分配窗口分配参数。
- 它们存储在 CPU 系统存储器中。
- 可在 CPU 处于 RUN 模式时使用 SFB 作业接口更改某些参数（请参阅『通过用户程序控制计数器（页码 186）』、『通过用户程序控制频率计数器（页码 204）』或『通过用户程序控制脉冲宽度调制（页码 214）』一节）。

#### 参数分配窗口

可在**参数分配窗口**中分配模块参数：

- 基本参数
- 连续计数、一次计数和周期性计数
- 频率计数
- 脉冲宽度调制

这些参数分配窗口大都是自说明的。可以在以下各节和参数分配窗口的集成帮助中找到参数说明。

---

#### 说明

如果使用通道 0 或通道 1，则不能再使用“定位”技术。

---

#### 要求

调用参数分配窗口的先决条件是已经创建了可以保存参数分配的项目。



## 步骤

1. 启动 SIMATIC 管理器，在项目中打开 HW Config。
2. 双击 CPU 的“计数”子模块。“特性”对话框打开。
3. 分配“计数”子模块的参数，使用“确定”(OK) 关闭参数分配窗口。
4. 使用“站 > 保存并编译”在 HW Config 中保存项目。
5. 当 CPU 处于 STOP 模式时，使用“PLC > 下载到模块...”将参数分配数据下载到 CPU。现在数据即存储在 CPU 的系统数据存储器中。
6. 启动 CPU。

## 在线帮助

分配参数时，参数分配窗口中的在线帮助可为您提供支持。可通过如下几种方式来调用在线帮助：

- 在相应视图中，按“F1”键
- 在不同的参数分配窗口单击**帮助按钮**。

## 5.3.2 基本参数

### 中断选择参数

参数	说明	值范围	缺省
中断选择	可在此选择应当触发技术操作的中断。	<ul style="list-style-type: none"> <li>• 无</li> <li>• 诊断</li> <li>• 处理</li> <li>• 诊断和处理</li> </ul>	无

## 5.3.3 连续、单独和周期计数参数

## 参数说明

参数	说明	值范围	缺省
主计数方向	<ul style="list-style-type: none"> <li>无：没有计数范围限制</li> <li>向上：限制正方向的计数范围。计数器从 0 或装载值开始，沿正方向计数，直到分配的结束值 -1，然后在下一个正编码器脉冲处跳回至装载值。</li> <li>向下：限制负方向的计数范围。计数器从分配的起始值或装载值开始，沿负方向计数到 1，然后在下一个负编码器脉冲处跳至起始值。</li> </ul>	<ul style="list-style-type: none"> <li>无</li> <li>向上（不连续计数）</li> <li>向下（不连续计数）</li> </ul>	无
结束值/起始值	<ul style="list-style-type: none"> <li>主正数方向的结束值</li> <li>主负计数方向的开始值</li> </ul>	2 到 2147483647 ( $2^{31} - 1$ )	2147483647 ( $2^{31} - 1$ )
门功能	<ul style="list-style-type: none"> <li>取消计数操作： 将门关闭并重新启动时，会从装载值开始重新计数。</li> <li>停止计数操作： 门关闭时，计数即停止，当门再次打开时，将从上一个实际值开始重新计数。</li> </ul>	<ul style="list-style-type: none"> <li>取消计数</li> <li>停止计数</li> </ul>	取消计数
比较值	将计数值与比较值比较。请参阅参数“输出特性”		0
	<ul style="list-style-type: none"> <li>无主计数方向</li> </ul>	$-2^{31}$ 到 $+2^{31}-1$	
	<ul style="list-style-type: none"> <li>主计数方向向上</li> </ul>	$-2^{31}$ 到结束值 -1	
	<ul style="list-style-type: none"> <li>主计数方向向下</li> </ul>	1 到 $+2^{31}-1$	
滞后	如果计数值在比较值范围内，则可使用滞后避免频繁的输出切换操作。 0 和 1 表示：关闭滞后。	0 到 255	0
最大频率：计数信号/HW 门	可用固定步长设置轨迹 A/脉冲、轨迹 B/方向和硬件门信号的最大频率。最大值依 CPU 而定：		
	CPU 312C	10、5、2、1 kHz	10 kHz
	CPU 313C、313C-2 DP/PtP	30、10、5、2、1 kHz	30 kHz
	CPU 314C-2 DP/PtP	60、30、10、5、2、1 kHz	60 kHz
最大频率：锁存器	可用固定步长设置锁存器信号的最大频率。最大值依 CPU 而定：		
	CPU 312C	10、5、2、1 kHz	10 kHz
	CPU 313C、313C-2 DP/PtP	30、10、5、2、1 kHz	10 kHz
	CPU 314C-2 DP/PtP	60、30、10、5、2、1 kHz	10 kHz

参数	说明	值范围	缺省
信号判断	<ul style="list-style-type: none"> <li>计数和方向信号与输入相连</li> <li>旋转编码器与输入连接 (单个、双重或四重判断)</li> </ul>	<ul style="list-style-type: none"> <li>脉冲/方向</li> <li>旋转编码器, 单个</li> <li>旋转编码器, 双重</li> <li>旋转编码器, 四重</li> </ul>	脉冲/方向
HW 门	<ul style="list-style-type: none"> <li>是: 通过 SW 和 HW 门进行门控制。</li> <li>否: 仅通过 SW 门进行门控制。</li> </ul>	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
反转计数方向	<ul style="list-style-type: none"> <li>是: 反转了“方向”输入信号。</li> <li>否: 未反转“方向”输入信号。</li> </ul>	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
输出特性	根据该参数置位输出和“比较器”(STS_CMP)状态位。	<ul style="list-style-type: none"> <li>不比较</li> <li>计数值 &gt;= 比较值</li> <li>计数值 &lt;= 比较值</li> <li>比较值时刻的脉冲</li> </ul>	不比较
脉冲持续时间	通过设置“输出特性: 比较值时刻的脉冲”设置, 可指定输出信号的脉冲持续时间。仅可使用偶数值。	0 到 510 ms	0
输入数据的分配	您可选择在“计数”子模块的输入数据(I 数据)中出现最大计数频率 1 kHz 时, 是否可以读取计数值或周期。如果最大计数频率大于 1 kHz, 只能读取“计数值”。	计数值 周期	计数值
时基	您可指定在出现最大计数频率 1 kHz 时, 是否以 125 ns 或 1 μs 为单位测量周期。如果最大计数频率大于 1 kHz, 则只能测量周期。	125 ns 1 μs	125 ns
硬件中断: HW 门打开	软件门打开时, 打开硬件门可产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断: HW 门关闭	软件门打开时, 关闭硬件门可产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断: 达到比较器时	达到比较器时发生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断: 上溢	上溢(超出计数上限)时发生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断: 下溢	下溢(超出计数下限)时发生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断: 计数跳沿	您可选择在出现最大计数频率 1 kHz 时, 每个计数跳沿上是否均发生硬件中断。如果最大计数频率大于 1 kHz, 则无法选择此硬件中断。在每个计数跳沿上发生硬件中断会导致出现较高计数频率时 CPU 利用率很高。因此, 您应该仅在计数跳沿小于 10 ms 时才启用此硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否

5.3 参数组态

5.3.4 频率测量

参数说明

参数	说明	值范围	缺省	
积分时间	测量进入脉冲的时间窗口。	10 到 10 000 ms	100	
下限	将测量值与下限比较。超出下限时，将置位状态位“下溢”(STS_UFLW)。下限必须小于上限。	CPU 312C: 0 到 9 999 999 mHz CPU 313C、 CPU 313C-2 DP/PtP: 0 到 29 999 999 mHz CPU 314C-2 DP/PtP: 0 到 59 999 999 mHz	0	
上限	将测量值与上限比较。超出上限时，置位状态位“上溢”(STS_OFLW)。上限必须大于下限。	CPU 312C: 1 到 9 999 999 mHz CPU 313C、 CPU 313C-2 DP/PtP: 1 到 29 999 999 mHz CPU 314C-2 DP/PtP: 1 到 59 999 999 mHz	CPU 312C: 10 000 000 mHz CPU 313C、 CPU 313C-2 DP/PtP: 30 000 000 mHz CPU 314C-2 DP/PtP: 60 000 000 mHz	
最大计数频率	可用固定步长设置轨迹 A/脉冲、轨迹 B/方向和硬件门信号的最大频率。最大值依 CPU 而定：			
		CPU 312C	10、5、2、1 kHz	10kHz
		CPU 313C、313C-2 DP/PtP	30、10、5、2、1 kHz	30kHz
		CPU 314C-2 DP/PtP	60、30、10、5、2、1 kHz	60kHz
测量值的输出	如果测量频率的周期超过了组态的积分时间： • 对于“直接”频率模式，在积分时间结束时输出“0”值。 • 对于“平均”频率模式，最后的值跨越后续的几个测量间隔分布，无跳沿。 (f >= 1 mHz)。这会延长积分时间。此时，用无跳沿的测量间隔数除上最后测量的值。	<ul style="list-style-type: none"> <li>直接</li> <li>平均</li> </ul>	直接	
信号判断	<ul style="list-style-type: none"> <li>计数和方向信号与输入相连</li> <li>连接带单重判断的旋转传感器</li> </ul>	<ul style="list-style-type: none"> <li>脉冲/方向</li> <li>旋转传感器，单个</li> </ul>	脉冲/方向	
反转计数方向	<ul style="list-style-type: none"> <li>是： 反转了“方向”输入信号。</li> <li>否： 未反转“方向”输入信号。</li> </ul>	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否	

参数	说明	值范围	缺省
HW 门	<ul style="list-style-type: none"> <li>是： 通过 SW 和 HW 门进行门控制。 HW 门信号的最大频率等于最大计数频率设置。</li> <li>否： 仅通过 SW 门进行门控制。</li> </ul>	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
输出反应	将测量值与上下限比较。根据此参数切换输出。	<ul style="list-style-type: none"> <li>不比较</li> <li>超出限制</li> <li>低于下限</li> <li>高于上限</li> </ul>	不比较
硬件中断： 打开 HW 门	软件门打开时，打开硬件门会发生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断： 关闭 HW 门	软件门打开时，关闭硬件门会发生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断： 测量结束	测量结束时产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断： 超出下限	超出下限时产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断： 超出上限	超出上限时产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否

5.3 参数组态

5.3.5 脉冲宽度调制

参数说明

参数	说明	值范围	缺省
输出格式	输出格式	<ul style="list-style-type: none"> <li>千分率</li> <li>S7 模拟值</li> </ul>	千分率
时基	时基 <ul style="list-style-type: none"> <li>接通延迟</li> <li>周期</li> <li>最小脉冲持续时间</li> </ul>	<ul style="list-style-type: none"> <li>0.1 ms</li> <li>1.0 ms</li> </ul>	0.1 ms
接通延迟	输出序列的开始和脉冲输出之间的时间间隔。	0 – 65535	0
周期	根据脉冲持续时间和脉冲间周期定义输出序列的长度。	<ul style="list-style-type: none"> <li>时基 0.1 ms: 4 到 65535</li> <li>时基 1 ms: 1 到 65535</li> </ul>	20 000
最小脉冲持续时间	短于最小脉冲持续时间的输出脉冲/脉冲间周期会被抑制。 对于 1 ms 的时基和值 0，可在内部将最小脉冲持续时间设置为 0.2 ms。	<ul style="list-style-type: none"> <li>时基 0.1 ms: 2 到周期/2</li> <li>时基 1 ms: 0 到周期/2</li> </ul>	2
HW 门	<ul style="list-style-type: none"> <li>是： 通过 SW 和 HW 门进行门控制。</li> <li>否： 仅通过 SW 门进行门控制。</li> </ul>	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
HW 门的过滤器频率	可用固定步长设置硬件门信号的过滤器频率。最大值依 CPU 而定：		
	CPU 312C	10、5、2、1 kHz	10 kHz
	CPU 313C、313C-2 DP/PtP	30、10、5、2、1 kHz	30 kHz
	CPU 314C-2 DP/PtP	60、30、10、5、2、1 kHz	60 kHz
硬件中断： HW 门打开	软件门打开时，打开硬件门可产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否

## 5.4 在用户程序中实现功能

### 步骤

在用户程序中控制各种功能。为此，可调用下列系统功能块：

功能	SFB
计数	SFB COUNT (SFB 47)
频率计数	SFB FREQUENC (SFB 48)
脉冲宽度调制	SFB PULSE (SFB 49)

SFB 在“标准库”的“系统功能块”下。

以下各节可帮助您为您的应用设计用户程序。

可以通过“计数”子模块中设置的输入地址（I 地址）直接在“计数”模式下读取实际计数值并在“频率计数”模式下读取实际频率值。

### 调用 SFB

通过对应的背景数据块调用 SFB

实例：CALL SFB 47, DB30

### 背景数据块

SFB 参数存储在背景数据块中。这些参数在『通过用户程序控制计数器（页码 186）』、『通过用户程序控制频率计数器（页码 204）』和『脉冲宽度调制的步骤（页码 213）』各节中介绍。

可以通过以下方式访问参数：

- DB 号和 DB 中的绝对地址
- DB 号和 DB 中的符号地址

该功能的主要参数也将与该块互连。可在 SFB 中直接分配输入参数，也可以判断输出参数。

---

### 说明

对于每个通道均必须始终通过同一背景数据块调用 SFB，因为该背景数据块中包含 SFB 内部处理过程所需的状态。

不允许对背景数据块的输出进行写访问。

---

5.4 在用户程序中实现功能

程序结构

必须周期性调用（例如 OB 1）SFB。

说明

因为 SFB 不能中断本身，所以不能在具有不同优先级等级的另一程序部分中调用已在程序中组态的 SFB。

例如：不允许在 OB 1 和中断 OB 中调用同一 SFB。

I/O 访问

在计数和频率计数模式下，可通过“计数”子模块的输入地址（I 地址）直接访问 I/O 来读取当前计数值/周期或频率值（取决于设置模式）。

该子模块的输入地址已在“HW Config”中指定。

该子模块的地址区域为 16 个字节。

I 地址	通道	类型	CPU	注释	值范围
n+0	0	DINT	312C、313C、313C-2 DP/PtP、 314C-2 DP/PtP	计数值/周期	-2 <sup>31</sup> 到 2 <sup>31</sup> -1
				频率值	0 到 2 <sup>31</sup> -1
n+4	1	DINT	312C、313C、313C-2 DP/PtP、 14C-2 DP/PtP	计数值/周期	-2 <sup>31</sup> 到 2 <sup>31</sup> - 1
				频率值	0 到 2 <sup>31</sup> - 1
n+8	2	DINT	313C、313C-2 DP/PtP、314C-2 DP/PtP	计数值/周期	-2 <sup>31</sup> 到 2 <sup>31</sup> - 1
				频率值	0 到 2 <sup>31</sup> - 1
			312 C	未分配	0
n + 12	3	DINT	314C-2 DP/PtP	计数值/周期	-2 <sup>31</sup> 到 2 <sup>31</sup> - 1
				频率值	0 到 2 <sup>31</sup> - 1
			312C、313C、313C-2 DP/PtP	未分配	0

n = “计数”子模块的输入地址

在计数模式下，必须指定要读取计数值还是周期。在脉冲宽度调制模式下，将用“0”（从 I 地址开始）填充子模块的整个 I/O 区域。

不能对“计数”子模块（从 Q 地址开始）进行写访问。



## 5.5 计数 - 功能说明

### 5.5.1 基本计数术语

#### 计数模式

计数模式支持计数应用。此时，由 CPU 捕捉并判断计数信号。可向上和向下计数。

可选择下列操作模式：

- 连续计数（例如，用于通过 24 V 增量编码器进行位置反馈）
- 一次计数（例如，用于零件计数，直到最大限值）
- 周期性计数（例如，用于间歇计数应用）

通过参数分配窗口选择操作模式。

#### 最大计数频率

CPU 312C	CPU 313C、CPU 313C-2 DP/PtP	CPU 314C-2 DP/PtP
10 kHz	30 kHz	60 kHz

#### 周期测量

在最大计数频率为 1 kHz 时，始终测量两个连续计数跳沿之间的时间。您可以通过“计数”子模块的输入数据（I 数据）或通过调用 SFC 47 (SFB COUNT) 并读出背景数据块（TIMEVAL 参数）来直接读取测量周期。

始终可在背景数据块中以分配的单位获得测量间隔（TIMEVAL）。必须指定分配的输入数据。可以读取计数值或周期。

如果最大计数频率大于 1 kHz，则不测量周期且该值为 0。

在每次参数分配后，测量周期将在第二个计数跳沿后获得；在此之前，该值为 0。

周期是根据分配的时基获得的无符号 32 位值。

时基为 1  $\mu$ s 时，可以测量的周期最大为 4294 s (71 min, 34 s)，时基为 125 ns 时，可以测量的周期最大为 178 ms (2 min, 58 s)。

如果计数跳沿在时间上相隔很远，则由于未考虑上溢，测量的周期将是不正确的。

5.5 计数 - 功能说明

计数值/装载值

可将缺省值分配给计数器。

这样可以：

- 直接设置计数值。然后，立即应用该计数值。
- 设置装载值。当特定事件发生时，装载值则应用为新的计数值（取决于设置的操作模式）。

主计数方向

可通过指定主计数方向限制计数范围。这将指定应用于“一次计数”和“周期性计数”操作模式下的开始值或结束值的计数限值。

在参数分配窗口中选择主计数方向。

• 无主计数方向：

使用此设置时整个计数范围均可用。

计数下限	-2 147 483 648	$(-2^{31})$
计数上限	+2 147 483 647	$(2^{31} - 1)$

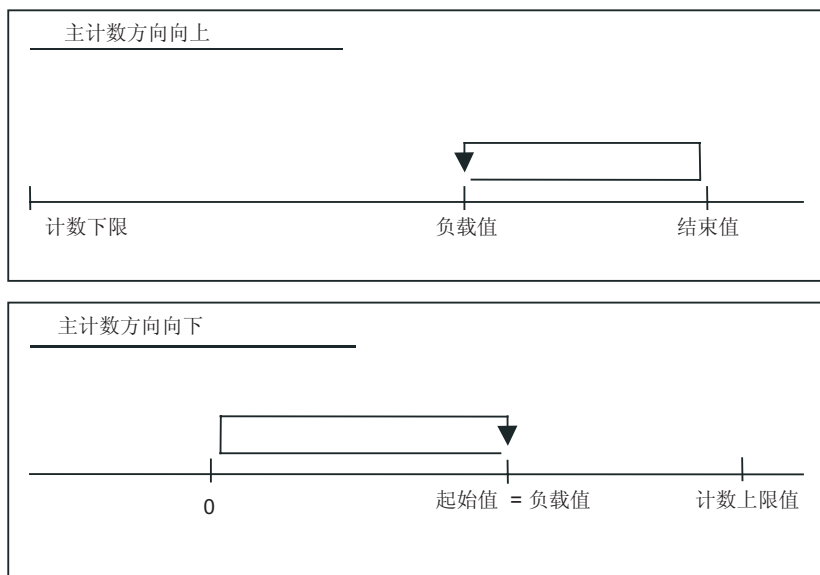
• 主计数方向向上：

主计数方向向上时，将计数范围限制在向上的方向上。计数器从 0 或装载值开始沿正方向计数，直到达到分配的结束值 - 1，然后在下一个正编码器脉冲处跳回至装载值。

• 主计数方向向下：

主计数方向向下时，将计数范围限制在向下的方向上。计数器从分配的起始值或装载值开始沿负方向计数，直到值 1，然后在下一个负编码器脉冲处跳回至起始值。

指定计数方向，这与“主计数方向”参数无关。为此，需应用适当的方向信号或在参数分配期间分配计数方向。



## 启动/停止计数器

使用门功能可启动、停止和中断计数功能。有关设置门功能的信息，请参考『计数器的门功能（页码 194）』一节。

## 上溢/过零点/下溢

超出计数上限时，将设置上溢位 (STS\_OFLW)。

低于计数下限时，将设置下溢位 (STS\_UFLW)。

通过设置过零点位 (STS\_ZP) 指示过零点。仅当在无主计数方向的情况下计数时，才设置此位。计数器被设置为 0 或从装载值 0 开始计数时，也将指示过零点。

## 5.5.2 连续计数

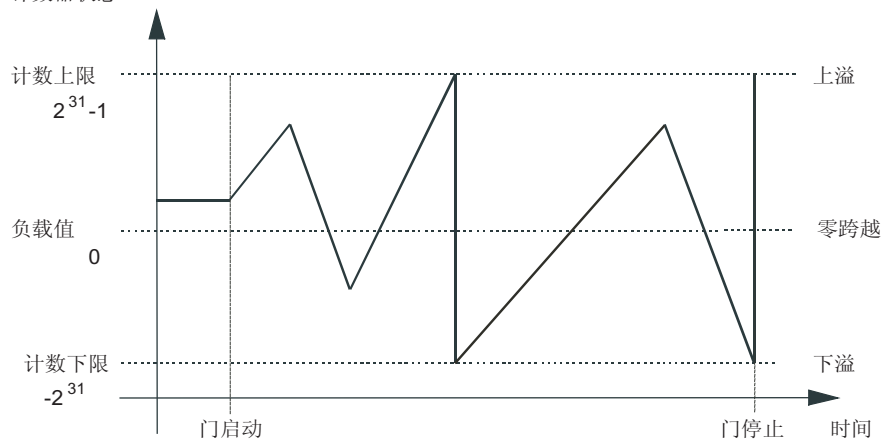
### 说明

在此操作模式下，CPU 从 0 或装载值开始计数。

- 向上计数达到上限时，它将在出现下一正计数脉冲时跳至下限处，并从此处恢复计数。
- 向下计数达到下限时，它将在出现下一负计数脉冲时跳至上限处，并从此处恢复计数。
- 计数限值设置为最大范围。

	有效范围	缺省值
计数上限	+2147483647 ( $2^{31} - 1$ )	/
计数下限	-2147483648 ( $-2^{31}$ )	/
计数值	-2147483648 ( $-2^{31}$ ) 到 +2147483647 ( $2^{31} - 1$ )	0
装载值	-2147483647 ( $-2^{31} + 1$ ) 到 +2147483646 ( $2^{31} - 2$ )	0

计数器状态



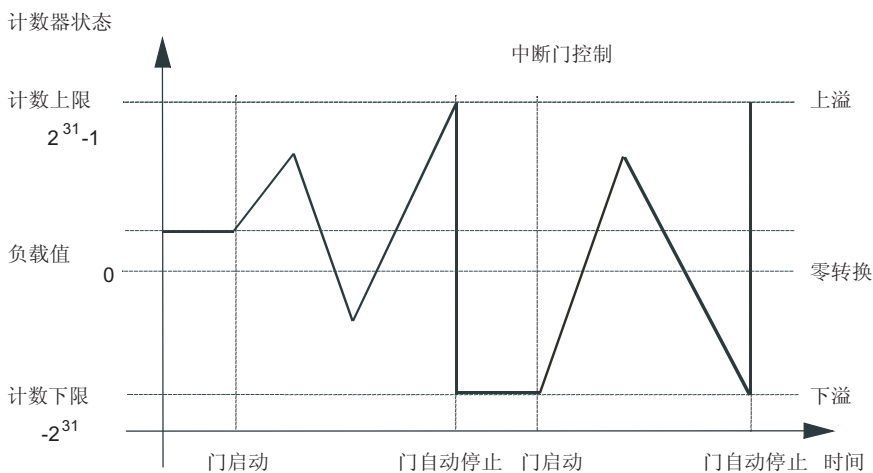
### 5.5.3 单循环计数

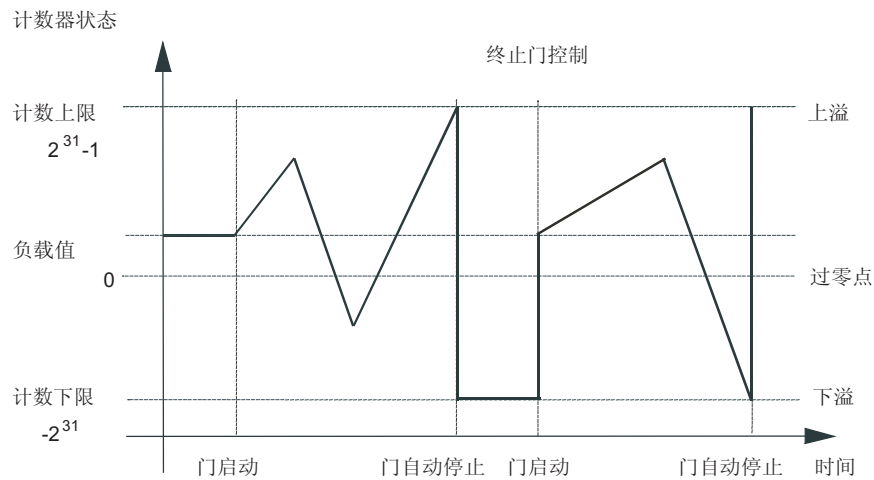
#### 说明

在此操作模式下，CPU 计数一次（取决于分配的主计数方向）。

- 无主计数方向：
  - CPU 从装载值开始计数一次。
  - CPU 向上或向下计数。
  - 计数限值设置为最大计数范围。
  - 计数限值处发生上溢或下溢时，计数器将跳至相反的计数限值，且门自动关闭。要重新启动计数，必须在门控制处生成一个正跳沿。
  - 停止门控制时，将从当前计数开始重新计数。
  - 取消门控制时，将从装载值开始重新计数。

	有效值范围	缺省值
计数上限	+2147483647 ( $2^{31} - 1$ )	/
计数下限	-2147483648 ( $-2^{31}$ )	/
计数值	-2147483648 ( $-2^{31}$ ) 到 +2147483647 ( $2^{31} - 1$ )	0
装载值	-2147483647 ( $-2^{31} + 1$ ) 到 +2147483646 ( $2^{31} - 2$ )	0

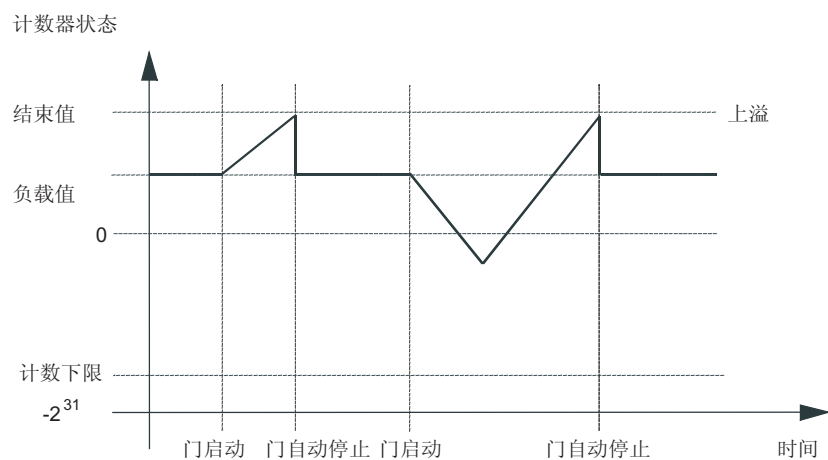




- 主计数方向向上:

- CPU 从装载值开始计数。
- CPU 向上或向下计数。
- 计数器在正方向上达到结束值 - 1 后，将在下一个正计数脉冲处跳回至装载值，并且门将自动关闭。  
要重新启动计数，必须在门控制处生成一个正跳沿。计数器从装载值开始计数。
- 您也可以超出计数下限进行计数。但在这种情况下，计数值与比较结果不匹配。因此，应避免在此范围内操作。

	有效值范围	缺省值
结束值	至 +2147483646 ( $2^{31} - 1$ )	可分配
计数下限	-2147483648 ( $-2^{31}$ )	/
计数值	-2147483648 ( $-2^{31}$ ) 到结束值 - 1	0
装载值	-2147483648 ( $-2^{31}$ ) 到结束值 - 2	0



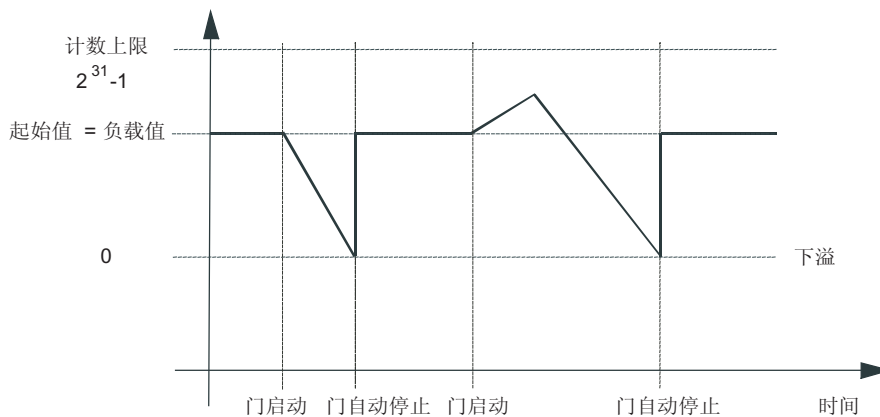
5.5 计数 - 功能说明

• 主计数方向向下:

- CPU 从装载值开始计数。
- CPU 向上或向下计数。
- 计数器在负方向上达到计数值 1 后，将在下一个负计数脉冲处跳回至装载值（起始值），并且门将自动关闭。  
要重新启动计数，必须在门控制处生成一个正跳沿（请参阅『计数器的门功能（页码 194）』一节）。计数器从装载值开始计数。
- 您也可以超出计数上限进行计数。但在这种情况下，计数值与比较结果不匹配。因此，应避免在此范围内进行操作。

	有效值范围	缺省值
起始值	至 +2147483647 ( $2^{31} - 1$ )	可分配
计数上限	+2147483647 ( $2^{31} - 1$ )	/
计数值	到 +2147483647 ( $2^{31} - 1$ )	起始值
装载值	到 +2147483647 ( $2^{31} - 1$ )	起始值

计数器状态



## 5.5.4 周期性计数

### 说明

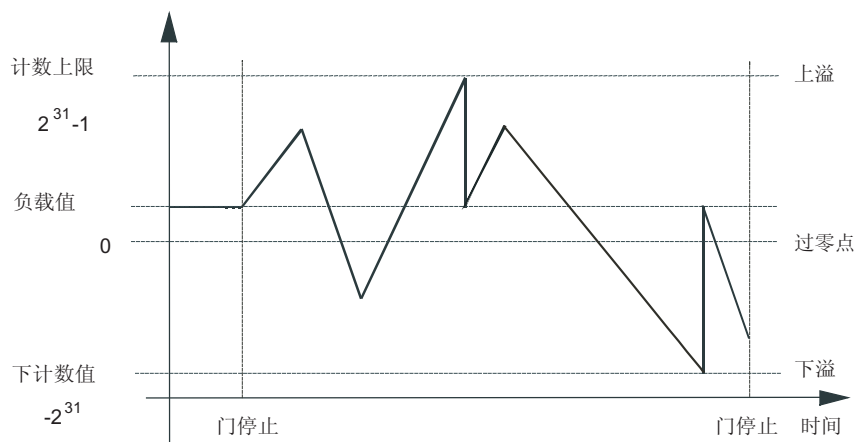
在此操作模式下，CPU 根据声明的缺省计数方向执行周期性计数。

- 无缺省计数方向：

- CPU 从装载值开始计数。
- CPU 向上或向下计数。
- 在相应的计数限值处上溢或下溢时，计数器将跳至装载值并从该值开始恢复计数。
- 计数限值设置为最大范围。

	有效范围	缺省值
计数上限	+2147483647 ( $2^{31} - 1$ )	/
计数下限	-2147483648 ( $-2^{31}$ )	/
计数值	-2147483648 ( $-2^{31}$ ) 到 +2147483647 ( $2^{31} - 1$ )	0
装载值	-2147483647 ( $-2^{31} + 1$ ) 到 +2147483646 ( $2^{31} - 2$ )	0

计数器状态



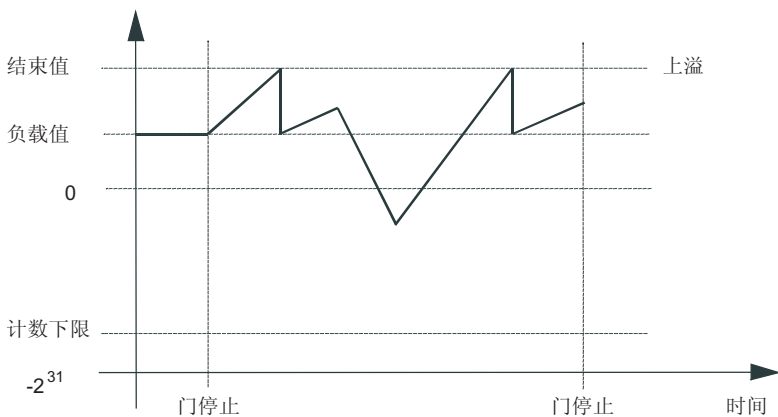
5.5 计数 - 功能说明

● 缺省为向上计数:

- CPU 从装载值开始计数。
- CPU 向上或向下计数。
- 计数器沿正方向计数到结束值-1 后，将在出现下一个正计数脉冲时跳回至装载值，并从该值开始恢复计数。
- 也可超出计数下限。但在这种情况下，计数值与比较结果不匹配。因此，应避免在此范围内进行操作。

	有效范围	缺省值
结束值	到 +2147483647 ( $2^{31} - 1$ )	可组态
计数下限	-2147483648 ( $-2^{31}$ )	/
计数值	-2147483648 ( $-2^{31}$ ) 到结束值 -1	0
装载值	-2147483648 ( $-2^{31}$ ) 到结束值 -2	0

计数器状态



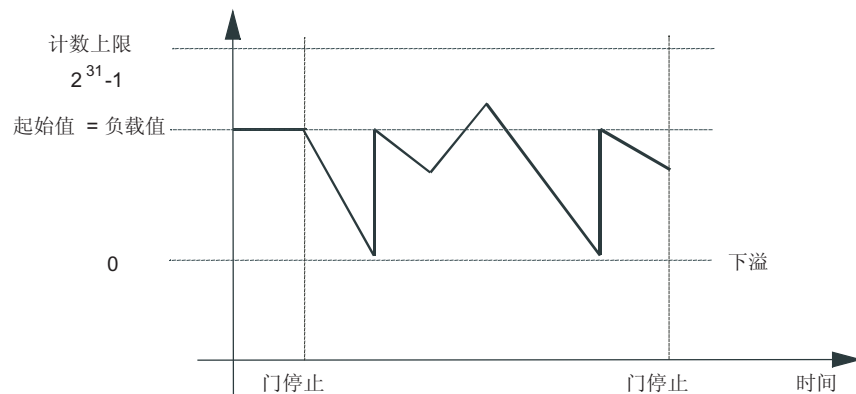


- 缺省为向下计数:

- CPU 从装载值开始计数。
- CPU 向上或向下计数。
- 计数器沿负方向计数到值 1 后, 将在出现下一个负计数脉冲时跳回至装载值 (开始值), 并该值开始恢复计数。
- 也可超出计数上限。但在这种情况下, 计数值与比较结果不匹配。因此, 应避免在此范围内进行操作。

	有效范围	缺省值
起始值	到 +2147483647 ( $2^{31} - 1$ )	可组态
计数上限	+2147483647 ( $2^{31} - 1$ )	/
计数值	到 +2147483647 ( $2^{31} - 1$ )	起始值
装载值	到 +2147483647 ( $2^{31} - 1$ )	起始值

计数器状态



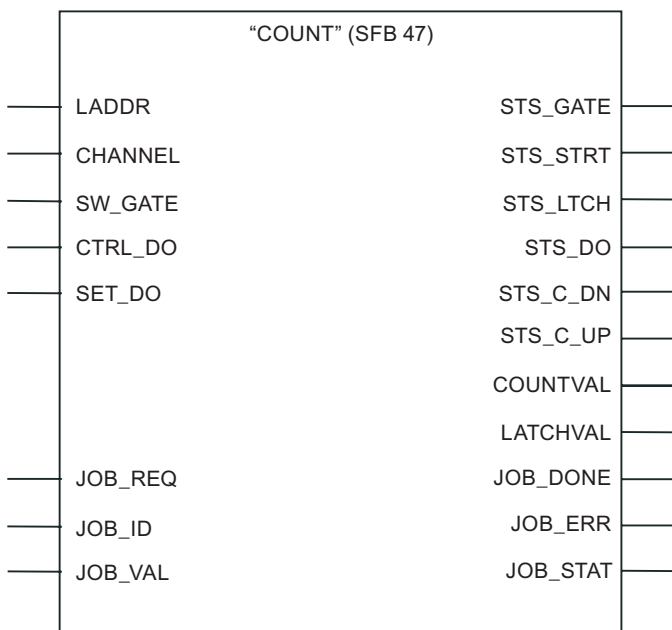
### 5.5.5 通过用户程序控制计数器

#### 说明

要从用户程序中控制计数器，请使用 **SFB COUNT (SFB 47)**。

可使用下列功能：

- 通过软件门 SW\_GATE 启动/停止计数器
- 启用/控制输出 DO
- 读出状态位
- 读取当前计数值和锁存器值
- 用于读/写内部计数寄存器的作业
- 读出当前周期（不与块互连，但仅在背景数据块中可用）



## 输入参数

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
LADDR	WORD	0	已在“HW Config”中指定的子模块 I/O 地址。 如果输入和输出地址不相同，则必须指定两者中较低地址。	CPU 特定	300 (十六进制)
CHANNEL	INT	2	通道号： CPU 312C CPU 313C CPU 313C-2 DP/PtP CPU 314C-2 DP/PtP	0-1 0-2 0-3	0
SW_GATE	BOOL	4.0	软件门 用于计数器启动/停止	TRUE/FALSE	FALSE
CTRL_DO	BOOL	4.1	启用输出	TRUE/FALSE	FALSE
SET_DO	BOOL	4.2	控制输出	TRUE/FALSE	FALSE

## 说明

如果您已通过参数分配窗口将“输出特性”参数设置为“不比较”，则将进行以下操作：

- 切换输出作为标准输出。
- SFB 输入参数 CTRL\_DO 和 SET\_DO 无效。
- 状态位 STS\_DO 和 STS\_CMP（IDB 中的状态比较器）保持复位状态。

## 不与块互连的输入参数（静态局部数据）

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
RES_STS	BOOL	32.2	复位状态位 复位状态位 STS_CMP、STS_OFLW、 STS_UFLW 和 STS_ZP。 必须调用两次 SFB 才能复位状态位。	TRUE/FALSE	FALSE

5.5 计数 - 功能说明

输出参数

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
STS_GATE	BOOL	12.0	内部门状态	TRUE/FALSE	FALSE
STS_STRT	BOOL	12.1	硬件门状态 (启动输入)	TRUE/FALSE	FALSE
STS_LTCH	BOOL	12.2	锁存器输入状态	TRUE/FALSE	FALSE
STS_DO	BOOL	12.3	输出状态	TRUE/FALSE	FALSE
STS_C_DN	BOOL	12.4	向下计数的状态。 始终指示最后的计数方向。在第一次调用 SFB 之后, STS_C_DN 的值为 FALSE。	TRUE/FALSE	FALSE
STS_C_UP	BOOL	12.5	向上计数的状态。 始终指示最后的计数方向。在第一次调用 SFB 之后, STS_C_UP 的值为 TRUE。	TRUE/FALSE	FALSE
COUNTVAL	DINT	14	当前计数值	-2 <sup>31</sup> 到 2 <sup>31</sup> - 1	0
LATCHVAL	DINT	18	当前锁存器值	-2 <sup>31</sup> 到 2 <sup>31</sup> - 1	0

不与块互连的输出参数 (静态局部数据)

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
STS_CMP	BOOL	26.3	比较器状态。*) 状态位 STS_CMP 表示是否满足比较器的比较条件。 STS_CMP 还表示已置位输出 (STS_DO = TRUE)	TRUE/FALSE	FALSE
STS_OFLW	BOOL	26.5	上溢状态 *)	TRUE/FALSE	FALSE
STS_UFLW	BOOL	26.6	下溢状态 *)	TRUE/FALSE	FALSE
STS_ZP	BOOL	26.7	过零点状态 *) 仅当在无主计数方向的情况下计数时, 才置位此位。 指示过零点。如果计数器设置为 0, 或计数器在装载值 = 0 时开始计数, 也会置位此位。	TRUE/FALSE	FALSE
TIMEVAL	DINT	34.0	测量周期	0 到 2 <sup>32</sup> - 1	0

\*) 使用 RES\_STS 复位

### 5.5.6 通过请求接口读写计数器

#### 说明

您可以使用作业接口来读/写计数寄存器。

#### 要求

最后一个作业必须完成 (JOB\_DONE = TRUE)。

#### 步骤

- 分配以下输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_REQ	BOOL	4.3	作业请求（正跳沿）	TRUE/FALSE	FALSE
JOB_ID	WORD	6	作业号： <ul style="list-style-type: none"> <li>• 不带有功能的作业</li> <li>• 写入计数值</li> <li>• 写装载值</li> <li>• 写入比较值</li> <li>• 写入滞后</li> <li>• 写入脉冲持续时间</li> <li>• 读装载值</li> <li>• 读比较值</li> <li>• 读取滞后</li> <li>• 读取脉冲持续时间</li> </ul>	00（十六进制） 01（十六进制） 02（十六进制） 04（十六进制） 08（十六进制） 10（十六进制） 82（十六进制） 84（十六进制） 88（十六进制） 90（十六进制）	0
JOB_VAL	DINT	8	写作业的值	-2 <sup>31</sup> 到 +2 <sup>31</sup> -1	0

- 调用 SFB。

SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_DONE	BOOL	22.0	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	BOOL	22.1	错误作业	TRUE/FALSE	FALSE
JOB_STAT	WORD	24	作业错误编号	0 到 FFFF （十六进制）	0

- 调用 SFB 后将立即处理作业。在一个 SFB 循环时间内 JOB\_DONE 被置位为 FALSE。
- 如果出现错误，则 JOB\_ERR = TRUE。随后将在 JOB\_STAT 中指出确切的出错原因。
- 可以通过 JOB\_DONE = TRUE 来启动新作业。

5.5 计数 - 功能说明

- 仅对于读取作业：从背景数据块读取参数 JOB\_OVAL 的当前值。

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
JOB_OVAL	DINT	28	读取作业的输出值	$-2^{31}$ 到 $2^{31} - 1$	0



JOB\_VAL 允许的值范围

连续计数

作业	有效值范围
直接写入计数器	$-2^{31} + 1$ 到 $+2^{31} - 2$
写入装载值	$-2^{31} + 1$ 到 $+2^{31} - 2$
写入比较值	$-2^{31}$ 到 $+2^{31} - 1$
写入滞后	0 到 255
写入脉冲持续时间。只允许偶数值。将自动舍入奇数值。	0 到 510 ms

一次计数/周期性计数，无主计数方向

作业	有效值范围
直接写入计数器	$-2^{31} + 1$ 到 $+2^{31} - 2$
写入装载值	$-2^{31} + 1$ 到 $+2^{31} - 2$
写入比较值	$-2^{31}$ 到 $+2^{31} - 1$
写入滞后	0 到 255
写入脉冲持续时间。只允许偶数值。将自动舍入奇数值。	0 到 510 ms

## 一次计数/周期性计数，主计数方向向上

作业	有效值范围
结束值	到 +2147483646 ( $2^{31} - 1$ )
直接写入计数器	-2147483648 ( $-2^{31}$ ) 到结束值 - 2
写入装载值	-2147483648 ( $-2^{31}$ ) 到结束值 - 2
写入比较值	-2147483648 ( $-2^{31}$ ) 到结束值 - 1
写入滞后	0 到 255
写入脉冲持续时间。只允许偶数值。将自动舍入奇数值。	0 到 510 ms

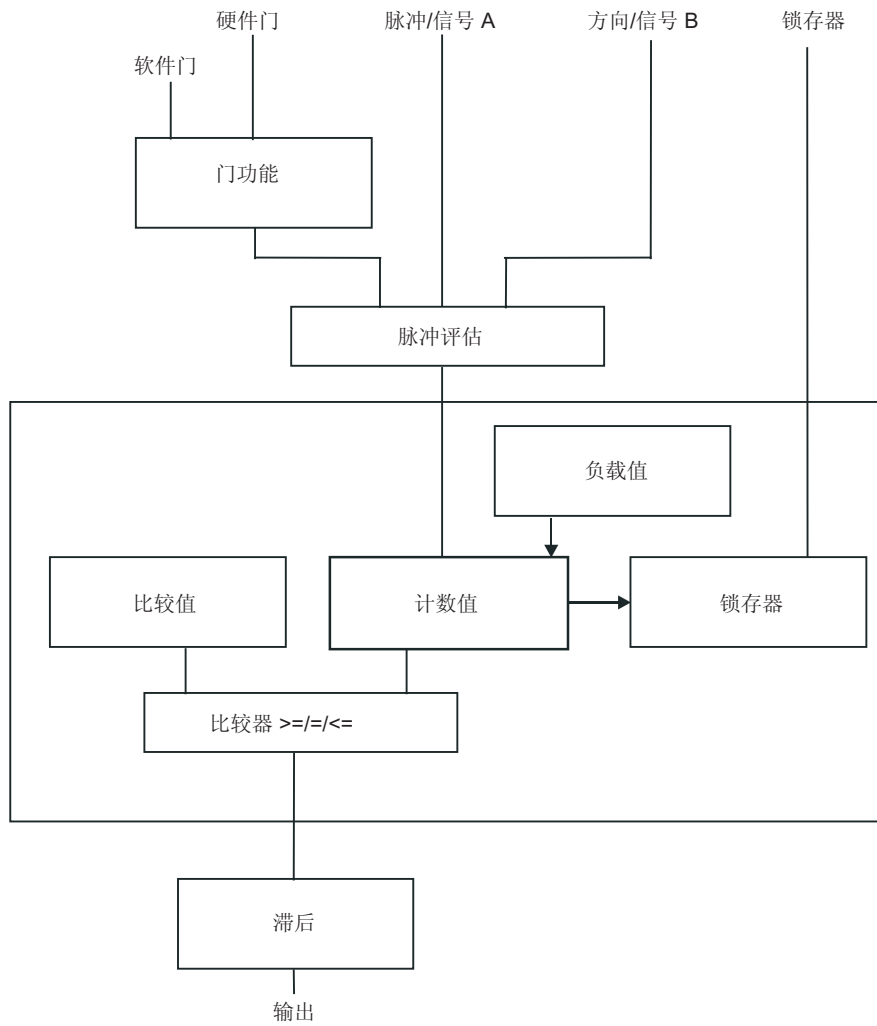
## 一次计数/周期性计数，主计数方向向下

作业	有效值范围
直接写入计数器	2 到 +2147483647 ( $2^{31} - 1$ )
写入装载值	2 到 +2147483647 ( $2^{31} - 1$ )
写入比较值	1 到 +2147483647 ( $2^{31} - 1$ )
写入滞后	0 到 255
写入脉冲持续时间。只允许偶数值。将自动舍入奇数值。	0 到 510 ms

### 5.5.7 计数器 FB

#### 结构

下图显示了将在以下各章节中说明的各种功能块：





## 5.5.8 计数器输入

### 脉冲/A

在此处连接计数信号或编码器的轨迹 **A**。可通过单、双或四重判断模式连接编码器。

### 方向/B

在此处连接方向信号或传感器的轨迹 **B**。可以在参数分配窗口指定对方向信号取反。

---

#### 说明

不会监视输入是否缺少脉冲。

---

### 锁存器

可通过在数字输入“锁存器”处生成一个正跳沿来保存实际的内部值。

这样便使您能实现与事件相关的计数值判断功能。可在每次调用 **SFB** 时在 **SFB** 参数 **LATCHVAL** 处输出实际锁存的值。

**CPU** 进行 **STOP-RUN** 转换后，会将 **LATCHVAL** 重置为计数器的开始值。

### 硬件门

可通过数字输入“硬件门”启动计数器。

### 5.5.9 计数器的门功能

#### 基本信息

可选择以下两个门来控制计数器：

- 通过用户程序控制的**软件门**（SW 门）。  
可通过 SFB 参数 **SW\_GATE** 上的正跳沿打开软件门。复位该参数以关闭它。
- **硬件门**（HW 门）。可以在参数分配窗口分配硬件门。在数字输入“硬件门”处，通过正跳沿打开门，通过负跳沿关闭门。

#### 内部门

内部门表示硬件和软件门的逻辑 AND 连接。仅当硬件和软件门均打开时才会启用计数操作。状态位 **STS\_GATE**（状态内部门）显示此状态。

如果尚未指定使用硬件门，则软件门组态便与之相关。

计数操作通过内部门电路启动、中断、恢复和取消。

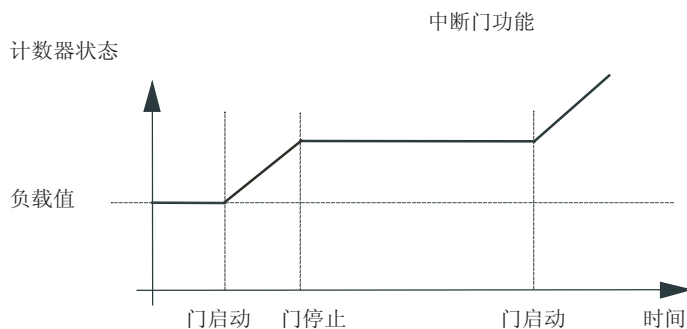
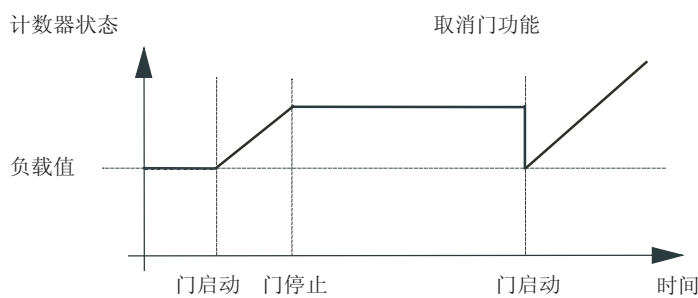
在单独计数模式下，内部门电路在出现计数限值处的上溢/下溢时自动关闭。

#### 取消和中断门功能

在进行门功能的参数分配时，可为内部门指定计数取消或计数中断操作：

- 设置为计数取消门操作时，在关闭并重新启动门后将从装载值开始重新开始计数操作。
- 设置为计数中断门操作时，在关闭门后将从最后的实际计数值开始恢复计数。

下图显示了计数取消和计数中断门操作的响应：



### 仅通过 SW 门来控制门

在参数分配窗口中，在“门功能”参数中指定 SW 门打开时 CPU 应如何响应：

- 通过 SW 门进行门控制，“取消计数”组态

操作	响应
SW 门 0 -> 1	从装载值开始

- 通过 SW 门进行门控制，“中断计数”组态

操作	响应
SW 门 0 -> 1	从实际计数值开始恢复计数

### 通过 SW 和 HW 门进行门控制

在参数分配窗口中，在“门功能”参数中指定 SW 和 HW 门打开时 CPU 应如何响应：

- 通过 SW 和 HW 门进行门控制，“取消计数”组态

要求	操作	响应
HW 门打开	SW 门 0 -> 1	从实际计数值开始重新计数
SW 门打开	HW 门 0 -> 1	从装载值开始

- 通过 SW 和 HW 门进行门控制，“中断计数”组态

要求	操作	响应
HW 门打开	SW 门 0 -> 1	从实际计数值开始重新计数
SW 门打开	HW 门 0 -> 1	从实际计数值开始重新计数

### 在“单独计数”模式中通过 SW 和 HW 门进行门控制

内部门自动关闭后，仅可通过以下方式重新打开：

- 在 SW 门打开时在 HW 门处生成一个正跳沿，或
- 在 HW 门处生成一个正跳沿并随后打开 SW 门。

## 5.5.10 计数器输出的响应

### 引言

该部分描述了数字输出的响应。

### 比较值

在 CPU 中，可以存储分配给数字输出，状态位“状态比较器”(STS\_CMP)中和到硬件中断的的比较值。可以根据计数值和比较值启用数字输出。

可以在参数分配窗口中指定比较值。在您的用户程序中，可以通过 SFB 的请求接口读取 (JOB\_ID = 84 十六进制) 和写入 (JOB\_ID = 04 十六进制) 它们。

