

功能手册 版本2012年2月

SINAMICS/SIMOTION

DCC标准块描述

运动控制

SIEMENS

SIEMENS

SINAMICS/SIMOTION

DCC 标准块描述

功能手册

适用于

驱动器

SINAMICS

控制系统

SIMOTION

产品

固件版本

2. 5. x/2. 6. x/4. 3. x/4. 4. x/4. 5. x

4. 3

DCC 2. 2

前言

简介

Arithmetic

Logic

Conversion

System

Technology

Closed-loop control

附录

附录

1

2

3

4

5

6

7

A

B

6SL3097-4AQ00-0RP2

2012 年 02 月

法律资讯

警告提示系统

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

 危险
表示如果不采取相应的小心措施， 将会 导致死亡或者严重的人身伤害。
 警告
表示如果不采取相应的小心措施， 可能 导致死亡或者严重的人身伤害。
 小心
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。
小心
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。
注意
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

本文件所属的产品/系统只允许由符合各项工作要求的**合格人员**进行操作。其操作必须遵照各自附带的文件说明，特别是其中的安全及警告提示。由于具备相关培训及经验，合格人员可以察觉本产品/系统的风险，并避免可能的危险。

Siemens 产品

请注意下列说明：

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

商标

所有带有标记符号 ® 的都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，这是出于保护所有权利的目的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

SIMOTION 文献

SIMOTION 文献一览表在参考文献列表中单独说明。

此文献以电子件形式随 SIMOTION SCOUT 发货。包含 10 个文献包。

SIMOTION 文献包括 9 个文献包，共约 80 个 SIMOTION 文档和系统（如 SINAMICS）相关文档。

SIMOTION V4.3 提供如下文献包：

- SIMOTION 工程系统
- SIMOTION 系统与功能描述
- SIMOTION 维修与诊断
- SIMOTION IT
- SIMOTION 编程
- SIMOTION 编程 参考文档
- SIMOTION C
- SIMOTION P
- SIMOTION D
- SIMOTION 其他文献

SINAMICS 文献

SINAMICS 文献分为两个部分：

- 一般文献 / 目录
- 制造商 / 维修文献

可从以下网址获得该资料现有语言版本的总览：

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

选择菜单项 “Support” > “Technical documentation” > “Overview of publications”。

可从以下网址获得 DOConCD (DOConWEB) 的网络版：

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

可从以下网址获得培训课程以及 FAQ (常见问题) 的相关信息：

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

选择菜单项 “Support”。

DCC 编辑器的更多文献

- SINAMICS/SIMOTION DCC 编辑器描述

其它信息

点击下面链接可获取下列主题的相关信息：

- 订购文档 / 查看文档一览表
- 进入文档的其它下载链接
- 在线使用文档 (查找搜索手册 / 信息)

<http://www.siemens.com/motioncontrol/docu>

如果您对技术文档有疑问 (例如：建议或修改)，请发送一份电子邮件到下列地址：

docu.motioncontrol@siemens.com

My Documentation Manager

点击下面链接，可以在西门子文档内容的基础上自行编译文档并创建自己的机床文档：

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

培训

点击下面链接，可获取关于 SITRAIN - 西门子自动化产品、系统以及解决方案的培训课程的相关信息：

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

常见问题

常见问题（FAQ）请参见 SIMOTION SCOUT 包装中的“SIMOTION 应用与工具”，或访问下面链接中产品支持下的“支持”页：

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

技术支持

各个国家的技术咨询电话请访问下列网址，而后进入联系人页：

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

目录

1	前言.....	15
1.1	Drive Control Chart (DCC: 驱动控制图) 入门.....	15
1.2	库.....	17
1.2.1	兼容性.....	18
1.3	功能块汇总表.....	19
1.4	多个块连接.....	21
1.5	字节排序.....	22
1.6	不同数据类型的直接互连.....	22
1.7	功能块初始化.....	23
1.8	在简单配置中实现复杂功能.....	24
1.8.1	导入示例配置.....	25
2	Arithmetic.....	29
2.1	ACOS 反余弦功能块.....	29
2.2	ADD 加法器 (REAL型).....	31
2.3	ADD_D 加法器 (DOUBLE INTEGER型).....	32
2.4	ADD_I 加法器 (INTEGER型).....	33
2.5	ADD_M 在正确的轴周期中增加值的模加法器.....	34
2.6	ASIN 反正弦功能块.....	36
2.7	ATAN 反正切功能块.....	38
2.8	AVA 支持符号评估的绝对值发生器.....	40
2.9	AVA_D 绝对值发生器 (DOUBLE INTEGER型).....	42
2.10	COS 余弦功能块.....	44
2.11	DIV 除法器 (REAL型).....	46
2.12	DIV_D 除法器 (DOUBLE INTEGER型).....	48
2.13	DIV_I 除法器 (INTEGER型).....	50
2.14	MAS 最大值评估器.....	52
2.15	MIS 最小值评估器.....	53
2.16	MUL 乘法器 (REAL型).....	54
2.17	MUL_D 乘法器 (DOUBLE INTEGER型).....	55
2.18	MUL_I 乘法器 (INTEGER型).....	56
2.19	PLI20 多段线, 20个断点.....	57
2.20	SII 反相器.....	61
2.21	SIN 正弦功能块.....	63
2.22	SQR 平方根提取器.....	65
2.23	SUB 减法器 (REAL型).....	66
2.24	SUB_D 减法器 (DOUBLE INTEGER型).....	67
2.25	SUB_I 减法器 (INTEGER型).....	68
2.26	TAN 正切.....	69
3	Logic.....	71
3.1	AND 逻辑AND运算 (BOOL型).....	71

3.2	AND_W	逻辑AND运算 (WORD型)	73
3.3	BF	闪烁功能块 (BOOL型)	75
3.4	BF_W	状态字闪烁功能块 (BOOL型)	77
3.5	BSW	BOOL型二位切换开关	79
3.6	CNM	可控数字存储器 (REAL型)	81
3.7	CNM_D	可控数字存储器 (DOUBLE INTEGER型)	83
3.8	CNM_I	可控数字存储器 (INTEGER型)	85
3.9	CTR	计数器 (BOOL型)	87
3.10	DFR	复位优先D型触发器 (BOOL型)	90
3.11	DFR_W	复位优先D型触发器 (WORD型)	92
3.12	DLB	延迟元素 (REAL型)	94
3.13	DX8	多路输出选择器, 8个输出, 可级联 (REAL型)	96
3.14	DX8_D	多路输出选择器, 8个输出, 可级联 (DOUBLE INTEGER型)	98
3.15	DX8_I	多路输出选择器, 8个输出, 可级联 (INTEGER型)	100
3.16	ETE	沿评估器 (BOOL型)	102
3.17	LVM	磁滞双面限制监视器 (BOOL型)	104
3.18	MFP	脉冲发生器 (BOOL型)	106
3.19	MUX8	多路复用器, 可级联 (REAL型)	108
3.20	MUX8_D	多路复用器, 可级联 (DOUBLE INTEGER型)	111
3.21	MUX8_I	多路复用器, 可级联 (INTEGER型)	114
3.22	NAND	逻辑AND运算 (BOOL型)	117
3.23	NCM	数字比较器 (REAL型)	119
3.24	NCM_D	数字比较器 (DOUBLE_INTEGER型)	120
3.25	NCM_I	数字比较器 (INTEGER型)	121
3.26	NOP1	预留功能块 (REAL型)	122
3.27	NOP1_B	预留功能块 (BOOL型)	123
3.28	NOP1_D	预留功能块 (DOUBLE INTEGER型)	124
3.29	NOP1_I	预留功能块 (INT型)	125
3.30	NOP8	预留功能块 (REAL型)	126
3.31	NOP8_B	预留功能块 (BOOL型)	128
3.32	NOP8_D	预留功能块 (DOUBLE INTEGER型)	130
3.33	NOP8_I	预留功能块 (INTEGERT型)	132
3.34	NOR	逻辑OR运算 (BOOL型)	134
3.35	NOT	反相器 (BOOL型)	136
3.36	NOT_W	状态字反相器 (WORD型)	137
3.37	NSW	数字切换开关 (REAL型)	139
3.38	NSW_D	数字切换开关 (DOUBLE INTEGER型)	141
3.39	NSW_I	数字切换开关 (INTEGER型)	143
3.40	OR	逻辑OR运算 (BOOL型)	145
3.41	OR_W	逻辑OR运算 (WORD型)	147
3.42	PCL	脉冲缩短设备 (BOOL型)	149
3.43	PDE	接通延迟设备 (BOOL型)	151
3.44	PDF	断开延迟设备 (BOOL型)	153
3.45	PST	脉冲展宽功能块 (BOOL型)	155
3.46	RSR	复位优先的RS触发器 (BOOL型)	157
3.47	RSS	设定优先的RS触发器 (BOOL型)	159

3.48	SH	移位功能块 (WORD型)	161
3.49	SH_DW	移位功能块 (DWORD型)	163
3.50	TRK	校正/存储元件 (REAL型)	165
3.51	TRK_D	校正/存储元件 (DOUBLE INTEGER型)	167
3.52	XOR	逻辑XOR运算 (BOOL型)	169
3.53	XOR_W	逻辑XOR运算 (WORD型)	170
4	Conversion		173
4.1	BY_B	转换器状态字节转换为8个二进制变量	173
4.2	BY_W	状态字节到状态字的转换器	176
4.3	B_BY	8个二进制变量到状态字节的转换器	178
4.4	B_DW	32个二进制变量到状态双字的转换器	180
4.5	B_W	16个二进制变量到状态字的转换器	183
4.6	DW_B	状态双字到32个二进制变量的转换器	186
4.7	DW_R	接受位串为真值	189
4.8	DW_W	状态双字到状态字的转换器	190
4.9	D_I	DOUBLE INTEGER到INTEGER的转换器	191
4.10	D_R	DOUBLE INTEGER到REAL的转换器	192
4.11	D_SI	DOUBLE INTEGER到SHORT INTEGER的转换器	193
4.12	D_UI	DOUBLE INTEGER到UNSIGNED INTEGER的转换器	194
4.13	D_US	DOUBLE INTEGER到UNSIGNED SHORT INTEGER的转换器	195
4.14	I_D	INTEGER到DOUBLE INTEGER的转换器	196
4.15	I_R	INTEGER到REAL的转换器	197
4.16	I_SI	INTEGER到SHORT INTEGER的转换器	198
4.17	I_UD	INTEGER到UNSIGNED DOUBLE INTEGER的转换器	199
4.18	I_US	INTEGER到UNSIGNED SHORT INTEGER的转换器	200
4.19	LR_R	LONG REAL到REAL的转换器	201
4.20	N2_R	转换16位定点格式 (N2) 为REAL	202
4.21	N4_R	转换32位定点格式 (N4) 为REAL	203
4.22	R_D	REAL到DOUBLE INTEGER的转换器	204
4.23	R_DW	以DWORD进行的位串传递	205
4.24	R_I	REAL到INTEGER的转换器	206
4.25	R_LR	REAL到LONG REAL的转换器	207
4.26	R_N2	转换REAL为16位定点格式 (N2)	208
4.27	R_N4	转换REAL为32位定点格式 (N4)	209
4.28	R_SI	REAL到SHORT INTEGER的转换器	210
4.29	R_UD	REAL到UNSIGNED DOUBLE INTEGER的转换器	211
4.30	R_UI	REAL到UNSIGNED INTEGER的转换器	212
4.31	R_US	REAL到UNSIGNED SHORT INTEGER的转换器	213
4.32	SI_D	SHORT INTEGER到DOUBLE INTEGER的转换器	214
4.33	SI_I	SHORT INTEGER到INTEGER的转换器	215
4.34	SI_R	SHORT INTEGER到REAL的转换器	216
4.35	SI_UD	SHORT INTEGER到UNSIGNED DOUBLE INTEGER的转换器	217
4.36	SI_UI	SHORT INTEGER到UNSIGNED INTEGER的转换器	218
4.37	UD_I	UNSIGNED DOUBLE INTEGER到INTEGER的转换器	219

4.38	UD_R	UNSIGNED DOUBLE INTEGER到REAL的转换器.....	220
4.39	UD_SI	UNSIGNED DOUBLE INTEGER到SHORT INTEGER的转换器.....	221
4.40	UI_D	UNSIGNED INTEGER 到DOUBLE_INTEGER的转换器.....	222
4.41	UI_R	UNSIGNED INTEGER到REAL的转换器.....	223
4.42	UI_SI	UNSIGNED INTEGER到SHORT INTEGER的转换器.....	224
4.43	US_D	UNSIGNED SHORT INTEGER 到DOUBLE_INTEGER的转换器.....	225
4.44	US_I	UNSIGNED SHORT INTEGER 到INTEGER的转换器.....	226
4.45	US_R	UNSIGNED SHORT INTEGER到REAL的转换器.....	227
4.46	W_B	状态字到16个二进制变量的转换器.....	228
4.47	W_BY	状态字到状态字节的转换器.....	230
4.48	W_DW	状态字到状态双字的转换器.....	232
5	System.....		233
5.1	CTD	由内部时间戳确定时差.....	233
5.2	GTS	读出时间戳.....	235
5.3	RAA	复位所有消息.....	236
5.4	RDA	读出消息.....	237
5.5	RDAA	读出所有消息.....	239
5.6	RDP	读取驱动参数 (REAL型).....	240
5.7	RDP_D	读取驱动参数 (DOUBLE INTEGER型).....	242
5.8	RDP_I	读取驱动参数 (INTEGER型).....	244
5.9	RDP_UD	读取驱动参数 (UNSIGNED DOUBLE INTEGER型).....	246
5.10	RDP_UI	读取驱动参数 (UNSIGNED INTEGER型).....	248
5.11	RDP_US	读取驱动参数 (UNSIGNED SHORT INTEGER型).....	250
5.12	RMDP	从控制器中读取驱动参数.....	252
5.13	SAH	采样与保持 (REAL型).....	259
5.14	SAH_B	采样与保持 (BOOL型).....	262
5.15	SAH_BY	采样与保持 (BYTE型).....	265
5.16	SAH_D	采样与保持 (DOUBLE INTEGER型).....	268
5.17	SAH_I	采样与保持 (INTEGER型).....	271
5.18	SAV	数值缓冲 (REAL型).....	274
5.19	SAV_BY	数值缓冲 (BYTE型).....	277
5.20	SAV_D	数值缓冲 (DOUBLE INTEGER型).....	280
5.21	SAV_I	数值缓冲 (INTEGER型).....	283
5.22	SRA	触发/复位消息.....	286
5.23	STM	故障/告警触发器.....	288
5.24	WMDP	从控制器中写入驱动参数.....	291
5.25	WRP	写入驱动参数 (REAL型).....	297
5.26	WRP_D	写入驱动参数 (DOUBLE INTEGER型).....	299
5.27	WRP_I	写入驱动参数 (INTEGER型).....	301
5.28	WRP_UD	写入驱动参数 (UNSIGNED DOUBLE INTEGER型).....	303
5.29	WRP_UI	写入驱动参数 (UNSIGNED INTEGER型).....	305
5.30	WRP_US	写入驱动参数 (UNSIGNED SHORT INTEGER型).....	307
6	Technology.....		309
6.1	DCA	直径计算器.....	309

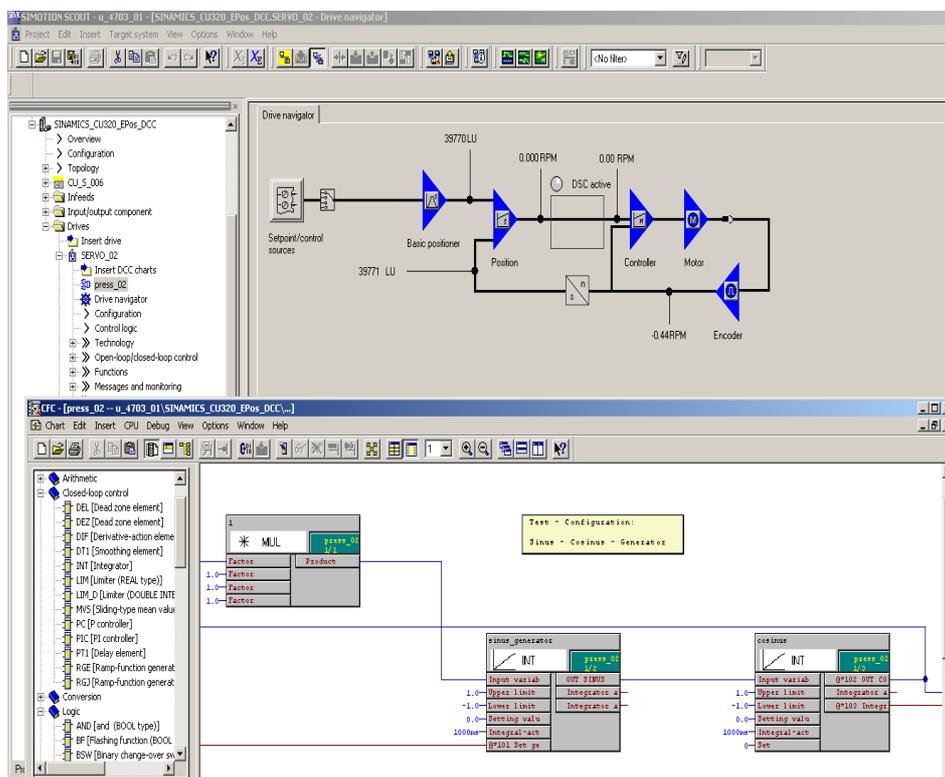
6.2	INCO	轴向轴机转动惯量.....	313
6.3	OCA	软件凸轮控制器.....	315
6.4	TTCU	绕组特性.....	317
6.5	WBG	抖动发生器.....	319
7	Closed-loop control.....		323
7.1	DEL	死区元素.....	323
7.2	DEZ	死区元素.....	326
7.3	DIF	微商作用元素.....	329
7.4	DT1	平滑滤波元素.....	332
7.5	INT	积分器.....	335
7.6	LIM	限制器 (REAL型).....	338
7.7	LIM_D	限制器 (DOUBLE INTEGER型).....	340
7.8	MVS	滑动平均值发生器.....	342
7.9	PC	P作用控制器.....	344
7.10	PIC	PI控制器.....	347
7.11	PT1	延迟元素.....	356
7.12	RGE	斜坡函数发生器.....	359
7.13	RGJ	带加加速度限制的斜坡函数发生器.....	367
A	附录.....		377
A.1	数据类型.....		377
A.2	误差值.....		381
A.3	功能块总览.....		385
B	附录.....		391
B.1	消息.....		391
B.2	参数.....		401
	索引.....		461

简介

1.1 Drive Control Chart (DCC) 简介

用于 SINAMICS 和 SIMOTION 的 Drive Control Chart (DCC: 驱动控制图) 能够借助自由使用的控制块、算法块以及逻辑块对设备功能进行图形化配置和扩展。

Drive Control Chart (DCC: 驱动控制图) 实现了对 SIMOTION 运动控制系统以及 SINAMICS 驱动系统技术功能的最简易配置。这为用户在使指定系统能够使用其机床特殊功能方面打开了新局面。DCC 对可用功能的数目没有限制; 该数目仅受目标平台性能限制。



DCC 包含 DCC 编辑器和 DCB 库 (带有标准 DCC 块的功能块库)。

用户友好型 DCC 编辑器能够提供控制环结构的简易图形化配置和清晰表现以及现有功能图的高度复用性。

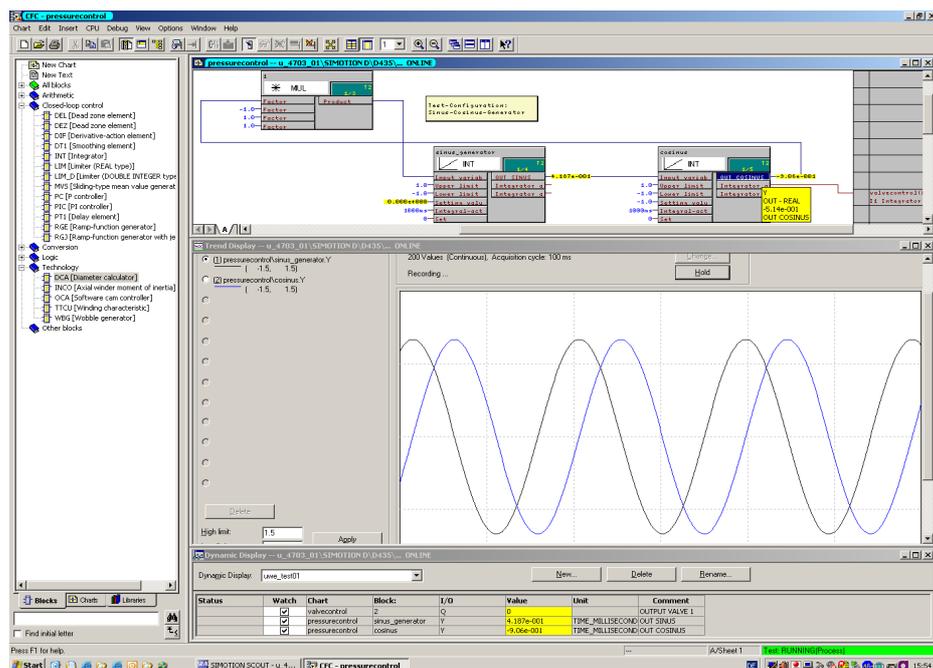
通过拖放操作从预先定义好的库（DCB 库）中选择具有多背景功能的功能块（Drive Control Blocks (DCB: 驱动控制块)）并对其进行图形化互连，即可实现开环和闭环控制功能。测试和诊断功能可帮助验证程序状态或者识别出错原因。

功能块库提供大量控制块、算法块和逻辑块以及扩展开环和闭环控制功能供选择。

所有通用逻辑功能（与、异或、接通 / 断开延迟、RS 双稳态多谐振荡器、计数器等）均可用于二进制信号的逻辑运算选择、评估和获取。多种算法功能（诸如绝对值生成、除法器 and 最小值 / 最大值分析）可用于监控和评估数值变量。除驱动控制外，还可简单配置轴机功能、PI 控制器、斜坡函数发生器或扫频信号发生器。

在结合 SIMOTION 运动控制系统使用时，能够几乎不受限制地对控制结构进行编程。在结合 SIMOTION 运动控制系统使用时，能够几乎不受限制地对控制结构进行编程。

用于 SINAMICS 驱动器的 Drive Control Chart 为直接在变频器中便捷处理驱动级开环和闭环控制任务奠定了基础。这使 SINAMICS 产品能够更好地适应设定的任务。驱动器就地处理支持模块化机床概念并使整体机床性能提升。



1.2 库

功能块位于库中，在 DCC 编辑器中作为技术包导入。

库分为两种：

1. SIMOTION 库包含本文档中标识的 SIMOTION 块。
2. SINAMICS 库包含本文档中标识的 SINAMICS 块。

在附录 A3 和块描述章节的概述中可以了解到 SIMOTION 和 / 或 SINAMICS 中可用的此处描述的功能块。

1.2.1 兼容性

SIMOTION

标准库版本为 `dcblibV4_0_simotion4_3`。
但此库可用于 V4.3 及以上版本的 SIMOTION 设备。

对于 SIMOTION V4.1 版本的目标设备，在 SIMOTION SCOUT 上默认安装 `dcblibV2_0_simotionV4.1.5` 版本的库。
`dcblibV2_0_simotionV4.1.5` 版本的库也可用于 SIMOTION V4.2 和 V4.3 设备。

对于 SIMOTION V4.2 版本的目标设备，在 SIMOTION SCOUT 上默认安装 `dcblibV3_0_simotionV4.2` 版本的库。
`dcblibV3_0_simotionV4.2` 版本的库也可用于 SIMOTION V4.3 设备。

SINAMICS

当为 SINAMICS V4.5 设备时，可执行以下标准库：

- SINAMICS V4.5 (`dcblibV3_0_sinamics4_5`)
- SINAMICS V4.4 (`dcblibV3_0_sinamics4_4`)
- SINAMICS V4.3 (`dcblibV2_0_sinamics4_3`)

当为 SINAMICS V4.4 设备时，可执行以下标准库：

- SINAMICS V4.4 (`dcblibV3_0_sinamics4_4`)
- SINAMICS V4.3 (`dcblibV2_0_sinamics4_3`)

当为 SINAMICS V4.3.x 设备时，可执行以下标准库：

- SINAMICS V4.3 (dcblibV2_0_sinamics4_3)

当为 SINAMICS V2.6.x 设备时，可执行以下标准库：

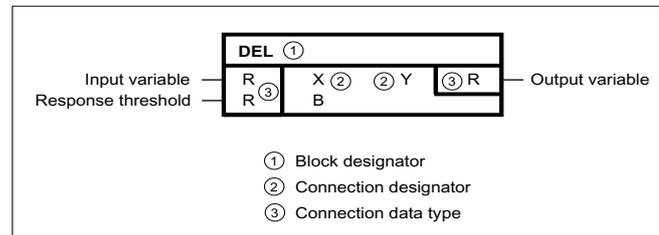
- SINAMICS V2.6 (dcblibV2_0_sinamics2_6)

当为 SINAMICS V2.5.SP1 设备时，可执行以下标准库：

- SINAMICS V2.5 (dcblibV2_0_sinamics2_5_1)

1.3 功能块术语表

一个功能块显示如下：



功能块通过以下属性识别：

块名称

每种数据类型都拥有对应的一种块类型。为了降低同一功能不同数据类型功能块的区分难度，功能块都带有与其数据类型相对应的后缀，但后缀很少用于 Real 和 Bool 数据类型（如 MUL_L: INTEGER 型乘法器，MUL: REAL 型乘法器）。常用扩展名如下表所示。

表 1-1 块名称

块名称后缀	输入 / 输出变量的数据类型
_I	整数
_D	双整数
_W	字
_R	实数（可选）
_B	布尔数（可选）
_SI	短整数
_M	模数
_BY	字节
_UI	无符号整数
_US	无符号短整数
_UD	无符号双整数
_DW	双字
_LR	长实数

连接名称

- 为了从输入或输出变量中识别字段，连接名称扩展了一个以 1 开头的索引，如 X1、X2、X3 等。
- 对于一般数目的输入（如 ADD），连接名称的索引开头为 1，如 X1、X2、X3 等。

下表概述了常用连接名称。

表 1-2 连接名称

连接名称	应用
X、X1、X2.....	数值输入变量
Y、Y1、Y2.....	数值输出变量
I、I1、I2.....	二进制输入变量
Q、Q1、Q2.....	二进制输出变量
IS	位串输入（字）
QS	位串输出（字）

如果主输入 / 输出变量（如限值、时间数据、替代值、状态显示）使用了更多的输入和输出，则不使用主输入 / 输出变量池中的名称。二级变量的优选名称如下表所示：

表 1-3

连接名称	应用
LU	输入：上限
LL	输入：下限值
SV	输入：设定值
S	输入：设置设定值
R	输入：复位设定值
QU	输出：达到上限
QL	输出：达到下限
QF	输出：错误指示
QE	输出 Y 等于输入 X
QN	反二进制变量

连接数据类型

数据类型的缩写名称如下表所示：

表 1-4

缩写	位宽	数据类型 (符合 IEC 61131-3)	说明
BO	1	BOOL	布尔数
BY	8	BYTE	位串，无符号整数
SI	8	SINT	带符号短整数
DI	32	DINT	带符号双整数
DW	32	DWORD	位串，无符号整数
I	16	INT	带符号整数
R	32	REAL	浮点单精度（符合 IEEE 754）
LR	64	LREAL	浮点双精度（符合 IEEE 754）
T	32	SDTIME	浮点单精度（符合 IEEE 754）
W	16	WORD	位串，无符号整数
AID	32	-	告警 ID

1.4 块连接

块连接显示 DCB 块的连接点，通过该连接点可以实现功能块的互联。
此处有如下选项：

- 块输出
- 块输入

它们具有如下属性：

- 输入位于该功能块的左边，且为互联的目标功能块。
- 输出位于该功能块的右边，且为互连的源功能块。

1.5 字节排序

在互连功能块时，并非必须考虑数据的字节排序问题。在数据类型转换和算术运算时，暗含考虑了目标系统的字节排序问题。为了处理系统外数据所进行的任何字节交换必须由该系统完成，如：通过 Profibus 传输数据前可能必须以大端格式完成字节交换。

1.6 不同数据类型的直接互连

在互连功能块时，目标与源功能块必须为同种数据类型。当数据类型不同时，有些特别转换功能块可以进行数据类型转换。

以下允许的暗含转换例外。允许的转换如下表所示。

以下不需要借助转换功能块就能进行互连的数据类型也是例外。此时，输出变量的二进制值不变地转换为输入变量。

表 1-5 转换

输入端	输出端	说明
WORD	INT	字变量到整数变量的互连
INT	WORD	整数变量到字变量的互连
DWORD	DINT	双字变量到双整数变量的互连
DINT	DWORD	双整数变量到双字变量的互连
BYTE	SINT	字节变量到短整数变量的互连
SINT	BYTE	短整数变量到字节变量的互连
USINT	BYTE	不带符号的短整数变量到字节变量的互连
BYTE	USINT	字节变量到不带符号的短整数变量的互连
USINT	SINT	不带符号的短整数变量到短整数变量的互连

表 1-5 转换

输入端	输出端	说明
SINT	USINT	短整数变量到不带符号的短整数变量的互连
UINT	WORD	不带符号的整数变量到字变量的互连
WORD	UINT	字变量到不带符号的整数变量的互连
UINT	INT	不带符号的整数变量到整数变量的互连
INT	UINT	整数变量到不带符号的整数变量的互连
UDINT	DWORD	不带符号的双整数变量到双字变量的互连
DWORD	UDINT	双字变量到不带符号的双整数变量的互连
UDINT	DINT	不带符号的双整数变量到双整数变量的互连
DINT	UDINT	双整数变量到不带符号的双整数变量的互连
SDTIME	REAL	SDTime 变量到实数变量的互连

1.7 块初始化

初始化决定功能块的启动条件。在循环处理¹ 功能块前由系统完成初始化。初始化功能块的顺序与配置好的优先级和处理顺序一致。初始化时，功能块已配置的互连和常量已经处于激活状态。此时，互连源功能块的值在功能块中已经可用。如果初始化时功能块状态特别，可以参见“初始化”章节下相应块的描述来了解更多信息。初始化时，功能块必须在一个时间片（SINAMICS）进行分配或分配给一个任务（SIMOTION）。

1. 对于 SIMOTION 4.1 SP2 及以上版本，在 STOP/RUN 过渡期（SIMOTION）或向循环运算的过渡期（SINAMICS）执行初始化。

1.8 在配置样例中执行复杂功能

配置样例适用于“便捷斜坡函数发生器”和“工艺控制器”。二者是基于 SIMOVERT MASTERDRIVES 变频器系列的“自由块”。

通过互连 DCB 块来执行便捷斜坡函数发生器的功能。DCB 块以配置样例形式呈现。

工艺控制器相关的注意事项

- 由于时间常量受限于功能块的采样时间，因此平滑滤波器无法通过设置时间常量 $T = 0$ 来停止运行。滤波器必须通过信号来明确停止运行。相应的二进制输入必须在配置样例中给出响应。
- D 分量无法通过设置延迟时间 $T_v = 0$ 来停止运行，必须通过二进制信号来明确停止运行。相应的二进制输入必须在配置样例给定值出响应。
- I 分量无法通过设置 $T_n = 0$ 来停止运行。因此 PIC 的 I 分量必须通过设置 $SV = 0$ 和 $S = 1$ 来明确复位。

注意

此时，还应该利用 DCC 块的优点，只配置 / 传递需要的功能块。也就是说，不用回到配置样例的最开始，而可以开始于基本块，如 RGJ (ramp-function generator with rounding: 支持取整的斜坡函数发生器) 或 PIC (PI controller: 比例积分控制器)，必要时 还可以扩展。

1.8.1 导入配置样例

要将项目样例导入 SCOUT/STARTER，需要从菜单中选择命令 Project -> Import...

SIMOTION

要在 SCOUT 上导入 DCC 图，项目需要包含 SIMOTION 设备。在 Programs 文件夹中，通过右键菜单中 Expert -> Import object 选择 DCC 图的导出路径。通常可以在以下路径中找到图文件：

C:\Program Files\Siemens\Step7\Examples\dcc\SIMOTION

SINAMICS

要在 SCOUT/STARTER 上导入 DCC 图，项目必须包含带有一个驱动对象 (Drive Object) 的 S120 驱动单元。通过驱动对象的右键菜单 Expert -> Import object 选择 DCC 图的导出路径。通常可以在以下路径中找到图文件：

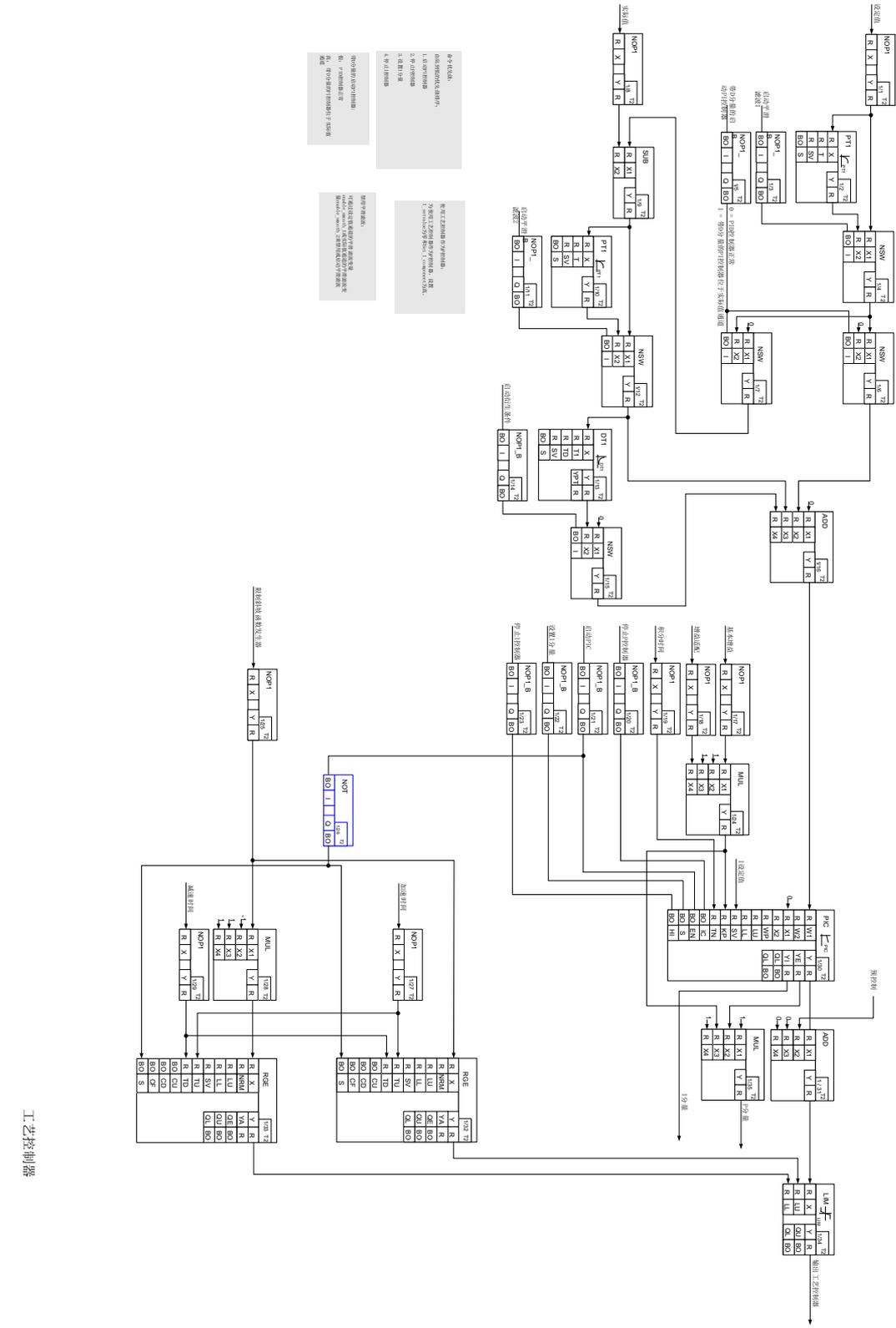
C:\Program Files\Siemens\Step7\Examples\dcc\SINAMICS

对于 V4.3 及以下版本的 SCOUT/STARTER，配置样例作为项目存档，导出路径如下：

C:\Program Files\Siemens\Step7\Examples\dcc\SIMOTION

注意

在基于 SCOUT 项目创建了项目样例之后，该项目样例同时包含了 SINAMICS_ Integrated 和 SINAMICS stand-alone/CU320 的配置。在导入支持 STARTER (独立式) 的项目样例时，自然会拒绝 SIMOTION 组件。然而 CU320 组件却可继续准确地导入，且可复制。



安全注意事项
 在启动或停止系统前，请确保所有人员都已撤离工作区域。
 请勿在运行过程中更改任何参数，以免造成设备损坏。
 如有疑问，请联系专业人员进行咨询。

警告
 请勿在运行过程中打开柜门，以免触电。
 请勿在运行过程中触摸任何带电部件，以免触电。
 请勿在运行过程中进行任何维修工作，以免触电。

重要提示
 本手册中的所有操作均应在断电状态下进行。
 请勿在运行过程中更改任何参数，以免造成设备损坏。

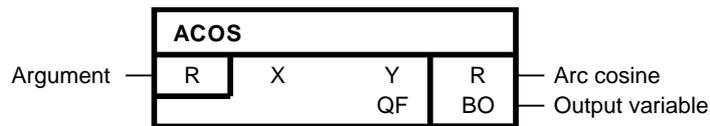
工控控制器

Arithmetic

2.1 ACOS 反余弦功能块

 SIMOTION SINAMICS

符号



简要说明

- 确定自变量的反余弦值

操作模式

该功能块通过弧度法确定X输入端自变量的相应反余弦值，并在Y输出端输出结果值。

$$Y = \arccos X$$

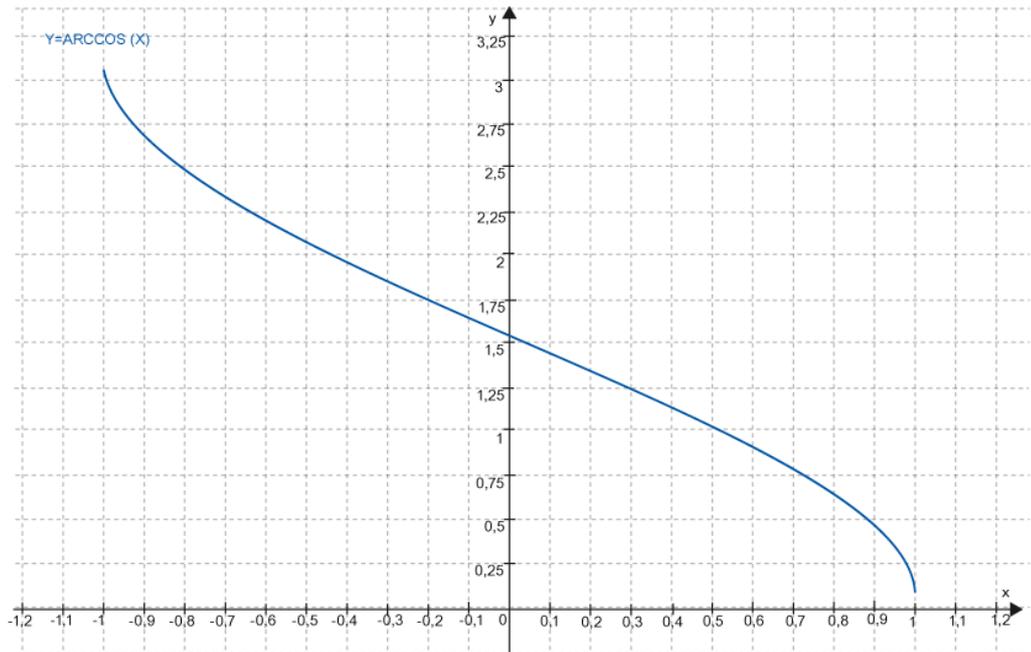
允许输入范围： $-1.0 \leq X \leq +1.0$

输出范围： $0.0 \leq Y \leq \pi$

当该自变量超出允许输入范围时，Y输出则限于 π （当 $X < -1.0$ ）或 0.0 （当 $X > +1.0$ ），并同时设置二进制输出 $QF = 1$ 。

2.1 ACOS 反余弦功能块

传输功能



多个块连接

块 连接	说明	预分配	取值范围	数
X	自变量	0.0	REAL	
Y	反余弦	$\pi / 2$	REAL	
QF	输出变量	0	0/1	

据

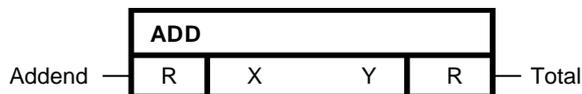
可在线加载	是
特别注意点	-

2.2 ADD 加法器 (REAL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个REAL型输入的加法器

操作模式

该功能块区分正负号，使X输入端的值相加。在Y输出端输出-3.402823 E38到3.402823 E38范围内的结果。

算法：

$$Y = X1 + X2 + X3 + X4$$

多个块连接

块 连接	说明	预分配	取值范围	数
X	加数	0.0	REAL	
Y	合计	0.0	REAL	

据

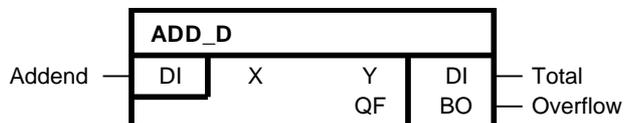
可在线加载	是
可以配置，通过：	循环任务
特别注意点	X包含最多4个输入 (X1到X4)

2.3 ADD_D 加法器 (DOUBLE INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个DOUBLE INTEGER型输入值的加法器

操作模式

该功能块区分正负号，使X输入端的值相加。结果值的范围为 $-2147483648 (2^{31})$ 到 $+2147483647 (2^{31}-1)$ ，并在Y输出端输出。

算法：

$$Y = X1 + X2 + X3 + X4$$

多个块连接

块 连接	说明	预分配	取值范围	数
X	加数	0	DINT	
Y	合计	0	DINT	
QF	溢位	0	0/1	

据

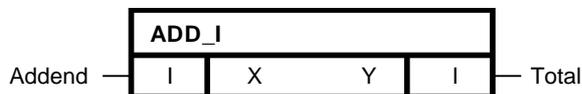
可在线加载	是
特别注意点	X包含最多4个输入 (X1到X4)

2.4 ADD_I 加法器 (INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个INTEGER型输入的加法器

操作模式

该功能块区分正负号，使X输入端的值相加。在Y输出端输出-32768到+32767范围内的结果。

算法：

$$Y = X1 + X2 + X3 + X4$$

多个块连接

块 连接	说明	预分配	取值范围	数
X	加数	0	INT	
Y	合计	0	INT	

据

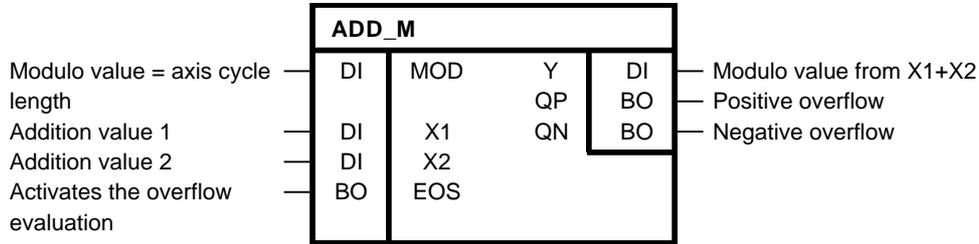
可在线加载	是
特别注意点	X包含最多4个输入 (X1到X4)

2.5 ADD_M 在正确的轴周期中增加值的模加法器

☑ SIMOTION

☑ SINAMICS

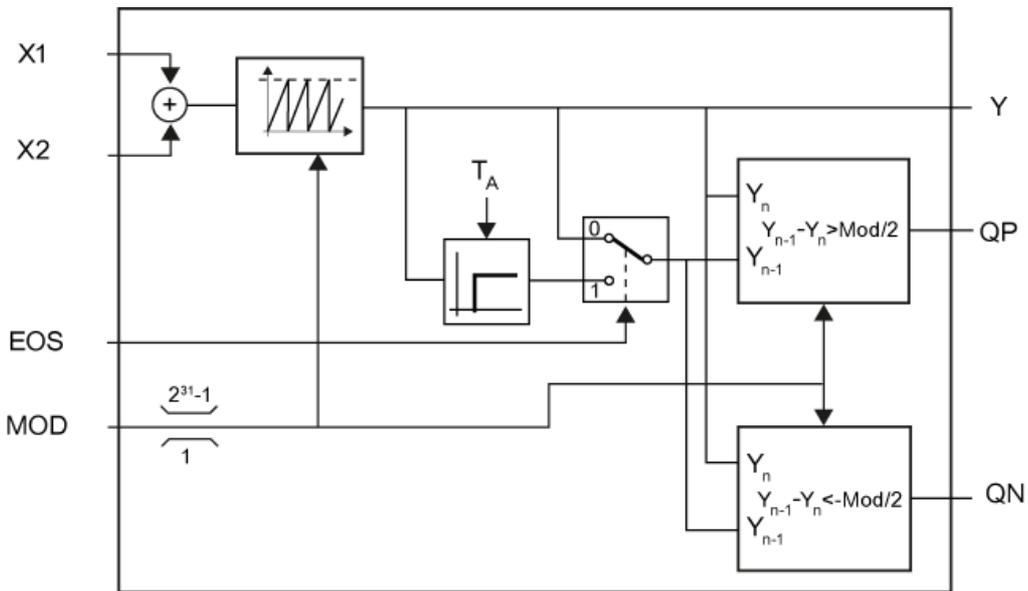
符号



简要说明

- ADD_M功能块用于增加位置值。可用于“统计”真实主设备中位置设定值或死时间补偿的偏移量。

方框图



操作模式

该功能块增加X1和X2输入值。在 $1 \dots 2^{31}$ 到1范围内的模态值可以在MOD输入端确定,并用于X1和X2的求和运算。因此,模运算的Y结果值总是介于0到MOD之间。

EOS输入可用于激活溢位评估。当EOS = 1

为正溢位: $QP = Y_{n-1} - Y_n > MOD/2$

为负溢位: $QN = Y_{n-1} - Y_n < -MOD/2$

当EOS = 0时,
QP = 0 QN = 0

这样可以在设置偏移量时撤消溢位评估。
在改变模值时，溢位评估会停止一个周期。

多个块连接

块 连接	说明	预分配	取值范围	数
MOD	模值 = 轴周期时间	1	DINT	
X1	增加值1	0	DINT	
X2	增加值2	0	DINT	
EOS	激活溢位评估	0	0/1	
Y	由X1+X2得出的模值	0	DINT	
QP	正溢位	0	0/1	
QN	负溢位	0	0/1	

据

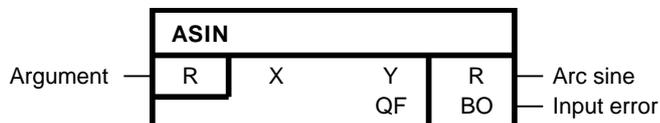
可在线插入	是
特别注意点	-

2.6 ASIN 反正弦功能块

SIMOTION

SINAMICS

符号



简要说明

- 确定自变量的反正弦值

操作模式

该功能块通过弧度法确定X输入端自变量的相应反正弦值，并在Y输出端输出结果值。

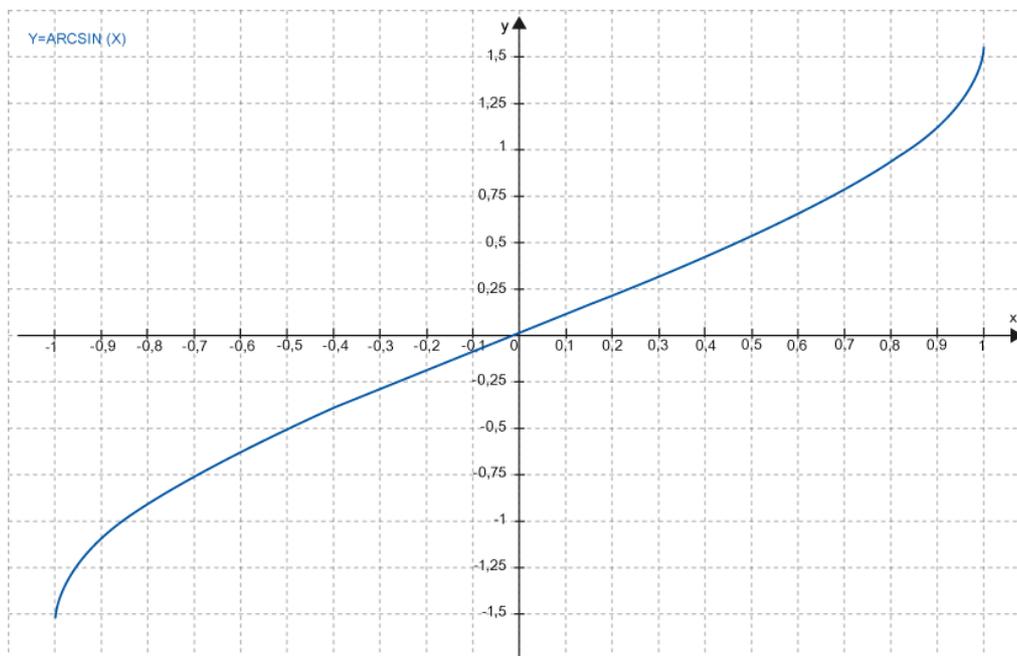
$$Y = \arcsin X$$

允许输入范围： $-1.0 \leq X \leq +1.0$

输出范围： $-\pi/2 \leq Y \leq \pi/2$

当该自变量超出允许输入范围 $|X| > 1.0$ 时，Y输出则限于 $-\pi/2$ （当 $X < -1.0$ ）或 $\pi/2$ （当 $X > +1.0$ ），并同时设置二进制输出 $QF = 1$ 。

XY图



多个块连接

块 连接	说明	预分配	取值范围	数
X	自变量	0.0	REAL	
Y	反正弦	0.0	REAL	
QF	输入错误	0	0/1	

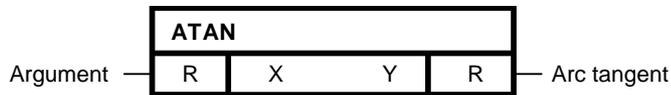
据

可在线插入	是
特别注意点	-

2.7 ATAN 反正切功能块

 SIMOTION SINAMICS

符号



简要说明

- 确定自变量的反正切值

操作模式

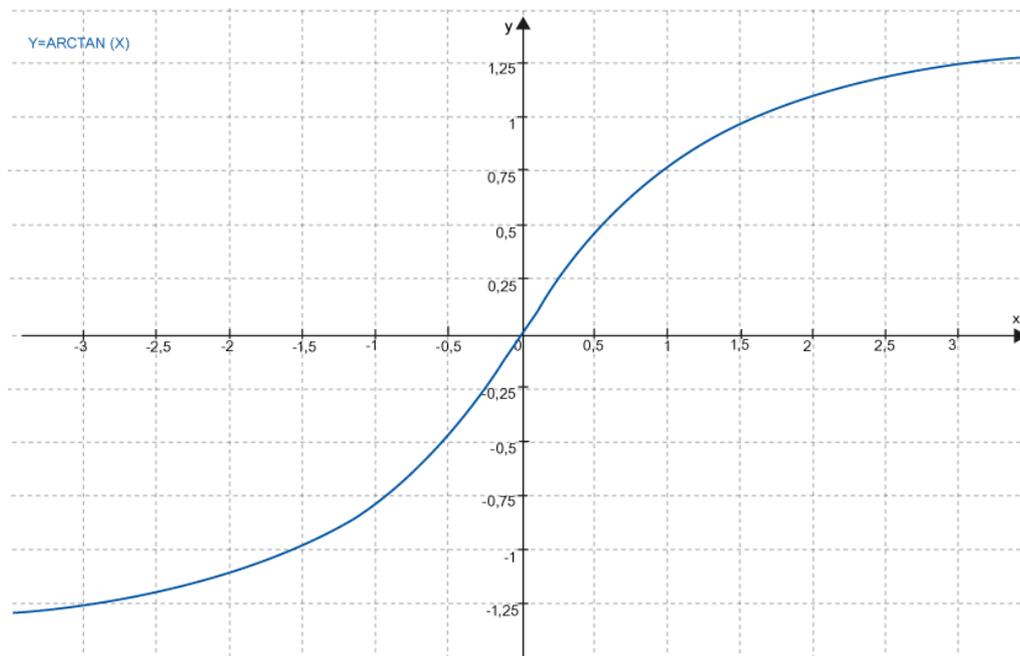
该功能块通过弧度法确定X输入端自变量的相应反正切值，并在Y输出端输出该结果值。

$$Y = \arctan X$$

允许输入范围：-3.402823 E38到3.402823 E38

输出范围： $-\pi/2 \leq Y \leq \pi/2$

XY图



多个块连接

块 连接	说明	预分配	取值范围	数
X	自变量	0.0	REAL	
Y	反正切	0.0	REAL	

据

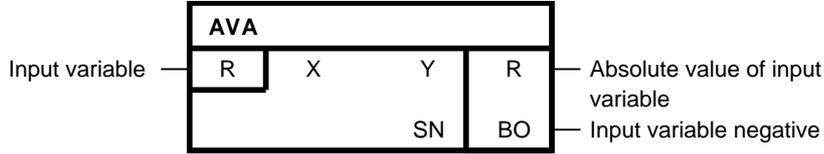
可在线插入	是
特别注意点	-

2.8 AVA 支持符号评估的绝对值发生器

SIMOTION

SINAMICS

符号



简要说明

- 生成REAL型绝对值的算法功能块

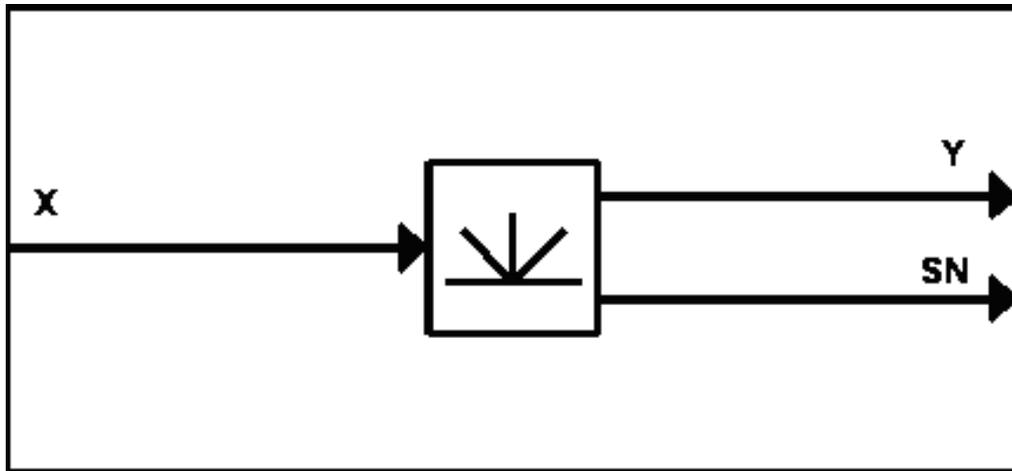
操作模式

该功能块生成X输入端输入值（输入变量）的绝对值 在Y输出端输出结果。

$$Y = |X|$$

当输入变量为负数时，同时设置二进制输出SN = 1。

方框图



多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输入变量的绝对值	0.0	REAL	
SN	输入变量负数	0	0/1	

据

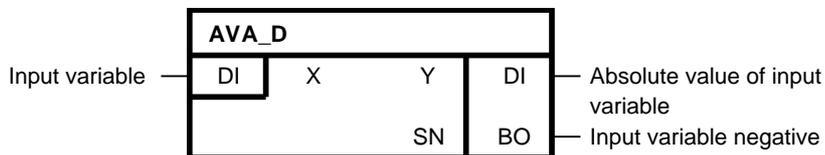
可在线加载	是
特别注意点	-

2.9 AVA_D 绝对值发生器 (DOUBLE INTEGER型)

SIMOTION

SINAMICS

符号



简要说明

- 生成DOUBLE INTEGER型绝对值的算法功能块

操作模式

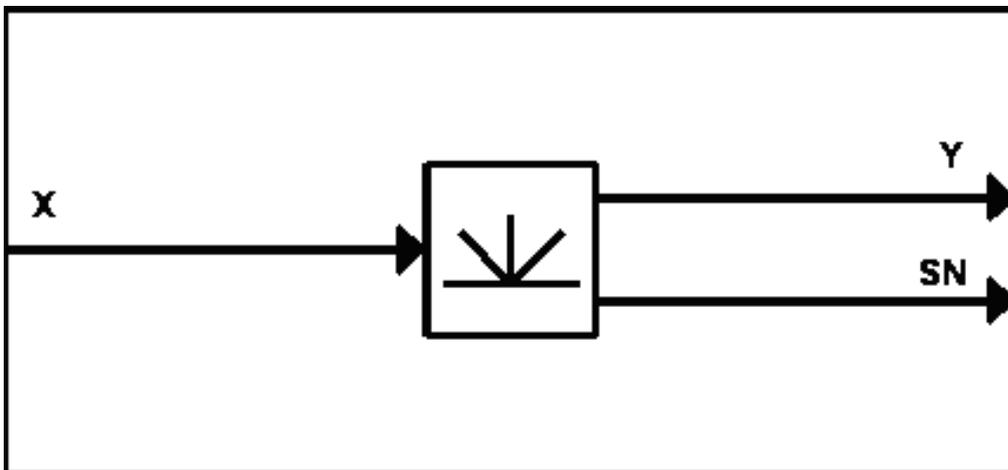
该功能块生成X输入端输入值（输入变量）的绝对值 在Y输出端输出结果。

$$Y = |X|$$

当输入变量为负数时，同时设置二进制输出SN = 1。

输出值Y -2147483648和SN 1对应输入值-2147483648。

方框图



多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
Y	输入变量的绝对值	0	DINT	
SN	输入变量负数	0	0/1	

据

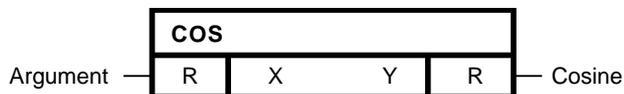
可在线加载	是
特别注意点	-

2.10 COS 余弦功能块

SIMOTION

SINAMICS

符号



简要说明

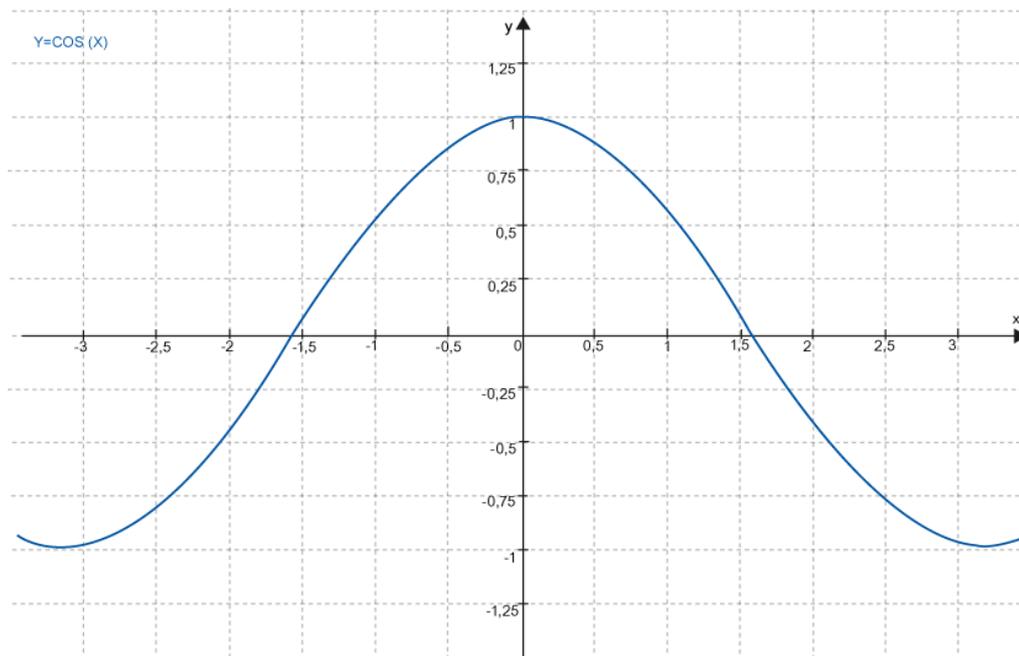
确定自变量的余弦值。

操作模式

该功能块通过弧度法确定X输入端自变量的相应余弦值，并在Y输出端输出结果值。

$$Y = \cos X$$

XY图



X 即是模 π

多个块连接

块 连接	说明	预分配	取值范围	数
X	自变量	0.0	REAL	

块 连接	说明	预分配	取值范围	数
Y	余弦	1	REAL	

据

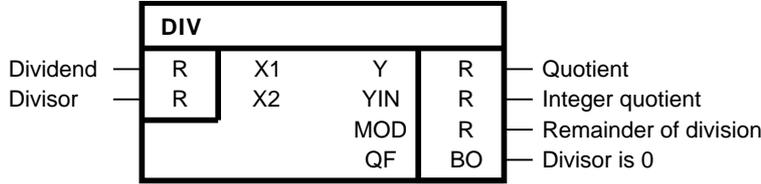
SIMOTION	V4.1及以上版本
SINAMICS	V4.4及以上版本
可在线加载	是
特别注意点	-

2.11 DIV 除法器 (REAL型)

SIMOTION

SINAMICS

符号



简要说明

- 支持两个REAL型输入的除法器

操作模式

该功能块用连接X1的输入值除以连接X2的输入值。

在输出端Y、YIN和MOD输出结果。

- Y输出包含该带有整数位和小数位的商值。
- YIN输出包含整数商值。
- MOD输出包含余数 (绝对余数值)

Y输出的近似取值范围为: -3.4×10^{38} 到 $+3.4 \times 10^{38}$ 。

$$Y = \frac{X1}{X2}$$

$$MOD = (Y - YIN) * X2$$

当Y输出值超出 -3.402823×10^{38} 到 3.402823×10^{38} 的允许取值范围 (由于除数X2太小或小于0) 时, 区分正负号的输出范围限值即是连接Y的输出。同时, 设置二进制输出QF=1。当X2为0时, 输出YIN和MOD保持各自的最后值。

当除法为0/0时, 功能块输出Y保持不变。二进制输出QF设置为1。当除以0时, 输出MOD保持其最后值。

真值表

以上情况下的功能块响应如下真值表所示。

X1/X2	Y	YIN	MOD	QF
X/0	带正确正负号的限值	YIN n-1	MOD n-1	1
0/0	Y n-1	YIN n-1	MOD n-1	1
0/X	0	0	0	0

多个块连接

块 连接	说明	预分配	取值范围	数
X1	被除数	0.0	REAL	
X2	除数	1	REAL	
Y	商	0.0	REAL	
YIN	整数商	0.0	REAL	
MOD	商余数	0.0	REAL	
QF	除数为0	0	0/1	

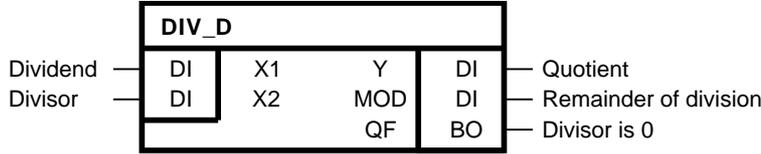
据

可在线插入	是
特别注意点	-

2.12 DIV_D 除法器 (DOUBLE INTEGER型)

SIMOTION SINAMICS

符号



简要说明

- 支持2个DOUBLE INTEGER型输入值的除法器

操作模式

该功能块区分正负号，以连接X1的输入值除以连接X2的输入值。商值的范围为 -2147483648 (2³¹)到2147483647 (2³¹ -1)，并在连接Y输出。

$$Y = \frac{X1}{X2}$$

商余数在连接MOD处输出。商余数MOD的正负号与被除数X1相匹配。

$$MOD = X1 \text{ MOD } X2$$

当Y输出值超出-2147483648 (2³¹)到+2147483647 (2³¹-1) 的允许取值范围（当除数X2等于0）时，区分正负号的输出范围限值即是连接Y的输出。同时，设置二进制输出QF=1。

当除法为0/0时，功能块输出Y保持不变。二进制输出QF设置为1。当除以0时，输出MOD保持其最后值。

真值表

以上情况下的功能块响应如以下真值表所示。

X1/X2	Y	MOD	OF
X/0	带正确正负号的限值	MOD n-1	1
0/0	Y n-1	MOD n-1	1
0/X	0	0	0

多个块连接

块 连接	说明	预分配	取值范围	数
X1	被除数	0	DINT	
X2	除数	1	DINT	

2.12 DIV_D 除法器 (DOUBLE INTEGER型)

块 连接	说明	预分配	取值范围	数
Y	商	0	DINT	
MOD	商余数	0	DINT	
QF	除数为0	0	0/1	

据

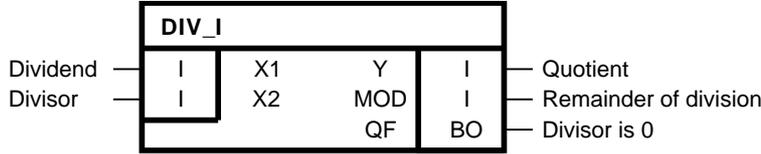
可在线插入	是
特别注意点	-

2.13 DIV_I 除法器 (INTEGER型)

SIMOTION

SINAMICS

符号



简要说明

- 支持两个INTEGER型输入的除法器

操作模式

该功能块区分正负号，以连接X1的输入值除以连接X2的输入值。商值的范围为-32768到+32767，并在连接Y输出。

$$Y = \frac{X1}{X2}$$

商余数在连接MOD处输出。商余数MOD的正负号与被除数X1相匹配。

$$MOD = X1 \text{ MOD } X2$$

当Y输出值超出-32768到+32767的允许取值范围（当除数等于0）时，区分正负号的输出范围限值即是连接Y的输出。同时，设置二进制输出QF=1。

当除法为0/0时，功能块输出Y保持不变。二进制输出QF设置为1。当除以0时，输出MOD保持其最后值。

真值表

以上情况下的功能块响应如以下真值表所示。

X1/X2	Y	MOD	QF
X/0	带正确正负号的限值	MOD n-1	1
0/0	Y n-1	MOD n-1	1
0/X	0	0	0

多个块连接

块 连接	说明	预分配	取值范围	数
X1	被除数	0	INT	
X2	除数	1	INT	
Y	商	0	INT	

块 连接	说明	预分配	取值范围	数
MOD	商余数	0	INT	
QF	除数为0	0	0/1	

据

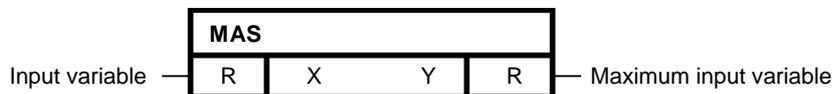
可在线插入	是
特别注意点	-

2.14 MAS 最大值评估器

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个REAL型输入的比较功能块，确定处理过程中的最大输入值。

操作模式

该功能块确定输入X1到X4处的最大值。

在Y输出端输出结果。

$$Y = \max. \{X1, X2, X3, X4\}$$

当所有输入对应同一个值时，该值即作为最大输入变量的输出。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	-3. 402823 E38	REAL	
Y	最大输入变量	0. 0	REAL	

据

可在线加载	是
特别注意点	X包含最多4个输入（X1到X4）

2.15 MIS 最小值评估器

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个REAL型输入的比较功能块，确定处理过程中的最小输入值。

操作模式

该功能块确定输入X1到X4处的最小值。

在Y输出端输出结果。

$$Y = \min. \{X1, X2, X3, X4\}$$

当所有输入对应同一个值时，该值即作为最小输入变量的输出。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	3.402823 E38	REAL	
Y	最小输入变量	0.0	REAL	

据

可在线加载	是
特别注意点	X包含最多4个输入（X1到X4）

2.16 MUL 乘法器 (REAL型)

 SIMOTION SINAMICS

符号



简要说明

- 最多支持4个REAL型输入的乘法器

操作模式

该功能块区分正负号，使一般输入X1到X4的值相乘。在输出端Y输出-3.402823 E38到+3.402823 E38范围内的结果。

$$Y = X1 \cdot X2 \cdot X3 \cdot X4$$

多个块连接

块 连接	说明	预分配	取值范围	数
X	因子	1.0	REAL	
Y	乘积	0.0	REAL	

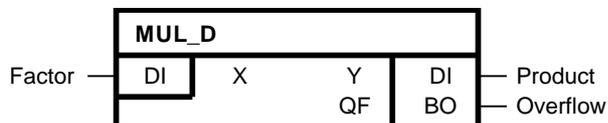
据

可在线加载	是
特别注意点	X包含最多4个输入 (X1到X4)

2.17 MUL_D 乘法器 (DOUBLE INTEGER型)

 SIMOTION SINAMICS

符号



简要说明

- 最多支持4个DOUBLE INTEGER型输入的乘法器

操作模式

该功能块区分正负号，使一般输入X1到X4的值相乘。结果值的范围为 $-2147483648 (2^{31})$ 到 $+2147483647 (2^{31}-1)$ ，并在Y输出端输出。

$$Y = X1 \cdot X2 \cdot X3 \cdot X4$$

多个块连接

块 连接	说明	预分配	取值范围	数
X	因子	1	DINT	
Y	乘积	0	DINT	
QF	溢位	0	0/1	

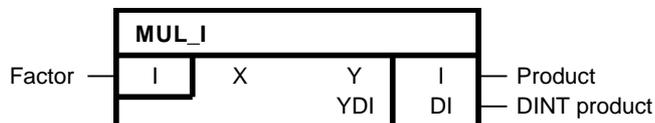
据

可在线加载	是
特别注意点	X包含最多4个输入 (X1到X4)

2.18 MUL_I 乘法器 (INTEGER型)

 SIMOTION SINAMICS

符号



简要说明

- 最多支持4个INTEGER型输入的乘法器

操作模式

该功能块区分正负号，使一般输入X1到X4的值相乘。在Y输出端输出-32768到+32767范围内的结果。另外，结果值的范围为-2147483648 (2^{31})到+2147483647 ($2^{31}-1$)，并在输出端YDI输出。

$$Y = X1 \cdot X2 \cdot X3 \cdot X4$$

多个块连接

块 连接	说明	预分配	取值范围	数
X	因子	1	INT	
Y	乘积	0	INT	
YDI	DINT乘积	0	DINT	

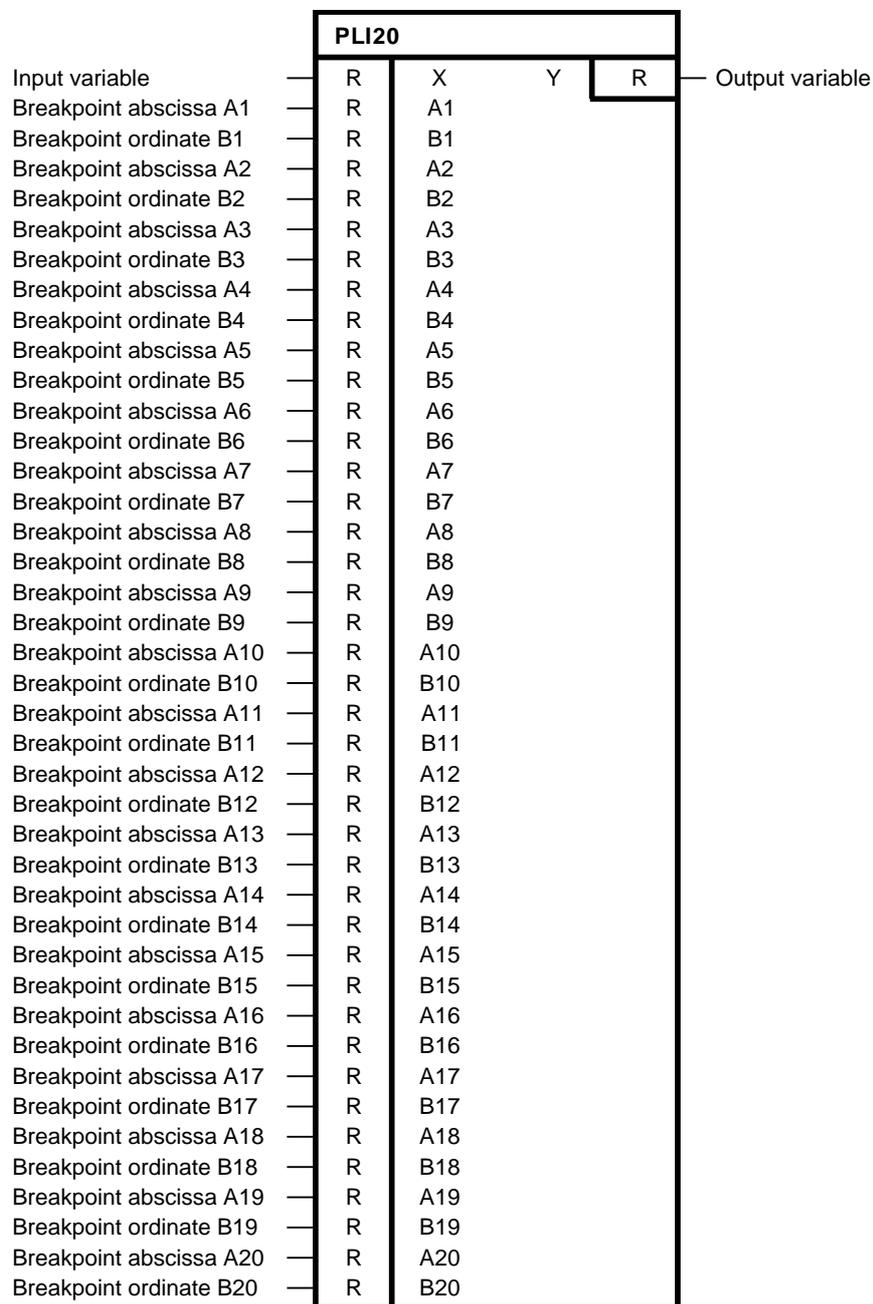
据

可在线加载	是
特别注意点	X包含最多4个输入 (X1到X4)

2.19 PLI20 多段线，20个断点

 SIMOTION SINAMICS

符号



简要说明

REAL型功能块

- 用于特性曲线的线性化
- 用于非线性传递元素的模拟
- 用于线段规定的控制器增益

操作模式

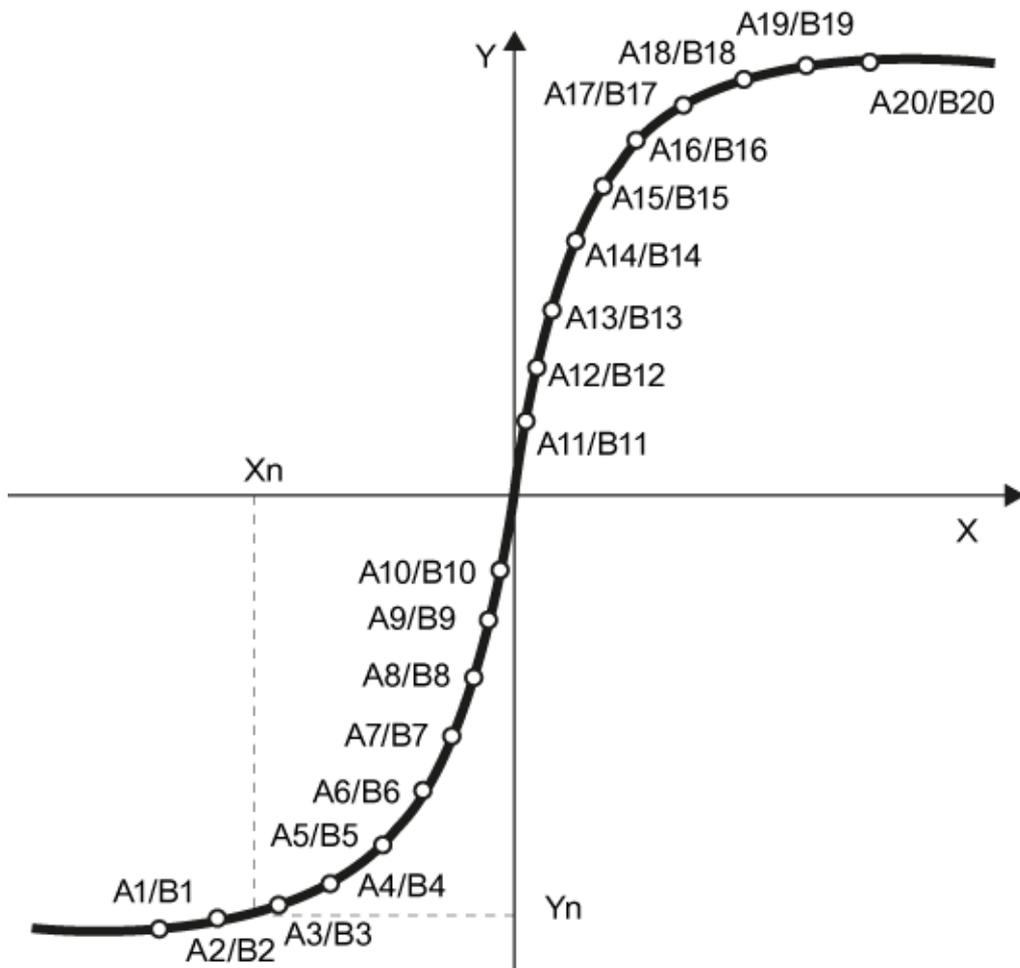
- 该功能块任意使输出变量Y适配于输入变量, 最多通过 四象限中的20个断点。
- 在断点间进行线性插值。 在横向上, 特性曲线在A1或A20之外。

设置指导

设置时必须保证A1到A20的值按升序排列, 否则会输出错误值。 B1到B20的纵坐标值可以任意选择, 无需考虑前一值。

当不需要断点(例如, 自A16/B16起)时, 下面的横坐标和纵坐标(A16/B16到A20/B20)必须获得与A15/B15相同的值。

示例



磁化特征曲线模拟

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
A1	断点横坐标A1	0.0	REAL	
B1	断点纵坐标B1	0.0	REAL	
A2	断点横坐标A2	0.0	REAL	
B2	断点纵坐标B2	0.0	REAL	
A3	断点横坐标A3	0.0	REAL	
B3	断点纵坐标B3	0.0	REAL	
A4	断点横坐标A4	0.0	REAL	
B4	断点纵坐标B4	0.0	REAL	
A5	断点横坐标A5	0.0	REAL	
B5	断点纵坐标B5	0.0	REAL	
A6	断点横坐标A6	0.0	REAL	
B6	断点纵坐标B6	0.0	REAL	
A7	断点横坐标A7	0.0	REAL	
B7	断点纵坐标B7	0.0	REAL	
A8	断点横坐标A8	0.0	REAL	
B8	断点纵坐标B8	0.0	REAL	
A9	断点横坐标A9	0.0	REAL	
B9	断点纵坐标B9	0.0	REAL	
A10	断点横坐标A10	0.0	REAL	
B10	断点纵坐标B10	0.0	REAL	
A11	断点横坐标A11	0.0	REAL	
B11	断点纵坐标B11	0.0	REAL	
A12	断点横坐标A12	0.0	REAL	
B12	断点纵坐标B12	0.0	REAL	
A13	断点横坐标A13	0.0	REAL	
B13	断点纵坐标B13	0.0	REAL	
A14	断点横坐标A14	0.0	REAL	
B14	断点纵坐标B14	0.0	REAL	
A15	断点横坐标A15	0.0	REAL	
B15	断点纵坐标B15	0.0	REAL	
A16	断点横坐标A16	0.0	REAL	
B16	断点纵坐标B16	0.0	REAL	
A17	断点横坐标A17	0.0	REAL	
B17	断点纵坐标B17	0.0	REAL	
A18	断点横坐标A18	0.0	REAL	
B18	断点纵坐标B18	0.0	REAL	
A19	断点横坐标A19	0.0	REAL	
B19	断点纵坐标B19	0.0	REAL	
A20	断点横坐标A20	0.0	REAL	

2.19 PLI20 多段线，20个断点

块 连接	说明	预分配	取值范围	数
B20	断点纵坐标B20	0.0	REAL	
Y	输出变量	0.0	REAL	

据

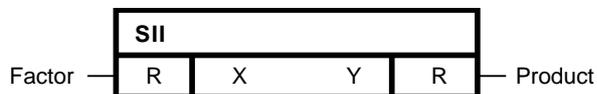
可在线加载	是
特别注意点	-

2.20 SII 反相器

☑ SIMOTION

☑ SINAMICS

符号



简要说明

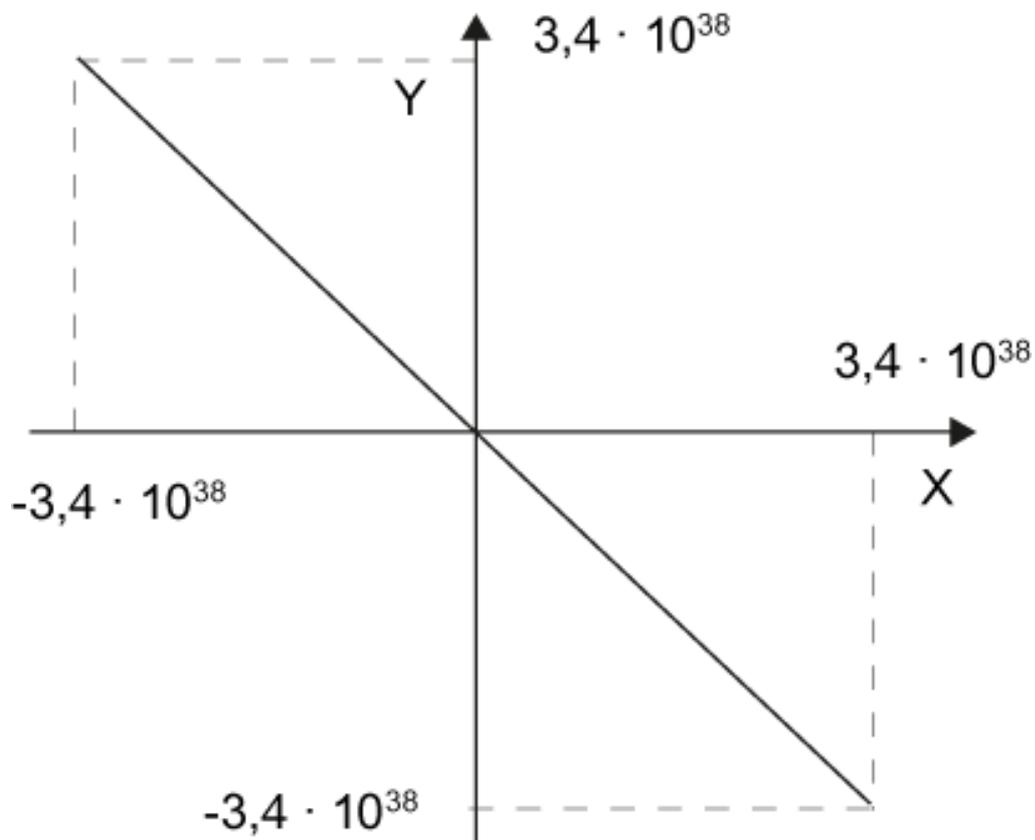
- 支持一个REAL型输入的反相器
- 用于变号的算术函数功能块

操作模式

该功能块反转输入变量X的正负号，并在块输出Y处输出结果（根据以下传送曲线）。

$$Y = -X$$

传输功能



多个块连接

块 连接	说明	预分配	取值范围	数
X	因子	0.0	REAL	
Y	乘积	0.0	REAL	

据

可在线加载	是
特别注意点	-

2.21 SIN 正弦功能块

 SIMOTION SINAMICS

符号



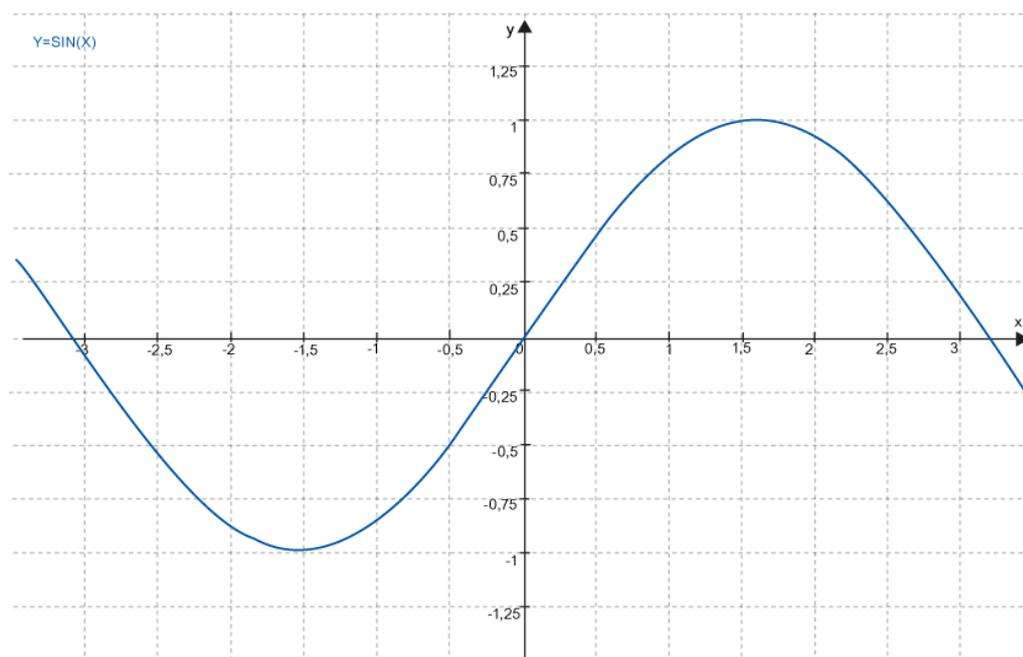
简要说明

确定自变量的正弦值。

操作模式

- 该功能块通过弧度法确定X输入端自变量的相应正弦值，并在Y输出端输出结果值。
- $Y = \sin X$

XY图

X 即是模 π

多个块连接

块 连接	说明	预分配	取值范围	数
X	自变量	0.0	REAL	

2.21 SIN 正弦功能块

块 连接	说明	预分配	取值范围	数
Y	正弦曲线	0.0	REAL	

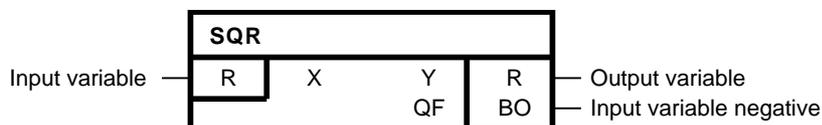
据

SIMOTION	V4.1及以上版本
SINAMICS	V4.4及以上版本
可在线插入	是
特别注意点	-

2.22 SQR 平方根提取器

 SIMOTION SINAMICS

符号



简要说明

用于确定平方根的算术函数功能块。

SQR功能块可用于SIMOTION V4.1及以上版本和SINAMICS V4.4及以上版本。

操作模式

该功能块计算在连接X处输入值的平方根。在连接Y处输出结果。

$$Y = \sqrt{X}$$

当输入变量为负数时，连接Y处则输出0。同时，设置二进制输出QF=1。

真值表

条件	Y	QF
$X > 0$	SQR(X)	0
$X = 0$	0	0
$X < 0$	0	1

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	0.0	REAL	
QF	输入变量负数	0	0/1	

据

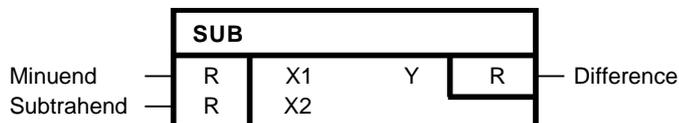
可在线插入	是
特别注意点	-

2.23 SUB 减法器 (REAL型)

SIMOTION

SINAMICS

符号



简要说明

- 支持两个REAL型输入的减法器

操作模式

- 该功能块从连接X1的输入值中减去连接X2的输入值（区分正负）。在Y输出端输出-3.402823 E38到3.402823 E38范围内的结果。
- $Y = X1 - X2$

多个块连接

块 连接	说明	预分配	取值范围	数
X1	被减数	0.0	REAL	
X2	减数	0.0	REAL	
Y	差	0.0	REAL	

据

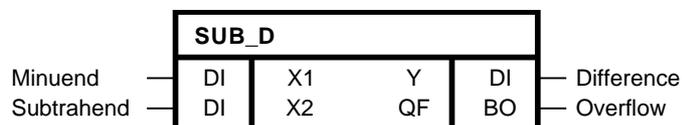
可在线加载	是
特别注意点	-

2.24 SUB_D 减法器 (DOUBLE INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持两个Double Integer型输入的减法器

操作模式

该功能块从连接X1的输入值中减去连接X2的输入值（区分正负）。结果值的范围为 -2147483648 (2^{31})到 $+2147483647$ ($2^{31}-1$), 并在Y输出端输出。当二进制输出QF=1时, 会发生溢位。

$$Y = X1 - X2$$

多个块连接

块 连接	说明	预分配	取值范围	数
X1	被减数	0	DINT	
X2	减数	0	DINT	
Y	差	0	DINT	
QF	溢位	0	0/1	

据

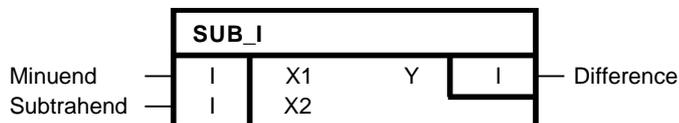
可在线加载	是
特别注意点	-

2.25 SUB_I 减法器 (INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持两个Integer型输入的减法器

操作模式

- 该功能块从连接X1的输入值中减去连接X2的输入值（区分正负）。在输出端Y输出大约在-32768到+32767范围内的结果。
- $Y = X1 - X2$

多个块连接

块 连接	说明	预分配	取值范围	数
X1	被减数	0	INT	
X2	减数	0	INT	
Y	差	0	INT	

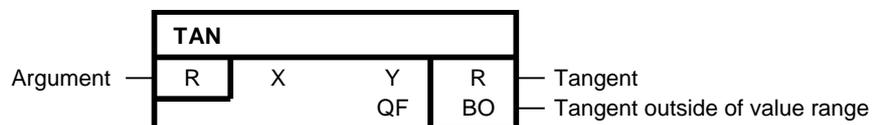
据

可在线加载	是
特别注意点	-

2.26 TAN 正切

 SIMOTION SINAMICS

符号



简要说明

- 确定角的正切值

操作模式

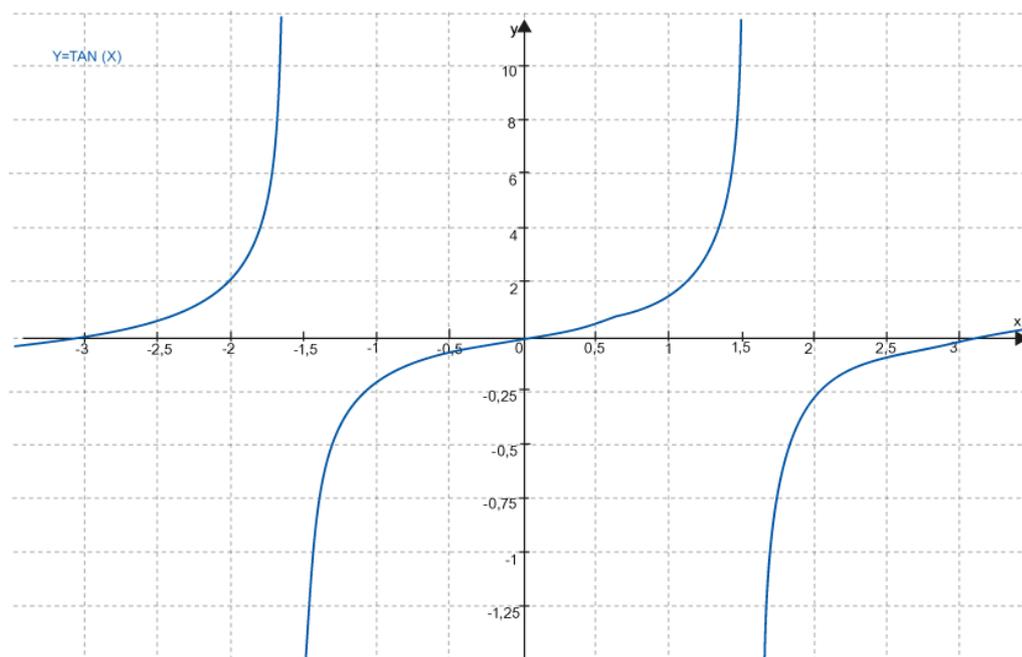
该功能块通过弧度法确定X输入端自变量的相应正切值，并在Y输出端输出该结果值。

$$Y = \tan X$$

输出范围：-3.402823 E38到3.402823 E38

当确定的正切值超出-3.402823 E38到3.402823 E38的范围时，块输出Y设为-3.402823 E38或+3.402823 E38，同时设置二进制输出QF = 1。

传输功能



2.26 TAN 正切

多个块连接

块 连接	说明	预分配	取值范围	数
X	自变量	0.0	REAL	
Y	正切	0.0	REAL	
QF	取值范围外的正切	0	0/1	

据

可在线插入	是
特别注意点	-

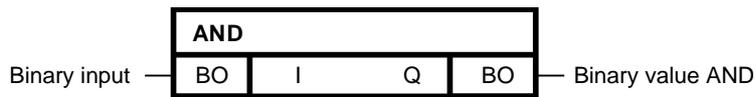
Logic

3.1 AND 逻辑AND运算（BOOL型）

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个BOOL型输入的AND功能块

操作模式

该功能块将I1到I4输入端的二进制值进行逻辑AND运算，并在其二进制输出Q处输出结果。

$$Q = I_{01} \wedge \dots \wedge I_{04}$$

当I1到I4的所有一般输入值为1时，输出Q = 1；其他情况下，输出Q = 0。

真值表

Input				Output
I01	I02	I03	I04	Q
0	*	*	*	0
*	0	*	*	0
*	*	0	*	0
*	*	*	0	0
1	1	1	1	1

***8 user-defined**

多个块连接

块 连接	说明	预分配	取值范围	数
I	二进制输入	1	0/1	
Q	二进制AND值	0	0/1	

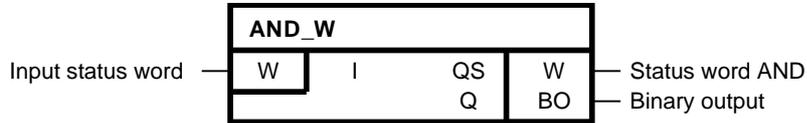
据

可在线加载	是
特别注意点	I包含最多4个连接 (I1到I4)

