

SIEMENS

SIMATIC

过程控制系统 PCS 7 PCS 7 标准库 V71




功能手册

关于块描述的常规信息	1
系列: COMM	2
系列: CONTROL	3
系列: DRIVER	4
系列: MAINT	5
系列: @SYSTEM	6
系列: TIME	7
系列: MATH	8
系列: CONVERT	9
系列: OPERATE	10
系列: MESSAGE	11
面板	12
块图标	13
附录	14
PCS 7 高级过程控制模板	15

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

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

按规定使用 Siemens 产品

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。

目录

1	关于块描述的常规信息.....	17
2	系列: COMM	23
2.1	REC_BO: 使用 BRCV 接收 128 个 BOOLEAN 值.....	23
2.1.1	REC_BO 的描述.....	23
2.1.2	REC_BO 的 I/O.....	26
2.2	REC_R: 使用 BRCV 接收 32 个 BOOLEAN 值和 32 个 REAL 值.....	27
2.2.1	REC_R 的描述.....	27
2.2.2	REC_R 的 I/O.....	29
2.3	SEND_BO: 使用 BSEND 发送 128 个 BOOLEAN 值.....	31
2.3.1	SEND_BO 的描述.....	31
2.3.2	SEND_BO 的 I/O.....	34
2.4	SEND_R: 使用 BSEND 发送 32 个 BOOL 值和 32 个 REAL 值 (受变化驱使).....	35
2.4.1	SEND_R 的描述.....	35
2.4.2	SEND_R 的 I/O.....	38
3	系列: CONTROL	41
3.1	使用 PID 调谐优化控制器.....	41
3.2	CPM: 监视控制回路性能.....	42
3.2.1	CPM 的描述.....	42
3.2.2	CPM 模式.....	48
3.2.3	CPM 的功能.....	48
3.2.4	CPM 故障处理.....	59
3.2.5	CPM 报告.....	60
3.2.6	CPM I/O.....	61
3.3	CTRL_PID: PID 控制器块.....	65
3.3.1	CTRL_PID 的描述.....	65
3.3.2	CTRL_PID 的设定值和实际值分支中的信号处理.....	68
3.3.3	CTRL_PID 调节变量的生成.....	70
3.3.4	CTRL_PID 的手动、自动和跟踪模式及级联.....	72
3.3.5	CTRL_PID 模式更改.....	74
3.3.6	CTRL_PID 的错误处理.....	76
3.3.7	CTRL_PID 的启动行为、动态响应和消息响应.....	77
3.3.8	CTRL_PID 的方框图.....	78
3.3.9	CTRL_PID 的 I/O.....	79
3.3.10	CTRL_PID 的消息文本和关联值.....	84
3.3.11	CTRL_PID 的 VSTATUS.....	86
3.3.12	CTRL_PID 的操作和监视.....	86

3.4	CTRL_S: PID 步控制器块	87
3.4.1	CTRL_S 的描述	87
3.4.2	CTRL_S 的设定值和实际值分支中的信号处理	91
3.4.3	CTRL_S 控制信号的生成	93
3.4.4	CTRL_S 的手动、自动和跟踪模式及级联	97
3.4.5	CTRL_S 模式更改	100
3.4.6	CTRL_S 的错误处理	103
3.4.7	CTRL_S 的启动行为、动态响应和消息响应	104
3.4.8	CTRL_S 的方框图	105
3.4.9	CTRL_S 的 I/O	107
3.4.10	CTRL_S 的消息文本和关联值	114
3.4.11	CTRL_S 的 VSTATUS	116
3.4.12	CTRL_S 的操作和监视	116
3.5	DEADT_P: 停滞时间元件	117
3.5.1	DEADT_P 的描述	117
3.5.2	DEADT_P 的 I/O	119
3.6	DIF_P: 微分	120
3.6.1	DIF_P 的描述	120
3.6.2	DIF_P 的 I/O	122
3.7	DIG_MON: 数字值监视	123
3.7.1	DIG_MON 的描述	123
3.7.2	DIG_MON 的 I/O	126
3.7.3	DIG_MON 的消息文本和关联值	128
3.7.4	DIG_MON 的 VSTATUS	130
3.7.5	DIG_MON 的操作和监视	130
3.8	DOSE: 定量给料过程	131
3.8.1	DOSE 的描述	131
3.8.2	DOSE 的 I/O	136
3.8.3	DOSE 的消息文本和关联值	140
3.8.4	DOSE 的 VSTATUS	142
3.8.5	DOSE 的操作员监控	142
3.9	ELAP_CNT: 运行时间计数器	143
3.9.1	ELAP_CNT 的描述	143
3.9.2	ELAP_CNT 的 I/O	145
3.9.3	ELAP_CNT 的消息文本和关联值	147
3.9.4	ELAP_CNT 的 VSTATUS	148
3.9.5	ELAP_CNT 的操作员监控	148

3.10	FMCS_PID: 控制器块	149
3.10.1	FMCS_PID 的描述.....	149
3.10.2	寻址.....	152
3.10.3	FMCS_PID 的功能.....	152
3.10.4	通过过程映像采集和写入过程值	154
3.10.5	设定值、限制值、误差信号和调节变量的生成.....	156
3.10.6	FMCS_PID 的手动、自动和跟踪模式	158
3.10.7	FMCS_PID 的模式更改	160
3.10.8	安全模式.....	161
3.10.9	将参数下载到模块.....	161
3.10.10	从模块中读取数据.....	162
3.10.11	FMCS_PID 的错误处理	162
3.10.12	FMCS_PID 的启动行为、动态响应和消息响应.....	163
3.10.13	FM 355 的备份模式	164
3.10.14	FMCS_PID 的 I/O.....	165
3.10.15	FMCS_PID 的消息文本和关联值	172
3.10.16	FMCS_PID 的 VSTATUS	174
3.10.17	FMCS_PID 的操作和监视.....	174
3.11	FMT_PID: 温度控制器块.....	175
3.11.1	FMT_PID 的描述	175
3.11.2	寻址.....	178
3.11.3	FMT_PID 的功能	178
3.11.4	通过过程映像采集和写入过程值	180
3.11.5	设定值、限制值、误差信号和调节变量的生成.....	182
3.11.6	FMT_PID 的手动、自动和跟踪模式.....	184
3.11.7	FMT_PID 的模式更改	186
3.11.8	安全模式.....	187
3.11.9	将参数下载到模块.....	187
3.11.10	从模块中读取数据/使用组态工具	188
3.11.11	优化.....	189
3.11.12	在两个不同的 PID 参数集间切换	190
3.11.13	FMT_PID 的错误处理	190
3.11.14	FMT_PID 的启动行为、动态响应和消息响应	191
3.11.15	FM 355-2 的备份模式	193
3.11.16	FMT_PID 的 I/O.....	193
3.11.17	FMT_PID 的消息文本和关联值	200
3.11.18	FMT_PID 的 VSTATUS	202
3.11.19	FMT_PID 的操作和监视.....	202

3.12	GAIN_SHD - 增益调整	203
3.12.1	GAIN_SHD 的描述	203
3.12.2	GAIN_SHD 的功能	206
3.12.3	GAIN_SHD I/O	207
3.13	INT_P: 积分 (Integration)	209
3.13.1	INT_P 的描述	209
3.13.2	INT_P 的 I/O	212
3.14	INTERLOK: 状态显示锁	213
3.14.1	INTERLOK 的描述	213
3.14.2	INTERLOK 的 I/O	215
3.14.3	INTERLOK 的 VSTATUS	217
3.14.4	INTERLOK 的操作员监控	217
3.15	LIMITS_P: 限制	218
3.15.1	LIMITS_P 的描述	218
3.15.2	LIMITS_P 的 I/O	220
3.16	MEAS_MON: 测量值监视	221
3.16.1	MEAS_MON 的描述	221
3.16.2	MEAS_MON 的 I/O	223
3.16.3	MEAS_MON 的消息文本和关联值	225
3.16.4	MEAS_MON 的 VSTATUS	226
3.16.5	MEAS_MON 的操作和监视	226
3.17	MOT_REV: 双向电机	227
3.17.1	MOT_REV 的描述	227
3.17.2	MOT_REV 的 I/O	232
3.17.3	MOT_REV 的消息文本和关联值	235
3.17.4	MOT_REV 的 VSTATUS	236
3.17.5	MOT_REV 的操作和监视	236
3.18	MOT_SPED: 双速电机	237
3.18.1	MOT_SPED 的描述	237
3.18.2	MOT_SPED 的 I/O	242
3.18.3	MOT_SPED 的消息文本和关联值	246
3.18.4	MOT_SPED 的 VSTATUS	247
3.18.5	MOT_SPED 的操作和监视	247
3.19	MOTOR: 具有一个控制信号的电机	248
3.19.1	MOTOR 的描述	248
3.19.2	MOTOR 的 I/O	253
3.19.3	MOTOR 的消息文本和关联值	256
3.19.4	MOTOR 的 VSTATUS	257
3.19.5	MOTOR 的操作和监视	257

3.20	MPC: 模型预测控制器.....	258
3.20.1	MPC 的描述.....	258
3.20.2	MPC 模式.....	263
3.20.3	MPC 的功能.....	265
3.20.4	MPC 错误处理.....	274
3.20.5	MPC I/O.....	275
3.21	NOISE_GN: 噪声信号发生器.....	278
3.21.1	NOISE_GN 描述.....	278
3.22	POLYG_P: 最多有 8 个点的多边形.....	279
3.22.1	POLYG_P 的描述.....	279
3.22.2	POLYG_P 的 I/O.....	280
3.23	PT1_P: 一阶滞后元件.....	281
3.23.1	PT1_P 的描述.....	281
3.23.2	PT1_P 的 I/O.....	283
3.24	RAMP_P: 斜坡生成.....	284
3.24.1	RAMP_P 的描述.....	284
3.24.2	RAMP_P 的 I/O.....	286
3.25	RATIO_P: 比率控制.....	287
3.25.1	RATIO_P 的描述.....	287
3.25.2	RATIO_P 的 I/O.....	289
3.25.3	RATIO_P 的 VSTATUS.....	290
3.25.4	RATIO_P 的操作和监视.....	290
3.26	READ355P: 从 FM 355 读取数字和模拟输入.....	291
3.26.1	READ355P 的描述.....	291
3.26.2	寻址.....	293
3.26.3	READ355P 的 I/O.....	293
3.27	SIG_SMTH: 低通滤波器.....	294
3.27.1	SIG_SMTH 描述.....	294
3.27.2	SIG_SMTH 的模式.....	295
3.27.3	SIG_SMTH 的功能.....	296
3.27.4	SIG_SMTH 的故障处理.....	297
3.27.5	SIG_SMTH I/O.....	298
3.28	SPLITR_P: 分割范围.....	300
3.28.1	SPLITR_P 的描述.....	300
3.28.2	SPLITR_P 的 I/O.....	302

3.29	SWIT_CNT: 切换周期计数器	303
3.29.1	SWIT_CNT 的描述	303
3.29.2	SWIT_CNT 的 I/O	305
3.29.3	SWIT_CNT 的消息文本和关联值	306
3.29.4	SWIT_CNT 的 VSTATUS	307
3.29.5	SWIT_CNT 的操作员监控	307
3.30	VAL_MOT: 电机阀门控制	308
3.30.1	VAL_MOT 的描述	308
3.30.2	VAL_MOT 的 I/O	314
3.30.3	VAL_MOT 的消息文本和关联值	317
3.30.4	VAL_MOT 的 VSTATUS	318
3.30.5	VAL_MOT 的操作和监视	318
3.31	VALVE: 阀控制	319
3.31.1	VALVE 的描述	319
3.31.2	VALVE 的 I/O	323
3.31.3	VALVE 的消息文本和关联值	326
3.31.4	VALVE 的 VSTATUS	327
3.31.5	VALVE 的操作和监视	327
4	系列: DRIVER	329
4.1	关于使用驱动程序块的注意事项	329
4.2	CH_AI: 模拟值输入	331
4.2.1	CH_AI 的描述	331
4.2.2	CH_AI 的 I/O	338
4.3	CH_AO: 模拟值输出	340
4.3.1	CH_AO 的描述	340
4.3.2	CH_AO 的 I/O	344
4.4	CH_CNT: 控制和读取 FM 350 模块	345
4.4.1	CH_CNT 的描述	345
4.4.2	CH_CNT 的 I/O	350
4.5	CH_CNT1: 控制和读取 ET 200iSP 的 8-DI-NAMUR 模块	353
4.5.1	CH_CNT1 的描述	353
4.5.2	CH_CNT1 的 I/O	360
4.6	CH_CNT2C: 控制和读取计数模式下的 1 COUNT 24V/100kHz 模块	362
4.6.1	CH_CNT2C 描述	362
4.6.2	CH_CNT2C 的 I/O	366
4.7	CH_CNT2M: 控制和读取测量模式下的 1 COUNT 24V/100kHz 模块	368
4.7.1	CH_CNT2M 描述	368
4.7.2	CH_CNT2M 的 I/O	372

4.8	CH_DI: 数字值输入	374
4.8.1	CH_DI 的描述	374
4.8.2	CH_DI 的 I/O	377
4.9	CH_DO: 数字值输出	378
4.9.1	CH_DO 的描述	378
4.9.2	CH_DO 的 I/O	381
4.10	CH_MS: ET 200S 电机起动器模块的信号处理	382
4.10.1	CH_MS 的描述	382
4.10.2	CH_MS 的 I/O	387
4.11	CH_U_AI: 模拟值输入 (通用)	389
4.11.1	CH_U_AI (通用) 的描述	389
4.11.2	CH_U_AI 的 I/O	396
4.12	CH_U_AO: 模拟值输出 (通用)	398
4.12.1	CH_U_AO 的描述	398
4.12.2	CH_U_AO 的 I/O	403
4.13	CH_U_DI: 数字值输入 (通用)	405
4.13.1	CH_U_DI (通用) 的描述	405
4.13.2	CH_U_DI 的 I/O	409
4.14	CH_U_DO: 数字值输出 (通用)	411
4.14.1	CH_U_DO (通用) 的描述	411
4.14.2	CH_U_DO 的 I/O	415
4.15	FF_A_AI: 使用 PA 配置文件发送器	416
4.15.1	FF_A_AI 的描述	416
4.15.2	FF_A_AI 的 I/O	420
4.16	FF_A_AO: 使用 PA 配置文件执行器	423
4.16.1	FF_A_AO 的描述	423
4.16.2	FF_A_AO 的 I/O	428
4.17	FF_A_DI: 读取数字值	432
4.17.1	FF_A_DI 的描述	432
4.17.2	FF_A_DI 的 I/O	436
4.18	FF_A_DO: 数字值输出	438
4.18.1	FF_A_DO 的描述	438
4.18.2	FF_A_DO 的 I/O	443
4.19	MSG_TS: 生成具有时间戳的过程值	447
4.19.1	MSG_TS 的描述	447
4.19.2	MSG_TS 的 I/O	450
4.19.3	MSG_TS 的消息文本	451

4.20	PA_AI: PROFIBUS PA 模拟值输入.....	452
4.20.1	PA_AI 的描述.....	452
4.20.2	PA_AI 的 I/O.....	456
4.21	PA_AO: PROFIBUS PA 模拟值输出.....	459
4.21.1	PA_AO 的描述.....	459
4.21.2	PA_AO 的 I/O.....	463
4.22	PA_DI: PROFIBUS PA 数字值输入.....	467
4.22.1	PA_DI 的描述.....	467
4.22.2	PA_DI 的 I/O.....	471
4.23	PA_DO: PROFIBUS PA 数字值输出.....	473
4.23.1	PA_DO 的描述.....	473
4.23.2	PA_DO 的 I/O.....	477
4.24	PA_TOT: PROFIBUS PA 总和器.....	481
4.24.1	PA_TOT 的描述.....	481
4.24.2	PA_TOT 的 I/O.....	485
4.25	RCV_341: 在串行模式下通过 CP 341 接收数据.....	488
4.25.1	RCV_341 的描述.....	488
4.25.2	RCV_341 的 I/O.....	493
4.25.3	RCV_341 的消息文本和关联值.....	495
4.26	SND_341: 通过 CP 341 发送串行数据.....	496
4.26.1	SND_341 的描述.....	496
4.26.2	SND_341 的 I/O.....	501
4.26.3	SND_341 的消息文本和关联值.....	503
4.27	内部块.....	504
4.27.1	MODB_341: 内部块.....	504
5	系列: MAINT.....	505
5.1	ASSETMON: 监视是否超出限制值的过程变量.....	505
5.1.1	ASSETMON 的描述.....	505
5.1.2	ASSETMON 的描述.....	511
5.1.3	ASSETMON 的消息文本和关联值.....	513
5.1.4	ASSETMON 的操作员监控.....	514
5.2	MS_MUX: 确定最差的个别状态.....	515
5.2.1	MS_MUX 的描述.....	515
5.2.2	MS_MUX 的 I/O.....	516

5.3	ST_MUX: 确定 FF 现场设备的状态值	517
5.3.1	ST_MUX 的描述	517
5.3.2	ST_MUX 的 I/O	518
5.4	STATEREPI: 块组的状态显示	519
5.4.1	STATEREPI 的描述	519
5.4.2	STATEREPI 的 I/O	520
6	系列: @SYSTEM.....	521
6.1	AL_DELAY - 报警延迟	521
6.1.1	AL_DELAY 描述	521
6.2	内部块	522
6.2.1	P_RCV_RK: 内部块	522
6.2.2	P_SND_RK: 内部块	522
7	系列: TIME	523
7.1	OB1_TIME: 确定 CPU 利用率	523
7.1.1	OB1_TIME 的描述	523
7.1.2	OB1_TIME 的 I/O	525
8	系列: MATH.....	527
8.1	ADD4_P: 最多处理 4 个值的加法器	527
8.1.1	ADD4_P 的描述	527
8.1.2	ADD4_P 的 I/O	528
8.2	ADD8_P: 最多处理 8 个值的加法器	529
8.2.1	ADD8_P 的描述	529
8.2.2	ADD8_P 的 I/O	529
8.3	AVER_P: 平均时间值	530
8.3.1	AVER_P 的描述	530
8.3.2	AVER_P 的 I/O	532
8.4	COUNT_P: 计数器	533
8.4.1	COUNT_P 的描述	533
8.4.2	COUNT_P 的 I/O	535
8.5	MEANTM_P: 计算平均时间值	536
8.5.1	MEANTM_P 的描述	536
8.5.2	MEANTM_P 的 I/O	538
8.6	MUL4_P: 最多处理 4 个值的乘法器	539
8.6.1	MUL4_P 的描述	539
8.6.2	MUL4_P 的 I/O	540
8.7	MUL8_P: 最多处理 8 个值的乘法器	541
8.7.1	MUL8_P 的描述	541
8.7.2	MUL8_P 的 I/O	542

9	系列: CONVERT	543
9.1	有关转换块的常规信息	543
9.2	R_TO_DW: 转换	544
9.2.1	R_TO_DW 的描述	544
9.2.2	R_TO_DW 的 I/O.....	544
10	系列: OPERATE	545
10.1	操作员控制块概述	545
10.2	OP_A: 模拟值操作	551
10.2.1	OP_A 的描述	551
10.2.2	OP_A 的 I/O	553
10.2.3	OP_A 的操作员监控	553
10.3	OP_A_LIM: 模拟值操作 (限制)	554
10.3.1	OP_A_LIM 的描述	554
10.3.2	OP_A_LIM 的 I/O.....	557
10.3.3	OP_A_LIM 的操作员监控	557
10.4	OP_A_RJC: 模拟值操作 (拒绝)	558
10.4.1	OP_A_RJC 的描述	558
10.4.2	OP_A_RJC 的 I/O.....	561
10.4.3	OP_A_RJC 的操作员监控	561
10.5	OP_D: 数字值的操作员输入 (双按钮控制)	562
10.5.1	OP_D 的描述.....	562
10.5.2	OP_D 的 I/O	565
10.5.3	OP_D 的操作员监控	565
10.6	OP_D3: 数字值的操作员输入 (3 个按钮控制)	566
10.6.1	OP_D3 的描述.....	566
10.6.2	OP_D3 的 I/O	570
10.6.3	OP_D3 的操作员监控	571
10.7	OP_TRIG: 数字值的操作员输入 (单按钮控制)	572
10.7.1	OP_TRIG 的描述.....	572
10.7.2	OP_TRIG 的 I/O	574
10.7.3	OP_TRIG 的操作员监控	574

11	系列: MESSAGE	575
11.1	消息块概述	575
11.2	MSG_NACK: 用户特定消息 (不需要确认)	576
11.2.1	MSG_NACK 的描述	576
11.2.2	MSG_NACK 的 I/O	578
11.3	MESSAGE: 消息块 (可组态消息)	579
11.3.1	MESSAGE 的描述	579
11.3.2	MESSAGE 的 I/O	581
11.3.3	MESSAGE 的消息文本和关联值	582
12	面板	583
12.1	面板: 设备块	583
12.1.1	CTRL_PID (所有视图)	583
12.1.2	CTRL_PID: 标准视图	584
12.1.3	CTRL_PID: 维护视图	586
12.1.4	CTRL_PID: 参数视图	588
12.1.5	CTRL_PID: 限制视图	590
12.1.6	CTRL_S (所有视图)	592
12.1.7	CTRL_S: 标准视图	593
12.1.8	CTRL_S: 维护视图	596
12.1.9	CTRL_S: 参数视图	596
12.1.10	CTRL_S: 限制视图	596
12.1.11	CTRL_S: StandardS 视图	596
12.1.12	DIG_MON (所有视图)	599
12.1.13	DIG_MON: 标准视图	599
12.1.14	DOSE (所有视图)	600
12.1.15	DOSE: 标准视图	600
12.1.16	DOSE: 维护视图	603
12.1.17	DOSE: 参数视图	604
12.1.18	DOSE: 限制视图	606
12.1.19	ELAP_CNT (所有视图)	608
12.1.20	ELAP_CNT: 标准视图	608
12.1.21	FMCS_PID (所有视图)	610
12.1.22	FMCS_PID: 标准视图	610
12.1.23	FMCS_PID: 维护视图	613
12.1.24	FMCS_PID: 参数视图	615
12.1.25	FMCS_PID: 限制视图	617
12.1.26	FMCS_PID: StandardS 视图	618
12.1.27	FMT_PID (所有视图)	620
12.1.28	FMT_PID: 标准视图	621
12.1.29	FMT_PID: 维护视图	624
12.1.30	FMT_PID: 参数视图	624

12.1.31	FMT_PID: 限制视图	624
12.1.32	FMT_PID: StandardS 视图	625
12.1.33	INTERLOK: 标准视图	626
12.1.34	MEAS_MON (所有视图)	627
12.1.35	MEAS_MON: 标准视图	627
12.1.36	MEAS_MON: 限制视图	629
12.1.37	MOT_REV (所有视图)	630
12.1.38	MOT_REV: 标准视图	630
12.1.39	MOT_REV: 维护视图	632
12.1.40	MOT_SPED (所有视图)	633
12.1.41	MOT_SPED: 标准视图	633
12.1.42	MOT_SPED: 维护视图	635
12.1.43	MOTOR (所有视图)	636
12.1.44	MOTOR: 标准视图	637
12.1.45	MOTOR: 维护视图	639
12.1.46	RATIO_P (所有视图)	640
12.1.47	RATIO_P: 标准视图	640
12.1.48	RATIO_P: 限制视图	642
12.1.49	SWIT_CNT (所有视图)	643
12.1.50	SWIT_CNT: 标准视图	643
12.1.51	VAL_MOT (所有视图)	645
12.1.52	VAL_MOT: 标准视图	645
12.1.53	VAL_MOT: 维护视图	647
12.1.54	VALVE (所有视图)	648
12.1.55	VALVE: 标准视图	648
12.1.56	VALVE: 维护视图	650
12.2	面板: 资产管理	651
12.2.1	ASSETMON 面板 [资产] 的视图	651
12.3	全局视图和表示	654
12.3.1	对象总览	654
12.3.2	全局视图: 消息视图	656
12.3.3	全局视图: 批生产视图	658
12.3.4	全局视图: 趋势视图	659
12.3.5	显示不带资产管理的“停止无效”	661
12.3.6	消息视图 [资产]	662
12.3.7	维护视图 [资产]	663
12.3.8	标识视图 [资产]	665
12.3.9	资产面板的全局表示和视图	671

12.4	面板： 操作员控制块.....	676
12.4.1	OP_A: 标准视图.....	676
12.4.2	OP_A_LIM: 标准视图.....	676
12.4.3	OP_A_RJC: 标准视图.....	677
12.4.4	OP_D: 标准视图.....	677
12.4.5	OP_D3: 标准视图.....	678
12.4.6	OP_TRIG: 标准视图.....	679
13	块图标.....	681
13.1	块图标的常规属性.....	681
13.2	面板的位置.....	683
13.3	突出显示“报警循环”和“通过过程标签选择画面”的块图标.....	684
13.4	块图标： 设备块.....	685
13.4.1	块图标： CTRL_S.....	685
13.4.2	块图标： CTRL_PID.....	687
13.4.3	块图标： DIG_MON.....	688
13.4.4	块图标： DOSE.....	689
13.4.5	块图标： ELAP_CNT.....	691
13.4.6	块图标： FMCS_PID.....	692
13.4.7	块图标： FMT_PID.....	693
13.4.8	块图标： INTERLOK.....	694
13.4.9	块图标： MEAS_MON.....	694
13.4.10	块图标： MOT_REV.....	696
13.4.11	块图标： MOT_SPED.....	697
13.4.12	块图标： MOTOR.....	698
13.4.13	块图标： RATIO_P.....	699
13.4.14	块图标： SWIT_CNT.....	700
13.4.15	块图标： VAL_MOT.....	700
13.4.16	块图标： VALVE.....	701
13.5	块图标： 资产管理.....	702
13.5.1	块图标： 资产管理.....	702
13.6	块图标： 操作员控制块.....	705
13.6.1	块图标： OP_A.....	705
13.6.2	块图标： OP_A_LIM.....	705
13.6.3	块图标： OP_A_RJC.....	705
13.6.4	块图标： OP_D.....	706
13.6.5	块图标： OP_D3.....	706
13.6.6	块图标： OP_TRIG.....	707

14	附录	709
14.1	SM 模块的 MODE 设置	709
14.2	SM 模块的 OMODE 设置	719
14.3	技术数据, “标准库块”	720
14.4	为 FF 设备自定义 MODE_LW	725
14.5	在资产中集成 FF 设备	727
14.6	归档过程值	728
14.7	文本库 ASSETMON	729
14.8	质量代码和状态显示	730
14.8.1	质量代码显示	730
14.8.2	MS 维护状态	732
14.8.3	冗余组件 [资产] 的状态显示	735
15	PCS 7 高级过程控制模板	741
15.1	过程标签类型 (可插入模板)	741
15.1.1	过程标签类型说明	741
15.1.2	具有安全逻辑和控制回路监视的 PID 控制器	742
15.1.3	可直接访问执行器而无位置反馈的步控制器	744
15.1.4	分程控制	744
15.1.5	比率控制	746
15.1.6	级联控制	748
15.2	实例项目	750
15.2.1	包括噪声发生器的过程仿真	750
15.2.2	使用热流对温度进行串级控制	752
15.2.3	具有有色噪声仿真的控制回路监视	754
15.2.4	补偿可测量变量的动态前馈	755
15.2.5	针对非线性过程的面向运行点的参数调整 (增益调整)	758
15.2.6	超驰控制	760
15.2.7	使用史密斯预测器应对时滞的 PID 控制器	763
15.2.8	过滤控制回路中的噪声测量值	764
15.2.9	基于模型的预测控制	765
15.2.10	示例项目 APC_ExaSP	767
15.2.10.1	PCS 7 高级过程控制模板简介	767
15.2.10.2	标签类型	768
	索引	769

关于块描述的常规信息

块描述的设置始终是一致的，其中包含以下部分：

块描述的标题

实例： CTRL_PID： PID 控制器块

标题以块的类型名称开头（例如，“CTRL_PID”）。此符号名称在符号表中输入，并且在项目内必须唯一。

除了类型名称之外，您还会看见指示块的用途或功能的关键字（例如，“PID 控制器块”）。

对象名（类型 + 编号）

FB x

块类型的对象名称由实现的类型（功能块 = FB、功能 = FC）和块编号 x 组成。

显示块 I/O 的链接

实例：

- CTRL_PID 块 I/O

单击“块 I/O”链接，显示指定块的块 I/O 列表。

显示块图标和面板的链接

如果要块用于操作员监控，且存在块图标和面板，则单击这些链接可直接显示相应的画面和描述。

实例：

- CTRL_PID 块图标
- CTRL_PID 面板

功能

在此可找到关于块功能的简短描述。

有关复杂块的更多信息，可参考“工作原理”部分。

工作原理

例如，在此可找到关于特定输入、运行模式或时间顺序的功能的更详细信息。为了有效地使用块，必须熟悉这些关系。

调用 OB

在此可找到关于组织块 (OB) 的信息，其中必须安装了所述的块。如果使用 CFC，会自动将块安装到循环 OB（循环中断）以及在块任务列表列出的 OB（例如，重启 OB100）中。

编译期间，CFC 生成所需的 OB。如果没有使用 CFC 而直接调用块，则必须对这些 OB 进行编程并在块内调用它们的实例。

错误处理

ENO 布尔块输出用于指示 CFC 图表有错。

该值相当于 **BIE**（块完成之后，STEP 7 STL 中的二进制结果）或 **OK** 位（采用 SCL 表示法）并指示：

ENO = BIE = OK = 1 (TRUE) -> 块的结果无错误。

ENO = BIE = OK = 0 (FALSE) -> 结果或约束条件（例如，输入值和模式）无效。

FB 也返回 **BIE** 的取反（在实例 DB 的 **QERR** 输出）。

QERR = NOT ENO

错误消息通过两个单独的操作来生成：

- 操作系统检查处理错误（例如，值上溢，调用的系统功能返回错误 ID 和 **BIE = 0**）。这是系统功能，在块描述中没有特别提到。
- 块算法检查值和运行模式的功能有效性。这些错误事件会记录到块描述中。

您可以评估错误显示，例如，对于生成消息或对于使用替代值替换无效结果。有关消息的更多信息，可参考“消息块”部分。

启动特征

下面是各种不同的启动行为：

- 初始启动

第一次从块所插入的 OB 中调用块。通常是执行标准、过程特定的操作的 OB（例如，循环中断 OB）。

块采用符合输入参数的状态。可能是默认值（更多相关信息，可参考“I/O”部分），也可能是已组态的值，例如在 CFC 中组态的值。并不单独描述初始启动特征，除非块不符合此规则。

- 启动

块在 CPU 启动期间执行一次。在启动 OB（可以通过 ES 自动安装或通过 STEP 7 手动安装）中调用该块。在这种情况下，描述启动特征。

请注意，块输出有默认值，并且，如果首先处理这些块输出，它们会在 CPU 启动过程中与其它块一起生效。

组态工程师应负责保证块的启动行为正确。

时间响应

必须将分配了此功能的块安装在循环中断 OB 中。该块根据采样时间（连续两次循环操作之间的时间）来计算时间常数/参数。

在 ES 上进行 CFC 组态时，采样时间还要由运行组的分段决定，这将确保不会在每次 OB 运行时都执行块。

此采样时间输入在 I/O 的 SAMPLE_T 参数中。

用 CFC 组态时，只要将块插入 OB 和运行组中，就会自动执行此操作。因此，此输入设置为在 CFC 中不可见。

STEP 7 组态时手动设置时间响应。

只有为块分配了该功能，才会涉及时间响应。

消息响应

具备消息响应的块会将各种事件报告给更高级别的 OS。生成消息所需的现有参数会记录下来。

对于不具备消息响应的块，可通过附加的消息块进行扩展。关于消息响应的参考信息，可在各消息块的描述中找到。

I/O

块的 I/O 代表其数据接口。这些 I/O 可用于将数据传送到块或从块获取结果。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
U1	加数 1	REAL	0	I	+	>0
.....						

“I/O”表列出块类型的所有 I/O 参数。可使用工程工具访问这些列表。它们以字母顺序排列。未列出只能通过块算法访问的元素（内部变量）。

各列的含义如下：

- **I/O**

参数名使用英语命名，例如 PV_IN = **Process Variable IN**put（过程变量，控制变量）。

这里应用了 SIMATIC 命名约定。

在 CFC 中提供的块表示法如下：

I/O 名称用**粗体**字符表示 I/O 可见，用正常体字符表示 I/O 不可见。

- **含义**

功能（也可能是简短描述）

- **数据类型**

参数的 S7 数据类型（BOOL、REAL 等）

- **默认设置（默认值）**

块首次运行之前的参数值（除非已在组态中更改）

- **类型**

块算法对参数访问的类型；有输入、输出和逆向输入：

缩写	类型
I	输入。为块提供值（在 CFC 中的表示法：块左侧）
O	输出。输出值。（在 CFC 中的表示法：块右侧）
IO	运行 = 输入/输出。可由 OS 写入并且可由块写回（在 CFC 中的表示法：块左侧）的逆向输入

- **OCM**

标记“+”的参数可通过相应的面板调整和监视。

- **允许值**

值的数据类型范围内的附加限制

操作和监视

如果 **AS** 块存在对应的面板，则指向相应面板和块图标的描述的连接可用（主题上部的按钮也可用）。

参见

块图标: [CTRL_PID \(页 687\)](#)

[CTRL_PID 的 I/O \(页 79\)](#)

系列: COMM

2.1 REC_BO: 使用 BRCV 接收 128 个 BOOLEAN 值

2.1.1 REC_BO 的描述

对象名 (类型 + 编号)

FB 208

- REC_BO 块 I/O (页 26)

应用领域

REC_BO 块表示连接 SFB 13“BRCV”的简单用户接口。

它通过 MPI、PROFIBUS 或以太网连接从另一个 S7 CPU 接收 128 个 BOOL 值。该 CPU 需要调用 PCS 7 库的功能块类型“SEND_BO”(FB 207) 才能发送数据。在 STEP 7 中, 必须为通讯双方组态同种连接, 并将其传送到 AS。

仅当完成了作业且输出 **NDR** 处有 0 -> 1 信号跳变后, 数据才可用。

调用 OB

这是在其中安装该块的循环中断 OB (例如 OB 35)。

工作原理

内部 SFB 13“BRCV”允许在通讯伙伴之间交换 128 个 BOOL 值。数据由 CPU 的操作系统接收，并输入到接收 FB (REC_BO) 的背景数据块中。操作系统必须发送已接收数据的内部确认，才能接收新数据。

数据输入到数据块与用户程序执行不同步。调用“REC_BO”后，只要作业正在进行中 (NDR = 0)，就不会计算背景数据块数据。如果作业完成无误，则在一个周期的持续时间内，输出 NDR = 1。在下一个周期中，FB 自动向 CPU 操作系统发出接收使能信号（自该调用起，NDR 复位到 0）。

接收使能信号比第一个进入接收作业更早生效。在此情况下，由操作系统存储接收使能信号。

ID 参数表示在连接组态中指定的连接号，并只在冷启动后第一次调用时应用。

R_ID 参数是一个随机数（建议：消息帧 ID）；但是，它在相应的发送和接收块中必须相同，并且只在冷启动后第一次调用时应用。

必须在每个程序周期内为各个 ID/R_ID 对调用“REC_BO”块（周期性调用，也可通过超时报警调用）。每个消息帧需要两次调用“REC_BO”。

ERR（错误）和 STAT（状态）输出指示有关 SFB 13 的具体错误信息（请参见“错误处理”）。

如果发生错误，可将替代值作为接收的数据输出。

错误处理

REC_BO 块的错误处理只限于下级 SFB 13“BRCV”的错误信息。有关介绍 ERR 和 STAT 输出的更多信息，请参考手册《S7-300/400 系统软件 - 系统和标准功能》(System Software for S7-300/400 - System and Standard Functions)。

如果输入 SUBS_ON = TRUE，则在发生接收错误或未收到新数据时，替代值将被输出到 REC_MON（周期数）。在 REC_MON 个周期期间，会将最后的有效值应用到输出。

如果 REC_MON 设置为 FALSE，则在发生错误时总是会将最后的有效值应用到输出。

启动特征

不可用。

时间响应

不可用。

消息响应

不可用。

操作和监视

不可用。

2.1 REC_BO: 使用 BRCV 接收 128 个 BOOLEAN 值

2.1.2 REC_BO 的 I/O

CFC 中块显示的工厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示 I/O 可见; 正常字体的 I/O 名称则表示 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息”。

I/O (参数)	含义	数据类型	默认值	类型
ERR	1 = 错误 (有关错误类型, 请参见 STAT)	BOOL	0	O
ID	连接 ID	WORD	0	I
NDR	1 = 接收到新数据	BOOL	0	O
QNO_REC	1 = 未接收到数据	BOOL	0	I
QSUBS_ON	1 = 替代值	BOOL	0	I
R_ID	消息帧 ID	DWORD	0	I
RD_BO_00	接收到值 00	BOOL	0	O
...	...			
RD_BO_15	接收到值 15	BOOL	0	O
RD_BO_16	接收到值 16	BOOL	0	O
...	...			
RD_BO_127	接收到值 127	BOOL	0	O
REC_MON	接收监视 (值 = 块调用的数量)	INT	3	I
STAT	错误 ID	WORD	0	O
SUBBO_00	替代值 00	BOOL	0	I
...	...			
SUBBO_15	替代值 15	BOOL	0	I
SUBBO_16	替代值 16	BOOL	0	I
...				
SUBBO127	替代值 127	BOOL	0	I
SUBS_ON	1 = 出错时使用替代值	BOOL	0	I

2.2 REC_R: 使用 BRCV 接收 32 个 BOOLEAN 值和 32 个 REAL 值

2.2.1 REC_R 的描述

对象名 (类型 + 编号)

FB 210

- REC_R 块 I/O (页 29)

应用领域

REC_R 块表示连接 SFB 13“BRCV”的简单用户接口。

它通过 MPI、PROFIBUS 或以太网连接从另一个 S7 CPU 接收 32 个 BOOL 值和 32 个 REAL 值。该 CPU 需要调用 PCS 7 库的功能块类型“SEND_R”(FB 209) 才能发送数据。在 STEP 7 中, 必须为通讯双方组态同种连接, 并将其传送到 AS。

仅当完成了作业且输出 **NDR** 处有 0 -> 1 信号跳变后, 数据才可用。

调用 OB

这是在其中安装该块的循环中断 OB (例如 OB 35)。

工作原理

内部 SFB 13“BRCV”允许在通讯伙伴之间交换 32 个 BOOL 值和 32 个 REAL 值。数据由 CPU 的操作系统接收, 并输入到接收 FB (REC_R) 的背景数据块中。操作系统必须发送已接收数据的内部确认, 才能接收新数据。

数据输入到背景数据块与用户程序执行不同步。调用“REC_R”后, 只要作业正在进行中 (**NDR = 0**), 就不会计算背景数据块数据。如果作业完成无误, **NDR** 输出被设置为 1, 并持续一个周期的时间。在下一个周期中, FB 自动向 CPU 操作系统发出接收使能信号 (自该调用起, **NDR** 复位到 0)。

接收使能信号比第一个进入接收作业更早生效。在此情况下, 由操作系统存储接收使能信号。

2.2 REC_R: 使用 BRCV 接收 32 个 BOOLEAN 值和 32 个 REAL 值

ID 参数表示在连接组态中指定的连接号，并只在冷启动后第一次调用时应用。

R_ID 参数是一个随机数（建议：消息帧 ID）；但是，它在相应的发送和接收块中必须相同，并且只在冷启动后第一次调用时应用。

必须在每个程序周期内为各个 **ID/R_ID** 对调用“REC_R”块（周期性调用，也可通过超时报警调用）。每个消息帧需要调用“REC_R”两次。

ERR（错误）和 **STAT**（状态）输出指示有关 **SFB 13** 的具体错误信息（请参见“错误处理”）。

如果发生错误，可将替代值作为接收的数据输出。

错误处理

REC_R 块的错误处理只限于下级 **SFB 13**“BRCV”的错误信息。有关介绍 **ERR** 和 **STAT** 输出的更多信息，请参考手册《S7-300/400 系统软件 - 系统和标准功能》(*System Software for S7-300/400 - System and Standard Functions*)。

如果输入 **SUBS_ON = TRUE**，则在发生接收错误或未收到新数据时，替代值将被输出到 **REC_MON**（周期数）。在 **REC_MON** 个周期期间，会将最后的有效值应用到输出。

如果 **REC_MON** 设置为 **FALSE**，则在发生错误时总是会将最后的有效值应用到输出。

启动特征

不可用。

时间响应

不可用。

消息响应

不可用。

操作和监视

不可用。

2.2.2 REC_R 的 I/O

CFC 中块显示的工厂设置在“**I/O**”列中标识:

加粗字体的 I/O 名称表示 I/O 可见; 正常字体的 I/O 名称则表示 I/O 不可见。

有关所用缩写的说明和信息, 可参考“关于块描述的常规信息”。

I/O (参数)	含义	数据类型	默认值	类型
ERR	1 = 错误 (有关错误类型, 请参见 STAT)	BOOL	0	O
ID	连接 ID	WORD	0	I
NDR	1 = 接收到新数据	BOOL	0	O
QNO_REC	1 = 未接收到数据	BOOL	0	I
QSUBS_ON	1 = 替代值	BOOL	0	I
R_ID	消息帧 ID	DWORD	0	I
RD_BO_00	接收值 BOOL 00	BOOL	0	O
...				
RD_BO_07	接收值 BOOL 07	BOOL	0	O
RD_BO_08	接收值 BOOL 08	BOOL	0	O
...				
RD_BO_31	接收值 BOOL 31	BOOL	0	O
RD_R_00	接收值 REAL 00	REAL	0	O
...				
RD_R_07	接收值 REAL 07	REAL	0	O
RD_R_08	接收值 REAL 08	REAL	0	O
...				
RD_R_31	接收值 REAL 31	REAL	0	O
REC_MON	接收监视 (值 = 块调用的数量)	INT	3	I
STAT	错误 ID	WORD	0	O
SUBBO_00	替代值 BOOL 00	BOOL	0	I
...				
SUBBO_07	替代值 BOOL 07	BOOL	0	I

2.2 REC_R: 使用 BRCV 接收 32 个 BOOLEAN 值和 32 个 REAL 值

I/O (参数)	含义	数据类型	默认值	类型
SUBBO_08	替代值 BOOL 08	BOOL	0	I
...				
SUBBO_31	替代值 BOOL 31	BOOL	0	I
SUBR_00	替代值 REAL 00	REAL	0	I
...				
SUBR_07	替代值 REAL 07	REAL	0	I
SUBR_08	替代值 REAL 08	REAL	0	I
...				
SUBR_31	替代值 REAL 31	REAL	0	I
SUBS_ON	1 = 出错时使用替代值	BOOL	0	I

参见

关于块描述的常规信息 (页 17)

2.3 SEND_BO: 使用 BSEND 发送 128 个 BOOLEAN 值

2.3.1 SEND_BO 的描述

对象名 (类型 + 编号)

FB 207

- SEND_BO 块 I/O (页 34)

应用领域

SEND_BO 块表示连接 SFB 12“BSEND”的简单用户接口。

它通过 MPI、PROFIBUS 或工业以太网连接将多达 128 个 BOOL 值发送到另一个 S7 CPU。该 CPU 需要调用 PCS 7 库的功能块类型“REC_BO”(FB 208) 才能接收数据。

仅当作业完成后 (即收到确认 DONE = TRUE 后), 方可在“REC_BO”中得到一致的数据。该确认由输出 CIW 的信号变为 0 来指示。

若参数 FAST = 1, 由于 FB 在内部调用两次 SFB 12“BSEND” (需要使用其控制输入 REQ 的上升沿才能启用 SFB 12), 所以 FB 可通过每次 FB 调用传送一个消息帧。不过, 只有在这两次 FB 调用之间有充足的时间用于传送消息帧时, 才能启动如此频繁的发送请求。

调用 OB

这是在其中安装该块的循环中断 OB (例如 OB 35)。

工作原理

内部 SFB 12“BSEND”允许在通讯伙伴之间交换 128 个 BOOL 值。数据由 CPU 的操作系统发送到通讯伙伴，并自动输入到接收 FB (REC_BO) 的背景数据块中。在发送任何新的 128 个 BOOL 值之前，系统等待操作系统传送一个针对刚刚发送的值的内部确认。

调用块时，如果控制输入 **COM** = 1，则启动数据传送。要传送所有数据，必须至少调用一次该块（与周期无关）。该块将作业传送给 CPU 操作系统，以完成作业处理。数据传送期间，允许多次调用具有相同 **ID** 和 **R_ID** 的块；但这些调用不起任何作用，即该块在每个周期中只能被调用一次。而是在 **STAT** 处输出值 11。从用户存储器读取数据与用户程序执行不同步。如果作业完成无误，则会将 **CIW** 设置为 0。（如出现错误，则 **ERR** = 1。新作业将使用实际数据自动启动，直至所有数据都传送完毕。）如果 **COM** 输入 = 0，将中止当前数据传送并且不会再继续；**CIW** 相应地被设置为 0。

ID 参数表示在连接组态中指定的连接号，并只在冷启动后第一次调用时应用。

R_ID 参数是一个随机数（建议：消息帧 ID）；但是，它在相应的发送和接收块中必须相同，并且只在冷启动后第一次调用时应用。

如果参数 **FAST** = 1，则可通过每次 FB 调用传送新的消息帧。这种情况下，FB 在内部调用两次 SFB 12“BSEND”（需要使用其控制输入 **REQ** 的上升沿才能启用 SFB 12）。在每个周期中调用 FB 可实现每个周期传送一个消息帧。但是，建议仅在以下情况下如此频繁的发送请求：

- 在两次 FB 调用之间有足够的用来传送消息帧。
- 接收 CPU 中“REC_BO”的调用速度比发送 CPU 中的“SEND_BO”的调用速度快（对于每个消息帧，接收 CPU 需要调用两次“REC_BO”（相当于 2 个周期））。

如果参数 **FAST** = 0，则只有每当第二次调用 FB 时才能启动新的发送作业。

错误处理

块的错误处理只限于下级 SFB 12“BSEND”的错误信息。有关介绍 **ERR** 和 **STAT** 输出的更多信息，请参考手册《S7-300/400 系统软件 - 系统和标准功能》(System Software for S7-300/400 - System and Standard Functions)。

如出现错误，新的作业将使用实际数据自动启动，直至所有数据都传送完毕。

启动特征

不可用。

时间响应

不可用。

消息响应

不可用。

操作和监视

不可用。

2.3.2 SEND_BO 的 I/O

CFC 中块显示的工厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示 I/O 可见; 正常字体的 I/O 名称则表示 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息”。

I/O (参数)	含义	数据类型	默认值	类型
BO_00	输入 00	BOOL	0	I
...	...			
BO_15	输入 15	BOOL	0	I
BO_16	输入 16	BOOL	0	I
...	
BO_127	输入 127	BOOL	0	I
CIW	1 = 作业正在进行中	BOOL	0	O
COM	1 = 连续发送, 0 = 不发送	BOOL	1	I
DONE	作业完成	BOOL	0	O
ERR	错误 (有关错误类型, 请参见 STAT)	BOOL	0	O
FAST	传输类型: 1 = 每次调用 FB 时可传送一个消息帧 0 = 调用两次 FB 时可传送一个消息帧	BOOL	0	I
ID	连接 ID	WORD	0	I
R_ID	消息帧 ID	DWORD	0	I
STAT	错误 ID	WORD	0	O

参见

关于块描述的常规信息 (页 17)

2.4 SEND_R: 使用 BSEND 发送 32 个 BOOL 值和 32 个 REAL 值 (受变化驱使)

2.4.1 SEND_R 的描述

对象名 (类型 + 编号)

FB 209

- SEND_R 块 I/O (页 38)

应用领域

SEND_R 块表示连接 SFB 12“BSEND”的简单用户接口。

它由变化驱动, 并通过 MPI、PROFIBUS 或工业以太网连接将多达 32 个 BOOL 值和 32 个 REAL 值发送到另一个 S7 CPU。该 CPU 需要调用 PCS 7 库的功能块类型“REC_R”(FB 210) 才能接收数据。

仅当作业完成后 (即收到确认 `DONE = TRUE` 后), 方可在“REC_R”中得到一致的数据。该确认由输出 **CIW** 的信号变为 0 来指示。

若参数 **FAST = 1**, 由于 FB 在内部调用两次 SFB 12“BSEND” (需要使用其控制输入 **REQ** 的上升沿才能启用 SFB 12), 所以 FB 可通过每次 FB 调用传送一个消息帧。不过, 只有在这两次 FB 调用之间有充足的时间用于传送消息帧时, 才能启动如此频繁的发送请求。

调用 OB

这是在其中安装该块的循环中断 OB (例如 OB 35)。

工作原理

该块监视 32 个 BOOL 和 32 个 REAL 输入的数据, 以检测先前被成功发送的数据的变化。在每个变化监视周期中, 都会为 REAL 值 (R_xx) 设置滞后量 HYS_R_xx (绝对值)。HYS_R_xx 的默认值为零 (无滞后量)。通过 EDC_MIN 和 EDC_MAX 参数可锁定或强制数据传送。

在 EDC_MIN 中, 可设置等待实际输入数据的下一个发送请求的周期数, 与数据中的任何变化无关。

在 EDC_MAX 中, 可设置上一次传送有效数据到下一次传送实际输入数据之间的等待周期数, 与数据中的任何变化或 REAL 值的变化是否在设定的滞后量范围内无关。

由于“SEND_R”和“REC_R”之间数据传输异步 (参见下文), 所以基于设定周期数的理论时间无法保证。

内部 SFB 12“BSEND”允许在通讯伙伴之间交换 32 个 BOOL 值和 32 个 REAL 值。数据由 CPU 的操作系统发送到通讯伙伴, 并自动输入到接收 FB (REC_R) 的背景数据块中。在发送新值前, 系统等待操作系统传送一个针对刚刚发送的值的内部确认。

调用块时, 如果控制输入 **COM = 1**, 则启动数据传送。要传送所有数据, 必须至少调用一次该块 (与周期无关)。该块将作业传送给 CPU 操作系统, 以完成作业处理。数据传送期间, 允许多次调用具有相同 ID 和 R_ID 的块; 但这些调用不起任何作用, 即该块在每个周期中只能调用一次。而是在 **STAT** 处输出值 11。从用户存储器读取数据与用户程序执行不同步。如果作业完成无误, 则会将 **CIW** 设置为 0。 (如出现错误, 则 **ERR = 1**。新作业将使用实际数据自动启动, 直至所有数据都传送完毕。) 如果 **COM** 输入 = 0, 将中止所有不完整的数据传送并且不会再继续; **CIW** 相应地被设置为 0。

ID 参数表示在连接组态中指定的连接号, 并只在冷启动后第一次调用时应用。

R_ID 参数是一个随机数 (建议: 消息帧 ID); 但是, 它在相应的发送和接收块中必须相同, 并且只在冷启动后第一次调用时应用。

如果参数 **FAST = 1**, 则可通过每次 FB 调用传送新的消息帧。这种情况下, FB 在内部调用两次 SFB 12“BSEND” (需要使用其控制输入 **REQ** 的上升沿才能启用 SFB 12)。在每个周期中调用 FB 可实现每个周期传送一个消息帧。

但是, 建议仅在以下情况下如此频繁的发送请求:

- 在两次 FB 调用之间有足够的用来传送消息帧。
- 接收 CPU 中“REC_R”的调用速度比发送 CPU 中“SEND_R”的调用速度快 (对于每个消息帧, 接收 CPU 需要调用两次“REC_R” (相当于 2 个周期))。

如果参数 **FAST = 0**, 则只有每当第二次调用 FB 时才能启动新的发送作业。

2.4 SEND_R: 使用 BSEND 发送 32 个 BOOL 值和 32 个 REAL 值 (受变化驱使)**错误处理**

块的错误处理只限于下级 SFB 12“BSEND”的错误信息。有关介绍 **ERR** 和 **STAT** 输出的更多信息, 请参考手册《S7-300/400 系统软件 - 系统和标准功能》(*System Software for S7-300/400 - System and Standard Functions*)。

如出现错误, 新的作业将使用实际数据自动启动, 直至所有数据都传送完毕。

启动特征

不可用。

时间响应

不可用。

消息响应

不可用。

操作和监视

不可用。

2.4 SEND_R: 使用 BSEND 发送 32 个 BOOL 值和 32 个 REAL 值 (受变化驱使)

2.4.2 SEND_R 的 I/O

CFC 中块显示的工厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示 I/O 可见; 正常字体的 I/O 名称则表示 I/O 不可见。

有关所用缩写的说明和信息, 可参考“关于块描述的常规信息”。

I/O (参数)	含义	数据类型	默认值	类型
BO_00	BOOL 输入 00	BOOL	0	I
...	...			
BO_07	BOOL 输入 07	BOOL	0	I
BO_08	BOOL 输入 08	BOOL	0	I
...	...			
BO_31	BOOL 输入 31	BOOL	0	I
CIW	1 = 作业正在进行中	BOOL	0	O
COM	1 = 连续发送, 0 = 不发送	BOOL	1	I
DONE	1 = 作业已完成	BOOL	0	O
EDC_MAX	在 n 个周期后, 不作任何更改强制传送	INT	10	I
EDC_MIN	若进行了更改, 则在 n 个周期后开始最早传送	INT	1	I
ERR	错误 (有关错误类型, 请参见 STAT)	BOOL	0	O
FAST	传输类型: 1 = 每次调用 FB 时可传送一个消息帧 0 = 调用两次 FB 时可传送一个消息帧	BOOL	0	I
HYS_R_00	滞后量输入 00	REAL	0	I
...	...			
HYS_R_07	滞后量输入 07	REAL	0	I
HYS_R_08	滞后量输入 08	REAL	0	I
...	...			
HYS_R_31	滞后量输入 31	REAL	0	I
ID	连接 ID	WORD	0	I
R_00	REAL 输入 00	REAL	0	I
...	...			

2.4 SEND_R: 使用 BSEND 发送 32 个 BOOL 值和 32 个 REAL 值 (受变化驱使)

I/O (参数)	含义	数据类型	默认值	类型
R_07	REAL 输入 07	REAL	0	I
R_08	REAL 输入 08	REAL	0	I
...	...			
R_31	REAL 输入 31	REAL	0	I
R_ID	消息帧 ID	DWORD	0	I
STAT	错误 ID	WORD	0	O

参见

关于块描述的常规信息 (页 17)

系列: *COMM*

2.4 *SEND_R*: 使用 *BSEND* 发送 32 个 *BOOL* 值和 32 个 *REAL* 值 (受变化驱使)

系列： CONTROL

3.1 使用 PID 调谐优化控制器

使用 PID 调谐器可调谐以下控制块：

- CTRL_PID
- CTRL_S
- FMCS_PID
- FMT_PID

工作原理

使用“控制器调谐”功能，可确定特定过程的最优控制器设置。向导将引导您完成各个步骤。

受控系统的测量数据会记录下来。最优 PID 参数会基于此数据计算，以备将来使用。

有关使用 PID 调谐器的更多相关信息，可参考相应的“帮助”文件。在 CFC 中的图表上选择一个块，然后激活菜单命令“**编辑(Edit)** > “**控制器优化(Controller Optimization)**”，可调用此帮助。

3.2 CPM: 监视控制回路性能

3.2.1 CPM 的描述

对象名 (类型 + 编号)

FB 140

- CPM 块 I/O (页 61)

CPM 的应用领域

该块用于连续监视控制回路的控制性能。

CPM 块可计算:

- 过程在稳定状态下控制性能的随机特征
 - 控制变量的平均值、方差和标准差
 - 调节变量的平均值和控制偏差
 - 控制性能指数
 - 估计的稳定状态过程增益
- 设定值阶跃变化的控制性能确定特征
 - 响应时间和稳定时间及稳定比
 - 有关阶跃高度的绝对超调量和相对于超调量。

在更长的可自由选择时间段内, 控制回路中信号的其他统计和图形评估信息可在 CPM 块的面板中显示。

在工厂或单元的总览画面中, 可以获得基于 CPM 块图标 (指示灯功能) 的所有控制回路的清晰状态画面。

目的是在错误发生前检测到这些错误, 并将用户的注意力集中到工厂中不再正常运行的控制回路上。

工作原理

CPM 块在滑动时间窗中评估设定值和过程值信号以及 PID 控制器的调节变量。也要考虑控制器的模式。对于稳定状态下的过程，将检测到的随机特征与调试过程中获得的参考值进行比较。如果存在设定值阶跃变化，则根据定义随机特征是不相关的且是暂时冻结的。而自动激活确定特征的监视。

如果控制质量下降到定义的限制之下，则会生成一条消息。当设定值中存在阶跃变化时，如果超出定义的超调量限制，也会生成一条消息。

组态

每个 PID 控制器都具有一个分配给它的 CPM 块，该块安装在同一 CFC 图表中并与控制器互连。这对于相应的过程标签类型已经是如此。CPM 块将根据命名约定链接到分配的控制块。CPM 实例的名称与控制器实例的名称完全相同，只是后面添加了 "_cpm"。

实例

TIC501_cpm 监视块从属于控制器 TIC501。

成功调试和优化要监视的 PID 控制器后，在过程处于稳定状态时，CPM 块将初始化，然后将相应的特征值存储为参考值。按照下面列出的步骤进行操作：

- 将要监视的 PID 控制器切换到自动模式，并将设定值设置为典型运行点。该运行状态旨在表示过程的正常运行；换句话说，整个工厂/单元应在生产条件下运行。在工程系统的 CFC 趋势视图中或者 OS 上的 WinCC 在线趋势控件中监视过程，并一直等到过程稳定为止。
- 要指定滑动时间窗的长度 TimeWindow，监视趋势中 CPM 块的块输出 PV_Variance。时间窗应足够长以保持相关小数位的方差十分恒定。如果选定的时间窗相对于控制回路中的时间常数和干扰信号谱来说过短，则方差将带有过多噪音和无用信息。如果选定的时间窗过长，则 CPM 块检测到控制质量退化所需的时间将较长。设定值阶跃变化后，恢复随机特征监视所需的时间也更长。TimeWindow 参数的起始值最好是最长过程时间常数的 10 倍或 PID 控制器复位时间的 20 倍。
- CPM 块可以进行初始化，条件是控制器
 - 已经过优化
 - 已达到稳定状态
 - 已指定了时间窗且
 - 以稳定状态的值进行了填充。

通过 CPM 面板参数视图中的“初始化”(Initialization) 按钮，或者当该块的参数 InitRefVar = true 时，执行初始化。这样会将当前时间窗中的 PV_Variance 参数存储，作为计算块中控制质量的参考值以及调节变量和过程值的参考值。

现在，控制性能指数 CPI 应约为 100%，表明控制回路运行正常。由于随机波动，CPI 也可能暂时超过 100% 标记。但是，如果 CPI 长时间内大幅下降，表明控制性能恶化。

有关解释块计算结果的详细信息，请参考 功能 (页 48) 部分。

注:

- 如果运行期间更改了时间窗的长度, CPI 将暂时明显偏离其旧值, 然后逐渐回到新的稳定值。建议在 CPI 值稳定到恒定级别后重新初始化 CPM 块。
- 如果认为噪声严重影响了计算出的 CPI 信号, 则可使用带有滤波器时间常数 `TimeWindowCPI_FiltFactor` 的集成低通滤波器 (参数 `CPI_FiltFactor`) 对其进行平滑处理。

CPM 面板可通过分配的 PID 控制器的面板打开, 这样就不必将 CPM 块图标分别安装到每个 OS 画面中。实际上, 建议在适当的层级对总览画面中某个工厂或单元的所有 CPM 块图标进行分组。

可以通过更长时间内所有控制回路的控制质量的趋势显示来扩展此总览画面, 以便可以识别逐步恶化情况 (例如反映磨损和断裂)。还可以将消息归档 (`WinCC AlarmLogging` 控制) 的更深一层视图显示为按发生频率排序的统计列表。在该列表中, 引起最多报警的控制回路将显示在顶部。还可在实例项目 (页 750) 中进行查看。

启动特征

启动 CPU 时, 该块将重新初始化, 但保留存储的参考值。消息将在达到 `RunUpCyc` 上设置的启动周期数后受到抑制。这意味着必须在启动 OB 中调用该块。在 CFC 工程组态中, 这由 CFC 处理。

时间响应

该块必须在循环中断 OB 中调用。块的采样时间由 CFC 在编译期间于 `SampleTime` 参数中设置。

状态字分配

位号:	3	2	1	0
参数:	-	-	BatchEn	Occupied

位号:	7	6	5	4
参数:	-	-	CPI_Suppress	PID_AutAct

位号:	11	10	9	8
参数:	MV_known	-	-	-

位号:	15	14	13	12
参数:	-	-	-	-

位号:	19	18	17	16
参数:	CPI_AlmAct	CPI_WrnAct	OvsWrnAct	OvsAlmAct

位号:	23	22	21	20
参数:	AlmOffDelayAct	AlmOnDelayAct	-	-

位号:	27	26	25	24
参数:	UserStatus bit 4	UserStatus bit 3	UserStatus bit 2	UserStatus bit 1

位号:	31	30	29	28
参数:	UserStatus bit 8	UserStatus bit 7	UserStatus bit 6	UserStatus bit 5

参见

CPM 报告 (页 60)

CPM 故障处理 (页 59)

CPM 模式 (页 48)

3.2.2 CPM 模式

CPM 模式

该块自身不提供任何模式更改。该块采用上一级别 PID 控制器的模式。
模式“手动/自动”在 CPM 块的块图标中显示 (A/M)。

参见

CPM I/O (页 61)

CPM 报告 (页 60)

CPM 故障处理 (页 59)

CPM 的功能 (页 48)

CPM 的描述 (页 42)

3.2.3 CPM 的功能

CPM 的功能

该块提供以下功能:

- 监视控制质量的随机特征
- 监视控制质量的确定特征
- 图形评估和长期统计
- 用于确定参考方差的供选方案
- 特殊情况串级控制和多变量控制
- 计算质量代码
- 报警延迟

监视控制质量的随机特征

遍历随机过程变量的平均值可通过长度为 $n = \text{TimeWindow} / \text{SampleTime}$ 的滑动时间窗确定, 例如, 对于控制变量 $y = \text{PV}$:

$$\bar{y} = \frac{1}{n} \sum_{i=1}^n y(i)$$

此计算的递归公式包含在 CPM 块调用的功能块 MEANTM_P 中。大多数稳态时间序列可视为可遍历的, 这样通过在有限长的窗口上求平均值便可以估算期望值。

平均控制偏差为 $\text{ER_Mean} = \text{SP} - \text{PV_Mean}$ 。如果控制器具有 I 动作, 则恒定设定值的平均稳态控制偏差不为零时, 表明控制回路中存在问题。然后应检查以下可能原因:

- 执行器干预受限且不足; 即执行器在其限制处停止。这可能是由于执行器尺寸不合适引起的, 或者可能只是由磨损和断裂造成的。
- 控制器需要的调节变量在过程中并未生效, 例如, 因为执行器有缺陷。

如果假定只有零平均值的干扰会产生影响, 则当稳态参考运行点 (MV_Ref , PV_Ref) 已知时, 这可用于估算当前平均过程增益。

$$\text{StatGain} = \frac{\text{PV}_{\text{Mean}} - \text{PV}_{\text{Ref}}}{\text{MV}_{\text{Mean}} - \text{MV}_{\text{Ref}}}$$

通常, 在初始化 CPM 块过程中获得参考运行点。但是, 之后不能在此运行点再次准确评估该过程。或者, 您还可以在相应块输入处手动输入参考值 PV_Ref 和 MV_Ref 。典型稳态运行点通常事先都是已知的, 例如

- 流量控制: 对于 $\text{MV} = 0$, $\text{PV} = 0$, 即阀已关闭。
- 温度控制: 对于 $\text{MV} = 0$, $\text{PV} = \text{PV}_{\text{amb}}$, 即环境温度。

如果过程增益随时间逐渐退化, 表明过程中存在磨损和断裂, 如热交换器堵塞、阀或闸上有沉淀物、单元效率降低等等。例如, 如果温度控制回路通过热交换器连接, 而在热交换器表面形成沉淀物, 则热传递系数将降低, 随之系统增益也会降低。在某些限制范围内, 这可通过闭环控制回路补偿 (以便控制器最初“掩盖”该问题)。虽然污染增加时, 可通过适当增大控制器增益将原始控制回路动力学恢复 (到某种程度), 但建议您消除问题的起因, 即清洁热交换器。

如果估算的过程增益突然暂时改变, 这很可能由于外部干扰。过程运行中出现此情况是正常的。但是, 如果这种现象更频繁地出现, 则值得找出原因

由于采用了此方法，方差 PV_Variance 或二阶中心矩需要计算每个当前测定值与（常数）平均值的差异：

$$\sigma_y^2 = \frac{1}{n-1} \sum_{i=1}^n (y(i) - \bar{y})^2 = \frac{1}{n} \left(\sum_{i=1}^n y^2(i) \right) - \bar{y}^2$$

但是，在该功能块内，使用了可节省计算时间的计算方案。标准差

$$PV_StdDev = \sigma_y = \sqrt{\sigma_y^2}$$

作为方差的平方根很容易解释，因为它具有与测定值相同的物理单位。

控制性能指数 CPI（以 [%] 为单位）描述了相对于基准的控制质量的当前方差。其定义如下

$$\zeta = \frac{\sigma_{ref}^2}{\sigma_y^2} 100\%$$

CPI 的范围为 $0 < \zeta \leq 100\%$ 。如果当前方差对应于基准值，则指数将达到值 100；反之，如果当前方差增大，控制性能指数将相应降低。理想情况下，参考方差在控制回路处于定义的良好状态时获得，并在初始化 CPM 块时存储。CPI 是否暂时达到了高于 100% 的值无关紧要。CPI > 100% 只表示当前控制变量的方差比参考状态下的稍微低。其它用于确定参考方差的供选方案将在单独部分中介绍。

这些随机特征的缺点是它们假定遍历性，即至少要在统计意义上于过程中达到稳定状态。从控制器的立场来说，每次阶跃变化都基本违反了这一要求，并暂时导致错误记录随机特征，例如方差增加过大。在 CPM 块中实现的组合方法的基本原理是使用控制性能的随机特征和确定特征，并根据运行状态自动选择合适的特征。

如果在控制回路中检测到设定值阶跃变化，CPM 块将冻结 CPI 值并自动抑制与该值相关的所有消息。作为用户，您也可以通过二进制输入 ManSupprCPI = 1 强行手动抑制消息。这对已知干扰出现时避免错误报警很有用，例如，连续过程中负载变化或批生产过程中定量给料时。对于这类事件，控制变量的方差暂时升高是很自然的，这不应解释为控制性能退化。

监视控制质量的确定特征

根据对设定值阶跃变化的响应评估控制性能相对简单，甚至通过肉眼便可实现。从自动监视意义上讲，CPM 块能够通过信号变化确定控制性能的重要特征，以便必要时系统可自动生成消息或报警。

要查找的第一件事始终是超调（如果存在并能与噪声级清楚地区分开）。对于正阶跃响应，

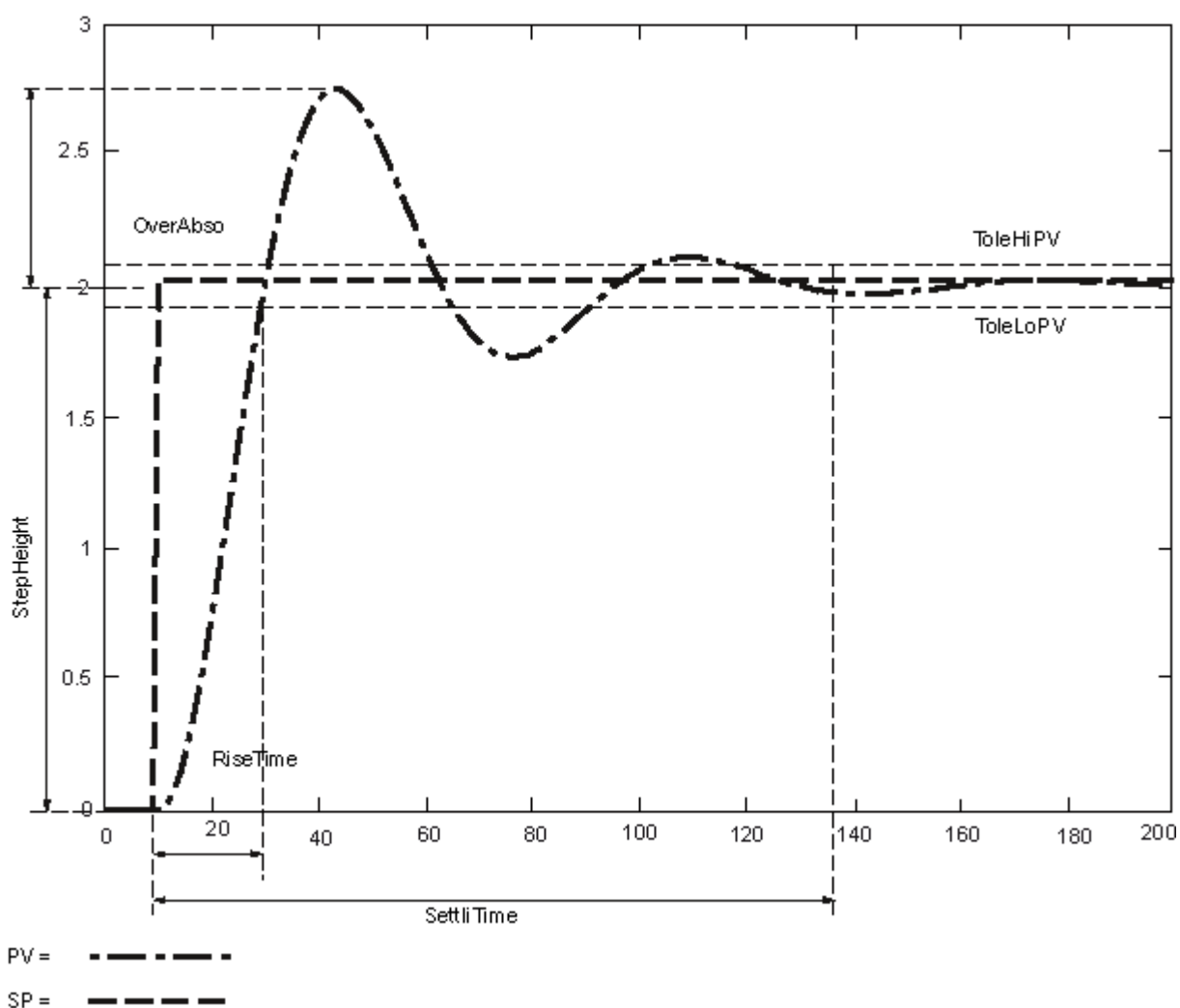
$$\text{OverAbso} = \max(\text{PV}) - \text{SP} > 0$$

是针对负阶跃响应（阶跃响应向下）的输出，而负值

$$\text{OverAbso} = \min(\text{PV}) - \text{SP} < 0$$

3.2 CPM: 监视控制回路性能

也是输出。标准化时，会参考绝对超调量；由于其与设定值阶跃变化的高度相关，所以始终为正值。百分比形式的相对超调量用于度量控制回路的阻尼。如果相对超调量大于 20% 或 30%，通常是环路增益（控制器的增益乘以控制系统的增益）过高，这是因为控制器一开始未很好地设置，或者控制系统的属性经过一段时间已发生了变化。如果超调量明显过高，则控制回路将在设备中生成弱阻尼振动。如果相对超调量超过指定的限制值，该块将发送一条说明该影响的消息。



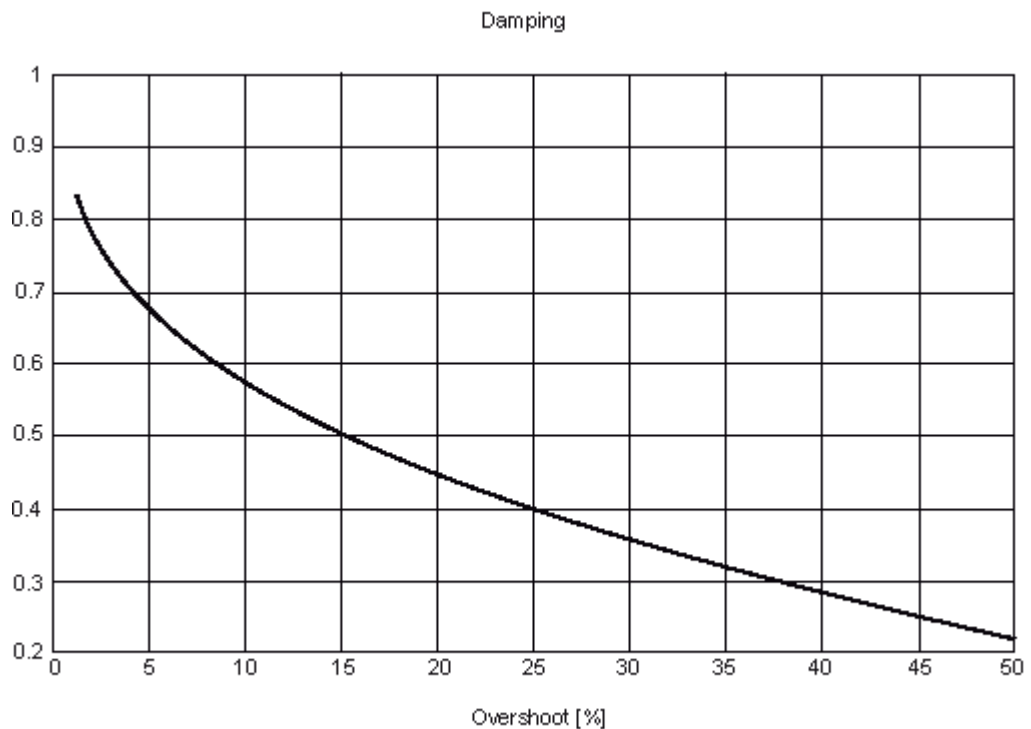
在每个控制回路中，超调量和相位余量之间存在常规相关性：超调量越高，相位余量越低。如果闭环控制回路的响应可由二阶传递函数近似描述

$$g_{cl}(s) = \frac{PV(s)}{SP(s)} = \frac{1}{\frac{1}{\omega_0^2} s^2 + 2 \frac{\delta}{\omega_0} s + 1}$$

以下关系已知：

1. 如果 $\delta \geq 1$ ，则超调量等于零，稳定响应是渐近的。
2. 如果 $\delta < 1$ ，则出现超调和振动。闭环的阻尼可通过超调量近似确定：

$$\delta = \frac{-\ln\left(\frac{\text{Overshoot}}{100\%}\right)}{\sqrt{\ln^2\left(\frac{\text{Overshoot}}{100\%}\right) + \pi^2}}$$



有用控制器设置通常为实现 5% 和 25% 之间的超调量，即 0.7 和 0.4 之间的阻尼。

3.2 CPM: 监视控制回路性能

如果超调量过高，通常有助于降低控制器的增益。

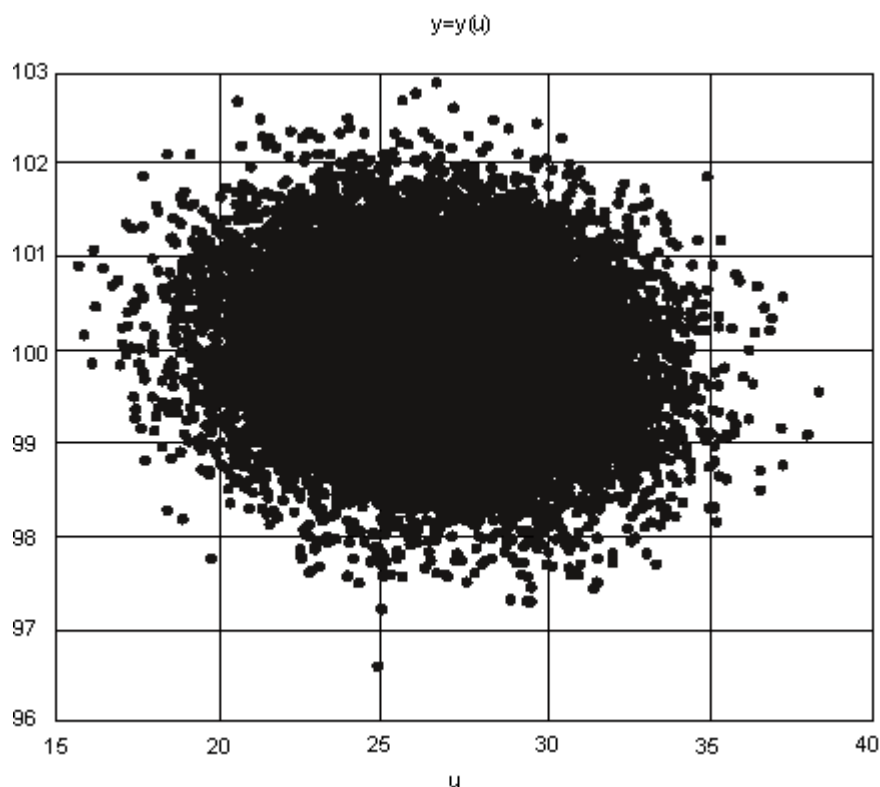
虽然超调量主要用于检查控制器增益，但还有另一个特征，可提供有关 I 动作的设置信息：如果复位时间的设置不合适，则设定值阶跃变化后过程值将向新的设定值缓慢移动。为允许标准化，稳定时间 **SettleTime** 与控制回路的阶跃响应的上升时间 **RiseTime** 相关。如果稳定比（即上升时间和稳定时间之商）小于约 25%，则通常可假定控制器的复位时间过长。要确定上升时间和稳定时间，则将设定值的容差范围取为 3σ ，并且在 CPM 块的面板中显示该范围。稳定时间和上升时间的绝对值可根据特定应用的具体过程控制要求进行估计。

设定值阶跃变化过程中，与稳定状态相比，控制变量的较大数学方差必定出现，因此，需要抑制由于超出方差限制而生成的警报，直到设定值阶跃变化后稳定过程接近完成。将输出计算出的确定特征，并再次激活随机评估。

图形评估和长期统计

面板的统计视图显示所监视控制回路的最重要测量值，并且这些值将自动存储于过程值归档中。设定值 (SP)、容差范围 (PV_ToHi 和 PV_ToLo) 为 3σ 的过程值 (PV)、调节变量 (MV) 和 CPI 以及二进制状态自动模式 (AutAct)、调节变量上限/下限 (HiLiAct 和 LoLiAct) 激活。可以借助于 WinCC 在线趋势控制的统计功能在可自由选择的（包括较长的）时间段内评估这些曲线：

- 过程值和调节变量的平均值、方差和标准差只能由块本身在短时间窗中计算。
- 您可以将二进制变量的平均值直接解释为时间组件。例如，如果某段时间内 AutAct 的平均值为 0.8，则控制器在这段时间内 80% 处于自动模式；即控制器的“利用率”为 80%。如果 MV_HiLim 的平均值为 0.6，则控制器在这段时间内 60% 处于上限。如果此类问题不是执行器尺寸不合适引起的，则表明执行器可能磨损和断裂。



散点图是调节变量上所有测量值对和过程值的二维表示。在设置正确的控制回路中，数据点应形成一个椭圆形。在散点图中，可以识别非线性特征，如阀中摩擦力的影响（导致生成一个类似平行四边形的轮廓），也可识别量化效应和其它异常统计分布。

用于确定参考方差的供选方案

按计划调试集成了 CPM 的工厂期间, 优化控制器后, 将为每个控制回路初始化 CPM 块, 并且将计算出的方差存储为用于计算 CPI 的参考方差。

另一种方法是, 设置 RefVarExtOn = 1 输入, 从而通过 RefVarExt 输入设置参考变量。有多种方法可获取参考方差的数字值:

- 采用自初始化 CPM 块后此控制回路中曾测量的最小方差。这在 PV_VarMin 输出中显示。仅当初始化 CPM 块后控制回路至少有一次较长时间处于稳定和理想运行状态时, 该值才有用。
- 采用带有理论最小方差控制器的控制回路的方差, 这可使用外部 CPM 工具根据已归档数据获取。这只取决于过程停滞时间和干扰模型。此种形式的 CPI 称为 Harris 指数, 表示 PID 控制器通常不能达到的较低限制, 这就是即使通过良好调整的控制器也很少达到值 100% 的原因。低 CPI 值首先指示控制器设置可以改进。但是, 您应该记住最小方差只是理论上可达到的值, 最小方差控制器具有实际应用中不需要的特征, 例如极大调节变量幅值。因此, 对于基于最小方差的 CPI, 不必想方设法使其尽可能接近 100%。

特殊情况串级控制和多变量控制

在串级控制中, 应将 CPM 块用于主控制器, 不要用于次级控制器。CPM 块无法生成有关次级控制器控制性能的任何有用语句, 因为

- 次级控制回路中的过程值的方差直接取决于被主控制器设置为调节变量的设定值的方差,
- 既不存在带有恒定设定值的运行阶段, 也没有定义的设定值阶跃变化。

除此之外, 从过程控制角度来看, 主控制回路理当然是应对其控制质量进行监视的回路, 而次级回路的控制质量是第二重要的。优化和监视主控制器之前, 建议您仍要仔细设置次级控制器, 因为主控制器无法补偿次级控制器的不良响应。

CPM 块的数学概念意在用于单变量控制回路。如果控制回路中的方差过高, 则块无法识别真正原因是在控制回路中还是这些影响是由于交互而从外部引起的。因此, 如果您注意到工厂中的不同控制回路之间有强交互或者甚至使用多变量控制器, 则您应慎重处理 CPM 块提供的信息。

不过, 仍有必要配备多变量控制器 (如带有控制回路监视的 MPC 块), 以确定在运行期间是否仍然能保持控制器调试期间所达到的控制性能。这种情况下, 多变量控制器的每个控制器通道具有单独的 CPM 块。多个附加逻辑功能需要从 ManSuprCPI 输入上游组态, 如相应仿真模板中所示:

- 如果多变量控制器的一个或多个其它通道处于由 $CPI_SupRoot = 1$ 输出指示的非稳定状态（例如，设定值阶跃变化），则在该控制器通道中无法避免方差暂时增大，而且暂时增大的方差不应引发 CPI 消息。
- 如果多变量控制器的一个或多个其它通道具有由适当的 $CPI_WrnAct = 1$ 输出指示的较高方差（控制性能差），由于交互，这些方差将引起该控制器通道中出现较高方差，同样，该方差也无法避免并且不应引发 CPI 警告。在多变量系统中，按照以下操作可以找到干扰的实际原因：首先检测通道的较高方差，相邻通道中的后继报警受到抑制时设置报警。

注：多变量情况下，根据单变量监视评估的过程增益是不相关的。通过设置输入值 $StatGainValid = 0$ ，这也可以在操作员画面中显示为质量“不确定，过程相关”(Uncertain, process related)。

计算质量代码

该块可以计算和显示多个质量代码。以下输入可用于参数分配：

- **MV_Known**: 如果使用不带有位置反馈的阶跃控制器，请将此输入设置为 0。这使计算出的输出值 **MV_Mean** 和 **StatGain** 无效。
- **StatGainValid**: 如果使用多变量控制器或相邻控制回路之间存在强交互，请将此输入设置为 0。这使计算出的输出值 **StatGain** 无效。如果已知的干扰影响您的过程（例如，批生产过程中的定量给料程序），则还可以使用配方控制临时设置此输入。
- **QC_CPI** 输出的质量代码主要取决于 **CPI_Suppress** 输入：如果 $CPI_Suppress=true$ ，输出无效。除此之外，计算方差过程中出现数值问题的特殊情况下，CPI 也可能变得无效。

评估设定值的阶跃变化时，如果其阶跃变化高度相对于噪声级过低，则将 **QC_OverAbso** 输出的质量代码设置为无效。

报警延迟

如果 CPI 暂时超过了组态的警告和报警限制，不必立即触发报警。控制回路监视的主要目的是需要在各个控制回路中采取维护或优化措施时，发送相应信号。

使用报警延迟功能，可确保只有报警原因存在的时间比组态的时间段 **AlmDelayTime** 长时才能触发报警。通过设置 **AlmOnDelayAct = 1** 可实现此目的。如果设置 **AlmOffDelayAct = 1** 输入，则也会延迟消息的复位，即报警原因必须清除一段时间后（长于组态的延迟），消息才能标记为已退出状态。

参见

CPM I/O (页 61)

CPM 报告 (页 60)

CPM 故障处理 (页 59)

CPM 模式 (页 48)

CPM 的描述 (页 42)

3.2.4 CPM 故障处理

CPM 故障排除

该块没有任何故障号。

参见

CPM I/O (页 61)

CPM 报告 (页 60)

CPM 的功能 (页 48)

CPM 模式 (页 48)

CPM 的描述 (页 42)

3.2.5 CPM 报告

消息响应

如果控制质量下降到定义的限制之下，则会生成一条消息。当设定值中存在阶跃变化时，如果超出定义的超调量限制，也会生成一条消息。

消息文本和消息类别

消息标识符	消息类别		事件	
SIG 1	报警 - 上限	是	\$\$块注释\$\$ 超调 - 超出报警上限	否
SIG 2	警告 - 上限	是	\$\$块注释\$\$ 超调 - 超出警告上限	否
SIG 3	警告 - 下限	是	\$\$块注释\$\$ 超调 - 超出警告下限	否
SIG 4	报警 - 下限	是	\$\$块注释\$\$ 超调 - 超出报警下限	否
SIG 5	<无消息>	否		否
SIG 6	<无消息>	否		否
SIG 7	<无消息>	否		否
SIG 8	<无消息>	否		否

参见

CPM I/O (页 61)

CPM 故障处理 (页 59)

CPM 的功能 (页 48)

CPM 模式 (页 48)

CPM 的描述 (页 42)

3.2.6 CPM I/O

CPM I/O

输入

I/O (参数)	描述	数据类型	默认值
AlmDelayTime	报警延迟时间 [s]	REAL	0
AlmOffDelayAct	1 = 延迟激活离开报警	BOOL	1
AlmOnDelayAct	1 = 延迟激活进入报警	BOOL	1
BA_ID	当前批生产 ID (号)	DWORD	0
BA_NA	当前批生产名称	STRING(32)	"
BATCH_EN	1 = 通过批生产激活控制	BOOL	0
BreakSuppress	在阶跃响应回路结束处断开控制质量指数报警的抑制	BOOL	0
CPI_AlmHys	控制质量指数的报警滞后量 [%]	REAL	5
CPI_LowAlm	控制质量指数的报警下限 [%]	REAL	30
CPI_LowWrn	控制质量指数的警告下限 [%]	REAL	70
CpiAlm_En	1 = 激活下限报警	BOOL	1
CpiAlm_MsgEn	1 = 激活控制性能指数的下限报警消息	BOOL	1
CpiWrn_En	1 = 激活下限警告	BOOL	1
CpiWrn_MsgEn	1 = 激活控制性能指数的下限警告消息	BOOL	1
InitOpEn	1 = 操作员可以初始化块	BOOL	1
InitRefVar	1 = 初始化块。在稳定状态下测量过程值的参考方差以及过程值和调节变量的参考值。	BOOL	0
ManSupprCPI	1 = 手动抑制 CPI 计算和消息 (例如, 在已知干扰期间)	BOOL	0
MsgEvID	消息号 (自动分配)	DWORD	16#29
MV_Known	0 = MV 未知 (专门用于无位置反馈的步控制器)	BOOL	1
MV_Mon	要监视的控制器的调节变量	REAL	0
MV_Ref	调节变量的参考值	REAL	0

I/O (参数)	描述	数据类型	默认值
OCCUPIED	被批生产占用	BOOL	0
OvsAlm_En	1 = 激活下限超调报警	BOOL	1
OvsAlm_MsgEn	1 = 如果值超调下限, 则激活报警消息	BOOL	1
OvsHiAlm	超调报警限制 [%]	REAL	25
OvsHiWrn	超调警告限制 [%]	REAL	15
OvsWrn_MsgEn	1 = 如果值下冲下限则激活警告消息	BOOL	0
PID_AutAct	1 = 闭环控制器处于自动模式	BOOL	1
PV_Mon	要监视的控制器过程值	REAL	0
PV_Ref	过程值的参考值	REAL	0
RefVarExt	控制回路良好状态下的参考值	REAL	0
RefVariance	控制回路良好状态下 PV_Variance 的参考值	REAL	0
ReVaExOn	1 = 使用 RefVarExt 的外部参考值	BOOL	0
RunUpCyc	启动周期数	INT	3
SampleTime	采样时间 [s] (自动分配)	REAL	0.1
SP_Mon	要监视的相应闭环控制器的设定值	REAL	0
STEP_NO	批生产的步号	DWORD	0
StGainValid	0 = StatGain 无效, 过程失败	BOOL	1
TimeWindow	时间窗的宽度 [s]	REAL	120
UserStatus	状态字的空闲用户区域 (位 24 至位 31)	BYTE	16#0

输出

I/O (参数)	描述	数据类型	默认值
CPI	控制质量指数	REAL	100
CPI_AlmAct	1 = 控制质量指数的报警被激活	BOOL	0
CPI_Suppress	1 = 控制性能指数的消息被抑制; 保持上一个有效值	BOOL	1
CPI_SuRoot	抑制控制回路 CPI 消息	BOOL	1
CPI_WrnAct	1 = 控制质量指数下限的消息被激活	BOOL	0
ER_Mean	时间窗中控制误差的平均值 [PV 的单位]	REAL	0
ErrorNum	用于故障消息的 I/O	INT	-1
MsgAckn	ALARM_8P: ACK_STATE 输出	WORD	0
MsgErr	1 = 出现消息处理故障	BOOL	0
MsgStatus	ALARM_8P: STATUS 输出 ALARM_8P 的故障信息	WORD	0
MV_Mean	时间窗中调节变量的平均值 [PV 的单位]	REAL	0
OverAbso	阶跃响应的绝对超调量 [PV 的单位]	REAL	0
超调	阶跃响应的相对超调量 [%]	REAL	10
OvsAlmAct	1 = 超调报警上限被激活	BOOL	0
OvsWrnAct	1 = 超调警告上限被激活	BOOL	0
PV_Mean	时间窗中过程值的平均值 [PV 的单位]	REAL	0
PV_StdDev	过程值的标准偏差	REAL	0
PV_Variance	过程值的方差	REAL	0
PV_VarMin	到目前为止过程值方差的最小值	REAL	10000.0
QC_CPI	CPI 的质量代码	BYTE	16#80
QC_MV_Mean	MV_MEAN 的质量代码	BYTE	16#80
QC_OverAbso	OverAbso 的质量代码	BYTE	16#80
QC_StatGain	StatGain 的质量代码	BYTE	16#80
RefStdDev	控制回路的过程值标准偏差的参考值	REAL	0
RefVariance	控制回路的过程值 PV_Variance 参数的参考值	REAL	0

3.2 CPM: 监视控制回路性能

I/O (参数)	描述	数据类型	默认值
RiseTime	阶跃响应的上升时间 [s]	REAL	0
SettliTime	阶跃响应的稳定时间 [s]	REAL	0
SettlRatio	比率 = RiseTime / SettliTime * 100%	REAL	0.3
StatGain	稳定状态过程增益 [PV/MV 的单位]	REAL	1
状态	状态字	DWORD	0
StepPhase	阶跃响应的阶段 0 = 准备就绪 1 = 正在上升 2 = 超调 3 = 已稳定	INT	0
ToleHiPV	围绕设定值的 3σ 范围的上限	REAL	0
ToleLoPV	围绕设定值的 3σ 范围的下限	REAL	0

参见

CPM 报告 (页 60)

CPM 故障处理 (页 59)

CPM 的功能 (页 48)

CPM 模式 (页 48)

CPM 的描述 (页 42)

3.3 CTRL_PID: PID 控制器块

3.3.1 CTRL_PID 的描述

对象名 (类型 + 编号)

FB61

- CTRL_PID 块 I/O (页 79)
- CTRL_PID 块图标
- CTRL_PID 面板 (页 583)

功能

块 CTRL_PID 是一个连续 PID 控制器，用于设置以下标准控制器电路：

- 固定设定值控制
- 级联控制 (单/多级联)
- 比率控制
- 同步控制
- 比例控制

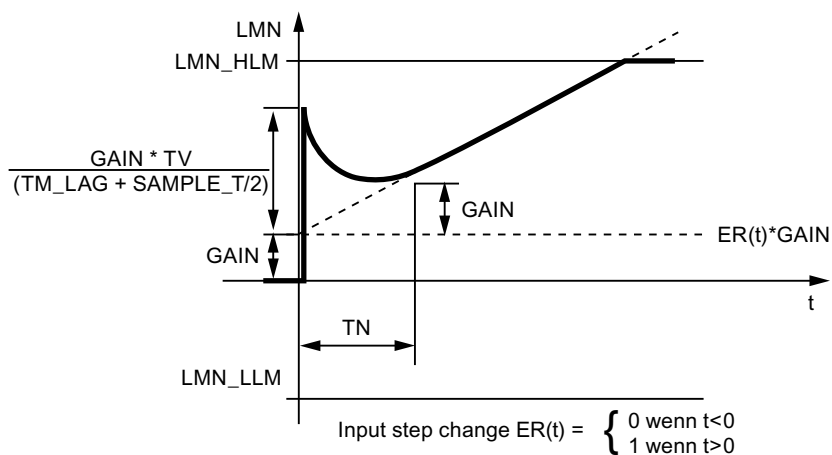
除实际控制器功能之外，控制器块还提供了以下处理选项：

- 模式：手动模式、自动或跟踪
- 通过 ALARM_8P 块对控制变量、错误信号及消息生成进行限制监视
- 前馈控制
- 设定值跟踪 (SP = PV_IN)
- 设定值和过程值范围设置 (物理标准化)
- 设置调节变量值的范围 (物理反向标准化)
- 错误信号分支中的死区 (响应阈值)
- 可单独启用和禁用 P、I 和 D 动作
- 可将 P 和 D 动作置于反馈路径中
- P 或 PD 控制器模式的运行点设置

3.3 CTRL_PID: PID 控制器块

工作原理

该块可用作 PID 控制器（带延迟微分动作），步响应如下所示。积分器按照梯形规则操作。



CTRL_PID 的步响应

说明

输入参数 **LMNR_IN** 作为调节变量显示在面板上（循环显示）。如果在过程中没有得到位置反馈，可将调节变量输出 **LMN** 与 **CFC** 中的 **LMNR_IN** 互连以循环显示调节变量。

调用 OB

安装了该块的循环中断 **OB**（例如 **OB32**）和 **OB100**。

更多信息

更多相关信息, 可参考:

CTRL_PID 的设定值和过程值分支中的信号处理 (页 68)

CTRL_PID 调节变量的生成 (页 70)

CTRL_PID 的手动、自动和跟踪模式及级联 (页 72)

CTRL_PID 模式更改 (页 74)

CTRL_PID 的错误处理 (页 76)

CTRL_PID 的启动特征、时间响应和消息响应 (页 77)

CTRL_PID 的方框图 (页 78)

CTRL_PID 的 I/O (页 79)

CTRL_PID 的消息文本和关联值 (页 84)

CTRL_PID 的 VSTATUS (页 86)

归档过程值 (页 728)

操作和监视 CTRL_PID (页 86)

3.3.2 CTRL_PID 的设定值和实际值分支中的信号处理

设定值的生成

可从三个不同的来源获得设定值 SP。下表定义所使用的来源，这取决于 SP_TRK_ON 和 SPEXTSEL_OP 输入的状态：

SP_TRK_ON	SPEXTSEL_OP	SP=	状态
0	0	SP_OP	内部设定值
不相关	1	SP_EXT	外部设定值
1	0	PV_IN **	跟踪设定值
		** 仅手动模式且 SPBUMPON = 1 时	

内部设定值

使用 OP_A_LIM (页 554) 设置和限制内部设定值 SP_OP (范围是 SP_LLM - SP_HLM)。

外部设定值

可互连外部设定值 SP_EXT 并限制在 (SPEXTLLM - SPEXTHLM) 范围内。

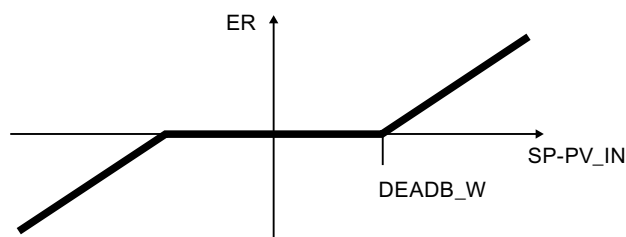
如果设定值斜坡已激活 (SPRAMPOF = 0)，则内部或外部设定值的变化受最大梯度 (SPDRLM、SPURLM) 的限制。

跟踪设定值

如果 SP_TRK_ON = 1，则使用控制变量 PV_IN 作为设定值。跟踪设定值至过程值仅在手动模式下启用 (使用内部设定值且当 SPBUMPON = 1 时)，主要用于在从手动模式切换到自动模式时提供合适的设定值。

误差信号生成

误差信号是基于有效设定值 SP 和过程值 PV_IN 生成的，并可在死区 DEADB_W 后从输出 ER 处获得。



误差信号监视

使用公用滞后量 (ER_HYS) 监视误差信号 ER 的报警限制值 (ERL_ALM、ERH_ALM)，并在相应的输出 (QERL_ALM、QERH_ALM) 指示。

过程值监视

使用公用滞后量 (HYS) 监视过程值 PV_IN 的警告和报警限制值 (PVL_ALM、PVL_WRN、PVH_WRN、PVH_ALM)，并在相应的输出 (QPVL_ALM、QPVL_WRN、QPVH_WRN、QPVH_ALM) 显示。

物理标准化

将误差信号 ER 标准化为基于过程值物理测量范围 (NM_PVHR、NM_PVLR) 的百分比。

$$ER_{normalized} = \frac{ER}{NM_PVHR - NM_PVL R} \cdot 100$$

执行 PID 算法后，将调节变量从百分比转换回调节值的物理测量范围 (NM_LMNHR、NM_LMNL R)。

$$LMN = NM_LMNL R + \frac{LMN_{normalized}}{100} \cdot (NM_LMNHR - NM_LMNL R)$$

按照过程值的物理测量范围输入内部或外部设定值、过程值以及相应参数。

手动值、调节变量的跟踪值、前馈控制和相应参数按照调节值的物理测量范围进行设置。

控制器 GAIN 按标准化 (无量纲) 格式指定。

3.3 CTRL_PID: PID 控制器块

3.3.3 CTRL_PID 调节变量的生成

调节变量 LMN

可从三个不同的来源获得调节变量 LMN。下表指定了使用的是哪个来源，这取决于 LMN_SEL、LIOP_MAN_SEL、AUT_L 和 AUT_ON_P 输入的状态：

LMN_SEL	LIOP_MAN_SEL	AUT_L	AUT_ON_OP	LMN=	状态
0	0	X	0	MAN_OP (受限制)	手动模式，通过 OS 设置
0	0	X	0	MAN_OP (受限制)	手动模式，通过 OS 设置
0	0	X	1	由 PID 算法计算	自动模式，通过 OS 设置
0	1	0	X	MAN_OP (受限制)	手动模式， 当 AUT_L = 0 时设置
0	1	0	X	MAN_OP (受限制)	手动模式， 当 AUT_L = 0 时设置
0	1	1	X	由 PID 算法计算	自动模式， 当 AUT_L = 1 时设置
1	X	X	X	LMN_TRK	跟踪的调节变量
X = 任何状态					

- 如果 LIOP_MAN_SEL = 0，则 AUT_ON_OP 参数将用于在 OS 上于手动模式和自动模式之间切换。
- 如果 LIOP_MAN_SEL = 1，则 AUT_L 参数将用于 CFC 中通过互连在手动模式和自动模式之间切换。
- 只有通过 LMN_SEL 参数的互连才能启用跟踪模式。跟踪模式优先于手动和自动模式。

在自动模式中，按照以下算法生成标准化调节变量：

$$LMN_{normalized} = GAIN \cdot \left(1 + \frac{1}{TN \cdot s} + \frac{TV \cdot s}{1 + TM_LAG \cdot s} \right) \cdot ER_{normalized}$$

s: 复杂编号

然后，反向标准化调节变量。

说明

此公式描述了一种标准情况，其中启用了 P、I 和 D 动作，并且 P 和 D 动作不在反馈路径中。（P_SEL = TRUE, TN <> 0, PFDB_SEL = FALSE 且 DFDB_SEL = FALSE）。

例如，如果 TN = 0，则加上从物理变量 LMN_OFF（运行点）计算出的偏移量。更多相关信息，可参考“CTRL_PID 的方框图 (页 78)”。

前馈控制和限制

在自动模式中，将干扰变量 DISV 加到 PID 算法的输出上。结果限制在 LMN_LLM 到 LMN_HLM 范围之间。

3.3.4 CTRL_PID 的手动、自动和跟踪模式及级联

手动模式

操作员在 OS 上通过输入 MAN_OP 来设置调节变量。调节变量通过 OP_A_LIM (页 554) 块设置和限制 (范围为 MAN_HLM - MAN_LLM)。OP_A_LIM 块的 QVHL 和 QVLL 输出值传送到 QLMN_HLM 和 QLMN_LLM 输出。

自动模式

PID 算法计算调节变量。默认情况下,无法互连控制参数 GAIN、TN、TV 和 TM_LAG。如果由于特殊应用 (如增益计划) 而必须将它们互连,则必须修改相应的系统属性 "s7_link"。请注意,自动模式下的参数更改可能会导致调节变量步进变化。

- 通过设置负比例 GAIN 值,可颠倒控制动作的有效方向 (上升误差信号将导致调节变量下降)。可通过设置 P_SEL = 0 禁用 P 动作。可通过设置 TN(TI) = 0 禁用 I 动作。如果要在自动模式下限制调节变量 LMN,则将积分器设置为保持 (防终止)。通过取反参数 TN(TI) 的符号,可颠倒积分器的动作方向。
- 运行点 (输入 LMN_OFF): 如果禁用 I 动作,则在自动模式下此值替换 PID 算法的 I 动作。运行点按照调节变量的测量范围进行输入。
- D 动作作为带延迟的微分元件来实现。可通过设置 TV(TD) = 0 将其禁用。通过取反参数 TV(TD) 值的符号,可颠倒微分器的动作方向。时延常量 TM_LAG 应与微分时间 TV(TD) 具有适当的关系。此比率也称作微分增益 (D 动作的最大单步响应),其通常位于 $5 < TV(TD)/TM_LAG < 10$ 范围中。
- 将 P 动作应用于反馈路径: PFDB_SEL = TRUE 时会在反馈路径中启用 P 动作。控制值步进变化不影响 P 动作,所以在不更改干扰特征的情况下可以减少或避免设定值步进变化所导致的过冲。在自动模式下切换 PFDB_SEL 会导致调节变量的极端变化。因此,建议在手动模式下更改 PFDB_SEL。
- 将 D 动作应用于反馈路径: 当 DFDB_SEL = TRUE 时会在反馈路径中启用 D 动作。在这种情况下,控制步进变化不影响 D 动作。DFDB_SEL 的切换不是无波动的。

跟踪模式

在此状态 (LMN_SEL = 1) 下,调节变量从互连的跟踪值 LMN_TRK 中获取并被设置在输出位置。输出 QLMN_HLM 和 QLMN_LLM 设置为 FALSE。“跟踪”模式优先于其它所有模式,以便您可以使用此输入为工厂组态一个安全关闭电路。

将 D 和 P 动作置于反馈路径中

P 和 D 动作可位于反馈路径中, 以降低或避免因设定值步进变化引起的过程值的过冲。在此模式下设定值步进变化不再对 P 或 D 动作产生任何影响。调节值不再对设定值步进变化后的步进变化做出响应。P 动作通过设置 PFDB_SEL = 1 在反馈路径中启用。D 动作通过设置 DFDB_SEL = 1 在反馈路径中启用。

如果 $TI = 0$, 将忽略 P 动作的反馈。

级联 PID 控制器

主控制器的调节变量 LMN 与二级控制器的输入 SP_EXT 相连。如果打开级联, 则必须将主控制器也更改为跟踪模式。QCAS_CUT 信号在二级控制器中生成, 并且与主控制器的 LMN_SEL 输入互连。通过设定值输入, 或通过二级控制器的设定值跟踪可以在手动或跟踪模式下将此互连断开:

```
QCAS_CUT = NOT(QMAN_AUT) OR LMN_SEL OR SP_TRK_ON OR  
NOT(QSPEXT_ON)
```

为防止级联再次闭合时发生步进变化, 将一级控制器的跟踪输入 LMN_TRK 与二级控制器的 SP 输出互连。

同时建议设置只要达到调节变量限制就激活的主控制器中的积分器的方向锁定。通过正控制动作, 将主控制器的 INT_HPOS 和 INT_HNEG 输入与二级控制器的 QLMN_HLM 和 QLMN_LLM 输出互连。

3.3.5 CTRL_PID 模式更改

模式更改

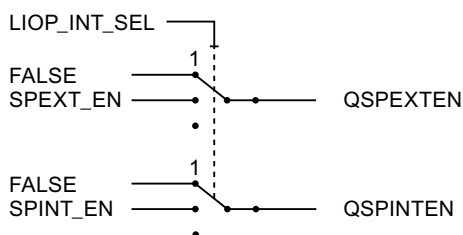
模式更改可由操作员或通过互连的输入触发。

外部/内部设定值

OS 操作员通过互连 SPEXON_L 设置 SPEXTSEL_OP 输入来启动外部和内部设定值之间的切换。设置相应的使能输入 SPINT_EN、SPEXT_EN 或选择输入 LIOP_INT_SEL 来启用这些切换。

如果 SPBUMPON = 1, 则用有效设定值来代替内部设定值, 从而实现从外部模式或跟踪模式到内部模式的无波动切换。

启用内部和外部设定值之间的切换



QSPEXTEN = TRUE: 可将 SPEXTSEL_OP 从 FALSE (内部设定值) 更改为 TRUE (外部设定值)。

QSPINTEN = TRUE: 可将 SPEXTSEL_OP 从 TRUE (外部设定值) 更改为 FALSE (内部设定值)。

根据需要对 SPEXTSEL_OP 进行跟踪或复位。

通过操作员输入 SP_OP_ON 启用设定值控制

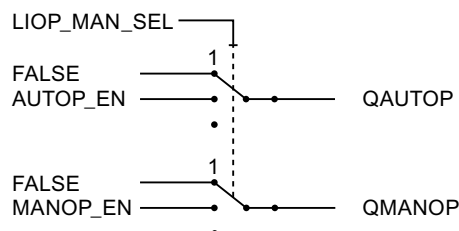
Q_SP_OP = TRUE: 可调整设定值 SP_OP。

根据需要对设定值 SP_OP 进行跟踪或复位。

手动/自动

通过 OS 操作员设置输入 AUT_ON_OP 或互连 AUT_L 启动手动和自动模式之间的切换。通过设置相应的使能输入 MANOP_EN、AUTOP_EN 或选择输入 LIOP_MAN_SEL 启用此切换。

启用手动和自动模式之间的切换

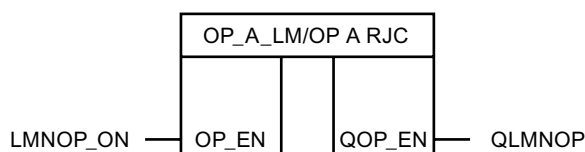


QAUTOP = TRUE: 可将 AUT_ON_OP 从 FALSE (手动模式) 更改为 TRUE (自动模式)。

QMANOP = TRUE: 可将 AUT_ON_OP 从 TRUE (自动模式) 更改为 FALSE (手动模式)。

根据需要对 SPEXTSEL_OP 进行跟踪或复位。

通过操作员控制输入启用手动值操作员控制



QLMNOP = TRUE: 可设置 MAN_OP。

根据需要对 MAN_OP 进行跟踪或复位。

为确保无波动切换, 针对下列模式采取了特殊措施:

- 外部设定值/跟踪设定值: 如果 SPBUMPON = TRUE, 内部设定值 SP_OP 将设置为有效 (外部或已跟踪) 设定值。
- 自动模式: 将跟踪手动值 MAN_OP 至有效的操纵变量。
- 跟踪模式: 将跟踪手动值 MAN_OP 至有效的操纵变量。
- 手动或跟踪模式: 跟踪积分器, 以便无波动切换至自动模式。

动作 = 操纵变量值 (百分比) - P 动作 - 干扰变量 (百分比)

注意: 应用上述公式时, 如果在切换时有异常的测量值 (换句话说, 有极高的 P 动作), 则积分器可能载有极高的数字值。自 V6.0 起, 已实施了附加措施, 以便能够灵活地限制 I 动作。

将禁用并补偿 D 动作分量。

3.3.6 CTRL_PID 的错误处理

操作员输入错误

在 SPEXTSEL_OP、AUT_ON_OP、SP_OP 或 MAN_OP 参数中，如果在更改其中某个参数时至少出现了一个操作员错误，将设置 QOP_ERR = 1。否则，设置 QOP_ERR = 0。操作员错误仅存在一个周期。

块算法处理以下情况：

- 参数分配错误 $NM_PVHR \leq NM_PVL R$ ：
错误信号 ER 设为零，并设置 $ENO = 0$ 或 $QERR = 1$ 。
- $NM_LMNHR \leq NM_LMNLR$ ：
在自动模式中，将输出干扰变量，并设置 $ENO=0$ 或 $QERR=1$ 。
- 绝对值 $(TN) < SAMPLE_T/2$ ：
如果 $TN(TI) > 0$ ，则用 $TN(TI) = SAMPLE_T/2$ 计算；如果 $TN(TI) < 0$ ，则用 $TN(TI) = -SAMPLE_T/2$ 计算。
如果 $TN(TI) = 0$ ，则关闭积分器并激活运行点 LMN_OFF 。
- 绝对值 $(TV) < SAMPLE_T$ ：
如果 $TV(TD) > 0$ ，则用 $TV(TD) = SAMPLE_T$ 计算；如果 $TV(TD) < 0$ ，则用 $TV(TD) = -SAMPLE_T$ 计算。
如果 $TV(TD) = 0$ ，则关闭微分器。
- $TM_LAG < SAMPLE_T/2$ ：
如果 $TM_LAG < SAMPLE_T/2$ ，则在内部用 $TM_LAG = SAMPLE_T/2$ 计算。在此类情况下，D 动作充当理想微分器。

3.3.7 CTRL_PID 的启动行为、动态响应和消息响应

启动特征

在 CPU 启动期间，使用内部设定值将 CTRL_PID 设置为手动模式。这意味着必须在启动 OB 中调用该块。在 CFC 工程组态中，这由 CFC 处理。使用 STEP 7 基本工具时，在启动 OB 中输入此调用。

启动后，消息将在 RUNUPCYC 所设置的周期数内受到抑制。

时间响应

块必须在循环中断 OB 中调用。块的采样时间在 SAMPLE_T 参数中设置。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“CTRL_PID 的 VSTATUS (页 86)”。

消息响应

CTRL_PID 块使用 ALARM_8P 块生成消息。

存在下列消息触发器：

- 用于监视过程值和系统偏差限制的功能
- 通过互连作为控制系统错误而接收的 CSF 信号

通过使用相关的 M_SUP_xx 输入可以分别抑制超出限制消息。可使用 MSG_LOCK 集中禁用过程消息（不是控制系统消息！）。

如果重新启动以来未满 RUNUPCYC 个周期，并且 MSG_LOCK = TRUE 或 MSG_STAT = 21，则会设置 QMSG_SUP。

下表列出了 CTRL_PID 块的消息文本 (页 84)及其向块参数的分配。

监视过程值

不可用

3.3.9 CTRL_PID 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认 值	类 型	OC M	允许 值
AUT_L	用于 MANUAL (手动) /AUTO (自动) 模式的可互连输入: (0 = 手动 /1 = 自动)	BOOL	0	I		
AUT_ON_OP	操作员输入: 0 = 手动; 1 = 自动	BOOL	0	IO	+	
AUTOP_EN	1 = 自动模式的操作员控制使能信号	BOOL	1	I		
AUX_PRx	关联值 x	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 连续批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	"	I	+	
CSF	控制系统错误	BOOL	0	I		
DEADB_W	死区	REAL	0	I	+	≥ 0
DFDB_SEL	将 D 动作置于反馈中 (1: 激活)	BOOL	0	I		
DISV	干扰变量	REAL	0	I		
ER	误差信号	REAL	0	O	+	
ER_HYS	用于监视误差信号的滞后量	REAL	0.1	I	+	≥ 0
ERH_ALM	误差信号: 报警上限	REAL	100	I	+	$> DEADBW$
ERL_ALM	误差信号: 报警下限	REAL	-100	I	+	$< -DEADBW$
GAIN	比例增益	REAL	1	I	+	
HYS	滞后量	REAL	5	I	+	≥ 0
INT_HNEG	冻结 I 动作 (负方向); 1 = 激活	BOOL	0	I		
INT_HPOS	冻结 I 动作 (正方向); 1 = 激活	BOOL	0	I		
LIOP_INT_SEL	1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		

3.3 CTRL_PID: PID 控制器块

I/O (参数)	含义	数据类型	默认 值	类 型	OC M	允许 值
LIOP_MAN_SE L	1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
LMN	调节值输出	REAL	0	O		
LMN_HLM	调节变量值的上限	REAL	100	I	+	LMN_HLM > LMN_LLM
LMN_LLM	调节变量值的下限	REAL	0	I	+	LMN_LLM < LMN_HLM
LMN_OFF	运行点	REAL	0	I	+	
LMN_SEL	外部调节变量; 1 = 激活	BOOL	0	I		
LMN_TRK	外部调节变量	REAL	0	I		
LMNOP_ON	1 = 用于调节变量 MAN_OP 值的操 作员控制使能	BOOL	1	I		
LMNR_IN	用于在 OS 上显示的位置反馈	REAL	0	I		
M_SUP_AH	1 = 消息抑制报警上限过程值	BOOL	0	I	+	
M_SUP_AL	1 = 消息抑制报警下限过程值	BOOL	0	I	+	
M_SUP_ER	抑制误差信号报警消息	BOOL	1	I	+	
M_SUP_WH	1 = 消息抑制: 过程值警告上限	BOOL	0	I	+	
M_SUP_WL	1 = 消息抑制警告下限过程值	BOOL	0	I	+	
MAN_HLM	调节变量手动值上限	REAL	100	I	+	
MAN_LLM	调节变量手动值下限	REAL	0	I	+	
MAN_OP	调节变量的操作员输入	REAL	0	IO	+	
MANOP_EN	1 = 手动模式的操作员控制使能信号	BOOL	1	I		
MO_PVHR	显示上限 (测量范围)	REAL	110	I	+	
MO_PVLR	显示下限 (测量范围)	REAL	-10	I	+	
MSG_ACK	确认消息	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		
MSG_LOCK	1 = 消息抑制取决于过程状态	BOOL	0	I	+	
MSG_STAT	错误消息状态	WORD	0	O		
NM_LMNHR	调节变量标准化上限 (测量范围)	REAL	100	I		

I/O (参数)	含义	数据类型	默认 值	类 型	OC M	允许 值
NM_LMNL	调节变量标准化下限 (测量范围)	REAL	0	I		
NM_PVHR	过程值标准化上限 (测量范围)	REAL	100	I		
NM_PVLR	过程值标准化下限 (测量范围)	REAL	0	I		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
OPTI_EN	1 = 控制器调节开启, 0 = 关闭	BOOL	0	I	+	
P_SEL	激活 P 动作 (1: 激活)	BOOL	1	I		
PFDB_SEL	将 P 动作置于反馈中; 1 = 激活	BOOL	0	I		
PV_IN	过程值	REAL	0	IO	+	
PVH_ALM	过程值: 报警上限	REAL	100	I	+	PVH_ALM > PVL_ALM
PVH_WRN	过程值: 警告上限	REAL	95	I	+	PVH_WRN > PVL_WRN
PVL_ALM	过程值: 报警下限	REAL	0	I	+	PVL_ALM < PVH_ALM
PVL_WRN	过程值: 警告下限	REAL	5	I	+	PVL_WRN < PVH_WRN
Q_SP_OP	1: 1 = 用于设置设定值的操作员控制使能	BOOL	0	O	+	
QAUT_OP	1 = 自动模式的操作员控制使能信号	BOOL	0	O	+	
QC_LMN	LMN 的质量代码	BYTE	16# 80	O		
QC_LMN_I	输出 LMN 的质量代码	BYTE	16# 80	I		
QC_LMNR_IN	LMNR_IN 的质量代码	BYTE	16# 80	I		
QC_PV_IN	PV_IN 的质量代码	BYTE	16# 80	I		
QCAS_CUT	1 = 级联已打开	BOOL	1	O		
QDNRLM	1 = 限制负设定值斜坡	BOOL	0	O		