### 数字输出的响应

可以通过参数分配窗口指定下列响应：

- 不比较
- 计数值  $\geq$  比较值
- 计数值  $\leq$  比较值
- 比较值时刻的脉冲

#### 不比较

将输出切换为标准输出。

禁用 SFB 输入参数 CTRL\_DO 和 SET\_DO。

状态位 STS\_DO 和 STS\_CMP (IDB 中的状态比较器) 保持复位状态。

#### 计数值 $\geq$ 比较值或计数值 $\leq$ 比较值

在达到比较值时，比较器切换输出。

此时，必须首先置位控制位 CTRL\_DO。

状态位 STS\_CMP 显示比较运算的结果。仅在所有比较条件不再相关之后才可以复位该状态位。

#### 比较值时刻的脉冲

当计数值达到比较值时，比较器将使输出持续激活指定的脉冲周期长度。如果已经组态了缺省计数方向，则仅会在达到为缺省方向指定的值之后才激活输出。

此时，必须首先置位控制位 CTRL\_DO。

位 STS\_DO 的状态总是与数字输出的状态相同。

状态位 STS\_CMP 显示比较运算的结果。仅可在脉冲周期过后复位该状态位。

## 状态位 STS\_CMP

状态位 **STS\_CMP** 指明是激活还是已经激活了相应的输出。必须用 **RES\_STS** 复位该状态位。如果输出仍处于激活状态，则将复位相应的位并立即再次置位。如果使用 **SET\_DO** 启用禁用的输出 (**CTRL\_DO = FALSE**)，该状态位也会置位。

---

### 说明

必须调用 **SFB** 两次，以使用 **RES\_STS** 复位该状态位。

---

## 同时控制输出和比较器

如果已选择了输出的比较功能，则可以使用 **SET\_DO** 同时控制输出（条件：**CTRL\_DO = TRUE**）。注意下列规则：

- 比较器将输出状态从“0”切换为“1”：  
而当 **SET\_DO = FALSE** 时，比较器将输出复位为“0”。每个进入的计数脉冲都会重新触发比较器运算。因此，输出将根据比较器运算的结果被复位或置位。
- 比较器在 **SET\_DO=TRUE** 从“0”变到“1”时进行切换：  
仅当 **SET\_DO = FALSE** 时，比较器将输出复位为“0”。

## “比较值时刻的脉冲”组态的特性

### 数字输出的响应

通过控制位 **SET\_DO** 置位数字输出后，将在脉冲周期过后复位。

- 在脉冲周期 = 0 并且计数值不在比较值范围内时，不能通过 **SET\_DO** 控制输出。
- 在脉冲周期 = 0 并且计数值 = 比较值时，可以通过 **SET\_DO** 控制输出。

### 脉冲宽度

可以指定一个脉冲周期，使其适应您当前使用的执行器。脉冲周期确定应当在多长时间内置位输出。您可以在 0 到 510 ms 内以 2 ms 为增量来指定脉冲周期。请注意，计数脉冲必须要比数字输出的最小切换时间长。

如果脉冲周期 = 0，则直到比较条件不再适用时才置位输出。

如果相应的数字输出置位，将触发启动脉冲周期。脉冲周期的误差 < 1 ms。

如果在脉冲输出期间再次超过或者达到比较值，不会重新触发脉冲周期。

可以在参数分配窗口中自定义脉冲周期，并可以在用户程序中通过 **SFB** 请求接口写入（**JOB\_ID = 10** [十六进制]）和读取（**JOB\_ID = 90** [十六进制]）它们。

在下一个脉冲到来之前，将不会应用在运行期间更改的脉冲周期。

### 5.5.11 滞后对计数器模式的影响

#### 说明

编码器可能停止在某个位置，并且随后在该位置附近“颤动”。在此状态下，计数会围绕一个特定值波动。例如，如果比较值位于该波动范围内，则关联的输出将按照波动的节奏打开和关闭。CPU 配有可分配的滞后，可防止发生微小波动时出现这种切换。

您可以在 0 到 255 内选择一个范围。设置为 0 和 1 时，将禁用滞后。

滞后还作用于过零点和上溢/下溢。

您可以在参数分配窗口中指定滞后，并可以在用户程序中通过 SFB 的作业接口写入 (**JOB\_ID = 08** [十六进制]) 或读取 (**JOB\_ID = 88** [十六进制]) 滞后。

#### 响应变化

更改后，激活的滞后仍处于激活状态。当达到下一个比较值时，应用新的滞后范围。

#### “计数器值 $\geq$ 比较值” 或 “计数器值 $\leq$ 比较值” 时的作用

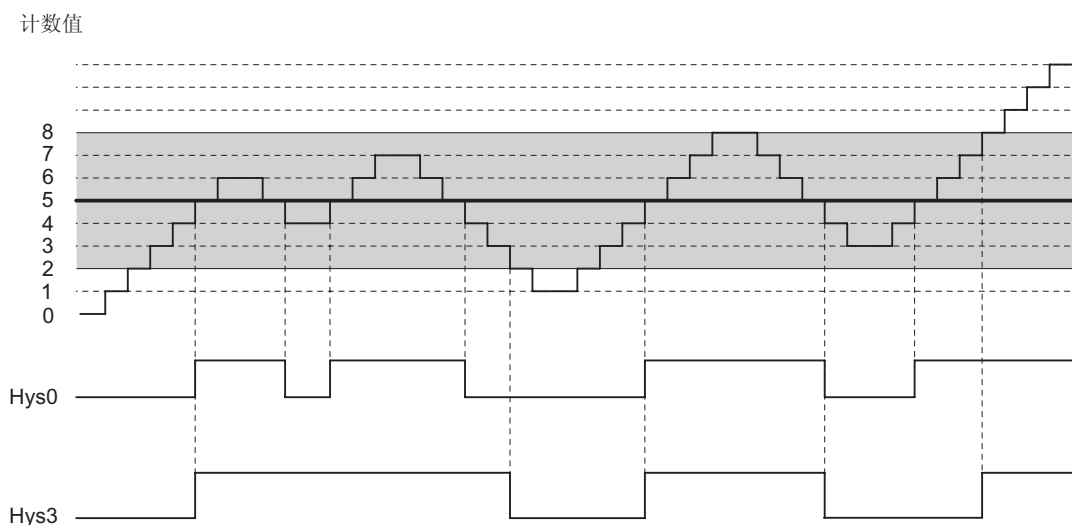
下图中的实例说明了滞后作用。此图显示了当分配的滞后值为 0 (= 关闭) 和 3 时，输出特性的区别。该实例中比较值 = 5。

如下所述组态计数器：

- “主计数方向向上”
- 输出 “计数器值  $\geq$  比较值时打开”

到达比较值时启用滞后。滞后处于激活状态时，比较结果保持不变。

如果计数值不在滞后范围内，则将禁用滞后。然后，比较器根据其比较条件重新进行切换。



### “比较值时刻的脉冲”和“脉冲持续时间等于零”时的作用

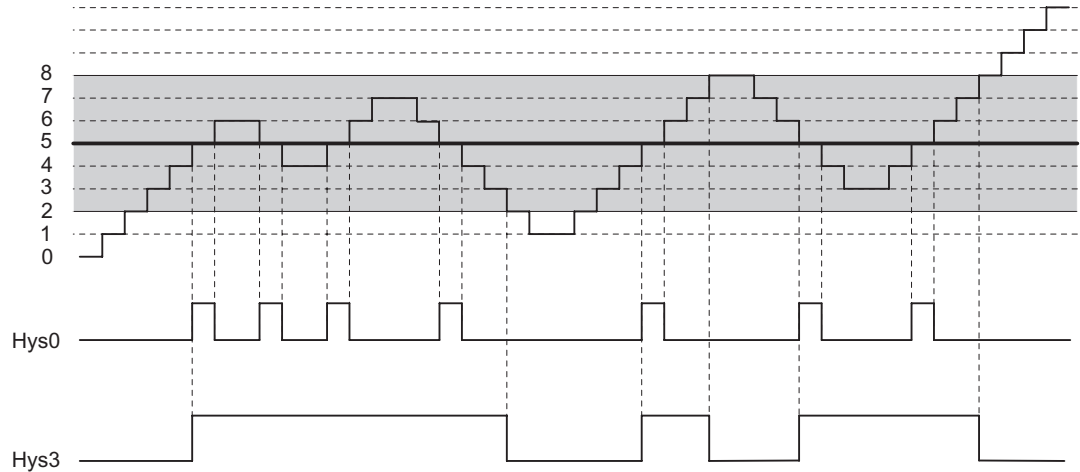
下图中的实例说明了滞后作用。此图显示了当分配的滞后值为 0 (= 关闭) 和 3 时，输出特性的区别。该实例中比较值 = 5。

如下组态计数器：

- “无主计数方向”
- “达到比较值时的脉冲”
- “脉冲持续时间 = 0”

达到比较条件时，激活滞后。滞后处于激活状态时，比较结果保持不变。如果计数值不在滞后范围内，则滞后将不再处于激活状态。

计数值



### “比较值时刻的脉冲”和“脉冲持续时间不等于零”时的作用

下图中的实例说明了滞后作用。此图显示了当分配的滞后值为 0 (= 关闭) 和 3 时，输出特性的区别。该实例中比较值 = 5。

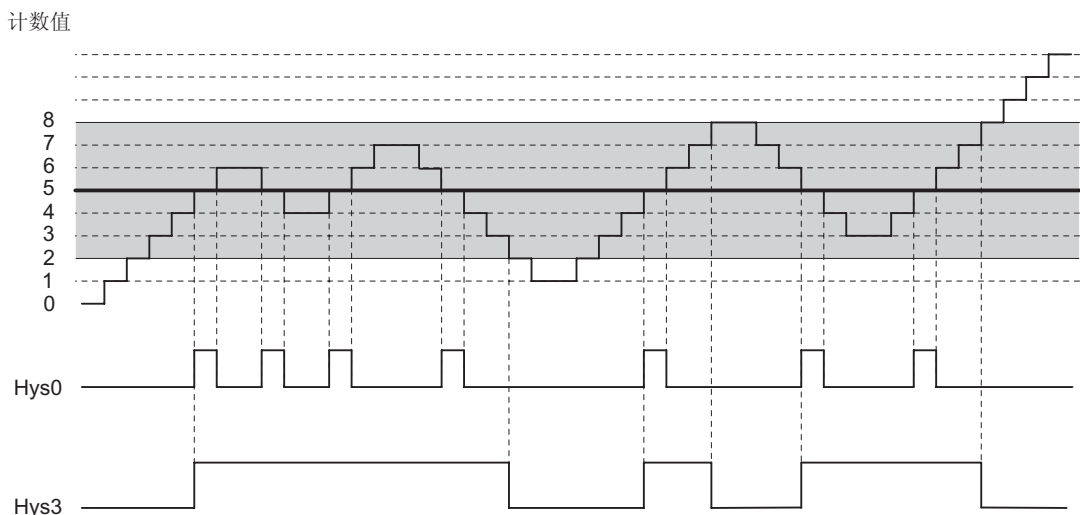
如下组态计数器：

- “无主计数方向”
- “达到比较值时的脉冲”
- “脉冲持续时间 > 0”

达到比较条件时，激活滞后且输出具有分配的持续时间的脉冲。

如果计数值不在滞后范围内，则滞后将不再处于激活状态。

滞后处于激活状态时，CPU 会记住计数方向。如果计数值在记住的计数方向上超出滞后范围，则输出脉冲。





## 5.5.12 计数运行期间的硬件中断

### 设置硬件中断

在参数分配窗口中启用硬件中断并指定触发硬件中断的事件：

- 当 SW 门关闭时打开 HW 门
- 当 SW 门打开时关闭 HW 门
- 上溢（超出计数上限）
- 下溢（低于计数下限）
- 达到比较器（计数值 = 比较值）
- 出现计数跳沿

---

#### 说明

在每个计数跳沿上触发硬件中断会导致计数频率较高时 CPU 利用率很高。如果“计数”子模块中硬件中断的发生速度比硬件中断 OB (OB 40) 中的处理速度快，则将产生诊断“硬件中断丢失”（假设启用了诊断中断）。

高计数频率会导致 CPU 利用率很高以至于超出了组态的扫描周期监视时间，或者导致 CPU 通讯不再响应或响应速度很慢。

因此，建议您不要为所有计数跳沿启用硬件中断，除非确定各个跳沿至少间隔 10 ms。

---

### 也参见

组态硬件中断（页码 227）

## 5.6 频率测量功能描述

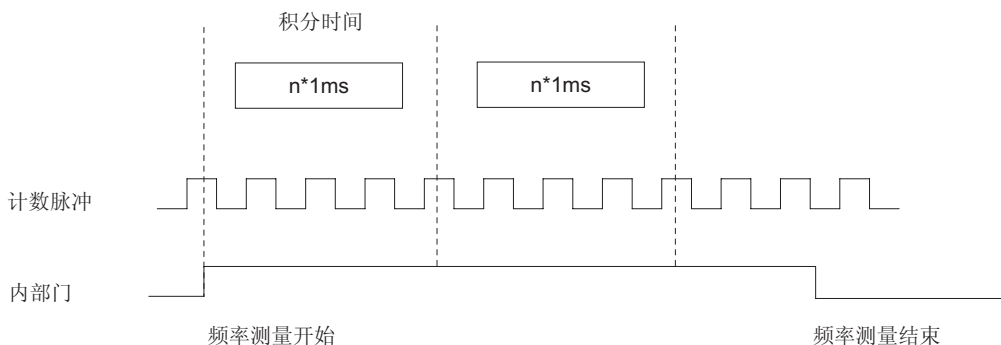
### 5.6.1 频率测量，步骤

#### 基本信息

在此操作模式下，CPU 在指定的积分时间内对进入脉冲进行计数并将其作为频率值输出。

您可以在 10 ms 到 10 000 ms 之间设置积分时间值（以 1 ms 为增量）。可以在参数分配窗口中设置积分时间，或者在用户程序中读写积分时间（请参阅『通过用户程序控制频率计数器（页码 204）』一节）。

计算得到的频率值的单位是“mHz”。您可以在用户程序中使用 SFB 参数 **MEAS\_VAL** 读出该值。当新值可用时，将置位 **STS\_CMP**（有关 SFB 参数的说明，请参考『通过用户程序控制频率计数器（页码 204）』一节）。



#### 测量步骤

在指定的积分时间内执行测量。在积分时间结束后更新测量值。

如果测量频率的周期超过分配的积分时间，则返回 0 值或平均值作为测量值（取决于参数分配）。

第一个积分时间结束之前，返回该值 - 1。

#### 频率范围

CPU 312C	CPU 313C、CPU 313C-2 DP/PtP	CPU 314C-2 DP/PtP
0 到 10 kHz	0 到 30 kHz	0 到 60 kHz

## 反向

如果在积分时间内发生反向，则该测量周期内的测量值是不确定的。您可以通过判断方向的状态位 STS\_C\_UP、STS\_C\_DN 来对可能的过程无规律性进行反应（请参阅『通过用户程序控制频率计数器（页码 204）』一节）。

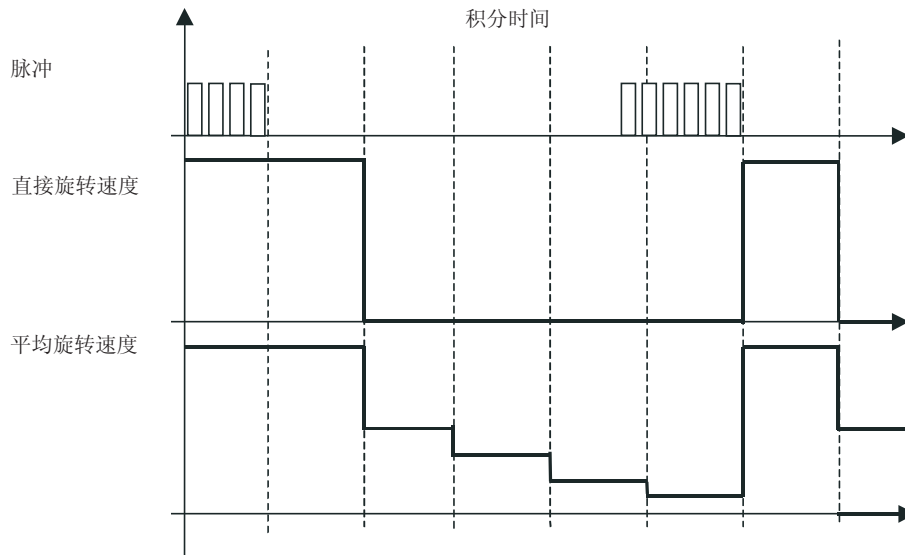
## 直接/平均频率值

在积分时间结束时显示测量频率 ( $f \geq 1 \text{ mHz}$ )。

如果测量频率的周期超过了分配的积分时间：

- 对于直接频率，在积分时间结束时输出“0”值。
- 对于平均频率，在后续无正跳沿的测量间隔内分布最后的值 ( $f \geq 1 \text{ mHz}$ )。这会延长积分时间。在这种情况下，使用最后测量的值除以无正跳沿的测量间隔数。

实例：如果最后测量的值是 12 000 mHz，则在三个测量间隔过后将输出值 4000 mHz。



## 可能出现错误指示的测量范围

积分时间	$f_{\text{最小}}$ /绝对误差	$f_{\text{最大}}$ /绝对误差	$f_{\text{最大}}$ /绝对误差	$f_{\text{最大}}$ /绝对误差
10 s	0.25 Hz/1 mHz	10 kHz/2 Hz	30 kHz/5 Hz	60 kHz/10 Hz
1 s	2.5 Hz/1 mHz	10 kHz/2 Hz	30 kHz/5 Hz	60 kHz/10 Hz
0.1 s	25 Hz/2 mHz	10 kHz/2 Hz	30 kHz/5 Hz	60 kHz/10 Hz
0.01 s	250 Hz/100 mHz	10 kHz/6 Hz	30 kHz/10 Hz	60 kHz/20 Hz

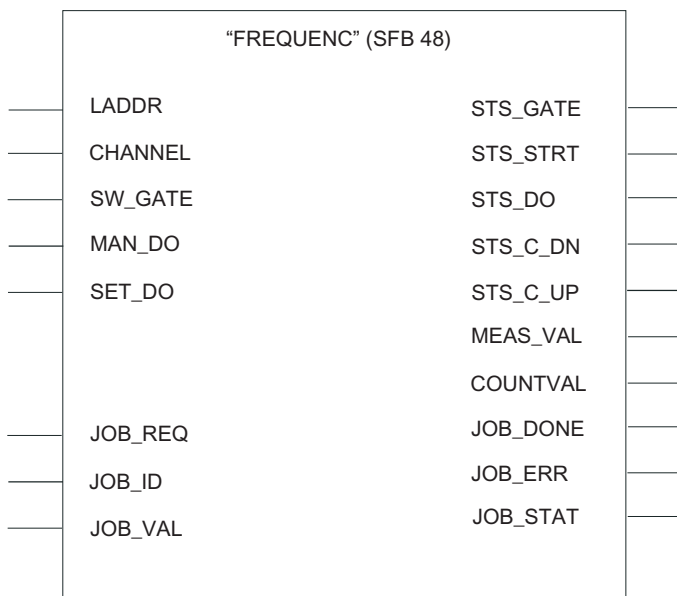
### 5.6.2 通过用户程序控制频率计数器

#### SFB FREQUENC 的功能范围

要从用户程序控制频率计数器，请使用 **SFB COUNT (SFB 48)**。

可使用以下功能：

- 通过软件门 SW\_GATE 启动/停止
- 启用/控制输出 DO
- 读出状态位
- 读出当前测量值
- 用于读取和写入内部频率计数寄存器的作业



## 输入参数

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
LADDR	WORD	0	在“HW Config”中指定的子模块 I/O 地址。 如果输入和输出地址不相同，则必须指定两者中较低地址。	依 CPU 而定	300 (十六进制)
CHANNEL	INT	2	通道号： CPU 312C CPU 313C CPU 313C-2 DP/PtP CPU 314C-2 DP/PtP	0-1 0-2 0-3	0
SW_GATE	BOOL	4.0	软件门 用于启动/停止频率测量	TRUE/FALSE	FALSE
MAN_DO	BOOL	4.1	启用输出的手动控制	TRUE/FALSE	FALSE
SET_DO	BOOL	4.2	控制输出	TRUE/FALSE	FALSE

## 说明

如果您已通过参数分配窗口将“输出特性”参数设置为“不比较”，则将进行以下操作：

- 将输出切换为标准输出。
- SFB 输入参数 MAN\_DO 和 SET\_DO 无效。
- 状态位 STS\_DO 保持复位状态。

## 不与块互连的输入参数（静态局部数据）

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
RES_STS	BOOL	32.2	复位状态位 复位状态位 STS_CMP、STS_OFLW 和 STS_UFLW。 必须调用 SFB 两次才能复位状态位。	TRUE/FALSE	FALSE

5.6 频率测量功能描述

输出参数

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
STS_GATE	BOOL	12.0	内部门状态	TRUE/FALSE	FALSE
STS_STRT	BOOL	12.1	硬件门状态 (启动输入)	TRUE/FALSE	FALSE
STS_DO	BOOL	12.2	输出状态	TRUE/FALSE	FALSE
STS_C_DN	BOOL	12.3	向下计数状态 始终指示最后的计数方向。在第一次调用 SFB 之后, STS_C_DN 的值为 FALSE。	TRUE/FALSE	FALSE
STS_C_UP	BOOL	12.4	向上计数状态 始终指示最后的计数方向。在第一次调用 SFB 之后, STS_C_UP 的值为 TRUE。	TRUE/FALSE	FALSE
MEAS_VAL	DINT	14	当前频率值	0 到 $2^{31} - 1$	0
COUNTVAL	DINT	18	当前计数值 每当打开内部门时都从 0 开始计数。	$-2^{31}$ 到 $2^{31} - 1$	0

不与块互连的输出参数 (静态局部数据)

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
STS_CMP	BOOL	26.3	测量结束状态 *) 在积分时间结束后更新测量值。在此处, 通过状态位 STS_CMP 发出测量结束信号	TRUE/FALSE	FALSE
STS_OFLOW	BOOL	26.5	上溢状态 *)	TRUE/FALSE	FALSE
STS_UFLOW	BOOL	26.6	下溢状态 *)	TRUE/FALSE	FALSE

\*: 使用 RES\_STS 复位

### 5.6.3 为频率测量读写请求接口

#### 引言

可以使用作业接口读/写频率寄存器。

#### 要求

最后一个作业必须完成 (JOB\_DONE = TRUE)。

#### 步骤

- 分配以下输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_REQ	BOOL	4.3	作业请求（正跳沿）	TRUE/FALSE	FALSE
JOB_ID	WORD	6	作业号：		0
			不带有功能的作业	00（十六进制）	
			写入下限	01（十六进制）	
			写入上限	02（十六进制）	
			写入积分时间	04（十六进制）	
			读下限	81（十六进制）	
			读上限	82（十六进制）	
读积分时间	84（十六进制）				
JOB_VAL	DINT	8	写入作业的值	-2 <sup>31</sup> 到 +2 <sup>31</sup> -1	0

- 调用 SFB。

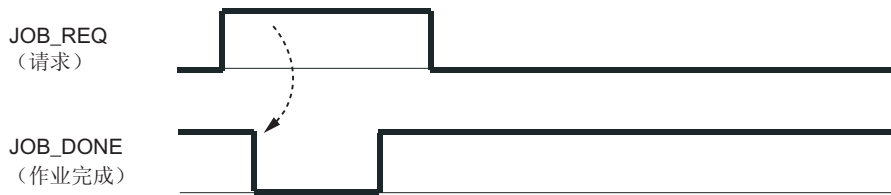
SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_DONE	BOOL	22.0	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	BOOL	22.1	错误作业	TRUE/FALSE	FALSE
JOB_STAT	WORD	24	作业错误编号	0 到 FFFF （十六进制）	0

5.6 频率测量功能描述

- 调用 SFB 之后将立即处理作业。在一个 SFB 循环时间内 **JOB\_DONE** 被设置为 **FALSE**。
  - 如果出现错误，则 **JOB\_ERR = TRUE**。随后将在 **JOB\_STAT** 中指出确切的出错原因。
  - 可以用 **JOB\_DONE = TRUE** 来启动新作业。
  - 仅用于读取作业：从背景数据块读取参数 **JOB\_OVAL** 的当前值。

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
<b>JOB_OVAL</b>	DINT	28	读取作业的输出值	-2 <sup>31</sup> 到 2 <sup>31</sup> -1	0



JOB\_VAL 允许的值范围

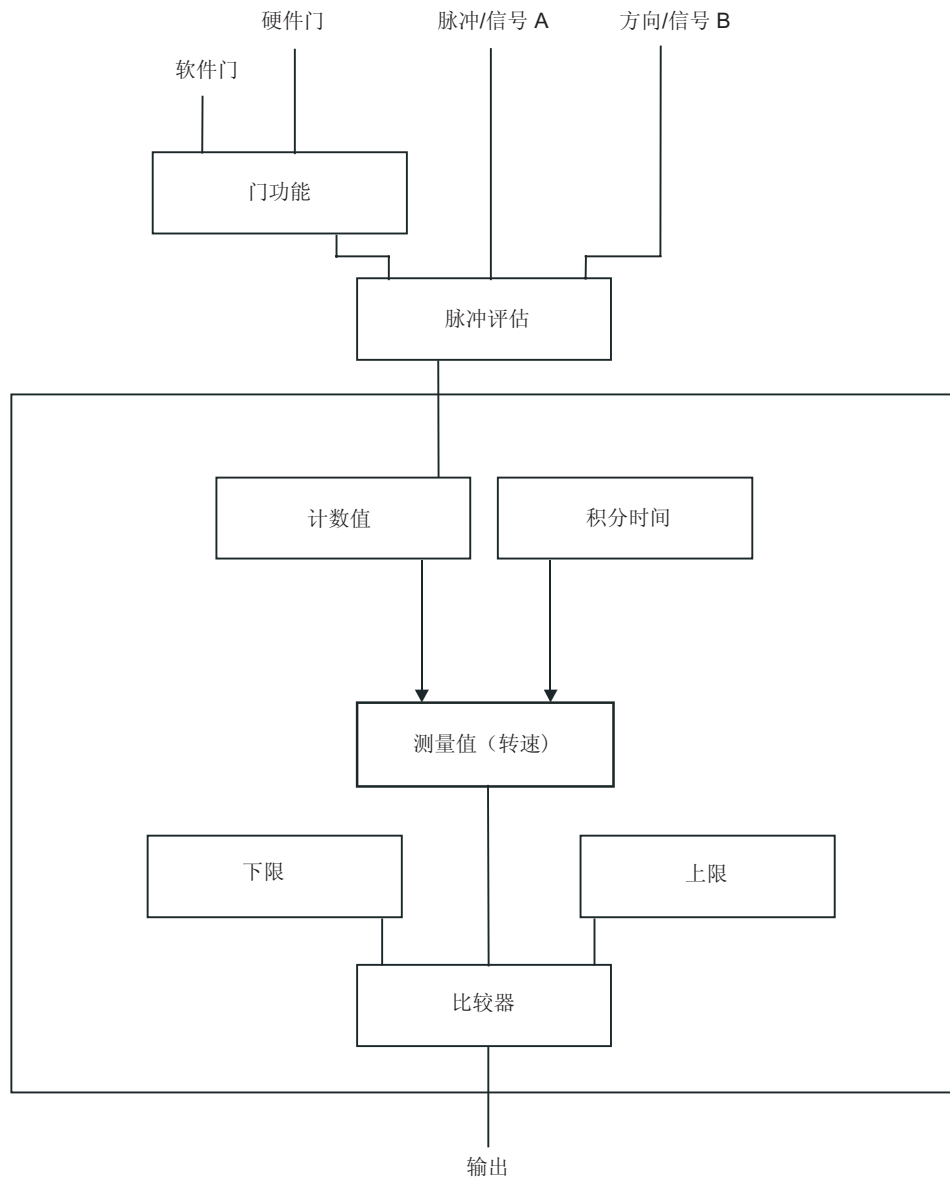
作业	有效值范围
写入下限 下限必须小于上限。	<ul style="list-style-type: none"> <li>• CPU 312C: 0 到 9 999 999 mHz</li> <li>• CPU 313C、CPU 313C-2 DP/PtP: 0 到 29 999 999 mHz</li> <li>• CPU 314C-2 DP/PtP: 0 到 59 999 999 mHz</li> </ul>
写入上限 上限必须大于下限。	<ul style="list-style-type: none"> <li>• CPU 312C: 1 到 10 000 000 mHz</li> <li>• CPU 313C、CPU 313C-2 DP/PtP: 1 到 30 000 000 mHz</li> <li>• CPU 314C-2 DP/PtP: 1 到 60 000 000 mHz</li> </ul>
写入积分时间	<ul style="list-style-type: none"> <li>• 10 到 10 000 ms</li> </ul>



### 5.6.4 频率计数器的功能块

#### 结构

下图显示了将在以下各章节中说明的各种功能块：



### 5.6.5 频率计数器输入

#### 脉冲/A

在此处连接要测量的信号或者编码器的轨迹 A。可通过单重判断模式连接编码器。

#### 方向/B

在此处连接方向信号或传感器的轨迹 B。可以在参数分配窗口指定对方向信号取反。

---

#### 说明

不会监视输入是否缺少脉冲。

---

#### 硬件门

可以用数字输入“硬件门”来控制频率测量。

### 5.6.6 频率测量的门功能

#### 基本信息

频率测量可使用两个门：

- 通过用户程序控制的**软件门**（SW 门）。  
可通过 SFB 参数 SW\_GATE 上的正跳沿打开软件门。复位该参数以关闭它。
- **硬件门**（HW 门）。可以在参数分配窗口分配硬件门。在数字输入“硬件门”处，通过正跳沿打开门，通过负跳沿关闭门。

#### 内部门

内部门表示 HW 门和 SW 门的逻辑 AND 连接。仅当 HW 门和 SW 门都打开时，测量循环才激活。状态位 STS\_GATE（状态内部门）显示此状态。如果未配置 HW 门，则仅有 SW 门设置是相关的。

#### 仅通过 SW 门来控制门

当 HW 门打开/关闭时，测量启动/停止。

#### 通过 SW 和 HW 门进行门控制

当两个门都打开时，测量开始。当其中一个门关闭时，测量停止。

## 5.6.7 频率测量输出的反应

### 下限/上限

在 CPU 中，可以存储分配给数字输出和硬件中断的一个上限和一个下限。可以根据计数值和上/下限启用数字输出。

可以在参数分配窗口中设置限制值，并可以在用户程序中通过 SFB 的作业接口写入（JOB\_ID = 01/02 [十六进制]）和读取（JOB\_ID = 81/82 [十六进制]）限制值。

### 数字输出的特性

可以在参数分配窗口中指定以下特性：

- 不比较
- 频率超出限制
- 频率低于下限
- 频率高于上限

#### 不比较

切换输出作为标准输出。

SFB 输入参数 MAN\_DO 和 SET\_DO 无效。

状态位 STS\_DO 保持复位状态。

#### 所有其它设置

可以手动或者用比较器控制输出：

- 手动控制

设置 SFB 参数 MAN\_DO 以切换到手动控制模式。然后可通过 SET\_DO 控制输出。

- 通过比较器控制

比较器在 MAN\_DO = FALSE 时执行控制。

比较器监视频率的上/下限值。

在达到比较条件时，比较器切换输出。

如果当前频率低于下限，则置位位 STS\_UFLW。

如果当前频率超过上限，则置位位 STS\_OFLW。

必须使用控制位 RES\_STS 复位这些位。

如果测量值复位后，频率仍然或者再次超出限制，则会再次置位对应的状态位。

---

### 说明

必须调用 SFB 两次，以使用 RES\_STS 复位该状态位。

---

### 5.6.8 频率测量和硬件中断

#### 设置硬件中断

在参数分配窗口中启用硬件中断并指定触发硬件中断的事件：

- 当 SW 门关闭时打开 HW 门
- 当 SW 门打开时关闭 HW 门
- 超出上限
- 超出下限
- 测量结束

#### 也参见

组态硬件中断（页码 227）

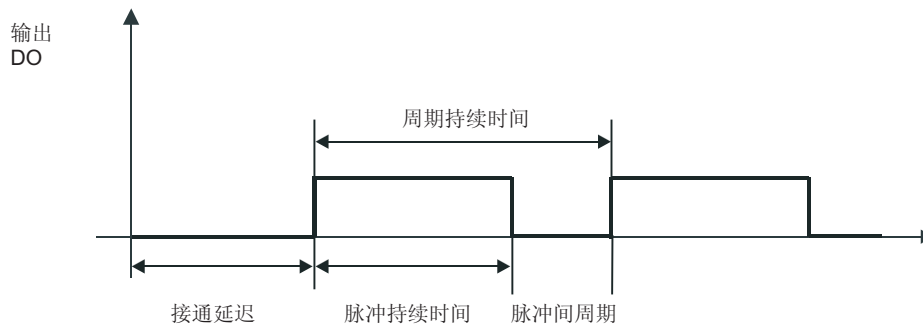
## 5.7 脉冲宽度调制功能说明

### 5.7.1 脉冲宽度调制的步骤

#### 基本信息

CPU 用对应的脉冲/间歇比将指定的输出值 (OUTP\_VAL) 转换为脉冲串（脉冲宽度调制）。在指定的接通延迟过后，在数字输出 DO（输出序列）处输出该脉冲串。

脉冲串的规范	
输出频率	0 到 2.5 kHz
最小脉冲宽度	200 $\mu$ s
脉冲间歇精度 接通延迟精度	$\pm$ (脉冲长度 $\times$ 100 ppm) $\pm$ 100 $\mu$ s ppm = 百万分率 0 到 250 $\mu$ s  在相同的脉冲宽度/间歇期间，除修改值外，如果最多只更改了一个其它的参数，则可以保持脉冲间歇的精度。如果修改了多个参数，则脉冲宽度/间歇可能比上述精度长或短一倍。



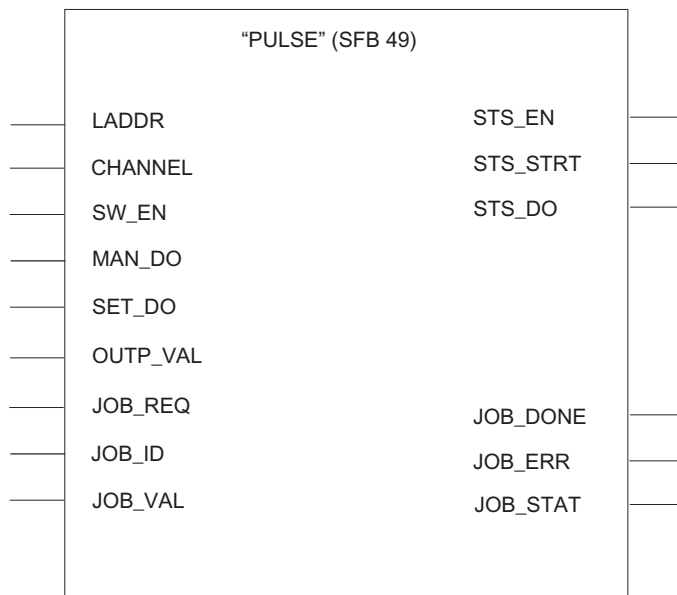
### 5.7.2 通过用户程序控制脉冲宽度调制

#### 控制功能块

要在用户程序中控制脉冲宽度调制，请使用 **SFB PULSE (SFB 49)**。

可使用下列功能：

- 通过软件门 SW\_EN 启动/停止
- 启用/控制输出 DO
- 读取状态位
- 输入输出值
- 请求读/写寄存器



表格 5-1 输入参数

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
LADDR	WORD	0	在“HW Config”中指定的子模块 I/O 地址。 如果 I/O 地址不相同，必须指定两者中的较低一个。	CPU 特定	300 (十六进制)
CHANNEL	INT	2	通道号：CPU 312C CPU 313C CPU 313C-2 DP/PtP CPU 314C-2 DP/PtP	0-1 0-2 0-3	0
SW_EN	BOOL	4.0	软件门 启动/停止数据输出	TRUE/FALSE	FALSE
MAN_DO	BOOL	4.1	启用输出的手动控制	TRUE/FALSE	FALSE
SET_DO	BOOL	4.2	输出控制	TRUE/FALSE	FALSE
OUTP_VAL	INT	6.0	指定缺省输出值： • 千分率形式 • S7 模拟值 如果指定一个大于 1,000 或 27,648 的值，CPU 会将该值限制为 1,000 或 27,648	0 到 1000 0 到 27648	0

表格 5-2 输出参数

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
STS_EN	BOOL	16.0	启用状态	TRUE/FALSE	FALSE
STS_STRT	BOOL	16.1	硬件门状态（启动输入）	TRUE/FALSE	FALSE
STS_DO	BOOL	16.2	输出状态	TRUE/FALSE	FALSE

5.7 脉冲宽度调制功能说明

5.7.3 为脉冲宽度调制读写请求接口

说明

可使用作业接口读/写寄存器。

要求

最后一个作业必须完成 (JOB\_DONE = TRUE)。

步骤

- 分配以下输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_REQ	BOOL	8	作业请求（正跳沿）	TRUE/FALSE	FALSE
JOB_ID	WORD	10	作业号： • 不带功能的作业 • 写入周期 • 写入接通延迟 • 写入最小脉冲持续时间 • 读取周期 • 读取接通延迟 • 读取最小脉冲持续时间	00（十六进制） 01（十六进制） 02（十六进制） 04（十六进制） 81（十六进制） 82（十六进制） 83（十六进制）	0
JOB_VAL	DINT	12	写入作业的值	-2 <sup>31</sup> 到 +2 <sup>31</sup> -1	0

- 调用 SFB。

SFB 的输出参数提供了以下信息：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_DONE	BOOL	16.3	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	BOOL	16.4	错误作业	TRUE/FALSE	FALSE
JOB_STAT	WORD	18	作业错误编号	0 到 FFFF （十六进制）	0

- 调用 SFB 之后立即处理作业。在一个 SFB 周期时间内，JOB\_DONE 被置位为 FALSE。
- 如果出现错误，则 JOB\_ERR = TRUE。然后将在 JOB\_STAT 中指出确切的出错原因。
- 可以用 JOB\_DONE = TRUE 来启动新作业。



- 仅用于读取作业：从背景数据块读取参数 JOB\_OVAL 的当前值。

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_OVAL	DINT	20	读取作业的输出值	$-2^{31}$ 到 $2^{31}-1$	0



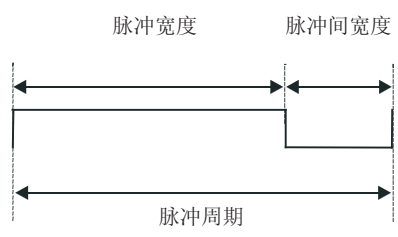
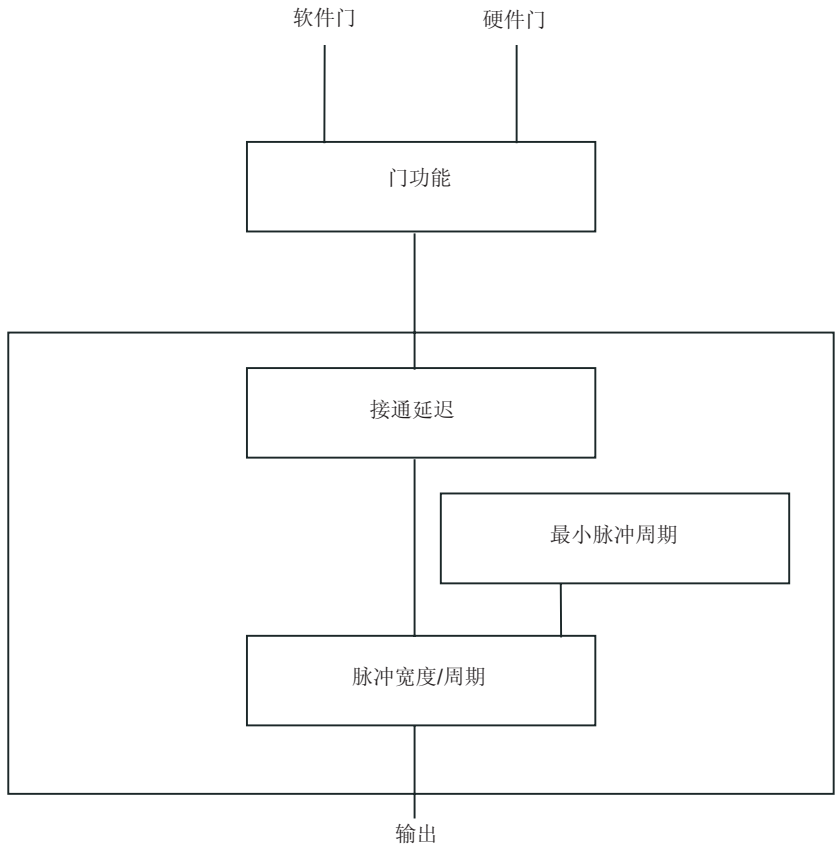
### JOB\_VAL 允许的值范围

作业	有效值范围	
写入周期	• 时基 0.1 ms:	• 4 到 65535
	• 时基 1 ms:	• 1 到 65535
写入接通延迟	• 0 到 65535	
写入最小脉冲持续时间	• 时基 0.1 ms:	• 2 到周期/2
	• 时基 1 ms:	• 0 到周期/2 (0 = 0.2 ms)

### 5.7.4 脉冲宽度调制功能块

#### 结构

下图显示了将在以下各章节中说明的各种功能块：



## 5.7.5 脉冲宽度调制的门功能

### 基本信息

有两个门可用于脉冲宽度调制操作：

- 通过用户程序控制的**软件门**（SW 门）。  
可通过 SFB 参数 SW\_EN 上的正跳沿打开软件门。复位该参数可关闭它。
- **硬件门**（HW 门）。可以在参数分配窗口分配硬件门。通过数字输入“硬件门”控制它。

### 内部门

内部门用于启动和停止脉冲宽度调制。

内部门表示 HW 门和 SW 门的逻辑 AND 连接。状态位 STS\_EN 指示内部门的状态。

启用内部门之后，接通延迟将会启动。脉冲串将在接通延迟时间结束时输出。如果置位了使能信号，该输出序列将无限地运行。

### 仅通过 SW 门来控制门

当 SW 门打开/关闭时，脉冲宽度调制启动/停止。

### 通过 SW 和 HW 门控制门

- 仅当在第一步打开 SW 门，并且第二步在 HW 门输入处生成正跳沿时，才有可能启动脉冲宽度调制。

要求	操作
SW 门打开	HW 门 0 -> 1

- 仅可以在 SW 门的负跳沿时停止脉冲宽度调制。HW 门可以有任意状态。

要求	操作
无，任何 HW 门状态	SW 门 1 -> 0

### 5.7.6 设置脉冲串的参数

参数，它们的设置和控制可能性

参数	可通过以下方式组态： 参数分配窗口	通过以下方式控制： SFB
时基	是	-
输出格式	是	-
输出值	-	写
周期	是	读/写
接通延迟	是	读/写
最小脉冲宽度	是	读/写

#### 时基

使用时基指定接通延迟的分辨率和值范围、周期和最小脉冲宽度。

#### 输出格式

在输出格式参数中，声明输出值的范围：

输出格式	值范围
千分率	0 到 1000
S7 模拟值	0 到 27648

#### 输出值

将输出值声明为 SFB 的输入参数 **OUTP\_VAL**。

CPU 用指定的输出值计算脉冲宽度：

输出格式	脉冲宽度
千分率	$(\text{输出值} / 1000) \times \text{周期}$
S7 模拟值	$(\text{输出值} / 27648) \times \text{周期}$

如果在输出脉冲序列期间更改输出值，CPU 将立即计算新的周期并且相应地切换输出。可通过这种操作来延长或缩短一个周期的长度：

- 如果在信号低且新输出值小于旧输出值时更改，将会由于新的脉冲间周期变得更长，因而延长一个循环中的周期长度。
- 如果在信号低且新输出值大于旧输出值时更改，将会由于新的脉冲间周期变得更短，因而缩短一个循环中的周期长度。
- 如果在信号高且新输出值小于旧输出值时更改，将会由于新的低信号变得更长，因而延长一个循环中的周期长度。
- 如果在信号高并且新的输出值大于旧的输出值时更改，则周期保持不变。

## 周期

用周期定义输出脉冲/脉冲间序列的长度。

周期 = 时基 × 指定的数值

周期必须至少是最小脉冲宽度长度的二倍。

如果在输出脉冲串期间更改输出值，CPU 将立即计算新的脉冲/脉冲间周期并且相应地切换输出。可通过这种操作来延长或缩短一个周期的长度：

- 如果在信号低且新的周期比前一个周期更短时更改，则生成的周期会比前一个周期要短而比新的周期要长。
- 如果在信号低且新的周期比前一个周期更长时更改，则生成的周期会比前一个周期要长而比新的周期要短。
- 如果在信号高且新的周期比前一个周期更短时更改，则生成的周期会比前一个周期要短而比新的周期要长。
- 如果在信号高且新的周期比前一个周期更长时更改，则生成的周期会比前一个周期要长而比新的周期要短。

## 接通延迟

表示输出序列的开始和输出第一个脉冲之间的时间间隔

接通延迟 = 时基 × 指定的数值

如果在接通延迟处于激活状态时更改它，将立即应用新设置：

- 如果新的接通延迟比前一个接通延迟更短，则一次可生成一个比前一个信号更短但比新信号更长的接通延迟。
- 如果新接通延迟比先前的延迟时间要长，则使用新的接通延迟。

## 最小脉冲宽度

比最小脉冲宽度更短的所有输出低/高信号均会受到抑制。

最小脉冲宽度 = 时基 × 指定的数值

如果在输出代码序列期间更改最小脉冲宽度，将立即应用新值：

- 如果在信号低且脉冲间宽度比新的最小脉冲宽度更短时应用更改，则会将输出置为“1”。
- 如果在信号低且宽度比新的最小脉冲宽度更长时应用更改，则会输出脉冲间宽度。
- 如果在信号高且脉冲宽度比新的最小脉冲宽度更短时应用更改，则会将输出置为“0”。
- 如果在信号高且脉冲宽度比新的最小脉冲宽度更长时应用更改，则会输出脉冲。

	时基: 0.1 ms	时基: 1 ms
周期	4 到 65535	1 到 65535
接通延迟	0 到 65535	0 到 65535
最小脉冲宽度	2 到周期/2	0 到周期/2 (0 = 0.2 ms)

### 5.7.7 脉冲宽度调制输出的反应

#### 引言

本节说明了数字输出的特性。

可手动控制输出，或用其输出脉冲串。

#### 手动控制

设置 SFB 参数 **MAN\_DO** 以切换到手动控制模式。然后可通过 **SET\_DO** 控制输出。

#### 脉冲串的输出

**MAN\_DO = FALSE** 可用于输出脉冲串。

### 5.7.8 脉冲宽度调制和硬件中断

#### 设置硬件中断

在参数分配窗口中启用硬件中断并指定触发硬件中断的事件：

- 当 SW 门关闭时打开 HW 门。

#### 也参见

组态硬件中断（页码 227）

## 5.8 错误处理和中断

### 5.8.1 错误显示

#### 基本信息

错误的指示方式

- 系统功能块 (SFB) 的错误消息
- 诊断中断

可对指定事件触发硬件中断。

### 5.8.2 系统功能块 (SFB) 的错误消息

#### 概述

SFB 指出了下表中列出的错误。

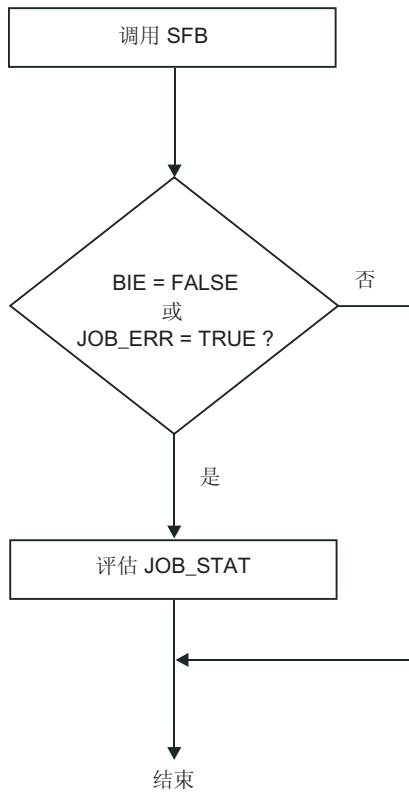
错误类型	通过 SFB 参数显示错误	在 SFB 参数中显示错误编号
作业错误	JOB_ERR = TRUE	JOB_STAT
系统错误	BIE = FALSE	JOB_STAT

解释/执行作业期间出现作业错误。出现错误时，JOB\_ERR 参数将被置位为 TRUE。

如果发生基本参数分配错误，则会触发系统错误，例如“错误的操作模式”。通过 BIE = FALSE 指示系统错误。

参数 JOB\_STAT 较详细地描述了出错原因。在『错误列表（页码 235）』一节中列出了可能的错误编号。

错误判断





### 5.8.3 组态诊断中断

#### 使用诊断中断

例如，出现以下错误时

- 参数分配错误（模块数据）和
- “硬件错误信号丢失”

可触发诊断中断。诊断中断随同进入和离开的错误事件一起显示。

在用户程序中，可借助诊断中断立即对错误作出响应。

#### 步骤

1. 在参数分配窗口中启用诊断中断。（中断选择：诊断或诊断/处理）
2. 可将诊断中断 OB (OB 82) 合并到用户程序中。

#### 通过诊断中断对错误作出响应

- 当前操作不受诊断中断影响。
- CPU 操作系统将在用户程序中调用 OB 82。

---

#### 说明

如果未装载相应的 OB，则在触发中断后，CPU 会切换为 STOP。

---

- CPU 将打开 SF LED 指示灯。
- 在 CPU 的诊断缓冲区中将错误报告为“进入”。清除所有未决错误前，不会将错误指示为“离开”。

### 如何在用户程序中判断诊断中断

触发诊断中断后，可判断 OB 82 以检查哪个诊断中断还未处理。

- 如果在 OB 82 的字节 6 + 7 (OB 82\_MDL\_ADDR) 中输入了子模块地址，则诊断中断是由 CPU 中的计数器触发的。
- 只要队列中有错误，就会置位 OB 82 中字节 8 的位 0（故障模块）。
- 队列中的所有错误均报告为“离开”后，将复位 OB 82 中字节 8 的位 0。
- 可判断字节 8 和 11，以确定出错原因。

OB 82, 字节 8	说明
位 0	故障模块
位 1	-
位 2	-
位 3	-
位 4	-
位 5	-
位 6	-
位 7	组态错误

OB 82, 字节 11	说明
位 0	-
位 1	-
位 2	-
位 3	-
位 4	-
位 5	-
位 6	硬件中断丢失
位 7	-

### 硬件中断丢失

由于启用了硬件中断，如果在清除上一个中断前再次出现触发事件的中断，CPU 会报告“丢失硬件中断”错误。

## 5.8.4 组态硬件中断

### 使用硬件中断

可对特定事件触发硬件中断。借助于硬件中断，您可以在用户程序中对事件进行即时响应。

### 步骤

1. 在参数分配窗口“基本参数”中启用硬件中断。（中断选择：硬件或诊断/硬件）
2. 在相应的“计数”、“频率计数”或“脉冲宽度调制”参数分配窗口中，启用触发硬件中断的事件。
3. 可在用户程序中集成硬件中断 OB (OB 40)。

### 对硬件中断作出响应

CPU 操作系统将在用户程序中调用 OB 40。

---

#### 说明

如果未装载相应的 OB，则在触发中断时，CPU 会切换为 STOP。

---

### 用户程序中硬件中断的判断

触发硬件中断后，可判断 OB 40 以检查哪些硬件中断还未处理。

- 如果在 OB 40 的字节 6 + 7 (OB 40\_MDL\_ADDR) 中输入了子模块的地址，则硬件中断是由 CPU 中的计数器触发的。
- 要确定确切的原因，请判断 DWord OB 40\_POINT\_ADDR 的字节 8 到 11。

#### 计数

OB 40, 字节 8	说明
位 0	通道 0: HW 门打开
位 1	通道 0: HW 门关闭
位 2	通道 0: 上溢/下溢
位 3	通道 0: 比较器已响应
位 4	通道 1: HW 门打开
位 5	通道 1: HW 门关闭
位 6	通道 1: 上溢/下溢
位 7	通道 1: 比较器已响应

5.8 错误处理和中断

OB 40, 字节 9	说明
位 0	通道 2: HW 门打开
位 1	通道 2: HW 门关闭
位 2	通道 2: 上溢/下溢
位 3	通道 2: 比较器已响应
位 4	通道 3: HW 门打开
位 5	通道 3: HW 门关闭
位 6	通道 3: 上溢/下溢
位 7	通道 3: 比较器已响应

OB 40, 字节 10	说明
位 0	通道 0: 出现计数跳沿
位 1	通道 0: -
位 2	通道 0: -
位 3	通道 0: -
位 4	通道 1: 出现计数跳沿
位 5	通道 1: -
位 6	通道 1: -
位 7	通道 1: -

OB 40, 字节 11	说明
位 0	通道 2: 出现计数跳沿
位 1	通道 2: -
位 2	通道 2: -
位 3	通道 2: -
位 4	通道 3: 出现计数跳沿
位 5	通道 3: -
位 6	通道 3: -
位 7	通道 3: -

## 频率计数

OB 40, 字节 8	说明
位 0	通道 0: HW 门打开
位 1	通道 0: HW 门关闭
位 2	通道 0: 超出频率的上/下限
位 3	通道 0: 测量结束
位 4	通道 1: HW 门打开
位 5	通道 1: HW 门关闭
位 6	通道 1: 超出频率的上/下限
位 7	通道 1: 测量结束