3.2 AND_W 逻辑AND运算（WORD型）

 SIMOTION SINAMICS

符号



简要说明

- 支持最多4个WORD型输入的AND_W功能块

操作模式

16个二进制状态合成于一个状态字中。

该功能块根据逻辑AND函数逐位合成I₀₁到I₁₆的状态字。然后在功能块QS输出端设置AND状态字的相应各位。

以下适用于AND状态字的第k位：

$$QS_k = I_{01_k} \wedge \dots \wedge I_{nn_k}, k = 1 \dots 16$$

当功能块输入端I1到I4中有一个及以上的等效位等于0时，状态字的一位等于0。

当AND状态字有一个及以上的位等于1时，二进制输出Q为1。

以下为三个输入的状态图

Status word



Status word

&



Status word

&



Status word



多个块连接

块 连接	说明	预分配	取值范围	数
I	输入状态字	16#FFFF	WORD	
QS	AND状态字	16#0000	WORD	
Q	二进制输出	0	0/1	

据

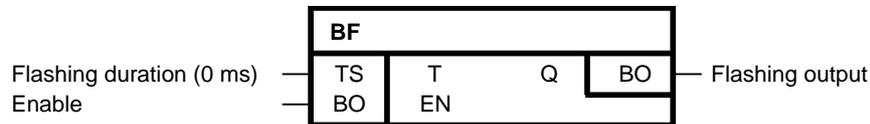
可在线加载	是
特别注意点	I包含最多4个输入 (I1到I4)

3.3 BF 闪烁功能块 (BOOL型)

SIMOTION

SINAMICS

符号



简要说明

BOOL型功能块

- 用于控制信号编码器
- 如时钟发生器一样

操作模式

当输入EN = 1时，该功能块按照时间间隔T交替设置输出Q的值为1和0。

当设置输入EN = 0时，那么输出Q = 0。

此时，T既是闪烁时长，又是黑暗时长。

说明

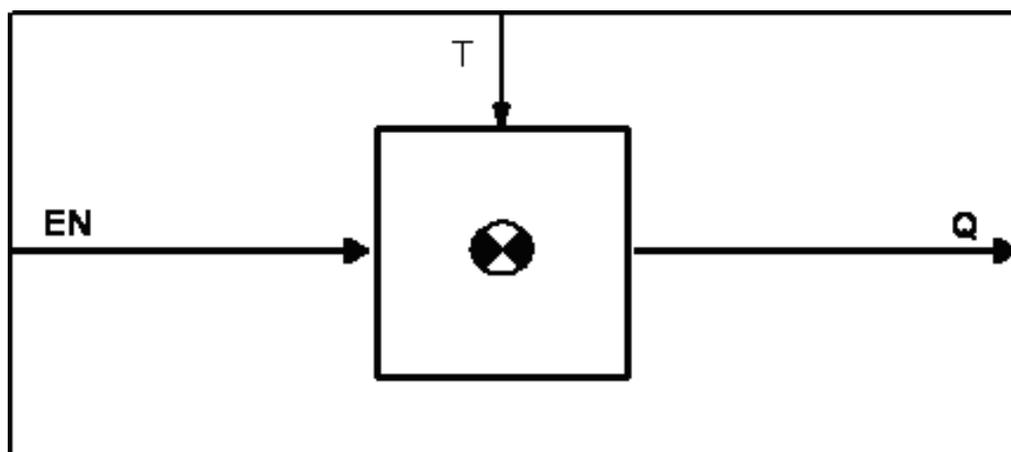
闪烁发生在基本周期时钟的倍数时间。如果功能块BF的输入值不是其运行的基本周期时钟的倍数，那么该输入值四舍五入为基本周期时钟。改变基本周期时钟或将闪烁时长T设定为基本周期时钟的倍数。

说明

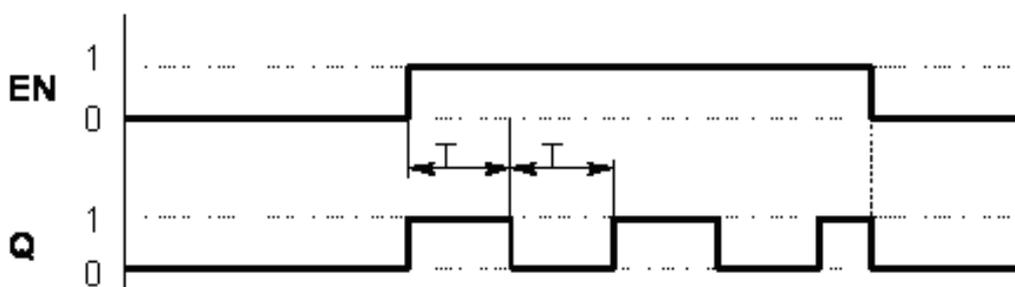
DCC SINAMICS: 如果您在以下执行组中使用该功能块，那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

3.3 BF 闪烁功能块 (BOOL型)

方框图



时序图



闪烁脉冲Q的值取决于闪烁时长T和使能输入EN。

多个块连接

块 连接	说明	预分配	取值范围	数
T	闪烁时长 (0 毫秒)	0	SDTIME	
EN	使能	0	0/1	
Q	闪烁输出	0	0/1	

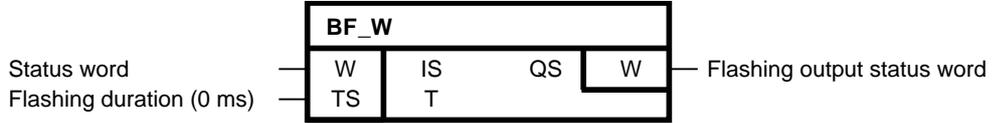
据

可在线加载	是
特别注意点	-

3.4 BF_W 状态字闪烁功能块 (BOOL型)

SIMOTION SINAMICS

符号



简要说明

- 控制传感器组合的WORD型功能块

操作模式

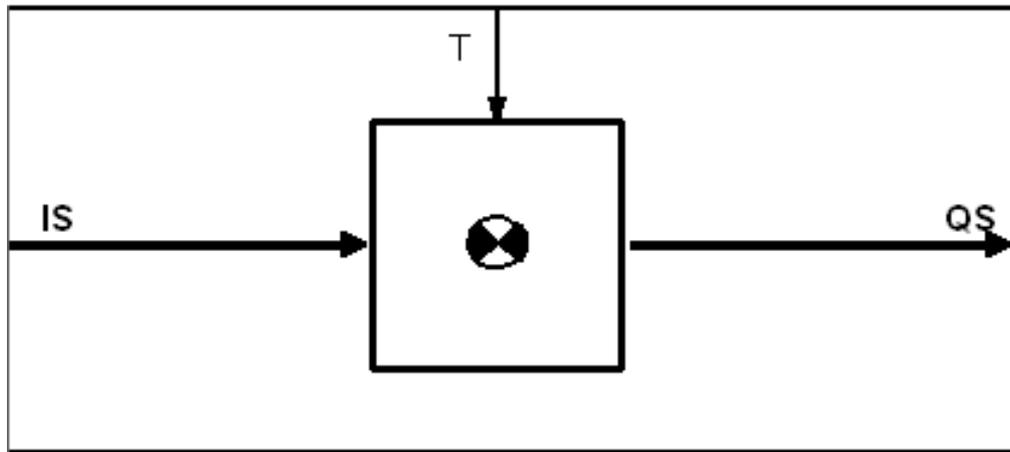
该功能块按照时间间隔T在输出状态字QS处，交替设置值为逻辑1的输入状态字IS的所有位为1和0。

此时，T既是闪烁时长，又是黑暗时长。

说明

闪烁发生在基本周期时钟的倍数时间。如果功能块BF_W的输入值不是其运行的基本周期时钟的倍数，那么该输入值四舍五入为基本周期时钟。改变基本周期时钟或将闪烁时长T设定为基本周期时钟的倍数。

方框图



多个块连接

块 连接	说明	预分配	取值范围	数
IS	状态字	16#0000	WORD	

3.4 BF_W 状态字闪烁功能块 (BOOL型)

块 连接	说明	预分配	取值范围	数
T	闪烁时长 (0 毫秒)	0	SDTIME	
QS	闪烁输出状态字	16#0000	WORD	

据

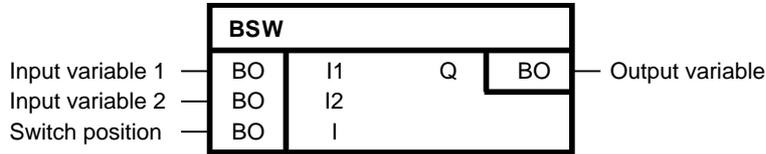
可在线加载	是
特别注意点	-

3.5 BSW BOOL型二位切换开关

SIMOTION

SINAMICS

符号



简要说明

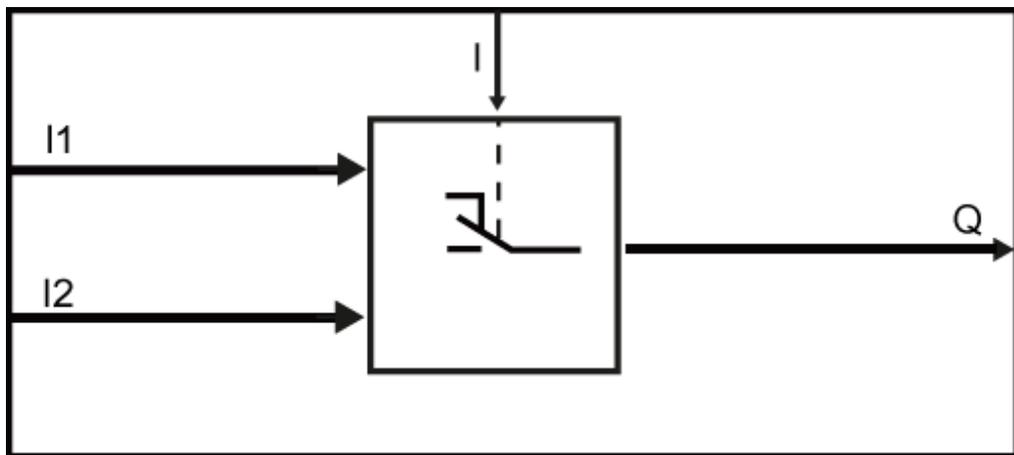
- 该功能块将两个二进制输入变量中的一个转化为输出。

操作模式

当输入I = 0时，输出Q为I1。

当输入I = 1时，输出Q为I2。

方框图



真值表

转换位置1	输出变量Q
0	Q = I1
1	Q = I2

初始化

当输入I = 0时，输出Q为I1。

3.5 BSW BOOL型二位切换开关

当输入I = 1时，输出Q为I2。

多个块连接

块 连接	说明	预分配	取值范围	数
I1	输入变量1	0	0/1	
I2	输入变量2	0	0/1	
I	转换位置	0	0/1	
Q	输出变量	0	0/1	

据

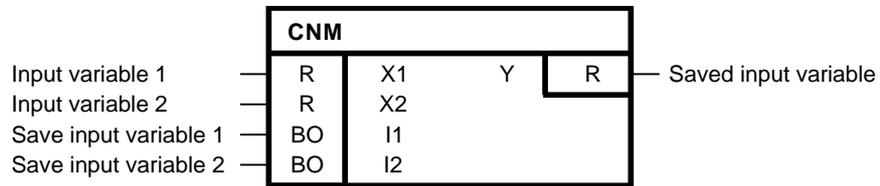
可在线加载	是
特别注意点	-

3.6 CNM 可控数字存储器 (REAL型)

SIMOTION

SINAMICS

符号



简要说明

用于存储当前输入值（采样和保持功能块），支持

- 可选输入
- 可选保存时间
- 上升沿启动触发

CNM_I和CNM_D功能块具备相同的功能。二者的唯一区别在于使用的数据类型。

操作模式

对于I1的上升沿，X1接到输出Y。

对于I2的上升沿，X2接到输出Y。

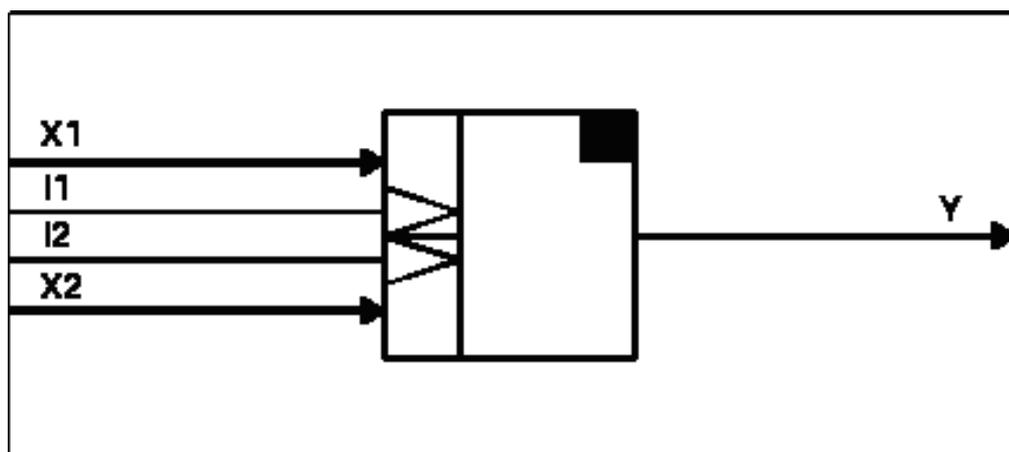
在下一个I1或I2的上升沿接到下一个瞬时值前，已存输入变量值会一直在Y处待定。

当I1和I2为同时上升沿时，I1优先，X1连到Y。

初始化

在上游输出初始化过程中，当输入I1或I2取值为1时，该功能块在第一个周期内无法检测正沿。否则，该功能块在第一个周期内检测正沿。在START模式（沿触发标志）下，I1和I2的值暂时储存。

方框图



真值表

输入		触发时的输出Y
I1	I2	
*	*	$Y_n = Y_{n-1}$
*	0 → 1	$Y_n = X2_n$
0 → 1	*	$Y_n = X1_n$
0 → 1	0 → 1	$Y_n = X1_n$

*: 无上升沿

0 → 1: 上升沿

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0.0	REAL	
X2	输入变量2	0.0	REAL	
I1	保存输入变量1	0.0	0/1	
I2	保存输入变量2	0.0	0/1	
Y	已存输入变量	0.0	REAL	

据

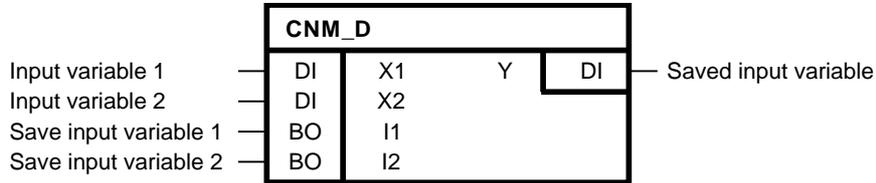
可在线加载	是
特别注意点	-

3.7 CNM_D 可控数字存储器 (DOUBLE INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

用于存储当前输入值（采样和保持功能块），支持

- 可选输入
- 可选保存时间
- 上升沿启动触发

CNM和CNM_i功能块具备相同的功能。二者的唯一区别在于使用的数据类型。

操作模式

对于I1的上升沿，X1接到输出Y。

对于I2的上升沿，X2接到输出Y。

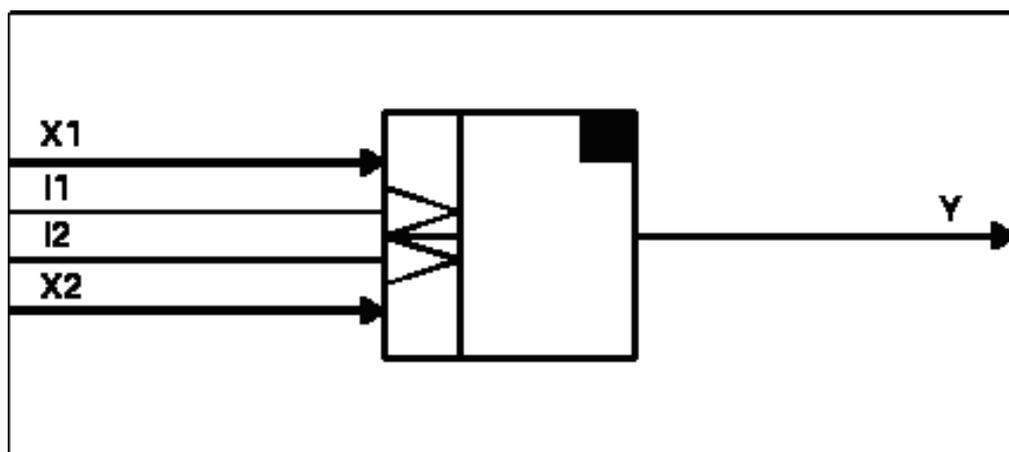
在下一个I1或I2的上升沿接到下一个瞬时值前，已存输入变量值会一直在Y处待定。

当I1和I2为同时上升沿时，I1优先，X1连到Y。

初始化

在上游输出初始化过程中，当输入I1或I2取值为1时，该功能块在第一个周期内无法检测正沿。该功能块在第一个周期内检测正沿。在START模式下，I1和I2的值暂时储存。

方框图



真值表

输入		触发时的输出Y
I1	I2	
*	*	$Y_n = Y_{n-1}$
*	0 → 1	$Y_n = X2_n$
0 → 1	*	$Y_n = X1_n$
0 → 1	0 → 1	$Y_n = X1_n$

*: 无上升沿

0 → 1: 上升沿

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	DINT	
X2	输入变量2	0	DINT	
I1	保存输入变量1	0	0/1	
I2	保存输入变量2	0	0/1	
Y	已存输入变量	0	DINT	

据

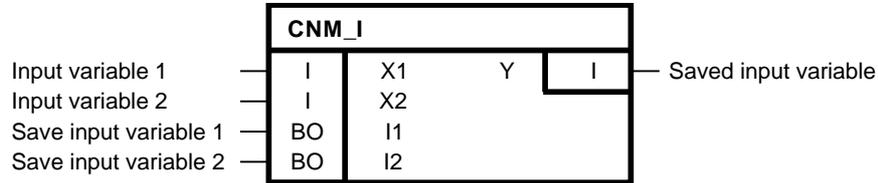
可在线加载	是
特别注意点	-

3.8 CNM_I 可控数字存储器（INTEGER型）

☑ SIMOTION

☑ SINAMICS

符号



简要说明

用于存储当前输入值（采样和保持功能块），支持

- 可选输入
- 可选保存时间
- 上升沿启动触发

CNM和CNM_D功能块具备相同的功能。二者的唯一区别在于使用的数据类型。

操作模式

对于I1的上升沿，X1接到输出Y。

对于I2的上升沿，X2接到输出Y。

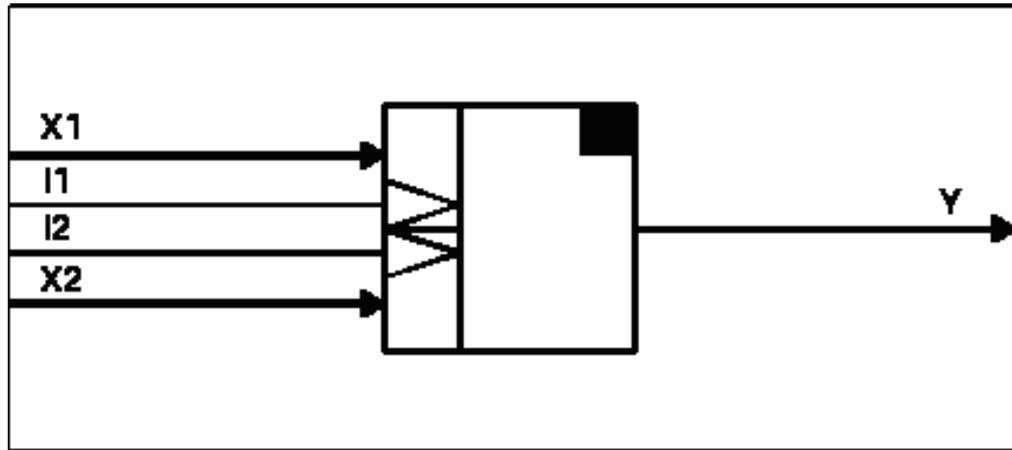
在下一个I1或I2的上升沿接到下一个瞬时值前，已存输入变量值会一直在Y处待定。

当I1和I2为同时上升沿时，I1优先，X1连到Y。

初始化

在上游输出初始化过程中，当输入I1或I2取值为1时，该功能块在第一个周期内无法检测正沿。该功能块在第一个周期内检测正沿。在START模式下，I1和I2的值暂时储存。

方框图



真值表

输入		触发时的输出Y
I1	I2	
*	*	$Y_n = Y_{n-1}$
*	0 → 1	$Y_n = X2_n$
0 → 1	*	$Y_n = X1_n$
0 → 1	0 → 1	$Y_n = X1_n$

*: 无上升沿

0 → 1: 上升沿

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	INT	
X2	输入变量2	0	INT	
I1	保存输入变量1	0	0/1	
I2	保存输入变量2	0	0/1	
Y	已存输入变量	0	INT	

据

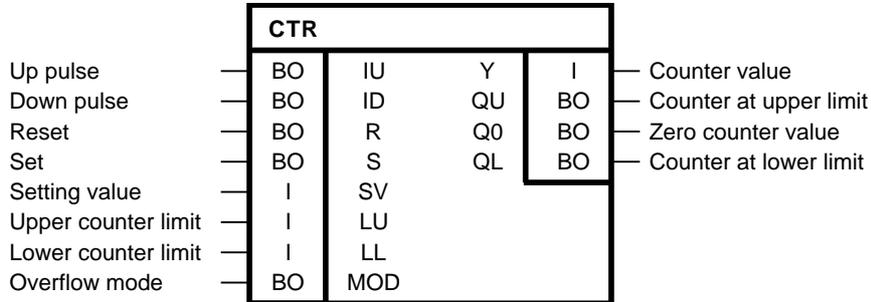
可在线加载	是
特别注意点	-

3.9 CTR 计数器 (BOOL型)

SIMOTION

SINAMICS

符号



简要说明

加减计数功能块，支持以下计数功能：

- 设计数为0
- 保持计数器为0（禁用）
- 设置计数器为初始值

独立设置加减计数的限值。

操作模式

该功能块形成一个沿触发的加减计数器。当输入IU的脉冲为上升沿时，计数值增加。

当输入ID的脉冲为上升沿时，计数值减少。计数值在输出端Y输出。控制计数器（参见真值表）当S=1时，计数值Y可预设为设定值SV。

然而，复位输入R优先于设置输入。只要R的值为逻辑1，Y则一直为0。当LL到LU之间的计数范围不包含0时，计数器锁定；当R = 1时，输出设置为有效限值。

计数器的工作范围通过LU（计数上限）和LL（计数下限）得到明确。

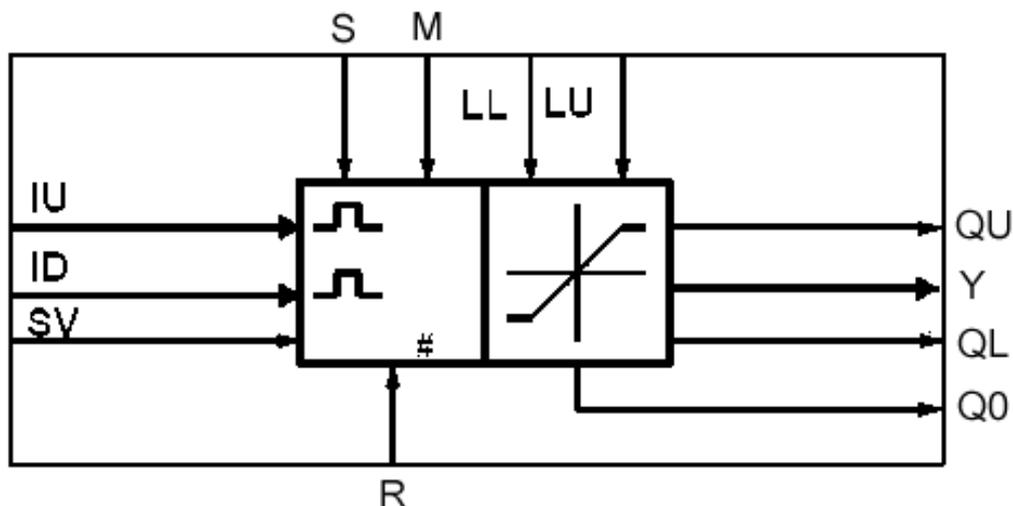
设定值（SV）的取值范围为 $LL \geq SV \geq LU$ 。

MOD=0	当达到限值时，计数器停止计算，且取值设置为显示QU（计数器上限值）或QL（计数器下限值）。
MOD=1	当达到上限值（LU），下一个加脉冲发生时，计数值设置为下限值；QU = 1表示周期内的正溢位。
	当达到下限值（LL），下一个减脉冲发生时，计数值设置为上限值；QL = 1表示周期内的负溢位。

当计数值为0时，输出Q0为1。

3.9 CTR 计数器 (BOOL型)

方框图



真值表

二进制指令	二进制指令	计数值Y
S	R	
0	0	保留Y
0	1	复位Y
1	0	Y = SV (设定值)
1	1	复位Y

设置/复位指令发出时的计数值

初始化

初始化定义了第一个周期的起始值。当输入ID或IU预设为1时，该功能块在第一个周期内无法检测正沿。

限制条件:

- $LL \leq Y \leq LU$ for $LL < LU$
- $Y = LU$ for $LL \geq LU$

多个块连接

块 连接	说明	预分配	取值范围	数
IU	加脉冲	0	0/1	
ID	减脉冲	0	0/1	
R	复位	0	0/1	
S	设定	0	0/1	
SV	设定值	0	INT	
LU	计数器上限值:	0	INT	
LL	计数器下限值:	0	INT	

块 连接	说明	预分配	取值范围	数
MOD	溢位模式	0	0/1	
Y	计数值	0	INT	
QU	计数器位于上限值	0	0/1	
Q0	零计数值	0	0/1	
QL	计数器位于低限	0	0/1	

据

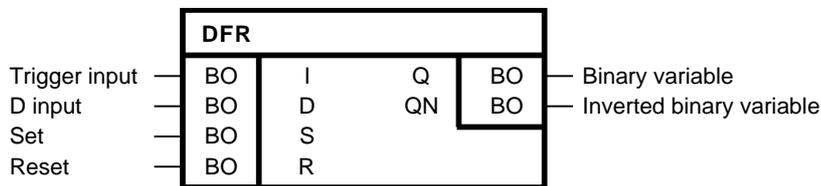
可在线加载	是
特别注意点	-

3.10 DFR 复位优先D型触发器 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 用作复位优先D型触发器的BOOL型功能块

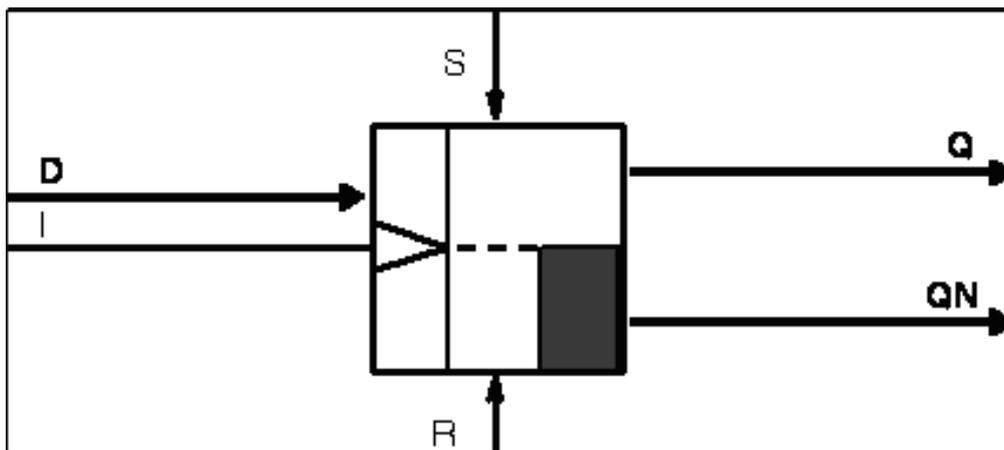
操作模式

当两个输入S和R的值为逻辑0时，输入D接到位于触发脉冲输入I的上升沿处的输出Q。输出QN的值总是与Q相反。当输入S的值为逻辑1时，输出Q的值为逻辑1。当输入R的值为逻辑1时，输出Q的值为逻辑0。当两个输入的值都为逻辑0时，输出Q则保持不变。然而，当两个输入S和R的值都为逻辑1时，由于复位输入处于优先地位，输出Q的值为逻辑0。

初始化

在上游输出初始化过程中，当输入I取值为1时，该功能块在第一个周期内无法检测正沿。该功能块在第一个周期内检测正沿。在START模式下，I的值暂时储存。

方框图

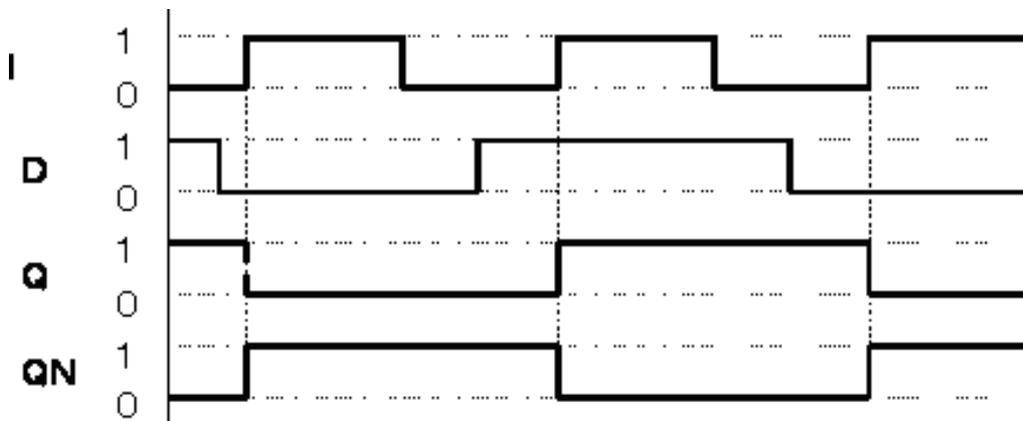


真值表

D	I	二进制指令		输出状态	
		S	R	Q	QN
0	0 → 1	0	0	0	1
1	0 → 1	0	0	1	0
*	1 → 0	0	0	Q_{n-1}	Q_{n-1}
*	*	0	1	0	1
*	*	1	0	1	0
*	*	1	1	0	1

时序图

相对于D和I



由于 $S = R = 0$ ，输出脉冲Q取决于D输入和输入脉冲I。

多个块连接

块 连接	说明	预分配	取值范围	数
I	触发脉冲输入	0	0/1	
D	D输入	0	0/1	
S	设定	0	0/1	
R	复位	0	0/1	
Q	二进制变量	0	0/1	
QN	反二进制变量	1	0/1	

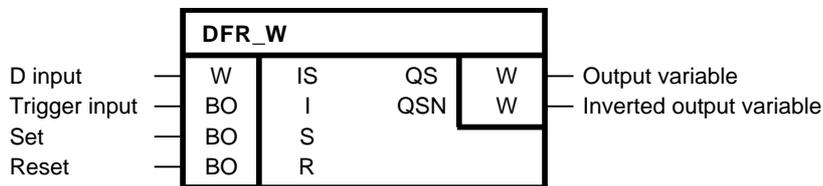
据

可在线加载	是
特别注意点	-

3.11 DFR_W 复位优先D型触发器 (WORD型)

SIMOTION SINAMICS

符号



简要说明

- 用作复位优先D型触发器的WORD型功能块

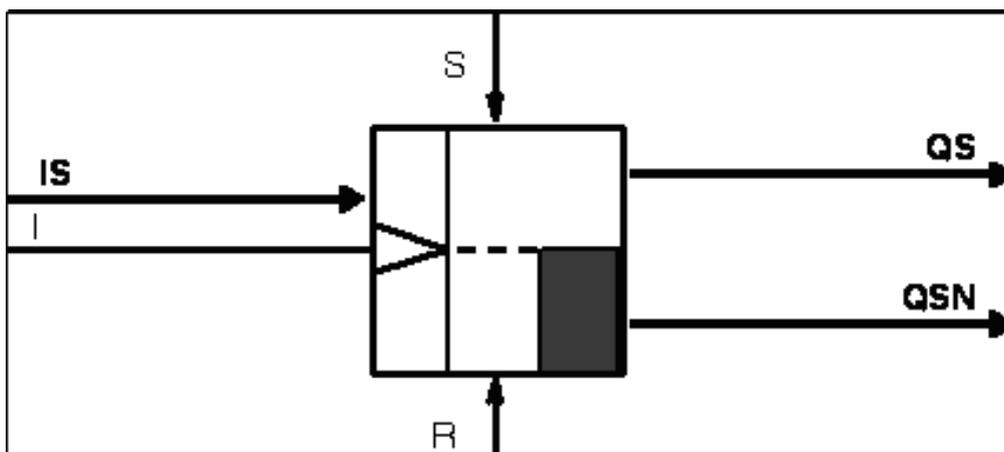
操作模式

当两个输入S和R值为逻辑0时，输入D接到位于触发脉冲输入I的上升沿处的输出QS。输出QSN的值总是与QS相反。当S = 1时，输出变量QS的所有位都为1。当R = 1时，输出变量QS的所有位都为0。当两个输入S和R的值都为逻辑0时，输出QS则保持不变。当两个输入S和R的值都为逻辑1时，由于复位输入处优先地位，输出变量QS的所有位都为0。

初始化

在上游输出初始化过程中，当输入I取值为1时，该功能块在第一个周期内无法检测正沿。该功能块在第一个周期内检测正沿。在START模式下，I的值暂时储存。

方框图



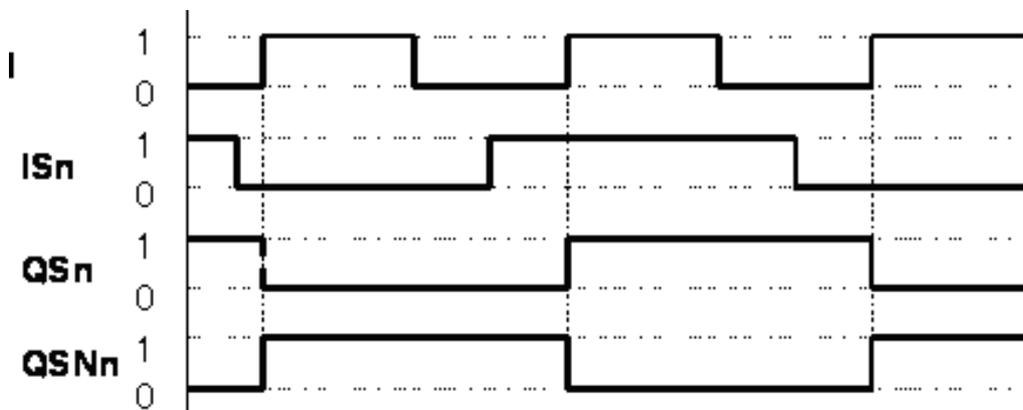
真值表

I	二进制指令		输出状态	
	S	R	QS	QSN
0 → 1	0	0	IS	反相IS
*	0	1	0	1
*	1	0	1	0
*	1	1	0	1

* 不限

时序图

相对于I和IS



由于 $S = R = 0$ ，输出变量QS和QSN（n表示位数）取决于触发脉冲输入I和D输入IS。

多个块连接

块 连接	说明	预分配	取值范围	数
IS	D输入	16#0000	WORD	
I	触发脉冲输入	0	0/1	
S	设定	0	0/1	
R	复位	0	0/1	
QS	输出变量	16#0000	WORD	
QSN	反相输出变量	16#FFFF	WORD	

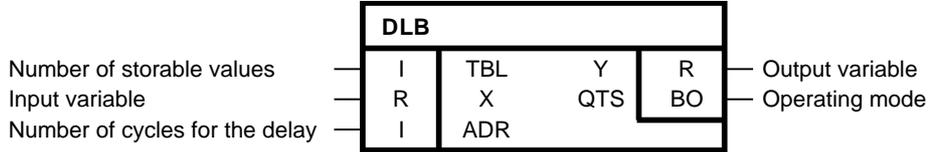
据

可在线加载	是
特别注意点	-

3.12 DLB 延迟元素 (REAL型)

SIMOTION SINAMICS

符号



简要说明

针对输入变量输出的REAL型功能块，输入变量会延迟可指定的采样时间。

操作模式

当运算状态为QTS = 1时，该功能块包含TBL变量的延迟存储器。延迟之后，输入X处的输入变量输出为输出变量Y。延迟时长由采样时间（计算功能块的时间片）的整数倍ADR确定。当运算状态为QTS = 0时，延迟存储器不会激活。此时，输入X处的输入变量立即输出为输出变量Y。

初始化

在初始化过程中，会请求延迟存储器来获取TBL输入变量。延迟存储器可最多包含1000个值。当TBL < 0时，TBL限于 0。QTS = 1表明TBL请求的延迟存储器可用。QTS = 0表明系统由于缺少资源没能使该存储器可用或者定义了一个大于1000的TBL值。此时，在循环运算过程中，输出Y校正为输入X。

说明

初始化过程中，请求的延迟存储器与ADR无关，全部取值为输入变量X的值。这样的话，即使在ADR延迟之后，运算过程中有效值在Y处依然可用。ADR延迟的值受限于TBL存储器大小（0 ≤ ADR ≤ TBL）。ADR延迟的值受限于存储器变量TBL。由于在运算时TBL不能动态改变，因此应该在配置时就为TBL明确最大需求延迟，当前延迟值应该通过ADR进行动态设置。

多个块连接

块 连接	说明	预分配	取值范围	数
TBL	可存储数值量	100	0...1000	
X	输入变量	0.0	REAL	
ADR	延迟的周期数	0	0...1000	
Y	输出变量	0.0	REAL	
QTS	运算状态	0	0/1	

据

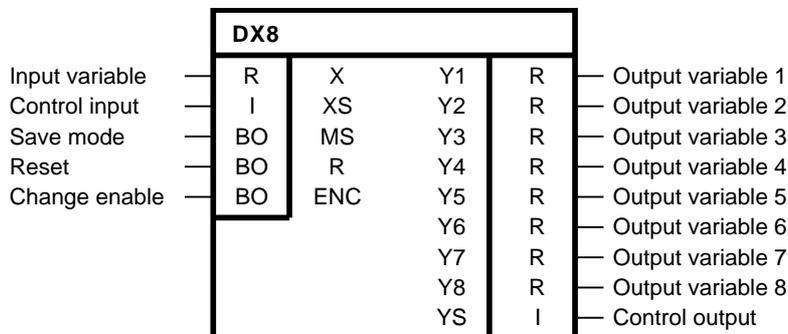
可在线加载	是
特别注意点	-

3.13 DX8 多路输出选择器，8个输出，可级联（REAL型）

☑ SIMOTION

☑ SINAMICS

符号



简要说明

解多工运算的REAL型功能块 该功能块可级联。

操作模式

根据ENC、R、MS和XS = 1到8，该功能块连接输入X到8个可选输出Y1到Y8中的一个（例如：XS = 3表示Y3 = X）。

当XS = 0或XS >= 9时，功能块输入Y1到Y8中的任何一个都不会被选择。非选择的输出不是设为0，就是保留其下次变化前的最后值。

以下优先序位适用于所有的控制输入：

ENC > R > MS

当ENC = 0时，所有Y1到Y8的输出都保持不变，与R和MS无关。

当ENC = 1时，所有Y1到Y8的输出都可以改变。

当R = 1时，所有Y1到Y8的输出都取值为0，与MS无关。

当MS = 0（非闭塞状态）时，所有XS未选择的Y1到Y8的输出都取值为0。

当MS = 1（存储模式）时，所有XS未选择的输出都保持不变。

3.13 DX8 多路输出选择器，8个输出，可级联（REAL型）

真值表

ENC	R	MS	XS	输出Y1到Y8
0	*	*	*	保留之前的值
1	1	*	*	Y1到Y8 = 0
1	0	0	$1 \leq XS \leq 8$	选择输出 = X 非选择输出 = 0
1	0	0	XS = 0或 XS \geq 9	Y1到Y8 = 0
1	0	1	$1 \leq XS \leq 8$	选择输出 = X 非选择输出保持不变
1	0	1	XS = 0或 XS \geq 9	所有之前的值都保持不变

级联

功能块输出YS必须与以下功能块的输入XS相连。

当XS = 0到8时，YS = 0

当XS > 8时：YS = XS-8

（用于级联）

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
XS	控制输入	0	INT	
MS	存储模式	0	0/1	
R	复位	0	0/1	
ENC	可以改变	0	0/1	
Y1	输出变量1	0.0	REAL	
Y2	输出变量2	0.0	REAL	
Y3	输出变量3	0.0	REAL	
Y4	输出变量4	0.0	REAL	
Y5	输出变量5	0.0	REAL	
Y6	输出变量6	0.0	REAL	
Y7	输出变量7	0.0	REAL	
Y8	输出变量8	0.0	REAL	
YS	控制输出	0	INT	

据

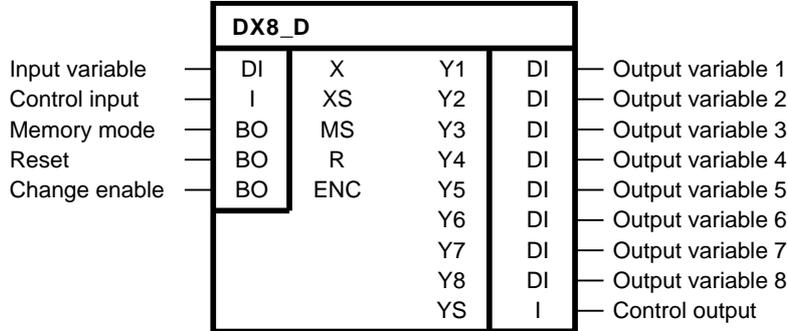
可在线加载	是
特别注意点	-

3.14 DX8_D 多路输出选择器，8个输出，可级联（DOUBLE INTEGER型）

3.14 DX8_D 多路输出选择器，8个输出，可级联（DOUBLE INTEGER型）

SIMOTION SINAMICS

符号



简要说明

- 解多工运算的DOUBLE INTEGER型功能块 该功能块可级联。

操作模式

根据ENC、R、MS和XS = 1到8，该功能块连接输入X到8个可选输出Y1到Y8中的一个（例如：XS = 3表示Y3 = X）。

当XS = 0或XS >= 9时，功能块输入Y1到Y8中的任何一个都不会被选择。非选择的输出不是设为0，就是保留其下次变化前的最后值。

以下优先序位适用于所有的控制输入：

ENC > R > MS

当ENC = 0时，所有Y1到Y8的输出都保持不变，与R和MS无关。

当ENC = 1时，所有Y1到Y8的输出都可以改变。

当R = 1时，所有Y1到Y8的输出都取值为0，与M无关。

当MS = 0（非闭塞状态）时，所有XS未选择的Y1到Y8的输出都取值为0。

当MS = 1（闭塞状态）时，所有XS未选择的输出都保持不变。

3.14 DX8_D 多路输出选择器，8个输出，可级联（DOUBLE INTEGER型）

真值表

ENC	R	MS	XS	输出Y1到Y8
0	*	*	*	保留之前的值
1	1	*	*	Y1到Y8 = 0
1	0	0	$1 \leq XS \leq 8$	选择输出 = X 非选择输出 = 0
1	0	0	XS = 0或 XS \geq 9	Y1到Y8 = 0
1	0	1	$1 \leq XS \leq 8$	选择输出 = X 非选择输出保持不变
1	0	1	XS = 0或 XS \geq 9	所有之前的值都保持不变

* 不限

当XS = 0到8时，YS = 0。

当XS > 8时：YS = XS - 8

（用于级联）

级联

功能块输出YS必须与以下功能块的输入XS相连。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
XS	控制输入	0	INT	
MS	存储器模式	0	0/1	
R	复位	0	0/1	
ENC	可以改变	0	0/1	
Y1	输出变量1	0	DINT	
Y2	输出变量2	0	DINT	
Y3	输出变量3	0	DINT	
Y4	输出变量4	0	DINT	
Y5	输出变量5	0	DINT	
Y6	输出变量6	0	DINT	
Y7	输出变量7	0	DINT	
Y8	输出变量8	0	DINT	
YS	控制输出	0	INT	

据

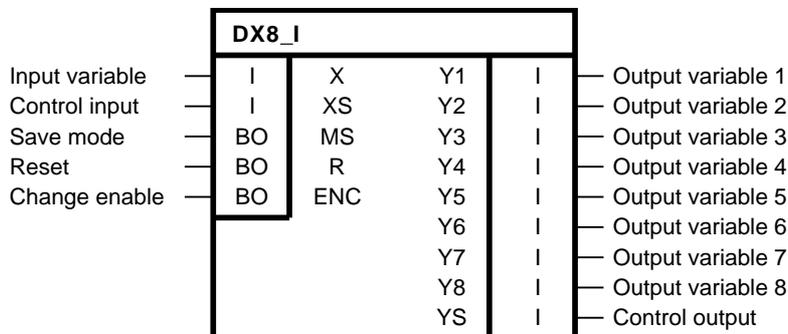
可在线插入	是
特别注意点	-

3.15 DX8_I 多路输出选择器，8个输出，可级联（INTEGER型）

☑ SIMOTION

☑ SINAMICS

符号



简要说明

解多工运算的INTEGER型功能块 该功能块可级联。

操作模式

根据ENC、R、MS和XS = 1到8，该功能块连接输入X到8个可选输出Y1到Y8中的一个（例如：XS = 3表示Y3 = X）。

当XS = 0或XS >= 9时，功能块输入Y1到Y8中的任何一个都不会被选择。非选择的输出不是设为0，就是保留其下次变化前的最后值。

以下优先序位适用于所有的控制输入：

ENC > R > MS

当ENC = 0时，所有Y1到Y8的输出都保持不变，与R和MS无关。

当ENC = 1时，所有Y1到Y8的输出都可以改变。

当R = 1时，所有Y1到Y8的输出都取值为0，与MS无关。

当MS = 0（非闭塞状态）时，所有XS未选择的Y1到Y8的输出都取值为0。

当MS = 1（存储模式）时，所有XS未选择的输出都保持不变。

真值表

ENC	R	MS	XS	输出Y1到Y8
0	*	*	*	保留之前的值
1	1	*	*	Y1到Y8 = 0
1	0	0	$1 \leq XS \leq 8$	选择输出 = X 非选择输出 = 0
1	0	0	XS = 0或 XS \geq 9	Y1到Y8 = 0
1	0	1	$1 \leq XS \leq 8$	选择输出 = X 非选择输出保持不变
1	0	1	XS = 0或 XS \geq 9	所有之前的值都保持不变

* 不限

当XS = 0到8时，YS = 0。

当XS > 8时：YS = XS - 8

（用于级联）

级联

功能块输出YS必须与以下功能块的输入XS相连。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
XS	控制输入	0	INT	
MS	存储模式	0	0/1	
R	复位	0	0/1	
ENC	可以改变	0	0/1	
Y1	输出变量1	0	INT	
Y2	输出变量2	0	INT	
Y3	输出变量3	0	INT	
Y4	输出变量4	0	INT	
Y5	输出变量5	0	INT	
Y6	输出变量6	0	INT	
Y7	输出变量7	0	INT	
Y8	输出变量8	0	INT	
YS	控制输出	0	INT	

据

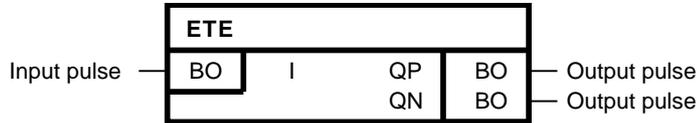
可在线加载	是
特别注意点	-

3.16 ETE 沿评估器 (BOOL型)

SIMOTION

SINAMICS

符号



简要说明

- 沿评估

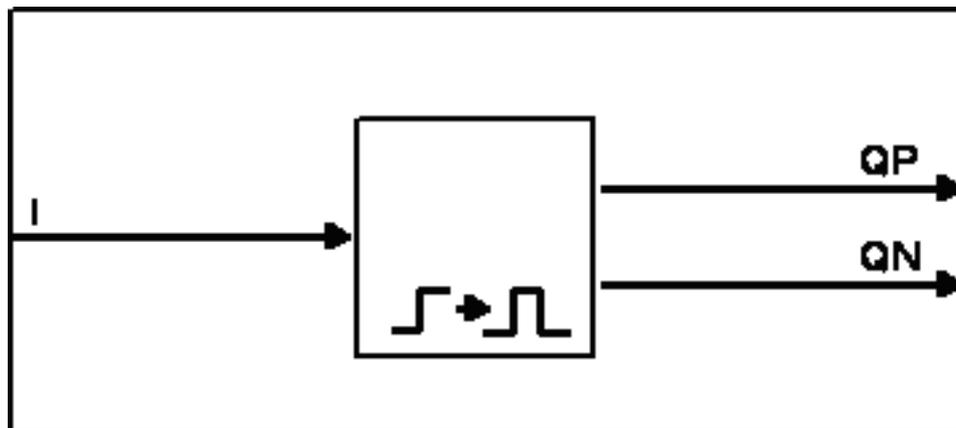
操作模式

该功能块检测输入I的信号变化。当输入I为正沿 (0→1) 时, 扫描时间TA的输出QP = 1。
当输入I为负沿沿 (1→0) 时, 扫描时间TA的输出QN = 1。

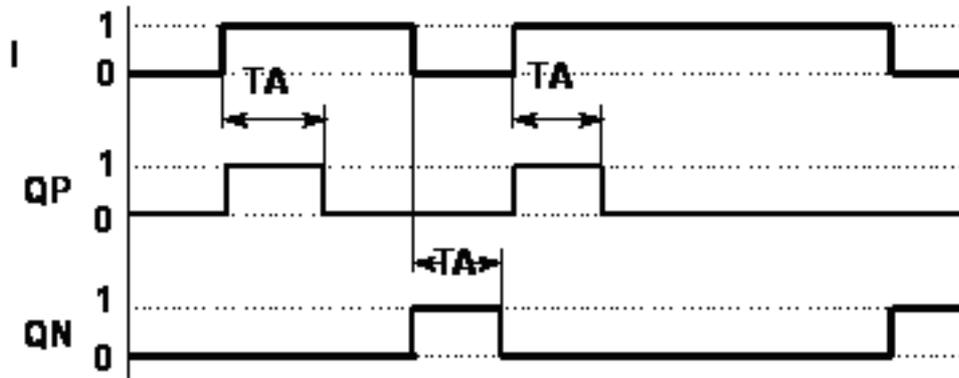
初始化

初始化定义了第一个周期的起始值。在上游功能块初始化过程中, 当输入I的取值为1时, 该功能块不可以在第一个周期检测正沿。在上游功能块初始化过程中, 当输入I的取值为0时, 该功能块不可以在第一个周期检测负沿。

方框图



时序图



Output pulses QP and QN as a function of scan time TA and input pulse I

多个块连接

块 连接	说明	预分配	取值范围	数
I	输入脉冲	0	0/1	
QP	输出脉冲	0	0/1	
QN	输出脉冲	0	0/1	

据

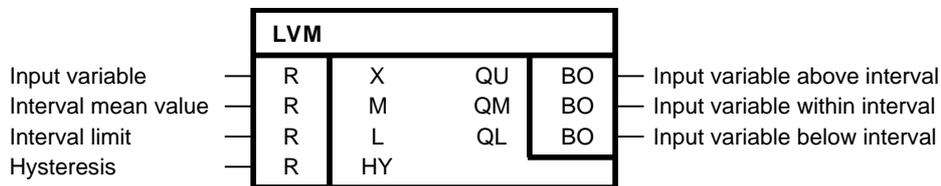
可在线加载	是
特别注意点	-

3.17 LVM 磁滞双面限制监视器 (BOOL型)

SIMOTION

SINAMICS

符号



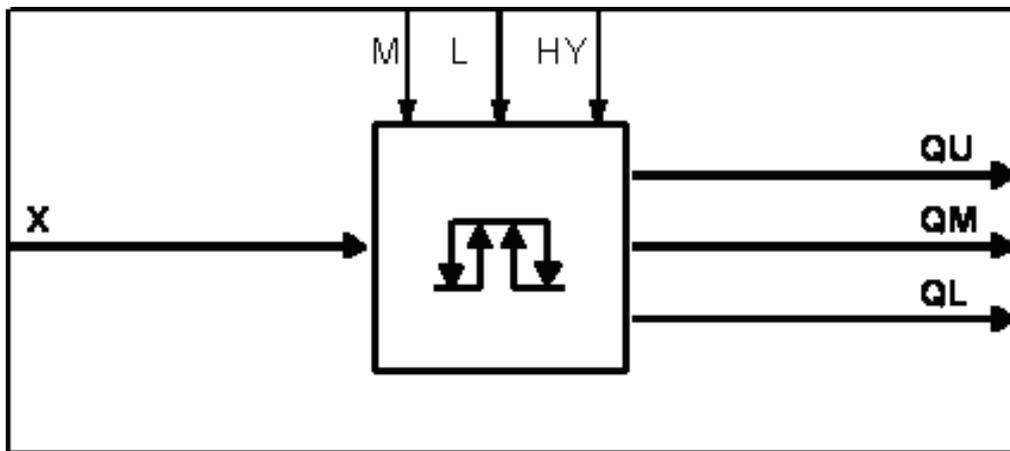
简要说明

- 该BOOL型功能块通过与可选基准变量相比较来监视输入变量。
- 可用于监视设定值、实际值和测量值以及抑制频繁转换。
- 该功能块支持窗口辨别功能。

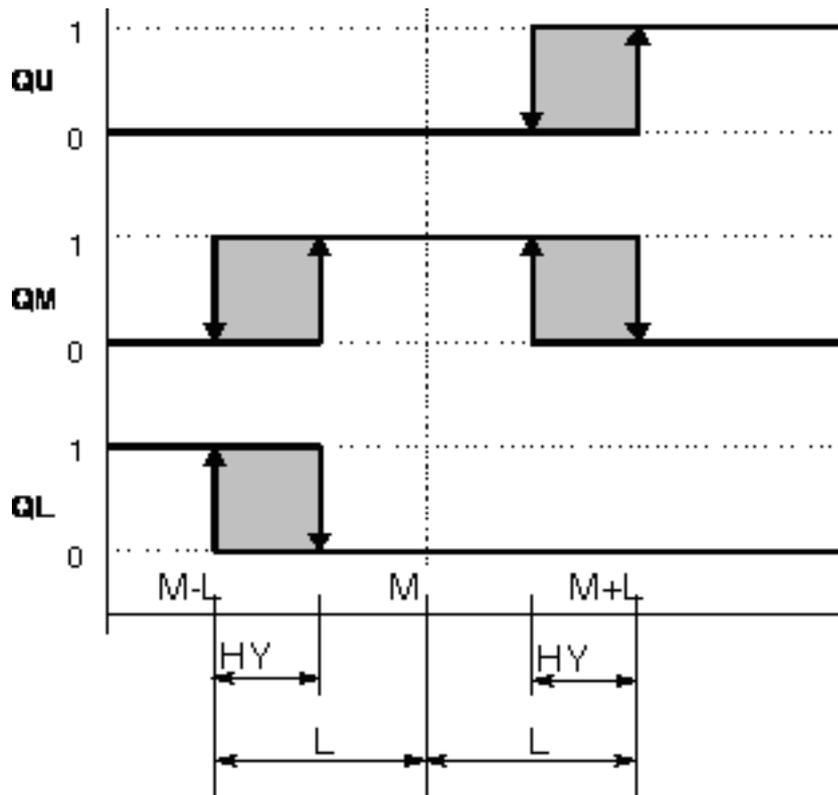
操作模式

该功能块基于带磁滞的传输特性（参见传输特性），计算中间值。该中间值会与间隔限制相比较，比较结果在输出QU、QM和QL处输出。该传输特性配置了平均值M、间隔限值L和磁滞值HY。

方框图



传输特性



多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
M	间隔平均值	0.0	REAL	
L	间隔限值	0.0	REAL	
HY	磁滞值	0.0	REAL	
QU	高于间隔的输入变量	0	0/1	
QM	间隔内的输入变量	0	0/1	
QL	低于间隔的输入变量	0	0/1	

据

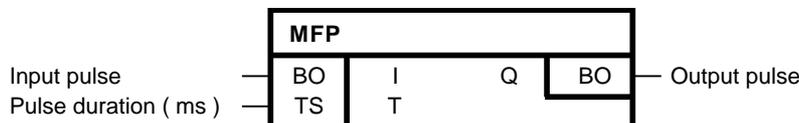
可在线加载	是
特别注意点	-

3.18 MFP 脉冲发生器 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 用于产生固定时长脉冲的定时器
- 用作脉冲收缩或脉冲展宽的单稳态触发器

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

针对脉冲时长T, 输入I处的脉冲上升沿设置输出Q为1。脉冲发生器不可以重新触发。当T=0时, 1个周期的脉冲时长生效。

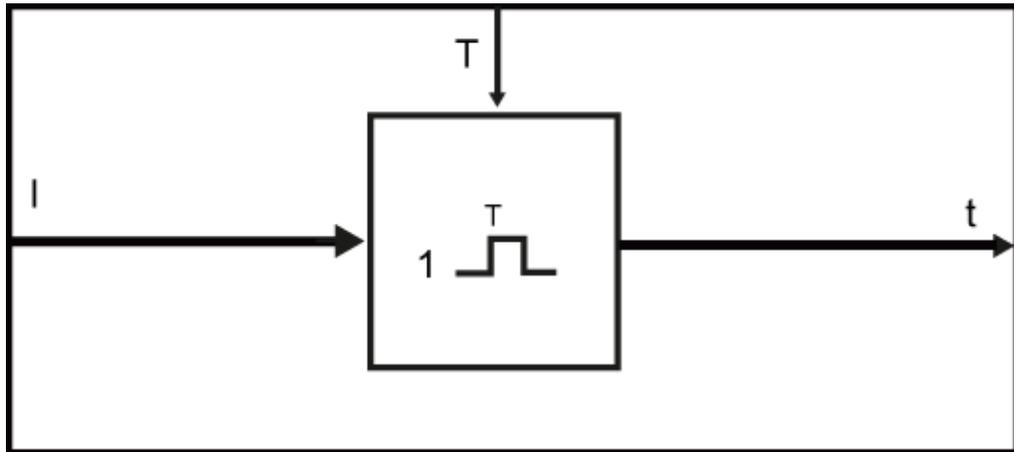
初始化

初始化定义了第一个周期的起始值。

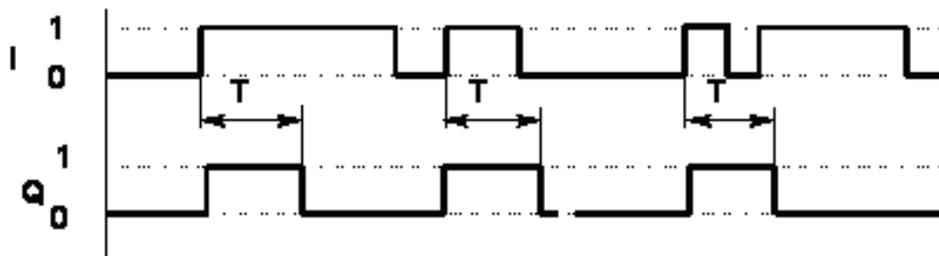
在上游功能块输出初始化过程中, 当输入I的取值为1时, 该功能块不可以在第一个周期检测正沿。

当输出Q取默认值1时, 那么在脉冲时长T内的初始化完成之后, 输出Q = 1。

方框图



时序图



Output pulse Q as a function of pulse duration T and input pulse I

多个块连接

块 连接	说明	预分配	取值范围	数
I	输入脉冲	0	0/1	
T	脉冲时长 (毫秒)	0	SDTIME	
Q	输出脉冲	0	0/1	

据

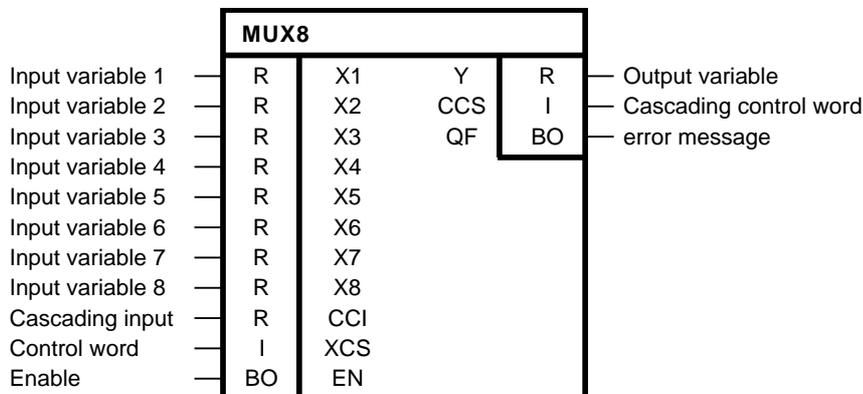
可在线加载	是
特别注意点	-

3.19 MUX8 多路复用器，可级联（REAL型）

SIMOTION

SINAMICS

符号



简要说明

8路复用运算的REAL型功能块 该功能块可级联。

操作模式

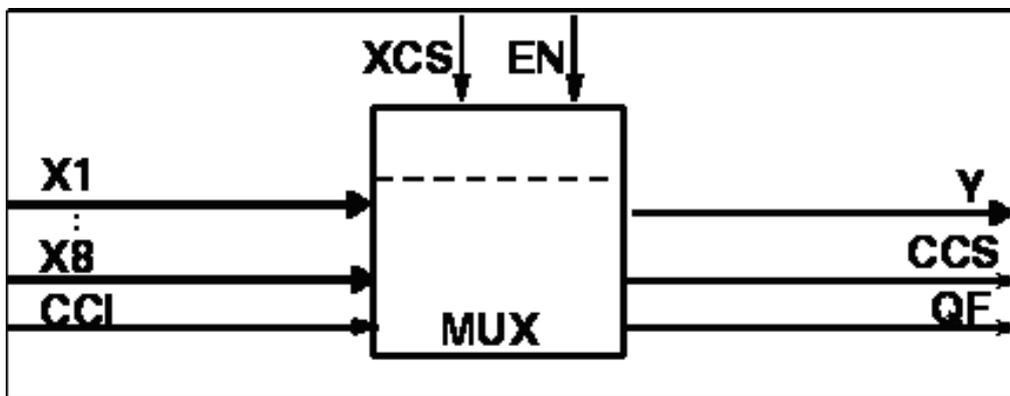
只要使能输入EN为逻辑0，该功能块就会在输出Y处输出级联输入CCI的值。

当EN的值为逻辑1时，只要16位的控制字XCS的取值介于1到8之间，输入变量X1到X8中的一个会连接到输出Y。

当输入XCS > 8时，输出Y取值为0，输出QF的值为逻辑1。级联控制字取值CCS = XCS-8，请参见真值表。

输出Y、CCS和QF可用于级联功能块。此时，第一个功能块的输出Y连接到下游多路复用器的输入CCI，连接输出CCS到下一个XCS，连接输出QF到下一个输入EN。

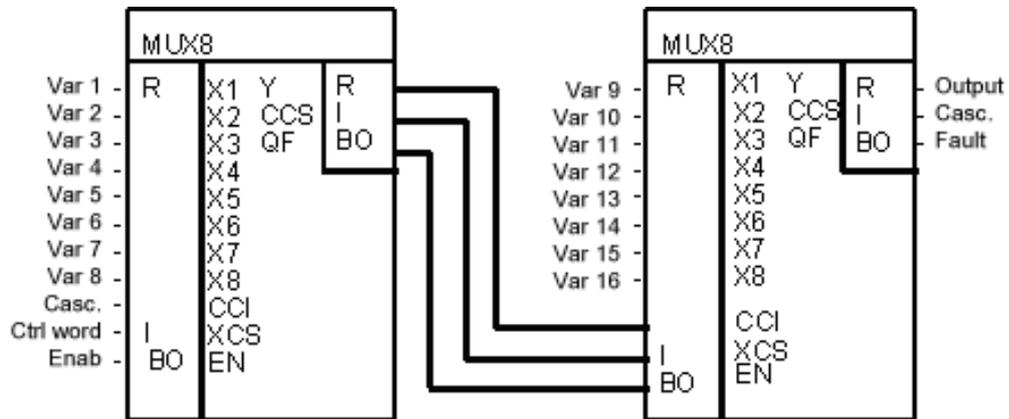
方框图



真值表

EN	XCS	Y	CSS	QF
0	不限	CCI	0	0
1	0	0	0	1
1	1	X1	0	0
1	2	X2	0	0
1	3	X3	0	0
1	4	X4	0	0
1	5	X5	0	0
1	6	X6	0	0
1	7	X7	0	0
1	8	X8	0	0
1	>8	0	XCS-8	1

级联



多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0.0	REAL	
X2	输入变量2	0.0	REAL	
X3	输入变量3	0.0	REAL	
X4	输入变量4	0.0	REAL	
X5	输入变量5	0.0	REAL	
X6	输入变量6	0.0	REAL	
X7	输入变量7	0.0	REAL	
X8	输入变量8	0.0	REAL	
CCI	级联输入	0.0	REAL	
XCS	控制字	0	0...32767	
EN	使能	0	0/1	
Y	输出变量	0.0	REAL	

3.19 MUX8 多路复用器，可级联（REAL型）

块 连接	说明	预分配	取值范围	数
CCS	级联控制字	0	0...32767	
QF	错误消息	0	0/1	

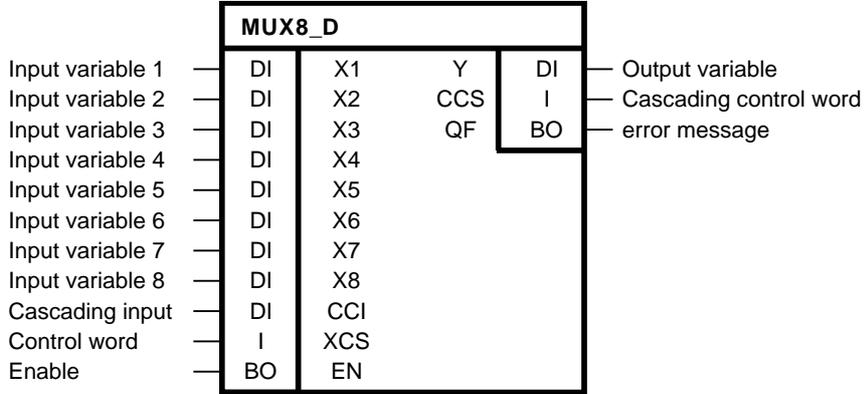
据

可在线加载	是
特别注意点	-

3.20 MUX8_D 多路复用器，可级联（DOUBLE INTEGER型）

SIMOTION SINAMICS

符号



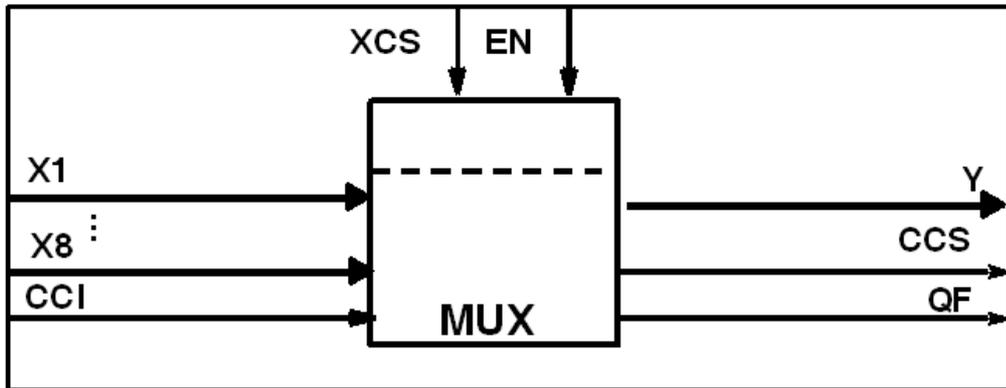
简要说明

- 8路运算的DOUBLE INTEGER型功能块 该功能块可级联。

操作模式

只要使能输入EN为逻辑0，该功能块就会在输出Y处输出级联输入CCI的值。当EN为逻辑1时，只要16位的控制字XCS的取值介于1到8之间，输入变量X1到X8中的一个会连接到输出Y。当输入XCS > 8时，输出Y取值为0，输出QF的值为逻辑1。级联控制字取值CCS = XCS-8，请参见真值表。输出Y、CCS和QF可用于级联功能块。此时，第一个功能块的输出Y连接到下游多路复用器的输入CCI，连接输出CCS到下一个XCS，连接输出QF到下一个输入EN。

方框图

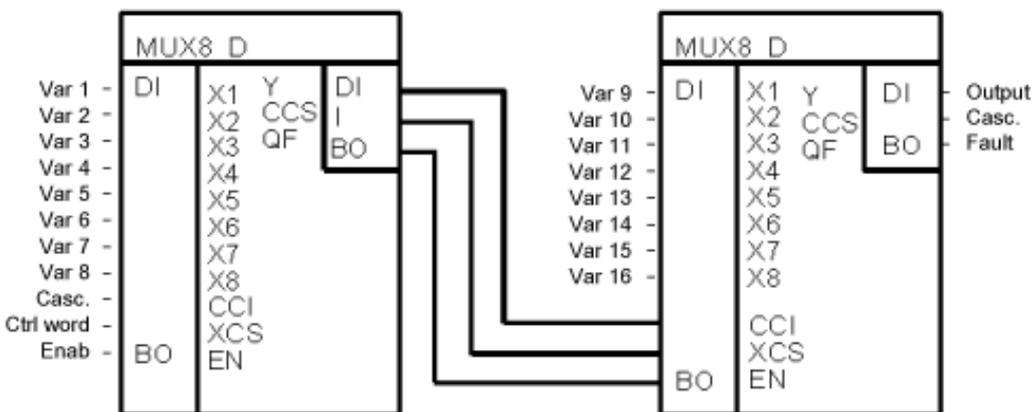


3.20 MUX8_D 多路复用器，可级联（DOUBLE INTEGER型）

真值表

EN	XCS	Y	CSS	QF
0	不限	CCI	0	0
1	0	0	0	1
1	1	X1	0	0
1	2	X2	0	0
1	3	X3	0	0
1	4	X4	0	0
1	5	X5	0	0
1	6	X6	0	0
1	7	X7	0	0
1	8	X8	0	0
1	>8	0	XCS-8	1

级联



多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	DINT	
X2	输入变量2	0	DINT	
X3	输入变量3	0	DINT	
X4	输入变量4	0	DINT	
X5	输入变量5	0	DINT	
X6	输入变量6	0	DINT	
X7	输入变量7	0	DINT	
X8	输入变量8	0	DINT	
CCI	级联输入	0	DINT	
XCS	控制字	0	0...32767	
EN	使能	0	0/1	
Y	输出变量	0	DINT	

3.20 MUX8_D 多路复用器，可级联（DOUBLE INTEGER型）

块 连接	说明	预分配	取值范围	数
CCS	级联控制字	0	0...32767	
QF	错误消息	0	0/1	

据

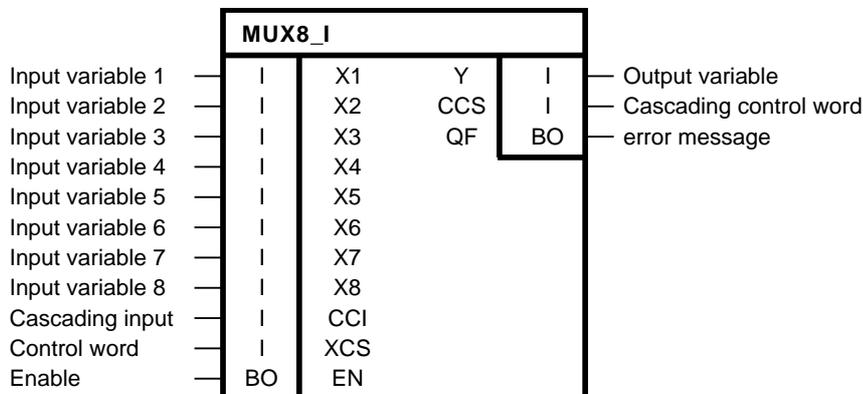
可在线插入	是
特别注意点	-

3.21 MUX8_I 多路复用器，可级联（INTEGER型）

SIMOTION

SINAMICS

符号



简要说明

8路运算的INTEGER型功能块 该功能块可级联。

操作模式

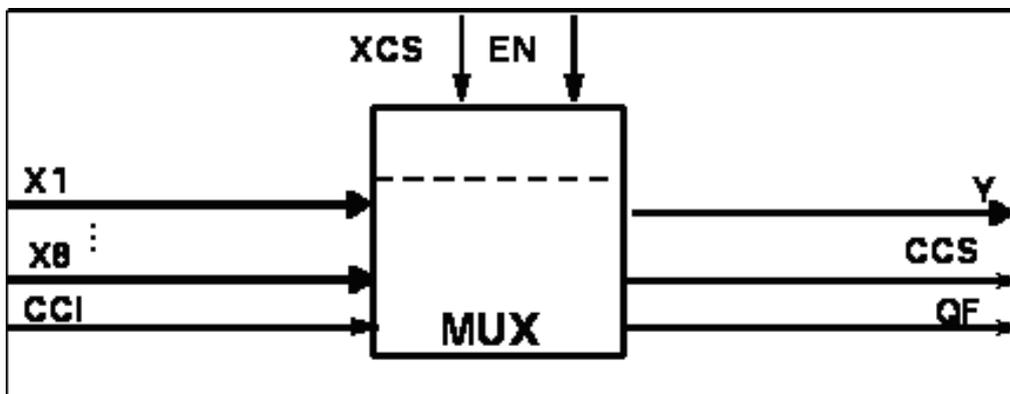
只要使能输入EN为逻辑0，该功能块就会在输出Y处输出级联输入CCI的值。

当EN的值为逻辑1时，只要16位的控制字XCS的取值介于1到8之间，输入变量X1到X8中的一个会连接到输出Y。

当输入XCS > 8时，输出Y取值为0，输出QF的值为逻辑1。级联控制字取值CCS = XCS-8，请参见真值表。

输出Y、CCS和QF可用于级联功能块。此时，第一个功能块的输出Y连接到下游多路复用器的输入CCI，连接输出CCS到下一个XCS，连接输出QF到下一个输入EN。

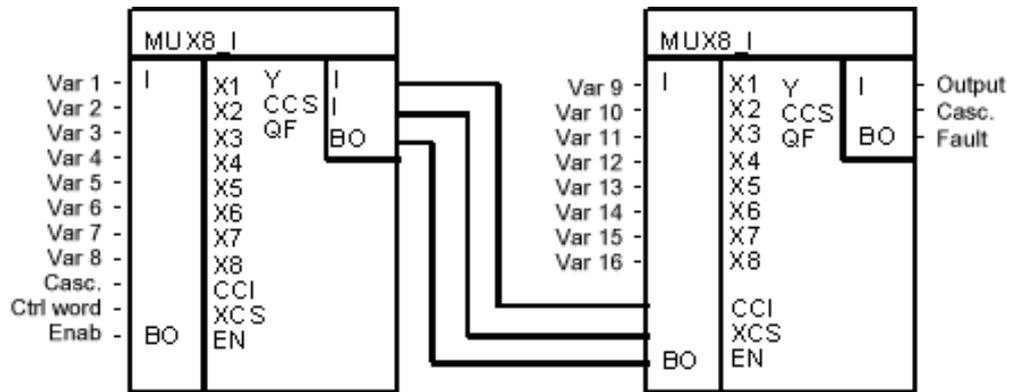
方框图



真值表

EN	XCS	Y	CSS	QF
0	不限	CCI	0	0
1	0	0	0	1
1	1	X1	0	0
1	2	X2	0	0
1	3	X3	0	0
1	4	X4	0	0
1	5	X5	0	0
1	6	X6	0	0
1	7	X7	0	0
1	8	X8	0	0
1	>8	0	XCS-8	1

级联



多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	INT	
X2	输入变量2	0	INT	
X3	输入变量3	0	INT	
X4	输入变量4	0	INT	
X5	输入变量5	0	INT	
X6	输入变量6	0	INT	
X7	输入变量7	0	INT	
X8	输入变量8	0	INT	
CCI	级联输入	0	INT	
XCS	控制字	0	0...32767	
EN	使能	0	0/1	
Y	输出变量	0	INT	

3.21 MUX8_I 多路复用器，可级联（INTEGER型）

块 连接	说明	预分配	取值范围	数
CCS	级联控制字	0	0...32767	
QF	错误消息	0	0/1	

据

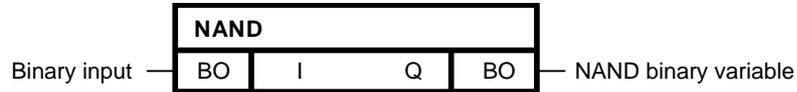
可在线加载	是
特别注意点	-

3.22 NAND 逻辑AND运算 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个BOOL型输入的NAND功能块

操作模式

该功能块将输入端I1到I4的二进制值进行逻辑AND运算，并反转运算结果，在二进制输出Q处输出。

$$Q = \overline{I_{01} \wedge \dots \wedge I_{nn}}$$

当I1到I4的所有一般输入值为1时，输出Q = 0；其他情况下，输出Q = 1。

真值表

Input				Output
I01	I02	I03	I04	Q
0	*	*	*	1
*	0	*	*	1
*	*	0	*	1
*	*	*	0	1
1	1	1	1	0

***8 user-defined**

多个块连接

块 连接	说明	预分配	取值范围	数
I	二进制输入	1	0/1	
Q	NAND二进制变量	0	0/1	

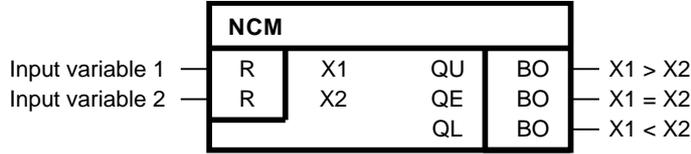
据

可在线加载	是
特别注意点	I包含最多4个输入 (I1到I4)

3.23 NCM 数字比较器 (REAL型)

SIMOTION SINAMICS

符号



简要说明

用于比较运算两个REAL型数字变量的功能块

操作模式

比较输入变量X1和X2，然后根据比较结果设置二进制输出QU、QE或QL。

真值表

输入变量比较	输出信号	输出信号Y	输出信号Y
	QU	QE	QL
X1 > X2	1	0	0
X1 = X2	0	1	0
X1 < X2	0	0	1

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	REAL	
X2	输入变量2	0	REAL	
QU	X1 > X2	0	0/1	
QE	X1 = X2	1	0/1	
QL	X1 < X2	0	0/1	

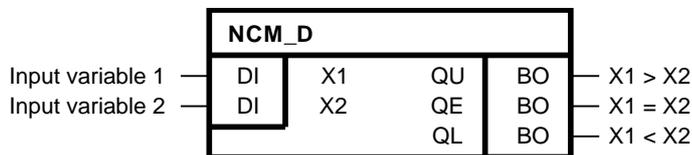
据

可在线加载	是
特别注意点	如果DCC图的执行组设置为“Do not calculate”，那么输出则设置为默认值。

3.24 NCM_D 数字比较器 (DOUBLE_INTEGER型)

SIMOTION SINAMICS

符号



简要说明

用于比较运算两个DOUBLE_INTEGER型数字变量的功能块

操作模式

比较输入变量X1和X2，然后根据比较结果设置二进制输出QU、QE或QL。

真值表

输入变量比较	输出信号	输出信号Y	输出信号Y
	QU	QE	QL
X1 > X2	1	0	0
X1 = X2	0	1	0
X1 < X2	0	0	1

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	DINT	
X2	输入变量2	0	DINT	
QU	X1 > X2	0	0/1	
QE	X1 = X2	1	0/1	
QL	X1 < X2	0	0/1	

据

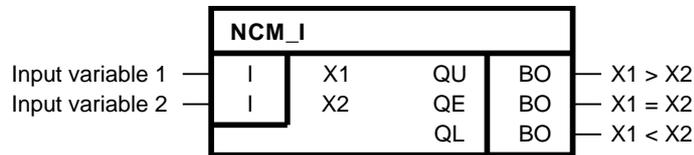
可在线加载	是
特别注意点	如果DCC图的执行组设置为“Do not calculate”，那么输出则设置为默认值。

3.25 NCM_I 数字比较器 (INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

用于比较运算两个INTEGER型数字变量的功能块

操作模式

比较输入变量X1和X2，然后根据比较结果设置二进制输出QU、QE或QL。

真值表

输入变量比较	输出信号	输出信号Y	输出信号Y
	QU	QE	QL
X1 > X2	1	0	0
X1 = X2	0	1	0
X1 < X2	0	0	1

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	INT	
X2	输入变量2	0	INT	
QU	X1 > X2	0	0/1	
QE	X1 = X2	1	0/1	
QL	X1 < X2	0	0/1	

据

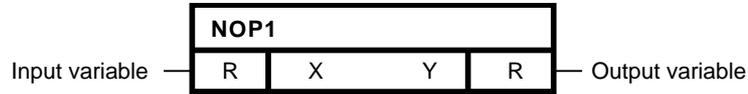
可在线加载	是
特别注意点	如果DCC图的执行组设置为“Do not calculate”，那么输出则设置为默认值。

3.26 NOP1 预留功能块 (REAL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- REAL型功能块用作预留功能块（无运算）。

操作模式

功能块在输入X处的输出值与输出Y处的一致。 这即是所谓的预留或无运算功能块。

初始化

功能块在输入X处的输出值与输出Y处的一致，从而为其他几个功能块的初始化提供了一般常量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	0.0	REAL	

据

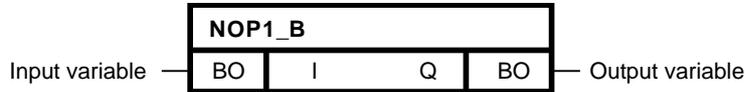
可在线加载	是
特别注意点	-

3.27 NOP1_B 预留功能块 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- BOOL型功能块用作预留功能块（无运算）。

操作模式

功能块在输入I处的输出值与输出Q处的一致。 这即是所谓的预留或无运算功能块。

初始化

功能块在输入I处的输出值与输出Q处的一致，从而为其他几个功能块的初始化提供了一般常量。

多个块连接

块 连接	说明	预分配	取值范围	数
I	输入变量	0	0/1	
Q	输出变量	0	0/1	

据

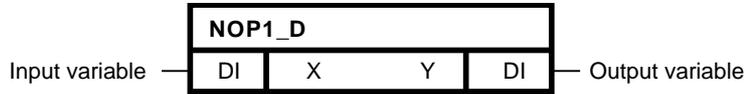
可在线加载	是
特别注意点	-

3.28 NOP1_D 预留功能块 (DOUBLE INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- DOUBLE INTEGER型功能块用作预留功能块（无运算）。

操作模式

功能块在输入X处的输出值与输出Y处的一致。 这即是所谓的预留或无运算功能块。

初始化

功能块在输入X处的输出值与输出Y处的一致，从而为其他几个功能块的初始化提供了一般常量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
Y	输出变量	0	DINT	

据

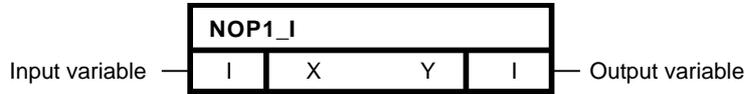
可在线加载	是
特别注意点	-

3.29 NOP1_I 预留功能块 (INT型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- INT型功能块用作预留功能块（无运算）。

操作模式

功能块在输入X处的输出值与输出Y处的一致。 这即是所谓的预留或无运算功能块。

初始化

功能块在输入X处的输出值与输出Y处的一致，从而为其他几个功能块的初始化提供了一般常量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
Y	输出变量	0	INT	

据

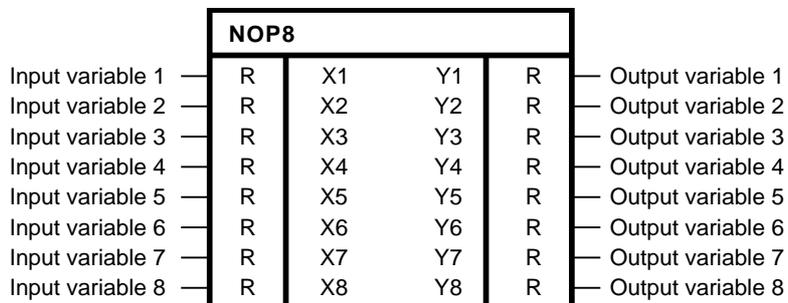
可在线加载	是
特别注意点	-

3.30 NOP8 预留功能块 (REAL型)

SIMOTION

SINAMICS

符号



简要说明

- REAL型功能块用作预留功能块（无运算）。

操作模式

功能块在输入X1到X8处的输出值与输出Y1到Y8处的一致。这即是所谓的预留或无运算功能块。

初始化

功能块在输入X1到X8处的输出值与输出Y1到Y8处的一致，从而为其他几个功能块的初始化提供了一般常量。

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0.0	REAL	
X2	输入变量2	0.0	REAL	
X3	输入变量3	0.0	REAL	
X4	输入变量4	0.0	REAL	
X5	输入变量5	0.0	REAL	
X6	输入变量6	0.0	REAL	
X7	输入变量7	0.0	REAL	
X8	输入变量8	0.0	REAL	
Y1	输出变量1	0.0	REAL	
Y2	输出变量2	0.0	REAL	
Y3	输出变量3	0.0	REAL	
Y4	输出变量4	0.0	REAL	

块 连接	说明	预分配	取值范围	数
Y5	输出变量5	0.0	REAL	
Y6	输出变量6	0.0	REAL	
Y7	输出变量7	0.0	REAL	
Y8	输出变量8	0.0	REAL	

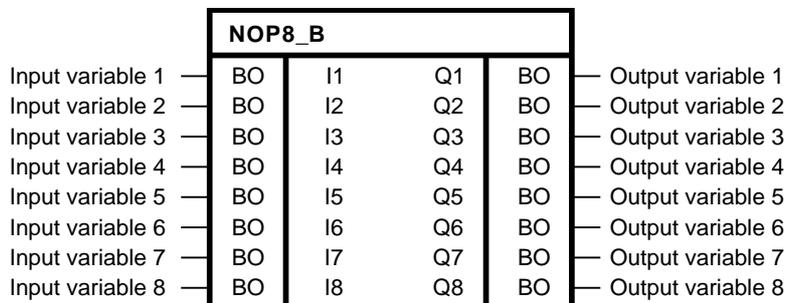
据

可在线加载	是
特别注意点	-

3.31 NOP8_B 预留功能块 (BOOL型)

SIMOTION SINAMICS

符号



简要说明

- BOOL型功能块用作预留功能块（无运算）。

操作模式

功能块在输入I1到I8处的输出值与输出Q1到Q8处的一致。这即是所谓的预留或无运算功能块。

初始化

功能块在输入I1到I8处的输出值与输出Q1到Q8处的一致，从而为其他几个功能块的初始化提供了一般常量。

多个块连接

块 连接	说明	预分配	取值范围	数
I1	输入变量1	0	0/1	
I2	输入变量2	0	0/1	
I3	输入变量3	0	0/1	
I4	输入变量4	0	0/1	
I5	输入变量5	0	0/1	
I6	输入变量6	0	0/1	
I7	输入变量7	0	0/1	
I8	输入变量8	0	0/1	
Q1	输出变量1	0	0/1	
Q2	输出变量2	0	0/1	
Q3	输出变量3	0	0/1	
Q4	输出变量4	0	0/1	

3.31 NOP8_B 预留功能块 (BOOL型)

块 连接	说明	预分配	取值范围	数
Q5	输出变量5	0	0/1	
Q6	输出变量6	0	0/1	
Q7	输出变量7	0	0/1	
Q8	输出变量8	0	0/1	

据

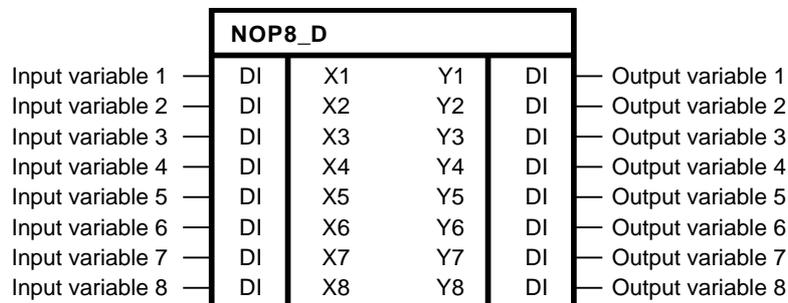
可在线加载	是
特别注意点	-

3.32 NOP8_D 预留功能块 (DOUBLE INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- DOUBLE INTEGER型功能块用作预留功能块（无运算）。

操作模式

功能块在输入X1到X8处的输出值与输出Y1到Y8处的一致。这即是所谓的预留或无运算功能块。

初始化

功能块在输入X1到X8处的输出值与输出Y1到Y8处的一致，从而为其他几个功能块的初始化提供了一般常量。

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	DINT	
X2	输入变量2	0	DINT	
X3	输入变量3	0	DINT	
X4	输入变量4	0	DINT	
X5	输入变量5	0	DINT	
X6	输入变量6	0	DINT	
X7	输入变量7	0	DINT	
X8	输入变量8	0	DINT	
Y1	输出变量1	0	DINT	
Y2	输出变量2	0	DINT	
Y3	输出变量3	0	DINT	
Y4	输出变量4	0	DINT	

3.32 NOP8_D 预留功能块 (DOUBLE INTEGER型)

块 连接	说明	预分配	取值范围	数
Y5	输出变量5	0	DINT	
Y6	输出变量6	0	DINT	
Y7	输出变量7	0	DINT	
Y8	输出变量8	0	DINT	

据

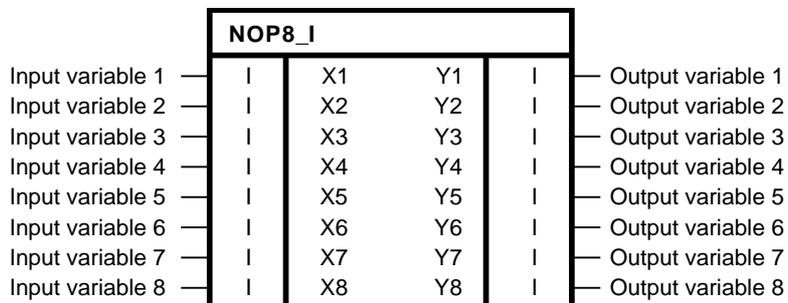
可在线加载	是
特别注意点	-

3.33 NOP8_I 预留功能块 (INTEGERT型)

SIMOTION

SINAMICS

符号



简要说明

- INTEGER型功能块用作预留功能块（无运算）。

操作模式

功能块在输入X1到X8处的输出值与输出Y1到Y8处的一致。这即是所谓的预留或无运算功能块。

初始化

功能块在输入X1到X8处的输出值与输出Y1到Y8处的一致，从而为其他几个功能块的初始化提供了一般常量。

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	INT	
X2	输入变量2	0	INT	
X3	输入变量3	0	INT	
X4	输入变量4	0	INT	
X5	输入变量5	0	INT	
X6	输入变量6	0	INT	
X7	输入变量7	0	INT	
X8	输入变量8	0	INT	
Y1	输出变量1	0	INT	
Y2	输出变量2	0	INT	
Y3	输出变量3	0	INT	
Y4	输出变量4	0	INT	

3.33 NOP8_I 预留功能块 (INTEGERT型)

块 连接	说明	预分配	取值范围	数
Y5	输出变量5	0	INT	
Y6	输出变量6	0	INT	
Y7	输出变量7	0	INT	
Y8	输出变量8	0	INT	

据

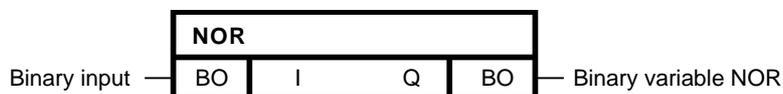
可在线加载	是
特别注意点	-

3.34 NOR 逻辑OR运算 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个BOOL型输入的NOR功能块

操作模式

该功能块将输入端I1到I4的二进制值进行逻辑OR运算，并在其二进制输出Q处输出结果。

$$Q = \overline{I_{01} \vee \dots \vee I_{m}}$$

当I1到I4的所有输入值为0时，输出Q = 1；其他情况下，输出Q = 0。

真值表

Input Output

I01	I02	I03	I04	Q
1	*	*	*	0
*	1	*	*	0
*	*	1	*	0
*	*	*	1	0
0	0	0	0	1

*8 user-defined

多个块连接

块 连接	说明	预分配	取值范围	数
I	二进制输入	0	0/1	
Q	二进制变量NOR	1	0/1	

据

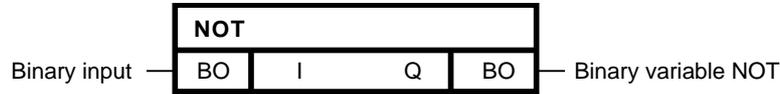
可在线加载	是
特别注意点	I包含最多4个连接 (I1到I4)

3.35 NOT 反相器 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- BOOL型反相器

操作模式

该功能块对输入I处的二进制变量反向运算，并将结果作为Q的输出值。

$$Q = \bar{I}$$

真值表

输入I	输出Q
1	0

多个块连接

块 连接	说明	预分配	取值范围	数
I	二进制输入	0	0/1	
Q	二进制变量NOT	1	0/1	

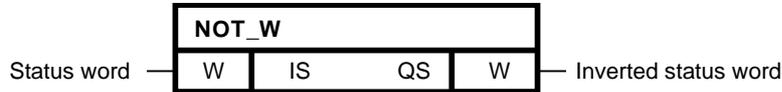
据

可在线加载	是
特别注意点	-

3.36 NOT_W 状态字反相器 (WORD型)

 SIMOTION SINAMICS

符号



简要说明

- WORD型状态字反相器
- IS的反码形式

操作模式

16个二进制状态合成于一个状态字中。

该功能块逐位反向运算状态字IS，并在输出QS处输出结果。

以下适用于反向状态字的第k位：

$$QS_k = \overline{IS_k}$$

反码形式

例如：IS = 15 → QS = -16

Status word



Inverted status word



多个块连接

块 连接	说明	预分配	取值范围	数
IS	状态字	16#0000	WORD	

3.36 NOT_W 状态字反相器 (WORD型)

块 连接	说明	预分配	取值范围	数
QS	反向状态字	16#FFFF	WORD	

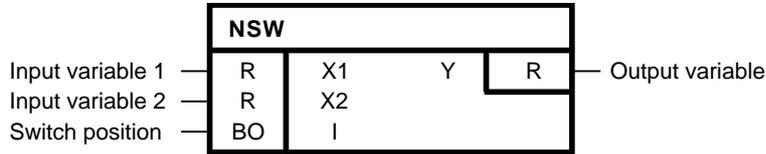
据

可在线加载	是
特别注意点	-

3.37 NSW 数字切换开关 (REAL型)

SIMOTION SINAMICS

符号



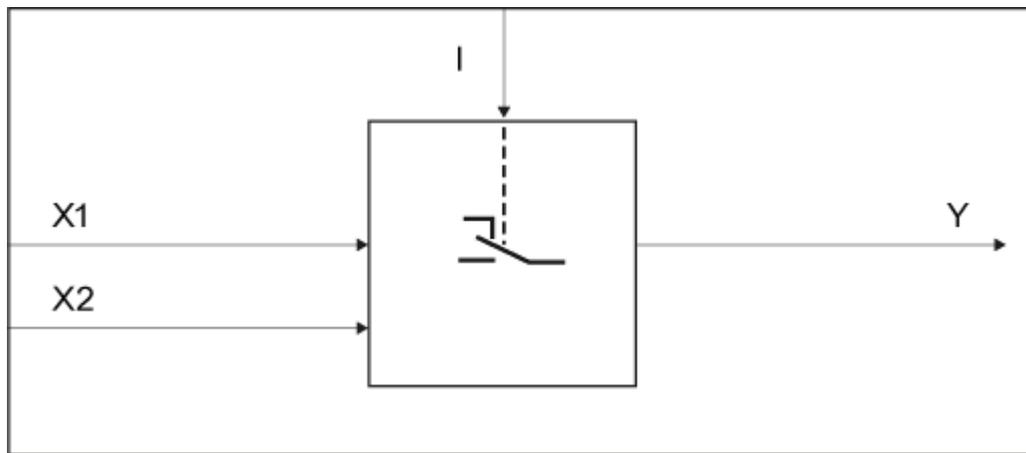
简要说明

- 该功能块将两个REAL型数字输入变量中的一个转化为输出。

操作模式

当输入I = 0时，输出Y为X1。
 当输入I = 1时，输出Y为X2。

方框图



真值表

转换位置1	输出变量Y
0	Y = X1
1	Y = X2

3.37 NSW 数字切换开关 (REAL型)

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	REAL	
X2	输入变量2	0	REAL	
I	转换位置	0	0/1	
Y	输出变量	0	REAL	