3.3 CTRL_PID: PID 控制器块

I/O (参数)	含义	数据类型	默认 值	类 型	OC M	允许 值
QERH_ALM	误差信号, 1 = 报警上限	BOOL	0	O	+	
QERL_ALM	误差信号, 1 = 报警下限	BOOL	0	O	+	
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	
QLMN_HLM	1 = 限制调节变量输出上限值	BOOL	0	O	+	
QLMN_LLM	1 = 限制调节变量输出下限值	BOOL	0	O	+	
QLMNOP	1 = 设置调节变量的操作员控制使能信号	BOOL	0	O	+	
QMAN_AUT	0 = 手动, 1 = 自动	BOOL	0	O	+	
QMANOP	1 = 手动模式的操作员控制使能信号	BOOL	0	O	+	
QMSG_ERR	1 = 消息错误	BOOL	0	O		
QMSG_SUP	1 = 消息抑制	BOOL	0	O	+	
QOP_ERR	1 = 组操作员输入错误	BOOL	0	O		
QPVH_ALM	1 = 报警上限	BOOL	0	O		
QPVH_WRN	1 = 警告上限	BOOL	0	O		
QPVL_ALM	1 = 报警下限	BOOL	0	O		
QPVL_WRN	1 = 警告下限	BOOL	0	O		
QSP_HLM	1 = 限制设定值输出上限	BOOL	0	O		
QSP_LLM	1 = 限制设定值输出下限	BOOL	0	O		
QSPEXTEN	1 = 外部模式的操作员控制使能信号	BOOL	0	O	+	
QSPEXTON	0 = 内部, 1 = 外部	BOOL	0	O	+	
QSPINTEN	1 = 内部模式的操作员控制使能信号	BOOL	0	O	+	
QUPRLM	1 = 限制正设定值斜坡	BOOL	0	O		
RUNUPCYC	起动周期数	INT	3	I		
SAMPLE_T	采样时间 (秒)	REAL	1	I		>= 0.001
SP	激活设定值	REAL	0	O	+	
SP_EXT	外部设定值	REAL	0	I		
SP_HLM	设定值上限	REAL	100	I	+	SP_HLM > SP_LLM

I/O (参数)	含义	数据类型	默认 值	类 型	OC M	允许 值
SP_LLM	设定值下限	REAL	0	I	+	SP_LLM < SP_HLM
SP_OP	设定值的操作员输入	REAL	0	IO	+	
SP_OP_ON	设定值 SP_OP 的操作员控制使能	BOOL	1	I		
SP_TRK_ON	1 = 跟踪设定值 SP_OP	BOOL	0	I	+	
SPBUMPON	1 = 无波动设定值	BOOL	1	I	+	
SPDRLM	最大负设定值斜率 [1/s]	REAL	100	I	+	
SPEXON_L	内部/外部模式的互连输入: 0 = 内部, 1 = 外部	BOOL	0	I		
SPEXT_EN	1 = 外部设定值的操作员控制使能	BOOL	1	I		
SPEXTHLM	外部设定值上限	REAL	100	I		SPEXTHLM > SPEXTLLM
SPEXTLLM	外部设定值下限	REAL	0	I		SPEXTLLM < SPEXTHLM
SPEXTSEL_OP	操作员输入: 0 = 内部, 1 = 外部	BOOL	0	IO	+	
SPINT_EN	1 = 内部模式的操作员控制使能信号	BOOL	1	I		
SPRAMPOF	1 = 禁用设定值斜坡限制	BOOL	1	I	+	
SPURLM	最大正设定值斜率 [1/s]	REAL	100	I	+	
STEP_NO	BATCH 步号	DWORD	0	I	+	
TM_LAG	D 动作的时延 (秒)	REAL	1	I	+	$\geq \pm \text{SAMPLE_T}/2$
TN	复位时间 (秒)	REAL	10	I	+	TN=0, $\geq \pm \text{SAMPLE_T}/2$
TV	微分时间 (秒)	REAL	0	I	+	TV=0, $\geq \pm \text{SAMPLE_T}$
USTATUS	VSTATUS 中的状态字, 可进行用户特定的组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.3.10 CTRL_PID 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由 此项抑制
1	QPVH_ALM	PV:\$\$BlockComment\$\$ 上上限报警	AH	M_SUP_AH 、 MSG_LOCK
2	QPVH_WRN	PV:\$\$BlockComment\$\$ 上限报警	WH	M_SUP_WH 、 MSG_LOCK
3	QPVL_WRN	PV:\$\$BlockComment\$\$ 下限报警	WL	M_SUP_WL 、 MSG_LOCK
4	QPVL_ALM	PV:\$\$BlockComment\$\$ 下下限报警	AL	M_SUP_AL 、 MSG_LOCK
5	CSF	\$\$BlockComment\$\$ 外部故障	S	-
6	QERH_ALM	ER:\$\$BlockComment\$\$ 上上限报警	AH	M_SUP_ER 、 MSG_LOCK
7	QERL_ALM	ER:\$\$BlockComment\$\$ 下下限报警	AL	M_SUP_ER 、 MSG_LOCK

将关联值分配给块参数

消息块的前三个关联值已分配给 SIMATIC BATCH 数据，第四个关联值留供 PV_IN 使用，其余关联值 (AUX_PRx) 可由用户随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	PV_IN
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR010

3.3 CTRL_PID: PID 控制器块

3.3.11 CTRL_PID 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	-	-	-	QSPEXTON	QMAN_AUT	MSG_LOCK	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	LMN_SEL	-	-	-	-	-

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.3.12 CTRL_PID 的操作和监视

更多信息

更多信息，请参考以下部分：

- CTRL_PID 块图标 (页 687)
- CTRL_PID 面板 (页 583)

3.4 CTRL_S: PID 步控制器块

3.4.1 CTRL_S 的描述

对象名 (类型 + 编号)

FB76

- CTRL_S 块 I/O (页 107)
- CTRL_S 块图标 (页 685)
- CTRL_S 面板 (页 728)

功能

CTRL_S 控制器块是过程控制系统的步控制器，其中使用了积分动作执行器（例如，电机驱动的阀门）。阀门由两个二进制控制信号控制。

步控制器的工作原理基于采样控制器和下游位置控制器的 PID 算法组合。连续的控制信号会转换成一系列控制脉冲。

使用适当的参数设置可启用或禁用 PID 算法的以下子功能，以适应受控系统：

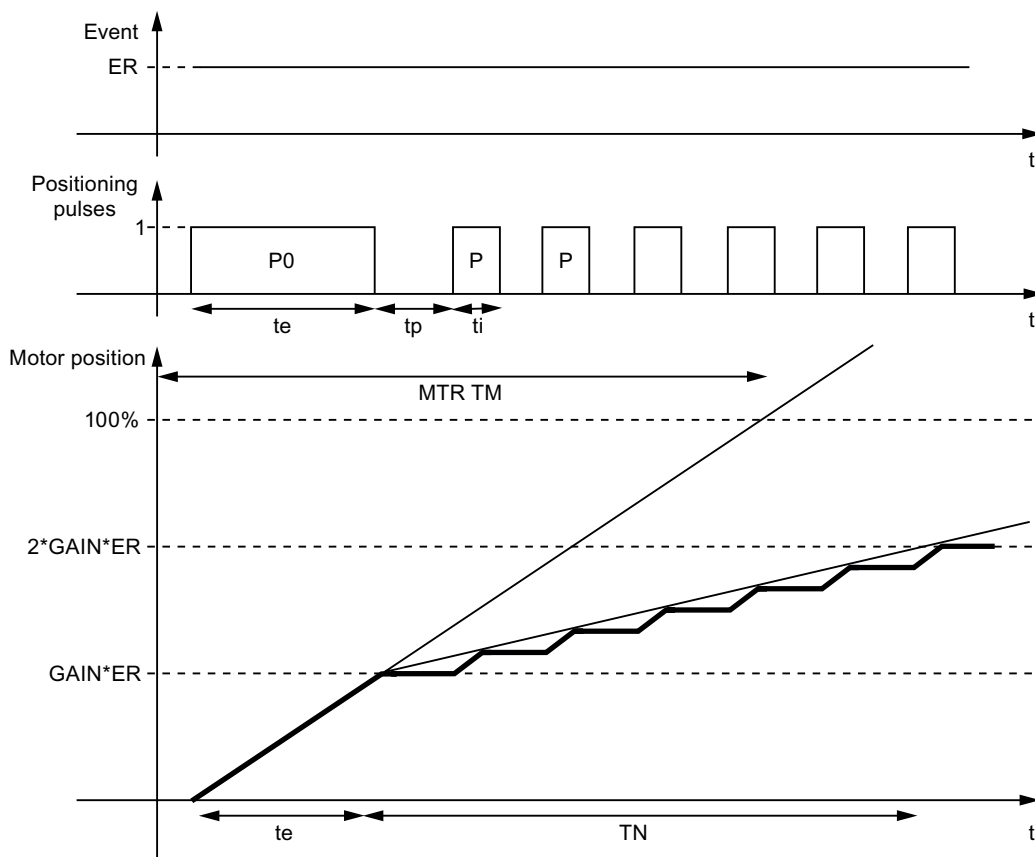
- 模式：手动模式、自动或跟踪
- 通过 ALARM_8P 块监视控制变量、误差信号以及消息生成的限制
- 前馈控制
- 设定值跟踪 (SP = PV_IN)
- 设定值和过程值范围设置 (物理标准化)
- 可单独启用和禁用 P、I 和 D 动作
- P 和 D 动作可应用于反馈路径 (P 动作只能在带位置反馈的步控制器中)。
- P 或 PD 控制器模式的运行点设置
- 下游位置控制器允许以下应用：
 - 通过位置反馈信号控制
 - 不通过位置反馈信号控制
 - 通过操作员或互连信号直接调整信号

3.4 CTRL_S: PID 步控制器块

- 根据来自电机（电机保护）或阀门（限位停止信号）的相应反馈信号来抑制控制信号
- 通过调整响应阈值减少控制脉冲的数目

工作原理

PI 步控制器是常用的应用程序。在此模式中，控制器的步响应如下所示：



标识:

P0	启动脉冲
P	顺序脉冲
t0	立即启动
te	启动脉冲持续时间
ti	脉冲持续时间 (= PULSE_TM)
tp	暂停持续时间 (取决于参数分配, 从而不与 BREAK_TM 相对应)

说明

面板 (循环显示) 显示为输入参数 LMNR_IN 的有效调节变量。位置反馈与此参数互连。可使用控制输入 LMNR_ON 在控制算法中设置此变量。如果 LMNR_ON = 0, 控制器不通过位置反馈进行操作。

如果因在位置反馈中出现延迟而导致在控制阀作用下产生振荡, 则使用无位置反馈的步控制器并将反馈切换到输入 PV_IN。您可在 PI 参数 GAIN 和 TN 中补偿控制阀的停滞时间响应。使用 CTRL_PID 作为主控制器 (另请参见级联控制) 来控制实际过程。

3.4 CTRL_S: PID 步控制器块

调用 OB

安装了该块的循环中断 OB（例如 OB32）和 OB100。

更多信息

更多相关信息，可参考：

CTRL_S 的设定值和过程值分支中的信号处理 (页 91)

CTRL_S 的控制信号生成 (页 93)

CTRL_S 的手动、自动和跟踪模式及级联 (页 97)

CTRL_S 模式更改 (页 100)

CTRL_S 的错误处理 (页 103)

CTRL_S 的启动行为、动态响应和消息响应 (页 104)

CTRL_S 的方框图 (页 105)

CTRL_S 的 I/O (页 107)

CTRL_S 的消息文本和关联值 (页 114)

CTRL_S 的 VSTATUS (页 116)

归档过程值 (页 728)

操作和监视 CTRL_S (页 592)

参见

CTRL_S 的操作和监视 (页 116)

3.4.2 CTRL_S 的设定值和实际值分支中的信号处理

设定值的生成

可从三个不同的来源获得设定值 SP。下表定义所使用的来源，这取决于 SP_TRK_ON 和 SPEXTSEL_OP 输入的状态：

SP_TRK_ON	SPEXTSEL_OP	SP=	状态
0	0	SP_OP	内部设定值
不相关	1	SP_EXT	外部设定值
1	0	PV_IN **	跟踪设定值
		** 仅手动模式且 SPBUMPON = 1 时	

内部设定值

使用 OP_A_LIM (页 554) 设置和限制内部设定值 SP_OP (范围是 SP_LLM - SP_HLM)。

外部设定值

可互连外部设定值 SP_EXT 并限制在 (SPEXTLLM - SPEXTHLM) 范围内。

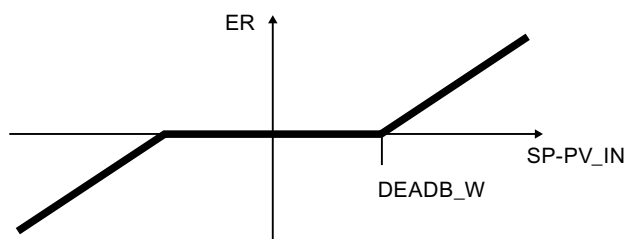
如果设定值斜坡已激活 (SPRAMPOF = 0)，则内部或外部设定值的变化受最大梯度 (SPDRLM、SPURLM) 的限制。

跟踪设定值

如果 SP_TRK_ON = 1，则使用控制变量 PV_IN 作为设定值。过程值的设定值跟踪仅在手动模式下激活 (使用内部设定值且当 SPBUMPON = 1 时)，主要用于在从手动模式变为自动模式时提供合适的设定值。

误差信号生成

误差信号是基于有效设定值 SP 和过程值 PV_IN 生成的，并可在死区 DEADB_W 后从输出 ER 处获得。



误差信号监视

使用公用滞后量 (ER_HYS) 监视误差信号 ER 的报警限制值 (ERL_ALM、ERH_ALM)，并在相应的输出 (QERL_ALM、QERH_ALM) 指示。

过程值监视

使用公用滞后量 (HYS) 监视过程值 PV_IN 的警告和报警限制值 (PVL_ALM、PVL_WRN、PVH_WRN、PVH_ALM)，并在相应的输出 (QPVL_ALM、QPVL_WRN、QPVH_WRN、QPVH_ALM) 显示。

物理标准化

将误差信号 ER 标准化为基于过程值物理测量范围 (NM_PVHR、NM_PVLR) 的百分比。

$$ER_{normalized} = \frac{ER}{NM_PVHR - NM_PVLR} \cdot 100$$

按照过程值的物理测量范围输入内部或外部设定值、过程值以及相应参数。

阀门的操作范围标准化为 0 到 100 之间。手动值、调节变量的跟踪值和前馈以百分比形式输入。

控制器 GAIN 按标准化（无量纲）格式指定。

3.4.3 CTRL_S 控制信号的生成

控制信号

控制信号可从各种来源生成。下表指定了使用的是哪个来源，这取决于控制输入的状态：

编号	工作原理	设定值 内部/外部	位置 反馈	有效来源	跟踪的信号	备注
				通常：根据设置， LMN_OFF 和 DISV 在自 动模式下也可以有效。	通常：仅当 SPBUMPON=1 时 才跟踪 SP_OP。	
2	启动/重新启动	-	-	启动值	启动值	
4	手动/跟踪	-	支持	MAN_OP	SP_OP=SP_EXT/ PV_IN	调节变量的操作员 控制
8			支持	LMN_TRK	SP_OP=SP_EXT/ PV_IN MAN_OP=LMNR_I N	跟踪外部调节变量
5			-	LMNUP_OP/LMNDN_OP	SP_OP=SP_EXT/ PV_IN MAN_OP= LMNR_IN	OS 上控制信号的 操作员控制
6			-	LMNUP/LMNDN	SP_OP=SP_EXT/P V_IN MAN_OP = LMNR_IN	通过互连直接调整 信号
9	自动	内部	-	SP_OP，来自 PID 算法 的控制信号	MAN_OP = LMNR_IN	源自 OS 的设定值
12				SP_EXT，来自 PID 算法 的控制信号	SP_OP=SP_EXT MAN_OP = LMNR_IN	互连的设定值

3.4 CTRL_S: PID 步控制器块

工作原理	跟踪		手动				自动					
源	LMN_TRK	LMNUP/ LMNDN	MAN_OP		LMNUP_OP/ LMNDN_OP		SP_EXT		SP_OP		PV_IN	
内部/外部	外部	外部	外部	内部	外部	内部	外部	内部	外部	内部	外部	内部
控制输入:												
AUT_L	-	-	0	-	0	-	1	-	1	-	1	-
AUT_ON_OP	-	-	-	0	-	0	-	1	-	1	-	1
AUTOP_EN	-	-	-	-	-	-	-	(1)	-	(1)	-	-
MANOP_EN	-	-	-	(1)	-	(1)	-	-	-	-	-	-
LIOP_MAN_SEL	-	-	1	0	1	0	1	0	1	0	1	0
SPEXON_L	-	-	-	-	-	-	1	-	0	-	-	-
SPEXTSEL_OP	-	-	-	-	-	-	-	1	-	0	-	-
SPINT_EN	-	-	-	-	-	-	-	-	-	(1)	-	-
SPEXT_EN	-	-	-	-	-	-	-	(1)	-	-	-	-
LIOP_INT_SEL	-	-	-	-	-	-	1	0	1	0	-	-
LMN_SEL	1	-	0	0	0	0	0	0	0	0	0	0
LMNS_ON	0	1	0	0	0	0	0	0	0	0	0	0
SP_OP_ON	-	-	-	-	-	-	-	-	1	1	-	-
LMNOP_ON	-	-	1	1	-	-	-	-	-	-	-	-
LMNSOPON	-	-	(2)	(2)	1	1	-	-	-	-	-	-
LMNR_ON	1	-	1	1	-	-	-	-	-	-	-	-
SP_TRK_ON	-	-	-	-	-	-	0	0	0	0	1	1

说明:

外部: 设定值由程序控制, 通过互连的输入或组态来设置。

内部: 通过 OS 上的操作员控制来设置设定值。

"-": 随机状态

(1): 仅当 OS 上存在切换时才检查该设置。

(2): 不能激活控制信号的操作员控制。(= not (LMNSOPON and (LMNUP_OP xor LMNDN_OP))) 通过 LMNUP_OP 或 LMNDN_OP 的控制信号的操作员控制优先于通过 MAN_OP 的调节变量的操作员控制。

PID 算法的模拟调节变量按以下方式生成:

$$LMN_{unlimited} = GAIN * \left(1 + \frac{1}{TN * s} + \frac{TV * s}{1 + TM_LAG * s} \right) * ER_{normalized}$$

s: 复杂编号

前馈控制和限制

在自动模式中, 将干扰变量 DISV 加到 PID 算法的输出上。结果限制在 LMN_LLM 到 LMN_HLM 范围之间。

带阈值调整的三步幅元件

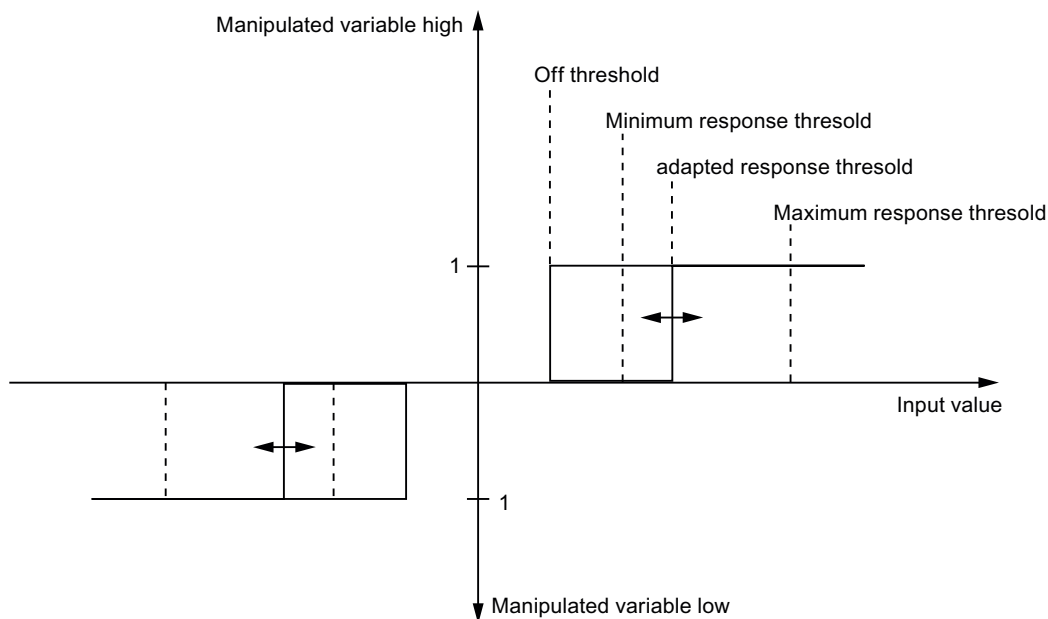
三点元件有开、关阈值, 可通过块进行调整以得到一个介于阈值之间的值。

说明

此公式描述了一种标准情况, 其中激活了 P、I 和 D 动作, 并且 P 和 D 动作不在反馈路径中。(P_SEL = TRUE, TN <> 0, PFDB_SEL = FALSE 且 DFDB_SEL = FALSE)。

例如, 如果 TN = 0, 则加上从物理变量 LMN_OFF (运行点) 计算出的偏移量。更多相关信息, 可参考“CTRL_PID 的方框图 (页 105)”。

3.4 CTRL_S: PID 步控制器块



关阈值: $55.0 / MTR_TM * SAMPLE_T$

最小响应阈值: $100.0 / MTR_TM * \text{Maximum}(PULSE_TM, SAMPLE_T)$

最大响应阈值: 10.0

为了减少在补偿较大误差时的切换频率，在运行期间将自动调整响应阈值。

通过设置 `THRADA_ON = FALSE` 可关闭响应阈值的调整。这会将响应阈值设置为最小值。

当前的有效响应阈值可在 `THRESON` 输出中监视。

在以下情况下（且 `THRADA = TRUE`）将响应阈值设置为最小值：

- 无位置反馈的步控制器的手动或跟踪模式
- 控制器 I 动作 ($TN = 0\text{ s}$)

3.4.4 CTRL_S 的手动、自动和跟踪模式及级联

手动模式

手动模式提供了三种影响控制信号的方式:

- 通过 MAN_OP 进行调节变量输入
- MAN_OP 的点动模式
- 使用控制命令直接激活控制信号

对于通过调节变量输入或点动进行的 MAN_OP 操作, 仅在具有位置反馈 (LMNR_ON = 1) 的控制系统中可行。通过 OP_A_LIM (页 554) 对其进行设置和限制 (范围是 MAN_HLM - MAN_LLM)。OP_A_LIM 的 QVHL 和 QVLL 输出值被传递到 QLMN_HLM 和 QLMN_LLM 输出中。在输出 LMN 中设置 MAN_OP 的值, 并且通过控制信号驱动电机, 直到位置反馈 LMNR_IN 的值达到 MAN_OP 的值。

但是, 无论是否通过位置反馈进行操作, 在控制系统中都可以通过执行命令直接操作控制信号。通过 LMNSOPON 启用信号操纵; 在 LMNUP_OP 或 LMNDN_OP 处设置控制信号。取消操作员输入或到达端位置时, 才会调整阀。

控制信号输入优先于调节变量输入。如果 MAN_OP 不是有效输入信号, 则跟踪该值直到 LMNR_IN。

自动模式

PID 算法计算调节变量 LMN。默认情况下, 无法互连控制参数 GAIN、TN、TV 和 TM_LAG。为了在特殊情况下互连这些参数 (例如, 为了增益计划), 必须更改相应的系统属性“s7_link”。请注意, 如果自动模式处于活动状态, 则参数更改可能导致操作变量的步进变化:

- 通过设置负比例 GAIN 值, 可颠倒控制动作的有效方向 (上升误差信号将导致调节变量下降)。通过设置 P_SEL = 0 可禁用 P 动作, 通过设置 TN(TI) = 0 可禁用 I 动作。对于带位置反馈的控制系统, 可通过互连输入 INTH_POS 或 INTH_NEG 针对某个特定方向禁用 I 动作。
- 在防终止措施的范围内, 当限位开关超出 LMNR_HS 或 LMNR_LS, 或者位置反馈 LMNR_IN 处于活动状态并且不在 LMN_HLM、LMN_LLM 范围时, 会自动停止积分器。
- 运行点 (输入 LMN_OFF): 如果禁用 I 动作, 则在自动模式下此值替换 PID 算法的 I 动作。运行点按照调节变量的测量范围进行输入。

- D 动作作为带延迟的微分元件来实现。通过设置 $TV(TD) = 0$ 可将其禁用。时延常量 TM_LAG 应与微分时间 $TV(TD)$ 具有适当的关系。此比率也称作微分增益 (D 动作的最大单步响应), 其通常位于 $5 < TV(TD)/TM_LAG < 10$ 范围中。
- 在闭环控制器的反馈路径中激活 P 动作: 通过设置 $PFDB_SEL = TRUE$ 在闭环控制器的反馈路径中启用 P 动作。控制值步进变化不影响 P 动作, 所以在不更改干扰特征的情况下可以减少或避免设定值步进变化所导致的过冲。在自动模式下切换 $PFDB_SEL$ 会导致调节变量的极端变化。因此, 建议在手动模式下更改 $PFDB_SEL$ 。
- 在闭环控制器的反馈路径中启用 D 动作: 通过设置 $DFDB_SEL = TRUE$ 在闭环控制器的反馈路径中启用 D 作用。在这种情况下, 控制步进变化不影响 D 动作。 $DFDB_SEL$ 的切换不是无波动的。

计算的调节变量转换成一系列执行脉冲。以下参数会影响到用于生成这些执行脉冲的算法:

- MTR_TM : 电机运行时间 = 覆盖阀的最大行程距离所需的时间。
- $PULSE_TM$: 最短脉冲持续时间; 阀可移动的最短步距离是 $100\% * PULSE_TM / MTR_TM$ 。
- $BREAK_TM$: 最短中断时间; 某执行脉冲终止后, 必须经过这一段时间间隔才能输出新脉冲。
- $LMNR_HS$ 、 $LMNR_LS$: 限位开关; 如果设置了一个限位开关, 则禁用相应的输出信号 $QLMNUP$ 或 $QLMNDN$ 。
- 电机保护信号 MSS 的负沿将电机保护故障信号设置为保持状态, 并将此信号传递给输出 $QMSS_ST$ 。参数 MSS_SIG 用于定义是仅显示错误消息 ($MSS_SIG = 0$), 还是不管其它所有输入和系统状态为何都要限制电机 ($MSS_SIG = 1$)。电机保护错误 ($QMSS_ST = 1$) 将报告给 OS 。 $QMSS_ST$ 既可以由操作员在 $RESET$ 输入上复位, 也可以通过将 L_RESET 与“1”互连, 在处于 MSS 的正沿时自动进行复位。
- $LMNR_ON$: 如果设置了 MNR_ON , 则通过位置反馈操作闭环控制。系统处于运行状态时, 不得更改此控制输入的状态。
- $DEADB_W$: 由于在零周围存在一些小的误差信号波动, 所以为了减小控制器的切换频率, 误差信号中的死区是非常必要的。在闭环的运行点处, 以下值会更改误差信号: $(100\% * PULSE_TM / MTR_TM) * \text{过程增益}$ 。因此, 死区应大于此值的 50%。如果 $DEADB_W$ 假定为负值, 则会忽略死区。

跟踪模式

在跟踪模式下可通过如下两种方式影响执行信号:

- 通过外部调节变量 LMN_TRK 跟踪
- 通过互连的输入 LMNS_ON、LMNUP 和 LMNDN 直接控制执行信号

对于使用外部调节变量 LMN_TRK 的跟踪模式, 仅在具有位置反馈的控制系统中可行。如果 LMN_SEL = 1, 则调节变量从互连的跟踪值 LMN_TRK 中获取并且应用到输出 LMN。通过执行信号调整阀, 直到位置反馈 LMNR_IN 的值达到 LMN_TRK 的值。

使用 LMNS_ON 且通过互连的 LMNUP 和 LMNDN 输入直接连接控制信号的跟踪模式在所有模式中具有最高优先级。如果设置了 LMNS_ON, 则只能通过 LMNUP 或 LMNDN 输入设置执行信号。只要设置了 LMNS_ON, 便可抑制对控制信号的任何其它影响。

将 D 和 P 动作置于反馈路径中

P 和 D 动作可位于反馈路径中, 以降低或避免因设定值步进变化引起的过程值的过冲。在此模式下设定值步进变化不再对 P 或 D 动作产生任何影响。调节值不再对设定值步进变化后的步进变化做出响应。

通过设置 PFDB_SEL = 1, 在反馈路径中启用 P 动作。无位置反馈的步控制器会忽略 PFDB_SEL。在此模式下, 在反馈路径中不能启用 P 动作。

通过设置 DFDB_SEL = 1, 在反馈路径中启用 D 动作。

级联 PID 控制器

主控制器的调节变量 LMN 与二级控制器的输入 SP_EXT 相连。(注意: 二级控制器是步控制器, 而主控制器不是步控制器!)。如果打开级联, 则必须将主控制器也更改为跟踪模式。QCAS_CUT 信号在二级控制器中生成, 并且与主控制器的 LMN_SEL 输入互连。通过设定值输入, 或通过二级控制器的设定值跟踪可以在手动或跟踪模式下将此互连断开:

QCAS_CUT = LMNS_ON or LMN_SEL or (not QMAN_AUT) or (QMAN_AUT and SP_TRK_ON)

为防止级联再次闭合时发生步进变化, 将一级控制器的跟踪输入 LMN_TRK 与二级控制器的 SP 输出互连。

同时建议设置只要达到调节变量限制就激活的主控制器中的积分器的方向锁定。通过正控制动作, 将主控制器的 INT_HPOS 和 INT_HNEG 输入与二级控制器的 QLMN_HLM 和 QLMN_LLM 输出互连。

3.4.5 CTRL_S 模式更改

模式更改

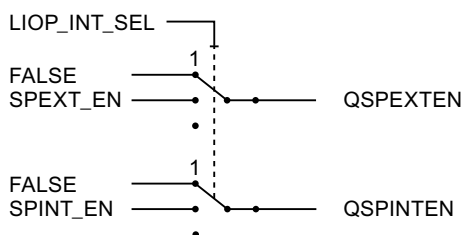
模式更改可由操作员或通过互连的输入触发。

外部/内部设定值

根据 LIOP_INT_SEL 选择输入的设置，通过 SPEXTSEL_OP 输入的 OS 操作或 SPEXON_L 的互连执行外部和内部设定值之间的切换。必须设置相应的使能输入 SPINT_EN、SPEXT_EN 来启用此切换。

如果 SPBUMPON = 1，则用有效设定值来代替内部设定值，从而实现从外部或跟踪模式到内部模式的无波动切换。

启用内部和外部设定值之间的切换



QSPEXTEN = TRUE: 可将 SPEXTSEL_OP 从 FALSE (内部设定值) 更改为 TRUE (外部设定值)。

QSPINTEN = TRUE: 可将 SPEXTSEL_OP 从 TRUE (外部设定值) 更改为 FALSE (内部设定值)。

根据需要对 SPEXTSEL_OP 进行跟踪或复位。

通过操作员输入 SP_OP_ON 启用设定值控制

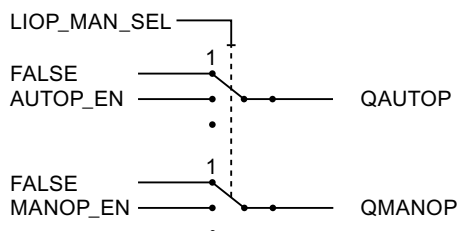
Q_SP_OP = TRUE: SP_OP 操作启用。

根据需要对 SP_OP 进行跟踪或复位。

手动/自动

根据 LIOP_MAN_SEL 选择输入的设置，通过 AUT_ON_OP 输入的 OS 操作或 AUT_L 的互连执行手动和自动模式之间的切换。必须设置相应的使能输入 MANOP_EN、AUTOP_EN 来启用此切换。

启用手动和自动模式之间的切换



QAUTOP = TRUE: 可将 AUT_ON_OP 从 FALSE (手动模式) 更改为 TRUE (自动模式)。

QMANOP = TRUE: 可将 AUT_ON_OP 从 TRUE (自动模式) 更改为 FALSE (手动模式)。

根据需要对 AUT_ON_OP 进行跟踪或复位。

启用手动模式

通过设置 QLMNVOP, 可在面板上通过 MAN_OP 来启用操纵变量操作, 通过 QLMNSOP 指令对控制信号进行控制:

	QLMNVOP	QLMNSOP
LMNOP_ON	1	-
LMNSOPON	-	1
LMNR_ON	1	-
LMNSOPON and (LMNUP_OP xor LMNDN_OP)	0	-
LMN_SEL	0	0
LMNS_ON	0	0
QMAN_AUT	0	0

3.4 CTRL_S: PID 步控制器块

为确保无波动切换，针对下列模式采取了特殊措施：

- 外部设定值/跟踪设定值： 如果 SPBUMPON = TRUE，内部设定值 SP_OP 将设置为有效（外部或已跟踪）设定值。
- 自动模式： 将跟踪手动值 MAN_OP 至有效操纵变量 LMNR_IN 的值。
- 跟踪模式： 当设置了 LMN_SEL 时，将跟踪手动输入值 MAN_OP 至 LMN_TRK 的值。这样，您可以看出阀最大可以行进到哪个值。取消 LMN_SEL 后，MAN_OP 将复位为 LMNR_IN 的值，以便确保设置 MAN_OP 时的无波动切换。
- 手动模式： 在带有位置反馈的控制系统中，跟踪积分器以便无波动切换至自动模式。
积分分量 = 操纵变量（百分比） -
比例分量 - 干扰变量（百分比） 微分分量被禁用，并得到补偿。

3.4.6 CTRL_S 的错误处理

操作员输入错误

在 SPEXTSEL_OP、AUT_ON_OP、SP_OP 或 MAN_OP 参数中，如果在更改其中某个参数时至少出现了一个操作员错误，将设置 QOP_ERR = 1。否则，设置 QOP_ERR = 0。操作员错误仅存在一个周期。

块算法处理以下情况：

- 参数分配错误：NM_PVHR \leq NM_PVLR：
误差信号 ER 设为零。设置 ENO = 0 或 QERR = 1。
- 参数分配错误：SAMPLE_T $<$ 0.001：
将采样时间 SAMPLE_T 设为 0.001。设置 ENO = 0 或 QERR = 1。
- 参数分配错误：GAIN = 0：
误差信号 ER 设为零。设置 ENO = 0 或 QERR = 1。
- TN(TI) $<$ SAMPLE_T/2：
如果 TN $>$ 0，则使用 TN = SAMPLE_T/2 进行计算。如果 TN(TI) = 0，积分器被禁用，同时运行点 LMN_OFF 处于活动状态。
- TV(TD) $<$ SAMPLE_T：
如果 TV $>$ 0，则使用 TV = SAMPLE_T 进行计算。如果 TV(TD) = 0，则禁用 D 动作。
- TM_LAG $<$ SAMPLE_T/2：
当 TM_LAG $<$ SAMPLE_T/2 时，使用 TM_LAG $<$ SAMPLE_T/2 进行计算。在此类情况下，D 动作充当理想微分器。

将 MTR_TM、PULSE_TM 和 BREAK_TM 向下限制为 SAMPLE_T 的值。

在 QMAN_ERR 或 QAUT_ERR 输出上会指出在激活操作期间复位了 *_EN 使能输入。

如果在控制器跟踪 LMN_TRK 时取消位置反馈 LMNR_ON 的控制输入，则还要设置 QMAN_ERR。这时，阀停止。

3.4.7 CTRL_S 的启动行为、动态响应和消息响应

启动特征

在 CPU 启动期间，使用内部设定值将 CTRL_S 设置为手动模式。这意味着必须在启动 OB 中调用该块。在 CFC 工程组态中，这由 CFC 处理。使用基本 STEP 7 工具时，在启动 OB 中输入此调用信息。

启动之后，将在 RUNUPCYC 所设置的周期数内抑制消息。

启动期间，初始化 MAN_OP、LMN 以及 LMNR_IN。积分动作设置为零。

时间响应

块必须在循环中断 OB 中调用。块的采样时间在 SAMPLE_T 参数中设置。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“CTRL_S 的 VSTATUS (页 116)”。

消息响应

CTRL_S 块使用 ALARM_8P 块生成消息。

存在下列消息触发器：

- 用于监视过程值和系统偏差限制的功能
- 通过互连作为控制系统错误而接收的 CSF 信号 QMSS_ST

通过使用相关的 M_SUP_xx 输入可以分别抑制超出限制消息。可使用 MSG_LOCK 集中禁用过程消息（不是控制系统消息！）。

如果重新启动以来未满 RUNUPCYC 个周期，并且 MSG_LOCK = TRUE 或 MSG_STAT = 21，则会设置 QMSG_SUP。

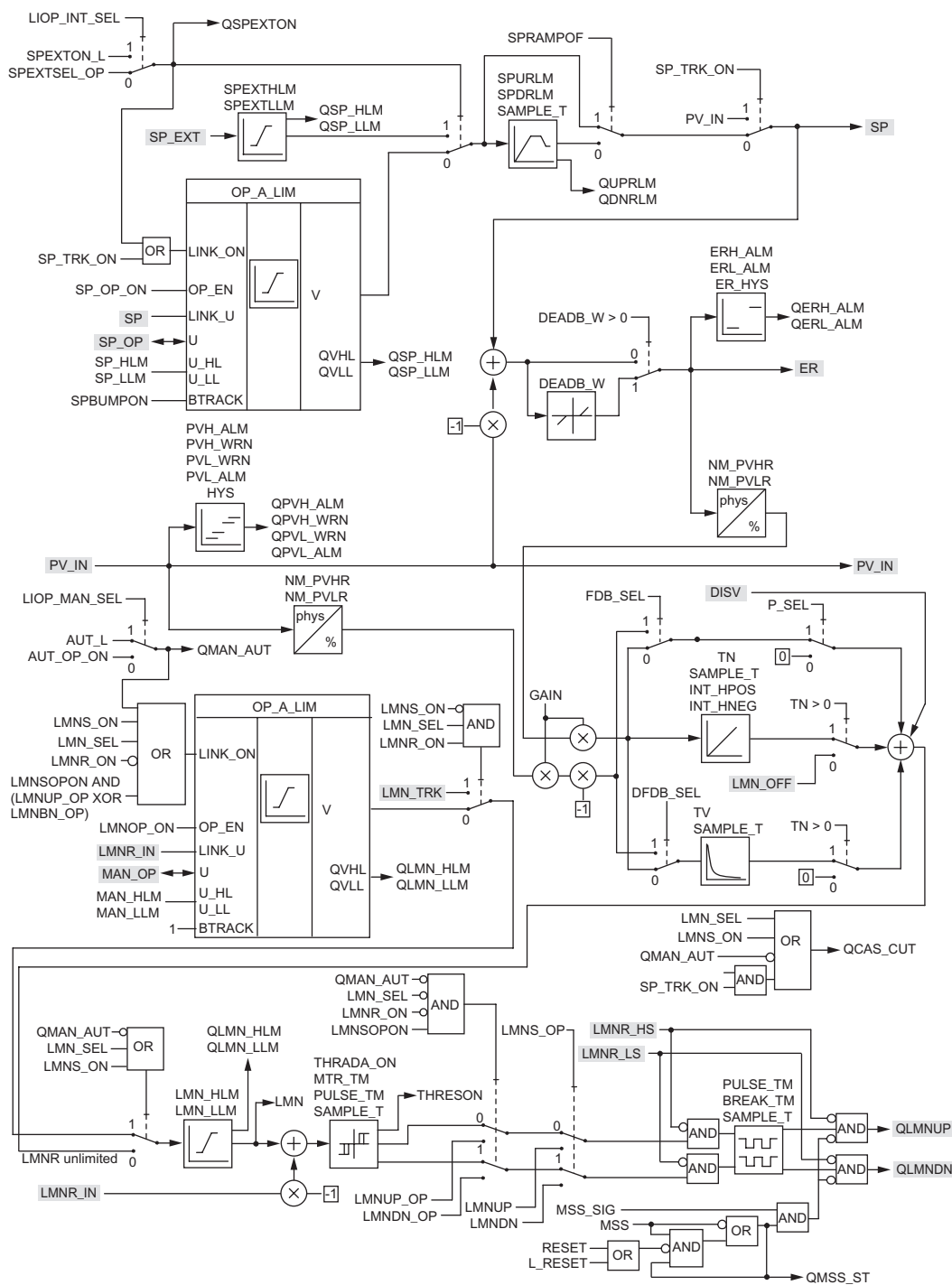
下表列出了 CTRL_S 块的消息文本 (页 114)及其到块参数的分配。

监视过程值

不可用

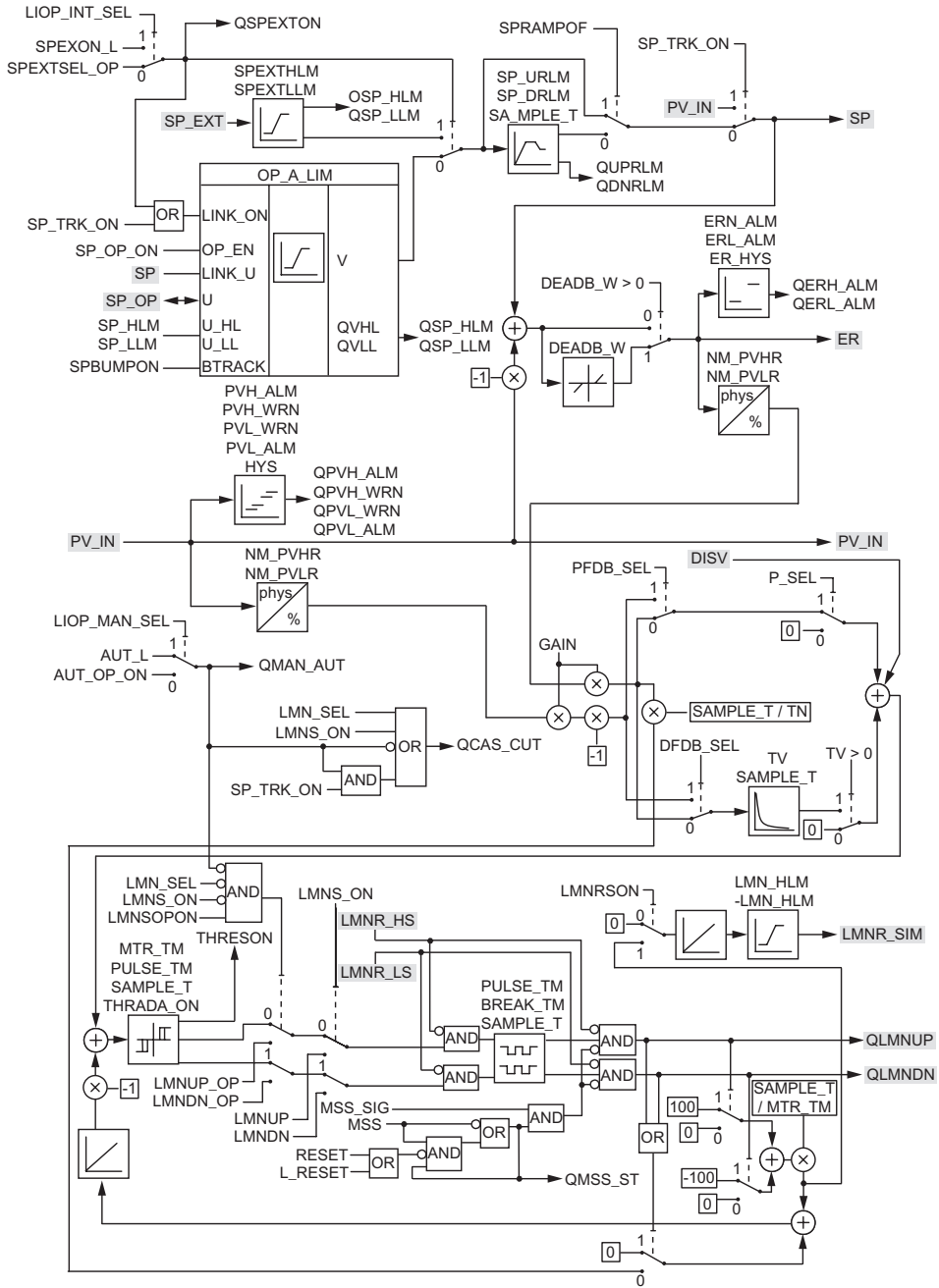
3.4.8 CTRL_S 的方框图

带位置反馈的 CTRL_S 的方框图



3.4 CTRL_S: PID 步控制器块

不带位置反馈的 CTRL_S 的方框图



3.4.9 CTRL_S 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认 值	类 型	OCM	允许 值
AUT_L	用于 MANUAL (手动) /AUTO (自动) 模式的互连输入: 0 = 手动; 1 = 自动	BOOL	0	I		
AUT_ON_OP	操作员输入: 0 = 手动; 1 = 自动	BOOL	0	IO	+	
AUTOP_EN	1 = 自动模式的操作员控制使能信号	BOOL	1	I		
AUX_PRx	关联值 x	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 连续批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	"	I	+	
BREAK_TM	最短中断时间 (秒)	REAL	1	I	+	
CSF	控制系统错误	BOOL	0	I		
DEADB_W	死区	REAL	0	I	+	≥ 0
DFDB_SEL	将 D 动作置于反馈中, 1 = 激活	BOOL	0	I		
DISV	干扰变量	REAL	0	I		
ER	误差信号	REAL	0	O	+	
ER_HYS	用于监视误差信号的滞后量	REAL	0.1	I	+	≥ 0
ERH_ALM	误差信号: 报警上限	REAL	100	I	+	> 0
ERL_ALM	误差信号: 报警下限	REAL	-100	I	+	< 0
GAIN	比例增益	REAL	1	I	+	
HYS	滞后量	REAL	5	I	+	≥ 0
INT_HNEG	根据方向锁定, 负沿	BOOL	0	I		
INT_HPOS	根据方向锁定, 正沿	BOOL	0	I		

3.4 CTRL_S: PID 步控制器块

I/O (参数)	含义	数据类型	默认 值	类 型	OCM	允许 值
L_RESET	电机保护错误 (QMSS_ST=0) 的可互连输入 RESET	BOOL	0	I		
LIOP_INT_SE L	1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
LIOP_MAN_S EL	1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
LMN	调节变量输出	REAL	0	O		
LMN_HLM	调节变量上限	REAL	100	I	+	LMN_HLM > LMN_LLM
LMN_LLM	调节变量下限	REAL	0	I	+	LMN_LLM < LMN_HLM
LMN_OFF	运行点	REAL	0	I	+	
LMN_SEL	1 = 外部调节变量激活	BOOL	0	I		
LMN_TRK	外部调节变量	REAL	0	I		
LMNDN	已互连的控制信号低	BOOL	0	I		
LMNDN_OP	可调整的控制信号低	BOOL	0	IO	+	
LMNOP_ON	1 = 调节变量 MAN_OP 的操作员控制使能信号	BOOL	1	I		
LMNR_HS	位置反馈的上限位停止信号	BOOL	0	I		
LMNR_IN	用于在 OS 上显示的位置反馈	REAL	0	I		
LMNR_LS	位置反馈的下限位停止信号	BOOL	0	I		
LMNR_ON	通过位置反馈控制	BOOL	0	I		
LMNR_SIM	模拟位置反馈	REAL	0	O	+	
LMNRSON	激活位置反馈仿真	BOOL	0	I	+	
LMNS_ON	互连的控制信号开启 (LMNDN、LMNUP)	BOOL	0	I		
LMNSOPON	激活控制信号的操作员控制	BOOL	1	I		
LMNUP	已互连的控制信号高	BOOL	0	I		
LMNUP_OP	可调整的控制信号高	BOOL	0	IO	+	

I/O (参数)	含义	数据类型	默认 值	类 型	OCM	允许 值
M_SUP_AH	1 = 消息抑制报警上限过程值	BOOL	0	I	+	
M_SUP_AL	1 = 消息抑制报警下限过程值	BOOL	0	I	+	
M_SUP_ER	抑制误差信号报警消息	BOOL	1	I	+	
M_SUP_WH	1 = 消息抑制警告上限过程值	BOOL	0	I	+	
M_SUP_WL	1 = 消息抑制警告下限过程值	BOOL	0	I	+	
MAN_HLM	调节变量手动值上限	REAL	100	I	+	
MAN_LLM	调节变量手动值下限	REAL	0	I	+	
MAN_OP	调节变量的操作员输入	REAL	0	IO	+	
MANOP_EN	1 = 手动模式的操作员控制使能信号	BOOL	1	I		
MO_PVHR	显示上限 (测量范围)	REAL	110	I	+	
MO_PVLR	显示下限 (测量范围)	REAL	-10	I	+	
MSG_ACK	确认消息	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		
MSG_LOCK	1 = 消息抑制取决于过程状态	BOOL	0	I	+	
MSG_STAT	消息错误信息	WORD	0	O		
MSS	电机保护低态有效	BOOL	1	I		
MSS_SIG	电机保护激活时复位控制信号	BOOL	0	I		
MTR_TM	电机运行时间 (秒)	REAL	60	I	+	
NM_PVHR	过程值标准化上限 (测量范围)	REAL	100	I		
NM_PVLR	过程值标准化下限 (测量范围)	REAL	0	I		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
OPTI_EN	1 = 启用控制器调节	BOOL	0	I	+	
P_SEL	激活 P 动作, 1 = 激活	BOOL	1	I		
PFDB_SEL	将 P 动作置于反馈中; 1 = 激活	BOOL	0	I		
PULSE_TM	最短脉冲持续时间 (秒)	REAL	1	I	+	
PV_IN	过程值	REAL	0	IO	+	

3.4 CTRL_S: PID 步控制器块

I/O (参数)	含义	数据类型	默认 值	类 型	OCM	允许 值
PVH_ALM	过程值: 报警上限	REAL	100	I	+	$PVH_ALM \geq PVL_ALM$
PVH_WRN	过程值: 警告上限	REAL	95	I	+	$PVH_WRN \leq PVL_WRN$
PVL_ALM	过程值: 报警下限	REAL	0	I	+	$PVL_ALM \leq PVH_ALM$
PVL_WRN	过程值: 警告下限	REAL	5	I	+	$PVL_WRN \leq PVH_WRN$
Q_SP_OP	1 = 设置设定值的操作员控制使能信号	BOOL	0	O	+	
QAUT_ERR	缺少自动模式的启用信号	BOOL	0	O		
QAUTOP	1 = 自动模式的操作员控制使能信号	BOOL	0	O	+	
QC_LMN	LMN 的质量代码	BYTE	16#80	O		
QC_LMN_I	输出 LMN 的质量代码	BYTE	16#80	I		
QC_LMNR_IN	LMNR_IN 的质量代码	BYTE	16#80	I		
QC_PV_IN	PV_IN 的质量代码	BYTE	16#80	I		
QCAS_CUT	1 = 级联已切断	BOOL	1	O		
QDNRLM	1 = 限制负设定值斜坡	BOOL	0	O		
QERH_ALM	误差信号, 1 = 报警上限	BOOL	0	O	+	
QERL_ALM	误差信号, 1 = 报警下限	BOOL	0	O	+	
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	
QLMN_HLM	1 = 限制调节变量输出上限值	BOOL	0	O		
QLMN_LLM	1 = 限制调节变量输出下限值	BOOL	0	O		
QLMN_SEL	1 = 跟踪 LMN_TRK 激活	BOOL	0	O	+	
QLMNDN	控制信号下限	BOOL	0	O	+	
QLMNOP	1 = 调节变量或控制信号的操作员控制使能信号	BOOL	0	O	+	
QLMNR_HS	已设置位置反馈的上限位停止信号	BOOL	0	O	+	
QLMNR_LS	已设置位置反馈的下限位停止信号	BOOL	0	O	+	

I/O (参数)	含义	数据类型	默认 值	类 型	OCM	允许 值
QLMNR_ON	位置反馈已激活	BOOL	0	O	+	
QLMNRSON	位置反馈仿真已激活	BOOL	0	O	+	
QLMNS_ON	1 = 信号调整激活	BOOL	0	O	+	
QLMNSOP	输出: 控制信号的操作员控制已激活	BOOL	1	O	+	
QLMNUP	控制信号高	BOOL	0	O	+	
QLMNVOP	输出: 调节变量的操作员控制已激活	BOOL	1	O	+	
QMAN_AUT	0 = 手动, 1 = 自动	BOOL	0	O	+	
QMAN_ERR	缺少手动模式的使能信号	BOOL	0	O		
QMANOP	1 = 手动模式的操作员控制使能信号	BOOL	0	O	+	
QMAN_ERR	Alarm_8P 的错误	BOOL	0	O		
QMSG_SUP	1 = 消息抑制激活	BOOL	0	O	+	
QMSS_ST	发电机保护 激活 (0 = 使用 RESET 复位)	BOOL	0	O	+	
QOP_ERR	1 = 组操作员输入错误	BOOL	0	O		
QPVH_ALM	1 = 报警上限	BOOL	0	O		
QPVH_WRN	1 = 警告上限	BOOL	0	O		
QPVL_ALM	1 = 报警下限	BOOL	0	O		
QPVL_WRN	1 = 警告下限	BOOL	0	O		
QSP_HLM	1 = 限制设定值输出上限	BOOL	0	O		
QSP_LLM	1 = 限制设定值输出下限	BOOL	0	O		
QSPEXTEN	1 = 外部模式的操作员控制使能信号	BOOL	0	O	+	
QSPEXTON	0 = 内部 1 = 外部	BOOL	0	O	+	
QSPINTEN	1 = 内部模式的操作员控制使能信号	BOOL	0	O	+	
QUPRLM	1 = 限制正设定值斜坡	BOOL	0	O		
RESET	已启用电机保护错误的复位输入 (QMSS_ST=0)	BOOL	0	IO	+	
RUNUPCYC	起动周期数	INT	3	I		

3.4 CTRL_S: PID 步控制器块

I/O (参数)	含义	数据类型	默认 值	类 型	OCM	允许 值
SAMPLE_T	采样时间 (秒)	REAL	1	I		≥ 0.001
SP	激活设定值	REAL	0	O	+	
SP_EXT	外部设定值	REAL	0	I		
SP_HLM	设定值上限	REAL	100	I	+	SP_HLM > SP_LLM
SP_LLM	设定值下限	REAL	0	I	+	SP_LLM < SP_HLM
SP_OP	设定值的操作员输入	REAL	0	IO	+	
SP_OP_ON	设定值 SP_OP 的操作员控制使能信号	BOOL	1	I		
SP_TRK_ON	1 = 跟踪设定值 SP_OP	BOOL	0	I	+	
SPBUMPON	1 = 无波动设定值	BOOL	1	I	+	
SPDRLM	最大负设定值斜率 [1/s]	REAL	100	I	+	
SPEXON_L	内部/外部设定值的 可互连输入 (0 = 内部/1 = 外部)	BOOL	0	I		
SPEXT_EN	1 = 外部设定值的操作员控制使能信号	BOOL	1	I		
SPEXTHLM	外部设定值上限	REAL	100	I		SPEXTHLM > SPEXTLLM
SPEXTLLM	外部设定值下限	REAL	0	I		SPEXTLLM < SPEXTHLM
SPEXTSEL_O P	操作员输入: 0 = 内部, 1 = 外部	BOOL	0	IO	+	
SPINT_EN	1 = 内部模式的操作员控制使能信号	BOOL	1	I		
SPRAMPOF	1 = 禁用设定值斜坡限制	BOOL	1	I	+	
SPURLM	最大正设定值斜率 [1/s]	REAL	100	I	+	
STEP_NO	BATCH 步号	DWORD	0	I	+	
THRADA_ON	调整响应阈值: 0 = 保持常量	BOOL	1	I		

I/O (参数)	含义	数据类型	默认 值	类 型	OCM	允许 值
THRESO	调整阈值	REAL	0.0	O		
TM_LAG	D 动作的时延 (秒)	REAL	1	I	+	\geq SAMPLE_T/2
TN	复位时间 (秒)	REAL	10	I	+	TN=0, \geq SAMPLE_T/2
TV	微分时间 (秒)	REAL	0	I	+	TV=0, \geq SAMPLE_T
USTATUS	VSTATUS 中的状态字, 可进行用户特定的组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.4.10 CTRL_S 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由此项抑制
1	QPVH_ALM (在 PV_IN ≥ PVH_ALM 时)	PV:\$\$BlockComment\$\$ 上上限报警	AH	M_SUP_AH、 MSG_LOCK
2	QPVH_WRN (在 PV_IN ≥ PVH_WRN 时)	PV:\$\$BlockComment\$\$ 上限报警	WH	M_SUP_WH、 MSG_LOCK
3	QPVL_WRN (在 PV_IN ≤ PVL_WRN 时)	PV:\$\$BlockComment\$\$ 下限报警	WL	M_SUP_WL、 MSG_LOCK
4	QPVL_ALM (在 PV_IN ≤ PVL_ALM 时)	PV:\$\$BlockComment\$\$ 下下限报警	AL	M_SUP_AL、 MSG_LOCK
5	CSF	\$\$BlockComment\$\$ 外部故障	S	-
6	QERH_ALM (在 ER ≥ ERH_ALM 时)	ER:\$\$BlockComment\$\$ 上上限报警	AH	M_SUP_ER、 MSG_LOCK
7	QERH_ALM (在 ER ≤ ERL_ALM 时)	ER:\$\$BlockComment\$\$ 下下限报警	AL	M_SUP_ER、 MSG_LOCK
8	QMSS_ST	\$\$BlockComment\$\$ 电机保护	S	-

将关联值分配给块参数

消息块的前三个关联值已分配给 SIMATIC BATCH 数据，第四个关联值留供 PV_IN 使用，其余关联值 (AUX_PRx) 可由用户随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	PV_IN
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.4 CTRL_S: PID 步控制器块

3.4.11 CTRL_S 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	QLMNR_ON	-	QMSS_ST	QSPEXTON	QMAN_AUT	MSG_LOCK	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	LMN_SEL	-	QLMNR_LS	QLMNR_HS	QLMNDN	QLMNUP

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.4.12 CTRL_S 的操作和监视

更多信息

更多信息，请参考以下部分：

- CTRL_S 块图标 (页 685)
- CTRL_S 面板 (页 592)

3.5 DEADT_P: 停滞时间元件

3.5.1 DEADT_P 的描述

对象名 (类型 + 编号)

FB37

- DEADT_P 块 I/O (页 119)

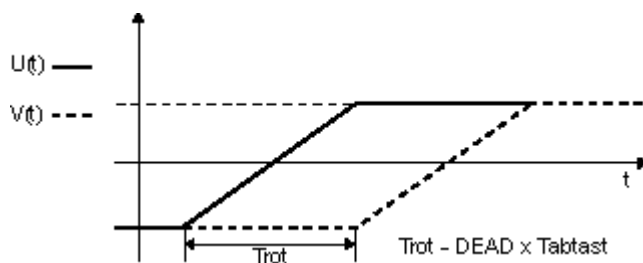
功能

经过了可定义的 DEADT 周期数后, 才会将输入 U 的模拟值传递到输出 V。下列公式适用:

$$V(t) = U(t - T_{\text{dead}}), \text{ 其中 } T_{\text{dead}} = \text{DEADT} * T_{\text{sampling}}$$

有关时间 $0 < t < T_{\text{dead}}$ 的更多信息, 可参考“启动特征”。

工作原理



DEADT_P 的工作原理

- 该块在当前周期内获取模拟输入值 U 并将其缓冲, 在经过所设置的 DEADT 周期数后, 将其输出到输出 V。缓冲值的最大数目限制为 16。更多信息, 可参考“错误处理”。
- 如果在执行该块时修改了 DEADT 参数, 则该块会像在 CPU 启动期间一样进行响应。

3.5 DEADT_P: 停滞时间元件

调用 OB

安装了该块的循环中断 OB（例如 OB32）和 OB100。

错误处理

如果参数 $DEADT < 0$ 或 $DEADT > 16$ ，则在内部使用值 $DEADT = 16$ 进行计算，并指示 $ENO = 0$ 或 $QERR = 1$ 。

启动特征

在 CPU 启动过程中或修改停滞时间参数 $DEADT$ 时，将活动输入值 U 写入内部停滞时间缓冲区中。

时间响应

为了使块能够执行所需功能，必须在循环中断 OB 中调用该块。可以根据下列公式计算停滞时间 T_{dead} ：

- $T_{dead} = DEADT * T_{sampling}$
 $T_{sampling}$ 是块的采样时间。
- 用 CFC 组态时，必须考虑较高级别运行组的块及其采样参数。

3.5.2 DEADT_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	允许值
DEADT	停滞时间 (以周期为单位)	INT	0	I	< 16
QERR	1 = 错误	BOOL	1	O	
U	输入	REAL	0	I	
V	输出	REAL	0	O	

3.6 DIF_P: 微分

3.6.1 DIF_P 的描述

对象名 (类型 + 编号)

FB38

- DIF_P 块 I/O (页 122)

功能

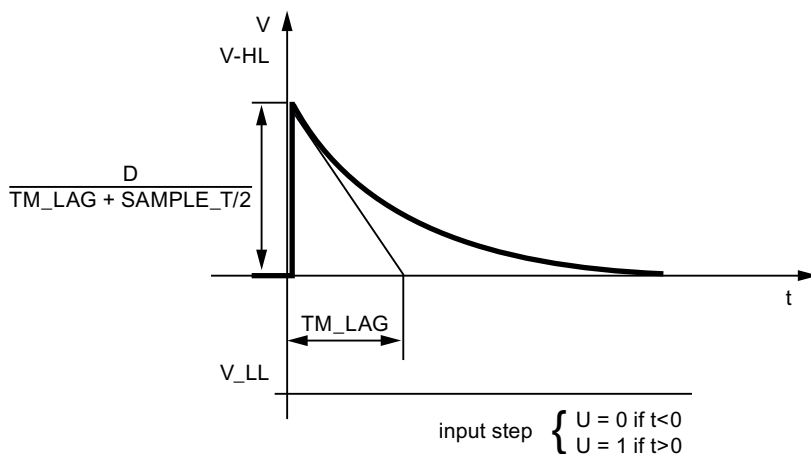
DIF_P 块近似于 DT1 动作, 它按照梯形规则操作:

$$v(s) = TD * s / (TM_LAG * s + 1) * u(s)$$

工作原理

该算法按照梯形规则运行。还执行下列步骤:

- 将输出 V 的上限限制为 V_HL, 下限限制为 V_LL。限制功能不影响内部计算。
- 当限制处于活动状态时, 设置所分配的输出 Q_HL 或 Q_LL。
- 允许参数设置 TD = 0, 结果为 V = 0 = 常数。还允许 TD 取负值, 并有将结果符号取反的作用。



DIF_P 的步响应

调用 OB

安装了该块的循环中断 OB（例如 OB32）和 OB100。

错误处理

如果出现上溢/下溢，将使用 REAL 类型超出的范围限制，并且将 ENO 设置为“0”。以下组态错误也会导致 ENO = 0 (QERR = TRUE) 和 V=0:

- $V_{LL} > 0$
- $V_{HL} < 0$

启动特征

CPU 启动后，将跟踪输入 U 旧值的内部存储器位至此输入。这将确保输出值 V 在第一次循环操作中为“0”。

时间响应

从循环中断 OB 中调用块。

3.6.2 DIF_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认 值	类 型	OCM	允许 值
QERR	1 = 错误	BOOL	1	O		
QVHL	初始值上限	BOOL	0	O		
QVLL	初始值下限	BOOL	0	O		
SAMPLE_T	采样时间 (秒)	REAL	1	I		> 0
TD	微分时间 (秒)	REAL	1	I		
TM_LAG	时延 (秒)	REAL	10	I		≥ 0
U	输入	REAL	0	I		
V	输出值	REAL	0	O		
V_HL	V 的上限	REAL	100	I		V_HL ≥ V_LL
V_LL	V 的下限	REAL	-100	I		V_LL ≥ V_HL

3.7 DIG_MON: 数字值监视

3.7.1 DIG_MON 的描述

对象名 (类型 + 编号)

FB62

- DIG_MON 块 I/O (页 126)
- DIG_MON 块图标 (页 688)
- DIG_MON 面板 (页 599)

功能

DIG_MON 块用于通过抑制波动来监视数字过程标签。信号状态和控制系统状态 (外部控制系统错误、通道错误) 均属于过程标签。MSG_CLAS 参数可用于确定过程标签的消息类别。

工作原理

该块监视输入 Ix 处数字值的变化。在输入信号的每个沿都重新触发一次定时器。经过于 SUPPTIME 中设置的空闲时间后, 输入值 I 就会被写入至输出 Q 中。

这样可确保只将其保持时间至少为 SUPPTIME 中设置的时间段的信号传递到输出。不传递以更快速度变化的信号。

当 $SUPPTIME < SAMPLE_T$ 时, 将输入信号 I 传递到输出 Q。

调用 OB

安装了该块的循环中断 OB (例如 OB 32) 和 OB100。

错误处理

如果消息类别组态无效, 则 QERR=1。在此情况下, 不输出错误消息。更多相关信息, 可参考“消息响应”。

启动特征

CPU 启动期间，保留原初始值 Q。重新启动后，开始对变化进行监视。这意味着必须在启动 OB 中调用该块。在 CFC 工程组态中，这由 CFC 处理。使用简单的 STEP 7 工具时，必须在启动 OB 中输入此调用。

启动后，消息将在 RUNUPCYC 值中设置的周期数内受到抑制。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“DIG_MON 的 VSTATUS (页 130)”。

时间响应

块必须在循环中断 OB 中调用。块的采样时间在 SAMPLE_T 参数中设置。

消息响应

DIG_MON 块使用 ALARM_8P 块 (MSG_EVID) 生成消息（1 到 8，3、4 和 7 除外）。消息 3、4 和 7 不需要确认并通过 NOTIFY_8P (MSG_EVID1) 生成。

存在下列消息触发器：

- 切换到输出信号 Q
- 通过互连作为控制系统错误而接收的 CSF 信号。

可使用 MSG_LOCK 锁定过程消息（不是控制系统消息！）。

如果重新启动以来未满 RUNUPCYC 个周期，并且 MSG_LOCK = TRUE 或 MSG_STAT = 21，则会设置 QMSG_SUP。

消息类别

通过组态 MSG_CLAS 输入，输出 Q 处的变化可通过可选的消息类别发送到 OS。

监视过程值

不可用

更多信息

更多相关信息, 可参考:

DIG_MON 的 I/O (页 126)

DIG_MON 的消息文本和关联值 (页 128)

DIG_MON 的 VSTATUS (页 130)

操作和监视 DIG_MON (页 130)

3.7.2 DIG_MON 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认 值	类型	OCM	允许 值
AUX_PRx	关联值 x 块	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 当前批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	0	I	+	
CSF	1 = 外部控制系统错误	BOOL	0	I		
I	输入信号	BOOL	0	I	+	
MSG_ACK	确认消息	WORD	0	O		
MSG_CLAS	信号的消息类别	INT	0	I		1 - 7
MSG_EVID	消息号	DWORD	0	I		
MSG_LOCK	1 = 消息抑制取决于过程状态	BOOL	0	I	+	
MSG_STAT	消息错误信息	WORD	0	O		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
Q	输出信号	BOOL	0	O	+	
QC_I	I 的质量代码	BYTE	16#80	I		
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	
QMSG_ERR	1 = 消息错误	BOOL	0	O	+	
QMSG_SUP	1 = 消息抑制	BOOL	0	O	+	
RUNUPCYC	起动周期数	INT	3	I		
SAMPLE_T	采样时间 (秒)	REAL	1.0	I		> 0
STEP_NO	BATCH 步号	DWORD	0	I	+	
SUPPTIME	在将输入处的沿跳变传递到输出之前所经过的时间 (秒)。	REAL	0	I	+	

I/O (参数)	含义	数据类型	默认 值	类型	OCM	允许 值
USTATUS	VSTATUS 中的状态字, 可进行用户特定的组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.7.3 DIG_MON 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号 MSG_EVID	块参数	默认消息文本	消息类别	可由此项抑制
1	Q & MSG_CLAS = 1	\$\$BlockComment\$\$ 上上限报警	AH	MSG_LOCK
2	Q & MSG_CLAS = 2	\$\$BlockComment\$\$ 上限报警	WH	MSG_LOCK
3			无消息	
4			无消息	
5	Q & MSG_CLAS = 5	\$\$BlockComment\$\$ 下限报警	WL	MSG_LOCK
6	Q & MSG_CLAS = 6	\$\$BlockComment\$\$ 下下限报警	AL	MSG_LOCK
7			无消息	
8	CSF	\$\$BlockComment\$\$ 外部故障	S	-
消息号 MSG_EVID1				
1			无消息	
2			无消息	
3	Q & MSG_CLAS = 3	\$\$BlockComment\$\$ 容差上限	TH	MSG_LOCK
4	Q & MSG_CLAS = 4	\$\$BlockComment\$\$ 容差下限	TL	MSG_LOCK
5			无消息	
6			无消息	
7	Q & MSG_CLAS = 7	\$\$BlockComment\$\$ 操作员请求	OR	MSG_LOCK
8			无消息	

将关联值分配给块参数

消息块的前三个关联值已分配给 SIMATIC BATCH 数据，其余的关联值 (AUX_PRx) 可由用户随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	AUX_PR04
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.7.4 DIG_MON 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块对低 16 位（0 - 15 位）的使用如下所示：

位号:	7	6	5	4	3	2	1	0
参数	I	-	-	-	-	MSG_LOCK	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	-	-	-	-	-	Q

16 位输入 USTATUS（数据类型 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.7.5 DIG_MON 的操作和监视

更多信息

更多信息，请参考以下部分：

- DIG_MON 块图标 (页 688)
- DIG_MON 面板 (页 599)

3.8 DOSE: 定量给料过程

3.8.1 DOSE 的描述

对象名 (类型 + 编号)

FB63

- DOSE 块 I/O (页 136)
- DOSE 块图标 (页 689)
- DOSE 面板 (页 600)

功能

DOSE 块用于在使用称重设备的单组件定量给料以及使用体积测量设备进行的定量给料中批生产的向上规模优化或向下规模优化。使用流量计时, 应能在输入 PV_IN 中提供积分流量值。只要定量给料处于活动状态, 输出 Q 就置位。此输出可控制允许定量给料的设备。定量给料结束时, 将自动对漏泄量进行更正, 更正结果在下次定量给料时生效。在输入 DRIBB 处预设初始漏泄量。

将根据设定值对定量给料值进行监视, 以防超出或低于容错值, 在定量给料过程结束时, 会将结果提供给两个对应输出。

运行模式

内部/外部模式可通过操作员输入 SPEXTSEL_OP 或切换输入 SPEXON_L 设置。仅在输入 SPEXT_ON = True 时, 切换输入 SPEXTON_L 才有效。结果在“内部设定值”和“外部设定值”之间切换:

- 内部。设定值 (SP) 由操作员在 SP_OP 中输入并限制为 (SP_LLM, SP_HLM)。
- 外部。设定值 (SP) 从 SP_EXT 获得并且如上所述进行限定。

调用 OB

安装了该块的循环中断 OB (例如 OB32) 和 OB100。

启动定量给料

定量给料的启动步骤如下所示:

- 可通过设置 START_OP 输入或通过同一块中互连的信号 L_START 的上升沿来启动定量给料。
- 在天平平衡 (即 STNDSTLL = 1) 前有一段等待时间。如果没有可用的平衡信号, 则必须将该输入设置为 1。紧接着进行称皮重, 也就是说, 将当前过程值 PV_IN 输入到皮重存储器中。
- 根据操作结果是上升还是下降 (通过 REVERSE 选择), 按以下公式计算实际定量给料值 PV_OUT:
 - REVERSE = 0: $PV_OUT = PV_IN - TARA$ (PV_IN 上升)
 - REVERSE = 1: $PV_OUT = TARA - PV_IN$ (PV_IN 下降)
- 设置 Q、QSTRTDOS 并且复位 QEND_DOS、QTOL_P 和 QTOL_N。更多相关信息, 可参见“组件更改”部分。

结束定量给料

定量给料的最后阶段按下列步骤进行:

- 一旦 $PV_OUT \geq SP - DRIBB_F$, 就会立即复位 Q。如果除此之外还设置了 STNDSTLL = 1, 则将同时复位 QSTRTDOS。
- 一旦报告天平平衡 (STNDSTLL = 1), 就会通过在 RELAXTME 中定义的时间 [s] 初始化计数器, 然后计数器按采样时间 SAMPLE_T 循环递减。只要计数器 > 0, 稳定时间 (QRELXING = 1) 就继续前行。
- 稳定时间过后, 系统将根据所设置的容差限制 TOL_N 和 TOL_P 来评估所有给料不足或给料过量。如果 DRIB_COR = 1, 则执行漏泄量更正。
- 如果定量给料量在容差范围内, 将设置定量给料结束 (QEND_DOS = 1)。

组件更改

如果更改了组件, 请先设置 COMP_CHG = 1, 然后再启动定量给料。启动定量给料 (QSTRTDOS = 1) 时, 当前在 DRIBB 中设置的漏泄量值将传送到输出 DRIBB_F。

漏泄量更正

如果请求执行漏泄量更正 (DRIB_COR = 1), 则漏泄量的值计算如下:

$DRIBB_F := DRIBB_F - (SP - PV_OUT) * DCF/100$, 其中满足以下条件:

$$0 \leq DRIBB_F \leq DRIBBMAX$$

更正因子 DCF 在内部限制在 0 - 100 之内。

有关漏泄量值的详细信息, 可参见“组件更改”部分。

给料过量/给料不足

- 如果给料过量 ($PV_OUT > SP + TOL_P$), 将会设置 QTOL_P 和 QEND_DOS。
- 如果给料不足 ($PV_OUT < SP - TOL_N$), 则仅设置 QTOL_N。操作员可手动更正定量给料量。一旦完成更正, 便会报告定量给料结束 (QEND_DOS = 1)。如果没有启动其它定量给料操作, 则块输出将不再更新。

后期定量给料

只有检测到给料不足时, 操作员才可以更正定量给料量。可通过设置 POSTDOSE 或通过可互连的 L_PDose 输入来执行此操作。

- 设置 DRIB_COR = 1
- 信号上升沿时, 会设置用于启动定量给料的 QSTRT_DOS 信号, 信号持续时间为 PDOS_TME。可一直重复执行此操作, 直到超过设定值, 或直到操作员已通过设置 ACK_TOL_OP 输入或通过可互连的 ACK_TOL 输入对此操作做出确认为止。
- 确认后, 系统将指示定量给料结束 (QEND_DOS = 1), 而且不再更新输出。

暂停

- 必要时, 可通过 PAUSE_OP 命令或通过可互连的 PAUSE 输入中断定量给料。仅当输入 LIOP_SEL = True 时, 可互连的 PAUSE 输入才有效。如果给料过量 ($PV_OUT > SP + TOL_P$), 将会设置 QTOL_P 和 QEND_DOS。
- 设置暂停“结束”信号后, 将继续进行定量给料。如果给料过量, 则必须先结束暂停, 方可启动新的定量给料操作。

3.8 DOSE: 定量给料过程

取消

可通过 **CANCEL_OP** 命令输入或可互连的 **CANCEL** 输入提前取消定量给料。然后便可启动新定量给料操作。

错误处理

控制块时检测到的操作员错误在逻辑上是 **OR** 的关系，并且应用到 **QOP_ERR** 组输出。如果检测到数学错误，则设置输出 **ENO = 0** 和 **QERR = 1**。

启动特征

CPU 启动时，将模拟“中止定量给料”，但不生成消息。这意味着必须在启动 **OB** 中调用该块。在 **CFC** 工程组态中，这由 **CFC** 处理。使用 **STEP 7** 基本工具时，在启动 **OB** 中输入此调用。启动后，消息会在 **RUNUPCYC** 所设置的周期数内受到抑制。

时间响应

块必须在循环中断 **OB** 中调用。块的采样时间在 **SAMPLE_T** 参数中设置。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“DOSE 的 VSTATUS (页 142)”。

消息响应

DOSE 块使用 **ALARM8_P (MSG_EVID)** 块生成消息。“请求确认”消息不要求确认，该消息由 **NOTIFY (MSG_EVID1)** 生成。

存在下列消息触发器：

- 定量给料限制监视功能
- 定量给料的结束或取消
- 由互连作为控制系统错误而引用的 **CSF** 信号

可通过相关输入 **M_SUP1** 到 **3** 分别抑制报告超出限制值的消息。可使用 **MSG_LOCK** 集中禁用过程消息（不是控制系统消息！）。

如果重新启动以来未满足 **RUNUPCYC** 个周期，并且 **MSG_LOCK = TRUE** 或 **MSG_STAT = 21**，则会设置 **QMSG_SUP**。

监视过程值

不适用

更多信息

更多相关信息, 可参考:

DOSE 的 I/O (页 136)

DOSE 的消息文本和关联值 (页 140)

DOSE 的 VSTATUS (页 142)

DOSE 的操作员监控 (页 142)

3.8.2 DOSE 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
ACK_TOL	ACK_TOL_OP 的可互连输入	BOOL	0	I		
ACK_TOL_OP	确认给料不足	BOOL	0	IO	+	
AK_OP_EN	1= 确认的操作员控制使能信号	BOOL	1	I		
AUX_PRx	关联值 x	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 连续批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	"	I	+	
CANCEL	用于取消的可互连输入	BOOL	0	I		
CANCEL_OP	在正沿时取消当前定量给料操作	BOOL	0	IO	+	
CN_OP_EN	1 = CANCEL 的操作员控制使能信号	BOOL	1	I		
COMP_CHG	1 = 下次开始定量给料时组件更改	BOOL	0	I	+	
CSF	控制系统错误	BOOL	0	I		
DCF	百分比形式的漏泄量更正因子	REAL	25	I	+	0...100
DRIB_COR	1 = 激活漏泄量更正	BOOL	0	I	+	
DRIBB	漏泄量初始值	REAL	0	I	+	
DRIBB_F	当前漏泄量值	REAL	0	O	+	
DRIBBMAX	最大漏泄量值 (由于在实例化前不知其尺寸, 因而可在此任意选择默认值。)	REAL	999	I	+	
ER	定量给料错误 (ER = SP - PV_OUT)	REAL	0	O	+	
L_PDOSE	可互连输入后期定量给料	BOOL	0	I		
L_START	用于启动的可互连输入	BOOL	0	I		
LIOP_SEL	1 = 互连 (即暂停) 处于激活状态, 0 = 操作员控制处于激活状态	BOOL	0	I	+	
M_SUP_1	正常定量给料消息抑制	BOOL	0	I	+	

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
M_SUP_2	给料过量消息抑制	BOOL	0	I	+	
M_SUP_3	给料不足消息抑制	BOOL	0	I	+	
MO_PVHR	显示上限 (测量范围)	REAL	110	I	+	
MO_PVLR	显示下限 (测量范围)	REAL	-10	I	+	
MSG_ACK	确认消息	WORD	0	O		
MSG_EVID	ALARM_8P 事件 ID	DWORD	0	I		
MSG_LOCK	1 = 消息锁定取决于过程状态	BOOL	0	I	+	
MSG_STAT	消息错误信息	WORD	0	O		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
P_OFF_EN	1 = “继续”的操作员控制使能信号	BOOL	1	I		
P_ON_EN	1 = “暂停”的操作员控制使能信号	BOOL	1	I		
PAUSE	PAUSE_OP 的可互连输入	BOOL	0	I		
PAUSE_OP	1 = 停止当前定量给料操作 (暂停), 0 = 继续	BOOL	0	IO	+	
PD_OP_EN	1 = 后期定量给料的操作员控制使能信号	BOOL	1	I		
PDOS_TME	后期定量给料时间 [s]	REAL	0	I	+	
POSTDOSE	在正沿时进行后期定量给料	BOOL	0	IO	+	
PV_IN	过程值加权 (加权输入)	REAL	0	I	+	
PV_OUT	定量给料过程值	REAL	0	O	+	
Q	1 = 控制定量给料设备	BOOL	0	O	+	
Q_SP_OP	1 = 设置设定值的操作员控制使能信号	BOOL	0	O	+	
QAK_OP	1 = 确认的操作员控制使能信号	BOOL	0	O	+	
QC_PV_IN	PV_IN 的质量代码	BYTE	16#80	I		
QC_Q	Q 的质量代码	BYTE	16#80	O		
QC_Q_I	输出 Q 的质量代码	BYTE	16#80	I		
QCN_OP	1 = CANCEL 的操作员控制使能信号	BOOL	0	O	+	
QEND_DOS	1 = 定量给料已完成	BOOL	1	O	+	
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
QMSG_ERR	1 = 消息错误	BOOL	0	O		
QMSG_SUP	1 = 消息抑制激活	BOOL	0	O	+	
QOP_ERR	1 = 操作员错误	BOOL	0	O		
QP_OFF_EN	1 = “继续”的操作员控制使能信号	BOOL	0	O	+	
QP_ON_EN	1 = “暂停”的操作员控制使能信号	BOOL	0	O	+	
QPD_OP	1 = 后期定量给料的操作员控制使能信号	BOOL	0	O	+	
QRELXING	1 = 设置时间处于激活状态	BOOL	0	O	+	
QSP_HLM	1 = 限制操作员设定值上限	BOOL	0	O		
QSP_LLM	1 = 限制操作员设定值下限	BOOL	0	O		
QSPEXTEN	1 = 外部切换的操作员控制使能信号	BOOL	0	O	+	
QSPEXTON	0 = 内部, 1 = 外部	BOOL	0	O	+	
QSPINTEN	1 = 内部切换的操作员控制使能信号	BOOL	0	O	+	
QSTRT_OP	1 = 启动定量给料的操作员控制使能信号	BOOL	0	O	+	
QSTRTDOS	1 = 定量给料已启动	BOOL	0	O	+	
QTOL_N	1 = 定量给料结束后给料不足	BOOL	0	O	+	
QTOL_P	1 = 定量给料结束后给料过量	BOOL	0	O	+	
RELAXTME	定量给料停止后的空闲时间 (秒)	REAL	3	I	+	
REVERSE	0 = 增重, 1 = 减重	BOOL	0	I	+	
RUNUPCYC	起动周期数	BYTE	3	I		
SAMPLE_T	周期时间 (以秒为单位)	REAL	1	I		> 0
SP_EXT	外部设定值	REAL	0	I	+	
SP_HLM	设定值上限	REAL	100	I	+	
SP_LLM	设定值下限	REAL	0	I	+	
SP_OP	设定值	REAL	0	IO	+	
SP_OP_ON	1 = 内部设定值的操作员控制使能信号	BOOL	1	I		
SPBUMPON	1 = 启用无波动设定值	BOOL	0	I	+	
SPEXON_L	用于选择 SP_EXT 的可互连输入 (0 = 内部, 1 = 外部)	BOOL	0	I		
SPEXT_EN	1 = 外部设定值的操作员控制使能信号	BOOL	1	I		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
SPEXT_ON	1 = 互连: SP_EXT 为激活状态, 0 = 操作员控制处于激活状态	BOOL	0	I	+	
SPEXTSEL_OP	用于选择 SP_EXT 的输入已启用 (0 = 内部, 1 = 外部)	BOOL	0	IO	+	
SPINT_EN	1 = 内部模式的操作员控制使能信号	BOOL	1	I		
ST_OP_EN	1 = 启动定量给料的操作员控制使能信号	BOOL	0	I		
START_OP	1 = 在正沿时启动定量给料	BOOL	0	IO	+	
STEP_NO	BATCH 步号	DWORD	0	I	+	
STNDSTLL	1 = 平衡	BOOL	1	I	+	
TOL_N	容差范围下限	REAL	0	I	+	
TOL_P	容差范围上限	REAL	0	I	+	
USTATUS	VSTATUS 中的状态字, 可由用户组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.8.3 DOSE 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号 MSG_EVID	块参数	默认消息 文本	消息 类别	可由 此项抑制
1	(SP-TOL_N ≤ PV_OUT ≤ SP+TOL_P)	\$\$BlockComment\$\$ 定量给料正常	PM	M_SUP_1、 MSG_LOCK
2	QTOL_P	\$\$BlockComment\$\$ 给料过量	AH	M_SUP_2、 MSG_LOCK
3	QTOL_N	\$\$BlockComment\$\$ 给料不足	AL	M_SUP_3、 MSG_LOCK
4	CSF	\$\$BlockComment\$\$ 外部故障	S	-
5	CANCEL	\$\$BlockComment\$\$ 定量给料已停止	PM	MSG_LOCK
6,7,8			无消息	-
消息号 MSG_EVID1	块参数	默认消息 文本	消息 类别	可由 此项抑制
	(PV_OUT < SP-TOL_N)	\$\$BlockComment\$\$ 确认请求	OR	-

消息块的前三个关联值分配有 SIMATIC BATCH 数据，第四个为 PV_OUT 保留，其余值 (AUX_PRx) 可随意分配。

将关联值分配给块参数

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	PV_OUT
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.8 DOSE: 定量给料过程

3.8.4 DOSE 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7		6	5	4		3	2		1	0
参数	QSTRTDOS		-		QSPEXTON		-	MSG_LOCK		BA_EN	OCCUPIED
位号:	15	14		13	12	11		10	9		8
参数	OOS	QMSG_SUP		-	-	QTOL_N		QTOL_P	QEND_DOS		QRELXING

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.8.5 DOSE 的操作员监控

更多信息

更多相关信息，请参考以下部分：

- DOSE 块图标 (页 689)
- DOSE 面板 (页 600)

3.9 ELAP_CNT: 运行时间计数器

3.9.1 ELAP_CNT 的描述

对象名 (类型 + 编号)

FB64

- ELAP_CNT 块 I/O (页 145)
- ELAP_CNT 块图标 (页 691)
- ELAP_CNT 面板 (页 145)

功能

ELAP_CNT 块可检测单元的运行时间。

工作原理

只要 ON_OFF 输入 = 1 (即, 只要连接单元可以运行), 块 ELAP_CNT 就会检测时间。每次执行时, 值 $SAMPLE_T[s]/3600$ 都会加到值 HOURS 中。因此, 输出 HOURS 指示了运行小时数。

调用 OB

这是在其中安装了该块的循环报警 OB (例如 OB32) 和 OB100。

设置计数器

某些情况下 (例如, 维护或替换单元后), 必须指定运行时间计数器的初始值 (通常为 0)。在 OS 上, 操作员在 HOURS_OP 输入中设置跟踪值。然后, 通过 TRACK_OP 输入, 或者通过将 TRACK 输入与 HOURS 输出互连来将该值传递给 HOURS 输出。

错误处理

运算错误由 $ENO = 0$ 或 $QERR = 1$ 表示。

3.9 ELAP_CNT: 运行时间计数器

启动特征

无特殊措施。启动之后，将在 RUNUPCYC 所设置的周期数内抑制消息。

时间响应

该块只能在循环中断 OB 中正常工作。为确保正确的时间采集，该块应该与受监视单元的控制块安装（如果在 CFC 中组态）在同一运行组中。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“ELAP_CNT 的 VSTATUS (页 148)”。

消息响应

ELAP_CNT 块使用 ALARM_8P 块生成消息。

消息由监视运行时间限制的功能触发。

通过使用相关的 M_SUP_xx 输入可以分别抑制超出限制消息。可使用 MSG_LOCK 集中禁用过程消息（不是控制系统消息！）。

如果重新启动以来未满足 RUNUPCYC 个周期，并且 MSG_LOCK = TRUE 或 MSG_STAT = 21，则会设置 QMSG_SUP。

监视过程值

不可用

更多信息

更多相关信息，可参考：

ELAP_CNT 的 I/O (页 145)

ELAP_CNT 的消息文本和关联值 (页 147)

ELAP_CNT 的 VSTATUS (页 148)

操作和监视 ELAP_CNT (页 148)

参见

ELAP_CNT（所有视图） (页 608)

3.9.2 ELAP_CNT 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
AUX_PRx	关联值 x	ANY	0	IO	
HOURS	运行时间 (小时)	REAL	0	O	+
HOURS_AH	报警上限 (小时)	REAL	100	I	+
HOURS_OP	跟踪值 (小时)	REAL	0	IO	+
HOURS_WH	警告上限 (小时)	REAL	95	I	+
M_SUP_AH	1 = 抑制上限报警	BOOL	0	I	+
M_SUP_WH	1 = 抑制上限警告	BOOL	0	I	+
MO_HOUHR	显示上限	REAL	120	I	+
MO_HOULR	显示下限	REAL	0	I	+
MSG_ACK	确认消息	WORD	0	O	
MSG_EVID	消息号	DWORD	0	I	
MSG_LOCK	1 = 锁定过程消息	BOOL	0	I	+
MSG_STAT	消息错误信息	WORD	0	O	
ON_OFF	设备状态: 1 = 打开, 0 = 关闭	BOOL	0	I	+
OOS	预留	BOOL	0	I	+
QC_ON_OFF	ON_OFF 的质量代码	BYTE	16#80	I	
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+
QH_ALM	1 = 激活上限报警	BOOL	0	O	
QH_WRN	1 = 激活上限警告	BOOL	0	O	
QMSG_ERR	1 = 消息错误	BOOL	0	O	+
QMSG_SUP	1 = 消息抑制	BOOL	0	O	+
RUNUPCYC	起动周期数	BYTE	3	I	
SAMPLE_T	采样时间 (秒)	REAL	1	I	

系列: CONTROL

3.9 ELAP_CNT: 运行时间计数器

I/O (参数)	含义	数据类型	默认值	类型	OCM
TRACK	TRACK 的可互连输入	BOOL	0	I	
TRACK_OP	1 = 将 HOURS 设置为 HOURS_OP	BOOL	0	IO	+
USTATUS	VSTATUS 中的状态字, 可由用户组态	WORD	0	I	
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+

3.9.3 ELAP_CNT 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由 此项抑制
1	QH_ALM	\$\$BlockComment\$\$ 上上限报警	M	M_SUP_AH、 MSG_LOCK
2	QH_WRN	\$\$BlockComment\$\$ 上限报警	M	M_SUP_WH、 MSG_LOCK

消息块的所有关联值 (AUX_PRx) 可由用户随意分配。

将关联值分配给块参数

关联值	块参数
1	AUX_PR01
2	AUX_PR02
3	AUX_PR03
4	AUX_PR04
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.9 ELAP_CNT: 运行时间计数器

3.9.4 ELAP_CNT 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	ON_OFF	-	-	-	-	MSG_LOCK	-	-
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	-	-	-	-	-	-

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.9.5 ELAP_CNT 的操作员监控

更多信息

更多相关信息，请参考以下部分：

- ELAP_CNT 块图标 (页 691)
- ELAP_CNT 面板 (页 608)

3.10 FMCS_PID: 控制器块

3.10.1 FMCS_PID 的描述

对象名 (类型 + 编号)

FB114

- FMCS_PID 块 I/O (页 165)
- FMCS_PID 块图标 (页 692)
- FMCS_PID 面板 (页 610)

应用领域

块 FMCS_PID 用于连接 FM 355 控制器块。

它可用于 C (连续控制器) 和 S (步控制器和脉冲控制器) 模块类型。它本身不包含任何控制算法, 因为控制功能只能在模块上执行。可使用 FMCS_PID 块来监视所有相关过程值以及更改所有相关控制器参数。

有关 FM 355 的应用实例以及输入和输出参数的详细描述, 可参考控制器模块 FM 355 的手册。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 设置参数 CO_NO
- 输入 EN_CO 互连到 FM_CO 块的输出 EN_CO_x (x = 机架号)
- 输出 ENCO 连接到 FM_CO 块的输入 ENCOx_yy (x = 机架号, yy = 配位号)。

将 FM 355 C 控制器模块作为连续控制实现

该块提供了下列显示和设置选项:

- 显示在模块上执行的针对过程值 PV 或误差信号 ER 的两个限制对 (QH_ALM、QH_WRN、QL_WRN、QL_ALM 输出) 的限制监视结果。MONERSEL 用于指定是监视 PV 还是监视 ER。
- 禁止在超出限制时生成各个消息
- 分割范围功能
- 误差信号分支中的死区 (DEADB_W, 打开阈值)
- 指定控制算法: PID 算法 (QFUZZY = 0) 或 FUZZY 算法 (QFUZZY = 1)
- 调节变量跟踪
- 禁用积分作用
- 设定值跟踪 (SP = PV)

将 FM 355 S 控制器模块作为脉冲控制实现

使用控制器模块作为脉冲控制器而不是连续控制器意味着无法实现分割范围控制。

可使用脉冲控制器生成脉冲调宽控制信号。控制信号将转换为二进制输出信号, 以使指定数字输出处的脉冲长度和已组态周期之间的比率与调节值 LNM 相符。

将 FM 355 S 控制器模块作为步控制器实现

使用控制器模块作为步控制器而不是连续控制器会导致以下差异:

- 输出 QLMNR_ON 指示反馈信号是否可用 (1: 存在; 0: 不存在)。
- 无法进行分割范围操作。

作为不带位置反馈 (QLMNR_ON = 0) 的步操作控制器使用时, 只能在端位置对操纵的变量进行手动调整。在这种情况下, 安全位置或外部调节变量值由控制器模块按以下所述进行解释:

值 < 40 %: 完全关闭执行元件

值 > 60 %: 完全打开执行元件

40% ≤ 值 ≤ 60%: 保持当前设置

调用 OB

循环中断 OB: 例如 OB 32

使用同一实例将该块安装在启动块 OB100 中。

还要注意 FM_CO 块的依赖性。

更多信息

更多相关信息, 可参考:

寻址 (页 152)

功能 (页 152)

通过过程映像采集过程值 (页 154)

生成设定值、限制、误差信号和调节变量 (页 156)

手动、自动和跟踪模式 (页 158)

模式更改 (页 160)

安全模式 (页 161)

将参数下载到模块 (页 161)

从模块中读取数据 (页 162)

错误处理 (页 162)

启动行为、动态响应和消息响应 (页 163)

FM 355 的备份模式 (页 164)

FMCS_PID 的 I/O (页 165)

FMCS_PID 的消息文本和关联值 (页 172)

FMCS_PID 的 VSTATUS (页 174)

归档过程值 (页 728)

操作和监视 FMCS_PID (页 174)

3.10.2 寻址

属于实例的 FM 355 的控制器通道通过其逻辑基地址（在 HW Config 中设置）（LADDR 输入）和控制器通道号（CHANNEL 输入）进行寻址。CHANNEL 输入的允许值为 1 到 4。对 CHANNEL 进行更改后必须将 ACC_MODE 设置为 TRUE。

FM 355 模块通过 PCS 7 库的块进行监视。MODE 输入与 MOD_D1 块的 OMODE 输出互连。该块仅通过 FM 355 控制通道进行通讯。因此，输出 OMODE 低位字中的测量范围编码并不相关，将为其分配零。

3.10.3 FMCS_PID 的功能

功能

FMCS_PID 块构成了控制模块 (FM 355) 与 SIMATIC PCS 7 库块之间的接口。它还可以与其它 SIMATIC S7 块互连。

该块与控制器模块相互按异步方式操作。

所有相关过程和干扰变量都由模块提供，且只能由该块读取。该块还可以向控制器模块传送不同的操作模式和设置。

过程值按周期读取。（例外：SP（来自 FM 的设定值）、ER（误差信号）、DISV（干扰）、LMN_A 和 LMN_B）。这些值每 4 个周期后才会进行更新。更多相关信息，请参考“通过过程映像采集过程值 (页 154)”。

只能从 FM 355 使用“读取数据记录”(SFC59) 读取参数 SP（来自 FM 的设定值）、ER（误差信号）、DISV（干扰变量）、LMN_A 及 LMN_B。更多信息，可参考“通过数据记录读取”。

将为每个过程值 PV 和 LMN 生成质量代码，它可具有以下状态：

状态	质量代码
有效值	16#80
无效值	16#00

OP_SEL 输入

FM 355 一般通过块获得其参数。对该块中参数的每一项更改都将传递到模块。不过, 如果想通过操作员面板 (OP) 直接在 FM 355 处更改参数, 必须先在 OP_SEL 处启用此功能 (设置 OP_SEL = 1)。如果在使用了 OP 后想再次通过块限制操作员控制和参数分配, 则必须复位 OP_SEL。

启用了通过 OP 的组态时, 控制器模块将不接受任何来自块的参数。不过, 块会继续更新过程值 SP_OP_ON、LMNOP_ON、SP_OP 和 LMN, 从而可无波动地切换到块执行组态时所在的模式。设置 OP_SEL = 0 时, 即会用块实例的数据覆盖其余参数 (例如 GAIN)。如果在反转 OP_SEL 前没有在块实例中输入数据, 通过 OP 建立的条目将会丢失。

SDB_SEL 输入

某些参数除了可使用组态工具进行指定外, 还可通过 FMCS_PID 块来指定。这两个参数集可能会有差异。可通过块的 SDB_SEL 输入避免此类冲突。SDB_SEL = 1 用于指定该模块仅从块读取这些参数, 而不从组态工具读取。设置 SDB_SEL = 0 用于指定模块接受来自参数分配工具和块的这些参数。请注意, CPU 每次 STOP-RUN 切换后都会将来自组态工具的参数发送到模块。另一方面, 每次块输入发生更改时也会将块的参数传送到模块。

允许通过 OP 的组态时, 将会禁用操作员输入, 但有以下例外: 可执行操作 OP_SEL = 0 (通过 OP 的操作禁用)。

3.10.4 通过过程映像采集和写入过程值

读取过程值

从过程映像周期性读取过程值（SP（来自 FM 的设定值）、ER（误差信号）、DISV（干扰）、LMN_A 和 LMN_B 除外）。这些值每 4 个周期过后才会进行更新。在正常情况下，这在安装 FMCS_PID 的周期性中断 OB 中进行。为获得更高的精确性，可将 FMCS_PID 额外安装在具有更高时钟频率的另一个周期性中断 OB 中。

重新启动后或将 ACC_MODE 设置为 1 时，块会确定所用的周期性中断（最多允许使用 2 个）；更多信息，请参考“启动特征 (页 163)”。

选择周期性中断时必需考虑以下情况：

- 安装块的周期性中断 OB 的运行速度必须在 25 ms 与 30 s 之间。

注：切勿使用 OB1！

- 在 HW Config 中必须为较快的 OB 设置过程映像分区 (TPA)。
- 如果块周期长于模块的周期时间，则在 4 个块周期后才会通过块的过程映像来完成读取（在 OB32 [1000 ms] 的实例中为 4 s 之后），否则将在 4 个模块周期后完成。
- **实例：**模块上有 4 个已处理的模拟输入时，模块的周期时间为 400 ms（请参见 FM 355 组态接口上的“模块参数”(Module parameters) 命令按钮）。例如，如果块安装在 OB32 (1000 ms) 中，则也可以通过在 OB33 (500 ms) 中同时进行附加安装来加快过程映像的读取速度。

写入过程值

每到第二个周期，就会将以下过程值写入过程映像中：

- 设定值（仅限自动模式）
- 操纵值（仅限手动模式）
- SP_OP_ON、SAFE_ON、QMAN_AUT、LMNTRKON、LMN_REON、LMNRHSRE、LMNRLSRE
- LMNSOPON
- LMNUP 或 LMNUP_OP
- LMNDN 或 LMNDN_OP
- FUZID_ON
- LMNRS_ON

在以下情况下，必须重新启动或在编译和下载后必须将 ACC_MODE 设置为 1：

- 如果已将块移动到其它周期
- 如果快速周期中也安装了块
- 如果还删除了安装的块

3.10.5 设定值、限制值、误差信号和调节变量的生成

通过 FMCS_PID 块生成设定值

可从四个不同的来源获得设定值 SP:

- 从控制器模块（已在 OP 上对其进行组态，或模块处于备份模式下）。在此情况下，将禁用设定值调整，且应用的值将写入到该块的操作员输入 SP_OP。
- 其它三个来源取决于输入 SP_TRK_ON、SPEXTSEL_OP 和 SP_OP_ON。

有关外部/内部设定值的更多信息，可参考“模式更改 (页 160)”。

SP_TRK_ON	SPEXTSEL_OP	SP_OP_ON	SP=	状态
不相关	0	0	SP_INT	内部设定值
0	0	1	SP_OP	内部（操作员控制）设定值
不相关	1	不相关	SP_EXT	外部设定值
1	0	1	PV **	跟踪设定值
			** 仅手动模式且 SPBUMPON = 1 时	

有效设定值限制在一定范围内 (SP_LLM, SP_HLM)。

如果设置 SP_TRK_ON，将在手动模式（当 SP_OP_ON = 1、SPEXTSEL_OP = 0 且 SPBUMPON = 1 时）下跟踪设定值 SP_OP。这使得能够从手动模式无扰动切换到自动模式。

通过回写激活设定值和调节变量确保无波动地切换到手动操作。

限制生成

这取决于 MONERSEL 的输入值，控制器模块既监视过程值 PV (MONERSEL = 0) 也监视误差信号 ER (MONERSEL = 1)，检查是否超过警告和报警极限值 (L_WRN、H_WRN、L_ALM、H_ALM)。将通过常用滞后量 HYS 进行监视。

块在 QL_WRN、QH_WRN、QL_ALM 和 QH_ALM 输出处提供监视结果。除非启用了消息抑制，监视过程值 PV 时，块会通知任何超出上限和下限的情况。

误差信号生成

误差信号是在激活设定值 SP 和过程值 PV 的基础上由控制器模块生成的，在块的输出 ER 处提供。

死区 DEADB_W 到期后，将在 PID 算法中进一步处理误差信号。而不会添加干扰变量。

通过 FMCS_PID 块生成调节变量

通过各种来源得出调节值 LMN。如果若干个控制输入同时为 TRUE，则其优先级如下：

优先级	控制输入	LMN	状态
1	LMNS_ON = 1	LMNUP 或 LMNDN	带步控制器的已互连控制信号
2	SAFE_ON = 1	= LMN_SAFE	安全模式
3	手动	= LMN_OP	手动模式 (QMAN_AUT = 0)
4	LMNTRKON = 1	= 模块上的模拟输入或 0.0	跟踪模式
5	LMN_REON = 1	= LMN_RE	外部调节变量
6	自动	= PID 算法	自动模式 (QMAN_AUT = 1)

- 如果 LIOP_MAN_SEL = 0，则 AUT_ON_OP 参数将用于在 OS 上于手动模式和自动模式之间切换。
- 如果 LIOP_MAN_SEL = 1，则 AUT_L 参数将用于 CFC 中通过互连在手动模式和自动模式之间切换。

在步控制器中，在考虑电机特有参数“电机运行时间”(MOTOR_TM)、最短脉冲持续时间 (PULSE_TM) 和最短中断持续时间 (BREAK_TM) 的情况下，调节变量将转换为控制脉冲 (QLMNUP; QLMNDN)。

3.10.6 FMCS_PID 的手动、自动和跟踪模式

手动模式

操纵变量由操作员在输入 LMN_OP 处设置（也可在此处使用点动模式）。切换到自动模式时，模块将接管“手动”设置为操作点的操纵变量。

“手动模式”优先于“跟踪模式”。

步控制器的手动模式:

可使用执行命令直接控制步控制器的控制信号。通过 LMNSOPON 启用信号操纵；在 LMNUP_OP 或 LMNDN_OP 处设置控制信号。取消操作员输入或到达端位置时，才会调整阀。

操作员调整控制信号的优先级高于通过 LMN_OP 调整调节值的优先级。

使用无位置反馈的步控制器时，无法控制 LMN_OP 输入。

自动模式

通过模块的 PID 或模糊算法来计算操纵变量。控制参数 GAIN、TI、TD 和 TM_LAG 可进行互连。

通过设置负比例 GAIN 值，可颠倒控制器的控制方向（上升的误差信号会导致操纵变量下降）。

可通过设置 $TI = 0$ 禁用 I 作用。

可由操作员控制的操纵变量输入 LMN_OP 会跟踪 LMN 输出，以确保无波动地从自动模式切换到手动模式。

外部设定值 (LMN_RE)

块会将值 LMN_RE 传送到 FM 355。如果设置 LNM_REON = 1，FM 355 将把外部操纵变量 LMN_RE 作为操纵变量 LMN 来接受。

“外部操纵变量”模式的优先级高于“自动模式”。

操纵变量跟踪

在操纵变量跟踪模式 (LMNTRKON = 1) 下，操纵变量会跟踪模块的模拟输入或值 0.0。

“跟踪”模式优先于“外部操纵变量”模式。

安全模式 (LMN_SAFE)

块会将值 LMN_SAFE 传送到 FM 355。如果设置 SAFE_ON = 1, FM 355 将把安全操纵变量 LMN_SAFE 作为操纵变量 LMN 来接受。

“安全模式”的优先级高于“跟踪”模式。

禁用控制信号和手动值的操作员控制使能

必须将 LMNOP_ON 和 LMNSOPON 设置为 0。

步控制器的可互连控制信号设置

使用 LMNS_ON 且通过互连的 LMNUP 和 LMNDN 输入直接连接控制信号的跟踪模式在所有模式中具有最高优先级。如果设置了 LMNS_ON, 则只能通过输入 LMNUP 或 LMNDN 设置控制信号。只要设置了 LMNS_ON, 便可抑制对控制信号的任何其它影响。

3.10.7 FMCS_PID 的模式更改

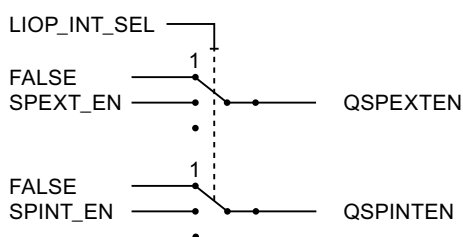
模式更改可由操作员或通过互连的输入触发。模式通过分配给模式的操作员控制块来更改。

外部/内部设定值

OS 操作员通过互连 SPEXON_L 设置 SPEXTSEL_OP 输入，依次来启动外部和内部设定值之间的切换。必须设置相应的使能输入 SPINT_EN、SPEXT_EN 或选择输入 LIOP_INT_SEL 来启用这些切换。

必须将 SP_OP_ON 设置为 TRUE 以启用操作员对设定值的控制。

启用内部和外部设定值之间的切换



QSPEXTEN = TRUE: 可将 SPEXTSEL_OP 从 FALSE (内部设定值) 更改为 TRUE (外部设定值)。

QSPINTEN = TRUE: 可将 SPEXTSEL_OP 从 TRUE (外部设定值) 更改为 FALSE (内部设定值)。

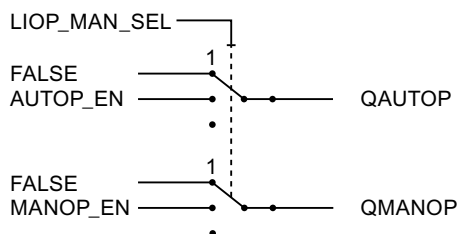
根据需要对 SPEXTSEL_OP 进行跟踪或复位。

手动/自动

OS 操作员通过设置 AUT_ON_OP 输入或互连 AUT_L 来切换操作模式。必须设置相应的使能输入 MANOP_EN、AUTOP_EN 或选择输入 LIOP_MAN_SEL 来启用该切换。

必须将 LMNOP_ON 设置为 TRUE 以启用手动变量控制。

启用手动和自动模式之间的切换



QAUTOP = TRUE: 可将 AUT_ON_OP 从 FALSE (手动模式) 更改为 TRUE (自动模式)。

QMANOP = TRUE: 可将 AUT_ON_OP 从 TRUE (自动模式) 更改为 FALSE (手动模式)。

根据需要对 AUT_ON_OP 进行跟踪或复位。

针对无波动切换的措施

为确保无波动切换, 针对下列模式采取了特殊措施:

- 外部设定值/跟踪设定值:
如果 SPBUMPON = TRUE, 内部设定值 SP_OP 将设置为有效 (外部或跟踪) 设定值。
- 自动模式、安全模式、跟踪模式或外部操纵变量 LMN_RE:
将跟踪操纵值 LMN_OP 至有效的操纵变量。
- 手动模式、安全模式、跟踪模式或外部操纵变量 LMN_RE:
将对积分器进行跟踪, 以便能够无波动地切换到自动模式 (不通过步控制器)。使用组态工具可以禁用默认无波动切换功能。这样可以更快地更正误差信号。

3.10.8 安全模式

可互连的输入 SAFE_ON 用于激活安全模式。该模式由具有最高优先级的控制器模块执行。在安全模式下, 块的 LMN_SAFE 输入处的值在控制输出处输出。

3.10.9 将参数下载到模块

只要相应的块参数发生变更, 便会将通道特有的控制器和操作参数传送到控制器模块中。只要禁用了通过 OP 的操作员控制, 模块就会拒绝块写入的参数。

将控制器和操作参数下载到控制器模块可能需要进行几次块调用。

3.10.10 从模块中读取数据

每当调用块时，控制器模块就会读取通道特定的过程值。尤其是在分布式操作中，读取可能需要进行几次块调用。

如果更改了模块上 OP 处的通道特定控制器和运行参数，该块也会从控制器模块中读取当前参数。它随后会更新 SP_OP_ON（设定值操作开启）、LMNOP_ON（调节变量操作开启）、SP_OP（操作设定值）和 LMN_OP（操作调节变量）输入。

3.10.11 FMCS_PID 的错误处理

错误显示

FMCS_PID 块提供以下错误显示：

错误显示	含义
QOP_ERR = 1	操作员输入错误。如果没有出现新的操作员输入错误，将在下一个块周期中复位 QOP_ERR。
QPARF_FM = 1	<ul style="list-style-type: none"> 使用参数分配工具直接向控制器模块分配参数时出错。 在块上设置了无效的控制通道号 (CHANNEL)。 在 OB 中安装时出错；更多信息，请参考“启动特征”。
QCH_F = 1	通道错误。由于硬件故障，属于实例的控制器通道无法返回有效结果。
QMODF = 1	控制器模块已移除或发生故障。
QPERAF = 1	I/O 访问错误。该块无法访问控制器模块。

3.10.12 FMCS_PID 的启动行为、动态响应和消息响应

启动特征

在 CPU 启动期间, FMCS_PID 块首次运行时, 将设置操作模式 MANUAL 和 INTERNAL。

在 OB100 中, QDONE 设置为 0。

CPU 重新启动后或 ACC_MODE 设置为 1 时, 块最多约 30 秒无法运行。

重新启动 (OB100) 或 ACC_MODE 设置为 1 后块将使用前 30 秒检测块实例是否安装在多个 OB 中。如果随后将块移动到其它 OB, 则编译和下载后必须执行重新启动或必须将 ACC_MODE 设置为 1 (否则将收到错误消息 QPARF_FM = 1, 而且块将不会再提供任何数据)。

说明

通过在 HW Config 中选择菜单命令“保存/编译并下载”(Save/Compile and Download), 可以将组态工具的参数设置写入 SDB (系统数据)。这些 SDB 参数可能与块参数有差异。SDB_SEL = 0 时, CPU 每次 STOP-RUN 切换时都会将 SDB 参数发送到模块。不过, 块参数几个周期后才会写入到模块。SDB_SEL = 1 时, CPU 进行 STOP-RUN 切换时模块不会加载 SDB 参数。这会避免在 SDB 参数与块参数有差异时在调节变量中发生跳转。

要区分以下情况:

- FM 355 出现故障, 且在出现故障之前并未通过 OP 对设置进行更改。
块会将当前控制器和运行参数传送到 FM 355。
- FM 355 出现故障, 且在出现故障之前通过 OP 对设置进行了更改。
块会从 FM 355 读取当前值并更新其 SP、LMN、Q_SP_OP 和 QLMNOP 输出。
- FM 355 未出现故障, 且通过 OP 对设置进行了更改。
块会从 FM 355 读取当前值并更新其 SP、LMN、Q_SP_OP 和 QLMNOP 输出。
- FM 355 未出现故障, 且未通过 OP 对设置进行更改。
FM 355 与块的控制器和运行参数完全相同。块不执行任何操作。

块在启动过程 (但不会在初始运行过程) 中将控制器和运行参数传送到 FM 355。

时间响应

不可用

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“FMCS_PID 的 VSTATUS (页 174)”。

消息响应

FMCS_PID 块使用 ALARM_8P 块生成消息。

存在下列消息触发器：

- 过程值或系统偏差限制监视功能
- 用于监视模块硬件的功能（主要由 MOD_D1 块进行处理）。

通过使用相关的 M_SUP_xx 输入可以分别抑制超出限制消息。可使用 MSG_LOCK 集中禁用过程消息（不是控制系统消息！）。

如果自上次重新启动后未满足 RUNUPCYC 周期数，并且 MSG_LOCK = TRUE 或 MSG_STAT = 21，则会设置 QMSG_SUP。

监视过程值

不可用

3.10.13 FM 355 的备份模式

如果 CPU 切换到 STOP 或出现故障，FM 355 将切换到备份模式。在此情况下，FM 355 会自动启用通过 OP 的操作员控制（像 OP_SEL = 1 时那样发挥作用）。

3.10.14 FMCS_PID 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
ACC_MODE	内部处理中包含 SUBN1_ID、SUBN2_ID、RACK_NO、SLOT_NO 和 CHANNEL	BOOL	0	IO		
AUT_L	用于 MANUAL (手动) /AUTO (自动) 模式的互连输入: 0 = 手动, 1 = 自动	BOOL	0	I		
AUT_ON_OP	用于 MANUAL (手动) /AUTO (自动) 模式的操作员输入: 0 = 手动, 1 = 自动	BOOL	0	IO	+	
AUTOP_EN	1 = 自动模式的操作员控制使能信号	BOOL	0	I		
AUX_PRxx	用户特定的关联过程值 6 ... 10	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 当前批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	0	I	+	
BREAK_TM	最短中断时间 (秒)	REAL	2	I		
CHANNEL	控制器通道号	INT	1	I		
CO_NO	读取数据记录的配位号	INT	0	I		
D_EL_SEL	D 操作输入	INT	0	I		
DEADB_W	死区宽度	REAL	0	I	+	
DISV	干扰变量	REAL	0	O		
EN_CO	当前配位号	STRUCT		IO		
ENCO	CFC 中的配位号“启用配位”	BYTE	16#00	O		
ER	误差信号	REAL	0	O		
FUZID_ON	模糊识别打开	BOOL	0	I		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
GAIN	比例增益	REAL	1	I	+	
H_ALM	报警上限	REAL	100	I	+	H_ALM > H_WRN >L_WRN > L_ALM
H_WRN	警告下限	REAL	90	I	+	H_ALM > H_WRN >L_WRN > L_ALM
HYS	滞后量	REAL	1	I	+	>= 0
L_ALM	报警下限	REAL	0	I	+	H_ALM > H_WRN >L_WRN > L_ALM
L_WRN	警告下限	REAL	10	I	+	H_ALM > H_WRN >L_WRN > L_ALM
LADDR	逻辑地址 FM 355	INT	0	I		
LIOP_INT_SEL	1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
LIOP_MAN_SEL	1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
LMN	调节值	REAL	0	O	+	
LMN_A	调节值 A: 分割范围功能/位置反馈	REAL	0	O		
LMN_B	调节值 B: 分割范围功能	REAL	0	O		
LMN_HLM	调节值上限	REAL	100	I	+	
LMN_LLM	调节值下限	REAL	0	I	+	
LMN_OP	调节变量的操作员输入	REAL	0	IO	+	
LMN_RE	外部调节变量	REAL	0	I		
LMN_REON	启用外部设定值	BOOL	0	I		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许 值
LMN_SAFE	安全调节值	REAL	0	I	+	
LMNDN	已互连的控制信号低	BOOL	0			
LMNDN_OP	可调整的控制信号低	BOOL	0	IO		
LMNOP_ON	1 = 操作员可输入 LMN_OP	BOOL	0	IO		
LMNRHSRE	位置反馈的上限位停止信号	BOOL	0	I		
LMNRLSRE	位置反馈的下限位停止信号	BOOL	0	I		
LMNRS_ON	开启位置反馈的模拟	BOOL	0	I		
LMNRSVAL	模拟位置反馈的开始值	REAL	0	I		
LMNS_ON	互连的控制信号开启 (LMNDN、 LMNUP)	BOOL	0	I		
LMNSOPON	激活控制信号的操作员控制使能	BOOL	0	I		
LMNTRKON	跟踪 (通过模拟输入跟踪调节变 量)	BOOL	0	I		
LMNUP	已互连的控制信号高	BOOL	0	I		
LMNUP_OP	可调整的控制信号高	BOOL	0	IO		
M_SUP_AH	1 = 消息抑制: 报警上限	BOOL	0	I	+	
M_SUP_AL	1 = 消息抑制: 报警下限	BOOL	0	I	+	
M_SUP_WH	1 = 消息抑制警告上限	BOOL	0	I	+	
M_SUP_WL	1 = 消息抑制警告下限	BOOL	0	I	+	
MANOP_EN	1 = MANUAL 的操作员控制使能 信号	BOOL	0	I		
MO_PVHR	显示上限 (测量范围)	REAL	110	I	+	
MO_PVLR	显示下限 (测量范围)	REAL	-10	I	+	
MODE	模式	DWORD	0			
MODE_CS	模式: 0 = 连续控制器, 1 = 步控 制器	BOOL	0	I	+	
MONERSEL	监视: 0 = 过程值, 1 = 误差信号	BOOL	0	I		
MSG_ACK	确认消息	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许 值
MSG_LOCK	1 = 消息抑制取决于过程状态	BOOL	0	I	+	
MSG_STAT	消息错误信息	WORD	0	O		
MTR_TM	电机运行时间 (秒)	REAL	60	I		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
OP_SEL	1 = 通过 OP 的操作员控制开启, 0 = 关闭	BOOL	0	I	+	
OPTI_EN	1 = 控制器调节开启, 0 = 关闭	BOOL	0	I	+	
P_SEL	激活 P 动作	BOOL	1	I		
PFDB_SEL	反馈路径中的 P 动作	BOOL	0	I		
PULSE_TM	最短脉冲持续时间 (秒)	REAL	2	I		
PV	过程值	REAL	0	O	+	
Q_SP_OP	1 = 设定值的操作员控制使能已激活	BOOL	0	O	+	
QAUTOP	1 = AUTO (自动) 模式的操作员 控制使能信号已激活	BOOL	0	O	+	
QC_LMN	LMN 的质量代码	BYTE	16#80	O		
QC_PV	PV 的质量代码	BYTE	16#80	I		
QCH_F	通道错误	BOOL	0	O		
QDNRLM	负设定值斜坡限制已触发	BOOL	0	O		
QDONE	1 = 参数已读取	BOOL	0	O		
QERR	ENO 取反	BOOL	1	O		
QFUZZY	0 = PID 算法 1 = 模糊	BOOL	0	O		
QID	1 = 模糊识别进行中	BOOL	0	O		
QH_ALM	上限: 触发报警	BOOL	0	O		
QH_WRN	上限: 触发警告	BOOL	0	O		
QL_ALM	下限: 触发报警	BOOL	0	O		
QL_WRN	下限: 触发警告	BOOL	0	O		
QLMN_HLM	已触发调节值的上限	BOOL	0	O	+	