OB 40, 字节 9	说明
位 0	通道 2: HW 门打开
位 1	通道 2: HW 门关闭
位 2	通道 2: 超出频率的上/下限
位 3	通道 2: 测量结束
位 4	通道 3: HW 门打开
位 5	通道 3: HW 门关闭
位 6	通道 3: 超出频率的上/下限
位 7	通道 3: 测量结束

OB 40, 字节 10 和 11: 未分配

5.9 安装实例

脉冲宽度调制

OB 40, 字节 8	说明
位 0	通道 0: HW 门打开
位 1	通道 0: -
位 2	通道 0: -
位 3	通道 0: -
位 4	通道 1: HW 门打开
位 5	通道 1: -
位 6	通道 1: -
位 7	通道 1: -

OB 40, 字节 9	说明
位 0	通道 2: HW 门打开
位 1	通道 2: -
位 2	通道 2: -
位 3	通道 2: -
位 4	通道 3: HW 门打开
位 5	通道 3: -
位 6	通道 3: -
位 7	通道 3: -

OB 40, 字节 10 和 11: 未分配

## 5.9 安装实例

### 使用实例

可在文档附带的 CD-ROM 中找到实例（程序和说明）。也可从 Internet 下载。项目由具有不同复杂度和针对性的多个带注释的 S7 程序组成。

CD 上的 Readme.wri 说明了安装实例的方法。安装完成后，实例即存储在 ...STEP7\EXAMPLES\ZDI26\_02\_TF\_\_\_\_31xC\_Cnt 目录下。

## 5.10 规范

### 5.10.1 功能

#### 计数

	CPU 312C	CPU 313C, CPU 313C-2 DP/PtP	CPU 314C-2 DP/PtP
最大频率 (轨迹 A/脉冲、轨迹 B/方向、硬件门和锁存器)	10 kHz	30 kHz	60 kHz
最小脉冲宽度/最小脉冲间宽度	48 $\mu$ s	16 $\mu$ s	8 $\mu$ s
最大电缆长度 (最大计数频率时)	100 m	100 m	50 m
计数范围	- 2 147 483 648 ( $-2^{31}$ ) 到 + 2 147 483 647 ( $2^{31} - 1$ )		

#### 频率测量

表格 5-3 频率范围

	CPU 312C	CPU 313C、 CPU 313C-2 DP/PtP	CPU 314C-2 DP/PtP
频率范围 (轨迹 A/脉冲、轨迹 B/方向和硬件门)	0 到 10 kHz	0 到 30 kHz	0 到 60 kHz
最小脉冲宽度/最小脉冲间宽度	48 $\mu$ s	16 $\mu$ s	8 $\mu$ s
最大电缆长度 (最大计数频率时)	100 m	100 m	50 m

表格 5-4 可能出现错误指示的测量范围

积分时间	$f_{\text{最小}}$ /绝对错误	$f_{\text{最大}}$ /绝对错误	$f_{\text{最大}}$ /绝对错误	$f_{\text{最大}}$ /绝对错误
10 s	0.25 Hz/1 mHz	10 kHz/2 Hz	30 kHz/5 Hz	60 kHz/10 Hz
1 s	2.5 Hz/1 mHz	10 kHz/2 Hz	30 kHz/5 Hz	60 kHz/10 Hz
0.1 s	25 Hz/2 mHz	10 kHz/2 Hz	30 kHz/5 Hz	60 kHz/10 Hz
0.01 s	250 Hz/100 mHz	10 kHz/6 Hz	30 kHz/10 Hz	60 kHz/20 Hz

5.10 规范

脉冲宽度调制

脉冲串规范	
输出频率	0 到 2.5 kHz
最小脉冲宽度	200 $\mu$ s
脉冲间歇精度 接通延迟精度	<p>+/- (脉冲长度 x 100 ppm) +/- 100 <math>\mu</math>s ppm = 百万分率 0 到 250 <math>\mu</math>s</p> <p>在相同的脉冲宽度/间歇期间，除修改值外，如果最多只更改了一个其它参数，则可以保持脉冲间歇的精度。如果修改了多个参数，则脉冲宽度/间歇可能比上述精度长或短一倍。</p>

	CPU 312C	CPU 313C、 CPU 313C-2 DP/PtP	CPU 314C-2 DP/PtP
过滤器频率 (硬件门)	10 kHz	30 kHz	60 kHz
最小脉冲宽度	48 $\mu$ s	16 $\mu$ s	8 $\mu$ s
最大电缆长度	100 m	100 m	50 m



## 5.10.2 增量编码器

### 可连接的增量编码器

支持非对称 24 V 增量编码器，该编码器可传送两个电相位差为 90° 的方波脉冲。

### 信号判断

#### 增量

一个增量标识两个编码器轨迹信号 A 和 B 的信号周期。该值在编码器的铭牌和/或技术规范中指定。

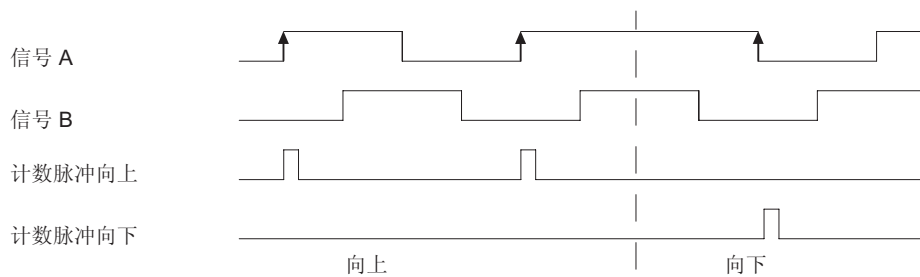
#### 轨迹 A 和 B 上的跳沿

CPU 可对轨迹的跳沿进行计数。通常，CPU 仅判断轨迹 A 上的跳沿（单重判断）。使用多重判断可以获得更高的分辨率。在参数分配窗口中，可指定轨迹的单重判断、双重判断或四重判断。

仅当使用配有两个轨迹 A 和 B（相位差为 90°）的非对称 24 V 增量编码器时，才能使用多重判断。

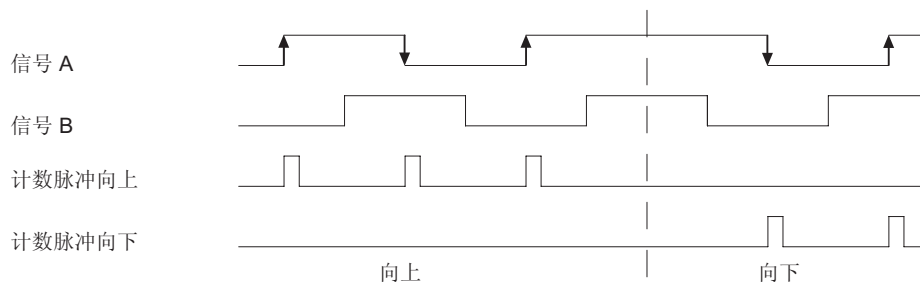
#### 单重判断

在单重判断模式下，仅判断轨迹 A 的一个跳沿，即在 A 的正跳沿和 B 处的低电平上测量向上计数脉冲，在 A 的正跳沿和 B 处的高电平上测量向下计数脉冲。



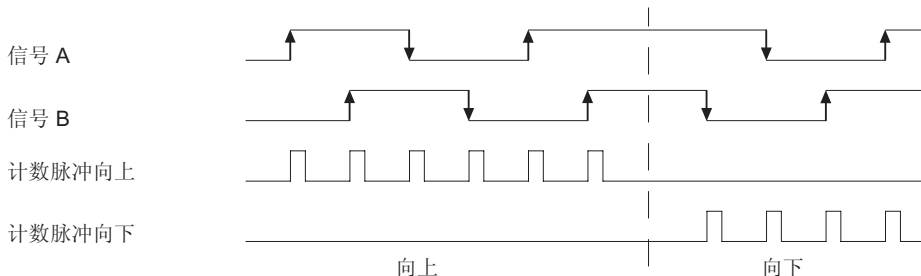
#### 双重判断

在双重判断模式下，判断轨迹 A 的正跳沿和负跳沿。轨迹 B 的电平决定了是产生向上计数脉冲还是向下计数脉冲。



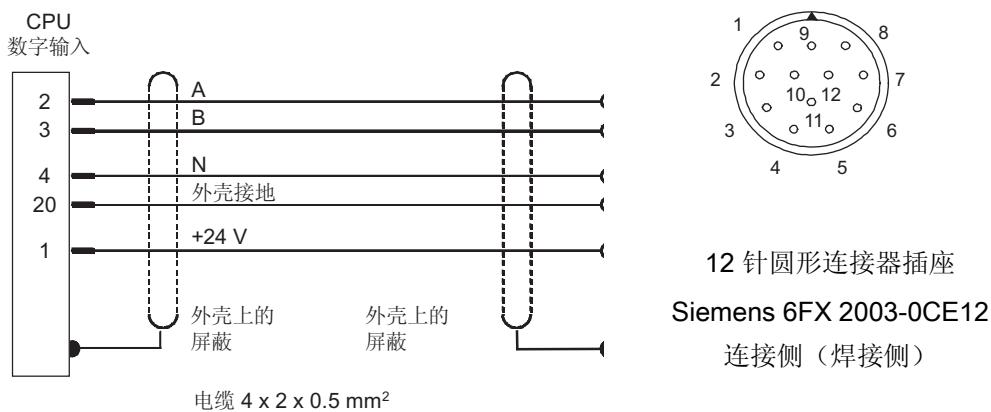
**四重判断**

在四重判断模式下，判断轨迹 A 和 B 的正跳沿和负跳沿。轨迹 A 和 B 的逻辑电平决定了是产生向上计数脉冲还是向下计数脉冲。



**增量编码器 Siemens 6FX 2001-4 (Up = 24 V; HTL) 的接线图**

下图显示了增量编码器 Siemens 6FX 2001-4xxxx (Up = 24 V; HTL) 的接线图：



### 5.10.3 错误列表

#### 基本信息

在下表中，可找到 SFB 输出 JOB\_STAT 的错误 ID 说明。错误 ID 由事件类别和编号组成。

#### 作业错误

表格 5-5 事件类别 01 (01H): “计数, SFB 参数 (SFB 47) 中的参数分配错误”

事件 ID	事件文本	补救措施
(01)21H	比较值过低	有关有效值的信息, 请参见在线帮助或用户手册
(01)22H	比较值过高	
(01)31H	滞后过窄	
(01)32H	滞后过宽	
(01)41H	脉冲宽度过短	
(01)42H	脉冲宽度过长	
(01)51H	装载值过低	
(01)52H	装载值过高	
(01)61H	计数值过低	
(01)62H	计数值过高	
(01)FFH	作业 ID 无效	

表格 5-6 事件类别 02 (02HH): “频率测量, SFB 参数 (SFB 48) 中的组态错误”

事件 ID	事件文本	补救措施
(02)21H	积分时间过低	有关有效值的信息, 请参考在线帮助或用户手册
(02)22H	积分时间过高	
(02)31H	频率下限过低	
(02)32H	频率下限过高	
(02)41H	频率上限过低	
(02)42H	频率上限过高	
(02)FFH	作业 ID 无效	

5.10 规范

表格 5-7 事件类别 04 (04H): “脉冲宽度调制, SFB 参数 (SFB 49) 中的组态错误”

事件 ID	事件文本	补救措施
(04)11H	脉冲周期过短	有关有效值的信息, 请参考在线帮助或用户手册
(04)12H	脉冲周期过长	
(04)21H	接通延迟过短	
(04)22H	接通延迟过长	
(04)31H	最小脉冲宽度过短	
(04)32H	最小脉冲宽度过长	
(04)FFH	作业 ID 无效	

系统错误

表格 5-8 事件类别 128 (80H): “全局 SFB 参数中的组态错误”

事件 ID	事件文本	补救措施
(80)01H	错误的操作模式或参数分配错误。	在“HW Config”中, 指定正确的操作模式或使用与设置的模式相匹配的 SFB。
(80)09H	通道号无效	有效通道号为: <ul style="list-style-type: none"> <li>• CPU 312C: 0-1</li> <li>• CPU 313C、CPU 313C-2 DP/PtP: 0-2</li> <li>• CPU 314C-2 DP/PtP: 0-3</li> </ul>
系统出错时, BIE 位被置位为 FALSE		

## 5.10.4 模块参数（概述）

### 引言

下表概述了可在参数分配窗口中设置的模块参数。

### 基本参数

参数	说明	值范围	缺省
中断选择	可在此选择技术功能应触发哪些中断。	<ul style="list-style-type: none"> <li>无</li> <li>诊断</li> <li>硬件</li> <li>诊断和硬件</li> </ul>	无

### 连续计数、仅计数一次和周期性计数

参数	说明	值范围	缺省
主计数方向	<ul style="list-style-type: none"> <li>无：没有计数范围的限制</li> <li>向上：限制向上计数范围。计数器从 0 或装载值开始，沿正方向计数，直至分配的结束值 -1，然后在下一个正编码器脉冲处跳回至装载值。</li> <li>向下：限制向下计数范围。计数器从分配的起始值或装载值开始，沿负方向计数到 1，然后在下一个负编码器脉冲处跳至起始值。</li> </ul>	<ul style="list-style-type: none"> <li>无</li> <li>向上（不连续计数）</li> <li>向下（不连续计数）</li> </ul>	无
结束值/起始值	<ul style="list-style-type: none"> <li>主计数方向向上的结束值</li> <li>主计数方向向下的起始值</li> </ul>	2 到 2147483647 ( $2^{31} - 1$ )	2147483647 ( $2^{31} - 1$ )
门功能	<ul style="list-style-type: none"> <li>取消计数操作： 将门关闭并重新启动时，会从装载值开始重新计数。</li> <li>停止计数操作： 门关闭时，计数即停止，当门再次打开时，将从上一个实际值开始重新计数。</li> </ul>	<ul style="list-style-type: none"> <li>取消计数</li> <li>停止计数</li> </ul>	取消计数
比较值	将计数值与比较值比较。请参阅参数“输出特性” <ul style="list-style-type: none"> <li>无主计数方向</li> <li>主计数方向向上</li> <li>主计数方向向下</li> </ul>	- $2^{31}$ 到 + $2^{31}-1$ - $2^{31}$ 到结束值 -1 1 到 + $2^{31}-1$	0
滞后	如果计数值在比较值范围内，则可使用滞后避免频繁的输出切换操作。 0 和 1 表示：关闭滞后。	0 到 255	0
最大频率： 计数信号/HW 门	CPU 312C	10、5、2、1 kHz	10 kHz
	CPU 313C、313C-2 DP/PtP	30、10、5、2、1 kHz	30 kHz
	CPU 314C-2 DP/PtP	60、30、10、5、2、1 kHz	60 kHz
最大频率： 锁存器	CPU 312C	10、5、2、1 kHz	10 kHz
	CPU 313C、313C-2 DP/PtP	30、10、5、2、1 kHz	10 kHz
	CPU 314C-2 DP/PtP	60、30、10、5、2、1 kHz	10 kHz

## 5.10 规范

参数	说明	值范围	缺省
信号判断	<ul style="list-style-type: none"> <li>计数和方向信号与输入相连接</li> <li>旋转编码器与输入相连接（单一、双重或四重判断）</li> </ul>	<ul style="list-style-type: none"> <li>脉冲/方向</li> <li>旋转编码器，单一</li> <li>旋转编码器，双重</li> <li>旋转编码器，四重</li> </ul>	脉冲/方向
HW 门	<ul style="list-style-type: none"> <li>是： 通过 SW 和 HW 门进行门控制</li> <li>否： 仅通过 SW 门进行门控制。</li> </ul>	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
反转计数方向	<ul style="list-style-type: none"> <li>是： 反转了“方向”输入信号。</li> <li>否： 未反转“方向”输入信号。</li> </ul>	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
输出特性	根据该参数置位输出和“比较器”(STS_CMP) 状态位。	<ul style="list-style-type: none"> <li>不比较</li> <li>计数值 <math>\geq</math> 比较值</li> <li>计数值 <math>\leq</math> 比较值</li> <li>比较值时刻的脉冲</li> </ul>	不比较
脉冲持续时间	通过设置“输出特性：比较值时刻的脉冲”，可指定输出信号的脉冲持续时间。仅可使用偶数值。	0 到 510 ms	0
输入数据的分配	您可以选择最大计数频率为 1 kHz 时，“计数”子模块的输入数据（I 数据）中可以读取计数值还是可以读取周期。如果最大计数频率大于 1 kHz，则仅可以读取“计数值”。	计数值 周期	计数值
时基	您可以指定最大计数频率为 1 kHz 时，是以 125 ns 为单位测量周期还是以 1 $\mu$ s 为单位测量周期。如果最大计数频率大于 1 kHz，则将不测量周期。	125 ns 1 $\mu$ s	125 ns
硬件中断： HW 门打开	软件门打开时，打开硬件门会产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断： HW 门关闭	软件门打开时，关闭硬件门可产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断： 达到比较值	达到比较器时会产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断： 上溢	上溢（超出计数上限）时会产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断： 下溢	下溢（低于计数下限）时会产生硬件中断。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
硬件中断： 计数跳沿	您可以选择在出现最大计数频率 1 kHz 时，是否在每个计数跳沿上产生硬件中断。如果最大计数频率大于 1 kHz，则不能选择此硬件中断。在每个计数跳沿产生硬件中断将导致计数频率较高时出现高 CPU 利用率。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否

## 频率计数

参数	说明	值范围	缺省
积分时间	测量进入脉冲的时间窗口。	1 到 10 000 ms	100
下限	将测量值与下限比较。低于下限时，将置位状态位“下溢”(STS_UFLW)。下限必须小于上限。	CPU 312C: 0 到 9 999 999 mHz CPU 313C, CPU 313C-2 DP/PtP: 0 到 29 999 999 mHz CPU 314C-2 DP/PtP: 0 到 59 999 999 mHz	0
上限	将测量值与上限比较。超出上限时，置位状态位“上溢”(STS_OFLW)。上限必须大于下限。	CPU 312C: 1 到 10 000 000 mHz CPU 313C, CPU 313C-2 DP/PtP: 1 到 30 000 000 mHz CPU 314C-2 DP/PtP: 1 到 60 000 000 mHz	CPU 312C: 10 000 000 mHz CPU 313C, CPU 313C-2 DP/PtP: 30 000 000 mHz CPU 314C-2 DP/PtP: 60 000 000 mHz
最大计数频率	CPU 312C	10、5、2、1 kHz	10 kHz
	CPU 313C、313C-2 DP/PtP	30、10、5、2、1 kHz	30 kHz
	CPU 314C-2 DP/PtP	60、30、10、5、2、1 kHz	60 kHz
输出测量值	如果测量频率的周期超过了分配的积分时间： • 对于“直接”频率，在积分时间结束时将输出“0”值。 • 对于“平均”频率，在后续的无跳沿的测量间隔内分配最后的值 ( $f \geq 1$ mHz)。这会延长积分时间。此时，将使用无跳沿的测量间隔数除以最后测量的值。	• 直接 • 平均	直接
信号判断	• 计数和方向信号与输入相连。 • 单重判断的旋转编码器与输入相连接。	• 脉冲/方向 • 旋转编码器，单一	脉冲/方向
反转计数方向	• 是： 反转了“方向”输入信号。 • 否： 未反转“方向”输入信号。	• 是 • 否	否
HW 门	• 是： 通过 SW 和 HW 门进行门控制。 • 否： 仅通过 SW 门进行门控制。	• 是 • 否	否
输出特性	将测量值与上下限比较。根据此参数切换输出。	• 不比较 • 超出限制 • 低于下限 • 高于上限	不比较

5.10 规范

参数	说明	值范围	缺省
硬件中断： HW 门打开	软件门打开时，打开硬件门会发生硬件中断。	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
硬件中断： HW 门关闭	如果在软件门打开时关闭硬件门，则会产生硬件中断。	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
硬件中断： 测量结束	测量结束时产生硬件中断。	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
硬件中断： 超出下限	低于下限时会产生硬件中断。	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
硬件中断： 超出上限	超出上限时产生硬件中断。	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否

脉冲宽度调制

参数	说明	值范围	缺省
输出格式	输出格式	<ul style="list-style-type: none"> <li>• 千分率</li> <li>• S7 模拟值</li> </ul>	<ul style="list-style-type: none"> <li>• 千分率</li> </ul>
时基	时基 <ul style="list-style-type: none"> <li>• 接通延迟</li> <li>• 周期</li> <li>• 最小脉冲持续时间</li> </ul>	<ul style="list-style-type: none"> <li>• 0.1 ms</li> <li>• 1.0 ms</li> </ul>	<ul style="list-style-type: none"> <li>• 0.1 ms</li> </ul>
接通延迟	介于输出序列的开始和脉冲输出之间的时间间隔。	0 - 65535	0
周期	根据脉冲持续时间和脉冲间周期定义输出序列的长度。	<ul style="list-style-type: none"> <li>• 时基 0.1 ms: 4 到 65535</li> <li>• 时基 1 ms: 1 到 65535</li> </ul>	20 000
最小脉冲持续时间	短于最小脉冲持续时间的输出脉冲/脉冲间周期会被抑制。 对于 1 ms 的时基和值 0，可在内部将最小脉冲持续时间设置为 0.2 ms。	<ul style="list-style-type: none"> <li>• 时基 0.1 ms: 2 到周期/2</li> <li>• 时基 1 ms: 0 到周期/2</li> </ul>	2
HW 门	<ul style="list-style-type: none"> <li>• 是： 通过 SW 和 HW 门进行门控制。</li> <li>• 否： 仅通过 SW 门进行门控制</li> </ul>	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
过滤器频率： HW 门	CPU 312C	10、5、2、1 kHz	10 kHz
	CPU 313C、313C-2 DP/PtP	30、10、5、2、1 kHz	30 kHz
	CPU 314C-2 DP/PtP	60、30、10、5、2、1 kHz	60 kHz
硬件中断： HW 门打开	软件门打开时，打开硬件门会发生硬件中断。	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否



## 5.10.5 SFB 的背景 DB

## SFB 47 “COUNT” 的参数

参数	声明	数据类型	地址（背景数据块）	说明	值范围	缺省
LADDR	IN	WORD	0	在“HW Config”中指定的子模块 I/O 地址。如果 I/O 地址不相同，必须指定两者中的较低一个。	CPU 特定	300 (十六进制)
CHANNEL	IN	INT	2	通道号:	0-1	0
				• CPU 312C	0-2	
				• CPU 313C, CPU 313C-2 DP/PtP • CPU 314C-2 DP/PtP	0-3	
SW_GATE	IN	BOOL	4.0	软件门 用于计数器启动/停止	TRUE/FALSE	FALSE
CTRL_DO	IN	BOOL	4.1	启用输出	TRUE/FALSE	FALSE
SET_DO	IN	BOOL	4.2	输出控制	TRUE/FALSE	FALSE
JOB_REQ	IN	BOOL	4.3	启动作业（正跳沿）	TRUE/FALSE	FALSE
JOB_ID	IN	WORD	6	作业 ID:		0
				• 不带有功能的作业	00 (十六进制)	
				• 写计数值	01 (十六进制)	
				• 写负载值	02 (十六进制)	
				• 写比较值	04 (十六进制)	
				• 写入滞后	08 (十六进制)	
				• 写入脉冲宽度	10 (十六进制)	
				• 读装载值	82 (十六进制)	
				• 读比较值	84 (十六进制)	
				• 读取滞后	88 (十六进制)	
• 读脉冲宽度	90 (十六进制)					
JOB_VAL	IN	DINT	8	写作业的值	-2 <sup>31</sup> 到 +2 <sup>31</sup> -1	0
STS_GATE	OUT	BOOL	12.0	内部门状态	TRUE/FALSE	FALSE
STS_STRT	OUT	BOOL	12.1	硬件门状态（启动输入）	TRUE/FALSE	FALSE
STS_LTCH	OUT	BOOL	12.2	锁存器输入状态	TRUE/FALSE	FALSE
STS_DO	OUT	BOOL	12.3	输出状态	TRUE/FALSE	FALSE
STS_C_DN	OUT	BOOL	12.4	反向计数状态 始终表示最后的计数方向。在第一次调用 SFB 后，STS_C_DN 被置位为 FALSE。	TRUE/FALSE	FALSE

参数	声明	数据类型	地址（背景数据块）	说明	值范围	缺省
STS_C_UP	OUT	BOOL	12.5	向上计数状态 始终表示最后的计数方向。在第一次调用 SFB 后，STS_C_UP 被置位为 TRUE。	TRUE/FALSE	FALSE
COUNTVAL	OUT	DINT	14	实际计数值	-2 <sup>31</sup> 到 2 <sup>31</sup> -1	0
LATCHVAL	OUT	DINT	18	实际锁存器值	-2 <sup>31</sup> 到 2 <sup>31</sup> -1	0
JOB_DONE	OUT	BOOL	22.0	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	OUT	BOOL	22.1	错误作业	TRUE/FALSE	FALSE
JOB_STAT	OUT	WORD	24	作业错误 ID	0 到 FFFF (十六进制)	0
STS_CMP	STAT	BOOL	26.3	比较器状态 *) 状态位 STS_CMP 表示是否满足比较器的比较条件。STS_CMP 还表示已置位输出 (STS_DO = TRUE)	TRUE/FALSE	FALSE
STS_OFLOW	STAT	BOOL	26.5	上溢状态 *)	TRUE/FALSE	FALSE
STS_UFLOW	STAT	BOOL	26.6	下溢状态 *)	TRUE/FALSE	FALSE
STS_ZP	STAT	BOOL	26.7	状态零标记 *) 仅当无主计数方向的情况下计数时，才置位此位。 指示零标记。如果计数器设置为 0，或计数器在装载值 = 0 时开始计数，也会置位此位。	TRUE/FALSE	FALSE
JOB_OVAL	STAT	DINT	28	读取请求的输出值	-2 <sup>31</sup> 到 2 <sup>31</sup> -1	0
RES_STS	STAT	BOOL	32.2	复位状态位 复位状态位 STS_CMP、STS_OFLOW、STS_UFLOW 和 STS_ZP。必须调用两次 SFB 才能复位状态位。	TRUE/FALSE	FALSE

\*) 使用 RES\_STS 复位

## SFB48 “FREQUENC” 的参数

参数	声明	数据类型	地址（背景数据块）	说明	值范围	缺省
LADDR	IN	WORD	0	在“HW Config”中指定的子模块 I/O 地址。如果 I/O 地址不相同，则您必须指定两者中较低的地址。	依 CPU 而定	十六进制
CHANNEL	IN	INT	2	通道号：	0-1	0
				• CPU 312C	0-2	
				• CPU 313C、CPU 313C-2 DP/PtP • CPU 314C-2 DP/PtP	0-3	
SW_GATE	IN	BOOL	4.0	软件门 用于启动/停止频率测量	TRUE/FALSE	FALSE
MAN_DO	IN	BOOL	4.1	启用输出的手动控制	TRUE/FALSE	FALSE
SET_DO	IN	BOOL	4.2	输出控制	TRUE/FALSE	FALSE
JOB_REQ	IN	BOOL	4.3	启动作业（正跳沿）	TRUE/FALSE	FALSE
JOB_ID	IN	WORD	6	作业 ID：		0
				• 不带功能的作业	00（十六进制）	
				• 写入下限	01（十六进制）	
				• 写入上限	02（十六进制）	
				• 写入积分时间	04（十六进制）	
				• 读下限	81（十六进制）	
				• 读上限	82（十六进制）	
• 读积分时间	84（十六进制）					
JOB_VAL	IN	DINT	8	写入作业的值	$-2^{31}$ 到 $+2^{31}-1$	0
STS_GATE	OUT	BOOL	12.0	内部门状态	TRUE/FALSE	FALSE
STS_STRT	OUT	BOOL	12.1	硬件门状态（启动输入）	TRUE/FALSE	FALSE
STS_DO	OUT	BOOL	12.2	输出状态	TRUE/FALSE	FALSE
STS_C_DN	OUT	BOOL	12.3	反向计数状态 始终指示最后的计数方向。在第一次调用 SFB 之后，STS_C_DN 被设置为 FALSE。	TRUE/FALSE	FALSE
STS_C_UP	OUT	BOOL	12.4	向上计数状态 始终表示最后的计数方向。在第一次调用 SFB 之后，STS_C_UP 被设置为 TRUE。	TRUE/FALSE	FALSE
MEAS_VAL	OUT	DINT	14	实际频率值	到 $2^{31}-1$	0
COUNTVAL	OUT	DINT	18	实际计数值 每当打开内部门时都从 0 开始计数。	$-2^{31}$ 到 $2^{31}-1$	0
JOB_DONE	OUT	BOOL	22.0	可启动新作业	TRUE/FALSE	TRUE

## 5.10 规范

参数	声明	数据类型	地址（背景数据块）	说明	值范围	缺省
JOB_ERR	OUT	BOOL	22.1	错误作业	TRUE/FALSE	FALSE
JOB_STAT	OUT	WORD	24	作业错误 ID	0 到 FFFF (十六进制)	0
STS_CMP	STAT	BOOL	26.3	测量结束状态 *) 每当积分时间结束，都要更新测量值。 此时，使用状态位 STS_CMP 报告测量的结束	TRUE/FALSE	FALSE
STS_OFLW	STAT	BOOL	26.5	上溢状态 *)	TRUE/FALSE	FALSE
STS_UFLW	STAT	BOOL	26.6	下溢状态 *)	TRUE/FALSE	FALSE
JOB_OVAL	STAT	DINT	28	读取请求的输出值	-2 <sup>31</sup> 到 2 <sup>31</sup> -1	0
RES_STS	STAT	BOOL	32.2	复位状态位 复位状态位 STS_CMP、STS_OFLW 和 STS_UFLW。 必须调用两次 SFB 才能复位状态位。	TRUE/FALSE	FALSE
*) 使用 RES_STS 复位						

## SFB 49 “PULSE” 的参数

参数	声明	数据类型	地址（背景数据块）	说明	值范围	缺省
LADDR	IN	WORD	0	在“HW Config”中指定的子模块 I/O 地址。如果 I/O 地址不相同，则您必须指定两者中较低地址。	依 CPU 而定	十六进制
CHANNEL	IN	INT	2	通道号：	0-3	0
				• CPU 312C	0-2	
				• CPU 313C、CPU 313C-2 DP/PtP • CPU 314C-2 DP/PtP	0-3	
SW_EN	IN	BOOL	4.0	软件门 启动/停止数据输出	TRUE/FALSE	FALSE
MAN_DO	IN	BOOL	4.1	启用输出的手动控制	TRUE/FALSE	FALSE
SET_DO	IN	BOOL	4.2	输出控制	TRUE/FALSE	FALSE
OUTP_VAL	IN	INT	6	指定缺省输出值： • 千分率形式 • S7 模拟值 如果指定一个大于 1,000 或 27,648 的值，CPU 会将该值限制为 1,000 或 27,648	0 到 1000 0 到 27648	0
JOB_REQ	IN	BOOL	8.0	启动作业（正跳沿）	TRUE/FALSE	FALSE
JOB_ID	IN	WORD	10	作业 ID：		0
				• 不带功能的作业	00（十六进制）	
				• 写周期长度	01（十六进制）	
				• 写接通延迟	02（十六进制）	
				• 写最小脉冲宽度	04（十六进制）	
				• 读周期长度	81（十六进制）	
				• 读上升延迟	82（十六进制）	
• 读最小脉冲宽度	84（十六进制）					
JOB_VAL	IN	DINT	12	写入作业的值	$-2^{31}$ 到 $+2^{31}-1$	0
STS_EN	OUT	BOOL	16.0	启用状态	TRUE/FALSE	FALSE
STS_STRT	OUT	BOOL	16.1	硬件门状态（启动输入）	TRUE/FALSE	FALSE
STS_DO	OUT	BOOL	16.2	输出状态	TRUE/FALSE	FALSE
JOB_DONE	OUT	BOOL	16.3	可启动新作业	TRUE/FALSE	TRUE
JOB_ERR	OUT	BOOL	16.4	错误作业	TRUE/FALSE	FALSE
JOB_STAT	OUT	WORD	18	作业错误 ID	0 到 FFFF （十六进制）	0
JOB_OVAL	STAT	DINT	20	读取请求的输出值	$-2^{31}$ 到 $2^{31}-1$	0



## 点对点通讯

### 6.1 概述

#### 6.1.1 产品描述

##### 基本信息

可使用串行接口进行 PtP 通讯，在可编程逻辑控制器、计算机或简单设备之间交换数据。伙伴设备间的通讯是基于串行异步传输而运行的。

CPU 313/314C-2PtP 的集成串行接口通过 X27 (RS422/485) 接口提供了通讯访问功能。

可使用下列协议：

- CPU 313C-2PtP: ASCII、3964(R)
- CPU 314C-2PtP: ASCII、3964(R) 和 RK 512

可以通过参数分配窗口组态通讯模式。

最多可传输 1,024 字节。全双工的可能传输率为 19.2 kbps，半双工为 38.4 kbps。

## 6.1.2 通讯伙伴

### 通讯伙伴的实例

利用 CPU 的串行接口，可与各种 Siemens 模块和第三方产品之间建立 PtP 连接。下面提供了一些实例：

- SIMATIC S5，通过 3964(R)/RK 512，S5 侧带有相应接口模块
- ES2 系列的 Siemens BDE 端子，通过 3964(R) 驱动程序
- MOBY I (ASM 420/421、SIM)，MOBY L (ASM 520) 和记录站 ES 030K，通过 3964(R) 驱动程序
- SIMOVERT 和 SIMOREG (USS 协议)，通过 ASCII 驱动程序 (ET 200S SI RS 422/485)，在 STEP 7 程序中具有相应的适配协议
- PC，通过 3964(R) 协议（提供了可在 PC 上编程的开发工具：PRODAVE DOS 64R (6ES5 897-2UD11) 用于 MS-DOS，PRODAVE WIN 64R (6ES5 897-VD01) 用于 Windows 或 ASCII 驱动程序）
- 条形码阅读器，通过 3964(R) 或 ASCII 驱动程序
- 其它厂商的 PLC，通过 RK512、3964(R) 或 ASCII 驱动程序
- 具有简单协议结构的其它设备，通过具有相应适配协议的 ASCII 驱动程序
- 同样配有 3964(R)/RK 512 的其它设备

## 6.1.3 PtP 通讯的组件

### 使用组件

串行连接的协议集成在 CPU 中。通过串行接口连接通讯伙伴。

使用屏蔽电缆作为**连接电缆**。在『电缆（页码 336）』一节中对用于各种通讯伙伴的连接电缆进行介绍。

对于**通讯伙伴**，您可连接配有支持相应协议的 RS422/485 接口的设备。

使用 PG/PC 以：

- 使用参数分配窗口为 CPU 的技术功能分配 CPU 参数。
- 为 CPU SFB 编程，您可直接将其集成在用户程序中。
- 借助标准 STEP 7 接口（监视功能和变量表）调试和测试 CPU。



## 6.1.4 (RS422/485) 接口的属性

### 定义

X27 (RS422/485) 接口代表符合 X27 标准并用于串行数据传输的差分电压接口。

- 在 RS422 模式中，通过四线制串行电缆发送数据（4 线操作）。两根导线（差分信号）用于发送方向，另两根用于接收方向。这意味着可以同时发送和接收数据（全双工操作）。
- 在 RS485 模式中，通过两线制串行电缆发送数据（2 线操作）。两根导线（差分信号）交替用于发送和接收方向。这意味着在给定的时间只能发送数据或接收数据（半双工操作）。完成发送操作后，电缆被立即切换至接收模式（发送器切换至高阻抗）。

通过参数分配窗口来选择操作模式。

### 属性

X27 (RS422/485) 接口具有下列属性并满足下列要求：

• 类型	差分电压接口
• 前连接器	带螺丝互锁的 15 针 D 型内孔连接器
• 最大传输率	38.4 kbps（半双工）
• 标准	DIN 66259 第 1 和第 3 部分, EIA-RS 422/485, CCITT V.11

## 6.1.5 串行传输字符

### 基本信息

在两个或多个通讯伙伴间传输数据时，有多种可选连网方案。在两个通讯伙伴之间建立 PtP 连接是进行信息交换的最简便方式。在 PtP 通讯中，数据串行发送。

### 串行数据传输

在串行传输中，以固定顺序逐位发送每个信息字节的各个位。通讯伙伴间的数据交换将通过串行接口自动处理。要实现此操作，CPU 要安装三个不同的驱动程序。

- ASCII 驱动程序
- 3964(R) 程序
- RK 512

### 半双工/全双工

我们将数据传输分为：

- 半双工（ASCII 驱动程序、3964(R) 协议、RK 512）  
数据交换在通讯伙伴间双向交替进行。半双工表示一次只能执行一个发送操作或一个接收操作。其中，单个数据流控制字符是例外（如 XON/XOFF）。在发送/接收操作期间同样可以发送/接收这些字符。
- 全双工（ASCII 驱动程序）  
在通讯伙伴之间同时进行数据交换。因此，可同时发送和接收数据。每个通讯伙伴必须能够同时操作发送器装置和接收器装置。

RS 485 模式（2 线）仅允许通过无流量控制的 ASCII 驱动程序执行半双工操作。

### 异步数据传输

串行数据是异步发送的。仅在传输字符期间才支持所谓的时基同步（在传输固定字符串时使用固定时间码）。要发送的每个字符前附加一个同步脉冲或起始位。字符传输结束时由停止位发出信号。

### 声明

除起始位和停止位外，在两个通讯伙伴间串行传输数据还需要其它附加声明。它们包括：

- 传输率（波特率）
- 字符延迟时间和应答延迟时间（如需要）
- 奇偶校验
- 数据位的数目
- 停止位的数目

## 字符帧

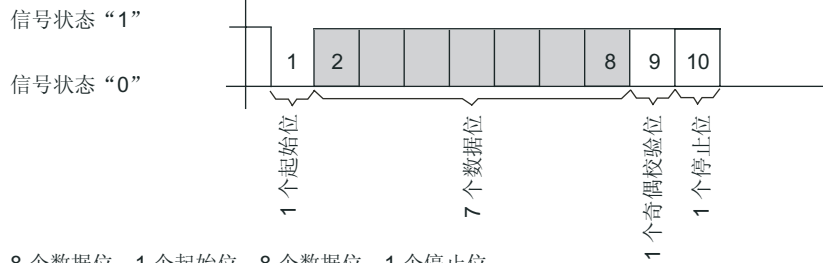
数据以字符帧的形式通过串行接口发送。每个字符帧可使用两种数据格式。不支持具有 7 个数据位，无奇偶校验位的操作。可借助参数组态工具组态所需的数据传输格式。

### 说明

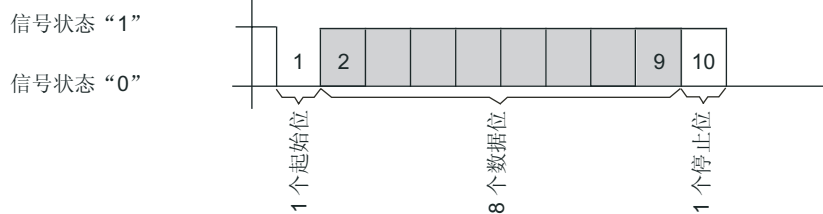
不支持具有 7 个数据位，无奇偶校验位的字符帧。

下面的示范视图说明了 10 位字符帧的两种数据格式：

7 个数据位：1 个起始位、7 个数据位、1 个奇偶校验位、1 个停止位

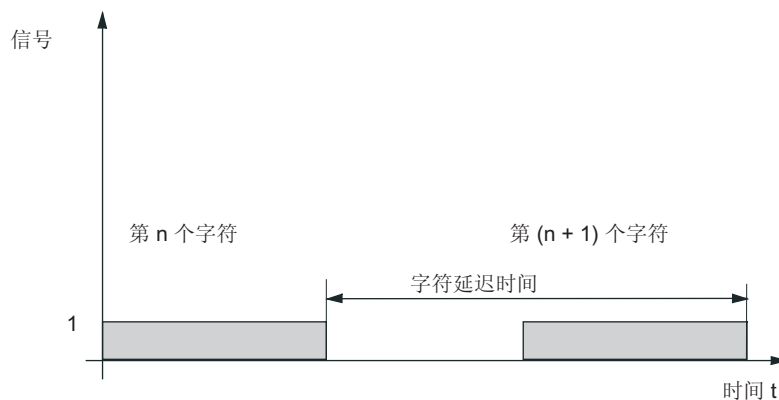


8 个数据位：1 个起始位、8 个数据位、1 个停止位



## 字符延迟时间

下图说明了一条消息的两个连续接收字符之间的最大允许时间间隔 = 字符延迟时间：



## 6.2 接线

### 6.2.1 接线规则

#### 连接电缆

- 必须将电缆屏蔽。
- 必须在两端端接电缆屏蔽。

#### 屏蔽连接元件

可使用屏蔽端接元件，将所有屏蔽电缆直接连接到安装导轨上，以此来进行接地。

#### 其它信息

有关其它信息，请参考“CPU 数据”手册和 CPU 安装手册。


## 6.2.2 连接串行电缆

### 针脚分配

下表显示了 CPU 前面板上 15 针 D 型子插座的针脚分配。

插座 RS 422/485 (前视图)	针脚	指示	输入/输出	说明	
	1	-	-	-	
	2	T (A) -	输出	发送数据 (四线制操作)	
	3	-	-	-	
	4	R (A) - R (A)/T (A) -	输入 输入/输出	接收数据 (四线制操作) 接收/发送数据 (两线制操作)	
	5	-	-	-	
	6	-	-	-	
	7	-	-	-	
	8	GND	-	-	功能性接地 (浮动)
	9	T (B) +	输出	发送数据 (四线制操作)	
	10	-	-	-	
	11	R (B) + R (B)/T (B) +	输入 输入/输出	接收数据 (四线制操作) 接收/发送数据 (两线制操作)	
	12	-	-	-	
	13	-	-	-	
	14	-	-	-	
	15	-	-	-	

在自己制作电缆时，请注意一定要使用屏蔽的连接器外壳。电缆屏蔽的两端必须连接到连接器外壳，且屏蔽电缆要通过较大的表面区域。

 <b>小心</b>
<p>绝不能将电缆屏蔽接地，这会毁坏表面。</p> <p>GND (针 8) 必须始终连接到两端以避免毁坏表面。</p>

在『电缆 (页码 336)』一节中对于各种通讯伙伴的连接电缆进行介绍。

## 6.3 参数组态

### 6.3.1 参数类型

#### 基本信息

在参数分配窗口中，可以调整串行通讯以适应具体应用。可以为参数分配两种参数类型：

- **模块参数**

有一些基本设置只指定一次，在过程运行时不再更改。本节将介绍这些参数。

- 通过参数分配窗口分配参数。
- 它们存储在 CPU 系统存储器中。

---

#### 说明

不能在 CPU 处于 RUN 模式时更改参数。

---

- **SFB 参数**

运行期间需要更改的参数位于系统功能块 (SFB) 的背景数据块中。在『ASCII/3964(R) 的通讯功能 - 基本功能 (页码 269)』一节中对 SFB 参数进行介绍。

- 可以在 DB 编辑器中离线编辑或在用户程序中在线编辑这些参数。
- 它们存储在 CPU 的工作存储器中。
- 可以在 CPU 处于 RUN 模式时，在用户程序中更改这些参数。

## 6.3.2 使用参数分配窗口进行组态

### 引言

借助于参数分配窗口可以自定义协议参数：

这些参数分配窗口基本都是自说明的。可以在接下来的章节和集成到参数分配窗口的帮助中找到参数说明。

### 要求

调用参数分配窗口的先决条件是已经创建了可以保存参数的项目。

### 步骤

1. 启动 SIMATIC 管理器，在项目中调用 HW Config。
2. 双击 CPU 的“PtP”子模块。“属性”对话框打开。
3. 编辑“PtP”子模块的参数，使用“确定”关闭参数分配窗口。
4. 使用“站 > 保存并编译”将组态保存在 HW Config 中。
5. 当 CPU 处于 STOP 模式时，使用“PLC > 下载到模块...”将参数数据下载到 CPU。现在数据即存储在 CPU 的系统数据存储器中。
6. 启动 CPU。

### 在线帮助

当组态参数时，可以在参数分配窗口集成的帮助中找到支持信息。可选择以下方式调用集成的帮助：

- 在相应视图中，按下 **F1** 键。
- 在不同的参数分配窗口单击**帮助按钮**。

### 6.3.3 基本参数

#### 说明

参数	说明	值范围	缺省
中断选择	在此可以指定是否触发诊断中断。	<ul style="list-style-type: none"> <li>• 无</li> <li>• 诊断</li> </ul>	无
对 CPU Stop 模式的反应	此参数影响接收缓冲区中对收到的消息帧的存储。 在两种情况下都取消传输过程。 在所有情况下，都将保持到此时为止存储的消息帧。 下列各表列有更详细的信息。	<ul style="list-style-type: none"> <li>• 继续</li> <li>• STOP</li> </ul>	继续

对“CPU Stop”模式的反应取决于执行操作时是否带数据流控制。

数据流控制	对 CPU Stop 模式的响应	刚刚到达的消息帧	新消息帧
无	继续	保存 如果缓冲区满，就放弃	保存到缓冲区满，然后放弃
	STOP	放弃	放弃
XON/XOFF	继续	保存 当缓冲区满时，激活流控制	保存 当缓冲区已满时，激活流控制
	STOP	因激活了流控制而不能再接收数据	因激活了流控制而不能再接收数据



### 6.3.4 ASCII 驱动程序的参数分配数据

#### 基本信息

在参数分配窗口中声明 ASCII 驱动程序的参数。下面是对这些参数的详细描述。

#### 说明

ASCII 驱动程序可用于 4 线操作 (RS 422) 和 2 线操作 (RS 485)。

#### 传输

参数	说明	值范围	缺省
波特率	以 bps 为单位的数据传输率 (波特) * 38,400 bps, 仅用于半双工	<ul style="list-style-type: none"> <li>• 300</li> <li>• 600</li> <li>• 1200</li> <li>• 2400</li> <li>• 4800</li> <li>• 9600</li> <li>• 19200</li> <li>• 38400*</li> </ul>	9600
起始位	传输期间, 每个要发送的字符前都附加一个起始位。	1 (固定值)	1
数据位	字符所映射到的位的数目。	<ul style="list-style-type: none"> <li>• 7</li> <li>• 8</li> </ul>	8
停止位	传输期间, 要发送的每个字符的尾部都附加一个停止位, 以表示字符结束。	<ul style="list-style-type: none"> <li>• 1</li> <li>• 2</li> </ul>	1
奇偶校验	可扩展信息位序列, 使其包括另一位, 即奇偶校验位。加上该位的值 (0 或 1) 之后, 所有位的值达到了定义的状态。从而增强了数据的完整性。 如果将奇偶校验指定为“无”, 将不发送奇偶校验位。 如果仅置位了 7 个数据位, 则无法指定“无”。	<ul style="list-style-type: none"> <li>• 无</li> <li>• 奇数</li> <li>• 偶数</li> </ul>	偶数

参数	说明	值范围	缺省
数据流控制	定义数据流控制要使用的方法。 仅在“全双工 (RS 422)4 线 PtP 操作”中使用流控制。 通过 XON/XOFF 打开软件数据流控制, 可避免因在不同速度下操作设备而导致发送的数据丢失。	<ul style="list-style-type: none"> <li>• 无</li> <li>• XON/XOFF</li> </ul>	无
XON 字符	XON 字符的代码 将 CPU 设置为使用流控制的操作模式后, CPU 将发送 XON 字符。 在取出消息帧后, 接收缓冲区准备再次接收字符时, CPU 将发送 XON 字符。	<ul style="list-style-type: none"> <li>• 7 个数据位: 0 到 7FH (十六进制)</li> <li>• 8 个数据位: 0 到 FFH (十六进制)</li> </ul>	11H = DC1
XOFF 字符	XOFF 字符的代码 当接收缓冲区 (接收缓冲区的大小: 2,048 字节) 在上溢之前接收到了声明的消息帧数, 或者 50 个字符时, CPU 将发送 XOFF 字符。如果通讯伙伴仍然继续发送数据, 则当接收缓冲区上溢时将生成一条错误消息。在最后一个消息帧中接收到的数据将被丢弃。	<ul style="list-style-type: none"> <li>• 具有 7 个数据位: 0 到 7FH (十六进制)</li> <li>• 具有 8 个数据位: 0 到 FFH (十六进制)</li> </ul>	13H = DC3
在 XOFF 后等待 XON	发送时, CPU 必须等待 XON 字符的时间。 当 CPU 收到 XOFF 字符时, 将中断数据传输。如果在指定组态时间内未收到 XON, 则取消发送操作, 并在 SFB 的 STATUS 输出中生成相应的错误消息 (0708H)。	20 到 65,530 ms 增量为 10 ms	20000 ms

文本结束字符

参数	说明	值范围	缺省																		
接收帧的消息结束识别	定义哪个标准表示消息帧结束。 <ul style="list-style-type: none"> <li>在字符延迟时间结束时： 消息帧没有固定长度也没有定义好的文本结束字符；消息结束由线路上的暂停（字符延迟时间结束）来定义。</li> <li>接收固定数目的字符： 接收的消息帧的长度始终相同。</li> <li>接收文本结束字符： 消息帧的结束由一个或两个定义好的文本结束字符来标记。</li> </ul>	<ul style="list-style-type: none"> <li>在字符延迟时间结束时</li> <li>在收到固定数目的字符时</li> <li>在收到文本结束字符时</li> </ul>	在字符延迟时间结束时																		
字符延迟时间 (CDT)	字符延迟时间定义两个连续接收到的字符间的最大允许间隔时间。	1 到 65,535ms 最短字符延迟时间取决于波特率	4 ms																		
缺少结束代码的监控时间	字符延迟时间被用作缺少结束代码的监控时间。它适用于结束代码的以下设置 <ul style="list-style-type: none"> <li>在收到固定长度的字符时</li> <li>在收到文本结束字符时</li> </ul>	<table border="1"> <thead> <tr> <th>波特</th> <th>字符延迟时间 [ms]</th> </tr> </thead> <tbody> <tr><td>300</td><td>130</td></tr> <tr><td>600</td><td>65</td></tr> <tr><td>1200</td><td>32</td></tr> <tr><td>2400</td><td>16</td></tr> <tr><td>4800</td><td>8</td></tr> <tr><td>9600</td><td>4</td></tr> <tr><td>19200</td><td>2</td></tr> <tr><td>38400</td><td>1</td></tr> </tbody> </table>	波特	字符延迟时间 [ms]	300	130	600	65	1200	32	2400	16	4800	8	9600	4	19200	2	38400	1	
波特	字符延迟时间 [ms]																				
300	130																				
600	65																				
1200	32																				
2400	16																				
4800	8																				
9600	4																				
19200	2																				
38400	1																				
消息帧之间的发送暂停时间等于监控时间	对于结束标准“收到固定长度的字符时”，将在两个消息帧之间保持与监控时间（缺少结束代码）等长的发送暂停，使伙伴能够与自身同步（识别收到的消息帧）。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	是																		
接收时的消息帧长度	对于结束标准“收到固定长度的字符时”，指定消息帧的字节长度。	1 到 1,024（字节）	1024																		
文本结束字符	可使用一个或两个文本结束字符。可选择在文本结束字符后接收一个或两个附加字符。您可以利用这些字符，例如，在传输中加入一个块检查字符 (BCC)。发送器中的计算以及接收器中的判断块检查字符必须由用户程序本身完成。	<ul style="list-style-type: none"> <li>1 个文本结束字符</li> <li>1 个文本结束字符，带 1 个 BCC</li> <li>1 个文本结束字符，带 2 个 BCC</li> <li>第 1 个和第 2 个文本结束字符</li> <li>第 1 个和第 2 个文本结束字符，带 2 个 BCC</li> <li>第 1 个和第 2 个文本结束字符，带 2 个 BCC</li> </ul>	1 个文本结束字符																		

## 6.3 参数组态

参数	说明	值范围	缺省
文本结束字符 1	第一个结束代码	<ul style="list-style-type: none"> <li>具有 7 个数据位： 0 到 7FH（十六进制）</li> <li>具有 8 个数据位： 0 到 FFH（十六进制）</li> </ul>	03H = ETX
文本结束字符 2	第 2 个结束代码（如果选择）	<ul style="list-style-type: none"> <li>具有 7 个数据位： 0 到 7FH（十六进制）</li> <li>8 个数据位： 0 到 FFH（十六进制）</li> </ul>	0
含有文本结束字符的传输	<p>对于结束标准“在收到文本结束字符时”，可在传输中加入一个文本结束字符。</p> <ul style="list-style-type: none"> <li>传输中包括文本结束字符： 结束代码必须包括在要发送的数据中。只能传输结束代码之前的所有数据，即使在 SFB 中指定了更大的数据长度。</li> <li>传输的数据长度最多等于在块参数中指定的长度： 传输的数据长度最多等于在 SFB 参数中声明的长度。最后一个字符必须是文本结束字符。</li> <li>传输的数据长度最多等于在块中指定的长度，并自动追加文本结束字符： 传输的数据长度最多等于在 SFB 参数中声明的长度。自动追加文本结束字符；即不得将结束代码含在要传输的数据中。根据结束代码的数目，向伙伴传输的字符数要比 SFB 中指定的字符数（最多 1024 字节）多 1 个或 2 个字符。</li> </ul>	<ul style="list-style-type: none"> <li>传输中包括文本结束字符</li> <li>传输的数据长度最多等于在块参数中指定的长度</li> <li>传输的数据长度最多等于在块参数中指定的长度，并自动追加文本结束字符</li> </ul>	传输中包括文本结束字符