据

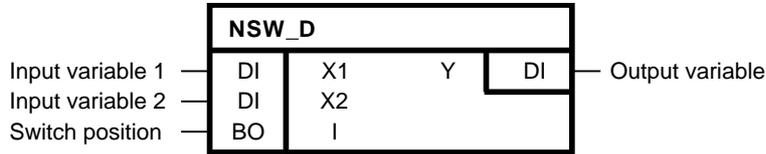
可在线加载	是
特别注意点	-

3.38 NSW_D 数字切换开关 (DOUBLE INTEGER型)

SIMOTION

SINAMICS

符号



简要说明

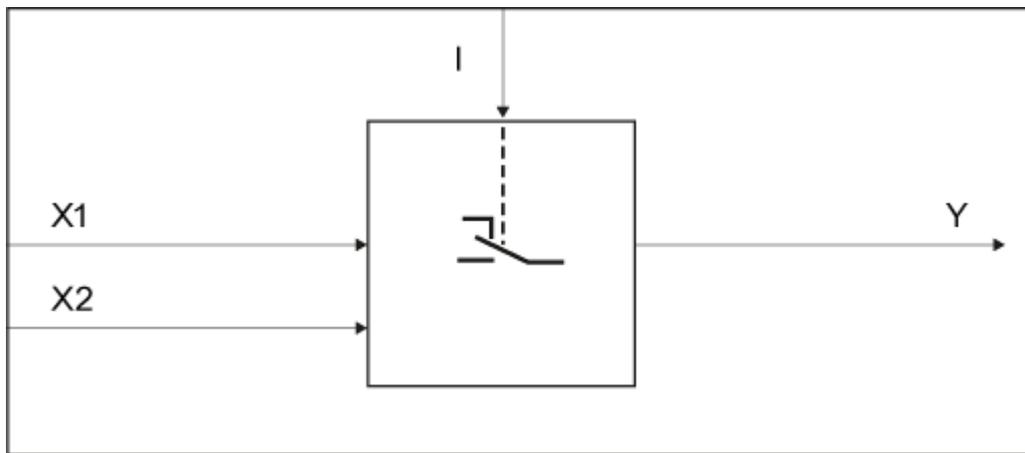
- 该功能块将两个DOUBLE INTEGER型数字输入变量中的一个转化为输出。

操作模式

当输入I = 0时，输出Y为X1。

当输入I = 1时，输出Y为X2。

方框图



真值表

转换位置1	输出变量Y
0	Y = X1
1	Y = X2

3.38 NSW_D 数字切换开关 (DOUBLE INTEGER型)

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	DINT	
X2	输入变量2	0	DINT	
I	转换位置	0	0/1	
Y	输出变量	0	DINT	

据

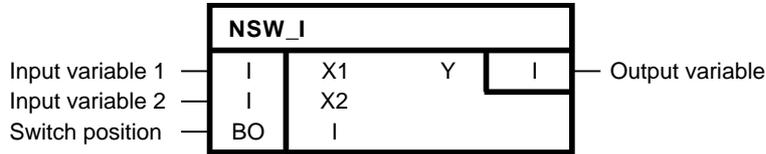
可在线加载	是
特别注意点	-

3.39 NSW_I 数字切换开关 (INTEGER型)

SIMOTION

SINAMICS

符号



简要说明

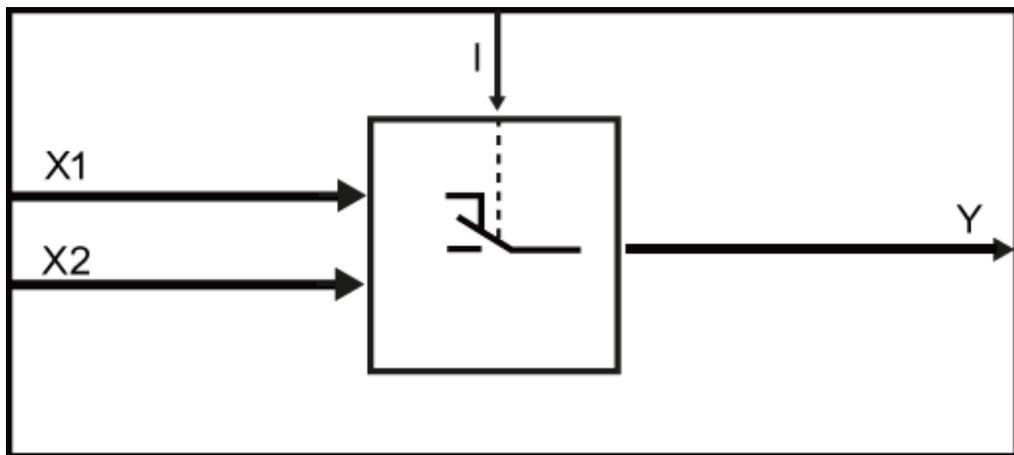
- 该功能块将两个INTEGER型数字输入变量中的一个转化为输出。

操作模式

当输入I = 0时，输出Y为X1。

当输入I = 1时，输出Y为X2。

方框图



真值表

转换位置1	输出变量Y
0	Y = X1
1	Y = X2

3.39 NSW_I 数字切换开关（INTEGER型）

多个块连接

块 连接	说明	预分配	取值范围	数
X1	输入变量1	0	INT	
X2	输入变量2	0	INT	
I	转换位置	0	0/1	
Y	输出变量	0	INT	

据

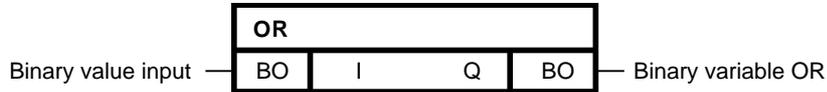
可在线加载	是
特别注意点	-

3.40 OR 逻辑OR运算 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个BOOL型输入的OR功能块

操作模式

该功能块将输入端I1到I4的二进制值进行逻辑OR（析取）运算，并在其二进制输出Q处输出结果。

$$Q = I_{01} \wedge \dots \wedge I_{04}$$

当I1到I4的所有输入值为0时，输出Q = 0；其他情况下，输出Q = 1。

真值表

Input Output

I01	I02	I03	I04	Q
1	*	*	*	1
*	1	*	*	1
*	*	1	*	1
*	*	*	1	1
0	0	0	0	0

*8 user-defined

3.40 OR 逻辑OR运算 (BOOL型)

多个块连接

块 连接	说明	预分配	取值范围	数
I	二进制值输入	0	0/1	
Q	二进制变量OR	0	0/1	

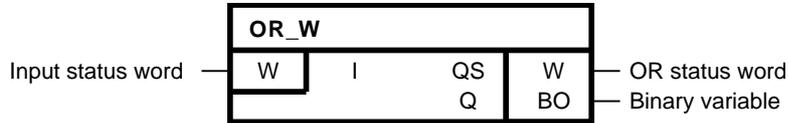
据

可在线加载	是
特别注意点	I包含最多4个连接 (I1到I4)

3.41 OR_W 逻辑OR运算 (WORD型)

 SIMOTION SINAMICS

符号



简要说明

- 支持最多4个WORD型输入的OR功能块

操作模式

16个二进制状态合成于一个状态字中。

该功能块根据逻辑OR函数逐位合成I1到I4的状态字。

块输出QS (状态字OR) 为该计算结果。

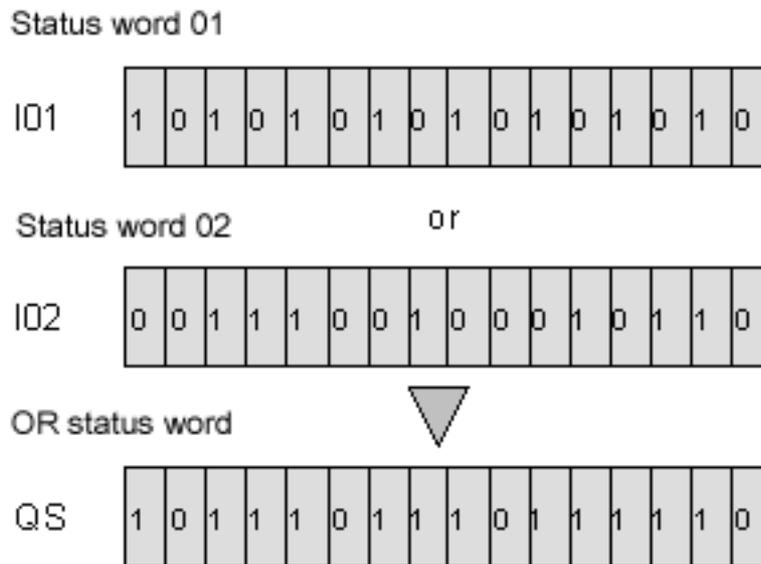
以下适用于状态字OR的第k位:

$$QS_k = I02_k \vee I02_k, \quad k = 1 \dots 16$$

当功能块输入端I1到I4中有一个及以上的等效位等于1时, OR状态字的一位等于1。

当OR状态字有一个及以上的位等于1时, 二进制输出Q为1。

以下为三个输入的状态图



多个块连接

块 连接	说明	预分配	取值范围	数
I	输入状态字	16#0000	WORD	
QS	OR状态字	16#0000	WORD	
Q	二进制变量	0	0/1	

据

可在线加载	是
特别注意点	I包含最多4个连接 (I1到I4)

3.42 PCL 脉冲缩短设备 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 用于限制脉冲时长的定时器

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

输入I处脉冲的上升沿设置输出Q为1。当输入I为0 (输入 $I = 0$) 或脉冲时长T已经过期时, 输出Q为0。当T=0时, 1个周期的脉冲时长生效。

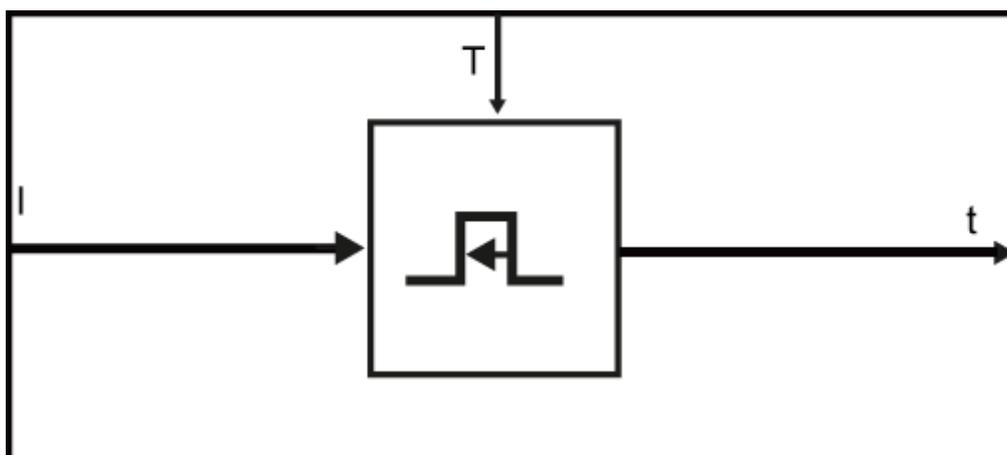
初始化

初始化定义了第一个周期的起始值。

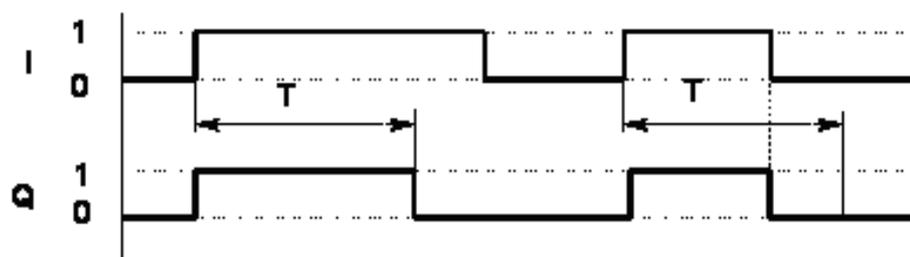
在上游功能块输出初始化过程中, 当输入I的取值为1时, 该功能块不可以在第一个周期检测正沿。

当输出Q取默认值1时, 那么在脉冲时长T内的初始化完成之后, 输出 $Q = 1$ 。

方框图



时序图



Output pulse Q as a function of pulse duration T and input pulse I

多个块连接

块 连接	说明	预分配	取值范围	数
I	输入脉冲	0	0/1	
T	脉冲时长 (毫秒)	0	SDTIME	
Q	输出脉冲	0	0/1	

据

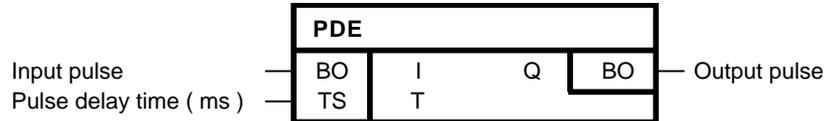
可在线加载	是
特别注意点	-

3.43 PDE 接通延迟设备 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持接通延迟功能的BOOL型定时器

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

输入T处的脉冲延迟时间以输入I处的上升沿接替。在这段延迟时间之后, 输出Q设置为1。

当I为0 ($I = 0$) 时, 输出Q为0。

当输入脉冲I时长短于脉冲延迟时长T时, Q依然为0。

当时间T太长, 超过最大可显示的内部值 (T/t_a 为32位值, 其中 t_a 为采样时间) 时, 将会进行最大值限制 (也就是当 $t_a=1$ 毫秒, 约等于 50天)。

当 $T=0$ 时, 1个周期的脉冲延迟时长生效。

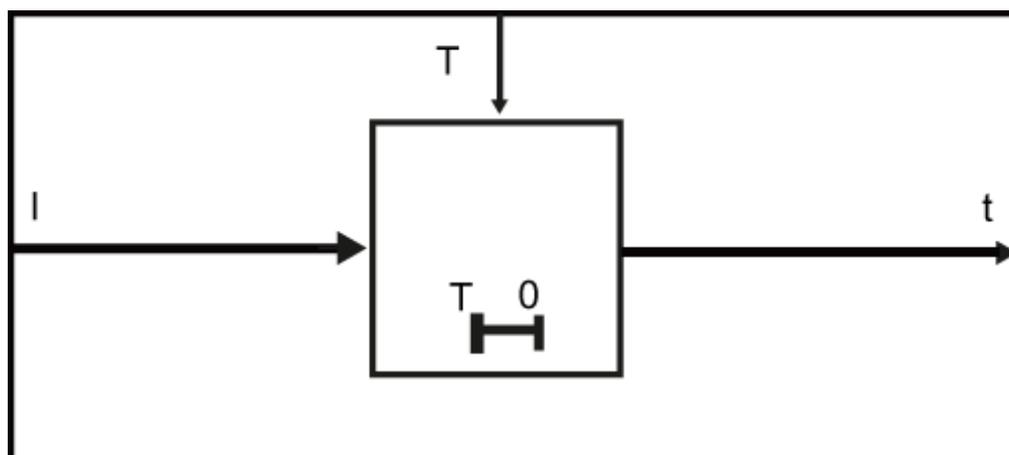
初始化

初始化定义了第一个周期的起始值。

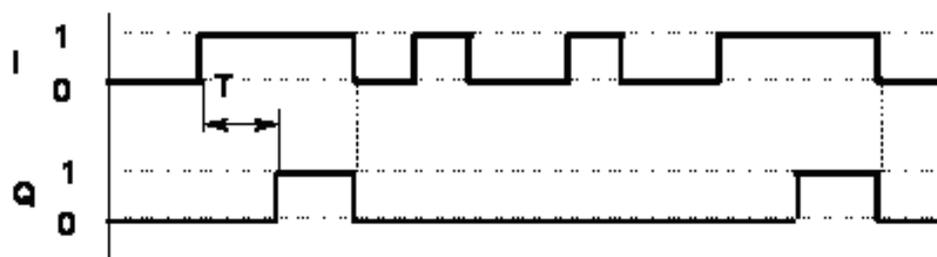
在上游功能块输出初始化过程中, 当输入I的取值为1时, 该功能块不可以在第一个周期检测正沿。因此, 在第一个周期内脉冲延迟时长T不会被 $I = 1$ 所接替, 初始化时指定的时间依然生效。

初始化过程中, 当输出Q取值1时, 如果 $I=1$, 初始化后立即设置输出 $Q=1$ 。

方框图



时序图



Output pulse Q as a function of pulse duration T and input pulse I

多个块连接

块 连接	说明	预分配	取值范围	数
I	输入脉冲	0	0/1	
T	脉冲延迟时长 (毫秒)	0	SDTIME	
Q	输出脉冲	0	0/1	

据

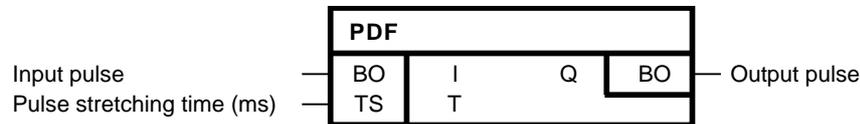
可在线加载	是
特别注意点	-

3.44 PDF 断开延迟设备 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持断开延迟功能的BOOL型定时器

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

在脉冲展宽时长T之后, 块输入I处的脉冲下降沿复位输出Q为0。

当I为1 (I = 0) 时, 输出Q为1。

当输入脉冲I = 0且接通延迟时间T已经过期时, 输出Q为0。

在时间T过期之前, 当输入I复位为1时, 输出Q依然为1。

当T=0时, 1个周期的脉冲展宽时长生效。

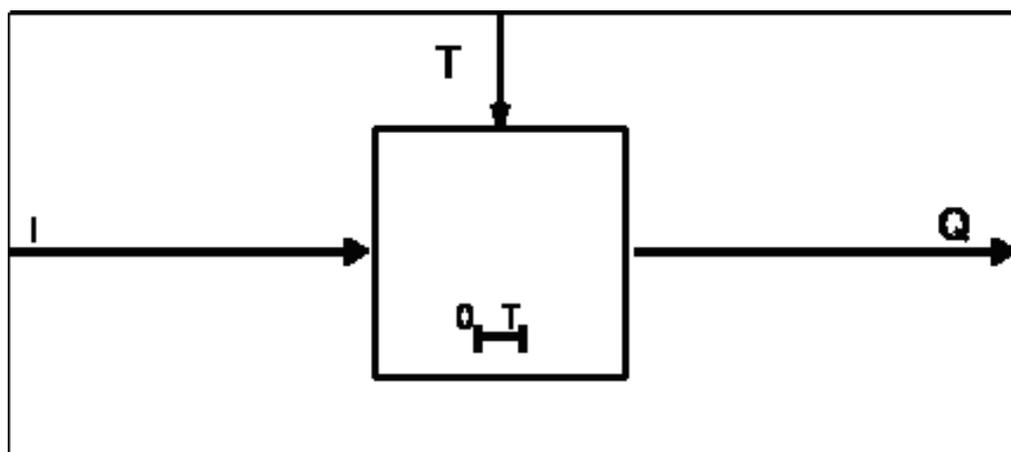
初始化

初始化定义了第一个周期的起始值。

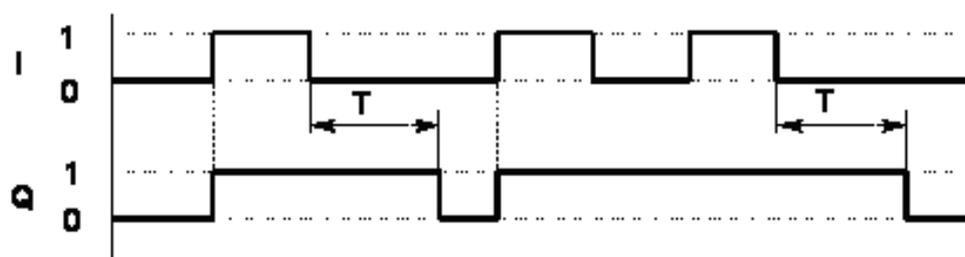
在上游功能块输出初始化过程中, 当输入I的取值为1时, 该功能块不可以在第一个周期检测负沿。

当输出Q在初始化过程中取值为1时, 那么在脉冲展宽时长T内的初始化完成之后, 输出Q = 1。

方框图



时序图



Output pulse Q as a function of pulse duration T and input pulse I

多个块连接

块 连接	说明	预分配	取值范围	数
I	输入脉冲	0	0/1	
T	脉冲展宽时长 (毫秒)	0	SDTIME	
Q	输出脉冲	0	0/1	

据

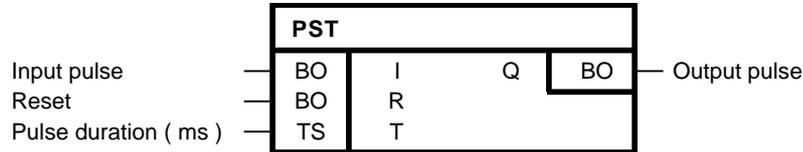
可在线加载	是
特别注意点	-

3.45 PST 脉冲展宽功能块 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 用于产生时长最小且带额外复位输入的脉冲的功能块

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

输入I处的脉冲上升沿设置输出Q为1。

直到输入脉冲I = 0且脉冲时长T过期时, 输出Q才会返回为0。

通过复位输入R (R = 1) 可以在任意时刻设置输出Q为0。

当T=0时, 1个周期的脉冲时长生效。

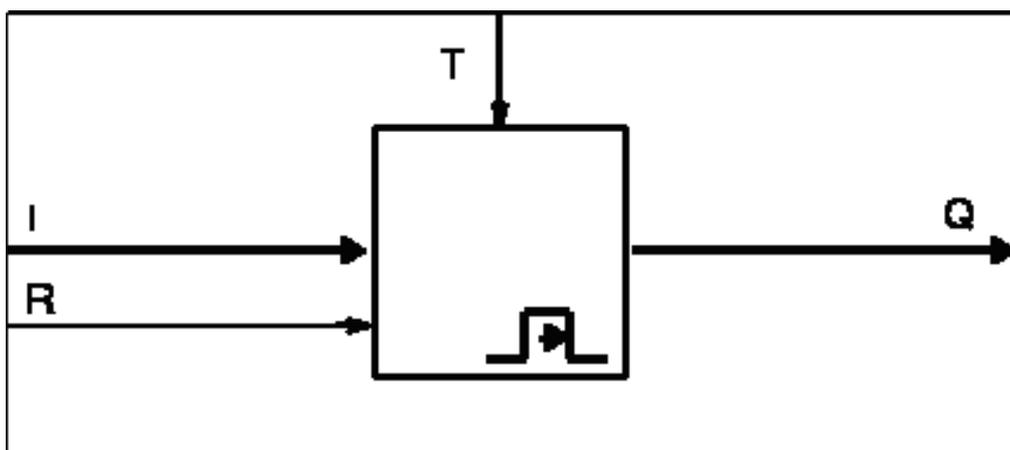
初始化

初始化定义了第一个周期的起始值。

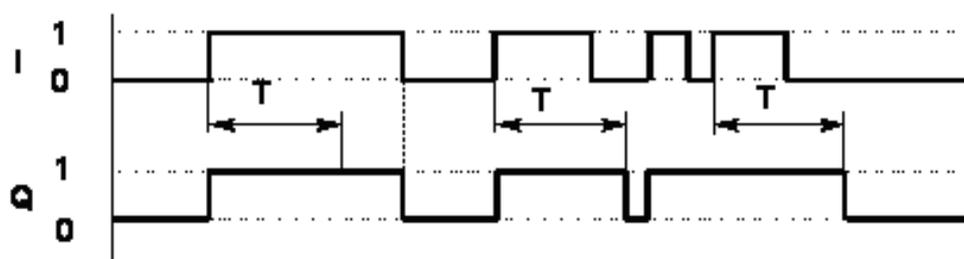
在上游功能块输出初始化过程中, 当输入I的取值为1时, 该功能块不可以在第一个周期检测正沿。

当输出Q在初始化过程中取值为1时, 那么在脉冲时长T内的初始化完成之后, 输出Q = 1。

方框图



时序图



Output pulse Q as a function of pulse duration T and input pulse I (if R=0)

多个块连接

块 连接	说明	预分配	取值范围	数
I	输入脉冲	0	0/1	
R	复位	0	0/1	
T	脉冲时长 (毫秒)	0	SDTIME	
Q	输出脉冲	0	0/1	

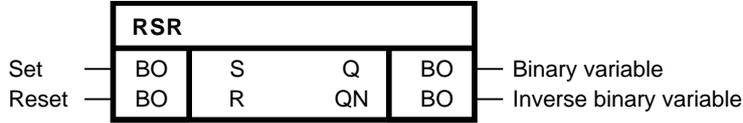
据

可在线加载	是
特别注意点	-

3.46 RSR 复位优先的RS触发器 (BOOL型)

SIMOTION SINAMICS

符号



简要说明

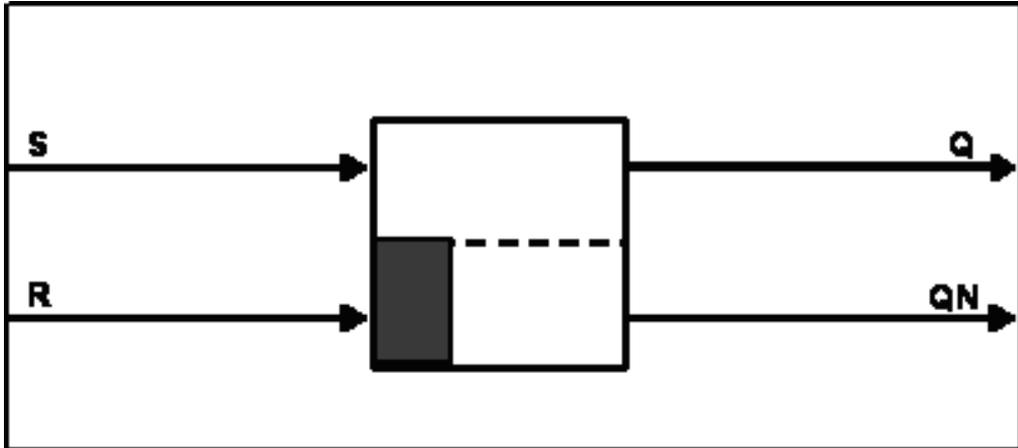
- 用作静态二进制值存储器

操作模式

当输入S的值为逻辑1时，输出Q的值为逻辑1。当输入R的值为逻辑1时，输出Q的值为逻辑0。当两个输入的值都为逻辑0时，输出Q则保持不变。然而，当两个输入的值都为逻辑1时，由于复位输入处于优先地位，输出Q的值为逻辑0。

输出QN的值总是与Q相反。

方框图



真值表

设置/复位指令发出时的二进制值

3.46 RSR 复位优先的RS触发器（BOOL型）

二进制指令		输出状态Q
S	R	
0	0	Q不变
0	1	Q = 0
1	0	Q = 1
1	1	Q = 0

多个块连接

块 连接	说明	预分配	取值范围	数
S	设定	0	0/1	
R	复位	0	0/1	
Q	二进制变量	0	0/1	
QN	反二进制变量	1	0/1	

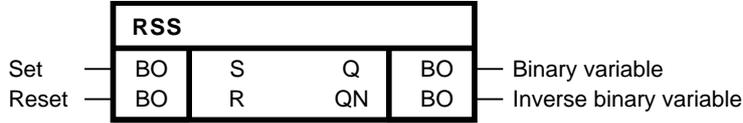
据

可在线加载	是
特别注意点	-

3.47 RSS 设定优先的RS触发器 (BOOL型)

SIMOTION SINAMICS

符号



简要说明

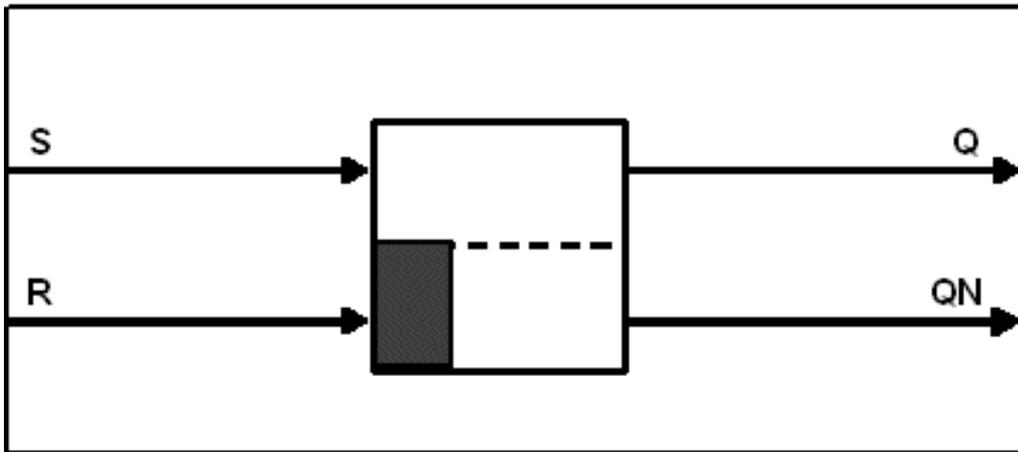
- BOOL型功能块，用作静态二进制存储器

操作模式

当输入S的值为逻辑1时，输出Q的值为逻辑1。当输入R的值为逻辑1时，输出Q的值为逻辑0。当两个输入的值都为逻辑0时，输出Q则保持不变。然而，当两个输入的值都为逻辑1时，由于设定输入处于优先地位，输出Q的值也为逻辑1。

输出QN的值总是与Q相反。

方框图



真值表

设置/复位指令发出时的二进制值

3.47 RSS 设定优先的RS触发器（BOOL型）

二进制指令		输出状态Q
S	R	
0	0	Q不变
0	1	Q = 0
1	0	Q = 1
1	1	Q = 1

多个块连接

块 连接	说明	预分配	取值范围	数
S	设定	0	0/1	
R	复位	0	0/1	
Q	二进制变量	0	0/1	
QN	反二进制变量	1	0/1	

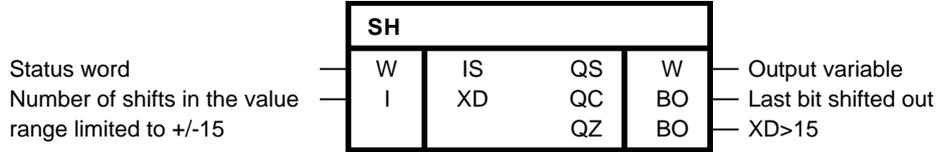
据

可在线加载	是
特别注意点	-

3.48 SH 移位功能块 (WORD型)

SIMOTION SINAMICS

符号



简要说明

- WORD型功能块逐位将状态字向左或向右移动。

操作模式

功能块根据输入XD处指定的位置数，逐位移动输入IS的状态字。

在移动过程中，输出变量QS的新位置都填充为0，与移动方向无关。

最后移动的位在输出QC处移出。当XD = 0时，QC = 0始终为真。当|XD| > 15时，QC = 0、QS = 0和QZ = 1始终为真。

向左移动的示例：

XD = 2; IS = 15

-> QS = 60; QC = 0

Binary number of IS



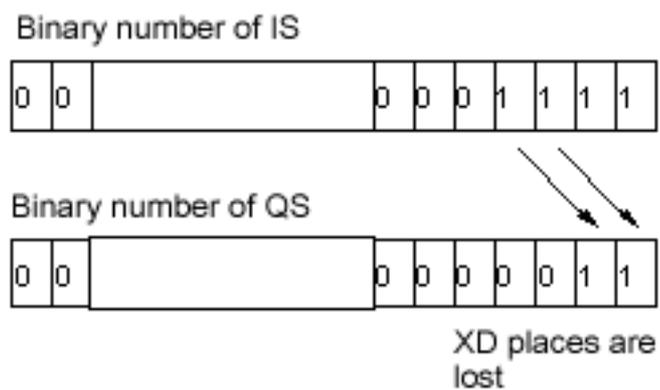
Binary number of QS



向右移动的示例：

XD = -2; IS = 15

-> QS = 3 (省略余数) ; QC = 1



多个块连接

块 连接	说明	预分配	取值范围	数
IS	状态字	16#0000	WORD	
XD	取值范围+/-15内的移位数	0	INT	
QS	输出变量	16#0000	WORD	
QC	最后移动的位	0	0/1	
QZ	XD>15	0	0/1	

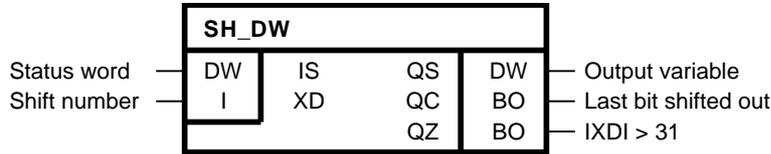
据

可在线加载	是
特别注意点	-

3.49 SH_DW 移位功能块 (DWORD型)

- SIMOTION
- SINAMICS

符号



简要说明

- DWORD型功能块逐位将状态字向左或向右移动。

操作模式

功能块根据输入XD处指定的位置数，逐位移动输入IS的状态字。

在移动过程中，输出变量QS的新位置都填充为0，与移动方向无关。

最后移动的位在输出QC处移出。当XD = 0时，QC = 0始终为真。当|XD| > 31时，QC = 0、QS = 0和QZ = 1始终为真。

向左移动的示例：

XD = 2; IS = 15
 -> QS = 60; QC = 0



向右移动的示例：

XD = -2; IS = 15
 -> QS = 3 (省略余数) ; QC = 1



多个块连接

块 连接	说明	预分配	取值范围	数
IS	状态字	16#00000000	DWORD	
XD	移位数	0	+/-31	
QS	输出变量	16#00000000	DWORD	
QC	最后移动的位	0	0/1	
QZ	IXDI > 31	0	0/1	

据

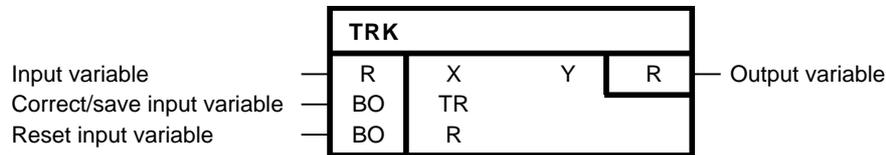
可在线加载	是
特别注意点	-

3.50 TRK 校正/存储元件 (REAL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

用于存储当前输入值的REAL型功能块，有如下属性：

- 针对输入值的沿控制锁功能
- 针对输出值的级控制校正

操作模式

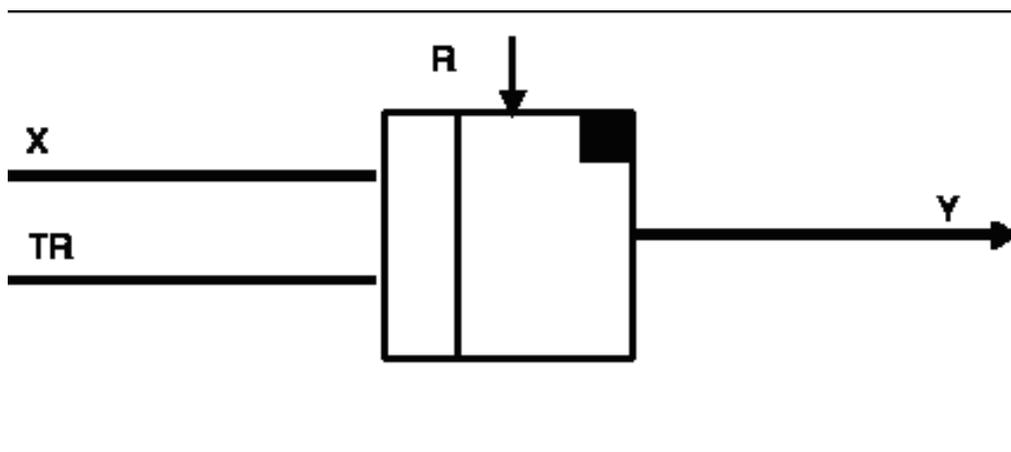
TRACK	TR = 1	直接校正输出值Y = X。
	TR = 1→ 0	当TR为负沿时，保存当前输入变量，并在Y输出端输出。
	TR = 0	输出Y的值不会改变
RESET	R = 1	输出Y复位为0。 复位输入优先。

初始化

在上游功能块输出初始化过程中，当输入TR的取值为1时，可以在第一个周期检测负沿。在START模式下，TR的值暂时储存。

在上游功能块输出初始化过程中，当输入TR的取值为0时，该功能块不可以在第一个周期检测负沿。

方框图



真值表

输入		触发时的输出Y
TR	R	
0	0	$Y_n = Y_{n-1}$
1	0	$Y_n = X_n$
1	1	$Y_n = 0$
1→0	0	$Y_n = X_n$
1→0	1	$Y_n = 0$

1 → 0: 下降

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
TR	校正/保存输入变量	0	0/1	
R	复位输入变量	0	0/1	
Y	输出变量	0.0	REAL	

据

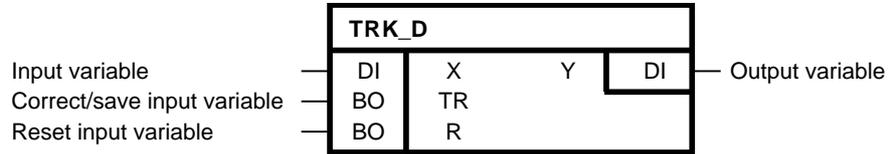
可在线加载	是
特别注意点	-

3.51 TRK_D 校正/存储元件 (DOUBLE INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

用于存储当前输入值的DOUBLE INTEGER型功能块，有如下属性：

- 针对输入值的沿控制锁功能
- 针对输出值的级控制校正

操作模式

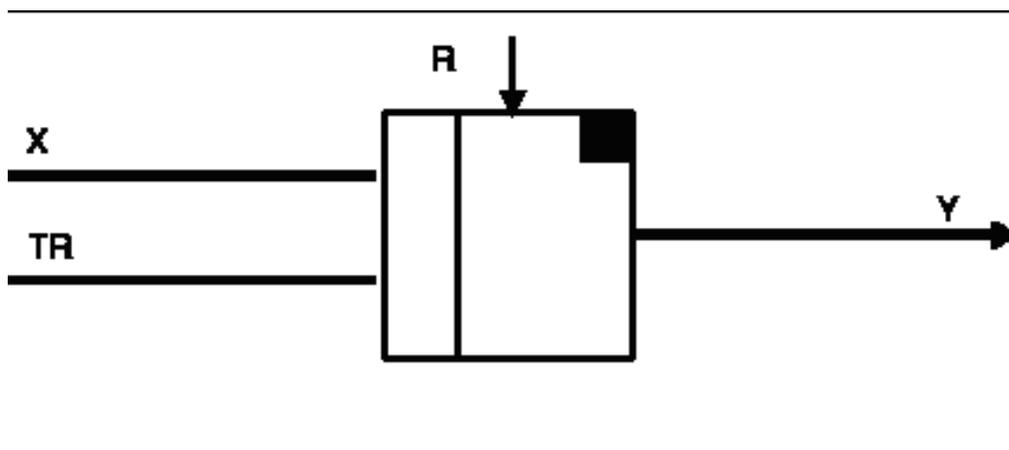
TRACK	TR = 1	直接校正输出值Y = X。
	TR = 1→ 0	当TR为负沿时，保存当前输入变量，并在Y输出端输出。
	TR = 0	输出Y的值不会改变
RESET	R = 1	输出Y复位为0。 复位输入优先。

初始化

在上游功能块输出初始化过程中，当输入TR的取值为1时，可以在第一个周期检测负沿。在START模式下，TR的值暂时储存。

在上游功能块输出初始化过程中，当输入TR的取值为0时，该功能块不可以在第一个周期检测负沿。

方框图



真值表

输入		触发时的输出Y
TR	R	
0	0	$Y_n = Y_{n-1}$
1	0	$Y_n = X_n$
1	1	$Y_n = 0$
1→0	0	$Y_n = X_n$
1→0	1	$Y_n = 0$

1 → 0: 下降

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
TR	校正/保存输入变量	0	0/1	
R	复位输入变量	0	0/1	
Y	输出变量	0	DINT	

据

可在线加载	是
特别注意点	-

3.52 XOR 逻辑XOR运算 (BOOL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持最多4个BOOL型输入的XOR功能块

操作模式

该功能块将输入端I1到I4的二进制值进行逻辑XOR运算，并在其二进制输出Q处输出结果。
 当I1到I4的所有输入为0或I1到I4中的偶数输入为1时，输出Q为0。
 当I1到I4中的奇数输入为1时，输出Q为1。

多个块连接

块 连接	说明	预分配	取值范围	数
I	二进制值输入	0	0/1	
Q	二进制变量XOR	0	0/1	

据

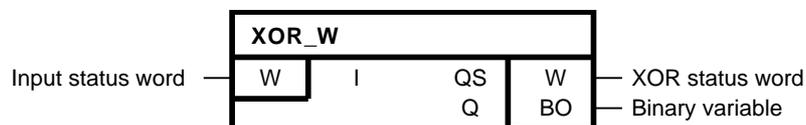
可在线加载	是
特别注意点	I包含最多4个连接 (I1到I4)

3.53 XOR_W 逻辑XOR运算 (WORD型)

☑ SIMOTION

☐ SINAMICS

符号



简要说明

- 支持最多4个WORD型输入的XOR功能块

操作模式

该功能块根据逻辑XOR函数逐位合成I1到I4的状态字。

块输出QS (状态字XOR) 为该计算结果。

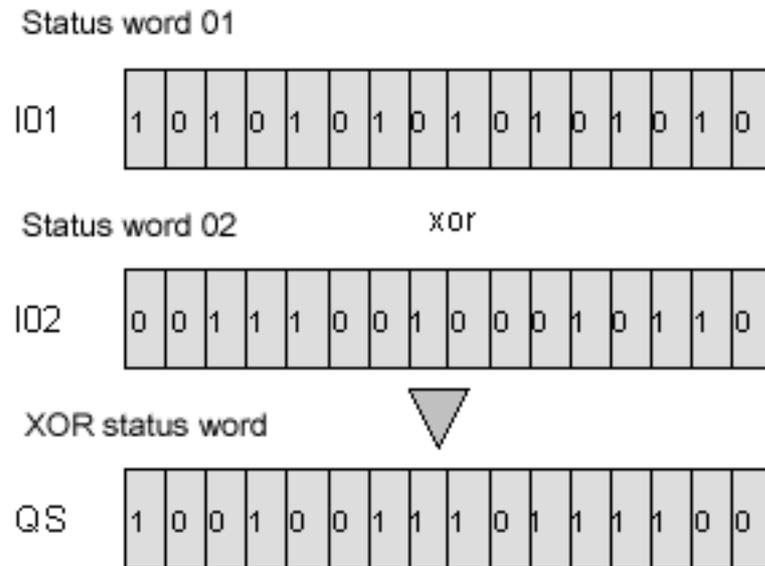
以下适用于状态字XOR的第k位:

$$QS_k = (\overline{I01_k} \wedge I02_k) \vee (I01_k \wedge \overline{I02_k}), \quad k = 1 \dots 16$$

当功能块输入端I1到I4中奇数的等效位等于1时, XOR状态字的一位等于1。

当XOR状态字有一个及以上的位等于1时, 二进制输出Q为1。

以下为三个输入的状态图



多个块连接

块 连接	说明	预分配	取值范围	数
I	输入状态字	16#0000	WORD	
QS	XOR状态字	16#0000	WORD	
Q	二进制变量	0	0/1	

据

可在线加载	是
特别注意点	包含最多4个连接 (I1到I4)

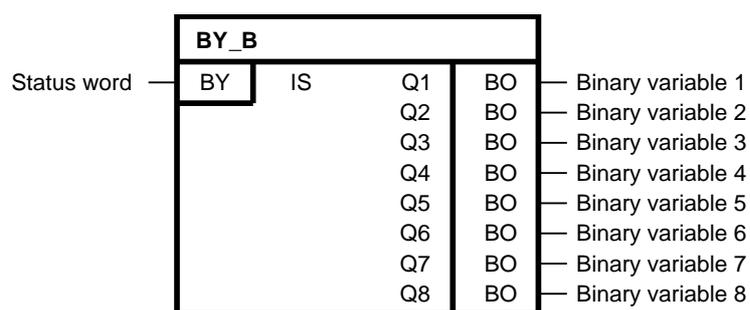
Conversion

4.1 BY_B 转换器状态字节转换为8个二进制变量

 SIMOTION

 SINAMICS

符号



简要说明

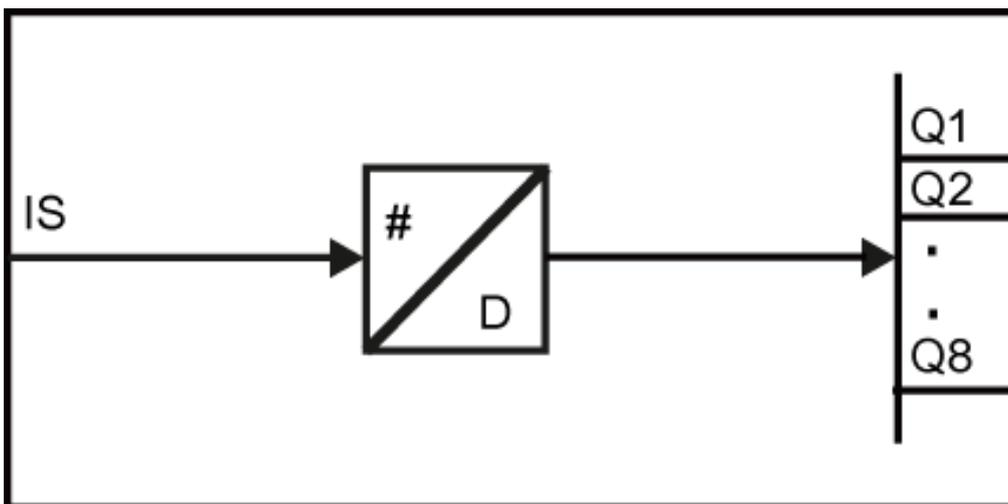
- 状态字解码为8个二进制变量

操作模式

该功能块将状态字IS解码为8个二进制变量，并将结果作为Q1到Q8的输出值。
输出Q1到Q8的二进制变量为状态字节的每个对偶等价量 2^0 到 2^7 。

4.1 BY_B 转换器状态字节转换为8个二进制变量

功能块图



映射方案

状态字节IS的数位位置（对偶等价量）	输出变量
0 (2^0)	Q1
1 (2^1)	Q2
2 (2^2)	Q3
3 (2^3)	Q4
4 (2^4)	Q5
5 (2^5)	Q6
6 (2^6)	Q7
7 (2^7)	Q8

多个块连接

块 连接	说明	预分配	取值范围	数
IS	状态字	16#00	BYTE	
Q1	二进制变量1	0	0/1	
Q2	二进制变量2	0	0/1	
Q3	二进制变量3	0	0/1	
Q4	二进制变量4	0	0/1	
Q5	二进制变量5	0	0/1	
Q6	二进制变量6	0	0/1	
Q7	二进制变量7	0	0/1	
Q8	二进制变量8	0	0/1	

据

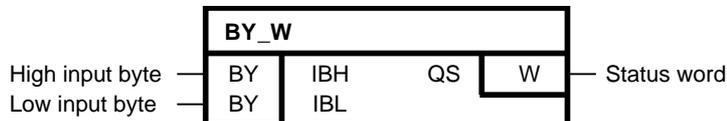
可在线插入	是
特别注意点	-

4.2 BY_W 状态字节到状态字的转换器

SIMOTION

SINAMICS

符号



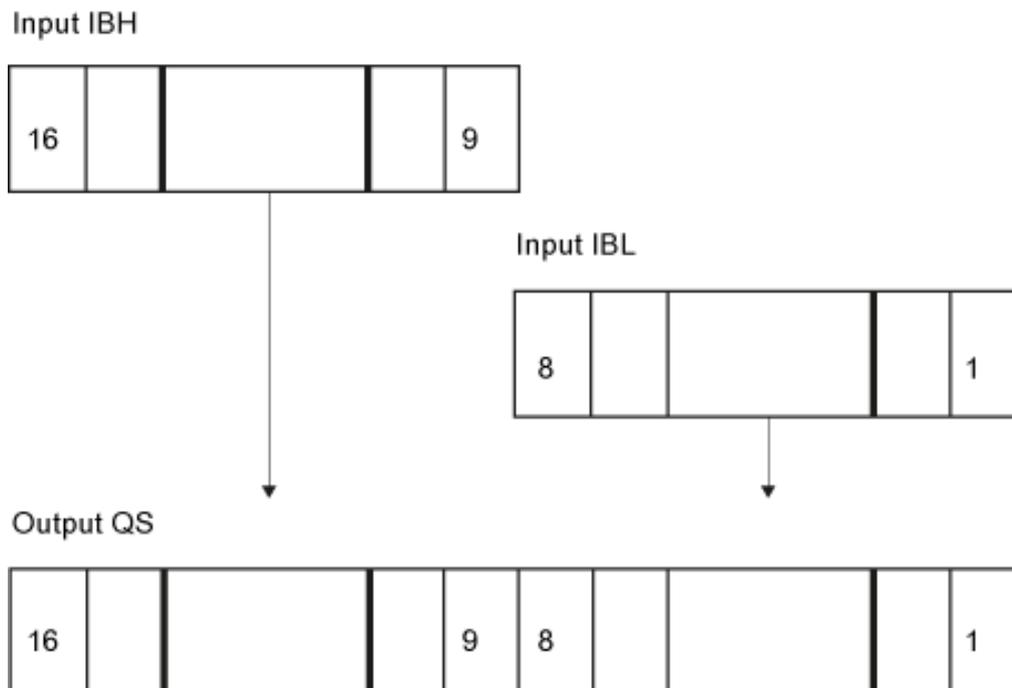
简要说明

- 组合2个字节为1个字

操作模式

该功能块组合2个字节为1个字。输出字的低字节为输入字节IBL，输出字的高字节为输入字节IBH。QS处的输出字根据以下方案转换。

转换方案



多个块连接

块 连接	说明	预分配	取值范围	数
IBH	高输入字节	16#00	BYTE	
IBL	低输入字节	16#00	BYTE	
QS	状态字	16#0000	WORD	

据

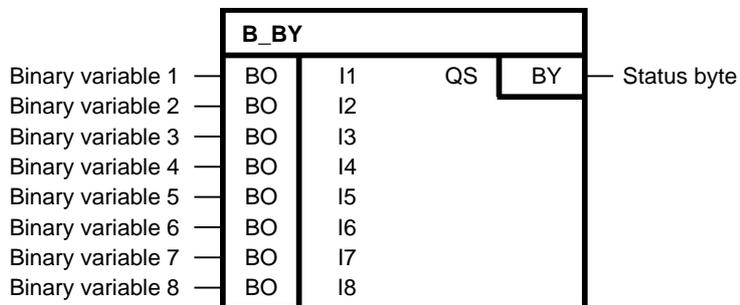
可在线插入	是
特别注意点	-

4.3 B_BY 8个二进制变量到状态字节的转换器

4.3 B_BY 8个二进制变量到状态字节的转换器

SIMOTION SINAMICS

符号



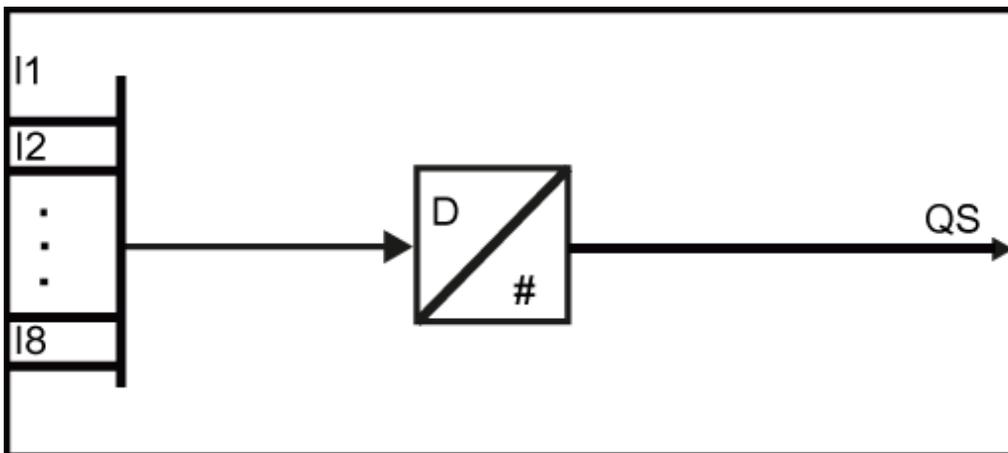
简要说明

- 由8个二进制变量生成状态字节

操作模式

该功能块组合I1到I8的二进制变量为1个状态字节，并将结果作为QS的输出值。
 每个I1到I8的二进制输入变量为生成状态字的对偶等价量 2^0 到 2^7 。

功能块图



映射方案

输入变量	状态字节QS的数位位置（对偶等价量）
I1	0 (2^0)
I2	1 (2^1)
I3	2 (2^2)
I4	3 (2^3)
I5	4 (2^4)
I6	5 (2^5)
I7	6 (2^6)
I8	7 (2^7)

多个块连接

块 连接	说明	预分配	取值范围	数
I1	二进制变量1	0	0/1	
I2	二进制变量2	0	0/1	
I3	二进制变量3	0	0/1	
I4	二进制变量4	0	0/1	
I5	二进制变量5	0	0/1	
I6	二进制变量6	0	0/1	
I7	二进制变量7	0	0/1	
I8	二进制变量8	0	0/1	
QS	状态字节	16#00	BYTE	

据

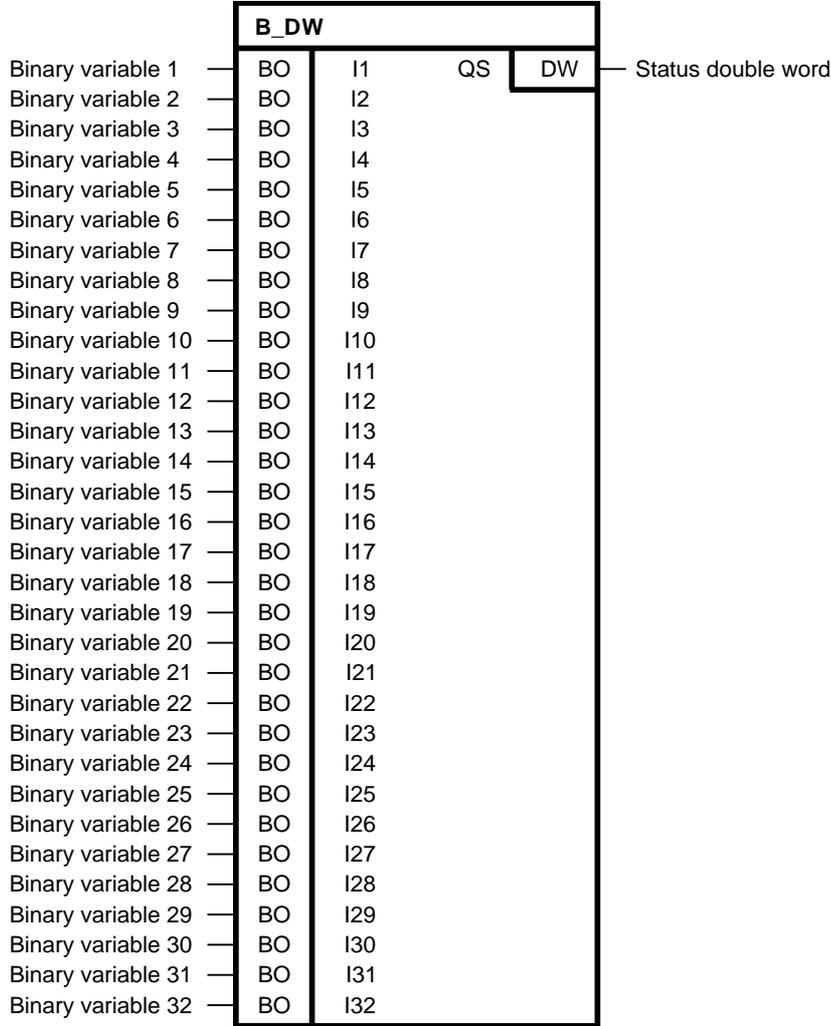
可在线插入	是
特别注意点	-

4.4 B_DW 32个二进制变量到状态双字的转换器

SIMOTION

SINAMICS

符号



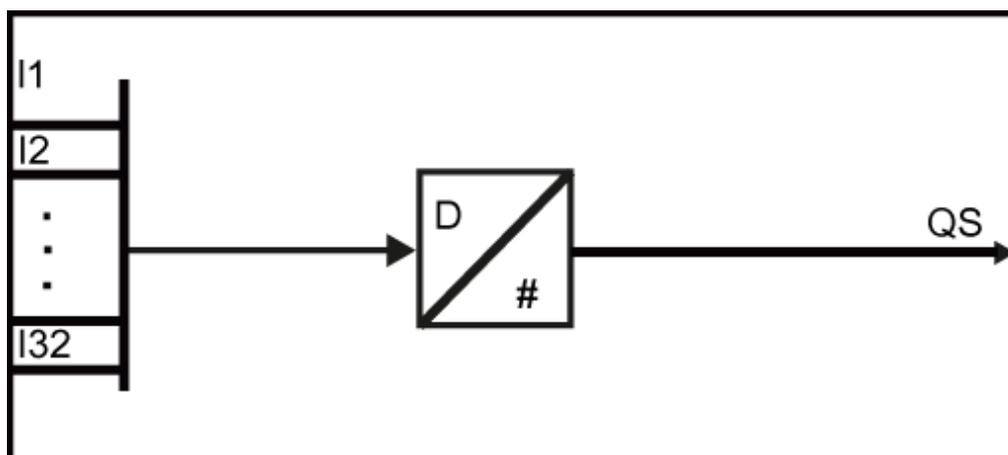
简要说明

由32个二进制变量生成状态双字

操作模式

该功能块组合I1到I32的二进制变量为1个状态双字，并将结果作为QS的输出值。每个I1到I32的二进制输入变量为生成状态字的对偶等价量 2^0 到 2^{31} 。

方框图



映射方案

输入参数	状态字节QS的数位位置（对偶等价量）
I1	0 (2^0)
I2	1 (2^1)
I3	2 (2^2)
...	...
I32	31 (2^{31})

多个块连接

块 连接	说明	预分配	取值范围	数
I1	二进制变量1	0	0/1	
I2	二进制变量2	0	0/1	
I3	二进制变量3	0	0/1	
I4	二进制变量4	0	0/1	
I5	二进制变量5	0	0/1	
I6	二进制变量6	0	0/1	
I7	二进制变量7	0	0/1	
I8	二进制变量8	0	0/1	
I9	二进制变量9	0	0/1	
I10	二进制变量10	0	0/1	
I11	二进制变量11	0	0/1	
I12	二进制变量12	0	0/1	
I13	二进制变量13	0	0/1	
I14	二进制变量14	0	0/1	
I15	二进制变量15	0	0/1	
I16	二进制变量16	0	0/1	

4.4 B_DW 32个二进制变量到状态双字的转换器

块 连接	说明	预分配	取值范围	数
I17	二进制变量17	0	0/1	
I18	二进制变量18	0	0/1	
I19	二进制变量19	0	0/1	
I20	二进制变量20	0	0/1	
I21	二进制变量21	0	0/1	
I22	二进制变量22	0	0/1	
I23	二进制变量23	0	0/1	
I24	二进制变量24	0	0/1	
I25	二进制变量25	0	0/1	
I26	二进制变量26	0	0/1	
I27	二进制变量27	0	0/1	
I28	二进制变量28	0	0/1	
I29	二进制变量29	0	0/1	
I30	二进制变量30	0	0/1	
I31	二进制变量31	0	0/1	
I32	二进制变量32	0	0/1	
QS	状态双字	16#00000000	DWORD	

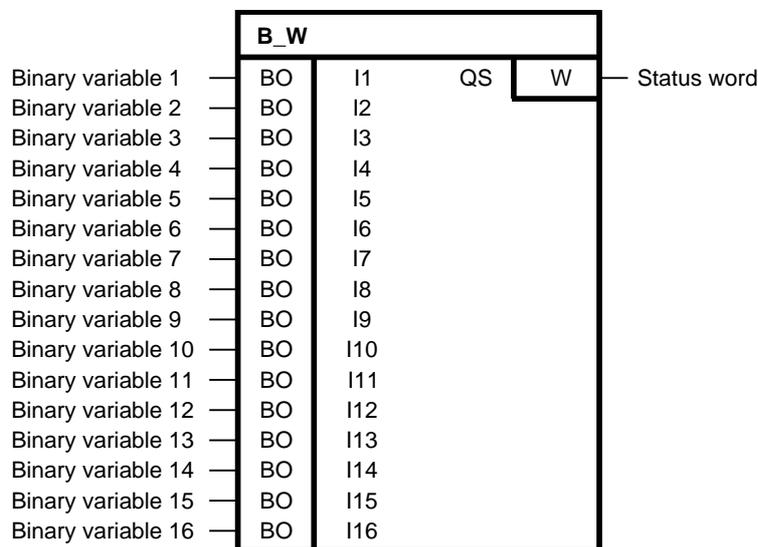
据

可在线插入	是
特别注意点	-

4.5 B_W 16个二进制变量到状态字的转换器

 SIMOTION SINAMICS

符号



简要说明

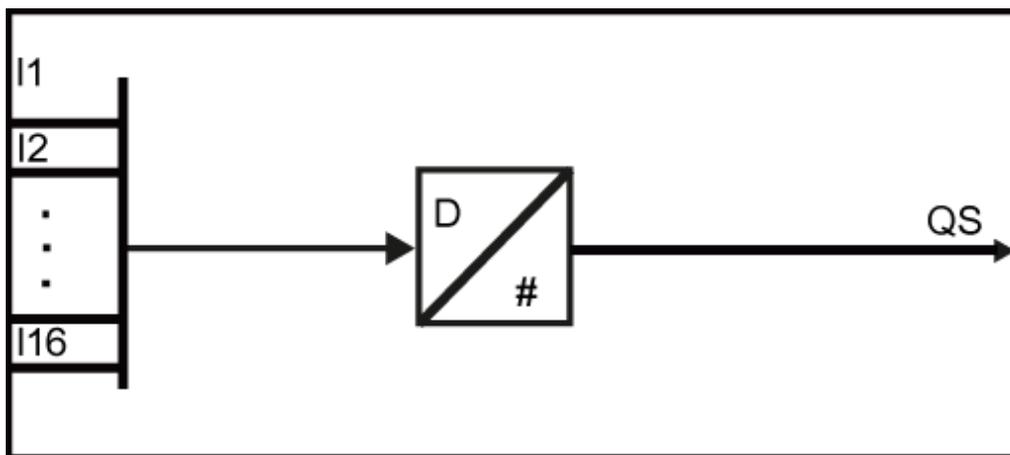
- 由16个二进制变量生成状态字

操作模式

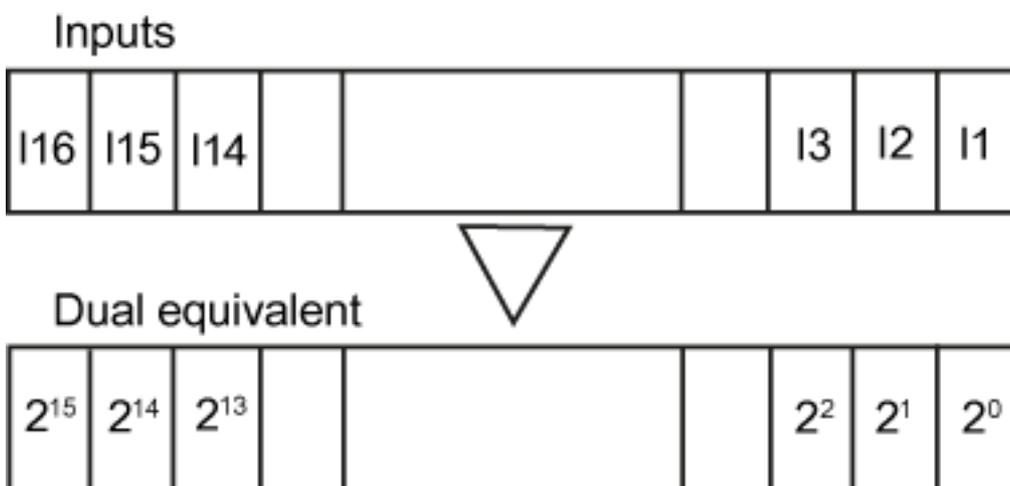
该功能块组合I1到I16的二进制变量为1个状态字，并将结果作为QS的输出值。
每个I1到I16的二进制输入变量为生成状态字的对偶等价量 2^0 到 2^{15} 。

4.5 B_W 16个二进制变量到状态字的转换器

方框图



转换方案



多个块连接

块 连接	说明	预分配	取值范围	数
I1	二进制变量1	0	0/1	
I2	二进制变量2	0	0/1	
I3	二进制变量3	0	0/1	
I4	二进制变量4	0	0/1	
I5	二进制变量5	0	0/1	
I6	二进制变量6	0	0/1	
I7	二进制变量7	0	0/1	
I8	二进制变量8	0	0/1	
I9	二进制变量9	0	0/1	
I10	二进制变量10	0	0/1	
I11	二进制变量11	0	0/1	

块 连接	说明	预分配	取值范围	数
I12	二进制变量12	0	0/1	
I13	二进制变量13	0	0/1	
I14	二进制变量14	0	0/1	
I15	二进制变量15	0	0/1	
I16	二进制变量16	0	0/1	
QS	状态字	16#0000	WORD	

据

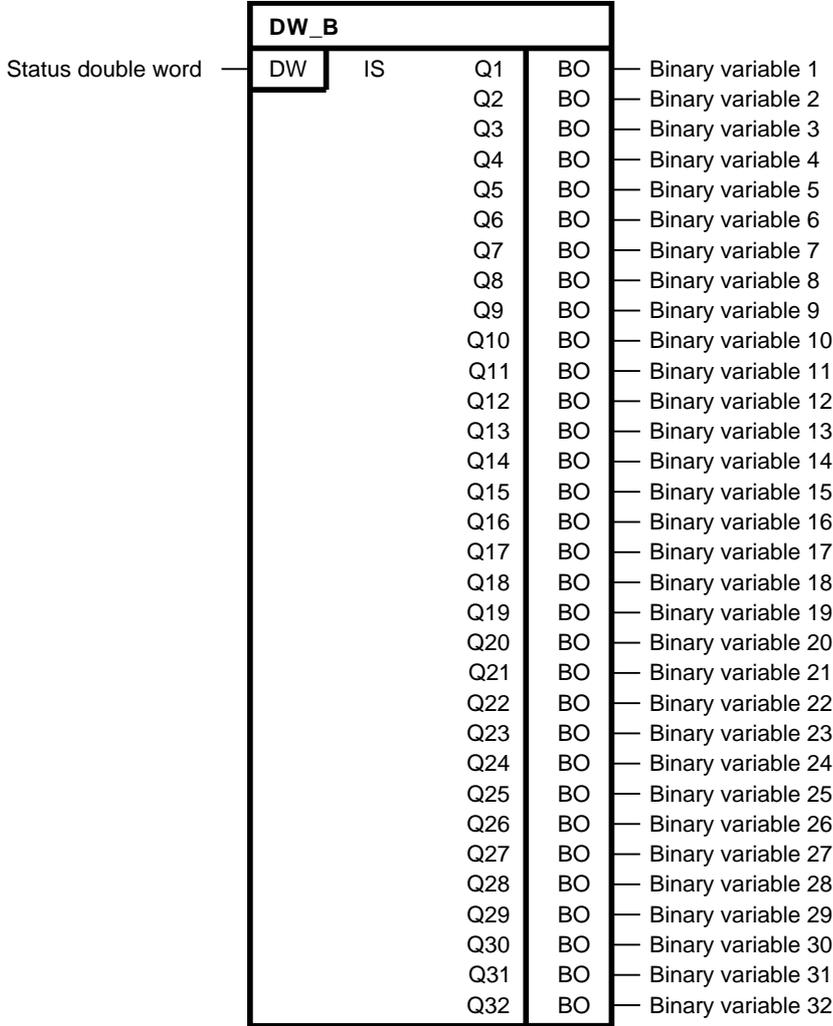
可在线插入	是
特别注意点	-

4.6 DW_B 状态双字到32个二进制变量的转换器

SIMOTION

SINAMICS

符号



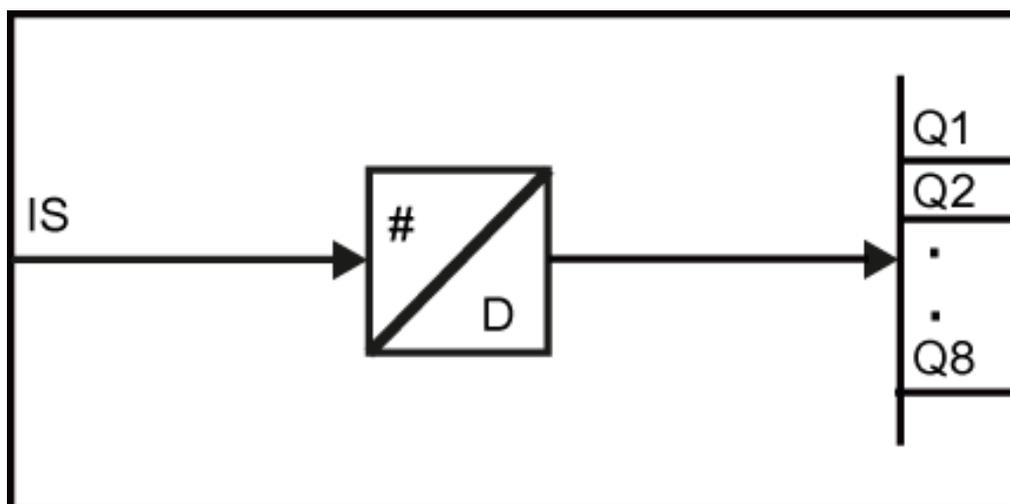
简要说明

- 状态双字解码为32个二进制变量

操作模式

该功能块将状态双字IS解码为32个二进制变量，并将结果作为Q1到Q32的输出值。
 输出Q1到Q32的二进制变量为状态字的每个对偶等价量 2^0 到 2^{31} 。

方框图



映射方案

状态双字IS的数位位置（对偶等价量）	输出变量
0 (2^0)	Q1
1 (2^1)	Q2
2 (2^2)	Q3
...	...
31 (2^{31})	Q32

多个块连接

块 连接	说明	预分配	取值范围	数
IS	状态双字	16#00000000	DWORD	
Q1	二进制变量1	0	0/1	
Q2	二进制变量2	0	0/1	
Q3	二进制变量3	0	0/1	
Q4	二进制变量4	0	0/1	
Q5	二进制变量5	0	0/1	
Q6	二进制变量6	0	0/1	
Q7	二进制变量7	0	0/1	
Q8	二进制变量8	0	0/1	
Q9	二进制变量9	0	0/1	
Q10	二进制变量10	0	0/1	
Q11	二进制变量11	0	0/1	
Q12	二进制变量12	0	0/1	
Q13	二进制变量13	0	0/1	
Q14	二进制变量14	0	0/1	

4.6 DW_B 状态双字到32个二进制变量的转换器

块 连接	说明	预分配	取值范围	数
Q15	二进制变量15	0	0/1	
Q16	二进制变量16	0	0/1	
Q17	二进制变量17	0	0/1	
Q18	二进制变量18	0	0/1	
Q19	二进制变量19	0	0/1	
Q20	二进制变量20	0	0/1	
Q21	二进制变量21	0	0/1	
Q22	二进制变量22	0	0/1	
Q23	二进制变量23	0	0/1	
Q24	二进制变量24	0	0/1	
Q25	二进制变量25	0	0/1	
Q26	二进制变量26	0	0/1	
Q27	二进制变量27	0	0/1	
Q28	二进制变量28	0	0/1	
Q29	二进制变量29	0	0/1	
Q30	二进制变量30	0	0/1	
Q31	二进制变量31	0	0/1	
Q32	二进制变量32	0	0/1	

据

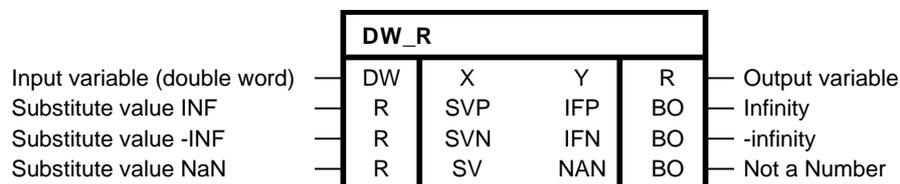
可在线插入	是
特别注意点	-

4.7 DW_R 接受位串为真值

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 该功能块接受输入的位串为实变量，并对值进行有效性检查。

操作模式

DW_R功能块接受输入的位串为实变量，并在输出Y处提供位串。

检查输入变量X的位模式。根据IEEE 754，当位模式对应表现为正负无穷或NaN时，相关二进制输出IFP、IFN或NAN都设为1，且各自指定的代入值在输出Y处生效。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量（双字）	16#00000000	DWORD	
SVP	代入值INF	3. 402823 E38	REAL	
SVN	代入值-INF	-3. 402823 E38	REAL	
SV	代入值NaN	0. 0	REAL	
IFP	正无穷	0	0/1	
IFN	负无穷	0	0/1	
NAN	非数值	0	0/1	
Y	输出变量	0. 0	REAL	

据

可在线插入	是
特别注意点	-

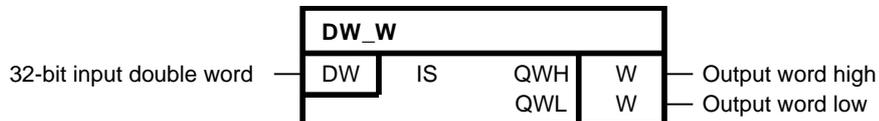
4.8 DW_W 状态双字到状态字的转换器

4.8 DW_W 状态双字到状态字的转换器

SIMOTION

SINAMICS

符号



简要说明

- 一个32位双字可分为两个16位字。

操作模式

输出变量根据以下规则计算：

$$QWL = IS \bmod 2^{16}$$

$$QWH = IS / 2^{16}$$

多个块连接

块 连接	说明	预分配	取值范围	数
IS	32位输入双字	16#00000000	DWORD	
QWH	输出字（高）	16#0000	WORD	
QWL	输出字（低）	16#0000	WORD	

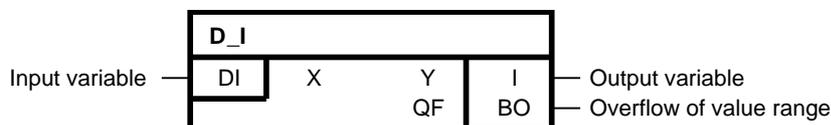
据

可在线插入	是
特别注意点	-

4.9 D_I DOUBLE INTEGER到INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- DOUBLE INTEGER变量到INTEGER变量的转换

操作模式

该功能块将DOUBLE INTEGER变量转换为INTEGER变量。也就是说，DOUBLE INTEGER输入变量的低字适用于输出变量Y。

当输入变量X的值超出输出变量Y的取值范围时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
Y	输出变量	0	INT	
QF	取值范围溢位	0	0/1	

据

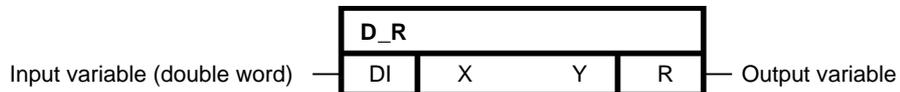
可在线插入	是
特别注意点	-

4.10 D_R DOUBLE INTEGER到REAL的转换器

4.10 D_R DOUBLE INTEGER到REAL的转换器

SIMOTION SINAMICS

符号



简要说明

- DOUBLE INTEGER变量到REAL变量的转换

操作模式

该功能块将DOUBLE INTEGER变量转换为REAL变量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量（双字）	0	DINT	
Y	输出变量	0.0	REAL	

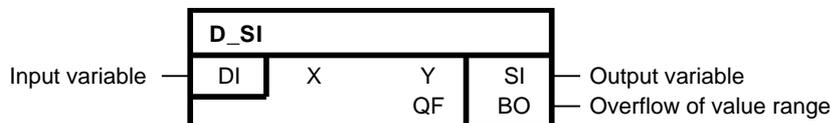
据

可在线插入	是
特别注意点	-

4.11 D_SI DOUBLE INTEGER到SHORT INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- DOUBLE INTEGER变量到SHORT INTEGER变量的转换

操作模式

该功能块将DOUBLE INTEGER变量转换为SHORT INTEGER变量。也就是说，DOUBLE INTEGER输入变量的最低有效字节适用于输出变量Y。

当输入变量X的值超出输出变量Y的取值范围时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
Y	输出变量	0	SINT	
QF	取值范围溢位	0	0/1	

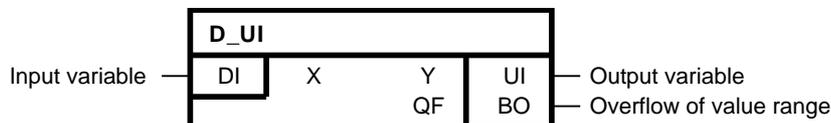
据

可在线插入	是
特别注意点	-

4.12 D_UI DOUBLE INTEGER到UNSIGNED INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- DOUBLE INTEGER变量到UNSIGNED INTEGER变量的转换

操作模式

该功能块将DOUBLE INTEGER变量转换为UNSIGNED INTEGER变量。也就是说，DOUBLE INTEGER输入变量的低字适用于输出变量。

当输入变量X的值超出输出变量Y的取值范围时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
Y	输出变量	0	UINT	
QF	取值范围溢位	0	0/1	

据

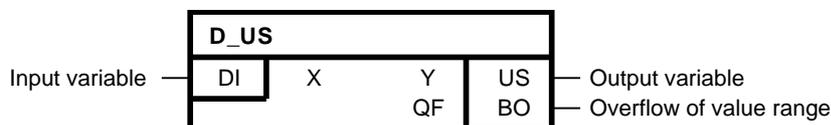
可在线插入	是
特别注意点	-

4.13 D_US DOUBLE INTEGER到UNSIGNED SHORT INTEGER的转换器

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- DOUBLE INTEGER变量到UNSIGNED SHORT INTEGER变量的转换

操作模式

该功能块将DOUBLE INTEGER变量转换为UNSIGNED SHORT INTEGER变量。也就是说，DOUBLE INTEGER输入变量的低字适用于输出变量。

当输入变量X的值超出输出变量Y的取值范围时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
Y	输出变量	0	USINT	
QF	取值范围溢位	0	0/1	

据

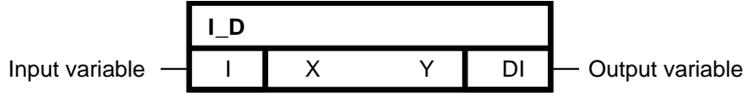
可在线插入	是
特别注意点	-

4.14 I_D INTEGER 到DOUBLE_INTEGER的转换器

4.14 I_D INTEGER 到DOUBLE_INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- INTEGER变量到DOUBLE_INTEGER变量的转换

操作模式

该功能块将INTEGER变量转换为DOUBLE_INTEGER变量。

数据类型INTEGER的输入变量复制到输出变量的低字。当输入变量为正数时，输出变量的高字取值为16#0000；相反，当为负数时，高字取值为16#FFFF。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
Y	输出变量	0	DINT	

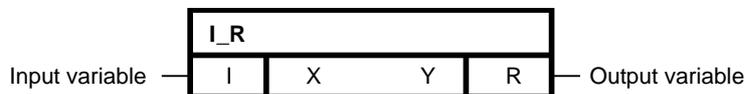
据

可在线插入	是
特别注意点	-

4.15 I_R INTEGER到REAL的转换器

 SIMOTION SINAMICS

符号



简要说明

- INTEGER变量到REAL变量的转换

操作模式

该功能块将INTEGER变量转换为REAL变量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
Y	输出变量	0.0	REAL	

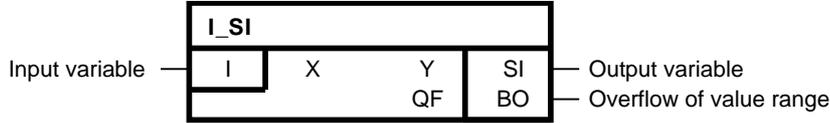
据

可在线插入	是
特别注意点	-

4.16 I_SI INTEGER到SHORT INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- INTEGER变量到SHORT INTEGER变量的转换

操作模式

该功能块将INTEGER变量转换为SHORT INTEGER变量。也就是说，INTEGER输入变量的低序位字节适用于输出变量Y。

当输入变量X的值超出输出变量Y的取值范围时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
Y	输出变量	0	SINT	
QF	取值范围溢位	0	0/1	

据

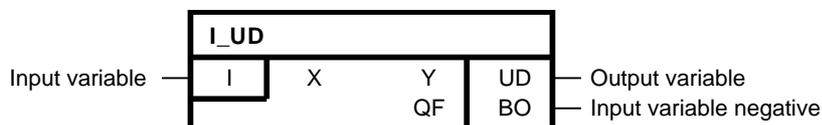
可在线插入	是
特别注意点	-

4.17 I_UD INTEGER到UNSIGNED DOUBLE INTEGER的转换器

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- INTEGER变量到UNSIGNED DOUBLE INTEGER变量的转换

操作模式

该功能块将INTEGER变量转换为UNSIGNED DOUBLE INTEGER变量。

数据类型INTEGER的输入变量复制到输出变量的低字。

输出变量的高字取值为16#0000。

当输入变量取值为负数时，设置 QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
Y	输出变量	0	UDINT	
QF	输入变量负数	0	0/1	

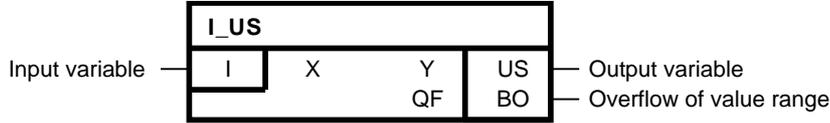
据

可在线插入	是
特别注意点	-

4.18 I_US INTEGER到UNSIGNED SHORT INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- INTEGER变量到UNSIGNED SHORT INTEGER变量的转换

操作模式

该功能块将INTEGER变量转换为UNSIGNED SHORT INTEGER变量。也就是说，DOUBLE INTEGER输入变量的最低有效字节适用于输出变量Y。

当输入变量X的值超出输出变量Y的取值范围时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
Y	输出变量	0	USINT	
QF	取值范围溢位	0	0/1	

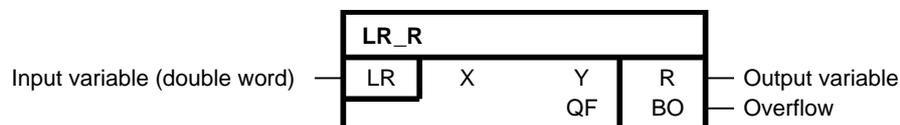
据

可在线插入	是
特别注意点	-

4.19 LR_R LONG REAL到REAL的转换器

 SIMOTION SINAMICS

符号



简要说明

- LONG REAL变量到REAL变量的转换

操作模式

该功能块将LONG REAL变量转换为REAL变量。转换结果受限于REAL型数据的最大范围。当输出变量已经受限时，设置QF = 1。

说明

转换可能会导致输入值的准确性降低。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量（双字）	0	LREAL	
Y	输出变量	0.0	REAL	
QF	溢位	0	0/1	

据

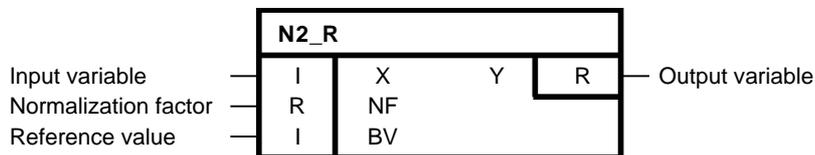
可在线插入	是
特别注意点	-

4.20 N2_R 转换16位定点格式 (N2) 为REAL

SIMOTION

SINAMICS

符号



简要说明

- 16位定点变量到REAL变量的转换。当X和BV = 16384（对应于标准化Profidrive行规的100%）时，输出Y取值为输入NF处的值。

操作模式

根据以下公式，映射输入变量X给输出Y：

$$Y = \frac{(X \cdot NF)}{BV}$$

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
NF	标准化因子	1.0	REAL	
BV	参考值	16384	INT	
Y	输出变量	0.0	REAL	

据

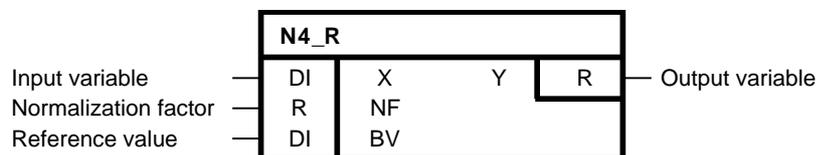
可在线插入	是
特别注意点	-

4.21 N4_R 转换32位定点格式 (N4) 为REAL

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 32位定点变量到REAL变量的转换。当X和BV = 1073741824（对应于标准化Profidrive行规的100%）时，输出Y取值为输入NF处的值。

操作模式

根据以下公式，映射输入变量X给输出Y：

$$Y = \frac{(X \cdot NF)}{BV}$$

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
NF	标准化因子	1.0	REAL	
BV	参考值	1073741824	DINT	
Y	输出变量	0.0	REAL	

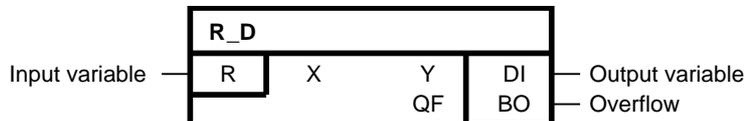
据

可在线插入	是
特别注意点	-

4.22 R_D REAL到DOUBLE INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- REAL变量到DOUBLE INTEGER变量的转换

操作模式

该功能块将REAL变量转换为DOUBLE INTEGER变量。转换时，会去掉输入变量的小数位。

说明：不进行四舍五入。

计算结果限于输出变量的数据类型，对应于 -2^{31} 或 $2^{31}-1$ 。当输出变量已受限时，则设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	0	DINT	
QF	溢位	0	0/1	

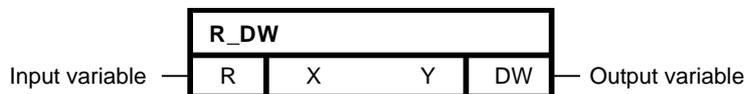
据

可在线插入	是
特别注意点	-

4.23 R_DW 以DWORD进行的位串传递

 SIMOTION SINAMICS

符号



简要说明

- 该功能块将输入变量的位串拷贝给输出变量。

操作模式

该功能块将输入变量X的位串拷贝给输出变量Y。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	16#00000000	DWORD	

据

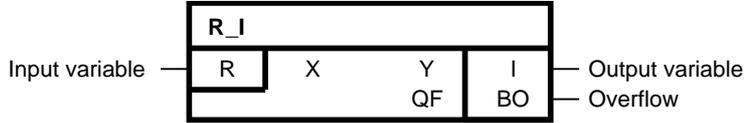
可在线插入	是
特别注意点	-

4.24 R_I REAL到INTEGER的转换器

4.24 R_I REAL到INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- REAL变量到INTEGER变量的转换

操作模式

该功能块将REAL变量转换为INTEGER变量。转换时，会去掉输入变量的小数位。该数字不进行四舍五入。计算结果限于输出变量的数据类型，对应于+32767或-32768。当输出变量已受限时，则设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	0	INT	
QF	溢位	0	0/1	

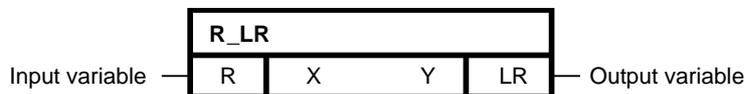
据

可在线插入	是
特别注意点	-

4.25 R_LR REAL到LONG REAL的转换器

 SIMOTION SINAMICS

符号



简要说明

- REAL变量到LONG REAL变量的转换

操作模式

该功能块将REAL变量转换为LONG REAL变量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	0.0	LREAL	

据

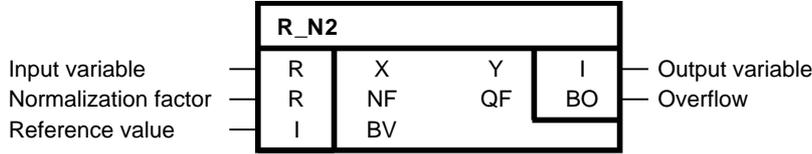
可在线插入	是
特别注意点	-

4.26 R_N2 转换REAL为16位定点格式 (N2)

4.26 R_N2 转换REAL为16位定点格式 (N2)

SIMOTION SINAMICS

符号



简要说明

- REAL变量到16位定点变量的转换 当X = NF且BV = 16384（默认值）时，输出Y取值16384（对应标准化Profidrive行规的100%）。

操作模式

根据以下公式（结果进行四舍五入），映射输入变量X给输出Y：

$$Y = \frac{X \cdot BV}{NF}$$

Y取值范围限于 $-32768 \leq Y \leq 32767$ （对应 $-200\% \leq Y < 200\%$ ）。

当X由于超出取值范围而不能映射给Y或已经设置NF = 0时，输出QF（溢位）设为1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
NF	标准化因子	1.0	REAL	
BV	参考值	16384	INT	
Y	输出变量	0	INT	
QF	溢位	0	0/1	

据

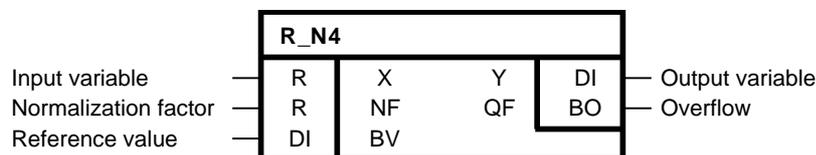
可在线插入	是
特别注意点	-

4.27 R_N4 转换REAL为32位定点格式 (N4)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- REAL变量到32位定点变量的转换 当X = NF且BV = 1073741824（默认值）时，输出Y取值1073741824（对应标准化Profidrive行规的100%）。

操作模式

根据以下公式（结果进行四舍五入），映射输入变量X给输出Y：

$$Y = \frac{X \cdot BV}{NF}$$

Y取值范围限于 $-2147483648 \leq Y \leq 2147483647$ （十进制）或 $16\#8000000 \leq Y \leq 16\#7FFFFFFF$ （16进制）（对应 $-200\% \leq Y < 200\%$ ）。

当X由于超出取值范围而不能映射给Y或已经设置NF = 0时，输出QF（溢位）设为1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
NF	标准化因子	1.0	REAL	
BV	参考值	1073741824	DINT	
Y	输出变量	0	DINT	
QF	溢位	0	0/1	

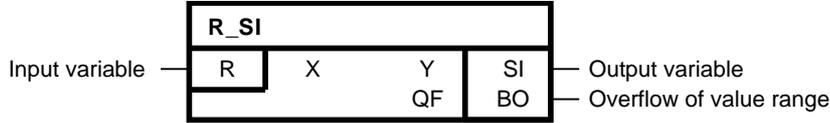
据

可在线插入	是
特别注意点	-

4.28 R_SI REAL到SHORT INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- REAL变量到SHORT INTEGER变量的转换

操作模式

该功能块将REAL变量转换为SHORT INTEGER变量。转换时，会去掉输入变量的小数位。该数字不进行四舍五入。计算结果限于输出变量的数据类型，对应于-128或127。当输出变量已受限时，则设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	0	SINT	
QF	取值范围溢位	0	0/1	

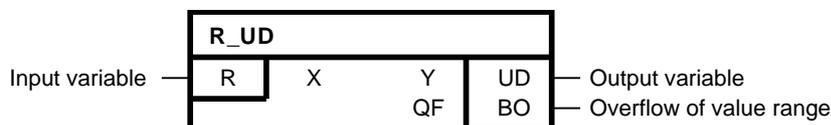
据

可在线插入	是
特别注意点	-

4.29 R_UD REAL到UNSIGNED DOUBLE INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- REAL变量到UNSIGNED DOUBLE INTEGER变量的转换

操作模式

该功能块将REAL变量转换为UNSIGNED DOUBLE INTEGER变量。转换时，会去掉输入变量的小数位。该数字不进行四舍五入。根据输出变量的数据类型，计算结果限于0或 $2^{32}-1$ 。当输出变量已受限时，则设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	0	UDINT	
QF	取值范围溢位	0	0/1	

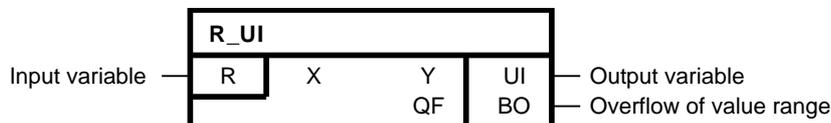
据

可在线插入	是
特别注意点	-

4.30 R_UI REAL到UNSIGNED INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- REAL变量到UNSIGNED INTEGER变量的转换

操作模式

该功能块将REAL变量转换为UNSIGNED INTEGER变量。转换时，会去掉输入变量的小数位。该数字不进行四舍五入。根据输出变量的数据类型，计算结果限于0或 $2^{16}-1$ 。当输出变量已受限时，则设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	0	UINT	
QF	取值范围溢位	0	0/1	

据

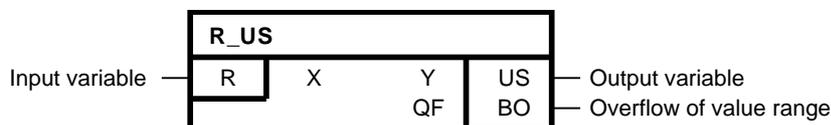
可在线插入	是
特别注意点	-

4.31 R_US REAL到UNSIGNED SHORT INTEGER的转换器

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- REAL变量到UNSIGNED SHORT INTEGER变量的转换