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许 值
QLMN_LLM	已触发调节值的下限	BOOL	0	O	+	
QLMN_RE	0 = 手动, 1 = 自动	BOOL	0	O		
QLMNDN	调节值低	BOOL	0	O		
QLMNOP	1 = 用于调整调节变量值的操作员 控制使能已激活	BOOL	0	O	+	
QLMNOPON	调节变量操作已启用	BOOL	0	O		
QLMNR_HS	位置反馈的上限位停止信号	BOOL	0	O		
QLMNR_LS	位置反馈的下限位停止信号	BOOL	0	O		
QLMNR_ON	位置反馈已激活	BOOL	0	O		
QLMNS_ON	1 = LMNS_ON 激活	BOOL	0	O		
QLMNSAFE	安全模式	BOOL	0	O		
QLMNSOP	控制信号的操作员控制使能已激 活	BOOL	1	O	+	
QLMNTRK	跟踪模式	BOOL	0	O		
QLMNUP	控制信号高	BOOL	0	O		
QLMNVOP	控制信号 LMN_OP 的操作员控制 使能已激活	BOOL	1	O	+	
QMAN_AUT	0 = 手动, 1 = 自动	BOOL	0	O	+	
QMAN_FC	1 = 跟踪模式或从控制器处的防复 位终止	BOOL	0	O		
QMANOP	1 = 手动模式的操作员控制使能已 激活	BOOL	0	O	+	
QMODF	1 = 模块故障	BOOL	0	O		
QMSG_ERR	1 = 消息错误	BOOL	0	O	+	
QMSG_SUP	1 = 消息抑制激活	BOOL	0	O	+	
QOP_ERR	1 = 组操作员输入错误	BOOL	0	O		
QOP_SEL	通过 OP 的操作员控制: 0 = 关, 1 = 开	BOOL	0	O		
QPARF_FM	1 = 直接模块参数分配错误或 CHANNEL 故障	BOOL	0	O		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许 值
QPERAF	1 = I/O 访问错误	BOOL	0	O		
QSP_HLM	1 = 设定值上限已触发	BOOL	0	O		
QSP_LLM	1 = 设定值下限已触发	BOOL	0	O		
QSPEXTEN	1 = 外部设定值的操作员控制使能 已激活	BOOL	0	O	+	
QSPINTEN	1 = 内部设定值的操作员控制使能 已激活	BOOL	0	O	+	
QSPINTON	内部设定值开	BOOL	0	O	+	
QSPLEPV	模糊控制器显示: 设定值 < 过程 值	BOOL	0	O		
QSPOPON	设定值调整已激活	BOOL	0	O		
QSPR	分割范围模式	BOOL	0	O		
QUPRLM	正设定值斜坡极限	BOOL	0	O		
RACK_NO	机架号	BYTE	255	I		
RET_VALU	RD_REC 的返回值	WORD	0	O		
RUNUPCYC	起动周期数	INT	3	I		
SAFE_ON	采用安全位置	BOOL	0	I		
SDB_SEL	1 = CPU 进行 STOP-RUN 转换时 模块不采用 SDB 参数	BOOL	1	I	+	
SLOT_NO	插槽号	BYTE	255	I		
SP	设定值	REAL	0	O	+	
SP_EXT	外部设定值	REAL	0	I		
SP_HLM	设定值上限	REAL	100	I	+	
SP_INT	内部设定值	REAL	0	I		
SP_LLM	设定值下限	REAL	0	I	+	
SP_OP	手动设定值	REAL	0	IO	+	
SP_OP_ON	手动设定值 SP_ON 的操作员控制 使能	BOOL	1	IO		
SP_TRK_ON	1 = SP_OP 跟踪 PV	BOOL	0	I	+	
SPBUMPON	1 = 无波动设定值	BOOL	1	I	+	

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许 值
SPEXON_L	SP_EXT 的可互连输入, 1 = SP_EXT 处于激活状态	BOOL	0	I		
SPEXT_EN	1 = 外部设定值的操作员控制使能	BOOL	0	I		
SPEXTSEL_OP	操作员输入: 0 = 内部设定值, 1 = 外部设定值	BOOL	0	IO	+	
SPINT_EN	1 = 内部设定值的操作员控制使能	BOOL	0	I		
STEP_NO	BATCH 步号	DWORD	0	I	+	
SUBN1_ID	一级 DP 主站系统的 ID	BYTE	255	I		
SUBN2_ID	冗余 DP 主站系统的 ID	BYTE	255	I		
TD	微分时间 (秒)	REAL	0	I	+	0 或 \geq 1.0
TI	积分时间 (秒)	REAL	3000	I	+	0 或 \geq 0.5
TM_LAG	微分操作的时延 (s)	REAL	5	I	+	0 或 \geq 0.5
USTATUS	VSTATUS 中的状态字, 可进行用户特定的组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

和与 FB“PID_CS”相关的参数具有相同名称的参数也具有相同的含义。更多信息, 可参考手册《控制器模块 FM 355, 结构和组态》(Controller Module FM 355, Structuring and Configuring)。

3.10.15 FMCS_PID 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由此项抑制
1	QPERAF/QMODF	@6%d@/@7%d@/@8%d@ 控制器通道错误 @5%d@	S	-
2	QPARF_FM	@6%d@/@7%d@/@8%d@ 控制器通道组态错误 @5%d@	S	-
3			无消息	-
4			无消息	-
5	QH_ALM	\$\$BlockComment\$\$ 上上限报警	AH	M_SUP_AL、 MSG_LOCK
6	QH_WRN	\$\$BlockComment\$\$ 上限报警	WH	M_SUP_ER、 MSG_LOCK
7	QL_WRN	\$\$BlockComment\$\$ 下限报警	WL	M_SUP_ER、 MSG_LOCK
8	QL_ALM	\$\$BlockComment\$\$ 下下限报警	AL	M_SUP_AL、 MSG_LOCK

将关联值分配给块参数

消息块的前三个关联值分配给 SIMATIC BATCH 数据，第四个关联值保留以供过程值使用，第五个关联值供控制器通道号使用。其余的关联值可随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	PV
5	CHANNEL
6	SUBNET_ID
7	RACK_NO
8	SLOT_NO
9	AUX_PR09
10	AUX_PR10

说明

FM 355 可在 HW Config 中使用组态工具进行组态。除了 PCS 7 中的标准计数方法 0 到 n，此组态工具会从 1 到 n 计数模块上的控制器和信号通道。如果存在 FM 355 硬件错误，则在消息文本中显示的通道数会因此比组态工具中的数字小一。

实例:

12.10.2002 10.20 来源 FM 355 断线 AE 通道 02 输入状态 (12.10.2002 10.20 Origin FM 355 Wire Break AE Channel 02 Entering State)。

不过，这对应的是组态工具中模拟输入通道 3 的断线。

3.10.16 FMCS_PID 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	-	-	-	QSPINTON	QMAN_AUT	MSG_LOCK	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	-	-	-	-	-	-

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.10.17 FMCS_PID 的操作和监视

更多信息

更多信息，请参考以下部分：

- FMCS_PID 块图标 (页 692)
- FMCS_PID 面板 (页 610)

3.11 FMT_PID: 温度控制器块

3.11.1 FMT_PID 的描述

对象名 (类型 + 编号)

FB77

- FMT_PID 块 I/O (页 193)
- FMT_PID 块图标 (页 693)
- FMT_PID 面板 (页 620)

应用领域

FMT_PID 块用于集成 FM 355-2 温度控制模块。

它可用于 FM 355-2 C (C 控制器) 和 FM 355-2 S (S 和 P 控制器) 这两种模块类型。它本身不包含控制算法, 因为 PID 控制功能仅在模块上执行。可使用它来监视所有相关过程值以及更改所有相关控制器参数。有关 FM 355-2 的实例应用和 I/O 参数的详细描述, 可参考《温度控制器模块 FM 355》(*Temperature-Controller Module FM 355-2*) 手册。

控制器参数可通过模块上的“自动调节”功能来设置 (TUN_ON = TRUE)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 设置参数 CO_NO
- 输入 EN_CO 互连到 FM_CO 块的输出 EN_CO_x (x = 机架号)

输出 ENCO 连接到 FM_CO 块的输入 ENCOx_yy (x = 机架号, yy = 配位号)。

将 FM 355-2 C 控制器模块用作连续控制器 (C 控制器)

该块提供了下列显示和设置选项:

- 显示在模块上执行的针对过程值 PV 或误差信号 ER 的两个限制对 (QH_ALM、QH_WRN、QL_WRN、QL_ALM 输出) 的限制监视结果。MONERSEL 用于指定是监视 PV 还是监视 ER。
- 禁止在超出限制时生成各个消息
- 分割范围功能
- 误差信号分支中的死区 (DEADB_W, 打开阈值)
- PID 算法
- 通过 PFAC_SP < 1.0 衰减 P 动作对设定值的更改 (避免过冲; 优化的控制和错误响应)
- 调节变量跟踪
- 禁用积分作用
- 手动模式下的设定值跟踪 (SP = PV)

将 FM 355-2 S 控制器模块用作脉冲控制器

使用控制器模块作为脉冲控制器而不是连续控制器意味着无法实现分割范围控制。

可使用脉冲控制器生成脉冲调宽控制信号。此变量将转换为二进制输出信号, 以使脉冲宽度和已组态周期之间的比率与调节变量 LNM 的值相符。

将 FM 355-2 S 控制器模块用作步控制器

使用控制器模块作为步控制器而不是连续控制器会导致以下差异:

- 输出 QLMNR_ON 指示反馈信号是否可用 (1 = 存在, 0 = 不存在)。
- 无法进行分割范围操作。

作为不带位置反馈 (QLMNR_ON = 0) 的步操作控制器使用时, 只能在端位置对操纵的变量进行手动调整。在这种情况下, 安全位置 LMN_SAFE 或外部控制值 LMN_RE 由控制器模块按以下所述进行解释:

值 < 40 %: 完全关闭执行元件

值 > 60 %: 完全打开执行元件

40% ≤ 值 ≤ 60%: 保持当前设置

调用 OB

循环中断 OB: 例如 OB 32

对于启动检测, 该块还通过驱动程序生成器安装在 OB100 中。

还要注意 FM_CO 块的依赖性。

更多信息

更多相关信息, 可参考:

寻址 (页 178)

功能 (页 178)

通过过程映像采集过程值 (页 180)

生成设定值、限制、误差信号和调节变量 (页 182)

手动、自动和跟踪模式 (页 184)

模式更改 (页 186)

安全模式 (页 187)

将参数下载到模块 (页 187)

从模块中读取数据/使用组态工具 (页 188)

优化 (概述) (页 189)

在两个不同的 PID 参数集间切换 (页 190)

错误处理 (页 190)

启动特征、时间响应和消息响应 (页 191)

FM 355-2 的备份模式 (页 193)

FMT_PID 的 I/O (页 193)

FMT_PID 的消息文本和关联值 (页 200)

FMT_PID 的 VSTATUS (页 202)

归档过程值 (页 728)

操作和监视 FMT_PID (页 202)

3.11.2 寻址

属于实例的 FM 355 的控制器通道是通过其逻辑基础地址（在 HW Config 中设置）（LADDR 输入）和控制器通道号（CHANNEL 输入，有效值 = 1 到 4）进行寻址的。CHANNEL 允许的输入值为 0 到 3。对 CHANNEL 进行更改后必须将 ACC_MODE 设置为 TRUE。

通过 PCS 7 库中的 PCS 7 块监视 FM 355-2 模块。MODE 输入与 MOD_D1 块的 OMODE 输出互连。该块仅通过 FM 355-2 控制通道进行通讯。因此，输出 OMODE 低位字中的测量范围编码相互之间无关，可以将其设置为零。

3.11.3 FMT_PID 的功能

FMT_PID 块是温度控制器模块 (FM 355-2) 与 SIMATIC PCS 7 库的各个块之间的接口。它还可以与其它 SIMATIC S7 块互连。

该块与 FM 355-2 相互按异步方式操作。

所有相关过程变量和干扰变量都由模块提供，且只能由该块读取。该块还可以向控制器模块传送各种运行模式和设置。

该块可用于读取和写入 FM 355-2 操作员控制和控制参数。对该块中参数的每一项更改都将传递到模块。

将循环读取过程值（SP（来自 FM 的设定值）、ER（误差信号）、DISV（干扰）、LMN_A 和 LMN_B、PHASE、STATUS_H、STATUS_C、STATUS_D 以及 ZONE_TUN 除外）。不过，每 4 个周期过后才会对它们进行更新。更多信息，请参考“通过过程映像采集和写入过程值 (页 180)”。

参数 SP（来自 FM 的设定值）、ER（误差信号）、DISV（干扰变量）、LMN_A、LMN_B、PHASE、STATUS_H、STATUS_C、STATUS_D 和 ZONE_TUN 只能使用“读取数据记录”(SFC 59) 从 FM 355-2 中读取。

将为每个过程值 PV 和 LMN 生成一个质量代码，状态如下：

状态	质量代码
有效值	16#80
无效值	16#00

参数分配

FM 355-2 通常从该块中接收其参数。不过,也可绕过该块(例如,使用组态工具)。随后将自动更新 FMT_PID 的参数。这可确保 FM 355-2 中的参数与该块中的参数始终保持同步。

有些参数既可以使用组态工具指定,也可以在该块中设置。这两个参数集可能会有差异。可通过块的 SDB_SEL 输入避免此类冲突。设置 SDB_SEL = 0 用于指定模块接受来自参数分配工具以及来自块的这些参数。请注意,CPU 每次从 STOP 转移到 RUN 后都会将参数分配工具中的参数传送到模块。另一方面,每次块输入发生更改时也会将块的参数传送到模块。

SDB_SEL = 1 (但未在内部激活,因为模块固件不支持此功能)用于定义模块只接受来自功能块的参数,而不接受来自参数分配工具的参数。

说明

下载 RUN 模式下的 HW Config (CiR, HW Config in Run) 后,不会将块参数调整为与模块参数保持同步。

3.11.4 通过过程映像采集和写入过程值

读取过程值

通过过程映像周期性地读取过程值 (SP (FM 的设定值)、ER (误差)、DISV (干扰)、LMN_A 和 LMN_B、PHASE、STATUS_H、STATUS_C、STATUS_D 以及 ZONE_TUN 除外)。不过, 每 4 个周期过后才会对它们进行更新。通常, 在安装了 FMT_PID 的循环中断 OB 中进行更新。为达到更高的精确性, 可以在具有更高时钟频率的第二个循环中断 OB 中安装 FMT_PID。

重新启动后或将 ACC_MODE 设置为 1 时, 此块将检测所用的循环中断 (最多允许使用 2 个; 更多相关信息, 可参考“启动特征”)。

选择循环中断时必需考虑以下各情况:

- 已安装块的循环中断 OB 的运行速度必须在 25 ms 与 30 s 之间。
注: 切勿使用 OB 1!
- 必须在 HW Config 中为较快的 OB 设置过程映像分区 (TPA)。
- 如果块周期长于模块的周期时间, 则在 4 个块周期后 (在 OB 32 [1,000 ms] 的实例中为 4 s 之后) 才会通过块的过程映像来完成读取, 否则将在 4 个模块周期后完成。
- **实例:** 模块上有 4 个已处理的模拟输入, 模块的周期时间通常为 400 ms (请参见 FM 355 参数分配界面上的“模块参数”(Module parameters) 按钮)。例如, 如果块安装在 OB 32 (1,000 ms) 中, 可借助在 OB 33 (500 ms) 中进行附加安装来加快通过过程映像的读取速度。

写入过程值

每到第二个周期，就会将以下过程值写入过程映像中：

- 设定值（仅限自动模式）
- 调节变量（仅限手动模式）
- `SAFE_ON`、`LMNTRKON`、`LMN_REON`、`LMNRHSRE`、`LMNRLSRE`
- `LMNS_ON` 或 `TRUE`（如果由控制信号设置）
- `LMNUP` 或 `LMNUP_OP`
- `LMNDN` 或 `LMNDN_OP`

在以下情况下，必需重新启动或在编译和下载后必需将 `ACC_MODE` 设置为 1：

- 已将块移动到其它周期
- 也在快速周期中安装了块
- 还删除了安装的块

3.11.5 设定值、限制值、误差信号和调节变量的生成

通过 FMT_PID 块生成设定值

设定值 SP 可从三个不同的来源获取，具体取决于 SP_TRK_ON 和 SPEXTSEL_OP 输入。更多关于外部/内部设定值的信息，请参考“模式更改 (页 186)”：

SP_TRK_ON	SPEXTSEL_OP	SP=	状态
0	0	SP_OP	内部（操作员控制的）设定值
不相关	1	SP_EXT	外部设定值
1	0	PV **	跟踪设定值
		** 仅手动模式且 SPBUMPON = 1 时	

有效设定值限制在一定范围内 (SP_LLM, SP_HLM)。

设置 SP_TRK_ON 后，将在手动模式下对 SP_OP 设定值进行跟踪（适用于内部设定值且 SPBUMPON = 1 的情况）。这使得能够从手动模式无扰动切换到自动模式。

生成限值

这取决于 MONERSEL 的输入值，控制器模块既监视过程值 PV (MONERSEL = 0) 也监视误差信号 ER (MONERSEL = 1)，检查是否超过警告和报警极限值 (L_WRN、H_WRN、L_ALM、H_ALM)。将通过常用滞后量 HYS 进行监视。

在 QL_WRN、QH_WRN、QL_ALM 和 QH_ALM 输出处，块将提供监视结果。除非启用了消息抑制，监视过程值 PV 时，块会通知任何超出上限和下限的情况。

误差信号的生成

误差信号由控制器模块根据激活的设定值 SP 和过程值 PV 生成，并可在块的输出 ER 处获得。

死区 DEADB_W 到期后，将在 PID 算法中进一步处理误差信号。而不会添加干扰变量。

由 FMT_PID 块生成调节变量

可以通过各种数据源获取调节变量 LMN。如果同时将若干个控制输入设置为 TRUE，其优先级如下：

优先级	控制输入	LMN	状态
1	LMNS_ON = 1	LMNUP 或 LMNDN	步控制器的已互连控制信号
2	SAFE_ON = 1	= LMN_SAFE	安全模式
3	TUN_ON = 1 **	= LMN _旧 + TUN_DLMN	优化模式 (PHASE = 2)
4	LMNTRKON = 1	= 模块上的模拟输入或 0.0	跟踪模式
5	手动	= LMN_OP	手动模式 (QMAN_AUT=0)
6	LMN_REON = 1	= LMN_RE	外部调节变量
7	自动	= PID 算法	自动模式 (QMAN_AUT=1)

** 还需要通过设定值阶跃变化或在阶段 2 中设置 TUN_ST/TUN_CST = 1 来设置优化。

- 如果 LIOP_MAN_SEL = 0，则 AUT_ON_OP 参数将用于在 OS 上于手动模式和自动模式之间切换。
- 如果 LIOP_MAN_SEL = 1，则 AUT_L 参数将用于 CFC 中通过互连在手动模式和自动模式之间切换。

在步控制器中，在考虑电机特有参数：电机运行时间 (MOTOR_TM)、最短脉冲持续时间 (PULSE_TM) 和最短中断持续时间 (BREAK_TM) 的情况下，可以将调节变量转换为控制脉冲 (QLMNUP; QLMNDN)。

3.11.6 FMT_PID 的手动、自动和跟踪模式

手动模式

操纵变量由操作员在输入 LMN_OP 处设置（也可在此处使用点动模式）。切换到自动模式时，模块将接管“手动”设置为操作点的操纵变量。

“手动模式”的优先级高于“外部设定值”。

步控制器的手动模式:

可使用执行命令直接控制步控制器的控制信号。通过 LMNSOPON 启用信号操纵；在 LMNUP_OP 或 LMNDN_OP 处设置控制信号。只有该命令被禁用或到达最终位置时，才会驱动阀门。

操作员调整控制信号的优先级高于通过 LMN_OP 调整调节值的优先级。

使用无位置反馈的步控制器时，无法控制 LMN_OP 输入。

自动模式

操纵变量通过模块的 PID 算法进行计算。控制参数 GAIN、TI、TD 和 TM_LAG 可进行互连。

通过设置负比例 GAIN 值，可颠倒控制器的控制方向（上升误差信号会导致操纵变量下降）。

可通过设置 $TI = 0$ 禁用 I 作用。

可由操作员控制的操纵变量输入 LMN_OP 会跟踪 LMN 输出，以确保无波动地从自动模式切换到手动模式。

外部设定值 (LMN_RE)

该块会将值 LMN_RE 传送到 FM 355-2。如果设置 LNM_REON = 1，FM 355-2 会将外部操纵变量 LMN_RE 用作操纵变量 LMN。

“外部操纵变量”模式的优先级高于“自动模式”。

操纵变量跟踪

在操纵变量跟踪模式 (LMNTRKON = 1) 下，操纵变量会跟踪模块的模拟输入或值 0.0。

“跟踪”模式的优先级高于“手动模式”。

安全模式 (LMN_SAFE)

该块会将值 LMN_SAFE 传送到 FM 355-2。如果 SAFE_ON = 1，FM 355-2 会将安全操纵变量 LMN_SAFE 作为操纵变量 LMN 来接受。

“安全模式”的优先级高于“跟踪”模式。

步控制器的可互连控制信号设置

使用 LMNS_ON 且通过互连的 LMNUP 和 LMNDN 输入直接连接控制信号的跟踪模式在所有模式中具有最高优先级。如果设置了 LMNS_ON，则只能通过输入 LMNUP 或 LMNDN 设置控制信号。只要设置了 LMNS_ON，便可抑制对控制信号的任何其它影响。

3.11.7 FMT_PID 的模式更改

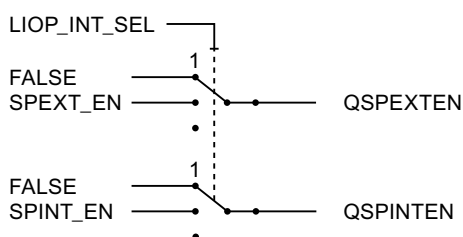
模式更改可由操作员或通过互连的输入触发。模式通过分配给模式的操作员控制块来更改。

外部/内部设定值

OS 操作员通过互连 SPEXON_L 设置 SPEXTSEL_OP 输入，依次来启动外部和内部设定值之间的切换。必须设置相应的使能输入 SPINT_EN、SPEXT_EN 或选择输入 LIOP_INT_SEL 来启用这些切换。

必须将 SP_OP_ON 设置为 TRUE 以启用操作员对设定值的控制。

启用内部和外部设定值之间的切换



QSPEXTEN = TRUE: 可将 SPEXTSEL_OP 从 FALSE (内部设定值) 更改为 TRUE (外部设定值)。

QSPINTEN = TRUE: 可将 SPEXTSEL_OP 从 TRUE (外部设定值) 更改为 FALSE (内部设定值)。

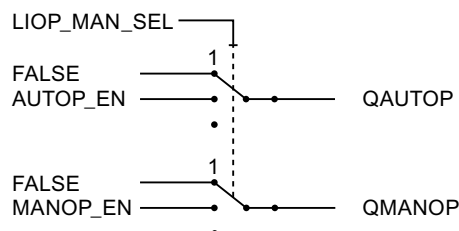
根据需要对 SPEXTSEL_OP 进行跟踪或复位。

手动/自动

OS 操作员通过设置 AUT_ON_OP 输入或互连 AUT_L 来切换操作模式。必须设置相应的使能输入 MANOP_EN、AUTOP_EN 或选择输入 LIOP_MAN_SEL 来启用该切换。

必须将 LMNOP_ON 设置为 TRUE 以启用手动变量控制。

启用手动和自动模式之间的切换



QAUTOP = TRUE: 可将 AUT_ON_OP 从 FALSE (手动模式) 更改为 TRUE (自动模式)。

QMANOP = TRUE: 可将 AUT_ON_OP 从 TRUE (自动模式) 更改为 FALSE (手动模式)。

根据需要对 SPEXTSEL_OP 进行跟踪或复位。

针对无波动切换的措施

为确保无波动切换, 针对下列模式采取了特殊措施:

- 外部设定值/跟踪设定值:
如果 SPBUMPON = TRUE, 内部设定值 SP_OP 将设置为有效 (外部或跟踪) 设定值。
- 自动模式、安全模式、跟踪模式或外部操纵变量 LMN_RE: 手动值 LMN_OP 跟踪有效的操纵变量。
- 手动模式、安全模式、跟踪模式或外部操纵变量 LMN_RE:
将对积分器进行跟踪, 以便能够无波动地切换到自动模式 (不通过步控制器)。使用组态工具可以禁用默认无波动切换功能。这样可以更快地更正误差信号。

3.11.8 安全模式

可互连的输入 SAFE_ON 用于设置安全模式。该模式由具有最高优先级的控制器模块执行。在安全模式下, 块的 LMN_SAFE 输入处的值在控制输出处输出。

3.11.9 将参数下载到模块

只要相应的块参数发生变更, 便会将通道特定的控制器和操作参数传送到控制器模块中。

将控制器和操作参数下载到控制器模块可能需要进行几次块调用。

3.11.10 从模块中读取数据/使用组态工具

通道特定的过程值由控制器模块来读取。更多相关信息，请参考“功能 (页 178)”。尤其是在分布式操作中，读取可能需要进行几次块调用。

例如，也可使用组态工具绕过块。随后模块将自动读取和更新 FMT_PID 的参数。这可确保 FM 355-2 中的参数与该块中的参数始终保持同步。

说明

请按照以下顺序进行操作：

- 使用菜单命令**上传到 PG (Upload to PG)** 将组态工具的数据管理与模块中有效的参数联系起来。
 - 在组态工具中，单击“下载到模块”(Download to module) 按钮。将写入参数。
 - 通过选择菜单命令**图表(Chart) > 读回...(Readback...)** 将已修改的参数传送到 CFC 的离线数据管理中。
-

3.11.11 优化

优化顺序

优化顺序如下:

1. 创建定态
2. 设置 PID_ON = TRUE (如果 PID 参数是必需的)
3. 组态 TUN_DLMN/TUN_CLMN
4. 设置 TUN_ON = TRUE (阶段 1, 优化就绪)
5. 使用设定值阶跃变化或通过设置 TUN_ST 开始优化

如果未发生任何组态错误, 则控制器优化此时应处于阶段 2, 且 STATUS_H 应为 0。

6. 到达拐点 (PHASE >= 3) 时, 评估 STATUS_H 参数处的诊断显示。对于过程类型 I, 几个周期后将到达阶段 0, 优化也将彻底结束。对于过程类型 II 和 III, 优化将进入阶段 7 (检查过程类型)。如果 STATUS_H > 20000, 则表明发生了评估错误或尚未到达拐点。在此状况下, 请重复该过程。

结果

一旦完成优化, 参数 PFAC_SP、GAIN、TI、TD、D_F、CON_ZONE 和 CONZ_ON 将立即更新 (在模块中和 FMT_PID 处均进行更新)。而且, PI 或 PID 参数集将保存在 FM 355-2 中。

有关优化步骤的更多详细信息, 请参考 FM 355-2 温度控制器模块手册。

已优化控制器参数的永久备份

- 保存、编译和下载硬件组态: 优化后的控制器参数此时存储于系统数据块 (SDB) 中。
- 通过选择菜单命令 **图表(Chart) > 读回...(Readback...)** 将已修改的参数传送到 CFC 的离线数据管理中。

3.11.12 在两个不同的 PID 参数集间切换

除了有效的 PID 参数集外，还会将另一个参数集存储在 FM 355-2 中。参数集使用 SAVE_PAR 保存，使用 UNDO_PAR 重置。这会影响以下参数：

PFAC_SP、GAIN、TI、TD、D_F、CON_ZONE、RATIOFAC、CONZ_ON 和 P_SEL

优化后，PI 和 PID 参数集都将保存在 FM 355-2 中。这些参数集通过设置 LOAD_PID 进行加载。如果 PID_ON = TRUE，则将 PID 参数集复制到有效的控制器参数，否则将加载 PI 参数集。这会影响以下参数：

GAIN、TI、TD 和 CON_ZONE

3.11.13 FMT_PID 的错误处理

该块提供以下错误显示：

错误显示	含义
QOP_ERR = 1	操作员输入错误。如果没有出现新的操作员输入错误，将在下一个块周期中复位 QOP_ERR。
QPARF_FM = 1	<ul style="list-style-type: none"> 使用参数分配工具直接向控制器模块分配参数时出错。 在块上设置了无效的控制通道号 (CHANNEL)。 在 OB 中安装时出错；更多相关信息，请参考“启动行为”。
QCH_F = 1	通道错误。由于硬件故障，属于实例的控制器通道无法返回有效结果。
QMODF = 1	控制器模块已移除或发生故障。
QPERAF = 1	I/O 访问错误。该块无法访问控制器模块。

3.11.14 FMT_PID 的启动行为、动态响应和消息响应

启动特征

在 CPU 启动期间，即 FM 启动或块初次运行时，将设置运行模式 MANUAL 和 INTERNAL，并将控制器参数从块写入到模块。

在 OB100 中，QDONE 设置为 0。

CPU 重新启动后或 ACC_MODE 设置为 1 时，块最多约 30 秒无法运行。

重新启动 (OB100) 或 ACC_MODE 设置为 1 后块将使用前 30 秒检测块实例是否安装在多个 OB 中。如果随后将块移动到其它 OB，则编译和下载后必须执行重新启动或必须将 ACC_MODE 设置为 1（否则将收到错误消息 QPARF_FM = 1，而且块将不会再提供任何数据）。

说明

可在 HW Config 中通过“保存/编译”(Save/Compile) 组态组态工具，然后“下载”(Download) 到 SDB（系统数据）。这些 SDB 参数可能与块参数有差异。SDB_SEL = 0 时，CPU 每次 STOP-RUN 转移时都会将 SDB 参数发送到模块。不过，块参数几个周期后才会写入到模块。SDB_SEL = 1 时，CPU 进行 STOP-RUN 转移时模块不会加载 SDB 参数。这会避免在 SDB 参数与块参数有差异时在操纵变量中发生跳转。

时间响应

不可用

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“FMT_PID 的 VSTATUS (页 202)”。

3.11 FMT_PID: 温度控制器块

消息响应

FMT_PID 块使用 ALARM_8P 块生成消息。

存在下列消息触发器:

- 监视过程值或系统偏差
- 如果无更高级别错误激活, 将出现模块访问错误

通过使用相关的 M_SUP_xx 输入可以分别抑制超出限制的消息。可使用 MSG_LOCK 集中禁用过程消息(不是控制系统消息!)。

如果自上次重新启动后未满足 RUNUPCYC 个周期数, 并且如果 MSG_LOCK = TRUE 或 MSG_STAT = 21, 则会设置 QMSG_SUP。

监视过程值

不可用

3.11.15 FM 355-2 的备份模式

CPU 发生故障或切换到 STOP 后, FM 355-2 将切换到备份模式。

3.11.16 FMT_PID 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	有效值
ACC_MODE	内部处理中包含 SUBN1_ID、 SUBN2_ID、RACK_NO、SLOT_NO 和 CHANNEL	BOOL	0	IO		
AUT_L	用于 MANUAL (手动) /AUTO (自动) 模式的可互连输入: 0 = 手动; 1 = 自动	BOOL	0	I		
AUT_ON_OP	操作员输入: MANUAL/AUTO: 0 = 手 动; 1 = 自动	BOOL	0	IO	+	
AUTOP_EN	1 = 自动模式的操作员控制使能信号	BOOL	1	I		
AUX_PRxx	用户特定的关联值 6 到 10	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 连续批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[3 2]	"	I	+	
BREAK_TM	最短中断时间 (秒)	REAL	0	IO		
CHANNEL	控制器通道号	INT	0	I		
CO_NO	用于读取数据记录的配位号	INT	0	O		
CON_ZONE	控制区	REAL	100	IO		
CONZ_ON	控制区开	BOOL	0	IO		
D_EL_SEL	D 操作输入	INT	0	IO		
D_F	微分因子	REAL	5	IO		
DEADB_W	死区宽度	REAL	0	IO	+	

3.11 FMT_PID: 温度控制器块

I/O (参数)	含义	数据类型	默认值	类型	OCM	有效值
DISV	干扰变量	REAL	0	O		
EN_CO	当前配位号	STRUCT				
ENCO	CFC 中的配位号“启用配位”	BOOL	0	IO		
ER	误差信号	REAL	0	O		
GAIN	比例增益	REAL	1	IO	+	
H_ALM	报警上限	REAL	100	IO	+	H_ALM > H_WRN >L_WRN > L_ALM
H_WRN	警告下限	REAL	95	IO	+	H_ALM > H_WRN >L_WRN > L_ALM
HYS	滞后量	REAL	1	IO	+	>= 0
L_ALM	报警下限	REAL	-5	IO	+	H_ALM > H_WRN >L_WRN > L_ALM
L_WRN	警告下限	REAL	-3	IO	+	H_ALM > H_WRN >L_WRN > L_ALM
LADDR	FM 355-2 的逻辑地址	INT	0	I		
LIOP_INT_SEL	1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
LIOP_MAN_SEL	1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
LMN	调节变量	REAL	0	O	+	
LMN_A	分割范围函数/位置反馈的调节变量 A	REAL	0	O		
LMN_B	分隔范围函数的调节变量 B	REAL	0	O		
LMN_HLM	调节值上限	REAL	100	IO	+	
LMN_LLM	调节值下限	REAL	0	IO	+	
LMN_OP	调节变量的操作员输入	REAL	0	IO	+	

I/O (参数)	含义	数据类型	默认值	类型	OCM	有效值
LMN_RE	外部调节变量	REAL	0	I		
LMN_REON	激活外部调节变量	BOOL	0	I		
LMN_SAFE	安全调节值	REAL	0	IO	+	
LMNDN	已互连的控制信号低	BOOL	0	I		
LMNDN_OP	控制信号下限操作	BOOL	0	IO		
LMNOP_ON	1 = 操作员可输入 LMN_OP	BOOL	1	IO		
LMNRHSRE	位置反馈的上限位停止信号	BOOL	0	I		
LMNRLSRE	位置反馈的下限位停止信号	BOOL	0	I		
LMNS_ON	互连的控制信号开启 (LMNDN、LMNUP)	BOOL	0	I		
LMNSOPON	激活控制信号的操作员控制使能	BOOL	0	IO		
LMNTRKON	跟踪 (通过模拟输入跟踪调节变量)	BOOL	0	I		
LMNUP	已互连的控制信号高	BOOL	0	I		
LMNUP_OP	可调整的控制信号高	BOOL	0	IO		
LOAD_PID	加载优化的 PI/PID 参数	BOOL	0	IO		
M_SUP_AH	1 = 消息抑制报警上限	BOOL	0	I	+	
M_SUP_AL	1 = 消息抑制报警下限	BOOL	0	I	+	
M_SUP_WH	1 = 消息抑制警告上限	BOOL	0	I	+	
M_SUP_WL	1 = 消息抑制警告下限	BOOL	0	I	+	
MANOP_EN	1 = MANUAL 的操作员控制使能信号	BOOL	1	I		
MO_PVHR	显示上限 (测量范围)	REAL	110	I	+	
MO_PVLR	显示下限 (测量范围)	REAL	-10	I	+	
MODE	模式	DWORD	0	I		
MONERSEL	监视: 0 = 过程值, 1 = 误差信号	BOOL	0	IO		
MSG_ACK	消息已确认	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		
MSG_LOCK	1 = 消息抑制取决于过程状态	BOOL	0	I	+	
MSG_STAT	消息错误信息	WORD	0	O		
MTR_TM	电机运行时间 (秒)	REAL	60	IO		

3.11 FMT_PID: 温度控制器块

I/O (参数)	含义	数据类型	默认值	类型	OCM	有效值
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
P_SEL	激活 P 动作	BOOL	1	IO		
PFAC_SP	比例增益	REAL	1	IO		
PHASE	自动调节阶段 (0.....7)	INT	0	O		
PID_ON	1 = 激活 PID 模式	BOOL	0	IO		
PULSE_TM	最短脉冲持续时间 (秒)	REAL	0	IO		
PV	过程值	REAL	0	O	+	
Q_SP_OP	1 = 设定值的操作员控制使能已激活	BOOL	0	O	+	
QAUTOP	1 = AUTO 的操作员控制使能已激活	BOOL	0	O	+	
QC_LMN	LMN 的质量代码	BYTE	16#80	O		
QC_PV	PV 的质量代码	BYTE	16#80	O		
QCH_F	通道错误	BOOL	0	O		
QDNRLM	负设定值斜坡限制已触发	BOOL	0	O		
QDONE	1 = 参数已读取	BOOL	0	O		
QERR	ENO 的取反值	BOOL	1	O		
QH_ALM	触发报警上限	BOOL	0	O		
QH_WRN	触发警告上限	BOOL	0	O		
QL_ALM	触发报警下限	BOOL	0	O		
QL_WRN	触发警告下限	BOOL	0	O		
QLMN_HLM	调节变量上限已触发	BOOL	0	O	+	
QLMN_LLM	调节变量下限已触发	BOOL	0	O	+	
QLMN_RE	1 = 外部调节值已激活	BOOL	0	O		
QLMNDN	调节变量信号低	BOOL	0	O		
QLMNOP	1 = 用于设置调节变量的操作员控制使能信号已激活	BOOL	0	O	+	
QLMNR_HS	位置反馈的上限位停止信号	BOOL	0	O		
QLMNR_LS	位置反馈的下限位停止信号	BOOL	0	O		
QLMNR_ON	位置反馈已激活	BOOL	0	O		

I/O (参数)	含义	数据类型	默认值	类型	OCM	有效值
QLMNS_ON	1 = LMNS_ON 已激活	BOOL	0	O		
QLMNSAFE	安全模式	BOOL	0	O		
QLMNSOP	控制信号的操作员控制使能已激活	BOOL	1	O	+	
QLMNTRK	跟踪模式	BOOL	0	O		
QLMNUP	控制信号高	BOOL	0	O		
QLMNVOP	调节变量 LMN_OP 的操作员控制使能信号已激活	BOOL	1	O	+	
QMAN_AUT	0 = 手动, 1 = 自动	BOOL	0	O	+	
QMAN_FC	1 = 跟踪模式或从控制器处的防复位终止	BOOL	0	O		
QMANOP	1 = 手动模式的操作员控制使能信号已激活	BOOL	0	O	+	
QMODF	1 = 模块错误	BOOL	0	O		
QMSG_ERR	1 = 消息错误	BOOL	0	O	+	
QMSG_SUP	1 = 消息抑制激活	BOOL	0	O	+	
QOP_ERR	1 = 组操作员输入错误	BOOL	0	O		
QPAR_ACT	1 = 更新控制器参数	BOOL	0	O		
QPARF_FM	1 = 直接模块参数分配错误或 CHANNEL 故障	BOOL	0	O		
QPERAF	1 = I/O 访问错误	BOOL	0	O		
QSP_HLM	1 = 设定值上限已触发	BOOL	0	O		
QSP_LLM	1 = 设定值下限已触发	BOOL	0	O		
QSPEXTEN	1 = 外部模式的操作员控制使能信号已激活	BOOL	0	O	+	
QSPEXTON	外部设定值已激活	BOOL	0	O		
QSPINTEN	1 = 内部模式的操作员控制使能信号已激活	BOOL	0	O	+	
QSPR	分割范围模式	BOOL	0	O		
QSTEPCON	1 = 步控制器	BOOL	0	O		
QTUN_ON	1 = 调节正在运行	BOOL	0	O		
QUPRLM	1 = 设定值斜率限制已触发	BOOL	0	O		

3.11 FMT_PID: 温度控制器块

I/O (参数)	含义	数据类型	默认值	类型	OCM	有效值
RACK_NO	机架号	BYTE	255	I		
RATIOFAC	比率因子	REAL	0	IO		
RET_VALU	SFC 58/59 的返回值 (WR_REC/RD_REC)	WORD	0	O		
RUNUPCYC	起动周期数	INT	3	I		
SAFE_ON	采用安全位置	BOOL	0	I		
SAVE_PAR	保存控制参数	BOOL	0	IO		
SDB_SEL	1 = CPU 进行 STOP-RUN 转换时模块不采用 SDB 参数	BOOL	1	I	+	
SLOT_NO	插槽号	BYTE	255	I		
SP	设定值	REAL	0	O	+	
SP_EXT	外部设定值	REAL	0	I		
SP_HLM	设定值上限	REAL	100	IO	+	
SP_LLM	设定值下限	REAL	0	IO	+	
SP_OP	手动设定值	REAL	0	IO	+	
SP_OP_ON	手动设定值 SP_ON 的操作员控制使能	BOOL	1	I		
SP_TRK_ON	1 = SP_OP 跟踪 PV	BOOL	0	I	+	
SPBUMPON	1 = 无波动设定值	BOOL	1	I	+	
SPEXON_L	SP_EXT 的可互连输入, 1 = SP_EXT 处于激活状态	BOOL	0	I		
SPEXT_EN	1 = 外部设定值的操作员控制使能	BOOL	1	I		
SPEXTSEL_OP	操作员输入 0: 内部设定值, 1 = 外部设定值	BOOL	0	IO	+	
SPINT_EN	1 = 内部设定值的操作员控制使能	BOOL	1	I		
STATUS_C	冷却调节的状态	INT	0	O		
STATUS_D	控制器设计的状态	INT	0	O		
STATUS_H	加热调节的状态	INT	0	O		
STEP_NO	BATCH 步号	DWORD	0	I	+	0 或 >= 1.0
SUBN1_ID	一级 DP 主站系统的 ID	BYTE	255	I		

I/O (参数)	含义	数据类型	默认值	类型	OCM	有效值
SUBN2_ID	冗余 DP 主站系统的 ID	BYTE	255	I		
TD	微分时间 (秒)	REAL	0	IO		0 或 \geq 0.5
TI	积分时间 (秒)	REAL	3,000	IO		
TUN_CLMN	用于冷却调节的 delta 调节值	REAL	-20	IO		
TUN_CST	启动冷却调节	BOOL	0	IO		
TUN_DLMN	用于过程激励的 delta 调节值	REAL	20	IO		
TUN_ON	启用控制器调节	BOOL	0	IO		
TUN_ST	启动控制器调节	BOOL	0	IO		
UNDO_PAR	撤消控制器参数更改	BOOL	0	IO		
USTATUS	VSTATUS 中的状态字, 可进行用户特定的组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	
ZONE_TUN	分组在一个区中, 用于并行调节的控制器通道	WORD	0	O		

FM 355-2 参数的更多相关信息, 可参考《温度控制器 FM 355-2》(Temperature Controller FM 355-2) 手册。

3.11.17 FMT_PID 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由此项抑制
1	QPERAF/QMODF	@6%d@/@7%d@/@8%d @ 控制器通道错误 @5%d@	S	-
2	QPARF_FM	@6%d@/@7%d@/@8%d @ 控制器通道组态错误 @5%d@	S	-
3			无消息	-
4			无消息	-
5	QH_ALM	\$\$BlockComment\$\$ 上上限报警	AH	M_SUP_AL、 MSG_LOCK,
6	QH_WRN	\$\$BlockComment\$\$ 上限报警	WH	M_SUP_ER、 MSG_LOCK,
7	QL_WRN	\$\$BlockComment\$\$ 下限报警	WL	M_SUP_ER、 MSG_LOCK,
8	QL_ALM	\$\$BlockComment\$\$ 下下限报警	AL	M_SUP_AL、 MSG_LOCK

将关联值分配给块参数

将为消息块的前三个关联值分配 SIMATIC BATCH 数据，保留第四个关联值以便供过程值使用，保留第五个关联值供控制器通道 ID 使用。其余关联值可随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	PV
5	CHANNEL
6	SUBNET_ID
7	RACK_NO
8	SLOT_NO
9	AUX_PR09
10	AUX_PR10

3.11 FMT_PID: 温度控制器块

3.11.18 FMT_PID 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	-	-	-	QSPEXTON	QMAN_AUT	MSG_LOCK	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	-	-	-	-	-	-

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.11.19 FMT_PID 的操作和监视

更多信息

更多相关信息，请参考以下部分：

- FMT_PID 块图标 (页 693)
- FMT_PID 面板 (页 620)

3.12 GAIN_SHD - 增益调整

3.12.1 GAIN_SHD 的描述

对象名 (类型 + 编号)

FB 141

- GAIN_SHD 块 I/O (页 207)

GAIN_SHD 的应用领域

该块用于下列应用:

- 以下项的参数值调整
 - 控制器增益
 - 积分作用时间
 - 微分作用时间
- 非线性过程当前运行点的连续调整

工作原理

如果过程由于在不同运行点存在非线性响应而需要不同的 PID 控制器参数, 则可以在 GAIN_SHD 块中以表格 (“时间表”) 的形式为多达三个的不同运行点存储最优参数设置。当前运行点由可连续测量的变量 X 表示, 通常由控制器本身的过程值表示。此块确保为控制器提供每个运行点 $X_{(i)}$ 的最优参数 $GAIN_{(i)}$ 、 $TI_{(i)}$ 和 $TD_{(i)}$ 。

如果过程位于两个运行点之间, 则由两个最近运行点的最优值之间的线性插值计算参数。当过程从一个运行点移动到另一个运行点时, 这允许平稳、连续地调整控制器参数。

此块应视为 PID 控制器的补充功能, 以改进非线性过程中 PID 控制器的控制性能。使用 “GAIN_SHD” 按钮, 可以从相应 PID 控制器的参数视图中调用 GAIN_SHD 面板。

在内部，GAIN_SHD 块实际上由 POLYG_P 块的三个实例组成。与所有其它功能块相比，该块作为 CFC 图表进行实现并通过“编译图表为块类型”功能生成。为源图表“fbGAIN_SHD”提供库，这样将有多个选项供您选择：

- 如果标准功能满足您的需要，您可以使用库中预编译的功能块 GAIN_SHD。
- 如果在应用中需要增益调整的其它特殊功能（例如运行点多于三个，用于选择参数的其它逻辑功能），您将需要修改 CFC 源图表并将其编译为具有不同 FB 编号的块类型。

组态

GAIN_SHD 块置于与分配给控制器的相同的 CFC 图表中，并如相应模板中所示与该控制器互连：将输出 Link2Gain、Link2TI、Link2TD 和 Link2TM_LAG 连接到 PID 控制器的输入 GAIN、TN、TV 和 TM_LAG。为 GAIN_SHD 的 X 输入提供运行点的测量值，通常提供与控制器的 PV_IN 相同的值。

通过命名约定将 CFC 中的 GAIN_SHD 实例连接到所分配控制器的实例：CFC 图表中 GAIN_SHD 实例的名称与 PID 实例的名称完全相同，只是后面添加了“_gsc”。

实例：

这意味着监视块 TIC501_gsc 应属于控制器 TIC501。控制器面板中的“GAIN_SHD”按钮仅对其上具有遵守此命名约定的增益调整块的控制器可见。

要指定增益调整的参数，请在每个预期运行点上运行单独的控制器优化（例如使用 PID 调节器等工具）。使用尽可能小的振幅来激活过程，以在进行研究的运行点区域中捕获近似线性响应。将由 PID 调节器计算得出的最优参数值输入至 GAIN_SHD 块的表格中属于该运行点的相关行。GAIN_SHD 面板的标准视图中清晰地显示了此表格。请确保数字值也永久存储在工程系统的数据管理中，为此，可将参数的数字值从 AS 回读到 ES 或在 CFC 块的输入中手动输入这些值。

注: 批生产过程的增益调整。

增益调整的典型应用区域是批生产过程, 与连续过程不同, 批生产过程无法围绕固定运行点进行线性化, 因为批生产过程中增益计划需要在不同的运行点之间前后移动。此处有三种应用方案:

- 控制器参数取决于代表运行点的单个可连续测量的变量, 例如反应釜温度。这是 GAIN_SHD 块的常规使用情况: 控制器参数的管理在该块中处理并与批生产配方无关。
- 控制器参数取决于代表运行点的可连续测量的变量, 但还取决于反应中所使用的材料。然后可以在配方中锚定增益调整的合适参数集并通过 SIMATIC BATCH 将其传送给 GAIN_SHD 块。
- 控制器参数仅取决于批生产的当前阶段。然后可将这些参数直接从批生产包写入 PID 控制器且不需要任何增益调整。此方法的缺点是从一个阶段向另一个阶段转移时控制器参数中有波动。转移时应将控制器暂时切换到手动模式, 以避免调节变量中的波动。
- 配方仅指定当前需要 GAIN_SHD 块中控制器参数集 1 到 3 中的哪一个。但未在配方中锚定参数的数字值。此时, 可以将 GAIN_SHD 块的输入参数 X 用作所需数据记录的编号并通过配方对其进行分配, 而不将其与可测量过程值链接。在这种情况下, X 仅有三个固定值, 并且因为未使用 GAIN_SHD 块的插值功能, 所以必须采取预防措施来防止由于上述波动而导致控制器参数发生更改。

通常, 不必管理 GAIN_SHD 块中的批生产参数 (批生产号、批生产名称等), 因为该块未生成任何单独的消息并且与知道批生产参数的控制器块始终存在 1:1 的关系。

启动特征

该块没有任何启动特征。

时间响应

该块没有任何时间响应。

消息响应

该块没有任何消息响应。

参见

GAIN_SHD 的功能 (页 206)

3.12.2 GAIN_SHD 的功能

GAIN_SHD 的功能

该块提供以下功能。

- 控制器参数的手动设置

控制器参数的手动设置

对于没有增益调整的控制器，控制器参数的数字值可在控制器面板的参数视图中进行更改。然而当使用 GAIN_SHD 块时，参数被互连并因此在面板中无法再进行访问。如果要手动输入控制器参数，而不依靠增益调整表中指定的值，则在 GAIN_SHD 面板的标准视图的“手动分配参数”(Manual parameter assignment) 中选中复选标记。输入域中的参数值、增益系数、积分时间和微分时间被传递给控制器。

参见

GAIN_SHD I/O (页 207)

GAIN_SHD 的描述 (页 203)

3.12.3 GAIN_SHD I/O

GAIN_SHD I/O

输入

I/O (参数)	描述	数据类型	默认值	属性
Gain1	运行点 1 的 PID 增益	REAL	1	S7_m_c:='true' S7_link := 'true'
Gain2	运行点 2 的 PID 增益	REAL	1	S7_m_c:='true' S7_link := 'true'
Gain3	运行点 3 的 PID 增益	REAL	1	S7_m_c:='true' S7_link := 'true'
GainOp	PID 增益: 手动模式的输入	REAL	1	S7_m_c:='true' S7_link := 'true'
ManParOn	1 = 手动模式下输入 PID 参数 0 = 自动模式下规划增益	BOOL	0	S7_m_c:='true'
TD_Op	PID 微分时间为常量 [s]: 操作员手动输入	REAL	0	S7_m_c:='true'
TD1	运行点 1 的 PID 微分时间为常量 [s]	REAL	0	S7_m_c:='true' S7_link := 'true'
TD2	运行点 2 的 PID 微分时间为常量 [s]	REAL	0	S7_m_c:='true' S7_link := 'true'
TD3	运行点 3 的 PID 微分时间为常量 [s]	REAL	0	S7_m_c:='true' S7_link := 'true'
TI_Op	PID 的积分时间 [s]: 操作员手动输入	REAL	10	S7_m_c:='true'
TI1	PID 的积分时间 [s]: 用于运行点 1	REAL	10	S7_m_c:='true' S7_link := 'true'

3.12 GAIN_SHD - 增益调整

I/O (参数)	描述	数据类型	默认值	属性
TI2	PID 的积分时间 [s]: 用于运行点 2	REAL	10	S7_m_c:='true' S7_link := 'true'
TI3	PID 的积分时间 [s]: 用于运行点 3	REAL	10	S7_m_c:='true' S7_link := 'true'
X	过程值定义运行点	REAL	0	S7_m_c := 'false' S7_link := 'true'
X1	X 的运行点 1 (插值点)	REAL	0	S7_m_c:='true'
X2	X 的运行点 2 (插值点)	REAL	50	S7_m_c:='true'
X3	X 的运行点 3 (插值点)	REAL	100	S7_m_c:='true'

输出

I/O (参数)	描述	数据类型	默认值	属性
Link2Gain	计算的控制器增益	REAL	1	S7_m_c := 'false' S7_link := 'true'
Link2TD	计算的微分作用时间	REAL	0	S7_m_c := 'false' S7_link := 'true'
Link2TI	计算的积分作用时间	REAL	10	S7_m_c := 'false' S7_link := 'true'

参见

GAIN_SHD 的功能 (页 206)

GAIN_SHD 的描述 (页 203)

3.13 INT_P: 积分 (Integration)

3.13.1 INT_P 的描述

对象名 (类型 + 编号)

FB40

- INT_P 块 I/O (页 212)

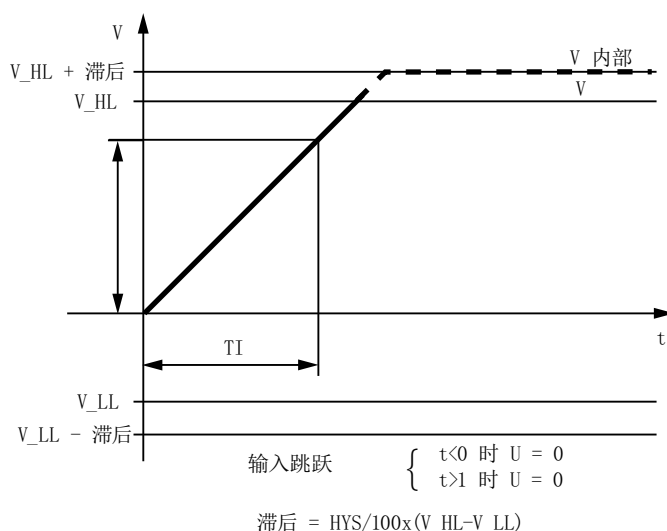
功能

构成所连接的输入值的时间积分:

$$v(s) = 1 / (TI * s) * u(s)$$

工作原理

块 INT_P 的工作方式是, 根据梯形规则按采样间隔 (SAMPLE_T) 来求和。结果 V internal 位于范围 V_HL - hyst 到 V_LL + hyst 之间, 如下图所示。附加值范围限制 (V_LL 与 V_HL 之间) 后, 该值随后将写入到输出值 V 中。



INT_P 的跳转响应

此外, 将监视内部结果 V internal, 看其是否超出极限 V_LL 和 V_HL, 并通过布尔型输出值 QVLL 和 QVHL 进行显示, 如图中所示。

调用 OB

安装了该块的循环中断 OB (例如 OB32) 和 OB100。

错误处理

除了可以通过操作系统识别的错误外, 还由块算法通过 $ENO = 0$ 和 $QERR = 1$ 来表示以下组态错误:

- $V_{LL} \geq V_{HL}$ ($V = 0$)
- $SAMPLE_T \leq 0$ (使用替代值 1 继续在内部进行计算)
- 滞后量 $HYS < 0$ (使用替代值 1 继续在内部进行计算)

当值在 V_{LL} 和 V_{HL} 限制范围内时, 如果 $TI = 0$, 则 $V = 0$, 否则 $V =$ 限制值。

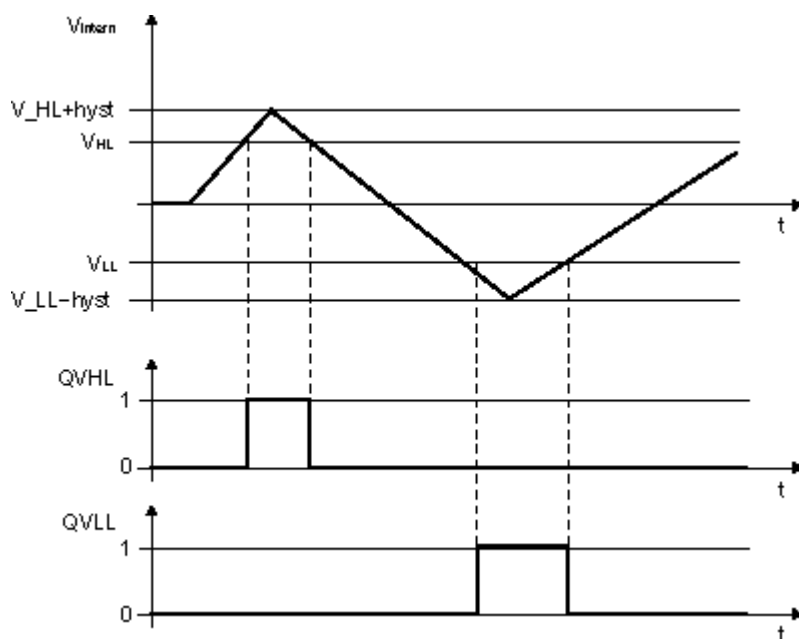


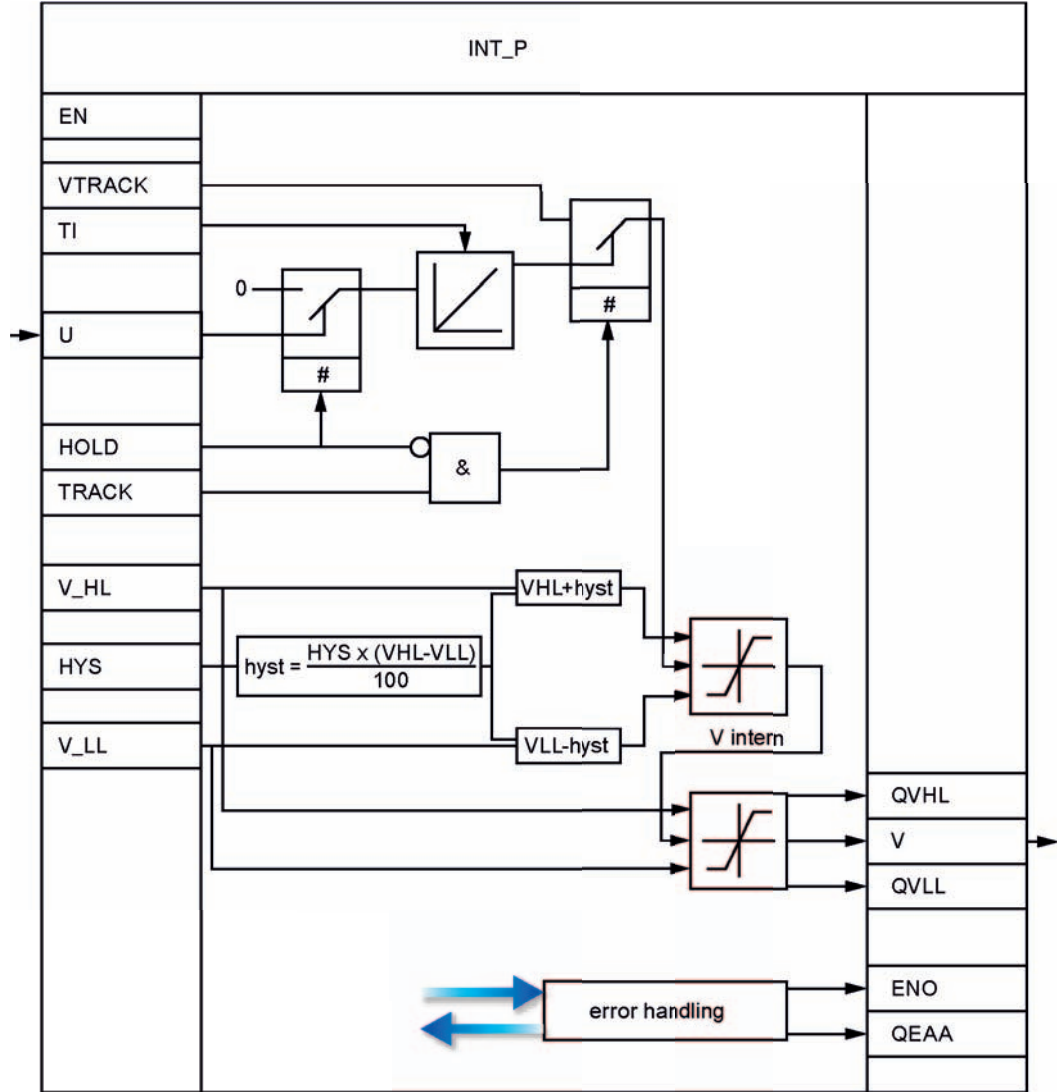
图 3-1 INT_P 的限制监视

启动特征

如果输入参数 $RESET = TRUE$, 将在启动期间复位内部历史过程数据和输出 V 。如果 $RESET = FALSE$, 将保留上一个值。这意味着还要必须在启动 OB (OB 100) 中调用该块。

时间响应

块必需安装在循环中断 OB 中。



INT_P 结构

3.13 INT_P: 积分 (Integration)

3.13.2 INT_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认 值	类型	允许值
HOLD	1 = 保持积分 (优先于 TRACK)	BOOL	0	I	
HYS	V_INTERN 的滞后量 ([%])	REAL	1	I	≥ 0
QERR	1 = 错误	BOOL	1	O	
QVHL	1 = 输出值上限	BOOL	0	O	
QVLL	1 = 输出值下限	BOOL	0	O	
RESET	1 = RESET (重新启动)	BOOL	1	I	
SAMPLE_T	采样时间 (秒)	REAL	1	I	> 0
TI	积分时间 (秒)	REAL	1	I	≥ 0
TRACK	1 = 跟踪	BOOL	0	I	
U	输入值	REAL	0	I	
V	输出值	REAL	0	O	
V_HL	V 的上限	REAL	100	I	V_HL > V_LL
V_LL	V 的下限	REAL	0	I	V_LL < V_HL
VTRACK	跟踪值	REAL	0	I	

3.14 INTERLOK: 状态显示锁

3.14.1 INTERLOK 的描述

对象名 (类型 + 编号)

FB 75

- INTERLOK 块 I/O (页 215)
- INTERLOK 块图标 (页 694)
- INTERLOK_Standard_View 面板 (页 626)

功能

INTERLOK 块用于实现一个可在 OS 上调用的标准互锁显示。可为该块分配最多 10 个输入信号, 每个信号都可根据需要取反。

工作原理

前五个输入 I1_1 到 I1_5 形成一个组。每个信号可通过设置对应的输入 NEG1_1 至 NEG1_5 来进行直接逻辑链接或者先取反再链接。

第一组逻辑操作的类型在 AND_OR1 参数中设置。NEGRES_1 = 1 通过 AND_OR3 将用于形成 Q 的 Q1 的结果取反。但输出 Q1 并不取反。

第一组的规则也同样适用于第二组的五个输入。
两组的结果可通过 AND/OR 运算进行逻辑链接。

如果输入 OVERWRITE = 1, 则互锁处于激活状态时 (Q = 1), 可将输出 Q 设置为 0。仅当 OVERW_EN = 1 时, 才可进行此操作。如果 OVERW_EN = 0 或不满足互锁条件, 则输入 OVERWRITE = 0。在输出中 Q_OVERWR = 1 表示输出 Q 已被覆盖。

仅在输入 CHECK_EN = TRUE 时, 下列内容才适用:

输出参数 FIRST_I 包含最初为 TRUE 或取反的 FALSE 的输入 Ix 编号 (1 到 10)。如果同时设置多个条件, 则会在 FIRST_I 中输入最小编号。如果上述条件无一满足, 那么输入 RESET 处的正沿会将 FIRST_I 设置为零。通常, 输出 Q 与 RESET 互连。

调用 OB

调用 OB 位于同一 OB 中, 并且处于最后一个其信号将在 INTERLOK 中显示的块之后。

3.14 INTERLOK: 状态显示锁

错误处理

仅通过操作系统

启动特征

无需特殊措施。

时间响应

该块没有时间响应。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“INTERLOK 的 VSTATUS (页 217)”。

消息响应

不可用

监视过程值

不可用

更多信息

更多相关信息，可参考：
INTERLOK 的 I/O (页 215)
INTERLOK 的 VSTATUS (页 217)
操作和监视 INTERLOK (页 217)

3.14.2 INTERLOK 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的详细信息, 可参考“关于块描述的常规信息 (页 17)”部分。

I/O (参数)	含义	数据类型	默认值	类型	OCM
AND_OR1	1 = AND, 0 = OR, 第一组	BOOL	0	I	+
AND_OR2	1 = AND, 0 = OR, 第二组	BOOL	0	I	+
AND_OR3	1 = AND, 0 = OR, 两个组	BOOL	0	I	+
CHECK_EN	1 = FIRST_I 检查使能端	BOOL	0	I	+
FIRST_I	首个为 TRUE (FALSE 取反) 的输入信号	BYTE	16#00	O	+
I1_1	输入信号 1, 第一组	BOOL	0	I	+
I1_2	输入信号 2, 第一组	BOOL	0	I	+
I1_3	输入信号 3, 第一组	BOOL	0	I	+
I1_4	输入信号 4, 第一组	BOOL	0	I	+
I1_5	输入信号 5, 第一组	BOOL	0	I	+
I2_1	输入信号 1, 第二组	BOOL	0	I	+
I2_2	输入信号 2, 第二组	BOOL	0	I	+
I2_3	输入信号 3, 第二组	BOOL	0	I	+
I2_4	输入信号 4, 第二组	BOOL	0	I	+
I2_5	输入信号 5, 第二组	BOOL	0	I	+
NEG1_1	1 = I1_1 取反	BOOL	0	I	+
NEG1_2	1 = I1_2 取反	BOOL	0	I	+
NEG1_3	1 = I1_3 取反	BOOL	0	I	+
NEG1_4	1 = I1_4 取反	BOOL	0	I	+
NEG1_5	1 = I1_5 取反	BOOL	0	I	+
NEG2_1	1 = I2_1 取反	BOOL	0	I	+
NEG2_2	1 = I2_2 取反	BOOL	0	I	+
NEG2_3	1 = I2_3 取反	BOOL	0	I	+

3.14 INTERLOK: 状态显示锁

I/O (参数)	含义	数据类型	默认值	类型	OCM
NEG2_4	1 = I2_4 取反	BOOL	0	I	+
NEG2_5	1 = I2_5 取反	BOOL	0	I	+
NEGRES_1	1 = 将第一组的结果取反	BOOL	0	I	+
NEGRES_2	1 = 将第二组的结果取反	BOOL	0	I	+
OVERW_EN	1 = OVERWRITE 使能信号	BOOL	0	I	+
OVERWRITE	1 = 覆盖	BOOL	0	IO	+
Q	输出信号	BOOL	0	O	+
Q_OVERWR	1 = 覆盖 Q	BOOL	0	O	+
Q1	中间结果, 第一组	BOOL	0	O	+
Q2	中间结果, 第二组	BOOL	0	O	+
QC_I1_1	I1_1 的质量代码	BYTE	16#80	I	
QC_I1_2	I1_2 的质量代码	BYTE	16#80	I	
QC_I1_3	I1_3 的质量代码	BYTE	16#80	I	
QC_I1_4	I1_4 的质量代码	BYTE	16#80	I	
QC_I1_5	I1_5 的质量代码	BYTE	16#80	I	
QC_I2_1	I2_1 的质量代码	BYTE	16#80	I	
QC_I2_2	I2_2 的质量代码	BYTE	16#80	I	
QC_I2_3	I2_3 的质量代码	BYTE	16#80	I	
QC_I2_4	I2_4 的质量代码	BYTE	16#80	I	
QC_I2_5	I2_5 的质量代码	BYTE	16#80	I	
QC_Q	Q 的质量代码	BYTE	16#80	O	
QC_Q_I	输出 Q 的质量代码	BYTE	16#80	I	
RESET	正沿 = 复位 FIRST_I	BOOL	0	I	+
USTATUS	VSTATUS 中的状态字, 可进行用户特定的组态	WORD	0	I	
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+

3.14.3 INTERLOK 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	-	-	-	I1_5	I1_4	I1_3	I1_2	I1_1
位号:	15	14	13	12	11	10	9	8
参数	-	-	Q	I2_5	I2_4	I2_3	I2_2	I2_1

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.14.4 INTERLOK 的操作员监控

更多信息

更多相关信息，请参考以下部分：

- INTERLOK 块图标 (页 694)
- INTERLOK 面板 (页 626)

3.15 LIMITS_P: 限制

3.15.1 LIMITS_P 的描述

对象名 (类型 + 编号)

FB41

- LIMITS_P 块 I/O (页 220)

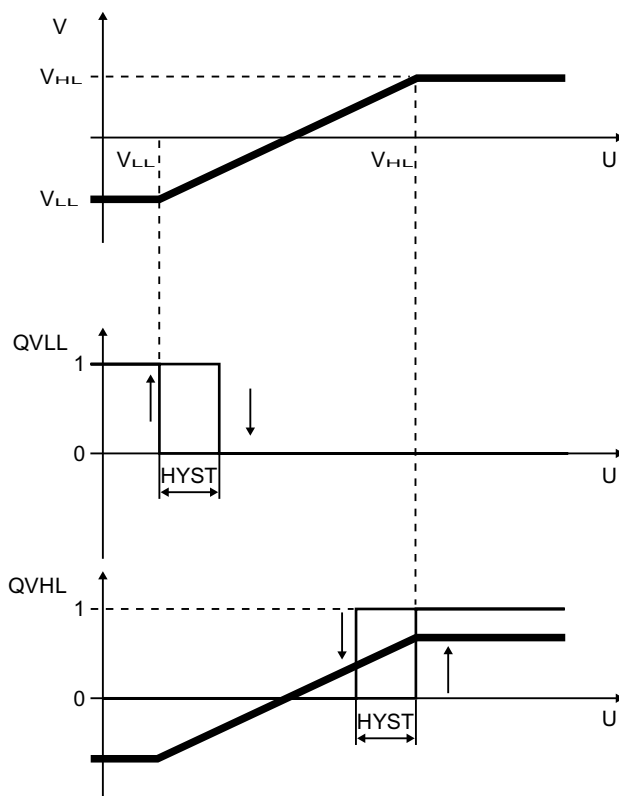
功能

LIMITS_P 块用于将模拟变量限定在可调整范围内。

工作原理

只要模拟输入值 U 在设置的限制范围内，块 LIMITS_P 就会将其传递给输出 V 。

- 如果该值超出下限，将输出限制值下限。如果该值超出上限，将输出上限值。
- 有效限制在二进制输出集上显示。输入值在限制值左右波动时，可设置滞后量，以避免显示抖动。



LIMITS_P 的工作原理

调用 OB

在其中安装了该块的 OB。

错误处理

如果 $V_{HL} - V_{LL} \leq HYS$ ，QVHL 和 QVLL 可同时为 1。没有完成 V_{LL} 与 V_{HL} 的真实性测试。

运算错误通过 $ENO = 0$ 或 $QERR = 1$ 表示。REAL 数据类型的范围限制超出上述公式结果时，将出现运算错误。在此情况下，保留上一周期的值 V 。如果 V 由于对应值 U 而取值 $\#+INF$ 或 $\#-INF$ ，那么在下一周期中也会出现运算错误。

3.15.2 LIMITS_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
HYS	滞后量	REAL	0.0	I
QERR	1 = 错误	BOOL	1	O
QVHL	上限已触发	BOOL	0	O
QVLL	下限已触发	BOOL	0	O
U	输入值	REAL	0.0	I
V	输出值	REAL	0.0	O
V_HL	上限	REAL	100.0	I
V_LL	下限	REAL	0.0	I

3.16 MEAS_MON: 测量值监视

3.16.1 MEAS_MON 的描述

对象名 (类型 + 编号)

FB 65

- MEAS_MON 块 I/O (页 223)
- MEAS_MON 块图标 (页 694)
- MEAS_MON 面板 (页 627)

功能

块 MEAS_MON 用于监视与以下限制对有关的测量值 (模拟信号):

- 警告限值 (上限/下限)
- 报警限值 (上限/下限)

工作原理

该块在输入 U 处监视测量值。如果超出上限或下限, 将会在相应的输出处加以指示, 并在适用时发出信号。更多信息, 可参考“消息响应”部分。

调用 OB

调用 OB 位于同一 OB 中将要监视其测量值的块之后, 它还位于 OB 100 中。

错误处理

发生算术错误时, 将设置输出 ENO = 0 和 QERR = 1。

启动特征

启动后, 将在 RUNUPCYC 所设置的周期数内抑制消息。

3.16 MEAS_MON: 测量值监视

时间响应

无时间响应。块将与测量值产生方在同一运行组中运行（请参见 CFC）。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“MEAS_MON 的 VSTATUS (页 226)”。

消息响应

MEAS_MON 块使用 ALARM8_8P 块生成消息。

存在下列消息触发器：

- 用于监视测量值限值的功能
- 作为控制系统错误通过互连来接收的 CSF 信号

可通过相应的 M_SUP_xx 输入分别抑制超出限制消息。可使用 MSG_LOCK 集中禁用过程消息（不是控制系统消息）。

如果重新启动以来未满足 RUNUPCYC 个周期，或者 MSG_LOCK = TRUE 或 MSG_STAT = 21，则会设置 QMSG_SUP。

监视过程值

不可用

更多信息

更多相关信息，可参考：

MEAS_MON 的 I/O (页 223)

MEAS_MON 的消息文本和关联值 (页 225)

MEAS_MON 的 VSTATUS (页 226)

操作和监视 MEAS_MON (页 226)

3.16.2 MEAS_MON 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认 值	类型	OCM	允许 值
AUX_PRx	关联值 x	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 当前批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	0	I	+	
CSF	控制系统错误	BOOL	0	I		
HYS	滞后量	REAL	5	I	+	≥ 0
M_SUP_AH	1 = 消息抑制报警上限	BOOL	0	I	+	
M_SUP_AL	1 = 消息抑制报警下限	BOOL	0	I	+	
M_SUP_WH	1 = 消息抑制警告上限	BOOL	0	I	+	
M_SUP_WL	1 = 消息抑制警告下限	BOOL	0	I	+	
MO_PVHR	显示上限 (测量范围) - 仅适用于 OS	REAL	110	I	+	
MO_PVLR	显示下限 (测量范围) - 仅适用于 OS	REAL	-10	I	+	
MSG_ACK	消息已确认	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		
MSG_LOCK	1 = 锁定过程消息	BOOL	0	I	+	
MSG_STAT	消息错误信息	WORD	0	O		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
QC_U	U 的质量代码	BYTE	16#80	I		
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	
QH_ALM	1 = 报警上限	BOOL	0	O		

I/O (参数)	含义	数据类型	默认 值	类型	OCM	允许 值
QH_WRN	1 = 警告上限	BOOL	0	O		
QL_ALM	1 = 报警下限	BOOL	0	O		
QL_WRN	1 = 警告下限	BOOL	0	O		
QMSG_ERR	1 = 消息错误	BOOL	0	O	+	
QMSG_SUP	1 = 消息抑制激活	BOOL	0	O	+	
RUNUPCYC	起动周期数	INT	3	I		
STEP_NO	BATCH 步号	DWORD	0	I	+	
U	模拟输入 (测量值)	REAL	0	I	+	
U_AH	报警上限	REAL	100	I	+	U_AH>U_WH
U_AL	报警下限	REAL	0	I	+	U_AL<U_WL
U_WH	警告上限	REAL	95	I	+	U_AH>U_WH >U_WL
U_WL	警告下限	REAL	5	I	+	U_WH>U_WL >U_AL
USTATUS	VSTATUS 中的状态字, 可由用户组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.16.3 MEAS_MON 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由 此项抑制
1	QH_ALM	\$\$BlockComment\$\$ 上上限报警	AH	M_SUP_AH、 MSG_LOCK
2	QH_WRN	\$\$BlockComment\$\$ 上限报警	WH	M_SUP_WH、 MSG_LOCK
3	QL_WRN	\$\$BlockComment\$\$ 下限报警	WL	M_SUP_WL、 MSG_LOCK
4	QL_ALM	\$\$BlockComment\$\$ 下下限报警	AL	M_SUP_AL、 MSG_LOCK
5	CSF	\$\$BlockComment\$\$ 外部故障	S	-

将关联值分配给块参数

消息块的前三个关联值已分配给 SIMATIC BATCH 数据，保留第四个关联值供 U 使用，其余关联值 (AUX_PRx) 可由用户随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	U
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.16.4 MEAS_MON 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	-	-	-			MSG_LOCK	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SU P	-	-	-	-	-	-

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.16.5 MEAS_MON 的操作和监视

更多信息

更多相关信息，请参考以下部分：

- MEAS_MON 块图标 (页 694)
- MEAS_MON 面板 (页 627)

3.17 MOT_REV: 双向电机

3.17.1 MOT_REV 的描述

对象名 (类型 + 编号)

FB67

- MOT_REV 块 I/O (页 232)
- MOT_REV 块图标 (页 696)
- MOT_REV 面板 (页 630)

功能

该块用于控制可逆电机（顺时针/逆时针方向）。您可以选择监视最多两个由辅助接触器生成的反馈信号。

注意
对于 PCS 7，不应在设备块和输出驱动程序之间插入其它块。 如果违背此原则，应确保在与该块互连时，从设备块的输出到输出驱动程序，所有组成输出信号的块都安装在相同的 OB 中。

工作原理

可以使用各种输入来控制电机。依彼此之间以及与电机状态的具体层级关系来使用这些输入。特别是，互锁、反馈或旋转方向监视以及电机保护断路器会影响控制信号 QSTART (1: 打开, 0: 关闭) 和 QDIR (1: 逆时针, 0: 顺时针)。

下表显示了各个输入变量和事件按其对控制信号的影响而进行的优先级分配。后面的部分会提供更详细的信息。

优先级:	事件:
高	电机保护故障 (如果 MSS_OFF = 1)
↑	旋转方向改变时的等待时间
	LOCK = 1
	LOCK_ON = 1 (设置 LOCK_DIR)
↓	监视错误 (如果 FAULT_OFF = 1)
低	自动/手动模式
无作用	电机保护故障 (如果 MSS_OFF = 0)
	监视错误 (如果 FAULT_OFF = 0)
	控制系统错误、操作员错误

手动/自动

操作模式通过 OS 上的操作员设置 AUT_ON_OP 或通过输入 AUT_L 上的互连来切换，前提是要启用需要的功能。设定的模式会在输出 QMAN_AUT 上指示 (1 = 自动, 0 = 手动)。

- **手动模式:** 该模式允许操作员在 OS 上或通过可链接输入进行控制。
 - **OS 操作员控制 (LINK_MAN = 0):** 在 OS 上设置 FORW_ON (顺时针) 和 REV_ON (逆时针) 输入以控制电机的旋转方向, 或设置 MOT_OFF 以关闭电机。相应的使能参数 (FW_OP_EN、RV_OP_EN 或 OFFOP_EN) 必须可用。
 - **通过可互连的输入操作 (LINK_MAN = 1):** 这些命令通过 L_FORW、L_REV 和 L_OFF 输入来接收。例如, 可将这些输入互连以便能进行跟踪或本地控制。请注意, 必须通过适合的逻辑设置开关 LINK_MAN、LIOP_SEL 和 AUT_L。
- **自动模式:** 自动控制系统通过到 AUTO_ON (1 = 打开, 0 = 关闭) 或 AUTO_DIR (1 = 逆时针, 0 = 顺时针) 输入的互连来提供自动命令。

互锁

互锁功能优先于所有其它控制信号和错误,但设置相应使能信号 ($MSS_OFF = 1$) 的电机保护开关及反转旋转方向期间的时间监视功能除外。设置了 $LOCK$ 时,电机会直接关闭。如果还未设置 $LOCK$,则当设置了 $LOCK_ON$ 时,电机会直接接通。 $LOCK_DIR$ 用于在 $LOCK_ON = 1$ 时设置所需的旋转方向。

监视

监视逻辑会监视控制命令 $QSTART$ 或 $QDIR$ 与过程值反馈 FB_ON 或 FB_DIR 之间的一致性,并通过 $QRUN$ 和 $QSTOP$ 输出实际状态。如果在 $TIME_ON$ 设置的时间内未返回与 $QSTART$ 或 $QDIR$ 对应的反馈信号,或者反馈信号在未经 $QSTART$ 或 $QDIR$ 请求的情况下意外地发生变化,则它会设置监视错误 ($QMON_ERR = 1$)。

如果无反馈,可将 $QSTART$ 与 FB_ON 互连,将 $QDIR$ 与 FB_DIR 互连,或通过 $MONITOR = 0$ 来禁用监视。

参数 $FAULT_OFF$ 定义了监视错误的重要性。如果 $FAULT_OFF = 1$ 并检测到故障,电机将停止。当 $FAULT_OFF = 0$ 时,此故障状态不会影响控制输出。

电机保护

电机保护信号 MSS 的负沿将电机保护故障信号设置为不变并将此信号传送给输出 $QMSS_ST$ 。参数 MSS_OFF 用于定义是仅指出故障状态 ($MSS_OFF = 0$),还是不管其它所有输入和系统状态为何都关闭电机 ($MSS_OFF = 1$)。

无波动切换

为了确保在所有运行阶段都能平稳切换到手动模式,手动值 $FORW_ON$ 、 REV_ON 和 MOT_OFF 始终会跟踪 $QSTART$ 和 $QDIR$ 的当前值(例外情况: 反转旋转方向)。

反转旋转方向

如果选择了反转旋转方向,将发生以下情况:

- 电机停止 ($QSTART = 0$)。
- 如果 OFF 监视器不报告错误,则内部 OFF 监视器在等到经过 $TIME_OFF$ 时间段后会以反方向启动电机。请注意, $TIME_OFF$ 参数用于定义在电机实际进入静止状态前所预期需要的时间,以避免反转旋转方向时造成任何损坏。电机静止反馈不会缩短此时间,因为该消息由辅助接触器生成,并不提供关于电机实际状态的任何信息。

调用 OB

安装了该块的循环中断 OB（例如 OB32）和 OB100。

错误处理

电机保护错误 (QMSS_ST = 1) 和监视错误 (QMON_ERR = 1) 将被报告给 OS，并且按上述模式影响块的工作。操作员可以通过 RESET 将这些状态复位，也可以通过将 L_RESET 与信号“1”互连，在 MSS 的下一个正沿自动复位。控制系统故障 CSF 只会报告给 OS 并与电机保护错误和监视错误一起应用到组错误 QGR_ERR。它不会对块算法产生任何进一步的影响。

操作员错误在输出 QOP_ERR 上指示，但不会出现消息。

错误状态后的启动

错误状态后的启动取决于复位时的模式设置：

- 在自动模式下，如果自动功能提供了相应的启动信号，电机就能在复位后重新启动。
- 在手动模式下，由于手动操作被跟踪到“OFF”，所以必须明确地启动电机。

启动特征

CPU 启动时，MOT_REV 块会切换到手动模式，并输出 OFF 命令。先决条件是必须在启动 OB 中调用该块。在 CFC 工程组态中，这由 CFC 处理。使用 STEP 7 基本工具时，在启动 OB 中输入此调用。启动后，消息会在 RUNUPCYC 所设置的周期数内受到抑制。

START_OFF 输入用于定义在 CPU 启动 (START_OFF = 1) 期间是否关闭电机或者是否保持前一运行状态。

时间响应

块必须在循环中断 OB 中调用。块的采样时间在 SAMPLE_T 参数中设置。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“MOT_REV 的 VSTATUS (页 236)”。

消息响应

MOT_REV 块使用 ALARM_8P 块生成消息。

存在下列消息触发器:

- 控制系统错误
- 电机断路器故障和监视错误 (运行错误)
- 通过互连接收的 CSF 信号

如果重新启动后未满 RUNUPCYC 个周期, 或是 MSG_STAT = 21, 则会设置 QMSG_SUP。

监视过程值

不可用

更多信息

更多相关信息, 可参考:

MOT_REV 的 I/O (页 232)

MOT_REV 的消息文本和关联值 (页 235)

MOT_REV 的 VSTATUS (页 236)

操作和监视 MOT_REV (页 236)

3.17.2 MOT_REV 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
AUT_L	用于 MANUAL (手动) /AUTO (自动) 模式的互连输入: 0 = 手动; 1 = 自动	BOOL	0	I		
AUT_ON_OP	操作员输入: 0 = 手动; 1 = 自动	BOOL	0	IO	+	
AUTO_DIR	AUTOMATIC 模式旋转方向: 1 = 逆时针, 0 = 顺时针	BOOL	0	I		
AUTO_ON	AUTOMATIC 值: 1 = 打开, 0 = 关闭	BOOL	0	I		
AUTOP_EN	1 = 自动模式的操作员控制使能信号	BOOL	1	I		
AUX_PRx	关联值 x	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 连续批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	0	I	+	
CSF	控制系统错误	BOOL	0	I		
FAULT_OFF	1 = 发生故障时关闭电机	BOOL	1	I		
FB_DIR	旋转方向反馈: 1 = 逆时针, 0 = 顺时针	BOOL	0	I		
FB_ON	反馈: 1 = 打开	BOOL	0	I		
FORW_ON	1 = 设置顺时针方向	BOOL	0	IO	+	
FW_OP_EN	1 = 顺时针方向的操作员控制使能信号	BOOL	1	I		
L_FORW	AUTOMATIC 模式: 1 = 设置顺时针方向	BOOL	0	I		
L_OFF	AUTOMATIC 模式: 1 = 电机关闭	BOOL	0	I		
L_RESET	可互连的 RESET 输入	BOOL	0	I		
L_REV	AUTOMATIC 值 1 = 逆时针旋转开启	BOOL	0	I		
LINK_MAN	0 = 操作员输入激活, 1 = 通过 L_FORW、L_REV、L_OFF 进行手动控制	BOOL	0	I		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许 值
LIOP_SEL	用于手动/自动切换 (AUT_L) 的可互连输入: 1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
LOCK	1 = 锁定 (关闭)	BOOL	0	I	+	
LOCK_DIR	LOCK_ON = 1 时的旋转方向: 1 = 正向, 0 = 反向	BOOL	0	I	+	
LOCK_ON	1 = 锁定 (打开)	BOOL	0	I	+	
MANOP_EN	1 = 手动模式的操作员控制使能信号	BOOL	1	I		
MONITOR	监视: 1 = 打开	BOOL	1	I	+	
MOT_OFF	操作员输入: 1 = 电机关闭	BOOL	0	IO	+	
MSG_ACK	消息已确认	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		
MSG_STAT	消息错误信息	WORD	0	O		
MSS	电机保护断路器 (低态有效, 即 0 = 错误)	BOOL	1	I		
MSS_OFF	1 = 发生 MSS 故障时: 电机停止	BOOL	1	I		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OFFOP_EN	1 = 电机关闭的操作员控制使能信号	BOOL	1	I		
OOS	预留	BOOL	0	I	+	
QAUTOP	1 = 自动模式的操作员控制使能信号	BOOL	0	O	+	
QC_FB_DIR	FB_DIR 的质量代码	BYTE	16#80	I		
QC_FB_ON	FB_ON 的质量代码	BYTE	16#80	I		
QC_QDIR	QDIR 的质量代码	BYTE	16#80	O		
QC_QDIR_I	输出 QDIR 的质量代码	BYTE	16#80	I		
QC_QSTART	QSTART 的质量代码	BYTE	16#80	O		
QC_QSTART_I	输出 QSTART 的质量代码	BYTE	16#80	I		
QDIR	方向控制输出: 1 = 逆时针	BOOL	0	O	+	
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	
QFORW_OP	1 = 用于激活正向旋转的操作员控制使能信号	BOOL	0	O	+	

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许 值
QGR_ERR	1 = 组错误	BOOL	0	O		
QMAN_AUT	0 = 手动, 1 = 自动	BOOL	0	O	+	
QMANOP	1 = 手动模式的操作员控制使能信号	BOOL	0	O	+	
QMON_ERR	1 = 监视错误	BOOL	0	O	+	
QMSG_ERR	1 = ALARM_8P 错误	BOOL	0	O	+	
QMSG_SUP	1 = 消息抑制	BOOL	0	O	+	
QMSS_ST	存储的电机保护断路器, 1 = 错误	BOOL	0	O	+	
QOFF_OP	1 = 电机关闭的操作员控制使能信号	BOOL	0	O	+	
QOP_ERR	1 = 组操作员输入错误	BOOL	0	O		
QREV_OP	1 = 用于激活反向旋转的操作员控制使能信号	BOOL	0	O	+	
QRUN	1 = 电机正在运转	BOOL	0	O	+	
QSTART	控制输出: 1 = 打开	BOOL	0	O	+	
QSTOP	1 = 电机停止	BOOL	0	O	+	
RESET	已启用错误复位输入	BOOL	0	IO	+	
REV_ON	操作员输入: 1 = 激活反向	BOOL	0	IO	+	
RUNUPCYC	起动周期数	INT	3	I		
RV_OP_EN	1 = 逆时针方向的操作员控制使能信号	BOOL	1	I		
SAMPLE_T	采样时间 (秒)	REAL	1.0	I		> 0
START_OFF	1 = 启动期间: 电机关闭	BOOL	1	I		
STEP_NO	BATCH 步号	DWORD	0	I	+	
TIME_OFF	关闭的监视时间 (秒)	REAL	3.0	I	+	≥ 0
TIME_ON	打开的监视时间 (秒)	REAL	3.0	I	+	≥ 0
USTATUS	VSTATUS 中的状态字, 可由用户组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.17.3 MOT_REV 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由此项抑制
1	QMSS_ST	\$\$BlockComment\$\$ 电机保护	S	-
2	QMON_ERR	\$\$BlockComment\$\$ 监视故障	S	-
3	CSF	\$\$BlockComment\$\$ 外部故障	S	-

将关联值分配给块参数

消息块的前三个关联值已分配给 SIMATIC BATCH 数据，其余的关联值 (AUX_PRx) 可由用户随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	AUX_PR04
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.17.4 MOT_REV 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	QMON_ERR	-	QMSS_ST		QMAN_AUT	-	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	-	-	QDIR	QSTOP	QRUN	LOCK

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.17.5 MOT_REV 的操作和监视

更多信息

更多相关信息，请参考以下部分：

- MOT_REV 块图标 (页 696)
- MOT_REV 面板 (页 630)

3.18 MOT_SPED: 双速电机

3.18.1 MOT_SPED 的描述

对象名 (类型 + 编号)

FB68

- MOT_SPED 块 I/O (页 242)
- MOT_SPED 块图标 (页 697)
- MOT_SPED 面板 (页 633)

功能

块 MOT_SPED 用于控制双速电机 (慢速/快速)。可对接触式继电器所输出的两个反馈信号进行监视。

注意
对于 PCS 7, 不应在设备块和输出驱动程序之间插入其它块。 如果违背此原则, 则在互连该块时, 应确保在设备块的输出与输出驱动程序之间, 所有组成输出信号的块都安装在同一 OB 中。

工作原理

可以使用各种输入来控制电机。依彼此之间以及与电机状态的具体层级关系来使用这些输入。具体来说，互锁、反馈监视及电机保护断路器功能会影响控制信号 QSTART（1：打开，0：关闭）和 QSPEED（1：快速，0：慢速）。

下表显示了各个输入变量和事件按其对控制信号的影响而进行的优先级分配。后面的部分会提供更详细的信息。

优先级:	事件:
高	电机保护故障（如果 MSS_OFF = 1）
↑	LOCK = 1
	LOCK_ON = 1（配合 LOCK_SPD）
↓	监视错误（如果 FAULT_OFF = 1）
低	自动/手动模式
无作用	电机保护故障（如果 MSS_OFF = 0）
	监视错误（如果 FAULT_OFF = 0）
	控制系统错误、操作员错误

说明

必须通过外部定时器（接通延迟）设置从快速切换到慢速期间的减速次数。

手动/自动

手动/自动模式通过 OS 上的操作员设置 AUT_ON_OP 或通过输入 AUT_L 上的互连来切换,前提是要设置了必要的使能信号。设定的运行模式会在输出 QMAN_AUT 上指示 (1: 自动, 0: 手动)。

- **手动模式:** 此运行模式允许在 OS 处或通过可互连的输入进行控制。
 - **OS 操作员控制 (LINK_MAN = 0):** 以下输入可在 OS 处进行操作: 用于低速的 SP1_ON, 用于高速的 SP2_ON 或用于关闭电机的 MOT_OFF。必须相应地设置适当的参数 (S1_OP_EN、S2_OP_EN 或 OFFOP_EN)。
 - **通过可互连的输入操作 (LINK_MAN = 1):** 在此情况下, 命令在 L_SP1、L_SP2 和 L_OFF 上输入。例如, 可将这些输入互连以便能进行跟踪或本地控制。请注意, 必须通过适合的逻辑设置开关 LINK_MAN、LIOP_SEL 和 AUT_L。
- **自动模式:** 自动功能块互连通过互连自动化系统将自动模式指令输出到 AUTO_ON (1: 启动, 0: 关闭) 或 AUTO_SPD (1: 高速, 0: 低速) 输入。

互锁

互锁功能的优先级高于其它所有控制信号和错误 - 设置了相应使能信号 (MSS_OFF = 1) 时的电机保护除外。设置了 LOCK 时, 电机会直接关闭。如果还未设置 LOCK, 则当设置了 LOCK_ON 时, 电机会直接接通。LOCK_SPD 用于指定 LOCK_ON = 1 时所需的速度 (1: 快速, 0: 慢速)。

监视

监视逻辑会监视控制命令 QSTART 或 QSPEED 与实际值反馈信号 FB_ON 或 FB_SPEED 之间的一致性, 并通过 QRUN 和 QSTOP 输出实际的状态。如果在 TIME_MON 设置的时间内未设置与 QSTART 或 QSPEED 对应的反馈信号, 或者反馈信号在未经 QSTART 或 QSPEED 请求的情况下意外地发生变化, 则它会设置监视错误 (QMON_ERR = 1)。

如果无反馈, 可将 QSTART 与 FB_ON、QSPEED 与 FB_SPEED 互连, 也可通过设置 MONITOR = 0 来禁用监视功能。

FAULT_OFF 参数决定着监视错误的重要性。如果 FAULT_OFF = 1, 则在发生故障时电机会关闭。当 FAULT_OFF = 0 时, 此错误状态不会影响控制输出。

电机保护

电机保护信号 **MSS** 的负沿将电机保护故障信号设置为不变并将此信号传送给输出 **QMSS_ST**。参数 **MSS_OFF** 用于定义是仅指示故障状态 (**MSS_OFF = 0**)，还是不管其它所有输入和系统状态为何都关闭电机 (**MSS_OFF = 1**)。

无波动切换

为了确保在所有运行阶段都能无扰动地切换到手动模式，手动值 **SP1_ON**、**SP2_ON** 和 **MOT_OFF** 始终会跟踪 **QSTART** 和 **QSPEED** 的当前值。

错误处理

电机保护错误 (**QMSS_ST = 1**) 和监视错误 (**QMON_ERR = 1**) 将被报告给 **OS**，并且按上述模式影响块的工作。操作员可以通过 **RESET** 将这些状态复位，也可以通过将 **L_RESET** 与信号“1”互连，在 **MSS** 的下一个正沿自动复位。控制系统故障 **CSF** 只会报告给 **OS** 并与电机保护错误和监视错误一起应用到组错误 **QGR_ERR**。它不会对块算法产生任何进一步的影响。

操作员错误在输出 **QOP_ERR** 上指示，但不会出现消息。

调用 OB

安装了该块的循环中断 **OB**（例如 **OB32**）和 **OB100**。

错误状态后的启动

错误状态后的启动取决于复位时的模式设置：

- 在自动模式下，只要自动模式提供了相应的启动信号，电机就能在复位后重新启动。
- 在手动模式下，由于手动操作被跟踪到“OFF”，所以必须明确地启动电机。

启动特性

CPU 启动时，**MOT_SPED** 块会切换到手动模式，并输出 **OFF** 命令。这意味着必须在启动 **OB** 中调用该块。在 **CFC** 工程组态中，这由 **CFC** 处理。使用 **STEP 7** 基本工具时，在启动 **OB** 中输入此调用。启动后，消息会在 **RUNUPCYC** 所设置的周期数内受到抑制。

START_OFF 输入用于定义在 **CPU** 启动 (**START_OFF = 1**) 期间是否关闭电机或者是否保留前一个操作状态。

时间响应

该块必须在循环中断 OB 中调用。块的采样时间在 SAMPLE_T 参数中设置。

32 位状态字 VSTATUS 的分配

更多相关信息, 可参考“MOT_SPED 的 VSTATUS (页 247)”。

消息响应

MOT_SPED 块使用 ALARM_8P 块来生成消息。

存在下列消息触发器:

- 控制系统错误
- 电机断路器故障和监视错误 (运行错误)
- 通过互连接收的 CSF 信号

如果重新启动后未满足 RUNUPCYC 个周期, 或是 MSG_STAT = 21, 则会设置 QMSG_SUP。

监视过程值

不可用

更多信息

更多相关信息, 可参考:

MOT_SPED 的 I/O (页 242)

MOT_SPED 的消息文本和关联值 (页 246)

MOT_SPED 的 VSTATUS (页 247)

操作和监视 MOT_SPED (页 247)

3.18.2 MOT_SPED 的 I/O

CFC 中块显示的工厂设置在“I/O”列中标识:

I/O 名称**加粗**表示 I/O 可见; 正常字体的 I/O 名称则表示 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	属性	OCM	允许值
AUT_L	用于 MAN/AUTO 的可互连输入 0 = 手动, 1 = 自动	BOOL	0	I	Q		
AUT_ON_OP	操作员输入: 0 = 手动, 1 = 自动	BOOL	0	IO	F	+	
AUTO_ON	AUTOMATIC 值: 1 = 打开, 0 = 关闭	BOOL	0	I	Q		
AUTO_SPD	自动速度值: 1 = 快速	BOOL	0	I	Q		
AUTOP_EN	1 = 自动模式的操作员控制使能信号	BOOL	1	I	Q		
AUX_PRx	关联值 x	ANY	0	IO	Q		
BA_EN	由 BATCH 发布	BOOL	0	I	Q	+	
BA_ID	BATCH: 当前批次号	DWORD	0	I	Q	+	
BA_NA	BATCH 名称	STRING[32]	0	I	Q	+	
CSF	控制系统错误	BOOL	0	I	Q		
FAULT_OFF	1 = 发生故障时关闭电机	BOOL	1	I	Q		
FB_ON	反馈: 1 = 打开	BOOL	0	I	Q		
FB_SPEED	速度反馈: 1 = 快速	BOOL	0	I	Q		
L_OFF	AUTOMATIC 模式: 1 = 电机关闭	BOOL	0	I	Q		
L_RESET	可互连的 RESET 输入	BOOL	0	I	Q		
L_SP1	AUTOMATIC 模式: 1 = 设置顺时针方向	BOOL	0	I	Q		
L_SP2	AUTOMATIC 值 1 = 逆时针旋转开启	BOOL	0	I	Q		
LINK_MAN	0 = 操作员输入已启用, 1 = 通过 L_SP1、 L_SP2 和 L_MOTOFF 进行手动控制	BOOL	0	I	Q		

I/O (参数)	含义	数据类型	默认值	类型	属性	OCM	允许值
LIOP_SEL	用于手动/自动切换的可互连输入 (AUT_L) 1 = 互连处于激活状态 0 = 操作员控制处于激活状态	BOOL	0	I	Q		
LOCK	1 = 锁定 (关闭)	BOOL	0	I	Q	+	
LOCK_ON	1 = 锁定 (打开)	BOOL	0	I	Q	+	
LOCK_SPD	LOCK_ON = 1 时的速度 1 = 快速, 0 = 慢速	BOOL	0	I	Q	+	
MANOP_EN	1 = 手动模式的操作员控制使能信号	BOOL	1	I	Q		
MONITOR	监视: 1 = 打开	BOOL	1	I		+	
MOT_OFF	操作员输入: 1 = 电机关闭	BOOL	0	IO	F	+	
MSG_ACK	消息已确认	WORD	0	O			
MSG_EVID	消息号	DWORD	0	I	M		
MSG_STAT	消息错误信息	WORD	0	O			
MSS	电机保护 断路器 (低态有效, 即 0 = 错误)	BOOL	1	I	Q		
MSS_OFF	1 = 电机保护故障时停止电机	BOOL	1	I	Q		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	Q	+	
OFFOP_EN	1 = 电机关闭的操作员控制使能信号	BOOL	1	I	Q		
OOS	预留	BOOL	0	I		+	
QAUTOP	1 = 自动模式的操作员控制使能信号	BOOL	0	O		+	
QC_FB_ON	FB_ON 的质量代码	BYTE	16#80	I			
QC_FB_SPE ED	FB_SPEED 的质量代码	BYTE	16#80	I			
QC_QSPEED	QSPEED 的质量代码	BYTE	16#80	O			
QC_QSPEED _I	输出 QSPEED 的质量代码	BYTE	16#80	I			
QC_QSTART	QSTART 的质量代码	BYTE	16#80	O			
QC_QSTART _I	输出 QSTART 的质量代码	BYTE	16#80	I			

3.18 MOT_SPED: 双速电机

I/O (参数)	含义	数据类型	默认值	类型	属性	OCM	允许值
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O		+	
QGR_ERR	1 = 组错误	BOOL	0	O			
QMAN_AUT	0 = 手动, 1 = 自动	BOOL	0	O		+	
QMANOP	1 = 手动模式的操作员控制使能信号	BOOL	0	O		+	
QMON_ERR	1 = 监视错误	BOOL	0	O		+	
QMSG_ERR	1 = 消息错误	BOOL	0	O		+	
QMSG_SUP	1 = 消息抑制激活	BOOL	0	O		+	
QMSS_ST	存储的电机保护 断路器 (1 = 错误)	BOOL	0	O		+	
QOFF_OP	1 = 电机 OFF 的操作员 控制使能信号	BOOL	0	O		+	
QOP_ERR	1 = 组操作员输入错误	BOOL	0	O			
QRUN	1 = 电机正在运转	BOOL	0	O		+	
QS1_OP	1 = 启动速度 1 的操作员控制使能信号	BOOL	0	O		+	
QS2_OP	1 = 启动速度 2 的操作员控制使能信号	BOOL	0	O		+	
QSPEED	速度控制输出: 1 = 快速	BOOL	0	O		+	
QSTART	控制输出: 1 = 打开	BOOL	0	O			
QSTOP	1 = 电机停止	BOOL	0	O		+	
QSTOPING	预留消息	BOOL	0	O		+	
QSTRTING	预留消息	BOOL	0	O		+	
RESET	已启用错误复位输入	BOOL	0	IO	F	+	
RUNUPCYC	起动周期数	INT	3	I			
S1_OP_EN	1 = 启动速度 1 的操作员控制使能信号	BOOL	1	I	Q		
S2_OP_EN	1 = 启动速度 2 的操作员控制使能信号	BOOL	1	I	Q		

I/O (参数)	含义	数据类型	默认值	类型	属性	OCM	允许值
SAMPLE_T	采样时间 (秒)	REAL	1.0	I			> 0
SP1_ON	操作员输入: 1 = 启动速度 1 开启	BOOL	0	IO	F	+	
SP2_ON	操作员输入: 1 = 启动速度 2 开启	BOOL	0	IO	F	+	
START_OFF	1 = 启动期间: 电机关闭	BOOL	1	I	Q		
STEP_NO	BATCH 步号	DWORD	0	I	Q	+	
TIME_MON	监视时间 (秒)	REAL	3.0	I		+	≥ 0
USTATUS	VSTATUS 中的状态字, 可进行用户特定的组态	WORD	0	I			
VSTATUS	块图标中的扩展状态显示	DWORD	0	O		+	

3.18.3 MOT_SPED 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由 此项抑制
1	QMSS_ST	\$\$BlockComment\$\$ 电机保护	S	-
2	QMON_ERR	\$\$BlockComment\$\$ 监视故障	S	-
3	CSF	\$\$BlockComment\$\$ 外部故障	S	-

将关联值分配给块参数

消息块的前三个关联值已分配给 SIMATIC BATCH 数据，其余的关联值 (AUX_PRx) 可由用户随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	AUX_PR04
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.18.4 MOT_SPED 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	QMON_ERR	-	QMSS_ST	-	QMAN_AUT	-	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	QSTOPING	QSTRTING	QSPEED	QSTOP	QRUN	LOCK

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.18.5 MOT_SPED 的操作和监视

更多信息

更多相关信息，请参考以下部分：

- MOT_SPED 块图标 (页 697)
- MOT_SPED 面板 (页 633)

3.19 MOTOR: 具有一个控制信号的电机

3.19.1 MOTOR 的描述

对象名 (类型 + 编号)

FB66

- MOTOR 块 I/O (页 253)
- MOTOR 块图标 (页 698)
- MOTOR 面板 (页 636)

功能

块 MOTOR 用于通过控制信号 (打开/关闭) 来控制电机。可以选择对电机速度反馈 (打开/关闭) 进行监视。此电机速度反馈信号由继电器式接触器提供。

注意
对于 PCS 7, 不应在设备块和输出驱动程序之间插入其它块。 如果违背此原则, 应确保在与该块互连时, 从设备块的输出到输出驱动程序, 所有组成输出信号的块都安装在相同的 OB 中。

工作原理

可以使用各种输入来控制电机。依彼此之间以及与电机状态的具体层级关系来使用这些输入。具体来说，锁定、反馈监视和电机断路器会影响控制信号 QSTART。

下表显示了各个输入变量和事件按其对控制信号的影响而进行的优先级分配。后面的部分会提供更详细的信息。

优先级:	事件:
高	电机保护故障 (如果 MSS_OFF = 1)
↑	LOCK = 1
	LOCK_ON = 1
↓	监视错误 (如果 FAULT_OFF = 1)
低	自动/手动模式
无作用	电机保护故障 (如果 MSS_OFF = 0)
	监视错误 (如果 FAULT_OFF = 0)
	控制系统错误、操作员错误

手动/自动

操作员模式通过结合使用 AUT_ON_OP 的 OS 操作 (LIOP_SEL = 0) 或 AUT_L 输入的互连 (LIOP_SEL = 1) 来进行切换。如果使用 OS 系统，则必须设置相应的使能信号 AUTOP_EN 和 MANOP_EN。设定的运行模式会在输出 QMAN_AUT 上指示 (1: 自动, 0: 手动)。

- **手动模式:** 如果设置了相应的使能信号 ON_OP_EN 和 OFFOP_EN，则由 OS 系统通过输入 MAN_ON 执行操作。
- **自动模式:** 自动单元通过互连的输入 AUTO_ON 来输出控制命令。

互锁

互锁功能的优先级高于其它所有控制信号和错误 - 设置了相应使能信号 (MSS_OFF = 1) 时的电机保护除外。设置了 LOCK 时，电机会直接关闭。如果还未设置 LOCK，则当设置了 LOCK_ON 时，电机会直接接通。

监视

监视逻辑会监视控制命令 QSTART 与过程值反馈 FB_ON 之间的一致性，并通过 QRUN 和 QSTOP 输出实际的状态。如果经过 TIME_MON 时间段之后未设置与 QSTART 相应的反馈，或是如果监视逻辑在未经 QSTART 请求的情况下意外地发生变化，则它会设置监视错误 (QMON_ERR = 1)。

如果无反馈，可将 QSTART 与 FB_ON 互连，也可通过设置 MONITOR = 0 来禁用监视功能。

FAULT_OFF 参数决定着监视错误的重要性。如果 FAULT_OFF = 1，则在发生故障时电机会关闭。当 FAULT_OFF = 0 时，此错误状态不会影响控制输出。

电机保护

电机保护信号 MSS 的负沿将电机保护故障信号设置为不变并将此信号传送给输出 QMSS_ST。参数 MSS_OFF 用于定义是仅指出故障状态 (MSS_OFF = 0)，还是不管其它所有输入和系统状态为何都关闭电机 (MSS_OFF = 1)。

无波动切换

为了确保无波动切换到手动模式，手动值 MAN_ON 始终会更正为 QSTART 的当前值。

错误处理

电机保护错误 (QMSS_ST = 1) 和监视错误 (QMON_ERR = 1) 将被报告给 OS，并且按上述模式影响块的工作。操作员可以通过 RESET 将这些状态复位，也可以通过将 L_RESET 与信号“1”互连，在 MSS 的下一个正沿自动复位。控制系统故障 CSF 只会报告给 OS 并与电机保护错误和监视错误一起应用到组错误 QGR_ERR。它不会对块算法产生任何进一步的影响。

操作员错误在输出 QOP_ERR 上指示，但不会出现消息。

调用 OB

安装了该块的循环中断 OB（例如 OB32）和 OB100。

错误状态后的启动

错误状态后的启动取决于复位时的模式设置:

- 在自动模式下, 如果自动功能提供了相应的启动信号, 电机就能在复位后重新启动。
- 在手动模式下, 由于手动操作已被跟踪到“OFF”, 所以必须明确启动电机。

启动特性

CPU 启动时, MOTOR 块会切换为手动模式, 并输出 OFF 命令。先决条件是必须在启动 OB 中调用该块。在 CFC 工程组态中, 这由 CFC 处理。使用 STEP 7 基本工具时, 在启动 OB 中输入此调用。启动后, 消息会在 RUNUPCYC 所设置的周期数内受到抑制。

START_OFF 输入用于定义在 CPU 启动 (START_OFF = 1) 期间是否关闭电机或者是否保留前一个操作状态。

时间响应

该块必须在循环中断 OB 中调用。块的采样时间在 SAMPLE_T 参数中设置。

32 位状态字 VSTATUS 的分配

更多相关信息, 可参考“MOTOR 的 VSTATUS (页 257)”。

消息响应

MOTOR 块使用 ALARM_8P 块来生成消息。

存在下列消息触发器:

- 控制系统错误
- 电机保护断路器错误和监视错误 (运行错误)
- 通过互连作为控制系统错误而接收的 CSF 信号。

如果重新启动后未满 RUNUPCYC 个周期, 或是 MSG_STAT = 21, 则会设置 QMSG_SUP。

监视过程值

不可用

3.19 *MOTOR*: 具有一个控制信号的电机

更多信息

更多相关信息, 可参考:

MOTOR 的 I/O (页 253)

MOTOR 的消息文本和关联值 (页 256)

MOTOR 的 VSTATUS (页 257)

操作和监视 *MOTOR* (页 257)

3.19.2 MOTOR 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
AUT_L	用于 MANUAL (手动) /AUTO (自动) 模式的互连输入: (0 = 手动/1 = 自动)	BOOL	0	I		
AUT_ON_OP	操作员输入: 0 = 手动; 1 = 自动	BOOL	0	IO	+	
AUTO_ON	AUTOMATIC 值: 1 = 打开, 0 = 关闭	BOOL	0	I		
AUTOP_EN	1 = 自动模式的操作员控制使能信号	BOOL	1	I		
AUX_PRx	关联值 x	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 连续批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	0	I	+	
CSF	控制系统错误	BOOL	0	I		
FAULT_OFF	1 = 发生故障时关闭电机	BOOL	1	I		
FB_ON	反馈: 1 = 打开	BOOL	0	I		
L_RESET	可互连的 RESET 输入	BOOL	0	I		
LIOP_SEL	用于手动/自动切换 (AUT_L) 的互连输入: 1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
LOCK	1 = 锁定 (关闭)	BOOL	0	I	+	
LOCK_ON	1 = 锁定 (打开)	BOOL	0	I	+	
MAN_ON	操作员输入: 1 = 打开, 0 = 关闭	BOOL	0	IO	+	
MANOP_EN	1 = 手动模式的操作员控制使能信号	BOOL	1	I		
MONITOR	监视: 1 = 打开	BOOL	1	I	+	
MSG_ACK	消息已确认	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		
MSG_STAT	消息错误信息	WORD	0	O		

3.19 MOTOR: 具有一个控制信号的电机

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
MSS	电机保护断路器（低态有效，即 0 = 错误）	BOOL	1	I		
MSS_OFF	1 = 发生电机保护错误时停止电机	BOOL	1	I		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OFFOP_EN	1 = 电机关闭的操作员控制使能信号	BOOL	1	I		
ON_OP_EN	1 = 启动的操作员控制使能信号	BOOL	1	I		
OOS	预留	BOOL	0	I	+	
QAUTOP	1 = 自动模式的操作员控制使能信号	BOOL	0	O	+	
QC_FB_ON	FB_ON 的质量代码	BYTE	16#80	I		
QC_QSTART	QSTART 的质量代码	BYTE	16#80	O		
QC_QSTART_I	输出 QSTART 的质量代码	BYTE	16#80	I		
QERR	1 = 错误输出（ENO 取反）	BOOL	1	O	+	
QGR_ERR	1 = 组错误	BOOL	0	O		
QMAN_AUT	0 = 手动，1 = 自动	BOOL	0	O	+	
QMANOP	1 = 手动模式的操作员控制使能信号	BOOL	0	O	+	
QMON_ERR	1 = 监视错误	BOOL	0	O	+	
QMSG_ERR	1 = 消息错误	BOOL	0	O	+	
QMSG_SUP	1 = 消息抑制激活	BOOL	0	O	+	
QMSS_ST	存储的电机保护断路器（1 = 错误）	BOOL	0	O	+	
QOFF_OP	1 = 电机关闭的操作员控制使能信号	BOOL	0	O	+	
QON_OP	1 = 启动的操作员控制使能信号	BOOL	0	O	+	
QOP_ERR	1 = 组操作员输入错误	BOOL	0	O		
QRUN	1 = 电机正在运转	BOOL	0	O	+	
QSTART	控制输出：1 = 打开	BOOL	0	O	+	
QSTOP	1 = 电机正在停止	BOOL	0	O	+	
RESET	已启用错误复位输入	BOOL	0	IO	+	
RUNUPCYC	起动周期数	INT	3	I		
SAMPLE_T	采样时间（秒）	REAL	1.0	I		> 0
START_OFF	1 = 启动时：电机关闭	BOOL	1	I		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许 值
STEP_NO	BATCH 步号	DWORD	0	I	+	
TIME_MON	监视时间 (秒)	REAL	3.0	I	+	> 0
USTATUS	VSTATUS 中的状态字, 可由用户组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.19.3 MOTOR 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由 此项抑制
1	QMSS_ST	\$\$BlockComment\$\$ 电机保护	S	-
2	QMON_ERR	\$\$BlockComment\$\$ 监视故障	S	-
3	CSF	\$\$BlockComment\$\$ 外部故障	S	-

将关联值分配给块参数

消息块的前三个关联值已分配给 SIMATIC BATCH 数据，其余的关联值 (AUX_PRx) 可由用户随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	AUX_PR04
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.19.4 MOTOR 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	QMON_ERR	-	QMSS_ST	-	QMAN_AUT	-	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_S UP	-	-	-	QSTOP	QRUN	LOCK

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.19.5 MOTOR 的操作和监视

更多信息

更多信息，请参考以下部分：

- MOTOR 块图标 (页 698)
- MOTOR 面板 (页 636)

3.20 MPC: 模型预测控制器

3.20.1 MPC 的描述

对象名 (类型 + 编号)

FB 142

MPC 块 I/O (页 275)

MPC 的应用领域

在以下应用中，该块的使用方式与 PID 控制器的使用方式大致相同：

- 固定设定值控制
- 串级控制
- 比率控制
- 分程控制

与 PID 控制器不同，MPC 是一种多变量控制器。

应用领域

该块用于动态过程的多变量控制。它最多可处理四个相关的调节变量和控制变量及一个可测定的干扰变量。

特殊情况下，MPC 块还可以用于难度极高的动态、单变量控制。例如，在非最小相位或强烈振荡响应的系统中，它要优于 PID 控制器。

MPC 算法只适用于具有可在有限时间内稳定到固定值的阶跃响应的稳定过程。如果过程不稳定或包括积分器（例如罐液位控制），则必须通过次级控制器稳定相应的局部传递函数。

对于集成系统，一个简单的 P 控制器就足够了。

控制器应用领域的说明: 较长的执行时间

由于多变量控制器的工作原理, 它的执行时间明显比 PID 控制器长。因此, 多变量控制器不适合快速控制。

CPU 所需的计算时间被以下事实所抵偿: 典型的 MPC 应用需要非常慢的采样时间 (> 20s) (请参阅 高级过程控制模板 (页 765))。因而, MPC 通常位于 OB 30 中, 并且可被较快的 OB 中断。

工作原理

MPC 块是一种基于模型的预测多变量控制器。它将过程 (包括交互) 的数学模型用作控制器的一部分。可通过此模型预测未来某一定义时段 (也称为预测时域) 内的过程响应。

根据此预测, 可在以下条件适用的情况下

$$J = (\bar{w} - \bar{y})^T \cdot R \cdot (\bar{w} - \bar{y}) + \Delta \bar{u}^T \cdot Q \Delta \bar{u}$$

优化 (最小化) 拟合 (质量) 标准:

- w 包含未来设定值的时间序列,
- y 包含未来控制变量的向量,
- Δu 包含调节变量的未来变化。

如果增大 Q 矩阵中的权重, 控制器将更慎重地移动其调节变量, 这将会产生更慢但鲁棒性更强的控制动作。使用 R 矩阵中的加权系数, 可以指定各个控制变量的相对重要性。控制变量的权重 (优先级) 越高, 意味着会更快地向设定值移动, 而且即使不能精确达到所有设定值, 稳定状态时结果也会更准确。

该算法是 DMC- 模式 (Dynamic Matrix Control), 的一个变型, 它能在忽略约束条件的设计阶段解决优化问题。该功能块本身包含优化问题的解析解法。调节变量限制 (绝对和相对梯度) 在该功能块的算法中视为不得违反的限制。这意味着优化中应该尽量考虑控制变量的准确设定值或时域。使用参考变量滤波器进行未来设定值的设置, 可在运行期间精细调整该功能块的控制动作。

当单个干扰变量可以测定时 (例如吞吐量变化), 便可显著改善控制质量。这种情况下, 在预测控制变量时最好考虑该干扰变量对控制变量的影响模型, 以便控制器能够提前对此类干扰作出反应。

组态

使用 CFC 编辑器以将该块安装到循环中断 OB (OB 3x) 中。该块还会自动安装到启动 OB (OB100) 和 OB 1 中。在 CFC 中完成安装后, 请按下述步骤操作:

1. 通过应用一系列调节变量阶跃变化, 使用控制器以手动模式激活过程。
2. 用 CFC 趋势显示来记录测量数据并将其导出到归档文件。
3. 在 Windows 中通过“开始”(Start) > SIMATIC > STEP 7 > Engineering Tool ModPreCon 为 MPC 块启动 Engineering Tool ModPreCon。
4. 使用该工具为用户数据块 (DB) 创建 SCL 源代码。它包括 MPC 实例所需的模型和矩阵。
5. 在工程系统中编译 SCL 源代码, 然后下载到 AS。
6. 在 MPC 块的输入 DB_No 中输入数据块的编号。控制器将通过在重新启动输入时设置“复位”(Reset) 来采用这些值。

说明

有关此过程的详细描述, 可参考 ModPreCon 工程工具的帮助。

库中的模板包括如何使用 MPC 块的实例。

启动特征

CPU 启动时, 该块始终以手动模式启动。只有加载了用户数据块且 MPC 的内部测量值存储器填充了数据时, 才能切换到自动模式。

时间响应

该块必须在循环中断 OB 中调用。块的采样时间由 CFC 于编译期间在 SampleTime 参数中设置。

说明

在 ModPreCon 工具中设计控制器期间, 会计算和显示控制器循环时间和 OB 采样时间。由您自己负责在适合 OB 采样时间的循环中断级中调用 MPC 块。必要时, 较慢的控制器循环时间将在块中通过内部向下采样功能自动实现。

消息响应

MPC 块没有单独的消息响应。如有必要, 您可以使用要监视的各控制变量的下游 MEAS_MON 块来扩展此功能。

状态字分配

位号:	3	2	1	0
参数:	-	-	BatchEn	Occupied

位号:	7	6	5	4
参数:	SP_TrkCV	DV_ModelAvailable	DB_Loaded	无自动模式

位号:	11	10	9	8
参数:	MV2TrkOn	MV1TrkOn	-	-

位号:	15	14	13	12
参数:	-	-	MV4TrkOn	MV3TrkOn

位号:	19	18	17	16
参数:	-	-	-	-

位号:	23	22	21	20
参数:	-	-	-	-

位号:	27	26	25	24
参数:	UserStatus bit 4	UserStatus bit 3	UserStatus bit 2	UserStatus bit 1

位号:	31	30	29	28
参数:	UserStatus bit 8	UserStatus bit 7	UserStatus bit 6	UserStatus bit 5

参见

MPC 错误处理 (页 274)

MPC 的功能 (页 265)

MPC 模式 (页 263)

3.20.2 MPC 模式

MPC 模式

该块可在控制器块可用的所有标准运行模式下运行:

- 自动模式
- 手动模式

在手动模式下, 操作员可以控制设备, 即手动操作。操作员将决定如何更改块的调节变量。

在自动模式下, 设备由块算法自动控制, 即自动控制过程。当从自动模式切换到手动模式时, 设备在自动模式下的最后一次有效控制在切换完成的初始时刻保持有效。

手动模式和自动模式适用于带有所有控制通道的整个块。与 PID 控制器相比, 允许在自动模式下运行 MPC 块且其控制信号不会影响过程, 因为不存在积分器饱和的风险。

下一部分提供与常规描述相关的附加块特定信息。其中包括用于模式更改的参数等。

自动模式

如果满足以下条件, 控制器将在自动模式下运行

- 设置了 $\text{AutOnOp} = 1$ I/O, 或者
- 在块的标准视图中激活了自动模式。

手动模式

如果满足以下条件, 控制器将在手动模式下运行

- 设置了 $\text{AutOnOp} = 0$ I/O, 或者
- 在块的标准视图中激活了手动模式。

在模式间切换

可从手动模式平稳地切换到自动模式。可在 SP_TrkCV I/O 处设置此响应。

- **无扰动切换**意味着自动调节变量会跟踪 (SP_TrkCV = 1) 手动调节变量。这实现了从手动模式到自动模式的平稳切换，而不会引起连续控制器中调节变量值的阶跃变化。也可以通过面板实现无扰动切换。
- **近似无扰动切换**意味着切换模式时，块立即根据设定值和过程值重新计算调节变量的值，即设定值可能有阶跃变化。

模式可由操作员（输入 AutOnOp）或更高级别的控制系统（输入 AutOnLi）进行更改。您通过相应地设置输入 LiOpMod. 来决定是启用操作员输入还是启用互连。

参见

MPC I/O (页 275)

MPC 错误处理 (页 274)

MPC 的功能 (页 265)

MPC 的描述 (页 258)

3.20.3 MPC 的功能

MPC 的功能

该块提供以下功能:

- 生成和限制调节变量
- 调节值跟踪
- 内部设置设定值
- 手动模式下的设定值跟踪
- 设定值滤波器
- 控制误差生成和死区
- 预测控制器算法
- 抗饱和
- 基于模型的干扰变量补偿
- 方形或非方系统控制
- 线性和非线性系统控制
- 质量代码的显示和输出
- BATCH 功能

生成和限制调节变量

调节变量 MV1 到 MV4 可从不同来源获取:

- 在自动模式下从预测控制器算法中获取,
- 在手动模式下从手动输入 MV1Man 到 MV4Man 中获取,
- 在调节变量跟踪模式下 (可针对每个通道单独激活) 从可互连的跟踪输入 MV1Trk 到 MV4Trk 中获取。

在自动模式下, 调节变量被限制为:

- 上限: MV1HiLim 到 MV4HiLim
- 下限: MV1LoLim 到 MV4LoLim.