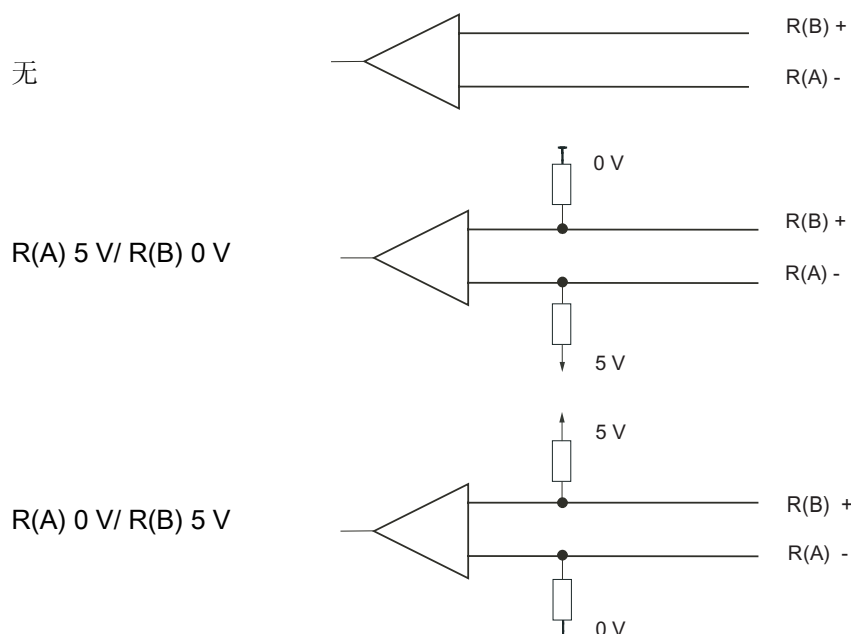
## 接收数据

参数	说明	值范围	缺省
启动时清除接收缓冲区	接收缓冲区在接通电源或 CPU 从 STOP 到 RUN 转换时清除。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	否
防止覆盖	可以使用该参数防止覆盖已满的接收缓冲区中的数据。	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	是
利用整个缓冲区	<p>可以使用整个接收缓冲区或指定要缓冲的接收到的消息帧数。</p> <p>如果使用 2048 字节长的整个缓冲区，缓冲的接收消息帧数只取决于帧的长度。</p>	<ul style="list-style-type: none"> <li>是</li> <li>否</li> </ul>	是
缓冲的已接收消息的最大帧数	<p>在“不使用整个缓冲区”组态中，可以指定要在接收缓冲区中缓冲的已接收消息帧数。</p> <p>如果在参数中声明“1”，取消激活“防止覆盖”参数，而且在用户程序中定期读出接收的数据，则当前消息帧总是传给目标块。</p>	1 到 10	10

### X27 (RS422/485) 接口的信号分配

参数	说明	值范围	缺省
操作模式	<p>指定是在全双工 (RS 422) 还是在半双工 (RS 485) 模式下运行 X27 (RS 422/485)。</p> <ul style="list-style-type: none"> <li>全双工 (RS 422) 4 线 PtP 操作模式 4 线 PtP 通讯的操作模式</li> <li>全双工 (RS 422) 4 线操作, 多点主站 CPU 为主站时, 支持多点 4 线运行的连接的操作模式。</li> <li>半双工 (RS 485) 2 线操作 PtP 通讯或支持多点 2 线操作的连接。CPU 可为主站或从站。</li> </ul>	<ul style="list-style-type: none"> <li>全双工 (RS 422) 4 线 PtP 通讯</li> <li>全双工 (RS 422) 4 线操作, 多点主站</li> <li>半双工 (RS 485) 2 线操作</li> </ul>	全双工 (RS 422) 4 线操作 PtP
缺省为接收线路	<ul style="list-style-type: none"> <li>无: 此设置只对具备总线能力的特定驱动程序有意义。</li> <li>信号 R(A) 5 伏/信号 R(B) 0 伏: 在此缺省状态, 可进行断点检测。 (对于全双工 (RS422) 4 线多点主站模式和半双工 (RS485) 2 线操作不可组态)</li> <li>信号 R(A) 0 伏/信号 R(B) 5 伏: 此缺省状态对应于空闲状态 (无激活的发送器)。在此缺省状态, 不可进行断点检测。</li> </ul>	<ul style="list-style-type: none"> <li>无</li> <li>信号 R(A) 5 伏/信号 R(B) 0 伏 (断点识别)</li> <li>信号 R(A) 0 伏/信号 R(B) 5 伏</li> </ul>	取决于设置的操作模式

以下视图显示 X27 (RS 422/485) 接口上接收器的接线图:



### 使用 CPU 的拓扑结构

在 RS422 或 RS485 操作模式中，CPU 可用于多种拓扑结构。

在与以下节点连接时存在区别：

- 两个节点 (PtP) 和
- 多个节点 (多点)。

其中，可将其用作

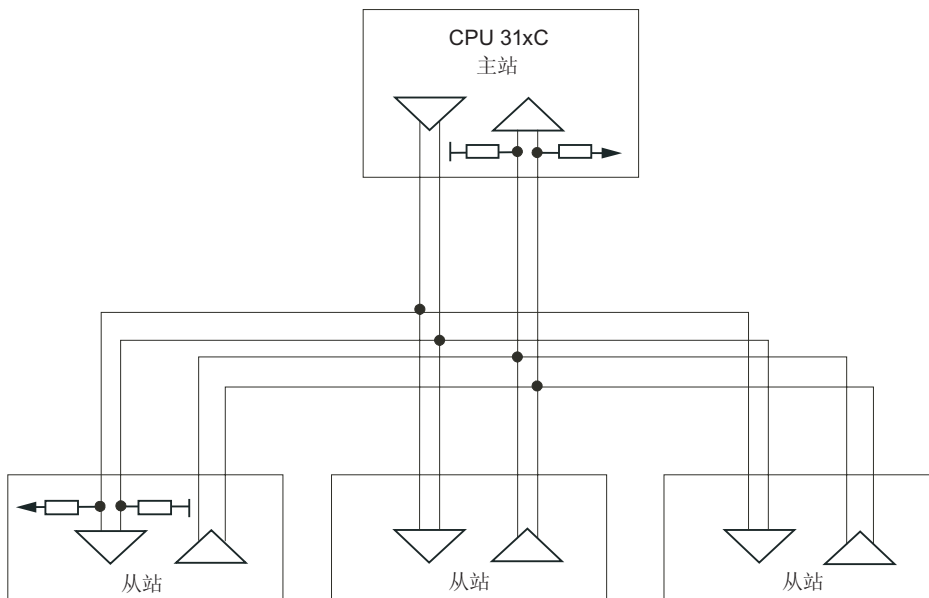
- 主站或
- 从站 (仅对于 RS485 操作)。

对于主站/从站拓扑结构，用户程序中必须有合适的消息帧。实例：主站向所有从站发送一个含有地址信息的信息帧。所有从站进行监听并与其自身的地址进行比较。如果地址相同，收到地址的从站将发送应答。

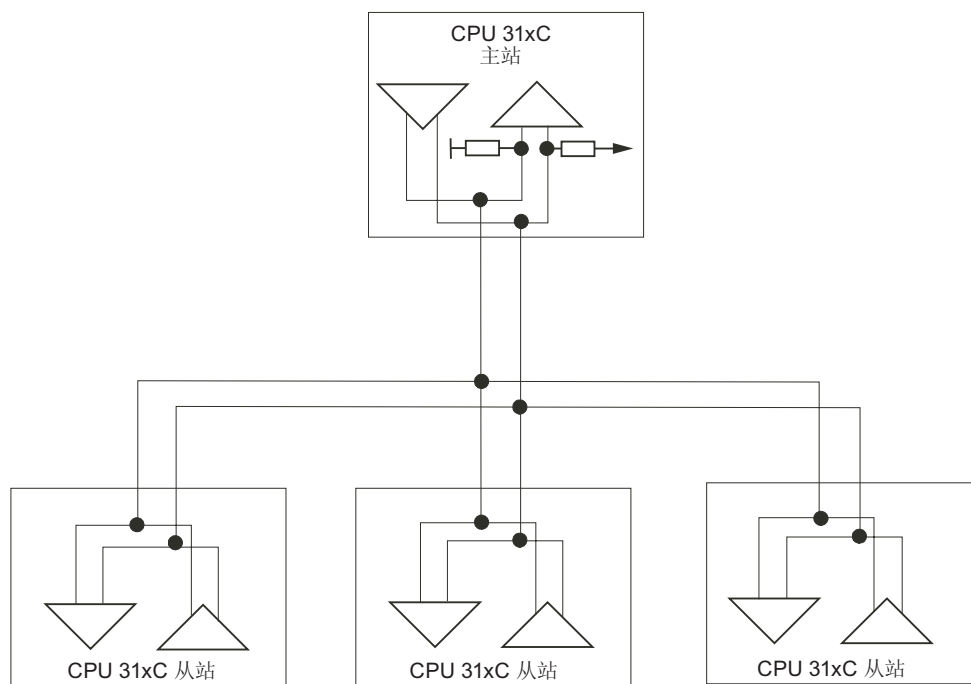
所有从站的发送器必须都能够切换至高阻抗。

- **RS422 操作中的主站/从站拓扑结构**

- 只能在主站模式下使用 CPU，
- 主站的发送器与所有从站的接收器互连。
- 从站的发送器与主站的接收器互连。
- 仅为主站和从站的接收器指定缺省设置。所有其它从站在无缺省设置的情况下运行。



- 在 RS485 操作的拓扑结构中，
  - 所有节点的发送/接收电缆对互连，
  - 仅一个节点的接收器具有缺省设置。所有其它模块在无缺省设置的情况下运行。



不同拓扑结构所需的组态在参数分配窗口的“接口”对话框中执行。

#### 说明

在 RS422 多点模式或 RS485 操作中编辑 ASCII 驱动程序时，必须始终确保用户程序中仅有一个节点正在发送数据。如果同时发送数据，消息帧将被破坏。

## 6.3.5 3964(R) 程序的参数分配数据

## 基本信息

在参数分配窗口，为 3964(R) 协议指定参数。下面是对这些参数的详细描述。

## 说明

3964(R) 协议只能用于四线操作模式 (RS 422)。

## 传输

参数	说明	值范围	缺省
波特率	以 bps (波特) 为单位的数据传输率。	<ul style="list-style-type: none"> <li>• 300</li> <li>• 600</li> <li>• 1200</li> <li>• 2400</li> <li>• 4800</li> <li>• 9600</li> <li>• 19200</li> <li>• 38400</li> </ul>	9600
起始位	传输期间，每个要发送的字符前都附加一个起始位。	1 (固定值)	1
数据位	一个字符对应的位数。	<ul style="list-style-type: none"> <li>• 7</li> <li>• 8</li> </ul>	8
停止位	传输期间，要发送的每个字符的尾部都附加一个停止位；该位表示字符结束。	<ul style="list-style-type: none"> <li>• 1</li> <li>• 2</li> </ul>	1
奇偶校验	可扩展信息位序列，使其再包括一位，即奇偶校验位。加上该位的值 (0 或 1) 之后，所有位的值都达到了定义的状态。这样可以增加数据可靠性。 如果将奇偶校验指定为“无”，将不发送奇偶校验位。 <b>如果置位了 7 个数据位，则无法指定“无”。</b>	<ul style="list-style-type: none"> <li>• 无</li> <li>• 奇数</li> <li>• 偶数</li> </ul>	偶数
优先级	如果某个伙伴发送的请求优先于其它伙伴发送的请求，则该伙伴具有高优先级。如果某个伙伴发送的请求必须等到其它伙伴的请求处理完毕后才能处理，则该伙伴具有低优先级。对于 3964(R) 协议，必须为通讯伙伴双方分配不同的优先级，即为一个伙伴分配高优先级，为另一个伙伴分配低优先级。	<ul style="list-style-type: none"> <li>• 低</li> <li>• 高</li> </ul>	高



参数	说明	缺省
带标准值和块检查的消息帧参数 3964(R)	协议参数设置为缺省值。 CPU 检测到字符串 DLE ETX BCC 时，停止接收。它将接收到的块检查字符 BCC 与内部计算的纵向奇偶校验加以比较。如果块检查字符正确而且没有发生其它接收错误，它将发送 DLE 字符（出错时，它将发送 NAK 字符到通讯伙伴）。	带标准值和块检查的 3964(R): 字符延迟 = 220 ms 确认延迟 = 2000 ms 重新尝试连接的次数 = 6 重新尝试传输的次数 = 6
带块检查的可分配消息帧参数 3964(R)	协议参数可自由编程。 CPU 检测到字符串 DLE ETX BCC 时，将停止接收。它将接收到的块检查字符 BCC 与内部计算的纵向奇偶校验加以比较。如果块检查字符正确而且没有发生其它接收错误，它将发送 DLE 字符（出错时，它将发送 NAK 字符到通讯伙伴）。	
带标准值但不带块检查的消息帧参数 3964	协议参数设置为缺省值。 当 CPU 检测到 DLE ETX 字符串时，如果块接收无误，它停止接收并发送 DLE 到通讯伙伴（或者出错时发送 NAK）。	
不带块检查的可分配消息帧参数 3964	协议参数可自由编程。 当 CPU 检测到 DLE ETX 字符串时，如果块接收无误，它将停止接收并发送 DLE 到通讯伙伴（如果接收出错，则发送 NAK）。	

参数	说明	值范围	缺省
字符延迟	字符延迟定义了消息帧的两个接收字符之间允许的最大时间间隔。	20 ms 到 65530 ms，以 10 ms 为增量 最短字符延迟取决于波特率： 300 bps: 60 ms 600 bps: 40 ms 1200 bps: 30 ms 2400 到 38400 bps: 20 ms	220 ms
确认延迟	确认延迟决定建立连接时伙伴发送确认信号所允许的最大延迟时间（STX 和伙伴的确认 DLE 之间的时间），或者是结束连接时伙伴发送确认信号所允许的最大延迟时间（DLE ETX [BCC] 和伙伴的 DLE 确认之间的时间）。	20 ms 到 65530 ms，以 10 ms 为增量 最短确认延迟取决于波特率： 300 bps: 60 ms 600 bps: 40 ms 1200 bps: 30 ms 2400 到 38400 bps: 20 ms	2000 ms（对于不带块检查的 3964，为 550 ms）
重新尝试连接的次数	该参数定义 CPU 尝试建立连接的最大次数。	1 到 255	6
重新尝试传输的次数	该参数定义了 在出错情况下尝试传输消息帧（包括第一个消息帧）的最大次数。	1 到 255	6

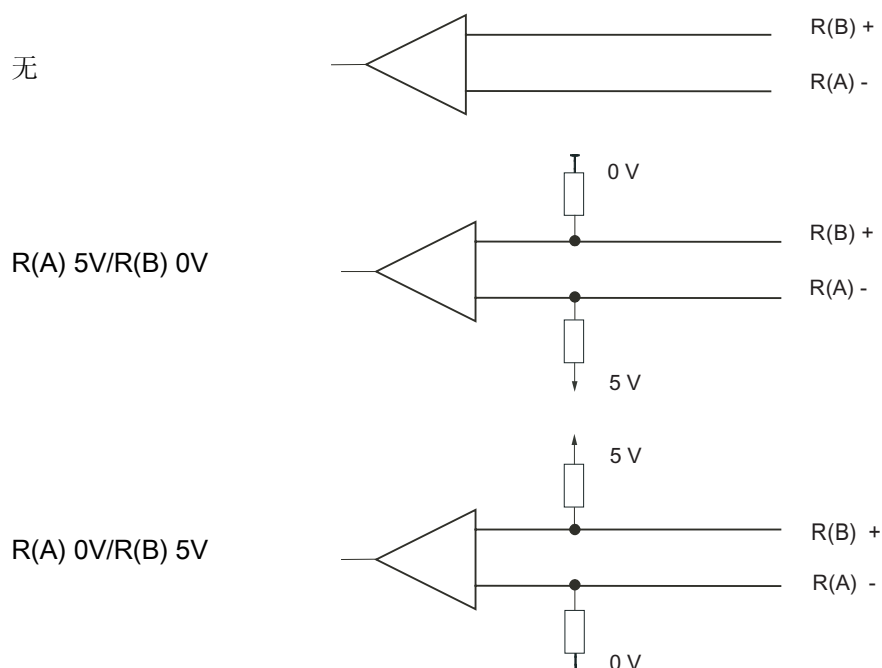
## 数据接收

参数	说明	值范围	缺省
启动时清空接收缓冲区	在接通电源或 CPU 从 STOP 跳转到 RUN 时清空接收缓冲区。	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	否
防止覆盖	可以使用该参数防止覆盖已满的接收缓冲区中的数据。	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	是
使用整个缓冲区	可以使用整个接收缓冲区或指定要缓冲的已接收帧数。 如果使用整个缓冲区（2,048 字节），缓冲的接收帧数只取决于帧的长度。	<ul style="list-style-type: none"> <li>• 是</li> <li>• 否</li> </ul>	是
缓冲的已接收帧的最大数目	通过设置“不使用整个缓冲区”，可以指定要在接收缓冲区中缓冲的已接收帧数。 如果分配“1”，则取消激活“防止覆盖”参数， <b>并且</b> 在用户程序中周期性读出接收的数据，始终将当前消息帧传送给目标数据块。	1 到 10	10

### X27 (RS 422/485) 接口的信号分配

参数	说明	值范围	缺省
接收线路初始状态	无：此设置仅对应用于具备总线能力的驱动程序。	无	R(A) 5V/R(B) 0V
	R(A) 5 伏/ R(B) 0 伏：在此缺省状态，可进行断点检测。	R(A) 5V/R(B) 0V	
	R(A) 0 伏/ R(B) 5 伏：在此缺省状态，不可进行断点检测。	R(A) 0V/R(B) 5V	

下图显示了接收器到 X27 (RS 422) 接口的接线情况：



### 6.3.6 用于 RK 512 通讯的参数分配数据

#### 基本信息

这些参数与 3964(R) 协议的参数相同，因为 3964(R) 代表 RK 512 的部分通讯量。

例外：

- RK 512 通讯的每个字符位数的设置固定为 8 位。
- 没有接收缓冲区（没有用于接收数据的参数）。

必须在使用的系统功能块 (SFB) 中为数据目标或源指定参数。

## 6.4 在用户程序中实现连接

### 步骤

使用用户程序控制串行连接。为此，可调用系统功能块 (SFB)。SFB 在“标准库”的“系统功能块”下。

以下各节可帮助您为您的应用程序设计用户程序。

### 调用 SFB

通过相应的背景数据块调用 SFB

实例：CALL SFB 60, DB20

### 背景数据块

SFB 需要的所有参数都存储在背景数据块中。

---

#### 说明

在用户程序中，您必须始终使用同一背景数据块调用每个 SFB 类型 (SEND、FETCH、RCV...)，这是因为内部 SFB 处理所需的状态存储在该背景数据块中。

不允许访问背景数据块中的数据。

---

### 程序结构

SFB 是以异步方式处理的。要完整地处理 SFB，必须根据需要频繁调用它，直至其有错或无错地结束。

---

#### 说明

因为 SFB 不能中断本身，所以不能在具有其它优先级等级的另一程序部分中调用已在程序中组态的 SFB。

实例：不允许在 OB 1 和中断 OB 中调用 SFB。

---

### SFB 参数分类

SFB 参数按功能可分为四类：

- **控制参数**用于激活块。
- **发送参数**指向要传输到远程伙伴的数据区。
- **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。
- **状态参数**用于监视块是否完成其任务且没有错误，或者分析出现的错误。状态参数只在一次调用期间置位。

## 6.5 通讯功能

### 6.5.1 ASCII/3964(R) 的通讯功能

#### 6.5.1.1 ASCII/3964(R) 的通讯功能 - 基本功能

##### 概述

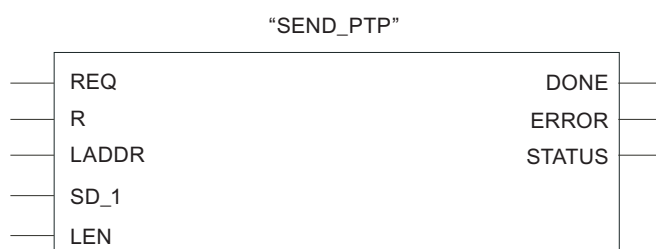
针对 ASCII 和 3964 协议提供了以下功能。

块		说明
SFB 60	SEND_PTP	将整个或部分数据块发送到通讯伙伴。
SFB 61	RCV_PTP	从通讯伙伴接收数据并保存到 DB 中。
SFB 62	RES_RCVB	复位 CPU 的接收缓冲区。

#### 6.5.1.2 用 SFB 60 “SEND\_PTP” 发送数据

##### 基本信息

使用该 SFB 从 DB 发送数据块。



块调用之后，在控制输入 **REQ** 的正跳沿上激活发送过程。要发送的数据区在 **SD\_1**（DB 号和起始地址）中指定，数据块长度在 **LEN** 中指定。

必须通过 **R**（复位）= FALSE 调用 SFB 以使其可处理请求。在控制输入 **R** 的正跳沿上，当前传输被取消，SFB 复位到初始状态。被取消的请求结束，并带有错误消息（STATUS 输出）。

在 **LADDR** 中，输入您在“HW Config”中指定的子模块 I/O 地址。

如果请求已关闭且没有错误，则 **NDR** 置位为 **TRUE**，或者如果请求已终止且具有错误，则 **ERROR** 置位为 **TRUE**。

如果请求完成且 **DONE = TRUE**，则说明：

- 使用 **ASCII** 驱动程序时：数据已传输到通讯伙伴。但是，并不确保数据被通讯伙伴接收到。
- 使用 **3964(R)** 协议时：数据传输到通讯伙伴，返回一个肯定确认。但是，并不确保数据被传递到通讯伙伴 **CPU**。

如果出现错误或警告，**STATUS** 将显示相应的事件编号（请参阅『错误消息（页码 340）』一节）。如果 **SFB** 复位 (**R = TRUE**)，还会输出 **DONE** 或 **ERROR/STATUS**。如果出现错误，则二进制结果 **BIE** 复位。如果块结束且无错，则二进制结果的状态是 **TRUE**。

---

#### 说明

**SFB** 没有参数检查。如果编程不正确，**CPU** 可能会转到 **STOP** 模式。

---

参数	声明	数据类型	说明	值范围	缺省
REQ	IN	BOOL	控制参数“请求”： 在正跳沿激活数据交换	TRUE/FALSE	FALSE
R	IN	BOOL	控制参数“复位”： 取消请求。发送被阻。	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址。	依 CPU 而定	3FF（十六进制）
DONE	OUT	BOOL	状态参数（仅在某个调用期间设置）： <ul style="list-style-type: none"> <li>FALSE：请求仍未启动或请求正被执行。</li> <li>TRUE：请求已完成且无错。</li> </ul>	TRUE/FALSE	FALSE
ERROR	OUT	BOOL	状态参数（仅在一次调用期间设置）： 请求已完成且有错。	TRUE/FALSE	FALSE
STATUS	OUT	WORD	状态参数（仅在某个调用期间设置。为了显示 STATUS，应将 STATUS 复制到空闲数据区）： 根据 ERROR 位，STATUS 有下列含义： <ul style="list-style-type: none"> <li>ERROR = FALSE： STATUS 的值是： <ul style="list-style-type: none"> <li>0000H：既不警告也不出错</li> <li>0000H：警告，STATUS 提供详细信息</li> </ul> </li> <li>ERROR = TRUE： 错误未决。STATUS 提供有关错误类型的详细信息（请参阅『错误消息（页码 340）』）。</li> </ul>	0 到 FFFF （十六进制）	0
SD_1	IN_OUT	ANY	发送参数： 在此指定： <ul style="list-style-type: none"> <li>要从其中发送数据的 DB 的编号。</li> <li>要发送数据的起始字节号。</li> </ul> 例如：DB10（从字节 2 开始）-> DB10.DBB2	依 CPU 而定	0
LEN	IN_OUT	INT	在此指定要传输的数据块的长度（以字节为单位）。（在此间接指定长度。）	1 到 1024	1

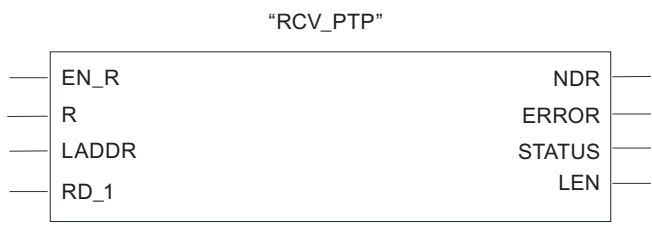
## 数据一致性

一致性数据限制到 206 字节。要传输超过 206 字节的一致性数据，请注意以下几点：  
传输过程完成之前，请勿向发送区 SD\_1 的当前使用部分写入数据。这就是状态参数 DONE = TRUE 时的情形。

### 6.5.1.3 用 SFB 61 “RCV\_PTP” 接收数据

#### 基本信息

使用此 SFB 接收数据并将其保存在数据块中。



在控制输入 **EN\_R** 的值为 **TRUE** 的情况下调用块之后，该块就处于准备接收数据的状态。可以通过参数 **EN\_R** 上的信号状态 **FALSE** 来取消当前传输。被取消的请求结束时带有错误消息（**STATUS** 输出）。只要参数 **EN\_R** 的信号状态为 **FALSE**，接收就被锁定。

接收区在 **RD\_1**（DB 号和起始地址）中指定，数据块长度在 **LEN** 中指定。

必须通过 **R**（复位）= **FALSE** 调用 SFB 以使其可处理请求。在控制输入 **R** 的正跳沿上，当前传输被取消，SFB 复位到初始状态。被取消的请求结束，并带有错误消息（**STATUS** 输出）。

在 **LADDR** 中，输入您在“HW Config”中指定的子模块 I/O 地址。

如果请求已关闭且没有错误，则 **NDR** 置位为 **TRUE**，或者如果请求已终止且具有错误，则 **ERROR** 置位为 **TRUE**。

如果出现错误或警告，**STATUS** 将显示相应的事件编号（请参阅『错误消息（页码 340）』一节）。如果 SFB 复位（**R** = **TRUE**）（参数 **LEN** = 16#00），也会输出 **NDR** 或 **ERROR/STATUS**。如果出现错误，则二进制结果 **BIE** 将复位。如果块结束且无错，则二进制结果的状态是 **TRUE**。

---

#### 说明

SFB 没有参数检查。如果编程不正确，CPU 可能会转到 STOP 模式。

---



参数	声明	数据类型	说明	值范围	缺省
EN_R	IN	BOOL	控制参数“启用接收”： 启用接收	TRUE/FALSE	FALSE
R	IN	BOOL	控制参数“复位”： 取消请求	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址。	CPU 特定	3FF（十六进制）
NDR	OUT	BOOL	状态参数“新数据就绪”： 请求完成且无错，接受了数据。 <ul style="list-style-type: none"> <li>FALSE：请求尚未启动或仍在运行。</li> <li>TRUE：请求已成功完成。</li> </ul>	TRUE/FALSE	FALSE
ERROR	OUT	BOOL	状态参数（仅在一次调用期间设置）： 请求完成，出现错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	状态参数（仅在一次调用期间设置。为了显示 STATUS，应将 STATUS 复制到空闲数据区）： 根据 ERROR 位，STATUS 有下列含义： <ul style="list-style-type: none"> <li>ERROR = FALSE： STATUS 的值是： <ul style="list-style-type: none"> <li>0000H：既不警告也不出错</li> <li>&lt;&gt; 0000H：警告，STATUS 提供详细信息</li> </ul> </li> <li>ERROR = TRUE： 错误未决。STATUS 提供有关错误类型的详细信息（有关错误编号，请参阅『错误消息（页码 340）』一节）。</li> </ul>	0 到 FFFF （十六进制）	0
RD_1	IN_OUT	ANY	接收参数： 在此指定： <ul style="list-style-type: none"> <li>存储接收数据的 DB 号。</li> <li>要用于存储接收数据的起始数据字节号。</li> </ul> 例如：DB20（从字节 5 开始）-> DB20.DBB5	依 CPU 而定	0
LEN	IN_OUT	INT	数据长度输出（字节数）	0 到 1024	0

## 数据一致性

一致性数据限制到 206 字节。要传输超过 206 字节的一致性数据，请注意以下几点：

除非已接受所有数据 (NDR = TRUE)，否则不要访问接收 DB。之后，锁定接收 DB 直到 (EN\_R = FALSE) 数据处理完毕。

### 6.5.1.4 用 SFB 62 “RES\_RCVB” 清除接收缓冲区

#### 基本信息

使用 SFB 可清空 CPU 的整个接收缓冲区。所有存储的消息帧都将被删除。调用 “RES\_RCVB” 时，进入的消息帧即被保存。



块调用之后，在控制输入 **REQ** 的正跳沿上激活请求。请求的运行可以跨多个调用（程序周期）。

必须通过 **R**（复位）= **FALSE** 调用 SFB 以使其可处理请求。在控制输入 **R** 的正跳沿上，清空过程被取消，SFB 复位到初始状态。被取消的请求结束，并带有错误消息（**STATUS** 输出）。

在 **LADDR** 中，输入您在 “HW Config” 中指定的子模块 I/O 地址。

如果请求已关闭且没有错误，则 **NDR** 置位为 **TRUE**，或者如果请求已终止且具有错误，则 **ERROR** 置位为 **TRUE**。

如果出现错误或警告，**STATUS** 将显示相应的事件编号（请参阅『错误消息（页码 340）』一节）。

如果 SFB 复位 (**R = TRUE**)，还会输出 **DONE** 或 **ERROR/STATUS**。

如果出现错误，则二进制结果 **BIE** 复位。如果块结束且无错误，则二进制结果的状态为 **TRUE**。

---

#### 说明

SFB 没有参数检查。如果编程不正确，CPU 可能会跳转到 **STOP** 模式。

---

参数	声明	数据类型	说明	值范围	缺省
REQ	IN	BOOL	控制参数“请求”： 在正跳沿激活请求	TRUE/FALSE	FALSE
R	IN	BOOL	控制参数“复位”： 取消请求	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址。	依 CPU 而定	3FF（十六进制）
DONE	OUT	BOOL	状态参数（仅在一次调用期间置位）： <ul style="list-style-type: none"> <li>FALSE：请求仍未启动或请求正被执行。</li> <li>TRUE：请求已完成且无错。</li> </ul>	TRUE/FALSE	FALSE
ERROR	OUT	BOOL	状态参数（仅在一次调用期间设置）： 请求已完成且有错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	状态参数（仅在一次调用期间设置。为了显示 STATUS，应将 STATUS 复制到空闲数据区）： 根据 ERROR 位，STATUS 具有以下意义： <ul style="list-style-type: none"> <li>ERROR = FALSE： STATUS 值为： <ul style="list-style-type: none"> <li>0000H：既没有警告也没有错误</li> <li>&lt;&gt; 0000H：警告，STATUS 提供详细信息</li> </ul> </li> <li>ERROR = TRUE： 错误未决。STATUS 提供有关错误类型的详细信息（有关错误编号，请参阅『错误消息（页码 340）』一节）。</li> </ul>	0 到 FFFF （十六进制）	0

## 6.5.2 RK 512 通讯功能

### 6.5.2.1 RK 512 计算机连接的通讯功能 - 基本功能

#### 概述

针对 RK 512 协议提供以下功能：

块		说明
SFB 63	SEND_RK	将整个或部分数据块发送到通讯伙伴。
SFB 64	FETCH_RK	将整个或部分数据块区域发送到通讯伙伴。
SFB 65	SERVE_RK	<ul style="list-style-type: none"> <li>从通讯伙伴接收数据并保存到 DB 中。</li> <li>为通讯伙伴提供数据。</li> </ul>

#### 请求的并行处理

在用户程序中，不能同时激活 SEND/FETCH 请求。举例来说就是，如果 SEND 请求尚未关闭，则不能启动 FETCH 请求。

#### SYNC\_DB

为了在设置时初始化，以及同步 SFB 之间的操作，用于 RK512 通讯的所有 SFB 需要一个公共数据区。通过参数 SYNC\_DB 确定 DB 号。DB 号对于用户程序中的所有 SFB 都必须相同。DB 最小长度必须为 240 字节。

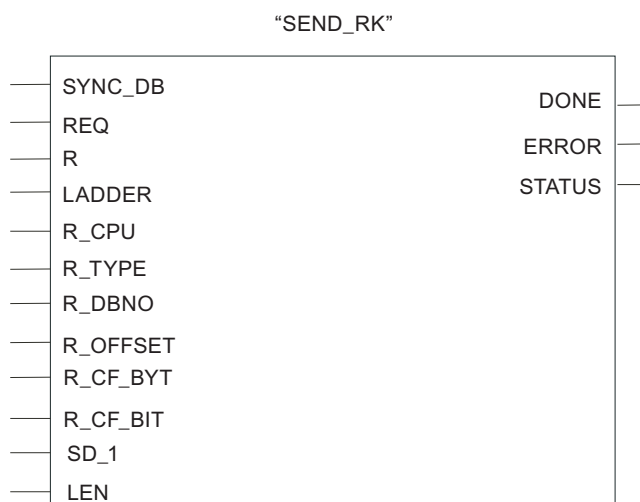
#### 处理器间通讯标志

为了协调 CPU 的数据处理和接收或提供数据时的异步覆盖，SIMATIC S5 通过 SFB “SERVE\_RK” (SFB 65) 支持处理器间通讯标志功能。

## 6.5.2.2 用 SFB 63 “SEND\_RK” 发送数据

### 基本信息

使用该 SFB 从 DB 传输数据块。



块调用之后，在控制输入 **REQ** 的正跳沿上激活发送过程。

要发送的数据区在 **SD\_1**（DB 号和起始地址）中指定，数据块长度在 **LEN** 中指定。

在 **SFB** 中，还可指定伙伴的接收区。此信息由 **CPU** 输入到消息帧头并传输给伙伴。

目标由 **CPU** 号 **R\_CPU**（仅与多处理器通讯有关）、数据类型 **R\_TYPE**（数据块 [DB] 和扩展数据块 [DX]）、数据块号 **R\_DBNO** 和偏移量 **R\_OFFSET** 指定。

在 **R\_CF\_BYT** 和 **R\_CF\_BIT** 中指定伙伴 **CPU** 上处理器间通讯标志字节和位。

在 **SYNC\_DB** 参数中，指定要在其中存储数据的 **DB**，这些数据对于所有用于启动初始化和同步例程的 **SFB** 来说是公用的。**DB** 号对于用户程序中的所有 **SFB** 都必须相同。

必须通过 **R**（复位）= **FALSE** 调用 **SFB** 以使其可处理请求。在控制输入 **R** 的正跳沿上，当前传输被取消，**SFB** 设置为初始状态。被取消的请求结束，并带有错误消息（**STATUS** 输出）。

在 **LADDR** 中，输入您在“HW Config”中指定的子模块 I/O 地址。

如果请求已关闭且没有错误，则 **NDR** 置位为 **TRUE**，或者如果请求已终止且具有错误，则 **ERROR** 置位为 **TRUE**。

如果请求已完成且 **DONE = TRUE**，则表示数据已传送到通讯伙伴且已得到肯定确认，并且数据已传送到伙伴 CPU。

如果出现错误或警告，**STATUS** 将显示相应的事件编号（请参阅『错误消息（页码 340）』一节）。

如果 **SFB** 复位 (**R = TRUE**)，也会输出 **DONE** 或 **ERROR/STATUS**。

如果出现错误，则二进制结果 **BIE** 复位。如果块结束且无错误，则二进制结果的状态为 **TRUE**。

#### 说明

**SFB** 没有参数检查。如果编程不正确，CPU 可能会跳转到 **STOP** 模式。

参数	声明	数据类型	说明	值范围	缺省
<b>SYNC_DB</b>	IN	INT	DB 号，其中存储了用于 RK SFB 同步的公用数据（最小长度为 240 字节）。	CPU 特定，不允许使用零。	0
<b>REQ</b>	IN	BOOL	控制参数“请求”： 在正跳沿激活数据交换	TRUE/FALSE	FALSE
<b>R</b>	IN	BOOL	控制参数“复位”： 取消请求。发送被阻。	TRUE/FALSE	FALSE
<b>LADDR</b>	IN	WORD	在“HW Config”中指定的子模块 I/O 地址。	依 CPU 而定	3FF（十六进制）
<b>R_CPU</b>	IN	INT	伙伴 CPU 号 （只用于多处理器模式）	0 到 4	1
<b>R_TYPE</b>	IN	CHAR	伙伴 CPU 的地址类型（只允许大写字母!）： <ul style="list-style-type: none"> <li>• ‘D’ = 数据块</li> <li>• ‘X’ = 扩展数据块</li> </ul>	‘D’，‘X’	‘D’
<b>R_DBNO</b>	IN	INT	伙伴 CPU 上的数据块号	0 到 255	0
<b>R_OFFSET</b>	IN	INT	伙伴 CPU 上的数据字节号	0 到 510 （只允许偶数值）	0
<b>R_CF_BYT</b>	IN	INT	伙伴 CPU 的处理器间通讯标志字节 （255 表示：没有处理器间通讯标记）	0 到 255	255
<b>R_CF_BIT</b>	IN	INT	伙伴 CPU 的处理器间通讯标志位	0 到 7	0

参数	声明	数据类型	说明	值范围	缺省
<b>DONE</b>	OUT	BOOL	状态参数（仅在一次调用期间设置）： <ul style="list-style-type: none"> <li>FALSE：请求尚未启动或仍在执行。</li> <li>TRUE：请求已完成且没有错误。</li> </ul>	TRUE/FALSE	FALSE
<b>ERROR</b>	OUT	BOOL	状态参数（仅在一次调用期间设置）： 请求已完成且有错误。	TRUE/FALSE	FALSE
<b>STATUS</b>	OUT	WORD	状态参数（仅在一次调用期间设置。为了显示 STATUS，应将 STATUS 复制到空闲数据区）： 根据 ERROR 位，STATUS 具有以下意义： <ul style="list-style-type: none"> <li>ERROR = FALSE： STATUS 值为： <ul style="list-style-type: none"> <li>0000H：既没有警告也没有错误</li> <li>&lt;&gt; 0000H：警告，STATUS 提供详细信息</li> </ul> </li> <li>ERROR = TRUE： 错误未决。STATUS 提供有关错误类型的详细信息（有关错误编号，请参阅『错误消息（页码 340）』一节）。</li> </ul>	0 到 FFFF (十六进制)	0
<b>SD_1</b>	IN_OUT	ANY	发送参数： 在此指定： <ul style="list-style-type: none"> <li>要从其中发送数据的 DB 号。</li> <li>要发送数据的起始数据字节号。</li> </ul> 例如：DB10（从字节 2 开始）-> DB10.DBB2	依 CPU 而定	0
<b>LEN</b>	IN_OUT	INT	在此指定要传输的数据块的长度（以字节为单位）。（在此间接指定长度。）	1 到 1024	1

## 数据一致性

数据一致性限制到 128 字节。要传输超过 128 字节的一致性数据，请注意以下几点：  
传输过程完成之前，请勿向发送区 SD\_1 的当前使用部分写入数据。这就是状态参数 DONE = TRUE 时的情形。

### 发送数据的特性

注意以下“发送数据”的特性：

- RK 512 仅允许传输偶数长度的数据。如果指定奇数数据长度 (LEN)，则会为传输的数据追加一个填充字节“0”。
- RK 512 仅允许指定偶数偏移量。如果指定奇数偏移量，则在伙伴站点上，数据从下一个较小的偶数偏移量开始存储。

实例：偏移量是 7，数据从字节 6 开始写入。

### 消息帧头中的信息

下表显示了 RK 512 消息帧头中的信息。

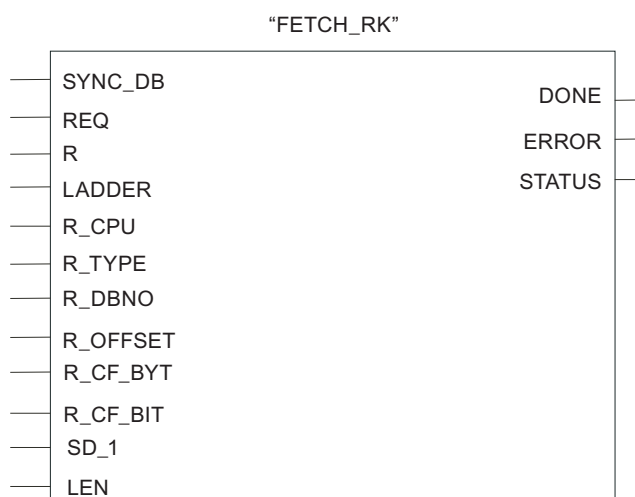
S7 自动化系统上的源 (本地 CPU)	至目标, 伙伴 CPU	消息帧头		
		字节 3/4: 命令类型	字节 5/6: Z-DBNR/Z 偏移	字节 7/8: 计量单位
数据块	数据块	AD	DB/DW	字
数据块	扩展数据块	AD	DB/DW	字
缩写: Z-DBNR: 目标数据块号 Z-Offset: 目标起始地址 DW: 以字为单位的偏移量				



### 6.5.2.3 用 SFB 64 “FETCH\_RK” 获取数据

#### 基本信息

使用此 SFB 从伙伴获取数据块并将数据保存到 DB。



块调用之后，在控制输入 **REQ** 的正跳沿上激活发送过程。

在 **RD\_1** 中指定用于存储所获取的数据的存储器区（DB 号和起始地址），数据块长度在 **LEN** 中指定。

在 **SFB** 中，还可指定从中获取数据的伙伴区域。此信息由 CPU 输入到 **RK 512** 消息帧头中，并传输给伙伴（请参阅『使用 **RK 512** 计算机连接传输数据 — 基本信息（页码 319）』一节）。

从中获取第一字节的伙伴区由 CPU 号 **R\_CPU**（仅与多处理器通讯有关）、数据类型 **R\_TYPE**（数据块、扩展数据块、存储器位、输入、输出、计数器和定时器）、数据块号 **R\_DBNO**（仅与数据块和扩展数据块有关）和偏移量参数 **R\_OFFSET** 指定。

伙伴 CPU 的处理器间通讯标志字节和位在 **R\_CF\_BYT** 和 **R\_CF\_BIT** 中指定。

在 **SYNC\_DB** 参数中，指定要在其中存储数据的 DB，这些数据对于所有用于启动初始化和同步例程的 **SFB** 来说是公用的。DB 号对于用户程序中的所有 **SFB** 都必须相同。

必须通过 **R**（复位）= **FALSE** 调用 **SFB** 以使其可处理请求。在控制输入 **R** 的正跳沿上，当前传输被取消，**SFB** 复位到初始状态。被取消的请求结束，并带有错误消息（**STATUS** 输出）。

在 **LADDR** 中，输入您在“**HW Config**”中指定的子模块 I/O 地址。

如果请求已关闭且没有错误，则 **DONE** 置位为 TRUE，或者如果请求已关闭且具有错误，则 **ERROR** 置位为 TRUE。

如果出现错误或警告，**STATUS** 将显示相应的事件编号。

如果 SFB 复位 (R = TRUE)，也会输出 DONE 或 ERROR/STATUS。

如果出现错误，则二进制结果 BIE 复位。如果块结束且无错误，则二进制结果的状态为 TRUE。

**说明**

SFB 没有参数检查。如果编程不正确，CPU 可能会跳转到 STOP 模式。

如果要从 CPU 获取数据，则必须在 CPU 上对“SERVE\_RK”SFB 编程。

参数	声明	数据类型	说明	值范围	缺省
SYNC_DB	IN	INT	DB 号，其中存储了用于 RK SFB 同步的公用数据（最小长度为 240 字节）。	依 CPU 而定，不允许为零。	0
REQ	IN	BOOL	控制参数“请求”： 在正跳沿激活数据交换	TRUE/FALSE	FALSE
R	IN	BOOL	控制参数“复位”： 取消请求	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址。	依 CPU 而定	3FF（十六进制）
R_CPU	IN	INT	伙伴 CPU 号（仅用于多处理器模式）	0 到 4	1
R_TYPE	IN	CHAR	伙伴 CPU 上的地址类型（仅大写字母！）： <ul style="list-style-type: none"> <li>• ‘D’ = 数据块</li> <li>• ‘X’ = 扩展数据块</li> <li>• ‘M’ = 存储器位</li> <li>• ‘E’ = 输入</li> <li>• ‘A’ = 输出</li> <li>• ‘Z’ = 计数器</li> <li>• ‘T’ = 定时器</li> </ul>	‘D’、‘X’、 ‘M’、‘E’、 ‘A’、‘Z’、 ‘T’	‘D’
R_DBNO	IN	INT	伙伴 CPU 上的数据块号	0 到 255	0
R_OFFSET	IN	INT	伙伴 CPU 上的数据字节号	参见表：“FB 中用于数据源（伙伴 CPU）的参数”	0
R_CF_BYT	IN	INT	伙伴 CPU 的处理器间通讯标志字节 (255: 表示: 没有处理器间通讯标记)	0 到 255	255
R_CF_BIT	IN	INT	伙伴 CPU 的处理器间通讯标志位	0 到 7	0

参数	声明	数据类型	说明	值范围	缺省
<b>DONE</b>	OUT	BOOL	状态参数（仅在一次调用期间设置）： <ul style="list-style-type: none"> <li>FALSE：请求尚未启动或仍在执行。</li> <li>TRUE：请求已完成且没有错误。</li> </ul>	TRUE/FALSE	FALSE
<b>ERROR</b>	OUT	BOOL	状态参数（仅在一次调用期间设置）： 请求已完成且有错误。	TRUE/FALSE	FALSE
<b>STATUS</b>	OUT	WORD	状态参数（仅在一次调用期间设置。为了显示 STATUS，应将 STATUS 复制到空闲数据区）： 根据 ERROR 位，STATUS 具有以下意义： <ul style="list-style-type: none"> <li>ERROR = FALSE： STATUS 值为： <ul style="list-style-type: none"> <li>0000H：既没有警告也没有错误</li> <li>&lt;&gt; 0000H：警告，STATUS 提供详细信息</li> </ul> </li> <li>ERROR = TRUE： 错误未决。STATUS 提供有关错误类型的详细信息。</li> </ul>	0 到 FFFF (十六进制)	0
<b>RD_1</b>	IN_OUT	ANY	接收参数： 在此指定： <ul style="list-style-type: none"> <li>存储所获取的数据的 DB 号。</li> <li>存储所获取的数据的起始数据字节号。</li> </ul> 例如：DB10（从字节 2 开始）-> DB10.DBB2	依 CPU 而定	0
<b>LEN</b>	IN_OUT	INT	在此指定要获取的消息帧的字节长度。 (在此间接指定长度。) 每个定时器和计数器必须指定两字节的长度。	1 到 1024	1

## 数据一致性

数据一致性限制为 128 字节。要传输超过 128 字节的一致性数据，请注意以下几点：  
传输过程结束前，不能向接收区 RD\_1 的当前使用部分写入数据。这就是状态参数 DONE = TRUE 时的情形。

### （扩展）数据块的特性

注意以下从 DB 和扩展 DB “获取数据”的特性：

- RK 512 仅允许获取偶数数据。如果指定奇数数据作为长度 (LEN)，则始终会传输一个额外字节。但在目标 DB 中输入的是正确的数据长度。
- RK 512 仅允许指定偶数偏移量。如果指定奇数偏移量，则在伙伴上，数据从下一个较小的偶数偏移量开始存储。

实例：偏移量是 7，数据从字节 6 开始获取。

### 定时器和计数器的特性

从通讯伙伴获取定时器和计数器时，必须注意要从每个定时器和计数器获取两字节。例如，如果要获取 10 个计数器，必须声明长度为 20。

### SFB 中用于数据源（伙伴 CPU）的参数

下表列出了可传输的数据类型。

伙伴 CPU 上的源	R_TYPE	R_DBNO	R_OFFSET** (以字节为单位)
数据块	'D'	0 - 255	0 - 510*
扩展数据块	'X'	0 - 255	0 - 510*
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
*: 仅偶数值适用!			
**: 此值由伙伴 CPU 指定。			

### 消息帧头中的信息

下表显示了 RK 512 消息帧头中的信息（请参阅『使用 RK 512 计算机连接传输数据 — 基本信息（页码 319）』一节）。

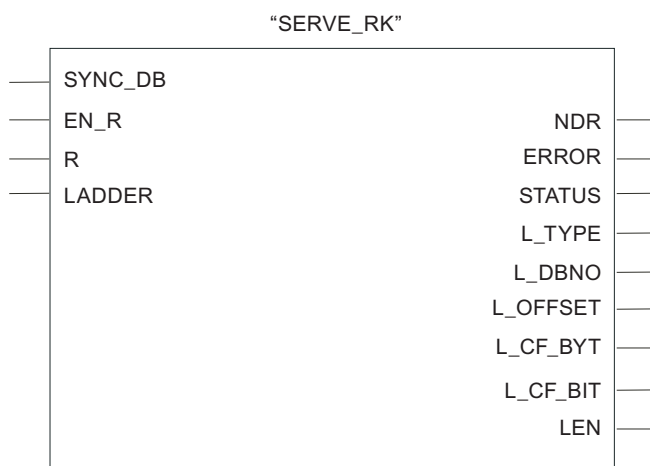
伙伴 CPU 上的源	至目标, S7 自动化系统 (本地 CPU)	消息帧头		
		字节 3/4: 命令类型	字节 5/6: Q-DBNR/Q-Offset	字节 7/8: 计量单位
数据块	数据块	ED	DB/DW	字
扩展数据块	数据块	EX	DB/DW	字
存储器位	数据块	EM	字节地址	字节
输入	数据块	EE	字节地址	字节
输出	数据块	EA	字节地址	字节
计数器	数据块	EZ	计数器号	字
定时器	数据块	ET	定时器号	字
缩写:				
Q-DBNR: 源数据块号				
Q-Offset: 源起始地址				

### 6.5.2.4 用 SFB 65 “SERVE\_RK” 接收/提供数据

#### 基本信息

使用此 SFB

- **接收数据:** 数据存储在由伙伴在 RK 512 消息帧头中指定的数据区内（请参阅『使用 RK 512 计算机连接传输数据 — 基本信息（页码 319）』一节）。当通讯伙伴执行“发送数据”请求（SEND 请求）时，需要 SFB 调用。
- **提供数据:** 从伙伴在 RK 512 消息帧头中指定的数据区中获取数据（请参阅『使用 RK 512 计算机连接传输数据 — 基本信息（页码 319）』一节）。当通讯伙伴执行“获取数据”请求（FETCH 请求）时，需要 SFB 调用。



用控制输入 **EN\_R** 上的值 TRUE 调用块之后，该块就处于准备好状态。可以通过参数 **EN\_R** 上的信号状态 FALSE 来取消当前传输。被取消的请求结束时带有错误消息（**STATUS** 输出）。只要参数 **EN\_R** 的信号状态为 FALSE，就锁定接收。

在 **SYNC\_DB** 参数中，指定要在其中存储数据的 **DB**，这些数据对于所有用于启动初始化和同步例程的 **SFB** 来说是公用的。**DB** 号对于用户程序中的所有 **SFB** 都必须相同。

必须通过 **R**（复位）= FALSE 调用 **SFB** 以使其可处理请求。在控制输入 **R** 的正跳沿上，当前传输被取消，**SFB** 复位到初始状态。被取消的请求结束，并带有错误消息（**STATUS** 输出）。

在 **LADDR** 中，输入您在“HW Config”中指定的子模块 I/O 地址。

如果请求已关闭且没有错误，则 **NDR** 置位为 TRUE，或者如果请求已终止且具有错误，则 **ERROR** 置位为 TRUE。

如果一次调用的 **NDR = TRUE**，则块将指明数据存储位置或从参数 **L\_TYPE**、**L\_DBNO**、和 **L\_OFFSET** 中获取数据的位置。此外，在一次调用期间内，还显示参数 **L\_CF\_BYT**、**L\_CF\_BIT** 以及相应请求的长度 **LEN**。

如果出现错误或警告，**STATUS** 将显示相应的事件编号（请参阅『错误消息（页码 340）』一节）。

如果 **SFB** 复位 (**R = TRUE**)（参数 **LEN = 16#00**），也会输出 **NDR** 或 **ERROR/STATUS**。