操作模式

该功能块将REAL变量转换为UNSIGNED SHORT INTEGER变量。

转换时，会去掉输入变量的小数位。该数字不进行四舍五入。根据输出变量的数据类型，计算结果限于0或 2^8-1 。当输出变量已受限时，则设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
Y	输出变量	0	USINT	
QF	取值范围溢位	0	0/1	

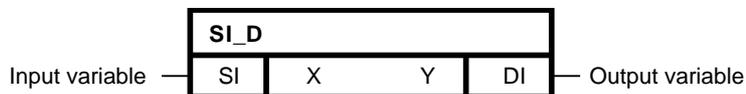
据

可在线插入	是
特别注意点	-

4.32 SI_D SHORT INTEGER 到DOUBLE_INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- SHORT INTEGER变量到DOUBLE INTEGER变量的转换

操作模式

该功能块将SHORT INTEGER变量转换为DOUBLE INTEGER变量。

数据类型SHORT INTEGER的输入变量拷贝至输出变量的低序位字节。当输入变量为正数时，输出变量的高序位字节取值为16#00；相反，当为负数时，高序位字节取值为16#FF。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	SINT	
Y	输出变量	0	DINT	

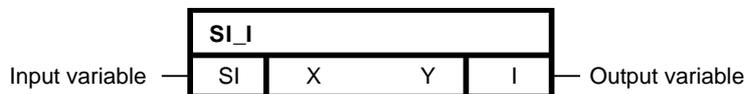
据

可在线插入	是
特别注意点	-

4.33 SI_I SHORT INTEGER 到INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- SHORT INTEGER变量到INTEGER变量的转换

操作模式

该功能块将SHORT INTEGER变量转换为INTEGER变量。

数据类型SHORT INTEGER的输入变量拷贝至输出变量的低序位字节。当输入变量为正数时，输出变量的高序位字节取值为16#00；相反，当为负数时，高序位字节取值为16#FF。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	SINT	
Y	输出变量	0	INT	

据

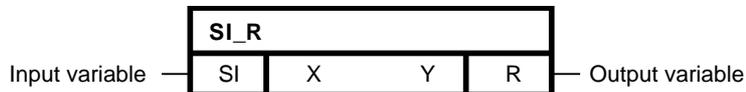
可在线插入	是
特别注意点	-

4.34 SI_R SHORT INTEGER到REAL的转换器

4.34 SI_R SHORT INTEGER到REAL的转换器

SIMOTION SINAMICS

符号



简要说明

- SHORT INTEGER变量到REAL变量的转换

操作模式

该功能块将SHORT INTEGER变量转换为REAL变量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	SINT	
Y	输出变量	0.0	REAL	

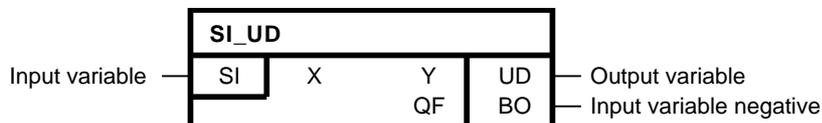
据

可在线插入	是
特别注意点	-

4.35 SI_UD SHORT INTEGER到UNSIGNED DOUBLE INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- SHORT INTEGER变量到UNSIGNED DOUBLE INTEGER变量的转换

操作模式

该功能块将SHORT INTEGER变量转换为UNSIGNED DOUBLE INTEGER变量。

数据类型SHORT INTEGER的输入变量拷贝至输出变量的低序位字节。输出变量的高序位字节取值为16#00。当输入变量取值为负数时，设置 QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	SINT	
Y	输出变量	0	UDINT	
QF	输入变量负数	0	0/1	

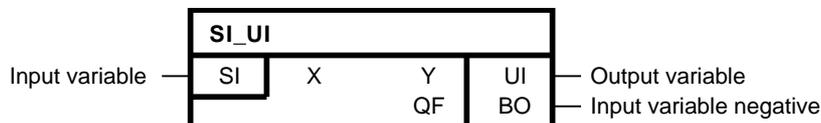
据

可在线插入	是
特别注意点	-

4.36 SI_UI SHORT INTEGER到UNSIGNED INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- SHORT INTEGER变量到UNSIGNED INTEGER变量的转换

操作模式

该功能块将SHORT INTEGER变量转换为UNSIGNED INTEGER变量。

数据类型SHORT INTEGER的输入变量拷贝至输出变量的低序位字节。输出变量的高序位取值为16#00。当输入变量的值为负数时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	SINT	
Y	输出变量	0	UDINT	
QF	输入变量负数	0	0/1	

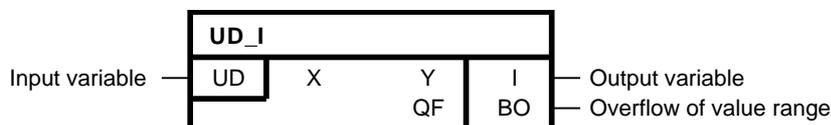
据

可在线插入	是
特别注意点	-

4.37 UD_I UNSIGNED DOUBLE INTEGER到INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- UNSIGNED DOUBLE INTEGER变量到INTEGER变量的转换

操作模式

该功能块将UNSIGNED DOUBLE INTEGER变量转换为INTEGER变量。也就是说，UNSIGNED DOUBLE INTEGER输入变量的低序位字在输出变量Y处被替代。

当输入变量X的值超出输出变量Y的取值范围时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	UDINT	
Y	输出变量	0	INT	
QF	取值范围溢位	0	0/1	

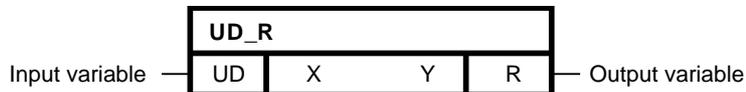
据

可在线插入	是
特别注意点	-

4.38 UD_R UNSIGNED DOUBLE INTEGER到REAL的转换器

SIMOTION SINAMICS

符号



简要说明

- UNSIGNED DOUBLE INTEGER变量到REAL变量的转换

操作模式

该功能块将UNSIGNED DOUBLE INTEGER变量转换为REAL变量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	UDINT	
Y	输出变量	0.0	REAL	

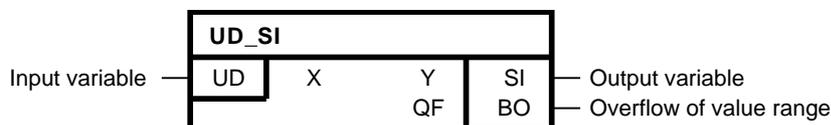
据

可在线插入	是
特别注意点	-

4.39 UD_SI UNSIGNED DOUBLE INTEGER到SHORT INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- UNSIGNED DOUBLE INTEGER变量到SHORT INTEGER变量的转换

操作模式

该功能块将UNSIGNED DOUBLE INTEGER变量转换为SHORT INTEGER变量。也就是说，UNSIGNED DOUBLE INTEGER输入变量的低序位字在输出变量Y处被替代。当输入变量X的值超出输出变量Y的取值范围时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	UDINT	
Y	输出变量	0	SINT	
QF	取值范围溢位	0	0/1	

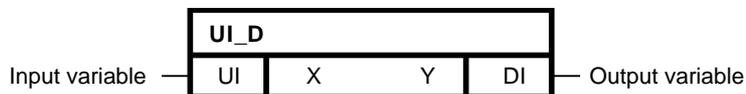
据

可在线插入	是
特别注意点	-

4.40 UI_D UNSIGNED INTEGER 到DOUBLE_INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- UNSIGNED INTEGER变量到DOUBLE INTEGER变量的转换

操作模式

该功能块将UNSIGNED INTEGER变量转换为DOUBLE INTEGER变量。
数据类型UNSIGNED INTEGER的输入变量复制到输出变量Y的低字。高字取值为16#0000。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	UINT	
Y	输出变量	0	DINT	

据

可在线插入	是
特别注意点	-

4. 41 UI_R UNSIGNED INTEGER到REAL的转换器

 SIMOTION SINAMICS

符号



简要说明

- UNSIGNED INTEGER变量到REAL变量的转换

操作模式

该功能块将UNSIGNED INTEGER变量转换为REAL变量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	UINT	
Y	输出变量	0.0	REAL	

据

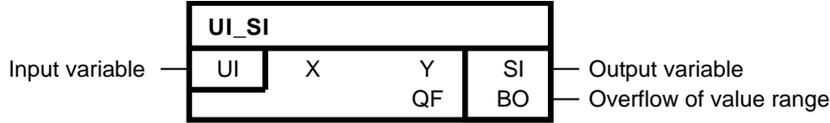
可在线插入	是
特别注意点	-

4.42 UI_SI UNSIGNED INTEGER到SHORT INTEGER的转换器

4.42 UI_SI UNSIGNED INTEGER到SHORT INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- UNSIGNED INTEGER变量到SHORT INTEGER变量的转换

操作模式

该功能块将UNSIGNED INTEGER变量转换为SHORT INTEGER变量。也就是说，UNSIGNED INTEGER输入变量的低字节在输出变量Y处被替代。当输入变量X的值超出输出变量Y的取值范围时，设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	UINT	
Y	输出变量	0	SINT	
QF	取值范围溢位	0	0/1	

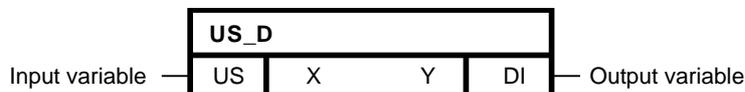
据

可在线插入	是
特别注意点	-

4.43 US_D UNSIGNED SHORT INTEGER 到DOUBLE_INTEGER的转换器

 SIMOTION SINAMICS

符号



简要说明

- UNSIGNED SHORT INTEGER变量到DOUBLE INTEGER变量的转换

操作模式

该功能块将UNSIGNED SHORT INTEGER变量转换为DOUBLE INTEGER变量。

数据类型SHORT SHORT INTEGER的输入变量拷贝至输出变量Y的低序位字节。 剩余的高序位字节取值为16#00。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	USINT	
Y	输出变量	0	DINT	

据

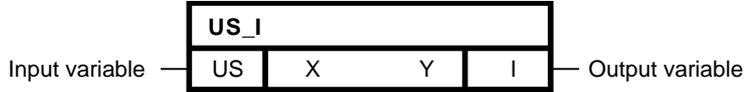
可在线插入	是
特别注意点	-

4.44 US_I UNSIGNED SHORT INTEGER 到INTEGER的转换器

4.44 US_I UNSIGNED SHORT INTEGER 到INTEGER的转换器

SIMOTION SINAMICS

符号



简要说明

- UNSIGNED SHORT INTEGER变量到INTEGER变量的转换

操作模式

该功能块将UNSIGNED SHORT INTEGER变量转换为INTEGER变量。
数据类型SHORT SHORT INTEGER的输入变量拷贝至输出变量Y的低序位字节。 剩余的高序位字节取值为16#00。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	USINT	
Y	输出变量	0	INT	

据

可在线插入	是
特别注意点	-

4.45 US_R UNSIGNED SHORT INTEGER到REAL的转换器

 SIMOTION SINAMICS

符号



简要说明

- UNSIGNED SHORT INTEGER变量到REAL变量的转换

操作模式

该功能块将UNSIGNED SHORT INTEGER变量转换为REAL变量。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	USINT	
Y	输出变量	0.0	REAL	

据

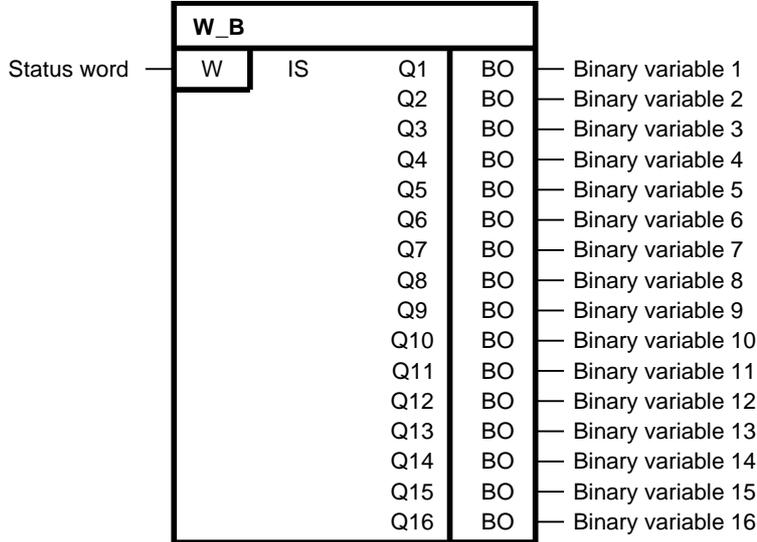
可在线插入	是
特别注意点	-

4.46 W_B 状态字到16个二进制变量的转换器

SIMOTION

SINAMICS

符号

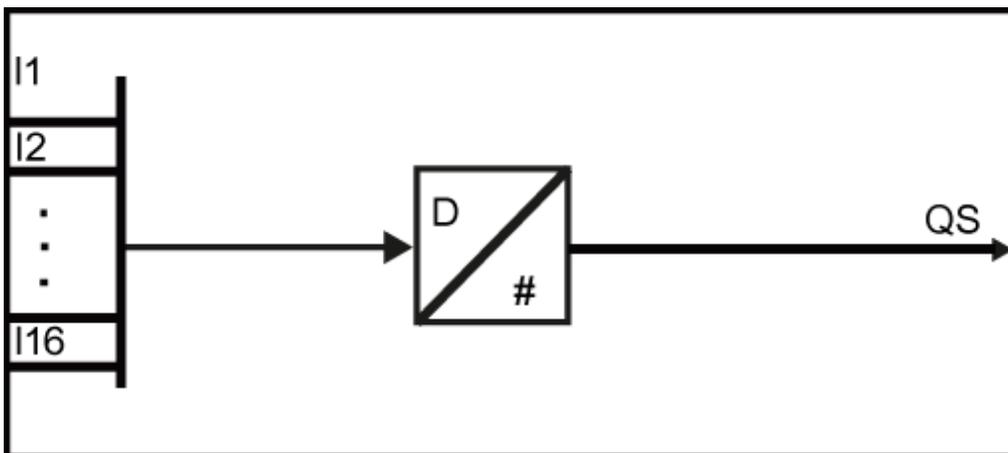


简要说明

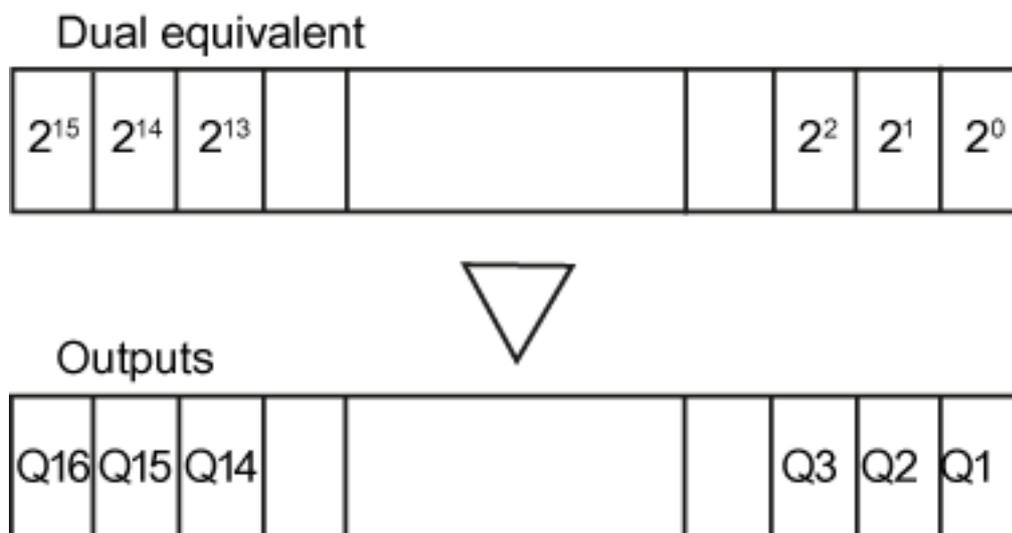
- 状态字解码为16个二进制变量

操作模式

该功能块将状态字IS解码为16个二进制变量，并将结果作为Q1到Q16的输出值。
输出Q1到Q16的二进制变量为状态字的每个对偶等价量 2^0 到 2^{15} 。



转换方案



多个块连接

块 连接	说明	预分配	取值范围	数
IS	状态字	16#0000	WORD	
Q1	二进制变量1	0	0/1	
Q2	二进制变量2	0	0/1	
Q3	二进制变量3	0	0/1	
Q4	二进制变量4	0	0/1	
Q5	二进制变量5	0	0/1	
Q6	二进制变量6	0	0/1	
Q7	二进制变量7	0	0/1	
Q8	二进制变量8	0	0/1	
Q9	二进制变量9	0	0/1	
Q10	二进制变量10	0	0/1	
Q11	二进制变量11	0	0/1	
Q12	二进制变量12	0	0/1	
Q13	二进制变量13	0	0/1	
Q14	二进制变量14	0	0/1	
Q15	二进制变量15	0	0/1	
Q16	二进制变量16	0	0/1	

据

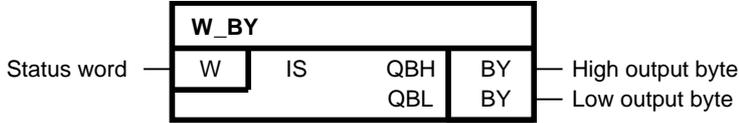
可在线插入	是
特别注意点	-

4.47 W_BY 状态字到状态字节的转换器

SIMOTION

SINAMICS

符号



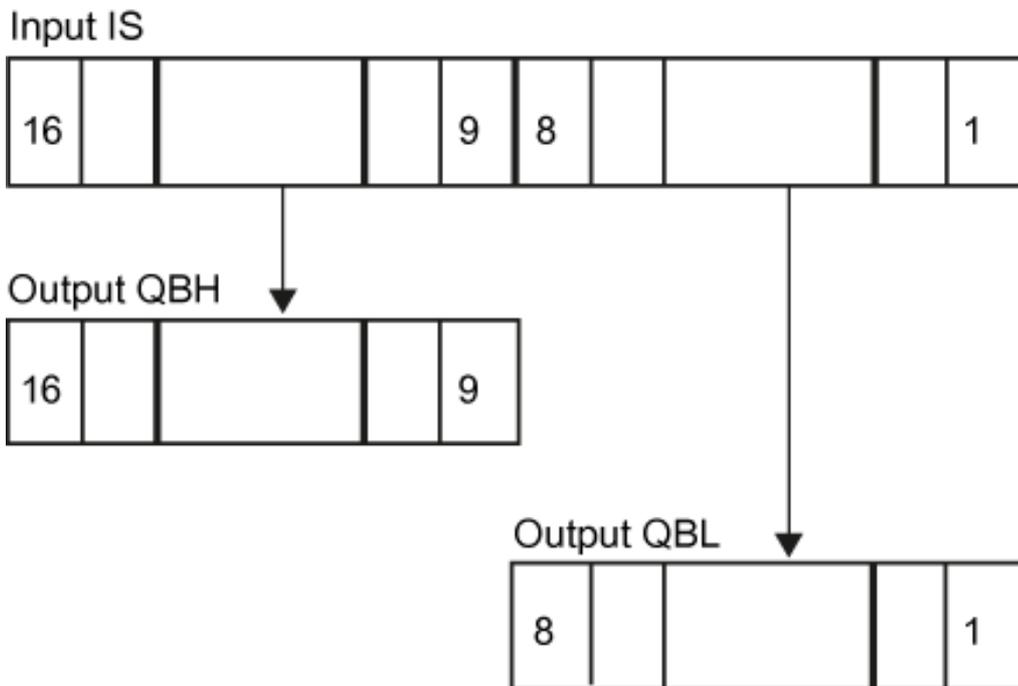
简要说明

- 1个字到2字节字的转换

操作模式

该功能块在IS处将输入字分为两个字节。两个字节可以通过SBQ功能块输出给I/O。输入IS字的高序位字节在QBH输出端输出，低序位字节在QBL输出端输出（参见如下转换方案）：

转换方案



多个块连接

块 连接	说明	预分配	取值范围	数
IS	状态字	16#0000	WORD	
QBH	高输出字节	16#00	BYTE	
QBL	低输出字节	16#00	BYTE	

据

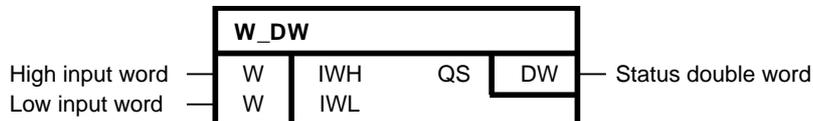
可在线插入	是
特别注意点	-

4.48 W_DW 状态字到状态双字的转换器

4.48 W_DW 状态字到状态双字的转换器

SIMOTION SINAMICS

符号



简要说明

- 两个16位字复制为一个32位双字

操作模式

根据以下公式映射输入变量：

$$QS = (IWL + IWH) * 2^{16}$$

到输出QS。

多个块连接

块 连接	说明	预分配	取值范围	数
IWH	高输入字	16#0000	WORD	
IWL	低输入字	16#0000	WORD	
QS	状态双字	16#00000000	DWORD	

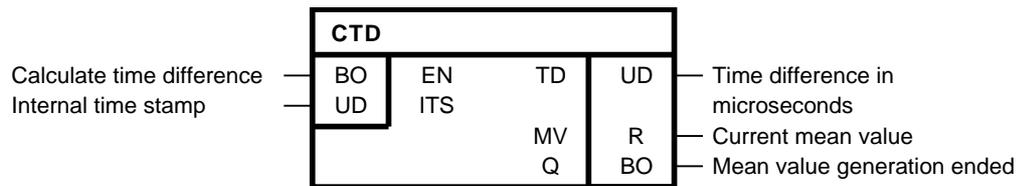
据

可在线插入	是
特别注意点	-

5.1 CTD 由内部时间戳确定时差

SIMOTION SINAMICS

符号



简要说明

用于确定微秒级时差的功能块。

操作模式

当EN = 1时，相对于时间戳ITS的时差可以确定，并在输出端TD输出。时间戳IST必须首先由GTS功能块确定。EN的正沿启动TD的平均值生成，并在MV端输出。在一万次平均值计算后，平均值生成结束，输出Q设置为1。当输入EN = 0时，平均值生成和输出Q则复位。输出TD和MV保留它们的最后一个值。

说明

最高可以测量1秒的时差。

多个块连接

块 连接	说明	预分配	取值范围	数
EN	计算时差	0	0/1	
ITS	内部时间戳	0	UDINT	
TD	微秒级时差	0	UDINT	
MV	当前平均值	0	REAL	
Q	平均值生成结束	0	0/1	

5.1 CTD 由内部时间戳确定时差

据

可在线加载	是
特别注意点	-

5.2 GTS 读出时间戳

 SIMOTION

 SINAMICS

符号



简要说明

为确定运行时间而读出内部时间戳的功能块。确定后的时间戳可以在CTD功能块处体现，以计算微秒级的时差。

操作模式

当EN = 1时，内部时间戳可以确定，并在输出端TS输出。当预设EN = 0时，最后确定的时间戳在输出端TS输出。

多个块连接

块 连接	说明	预分配	取值范围	数
EN	输出时间戳	0	0/1	
ITS	内部时间戳	0	UDINT	

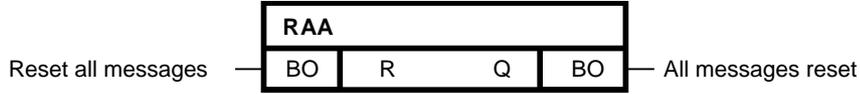
据

可在线加载	是
特别注意点	-

5.3 RAA 复位所有消息

SIMOTION SINAMICS

符号



简要说明

通过RAA (Reset all alarm: 复位所有告警) 功能块复位所有有效消息。

操作模式

当输入R = 1时，复位所有有效消息。
输出Q表示已经进行了复位。

多个块连接

块 连接	说明	预分配	取值范围	数
R	复位所有消息	0	0/1	
Q	所有消息复位	0	0/1	

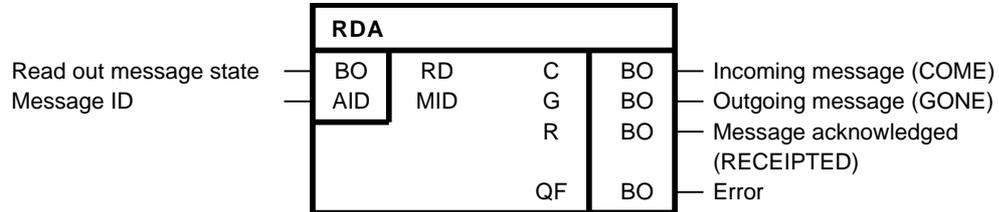
据

SIMOTION	4.3
SINAMICS	-
可在线插入	是
特别注意点	-

5.4 RDA 读出消息

 SIMOTION SINAMICS

符号



简要说明

该功能块读出消息的状态及其确认状态。

操作模式

- 消息通过SIMOTION SCOUT进行配置，并通过项目唯一ID进行引用。
- 输入MID包含该消息ID，例如：_alarm.Message。
- 当输入RD=1时，消息的状态是确定的。
- 可以改变消息ID。在每个周期都读出在输入MID处指定的消息ID的状态。
- 该输出表示消息的状态。
可以进行如下组合：

C (接收消息)	G (发出消息)	R (确认消息)	表示含义
0	1	0	发出消息，未确认
1	0	0	接收消息，未确认
1	0	1	接收消息，已确认
0	0	0	非消息缓存内的消息*)

- *) 非消息缓存内的消息 - 主要有三个原因
- 消息从未被触发
 - 通过AlarmS触发消息，但也消
 - 通过_AlarmSq触发消息，消失并在显示设备上

当RD=1时，刷新输出。当RD=0时，保持消息缓存的最后状态。
发生错误时，设置输出Q，如：未配置消息ID。

说明

无需确认的告警在复位后会标记为已确认。

5.4 RDA 读出消息

多个块连接

块 连接	说明	预分配	取值范围	数
RD	读出消息状态	0	0/1	
MID	消息ID	STRUCTALARMID#NIL	StructAlarmId	
C	接收消息 (COME)	0	0/1	
G	发出消息 (GONE)	0	0/1	
R	确认消息 (RECEIPTED)	0	0/1	
QF	错误	0	0/1	

据

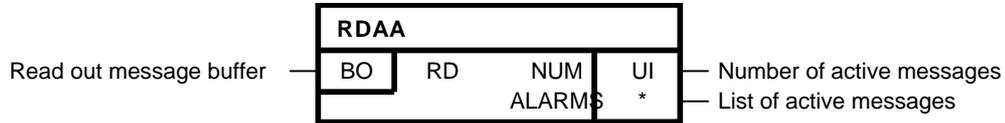
SIMOTION	4.3
SINAMICS	-
可在线插入	是
特别注意点	-

5.5 RDAA 读出所有消息

☑ SIMOTION

☐ SINAMICS

符号



简要说明

读出SIMOTION目标设备中的有效消息列表。

操作模式

- 输入RD的上升沿发起读出所有有效消息。
- 在NUM输出处返回有效消息数。
- 在ALARMS输出处显示一个含最多40个有效消息的域。
每个告警的显示如下：
 - 消息ID
 - 不可确认消息的标识符（0），可确认消息的标识符（1）
 - 告警状态： OUTGOING_ALARM（0）， INCOMING_ALARM（1）。
 当RD =1时，刷新输出。当RD=0时，保持消息缓存的最后状态。

多个块连接

块 连接	说明	预分配	取值范围	数
RD	读出消息缓冲	0	0/1	
NUM	有效消息数	0	0..40	
ALARMS	有效消息列表			
ALARMS[]	最多可以有40个有效消息。	0		
ALARMS[]. Id	消息ID	STRUCTALARMID#NIL	StructAlarmId	
ALARMS[]. type	对应enumAlarmIdType（0: ALARM_S, 1: ALARM_SQ）	0	0/1	
ALARMS[]. InOut	对应enumAlarmState OUTGOING_ALARM（0）， INCOMING_ALARM（1）	0	0/1	

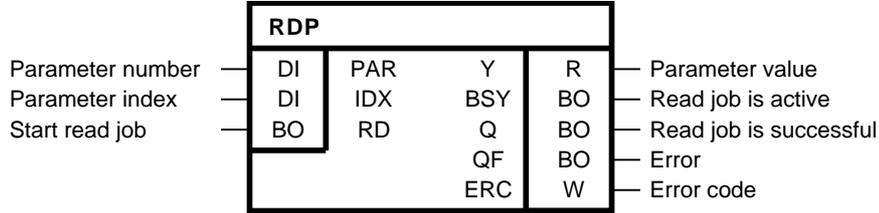
据

SIMOTION	4.3
SINAMICS	–
可在线插入	是
特别注意点	–

5.6 RDP 读取驱动参数 (REAL型)

SIMOTION SINAMICS

符号



简要说明

该功能块在本地驱动对象上启动REAL型驱动参数的异步读取。

操作模式

读出的参数编号和参数索引分别在输入PAR和IDX处体现。当参数未索引时，必须设置IDX = 0。参数总是在驱动对象上读取，而功能块图在该驱动对象上计算。不可以在多个驱动对象上存取参数。

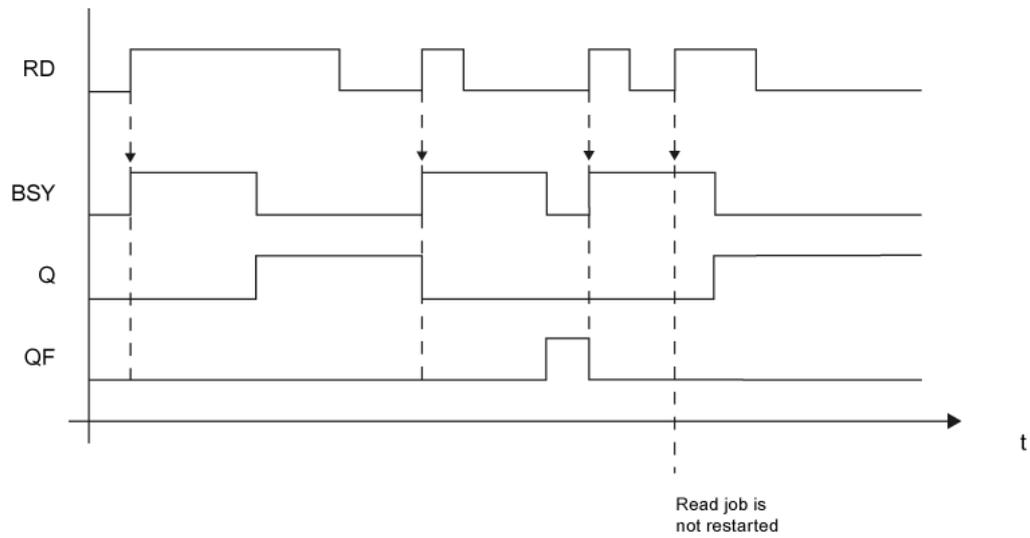
在输入RD的正沿上启动异步读任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在读任务执行时，忽略输入RD的其他正沿。

输出Q = 1表示参数已经成功读取，且参数值在输出Y处可用。在读取新值之前，Y会保留其当前的值。当在存取过程中发生错误时，通过QF = 1来表示。输出Y保留其最后的值。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	$0..2^{16}$	
IDX	参数索引	0	$0..2^{16}$	
RD	开始读任务	0	0/1	
Y	参数值	0.0	REAL	
BSY	执行读任务	0	0/1	
Q	成功完成读任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	DWORD	

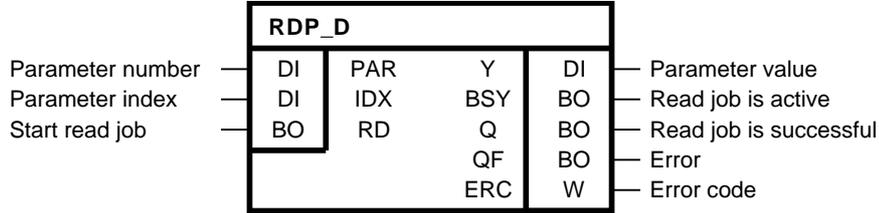
据

可在线加载	是
特别注意点	-

5.7 RDP_D 读取驱动参数 (DOUBLE INTEGER型)

SIMOTION SINAMICS

符号



简要说明

该功能块在本地驱动对象上启动DOUBLE INTEGER型驱动参数的异步读取。

操作模式

读出的参数编号和参数索引分别在输入PAR和IDX处体现。当参数未索引时，必须设置IDX = 0。参数总是在驱动对象上读取，而功能块图在该驱动对象上计算。不可以在多个驱动对象上存取参数。

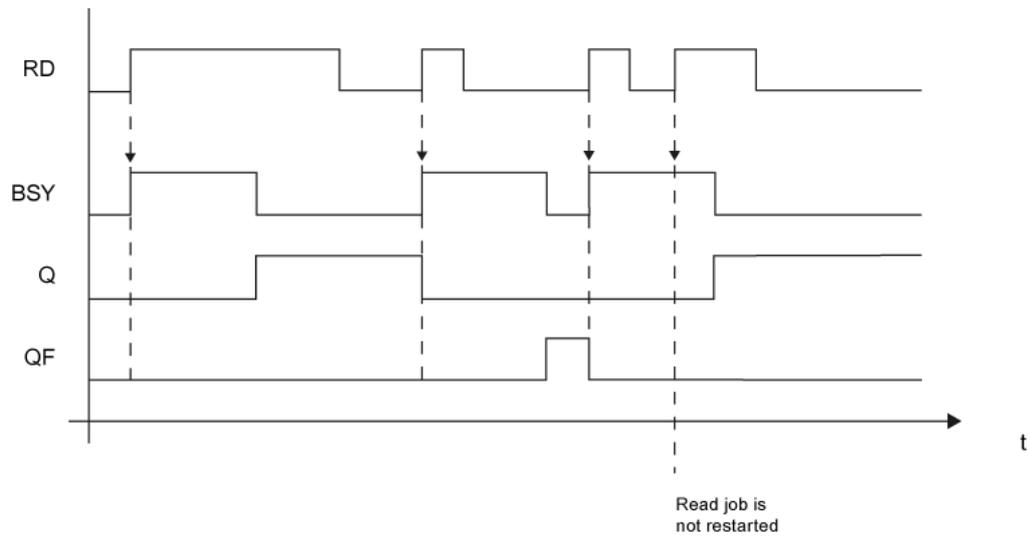
在输入RD的正沿上启动异步读任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在读任务执行时，忽略输入RD的其他正沿。

输出Q = 1表示参数已经成功读取，且参数值在输出Y处可用。在读取新值之前，Y会保留其当前的值。当在存取过程中发生错误时，通过QF = 1来表示。输出Y保留其最后的值。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	$0..2^{16}$	
IDX	参数索引	0	$0..2^{16}$	
RD	开始读任务	0	0/1	
Y	参数值	0	DINT	
BSY	执行读任务	0	0/1	
Q	成功完成读任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

据

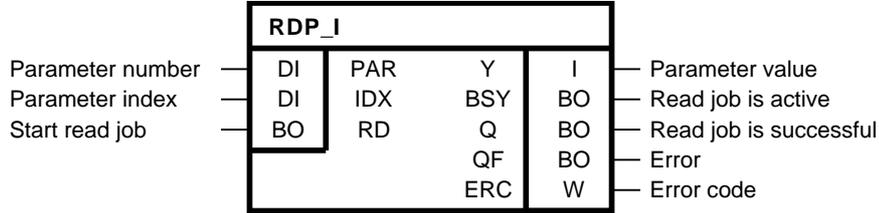
可在线加载	是
特别注意点	-

5.8 RDP_I 读取驱动参数 (INTEGER型)

SIMOTION

SINAMICS

符号



简要说明

该功能块在本地驱动对象上启动INTEGER型驱动参数的异步读取。

操作模式

读出的参数编号和参数索引分别在输入PAR和IDX处体现。当参数未索引时，必须设置IDX = 0。参数总是在驱动对象上读取，而功能块图在该驱动对象上计算。不可以在多个驱动对象上存取参数。

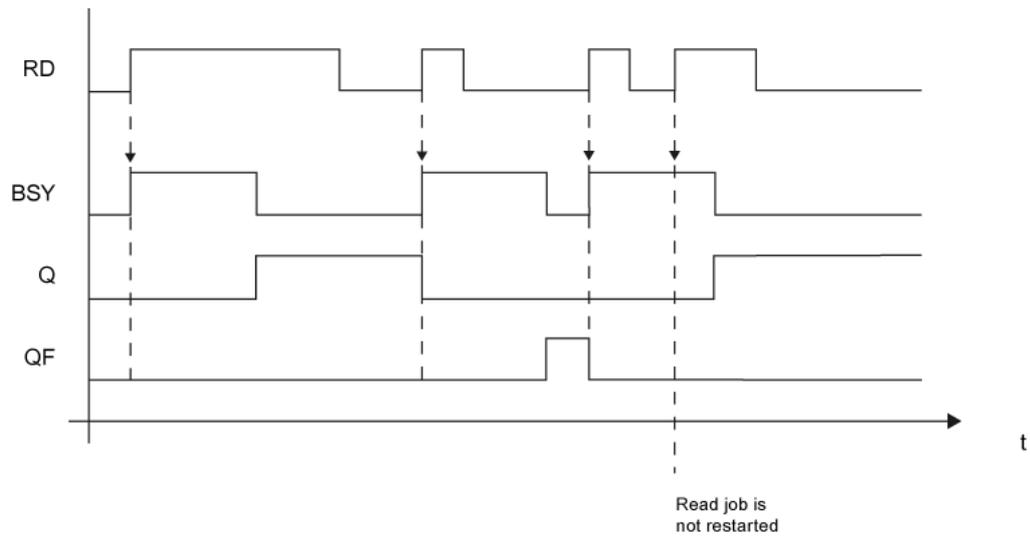
在输入RD的正沿上启动异步读任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在读任务执行时，忽略输入RD的其他正沿。

输出Q = 1表示参数已经成功读取，且参数值在输出Y处可用。在读取新值之前，Y会保留其当前的值。当在存取过程中发生错误时，通过QF = 1来表示。输出Y保留其最后的值。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	$0..2^{16}$	
IDX	参数索引	0	$0..2^{16}$	
RD	开始读任务	0	0/1	
Y	参数值	0	INT	
BSY	执行读任务	0	0/1	
Q	成功完成读任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

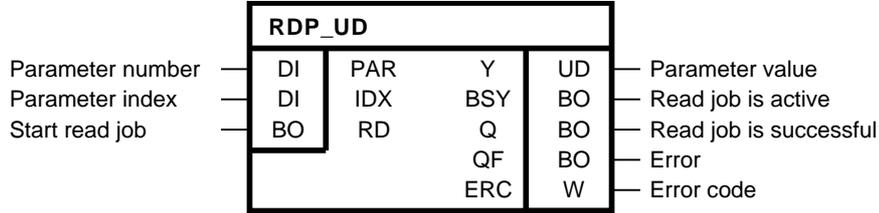
据

可在线加载	是
特别注意点	-

5.9 RDP_UD 读取驱动参数 (UNSIGNED DOUBLE INTEGER型)

SIMOTION SINAMICS

符号



简要说明

RDP_UD (读参数) 在本地驱动对象上启动UNSIGNED DOUBLE INTEGER型驱动参数的异步读取。

操作模式

读出的参数编号和参数索引分别在输入PAR和IDX处体现。当参数未索引时，必须设置IDX = 0。参数总是在驱动对象上读取，而功能块图在该驱动对象上计算。不可以在多个驱动对象上存取参数。

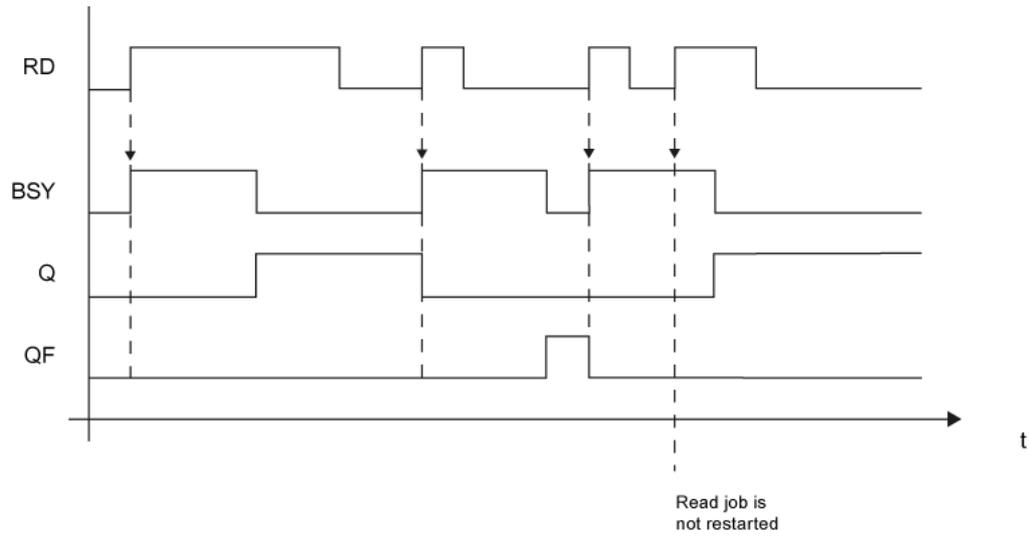
在输入RD的正沿上启动异步读任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在读任务执行时，忽略输入RD的其他正沿。

输出Q = 1表示参数已经成功读取，且参数值在输出Y处可用。在读取新值之前，Y会保留其当前的值。当在存取过程中发生错误时，通过QF = 1来表示。输出Y保留其最后的值。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	$0..2^{16}$	
IDX	参数索引	0	$0..2^{16}$	
RD	开始读任务	0	0/1	
Y	参数值	0	UDINT	
BSY	执行读任务	0	0/1	
Q	成功完成读任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

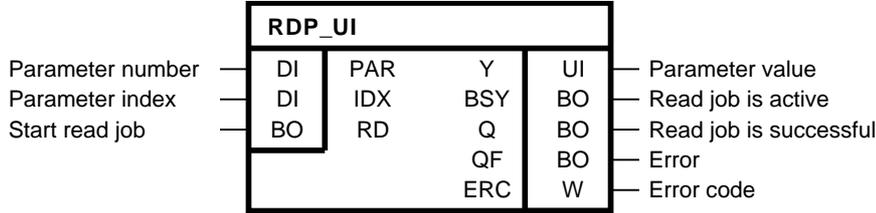
据

可在线加载	是
特别注意点	-

5.10 RDP_UI 读取驱动参数 (UNSIGNED INTEGER型)

SIMOTION SINAMICS

符号



简要说明

RDP_UI (读参数) 在本地驱动对象上启动UNSIGNED INTEGER型驱动参数的异步读取。

操作模式

读出的参数编号和参数索引分别在输入PAR和IDX处体现。当参数未索引时，必须设置IDX = 0。参数总是在驱动对象上读取，而功能块图在该驱动对象上计算。不可以在多个驱动对象上存取参数。

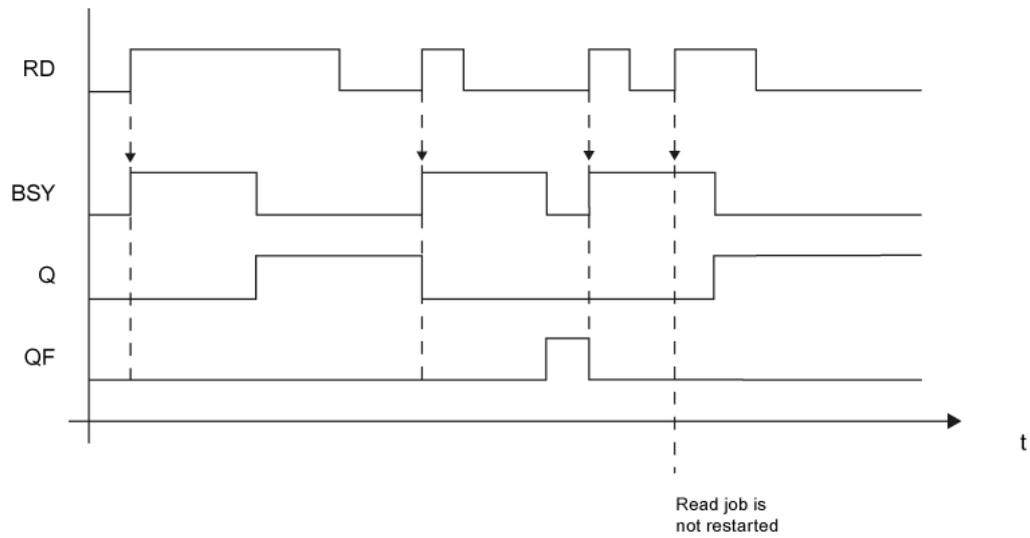
在输入RD的正沿上启动异步读任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在读任务执行时，忽略输入RD的其他正沿。

输出Q = 1表示参数已经成功读取，且参数值在输出Y处可用。在读取新值之前，Y会保留其当前的值。当在存取过程中发生错误时，通过QF = 1来表示。输出Y保留其最后的值。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	$0..2^{16}$	
IDX	参数索引	0	$0..2^{16}$	
RD	开始读任务	0	0/1	
Y	参数值	0	UINT	
BSY	执行读任务	0	0/1	
Q	成功完成读任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

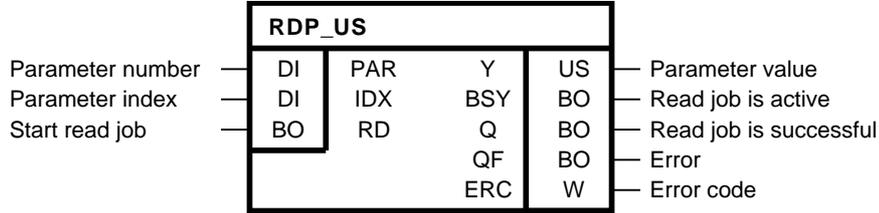
据

可在线加载	是
特别注意点	-

5.11 RDP_US 读取驱动参数 (UNSIGNED SHORT INTEGER型)

SIMOTION SINAMICS

符号



简要说明

RDP_US (读参数) 在本地驱动对象上启动UNSIGNED SHORT INTEGER型驱动参数的异步读取。

操作模式

读出的参数编号和参数索引分别在输入PAR和IDX处体现。当参数未索引时，必须设置IDX = 0。参数总是在驱动对象上读取，而功能块图在该驱动对象上计算。不可以在多个驱动对象上存取参数。

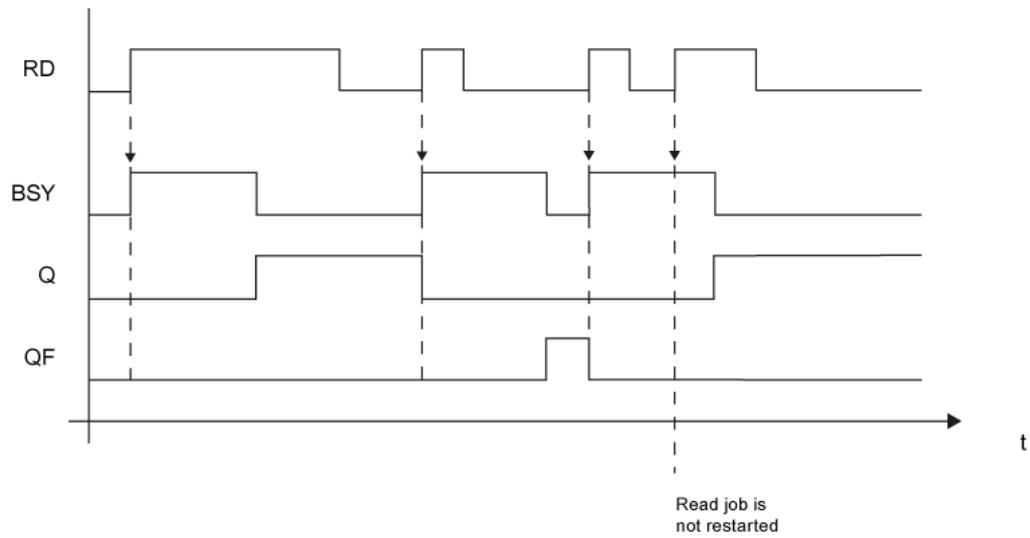
在输入RD的正沿上启动异步读任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在读任务执行时，忽略输入RD的其他正沿。

输出Q = 1表示参数已经成功读取，且参数值在输出Y处可用。在读取新值之前，Y会保留其当前的值。当在存取过程中发生错误时，通过QF = 1来表示。输出Y保留其最后的值。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	$0..2^{16}$	
IDX	参数索引	0	$0..2^{16}$	
RD	开始读任务	0	0/1	
Y	参数值	0	USINT	
BSY	执行读任务	0	0/1	
Q	成功完成读任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

据

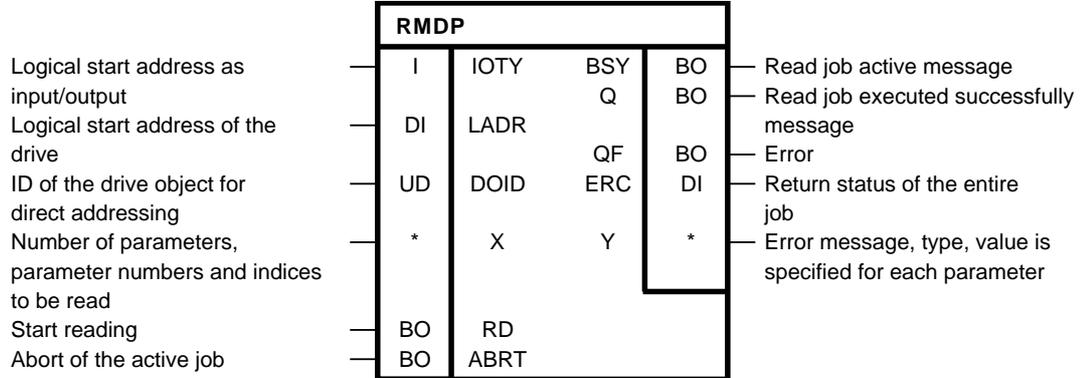
可在线加载	是
特别注意点	-

5.12 RMDP 从控制器中读取驱动参数

SIMOTION

SINAMICS

符号



简要说明

RMDP功能块最大允许从DCC SIMOTION程序中读取39个SINAMICS参数。

只支持SINAMICS驱动。

对于错误诊断，可以进行错误码ERC评估。ERC会根据PROVIDrive DPV1对应上参数存取的错误码。

在本文档的附录A.2中可以找到可能的错误码，还可以在SIMOTION通讯系统手册，PROVIDrive小节，非循环通讯（基础模式参数存取）→错误评估子小节，基础模式参数存取响应中的错误码表中找到。

RMDP功能块可用于SIMOTION V4.2及以上版本。

操作模式

首先，输入用于寻址驱动的块输入并且选择待读参数。通过输入RD的正沿启动异步读任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率和通讯负荷，且会随任务而改变。在读任务执行时，忽略输入RD的其他正沿。

没有在DCC任务中进行真正的参数读写。功能块背景仅控制通讯命令。读写任务的结果必须在随后的任务周期内在块输出处获得。

通过全局变量或自定义块类型进行评估。

输出Q = 1表示参数已成功读取，其值可以在输出Y处输出。Y保留其原值，直到成功完成一个新任务。

当在存取过程中发生错误时，通过QF = 1来表示。输出Y保留其最后的值。

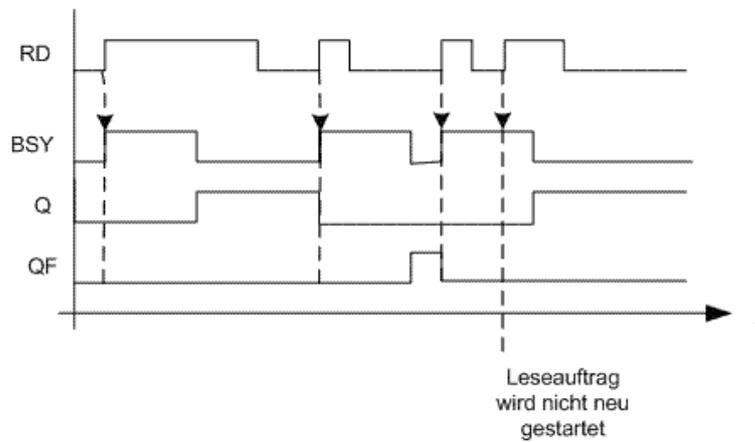
在错误诊断时，可以评估错误码ERC。

当整体任务成功完成（输出Q=1），而单个任务没有成功完成（ERC不等于0）时，显示从驱动读取的值。读取ERC = 0的其他参数没有发生错误。

可以针对参数特定的返回值PRES评估单个读任务的错误状态。

通过ABRT输入的正沿可以中止一个激活的任务。ABRT信号必须在至少一个周期内取值为1。

时序图



数量框架

该功能块最多支持读取39个SINAMICS参数。

说明

该功能块类型可以对DCC图中的每个驱动对象进行多次背景化。

最多可以并行执行16个参数任务（项

参数任务不仅包括DCB功能块RMDP和WMDP，还包括ST功能

`_readDriveParameter`、`_readDrive(Multi)Parameter-Description`、`_writeDrive(Multi)Parameter`和`_readDriveFaults`）。

如果在执行参数任务的同时，系统功能`_readRecord`和

`_writeRecord`（包括参数任务）也在并行使用，请确保它们没有被一个DP站

同时执行。原因：使用了同样的通讯服务。

当所有的通讯通道被占用时，功能块会等待，直至有通讯通道可用。BSY激活。RD信号不需要再次设置。

每个配置的DP站始终只允许处理一个参数任务。当一个新任务发送给同一个DP站时，功能块会内部等待，直至通讯可用：BSY激活

当参数任务不需要进一步执行时，可以通过ABRT输入来中止。

以下数据集可用于读出参数：

对于（外部或内部集成）PROFIBUS，不管是有效（ $0 \leq doId \leq 254$ ）还是无效‘doId’（doId=255）调用该函数，总是读出数据集47。

对于PROFINET，两个数据集可用：

基础模式参数存取 - 本地（数据集0xB02E）

当函数没有指定或只指定一个无效‘doId’（doId=255）时，此数据集（DS）用于SIMOTION。然后通过参数存取点（PAP）对相应的DO进行存取。PAP地址可直接指定，也可通过周期数据的逻辑地址指定（如DO轴的256）。接着SIMOTION根据此地址确定相关PAP，存取正确的地址。

PAP必须位于1号子槽位（HW Config中的配置）。

-基础模式参数存取 - 全局（数据集0xB02F）

当输入有效doID（ $0 \leq doId \leq 254$ ）时，使用该数据集。由于仅通过doID进行赋值，因此任何有效PAP或地址都可以指定。

块输入描述

‘IOTY’：驱动逻辑起始地址的输入/输出赋值。

当为198时：INPUT，驱动的逻辑地址位于输入范围内。

当为199时：OUTPUT，驱动的逻辑地址位于输出范围内。

诊断地址总是为INPUT型。

‘LADR’：驱动逻辑起始地址的指定。当同时使用可选参数DOID时，可以指定该站的任意地址（诊断地址优先）。

对于PROFINET，通过驱动对象的参数存取点（PAP）进行参数存取。

作为驱动逻辑起始地址的备选，推荐使用相关PAP的诊断地址。

‘DOID’：用于驱动对象的直接寻址。

在如下条件下，DOID无法指定或只能指定无效值（ >254 ）：

-DP从I/O设备不支持通过DOID进行存取（P978不执行）。

-数据集0xB02F不支持（仅对于PROFINET）。

-通过DO的PAP进行存取（仅对于PROFINET）。X：待读参数在输入X下指定。

‘X.NUMP’：待读参数的数量。

‘X.PAR[].NUM’：指定参数编号，根据该编号读取值。

‘X.PAR[].IDX’：索引值的参数索引，0表示索引0。对于非索引值，必须指定参数索引0。

‘RD’：启动读任务。

‘ABRT’：中止运行的任务。

块输出描述

‘Q’：无错完成的任务。

‘QF’含错完成的任务。

‘ERC’：对应_readDriveMultiParameter函数返回值functionResult的值。

‘Y’：参数值描述。读出每个参数的错误码、数据类型和取值。关于返回值parameterResult的更多信息，参见SIMOTION列表手册系统功能/变量设备 → 系统功能 - 设备1 → _readDriveMultiParameterDescription。

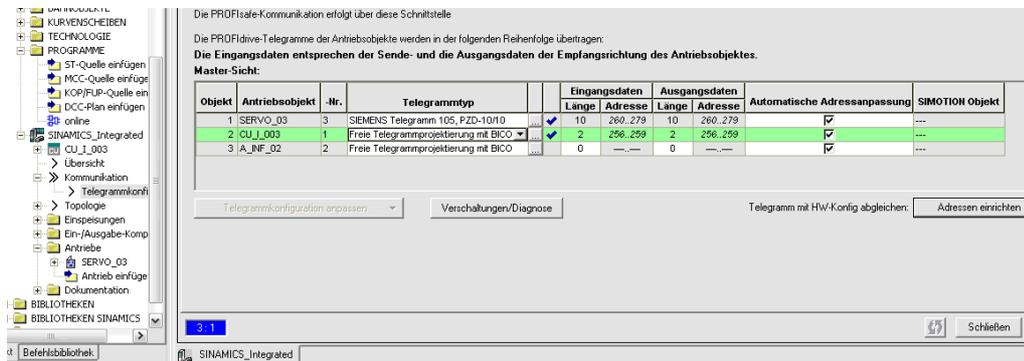
‘Y[].PRES’：对应特定参数的返回值。编码对应_readDriveMultiParameter ST函数的返回参数parameterResult。关于返回参数parameterResult的更多信息，参见SIMOTION列表手册系统功能/变量设备 → 系统功能 - 设备1 → _readDriveMultiParameterDescription。

‘Y[].DTYP’：返回参数的数据类型（关于编码情况，参见PROFIdrive分布图）。

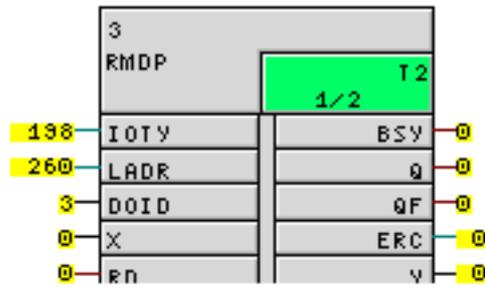
‘Y[].VAL’：从驱动中读取的参数值；数据类型即为返回的数据类型。必须为不同的数据类型调用转换功能块。当存取REAL参数时，通过转换功能块DW_R进行转换。

参数设置示例

为了能够读取驱动对象（如：SERVO_03）的某些参数， 执行以下操作：
首先， 设置正确的消息框架配置。



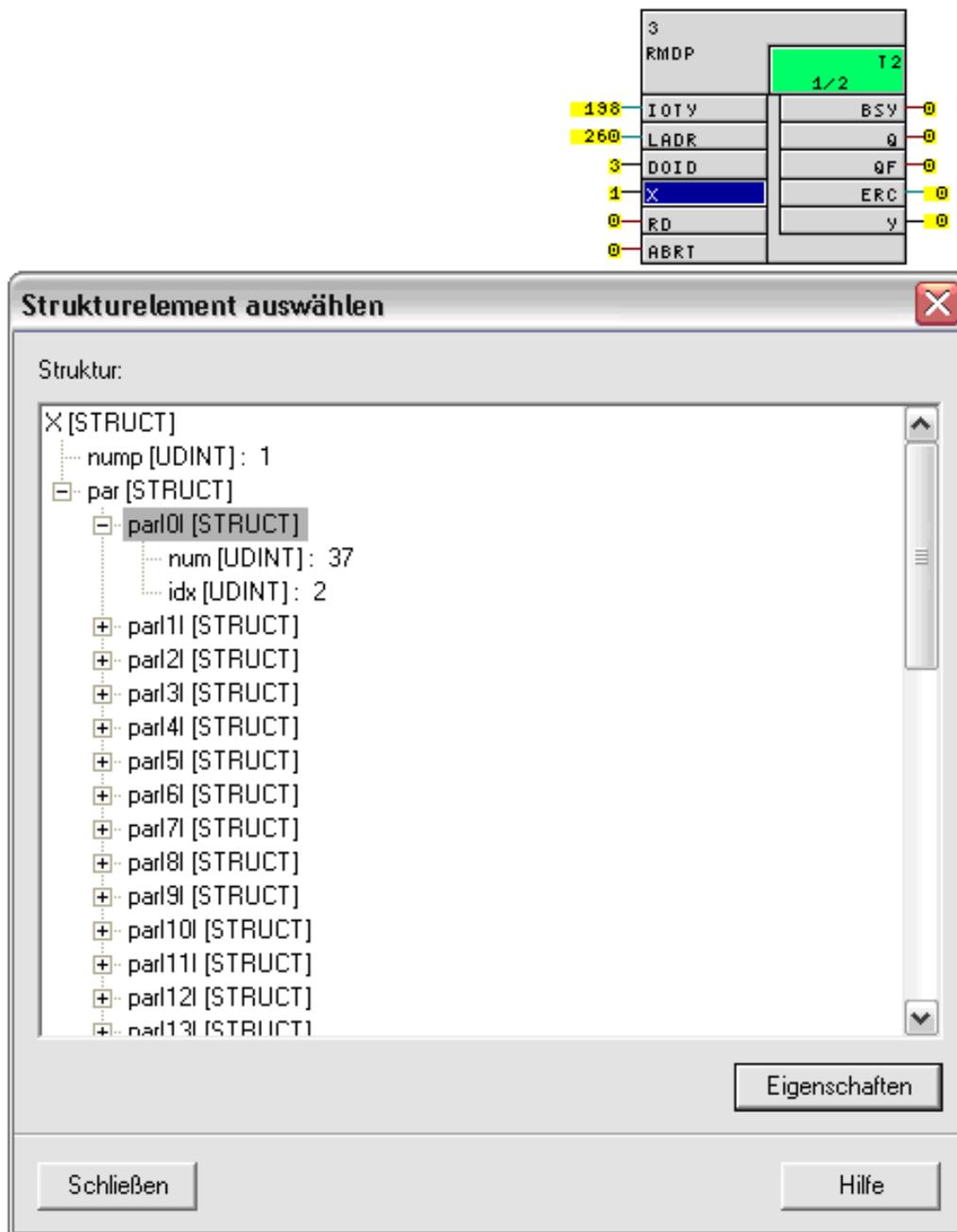
接着在RMDP功能块中设置所需的DO地址。 为此， 需要设置块输入LADR为消息框架中设置的地址（260）， 并设置块输入DOID为消息框架中设置的编号（3）。



在块输入X处选择待读参数， 如： 专家列表中的“r37(2) power unit temperatures, rectifier maximum value”。



为此， 需要双击块输入X， 选择第一个结构件， 再分别在num和idx处输入参数编号（37）和索引（2）。



然后设置块输入RD为1，以此来启动读任务。

is	B Plan	Baustein	Anschluss	Wert	Einheit	Kom
	<input checked="" type="checkbox"/> online	3	Y.yIOI.erc	0		
	<input checked="" type="checkbox"/> online	3	Y.yIOI.dtyp	8		
	<input checked="" type="checkbox"/> online	3	Y.yIOI.val	16#C3480000		

多个块连接

块 连接	说明	预分配	取值范围	数
IOTY	作为输入/输出的逻辑起始地址	0	0: 无效 198: 输入地址 199: 输出地址	
LADR	驱动的逻辑起始地址	-1	DINT	
DOID	用于直接寻址的驱动对象ID	255	0 .. 254, 255: 无效	
X	参数数量、待读取的参数编号和索引			
X.NUMP	待读参数的数量	1	1..39	
X.PAR	参数描述			
X.PAR[].NUM	参数编号	1	1..65535	
X.PAR[].IDX	参数索引	0	0..65535	
RD	开始读任务	0	0/1	
ABRT	中止运行的任务	0	0/1	
BSY	读任务运行消息	0	0/1	
Q	成功执行读任务的消息	0	0/1	
QF	错误	0	0/1	
ERC	整个任务的返回状态	16#0000	DWORD	
Y	为每个参数指定错误消息、类型和值			
Y[].ERC	参数的返回值	0	DINT	
Y[].DTYP	读参数的数据类型	0	USINT	
Y[].VAL	从驱动中读取的参数值	0	DWORD	

5.12 RMDP 从控制器中读取驱动参数

据

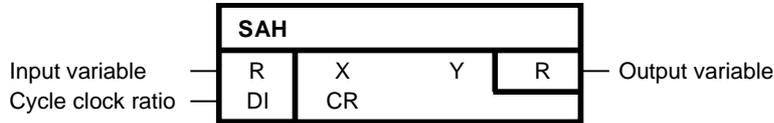
SIMOTION	V4.2及以上版本
SINAMICS	-
可在线加载	否
处理环境	循环, 等距
特别注意点	-

5.13 SAH 采样与保持 (REAL型)

□ SIMOTION

☑ SINAMICS

符号



简要说明

采样与保持功能块用于在不同的采样时间点，在功能块之间进行等距值传递 (REAL型)。

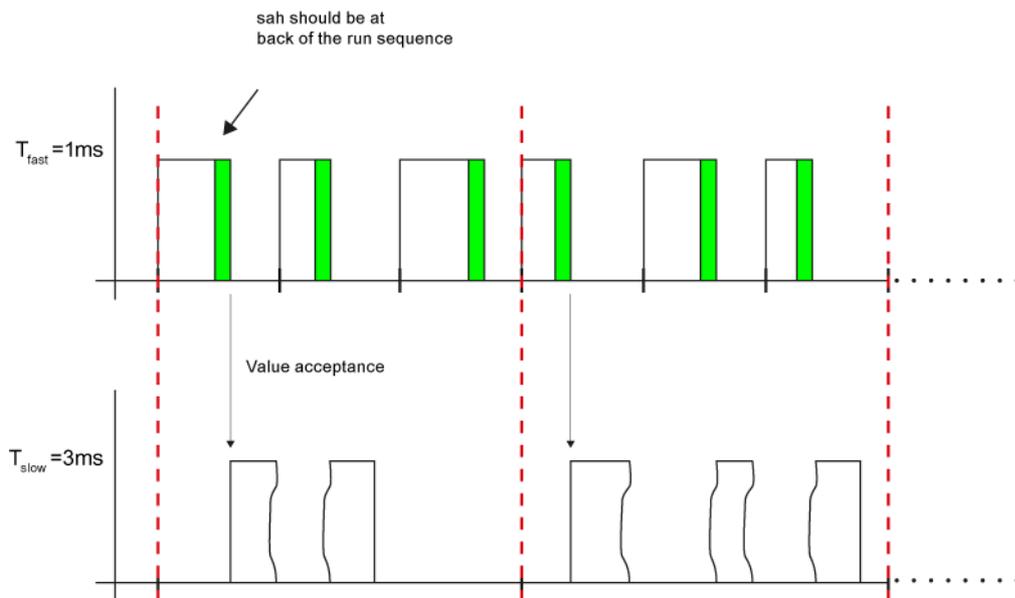
操作模式

在每个CR周期内，输入变量X的值传递给输出变量Y。数值传递周期与执行系统的周期控制点进行同步。周期控制点定义了周期时钟，在周期时钟内重启执行系统的所有采样时间。

在相对于周期控制点的每个CR周期内进行数值传递。CR的绝对值总是为周期时钟比生成。在CR = 0这一特殊情况下时，该功能块表现为CR = 1时的特性。功能块必须总是在较快的采样时间点进行配置。当传递较慢扫描时间点的数值时，应该在执行顺序的最开始进行传递。当数值由较快采样时间替换为较慢采样时间时，应该在执行顺序的最后计算时钟。

下图显示了由1毫秒级到3毫秒级的数值传递。以下随时间的变化图针对执行组的计算情况。

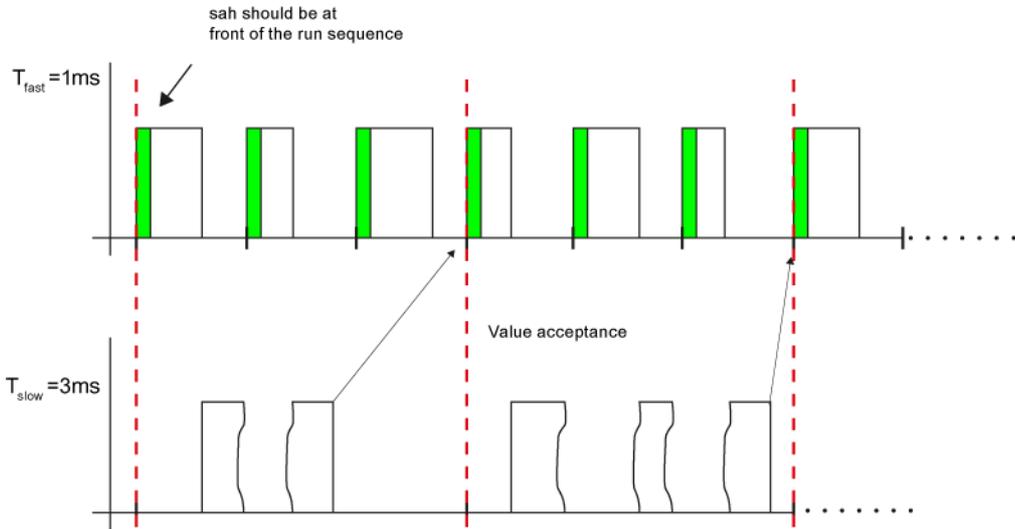
$$CR = \frac{3ms}{1ms} = 3$$



5.13 SAH 采样与保持 (REAL型)

下图显示了由3毫秒级到1毫秒级的数值传递。 以下随时间的变化图针对执行组的计算情况。

$$CR = \frac{3\text{ms}}{1\text{ms}} = 3$$



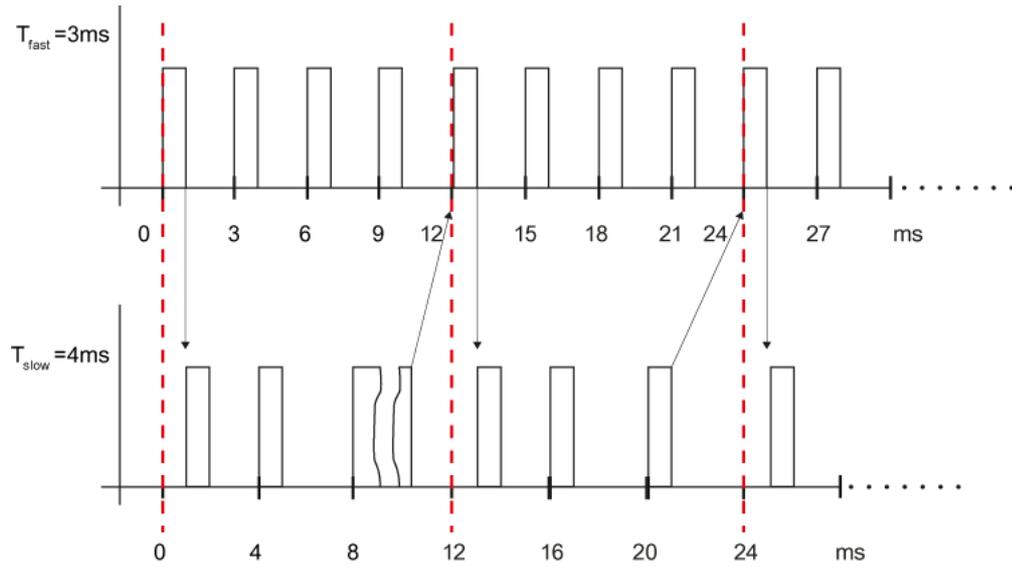
当较慢扫描时间不是较快扫描时间的倍数时，如果二者的扫描过程在CR周期后同时进行重启，那么该数值只能进行一致性传递。 该值对应于两个扫描时间的最小公倍数。 那么CR则根据以下公式进行计算：

$$CR = \frac{g(T_{fast}, T_{slow})}{T_{fast}}$$

$g(T_{fast}, T_{slow})$: 最小公倍数

下图显示了 $T_{fast} = 3$ 毫秒和 $T_{slow} = 4$ 毫秒间的数值传递。 数值传递在双向进行。

$$CR = \frac{g(3\text{ms}, 4\text{ms})}{3\text{ms}} = \frac{12\text{ms}}{3\text{ms}} = 4$$



要使数值在最佳速度进行传递，建议一直保持较慢扫描时间为较快扫描时间的倍数。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
CR	周期时钟比	1	$0 - (2^{31}-1)$	
Y	输出变量	0.0	REAL	

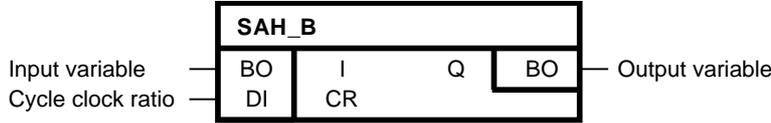
据

可在线加载	是
特别注意点	-

5.14 SAH_B 采样与保持 (BOOL型)

SIMOTION SINAMICS

符号



简要说明

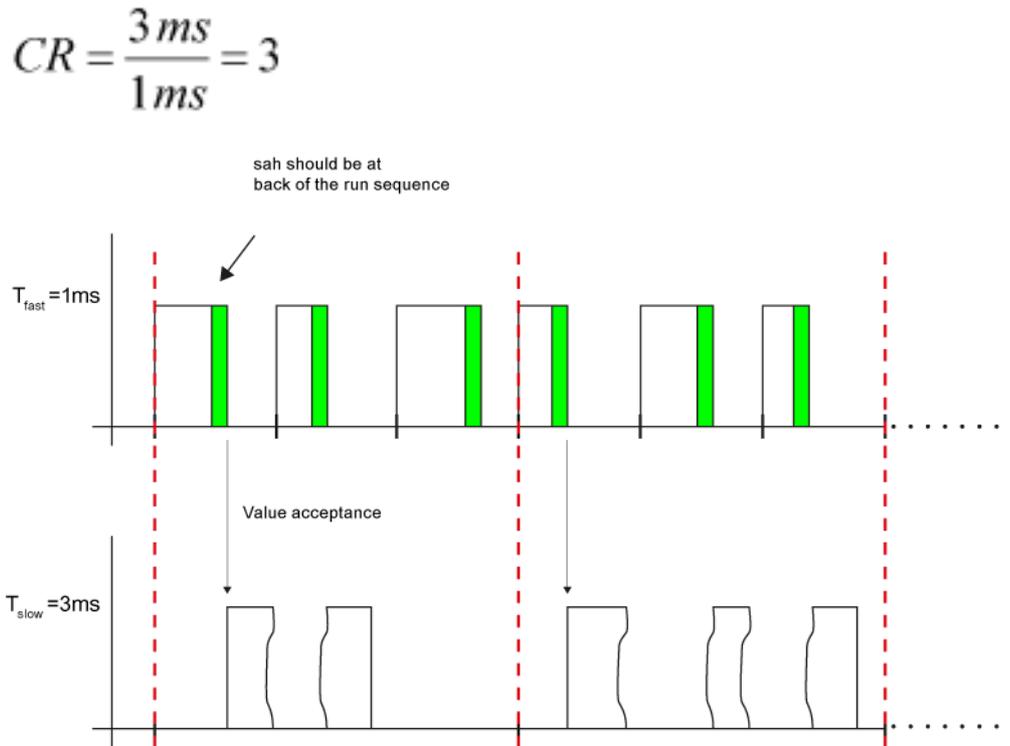
采样与保持功能块用于在不同的采样时间点，在功能块之间进行等距值传递 (BOOL型)。

操作模式

在每个CR周期内，输入变量I的值传递给输出变量Q。数值传递周期与执行系统的周期控制点进行同步。周期控制点定义了周期时钟，在周期时钟内重启执行系统的所有采样时间。

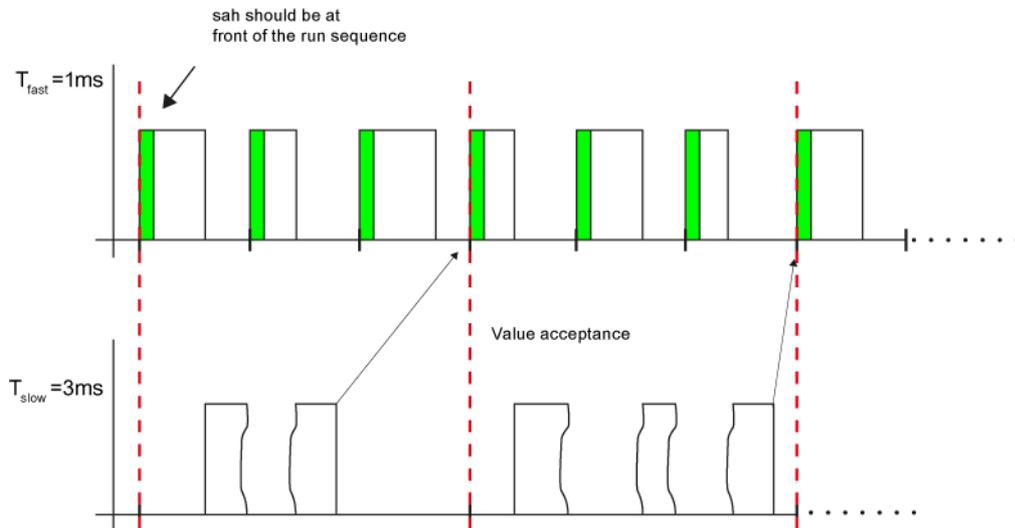
在相对于周期控制点的每个CR周期内进行数值传递。CR的绝对值总是为周期时钟比生成。在CR = 0这一特殊情况下时，该功能块表现为CR = 1时的特性。功能块必须总是在较快的采样时间点进行配置。当传递较慢扫描时间点的数值时，应该在执行顺序的最开始进行传递。当数值由较快采样时间替换为较慢采样时间时，应该在执行顺序的最后计算时钟。

下图显示了由1毫秒级到3毫秒级的数值传递。以下随时间的变化图针对执行组的计算情况。



下图显示了由3毫秒级到1毫秒级的数值传递。 以下随时间的变化图针对执行组的计算情况。

$$CR = \frac{3ms}{1ms} = 3$$



当较慢扫描时间不是较快扫描时间的倍数时，如果二者的扫描过程在CR周期后同时进行重启，那么该数值只能进行一致性传递。 该值对应于两个扫描时间的最小公倍数。 那么CR则根据以下公式进行计算：

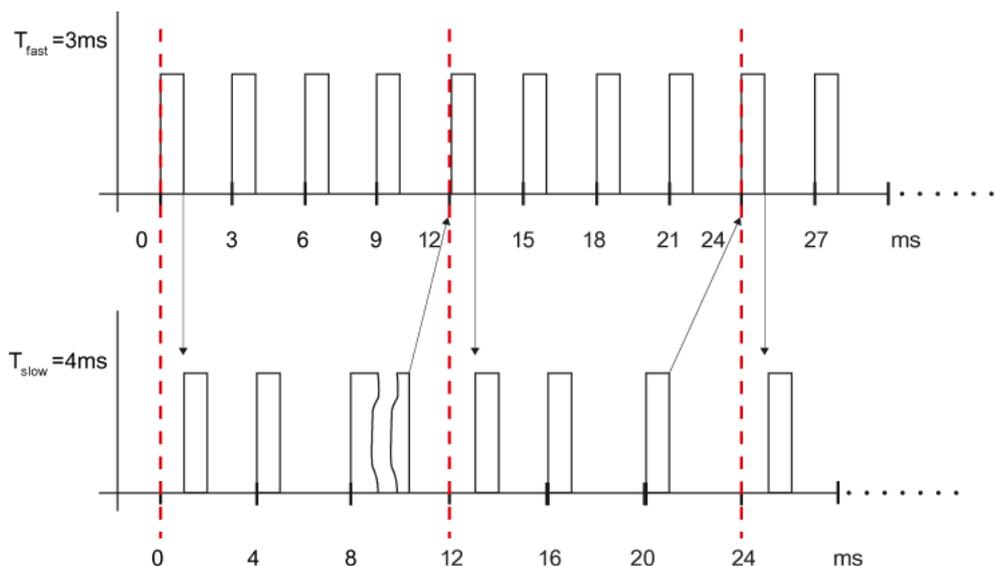
$$CR = \frac{g(T_{fast}, T_{slow})}{T_{fast}}$$

$g(T_{fast}, T_{slow})$ ：最小公倍数

下图显示了 $T_{fast} = 3$ 毫秒和 $T_{slow} = 4$ 毫秒间的数值传递。 数值传递在双向进行。

$$CR = \frac{g(3ms, 4ms)}{3ms} = \frac{12ms}{3ms} = 4$$

5.14 SAH_B 采样与保持 (BOOL型)



要使数值在最佳速度进行传递，建议一直保持较慢扫描时间为较快扫描时间的倍数。

多个块连接

块 连接	说明	预分配	取值范围	数
I	输入变量	0	0/1	
CR	周期时钟比	1	$0 - (2^{31}-1)$	
Q	输出变量	0	0/1	

据

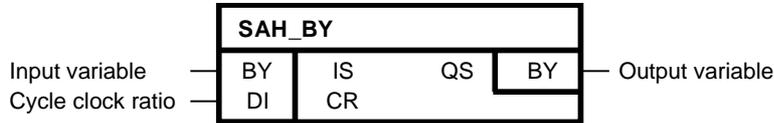
可在线加载	是
特别注意点	-

5.15 SAH_BY 采样与保持 (BYTE型)

□ SIMOTION

☑ SINAMICS

符号



简要说明

采样与保持功能块用于在不同的采样时间点，在功能块之间进行等距值传递 (BYTE型)。

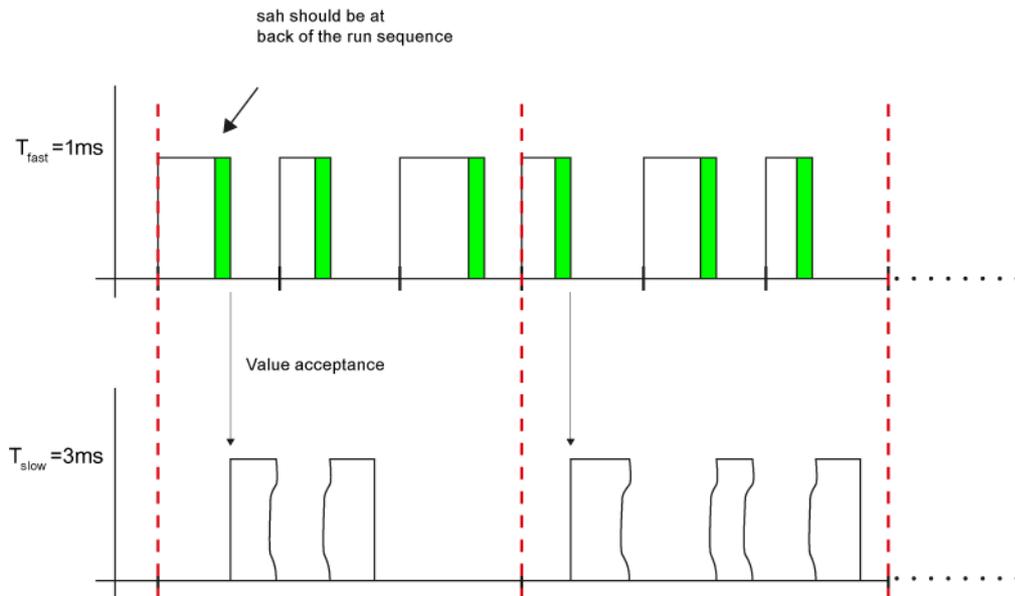
操作模式

在每个CR周期内，输入变量IS的值传递给输出变量QS。数值传递周期与执行系统的周期控制点进行同步。周期控制点定义了周期时钟，在周期时钟内重启执行系统的所有采样时间。

在相对于周期控制点的每个CR周期内进行数值传递。CR的绝对值总是为周期时钟比生成。在CR = 0这一特殊情况下时，该功能块表现为CR = 1时的特性。功能块必须总是在较快的采样时间点进行配置。当传递较慢扫描时间点的数值时，应该在执行顺序的最开始进行传递。当数值由较快采样时间替换为较慢采样时间时，应该在执行顺序的最后计算时钟。

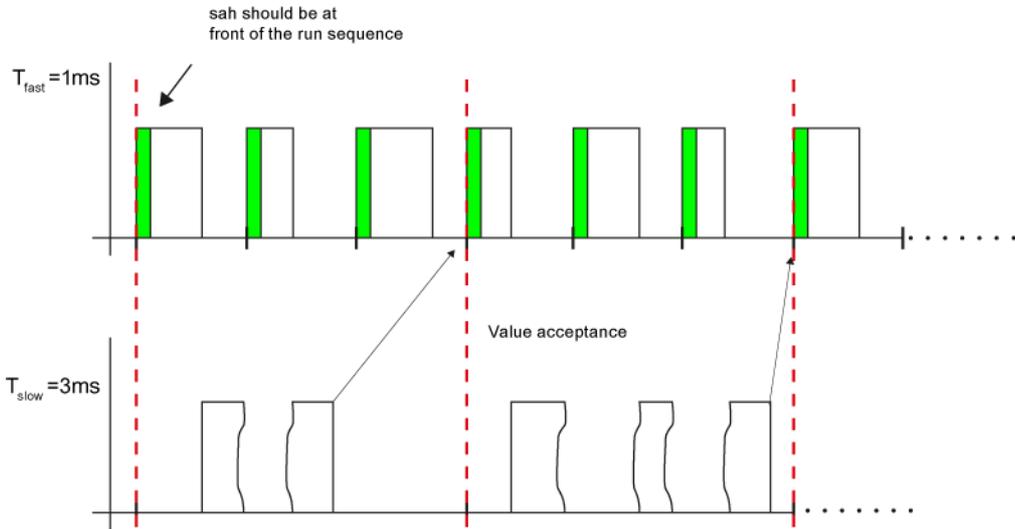
下图显示了由1毫秒级到3毫秒级的数值传递。以下随时间的变化图针对执行组的计算情况。

$$CR = \frac{3ms}{1ms} = 3$$



下图显示了由3毫秒级到1毫秒级的数值传递。 以下随时间的变化图针对执行组的计算情况。

$$CR = \frac{3\text{ms}}{1\text{ms}} = 3$$



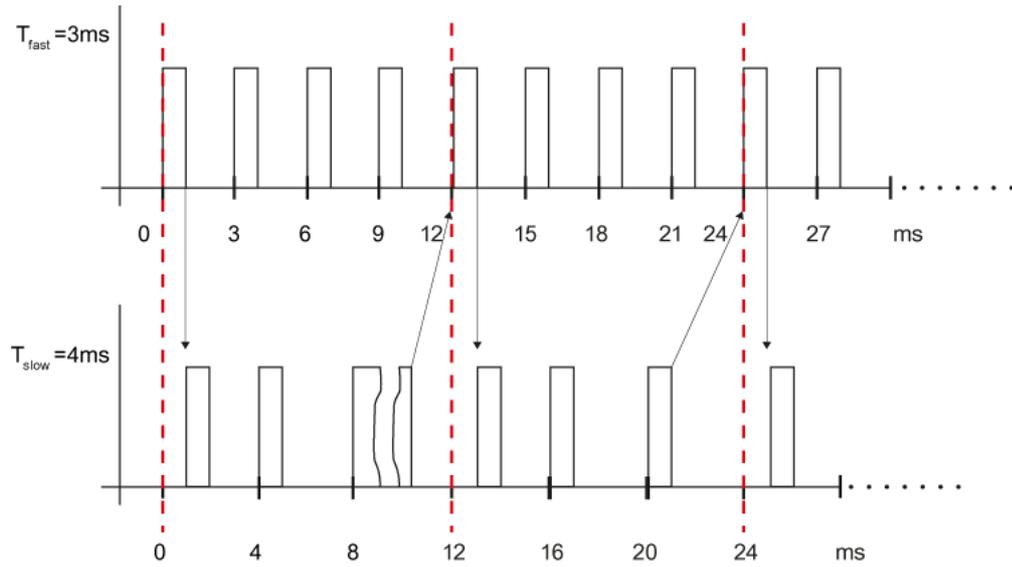
当较慢扫描时间不是较快扫描时间的倍数时，如果二者的扫描过程在CR周期后同时进行重启，那么该数值只能进行一致性传递。 该值对应于两个扫描时间的最小公倍数。 那么CR则根据以下公式进行计算：

$$CR = \frac{g(T_{fast}, T_{slow})}{T_{fast}}$$

$g(T_{fast}, T_{slow})$: 最小公倍数

下图显示了 $T_{fast} = 3$ 毫秒和 $T_{slow} = 4$ 毫秒间的数值传递。 数值传递在双向进行。

$$CR = \frac{g(3\text{ms}, 4\text{ms})}{3\text{ms}} = \frac{12\text{ms}}{3\text{ms}} = 4$$



要使数值在最佳速度进行传递，建议一直保持较慢扫描时间为较快扫描时间的倍数。

多个块连接

块 连接	说明	预分配	取值范围	数
IS	输入变量	16#00	BYTE	
CR	周期时钟比	1	$0 - (2^{31}-1)$	
QS	输出变量	16#00	BYTE	

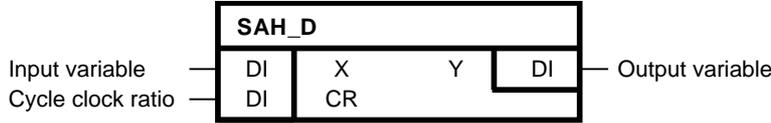
据

可在线加载	是
特别注意点	-

5.16 SAH_D 采样与保持 (DOUBLE INTEGER型)

SIMOTION SINAMICS

符号



简要说明

采样与保持功能块用于在不同的采样时间点，在功能块之间进行等距值传递 (DOUBLE INTEGER型)。

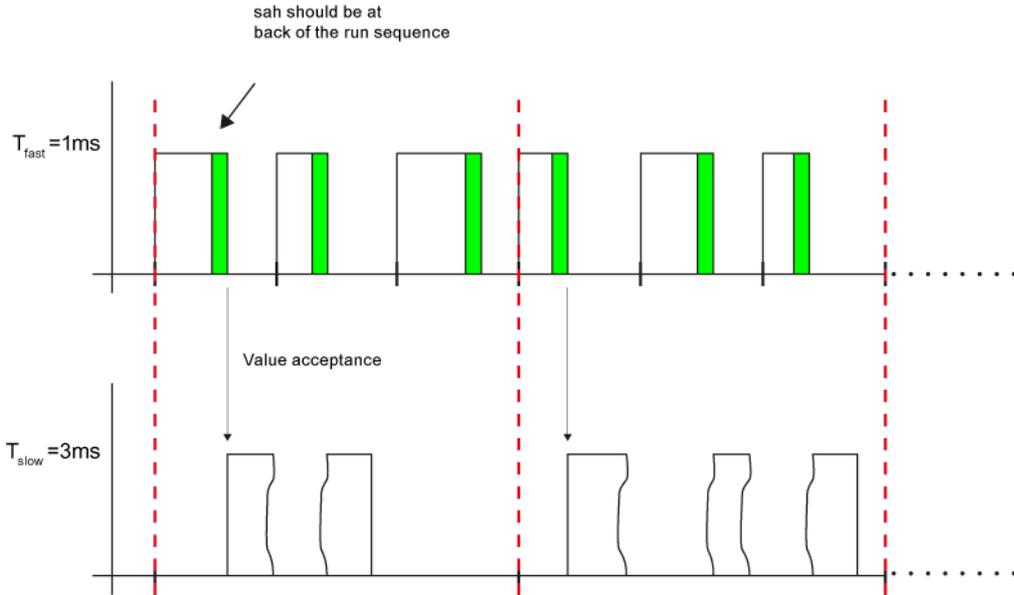
操作模式

在每个CR周期内，输入变量X的值传递给输出变量Y。数值传递周期与执行系统的周期控制点进行同步。周期控制点定义了周期时钟，在周期时钟内重启执行系统的所有采样时间。

在相对于周期控制点的每个CR周期内进行数值传递。CR的绝对值总是为周期时钟比生成。在CR = 0这一特殊情况下时，该功能块表现为CR = 1时的特性。功能块必须总是在较快的采样时间点进行配置。当传递较慢扫描时间点的数值时，应该在执行顺序的最开始进行传递。当数值由较快采样时间替换为较慢采样时间时，应该在执行顺序的最后计算时钟。

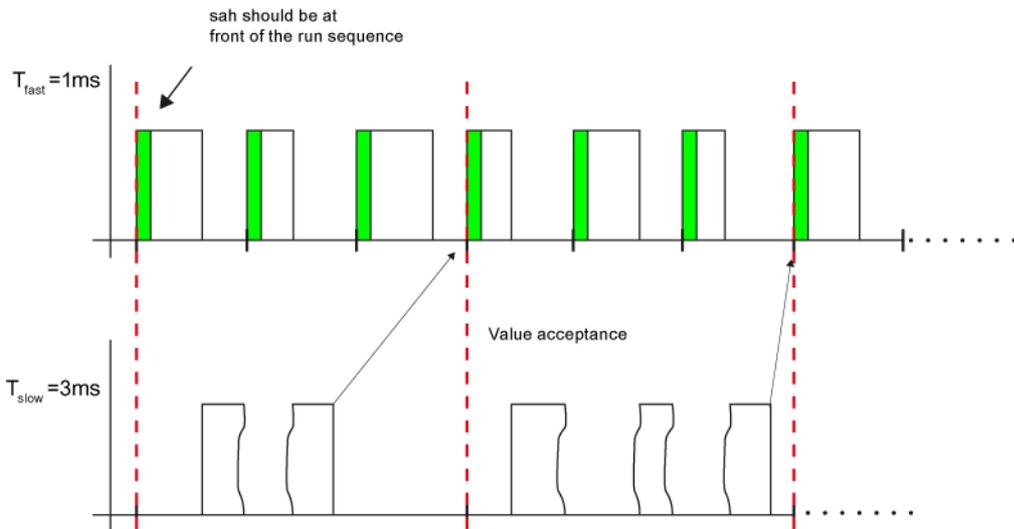
下图显示了由1毫秒级到3毫秒级的数值传递。以下随时间的变化图针对执行组的计算情况。

$$CR = \frac{3ms}{1ms} = 3$$



下图显示了由3毫秒级到1毫秒级的数值传递。 以下随时间的变化图针对执行组的计算情况。

$$CR = \frac{3ms}{1ms} = 3$$



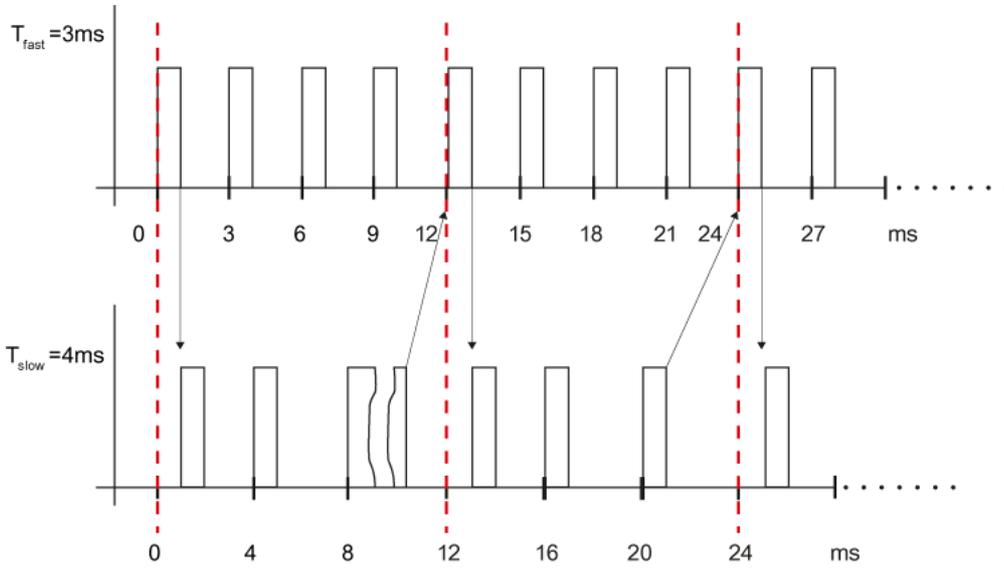
当较慢扫描时间不是较快扫描时间的倍数时，如果二者的扫描过程在CR周期后同时进行重启，那么该数值只能进行一致性传递。该值对应于两个扫描时间的最小公倍数。那么CR则根据以下公式进行计算：

$$CR = \frac{g(T_{fast}, T_{slow})}{T_{fast}}$$

$g(T_{fast}, T_{slow})$: 最小公倍数

下图显示了 $T_{fast} = 3$ 毫秒和 $T_{slow} = 4$ 毫秒间的数值传递。数值传递在双向进行。

$$CR = \frac{g(3\text{ ms}, 4\text{ ms})}{3\text{ ms}} = \frac{12\text{ ms}}{3\text{ ms}} = 4$$



要使数值在最佳速度进行传递，建议一直保持较慢扫描时间为较快扫描时间的倍数。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
CR	周期时钟比	1	0 - (2 ³¹ -1)	
Y	输出变量	0	DINT	

据

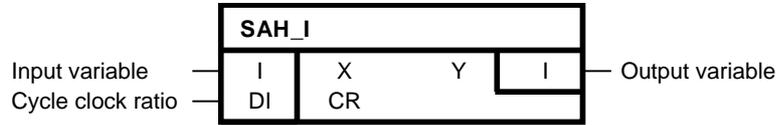
可在线加载	是
特别注意点	-

5.17 SAH_I 采样与保持 (INTEGER型)

□ SIMOTION

☑ SINAMICS

符号



简要说明

采样与保持功能块用于在不同的采样时间点，在功能块之间进行等距值传递 (INTEGER型)。

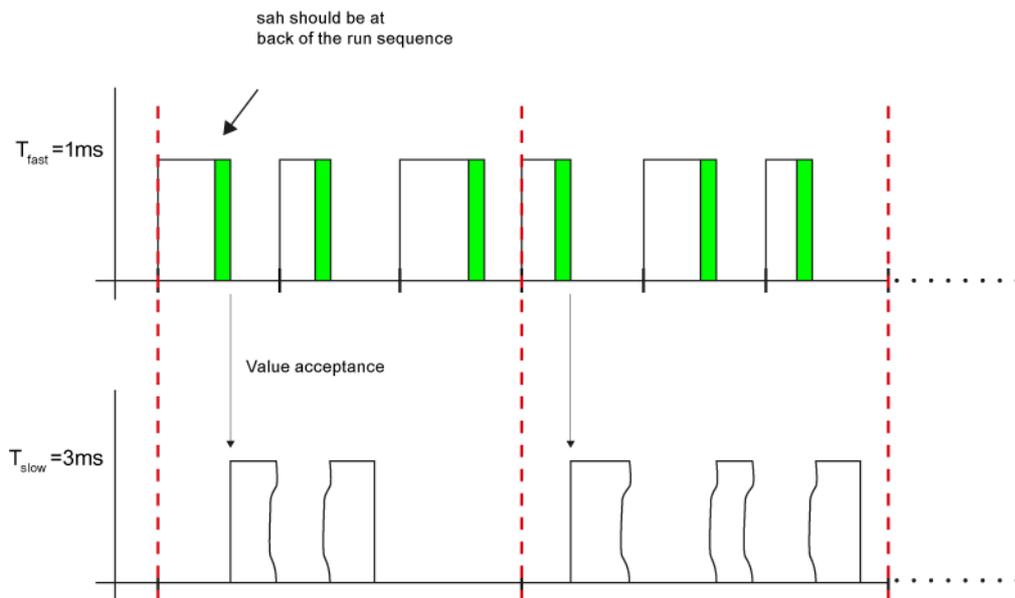
操作模式

在每个CR周期内，输入变量X的值传递给输出变量Y。数值传递周期与执行系统的周期控制点进行同步。周期控制点定义了周期时钟，在周期时钟内重启执行系统的所有采样时间。

在相对于周期控制点的每个CR周期内进行数值传递。CR的绝对值总是为周期时钟比生成。在CR = 0这一特殊情况下时，该功能块表现为CR = 1时的特性。功能块必须总是在较快的采样时间点进行配置。当传递较慢扫描时间点的数值时，应该在执行顺序的最开始进行传递。当数值由较快采样时间替换为较慢采样时间时，应该在执行顺序的最后计算时钟。

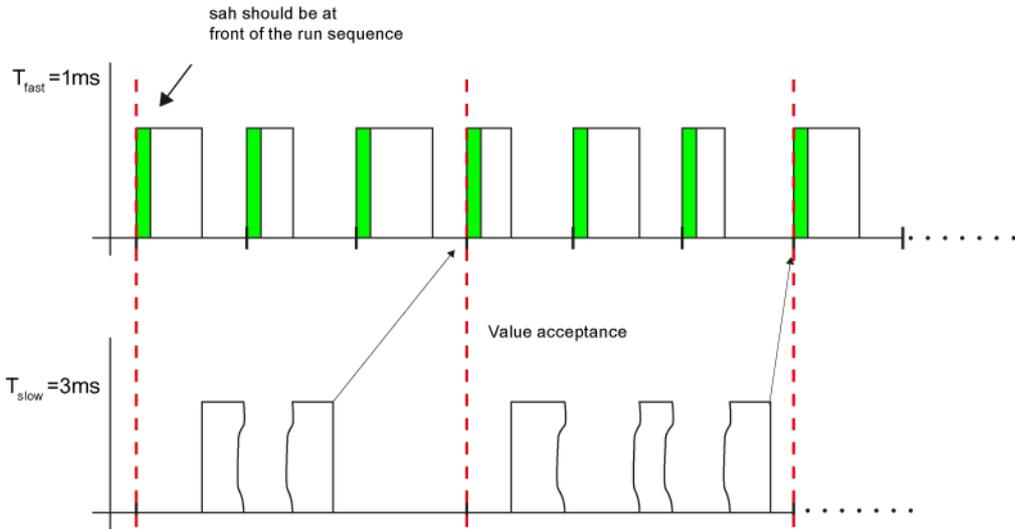
下图显示了由1毫秒级到3毫秒级的数值传递。以下随时间的变化图针对执行组的计算情况。

$$CR = \frac{3ms}{1ms} = 3$$



下图显示了由3毫秒级到1毫秒级的数值传递。 以下随时间的变化图针对执行组的计算情况。

$$CR = \frac{3\text{ms}}{1\text{ms}} = 3$$



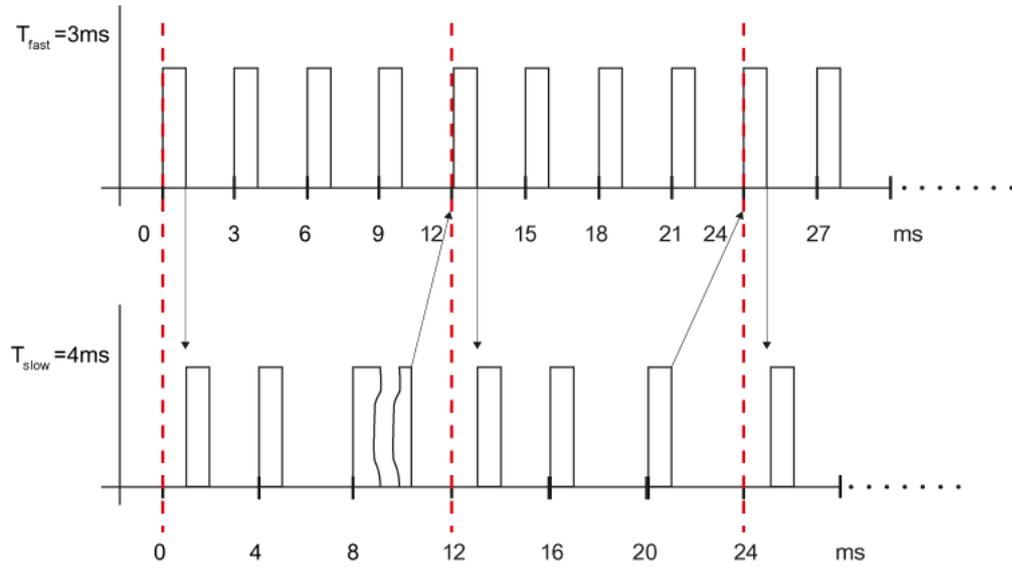
当较慢扫描时间不是较快扫描时间的倍数时，如果二者的扫描过程在CR周期后同时进行重启，那么该数值只能进行一致性传递。 该值对应于两个扫描时间的最小公倍数。 那么CR则根据以下公式进行计算：

$$CR = \frac{g(T_{fast}, T_{slow})}{T_{fast}}$$

$g(T_{fast}, T_{slow})$: 最小公倍数

下图显示了 $T_{fast} = 3$ 毫秒和 $T_{slow} = 4$ 毫秒间的数值传递。 数值传递在双向进行。

$$CR = \frac{g(3\text{ms}, 4\text{ms})}{3\text{ms}} = \frac{12\text{ms}}{3\text{ms}} = 4$$



要使数值在最佳速度进行传递，建议一直保持较慢扫描时间为较快扫描时间的倍数。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
CR	周期时钟比	1	$0 - (2^{31}-1)$	
Y	输出变量	0	INT	

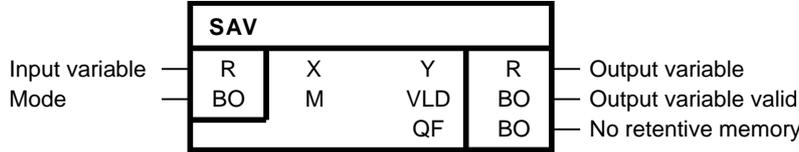
据

可在线加载	是
特别注意点	-

5.18 SAV 数值缓冲 (REAL型)

SIMOTION SINAMICS

符号



简要说明

- SAV (保存) 启用REAL型输入变量的保留存储。

操作模式

该功能块是BYTE数值的保留读/写存储器。

SAV功能块的保存数值在以下情况下不会保留:

- 用户已经清空了保留存储器。
- 配置功能块的DCC图已经删除, 且修改已经传递给目标设备。
- 功能块已经删除, 且修改已经传递给目标设备。
- 背景名称已经修改, 且相应的修改已经传递给目标设备。

该值在以下情况保留:

- 下载时, 背景名称没有修改
- 目标设备斜升, 且存储卡上无配置数据。 只有在下载后才会释放缺失SAV功能块的存储。 这样的话, 当固件更新时, 也会保留数据。
- 当已经添加或移除了另一个SAV功能块时
- 当DCBLIB更新后进行配置数据下载时
- 当已经添加或移除了另一个DO并将其下载至目标设备时
- 当已经添加或移除了另一个DCC图并将其下载至目标设备时
- 当目标设备按电源故障前相同的配置数据斜升时

只有当输出QF为0时功能块才处于激活状态。这表明目标设备上的预留存储空间可用于存储输入值。

说明

当新于版本D的硬件可用时, SAV功能块仅应用于SINAMICS INTEGRATED上的D425 (6AU1425-0AA00-0AA0)、D435 (6AU1435-0AA00-0AA1) 和D445 (6AU1445-0AA00-0AA0)。

告警!