在手动模式下, 调节变量被限制为:

- 上限: MV1ManHiLim 到 MV4ManHiLim
- 下限: MV1ManLoLim 到 MV4ManLoLim.

自动模式下的限制通常比手动模式下的限制更严格。对于近似非线性过程响应的线性过程模型的有效限制范围, 这可保证自动模式下在设置范围内闭环控制回路的稳定性。

在自动模式下, 调节变量的梯度 (每秒变化次数) 限制为 MV1RaLim 到 MV4RaLim。梯度限制适用于正方向和负方向。

调节值跟踪

可通过 MV1TrkOn 到 MV4TrkOn 中的任一输入来单独激活各个通道的跟踪。然后通过互连的跟踪输入 MV1Trk 到 MV4Trk 设置相应的调节变量。手动模式的优先级高于跟踪模式, 这样当跟踪调节变量值时用户仍可以进行干预。

典型的使用案例是串级控制: 如果分配给某个控制通道的次级控制器不再处于通过外部设定值激发的自动模式, 则 MPC 主控制器的相应控制通道必须切换到跟踪模式。

内部设置设定值

对于此块, 必须始终内部设置 SP1 到 SP4 I/O 的设定值。通常在面板上设置这些值。特殊情况下, 可以互连设定值, 但以后无法再通过面板进行更改。

手动模式下的设定值跟踪

此种情况下 ($SP_TrackCV = 1$)，内部设定值 SP1 到 SP4 将在手动模式下跟踪分配的控制变量 CV1 到 CV4。此功能允许平稳地切换到自动模式。切换后，操纵员可再次更改设定值。

设定值滤波器

要改变预测控制器的动作，且不使用 MPC 工具创建新用户数据块和重新初始化控制器，设定值滤波器是唯一方法。设定值滤波器的指定时间常数 PreFilt1 到 PreFilt4 可解释为设定值阶跃变化后此 CV 通道所需的稳定时间。随着时间常数设置的增大，控制器的工作将会越来越慢且更不主动。尤其是，这会降低某个控制通道中设定值阶跃变化对邻近控制通道的影响。

MPC 块内部使用与预测的控制变量变化相比的未来设定值集合。如果没有设定值滤波器，将假定当前设定值在以后的预测时域范围内继续有效。如果存在设定值阶跃变化，这意味着在不久的将来会需要新设定值的全值，虽然过程无法达到这一设定值(根据预测)。如果有设定值滤波器，将计算从当前过程值到所需设定值的渐近设定值轨迹（一阶），以便在指定时间内到达所需设定值。

说明

如果过程值因干扰明显偏离设定值，则设定值滤波器也可在无设定值阶跃变化的情况下生效。这意味着滤波器不仅可以减慢控制动作还可以间接对干扰作出响应。

设定值滤波器只能减慢而不能加快控制动作；当值为 0 时，禁用预滤波器。因此，建议在 MPC 工程工具中通过“调节变量变化补偿”(Manipulated variable change penalty) 参数设置基本控制器动作，然后在软件中使用仿真闭环控制回路功能优化此动作。然后只应使用软件滤波器在运行系统对动作进行精细修改。

控制误差生成和死区

在预测控制器中，作为预测的过程变化（开始于控制变量 CV1 到 CV4 的当前值）与未来设定值设置（结束于 SP1 到 SP4）之间偏差的误差信号会在整个预测时域内为每个控制通道都生成，并用于计算调节变量。

原则上，死区 SP1DeadBand 到 SP4Deadband 的影响与 PID 控制器中的影响相同，但是它遍布整个未来预测时域。换句话说，例如，如果整个预测时域中的预测控制变量 CV1 在 $SP1 \pm SP1DeadBand$ 范围内，则控制器没有任何理由改变任一调节变量。因此，这些范围也称为 CV 边带。与调节变量限制相比，这些限制不是必须遵守的硬性约束条件。

在多变量控制器中，建议利用这一事实：从应用角度来讲，某些控制变量需要精确移动到指定设定值，而其它变量只需保留在定义的范围內。

典型实例是为其指定容差范围的质量特征。PID 控制器中的死区趋向于影响稳定性，而各个控制器通道中的 CV 边带通常有助于多变量控制器的整体稳定性。

通过 CV 边带，可实现软件超驰控制的动作。

带有死区的误差信号生成的使用案例

只要反应釜中的压力保持在设定的安全限制范围内，控制器将只关心产品质量。但是，一旦压力有离开允许范围的危险（即预测它以后将向非法值移动），则将启动压力控制。通过在拟合标准中对控制变量加权（请参见 MPC 工程软件），用户可以指定为可能超出的压力限制指定特别高的权重。

预测控制器算法

MPC 块源于著名的 DMC 算法 (Dynamic Matrix Control)。根据下面的公式计算控制时域内调节变量的未来变化

$$\Delta \bar{u} = \underline{C} \cdot (\bar{w} - \bar{f})$$

其中:

- w 包含未来设定值的时间序列
- f 包含预测的控制变量 (调节变量恒定时) 将来的自由变化
- C 是 MPC 工程工具计算的恒定控制器矩阵。 C 包括过程模型和通过优化拟合标准得出的调节变量变化和控制变量的加权。

根据滚动时域原理, 只从整个控制时域内最优调节变量变化的向量中获取第一个值, 并将其应用到过程。在下一步中, 将考虑最新到达的过程值, 并在整个预测时域内重复计算。

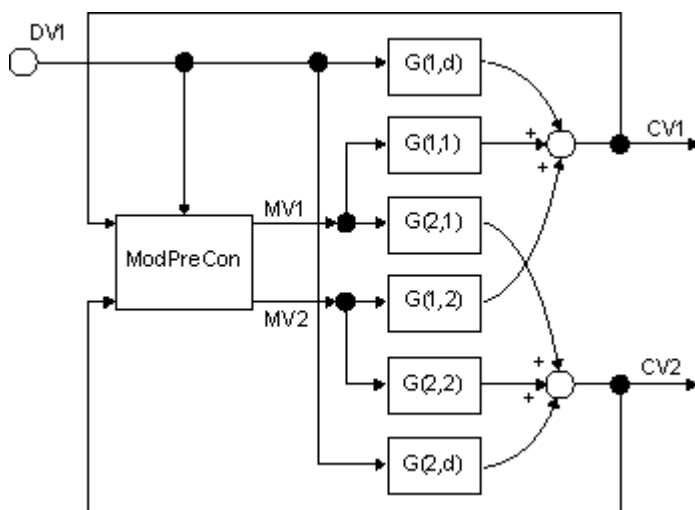
对于预测控制器, 调节变量变化基于预测的未来控制偏差, 而对于 PID 控制器, 则基于以前的控制误差 (可能也已集成)。这可以解释为“预测未来”策略。

抗饱和

激活调节变量限制时, 控制器中将自动采取抗饱和措施。预测方程将使用调节变量的实际限制值而不是计算出的理论值。

基于模型的干扰变量补偿

当已知干扰对过程有很强的影响且原因可测量时，可以并且应该使用基于模型的干扰变量补偿。



控制器处于手动模式时，可以估计可测量干扰 (DV1 I/O) 对所有控制变量 CV1 到 CV4 的影响。这意味着任何控制变量变化都不是调节变量变化造成的，所有运动都是由干扰变量引起的。如果干扰变量可测量但无法主动调整，则可能需要在整个数据归档中进行搜索以找到该干扰变量发生变化的时间段。

从干扰变量 DV1 至所有控制变量 CV1 到 CV4 的传递函数（干扰模型，上图中的 $G(1,d)$ 和 $G(2,d)$ ）的识别通过 MPC 工程工具完成，与主传递函数 ($G(1,1)$ 到 $G(2,2)$) 识别类似。然后，测量的干扰变量切换为 MPC 块的 DV1 输入，并通过 $DV_On = 1$ 激活干扰变量补偿。因此，预测中应考虑可测量干扰的影响，以便控制器能够在干扰对控制变量造成严重影响之前启动应对措施。如果用户数据块中不存在干扰模型，则忽略 DV1 输入。

可测量干扰变量的典型实例是蒸馏塔的入气量或连续反应釜的吞吐量。

方形或非方系统控制

在多变量控制器中,理想状态下,调节变量的数量应与控制变量的数量相同。这称为“方形系统”。只要约束条件不影响运行,原则上,控制器就可以控制选定的设定值。

如果调节变量比控制变量少,或各个调节变量均已达到极限,说明控制问题中缺少自由度。这意味着所有设定值不能都精确到达。

然后, MPC 算法找到一个折衷方案,该方案可受 MPC 工程工具中控制值权重(优先级)选择的影响:具有较高优先级的控制变量将具有较低的控制偏差。

说明

因为 MPC 是不带有实时优化的倾向预测控制器算法,所以一般不能保证找到的折衷方案从数学意义上而言是最优的;换言之,该方案是考虑调节变量限制的拟合函数的最小值。但大多数实际情况下,控制器可找到切合实际的折衷方案。

如果调节变量比控制变量多或某些控制变量已位于设定值范围内,控制问题中将存在过剩的自由度。但是,倾向预测控制器算法无法明确识别这种情况,而且也无法将这些自由调节变量用于优化。因此, MPC 块将把所有调节变量都移动到控制变量要达到的目标值,然后让它们保持不变。但在某些情况下,为控制器提供的调节变量比控制变量多会很有用,例如,各个调节变量的影响过于受限时。

另一种方法是同时将过剩的调节变量定义为伪控制变量。可通过为这些伪控制变量分配低优先级的设定值来实现此目的。然后控制器将重要控制目标作为最高优先级加以实现,同时还尝试到达单个调节变量的某些理想值。

线性和非线性系统控制

MPC 算法基于线性、非时变过程模型。因此,与 PID 控制器十分相像,它首先适用于对围绕固定运行点的非线性系统进行控制。

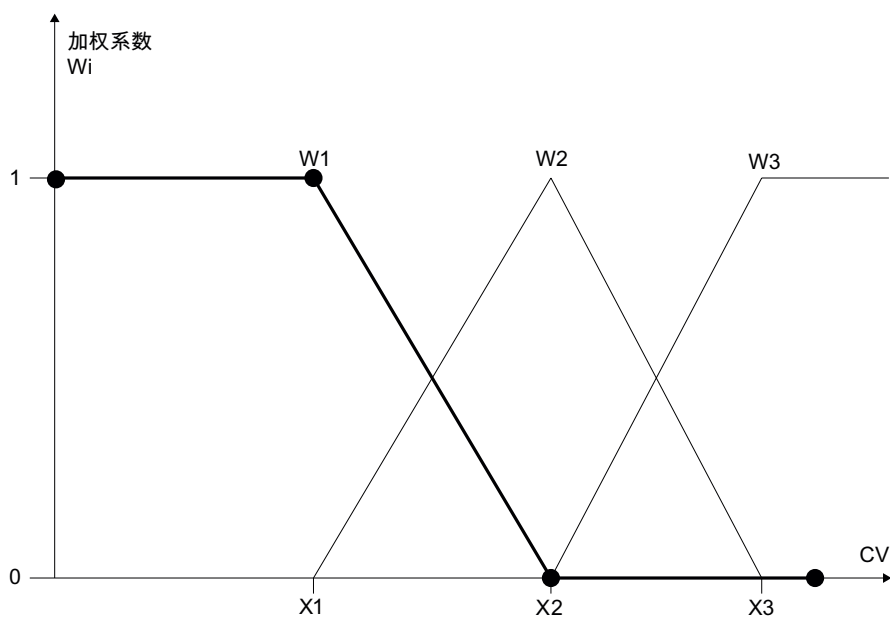
而且还有通过该算法来扩展非线性系统应用领域的多种方法,这也类似于 PID 控制器:

控制器和受控系统之间的补偿功能:

例如,可以在 MV 输出和阀块控制输入之间使用折线块补偿非线性阀特征曲线的影响。实现调节变量限制时务必小心。同样,通过在控制器 CV 输入之前使用折线块,也可以补偿受控系统(例如,传感器特征曲线)输出处的非线性影响。请记住还必须对相应的 SP 进行相应地转换。在这两个例子中,从控制器角度看,补偿功能已成为受控系统的一部分。目的是始终保持包含过程和补偿元件的控制系统的总体响应尽可能是线性的。

多模型控制:

具有不同运行点的不同模型的多个 MPC 实例同时运行。调节变量的最终值成为了各个控制器推荐的调节变量的加权平均值。加权系数 0...1 与从模糊逻辑推算出的隶属函数的构成方式相同，这样，所有加权的总和始终为 1，且每个控制器在其自身运行点处具有最高加权。为确保整个控制回路的稳定性，所有子控制器必须至少在所有运行点稳定。与 PID 控制器相比，如果 MPC 临时在自动模式下运行但无法干预实际过程（加权系数为零），则它不会受到饱和问题的影响。



轨迹控制:

该方法巧妙地组合了开环控制器 (Feedforward Control) 和带有过程值反馈 (Closed Loop Control) 的闭环控制器的优点。控制器将跟踪先前优化的设定值和调节值的轨迹；即，它只需补偿已存储轨迹和当前设备状态之间的细微偏差。轨迹是随时间变化的最优调节变量序列和与它们匹配的过程值。所需的调节变量通过输入 MV1Traj 到 MV4Traj 读入 MPC 块，并添加到通过算法计算出的调节变量值中（仅适用于自动模式）。此方法的突出优点是可以组态作用于过程的有效调节变量，并将其限制为轨迹和控制器动作的总和。轨迹的过程值可转换到控制器的相应设定值输入 SP1 到 SP4。只要过程反应与轨迹中计划的相同，它就会对轨迹的调节变量序列及相应的过程值序列作出响应，且控制偏差为零。一般来说，非线性动态过程可围绕固定运行点或系统的某个稳态线性化。也可以围绕某个轨迹线性化。

质量代码的显示和输出

该块可以处理和输出多个质量代码。

可使用以下输入：

- 控制变量的质量：QC_CV1 到 QC_CV4
- 可测量干扰变量的质量：QC_DV

以下附加输出也可用于进一步处理：

- 最差的质量代码：QC_Worst

有关质量代码的常规信息，请参考“块的功能 > 质量代码”部分

BATCH 功能

此块具有 BATCH 接口。您可以通过将 BatchEn、BatchID、BatchName、StepNo 和已占用的 I/O 与相应的 BATCH 块互连来使用 SIMATIC BATCH。请参考 SIMATIC BATCH 文档。

参见

MPC I/O (页 275)

MPC 错误处理 (页 274)

MPC 模式 (页 263)

MPC 的描述 (页 258)

3.20.4 MPC 错误处理

MPC 错误处理

ErrorNum I/O 可用于输出错误号。

错误号概述

此块输出的错误号:

错误号	错误号的含义
-1	安装此块时的默认值; 此消息不相关
0	没有错误。
2	SampleTime < 0,001 [s]
21	MV_Trk1 的值不再显示在 REAL 数域中。
22	MV_Trk2 的值不再显示在 REAL 数域中。
23	MV_Trk2 的值不再显示在 REAL 数域中。
24	MV_Trk3 的值不再显示在 REAL 数域中。
31	CV1 的值不再显示在 REAL 数域中。
32	CV2 的值不再显示在 REAL 数域中。
33	CV3 的值不再显示在 REAL 数域中。
34	CV4 的值不再显示在 REAL 数域中。
90	无法从用户数据块加载控制器矩阵。

参见

MPC I/O (页 275)

MPC 的功能 (页 265)

MPC 模式 (页 263)

MPC 的描述 (页 258)

3.20.5 MPC I/O

ModPreCon I/O

输入

I/O (参数)	描述	数据类型	默认值
AutOnLi	1 = 通过互连激活自动模式	BOOL	0
AutOpEn	1 = 操作员可激活自动模式	BOOL	1
AutOnOp	1 = 由操作员激活自动模式	BOOL	0
BatchEn	1 = 通过批生产激活控制	BOOL	0
BatchID	当前批生产 ID (号)	DWORD	16#FF
BatchName	当前批生产名称	字符串 [32]	"
CVx	(x = 1 - 4) 控制变量 1 - 4 (过程值)	REAL	0
DB_No	存储控制数据的数据库的编号	INT	
DV_On	选择干扰前馈	BOOL	
DV1	干扰变量	REAL	0
LiOpMod	模式转换: 0 = 操作员 1 = 互连	BOOL	0
ManOpEn	1 = 操作员可激活手动模式	BOOL	1
MV_ManOpEn	1 = 操作员可以更改调节变量	BOOL	1
MV1HiLim	调节变量 1 的上限	REAL	1
MV1LoLim	调节变量 1 的下限	REAL	0
MVxMan	(x = 1 - 4) 1 = 手动模式下设置调节变量 1 - 4	REAL	0
MVxManHiLim	(x = 1 - 4) 手动模式下调节变量 1 - 4 的上限	REAL	1
MVxManLoLim	(x = 1 - 4) 手动模式下调节变量 1 - 4 的下限	REAL	1
MVxRaLim	(x = 1 - 4) 调节变量 1 - 4 的变化率 (每秒)	REAL	1

3.20 MPC: 模型预测控制器

I/O (参数)	描述	数据类型	默认值
MVxTrk	(x = 1 - 4) 跟踪模式下的调节变量 1 - 4	REAL	0
MVxTrkOn	(x = 1 - 4) 1 = 激活调节变量 1 - 4 的跟踪模式	BOOL	0
Occupied	被批生产占用	BOOL	0
PreFiltx	(x = 1 - 4) 预滤波器稳定时间的设定值 1 - 4 [s]	REAL	0
QC_CVx	(x = 1 - 4) 信号 CV1 - CV4 的质量代码	BYTE	16#80
QC_DV1	信号 DV1 的质量代码	BYTE	16#80
Restart	以从用户数据块传送数据重新启动	BOOL	1
SampleTime	采样时间 [s] (自动分配)	REAL	0.1
SP_OpEn	1 = 操作员可以更改设定值	BOOL	1
SP_TrkCV	手动模式和跟踪模式下设定值跟踪 CV	BOOL	
SPx	(x = 1 - 4) 设定值 1 - 4	REAL	0
SPxDeadBand	(x = 1 - 4) CV1 - CV4 的范围控制半径	REAL	0
SPxHiLim	(x = 1 - 4) 设定值 1 - 4 的上限	REAL	1
SPxLoLim	(x = 1 - 4) 设定值 1 - 4 的下限	REAL	0
StepNo	批生产的步号	DWORD	16#00
UserStatus	状态字的空闲用户区域 (位 24 至位 31)	BYTE	16#00

输出

I/O (参数)	描述	数据类型	默认值
ActInOuts	状态字在屏幕上显示激活的输入和输出	WORD	
AutAct	1 = 自动模式处于活动状态 0 = 手动模式处于活动状态	BOOL	0
CVxOut	(x = 1 - 4) 复制控制变量 1 - 4 作为输出参数	REAL	0
DV_Model_Available	1 = 由于干扰模型存在, 干扰前馈可激活	BOOL	1
ErrorNum	组态错误 有关错误号的信息, 请参考 错误处理 (页 274)	INT	0
MVx	(x = 1 - 4) 调节变量 1 - 4 (控制器输出)	REAL	0
MVxAutAct	(x = 1 - 4) 自动设置调节变量 1 - 4	BOOL	0
NumberCVs	CV 的数量	INT	0
NumberDVs	DV 的数量	INT	0
NumberMVs	调节变量的数量	INT	0
Q_AutOpEn	1 = 操作员可激活自动模式	BOOL	1
Q_ManOpEn	1 = 操作员可激活手动模式	BOOL	1
Q_SP_OpEn	1 = 操作员可以更改设定值	BOOL	1
Q_MV_ManOpEn	1 = 操作员可以更改调节变量	BOOL	1
SPxOut	(x = 1 - 4) 操作员使用的激活设定值 1 - 4	REAL	0
Status	状态字	DWORD	0

参见

MPC 的功能 (页 265)

MPC 模式 (页 263)

MPC 的描述 (页 258)

3.21 NOISE_GN: 噪声信号发生器

3.21.1 NOISE_GN 描述

对象名 (类型 + 编号)

FB 143

- NOISE_GN 块 I/O

工作原理

此块相当于一个噪声发生器。它用于生成受噪声影响的仿真信号。
在控制实例中使用这些仿真信号。

启动特征

该块没有任何启动特征。

时间响应

该块没有任何时间响应。

状态字分配

该块没有状态字分配。

3.22 POLYG_P: 最多有 8 个点的多边形

3.22.1 POLYG_P 的描述

对象名 (类型 + 编号)

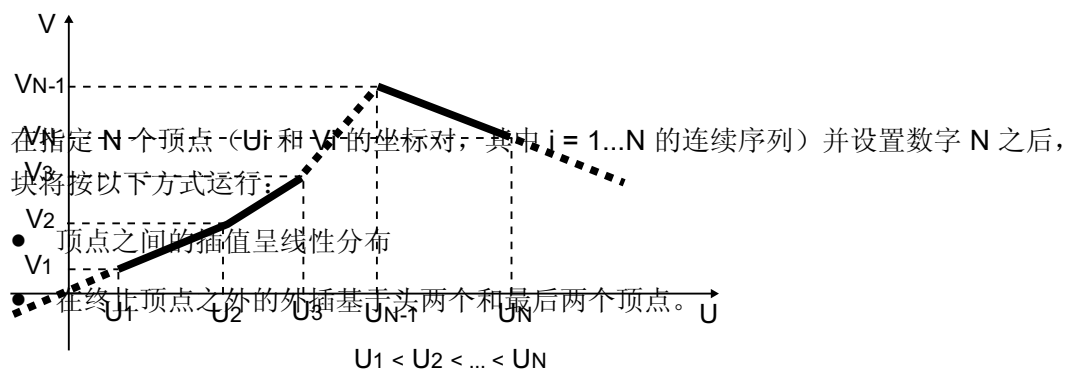
FC271

- POLYG_P 块 I/O (页 280)

功能

按照非线性特征曲线以及最多只能有 8 个顶点的原则将输入 U 转换为输出 V 。

工作原理



特征曲线表示

调用 OB

在其中安装了该块的 OB。

错误处理

如果满足以下条件, 则输出 $ENO = 0$ 和 $V = U$:

- 顶点数 $N < 2$ 或 $N > 8$
- $U_i \geq U_{i+1}$, 其中 $i = 1, 2 \dots N-1$

3.22 POLYG_P: 最多有 8 个点的多边形

3.22.2 POLYG_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许 值
N	顶点数	INT	0	I		$2 \leq N \leq 8$
U	输入	REAL	0.0	I		
U1	顶点 1 的 U 值	REAL	0.0	I		
U2	顶点 2 的 U 值	REAL	0.0	I		
...		
U8	顶点 8 的 U 值	REAL	0.0	I		
V1	顶点 1 的 V 值	REAL	0.0	I		
V2	顶点 2 的 V 值	REAL	0.0	I		
...		
V8	顶点 8 的 V 值	REAL	0.0	I		
V	输出值	REAL	0.0	O		

3.23 PT1_P: 一阶滞后元件

3.23.1 PT1_P 的描述

对象名 (类型 + 编号)

FB51

- PT1_P 块 I/O (页 283)

功能

PT1_P 块按照以下公式运行:

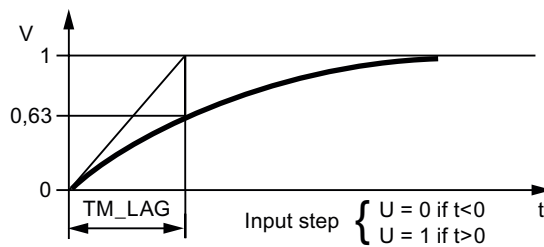
$$v(s) = 1 / (TM_LAG * s + 1) * u(s)$$

工作原理

按照时间常量 TM_LAG 将输入信号 U 传递到输出 V。

STOP_RES 输入具有以下影响:

- 如果 STOP_RES = "1", 则停止计算过程。在此期间内, 输出值保持不变。
- 通过下降沿 1 → 0 重置输出 (V = U)。



PTI_P 的步响应

调用 OB

这是在其中安装了该块的 OB (例如 OB32)。

错误处理

允许组态 $TM_LAG = 0$, 表示没有滞后, 即 $V = U$ 。相反, TM_LAG 的负值将被视为参数分配错误, 并通过 $QERR = TRUE$ 进行确认。

3.23.2 PT1_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	允许值
QERR	1 = 错误	BOOL	1	O	
SAMPLE_T	采样时间 (秒)	REAL	1.0	I	> 0
STOP_RES	停止/复位 PT1 功能	BOOL	0	I	
TM_LAG	时间常量的长度	REAL	0.0	I	> 0
U	输入值	REAL	0.0	I	
V	输出值	REAL	0.0	O	

3.24 RAMP_P: 斜坡生成

3.24.1 RAMP_P 的描述

对象名 (类型 + 编号)

FB52

- RAMP_P 块 I/O (页 286)

功能

限制模拟信号的斜坡

工作原理

块将计算输入信号 dU/dt 的斜坡并将其与两个限制 (正变化极限 URLM 或负变化极限 DRLM) 进行比较 (请参考下表)。

- 如果斜坡 (作为数量) 超出了相关最大斜坡 (URLM 或 DRLM), 输出 V 将只能更改为允许的值, 并将设置相应的限制显示 QLIM_U 或 QLIM_D。
- 如果变化率在有效范围内, 将传递输入值 ($U = V$) 并复位 QLIM_U 和 QLIM_D 这两个值。
- 如果输入 RATE_OFF=1, 将禁用斜坡生成, 以使 $V = U$ 且 $QLIM_U = QLIM_D = 0$ 。

RATE_OFF	dU/dt	含义	输出 V	QLIM_D	QLIM_U
0	$< - DRLM$	输入值下降速度太快	$V-(DRLM * SAMPLE_T)$	1	0
0	- DRLM 到 URLM	允许变化率	U	0	0
0	$> URLM$	输入值 U 上升过快	$V+(URLM * SAMPLE_T)$	0	1
1	无含义	斜坡已禁用	U	0	0

调用 OB

这是在其中安装了该块的循环报警 OB (例如 OB32) 和 OB100。

错误处理

如果 $SAMPLE_T < 0$, 将输出 $ENO = 0$ 或 $QERR = 1$ 。

启动特征

在启动期间重置输出 V 。这意味着还要必须在启动 OB ($OB\ 100$) 中调用该块。

时间响应

必须从循环中断 OB 中调用块。

3.24.2 RAMP_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	允许值
DRLM	输出值的最大负变化 ([单元/秒])	REAL	3.0	I	DRLM < URLM
QERR	1 = 错误	BOOL	1	O	
QLIM_D	负梯度过陡	BOOL	0	O	
QLIM_U	正梯度过陡	BOOL	0	O	
RATE_OFF	变化率监视关闭	BOOL	0	I	
SAMPLE_T	采样时间 (秒)	REAL	1.0	I	> 0
U	模拟输入 (测量值)	REAL	0.0	I	
URLM	输出值的最大正变化 (单位/秒)	REAL	3.0	I	URLM > DRLM
V	斜坡输出	REAL	0.0	O	

3.25 RATIO_P: 比率控制

3.25.1 RATIO_P 的描述

对象名 (类型 + 编号)

FB70

- RATIO_P 块 I/O (页 289)
- RATIO_P 块图标 (页 699)
- RATIO_P 面板 (页 640)

功能

该块用于创建比率（例如，在比率控制中）。它还用来设置元件（例如，同步控制循环）或用于影响级联的引用变量。

工作原理

RATIO_P 块按照以下公式工作： $V = U1 * U2 + BIAS$

U1 通过互连获得，而 U2 根据内部/外部模式来选择。

内部/外部切换

模式通过以下方法进行选择，并在输出 QIN_EX 处加以指示：

- 设置输入 IN_EX，当 L_IE_ON = 0 且使能信号 IN_OP_EN 和 EX_OP_EN 有效时。
- 互连 L_IN_EX，当 L_IE_ON = 1 时。

内部：公式中的参数 U2 由操作员指定，施加限制 (U2_LL, U2_HL) 后将包括在公式中。需要操作员控制使能信号 U2_OP_EN。

外部：参数 U2 通过互连输入 U2_EXT 来指定，施加限制 (U2_LL, U2_HL) 后将包含到公式中。可控制输入 U2 跟踪 U2_EXT 以便顺利切换为“内部”。

3.25 RATIO_P: 比率控制

调用 OB

RATIO_P 块安装在其中包含了利用该结果的块的 OB 中。RATIO_P 块必须安装在此块之前（先计算，后使用）。

错误处理

运算错误由 ENO = 0 或 QERR = 1 指示。
操作员输入错误在输出 QOP_ERR 处集中指示。

启动特征

无特殊措施。

时间响应

如果具有动态响应的块的结果有意义（例如，比率控制、同步控制），则该块应安装在同一 OB 中的这些块之前。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“RATIO_P 的 VSTATUS (页 290)”。

消息响应

不可用

监视过程值

不可用

更多信息

更多相关信息，可参考：
RATIO_P 的 I/O (页 289)
RATIO_P 的 VSTATUS (页 290)
操作和监视 RATIO_P (页 290)

3.25.2 RATIO_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
BIAS	V 移动的份额	REAL	0	I	+	
EX_OP_EN	1 = 外部模式的操作员控制使能信号	BOOL	0	I	+	
IN_EX	操作员输入: 0 = 内部, 1 = 外部	BOOL	0	IO	+	
IN_OP_EN	1 = 内部模式的操作员控制使能信号	BOOL	0	I	+	
L_IE_ON	1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I	+	
L_IN_EX	IN_EX 的可互连输入	BOOL	0	I		
MO_U1HR	显示上限	REAL	110	I	+	
MO_U1LR	显示下限	REAL	-10	I	+	
QC_U1	U1 的质量代码	BYTE	16#80	I		
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	
QIE_OP	1 = 内部/外部切换的操作员控制使能信号	BOOL	0	O	+	
QIN_EX	0 = 内部, 1 = 外部	BOOL	0	O	+	
QOP_ERR	1 = 操作员错误输出	BOOL	0	O	+	
QU2_OP	1 = U2 操作员控制的操作员控制使能信号	BOOL	0	O	+	
QVHL	1 = 输出值 V 的上限已触发	BOOL	0	O	+	
QVLL	1 = 输出值 V 的下限已触发	BOOL	0	O	+	
U1	输入值	REAL	0	I	+	
U2	内部因子	REAL	1	IO	+	
U2_EXT	外部因子	REAL	1	I	+	
U2_HL	U2 的上限	REAL	1	I	+	
U2_LL	U2 的下限	REAL	0	I	+	
U2_OP_EN	1 = U2 操作员控制的操作员控制使能信号	BOOL	1	I		
V	输出值	REAL	0	O	+	

3.25 RATIO_P: 比率控制

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
V_HL	V 的上限	REAL	100	I	+	V_HL>V_LL
V_LL	V 的下限	REAL	0	I	+	V_LL<V_HL
USTATUS	VSTATUS 中的状态字, 可由用户组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.25.3 RATIO_P 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位 (0 - 15 位) 如下所示:

位号:	7	6	5	4	3	2	1	0
参数	QIN_EX	-	-			-	-	-
位号:	15	14	13	12	11	10	9	8
参数	-	-	-	-	-	-	QVLL	QVHL

16 位输入 USTATUS (数据类型为 WORD) 使用最高有效位 16 到 31。可根据需要使用这些位。

3.25.4 RATIO_P 的操作和监视

更多信息

更多相关信息, 请参考以下部分:

- RATIO_P 块图标 (页 699)
- RATIO_P 面板 (页 640)

3.26 READ355P: 从 FM 355 读取数字和模拟输入

3.26.1 READ355P 的描述

对象名 (类型 + 编号)

FB 72

- READ355P 块 I/O (页 293)

功能

块 READ355P 用于读取 FM 355 或 FM355-2 模块的数字和模拟输入。

依赖性

驱动程序生成器完成后, 此块以及块 FMCS_PID 和 FMT_PID 将位于 FM_CO 控制的块链中。

还要注意 FM_CO 块的依赖性。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 设置参数 CO_NO
- 输入 EN_CO 互连到 FM_CO 块的输出 EN_CO_x (x = 机架号)
- 输出 ENCO 互连到 FM_CO 块的输入 ENCOx_yy (x = 机架号, yy = 配位号)

工作原理

如果设置输入/输出参数 EN_CO.CO_ACT = CO_NO, 块将开始通过读取数据记录 (SFC RED_REC) 来确定以下值:

- CJ_TEMP 参数包含在冷接点处测量的冷接点温度 (摄氏度或华氏度, 其大小视所选温度单位而定)。如果未选择“热电偶”传感器类型或为所有使用“热电偶”传感器类型的模拟输入选择了设置冷接点温度, CJ_TEMP 参数的值将是“0.0”。
- 参数 DI_0 到 DI_7 表示数字输入 0 到 7 的实际状态。
- 参数 PV_PER_0 到 PV_PER_3 表示模拟输入 0 到 3 的值 (以 mA 或 mV 为单位)。
- 参数 PV_PHY_0 到 PV_PHY_3 表示预处理模拟输入值 0 到 3 (物理单位)。

启动特征

不可用

时间响应

不可用

消息响应

不可用

更多信息

更多相关信息, 可参考:

寻址 (页 293)

READ355P 的 I/O (页 293)

3.26.2 寻址

使用逻辑基础地址（通过 HW Config 设置，LADDR 输入）对属于实例的 FM 355 的控制器通道进行寻址。

FM 355 模块通过 PCS 7 库的块进行监视。MODE 输入将与 MOD_D1 块的 OMODE 输出相连。由于块只通过读取数据记录与 FM 355 进行通讯，因此 OMODE 输出低位字中的测量范围编码部分是不相关的，在此处设置为零。

3.26.3 READ355P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识：

加粗字体的 I/O 名称表示该 I/O 可见；正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息，可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
CJ_TEMP	在冷接点处测量的冷接点温度	REAL	0	O		
CO_NO	在冷接点处测量的冷接点温度	INT	0	O		
DI_x	数字输入 (x = 0 到 7)	BOOL	0	O		
EN_CO	当前配位号	STRUCT				
ENCO	配位号	BOOL	0	IO		
LADDR	逻辑地址 FM 355	INT	0	I		
MODE	模式	DWORD	0	I		
PV_PER_x	模拟输入 (x = 0 到 3)	REAL	0	O		
PV_PHY_x	物理模拟输入 (x = 0 到 3)	REAL	0	O		
QDONE	1 = 参数已读取	BOOL	0	O		
QMODF	1 = 模块错误	BOOL	0	O		
RET_VALU	返回值 SFC 59	INT	0	O		

3.27 SIG_SMTH: 低通滤波器

3.27.1 SIG_SMTH 描述

对象名 (类型 + 编号)

FB 144

- SIG_SMTH 块 I/O (页 298)

SIG_SMTH 的应用领域

该块用于下列应用:

- 二阶巴特沃兹低通滤波器
- 异常信号的检测和消除

工作原理

此块用作低通滤波器。此滤波器允许频率低于其截止频率的信号分量几乎无衰减地通过, 而高频信号分量将被衰减掉。

异常检测功能会监视相邻信号。如果检测到信号异常, 则将不再对其作进一步处理。该块输出上一个有效信号。

组态

使用 CFC 编辑器以将该块安装到循环中断 OB (OB 3x) 中。

不需要进一步寻址。

高级过程控制的模板包含如何使用 SIG_SMTH 的实例。

启动特征

调用 OB 100 时, 会根据当前过程值初始化巴特沃兹滤波器的状态变量。

时间响应

该块必须在循环中断 OB 中调用。块的采样时间由 CFC 在编译期间于 SampleTime 参数中设置。

消息响应

该块没有任何消息响应。

参见

SIG_SMTH 的故障处理 (页 297)

SIG_SMTH 的功能 (页 296)

SIG_SMTH 的模式 (页 295)

3.27.2 SIG_SMTH 的模式

SIG_SMTH 的模式

此块不提供任何模式。

参见

SIG_SMTH I/O (页 298)

SIG_SMTH 的故障处理 (页 297)

SIG_SMTH 的功能 (页 296)

SIG_SMTH 描述 (页 294)

3.27.3 SIG_SMTH 的功能

SIG_SMTH 的功能

该块提供以下功能:

- 重启低通滤波器
- 激活和禁止异常检测

重启低通滤波器

可以重新计算巴特沃兹滤波器的系数。为此, 必须重启滤波器 (Restart = 1)。

随后将重新初始化滤波器算法, 与启动 CPU 或者更改输入参数 TimeConstant 的计数值时完全相同。重新计算巴特沃兹滤波器的系数并预分配滤波器的内部状态存储器, 因而 CleanPV 输出参数等于 PV 输入参数。

激活和禁止异常检测

异常检测功能会监视 PV 过程值是否一致。设置 Out1DetOn = 1 输入参数以激活异常检测。

如果 PV 过程值中有异常信号, 此块会选择这些信号, 并且输出上一个有效值。

您可以通过 Out1Cycles 输入参数设置异常信号的数量, 经过这些异常信号后, 将输出上一个有效值。

参见

SIG_SMTH I/O (页 298)

SIG_SMTH 的故障处理 (页 297)

SIG_SMTH 的模式 (页 295)

SIG_SMTH 描述 (页 294)

3.27.4 SIG_SMTH 的故障处理

SIG_SMTH 的故障处理

ErrorNum I/O 可用于输出错误号。

错误号概述

此块输出的错误号:

错误号	错误号的含义
0	没有错误。
2	SampleTime < 0,001 [s]
32	PV 的值不再显示在实数域中, 或者它不是数字
72	TimeConstant < 2 · SampleTime

参见

SIG_SMTH I/O (页 298)

SIG_SMTH 的功能 (页 296)

SIG_SMTH 的模式 (页 295)

SIG_SMTH 描述 (页 294)

3.27.5 SIG_SMTH I/O

SIG_SMTH I/O

输入

I/O (参数)	描述	数据类型	默认值	属性
FilterOn	1 = 激活滤波器	BOOL	0	S7_m_c := 'false' S7_link := 'true'
OutlCycles	为避免特殊情况而设置的周期数	REAL	3	S7_m_c := 'false' S7_link := 'true'
OutlDetOn	1 = 特殊情况检测已激活	BOOL	0	S7_m_c := 'false' S7_link := 'true'
OutlThreshold	特殊情况检测的阈值	REAL	10	S7_m_c := 'false' S7_link := 'true'
PV	过程值	REAL	0	S7_m_c := 'false' S7_link := 'true'
Restart	复位滤波器算法	BOOL	0	S7_m_c := 'false' S7_link := 'true'
SampleTime	采样时间 [s] (自动分配)	REAL	0.1	S7_m_c := 'false' S7_link := 'true'
TimeConstant	巴特沃兹: 滤波时间为常量 [s]	REAL	10	S7_m_c := 'false' S7_link := 'true'

输出

I/O (参数)	描述	数据类型	默认值	属性
CleanPV	过程值的输出	REAL	0	S7_m_c := 'false' S7_link := 'true'
ErrorNum	输出当前错误号。 有关该块可以输出的错误号 的信息, 请参考“错误处理”。	INT	0	S7_m_c := 'false' S7_link := 'true'
OutIDetected	1 = 检测到特殊情况	BOOL	0	S7_m_c := 'false' S7_link := 'true'

参见

SIG_SMTH 的故障处理 (页 297)

SIG_SMTH 的功能 (页 296)

SIG_SMTH 的模式 (页 295)

SIG_SMTH 描述 (页 294)

3.28 SPLITR_P: 分割范围

3.28.1 SPLITR_P 的描述

对象名 (类型 + 编号)

FC272

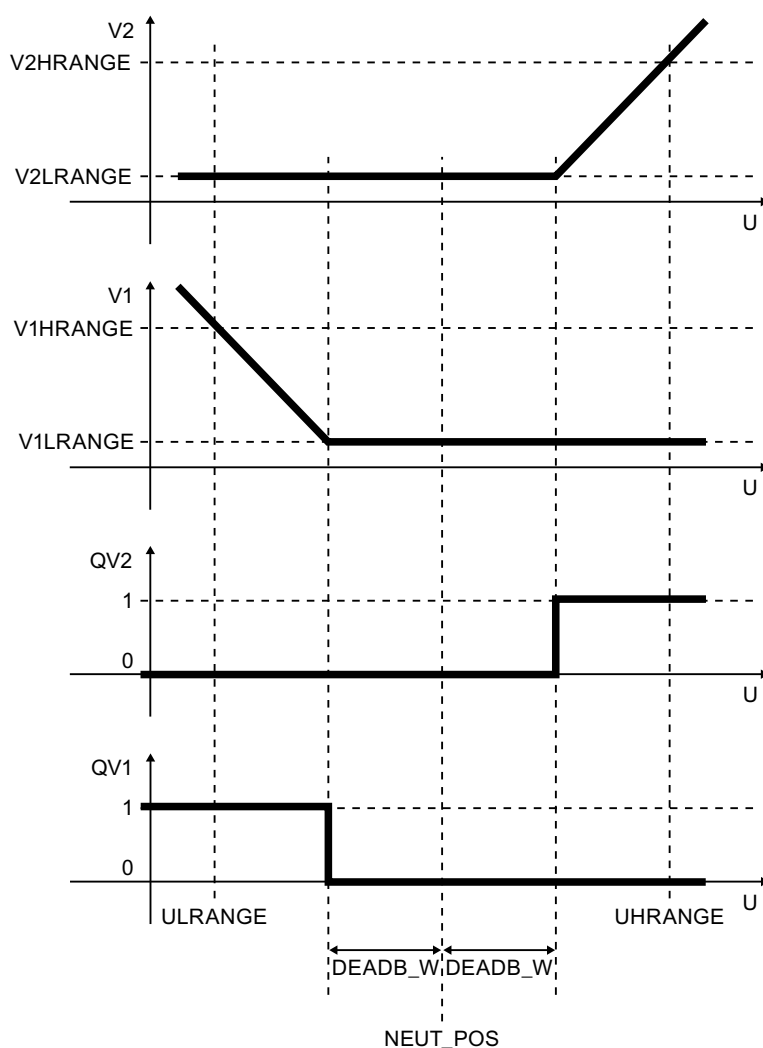
- SPLITR_P 块 I/O (页 302)

功能

该块与控制器块一起用于执行分割范围控制。

工作原理

该块安装在控制器块下游的运行顺序中。控制器块的控制器输出与 SPLITR_P 块的输入 U 互连。通过相应的参数设置常规位置和死区。通过组态 V1 和 V2 的上/下限来根据物理变量调整 V1 和 V2。传送特征具有以下外观：



SPLITR_P 的传送特征

调用 OB

控制器块在其中运行的 OB，在该 OB 中处理控制器块的调节变量。

错误处理

出现以下错误时输出 ENO = 0:

- V1 的计算不正确 (其中 V1 = V1LRANGE 也适用)
- V2 的计算不正确 (其中 V2 = V2LRANGE 也适用)

3.28.2 SPLITR_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的详细信息, 可参考“关于块描述的常规信息 (页 17)”部分。

I/O (参数)	含义	数据类型	默认值	类型	有效值
DEADB_W	死区宽度	REAL	10.0	I	
NEUT_POS	中间位置	REAL	50.0	I	> ULRANGE < UHRANGE
QV1	1 = 输出 1 激活	BOOL	0	O	
QV2	1 = 输出 2 激活	BOOL	0	O	
U	输入值	REAL	0.0	I	
UHRANGE	U 测量范围的上限	REAL	100.0	I	> ULRANGE
ULRANGE	U 测量范围的下限	REAL	0.0	I	< UHRANGE
V1	输出 1	REAL	0.0	O	
V1HRANGE	V1 测量范围的最终值	REAL	100.0	I	> V1LRANGE
V1LRANGE	V1 测量范围的初始值	REAL	0.0	I	< V1HRANGE
V2	输出 2	REAL	0.0	O	
V2HRANGE	V2 测量范围的最终值	REAL	100.0	I	> V2LRANGE
V2LRANGE	V2 测量范围的初始值	REAL	0.0	I	< V2HRANGE

3.29 SWIT_CNT: 切换周期计数器

3.29.1 SWIT_CNT 的描述

对象名 (类型 + 编号)

FB71

- SWIT_CNT 块 I/O (页 305)
- SWIT_CNT 块图标 (页 700)
- SWIT_CNT 面板 (页 643)

功能

SWIT_CNT 块可对单元的切换操作计数。

工作原理

块 SWIT_CNT 在输入 ON_OFF 的正沿 (0 → 1) 对切换操作计数, 并在输出 V 中输出结果。由于计数值是 DINT 数据类型, 因此将最大计数值将限制为 2 的 31 次方个切换操作。

调用 OB

调用 OB 位于同一 OB 中, 并在提供切换信号的块之后, 后者也存在于 OB 100 中。

跟踪

通过在 OS 设置 TRACK_OP 或互连 TRACK, 计数器输出 V 可跟踪值 VTRACK_OP。反过来, VTRACK_OP 可由操作员控制。

监视限制值

可在每个输入 VWH 和 VAH 设置警告或报警限制。超出这些限制时, 将显示计数值 V (在 QH_WRN/QH_ALM), 并在必要时生成消息。更多相关信息, 可参考“消息响应”。

3.29 SWIT_CNT: 切换周期计数器

错误处理

如果在 V 上发生计数器溢出，则设置一个周期的错误输出 QERR，并发送控制系统消息。然后继续从 0 开始计数。
操作员输入错误显示在 QOP_ERR 输出中。

启动特征

启动期间 (OB100)，设置输出 V = 0。启动之后，将在 RUNUPCYC 所设置的周期数内抑制消息。

时间响应

必须在所监视单元的控制块所在的同一运行组（如果已在 CFC 中组态）中执行该块。必须可靠检测沿。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“SWIT_CNT 的 VSTATUS (页 307)”。

消息响应

SWIT_CNT 块使用 ALARM_8P 块生成消息。

消息由监视切换操作限制的功能触发。

通过使用相关的 M_SUP_xx 输入可以分别抑制超出限制消息。可使用 MSG_LOCK 集中禁用过程消息（不是控制系统消息！）。

如果自上次重新启动后未满足 RUNUPCYC 周期数，并且如果 MSG_LOCK = TRUE 或 MSG_STAT = 21，则会设置 QMSG_SUP。

监视过程值

不可用

更多信息

更多相关信息，可参考：

SWIT_CNT 的 I/O (页 305)

SWIT_CNT 的消息文本和关联值 (页 306)

SWIT_CNT 的 VSTATUS (页 307)

操作和监视 SWIT_CNT (页 307)

3.29.2 SWIT_CNT 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
AUX_PRx	关联值 x	ANY	0	IO		
M_SUP_AH	1 = 抑制 HL 报警	BOOL	0	I	+	
M_SUP_WH	1 = 抑制 HL 警告	BOOL	0	I	+	
MO_VHR	显示上限	REAL	120	I	+	
MO_VLR	显示下限	REAL	0	I	+	
MSG_ACK	确认消息	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		
MSG_LOCK	1 = 锁定过程消息	BOOL	0	I	+	
MSG_STAT	消息错误信息	WORD	0	O		
ON_OFF	要计数的设备状态: 1 = 打开, 0 = 关闭	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
QC_ON_OFF	ON_OFF 的质量代码	BYTE	16#80	I		
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	
QH_ALM	1 = 超出报警上限	BOOL	0	O		
QH_WRN	1 = 超出警告上限	BOOL	0	O		
QMSG_ERR	1 = ALARM_8P 错误	BOOL	0	O	+	
QMSG_SUP	1 = 消息抑制	BOOL	0	O	+	
RUNUPCYC	起动周期数	INT	3	I		
TRACK	TRACK 的可互连输入	BOOL	0	I		
TRACK_OP	1 = 跟踪 V 至 VTRACK_OP	BOOL	0	IO	+	
V	切换周期数	DINT	0	O	+	
VAH	报警上限, 计数值	DINT	100	I	+	
VTRACK_OP	跟踪值	DINT	0	IO	+	> 0

3.29 SWIT_CNT: 切换周期计数器

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
VWH	警告上限, 计数值	DINT	95	I	+	
USTATUS	VSTATUS 中的状态字, 可由用户组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.29.3 SWIT_CNT 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由 此项抑制
1	QAH	\$\$BlockComment\$\$ 上上限报警	M	M_SUP_AH、 MSG_LOCK
2	QWH	\$\$BlockComment\$\$ 上限报警	M	M_SUP_WH、 MSG_LOCK

消息块的所有关联值 (AUX_PRx) 可由用户随意分配。

将关联值分配给块参数

关联值	块参数
1	AUX_PR01
2	AUX_PR02
3	AUX_PR03
4	AUX_PR04
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.29.4 SWIT_CNT 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	ON_OFF	-	-	-	-	MSG_LOCK	-	-
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	-	-	-	-	-	-

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.29.5 SWIT_CNT 的 操作员监控

更多信息

更多信息可参考以下部分：

- SWIT_CNT 块图标 (页 700)
- SWIT_CNT 面板 (页 643)

3.30 VAL_MOT: 电机阀门控制

3.30.1 VAL_MOT 的描述

对象名 (类型 + 编号)

FB 74

- VAL_MOT 块 I/O (页 314)
- VAL_MOT 块图标 (页 700)
- VAL_MOT 面板 (页 645)

功能

块 VAL_MOT 用于通过两个控制信号来控制电机驱动的阀门。阀门可以停在任何位置。可以选择对两个位置反馈信号（打开/关闭）进行监视。位置反馈信号由限位开关产生。

注意
对于 PCS 7, 不应在设备块和输出驱动程序之间插入其它块。 如果违背此原则, 应确保在与该块互连时, 从设备块的输出到输出驱动程序, 所有组成输出信号的块都安装在相同的 OB 中。

工作原理

可以使用各种输入来控制电机驱动的阀门。这些输入彼此之间以及与系统状态之间有着所定义的层级相关性。特别是，互锁、反馈或旋转方向监视以及电机保护断路器会影响控制信号 QSTART（1：电机开启，0：电机关闭）和 QOC（1：打开，0：关闭）。

下表显示了各个输入变量和事件按其对控制信号的影响而进行的优先级分配。后面的部分会提供更详细的信息。

优先级:	事件:
高	电机保护故障（如果 MSS_OFF = 1）
↑	监视错误（如果 FAULT_OFF = 1）
	旋转方向改变时的等待时间
	V_LOCK = 1
	VL_CLOSE = 1
	VL_OPEN = 1
↓	VL_HOLD = 1
低	自动/手动模式
无作用	电机保护故障（如果 MSS_OFF = 0）
	监视错误（如果 FAULT_OFF = 0）
	控制系统错误、操作员错误

手动/自动

如果设置了必要的使能信号，该模式通过 OS 上的操作员设置 AUT_ON_OP 或通过输入 AUT_L 上的互连来切换。设置的运行模式由输出 QMAN_AUT 加以指示（1：自动，0：手动）。

- **手动模式：** 此运行模式允许在 OS 处或通过可互连的输入进行控制。
 - **OS 操作员控制 (LINK_MAN = 0)：** OPEN_VAL 输入用于打开，CLOS_VAL 用于关闭，STOP_VAL 用于停止。必须启用所需的功能（OP_OP_EN、CL_OP_EN 或 ST_OP_EN）。
 - **通过可互连的输入操作 (LINK_MAN = 1)：** 在此情况下，各命令在输入 L_OPEN、L_CLOSE 和 L_STOP 上设置，例如，可以将这些输入互连以便能进行跟踪或本地控制。请注意，必须通过适合的逻辑设置开关 LINK_MAN、LIOP_SEL 和 AUT_L。
- **自动模式：** 互连的自动功能块会将其指令输出到输入 AUTO_ON（1：打开，0：关闭）或 AUTO_OC（1：打开，0：关闭）。

互锁

互锁功能的优先级高于其它所有控制输入，只有在设置相关使能信号后（MSS_OFF = 1，FAULT_OFF = 1），其优先级才会被电机保护故障或监视错误所超越。如果设置了 V_LOCK，电机驱动的阀门会被置于闲置位置（由 SS_POS 定义）。阀门分别通过 VL_OPEN 或 VL_CLOSE 来打开或关闭。VL_HOLD 会阻止自动和手动输入并保持上次的状态请求。各互锁输入的优先级在“工作原理”部分介绍。

监视

监视逻辑（通过 MONITOR = 1 启用）会验证设定值状态（在 QSTART 和 QOC 中确定）与阀门过程值反馈（由 FB_OPEN 和 FB_CLOSE 提供）之间的一致性。如果在监视时间 TIME_ON 过后未达到设定状态，则会设置输出 QMON_ERR。如果反馈无故（命令）发生变化，则会立即设置 QMON_ERR。

如果未连接任何反馈信号，必须将 MONITOR = 0 信号输出到监视功能。在此情况下，监视器假定在时间 TIME_ON 内已达到阀门的设定值状态。

如果监视功能工作正常，QOPENING 和 QCLOSING 输出将指示阀门是正在打开还是正在关闭。QOPENED 和 QCLOSED 指示阀门是否已到达最终位置。

如果阀门停止于中间位置，则以 QOPENING = 0 或 QCLOSING = 0 来指示移动方向，而 0 将指示最终位置。

FAULT_OFF 参数决定着监视错误的重要性。如果 FAULT_OFF = 1, 则在检测到故障时电机会被关闭, 阀门保持其当前位置。当 FAULT_OFF = 0 时, 此故障状态对控制输出没有影响, 因而块会像监视功能被关闭一样运行, 只会在 QMON_ERR 输出上指出监视错误。

TIME_OFF 参数决定着电机可再次启动前的等待时间。如果阀门已到达最终位置, 则 QSTART = FALSE。如果 QSTART = TRUE, 则直到所设的 TIME_OFF 期满后, 电机才会沿反方向重新启动。更多相关信息, 可参考“反转运行方向”。

电机保护

电机保护信号 MSS 的负沿将电机保护故障信号设为保持状态并将该信号传送给输出 QMSS_ST。MSS_OFF 参数决定着是只指示错误状态 (MSS_OFF = 0), 还是不管其它任何输入和系统状态如何都要关闭电机 (MSS_OFF = 1), 而且阀门在其当前位置处于闲置状态。

无波动切换

为了确保在所有运行模式下无波动切换到手动模式, 手动值 OPEN_VAL、CLOS_VAL 和 STOP_VAL 始终会跟踪到 QSTART 和 QDIR 的当前值 (例外: 反转旋转方向)。

反转行程方向

如果阀门在到达其最终位置前运行方向发生反转, 将会采取以下措施:

- 电机停止 (QSTART = 0)。
- 如果 OFF 监视器不报告错误, 则内部 OFF 监视器在等到经过 TIME_OFF 时间段后会以反方向启动电机。

错误处理

电机保护错误 (QMSS_ST = 1) 和监视错误 (QMON_ERR = 1) 将被报告给 OS, 并且按上述模式影响块的工作。操作员可以通过 RESET 将这些状态复位, 也可以通过将 L_RESET 与信号“1”互连, 在 MSS 的下一个正沿自动复位。控制系统故障 CSF 只会报告给 OS 并与电机保护错误和监视错误一起应用到组错误 QGR_ERR。它不会对块算法产生任何进一步的影响。

操作员错误在输出 QOP_ERR 上指示, 但不会出现消息。

调用 OB

安装了该块的循环中断 OB（例如 OB32）和 OB100。

错误状态后的启动

错误状态后的启动取决于复位时的模式设置：

- 在自动模式下，只有将监视或电机保护错误复位并由自动化系统输出相应的启动信号，电机阀门才能再次启动。
- 在手动模式下，由于手动操作已被跟踪到“HOLD”，所以必须明确启动电机。

启动特征

CPU 启动时，VAL_MOT 块会切换为手动模式，并输出 HOLD 命令。这意味着必须在启动 OB 中调用该块。在 CFC 工程组态中，这由 CFC 处理。使用 STEP 7 基本工具时，在启动 OB 中输入此调用。启动后，消息会在 RUNUPCYC 所设置的周期数内受到抑制。

时间响应

块必须在循环中断 OB 中调用。块的采样时间在 SAMPLE_T 参数中设置。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“VAL_MOT 的 VSTATUS (页 318)”。

消息响应

VAL_MOT 块使用 ALARM_8P 块生成消息。

存在下列消息触发器：

- 控制系统错误
- 电机断路器故障和监视错误（运行错误）
- 通过互连作为控制系统错误而接收的 CSF 信号

如果重新启动后未满足 RUNUPCYC 个周期，或是 MSG_STAT = 21，则会设置 QMSG_SUP。

监视过程值

不可用

更多信息

更多相关信息, 可参考:

VAL_MOT 的 I/O (页 314)

VAL_MOT 的消息文本和关联值 (页 317)

VAL_MOT 的 VSTATUS (页 318)

操作和监视 VAL_MOT (页 318)

3.30.2 VAL_MOT 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
AUT_L	用于 MANUAL (手动) /AUTO (自动) 模式的互连输入: (0: 手动/1: 自动)	BOOL	0	I		
AUT_ON_OP	操作员输入: (0 = 手动/1 = 自动)	BOOL	0	IO	+	
AUTO_OC	AUTOMATIC 模式旋转方向: 1 = 打开, 0 = 关闭	BOOL	0	I		
AUTO_ON	AUTOMATIC 值: 1: 开	BOOL	0	I		
AUTOP_EN	1 = 自动模式的操作员控制使能信号	BOOL	1	I		
AUX_PRx	关联值 x	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 当前批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	0	I	+	
CL_OP_EN	1 = 关闭阀的操作员控制使能信号	BOOL	1	I		
CLOS_VAL	操作员输入: 1 = 关闭阀	BOOL	0	IO	+	
CSF	1 = 外部错误	BOOL	0	I		
FAULT_OFF	1 = 发生故障时关闭电机	BOOL	1	I		
FB_CLOSE	反馈: 1 = 已关闭	BOOL	0	I		
FB_OPEN	反馈: 1 = 打开	BOOL	0	I		
L_CLOSE	AUTOMATIC 值 1 = 关闭 (CLOSE) 阀	BOOL	0	I		
L_OPEN	AUTOMATIC 值 1 = 打开 (OPEN) 阀	BOOL	0	I		
L_RESET	可互连的 RESET 输入	BOOL	0	I		
L_STOP	AUTOMATIC 值 1 = 停止 (STOP) 阀	BOOL	0	I		
LINK_MAN	0 = 操作员输入激活, 1 = 通过 L_OPEN、L_CLOSE 和 L_STOP 进行手动控制	BOOL	0	I		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
LIOP_SEL	用于手动/自动切换 (AUT_L) 的可互连输入: 1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
MANOP_EN	1 = 手动模式的操作员控制使能信号	BOOL	1	I		
MONITOR	1 = 监视开启	BOOL	1	I	+	
MSG_ACK	消息已确认	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		
MSG_STAT	消息错误信息	WORD	0	O		
MSS	电机保护断路器 (低态有效, 即 0 = 错误)	BOOL	0	I		
MSS_OFF	1 = 发生 MSS 故障时: 电机停止	BOOL	1	I		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
OP_OP_EN	1 = 打开阀的操作员控制使能信号	BOOL	1	I		
OPEN_VAL	操作员输入: 1 = 打开阀门	BOOL	0	IO	+	
QAUTOP	1 = 自动模式的操作员控制使能信号	BOOL	0	O	+	
QC_FB_CLOS E	FB_CLOSE 的质量代码	BYTE	16#80	I		
QC_FB_OPE N	FB_OPEN 的质量代码	BYTE	16#80	I		
QC_QSTART	QSTART 的质量代码	BYTE	16#80	O		
QC_QSTART_ I	输出 QSTART 的质量代码	BYTE	16#80	I		
QC_QOC	QOC 的质量代码	BYTE	16#80	O		
QC_QOC_I	输出 QOC 的质量代码	BYTE	16#80	I		
QCL_OP	1 = 关闭阀的操作员控制使能信号	BOOL	0	O	+	
QCLOSED	1 = 阀门已关闭	BOOL	0	O	+	
QCLOSING	1 = 阀门正在关闭	BOOL	0	O	+	
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	
QGR_ERR	1 = 组错误	BOOL	0	O		
QMAN_AUT	0 = 手动, 1 = 自动	BOOL	0	O	+	

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
QMANOP	1 = 手动模式的操作员控制使能信号	BOOL	0	O	+	
QMON_ERR	1 = 监视错误	BOOL	0	O	+	
QMSG_ERR	1 = 消息错误	BOOL	0	O	+	
QMSG_SUP	1 = 消息抑制激活	BOOL	0	O	+	
QMSS_ST	存储的电机保护断路器 (1 = 错误)	BOOL	0	O	+	
QOC	方向控制输出: 1 = 打开 (OPEN)	BOOL	0	O		
QOP_ERR	1 = 组操作员输入错误	BOOL	0	O		
QOP_OP	1 = 打开阀的操作员控制使能信号	BOOL	0	O	+	
QOPENED	1 = 阀门已打开 (OPENED)	BOOL	0	O	+	
QOPENING	1 = 阀门正在打开 (OPENING)	BOOL	0	O	+	
QST_OP	1 = 关闭阀门的操作员控制使能信号	BOOL	0	O	+	
QSTART	控制输出: 1 = 电机开启	BOOL	0	O	+	
RESET	已启用错误复位输入	BOOL	0	IO	+	
RUNUPCYC	起动周期数	INT	3	I		
SAMPLE_T	采样时间 (秒)	REAL	1.0	I		> 0
SS_POS	0 = 空闲位置已关闭	BOOL	0	I		
ST_OP_EN	1 = 关闭阀门的操作员控制使能信号	BOOL	1	I		
STEP_NO	BATCH 步号	DWORD	0	I	+	
STOP_VAL	操作员输入: 1 = 停止阀门	BOOL	0	IO	+	
TIME_OFF	电机关闭监视时间 (秒)	REAL	3.0	I	+	≥ 0
TIME_ON	阀门运行监视时间 (秒)	REAL	3.0	I	+	≥ 0
V_LOCK	1 = 锁定 (SS_POS)	BOOL	0	I	+	
VL_CLOSE	1 = 锁定 (已关闭)	BOOL	0	I	+	
VL_HOLD	1 = 锁定 (保持/禁用)	BOOL	0	I	+	
VL_OPEN	1 = 锁定 (打开)	BOOL	0	I	+	
USTATUS	VSTATUS 中的状态字, 可由用户组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.30.3 VAL_MOT 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由 此项抑制
1	QMSS_ST	\$\$BlockComment\$\$ 电机保护	S	-
2	QMON_ERR	\$\$BlockComment\$\$ 监视故障	S	-
3	CSF	\$\$BlockComment\$\$ 外部故障	S	-

将关联值分配给块参数

消息块的前三个关联值已分配给 SIMATIC BATCH 数据，其余的关联值 (AUX_PRx) 可由用户随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	AUX_PR04
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.30 VAL_MOT: 电机阀门控制

3.30.4 VAL_MOT 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	QMON_ERR	-	QMSS_ST	-	QMAN_AUT	-	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	-	QCLOSING	QOPENING	QCLOSED	QOPENED	V_LOCK

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.30.5 VAL_MOT 的操作和监视

更多信息

更多相关信息，请参考以下部分：

- VAL_MOT 块图标 (页 700)
- VAL_MOT 面板 (页 645)

3.31 VALVE: 阀控制

3.31.1 VALVE 的描述

对象名 (类型 + 编号)

FB73

- VALVE 块 I/O (页 323)
- VALVE 块图标 (页 701)
- VALVE 面板 (页 648)

功能

块 VALVE 用于通过一个控制信号 (打开/关闭) 来操作控制阀门 (打开/关闭配件)。阀门的静止位置可以是已关闭或已打开状态。可以选择对两个位置反馈信号 (打开/关闭) 进行监视。位置反馈信号由限位开关产生。

注意

对于 PCS 7, 不应在设备块和输出驱动程序之间插入其它块。

如果违背此原则, 应确保在与该块互连时, 从设备块的输出到输出驱动程序, 所有组成输出信号的块都安装在相同的 OB 中。
--

工作原理

可以使用各种输入来控制阀门。依彼此之间以及与阀状态的定义的层级关系来使用这些输入。具体来说，锁定及反馈监视会影响控制信号 QCONTROL。

下表显示了各个输入变量和事件按其对控制信号的影响而进行的优先级分配。后面的部分会提供更详细的信息。

优先级:	事件:
高	V_LOCK = 1
↑	VL_CLOSE = 1
	VL_OPEN = 1
↓	监视错误 (如果 FAULT_OFF = 1)
低	自动/手动模式
无作用	监视错误 (如果 FAULT_OFF = 0)
	控制系统错误、操作员错误

静止位置

被控制阀门的静止位置由输入 SS_POS (1: 打开, 0: 关闭) 的信号控制。这只影响控制输出 QCONTROL (0: 静止位置, 阀门已终止) 的定义。输入端的命令则不会受到影响 (输入端的信号“1”始终表示“打开”)。

实例: 当 SS_POS = 1 (阀门处于静止位置“打开”) 时, 控制输出 QCONTROL = 1 信号会执行“关闭阀门”命令。

手动/自动

如果设置了必要的使能信号, 该模式通过 OS 上的操作员设置 AUT_ON_OP 或通过输入 AUT_L 上的互连来切换。设置的运行模式由输出 QMAN_AUT 加以指示 (1: 自动, 0: 手动)。

- **手动模式:** 输入 MAN_OC 通过 OS 进行操作。必须设置相应的使能参数 (OP_OP_EN 或 CL_OP_EN)。
- **自动模式:** 自动化系统的互连会将控制命令输出到输入 AUTO_OC (1: 打开, 0: 关闭)。

互锁

互锁功能的优先级高于其它所有信号和错误。如果设置了 V_LOCK, 则阀门会设置到其静止位置 (QCONTROL = 0)。如果未设置 V_LOCK, 也可通过输入 VL_OPEN 和 VL_CLOSE 来直接选择锁定状态 (打开/关闭)。信号 VL_CLOSE 会将 VL_OPEN 锁定。

监视

监视逻辑会验证输出控制命令 QCONTROL 与阀门的过程值反馈 (FB_OPEN、FB_CLOSE) 之间的一致性。如果在监视时间 TIME_ON 过后未到达最终位置, 则会设置输出 QMON_ERR。如果反馈无故 (命令) 发生变化, 则会立即设置 QMON_ERR。阀门将被设置为静止位置 (断电)。

如果未连接任何限制反馈信号, 则必须将 MONITOR = 0 信号输出给监视功能, 随后, 监视功能会假定在时间 TIME_MON 内已达到阀门限制。直到此时, 才会显示 QOPENING 或 QCLOSING。

如果监视功能工作正常, QOPENING 和 QCLOSING 输出将指示阀门是正在打开还是正在关闭。QOPENED 和 QCLOSED 指示阀门是否已到达最终位置。

输入 NO_FB_CL 和 NO_FB_OP 用于设置没有“打开”和“关闭”状态的反馈 (NO_FB_xx = 1)。输入 NOMON_CL 和 NOMON_OP 用于禁用因存在限位开关故障而对现有反馈进行的监视 (NOMON_xx = 1)。

参数 FAULT_SS 定义了监视错误的重要性。如果 FAULT_SS = 1, 出现错误时电机将停到 SS_POS 中所定义的静止位置。如果 FAULT_SS = 0, 则错误对控制输出没有影响。

无波动切换

为了确保无波动切换到手动模式, 手动值 MAN_OC 始终会跟踪到 QCONTROL 的当前值。

错误处理

监视错误 (QMON_ERR = 1) 将被报告给 OS, 并对上述块工作原理产生影响。它可以通过运行 RESET 复位, 也可利用 L_RESET 正沿通过互连来自动复位。控制系统故障 CSF 只会报告给 OS 并同监视错误一起应用于组错误 QGR_ERR。它不会对块算法产生任何进一步的影响。

操作员错误在输出 QOP_ERR 上指示, 但不会出现消息。

调用 OB

安装了该块的周期性中断 OB（例如 OB32）和 OB100。

错误状态后的启动

错误状态后的启动取决于复位时的模式设置：

- 在自动模式下，只有将监视或电机保护错误复位并由自动化系统输出相应的启动信号，电机阀门才能再次启动。
- 在手动模式下，由于手动操作已被跟踪到“HOLD”，所以必须明确启动电机。

启动特征

CPU 启动期间，VALVE 块将切换为手动模式，并输出 QCONTROL= 0（静止位置）信号。这意味着必须在启动 OB 中调用该块。在 CFC 工程组态中，这由 CFC 处理。使用 STEP 7 基本工具时，在启动 OB 中输入此调用。启动后，消息会在 RUNUPCYC 所设置的周期数内受到抑制。

在 START_SS 输入参数中，可以决定是在启动 CPU 时将阀门设置为安全状态 (START_SS = 1)，还是保留其上次的操作状态。

时间响应

块必须在循环中断 OB 中调用。块的采样时间在 SAMPLE_T 参数中设置。

32 位状态字 VSTATUS 的分配

更多相关信息，可参考“VALVE 的 VSTATUS (页 327)”。

消息响应

VALVE 块使用 ALARM_8P 块生成消息。

消息由以下控制系统错误触发：

- 监视错误（运行错误）
- 通过互连作为控制系统错误而接收的 CSF 信号

如果重新启动后未满 RUNUPCYC 个周期，或是 MSG_STAT = 21，则会设置 QMSG_SUP。

监视过程值

不可用

更多信息

更多相关信息，可参考：

VALVE 的 I/O (页 323)

VALVE 的消息文本和关联值 (页 326)

VALVE 的 VSTATUS (页 327)

操作和监视 VALVE (页 327)

3.31.2 VALVE 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识：

加粗字体的 I/O 名称表示该 I/O 可见；正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息，可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
AUT_L	用于 MANUAL (手动) /AUTO (自动) 模式的可互连输入：0 = 手动；1 = 自动	BOOL	0	I		
AUT_ON_OP	操作员输入：0 = 手动；1 = 自动	BOOL	0	IO	+	
AUTO_OC	自动模式下的旋转方向：1 = 打开，0 = 关闭	BOOL	0	I		
AUTOP_EN	1 = 自动模式的操作员控制使能信号	BOOL	1	I		
AUX_PRx	关联值 x	ANY	0	IO		
BA_EN	由 BATCH 发布	BOOL	0	I	+	
BA_ID	BATCH: 连续批次号	DWORD	0	I	+	
BA_NA	BATCH 名称	STRING[32]	0	I	+	
CL_OP_EN	1 = CLOSE 的操作员控制使能信号	BOOL	1	I		
CSF	1 = 外部错误	BOOL	0	I		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
FAULT_SS	1 = 出错时移动到安全位置	BOOL	1	I		
FB_CLOSE	反馈: 1 = 已关闭	BOOL	0	I		
FB_OPEN	反馈: 1 = 打开	BOOL	0	I		
L_RESET	可互连的 RESET 输入	BOOL	0	I		
LIOP_SEL	用于手动/自动切换 (AUT_L) 的可互连输入: 1 = 互连激活, 0 = 操作员控制激活	BOOL	0	I		
MAN_OC	操作员输入: 0 = 关闭, 1 = 打开	BOOL	0	IO	+	
MANOP_EN	1 = 手动模式的操作员控制使能信号	BOOL	1	I		
MONITOR	1 = 监视开启, 0 = 监视关闭	BOOL	1	I	+	
MSG_ACK	消息已确认	WORD	0	O		
MSG_EVID	消息号	DWORD	0	I		
MSG_STAT	消息错误信息	WORD	0	O		
NO_FB_CL	1 = 不存在反馈“已关闭”	BOOL	0	I		
NO_FB_OP	1 = 不存在“已打开”反馈	BOOL	0	I		
NOMON_CL	1 = 不监视“已关闭”反馈	BOOL	0	I		
NOMON_OP	1 = 不监视“已打开”反馈	BOOL	0	I		
OCCUPIED	BATCH 占用的 ID	BOOL	0	I	+	
OOS	预留	BOOL	0	I	+	
OP_OP_EN	1 = OPEN 的操作员控制使能信号	BOOL	1	I		
QAUTOP	1 = 自动模式的操作员控制使能信号	BOOL	0	O	+	
QC_FB_CLOSE	FB_CLOSE 的质量代码	BYTE	16#80	I		
QC_FB_OPEN	FB_OPEN 的质量代码	BYTE	16#80	I		
QC_QCONTROL	QCONTROL 的质量代码	BYTE	16#80	O		
QC_QCONTROL_I	输出 QCONTROL 的质量代码	BYTE	16#80	I		
QCL_OP	1 = CLOSE 的操作员控制使能信号	BOOL	0	O	+	
QCLOSED	1 = 阀门已关闭	BOOL	0	O	+	
QCLOSING	1 = 阀门正在关闭	BOOL	0	O	+	
QCONTROL	控制输出: 0 = 静止位置	BOOL	0	O		

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O	+	
QGR_ERR	1 = 组错误	BOOL	0	O		
QMAN_AUT	0 = 手动, 1 = 自动	BOOL	0	O	+	
QMANOP	1 = 手动模式的操作员控制使能信号	BOOL	0	O	+	
QMON_ERR	1 = 监视错误	BOOL	0	O	+	
QMSG_ERR	1 = 消息错误	BOOL	0	O	+	
QMSG_SUP	1 = 消息抑制激活	BOOL	0	O	+	
QOP_ERR	1 = 组操作员输入错误	BOOL	0	O		
QOP_OP	1 = OPEN 的操作员控制使能信号	BOOL	0	O	+	
QOPENED	1 = 阀门已打开	BOOL	0	O	+	
QOPENING	1 = 阀门正在打开	BOOL	0	O	+	
RESET	已启用错误复位输入	BOOL	0	IO	+	
RUNUPCYC	起动周期数	INT	3	I		
SAMPLE_T	采样时间 (秒)	REAL	1.0	I		> 0
SS_POS	静止位置: 0 = 关闭 (类型 C): 1 = 打开 (类型 O)	BOOL	0	I		
START_SS	1 = 安全状态和手动模式下的启动	BOOL	1	I		
STEP_NO	BATCH 步号	DWORD	0	I	+	
TIME_MON	监视时间 (秒)	REAL	3.0	I	+	≥ 0
V_LOCK	1 = 锁定 (SS_POS)	BOOL	0	I	+	
VL_CLOSE	1 = 锁定 (已关闭)	BOOL	0	I	+	
VL_OPEN	1 = 锁定 (打开)	BOOL	0	I	+	
USTATUS	VSTATUS 中的状态字, 可由用户组态	WORD	0	I		
VSTATUS	块图标中的扩展状态显示	DWORD	0	O	+	

3.31.3 VALVE 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别	可由 此项抑制
1	QMON_ERR	\$\$BlockComment\$\$ 监视故障	S	-
2	CSF	\$\$BlockComment\$\$ 外部故障	S	-

将关联值分配给块参数

消息块的前三个关联值已分配给 SIMATIC BATCH 数据，其余的关联值 (AUX_PRx) 可由用户随意分配。

关联值	块参数
1	BA_NA
2	STEP_NO
3	BA_ID
4	AUX_PR04
5	AUX_PR05
6	AUX_PR06
7	AUX_PR07
8	AUX_PR08
9	AUX_PR09
10	AUX_PR10

3.31.4 VALVE 的 VSTATUS

该 32 位状态字扩展了块图标和面板中的状态显示。块使用的最低 16 个有效位（0 - 15 位）如下所示：

位号:	7	6	5	4	3	2	1	0
参数	QMON_ERR	-			QMAN_AUT	-	BA_EN	OCCUPIED
位号:	15	14	13	12	11	10	9	8
参数	OOS	QMSG_SUP	-	QCLOSING	QOPENING	QCLOSED	QOPENED	V_LOCK

16 位输入 USTATUS（数据类型为 WORD）使用最高有效位 16 到 31。可根据需要使用这些位。

3.31.5 VALVE 的操作和监视

更多信息

更多相关信息，请参考以下部分：

- VALVE 块图标 (页 701)
- VALVE 面板 (页 648)

系列: CONTROL

3.31 VALVE: 阀控制

系列： DRIVER

4.1 关于使用驱动程序块的注意事项

概述

- 有关驱动程序块的描述指定了安装这些块的 OB。请注意，不会为所有 CPU 生成所有列出的 OB。更多相关信息，可参考特定 OB 的在线帮助。
- 如果驱动程序生成器使用 PCS 7 库的驱动程序块，则 CPU 上需要 V3.1 版本或更高版本的固件。
- CFC 功能“生成模块驱动程序”会自动互连并组态所需的 I/O。例如，如果在编译程序时检测到硬件修改，就会调用并执行此功能。

说明

请注意，有些模拟输出模块不支持“响应 CPU 停止”。在这种情况下，如果使用模块驱动程序，会将输入 START_ON 设置为“0”并删除与该参数的互连。

信号处理块

可用 PCS 7 库的驱动程序块为信号处理提供了三种类型的通道块：

1. 标准通道块：

CH_AI、CH_AO、CH_DI、CH_DO

这些块只用于处理 S7-300/400 SM 模块的信号。如果要优化内存和运行利用率，且无需处理任何 PA 设备，则可使用这些标准块。

2. 通用通道块：

CH_U_AI、CH_U_AO、CH_U_DI、CH_U_DO

这些块用于处理 S7-300/400 SM 模块或 PA 现场设备的信号。这些块的优点在于可以创建 CFC 图表，而无需考虑以后要使用的硬件 I/O。通用块的缺点是需要占用更多内存和运行时间。这些块没有消息响应。

4.1 关于使用驱动程序块的注意事项

3. PA 通道块:

PA_AI、PA_AO、PA_DI、PA_DO、PA_TOT

专门为与 PA 现场设备配合使用而设计。如果要利用这些设备的特殊功能时, 尤其应该使用这些块。与 CH 块不同, 依照在硬件配置中所选的期望设备组态, PA 通道块不仅处理实际信号, 而且处理所有变量。

4. FF 通道块:

FF_A_AI、FF_A_AO、FF_A_DI、FF_A_DO

专门为与 PA 现场设备和 AB7000 PROFIBUS 从站配合使用而设计。这些块在用法和功能方面都与相应的 PA 通道块类似。

5. 特定通道块:

CH_CNT、CH_CNT1、CH_MS

某些特殊应用需要使用这些块, 例如, 控制和读取 ET 200iSP 的 FM 350-1/-2 模块和 8-DI NAMUR 模块的计数或频率值, 以及处理 ET 200S 电机启动器模块的信号。

4.2 CH_AI: 模拟值输入

4.2.1 CH_AI 的描述

对象名 (类型 + 编号)

FC275

- CH_AI 块 I/O (页 338)

应用领域

CH_AI 块用于处理 S7-300/400 SM 模拟输入模块的模拟输入值信号。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32) 和重启 OB100。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- MODE 输入与 MOD 块的相应 OMODE_xx 输出互连。

功能和工作原理

循环块 CH_AI 处理模拟输入模块的所有通道特定信号功能。

块从过程映像（分区）读取原始模拟值，然后根据此原始值将其转换为相应的物理值或计算百分比值。MODE (页 709) 输入用于定义供处理的原始值的格式。如果 MODE 输入参数的高位字节 = 16#40（值状态 = 较高级别错误，QMOD_ERR = TRUE），则将原始值视为无效。

程序将生成结果值的质量代码，可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
上一个有效值	16#44
替代值	16#48
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成的。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

寻址

将在 HW Config（符号表）中为模拟输入通道生成的符号连接到 VALUE 输入参数。

原始值检查

根据测量类型和模拟输入模块的范围，标称范围设置了模拟信号转换为数字值（原始值）的范围。这也包括过冲/下冲范围，在该范围内模拟信号仍可转换。如果值超出此范围，就构成上溢或下溢。块会指出原始值是否在模块的标称范围内。

如果值超出了标称范围的下限，输出参数 QCHF_LL = TRUE。如果值超出了标称范围的上限，输出参数 QCHF_HL = TRUE。如果由于模块诊断“过冲/下冲测量范围”而出现通道错误，QCHF_LL/QCHF_HL 将保持设置为 TRUE。

发生信号上溢或下溢错误时也会设置 QBAD = TRUE（通道错误）。

说明

模块对于 4 mA 到 20 mA 信号线的断路反应不一致。根据模块不同，会将 16#7FFF（上溢）或 16#8000（下溢）作为原始值写入过程映像。CH_AI 通道块相应地输出上溢 (QCHF_HL = TRUE) 或下溢 (QCHF_LL = TRUE) 以及 QBAD = TRUE。

例外：如果在 HW Config 中对模拟输入模块启用了“诊断中断”，则只有在由于检测到“通道错误”（例如，断路）而触发诊断中断的情况下，才会设置 QBAD = TRUE。

NAMUR 限制检查

模拟信号处理的 NAMUR 准则为有通道错误的零点（4 mA 到 20 mA）模拟信号定义以下限制：

$$3.6 \text{ mA} \leq \text{模拟信号} \leq 21 \text{ mA}$$

上述 NAMUR 限制设置为限制值监视的固定默认值。可以通过设置输入参数 CH_F_ON = TRUE，并以 [mA] 为单位在 CH_F_HL 和 CH_F_LL 输入参数中设置对应的新限制来定义其它限制。如果零点模拟信号超出电流上下限范围，则 QBAD = TRUE。

说明

可选择的限制必须在模块的过冲和下冲范围内。如果模块不自动限制测量值，也有可能

出现超出 NAMUR 范围的值。

正常值

原始值根据输入参数 VLRANGE、VHRANGE 和 MODE 中的设置转换为相应的物理值（更多信息，可参考“OMODE_xx (页 719)”）。这些值将写入输出 OVL RANGE 和 OVHRANGE，以允许 VLRANGE 和 VHRANGE 的设置与其它块 I/O 互连。转换算法取决于线性输入信号。如果 VLRANGE = 0 且 VHRANGE = 100，则转换为百分比值。如果设置了 VHRANGE = VLRANGE，则将按照 MODE (页 709) 设置获取模拟输入模块的输入信号（例如，mA）。如果原始值已经是物理值，则设置 VLRANGE = 0 和 VHRANGE = 1。质量代码设置为 QUALITY = 16#80。

在 PTC 测量模式下运行时，模拟值包含编码的二进制信号。输出提供以下信息：

- 如果被测电阻在正常范围内，则 PV_Out = 0.0。
- 如果被测电阻在预警范围内，则 PV_Out = 4.0。
- 如果被测电阻在工作范围内，则 PV_Out = 1.0。

这只有在设置输入参数 VLRANGE = 0 且 VHRANGE = 1 时才适用。仿真值 SIM_V 和替代值 SUBS_V 仅应设为 0 或 1。

说明

在“热电偶值的外部或内部比较”测量模式下，物理单元将调整到 S7 300 模块的 +/- 80 mV 范围。必须通过相应的转换表确定温度。

模块返回以 [mV] 为单位的物理当量作为原始值。VHRANGE 和 VLRANGE 设置为 +/- 80 mV。

仿真

如果输入参数 SIM_ON = TRUE, 则输出 SIM_V 输入参数值和质量代码 QUALITY = 16#60。QBAD = TRUE: 由于存在更高优先级错误而复位。还必须在仿真模式下在 MODE (页 709) 输入的低位字中设置有效运行模式。否则, 将输出 QBAD = 1。仿真拥有最高优先级。仿真值将根据运行模式以及输入参数 VHRANGE 和 VLRANGE 转换为原始值。该值的验证方式与过程映像中原始值的验证方式相同。这允许仿真 QBAD、QCHF_LL 和 QCHF_HL 状态。

如果 QBAD 会在单极测量范围的负范围内形成, 则必须将该值设置为 -119%。

如果 VLRANGE > VHRANGE, 则不能仿真状态 QBAD = TRUE。按照 SIM_V 值设置 QCHF_LL 和 QCHF_HL 输出。如果块处于仿真状态下, 则 QSIM = TRUE。

说明

请记住, 仿真值总是在仿真模式下输出, 与任一参数 LAST_ON (替代值) 或 SUBS_ON (上一个有效值) 都无关。

替代值

如果输入参数 SUBS_ON = TRUE, LAST_ON = FALSE 且原始值无效, 那么程序将输出输入参数 SUBS_V 的值。质量代码将设置为 QUALITY = 16#48 且 QBAD = 1。

保持上一个值, V_DELTA 参数

程序将在输入参数 LAST_ON = TRUE 和 SUBS_ON = FALSE 时输出上一个输出值或倒数第二个有效输出值 (V_LAST, VLAST1), 这取决于 V_DELTA 参数中的设置。可以在 V_DELTA 参数中定义有效过程值更改。通过设置 $V_DELTA \leq 0$ 可禁用此功能。以下条件适用:

无效原始值和 $V_DELTA > 0$ 的规则:

- $ABS(V - V_LAST) > V_DELTA$: $V = V_LAST1$ (倒数第二个有效输出值), $DELTA_ON = 1$
- $ABS(V - V_LAST) \leq V_DELTA$: $V = V_LAST$ (上一个有效输出值), $DELTA_ON = 0$
- 质量代码将设置为 QUALITY = 16#44 且 QBAD = 1

有效原始值和 $V_DELTA > 0$ 的规则:

V_DELTA 还用来限制有效原始值的更改。如果两次调用期间值的更改大于 V_DELTA , 则上一个值 (V_LAST) 将在输出 V 中保留一个周期的时间。应该十分小心地选择 V_DELTA 值。如果该值过小, 质量代码可能会在 16#80 和 16#44 之间波动, 无论原始值是否正确。

- $ABS(V - V_LAST) > V_DELTA$: 持续一个周期时间 $V = V_LAST$
- 质量代码将设置为 $QUALITY = 16\#44$ 、 $DELTA_ON=1$ 和 $QBAD = 0$ 。

输出无效值

如果输入参数 $SUBS_ON$ 和 $LAST_ON$ 两者均为 $FALSE$, 或均为 $TRUE$, 并且存在一个无效过程值, 则会输出此无效值并将 $QBAD$ 设置为 1。

延迟值接受

质量代码从“BAD”更改为“GOOD”时或者重新启动后, 如果 CNT_LIM 中指定的周期数没有结束, 不会更新质量代码和值。 $CNT_LIM = 0$ (默认设置) 时, 将禁用此功能。在延迟时间内将保留 $Quality\ Code = 16\#00$ 且 $QBAD = 1$ 和上一个值。

错误处理

不检查输入参数的似然性。如果在 $MODE$ (页 709) 输入的低位字中设置了无效模式, 则假定原始值是无效的。

启动特征

CNT_LIM 为 # 0 时启动接受值延迟。

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：
关于使用驱动程序块的注意事项 (页 329)

参见

SM 模块的 MODE 设置

4.2.2 CH_AI 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
CH_F_HL	输入值的过冲上限 (mA)	REAL	0	IO
CH_F_LL	输入值的下冲下限 (mA)	REAL	0	IO
CH_F_ON	1 = 激活限制监视	BOOL	0	IO
CNT_LIM	启动计数器限制	INT	0	IO
CNT_RES	启动计数器	INT	0	IO
DELTA_ON	超过上一个 delta 过程值	BOOL	0	IO
LAST_BAD	上一个 QBAD	BOOL	0	IO
LAST_ON	1= 上一个有效值: 启用注入	BOOL	0	IO
MODE	值状态和模式	DWORD	0	IO
OVHRANGE	过程值上限 (复制)	REAL	0	O
OVLRange	过程值下限 (复制)	REAL	0	O
QBAD	1 = 无效过程值	BOOL	0	O
QCHF_HL	1 = 过程值过冲	BOOL	0	O
QCHF_LL	1 = 过程值下冲	BOOL	0	O
QLAST	1= 上一个有效值: 注入激活	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O
QUALITY	过程值状态	BYTE	0	O
SIM_ON	1 = 激活仿真	BOOL	0	IO
SIM_V	仿真值	REAL	0	IO
SUBS_ON	1 = 启用替代	BOOL	0	IO
SUBS_V	替代值	REAL	0	IO
V	过程值	REAL	0	O

I/O (参数)	含义	数据类型	默认值	类型
VALUE	输入值	WORD	0	IO
VHRANGE	过程值上限	REAL	100	IO
VLRange	过程值下限	REAL	0	IO
V_DELTA	过程值增量 (V - V_LAST)	REAL	0	IO
V_LAST	上一个有效过程值	REAL	0	IO
V_LAST1	倒数第二个有效过程值	REAL	0	IO

4.3 CH_AO: 模拟值输出

4.3.1 CH_AO 的描述

对象名 (类型 + 编号)

FC 276

- CH_AO 块 I/O (页 344)

应用领域

块 CH_AO 用于处理 S7-300/400 SM 模拟输出模块的模拟输出值信号。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32) 和重启 OB 100。

注意
对于 PCS 7, 不应在设备块和输出驱动程序之间插入其它块。 如果违背此原则, 则在互连该块时, 应确保在设备块的输出与输出驱动程序之间, 所有组成输出信号的块都安装在同一 OB 中。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- MODE 输入与 MOD 块的相应 OMODE_xx 输出互连。
- CH_AO 块安装在 OB 100 中为其分配的 MOD 块的下游。

说明

如果不使用 CFC 功能“生成模块驱动程序”, 则必须确保 CH_AO 块安装在 OB 100 中为其分配的 MOD 块的下游。

功能和工作原理

块 CH_AO 循环处理所有通道特定信号功能。

块将过程值作为模拟原始值写入过程映像（分区）。MODE (页 709) 输入参数确定生成原始值所采用的格式。

如果输入参数 MODE 的高位字节为 0（值状态），则原始值仍然写入过程映像（分区），但质量代码为“无效值”。

质量代码可有以下状态：

状态	质量代码
有效值	16#80
上限值	16#56
下限值	16#55
仿真	16#60
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

寻址

在 HW Config 中为模拟输出通道生成的符号（符号表）必须互连到 VALUE 输出参数。

正常值

- 参数 ULRANGE 和 UHRANGE 依照 MODE (页 709) 将过程值 U 映射到模拟输出模块的原始 VALUE (量化步骤)。实例: 在 4 mA 到 20 mA (16#0203) 模式下, 如果 $U = \text{ULRANGE}$, 则输出 4 mA 的原始值; 如果 $U = \text{UHRANGE}$, 则输出 20 mA 的原始值。
- 该块切换参数 UHRANGE 和 ULRANGE 直到输出 OVHRANGE 和 OVL RANGE。例如, 可以互连这些输出与控制器 CTRL_PID 的调节变量限值 NM_LMNHR 和 NM_LMNL R。
- 可以用 PHYS_LIM 设置初始 VALUE 的限制值。默认设置 $\text{PHYS_LIM} = 0$ 将输出 VALUE 的值限制到模块的默认限值内。按照上述实例, 如果 $U > \text{UHRANGE}$, 则块计算 20 mA 的原始值; 如果 $U < \text{ULRANGE}$, 则计算 4 mA 的原始值。因此, QUALITY 输出应用质量代码 16#56 (上限值) 和 16#55 (下限值), 而非 16#80 (有效值)。
- 如果要输出的模拟值位于默认限值以外, 但在模块的物理限值内, 则设置 $\text{PHYS_LIM} = 1$ 。例如, 在上述实例中, 只有当 $\text{ULRANGE} = 0$ 且 $\text{UHRANGE} = 100$ 时, 通过指定 $U = 200$ (36 mA) 或 $U = -50$ (-4 mA) 来超出模块限值时, 才可限制输出值。这时输出值被限制在模块的数据表中指定的物理限值内, 同时输出相应的质量代码。
- 输出值 QCHF_HL 和 QCHF_LL 也提供有关是否已设置输出值限制的信息。

仿真

如果设置输入参数 $\text{SIM_ON} = \text{TRUE}$, 则输出 SIM_U 的值和质量代码 (QUALITY) = 16#60。并会复位 $\text{QBAD} = \text{TRUE}$ 。仿真拥有最高优先级。如果块处于仿真状态, 则 $\text{QSIM} = \text{TRUE}$ 。

I/O 故障

如果 MODE (页 709) 输入参数的高位字节设置为 0 (值状态), 则质量代码设置为 $\text{QUALITY} = 16\#00$ 。实际的原始值将始终写入过程映像 (分区)。

值限制

在将过程值输入过程映像（分区）之前，可以限制可导致错误 (QBAD = TRUE) 的非常低或非常高的过程值。

如果 LIMIT_ON 开关 = TRUE，则按如下所述限制过程值 (U)：

- 如果 $U > V_{HL}$ ，则为 V_{HL}
- 如果 $U < V_{LL}$ ，则为 LL_V 。

错误处理

不检查输入参数的似然性。如果 MODE (页 709) 输入的低位字中设置的运行模式无效，则会将数字化输出值设置为 0，并输出 QUALITY = 16#00。

启动特征

MOD 块在 OB 100 中的 OMODE_xx (页 719) 输出参数的字节 2 中设置 LSB。如果块检测到此代码，则会发送确认响应，并且：

如果未设置 START_ON，则计算过程值 U，并将结果写入过程映像。如果置位了 START_ON，则将与 START_U 过程值对应的原始值写入过程映像。

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.3.2 CH_AO 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
LIMIT_ON	1 = 启用过程值限制	BOOL	0	IO
LL_V	过程值, 如果 $U < V_{LL}$	REAL	0	IO
MODE	值状态和模式	DWORD	0	IO
OVHRANGE	过程值的输出上限	REAL	100	O
OVLRange	过程值的输出下限	REAL	0	O
PHYS_LIM	1 = 启用模块的物理限值	BOOL	0	IO
QBAD	1 = 输出值无效	BOOL	0	O
QCHF_HL	1 = 过程值过冲	BOOL	0	O
QCHF_LL	1 = 过程值下冲	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QUALITY	输出值的值状态	BYTE	0	O
SIM_ON	1 = 激活仿真	BOOL	0	IO
SIM_U	仿真值	REAL	0	IO
START_ON	1 = 启动时替代	BOOL	0	IO
START_U	启动时的替代值	REAL	0	IO
U	过程值	REAL	0	IO
UHRANGE	过程值上限	REAL	0	IO
ULRANGE	过程值下限	REAL	0	IO
V_HL	上限	REAL	0	IO
V_LL	下限	REAL	0	IO
VALUE	PI 输出值	WORD	0	O

4.4 CH_CNT: 控制和读取 FM 350 模块

4.4.1 CH_CNT 的描述

对象名 (类型 + 编号)

FB 127

- CH_CNT 块 I/O (页 350)

应用领域

块 CH_CNT 用于控制和读取 FM 350-1 或 FM 350-2 模块的计数或测量值。

调用 OB

接收和发送数据的循环 OB (建议 100 毫秒)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 将组态 LADDR 和 CHANNEL 输入。
- MODE 输入互连到 FM_CNT 块的 OMODEx 输出。
- FM_DATA 结构与 FM_CNT 块的同名结构互连。

功能和工作原理

FM 350-1 和 FM 350-2 的通讯接口存在以下差异:

- 块仅通过 FM 350-1 的过程映像通讯。数据读写不间断。
- FM 350-2 的过程映像中包含控制和状态信息以及所选的计数和测量值。其它计数和测量值可通过数据记录读取。

在 HW Config 中, 可定义计数和测量值在过程映像中的保存方法 (User_Type1 和 User_Type2)。参数 LOAD_VAL 和 CMP_VALx 使用数据记录从 FM_CNT 块加载到 FM 350-2。在 FM_CNT 块的后续周期中首先触发参数写入。

下文中的“FM 350”是指 FM 350-1 和 FM 350-2 模块。

如果正在使用 FM 350-2 模块, 该块将通过数据记录将 LOAD_VALx (立即加载计数值)、PREP_VALx (准备加载计数值) 或 CMP_VALx (比较值) 参数写入此模块 (x = 通道号)。如果在块中已设置参数 LOAD_DIR = TRUE, 则该块将写入 LOAD_VALx。当设置 LOAD_PRE = TRUE 时, 该块写入 PREP_VALx。每次更改后, 都会写入 CMP_VALx 参数。

MODE 输入指示计数和/或测量值在过程映像中以哪种格式提供。如果输入参数 MODE 的高位字 = 16#40xxxx (值状态 = 较高级别错误、QMOD_ERR = TRUE), 则将计数或测量值视为无效。

说明

自动确认状态 QCOMP1 (比较器 1)、QCOMP2 (比较器 2)、QZERO (零交叉)、QOFLW (上溢) 和 QUFLW (下溢)。这些状态至少激活一个周期。

测量值通过 FM 350 以数值的形式输出。更多相关信息, 可参考该模块的手册。

质量代码

质量代码描述 CH_CNT 块 MODE 输入的信号状态。

程序将生成结果值的质量代码，可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
上一个有效值	16#44
替代值	16#48
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围内的值。

质量代码保存在 MODE 参数的字节 2 和 3 中。

寻址

1. 根据 FM 350 模块的基地址，在图标表中为所需的计数或测量值创建图标 请注意以下几点：

- FM 350-1: 计数或测量值始终位于过程映像中：
 - 选择模块的“ED 基地址”（例如，ED512）作为地址。
- FM 350-2: 所需通道的计数或测量值位于过程映像中：
 - 在“HW Config FM 350-2 组态计数器”中，可指定计数或测量值在过程映像中的存储位置。根据 User_Type1 或 User_Type2 的组态，必须对 WORD 选择 EW 或者对 DWORD 选择 ED。将按照下表计算地址：

计数或测量值被定义为:	测量或计数值位于 User_Type1 中:	测量或计数值位于 User_Type2 中:
DWORD 或 LOW WORD	FM 350-2 基地址 + 8 字节	FM 350-2 基地址 + 12 字节
HIGH WORD	FM 350-2 基地址 + 10 字节	FM 350-2 基地址 + 14 字节
实例:		
通道 2 所需的计数值在 User_type2 的高位字中。基址为 512 时，该地址计算为：地址 = EW 526。		

- FM 350-2: 所需通道的计数或测量值不在过程映像中:
 - 选为地址: 互连的输入字“模块的基地址 + 通道号”(例如, 基地址 = 512, 通道号 = 5; EW517)。
 - 通过“互连到操作数...”使 CFC 图表中的输入 LATCH 与先前创建的图标相连接。

如果输入 USE_CNT 或 USE_MSrv 设置为 TRUE, 将从该模块中以数据记录的形式循环读出不在 FM 350-2 的过程映像中的计数和测量值。如果用户程序中不需要通道的计数或测量值, 出于性能方面的考虑, 应将两个输入都设置为 FALSE。如此可防止计数值或测量值不在过程映像中时, 通过数据记录来读取这些值。

说明

即使未设置输入 USE_CNT 或 USE_MSrv, 只要在相关 FM350-2 的 CH_CNT (其它通道) 的不同实例上设置了输入 USE_CNT 或 USE_MSrv, 也可以通过数据记录来执行读出操作。

仿真

如果设置输入参数 SIM_ON = TRUE, 则输出输入参数 SIM_V 的值和质量代码 QUALITY = 16#60, 并设置 QBAD = FALSE。仿真拥有最高优先级。如果块处于仿真状态, 则 QSIM = TRUE。

替代值

如果输入参数 SUBS_ON = TRUE, 且计数或测量值无效, 则会将输入参数 SUBS_V 的值作为替代值输出。质量代码将设置为 QUALITY = 16#48 且 QBAD = 1。

保持上一个值

如果输入参数 LAST_ON = TRUE, 且计数或测量值无效, 则会输出上一个有效输出值。质量代码设置将为 QUALITY = 16#44 且 QBAD = 1。

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE, 或均为 TRUE, 并且存在一个无过程值, 则会输出此无效值并将 QBAD 设置为 1。

冗余

在 H 系统中, 更高级别的 MOD_D1 块评估 DP 主站系统的冗余。

错误处理

不检查输入参数的似然性。

启动特征

启动和重新启动时, FM_CNT 块将参数 CMP_VAL0 和 CMP_VAL1 (仅限 FM 350-1) (比较值; 定量给料时还包括 CMP_VAL1、2、3 (FM 350-2)) 发送到 FM 350。

过载行为

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息, 请参考以下部分:
关于使用驱动程序块的注意事项 (页 329)

4.4.2 CH_CNT 的 I/O

CFC 中块显示的工厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示 I/O 可见; 正常字体的 I/O 名称则表示 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
ACT_CNTV	当前负载或 LATCH 值/当前测量值	DINT	0	O
ACT_MSrv	当前测量值	DINT	0	O
CHANNEL	通道 FM 350	INT	0	I
CMP_V0	新比较值 1/上限	DINT	0	I
CMP_V1	新比较值 2/更新时间	DINT	0	I
CMP_V2	新比较值 3 (定量给料模式)	DINT	0	I
CMP_V3	新比较值 4 (定量给料模式)	DINT	0	I
CTRL_DO0	1 = 启用数字输出 DO	BOOL	0	I
CTRL_DO1	1 = 启用数字输出 DO1 (仅限 FM 350-1 或 FM 350-2, 定量给料模式)	BOOL	0	I
CTRL_DO2	1 = 启用数字输出 DO2 (仅限 FM 350-2, 定量给料模式)	BOOL	0	I
CTRL_DO3	1 = 启用数字输出 DO3 (仅限 FM 350-2, 定量给料模式)	BOOL	0	I
ENSET_DN	1 = 启用倒退方向设置	BOOL	1	I
ENSET_UP	1 = 启用前进方向设置	BOOL	1	I
FM_DATA	结构 FM 350 数据	STRUCT		IO
GATE_STP	1 = 常规 GATE 停止	BOOL	0	I
LADDR	逻辑地址 FM 350	INT	0	I
LAST_ON	1 = 上一个有效值: 启用注入	BOOL	0	IO
LATCH	当前计数值	ANY	0	I
LOAD_DIR	1 = 立即加载	BOOL	0	IO
LOAD_PRE	1 = 准备加载	BOOL	0	IO

I/O (参数)	含义	数据类型	默认值	类型
LOAD_VAL	新负载值/下限	DINT	0	I
MODE	通道模式	DWORD	0	I
QBAD	1 = 无效值	BOOL	0	O
QCMP1	1 = 比较值 1	BOOL	0	O
QCMP2	1 = 比较值 2	BOOL	0	O
QCMP3	1 = 比较值 3	BOOL	0	O
QCMP4	1 = 比较值 4	BOOL	0	O
QCOMP1	1 = 比较器 1 的保存状态 (对应于 FM 350-2 的 STS_CMP)	BOOL	0	O
QCOMP2	1 = 比较器 2 的保存状态 (仅限 FM 350-1 或 FM 350-2), 定量给料模式	BOOL	0	O
QCOMP3	1 = 比较器 2 的保存状态 (仅限 FM 350-2), 定量给料模式	BOOL	0	O
QCOMP4	1 = 比较器 2 的保存状态 (仅限 FM 350-2), 定量给料模式	BOOL	0	O
QDIR	1 = 状态计数器计数方向	BOOL	0	O
QGATE	1 = 状态内部 GATE	BOOL	0	O
QLAST	1 = 上一个有效值: 注入激活	BOOL	0	O
QLATCH	1 = 新 LATCH 值 (仅限时钟同步模式)	BOOL	0	O
QMOD_ER R	1 = 较高级别错误	BOOL	0	O
QOFLW	1 = 状态上溢	BOOL	0	O
QOP_ERR	1 = 操作员错误	BOOL	0	O
QRUN	1 = 状态计数器正在工作	BOOL	0	O
QSET	1 = 状态数字输入 DI 已设置	BOOL	0	O
QSIM	1 = 仿真值	BOOL	0	O
QSTA	1 = 数字输入 DI 启动	BOOL	0	O

4.4 CH_CNT: 控制和读取 FM 350 模块

I/O (参数)	含义	数据类型	默认值	类型
QSTP	1 = 状态数字输入 DI 停止	BOOL	0	O
QSUBS	1 = 错误替代值激活	BOOL	0	O
QSW_G	1 = 状态 SW GATE	BOOL	0	O
QSYNC	1 = 状态计数器已同步	BOOL	0	O
QUALITY	过程值状态	BYTE	0	O
QUFLW	1 = 状态下溢	BOOL	0	O
QZERO	1 = 过零点状态	BOOL	0	O
R_OP_ERR	1 = 复位操作员错误	BOOL	0	IO
RES_SYNC	1 = 复位同步	BOOL	0	IO
SET_DO0	1 = 打开 DO0	BOOL	0	I
SET_DO1	1 = 打开 DO1 (仅限 FM 350-1 或 FM 350-2, 定量给料模式)	BOOL	0	I
SET_DO2	1 = 打开 DO2 (仅限 FM 350-2, 定量给料模式)	BOOL	0	I
SET_DO3	1 = 打开 DO3 (仅限 FM 350-2, 定量给料模式)	BOOL	0	I
SIM_CNT	仿真计数值	DINT	0	I
SIM_MS RV	仿真测量值	DINT	0	I
SIM_ON	1 = 仿真激活	BOOL	0	I
SUBS_CNT	计数替代值	DINT	0	I
SUBS_MS RV	测量替代值	DINT	0	I
SUBS_ON	1 = 替代值激活	BOOL	0	I
SW_GATE	1 = 启用 SW GATE	BOOL	0	I
USE_CNT	1 = 计数值已使用	BOOL	1	I
USE_MS RV	1 = 测量值已使用	BOOL	1	I

4.5 CH_CNT1: 控制和读取 ET 200iSP 的 8-DI-NAMUR 模块

4.5.1 CH_CNT1 的描述

对象名 (类型 + 编号)

FB 59

- CH_CNT1 块 I/O (页 360)

应用领域

块 CH_CNT1 用于控制和读取 ET 200iSP 的 8-DI NAMUR 模块的计数或频率值。块支持模块的下列组态:

- 2 个计数器或 1 个计数器级联
- 2 个频率测量

调用 OB

接收和发送数据的循环 OB (建议 100 毫秒)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 组态输入 LADDR、LADDR1 和 CHANNEL。
- MODE 输入将与 MOD_D1 块的 OMODEx 输出互连。

4.5 CH_CNT1: 控制和读取 ET 200iSP 的 8-DI-NAMUR 模块

功能和工作原理

根据 HW Config 中模块的模式设置，模块的用户数据存储的过程映像中。块 CH_CNT1 区分以下模式：

MODE (低位字)	模式	HW Config 设置: “组态”	HW Config 设置: “通道 (0 到 1) 模式”
1	计数器 (16 位)，没有通过数字信号实现的控制功能	(通道 0 到 1): COUNT (通道 2 到 7): DI	周期或常规计数功能 (向上或向下计数器)
2	计数器 (32 位)，没有通过数字信号实现的控制功能	(通道 0 到 1): COUNT (通道 2 到 7): DI	级联功能 (仅限通道 0) (向下计数器)
3	计数器 (16 位)，具有通过数字信号实现的控制功能	(通道 0 到 1): COUNT (通道 2 到 7): CONTROL	周期或常规计数功能 (向上或向下计数器)
4	计数器 (32 位)，具有通过数字信号实现的控制功能	(通道 0 到 1): COUNT (通道 2 到 7): CONTROL	级联功能 (仅限通道 0) (向下计数器)
5	频率 (16 位)	(通道 0 到 1): TRACE (通道 2 到 7): DI	-

驱动程序生成器在模块相应通道上 MOD_D1 块的 MODE 输入处设置 HW Config 中组态的模块模式。MODE 输入指示在过程映像中计数值或频率值以哪种格式提供。如果输入参数 MODE 的高位字 = 16#40xxxx (值状态 = 较高级别错误、QMOD_ERR = TRUE)，则计数或频率值视为无效。

根据模式不同，过程映像中可能存在两个独立的计数器 (16 位) 或一个计数器 (32 位)。CHANNEL 输入指定块所负责的模块计数器。

计数器功能可通过某些信号进行控制，而模块的数字输入或过程映像的用户数据均可影响这些信号。

说明

请注意，数字输入信号与模块中 PIO 的相当信号是“或”逻辑关系。

可使用以下信号:

块输入	模块	含义
-	C1	计数器脉冲计数器 1
-	C2	计数器脉冲计数器 2
GATE_STP (CHANNEL =0)	GATE1	<p>可使用激活的 GATE 信号中断激活的计数操作。</p> <p>无论待解决的计数脉冲如何, 信号 GATE = "1" 都将停止计数操作。同时, 如果分配的输出原来处于激活状态, 则禁用该输出。此状态一直保持到 GATE 信号设为 "0"。输出恢复到先前的状态, 而计数操作则继续。</p> <p>GATE 信号从属于 RSO 和 RSC 信号, 换句话说, 无论 GATE 信号是否激活, RSO 和 RSC 信号都会产生上述影响。</p>
GATE_STP (CHANNEL =1)	GATE2	请参见“GATE 1”的描述。
RES_CNT (CHANNEL =0)	RSC1	<p>RSC 信号的上升沿按以下方式设置已分配通道的计数:</p> <ul style="list-style-type: none"> 向上计数 (正常计数器功能) 时, 复位回零 向下计数 (定期计数器功能和级联功能) 时, 设置为定义的设定值 <p>向下计数 (定期计数器功能和级联功能) 时, 还复位设置的任意输出。</p>
RES_CNT (CHANNEL =1)	RSC2	请参见“RSC1”的描述。
RES_DO (CHANNEL =0)	RSO1	<p>在 RSO 信号的上升沿, 可复位分配的输出。</p> <p>设置 RSO 不影响计数。</p>
RES_DO (CHANNEL =1)	RSO2	请参见“RSO2”的描述。