如果出现错误，则二进制结果 **BIE** 复位。如果块结束且无错误，则二进制结果的状态为 **TRUE**。

**说明**

**SFB** 没有参数检查。如果编程不正确，CPU 可能会跳转到 **STOP** 模式。

参数	声明	数据类型	说明	值范围	缺省
<b>SYNC_DB</b>	IN	INT	DB 号，其中存储了用于 RK SFB 同步的公用数据（最小长度为 240 字节）。	依 CPU 而定，不允许为零。	0
<b>EN_R</b>	IN	BOOL	控制参数“启用接收”： 启用请求	TRUE/FALSE	FALSE
<b>R</b>	IN	BOOL	控制参数“复位”： 取消请求。	TRUE/FALSE	FALSE
<b>LADDR</b>	IN	WORD	在“HW Config”中指定的子模块 I/O 地址。	依 CPU 而定	3FF（十六进制）
<b>NDR</b>	OUT	BOOL	状态参数“新数据就绪”（仅在一次调用期间设置）： 请求完成，没有错误 <ul style="list-style-type: none"> <li>FALSE：请求尚未启动或仍在运行。</li> <li>TRUE：请求已成功完成。</li> </ul>	TRUE/FALSE	FALSE
<b>ERROR</b>	OUT	BOOL	状态参数（仅在一次调用期间设置）： 请求已完成且有错误。	TRUE/FALSE	FALSE
<b>STATUS</b>	OUT	WORD	状态参数（仅在一次调用期间设置。为了显示 <b>STATUS</b> ，应将 <b>STATUS</b> 复制到空闲数据区）： 根据 <b>ERROR</b> 位， <b>STATUS</b> 具有以下意义： <ul style="list-style-type: none"> <li><b>ERROR = FALSE:</b>  <b>STATUS</b> 值为：                             <ul style="list-style-type: none"> <li>0000H：既没有警告也没有错误</li> <li>&lt;&gt; 0000H：警告，<b>STATUS</b> 提供详细信息</li> </ul> </li> <li><b>ERROR = TRUE:</b>                              错误未决。<b>STATUS</b> 提供有关错误类型的详细信息（有关错误编号，请参阅『错误消息（页码 340）』一节）。</li> </ul>	0 到 FFFF （十六进制）	0

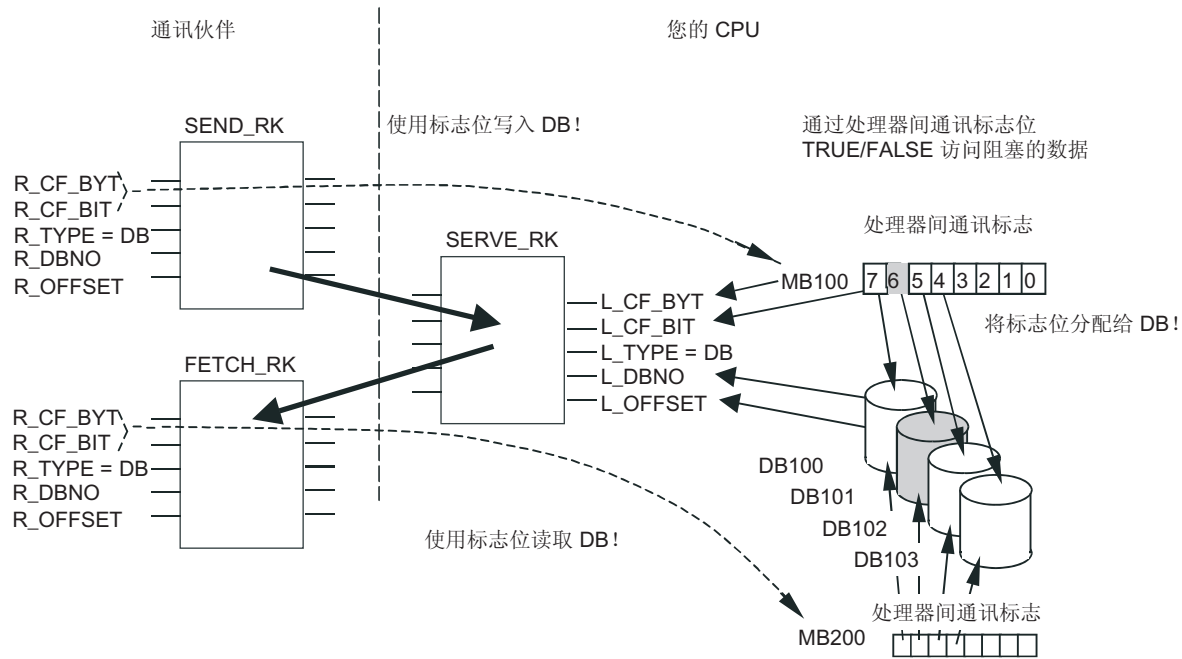
参数	声明	数据类型	说明	值范围	缺省
LEN	IN_OUT	INT	消息帧长度（字节数） （仅在一次调用期间设置）	0 到 1024	0
L_TYPE	OUT	CHAR	（L...参数仅在一次调用期间设置。） 接收数据： 本地 CPU 上目标区域的类型（仅大写字母）： ‘D’ = 数据块	‘D’	''
			提供数据： 本地 CPU 上源区域的类型（仅大写字母）： <ul style="list-style-type: none"> <li>• ‘D’ = 数据块</li> <li>• ‘M’ = 存储器位</li> <li>• ‘E’ = 输入</li> <li>• ‘A’ = 输出</li> <li>• ‘Z’ = 计数器</li> <li>• ‘T’ = 定时器</li> </ul>	‘D’、‘M’、 ‘E’、‘A’、 ‘Z’、‘T’	
L_DBNO	OUT	INT	本地 CPU 上的数据块号	依 CPU 而定	0
L_OFFSET	OUT	INT	本地 CPU 上的数据字节号	0-510	0
L_CF_BYT	OUT	INT	本地 CPU 的处理器间通讯标志字节 （255：表示：没有处理器间通讯标志）	0 到 255	0
L_CF_BIT	OUT	INT	本地 CPU 的处理器间通讯标志位	0 到 7	0

### 6.5.2.5 实例：使用处理器间通讯标志

#### 基本信息

通过处理器间通讯标志可阻塞和启用通讯伙伴的 SEND/FETCH 请求。这可防止覆盖和读取尚未处理的数据。

可为每个请求都指定一个处理器间通讯标志。



#### 也参见

使用 RK 512 计算机连接传输数据 — 基本信息 (页码 319)



### 6.5.2.6 实例：带处理器间通讯标志的 SEND\_RK

#### 步骤

在本例中，通讯伙伴将数据发送到您 CPU 中的 DB101。

1. 在您的 CPU 上，将处理器间通讯标志 100.6 置位为 FALSE。
2. 在通讯伙伴处，针对 SEND 请求指定处理器间通讯标志 100.6（参数 R\_CF\_BYT、R\_CF\_BIT）。

处理器间通讯标志将以 RK 512 消息帧头的形式发送到您的 CPU。

处理该请求之前，CPU 将检查于 RK 512 消息帧头中指定的处理器间通讯标志。仅当您 CPU 上的处理器间通讯标志状态为 FALSE 时，才处理该请求。如果处理器间通讯标志状态为 TRUE，则以响应消息帧的形式向通讯伙伴返回错误消息“32 十六进制”。

数据传送到 DB101 之后，SFB SERVE 将您 CPU 上的处理器间通讯标志 100.6 的状态置位为 TRUE，在一次调用期间内将标志字节和位输出到 SFB SERVE（如果 NDR = TRUE）。

3. 当您在用户程序中判断处理器间通讯标志（处理器间通讯标志 100.6 = TRUE）时，可以识别出请求已完成，发送的数据已准备好用于处理。
4. 在用户程序中编辑完数据之后，必须将处理器间通讯标志 100.6 复位为 FALSE。从而，通讯伙伴能重复执行请求而不会产生错误。

#### 数据一致性

数据一致性限制为 128 字节。对于传输超过 128 字节的一致数据，应注意以下几点：

使用处理器间通讯标志功能。在未发送完所有数据前不要访问数据（通过为此请求指定的处理器间通讯标志来判断；如果 NDR = TRUE，则在一次调用期间内，SFB 的处理器间通讯标志处于激活状态）。在未编辑完数据前不要将处理器间通讯标志状态复位为 FALSE。

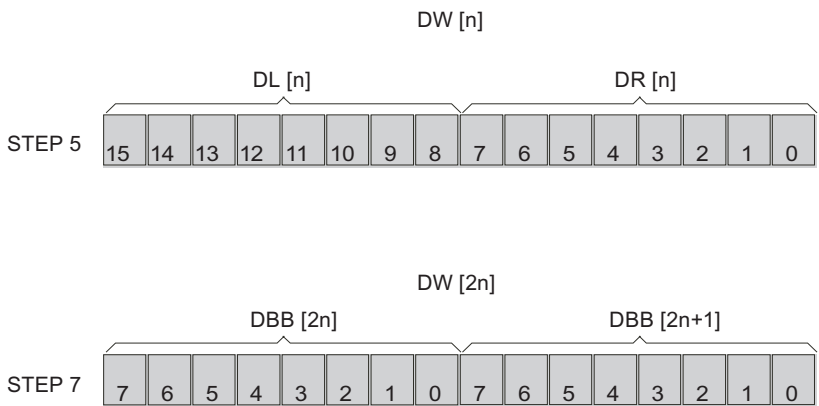
### 6.5.3 有关系统功能块编程的信息

#### 引言

本章面向要将 SIMATIC S5 升级到 SIMATIC S7 的所有用户。以下各节介绍在 STEP 7 中对 FB 编程时必须注意的事项。

#### 寻址

在 STEP 7 中，DB 中的数据操作数按字节寻址（相反，在 STEP 5 中是按字寻址。因此必须转换数据操作数的地址。）



与 STEP 5 不同，STEP 7 中的数据字地址是重复的，不再分为一个低数据字节和一个高数据字节。位始终从 0 到 7 进行编号。

#### 实例

STEP 5 数据操作数（表中左列）到 STEP 7 数据操作数（表中右列）的转换。

STEP 5	STEP 7
DW 10	DBW 20
DL 10	DBB 20
DR 10	DBB 21
D 10.0	DBX 21.0
D 10.8	DBX 20.0
D 255.7	DBX 511.7

## 分配块参数

### 直接/间接参数分配

STEP 5 中的间接参数分配（在当前打开的 DB 传递参数）不再适用于 STEP 7 块。

您可在所有块参数中声明常数及变量。因此，在 STEP 7 中不再需要区分直接参数分配与间接参数分配。

SFB 60、SFB 63 和 SFB 64 的参数“LEN”是一个例外：只能间接分配该参数。

### “直接参数分配”实例

根据“直接参数分配”调用 SFB 60 “SEND\_PTP”：

```

STL
Network 1:
  CALL SFB 60, DB10
    REQ      := M 0.6           //启动 SEND
    R        := M 5.0           //启动 RESET
    LADDR    := +336            //I/O 地址
    DONE     := M 26.0         //结束，没有错误
    ERROR    := M 26.1         //结束，出现错误
    STATUS   := MW 27          //状态字
    SD 1     := P#DB11.DBX0.0  //数据块 DB 11,
                                //自数据字节 DBB 0 开始
    LEN      := DB10.DBW20     //长度是间接分配的参数

```

### “当前操作数的符号寻址”实例

对于当前操作数的符号寻址调用 SFB 60 “SEND\_PTP”：

```

STL
Network 1:
  CALL SFB 60, DB10
    REQ      := SEND_REQ       //启动 SEND
    R        := SEND_R         //启动 RESET
    LADDR    := BGADR          //I/O 地址
    DONE     := SEND_DONE      //结束，没有错误
    ERROR    := SEND_ERROR     //结束，有错误
    STATUS   := SEND_STATUS    //状态字
    SD_1     := QUELLZEIGER    //指向目标区域的 ANY 指针
    LEN      := CPU_DB.SEND_LAE //消息帧长度

```

## 6.6 调试

### 6.6.1 调试接口硬件

#### 步骤

完成组态后，如果不能建立到伙伴设备的通讯，则应测试连接。步骤如下：

步骤	要执行的操作
1	查找出错原因： <ul style="list-style-type: none"> <li>• 发送/接收线路的极性是否接反？</li> <li>• 缺省设置是否正确？若干可能已执行的缺省设置可能缺省使用了不同的极性。有些缺省设置被永久集成在设备中。</li> <li>• 终端电阻是否缺少或有错误？</li> <li>• 是否颠倒了安全字（例如 CRC）的高低字节？</li> </ul>
2	步骤： <ul style="list-style-type: none"> <li>• 首先，借助手册检查电缆连接：               <ul style="list-style-type: none"> <li>– 分配/极性（请参阅『连接串行电缆（页码 253）』一节）</li> <li>– 缺省设置（请参阅『基本参数（页码 256）』一节）</li> </ul> </li> <li>• 然后测试每个测试设置</li> </ul>
3	尽量简单地准备测试设置： <ul style="list-style-type: none"> <li>• 仅互连 2 个节点</li> <li>• 如果可能，设置 RS485 模式（2 线操作）</li> <li>• 使用短的连接电缆</li> <li>• 短距离内无需终端电阻</li> <li>• 双向测试数据传输</li> </ul>
4	检查： <ul style="list-style-type: none"> <li>• 第 1 种情况：线路极性完全正确               <ul style="list-style-type: none"> <li>– 改变缺省设置（所有选项）</li> <li>– 检查安全字（例如 CRC）</li> </ul> </li> <li>• 第 2 种情况：缺省设置完全正确               <ul style="list-style-type: none"> <li>– 交叉链接连接（注意：对于 RS422，交叉链接两个线路对）</li> <li>– 检查安全字（例如 CRC）</li> </ul> </li> <li>• 第 3 种情况：不知道极性和缺省设置是否正确               <ul style="list-style-type: none"> <li>– 交叉链接连接（注意：对于 RS422，交叉链接两个线路对）</li> <li>– 如果不正确，更改缺省设置（所有选项），并尝试建立通讯</li> <li>– 如果不正确，将连接线路再更改回来，然后更改缺省设置（所有选项）</li> <li>– 检查安全字（例如 CRC）</li> </ul> </li> <li>• 随后重新装配系统时，不要忘记重新连接可能在先前移除的终端电阻</li> </ul>
5	其它提示： <ul style="list-style-type: none"> <li>• 如果可用，请将接口测试设备（如果需要 RS422/485 -&gt; V.24 转换器）互连到连接线路上。</li> <li>• 用测量设备检查信号电平（测量针脚 8 上的 GND 的信号电平）。</li> <li>• 如果正在接收数据而 CRC 安全字不正确，有些设备将不会发出已接收信号。</li> <li>• 如必要，请更换 CPU 以排除电气故障。</li> </ul>

## 6.7 错误处理和中断

### 6.7.1 错误定位和诊断

#### 诊断可能性

使用诊断功能可快速查找未决错误。提供了以下诊断方式：

- 系统功能块 (SFB) 的错误消息
- 对于 RK512：响应消息帧中的错误 ID
- 诊断中断

### 6.7.2 系统功能块 (SFB) 的错误消息

#### 基本信息

出现错误时，参数 **ERROR** 将置位为 **TRUE**。**STATUS** 参数指示出错原因。在『错误消息（页码 340）』一节中列出了可能的错误编号。

---

#### 说明

仅当置位 **ERROR** 位时才输出错误消息（请求已完成且有错）。在所有其它情况下，**STATUS** 字为零。如果置位了 **ERROR** 位，应将 **STATUS** 复制到空闲数据区以显示 **STATUS**。

---

### 6.7.3 响应消息帧中的错误 ID

#### 基本信息

如果您正在使用 RK 512 计算机连接，通讯伙伴处出现 SEND/FETCH 消息帧错误，则通讯伙伴将返回一个响应消息帧，其中第 4 字节含有错误 ID。

#### 错误 ID 表

在下表中可找到响应消息帧中的错误 ID (REATEL) 与通讯伙伴的 STATUS 中的事件类别/编号的分配情况。响应消息帧中的错误 ID 以十六进制值的形式输出。

REATEL	错误消息 (事件类别/事件编号)
0AH	0905H
0CH	0301H、0609H、060AH、0902H
10H	0301H、0601H、0604H
12H	0904H
14H	0903H
16H	0602H、0603H、090AH
2AH	090DH
32H	060FH、0909H
34H	090CH
36H	060EH、0908H

## 6.7.4 组态和判断诊断中断

### 基本信息

在到通讯伙伴的串行连接中出现断线时，可触发一个诊断中断 (080DH)。诊断中断随同进入和离开的错误事件一起显示。

在用户程序中，可借助诊断中断立即对错误作出响应。

### 步骤

1. 在参数分配窗口的“基本参数”对话框中启用诊断中断。
2. 可将诊断中断 OB (OB 82) 合并到用户程序中。

### 通过诊断中断对错误作出响应

- 当前操作不受诊断中断影响。
- CPU 操作系统将在用户程序中调用 OB 82。

---

#### 说明

如果未装载相应的 OB，则在触发中断后，CPU 会切换为 STOP。

---

- CPU 将打开 SF LED 指示灯。
- 在 CPU 的诊断缓冲区中将错误显示为“进入”和“离开”事件。

### 如何在用户程序中判断诊断中断

触发诊断中断后，可判断 OB 82 以检查哪个诊断中断还未处理。

- 如果在 OB 82 的字节 6 + 7 (OB 82\_MDL\_ADDR) 中写入了子模块的地址，则诊断中断是由 CPU 的 PtP 连接触发的。
- 只要队列中有错误，就会置位 OB 82 中字节 8 的位 0（故障模块）。
- 队列中的所有错误均报告为“离开”后，将复位 OB 82 中字节 8 的位 0。
- 如果串行连接中发生断线，将同时置位字节 8 中的“故障模块”、“断线”、“外部错误”和“通讯错误”位。

OB 82, 字节 8	说明
位 0	故障模块
位 1	-
位 2	外部错误
位 3	-
位 4	-
位 5	断线
位 6	-
位 7	-

OB 82, 字节 10	说明
位 0	-
位 1	通讯错误
位 2	-
位 3	-
位 4	-
位 5	-
位 6	硬件中断丢失
位 7	-

## 6.8 安装实例

### 使用实例

可在文档附带的 CD-ROM 中找到实例（程序和说明）。也可从 Internet 下载。项目由具有不同复杂度和针对性的多个带注释的 S7 程序组成。

CD 上的 Readme.wri 说明了安装实例的方法。安装完成后，实例即存储在  
 ...\\STEP7\\EXAMPLES\\ZDI26\_01\_TF\_\_\_\_31xC\_PtP 目录下。



## 6.9 协议说明

### 6.9.1 使用 ASCII 驱动程序传输数据

#### 6.9.1.1 使用 ASCII 驱动程序传输数据 – 基本信息

##### 基本信息

ASCII 驱动程序控制 CPU 和通讯伙伴之间通过 PtP 通讯进行的数据传输。

这为消息帧提供了一种开放式结构，因为 S7 用户将完整的发送消息帧传递给 PtP 接口。对于接收方向，必须在参数中声明消息帧的结束标准。发送消息帧的结构与接收消息帧的结构可能会不同。

ASCII 驱动程序允许发送和接收开放式结构的数据（所有可打印的 ASCII 字符以及所有其它从 00 到 FFh（带有 8 个数据位的字符帧）或从 00 到 7Fh（带有 7 个数据位的字符帧）的字符）。

操作适用于 RS422 和 RS485。

##### RS422 操作

在 RS422 模式中，通过四线串行电缆发送数据（4 线操作）。两根导线（差分信号）用于发送方向，另两根用于接收方向。这意味着可以同时发送和接收数据（全双工操作）。

##### RS485 操作

在 RS485 模式中，通过两线制串行电缆发送数据（2 线操作）。两根导线（差分信号）交替用于发送和接收方向。这意味着在给定的时间只能发送数据或接收数据（半双工操作）。完成发送操作后，电缆被立即切换至接收模式（发送器切换至高阻抗）。最长切换时间为 1 ms。

### 6.9.1.2 使用 ASCII 驱动程序发送数据

#### 基本信息

当调用 SFB 执行发送操作时，请在“LEN”参数中为要发送的用户数据指定字节数。

使用结束标准“字符延迟时间结束”时，在发送操作期间，ASCII 驱动程序还将在两个消息帧间生成一个等待时间。可以一直调用 SFB。不过，自上次传输消息帧以后，ASCII 驱动程序不会启动数据输出，直到时间间隔超过组态的字符延迟时间。

如果使用结束标准“固定字符长度”，将根据在 SFB SEND\_PTP 的参数“LEN”中声明的数据长度，按照发送方向发送数据。在接收方向（即在接收 DB 中），数据长度与在接收器中，通过参数分配窗口在“固定字符长度”参数中输入的数据长度相同。两个参数设置必须相同，以确保进行无错数据通讯。发送数据时，在两个消息帧间插入一个与缺少结束代码的监控时间等长的暂停，以便使伙伴同步（识别消息帧的启动）。

如果使用其它同步方法，可通过参数分配窗口取消激活发送暂停。

如果使用文本结束字符标准，可有以下三种选择：

1. 传输中包括文本结束字符：

结束代码必须包括在要发送的数据中。只能传输结束代码之前的所有数据，即使在 SFB 中指定了更大的数据长度。

2. 传输的数据长度最多等于在 SFB 参数中指定的长度：

传输的数据长度最多等于在 SFB 参数中声明的长度。最后一个字符必须是文本结束字符。

3. 传输的数据长度最多等于在 SFB 参数中组态的长度，并自动追加结束代码/s：

传输的数据长度最多等于在 SFB 参数中声明的长度。自动追加文本结束字符；即不得将结束代码含在要传输的数据中。根据结束代码的数目，向伙伴传输的字符数要比 SFB 中指定的字符数（最大 1024 字节）多 1 个或 2 个字符。

---

#### 说明

组态 XON/XOFF 流控制时，用户数据不得包含任何已组态的 XON 或 XOFF 字符。缺省设置为 DC1 = 11H(XON) 和 DC3 = 13H(XOFF)。

---

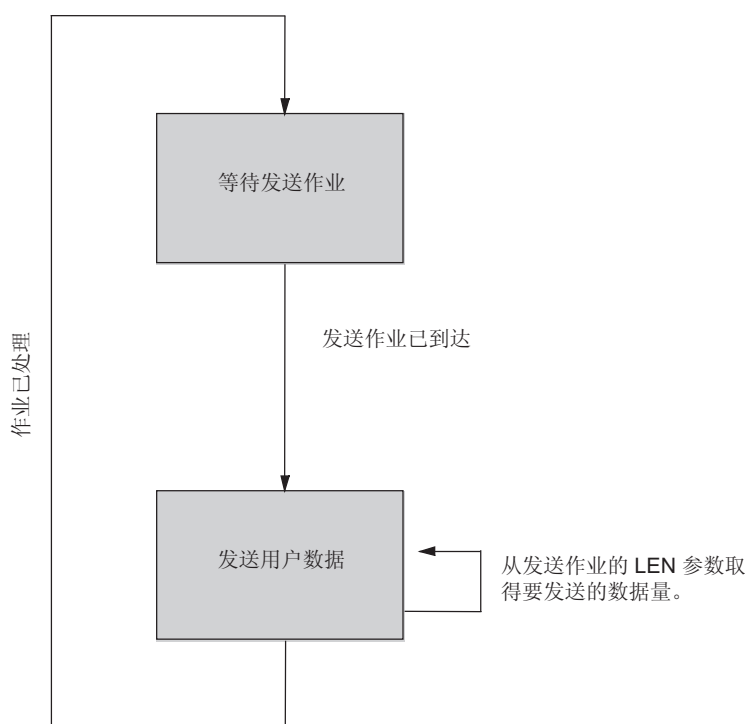
## 传输块检查字符

如果要使用一个（或两个）块检查字符 (BCC) 来确保数据的完整性，必须在结束标准“文本结束字符”中使用设置“发送的数据长度最多等于在 SFB 参数中声明的长度”。随后可将一个（两个）附加的块检查字符追加到发送的文本结束字符之后。

必须直接在用户程序中计算块检查字符。

## 发送数据

下图说明了一个发送操作：



### 6.9.1.3 使用 ASCII 驱动程序接收数据

#### 基本信息

对于使用 ASCII 驱动程序的数据传输，可在三种不同的结束标准中进行选择。结束标准用于定义完整接收消息帧的位置。可组态的结束标准有：

- 字符延迟时间结束：  
消息帧没有固定长度也没有定义的文本结束字符；消息结束由线路暂停（字符延迟时间结束）来定义。
- 接收固定数目的字符：  
接收的消息帧的长度始终相同。
- 接收文本结束字符：  
消息帧的结束由一个或两个定义好的文本结束字符来标记。

#### 明码性

程序的明码性取决于所组态的结束标准和流控制的选择：

- 带有一个（两个）结束字符：  
非明码性
- 结束标准为字符延迟时间或固定字符长度：  
明码性
- 明码性操作不适用于 XON/XOFF 流控制操作。  
“明码性”表示用户数据可以包含任何字符组合，无需识别结束标准。

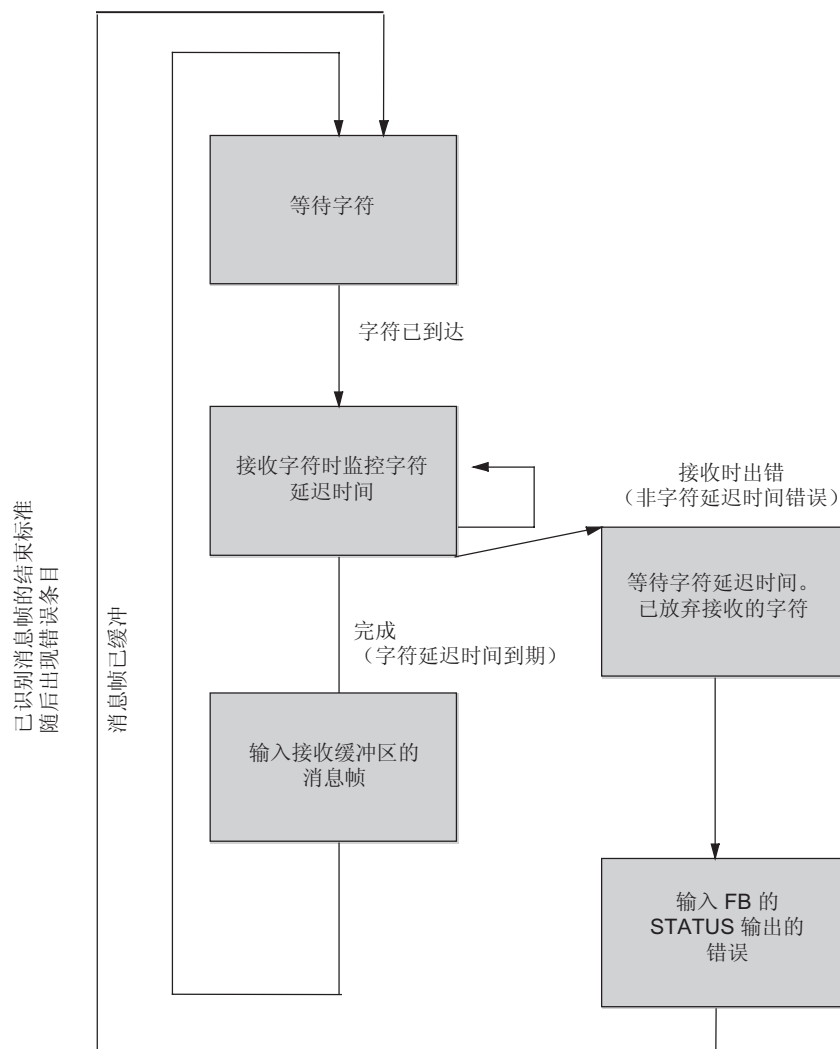
#### 结束标准“字符延迟时间结束”

接收数据时，如果字符延迟时间结束则识别为消息帧结束。CPU 接受所接收的数据。

在这种情况下，必须组态字符延迟时间，以确保两个连续消息帧间相隔该段时间。不过，该时间应该足够长，以便通讯伙伴在一个消息帧内执行发送暂停时，不会错误地识别该消息帧已结束。

### 步骤

下图说明了结束标准为“在字符延迟时间结束时”的接收操作：



### 结束标准“固定字符长度”

接收数据时，达到所声明的字符数之后识别为消息帧结束。CPU 接受所接收的数据。

如果在达到所声明的字符数之前，字符延迟时间已结束，则接收操作停止。在这种情况下，字符延迟时间被用作监控时间。生成一条错误消息并丢弃消息帧碎片。

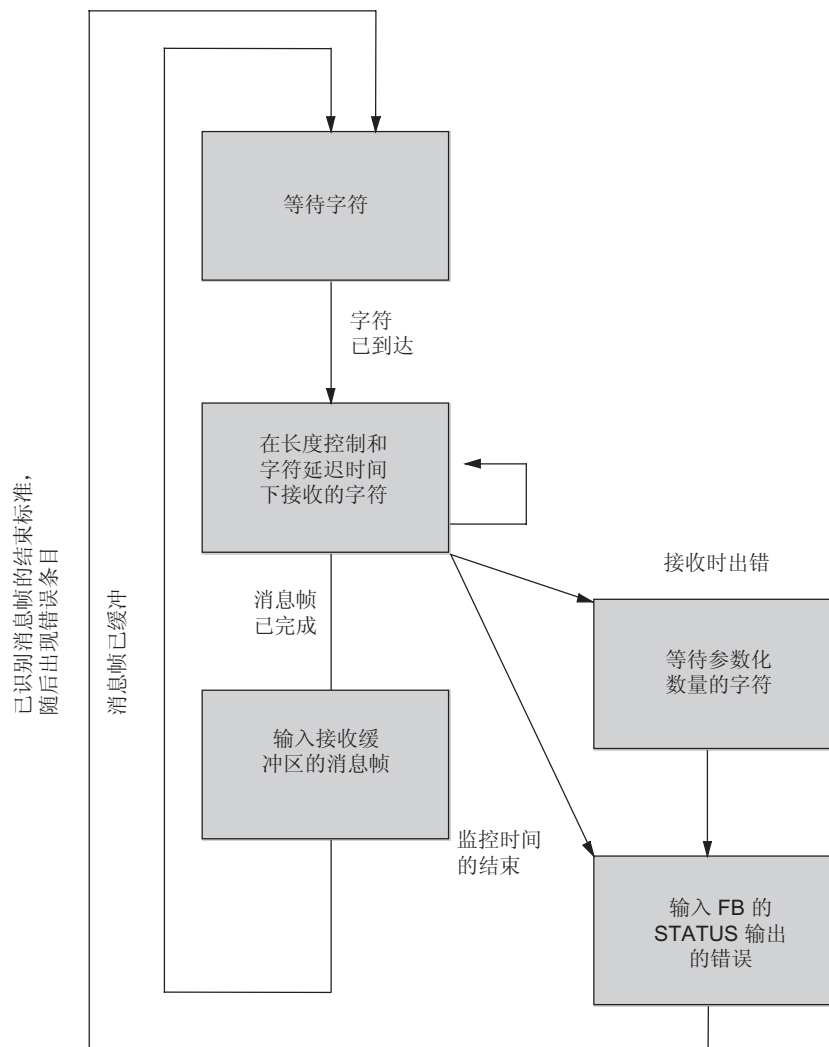
### 特殊注意事项

如果字符长度与组态的固定长度不相符，请注意以下几点：

- 已接收的字符长度超过组态的固定长度：  
达到组态的固定字符长度之后，接收的所有字符将：
  - 被丢弃（如果在消息帧结束时监视时间已结束）。
  - 与下一个消息帧合并（如果在监视时间结束前接收了新的消息帧）。
- 已接收的字符长度少于组态的固定长度：  
消息帧将：
  - 被丢弃（如果在消息帧结束时监视时间已结束）。
  - 与下一个消息帧合并（如果在监视时间结束前接收了新的消息帧）。

### 步骤

下图说明了结束标准为“固定字符长度”的接收操作：



### 结束标准“文本结束字符”

接收数据时，接收到所声明的文本结束字符时识别为消息帧结束。有以下几种选择：

- 一个文本结束字符
- 两个文本结束字符

CPU 接受所接收的、包括文本结束字符的数据。

如果接收的数据中缺少结束代码，则在接收期间字符延迟时间结束时，该帧将终止。在这种情况下，字符延迟时间被用作监控时间。发出一条错误消息并丢弃消息帧碎片。

如果使用文本结束字符，则传输为非明码性。还要必须确保结束代码不含在用户的用户数据中。

#### 特别注意事项

如果接收的消息帧中的最后一个字符不是文本结束字符，请注意以下几点：

- 文本结束字符在消息帧中的其它位置：

所有字符（包括文本结束字符）都被写入 DB。如果在消息帧结束时监控时间已结束，在文本结束字符之后的字符将：

- 被丢弃（如果在消息帧结束时监视时间已结束）。
- 与下一个消息帧合并（如果在监视时间结束前接收了新的消息帧）。

- 文本结束字符不在消息帧中：

消息帧将：

- 被丢弃（如果在消息帧结束时监视时间已结束）。
- 与下一个消息帧合并（如果在监视时间结束前接收了新的消息帧）。

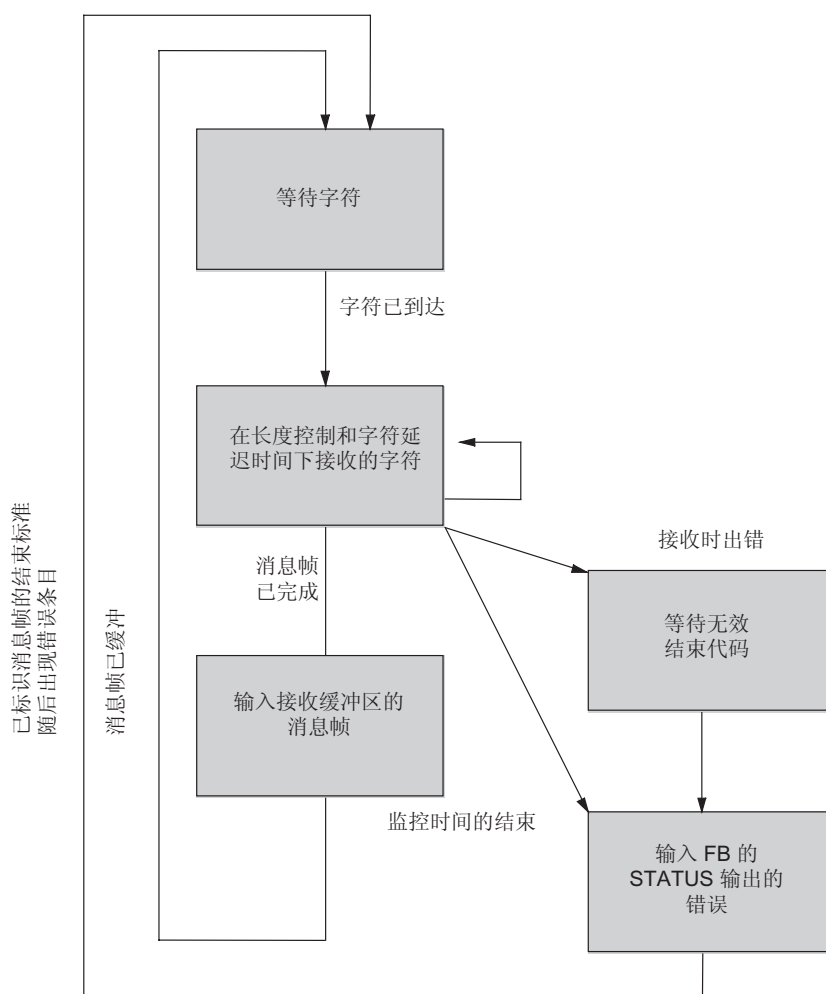


## 通过块检查字符接收

在参数分配窗口中，除文本结束字符外，还可选择使用一个（或两个）块检查字符 (BCC)。附加在文本结束字符后面的字符（一个或两个）也被写入接收 DB 中。

必须直接在用户程序中计算块检查字符。

下图说明了结束标准为“文本结束字符”的接收操作：



## CPU 的接收缓冲区

接收缓冲区大小为 2048 字节。在组态中，可以选择防止覆盖缓冲区中的数据。也可以为缓冲的已接收帧数指定值范围（1 到 10），或使用整个接收缓冲区。

可在启动时清除接收缓存区。您可以在参数分配窗口中指定此设置，也可以调用 SFB RES\_RCV 进行设置（请参阅『用 SFB 62 “RES\_RCVB” 清除接收缓冲区（页码 274）』一节）。

接收缓冲区是个环形缓冲区：

- 如果将多个消息帧写入接收缓冲区：总是将第一个收到的消息帧发送到目标块。
- 如果您始终要将最后收到的消息帧传输到目标数据块，则必须为缓冲的消息帧数分配值“1”，并取消激活覆盖保护。

---

### 说明

如果在某个时刻，在用户程序中连续读取接收数据时被中断，并且请求新的接收数据，则旧的消息帧可能会在最后一个消息帧之前写入目标块。

旧消息帧是中断时在 CPU 和伙伴之间的在途帧，或是已经由 SFB 收到的消息帧。

---

### 6.9.1.4 数据流控制/握手

#### 基本信息

握手用于控制两个通讯伙伴之间的数据流。握手可确保在以不同速度运行的设备之间进行通讯时不会丢失数据。CPU 通过 XON/XOFF 支持软件握手。

#### 步骤

实现数据流控制的步骤如下：

1. 将 CPU 组态为使用流控制后，CPU 发送 XON 字符。
2. 当接收缓冲区（接收缓冲区的大小：2,048 字节）在上溢之前接收到了声明的消息帧数，或者 50 个字符时，CPU 将发送 XOFF 字符。如果通讯伙伴仍然继续发送数据，则当接收缓冲区上溢时将生成一条错误消息。在最后一个消息帧中接收到的数据将被丢弃。
3. 在从接收缓冲区取出消息帧后，接收缓冲区准备接收数据时，CPU 将发送 XON 字符。
4. 当 CPU 收到 XOFF 字符时，将中断数据传输。如果在指定组态时间内未收到 XON，则取消发送操作，并在 SFB 的 STATUS 输出中生成相应的错误消息 (0708H)。

## 6.9.2 用 3964(R) 程序传输数据

### 6.9.2.1 用 3964(R) 程序传输数据 – 基本信息

#### 基本信息

CPU 和通讯伙伴之间的 PtP 通讯中，3964(R) 程序控制数据传输。

#### 控制字符

对于数据传输，3964(R) 计算机连接在用户数据的后面附加控制字符。利用这些控制字符，通讯伙伴可验证数据是否全部达到且没有错误。

3964(R) 程序分析下列控制代码：

- STX: 文本开始，要发送的字符串的起始部分
- DLE: 数据链接转换（数据发送切换）或肯定确认
- ETX: 文本结束，要发送的字符串的结束部分
- BCC: 块检查字符（仅用于 3964(R)）；块检查字符
- NAK: 否定应答

---

#### 说明

即将建立和关闭连接时，作为信息字符发送的 DLE 字符将在整个发送线路中传输两次（DLE 重复），以便将其与 DLE 控制字符区分开来。接收器将取消此 DLE 重复。

---

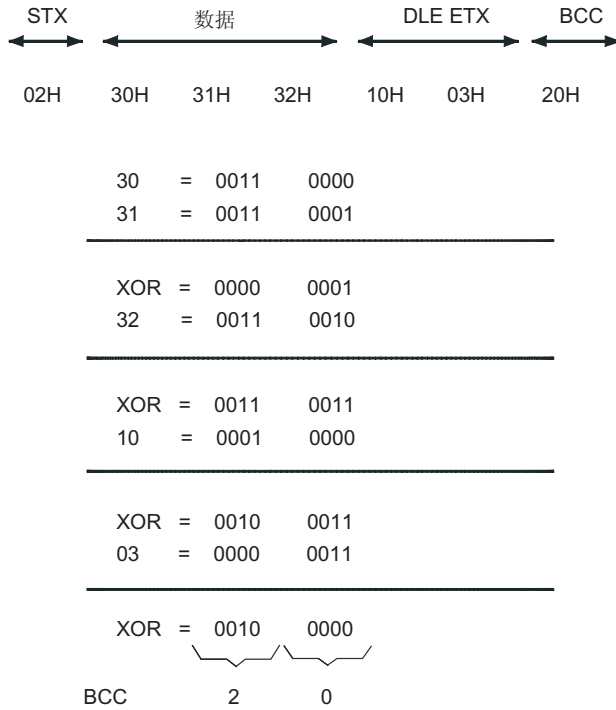
#### 优先级

在 3964(R) 程序中，必须为一个通讯伙伴指定较高的优先级，为另一个伙伴指定较低的优先级。如果两个伙伴同时开始建立连接，则低优先级的伙伴将推迟其发送请求。

### 块检验和

使用 3964(R) 数据传输协议时，通过发送附加的块检查字符（BCC = 块检查字符）可增强数据完整性。

消息帧：



块检验和表示发送或接收块的偶纵向校验（所有数据字节的 XOR 链接）。计算从建立连接后的用户数据的第一字节（消息帧的第一字节）开始，在断开连接时的 DLE ETX 代码之后结束。

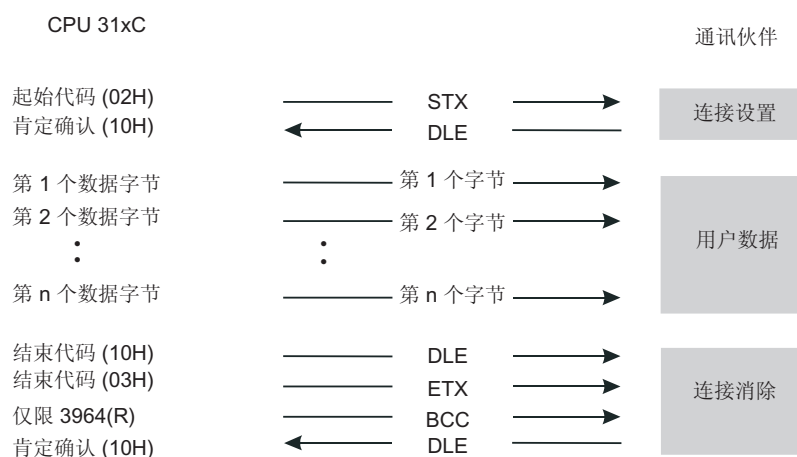
#### 说明

通过 DLE 重复，DLE 代码被两次包括在 BCC 计算中。

## 6.9.2.2 用 3964(R) 发送数据

### 步骤

下图说明使用 3964 (R) 程序发送数据时的数据流:



### 在数据传输期间建立连接

要建立连接, 3964 (R) 程序需发送控制字符 **STX**。如果通讯伙伴在应答延迟时间 (ADT) 结束之前用 **DLE** 字符响应, 程序将切换至发送模式。

如果通讯伙伴返回 **NAK** 或任何其它控制代码 (除 **DLE** 或 **STX** 外), 或者如果应答延迟时间结束但未发出响应, 程序将重新尝试连接。在尝试连接的次数达到了声明的不成功次数后, 程序中止建立连接的尝试, 并向通讯伙伴发送一个 **NAK**。CPU 向 **SFB SEND\_PTP** (输出参数 **STATUS**) 报告该错误。

### 发送数据

成功建立连接后, 使用选定的传输参数向通讯伙伴传输数据。伙伴将监控进入的字符之间的时间间隔。两个字符间的间隔不得超过字符延迟时间。

如果通讯伙伴在繁忙的会话期间发送 **NAK**, 程序将中止该块并如上所述重新尝试开始建立连接。如果发送了另外一个字符, 程序将首先等待字符延迟时间结束, 然后发送 **NAK** 字符, 将通讯伙伴切换至空闲状态。随后, 通过建立连接 **STX**, 程序重新启动发送操作。

### 发送时关闭

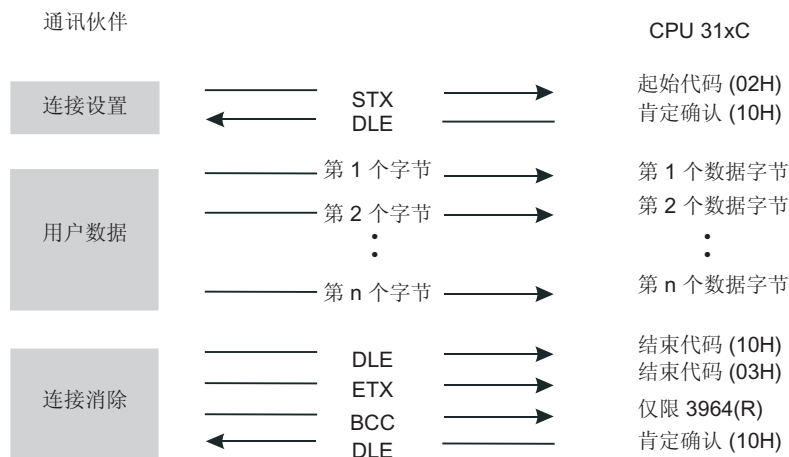
发送完缓冲区中的内容后, 程序将追加 **DLE ETX** 字符 (仅对于 3964(R), 将块检验和 **BCC** 作为结束代码), 随后等待应答字符。如果通讯伙伴在应答延迟时间内发送了 **DLE**, 则表明已接收数据块且没有错误。如果通讯伙伴用 **NAK**、任何其它代码 (除 **DLE** 外)、一个损坏的字符响应, 或者如果应答延迟时间结束但未发出响应, 程序将通过建立连接 **STX** 来重新启动发送数据操作。

在尝试发送数据块的次数达到组态的次数后, 程序将取消操作并向通讯伙伴发送 **NAK**。在 **SFB SEND\_PTP** (输出参数 **STATUS**) 中显示错误。

### 6.9.2.3 用 3964(R) 接收数据

#### 步骤

下图说明使用 3964 (R) 程序接收数据时的数据流:



#### 说明

操作就绪后, 3964(R) 程序发送一次 NAK 字符以便将通讯伙伴切换至空闲状态。

#### 接收数据时建立连接

在空闲模式中, 如果没有要处理的发送请求, 程序将等待通讯伙伴建立连接。

如果尝试通过 STX 建立连接时没有可用的空接收缓冲区, 则启动 400 ms 的等待时间。如果经过该段时间后, 仍没有空的接收缓冲区, SFB 的 STATUS 输出将显示错误。程序会发送一个 NAK 字符并返回到空闲模式。否则, 程序会发送一个 DLE 字符并接收数据。

如果空闲模式下的程序接收了任何字符 (STX 或 NAK 除外), 它将等待字符延迟时间结束, 然后发送 NAK 字符串。错误显示在 SFB 的 STATUS 输出中。

## 接收数据

成功建立连接之后，进入的用户数据被写入接收缓冲区。在接收到的两个连续 DLE 字符中，仅将其中一个存储在接收缓冲区中。

在接收每个字符后，期望在字符延迟时间内接收到下一个字符。如果在接收另一个字符前，此间隔已结束，则向通讯伙伴发送一个 NAK。系统程序将错误报告给 SFB RCV\_PTP（输出参数 STATUS）。

如果在接收期间出现传输错误（丢失字符、帧错误、奇偶校验错误等），程序将继续接收数据，直到连接断开。然后，向通讯伙伴发送一个 NAK。随后期望重新尝试连接。如果经过了静态参数记录中所声明的重试次数后仍然不能接收到没有错误的块，或者如果在指定的块检查时间（对应于应答延迟时间）之内通讯伙伴未启动重试操作，程序将取消接收操作。CPU 向 SFB RCV\_PTP（输出参数 STATUS）报告第一个错误传输和最后的取消操作。

## 接收数据时关闭连接

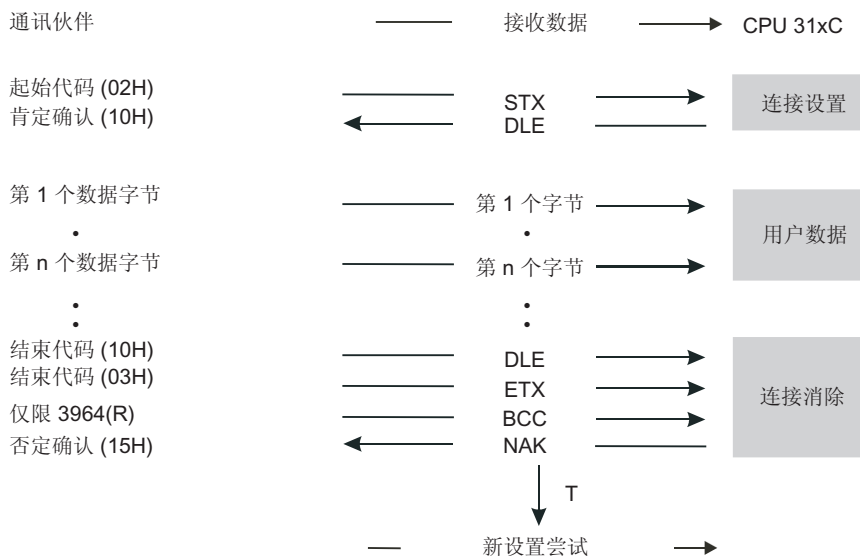
当 3964 程序检测到 DLE ETX 字符串时，如果块接收无误，它将取消接收操作并向通讯伙伴发送一个 DLE。如果在接收数据时出现错误，它将发送一个 NAK。随后期望重新尝试连接。

3964(R) 程序检测到字符串 DLE ETX BCC 时，停止接收。它将接收到的 BCC 同内部计算的纵向奇偶性加以比较。如果 BCC 正确且未出现其它接收错误，3964(R) 将发送一个 DLE 并返回到空闲模式。如果 BCC 错误或者出现其它接收错误，则向通讯伙伴发送一个 NAK。随后期望重新尝试连接。

### 6.9.2.4 用 3964 (R) 程序发送和接收时的错误处理

#### 处理损坏的数据

下图显示了如何用 3964(R) 处理损坏的数据：



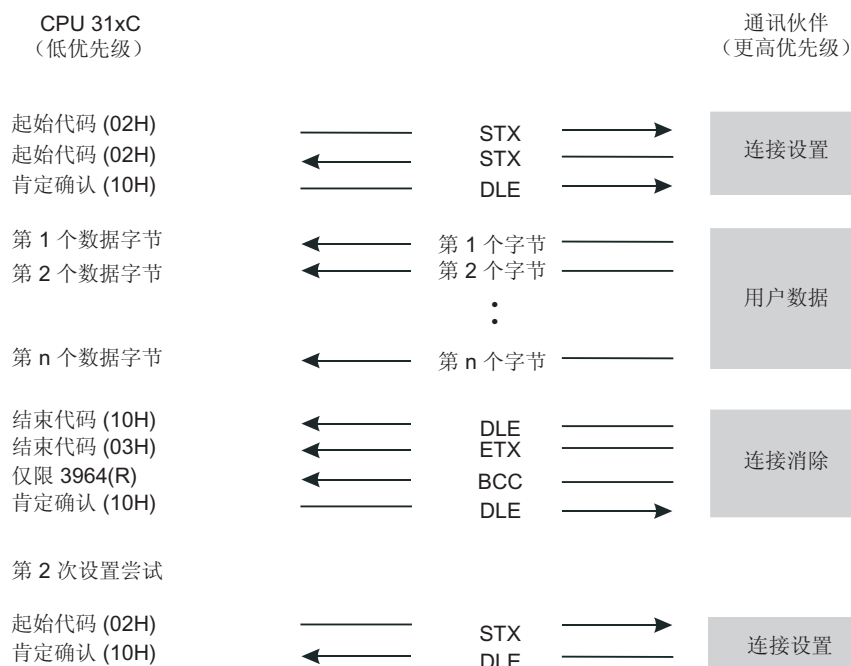
接收到 DLE、ETX 和 BCC 之后，CPU 将通讯伙伴的 BCC 与其内部计算的值进行比较。如果 BCC 正确且未出现其它接收错误，CPU 将用 DLE 响应。

否则，它将用 NAK 响应并在块检查时间内重试。如果在组态的尝试次数内不能接收到该块，或者如果在块检查时间内未进行进一步的尝试，它将取消接收操作。



## 初始化冲突

下图说明发生初始化冲突时的数据流：



如果一个设备在应答延迟时间 (ADT) 内通过发送代码 STX 而不是发送确认 DLE 或 NAK 来响应通讯伙伴的发送请求（代码 STX），则将出现初始化冲突。两个设备都请求发送。低优先级的设备推迟其发送请求并用代码 DLE 响应。高优先级的设备将按如上所述发送其数据。一旦关闭连接，低优先级设备便可执行其发送请求。

要解决初始化冲突，必须为通讯伙伴组态不同的优先级。

## 程序错误

程序可识别由通讯伙伴的错误操作及线路干扰所引起的错误。

在这两种情况下，程序都将重试正确地发送/接收数据块。如果在设置的最大传输尝试次数内无法实现（或者如果出现新的错误状态），程序将取消发送或接收过程。它生成所识别的第一个错误的错误 ID，然后返回至空闲模式。这些错误消息显示在 SFB STATUS 输出上。