对于模块SIMOTION D410 / CX32、D425、D435、D445、D445-1/CX32-2、D445-2和D455-2, 必须考虑到非易失数据的行为特性。关于非易失数据, 参见版本11/2010及以上的D410调试与硬件安装手册/D4x5调试与硬件安装手册/ D4x5-2调试与硬件安装手册中的调试(硬件) → 用户存储器概念 → 用户存储器属性。视硬件设备而定, 那些模块上的保留数据只保留有限的一段时间。对于SIMOTION V4.2及以上版本, SIMOTION系统功能“savePersistentMemoryData”包括SINAMICS_Integrated(如: DCC功能块SAV、SAV_BY、SAV_D、SAV_I)上属于D425、D435、D445和D445-1模块的保留数据, 该数据备份存储于CF卡上。

说明

对于SCOUT V4.2及以上版本, DCC SIMOTION的SAV功能块数值可以通过“保存和恢复保留数据”功能进行备份。

输入M处功能块模式设置为:

写模式 (M = 1)

- 输入变量X周期性写给输出Y。
- 另外, 输入变量X传递给系统, 进行保留存储。借此, 覆盖已经保存的数值。

读模式 (M = 0)

- 当前保存的数值在输出Y处输出。输入X的数值不保存。
- 输出VLD = 1表示Y有效。当功能块初始化, 系统的保留存储空间重新创建时, VLD = 0。此时, Y无效, 包含其默认值。初次写入数值 (M = 1) 时, VLD状态则变为1。

初始化

通过功能块的背景名称在SAV功能块和保留存储中的数值之间进行分配。当功能块插入图中时, DCC编辑器自动生成唯一的背景名称。背景名称由功能块的调用路径构成, 命名规则如下:

(图名称)/(子图1名称)/(子图2名称)/../(功能块名称)

例示背景名称如下:

DCC_1/CFC1/CFC2/CFC3/SAV1

图名称	DCC_1
子图1名称	CFC1
子图2名称	CFC2
子图3名称	CFC3
功能块名称	SAV1

此背景名称控制输出Y是初始化为默认值还是在INIT模式下输出最近保存的数值。目标设备会检查保留数值是否已经为功能块的背景名称而保存。如果没有保存的话, 系统会重新创建存储空间, 输出变量Y的默认值会传递给系统进行保留存储, 并设置VLD = 0。当一个保留数值已经为背景名称而保存时, 那么该值读/写给输出Y, 并输出状态VLD = 1。

如果功能块无存储空间可用, 则设置QF = 1。

5.18 SAV 数值缓冲 (REAL型)

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
M	模式	0	0/1	
Y	输出变量	0.0	REAL	
VLD	输出变量有效	0	0/1	
QF	无保留存储	0	0/1	

据

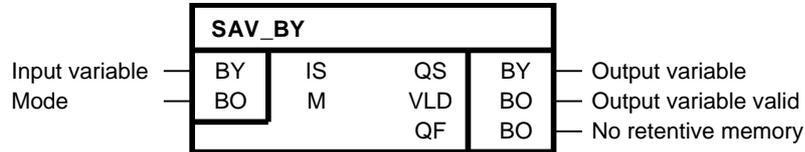
可在线插入	否
特别注意点	每个SINAMICS或SINAMICS INTEGRATED驱动单元最多支持10个功能块 (SAV、SAV_BY、SAV_I、SAV_D) 用于保留存储。保留存储最大支持40字节的用户数据。对于STARTER/SCOUT V4.2及以上版本, 只在一致性检查时检查功能块数量, 一致性检查的菜单入口为“Project -> Check consistency”或驱动单元的右键菜单中“Check consistency”。

5.19 SAV_BY 数值缓冲 (BYTE型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- SAV_BY (保存) 启用BYTE型输入变量的保留存储。

操作模式

该功能块是BYTE数值的保留读/写存储器。

SAV功能块的保存数值在以下情况下不会保留：

- 用户已经清空了保留存储器。
- 配置功能块的DCC图已经删除，且修改已经传递给目标设备。
- 功能块已经删除，且修改已经传递给目标设备。
- 背景名称已经修改，且相应的修改已经传递给目标设备。

该值在以下情况保留：

- 下载时，背景名称没有修改
- 目标设备斜升，且存储卡上无配置数据。只有在下载后才会释放缺失SAV功能块的存储。这样的话，当固件更新时，也会保留数据。
- 当已经添加或移除了另一个SAV功能块时
- 当DCBLIB更新后进行配置数据下载时
- 当已经添加或移除了另一个DO并将其下载至目标设备时
- 当已经添加或移除了另一个DCC图并将其下载至目标设备时
- 当目标设备按电源故障前相同的配置数据斜升时

只有当输出QF为0时功能块才处于激活状态。这表明目标设备上的预留存储空间可用于存储输入值。

说明

当新于版本D的硬件可用时，SAV功能块仅应用于SINAMICS INTEGRATED上的D425 (6AU1425-0AA00-0AA0)、D435 (6AU1435-0AA00-0AA1) 和D445 (6AU1445-0AA00-0AA0)。

告警!

对于模块SIMOTION D410 / CX32、D425、D435、D445、D445-1/CX32-2、D445-2和D455-2, 必须考虑到非易失数据的行为特性。关于非易失数据, 参见版本11/2010及以上的D410调试与硬件安装手册/D4x5调试与硬件安装手册/ D4x5-2调试与硬件安装手册中的调试(硬件) → 用户存储器概念 → 用户存储器属性。视硬件设备而定, 那些模块上的保留数据只保留有限的一段时间。对于SIMOTION V4.2及以上版本, SIMOTION系统功能“savePersistentMemoryData”包括SINAMICS_Integrated(如: DCC功能块SAV、SAV_BY、SAV_D、SAV_I)上属于D425、D435、D445和D445-1模块的保留数据, 该数据备份存储于CF卡上。

说明

对于SCOUT V4.2及以上版本, DCC SIMOTION的SAV功能块数值可以通过“保存和恢复保留数据”功能进行备份。

输入M处功能块模式设置为:

写模式 (M = 1)

- 输入变量IS周期性写给输出QS。
- 输入变量IS同样传递给系统, 进行保留存储。借此, 覆盖已经保存的数值。

读模式 (M = 0)

- 当前保存的数值在输出QS处输出。输入IS的数值不保存。
- 输出VLD = 1表示QS有效。当功能块初始化, 系统的保留存储空间重新创建时, VLD = 0。此时, QS无效, 包含其默认值。初次写入数值(M = 1)时, VLD状态则变为1。

初始化

通过功能块的背景名称在SAV功能块和保留存储中的数值之间进行分配。当功能块插入图中时, DCC编辑器自动生成唯一的背景名称。背景名称由功能块的调用路径构成, 命名规则如下:

(图名称)/(子图1名称)/(子图2名称)/../(功能块名称)

例示背景名称如下:

DCC_1/CFC1/CFC2/CFC3/SAV1

图名称	DCC_1
子图1名称	CFC1
子图2名称	CFC2
子图3名称	CFC3
功能块名称	SAV1

此背景名称控制输出QS是否初始化为默认值或是否在INIT模式下输出最近保存的数值。目标设备会检查保留数值是否已经为功能块的背景名称而保存。如果没有保存的话, 系统会重新创建存储空间, 输出变量QS的默认值会传递给系统进行保留存储, 并设置VLD = 0。当一个保留数值已经为背景名称而保存时, 那么该值读/写给输出QS, 并输出状态VLD = 1。

如果功能块无存储空间可用, 则设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
IS	输入变量	16#00	BYTE	
M	模式	0	0/1	
QS	输出变量	16#00	BYTE	
VLD	输出变量有效	0	0/1	
QF	无保留存储	0	0/1	

据

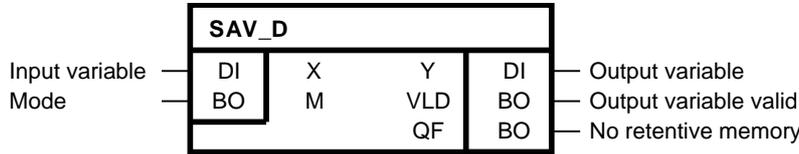
可在线插入	否
特别注意点	每个SINAMICS或SINAMICS INTEGRATED驱动单元最多支持10个功能块 (SAV、SAV_BY、SAV_I、SAV_D) 用于保留存储。保留存储最大支持40字节的用户数据。对于STARTER/SCOUT V4.2及以上版本, 只在一致性检查时检查功能块数量, 一致性检查的菜单入口为“Project -> Check consistency”或驱动单元的右键菜单中“Check consistency”。

5.20 SAV_D 数值缓冲 (DOUBLE INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- SAV_D (保存) 启用DOUBLE INTEGER型输入变量的保留存储。

操作模式

该功能块是DOUBLE INTEGER数值的保留读/写存储器。

SAV功能块的保存数值在以下情况下不会保留：

- 用户已经清空了保留存储器。
- 配置功能块的DCC图已经删除，且修改已经传递给目标设备。
- 功能块已经删除，且修改已经传递给目标设备。
- 背景名称已经修改，且相应的修改已经传递给目标设备。

该值在以下情况保留：

- 下载时，背景名称没有修改
- 目标设备斜升，且存储卡上无配置数据。 只有在下载后才会释放缺失SAV功能块的存储。 这样的话，当固件更新时，也会保留数据。
- 当已经添加或移除了另一个SAV功能块时
- 当DCBLIB更新后进行配置数据下载时
- 当已经添加或移除了另一个DO并将其下载至目标设备时
- 当已经添加或移除了另一个DCC图并将其下载至目标设备时
- 当目标设备按电源故障前相同的配置数据斜升时

只有当输出QF为0时功能块才处于激活状态。这表明目标设备上的预留存储空间可用于存储输入值。

说明

当新于版本D的硬件可用时，SAV功能块仅应用于SINAMICS INTEGRATED上的D425 (6AU1425-0AA00-0AA0)、D435 (6AU1435-0AA00-0AA1) 和D445 (6AU1445-0AA00-0AA0)。

告警!

对于模块SIMOTION D410 / CX32、D425、D435、D445、D445-1/CX32-2、D445-2和D455-2，必须考虑到非易失数据的行为特性。关于非易失数据，参见版本11/2010及以上的D410调试与硬件安装手册/D4x5调试与硬件安装手册/ D4x5-2调试与硬件安装手册中的调试（硬件）→ 用户存储器概念 → 用户存储器属性。视硬件设备而定，那些模块上的保留数据只保留有限的一段时间。对于SIMOTION V4.2及以上版本，SIMOTION系统功能“savePersistentMemoryData”包括SINAMICS_Integrated（如：DCC功能块SAV、SAV_BY、SAV_D、SAV_I）上属于D425、D435、D445和D445-1模块的保留数据，该数据备份存储于CF卡上。

说明

对于SCOUT V4.2及以上版本，DCC SIMOTION的SAV功能块数值可以通过“保存和恢复保留数据”功能进行备份。

输入M处功能块模式设置为：

写模式 (M = 1)

- 输入变量X周期性写给输出Y。
- 另外，输入变量X传递给系统，进行保留存储。借此，覆盖已经保存的数值。

读模式 (M = 0)

- 当前保存的数值在输出Y处输出。输入X的数值不保存。
- 输出VLD = 1表示Y有效。当功能块初始化，系统的保留存储空间重新创建时，VLD = 0。此时，Y无效，包含其默认值。初次写入数值 (M = 1) 时，VLD状态则变为1。

初始化

通过功能块的背景名称在SAV功能块和保留存储中的数值之间进行分配。当功能块插入图中时，DCC编辑器自动生成唯一的背景名称。背景名称由功能块的调用路径构成，命名规则如下：

(图名称)/(子图1名称)/(子图2名称)/../(功能块名称)

例示背景名称如下：

DCC_1/CFC1/CFC2/CFC3/SAV1

图名称	DCC_1
子图1名称	CFC1
子图2名称	CFC2
子图3名称	CFC3
功能块名称	SAV1

此背景名称控制输出Y是初始化为默认值还是在INIT模式下输出最近保存的数值。目标设备会检查保留数值是否已经为功能块的背景名称而保存。如果没有保存的话，系统会重新创建存储空间，输出变量Y的默认值会传递给系统进行保留存储，并设置VLD = 0。当一个保留数值已经为背景名称而保存时，那么该值读/写给输出Y，并输出状态VLD = 1。

如果功能块无存储空间可用，则设置QF = 1。

5.20 SAV_D 数值缓冲 (DOUBLE INTEGER型)

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
M	模式	0	0/1	
Y	输出变量	0	DINT	
VLD	输出变量有效	0	0/1	
QF	无保留存储	0	0/1	

据

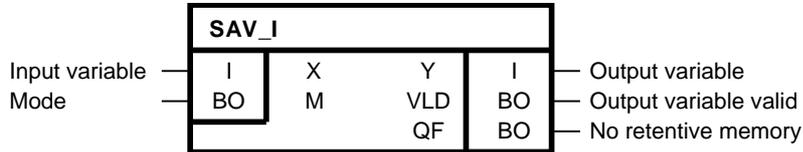
可在线插入	否
特别注意点	每个SINAMICS或SINAMICS INTEGRATED驱动单元最多支持10个功能块 (SAV、SAV_BY、SAV_I、SAV_D) 用于保留存储。保留存储最大支持40字节的用户数据。对于STARTER/SCOUT V4.2及以上版本, 只在一致性检查时检查功能块数量, 一致性检查的菜单入口为“Project -> Check consistency”或驱动单元的右键菜单中“Check consistency”。

5.21 SAV_I 数值缓冲 (INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- SAV_I (保存) 启用INTEGER型输入变量的保留存储。

操作模式

该功能块是INTEGER数值的保留读/写存储器。

SAV功能块的保存数值在以下情况下不会保留：

- 用户已经清空了保留存储器。
- 配置功能块的DCC图已经删除，且修改已经传递给目标设备。
- 功能块已经删除，且修改已经传递给目标设备。
- 背景名称已经修改，且相应的修改已经传递给目标设备。

该值在以下情况保留：

- 下载时，背景名称没有修改
- 目标设备斜升，且存储卡上无配置数据。 只有在下载后才会释放缺失SAV功能块的存储。 这样的话，当固件更新时，也会保留数据。
- 当已经添加或移除了另一个SAV功能块时
- 当DCBLIB更新后进行配置数据下载时
- 当已经添加或移除了另一个DO并将其下载至目标设备时
- 当已经添加或移除了另一个DCC图并将其下载至目标设备时
- 当目标设备按电源故障前相同的配置数据斜升时

只有当输出QF为0时功能块才处于激活状态。这表明目标设备上的预留存储空间可用于存储输入值。

说明

当新于版本D的硬件可用时，SAV功能块仅应用于SINAMICS INTEGRATED上的D425 (6AU1425-0AA00-0AA0)、D435 (6AU1435-0AA00-0AA1) 和D445 (6AU1445-0AA00-0AA0)。

告警!

对于模块SIMOTION D410 / CX32、D425、D435、D445、D445-1/CX32-2、D445-2和D455-2, 必须考虑到非易失数据的行为特性。关于非易失数据, 参见版本11/2010及以上的D410调试与硬件安装手册/D4x5调试与硬件安装手册/ D4x5-2调试与硬件安装手册中的调试(硬件) → 用户存储器概念 → 用户存储器属性。视硬件设备而定, 那些模块上的保留数据只保留有限的一段时间。对于SIMOTION V4.2及以上版本, SIMOTION系统功能“savePersistentMemoryData”包括SINAMICS_Integrated(如: DCC功能块SAV、SAV_BY、SAV_D、SAV_I)上属于D425、D435、D445和D445-1模块的保留数据, 该数据备份存储于CF卡上。

说明

对于SCOUT V4.2及以上版本, DCC SIMOTION的SAV功能块数值可以通过“保存和恢复保留数据”功能进行备份。

输入M处功能块模式设置为:

写模式 (M = 1)

- 输入变量X周期性写给输出Y。
- 另外, 输入变量X传递给系统, 进行保留存储。借此, 覆盖已经保存的数值。

读模式 (M = 0)

- 当前保存的数值在输出Y处输出。输入X的数值不保存。
- 输出VLD = 1表示Y有效。当功能块初始化, 系统的保留存储空间重新创建时, VLD = 0。此时, Y无效, 包含其默认值。初次写入数值 (M = 1) 时, VLD状态则变为1。

初始化

通过功能块的背景名称在SAV功能块和保留存储中的数值之间进行分配。当功能块插入图中时, DCC编辑器自动生成唯一的背景名称。背景名称由功能块的调用路径构成, 命名规则如下:

(图名称)/(子图1名称)/(子图2名称)/../(功能块名称)

例示背景名称如下:

DCC_1/CFC1/CFC2/CFC3/SAV1

图名称	DCC_1
子图1名称	CFC1
子图2名称	CFC2
子图3名称	CFC3
功能块名称	SAV1

此背景名称控制输出Y是初始化为默认值还是在INIT模式下输出最近保存的数值。目标设备会检查保留数值是否已经为功能块的背景名称而保存。如果没有保存的话, 系统会重新创建存储空间, 输出变量Y的默认值会传递给系统进行保留存储, 并设置VLD = 0。当一个保留数值已经为背景名称而保存时, 那么该值读/写给输出Y, 并输出状态VLD = 1。

如果功能块无存储空间可用, 则设置QF = 1。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	INT	
M	模式	0	0/1	
Y	输出变量	0	INT	
VLD	输出变量有效	0	0/1	
QF	无保留存储	0	0/1	

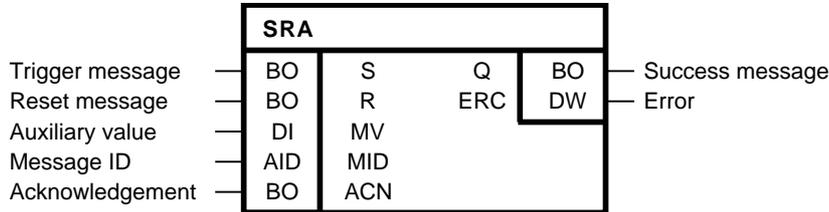
据

可在线插入	否
特别注意点	每个SINAMICS或SINAMICS INTEGRATED驱动单元最多支持10个功能块 (SAV、SAV_BY、SAV_I、SAV_D) 用于保留存储。保留存储最大支持40字节的用户数据。对于STARTER/SCOUT V4.2及以上版本, 只在一致性检查时检查功能块数量, 一致性检查的菜单入口为“Project -> Check consistency”或驱动单元的右键菜单中“Check consistency”。

5.22 SRA 触发/复位消息

SIMOTION SINAMICS

符号



简要说明

SCOUT上配置的消息可以通过SRA（设定告警）功能块进行触发或复位。消息发送给HMI，并在SIMOTION目标设备的消息缓存中输入。消息缓存包含所有的激活信息。配置SRA功能块是为了实现多重背景功能。多个功能块背景可以触发同一个消息编号。

操作模式

- SCOUT上配置的消息在MID消息编号处输入，如：消息
- 输入ACN =1表示该消息是一条可确认的消息。在这种情况下，当用户通过HMI确认了该消息时，该消息同样只会在复位后消失。对于不可确认的消息，配置ACN = 0。
- 如果该参数在SCOUT上配置时已经明确，必须在参数MV处输入过程值/校正值。可以配置DINT型数值。配置消息过程中，校正值插入具有特殊句法的消息描述中。调用过程值以@开头以@结尾。其中的参数指定数值的输出和格式。只有DINT型校正值可以适用于通过SRA功能块触发的消息。关于消息配置中校正值的句法，参见SIMOTION SCOUT在线帮助。
- 该消息通过输入S处的上升沿触发。如果通过输入S处的新消息编号1来调用该功能块，那么该消息同样触发。消息缓存最多支持输入40条消息。当消息缓存已满，SRM功能块通过输入IS的上升沿来激活时，该功能块会确认应答一条错误消息。消息未输入。
- 该消息通过输入R处的上升沿复位。
- 当在调用过程中同时设置输入S和输入R处的上升沿时，R具有优先权，即复位该消息。
- 块输出Q =1表示消息已经成功设置或复位。输出再次通过输入S或输入R处的上升沿进行设置。
- 当Q = 0时，输出ERC显示错误码。该错误码说明了消息不可以发布的原因：
- 指定值可以显示为常量间OR逻辑运算。
 - 16#0000无错误
在消息列表中为接受的消息创建一个条目。
从消息列表中为外发的消息删除该条目。
 - 16#8001该消息名称不允许。
 - 16#8002溢位导致消息丢失
消息列表中已经有40个条目。消息列表中并没有创建条目。
 - 16#8003溢位导致消息丢失（信号未发出，信号溢位）。

用于客户端通知的发送缓存器依然被上一事件占用。

消息列表中没有创建条目。

当连续高频率地进行上升沿和下降沿的函数调用时，也可能会发生错误。

- 16#8004消息重复，拒绝该消息（连续两次调用接受或发送的消息）。

消息列表中没有创建条目。

- 16#8005无显示设备信号。依然在列表中输入消息。

- 16#8007该消息名称的任务还未启动

（初始调用S = FALSE）。

下降沿（发出消息）之前没有上升沿（接收消息）。

消息列表中没有创建条目。

- 16#8008消息缓存中该ID的消息已经激活。

当该ID的消息存在于消息缓存且再次设置同样的ID时，会发出该消息。没有生成新的消息缓冲条目。

- 16#8009内部错误

- 16#8010条目已拒绝：消息确认存储已满。

多个块连接

块 连接	说明	预分配	取值范围	数
S	触发消息	0	0/1	
R	复位消息	0	0/1	
MV	校正值	0	DINT	
MID	消息ID	STRUCTALARMID#NIL	StructAlarmId	
ACN	确认	0	0/1	
Q	成功消息	0	0/1	
ERC	错误	0	0- 0x80FF	

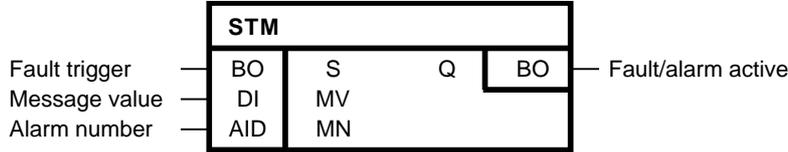
据

SIMOTION	4.3
SINAMICS	-
可在线插入	是
特别注意点	-

5.23 STM 故障/告警触发器

SIMOTION SINAMICS

符号



简要说明

通过STM（设定消息）功能块可以在DO上触发预定义的消息（故障或告警）。故障（如：STARTER、AOP）在DO的故障缓存或告警缓存中显示和输入。以下规则适用于该功能块类型：

- 分配给背景的消息号（故障/告警号）必须在51050和51069之间（预设值为51050）。
- DO中多个背景可以共用同一个消息号（可以根据不同的背景设置消息）。
然而，出于性能考虑，SFM功能块不配置多重背景功能。对于同一DO的故障，当多重背景共用同一消息号时，下图显示了其行为特性。当没有附加电路时，同一消息号的块背景不进行协调。当块背景要在不同的采样时间点运行时，不可以进行块背景协调。因此，建议为DO中的每个背景分配唯一的消息号。
- 消息描述预先确定，且不可以修改（参见下表）。
- 消息类型不可以改变。故障不可以再定义给告警，反之亦然。
- 故障响应的默认设置为OFF2。该设置可以在SINAMICS基础系统参数中进行修改：
 - p2100[0..19] “设置故障响应的故障号” 和
 - p2101[0..19] “设置故障响应”
- 确认模式的默认设置为IMMEDIATE。该设置可以在SINAMICS基础系统参数中进行修改：
 - p2126[0..19] “设置确认模式的故障号” 和
 - p2127[0..19] “设置确认模式”

下表说明了该属性的默认设置。可在用户文档中找到各种设置的可能选项：

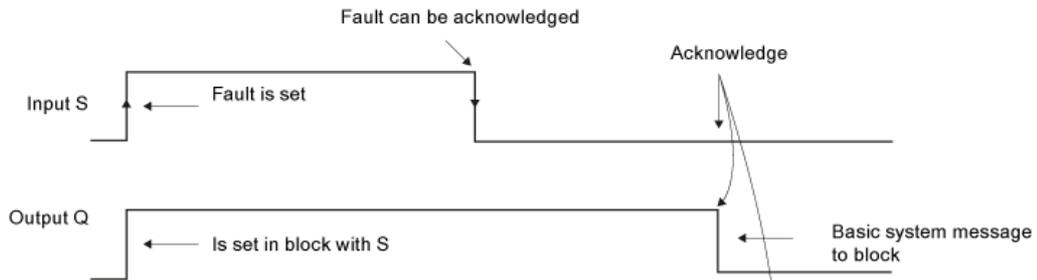
消息类型	告警号	响应	确认	消息描述
故障（不可以修改）	F51050-F51059	OFF2（可以通过p2100/p2101进行修改）	IMMEDIATE（可以通过p2126/p2127进行修改）	DCC：故障F5105x 附加值：%d（x：= 0到9）
告警（不可以修改）	A51060-A51069			DCC：告警A5106x 附加值：%d（x：= 0到9）

故障

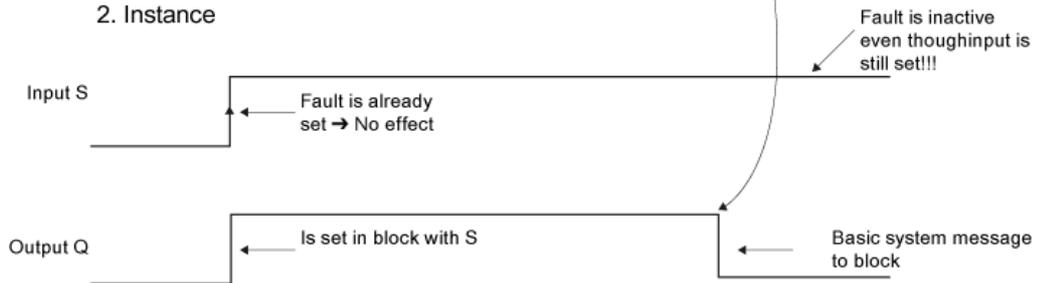
必须在输入MN处指定待触发故障号（F51050 - F51059）。输入S的正沿触发DO处的故障。该值在DO的故障缓存中输入，执行DO指定的响应。借此，功能块设置输出Q。只要故障处于激活状态，输出Q就保持设定值。在输入S的负沿之后，故障可以根据消息的确认属性进行确认（模拟系统故障：参见下图中的第一个背景实例）。

输入MV可以用于给故障添加附加信息（故障值）。当故障在正沿处触发并在D0的故障缓存中输入时，故障值传递给输入S。

1. Instance



2. Instance

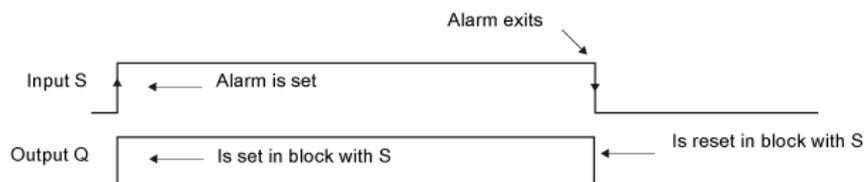


一个D0中带有相同故障号的双重背景举例（无附加RC电路）

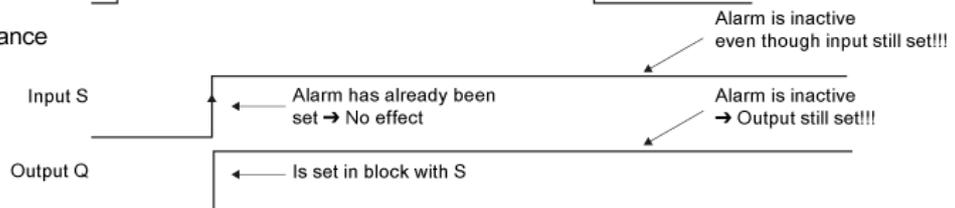
告警

必须在输入MN处指定待触发告警号（A51060 -A51069）。输入S的正沿触发分配给功能块的告警。在D0的告警缓存中输入该告警号。借此可以设置输出Q。只要告警处于激活状态，输出就保持设定值。告警进行自确认，但当输入S复位时，则被确认（参见下图）。同样在告警缓存中输入的附加信息（告警值）可以通过输入MV添加到告警中。

1. Instance



2. Instance



一个D0中带有相同告警号的双重背景举例（无附加RC电路）

说明

关于向其他驱动对象转发故障和告警的原则，参见版本07/2007及以上版本的SINAMICS S120调试手册中“诊断 > 消息-故障和告警”章节。该原则同样适用于DCC驱动对象上STM功能块触发的故障和告警。

5.23 STM 故障/告警触发器

多个块连接

块 连接	说明	预分配	取值范围	数
S	故障触发器	0	0/1	
MV	消息值	0	DINT	
MN	告警号	F51050	F51050- F51059、A51060- A51069	
Q	故障/告警激活	0	0/1	

据

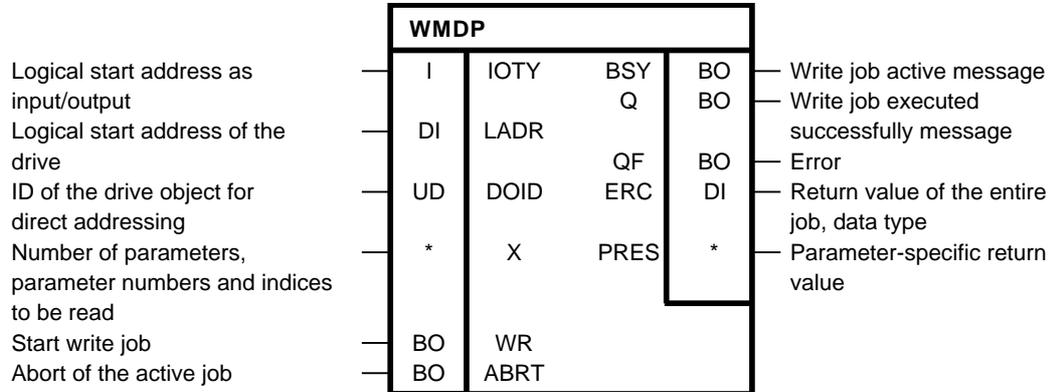
可在线插入	否
特别注意点	-

5.24 WMDP 从控制器中写入驱动参数

☑ SIMOTION

☐ SINAMICS

符号



简要说明

WMDP功能块最大允许从DCC SIMOTION程序中写入23个SINAMICS参数。

只支持SINAMICS驱动。

对于错误诊断，可以进行错误码ERC评估。ERC会根据PROVIDrive DPV1对应上参数存取的错误码。

在本文档的附录A.2中可以找到可能的错误码，还可以在SIMOTION通讯系统手册，PROVIDrive小节，非循环通讯（基础模式参数存取）→错误评估 子小节，基础模式参数存取响应中的错误码表中找到。

WMDP功能块可用于SIMOTION V4.2及以上版本。

操作模式

首先，输入用于寻址驱动的块输入并且选择待写参数和待写值。当参数未索引时，必须设置IDX = 0。通过输入WR的正沿启动异步写任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率和通讯负荷，且会随任务而改变。在写任务执行时，忽略输入WR的其他正沿。

没有在DCC任务中进行真正的参数读写。功能块背景仅控制通讯命令。读写任务的结果必须在随后的任务周期内在块输出处获得。

通过全局变量或自定义块类型进行评估。

输出Q = 1表示参数已成功写入。

当存取发生错误时，通过设置QF = 1来表示。

对于错误诊断，会进行错误码ERC评估。

尽管整体状态Q = 1，单个参数的写任务可能已经由于错误而中止。PRES没有出错的写任务已经执行。

可以针对参数特定的返回值PRES评估单个写任务的错误状态。

通过ABRT输入的正沿可以中止一个激活的任务。ABRT信号必须在至少一个周期内取值为1。

注意

用来描述一个驱动参数的块不能因安全原因而用于修改SINAMICS Safety Integrated功能的参数。在功能性安全层面（Safety Integrated）不能将DCC认为是安全的。

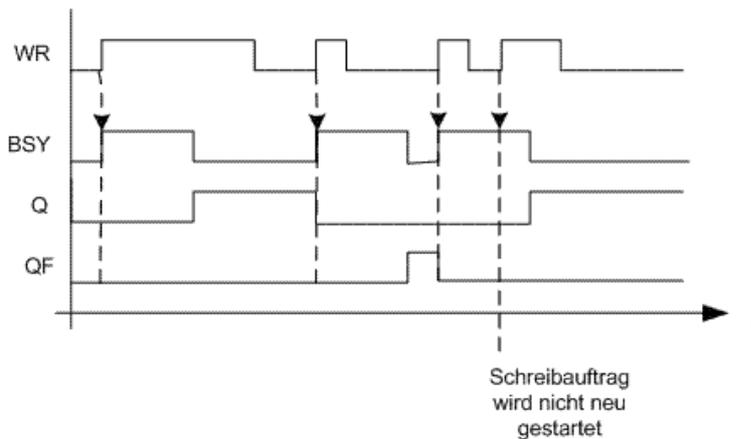
数量框架

该功能块最多支持写入23个SINAMICS参数。

说明

该功能块类型可以对DCC图中的每个驱动对象进行多次背景化。最多可以并行执行16个参数任务（项参数任务不仅包括DCB功能块RMDP和WMDP，还包括ST功能_readDriveParameter、_readDrive(Multi)Parameter-Description、_writeDrive(Multi)Parameter和_readDriveFaults）。另外，如果在执行参数任务的同时，系统功能_readRecord和_writeRecord也在并行使用，请确保它们没有被一个DP站同时执行。原因：使用了同样的通讯服务。当所有的通讯通道被占用时，功能块会等待，直至有通讯通道可用：BSY激活。无需再次设置RD信号。每个配置的DP站始终只允许处理一个参数任务。当一个新任务发送给同一个DP站时，功能块会内部等待，直至通讯可用：BSY激活。当参数任务（仅对于异步调用）不需要进一步执行时，可以通过ABRT输入来中止。参数函数最大支持240字节的数据长度。最大可写参数数量取决于数据类型和参数必需元素的数量。因此，最多可同时写23个参数，但参数类型必须为WORD且元素数量为1。当任务与此相悖（如参数类型为DWORD）时，则减小数量。在写任务开始时检查数据长度，一旦发生错误时，则在输出ERC处用16#FFFF8110表示。

时序图



以下数据集可用于写入参数:

对于(外部或内部集成)PROFIBUS,不管是通过有效(0>=doId<=254)还是无效(doId=255)doID调用该函数,总是读出数据集47。

对于PROFINET,两个数据集可用:

-基础模式参数存取 -本地(数据集0xB02E)

当函数没有指定或只指定一个无效doID(doId=255)时,此数据集用于SIMOTION。然后通过参数存取点(PAP)对适当的DO执行存取。

-PAP地址可直接指定,也可通过周期数据的逻辑地址指定(如DO轴的256)。接着SIMOTION根据此地址确定相关程序流程图,存取正确的地址。

程序流程图必须位于1号子槽位(HW Config中的配置)。

-基础模式参数存取 -全局(数据集0xB02F)

当输入有效doID(0>=doId<=254)时,使用该数据集。由于仅通过doID进行赋值,因此任何有效程序流程图或地址都可以指定。

块输入描述

'IOTY': 驱动逻辑起始地址的输入/输出赋值。

当为198时: INPUT,驱动的逻辑地址位于输入范围内。

当为199时: OUTPUT,驱动的逻辑地址位于输出范围内。

诊断地址总是为INPUT型。

'LADR': 驱动逻辑起始地址的指定。当同时使用可选参数DOID时,可以指定该站的任意地址(诊断地址优先)。

对于PROFINET,通过驱动对象的参数存取点(PAP)进行参数存取。

作为驱动逻辑起始地址的备选,推荐使用相关PAP的诊断地址。

'DOID': 用于驱动对象的直接寻址。

在如下条件下,DOID无法指定或只能指定无效值(>254):

-DP从I/O设备不支持通过DOID进行存取(P978不执行)。

-数据集0xB02F不支持(仅对于PROFINET)。

-通过DO的PAP进行存取(仅对于PROFINET)。

'X.NUMP': 待写入参数的数量。

'X.PAR[].NUM': 指定参数编号,根据该编号写入值。

'X.PAR.IDX': 索引值的参数索引,0表示索引0。对于非索引值,必须指定参数索引0。

'X.PAR.DTYP': 返回参数的数据类型(关于编码情况,参见PROFIdrive分布图)。

数据类型必须与驱动中的参数类型相匹配。该功能块执行特定数据类型的传递。

当指定的数据类型与SINAMICS中参数的实际数据类型不匹配时,则返回错误。

'X.PAR.X': 待写入驱动的数据,DWORD。转换功能块要求针对不同的数据类型。转换功能块R_DW应用于写入REAL参数。转换功能块B_DW应用于写入BYTE参数。

'WR': 启动写任务。

'ABRT': 中止运行的任务。

块输出描述

'Q': 无错完成的任务。

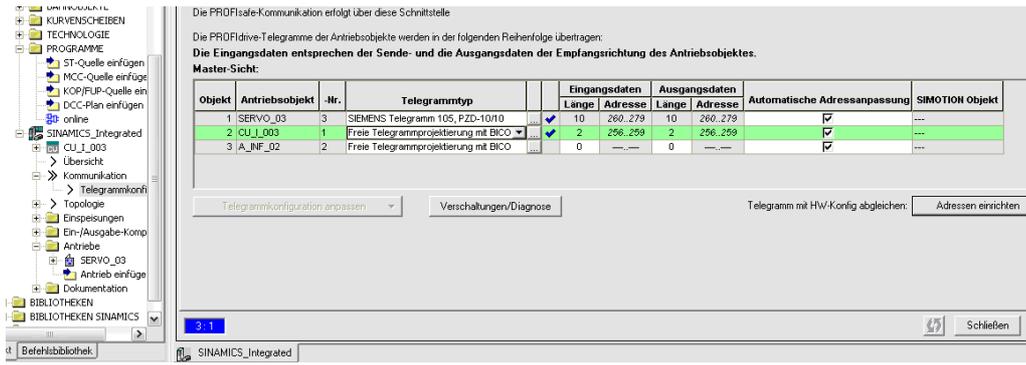
'QF' 含错完成的任务。

'ERC': 对应_writeDriveMultiParameter函数返回值functionResult的值。

参数设置示例

为了能够写入驱动对象(如:SERVO_03)的某些参数,执行以下操作:

首先，设置正确的消息框架配置。



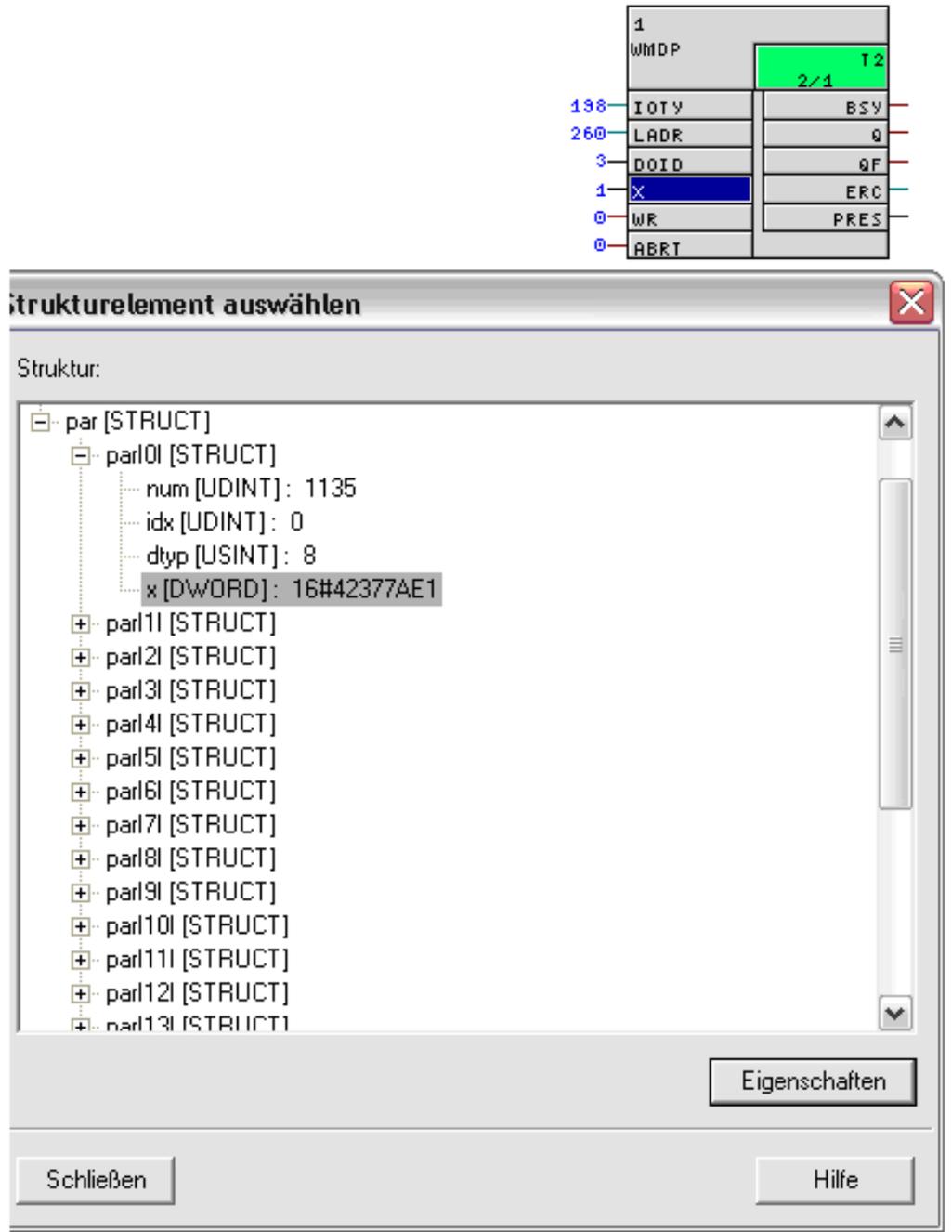
接着在WMDP功能块中设置所需的DO地址。为此，需要设置块输入LADR为消息框架中设置的地址（260），并设置块输入DOID为消息框架中设置的编号（3）。



选择输入X处的待写入参数，如 P1135(0) OFF3斜降时间。

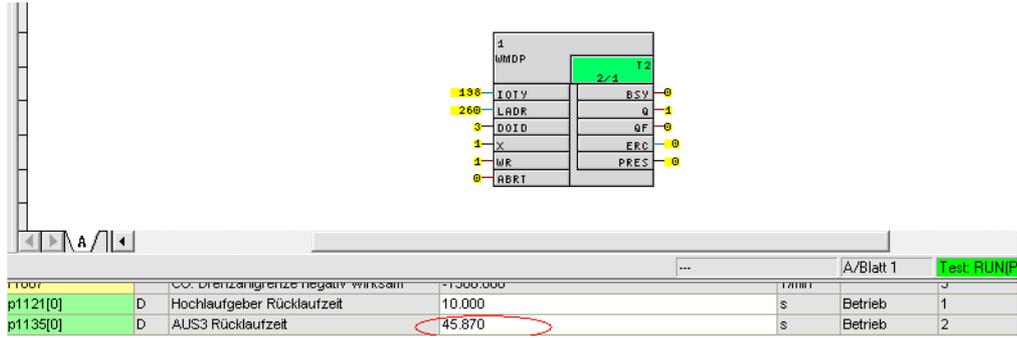
330	p1135[D]	D	hochlaufgebet_rücklaufzeit	10.000	s	Betrieb	1	0	5995
331	p1135[D]	D	AUS3 Rücklaufzeit	0.000	s	Betrieb	2	0	600
332	rt140[D]	C	Bl: Hochlaufgeber freigegeben	SERVO_03:r2090.4		Betriebsbereit	3		

为此，需要双击块输入X，选择第一个结构件，再分别在num和idx处输入参数编号（1135）和索引（0）。在 ' dtyp' 处输入数据类型。输入的数据类型必须对应于参数的数据类型，如例8（浮点）。作为DWORD，在X处输入值16#4237AE1。关于数据类型的编码，参见SIMOTION列表手册系统功能/变量设备 → 系统功能 - 设备1 → _readDriveMultiParameterDescription。



然后设置块输入WR为1，以此来启动写任务。 计算结果在专家列表中显示。

5.24 WMDP 从控制器中写入驱动参数



多个块连接

块 连接	说明	预分配	取值范围	数
IOTY	作为输入/输出的逻辑起始地址	0	0: 无效 198: 输入地址 199: 输出地址	
LADR	驱动的逻辑起始地址	-1	DINT	
DOID	用于直接寻址的驱动对象ID	255	0 .. 254, 255: 无效	
X	参数数量、待读取的参数编号和索引			
X. NUMP	待写入参数的数量	1	1 .. 23	
X. PAR	参数描述			
X. PAR. NUM	参数编号	0	1..65535	
X. PAR. IDX	参数索引	0	1..65535	
X. PAR. DTYP	待写入参数的数据类型	0	USINT	
X. PAR. X	参数值	0	DWORD	
WR	启动写任务	0	0/1	
ABRT	中止运行的任务	0	0/1	
BSY	写任务运行消息	0	0/1	
Q	成功执行写任务的消息	0	0/1	
QF	错误	0	0/1	
ERC	整个任务的返回值, 数据类型	16#0000	DWORD	
PRES	参数的返回值	0	DWORD	

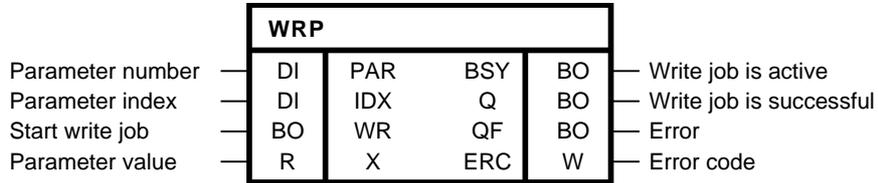
据

SIMOTION	V4.2及以上版本
SINAMICS	-
可在线加载	否
处理环境	循环, 等距
特别注意点	-

5.25 WRP 写入驱动参数 (REAL型)

 SIMOTION SINAMICS

符号



简要说明

该功能块在本地驱动对象上启动REAL型驱动参数的异步写入。

操作模式

待写入参数的编号与索引分别在输入PAR和IDX处表示。当参数未索引时，必须设置IDX = 0。该参数始终在计算功能块功能图的驱动对象上写入。不可以在多个驱动对象上存取参数。

参数值通过输入X指定。可以在输入WR的正沿上启动异步写任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在写任务运行时，忽略输入WR的其他正沿。

Output Q = 1表示参数成功写入。当存取发生错误时，则通过QF = 1来表示。

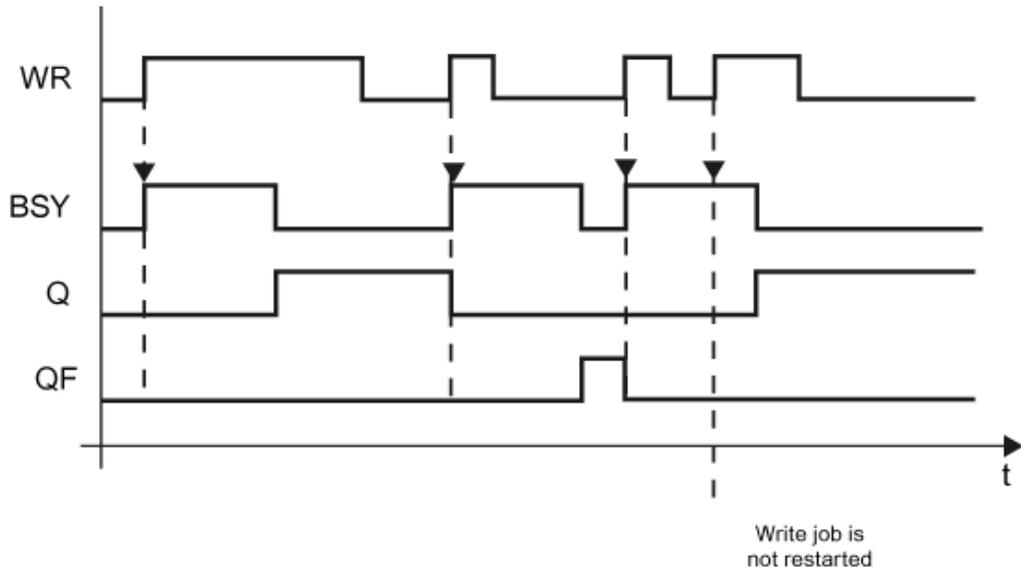
在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

注意

用来描述一个驱动参数的块不能因安全原因而用于修改SINAMICS Safety Integrated功能的参数。在功能性安全层面 (Safety Integrated) DCC被认为是与安全无关的。

时序图



数量框架

不同块背景的异步任务数可以并行发布。 一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	0..2 ¹⁶	
IDX	参数索引	0	0..2 ¹⁶	
WR	启动写任务	0	0/1	
X	参数值	0.0	REAL	
BSY	执行写任务	0	0/1	
Q	成功完成写任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

据

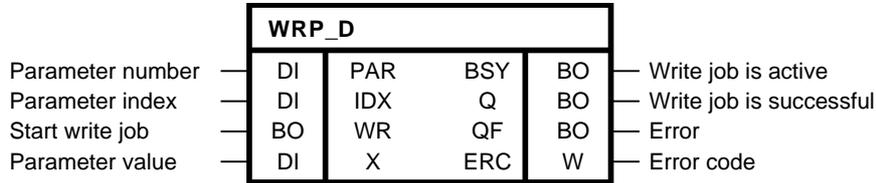
可在线加载	是
特别注意点	-

5.26 WRP_D 写入驱动参数 (DOUBLE INTEGER型)

□ SIMOTION

☑ SINAMICS

符号



简要说明

该功能块在本地驱动对象上启动DOUBLE INTEGER型驱动参数的异步写入。

注意

用来描述一个驱动参数的块不能因安全原因而用于修改SINAMICS Safety Integrated功能的参数。在功能性安全层面 (Safety Integrated) DCC被认为是与安全无关的。

操作模式

待写入参数的编号与索引分别在输入PAR和IDX处表示。当参数未索引时，必须设置IDX = 0。该参数始终在计算功能块功能图的驱动对象上写入。不可以在多个驱动对象上存取参数。

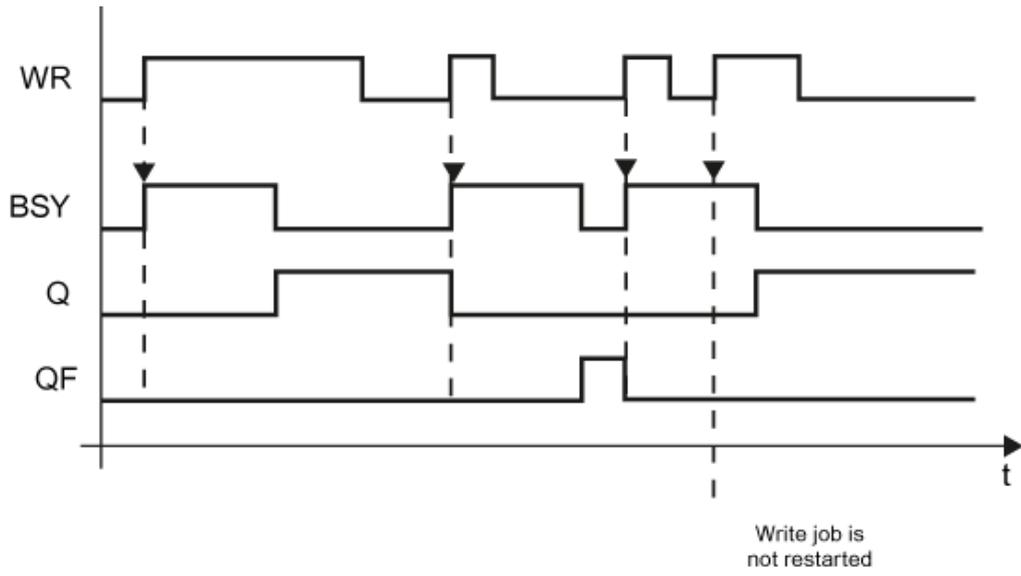
参数值通过输入X指定。可以在输入WR的正沿上启动异步写任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在写任务运行时，忽略输入WR的其他正沿。

Output Q = 1表示参数成功写入。当存取发生错误时，则通过QF = 1来表示。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。 一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	0..2 ¹⁶	
IDX	参数索引	0	0..2 ¹⁶	
WR	启动写任务	0	0/1	
X	参数值	0	DINT	
BSY	执行写任务	0	0/1	
Q	成功完成写任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

据

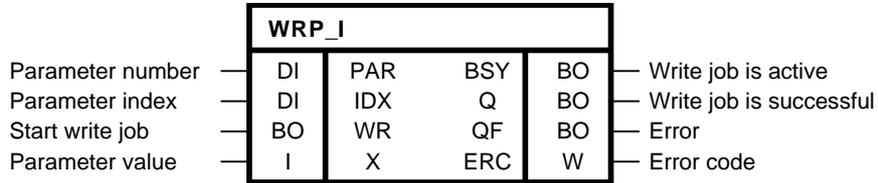
可在线加载	是
特别注意点	-

5.27 WRP_I 写入驱动参数 (INTEGER型)

□ SIMOTION

☑ SINAMICS

符号



简要说明

该功能块允许在本地驱动对象上进行INTEGER型驱动参数的异步写入

注意

用来描述一个驱动参数的块不能因安全原因而用于修改SINAMICS Safety Integrated功能的参数。在功能性安全层面 (Safety Integrated) DCC被认为是与安全无关的。

操作模式

待写入参数的编号与索引分别在输入PAR和IDX处表示。当参数未索引时，必须设置IDX = 0。该参数始终在计算功能块功能图的驱动对象上写入。不可以在多个驱动对象上存取参数。

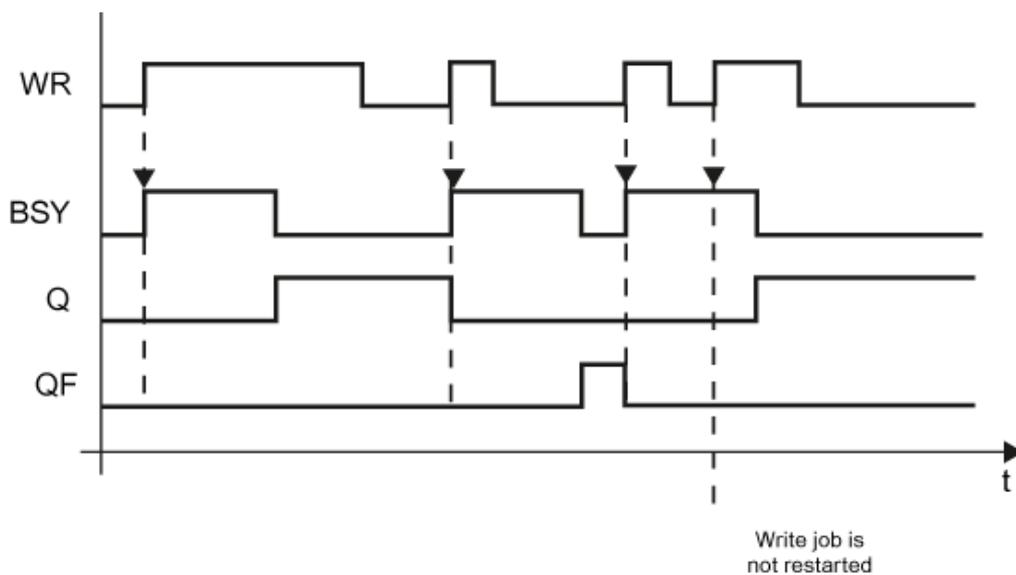
参数值通过输入X指定。可以在输入WR的正沿上启动异步写任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在写任务运行时，忽略输入WR的其他正沿。

Output Q = 1表示参数成功写入。当存取发生错误时，则通过QF = 1来表示。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	$0..2^{16}$	
IDX	参数索引	0	$0..2^{16}$	
WR	启动写任务	0	0/1	
X	参数值	0	INT	
BSY	执行写任务	0	0/1	
Q	成功完成写任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

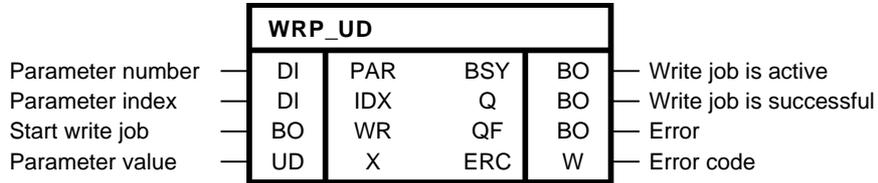
据

可在线加载	是
特别注意点	-

5.28 WRP_UD 写入驱动参数 (UNSIGNED DOUBLE INTEGER型)

 SIMOTION SINAMICS

符号



简要说明

WRP_UD (写参数) 在本地驱动对象上启动UNSIGNED DOUBLE INTEGER型驱动参数的异步写入。

注意

用来描述一个驱动参数的块不能因安全原因而用于修改SINAMICS Safety Integrated功能的参数。在功能性安全层面 (Safety Integrated) DCC被认为是与安全无关的。

操作模式

待写入参数的编号与索引分别在输入PAR和IDX处表示。当参数未索引时，必须设置IDX = 0。该参数始终在计算功能块功能图的驱动对象上写入。不可以在多个驱动对象上存取参数。

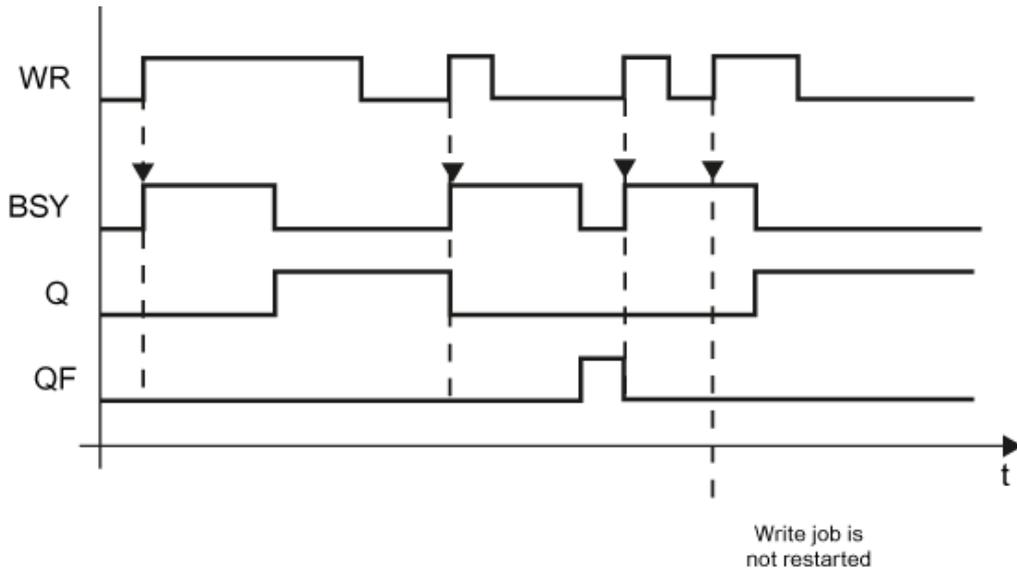
参数值通过输入X指定。可以在输入WR的正沿上启动异步写任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在写任务运行时，忽略输入WR的其他正沿。

Output Q = 1表示参数成功写入。当存取发生错误时，则通过QF = 1来表示。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。 一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	0..2 ¹⁶	
IDX	参数索引	0	0..2 ¹⁶	
WR	启动写任务	0	0/1	
X	参数值	0	UDINT	
BSY	执行写任务	0	0/1	
Q	成功完成写任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

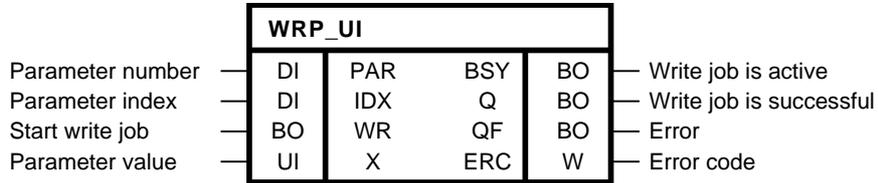
据

可在线加载	是
特别注意点	-

5.29 WRP_UI 写入驱动参数 (UNSIGNED INTEGER型)

SIMOTION SINAMICS

符号



简要说明

WRP_UI (写参数) 在本地驱动对象上启动UNSIGNED INTEGER型驱动参数的异步写入。

注意

用来描述一个驱动参数的块不能因安全原因而用于修改SINAMICS Safety Integrated 功能的参数。在功能性安全层面 (Safety Integrated) DCC被认为是与安全无关的。

操作模式

待写入参数的编号与索引分别在输入PAR和IDX处表示。当参数未索引时，必须设置IDX = 0。该参数始终在计算功能块功能图的驱动对象上写入。不可以在多个驱动对象上存取参数。

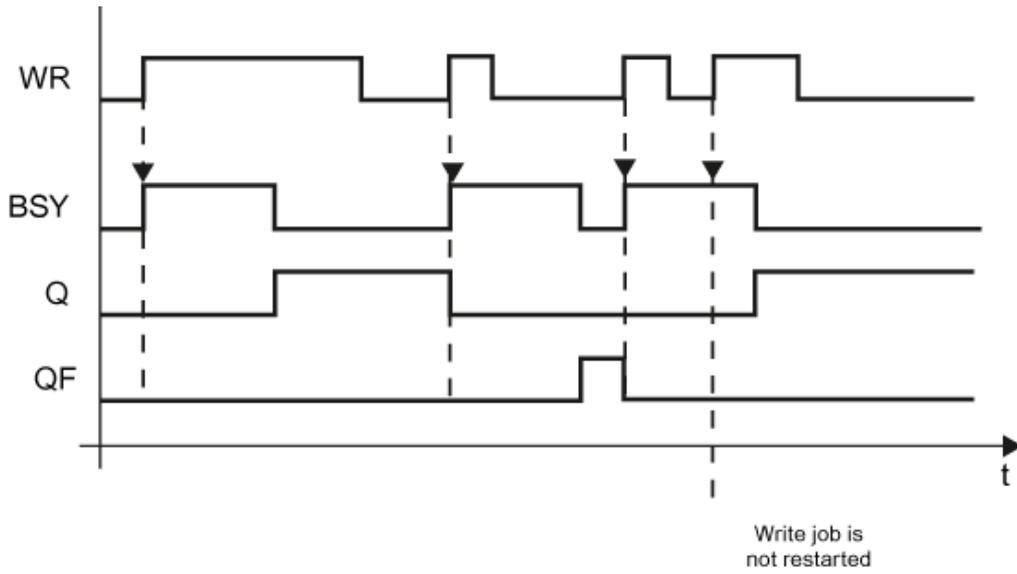
参数值通过输入X指定。可以在输入WR的正沿上启动异步写任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在写任务运行时，忽略输入WR的其他正沿。

Output Q = 1表示参数成功写入。当存取发生错误时，则通过QF = 1来表示。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。 一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	0..2 ¹⁶	
IDX	参数索引	0	0..2 ¹⁶	
WR	启动写任务	0	0/1	
X	参数值	0	UINT	
BSY	执行写任务	0	0/1	
Q	成功完成写任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

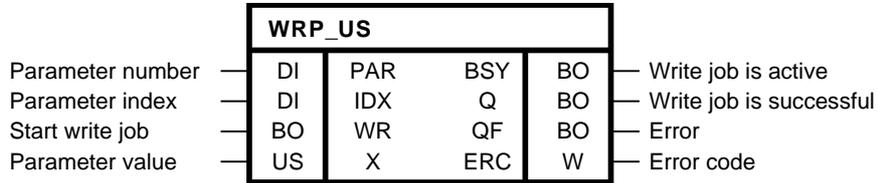
据

可在线加载	是
特别注意点	-

5.30 WRP_US 写入驱动参数 (UNSIGNED SHORT INTEGER型)

 SIMOTION SINAMICS

符号



简要说明

WRP_US (写参数) 在本地驱动对象上启动UNSIGNED SHORT INTEGER型驱动参数的异步写入。

注意

用来描述一个驱动参数的块不能因安全原因而用于修改SINAMICS Safety Integrated 功能的参数。在功能性安全层面 (Safety Integrated) DCC被认为是与安全无关的。

操作模式

待写入参数的编号与索引分别在输入PAR和IDX处表示。当参数未索引时，必须设置IDX = 0。该参数始终在计算功能块功能图的驱动对象上写入。不可以在多个驱动对象上存取参数。

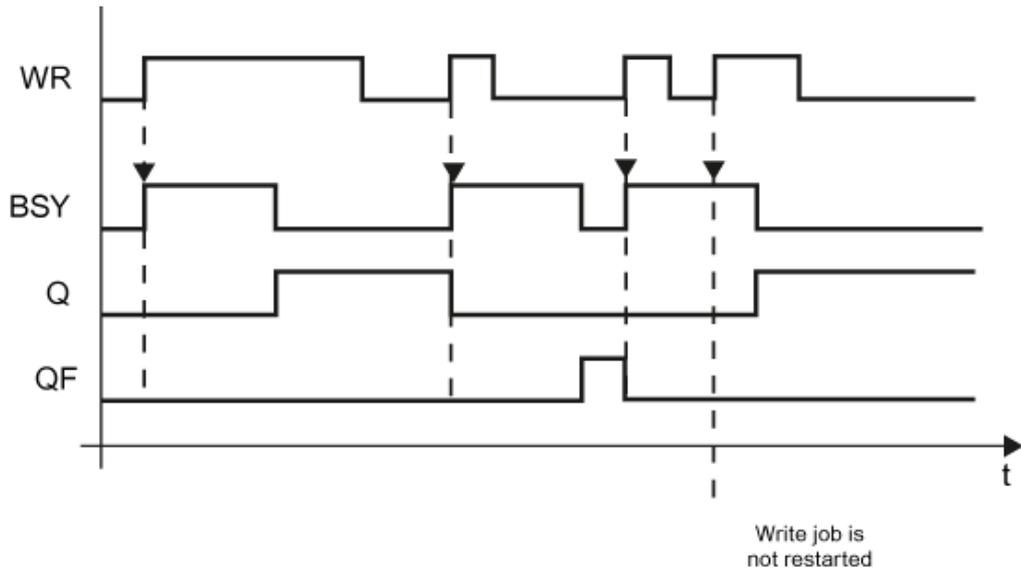
参数值通过输入X指定。可以在输入WR的正沿上启动异步写任务。只要任务启动时，即设置BSY标志位。参数存取的周期数取决于系统利用率，且会随任务而改变。在写任务运行时，忽略输入WR的其他正沿。

Output Q = 1表示参数成功写入。当存取发生错误时，则通过QF = 1来表示。

在错误诊断时，可以评估错误码ERC。ERC会根据PROFIdrive DPV1对应上参数存取的错误码。在本文档的附录A.2中可以找到可能的错误码，还可以在SINAMICS功能手册FH1，PROFIBUS DP / PROFINET IO通讯小节，根据PROFIdrive → 非循环通讯 → 任务及响应的配置进行通讯子小节，DPV1参数响应中的错误值表中找到。

只有当QF = 1时，ERC才有效。

时序图



数量框架

不同块背景的异步任务数可以并行发布。一个块背景只能处理一个任务。

多个块连接

块 连接	说明	预分配	取值范围	数
PAR	参数编号	0	0..2 ¹⁶	
IDX	参数索引	0	0..2 ¹⁶	
WR	启动写任务	0	0/1	
X	参数值	0	USINT	
BSY	执行写任务	0	0/1	
Q	成功完成写任务	0	0/1	
QF	错误	0	0/1	
ERC	错误码	16#0000	WORD	

据

可在线加载	是
特别注意点	-

6.1 DCA 直径计算器

SIMOTION

SINAMICS

符号

DCA					
Line speed [m/min]	R	LS	D	R	Calculated diameter [m]
Motor speed [rpm]	R	MS	LSF	R	Multiplier for setpoint channel [rpm]
Gear ratio	R	GF			
Hold diameter	BO	HLD	RU	BO	Diameter increase is limited
Set diameter	BO	S	RD	BO	Diameter decrease is limited
Set value [m]	R	SV	MAXD	BO	D is limited to DMAX
Invert winding direction	BO	INV	MIND	BO	D is limited to DMIN
Time constant for symmetry of line speed [ms]	TS	T1			
Time constant for smoothing of diameter [ms]	TS	T2			
Tolerance factor for plausibility check	R	TOL			
Material thickness [mm]	R	WTH			
Minimum speed [rpm]	R	MMIN			
Minimum line speed [m/min]	R	LMIN			
Maximum diameter [m]	R	DMAX			
Minimum diameter [m]	R	DMIN			

简要说明

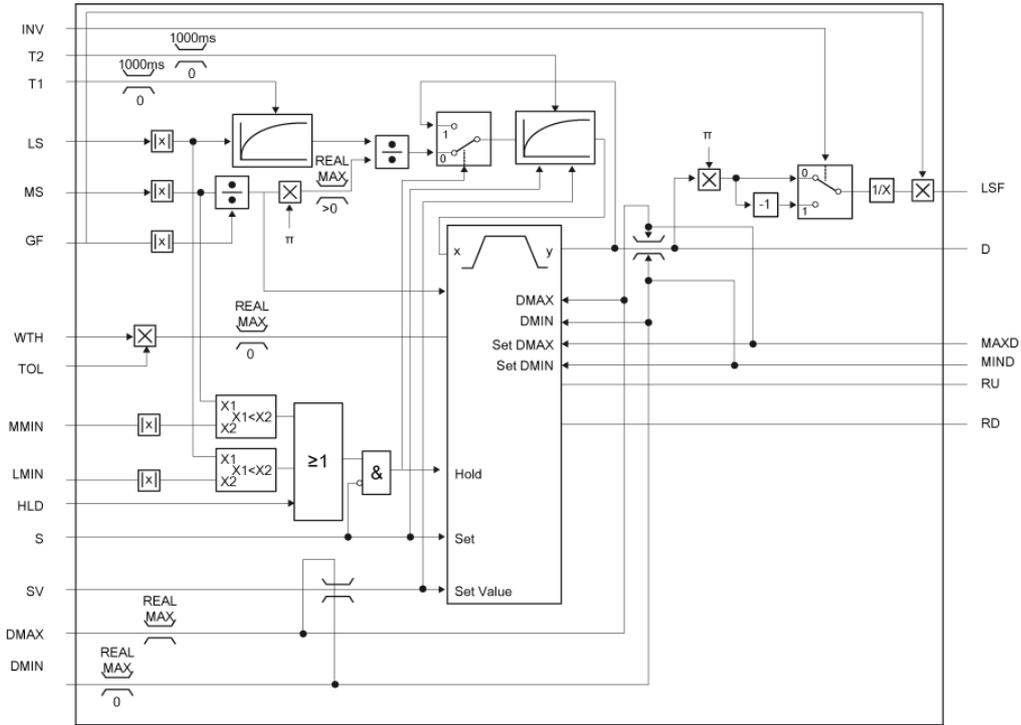
轴机应用的技术功能块 用于根据线速和电机速度确定轴机的当前直径。 计算直径进行真实性检查。

说明

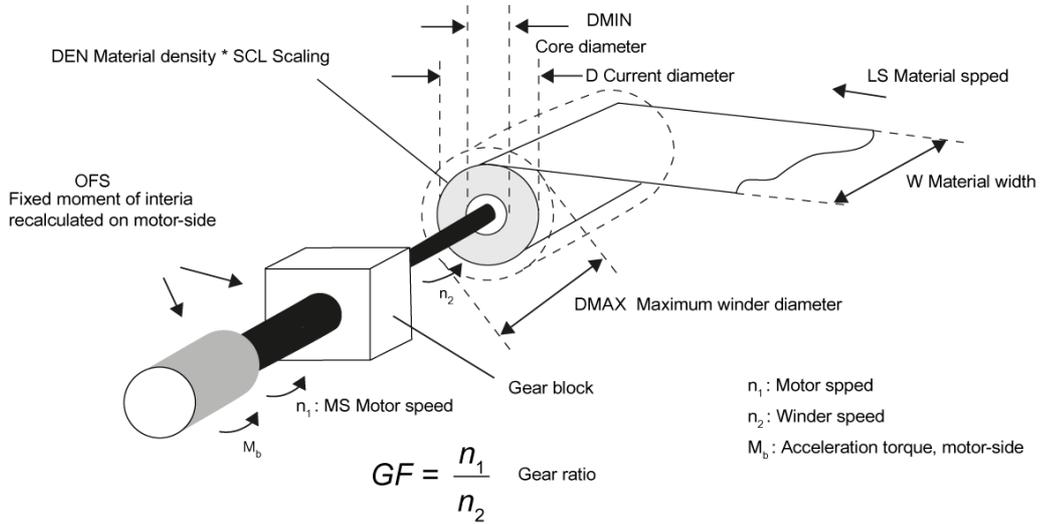
DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

6.1 DCA 直径计算器

方框图



操作模式



根据输入LS和MS提供的当前线速和电机速度，该功能块循环计算轴机的直径。当前电机速度在输入端MS确定。与电机速度相比，线速可以延时T1。

根据以下公式计算当前直径：

$$D = \frac{LS \left[\frac{m}{min} \right] * \bar{u}}{n_l \left[\frac{l}{min} \right] * \pi}$$

使用时间常量T2的平滑滤波元素可以再次使计算结果平滑化。平滑滤波器T1和T2具有PT1的行为特征。当时间常量T1或T2 = 0时，平滑输入值直接写为输出。只有当线速LS或电机速度MS大于各自的阈值LMIN或LMIN时，才会进行直径计算。否则，保持最后的计算直径值。根据保持的值，平滑滤波元素T2转换为衍生直径D。直径D保持也可以通过输入HLD = 1直接激活。输入SV可为该直径赋一个预设值；当S = 1时，该直径适用。同时，平滑滤波元素T2也初始化。只有当S = 0时，才会启动直径D的计算和平滑滤波元素T2。直径设定值优先于直径保持值。

在平滑滤波元素T2之后，计算出的直径会进行合理性检验；如果验证到不合理之处，则对其进行校正。该检测功能等同于单一斜坡函数发生器的检测功能。斜升时间或斜降时间根据材料厚度WTH、公差因数TOL和提升速度WTH进行动态计算。当材料厚度WTH = 0时，合理性检验无效。

每个扫描间隔内，最大直径改变值 ΔD_{max} 根据以下参数确定：

$$\Delta D_{max} = TOL \cdot 2 \cdot \frac{MS}{60 \cdot GF} \cdot \frac{WTH}{1000} \cdot T_A$$

其中：

ΔD_{max}	每个扫描间隔内的最大直径改变值 [m]
TOL	公差因数
MS	电机速度[rpm]
GF	齿轮齿数比
WTH	材料厚度[mm]
T_A	功能块采样时间[s]

直径D的取值范围为：

$$D_n \leq D_{n-1} + \Delta D_{max_n}; D_n (\text{无限制}) \geq D_{n-1} (\text{斜升限制})$$

$$D_n \geq D_{n-1} - \Delta D_{max_n}; D_n (\text{无限制}) \leq D_{n-1} (\text{斜降限制})$$

设置输出RU（斜升限制）或RD（斜降限制）是为了显性表示限制生效。当限制再次撤消时，相应的输出也设为0。当保持值Hold = 1或设定值Set = 1时，两个输出都复位。当设置直径时，斜坡函数发生器失效。合理性检验向下连至限制器。如果当前直径限于DMAX，那么输出MAXD = 1。DMIN的限制在输出MIND处表示。当限制生效时，斜坡函数发生器会根据有效限值进行校正，以避免“大规模积分”（抗积分饱和）。此时，以下公式适用于下一轮斜坡函数发生器周期：

$$D_{n-1} = DMAX_{n-1} (\text{当DMAX存在限制时})$$

$$D_{n-1} = DMIN_{n-1} (\text{当DMIN存在限制时})$$

输出LSF为设定值通道循环提供倍增因数，以根据当前线速计算出电机速度的设定值。通过设置INV = 1可以改变绕线方向。

注意“间接”张力控制的使用

当带材拉断发生时，输入HLD应设置为保持直径。否则，基于当前再次加快或减慢（解线/绕线）的线速和电机速度，DCA功能块会通过直径计算返回一个直径值。因此，轴机将会加速。

6.1 DCA 直径计算器

多个块连接

块 连接	说明	预分配	取值范围	数
LS	线速[m/min]	0.0	0.. REAL MAX	
MS	电机速度[rpm]	1.0	0.. REAL MAX	
GF	齿轮齿数比	1.0	0.. REAL MAX	
HLD	保持直径	0	0/1	
S	给定直径	0	0/1	
SV	设定值[m]	0.0	0.. REAL MAX	
INV	反绕线方向	0	0/1	
T1	用于线速对称的时间常量[毫秒]	0.0	0.. REAL MAX	
T2	用于直径平滑滤波的时间常量[毫秒]	0.0	0.. REAL MAX	
TOL	用于合理性检验的公差因数	1,5	0.. REAL MAX	
WTH	材料厚度[mm]	0.0	0.. REAL MAX	
MMIN	最低速度[rpm]	1.0	0.. REAL MAX	
LMIN	最小线速[m/min]	0.1	0.. REAL MAX	
DMAX	最大直径[m]	0.1	0.. REAL MAX	
DMIN	最小直径[m]	0.01	0.. REAL MAX	
D	计算直径[m]	0.0	0.. REAL MAX	
LSF	设定值通道的乘法器	1.0	0.. REAL MAX	
RU	直径增加是受限的。	0	0/1	
RD	直径减少是受限的。	0	0/1	
MAXD	D受限于DMAX	0	0/1	
MIND	D受限于DMIN	0	0/1	

据

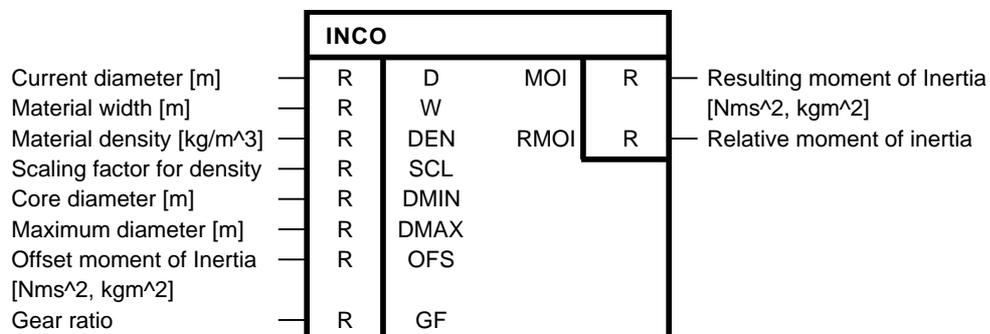
可在线加载	是
特别注意点	-

6.2 INCO 轴向轴机转动惯量

☑ SIMOTION

☑ SINAMICS

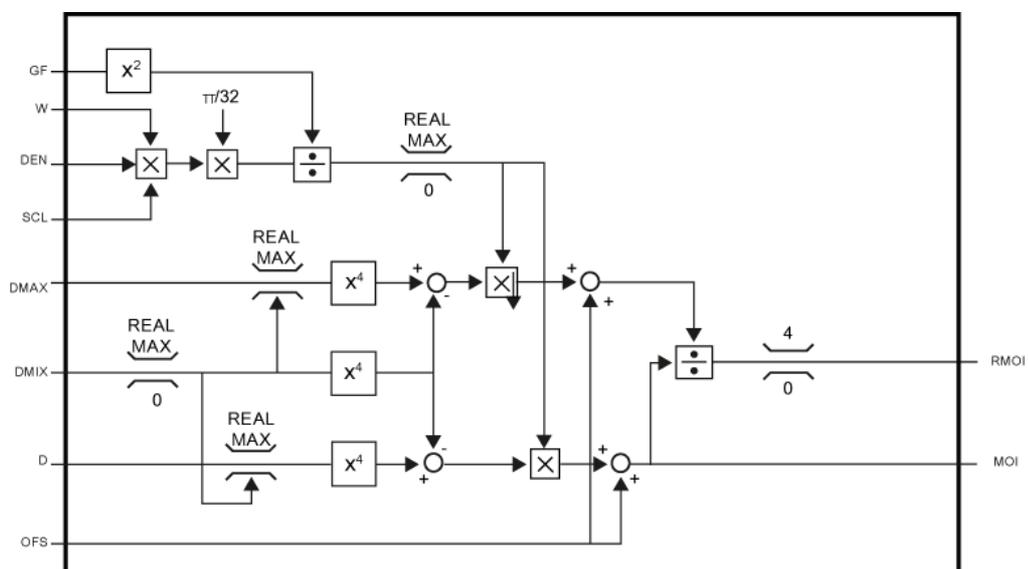
符号



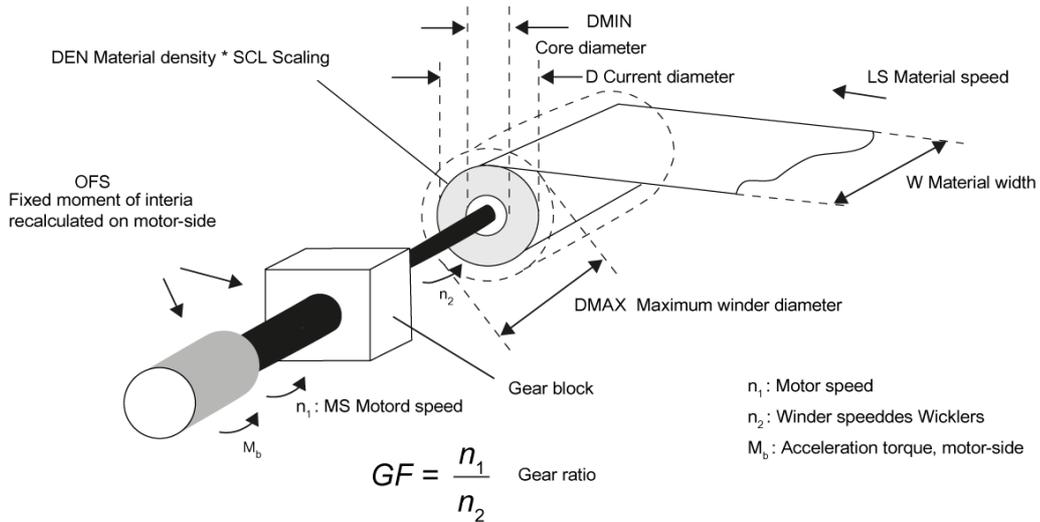
简要说明

轴机应用的技术功能块 用于确定进行转矩前馈控制的轴机的转动惯量。

方框图



操作模式



该功能块计算轴向轴机电机侧的转动惯量。输入变量D指定线圈的当前直径[m]。可以通过DEN和针对通过SCL指定密度的校正因子来明确被绕材料的密度[kg/ m3]。输入变量DMIN [m]用于明确线圈芯的直径或最小直径。为了能够计算调速器Kp适应的相对转动惯量RMOI，功能块需要达到整体的最大转动惯量。为了计算此相对转动惯量，必须在输入DMAX [m]处明确最大绕线直径。通过输入OFS [Nms², kgm²]处明确电机侧的全部静态转动惯量，包括电机、空载轴机和有可能需要的变速箱。齿轮齿数比在输入GF处明确。在输出MOI处明确整体轴机在电机侧的当前转动惯量。