输入/输出参数 RES_CNT 和 RES_DO 始终复位为零。复位后, 最早在下一个周期才能更新复位, 但在一个周期内 (上升沿) 无法进行。

4.5 CH_CNT1: 控制和读取 ET 200iSP 的 8-DI-NAMUR 模块

计数值或频率值及其状态存储在过程映像中（如下所述），并在以下块输出中指示：

字节	位	输入信号	块输出	含义
0, 1	0-15	过程值计数器 1	ACT_CNTV	16 位计数器 1 或 32 位计数器（字节 0 到 3）或频率值 1
2, 3	0-15	过程值计数器 2		16 位计数器 2（仅与 16 位计数器 1 结合使用）或频率值 2
4	0	A1	QZERO	过零点计数器 1
	1	A2		过零点计数器 2
	2	GATE 1	QGATE	状态门 1
	3	GATE 2		状态门 2
	4	RSC1	QRES_CNT	状态复位计数器 1
	5	RSC2		状态复位计数器 2
	6	RSO1	QRES_DO	状态复位输出计数器 1
	7	RSO2		状态复位输出计数器 2

LOAD_VAL 参数始终写入过程映像。根据使用 HW Config 设置的模式，可能是 16 位或 32 位设定值（向下计数器）或计数限值（向上计数器）。

根据模式设置，仅将 LOAD_VAL 的以下整数值传送到模块：

- 16 位计数器：0 到 65,535
- 32 位计数器：0 到 2,147,483,647

如果 LOAD_VAL 的值超出这些限值，则在模块中保持 LOAD_VAL 的上一个有效值，并设置 QOP_ERR = TRUE。

质量代码

程序将生成结果值的质量代码，可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
上一个有效值	16#44
替代值	16#48
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

注：

在 HW Config 中，可以只分配模块的数字信号 DI2 到 DI7（HW Config 通道 2 到 7 = DI），而不分配控制信号 GATE 1 到 RSO2。在 DI NAMUR 模块的此组态中，输出 QGATE、QZERO、QRES_CNT 和 QRES_DO 的状态取决于块的输入。

使用模块的数字控制信号 GATE 1 到 RSO2 时（HW Config 通道 2 到 7 = CONTROL），视信号状态而定，可能会与块数字信号发生冲突。在这种情况下，数字信号失效。如果要对块进行控制，则不能在 HW Config 中分配控制信号。

实例：

模块	块	具有以下效果	
GATE 1 = 1	GATE_STP = 0	→	GATE 打开
GATE 1 = 0	GATE_STP = 1	→	GATE 打开
GATE 1 = 0	GATE_STP = 0	→	GATE 关闭

寻址

计数值或频率值的图标（来自图标表）必须连接到 VALUE 输入参数。

仿真

如果设置输入参数 SIM_ON = TRUE, 则输出输入参数 SIM_V 的值和质量代码 QUALITY = 16#60, 并设置 QBAD = FALSE。仿真拥有最高优先级。如果该块处于仿真状态, 则会设置 QSIM = TRUE。

替代值

如果输入参数 SUBS_ON = TRUE, 且计数或测量值无效, 则会将输入参数 SUBS_V 的值作为替代值输出。质量代码将设置为 QUALITY = 16#48 且 QBAD = 1。

保持上一个值

如果输入参数 LAST_ON = TRUE, 且计数或测量值无效, 则会输出上一个有效输出值。质量代码设置将为 QUALITY = 16#44 且 QBAD = 1。

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE, 或均为 TRUE, 并且存在一个无效过程值, 则会输出此无效值并将 QBAD 设置为 1。

冗余

在 H 系统中, 更高级别的 MOD_D1 块评估 DP 主站系统的冗余。

错误处理

不检查输入参数的似然性。例外: 系统会检查 LOAD_VAL 的输入值是否有效。更多相关信息, 可参考“功能和工作原理”。

启动特征

不可用

过载行为

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.5.2 CH_CNT1 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
ACT_CNTV	当前计数值/频率值	DINT	0	O
CHANNEL	通道 8 DI NAMUR	INT	0	I
GATE_STP	1 = GATE 打开 (停止计数)	BOOL	0	I
LADDR	逻辑地址 (输入)	INT	0	I
LADDR1	逻辑地址 (输出)	INT	0	I
LAST_ON	1 = 上一个有效值: 启用注入	BOOL	0	IO
LOAD_VAL	计数器负载值	DINT	0	I
MODE	通道模式	DWORD	0	I
QBAD	1 = 无效值	BOOL	0	O
QGATE	1 = GATE 打开	BOOL	0	O
QLAST	1 = 上一个有效值: 注入激活	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QOP_ERR	1 = 操作员错误	BOOL	0	O
QSIM	1 = 仿真值	BOOL	0	O
QRES_CNT	1 = 计数值复位	BOOL	0	O
QRES_DO	1 = 数字输出复位	BOOL	0	O
QSUBS	1 = 错误替代值激活	BOOL	0	O
QUALITY	过程值状态	BYTE	0	O
QZERO	1 = 过零点状态	BOOL	0	O
RES_CNT	1 = 复位计数值	BOOL	0	IO
RES_DO	1 = 复位数字输出	BOOL	0	IO
SIM_CNT	仿真值	DINT	0	I
SIM_ON	1 = 仿真激活	BOOL	0	I
SUBS_CNT	替代值	DINT	0	I

4.5 CH_CNT1: 控制和读取 ET 200iSP 的 8-DI-NAMUR 模块

I/O (参数)	含义	数据类型	默认值	类型
SUBS_ON	1 = 替代值激活	BOOL	0	I
VALUE	图标计数值	ANY	0	I

4.6 CH_CNT2C: 控制和读取计数模式下的 1 COUNT 24V/100kHz 模块

4.6.1 CH_CNT2C 描述

对象名 (类型 + 编号)

FB 242

- CH_CNT2C 的 I/O (页 366)

应用领域

该块用于控制和读取计数模式下的“1 COUNT 24V/100kHz”模块的计数值和锁存值。

调用 OB

OB 100 以及接收和发送数据的循环 OB (建议 100 毫秒)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动出现以下情况:

- 给 LADDR 输入分配参数值以及
- MODE 输入与 MOD_D1 块的 OMODE_00 输出互连。

4.6 CH_CNT2C: 控制和读取计数模式下的 1 COUNT 24V/100kHz 模块

功能和工作原理

该块通过过程映像通讯。连续读写数据。

在输入/输出参数 L_PREPAR (准备装载计数器) 或 L_DIRECT (直接装载计数器) 上检测到正沿时将 LOAD_VAL 参数传送给模块。CMP_VAL1 (比较值 1) 和 CMP_VAL2 (比较值 2) 参数在有变化时和启动期间将传送到模块。

根据 HW Config 中模块的模式设置, 模块的用户数据存储过程映像中。该块提供以下不同模式:

MODE (LowWord)	模式	描述
1	连续计数 (Continuous counting)	在该模式下, 1Count24V/100kHz 从装载值开始不停地计数。
2	一次性计数 (One-time counting)	在该模式下, 1Count24V/100kHz 仅计数一次; 这取决于所设置的主计数方向
3	周期性计数 (Periodic counting)	在该模式下, 1Count24V/100kHz 进行周期性计数; 这取决于所设置的主计数方向

MODE 输入指示在过程映像中以何种形式提供计数值或锁存值。如果 MODE 输入参数的高位字节 = 16#40 (值状态 = 较高级别错误、QMOD_ERR = TRUE), 则将计数值和锁存值视为无效。

将为每个结果值 ACT_LATCH 和 ACT_CNTV 生成一个质量代码, 质量代码可有以下状态:

质量代码

状态	质量代码
有效值	16#80
仿真	16#60
上一个有效值	16#44
替代值	16#48
无效值	16#00

注:

该块自动确认状态 QSYNC (同步)、QCOMP1 (比较器 1)、QCOMP2 (比较器 2)、QOFLW (上溢)、QUFLW (下溢) 和 QZERO (过零点)。这些状态至少激活一个周期。

寻址

连接计数值符号 (来自符号表) 与 LATCH 输入参数。

在符号表中 (符号列) 输入符号并在地址列的行中添加模块的 ID 基地址 (例如, ID512)。

仿真

如果设置输入参数 SIM_ON = TRUE, 则输出输入参数 SIM_CNTV 和 SIM_LATCH 的值和质量代码 QUALITY = 16#60, 并设置 QBAD = FALSE。仿真拥有最高优先级。如果该块处于仿真状态, 则会设置 QSIM = TRUE。

替代值

输入参数 SUBS_ON = TRUE 且计数值无效时, 将输出输入参数 SUBS_CNTV 和 SUBS_LATCH 的值作为替代值。质量代码将设置为 QUALITY = 16#48 且 QBAD = 1。

保持上一个值

如果输入参数 LAST_ON = TRUE 和 SUBS_ON = FALSE, 且计数值或测量值无效, 则会输出上一个有效输出值。质量代码将设置为 QUALITY = 16#44 且 QBAD = 1。

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE, 或均为 TRUE, 并且存在一个无效过程值, 则会输出此无效值并将 QBAD 设置为 1。

错误处理

对输入参数不执行似然性检查。模块错误 (QERR_24V = TRUE、QERR_DO1 = TRUE)，需要通过 EXTF_ACK 参数确认。

当 QERR_24V = 1 时，计数值和锁存值被视为无效且设置 QBAD = 1，对于 QERR_DO1，质量代码仍保持为 QUALITY 有效且 QBAD = 0。

通过正确的参数分配确认参数分配错误 QERR_PARA，并且通过后续正确的操作员输入删除 QERR_LOAD 装载功能错误。

启动特征

启动期间，CMP_VAL1（比较值 1）和 CMP_VAL2（比较值 2）参数将传送到模块。

过载行为

不可用。

时间响应

不可用。

消息响应

不可用。

操作和监视

该块没有面板。

4.6.2 CH_CNT2C 的 I/O

CFC 中块显示的出厂设置在 I/O 列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关缩写的说明和含义, 可参考“关于块描述的常规信息 (页 17)”

I/O (参数)	含义	数据类型	默认值	类型
ACT_CNTV	当前计数值	DINT	0	O
ACT_LATCH	当前计数值, 或者在数字输入端使用了锁存功能时存储的计数值	DINT	0	O
CMP_VAL1	比较值 1	DINT	0	I
CMP_VAL2	比较值 2	DINT	0	I
CTRL_DO1	1 = 启用 DO1	BOOL	0	I
CTRL_DO2	1 = 启用 DO2	BOOL	0	I
CTRL_SYN	1 = 启用同步	BOOL	0	I
EXTF_ACK	1 = 错误确认	BOOL	0	I
L_DIRECT	1 = 直接装载计数器	BOOL	0	I
L_PREPAR	1 = 准备装载计数器	BOOL	0	I
LADDR	模块的逻辑地址	INT	0	I
LAST_ON	1 = 上一个有效值: 启用灌注	BOOL	0	I
LATCH	互连的锁存值	DWORD	0	I
LOAD_VAL	直接/准备的装载值	DINT	0	I
MODE	模式	DWORD	0	I
QBAD	1 = 无效过程值	BOOL	0	O
QCMP1	1 = 比较器 1 的状态	BOOL	0	O
QCMP2	1 = 比较器 2 的状态	BOOL	0	O
QCNT_DN	1 = 减计数状态	BOOL	0	O
QCNT_UP	1 = 加计数状态	BOOL	0	O
QDI	1 = DI 状态	BOOL	0	O
QDO1	1 = DO1 状态	BOOL	0	O
QDO2	1 = DO2 状态	BOOL	0	O

4.6 CH_CNT2C: 控制和读取计数模式下的 1 COUNT 24V/100kHz 模块

I/O (参数)	含义	数据类型	默认值	类型
QERR_24V	1 = 传感器电源短路	BOOL	0	O
QERR_DO1	1= 短路/断路/过热	BOOL	0	O
QERR_LOAD	1 = 装载功能有错误	BOOL	0	O
QERR_PARA	1 = 参数分配错误	BOOL	0	O
QGATE	1 = 内部门的状态	BOOL	0	O
QLAST	1 = 上一个有效值: 灌注激活	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QOFLW	1 = 计数值上限	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代值激活	BOOL	0	O
QSYNC	1 = 计数器已同步状态	BOOL	0	O
QUALITY	过程值的值状态	BYTE	0	O
QUFLW	1 = 计数值下限	BOOL	0	O
QZERO	1 = 过零点状态	BOOL	0	O
SET_DO1	1 = 打开 DO1	BOOL	0	I
SET_DO2	1 = 打开 DO2	BOOL	0	I
SIM_CNTV	仿真计数值	DINT	0	I
SIM_LATCH	仿真锁存值	DINT	0	I
SIM_ON	1 = 启用仿真	BOOL	0	I
SUBS_CNTV	计数替代值	DINT	0	I
SUBS_LATCH	锁存替代值	DINT	0	I
SUBS_ON	1 = 启用替代值	BOOL	0	I
SW_GATE	1= 启用软件门	BOOL	0	I

4.7 CH_CNT2M: 控制和读取测量模式下的 1 COUNT 24V/100kHz 模块

4.7.1 CH_CNT2M 描述

对象名 (类型 + 编号)

FB 243

- CH_CNT2M 的 I/O (页 372)

应用领域

该块用于控制和读取测量模式下的“1 COUNT 24V/100kHz”模块的计数值和测量值。

调用 OB

OB 100 以及接收和发送数据的循环 OB (建议 100 毫秒)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动出现以下情况:

- 给 LADDR 输入分配参数值以及
- MODE 输入与 MOD_D1 块的 OMODE_00 输出互连。

4.7 CH_CNT2M: 控制和读取测量模式下的 1 COUNT 24V/100kHz 模块

功能和工作原理

该块通过过程映像通讯。连续读写数据。

UFLW（下限）和 OFLW（上限）参数在有变化时和启动期间将传送到模块。

根据 HW Config 中模块的模式设置，模块的用户数据存储过程映像中。该块提供以下不同模式：

MODE (LowWord)	模式	描述
4	频率测量模式 (Frequency measurement)	1Count24V/100kHz 确定输入端脉冲序列的频率。
5	速度测量模式 (Speed measurement)	1Count24V/100kHz 确定连接到输入端的设备的转速。
6	周期测量模式 (Period duration measurement)	1Count24V/100kHz 确定输入端脉冲序列的脉宽。

MODE 输入指示在过程映像中以何种形式提供计数值或测量值。如果 MODE 输入参数的高位字节 = 16#40（值状态 = 较高级别错误、QMOD_ERR = TRUE），则将计数值和测量值视为无效。

质量代码

将为每个结果值 ACT_MSRV 和 ACT_CNTV 生成一个质量代码，质量代码可有以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
上一个有效值	16#44
替代值	16#48
无效值	16#00

4.7 CH_CNT2M: 控制和读取测量模式下的 1 COUNT 24V/100kHz 模块

注意事项

该块自动确认状态 QCMP1（测量完成）、QOFLW（上溢）和 QUFLW（下溢）。这些状态至少激活一个周期。

寻址

连接计数值符号（来自符号表）与 MSRV 输入参数。

在符号表中（符号列）输入符号并在地址列的行中添加模块的 ID 基地址（例如，ID512）。

仿真

如果设置输入参数 SIM_ON = TRUE，则输出输入参数 SIM_CNTV 和 SIM_MS RV 的值和质量代码 QUALITY = 16#60，并设置 QBAD = FALSE。仿真拥有最高优先级。如果该块处于仿真状态，则会设置 QSIM = TRUE。

替代值

输入参数 SUBS_ON = TRUE 且计数值无效时，将输出输入参数 SUBS_CNTV 和 SUBS_MS RV 的值作为替代值。质量代码将设置为 QUALITY = 16#48 且 QBAD = 1。

保持上一个值

如果输入参数 LAST_ON = TRUE 和 SUBS_ON = FALSE，且计数值或测量值无效，则会输出上一个有效输出值。质量代码将设置为 QUALITY = 16#44 且 QBAD = 1。

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE，或均为 TRUE，并且存在一个无效过程值，则会输出此无效值并将 QBAD 设置为 1。

错误处理

对输入参数不执行似然性检查。模块错误 (QERR_24V = TRUE、QERR_DO1 = TRUE)，需要通过 EXTF_ACK 参数确认。

当 QERR_24V = 1 时，计数值和测量值被视为无效且设置 QBAD = 1，对于 QERR_DO1，质量代码仍保持为 QUALITY 有效且 QBAD = 0。

通过正确的参数分配确认参数分配错误 QERR_PARA，并且通过后续正确的操作员输入删除 QERR_LOAD 装载功能错误。

启动特征

启动期间，UFLW（下限）和 OFLW（上限）参数将传送到模块。

过载行为

不可用。

时间响应

不可用。

消息响应

不可用。

操作和监视

该块没有面板。

4.7 CH_CNT2M: 控制和读取测量模式下的 1 COUNT 24V/100kHz 模块

4.7.2 CH_CNT2M 的 I/O

CFC 中块显示的出厂设置在 I/O 列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关缩写的说明和含义, 可参考“关于块描述的常规信息 (页 17)”

I/O (参数)	含义	数据类型	默认值	类型
ACT_CNTV	当前计数值	DINT	0	O
ACT_MSrv	当前测量值	DINT	0	O
CTRL_DO1	1 = 启用 DO1	BOOL	0	I
EXTF_ACK	1 = 错误确认	BOOL	0	I
LADDR	模块的逻辑地址	INT	0	I
LAST_ON	1 = 上一个有效值: 启用灌注	BOOL	0	I
MODE	模式	DWORD	0	I
MSRV	互连的测量值	DWORD	0	I
OFLW	上限	DINT	0	I
QBAD	1 = 无效过程值	BOOL	0	O
QCMP1	1 = 测量已完成	BOOL	0	O
QCNT_DN	1 = 减计数状态	BOOL	0	O
QCNT_UP	1 = 加计数状态	BOOL	0	O
QDI	1 = DI 状态	BOOL	0	O
QDO1	1 = DO1 状态	BOOL	0	O
QERR_24V	1 = 传感器电源短路	BOOL	0	O
QERR_DO1	1 = 短路/断路/过热	BOOL	0	O
QERR_LOAD	1 = 装载功能有错误	BOOL	0	O
QERR_PARA	1 = 参数分配错误	BOOL	0	O
QGATE	1 = 内部门的状态	BOOL	0	O
QLAST	1 = 上一个有效值: 灌注激活	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QOFLW	1 = 测量范围上限	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O

4.7 CH_CNT2M: 控制和读取测量模式下的 1 COUNT 24V/100kHz 模块

I/O (参数)	含义	数据类型	默认值	类型
QSUBS	1 = 替代值激活	BOOL	0	O
QUALITY	过程值的值状态	BYTE	0	O
QUFLW	计数下限	BOOL	0	O
SET_DO1	1 = 打开 DO1	BOOL	0	I
SIM_CNTV	仿真计数值	DINT	0	I
SIM_MS RV	仿真测量值	DINT	0	I
SIM_ON	1 = 启用仿真	BOOL	0	I
SUBS_CNTV	计数替代值	DINT	0	I
SUBS_MS RV	测量替代值	DINT	0	I
SUBS_ON	1 = 启用替代值	BOOL	0	I
SW_GATE	1= 启用软件门	BOOL	0	I
UFLW	下限	DINT	0	I

4.8 CH_DI: 数字值输入

4.8.1 CH_DI 的描述

对象名 (类型 + 编号)

FC277

- CH_DI 块 I/O (页 377)

应用领域

CH_DI 块用于 S7-300/400 SM 数字输入模块的数字输入值信号处理。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, MODE 输入会自动与 MOD 块的对应输出 OMODE_xx 互连。

功能和工作原理

块 CH_DI 循环处理所有通道特定信号功能。

块从过程映像 (分区) 读取 BOOL 数据类型的数字值。如果 MODE (页 709) 输入参数的高位字节 = 16#40 (值状态 = 较高级别错误、QMOD_ERR = TRUE), 则将该数字值视为无效。如果输入参数 PQC = TRUE, 则从过程映像 (分区) 读取数字值的值状态。

质量代码

程序将生成结果值的质量代码，可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
上一个有效值	16#44
替代值	16#48
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成的。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

寻址

在 HW Config 中为数字输入通道生成的符号（符号表）必须互连到 VALUE 输入。如果过程映像（分区）中还包含数字输入通道的值状态，则将相应的符号互连到输入 VALUE_QC 并设置输入 PQC = TRUE。

正常值

过程映像（分区）的数字值和质量代码 QUALITY = 16#80 都应用于输出 Q。

仿真

如果输入参数 SIM_ON = TRUE，则输出输入参数 SIM_I 的值到输出参数 Q 且质量代码 QUALITY = 16#60。复位 QBAD = TRUE。仿真拥有最高优先级。如果该块处于仿真状态，则会设置 QSIM = TRUE。

说明

请记住，仿真值总是在仿真模式下输出，与任一参数 LAST_ON（替代值）或 SUBS_ON（上一个有效值）都无关。

4.8 CH_DI: 数字值输入

替代值

输入参数 SUBS_ON = TRUE 和过程映像（分区）的数字值无效时，此功能将信号 QBAD = 1 和输入参数 SUBS_I 的值以及质量代码 QUALITY = 16#48 输出到输出参数 Q。

保持上一个值

如果输入参数 LAST_ON = TRUE，且计数值或测量值无效，则会输出上一个有效输出值。质量代码将设置为 QUALITY = 16#44 且 QBAD = 1。

上一个有效输出值 = Q_LAST。

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE，或均为 TRUE，并且存在一个无过程值，则会输出此无效值并将 QBAD 设置为 1。

错误处理

不检查输入参数的似然性。

启动特征

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.8.2 CH_DI 的 I/O

CFC 中块显示的出厂设置在“**I/O**”列中标识：

加粗字体的 I/O 名称表示该 I/O 可见；正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息，可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
LAST_ON	1 = 上一个有效值： 启用注入	BOOL	0	IO
MODE	值状态和模式	DWORD	0	IO
PQC	1 = 在过程映像中使用值状态	BOOL	0	IO
Q	过程值	BOOL	0	O
QBAD	1 = 无效过程值	BOOL	0	O
QLAST	1= 上一个有效值注入激活	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O
QUALITY	过程值状态	BYTE	0	O
SIM_I	仿真值	BOOL	0	IO
SIM_ON	1 = 激活仿真	BOOL	0	IO
SUBS_I	替代值	BOOL	0	IO
SUBS_ON	1 = 启用替代	BOOL	0	IO
VALUE	输入值	BOOL	0	IO
VALUE_QC	过程映像中的值状态	BOOL	0	IO

4.9 CH_DO: 数字值输出

4.9.1 CH_DO 的描述

对象名 (类型 + 编号)

FC 278

- CH_DO 块 I/O (页 381)

应用领域

块 CH_DO 处理 S7-300/400 SM 数字输出模块的数字输出信号。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32) 和重启 OB 100。

注意

对于 PCS 7, 不应在设备块和输出驱动程序之间插入其它块。

如果违背此原则, 则在互连该块时, 应确保在设备块的输出与输出驱动程序之间, 所有组成输出信号的块都安装在同一 OB 中。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- MODE 输入与 MOD 块的相应 OMODE_xx 输出互连。
- CH_DO 块安装在 OB 100 中为其分配的 MOD 块的下游。
- START_ON 输入采用相应的值进行组态。仅当 START_ON = 1 时, 才组态 START_I 输入。

功能和工作原理

块 CH_DO 循环处理所有通道特定信号功能。

块将数字值写入过程映像 (分区)。如果 SM 模块的 MODE 设置中输入参数的高位字节 = 0 (值状态), 则仍会将此数字值写入过程映像 (分区), 但会设置“无效值”质量代码。

质量代码

质量代码可有以下状态:

状态	质量代码
有效值	16#80
仿真	16#60
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成的。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

寻址

符号表中通过 HW Config 为数字输出通道生成的符号必须与 VALUE 输出参数互连。

正常值

数字值被写入过程映像（分区），并且质量代码 (QUALITY) = 16#80。

仿真

输入参数 SIM_ON = TRUE 时，则会将输入参数 SIM_I 的值写入过程映像（分区），并设置质量代码 QUALITY = 16#60。复位 QBAD = TRUE。仿真拥有最高优先级。如果块处于仿真状态，则 QSIM = TRUE。

I/O 故障

如果 MODE (页 709) 输入参数的高位字节为 0（值状态），则质量代码设置为 QUALITY = 16#00。该功能始终将当前数字值写入过程映像（分区）。

错误处理

不检查输入参数的似然性。

4.9 CH_DO: 数字值输出

启动特征

MOD 块在 OB 100 中的 OMODE (页 719)_xx 输出参数的字节 2 中设置 LSB。如果块检测到此代码,则会发送确认响应,并且:

如果未设置 START_ON,则将过程值 I 写入过程映像;否则将使用 START_I 替代此过程值。

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息,请参考以下部分:

关于使用驱动程序块的注意事项 (页 329)

4.9.2 CH_DO 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
I	过程值	BOOL	0	IO
MODE	值状态和模式	DWORD	0	IO
QBAD	1 = 输出值无效	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QUALITY	输出值的值状态	BYTE	0	O
SIM_I	仿真值	BOOL	0	IO
SIM_ON	1 = 激活仿真	BOOL	0	IO
START_I	启动时的替代值	BOOL	0	IO
START_ON	1 = 启动时替代	BOOL	0	IO
VALUE	PI 输出值	BOOL	0	O

4.10 CH_MS: ET 200S 电机起动器模块的信号处理

4.10.1 CH_MS 的描述

对象名 (类型 + 编号)

FB 60

- CH_MS 块 I/O (页 387)

应用领域

块 CH_MS 用于处理 ET 200S 电机启动器模块的信号。

调用 OB

接收和发送数据的循环 OB (建议 100 毫秒)。

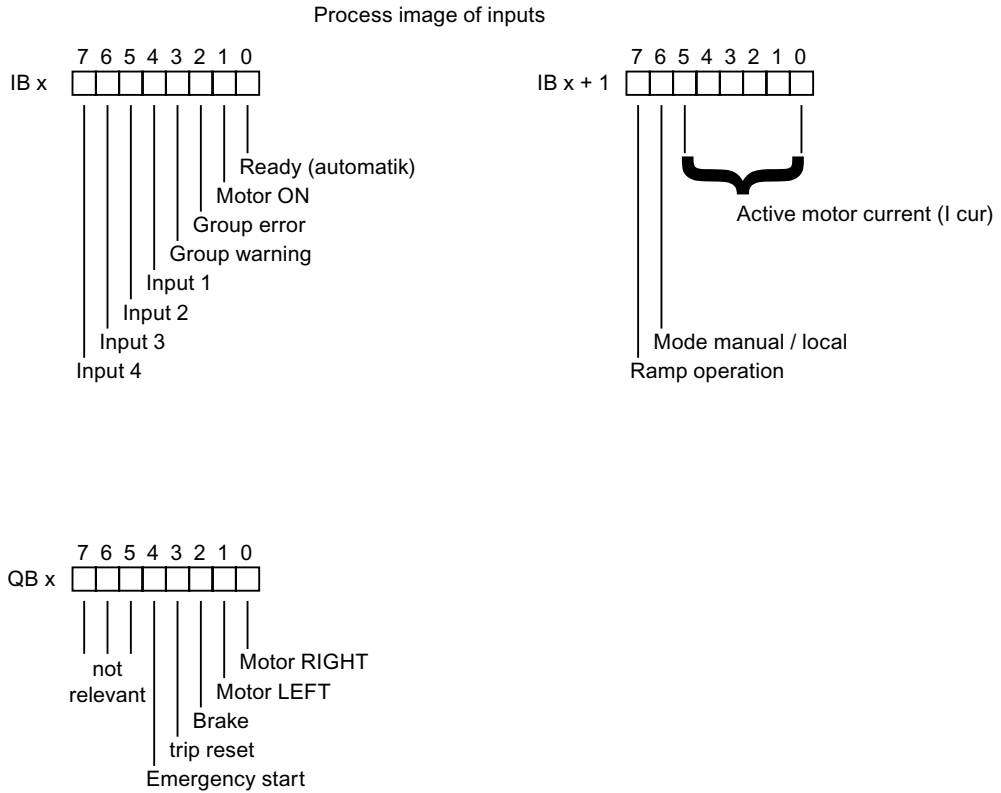
在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- CH_MS 块安装在 OB 100 中为其分配的 MOD 块的下游。
- MODE 输入与 MOD 块的相应 OMODE_xx 输出互连。
- 输入 IN_x 和输出 OUT_x 与电机起动器模块图标互连。

功能和工作原理

电机起动器模块按如下方式占用过程映像:



示意图中给出的输入均从过程映像中获取，再分别用于输出。

电机起动器模块提供的 6 位值指定电机电流比，即 I 电流/I 的额定电流（额定电流 = 在 HW Config 中设置的额定工作电流）。指定值的表达方式为：小数点前一位（位 5），小数点后五位（位 4 到 0）。因此，I 电流/I 的额定电流的最大比率是 1.96875（约为 197%）。

$$I \text{ 比率} = I \text{ 额定电流} \times \text{值 (位 5 到 0)}$$

位 5	位 4	位 3	位 2	位 1	位 0	
1	0,5	0,25	0,125	0,0625	0,03125	合计 = 1.96875
0	0	0	0	0	0	I 比率 = 0
1	0	0	0	0	0	I 比率 = 1
1	0	1	1	0	0	I 比率 = 1.375
1	1	1	1	1	1	I 比率 = 1.96875

将电机电流比的各个位分组，然后作为 REAL 值输出。

如果 OMODE (页 719) 输入参数的高位字节 = 16#40xxxxxx (值状态 = 较高级别错误、QMOD_ERR = TRUE)，则将该数字值视为无效。

质量代码

程序将生成结果值的质量代码，可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
上一个有效值	16#44
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

质量代码保存在 MODE 参数的字节 2 和 3 中。

寻址

将 HW Config 为电机起动器模块输入生成的符号（符号表）中的第一个符号与 VALUE 输入互连。

仿真

如果输入参数 SIM_ON = TRUE，则输出输入参数 SIM_U 的值（编码方式类似过程输入映像中两字节的结构）和质量代码 QUALITY = 16#60。复位 QBAD = TRUE。仿真拥有最高优先级。如果该块处于仿真状态，则会设置 QSIM = TRUE。

替代值

不能设置替代值。

保持上一个值

如果输入参数 `LAST_ON = TRUE`, 则在数字信号无效时, 输出上一个有效输出值。质量代码将设置为 `QUALITY = 16#44` 且 `QBAD = 1`。

输出无效值

如果输入参数 `LAST_ON = FALSE`, 并且存在一个无效过程值, 则输出此无效值并将 `QBAD` 设置为 1。

冗余

在 H 系统中, 较高级别的块评估 DP 主站系统的冗余。

错误处理

不检查输入参数的似然性。

启动特征

启动和初始化运行时, 将输入的当前过程值写入过程映像。

过载行为

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

4.10 CH_MS: ET 200S 电机起动器模块的信号处理

更多信息

更多相关信息, 可参考:
关于使用驱动程序块的注意事项 (页 329)

参见

SM 模块的 MODE 设置 (页 709)

4.10.2 CH_MS 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
BRAKE	制动器	BOOL	0	I
EMCY_ST	紧急启动	BOOL	0	I
FORWARD	电机正转	BOOL	0	I
IN_x	输入值 (x = 0 到 15)	BOOL	0	O
IN_NO	电机起动器模块输入的位数	BYTE	0	I
LAST_ON	1 = 上一个有效值: 启用注入	BOOL	0	IO
MODE	值状态和模式	DWORD	0	I
OUT_x	输出值 (x = 0 到 15)	BOOL	0	O
OUT_NO	电机起动器模块输出的位数	BYTE	0	I
QAUTO	就绪 (自动)	BOOL	0	O
QBAD	1 = 无效过程值	BOOL	0	O
QBRAKE	制动器	BOOL	0	O
QEMCY_ST	紧急启动	BOOL	0	O
QERROR	组错误	BOOL	0	O
QFORWARD	电机正转	BOOL	0	O
QINPUT_1	输入 1	BOOL	0	O
QINPUT_2	输入 2	BOOL	0	O
QINPUT_3	输入 3	BOOL	0	O
QINPUT_4	输入 4	BOOL	0	O
QLAST	1 = 上一个有效值: 注入激活	BOOL	0	O
QMANUAL	手动/本地	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QRAMP	斜坡操作	BOOL	0	O
QRESET	跳闸复位	BOOL	0	O

4.10 CH_MS: ET 200S 电机起动器模块的信号处理

I/O (参数)	含义	数据类型	默认值	类型
QRES_x	保留 (x = 1 到 11)	BOOL	0	O
QREVERS	电机反转	BOOL	0	O
QRUN	电机启动	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QUALITY	过程值状态	BYTE	0	O
QWARN	组警告	BOOL	0	O
RESET	跳闸复位	BOOL	0	I
REVERS	电机反转	BOOL	0	I
RAT_CURR	电机电流比	REAL	0	O
RES_x	保留 (x = 1 到 11)	BOOL	0	I
SIM_ON	1 = 激活仿真	BOOL	0	I
SIM_U	仿真值	WORD	0	I
VALUE	输入值	BOOL	0	I

4.11 CH_U_AI: 模拟值输入 (通用)

4.11.1 CH_U_AI (通用) 的描述

对象名 (类型 + 编号)

FC283

- CH_U_AI 块 I/O (页 396)

应用领域

块 CH_U_AI 处理 PA 现场设备 (PA 配置文件 3.0 模拟输入) 或 HART 现场设备 (主要或辅助变量) 的 S7-300/400 SM 模拟输入模块的模拟输入信号。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32) 和重启 OB 100。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 根据所使用的 I/O 设备设置输入 PA_ON 参数, 即, 对于 S7 信号模块设置为 0, 对于 PA 现场设备或 HART 现场设备的主要/辅助变量设置为 1。
- 模拟输入通道的质量代码的图标与 (PA 设备的) 输入 QC 互连。
- 输入 MODE 与 MOD 块 (或 PADP 块) 的相应输出 OMODE_xx 互连。

4.11 CH_U_AI: 模拟值输入 (通用)

功能和工作原理

块 CH_U_AI 循环处理所有通道特定信号功能或现场设备的 REAL 数据类型信号 (带或不带质量代码)。

该块使用变量 (输入参数 PA_ON) 区别模拟原始值和带质量代码的 PA 现场设备 REAL 值。更多相关信息, 可参考“寻址”部分。

- PA_ON = TRUE

必须满足条件 PQC = TRUE, 因为始终使用质量代码定义 PA 现场设备的 REAL 值或 HART 现场设备的主要/辅助变量。

如果 MODE (页 709) 输入参数的高位字节 = 16#40 (值状态 = 较高级别错误、QMOD_ERR = TRUE), 则根据 PA_ON = FALSE 确定质量代码。

- PA_ON = FALSE

该块从过程映像 (分区) 读取模拟原始值, 然后转换为相应的物理值或计算百分比值。输入 MODE 的状态确定原始值的格式和处理方法。如果 MODE 输入参数的高位字节是 16#40 (值状态 = 较高级别错误), 则原始值将被视为无效。

质量代码

程序将生成结果值的质量代码, 可能为以下状态:

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误, 上一个有效值	16#14
较高级别错误, 替代值	16#18
不良, 过程相关	16#28
不确定, 设备相关	16#68
不确定, 过程相关	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成的。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

寻址

必须将在 HW Config 中为模拟输入通道的符号表生成的符号与输入参数 VALUE 或 (PA_ON = TRUE) OUT (过程值) 和 QC (质量代码) 互连。

原始值检查

只有 PA_ON = FALSE 时：根据测量类型和模拟输入模块的范围，标称范围设置了模拟信号转换为数字值（原始值）的范围。还存在一个过冲/下冲范围，在该范围内模拟信号仍可转换。如果值超出此范围，就构成上溢或下溢。块会指出原始值是否在模块的标称范围内。

如果值超出了标称范围的下限，输出参数 QCHF_LL = TRUE。如果值超出了标称范围的上限，输出参数 QCHF_HL = TRUE。如果由于模块诊断“过冲/下冲测量范围”而出现通道错误，QCHF_LL/QCHF_HL 将保持设置为 TRUE。

发生信号上溢或下溢错误时也会设置 QBAD = TRUE（通道错误）。

说明

模块对于 4 mA 到 20 mA 信号线的断路反应不一致。根据模块不同，会将 16#7FFF（上溢）或 16#8000（下溢）作为原始值写入过程映像。然后，通道块 CH_U_AI 将相应地输出上溢 (QCHF_HL = TRUE) 或下溢 (QCHF_LL = TRUE) 信号，每个信号都带有 QBAD = TRUE。例外：在 HW Config 中设置了模拟输入模块的“诊断中断”后，如果发生“通道错误”（例如由于断路）后触发诊断中断，则将只设置 QBAD = TRUE。

NAMUR 限制检查

只有 PA_ON = FALSE 时: 模拟信号处理的 NAMUR 规则为存在通道错误的零点 (4 到 20 mA) 模拟信号定义以下限制:

$$3.6 \text{ mA} \leq \text{模拟信号} \leq 21 \text{ mA}$$

上述 NAMUR 限制设置为限制值监视的固定默认值。如果要设置其它限制, 必须将 CH_F_ON 输入参数设置为 TRUE, 然后在 CH_F_HL 和 CH_F_LL 输入参数相应设置以 mA 为单位的新限制。如果零点模拟信号超出电流上下限范围, 则 QBAD = TRUE。

说明

可选择的限制必须在模块的过冲和下冲范围内。如果模块不自动限制测量值, 也有可能出现超出 NAMUR 范围的值。

正常值

只有 PA_ON = FALSE 时: 原始值根据输入参数 VLRANGE、VHRANGE 和 MODE 中的设置转换为相应的物理值。这些值将写入输出 OVL RANGE 和 OVHRANGE, 以允许 VLRANGE 和 VHRANGE 的设置与其它块 I/O 互连。转换算法取决于线性输入信号。如果 VLRANGE = 0 且 VHRANGE = 100, 则转换为百分比值。如果设置了 VHRANGE = VLRANGE, 则将按照 MODE (页 709) 设置获取模拟输入模块的输入信号 (例如, mA)。如果原始值已经是物理值, 则设置 VLRANGE = 0 和 VHRANGE = 1。质量代码设置为 QUALITY = 16#80。

在 PTC 测量模式下运行时, 模拟值包含编码的二进制信号。输出提供以下信息:

- 如果被测电阻在正常范围内, 则 PV_Out = 0.0。
- 如果被测电阻在预警范围内, 则 PV_Out = 4.0。
- 如果被测电阻在工作范围内, 则 PV_Out = 1.0。

这只有在设置输入参数 VLRANGE = 0 且 VHRANGE = 1 时才适用。仿真值 SIM_V 和替代值 SUBS_V 仅应设为 0 或 1。

说明

在“热电偶值的外部或内部比较”测量模式下, 物理单元将调整到 S7 300 模块的 +/- 80 mV 范围。必须通过相应的转换表确定温度。

如果模块将以 mV 为单位的物理当量作为原始值传送, 则将 VHRANGE 和 VLRANGE 设置为 +/- 80 mV。

仿真

如果输入参数 $SIM_ON = TRUE$, 则输出 SIM_V 输入参数值和质量代码 $QUALITY = 16\#60$ 。 $QBAD = TRUE$: 由于较高级别错误 ($QMOD_ERR = TRUE$) 复位。还必须在仿真模式下在 $MODE$ (页 709) 输入的低位字中设置有效运行模式。否则, 将输出 $QBAD = 1$ 。仿真拥有最高优先级。仿真值将根据运行模式以及输入参数 $VHRANGE$ 和 $VLRANGE$ 转换为原始值。该值的验证方式与过程映像中原始值的验证方式相同。这允许仿真 $QBAD$ 、 $QCHF_LL$ 和 $QCHF_HL$ 状态。

如果 $VLRANGE > VHRANGE$, 则不能对状态 $QBAD = TRUE$ 进行仿真。根据 SIM_V 值设置 $QCHF_LL$ 和 $QCHF_HL$ 输出。

如果 $QBAD$ 会在单极测量范围的负范围内形成, 则必须将该值设置为 -119% 。

如果 $PA_ON = TRUE$, 则不执行原始值转换。 $SIM_ON = TRUE$ 时, 始终设置 $QBAD = FALSE$ 。

如果块处于仿真状态, 则 $QSIM = TRUE$ 。

说明

请记住, 仿真值总是在仿真模式下输出, 与任一参数 $LAST_ON$ (替代值) 或 $SUBS_ON$ (上一个有效值) 都无关。

替代值

输入参数 $SUBS_ON = TRUE$ 且原始值无效时, 将输出输入参数 $SUBS_V$ 的值作为替代值。质量代码将设置为 $QUALITY = 16\#48$ 且 $QBAD = 1$ 。

保持上一个值

如果输入参数 $LAST_ON = TRUE$, 则在 δ 值无效时, 输出上一个有效输出值 (V_LAST)。如果 $V_DELTA > 0$, 则应用以下设置:

- $ABS(V - V_LAST) > V_DELTA$: $V = V_LAST1$ (倒数第二个有效输出值)
- $ABS(V - V_LAST) \leq V_DELTA$: $V = V_LAST$ (上一个有效输出值)

质量代码将设置为 $QUALITY = 16\#44$ 、 $DELTA_ON$ 和 $QBAD = 1$ 。

如果有效原始值可用, $V_DELTA > 0$ 且 $ABS(V - V_LAST) > V_DELTA$, 则上一个有效输出值 (V_LAST) 和 $QUALITY = 16\#44$ ($QBAD = 0$) 将保持一个周期。

4.11 CH_U_AI: 模拟值输入 (通用)

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE, 或均为 TRUE, 并且存在一个无效过程值, 则会输出此无效值并将 QBAD 设置为 1。

延迟值接受

只有 PA_ON = FALSE 时: 质量代码从“BAD”更改为“GOOD”时或者重新启动后, 如果 CNT_LIM 中指定的周期数没有结束, 不会更新质量代码和值。CNT_LIM = 0 (默认设置) 时, 将禁用此功能。在延迟时间内将保留 Quality Code = 16#00 且 QBAD = 1 和上一个值。

值限制

PA_ON = TRUE 时, 可设置过程映像 (分区) 过程值的限制过滤器。

如果开关 LIMIT_ON = TRUE, 则按如下所示限制过程值 (V):

- 如果 $V > V_{HL}$, 则为 V_{HL}
- 如果 $V < V_{LL}$, 则为 LL_V

错误处理

不检查输入参数的似然性。如果在 MODE (页 709) 输入参数的低位字中设置了无效模式, 则假定原始值是无效的。

启动特征

CNT_LIM 为 # 0 时启动接受值延迟。

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息, 请参考以下部分:

关于使用驱动程序块的注意事项 (页 329)

4.11.2 CH_U_AI 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
CH_F_HL	输入值的过冲上限 (mA)	REAL	0	IO
CH_F_LL	输入值的下冲下限 (mA)	REAL	0	IO
CH_F_ON	1 = 激活限制监视	BOOL	0	IO
CNT_LIM	启动计数器限制	INT	0	IO
CNT_RES	启动计数器	INT	0	IO
DELTA_ON	超过上一个 delta 过程值	BOOL	0	IO
LAST_BAD	上一个无效过程值	BOOL	0	IO
LAST_ON	1 = 上一个有效值: 启用注入	BOOL	0	IO
LIMIT_ON	1 = 启用对 PA 现场设备过程值的限制	BOOL	0	IO
LL_V	过程值, 如果 $V < V_{LL}$	REAL	0	IO
MODE	值状态和模式	DWORD	0	IO
OUT	过程映像输入值	REAL	0	IO
OVHRANGE	过程值上限 (复制)	REAL	0	O
OVL RANGE	过程值下限 (复制)	REAL	0	O
PA_ON	1 = 在过程映像中使用 PA 现场设备	BOOL	0	IO
PQC	1 = 在过程映像中使用值状态	BOOL	0	IO
QBAD	1 = 无效过程值	BOOL	0	O
QC	输入过程值的状态	BYTE	0	IO
QCHF_HL	1 = 输入值上限	BOOL	0	O
QCHF_LL	1 = 输入值下限	BOOL	0	O
QLAST	1 = 上一个有效值: 注入激活	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O

I/O (参数)	含义	数据类型	默认值	类型
QUALITY	过程值状态	BYTE	0	O
SIM_ON	1 = 激活仿真	BOOL	0	IO
SIM_V	仿真值	REAL	0	IO
STATUS	过程值状态	BYTE	0	O
SUBS_ON	1 = 启用替代	BOOL	0	IO
SUBS_V	替代值	REAL	0	IO
V	过程值	REAL	0	O
V_DELTA	过程值增量 (V - V_LAST)	REAL	0	IO
V_HL	上限	REAL	0	IO
V_LAST	上一个有效过程值	REAL	0	IO
V_LAST1	倒数第二个有效过程值	REAL	0	IO
V_LL	下限	REAL	0	IO
VALUE	输入值	WORD	0	IO
VHRANGE	过程值上限	REAL	100	IO
VLRANGE	过程值下限	REAL	0	IO

4.12 CH_U_AO: 模拟值输出 (通用)

4.12.1 CH_U_AO 的描述

对象名 (类型 + 编号)

FC 284

- CH_U_AO 块 I/O (页 403)

应用领域

块 CH_U_AO 处理 S7-300/400 SM 模拟输出模块或 PA 现场设备 (PA 配置文件 3.0 模拟输出, 仅输出 REAL 值 [例如, SP] 和质量代码) 的模拟输出信号。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32) 和重启 OB 100。

注意
对于 PCS 7, 不应在设备块和输出驱动程序之间插入其它块。 如果违背此原则, 则在互连该块时, 应确保在设备块的输出与输出驱动程序之间, 所有组成输出信号的块都安装在同一 OB 中。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时，将自动执行以下操作：

- 根据所使用的 I/O 设备组态 PA_ON 输入（S7 信号模块 [= 0] 或 PA 现场设备 [= 1]）。
- 模拟输出通道质量代码的图标与（PA 设备的）QC_SP 输出互连。
- 输入 MODE 与 MOD 块（或 PADP 块）的相应输出 OMODE_xx 互连。
- CH_U_AO 块安装在在 OB 100 中为其分配的 MOD/PADP 块的下游。

说明

如果不使用 CFC 功能“生成模块驱动程序”，则必须确保 CH_U_AO 块安装在 OB 100 中为其分配的 MOD/PADP 块的下游。

功能和工作原理

块 CH_U_AO 循环处理所有特定通道信号功能/带质量代码的 REAL 值。

块 CH_U_AO 使用变量（输入参数 PA_ON）区别模拟原始值和带有质量代码的 PA 现场设备的 REAL 值。更多相关信息，可参考“寻址”部分。

- PA_ON = TRUE

块将 PA 现场设备的 REAL 值（SP）和质量代码（ST_SP）写入过程映像（分区）。

- PA_ON = FALSE

块将过程值作为模拟原始值写入过程映像（分区）。MODE 输入参数确定生成原始值所采用的格式。

如果 MODE (页 709) 输入参数的高位字节 = 0（值状态），则原始值仍被写入过程映像（分区），但质量代码为“无效值”。

质量代码

质量代码可有以下状态:

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误, 上一个有效值	16#14
较高级别错误, 替代值	16#18
不良, 过程相关	16#28
不确定, 设备相关	16#68
不确定, 过程相关	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
上限值	16#56
下限值	16#55
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件 (例如通道错误、较高级别错误或仿真) 以及直接来自该设备的质量代码 (参数 QC) 形成。

根据 PROFIBUS 要求, 直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

寻址

在 HW Config 中为模拟输出通道生成的符号 (符号表) 必须互连到 VALUE 输出参数。

对于 PA 现场设备 (PA_ON = TRUE), 必须将 REAL 值的图标与 O_SP 输出参数互连。

正常值

- 只有 PA_ON = FALSE 时: 参数 ULRANGE 和 UHRANGE 依照 MODE (页 709) 将过程值 U 映射到模拟输出模块的原始 VALUE (量化步骤)。例如, 在 4 mA 到 20 mA (16#0203) 模式下, 如果 U = ULRANGE, 则输出 4 mA 的原始值; 如果 U = UHRANGE, 则输出 20 mA 的原始值。
- 该块切换参数 UHRANGE 和 ULRANGE 直到输出 OVHRANGE 和 OVLRange。例如, 可以互连这些输出与控制器 CTRL_PID 的调节变量限值 NM_LMNHR 和 NM_LMNLr。
- 可以用 PHYS_LIM 设置初始 VALUE 的限制值。默认设置 PHYS_LIM = 0 将输出 VALUE 的值限制到模块的默认限值内。按照上述实例, 如果 U > UHRANGE, 则块计算 20 mA 的原始值; 如果 U < ULRANGE, 则计算 4 mA 的原始值。因此, QUALITY 输出应用质量代码 16#56 (上限值) 和 16#55 (下限值), 而非 16#80 (有效值)。
- 如果要输出的模拟值位于默认限值以外, 但在模块的物理限值内, 则设置 PHYS_LIM = 1。例如, 在上述实例中, 只有当 ULRANGE = 0 且 UHRANGE = 100 时, 通过指定 U = 200 (36 mA) 或 U = -50 (-4 mA) 来超出模块限值时, 才可限制输出值。这时输出值被限制在模块的数据表中指定的物理限值内, 同时输出相应的质量代码。
- QCHF_HL 和 QCHF_LL 输出同样也提供有关是否已设置输出值限制的信息。

仿真

如果设置输入参数 SIM_ON = TRUE, 则输出 SIM_U 的值和质量代码 (QUALITY) = 16#60。始终复位 QBAD。仿真拥有最高优先级。如果块处于仿真状态, 则 QSIM = TRUE。

I/O 故障

如果 MODE 输入参数的高位字节设置为 0 (值状态), 则质量代码设置为 QUALITY = 16#00。实际的原始值将始终写入过程映像 (分区)。

值限制

只有 PA_ON = FALSE 时: 在将过程值输入过程映像 (分区) 之前, 可以限制可导致错误 (QBAD = TRUE) 的非常低或非常高的过程值。

如果 LIMIT_ON 开关 = TRUE, 则按如下所述限制过程值 (U):

- 如果 $U > V_{HL}$, 则为 V_{HL}
- 如果 $U < V_{LL}$, 则为 LL_V 。

错误处理

不检查输入参数的似然性。如果 MODE (页 709) 输入的低位字中设置的运行模式无效, 则会将数字化输出值设置为 0, 并输出 QUALITY = 16#00。

启动特征

MOD 块在 OB 100 中的 OMODE_xx (页 719) 输出参数的字节 2 中设置 LSB。如果块检测到此代码, 则会发送确认响应, 并且:

- 如果未设置 START_ON, 则计算过程值 U, 并将结果写入过程映像。
如果置位了 START_ON, 则将与 START_U 过程值对应的原始值写入过程映像。
- 启用 PA 现场设备时 (PA_ON = TRUE), 实际 REAL 值和质量代码将写入过程映像。

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息, 请参考以下部分:

关于使用驱动程序块的注意事项 (页 329)

4.12.2 CH_U_AO 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
LIMIT_ON	1 = 启用过程值限制	BOOL	0	IO
LL_V	过程值, 如果 $U < V_{LL}$	REAL	0	IO
MODE	值状态和模式	DWORD	0	IO
O_SP	过程映像设定值	REAL	0	O
OVHRANGE	过程值的输出上限	REAL	100	O
OVLRange	过程值的输出下限	REAL	0	O
PA_ON	1 = PA 现场设备、0 = 信号模块	BOOL	0	IO
PHYS_LIM	1 = 启用模块的物理限值	BOOL	0	IO
QBAD	1 = 输出值无效	BOOL	0	O
QCHF_HL	1 = 过程值过冲	BOOL	0	O
QCHF_LL	1 = 过程值下冲	BOOL	0	O
QC_SP	过程映像质量代码设定值	BYTE	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QUALITY	输出值的值状态	BYTE	0	O
SIM_ON	1 = 激活仿真	BOOL	0	IO
SIM_U	仿真值	REAL	0	IO
START_ON	1 = 启动时替代	BOOL	0	IO
START_U	启动时的替代值	REAL	0	IO
ST_SP	设定值状态	BYTE	0	IO
U	过程值	REAL	0	IO
UHRANGE	过程值上限	REAL	100	IO
ULRANGE	过程值下限	REAL	0	IO
V_HL	上限	REAL	0	IO

系列: DRIVER

4.12 CH_U_AO: 模拟值输出 (通用)

I/O (参数)	含义	数据类型	默认值	类型
V_LL	下限	REAL	0	IO
VALUE	输出值	WORD	0	O

4.13 CH_U_DI: 数字值输入 (通用)

4.13.1 CH_U_DI (通用) 的描述

对象名 (类型 + 编号)

FC285

- CH_U_DI 块 I/O (页 409)

应用领域

块 CH_U_DI 处理 S7-300/400 SM 数字输入模块或 PA 现场设备 (PA 配置文件 3.0 离散输入) 的数字输入信号。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 根据所使用的 I/O 设备组态 PA_ON 输入 (S7 信号模块 [= 0] 或 PA 现场设备 [= 1])。
- 数字输入通道的质量代码的图标与 (PA 设备的) 输入 QC 互连。
- 输入 MODE 与 MOD 块 (或 PADP 块) 的相应输出 OMODE_xx 互连。

功能和工作原理

块 CH_U_DI 循环处理 PA 现场设备的所有通道特定信号功能/带质量代码的过程值。

块使用变量 (PA_ON 输入参数) 区别 PA 现场设备的带或不带 BOOL 数据类型质量代码的数字值和带 BYTE 数据类型质量代码的数字值。更多相关信息, 可参考“寻址”部分。

- PA_ON = TRUE

该块从过程映像 (分区) 循环读取 PROFIBUS PA 现场设备 (结构按照 PA 配置文件的离散输入) 的过程值 (OUT_D) 和状态字节 (QUALITY, 请参见“寻址”)。状态字节中包含有关 PROFIBUS 现场设备的测量值和状态的信息。块将过程值传送到输出 Q, 如下所示:

- 如果过程值 = 0, 则 Q = FALSE
- 如果过程值 <> 0, 则 Q = TRUE
如果 MODE (页 709) 输入参数的高位字节 = 16#40 (值状态 = 较高级别错误), 则将过程值和质量代码视为 PA_ON = FALSE。

- PA_ON = FALSE

块从过程映像 (分区) 读取 BOOL 数据类型的数字值。如果 MODE 输入参数的高位字节 = 16#40 (值状态 = 较高级别错误、QMOD_ERR = TRUE), 则将该数字值视为无效。如果输入参数 PQC = TRUE, 则从过程映像 (分区) 读取数字值的值状态。

质量代码

程序将生成结果值的质量代码, 可能为以下状态:

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误, 上一个有效值	16#14
较高级别错误, 替代值	16#18
不良, 过程相关	16#28
不确定, 设备相关	16#68
不确定, 过程相关	16#78
上一个有效值	16#44
替代值	16#48

状态	质量代码
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

寻址

在 HW Config 中为数字输入通道生成的符号（符号表）必须互连到 VALUE 输入。如果过程映像（分区）中还包含数字输入通道的值状态，则将相应的符号互连到输入 VALUE_QC 并设置输入 PQC = TRUE。

使用 PA 现场设备 (PA_ON = TRUE) 时，将 HW Config（符号表）中为数字输入通道生成的符号与输入 I_OUT_D 互连。

正常值

过程映像（分区）的数字值和质量代码 QUALITY = 16#80 都应用于输出 Q。

仿真

如果 SIM_ON 输入参数 = TRUE，则输入参数 SIM_I (PA_ON = FALSE) 或 SIM_OUT (PA_ON = TRUE) 的值将与质量代码 QUALITY = 16#60 一起输出到输出参数 Q。始终会复位 QABD。仿真拥有最高优先级。

如果块处于仿真状态，则 QSIM = TRUE。

说明

请记住，仿真值总是在仿真模式下输出，与任一参数 LAST_ON（替代值）或 SUBS_ON（上一个有效值）都无关。

替代值

输入参数 SUBS_ON = TRUE 和过程映像（分区）的数字值无效时，此功能将信号 QBAD = 1 和输入参数 SUBS_I 的值以及质量代码 QUALITY = 16#48 输出到输出参数 Q。

4.13 CH_U_DI: 数字值输入 (通用)

保持上一个值

如果输入参数 SUBS_ON = FALSE, 则当原始值无效时, 该功能输出上一个有效输出值。质量代码将设置为 QUALITY = 16#44 且 QBAD = 1

上一个有效输出值 = Q_LAST。

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE, 或均为 TRUE, 并且存在一个无效过程值, 则会输出此无效值并将 QBAD 设置为 1。

错误处理

不检查输入参数的似然性。

启动特征

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息, 请参考以下部分:

关于使用驱动程序块的注意事项 (页 329)

4.13.2 CH_U_DI 的 I/O

CFC 中块显示的出厂设置在“**I/O**”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
I_OUT_D	过程映像的输入值	BYTE	0	IO
LAST_ON	1 = 上一个有效值: 启用注入	BOOL	0	IO
MODE	值状态和模式	DWORD	0	IO
OUT_D	过程值	BYTE	0	O
PA_ON	1 = PA 现场设备	BOOL	0	IO
PQC	1 = 在过程映像中使用值状态	BOOL	0	IO
Q	过程值	BOOL	0	O
Q_LAST	上一个有效过程值	BOOL	0	IO
QBAD	1 = 无效过程值	BOOL	0	O
QC	过程映像中的值状态 (图标)	BYTE	0	IO
QLAST	1 = 上一个有效值: 注入激活	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O
QUALITY	过程值状态	BYTE	0	O
SIM_I	输入仿真值	BOOL	0	IO
SIM_ON	1 = 激活仿真	BOOL	0	IO
SIM_OUT	输出模块值	BYTE	0	IO
STATUS	过程值状态	BYTE	0	O
SUBS_I	输入替代值	BOOL	0	IO
SUBS_ON	1 = 启用替代	BOOL	0	IO
SUBS_OUT	输出替代值	BYTE	0	IO
V_LAST	上一个有效过程值	BYTE	0	IO

系列: DRIVER

4.13 CH_U_DI: 数字值输入 (通用)

I/O (参数)	含义	数据类型	默认值	类型
VALUE	输入值	BOOL	0	IO
VALUE_QC	过程映像中的值状态	BOOL	0	IO

4.14 CH_U_DO: 数字值输出 (通用)

4.14.1 CH_U_DO (通用) 的描述

对象名 (类型 + 编号)

FC 286

- CH_U_DO 块 I/O (页 415)

应用领域

块 CH_U_DO 处理 S7-300/400 SM 数字输出模块或 PA 现场设备 (PA 配置文件 3.0 离散输出, 仅限 SP 或 RCAS_IN) 的数字输出信号。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32) 和重启 OB 100。

注意
对于 PCS 7, 不应在设备块和输出驱动程序之间插入其它块。 如果违背此原则, 则在互连该块时, 应确保在设备块的输出与输出驱动程序之间, 所有组成输出信号的块都安装在同一 OB 中。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时，将自动执行以下操作：

- 根据所使用的 I/O 设备组态 PA_ON 输入（S7 信号模块 [= 0] 或 PA 现场设备 [= 1]）。
- 数字输出通道质量代码图标与（PA 设备的）QC_SP 输出互连。
- 输入 MODE 与 MOD 块（或 PADP 块）的相应输出 OMODE_xx 互连。
- CH_U_DO 块安装在 OB 100 中为其分配的 MOD/PADP 块的下游。

说明

如果不使用 CFC 功能“生成模块驱动程序”，则必须确保 CH_U_DO 块安装在 OB 100 中为其分配的 MOD/PADP 块的下游。

功能和工作原理

块 CH_U_DO 循环处理 PA 现场设备的所有特定通道信号功能/带质量代码的过程值。

块使用变量（PA_ON 输入参数）区别 PA 现场设备的不带 BOOL 数据类型质量代码的数字输出值和带有 BYTE 数据类型质量代码的数字输出值。更多相关信息，可参考“寻址”部分。

- PA_ON = TRUE

该块将过程值与质量代码一起写入过程映像（分区）（过程值的结构与 PA 配置文件的数字输出相对应，1 字节 (SP) 对应于 1 字节 (ST_SP) 质量代码）。质量代码包含有关过程值状态的信息。在 PROFIBUS 3.0“常规要求”中介绍了质量代码的编码。

如果 MODE (页 709) 输入参数的高位字节 = 16#40（值状态 = 较高级别错误，QMOD_ERR = TRUE），此功能会继续将过程值和质量代码写入过程映像（分区），但将 QUALITY 块输出处的质量代码设置为“无效值”。

- PA_ON = FALSE

块将数字值写入过程映像（分区）。如果 MODE 输入参数的高位字节 = 0（值状态），仍将此数字值写入过程映像（分区），但会设置“无效值”质量代码。

质量代码

质量代码可有以下状态:

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误, 上一个有效值	16#14
较高级别错误, 替代值	16#18
不良, 过程相关	16#28
不确定, 设备相关	16#68
不确定, 过程相关	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件 (例如通道错误、较高级别错误或仿真) 以及直接来自该设备的质量代码 (参数 QC) 形成。

根据 PROFIBUS 要求, 直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

寻址

符号表中通过 HW Config 为数字输出通道生成的符号必须与 VALUE 输出参数互连。

使用 PA 现场设备时 (PA_ON = TRUE), 符号表中为带有质量代码的过程值生成的符号必须与输出变量 O_SP 互连。

正常值

数字值被写入过程映像 (分区), 并且质量代码 (QUALITY) = 16#80。

4.14 CH_U_DO: 数字值输出 (通用)

仿真

输入参数 SIM_ON = TRUE 时, 输入参数 SIM_I (PA_ON = FALSE) 或 SIM_SP (PA_ON = TRUE) 的值被写入过程映像 (分区), 并设置质量代码 QUALITY = 16#60。始终复位 QBAD。仿真拥有最高优先级。如果块处于仿真状态, 则 QSIM = TRUE。

I/O 故障

如果 MODE (页 709) 输入参数的高位字节为 0 (值状态), 则质量代码设置为 QUALITY = 16#00。该功能始终将当前数字值写入过程映像 (分区)。

错误处理

不检查输入参数的似然性。

启动特征

MOD 块在 OB 100 中的 OMODE_xx (页 719) 输出参数的字节 2 中设置 LSB。如果块检测到此代码, 则会发送确认响应, 并且:

- 如果未设置 START_ON, 则将过程值 I 写入过程映像。
- 如果已设置 START_ON, 则使用 START_I 代替过程值 I。

使用 PA 现场设备时 (PA_ON = TRUE), 会将当前 BYTE 值和质量代码写入过程映像。

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息, 请参考以下部分:

关于使用驱动程序块的注意事项 (页 329)

4.14.2 CH_U_DO 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
I	过程值	BOOL	0	IO
MODE	值状态和模式	DWORD	0	IO
O_SP	过程映像设定值	BYTE	0	O
PA_ON	1 = PA 现场设备	BOOL	0	IO
QBAD	1 = 输出值无效	BOOL	0	O
QC_SP	过程映像质量代码设定值	BYTE	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QUALITY	输出值的值状态	BYTE	0	O
SIM_I	仿真值	BOOL	0	IO
SIM_ON	1 = 激活仿真	BOOL	0	IO
SIM_SP	仿真设定值	BYTE	0	IO
SP	Setpoint	BYTE	0	IO
ST_SP	设定值状态	BYTE	0	IO
START_I	启动时的替代值	BOOL	0	IO
START_ON	1 = 启动时替代	BOOL	0	IO
VALUE	输出值	BOOL	0	O

4.15 FF_A_AI: 使用 PA 配置文件发送器

4.15.1 FF_A_AI 的描述

对象名 (类型 + 编号)

FB 410

- FF_A_AI 块 I/O (页 420)

应用领域

块 FF_A_AI 处理 (循环服务) FF 现场设备的 PA 配置文件“发送器”。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 模拟输入通道的质量代码图标与输入 QC_W 互连。
- MODE 输入与 MOD_PAL0 块或 MOD_PAX0 块的 OMODE_00 输出互连。

功能和工作原理

块 FF_A_AI 从过程映像分区中循环读取 FF 现场设备的过程值 (质量代码)。得到的过程变量带有物理单位。状态字节 (STATUS) 包含 FF 现场设备状态的相关信息。

ASSET 中的链接

可在 ASSET 中链接块 FF_A_AI。

有关详细信息, 请参考以下部分:

在 ASSET 中集成 FF 设备 (页 727)

质量代码

程序将生成结果值的质量代码 (QUALITY 输出), 可能为以下状态:

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误, 上一个有效值	16#14
较高级别错误, 替代值	16#18
不良, 由过程引起	16#28
不确定, 由设备引起	16#68
不确定, 由过程引起	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件 (例如通道错误、较高级别错误或仿真) 以及直接来自该设备的质量代码 (参数 QC-W) 形成。

直接来自该设备的质量代码可以是 16#00 – 16#FF 之间的值。分析 FF 现场设备的质量代码 16#84-16#87 和 16#90 – 16#93 类似于分析质量代码 16#80 – 16#83。如果从 FF 设备传送质量代码 16#80 (有效值) 和一个位模式为 16#7FFFFFFF (无效值) 的过程值 (输入 VALUE), 则该 FF 现场设备将如同对质量代码 16#00 (无效值) 一样对它们进行处理。

除了提供状态字节及提高可互连性, 输出接口还提供了更多的布尔 (BOOL) 值, 它们包含重要的详细信息。它们符合 PROFIBUS PA 3.0“常规要求”中的位组合规范。

该块可通过 MODE (页 725) 输入参数识别较高级别错误, 例如, DP/PA 连接器的故障。

- 如果 MODE 高位字节 = 16#80, 则过程映像 (分区) 中的值有效。
- 如果 MODE 高位字节 = 16#40 (值状态 = 较高级别错误), 则模拟值将被视为无效。

不考虑输入参数 MODE_LW 的模式。

4.15 FF_A_AI: 使用 PA 配置文件发送器

寻址

必须在图标表中为模拟输入通道组态一个图标并将其与输入 VALUE 相连。CFC 功能“生成模块驱动程序”将模拟输入通道的质量代码图标与输入参数 QC_W 互连。

仿真

输入参数 SIM_ON = TRUE 时，将输出输入参数 SIM_V 的值及质量代码 (QUALITY =) 16#60。仿真拥有最高优先级。QBAD 始终设置为 FALSE。如果块处于仿真状态，则设置 QSIM = TRUE。

替代值

如果输入参数 SUBS_ON = TRUE，则在值无效时，将输入参数 SUBS_V 的值将作为输出值。将质量代码 (QUALITY) 设置为 16#44 且 QBAD = 1。

输出无效值

如果输入参数 SUBS_ON = FALSE，并且存在一个无效过程值，则输出此无效值并将 QBAD 设置为 1。

值限制

可限制过程映像（分区）中过程变量的最低和最高值的范围。

LIMIT_ON = TRUE 限制过程变量 (V):

- 如果 $V > V_{HL}$ ，则为 V_{HL}
- 如果 $V < V_{LL}$ ，则为 LL_V

错误处理

不检查输入参数的似然性。

启动特征

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.15.2 FF_A_AI 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
LIMIT_ON	1 = 启用限制输入值	BOOL	0	I
LL_V	在 $V < V_{LL}$ 时的值	REAL	0	I
MODE	值状态	DWORD	0	I
MODE_LW	模式	WORD	1	I
QA_1	1 = 报警/警告 1	BOOL	0	O
QA_2	1 = 报警/警告 2	BOOL	0	O
QBAD	1 = 组事件 QBAD_X	BOOL	0	O
QBAD_0	1 = 未指定	BOOL	0	O
QBAD_1	1 = 组态错误	BOOL	0	O
QBAD_2	1 = 未连接	BOOL	0	O
QBAD_3	1 = 设备故障	BOOL	0	O
QBAD_4	1 = 传感器故障	BOOL	0	O
QBAD_5	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_6	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_7	1 = 服务中断	BOOL	0	O
QBAD_HL	1 = 已达到传感器物理范围的上限	BOOL	0	O
QBAD_LL	1 = 已达到传感器物理范围的下限	BOOL	0	O
QC_W	输入过程值的状态	WORD	0	I
QCASCAD0	1 = 正常 (级联)	BOOL	0	O
QCASCAD1	1 = 初始化已确认	BOOL	0	O
QCASCAD2	1 = 初始化请求	BOOL	0	O
QCASCAD3	1 = 不需要	BOOL	0	O
QCASCAD4	1 = 保留	BOOL	0	O

I/O (参数)	含义	数据类型	默认值	类型
QCASCAD5	1 = 不选择	BOOL	0	O
QCASCAD6	1 = 本地覆盖	BOOL	0	O
QCASCAD7	1 = 保留	BOOL	0	O
QCASCAD8	1 = 启动故障安全	BOOL	0	O
QCONST	1 = 常量	BOOL	0	O
QERR	1 = 输出错误 (ENO 的取反值)	BOOL	1	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QOUT_HHL	1 = 激活临界报警, 已超出 OUT 的上限	BOOL	0	O
QOUT_HL	1 = 激活警告, 已超出 OUT 的上限	BOOL	0	O
QOUT_LL	1 = 激活警告, 已超出 OUT 的下限	BOOL	0	O
QOUT_LLL	1 = 激活临界报警, 已下冲 OUT 的下限	BOOL	0	O
QNONCAS0	1 = 正常 (非级联)	BOOL	0	O
QNONCAS1	1 = 更新事件	BOOL	0	O
QNONCAS2	1 = 激活警告 (优先级 <8)	BOOL	0	O
QNONCAS3	1 = 激活临界报警 (优先级 >8)	BOOL	0	O
QNONCAS4	1 = 未确认的更新事件	BOOL	0	O
QNONCAS5	1 = 未确认的警告	BOOL	0	O
QNONCAS6	1 = 未确认的临界报警	BOOL	0	O
QNONCAS7	1 = 启动故障安全	BOOL	0	O
QNONCAS8	1 = 需要维护	BOOL	0	O
QNONCAS9	1 = 功能测试/本地覆盖; 可用值	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O
QUALITY	过程变量的值状态	BYTE	0	O
QUNCERT	1 = 组消息 QUNCERTx	BOOL	0	O
QUNCERT0	1 = 未指定	BOOL	0	O
QUNCERT1	1 = 上一个可用值	BOOL	0	O
QUNCERT2	1 = 设置替代值	BOOL	0	O

4.15 FF_A_AI: 使用 PA 配置文件发送器

I/O (参数)	含义	数据类型	默认值	类型
QUNCERT3	1 = 初始值	BOOL	0	O
QUNCERT4	1 = 未确认的更新事件	BOOL	0	O
QUNCERT5	1 = 工程单位违例 (单位不在有效范围内)	BOOL	0	O
QUNCERT6	1 = 低于正常值	BOOL	0	O
QUNCERT7	1 = 组态错误	BOOL	0	O
QUNCERT8	1 = 传感器校准	BOOL	0	O
QUNCERT9	1 = 仿真值	BOOL	0	O
SIM_ON	1 = 激活仿真	BOOL	0	I
SIM_V	仿真值	REAL	0	I
STATUS	过程值状态	BYTE	0	O
SUBS_ON	1 = 启用替代	BOOL	0	I
SUBS_V	替代值	REAL	0	I
V	过程值	REAL	0	O
V_HL	输入值上限	REAL	0	I
V_LL	输入值下限	REAL	0	I
VALUE	过程映像输入值	REAL	0	I

4.16 FF_A_AO: 使用 PA 配置文件执行器

4.16.1 FF_A_AI 的描述

对象名 (类型 + 编号)

FB 411

- FF_A_AO 块 I/O (页 428)

应用领域

块 FF_A_AO 处理 (循环服务) FF 现场设备的 PA 配置文件“执行器”。

调用 OB

调用 OB 是在其中安装了该块并重启 OB 100 的循环中断 OB 3x (例如 OB 32)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- MODE 输入与 MOD_PAL0 块或 MOD_PAX0 块的 OMODE_00 输出互连。

功能和工作原理

该块从分区过程映像中读取用户数据以及向其中写入用户数据, 这取决于所选的用户数据组态 (输入参数 MODE_LW 的组态)。

必须根据 FF 现场设备的 MODE (页 725), 在输入变量 MODE_LW 中设置所设置用户数据组态的编码, 并指定要通过过程映像 (分区) 读写的变量。

该块将具有质量代码的设定值 (SP) (按照 PA 配置文件的模拟输出组态设定值和过程值, REAL 具有 1 个字节质量代码) 循环写入过程映像 (分区) 中。PA 配置文件包含设定值及其它作为物理单位的模拟值。质量代码包含有关设定值状态的信息。在 PROFIBUS 3.0“常规要求”中介绍了质量代码的编码。可在同一周期内将参考变量 (RCAS_IN) 随质量代码一起传送到过程映像 (分区) 中。

ASSET 中的链接

可在 ASSET 中链接块 FF_A_AO。

有关详细信息，请参考以下部分：

在 ASSET 中集成 FF 设备 (页 727)

质量代码

程序将生成结果值的质量代码（QUALITY 输出），可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误，上一个有效值	16#14
较高级别错误，替代值	16#18
不良，由过程引起	16#28
不确定，由设备引起	16#68
不确定，由过程引起	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
范围下限过冲	16#55
范围上限过冲	16#56
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC_POS_D_W、QC_RCAS_OUT_W、QC_READBACK_W）形成。

直接来自该设备的质量代码可以是 16#00 – 16#FF 之间的值。分析 FF 现场设备的质量代码 16#84-16#87 和 16#90 – 16#93 类似于分析质量代码 16#80 – 16#83。如果从 FF 设备传送质量代码 16#80（有效值）和一个位模式为 16#7FFFFFFF（无效值）的关联过程值（输入 I_RCAS_OUT 或 I_READBACK），则该 FF 现场设备将如同对质量代码 16#00（无效值）一样对它们进行处理。

从过程映像（分区）中循环读取 FF 现场设备的数据以及过程变量 (READBACK) 和离散位置反馈 (READBACK)。另外，还可读取激活的参考变量 (RCAS_OUT) 和详细的设备信息 (CHECKBACK)。可在块输出中通过相应的位获得设备信息。从过程映像（分区）中读取数据。为提高可互连性，重要的详细信息由读取的状态字节 (ST_READBACK) 或 ST_RCAS_OUT（如果 READBACK 不可用）在输出接口处以布尔 (BOOL) 值形式提供。这些都符合 PROFIBUS 3.0“常规要求”中指定的位组合。

如果出现较高级别错误 (QMOD_ERR = TRUE)，则继续向过程映像（分区）中写入数据，但不会从其中读取任何数据。只要激活了优先级较高的错误，就会以 QBAD = TRUE 保持上一个值。

说明

参考变量 (ST_RCAS_IN) 的状态字节预定义为零。仅当将质量代码设置为 16#80 时，参考变量才会在 FF 现场设备中变为激活状态。

组态 MODE_LW 输入

除了 MODE 输入外，还必须根据所用的参数来组态 MODE_LW（模式低位字）输入：

使用的参数	MODE_LW
SP	16#0100
SP、READBACK、POS_D	16#0103
SP、CHECK_BACK	16#0104
SP、READBACK、POS_D、CHECK_BACK	16#0105
RCAS_IN、RCAS_OUT	16#0206
RCAS_IN、RCAS_OUT、CHECK_BACK	16#0207
SP、RCAS_IN、READBACK、RCAS_OUT、POS_D、CHECK_BACK	16#0308

寻址

必须根据用户数据组态，将该块使用的每个连接与在 HW Config 中或在图标表中组态的图标互连：

I/O	数据类型
I_READBACK	REAL
I_RCAS_OUT	REAL
I_POS_D_W	WORD
O_SP	REAL
O_RCAS_IN	REAL

另外，必须将所用 I/O 的质量代码与在 HW Config 中组态的图标互连：

I/O	数据类型
CHECK_0_W	WORD
CHECK_1_W	WORD
CHECK_2_W	WORD
QC_READBACK_W	WORD
QC_RCAS_OUT_W	WORD
QC_POS_D_W	WORD
QC_SP_W	WORD
QC_RCAS_IN_W	WORD

仿真

如果输入参数 SIM_ON = TRUE，则输出输入 SIM_SP（以及选项 SIM_RCAS_IN）的值，质量代码（QUALITY）= 16#60。仿真拥有最高优先级。QBAD 设置为 FALSE。如果块处于仿真状态，则设置 QSIM = TRUE。

值限制

可限制过程映像（分区）中过程变量的最低和最高值的范围。LIMIT_ON = TRUE 时，过程变量“V”（READBACK 和 RCAS_OUT）的限制如下：

- 如果 $V > V_{HL}$ ，则为 V_{HL} 。
- 如果 $V < V_{LL}$ ，则为 LL_V 。

错误处理

不检查输入参数的似然性。

启动特征

该块将在系统启动时在 OB 100 中从前到后运行一次。将计算输出和输入/输出参数。

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.16.2 FF_A_AO 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
CHECK_0_W	现场设备附加信息	WORD	0	I
CHECK_1_W	现场设备附加信息	WORD	0	I
CHECK_2_W	现场设备附加信息	WORD	0	I
I_POS_D_W	阀的当前位置 (离散) (符号)	WORD	0	I
I_RCAS_OUT	功能块设定值	REAL	0	I
I_READBACK	运行期间, 最终控制元件在 PV 单元中的当前位置 (介于 OPEN 和 CLOSE 位置之间) (图标)	REAL	0	I
LIMIT_ON	1 = 启用限制输入值	BOOL	0	I
LL_V	输入值上限。V < V_LL 时的设置值	REAL	0	I
MODE	值状态	DWORD	0	I
MODE_LW	模式	WORD	0	I
O_RCAS_IN	监视主机提供给模拟控制或输出块的过程映像目标设定值 (符号)	REAL	0	O
O_SP	设定值 (符号)	REAL	0	O
POS_D	阀的当前位置 (离散)。POS 字节的编码如下: 0 = 未初始化, 1 = 关闭, 2 = 打开, 3 = 中间位置	BYTE	0	O
QA_1	1 = 报警/警告 1	BOOL	0	O
QA_2	1 = 报警/警告 2	BOOL	0	O
QBAD	1 = 组事件 QBAD_X	BOOL	0	O
QBAD_0	1 = 未指定	BOOL	0	O
QBAD_1	1 = 组态错误	BOOL	0	O
QBAD_2	1 = 未连接	BOOL	0	O
QBAD_3	1 = 设备故障	BOOL	0	O
QBAD_4	1 = 传感器故障	BOOL	0	O

I/O (参数)	含义	数据类型	默认值	类型
QBAD_5	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_6	1 = 无通讯 (无可用值)	BOOL	0	O
QBAD_7	1 = 服务中断	BOOL	0	O
QCASCAD0	1 = 正常 (级联)	BOOL	0	O
QCASCAD1	1 = 初始化已确认	BOOL	0	O
QCASCAD2	1 = 初始化请求	BOOL	0	O
QCASCAD3	1 = 不需要	BOOL	0	O
QCASCAD4	1 = 保留	BOOL	0	O
QCASCAD5	1 = 不选择	BOOL	0	O
QCASCAD6	1 = 本地覆盖	BOOL	0	O
QCASCAD7	1 = 保留	BOOL	0	O
QCASCAD8	1 = 启动故障安全	BOOL	0	O
QCB_0	1 = 处于故障安全位置的现场设备已激活	BOOL	0	O
QCB_1	1 = 请求手动模式	BOOL	0	O
QCB_2	1 = 现场设备处于手动模式, LOCKED OUT 开关处于激活状态	BOOL	0	O
QCB_3	1 = 紧急控制激活	BOOL	0	O
QCB_4	1 = 实际位置反馈不同于预期位置	BOOL	0	O
QCB_5	1 = 已超出 OPEN 方向的扭矩限值	BOOL	0	O
QCB_6	1 = 已超出 CLOSE 方向的扭矩限制	BOOL	0	O
QCB_7	1 = 行程监视设备的状态; 如果为“是”(YES), 则表示已超出执行器的行程时间	BOOL	0	O
QCB_8	1 = 执行器正在打开	BOOL	0	O
QCB_9	1 = 执行器正在关闭	BOOL	0	O
QCB_10	1 = 因更改静态数据 (功能块和传感器块) 产生的中断	BOOL	0	O
QCB_11	1 = 过程值的仿真已启用	BOOL	0	O
QCB_12	未使用	BOOL	0	O
QCB_13	1 = 内部控制循环已中断	BOOL	0	O
QCB_14	1 = 定位器未激活 (OUT 状态 = BAD)	BOOL	0	O

4.16 FF_A_AO: 使用 PA 配置文件执行器

I/O (参数)	含义	数据类型	默认值	类型
QCB_15	1 = 设备正在进行自检	BOOL	0	O
QCB_16	1 = 超出阀行程限制	BOOL	0	O
QCB_17	1 = 激活附加输入 (例如, 用于诊断)	BOOL	0	O
QCONST	1 = 常量	BOOL	0	O
QC_POS_D_O	质量代码 POS 输出	BYTE	0	O
QC_POS_D_W	质量代码 POS	WORD	16#80	I
QC_RCAS_IN	质量代码 RCAS_IN	BYTE	0	O
QC_RCAS_IN_W	质量代码 RCAS_IN (符号)	WORD	0	O
QC_RCAS_OUT_W	质量代码功能块设定值	WORD	16#80	I
QC_READBACK_O	质量代码功能块设定值输出	BYTE	0	O
QC_READBACK_W	质量代码功能块设定值 (符号)	WORD	16#80	I
QC_SP	质量代码设定值	BYTE	0	O
QC_SP_W	质量代码设定值 (符号)	WORD	0	O
QERR	1 = 输出错误 (ENO 的取反值)	BOOL	1	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QNONCAS0	1 = 正常 (非级联)	BOOL	0	O
QNONCAS1	1 = 更新事件	BOOL	0	O
QNONCAS2	1 = 激活警告 (优先级 <8)	BOOL	0	O
QNONCAS3	1 = 激活临界报警 (优先级 >8)	BOOL	0	O
QNONCAS4	1 = 未确认的更新事件	BOOL	0	O
QNONCAS5	1 = 未确认的警告	BOOL	0	O
QNONCAS6	1 = 未确认的临界报警	BOOL	0	O
QNONCAS7	1 = 启动故障安全	BOOL	0	O
QNONCAS8	1 = 需要维护	BOOL	0	O
QNONCAS9	1 = 功能测试/本地覆盖; 可用值	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QUNCERT	1 = 组事件 QUNCERTx	BOOL	0	O
QUNCERT0	1 = 未指定	BOOL	0	O

I/O (参数)	含义	数据类型	默认值	类型
QUNCERT1	1 = 上一个可用值	BOOL	0	O
QUNCERT2	1 = 设置替代值	BOOL	0	O
QUNCERT3	1 = 初始值	BOOL	0	O
QUNCERT4	1 = 不精确的传感器转换	BOOL	0	O
QUNCERT5	1 = 工程单位违例 (单位不在有效范围内)	BOOL	0	O
QUNCERT6	1 = 低于正常值	BOOL	0	O
QUNCERT7	1 = 组态错误	BOOL	0	O
QUNCERT8	1 = 传感器校准	BOOL	0	O
QUNCERT9	1 = 仿真值	BOOL	0	O
RCAS_IN	监视主机提供给模拟控制或输出块的目标设定值	REAL	0	I
RCAS_OUT	功能块设定值	REAL	0	O
READBACK	最终控制元件在 PV 单元中行程范围内的当前位置 (介于 OPEN 和 CLOSE 位置之间)。	REAL	0	O
SIM_ON	1 = 仿真激活	BOOL	0	I
SIM_POS_D	阀的当前位置 (离散)	BYTE	0	I
SIM_RCAS_IN	仿真 RCAS_IN	REAL	0	I
SIM_READBACK	运行期间, 最终控制元件在 PV 单元中的当前位置仿真 (介于 OPEN 和 CLOSE 位置之间)。	REAL	0	I
SIM_SP	仿真设定值	REAL	0	I
SP	Setpoint	REAL	0	I
ST_POS_D	状态 POS	BYTE	0	O
ST_READBACK	功能块回读值状态	BYTE	0	O
ST_RCAS_IN	RCAS_IN 状态	BYTE	0	I
ST_RCAS_OUT	功能块设定值状态	BYTE	0	O
ST_SP	设定值状态	BYTE	16#80	I
V_HL	输入值上限	REAL	0	I
V_LL	输入值下限	REAL	0	I

4.17 FF_A_DI: 读取数字值

4.17.1 FF_A_DI 的描述

对象名 (类型 + 编号)

FB 412

- FF_A_DI 块 I/O (页 436)

应用领域

块 FF_A_DI 用于从 FF 设备读入 (循环服务) 数字值 (离散输入)。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 数字输入通道的质量代码符号与输入 QC_W 互连。
- MODE 输入与 MOD_PAL0 块或 MOD_PAX0 块的相应 OMODE_00 输出互连。

功能和工作原理

块 FF_A_DI 用于将 FF 现场设备与 SIMATIC PCS 7 库的各个块连接在一起。它还可以与其它 SIMATIC S7 块互连。

该块循环读取带有 FF 现场设备状态字节的过程值 (I_OUT_D_W) (结构符合 PA 配置文件的离散输入)。状态字节 (STATUS) 包含 FF 现场设备状态的相关信息。可以按位访问过程值和重要状态字节信息, 从而在输出接口上获得更好的可互连性。这些都符合 PROFIBUS“常规要求”中指定的位组合。

ASSET 中的链接

可在 ASSET 中链接块 FF_A_DI。

有关详细信息，请参考以下部分：

在 ASSET 中集成 FF 设备 (页 727)

质量代码

为结果值生成了质量代码（QUALITY 输出），质量代码可采用下列值：

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误，上一个有效值	16#14
较高级别错误，替代值	16#18
不良，由过程引起	16#28
不确定，由设备引起	16#68
不确定，由过程引起	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC-W）形成。

直接来自该设备的质量代码可以是 16#00 – 16#FF 之间的值。分析 FF 现场设备的质量代码 16#84-16#87 和 16#90 – 16#93 类似于分析质量代码 16#80 – 16#83。

该块可通过 MODE (页 725) 输入参数识别较高级别错误，例如，DP/PA 连接器的故障。

- 如果 MODE 高位字节 = 16#80，则过程映像（分区）中的值有效。
- 如果 MODE 高位字节 = 16#40（值状态 = 较高级别错误，QMOD_ERR = TRUE），则过程值将被视为无效。

不考虑输入参数 MODE_LW 的模式。

4.17 FF_A_DI: 读取数字值

寻址

在 HW Config 中, 必须为数字输入通道组态一个图标并将该图标与输入 I_OUT_D_W 相连。CFC 功能“生成模块驱动程序”使数字输入通道的质量代码图标与输入参数 QC_W 互连。

仿真

输入参数 SIM_ON = TRUE 时, 将输出包含质量代码 (QUALITY =) 16#60 的输入 SIM_I 的值。仿真拥有最高优先级。QBAD 始终设置为 FALSE。如果块处于仿真状态, 则设置 QSIM = TRUE。

替代值

如果输入参数 SUBS_ON = TRUE, 则在过程值无效时, 将输出输入参数 SUBS_I 的值来作为替代值。将质量代码 (QUALITY) 设置为 16#44 且 QBAD = 1。

输出无效值

如果输入参数 SUBS_ON = FALSE, 并且存在一个无效过程值, 则输出此无效值并将 QBAD 设置为 1。

错误处理

不检查输入参数的似然性。

启动特征

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.17.2 FF_A_DI 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
I_OUT_D_W	过程映像的过程值	WORD	0	I
MODE	值状态	DWORD	0	I
MODE_LW	模式	WORD	2	I
OUT_D	过程值	BYTE	0	O
QA_1	1 = 报警/警告 1	BOOL	0	O
QA_2	1 = 报警/警告 2	BOOL	0	O
QBAD	1 = 组事件 QBAD_X	BOOL	0	O
QBAD_0	1 = 未指定	BOOL	0	O
QBAD_1	1 = 组态错误	BOOL	0	O
QBAD_2	1 = 未连接	BOOL	0	O
QBAD_3	1 = 设备故障	BOOL	0	O
QBAD_4	1 = 传感器故障	BOOL	0	O
QBAD_5	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_6	1 = 无通讯 (无可值)	BOOL	0	O
QBAD_7	1 = 服务中断	BOOL	0	O
QC_W	过程映像的质量代码值	WORD	0	I
QCONST	1 = 常量	BOOL	0	O
QERR	1 = 输出错误 (ENO 的取反值)	BOOL	1	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QNONCAS0	1 = 正常 (非级联)	BOOL	0	O
QNONCAS1	1 = 更新事件	BOOL	0	O
QNONCAS4	1 = 未确认的更新事件	BOOL	0	O
QNONCAS7	1 = 启动故障安全	BOOL	0	O
QNONCAS8	1 = 需要维护	BOOL	0	O

I/O (参数)	含义	数据类型	默认值	类型
QNONCAS9	1 = 功能测试/本地覆盖; 可用值	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O
QUALITY	过程变量的值状态	BYTE	0	O
QUNCERT	1 = 组事件 QUNCERTx	BOOL	0	O
QUNCERT0	1 = 未指定	BOOL	0	O
QUNCERT1	1 = 上一个可用值	BOOL	0	O
QUNCERT2	1 = 设置替代值	BOOL	0	O
QUNCERT3	1 = 初始值	BOOL	0	O
QUNCERT4	1 = 不精确的传感器转换	BOOL	0	O
QUNCERT5	1 = 工程单位违例 (单位不在有效范围内)	BOOL	0	O
QUNCERT6	1 = 低于正常值	BOOL	0	O
QUNCERT7	1 = 组态错误	BOOL	0	O
QUNCERT8	1 = 传感器校准	BOOL	0	O
QUNCERT9	1 = 仿真值	BOOL	0	O
Q0	过程值位 0	BOOL	0	O
Q1 Q7	过程值位 1 ... 位 7	BOOL	0	O
SIM_I	仿真值	BYTE	0	I
SIM_ON	1 = 激活仿真	BOOL	0	I
STATUS	过程值状态	BYTE	0	O
SUBS_I	替代值	BYTE	0	I
SUBS_ON	1 = 替代激活	BOOL	0	I

4.18 FF_A_DO: 数字值输出

4.18.1 FF_A_DO 的描述

对象名 (类型 + 编号)

FB 413

- FF_A_DO 块 I/O (页 443)

应用领域

块 FF_A_DO (循环服务) 用于通过 FF 现场设备输出数字值 (SP 或 RCAS_IN, 最多 8 个)。

调用 OB

调用 OB 是在其中安装了该块并重启 OB 100 的循环中断 OB 3x (例如 OB 32)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- MODE 输入与 MOD_PAL0 块或 MOD_PAX0 块的 OMODE_00 输出互连。

功能和工作原理

块 FF_A_DO 从 (分区) 过程映像中读取用户数据以及向其中写入用户数据, 这取决于所选的用户数据组态 (输入 MODE_LW 的组态)。

对于 FF 设备, 必须根据其 MODE_LW 设置 (页 725), 在输入变量 MODE_LW 中设置已设定用户数据组态的编码。这指定了在过程映像 (分区) 中要读取和写入的变量。

该块将设定值 (O_SP_W) 与质量代码 (QC_SP_W) 一起写入到过程映像 (分区) 中。质量代码包含有关设定值状态的信息。在 PROFIBUS 3.0“常规要求”中介绍了质量代码的编码。可以选择在同一周期内将 RCAS (远程级联) 状态 (RCAS_IN) 中的设定值与质量代码一起传送到过程映像 (分区) 中。

ASSET 中的链接

可在 ASSET 中链接块 FF_A_DO。

有关详细信息，请参考以下部分：

在 ASSET 中集成 FF 设备 (页 727)

质量代码

程序将生成结果值的质量代码（QUALITY 输出），可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误，上一个有效值	16#14
较高级别错误，替代值	16#18
不良，由过程引起	16#28
不确定，由设备引起	16#68
不确定，由过程引起	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC_READBACK_W、QC_RCAS_OUT_W）形成。

直接来自该设备的质量代码可以是 16#00 – 16#FF 之间的值。分析 FF 现场设备的质量代码 16#84-16#87 和 16#90 – 16#93 类似于分析质量代码 16#80 – 16#83。可以选择从过程映像（分区）中循环读取 FF 现场设备的数据、阀的状态 (READBACK)、状态为 RCAS 时阀设置的过程值 (RCAS_OUT) 以及详细的设备信息 (CHECKBACK)。可在块输出中通过相应的位获得设备信息。

为了提高可互连性，在输出接口处以布尔 (BOOL) 值的形式提供了来自读取状态字节的重要详细信息。这些都符合 PROFIBUS“常规要求”中指定的位组合。如果 READBACK 和 RCAS_OUT 同时存在，则从 READBACK 状态字节获得详细信息。

4.18 FF_A_DO: 数字值输出

如果出现较高级别错误 (QMOD_ERR = TRUE), 则继续向过程映像 (分区) 中写入数据, 但不会从其中读取任何数据。只要激活了优先级较高的错误, 就会以 QBAD = TRUE 保持上一个值。

说明

设定值状态字节 (ST_SP) 的默认值和参考变量 (ST_RCAS_IN) 的默认值均为零。仅当将相应的状态字节设置为 16#80, 设定值和参考变量才会在 PROFIBUS PA 现场设备中变为激活状态。

组态 MODE_LW 输入

除了 MODE 输入外, 还必须根据所用的参数来组态 MODE_LW (模式低位字) 输入:

使用的参数	MODE_LW
SP_D	16#0400
SP_D、READBACK_D	16#0409
SP_D、CHECKBACK_D	16#040A
SP_D、READBACK_D、CHECK_BACK_D	16#040B
RCAS_IN_D、RCAS_OUT_D	16#050C
RCAS_IN_D、RCAS_OUT_D、CHECK_BACK_D	16#050D
SP_D、RCAS_IN_D、READBACK_D、RCAS_OUT_D、CHECK_BACK_D	16#060E

寻址

必须将块所用的每个连接与在 HW Config 中组态的图标互连:

I/O	数据类型
I_READBACK_W	WORD
I_RCAS_OUT_W	WORD
O_SP_W	WORD
O_RCAS_IN_W	WORD

另外, 必须将所用 I/O 的质量代码与在 HW Config 中组态的图标互连:

I/O	数据类型
CHECK_0_W	WORD
CHECK_1_W	WORD
CHECK_2_W	WORD
QC_READBACK_W	WORD
QC_RCAS_OUT_W	WORD
QC_SP_W	WORD
QC_RCAS_IN_W	WORD

仿真

如果输入参数 SIM_ON = TRUE, 则输入参数 SIM_SP (以及 SIM_RCAS_IN 选项, 如果选定) 的值与质量代码 (QUALITY) = 16#60 一起输出。仿真拥有最高优先级。QBAD 设置为 FALSE。如果块处于仿真状态, 则设置 QSIM = TRUE。

错误处理

不检查输入参数的似然性。

启动特征

该块将在系统启动时在 OB 100 中从前到后运行一次。将计算输出和输入/输出参数。

系列: DRIVER

4.18 FF_A_DO: 数字值输出

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

4.18.2 FF_A_DO 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
CHECK_0_W	现场设备附加信息	WORD	0	I
CHECK_1_W	现场设备附加信息	WORD	0	I
CHECK_2_W	现场设备附加信息	WORD	0	I
I_RCAS_OUT_W	功能块设定值	WORD	0	I
I_READBACK_W	过程值 (READBACK)。 (符号)	WORD	0	I
MODE	值状态	DWORD	0	I
MODE_LW	模式	WORD	0	I
O_RCAS_IN_W	监视主机提供给模拟控制或输出块的目标设定值 (符号)	WORD	0	O
O_SP_W	设定值 (符号)	WORD	0	O
QA_1	1 = 报警/警告 1	BOOL	0	O
QA_2	1 = 报警/警告 2	BOOL	0	O
QBAD	1 = 组事件 QBAD_X	BOOL	0	O
QBAD_0	1 = 未指定	BOOL	0	O
QBAD_1	1 = 组态错误	BOOL	0	O
QBAD_2	1 = 未连接	BOOL	0	O
QBAD_3	1 = 设备故障	BOOL	0	O
QBAD_4	1 = 传感器故障	BOOL	0	O
QBAD_5	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_6	1 = 无通讯 (无可用值)	BOOL	0	O
QBAD_7	1 = 服务中断	BOOL	0	O
QCB_0	1 = 故障安全位置的现场设备	BOOL	0	O
QCB_1	1 = 请求以手动模式操作设备	BOOL	0	O
QCB_2	1 = 现场设备处于手动模式	BOOL	0	O

4.18 FF_A_DO: 数字值输出

I/O (参数)	含义	数据类型	默认值	类型
QCB_3	1 = 紧急控制激活	BOOL	0	O
QCB_4	1 = 执行器离开已达到的最终位置	BOOL	0	O
QCB_5	1 = 阀连接断开	BOOL	0	O
QCB_6	1 = 指示阀连接短路	BOOL	0	O
QCB_7	未使用	BOOL	0	O
QCB_8	1 = 执行器正在打开	BOOL	0	O
QCB_9	1 = 执行器正在关闭	BOOL	0	O
QCB_10	1 = 因更改 FB 和 TB 的静态数据产生的报警	BOOL	0	O
QCB_11	1 = 过程值的仿真已启用	BOOL	0	O
QCB_12	未使用	BOOL	0	O
QCB_13	1 = 内部控制循环已中断	BOOL	0	O
QCB_14	1 = 阀未激活 (状态 OUT_D 不良)	BOOL	0	O
QCB_15	1 = 设备正在进行自检	BOOL	0	O
QCB_16	1 = 超出阀行程限制	BOOL	0	O
QCB_17	1 = 从 OPEN 切换为 CLOSE 时, 超出中断时间	BOOL	0	O
QCB_18	1 = 从 CLOSE 切换为 OPEN 时, 超出中断时间	BOOL	0	O
QCB_19	1 = 内部循环测试中出错	BOOL	0	O
QCB_20	1 = 从 OPEN 转移到 CLOSE 期间超时	BOOL	0	O
QCB_21	1 = 从 CLOSE 转移到 OPEN 期间超时	BOOL	0	O
QCB_22	1 = 阀机械阻塞	BOOL	0	O
QCB_23	未使用	BOOL	0	O
QCONST	1 = 常量	BOOL	1	O
QC_RCAS_IN	质量代码目标设定值	BYTE	0	O
QC_RCAS_IN_W	质量代码目标设定值 (符号)	WORD	0	O
QC_RCAS_OUT_W	质量代码功能块设定值 (符号)	WORD	0	I
QC_READBACK_O	质量代码功能块设定值输出	BYTE	0	O
QC_READBACK_W	质量代码功能块设定值 (符号)	WORD	0	I
QC_SP	质量代码设定值	BYTE	0	O
QC_SP_W	质量代码设定值 (符号)	WORD	0	O

I/O (参数)	含义	数据类型	默认值	类型
QERR	1 = 输出错误 (ENO 的取反值)	BOOL	1	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QNONCAS0	1 = 正常 (非级联)	BOOL	0	O
QNONCAS1	1 = 更新事件	BOOL	0	O
QNONCAS4	1 = 未确认的更新事件	BOOL	0	O
QNONCAS7	1 = 启动故障安全	BOOL	0	O
QNONCAS8	1 = 需要维护	BOOL	0	O
QNONCAS9	1 = 功能测试/本地覆盖; 可用值	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QUNCERT	1 = 组事件 QUNCERTx	BOOL	0	O
QUNCERT0	1 = 未指定	BOOL	0	O
QUNCERT1	1 = 上一个可用值	BOOL	0	O
QUNCERT2	1 = 设置替代值	BOOL	0	O
QUNCERT3	1 = 初始值	BOOL	0	O
QUNCERT4	1 = 不精确的传感器转换	BOOL	0	O
QUNCERT5	1 = 工程单位违例 (单位不在有效范围内)	BOOL	0	O
QUNCERT6	1 = 低于正常值	BOOL	0	O
QUNCERT7	1 = 组态错误	BOOL	0	O
QUNCERT8	1 = 传感器校准	BOOL	0	O
QUNCERT9	1 = 仿真值	BOOL	0	O
RCAS_IN	目标设定值	BYTE	0	I
RCAS_OUT	功能块设定值	BYTE	0	O
READBACK	过程值 (READBACK)	BYTE	0	O
SIM_ON	1 = 激活仿真	BOOL	0	I
SIM_RCAS_IN	仿真目标设定值	BYTE	0	I
SIM_READBACK	运行期间, 最终控制元件在 PV 单元中的当前位置仿真 (介于 OPEN 和 CLOSE 位置之间)。	BYTE	0	I
SIM_SP	仿真值	BYTE	0	I
SP	设定值	BYTE	0	I

系列: DRIVER

4.18 FF_A_DO: 数字值输出

I/O (参数)	含义	数据类型	默认值	类型
ST_READBACK	功能块设定值状态	BYTE	0	O
ST_RCAS_IN	RCAS_IN 状态	BYTE	0	I
ST_RCAS_OUT	功能块设定值状态	BYTE	0	O
ST_SP	设定值状态	BYTE	0	I

4.19 MSG_TS: 生成具有时间戳的过程值

4.19.1 MSG_TS 的描述

对象名 (类型 + 编号)

FB131

- MSG_TS 块 I/O (页 450)

功能

MSG_TS 消息块用于生成具有时间戳的过程消息。该块构成 IMDRV_TS 块输出和 ALARM_8P 块输出之间的接口, 其 8 条消息的时间戳输入到第一个关联值的字节数组中。

工作原理

该块具有输入 (VALUE_xx), “生成模块驱动程序”功能使用其信息将输入结构/参数 TS_xx 和 TS_C_xx 与 IMDRV_TS 的输出结构/参数 TS_xxx 和 TS_C_xxx 互连。具有时间戳 (TS_xx) 的各个过程报警的信息被分配给两个 ALARM_8P 块相应的消息号 (EV_IDxx) 和通道号 (1 到 16)。OR_32_TS 总是互连在 MSG_TS 和 IMDRV_TS 之间。

具有 EV_ID_01 的 ALARM_8P:

- SIG_1 = TS_00.MSG_SIG
- SIG_2 = TS_01.MSG_SIG
- ...
- SIG_8 = TS_07.MSG_SIG

TS_00.TS 到 TS_07.TS 的时间戳在第一个关联值 SD_1 的字节数组中输入。

4.19 MSG_TS: 生成具有时间戳的过程值

具有 EV_ID_02 的 ALARM_8P:

- SIG_1 = TS_08.MSG_SIG
- SIG_2 = TS_09.MSG_SIG
-
- SIG_8 = TS_15.MSG_SIG

TS_08.TS 到 TS_15.TS 的时间戳在第一个关联值 SD_1 的字节数组中输入。

在分配了所有消息后，该块在 OB 1 中调用 ALARM_8P 块，并向 OS 传送新消息。在该块和接口模块 (IM, Interface Module) 之间进行数据通讯期间可能出现的错误（例如，I/O 访问错误）将通过 ALARM_8P 块在 IMDRV_TS 中报告。来自 ALARM_8P 块的反馈（STAT_xx、M_ACK_xx）可在块输出中提供。如果 STAT_xx = 11（先前的作业未完成），该块将在下一个周期中再次调用 ALARM_8P。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x（例如 OB 32）和 OB1。

在 CFC 中的使用

通过 HW Config 为数字输入通道生成的符号（符号表）必须与 VALUE 输入互连。

根据 HW Config 中用来确定哪些通道将具有时间戳的设置以及 VALUE_xx 输入中的地址信息，“生成模块驱动程序”功能将 I/O 与驱动程序块 IMDRV_TS 的相关 I/O 互连。

OR_32_TS 总是互连在 MSG_TS 和 IMDRV_TS 之间。

消息响应

该块使用两个 ALARM_8P 块，这两个块被作为多个实例进行调用。每次 ALARM_8P 调用的 8 个时间戳值通过第 1 个关联值的字节数组 [0..65] 进行传送。数组的结构如下：

字节 0	下列日期/时间戳的格式标识符 始终为 0
字节 1	下列日期/时间戳的格式标识符 始终为 1 (ISP 格式)
字节 2 - 9:	signal_1 的日期/时间戳
字节 10 - 17:	signal_2 的日期/时间戳
.	.
.	.
字节 58 - 65:	signal_8 的日期/时间戳

字节 0 - 1 的格式标识符用于指定时间戳结构的位编码（一个时间戳值分配有 8 个字节）。始终支持 ISP 格式。

采用 ISP 格式的时间戳

符合 ISP 约定的完整时间（从 1900-1-1; 00:00 h 算起的时间）。由于秒占 4 个字节，因此可用秒表示从 1900-1-1; 0:00 h 起经过的时间。

字节	内容	区域
0 - 3	自 1900-1-1 0:00.00,000 开始的秒数	对应于 1900-1-1 ... 2036-2-6
4 - 7	以 $1/2 \wedge 32$ s 的倍数表示的秒的小数部分	0 .. <1

时间转换在这几种时间格式涵盖的日期范围内有效，也就是说从 1900-1-1 直到（并包括）2036-2-6（2月6日）。

该驱动程序块将以 ISP 格式输出由 IM 提供的时间戳，并且不会进行任何更改。

更多信息

更多相关信息，请参考以下部分：

MSG_TS 的消息文本 (页 451)

4.19 MSG_TS: 生成具有时间戳的过程值

4.19.2 MSG_TS 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
EV_ID_01	消息 ID 1	DWORD	16#20	I
EV_ID_02	消息 ID 2	DWORD	16#21	I
M_ACK_01	消息确认 1	WORD	0	O
M_ACK_02	消息确认 2	WORD	0	O
Q_ERR_01	消息错误 1	BOOL	0	O
Q_ERR_02	消息错误 2	BOOL	0	O
QBAD	1 = 无效过程值	BOOL	0	O
QERR	1 = 处理错误	BOOL	1	O
STAT_01	状态输出 1	WORD	0	O
STAT_02	状态输出 2	WORD	0	O
TS_xx	时间戳 (xx = 00 – 15) 字节 0: 位 0: 消息信号状态 (MsgSig) 位 1: 沿变化信息 (TriInf) 位 2: 握手 (Hd) 字节 1: 时间戳的质量代码 (ST) DWORD TS0: ISP 格式的日期/时间戳 (秒) DWORD TS1: ISP 格式的日期/时间戳 (秒的小数部分)	STRUCT		I
TS_C_xx	TS 通讯 (xx = 00 - 15) 位 0: 传送确认 (HS) 位 1: 互连检查 (LI)	BYTE	0	IO
VALUE_xx	输入值 (xx = 00 – 15)	BOOL	0	I

4.19.3 MSG_TS 的消息文本

消息文本和消息类别的分配

更多相关信息，可参考消息类别

消息块	消息号	默认消息文本	消息类别
EV_ID_01 (ALARM_8P)			
	1	TEXT S_CH_00	S
	2	TEXT S_CH_01	S
	3	TEXT S_CH_02	S
	4	TEXT S_CH_03	S
	5	TEXT S_CH_04	S
	6	TEXT S_CH_05	S
	7	TEXT S_CH_06	S
EV_ID_02 (ALARM_8P)	8	TEXT S_CH_07	S
	1	TEXT S_CH_08	S
	2	TEXT S_CH_09	S
	3	TEXT S_CH_10	S
	4	TEXT S_CH_11	S
	5	TEXT S_CH_12	S
	6	TEXT S_CH_13	S
7	TEXT S_CH_14	S	
8	TEXT S_CH_15	S	

4.20 PA_AI: PROFIBUS PA 模拟值输入

4.20.1 PA_AI 的描述

对象名 (类型 + 编号)

FB101

- PA_AI 块 I/O (页 456)

应用领域

PA_AI 块用于处理 (循环服务) PROFIBUS 3.0 A 类和 B 类 PA 现场设备的“发送器”PA 配置文件或 HART 现场设备的主要或辅助变量。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 将模拟输入通道的质量代码图标与输入 QC 互连。
- 将 MODE 输入与 PADP_L0x 块相应的输出 OMODE_xx 互连。

功能和工作原理

PA_AI 块从过程映像 (分区) 中循环读取 PROFIBUS 或 HART 现场设备的过程值和状态字节 (质量代码) (结构符合 PA 配置文件的模拟输入)。过程值用作物理单位。状态字节 (STATUS) 包含 PROFIBUS 现场设备状态的相关信息。

质量代码

程序将生成结果值的质量代码 (QUALITY 输出), 可能为以下状态:

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误, 上一个有效值	16#14
较高级别错误, 替代值	16#18
不良, 过程相关	16#28
不确定, 设备相关	16#68
不确定, 过程相关	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件 (例如通道错误、较高级别错误或仿真) 以及直接来自该设备的质量代码 (参数 QC) 形成的。

根据 PROFIBUS 要求, 直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

除了提供状态字节及提高可互连性, 输出接口还提供了更多的布尔 (BOOL) 值, 它们包含重要的详细信息。它们符合 PROFIBUS PA 3.0“常规要求”中的位组合规范。设备特定的控制系统消息基于 QC 通过 ALARM_8P 生成。

该块可通过 MODE 输入参数识别较高级别错误, 例如, DP/PA 连接器的故障。

- 如果 MODE 高位字节 = 16#80, 则过程映像 (分区) 中的值有效。
- 如果 MODE 高位字节 = 16#40 (值状态 = 较高级别错误), 则模拟值将被视为无效。

在 MODE 输入参数低位字中设置的运行模式将被忽略。

寻址

生成模拟输入通道的符号，并通过 HW Config 将其输入到符号表中。必须将其与 VALUE 输入互连。CFC 功能“生成模块驱动程序”将模拟输入通道的质量代码图标与输入参数 QC 互连。

仿真

输入参数 SIM_ON = TRUE 时，将输出输入参数 SIM_V 的值及质量代码 (QUALITY =) 16#60。仿真拥有最高优先级。QBAD 始终设置为 FALSE。如果该块处于仿真状态，则会设置 QSIM = TRUE。

说明

请记住，仿真值总是在仿真模式下输出，与任一参数 LAST_ON（替代值）或 SUBS_ON（上一个有效值）都无关。

替代值

如果输入参数 SUBS_ON = TRUE，则在值无效时，将输入参数 SUBS_V 的值将作为输出值。将质量代码 (QUALITY) 设置为 16#44 且 QBAD = 1。

保持上一个值

如果输入参数 LAST_ON = TRUE，则在过程值无效时，输出上一个有效输出值。将质量代码 (QUALITY) 设置为 16#44 且 QBAD = 1。

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE，或均为 TRUE，并且存在一个无效过程值，则会输出此无效值并将 QBAD 设置为 1。

值限制

可限制过程映像（分区）中过程值的最低和最高范围。

如果开关 LIMIT_ON = TRUE，则按如下所示限制过程值 (V):

- 如果 $V > V_{HL}$ ，则为 V_{HL}
- 如果 $V < V_{LL}$ ，则为 LL_V

错误处理

不检查输入参数的似然性。

启动特征

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.20.2 PA_AI 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
LAST_ON	1 = 上一个有效值: 启用注入	BOOL	0	IO
LIMIT_ON	1 = 启用限制输入值	BOOL	0	I
LL_V	在 $V < V_{LL}$ 时的值	REAL	0	I
MODE	值状态和模式	DWORD	0	I
QA_1	1 = 报警/警告 1	BOOL	0	O
QA_2	1 = 报警/警告 2	BOOL	0	O
QBAD	1 = 组事件 QBAD_X	BOOL	0	O
QBAD_0	1 = 未指定	BOOL	0	O
QBAD_1	1 = 组态错误	BOOL	0	O
QBAD_2	1 = 未连接	BOOL	0	O
QBAD_3	1 = 设备故障	BOOL	0	O
QBAD_4	1 = 传感器故障	BOOL	0	O
QBAD_5	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_6	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_7	1 = 服务中断	BOOL	0	O
QBAD_HL	1 = 已达到传感器物理范围的上限	BOOL	0	O
QBAD_LL	1 = 已达到传感器物理范围的下限	BOOL	0	O
QC	输入过程值的状态	BYTE	0	I
QCASCAD0	1 = 正常 (级联)	BOOL	0	O
QCASCAD1	1 = 初始化已确认	BOOL	0	O
QCASCAD2	1 = 初始化请求	BOOL	0	O
QCASCAD3	1 = 不需要	BOOL	0	O
QCASCAD4	1 = 保留	BOOL	0	O
QCASCAD5	1 = 不选择	BOOL	0	O

I/O (参数)	含义	数据类型	默认值	类型
QCASCAD6	1 = 本地覆盖	BOOL	0	O
QCASCAD7	1 = 保留	BOOL	0	O
QCASCAD8	1 = 启动故障安全	BOOL	0	O
QCONST	1 = 常量	BOOL	0	O
QERR	1 = 输出错误 (ENO 的取反值)	BOOL	1	O
QLAST	1 = 上一个有效值: 注入激活	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QOUT_HHL	1 = 激活临界报警, 已超出 OUT 的上限	BOOL	0	O
QOUT_HL	1 = 激活警告, 已超出 OUT 的上限	BOOL	0	O
QOUT_LL	1 = 激活警告, 已下冲 OUT 的下限	BOOL	0	O
QOUT_LLL	1 = 激活临界报警, 已下冲 OUT 的下限	BOOL	0	O
QNONCAS0	1 = 正常 (非级联)	BOOL	0	O
QNONCAS1	1 = 更新事件	BOOL	0	O
QNONCAS2	1 = 激活警告 (优先级 <8)	BOOL	0	O
QNONCAS3	1 = 激活临界报警 (优先级 >8)	BOOL	0	O
QNONCAS4	1 = 未确认的更新事件	BOOL	0	O
QNONCAS5	1 = 未确认的警告	BOOL	0	O
QNONCAS6	1 = 未确认的临界报警	BOOL	0	O
QNONCAS7	1 = 启动故障安全	BOOL	0	O
QNONCAS8	1 = 需要维护	BOOL	0	O
QNONCAS9	1 = 功能测试/本地覆盖; 可用值	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O
QUALITY	过程值的质量代码	BYTE	0	O
QUNCERT	1 = 组消息 QUNCERTx	BOOL	0	O
QUNCERT0	1 = 未指定	BOOL	0	O
QUNCERT1	1 = 上一个可用值	BOOL	0	O
QUNCERT2	1 = 设置替代值	BOOL	0	O
QUNCERT3	1 = 初始值	BOOL	0	O

4.20 PA_AI: PROFIBUS PA 模拟值输入

I/O (参数)	含义	数据类型	默认值	类型
QUNCERT4	1 = 未确认的更新事件	BOOL	0	O
QUNCERT5	1 = 工程单位违例 (单位不在有效范围内)	BOOL	0	O
QUNCERT6	1 = 低于正常值	BOOL	0	O
QUNCERT7	1 = 组态错误	BOOL	0	O
QUNCERT8	1 = 传感器校准	BOOL	0	O
QUNCERT9	1 = 仿真值	BOOL	0	O
SIM_ON	1 = 激活仿真	BOOL	0	I
SIM_V	仿真值	REAL	0	I
STATUS	过程值状态	BYTE	0	O
SUBS_ON	1 = 启用替代	BOOL	0	I
SUBS_V	替代值	REAL	0	I
V	过程值	REAL	0	O
V_HL	输入值上限	REAL	0	I
V_LL	输入值下限	REAL	0	I
VALUE	过程映像输入值	REAL	0	I

4.21 PA_AO: PROFIBUS PA 模拟值输出

4.21.1 PA_AO 的描述

对象名 (类型 + 编号)

FB 103

- PA_AO 块 I/O (页 463)

应用领域

块 PA_AO 用于处理 (循环服务) PROFIBUS PA 3.0 A 类和 B 类 PA 现场设备的“执行器”PA 配置文件。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32) 和重启 OB 100。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 模拟输出通道 O_SP 的质量代码图标与输出 QC_SP 和其它选定选项互连。
- 将 MODE 输入与 PADP_L0x 块相应的输出 OMODE_xx 互连。

功能和工作原理

该块用于从过程映像 (分区) 中读取用户数据以及向其中写入用户数据, 这取决于按照 PROFIBUS PA 3.0 对“模拟输出”PA 配置文件中用户数据组态的选择 (使用 HW Config 或 SIMATIC PDM)。

输入变量 MODE 的低位字包含在 PROFIBUS PA 3.0“模拟输出”配置文件中设置的用户数据组态的编码。这指定了要读取和写入的过程映像 (分区) 中的变量。

该块将具有质量代码的设定值 (SP) (按照 PA 配置文件的模拟输出组态设定值和过程值, REAL 具有 1 个字节质量代码) 循环写入过程映像 (分区) 中。PA 配置文件包含设定值及其它作为物理单位的模拟值。质量代码包含有关设定值状态的信息。在 PROFIBUS 3.0“常规要求”中介绍了质量代码的编码。可在同一周期内将参考变量 (RCAS_IN) 随质量代码一起传送到过程映像 (分区) 中。