如果 SFB 的 STATUS 输出指示频繁重复的发送和接收尝试错误，则可以认为是数据传输的临时中断。不过，大量地进行传输尝试将弥补这种情形。在这种情况下，建议检查通讯链接以查找可能的干扰源，因为频繁重复将降低用户数据的传输率和传输完整性。干扰也可能由通讯伙伴方的故障而引起。

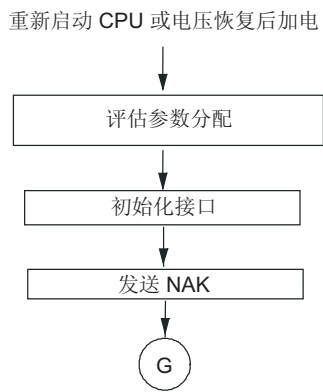
如果接收线路上出现 BREAK 状态（接收线路中断），将在 SFB 的 STATUS 输出上显示一条错误消息。不启动任何重试。重新连接线路后，BREAK 状态将被自动复位。

对于识别的所有传输错误，将在接收数据块时报告统一的错误编号。但仅在重复尝试不成功之后才会报告该错误。

### 6.9.2.5 3964(R) 程序启动顺序

#### 步骤

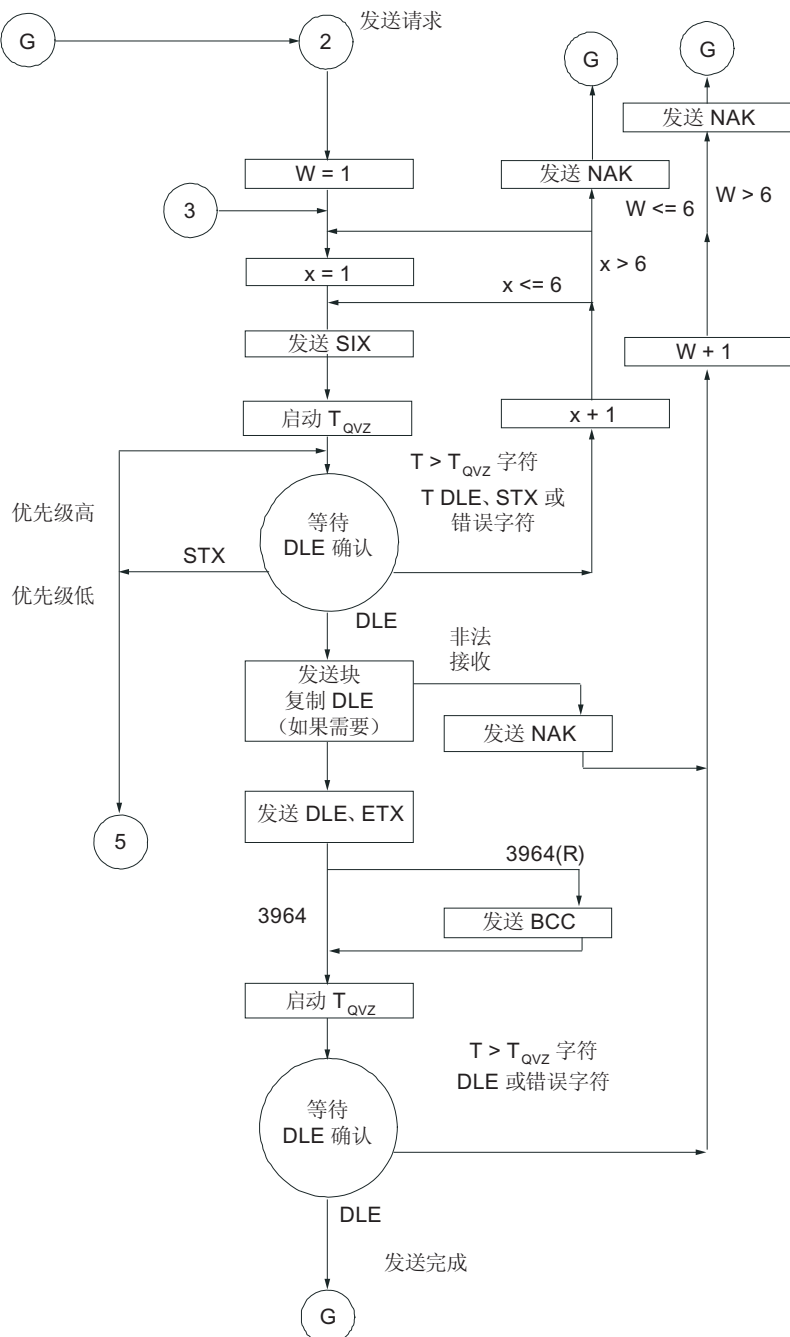
下图说明了 3964(R) 程序的启动顺序:



### 6.9.2.6 用 3964(R) 程序发送

#### 步骤

下图说明用 3964(R) 程序发送的顺序：



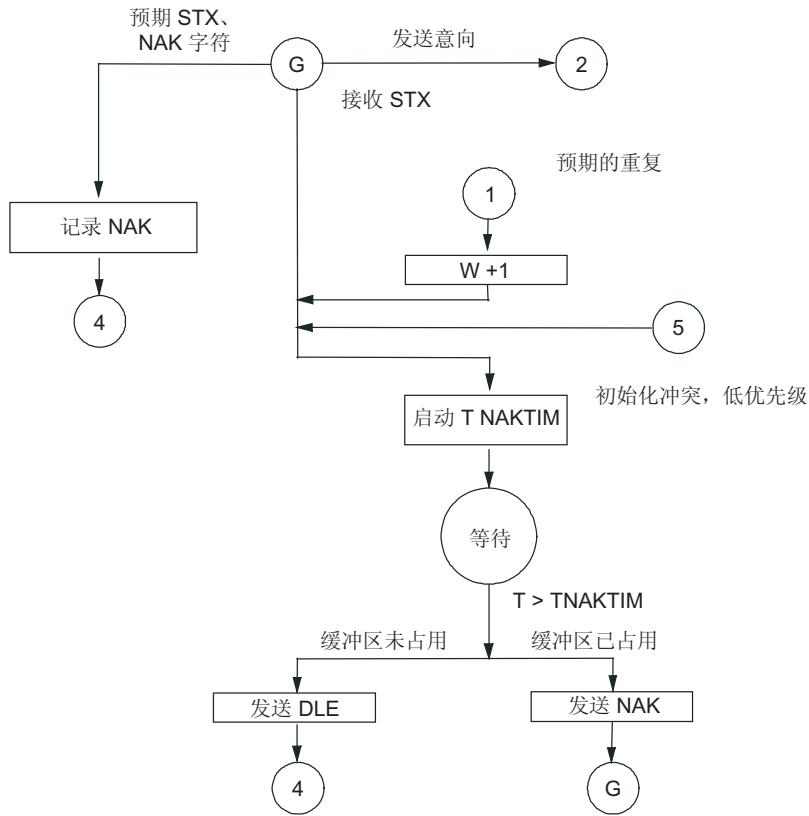
仅使用 3964(R) 的 BCC  
 $x$  = 计数器重新尝试连接的次数  
 $W$  = 计数器重新尝试传输的次数

$T_{qvz} = 500 \text{ ms}$  (3964(R)  $QVZ = 2\text{s}$ )  
 如果出现断线 BREAK, 则立即转为初始状态

### 6.9.2.7 用 3964(R) 程序接收

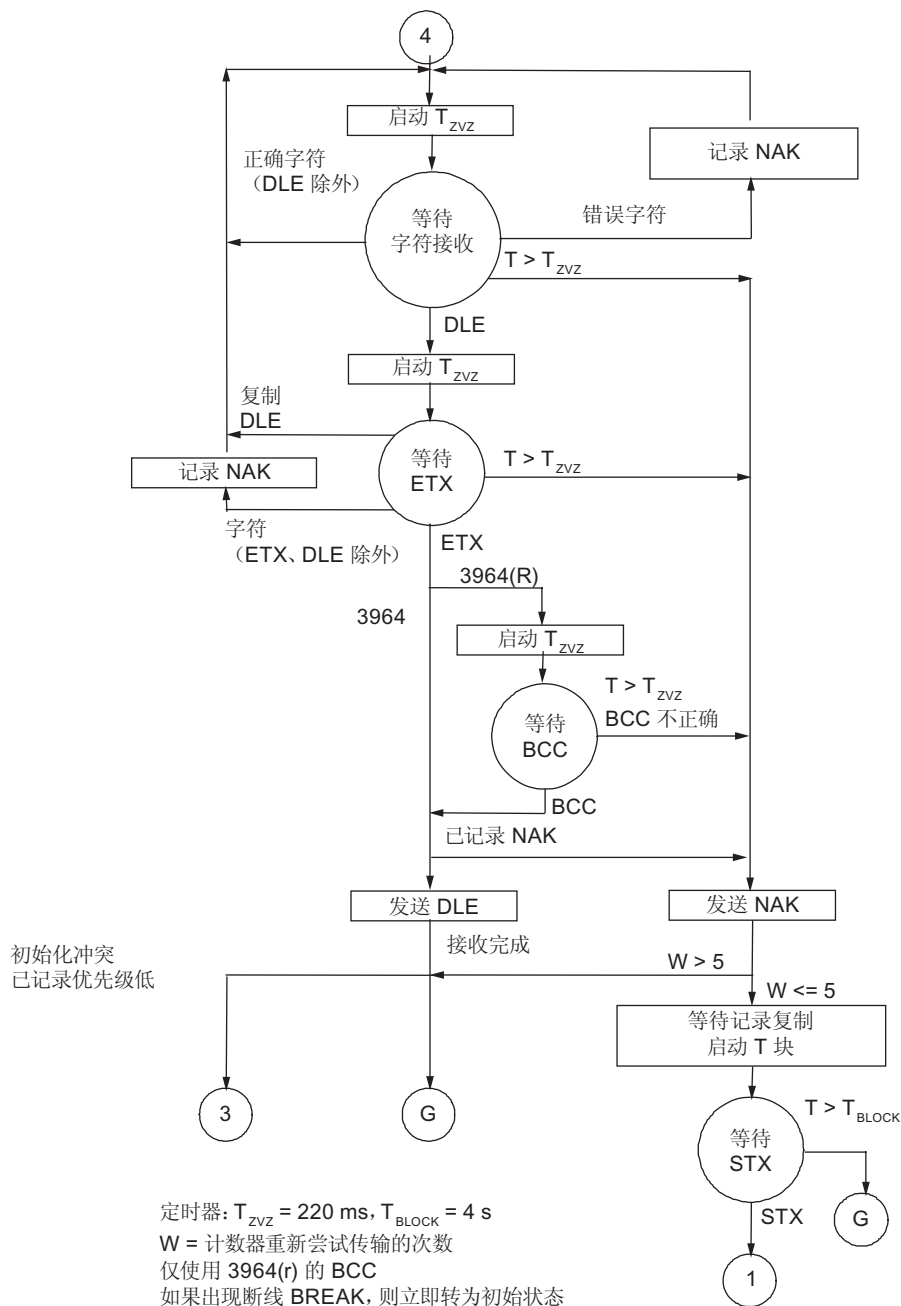
#### 用 3964(R) 程序接收（第 1 部分）

下图说明用 3964(R) 程序接收的顺序：



### 用 3964(R) 程序接收 (第 2 部分)

下图显示了使用 3964(R) 程序的接收顺序:



## CPU 的接收缓冲区

接收缓冲区大小为 2048 字节。在组态中，可以选择防止覆盖缓冲区中的数据。也可以为缓冲的已接收帧数指定值范围（1 到 10），或使用整个接收缓冲区。

可在启动时清除接收缓存区。您可以在参数分配窗口中指定此设置，也可以调用 SFB RES\_RCV 进行设置（请参阅『用 SFB 62 “RES\_RCVB” 清除接收缓冲区（页码 274）』一节）。

接收缓冲区是个环形缓冲区：

- 如果将多个消息帧写入接收缓冲区：总是将第一个收到的消息帧传送到目标 DB。
- 如果您始终要将最后收到的消息帧传输到目标数据块，则必须为缓冲的消息帧数分配值“1”，并取消激活覆盖保护。

---

### 说明

如果在某个时刻在用户程序中连续读取接收数据时中断，并且请求新的接收数据，则第一个接收的消息帧可能会在最后一个接收的帧之前传输到目标块。

旧消息帧是中断时在 CPU 和伙伴之间的在途帧，或是已经由 SFB 收到的消息帧。

---

## 6.9.3 用 RK 512 计算机连接传输数据

### 6.9.3.1 使用 RK 512 计算机连接传输数据 — 基本信息

#### 引言

RK 512 计算机连接控制 CPU 和通讯伙伴之间的 PtP 数据交换。

与 3964(R) 相反，RK 512 计算机连接提供较高数据完整性和高级的寻址选项。

#### 响应消息帧

RK 512 计算机连接响应每个正确接收的指令帧，并向 CPU 发送响应消息帧。这允许发送站验证数据是否被 CPU 接收且无错，或验证它需要的数据是否可以在 CPU 中获得。

#### 指令帧

指令帧是 SEND 消息帧或 FETCH 消息帧之一。

#### SEND 消息帧

在传输 SEND 消息帧时，CPU 传输一个包含用户数据的指令帧。通讯伙伴用不包含用户数据的响应消息帧响应。

#### FETCH 消息帧

在传输 FETCH 消息帧时，CPU 传输一个不包含用户数据的指令帧。通讯伙伴用包含用户数据的响应消息帧响应。

#### 连续的消息帧

对于 SEND 和 FETCH 消息帧，如果数据长度超过 128 字节，便传输连续的消息帧。

## 消息帧头

对于 RK 512 计算机连接，每个消息帧都用一个头初始化。它包含消息帧 ID，有关数据源和目标的信息以及错误编号。

下表说明了指令帧头的结构：

字节	说明
1	指令帧中的消息帧 ID(00H)， 用于连续的指令帧 (FFH)
2	消息帧 ID (00H)
3	<ul style="list-style-type: none"> <li>‘A’ (41H)：带目标 DB 的 SEND 请求</li> <li>‘O’ (4FH)：带目标 DX 的 SEND 请求</li> <li>‘E’ (45H)：FETCH 请求</li> </ul>
4	要传送的数据的获取来源（仅“D”可用于发送）： <ul style="list-style-type: none"> <li>‘D’ (44H)：数据块“X” (58H) = 扩展数据块</li> <li>‘E’ (45H)：输入字节“A” (41H) = 输出字节</li> <li>‘M’ (4DH)：存储器字节“T” (54H) = 时间单元</li> <li>‘Z’ (5AH)：计数器单元</li> </ul>
5	带 SEND 请求的数据目标或带 FETCH 请求的数据源，例如，字节 5 = DB 号，字节 6 = DW 号。 <sup>1)</sup>
6	
7	高位字节长度：根据类型，数据长度以字节或字格式传输
8	低位字节长度：根据类型，数据长度以字节或字格式传输
9	处理器间通讯标志字节号；如果尚未指定处理器间通讯标志，在此输入 FFh 值。
10	<ul style="list-style-type: none"> <li>位 0 到 3：如果尚未指定处理器间通讯标志，在此输入 FH 值。</li> <li>位 4 到 7：CPU 号（1 到 4）；如果未指定 CPU 号（编号 0），但却指定了处理器间通讯标志，则在此输入 0H；如果未指定 CPU 号或处理器间通讯标志，则在此输入 FH。</li> </ul>
<sup>1)</sup> RK 512 寻址说明了带字限制的数据源和目标。在 SIMATIC S7 中自动转换为字节地址。	

字节 3 和 4 包含 ASCII 字符。

连续的指令帧头仅由字节 1 到 4 组成。



## 响应消息帧的结构和内容

传输指令帧后，RK 512 在监视时间内等待通讯伙伴的响应消息帧。监视时间的长度为 20 s。

响应消息帧由 4 字节组成，包含请求处理信息：

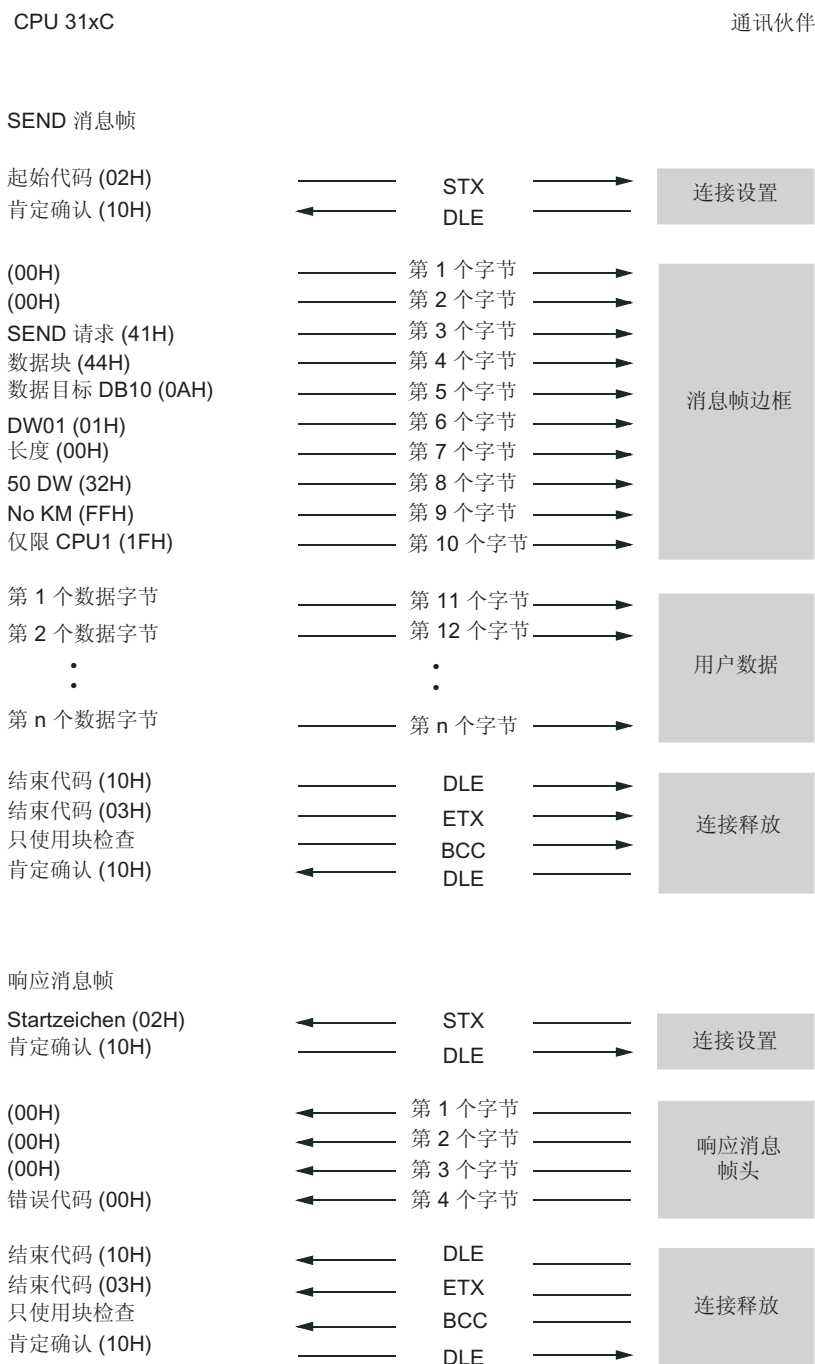
下表提供响应消息帧的结构和内容。

字节	说明
1	响应消息帧的消息帧 ID(00H), 用于连续的响应消息帧 (FFH)
2	消息帧 ID (00H)
3	分配的值 00H
4	响应消息帧中通讯伙伴的错误编号：* <ul style="list-style-type: none"><li>• 如果传输过程中无错，则为 00H</li><li>• &gt; 00H 错误编号</li></ul>
* 如果响应消息帧中有错误编号，则在 SFB 的 STATUS 输出中自动生成一个事件 ID。	

### 6.9.3.2 使用 RK 512 发送数据

#### 步骤

下图说明了使用 RK 512 计算机连接发送响应消息帧时数据传输的顺序：



## 发送数据

SEND 请求按下面顺序执行：

- **主动伙伴**  
传输 SEND 消息帧。包含消息帧头和数据。
- **被动伙伴**  
接收消息帧，验证头和数据，并且在数据写入目标块后用响应消息帧进行应答。
- **主动伙伴**  
接收响应消息帧。  
如果用户数据长度超过 128 字节，将发送连续的 SEND 消息帧。
- **被动伙伴**  
接收连续的 SEND 消息帧，验证头和数据，并且在数据写入目标块后用连续的响应消息帧进行应答。

---

### 说明

如果 CPU 没有在未出错的情况下收到 SEND 消息帧，或在消息帧头中出现错误，则通讯伙伴在响应消息帧的字节 4 中输入一个错误编号。在协议出错的情况下，在响应消息帧中不会有条目。

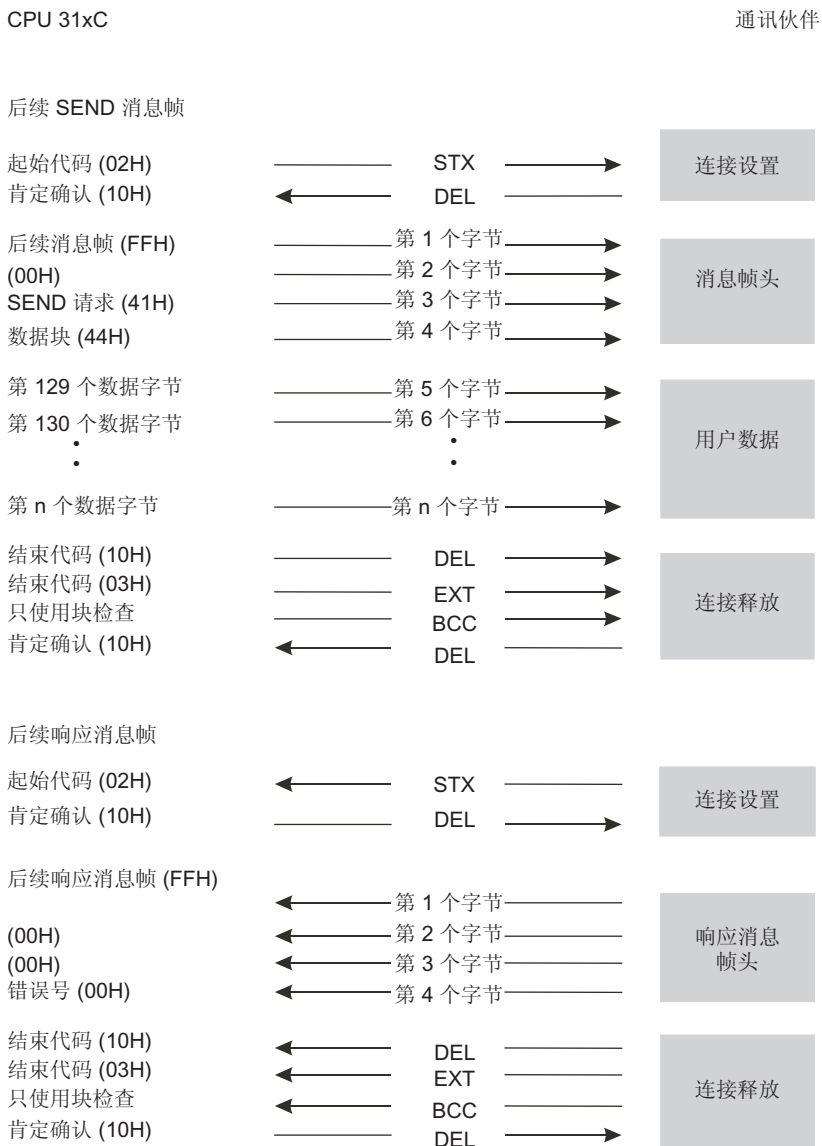
---

### 连续的 SEND 消息帧

如果用户数据长度超过 128 字节，将启动连续的 SEND 消息帧。顺序对应 SEND 消息帧的顺序。

超出 128 字节的发送字节将自动分成一个或多个消息帧传输。

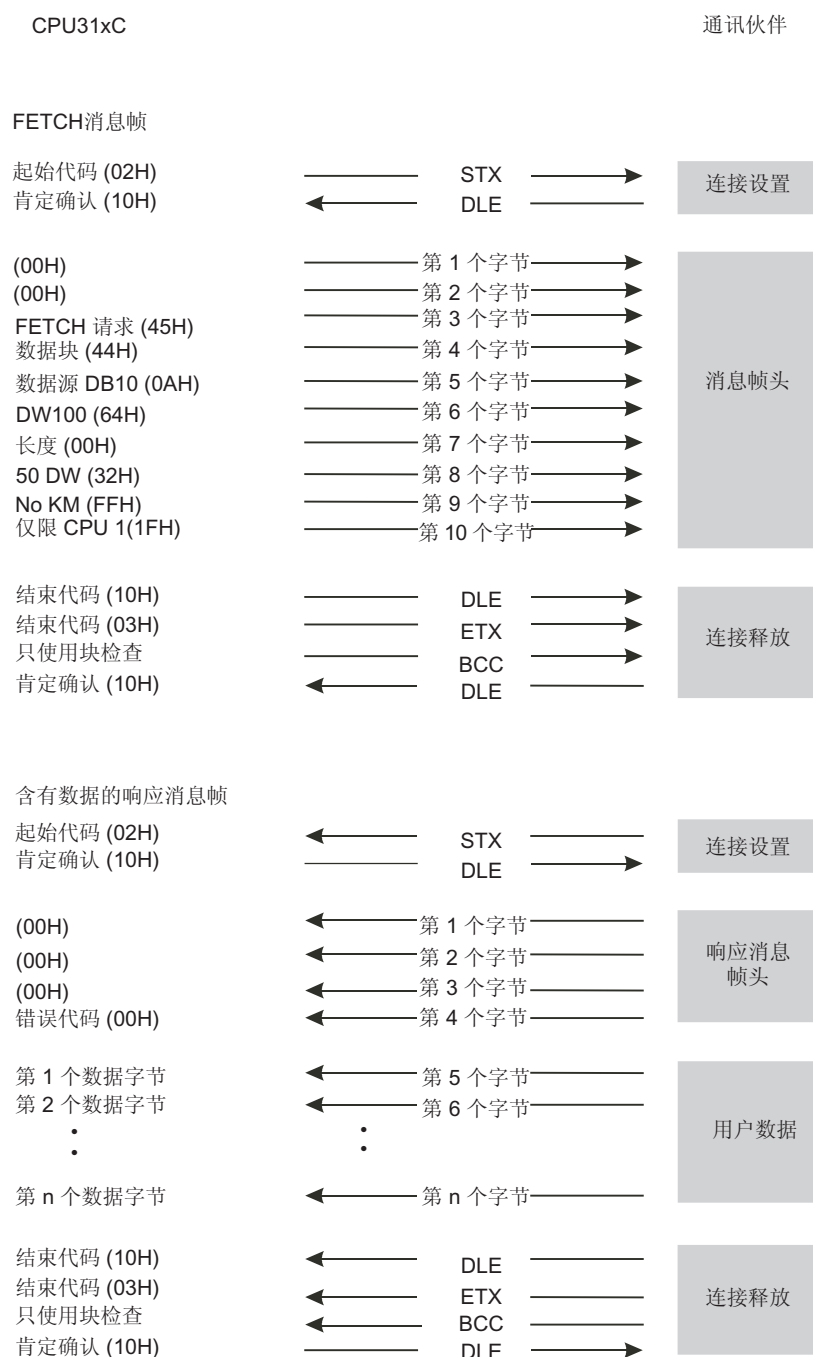
下图说明了使用连续的响应消息帧发送连续的 SEND 消息帧时数据传输的顺序：



### 6.9.3.3 用 RK 512 获取数据

#### 步骤

下图说明了通过具有 RK 512 计算机连接的响应消息帧获取数据时的数据传输顺序。



## 获取数据

FETCH 请求按下面的顺序执行：

- 1. 主动伙伴：**  
传输 SEND 消息帧。其中包含消息头。
- 2. 被动伙伴：**  
接收消息帧，验证头，从 CPU 获取数据并用响应消息帧应答。此帧包含数据。
- 3. 主动伙伴：**  
接收响应消息帧。
- 4. 如果用户数据长度超过 128 字节，它将传输连续的 FETCH 消息帧。此帧包含头字节 1 到 4。**
- 5. 被动伙伴：**  
接收连续的 FETCH 消息帧，验证头，从 CPU 获取数据，并用包含更多数据的连续响应消息帧进行应答。

如果在第 4 字节中含有错误编号（不为 0），则响应消息帧不包含任何数据。

如果请求的字节数超过 128，则会在一个（或多个）连续的消息帧中自动获取额外的字节。

---

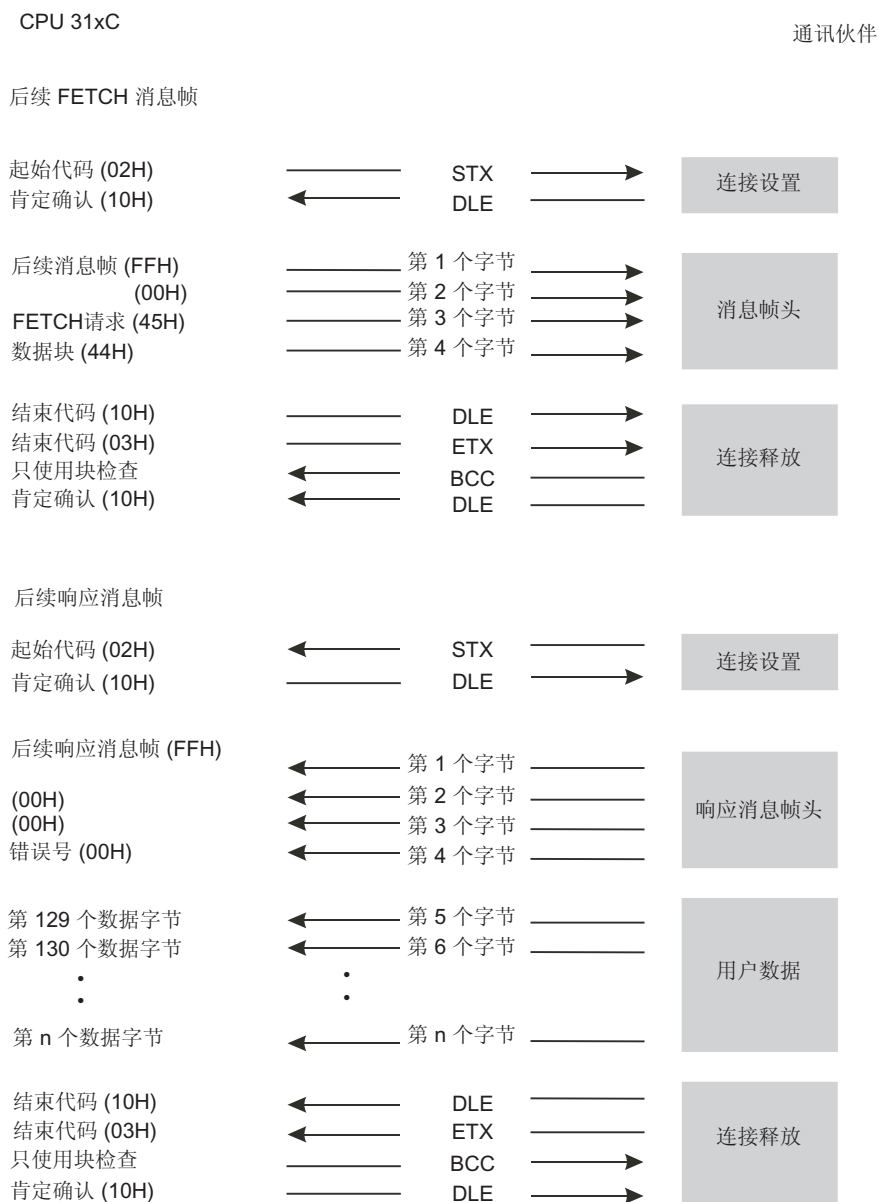
### 说明

如果 CPU 没有在未出错的情况下收到 FETCH 消息帧，或在消息帧头中出现错误，则通讯伙伴在响应消息帧的字节 4 中输入一个错误编号。在协议出错的情况下，在响应消息帧中不会有条目。

---

## 连续的 FETCH 消息帧

下图说明了通过连续响应消息帧获取数据时数据传输的顺序。



### 伪全双工模式

伪全双工表示：通讯伙伴可随时发送指令和响应消息帧，即使其它通讯伙伴当前也正在发送。指令和响应消息帧的最大嵌套深度是“1”。因此，在用响应消息帧应答上一个帧之前，不能处理其它指令帧。

在某些情况下，如果同时有两个通讯伙伴请求发送，则可在传输响应消息帧之前传输 SEND 消息帧。

在下图中，在通讯伙伴传输 SEND 消息帧之前，没有发送响应第一个 SEND 消息帧的连续响应消息帧：

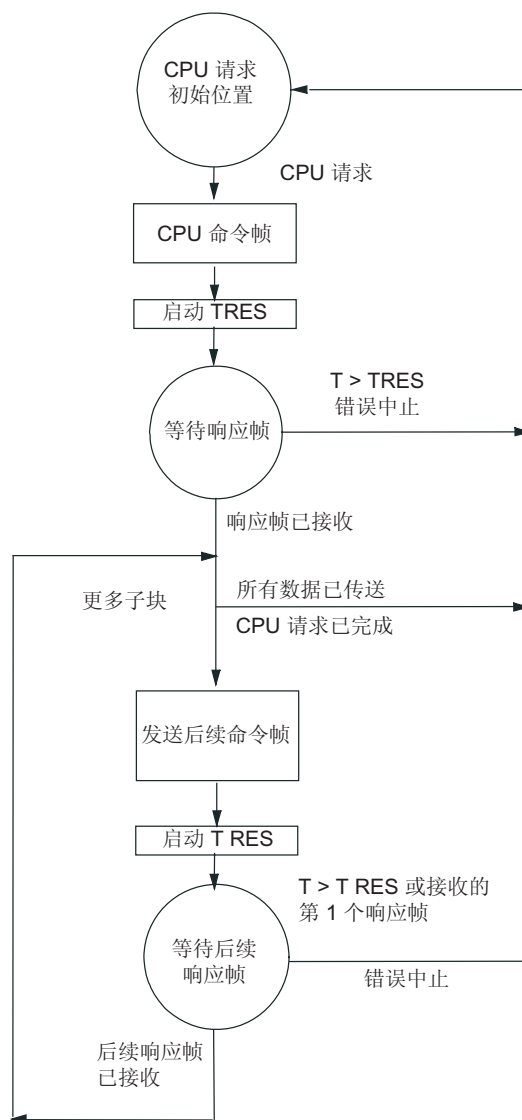




### 6.9.3.4 RK 512 处理请求的顺序

#### RK 512 CPU 请求

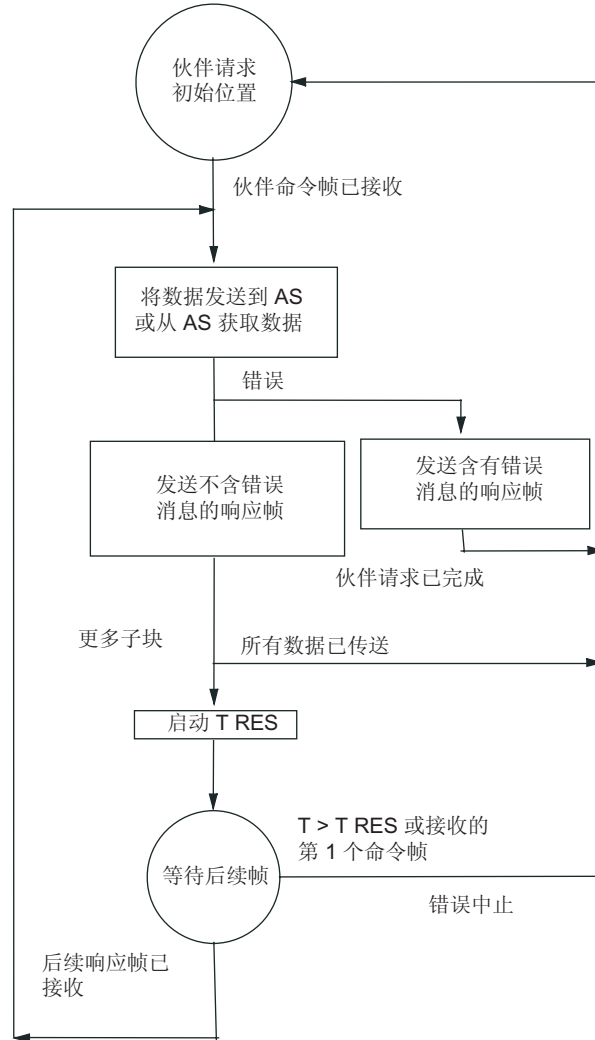
在下图中可看到带 CPU 请求的 RK 512 计算机连接的顺序：



响应帧监控时间  
 $T_{RES} = 10$  秒

### RK 512 伙伴请求

在下图中可看到由伙伴请求初始化的 RK 512 程序的顺序：



响应帧监控时间  
T RES = 10 秒  
AS = 自动化系统

## 6.10 规范

### 6.10.1 常规规范

#### 概述

下表提供了常规规范。

关于与 SIMATIC S7 300 相关的附加技术规范，请参见参考手册 *S7 300 自动化系统，模块数据* 和安装手册 *S7 300 自动化系统，装配*：

- EMC（电磁兼容性）
- 运输与储存条件
- 机械和环境气候条件
- 关于绝缘测试、安全等级以及防护等级的信息
- 认证

规范	
可用的协议驱动程序	ASCII 驱动程序 3964(R) 程序 RK 512
3964(R) 程序和 RK 512 的传输速度	300、600、1200、2400、4800、9600、19200、38400 波特
ASCII 驱动程序的传输速度	300、600、1200、2400、4800、9600、19200、38400（半双工）
字符帧	<ul style="list-style-type: none"> <li>• 每个字符的位（7 或 8）。RK 512 仅 8 个字符。</li> <li>• 启动/停止位（1 或 2）。</li> <li>• 奇偶校验（无、奇校验、偶校验）；对于 7 位/字符，只允许“奇校验”或“偶校验”组态。</li> </ul>

### X27 (RS 422/485) 接口的规范

下表说明了 X27 (RS 422/ 485) 接口 RS 422/485 的规范：

规范	
接口	RS 422 或 RS 485，15 针 D 型子插座
RS 422 信号	TXD (A)、RXD (A)、TXD (B)、RXD (B)、GND R/T (A)、R/T (B)、GND
RS 485 信号	与 S7 内部电源（背板总线）和辅助 24V DC 电源完全隔离
最大传输距离	1200 m
最大传输率	38400 kbps

### 6.10.2 ASCII 驱动程序的规范

#### 概述

下表说明了 ASCII 驱动程序的规范：

<b>ASCII 驱动程序</b>	
最大消息帧长度	1024 字节
参数	可组态： <ul style="list-style-type: none"> <li>• 传输率：300、600、1200、2400、4800、9600、19200 波特，38400 波特（半双工模式）</li> <li>• 字符帧：10、11 或 12 位</li> <li>• 字符延迟时间：1 ms 到 65535 ms（1 ms 步长）</li> <li>• 流控制：无，XON/XOFF</li> <li>• XON/XOFF 字符（仅“流控制” = “XON/XOFF”）</li> <li>• 在 XOFF 后等待 XON：20 ms 到 65530 ms（10 ms 步长）</li> <li>• 要缓冲的消息帧数：1 到 10，使用整个缓冲区</li> <li>• 防止覆盖：是/否</li> <li>• 收到的消息帧的消息结束识别：                             <ul style="list-style-type: none"> <li>- 在字符延迟时间结束时</li> <li>- 在收到文本结束字符时</li> <li>- 在收到固定数目的字符时</li> </ul> </li> </ul>
<b>超出字符延迟时间时的 ASCII 驱动程序消息结束识别</b>	
参数	不需要更多参数分配。在超出声明的字符延迟时间时识别为消息结束。
<b>通过可组态的文本结束字符识别消息结束的 ASCII 驱动程序</b>	
参数	也可组态： <ul style="list-style-type: none"> <li>• 文本结束字符数：1, 2</li> <li>• 第一/第二个结束代码的十六进制代码</li> <li>• BCC 字符数：1, 2</li> </ul>
<b>通过组态的字符长度识别消息结束的 ASCII 驱动程序</b>	
参数	也可组态： <ul style="list-style-type: none"> <li>• 字符长度：1 至 1024 字节</li> </ul>

### 6.10.3 3964(R) 程序的规范

#### 概述

下表说明了 3964(R) 程序的规范。

带缺省值的 3964(R) 程序	
最大消息帧长度	1024 字节
参数	可组态： <ul style="list-style-type: none"> <li>• 带/不带块检查字符</li> <li>• 优先级：低/高</li> <li>• 传输率：300、600、1200、2400、4800、9600、19200、38400 波特</li> <li>• 字符帧：10、11 或 12 位</li> <li>• 缺省接收线路：无、R(A) 5V/R(B) 0V、R(A) 0V/R(B) 5V</li> <li>• 要缓冲的消息帧数：1 到 10，使用整个缓冲区</li> </ul>

可用 3964(R) 程序编程	
最大消息帧长度	1024 字节
参数	可组态： <ul style="list-style-type: none"> <li>• 带/不带块检查字符</li> <li>• 优先级：低/高</li> <li>• 传输率：300、600、1200、2400、4800、9600、19200、38400 波特</li> <li>• 字符帧：10、11 或 12 位</li> <li>• 字符延迟时间：20 ms 到 65530 ms（10 ms 步长）</li> <li>• 应答延迟时间：20 ms 到 65530 ms（以 10 ms 为增量）</li> <li>• 连接尝试次数：1 到 255</li> <li>• 传输尝试次数：1 到 255</li> <li>• 缺省接收线路：无、R(A) 5V/R(B) 0V、R(A) 0V/R(B) 5V</li> </ul>

### 6.10.4 RK 512 计算机连接的规范

#### 概述

下表说明了 RK 512 计算机连接的规范：

RK 512 计算机连接	
最大消息帧长度	1024 字节
参数	可组态： <ul style="list-style-type: none"> <li>• 传输率：300、600、1200、2400、4800、9600、19200、38400 波特</li> <li>• 字符帧：10、11 或 12 位</li> <li>• 字符延迟时间：20 ms 到 65530 ms（以 10 ms 为增量）</li> <li>• 确认延迟时间：20 ms 到 65530 ms（以 10 ms 为增量）</li> <li>• 尝试连接的次数：1 到 255</li> <li>• 尝试传输的次数：1 到 255</li> <li>• 缺省接收线路：无、R(A) 5V/R(B) 0V、R(A) 0V/R(B) 5V</li> </ul>

### 6.10.5 最小 CPU 周期数

#### 概述

下表说明处理请求需要的最小 CPU 周期（SFB 调用）数：

块	名称	进行处理所需的 CPU 周期数...		
		完成且无错	完成且有错	RESET/RESTART
SFB 60	SEND_PTP	≥ 2	≥ 2	≥ 3
SFB 61	RCV_PTP	≥ 2	≥ 2	≥ 3
SFB 62	RES_RCVB	≥ 2	≥ 2	≥ 3
SFB 63	SEND_RK	≥ 2	≥ 2	≥ 3
SFB 64	FETCH_RK	≥ 2	≥ 2	≥ 3
SFB 65	SERVE_RK	≥ 2	≥ 2	≥ 3

## 6.10.6 传输时间

### 概述

下表包含测量的传输时间（取决于选择的通讯协议）。

进行此测量时，互联了两个 CPU 314C-2PtP。测量的是通讯链接上从第一个消息帧的第一个字符出现到下一个连续消息帧的第一个字符出现这段时间。

使用 ASCII 驱动程序时，测量基于最快的协议版本（带一个文本结束字符的消息结束识别并且无软件流控制）。

使用 3964(R) 和 RK 512 计算机连接时，以相应的缺省设置（即带 BCC 的缺省值）执行测量。

### ASCII 驱动程序（传输时间以毫秒计）

用户数据的传输率 (Bd)	38400	19200	9600	4800	2400	1200	600	300
1 字节	5	6	7	9	13	23	41	78
10 字节	7	11	17	28	51	97	190	376
20 字节	11	17	28	51	97	190	374	744
50 字节	19	34	62	120	236	465	927	1847
100 字节	35	64	121	236	466	926	1846	3685
200 字节	64	120	237	467	927	1845	3686	7363
500 字节	154	298	586	1160	2309	4607	9204	13398
1000 字节	305	591	1168	2316	4613	9210	18402	36788

### 3964(R) 程序（传输时间以毫秒计）

用户数据的传输率 (Bd)	38400	19200	9600	4800	2400	1200	600	300
1 字节	8	11	14	22	38	71	137	267
10 字节	11	16	25	43	80	154	302	601
20 字节	14	22	36	66	126	246	487	966
50 字节	23	38	71	136	264	522	1037	2071
100 字节	38	68	130	250	494	982	1958	3907
200 字节	67	126	246	482	956	1902	3798	7586
500 字节	158	303	595	1175	2838	4664	9316	18620
1000 字节	308	597	1177	2330	4642	9266	18515	37011


## RK 512 计算机连接（传输时间以毫秒计）

用户数据的传输率 (Bd)	38400	19200	9600	4800	2400	1200	600	300
1 字节	21	29	44	75	134	253	501	1002
10 字节	33	42	63	101	180	337	667	1334
20 字节	37	48	74	124	228	430	851	1701
50 字节	48	71	112	199	368	709	1402	2804
100 字节	70	105	178	321	605	1176	2323	4642
200 字节	126	196	336	618	1173	2293	4543	9064
500 字节	278	445	778	1450	2784	5450	10836	21608
1000 字节	545	878	1554	2876	5534	10860	21571	43027

## 6.10.7 电缆

## 引言

在自己制作电缆时，请注意一定要使用带屏蔽的连接器的外壳。电缆屏蔽两端必须有大部分的表面积接触到连接器的外壳和屏蔽触点。

 <b>小心</b> 切勿将电缆屏蔽接地。否则，接口将会损毁。 <b>GND（针 8）必须始终连接到两端。</b>
--

否则，接口可能会损毁。下面几页说明了 CPU 与 S7 模块或 SIMATIC S5 之间点对点连接的一些电缆实例。



### 电缆 X 27/RS422 (CPU 31xC-CPU 31xC/CP 340/CP 341/CP 440/CP 441)

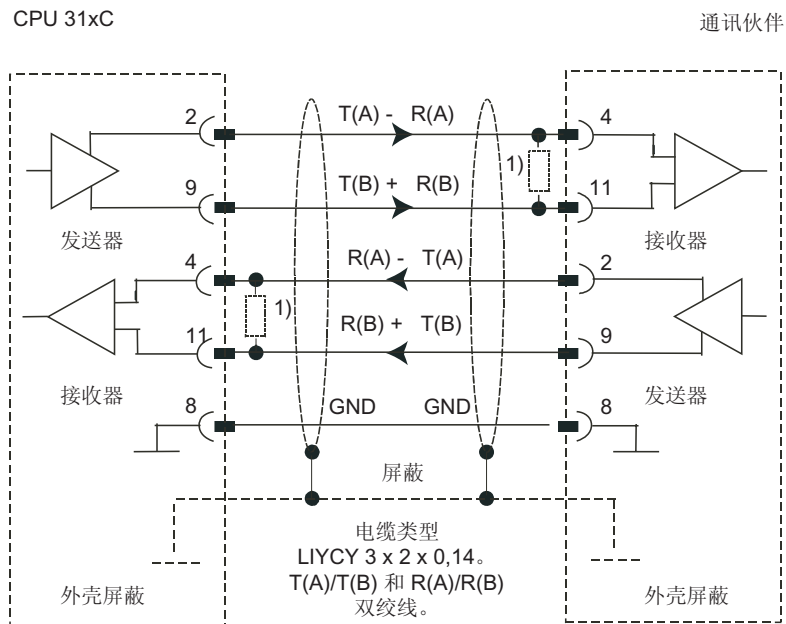
可用的电缆缺省长度：5 m、10 m 和 50 m。

类型	订货号
X27 (RS 422), 5 m	6ES7 9023AB000AA0
X27 (RS 422), 10 m	6ES7 9023AC000AA0
X27 (RS 422), 50 m	6ES7 902-3AG000AA0

下图说明了 CPU 31xC 和 CPU 31xC/CP 340/CP341/CP 440/CP 441 之间用于 RS422 操作的电缆。

对于连接电缆，需要以下插入连接器：

- CPU 31xC 端：带螺丝互锁的 15 针 D 型子插入连接器
- 通讯伙伴端：带螺丝互锁的 15 针 D 型子插入连接器



1) 为确保电缆长度 > 50 米时数据交换无干扰，必须在接收端焊接一个大约 330Ω 的终端电阻。

#### 说明

对于使用的各类电缆，允许长度如下：

- 19200 波特时最长 1200 m
- 38400 波特时最长 500 m



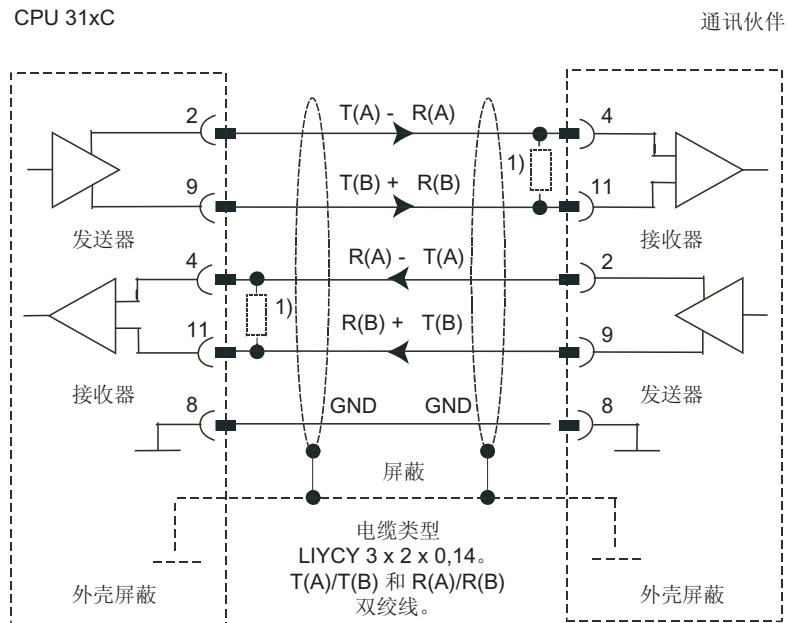
### 电缆 X27/RS422 (CPU 31xC-CP 544, CP 524, CPU 928B, CPU 945, CPU 948)

下图说明了 CPU 31xC 和 CP 544、CP 524、CPU 928B、CPU945、CPU 948 之间用于 RS422 操作的电缆。

Siemens 不提供现成的电缆。

对于连接电缆，需要以下插入连接器：

- 在 CPU 31xC 端：带螺丝互锁的 15 针 D 型子插入连接器
- 通讯伙伴端：带滑动互锁的 15 针 D 型子插入连接器



1) 为确保电缆长度 > 50 m 时数据交换无干扰，必须在接收器端焊接一个大约 330Ω 的终端电阻。

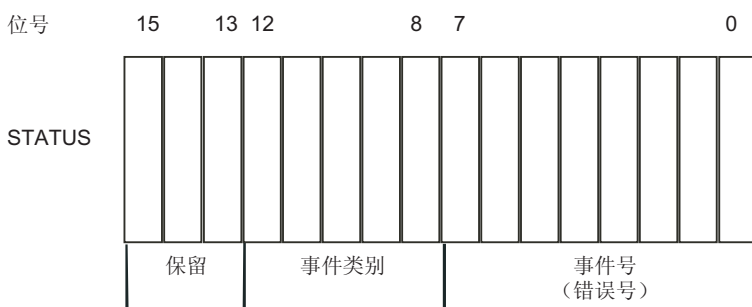
### 6.10.8 错误消息

#### 基本信息

每个 SFB 都分配有一个用于错误诊断的 STATUS 参数。STATUS 消息 ID 始终具有相同含义，而与所用 SFB 无关。

#### 事件类别/事件编号的编号方案

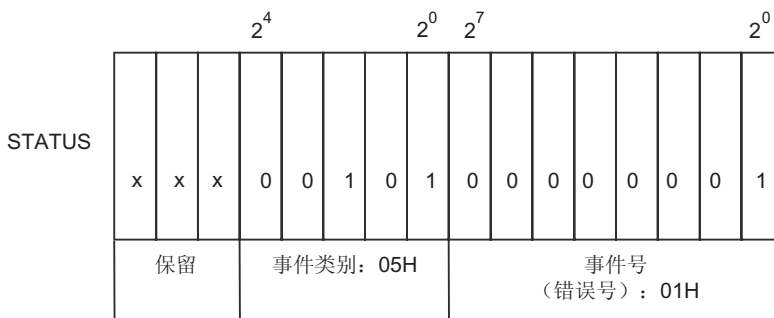
下图说明了 STATUS 参数的结构：



#### 实例

下图说明了事件“请求因重新启动或复位而取消”的 STATUS 参数的内容（事件类别：05H，事件编号 = 01H）：

在因暖启动或复位而中止作业的情况下



## 事件类别

下表提供了对各种事件类别和 ID 的说明：

事件类别 3 (03H): “SFB 参数分配出错”		
事件编号	事件	补救措施
(03)01H	源/目标数据类型非法或缺失。 范围（起始地址，长度）非法。 DB 不允许或不存在（例如，DB 0）或 其它源/目标数据类型非法或缺失。 处理器间通讯的字节号或位号无效。	进行检查并在需要时更正参数。 在消息帧头中，伙伴提供了非法参数。 检查参数并在需要时创建块。 参见请求表以获得有效数据类型。 在消息帧头中，伙伴提供了错误参数。
(03)03H	访问区域被拒绝。	检查参数。请参见请求表，以获得允许的起始地址和长度。否则，伙伴会在消息帧头中提供错误参数。