多个块连接

块 连接	说明	预分配	取值范围	数
D	当前直径[m]	0.0	0.. REAL MAX	
W	材料宽度[m]	0.0	0.. REAL MAX	
DEN	材料密度[kg/m ³]	0.0	0.. REAL MAX	
SCL	密度比例因子	1.0	0.. REAL MAX	
DMIN	圈心直径[m]	0.01	0.. REAL MAX	
DMAX	最大直径[m]	0.1	0.. REAL MAX	
OFS	偏移的转动惯量[Nms ² , kgm ²]	0.0	0.. REAL MAX	
GF	齿轮齿数比	1.0	0.. REAL MAX	
MOI	最终转动惯量[Nms ² , kgm ²]	0.0	0.. REAL MAX	
RMOI	相对转动惯量	0.0	0.. REAL MAX	

据

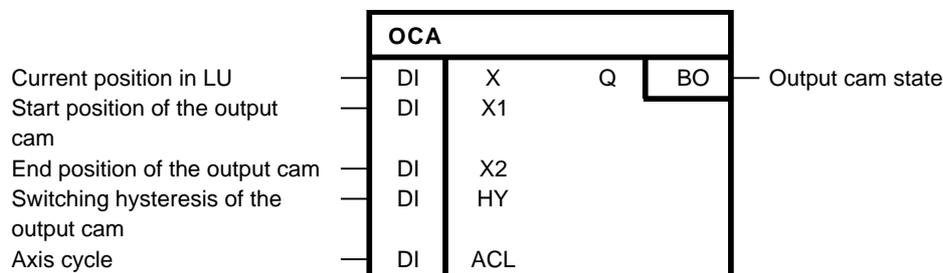
可在线加载	是
特别注意点	-

6.3 OCA 软件凸轮控制器

☑ SIMOTION

☑ SINAMICS

符号

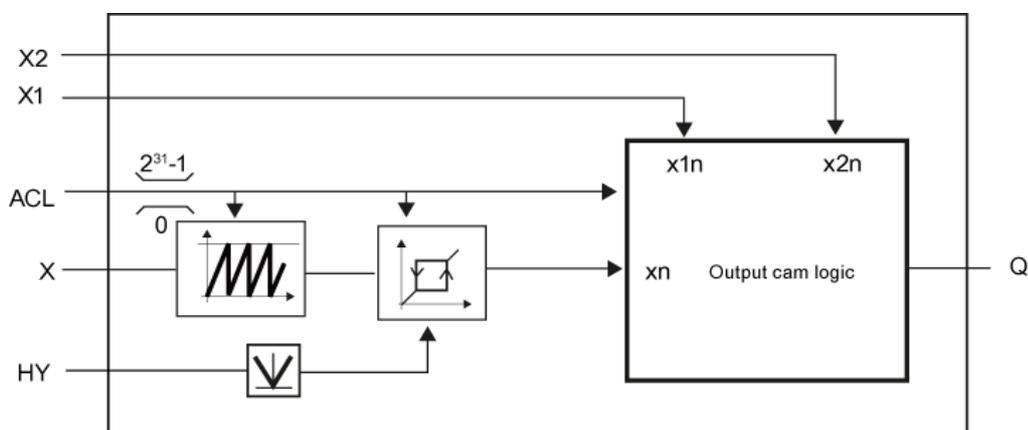


简要说明

软件凸轮控制器支持以下特性：

- 基于位置的凸轮
- 可动态改变打开/关闭位置。
- 实际值相关的输出凸轮的可变磁滞

方框图



操作模式

输入X1[LU]指定正向运动中基于位置的凸轮打开位置和负向运动中的关闭位置。X2[LU]指定正向运动中的关闭位置或负向运动中的打开位置

在输入ACL处可以指定轴周期，从而使凸轮控制器可以与模态轴协同工作。当ACL = 0时，不会进行模态校正。

输入X的磁滞带可以通过HY设置，以确保在实际值相关的输出凸轮处于静止时，没有进行转换操作。

6.3 OCA 软件凸轮控制器

凸轮逻辑进行以下评估:

非模态轴 (ACL = 0)

$x1n < x2n$	$Q = (x1n \leq xn) \text{ AND } (x2n > xn)$
$x1n \geq x2n$	$Q = 0$

非模态轴 (ACL \neq 0) :

$x1n < x2n$	$Q = (x1n \leq xn) \text{ AND } (x2n > xn)$
$x1n > x2n$	$Q = (x1n \leq xn) \text{ OR } (x2n > xn)$
$x1n \geq x2n$	$Q = 0$

多个块连接

块 连接	说明	预分配	取值范围	数
X	LU处的当前位置	0	DINT	
X1	输出凸轮的开始位置	0	DINT	
X2	输出凸轮的结束位置	0	DINT	
HY	输出凸轮的转换磁滞	0	DINT	
ACL	轴周期	0	$0 \dots 2^{31}-1$	
Q	输出凸轮状态	0	0/1	

据

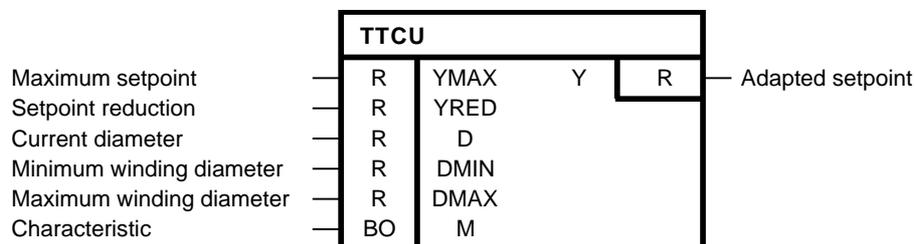
可在线加载	是
特别注意点	-

6.4 TTCU 绕组特性

☑ SIMOTION

☑ SINAMICS

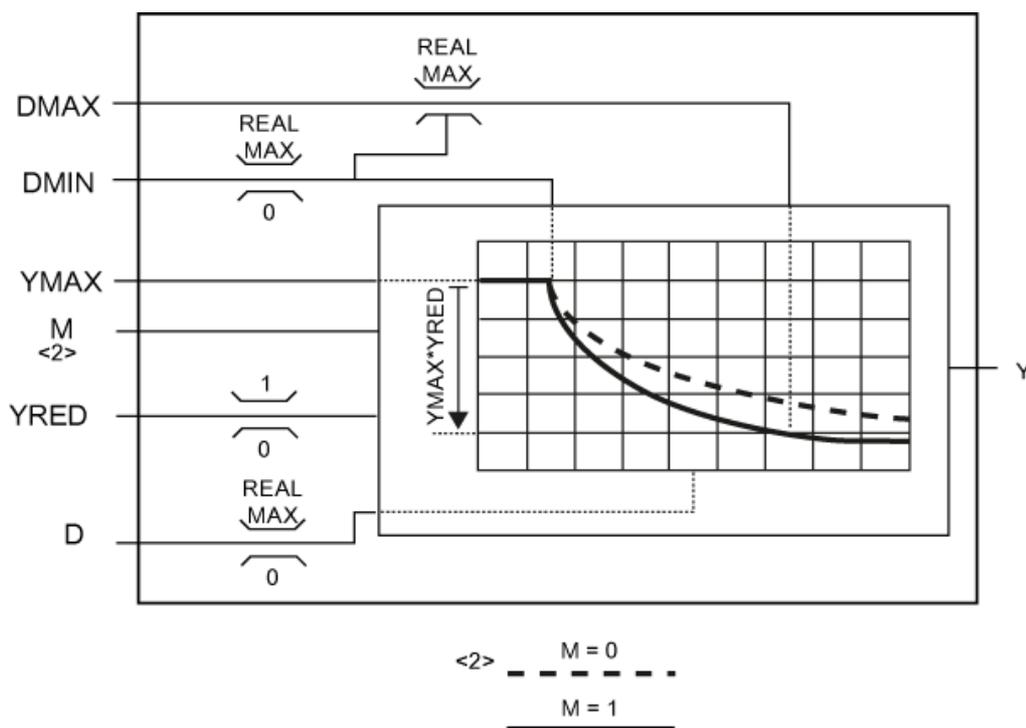
符号



简要说明

根据指定特性值进行设定值调整。用于使轴机应用根据当前轴机直径确定张力设定值的功能块。

方框图



操作模式

当 $D > DMIN$ 为真时，特性值开始减小。输入变量YRED指定其相对于输入变量YMAX的减小程度。输入M通过上升输入变量指定输出变量的减小情况，因而特性值可以预选输入M。当预选 $M = 0$ 时，特性值随因子 $YMAX * YRED$ 渐近地减小。此时，不考虑输入变量DMAX。当预选 $M = 1$ 时，输入变量DMAX可用于指定特性值从YMAX变为 $YMAX * YRED$ 时的输入变量 $D = DMAX$ 。

特性值的计算如下：

$D \leq DMIN$ 为真

$$Y = YMAX$$

$D > DMIN$ 且 $M = 0$ （达到 $D \rightarrow \infty$ 的减小系数）

$$Y = YMAX \left(1 - YRED \left(1 - \frac{DMIN}{D} \right) \right)$$

$D > DMIN$ 且 $M = 1$ （达到 $D = DMAX$ 的减小系数）

$$DMAX > DMIN: Y = YMAX \left(1 - YRED \frac{DMAX}{DMAX - DMIN} \left(1 - \frac{DMIN}{D} \right) \right)$$

$$DMAX = DMIN: Y = YMAX(1 - YRED)$$

多个块连接

块 连接	说明	预分配	取值范围	数
YMAX	最大设定值	0.0	0..REAL MAX	
YRED	设定值减小	0.0	0..1	
D	当前直径	0.0	0..REAL MAX	
DMIN	最小绕组直径	1.0e-2	0..REAL MAX	
DMAX	最大绕组直径	0.1	0..REAL MAX	
M	曲线	1	0/1	
Y	适配的设定值	0.0	0..REAL MAX	

据

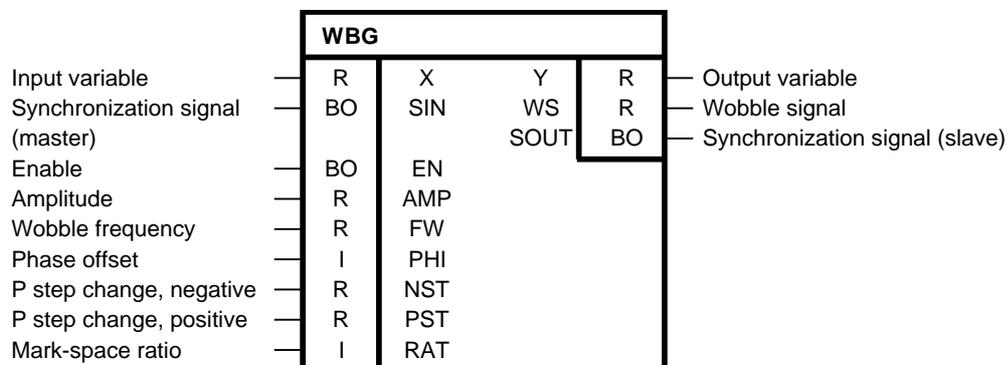
可在线加载	是
特别注意点	-

6.5 WBG 抖动发生器

☑ SIMOTION

☑ SINAMICS

符号



简要说明

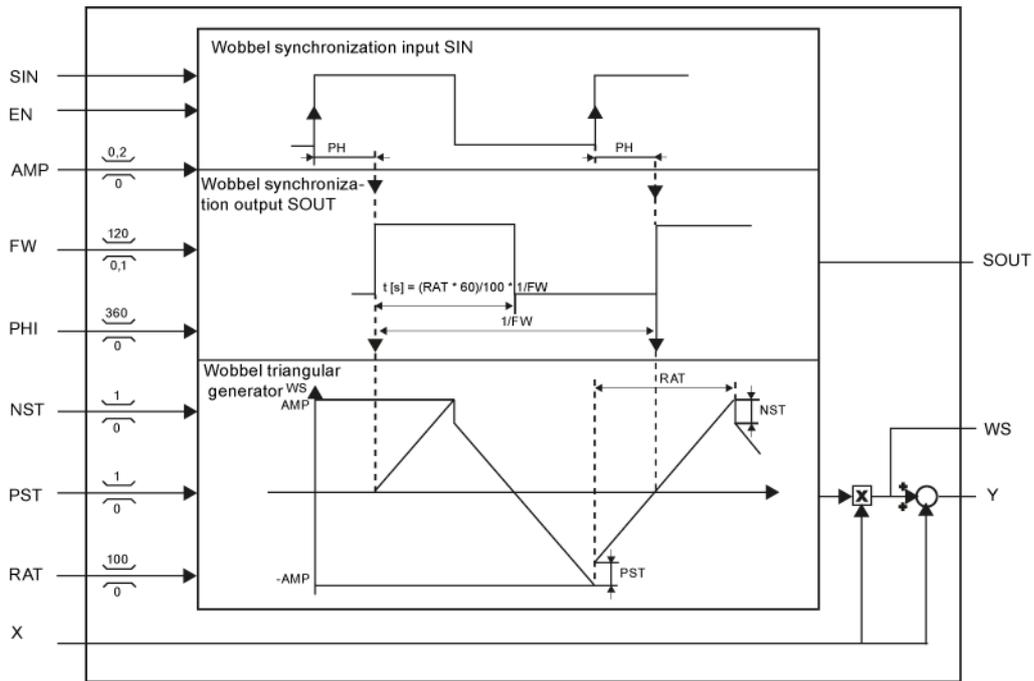
支持可变频率和振幅的三角形发生器，用于在排线驱动上为缠绕纺线标记“故障”。该功能块支持以下特性

- 正负P阶跃变化量可以单独调整。
- 随可变周相移动同步至主驱动。
- 使能输入

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块，那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

方框图



操作模式

抖动发生器在EN = 1时启动，触发动抖信号（WS）的输出和同步信号SOUT。信号生成始终开始于正零交点或同步输出SOUT的正沿。当EN再复位时，抖动发生器会继续工作至WS的下一个零交点；此后发生器停止工作，且SOUT = 0。输入PHI（0-360°）可用于设置同步输入SIN的正沿与抖动信号开始点间的周相移动。接着生成相应信号周期的信号。对于连续信号生成，SIN必须周期性用作触发器。当在新的开始点依然进行前一个信号周期的信号生成时，发生器则撤消。在PHI = 360这一下特殊情况下，会激活自由抖动。然后周期性运行信号生成，并与同步输入SIN解耦。抖动信号切换为输入X和输出Y的输出。

抖动信号属性

输入	取值范围	描述
AMP	0..0.2	抖动信号的相对振幅
FW	0.1..120 rpm	抖动信号频率
PHI	0..360°	抖动信号相对于同步输入SIN处正沿的周相移动
NST	0.0..1.0	相对量，信号正沿末端抖动信号的负阶跃变化量。
PST	0.0..1.0	相对量，信号负沿末端抖动信号的正阶跃变化量。
RAT	0..100%	信号上升沿比率/信号周期

有效振幅(WS) = ABS(X) * AMP

有效负阶跃变化量= -ABS(X) * AMP * NST

有效正阶跃变化量= ABS(X) * AMP * PST

上升沿/下降沿比率= RAT/(100-RAT)

当抖动信号属性动态改变时，改变的属性在新信号周期（正零交点）的开始时生效。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
SIN	同步信号（主站）	0	0/1	
EN	使能	0	0/1	
AMP	振幅	0.0	0..0.2	
FW	抖动频率	60	0.1..120	
PHI	相位偏移	360	0..360	
NST	P阶跃变化量，负值	0.0	0.0..1.0	
PST	P阶跃变化量，正值	0.0	0.0..1.0	
RAT	脉冲间隔比	50	0..100	
Y	输出变量	0.0	REAL	
WS	抖动信号	0.0	REAL	
SOUT	同步信号（从站）	0	0/1	

据

可在线加载	是
特别注意点	-

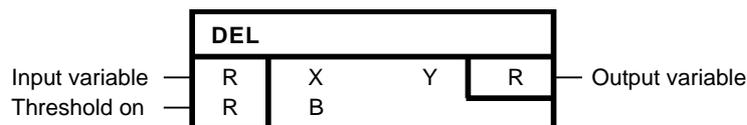
Closed-loop control

7.1 DEL 死区元素

SIMOTION

SINAMICS

符号



简要说明

- 可变死带
- 设置零点对称值范围为0。

操作模式

- 当X的绝对值小于B时， $Y = 0$ 。
- 当X大于等于B时， $Y = X - B$ 。
- 当X小于等于-B时， $Y = X + B$ 。

零点对称死带可以通过运算值B进行设置。

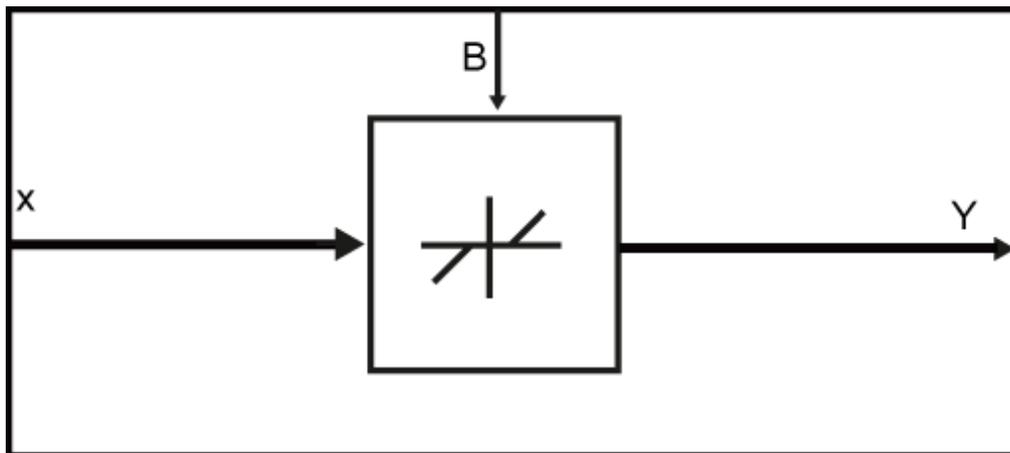
算法：

限制条件 $B \geq 0$

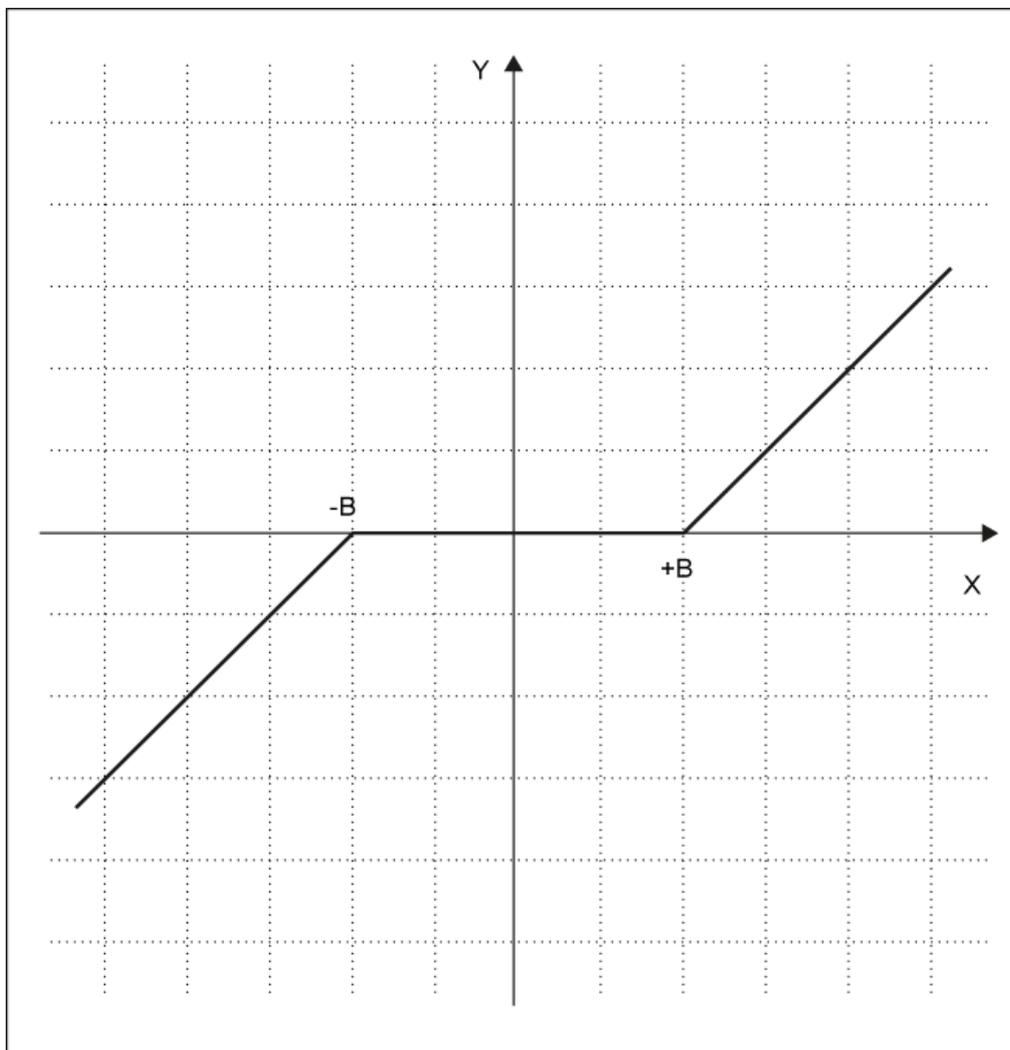
当 $B < 0$ 时，以下公式适用于所有X： $Y = X$ 。

$$Y = \begin{cases} X + B & \text{for } X \leq -B \\ 0 & \text{for } -B < X < B \\ X - B & \text{for } X \geq B \end{cases}$$

方框图



XY图



多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
B	阈值	0.0	REAL	
Y	输出变量	0.0	REAL	

据

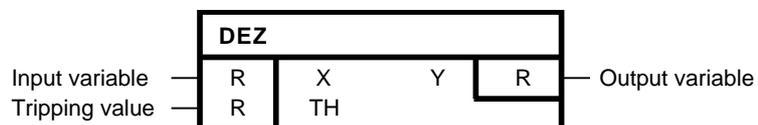
可在线加载	是
特别注意点	-

7.2 DEZ 死区元素

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 可变死区
- 设置零点对称值范围为0

操作模式

当X的绝对值小于TH时， $Y = 0$ 。

当X的绝对值大于等于TH时， $Y = X$ 。

零点对称死区可以通过运算值TH进行设置。

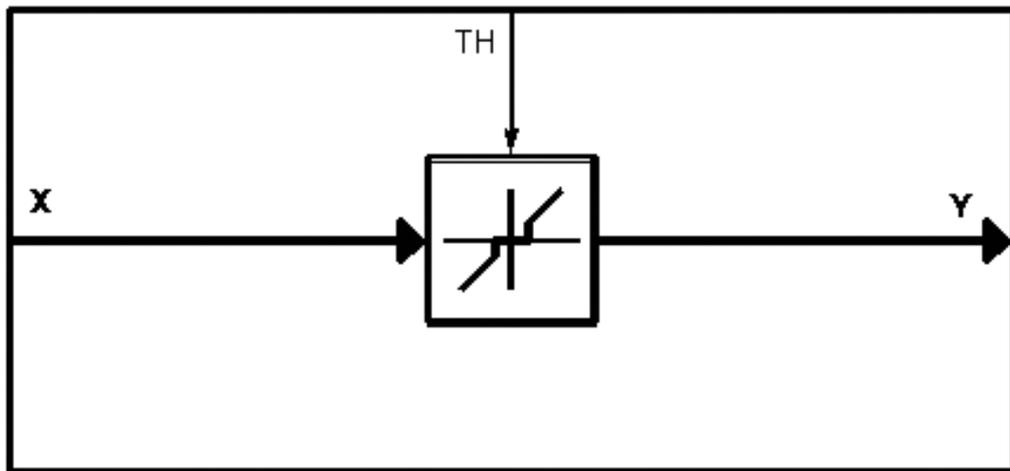
算法：

$$Y = \begin{cases} X & \text{for } X \leq -TH \\ 0 & \text{for } -TH < X < TH \\ X & \text{for } X \geq TH \end{cases}$$

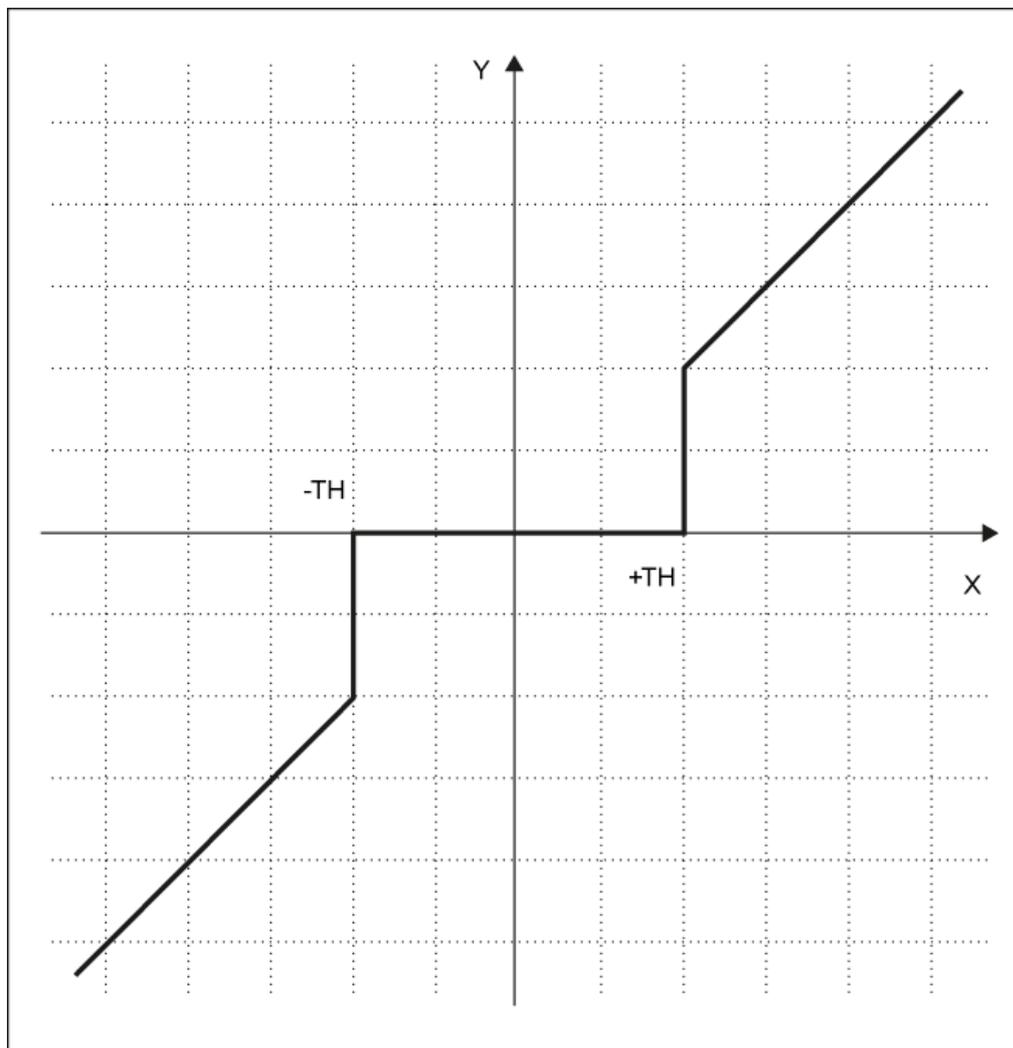
限制条件 $TH \geq 0$

当 $TH < 0$ 时，以下公式适用于所有X： $Y = X$ 。

方框图



XY图



多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
TH	跳闸值	0.0	REAL	
Y	输出变量	0.0	REAL	

据

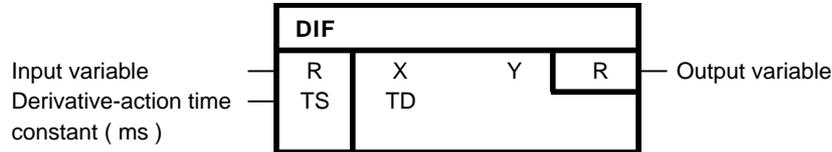
可在线加载	是
特别注意点	-

7.3 DIF 微商作用元素

SIMOTION

SINAMICS

符号



简要说明

- 支持微商作用响应的功能块

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

- 输出变量Y与输入变量X变化速度和微商作用时间常量TD的乘积成正比。
- 根据该算法计算离散值。

算法:

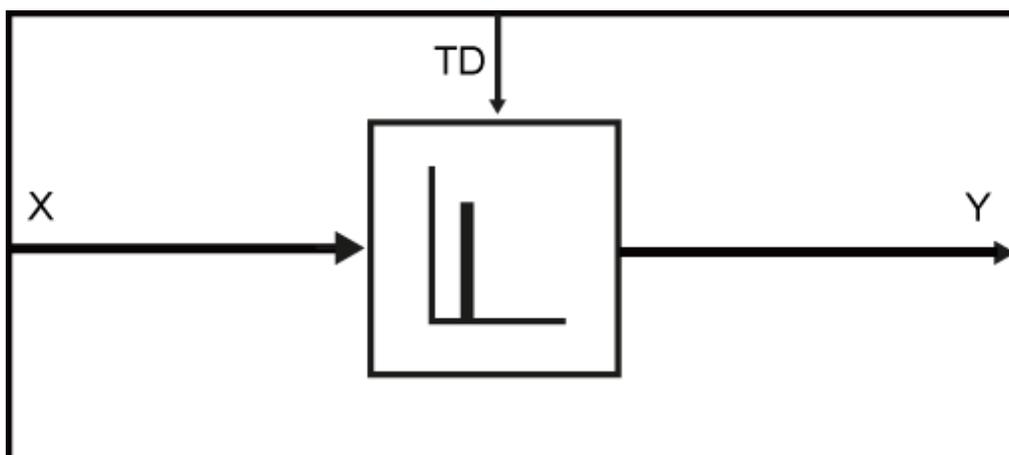
$$Y_n = (X_n - X_{n-1}) \cdot \frac{TD}{TA}$$

Y _n	扫描间隔n内的Y值
X _n	扫描间隔n内的X值
X _{n-1}	扫描间隔n-1内的X值

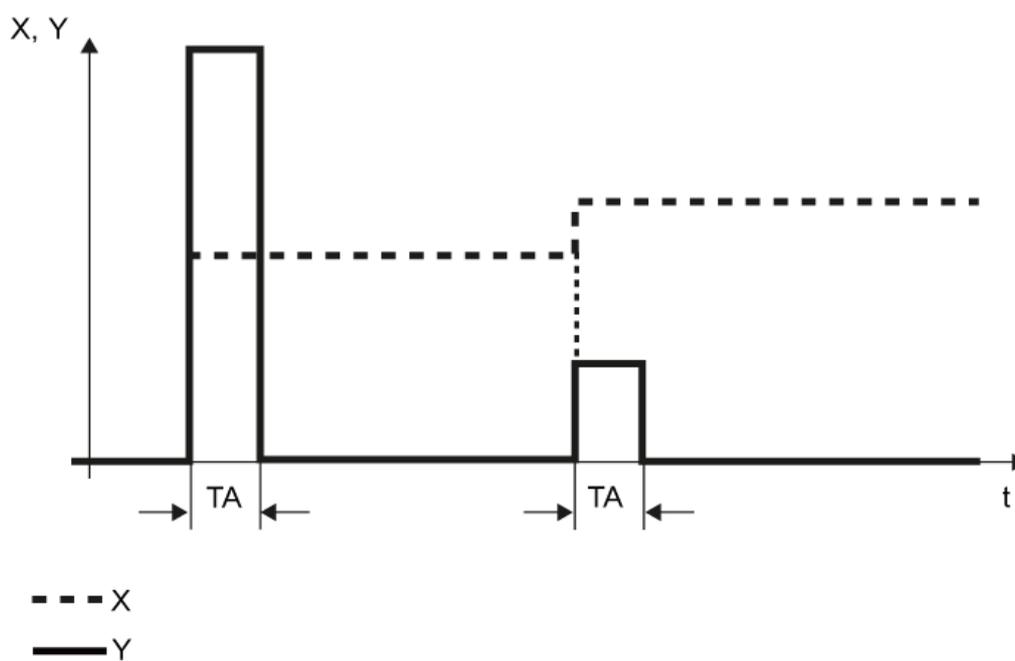
说明

当在X处的取值变化时, Y值随TD/TA的增加而增加。TA即是配置功能块时的采样时间。TD的内部取值范围为TD >= 0。警告: 可能超载!

方框图



传递函数



多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
TD	微商作用时间常量 (毫秒)	0	SDTIME	
Y	输出变量	0.0	REAL	

据

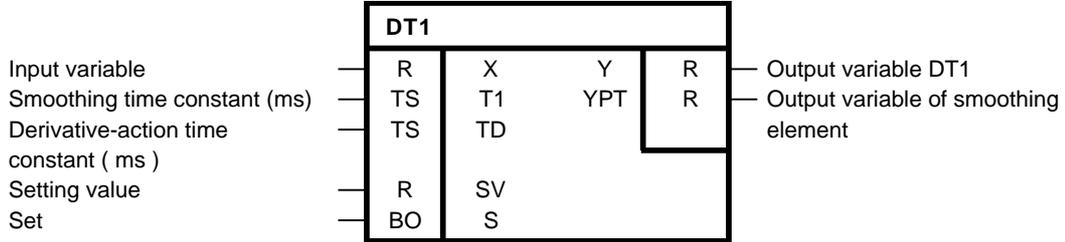
可在线加载	是
特别注意点	-

7.4 DT1 平滑滤波元素

SIMOTION

SINAMICS

符号



简要说明

- 支持微商作用响应和平滑滤波的功能块 可以设置DT1输出。

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

设定函数未激活 (S = 0)

输入变量X因平滑滤波时间常量T1而动态延迟, 并将其取值赋给微商作用元素和功能块输出YPT。

整个DT1元素的输出变量Y与YPT (微商) 变化速度和微商作用时间常量TD的乘积成正比。

T1决定输出变量值的下降陡度。在经过平滑滤波和微分后, T1确定传递函数下降到X TD/T1的37%的时间。当T1/TA足够大时, 传递函数符合以下公式:

$$Y(t) = X \cdot (TD/T1) \cdot e^{-t/T1}$$

其中, $t = n \cdot TA$

算法:

$$Y_n = \frac{TD}{T1} \cdot (X_n - YPT_{n-1})$$

$$YPT_n = YPT_{n-1} + \frac{TA}{T1} \cdot (X_n - YPT_{n-1})$$

YPTn	扫描间隔n内的YPT值
Yn	扫描间隔n内的Y值
Xn	扫描间隔n内的X值
YPTn-1	扫描间隔n-1内的YPT值

T1/TA越大，Y和YPT从一个采样时间到下一个时间的振幅差就越小。TA是配置功能块的采样时间。TD/TA越大，Y从一个采样时间到下一个时间的振幅差就越大。TD和T1内部受限：TD ≥ 0，T1 ≥ TA。

设定函数激活 (S = 1)

设定函数激活时，设定值SV适用于dt1输出Y (Y=SV)。以下结果为针对平滑滤波元素的输出：

$$YPT_n = X_n - \frac{T1}{Td} \cdot SV_n$$

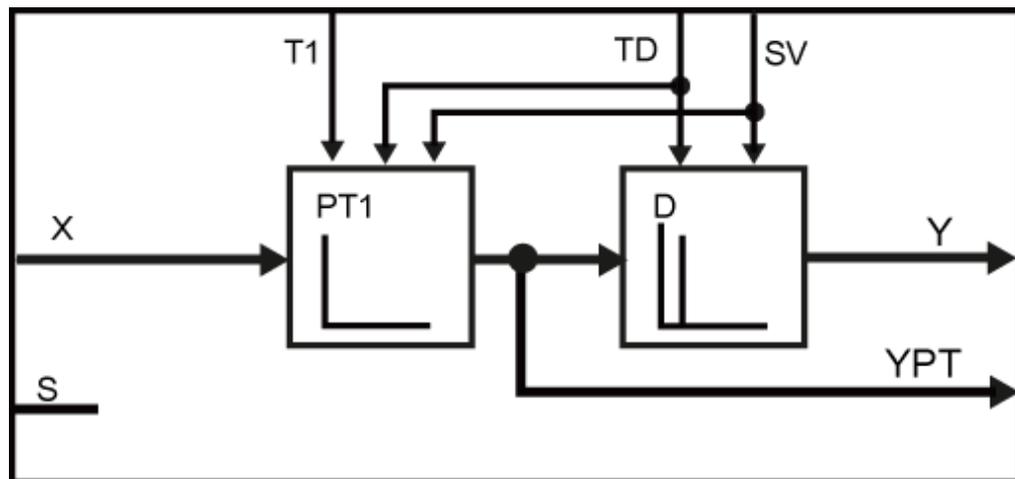
当TD ≠ 0

T1和TD的内部限制适用于此种情况。当TD=0时，只要S=1，输出变量就保持不变。

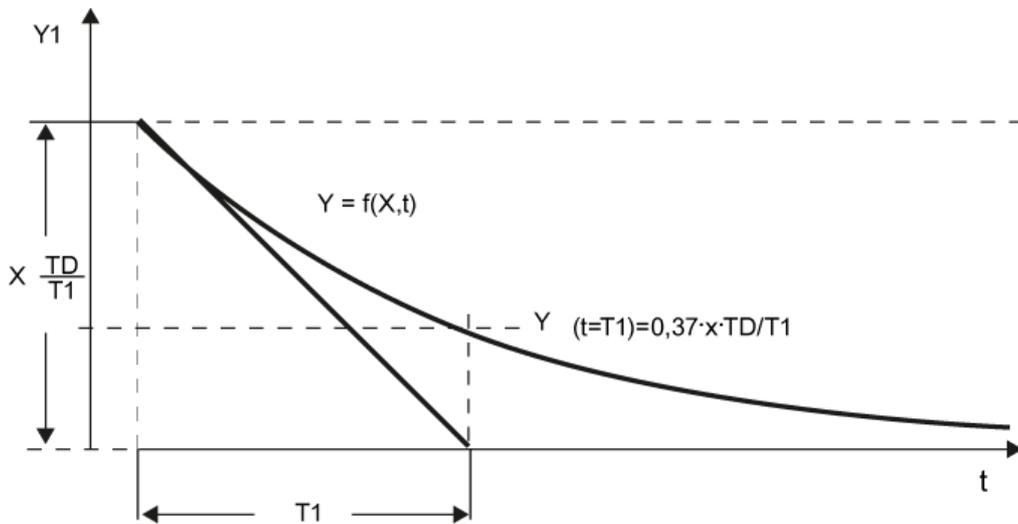
警告

激活和未激活的设定函数都存在超载可能。

方框图



传递函数



初始化

当输入S的值在初始化时为逻辑1时，设定值SV适用于输出Y和 $YPT = T1/TD * (X - SV)$ 设定。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
T1	平滑滤波时间常量（毫秒）	0.0	SDTIME	
TD	微商作用时间常量（毫秒）	0.0	SDTIME	
SV	设定值	0.0	REAL	
S	设定	0	BOOL	
Y	输出变量DT1	0.0	REAL	
YPT	平滑滤波元素的输出变量	0.0	REAL	

据

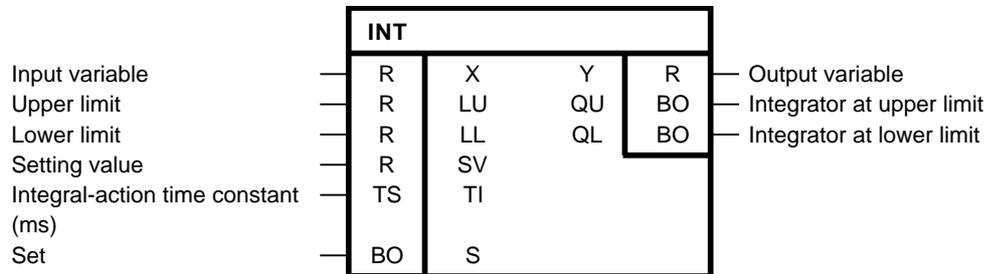
可在线加载	是
特别注意点	-

7.5 INT 积分器

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持积分行为的功能块
- 积分器功能：
 - 设置初始值
 - 可变积分作用时间常量
 - 可变限制
 - 对于正常积分状态，LU的正限值必须明确，LL的负限值也必须明确。

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块，那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

输出变量Y的改变量与输入变量X成正比，而与积分作用时间常量TI成反比。

积分器的输出Y可以通过输入LU和LL来确定取值范围。当输出达到其中一个限值时，就会通过输出QU或QL来发送一个消息。当LL >= LU时，输出Y = LU。

根据以下算法来计算离散值（TA是配置功能块的采样时间）：

算法：

$$Y_n = Y_{n-1} + \frac{TA}{TI} \cdot X_n$$

Y _n	扫描间隔n内的Y值
Y _{n-1}	扫描间隔n-1内的Y值
X _n	扫描间隔n内的X值

7.5 INT 积分器

当S = 1时，输出变量Y设置为设定值SV。 通过S可以实现两个功能：

轨迹积分器 (Y = SV)

二进制输入为S = 1，且设定值SV会发生改变。 如果适用，调整操作之后输出会立刻跳变为设定值。

设置积分器为初始值SV

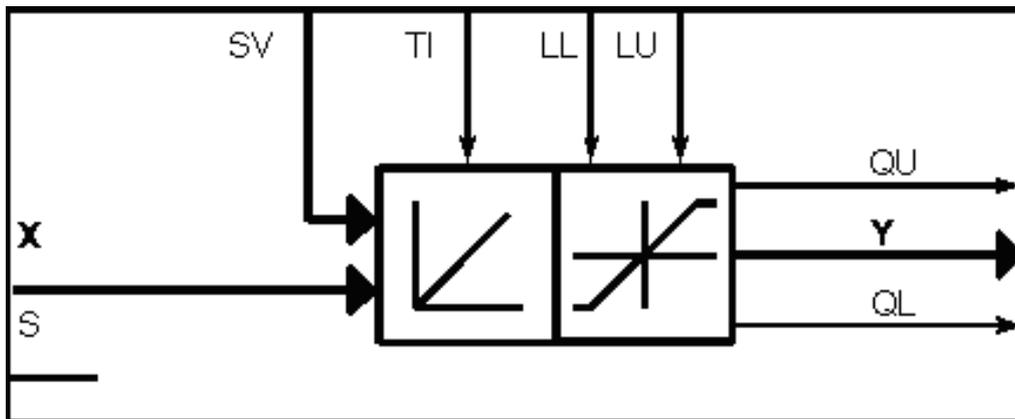
S转换为1。然后S设置为0，积分器从SV开始，沿着输入变量X的极性所指定的方向运算。

说明

必须保证取样时间与积分作用时间常量TI相比，足够得短。

TI内部受限。 $TI \geq TA$ 。Y端的振幅改变量取决于限制生效的周期时间。

方框图



真值表

S	条件	Y	QU	QL	模式
0	$LL < Y_{n-1} + X \times TA / TI < LU$	Y_n	0	0	积分
0	$Y_{n-1} + X \times TA / TI \geq LU$	LU	1	0	积分器位于上限值
0	$Y_{n-1} + X \times TA / TI \leq LL$	LL	0	1	积分器位于下限值
1	$LL < SV < LU$	SV_n	0	0	设定
1	$SV \geq LU$	LU	1	0	积分器位于上限值
1	$SV \leq LL$	LL	0	1	积分器位于下限值

LL >= LU时的真值表

S	条件	Y	QU	QL	模式
(任意值)	$LL \geq LU$	LU	1	1	积分器位于上限值

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
LU	上限	0.0	REAL	

块 连接	说明	预分配	取值范围	数
LL	下限	0.0	REAL	
SV	设定值	0.0	REAL	
TI	积分作用时间常量（毫秒）	0.0	SDTIME	
S	设定	0	0/1	
Y	输出变量	0.0	REAL	
QU	取上限值的积分器	0	0/1	
QL	取下限值的积分器	0	0/1	

据

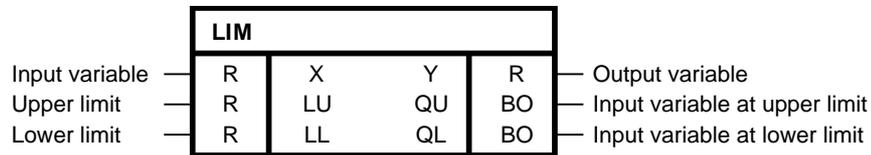
可在线加载	是
特别注意点	-

7.6 LIM 限制器 (REAL型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 用于限制的功能块
- 可变上下限
- 达到给定限值时的指示

操作模式

该功能块将输入变量X转移给输出Y。在此过程中，输入变量的限值取决于LU和LL。

当输入变量达到上限值LU时，设置输出QU = 1。

当输入变量达到下限值LL时，设置输出QL = 1。

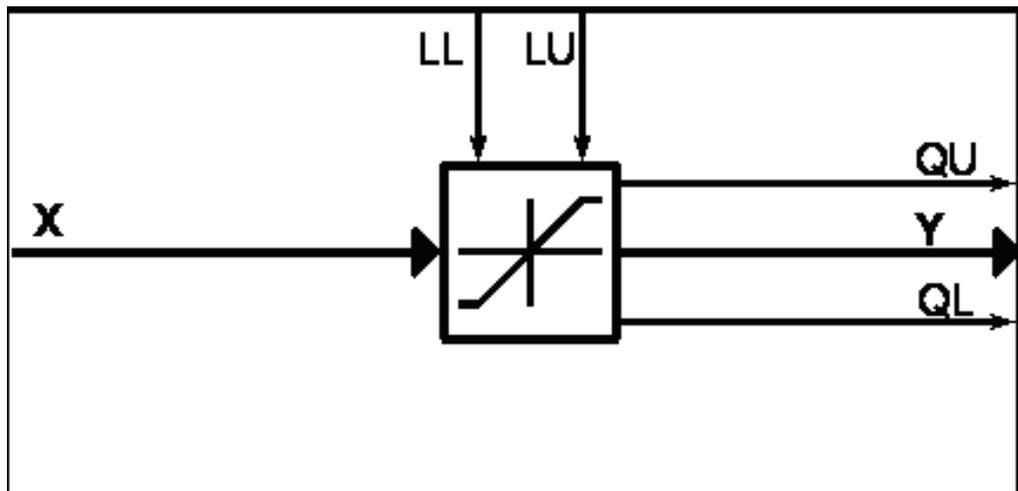
当下限值大于等于上限值时，设置输出Y为上限值LU。

算法：

$$Y = \begin{cases} LU & \text{for } X \geq LU \\ X & \text{for } LL < X < LU \\ LL & \text{for } X \leq LL \end{cases}$$

限制条件：LL < LU

方框图



真值表

条件	Y	QU	QL	模式
$LL < X < LU$	X	0	0	
$X \geq LU$	LU	1	0	上限处的输入变量
$X \leq LL$	LL	0	1	下限处的输入变量

LL >= LU时的真值表

条件	Y	QU	QL	模式
$LL \geq LU$	LU	1	1	上限处的输入变量

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
LU	上限	0.0	REAL	
LL	下限	0.0	REAL	
Y	输出变量	0.0	REAL	
QU	上限处的输入变量	1	0/1	
QL	下限处的输入变量	1	0/1	

据

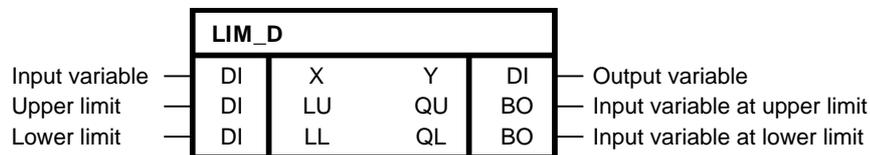
可在线加载	是
特别注意点	-

7.7 LIM_D 限制器 (DOUBLE INTEGER型)

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 用于DOUBLE INTEGER型限制的功能块
- 可变上下限
- 达到给定限值时的指示

操作模式

该功能块将输入变量X转移给输出Y。在此过程中，输入变量的限值取决于LU和LL。

当输入变量达到上限值LU时，设置输出QU = 1。

当输入变量达到下限值LL时，设置输出QL = 1。

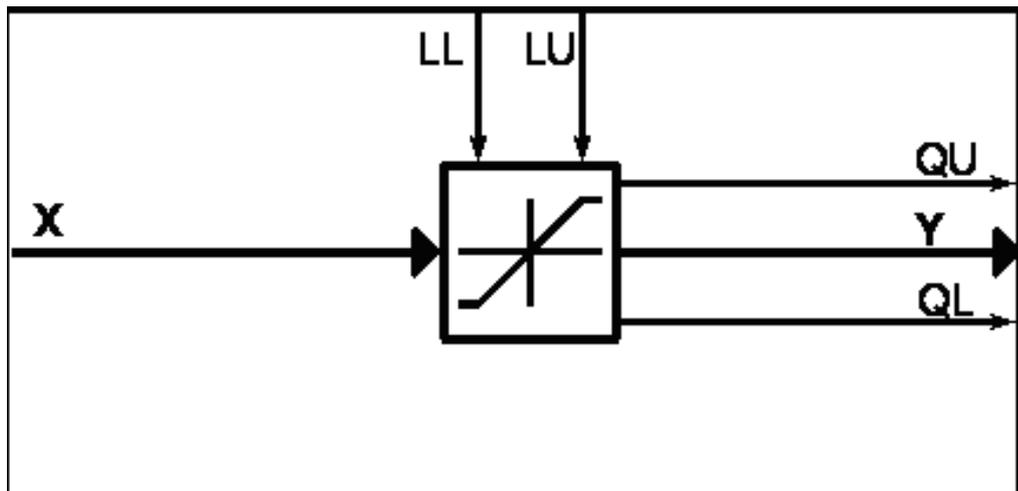
当下限值大于等于上限值时，设置输出Y为上限值LU。

算法：

$$Y = \begin{cases} LU & \text{for } X \geq LU \\ X & \text{for } LL < X < LU \\ LL & \text{for } X \leq LL \end{cases}$$

限制条件：LL < LU

方框图



真值表

条件	Y	QU	QL	模式
$LL < X < LU$	X	0	0	
$X \geq LU$	LU	1	0	上限处的输入变量
$X \leq LL$	LL	0	1	下限处的输入变量

LL >= LU时的真值表

条件	Y	QU	QL	模式
$LL \geq LU$	LU	1	1	上限处的输入变量

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0	DINT	
LU	上限	0	DINT	
LL	下限	0	DINT	
Y	输出变量	0	DINT	
QU	上限处的输入变量	1	0/1	
QL	下限处的输入变量	1	0/1	

据

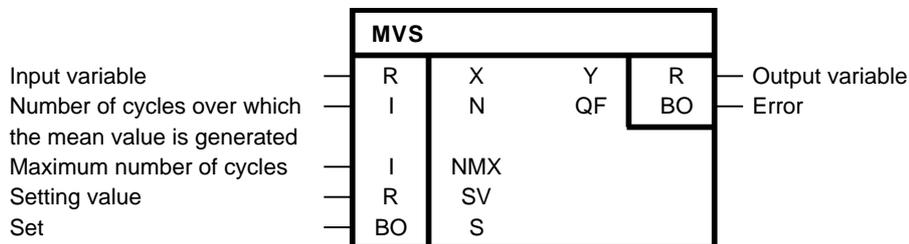
可在线加载	是
特别注意点	-

7.8 MVS 滑动平均值发生器

SIMOTION

SINAMICS

符号



简要说明

该功能块通过输入变量X计算滑动平均值。

操作模式

平均值在最后N个周期内生成。

$$Y_k = \frac{1}{N} \cdot \sum_{i=k-(N-1)}^k X_i$$

X_k = 周期K内的X

$k = 0$ 表示当前的周期。

周期数的可变范围为 $1 \leq N \leq NMX$ 。最大周期数由NMX确定，且在运算过程中不可改变。该功能块对输入N的限制范围为 $1 \leq N \leq NMX$ 。输入值的缓冲区一直取值NMAX，与N无关。这样的话，当窗口长度改变时，功能块可以通过所有变量来重新确定当前的平均值。

当 $S = 1$ 时，平均值设置为设定值SV。

初始化

初始化时，NMX用于指定X值的最大平均值缓冲。因此，NMX应按照运算时的要求，设置为N的最大值。NMX取值限于1到1000之间。如果目标设备不能为NMX提供足够的存储空间，或NMX受限，那么输出QF则设置为1，输出Y保留其循环运算时的默认值。

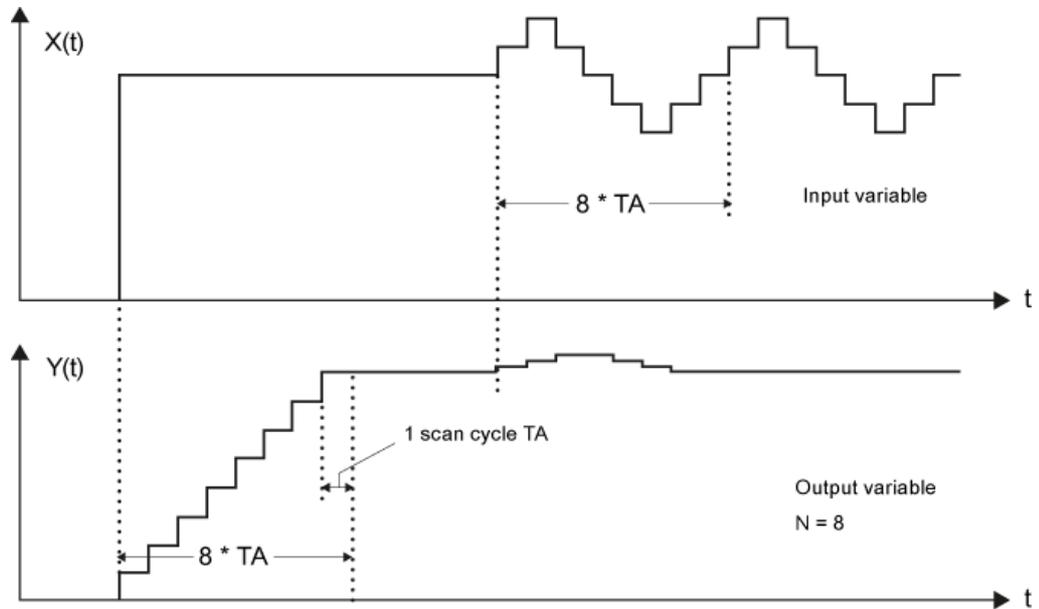
由于NMX在运算时不可以动态改变，因此NMX应该指定为常量。

应用场景

该功能块可用作计算平均值的斜坡函数发生器或滤波块。它可以充当频率 f_k 的低通带阻滤波器。

$$f_k = \frac{k}{N \cdot T_A}$$

$k = 1, 2, \dots$



多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
N	计算平均值所用的周期数	10	1...1000	
NMX	最大周期数	100	1...1000	
SV	设定值	0.0	REAL	
S	设定	0	0/1	
Y	输出变量	0.0	REAL	
QF	错误	0	0/1	

据

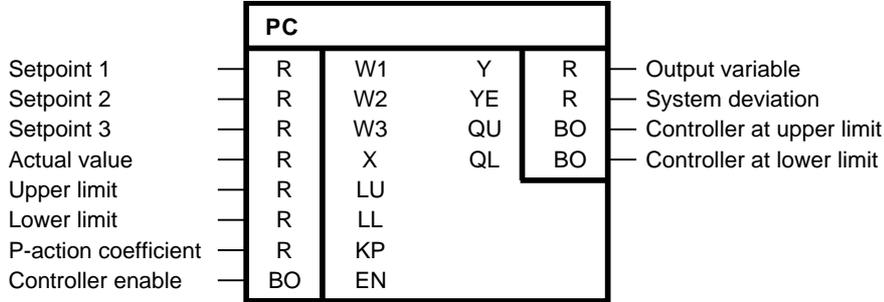
可在线插入	是
特别注意点	-

7.9 PC P作用控制器

☑ SIMOTION

☑ SINAMICS

符号



简要说明

- 支持3个设定值输入和1个实际值输入的P控制器
- 功能块实际值的符号颠倒
- 达到给定限值时的指示
- 对于正常控制器运算，LU的正限值必须明确，LL的负限值也必须明确。

操作模式

三个设定值W1、W2和W3相加，再从设定值的总和中减去实际值X。 计算结果YE乘以比例系数KP，在输出Y处输出。

算法：

$$Y = KP \cdot YE = KP \cdot (W1 + W2 + W3 - X)$$

$$YE = W1 + W2 + W3 - X$$

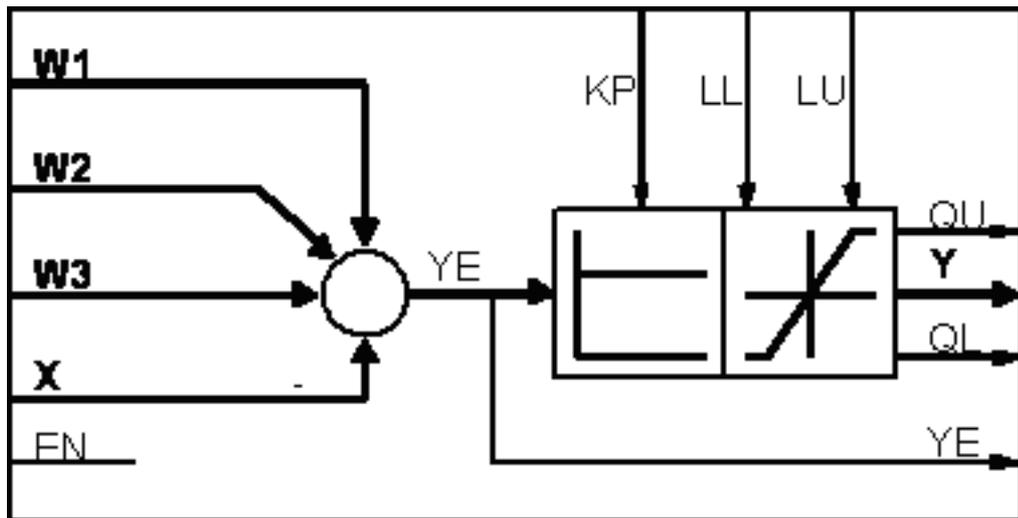
在任何运算模式下都会计算系统偏差YE，且单独开始。

控制器的输出Y可以通过输入LU和LL来确定取值范围。 当输出Y达到其中一个限值时，就会通过输出QU和QL来发送一个消息。 当LL >= LU时，输出Y = LU。

当EN = 1时，控制器启动。当EN = 0时，输出变量Y设置为0。 控制器禁用。 按照KP*YE为0时处理二进制输出QU和QL。

当选择负数KP值（反向放大器）时，控制器反向运算。

方框图



真值表

EN	条件	Y	QU	QL	模式
0	$LL < 0 < LU$	0	0	0	控制器禁用
0	$LU \leq 0$	0	1	0	控制器禁用
0	$LL \geq 0$	0	0	1	控制器禁用
1	$LL < YE * KP < LU$	$KP \times YE$	0	0	控制器启用
1	$YE * KP \geq LU$	LU	1	0	上限处的控制器
1	$YE * KP \leq LL$	LL	0	1	下限处的控制器

LL>= LU时的真值表

EN	条件	Y	QU	QL	模式
0	无	0	1	1	控制器禁用
1	$LL \geq LU$	LU	1	1	上限处的控制器

多个块连接

块 连接	说明	预分配	取值范围	数
W1	设定值1	0.0	REAL	
W2	设定值2	0.0	REAL	
W3	设定值3	0.0	REAL	
X	实际值	0.0	REAL	
LU	上限	0.0	REAL	
LL	下限	0.0	REAL	
KP	P作用系数	0.0	REAL	
EN	控制器启用	0	0/1	
Y	输出变量	0.0	REAL	

7.9 PC P作用控制器

块 连接	说明	预分配	取值范围	数
YE	系统偏差	0.0	REAL	
QU	上限处的控制器	1	0/1	
QL	下限处的控制器	1	0/1	

据

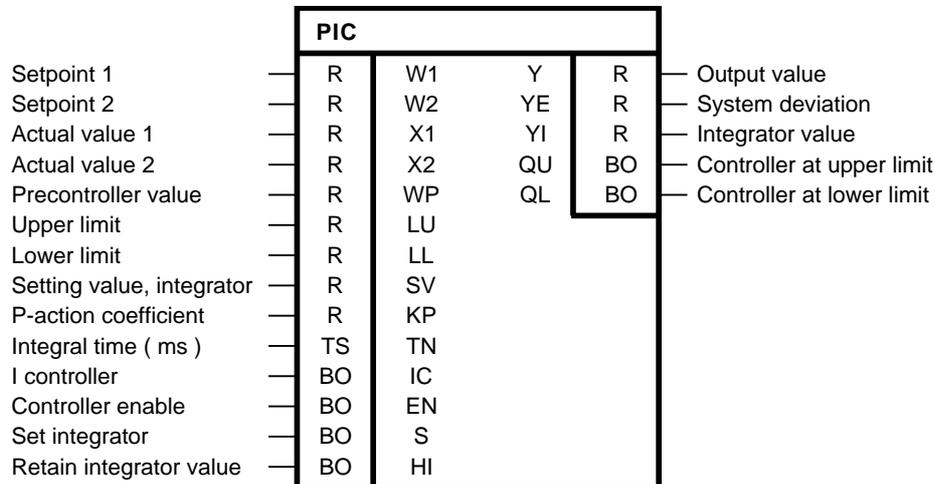
可在线加载	是
特别注意点	-

7.10 PIC PI控制器

SIMOTION

SINAMICS

符号



简要说明

- 通用PI控制器，可转换为P控制器模式和I控制器模式。 可用作速度控制器或主控制器。 适用于动态倍率控制
- 灵活的积分器具有以下功能：
 - 设置初始值 积分器中加载SV
 - 保留积分器当前值 P控制器
 - 通过SV的积分器控制
 - 通过控制限制的积分器控制
 - 增益关闭 I控制器
- 整体控制功能：
 - 运算过程中独立设置修改以下变量：
 - 比例系数KP
 - 积分作用时间TN
 - 控制限值LU和LL
 - 预先控制值WP，如用于加速供油
- 第二个实际值输入X2，如用于动态供油
- 达到给定限值时的指示

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块，那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

根据以下公式，从设定值W1+W2之和中减去实际值X1+X2之和：

$$YE = (W1 + W2) - (X1 + X2)$$

计算结果即为系统偏差YE，再乘以可变比例系数KP。乘积被传递给输出求和设备积分器。可变积分时间TN决定着控制器的积分行为。输出变量YI改变量与输入变量KP*YE成正比，而与积分作用时间TN成反比。积分值YI提供给输出求和设备。通过输入WP，另一个区分正负号的值可以与输出值Y相加。

根据该算法计算离散值。

算法：

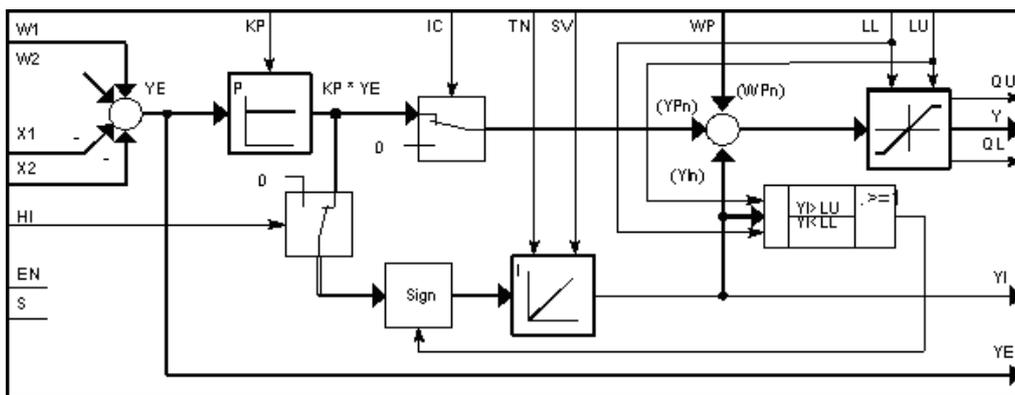
$$Y_n = Y_{n-1} + KP \cdot \left[\left(1 + \frac{TA}{TN} \right) \cdot YE_n - YE_{n-1} \right]$$

限制条件：LL < Y < LU且LL < LU

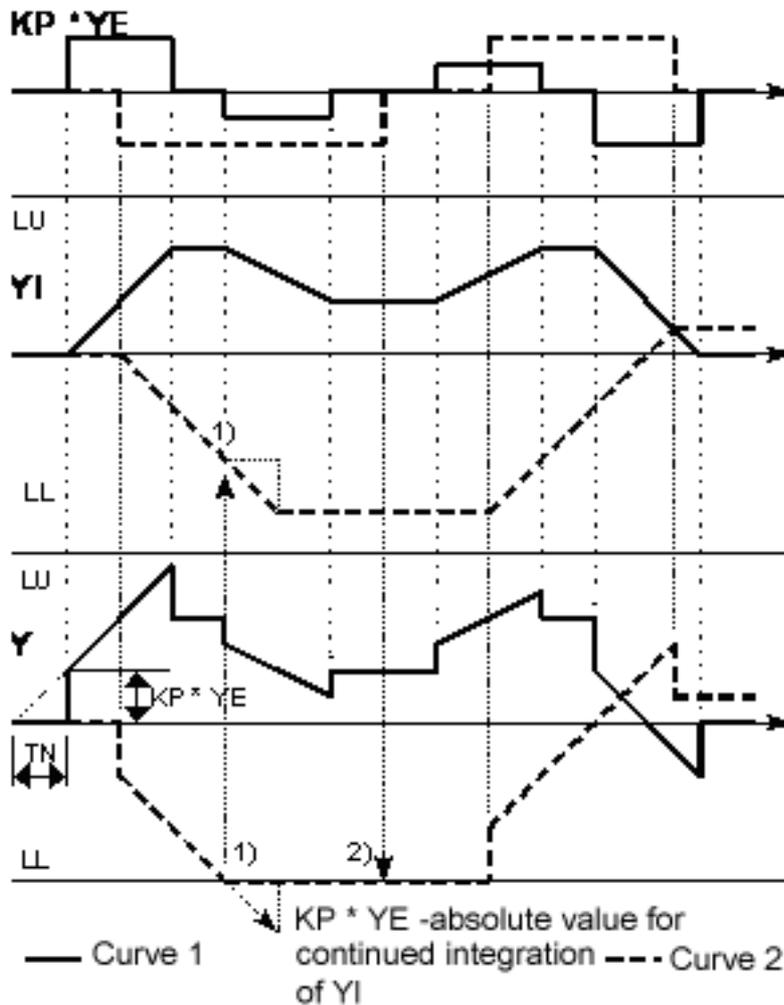
Y _n	扫描间隔n内的Y值
Y _{n-1}	扫描间隔n-1内的Y值

TA即是配置功能块时的采样时间。

方框图



XY图



曲线1和曲线2表示YE跳变过程中Y和YI的特性：

- 曲线1，正常运算，无限制
- 曲线2，有限制（如 LL）

对于曲线2) $YE * KP$ 会下降，但可以通过曲线1) 的连续积分中止。

控制器的运算模式与控制

控制器的输出值Y和积分值YI可以通过输入LU和LL来确定取值范围。当输出变量Y达到给定限值时，则会发出消息 $QU = 1$ 或 $QL = 1$ 。

以下优先序位适用于所有的控制输入：

$EN > IC > S > HI$ 。

控制输入处的指令输入：

控制输入	值	功能
EN	1	控制器启用
IC	1	PI控制器到I控制器的转换
S	1	接受积分设定值，但不做积分运算
HI	1	保留积分输出YI，但不做积分运算

7.10 PIC PI控制器

控制输入处的指令组合和可能的运算模式可参见真值表。

在正常的控制器运算中, $LL \leq 0 \leq LU$ 且 $LL < Y_n < LU$ 。然而, 以下的其他设置是有可能的。此处会适当转换算法:

$$Y_n = KP \cdot YE_n + YI_n + WP_n$$

与LU和LL相组合, 有5种不同的运算条件:

编号	条件	Y_n
	$LL < LU$	
1	$LL < KP \cdot YE_n + YI_n + WP_n < LU$	$KP \cdot YE_n + YI_n + WP_n$
2	$KP \cdot YE_n + YI_n + WP_n \geq LU$	LU
3	$KP \cdot YE_n + Y_n + WP_n \leq LL$	LL
	$LL = LU$	
4	无	LU
	$LL > LU$	
5	无	LU

通过自限制的积分器控制

控制过程中当输出Y达到设定的限值LL或LU时, 积分YI可能会继续运行, 直达到限值而保持该值。

当控制器达到限值, 且限值又改变时, 在倍率确定的情况下输出Y立即取新限值。然而, 在改变速度YIn处积分器会更新为新的限值。

真值表

运算条件1

EN	IC	S	HI	ΔYI_n	YI_n	Y_n	模式	备注
0	*	*	*	*	0	0	控制器禁用	KP、RN、WP、LU、LL、YE的任意值

EN	IC	S	HI	ΔYI_n	YI_n	Y_n	模式	备注
1	0	0	0	$KP \cdot YE_n \times TA/TN$	$YI_{n-1} + \Delta YI_n$	$KP \cdot YE_n + YI_n + WP_n$	PI控制器	控制器启用, 正常运算
1	1	0	0	$KP \cdot YE_n \cdot TA/TN$	$YI_{n-1} + \Delta YI_n$	$YI_n + WP_n$	I控制器	P作用分量 = 0
1	0	1	*	*	SV_n	$KP \cdot YE_n + YI_n + WP_n$	P控制器, 积分引导	$YI_n = SV_n$
1	1	1	*	*	SV_n	$YI_n + WP_n$	I控制器, 积分引导	$YI_n = SV_n$
1	0	0	1	0	YI_{n-1}	$KP \cdot YE_n + YI_n + WP$	P控制器, 积分 = 常量	$YI_n = YI_{n-1}$
1	1	0	1	0	YI_{n-1}	$YI_n + WP_n$	I控制器, 积分 = 常量	$YI_n = YI_{n-1}$

*= 任意值

运算条件2

EN	IC	S	HI	ΔYI_n	YI_n	Y_n	模式	备注
1	0	0	0	$KP*YE_n \times TA/TN$	$YI_{n-1} + \Delta YI_n$ (当 $YI_{n-1} < LU$ 时) $YI_{n-1} - \Delta YI_n$ (当 $YI_{n-1} > LU$ 时) LU (当 $YI_{n-1} = LU$ 时)	LU	上限处的PI控制器	YI_n 积分 $\rightarrow LU$, 可能带负号 (-)
1	1	0	0	$KP*YE_n \times TA/TN$	$YI_{n-1} + \Delta YI_n$ (当 $YI_{n-1} < LU$ 时) $YI_{n-1} - \Delta YI_n$ (当 $YI_{n-1} > LU$ 时) LU (当 $YI_{n-1} = LU$ 时)	LU	上限处的I控制器	YI_n 积分 $\rightarrow LU$, 可能带负号 (-)
1	0	1	*	*	SV_n (当 $SV_n < LU$ 时) LU (当 $SV_n \geq LU$ 时)	LU	上限处的P控制器	$YI_n = SV_n$ 或 $YI_n = LU$
1	1	1	*	*	SV_n (当 $SV_n < LU$ 时) LU (当 $SV_n \geq LU$ 时)	LU	上限处的I控制器	$YI_n = SV_n$ 或 $YI_n = LU$, P作用分量 = 0
1	0	0	1	0	YI_{n-1}	LU	P控制器, 积分 = 常量	$YI_n = YI_{n-1}$ 或 $YI_{n-1} = LU$
1	1	0	1	0	YI_{n-1}	LU	I控制器, 积分 = 常量	$YI_n = YI_{n-1}$ 或 $YI_{n-1} = LU$, P作用分量 = 0

*= 任意值

运算条件3

EN	IC	S	HI	ΔYI_n	YI_n	Y_n	模式	备注
1	0	0	0	$KP*YE_n*TA/TN$	$YI_{n-1} + \Delta YI_n$ (当 $YI_{n-1} < LL$ 时) $YI_{n-1} - \Delta YI_n$ (当 $YI_{n-1} > LL$ 时) LL (当 $YI_{n-1} = LL$ 时)	LL	下限处的PI控制器	YI_n 积分 $\rightarrow LL$, 可能带负号 (-)
1	1	0	0	$KP*YE_n \times TA/TN$	$YI_{n-1} + \Delta YI_n$ (当 $YI_{n-1} < LL$ 时) $YI_{n-1} - \Delta YI_n$ (当 $YI_{n-1} > LL$ 时) LL (当 $YI_{n-1} = LL$ 时)	LL	下限处的I控制器	YI_n 积分 $\rightarrow LL$, 可能带负号 (-)
1	0	1	*	*	SV_n (当 $V_n > LL$ 时) LL (当 $SV_n \leq LL$ 时)	LL	下限处的P控制器	$YI_n = SV_n$ 或 $YI_n = LL$
1	1	1	*	*	SV_n (当 $V_n > LL$ 时) LL (当 $SV_n \leq LL$ 时)	LL	下限处的I控制器	$YI_n = SV_n$ 或 $YI_n = LL$, P作用分量 = 0
1	0	0	1	0	YI_{n-1}	LL	P控制器, 积分 = 常量	$YI_n = YI_{n-1}$ 或 $YI_{n-1} = LL$
1	1	0	1	0	YI_{n-1}	LL	I控制器, 积分 = 常量	$YI_n = YI_{n-1}$ 或 $YI_{n-1} = LL$, P作用分量 = 0

*= 任意值

运算条件4

EN	IC	S	HI	ΔYI_n	YI_n	Y_n	模式	备注
1	*	*	*	*	*	LL=LU	参见运算条件2或运算条件3	-

*= 任意值

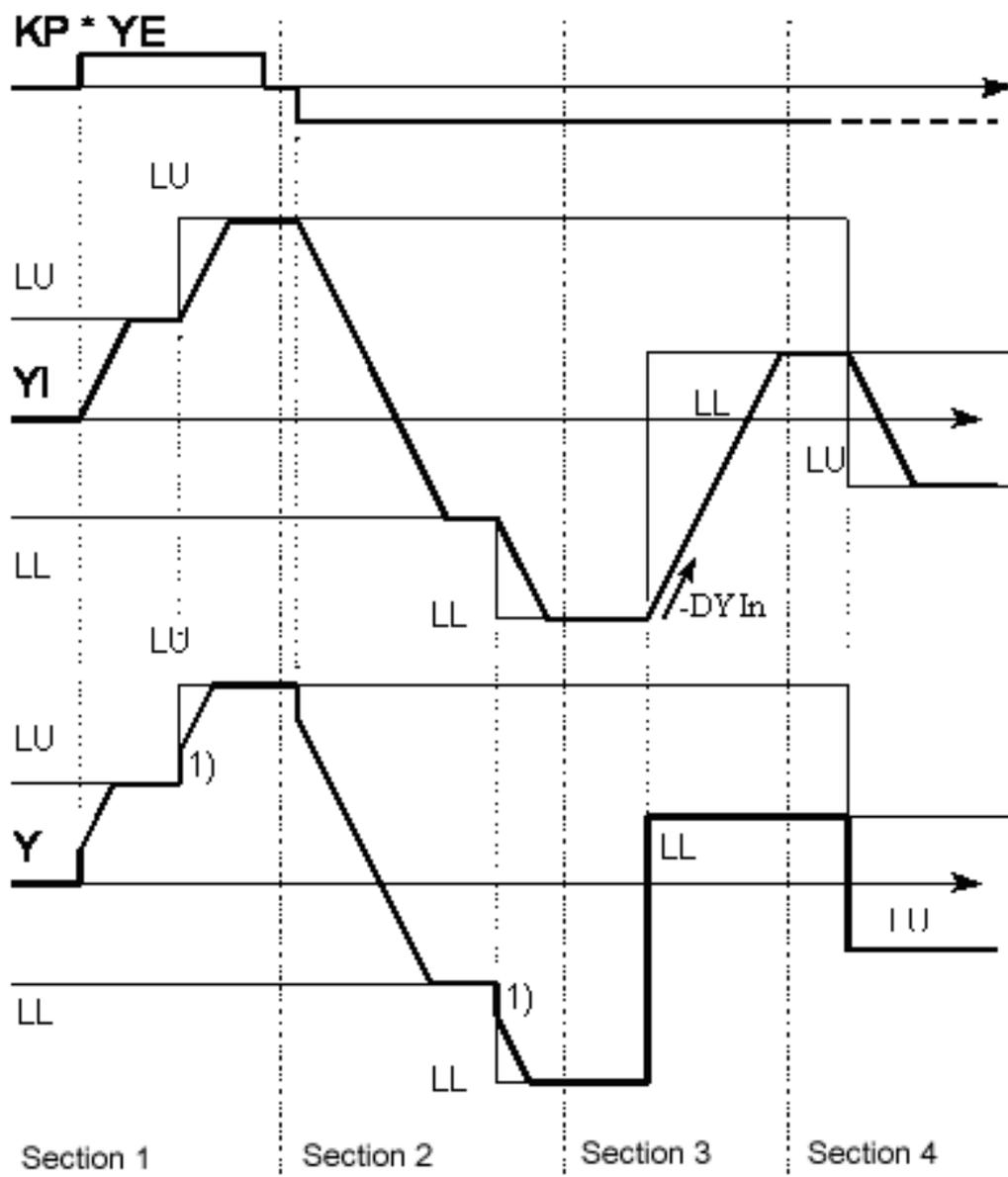
运算条件5

EN	IC	S	HI	ΔYI_n	YI_n	Y_n	模式	备注
1	*	*	*	$KP * YE_n TA / TN$	$YI_{n-1} + \Delta YI_n$ (当 $YI_{n-1} < LU$ 时) $YI_{n-1} - \Delta YI_n$ (当 $YI_{n-1} > LU$ 时) LU (当 $YI_{n-1} = LU$ 时)	LU	上限处的PI控制器	-

视限值改变的方向而定，必要的话会颠倒积分的正负号。

传递函数

在条件2、条件3和条件5下控制器修调倍率时的传递函数：



第1段：根据条件2，当 $LU_n > LU_{n-1}$ 时的曲线
 第2段：根据条件3，当 $LL_n < LL_{n-1}$ 时的曲线
 第3段：根据条件3，当 $LL_n > LL_{n-1}$ 时的曲线，当
 相对于控制方向的限值移动
 在积分输入处符号颠倒

第4段：根据条件5，当 $LLn > LUn$ 时的曲线

1) 由于积分值达到限值， $KP * YE$ 跳变。

PI控制器到I控制器的转换

当 $EN = 1$ 且 $IC = 1$ 时，P作用分量保留0，控制器由PI模式转换为I模式。输出Y取积分值YI。当控制过程中这种情况发生时， $-KP * YE$ 会在输出Y处跳变。当复位 $IC = 0$ 时，P分量复位为 $KP * YE$ 的当前值。控制器再次表现出PI行为特征。当控制过程中这种情况发生时， $KP * YE$ 会在输出Y处跳变。

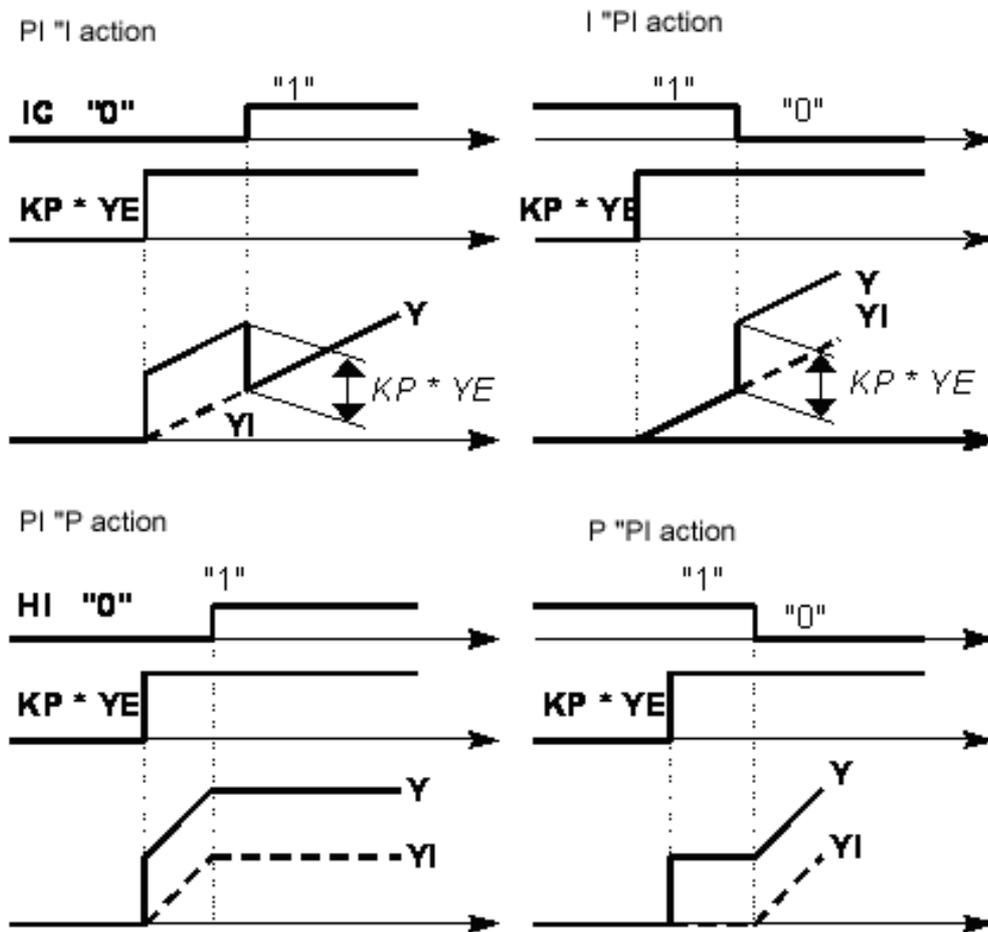
PI模式到P模式的转换

当块输入变为 $EN = 1$ 和 $HI = 1$ 时，保留积分YI，且控制器由PI模式平滑转换为P模式。YI继续充当输出Y的加数。

在复位 $HI = 0$ 时，积分再次启动。控制器再次表现出PI行为特征。

传递函数

转换过程中无控制器倍率修调的传递函数：当 $EN=1 \wedge S=0$ 时，例如



说明

总会计算输出系统偏差YE，与遗留控制指令和运算模式无关。积分器内部运算的准确性更高，因而即使在系统偏差很小的情况下，也可以进行积分运算。必须保证采样时间与积分时间 TN 相比，足够得短。T1的内部限制范围为： $TN \geq TA$ 。

多个块连接

块 连接	说明	预分配	取值范围	数
W1	设定值1	0.0	REAL	
W2	设定值2	0.0	REAL	
X1	实际值1	0.0	REAL	
X2	实际值2	0.0	REAL	
WP	预先控制值	0.0	REAL	
LU	上限	0.0	REAL	
LL	下限	0.0	REAL	
SV	设定值, 积分	0.0	REAL	
KP	P作用系数	0.0	REAL	
TN	积分时间 (毫秒)	0.0	SDTIME	
IC	I控制器	0	0/1	
EN	控制器启用	0	0/1	
S	给定积分	0	0/1	
HI	保留积分值	0	0/1	
Y	输出值	0.0	REAL	
YE	系统偏差	0.0	REAL	
YI	积分值	0.0	REAL	
QU	上限处的控制器	1	0/1	
QL	下限处的控制器	1	0/1	

据

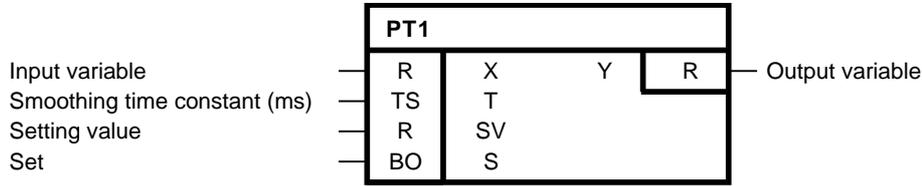
可在线加载	是
特别注意点	-

7.11 PT1 延迟元素

SIMOTION

SINAMICS

符号



简要说明

- 支持设定函数的一阶延迟元素
- 用作平滑滤波元素

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

设定函数未激活 (S = 0)

输入变量X因平滑滤波时间常量T而动态延迟, 并将其取值赋给功能块输出Y。

T决定输出变量值的上升陡度。它指定传递函数在何时上升至其最终值的63%。

在t = 3T之后, 传递函数大约达到其最终值的95%。

内部固定比例增益为1, 且不会改变。

当T/TA (T/TA > 10) 足够大时, 传递函数对应曲线:

$$Y(t) = X \cdot (1 - e^{-t/T})$$

其中, t = n * TA。

根据该算法计算离散值:

算法:

$$Y_n = Y_{n-1} + \frac{TA}{T} \cdot (X_n - Y_{n-1})$$

Y _n	扫描间隔n内的Y值
Y _{n-1}	扫描间隔n-1内的Y值
X _n	扫描间隔n内的X值

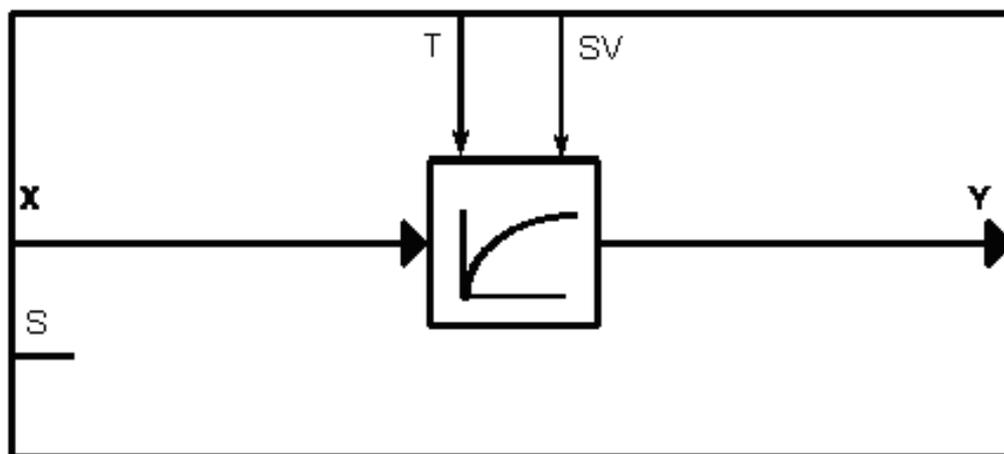
设定函数激活 ($S = 1$)

当设定函数激活时，当前设定值 SV_n 在输出变量处接受： $Y_n = SV_n$

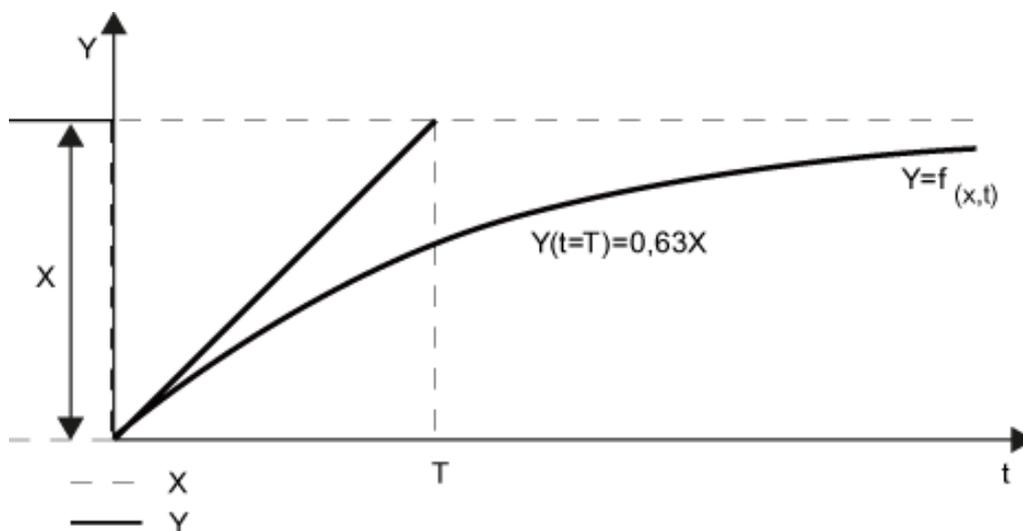
说明

$T1/TA$ 越大， Y 从一个扫描时间到下一个时间的振幅差就越小。 TA 即是配置功能块时的采样时间。 T 内部受限： $T \geq TA$ 。

方框图



传递函数



初始化

当输入 S 的值在初始化时为逻辑1时，设定值 SV 适用于输出 Y 。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
T	平滑滤波时间常量（毫秒）	0.0	SDTIME	
SV	设定值	0.0	REAL	
S	设定	0	0/1	
Y	输出变量	0.0	REAL	

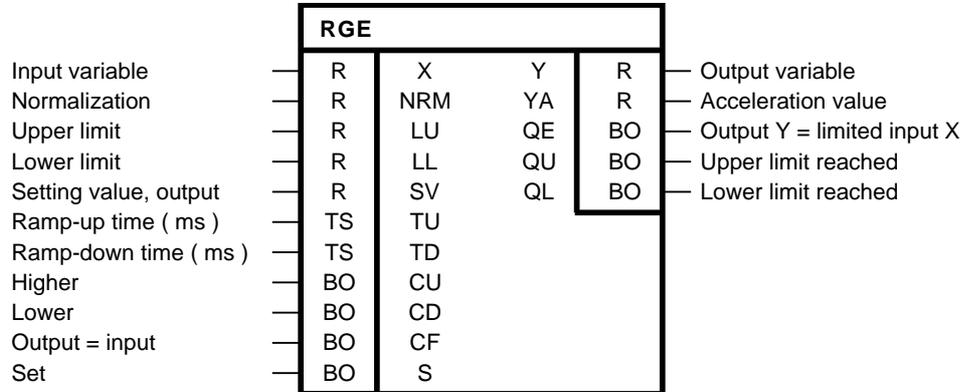
据

可在线加载	是
特别注意点	-

7.12 RGE 斜坡函数发生器

SIMOTION SINAMICS

符号



简要说明

- 用于限制输入变量X改变速度的斜坡函数发生器
- 可以对输出变量进行限制：
- 运算过程中独立设置修改以下变量：
 - 斜升时间和斜降时间
 - 输出限值LU和LL
 - 设定值
- 灵活的斜坡函数发生器具有以下功能：
 - 将校正合入设定值X
 - 斜坡函数发生器输出的初始值设置（->将SV载入积分器）
 - 求斜坡函数发生器输出的增加和减少的积分

说明

DCC SINAMICS：如果您在以下执行组中使用该功能块，那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

该功能块包含一个带有两个积分时间常数的积分器；其中，两个积分时间常数可以分别设置。输出Y根据算法改变：

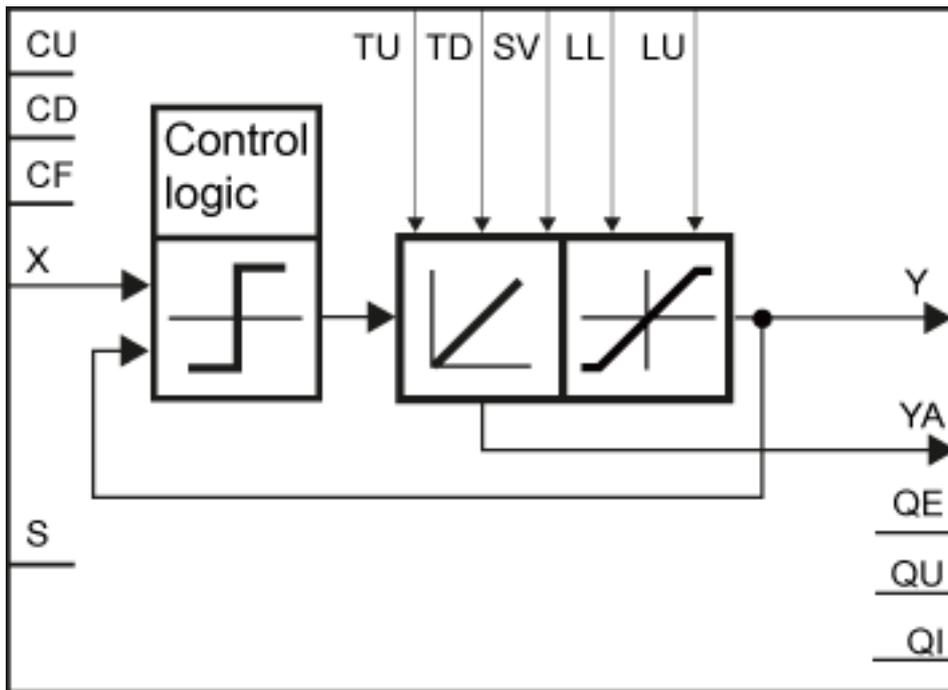
$$Y_n = Y_{n-1} + YA_n$$

斜升和斜降的加速度值YA分别计算，并在输出处输出。

输出值Y离开原点的过程称作斜升。

输出值Y趋近原点的过程称作斜降。

方框图



以下公式适用于斜升加速度值:

$$YA = \frac{TA}{TU} * NRM \text{ für } Y > 0$$

$$YA = -\frac{TA}{TU} * NRM \text{ für } Y < 0$$

以下公式适用于斜降加速度值:

$$YA = -\frac{TA}{TD} * NRM \text{ für } Y > 0$$

$$YA = \frac{TA}{TD} * NRM \text{ für } Y < 0$$

当方向改变或位于传递函数的原点时，斜升时间和斜降时间之间会发生转换。

运算模式由控制逻辑预先确定，取决于控制输入S、CF、CU和CD的逻辑状态。

输出变量可以通过输入LU和LL来确定取值范围。当Y达到设定限值时，二进制输出QU或QL为1。当Y = X时，二进制输出QE为1。

斜升时间和斜降时间

斜升时间TU是指输出变量的绝对值增加NRM所用的时间。

斜降时间TD是指输出变量的绝对值减少NRM所用的时间。斜升时间和斜降时间可以分别选择。

TA/TU或TA/TD越小，从一个采样时间到另一个采样时间的Y振幅改变就越小。TA是指处理该功能块的采样时间。

以下优先序位适用于所有的控制输入：

S > CF > CU和CD。

控制输入的函数：

S=1	向积分器中载入设定值SV，但不做积分。
CF=1	校正输出Y为设定值X，做积分。
CU=1	在方向LU上校正输出Y，做积分。
CD=1	在方向LL上校正输出Y，做积分。