质量代码

程序将生成结果值的质量代码（QUALITY 输出），可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误，上一个有效值	16#14
较高级别错误，替代值	16#18
不良，过程相关	16#28
不确定，设备相关	16#68
不确定，由过程引起	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
范围下限过冲	16#55
范围上限过冲	16#56
提出维护请求	16#A4
无效值	16#00

质量代码是由内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC_X，其中 X 是参数名）形成。

按照 PROFIBUS 的要求，直接从该设备返回的 QC 可能是 00 - FF 范围内的值。

从过程映像（分区）中循环读取 PROFIBUS PA 现场设备的数据，以及过程值 (READBACK) 和离散位置反馈 (POS_D)。另外，还可读取激活的参考变量 (RCAS_OUT) 和详细的设备信息 (CHECKBACK)。在块输出中按位提供了设备信息。从过程映像（分区）中读取数据。为提高可互连性，重要的详细信息由读取的状态字节 (ST_READBACK) 或 ST_RCAS_OUT（如果 READBACK 不可用）在输出接口处以布尔 (BOOL) 值形式提供。这些都符合 PROFIBUS 3.0“常规要求”中指定的位组合。

如果出现较高级别错误 (QMOD_ERR = TRUE), 则继续向过程映像 (分区) 中写入数据, 但不会从其中读取任何数据。只要激活了较高级别错误, 就会通过 QBAD = TRUE 保持上一个值。

说明

状态字节 (ST_RCAS_IN) 预定义为零。仅当将质量代码设置为 16#80 时, 参考变量才会在 PROFIBUS 现场设备中变为激活状态。

寻址

必须将使用 HW Config 为模拟输出通道 (PROFIBUS PA 3.0 配置文件“模拟输出”) 组态的图标之一 (例如 SP) 与对应的 I/O 互连:

I/O	数据类型
I_READBACK	REAL
I_RCAS_OUT	REAL
I_POS_D	BYTE
O_SP	REAL
O_RCAS_IN	REAL

在 CFC 中, “生成模块驱动程序”功能会自动将 I/O 的相应质量代码图标与模拟输出通道 (带质量代码) 的其余已组态图标互连。

仿真

如果输入参数 SIM_ON = TRUE, 则输出输入 SIM_SP (以及选项 SIM_RCAS_IN) 的值, 质量代码 (QUALITY) = 16#60。仿真拥有最高优先级。QBAD 设置为 FALSE。如果该块处于仿真状态, 则会设置 QSIM = TRUE。

值限制

可限制过程映像 (分区) 中过程值的最低和最高范围。如果 LIMIT_ON = TRUE, 则对过程值“V” (READBACK 和 RCAS_OUT) 将有如下限制:

- 如果 $V > V_{HL}$, 则为 V_{HL} 。
- 如果 $V < V_{LL}$, 则为 LL_V 。

错误处理

不检查输入参数的似然性。

启动特征

该块将在系统启动时在 OB 100 中从前到后运行一次。将计算输出和输入/输出参数。

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.21.2 PA_AO 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考以下部分:

"关于块描述的常规信息 (页 17)"。

I/O (参数)	含义	数据类型	默认值	类型
CHECK_0	现场设备附加信息	BYTE	0	I
CHECK_1	现场设备附加信息	BYTE	0	I
CHECK_2	现场设备附加信息	BYTE	0	I
I_POS_D	阀的当前位置 (离散) (符号)	BYTE	0	I
I_RCAS_OUT	功能块设定值	REAL	0	I
I_READBACK	运行期间, 最终控制元件在 PV 单元中的当前位置 (介于 OPEN 和 CLOSE 位置之间) (符号)	REAL	0	I
LIMIT_ON	1 = 启用限制输入值	BOOL	0	I
LL_V	输入值上限。 V < V_LL 时的设置值	REAL		I
MODE	MODE 输入参数	DWORD	0	I
O_RCAS_IN	监视主机提供给模拟控制或输出块的目标设定值 (符号)	REAL	0	O
O_SP	设定值 (符号)	REAL	0	O
POS_D	阀的当前位置 (离散)。POS 字节的编码如下: 0 = 未初始化, 1 = 关闭, 2 = 打开, 3 = 中间位置	BYTE	0	O
QA_1	1 = 报警/警告 1	BOOL	0	O
QA_2	1 = 报警/警告 2	BOOL	0	O
QBAD	1 = 组事件 QBAD_X	BOOL	0	O
QBAD_0	1 = 未指定	BOOL	0	O
QBAD_1	1 = 组态错误	BOOL	0	O
QBAD_2	1 = 未连接	BOOL	0	O
QBAD_3	1 = 设备故障	BOOL	0	O
QBAD_4	1 = 传感器故障	BOOL	0	O

I/O (参数)	含义	数据类型	默认值	类型
QBAD_5	1 = 无通讯 (上一个可用值)	BOOL	0	0
QBAD_6	1 = 无通讯 (无可用值)	BOOL	0	0
QBAD_7	1 = 服务中断	BOOL	0	0
QCASCAD0	1 = 正常 (级联)	BOOL	0	0
QCASCAD1	1 = 初始化已确认	BOOL	0	0
QCASCAD2	1 = 初始化请求	BOOL	0	0
QCASCAD3	1 = 不需要	BOOL	0	0
QCASCAD4	1 = 保留	BOOL	0	0
QCASCAD5	1 = 不选择	BOOL	0	0
QCASCAD6	1 = 本地覆盖	BOOL	0	0
QCASCAD7	1 = 保留	BOOL	0	0
QCASCAD8	1 = 启动故障安全	BOOL	0	0
QNONCAS9	1 = 功能测试/本地覆盖; 可用值	BOOL	0	0
QCB_0	1 = 处于故障安全位置的现场设备已激活	BOOL	0	0
QCB_1	1 = 请求手动模式	BOOL	0	0
QCB_2	1 = 现场设备处于手动模式, LOCKED OUT 开关处于激活状态	BOOL	0	0
QCB_3	1 = 紧急控制激活	BOOL	0	0
QCB_4	1 = 实际位置反馈不同于预期位置	BOOL	0	0
QCB_5	1 = 已超出 OPEN 方向的扭矩限值	BOOL	0	0
QCB_6	1 = 已超出 CLOSE 方向的扭矩限值	BOOL	0	0
QCB_7	1 = 行程监视设备的状态; 如果为“是”(YES), 则表示已超出执行器的行程时间	BOOL	0	0
QCB_8	1 = 执行器正在打开	BOOL	0	0
QCB_9	1 = 执行器正在关闭	BOOL	0	0
QCB_10	1 = 因更改静态数据 (功能块和传感器块) 产生的中断	BOOL	0	0
QCB_11	1 = 过程值的仿真已启用	BOOL	0	0
QCB_12	未使用	BOOL	0	0
QCB_13	1 = 内部控制循环已中断	BOOL	0	0

I/O (参数)	含义	数据类型	默认值	类型
QCB_14	1 = 定位器未激活 (OUT 状态 = BAD)	BOOL	0	O
QCB_15	1 = 设备正在进行自检	BOOL	0	O
QCB_15	1 = 设备正在进行自检 (MODE = 服务中断)	BOOL	0	O
QCB_16	1 = 已超出阀行程限制	BOOL	0	O
QCB_17	1 = 激活附加输入 (例如, 用于诊断)	BOOL	0	O
QCONST	1 = 常量	BOOL	0	O
QC_POS_D	质量代码 POS	BYTE	0	I
QC_POS_D_O	质量代码 POS 输出	BYTE	0	O
QC_RCAS_IN	质量代码 RCAS_IN (符号)	BYTE	0	O
QC_RCAS_OUT	质量代码功能块设定值	BYTE	0	I
QC_READBACK	质量代码功能块设定值 (符号)	BYTE	0	I
QC_READBACK_O	质量代码功能块设定值输出	BYTE	0	O
QC_SP	质量代码设定值 (符号)	BYTE	0	O
QERR	1 = 输出错误 (ENO 的取反值)	BOOL	1	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QNONCAS0	1 = 正常 (非级联)	BOOL	0	O
QNONCAS1	1 = 更新事件	BOOL	0	O
QNONCAS2	1 = 激活警告 (优先级 <8)	BOOL	0	O
QNONCAS3	1 = 激活临界报警 (优先级 >8)	BOOL	0	O
QNONCAS4	1 = 未确认的更新事件	BOOL	0	O
QNONCAS5	1 = 未确认的警告	BOOL	0	O
QNONCAS6	1 = 未确认的临界报警	BOOL	0	O
QNONCAS7	1 = 启动故障安全	BOOL	0	O
QNONCAS8	1 = 需要维护	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QUNCERT	1 = 组事件 QUNCERTx	BOOL	0	O
QUNCERT0	1 = 未指定	BOOL	0	O
QUNCERT1	1 = 上一个可用值	BOOL	0	O

4.21 PA_AO: PROFIBUS PA 模拟值输出

I/O (参数)	含义	数据类型	默认值	类型
QUNCERT2	1 = 设置替代值	BOOL	0	O
QUNCERT3	1 = 初始值	BOOL	0	O
QUNCERT4	1 = 不精确的传感器转换	BOOL	0	O
QUNCERT5	1 = 工程单位违例 (单位不在有效范围内)	BOOL	0	O
QUNCERT6	1 = 低于正常值	BOOL	0	O
QUNCERT7	1 = 组态错误	BOOL	0	O
QUNCERT8	1 = 传感器校准	BOOL	0	O
QUNCERT9	1 = 仿真值	BOOL	0	O
RCAS_IN	监视主机提供给模拟控制或输出块的目标设定值	REAL	0	I
RCAS_OUT	功能块设定值	REAL	0	O
READBACK	最终控制元件在 PV 单元中行程范围内的当前位置 (介于 OPEN 和 CLOSE 位置之间)。	REAL	0	O
SIM_ON	1 = 仿真激活	BOOL	0	I
SIM_POS_D	阀的当前位置 (离散)	BYTE	0	I
SIM_RCAS_IN	仿真 RCAS_IN	REAL	0	I
SIM_READBACK	运行期间, 最终控制元件在 PV 单元中的当前位置仿真 (介于 OPEN 和 CLOSE 位置之间)。	REAL	0	I
SIM_SP	仿真设定值	REAL	0	I
SP	设定值	REAL	0	I
ST_POS_D	状态 POS	BYTE	0	O
ST_READBACK	功能块回读值状态	BYTE	0	O
ST_RCAS_IN	RCAS_IN 状态	BYTE	0	I
ST_RCAS_OUT	功能块设定值状态	BYTE	0	O
ST_SP	设定值状态	BYTE	0	I
V_HL	输入值上限	REAL	0	I
V_LL	输入值下限	REAL	0	I

4.22 PA_DI: PROFIBUS PA 数字值输入

4.22.1 PA_DI 的描述

对象名 (类型 + 编号)

FB104

- PA_DI 块 I/O (页 471)

应用领域

PA_DI 块 (循环服务) 用于通过 PROFIBUS A 类和 B 类现场设备读入数字值 (离散输入)。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 数字输入通道的质量代码图标与输入 QC 互连。
- 将 MODE 输入与 PADP_L0x 块相应的输出 OMODE_xx 互连。

功能和工作原理

PA_DI 块作为 PROFIBUS 现场设备与 SIMATIC PCS 7 库的各个块之间的接口。它还可以与其它 SIMATIC S7 块互连。该块需要 PROFIBUS DP 接口。该接口可以集成在 CPU 中, 也可以是外部 DP 接口 (CP)。PROFIBUS PD 到 PROFIBUS PA 的转换是通过 SIMATIC 段耦合器 DP/PA 连接器 (或 DP/PA 耦合器) 或者符合标准 (EN 50170, 卷 2、EN 61158-2、IEC 1158-2) 的类似段耦合器实现的。

该块循环读取 PROFIBUS PA 现场设备的过程值 (OUT_D) 和状态字节 (结构符合 PA 配置文件的离散输入)。状态字节 (STATUS) 包含 PROFIBUS 现场设备状态的相关信息。过程值和状态字节 (2 个字节) 将直接读取并始终作为一个“字”读取。可以按位访问过程值和重要状态字节信息, 从而在输出接口上获得更好的可互连性。这些都符合 PROFIBUS“常规要求”中指定的位组合。

质量代码

程序将生成结果值的质量代码（QUALITY 输出），可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误，上一个有效值	16#14
较高级别错误，替代值	16#18
不良，过程相关	16#28
不确定，设备相关	16#68
不确定，过程相关	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

该块可通过 MODE 输入参数识别较高级别错误，例如，DP/PA 连接器的故障。

- 如果 MODE 高位字节 = 16#80，则过程映像（分区）中的值有效。
- 如果 MODE 高位字节 = 16#40（值状态 = 较高级别错误，QMOD_ERR = TRUE），则过程值将被视为无效。

在 MODE 输入参数低位字中设置的运行模式将被忽略。

寻址

符号表中用 HW Config 为数字输入通道生成的符号必须与 I_OUT_D 输入互连。CFC 功能“生成模块驱动程序”将数字输入通道的质量代码图标与输入参数 QC 互连。

说明

请记住，仿真值总是在仿真模式下输出，与任一参数 LAST_ON（替代值）或 SUBS_ON（上一个有效值）都无关。

仿真

输入参数 SIM_ON = TRUE 时，将输出包含质量代码 (QUALITY =) 16#60 的输入 SIM_I 的值。仿真拥有最高优先级。QBAD 始终设置为 FALSE。如果该块处于仿真状态，则会设置 QSIM = TRUE。

替代值

如果输入参数 SUBS_ON = TRUE，则在过程值无效时，将输出输入参数 SUBS_I 的值来作为替代值。将质量代码 (QUALITY) 设置为 16#44 且 QBAD = 1。

保持上一个值

如果输入参数 LAST_ON = TRUE，过程值无效时将输出上一个有效输出值，并设置 QBAD = 1 且设置质量代码 QUALITY = 16#44。

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE，或均为 TRUE，并且存在一个无效过程值，则会输出此无效值并将 QBAD 设置为 1。

错误处理

不检查输入参数的似然性。

启动特征

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.22.2 PA_DI 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
I_OUT_D	过程映像的过程值	BYTE	0	I
LAST_ON	1 = 上一个有效值: 启用注入	BOOL	0	IO
MODE	MODE 输入参数	DWORD	0	I
OUT_D	过程值	BYTE	0	O
QA_1	1 = 报警/警告 1	BOOL	0	O
QA_2	1 = 报警/警告 2	BOOL	0	O
QBAD	1 = 组事件 QBAD_X	BOOL	0	O
QBAD_0	1 = 未指定	BOOL	0	O
QBAD_1	1 = 组态错误	BOOL	0	O
QBAD_2	1 = 未连接	BOOL	0	O
QBAD_3	1 = 设备故障	BOOL	0	O
QBAD_4	1 = 传感器故障	BOOL	0	O
QBAD_5	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_6	1 = 无通讯 (无可值)	BOOL	0	O
QBAD_7	1 = 服务中断	BOOL	0	O
QC	过程映像的质量代码值	BYTE	0	I
QCONST	1 = 常量	BOOL	0	O
QERR	1 = 输出错误 (ENO 的取反)	BOOL	1	O
QLAST	1 = 上一个有效值: 注入激活	BOOL	0	O
QMOD_ERR	1 = 较高级别错误	BOOL	0	O
QNONCAS0	1 = 正常 (非级联)	BOOL	0	O
QNONCAS1	1 = 更新事件	BOOL	0	O
QNONCAS4	1 = 未确认的更新事件	BOOL	0	O
QNONCAS7	1 = 启动故障安全	BOOL	0	O

I/O (参数)	含义	数据类型	默认值	类型
QNONCAS8	1 = 需要维护	BOOL	0	O
QNONCAS9	1 = 功能测试/本地覆盖; 可用值	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O
QUALITY	过程值的质量代码	BYTE	0	O
QUNCERT	1 = 组事件 QUNCERTx	BOOL	0	O
QUNCERT0	1 = 未指定	BOOL	0	O
QUNCERT1	1 = 上一个可用值	BOOL	0	O
QUNCERT2	1 = 设置替代值	BOOL	0	O
QUNCERT3	1 = 初始值	BOOL	0	O
QUNCERT4	1 = 不精确的传感器转换	BOOL	0	O
QUNCERT5	1 = 工程单位违例 (单位不在有效范围内)	BOOL	0	O
QUNCERT6	1 = 低于正常值	BOOL	0	O
QUNCERT7	1 = 组态错误	BOOL	0	O
QUNCERT8	1 = 传感器校准	BOOL	0	O
QUNCERT9	1 = 仿真值	BOOL	0	O
Q0	过程值位 0	BOOL	0	O
Q1 Q7	过程值位 1 ... 位 7	BOOL	0	O
SIM_I	仿真值	BYTE	0	I
SIM_ON	1 = 激活仿真	BOOL	0	I
STATUS	过程值状态	BYTE	0	O
SUBS_I	替代值	BYTE	0	I
SUBS_ON	1 = 替代激活	BOOL	0	I

4.23 PA_DO: PROFIBUS PA 数字值输出

4.23.1 PA_DO 的描述

对象名 (类型 + 编号)

FB 105

- PA_DO 块 I/O (页 477)

应用领域

块 PA_DO (循环服务) 用于将数字值 (SP/RCAS_IN, 最多 8 个) 输出到 A 类和 B 类 PROFIBUS 现场设备。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32) 和重启 OB 100。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 数字输出通道 O_SP 的质量代码图标与输出 QC_SP 和其它选定选项互连。
- 将 MODE 输入与 PADP_L0x 块相应的输出 OMODE_xx 互连。

功能和工作原理

块 PA_DO 用于从过程映像 (分区) 中读取用户数据以及向其中写入用户数据, 这取决于符合 PROFIBUS 3.0 标准的 PA 配置文件“数字输出”中用户数据组态的选择 (使用 HW Config 或 SIMATIC PDM)。

输入变量 MODE 的低位字包含在 PROFIBUS 3.0 配置文件“离散输出”中所设置的用户数据组态的编码。这指定了在过程映像 (分区) 中要读取和写入的变量。

该块将具有质量代码的设定值 (SP) 写入过程映像 (分区) 中 (设定值和过程值的结构对应于 PA 配置文件的数字输出, 1 个字节具有 1 个字节质量代码)。质量代码包含有关设定值状态的信息。在 PROFIBUS 3.0“常规要求”中介绍了质量代码的编码。可以选择在同一周期内将 RCAS (远程级联) 状态 (RCAS_IN) 中的设定值与质量代码一起传送到过程映像 (分区) 中。

质量代码

程序将生成结果值的质量代码（QUALITY 输出），可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误，上一个有效值	16#14
较高级别错误，替代值	16#18
不良，过程相关	16#28
不确定，设备相关	16#68
不确定，过程相关	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是由内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC_X，其中 X 是参数名）形成。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

可选择从过程映像（分区）中循环读取 PROFIBUS 现场设备的数据（READBACK 阀的状态）、状态 RCAS 时阀设置的过程值 (RCAS_OUT) 以及详细的设备信息 (CHECKBACK)。在块输出中按位提供了设备信息。

为了提高可互连性，在输出接口处以布尔 (BOOL) 值的形式提供了来自读取状态字节的重要详细信息。这些都符合 PROFIBUS“常规要求”中指定的位组合。如果 READBACK 和 RCAS_OUT 同时存在，则从 ST_READBACK 状态字节获得详细信息。

如果出现较高级别错误 (QMOD_ERR = TRUE), 则继续向过程映像 (分区) 中写入数据, 但不会从其中读取任何数据。只要激活了较高级别错误, 就会通过 QBAD = TRUE 保持上一个值。

说明

设定值状态字节 (ST_SP) 的默认值和参考变量 (ST_RCAS_IN) 的默认值均为零。只有当将相应的状态字节设置为 16#80 时, 设定值和参考变量才会在 PROFIBUS 现场设备中变为激活状态。

寻址

必须将使用 HW Config 为数字输出通道 (PROFIBUS 3.0 配置文件“模拟输出”) 组态的图标之一 (例如 SP) 与对应的 I/O 互连:

I/O	数据类型
I_READBACK	BYTE
I_RCAS_OUT	BYTE
O_SP	BYTE
O_RCAS_IN	BYTE

在 CFC 中, “生成模块驱动程序”功能会自动将 I/O 的相应质量代码 (如果存在) 图标与数字输出通道 (带质量代码) 的其余已组态图标互连。

仿真

如果输入参数 SIM_ON = TRUE, 则输入参数 SIM_SP (以及 SIM_RCAS_IN 选项, 如果选定) 的值与质量代码 (QUALITY) = 16#60 一起输出。仿真拥有最高优先级。QBAD 设置为 FALSE。如果该块处于仿真状态, 则会设置 QSIM = TRUE。

错误处理

不检查输入参数的似然性。

启动特征

该块将在系统启动时在 OB 100 中从前到后运行一次。将计算输出和输入/输出参数。

4.23 PA_DO: PROFIBUS PA 数字值输出

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息，请参考以下部分：

关于使用驱动程序块的注意事项 (页 329)

4.23.2 PA_DO 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
CHECK_0	现场设备附加信息	BYTE	0	I
CHECK_1	现场设备附加信息	BYTE	0	I
CHECK_2	现场设备附加信息	BYTE	0	I
I_RCAS_OUT	功能块设定值	BYTE	0	I
I_READBACK	过程值 (READBACK)。 (符号)	BYTE	0	I
MODE	MODE 输入参数	DWORD	0	I
O_RCAS_IN	监视主机提供给模拟控制或输出块的目标设定值 (符号)	BYTE	0	O
O_SP	设定值 (符号)	BYTE	0	OI
QA_1	1 = 报警/警告 1	BOOL	0	O
QA_2	1 = 报警/警告 2	BOOL	0	O
QBAD	1 = 组事件 QBAD_X	BOOL	0	O
QBAD_0	1 = 未指定	BOOL	0	O
QBAD_1	1 = 组态错误	BOOL	0	O
QBAD_2	1 = 未连接	BOOL	0	O
QBAD_3	1 = 设备故障	BOOL	0	O
QBAD_4	1 = 传感器故障	BOOL	0	O
QBAD_5	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_6	1 = 无通讯 (无可值)	BOOL	0	O
QBAD_7	1 = 服务中断	BOOL	0	O
QCONST	1 = 常量	BOOL	0	O
QCB_0	1 = 故障安全位置的现场设备	BOOL	0	O
QCB_1	1 = 请求以手动模式操作设备	BOOL	0	O
QCB_2	1 = 现场设备处于手动模式	BOOL	0	O

4.23 PA_DO: PROFIBUS PA 数字值输出

I/O (参数)	含义	数据类型	默认值	类型
QCB_3	1 = 紧急控制激活	BOOL	0	0
QCB_4	1 = 执行器离开已达到的最终位置	BOOL	0	0
QCB_5	1 = 阀连接断开	BOOL	0	0
QCB_6	1 = 指示阀连接短路	BOOL	0	0
QCB_7	未使用	BOOL	0	0
QCB_8	1 = 执行器正在打开	BOOL	0	0
QCB_9	1 = 执行器正在关闭	BOOL	0	0
QCB_10	1 = 因更改 FB 和 TB 的静态数据产生的报警	BOOL	0	0
QCB_11	1 = 过程值的仿真已启用	BOOL	0	0
QCB_12	未使用	BOOL	0	0
QCB_13	1 = 内部控制循环已中断	BOOL	0	0
QCB_14	1 = 阀未激活 (状态 OUT_D 不良)	BOOL	0	0
QCB_15	1 = 设备正在进行自检	BOOL	0	0
QCB_16	1 = 已超出阀行程限制	BOOL	0	0
QCB_17	1 = 从 OPEN 切换为 CLOSE 时, 超出中断时间	BOOL	0	0
QCB_18	1 = 从 CLOSE 切换为 OPEN 时, 超出中断时间	BOOL	0	0
QCB_19	1 = 内部循环测试中出错	BOOL	0	0
QCB_20	1 = 从 OPEN 转移到 CLOSE 期间超时	BOOL	0	0
QCB_21	1 = 从 CLOSE 转移到 OPEN 期间超时	BOOL	0	0
QCB_22	1 = 阀机械阻塞	BOOL	0	0
QCB_23	未使用	BOOL	0	0
QC_RCAS_IN	质量代码目标设定值 (符号)	BYTE	0	0
QC_RCAS_OUT	质量代码功能块设定值 (符号)	BYTE	0	1
QC_READBACK	质量代码功能块设定值 (符号)	BYTE	0	1
QC_READBACK_O	质量代码功能块 设定值输出	BYTE	0	0
QC_SP	质量代码设定值 (符号)	BYTE	0	0
QERR	1 = 输出错误 (ENO 的取反)	BOOL	1	0
QMOD_ERR	1 = 较高级别错误	BOOL	0	0

I/O (参数)	含义	数据类型	默认值	类型
QNONCAS0	1 = 正常 (非级联)	BOOL	0	O
QNONCAS1	1 = 更新事件	BOOL	0	O
QNONCAS4	1 = 未确认的更新事件	BOOL	0	O
QNONCAS7	1 = 启动故障安全	BOOL	0	O
QNONCAS8	1 = 需要维护	BOOL	0	O
QNONCAS9	1 = 功能测试/本地覆盖; 可用值	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O
QUNCERT	1 = 组事件 QUNCERTx	BOOL	0	O
QUNCERT0	1 = 未指定	BOOL	0	O
QUNCERT1	1 = 上一个可用值	BOOL	0	O
QUNCERT2	1 = 设置替代值	BOOL	0	O
QUNCERT3	1 = 初始值	BOOL	0	O
QUNCERT4	1 = 不精确的传感器转换	BOOL	0	O
QUNCERT5	1 = 工程单位违例 (单位不在有效范围内)	BOOL	0	O
QUNCERT6	1 = 低于正常值	BOOL	0	O
QUNCERT7	1 = 组态错误	BOOL	0	O
QUNCERT8	1 = 传感器校准	BOOL	0	O
QUNCERT9	1 = 仿真值	BOOL	0	O
RCAS_IN	目标设定值	BYTE	0	I
RCAS_OUT	功能块设定值	BYTE	0	O
READBACK	过程值 (READBACK)	BYTE	0	O
SIM_ON	1 = 激活仿真	BOOL	0	I
SIM_RCAS_IN	仿真目标设定值	BYTE	0	I
SIM_READBACK	运行期间, 最终控制元件在 PV 单元中的当前位置仿真 (介于 OPEN 和 CLOSE 位置之间)	REAL	0	I
SIM_SP	仿真值	BYTE	0	I
SP	设定值	BYTE	0	I
ST_READBACK	功能块设定值状态	BYTE	0	O

系列: DRIVER

4.23 PA_DO: PROFIBUS PA 数字值输出

I/O (参数)	含义	数据类型	默认值	类型
ST_RCAS_IN	RCAS_IN 状态	BYTE	0	I
ST_RCAS_OUT	功能块设定值状态	BYTE	0	O
ST_SP	设定值状态	BYTE	0	I
V_HL	输入值上限	REAL	0	I
V_LL	输入值下限	REAL	0	I

4.24 PA_TOT: PROFIBUS PA 总和器

4.24.1 PA_TOT 的描述

对象名 (类型 + 编号)

FB 102

- PA_TOT 块 I/O (页 485)

应用领域

块 PA_TOT 用于处理符合 PROFIBUS 3.0 A 类和 B 类标准的 PA 现场设备的“总和器”PA 配置文件的循环参数。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 模拟输入通道的质量代码符号与输入 QC 和其它选定选项互连。
- 将 MODE 输入与 PADP_L0x 块相应的输出 OMODE_xx 互连。

功能和工作原理

块 PA_TOT 从过程映像（分区）中循环读取 PROFIBUS 现场设备的具有状态字节（质量代码）的过程值 (TOTAL)（结构符合 PA 配置文件的总和器）。过程值用作物理单位。状态字节 (STATUS) 包含 PROFIBUS 现场设备状态的相关信息。

除了提供状态字节及提高可互连性，输出接口还提供了更多的布尔 (BOOL) 值，它们包含重要的详细信息。这些都符合 PROFIBUS 3.0“常规要求”中指定的位组合。

也可选择将输入变量 SET_TOT 和 MODE_TOT 写入过程映像（分区）中。

该块通过 MODE 输入参数检测较高级别错误，例如，DP/PA 连接器的故障。

- 如果 MODE 高位字节 = 16#80，则过程映像（分区）中的值有效。
- 如果 MODE 高位字节 = 16#40（值状态 = 较高级别错误，QMOD_ERR = TRUE），则模拟值将被视为无效。

质量代码

程序将生成结果值的质量代码（QUALITY 输出），可能为以下状态：

状态	质量代码
有效值	16#80
仿真	16#60
较高级别错误，上一个有效值	16#14
较高级别错误，替代值	16#18
不良，过程相关	16#28
不确定，设备相关	16#68
不确定，过程相关	16#78
上一个有效值	16#44
替代值	16#48
范围过冲	16#54
提出维护请求	16#A4
无效值	16#00

质量代码是通过内部事件（例如通道错误、较高级别错误或仿真）以及直接来自该设备的质量代码（参数 QC）形成。

根据 PROFIBUS 要求，直接源于设备的质量代码可采用 16#00 – 16#FF 范围的值。

寻址

符号表中通过 HW Config 为模拟输入通道生成的符号必须与 TOTAL 输入参数互连。CFC 功能“生成模块驱动程序”将模拟输入通道的质量代码符号与输入参数 QC 以及输出参数 O_SET_TOT 和 O_MODE_TOT（如果它们存在）互连。

仿真

如果输入参数 SIM_ON = TRUE，则输入参数 SIM_V 的值与质量代码 (QUALITY =) 16#60 一起输出。仿真拥有最高优先级。QBAD 始终设置为 FALSE。如果该块处于仿真状态，则会设置 QSIM = TRUE。

替代值

如果输入参数 SUBS_ON = TRUE，则在过程值无效时，将输入参数 SUBS_V 的值作为值输出。将质量代码 (QUALITY) 设置为 16#44 且 QBAD = 1。

保持上一个值

如果输入参数 LAST_ON = TRUE，则在过程值无效时，输出上一个有效输出值。将质量代码 (QUALITY) 设置为 16#44 且 QBAD = 1。

输出无效值

如果输入参数 SUBS_ON 和 LAST_ON 两者均为 FALSE，或均为 TRUE，并且存在一个无效过程值，则会输出此无效值并将 QBAD 设置为 1。

值限制

可限制过程映像（分区）中过程值的最低和最高范围。

如果开关 LIMIT_ON = TRUE，则按如下所示限制过程值 (V):

- 如果 $V > V_{HL}$ ，则为 V_{HL}
- 如果 $V < V_{LL}$ ，则为 LL_V

错误处理

不检查输入参数的似然性。

启动特征

不可用

时间响应

不可用

消息响应

不可用

操作和监视

该块没有面板。

更多信息

更多相关信息, 请参考以下部分:

关于使用驱动程序块的注意事项 (页 329)

4.24.2 PA_TOT 的 I/O

CFC 中块显示的工厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示 I/O 可见; 正常字体的 I/O 名称则表示 I/O 不可见。

有关所使用缩写的说明和信息, 可参考

“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
LAST_ON	1 = 上一个有效值: 启用注入	BOOL	0	IO
LIMIT_ON	1 = 启用限制输入值	BOOL	0	I
LL_V	在 $V < V_{LL}$ 时的输入值	REAL	0	I
MODE	MODE 输入参数	DWORD	0	I
MODE_TOT	总和器模式: 0 = 平衡、 1 = 仅为正、2 = 仅为负、3 = 保持	BYTE	0	I
O_MODE_TOT	总和器模式	BYTE	0	O
O_SET_TOT	总和器算法	BYTE	0	O
QA_1	1 = 报警/警告 1	BOOL	0	O
QA_2	1 = 报警/警告 2	BOOL	0	O
QBAD	1 = 组事件 QBAD_X	BOOL	0	O
QBAD_0	1 = 未指定	BOOL	0	O
QBAD_1	1 = 组态错误	BOOL	0	O
QBAD_2	1 = 未连接	BOOL	0	O
QBAD_3	1 = 设备故障	BOOL	0	O
QBAD_4	1 = 传感器故障	BOOL	0	O
QBAD_5	1 = 无通讯 (上一个可用值)	BOOL	0	O
QBAD_6	1 = 无通讯 (无可值)	BOOL	0	O
QBAD_7	1 = 服务中断	BOOL	0	O
QBAD_HL	1 = 已达到传感器的物理上限	BOOL	0	O
QBAD_LL	1 = 已达到传感器的物理下限	BOOL	0	O
QC	输入过程值质量代码	BYTE	0	I

4.24 PA_TOT: PROFIBUS PA 总和器

I/O (参数)	含义	数据类型	默认值	类型
QCASCAD0	1 = 正常 (级联)	BOOL	0	0
QCASCAD1	1 = 初始化已确认	BOOL	0	0
QCASCAD2	1 = 初始化请求	BOOL	0	0
QCASCAD3	1 = 不需要	BOOL	0	0
QCASCAD4	1 = 保留	BOOL	0	0
QCASCAD5	1 = 不选择	BOOL	0	0
QCASCAD6	1 = 本地覆盖	BOOL	0	0
QCASCAD7	1 = 保留	BOOL	0	0
QCASCAD8	1 = 启动故障安全	BOOL	0	0
QCONST	1 = 常量	BOOL	0	0
QERR	1 = 输出错误 (ENO 的取反值)	BOOL	1	0
QLAST	1 = 上一个有效值: 注入激活	BOOL	0	0
QMOD_ERR	1 = 较高级别错误	BOOL	0	0
QNONCAS0	1 = 正常 (非级联)	BOOL	0	0
QNONCAS1	1 = 更新事件	BOOL	0	0
QNONCAS2	1 = 激活警告 (优先级 <8)	BOOL	0	0
QNONCAS3	1 = 激活临界报警 (优先级 >8)	BOOL	0	0
QNONCAS4	1 = 未确认的更新事件	BOOL	0	0
QNONCAS5	1 = 未确认的警告	BOOL	0	0
QNONCAS6	1 = 未确认的临界报警	BOOL	0	0
QNONCAS7	1 = 启动故障安全	BOOL	0	0
QNONCAS8	1 = 需要维护	BOOL	0	0
QNONCAS9	1 = 功能测试/本地覆盖; 可用值	BOOL	0	0
QOUT_HHL	1 = 激活临界报警, 已超出 OUT 的上限	BOOL	0	0
QOUT_HL	1 = 激活警告, 已超出 OUT 的上限	BOOL	0	0
QOUT_LL	1 = 激活警告, 已下冲 OUT 的下限	BOOL	0	0

I/O (参数)	含义	数据类型	默认值	类型
QOUT_LLL	1 = 激活临界报警, 已下冲 OUT 的下限	BOOL	0	O
QSIM	1 = 仿真激活	BOOL	0	O
QSUBS	1 = 替代激活	BOOL	0	O
QUALITY	过程值的质量代码	BYTE	0	O
QUNCERT	1 = 组事件 QUNCERTx	BOOL	0	O
QUNCERT0	1 = 未指定	BOOL	0	O
QUNCERT1	1 = 上一个可用值	BOOL	0	O
QUNCERT2	1 = 设置替代值	BOOL	0	O
QUNCERT3	1 = 初始值	BOOL	0	O
QUNCERT4	1 = 不精确的传感器转换	BOOL	0	O
QUNCERT5	1 = 工程单位违例 (单位不在有效范围内)	BOOL	0	O
QUNCERT6	1 = 低于正常值	BOOL	0	O
QUNCERT7	1 = 组态错误	BOOL	0	O
QUNCERT8	1 = 传感器校准	BOOL	0	O
QUNCERT9	1 = 仿真值	BOOL	0	O
SET_TOT	算法: 0 = 总和器, 1 = 复位 0, 2 = 预设 PRESET_TOT	BYTE	0	I
SIM_ON	1 = 激活仿真	BOOL	0	I
SIM_V	仿真值	REAL	0	I
STATUS	过程值状态	BYTE	0	O
SUBS_ON	1 = 启用替代	BOOL	0	I
SUBS_V	替代值	REAL	0	I
TOTAL	输入值	REAL	0	I
V	过程值	REAL	0	O
V_HL	输入值上限	REAL	0	I
V_LL	输入值下限	REAL	0	I

4.25 RCV_341: 在串行模式下通过 CP 341 接收数据

4.25.1 RCV_341 的描述

对象名 (类型 + 编号)

FB 121

- RCV_341 块 I/O (页 493)

应用领域

块 RCV_341 用于通过 CP 341 模块接收串行数据。

调用 OB

OB 100 和用于接收数据的循环 OB (建议 100 ms)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 组态 RACK_NO、SUBN1_ID、SUBN2_ID 和 SLOT_NO 输入。
- MODE 输入与 MOD_CP 块的 OMODE_00 输出互连。

创建用于接收串行数据的用户块的说明

要求: 必须安装用于组态点对点连接 (CP PtP Param) 的可选数据包。

在 HW Config 中, 也可以设置如下的传输类型 (协议):

- DK 3964R
- RK 512
- ASCII
- MODBUS 主站
- MODBUS 从站

步骤

- 在 CFC 图表中插入 RCV_341 块。
- 在 LADDR 块输入中设置 CP 341 模块的逻辑基址。
- 为块中的用户数据定义输入缓冲器（以下称为 RCV_DATA）。
- 在同一循环 OB 的 RCV_341 块的下游安装 RCV_DATA。输入缓冲器的定义可以是一个简单变量或一个变量数组。变量定义允许除 ANY 外的所有 S7 数据类型。
- 将输入缓冲器应用到 RCV_DATA 块的输出。
- 将此输出与 RCV_341 块的 R_DATA 输入互连。
- 为了对接收到的数据进行控制和评估，需在 RCV_DATA 块上定义如下 I/O，并将其与 RCV_341 块的相应 I/O 互连：

I/O	数据类型	含义
输入:		
NDR	BOOL	新数据已接收
ERROR	BOOL	接收新数据时出错
STATUS	WORD	错误状态
LEN	INT	接收数据的字节长度
输出:		
EN_R	BOOL	启用数据接收
R	BOOL	重置数据接收

EN_R = TRUE 时，块 RCV_341 准备接收数据。如果 NDR = TRUE，新数据将存储在 与 R_DATA 互连的 RCV_DATA 块的数据区域中。变量 LEN 指示所接收数据的长度。必须将新接收到的数据保存在块中或设置 EN_R = FALSE，因为所有数据可能会在下一个周期中被覆盖。

如果变量 ERROR = TRUE，会在 STATUS 中输入错误事件编号。（有关这些事件编号的含义，可参考 CP 341 手册中的说明）。应根据所选的传输类型对事件类别 8 进行评估，以确定如何处理所接收到的出现错误的的数据。

对于错误情况 (ERROR = TRUE)，不应为 STATUS = 16#1E0D 设置复位 (R = TRUE)。在其它情况下，建议您在 一个周期内设置 R = TRUE。

4.25 RCV_341: 在串行模式下通过 CP 341 接收数据

- **程序 DK 3964R**

此程序不需要对 RCV_341 块的更多变量进行分配或评估。

更多相关信息，可参考 CP 341 手册。

- **远程耦合器 RK 512**

RCV_341 块的变量（隐藏输出）指示用户数据的来源。

I/O	数据类型	含义
L_TYP	CHAR	远程 CPU 的区域类型
L_NO	INT	远程 CPU 的 DB 号
L_OFFSET	INT	远程 CPU 的 DB 偏移量
L_CF_BYT	INT	通讯标记字节号，远程 CPU
L_CF_BIT	INT	通讯标记位号，远程 CPU

- 更多相关信息，可参考 CP 341 手册。

该手册还介绍了当 RCV_341 为通讯伙伴提供数据时，如何评估这些变量（请参见“提供数据”）。

- **ASCII 驱动程序**

无需对 RCV_341 的更多变量进行分配或评估。

更多相关信息，可参考 CP 341 手册。

如果在 HW Config 中设置了“分隔符”(Delimiter) 模式，请注意以下内容也适用：

输入缓冲区的长度 = 用户数据 + 分隔符。

- **MODBUS 主站**

无需对 RCV_341 的更多变量进行分配或评估。

根据所选的功能代码，将接收自伙伴的用户数据输入到输入缓冲器中。

更多相关信息，可参考手册《点对点 CP MODBUS 协议 RTU 格式的可装载驱动程序；S7 为主站》(Loadable Drivers for Point-to-Point CPs MODBUS Protocol RTU Format; S7 Is Master)。

- **MODBUS 从站**

在 MODBUS SLAVE 模式下，CP 341 模块的驱动程序控制数据交换。更多相关信息，可参考手册《点对点 CP MODBUS 协议 RTU 格式的可装载驱动程序；S7 为从站》(Loadable Drivers for Point-to-Point CPs MODBUS Protocol RTU Format; S7 Is Slave)。

必须在 RCV_341 块中设置输入 MODB_SL = TRUE。

功能和工作原理

块通过 P_RCV_RK (FB 122) 块（等同于 CP PtP 库中的 P_RCV_RK (FB 7) 块）从与 CP 341 连接的伙伴处接收数据。如果没有较高级别的错误排队，则由 P_RCV_RK 检测到的诊断事件将通过 ALARM_8P 进行报告。消息功能可以禁用。

- 通过设置输出 NDR = TRUE 接收新数据。NDR 将在下一次循环中复位。所接收的数据必须在该周期中由用户程序进行清除，并输入到与输入 R_DATA 互连的用户程序结构中。
- P_RCV_RK 输出的数据按 1:1 传送给 RCV_341 块的输出。只要没有未决的较高级别错误 (MODE = 16#40xxxxxx)，则 ALARM_8P 会在接收操作因错误而中止将生成错误消息。
- 在 MODBUS 从站模式中，MODB_341 (FB 80) 块控制 CP 341（作为 MODBUS 从站运行）和 MODBUS 主站之间的数据交换。MODB_341 等同于 MODBUS 库的 MODB_341。

冗余

在 H 系统中，更高级别的 MOD_CP 块评估 DP 主站系统的冗余。不支持冗余的串行通讯，因此必须由用户手动控制，且独立于此块。

错误处理

不检查输入参数的似然性。

4.25 RCV_341: 在串行模式下通过 CP 341 接收数据

启动特征

通过 OMODE 输出的字节 2 中的 LSB 报告重启 (OB 100)。

ALARM_8P 将被初始化。

过载行为

不可用

时间响应

不可用

消息响应

块使用 ALARM_8P 报告 P_RCV_RK 的诊断信息。可通过设置 EN_MSG = FALSE 禁用该消息功能。

操作和监视

该块没有面板。

更多信息

更多相关信息，可参考：

RCV_341 的消息文本和关联值 (页 495)

4.25.2 RCV_341 的 I/O

CFC 中块显示的工厂设置在“**I/O**”列中标识:

加粗字体的 I/O 名称表示 I/O 可见; 正常字体的 I/O 名称则表示 I/O 不可见。

有关所用缩写的说明和信息, 可参考“关于块描述的常规信息”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
ACC_MODE	1 = 接受 MODE 设置	BOOL	1	IO	
EN_MSG	1 = 启用消息	BOOL	1	I	
EN_R	启用接收/获取	BOOL	0	I	
ERROR	1 = 接收数据时出错	BOOL	0	O	
EV_ID	消息号	DWORD	0	I	
L_CF_BIT	协议 RK512: 通讯标记位号, 远程 CPU	INT	0	O	
L_CF_BYT	协议 RK512: 通讯标记字节号, 远程 CPU	INT	255	O	
L_NO	协议 RK512: 远程 CPU 的 DB 号	INT	0	O	
L_OFFSET	协议 RK512: 远程 CPU 的 DB 偏移量	INT	0	O	
L_TYP	协议 RK512: 远程 CPU 的区域类型	CHAR	''	O	
LADDR	逻辑地址 CP 341	INT	0	I	
LEN	接收数据的长度	INT	0	O	
MODB_SL	1 = MODBUS 从站处于激活状态	BOOL	0	I	
MODE	模块模式 (xx = 00 - 06/00 - 15/00 - 31)	DWORD	0	I	
MSG_ACK	消息确认	WORD	0	O	
MSG_STAT	消息错误信息	WORD	0	O	
NDR	1 = 接收新数据时未出错	BOOL	0	O	
QERR	1 = 程序错误	BOOL	1	O	
QMODF	1 = CP 341 错误	BOOL	0	O	
R	复位	BOOL	0	I	
R_DATA	接收的数据	ANY		I	

4.25 RCV_341: 在串行模式下通过 CP 341 接收数据

I/O (参数)	含义	数据类型	默认值	类型	OCM
RACK_NO	机架号	BYTE	0	I	
SLOT_NO	插槽号 (在 DP/PA 连接器中为 0)	BYTE	0	I	
STATUS	错误状态	WORD	0	O	
SUBN1_ID	一级 DP 主站系统的 ID	BYTE	16#FF	I	
SUBN2_ID	冗余 DP 主站系统的 ID	BYTE	16#FF	I	

更多信息

更多相关信息, 请参考以下部分:

RCV_341 的消息文本和关联值 (页 495)

4.25.3 RCV_341 的消息文本和关联值

分配消息文本和消息类别

消息块 ALARM_8P	消息号	默认消息文本	消息类别
EV_ID	1	CP 341 @1%d@/@2%d@/@3%d @事件等级@4%d@ 编号 @5%d@	S
	2		无消息
	3		无消息
	4		无消息
	5		无消息
	6		无消息
	7		无消息
	8		无消息

分配关联值

消息块 ALARM_8P	关联值	块参数	含义
EV_ID	1	SUBN_ID	DP 主站系统的编号 (字节)
	2	RACK_NO	机架/站号 (字节)
	3	SLOT_NO	插槽号 (字节)
	4	STATUS(EV_CLAS)	事件类别
	5	STATUS(EV_NO)	事件编号

4.26 SND_341: 通过 CP 341 发送串行数据

4.26.1 SND_341 的描述

对象名 (类型 + 编号)

FB 120

- SND_341 块 I/O (页 501)

应用领域

块 SND_341 用于通过 CP 341 模块传送串行数据。

调用 OB

OB 100 和用于传输数据的循环 OB (建议 100 ms)。

在 CFC 中的使用

使用 CFC 功能“生成模块驱动程序”时, 将自动执行以下操作:

- 组态 RACK_NO、SUBN1_ID、SUBN2_ID 和 SLOT_NO 输入。
- MODE 输入与 MOD_CP 块的 OMODE_00 输出互连。

创建用于传输串行数据的用户块的说明

要求: 必须安装用于组态点对点连接 (CP PtP Param) 的可选数据包。

在 HW Config 中, 也可以设置如下的传输类型 (协议):

- DK 3964R
- RK 512
- ASCII
- MODBUS 主站
- MODBUS 从站

步骤

- 在 CFC 图表中安装 SND_341 块。
- 在 LADDR 块输入中设置 CP 341 模块的逻辑基址。
- 为块中的用户数据定义发送缓冲器（以下称为 SND_DATA）。
- 在同一循环 OB 中 SND_341 的下游安装 SND_DATA。发送缓冲器的定义可以是一个简单变量或一个变量数组。变量定义允许除 ANY 外的所有 S7 数据类型。
- 将发送缓冲器应用到 SND_DATA 的输出。
- 将此输出与 SND_341 块的 S_DATA 输入互连。互连的数据结构长度决定要传送的协议长度。
- 要对传输的数据进行控制和评估，需在 SND_DATA 中定义如下 I/O，并将其与 SND_341 的相应 I/O 互连：

I/O	数据类型	含义
输入:		
DONE	BOOL	发送请求以无错结束
ERROR	BOOL	发送请求以出错结束
STATUS	WORD	错误状态
输出:		
REQ	BOOL	初始化发送请求
R	BOOL	重置数据传输

当输入 REQ = TRUE 时，块 SND_341 初始化发送请求。如果成功完成发送请求（DONE = TRUE），并且如果 REQ = TRUE，则自动初始化新的发送请求。如果要发送的数据仍不可用，则必须设置 REQ = FALSE。如果变量 ERROR = TRUE，会在 STATUS 中输入错误事件编号。有关这些事件编号的含义，可参考 CP 341 手册中的说明。应根据所选的传输类型对事件类别 7 进行评估，以确定如何处理所发送的出错的数据或未传输的数据。

- **程序 DK 3964R**

此程序无需对 SND_341 的更多变量进行任何评估或分配。

更多相关信息，可参考 CP 341 手册。

4.26 SND_341: 通过 CP 341 发送串行数据

- 远程耦合器 RK 512

SND_341 的这些变量（隐藏输入）设置如下：

I/O	数据类型	含义
R_CPU_NO	INT	远程 CPU 的编号
R_TYP	CHAR	远程 CPU 的区域类型
R_NO	INT	远程 CPU 的 DB 号
R_OFFSET	INT	远程 CPU 的 DB 偏移量
R_CF_BYT	INT	通讯标记字节号，远程 CPU
R_CF_BIT	INT	通讯标记位号，远程 CPU
<p>更多信息，可参考 CP 341 手册。 该手册还介绍了当 SND_341 从通讯伙伴获取数据时如何设置变量（请参见“获取数据”）。在这种情况下，输入变量被隐藏。 必须设置 SF（发送或获取，数据类型 CHAR）=“F”。</p>		

- ASCII 驱动程序

无需对 SND_341 的更多变量进行分配或评估。

更多信息，可参考 CP 341 手册。

- MODBUS 主站

SND_341 的变量（隐藏输入）R_TYP 必须设置为等于“X”。下表根据传输协议的功能代码给出了请求帧发送缓冲器的构造方法：

字节	含义
1	MODBUS 从站地址
2	MODBUS 功能代码
3	参见功能代码 x
4	参见功能代码 x
:	
x	CRC 检查（消息帧校验和）
x+1	CRC 检查（消息帧校验和）

“主站-从站”数据传送起始于从站地址，后跟功能代码和数据的传送。数据域的结构由所使用的功能代码确定。

更多相关信息，可参考手册《点对点 CP MODBUS 协议 RTU 格式的可装载驱动程序；S7 为主站》(Loadable Drivers for Point-to-Point CPs MODBUS Protocol RTU Format; S7 Is Master)。消息帧末尾的 CRC 检查由 CP 341 模块上的 MODBUS 主站驱动程序形成。

- **MODBUS 从站**

在 MODBUS SLAVE 模式下，CP 341 模块的驱动程序控制数据交换。更多信息，可参考手册《点对点 CP MODBUS 协议 RTU 格式的可装载驱动程序；S7 为从站》(Loadable Drivers for Point-to-Point CPs MODBUS Protocol RTU Format; S7 Is Slave)。

无需在图表中插入 SND_341 块。

更多相关信息，可参考“RCV_341 MODBUS 从站”。

功能和工作原理

该块使用 P_SND_RK (FB 123) 块将数据传送到与 CP 341 连接的通讯伙伴。P_SND_RK 等同于 CP PtP 库中的 P_SND_RK (FB 8)。只要没有较高级别错误 (MODE = 16#40xxxxxx) 处于待决状态，就会通过 ALARM_8P 报告 P_SND_RK 所检测到的诊断事件。消息功能可以禁用。

- 输入 REQ = TRUE 后即开始数据传送。只有通过 P_SND_RK 设置 DONE = TRUE 或 ERROR = TRUE 之后，才可能产生新的发送请求。P_SND_RK 输出中的数据按 1:1 传送为 SND_341 的输出。
- 要传送的数据的长度由与输入 S_DATA 互连的用户块中的传输数据结构的长度确定。

冗余

在 H 系统中，更高级别的 MOD_CP 块评估 DP 主站系统的冗余。不支持冗余的串行通讯，因此必须由用户手动控制，且独立于此块。

错误处理

不检查输入参数的似然性。

4.26 SND_341: 通过 CP 341 发送串行数据

启动特征

通过 OMODE 输出的字节 2 中的 LSB 报告重启 (OB 100)。

ALARM_8P 将被初始化。

过载行为

不可用

时间响应

不可用

消息响应

块使用 ALARM_8P 报告 P_SND_RK 的诊断信息。

可通过设置 EN_MSG = FALSE 禁用该消息功能。

操作和监视

该块没有面板。

注: 如果在 CFC 的块对象属性中选择了“启用操作员监控”(Enable operator control and monitoring) 选项, 则会在“...的 I/O” (OCM 列, “+”) 下标识传送到 OS 的变量。默认设置: 不激活选项。

更多信息

更多相关信息, 可参考:

SND_341 的消息文本和关联值 (页 503)

4.26.2 SND_341 的 I/O

CFC 中块显示的工厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示 I/O 可见; 正常字体的 I/O 名称则表示 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
ACC_MODE	1 = 接受 MODE 设置	BOOL	1	IO	
DONE	1 = 请求结束, 且未出错	BOOL	0	O	
EN_MSG	1 = 启用消息	BOOL	1	I	
ERROR	1 = 请求结束, 但出错	BOOL	0	O	
EV_ID	消息号	DWORD	0	I	
LADDR	逻辑地址 CP 341	INT	0	I	
MODE	模块模式 (xx = 00 - 06/00 - 15/00 - 31)	DWORD	0	I	
MSG_ACK	消息确认	WORD	0	O	
MSG_STAT	消息错误信息	WORD	0	O	
QERR	1 = 程序错误	BOOL	1	O	
QMODF	1 = CP 341 错误	BOOL	0	O	
R	复位	BOOL	0	I	
R_CF_BIT	协议 RK512: 远程 CPU 的编号	INT	0	I	
R_CF_BYT	协议 RK512: 远程 CPU 的编号	INT	255	I	
R_CPU_NO	协议 RK512: 远程 CPU 的编号	INT	1	I	
R_NO	协议 RK512: 远程 CPU 的编号	INT	0	I	
R_OFFSET	协议 RK512: 远程 CPU 的编号	INT	0	I	
R_TYP	协议 RK512: 远程 CPU/MODBUS 主站的编号 = X	CHAR	'X'	I	
RACK_NO	机架号	BYTE	16#FF	I	
REQ	初始化请求	BOOL	0	I	
S_DATA	发送数据	ANY		I	
SF	发送 (S) 或接收/获取 (F)	CHAR	'S'	I	
SLOT_NO	插槽号	BYTE	16#FF	I	

4.26 SND_341: 通过 CP 341 发送串行数据

I/O (参数)	含义	数据类型	默认值	类型	OCM
STATUS	错误状态	WORD	0	O	
SUBN1_ID	一级 DP 主站系统的 ID	BYTE	16#FF	I	
SUBN2_ID	冗余 DP 主站系统的 ID	BYTE	16#FF	I	

更多信息

更多相关信息，请参考以下部分：

SND_341 的消息文本和关联值 (页 503)

4.26.3 SND_341 的消息文本和关联值

分配消息文本和消息类别

消息块 ALARM_8P	消息号	默认消息文本	消息类别
EV_ID	1	CP 341 @1%d@/@2%d@/@3%d @事件等级@4%d@ 编号 @5%d@	S
	2		无消息
	3		无消息
	4		无消息
	5		无消息
	6		无消息
	7		无消息
	8		无消息

分配关联值

消息块 ALARM_8P	关联值	块参数	含义
EV_ID	1	SUBN_ID	DP 主站系统的编号 (字节)
	2	RACK_NO	机架/站号 (字节)
	3	SLOT_NO	插槽号 (字节)
	4	STATUS(EV_CLAS)	事件类别
	5	STATUS(EV_NO)	事件编号

4.27 内部块

4.27.1 MODB_341: 内部块

对象名 (类型 + 编号)

FB 80

该块用于 MODBUS SLAVE 模式。

说明

在 MODBUS SLAVE 模式中, CP 341 模块的驱动程序控制数据交换。

更多相关信息, 请参考手册《点对点 CP MODBUS 协议 RTU 格式的可装载驱动程序; S7 为从站》(Loadable Drivers for Point-to-Point CPs MODBUS Protocol RTU Format; S7 Is Slave)。

系列： MAINT

5.1 ASSETMON： 监视是否超出限制值的过程变量

5.1.1 ASSETMON 的描述

对象名（类型 + 编号）

FB86

- ASSETMON 块 I/O (页 511)
- 资产管理块图标 (页 702)
- ASSETMON 面板 (页 651)

应用领域

该块用于监视三个仿真过程变量是否超出三个限制值。它报告过程变量等待状态的时间和方式如下：

- 限制值过冲时
- 通过基于设备的质量代码或
- 通过二进制消息输入。

说明

该块不包含任何基于工艺或设备的处理部分，因此不在驱动程序生成器的考虑范围之内。

调用 OB

在其中安装了该块的循环中断 OB（例如 OB32）和 OB100。

5.1 ASSETMON: 监视是否超出限制值的过程变量

功能和工作原理

如果输入 MONITOR = 1 (默认设置), 则该块监视输入 PV_0、PV_1 和 PV_2 是否下冲或过冲三个限制值。当达到或过冲限制值时, 系统会将相应的输出 QLR_PV_x (维护要求)、QLD_PV_x (维护请求) 和 QLA_PV_x (维护报警) (x = 0、1、2) 设置为 TRUE, 并输出适用的消息。

只要其中一个过程变量报告某限制值 (例如维护请求) 过冲, 这三个过程变量就都会进行限制监视。只有所有过程变量下冲此限制值至少一个周期时, 才会再次创建一条新消息。将分别向这三个过程变量发送一条消息。

在 EDD 的参数数据中为相关实例输入监视视图中过程值的标签 (另请参阅 PLT ID 部分)。

通过将 SUP_MR_x、SUP_MD_x 或 SUP_MA_x 设置为 1 可关闭对各限制值的监视。

除限制值监视外, 还会分析各过程变量的质量代码。过程值的特定于设备的质量代码会触发同样为限制值监视所使用的相应消息。

此外, 还有七个消息输入 (MESSAGE1, ..., MESSAGE7) 可用; 如果设置了状态 = 1, 这些输入会生成消息。

二进制消息输入

MESSAGE1	维护报警	消息 (S) 需要确认
MESSAGE2	维护请求	消息 (F) 需要确认
MESSAGE3	需要维护	消息 (M) 需要确认
MESSAGE4	本地操作员控制	消息 (SA) 不需要确认
MESSAGE5	仿真	消息 (SA) 不需要确认
MESSAGE6	服务中断	消息 (S) 需要确认
MESSAGE7	已钝化	消息 (SA) 不需要确认

对于每个 PV_x, 输入 (RESETx) 和输出 (QRESETx) 都是可用的, 它们可用于将工艺块中 PV_x 的计数值复位 (例如 COUNT_P 块的输出 QRESETx 与输入 RESET 的互连)。

如果 MONITOR = 0, 则不会分析这三个过程变量及其质量代码, 而是只分析 7 个输入 (MESSAGE1, ..., MESSAGE7)。

所创建的带有 ALARM_8P 的状态代表需要确认的消息，而带有 NOTIFY_8P 的状态则代表不需要确认的消息。可通过设置 EN_MSG = 0 禁用该消息功能。此时设置为 MS = 8。

详细诊断通过布尔输入 DIAG1 至 DIAG16 显示在面板的诊断视图中。

如果将其中一个输入设置为 1，则状态显示即会显示在相应文本之前。

在 EDD 的参数数据中为相关实例输入对相关输入 DIAG1 至 DIAG16 的测试（另请参阅 PLT ID 部分）。

如果将 DIAG1 至 DIAG16 中的一个输入设置为 TRUE，则如果通过 PV_0、PV_1 或 PV_2 触发内部报警，此时就会输出说明性文本消息“更多状态可用”。

PLT-ID

PLT-ID 是 PDM 对象（参数数据 EDD）和维护工作站面板之间的连接参数。PLT-ID 与 PDM 对象相连。

PDM 对象在 SIMATIC 管理器中的生成方式如下：

1. 在 SIMATIC 管理器中选择“视图”(View) > “过程设备工厂视图”(Process device plant view)。
2. 选择“插入”(Insert) > SIMATIC PDM > TAG。
3. 突出显示所插入的 TAG 对象并选择上下文菜单命令 SIMATIC PDM > “设备选择...”(Device Selection...)
4. 在树形结构 CFC > DATA_OBJECTS > CFC > 中，选择 AssetMon 然后单击“确定”(OK) 关闭对话框。
5. 在上下文菜单中选择“打开对象”(Open Object) 然后在参数分配画面窗体中输入所有必需的数据。
6. 选择“文件”(File) > “保存”(Save)。
参数分配画面窗体将关闭。
7. 选择此 TAG 对象然后选择“工具”(Tools) > SIMATIC PDM > “创建 PLT-ID”(Create PLT-ID)。

然后可以在相关联的参数“PLT_ID”中为所生成的 PLT-ID 分配参数。

说明

不可以单独更改或删除 PLT-ID。

5.1 ASSETMON: 监视是否超出限制值的过程变量

创建维护状态 (MS)

MS 取决于:

- 质量代码输入 QC_0、QC_1 和 QC_2。仅基于设备的错误有影响，而过程相关的错误没有任何影响。
- PV_0、PV_1 和 PV_2 的限制监视。
- 二进制消息输入（外部 MS）。
- 可互连的输入 MS_IN（外部 MS）。
- 可互连的输入 STATUS（不可更改的设备质量代码）。

所有这些事件中优先级最高的事件将显示在 MS 中。

16 个 DIAGx 对 MS 没有任何影响，相反它们仅用于面板的诊断视图中详细诊断的可视化。

可用于创建维护状态 (MS) 的质量代码

说明

下表专用于 MS，而不用于各过程值的质量代码显示，这些过程值仅由 QC_x 参数创建。优先级与 MS 编码类似，因此以下规则适用：MS 的数值越高，优先级越高。

信号	信号详细信息	质量代码	MS
PV_x QualityCode 输入	故障，由设备引起	0x00	7
PV_x QualityCode 输入	故障，由设备引起	0x14	7
PV_x QualityCode 输入	故障，由设备引起	0x18	7
PV_x QualityCode 输入	不确定，特定于设备	0x44	7
PV_x QualityCode 输入	不确定，特定于设备	0x48	7
PV_x QualityCode 输入	不确定，特定于设备	0x68	6
PV_x QualityCode 输入	不确定，特定于设备	0x54	6
PV_x QualityCode 输入	需要维护	0xA4	5
PV_x QualityCode 输入	至少仿真 1 个 PV	0x60	3
PV_x 上限值超出	维护报警	0x24	7

信号	信号详细信息	质量代码	MS
PV_x 上限值超出	维护请求	0xA8	6
PV_x 上限值超出	需要维护	0xA4	5
MESSAGE1	维护报警	0x24	7
MESSAGE2	维护请求	0xA8	6
MESSAGE3	需要维护	0xA4	5
MESSAGE4	本地操作员控制	0x3C	4
MESSAGE5	仿真	0x60	3
MESSAGE6	服务中断	0x1C	2
MESSAGE7	已钝化	0x23	1
消息被禁用			8
MS_IN	外部维护状态		0-9
STATUS	从 0 到 255 的所有质量代码都是可能的, 根据 QC_MS_STAT 表创建 MS	0x00 ... 0xFF	0-9

错误处理

发生算术错误时, 将设置输出 ENO = 0 和 QERR = 1。

启动特征

启动之后, 将在 RUNUPCYC 所设置的周期数内抑制消息。

时间响应

不适用。

5.1 ASSETMON: 监视是否超出限制值的过程变量

消息响应

该块通过 ALARM_8P 和 NOTIFY_8P 进行报告。

更多信息

有关此主题的更多信息，可参考以下部分：

操作和监视 ASSETMON (页 514)

ASSETMON 的消息文本和关联值 (页 513)

MS 的维护状态 (页 732)

关于使用驱动程序块的注意事项 (页 329)

5.1.2 ASSETMON 的描述

CFC 中块显示的出厂设置在“**I/O**”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	类型	默认值	类型	OCM
DIAGx	资产详细信息诊断 (x = 1 到 16)	BOOL	0	I	+
DIFFALMx	下一预期报警的差异值 (x = 0、1 或 2)	REAL	0	O	+
EN_MSG	1 = 启用消息	BOOL	1	I	
EV_IDx	消息号 (x = 1、2 或 3)	DWORD	0	I	
LA_PV_x	限制值 PV_x (x = 0、1 或 2) 维护报警	REAL	100	i	+
LD_PV_x	限制值 PV_x (x = 0、1 或 2) 维护请求	REAL	100	i	+
LR_PV_x	需要限制值 PV_x (x = 0、1 或 2) 维护	REAL	100	i	+
MESSAGE1	1 = 消息: 不良, 维护报警	BOOL	0	I	
MESSAGE2	1 = 消息: 不确定, 维护请求	BOOL	0	I	
MESSAGE3	1 = 消息: 良好, 需要维护	BOOL	0	I	
MESSAGE4	1 = 消息: 钝化	BOOL	0	I	
MESSAGE5	1 = 消息: 不良, 服务中断	BOOL	0	I	
MESSAGE6	1 = 消息: 不良, 本地操作/功能控制	BOOL	0	I	
MESSAGE7	1 = 消息: 不确定, 仿真	BOOL	0	I	
MONITOR	1 = 监视 PV_x (x = 0、1 或 2) 打开	BOOL	0	I	
MS	维护状态	DWORD	0	I	+
MS_IN	外部可互连的 MS	DWORD	0	I	
STATUS	设备的 QC 未改变	BYTE	16#80	I	
MSG_ACKx	消息确认 (x = 1 或 2)	WORD	0	O	
MSGSTATx	消息错误信息 (x = 1、2 或 3)	WORD	0	O	
O_MS	维护状态	DWORD	0	O	+
PLT_ID	EDD 的资产 ID	DWORD	0	I	+
PV_x	过程值 PV_x (x = 0、1 或 2)	REAL	0	I	+

5.1 ASSETMON: 监视是否超出限制值的过程变量

I/O (参数)	含义	类型	默认值	类型	OCM
QC_x	质量代码 PV_x (x = 0、1 或 2)	BYTE	16#80	I	
QERR	1 = 程序错误	BOOL	1	O	
QLA_PV_x	1 = 限制值 PV_x (x = 0、1 或 2), 超出维护报警	BOOL	0	O	+
QLD_PV_x	1 = 限制值 PV_x (x = 0、1 或 2), 超出维护请求	BOOL	0	O	+
QLR_PV_x	1 = 限制值 PV_x (x = 0、1 或 2), 超出需要维护	BOOL	0	O	+
QRESET_x	1 = 复位 PV_x (x = 0、1 或 2)	BOOL	0	O	
RESET_x	1 = 复位 PV_x (x = 0、1 或 2)	BOOL	0	I	+
RUNUPCYC	起动机周期数	INT	3	I	
SUP_MA_x	1 = 抑制监视限制值 PVx (x = 0、1 或 2), 超出维护报警	BOOL	0	I	+
SUP_MD_x	1 = 抑制监视限制值 PVx (x = 0、1 或 2), 超出维护请求	BOOL	0	I	+
SUP_MR_x	1 = 抑制监视限制值 PVx (x = 0、1 或 2), 超出需要维护	BOOL	0	I	+

5.1.3 ASSETMON 的消息文本和关联值

分配消息文本和消息类别

消息块	消息号	默认消息文本	消息类别
EV_ID1 (ALARM_8P)	1	不良, 维护报警, PV_0 @4W%t#ASSETMON_TXT@	S
	2	不确定, 维护请求, PV_0 @4W%t#ASSETMON_TXT@	F
	3	良好, 需要维护, PV_0 @4W%t#ASSETMON_TXT@	M
	4	不良, 维护报警, PV_1 @4W%t#ASSETMON_TXT@	S
	5	不确定, 维护请求, PV_1 @4W%t#ASSETMON_TXT@	F
	6	良好, 需要维护, PV_1 @4W%t#ASSETMON_TXT@	M
	7	不良, 维护报警, PV_2 @4W%t#ASSETMON_TXT@	S
	8	不确定, 维护请求, PV_2 @4W%t#ASSETMON_TXT@	F
EV_ID2 (ALARM_8P)	1	良好, 需要维护, PV_2 @2W%t#ASSETMON_TXT@	M
	2	不良, 维护报警, 外部 @2W%t#ASSETMON_TXT@	S
	3	不确定, 维护请求, 外部 @2W%t#ASSETMON_TXT@	F
	4	良好, 需要维护, 外部 @2W%t#ASSETMON_TXT@	M
	5	不良, 设备中断服务	S

5.1 ASSETMON: 监视是否超出限制值的过程变量

消息块	消息号	默认消息文本	消息类别
EV_ID3 (NOTIFY_8P)	1	不良, 钝化	SA
	3	不良, 本地操作/功能控制	SA
	4	不确定, 仿真	SA
	5	组态更改	SA
	6	过程相关故障	SA

关联值

消息块 ALARM 8P	关联值	含义
EV_ID1	4	来自 ASSETMON_TXT 的文本编号
EV_ID2	2	来自 ASSETMON_TXT 的文本编号

有关消息文本及其文本编号的信息, 请参考“ASSETMON 的文本库 (页 729)”。

5.1.4 ASSETMON 的操作员监控

更多信息

更多信息可参考以下部分:

- 资产管理块图标 (页 702)
- ASSETMON 面板 (页 651)

5.2 MS_MUX: 确定最差的个别状态

5.2.1 MS_MUX 的描述

对象名 (类型 + 编号)

FC 288

- MS_MUX 块 I/O (页 516)

应用领域

块 MS_MUX 将通过选择 FF 现场设备最差的个别状态, 确定由多个 FF 现场设备组成的功能单元的维护状态 (MS, Maintenance Status)。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32)。

功能和工作原理

输入 MS_x (0<=x <= 9) 与 ASSETMON 块 (每个块代表一个 FF 现场设备) 的输出 Q_MS 互连。

在输入 MS_x 中, 最差维护状态 (MS, Maintenance Status) 会在输出 QMS 处确定并提供, 以便进一步与功能单元的 ASSETMON 块的输入 MS_IN 互连。

ASSET 中的安装

有关安装 MS_MUX 块的信息, 请参考在 ASSET 中链接 FF 设备 (页 727)。

启动特征

不可用

时间响应

不可用

5.2 MS_MUX: 确定最差的个别状态

消息响应

不可用

操作和监视

不可用

5.2.2 MS_MUX 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
MS_x	维护状态 _x (x = 0- 3)	DWORD	0	I
MS_x	维护状态 _x (x = 4-9)	DWORD	0	I
QMS	最差维护状态	DWORD	0	O

5.3 ST_MUX: 确定 FF 现场设备的状态值

5.3.1 ST_MUX 的描述

对象名 (类型 + 编号)

FC 287

- ST_MUX 块 I/O (页 518)

应用领域

块 ST_MUX 可确定 FF 现场设备的状态, 方法是从现场设备的信号处理块 FF_A_DI、FF_A_AI、FF_A_DO 和 FF_A_AO 中选择最差的信号处理块。

调用 OB

调用 OB 是在其中安装了该块的循环中断 OB 3x (例如 OB 32)。

功能和工作原理

输入 ST_x (0 ≤ x ≤ 9) 与 FF 现场设备所有信号处理块的输出 STATUS (FF_A_DI 和 FF_A_AI) 或 ST_xx 或 QC_xx (FF_A_DO 和 FF_A_AO) 互连。在 ST_x 输入中, ST_MUX 可确定最差状态并使其可在输出 QST 中使用, 以便进一步与 ASSETMON 组件互连。

如果存在多个具有最高优先级的状态, 则将使用第一个 ST_x 输入的值。

ASSET 中的安装

有关安装 ST_MUX 块的信息, 请参考在 ASSET 中链接 FF 设备 (页 727)。

启动特征

不可用

时间响应

不可用

5.3 ST_MUX: 确定 FF 现场设备的状态值

消息响应

不可用

操作和监视

不可用

5.3.2 ST_MUX 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
QST	最差状态	BYTE	16#80	O
ST_x	Status_x (x = 0- 3)	BYTE	16#80	I
ST_X	Status_x (x = 4-9)	BYTE	16#80	I

5.4 STATEREP: 块组的状态显示

5.4.1 STATEREP 的描述

对象名 (类型 + 编号)

FB 87

- STATEREP 块 I/O (页 520)

应用领域

块 STATEREP 用于显示专门用来自动隐藏消息的一组块的状态。

功能和工作原理

该块具有 32 个 BOOL 输入，用于描述已定义状态。根据所设置的状态，将在 INT 输出处输出该状态。

错误处理

如果设置了多个参数，则该块将输出消息 QSTATE=0，以及 QERR=1。

启动特征

不可用。

时间响应

不可用。

消息响应

不可用。

5.4.2 STATEREP 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O	含义	数据类型	默认值	类型	OCM
状态 1 .. 10	过程状态 1 – 10	BOOL	0	I	
状态 11 .. 32 32	过程状态 11 - 32	BOOL	0	I	
QERR	1 = 错误, 激活了多个状态	BOOL	1	O	
QSTATE	过程状态	INT	0	O	+

系列: @SYSTEM

6.1 AL_DELAY - 报警延迟

6.1.1 AL_DELAY 描述

对象名（类型 + 编号）和系列

FC 290

系列: @SYSTEM

工作原理

该块被所有具有消息功能的块使用，因此它没有完整的帮助系统。它用于延迟报警的触发，即仅当报警原因存在的时间长于指定限时时才触发报警。还可以对退出报警状态进行延迟。

6.2 内部块

6.2.1 P_RCV_RK: 内部块

对象名 (类型 + 编号)

FB 122

此块为系统块, 仅供内部使用。因此, 该块没有帮助信息。

6.2.2 P_SND_RK: 内部块

对象名 (类型 + 编号)

FB 123

此块为系统块, 仅供内部使用。因此, 该块没有帮助信息。

系列: TIME

7.1 OB1_TIME: 确定 CPU 利用率

7.1.1 OB1_TIME 的描述

对象名 (类型 + 编号)

FB 69

- OB1_TIME 块 I/O (页 525)

功能

OB1_TIME 块提供有关 CPU 利用率的信息。

工作原理

块 OB1_TIME 安装在 OB 1 中。

- 该块由输入 STOP_RES 处的负沿 (1 → 0) 复位 (即, 复位 CNT、MAX、MIN、MEAN 和内部 ACT_TME, MIN = 2147483000) 和启动。可确定当前系统时间并将其内部保存在 L_TME 中。
- 在每个执行周期内, 该块确定系统的日时钟 (以 ms 计), 将其内部保存在 ACT_TIME 中, 并计算自复位时间以来的最大值 (MAX), 以及自最后一次执行 (OB_1_TIME = ACT_TIME - L_TIME) 后所经过时间的均方根值 (MEAN) 和最小值 (MIN)。接下来, 计数器 CNT 增加 1, 并复位 L_TIME = ACT_TIME。均方根值按如下方式计算:

$$MEAN = \sqrt{\frac{1}{CNT+1} (CNT * MEAN^2 + OB1_TIME^2)}$$

- 计算出的值必须经由调试人员解释, 以便得到 CPU 的利用率。
- 输入 STOP_RES 中的 1 将使得系统不再进一步处理块算法 (处理为“暂停”)。在此期间 ENO 复位为 0。

系列: *TIME*

7.1 *OB1_TIME*: 确定 *CPU* 利用率

调用 OB

OB 1

启动特征

复位为默认值。

消息响应

不可用

错误处理

仅通过操作系统。

操作和监视

不可用

7.1.2 OB1_TIME 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
CNT	计数器	DINT	0	O
MAX	最大时间值 (以毫秒计)	DINT	0	O
MAX_CNT	最大 CNT 值 (以毫秒计)	DINT	10000	I
MEAN	二次均方值 (以毫秒计)	DINT	0	O
MIN	最小时间值 (以毫秒计)	DINT	0	O
OB1_TIME	循环执行时间: ACT_TIME - L_TIME	DINT	0	O
QERR	ENO 的取反值	BOOL	1	O
STOP_MAX	如果 CNT = MAX_CNT, 那么 1 = STOP	BOOL	0	I
STOP_RES	停止/复位: 1 = 停止, 0 = 复位	BOOL	1	I

系列: *TIME*

7.1 *OB1_TIME*: 确定 *CPU* 利用率

系列: MATH

8.1 ADD4_P: 最多处理 4 个值的加法器

8.1.1 ADD4_P 的描述

对象名 (类型 + 编号)

FC256

- ADD4_P 块 I/O (页 528)

功能

块 ADD4_P 计算最多 4 个值的总和:

$$V = U1 + \dots + Un \quad (n \leq 4)$$

调用 OB

仅适用于其中安装了该块的 OB。

错误处理

出现上溢/下溢时, 上/下限的 REAL 值在结果 V 中设置。ENO 将被设置为下限, QERR 被设置为上限。

8.1 ADD4_P: 最多处理 4 个值的加法器

8.1.2 ADD4_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
U1	加数 1	REAL	0.0	I
U2	加数 2	REAL	0.0	I
U3	加数 3	REAL	0.0	I
U4	加数 4	REAL	0.0	I
V	结果	REAL	0.0	O

8.2 ADD8_P: 最多处理 8 个值的加法器

8.2.1 ADD8_P 的描述

对象名 (类型 + 编号)

FC 257

- ADD8_P 块 I/O (页 529)

功能

ADD8_P 块最多可计算 8 个值的和:

$$V = U1 + U2 + U3 + \dots + Un \quad (n \leq 8)$$

错误处理

出现上溢/下溢时, 将在结果 V 中设置已违反的上/下限的 REAL 值, 并将 ENO 设置为 0。

调用 OB

仅适用于其中安装了该块的 OB。

8.2.2 ADD8_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
U1	加数 1	REAL	0.0	I
....
U8	加数 8	REAL	0.0	I
V	结果	REAL	0.0	O

8.3 AVER_P: 平均时间值

8.3.1 AVER_P 的描述

对象名 (类型 + 编号)

FB34

- AVER_P 块 I/O (页 532)

功能

块 AVER_P 根据其启动后所经过的时间计算活动参数的平均时间值。使用以下等式:

$$V = (N * Vold + U)/(N+1)$$

说明:

- U = 应用的参数
- V = 当前平均值
- Vold: 启动后所执行周期数的平均值
- N = 用于计算平均值的周期数

工作原理

该块的工作原理:

- 从 RUN 输入的上升沿开始计算。现有结果 V 被输入值 U 覆盖。更多相关信息, 可参考“启动特征”部分。
- 在后续周期中, 将重新计算输出 V 中的结果, 并且周期计数器 N 的值增加。
- 通过复位 RUN 输入终止计算, 并保存结果 V 和 N 的实际值。

调用 OB

安装了块的循环中断 OB (例如 OB32) 和 OB100。

错误处理

运算错误通过 $ENO = 0$ 或 $QERR = 1$ 表示。REAL 数据类型的范围限制超出上述公式结果时, 将出现运算错误。在此情况下, 保留上一周期的值 V 。如果 V 由于对应值 U 而取值 $\# + INF$ 或 $\# - INF$, 那么在下一周期中也会出现运算错误。

启动特征

初始运行期间和 CPU 启动期间, 将执行以下动作:

- 输入值 U 写入到输出值 V 中。
- 将复位周期计数器 N 。

这通过在启动 OB 中调用该块来完成。

时间响应

为了使块能够执行需要的功能, 必须在循环中断 OB 中调用该块。可以使用下面的公式计算 $T_{average}$ 时间:

$$T_{average} = N * T_{sampling}$$

, 其中 $T_{sampling}$ 是该块的采样时间。

用 CFC 组态时, 必须考虑较高级别运行组的块及其采样参数。

8.3.2 AVER_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
N	计算中使用的周期数	REAL	0	O
QERR	1 = 错误输出 (ENO 取反)	BOOL	1	O
运行	求平均值: 0 = 关闭, 1 = 开启	BOOL	0	I
U	输入	REAL	0	I
V	平均值	REAL	0	O

8.4 COUNT_P: 计数器

8.4.1 COUNT_P 的描述

对象名 (类型 + 编号)

FB36

- COUNT_P 块 I/O (页 535)

功能

根据所设置模式的不同, 二进制输入信号 I0 的上升沿使计数器值 V 递增或递减。

工作原理

块 COUNT_P 的工作原理如下:

- 运行模式可在 MODE 参数中设置:
 - MODE = 0 = 向上计数
 - MODE = 1 = 向下计数
- 用作向上计数器时的块行为:
 - I0 输入的每个上升沿使计数器递增。
 - 达到上限 $V = V_{HL}$ 时, 计数器值不再增加, 并且输出 QVHL 被设置为上限。
 - 模式转为“向下计数”时, 输出 V 在 I0 的下一个上升沿减少, 并将 QVHL 复位。
 - RESET = 1 可设置 $V = V_{LL}$ 、QVLL = 1、QVHL = 0, 并且内部沿标记更正为输入值。
- 用作向下计数器时的块行为:
 - I0 输入的每个上升沿使计数器值递减。
 - 达到下限 $V = V_{LL}$ 时, 计数器值不再减少, 并且将输出 QVLL 设置为上限。
 - 模式转为“向上计数”时, 输出 V 在 I0 的下一个上升沿递增, 并复位 QVLL。
 - RESET = 1 可设置 $V = V_{HL}$ 、QVHL = 1、QVLL = 0, 并且内部沿标记更正为输入值。

调用 OB

仅适用于其中安装了该块的 OB（例如 OB32）。

错误处理

运算错误由 $ENO = 0$ 或 $QERR = 1$ 表示。

启动特征

根据运行模式设置，初始运行期间和 CPU 启动期间，该块执行一次 RESET。更多相关信息，可参考“工作原理，RESET”。

时间响应

不适用。然而，建议还在包含边沿触发块的 OB 中安装该块。

8.4.2 COUNT_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	允许值
I0	输入	BOOL	0	I	
MODE	模式: 0 = 向上计数, 1 = 向下计数	BOOL	0	I	
QERR	1 = 错误	BOOL	1	O	
QVHL	1 = 已达到 V_HL	BOOL	0	O	
QVLL	1 = 已达到 V_LL	BOOL	0	O	
RESET	1 = 复位	BOOL	0	I	
V	计数器值	DINT	0	O	
V_HL	V 的上限	DINT	100	I	$V_{HL} \geq V_{LL}$
V_LL	V 的下限	DINT	0	I	$V_{LL} \leq V_{HL}$

8.5 MEANTM_P: 计算平均时间值

8.5.1 MEANTM_P 的描述

对象名 (类型 + 编号)

FB42

- MEANTM_P 块 I/O (页 538)

功能

块 MEANTM_P 用于根据以下方程式, 计算模拟输入信号在可组态的过去时段内的平均时间值:

$$V_n = (U_1 + U_2 + \dots + U_n) / n$$

其中 $U_1 \dots U_n$ 是用于求平均值的检测值。

工作原理

每次块执行期间, 通过当前输入值 U 和时间 T_WINDOW 期间内保存的值计算算术平均值。然后在输出 V 处进行更新。当前输入值随后会覆盖最早的先前值。

- 在参数 T_WINDOW 中输入待求平均值的时间窗口。
- 该块将根据 $T_WINDOW / SAMPLE_T$ 商的整数值, 确定需要保存值的个数 n 。
- 块的内部存储器中最多可保存 20 个先前的值。如果时间窗口变长, 数据将减少。
- $STOP_RES$ 输入具有以下影响:
 - 如果 $STOP_RES = "1"$, 则停止计算过程。在此期间内, 输出值保持不变。
 - 通过下降沿 $1 \rightarrow 0$ 重置平均时间值计算。
- 如果更改了 $SAMPLE_T$ 或 T_WINDOW , 则重置平均值。

说明

必须以这种方式设置参数 T_WINDOW 和 $SAMPLE_T$, 使 $T_WINDOW / SAMPLE_T$ 之商不大于 32748。

调用 OB

这是在其中安装了该块的循环中断 OB（例如 OB32）。

错误处理

仅通过操作系统

启动特征

不可用

如果 CPU 停止前块是激活的，而且之后继续进行计算，则需要考虑与 T_WINDOW 有关的 CPU 停止时间。这样就可决定是否仍可使用该结果或是否需要通过输入 STOP_RES 来重置计算过程。

时间响应

必须从循环中断 OB 中调用块。

8.5 MEANTM_P: 计算平均时间值

8.5.2 MEANTM_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	允许值
QERR	1 = 错误	BOOL	1	O	
SAMPLE_T	采样时间 (秒)	REAL	1.0	I	>0
STOP_RES	停止/重置平均值计算	BOOL	0	I	
T_WINDOW	时间窗口的大小 (秒)	REAL	20	I	
U	输入值	REAL	0	I	
V	输出值	REAL	0	O	

8.6 MUL4_P: 最多处理 4 个值的乘法器

8.6.1 MUL4_P 的描述

对象名 (类型 + 编号)

FC262

- MUL4_P 块 I/O (页 540)

功能

块 MUL4_P 最多可实现 4 个值相乘

$V := U1 * \dots * Un$ ($n \leq 4$)

参数值	类型	编号	含义
MUL4_P	FC	262	带有 4 个输入的乘法器

调用 OB

在其中安装了该块的 OB。

错误处理

出现上溢/下溢时, 上/下限的 REAL 值在结果 V 中设置。ENO 将被设置为下限, QERR 被设置为上限。

8.6 MUL4_P: 最多处理 4 个值的乘法器

8.6.2 MUL4_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
U1	输入 1	REAL	1.0	I
....
U4	输入 4	REAL	1.0	I
V	结果	REAL	1.0	O

8.7 MUL8_P: 最多处理 8 个值的乘法器

8.7.1 MUL8_P 的描述

对象名 (类型 + 编号)

FC263

- MUL8_P 块 I/O (页 542)

功能

块 MUL8_P 最多可实现 8 个值相乘

$V := U1 * U2 * U3 * \dots * Un$ ($n \leq 8$)

说明

如果最多要使 4 个值相乘, 请使用 MUL4_P 取代 MUL8_P, 以减少计算时间。

参数值	类型	编号	含义
MUL8_P	FC	263	带有 8 个输入的乘法器

调用 OB

在其中安装了该块的 OB。

错误处理

出现上溢/下溢时, 上/下限的 REAL 值在结果 V 中设置。ENO 将被设置为下限, QERR 被设置为上限。

8.7 MUL8_P: 最多处理 8 个值的乘法器

8.7.2 MUL8_P 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型
U1	输入 1	REAL	1.0	I
....	
U8	输入 8	REAL	1.0	I
V	结果	REAL	1.0	O

系列： CONVERT

9.1 有关转换块的常规信息

转换块的用途

在 CFC 中，仅可将块输出（源类型）连接到相同类型数据的输入（目标类型）（例如，REAL 输出与 REAL 输入连接）。必须使用转换块才能互连不同类型的数据。块的输入数据与输出数据类型不同，因此，应根据在输出中设置的数据类型来转换输入数据的类型。CFC 块库 ELEMENTA 包含进行这些互连所需的转换块。还有一个用于过程工程应用的具有扩展属性的附加 R_TO_DW 块。

调用 OB

转换块必须在用于评估转换结果的块之前安装在 OB 中。

9.2 R_TO_DW: 转换

9.2.1 R_TO_DW 的描述

对象名 (类型 + 编号)

FC 282

- R_TO_DW 块 I/O (页 544)

功能

块 R_TO_DW 可将 REAL 类型的数字转换为双字 (DW) 类型的数字。可接受介于 0 和 4,294,967,000 之间的 REAL 类型数字。

错误处理

如果值超出上述限定, 将设置 ENO = 0 及下限 (= 0) 或上限 (= 4,294,967,000)。

9.2.2 R_TO_DW 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 543)”。

I/O (参数)	含义	数据类型	默认值	类型
U	将转换的值	REAL	0.0	I
V	转换后的值	DWORD	0	O

系列： OPERATE

10.1 操作员控制块概述

在 ASSET 中集成 FF 设备

本章将介绍操作员控制块，并说明如何用这些块来控制块参数。

操作员控制块的用途

操作员控制块代表 AS 中的块与 OS 中的块之间的操作员控制接口。它提供下列标准解决方案：

操作员如何控制功能块“FB_yyy”的输入参数“W”，以及 AS 程序如何修改该输入参数？

10.1 操作员控制块概述

下图给出了基本的解决方案，其中使用了操作员控制块及其两个部分 OP_AS 和 OP_OS。

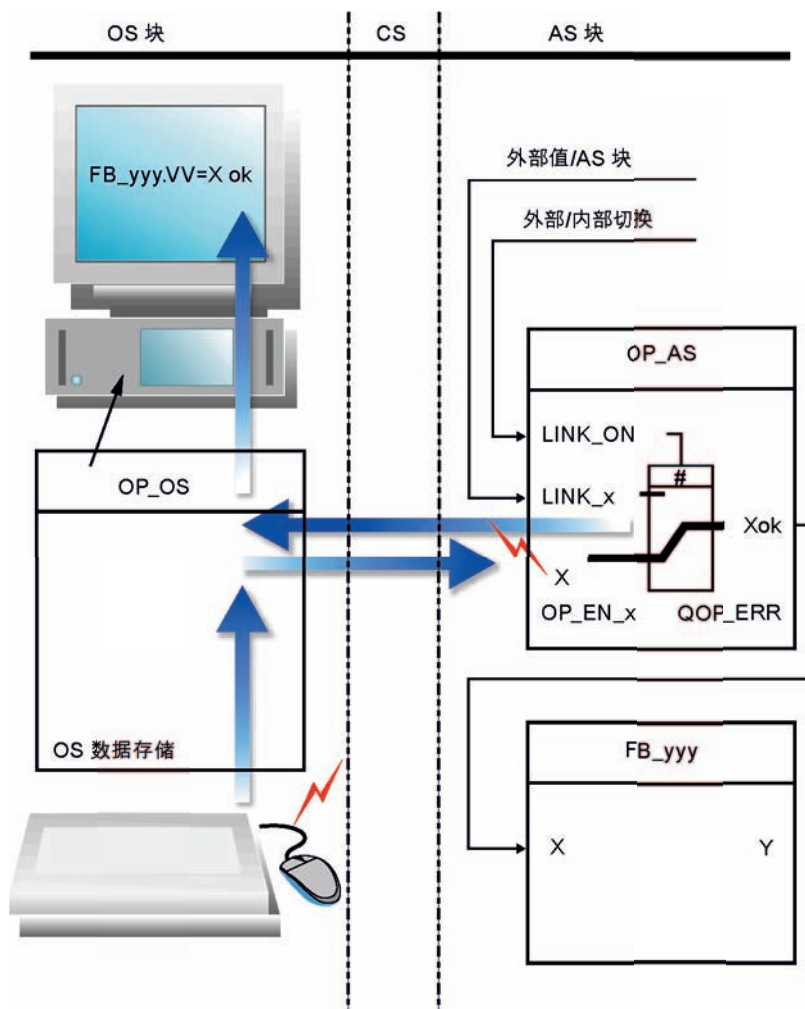


图 10-1 操作员控制的概念

操作员控制的概念

操作员控制块必须在两个位置同时使用:

- 在 AS (AS 块, 缩写为 OP_AS) 中
- 在 OS (OS 块, 缩写为 OP_OS) 中

可通过以下两种不同的方式获得 OP_AS 提供给输出 Xok 的有效值:

- 从外部, 即由其他 AS 块提供 (通过 LINK_x 输入)
- 从内部, 由操作员使用 OS 块 OP_OS 设置

常规操作过程:

- 在 OS 中选择的 OP_OS 查询 OP_AS 值或操作员控制的启用/禁用状态。显示使操作员不断获知 OP_AS 的当前状态。结果在 OS 特定的运行时周期中对 OP_AS 异步显示。
- 操作员控制/修改 OP_OS 的操作员可控制 I/O。其算法会检查输入数据的有效性:
 - 根据情况, 将更正或拒绝无效输入 (块特定)。相应的消息输出给操作员。
 - 更正后的值或有效值传送到 OP_AS, 并在 OS 中记录。
- 由于上次更改 OP_OS 后当前 AS 状态可能已更改, 因此 OP_AS 接收值并执行有效性检查。
 - 根据情况更正或拒绝当前无效的输入 (块特定)。OP_AS 在布尔输出 QOP_ERR 上报告结果, 即输出持续时间等于 OP_AS 采样时间的脉冲。
 - 将输出更正后的值或有效值 (或者, 如果新值被拒绝, 输出原来的有效值), 以便在对应的 OP_AS 的 Xok 输出上供进一步使用。

10.1 操作员控制块概述

概述

下表提供了操作员控制块的概述。这些块以 FB 的方式实现，每个 FB 都需要一个背景数据块，用于每个应用程序。

块名称	含义	操作方法	FB 号
OP_A	模拟值的操作员控制		45
OP_A_LIM	模拟值的操作员控制	限制	46
OP_A_RJC	模拟值的操作员控制	拒绝	47
OP_D	数字值控制，2 个按钮		48
OP_D3	数字值控制，3 个按钮		49
OP_TRIG	数字值控制，按钮功能		50

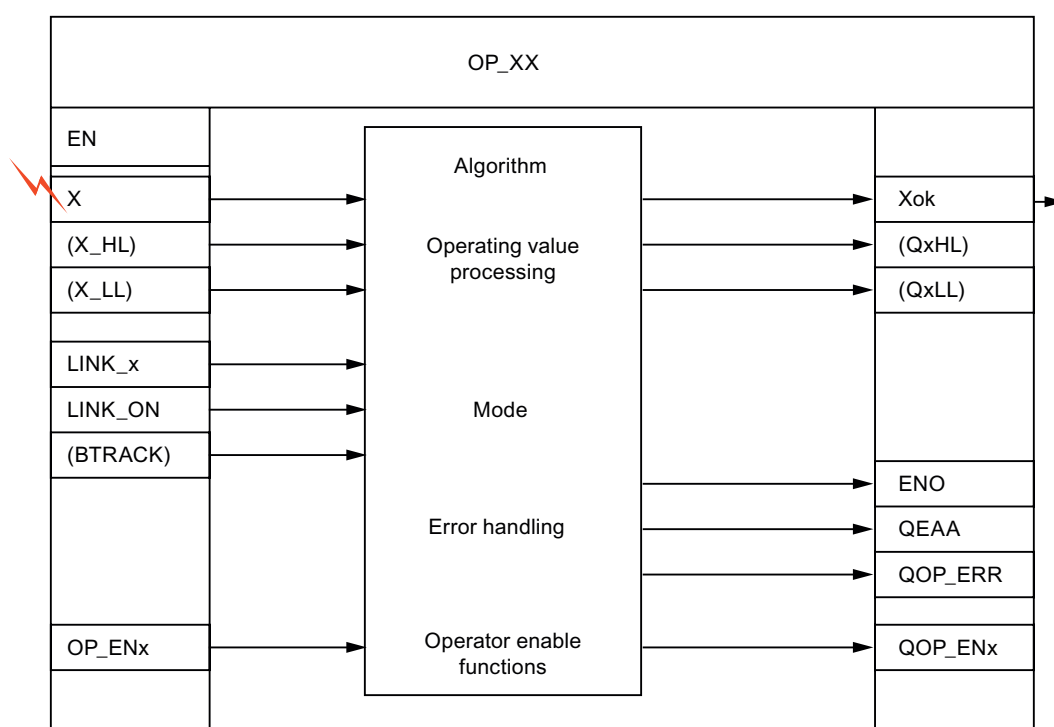
常规 I/O

操作员控制块（请参阅上图）具有用于二进制和模拟值控制的准确定义的 I/O。所有操作员控制块以及仅处理模拟值的操作员控制块（在画面中由括号表示）的 I/O 的重要性均相同。

下面简述这些 I/O 的功能。

输入

- EN 用于设置/复位块算法。
 - EN = 1: 从块所在的 OB 中调用该块。
 - EN = 0: 在 OB 中跳过块调用。
- X (代表操作员所控制的输入的标识符) 由 OS 以 IO 类型的形式写入, 由 AS 块采样并覆盖 (如果适当)。此输入为可追溯的。不可进行互连。
- X_HL 和 X_LL 定义 X 的调整上限和下限 (仅适用于模拟值操作)。根据操作员控制块的类型, 由操作员设置的超出这些限制的值将受到限制或拒绝。



操作员控制块结构 (常规)

- LINK_x 与外部值互连。这将为内部 (由操作员) 提供的输入 X 提供外部选择 (另请参见 LINK_ON)。
- LINK_ON 更改选择要处理的值的模式:
 - LINK_ON = 1: 输入 LINK_x 的值 (通常从其它互连块接收) 将视为外部默认值。
 - LINK_ON = 0: 输入 X 的值 (通常在 OS 输入) 将视为内部默认值 (来自本身的 OP_XX)。

10.1 操作员控制块概述

- 在 LINK_ON = 1 转为 LINK_ON = 0 期间, 使用 BTRACK (如果存在) 可进行无波动切换。
 - BTRACK = 1: LINK_ON = 1 时, 算法会确保操作员控制的输入 X 跟踪 LINK_x 输入。此操作的目的是确保块不处理任何 X 输入的旧操作值, 从而在切换为手动模式 (LINK_ON = 0) 期间更改当前值 (已输出)。
 - BTRACK = 0: LINK_ON = 1 时, 操作员控制的输入 X 不会被覆盖, 因此, 其值通常与 LINK_x 的值不同。切换 LINK_ON = 0 期间, 这些原有值再次生效, 并导致活动 Xok 输出值的相应变化 (称为波动)。
- 使用 OP_EN_x 参数可启用/禁用已分配的输入 X 的操作员控制:
 - OP_EN_x = 1: 启用输入 X 进行操作员控制。
 - OP_EN_x = 0: 操作员控制被阻塞或拒绝。

输出

- ENO 表示结果 Xok 的有效性 (1 = 有效, 0 = 无效)。
- QERR = ENO 取反 (存储在块实例中)。
- QOP_ERR 参数中的逻辑“1”表示操作员输入错误。输出将在一个周期 (采样时间) 后再次复位。
- Xok (有效活动输出的代表标识符)。根据操作员控制块类型和模式, 此输出包含有效输出值。可通过互连到要调整其输入的 AS 块来实现。例如, 根据操作员控制块, 特定标识符为 V 或 Q_1。
- QxHL 和 QxLL 分别表示超出 X 的调整上限和下限 (仅适用于模拟值调整)。根据操作员控制块的类型, 由操作员输入的超出这些限制的值将受到限制或拒绝。
- QOP_ENx 包含传递给 OP_ENx 的输出值。可通过其它 AS 块查询该输出值, 并提供 OP_AS 操作员控制的当前启用/禁用状态信息。

10.2 OP_A: 模拟值操作

10.2.1 OP_A 的描述

对象名 (类型 + 编号)

FB45

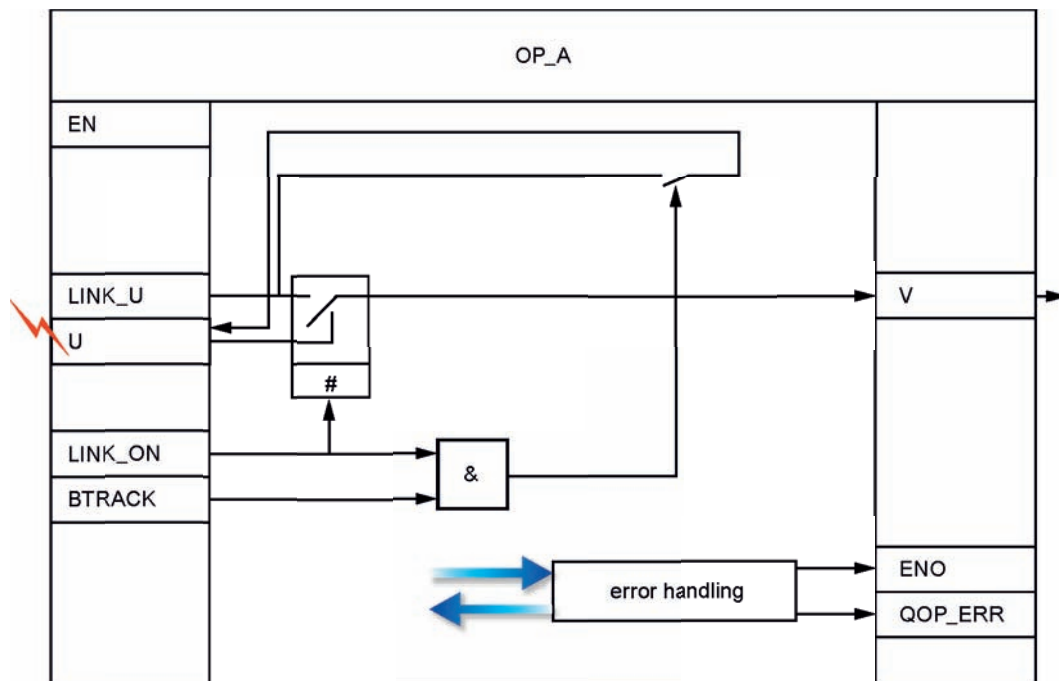
- OP_A 块 I/O (页 553)
- OP_A 块图标 (页 705)
- OP_A 面板 (页 676)

功能

OP_A 块是用于调整 AS 块模拟值的基本操作员控制块, 它对启用监视和操作员控制没有限制。

工作原理

OP_A 块的工作原理如下所示:



OP_A 结构

- U 值由操作员在 OS 上写入。
- LINK_U 被提供外部值（已组态或互连）。
- LINK_ON 启用外部/内部值：
 - LINK_ON = 1: LINK_U 传递给 V。
 - LINK_ON = 0: U 传递给 V。
- BTRACK 允许跟踪操作员控制的输入 U（仅在 LINK_ON = 1 时）。
 - BTRACK = 1: U 跟踪 LINK_U 的值。这可确保在转移到 LINK_ON = 0 期间，输出 V 不会出现波动。
 - BTRACK = 0: U 保留其以前（操作员输入的）值。它将在转移到 LINK_ON = 0 后再次变为活动状态。

调用 OB

操作员控制块必须安装在同一 OB 中，并在评估操作员输入的块之前。

错误处理

存在下列错误指示：

- ENO = 0, 仅限系统指示（块中无特殊处理）
- 如果在设置活动外部值 (LINK_ON = 1) 时在输入 U 中执行更改，那么将 QOP_ERR = 1 设置为持续一个周期的时间。输入 U 保留其先前值（在操作员干预之前）。

OP_A 的错误指示

ENO	QOP_ERR	原因, 反应 (如果适用)
0	X	系统检测的错误 (块中无特殊处理例程)
1	1 (π)	如果 LINK_ON=1, 不允许操作员输入。输入 U 保留其值。

π : 带采样持续时间的脉冲

X: 任意值

时间响应

不适用。

消息响应

不适用。输出 QOP_ERR 可与消息块互连以便报告操作员错误。更多相关信息，可参考“消息块 (页 575)”。

更多信息

更多相关信息，可参考：

OP_A 的 I/O (页 553)

操作和监视 OP_A (页 553)

10.2.2 OP_A 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识：

加粗字体的 I/O 名称表示该 I/O 可见；正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息，可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
BTRACK	1 = 无波动切换	BOOL	1	I	+
LINK_ON	0= 操作员控制激活, 1 = 互连激活	BOOL	0	I	+
LINK_U	U 的可互连输入	REAL	0.0	I	
QOP_ERR	1 = 操作员错误	BOOL	1	O	
U	操作员输入, 模拟值	REAL	0.0	IO	+
V	模拟值	REAL	0.0	O	+

10.2.3 OP_A 的操作员监控

更多信息

更多信息可参考以下部分：

- OP_A 块图标 (页 705)
- OP_A 面板 (页 676)

10.3 OP_A_LIM: 模拟值操作 (限制)

10.3.1 OP_A_LIM 的描述

对象名 (类型 + 编号)

FB46

- OP_A_LIM 块 I/O (页 557)
- OP_A_LIM 块图标 (页 705)
- OP_A_LIM 面板 (页 676)

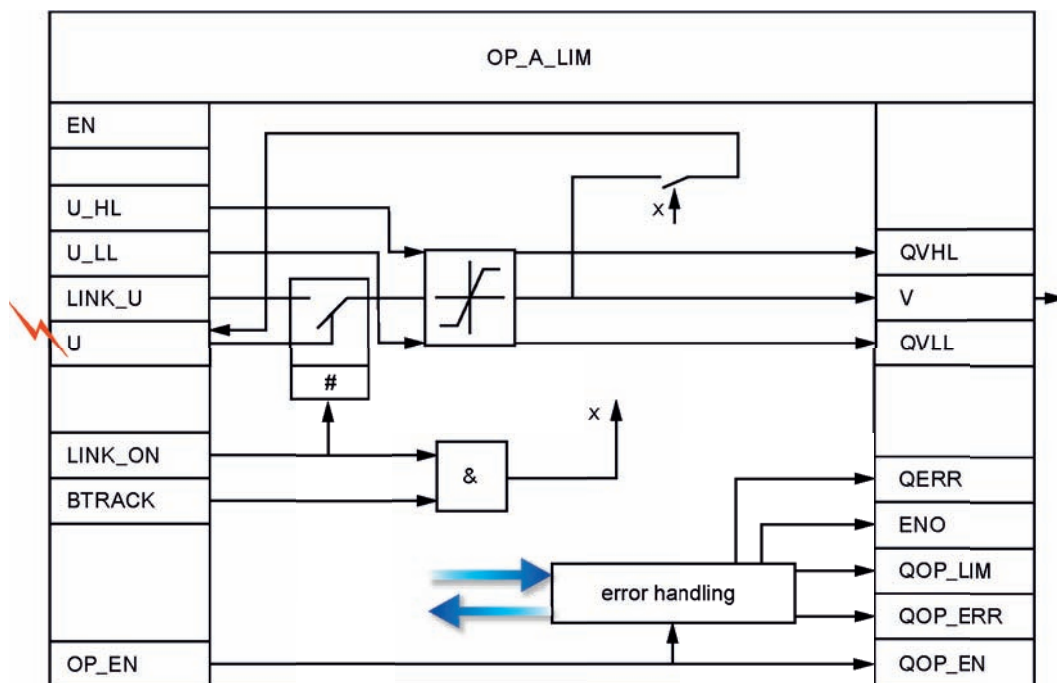
功能

OP_A_LIM (受限制的模拟量操作) 块用于控制块的模拟值。

超出操作限制的操作员控制将限制为相应的超出限制值。可检查已互连或组态的值 (LINK_U), 而非由操作员设置的值 (U) (LINK_ON = 1)。

工作原理

该块的工作原理:



OP_A_LIM 结构

- U 值由操作员在 OS 上写入。操作员控制如下设置：
 - OP_EN = 1 时启用，
 - OP_EN = 0 时禁用
- LINK_U 被提供外部值（已组态或互连）。
- LINK_ON 将受限的外部/内部值传递给 U_LL 或 U_HL：
 - LINK_ON = 1: 限制值 LINK_U 传递给 V
 - LINK_ON = 0: 限制值 U 传递给 V 并写回到输入 U，即在没有操作员输入的情况下，输入 U 可能因限制的修改而更改。
- BTRACK 允许操作员控制的输入 U 跟踪 LINK_U（仅当 LINK_ON = 1 时）
 - BTRACK = 1: 操作员输入 U 跟踪 LINK_U 限制值。这可确保在转移到 LINK_ON = 0 期间，输出 V 不会出现波动。
 - BTRACK = 0: U 保留其以前（操作员输入的）值。它将在转移到 LINK_ON = 0 后再次传递给输出 V。

调用 OB

操作员控制块必须安装在同一 OB 中, 并在使用操作员输入的块之前。

错误处理

显示下列错误:

ENO	QOP_ERR	QOP_LIM	原因, 反应 (如果适用)
0	X	X	系统识别的错误
1	1 (π)	0	如果 LINK_ON=1, 不允许操作员输入。输入 U 保留其值。
1	1 (π)	0	操作员输入在 OS 上启用 (OP_EN=1), 但同时在 AS 上被禁用 (OP_EN=0)。
1	1 (π)	1 (π)	启用的操作员控制超出限制。输入 U 受到限制。

π : 采样持续时间为

X 的脉冲: 任意值

OP_A_LIM 的错误显示 (限制)

QVLL	QVHL	原因
1	0	输入值 < U_LL (输入值 = U 或 LINK_U)
0	1	输入值 > U_HL (输入值 = U 或 LINK_U)

时间响应

不适用。

消息响应

不适用。输出 QOP_ERR 或 QOP_LIM 可与消息块互连以便报告操作员错误。更多相关信息, 可参考“消息块 (页 575)”。

更多信息

更多相关信息, 可参考:

OP_A_LIM 的 I/O (页 557)

操作和监视 OP_A_LIM (页 557)

10.3.2 OP_A_LIM 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
BTRACK	1 = 无波动切换	BOOL	1	I	+
LINK_ON	0= 操作员控制激活, 1 = 互连激活	BOOL	0	I	+
LINK_U	U 的可互连输入	REAL	0.0	I	
OP_EN	1 = 操作员输入已启用	BOOL	1	I	
QERR	1 = 处理错误	BOOL	1	O	+
QOP_EN	1 = 操作员控制启用	BOOL	0	O	+
QOP_ERR	1 = 操作员错误	BOOL	0	O	
QOP_LIM	1 = 操作员错误, 限制	BOOL	0	O	
QVHL	1 = 上限处于激活状态	BOOL	0	O	
QVLL	1 = 下限处于激活状态	BOOL	0	O	
U	操作员输入, 模拟值	REAL	0.0	IO	+
U_HL	上限	REAL	100.0	I	+
U_LL	下限	REAL	0.0	I	+
V	模拟值	REAL	0.0	O	+

10.3.3 OP_A_LIM 的操作员监控

更多信息

更多信息可参考以下部分:

- OP_A_LIM 块图标 (页 705)
- OP_A_LIM 面板 (页 676)

10.4 OP_A_RJC: 模拟值操作 (拒绝)

10.4.1 OP_A_RJC 的描述

对象名 (类型 + 编号)

FB47

- OP_A_RJC 块 I/O (页 561)
- OP_A_RJC 块图标 (页 705)
- OP_A_RJC 面板 (页 677)

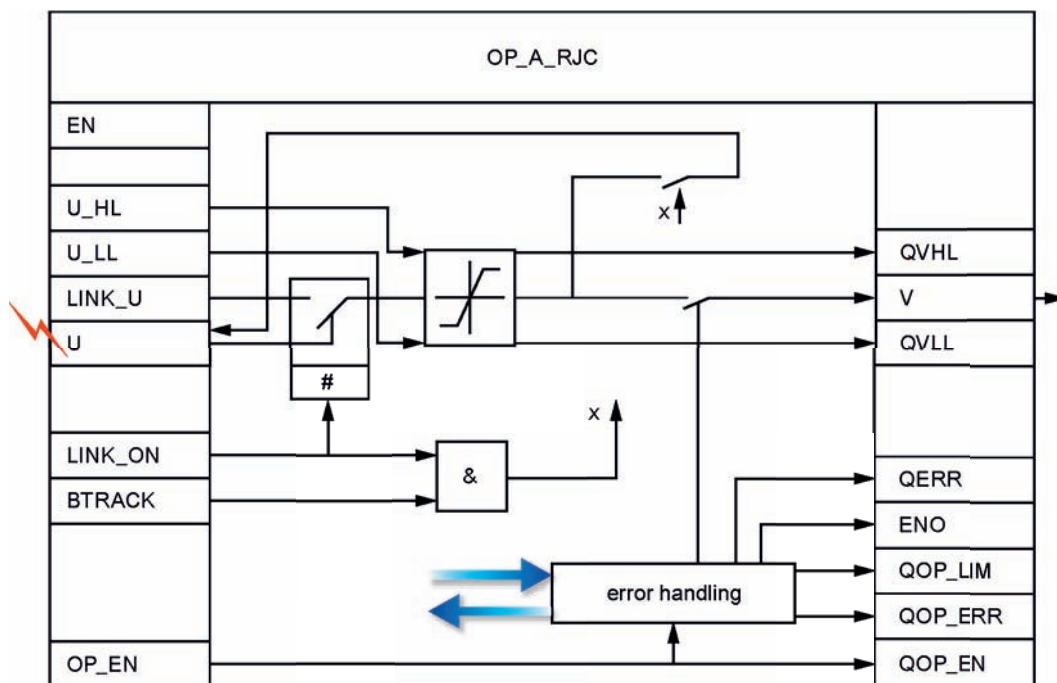
功能

OP_A_RJC (被拒绝的模拟量操作) 块用于控制块的模拟值。

将放弃任何超出操作限制的操作。可检查已互连或组态的值 (LINK_U) 而非输入值 (U) (LINK_ON = 1)。在此情况下, 块根据 OP_A_LIM 限制数值。

工作原理

该块的工作原理:



OP_A_RJC 结构

- U 值由操作员在 OS 上写入。操作员控制如下设置：
 - OP_EN = 1 时启用，
 - OP_EN = 0 时禁用
- LINK_U 分配有外部值（已组态或互连）。
- LINK_ON 将受限的外部/内部值传递给 U_LL 或 U_HL：
 - LINK_ON = 1: 受限制的值 LINK_U 传递给 V。
 - LINK_ON = 0: 旧的（受限制的）U 值传递给 V 并写回到输入 U，即在没有操作员干预的情况下，输入 U 可能由于操作员输入限制的更改而更改。
- BTRACK 允许跟踪操作员控制的输入 U（仅在 LINK_ON = 1 时）。
 - BTRACK = 1: 操作员输入 U 跟踪 LINK_U 限制值。这可确保在转移到 LINK_ON = 0 期间，输出 V 不会出现波动。
 - BTRACK = 0: U 保留其以前（操作员输入的）值。它将在转移到 LINK_ON = 0 后再次传递给输出 V。

调用 OB

操作员控制块必须安装在同一 OB 中, 并在使用操作员输入的块之前。

错误处理

显示下列错误:

ENO	QOP_ERR	QOP_RJC	原因, 反应 (如果适用)
0	X	X	系统识别的错误
1	1 (π)	0	如果 LINK_ON=1, 不允许操作员输入。输入 U 保留其值。
1	1 (π)	0	操作员输入在 OS 上启用 (OP_EN=1), 但同时在 AS 上被禁用 (OP_EN=0)。
1	1 (π)	1 (π)	启用的操作员控制超出限制。 操作员输入被放弃。

π : 带采样持续时间的脉冲

X: 任意值

OP_A_RJC 的错误显示 (仅在 LINK_ON=1 时限制)

QVLL	QVHL	原因
1	0	输入值 < U_LL (输入值 = LINK_U)
0	1	输入值 > U_HL (输入值 = LINK_U)

时间响应

不适用。

消息响应

不适用。 输出 QOP_ERR 或 QOP_RJC 可与消息块互连以便报告操作员错误。 更多相关信息, 可参考“消息块 (页 575)”。

更多信息

更多相关信息, 可参考:

OP_A_RJC 的 I/O (页 561)

操作和监视 OP_A_RJC (页 561)

10.4.2 OP_A_RJC 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
BTRACK	1 = 无波动切换	BOOL	1	I	+
LINK_ON	0= 操作员控制激活, 1 = 互连激活	BOOL	0	I	+
LINK_U	U 的可互连输入	REAL	0.0	I	
OP_EN	1 = 操作员控制启用	BOOL	1	I	
QERR	1 = 处理错误	BOOL	1	O	+
QOP_EN	1 = 操作员控制启用	BOOL	0	O	+
QOP_ERR	1 = 操作员错误, 拒绝	BOOL	0	O	
QOP_RJC	1 = 操作员错误, 拒绝	BOOL	0	O	
QVHL	1 = 上限处于激活状态	BOOL	0	O	
QVLL	1 = 下限处于激活状态	BOOL	0	O	
U	操作员输入, 模拟值	REAL	0.0	IO	+
U_HL	上限	REAL	100.0	I	+
U_LL	下限	REAL	0.0	I	+
V	模拟值	REAL	0.0	O	+

10.4.3 OP_A_RJC 的操作员监控

更多信息

更多信息, 请参考以下部分:

- OP_A_RJC 块图标 (页 705)
- OP_A_RJC 面板 (页 677)

10.5 OP_D: 数字值的操作员输入 (双按钮控制)

10.5.1 OP_D 的描述

对象名 (类型 + 编号)