事件类别 5 (05H): “请求处理错误”		
事件编号	事件	补救措施
(05)01H	当前请求因冷重新启动或复位而被取消。	重复已取消的请求。通过编程设备重新分配接口参数时，应确保在开始写入操作前，没有其它活动的请求。
(05)02H	在此操作状态（例如，没有组态设备接口）下不允许请求。	组态设备接口。
(05)0EH	<ul style="list-style-type: none"> <li>消息帧长度无效</li> <li>或</li> <li>在最大允许长度内没有出现声明的结束代码。</li> </ul>	<ul style="list-style-type: none"> <li>消息帧长度 &gt; 1024 字节。选择较短的消息帧长度。</li> <li>或</li> <li>在发送缓冲区的适当位置添加结束代码。</li> </ul>
(05)13H	数据类型错误 (DB ...): <ul style="list-style-type: none"> <li>数据类型未知或非法（例如，DE）</li> <li>在 SFB 中指定了不匹配的源和目标数据类型。</li> </ul>	请参见请求表，以获得允许的数据类型及其组合。
(05)15H	为协调标志指定的位号不正确。	允许的位号为：0 到 7
(05)16H	指定的 CPU 号太大。	允许的 CPU 号为：0、1、2、3 或 4
(05)17H	传输的数据过长，长度 > 1024 字节	将请求分成较短的多个请求。
(05)1DH	发送/接收请求被中止，原因为 <ul style="list-style-type: none"> <li>通讯块复位</li> <li>新的参数分配</li> </ul>	重复调用通讯块。
(05)22H	前一请求尚未完成就启动了新的 SEND 请求。	在前一请求以 DONE 或 ERROR 关闭前，不要启动新的 SEND 请求。

事件类别 6 (06H): “处理伙伴请求时出错” (仅对于 RK 512)		
事件编号	事件	补救措施
(06)01H	在第一个命令字节中出错 (非 00 或 FFH)	伙伴上的基本头结构错误。可以使用数据链接中互联的接口测试设备检查伙伴设备的故障。
(06)02H	第三个命令字节出错 (非 A、0 或 E)	伙伴上的基本头结构错误。可以使用数据链接中互联的接口测试设备检查伙伴设备的故障。
(06)03H	在有连续消息帧的情况下, 第三个命令字节出错 (与第一个消息帧的命令不同)	伙伴上的基本头结构错误。可以使用数据链接中互联的接口测试设备检查伙伴设备的故障。
(06)04H	第四个命令字节出错 (命令字母不正确)	伙伴上的基本头结构错误, 或请求了非法命令组合。检查允许的命令。可以使用数据链接中互联的接口测试设备检查伙伴设备的故障。
(06)06H	第五个命令字节出错 (DB 号不允许)	请参见请求表, 以获得允许的 DB 号、起始地址或长度。
(06)07H	第五个或第六个命令字节出错 (起始地址过高)	请参考请求表, 以获得允许的 DB 号、起始地址或长度。
(06)09H	第 9 个和第 10 个命令字节出错 (此数据类型的协调标志非法或位号过大)。	伙伴上的基本头结构错误。在请求表中查找何时允许使用协调标志。
(06)0AH	第十个命令字节出错 (CPU 号非法)	伙伴上的基本头结构错误。

事件类别 7 (07H): “发送错误”		
事件编号	事件	补救措施
(07)01H	仅 3964(R): 发送首个重复: <ul style="list-style-type: none"> <li>传输消息帧时检测到错误, 或</li> <li>伙伴使用否定应答代码 (NAK) 请求一个重复。</li> </ul>	重复不是错误, 但它可以指示传输线路有干扰或伙伴设备有故障。如果达到最大重复次数后仍未传输消息帧, 则输出一个错误编号, 说明第一个出现的错误。
(07)02H	仅对于 3964(R): 连接错误: 在发送 STX 后, 收到了 NAK 或其它代码 (DLE 或 STX 除外)。	可以使用切换到传输线路的接口测试设备检查伙伴设备的故障。
(07)03H	仅对于 3964(R): <ul style="list-style-type: none"> <li>超出应答延迟时间:</li> <li>发送 STX 后, 伙伴在应答延迟时间内没有响应。</li> </ul>	伙伴设备太慢或没有准备好接收, 或者出现例如发送线路断开的情况。可以使用数据链接中互联的接口测试设备检查伙伴设备的故障。
(07)04H	仅对于 3964(R): 在伙伴端取消: 在当前的发送操作中, 从伙伴处收到了一个或多个字符。	检查伙伴是否也指示可能因未收到所有的发送数据 (例如, 发送线路断开), 或致命错误未决, 或者伙伴设备故障而导致的错误。如果需要, 可使用数据链接中互联的接口测试设备对此进行验证。
(07)05H	仅对于 3964(R): 发送时否定应答	检查伙伴是否也指示可能因未收到所有的发送数据 (例如, 发送线路断开), 或致命错误未决, 或者伙伴设备故障而导致的错误。如果需要, 可使用数据链接中互联的接口测试设备对此进行验证。

事件类别 7 (07H): “发送错误”		
事件编号	事件	补救措施
(07)06H	仅对于 3964(R): 连接结束错误: <ul style="list-style-type: none"> <li>• 伙伴在连接结束时以 NAK 或一个随机字符串 (DLE 除外) 拒绝了消息帧, 或</li> <li>• 过早收到应答字符 (DLE)。</li> </ul>	检查伙伴是否也指示可能因未收到所有的发送数据 (例如, 发送线路断开), 或致命错误未决, 或者伙伴设备故障而导致的错误。如果需要, 可使用数据链接中互联的接口测试设备对此进行验证。
(07)07H	仅对于 3964(R): 连接结束时超出确认延迟, 或发送消息帧之后超出回复监视时间: DLE ETX 断开连接后, 在应答延迟时间内没有收到伙伴的响应。	伙伴设备有故障或太慢。如果需要, 可使用数据链接中互联的接口测试设备对此进行验证。
(07)08H	仅 ASCII 驱动程序: 已超过 XON 的等待时间。	通讯伙伴有故障、太慢或已离线关闭。检查通讯伙伴或在需要时更改参数分配。
(07)09H	仅对于 3964(R): 不能连接。超出允许的连接请求数。	检查接口电缆或传输参数。 还要检查在伙伴设备上是否正确组态了 CPU 和 CP 之间的接收功能。
(07)0AH	仅对于 3964(R): 不能传输数据。超出允许的传输尝试次数。	检查接口电缆或传输参数。
(07)0BH	仅对于 3964(R): 由于两个伙伴均为高优先级, 故无法解决初始化冲突。	更改参数分配。
(07)0CH	仅对于 3964(R): 由于两个伙伴均为低优先级, 故无法解决初始化冲突。	更改参数分配。

事件类别 8 (08H): “接收错误”		
事件编号	事件	补救措施
(08)01H	仅 3964®: 期待首个重复: 收到消息帧时检测到错误。CPU 通过否定应答 (NAK) 请求伙伴重复。	重复不是错误, 但它可以指示传输线路有干扰或伙伴设备有故障。如果达到最大重复次数后仍未传输消息帧, 则输出一个错误编号, 说明第一个出现的错误。
(08)02H	仅对于 3964®: 连接错误: <ul style="list-style-type: none"> <li>在空闲模式下, 收到了一个或多个随机代码 (NAK 或 STX 除外), 或</li> <li>收到 STX 后, 伙伴不等待响应 DLE 就发送了更多代码。</li> </ul> 伙伴通电后: <ul style="list-style-type: none"> <li>伙伴接通电源时, CPU 收到一个未定义的字符。</li> </ul>	可以使用数据链接中互联的接口测试设备确定伙伴设备的故障。
(08)05H	仅对于 3964®: 接收时出现逻辑错误: 收到 DLE 后, 又收到一个随机代码 (DLE 或 ETX 以外)。	检查伙伴是否总是复制消息帧头中的 DLE 和数据字符串, 或连接是否通过 DLE ETX 断开。可以使用数据链接中互联的接口测试设备确定伙伴设备的故障。
(08)06H	超过字符延迟时间: <ul style="list-style-type: none"> <li>在字符延迟时间内未收到两个连续字符, 或</li> </ul> 仅对于 3964®: <ul style="list-style-type: none"> <li>在字符延迟时间内, 未收到在建立连接时发送 DLE 后的首个字符。</li> </ul>	伙伴设备有故障或太慢。如果需要, 可使用数据链接中互联的接口测试设备对此进行验证。
(08)07H	消息帧长度非法: 收到一个长度为零的消息帧。	收到长度为零的消息帧并非错误。 检查通讯伙伴为何发送不含用户数据的消息帧。
(08)08H	仅对于 3964®: 块检查字符 (BCC) 中出错: 内部计算的 BCC 值与伙伴在连接结束时收到的 BCC 不匹配。	检查连接是否被严重破坏; 此时也可查看错误代码。可以使用数据链接中互联的接口测试设备确定伙伴设备的故障。
(08)09H	仅对于 3964®: 已超过块重复的等待时间。	在通讯伙伴方声明一个与您的模块相同的块等待时间。可以使用数据链接中互联的接口测试设备检查通讯伙伴的故障。
(08)0AH	没有可用的空闲接收缓冲区: 没有可用于接收数据的缓冲区空间。	必须更频繁地调用 SFB RCV。
(08)0CH	传输错误: <ul style="list-style-type: none"> <li>检测到传输错误 (奇偶校验/停止位/溢出错误)。</li> </ul> 仅对于 3964®: <ul style="list-style-type: none"> <li>如果在空闲模式下收到一个损坏的字符, 将立即报告错误以便及早检测到传输线路上的干扰。</li> </ul> 仅对于 3964®: <ul style="list-style-type: none"> <li>如果在发送或接收操作过程中发生此错误, 则启动重复。</li> </ul>	传输线路上的干扰造成消息帧重复, 因此降低了用户数据的吞吐量。漏检错误的风险增加。更改系统设置或电缆接线。 检查通讯伙伴的连接电缆, 或检查双方设备的波特率、奇偶校验和停止位的数目设置是否相同。



事件类别 8 (08H): “接收错误”		
事件编号	事件	补救措施
(08)0DH	BREAK: 连接到伙伴的接收线路中断。	重新连接或接通伙伴电源。
(08)0EH	因禁用流控制, 接收缓冲区溢出	在用户程序中必须更频繁地调用接收 SFB, 或必须将通讯组态为允许流控制。
(08)10H	奇偶校验错误	检查通讯伙伴的连接电缆, 或检查双方设备的波特率、奇偶校验和停止位的数目设置是否相同。
(08)11H	字符帧错误	检查通讯伙伴的连接电缆, 或检查双方设备的波特率、奇偶校验和停止位的数目设置是否相同。 更改系统设置或电缆接线。
(08)12H	仅对于 ASCII 驱动程序: 在收到 CPU 发送的 XOFF 后收到更多的字符。	重新组态通讯伙伴或更快速地处理数据。
(08)14H	仅对于 ASCII 驱动程序: 由于未使用流控制, 导致一个或多个消息帧丢失。	尽可能多地使用流控制。使用整个接收缓冲区。在基本参数中, 将“对 CPU STOP 模式的响应”参数设置为“连续”。
(08)16H	接收消息帧的长度超过指定的最大长度。	需要在伙伴方更正。

事件类别 9 (09H): “从互连伙伴接收响应消息帧出现错误或错误消息帧”		
事件编号	事件	补救措施
(09)02H	仅 RK 512: 伙伴方的存储器访问错误 (存储器不存在) 伙伴使用 SIMATIC S5: <ul style="list-style-type: none"> <li>• 状态字区域不正确, 或</li> <li>• 数据区不存在 (DB/DX 除外), 或</li> <li>• 数据区太短 (DB/DX 除外)</li> </ul>	检查伙伴是否具有所需的数据区以及该区域是否足够大, 或检查已调用系统功能块的参数。 检查在系统功能块中指定的长度。
(09)03H	仅对于 RK 512: 伙伴处的 DB/DX 访问错误 (DB/DX 不存在或太短) 使用 SIMATIC S5 作为伙伴: <ul style="list-style-type: none"> <li>• DB/DX 不存在, 或</li> <li>• DB/DX 太短, 或</li> <li>• DB/DX 号非法。</li> </ul> FETCH 请求超出允许的源范围。	检查伙伴是否具有所需的数据区 <ul style="list-style-type: none"> <li>• 以及该区域是否足够大。</li> <li>• 或检查调用的 SFB 的参数。</li> <li>• 检查在系统功能块中指定的长度。</li> </ul>

事件类别 9 (09H): “从互连伙伴接收响应消息帧出现错误或错误消息帧”		
事件编号	事件	补救措施
(09)04H	仅对于 RK 512: 伙伴报告“请求类型非法”。	伙伴行为错误, 因为 CPU 从不输出系统命令。
(09)05H	仅对于 RK 512: 伙伴错误或伙伴为 SIMATIC S5: <ul style="list-style-type: none"> <li>• 源/目标类型非法, 或</li> <li>• 伙伴站上的存储器错误, 或</li> <li>• 伙伴上的 CP/CPU 通讯错误, 或</li> <li>• 伙伴站处于 STOP 状态。</li> </ul>	<ul style="list-style-type: none"> <li>• 检查伙伴是否可以传输请求的数据类型。</li> <li>• 检查伙伴的硬件结构。</li> <li>• 将伙伴站的模式选择器开关设置为 RUN。</li> </ul>
(09)08H	仅对于 RK 512: 伙伴检测到同步错误: 消息帧顺序错误。	此错误在重新启动您自己的或伙伴的可编程控制器时发生。这是正常的系统启动行为。不需要进行任何修正。在运行过程中, 此错误也可能作为前一个错误的结果而出现。对于其它情况, 您可假定是伙伴设备的故障。
(09)09H	仅对于 RK 512: 在伙伴处 DB/DX 被协调标志锁定。	<ul style="list-style-type: none"> <li>• 在伙伴程序中: 处理完最后一个输数据后复位协调存储器!</li> <li>• 在程序中: 重复请求!</li> </ul>
(09)0AH	仅对于 RK 512: 伙伴检测到消息帧头出错: 头中的第 3 个命令字节错误	检查错误是否是干扰或伙伴故障造成的。可使用数据链接中互连的接口测试设备对此进行验证。
(09)0CH	仅对于 RK 512: 伙伴检测到不正确的消息帧长度 (总长)。	检查错误是否是干扰或伙伴故障造成的。可使用数据链接中互连的接口测试设备对此进行验证。
(09)0DH	仅对于 RK 512: 尚未重新启动伙伴。	重新启动伙伴站或将模式选择器开关设置为 RUN。
(09)0EH	仅对于 RK 512: 在响应消息帧中收到的错误编号未知。	检查错误是否是干扰或伙伴故障造成的。可使用数据链接中互连的接口测试设备对此进行验证。

事件类别 10 (0AH): “CPU 已检测到的伙伴的响应消息帧出错”		
事件编号	事件	补救措施
(0A)02H	仅对于 RK 512: 收到的响应消息帧的结构中的错误 (首字节不是 00H 或 FFH)	可以使用数据链接中互联的接口测试设备检查伙伴设备的故障。
(0A)03H	仅对于 RK 512: 收到的响应消息帧的数据量过多或过少。	可以使用数据链接中互联的接口测试设备检查伙伴设备的故障。
(0A)05H	仅对于 RK 512: 在监视时间内没有收到伙伴的响应消息帧。	伙伴设备速度是否太慢? 此错误也经常作为上一个错误的结果而出现。例如, 在发送 <b>FETCH</b> 消息帧后可以显示程序接收错误 (事件类别 8)。 原因: 由于干扰而无法收到响应消息帧。已超出监视时间。如果在伙伴能对最后收到的 <b>FETCH</b> 消息帧响应之前重新启动该伙伴, 也可能发生此错误。

事件类别 11 (0BH): “警告”		
事件编号	事件	补救措施
(0B)01H	接收缓冲区容量的 2/3 已满	更频繁地调用接收块可防止接收缓冲区溢出。

## 6.10.9 SFB 的参数

## SFB 60 “SEND\_PTP” 的参数

参数	声明	数据类型	说明	值范围	缺省
REQ	IN	BOOL	在正跳沿发起请求	TRUE/FALSE	FALSE
R	IN	BOOL	取消请求。发送被阻。	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址	依 CPU 而定	3FF（十六进制）
DONE	OUT	BOOL	请求完成，没有错误	TRUE/FALSE	FALSE
ERROR	OUT	BOOL	请求已完成且有错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	错误 ID	0 到 FFFF （十六进制）	0
SD_1	IN_OUT	ANY	发送参数： 在此指定： <ul style="list-style-type: none"> <li>要从其中发送数据的 DB 号。</li> <li>要发送数据的起始数据字节号。</li> </ul> 例如：DB10（从字节 2 开始）-> DB10.DBB2	依 CPU 而定	0
LEN	IN_OUT	INT	在此指定要传输的数据块的字节长度	1 到 1024	1

## SFB 61 “RCV\_PTP” 的参数

参数	声明	数据类型	说明	值范围	缺省
EN_R	IN	BOOL	启用接收	TRUE/FALSE	FALSE
R	IN	BOOL	取消请求	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址	依 CPU 而定	3FF（十六进制）
NDR	OUT	BOOL	请求已完成且没有错误	TRUE/FALSE	FALSE
ERROR	OUT	BOOL	请求已完成且有错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	错误 ID	0 到 FFFF （十六进制）	0
RD_1	IN_OUT	ANY	接收参数： 在此指定： <ul style="list-style-type: none"> <li>存储接收数据的 DB 号。</li> <li>要用于存储接收数据的起始数据字节号。</li> </ul> 例如：DB20（从字节 5 开始）-> DB20.DBB5	依 CPU 而定	0
LEN	IN_OUT	INT	数据长度输出（字节数）	0 到 1024	0

## SFB 62 “RES\_RCVB” 的参数

参数	声明	数据类型	说明	值范围	缺省
REQ	IN	BOOL	在正跳沿启动请求	TRUE/FALSE	FALSE
R	IN	BOOL	取消请求	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址	依 CPU 而定	3FF (十六进制)
DONE	OUT	BOOL	请求已完成且没有错误	TRUE/FALSE	FALSE
ERROR	OUT	BOOL	请求已完成且有错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	错误 ID	0 到 FFFF (十六进制)	0

## SFB 63 “SEND\_RK” 的参数

参数	声明	数据类型	说明	值范围	缺省
SYNC_DB	IN	INT	DB 号，在其中存储用于 RK SFB 同步的公用数据（最小长度为 240 字节）。	CPU 特定，不允许使用零。	0
REQ	IN	BOOL	在正跳沿启动请求	TRUE/FALSE	FALSE
R	IN	BOOL	取消请求。发送被阻止。	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址。	依 CPU 而定	3FF (十六进制)
R_CPU	IN	INT	伙伴 CPU 号 (只用于多处理器模式)	0 到 4	1
R_TYPE	IN	CHAR	伙伴 CPU 的地址类型 <ul style="list-style-type: none"> <li>‘D’ = 数据块</li> <li>‘X’ = 扩展数据块</li> </ul>	‘D’ , ‘X’	‘D’
R_DBNO	IN	INT	伙伴 CPU 的数据块号	0 到 255	0
R_OFFSET	IN	INT	伙伴 CPU 的数据字节号	0 到 510 (只允许偶数值)	0
R_CF_BYT	IN	INT	伙伴 CPU 的处理器间通讯标志字节 (255: 表示: 没有处理器间通讯标记)	0 到 255	255
R_CF_BIT	IN	INT	伙伴 CPU 的处理器间通讯标志字节	0 到 7	0
DONE	OUT	BOOL	请求已完成且没有错误	TRUE/FALSE	FALSE
ERROR	OUT	BOOL	请求已完成且有错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	错误 ID	0 到 FFFF (十六进制)	0
SD_1	IN_OUT	ANY	发送参数: 在此指定: <ul style="list-style-type: none"> <li>要从其中发送数据的 DB 号。</li> <li>要发送数据的起始数据字节号。</li> </ul> 例如: DB10 (从字节 2 开始) -> DB10.DBB2	依 CPU 而定	0
LEN	IN_OUT	INT	在此指定要传输的数据块的字节长度	1 到 1024	1

## SFB 64 “FETCH\_RK” 的参数

参数	声明	数据类型	说明	值范围	缺省
SYNC_DB	IN	INT	DB 号，在其中存储用于 RK SFB 同步的公用数据（最小长度为 240 字节）。	依 CPU 而定，不允许为零。	0
REQ	IN	BOOL	在正跳沿启动请求	TRUE/FALSE	FALSE
R	IN	BOOL	取消请求	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址。	依 CPU 而定	3FF（十六进制）
R_CPU	IN	INT	伙伴 CPU 号 (仅用于多处理器模式)	0 到 4	1
R_TYPE	IN	CHAR	伙伴 CPU 上的地址类型 <ul style="list-style-type: none"> <li>‘D’ = 数据块</li> <li>‘X’ = 扩展数据块</li> <li>‘M’ = 存储器位</li> <li>‘E’ = 输入</li> <li>‘A’ = 输出</li> <li>‘Z’ = 计数器</li> <li>‘T’ = 定时器</li> </ul>	‘D’、‘X’、 ‘M’、‘E’、 ‘A’、‘Z’、 ‘T’	‘D’
R_DBNO	IN	INT	伙伴 CPU 上的数据块号	0 到 255	0
R_OFFSET	IN	INT	伙伴 CPU 上的数据字节号	参见表：“FB 中用于数据源（伙伴 CPU）的参数”	0
R_CF_BYT	IN	INT	伙伴 CPU 的处理器间通讯标志字节 (255: 表示: 没有处理器间通讯标志)	0 到 255	255
R_CF_BIT	IN	INT	伙伴 CPU 的处理器间通讯标志字节	0 到 7	0
DONE	OUT	BOOL	请求已完成且没有错误	TRUE/FALSE	FALSE
ERROR	OUT	BOOL	请求已完成且有错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	错误 ID	0 到 FFFF (十六进制)	0
RD_1	IN_OUT	ANY	接收参数: 在此指定: <ul style="list-style-type: none"> <li>存储所获取的数据的 DB 号。</li> <li>存储所获取的数据的起始数据字节号。</li> </ul> 例如: DB10 (从字节 2 开始) -> DB10.DBB2	依 CPU 而定	0
LEN	IN_OUT	INT	在此指定要获取的消息帧的字节长度。 每个计时器和计数器必须声明两个字节的长度。	1 到 1024	1

## 用于接收/提供数据的 SFB 65 “SERVE\_RK” 的参数

参数	声明	数据类型	说明	值范围	缺省
SYNC_DB	IN	INT	DB 号，在其中存储用于 RK SFB 同步的公用数据（最小长度为 240 字节）	依 CPU 而定，不允许为零。	0
EN_R	IN	BOOL	启用请求	TRUE/FALSE	FALSE
R	IN	BOOL	取消请求	TRUE/FALSE	FALSE
LADDR	IN	WORD	在“HW Config”中指定的子模块 I/O 地址。	依 CPU 而定	3FF（十六进制）
L_TYPE	OUT	CHAR	接收数据： 本地 CPU 中目标区域的类型（只允许大写字母）	‘D’	''
			提供数据： 本地 CPU 中源区域的类型（只允许大写字母）：		
			<ul style="list-style-type: none"> <li>‘D’ = 数据块</li> </ul>		
			<ul style="list-style-type: none"> <li>‘D’ = 数据块</li> <li>‘M’ = 存储器位</li> <li>‘E’ = 输入</li> <li>‘A’ = 输出</li> <li>‘Z’ = 计数器</li> <li>‘T’ = 定时器</li> </ul>		
L_DBNO	OUT	INT	本地 CPU 上的数据块号（目的地）	依 CPU 而定，不允许为零。	0
L_OFFSET	OUT	INT	本地 CPU 上的数据字节号（目的地）	0-510	0
L_CF_BYT	OUT	INT	本地 CPU 的处理器间通讯标志字节（255：表示：没有处理器间通讯标志）	0 到 255	0
L_CF_BIT	OUT	INT	本地 CPU 的处理器间通讯标志位	0 到 7	0
NDR	OUT	BOOL	请求已完成且没有错误	TRUE/FALSE	FALSE
ERROR	OUT	BOOL	请求已完成且有错误	TRUE/FALSE	FALSE
STATUS	OUT	WORD	错误 ID	0 到 FFFF（十六进制）	0
LEN	IN_OUT	INT	用字节表示的消息帧长度	0 到 1024	0

也参见

错误消息（页码 340）





## 控制

### 7.1 概述

#### 7.1.1 集成控制的原理

##### 概述

以下 SFB 可用于通过 CPU 313C、CPU 313C-2 DP/PtP 和 CPU 314C-2 DP/PTP 来实现控制功能：

- SFB 41 用于连续控制 (CONT\_C)
- SFB 42 用于分步控制 (CONT\_S)
- SFB 43 用于脉冲宽度调制 (PULSEGEN)

这些 SFB 与 FB 41 到 43 兼容。该软件控制块解决方案在每个块中可提供完善的控制器功能。周期性计算所需的数据存储在指定的 DB 中（背景数据块）。这样便允许对 SFB 的多重调用。SFB PULSEGEN 常与 SFB CONT\_C 组合用来获得能够生成控制比例执行器（例如加热和冷却机组）的脉冲输出的控制器。

##### 基本功能

借助 SFB 创建的控制器由许多可组态的单元组成。除了具有 PID 算法的实际控制器外，还集成了用于处理设定值和实际值的功能以及用于后期处理计算得出的操作值的功能。

##### 应用

原则上，用两个控制块创建的控制不偏向哪个具体的应用领域。控制效率（体现为处理速度）完全取决于所使用的 CPU 的性能。对于给定 CPU，必须在控制器数量和控制器要求的处理频率之间找到平衡。所连接的控制电路越快，也就是每个时间单位必须更为频繁地计算调节值，则可安装的控制器的数量就越少。对于可控制过程的类型没有限制。慢速（温度、填充量等）以及快速控制系统（流量、速度等）均可进行控制。

## 控制系统分析

控制系统的静态响应（增益）和动态属性（延迟、停滞时间、积分常数等）构成了控制器布局或设计以及计算其静态参数（比例操作）和动态参数（积分和微分操作）的决定性因素。

因此必须精确了解控制系统的类型和特性。

可选软件包“PID Self Tuner”可用于帮助您实现控制优化。

## 控制器选择

控制系统的属性由工艺过程/机器环境确定且可能受到严重影响。因此，要达到良好的控制效果并使控制器适应系统的恢复特性，必须选择最适合的系统控制器类型。

## 创建控制

您无需编程即可创建范围广泛的控制（从结构和参数分配到由系统程序执行的面向时间的调用）。然而，必须掌握 STEP 7 的知识。

## 在线帮助

STEP 7 在线帮助也提供了有关各个 SFB 的信息。

## 其它信息

集成控制是标准控制的一部分。有关“标准控制”主题的更多信息，请参见：

- “标准 PID 控制”：“SIMATIC S7 手册”和具有现成控制器结构和方便的参数分配窗口的组态包。
- “模块化 PID 控制”：“SIMATIC S7 手册”和组态包，该组态包含有灵活的控制工具包，对于复杂任务也适合。
- “通过 SIMATIC 进行控制”，作者 Jürgen Müller：通过 SIMATIC S7 和 SIMATIC PCS7 进行控制的实用手册。
- “PID 自调节器”：SIMATIC S7 手册和用于 PID 控制器在线自我优化的软件包。
- FM 355/FM 455 作为独立的控制器模块，不增加 CPU 的负载。

## 7.1.2 基本信息

### 连续/开关控制器

连续控制器输出线性（模拟）值。

开关控制器输出二进制（数字）值。

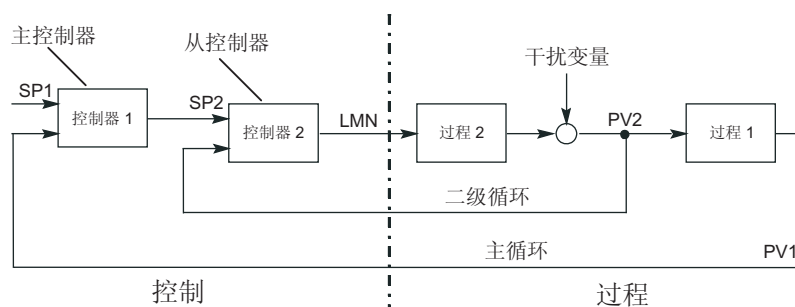
### 设定值控制

设定值控制是指用一个仅可能偶尔进行修改的固定参考变量实施的控制。这种控制将调整过程中的偏差。

### 串级控制

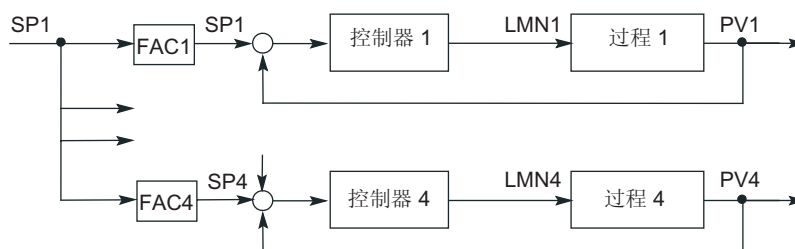
串级控制表示将控制器串联起来实现控制。第一个控制器（主控制器）决定串联（从）控制器的设定值，或者根据过程变量的实际误差影响从控制器的设定值。

串级控制的控制性能可通过附加过程变量来改进。为此，需要将适当控制点获得的辅助过程变量  $PV2$  组合到主过程变量（主控制器输出  $SP2$ ）中。主控制器将过程变量  $PV1$  应用于设定值  $SP1$ ，并调整  $SP2$  以便尽快实现控制目标而不过调。



### 混合控制

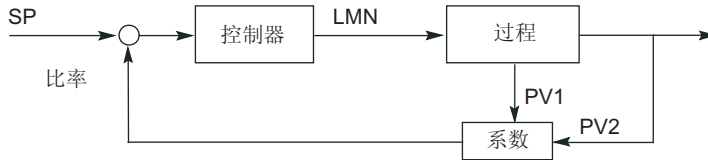
混合控制代表一种控制结构，其中通过将各个受控元件的需要量按一定的百分比来计算，而获得控制总量的设定值  $SP$ 。在此，混合因子  $FAC$  的总和必须为 1。



### 比率控制

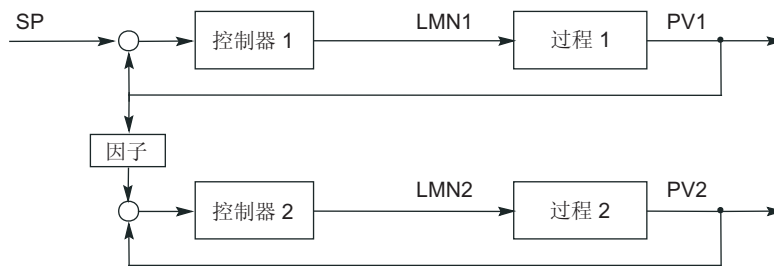
- 单循环比率控制器

举例来说，在两个过程变量之比相对过程变量的绝对值更重要时使用单循环比率控制（例如速度控制）。



- 多循环比率控制器

对于多循环比率控制，两个过程变量 PV1 和 PV2 之比保持恒定。此处，第二个控制循环的设定值用第一个控制循环的过程值来计算。对于过程变量 x1 的动态变化，也可确保维持指定的比率。



### 两步控制

两步控制只能获得两个输出状态（例如 On-Off）。典型的控制是通过继电器输出控制加热系统的脉冲宽度调制。

### 三步控制

三步控制只能获得三个离散的输出状态。此处必须区分脉冲宽度调制，例如加热和冷却（加热 - 停止 - 冷却），与使用集成执行器的分步控制（例如右 - 停 - 左）。

## 7.2 接线

### 7.2.1 接线规则

#### 基本信息


控制器不具有集成 I/O。必须使用 CPU 的空闲 I/O 或附加 I/O 模块进行输入和输出。

#### 连接电缆

- 连接数字 I/O 的电缆如果长度超过 100m，则必须屏蔽。
- 在电缆屏蔽的两端，必须对屏蔽层进行端接。
- 软电缆，横截面积为 0.25 到 1.5 mm<sup>2</sup>。
- 无需电缆套。如果坚持使用电缆套，请使用不带绝缘环的电缆套（DIN 46228，A 形，短型）。

#### 屏蔽端接元件

可使用屏蔽端接元件将所有屏蔽电缆通过成型导轨直接接地。

 <b>警告</b>
如果不切断电源，可能会危及人身安全和导致财产损失。 如果连接带电模块的前插头，会有触电危险。 必须在断电状态下连接模块！

#### 其它信息

有关其它信息，请参见 *CPU 数据手册* 和 *CPU 安装说明*。

## 7.3 参数组态

### 7.3.1 用参数分配窗口组态 SFB

#### 参数分配窗口

借助“PID 控制”参数屏幕为 SFB 41、42 和 43 组态参数（背景数据块）。

这些参数分配窗口基本都是自说明的。可以在『用 SFB 41 “CONT\_C” 实现连续控制（页码 360）』一节和参数分配窗口的在线帮助中找到参数说明。

#### 步骤

要求：先前已经在 S7 程序中实现了具有背景数据块的 SFB。SFB 在“标准库”的“系统功能块”下。

1. 通过“SIMATIC” / “STEP7” / “组态 PID 控制” (Configure PID control) 打开参数分配窗口。
2. 在“PID 控制”下，通过“文件 > 打开”打开项目，选择背景数据块。
3. 自定义参数。
4. 保存参数（在背景数据块中），将程序下载到您的 CPU 中。

#### 在线帮助

分配参数时，参数分配窗口的在线帮助可为您提供支持。可通过如下几种方式来调用在线帮助：

- 通过菜单命令“帮助 > 帮助主题...”
- 在相应的视图中，按下 **F1** 键。

## 7.4 在用户程序中实现控制

### 概述

下表概要说明了模块的控制功能及其已分配的 SFB:

功能	SFB
连续控制器	SFB CONT_C (SFB 41)
步控制器	SFB CONT_S (SFB 42)
脉冲宽度调制	SFB PULSEGEN (SFB 43)

SFB 在“标准库”的“系统功能块”下。

以下各节可帮助您为您的应用程序设计用户程序。

### 调用 SFB

通过相应的背景数据块调用 SFB

实例: CALL SFB 41, DB30

### 背景数据块

SFB 参数存储在背景数据块中。在『用 SFB 41 “CONT\_C”实现连续控制 (页码 360)』一节中介绍这些参数。

可以通过以下方式访问参数:

- DB 号和偏移量地址
- DB 号和 DB 中的符号地址

### 程序结构

SFB 必须在重启 OB 和超时中断 OB 中调用。方案:

OB100      调用 SFB 41, 42, 43  
OB35        调用 FB 41, 42, 43

## 7.5 功能说明

### 7.5.1 用 SFB 41 “CONT\_C” 实现连续控制

#### 引言

SFB “CONT\_C”（连续控制器）用于控制 SIMATIC S7 自动化系统上有连续 I/O 变量的工艺过程。可以通过参数开启或关闭 PID 控制器的部分操作，从而使它与受控制的系统相适应。使用参数分配窗口很容易就可以做到这一点（菜单路径：“开始” [Start] > “Simatic” > “STEP 7” > “分配 PID 控制参数” [Assign PID control parameters]）。可在“开始 > Simatic > S7 手册 > PID 控制（英文）”下找到在线电子手册。

#### 应用

可以把控制器用作单一的 PID 固定设定值控制器，或者在多控制循环中用作串级、混合或比率控制。控制器的功能基于采样控制器的 PID 控制算法，增补了通过模拟起动信号生成二进制输出信号的功能。

#### 说明

除了设定值和过程值分支中的功能外，SFB/FB 实现了完整的 PID 控制器，可连续输出调节值并可以手动控制调节值。

下面是对这些子功能的详细描述：

##### 设定值操作

设定值在输入 **SP\_INT** 处以浮点值格式输入。

##### 实际值操作

过程变量可以以外设 (I/O) 输入或浮点值格式输入。CRP\_IN 函数会按以下公式将 PV\_PER 外设定值转换成 -100 到 +100 % 的浮点值格式：

$$\text{输出 CPR\_IN} = \text{PV\_PER} \times \frac{100}{27648}$$

PV\_NORM 函数按以下公式标准化 CRP\_IN 的输出。

$$\text{输出 PV\_NORM} = (\text{输出 CPR\_IN}) \times \text{PV\_FAC} + \text{PV\_OFF}$$

PV\_FAC 的缺省值是 1，PV\_OFF 的缺省值是 0。

变量 PV\_FAC 和 PV\_OFF 是以下公式转换的结果：

$$\text{PV\_OFF} = (\text{输出 PV\_NORM}) - (\text{输出 CPR\_IN}) \times \text{PV\_FAC}$$

$$\text{PV\_FAC} = \frac{(\text{PV\_NORM 的输出}) - \text{PV\_OFF}}{\text{CPR\_IN 的输出}}$$

并非必须转换成百分数值。如果需要物理上确定设定值，则实际值也可转换成此物理值。



### 计算负偏差

设定值和实际值之间的差构成负偏差。为了抑制调节值量化导致的微小连续振荡（例如通过 PULSEGEN 的脉冲宽度调制），需要将负偏差应用到一个死区 (DEADBAND)。如果 DEADB\_W = 0，则死区已关闭。

### PID 算法

PID 算法充当定位算法。比例、积分 (INT) 和微分 (DIF) 操作并行连接，并可分别激活或取消激活。从而可组态 P、PI、PD 和 PID 控制器。不过，也可实现独立的 I 或 D 控制器。

### 手动模式

可以在手动和自动模式之间切换。在手动模式中，调节值被修正为手动选择的值。

积分器 (INT) 内部设定为 LMN-LMN\_P-DISV，微分单元 (DIF) 设定为 0 且内部匹配。这意味着转换为自动模式并不会导致调节值突然变化。

### 调节值处理

使用 LMNLIMIT 功能，可将调节值限制为一个选定的值。信号位会指出输入变量超出限制的情况。

LMN\_NORM 函数按以下公式规格化 LMNLIMIT 的输出：

$$\text{LMN} = (\text{LMNLIMIT 的输出}) \times \text{LMN\_FAC} + \text{LMN\_OFF}$$

LMN\_FAC 的缺省值是 1，LMN\_OFF 的缺省值是 0。

调节值也会以外设 (I/O) 格式提供。CPR\_OUT 函数按以下公式把浮点值 LMN 转换成外设值：

$$\text{LMN\_PER} = \text{LMN} \times \frac{2764}{100}$$

### 误差值混合

干扰变量可在输入 DISV 处前向馈入。

## 初始化

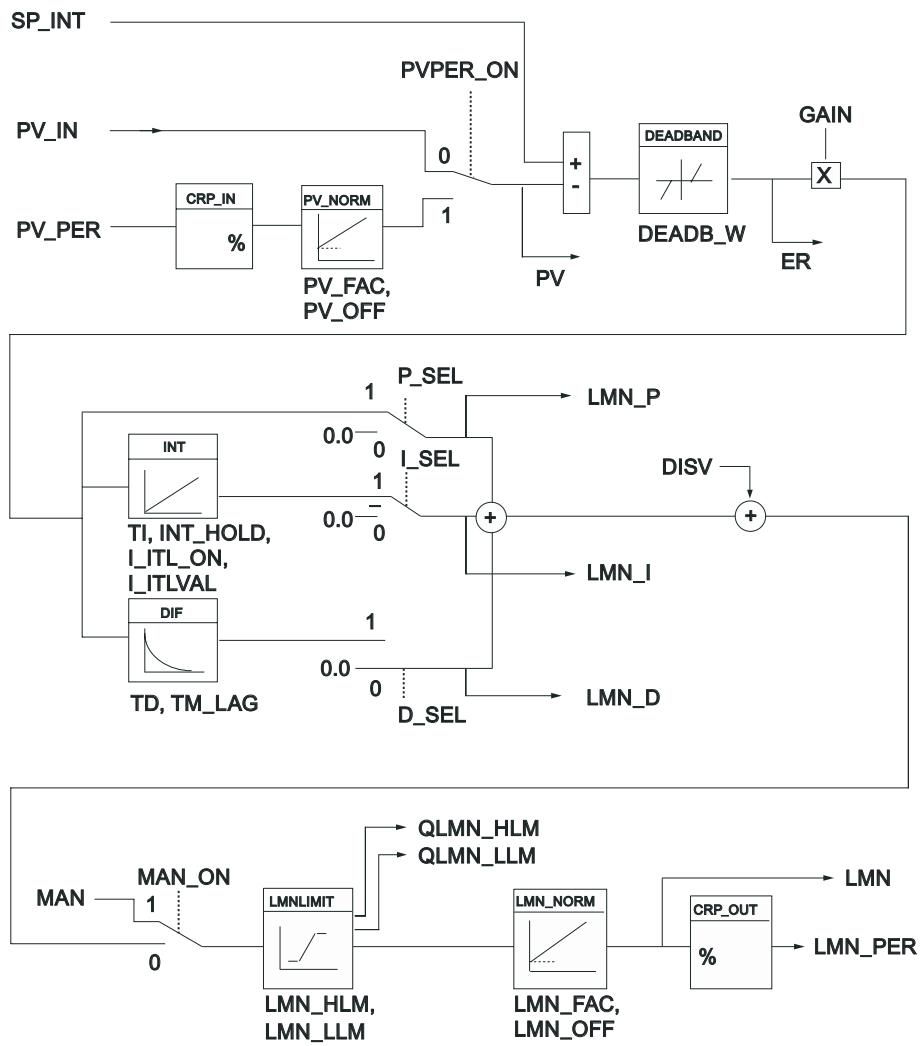
SFB “CONT\_C” 有一个初始化例程，在设置输入参数 COM\_RST = TRUE 时运行。

初始化期间，在内部将积分器设定为初始化值 \_LITVAL。如果在一个循环中断优先级等级中被调用，它将以此值开始继续运行。所有其它输出被设置为它们的缺省值。

## 错误信息

通过“参数分配工具”执行参数检查。

方框图 CONT\_C



## SFB 41 的参数

下表包含 SFB 41 “CONT\_C” 的输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
COM_RST	BOOL	0.0	完全重启动 该块有一个初始化例程，在输入 COM_RST 置位时进行处理。	TRUE: 重启动 FALSE: 控制器操作	FALSE
MAN_ON	BOOL	0.1	启用手动值 如果输入“启用手动值”置位，则控制循环中断。设置手动值作为调节值。		TRUE
PVPER_ON	BOOL	0.2	启用外围过程变量 如果从 I/O 读入过程变量，输入 PV_PER 必须连接到外设且输入“启用外围过程变量”必须置位。		FALSE
P_SEL	BOOL	0.3	启用比例操作 PID 算法允许启用/禁用单个的 PID 操作。P 操作会在输入“启用比例操作”置位时启用。		TRUE
I_SEL	BOOL	0.4	启用积分操作 PID 算法允许启用/禁用单个的 PID 操作。I 操作会在输入“启用积分操作”置位时启用。		TRUE
INT_HOLD	BOOL	0.5	冻结积分操作 可以冻结积分器输出。为此，输入“冻结积分操作”必须置位。		FALSE
I_ITL_ON	BOOL	0.6	积分操作初始化 可以在输入 I_ITLVAL 处使积分器输出置位。为此，输入“积分操作初始化”必须置位。		FALSE
D_SEL	BOOL	0.7	启用微分操作 PID 算法允许启用/禁用单个的 PID 操作。D 操作会在输入“启用微分操作”置位时启用。		FALSE
CYCLE	TIME	2	采样时间 块调用之间的时间必须恒定。输入“采样时间”指定块调用之间的时间。	$\geq 20$ ms	T#1s
SP_INT	REAL	6	内部设定值 输入“内部设定值”用来指定设定值。	-100.0...100.0 (%) 或物理值 <sup>1)</sup>	0.0
PV_IN	REAL	10	过程变量输入 初始化值可以在输入“过程变量输入”处设定，或者可以连接浮点格式的外部过程变量。	-100.0...100.0 (%) 或物理值 <sup>1)</sup>	0.0
PV_PER	WORD	14	外围过程变量 I/O 格式的过程变量在输入“外围过程变量”处连接到控制器。		W#16# 0000
MAN	REAL	16	手动值 输入“手动值”用于通过操作员控制/监视功能设置缺省手动值。	-100.0...100.0 (%) 或物理值 <sup>2)</sup>	0.0

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
<b>GAIN</b>	REAL	20	比例增益 输入“比例增益”用于设置控制器增益。	前缀指定控制器的操作方向（例如负增益用于冷却操作）。	2.0
<b>TI</b>	TIME	24	复位时间 输入“复位时间”确定积分器的时间响应。	>= CYCLE	T#20 s
<b>TD</b>	TIME	28	微分时间 输入“微分时间”决定微分单元的时间响应。	>= CYCLE	T#10s
<b>TM_LAG</b>	TIME	32	微分操作的时延/D操作的延迟时间 D操作的算法包括时间延迟，该延迟可分配给输入“微分操作的时延”。	>= CYCLE/2 推荐 1/5 TD	T#2s
<b>DEADB_W</b>	REAL	36	死区宽度 死区应用于错误。输入“死区宽度”决定死区的大小。	>= 0.0 (%) 或物理值 <sup>1)</sup>	0.0
<b>LMN_HLM</b>	REAL	40	调节值上限 调节值总是要限定上限和下限。输入“调节值上限”指定上限。	LMN_LLM ... 100.0 (%) 或物理值 <sup>2)</sup>	100.0
<b>LMN_LLM</b>	REAL	44	调节值下限 调节值总是要限定上限和下限。输入“调节值下限”指定下限。	-100.0... LMN_HLM (%) 或物理值 <sup>2)</sup>	0.0
<b>PV_FAC</b>	REAL	48	过程变量因子 输入“过程变量因子”用来与过程变量相乘。该输入用来调整过程变量的范围。		1.0
<b>PV_OFF</b>	REAL	52	过程变量偏移量 输入“过程变量偏移量”用来与过程变量相加。该输入用于调整过程变量的范围。		0.0
<b>LMN_FAC</b>	REAL	56	调节值因子 输入“调节值因子”用来与调节值相乘。该输入用来调整调节值的范围。		1.0
<b>LMN_OFF</b>	REAL	60	调节值偏移量 输入“调节值偏移量”用来与调节值相加。该输入用来调整调节值的范围。		0.0
<b>I_ITLVAL</b>	REAL	64	积分操作的初始化值/ 积分器的输出可以在输入 I_ITL_ON 处设置。初始化值在输入“积分操作的初始化值”处指定。	-100.0...100.0 (%) 或物理值 <sup>2)</sup>	0.0
<b>DISV</b>	REAL	68	干扰变量 对于前馈控制，干扰变量连接到输入“干扰变量”。	-100.0...100.0 (%) 或物理值 <sup>2)</sup>	0.0

1) 在设定值和实际值操作中的参数单位相同

2) 在限制操作中的参数单位相同

下表说明了 SFB 41 “CONT\_C” 的输出参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
LMN	REAL	72	调节值 有效调节值以浮点格式在输出“调节值”处输出。		0.0
LMN_PER	WORD	76	外围调节值 I/O 格式的调节值在输入“外围调节值”处连接到控制器。		W#16#0000
QLMN_HLM	BOOL	78.0	达到调节值上限 调节值总是要限定上限和下限。输出“达到调节值上限”用于指示上限被超出。		FALSE
QLMN_LLM	BOOL	78.1	达到调节值下限 调节值总是要限定上限和下限。输出“达到调节值下限”用于指示下限被超出。		FALSE
LMN_P	REAL	80	比例分量/ 输出“比例分量”含有调节值的比例分量。		0.0
LMN_I	REAL	84	积分分量 输出“积分分量”含有调节值的积分分量。		0.0
LMN_D	REAL	88	微分分量 输出“微分分量”含有调节值的微分分量。		0.0
PV	REAL	92	过程变量 有效过程变量在输出“过程变量”处输出。		0.0
ER	REAL	96	误差信号 有效误差在输出“误差信号”处输出。		0.0

## 7.5.2 使用 SFB 42 “CONT\_S” 进行步控制

### 引言

SFB/FB “CONT\_S”（步控制器）用在 SIMATIC S7 可编程逻辑控制器上，以使用集成执行器的二进制调节值输出信号控制工艺过程。在参数分配期间，可以激活或取消激活 PI 步控制器的子功能，以使控制器与工艺过程相适应。使用参数分配窗口很容易就可以做到这一点（菜单路径：“开始” [Start] > “Simatic” > “STEP 7” > “分配 PID 控制参数” [Assign PID control parameters]）。可在“开始” (Start) > “Simatic” > “S7 手册” (S7 Manuals) > “PID 控制（英文）(PID Control English)” 下找到在线电子手册。

### 应用

可以将控制器用作 PI 固定设定值控制器，或者二级控制循环中用作串联、混合或比率控制器，但是不能用作主控制器。控制器的功能基于采样控制器的 PI 控制算法，增补了通过模拟起动信号生成二进制输出信号的功能。

可以用  $TI = T\#0 \text{ ms}$  来关闭控制器的 I 操作。因此可将块用作 P 控制器。

由于控制器可以在没有任何位置反馈信号的情况下工作，因此内部计算的调节变量与信号控制元件的位置不完全匹配。如果调节变量 (ER\*GAIN) 为负值，则会进行调整。控制器随后设置输出 QLMNDN（调节值信号低），直到置位了 LMNR\_LS（位置反馈信号的下限）为止。

也可以在控制器级联中将控制器用作次级执行器。设定值输入 SP\_INT 用于分配控制元件的位置。在此情况下，实际值输入和参数 TI（积分时间）必须设置为零。应用领域包括如通过电机操作的阀瓣进行温度控制。在此情况下，要将阀完全关闭，调节变量 (ER\*GAIN) 应有一个负设置。

## 说明

除了过程值电路中的功能以外，SFB 实现了完整的 PI 控制器，该控制器可输出数字调节值，并可手动控制调节值。步控制器可在没有位置反馈信号的情况下运行。可以用限制停止信号限制脉冲输出。

下面是对这些子功能的详细描述：

### 设定值操作

设定值在输入 **SP\_INT** 处以浮点格式输入。

### 实际值操作

过程变量可以以外设 (I/O) 输入或浮点值格式输入。CPR\_IN 函数会按以下公式将 PV\_PER 外设定值转换成 -100 到 +100 % 的浮点值格式：

$$\text{CPR\_IN} = \text{PV\_PER 的输出} \times \frac{100}{27648}$$

PV\_NORM 函数按以下公式标准化 CPR\_IN 的输出：

$$\text{PV\_NORM 的输出} = (\text{CPR\_IN 的输出}) \times \text{PV\_FAC} + \text{PV\_OFF}$$

PV\_FAC 的缺省值是 1，PV\_OFF 的缺省值是 0。

变量 PV\_FAC 和 PV\_OFF 是以下公式转换的结果：

$$\text{PV\_OFF} = (\text{PV\_NORM 的输出}) - (\text{CPR\_IN 的输出}) \times \text{PV\_FAC}$$

$$\text{PV\_FAC} = \frac{(\text{PV\_NORM 的输出}) - \text{PV\_OFF}}{\text{CPR\_IN 的输出}}$$

### 计算负偏差

设定值和实际值之间的差构成负偏差。要抑制由于调节变量量化（例如由于执行器阀的调节值的有限精度）而导致的微小连续振荡，需要对误差信号应用死区 (DEADBAND)。如果 DEADB\_W = 0，则死区关闭。

### PI 分步算法

SFB 在没有位置反馈的情况下运行。PI 算法的 I 操作和假设的位置反馈信号在一个积分器 (INT) 中计算，并且作为反馈值与剩余的 P 操作比较。差值应用于三步元件 (THREE\_ST) 以及为执行器创建脉冲的脉冲生成器 (PULSEOUT)。可通过调节三步元件上的阈值减少控制器的切换频率。