斜坡函数发生器的运算模式与控制

控制输入处的指令组合和可能的运算模式可参见真值表。

在正常的斜升模式下， $LL \leq 0 \leq LU$ 且 $LL < Y_n < LU$ 。然而，以下的其他设置是有可能的。

以下原则适用于 $LL \geq LU$ 的条件：LU限值优先于LL限值。

位于限值时的积分器行为

控制过程中当输出Y达到设定的限值时，积分值则保持。然后输出值Y保持不变，直至积分值由于输入变量的改变而超出限值。

当积分器位于限值且限值改变时，积分器的行为视限值的改变方向而有所不同。

当限值的绝对值增加且控制逻辑规定了斜坡函数发生器在一个方向上运行时，积分器会依据设定斜升时间继续从先前保留值开始做积分，直至输出再次达到限值。

当限制的绝对值减小时，积分器会依据设定斜降时间从先前保留值开始做积分，直至输出再次达到限值。

说明

积分器内部运算的准确性更高，因而即使在给定实际值存在差异的情况下，也可以进行积分运算。相较于斜升时间或斜降时间，必须扫描时间足够得小。

说明

当限制的绝对值减小且当前的输出值超出取值范围时，积分器会依据设定斜降时间从当前值开始做积分，直至输出再次达到限值。此过程适用于所有运算模式。

TU和TD内部受限：TU \geq TA，TD \geq TA

真值表

S	CF	CU	CD	Y_{An}	Y_n	模式	备注
0	0	0	0	0	Y_{n-1}	停止	Y是常数

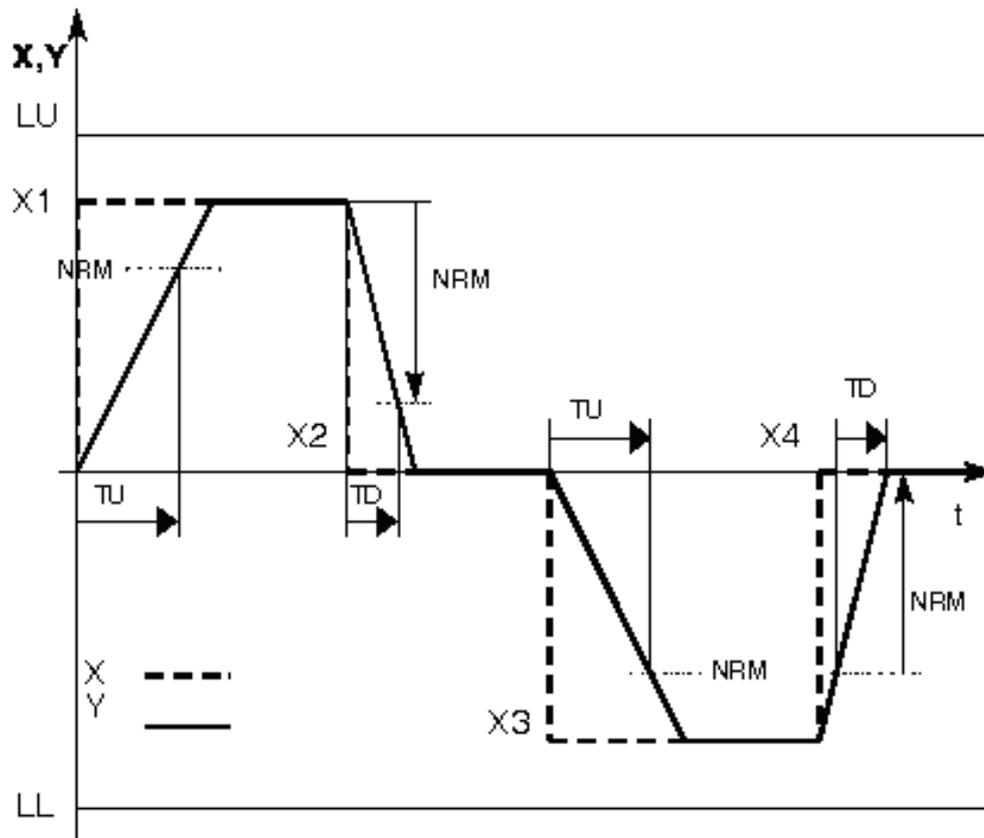
LL < LU和LL < 实际值 Y_{n-1} < LU

7.12 RGE 斜坡函数发生器

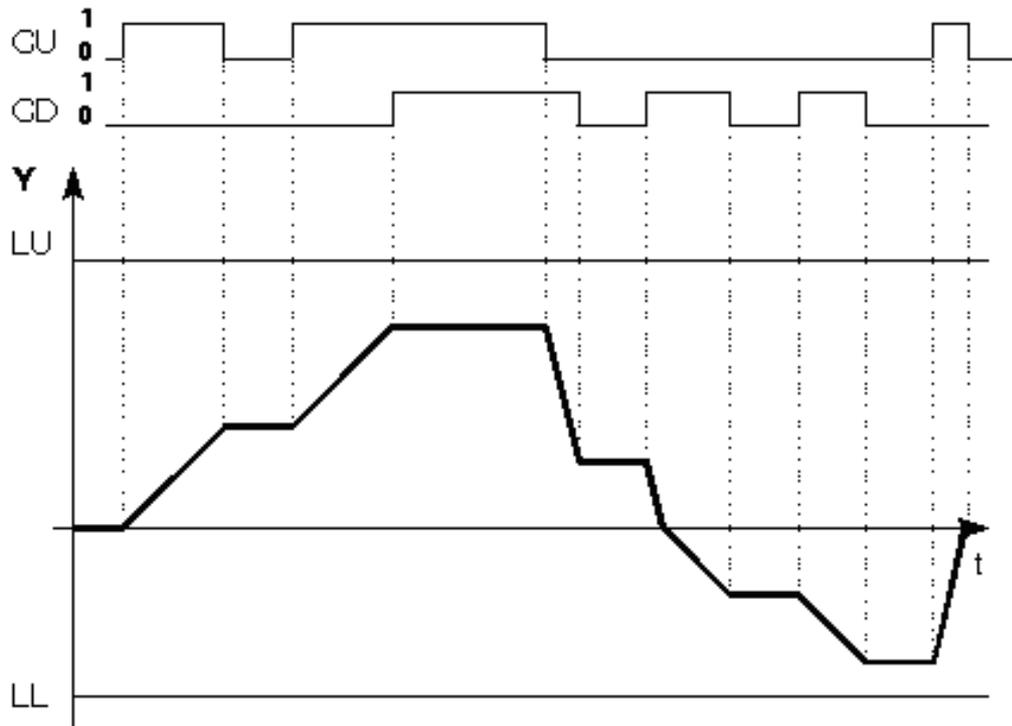
S	CF	CU	CD	YAn	Yn	模式	备注
1	*	*	*	跳变	SV _n	设置输出为SV	任意SV，固定值或变量
0	1	*	*	TA/TU; TA/TD	Y _{n-1} +YA _n	正常模式Y → X	当[X > Y ∧ Y ≥ 0] ∨ [X < Y ∧ Y ≤ 0]时为TU 当[X > Y ∧ Y < 0] ∨ [X < Y ∧ Y > 0]时为TD
0	0	1	0	TA/TU (TA/TD)	Y _{n-1} +YA _n	达到上限 限值Y → LU	TU和TD与先前一样，取决于起始位置
0	0	0	1	TA/TD (TA/TU)	Y _{n-1} +YA _n	达到下限 限值Y → LL	TU和TD与先前一样，取决于起始位置

* 不限

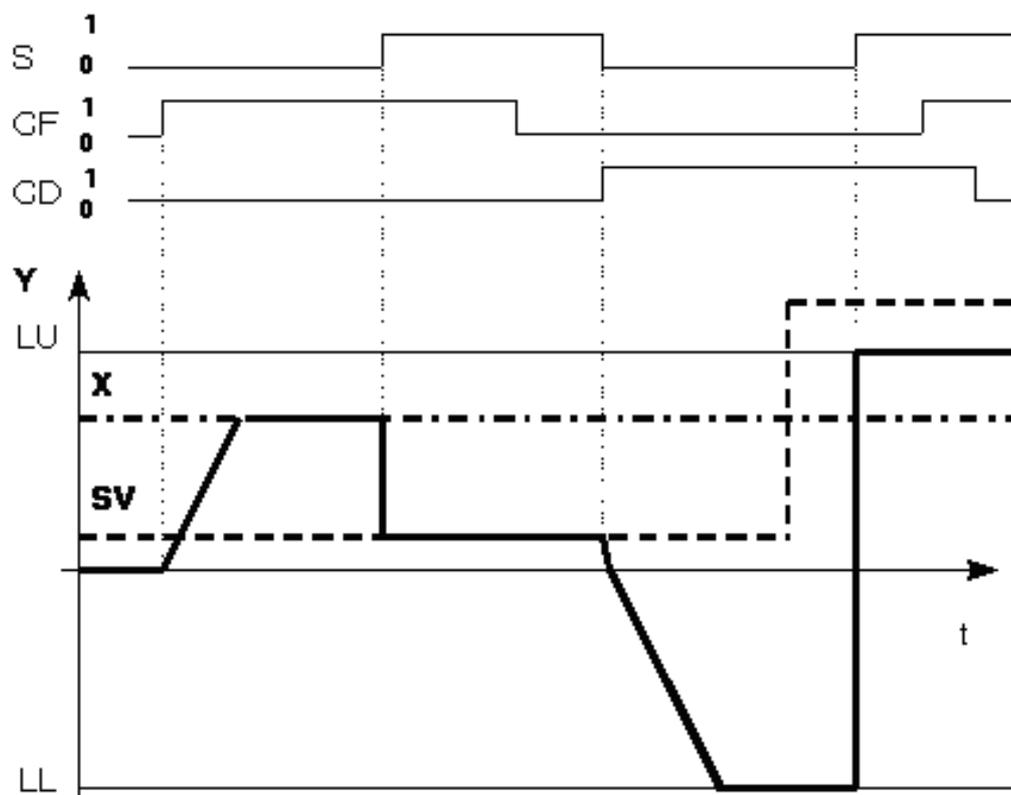
传递函数



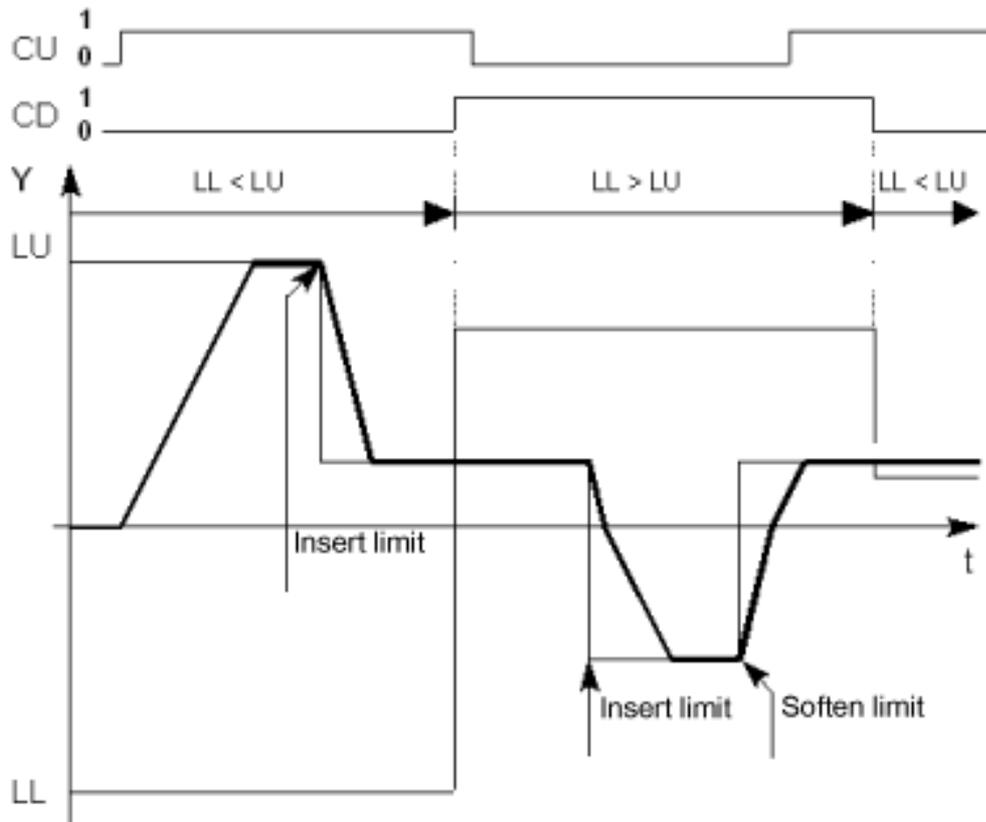
例1: 当 $LL < LU$ 且 $LL < X < LU$, 另外 $X_1 = 1.5$, $X_2 = X_4 = 0.0$, $X_3 = -1.5$, $LU = 2.0$, $LL = -2.0$, $TU > TD$, $NRM > 0$ 时, $CF = 1$



例2: 当CU和CD确定如图且 $LL < LU$ 时, 电机电势计函数曲线



例3: 当 $LL < LU$ 时的设定积分



例4：限值的改变与互换

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
NRM	标准化	1.0	REAL	
LU	上限	0.0	REAL	
LL	下限	0.0	REAL	
SV	设定值, 输出	0.0	REAL	
TU	斜升时间 (毫秒)	0.0	SDTIME	
TD	斜降时间 (毫秒)	0.0	SDTIME	
CU	更高	0	0/1	
CD	更低	0	0/1	
CF	输出 = 输入	0	0/1	
S	设定	0	0/1	
Y	输出变量	0.0	REAL	
YA	加速度值	0.0	REAL	
QE	输出Y = 受限输入X	0	0/1	
QU	达到上限	0	0/1	
QL	达到下限	0	0/1	

据

可在线加载	是
特别注意点	-

7.13 RGJ 带加加速度限制的斜坡函数发生器

☑ SIMOTION

☑ SINAMICS

符号

RGJ		
Input variable	R X Y R	Output variable
Normalization, TD/TU	R NRM YL R	Limited output variable
Control error	R EV YA R	Acceleration value
Upper limit	R LU YB R	Jerk value
Lower limit	R LL QE BO	Output Y = limited input X
Setting value, output	R SV QU BO	Upper limit reached
Acceleration setting value	R ASV QL BO	Lower limit reached
Weighting of the standard deviation	R WD	
Ramp-up time (ms)	TS TU	
Ramp-down time (ms)	TS TD	
Smoothing time during ramp up (ms)	TS TRU	
Smoothing time during ramp up (ms)	TS TR1	
Smoothing time during ramp up (ms)	TS TR2	
Smoothing time during ramp down (ms)	TS TRD	
Smoothing time during ramp down (ms)	TS TR3	
Smoothing time during ramp down (ms)	TS TR4	
Higher	BO CU	
Lower	BO CD	
Output = input	BO CF	
Upper limit reached	BO ULR	
Lower limit reached	BO LLR	
Smoothing on	BO RQN	
Set acceleration	BO SA	
Set	BO S	
Enable	BO EN	

简要说明

- 带加加速度限制和校正功能的斜坡函数发生器
- 斜坡函数发生器具有以下功能：
 - 设置输出Y或加速度YA
 - 校正斜坡函数发生器输出为设定值X，带积分和加加速度限制
 - 求斜坡函数发生器输出的增加和减少的积分
 - 根据限制时低平控制器的系统偏差校正斜坡函数发生器

说明

DCC SINAMICS: 如果您在以下执行组中使用该功能块, 那么您必须设定参数p2048的值为同步主周期时钟。- 接收AFTER IF1 PROFIdrive PZD- 发送BEFORE IF1 PROFIdrive PZD- 接收AFTER IF1 PROFIdrive flexible PZD- 接收AFTER IF2 PZD- 发送BEFORE IF2 PZD- 接收AFTER IF2 flexible PZD

操作模式

该功能块限制设定值的加速度（速度改变）和加加速度（加速度改变）。

适用以下公式:

$$\begin{aligned} Y_n &= \bar{Y}_{n-1} + \bar{Y}A_n \\ YA_n &= YA_{n-1} + YB_n \end{aligned}$$

斜升和斜降的加速度值YA和加加速度值YB分别计算。这就要求配置在斜升TRU区间内的斜升时间TU和平滑滤波时间以及在斜降TRD区间内的斜降时间TD和平滑滤波时间。

以下公式适用于斜升区间内平滑滤波时间以外的加速度值YA:

$$YA = YA_{\max} = \frac{TA}{TU} * NRM \quad \text{for } Y > 0$$

$$YA = YA_{\max} = -\frac{TA}{TU} * NRM \quad \text{for } Y < 0$$

以下公式适用于斜降区间内平滑滤波时间以外的加速度值YA:

$$YA = YA_{\max} = -\frac{TA}{TD} * NRM \quad \text{for } Y > 0$$

$$YA = YA_{\max} = \frac{TA}{TD} * NRM \quad \text{for } Y < 0$$

以下公式适用于斜升区间内平滑滤波时间以内的加加速度值YB:

$$YB = \frac{TA \cdot YA_{\max}}{TRU}$$

or

$$YB = \frac{TA \cdot YA_{\max}}{TR1} \qquad YB = \frac{TA \cdot YA_{\max}}{TR2}$$

以下公式适用于斜降区间内平滑滤波时间以内的加加速度值YB:

$$YB = \frac{TA \cdot YA_{max}}{TRD}$$

or

$$YB = \frac{TA \cdot YA_{max}}{TR3}$$

$$YB = \frac{TA \cdot YA_{max}}{TR4}$$

运算模式由控制逻辑预先确定，取决于二进制变量EN、S、SA、CF、CU和CD的逻辑状态。

输入变量X和其间接输出变量Y通过块输入LU和LL来确定取值范围。当Y达到给定限值时，则会向二进制输出发出消息QU = 1或QL = 1。

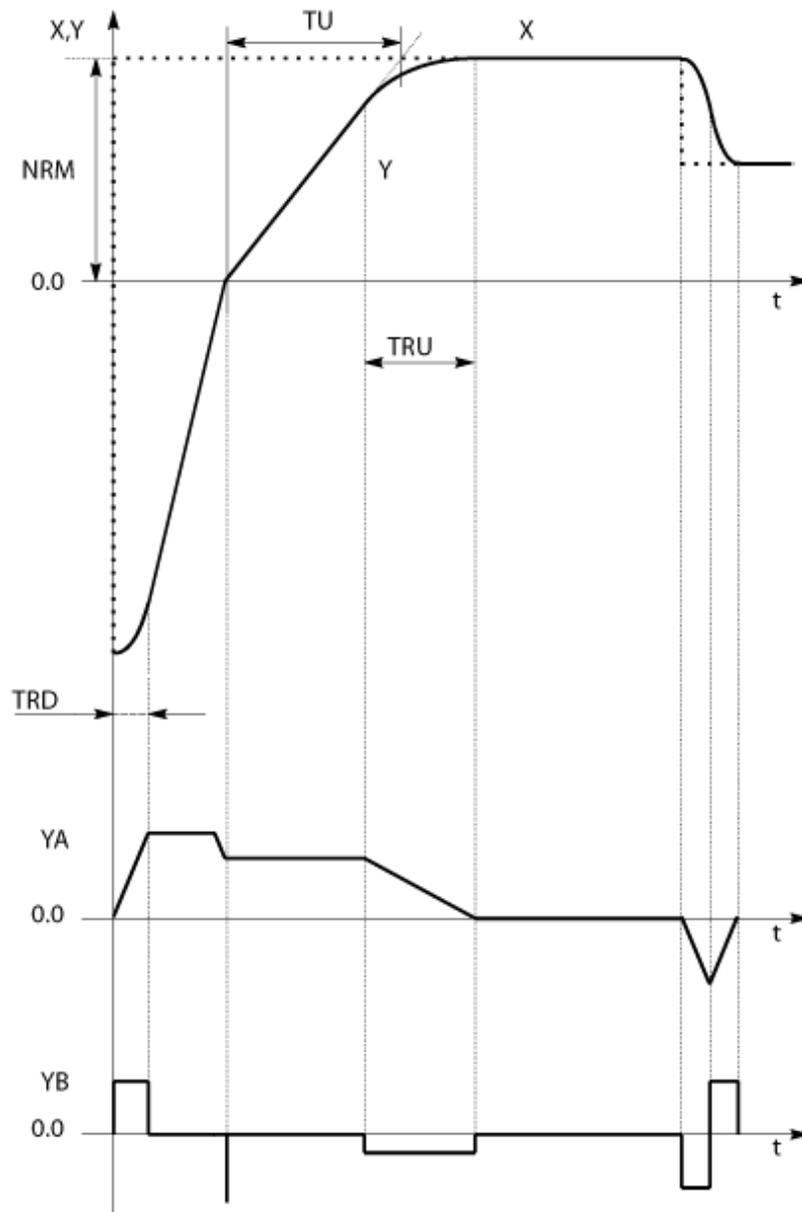
当输出变量Y等于输入变量X的限值时，二进制输出QE为1。

斜升过程细分为三个阶段：

- 阶段1
 - 当设定值X增加时，最大加加速度值YB（取决于TRU或TR1）在第一区段确定。因此，加速度随时间按比例增加；在此平滑滤波阶段，输出Y随时间按平方增加。
- 阶段2
 - 一旦在规定的斜升时间TU内达到最大加速度YA，加速度不变。输出变量Y随时间按比例增加。
- 阶段3
 - 在第三阶段，加速度随时间按比例减少。在此平滑滤波阶段，输出变量随时间（取决于TRU或TR2）按平方接近YB上的输入变量X。

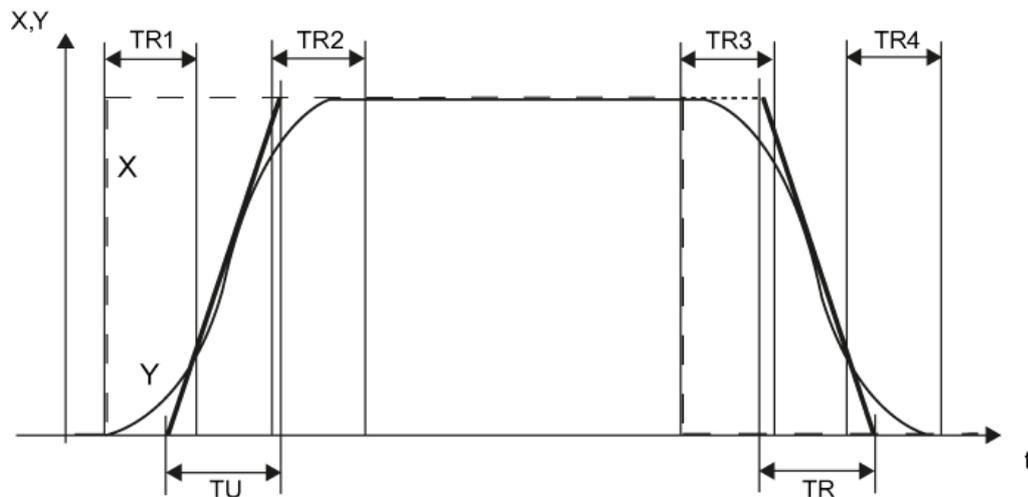
斜降过程的走势与之类似。

RGJ传递函数（正常斜升）



Ramp-up and ramp-down (not true to scale)

TRU=00.0 毫秒且TRD=0.0 毫秒时的取整时间



Ramp-up and ramp-down (not true to scale)

斜升时间和斜降时间

斜升时间 TU 是指输出变量的绝对值随时间按比例增加 NRM 所用的时间。

斜降时间 TD 是指输出变量的绝对值随时间按比例减少 NRM 所用的时间。

斜升时间和斜降时间可以分别选择。

斜升和斜降区间的平滑滤波时间

平滑滤波时间是指输出变量从不变初始值增加到最大加速度值所用的时间。在这段时间内，加加速度值不变且不等于0（参见阶段1）。

平滑滤波时间同样指输出变量从最大加速度值增加到恒定最终值所用的时间（参见阶段3）。斜升区间的平滑滤波时间通过 TRU 或 $TR1$ 和 $TR2$ 来确定，斜降区间的平滑滤波时间则通过 TRD 或 $TR3$ 或 $TR4$ 来确定。

每当设定值改变方向时，系统就会随相关的平滑滤波过程，由斜升转为斜降或由斜降转为斜升，具体为何种转换取决于初始位置。当运算过程中斜升或斜降时间改变时，同样会发生以上转换。

当斜降之后接着斜升，而 TRD 和 TD 小， TRU 和 TU 大时， YA 在斜降时就已经减少了。因而，只要目标值（ X ， LL 或 LU ）和斜坡函数发生器时间（ TU 、 TD 、 TRU 、 TRD ）不发生改变，在斜升时就不会出现突增。

当平滑滤波（ $RQN=0$ ）和校正功能（ $ULR=LLR=0$ ）关闭时， RGJ 功能块的行为与 RGE 相同。

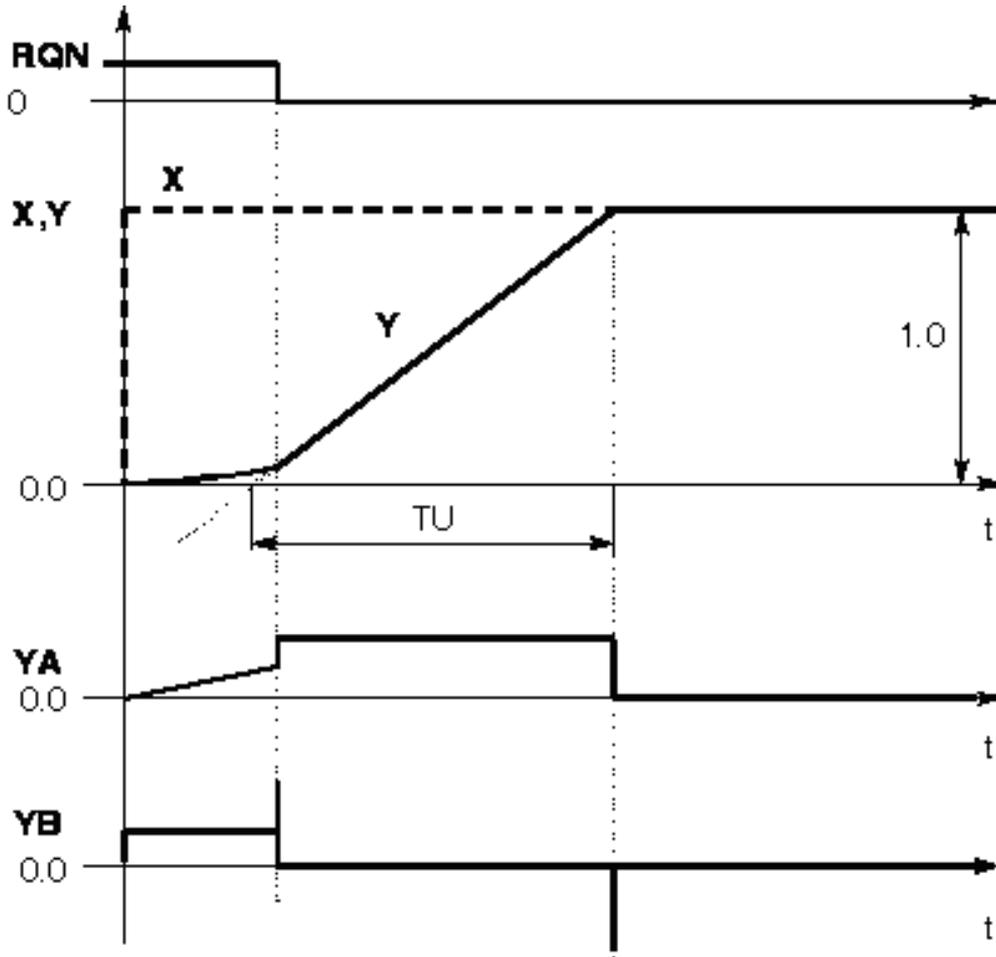
说明

当在零交叉时，不会进行平滑滤波。因此，由于物理原因，在斜降时间内的不同斜升时间点，加速度会发生跳变。如果需要，可以通过在 RGJ 输入处指定“中间设定值=0”来在零交叉时强制进行平滑滤波。

启动平滑滤波（加加速度限制）

当 $RQN=1$ 时，斜升和斜降区间的平滑滤波生效。

传递函数：斜升时撤消平滑滤波



当RQN = 0时，平滑滤波停止。根据TU或TD指定的斜升/斜降时间进行斜升/斜降。

当平滑滤波时加加速度限制关闭时，同样会根据TU或TD指定的斜升/斜降时间进行剩余的斜升/斜降。

“平滑滤波关闭”模式

如果在此模式下使用该功能块，按以下操作执行：

- 设置连接TRU、TR1、TR2、TRD、TR3和TR4为0（所有平滑滤波时间为0）。
- 设置连接RQN为1（“平滑滤波打开”模式）。

做这些设置后，RGJ功能块的行为与“平滑滤波关闭”模式（RQN = 0）中描述的一致。

斜坡函数发生器的运算模式与控制

控制输入的定义如下：

EN=1	启动斜坡函数发生器
S=1	设置输出Y为设定值SV，不做积分
SA=1	设置加速度YA为设定值ASV，不做积分
CF=1	校正输出Y为设定值X，做积分。
CU=1	在方向LU上校正输出Y，做积分。
CD=1	在方向LL上校正输出Y，做积分。

说明

以下优先序位适用于所有的控制输入：EN > S > SA > CF > CU和CD。控制输入处的指令组合和可能的运算模式可参见真值表。在正常的斜坡函数发生器运算中， $LL \leq 0 \leq LU$ 且 $LL \leq Y_n \leq LU$ 。然而，以下的其他设置也是有可能的。以下原则适用于 $LL \geq LU$ 的条件：LU限值优先于LL限值。在转换过渡过程中，不会超过加速度和加加速度的值。如果适用的话，根据设定值的定义或限制偏移量，在阶段1和阶段3会出现只有平滑滤波过渡的曲线。此时，不会出现与时间成正比的输出变量Y的曲线。

斜坡函数发生器停止

EN	S	SF	CF	CU	CD	Y _{An}	Y _n	模式	备注
0	*	*	*	*	*	0	0	抑制	Y=0
1	0	0	0	0	0	0	Y _{n-1}	抑制	Y = 常数

*= 任意值

LL < LU和LL < 实际值 Y_{n-1} < LU

EN	S	SF	CF	CU	CD	Y _{An}	Y _n	模式	备注
1	1	*	*	*	*	跳变	SV _n	设置输出为SV	任意SV，固定值或变量
1	0	1	*	*	*	ASV _n	Y _{n-1} + Y _{An}	在ASV上设置输出为积分器 1	任意ASV，固定值或变量
1	0	0	1	*	*	TA/TU (TA/TD)	Y _{n-1} + Y _{An}	正常模式 Y → X	当[X>Y ∧ Y ≥ 0] ∨ [X<Y ∧ Y<=0]时为TU 当[X>Y ∧ Y<0] ∨ [X<Y ∧ Y>0]时为TD 当达到Y=X时，设置QE=1。
1	0	0	0	1	0	TA/TU (TA/TD)	Y _{n-1} + Y _{An}	达到上 限值 Y → LU	如上TU和TD，取决于起始位置 当达到Y=LU时，设置QU=1和QE=1。
1	0	0	0	0	1	TA/TD (TA/TU)	Y _{n-1} + Y _{An}	达到上 限值 Y → LL	如上TU和TD，取决于起始位置 当达到Y=LL时，设置QL=1和QE=1。

校正斜坡函数发生器

通常情况下，斜坡函数发生器的输出Y引导为从属控制环（如：速度控制器）的设定值。

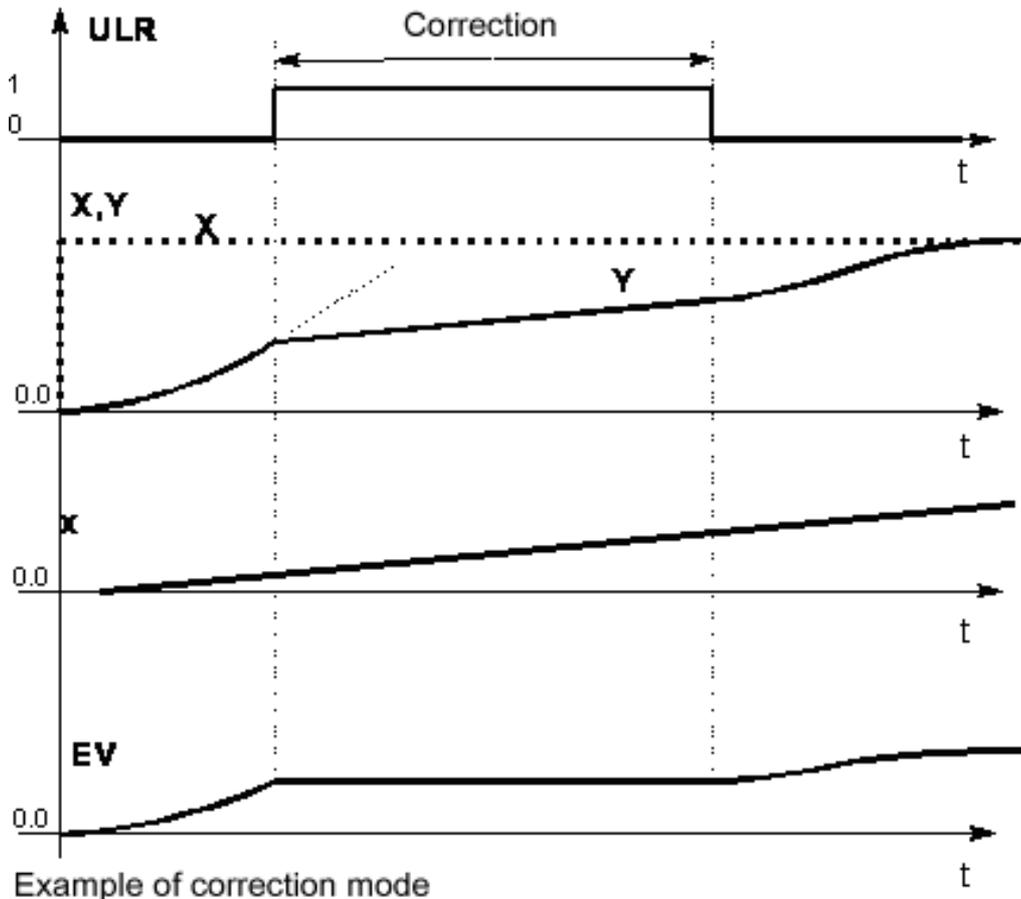
当控制器由于某种改变（如：斜升时）而达到限值时，斜坡函数发生器可能不会按照斜升时间增加输出。此时，通过系统偏差EV和权重因数WD来校正输出Y：

$$Y_n = Y_{n-1} - EV_n + WD \cdot EV_k$$

n= 扫描间隔n

k= 控制器初次达到限值（0 → ULR或LLR上的1沿）的时间

7.13 RGJ 带加加速度限制的斜坡函数发生器



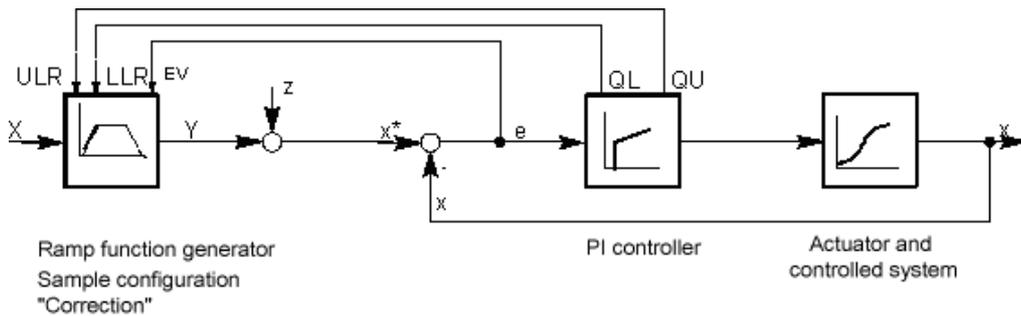
Example of correction mode

通常情况下，校正仅用于“常规控制环”（如：PI速度控制器）。控制器限值必须正确设置（如：与当前限值一致）。

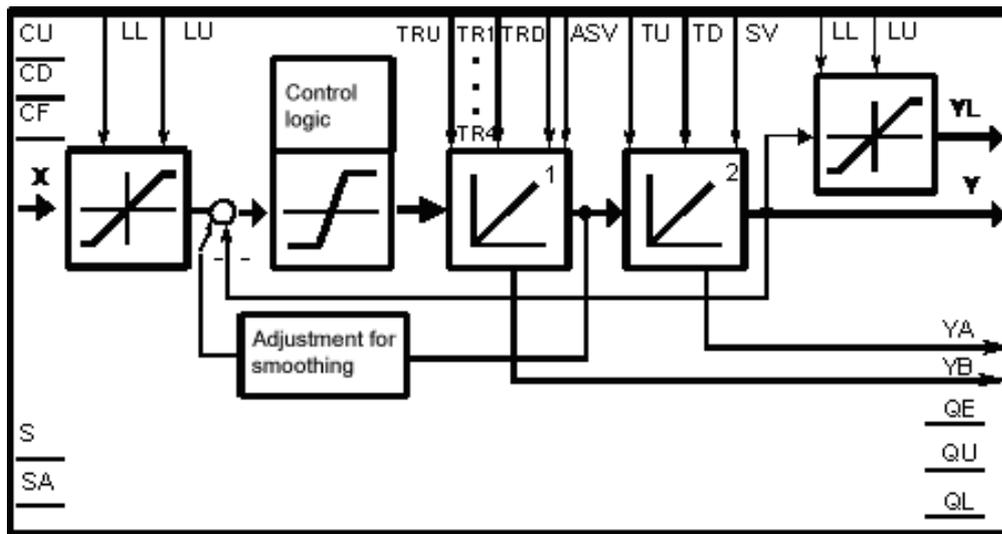
通常情况下，WD在1.01到1.1之间（> 1.0!）。校正时加加速度限制未激活。

控制器的二进制输出（“达到的上限/下限”）返回给二进制输入ULR或LLR。当达到限值时，通过RGJ功能块的反馈来设置二进制输入ULR = 1或LLR = 1，激活校正。

当没有使用校正时，ULR和LLR必须设为0。



方框图



说明

斜坡函数发生器运算的准确性更高，因而即使在给定实际值存在差异的情况下，也可以进行积分运算。必须保证采样时间与TU、TD、TRU、TR1、TR2、TR3、TR4和TRD时间相比，足够得短。

当 $NRM < 1.0e-18$ 时，输入值NRM内部设为1.0。

多个块连接

块 连接	说明	预分配	取值范围	数
X	输入变量	0.0	REAL	
NRM	标准化, TD/TU	1.0	REAL	
EV	控制错误	0.0	REAL	
LU	上限	0.0	REAL	
LL	下限	0.0	REAL	
SV	设定值, 输出	0.0	REAL	
ASV	加速度设定值	0.0	REAL	
WD	标准偏差权重	0.0	REAL	
TU	斜升时间 (毫秒)	0.0	SDTIME	
TD	斜降时间 (毫秒)	0.0	SDTIME	
TRU	斜升时的平滑滤波时间 (毫秒)	0.0	SDTIME	
TR1	斜升时的平滑滤波时间 (毫秒)	0.0	SDTIME	
TR2	斜升时的平滑滤波时间 (毫秒)	0.0	SDTIME	
TRD	斜降时的平滑滤波时间 (毫秒)	0.0	SDTIME	
TR3	斜降时的平滑滤波时间 (毫秒)	0.0	SDTIME	
TR4	斜降时的平滑滤波时间 (毫秒)	0.0	SDTIME	
CU	更高	0	0/1	
CD	更低	0	0/1	
CF	输出 = 输入	0	0/1	

7.13 RGJ 带加加速度限制的斜坡函数发生器

块 连接	说明	预分配	取值范围	数
ULR	达到上限	0	0/1	
LLR	达到下限	0	0/1	
RQN	平滑滤波打开	0	0/1	
SA	设置加速度	0	0/1	
S	设定	0	0/1	
EN	使能	0	0/1	
Y	输出变量	0.0	REAL	
YL	受限的输出变量	0.0	REAL	
YA	加速度值	0.0	REAL	
YB	加加速度值	0.0	REAL	
QE	输出Y = 受限输入X	0	0/1	
QU	达到上限	0	0/1	
QL	达到下限	0	0/1	

据

可在线加载	是
特别注意点	-

A.1 数据类型

DCBLIB 相关的数据类型如下表所示。

表 A1-1 块接口数据类型概览

缩写	数据宽	数据类型 (符合 IEC 61131-3)	DCB 块名称 后缀	PIN 名称 - 输入 - 输出	可互连的数据类型	说明
BO/B	1 位	BOOL	_B*	I、I1、I2..... Q、Q1、Q2.....	BOOL	布尔数
BY	8 位	BYTE	_BY	IS QS	BY、SINT、USINT	位串
W	16 位	WORD	_W		WORD、INT、UINT	位串
DW	32 位	DWORD	_DW		DWORD、DINT、UDINT	位串
SI	8 位	SINT	_SI	X、X1、X2..... Y、Y1、Y2.....	SINT、USINT、BY	带符号短整数
I	16 位	INT	_I		INT、UINT、WORD	带符号整数
DI/D	32 位	DINT	_D		DINT、UDINT、DWORD	带符号双整数
US	8 位	USINT	_US		SINT、USINT、BY	无符号短整数
UI	16 位	UINT	_UI		INT、UINT、WORD	无符号整数
UD	32 位	UDINT	_UD		DINT、UDINT、DWORD	无符号双整数
R	32 位	REAL	_R*		REAL、SDTIME	浮点单精度 (符合 IEEE 754)
LR	64 位	LREAL	_LR		LREAL	浮点双精度 (符合 IEEE 754)
TS	32 位	(SDTIME)	-	-	SDTIME、REAL	SDTIME 数据类型源自 REAL 数据类型，1.0 对应于 1.0 毫秒。 无负数值。
AID	32 位	-	-	-	DINT、UDINT、DWORD	告警 ID
*	-	定义的块	-	-	参见 DCC 编辑器描述	参见 DCC 编辑器描述

* 可选标识

DPV1 参数请求和响应的字段如下表所示。

表 A1-2 DPV1 参数请求和响应字段概览

字段	数据类型	取值	注释
请求参考	无符号 8	0x01 到 0xFF	主站请求 / 响应对的唯一标识。主站以新请求替换请求参考。从站在其响应中反映该请求参考。
请求 ID	无符号 8	0x01 0x02	读请求 写请求
			表示请求类型。对于写请求，更改是在非易失存储（RAM）中进行。必须执行保存操作，将更改数据转移至非易失存储（p0971, p0977）。
响应 ID	无符号 8	0x01 0x02 0x81 0x82	读请求 (+) 写请求 (+) 读请求 (-) 写请求 (-)
			反映请求 ID，表示对请求进行正向执行还是负向执行。负向表示：该请求可完全或部分不执行。对于每个子响应，转移错误值而非数值。
驱动对象编号	无符号 8	0x00 到 0xFF	编号
			为包含多个驱动对象的驱动单元设置驱动对象编号。处于独立参数编号区间的不同驱动对象可以通过同一个 DPV1 连接进行存取。
参数数目	无符号 8	0x01 到 0x27	No. 1 到 39 受限于 DPV1 消息帧长度
			对于多参数请求，定义以下字段数目： 参数地址和 / 或参数取值。对于单参数请求，参数数目为 1。
属性	无符号 8	0x10 0x20 0x30	取值 描述 文本（未执行）
			存取参数元素类型
元素数目	无符号 8	0x00 0x01 到 0x75	特别功能 No. 1 到 117 受限于 DPV1 消息帧长度
			存取数组元素数目
参数编号	无符号 16	0x0001 到 0xFFFF	No. 1 到 65535
			寻址存取参数。

表 A1-2 DPV1 参数请求和响应字段概览

字段	数据类型	取值	注释
子索引	无符号 16	0x0000 到 0xFFFF	No. 0 到 65535
	寻址待存取参数的第一个数组元素。		
格式	无符号 8	0x02	数据类型为整数 8
		0x03	数据类型为整数 16
		0x04	数据类型为整数 32
		0x05	数据类型为无符号 8
		0x06	数据类型为无符号 16
		0x07	数据类型为无符号 32
		0x08	数据类型为浮点
		其他取值	参见 PROFIdrive Profile V3.1
		0x40	零（无值作为写请求的正向子响应）
		0x41	字节
		0x42	字
		0x43	双字
		0x44	错误
格式与编号表示消息帧中包含取值的相邻空位。符合 PROFIdrive Profile 的数据类型优选用于写存取。字节、字和双字也可作为替代类型。			
值数目	无符号 8	0x00 到 0xEA	No. 0 到 234 受限于 DPV1 消息帧长度
	表示后续值数目。		
错误值	无符号 16	0x0000 到 0x00FF	错误值含意，参见附录 A.2
	负响应时的错误值。当值由奇数数目的字节构成时，则附加零字节。这保证了消息帧字结构的完整性。		
取值	无符号 16	0x0000 到 0x00FF	
	用于读存取或写存取的参数值当值由奇数数目的字节构成时，则附加零字节。这保证了消息帧字结构的完整性。		

A. 2 PROFIdrive 参数响应的错误值 数据类型

表 A2-1 DPV1 参数响应的错误值

错误值	含义	注释	其它信息
0x00	参数编号不合法。	要存取参数不存在。	-
0x01	参数值不可更改。	要修改的参数值不可更改。	子索引
0x02	超出上限值或下限值。	要修改的值超出限值。	子索引
0x03	子索引无效。	要存取的子索引不存在。	子索引
0x04	无数组。	通过子索引存取的参数无索引。	-
0x05	数据类型错误。	要修改的值与参数的数据类型不匹配。	-
0x06	设置操作不合法 (只允许复位操作)。	要修改的值不等于 0, 不允许修改。	子索引
0x07	描述元素不可更改。	要修改的描述元素不可更改。	子索引
0x09	描述数据不存在。	要存取的描述数据不存在 (参数值存在)。	-
0x0B	无操作优先权。	修改操作无优先权。	-
0x0F	无文本数组存在。	要存取的文本数组不存在 (参数值存在)。	-
0x11	当前运行状态不允许执行 该请求。	由于不明原因暂时不允许存取操作。	-
0x14	值不合法。	要存取的值处于取值范围内, 但由于其他固定原因 (参数已定义了个体值), 该值不合法。	子索引
0x15	响应过长。	当前响应长度超出最大可传输长度。	-
0x16	参数地址不合法。	属性、元素数目、参数编号、子索引或以上全部因素不允许或不支持该值。	-
0x17	格式不合法。	写请求: 参数数据格式不合法或不支持。	-
0x18	值数目不一致。	写请求: 参数数据的值数目与参数地址的元素数目不匹配。	-

表 A2-2 DPV1 参数响应的错误值

错误值	含义	注释	其它信息
0x19	驱动对象不存在。	要存取的驱动对象不存在。	-
0x20	参数的文本元素不可更改。	-	-
0x21	不支持 BMP 服务；请求 ID 无效。	-	-
0x22	不支持多参数存取。	-	-
0x65	当前参数未激活。	要存取的可用参数当前未激活（如 n 控制设置和对 V/f 控制参数的存取）。	-
0x6B	参数 %s [%s]：启动的控制器无写存取操作。	-	-
0x6C	参数 %s [%s]：未知单元。	-	-
0x6D	参数 %s [%s]：只能在调试状态下进行写存取，编码器 (p0010 = 4)。	-	-
0x6E	参数 %s [%s]：只能在调试状态下进行写存取，电机 (p0010 = 3)。	-	-
0x6F	参数 %s [%s]：只能在调试状态下进行写存取，电源模块 (p0010 = 2)。	-	-
0x70	参数 %s [%s]：只能在快速调试状态下进行写存取 (p0010 = 1)。	-	-
0x71	参数 %s [%s]：只能在准备状态下进行写存取 (p0010 = 0)。	-	-
0x72	参数 %s [%s]：只能在调试状态下进行写存取，参数复位 (p0010 = 30)。	-	-

表 A2-2 DPV1 参数响应的错误值

错误值	含义	注释	其它信息
0x73	参数 %s [%s]: 只能在调试状态下进行写存取, 安全 (p0010 = 95)。	-	-
0x74	参数 %s [%s]: 只能在调试状态下进行写存取, 工艺应用 / 单元 (p0010 = 5)。	-	-
0x75	参数 %s [%s]: 只能在调试状态下进行写存取 (p0010 不等于 0)。	-	-
0x76	参数 %s [%s]: 只能在调试状态下进行写存取, 下载 (p0010 = 29)。	-	-
0x77	参数 %s [%s] 不可在下载时进行写存取。	-	-
0x78	参数 %s [%s]: 只能在调试状态下进行写存取, 驱动配置 (设备: p0009 = 3)。	-	-
0x79	参数 %s [%s]: 只能在调试状态下进行写存取, 定义驱动类型 (设备: p0009 = 2)。	-	-
0x7A	参数 %s [%s]: 只能在调试状态下进行写存取, 数据集基础配置 (设备: p0009 = 4)。	-	-
0x7B	参数 %s [%s]: 只能在调试状态下进行写存取, 设备配置 (设备: p0009 = 1)。	-	-
0x7C	参数 %s [%s]: 只能在调试状态下进行写存取, 设备下载 (设备: p0009 = 29)。	-	-
0x7D	参数 %s [%s]: 只能在调试状态下进行写存取, 设备参数复位 (设备: p0009 = 30)。	-	-
0x7E	参数 %s [%s]: 只能在调试状态下进行写存取, 设备准备 (设备: p0009 = 0)。	-	-

表 A2-2 DPV1 参数响应的错误值

错误值	含义	注释	其它信息
0x7F	参数 %s [%s]: 只能在调试状态下进行写存取, 设备 (设备: p0009 不等于 0)。	-	-
0x81	参数 %s [%s] 不可在下载时进行写存取。	-	-
0x82	BI 禁止转移控制权 (主站): p0806。	-	-
0x83	参数 %s [%s]: 不允许请求的 BICO 互连。	BICO 输出不会为浮动值, 但 BICO 输入要求为浮动值。	-
0x84	参数 %s [%s]: 禁止更改参数 (参见 p0300、p0400、p0922)。	-	-
0x85	参数 %s [%s]: 存取方式未明确。	-	-
0xC8	低于有效值。	要修改的值虽然位于“绝对”限值范围内, 但低于当前有效下限值。	-
0xC9	高于有效值。	要修改的值虽然位于“绝对”限值范围内, 但高于当前有效上限值 (如由实际转换额定值确定)。	-
0xCC	不允许写存取。	由于存取密钥不可用, 因此不允许写存取。	-
0xFF	成功读 / 写过程。	已成功读取或写入该值。	-

A.3 功能块总览

对于新功能块而言，标明支持这些功能块的最低版本。

功能块	说明	SIMOTION	SINAMICS
ACOS	反余弦功能块	X	
ADD	加法器 (REAL型)	X	X
ADD_D	加法器 (DOUBLE INTEGER型)	X	X
ADD_I	加法器 (INTEGER型)	X	X
ADD_M	在正确的轴周期中增加值的模加法器	X	X
AND	逻辑AND运算 (BOOL型)	X	X
AND_W	逻辑AND运算 (WORD型)	X	
ASIN	反正弦功能块	X	
ATAN	反正切功能块	X	
AVA	支持符号评估的绝对值发生器	X	X
AVA_D	绝对值发生器 (DOUBLE INTEGER型)	X	X
B_BY	8个二进制变量到状态字节的转换器	X	
B_DW	32个二进制变量到状态双字的转换器	X	X
B_W	16个二进制变量到状态字的转换器	X	X
BF	闪烁功能块 (BOOL型)	X	X
BF_W	状态字闪烁功能块 (BOOL型)	X	
BSW	BOOL型二位切换开关	X	X
BY_B	转换器状态字节转换为8个二进制变量	X	
BY_W	状态字节到状态字的转换器	X	X
CNM	可控数字存储器 (REAL型)	X	X
CNM_D	可控数字存储器 (DOUBLE INTEGER型)	X	X
CNM_I	可控数字存储器 (INTEGER型)	X	X
COS	余弦功能块	X	V4.4
CTD	由内部时间戳确定时差	X	
CTR	计数器 (BOOL型)	X	X
D_I	DOUBLE INTEGER到INTEGER的转换器	X	X
D_R	DOUBLE INTEGER到REAL的转换器	X	X
D_SI	DOUBLE INTEGER到SHORT INTEGER的转换器	X	
D_UI	DOUBLE INTEGER到UNSIGNED INTEGER的转换器	X	X
D_US	DOUBLE INTEGER到UNSIGNED SHORT INTEGER的转换器	X	X
DCA	直径计算器	X	X
DEL	死区元素	X	X
DEZ	死区元素	X	X
DFR	复位优先D型触发器 (BOOL型)	X	X
DFR_W	复位优先D型触发器 (WORD型)	X	
DIF	微商作用元素	X	X
DIV	除法器 (REAL型)	X	X
DIV_D	除法器 (DOUBLE INTEGER型)	X	X

功能块	说明	SIMOTION	SINAMICS
DIV_I	除法器 (INTEGER型)	X	X
DLB	延迟元素 (REAL型)	X	X
DT1	平滑滤波元素	X	X
DW_B	状态双字到32个二进制变量的转换器	X	X
DW_R	接受位串为真值	X	X
DW_W	状态双字到状态字的转换器	X	X
DX8	多路输出选择器, 8个输出, 可级联 (REAL型)	X	X
DX8_D	多路输出选择器, 8个输出, 可级联 (DOUBLE INTEGER型)	X	X
DX8_I	多路输出选择器, 8个输出, 可级联 (INTEGER型)	X	X
ETE	沿评估器 (BOOL型)	X	X
GTS	读出时间戳	X	
I_D	INTEGER 到DOUBLE_INTEGER的转换器	X	X
I_R	INTEGER到REAL的转换器	X	X
I_SI	INTEGER到SHORT_INTEGER的转换器	X	
I_UD	INTEGER到UNSIGNED_DOUBLE_INTEGER的转换器	X	X
I_US	INTEGER到UNSIGNED_SHORT_INTEGER的转换器	X	X
INCO	轴向轴机转动惯量	X	X
INT	积分器	X	X
LIM	限制器 (REAL型)	X	X
LIM_D	限制器 (DOUBLE_INTEGER型)	X	X
LR_R	LONG_REAL到REAL的转换器	X	
LVM	磁滞双面限制监视器 (BOOL型)	X	X
MAS	最大值评估器	X	X
MFP	脉冲发生器 (BOOL型)	X	X
MIS	最小值评估器	X	X
MUL	乘法器 (REAL型)	X	X
MUL_D	乘法器 (DOUBLE_INTEGER型)	X	X
MUL_I	乘法器 (INTEGER型)	X	X
MUX8	多路复用器, 可级联 (REAL型)	X	X
MUX8_D	多路复用器, 可级联 (DOUBLE_INTEGER型)	X	X
MUX8_I	多路复用器, 可级联 (INTEGER型)	X	X
MVS	滑动平均值发生器	X	X
N2_R	转换16位定点格式 (N2) 为REAL	X	X
N4_R	转换32位定点格式 (N4) 为REAL	X	X
NAND	逻辑AND运算 (BOOL型)	X	X
NCM	数字比较器 (REAL型)	X	X
NCM_D	数字比较器 (DOUBLE_INTEGER型)	X	X
NCM_I	数字比较器 (INTEGER型)	X	X
NOP1	预留功能块 (REAL型)	X	X
NOP1_B	预留功能块 (BOOL型)	X	X
NOP1_D	预留功能块 (DOUBLE_INTEGER型)	X	X

功能块	说明	SIMOTION	SINAMICS
NOP1_I	预留功能块 (INT型)	X	X
NOP8	预留功能块 (REAL型)	X	X
NOP8_B	预留功能块 (BOOL型)	X	X
NOP8_D	预留功能块 (DOUBLE INTEGER型)	X	X
NOP8_I	预留功能块 (INTEGERT型)	X	X
NOR	逻辑OR运算 (BOOL型)	X	X
NOT	反相器 (BOOL型)	X	X
NOT_W	状态字反相器 (WORD型)	X	
NSW	数字切换开关 (REAL型)	X	X
NSW_D	数字切换开关 (DOUBLE INTEGER型)	X	X
NSW_I	数字切换开关 (INTEGER型)	X	X
OCA	软件凸轮控制器	X	X
OR	逻辑OR运算 (BOOL型)	X	X
OR_W	逻辑OR运算 (WORD型)	X	
PC	P作用控制器	X	X
PCL	脉冲缩短设备 (BOOL型)	X	X
PDE	接通延迟设备 (BOOL型)	X	X
PDF	断开延迟设备 (BOOL型)	X	X
PIC	PI控制器	X	X
PLI20	多段线, 20个断点	X	X
PST	脉冲展宽功能块 (BOOL型)	X	X
PT1	延迟元素	X	X
R_D	REAL到DOUBLE INTEGER的转换器	X	X
R_DW	以DWORD进行的位串传递	X	X
R_I	REAL到INTEGER的转换器	X	X
R_LR	REAL到LONG REAL的转换器	X	
R_N2	转换REAL为16位定点格式 (N2)	X	X
R_N4	转换REAL为32位定点格式 (N4)	X	X
R_SI	REAL到SHORT INTEGER的转换器	X	
R_UD	REAL到UNSIGNED DOUBLE INTEGER的转换器	X	X
R_UI	REAL到UNSIGNED INTEGER的转换器	X	X
R_US	REAL到UNSIGNED SHORT INTEGER的转换器	X	X
RAA	复位所有消息	V4.3	
RDA	读出消息	V4.3	
RDAA	读出所有消息	V4.3	
RDP	读取驱动参数 (REAL型)		X
RDP_D	读取驱动参数 (DOUBLE INTEGER型)		X
RDP_I	读取驱动参数 (INTEGER型)		X
RDP_UD	读取驱动参数 (UNSIGNED DOUBLE INTEGER型)		X
RDP_UI	读取驱动参数 (UNSIGNED INTEGER型)		X
RDP_US	读取驱动参数 (UNSIGNED SHORT INTEGER型)		X

功能块	说明	SIMOTION	SINAMICS
RGE	斜坡函数发生器	X	X
RGJ	带加加速度限制的斜坡函数发生器	X	X
RMDP	从控制器中读取驱动参数	V4.2	
RSR	复位优先的RS触发器 (BOOL型)	X	X
RSS	设定优先的RS触发器 (BOOL型)	X	X
SAH	采样与保持 (REAL型)		X
SAH_B	采样与保持 (BOOL型)		X
SAH_BY	采样与保持 (BYTE型)		X
SAH_D	采样与保持 (DOUBLE INTEGER型)		X
SAH_I	采样与保持 (INTEGER型)		X
SAV	数值缓冲 (REAL型)	X	X
SAV_BY	数值缓冲 (BYTE型)	X	X
SAV_D	数值缓冲 (DOUBLE INTEGER型)	X	X
SAV_I	数值缓冲 (INTEGER型)	X	X
SH	移位功能块 (WORD型)	X	
SH_DW	移位功能块 (DWORD型)	X	X
SI_D	SHORT INTEGER 到DOUBLE_INTEGER的转换器	X	
SI_I	SHORT INTEGER 到INTEGER的转换器	X	
SI_R	SHORT INTEGER到REAL的转换器	X	
SI_UD	SHORT INTEGER到UNSIGNED DOUBLE INTEGER的转换器	X	
SI_UI	SHORT INTEGER到UNSIGNED INTEGER的转换器	X	
SII	反相器	X	X
SIN	正弦功能块	X	V4.4
SQR	平方根提取器	X	V4.4
SRA	触发/复位消息	V4.3	
STM	故障/告警触发器		X
SUB	减法器 (REAL型)	X	X
SUB_D	减法器 (DOUBLE INTEGER型)	X	X
SUB_I	减法器 (INTEGER型)	X	X
TAN	正切	X	
TRK	校正/存储元件 (REAL型)	X	X
TRK_D	校正/存储元件 (DOUBLE INTEGER型)	X	X
TTCU	绕组特性	X	X
UD_I	UNSIGNED DOUBLE INTEGER到INTEGER的转换器	X	X
UD_R	UNSIGNED DOUBLE INTEGER到REAL的转换器	X	X
UD_SI	UNSIGNED DOUBLE INTEGER到SHORT INTEGER的转换器	X	
UI_D	UNSIGNED INTEGER 到DOUBLE_INTEGER的转换器	X	X
UI_R	UNSIGNED INTEGER到REAL的转换器	X	X
UI_SI	UNSIGNED INTEGER到SHORT INTEGER的转换器	X	
US_D	UNSIGNED SHORT INTEGER 到DOUBLE_INTEGER的转换器	X	X
US_I	UNSIGNED SHORT INTEGER 到INTEGER的转换器	X	X

功能块	说明	SIMOTION	SINAMICS
US_R	UNSIGNED SHORT INTEGER到REAL的转换器	X	X
W_B	状态字到16个二进制变量的转换器	X	X
W_BY	状态字到状态字节的转换器	X	X
W_DW	状态字到状态双字的转换器	X	X
WBG	抖动发生器	X	X
WMDP	从控制器中写入驱动参数	V4.2	
WRP	写入驱动参数 (REAL型)		X
WRP_D	写入驱动参数 (DOUBLE INTEGER型)		X
WRP_I	写入驱动参数 (INTEGER型)		X
WRP_UD	写入驱动参数 (UNSIGNED DOUBLE INTEGER型)		X
WRP_UI	写入驱动参数 (UNSIGNED INTEGER型)		X
WRP_US	写入驱动参数 (UNSIGNED SHORT INTEGER型)		X
XOR	逻辑XOR运算 (BOOL型)	X	X
XOR_W	逻辑XOR运算 (WORD型)	X	

B.1 消息

Note:

本章中所介绍的消息仅适用于SINAMICS版本: 4.50.04

关于SIMOTION版本, 可在SIMOTION参考文献列表中找到相关信息。可变的

F51000	DCC: Logon of the run-time group with sampling time management rejected
Drive object:	All objects
Reaction:	NONE
Acknowledge- ment:	IMMEDIATELY
Cause:	The OA application "Drive Control Chart" (DCC) attempted to log on a sampling time that cannot be implemented with the sampling time management of the basic SINAMICS system. The logon was rejected.
Remedy:	Try to assign this run-time group another fixed or free run-time group. The assignment is set in STARTER in the context menu of the DCC chart via sampling times. Then compile the chart and download it again into the drive unit.

F51001	DCC: No further hardware sampling times available
Drive object:	All objects
Reaction:	NONE
Acknowledge- ment:	IMMEDIATELY
Cause:	The drive unit can no longer provide any additional hardware sampling times, whose sampling time deviates from the sampling times already logged on.
Remedy:	The fault can be immediately acknowledged, as the system run-time group 0 (corresponds to "Do not calculate") was assigned in p21000[x]. Fault value (r0949, interpret hexadecimal): yyyyxxxx hex yyyy: The upper 16 bits of the fault value specify the number of the drive object. xxxx: The lower 16 bits specify the index of the run-time group in p21000. Note: In window "Set run-time groups" in the context menu of the chart, p21000[0] is the topmost entry and p21000[9] the lowest entry. The current assignment of hardware sampling times can be read-out in r21008.

F51004 DCC: Sampling time of the free run-time group differs at download

Drive object: All objects

Reaction: NONE

Acknowledge-ment: IMMEDIATELY

Cause: In the STARTER/SCOUT project that was downloaded, the hardware sampling time of a free run-time group ($1 \leq p21000[i] \leq 256$) was set to a value that was either too low or too high. The sampling time must be between 1 ms and the value ($r21003 - r21002$).
 If the sampling time of the selected free run-time group is < 1 ms, the equivalent value of 1 ms is used.
 If the value $\geq r21003$, then the sampling time is set to the next higher or the same software sampling time $\geq r21003$.
 The free run-time group involved is assigned as a minimum to one block.
 If this fault still occurs during download after the selection in $p21000[i]$ in the project has been corrected, please check which run-time group is involved on the basis of the fault value (r0949). Only one F51004 fault is signaled at a time, even if several run-time groups have been incorrectly parameterized in $p21000[]$.
 Fault value (r0949, decimal interpretation):
 Number of the $p21000$ index of the run-time group where the sampling time was incorrectly set.
 Number of the run-time group = fault value + 1
 Note:
 With SIMOTION D410, $r21003$ (unlike all the other Control Units) is automatically set the same as the PROFIBUS sampling time.

Remedy: Correctly set the sampling time of the run-time group or remove all of the blocks from the run-time group.

F51005 DCC: Sampling time of the fixed run-time group differs online

Drive object: All objects

Reaction: NONE

Acknowledge-ment: IMMEDIATELY

Cause: Generally, the sampling times of the fixed run-time groups correspond to the sampling times of the associated system function (e.g. the sampling time of the fixed run-time group "BEFORE speed controller" generally corresponds to the sampling of the speed controller $p0115[1]$).
 The sampling time of a system function online was set to a lower value (e.g. with $p0112$, $p0115$, $p0799$, $p4099$) than the smallest permissible sampling time that is allowed for the fixed run-time group belonging to this system function (1 ms). The sampling time is set to 1 ms. The fixed run-time group involved is assigned as a minimum to one block.
 Fault value (r0949, decimal interpretation):
 Number of the $p21000$ index of the run-time group where the sampling time was incorrectly set.
 Number of the run-time group = fault value + 1

Remedy: Using parameter $p0112$ or $p0115$, increase the sampling time of the system function to the minimum permissible sampling time for the run-time groups of 1 ms or remove all of the blocks from the run-time group.

F51006	DCC: Sampling time of the fixed run-time group differs at download
Drive object:	All objects
Reaction:	NONE
Acknowledge- ment:	IMMEDIATELY
Cause:	Generally, the sampling times of the fixed run-time groups correspond to the sampling times of the associated system function (e.g. the sampling time of the fixed run-time group "BEFORE speed controller" generally corresponds to the sampling of the speed controller p0115[1]). During a download, the sampling time of a system function was set to a lower value (p0112, p0115) than the smallest permissible sampling time that is allowed for the fixed run-time group belonging to this system function (1 ms). The sampling time is set to the smallest possible value (r21002 on the drive object). Fault value (r0949, decimal interpretation): Number of the p21000 index of the run-time group where the sampling time was incorrectly set. Number of the run-time group = fault value + 1
Remedy:	Using parameter p0112 or p0115, increase the sampling time of the system function to the minimum permissible sampling time for the run-time groups of 1 ms or remove all of the blocks from the run-time group.

F51008	DCC: No NVRAM available
Drive object:	All objects
Reaction:	OFF2
Acknowledge- ment:	IMMEDIATELY
Cause:	The DCC project contains at least one block that requires remanent memory from the basic SINAMICS system (e.g. SAV, SAV_BY, SAV_D, SAV_I). The request for remanent memory was rejected by the basic SINAMICS system. Fault value (r0949, decimal interpretation): 0: There is no more free remanent memory available on the drive unit. 1: The EPROM data of the drive unit indicates that there is no remanent memory on the module.
Remedy:	For fault value = 0: - Deactivate other applications on the drive unit that use remanent memory. - Do not use blocks that require remanent memory in your DCC charts. For fault value = 1: - For modules D425 or D435, use hardware version D or higher. Note: You can read out the hardware version using SCOUT in online mode under Target system --> Device diagnostics --> tab "General" in the lower window, 3rd column in the line of the CPU.

F51009	DCC: Project data and block library are incompatible
Drive object:	All objects
Reaction:	OFF2
Acknowledge- ment:	IMMEDIATELY
Cause:	The block library and the saved or downloaded project data are incompatible.
Remedy:	Make sure that the block library and project data match. - Update the block library in SINAMICS by downloading the technology package. or - Update the project data in the DCC Editor by importing the correct block library.

B.1 消息

A51032 DCC: Internal measurement active
 Drive object: All objects
 Reaction: NONE
 Acknowledge-
 ment: NONE
 Cause: A Siemens internal measurement has been activated.
 Remedy: Carry out a POWER ON (power off/on) for the Control Unit involved.

F51050 DCC: Fault initiated by block STM
 Drive object: All objects
 Reaction: Infeed: OFF2 (NONE, OFF1)
 Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Acknowledge-
 ment: IMMEDIATELY (POWER ON)
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message
 via the block "Set Message" (STM).
 Fault value (r0949, decimal interpretation):
 The configured message value is displayed in r0949.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the
 corresponding project documentation.

F51051 DCC: Fault initiated by block STM
 Drive object: All objects
 Reaction: Infeed: OFF2 (NONE, OFF1)
 Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Acknowledge-
 ment: IMMEDIATELY (POWER ON)
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message
 via the block "Set Message" (STM).
 Fault value (r0949, decimal interpretation):
 The configured message value is displayed in r0949.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the
 corresponding project documentation.

F51052 DCC: Fault initiated by block STM
 Drive object: All objects
 Reaction: Infeed: OFF2 (NONE, OFF1)
 Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Acknowledge-
 ment: IMMEDIATELY (POWER ON)
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message
 via the block "Set Message" (STM).
 Fault value (r0949, decimal interpretation):
 The configured message value is displayed in r0949.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the
 corresponding project documentation.

F51053	DCC: Fault initiated by block STM
Drive object:	All objects
Reaction:	Infeed: OFF2 (NONE, OFF1) Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2) Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
Acknowledge- ment:	IMMEDIATELY (POWER ON)
Cause:	The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM). Fault value (r0949, decimal interpretation): The configured message value is displayed in r0949.
Remedy:	This message was configured with "Drive Control Chart" (DCC). The cause and remedy depend on the project and should be described in the corresponding project documentation.

F51054	DCC: Fault initiated by block STM
Drive object:	All objects
Reaction:	Infeed: OFF2 (NONE, OFF1) Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2) Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
Acknowledge- ment:	IMMEDIATELY (POWER ON)
Cause:	The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM). Fault value (r0949, decimal interpretation): The configured message value is displayed in r0949.
Remedy:	This message was configured with "Drive Control Chart" (DCC). The cause and remedy depend on the project and should be described in the corresponding project documentation.

F51055	DCC: Fault initiated by block STM
Drive object:	All objects
Reaction:	Infeed: OFF2 (NONE, OFF1) Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2) Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
Acknowledge- ment:	IMMEDIATELY (POWER ON)
Cause:	The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM). Fault value (r0949, decimal interpretation): The configured message value is displayed in r0949.
Remedy:	This message was configured with "Drive Control Chart" (DCC). The cause and remedy depend on the project and should be described in the corresponding project documentation.

B.1 消息

F51056 DCC: Fault initiated by block STM
 Drive object: All objects
 Reaction: Infeed: OFF2 (NONE, OFF1)
 Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Acknowledge- IMMEDIATELY (POWER ON)
 ment:
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message
 via the block "Set Message" (STM).
 Fault value (r0949, decimal interpretation):
 The configured message value is displayed in r0949.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the
 corresponding project documentation.

F51057 DCC: Fault initiated by block STM
 Drive object: All objects
 Reaction: Infeed: OFF2 (NONE, OFF1)
 Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Acknowledge- IMMEDIATELY (POWER ON)
 ment:
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message
 via the block "Set Message" (STM).
 Fault value (r0949, decimal interpretation):
 The configured message value is displayed in r0949.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the
 corresponding project documentation.

F51058 DCC: Fault initiated by block STM
 Drive object: All objects
 Reaction: Infeed: OFF2 (NONE, OFF1)
 Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
 Acknowledge- IMMEDIATELY (POWER ON)
 ment:
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message
 via the block "Set Message" (STM).
 Fault value (r0949, decimal interpretation):
 The configured message value is displayed in r0949.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the
 corresponding project documentation.

F51059	DCC: Fault initiated by block STM
Drive object:	All objects
Reaction:	Infeed: OFF2 (NONE, OFF1) Servo: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2) Vector: OFF2 (ENCODER, IASC/DCBRAKE, NONE, OFF1, OFF3, STOP1, STOP2)
Acknowledgement:	IMMEDIATELY (POWER ON)
Cause:	The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM). Fault value (r0949, decimal interpretation): The configured message value is displayed in r0949.
Remedy:	This message was configured with "Drive Control Chart" (DCC). The cause and remedy depend on the project and should be described in the corresponding project documentation.

A51060	DCC: Alarm initiated by block STM
Drive object:	All objects
Reaction:	NONE
Acknowledgement:	NONE
Cause:	The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM). Alarm value (r2124, interpret decimal): The configured message value is displayed in r2124.
Remedy:	This message was configured with "Drive Control Chart" (DCC). The cause and remedy depend on the project and should be described in the corresponding project documentation.

A51061	DCC: Alarm initiated by block STM
Drive object:	All objects
Reaction:	NONE
Acknowledgement:	NONE
Cause:	The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM). Alarm value (r2124, interpret decimal): The configured message value is displayed in r2124.
Remedy:	This message was configured with "Drive Control Chart" (DCC). The cause and remedy depend on the project and should be described in the corresponding project documentation.

A51062	DCC: Alarm initiated by block STM
Drive object:	All objects
Reaction:	NONE
Acknowledgement:	NONE
Cause:	The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM). Alarm value (r2124, interpret decimal): The configured message value is displayed in r2124.
Remedy:	This message was configured with "Drive Control Chart" (DCC). The cause and remedy depend on the project and should be described in the corresponding project documentation.

B.1 消息

A51063 DCC: Alarm initiated by block STM
 Drive object: All objects
 Reaction: NONE
 Acknowledge-ment: NONE
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM).
 Alarm value (r2124, interpret decimal):
 The configured message value is displayed in r2124.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the corresponding project documentation.

A51064 DCC: Alarm initiated by block STM
 Drive object: All objects
 Reaction: NONE
 Acknowledge-ment: NONE
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM).
 Alarm value (r2124, interpret decimal):
 The configured message value is displayed in r2124.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the corresponding project documentation.

A51065 DCC: Alarm initiated by block STM
 Drive object: All objects
 Reaction: NONE
 Acknowledge-ment: NONE
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM).
 Alarm value (r2124, interpret decimal):
 The configured message value is displayed in r2124.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the corresponding project documentation.

A51066 DCC: Alarm initiated by block STM
 Drive object: All objects
 Reaction: NONE
 Acknowledge-ment: NONE
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message via the block "Set Message" (STM).
 Alarm value (r2124, interpret decimal):
 The configured message value is displayed in r2124.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the corresponding project documentation.

A51067 DCC: Alarm initiated by block STM
 Drive object: All objects
 Reaction: NONE
 Acknowledge-
 ment: NONE
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message
 via the block "Set Message" (STM).
 Alarm value (r2124, interpret decimal):
 The configured message value is displayed in r2124.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the
 corresponding project documentation.

A51068 DCC: Alarm initiated by block STM
 Drive object: All objects
 Reaction: NONE
 Acknowledge-
 ment: NONE
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message
 via the block "Set Message" (STM).
 Alarm value (r2124, interpret decimal):
 The configured message value is displayed in r2124.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the
 corresponding project documentation.

A51069 DCC: Alarm initiated by block STM
 Drive object: All objects
 Reaction: NONE
 Acknowledge-
 ment: NONE
 Cause: The OA application "Drive Control Chart" (DCC) has initiated this message
 via the block "Set Message" (STM).
 Alarm value (r2124, interpret decimal):
 The configured message value is displayed in r2124.
 Remedy: This message was configured with "Drive Control Chart" (DCC).
 The cause and remedy depend on the project and should be described in the
 corresponding project documentation.

B. 2 参数

Note:

本章中所介绍的参数仅适用于SINAMICS版本 4.50.04

关于SIMOTION版本，可在SIMOTION参考文献列表中找到相关信息。可变的

p21000[0..9]	Run-time group properties / RTG property		
A_INF	Changeable: T	Calculated: -	Access level: 1
	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4004	[0] 0

Description: Allocates properties to run-time groups 1 to 10.
This property comprises the sampling time and, for p21000[x] >= 2000, the instant of the call within the sampling time.
The index x + 1 of p21000 corresponds to the number of the run-time group:

- p21000[0] is used to set the property of the run-time group 1
- ...
- p21000[9] is used to set the property of the run-time group 10

Values:	0:	Do not calculate run-time group
	1:	T = 1 * r21002
	2:	T = 2 * r21002
	3:	T = 3 * r21002
	4:	T = 4 * r21002
	5:	T = 5 * r21002
	6:	T = 6 * r21002
	7:	T = 7 * r21002
	8:	T = 8 * r21002
	9:	T = 9 * r21002
	10:	T = 10 * r21002
	11:	T = 11 * r21002
	12:	T = 12 * r21002
	13:	T = 13 * r21002
	14:	T = 14 * r21002
	15:	T = 15 * r21002
	16:	T = 16 * r21002
	17:	T = 17 * r21002
	18:	T = 18 * r21002
	19:	T = 19 * r21002
	20:	T = 20 * r21002
	21:	T = 21 * r21002
	22:	T = 22 * r21002
	23:	T = 23 * r21002
	24:	T = 24 * r21002
	25:	T = 25 * r21002
	26:	T = 26 * r21002
	27:	T = 27 * r21002
	28:	T = 28 * r21002
	29:	T = 29 * r21002
	30:	T = 30 * r21002
	31:	T = 31 * r21002
	32:	T = 32 * r21002
	33:	T = 33 * r21002
	34:	T = 34 * r21002

35: T = 35 * r21002
36: T = 36 * r21002
37: T = 37 * r21002
38: T = 38 * r21002
39: T = 39 * r21002
40: T = 40 * r21002
41: T = 41 * r21002
42: T = 42 * r21002
43: T = 43 * r21002
44: T = 44 * r21002
45: T = 45 * r21002
46: T = 46 * r21002
47: T = 47 * r21002
48: T = 48 * r21002
49: T = 49 * r21002
50: T = 50 * r21002
51: T = 51 * r21002
52: T = 52 * r21002
53: T = 53 * r21002
54: T = 54 * r21002
55: T = 55 * r21002
56: T = 56 * r21002
57: T = 57 * r21002
58: T = 58 * r21002
59: T = 59 * r21002
60: T = 60 * r21002
61: T = 61 * r21002
62: T = 62 * r21002
63: T = 63 * r21002
64: T = 64 * r21002
65: T = 65 * r21002
66: T = 66 * r21002
67: T = 67 * r21002
68: T = 68 * r21002
69: T = 69 * r21002
70: T = 70 * r21002
71: T = 71 * r21002
72: T = 72 * r21002
73: T = 73 * r21002
74: T = 74 * r21002
75: T = 75 * r21002
76: T = 76 * r21002
77: T = 77 * r21002
78: T = 78 * r21002
79: T = 79 * r21002
80: T = 80 * r21002
81: T = 81 * r21002
82: T = 82 * r21002
83: T = 83 * r21002
84: T = 84 * r21002
85: T = 85 * r21002
86: T = 86 * r21002
87: T = 87 * r21002
88: T = 88 * r21002
89: T = 89 * r21002
90: T = 90 * r21002
91: T = 91 * r21002
92: T = 92 * r21002
93: T = 93 * r21002
94: T = 94 * r21002

95: T = 95 * r21002
96: T = 96 * r21002
97: T = 97 * r21002
98: T = 98 * r21002
99: T = 99 * r21002
100: T = 100 * r21002
101: T = 101 * r21002
102: T = 102 * r21002
103: T = 103 * r21002
104: T = 104 * r21002
105: T = 105 * r21002
106: T = 106 * r21002
107: T = 107 * r21002
108: T = 108 * r21002
109: T = 109 * r21002
110: T = 110 * r21002
111: T = 111 * r21002
112: T = 112 * r21002
113: T = 113 * r21002
114: T = 114 * r21002
115: T = 115 * r21002
116: T = 116 * r21002
117: T = 117 * r21002
118: T = 118 * r21002
119: T = 119 * r21002
120: T = 120 * r21002
121: T = 121 * r21002
122: T = 122 * r21002
123: T = 123 * r21002
124: T = 124 * r21002
125: T = 125 * r21002
126: T = 126 * r21002
127: T = 127 * r21002
128: T = 128 * r21002
129: T = 129 * r21002
130: T = 130 * r21002
131: T = 131 * r21002
132: T = 132 * r21002
133: T = 133 * r21002
134: T = 134 * r21002
135: T = 135 * r21002
136: T = 136 * r21002
137: T = 137 * r21002
138: T = 138 * r21002
139: T = 139 * r21002
140: T = 140 * r21002
141: T = 141 * r21002
142: T = 142 * r21002
143: T = 143 * r21002
144: T = 144 * r21002
145: T = 145 * r21002
146: T = 146 * r21002
147: T = 147 * r21002
148: T = 148 * r21002
149: T = 149 * r21002
150: T = 150 * r21002
151: T = 151 * r21002
152: T = 152 * r21002
153: T = 153 * r21002
154: T = 154 * r21002

155: T = 155 * r21002
156: T = 156 * r21002
157: T = 157 * r21002
158: T = 158 * r21002
159: T = 159 * r21002
160: T = 160 * r21002
161: T = 161 * r21002
162: T = 162 * r21002
163: T = 163 * r21002
164: T = 164 * r21002
165: T = 165 * r21002
166: T = 166 * r21002
167: T = 167 * r21002
168: T = 168 * r21002
169: T = 169 * r21002
170: T = 170 * r21002
171: T = 171 * r21002
172: T = 172 * r21002
173: T = 173 * r21002
174: T = 174 * r21002
175: T = 175 * r21002
176: T = 176 * r21002
177: T = 177 * r21002
178: T = 178 * r21002
179: T = 179 * r21002
180: T = 180 * r21002
181: T = 181 * r21002
182: T = 182 * r21002
183: T = 183 * r21002
184: T = 184 * r21002
185: T = 185 * r21002
186: T = 186 * r21002
187: T = 187 * r21002
188: T = 188 * r21002
189: T = 189 * r21002
190: T = 190 * r21002
191: T = 191 * r21002
192: T = 192 * r21002
193: T = 193 * r21002
194: T = 194 * r21002
195: T = 195 * r21002
196: T = 196 * r21002
197: T = 197 * r21002
198: T = 198 * r21002
199: T = 199 * r21002
200: T = 200 * r21002
201: T = 201 * r21002
202: T = 202 * r21002
203: T = 203 * r21002
204: T = 204 * r21002
205: T = 205 * r21002
206: T = 206 * r21002
207: T = 207 * r21002
208: T = 208 * r21002
209: T = 209 * r21002
210: T = 210 * r21002
211: T = 211 * r21002
212: T = 212 * r21002
213: T = 213 * r21002
214: T = 214 * r21002

215: T = 215 * r21002
 216: T = 216 * r21002
 217: T = 217 * r21002
 218: T = 218 * r21002
 219: T = 219 * r21002
 220: T = 220 * r21002
 221: T = 221 * r21002
 222: T = 222 * r21002
 223: T = 223 * r21002
 224: T = 224 * r21002
 225: T = 225 * r21002
 226: T = 226 * r21002
 227: T = 227 * r21002
 228: T = 228 * r21002
 229: T = 229 * r21002
 230: T = 230 * r21002
 231: T = 231 * r21002
 232: T = 232 * r21002
 233: T = 233 * r21002
 234: T = 234 * r21002
 235: T = 235 * r21002
 236: T = 236 * r21002
 237: T = 237 * r21002
 238: T = 238 * r21002
 239: T = 239 * r21002
 240: T = 240 * r21002
 241: T = 241 * r21002
 242: T = 242 * r21002
 243: T = 243 * r21002
 244: T = 244 * r21002
 245: T = 245 * r21002
 246: T = 246 * r21002
 247: T = 247 * r21002
 248: T = 248 * r21002
 249: T = 249 * r21002
 250: T = 250 * r21002
 251: T = 251 * r21002
 252: T = 252 * r21002
 253: T = 253 * r21002
 254: T = 254 * r21002
 255: T = 255 * r21002
 256: T = 256 * r21002
 1001: T = 1 * r21003
 1002: T = 2 * r21003
 1003: T = 3 * r21003
 1004: T = 4 * r21003
 1005: T = 5 * r21003
 1006: T = 6 * r21003
 1008: T = 8 * r21003
 1010: T = 10 * r21003
 1012: T = 12 * r21003
 1016: T = 16 * r21003
 1020: T = 20 * r21003
 1024: T = 24 * r21003
 1032: T = 32 * r21003
 1040: T = 40 * r21003
 1048: T = 48 * r21003
 1064: T = 64 * r21003
 1080: T = 80 * r21003
 1096: T = 96 * r21003

	3003:	BEFORE speed setpoint channel
	4000:	Receive AFTER IF1 PROFIdrive PZD
	4001:	Send BEFORE IF1 PROFIdrive PZD
	4004:	Receive AFTER IF1 PROFIdrive flexible PZD
Index:	[0]:	Run-time group 1
	[1]:	Run-time group 2
	[2]:	Run-time group 3
	[3]:	Run-time group 4
	[4]:	Run-time group 5
	[5]:	Run-time group 6
	[6]:	Run-time group 7
	[7]:	Run-time group 8
	[8]:	Run-time group 9
	[9]:	Run-time group 10
Caution:	The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.	
Note:	Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a setpoint channel has not been configured ($p0108.8 = 0$), the calculation is made before function chart 3095.	

p21000[0...9]	Run-time group properties / RTG property		
CU_DC	Changeable: T	Calculated: -	Access level: 1
	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4005	[0] 0

Description: Allocates properties to run-time groups 1 to 10.
This property comprises the sampling time and, for $p21000[x] \geq 2000$, the instant of the call within the sampling time.
The index $x + 1$ of $p21000$ corresponds to the number of the run-time group:

- $p21000[0]$ is used to set the property of the run-time group 1
- ...
- $p21000[9]$ is used to set the property of the run-time group 10

Values:	0:	Do not calculate run-time group
	1:	$T = 1 * r21002$
	2:	$T = 2 * r21002$
	3:	$T = 3 * r21002$
	4:	$T = 4 * r21002$
	5:	$T = 5 * r21002$
	6:	$T = 6 * r21002$
	7:	$T = 7 * r21002$
	8:	$T = 8 * r21002$

9: T = 9 * r21002
 10: T = 10 * r21002
 11: T = 11 * r21002
 12: T = 12 * r21002
 13: T = 13 * r21002
 14: T = 14 * r21002
 15: T = 15 * r21002
 16: T = 16 * r21002
 17: T = 17 * r21002
 18: T = 18 * r21002
 19: T = 19 * r21002
 20: T = 20 * r21002
 21: T = 21 * r21002
 22: T = 22 * r21002
 23: T = 23 * r21002
 24: T = 24 * r21002
 25: T = 25 * r21002
 26: T = 26 * r21002
 27: T = 27 * r21002
 28: T = 28 * r21002
 29: T = 29 * r21002
 30: T = 30 * r21002
 31: T = 31 * r21002
 32: T = 32 * r21002
 33: T = 33 * r21002
 34: T = 34 * r21002
 35: T = 35 * r21002
 36: T = 36 * r21002
 37: T = 37 * r21002
 38: T = 38 * r21002
 39: T = 39 * r21002
 40: T = 40 * r21002
 41: T = 41 * r21002
 42: T = 42 * r21002
 43: T = 43 * r21002
 44: T = 44 * r21002
 45: T = 45 * r21002
 46: T = 46 * r21002
 47: T = 47 * r21002
 48: T = 48 * r21002
 49: T = 49 * r21002
 50: T = 50 * r21002
 51: T = 51 * r21002
 52: T = 52 * r21002
 53: T = 53 * r21002
 54: T = 54 * r21002
 55: T = 55 * r21002
 56: T = 56 * r21002
 57: T = 57 * r21002
 58: T = 58 * r21002
 59: T = 59 * r21002
 60: T = 60 * r21002
 61: T = 61 * r21002
 62: T = 62 * r21002
 63: T = 63 * r21002
 64: T = 64 * r21002
 65: T = 65 * r21002
 66: T = 66 * r21002
 67: T = 67 * r21002
 68: T = 68 * r21002

69: T = 69 * r21002
70: T = 70 * r21002
71: T = 71 * r21002
72: T = 72 * r21002
73: T = 73 * r21002
74: T = 74 * r21002
75: T = 75 * r21002
76: T = 76 * r21002
77: T = 77 * r21002
78: T = 78 * r21002
79: T = 79 * r21002
80: T = 80 * r21002
81: T = 81 * r21002
82: T = 82 * r21002
83: T = 83 * r21002
84: T = 84 * r21002
85: T = 85 * r21002
86: T = 86 * r21002
87: T = 87 * r21002
88: T = 88 * r21002
89: T = 89 * r21002
90: T = 90 * r21002
91: T = 91 * r21002
92: T = 92 * r21002
93: T = 93 * r21002
94: T = 94 * r21002
95: T = 95 * r21002
96: T = 96 * r21002
97: T = 97 * r21002
98: T = 98 * r21002
99: T = 99 * r21002
100: T = 100 * r21002
101: T = 101 * r21002
102: T = 102 * r21002
103: T = 103 * r21002
104: T = 104 * r21002
105: T = 105 * r21002
106: T = 106 * r21002
107: T = 107 * r21002
108: T = 108 * r21002
109: T = 109 * r21002
110: T = 110 * r21002
111: T = 111 * r21002
112: T = 112 * r21002
113: T = 113 * r21002
114: T = 114 * r21002
115: T = 115 * r21002
116: T = 116 * r21002
117: T = 117 * r21002
118: T = 118 * r21002
119: T = 119 * r21002
120: T = 120 * r21002
121: T = 121 * r21002
122: T = 122 * r21002
123: T = 123 * r21002
124: T = 124 * r21002
125: T = 125 * r21002
126: T = 126 * r21002
127: T = 127 * r21002
128: T = 128 * r21002

129: T = 129 * r21002
130: T = 130 * r21002
131: T = 131 * r21002
132: T = 132 * r21002
133: T = 133 * r21002
134: T = 134 * r21002
135: T = 135 * r21002
136: T = 136 * r21002
137: T = 137 * r21002
138: T = 138 * r21002
139: T = 139 * r21002
140: T = 140 * r21002
141: T = 141 * r21002
142: T = 142 * r21002
143: T = 143 * r21002
144: T = 144 * r21002
145: T = 145 * r21002
146: T = 146 * r21002
147: T = 147 * r21002
148: T = 148 * r21002
149: T = 149 * r21002
150: T = 150 * r21002
151: T = 151 * r21002
152: T = 152 * r21002
153: T = 153 * r21002
154: T = 154 * r21002
155: T = 155 * r21002
156: T = 156 * r21002
157: T = 157 * r21002
158: T = 158 * r21002
159: T = 159 * r21002
160: T = 160 * r21002
161: T = 161 * r21002
162: T = 162 * r21002
163: T = 163 * r21002
164: T = 164 * r21002
165: T = 165 * r21002
166: T = 166 * r21002
167: T = 167 * r21002
168: T = 168 * r21002
169: T = 169 * r21002
170: T = 170 * r21002
171: T = 171 * r21002
172: T = 172 * r21002
173: T = 173 * r21002
174: T = 174 * r21002
175: T = 175 * r21002
176: T = 176 * r21002
177: T = 177 * r21002
178: T = 178 * r21002
179: T = 179 * r21002
180: T = 180 * r21002
181: T = 181 * r21002
182: T = 182 * r21002
183: T = 183 * r21002
184: T = 184 * r21002
185: T = 185 * r21002
186: T = 186 * r21002
187: T = 187 * r21002
188: T = 188 * r21002

189: T = 189 * r21002
190: T = 190 * r21002
191: T = 191 * r21002
192: T = 192 * r21002
193: T = 193 * r21002
194: T = 194 * r21002
195: T = 195 * r21002
196: T = 196 * r21002
197: T = 197 * r21002
198: T = 198 * r21002
199: T = 199 * r21002
200: T = 200 * r21002
201: T = 201 * r21002
202: T = 202 * r21002
203: T = 203 * r21002
204: T = 204 * r21002
205: T = 205 * r21002
206: T = 206 * r21002
207: T = 207 * r21002
208: T = 208 * r21002
209: T = 209 * r21002
210: T = 210 * r21002
211: T = 211 * r21002
212: T = 212 * r21002
213: T = 213 * r21002
214: T = 214 * r21002
215: T = 215 * r21002
216: T = 216 * r21002
217: T = 217 * r21002
218: T = 218 * r21002
219: T = 219 * r21002
220: T = 220 * r21002
221: T = 221 * r21002
222: T = 222 * r21002
223: T = 223 * r21002
224: T = 224 * r21002
225: T = 225 * r21002
226: T = 226 * r21002
227: T = 227 * r21002
228: T = 228 * r21002
229: T = 229 * r21002
230: T = 230 * r21002
231: T = 231 * r21002
232: T = 232 * r21002
233: T = 233 * r21002
234: T = 234 * r21002
235: T = 235 * r21002
236: T = 236 * r21002
237: T = 237 * r21002
238: T = 238 * r21002
239: T = 239 * r21002
240: T = 240 * r21002
241: T = 241 * r21002
242: T = 242 * r21002
243: T = 243 * r21002
244: T = 244 * r21002
245: T = 245 * r21002
246: T = 246 * r21002
247: T = 247 * r21002
248: T = 248 * r21002

249: T = 249 * r21002
 250: T = 250 * r21002
 251: T = 251 * r21002
 252: T = 252 * r21002
 253: T = 253 * r21002
 254: T = 254 * r21002
 255: T = 255 * r21002
 256: T = 256 * r21002
 1001: T = 1 * r21003
 1002: T = 2 * r21003
 1003: T = 3 * r21003
 1004: T = 4 * r21003
 1005: T = 5 * r21003
 1006: T = 6 * r21003
 1008: T = 8 * r21003
 1010: T = 10 * r21003
 1012: T = 12 * r21003
 1016: T = 16 * r21003
 1020: T = 20 * r21003
 1024: T = 24 * r21003
 1032: T = 32 * r21003
 1040: T = 40 * r21003
 1048: T = 48 * r21003
 1064: T = 64 * r21003
 1080: T = 80 * r21003
 1096: T = 96 * r21003
 4000: Receive AFTER IF1 PROFIdrive PZD
 4001: Send BEFORE IF1 PROFIdrive PZD
 4002: Receive AFTER IF2 PZD
 4003: Send BEFORE IF2 PZD
 4004: Receive AFTER IF1 PROFIdrive flexible PZD
 4005: Receive AFTER IF2 flexible PZD

Index: [0]: Run-time group 1
 [1]: Run-time group 2
 [2]: Run-time group 3
 [3]: Run-time group 4
 [4]: Run-time group 5
 [5]: Run-time group 6
 [6]: Run-time group 7
 [7]: Run-time group 8
 [8]: Run-time group 9
 [9]: Run-time group 10

Caution: The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.