FB48

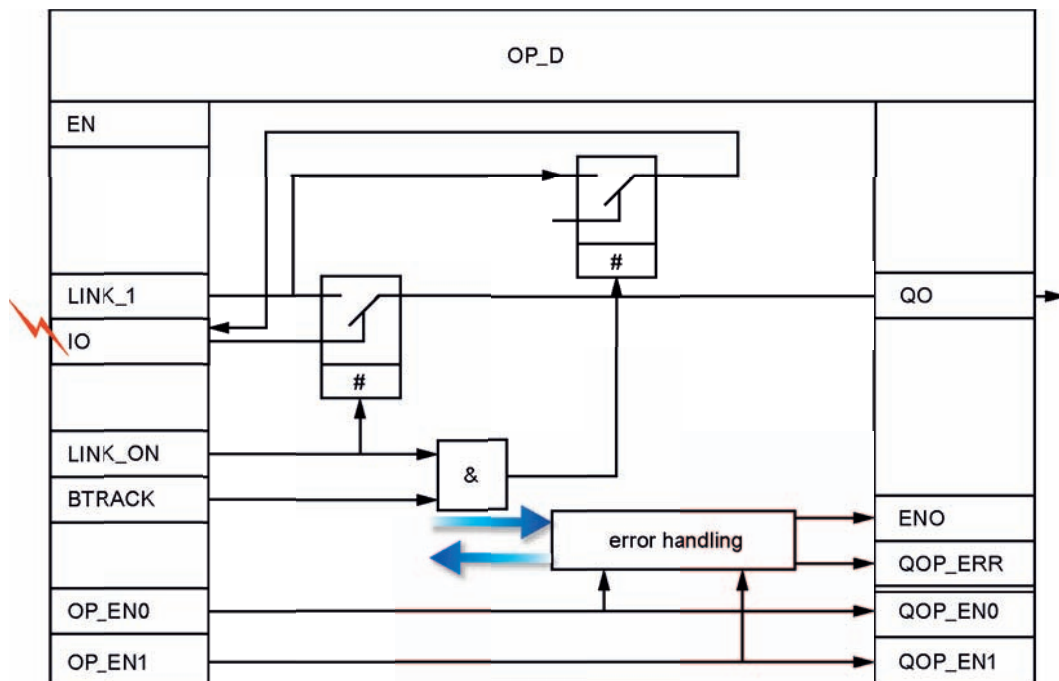
- OP_D 块 I/O (页 565)
- OP_D 块图标 (页 706)
- OP_D 面板 (页 677)

功能

操作员控制块 OP_D 用于通过两个按钮控制块的数字值。如果操作员输入有效值，它将输出到 Q 输出。

工作原理

该块的工作原理:



OP_D 结构

- I0 由操作员在 OS 写入。它可通过两个独立的输入启用：
 - OP_EN0 = 1 (对于设置“0”)，
 - OP_EN1 = 1 (对于设置“1”)。
- LINK_I 提供有已组态或互连的外部值。
- LINK_ON 启用外部/内部值：
 - LINK_ON = 1: LINK_U 传递给 Q0，
 - LINK_ON = 0: 输入的 I0 值传递给 Q0。
- BTRACK 允许操作员控制的输入 I0 跟踪 LINK_I (仅当 LINK_ON = 1 时)：
 - BTRACK = 1: 操作员输入 I0 跟踪 LINK_I。这可确保在 LINK_ON = 0 转移期间输出 Q0 不会出现波动。
 - BTRACK = 0: I0 保留其上一个 (输入的) 值。它将在转移到 LINK_ON = 0 后再次传递给输出 Q0。

调用 OB

操作员控制块必须安装在同一 OB 中，并在使用操作员输入的块之前。

错误处理

显示下列错误：

ENO	QOP_ERR	原因, 反应 (如果适用)
0	X	系统检测的错误 (块中无特殊处理例程)
1	1 (π)	LINK_ON=1 时, 不启用或执行操作员控制。输入 I0 保留其值。

π : 带采样持续时间的脉冲

X: 任意值

时间响应

不适用。

10.5 OP_D: 数字值的操作员输入 (双按钮控制)

消息响应

不适用。输出 QOP_ERR 可与消息块互连以便报告操作员错误。更多相关信息,可参考“消息块 (页 575)”。

更多信息

更多相关信息,可参考:

OP_D 的 I/O (页 565)

操作和监视 OP_D (页 565)

10.5.2 OP_D 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
BTRACK	1 = 无波动切换	BOOL	1	I	+
I0	操作员输入 0	BOOL	0	IO	+
LINK_I	I 的可互连输入	BOOL	0	I	
LINK_ON	0= 操作员控制激活, 1 = 互连激活	BOOL	0	I	+
OP_EN0	禁用操作员控制使能输入	BOOL	1	I	
OP_EN1	激活操作员控制使能输入	BOOL	1	I	
Q0	二进制输出	BOOL	0	O	+
QOP_EN0	禁用操作员控制使能输出	BOOL	0	O	+
QOP_EN1	激活操作员控制使能输出	BOOL	0	O	+
QOP_ERR	1 = 操作员错误	BOOL	0	O	

10.5.3 OP_D 的操作员监控

更多信息

更多信息, 请参考以下部分:

- OP_D 块图标 (页 706)
- OP_D 面板 (页 677)

10.6 OP_D3: 数字值的操作员输入 (3 个按钮控制)

10.6.1 OP_D3 的描述

对象名 (类型 + 编号)

FB49

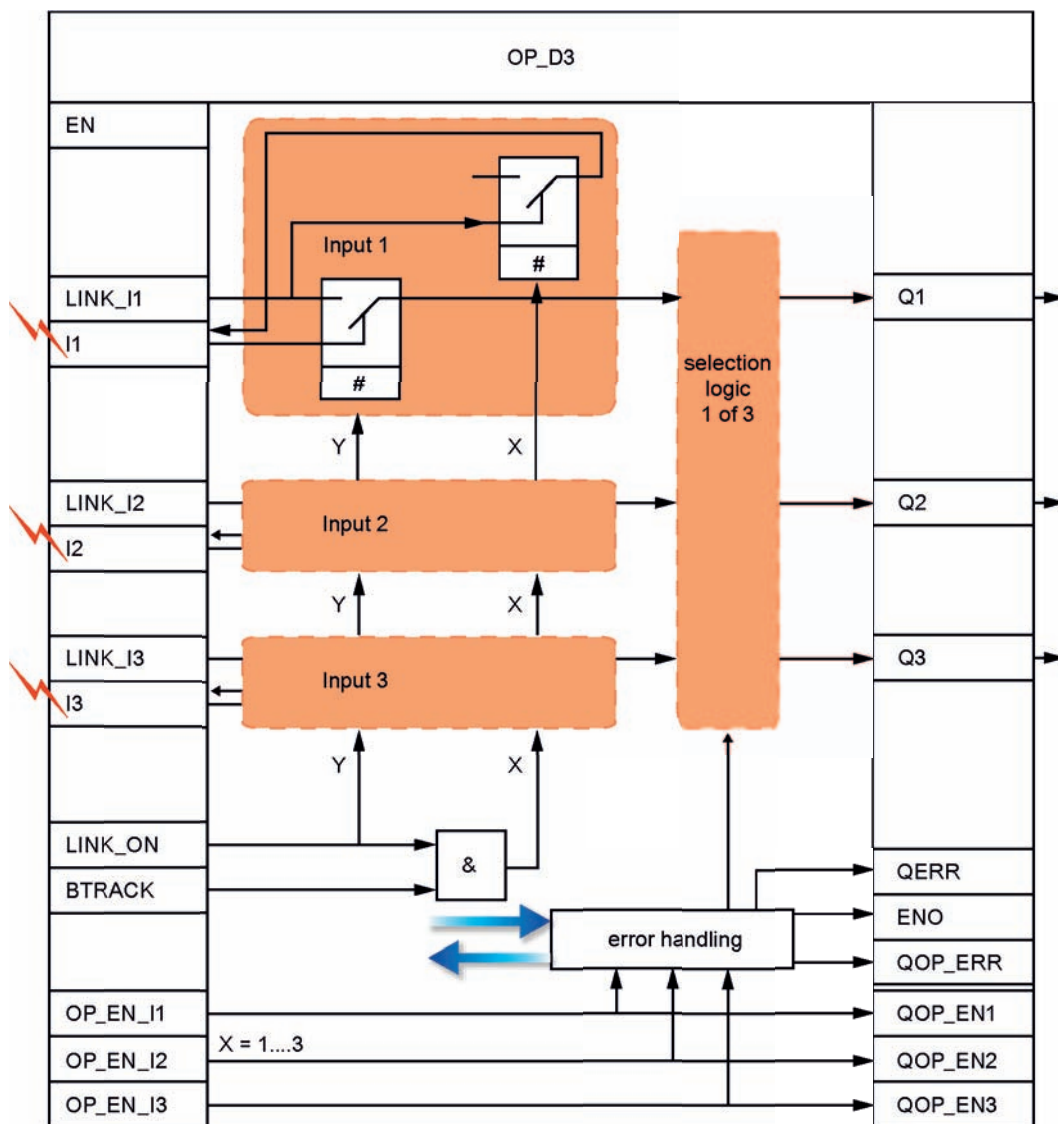
- OP_D3 块 I/O (页 570)
- OP_D3 块图标 (页 706)
- OP_D3 面板 (页 678)

功能

操作员控制块 OP_D3 用于执行三者之一的逻辑数字值操作。如果设置了三个操作员输入 I1、I2 或 I3 中的一个, 则只要允许操作员输入, 对应的输出就会设置为 1 并复位其它输出。

工作原理

该块的工作原理如下所示：这里使用表达式 $x = 1..3$ 作为相关三个输入/输出的索引：



OP_D3 结构

- OS 控制系统同时设置输入 I1、I2 以及 I3 (“1”表示要激活的输入，“0”表示其它两个输入)。使用三个单独的输入启用/禁用操作员控制：
 - OP_EN_Ix = 1: 启用输入 Ix 的操作员控制
 - OP_EN_Ix = 0: 禁用输入 Ix 的操作员控制
- 每个 LINK_Ix 都提供有已组态或互连的外部值。

10.6 OP_D3: 数字值的操作员输入 (3 个按钮控制)

- LINK_ON 启用外部/内部值:
 - LINK_ON = 1: 处理 LINK_Ix 并将其传递给 Qx。
 - LINK_ON = 0: 处理操作员控制的 Ix 输入并将其传递给 Qx。
- BTRACK 允许跟踪操作员控制的输入 Ix (仅在 LINK_ON = 1 时):
 - BTRACK = 1: 将操作员控制的输入 Ix 跟踪至 LINK_Ix。这可确保在 LINK_ON = 0 转移期间输出 Qx 不会出现波动。
 - BTRACK = 0: Ix 保留其上一个 (操作) 值。它将在转移到 LINK_ON = 0 后再次传递给输出 Qx。
- 选择逻辑按 x = 1、2、3 的顺序采用三个输入值 (Ix 或 LINK_Ix)，并记住逻辑状态为“1”的输入的最高索引“x”。将设置由索引指定的输出 Qx (“1”)，同时复位其余两个输出 Qx (“0”)。如果所有三个输入 I1 = I2 = I3 = 0，输出将不会更改。

调用 OB

操作员控制块必须安装在同一 OB 中，并在使用操作员输入的块之前。

错误处理

显示下列错误:

ENO	QOP_ERR	原因, 反应 (如果适用)
0	X	系统识别的错误
0	0	所有输入均为“0”或不止一个输入为“1”。根据设置的最高输入来设置输出。 更多相关信息, 可参考“工作原理”和“选择逻辑”部分。
0	1	不止一个输入为“1”。根据设置的最高输入来设置输出。 更多相关信息, 可参考“工作原理”和“选择逻辑”部分。 错误处理例程还将根据以下规则更改输入 Ix: “具有最高索引“x”的输入 Ix 保持置位, 而其余输入将被复位”。

时间响应

不适用。

消息响应

不适用。输出 QOP_ERR 可与消息块互连以便报告操作员错误。更多信息, 可参考“消息块 (页 575)”。

更多信息

更多信息, 可参考:

OP_D3 的 I/O (页 570)

操作和监视 OP_D3 (页 571)

10.6 OP_D3: 数字值的操作员输入 (3 个按钮控制)

10.6.2 OP_D3 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
BTRACK	1 = 无波动切换	BOOL	1	I	+
I1	操作员输入 1	BOOL	0	IO	+
I2	操作员输入 2	BOOL	0	IO	+
I3	操作员输入 3	BOOL	1	IO	+
LINK_I1	I1 的可互连输入	BOOL	0	I	
LINK_I2	I2 的可互连输入	BOOL	0	I	
LINK_I3	I3 的可互连输入	BOOL	0	I	
LINK_ON	0= 操作员控制激活, 1 = 互连激活	BOOL	0	I	+
OP_EN_I1	操作员控制使能输入 1	BOOL	1	I	
OP_EN_I2	操作员控制使能输入 2	BOOL	1	I	
OP_EN_I3	操作员控制使能输入 3	BOOL	1	I	
Q1	二进制输出 1 (SWITCH1)	BOOL	0	O	+
Q2	二进制输出 2 (SWITCH2)	BOOL	0	O	+
Q3	二进制输出 3 (SWITCH3)	BOOL	1	O	+
QERR	1 = 处理错误	BOOL	1	O	+
QOP_EN1	操作员控制使能输出 1	BOOL	0	O	+
QOP_EN2	操作员控制使能输出 2	BOOL	0	O	+
QOP_EN3	操作员控制使能输出 3	BOOL	0	O	+
QOP_ERR	1 = 操作员错误	BOOL	0	O	

10.6.3 OP_D3 的操作员监控

更多信息

更多信息, 请参考以下部分:

- OP_D3 块图标 (页 706)
- OP_D3 面板 (页 678)

10.7 OP_TRIG: 数字值的操作员输入 (单按钮控制)

10.7.1 OP_TRIG 的描述

对象名 (类型 + 编号)

FB50

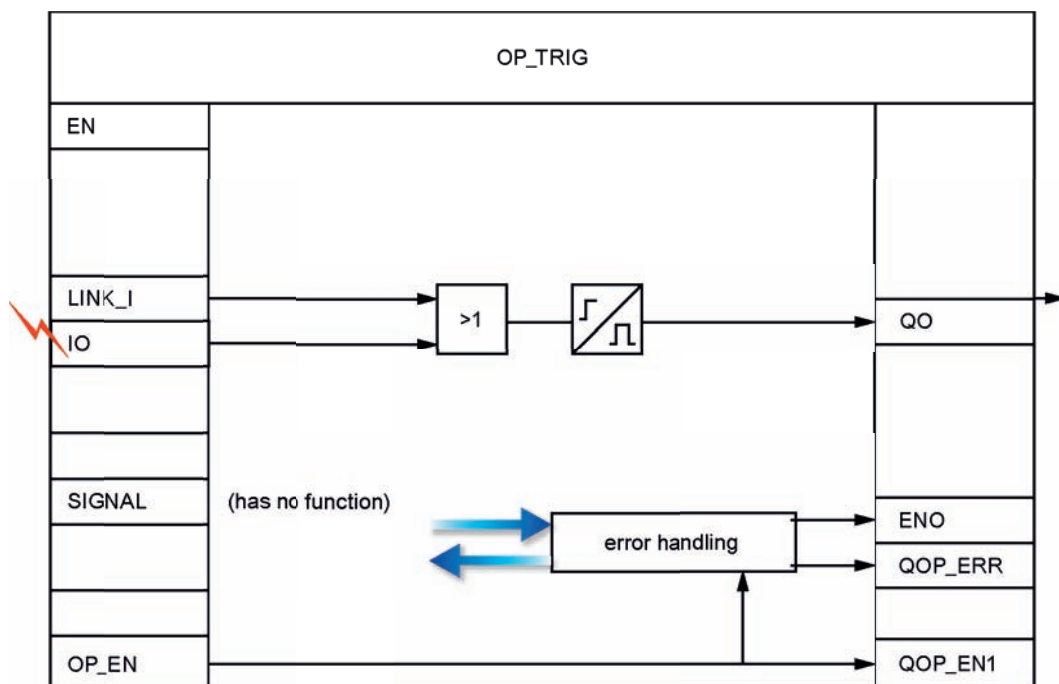
- 块 I/O (页 574)
- OP_TRIG 块图标 (页 707)
- OP_TRIG 面板 (页 679)

功能

操作员控制块 OP_TRIG 用于实现单按钮控制 (相当于 RESET 按钮)。

工作原理

该块的工作原理:



OP_TRIG 结构

- 操作员将值 1 写入 I0 (如果通过设置 OP_EN = 1 允许此操作)。输出 Q0 在一个周期内 (采样时间) 被设置为 1, 然后再次复位。处理后, 操作员输入 I0 由操作员控制块复位。
- 可互连输入 (LINK_I) 与操作员输入构成冗余。正沿时, 将在输出 Q0 中设置一个周期 (采样时间) 时间段的逻辑“1”, 然后复位。LINK_I 对操作员控制使能 QOP_EN 无任何影响。
- 该块具有一个显示在 OS 中的可互连输入 (SIGNAL)。它无任何功能, 仅用于 OS 显示。在此建议互连要复位的信号, 因为它对仅在一个周期内设置的块输出信号 Q0 的使用没有任何意义。

调用 OB

操作员控制块必须安装在同一 OB 中, 并在使用操作员输入的块之前。

错误处理

显示下列错误:

ENO	QOP_ERR	原因, 反应 (如果适用)
0	X	系统检测的错误 (块中无特殊处理例程)
1	1 (π)	未启用操作员控制。输入 I0 设置为“0”。

π : 采样持续时间为

X 的脉冲: 任意值

时间响应

不适用。

消息响应

不适用。输出 QOP_ERR 可与消息块互连以便报告操作员错误。更多相关信息, 可参考“消息块 (页 575)”。

更多信息

更多相关信息, 可参考:

OP_TRIG 的 I/O (页 574)

操作和监视 OP_TRIG (页 574)

10.7 OP_TRIG: 数字值的操作员输入 (单按钮控制)

10.7.2 OP_TRIG 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM	允许值
I0	操作员输入	BOOL	0	IO	+	
LINK_I	可与 I0 互连的输入	BOOL	0	I		
OP_EN	1 = 操作员控制启用	BOOL	1	I		
Q0	二进制输出	BOOL	0	O		
QOP_EN	1 = 操作员控制启用	BOOL	0	O	+	
QOP_ERR	1 = 操作员错误	BOOL	0	O		
SIGNAL	用于在 OS 显示的可互连输入	BOOL	0	I	+	

10.7.3 OP_TRIG 的操作员监控

更多信息

更多信息可参考以下部分:

- OP_TRIG 块图标 (页 707)
- OP_TRIG 面板 (页 679)

系列： MESSAGE

11.1 消息块概述

消息块的用途

操作员可能需要涉及数字值更改或 AS 中状态的事件信息。由于消息块在 AS 中实现，因此 OS 系统无需为获得此信息而轮询 AS。这些块监视数字值并将更改报告给 OS（包括附加的和可组态的信息）。OS 系统可显示、记录以及归档这些信息。

下表对作为 FB 实现的消息块进行了概述。

消息块概述

对象名称	类型名称	含义	操作方法
43	MESSAGE	生成可组态的消息	SIMATIC 过程控制 - 标准
59	MSG_NACK	生成用户特定的无需确认的消息	SIMATIC 过程控制 - 标准

修改单独块的消息将在“消息响应”中介绍。

11.2 MSG_NACK: 用户特定消息 (不需要确认)

11.2.1 MSG_NACK 的描述

对象名 (类型 + 编号)

FB 78

- MSG_NACK 块 I/O (页 578)

应用领域

MSG_NACK 块用于生成无需确认的用户特定消息 (运行消息)。

功能

该块可生成多达八个此类型的用户特定消息。

工作原理

无需确认的消息通过 NOTIFY_8P 输出。可阻止单个或所有消息的输出。

- I_1 到 I_8: 报告这些受监视信号的更改。将可组态消息文本分配给所有这些信号。可对此文本进行修改,并随后在 OS 组态中重新使用它。这些输入的每个更改都会导致将消息发送到 OS (除非禁用了此功能)。
- MSG_LOCK: 允许对此块的消息输出进行过程特定锁定。在锁定信号的正沿,复位所有激活的过程消息,然后将其作为已退出状态发送给 OS。
- AUX_PR01 至 AUX_PR10: 这些输入可互连到任意数据类型的任何值。这些值称为关联值,其长度被限定为 16 个字符,并包含在发送到 OS 的消息中,从而提供触发消息的事件的更详细信息。
- 操作员可在 OS 中锁定过程标签的消息。OS 将此状态报告给相应的消息块,然后由该消息块返回确认 (通过其 NOTIFY_8P) 给 OS。然后在 OS 的消息事件日志中以已确认和已退出状态输入该消息。
- QMSG_SUP 表示启用了消息抑制。
- OS 评估 MSG_STAT、QMSG_ERR 以及 MSG_ACK。

调用 OB

消息块必须安装在产生报告的 OB (例如, OB 35) 中以及 OB 100 中。

错误处理

块 MSG_NACK 的错误处理限定为对 ALARM_8P 的错误信息进行判断。有关详细信息, 可参考手册《S7-300/400 系统软件 - 系统和标准功能》(*System Software for S7-300/400 - System and Standard Functions*)。

有关 MSG_STAT 参数的错误消息的信息, 可参考 NOTIFY_8P; STATUS 参数的在线帮助。

启动特征

启动时, 消息块将抑制所有消息。消息抑制的持续时间 (周期数) 在 RUNUPCYC 参数中设置。每次执行该块时, 重新启动期间使用此参数值初始化的内部计数器会递减。如果计数器值不等于零, 则不生成消息。

消息

消息通过 NOTIFY_8P (SFB 31) 生成。NOTIFY_8P 分配有 8 个数字输入和 10 个关联值。每次在一个或多个数字输入处检测到沿跳变时均会触发消息。评估脉冲沿时始终会将关联值分配给消息。所有八个信号均分配有通用消息号 (MSG_ID), 它将在 OS 上细分为 8 条消息。ES 通过调用消息服务器自动分配消息号。

消息文本

每个块消息均有默认的消息文本, 并且被分配给一个内部或外部块参数以及消息类别 (操作消息 - 无确认)。可将消息文本当作部分组态进行更改。

关联值

可将关联值以不同的编号和顺序分配给每条消息。未在块算法中使用的关联值可在块中作为输入参数 AUX_PRx 随意互连。

允许的关联值数据类型: BOOL、BYTE、WORD、DWORD、CHAR、INT、DINT、REAL 和 ARRAY OF BYTE

11.2.2 MSG_NACK 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
AUX_PRxx	(xx = 01 - 10) 关联值 1 - 10	*1)	0	IO	
EV_ID	消息号 NOTIFY_8P (由 ES 分配)	DWORD	0	I	
I_x	(x = 1 - 8) 输入 1 - 8	BOOL	0	I	
EN_I_x	(x = 1 - 8) 1 = 启用消息 1 - 8	BOOL	1	I	
MSG_LOCK	锁定所有消息	BOOL	0	I	
MSG_STAT	NOTIFY_8P 的状态输出	WORD	0	O	
OOS	预留	BOOL	0	I	+
QERR	1 = 处理错误	BOOL	1	O	
QMSG_ERR	NOTIFY_8P 的错误输出	BOOL	0	O	
QEN_I_x	(x = 1 - 8) 1 = 启用消息 1 - 8	BOOL	0	O	
QMSG_SUP	通过操作员控制进行的过程消息抑制处于激活状态	BOOL	0	O	
RUNUPCYC	起动周期数	INT	3	I	

*1) 关联值分配给 NOTIFY_8P。允许的关联值数据类型: BOOL、BYTE、WORD、DWORD、CHAR、INT、DINT、REAL 和 ARRAY OF BYTE

11.3 MESSAGE: 消息块 (可组态消息)

11.3.1 MESSAGE 的描述

对象名 (类型 + 编号)

FB43

- MESSAGE 块 I/O (页 581)

功能

MESSAGE 块用于生成可组态 (需要确认) 消息。它构成了要报告其更改的块输出和 S7 块 ALARM_8P 之间的接口。

工作原理

可使用块输入将单条消息分配给受监视的信号, 还可根据过程状态启用或禁用消息。

- I_1 到 I_8: 报告这些受监视信号的更改。将可组态消息文本 (24 个字符) 分配给这些信号的每一个。可对此文本进行修改, 并随后在 OS 组态中重新使用它。这些输入的每个更改都会导致将消息发送到 OS (除非禁用了此功能)。
- I_1ISCSF 到 I_8ISCSF: 通过将参数设置为“1”, 就可以像处理控制系统消息 (CSF) 一样在块内处理关联消息。通过此参数, 可指定相应输入 (I_1...I_8) 的消息是由输入参数 MSG_LOCK=1 (I_x = 0) 锁定还是不锁定 (I_x = 1)。
- MSG_LOCK: 允许对此块的消息输出进行过程特定锁定。在锁定信号的正沿, 将复位所有激活的过程消息 (不是控制系统消息), 并且向 OS 报告已退出状态。
- AUX_PR01 至 AUX_PR10: 这些输入可互连到任何数据类型的任何值。这些值称为关联值, 限制为 16 个字符, 并包含在输出到 OS 的消息中。可以使用这些值详细描述消息的原因。
- 操作员可在 OS 中锁定过程标签的消息。OS 将此状态报告给相应的消息块。该块将块的确认 (通过它的 ALARM_8P) 返回到 OS。然后在 OS 的消息事件日志中输入该消息, 状态为确认并已退出。
- QMSG_SUP 表示消息正在被抑制。
- OS 系统评估 MSG_STAT、QMSG_ERR 以及 MSG_ACK。

11.3 MESSAGE: 消息块 (可组态消息)

调用 OB

该块必须安装在将在其中检测要监视事件的 OB (例如, OB32) 以及 OB100 中。

错误处理

MESSAGE 块的错误处理限定为评估 ALARM_8P 的错误信息。有关详细信息, 可参考手册《S7-300/400 系统软件 - 系统和标准功能》(*System Software for S7-300/400 - System and Standard Functions*)。

有关 MSG_STAT 参数的错误消息的信息, 可参考 NOTIFY_8P; STATUS 参数的在线帮助。

启动特征

启动期间消息块抑制所有消息, 其中包括控制系统消息。消息抑制的持续时间 (周期数) 在 RUNUPCYC 参数中设置。每次执行该块时, 重新启动期间使用此参数值初始化的内部计数器会递减。如果计数器值不等于零, 则不生成消息。

消息

消息通过 ALARM_8P (SFB35) 生成。所有块都使用 PMC 通讯通道。ALARM_8P 分配有 8 个数字输入和 10 个关联值。一个或多个数字输入中的每个检测到的沿转变均会触发消息, 而与确认与否无关。关联值在评估脉冲沿时一致地分配给消息。所有 8 个信号均分配有通用消息号 (MSG_ID), 它将在 OS 上细分为 8 条消息。ES 通过调用消息服务器自动分配消息号。

消息文本

每条块消息均有默认的消息文本, 并且分配给一个内部或外部块参数以及一个特定的消息类别。在组态期间可更改消息文本和消息类别。消息类别中的更改不影响块算法。

更多信息

更多相关信息, 可参考:

MESSAGE 的 I/O (页 581)

MESSAGE 的消息文本和关联值 (页 582)

11.3.2 MESSAGE 的 I/O

CFC 中块显示的出厂设置在“I/O”列中标识:

加粗字体的 I/O 名称表示该 I/O 可见; 正常字体的 I/O 名称表示该 I/O 不可见。

有关所使用缩写的说明和信息, 可参考“关于块描述的常规信息 (页 17)”。

I/O (参数)	含义	数据类型	默认值	类型	OCM
AUX_PRxx	(xx = 01 - 10) 关联值 1 - 10	*1)	0	IO	
EV_ID	消息号 ALARM_8P (由 ES 分配)	DWORD	0	I	
I_x	(x = 1 - 8) 输入 1 - 8	BOOL	0	I	
I_xISCSF	(x = 1 - 8) 1 = 控制系统消息	BOOL	0	I	
MSG_ACK	ALARM_8P 的 ACK_STATE 输出	WORD	0	O	
MSG_LOCK	根据过程锁定消息	BOOL	0	I	
MSG_STAT	ALARM_8P 的 STATUS 输出	WORD	0	O	
OOS	预留	BOOL	0	I	+
QERR	1 = 处理错误	BOOL	1	O	
QMSG_ERR	ALARM_8P 的 ERROR 输出	BOOL	0	O	
QMSG_SUP	通过操作员控制进行的过程消息抑制处于激活状态	BOOL	0	O	
RUNUPCYC	起动车周期数	INT	3	I	

*1) 关联值允许使用下列数据类型: BOOL、BYTE、WORD、DWORD、CHAR、INT、DINT、REAL 和 ARRAY OF BYTE

11.3.3 MESSAGE 的消息文本和关联值

消息文本和消息类别到块参数的分配

消息号	块参数	默认消息文本	消息类别
1	I1	TEXT 1	F
:	:	:	:
8	I8	TEXT 8	F

关联值

可将关联值以不同的编号和顺序分配给每条消息。未在块算法中使用的关联值可在块中作为输入参数 AUX_PRx 随意互连。

允许的关联值数据类型: BOOL、BYTE、WORD、DWORD、CHAR、INT、DINT、REAL 和 ARRAY OF BYTE

面板

12.1 面板：设备块

12.1.1 CTRL_PID（所有视图）

总览

CTRL_PID: 标准视图 (页 584)

CTRL_PID: 维护视图 (页 586)

CTRL_PID: 参数视图 (页 588)

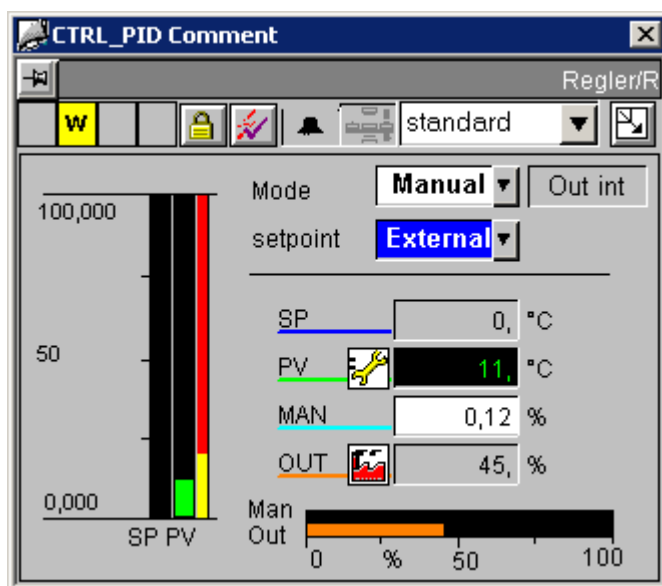
CTRL_PID: 限制视图 (页 590)

全局视图: 消息视图 (页 656)

全局视图: 批生产视图 (页 658)

全局视图: 趋势视图 (页 659)

12.1.2 CTRL_PID：标准视图



模拟显示和数字格式

所有模拟显示均通过“AdvancedAnalogDisplay”对象实现。数字格式通过块符号的“Format_InputValue”和“Format_OutputValue”属性定义。

控制权限

此视图具有以下 2 个用于输入设定值和调节变量的“权限”对象，因为这些变量的控制权限取决于各种因素：

- “Permission_Setpoint”
- “Permission_Manual”

除 WinCC 授权级别外，权限对象还评估下列参数：

权限对象	参数
“Permission_Setpoint”	“Q_SP_OP = TRUE”
“Permission_Manual”	“QLMNOP = TRUE”

操作 PID 调节器与调节

可在参数视图对 PID 调节器进行操作（调节开启/关闭）。

在参数视图中开启了调节时，标准视图会在“手动/自动”(Manual/Auto) 模式组合框上方显示一个组合框。也可使用此组合框再次从标准视图中关闭调节。当设置了“调节开启”(Tuning On) 时，将会禁用控制器的所有其它功能。

与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能		
Manual_COMBOBOX	-->	操作员控制使能		
External_COMBOBOX	-->	操作员控制使能		
Permission_Setpoint	-->	Level_Source	-->	Level_Target
Permission_Manual	-->	Level_Source		
Permission_Setpoint	-->	Target_Operator- ControlEnable		
Setpoint_AnalogValue	-->	操作员控制使能		
Permission_Manual	-->	Target_Operator- ControlEnable		
Manual_AnalogValue	-->	操作员控制使能		
格式	-->	Format_InputValue		
Setpoint_AnalogValue	-->	格式		
ProcessValue_AnalogValue	-->	格式		
格式	-->	Format_OutputValue		
Manual_AnalogValue	-->	格式		
Output_AnalogValue	-->	格式		

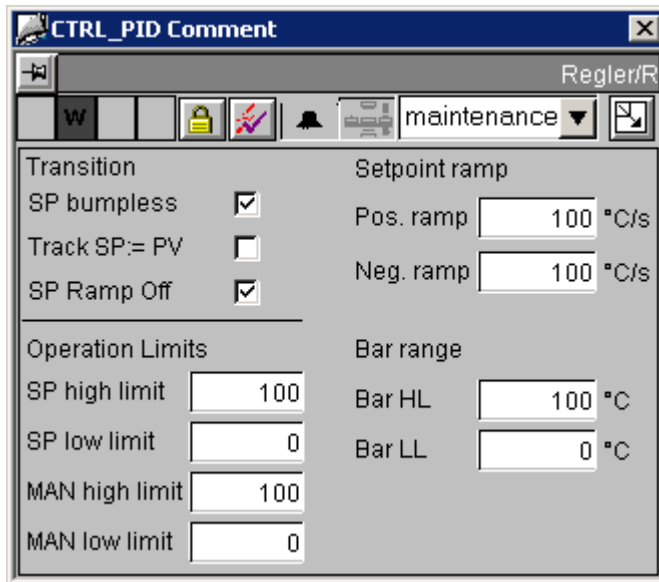
更多信息


更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.3 CTRL_PID：维护视图



如果在控制器图表中实例化一个 CPM 类型的块，且此块具有扩展名为“_cpm”的控制器块名称，则会显示一个附加按钮 ，可通过此按钮调用 CPM 面板。

控制权限

“Permission_SP_Bumpless”对象评估 WinCC 权限级别和“OPTI_EN = FALSE”参数。

与可控制对象直接连接的顺序和定位

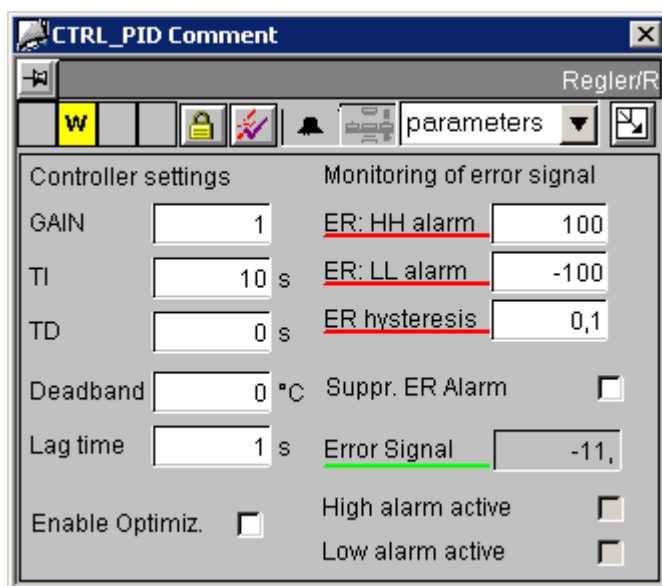
@Level6	-->	操作员控制使能
Permission_SP_Bumpless	-->	Level_Source
Permission_SP_Bumpless	-->	Target_Operator-ControlEnable
Bumpless_CHECKBOX_L	-->	操作员控制使能
SP_TRK_ON_CHECKBOX_L	-->	操作员控制使能
SPRAMP_OFF_CHECKBOX_L	-->	操作员控制使能
SPHighLimit_AnalogValue	-->	操作员控制使能
SPLowLimit_AnalogValue	-->	操作员控制使能
ManHighLimit_AnalogValue	-->	操作员控制使能
ManLowLimit_AnalogValue	-->	操作员控制使能
SPURLM_AnalogValue	-->	操作员控制使能
SPDRLM_AnalogValue	-->	操作员控制使能
MO_PVHR_AnalogValue	-->	操作员控制使能
MO_PVLR_AnalogValue	-->	操作员控制使能
Permission_SP_Bumpless	-->	Target_BackColor
SPHighLimit_AnalogValue	-->	BackColor_Value
SPLowLimit_AnalogValue	-->	BackColor_Value
ManHighLimit_AnalogValue	-->	BackColor_Value
ManLowLimit_AnalogValue	-->	BackColor_Value
SPURLM_AnalogValue	-->	BackColor_Value
SPDRLM_AnalogValue	-->	BackColor_Value
MO_PVHR_AnalogValue	-->	BackColor_Value
MO_PVLR_AnalogValue	-->	BackColor_Value

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.4 CTRL_PID: 参数视图



如果在控制器图表中实例化一个 GAIN_SHD 类型的块，且此块具有扩展名为“_gsc”的控制器块名称，则会显示一个附加按钮 **GAIN_SHD**，可通过此按钮调用 GAIN_SHD 面板。

模拟显示和数字格式

“ControlError_AnalogValue”过程值通过“AdvancedAnalogDisplay”对象实现。数字格式在块图标中的“Format_InputValue”属性中定义。

其它所有模拟显示都是通过常规的“浮点格式”(Floating-point format) I/O 域实现的。

控制权限

“Permission_Gain”对象评估 WinCC 权限级别和“OPTI_EN = FALSE”参数。

与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
Permission_Gain	-->	Level_Source
OPTI_EN_CHECKBOX_L	-->	操作员控制使能
Permission_Gain	-->	Target_Operator-ControlEnable
Gain_AnalogValue	-->	操作员控制使能
TN_AnalogValue	-->	操作员控制使能
TV_AnalogValue	-->	操作员控制使能
DEADB_W_AnalogValue	-->	操作员控制使能
TM_LAG_AnalogValue	-->	操作员控制使能
ERH_ALM_AnalogValue	-->	操作员控制使能
ERL_ALM_AnalogValue	-->	操作员控制使能
ER_HYS_AnalogValue3	-->	操作员控制使能
M_SUP_ER_CHECKBOX_L	-->	操作员控制使能
Permission_Gain	-->	Target_BackColor
Gain_AnalogValue	-->	BackColor_Value
TN_AnalogValue	-->	BackColor_Value
TV_AnalogValue	-->	BackColor_Value
DEADB_W_AnalogValue	-->	BackColor_Value
TM_LAG_AnalogValue	-->	BackColor_Value
ERH_ALM_AnalogValue	-->	BackColor_Value
ERL_ALM_AnalogValue	-->	BackColor_Value
ER_HYS_AnalogValue3	-->	BackColor_Value
格式	-->	Format_InputValue
ControlError_AnalogValue	-->	格式

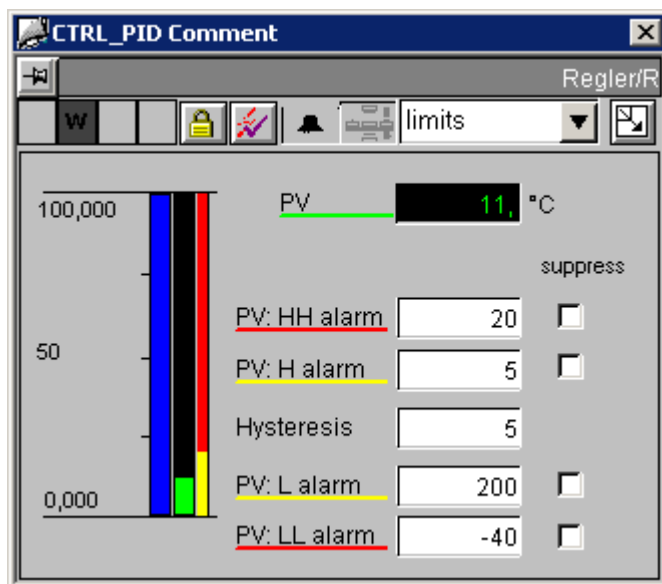
更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.5 CTRL_PID：限制视图

设定值操作限制



此视图中的设定值棒图显示相对于棒图限制的设定值输入限制。

设定值输入限制在维护视图中进行设置。

模拟显示和数字格式

“ProcessValue_AnalogValue”过程值通过“AdvancedAnalogDisplay”对象实现。数字格式通过块图标的“Format_InputValue”属性定义。

其它所有模拟显示都是通过常规的“浮点格式”(Floating-point format) I/O 域实现的。

控制权限

除 WinCC 授权级别外，“Permission_AlarmHigh_AnalogValue”权限对象还评估“OPTI_EN = FALSE”参数。

与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
Permission_AlarmHigh_AnalogValue	-->	Level_Source
Permission_AlarmHigh_AnalogValue	-->	Target_Operator-ControlEnable
AlarmHigh_AnalogValue	-->	操作员控制使能
WarningHigh_AnalogValue	-->	操作员控制使能
Hysteresis_AnalogValue	-->	操作员控制使能
WarningLow_AnalogValue	-->	操作员控制使能
AlarmLow_AnalogValue	-->	操作员控制使能
AlarmHigh_CHECKBOX_R	-->	操作员控制使能
WarningHigh_CHECKBOX_R	-->	操作员控制使能
WarningLow_CHECKBOX_R	-->	操作员控制使能
AlarmLow_CHECKBOX_R	-->	操作员控制使能
Permission_AlarmHigh_AnalogValue	-->	Target_BackColor
AlarmHigh_AnalogValue	-->	BackColor_Value
WarningHigh_AnalogValue	-->	BackColor_Value
Hysteresis_AnalogValue	-->	BackColor_Value
WarningLow_AnalogValue	-->	BackColor_Value
AlarmLow_AnalogValue	-->	BackColor_Value
格式	-->	Format_InputValue
ProcessValue_AnalogValue	-->	Format_InputValue

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.6 CTRL_S（所有视图）

总览

CTRL_S: 标准视图 (页 593)

CTRL_S: 维护视图 (页 596)

CTRL_S: 参数视图 (页 596)

CTRL_S: 限制视图 (页 596)

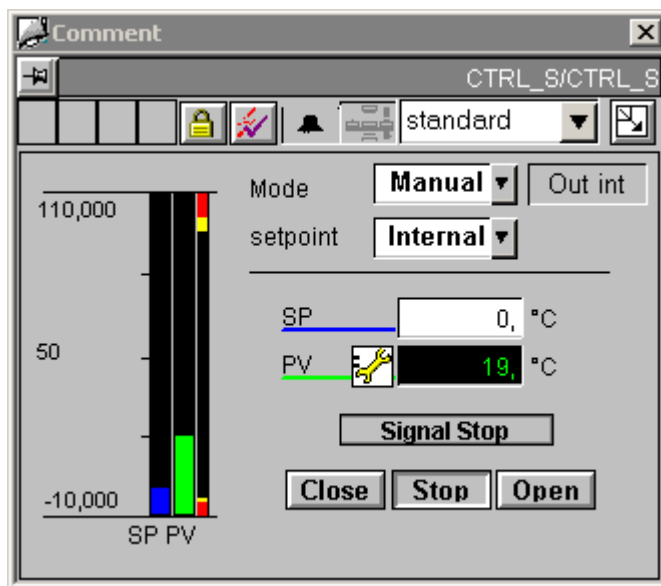
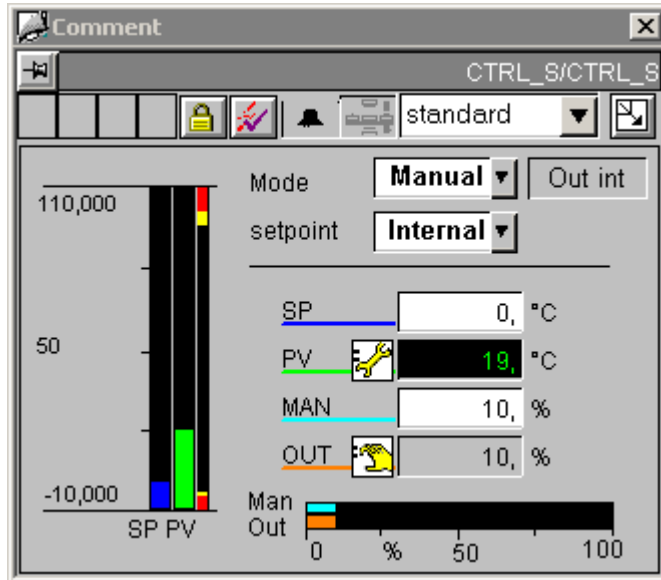
CTRL_S: StandardS 视图 (页 596)

全局视图: 消息视图 (页 656)

全局视图: 批生产视图 (页 658)

全局视图: 趋势视图 (页 659)

12.1.7 CTRL_S: 标准视图



键显示

与 **CTRL_PID** 标准视图相比，在 **CTRL_S** 标准视图中，棒图显示与手动及调节变量模拟值是根据 **LMNR_ON** 参数中指定的值来显示的。当“**LMNR_ON = FALSE**”时，将显示用于操作 **LMNDN_OP** 和 **LMNUP_OP** 参数的键。

LMNDN_OP_BinOp、**LMNUP_OP_BinOp** 和 **LMN_OP_Stop_BinOp** 键的可见性以及 **QLMNUP_QLMNDN** 的状态显示由脚本进行控制。当“**Output_BarStandard_3**”对象的“其它/Display”属性改变时，将会调用此脚本。

由于“可见”(Visible) 属性已用于内部目标函数，所以也可通过几何形状的 **X** 位置来控制这些对象的可见性。

LMNDN_OP 和 **LMNUP_OP** 只有一个“停止”(Stop) 键。由脚本来决定将值“0”写入这两个参数中的哪一个。如果“Display_Variable”属性发生变化，则会在“**LMNDN_OP_BinOp**”及“**LMNUP_OP_BinOp**”对象中激活这些脚本。

控制权限

此视图有三个权限对象：

- “Permission_Setpoint”
- “Permission_Manual”
- “Permission_LMNDN_OP”

除 WinCC 授权级别外，权限对象还评估下列参数：

权限对象	参数
“Permission_Setpoint”	“Q_SP_OP = TRUE”
“Permission_Manual”	“QLMNVOP = TRUE”
“Permission_LMNDN_OP”	“QLMNSOP = TRUE”

操作 PID 调节器与调节

可在参数视图对 **PID** 调节器进行操作（调节开启/关闭）。

当调节开启时，标准视图会在“手动/自动”(Manual/Auto) 模式组合框上方显示一个组合框。可使用此组合框再次关闭调节。

与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能		
Manual_COMBOBOX	-->	操作员控制使能		
External_COMBOBOX	-->	操作员控制使能		
Permission_Setpoint	-->	Level_Source	-->	Level_Target
Permission_Manual	-->	Level_Source	-->	Level_Target
Permission_LMNDN_OP	-->	Level_Source		
Permission_Setpoint	-->	Target_Operator- ControlEnable		
Setpoint_AnalogValue	-->	操作员控制使能		
Permission_Manual	-->	Target_Operator- ControlEnable		
Manual_AnalogValue	-->	操作员控制使能		
Permission_LMNDN_OP	-->	Target_Operator- ControlEnable		
LMN_OP_Stop_BinOp	-->	操作员控制使能		
LMNDN_OP_BinOp	-->	操作员控制使能		
LMNUP_OP_BinOp	-->	操作员控制使能		
格式	-->	Format_InputValue		
Setpoint_AnalogValue	-->	格式		
ProcessValue_AnalogValue	-->	格式		
格式	-->	Format_OutputValue		
Manual_AnalogValue	-->	格式		
Output_AnalogValue	-->	格式		

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.8 CTRL_S: 维护视图

请参见：CTRL_PID: 维护视图 (页 586)

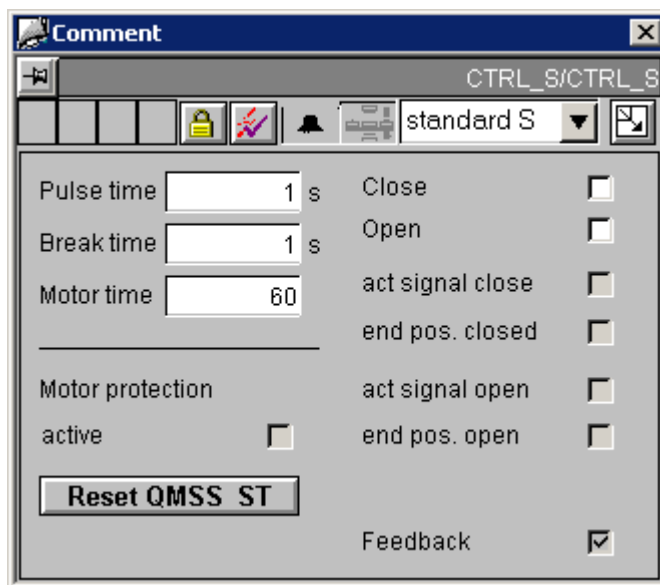
12.1.9 CTRL_S: 参数视图

请参见：CTRL_PID: 参数视图 (页 588)

12.1.10 CTRL_S: 限制视图

请参见：CTRL_PID: 限制视图 (页 590)

12.1.11 CTRL_S: StandardS 视图



控制权限

此视图有两个权限对象：

- “Permission_LMNDN_OP”
- “Permission_PulseTime_AnalogValue”

除 WinCC 授权级别外，权限对象还评估下列参数：

权限对象	参数
“Permission_LMNDN_OP”	“QLMNSOP = TRUE”
“Permission_PulseTime_AnalogValue”	“OPTI_EN = FALSE”

与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能		
Permission_LMNDN_OP	-->	Level_Source	-->	Level_Target
@Level6	-->	操作员控制使能		
Permission_PulseTime_AnalogValue	-->	Level_Source		
Reset_ButtonBit	-->	操作员控制使能		
Permission_LMNDN_OP	-->	Target_Operator-ControlEnable		
LMNDN_OP_Checkbox	-->	操作员控制使能		
LMNUP_OP_Checkbox	-->	操作员控制使能		
Permission_PulseTime_AnalogValue	-->	Target_Operator-ControlEnable		
PulseTime_Analogvalue	-->	操作员控制使能		
BreakTime_Analogvalue	-->	操作员控制使能		
MTR_TM_AnalogValue	-->	操作员控制使能		
Permission_PulseTime_AnalogValue	-->	Target_BackColor		
PulseTime_Analogvalue	-->	BackColor_Value		

BreakTime_Analogvalue	-->	BackColor_Value		
MTR_TM_AnalogValue	-->	BackColor_Value		

更多信息

更多相关信息, 可参考:

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.12 DIG_MON（所有视图）

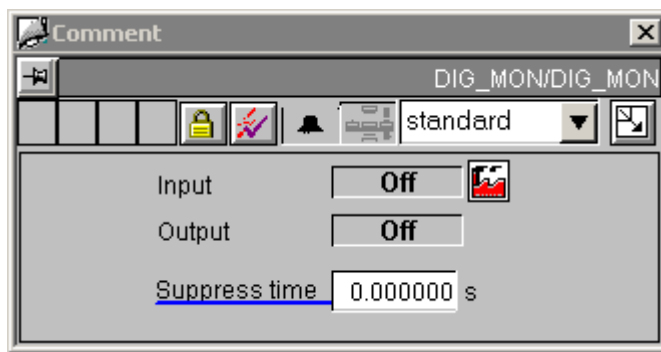
总览

DIG_MON：标准视图 (页 599)

全局视图：消息视图 (页 656)

全局视图：批生产视图 (页 658)

12.1.13 DIG_MON：标准视图



与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能
SuppressTime_PCS7_AnalogValue	-->	操作员控制使能
@Level6	-->	背景色
SuppressTime_PCS7_AnalogValue	-->	背景色

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

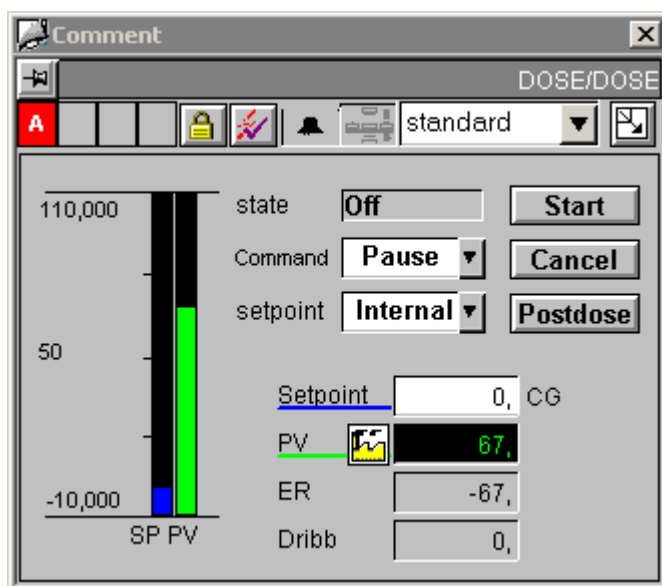
质量代码显示 (页 730)

12.1.14 DOSE（所有视图）

总览

- DOSE：标准视图 (页 600)
- DOSE：维护视图 (页 603)
- DOSE：参数视图 (页 604)
- DOSE：限制视图 (页 606)
- 全局视图：消息视图 (页 656)
- 全局视图：批生产视图 (页 658)
- 全局视图：趋势视图 (页 659)

12.1.15 DOSE：标准视图



设定值显示

“Setpoint_AnalogValue”和“BarStandard_2”设定值显示必须显示“内部设定值”的“SP_OP”变量和“外部设定值”的“SP_EXT”变量。由脚本使用“SetLink”命令来控制这两个对象的变量链接。

当“External_COMBOBOX”对象的“Display_Variable1”属性被更改时，将会调用此脚本。

控制权限

此视图有四个权限对象：

- “Permission_Setpoint”
- “Permission_Start”
- “Permission_Cancel”
- “Permission_Post_Dose”

除 WinCC 授权级别外，权限对象还评估下列参数：

权限对象	参数
“Permission_Setpoint”	“Q_SP_OP = TRUE”
“Permission_Start”	“QSTRT_OP = TRUE”
“Permission_Cancel”	“QCN_OP = TRUE”
“Permission_Post_Dose”	“QPD_OP = TRUE”

与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能		
Pause_COMBOBOX	-->	操作员控制使能		
External_COMBOBOX	-->	操作员控制使能		
Permission_Setpoint	-->	Level_Source	-->	Level_Target
Permission_Start	-->	Level_Source	-->	Level_Target
Permission_Cancel	-->	Level_Source	-->	Level_Target
Permission_Post_Dose	-->	Level_Source		
Permission_Setpoint	-->	Target_Operator-ControlEnable		
Setpoint_AnalogValue	-->	操作员控制使能		
Permission_Start	-->	Target_Operator-ControlEnable		
Start_ButtonBit	-->	操作员控制使能		

Permission_Cancel	-->	Target_Operator- ControlEnable		
Cancel_ButtonBit	-->	操作员控制使能		
Permission_Post_Dose	-->	Target_Operator- ControlEnable		
Post_Dose_ButtonBit	-->	操作员控制使能		
格式	-->	Format_InputValue		
Setpoint_AnalogValue	-->	格式		
ProcessValue_AnalogValue	-->	格式		

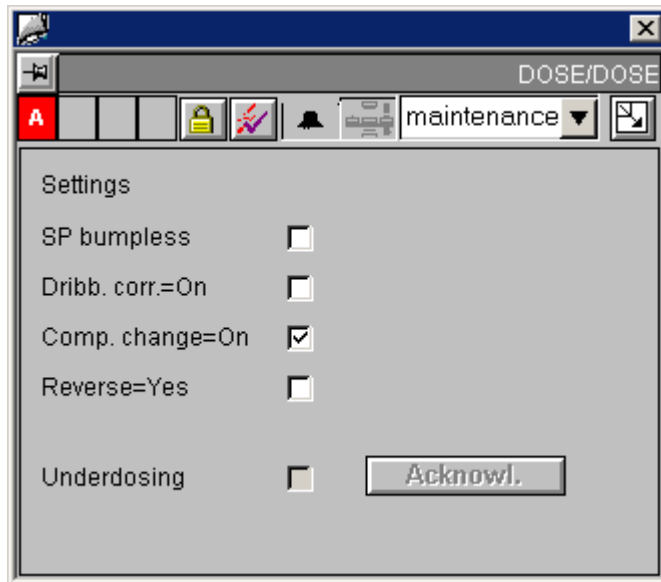
更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.16 DOSE：维护视图



控制权限

除 WinCC 授权级别外，“Permission_ACK_TOL_OP”权限对象还评估“QTOL_N = TRUE”和“QAK_OP= TRUE”参数。

与可控制对象直接连接的顺序和定位

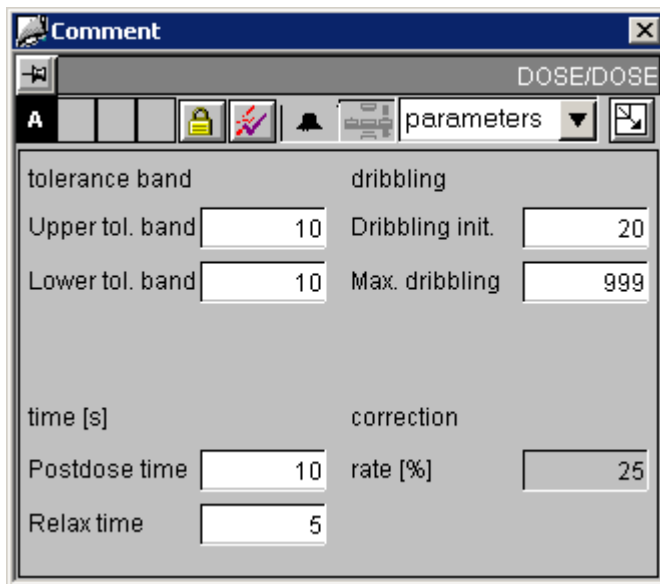
@Level6	-->	操作员控制使能
SPBUMPON_CHECKBOX_L	-->	操作员控制使能
DRIB_COR_CHECKBOX_L	-->	操作员控制使能
COMP_CHG_CHECKBOX_L2	-->	操作员控制使能
REVERSE_CHECKBOX_L	-->	操作员控制使能
Permission_ACK_TOL_OP	-->	Level_Source
Permission_ACK_TOL_OP	-->	Target_Operator-ControlEnable
ACK_TOL_OP_ButtonBit	-->	操作员控制使能

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.17 DOSE：参数视图



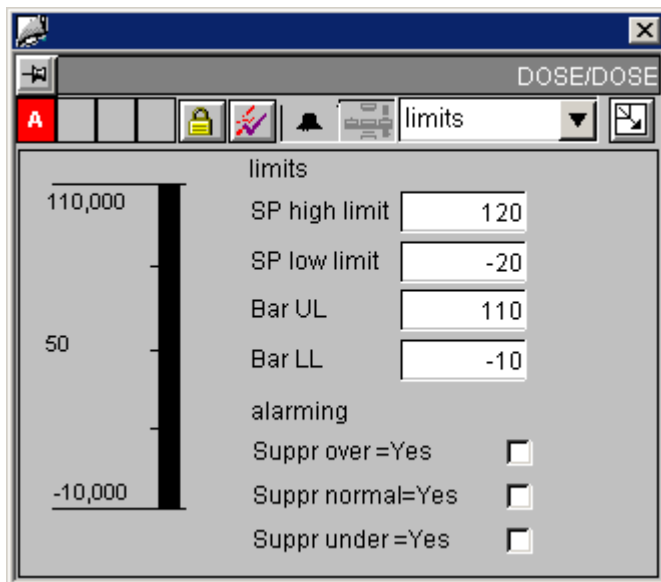
与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
TOL_P_PCS7_AnalogValue	-->	操作员控制使能
TOL_N_PCS7_AnalogValue	-->	操作员控制使能
PDOS_TME_PCS7_AnalogValue	-->	操作员控制使能
RELAXTME_PCS7_AnalogValue	-->	操作员控制使能
DRIBB_PCS7_AnalogValue	-->	操作员控制使能
DRIBBMAX_PCS7_AnalogValue	-->	操作员控制使能
@Level6	-->	背景色
TOL_P_PCS7_AnalogValue	-->	BackColor_Value
TOL_N_PCS7_AnalogValue	-->	BackColor_Value
PDOS_TME_PCS7_AnalogValue	-->	BackColor_Value
RELAXTME_PCS7_AnalogValue	-->	BackColor_Value
DRIBB_PCS7_AnalogValue	-->	BackColor_Value
DRIBBMAX_PCS7_AnalogValue	-->	BackColor_Value

更多信息

更多相关信息, 可参考:
总览中的对象 (页 654)

12.1.18 DOSE：限制视图



与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
SP_HLM_PCS7_AnalogValue	-->	操作员控制使能
SP_LLM_PCS7_AnalogValue	-->	操作员控制使能
MO_PVHR_PCS7_AnalogValue	-->	操作员控制使能
MO_PVLR_PCS7_AnalogValue	-->	操作员控制使能
M_SUP_2_UEBERDOS_CHECKBOX_L	-->	操作员控制使能
M_SUP_1_Dos_ok_CHECKBOX_L	-->	操作员控制使能
M_SUP_3_UNTERDOS_CHECKBOX_L	-->	操作员控制使能
@Level6	-->	背景色
SP_HLM_PCS7_AnalogValue	-->	BackColor_Value
SP_LLM_PCS7_AnalogValue	-->	BackColor_Value
MO_PVHR_PCS7_AnalogValue	-->	BackColor_Value
MO_PVLR_PCS7_AnalogValue	-->	BackColor_Value

更多信息

更多相关信息，可参考：
总览中的对象 (页 654)

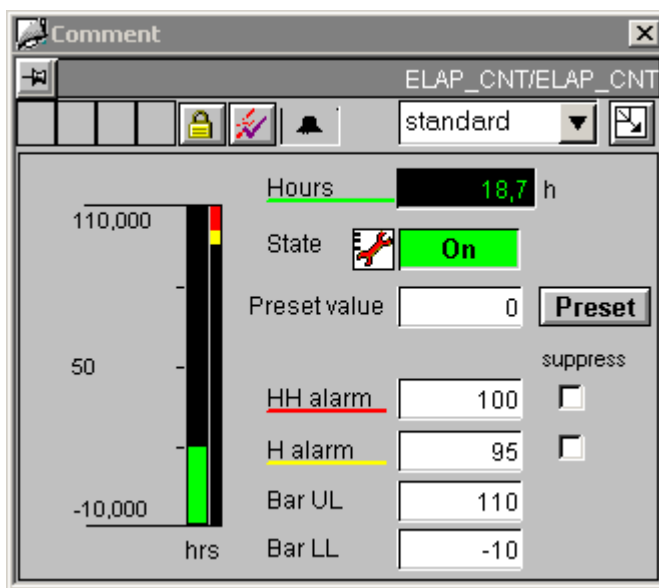
12.1.19 ELAP_CNT（所有视图）

总览

ELAP_CNT：标准视图 (页 608)

全局视图：消息视图 (页 656)

12.1.20 ELAP_CNT：标准视图



与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能
HOURS_OP_AnalogValue	-->	操作员控制使能
TRACK_OP_ButtonBit	-->	操作员控制使能
@Level5	-->	背景色
HOURS_OP_AnalogValue	-->	BackColor_Value
@Level6	-->	操作员控制使能
AlarmHigh_AnalogValue	-->	操作员控制使能
WarningHigh_AnalogValue	-->	操作员控制使能
MO_HOUHR_PCS7_AnalogValue	-->	操作员控制使能
MO_HOULR_PCS7_AnalogValue	-->	操作员控制使能
AlarmHigh_CHECKBOX_R	-->	操作员控制使能
WarningHigh_CHECKBOX_R	-->	操作员控制使能
@Level6	-->	背景色
AlarmHigh_AnalogValue	-->	BackColor_Value
WarningHigh_AnalogValue	-->	BackColor_Value
MO_HOUHR_PCS7_AnalogValue	-->	BackColor_Value
MO_HOULR_PCS7_AnalogValue	-->	BackColor_Value
格式	-->	Format_InputValue
HOURS_AnalogValue	-->	格式

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

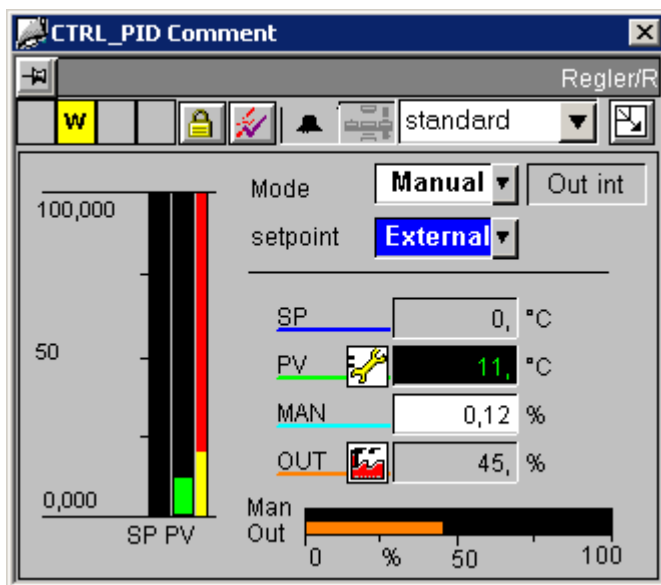
质量代码显示 (页 730)

12.1.21 FMCS_PID（所有视图）

总览

- FMCS_PID: 标准视图 (页 610)
- FMCS_PID: 维护视图 (页 613)
- FMCS_PID: 参数视图 (页 615)
- FMCS_PID: 限制视图 (页 617)
- FMCS_PID: StandardS 视图 (页 618)
- 全局视图: 消息视图 (页 656)
- 全局视图: 批生产视图 (页 658)
- 全局视图: 趋势视图 (页 659)

12.1.22 FMCS_PID: 标准视图



此 FMCS_PID 标准视图的外观与 CTRL_PID 标准视图 (页 584) 相同。

模拟显示和数字格式

所有模拟显示均通过“AdvancedAnalogDisplay”对象实现。值格式通过“Format_InputValue”及“Format_OutputValue”块图标属性进行设置。

控制权限

此视图有三个权限对象：

- “Permission_Manual_COMBOBOX”
- “Permission_Setpoint”
- “Permission_Manual”

除 WinCC 授权级别外，权限对象还评估下列参数：

权限对象	参数
“Permission_Manual_COMBOBOX”	<ul style="list-style-type: none"> • “QMODF = FALSE”
“Permission_Setpoint”	<ul style="list-style-type: none"> • “Q_SP_OP = TRUE” • “QMODF = FALSE”
“Permission_Manual”	<ul style="list-style-type: none"> • “QLMNOP = TRUE” • “QMODF = FALSE”

与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能		
Permission_Manual_COMBOBOX	-->	Level_Source	-->	Level_Target
Permission_Setpoint	-->	Level_Source	-->	Level_Target
Permission_Manual	-->	Level_Source		
Permission_Manual_COMBOBOX	-->	Target_Operator-ControlEnable		
Manual_COMBOBOX	-->	操作员控制使能		
External_COMBOBOX	-->	操作员控制使能		
Permission_Setpoint	-->	Target_Operator-ControlEnable		
Setpoint_AnalogValue	-->	操作员控制使能		
Permission_Manual	-->	Target_Operator-ControlEnable		
Manual_AnalogValue	-->	操作员控制使能		

格式	-->	Format_InputValue		
Setpoint_AnalogValue	-->	格式		
ProcessValue_AnalogValue	-->	格式		
格式	-->	Format_OutputValue		
Manual_AnalogValue	-->	格式		
Output_AnalogValue	-->	格式		

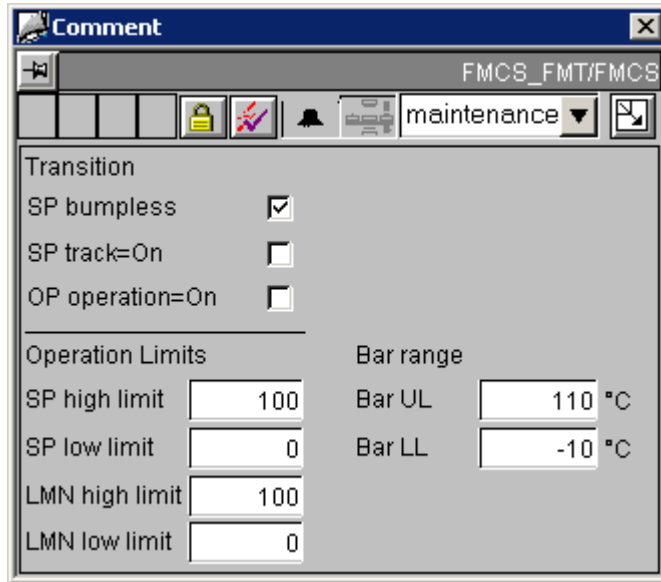
更多信息

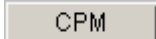
更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.23 FMCS_PID: 维护视图



如果在控制器图表中实例化一个 CPM 类型的块，且此块具有扩展名为“_cpm”的控制器块名称，则会显示一个附加按钮 ，可通过此按钮调用 CPM 面板。

控制权限

除 WinCC 授权级别外，“Permission_all”权限对象也会对“QMODF = FALSE”参数进行评估。

与可控制对象直接连接的顺序和定位

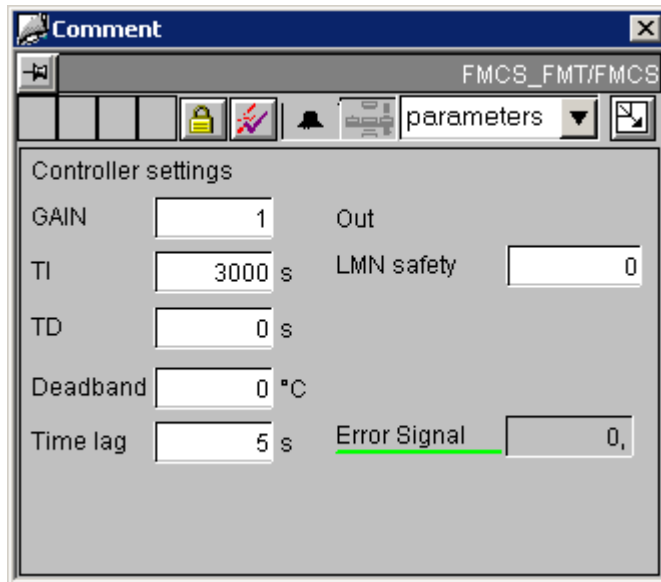
@Level6	-->	操作员控制使能
Permission_all	-->	Level_Source
Permission_all	-->	Target_Operator-ControlEnable
Bumpless_CHECKBOX_L	-->	操作员控制使能
SP_TRK_ON_CHECKBOX_L	-->	操作员控制使能
Oper_OP_CHECKBOX	-->	操作员控制使能
SPHighLimit_AnalogValue	-->	操作员控制使能
SPLowLimit_AnalogValue	-->	操作员控制使能
ManHighLimit_AnalogValue	-->	操作员控制使能
ManLowLimit_AnalogValue	-->	操作员控制使能
MO_PVHR_AnalogValue	-->	操作员控制使能
MO_PVLR_AnalogValue	-->	操作员控制使能
Permission_all	-->	Target_BackColor
SPHighLimit_AnalogValue	-->	BackgroundColor_Value
SPLowLimit_AnalogValue	-->	BackgroundColor_Value
ManHighLimit_AnalogValue	-->	BackgroundColor_Value
ManLowLimit_AnalogValue	-->	BackgroundColor_Value
MO_PVHR_AnalogValue	-->	BackgroundColor_Value
MO_PVLR_AnalogValue	-->	BackgroundColor_Value

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.24 FMCS_PID: 参数视图



如果在控制器图表中实例化一个 GAIN_SHD 类型的块，且此块具有扩展名为“_gsc”的控制器块名称，则会显示一个附加按钮 **GAIN_SHD**，可通过此按钮调用 GAIN_SHD 面板。

模拟显示和数字格式

“ControlError_AnalogValue”过程值通过“AdvancedAnalogDisplay”对象实现。数字格式在块图标中的“Format_InputValue”属性中定义。

其它所有模拟显示都是通过常规的“浮点格式”(Floating-point format) I/O 域实现的。

控制权限

除 WinCC 授权级别外，“Permission_all”权限对象也会对“QMODF = FALSE”参数进行评估。

与可控制对象直接连接的顺序和定位

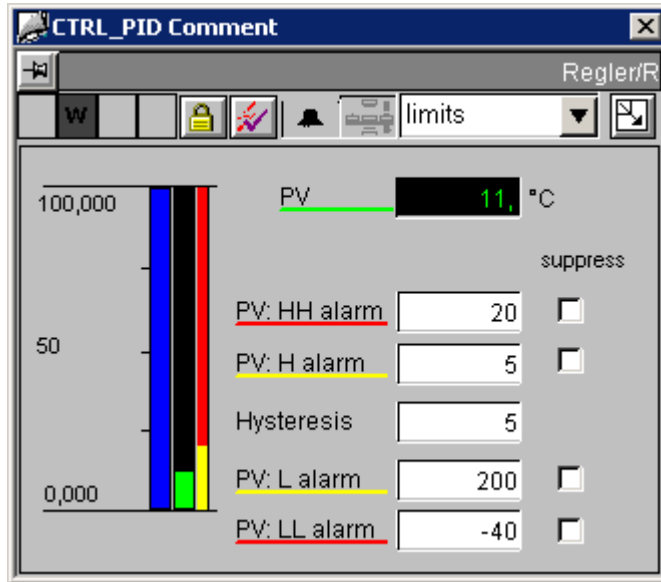
@Level6	-->	操作员控制使能		
Permission_all	-->	Level_Source	-->	Level_Target
OPTI_EN_CHECKBOX_L	-->	操作员控制使能		
Permission_all	-->	Target_Operator- ControlEnable		
Gain_AnalogValue	-->	操作员控制使能		
TN_AnalogValue	-->	操作员控制使能		
TV_AnalogValue	-->	操作员控制使能		
DEADB_W_AnalogValue	-->	操作员控制使能		
TM_LAG_AnalogValue	-->	操作员控制使能		
LMN_SAVE_PCS7_AnalogValue2	-->	操作员控制使能		
Permission_all	-->	Target_BackColor		
Gain_AnalogValue	-->	BackgroundColor_Value		
TN_AnalogValue	-->	BackgroundColor_Value		
TV_AnalogValue	-->	BackgroundColor_Value		
DEADB_W_AnalogValue	-->	BackgroundColor_Value		
TM_LAG_AnalogValue	-->	BackgroundColor_Value		
LMN_SAVE_PCS7_AnalogValue2	-->	BackgroundColor_Value		
格式	-->	Format_InputValue		
ErrorSignal_AnalogValue	-->	格式		

更多信息

更多相关信息，可参考：

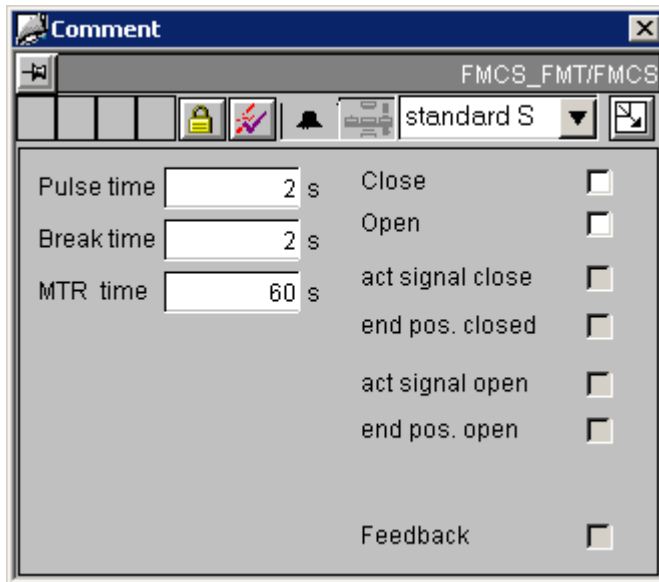
总览中的对象 (页 654)

12.1.25 FMCS_PID: 限制视图



此 FMCS_PID 限制视图与 CTRL_PID 限制视图 (页 590)几乎相同。唯一的差别是，对于控制权限对象，还会查询“QMODF = FALSE”。

12.1.26 FMCS_PID: StandardS 视图



控制权限

此视图有两个权限对象：

- “Permission_LMNDN_OP”
- “Permission_PulseTime_AnalogValue”

除 WinCC 授权级别外，权限对象还评估下列参数：

权限对象	参数
“Permission_LMNDN_OP”	<ul style="list-style-type: none"> • “QLMNSOP = TRUE” • “QMODF = FALSE”
“Permission_Pulse_TM”	<ul style="list-style-type: none"> • “QMODF = FALSE”

与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能		
Permission_LMNDN_OP	-->	Level_Source		
@Level6	-->	操作员控制使能		
Permission_Pulse_TM	-->	Level_Source	-->	Level_Target
Permission_LMNDN_OP	-->	Target_Operator- ControlEnable		
LMNDN_OP_Checkbox	-->	操作员控制使能		
LMNUP_OP_Checkbox	-->	操作员控制使能		
Permission_Pulse_TM	-->	Target_Operator- ControlEnable		
PulseTime_Analogvalue	-->	操作员控制使能		
BreakTime_Analogvalue	-->	操作员控制使能		
MTR_TM_AnalogValue	-->	操作员控制使能		
Permission_Pulse_TM	-->	Target_BackColor		
PulseTime_Analogvalue	-->	BackColor_Value		
BreakTime_Analogvalue	-->	BackColor_Value		
MTR_TM_AnalogValue	-->	BackColor_Value		

更多信息

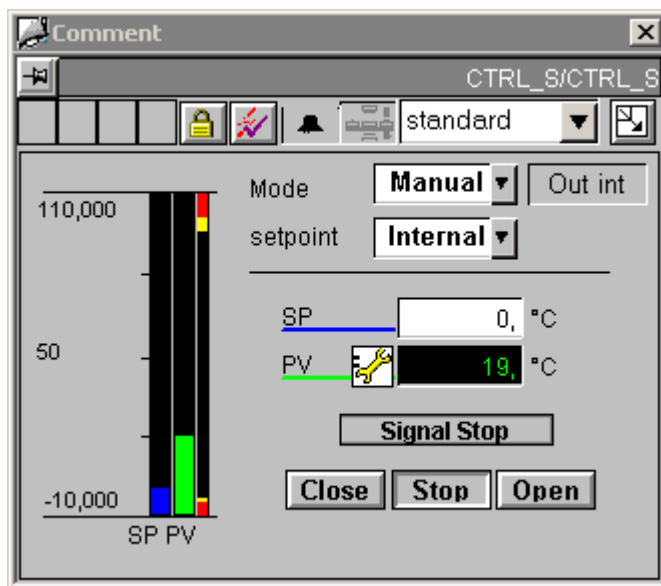
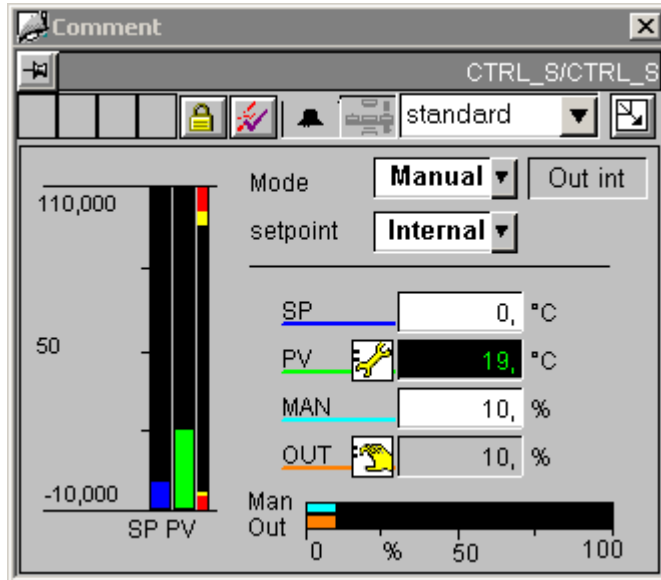
更多相关信息，可参考：
总览中的对象 (页 654)

12.1.27 FMT_PID（所有视图）

总览

- FMT_PID: 标准视图 (页 621)
- FMT_PID: 维护视图 (页 624)
- FMT_PID: 参数视图 (页 624)
- FMT_PID: 限制视图 (页 624)
- FMT_PID: StandardS 视图 (页 625)
- 全局视图: 消息视图 (页 656)
- 全局视图: 批生产视图 (页 658)
- 全局视图: 趋势视图 (页 659)

12.1.28 FMT_PID：标准视图



此 FMT_PID 标准视图的外观与 CTRL_S 标准视图 (页 593)几乎相同。

与 CTRL_S 标准视图相比，在 FMT_PID 标准视图中，棒图显示与手动及调节变量模拟值是根据 LMNR_ON 和 QSTEPCON 参数中指定的值来显示的。

棒图、模拟值和键的显示

当“LMNR_ON = FALSE”和“QSTEPCON = TRUE”时，将会显示用于操作 LMNDN_OP 及 LMNUP_OP 参数的键。

LMNDN_OP_BinOp、LMNUP_OP_BinOp 和 LMN_OP_Stop_BinOp 键的可见性以及 QLMNUP_QLMNDN 的状态显示由脚本进行控制。当“VISIBLE_Analog_Output”对象的“链接/QSTEPCON”及“链接/QLMNR_ON”属性被更改时，将会调用此脚本。

由于“可见”(Visible) 属性已用于内部目标函数，所以也可通过几何形状的 X 位置来控制这些对象的可见性。

LMNDN_OP 和 LMNUP_OP 只有一个“停止”(Stop) 键。由脚本来决定将值“0”写入这两个参数中的哪一个。如果“Display_Variable”属性发生变化，则会在“LMNDN_OP_BinOp”及“LMNUP_OP_BinOp”对象中激活这些脚本。

控制权限

此视图有四个权限对象：

- “Permission_Manual_COMBOBOX”
- “Permission_Setpoint”
- “Permission_Manual”
- “Permission_LMNDN_OP”

除 WinCC 授权级别外，权限对象还评估下列参数：

权限对象	参数
“Permission_Manual_COMBOBOX”	<ul style="list-style-type: none"> ● “QMODF = FALSE”
“Permission_Setpoint”	<ul style="list-style-type: none"> ● “Q_SP_OP = TRUE” ● “QMODF = FALSE”
“Permission_Manual”	<ul style="list-style-type: none"> ● “QLMNVOP = TRUE” ● “QMODF = FALSE”
“Permission_LMNDN_OP”	<ul style="list-style-type: none"> ● “QLMNSOP = TRUE” ● “QMODF = FALSE”

与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能		
Permission_Manual_COMBOBOX	-->	Level_Source	-->	Level_Target
Permission_Setpoint	-->	Level_Source	-->	Level_Target
Permission_Manual	-->	Level_Source	-->	Level_Target
Permission_LMNDN_OP	-->	Level_Source		
Permission_Manual_COMBOBOX	-->	Target_Operator- ControlEnable		
Manual_COMBOBOX	-->	操作员控制使能		
External_COMBOBOX	-->	操作员控制使能		
Permission_Setpoint	-->	Target_Operator- ControlEnable		
Setpoint_AnalogValue	-->	操作员控制使能		
Permission_Manual	-->	Target_Operator- ControlEnable		
Manual_AnalogValue	-->	操作员控制使能		
Permission_LMNDN_OP	-->	操作员控制使能		
LMN_OP_Stop_BinOp	-->	操作员控制使能		
LMNDN_OP_BinOp	-->	操作员控制使能		
LMNUP_OP_BinOp	-->	操作员控制使能		
格式	-->	Format_InputValue		
Setpoint_AnalogValue	-->	格式		
ProcessValue_AnalogValue	-->	格式		
格式	-->	Format_OutputValue		
Manual_AnalogValue	-->	格式		
Output_AnalogValue	-->	格式		

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.29 FMT_PID：维护视图

请参见：FMCS_PID：维护视图 (页 613)

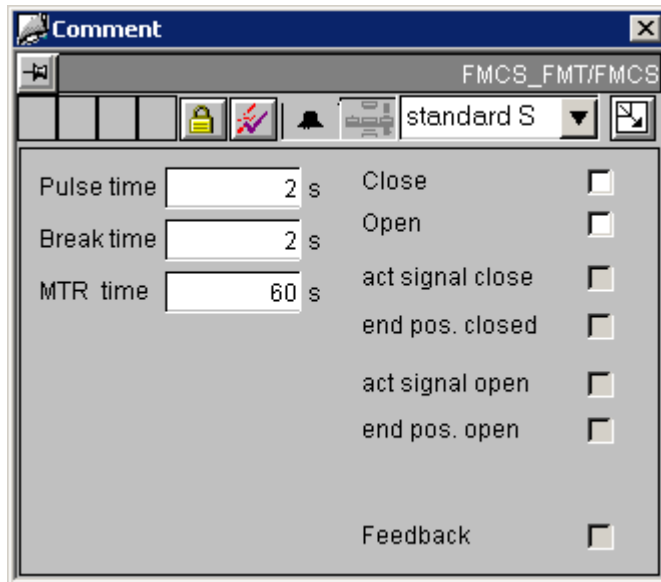
12.1.30 FMT_PID：参数视图

请参见：FMCS_PID：参数视图 (页 615)

12.1.31 FMT_PID：限制视图

请参见：FMCS_PID：限制视图 (页 617)

12.1.32 FMT_PID: StandardS 视图



此 FMT_PID standardS 视图的外观与 FMCS_PID standardS 视图 (页 618)几乎相同。

与 FMCS_PID 的差别

FMT_PID standardS 视图与 FMCS_PID standardS 视图在控制权限和显示方面存在差别：

- 除 WinCC 授权级别外，“Permission_LMNDN_OP”权限对象还评估“QLMNSOP = TRUE”及“QMODF = FALSE”参数。
- 仅当参数“QSTEPCON = TRUE”时，才会显示 standardS 视图。否则，此视图中的所有元素均不可见。

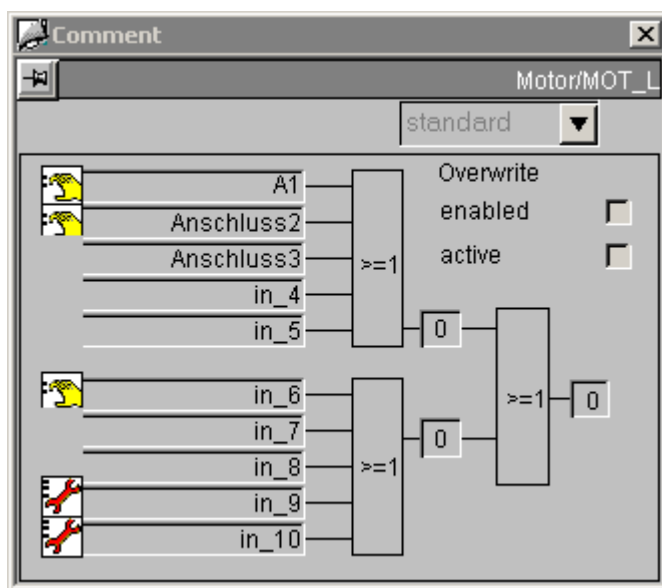
更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.33 INTERLOK: 标准视图



与可控制对象直接连接的顺序和定位

@Level6	-->	权限
overwrite_active_CHECKBOX_L2	-->	权限

输入“l1_1”至“l2_5”始终显示“log 1”文本 (string_1)。

输入的基本状态如下：

- “逻辑 0”状态在灰色背景上以黑色字符显示。
- “逻辑 1”状态在红色背景上以白色字符显示。

如果通过 NEG1_1 对输入取反，则显示颜色也会反相。

NEGRES_1 = 1 时同样如此。此时，第一个逻辑元素的全部五个输入的颜色都会反相。这样，以红色背景标记的输入将指示与错误状态有关的总和输出“Q”。

为避免在此种情况下产生误解，建议不要使用 NEGRES_1 及 NEGRES_2 输入。

此外，最好使用逻辑 OR 操作，因为只有这种方法才能确保颜色正确，并确保具有红色文本的输入将会导致互锁。

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.34 MEAS_MON (所有视图)

总览

MEAS_MON: 标准视图 (页 627)

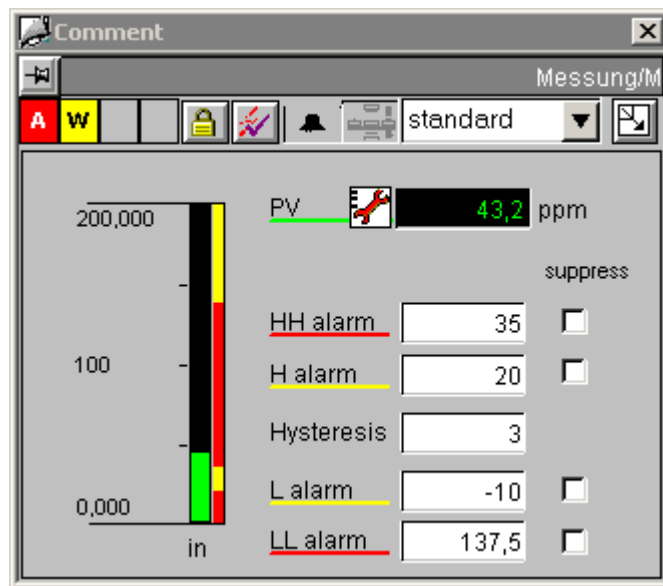
MEAS_MON: 限制视图 (页 629)

全局视图: 消息视图 (页 656)

全局视图: 批生产视图 (页 658)

全局视图: 趋势视图 (页 659)

12.1.35 MEAS_MON: 标准视图



模拟显示和数字格式

“PV_AnalogValue”过程值通过“AdvancedAnalogDisplay”设置。数字格式通过块图标的“Format_InputValue”属性定义。

其它模拟显示是通过常规的“浮点格式”(Floating-point format) I/O 域实现的。

与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
AlarmHigh_AnalogValue	-->	操作员控制使能
WarningHigh_AnalogValue	-->	操作员控制使能
Hysteresis_AnalogValue	-->	操作员控制使能
WarningLow_AnalogValue	-->	操作员控制使能
WarningHigh_AnalogValue	-->	操作员控制使能
AlarmHigh_CHECKBOX_R	-->	操作员控制使能
WarningHigh_CHECKBOX_R	-->	操作员控制使能
WarningLow_CHECKBOX_R	-->	操作员控制使能
AlarmLow_CHECKBOX_R	-->	操作员控制使能
@Level6	-->	背景色
AlarmHigh_AnalogValue	-->	背景色
WarningHigh_AnalogValue	-->	背景色
Hysteresis_AnalogValue	-->	背景色
WarningLow_AnalogValue	-->	背景色
WarningHigh_AnalogValue	-->	背景色
格式	-->	Format_InputValue
PV_AnalogValue	-->	格式

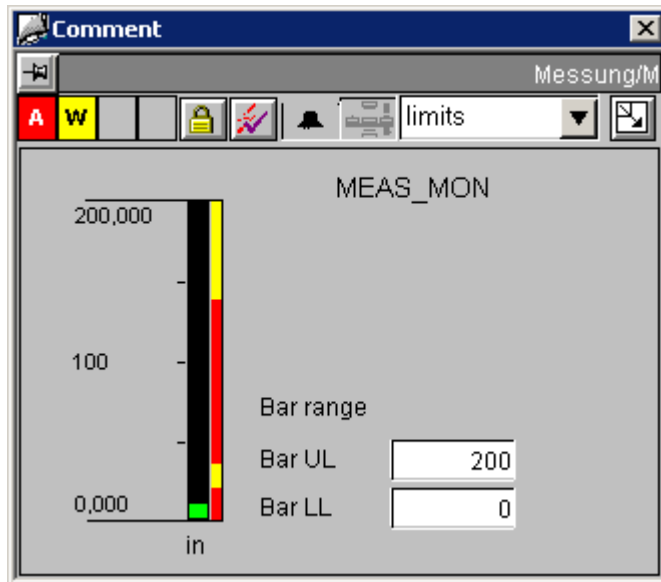
更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.36 MEAS_MON：限制视图



模拟显示

两种模拟显示都是通过常规的“浮点格式”(Floating-point format) I/O 域实现的。

与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
MO_PVHR_AnalogValue	-->	操作员控制使能
MO_PVLR_AnalogValue	-->	操作员控制使能
@Level6	-->	背景色
MO_PVHR_AnalogValue	-->	背景色
MO_PVLR_AnalogValue	-->	背景色

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.37 MOT_REV (所有视图)

总览

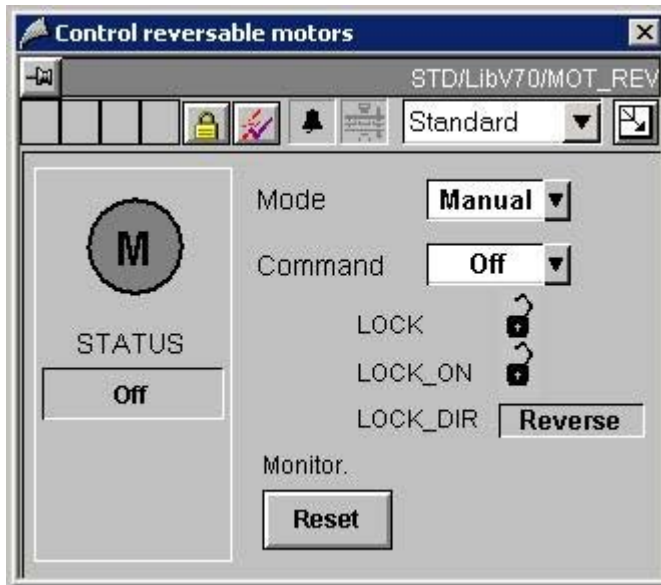
MOT_REV: 标准视图 (页 630)

MOT_REV: 维护视图 (页 632)

全局视图: 消息视图 (页 656)

全局视图: 批生产视图 (页 658)

12.1.38 MOT_REV: 标准视图



与可控制对象直接连接的顺序和定位

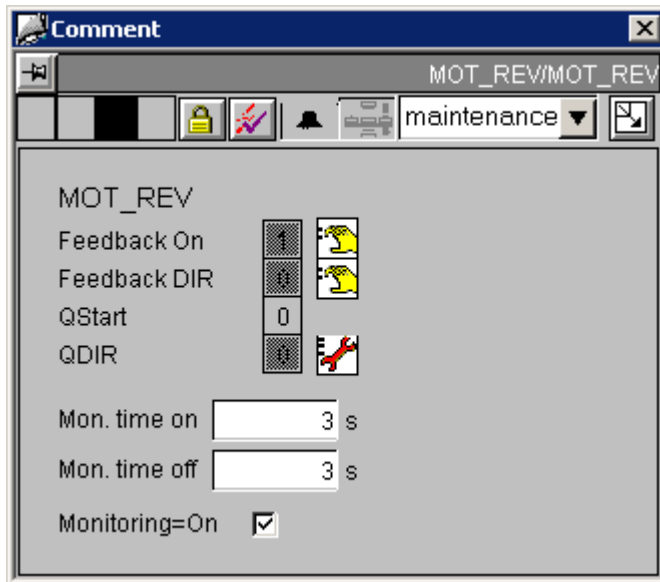
@Level5	-->	操作员控制使能
Auto_Manual_COMBOBOX	-->	操作员控制使能
Open_Close_Stop_3COMBOBOX	-->	操作员控制使能
Reset_ButtonBit	-->	操作员控制使能

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.39 MOT_REV: 维护视图



与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
TIME_ON_AnalogValue	-->	操作员控制使能
TIME_OFF_AnalogValue	-->	操作员控制使能
Monitoring_ON_CHECKBOX_L1	-->	操作员控制使能
@Level6	-->	背景色
TIME_MON_PCS7_AnalogValue	-->	背景色

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.40 MOT_SPED（所有视图）

总览

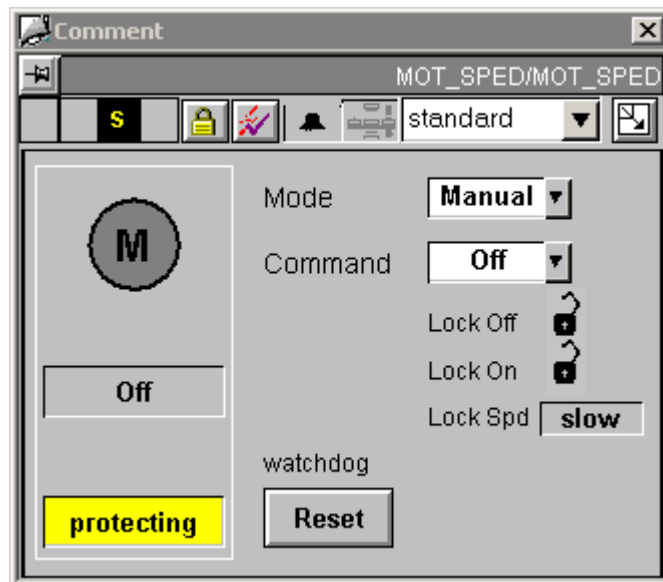
MOT_SPED：标准视图 (页 633)

MOT_SPED：维护视图 (页 635)

全局视图：消息视图 (页 656)

全局视图：批生产视图 (页 658)

12.1.41 MOT_SPED：标准视图



与可控制对象直接连接的顺序和定位

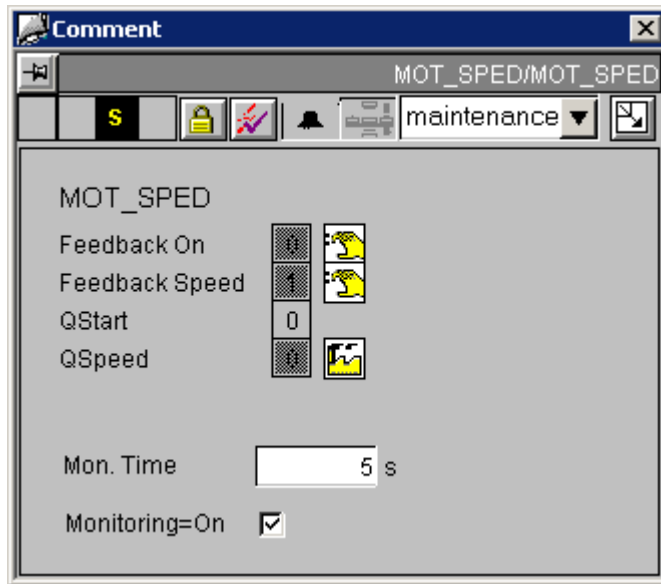
@Level5	-->	操作员控制使能
Auto_Manual_COMBOBOX	-->	操作员控制使能
Open_Close_Stop_3COMBOBOX	-->	操作员控制使能
Reset_ButtonBit	-->	操作员控制使能

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.42 MOT_SPED：维护视图



与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
TIME_MON_PCS7_AnalogValue	-->	操作员控制使能
Monitoring_ON_CHECKBOX_L1	-->	操作员控制使能
@Level6	-->	背景色
TIME_OFF_PCS7_AnalogValue	-->	背景色

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.43 MOTOR（所有视图）

总览

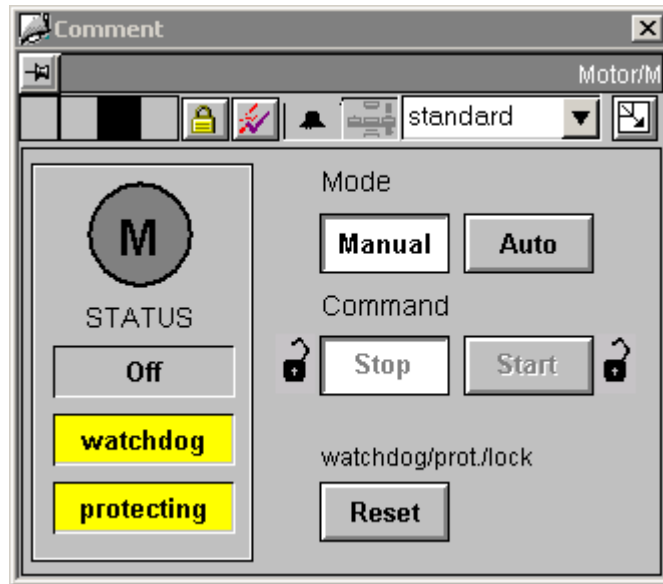
MOTOR: 标准视图 (页 637)

MOTOR: 维护视图 (页 639)

全局视图: 消息视图 (页 656)

全局视图: 批生产视图 (页 658)

12.1.44 MOTOR：标准视图



与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能		
Permission_Manual_BinOp0	-->	Level_Source	-->	Level_Target
Permission_Automatic_BinOp1	-->	Level_Source	-->	Level_Target
Permission_Off_BinOp2	-->	Level_Source	-->	Level_Target
Permission_On_BinOp1	-->	Level_Source	-->	Level_Target
Reset_ButtonBit	-->	操作员控制使能		
Permission_Manual_BinOp0	-->	Target_Operator-ControlEnable		
Manual_BinOp0	-->	操作员控制使能		
Permission_Automatic_BinOp1	-->	Target_Operator-ControlEnable		
Automatic_BinOp1	-->	操作员控制使能		
Permission_Off_BinOp2	-->	Format_InputValue		
Off_BinOp2	-->	格式		
Permission_On_BinOp1	-->	Format_OutputValue		

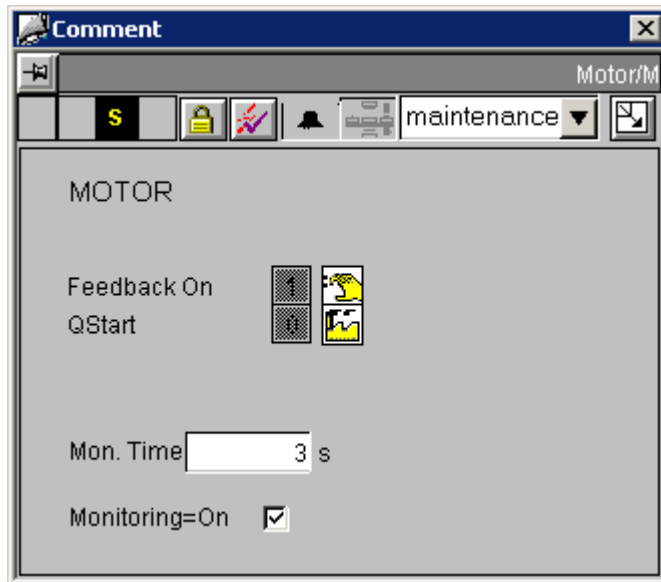
On_BinOp1	-->	格式		
除 WinCC 授权级别外，权限对象还评估以下 AS 块参数：				
Permission_Manual_BinOp0	-->	“QMANOP = TRUE”		
Permission_Automatic_BinOp1	-->	“QAUTOP = TRUE”		
Permission_Off_BinOp2	-->	“QOFF_OP = TRUE”		
Permission_On_BinOp1	-->	“QON_OP = TRUE”		

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.45 MOTOR：维护视图



与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
Monitoring_ON_CHECKBOX_L1	-->	操作员控制使能
Monitoring_PCS7_AnalogValue	-->	操作员控制使能

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

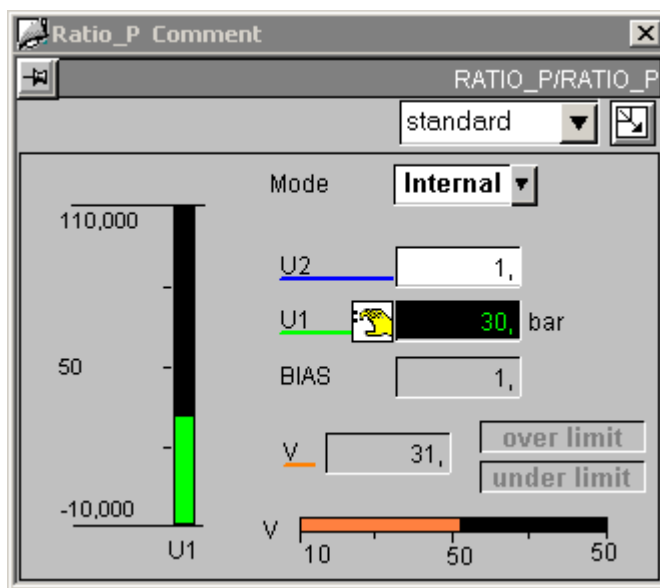
12.1.46 RATIO_P（所有视图）

总览

RATIO_P：标准视图 (页 640)

RATIO_P：限制视图 (页 642)

12.1.47 RATIO_P：标准视图



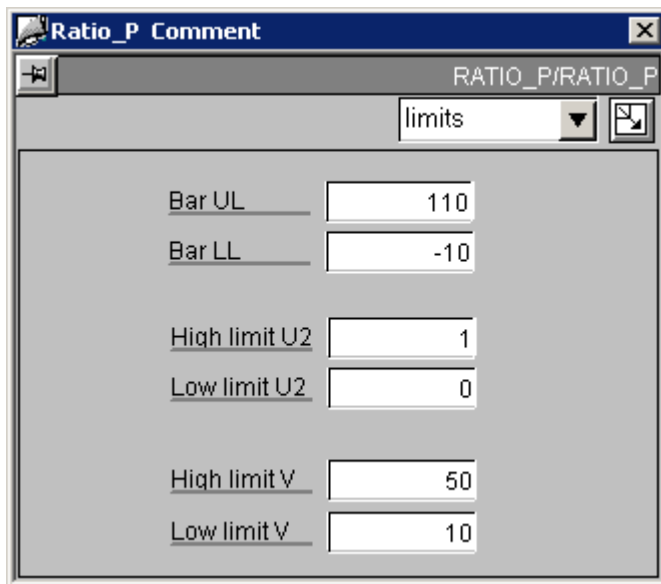
与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能		
Permission_ExternalComboBox	-->	Level_Source	-->	Level_Target
Permission_U2	-->	Level_Source		
Permission_ExternalComboBox	-->	Target_Operator- ControlEnable		
External_COMBOBOX	-->	操作员控制使能		
Permission_U2	-->	Target_Operator- ControlEnable		
U2_AnalogValue	-->	操作员控制使能		
格式	-->	Format_InputValue		
U2_AnalogValue	-->	格式		
格式	-->	Format_OutputValue		
V_AnalogValue	-->	格式		
格式	-->	Format_xx		
BIAS_AnalogValue	-->	格式		

更多信息

更多相关信息，可参考：
质量代码显示 (页 730)

12.1.48 RATIO_P：限制视图



与控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
MO_U1HR_AnalogValue	-->	操作员控制使能
MO_U1LR_AnalogValue	-->	操作员控制使能
U2_HL_AnalogValue1	-->	操作员控制使能
U2_LL_AnalogValue2	-->	操作员控制使能
V_HL_AnalogValue3	-->	操作员控制使能
V_LL_AnalogValue4	-->	操作员控制使能
@Level6	-->	BackColor
MO_U1HR_AnalogValue	-->	BackColor_Value
MO_U1LR_AnalogValue	-->	BackColor_Value
U2_HL_AnalogValue1	-->	BackColor_Value
U2_LL_AnalogValue2	-->	BackColor_Value
V_HL_AnalogValue3	-->	BackColor_Value
V_LL_AnalogValue4	-->	BackColor_Value

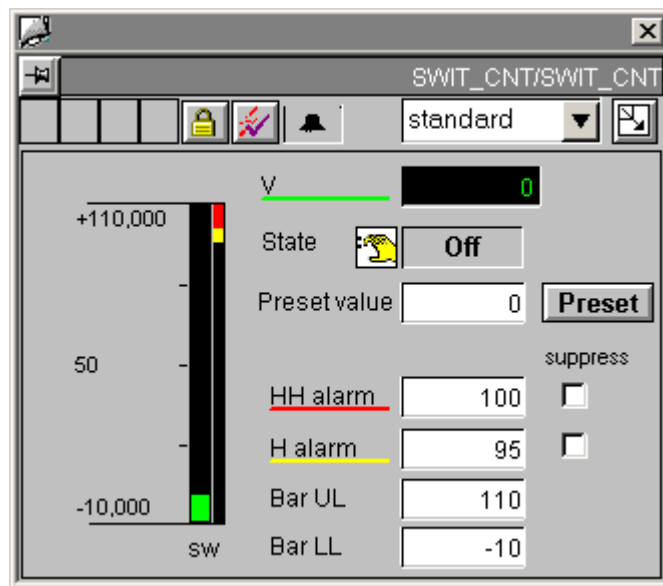
12.1.49 SWIT_CNT（所有视图）

总览

SWIT_CNT：标准视图 (页 643)

全局视图：消息视图 (页 656)

12.1.50 SWIT_CNT：标准视图



与可控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能
VTRACK_OP_AnalogValue	-->	操作员控制使能
TRACK_OP_ButtonBit	-->	操作员控制使能
@Level5	-->	背景色
VTRACK_OP_AnalogValue	-->	BackColor_Value
@Level6	-->	操作员控制使能
AlarmHigh_AnalogValue	-->	操作员控制使能
WarningHigh_AnalogValue	-->	操作员控制使能
MO_HOUHR_PCS7_AnalogValue	-->	操作员控制使能
MO_HOULR_PCS7_AnalogValue	-->	操作员控制使能
AlarmHigh_CHECKBOX_R	-->	操作员控制使能
WarningHigh_CHECKBOX_R	-->	操作员控制使能
@Level6	-->	背景色
AlarmHigh_AnalogValue	-->	BackColor_Value
WarningHigh_AnalogValue	-->	BackColor_Value
MO_HOUHR_PCS7_AnalogValue	-->	BackColor_Value
MO_HOULR_PCS7_AnalogValue	-->	BackColor_Value

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.51 VAL_MOT（所有视图）

总览

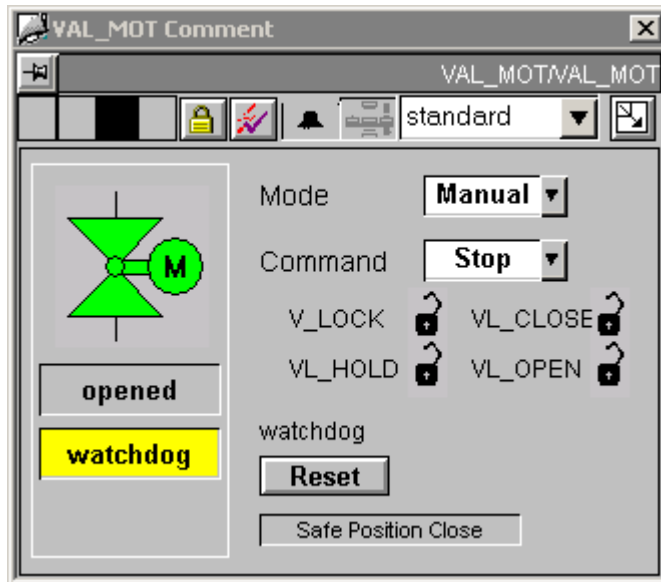
VAL_MOT：标准视图 (页 645)

VAL_MOT：维护视图 (页 647)

全局视图：消息视图 (页 656)

全局视图：批生产视图 (页 658)

12.1.52 VAL_MOT：标准视图



与可控制对象直接连接的顺序和定位

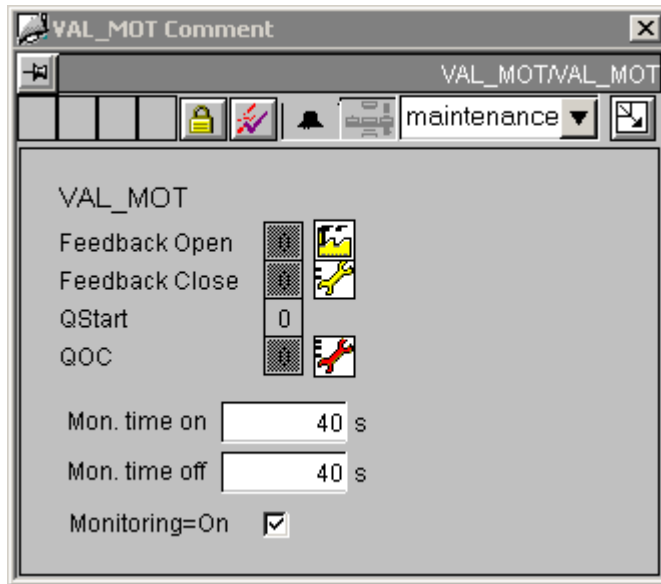
@Level5	-->	操作员控制使能
Auto_Manual_COMBOBOX	-->	操作员控制使能
Open_Close_Stop_3COMBOBOX	-->	操作员控制使能
Reset_ButtonBit	-->	操作员控制使能

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.53 VAL_MOT：维护视图



与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
Time_ON_AnalogValue	-->	操作员控制使能
TIME_OFF_PCS7_AnalogValue	-->	操作员控制使能
@Level6	-->	背景色
Time_ON_AnalogValue	-->	背景色
TIME_OFF_PCS7_AnalogValue	-->	背景色
Monitoring_ON_CHECKBOX_L1	-->	背景色

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.1.54 VALVE（所有视图）

总览

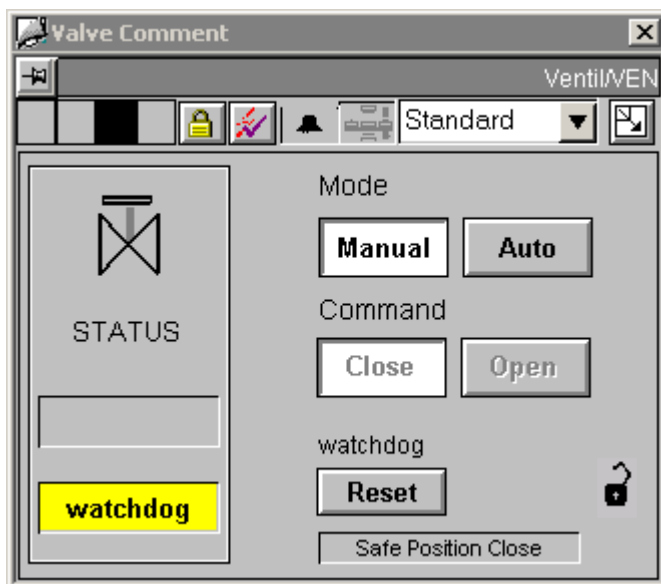
VALVE：标准视图 (页 648)

VALVE：维护视图 (页 650)

全局视图：消息视图 (页 656)

全局视图：批生产视图 (页 658)

12.1.55 VALVE：标准视图



邻近“关闭”(Close) 和“打开”(Open) 的两个互锁图标属于“锁定以关闭”(Lock to Close) 及“锁定以打开”(Lock to Open) 互锁。默认情况下，它们被设置为不可见，因为主要使用的是右下方的“锁定至安全位置”(Lock to Safe Position) 互锁图标。

对于 VALVE 类型的所有实例，必要时，可始终将这些图标切换为可见状态。

与可控制对象直接连接的顺序和定位

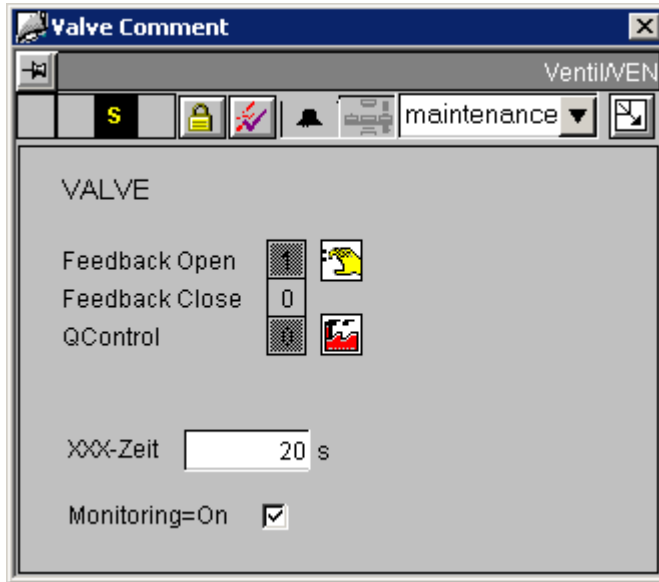
@Level5	-->	操作员控制使能		
Permission_Manual_BinOp0	-->	Level_Source	-->	Level_Target
Permission_Automatic_BinOp1	-->	Level_Source	-->	Level_Target
Permission_Off_BinOp2	-->	Level_Source	-->	Level_Target
Permission_On_BinOp1	-->	Level_Source	-->	Level_Target
Reset_ButtonBit	-->	操作员控制使能		
Permission_Manual_BinOp0	-->	Target_Operator-ControlEnable		
Manual_BinOp0	-->	操作员控制使能		
Permission_Automatic_BinOp1	-->	Target_Operator-ControlEnable		
Automatic_BinOp1	-->	操作员控制使能		
Permission_Close_BinOp2	-->	Target_Operator-ControlEnable		
Close_BinOp2	-->	操作员控制使能		
Permission_Open_BinOp1	-->	Target_Operator-ControlEnable		
Open_BinOp1	-->	操作员控制使能		
除 WinCC 授权级别外，权限对象还评估以下 AS 块参数：				
Permission_Manual_BinOp0	-->	"QMANOP = TRUE"		
Permission_Automatic_BinOp1	-->	"QAUTOP = TRUE"		
Permission_Off_BinOp2	-->	"QOFF_OP = TRUE"		
Permission_On_BinOp1	-->	"QON_OP = TRUE"		

更多信息

更多相关信息，可参考：

总览中的对象 (页 654)

12.1.56 VALVE： 维护视图



与可控制对象直接连接的顺序和定位

@Level6	-->	操作员控制使能
Monitoring_ON_CHECKBOX_L1	-->	操作员控制使能
Monitoring_PCS7_AnalogValue	-->	操作员控制使能

更多信息

更多信息，请参考以下部分：

总览中的对象 (页 654)

质量代码显示 (页 730)