### 误差值混合

干扰变量可在 **DISV** 输入时前馈。

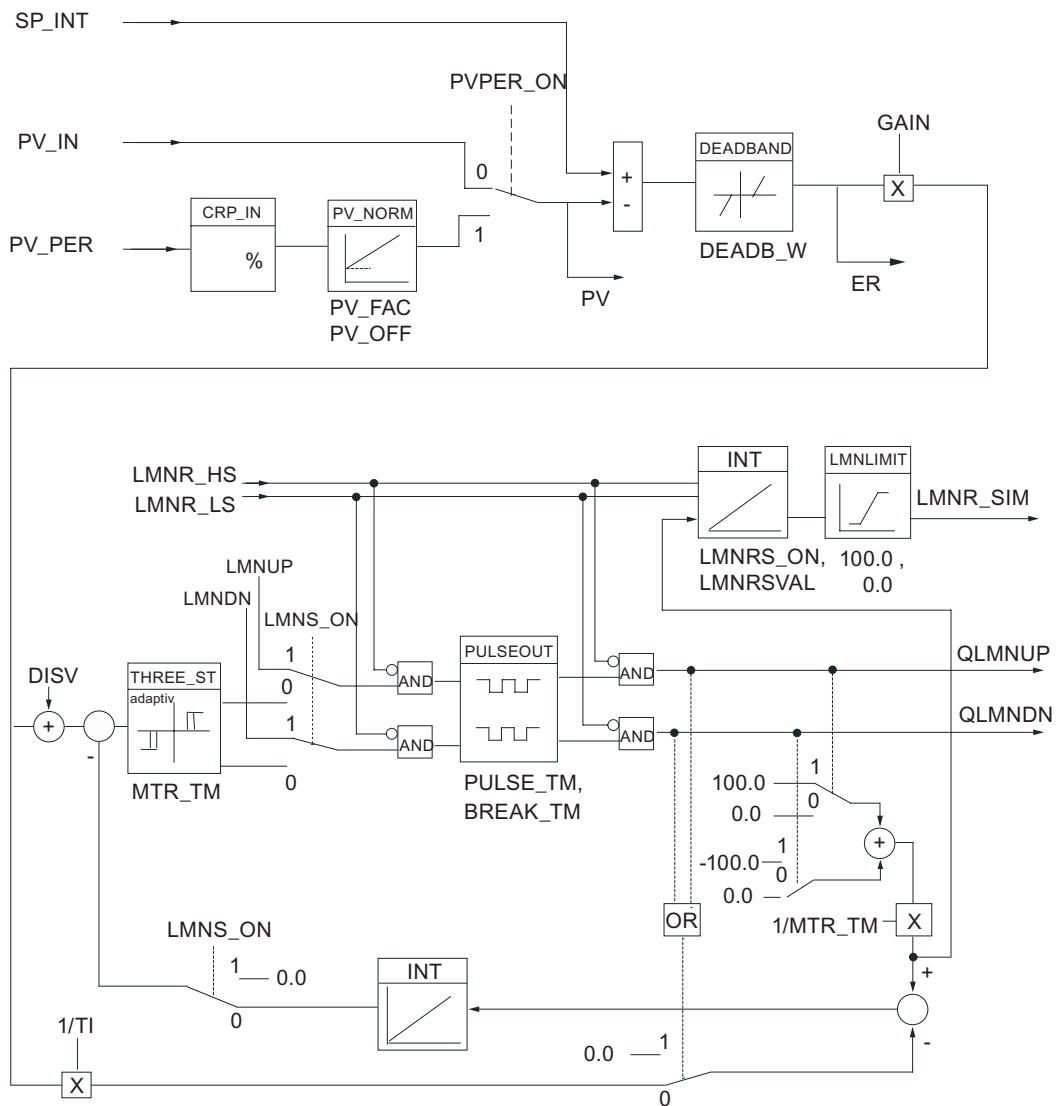
### 初始化

SFB “CONT\_S” 有一个初始化例程，该例程在设置输入参数 COM\_RST = TRUE 时运行。所有其它输出被设置为它们的缺省值。

### 错误信息

通过参数分配工具执行参数检查。

### CONT\_S 方框图





## SFB 42 的参数

下表含有 SFB 42 “CONT\_S” 的输入参数:

参数	数据类型	地址 (背景数据块)	说明	值范围	缺省
COM_RST	BOOL	0.0	完全重启动 该块有一个初始化例程, 在输入 COM_RST 置位时进行处理。	TRUE: 重启动 FALSE: 控制器操作	FALSE
LMNR_HS	BOOL	0.1	重复调节值的上限信号 “执行器在上限停止”信号被连接到“位置反馈信号的上限”输入。LMNR_HS = TRUE 表示: 控制阀在上限停止处。		FALSE
LMNR_LS	BOOL	0.2	重复调节值的下限信号 “执行器在下限停止”信号被连接到“位置反馈信号的下限”输入。LMNR_LS = TRUE 表示: 控制阀在下限停止处。		FALSE
LMNS_ON	BOOL	0.3	启用调节信号 起动信号处理在“启用手动起动信号”输入处被切换为手动。		TRUE
LMNUP	BOOL	0.4	调节信号增加 对于手动起动值信号, 输出信号 QLMNUP 在输入“执行信号增加”处置位。		FALSE
LMNDN	BOOL	0.5	调节信号下降 对于手动起动值信号, 输出信号 QLMNDN 在输入“执行信号下降”处置位。		FALSE
PVPER_ON	BOOL	0.6	启用外围过程变量 如果从 I/O 读入过程变量, 输入 PV_PER 必须连接到外设且必须置位输入“启用外围过程变量”。		FALSE
CYCLE	TIME	2	采样时间 块调用之间的时间必须恒定。输入“采样时间”指定块调用之间的时间。	>= 20 ms	T#1s
SP_INT	REAL	6	内部设定值 输入“内部设定值”用于指定设定值。	-100.0...100.0 (%) 或物理值 <sup>1)</sup>	0.0
PV_IN	REAL	10	过程变量输入 初始化值可以在输入“过程变量输入”处设置, 或者可以连接浮点格式的外部过程变量。	-100.0...100.0 (%) 或物理值 <sup>1)</sup>	0.0
PV_PER	WORD	14	外围过程变量 I/O 格式的过程变量在输入“外围过程变量”处连接到控制器。		W#16# 0000
GAIN	REAL	16	比例增益 输入“比例增益”用于设置控制器增益。	前缀指定控制器的操作方向 (例如负增益用于冷却操作)。	2.0
TI	TIME	20	复位时间 输入“复位时间”确定积分器的时间响应。	T#0 ms 或者 >= CYCLE	T#20 s

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
DEADB_W	REAL	24	死区宽度 死区应用于错误。输入“死区宽度”决定死区的大小。	100.0 (%) 或物理值 <sup>1)</sup>	1.0
PV_FAC	REAL	28	过程变量因子 输入“过程变量因子”用来与过程变量相乘。该输入用于调整过程变量的范围。		1.0
PV_OFF	REAL	32	过程变量偏移量 输入“过程变量偏移量”用于与过程变量相加。该输入用于调整过程变量的范围。		0.0
PULSE_TM	TIME	36	最小脉冲时间 可以用参数“最小脉冲时间”来分配最小脉冲持续时间。	>= CYCLE CYCLE 的整数倍	T#3 s
BREAK_TM	TIME	40	最小断线时间 可以用参数“最小断线时间”来分配最小断线持续时间。	>= CYCLE CYCLE 的整数倍	T#3 s
MTR_TM	TIME	44	电机调节值 在“电机执行时间”参数中输入执行器从限制停止移动到限制停止所需的时间。	>= CYCLE	T#30 s
DISV	REAL	48	干扰变量 对于前馈控制，干扰变量连接到输入“干扰变量”。	-100.0...100.0 (%) 或物理值 <sup>2)</sup>	0.0

1) 在设定值和实际值操作中的参数单位相同

2) 在限制操作中的参数单位相同

下表含有对 SFB 42 “CONT\_S” 的输出参数的说明：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
QLMNUP	BOOL	52.0	调节信号增加 如果设置了“执行信号增加”，则控制阀被打开。		FALSE
QLMNDN	BOOL	52.1	调节信号下降 如果设置了“执行信号下降”，则控制阀被打开。		FALSE
PV	REAL	54	过程变量 有效过程变量在输出“过程变量”处输出。		0.0
ER	REAL	58	误差信号 有效误差在输出“误差信号”处输出。		0.0

### 7.5.3 用 SFB 43 “PULSEGEN” 生成脉冲

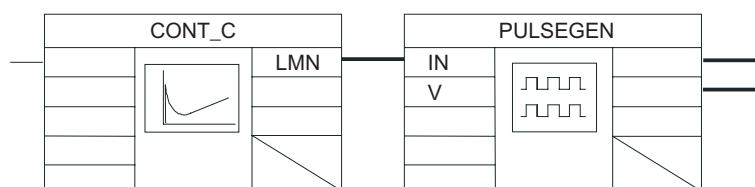
#### 引言

SFB “PULSEGEN”（脉冲生成器）用于为比例执行器构建带有脉冲输出的 PID 控制器。

可在“开始 (Start) > Simatic > S7 手册 (S7 manuals) > PID 控制英文 (PID Control English)”下找到在线电子手册。

#### 应用

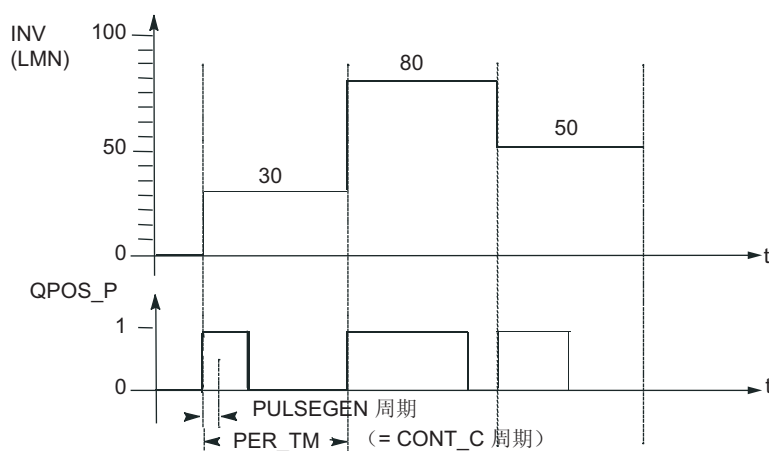
可以使用 SFB “PULSEGEN” 组态带有脉冲宽度调制的 PID 两步或三步控制器。该功能通常与连续控制器 “CONT\_C” 一起使用。



#### 说明

功能 PULSEGEN 以恒定周期将脉冲持续时间调制为脉冲序列，以传输输入变量 INV (= PID 控制器的 LMN)，该恒定周期对应于更新输入变量的周期时间，并且必须在 PER\_TM 中分配。

每个周期的脉冲持续时间与输入变量成比例。PER\_TM 中的周期组态与 SFB “PULSEGEN” 的处理周期不同。而且，PER\_TM 周期表示 SFB “PULSEGEN” 的多个处理周期之和。此处，每个 PER\_TM 周期调用 SFB “PULSEGEN” 的次数可用于评测脉冲宽度的精度。此处的最小调节值在 P\_B\_TM 参数中确定。

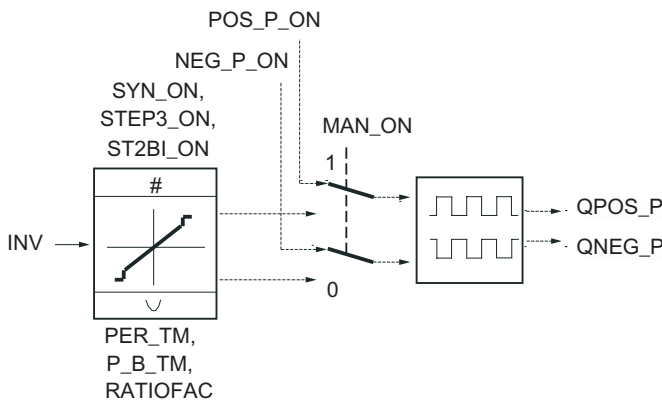


脉冲宽度调制

因此，30 % 和每个 PER\_TM 周期调用 10 次 SFB “PULSEGEN” 的输入变量意味着：

- QPOS 输出处的 “一” 表示 SFB “PULSEGEN” 的前三次调用（10 次调用的 30 %）
- QPOS 输出处的 “零” 表示 SFB “PULSEGEN” 的后七次调用（10 次调用的 70 %）

方框图



调节值的精度

在此例中，1:10（CONT\_C 调用次数与 PULSEGEN 调用次数之比）的“采样率”将调节值的精度降低至 10 %，即，缺省输入值 INV 只能以 10 % 的程度映射到输出 QPOS 上。

精度随每调用一次 CONT\_C 时 SFB “PULSEGEN” 的调用次数的增加而增加。

例如，如果 PULSEGEN 的调用次数比 CONT\_C 的调用次数多 100 倍以上，则调节值范围的精度将达到 1 %（建议精度 <= 5 %）。

说明

调用比率必须由用户编程。

## 自动同步

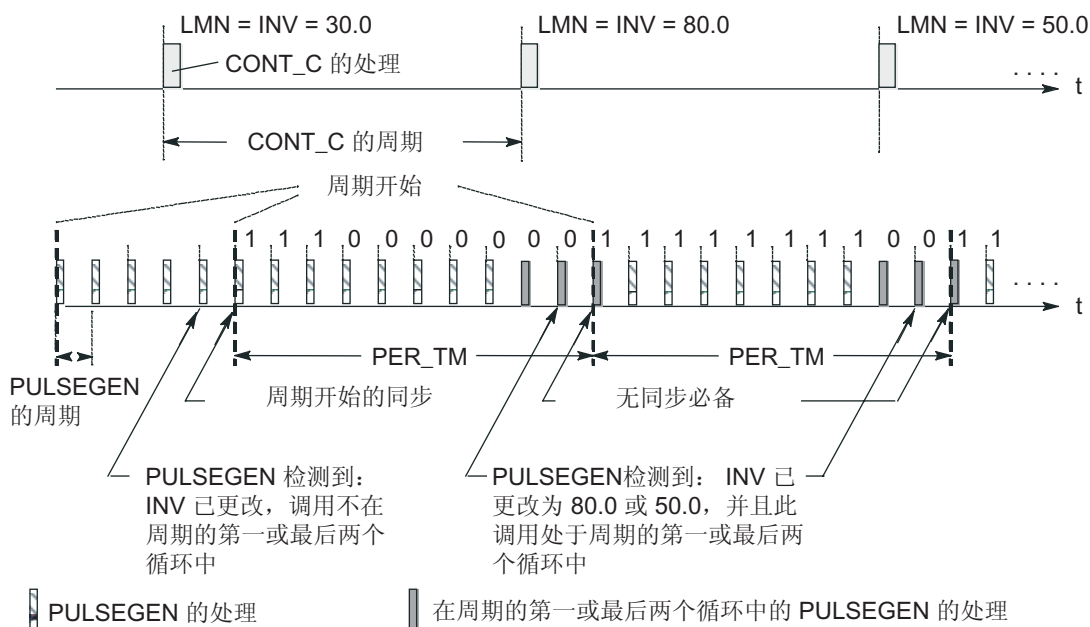
可以将脉冲输出与更新输入变量（INV，例如，CONT\_C）的块自动同步。这样可确保输入变量中的更改作为脉冲尽快输出。

脉冲生成器在对应于周期 PER\_TM 的时间间隔处评估输入值 INV，并将该值转换为相应长度的脉冲信号。

然而，由于通常用较慢循环的中断等级计算 INV，因此，脉冲生成器应当在更新 INV 之后尽快启动离散值到脉冲信号的转换。

要允许此操作，该块可以用下列步骤同步周期的开始：

如果 INV 更改并且块调用不在周期的第一个或者最后两个调用循环中，则执行同步操作。脉冲持续时间被重新计算并且在下一次循环中通过新周期输出。



每个“SYN\_ON” (= FALSE) 输入都可以关闭自动同步。

### 说明

在新周期开始时，INV 的旧值（即 LMN 的值）的映像不太精确地混入脉冲信号中。

## 操作模式

根据分配到脉冲生成器的参数，可以组态带有三步输出或者带有双极性或单极性两步输出的 PID 控制器。下表说明了可能模式的切换组合的设置：

操作模式	MAN_ON	切换 STEP3_ON	ST2BI_ON
三步控制	FALSE	TRUE	任意
带双极性的两步控制 控制范围 (-100% ... 100%)	FALSE	FALSE	TRUE
带单极性的两步控制 控制范围 (0% ... 100%)	FALSE	FALSE	FALSE
手动模式	TRUE	任意	任意

## 三步控制

“三步控制”可以为控制信号生成三个状态。二进制输出信号 QPOS\_P 和 QNEG\_P 的值被分配给执行器的状态。下表显示了温度控制的实例：

输出信号	加热	执行器关	冷却
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

根据输入变量，用一条特征曲线计算脉冲持续时间。特征曲线的形状由最小脉冲或最小断线时间以及比率因子定义。

正常的比率因子值为 1。

曲线中的“折线”是由最小脉冲或最小断线时间引起的。

### 最小脉冲或最小断线时间

正确分配的最小脉冲或最小断线时间 P\_B\_TM 可以防止开/关间隔时间过短，开/关间隔时间过短会减少开关元件和执行器的工作寿命。

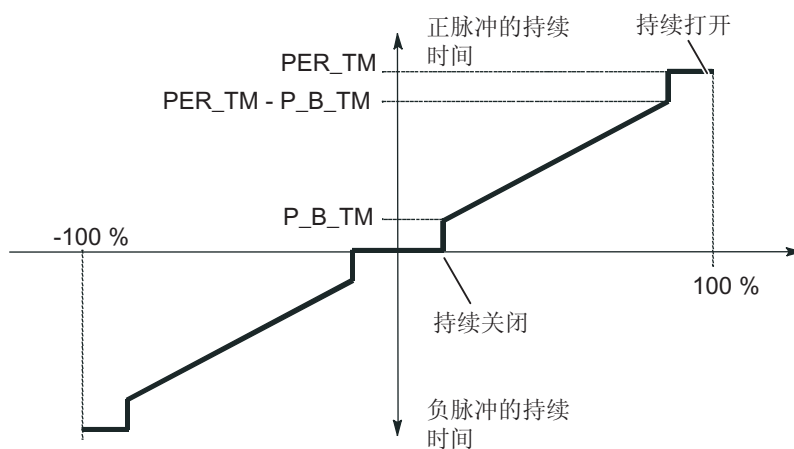
### 说明

输入变量 LMN 处的小绝对值（该值在其它位置生成比 P\_B\_TM 短的脉冲持续时间）被抑制。将生成比 (PER\_TM-P\_B\_TM) 长的脉冲持续时间的较大输入值被设置为 100 % 或 -100 %。

用输入变量（用 % 表示）乘以周期时间来计算正或负脉冲的持续时间：

$$\text{脉冲周期} = \frac{\text{INV}}{100} \times \text{PER\_TM}$$

下图显示了三步控制器的对称曲线（比率因子 = 1）：



使用比率因子 **RATIOFAC**，可以更改正负脉冲的持续时间的比率。例如，对于热处理，这可用于考虑加热和冷却执行器的不同的时间常数。

比率因子也会影响最小脉冲或最小断线时间。比率因子  $< 1$  意味着负脉冲的阈值乘上了比率因子。

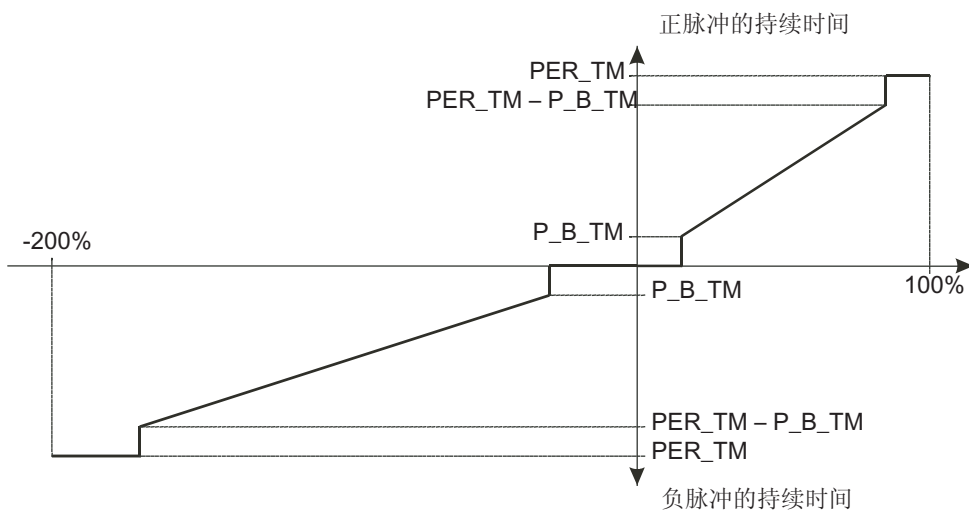
#### 比率因子 $< 1$

负脉冲输出上的脉冲周期由于比率因子而减小，该比率因子是输入值乘以脉冲周期所计算出的结果。

$$\text{正脉冲周期} = \frac{\text{INV}}{100} \times \text{PER\_TM}$$

$$\text{负脉冲周期} = \frac{\text{INV}}{100} \times \text{PER\_TM} \times \text{RATIOFAC}$$

下图显示了三步控制器的非对称曲线（比率因子 = 0.5）：



**比率因子 > 1**

正脉冲输出上的脉冲持续时间因比率因子而减小，该持续时间是输入变量乘以周期所计算出的结果。

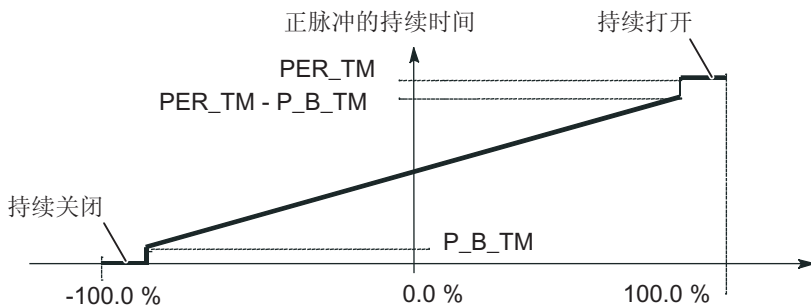
$$\text{负脉冲周期} = \frac{\text{INV}}{100} \times \text{PER\_TM}$$

$$\text{正脉冲周期} = \frac{\text{INV}}{100} \times \frac{\text{PER\_T}}{\text{RATIOFAC}}$$

**两步控制**

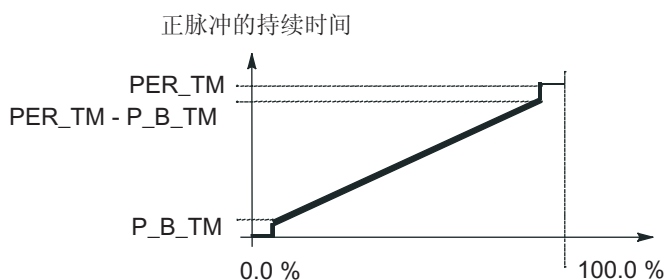
对于两步控制，仅 PULSEGEN 的正脉冲输出 QPOS\_P 连接到 I/O 执行器。根据使用的调节值范围，两步控制器有一个双极性或单极性调节值范围。

带有双极性调节值范围 (-100 %...100 %) 的两步控制：





带有单极性调节变量范围 (0%...100%) 的两步控制:



如果控制循环中两步控制器的连接需要一个逻辑反向二进制信号用于控制脉冲, 则可在 QNEG\_P 处对输出信号取反。

脉冲	执行器开	执行器关
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

### 带两步或三步控制的手动模式

在手动模式 (MAN\_ON = TRUE) 中, 可以用信号 POS\_P\_ON 和 NEG\_P\_ON 设置三步或者两步控制器的二进制输出, 而不考虑 INV。

	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	任意	FALSE	TRUE
	TRUE	任意	TRUE	FALSE

### 初始化

SFB “PULSGEN” 有一个初始化例程, 它在设置了输入参数 COM\_RST = TRUE 时执行。

所有信号输出都设置为零。

### 错误信息

通过参数分配工具执行参数检查。

## SFB 43 的参数

下表包含 SFB 43 “PULSEGEN” 的输入参数：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
INV	REAL	0	输入变量 模拟调节值分配给输入参数“输入变量”。		0.0
			<ul style="list-style-type: none"> <li>对于 <math>RATIOFAC &lt; 1</math> 的三步控制：</li> </ul>	-100/RATIOFAC 到 100 (%)	
			<ul style="list-style-type: none"> <li>对于 <math>RATIOFAC &gt; 1</math> 的三步控制：</li> </ul>	-100 到 100/ RATIOFAC (%)	
			<ul style="list-style-type: none"> <li>对于双极性两步控制：</li> <li>对于单极性两步控制：</li> </ul>	-100 到 100 (%) 到 100 (%)	
PER_TM	TIME	4	周期时间 为“周期时间”输入参数输入脉冲宽度调制的恒定周期。这与控制器的采样时间相对应。脉冲生成器的采样时间和控制器的采样时间之间的比率确定了脉冲宽度调制的精度。	$\geq 20 \times \text{SFB 43 的周期}$  (对应于采样时间 SFB 41)	T#1s
P_B_TM	TIME	8	最小脉冲/断线时间 可以在输入参数“最小脉冲或最小断线时间”处分配最小脉冲或最小断线时间。	$\geq \text{CYCLE}$	T#50 ms
RATIOFAC	REAL	12	比率因子 可以用输入参数“比率因子”更改负脉冲与正脉冲的持续时间比率。例如，在热处理中，这允许为加热和冷却补偿不同时间常数（例如，用电加热和水冷却进行处理）。	10.0	1.0
STEP3_ON	BOOL	16.0	启用三步控制 输入参数“启用三步控制”激活此模式。在三步控制中，两个输出信号都处于激活状态。		TRUE
ST2BI_ON	BOOL	16.1	启用双极性调节值范围的两步控制 利用输入参数“启用双极性调节值范围的两步控制”，可以在“双极性调节值范围的两步控制”和“单极性调节值范围的两步控制”这两种模式间进行选择。此处，STEP3_ON = FALSE。		FALSE
MAN_ON	BOOL	16.2	启用手动模式 通过设置输入参数“启用手动模式”，可以手动设置输出信号。		FALSE
POS_P_ON	BOOL	16.3	启用正模式 在带三步控制的手动模式中，可以在输入参数“启用正脉冲”处操作输出信号 QPOS_P。在带有两步控制的手动模式中，总是将 QNEG_P 设置与 QPOS_P 相反。		FALSE

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
NEG_P_ON	BOOL	16.4	开启负脉冲 在带三步控制的手动模式中，可以在输入参数“启用正脉冲”处操作输出信号 QPOS_P。在带有两步控制的手动模式中，总是将 QNEG_P 设置与 QPOS_P 相反。		FALSE
SYN_ON	BOOL	16.5	启用同步 通过设置输入参数“启用同步”，可与更新输入变量 INV 的块自动同步。这样可确保更改的输入变量作为脉冲尽快输出。	条件： PER_TM = SFB 41 的采样时间	TRUE
COM_RST	BOOL	16.6	完全重启动 该块有一个初始化例程，在输入 COM_RST 置位时进行处理。	TRUE: 重启动 FALSE: 控制器操作	FALSE
CYCLE	TIME	18	采样时间 块调用之间的时间必须恒定。输入“采样时间”指定块调用之间的时间。	>= 20 ms	T#10 ms

### 说明

输入参数的值在块中不受限制。没有参数检查。

下表包含了 SFB 43 “PULSEGEN” 的输出参数的说明：

参数	数据类型	地址（背景数据块）	说明	值范围	缺省
QPOS_P	BOOL	22.0	输出正脉冲 输出脉冲时，设置输出参数“输出正脉冲”。在三步控制中，这始终是正脉冲。对于两步控制，QNEG_P 始终设置与 QPOS_P 相反。		FALSE
QNEG_P	BOOL	22.1	输出负脉冲 输出脉冲时，设置输出参数“输出负脉冲”。在三步控制中，这始终是负脉冲。对于两步控制，QNEG_P 始终设置与 QPOS_P 相反。		FALSE

## 7.6 诊断/错误处理

### 基本信息

通过参数分配工具执行参数检查。如果在用户程序中更改了参数，则不检查“无意义”的参数。在这种情况下，您将不会接收到错误信息。

## 7.7 安装实例

### 使用实例

可在文档附带的 CD-ROM 中找到实例（程序和说明）。也可从 Internet 下载。项目由具有不同复杂度和针对性的多个带注释的 S7 程序组成。

CD 上的 Readme.wri 说明了安装实例的方法。安装完成后，可以在目录 `...\STEP7\EXAMPLES\ZDI26_04_TF____31xC_PID` 下找到实例。

# 索引

## 数字

- 2 线操作, 251, 261
- 3964 程序
  - 接收缓冲区, 318
- 3964(R) 程序
  - 参数, 264
  - 带缺省值, 264
  - 可分配, 264
- 3964(R)程序
  - 程序错误, 313
  - 初始化冲突, 313
  - 处理损坏的数据, 312
  - 点对点通讯, 307
  - 发送, 315
  - 发送数据, 309
  - 规范, 333
  - 接收, 316
  - 接收数据, 310
  - 控制字符, 307
  - 块检查字符, 308
  - 启动, 314
  - 优先级, 307
- 4 线操作, 251, 261

## A

- ACCEL, 72
- ASCII 驱动程序
  - 参数, 257
  - 点对点通讯, 297
  - 发送数据, 298
  - 规范, 332
  - 接收缓冲区, 306
  - 接收数据, 300
  - 数据流控制, 306
- 安全概念
  - 定位, 93
- 安全规则
  - 定位, 23, 93
- 安全原则
  - 定位, 23

## B

- BCC(块检查字符), 308
- BIE, 76, 144, 223
- 半双工, 261
- 半双工操作
  - 点对点通讯, 250
- 背景 DB, 40
- 背景数据块, 112, 268
- 背景数据块, 访问, 175
- 本手册的有效性, 3
- 比较器
  - 控制输出, 211
- 比较值, 170
  - 计数, 196
- 比较值时刻的脉冲, 196
- 比率控制, 356
- 编码器
  - 技术规范, 81, 149
  - 可连接的, 233
- 编码器参数, 38, 70, 110, 140
- 编码器每转增量, 38, 70, 88, 110, 140, 156
- 表
  - SFB 60 SEND\_PTP, 348
  - SFB 61 RCV\_PTP, 348
  - SFB 62 RES\_RCVB, 349
  - SFB 63 SEND\_RK, 349
  - SFB 64 FETCH\_RK, 350
  - SFB 65 SERVE\_RK, 351

## C

- CHGDIFF\_M, 72, 141
- CHGDIFF\_P, 72, 141
- CONT\_S, 366
- CONV\_DIR, 43
- CONV\_EN, 43
- CPU
  - 支持的功能, 15
- CPU 314C-2 DP/PtP 前连接器, 95
- CPU 的操作模式, 159
- CPU 的特性, 159
- CPU 和功能概述, 15

- CPU 通道, 编号, 159
  - CUTOFFDIFF\_M, 72, 141
  - CUTOFFDIFF\_P, 72, 141
  - 参考点, 52, 122
  - 参考点逼近
    - 步骤, 56, 126
    - 使用模拟输出定位, 52
    - 使用数字输出定位, 122
  - 参考点开关, 52, 122
  - 参考点开关的参考点位置, 37, 88, 107, 109, 156
  - 参考点坐标, 37, 88, 107, 109
  - 参考转速
    - 调整, 71
  - 参数
    - SFB 41 CONT\_C, 363
    - SFB 42 CONT\_S, 369
    - SFB 43 PULSEGEN, 378
    - SFB 46 DIGITAL, 157
    - SFB 47 COUNT, 241
    - SFB 48 FREQUENC, 243
    - SFB 49 PULSE, 245
    - SFB 60 SEND\_PTP, 348
    - SFB 61 RCV\_PTP, 348
    - SFB 62 RES\_RCVB, 349
    - SFB 63 SEND\_RK, 349
    - SFB 64 FETCH\_RK, 350
    - SFB 65 SERVE\_RK, 351
    - SFB ANALOG 的背景数据块, 89
    - SFB DIGITAL 的背景数据块, 157
  - 参数分配, 169, 358
    - 点对点连接, 254
    - 定位功能, 29, 100
    - 间接, 291
    - 直接, 291
  - 参数分配窗口, 358
    - 点对点通讯, 255
    - 定位, 30, 100
    - 模块参数概述, 87, 155
  - 参数分配数据
    - 3964(R) 程序, 264
    - ASCII 驱动程序, 257
    - RK 512, 267
  - 参数组态, 101
  - 操作模式
    - 参考点逼近, 52, 122
    - 长度测量, 68, 138
    - 点动模式, 50, 120
    - 绝对增量逼近, 61, 131
    - 相对增量逼近, 58, 128
  - 操作模式错误, 75, 143
  - 测量错误
    - 频率测量, 231
    - 频率计数, 203
  - 测量值的输出, 172
  - 长度测量, 37, 88, 107, 108, 156
    - 使用数字输出定位, 138
    - 用模拟输出定位, 68
  - 程序结构, 176, 268
  - 重新尝试传输的次数, 264
  - 重新尝试连接的次数, 264
  - 初始化冲突, 313
  - 处理器间通讯标志, 320
    - 实例应用, 289
    - 使用, 288
  - 传输时间
    - 点对点通讯, 335
  - 串级控制, 355
  - 串行电缆
    - 连接, 253
  - 串行数据传输
    - 点对点通讯, 250
  - 创建控制, 354
  - 从站
    - 点对点通讯, 262
  - 错误
    - 定位, 293
    - 诊断, 293
  - 错误编号
    - JOB\_STAT SFB 参数, 86, 154
    - STATUS SFB 参数, 83
  - 错误编号:, 151
  - 错误处理, 75, 143, 223
    - 控制, 380
  - 错误处理和中断, 293
  - 错误类型, 75, 143
  - 错误列表
    - JOB\_STAT, 235
  - 错误列表,实例, 83, 151
  - 错误判断, 76, 144
  - 错误诊断
    - 点对点通讯, 293
- ## D
- DECEL, 72
  - DIGITAL, 117
  - 单重判断, 233
  - 当前操作数
    - 符号寻址, 291
  - 点动模式
    - 使用模拟输出定位, 50
    - 使用数字输出定位, 120
  - 点对点
    - 点对点通讯, 262

电机保护  
  定位, 23, 93  
 电缆  
  点对点通讯, 336  
 电缆屏蔽, 253  
  点对点通讯, 336  
 定位  
  参数类型, 29, 100  
  功能范围, 20  
  规范, 19  
  设计和组件, 21  
  使用模拟输出, 概述, 17  
  使用数字输出, 概述, 18  
 动力装置  
  连接到数字输出, 98  
 动力装置连接  
  使用数字输出定位, 98  
 对问题的回答, 6  
 多点  
  点对点通讯, 262

## E

ERR, 76, 144  
 ERROR, 75, 143  
 ERR-字, 结构, 86, 154

## F

FETCH 消息帧, 319  
 FETCH\_RK, 281  
 发送数据  
  3964(R)程序, 309  
  ASCII 驱动程序, 298  
  RK 512, 322  
 发送暂停, 259  
 反向  
  频率计数, 203  
 反转计数方向, 170, 172  
 方向信号, 43  
 防止覆盖  
  3964(R) 程序, 266  
 防止覆盖, 260  
 非对称输出信号, 81, 149  
 符号寻址当前操作数, 291

## G

概述  
  控制, 353  
 工作范围, 36, 37, 44, 45, 88, 107, 109, 115, 156  
  监视, 76, 86, 144  
  监视, 154  
 工作范围监视, 37  
 功能  
  计数, 177  
  脉冲宽度调制, 213  
  频率计数, 202  
 功能块  
  计数, 192  
  脉冲宽度调制, 218  
  频率测量, 209  
 固定的消息帧长度, 298, 302  
 关断差程, 42, 47, 113, 117  
 关断点, 42  
 关断位置, 113  
 关断延迟, 33  
  定位, 44  
 规范, 331  
  计数、频率测量、脉冲宽度调制, 231  
 过零点, 179

## H

HW 门, 170, 172, 174  
  计数, 194  
  脉冲宽度调制, 219  
  频率测量, 210  
 HW 门的过滤器频率, 174  
 换向点, 42, 113  
 混合控制, 355  
 获取数据  
  RK 512, 325

## I

I/O 访问, 直接, 176

## J

JOB\_ERR, 75, 143  
 JOB\_ID  
  计数, 189  
  脉冲宽度调制, 216  
  频率计数, 207  
 JOB\_STAT SFB 参数  
  错误编号, 86, 154

JOB\_VAL 值范围  
  计数, 190  
  脉冲宽度调制, 217  
  频率计数, 208  
奇偶校验, 257, 264  
基本参数, 31, 101, 169, 237  
  点对点通讯, 256  
基本功能, 353  
集成控制, 353  
集成在应用程序中  
  点对点连接, 268  
计数  
  功能, 177  
  功能范围, 160  
  功能块, 192  
  规范, 231  
  连接组件, 167  
  门功能, 194  
  术语, 178  
  针脚分配, 163  
计数方向, 38, 88, 110, 156  
  调整, 71  
  选择, 140  
计数模式, 177  
计数频率, 159  
计数器应用  
  组件, 161  
技术功能  
  支持 CPU, 15  
技术规范  
  增量编码器, 81, 149  
技术支持, 6  
间接参数分配, 291  
监视  
  工作范围, 107, 109, 156  
  目标逼近, 155  
  目标范围, 155  
  缺少脉冲(零标记), 156  
  实际值, 155  
  行程范围, 107, 109, 156  
监视功能, 45  
  使用数字输出定位, 115  
监视时间  
  选择, 140  
监视时间, 155  
  调整, 71  
检查监视时间  
  核对清单, 73, 142  
接口 X27 (RS 422/485), 249  
接口硬件, 292  
接收缓冲区, 260, 306  
  点对点连接, 318

接收数据  
  3964(R)程序, 310  
  ASCII 驱动程序, 300  
接收帧的消息结束识别, 259  
接通延迟, 174  
  脉冲宽度调制, 221  
接线  
  安全规则, 93, 162  
  定位, 23  
  控制, 357  
  前连接器, 252  
接线连接  
  定位, 24, 94  
结束标准, 298, 300  
  固定的消息帧长度, 302  
  文本结束字符, 304  
  在字符延迟时间结束时, 300  
结束值/开始值, 170  
紧急切断开关  
  定位, 23  
紧急停车开关  
  定位, 93  
绝对增量逼近  
  使用数字输出定位, 131  
  用模拟输出定位, 61

## K

开关控制器, 355  
控制  
  集成控制, 353  
  使用 SFB 42, 366  
  用 SFB 41 连续控制, 360  
控制模式, 32, 34, 87, 155  
  选择, 140  
控制器  
  开关控制器, 355  
  连续控制器, 355  
控制器选择, 354  
控制输出  
  计数, 197  
  脉冲宽度调制, 222  
  频率计数, 211  
控制系统分析, 354  
控制字符  
  3964(R)程序, 307  
块检验和  
  点对点通讯, 308



**L**

- 连接电缆
  - 安全规则, 162
  - 点对点通讯, 336
  - 点对点通讯, 336
  - 控制, 357
- 连接器 X1, 26
  - CPU 312C, 164
  - CPU 313C-2 DP/PtP, 165
- 连接器 X2, 26
- 连接组件
  - 定位, 28, 97
  - 计数、频率计数、脉冲宽度调制, 167
- 连续的 FETCH 消息帧, 327
- 连续的 SEND 消息帧, 324
- 连续的消息帧, 319
- 连续计数
  - 说明, 179
- 连续控制
  - SFB 41, 360
- 连续控制器, 355
- 两步控制, 356
- 零标记信号, 53, 123

**M**

- 脉冲, 150
- 脉冲参数
  - 脉冲宽度调制, 220
- 脉冲持续时间, 170
- 脉冲宽度调制
  - 功能范围, 160
  - 规范, 232
  - 连接组件, 167
  - 针脚分配, 163
- 门功能, 170
  - 计数, 194
  - 脉冲宽度调制, 219
  - 频率测量, 210
- 门控制
  - 计数, 195
  - 脉冲宽度调制, 219
- 明码性, 300
- 模块参数, 31, 237
  - 点对点连接, 254
  - 定位, 29, 100
  - 计数, 237
  - 计数功能, 170
  - 脉冲宽度调制, 174, 240
  - 频率测量, 172
  - 频率计数, 239
  - 通过参数分配窗口, 87, 155

- 目标逼近, 32, 45, 46, 87, 115, 116, 155
  - 工作范围, 88
  - 监视, 76, 86, 144
  - 监视, 154
  - 目标逼近, 32, 34, 87
  - 目标范围, 32, 34, 87
  - 缺少脉冲(零标记), 88
  - 实际值, 32, 34, 87
  - 行程范围, 88
- 目标逼近监视, 106
- 目标范围, 32, 42, 45, 87, 105, 113, 115, 155
  - 监视, 76, 86, 144
  - 监视, 154
- 目标范围监视, 106

**N**

- 内部门
  - 计数, 194
  - 脉冲宽度调制, 219
  - 频率测量, 210

**P**

- PULSEGEN, 371
- 频率测量
  - 规范, 231
  - 门功能, 210
  - 针脚分配, 163
- 频率测量步骤, 202
- 频率范围, 202
  - 频率测量, 231
- 频率计数
  - 功能范围, 160
  - 连接组件, 167
- 平均频率, 203
- 屏蔽
  - 安全规则, 162
  - 定位, 24, 94
- 屏蔽端接元件, 162
  - 定位, 24, 94
  - 控制, 357
- 屏蔽连接元件, 252

**Q**

- 启用动力装置, 43
- 起始位, 257, 264
- 前连接器
  - CPU 314C-2 DP/PtP, 25
  - 接线, 252

- 前连接器 X2
    - CPU 313C, 165
    - CPU 314C-2 DP/PtP, 95
    - CPU 314C-2 DP/PtP, 166
  - 切断延迟, 87
  - 驱动器参数, 32, 102
  - 驱动器的控制模式, 102
  - 取消
    - 使用数字输出定位, 116
  - 取消定位, 46
  - 取消激活
    - 使用数字输出定位, 116
  - 取消激活定位, 46
  - 取消门功能
    - 计数, 194
  - 全双工, 261
  - 全双工操作
    - 点对点通讯, 250
  - 缺少脉冲(零标记), 38, 45, 88, 110, 115, 156
    - 监视, 76, 86, 144
    - 监视, 154
  - 缺少脉冲(零标记)监视, 38, 110
  - 缺省, 261, 267
    - 接收线路, 261, 267
  - 确认延迟, 264
- R**
- RCV\_PTP, 272
  - RES\_RCVB, 274
  - RK 512
    - CPU 请求, 329
    - 伙伴请求, 330
  - RK 512 计算机连接
    - 参数, 267
    - 点对点通讯, 319
    - 发送数据, 322
    - 获取数据, 325
    - 响应消息帧, 319, 321
    - 指令帧, 319, 320
  - RS 422/485
    - 点对点连接, 249
  - RS422, 251, 261
  - RS422 操作
    - 点对点通讯, 297
  - RS485, 251, 261
  - RS485 操作
    - 点对点通讯, 297
  - 认证, 331
  - 蠕行速度
    - 调整, 71
  - 蠕行速度/参考转速, 33, 87
- 软件门
- 计数, 194
  - 脉冲宽度调制, 219
  - 频率测量, 210
- 软件限位开关结束, 36, 88, 107, 108, 156
- 软件限位开关开始, 36, 88, 107, 108, 156
- S**
- SEND 消息帧, 319
  - SEND\_PTP, 269
  - SEND\_RK, 277
  - SERVE\_RK, 285
  - SET\_DO
    - 计数, 197
    - 脉冲宽度调制, 222
    - 频率计数, 211
  - SFB
    - 错误消息, 75, 143, 223
  - SFB 41, 363
  - SFB 41 CONT\_C
    - 方框图, 362
  - SFB 42, 369
  - SFB 42 CONT\_S
    - 方框图, 368
  - SFB 43 PULSEGEN, 371
    - 参数, 378
    - 操作模式, 374
    - 初始化, 377
    - 错误信息, 377
    - 两步控制, 376
    - 三步控制, 374
    - 自动同步, 373
  - SFB 44, 40
    - 基本参数, 47
  - SFB 44 的背景数据块
    - 参数, 89
  - SFB 46, 112
    - 基本参数, 117
  - SFB 46 的背景数据块
    - 参数, 157
  - SFB 47, 186
  - SFB 48, 204
  - SFB 49, 214
  - SFB ANALOG, 40
    - 基本参数, 47
  - SFB ANALOG 的背景数据块
    - 参数, 89
  - SFB CONT\_C, 363
  - SFB CONT\_S, 369
  - SFB COUNT, 186

- SFB DIGITAL, 112
  - 基本参数, 117
- SFB DIGITAL 的背景数据块
  - 参数, 157
- SFB FETCH\_RK, 281
- SFB FREQUENC, 204
- SFB PULSE, 214
- SFB RCV\_PTP, 272
- SFB RES\_RCVB, 274
- SFB SEND\_PTP, 269
- SFB SEND\_RK, 277
- SFB 参数
  - 点对点连接, 254
  - 定位, 29, 100
- SFB\_SERVE\_RK, 285
- SLSE, 36, 107
- SLSS, 36, 107
- STATUS
  - 编号方案, 340
- STATUS SFB 参数
  - 错误编号, 83
- STS\_CMP(状态位), 197
- SW 门
  - 计数, 194
  - 脉冲宽度调制, 219
  - 频率测量, 210
- 三步控制, 356
- 删除剩余行程
  - 使用模拟输出定位, 67
  - 使用数字输出定位, 137
- 上限, 172
- 上溢, 179
- 设定值控制, 355
- 设置参考点
  - 使用模拟输出定位, 64
  - 使用数字输出定位, 134
- 设置计数值, 178
- 设置装载值, 178
- 生成脉冲, 371
  - 用 SFB 43 PULSEGEN, 371
- 时基, 170, 174
  - 脉冲宽度调制, 220
- 时间监视, 32, 87, 105
  - 缺少结束代码, 259
- 实际值, 45, 115, 155
  - 监视, 76, 86, 144
  - 监视, 154
- 实际值监视, 105
- 实例文件
  - 点对点通讯, 296
  - 计数、频率测量、脉冲宽度调制, 230
  - 控制, 380
  - 使用模拟输出定位, 80
  - 使用数字输出定位, 148
- 使用 SFB 42, 366
- 使用参数分配窗口, 168
- 使用数字输出定位
  - 引脚分配, 95
- 使用数字输出定位
  - 断路器的操作模式, 99
- 事件编号, 83, 151
- 事件类别, 83, 151
- 输出
  - 计数, 196
  - 脉冲宽度调制, 222
  - 频率测量, 210
- 输出反应, 172
- 输出格式, 174
  - 脉冲宽度调制, 220
- 输出特性, 170
- 输出值
  - 脉冲宽度调制, 220
- 输入
  - 计数器, 193
  - 频率测量, 210
- 输入方向/B
  - 计数, 193
  - 频率测量, 210
- 输入脉冲/A
  - 计数, 193
  - 频率测量, 210
- 输入数据的分配, 170
- 输入锁存器
  - 计数, 193
- 术语
  - 计数, 177, 178
- 数据操作数
  - 寻址, 290
- 数据操作数的寻址, 290
- 数据传输格式
  - 点对点通讯, 251
- 数据流控制
  - 点对点通讯, 306
- 数据位, 257, 264
- 数据一致性, 271, 273, 279, 283
- 数字输出上的断路器
  - 工作原理, 99
- 数字输入的故障
  - 定位, 28, 97
- 双向数据交换, 250
- 双重判断, 233
- 四重判断, 233
- 锁存器功能, 193

**T**

- 调试接口硬件, 292
- 调整参数
  - 安全规则, 70, 140
- 停止位, 257, 264
- 通讯伙伴
  - 点对点通讯, 248

**W**

- 外部错误, 76, 144
  - 结构, 86, 154
- 伪全双工模式
  - 点对点通讯, 328
- 文本结束字符, 259, 298
- 握手
  - 点对点通讯, 306

**X**

- X27 (RS 422/485) 接口, 336
  - 规范, 331
- X27 接口
  - 属性, 249
- XOFF 字符, 257
- XON 字符, 257
- XON/XOFF, 257
- 系统错误, 76, 144
- 系统功能块
  - SFB 41 CONT\_C, 363
  - SFB 42 CONT\_S, 369
  - 错误消息, 75, 143
- 系统功能块 (SFB) 的错误消息
  - 点对点连接, 293
- 系统功能块(SFB)的错误消息
  - 计数、频率计数、脉冲宽度调制, 223
- 下限, 172
- 下溢, 179
- 线性轴, 35, 107
- 相对增量逼近
  - 使用模拟输出定位, 58
  - 使用数字输出定位, 128
- 响应消息帧, 319
- 消息帧长度, 259
- 消息帧头
  - RK 512 指令帧的结构, 320
- 信号判断, 170, 172
  - 非对称输出, 81, 149
- 行程范围, 36, 37, 45, 88, 107, 115, 156
  - 监视, 76, 86, 144
  - 监视, 154

- 行程范围监视, 37
- 旋转轴, 35, 107
- 旋转轴终点, 35, 36, 88, 107, 108, 156

**Y**

- 一次计数
  - 说明, 180
- 已缓冲的接收消息帧
  - 3964(R) 程序, 266
- 已缓冲的接收消息帧数, 260
- 异步数据传输
  - 点对点通讯, 250
- 应用, 353
  - 点对点通讯, 247
- 硬件门
  - 计数, 194
  - 脉冲宽度调制, 219
  - 频率测量, 210
- 硬件限位开关
  - 定位, 23, 93
- 硬件中断, 223
  - HW 门打开, 170, 174
  - HW 门关闭, 170
  - 测量结束, 172
  - 达到比较器时, 170
  - 打开 HW 门, 172
  - 丢失, 226
  - 关闭 HW 门, 172
  - 计数, 201
  - 计数跳沿, 170
  - 计数跳沿, 170
  - 脉冲宽度调制, 222
  - 判断, 227
  - 频率测量, 212
  - 上溢, 170, 172
  - 使用, 227
  - 下溢, 170, 172
- 用 SFB 41 控制, 360
- 用 SFB 43, 371
- 优先级, 264
- 有两个连接器的 CPU 的前连接器, 163
- 运行顺序
  - 定位, 42
  - 使用数字输出定位, 113

## Z

在线帮助, 255, 358  
 参数分配窗口, 31, 101, 169  
 增量  
 定义, 82, 150, 233  
 增量编码器  
 技术规范, 81, 149  
 可连接的, 233  
 增量编码器的接线图, 82, 150, 234  
 针脚分配, 26  
 计数、频率测量、脉冲宽度调制, 163  
 使用数字输出定位, 95  
 诊断  
 参数, 39, 111  
 控制, 380  
 诊断中断, 78, 146  
 激活, 39, 111  
 判断, 79, 147, 226  
 使用, 225  
 直接参数分配, 291  
 实例, 291  
 直接频率, 203  
 指令帧, 319  
 滞后, 170  
 滞后、计数模式, 198  
 中断, 223  
 点对点通讯, 293  
 中断门功能  
 计数, 194  
 中断选择, 31, 87, 101, 155, 169, 237  
 终止运行  
 定位, 46  
 使用数字输出定位, 116  
 周期, 174  
 脉冲宽度调制, 221  
 周期测量, 177  
 周期性计数  
 说明, 183  
 轴参数, 35, 107  
 轴类型, 35, 88, 107, 156  
 主计数方向, 170, 178  
 Down(下), 178  
 Up(上), 178  
 主站  
 点对点通讯, 262  
 转换差程, 42, 47, 113, 117  
 状态位 STS\_CMP, 197  
 字符延迟, 264  
 字符延迟时间, 298, 300  
 字符延迟时间(CDT), 251, 259  
 字符帧  
 点对点通讯, 251

## 组件

点对点连接, 248  
 计数器应用, 161  
 组态诊断中断  
 点对点通讯, 295  
 最大计数频率, 172  
 CPU, 177  
 最大计数频率伴随信号, 155  
 最大计数频率位置反馈, 155  
 最大频率  
 计数信号/HW 门, 170  
 锁存器, 170  
 最大频率  
 伴随信号, 106  
 伴随信号, 33, 87  
 位置反馈, 106  
 位置反馈, 33, 87  
 最高转速, 32, 87  
 计算, 70  
 最小 CPU 周期数  
 点对点通讯, 334  
 最小脉冲持续时间, 174  
 最小脉冲宽度  
 脉冲宽度调制, 221  
 作业  
 删除剩余行程, 67, 137  
 设置参考点, 64, 134  
 作业错误, 75, 143, 223  
 作业号  
 计数, 189  
 脉冲宽度调制, 216  
 频率计数, 207  
 作业接口  
 计数, 189  
 脉冲宽度调制, 216  
 频率计数, 207