Note: Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a

setpoint channel has not been configured (p0108.8 = 0), the calculation is made before function chart 3095.

p21000[0..9]	Run-time group properties / RTG property		
CU_I, TM15BASE	Changeable: T	Calculated: -	Access level: 1
	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4004	[0] 0

Description: Allocates properties to run-time groups 1 to 10.
 This property comprises the sampling time and, for p21000[x] >= 2000, the instant of the call within the sampling time.
 The index x + 1 of p21000 corresponds to the number of the run-time group:

- p21000[0] is used to set the property of the run-time group 1
- ...
- p21000[9] is used to set the property of the run-time group 10

Values:	0:	Do not calculate run-time group
	1:	T = 1 * r21002
	2:	T = 2 * r21002
	3:	T = 3 * r21002
	4:	T = 4 * r21002
	5:	T = 5 * r21002
	6:	T = 6 * r21002
	7:	T = 7 * r21002
	8:	T = 8 * r21002
	9:	T = 9 * r21002
	10:	T = 10 * r21002
	11:	T = 11 * r21002
	12:	T = 12 * r21002
	13:	T = 13 * r21002
	14:	T = 14 * r21002
	15:	T = 15 * r21002
	16:	T = 16 * r21002
	17:	T = 17 * r21002
	18:	T = 18 * r21002
	19:	T = 19 * r21002
	20:	T = 20 * r21002
	21:	T = 21 * r21002
	22:	T = 22 * r21002
	23:	T = 23 * r21002
	24:	T = 24 * r21002
	25:	T = 25 * r21002
	26:	T = 26 * r21002
	27:	T = 27 * r21002
	28:	T = 28 * r21002
	29:	T = 29 * r21002
	30:	T = 30 * r21002
	31:	T = 31 * r21002
	32:	T = 32 * r21002
	33:	T = 33 * r21002
	34:	T = 34 * r21002
	35:	T = 35 * r21002
	36:	T = 36 * r21002
	37:	T = 37 * r21002
	38:	T = 38 * r21002
	39:	T = 39 * r21002

40: T = 40 * r21002
 41: T = 41 * r21002
 42: T = 42 * r21002
 43: T = 43 * r21002
 44: T = 44 * r21002
 45: T = 45 * r21002
 46: T = 46 * r21002
 47: T = 47 * r21002
 48: T = 48 * r21002
 49: T = 49 * r21002
 50: T = 50 * r21002
 51: T = 51 * r21002
 52: T = 52 * r21002
 53: T = 53 * r21002
 54: T = 54 * r21002
 55: T = 55 * r21002
 56: T = 56 * r21002
 57: T = 57 * r21002
 58: T = 58 * r21002
 59: T = 59 * r21002
 60: T = 60 * r21002
 61: T = 61 * r21002
 62: T = 62 * r21002
 63: T = 63 * r21002
 64: T = 64 * r21002
 65: T = 65 * r21002
 66: T = 66 * r21002
 67: T = 67 * r21002
 68: T = 68 * r21002
 69: T = 69 * r21002
 70: T = 70 * r21002
 71: T = 71 * r21002
 72: T = 72 * r21002
 73: T = 73 * r21002
 74: T = 74 * r21002
 75: T = 75 * r21002
 76: T = 76 * r21002
 77: T = 77 * r21002
 78: T = 78 * r21002
 79: T = 79 * r21002
 80: T = 80 * r21002
 81: T = 81 * r21002
 82: T = 82 * r21002
 83: T = 83 * r21002
 84: T = 84 * r21002
 85: T = 85 * r21002
 86: T = 86 * r21002
 87: T = 87 * r21002
 88: T = 88 * r21002
 89: T = 89 * r21002
 90: T = 90 * r21002
 91: T = 91 * r21002
 92: T = 92 * r21002
 93: T = 93 * r21002
 94: T = 94 * r21002
 95: T = 95 * r21002
 96: T = 96 * r21002
 97: T = 97 * r21002
 98: T = 98 * r21002
 99: T = 99 * r21002

100: T = 100 * r21002
101: T = 101 * r21002
102: T = 102 * r21002
103: T = 103 * r21002
104: T = 104 * r21002
105: T = 105 * r21002
106: T = 106 * r21002
107: T = 107 * r21002
108: T = 108 * r21002
109: T = 109 * r21002
110: T = 110 * r21002
111: T = 111 * r21002
112: T = 112 * r21002
113: T = 113 * r21002
114: T = 114 * r21002
115: T = 115 * r21002
116: T = 116 * r21002
117: T = 117 * r21002
118: T = 118 * r21002
119: T = 119 * r21002
120: T = 120 * r21002
121: T = 121 * r21002
122: T = 122 * r21002
123: T = 123 * r21002
124: T = 124 * r21002
125: T = 125 * r21002
126: T = 126 * r21002
127: T = 127 * r21002
128: T = 128 * r21002
129: T = 129 * r21002
130: T = 130 * r21002
131: T = 131 * r21002
132: T = 132 * r21002
133: T = 133 * r21002
134: T = 134 * r21002
135: T = 135 * r21002
136: T = 136 * r21002
137: T = 137 * r21002
138: T = 138 * r21002
139: T = 139 * r21002
140: T = 140 * r21002
141: T = 141 * r21002
142: T = 142 * r21002
143: T = 143 * r21002
144: T = 144 * r21002
145: T = 145 * r21002
146: T = 146 * r21002
147: T = 147 * r21002
148: T = 148 * r21002
149: T = 149 * r21002
150: T = 150 * r21002
151: T = 151 * r21002
152: T = 152 * r21002
153: T = 153 * r21002
154: T = 154 * r21002
155: T = 155 * r21002
156: T = 156 * r21002
157: T = 157 * r21002
158: T = 158 * r21002
159: T = 159 * r21002

160: T = 160 * r21002
 161: T = 161 * r21002
 162: T = 162 * r21002
 163: T = 163 * r21002
 164: T = 164 * r21002
 165: T = 165 * r21002
 166: T = 166 * r21002
 167: T = 167 * r21002
 168: T = 168 * r21002
 169: T = 169 * r21002
 170: T = 170 * r21002
 171: T = 171 * r21002
 172: T = 172 * r21002
 173: T = 173 * r21002
 174: T = 174 * r21002
 175: T = 175 * r21002
 176: T = 176 * r21002
 177: T = 177 * r21002
 178: T = 178 * r21002
 179: T = 179 * r21002
 180: T = 180 * r21002
 181: T = 181 * r21002
 182: T = 182 * r21002
 183: T = 183 * r21002
 184: T = 184 * r21002
 185: T = 185 * r21002
 186: T = 186 * r21002
 187: T = 187 * r21002
 188: T = 188 * r21002
 189: T = 189 * r21002
 190: T = 190 * r21002
 191: T = 191 * r21002
 192: T = 192 * r21002
 193: T = 193 * r21002
 194: T = 194 * r21002
 195: T = 195 * r21002
 196: T = 196 * r21002
 197: T = 197 * r21002
 198: T = 198 * r21002
 199: T = 199 * r21002
 200: T = 200 * r21002
 201: T = 201 * r21002
 202: T = 202 * r21002
 203: T = 203 * r21002
 204: T = 204 * r21002
 205: T = 205 * r21002
 206: T = 206 * r21002
 207: T = 207 * r21002
 208: T = 208 * r21002
 209: T = 209 * r21002
 210: T = 210 * r21002
 211: T = 211 * r21002
 212: T = 212 * r21002
 213: T = 213 * r21002
 214: T = 214 * r21002
 215: T = 215 * r21002
 216: T = 216 * r21002
 217: T = 217 * r21002
 218: T = 218 * r21002
 219: T = 219 * r21002

220:	T = 220 * r21002
221:	T = 221 * r21002
222:	T = 222 * r21002
223:	T = 223 * r21002
224:	T = 224 * r21002
225:	T = 225 * r21002
226:	T = 226 * r21002
227:	T = 227 * r21002
228:	T = 228 * r21002
229:	T = 229 * r21002
230:	T = 230 * r21002
231:	T = 231 * r21002
232:	T = 232 * r21002
233:	T = 233 * r21002
234:	T = 234 * r21002
235:	T = 235 * r21002
236:	T = 236 * r21002
237:	T = 237 * r21002
238:	T = 238 * r21002
239:	T = 239 * r21002
240:	T = 240 * r21002
241:	T = 241 * r21002
242:	T = 242 * r21002
243:	T = 243 * r21002
244:	T = 244 * r21002
245:	T = 245 * r21002
246:	T = 246 * r21002
247:	T = 247 * r21002
248:	T = 248 * r21002
249:	T = 249 * r21002
250:	T = 250 * r21002
251:	T = 251 * r21002
252:	T = 252 * r21002
253:	T = 253 * r21002
254:	T = 254 * r21002
255:	T = 255 * r21002
256:	T = 256 * r21002
1001:	T = 1 * r21003
1002:	T = 2 * r21003
1003:	T = 3 * r21003
1004:	T = 4 * r21003
1005:	T = 5 * r21003
1006:	T = 6 * r21003
1008:	T = 8 * r21003
1010:	T = 10 * r21003
1012:	T = 12 * r21003
1016:	T = 16 * r21003
1020:	T = 20 * r21003
1024:	T = 24 * r21003
1032:	T = 32 * r21003
1040:	T = 40 * r21003
1048:	T = 48 * r21003
1064:	T = 64 * r21003
1080:	T = 80 * r21003
1096:	T = 96 * r21003
2000:	Read-in AFTER digital inputs
2001:	Output BEFORE digital outputs
4000:	Receive AFTER IF1 PROFIdrive PZD
4001:	Send BEFORE IF1 PROFIdrive PZD
4004:	Receive AFTER IF1 PROFIdrive flexible PZD

Index: [0]: Run-time group 1
 [1]: Run-time group 2
 [2]: Run-time group 3
 [3]: Run-time group 4
 [4]: Run-time group 5
 [5]: Run-time group 6
 [6]: Run-time group 7
 [7]: Run-time group 8
 [8]: Run-time group 9
 [9]: Run-time group 10

Caution: The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.

Note: Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a setpoint channel has not been configured ($p0108.8 = 0$), the calculation is made before function chart 3095.

p21000[0...9] Run-time group properties / RTG property

CU_S, _G, _GM, _GL	Changeable: T	Calculated: -	Access level: 1
	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4005	[0] 0

Description: Allocates properties to run-time groups 1 to 10. This property comprises the sampling time and, for $p21000[x] \geq 2000$, the instant of the call within the sampling time. The index $x + 1$ of $p21000$ corresponds to the number of the run-time group:

- $p21000[0]$ is used to set the property of the run-time group 1
- ...
- $p21000[9]$ is used to set the property of the run-time group 10

Values: 0: Do not calculate run-time group
 1: $T = 1 * r21002$
 2: $T = 2 * r21002$
 3: $T = 3 * r21002$
 4: $T = 4 * r21002$
 5: $T = 5 * r21002$
 6: $T = 6 * r21002$
 7: $T = 7 * r21002$
 8: $T = 8 * r21002$
 9: $T = 9 * r21002$
 10: $T = 10 * r21002$
 11: $T = 11 * r21002$
 12: $T = 12 * r21002$

13: T = 13 * r21002
14: T = 14 * r21002
15: T = 15 * r21002
16: T = 16 * r21002
17: T = 17 * r21002
18: T = 18 * r21002
19: T = 19 * r21002
20: T = 20 * r21002
21: T = 21 * r21002
22: T = 22 * r21002
23: T = 23 * r21002
24: T = 24 * r21002
25: T = 25 * r21002
26: T = 26 * r21002
27: T = 27 * r21002
28: T = 28 * r21002
29: T = 29 * r21002
30: T = 30 * r21002
31: T = 31 * r21002
32: T = 32 * r21002
33: T = 33 * r21002
34: T = 34 * r21002
35: T = 35 * r21002
36: T = 36 * r21002
37: T = 37 * r21002
38: T = 38 * r21002
39: T = 39 * r21002
40: T = 40 * r21002
41: T = 41 * r21002
42: T = 42 * r21002
43: T = 43 * r21002
44: T = 44 * r21002
45: T = 45 * r21002
46: T = 46 * r21002
47: T = 47 * r21002
48: T = 48 * r21002
49: T = 49 * r21002
50: T = 50 * r21002
51: T = 51 * r21002
52: T = 52 * r21002
53: T = 53 * r21002
54: T = 54 * r21002
55: T = 55 * r21002
56: T = 56 * r21002
57: T = 57 * r21002
58: T = 58 * r21002
59: T = 59 * r21002
60: T = 60 * r21002
61: T = 61 * r21002
62: T = 62 * r21002
63: T = 63 * r21002
64: T = 64 * r21002
65: T = 65 * r21002
66: T = 66 * r21002
67: T = 67 * r21002
68: T = 68 * r21002
69: T = 69 * r21002
70: T = 70 * r21002
71: T = 71 * r21002
72: T = 72 * r21002

73: T = 73 * r21002
 74: T = 74 * r21002
 75: T = 75 * r21002
 76: T = 76 * r21002
 77: T = 77 * r21002
 78: T = 78 * r21002
 79: T = 79 * r21002
 80: T = 80 * r21002
 81: T = 81 * r21002
 82: T = 82 * r21002
 83: T = 83 * r21002
 84: T = 84 * r21002
 85: T = 85 * r21002
 86: T = 86 * r21002
 87: T = 87 * r21002
 88: T = 88 * r21002
 89: T = 89 * r21002
 90: T = 90 * r21002
 91: T = 91 * r21002
 92: T = 92 * r21002
 93: T = 93 * r21002
 94: T = 94 * r21002
 95: T = 95 * r21002
 96: T = 96 * r21002
 97: T = 97 * r21002
 98: T = 98 * r21002
 99: T = 99 * r21002
 100: T = 100 * r21002
 101: T = 101 * r21002
 102: T = 102 * r21002
 103: T = 103 * r21002
 104: T = 104 * r21002
 105: T = 105 * r21002
 106: T = 106 * r21002
 107: T = 107 * r21002
 108: T = 108 * r21002
 109: T = 109 * r21002
 110: T = 110 * r21002
 111: T = 111 * r21002
 112: T = 112 * r21002
 113: T = 113 * r21002
 114: T = 114 * r21002
 115: T = 115 * r21002
 116: T = 116 * r21002
 117: T = 117 * r21002
 118: T = 118 * r21002
 119: T = 119 * r21002
 120: T = 120 * r21002
 121: T = 121 * r21002
 122: T = 122 * r21002
 123: T = 123 * r21002
 124: T = 124 * r21002
 125: T = 125 * r21002
 126: T = 126 * r21002
 127: T = 127 * r21002
 128: T = 128 * r21002
 129: T = 129 * r21002
 130: T = 130 * r21002
 131: T = 131 * r21002
 132: T = 132 * r21002

133: T = 133 * r21002
134: T = 134 * r21002
135: T = 135 * r21002
136: T = 136 * r21002
137: T = 137 * r21002
138: T = 138 * r21002
139: T = 139 * r21002
140: T = 140 * r21002
141: T = 141 * r21002
142: T = 142 * r21002
143: T = 143 * r21002
144: T = 144 * r21002
145: T = 145 * r21002
146: T = 146 * r21002
147: T = 147 * r21002
148: T = 148 * r21002
149: T = 149 * r21002
150: T = 150 * r21002
151: T = 151 * r21002
152: T = 152 * r21002
153: T = 153 * r21002
154: T = 154 * r21002
155: T = 155 * r21002
156: T = 156 * r21002
157: T = 157 * r21002
158: T = 158 * r21002
159: T = 159 * r21002
160: T = 160 * r21002
161: T = 161 * r21002
162: T = 162 * r21002
163: T = 163 * r21002
164: T = 164 * r21002
165: T = 165 * r21002
166: T = 166 * r21002
167: T = 167 * r21002
168: T = 168 * r21002
169: T = 169 * r21002
170: T = 170 * r21002
171: T = 171 * r21002
172: T = 172 * r21002
173: T = 173 * r21002
174: T = 174 * r21002
175: T = 175 * r21002
176: T = 176 * r21002
177: T = 177 * r21002
178: T = 178 * r21002
179: T = 179 * r21002
180: T = 180 * r21002
181: T = 181 * r21002
182: T = 182 * r21002
183: T = 183 * r21002
184: T = 184 * r21002
185: T = 185 * r21002
186: T = 186 * r21002
187: T = 187 * r21002
188: T = 188 * r21002
189: T = 189 * r21002
190: T = 190 * r21002
191: T = 191 * r21002
192: T = 192 * r21002

193: T = 193 * r21002
194: T = 194 * r21002
195: T = 195 * r21002
196: T = 196 * r21002
197: T = 197 * r21002
198: T = 198 * r21002
199: T = 199 * r21002
200: T = 200 * r21002
201: T = 201 * r21002
202: T = 202 * r21002
203: T = 203 * r21002
204: T = 204 * r21002
205: T = 205 * r21002
206: T = 206 * r21002
207: T = 207 * r21002
208: T = 208 * r21002
209: T = 209 * r21002
210: T = 210 * r21002
211: T = 211 * r21002
212: T = 212 * r21002
213: T = 213 * r21002
214: T = 214 * r21002
215: T = 215 * r21002
216: T = 216 * r21002
217: T = 217 * r21002
218: T = 218 * r21002
219: T = 219 * r21002
220: T = 220 * r21002
221: T = 221 * r21002
222: T = 222 * r21002
223: T = 223 * r21002
224: T = 224 * r21002
225: T = 225 * r21002
226: T = 226 * r21002
227: T = 227 * r21002
228: T = 228 * r21002
229: T = 229 * r21002
230: T = 230 * r21002
231: T = 231 * r21002
232: T = 232 * r21002
233: T = 233 * r21002
234: T = 234 * r21002
235: T = 235 * r21002
236: T = 236 * r21002
237: T = 237 * r21002
238: T = 238 * r21002
239: T = 239 * r21002
240: T = 240 * r21002
241: T = 241 * r21002
242: T = 242 * r21002
243: T = 243 * r21002
244: T = 244 * r21002
245: T = 245 * r21002
246: T = 246 * r21002
247: T = 247 * r21002
248: T = 248 * r21002
249: T = 249 * r21002
250: T = 250 * r21002
251: T = 251 * r21002
252: T = 252 * r21002

253:	T = 253 * r21002
254:	T = 254 * r21002
255:	T = 255 * r21002
256:	T = 256 * r21002
1001:	T = 1 * r21003
1002:	T = 2 * r21003
1003:	T = 3 * r21003
1004:	T = 4 * r21003
1005:	T = 5 * r21003
1006:	T = 6 * r21003
1008:	T = 8 * r21003
1010:	T = 10 * r21003
1012:	T = 12 * r21003
1016:	T = 16 * r21003
1020:	T = 20 * r21003
1024:	T = 24 * r21003
1032:	T = 32 * r21003
1040:	T = 40 * r21003
1048:	T = 48 * r21003
1064:	T = 64 * r21003
1080:	T = 80 * r21003
1096:	T = 96 * r21003
2000:	Read-in AFTER digital inputs
2001:	Output BEFORE digital outputs
4000:	Receive AFTER IF1 PROFIdrive PZD
4001:	Send BEFORE IF1 PROFIdrive PZD
4002:	Receive AFTER IF2 PZD
4003:	Send BEFORE IF2 PZD
4004:	Receive AFTER IF1 PROFIdrive flexible PZD
4005:	Receive AFTER IF2 flexible PZD
Index:	[0]: Run-time group 1
	[1]: Run-time group 2
	[2]: Run-time group 3
	[3]: Run-time group 4
	[4]: Run-time group 5
	[5]: Run-time group 6
	[6]: Run-time group 7
	[7]: Run-time group 8
	[8]: Run-time group 9
	[9]: Run-time group 10
Caution:	The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.
Note:	Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a setpoint channel has not been configured ($p0108.8 = 0$), the calculation is made before function chart 3095.

p21000[0...9]	Run-time group properties / RTG property		
DC_CTRL	Changeable: T	Calculated: -	Access level: 1
	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4005	[0] 0

Description: Allocates properties to run-time groups 1 to 10.
 This property comprises the sampling time and, for p21000[x] >= 2000, the instant of the call within the sampling time.
 The index x + 1 of p21000 corresponds to the number of the run-time group:

- p21000[0] is used to set the property of the run-time group 1
- ...
- p21000[9] is used to set the property of the run-time group 10

Values:	0:	Do not calculate run-time group
	1:	T = 1 * r21002
	2:	T = 2 * r21002
	3:	T = 3 * r21002
	4:	T = 4 * r21002
	5:	T = 5 * r21002
	6:	T = 6 * r21002
	7:	T = 7 * r21002
	8:	T = 8 * r21002
	9:	T = 9 * r21002
	10:	T = 10 * r21002
	11:	T = 11 * r21002
	12:	T = 12 * r21002
	13:	T = 13 * r21002
	14:	T = 14 * r21002
	15:	T = 15 * r21002
	16:	T = 16 * r21002
	17:	T = 17 * r21002
	18:	T = 18 * r21002
	19:	T = 19 * r21002
	20:	T = 20 * r21002
	21:	T = 21 * r21002
	22:	T = 22 * r21002
	23:	T = 23 * r21002
	24:	T = 24 * r21002
	25:	T = 25 * r21002
	26:	T = 26 * r21002
	27:	T = 27 * r21002
	28:	T = 28 * r21002
	29:	T = 29 * r21002
	30:	T = 30 * r21002
	31:	T = 31 * r21002
	32:	T = 32 * r21002
	33:	T = 33 * r21002
	34:	T = 34 * r21002
	35:	T = 35 * r21002
	36:	T = 36 * r21002
	37:	T = 37 * r21002
	38:	T = 38 * r21002
	39:	T = 39 * r21002
	40:	T = 40 * r21002
	41:	T = 41 * r21002
	42:	T = 42 * r21002

43: T = 43 * r21002
44: T = 44 * r21002
45: T = 45 * r21002
46: T = 46 * r21002
47: T = 47 * r21002
48: T = 48 * r21002
49: T = 49 * r21002
50: T = 50 * r21002
51: T = 51 * r21002
52: T = 52 * r21002
53: T = 53 * r21002
54: T = 54 * r21002
55: T = 55 * r21002
56: T = 56 * r21002
57: T = 57 * r21002
58: T = 58 * r21002
59: T = 59 * r21002
60: T = 60 * r21002
61: T = 61 * r21002
62: T = 62 * r21002
63: T = 63 * r21002
64: T = 64 * r21002
65: T = 65 * r21002
66: T = 66 * r21002
67: T = 67 * r21002
68: T = 68 * r21002
69: T = 69 * r21002
70: T = 70 * r21002
71: T = 71 * r21002
72: T = 72 * r21002
73: T = 73 * r21002
74: T = 74 * r21002
75: T = 75 * r21002
76: T = 76 * r21002
77: T = 77 * r21002
78: T = 78 * r21002
79: T = 79 * r21002
80: T = 80 * r21002
81: T = 81 * r21002
82: T = 82 * r21002
83: T = 83 * r21002
84: T = 84 * r21002
85: T = 85 * r21002
86: T = 86 * r21002
87: T = 87 * r21002
88: T = 88 * r21002
89: T = 89 * r21002
90: T = 90 * r21002
91: T = 91 * r21002
92: T = 92 * r21002
93: T = 93 * r21002
94: T = 94 * r21002
95: T = 95 * r21002
96: T = 96 * r21002
97: T = 97 * r21002
98: T = 98 * r21002
99: T = 99 * r21002
100: T = 100 * r21002
101: T = 101 * r21002
102: T = 102 * r21002

103: T = 103 * r21002
 104: T = 104 * r21002
 105: T = 105 * r21002
 106: T = 106 * r21002
 107: T = 107 * r21002
 108: T = 108 * r21002
 109: T = 109 * r21002
 110: T = 110 * r21002
 111: T = 111 * r21002
 112: T = 112 * r21002
 113: T = 113 * r21002
 114: T = 114 * r21002
 115: T = 115 * r21002
 116: T = 116 * r21002
 117: T = 117 * r21002
 118: T = 118 * r21002
 119: T = 119 * r21002
 120: T = 120 * r21002
 121: T = 121 * r21002
 122: T = 122 * r21002
 123: T = 123 * r21002
 124: T = 124 * r21002
 125: T = 125 * r21002
 126: T = 126 * r21002
 127: T = 127 * r21002
 128: T = 128 * r21002
 129: T = 129 * r21002
 130: T = 130 * r21002
 131: T = 131 * r21002
 132: T = 132 * r21002
 133: T = 133 * r21002
 134: T = 134 * r21002
 135: T = 135 * r21002
 136: T = 136 * r21002
 137: T = 137 * r21002
 138: T = 138 * r21002
 139: T = 139 * r21002
 140: T = 140 * r21002
 141: T = 141 * r21002
 142: T = 142 * r21002
 143: T = 143 * r21002
 144: T = 144 * r21002
 145: T = 145 * r21002
 146: T = 146 * r21002
 147: T = 147 * r21002
 148: T = 148 * r21002
 149: T = 149 * r21002
 150: T = 150 * r21002
 151: T = 151 * r21002
 152: T = 152 * r21002
 153: T = 153 * r21002
 154: T = 154 * r21002
 155: T = 155 * r21002
 156: T = 156 * r21002
 157: T = 157 * r21002
 158: T = 158 * r21002
 159: T = 159 * r21002
 160: T = 160 * r21002
 161: T = 161 * r21002
 162: T = 162 * r21002

163: T = 163 * r21002
164: T = 164 * r21002
165: T = 165 * r21002
166: T = 166 * r21002
167: T = 167 * r21002
168: T = 168 * r21002
169: T = 169 * r21002
170: T = 170 * r21002
171: T = 171 * r21002
172: T = 172 * r21002
173: T = 173 * r21002
174: T = 174 * r21002
175: T = 175 * r21002
176: T = 176 * r21002
177: T = 177 * r21002
178: T = 178 * r21002
179: T = 179 * r21002
180: T = 180 * r21002
181: T = 181 * r21002
182: T = 182 * r21002
183: T = 183 * r21002
184: T = 184 * r21002
185: T = 185 * r21002
186: T = 186 * r21002
187: T = 187 * r21002
188: T = 188 * r21002
189: T = 189 * r21002
190: T = 190 * r21002
191: T = 191 * r21002
192: T = 192 * r21002
193: T = 193 * r21002
194: T = 194 * r21002
195: T = 195 * r21002
196: T = 196 * r21002
197: T = 197 * r21002
198: T = 198 * r21002
199: T = 199 * r21002
200: T = 200 * r21002
201: T = 201 * r21002
202: T = 202 * r21002
203: T = 203 * r21002
204: T = 204 * r21002
205: T = 205 * r21002
206: T = 206 * r21002
207: T = 207 * r21002
208: T = 208 * r21002
209: T = 209 * r21002
210: T = 210 * r21002
211: T = 211 * r21002
212: T = 212 * r21002
213: T = 213 * r21002
214: T = 214 * r21002
215: T = 215 * r21002
216: T = 216 * r21002
217: T = 217 * r21002
218: T = 218 * r21002
219: T = 219 * r21002
220: T = 220 * r21002
221: T = 221 * r21002
222: T = 222 * r21002

223: T = 223 * r21002
 224: T = 224 * r21002
 225: T = 225 * r21002
 226: T = 226 * r21002
 227: T = 227 * r21002
 228: T = 228 * r21002
 229: T = 229 * r21002
 230: T = 230 * r21002
 231: T = 231 * r21002
 232: T = 232 * r21002
 233: T = 233 * r21002
 234: T = 234 * r21002
 235: T = 235 * r21002
 236: T = 236 * r21002
 237: T = 237 * r21002
 238: T = 238 * r21002
 239: T = 239 * r21002
 240: T = 240 * r21002
 241: T = 241 * r21002
 242: T = 242 * r21002
 243: T = 243 * r21002
 244: T = 244 * r21002
 245: T = 245 * r21002
 246: T = 246 * r21002
 247: T = 247 * r21002
 248: T = 248 * r21002
 249: T = 249 * r21002
 250: T = 250 * r21002
 251: T = 251 * r21002
 252: T = 252 * r21002
 253: T = 253 * r21002
 254: T = 254 * r21002
 255: T = 255 * r21002
 256: T = 256 * r21002
 1001: T = 1 * r21003
 1002: T = 2 * r21003
 1003: T = 3 * r21003
 1004: T = 4 * r21003
 1005: T = 5 * r21003
 1006: T = 6 * r21003
 1008: T = 8 * r21003
 1010: T = 10 * r21003
 1012: T = 12 * r21003
 1016: T = 16 * r21003
 1020: T = 20 * r21003
 1024: T = 24 * r21003
 1032: T = 32 * r21003
 1040: T = 40 * r21003
 1048: T = 48 * r21003
 1064: T = 64 * r21003
 1080: T = 80 * r21003
 1096: T = 96 * r21003
 3001: BEFORE speed ctrl
 3003: BEFORE speed setpoint channel
 4000: Receive AFTER IF1 PROFIdrive PZD
 4001: Send BEFORE IF1 PROFIdrive PZD
 4002: Receive AFTER IF2 PZD
 4003: Send BEFORE IF2 PZD
 4004: Receive AFTER IF1 PROFIdrive flexible PZD
 4005: Receive AFTER IF2 flexible PZD

B.2 参数

Index: [0]: Run-time group 1
 [1]: Run-time group 2
 [2]: Run-time group 3
 [3]: Run-time group 4
 [4]: Run-time group 5
 [5]: Run-time group 6
 [6]: Run-time group 7
 [7]: Run-time group 8
 [8]: Run-time group 9
 [9]: Run-time group 10

Caution: The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.

Note: Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a setpoint channel has not been configured ($p0108.8 = 0$), the calculation is made before function chart 3095.

p21000[0...9]	Run-time group properties / RTG property		
INFEED, TM120	Changeable: T	Calculated: -	Access level: 1
	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4004	[0] 0

Description: Allocates properties to run-time groups 1 to 10.
 This property comprises the sampling time and, for $p21000[x] \geq 2000$, the instant of the call within the sampling time.
 The index $x + 1$ of $p21000$ corresponds to the number of the run-time group:

- $p21000[0]$ is used to set the property of the run-time group 1
- ...
- $p21000[9]$ is used to set the property of the run-time group 10

Values:

0:	Do not calculate run-time group
1:	$T = 1 * r21002$
2:	$T = 2 * r21002$
3:	$T = 3 * r21002$
4:	$T = 4 * r21002$
5:	$T = 5 * r21002$
6:	$T = 6 * r21002$
7:	$T = 7 * r21002$
8:	$T = 8 * r21002$
9:	$T = 9 * r21002$
10:	$T = 10 * r21002$
11:	$T = 11 * r21002$
12:	$T = 12 * r21002$

13: T = 13 * r21002
14: T = 14 * r21002
15: T = 15 * r21002
16: T = 16 * r21002
17: T = 17 * r21002
18: T = 18 * r21002
19: T = 19 * r21002
20: T = 20 * r21002
21: T = 21 * r21002
22: T = 22 * r21002
23: T = 23 * r21002
24: T = 24 * r21002
25: T = 25 * r21002
26: T = 26 * r21002
27: T = 27 * r21002
28: T = 28 * r21002
29: T = 29 * r21002
30: T = 30 * r21002
31: T = 31 * r21002
32: T = 32 * r21002
33: T = 33 * r21002
34: T = 34 * r21002
35: T = 35 * r21002
36: T = 36 * r21002
37: T = 37 * r21002
38: T = 38 * r21002
39: T = 39 * r21002
40: T = 40 * r21002
41: T = 41 * r21002
42: T = 42 * r21002
43: T = 43 * r21002
44: T = 44 * r21002
45: T = 45 * r21002
46: T = 46 * r21002
47: T = 47 * r21002
48: T = 48 * r21002
49: T = 49 * r21002
50: T = 50 * r21002
51: T = 51 * r21002
52: T = 52 * r21002
53: T = 53 * r21002
54: T = 54 * r21002
55: T = 55 * r21002
56: T = 56 * r21002
57: T = 57 * r21002
58: T = 58 * r21002
59: T = 59 * r21002
60: T = 60 * r21002
61: T = 61 * r21002
62: T = 62 * r21002
63: T = 63 * r21002
64: T = 64 * r21002
65: T = 65 * r21002
66: T = 66 * r21002
67: T = 67 * r21002
68: T = 68 * r21002
69: T = 69 * r21002
70: T = 70 * r21002
71: T = 71 * r21002
72: T = 72 * r21002

73: T = 73 * r21002
74: T = 74 * r21002
75: T = 75 * r21002
76: T = 76 * r21002
77: T = 77 * r21002
78: T = 78 * r21002
79: T = 79 * r21002
80: T = 80 * r21002
81: T = 81 * r21002
82: T = 82 * r21002
83: T = 83 * r21002
84: T = 84 * r21002
85: T = 85 * r21002
86: T = 86 * r21002
87: T = 87 * r21002
88: T = 88 * r21002
89: T = 89 * r21002
90: T = 90 * r21002
91: T = 91 * r21002
92: T = 92 * r21002
93: T = 93 * r21002
94: T = 94 * r21002
95: T = 95 * r21002
96: T = 96 * r21002
97: T = 97 * r21002
98: T = 98 * r21002
99: T = 99 * r21002
100: T = 100 * r21002
101: T = 101 * r21002
102: T = 102 * r21002
103: T = 103 * r21002
104: T = 104 * r21002
105: T = 105 * r21002
106: T = 106 * r21002
107: T = 107 * r21002
108: T = 108 * r21002
109: T = 109 * r21002
110: T = 110 * r21002
111: T = 111 * r21002
112: T = 112 * r21002
113: T = 113 * r21002
114: T = 114 * r21002
115: T = 115 * r21002
116: T = 116 * r21002
117: T = 117 * r21002
118: T = 118 * r21002
119: T = 119 * r21002
120: T = 120 * r21002
121: T = 121 * r21002
122: T = 122 * r21002
123: T = 123 * r21002
124: T = 124 * r21002
125: T = 125 * r21002
126: T = 126 * r21002
127: T = 127 * r21002
128: T = 128 * r21002
129: T = 129 * r21002
130: T = 130 * r21002
131: T = 131 * r21002
132: T = 132 * r21002

133: T = 133 * r21002
 134: T = 134 * r21002
 135: T = 135 * r21002
 136: T = 136 * r21002
 137: T = 137 * r21002
 138: T = 138 * r21002
 139: T = 139 * r21002
 140: T = 140 * r21002
 141: T = 141 * r21002
 142: T = 142 * r21002
 143: T = 143 * r21002
 144: T = 144 * r21002
 145: T = 145 * r21002
 146: T = 146 * r21002
 147: T = 147 * r21002
 148: T = 148 * r21002
 149: T = 149 * r21002
 150: T = 150 * r21002
 151: T = 151 * r21002
 152: T = 152 * r21002
 153: T = 153 * r21002
 154: T = 154 * r21002
 155: T = 155 * r21002
 156: T = 156 * r21002
 157: T = 157 * r21002
 158: T = 158 * r21002
 159: T = 159 * r21002
 160: T = 160 * r21002
 161: T = 161 * r21002
 162: T = 162 * r21002
 163: T = 163 * r21002
 164: T = 164 * r21002
 165: T = 165 * r21002
 166: T = 166 * r21002
 167: T = 167 * r21002
 168: T = 168 * r21002
 169: T = 169 * r21002
 170: T = 170 * r21002
 171: T = 171 * r21002
 172: T = 172 * r21002
 173: T = 173 * r21002
 174: T = 174 * r21002
 175: T = 175 * r21002
 176: T = 176 * r21002
 177: T = 177 * r21002
 178: T = 178 * r21002
 179: T = 179 * r21002
 180: T = 180 * r21002
 181: T = 181 * r21002
 182: T = 182 * r21002
 183: T = 183 * r21002
 184: T = 184 * r21002
 185: T = 185 * r21002
 186: T = 186 * r21002
 187: T = 187 * r21002
 188: T = 188 * r21002
 189: T = 189 * r21002
 190: T = 190 * r21002
 191: T = 191 * r21002
 192: T = 192 * r21002

193: T = 193 * r21002
194: T = 194 * r21002
195: T = 195 * r21002
196: T = 196 * r21002
197: T = 197 * r21002
198: T = 198 * r21002
199: T = 199 * r21002
200: T = 200 * r21002
201: T = 201 * r21002
202: T = 202 * r21002
203: T = 203 * r21002
204: T = 204 * r21002
205: T = 205 * r21002
206: T = 206 * r21002
207: T = 207 * r21002
208: T = 208 * r21002
209: T = 209 * r21002
210: T = 210 * r21002
211: T = 211 * r21002
212: T = 212 * r21002
213: T = 213 * r21002
214: T = 214 * r21002
215: T = 215 * r21002
216: T = 216 * r21002
217: T = 217 * r21002
218: T = 218 * r21002
219: T = 219 * r21002
220: T = 220 * r21002
221: T = 221 * r21002
222: T = 222 * r21002
223: T = 223 * r21002
224: T = 224 * r21002
225: T = 225 * r21002
226: T = 226 * r21002
227: T = 227 * r21002
228: T = 228 * r21002
229: T = 229 * r21002
230: T = 230 * r21002
231: T = 231 * r21002
232: T = 232 * r21002
233: T = 233 * r21002
234: T = 234 * r21002
235: T = 235 * r21002
236: T = 236 * r21002
237: T = 237 * r21002
238: T = 238 * r21002
239: T = 239 * r21002
240: T = 240 * r21002
241: T = 241 * r21002
242: T = 242 * r21002
243: T = 243 * r21002
244: T = 244 * r21002
245: T = 245 * r21002
246: T = 246 * r21002
247: T = 247 * r21002
248: T = 248 * r21002
249: T = 249 * r21002
250: T = 250 * r21002
251: T = 251 * r21002
252: T = 252 * r21002

253: T = 253 * r21002
 254: T = 254 * r21002
 255: T = 255 * r21002
 256: T = 256 * r21002
 1001: T = 1 * r21003
 1002: T = 2 * r21003
 1003: T = 3 * r21003
 1004: T = 4 * r21003
 1005: T = 5 * r21003
 1006: T = 6 * r21003
 1008: T = 8 * r21003
 1010: T = 10 * r21003
 1012: T = 12 * r21003
 1016: T = 16 * r21003
 1020: T = 20 * r21003
 1024: T = 24 * r21003
 1032: T = 32 * r21003
 1040: T = 40 * r21003
 1048: T = 48 * r21003
 1064: T = 64 * r21003
 1080: T = 80 * r21003
 1096: T = 96 * r21003
 4000: Receive AFTER IF1 PROFIdrive PZD
 4001: Send BEFORE IF1 PROFIdrive PZD
 4004: Receive AFTER IF1 PROFIdrive flexible PZD

Index: [0]: Run-time group 1
 [1]: Run-time group 2
 [2]: Run-time group 3
 [3]: Run-time group 4
 [4]: Run-time group 5
 [5]: Run-time group 6
 [6]: Run-time group 7
 [7]: Run-time group 8
 [8]: Run-time group 9
 [9]: Run-time group 10

Caution: The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.

Note: Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a setpoint channel has not been configured ($p0108.8 = 0$), the calculation is made before function chart 3095.

p21000[0..9]	Run-time group properties / RTG property		
SERVO, VECTOR	Changeable: T	Calculated: -	Access level: 1
	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4005	[0] 0

Description: Allocates properties to run-time groups 1 to 10.
 This property comprises the sampling time and, for p21000[x] >= 2000, the instant of the call within the sampling time.
 The index x + 1 of p21000 corresponds to the number of the run-time group:

- p21000[0] is used to set the property of the run-time group 1
- ...
- p21000[9] is used to set the property of the run-time group 10

Values:	0:	Do not calculate run-time group
	1:	T = 1 * r21002
	2:	T = 2 * r21002
	3:	T = 3 * r21002
	4:	T = 4 * r21002
	5:	T = 5 * r21002
	6:	T = 6 * r21002
	7:	T = 7 * r21002
	8:	T = 8 * r21002
	9:	T = 9 * r21002
	10:	T = 10 * r21002
	11:	T = 11 * r21002
	12:	T = 12 * r21002
	13:	T = 13 * r21002
	14:	T = 14 * r21002
	15:	T = 15 * r21002
	16:	T = 16 * r21002
	17:	T = 17 * r21002
	18:	T = 18 * r21002
	19:	T = 19 * r21002
	20:	T = 20 * r21002
	21:	T = 21 * r21002
	22:	T = 22 * r21002
	23:	T = 23 * r21002
	24:	T = 24 * r21002
	25:	T = 25 * r21002
	26:	T = 26 * r21002
	27:	T = 27 * r21002
	28:	T = 28 * r21002
	29:	T = 29 * r21002
	30:	T = 30 * r21002
	31:	T = 31 * r21002
	32:	T = 32 * r21002
	33:	T = 33 * r21002
	34:	T = 34 * r21002
	35:	T = 35 * r21002
	36:	T = 36 * r21002
	37:	T = 37 * r21002
	38:	T = 38 * r21002
	39:	T = 39 * r21002
	40:	T = 40 * r21002
	41:	T = 41 * r21002
	42:	T = 42 * r21002

43: T = 43 * r21002
 44: T = 44 * r21002
 45: T = 45 * r21002
 46: T = 46 * r21002
 47: T = 47 * r21002
 48: T = 48 * r21002
 49: T = 49 * r21002
 50: T = 50 * r21002
 51: T = 51 * r21002
 52: T = 52 * r21002
 53: T = 53 * r21002
 54: T = 54 * r21002
 55: T = 55 * r21002
 56: T = 56 * r21002
 57: T = 57 * r21002
 58: T = 58 * r21002
 59: T = 59 * r21002
 60: T = 60 * r21002
 61: T = 61 * r21002
 62: T = 62 * r21002
 63: T = 63 * r21002
 64: T = 64 * r21002
 65: T = 65 * r21002
 66: T = 66 * r21002
 67: T = 67 * r21002
 68: T = 68 * r21002
 69: T = 69 * r21002
 70: T = 70 * r21002
 71: T = 71 * r21002
 72: T = 72 * r21002
 73: T = 73 * r21002
 74: T = 74 * r21002
 75: T = 75 * r21002
 76: T = 76 * r21002
 77: T = 77 * r21002
 78: T = 78 * r21002
 79: T = 79 * r21002
 80: T = 80 * r21002
 81: T = 81 * r21002
 82: T = 82 * r21002
 83: T = 83 * r21002
 84: T = 84 * r21002
 85: T = 85 * r21002
 86: T = 86 * r21002
 87: T = 87 * r21002
 88: T = 88 * r21002
 89: T = 89 * r21002
 90: T = 90 * r21002
 91: T = 91 * r21002
 92: T = 92 * r21002
 93: T = 93 * r21002
 94: T = 94 * r21002
 95: T = 95 * r21002
 96: T = 96 * r21002
 97: T = 97 * r21002
 98: T = 98 * r21002
 99: T = 99 * r21002
 100: T = 100 * r21002
 101: T = 101 * r21002
 102: T = 102 * r21002

103: T = 103 * r21002
104: T = 104 * r21002
105: T = 105 * r21002
106: T = 106 * r21002
107: T = 107 * r21002
108: T = 108 * r21002
109: T = 109 * r21002
110: T = 110 * r21002
111: T = 111 * r21002
112: T = 112 * r21002
113: T = 113 * r21002
114: T = 114 * r21002
115: T = 115 * r21002
116: T = 116 * r21002
117: T = 117 * r21002
118: T = 118 * r21002
119: T = 119 * r21002
120: T = 120 * r21002
121: T = 121 * r21002
122: T = 122 * r21002
123: T = 123 * r21002
124: T = 124 * r21002
125: T = 125 * r21002
126: T = 126 * r21002
127: T = 127 * r21002
128: T = 128 * r21002
129: T = 129 * r21002
130: T = 130 * r21002
131: T = 131 * r21002
132: T = 132 * r21002
133: T = 133 * r21002
134: T = 134 * r21002
135: T = 135 * r21002
136: T = 136 * r21002
137: T = 137 * r21002
138: T = 138 * r21002
139: T = 139 * r21002
140: T = 140 * r21002
141: T = 141 * r21002
142: T = 142 * r21002
143: T = 143 * r21002
144: T = 144 * r21002
145: T = 145 * r21002
146: T = 146 * r21002
147: T = 147 * r21002
148: T = 148 * r21002
149: T = 149 * r21002
150: T = 150 * r21002
151: T = 151 * r21002
152: T = 152 * r21002
153: T = 153 * r21002
154: T = 154 * r21002
155: T = 155 * r21002
156: T = 156 * r21002
157: T = 157 * r21002
158: T = 158 * r21002
159: T = 159 * r21002
160: T = 160 * r21002
161: T = 161 * r21002
162: T = 162 * r21002

163: T = 163 * r21002
164: T = 164 * r21002
165: T = 165 * r21002
166: T = 166 * r21002
167: T = 167 * r21002
168: T = 168 * r21002
169: T = 169 * r21002
170: T = 170 * r21002
171: T = 171 * r21002
172: T = 172 * r21002
173: T = 173 * r21002
174: T = 174 * r21002
175: T = 175 * r21002
176: T = 176 * r21002
177: T = 177 * r21002
178: T = 178 * r21002
179: T = 179 * r21002
180: T = 180 * r21002
181: T = 181 * r21002
182: T = 182 * r21002
183: T = 183 * r21002
184: T = 184 * r21002
185: T = 185 * r21002
186: T = 186 * r21002
187: T = 187 * r21002
188: T = 188 * r21002
189: T = 189 * r21002
190: T = 190 * r21002
191: T = 191 * r21002
192: T = 192 * r21002
193: T = 193 * r21002
194: T = 194 * r21002
195: T = 195 * r21002
196: T = 196 * r21002
197: T = 197 * r21002
198: T = 198 * r21002
199: T = 199 * r21002
200: T = 200 * r21002
201: T = 201 * r21002
202: T = 202 * r21002
203: T = 203 * r21002
204: T = 204 * r21002
205: T = 205 * r21002
206: T = 206 * r21002
207: T = 207 * r21002
208: T = 208 * r21002
209: T = 209 * r21002
210: T = 210 * r21002
211: T = 211 * r21002
212: T = 212 * r21002
213: T = 213 * r21002
214: T = 214 * r21002
215: T = 215 * r21002
216: T = 216 * r21002
217: T = 217 * r21002
218: T = 218 * r21002
219: T = 219 * r21002
220: T = 220 * r21002
221: T = 221 * r21002
222: T = 222 * r21002

223: T = 223 * r21002
224: T = 224 * r21002
225: T = 225 * r21002
226: T = 226 * r21002
227: T = 227 * r21002
228: T = 228 * r21002
229: T = 229 * r21002
230: T = 230 * r21002
231: T = 231 * r21002
232: T = 232 * r21002
233: T = 233 * r21002
234: T = 234 * r21002
235: T = 235 * r21002
236: T = 236 * r21002
237: T = 237 * r21002
238: T = 238 * r21002
239: T = 239 * r21002
240: T = 240 * r21002
241: T = 241 * r21002
242: T = 242 * r21002
243: T = 243 * r21002
244: T = 244 * r21002
245: T = 245 * r21002
246: T = 246 * r21002
247: T = 247 * r21002
248: T = 248 * r21002
249: T = 249 * r21002
250: T = 250 * r21002
251: T = 251 * r21002
252: T = 252 * r21002
253: T = 253 * r21002
254: T = 254 * r21002
255: T = 255 * r21002
256: T = 256 * r21002
1001: T = 1 * r21003
1002: T = 2 * r21003
1003: T = 3 * r21003
1004: T = 4 * r21003
1005: T = 5 * r21003
1006: T = 6 * r21003
1008: T = 8 * r21003
1010: T = 10 * r21003
1012: T = 12 * r21003
1016: T = 16 * r21003
1020: T = 20 * r21003
1024: T = 24 * r21003
1032: T = 32 * r21003
1040: T = 40 * r21003
1048: T = 48 * r21003
1064: T = 64 * r21003
1080: T = 80 * r21003
1096: T = 96 * r21003
3001: BEFORE speed ctrl
3003: BEFORE speed setpoint channel
3004: BEFORE pos ctrl
3005: BEFORE basic positioner
3006: BEFORE standard technology controller
3007: BEFORE act p v
4000: Receive AFTER IF1 PROFIdrive PZD
4001: Send BEFORE IF1 PROFIdrive PZD

4002: Receive AFTER IF2 PZD
 4003: Send BEFORE IF2 PZD
 4004: Receive AFTER IF1 PROFIdrive flexible PZD
 4005: Receive AFTER IF2 flexible PZD

Index: [0]: Run-time group 1
 [1]: Run-time group 2
 [2]: Run-time group 3
 [3]: Run-time group 4
 [4]: Run-time group 5
 [5]: Run-time group 6
 [6]: Run-time group 7
 [7]: Run-time group 8
 [8]: Run-time group 9
 [9]: Run-time group 10

Caution: The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.

Note: Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a setpoint channel has not been configured ($p0108.8 = 0$), the calculation is made before function chart 3095.

p21000[0...9]	Run-time group properties / RTG property		
TB30, TM31	Changeable: T	Calculated: -	Access level: 1
	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4004	[0] 0

Description: Allocates properties to run-time groups 1 to 10. This property comprises the sampling time and, for $p21000[x] \geq 2000$, the instant of the call within the sampling time. The index $x + 1$ of $p21000$ corresponds to the number of the run-time group:

- $p21000[0]$ is used to set the property of the run-time group 1
 ...
 - $p21000[9]$ is used to set the property of the run-time group 10

Values: 0: Do not calculate run-time group
 1: $T = 1 * r21002$
 2: $T = 2 * r21002$
 3: $T = 3 * r21002$
 4: $T = 4 * r21002$
 5: $T = 5 * r21002$
 6: $T = 6 * r21002$
 7: $T = 7 * r21002$
 8: $T = 8 * r21002$

9: T = 9 * r21002
10: T = 10 * r21002
11: T = 11 * r21002
12: T = 12 * r21002
13: T = 13 * r21002
14: T = 14 * r21002
15: T = 15 * r21002
16: T = 16 * r21002
17: T = 17 * r21002
18: T = 18 * r21002
19: T = 19 * r21002
20: T = 20 * r21002
21: T = 21 * r21002
22: T = 22 * r21002
23: T = 23 * r21002
24: T = 24 * r21002
25: T = 25 * r21002
26: T = 26 * r21002
27: T = 27 * r21002
28: T = 28 * r21002
29: T = 29 * r21002
30: T = 30 * r21002
31: T = 31 * r21002
32: T = 32 * r21002
33: T = 33 * r21002
34: T = 34 * r21002
35: T = 35 * r21002
36: T = 36 * r21002
37: T = 37 * r21002
38: T = 38 * r21002
39: T = 39 * r21002
40: T = 40 * r21002
41: T = 41 * r21002
42: T = 42 * r21002
43: T = 43 * r21002
44: T = 44 * r21002
45: T = 45 * r21002
46: T = 46 * r21002
47: T = 47 * r21002
48: T = 48 * r21002
49: T = 49 * r21002
50: T = 50 * r21002
51: T = 51 * r21002
52: T = 52 * r21002
53: T = 53 * r21002
54: T = 54 * r21002
55: T = 55 * r21002
56: T = 56 * r21002
57: T = 57 * r21002
58: T = 58 * r21002
59: T = 59 * r21002
60: T = 60 * r21002
61: T = 61 * r21002
62: T = 62 * r21002
63: T = 63 * r21002
64: T = 64 * r21002
65: T = 65 * r21002
66: T = 66 * r21002
67: T = 67 * r21002
68: T = 68 * r21002

69: T = 69 * r21002
70: T = 70 * r21002
71: T = 71 * r21002
72: T = 72 * r21002
73: T = 73 * r21002
74: T = 74 * r21002
75: T = 75 * r21002
76: T = 76 * r21002
77: T = 77 * r21002
78: T = 78 * r21002
79: T = 79 * r21002
80: T = 80 * r21002
81: T = 81 * r21002
82: T = 82 * r21002
83: T = 83 * r21002
84: T = 84 * r21002
85: T = 85 * r21002
86: T = 86 * r21002
87: T = 87 * r21002
88: T = 88 * r21002
89: T = 89 * r21002
90: T = 90 * r21002
91: T = 91 * r21002
92: T = 92 * r21002
93: T = 93 * r21002
94: T = 94 * r21002
95: T = 95 * r21002
96: T = 96 * r21002
97: T = 97 * r21002
98: T = 98 * r21002
99: T = 99 * r21002
100: T = 100 * r21002
101: T = 101 * r21002
102: T = 102 * r21002
103: T = 103 * r21002
104: T = 104 * r21002
105: T = 105 * r21002
106: T = 106 * r21002
107: T = 107 * r21002
108: T = 108 * r21002
109: T = 109 * r21002
110: T = 110 * r21002
111: T = 111 * r21002
112: T = 112 * r21002
113: T = 113 * r21002
114: T = 114 * r21002
115: T = 115 * r21002
116: T = 116 * r21002
117: T = 117 * r21002
118: T = 118 * r21002
119: T = 119 * r21002
120: T = 120 * r21002
121: T = 121 * r21002
122: T = 122 * r21002
123: T = 123 * r21002
124: T = 124 * r21002
125: T = 125 * r21002
126: T = 126 * r21002
127: T = 127 * r21002
128: T = 128 * r21002

129: T = 129 * r21002
130: T = 130 * r21002
131: T = 131 * r21002
132: T = 132 * r21002
133: T = 133 * r21002
134: T = 134 * r21002
135: T = 135 * r21002
136: T = 136 * r21002
137: T = 137 * r21002
138: T = 138 * r21002
139: T = 139 * r21002
140: T = 140 * r21002
141: T = 141 * r21002
142: T = 142 * r21002
143: T = 143 * r21002
144: T = 144 * r21002
145: T = 145 * r21002
146: T = 146 * r21002
147: T = 147 * r21002
148: T = 148 * r21002
149: T = 149 * r21002
150: T = 150 * r21002
151: T = 151 * r21002
152: T = 152 * r21002
153: T = 153 * r21002
154: T = 154 * r21002
155: T = 155 * r21002
156: T = 156 * r21002
157: T = 157 * r21002
158: T = 158 * r21002
159: T = 159 * r21002
160: T = 160 * r21002
161: T = 161 * r21002
162: T = 162 * r21002
163: T = 163 * r21002
164: T = 164 * r21002
165: T = 165 * r21002
166: T = 166 * r21002
167: T = 167 * r21002
168: T = 168 * r21002
169: T = 169 * r21002
170: T = 170 * r21002
171: T = 171 * r21002
172: T = 172 * r21002
173: T = 173 * r21002
174: T = 174 * r21002
175: T = 175 * r21002
176: T = 176 * r21002
177: T = 177 * r21002
178: T = 178 * r21002
179: T = 179 * r21002
180: T = 180 * r21002
181: T = 181 * r21002
182: T = 182 * r21002
183: T = 183 * r21002
184: T = 184 * r21002
185: T = 185 * r21002
186: T = 186 * r21002
187: T = 187 * r21002
188: T = 188 * r21002

189: T = 189 * r21002
190: T = 190 * r21002
191: T = 191 * r21002
192: T = 192 * r21002
193: T = 193 * r21002
194: T = 194 * r21002
195: T = 195 * r21002
196: T = 196 * r21002
197: T = 197 * r21002
198: T = 198 * r21002
199: T = 199 * r21002
200: T = 200 * r21002
201: T = 201 * r21002
202: T = 202 * r21002
203: T = 203 * r21002
204: T = 204 * r21002
205: T = 205 * r21002
206: T = 206 * r21002
207: T = 207 * r21002
208: T = 208 * r21002
209: T = 209 * r21002
210: T = 210 * r21002
211: T = 211 * r21002
212: T = 212 * r21002
213: T = 213 * r21002
214: T = 214 * r21002
215: T = 215 * r21002
216: T = 216 * r21002
217: T = 217 * r21002
218: T = 218 * r21002
219: T = 219 * r21002
220: T = 220 * r21002
221: T = 221 * r21002
222: T = 222 * r21002
223: T = 223 * r21002
224: T = 224 * r21002
225: T = 225 * r21002
226: T = 226 * r21002
227: T = 227 * r21002
228: T = 228 * r21002
229: T = 229 * r21002
230: T = 230 * r21002
231: T = 231 * r21002
232: T = 232 * r21002
233: T = 233 * r21002
234: T = 234 * r21002
235: T = 235 * r21002
236: T = 236 * r21002
237: T = 237 * r21002
238: T = 238 * r21002
239: T = 239 * r21002
240: T = 240 * r21002
241: T = 241 * r21002
242: T = 242 * r21002
243: T = 243 * r21002
244: T = 244 * r21002
245: T = 245 * r21002
246: T = 246 * r21002
247: T = 247 * r21002
248: T = 248 * r21002

249:	T = 249 * r21002
250:	T = 250 * r21002
251:	T = 251 * r21002
252:	T = 252 * r21002
253:	T = 253 * r21002
254:	T = 254 * r21002
255:	T = 255 * r21002
256:	T = 256 * r21002
1001:	T = 1 * r21003
1002:	T = 2 * r21003
1003:	T = 3 * r21003
1004:	T = 4 * r21003
1005:	T = 5 * r21003
1006:	T = 6 * r21003
1008:	T = 8 * r21003
1010:	T = 10 * r21003
1012:	T = 12 * r21003
1016:	T = 16 * r21003
1020:	T = 20 * r21003
1024:	T = 24 * r21003
1032:	T = 32 * r21003
1040:	T = 40 * r21003
1048:	T = 48 * r21003
1064:	T = 64 * r21003
1080:	T = 80 * r21003
1096:	T = 96 * r21003
2000:	Read-in AFTER digital inputs
2001:	Output BEFORE digital outputs
2002:	Read-in AFTER analog inputs
2003:	Output BEFORE analog outputs
4000:	Receive AFTER IF1 PROFIdrive PZD
4001:	Send BEFORE IF1 PROFIdrive PZD
4004:	Receive AFTER IF1 PROFIdrive flexible PZD
Index:	[0]: Run-time group 1
	[1]: Run-time group 2
	[2]: Run-time group 3
	[3]: Run-time group 4
	[4]: Run-time group 5
	[5]: Run-time group 6
	[6]: Run-time group 7
	[7]: Run-time group 8
	[8]: Run-time group 9
	[9]: Run-time group 10
Caution:	The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.
Note:	Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a

setpoint channel has not been configured (p0108.8 = 0), the calculation is made before function chart 3095.

p21000[0...9]	Run-time group properties / RTG property		
TM41	Changeable: T	Calculated: -	Access level: 1
	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4004	[0] 0

Description: Allocates properties to run-time groups 1 to 10.
This property comprises the sampling time and, for p21000[x] >= 2000, the instant of the call within the sampling time.
The index x + 1 of p21000 corresponds to the number of the run-time group:

- p21000[0] is used to set the property of the run-time group 1
- ...
- p21000[9] is used to set the property of the run-time group 10

Values:	0:	Do not calculate run-time group
	1:	T = 1 * r21002
	2:	T = 2 * r21002
	3:	T = 3 * r21002
	4:	T = 4 * r21002
	5:	T = 5 * r21002
	6:	T = 6 * r21002
	7:	T = 7 * r21002
	8:	T = 8 * r21002
	9:	T = 9 * r21002
	10:	T = 10 * r21002
	11:	T = 11 * r21002
	12:	T = 12 * r21002
	13:	T = 13 * r21002
	14:	T = 14 * r21002
	15:	T = 15 * r21002
	16:	T = 16 * r21002
	17:	T = 17 * r21002
	18:	T = 18 * r21002
	19:	T = 19 * r21002
	20:	T = 20 * r21002
	21:	T = 21 * r21002
	22:	T = 22 * r21002
	23:	T = 23 * r21002
	24:	T = 24 * r21002
	25:	T = 25 * r21002
	26:	T = 26 * r21002
	27:	T = 27 * r21002
	28:	T = 28 * r21002
	29:	T = 29 * r21002
	30:	T = 30 * r21002
	31:	T = 31 * r21002
	32:	T = 32 * r21002
	33:	T = 33 * r21002
	34:	T = 34 * r21002
	35:	T = 35 * r21002
	36:	T = 36 * r21002
	37:	T = 37 * r21002
	38:	T = 38 * r21002
	39:	T = 39 * r21002

40: T = 40 * r21002
41: T = 41 * r21002
42: T = 42 * r21002
43: T = 43 * r21002
44: T = 44 * r21002
45: T = 45 * r21002
46: T = 46 * r21002
47: T = 47 * r21002
48: T = 48 * r21002
49: T = 49 * r21002
50: T = 50 * r21002
51: T = 51 * r21002
52: T = 52 * r21002
53: T = 53 * r21002
54: T = 54 * r21002
55: T = 55 * r21002
56: T = 56 * r21002
57: T = 57 * r21002
58: T = 58 * r21002
59: T = 59 * r21002
60: T = 60 * r21002
61: T = 61 * r21002
62: T = 62 * r21002
63: T = 63 * r21002
64: T = 64 * r21002
65: T = 65 * r21002
66: T = 66 * r21002
67: T = 67 * r21002
68: T = 68 * r21002
69: T = 69 * r21002
70: T = 70 * r21002
71: T = 71 * r21002
72: T = 72 * r21002
73: T = 73 * r21002
74: T = 74 * r21002
75: T = 75 * r21002
76: T = 76 * r21002
77: T = 77 * r21002
78: T = 78 * r21002
79: T = 79 * r21002
80: T = 80 * r21002
81: T = 81 * r21002
82: T = 82 * r21002
83: T = 83 * r21002
84: T = 84 * r21002
85: T = 85 * r21002
86: T = 86 * r21002
87: T = 87 * r21002
88: T = 88 * r21002
89: T = 89 * r21002
90: T = 90 * r21002
91: T = 91 * r21002
92: T = 92 * r21002
93: T = 93 * r21002
94: T = 94 * r21002
95: T = 95 * r21002
96: T = 96 * r21002
97: T = 97 * r21002
98: T = 98 * r21002
99: T = 99 * r21002

100: T = 100 * r21002
101: T = 101 * r21002
102: T = 102 * r21002
103: T = 103 * r21002
104: T = 104 * r21002
105: T = 105 * r21002
106: T = 106 * r21002
107: T = 107 * r21002
108: T = 108 * r21002
109: T = 109 * r21002
110: T = 110 * r21002
111: T = 111 * r21002
112: T = 112 * r21002
113: T = 113 * r21002
114: T = 114 * r21002
115: T = 115 * r21002
116: T = 116 * r21002
117: T = 117 * r21002
118: T = 118 * r21002
119: T = 119 * r21002
120: T = 120 * r21002
121: T = 121 * r21002
122: T = 122 * r21002
123: T = 123 * r21002
124: T = 124 * r21002
125: T = 125 * r21002
126: T = 126 * r21002
127: T = 127 * r21002
128: T = 128 * r21002
129: T = 129 * r21002
130: T = 130 * r21002
131: T = 131 * r21002
132: T = 132 * r21002
133: T = 133 * r21002
134: T = 134 * r21002
135: T = 135 * r21002
136: T = 136 * r21002
137: T = 137 * r21002
138: T = 138 * r21002
139: T = 139 * r21002
140: T = 140 * r21002
141: T = 141 * r21002
142: T = 142 * r21002
143: T = 143 * r21002
144: T = 144 * r21002
145: T = 145 * r21002
146: T = 146 * r21002
147: T = 147 * r21002
148: T = 148 * r21002
149: T = 149 * r21002
150: T = 150 * r21002
151: T = 151 * r21002
152: T = 152 * r21002
153: T = 153 * r21002
154: T = 154 * r21002
155: T = 155 * r21002
156: T = 156 * r21002
157: T = 157 * r21002
158: T = 158 * r21002
159: T = 159 * r21002

160: T = 160 * r21002
161: T = 161 * r21002
162: T = 162 * r21002
163: T = 163 * r21002
164: T = 164 * r21002
165: T = 165 * r21002
166: T = 166 * r21002
167: T = 167 * r21002
168: T = 168 * r21002
169: T = 169 * r21002
170: T = 170 * r21002
171: T = 171 * r21002
172: T = 172 * r21002
173: T = 173 * r21002
174: T = 174 * r21002
175: T = 175 * r21002
176: T = 176 * r21002
177: T = 177 * r21002
178: T = 178 * r21002
179: T = 179 * r21002
180: T = 180 * r21002
181: T = 181 * r21002
182: T = 182 * r21002
183: T = 183 * r21002
184: T = 184 * r21002
185: T = 185 * r21002
186: T = 186 * r21002
187: T = 187 * r21002
188: T = 188 * r21002
189: T = 189 * r21002
190: T = 190 * r21002
191: T = 191 * r21002
192: T = 192 * r21002
193: T = 193 * r21002
194: T = 194 * r21002
195: T = 195 * r21002
196: T = 196 * r21002
197: T = 197 * r21002
198: T = 198 * r21002
199: T = 199 * r21002
200: T = 200 * r21002
201: T = 201 * r21002
202: T = 202 * r21002
203: T = 203 * r21002
204: T = 204 * r21002
205: T = 205 * r21002
206: T = 206 * r21002
207: T = 207 * r21002
208: T = 208 * r21002
209: T = 209 * r21002
210: T = 210 * r21002
211: T = 211 * r21002
212: T = 212 * r21002
213: T = 213 * r21002
214: T = 214 * r21002
215: T = 215 * r21002
216: T = 216 * r21002
217: T = 217 * r21002
218: T = 218 * r21002
219: T = 219 * r21002

220:	T = 220 * r21002
221:	T = 221 * r21002
222:	T = 222 * r21002
223:	T = 223 * r21002
224:	T = 224 * r21002
225:	T = 225 * r21002
226:	T = 226 * r21002
227:	T = 227 * r21002
228:	T = 228 * r21002
229:	T = 229 * r21002
230:	T = 230 * r21002
231:	T = 231 * r21002
232:	T = 232 * r21002
233:	T = 233 * r21002
234:	T = 234 * r21002
235:	T = 235 * r21002
236:	T = 236 * r21002
237:	T = 237 * r21002
238:	T = 238 * r21002
239:	T = 239 * r21002
240:	T = 240 * r21002
241:	T = 241 * r21002
242:	T = 242 * r21002
243:	T = 243 * r21002
244:	T = 244 * r21002
245:	T = 245 * r21002
246:	T = 246 * r21002
247:	T = 247 * r21002
248:	T = 248 * r21002
249:	T = 249 * r21002
250:	T = 250 * r21002
251:	T = 251 * r21002
252:	T = 252 * r21002
253:	T = 253 * r21002
254:	T = 254 * r21002
255:	T = 255 * r21002
256:	T = 256 * r21002
1001:	T = 1 * r21003
1002:	T = 2 * r21003
1003:	T = 3 * r21003
1004:	T = 4 * r21003
1005:	T = 5 * r21003
1006:	T = 6 * r21003
1008:	T = 8 * r21003
1010:	T = 10 * r21003
1012:	T = 12 * r21003
1016:	T = 16 * r21003
1020:	T = 20 * r21003
1024:	T = 24 * r21003
1032:	T = 32 * r21003
1040:	T = 40 * r21003
1048:	T = 48 * r21003
1064:	T = 64 * r21003
1080:	T = 80 * r21003
1096:	T = 96 * r21003
2000:	Read-in AFTER digital inputs
2001:	Output BEFORE digital outputs
2002:	Read-in AFTER analog inputs
4000:	Receive AFTER IF1 PROFIdrive PZD
4001:	Send BEFORE IF1 PROFIdrive PZD

	4004:	Receive AFTER IF1 PROFIdrive flexible PZD
Index:	[0]:	Run-time group 1
	[1]:	Run-time group 2
	[2]:	Run-time group 3
	[3]:	Run-time group 4
	[4]:	Run-time group 5
	[5]:	Run-time group 6
	[6]:	Run-time group 7
	[7]:	Run-time group 8
	[8]:	Run-time group 9
	[9]:	Run-time group 10
Caution:	The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.	
Note:	Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a setpoint channel has not been configured ($p0108.8 = 0$), the calculation is made before function chart 3095.	

p21000[0...9]	Run-time group properties / RTG property		
VECTORMV,	Changeable: T	Calculated: -	Access level: 1
VECTORGL	Data type: Integer16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	4005	[0] 0

Description: Allocates properties to run-time groups 1 to 10.
This property comprises the sampling time and, for $p21000[x] \geq 2000$, the instant of the call within the sampling time.
The index $x + 1$ of $p21000$ corresponds to the number of the run-time group:

- $p21000[0]$ is used to set the property of the run-time group 1
- ...
- $p21000[9]$ is used to set the property of the run-time group 10

Values:	0:	Do not calculate run-time group
	1:	$T = 1 * r21002$
	2:	$T = 2 * r21002$
	3:	$T = 3 * r21002$
	4:	$T = 4 * r21002$
	5:	$T = 5 * r21002$
	6:	$T = 6 * r21002$
	7:	$T = 7 * r21002$
	8:	$T = 8 * r21002$
	9:	$T = 9 * r21002$
	10:	$T = 10 * r21002$
	11:	$T = 11 * r21002$

12: T = 12 * r21002
 13: T = 13 * r21002
 14: T = 14 * r21002
 15: T = 15 * r21002
 16: T = 16 * r21002
 17: T = 17 * r21002
 18: T = 18 * r21002
 19: T = 19 * r21002
 20: T = 20 * r21002
 21: T = 21 * r21002
 22: T = 22 * r21002
 23: T = 23 * r21002
 24: T = 24 * r21002
 25: T = 25 * r21002
 26: T = 26 * r21002
 27: T = 27 * r21002
 28: T = 28 * r21002
 29: T = 29 * r21002
 30: T = 30 * r21002
 31: T = 31 * r21002
 32: T = 32 * r21002
 33: T = 33 * r21002
 34: T = 34 * r21002
 35: T = 35 * r21002
 36: T = 36 * r21002
 37: T = 37 * r21002
 38: T = 38 * r21002
 39: T = 39 * r21002
 40: T = 40 * r21002
 41: T = 41 * r21002
 42: T = 42 * r21002
 43: T = 43 * r21002
 44: T = 44 * r21002
 45: T = 45 * r21002
 46: T = 46 * r21002
 47: T = 47 * r21002
 48: T = 48 * r21002
 49: T = 49 * r21002
 50: T = 50 * r21002
 51: T = 51 * r21002
 52: T = 52 * r21002
 53: T = 53 * r21002
 54: T = 54 * r21002
 55: T = 55 * r21002
 56: T = 56 * r21002
 57: T = 57 * r21002
 58: T = 58 * r21002
 59: T = 59 * r21002
 60: T = 60 * r21002
 61: T = 61 * r21002
 62: T = 62 * r21002
 63: T = 63 * r21002
 64: T = 64 * r21002
 65: T = 65 * r21002
 66: T = 66 * r21002
 67: T = 67 * r21002
 68: T = 68 * r21002
 69: T = 69 * r21002
 70: T = 70 * r21002
 71: T = 71 * r21002

72: T = 72 * r21002
73: T = 73 * r21002
74: T = 74 * r21002
75: T = 75 * r21002
76: T = 76 * r21002
77: T = 77 * r21002
78: T = 78 * r21002
79: T = 79 * r21002
80: T = 80 * r21002
81: T = 81 * r21002
82: T = 82 * r21002
83: T = 83 * r21002
84: T = 84 * r21002
85: T = 85 * r21002
86: T = 86 * r21002
87: T = 87 * r21002
88: T = 88 * r21002
89: T = 89 * r21002
90: T = 90 * r21002
91: T = 91 * r21002
92: T = 92 * r21002
93: T = 93 * r21002
94: T = 94 * r21002
95: T = 95 * r21002
96: T = 96 * r21002
97: T = 97 * r21002
98: T = 98 * r21002
99: T = 99 * r21002
100: T = 100 * r21002
101: T = 101 * r21002
102: T = 102 * r21002
103: T = 103 * r21002
104: T = 104 * r21002
105: T = 105 * r21002
106: T = 106 * r21002
107: T = 107 * r21002
108: T = 108 * r21002
109: T = 109 * r21002
110: T = 110 * r21002
111: T = 111 * r21002
112: T = 112 * r21002
113: T = 113 * r21002
114: T = 114 * r21002
115: T = 115 * r21002
116: T = 116 * r21002
117: T = 117 * r21002
118: T = 118 * r21002
119: T = 119 * r21002
120: T = 120 * r21002
121: T = 121 * r21002
122: T = 122 * r21002
123: T = 123 * r21002
124: T = 124 * r21002
125: T = 125 * r21002
126: T = 126 * r21002
127: T = 127 * r21002
128: T = 128 * r21002
129: T = 129 * r21002
130: T = 130 * r21002
131: T = 131 * r21002

132: T = 132 * r21002
133: T = 133 * r21002
134: T = 134 * r21002
135: T = 135 * r21002
136: T = 136 * r21002
137: T = 137 * r21002
138: T = 138 * r21002
139: T = 139 * r21002
140: T = 140 * r21002
141: T = 141 * r21002
142: T = 142 * r21002
143: T = 143 * r21002
144: T = 144 * r21002
145: T = 145 * r21002
146: T = 146 * r21002
147: T = 147 * r21002
148: T = 148 * r21002
149: T = 149 * r21002
150: T = 150 * r21002
151: T = 151 * r21002
152: T = 152 * r21002
153: T = 153 * r21002
154: T = 154 * r21002
155: T = 155 * r21002
156: T = 156 * r21002
157: T = 157 * r21002
158: T = 158 * r21002
159: T = 159 * r21002
160: T = 160 * r21002
161: T = 161 * r21002
162: T = 162 * r21002
163: T = 163 * r21002
164: T = 164 * r21002
165: T = 165 * r21002
166: T = 166 * r21002
167: T = 167 * r21002
168: T = 168 * r21002
169: T = 169 * r21002
170: T = 170 * r21002
171: T = 171 * r21002
172: T = 172 * r21002
173: T = 173 * r21002
174: T = 174 * r21002
175: T = 175 * r21002
176: T = 176 * r21002
177: T = 177 * r21002
178: T = 178 * r21002
179: T = 179 * r21002
180: T = 180 * r21002
181: T = 181 * r21002
182: T = 182 * r21002
183: T = 183 * r21002
184: T = 184 * r21002
185: T = 185 * r21002
186: T = 186 * r21002
187: T = 187 * r21002
188: T = 188 * r21002
189: T = 189 * r21002
190: T = 190 * r21002
191: T = 191 * r21002

192: T = 192 * r21002
193: T = 193 * r21002
194: T = 194 * r21002
195: T = 195 * r21002
196: T = 196 * r21002
197: T = 197 * r21002
198: T = 198 * r21002
199: T = 199 * r21002
200: T = 200 * r21002
201: T = 201 * r21002
202: T = 202 * r21002
203: T = 203 * r21002
204: T = 204 * r21002
205: T = 205 * r21002
206: T = 206 * r21002
207: T = 207 * r21002
208: T = 208 * r21002
209: T = 209 * r21002
210: T = 210 * r21002
211: T = 211 * r21002
212: T = 212 * r21002
213: T = 213 * r21002
214: T = 214 * r21002
215: T = 215 * r21002
216: T = 216 * r21002
217: T = 217 * r21002
218: T = 218 * r21002
219: T = 219 * r21002
220: T = 220 * r21002
221: T = 221 * r21002
222: T = 222 * r21002
223: T = 223 * r21002
224: T = 224 * r21002
225: T = 225 * r21002
226: T = 226 * r21002
227: T = 227 * r21002
228: T = 228 * r21002
229: T = 229 * r21002
230: T = 230 * r21002
231: T = 231 * r21002
232: T = 232 * r21002
233: T = 233 * r21002
234: T = 234 * r21002
235: T = 235 * r21002
236: T = 236 * r21002
237: T = 237 * r21002
238: T = 238 * r21002
239: T = 239 * r21002
240: T = 240 * r21002
241: T = 241 * r21002
242: T = 242 * r21002
243: T = 243 * r21002
244: T = 244 * r21002
245: T = 245 * r21002
246: T = 246 * r21002
247: T = 247 * r21002
248: T = 248 * r21002
249: T = 249 * r21002
250: T = 250 * r21002
251: T = 251 * r21002

252: T = 252 * r21002
 253: T = 253 * r21002
 254: T = 254 * r21002
 255: T = 255 * r21002
 256: T = 256 * r21002
 1001: T = 1 * r21003
 1002: T = 2 * r21003
 1003: T = 3 * r21003
 1004: T = 4 * r21003
 1005: T = 5 * r21003
 1006: T = 6 * r21003
 1008: T = 8 * r21003
 1010: T = 10 * r21003
 1012: T = 12 * r21003
 1016: T = 16 * r21003
 1020: T = 20 * r21003
 1024: T = 24 * r21003
 1032: T = 32 * r21003
 1040: T = 40 * r21003
 1048: T = 48 * r21003
 1064: T = 64 * r21003
 1080: T = 80 * r21003
 1096: T = 96 * r21003
 3001: BEFORE speed ctrl
 3003: BEFORE speed setpoint channel
 3006: BEFORE standard technology controller
 4000: Receive AFTER IF1 PROFIdrive PZD
 4001: Send BEFORE IF1 PROFIdrive PZD
 4002: Receive AFTER IF2 PZD
 4003: Send BEFORE IF2 PZD
 4004: Receive AFTER IF1 PROFIdrive flexible PZD
 4005: Receive AFTER IF2 flexible PZD

Index: [0]: Run-time group 1
 [1]: Run-time group 2
 [2]: Run-time group 3
 [3]: Run-time group 4
 [4]: Run-time group 5
 [5]: Run-time group 6
 [6]: Run-time group 7
 [7]: Run-time group 8
 [8]: Run-time group 9
 [9]: Run-time group 10

Caution: The properties of the run-time groups must not be changed during operation as this could result in discontinuous signal transitions.

Note: Re value = 1 ... 256 (free run-time group): This selection value can only be selected online if the following applies for sampling time T_{sample} of this run-time group: $1 \text{ ms} \leq T_{\text{sample}} < r21003$. At download, a value that violates this condition is not rejected, but a permissible equivalent value is set automatically and fault F51004 is output. Re value > 2000 (fixed run-time group): The fixed run-time groups $p21000[x] \geq 2000$ log on with the sampling time of the associated basic system function, subject to a minimum sampling time of 1 ms. If, as a result of this limit, the actual sampling time deviates from the sampling time of the basic system function, then fault F51005 (during F51006 download) is output. In this case, another run-time group with a sampling time $\geq 1 \text{ ms}$ should be selected. When selecting the fixed run-time groups, a check is not made as to whether the associated system block exists. Example: "BEFORE speed setpoint channel" means before function charts 3010, 3020, 3030, 3040, etc. are calculated, if the setpoint channel is activated. If, e.g. for SERVO, a

setpoint channel has not been configured (p0108.8 = 0), the calculation is made before function chart 3095.

r21001[0..9]	Run-time group sampling time / RTG sampling time		
All objects	Changeable: - Data type: FloatingPoint32 P group: - Not for motor type: - Min - [ms]	Calculated: - Dynamic index: - Units group: - Max - [ms]	Access level: 1 Function plan: - Units selection: - Expert list: 1 Factory setting [] - [ms]
Description:	Displays the current sampling time of the run-time groups.		
Index:	[0]: Run-time group 1 [1]: Run-time group 2 [2]: Run-time group 3 [3]: Run-time group 4 [4]: Run-time group 5 [5]: Run-time group 6 [6]: Run-time group 7 [7]: Run-time group 8 [8]: Run-time group 9 [9]: Run-time group 10		
r21002	Basis sampling time, hardware / Basis samp time HW		
All objects	Changeable: - Data type: FloatingPoint32 P group: - Not for motor type: - Min - [ms]	Calculated: - Dynamic index: - Units group: - Max - [ms]	Access level: 1 Function plan: - Units selection: - Expert list: 1 Factory setting [] - [ms]
Description:	Displays the basis sampling time effective at this drive object for values 1 to 256 of p21000. Sampling time $T = p21000 * r21002$		
r21003	Basis sampling time, software / Basis samp time SW		
All objects	Changeable: - Data type: FloatingPoint32 P group: - Not for motor type: - Min - [ms]	Calculated: - Dynamic index: - Units group: - Max - [ms]	Access level: 1 Function plan: - Units selection: - Expert list: 1 Factory setting [] - [ms]
Description:	Displays the basis sampling time effective at this drive object for p21000 = 1002 to 1096 as factor. Sampling time $T = (p21000 - 1000) * r21003$		
r21005[0..9]	Computing time load of the run-time group / RTG load		
All objects	Changeable: - Data type: FloatingPoint32 P group: - Not for motor type: - Min - [%]	Calculated: - Dynamic index: - Units group: - Max - [%]	Access level: 3 Function plan: - Units selection: - Expert list: 1 Factory setting [] - [%]

Description: Percentage of the computing time load that the DCC run-time group uses in the sampling time in which it is called.

Index: [0]: Run-time group 1
[1]: Run-time group 2
[2]: Run-time group 3
[3]: Run-time group 4
[4]: Run-time group 5
[5]: Run-time group 6
[6]: Run-time group 7
[7]: Run-time group 8
[8]: Run-time group 9
[9]: Run-time group 10

Note: The computing time load can only be displayed for the run-time groups which are logged on (p21000[x] > 0). The value of the computing time load is calculated in the drive device using the loaded project with DCC chart. As a consequence, the values r21005[x] are not available in the offline mode of SCOUT/STARTER. In r21005 the computing time load is displayed, with which the DCC runtime group utilizes the sampling time in which it is called. The runtime groups "Receive AFTER IF1 PROFIdrive PZD" (p21000 = 4000), "Send BEFORE IF1 PROFIdrive PZD" (p21000 = 4001), "Receive BEFORE IF2 PZD" (p21000 = 4002) and "Send BEFORE IF2 PZD" (p21000 = 4003) are called in the isochronous mode and in the non-isochronous mode, in different sampling times. In the non-isochronous mode, these are IF1 / IF2 PZD sampling time (p2048 for p21000 = 4000 or 4001, p8848 for p21000 = 4002 or 4003). In the isochronous mode, this is the current controller sampling time (p115[0]) which is periodically called with the isochronous bus cycle time. The computing time load displayed in r21005 is always calculated for the (more unfavorable) case of isochronous operation. This is why this value does not (always) act to the full amount on the computing time load of the complete system.

r21008[0...24] Hardware sampling times available / HW t_samp

All objects Changeable: - Calculated: - Access level: 3
Data type: FloatingPoint32 Dynamic index: - Function plan: -
P group: - Units group: - Units selection: -
Not for motor type: - Expert list: 1
Min Max Factory setting
- [ms] - [ms] [] - [ms]

Description: Displays the assignment of the available hardware sampling times of the drive unit.
The designated sampling times are those created as a multiple of the hardware basis sampling time (r21002) and which are always < r21003.

Index: [0]: Hardware 1
[1]: Hardware 2
[2]: Hardware 3
[3]: Hardware 4
[4]: Hardware 5
[5]: Hardware 6
[6]: Hardware 7
[7]: Hardware 8
[8]: Hardware 9
[9]: Hardware 10
[10]: Hardware 11
[11]: Hardware 12
[12]: Hardware 13
[13]: Hardware 14
[14]: Hardware 15

B.2 参数

- [15]: Hardware 16
- [16]: Hardware 17
- [17]: Hardware 18
- [18]: Hardware 19
- [19]: Hardware 20
- [20]: Hardware 21
- [21]: Hardware 22
- [22]: Hardware 23
- [23]: Hardware 24
- [24]: Hardware 25

See also: F51001

Attention: For internal purposes, the drive unit always requires at least two (or several, depending on how p0115[0] is parameterized for the drive objects) free hardware sampling times. Therefore the current number of free hardware sampling times can be read out in r7903. If r7903=0, no additional sampling time different from r21008[0..24] may be provided from the Control Unit. When selecting in this state, if a run-time group with a sampling time < r21003 (p21000 <= 255) is selected in p21000, only run-time groups whose sampling time is already provided in r21008[0..24] may be selected.

Note: A sampling time that is provided can be simultaneously used by system functions, several FBLOCK run-time groups and several DCC run-time groups.

The sampling time of run-time groups that have been assigned to the PROFIBUS run-time groups (p21000 = 4000 ... 4004) is not displayed in r21008. For this sampling time, one of the internally and permanently assigned hardware sampling times is used. If the value of r21008[x] != 0 (not equal to 0), then the sampling time is specified in ms. If the value of r21008[x] = 0, this sampling time can still be freely assigned. It should be noted that the basic system, depending on the selected basis sampling times p0115[0], requires at least two (sometimes more) freely assignable hardware sampling times for internal functions. The number of hardware sampling times that can still be freely assigned can be read out in r7903. If the value r21008[x] = 99999.00000, this hardware sampling time is not supported.

p21030	Run-time group, computing time measurement / RTG comp_ti_meas		
All objects	Changeable: U T	Calculated: -	Access level: 4
	Data type: Unsigned16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	0	10	[0] 0

Description: Only for internal Siemens service purposes.

See also: p21032, r21035, r21036, r21037

p21032	Computing time measurement, duration / Comp_ti_meas dur.		
All objects	Changeable: U T	Calculated: -	Access level: 4
	Data type: Unsigned16	Dynamic index: -	Function plan: -
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	60 [s]	10000 [s]	[0] 60 [s]

Description: Only for internal Siemens service purposes.

See also: p21030, r21035, r21036, r21037

r21035[0..9]	Computing time, minimum value / Computing time min		
All objects	Changeable: -	Calculated: -	Access level: 4
	Data type:	Dynamic index: -	Function plan: -
	FloatingPoint32		
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	- [μs]	- [μs]	[] - [μs]
Description:	Only for internal Siemens service purposes.		
Index:	[0]:	Run-time group 1	
	[1]:	Run-time group 2	
	[2]:	Run-time group 3	
	[3]:	Run-time group 4	
	[4]:	Run-time group 5	
	[5]:	Run-time group 6	
	[6]:	Run-time group 7	
	[7]:	Run-time group 8	
	[8]:	Run-time group 9	
	[9]:	Run-time group 10	
See also:	p21030, p21032, r21036, r21037		

r21036[0..9]	Computing time, mean value / Computing tim av		
All objects	Changeable: -	Calculated: -	Access level: 4
	Data type:	Dynamic index: -	Function plan: -
	FloatingPoint32		
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	- [μs]	- [μs]	[] - [μs]
Description:	Only for internal Siemens service purposes.		
Index:	[0]:	Run-time group 1	
	[1]:	Run-time group 2	
	[2]:	Run-time group 3	
	[3]:	Run-time group 4	
	[4]:	Run-time group 5	
	[5]:	Run-time group 6	
	[6]:	Run-time group 7	
	[7]:	Run-time group 8	
	[8]:	Run-time group 9	
	[9]:	Run-time group 10	

r21037[0..9]	Computing time, maximum value / Computing time max		
All objects	Changeable: -	Calculated: -	Access level: 4
	Data type:	Dynamic index: -	Function plan: -
	FloatingPoint32		
	P group: -	Units group: -	Units selection: -
	Not for motor type: -		Expert list: 1
	Min	Max	Factory setting
	- [μs]	- [μs]	[] - [μs]
Description:	Only for internal Siemens service purposes.		
Index:	[0]:	Run-time group 1	
	[1]:	Run-time group 2	
	[2]:	Run-time group 3	
	[3]:	Run-time group 4	
	[4]:	Run-time group 5	

B.2 参数

[5]: Run-time group 6
[6]: Run-time group 7
[7]: Run-time group 8
[8]: Run-time group 9
[9]: Run-time group 10

See also: p21030, p21032, r21035, r21036

索引

A

ACOS (SIMOTION) , 29
ADD (SIMOTION, SINAMICS) , 31
ADD_D (SIMOTION, SINAMICS) , 32
ADD_I (SIMOTION, SINAMICS) , 33
ADD_M (SIMOTION, SINAMICS) , 34
AND (SIMOTION, SINAMICS) , 71
AND_W (SIMOTION) , 73
ASIN (SIMOTION) , 36
ATAN (SIMOTION) , 38
AVA (SIMOTION, SINAMICS) , 40
AVA_D (SIMOTION, SINAMICS) , 42

B

B_BY (SIMOTION) , 178
B_DW (SIMOTION, SINAMICS) , 180
B_W (SIMOTION, SINAMICS) , 183
BF (SIMOTION, SINAMICS) , 75
BF_W (SIMOTION) , 77
BSW (SIMOTION, SINAMICS) , 79
BY_B (SIMOTION) , 173
BY_W (SIMOTION, SINAMICS) , 176

C

CNM (SIMOTION, SINAMICS) , 81
CNM_D (SIMOTION, SINAMICS) , 83
CNM_I (SIMOTION, SINAMICS) , 85
COS (SIMOTION) , 44
CTD (SIMOTION) , 233
CTR (SIMOTION, SINAMICS) , 87

D

D_I (SIMOTION, SINAMICS) , 191
D_R (SIMOTION, SINAMICS) , 192
D_SI (SIMOTION) , 193
D_UI (SIMOTION, SINAMICS) , 194
D_US (SIMOTION, SINAMICS) , 195

DCA (SIMOTION, SINAMICS) , 309
DEL (SIMOTION, SINAMICS) , 323
DEZ (SIMOTION, SINAMICS) , 326
DFR (SIMOTION, SINAMICS) , 90
DFR_W (SIMOTION) , 92
DIF (SIMOTION, SINAMICS) , 329
DIV (SIMOTION, SINAMICS) , 46
DIV_D (SIMOTION, SINAMICS) , 48
DIV_I (SIMOTION, SINAMICS) , 50
DLB (SIMOTION, SINAMICS) , 94
DT1 (SIMOTION, SINAMICS) , 332
DW_B (SIMOTION, SINAMICS) , 186
DW_R (SIMOTION, SINAMICS) , 189
DW_W (SIMOTION, SINAMICS) , 190
DX8 (SIMOTION, SINAMICS) , 96
DX8_D (SIMOTION, SINAMICS) , 98
DX8_I (SIMOTION, SINAMICS) , 100

E

ETE (SIMOTION, SINAMICS) , 102

G

GTS (SIMOTION) , 235

I

I_D (SIMOTION, SINAMICS) , 196
I_R (SIMOTION, SINAMICS) , 197
I_SI (SIMOTION) , 198
I_UD (SIMOTION, SINAMICS) , 199
I_US (SIMOTION, SINAMICS) , 200
INCO (SIMOTION, SINAMICS) , 313
INT (SIMOTION, SINAMICS) , 335

L

LIM (SIMOTION, SINAMICS) , 338
LIM_D (SIMOTION, SINAMICS) , 340
LR_R (SIMOTION) , 201

LVM (SIMOTION, SINAMICS) , 104

M

MAS (SIMOTION, SINAMICS) , 52

MFP (SIMOTION, SINAMICS) , 106

MIS (SIMOTION, SINAMICS) , 53

MUL (SIMOTION, SINAMICS) , 54

MUL_D (SIMOTION, SINAMICS) , 55

MUL_I (SIMOTION, SINAMICS) , 56

MUX8 (SIMOTION, SINAMICS) , 108

MUX8_D (SIMOTION, SINAMICS) , 111

MUX8_I (SIMOTION, SINAMICS) , 114

MVS (SIMOTION, SINAMICS) , 342

N

N2_R (SIMOTION, SINAMICS) , 202

N4_R (SIMOTION, SINAMICS) , 203

NAND (SIMOTION, SINAMICS) , 117

NCM (SIMOTION, SINAMICS) , 119

NCM_D (SIMOTION, SINAMICS) , 120

NCM_I (SIMOTION, SINAMICS) , 121

NOP1 (SIMOTION, SINAMICS) , 122

NOP1_B (SIMOTION, SINAMICS) , 123

NOP1_D (SIMOTION, SINAMICS) , 124

NOP1_I (SIMOTION, SINAMICS) , 125

NOP8 (SIMOTION, SINAMICS) , 126

NOP8_B (SIMOTION, SINAMICS) , 128

NOP8_D (SIMOTION, SINAMICS) , 130

NOP8_I (SIMOTION, SINAMICS) , 132

NOR (SIMOTION, SINAMICS) , 134

NOT (SIMOTION, SINAMICS) , 136

NOT_W (SIMOTION) , 137

NSW (SIMOTION, SINAMICS) , 139

NSW_D (SIMOTION, SINAMICS) , 141

NSW_I (SIMOTION, SINAMICS) , 143

O

OCA (SIMOTION, SINAMICS) , 315

OR (SIMOTION, SINAMICS) , 145

OR_W (SIMOTION) , 147

P

PC (SIMOTION, SINAMICS) , 344

PCL (SIMOTION, SINAMICS) , 149

PDE (SIMOTION, SINAMICS) , 151

PDF (SIMOTION, SINAMICS) , 153

PIC (SIMOTION, SINAMICS) , 347

PLI20 (SIMOTION, SINAMICS) , 57

PST (SIMOTION, SINAMICS) , 155

PT1 (SIMOTION, SINAMICS) , 356

R

R_D (SIMOTION, SINAMICS) , 204

R_DW (SIMOTION, SINAMICS) , 205

R_I (SIMOTION, SINAMICS) , 206

R_LR (SIMOTION) , 207

R_N2 (SIMOTION, SINAMICS) , 208

R_N4 (SIMOTION, SINAMICS) , 209

R_SI (SIMOTION) , 210

R_UD (SIMOTION, SINAMICS) , 211

R_UI (SIMOTION, SINAMICS) , 212

R_US (SIMOTION, SINAMICS) , 213

RAA (SIMOTION) , 236

RDA (SIMOTION) , 237

RDAA (SIMOTION) , 239

RDP (SINAMICS) , 240

RDP_D (SINAMICS) , 242

RDP_I (SINAMICS) , 244

RDP_UD (SINAMICS) , 246

RDP_UI (SINAMICS) , 248

RDP_US (SINAMICS) , 250

RGE (SIMOTION, SINAMICS) , 359

RGJ (SIMOTION, SINAMICS) , 367

RMDP (SIMOTION) , 252

RSR (SIMOTION, SINAMICS) , 157

RSS (SIMOTION, SINAMICS) , 159

S

SAH (SINAMICS) , 259

SAH_B (SINAMICS) , 262

SAH_BY (SINAMICS) , 265

SAH_D (SINAMICS) , 268
SAH_I (SINAMICS) , 271
SAV (SIMOTION, SINAMICS) , 274
SAV_BY (SIMOTION, SINAMICS) , 277
SAV_D (SIMOTION, SINAMICS) , 280
SAV_I (SIMOTION, SINAMICS) , 283
SH (SIMOTION) , 161
SH_DW (SIMOTION, SINAMICS) , 163
SI_D (SIMOTION) , 214
SI_I (SIMOTION) , 215
SI_R (SIMOTION) , 216
SI_UD (SIMOTION) , 217
SI_UI (SIMOTION) , 218
SII (SIMOTION, SINAMICS) , 61
SIN (SIMOTION) , 63
SQR (SIMOTION) , 65
SRA (SIMOTION) , 286
STM (SINAMICS) , 288
SUB (SIMOTION, SINAMICS) , 66
SUB_D (SIMOTION, SINAMICS) , 67
SUB_I (SIMOTION, SINAMICS) , 68

T

TAN (SIMOTION) , 69
TRK (SIMOTION, SINAMICS) , 165
TRK_D (SIMOTION, SINAMICS) , 167
TTCU (SIMOTION, SINAMICS) , 317

U

UD_I (SIMOTION, SINAMICS) , 219
UD_R (SIMOTION, SINAMICS) , 220
UD_SI (SIMOTION) , 221
UI_D (SIMOTION, SINAMICS) , 222
UI_R (SIMOTION, SINAMICS) , 223
UI_SI (SIMOTION) , 224
US_D (SIMOTION, SINAMICS) , 225
US_I (SIMOTION, SINAMICS) , 226
US_R (SIMOTION, SINAMICS) , 227

W

W_B (SIMOTION, SINAMICS) , 228
W_BY (SIMOTION, SINAMICS) , 230
W_DW (SIMOTION, SINAMICS) , 232
WBG (SIMOTION, SINAMICS) , 319
WMDP (SIMOTION) , 291
WRP (SINAMICS) , 297
WRP_D (SINAMICS) , 299
WRP_I (SINAMICS) , 301
WRP_UD (SINAMICS) , 303
WRP_UI (SINAMICS) , 305
WRP_US (SINAMICS) , 307

X

XOR (SIMOTION, SINAMICS) , 169
XOR_W (SIMOTION) , 170

Siemens AG

Automation and Drives
Motion Control Systems
Postfach 3180
91050 ERLANGEN
GERMANY
www.siemens.com/motioncontrol