12.2 面板：资产管理

12.2.1 ASSETMON 面板 [资产] 的视图

视图

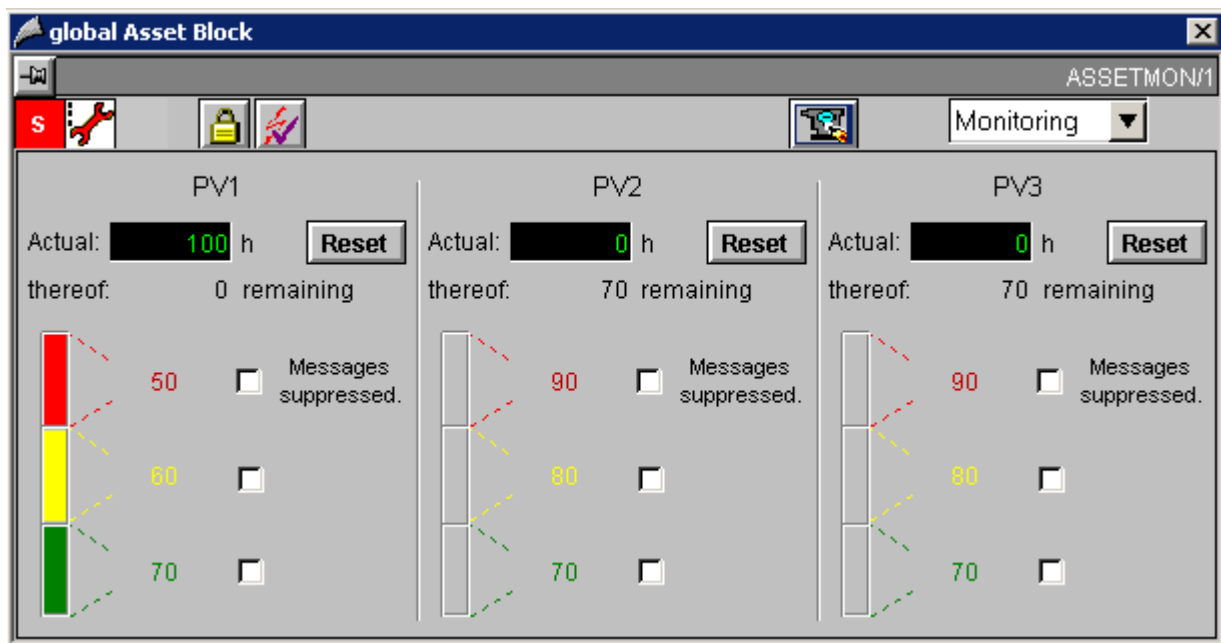
该面板有 5 个视图：

- 监视视图
- 诊断视图
- 标识视图
- 维护视图
- 消息视图

监视视图

在此视图中，显示了多达三个过程值。从 EDD 读取过程值的文本。

如果没有过程值文本，则监视视图不会被激活（灰显）。

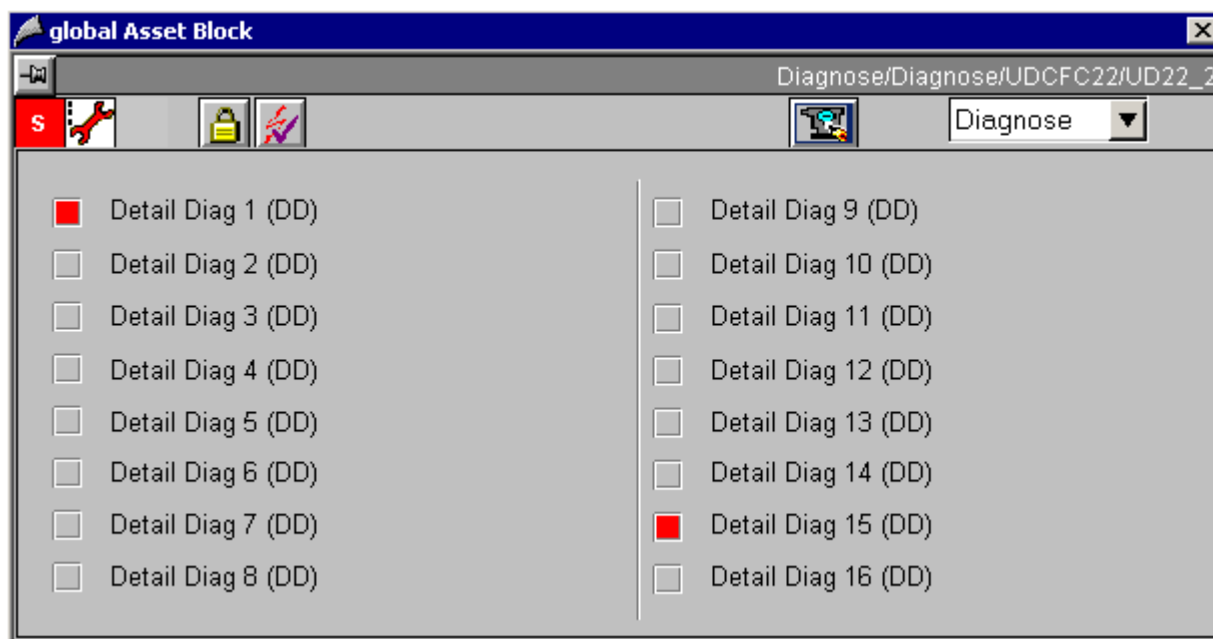


实际值显示在画面的上部。在其下面的框中，显示了实际值和下一个限制值间的差异。

限制值显示以灰色棒图形式显示“良好”(GOOD) 状态。如果达到了某个限制，则彩色框会出现在相应位置。棒图不表示模拟值，而是表示状态。使用限制显示旁的选项框可以抑制此消息（通过选中此框）。

诊断视图

在诊断视图中，显示了 16 个详细诊断的文本。这些文本对应于在 PDM 的 EDD 中分配的条目。



在诊断视图中，详细诊断的文本由状态显示标识，因此，如果设置了块输入 $DIAGx = 1$ ，则该文本是关联文本。

标识视图

请参见 标识视图 [资产] (页 665)

维护视图

请参见 维护视图 [资产] (页 663)

消息视图

请参见 消息视图 [资产] (页 662)

更多信息

更多相关信息，可参考：

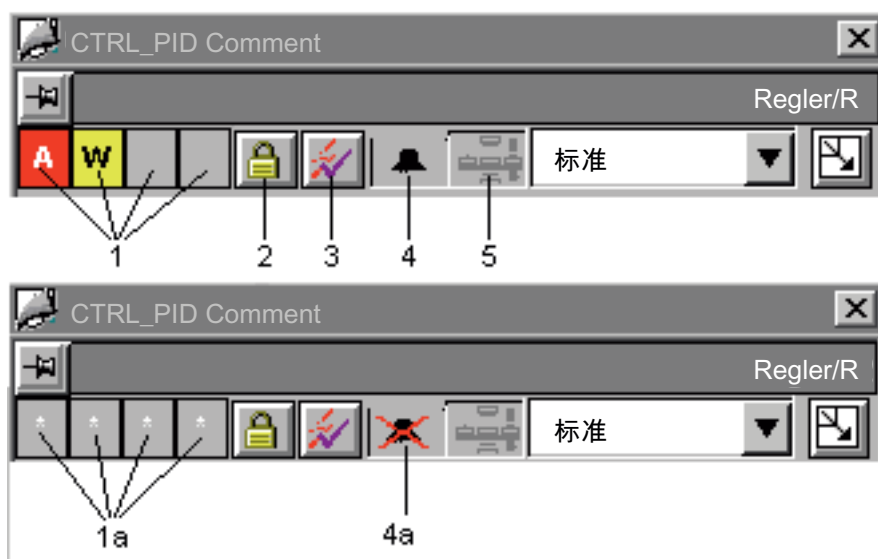
资产面板的全局表示和视图 (页 671)

12.3 全局视图和表示

12.3.1 对象总览

工具栏图标

每个面板都有一个工具栏，其中包含以下图标：



固定面板

总览行左上角有一个图标，可用来“固定”面板，使其不受区域变化的影响。该键显示如下：

- 未固定  （面板被调用后）
- 已固定  （按钮被按下之后）

面板被再次关闭之前，将一直保持固定状态。再次按该键也不起作用。

组显示

组显示 [1] 显示从块实例的 ALARM_8P 传送到 WinCC 的信息。组显示链接到变量的“EventState”。

锁定/解锁消息

“锁定/解锁消息”功能在总览中是通过键 [2] 实现的。

组显示会指明块实例的所有消息处于锁定还是解锁状态。如果在所有 4 个组显示域中都出现了白色的 * 字符 [1a]，则表明所有消息均被锁定。

只有分配了在块符号属性“Processcontrolling_backup”中定义的权限级别的操作员才能看到该按钮。

确认消息

使用键 [3]，可以对块实例的所有消息进行确认。

只有分配了在块符号属性“Processcontrolling_backup”中定义的权限级别的操作员才能看到该按钮。

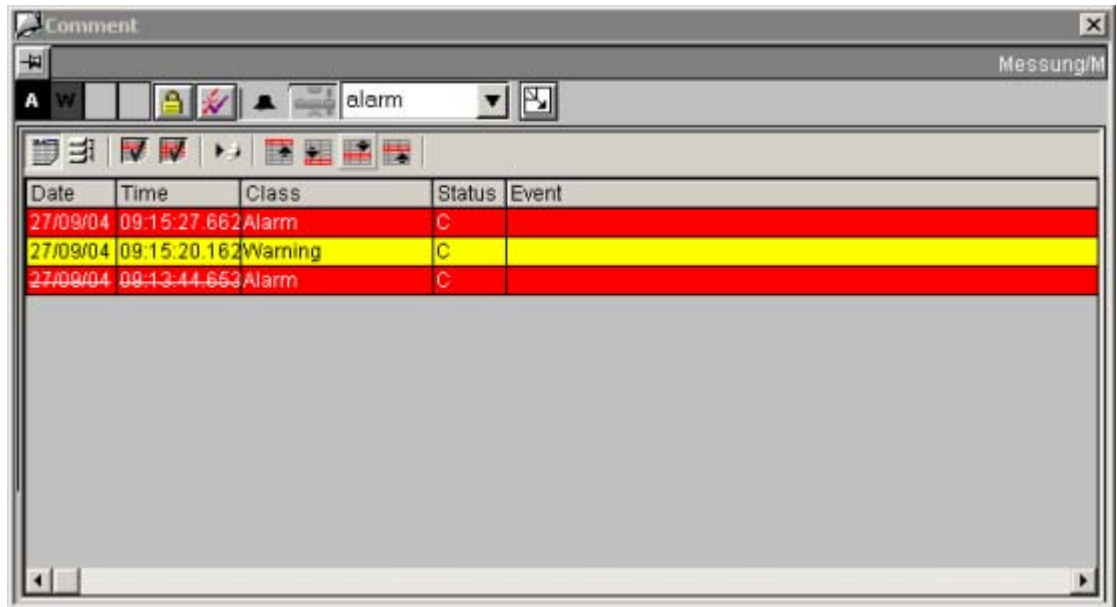
抑制消息

消息抑制指示是否使用“MSG_LOCK”参数禁用 [4] 或激活 [4a] AS 块中的“抑制过程消息”功能。如果激活了消息抑制，则会抑制此块实例中的所有消息（过程控制消息除外）。

占用显示

占用显示 [5] 指示 SIMATIC BATCH 块实例是否已被占用（“OCCUPIED”参数）。此时会在批生产视图中显示更多信息。

12.3.2 全局视图： 消息视图



含有消息视图的面板

下列面板包含消息视图：

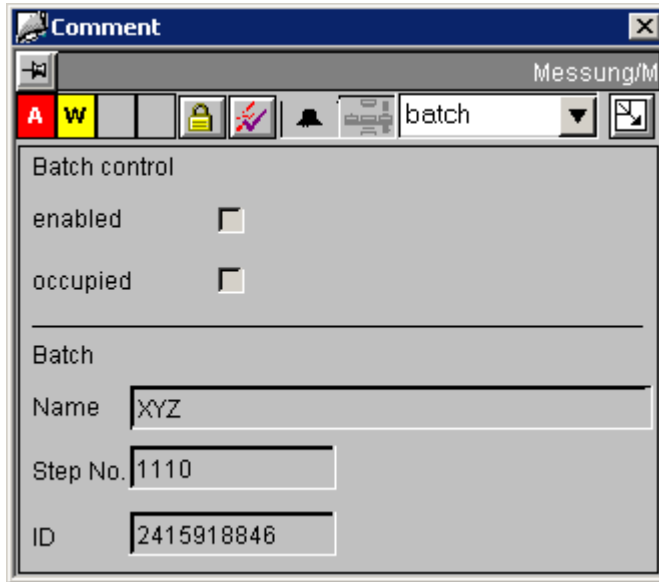
- CTRL_PID
- CTRL_S
- ELAP_CNT
- FMCS_PID
- FMT_PID
- DIG_MON
- DOSE
- MEAS_MON
- MOT_REV
- MOT_SPED
- MOTOR
- SWIT_CNT

- VAL_MOT
- VALVE

更多信息

更多相关信息，可参考：
总览中的对象 (页 654)

12.3.3 全局视图：批生产视图



包含批生产视图的面板

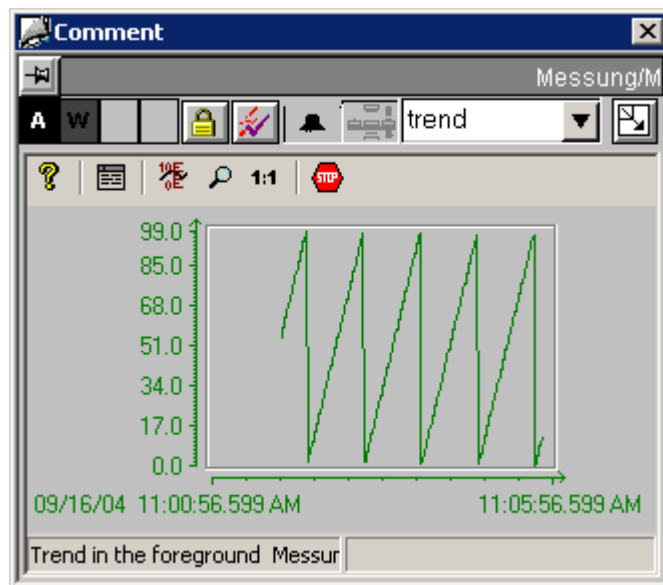
下列面板包含批生产视图：

- CTRL_PID
- CTRL_S
- FMCS_PID
- FMT_PID
- DIG_MON
- DOSE
- MEAS_MON
- MOT_REV
- MOT_SPED
- MOTOR
- VAL_MOT
- VALVE

更多信息

更多相关信息，可参考：
总览中的对象 (页 654)

12.3.4 全局视图：趋势视图



含有趋势视图的面板

下列面板包含趋势视图:

- CTRL_PID
- CTRL_S
- FMCS_PID
- FMT_PID
- DOSE
- MEAS_MON

说明

有关组态趋势变量的更多信息, 可参考手册《PCS 7 — 块编程指令》(PCS 7 - Programming Instructions for Blocks) 中的“组态趋势视图”一章。

更多信息

更多相关信息, 可参考:

总览中的对象 (页 654)

12.3.5 显示不带资产管理的“停止无效”

“OB_BEGIN”块图标

如果您的系统没有 ASSET 诊断，则在 OS 上提供了一个单独的块图标以显示模板 @Template.pdl 中的停止无效。该块图标以名称“OB_BEGIN”存储在“诊断”(Diagnostics) 部分中。



组态

为每个 AS 组态 OB_BEGIN 块图标。然后将每个块图标与相应的 OB_BEGIN 结构变量互连。

要实现与块图标的全部所需互连，最好使用 PCS 7 WinCC 向导将面板互连到过程标签。在变量对话框“所有结构变量的列表”(List of all structure variables) 中，可以选择相关的 OB_BEGIN 实例。

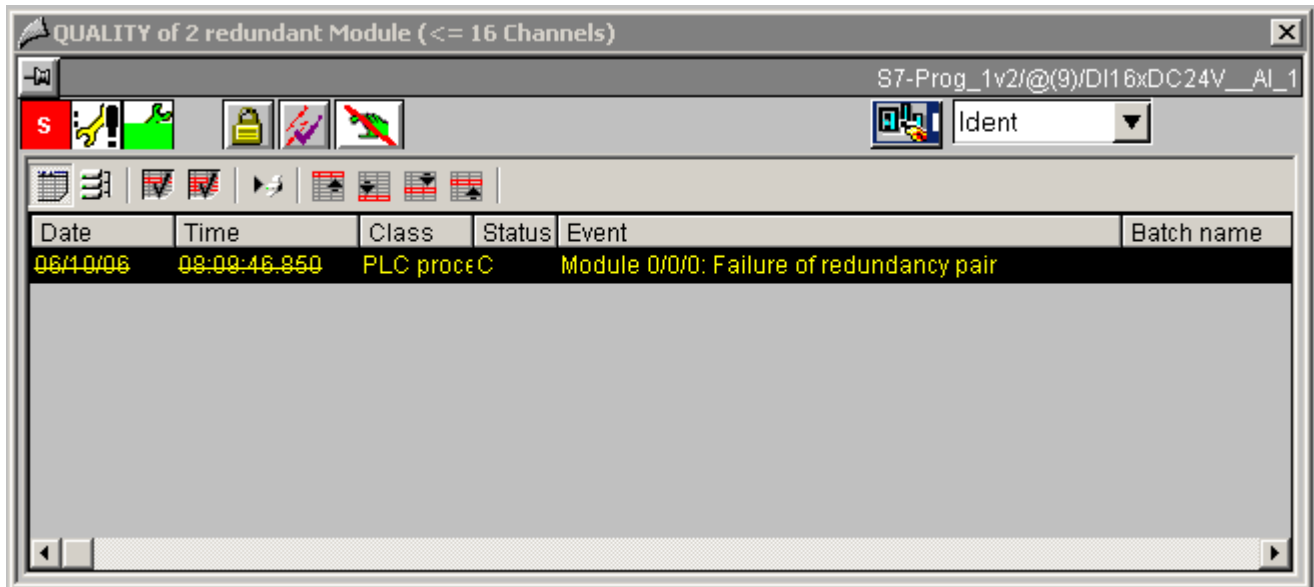
有关“OB_BEGIN”面板的说明

如果 AS 上支持 SFC78，则在不带资产管理的 OB_BEGIN 和 CPU_RT 块的 OB_BEGIN 面板中，将显示消息视图、性能视图和详细视图（OB3x 和 OB8x/OB1）。如果不支持 SFC78，则仅显示面板的消息视图。

不显示标识视图和参数视图。

12.3.6 消息视图 [资产]

布局



与全局消息视图的差异:

当前未决消息不管其确认状态为何都会进行显示。

除“tagname”过滤器外，仅会显示与“诊断”相关的消息。

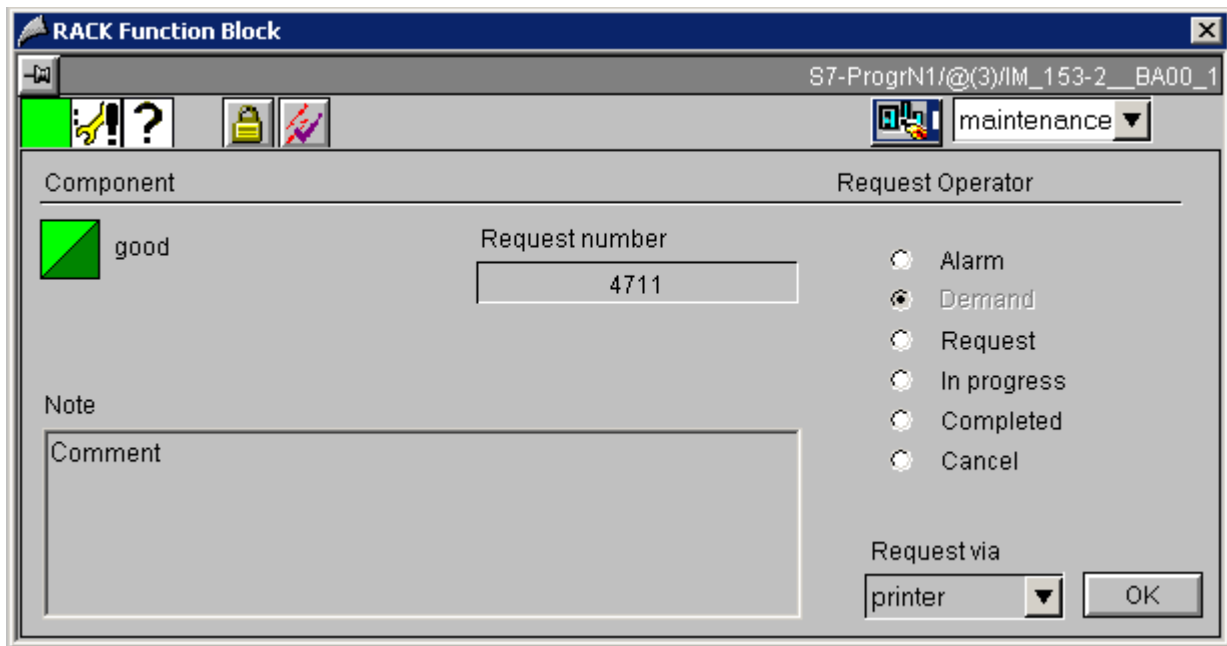
更多信息

更多相关信息，可参考:

资产面板的全局表示和视图 (页 671)

12.3.7 维护视图 [资产]

布局



资产面板的维护视图具有以下显示及操作员控制元素：

- **请求号 (Request number):** 在此显示分配给维护作业的作业号。
- **请求操作员 (Request operator):** 在此定义来自不同组件的维护请求。维护操作员将决定所设置的维护状态。可用的状态如下：
 - 报警 (Alarm)
 - 要求 (Demand)
 - 请求 (Request)
 - 进行中 (In progress)
 - 已完成 (Completed)
 - 取消 (Cancel)

如果单击所列出的任一单选按钮，将打开一个对话框，可在其中输入注释和作业号。请求状态图标随即会改变。

12.3 全局视图和表示

- **注 (Note):** 在此显示所输入的维护作业注释。
- **请求方式 (Request via):** （目前，仅支持“打印机”(Printer)）为所选面板打印输出报表

说明

上个作业完成后，才可以启动另一个打印作业。

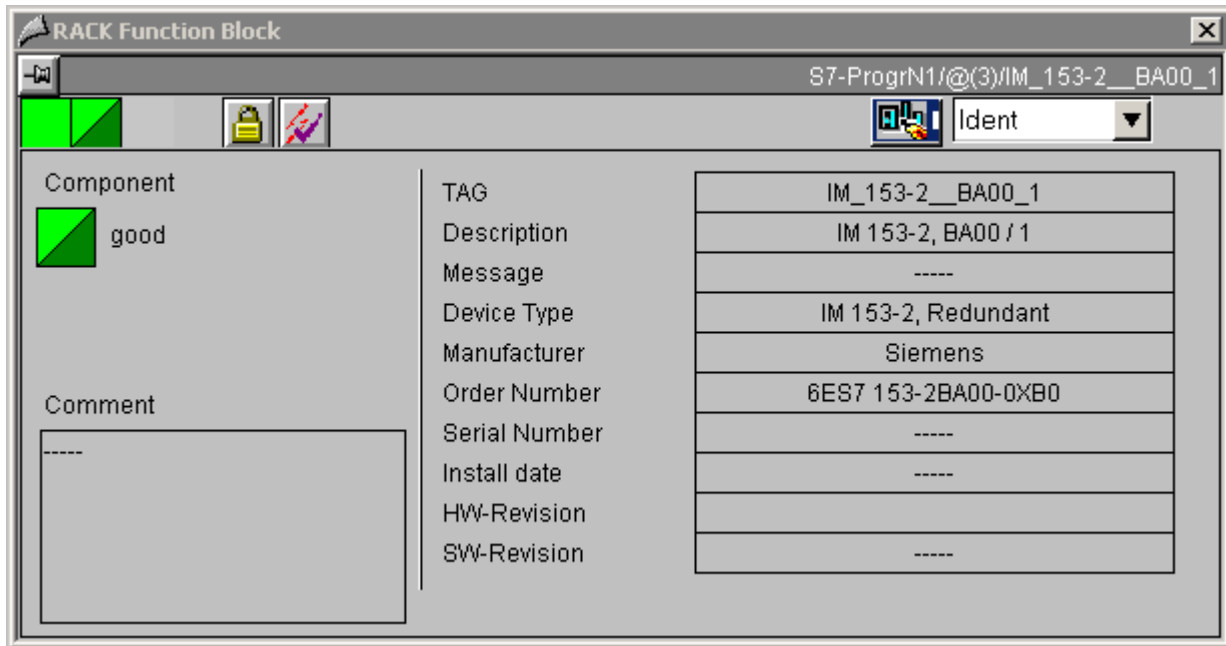
更多信息

更多相关信息，可参考：

资产面板的全局表示和视图 (页 671)

12.3.8 标识视图 [资产]

@PG_ASSETAS_Ident



标识页面的布局对于所有面板都是相同的；只有标识数据的动态采样不同。

例外：对于包含 Web 服务器的设备，提供了一个按钮以调用查看器来显示设备上的 HTML 页。



对于以下情况，此按钮显示在面板的总览行：

- 为 SNMP 组态了设备
- 属性“Web 兼容”(Web-compliant) 存储在块图标中。

此按钮用于 OSM 面板。由于某些原因无法访问网络组件时，也会显示此按钮。

请注意，对于 MS Windows 2003 Server 操作系统，要访问的网络组件的 IP 地址必须包含在“受信任的站点”(Trusted sites) 安全设置中。

AS 面板

在 AS 面板上，标识数据是从调用块图标的属性中读出的。创建/更新块图标时，会通过 HW Config 数据确定该信息。

标识数据	块图标中的属性	来源
TAG	“tag”	CFC“块名称”
描述	“HWComment”	HW Config“标识”/“名称”
通知 (Notification)	- - -	不可用
设备类型 (Device type)	“HWType”	HW Config“类型”
制造商 (Manufacturer)	“HWVendor”	不可访问
订货号 (Order number)	“HWOrderNo”	HW Config“订货号”
序列号 (Serial number)	- - -	不可用
安装日期 (Install date)	- - -	不可用
HW 修订版 (HW revision)	- - -	不可用
SW 修订版 (SW revision)	“HWSWVersion”	不可访问
注释 (Comment)	“HWComment”	HW Config“注释”
“不可访问”意味着用户不能在此进行任何输入或修改。		

PC 面板

当前没有任何标识数据显示在 PC 面板上。

IPC 面板

对于 IPC 面板，从数据管理器读取变量的标识数据。这些变量是使用 OPC 服务器属性中的“导出 WinCC 变量”功能，或使用“创建/更新诊断屏幕”功能然后对 OS 进行编译来创建的。

标识数据	变量/块图标中的属性	来源
TAG	“sysName”	Windows 计算机名称
描述	“sysLocation”	DiagMonitor
通知 (Notification)	“&ipAddress()”	Windows IP 地址
设备类型 (Device type)	“ProductName”	DiagMonitor
制造商 (Manufacturer)	“Manufacturer”	DiagMonitor
订货号 (Order number)	---	不可用
序列号 (Serial number)	“SerialNumber”	DiagMonitor
安装日期 (Install date)	---	不可用
HW 修订版 (HW revision)	“HwVersion”	DiagMonitor
SW 修订版 (SW revision)	“SwVersion”	DiagMonitor
注释 (Comment)	“&comment”	对象属性 PC 站“注释”

OSM 面板

对于 OSM 面板，从数据管理器读取变量的标识数据。这些变量是使用 OPC 服务器属性中的“导出 WinCC 变量”功能，或使用“创建/更新诊断屏幕”功能然后对 OS 进行编译来创建的。

标识数据	变量/块图标中的属性	来源
TAG	“sysName”	WBM *)“系统名称”
描述	“sysLocation”	WBM *)“系统位置”
通知 (Notification)	“&ipAddress()”	HW Config/OPC 服务器“IP 地址”
设备类型 (Device type)	“sysDescr”	WBM“设备类型”
制造商 (Manufacturer)	- - -	不可用
订货号 (Order number)	- - - / "snInfoOrderNr" **)	不可用/WBM“订货号”
序列号 (Serial number)	- - - / "snInfoSerialNr" **)	不可用/WBM“MAC 地址”
安装日期 (Install date)	- - -	不可用
HW 修订版 (HW revision)	- - - / "snHWVersion" **)	不可用/WBM“硬件”
SW 修订版 (SW revision)	- - - / "snSWVersion" **)	不可用/WBM“固件”
注释 (Comment)	“HWComment”	HW Config/OPC 服务器“注释”
*) WBM = 基于 Web 的管理（网络对象的 Web 界面）		
**) 如果使用了 SNMP“MIB-II”配置文件，则此数据不可用		

PDM 面板

在 PDM 面板上，标识数据通过 PDM 的 COM 接口以 XML 格式读出并显示。数据存储在下列标签下：

标识数据	标签/块图标中的属性	来源
TAG	"Device.App.Ident.TAG"	EDD PDM“TAG”
描述	"Device.App.Ident.Description"	EDD PDM“描述信息”
通知 (Notification)	"Device.App.Ident.Message"	EDD PDM“消息”
设备类型 (Device type)	"Device.Type.Ident.Type"	EDD PDM“产品标识”
制造商 (Manufacturer)	"Device.Type.Ident.Manufacturer"	EDD PDM“制造商”
订货号 (Order number)	"Device.Type.Ident.OrderNumber"	EDD PDM“设备订货号”
序列号 (Serial number)	"Device.Ident.SerialNumber"	EDD PDM“设备序列号”
安装日期 (Install date)	"Device.App.Ident.InstallDate"	EDD PDM“安装日期”
HW 修订版 (HW revision)	"Device.Type.Ident.HwRevision"	EDD PDM“硬件修订版”
SW 修订版 (SW revision)	"Device.Type.Ident.SwRevision"	EDD PDM“软件修订版”
注释 (Comment)	"Device.Type.Ident.Comment"	HW Config“注释”

ASSETMON 面板

在 ASSETMON 面板上，标识数据通过 PDM 的 COM 接口以 XML 格式读出并显示。数据存储在下列标签下：

标识数据	标签/变量	来源
TAG	"Device.App.Ident.TAG"	EDD PDM“TAG”
描述	"Device.App.Ident.Description"	EDD PDM“描述信息”
通知 (Notification)	"Device.App.Ident.Message"	EDD PDM“安装位置”
设备类型 (Device type)	"Device.Type.Ident.Type"	不可用
制造商 (Manufacturer)	"Device.Type.Ident.Manufacturer"	EDD PDM“制造商”
订货号 (Order number)	"Device.Type.Ident.OrderNumber"	EDD PDM“设备订货号”
序列号 (Serial number)	"Device.Ident.SerialNumber"	EDD PDM“序列号”
安装日期 (Install date)	"Device.App.Ident.InstallDate"	EDD PDM“安装日期”
HW 修订版 (HW revision)	"Device.Type.Ident.HwRevision"	EDD PDM“硬件修订版”
SW 修订版 (SW revision)	"Device.Type.Ident.SwRevision"	EDD PDM“软件修订版”
注释 (Comment)	"&comment"	CFC“块注释”

更多信息

更多相关信息，可参考：

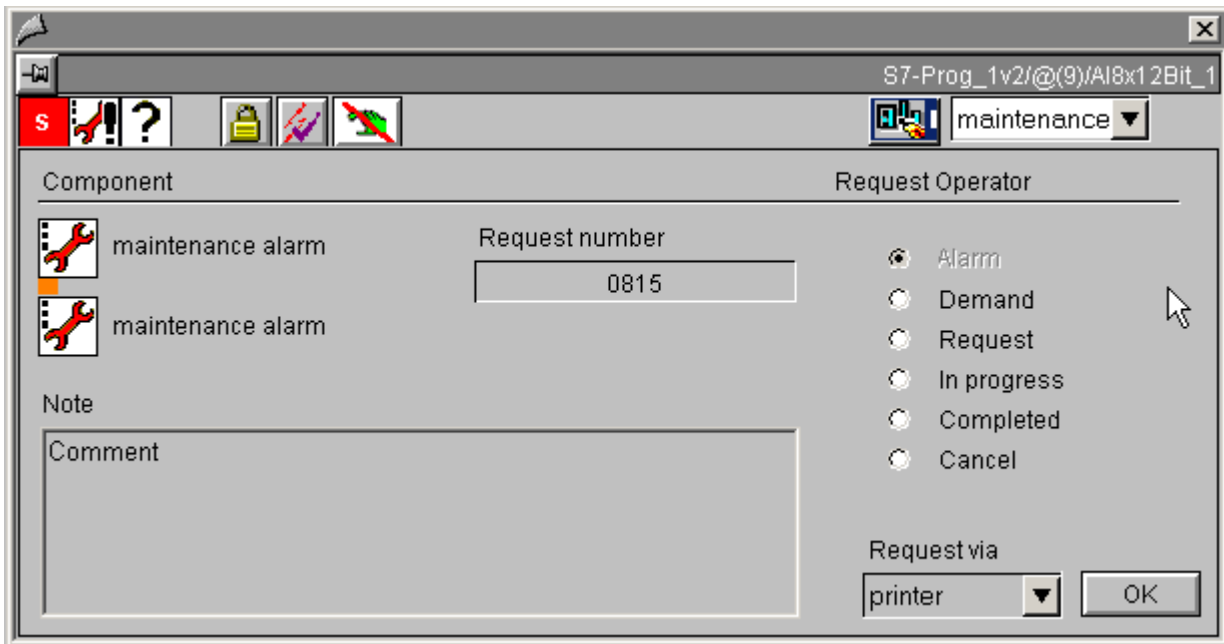
资产面板的全局表示和视图 (页 671)






12.3.9 资产面板的全局表示和视图




简介

以下表示和视图在所有资产面板中都会出现。

诊断和维护图标



面板中的图标	含义	图标
	<p>含有维护报警的组显示</p> <p>指示用于表示层级组显示中组件的组请求状态。</p>	<p>可能的组显示状态</p> <ul style="list-style-type: none">  维护报警  维护请求  需要维护  良好

面板中的图标	含义	图标
	<p>含有维护报警的维护状态显示</p> <p>指示组件的当前的整体状态。它由冗余组件以及操作员响应/操作的各个状态的总和构成。</p> <p>如果操作员未进行任何设置，则在状态显示中会显示“操作员操作前”(Before operator action) 下列出的 8 个图标。</p> <p>如果操作员执行了状态更改，则在状态显示中会显示“操作员操作后”(After operator action) 下列出的 3 个图标。</p> <p>操作员可在面板“维护”视图中更改维护状态。</p>	<p>操作员操作前：</p> <ul style="list-style-type: none">  维护报警  维护请求  需要维护  组件良好  本地操作  至少要模拟一个过程值  服务中断  组件钝化  未测试/未知  组态已更改 <p>操作员操作后：</p> <ul style="list-style-type: none">  状态变为维护报警  状态变为需要维护  状态变为维护请求

面板中的图标	含义	图标
	<p>图标所显示的状态</p> <p>问号 = 状态不清楚或尚不明确。 指示组件的当前处理状态。</p> <ol style="list-style-type: none"> 1. 正常状态（无未决诊断） --> 无图标。 2. 无操作员响应的未决诊断 --> 无图标。 3. 操作员以确认/重新分类/作业形式触发的操作 --> “?”图标；组件上无活动。 4. 操作进行中 --> “进行中”(In progress) 图标 	 维修 组件（进行中）
	<p>组件状态显示</p> <p>如果“组件”(Component)下方显示 2 个图标，则它们表示组件自身及其冗余组件。</p> <p>如果组件不是冗余的，则“组件”(Component) 下方仅会出现一个带有文本的图标。</p> <p>“组件”(Component) 下方列出的图标指示 AS 所报告的原始组件状态。</p>	<p>状态显示中的可能状态</p> <ul style="list-style-type: none">  维护报警  维护请求  需要维护  组件良好  本地操作/本地覆盖  至少要模拟一个过程值  服务中断  组件钝化  未测试/未知  组态已更改

说明

如果维护状态为“未测试/未知”，则资产管理面板中的其它所有动态显示均与此实例无关。

总览



资产面板的总览中添加了以下各键，总览的外观始终相同：

- **@PG_ASSETAS_Overview** 中添加了用于调用 HW Config 的键：



仅当打开面板的计算机上也安装了 STEP 7 时，该图标才会出现。

- **@PG_ASSETPDM_Overview** 中添加了用于调用 PDM 和 HW Config 的键：



仅当打开面板的计算机上也安装了 PDM 和 STEP 7 时，这些键才会出现。

- **@PG_ASSETAS_Overview** 中添加了用于取消钝化的键：



仅当打开面板的计算机上也安装了 STEP 7 时，该图标才会出现。

更多信息

更多相关信息，可参考：

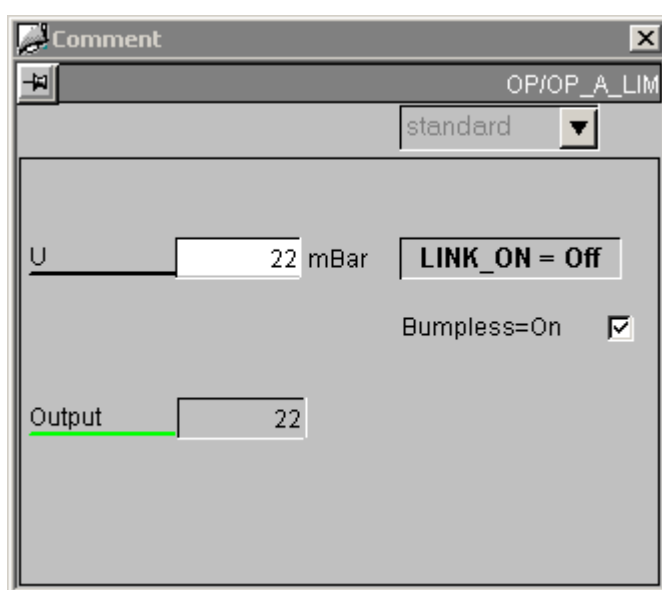
总览中的对象 (页 654)

12.4 面板： 操作员控制块

12.4.1 OP_A: 标准视图

请参见：OP_A_LIM: 标准视图 (页 676)。

12.4.2 OP_A_LIM: 标准视图



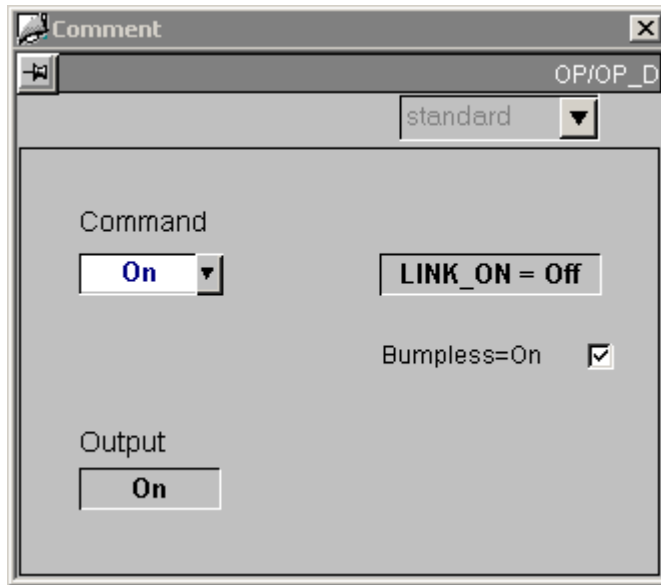
与控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能
权限设定值	-->	LevelSource
权限设定值	-->	Target_Operator-ControlEnable
Setpoint_PCS7_AnalogValue	-->	操作员控制使能
@Level6	-->	操作员控制使能
BumplessOn_CHECKBOX_L	-->	操作员控制使能

12.4.3 OP_A_RJC: 标准视图

请参见： OP_A_LIM: 标准视图 (页 676)。

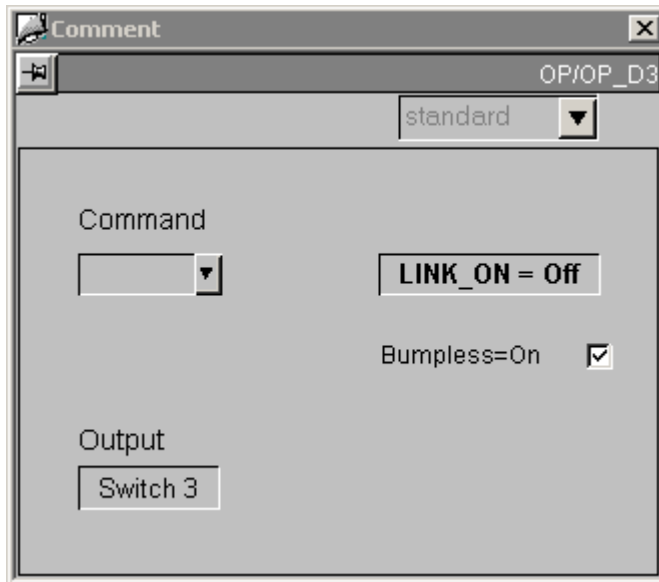
12.4.4 OP_D: 标准视图



与控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能
I0_PCS7_COMBOBOX	-->	操作员控制使能
@Level5	-->	BackColor
I0_PCS7_COMBOBOX	-->	背景色文本 1 (BackColor Text1)
I0_PCS7_COMBOBOX	-->	背景色文本 2 (BackColor Text2)
@Level6	-->	操作员控制使能
BumplessOn_CHECKBOX_L	-->	操作员控制使能

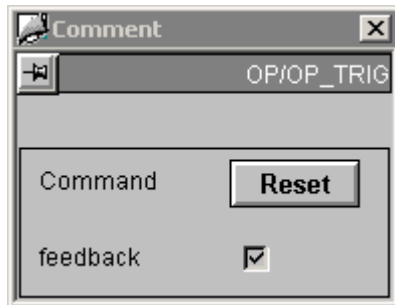
12.4.5 OP_D3: 标准视图



与控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能
Command_PCS7_3COMBOBOX	-->	操作员控制使能
@Level5	-->	BackColor
Command_PCS7_3COMBOBOX	-->	背景色文本 1 (BackColor Text1)
Command_PCS7_3COMBOBOX	-->	背景色文本 2 (BackColor Text2)
Command_PCS7_3COMBOBOX	-->	背景色文本 3 (BackColor Text3)
@Level6	-->	操作员控制使能
BumplessOn_CHECKBOX_L	-->	操作员控制使能

12.4.6 OP_TRIG: 标准视图



标有“复位”(Reset) 的键会将逻辑 1 写入 OP_TRIG 块的“IO”参数。

“复位”(Reset) 文本读取自参数的“s7_shortcut”属性，并可针对具体情况进行修改。

如果选中“反馈”(Feedback) 复选框，则会读取 OP_TRIG 块的“SIGNAL”参数。

“反馈”(Feedback) 文本在面板中进行组态，因而是类型特定文本。

访问控件

除 WinCC 授权级别外，“Permission_Reset”权限对象也会对“QOP_EN = TRUE”参数进行评估。

与控制对象直接连接的顺序和定位

@Level5	-->	操作员控制使能
Permission_Reset	-->	Level_Source
Permission_Reset	-->	Target_Operator-ControlEnable
BumplessOn_CHECKBOX_L1	-->	操作员控制使能

12.4 面板： 操作员控制块

块图标

13.1 块图标的常规属性

基本属性

不得修改所有“@@PCS7Typicals”画面块图标的下列属性：

- 几何形状/Width
- 几何形状/Height
- 其它/OP_enabled
- 其它/Password
- 其它/Display
- 常规/Servername
- 样式/Group-relevant（仅用于包含 ALARM_8P 消息的块）

存在于所有块符号中的属性：

属性	用户对象中的元素和属性	对象	描述
其它/Processcontrolling_backup	POP.Permission	I/O box	实例特定的控制权限 默认值 = 5
其它/HigherProcesscontrolling_backup	HIPOP.Permission	I/O box	实例特定的控制权限 默认值 = 6
常规/tag	NameOfTag.OutputValue	I/O box	图标中显示的文本
常规/type	Type.OutputValue	I/O box	从 PH 生成图标及向导的参考
常规/tagname	Tagname.OutputValue	I/O box	传递给画面窗口的变量前缀的实际变量名
常规/Servername	Servername.OutputValue	I/O box	块/面板类型
常规/Version	Version.OutputValue	I/O box	版本号

13.1 块图标的常规属性

属性	用户对象中的元素和属性	对象	描述
样式/View_Tag	NameOfTag.Display Rectangle17.Display (如果存在)	矩形 I/O box	可用于隐藏变量名
MouseClicked left	PCS7_OpenGroupDisplay_V6(lpszPictureName, lpszObjectName)		调用面板

更新工厂层级时不会被覆盖的块图标的属性:

存在于所有块图标中的属性:

- HigherProcesscontrolling_backup
- Processcontrolling_backup
- View_Tag

不存在于所有块图标中的属性:

- ReturnPath
- StandardTrend
- Format_InputValue
- Format_OutputValue
- Format_xx

更多信息

更多信息, 请参考以下部分:

面板的位置 (页 683)

突出显示“报警循环”和“通过过程标签选择画面”的块图标 (页 684)

13.2 面板的位置

指定位置

块图标打开时可指定面板的位置。

每个块图标有三个用于此用途的属性：

- **DefaultPos**。如果该属性设置为 **TRUE**，面板调用行为与从前相同；这也是默认设置。如果该属性设置为 **FALSE**，则在通过属性“leftPos/topPos”指定的位置调用面板。
- **LeftPos** = 面板左上点所在的水平位置。
- **TopPos** = 面板左上点所在的垂直位置。

13.3 突出显示“报警循环”和“通过过程标签选择画面”的块图标

设置突出显示

“报警循环”和“通过过程标签选择画面”功能可使程序跳到过程标签的相应过程画面。

过程标签的关联块图标是使用“HighlightBlockIcon”功能突出显示的。“tagname”变量以青蓝色突出显示。



该块图标有两个用于此用途的属性：

- **HighLightBlockIcon** — 通过功能“报警循环”和“通过过程标签选择画面”进行设置，以青蓝色突出显示“tagname”变量。
- **HighLightBlockIconBackColor** — 用于突出显示的颜色，默认设置是“青色”(cyan)。

13.4 块图标：设备块

13.4.1 块图标：CTRL_S

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 110/Height = 77		
常规/UnitPV	Unit.Text/.PV_IN#unit	Stat.Text	显示：PV 单位
常规/Unit_MAN_OP	Unit.Text/.MAN_OP#unit	Stat.Text	显示：调节变量单位
链接/CollectValue	GroupDisplay.CollectValue/.EventState	组显示	
链接/SetpointValue	SetpointValue_AdvancedAnalogDisplay.Value	AdvancedAnalogDisplay	显示：设定值
链接/ProcessValue	ProcessValue_AdvancedAnalogDisplay.Value	AdvancedAnalogDisplay	显示：过程值
链接/OutputValue	OutputValue_AdvancedAnalogDisplay.Value	AdvancedAnalogDisplay	显示：调节变量
链接/Mode_MAN_AUT	Manual_AdvancedStatusDisplay.Status	AdvancedStatusDisplay	显示：手动/自动
链接/Mode_INT_EXT	External_AdvancedStatusDisplay.Status	AdvancedStatusDisplay	显示：外部/内部
链接/LMN_SEL	Tracking_AdvancedStatusDisplay.Status	AdvancedStatusDisplay	显示：调节变量校正

13.4 块图标：设备块

属性	用户对象中的元素和属性	对象	描述
链接/QLMNR_ON	OutputValue_AdvancedAnalog Display.Display Unit_MAN_OP.Display	AdvancedAnalogDis . Stat.Text	请参见以下 说明
链接/QLMNUP	LMNUP_StatusDisplay	Stat.Text	显示： QLMNUP
链接/QLMNDN	LMNDN_StatusDisplay	Stat.Text	显示： QLMNDN
样式/ReturnPath	TrendFunctions2 .OutputValue	I/O box	
样式/StandardTrend	TrendFunctions2 .FontSize	I/O box	
样式/Format_InputValue	ProcessValue_AdvancedAnalog Display.Format SetpointValue_AdvancedAnalog Display.Format	AdvancedAnalogDis . AdvancedAnalogDis .	设置过程值和设定 值的数字格式
样式/Format_OutputValue	OutputValue_AdvancedAnalog Display.Format	AdvancedAnalogDis .	设置调节变量的数 字格式
样式/Format_xx	Format_xx.OutputValue	I/O box	其它格式设置

CTRL_S 块图标与 CTRL PID 块图标的不同之处如下： 如果位置反馈 (LMNR_ON = 0) 不可用，则程序显示二进制控制信号 QLMNUP 和 QLMNDN 而不是调节变量。

也可通过脚本控制这些文本的可见性。更改 QLMNUP 和 QLMNDN 属性时会调用脚本。

注意： 为确保可视化控制正常工作，“OutputValue_AdvancedAnalogDisplay”和“Unit_MAN_OP”对象必须总是置于用户对象中的前台。

13.4.2 块图标：CTRL_PID

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 110/Height = 77		
常规/UnitPV	UnitPV.Text	Stat.Text	显示：PV 单位
常规/Unit_MAN_OP	Unit_MAN_OP.Text	Stat.Text	显示：MAN_OP 单位
链接/CollectValue	GroupDisplay.CollectValue	GroupDisplay.	
链接/SetpointValue	SetpointValue_AdvancedAnalog Display.Value	AdvancedAnalogDis.	显示：设定值
链接/ProcessValue	ProcessValue_AdvancedAnalog Display.Value	AdvancedAnalogDis.	显示：过程值
链接/OutputValue	OutputValue_AdvancedAnalog Display.Value	AdvancedAnalogDis.	显示：调节变量
链接/LMN_SEL	Tracking_AdvancedStatus Display.Status	AdvancedStatusDis.	显示：调节变量校正
链接/Mode_MAN_AUT	Manual_AdvancedStatus Display.Status	AdvancedStatusDis.	显示：手动/自动
链接/Mode_INT_EXT	External_AdvancedStatus Display.Status	AdvancedStatusDis.	显示：外部/内部
样式/ReturnPath	TrendFunctions2 .OutputValue	I/O box	
样式/StandardTrend	TrendFunctions2 .FontSize	I/O box	

13.4 块图标：设备块

属性	用户对象中的元素和属性	对象	描述
样式/Format_InputValue	ProcessValue_AdvancedAnalog Display.Format SetpointValue_AdvancedAnalog Display.Format	AdvancedAnalogDis. AdvancedAnalogDis.	设置过程值和设定值的数字格式
样式/Format_OutputValue	OutputValue_AdvancedAnalog Display.Format	AdvancedAnalogDis.	设置调节变量的数字格式
样式/Format_xx	Format_xx.OutputValue	I/O box	其它格式设置

13.4.3 块图标：DIG_MON

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 90/Height = 40		
链接/Status	StatusDisplay.ActualStatus	状态显示	.Q
链接/CollectValue	GroupDisplay3.CollectValue	组显示	.EventState

13.4.4 块图标：DOSE

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 110/Height = 63		
常规/UnitPV	UnitPV.Text	Stat.Text	显示： PV 单位
链接/CollectValue	GroupDisplay.CollectValue	组显示	.EventState
链接/ProcessValue	ProcessValue_AdvancedAnalog Display.Value	AdvancedAnalogDis .	显示： 过程值
链接/SetpointValue	SetpointValue_AdvancedAnalog Display.Value	AdvancedAnalogDis .	显示： 设定值
链接/SetpointExternal	External_AdvancedStatus Display.Status SetpointExternValue_ AdvancedAnalogDisplay.Display	AdvancedStatusDis. AdvancedAnalogDis .	.QSPEXTON 请参见以下说明
链接 /SetpointExternalValue	SetpointExternValue_ AdvancedAnalogDisplay.Value	AdvancedAnalogDis .	使用设定值显 示 .QSPEXTON
样式/ReturnPath	TrendFunctions2 .OutputValue	I/O box	
样式/StandardTrend	TrendFunctions2 .FontSize	I/O box	
样式/Format_InputValue	ProcessValue_AdvancedAnalog Display.Format SetpointValue_AdvancedAnalog Display.Format	AdvancedAnalogDis . AdvancedAnalogDis .	设置过程值和设定 值的数字格式

13.4 块图标：设备块

属性	用户对象中的元素和属性	对象	描述
样式/Format_OutputValue	OutputValue_AdvancedAnalog Display.Format	AdvancedAnalogDis .	设置调节变量的数字格式
样式/Format_xx	Format_xx.OutputValue	I/O box	其它格式设置

DOSE 块不包含表示有效设定值的参数。因此，程序输出设定值显示（取决于 QSPEXTON 的状态）。

QSPEXTON = 0 --> 将显示“SetpointValue_AdvancedAnalogDisplay”。

QSPEXTON = 1 --> 将显示“SetpointExternValue_AdvancedAnalogDisplay”。

13.4.5 块图标：ELAP_CNT

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 97/Height = 45		
常规/Unit	Unit.Text	Stat.Text	
链接/CollectValue	GroupDisplay.CollectValue	组显示	.EventState
链接/Output_Value	ProcessValue_AdvancedAnalog Display.Value	AdvancedAnalogDis .	.HOURS 显示：最多 7 位
样式/Format_InputValue	ProcessValue_AdvancedAnalog Display.Format	AdvancedAnalogDis .	设置过程值和设定值的数字格式
样式/Format_OutputValue	Format_OutputValue .OutputValue	I/O box	设置调节变量的数字格式
样式/Format_xx	Format_xx.OutputValue	I/O box	其它格式设置

13.4.6 块图标： FMCS_PID

属性



更多相关信息，请参考以下部分： "块图标的常规属性 (页 681)"

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 110/Height = 77		
常规/UnitPV	Unit.Text/.PV#unit	Stat.Text	
常规/Unit_MAN_OP	Unit.Text/.LMN#unit	Stat.Text	
链接/CollectValue	GroupDisplay.CollectValue/.EventState	组显示	
链接/SetpointValue	SetpointValue_AdvancedAnalogDisplay.Value/.SP	AdvancedAnalogDis.	显示：设定值
链接/ProcessValue	ProcessValue_AdvancedAnalogDisplay.Value/.PV	AdvancedAnalogDis.	显示：过程值
链接/OutputValue	OutputValue_AdvancedAnalogDisplay.Value/.LMN	AdvancedAnalogDis.	显示：调节变量
链接/Tracking	Tracking_AdvancedStatusDisplay.Status	AdvancedStatusDis.	显示：跟踪 LMN
链接/Mode_MAN_AUT	Manual_AdvancedStatusDisplay.Status	AdvancedStatusDis.	显示：手动/自动
链接/Mode_INT_EXT	External_AdvancedStatusDisplay.Status	AdvancedStatusDis.	显示：外部/内部
样式/ReturnPath	TrendFunctions2.OutputValue	I/O box	

属性	用户对象中的元素和属性	对象	描述
样式/StandardTrend	TrendFunctions2 .FontSize	I/O box	
样式/Format_InputValue	ProcessValue_AdvancedAnalog Display.Format SetpointValue_AdvancedAnalog Display.Format	AdvancedAnalogDis. AdvancedAnalogDis.	设置过程值和设定值的数字格式
样式/Format_OutputValue	OutputValue_AdvancedAnalog Display.Format	AdvancedAnalogDis.	设置调节变量的数字格式
样式/Format_xx	Format_xx.OutputValue	I/O box	其它格式设置

13.4.7 块图标： FMT_PID

属性

tagname	
999999,9	
999999,9	单位
999999,9	
M	I T A

有关属性的更多信息，可参考以下部分：“FMCS_PID 块图标 (页 692)”

13.4 块图标：设备块

13.4.8 块图标：INTERLOK

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 108/Height = 20		
链接/Link	Lock.ActualStatus	StatDis.	锁定图标

13.4.9 块图标：MEAS_MON

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 97/Height = 45		
常规/Unit	Unit.Text	Stat.Text	
链接/CollectValue	GroupDisplay.CollectValue	组显示	
链接/OutputValue	ProcessValue_AdvancedAnalog Display.Value	AdvancedAnalogDis.	显示：过程值
样式/ReturnPath	TrendFunctions2.OutputValue	I/O box	
样式/StandardTrend	TrendFunctions2 .FontSize	I/O box	
样式/Format_InputValue	ProcessValue_AdvancedAnalog Display.Format	AdvancedAnalogDis.	设置过程值和设定值的数字格式

属性	用户对象中的元素和属性	对象	描述
样式/Format_OutputValue	Format_OutputValue .OutputValue	I/O box	设置调节变量的数字格式
样式/Format_xx	Format_xx.OutputValue	I/O box	其它格式设置

13.4.10 块图标：MOT_REV

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 90/Height = 53		
链接/CollectValue	GroupDisplay_WithASD1.CollectValue	AdvancedStatusDis.	
链接/QMAN_AUT	Mode.Status1	AdvancedStatusDis.	显示：自动/手动
链接/LOCK	Interlock.Status1	AdvancedStatusDis.	显示：锁定
链接/QRUN	Motor_Status1.Status1	AdvancedStatusDis.	显示：电机
链接/QSTOP	Motor_Status1.Status2	AdvancedStatusDis.	显示：电机
链接/QDIR	Motor_Status1.Status3	AdvancedStatusDis.	显示：电机

单击左键可调用 MOT_REV 面板。单击右键可调用关联的 INTERLOK 面板。

INTERLOK 块的名称作为脚本传送参数存储。

默认的块名称为“ILOCK”。INTERLOK 块和 MOT_REV 必须置于同一 CFC 图表中。

13.4.11 块图标：MOT_SPED

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 90/Height = 53		
链接/CollectValue	GroupDisplay_WithASD1.CollectValue	AdvancedStatusDis.	
链接/QMAN_AUT	Mode.Status1	AdvancedStatusDis.	显示：自动/手动
链接/LOCK	Interlock.Status1	AdvancedStatusDis.	显示：锁定
链接/QRUN	Motor_Status.Status1	AdvancedStatusDis.	显示：电机
链接/QSTOP	Motor_Status.Status2	AdvancedStatusDis.	显示：电机
链接/QSPEED	Motor_Status.Status3	AdvancedStatusDis.	显示：电机
链接/QSTOPING	Motor_Status.Status4	AdvancedStatusDis.	显示：电机

单击左键可调用 MOT_SPED 面板。单击右键可调用关联的 INTERLOK 面板。

INTERLOK 块的名称作为脚本传送参数存储。

默认的块名称为“ILOK”。INTERLOK 块和 MOT_SPED 必须置于同一 CFC 图表中。

13.4.12 块图标：MOTOR

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 90/Height = 54		
链接/CollectValue	GroupDisplay_WithASD.CollectValue	AdvancedStatusDis.	
链接/QMAN_AUT	Mode.Status1	AdvancedStatusDis.	显示：自动/手动
链接/LOCK	Interlock.Status1	AdvancedStatusDis.	显示：锁定
链接/QRUN	Motor_Status.Status1	AdvancedStatusDis.	显示：电机
链接/QSTOP	Motor_Status.Status2	AdvancedStatusDis.	显示：电机

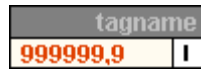
单击左键可调用 MOTOR 面板。单击右键可调用关联的 INTERLOK 面板。

INTERLOK 块的名称作为脚本传送参数存储。

默认的块名称为“ILOCK”。INTERLOK 块和 MOTOR 必须置于同一 CFC 图表中。

13.4.13 块图标：RATIO_P

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 97/Height = 32		
常规/Unit	Unit.Text	Stat.Text	显示： 单位
链接/OutputValue	ProcessValue_AdvancedAnalog Display.Value	AdvancedAnalogDis .	显示： 过程值
链接/Mode_INT_EXT	External_AdvancedStatus Display.Status	AdvancedStatusDis.	显示： 外部/内部
样式/Format_InputValue	ProcessValue_AdvancedAnalog Display.Format	AdvancedAnalogDis .	设置过程值和设定值的数字格式
样式/Format_OutputValue	Format_OutputValue.OutputValue	I/O box	设置调节变量的数字格式
样式/Format_xx	Format_xx.OutputValue	I/O box	其它格式设置

13.4.14 块图标：SWIT_CNT

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 97/Height = 45		
常规/Unit	Unit.Text	Stat.Text	.V#UNIT
链接/CollectValue	GroupDisplay.CollectValue	组显示	.EventState
链接/OutputValue	ProcessValue_AdvancedAnalog Display.Value	AdvancedAnalogDis .	显示：过程值
样式/Format_InputValue	ProcessValue_AdvancedAnalog Display.Format	AdvancedAnalogDis .	设置过程值和设定值的数字格式
样式/Format_OutputValue	Format_OutputValue.OutputValue	I/O box	设置调节变量的数字格式
样式/Format_xx	Format_xx.OutputValue	I/O box	其它格式设置

13.4.15 块图标：VAL_MOT

属性

块图标：VALVE (页 701) 的属性和显示

13.4.16 块图标： VALVE

属性



更多相关信息，请参考以下部分：“块图标的常规属性 (页 681)”

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 90/Height = 67		
链接/CollectValue	GroupDisplay_WithASD.CollectValue	AdvancedStatusDis.	
链接/QMAN_AUT	Mode.Status1	AdvancedStatusDis.	显示： 自动/手动
链接/V_LOCK	Interlock.Status1	AdvancedStatusDis.	显示： 锁定
链接/QOPENED	Valve_Status.Status1	AdvancedStatusDis.	显示： 阀
链接/QCLOSED	Valve_Status.Status2	AdvancedStatusDis.	显示： 阀
链接/QOPENING	Valve_Status.Status3	AdvancedStatusDis.	显示： 阀
Links/QCLOSING	Valve_Status.Status4	AdvancedStatusDis.	显示： 阀

单击左键可调用 VALVE 面板。单击右键可调用 INTERLOK 面板。

INTERLOK 块的名称作为脚本传送参数存储。

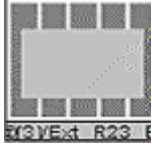
默认的块名称为“ILOK”。INTERLOK 块和 VALVE 块必须置于同一 CFC 图表中。




















13.5 块图标：资产管理






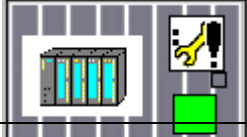


13.5.1 块图标：资产管理

系统支持的块图标

系统支持的用于资产管理的块图标包含在“@@maintenancetypicals.pdl”画面中。此文件位于“..\SIEMENS\WinCC\options\pdl\faceplatedesigner_v6”下，并在生成 PCS 7 项目或运行 OS 项目编辑器时，将其从安装目录复制到项目目录的“..\wincproj\

块图标中的图标	含义	本地操作
	带有自诊断图标（右上方）和包含下级层级状态的组显示（右下方）的 诊断块图标	

块图标中的图标	含义	本地操作
	<p>组件的“自诊断（维护状态）”图标</p> <p>根据维护状态，右侧列中显示的图标可在组件内显示。这只适用于非冗余组件。</p> <p>无论显示哪个图标，它都用于打开组件的面板。</p> <p>“冗余组件 [资产] 的状态显示 (页 735)”下显示的表格适用于冗余组件。</p>	<p> 维护报警</p> <p> 维护请求</p> <p> 需要维护</p> <p> 组件良好</p> <p> 本地操作</p> <p> 至少要仿真一个过程值</p> <p> 服务中断</p> <p> 组件钝化</p> <p> 未检查/未知</p> <p> 正在进行维护</p> <p> 维护作业已请求，报警优先级</p> <p> 维护作业已请求，要求优先级</p> <p> 维护作业已请求，请求优先级</p>
	<p>同级组件的组显示</p> <p>右下方的小方块（组显示）指示下列内容：</p> <p>报警尚未确认。报警也可能已结束。</p> <p>报警保持闪烁直至得到确认。如果报警仍待确认，图标将停止闪烁。如果报警不再等待确认，图标会消失。</p>	<p>组显示中的可能图标</p> <p> 维护报警</p> <p> 维护请求</p> <p> 需要维护</p> <p> 良好</p>

块图标中的图标	含义	本地操作
	<p>下级层级的组显示</p> <p>例如，维护报警在嵌套诊断画面中待确认。</p> <p>只显示优先级最高的内容。</p> <p>无论在组显示中显示哪个图标，左侧图标都用于打开下级层级。</p>	<p>组显示中的可能图标</p> <p> 维护报警</p> <p> 维护请求</p> <p> 需要维护</p> <p> 良好</p> 
	<p>Devicelcon</p> <p>如果 HW Config 包含位图，则显示设备图标的位图将在“Devicelcon”属性中显示。</p>	

13.6 块图标: 操作员控制块

13.6.1 块图标: OP_A

属性



更多相关信息, 请参考以下部分: "块图标的常规属性 (页 681)"

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 97/Height = 32		
常规/Unit	Unit.Text	Stat.Text	
链接/OutputValue	ProcessValue_AdvancedAnalog Display.Value	AdvancedAnalogDis .	显示: 过程值
样式/Format_InputValue	ProcessValue_AdvancedAnalog Display.Format	AdvancedAnalogDis .	设置过程值和设定值的数字格式
样式/Format_OutputValue	Format_OutputValue .OutputValue	I/O box	设置调节变量的数字格式
样式/Format_xx	Format_xx.OutputValue	I/O box	其它格式设置

13.6.2 块图标: OP_A_LIM

属性

块图标: OP_A (页 705) 的属性和显示。

13.6.3 块图标: OP_A_RJC

属性

块图标: OP_A (页 705) 的属性和显示。

13.6 块图标： 操作员控制块

13.6.4 块图标： OP_D

属性



更多相关信息，请参考以下部分： "块图标的常规属性 (页 681)"

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 90/Height = 45		
链接/Status	StatusDisplay.ActualStatus	状态显示	.Q0

13.6.5 块图标： OP_D3

属性



更多相关信息，请参考以下部分： "块图标的常规属性 (页 681)"

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 90/Height = 45		
链接/Output1	StatusDisplay1.Display	状态显示	.Q1
链接/Output2	StatusDisplay2.Display	状态显示	.Q2
链接/Output3	StatusDisplay3.Display	状态显示	.Q3

13.6.6 块图标: OP_TRIG

属性



更多相关信息, 请参考以下部分: "块图标的常规属性 (页 681)"

属性	用户对象中的元素和属性	对象	描述
几何形状	Width = 90/Height = 40		
链接/Status	StatusDisplay.ActualStatus	状态显示	.SIGNAL

13.6 块图标： 操作员控制块

附录

14.1 SM 模块的 MODE 设置

模拟输入模块的测量范围编码

根据模拟输入模块的测量范围编码，与通道对应的参数 **MODE_xx**（测量范围编码）必须依照下表指定。使用热电偶时，有多种方法来组合测量类型（编码 **A**）和测量范围（编码 **B**）。在这种情况下，必须按照以下公式计算参数 **MODE_xx**，然后将结果作为 **INTEGER** 值写入到 **MODE** 输入中：

$$\text{MODE} = 256 * \text{代码 A} + \text{代码 B}$$

请注意：下表以二进制格式显示代码 **A** 和 **B**，并以十六进制格式显示 **MODE** 参数中的结果。

测量类型	代码 (A)	测量范围	代码 (B)	MODE (256*A+B)
停用				16#0000
电压	2#0001	± 25 mV	2#1010	16#010A
		± 50 mV	2#1011	16#010B
		± 80 mV	2#0001	16#0101
		± 250 mV	2#0010	16#0102
		± 500 mV	2#0011	16#0103
		± 1 V	2#0100	16#0104
		± 2.5 V	2#0101	16#0105
		± 5 V	2#0110	16#0106
		1 至 5 V	2#0111	16#0107
		0 至 10 V	2#1000	16#0108
		± 10 V	2#1001	16#0109
		± 100 mV	2#1100	16#010C

14.1 SM 模块的 MODE 设置

测量类型	代码 (A)	测量范围	代码 (B)	MODE (256*A+B)
4 线测量传感器	2#0010	± 3.2 mA	2#0000	16#0200
		± 5 mA	2#0101	16#0205
		± 10 mA	2#0001	16#0201
		0 至 20 mA	2#0010	16#0202
		4 至 20 mA	2#0011	16#0203
		± 20 mA	2#0100	16#0204
HART 接口	2#0111	4 至 20 mA	2#1100	16#070C
2 线测量传感器	2#0011	4 至 20 mA	2#0011	16#0303
		± 20 mA	2#0100	16#0304
电阻 4 线连接	2#0100	48 Ω	2#0000	16#0400
		150 Ω	2#0010	16#0402
		300 Ω	2#0100	16#0404
		600 Ω	2#0110	16#0406
		1000 Ω	2#0111	16#040E
		3000 Ω	2#0111	16#0407
		6000 Ω	2#1000	16#0408
		PTC	2#1111	16#040F
电阻 3 线连接	2#0101	48 Ω	2#0000	16#0500
		150 Ω	2#0010	16#0502
		300 Ω	2#0100	16#0504
		600 Ω	2#0110	16#0506
		1000 Ω	2#0111	16#050E
		3000 Ω	2#0111	16#0507
		6000 Ω	2#1000	16#0508
		PTC	2#1111	16#050F

测量类型	代码 (A)	测量范围	代码 (B)	MODE (256*A+B)
电阻 2 线连接	2#0110	48 Ω	2#0000	16#0600
		150 Ω	2#0010	16#0602
		300 Ω	2#0100	16#0604
		600 Ω	2#0110	16#0606
		1000 Ω	2#0111	16#060E
		3000 Ω	2#0111	16#0607
		6000 Ω	2#1000	16#0608
		PTC	2#1111	16#060F
热电偶, 线性, 4 线连接	2#1000	Pt 100 气候范围	2#0000	16#0800
		Pt 200 气候范围	2#0111	16#0807
		Pt 500 气候范围	2#1000	16#0808
		Pt 1000 气候范围	2#1001	16#0809
		Ni 100 气候范围	2#0001	16#0801
		Ni 1000 气候范围	2#1010	16#080A
		Pt 100 标准范围	2#0010	16#0802
		Pt 200 标准范围	2#0011	16#0803
		Pt 500 标准范围	2#0100	16#0804
		Pt 1000 标准范围	2#0101	16#0805
		Ni 100 标准范围	2#1011	16#080B
		Ni 1000 标准范围	2#0110	16#0806
		Ni 120 标准范围	2#1100	16#080C
		Ni 120 气候范围	2#1101	16#080D
		Cu 10 气候范围	2#1110	16#080E
		Cu 10 标准范围	2#1111	16#080F
		Ni 200 标准范围	2#10000	16#0810
		Ni 200 气候范围	2#10001	16#0811
Ni 500 标准范围	2#10010	16#0812		
Ni 500 气候范围	2#10011	16#0813		

14.1 SM 模块的 MODE 设置

测量类型	代码 (A)	测量范围	代码 (B)	MODE (256*A+B)
		Pt 10 GOST 气候	2#10100	16#0814
		Pt 10 GOST 标准 (TC = 3910)	2#10101	16#0815
		Pt 50 GOST 气候	2#10110	16#0816
		Pt 50 GOST 标准 (TC = 3910)	2#10111	16#0817
		Pt 100 GOST 气候	2#11000	16#0818
		Pt 100 GOST 标准 (TC = 3910)	2#11001	16#0819
		Pt 500 GOST 气候	2#11010	16#081A
		Pt 500 GOST 标准 (TC = 3910)	2#11011	16#081B
		Cu 10 GOST 气候	2#11100	16#081C
		Cu 10 GOST 标准 (TC = 426)	2#11101	16#081D
		Cu 50 GOST 气候	2#11110	16#081E
		Cu 50 GOST 标准 (TC = 426)	2#11111	16#081F
		Cu 100 GOST 气候	2#100000	16#0820
		Cu 100 GOST 标准 (TC = 426)	2#100001	16#0821
		Ni 100 GOST 气候	2#100010	16#0822
		Ni 100 GOST 标准	2#100011	16#0823
		Pt 10 GOST 标准 (TC = 3850)	2#1010101	16#0855
		Pt 50 GOST 标准 (TC = 3850)	2#1010111	16#0857
		Pt 100 GOST 标准 (TC = 3850)	2#1011001	16#0859
		Pt 500 GOST 标准 (TC = 3850)	2#1011011	16#085B

测量类型	代码 (A)	测量范围	代码 (B)	MODE (256*A+B)
		Cu 10 GOST 标准 (TC = 428)	2#10011101	16#089D
		Cu 50 GOST 标准 (TC = 428)	2#10011111	16#089F
		Cu 100 GOST 标准 (TC = 428)	2#10100001	16#08A1
热电偶, 线性, 3 线连接	2#1001	Pt 100 气候范围	2#0000	16#0900
		Pt 200 气候范围	2#0111	16#0907
		Pt 500 气候范围	2#1000	16#0908
		Pt 1000 气候范围	2#1001	16#0909
		Ni 100 气候范围	2#0001	16#0901
		Ni 1000 气候范围	2#1010	16#090A
		Pt 100 标准范围	2#0010	16#0902
		Pt 200 标准范围	2#0011	16#0903
		Pt 500 标准范围	2#0100	16#0904
		Pt 1000 标准范围	2#0101	16#0905
		Ni 100 标准范围	2#1011	16#090B
		Ni 1000 标准范围	2#0110	16#0906
		Ni 120 标准范围	2#1100	16#090C
		Ni 120 气候范围	2#1101	16#090D
		Cu10 气候范围	2#1110	16#090E
		Cu10 标准范围	2#1111	16#090F
		Ni 200 标准范围	2#10000	16#0910
		Ni 200 气候范围	2#10001	16#0911
		Ni 500 标准范围	2#10010	16#0912
		Ni 500 气候范围	2#10011	16#0913
Pt 10 GOST 气候	2#10100	16#0914		

14.1 SM 模块的 MODE 设置

测量类型	代码 (A)	测量范围	代码 (B)	MODE (256*A+B)
		Pt 10 GOST 标准 (TC = 3910)	2#10101	16#0915
		Pt 50 GOST 气候	2#10110	16#0916
		Pt 50 GOST 标准 (TC = 3910)	2#10111	16#0917
		Pt 100 GOST 气候	2#11000	16#0918
		Pt 100 GOST 标准 (TC = 3910)	2#11001	16#0919
		Pt 500 GOST 气候	2#11010	16#091A
		Pt 500 GOST 标准 (TC = 3910)	2#11011	16#091B
		Cu 10 GOST 气候	2#11100	16#091C
		Cu 10 GOST 标准 (TC = 426)	2#11101	16#091D
		Cu 50 GOST 气候	2#11110	16#091E
		Cu 50 GOST 标准 (TC = 426)	2#11111	16#091F
		Cu 100 GOST 气候	2#100000	16#0920
		Cu 100 GOST 标准 (TC = 426)	2#100001	16#0921
		Ni 100 GOST 气候	2#100010	16#0922
		Ni 100 GOST 标准	2#100011	16#0923
		Pt 10 GOST 标准 (TC = 3850)	2#1010101	16#0955
		Pt 50 GOST 标准 (TC = 3850)	2#1010111	16#0957
		Pt 100 GOST 标准 (TC = 3850)	2#1011001	16#0959
		Pt 500 GOST 标准 (TC = 3850)	2#1011011	16#095B

测量类型	代码 (A)	测量范围	代码 (B)	MODE (256*A+B)
		Cu 10 GOST 标准 (TC = 428)	2#10011101	16#099D
		Cu 50 GOST 标准 (TC = 428)	2#10011111	16#099F
		Cu 100 GOST 标准 (TC = 428)	2#10100001	16#09A1
热电偶, 线性, 2 线连接	2#1111	Pt 100 气候范围	2#0000	16#0F00
		Pt 200 气候范围	2#0111	16#0F07
		Pt 500 气候范围	2#1000	16#0F08
		Pt 1000 气候范围	2#1001	16#0F09
		Ni 100 气候范围	2#0001	16#0F01
		Ni 1000 气候范围	2#1010	16#0F0A
		Pt 100 标准范围	2#0010	16#0F02
		Pt 200 标准范围	2#0011	16#0F03
		Pt 500 标准范围	2#0100	16#0F04
		Pt 1000 标准范围	2#0101	16#0F05
		Ni 100 标准范围	2#1011	16#0F0B
		Ni 1000 标准范围	2#0110	16#0F06
		Ni 120 标准范围	2#1100	16#0F0C
		Ni 120 气候范围	2#1101	16#0F0D
		Cu10 气候范围	2#1110	16#0F0E
		Cu10 标准范围	2#1111	16#0F0F
		Ni 200 标准范围	2#10000	16#0F10
		Ni 200 气候范围	2#10001	16#0F11
		Ni 500 标准范围	2#10010	16#0F12
Ni 500 气候范围	2#10011	16#0F13		

14.1 SM 模块的 MODE 设置

测量类型	代码 (A)	测量范围	代码 (B)	MODE (256*A+B)
热电偶, 线性, 参考温度 0 °C	2#1010	B 型 [PtRh-PtRh]	2#0000	16#0A00
		N 型 [NiCrSi-NiSi]	2#0001	16#0A01
		E 型 [NiCr-CuNi]	2#0010	16#0A02
		R 型 [PtRh-Pt]	2#0011	16#0A03
		S 型 [PtRh-Pt]	2#0100	16#0A04
		J 型 [Fe-CuNi IEC]	2#0101	16#0A05
		L 型 [Fe-CuNi DIN]	2#0110	16#0A06
		T 型 [Cu-CuNi IEC]	2#0111	16#0A07
		K 型 [NiCr-Ni]	2#1000	16#0A08
		U 型 [Cu-CuNi DIN]	2#1001	16#0A09
		C 型	2#1010	16#0A0A
		TXK/XK(L) 型	2#1011	16#0A0B
热电偶, 线性, 参考温度 50 °C	2#1011	B 型 [PtRh-PtRh]	2#0000	16#0B00
		N 型 [NiCrSi-NiSi]	2#0001	16#0B01
		E 型 [NiCr-CuNi]	2#0010	16#0B02
		R 型 [PtRh-Pt]	2#0011	16#0B03
		S 型 [PtRh-Pt]	2#0100	16#0B04
		J 型 [Fe-CuNi IEC]	2#0101	16#0B05
		L 型 [Fe-CuNi DIN]	2#0110	16#0B06
		T 型 [Cu-CuNi IEC]	2#0111	16#0B07
		K 型 [NiCr-Ni]	2#1000	16#0B08
		U 型 [Cu-CuNi DIN]	2#1001	16#0B09
		C 型	2#1010	16#0B0A
		TXK/XK(L) 型	2#1011	16#0B0B

测量类型	代码 (A)	测量范围	代码 (B)	MODE (256*A+B)
热电偶, 线性, 内部补偿	2#1101	B 型 [PtRh-PtRh]	2#0000	16#0D00
		N 型 [NiCrSi-NiSi]	2#0001	16#0D01
		E 型 [NiCr-CuNi]	2#0010	16#0D02
		R 型 [PtRh-Pt]	2#0011	16#0D03
		S 型 [PtRh-Pt]	2#0100	16#0D04
		J 型 [Fe-CuNi IEC]	2#0101	16#0D05
		L 型 [Fe-CuNi DIN]	2#0110	16#0D06
		T 型 [Cu-CuNi IEC]	2#0111	16#0D07
		K 型 [NiCr-Ni]	2#1000	16#0D08
		U 型 [Cu-CuNi DIN]	2#1001	16#0D09
		C 型	2#1010	16#0D0A
		TXK/XK(L) 型	2#1011	16#0D0B
		热电偶, 线性, 外部补偿	2#1110	B 型 [PtRh-PtRh]
N 型 [NiCrSi-NiSi]	2#0001			16#0E01
E 型 [NiCr-CuNi]	2#0010			16#0E02
R 型 [PtRh-Pt]	2#0011			16#0E03
S 型 [PtRh-Pt]	2#0100			16#0E04
J 型 [Fe-CuNi IEC]	2#0101			16#0E05
L 型 [Fe-CuNi DIN]	2#0110			16#0E06
T 型 [Cu-CuNi IEC]	2#0111			16#0E07
K 型 [NiCr-Ni]	2#1000			16#0E08
U 型 [Cu-CuNi DIN]	2#1001			16#0E09
C 型	2#1010			16#0E0A
TXK/XK(L) 型	2#1011			16#0E0B

温度系数对测量范围的影响

- 对 GOST 标准的 Pt10、Pt50、Pt100 和 Pt500 设置 TC = 3850 时，将置位测量范围字节 (0x40) 中的位 7
- 对 GOST 标准的 Cu10、Cu50 和 Cu100 设置 TC = 428 时，将置位测量范围字节 (0x80) 中的位 8

模拟输出模块的测量范围编码

根据模拟输出模块的测量范围编码，与通道对应的参数 MODE_xx（测量范围编码）必须依照下表指定。

测量类型	测量范围	MODE
电压	± 5 V	16#0106
	1 至 5 V	16#0107
	0 至 10 V	16#0108
	± 10 V	16#0109
电流	0 至 20 mA	16#0202
	4 至 20 mA	16#0203
	± 20 mA	16#0204
HART 接口	4 至 20 mA	16#070C

数字输入和输出模块的测量范围编码

对于数字输入模块和数字输出模块，无测量类型和测量范围：

MODE = 16#FFFF（对于 DI）

MODE = 16#FFFE（对于 DO）

14.2 SM 模块的 OMODE 设置

OMODE 结构

下表给出了数据类型为 DWORD 的输出 OMODE_xx 的结构和含义：

字节 3:	16#80: 值状态“有效值” 16#00: 值状态“无效值” 16#40: 值状态“无效值”	(通道错误) (较高级别错误)
字节 2:	16#01: 已执行了重启 (OB 100) 16#02: 超出测量范围上限 16#04: 超出测量范围下限	(通道错误诊断) (通道错误诊断)
字节 1、0 (低位字):	MODE (见上文)	

实例:

16#80010203 = 值状态“有效值”，已执行重启，电流 4 mA 至 20 mA。

14.3 技术数据, “标准库块”

概述

下表包含与块有关的技术数据。表的各列分别有以下含义:

- **块类型名称**

相关 FB 或 FC 在库的符号表中的符号标识符。该名称在项目中必须唯一。

- **FB/FC 号**

块编号

- **典型执行时间**

正常情况下 CPU 处理相应块程序所用的时间(例如,对于驱动程序,是指未生成通道错误消息的情况下,该驱动程序在循环中断 OB 中的执行时间)。

下表给出了各块在 417-4 CPU 中的执行时间。在其它 CPU 上的块执行时间取决于 CPU 的性能。

- **块长度**

程序代码的内存要求,各种块类型的要求不同。

- **背景数据的长度**

背景数据块的内存要求。

- **临时存储器**

调用该块时,某一优先级下所需的本地数据存储器。该限制取决于具体的 CPU。超出该限制时,请检查 CPU 组态,并在必要时将该限制分布在各个 OB 上,以满足实际需求。

- 多重背景块

指定块由相关块使用, 且必须存在于用户程序中。可在同一个库中找到这些块。

块 (类型名称)	FB/FC 号	典型运行 时间 CPU 417-4 (μ s)	装载/工作存储器 中的 块长度 (字节)	装载/工作 存储器中的 背景数据长度 (字节)	临时存储器 (字节)	多 重背景 块
ADD4_P	FC256	6	194/122	-/-	2	
ADD8_P	FC257	8	298/202	-/-	2	
AL_DELAY	FC 290					
ASSETMON	FB 86	84	6474/5486	1102/546	90	SFB 31, 2 x SFB 35
AVER_P	FB 34	16	508/368	156/54	48	
CH_AI	FC 275	36	6254/5414	-/-	28	
CH_AO	FC 276	30	1782/1510	-/-	34	
CH_CNT	FB 127	33	6466/5928	452/124	70	
CH_CNT1	FB 59	16	1342/1152	214/92	36	
CH_DI	FC 277	14	502/392	-/-	4	
CH_DO	FC 278	11	358/274	-/-	2	
CH_MS	FB 60	22	1680/1366	310/74	20	
CH_U_AI	FC 283	46	8826/7688	-/-	28	
CH_U_AO	FC 284	31	2674/2302	-/-	34	
CH_U_DI	FC 285	20	3064/2666	-/-	6	
CH_U_DO	FC 286	13	1268/1080	-/-	2	
COUNT_P	FB 36	16	484/340	166/54	44	
CPM	FB 140					
CPU_RT	FB 128	67	31434/27370	2800/1784	86	
CTRL_PID	FB 61	178	7922/6502	1322/604	164	2 x FB 46 + SFB 35
CTRL_S	FB 76	189	10386/8672	1496/644	198	2 x FB 46 + SFB 35
DEADT_P	FB 37	20	908/704	252/126	48	

块 (类型名称)	FB/FC 号	典型运行 时间 CPU 417-4 (μ s)	装载/工作存储器 中的 块长度 (字节)	装载/工作 存储器中的 背景数据长度 (字节)	临时存储器 (字节)	多 重背景 块
DIF_P	FB 38	21	710/518	208/72	56	
DIG_MON	FB 62	104	2078/1692	802/482	56	SFB 35 + SFB 36
DOSE	FB 63	127	5250/4312	1178/592	76	FB 46 + SFB 35 + SFB 36
ELAP_CNT	FB64	101	1434/1086	610/334	54	SFB 35
FF_A_AI	FB 410	38	6882/5618	1470/516	50	
FF_A_AO	FB 411	56	1174/814	492/154	14	
FF_A_DI	FB 412	32	1976/1438	740/252	20	
FF_A_DO	FB 413	49	926/634	364/90	8	
FMCS_PID	FB 114	143	9712/8112	1870/846	116	FB 46 + SFB 35
FMT_PID	FB 77	142	9422/8002	1856/894	82	FB 46 + SFB 35
GAIN_SHD	FB 141					
INT_P	FB 40	24	1142/892	228/84	60	
INTERLOK	FB 75	27	1492/1154	300/78	46	
LIMITS_P	FB 41	6	308/216	124/58	6	
MEANTM_P	FB 42	53	1586/1306	264/154	20	
MEAS_MON	FB 65	108	1910/1486	690/376	56	SFB 35
MESSAGE	FB 43	98	932/684	506/278	44	SFB 35
MODB_341	FB 80	594	4388/3666	1012/490	54	2 x SFB 35
MOT_REV	FB 67	125	4466/3770	828/382	70	SFB 35
MOT_SPED	FB 68	122	4304/3618	824/382	66	SFB 35
MOTOR	FB 66	114	2570/2108	714/364	54	SFB 35
MPC	FB 142					

块 (类型名称)	FB/FC 号	典型运行 时间 CPU 417-4 (μ s)	装载/工作存储器 中的 块长度 (字节)	装载/工作 存储器中的 背景数据长度 (字节)	临时存储器 (字节)	多 重背景 块
MS_MUX	FC 288	9	382/288	-	6	SFB 54 + SFB 35
MSG_NACK	FB 78	99	998/732	520/274	44	SFB 31
MSG_TS	FB 131					
MUL4_P	FC262	5	214/122	-/-	2	
MUL8_P	FC263	9	334/202	-/-	2	
NOISE_GN	FB 143					
OB1_TIME	FB 69	57	2140/1822	216/70	84	
OP_A	FB 45	4	232/156	114/56	2	
OP_A_LIM	FB 46	8	486/358	160/68	6	
OP_A_RJC	FB 47	8	518/388	160/68	6	
OP_D	FB 48	6	376/286	112/44	2	
OP_D3	FB 49	12	1354/1136	150/46	8	
OP_TRIG	FB 50	5	244/166	104/44	2	
PA_AI	FB 101	21	3652/3162	290/96	6	
PA_AO	FB 103	45	4054/3524	406/140	10	
PA_DI	FB 104	19	3438/2992	224/62	6	
PA_DO	FB 105	28	2960/2572	314/82	12	
PA_TOT	FB 102	21	3774/3264	306/100	12	
POLYG_P	FC271	6	1446/1176	-/-	24	
PT1_P	FB 51	7	446/330	136/66	2	
R_TO_DW	FC282	6	344/262	-/-	10	
RAMP_P	FB 52	19	708/524	188/64	54	
RATIO_P	FB 70	19	742/526	316/134	12	FB 46 + SFB 35
RCV_341	FB 121	9	1940/1070	1694/874	12	SFB 35
READ355P	FB 72	18	984/784	258/94	94	
REC_BO	FB 208	69	3246/2356	992/128	2	SFB 13

14.3 技术数据, “标准库块”

块 (类型名称)	FB/FC 号	典型运行 时间 CPU 417-4 (μ s)	装载/工作存储器 中的 块长度 (字节)	装载/工作 存储器中的 背景数据长度 (字节)	临时存储器 (字节)	多 重背景 块
REC_R	FB 210	69	1838/1332	956/476	2	SFB 13
SEND_BO	FB 207	163	2298/1668	718/110	2	SFB 12
SEND_R	FB 209	195	4486/3886	908/478	2	SFB 12
SIG_SMTH	FB 144					
SND_341	FB 120	86	1608/1226	616/288	12	SFB 35
SPLITR_P	FC272	17	832/644	-/-	10	
ST_MUX	FC 287	114	1612/1408	-	20	
STATEREP	FB 87	15	1900/1574	182/58	4	
SWIT_CNT	FB 71	102	1332/988	606/316	92	SFB 35
VAL_MOT	FB 74	127	4498/3782	828/382	70	SFB 35
VALVE	FB 73	120	3280/2740	734/366	54	SFB 35

14.4 为 FF 设备自定义 MODE_LW

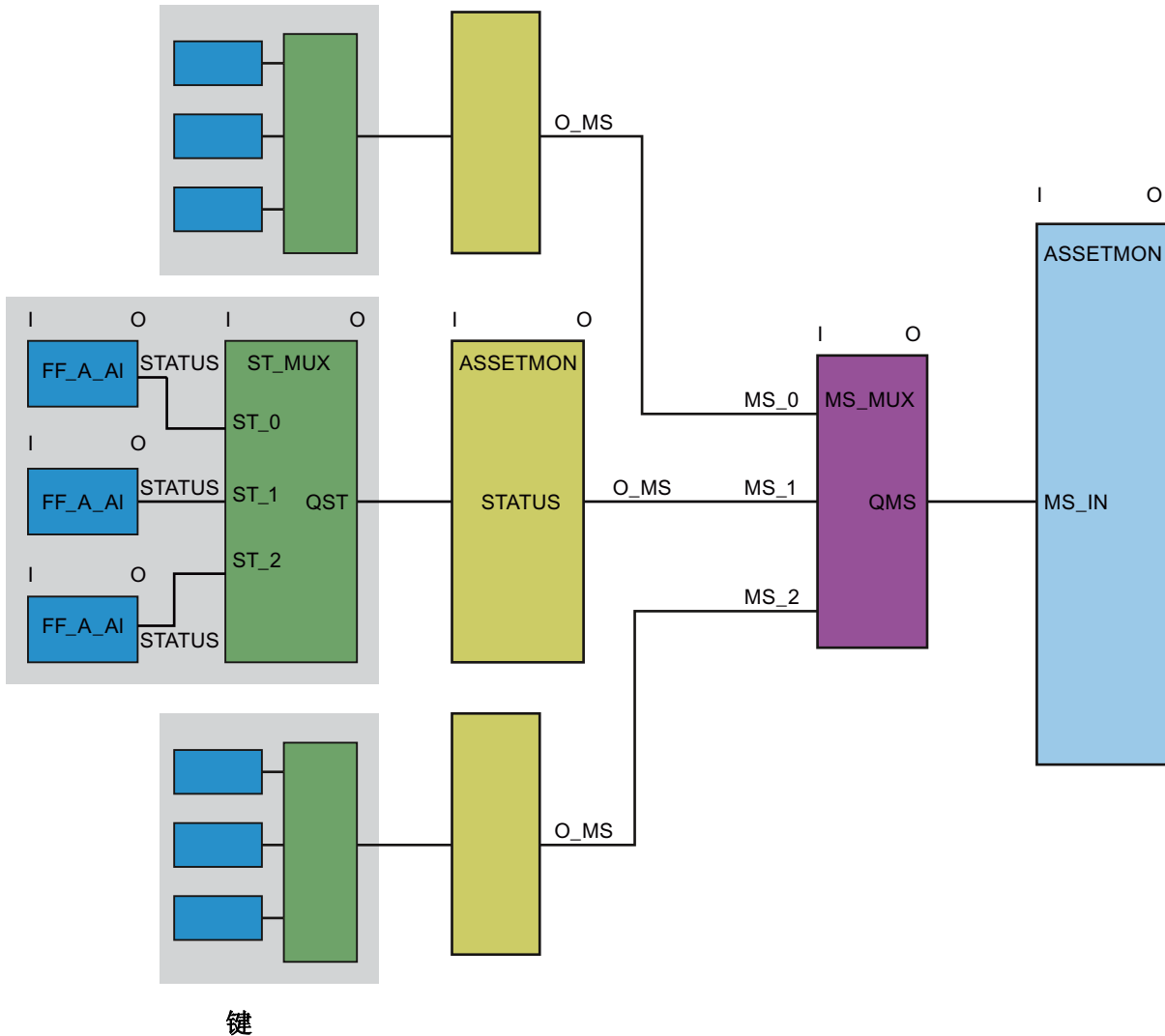
块	I/O (参数) (循环数据) 允许的组合 和序列	输入 (I)/输出 (O) (PLS 视图)	模式 16#xxyy, O=xx I=yy
模拟输入 (FF_A_AI)	OUT	I	16#0001
离散输入 (FF_A_DI)	OUT_D	I	16#0002
模拟输出 (FF_A_AO)	SP	O	16#0100
模拟输出 (FF_A_AO)	SP READBACK POS_D	O I I	16#0103
模拟输出 (FF_A_AO)	SP CHECK_BACK	O I	16#0104
模拟输出 (FF_A_AO)	SP READBACK POS_D CHECK_BACK	O I I I	16#0105
模拟输出 (FF_A_AO)	RCAS_IN RCAS_OUT	O I	16#0206
模拟输出 (FF_A_AO)	RCAS_IN RCAS_OUT CHECK_BACK	O I I	16#0207
模拟输出 (FF_A_AO)	SP RCAS_IN READBACK RCAS_OUT POS_D CHECK_BACK	O O I I I I	16#0308
离散输出 (FF_A_DO)	SP_D	I	16#0400
离散输出 (FF_A_DO)	SP_D READBACK_D	O I	16#0409
离散输出 (FF_A_DO)	SP_D CHECK_BACK_D	O I	16#040A






14.4 为 FF 设备自定义 MODE_LW

块	I/O (参数) (循环数据) 允许的组 和序列	输入 (I)/输出 (O) (PLS 视图)	模式 16#xyy, O=xx I=yy
离散输出 (FF_A_DO)	SP_D READBACK_D CHECK_BACK_D	O I I	16#040B
离散输出 (FF_A_DO)	RCAS_IN_D RCAS_OUT_D	O I	16#050C
离散输出 (FF_A_DO)	RCAS_IN_D RCAS_OUT_D CHECK_BACK_D	O I I	16#0D05
离散输出 (FF_A_DO)	SP_D RCAS_IN_D READBACK_D RCAS_OUT_D CHECK_BACK_D	O O I I I	16#0E06

14.5 在资产中集成 FF 设备

可按如下所示在资产中集成 FF 设备：



	FF 现场设备的信号处理驱动程序块 [FF_A_xx]，其中 xx = AI、DI、AO 或 DO
	针对一台现场设备汇总 [ST_MUX]
	代表一台现场设备 [ASSETMON]
	针对一个功能单元汇总 [MS_MUX]
	代表一个功能单元 [ASSETMON]

14.6 归档过程值

系统属性“s7_archive”

系统属性“s7_archive”通知 WinCC 在运行期间应该归档哪些过程值。该属性还可用于区分长期归档 (s7_archive=longterm)、短期归档 (s7_archive=shortterm) 和不归档 (s7_archive=false)。

当激活由 ConPerMon 块执行控制器监视功能时，归档数据将通过 ConPerMon 面板进行评估。

对于下列控制器而言，默认的参数设置是短期归档：

CTRL_PID	CTRL_S	FMCS_PID	FMT_PID
LMN	LMNR_IN	LMN	LMN
PV_IN	PV_IN	PV	PV
QLMN_HLM	QLMNR_HS	QLMN_HLM	QLMN_HLM
QLMN_LLM	QLMNR_LS	QLMN_LLM	QLMN_LLM
QMAN_AUT	QMAN_AUT	QMAN_AUT	QMAN_AUT
SP	SP	SP	SP

对于 FMCS_PID 和 FMT_PID 块来说，该默认归档设置是用于连续控制器。当它们用作步控制器时，必须为短期归档设置参数 LMN_A、QLMNR_HS 和 QLMNR_LS。

为节省 WinCC 中的资源，可使用“s7_archive=false”禁用不必要的归档。

14.7 文本库 ASSETMON

下表列出了块 ASSETMON (FB 86) 的文本库消息文本及其编号：







文本号	消息文本
1	更多状态可用 (Further status available)
2	无更多状态可用 (No further status available)

14.8 质量代码和状态显示

14.8.1 质量代码显示

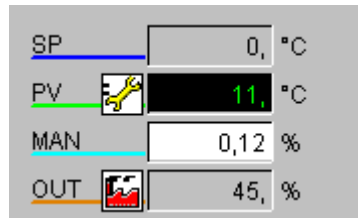
质量代码显示的属性

质量代码显示是一种扩展的状态显示，有七种备选形式。

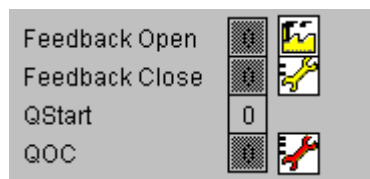
优先级	质量代码	纯文本	符号
1	0x44 0x48 0x60	模拟	
2	0x00 0x14 0x18	不良，设备相关	
3	0x28 *) 0x2B	不良，过程相关	
4	0x68	不确定，设备相关	
5	0x54 0x55 0x56 0x78	不确定，过程相关	
6	0xA4	需要维护	
7	0x80	良好	
*) 如果与 AS 的通讯是下行的，则在 OS 中生成代码 0x08。			

将鼠标指针置于质量代码显示上时，将显示纯文本短信息。

质量代码在**模拟值**之前显示。



对于**二进制值**，质量代码显示在维护视图中，在阀门及电机面板中的二进制值显示之后出现。



更多信息

可在 LIB 自述文件中找到详尽的 Excel 表格“QC_MS”，包含 PA 设备的质量代码，以及在资产管理中的应用。

14.8.2 MS 维护状态

维护状态的布局

维护状态 MS（DWORD 数据类型）的布局如下所示：

位 0 到 7	显示 MS
位 8 到 15	显示冗余伙伴的 MS
位 16	1 = 冗余伙伴可用
位 17	0 = 主伙伴是主站， 1 = 冗余伙伴是主站
位 18	PDM-MS 比设备状态更糟
位 19 到 21	预留
位 22	PDM 已检测到状态更改
位 23	块参与循环更新 PDM
位 24 到 27	OS 操作
位 28 到 31	PDM-MS

按 1:1 将 MS 复制到 O_MS 输出中。

可显示的状态

维护状态 (MS) 可显示下列状态，这些状态在位 0 至 7 或位 8 至 15（针对冗余伙伴）中输入：





位号								状态	符号	优先级
7	6	5	4	3	2	1	0			0 = 最高优先级
0	0	0	0	0	0	0	0	良好		9
0	0	0	0	0	0	0	1	已钝化		7
0	0	0	0	0	0	1	0	服务中断		6
0	0	0	0	0	0	1	1	至少模拟一个 PV		5
0	0	0	0	0	1	0	0	本地操作/功能测试		4
0	0	0	0	0	1	0	1	需要维护		3
0	0	0	0	0	1	1	0	维护请求		2
0	0	0	0	0	1	1	1	维护报警		1
0	0	0	0	1	0	0	0	未测试/未知		0
0	0	0	0	1	0	0	1	组态已更改		8

说明

如果维护状态为“未测试/未知”，则资产管理面板中的所有其它动态显示均与本实例无关。

消息的维护状态

该维护状态在驱动程序块中通过消息进行更新。驱动程序块会生成以下消息类别的消息：

消息类别	事件状态位	符号
AS 控制系统消息 (S) = 故障	25	
AS 控制系统消息 (F) = 错误	24	
预防性维护 (M) = 维护	23	
状态 AS (SA)	18	

冗余

冗余的情况下，可能有多种显示组合。请参阅：

冗余组件 [资产] 的状态显示 (页 735)

14.8.3 冗余组件 [资产] 的状态显示










状态显示图标

冗余组件 A 和冗余组件 B 构成了冗余组件的状态显示（维护状态）。下表中列出了根据此规则得出的状态显示的图标（表格中未列出的位号始终为 0）。

说明


状态 MS = 9 是针对冗余组件的“组态已更改”，因此不在其中列出。

PV = 过程值
















位号								状态		
11	10	9	8	3	2	1	0	冗余组件 A	冗余组件 B	状态显示图标
0	0	0	0	0	0	0	0	良好	良好	良好 
0	0	0	1	0	0	0	0	良好	已钝化	良好 
0	0	1	0	0	0	0	0	良好	服务中断	维护请求 
0	0	1	1	0	0	0	0	良好	至少模拟 1 个 PV	良好 
0	1	0	0	0	0	0	0	良好	本地操作/功能测试	良好 
0	1	0	1	0	0	0	0	良好	需要维护	需要维护 
0	1	1	0	0	0	0	0	良好	维护请求	维护请求 
0	1	1	1	0	0	0	0	良好	维护报警	维护请求 
1	0	0	0	0	0	0	0	良好	未测试/未知	良好 

















14.8 质量代码和状态显示

位号								状态			
0	0	0	0	0	0	0	1	已钝化	良好	良好	
0	0	0	1	0	0	0	1	已钝化	已钝化	已钝化	
0	0	1	0	0	0	0	1	已钝化	服务中断	服务中断	
0	0	1	1	0	0	0	1	已钝化	至少模拟 1 个 PV	至少模拟 1 个 PV	
0	1	0	0	0	0	0	1	已钝化	本地操作/ 功能测试	本地操作/ 功能测试	
0	1	0	1	0	0	0	1	已钝化	需要维护	需要维护	
0	1	1	0	0	0	0	1	已钝化	维护 请求	维护 请求	
0	1	1	1	0	0	0	1	已钝化	维护报警	维护报警	
1	0	0	0	0	0	0	1	已钝化	未测试/ 未知	已钝化	
0	0	0	0	0	0	1	0	服务中断	良好	维护 请求	
0	0	0	1	0	0	1	0	服务中断	已钝化	服务中断	
0	0	1	0	0	0	1	0	服务中断	服务中断	服务中断	
0	0	1	1	0	0	1	0	服务中断	至少模拟 1 个 PV	至少模拟 1 个 PV	
0	1	0	0	0	0	1	0	服务中断	本地操作/ 功能测试	本地操作/ 功能测试	
0	1	0	1	0	0	1	0	服务中断	需要维护	需要维护	
0	1	1	0	0	0	1	0	服务中断	维护 请求	维护 请求	
0	1	1	1	0	0	1	0	服务中断	维护报警	维护报警	




位号								状态			
1	0	0	0	0	0	1	0	服务中断	未测试/ 未知	服务中断	
0	0	0	0	0	0	1	1	至少模拟 1 个 PV	良好	良好	
0	0	0	1	0	0	1	1	至少模拟 1 个 PV	已钝化	至少模拟 1 个 PV	
0	0	1	0	0	0	1	1	至少模拟 1 个 PV	服务中断	至少模拟 1 个 PV	
0	0	1	1	0	0	1	1	至少模拟 1 个 PV	至少模拟 1 个 PV	至少模拟 1 个 PV	
0	1	0	0	0	0	1	1	至少模拟 1 个 PV	本地操作/ 功能测试	本地操作/ 功能测试	
0	1	0	1	0	0	1	1	至少模拟 1 个 PV	需要维护	需要维护	
0	1	1	0	0	0	1	1	至少模拟 1 个 PV	维护 请求	维护 请求	
0	1	1	1	0	0	1	1	至少模拟 1 个 PV	维护报警	维护报警	
1	0	0	0	0	0	1	1	至少模拟 1 个 PV	未测试/ 未知	至少模拟 1 个 PV	
0	0	0	0	0	1	0	0	本地操作/ 功能测试	良好	良好	
0	0	0	1	0	1	0	0	本地操作/ 功能测试	已钝化	本地操作/ 功能测试	
0	0	1	0	0	1	0	0	本地操作/ 功能测试	服务中断	本地操作/ 功能测试	
0	0	1	1	0	1	0	0	本地操作/ 功能测试	至少模拟 1 个 PV	本地操作/ 功能测试	
0	1	0	0	0	1	0	0	本地操作/ 功能测试	本地操作/ 功能测试	本地操作/ 功能测试	

14.8 质量代码和状态显示

位号								状态			
0	1	0	1	0	1	0	0	本地操作/ 功能测试	需要维护	需要维护	
0	1	1	0	0	1	0	0	本地操作/ 功能测试	维护 请求	维护 请求	
0	1	1	1	0	1	0	0	本地操作/ 功能测试	维护报警	维护报警	
1	0	0	0	0	1	0	0	本地操作/ 功能测试	未测试/ 未知	本地操作/ 功能测试	
0	0	0	0	0	1	0	1	需要维护	良好	需要维护	
0	0	0	1	0	1	0	1	需要维护	已钝化	需要维护	
0	0	1	0	0	1	0	1	需要维护	服务中断	需要维护	
0	0	1	1	0	1	0	1	需要维护	至少模拟 1个PV	需要维护	
0	1	0	0	0	1	0	1	需要维护	本地操作/ 功能测试	需要维护	
0	1	0	1	0	1	0	1	需要维护	需要维护	需要维护	
0	1	1	0	0	1	0	1	需要维护	维护 请求	维护 请求	
0	1	1	1	0	1	0	1	需要维护	维护报警	维护报警	
1	0	0	0	0	1	0	1	需要维护	未测试/ 未知	需要维护	
0	0	0	0	0	1	1	0	维护 请求	良好	维护 请求	
0	0	0	1	0	1	1	0	维护 请求	已钝化	维护 请求	

位号								状态			
0	0	1	0	0	1	1	0	维护请求	服务中断	维护请求	
0	0	1	1	0	1	1	0	维护请求	至少模拟 1个PV	维护请求	
0	1	0	0	0	1	1	0	维护请求	本地操作/ 功能测试	维护请求	
0	1	0	1	0	1	1	0	维护请求	需要维护	维护请求	
0	1	1	0	0	1	1	0	维护请求	维护请求	维护请求	
0	1	1	1	0	1	1	0	维护请求	维护请求	维护报警	
1	0	0	0	0	1	1	0	维护请求	未测试/ 未知	维护请求	
0	0	0	0	0	1	1	1	维护报警	良好	维护请求	
0	0	0	1	0	1	1	1	维护报警	已钝化	维护报警	
0	0	1	0	0	1	1	1	维护报警	服务中断	维护报警	
0	0	1	1	0	1	1	1	维护报警	至少模拟 1个PV	维护报警	
0	1	0	0	0	1	1	1	维护报警	本地操作/ 功能测试	维护报警	
0	1	0	1	0	1	1	1	维护报警	需要维护	维护报警	
0	1	1	0	0	1	1	1	维护报警	维护请求	维护报警	
0	1	1	1	0	1	1	1	维护报警	维护报警	维护报警	
1	0	0	0	0	1	1	1	维护报警	未测试/ 未知	维护报警	

14.8 质量代码和状态显示

位号								状态			
1	0	0	0	1	0	0	0	未测试/ 未知	未测试/ 未知	未测试/ 未知	
0	0	0	0	1	0	0	0	未测试/ 未知	良好	良好	
0	0	0	1	1	0	0	0	未测试/ 未知	已钝化	已钝化	
0	0	1	0	1	0	0	0	未测试/ 未知	服务中断	服务中断	
0	0	1	1	1	0	0	0	未测试/ 未知	至少模拟 1个PV	至少模拟 1个PV	
0	1	0	0	1	0	0	0	未测试/ 未知	本地操作/ 功能测试	本地操作/ 功能测试	
0	1	0	1	1	0	0	0	未测试/ 未知	需要维护	需要维护	
0	1	1	0	1	0	0	0	未测试/ 未知	维护 请求	维护 请求	
0	1	1	1	1	0	0	0	未测试/ 未知	维护报警	维护报警	

PCS 7 高级过程控制模板

15.1 过程标签类型（可插入模板）

15.1.1 过程标签类型说明

使用过程标签类型

在 PCS 7 多项目中使用过程标签类型时，建议应用下列步骤：

- 将所有需要的功能块复制到多项目的主数据库的块文件夹中（<项目名>_Lib）。
- 然后，在工厂视图中，将需要的过程标签类型复制到多项目的主数据库的“processtagtypes”文件夹中。
- 自定义过程标签类型。

生成过程标签

过程标签是过程标签类型的实例。

有两种方法可生成过程标签：

1. 从主数据库中复制过程标签类型并将其插入到工厂层级的目标文件夹。然后可以为其设置参数并在 CFC 编辑器中互连 CFC 图表。
2. 使用导入-导出助手，可以生成所需的过程标签类型的过程标签，然后根据导入文件为其分配参数并进行互连。

15.1.2 具有安全逻辑和控制回路监视的 PID 控制器

模板“具有安全逻辑和控制回路监视的 PID 控制器”的描述

此过程标签类型形成了为连续过程生成 PID 控制实例的基础。除了实际 PID 块外，它还包含应在每个控制回路中始终执行的功能。但是，此功能不在 PID 块本身内执行，以便用户可以根据特定项目调整以下功能：

- 过程值 PV_IN 的模拟输入驱动程序、位置反馈 LMNR_IN（如果可用），以及调节变量 LMN 的模拟输出驱动程序。
- 简单的一阶过程仿真，由调节变量 LMN 控制，它为模拟输入 PV_IN 提供仿真值。此功能允许在连接实际过程之前，至少由基本功能测试控制回路。
- 过程值测量失败时将控制回路更改为安全模式的某些逻辑块。PV_IN 的质量代码发出此状态的通知信号。
- 用于控制回路监视的附加功能块，应在所有 PID 控制回路中安装该块。

有关“测量值失败”的说明

如果 PV_IN 没有有效的测量值，则控制器不能在自动模式下继续运行，因为闭环控制回路已不再存在。如果没有实际过程状态的反馈，控制器无法计算有用的调节变量。这时必须将控制器更改为手动模式或跟踪模式。会出现一条相应的消息，提示操作员必须注意这种情况。根据应用背景的不同，针对测量值丢失可能出现不同反应：

1. 跟踪一个固定、已组态的安全值，例如阀关闭或加热器关闭。
2. 保持上一个有效调节值 LMN 常量，以尽可能长的保持稳定的过程状态（如果过程已处于此状态）。
3. 更改为手动模式，以便操作员可以取代过程控制的职责。

在模板中实现反应 2 和 3 的组合。控制器更改为手动跟踪模式时，上一个有效调节值作为起始值。在 PID 块的 AUT_L 输入处进行更改并将 LIOP_MAN_SEL 参数暂时设置为 1。在名为 PV_BadSafetyLogic 的图表中作为 CFC 图表实现“测量值失败”功能。

注意：如果信号容易受强噪声干扰，或在设定值频繁更改的控制回路中，不建议冻结上一个有效调节变量。在这种情况下，建议更改为变体 1。

如果控制器通过执行器（例如，带有电气动位置控制器 Sipart PS 的阀）本地干预过程，则必须采用类似于针对级联控制器的特殊措施。

- 如果有本地干涉，则控制器跟踪当前的执行器位置。在这种情况下，反馈信号应用于输入 LMN_TRK，并且在输入 LMN_SEL 前设置 OR 元素。
- 当测量值丢失时，跟踪模式被激活。此变体未在模板中实现。

有关控制回路监视的信息，可参考 CPM 块的在线帮助。

15.1.3 可直接访问执行器而无位置反馈的步控制器

模板“无位置反馈的步控制器”的描述

CTRL_S 块的输出通过两个数字输出驱动程序直接连接到过程。这是步控制的最简单形式。操作员可在控制器面板中控制执行器（点动模式打开/关闭）。当运行无位置反馈的简单执行器时，此点动模式更可取，因为它允许平稳地切换到自动模式。如果从控制块之外的任何位置手动运行无位置反馈的执行器，则不能进行此切换。因此，采用无位置反馈的 CTRL_S 块设计此模板。

15.1.4 分程控制

模板“Split-Range control”的描述

PID 闭环控制器允许在该控制器输出的下游使用分程块向多个执行器分发其调节变量，这些执行器基于不同物理原理并以不同的方向影响同一控制变量。这一应用的典型示例是温度控制，通过新汽阀加热和通过冷却水阀冷却。控制器可根据误差信号的标志（或者更精确的调节变量）请求加热或冷却能量。换句话说，它可以使用双向 LMN 输出（例如： $-100\% < LMN < 100\%$ ），虽然每个执行器仅支持单极性运行（例如： $0\% < \text{阀位置} < 100\%$ ）。

Split-Range 功能块包含两个执行器的两个单独的（静态）特征。两个执行器在性能方面的任何明显差异（可理解为加热和冷却的不同过程增益），均可通过设置特征的不同梯度来进行补偿，以便控制器尽可能获得线性过程响应（与标志无关）。

实例：冷却效果没有加热效果明显。

如果冷却的有效性是加热的一半，则冷却的分程特征的梯度应是加热的二倍。

注：调节变量的最大有效值通过控制器的调节变量限制乘以分程特征的梯度获得。

冷却阀不能超过完全打开，即打开阀门的有效限制是 100%。如果使用梯度为 2 的分程特征进行冷却，则必须在控制器上将调节变量下限设置为 $LMN_LLM = -50\%$ 。

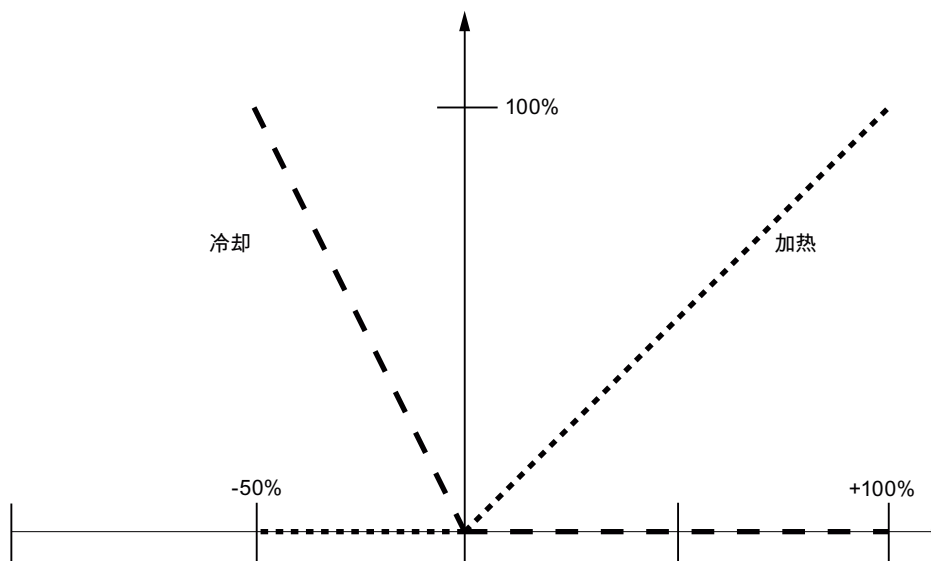


图 15-1 实例的分程功能

实际分程功能作为单独的 `SPLITR_P` 功能块来提供，有关此块的详细信息，请参阅在线帮助。

应用实例：

- 通过蒸汽加热阀和水冷却阀控制化学反应釜的温度。
- 通过气体燃烧器和冷却风扇控制玻璃熔炉或注料道的温度
- 通过电加热和冷却风扇控制挤压机的温度
- 通过进气阀和出气阀控制气相反应釜中的压力
- 通过多个蒸汽发生器的进气阀或者多个蒸汽发生器的加热功率，控制向多个装置提供蒸汽的蒸汽收集轨道的压力。

15.1.5 比率控制

模板“比率控制”的描述

通过由多个流量控制器和一个 **RATIO_P** 块组成的比率控制系统，可产生多种液体或气体的特殊混合物。添加剂的流量设定值可从下列值之一中获得：

1. 从主流量的当前 **PV_IN** 值获得。
如果主流量控制器在具有稳态偏差的情况下运行，这是首选方案。
或者
2. 从主流量的设定值 **SP** 获得。
此方案为次级控制器提供平滑、无噪声的设定值信号，并可以对过渡状态下（在该状态下两个流量控制回路几乎具有相同的动态响应）的指定比率进行更精确控制。

通常建议对具有 **I** 动作的主流量控制器使用第二个“面向设定值”的备选方案。

可通过添加更多组件扩展比率控制，即利用附加 **Ratio** 块可从 **SP 1** (或 **PV_IN 1**) 获取设定值 **3** 到 **n**。

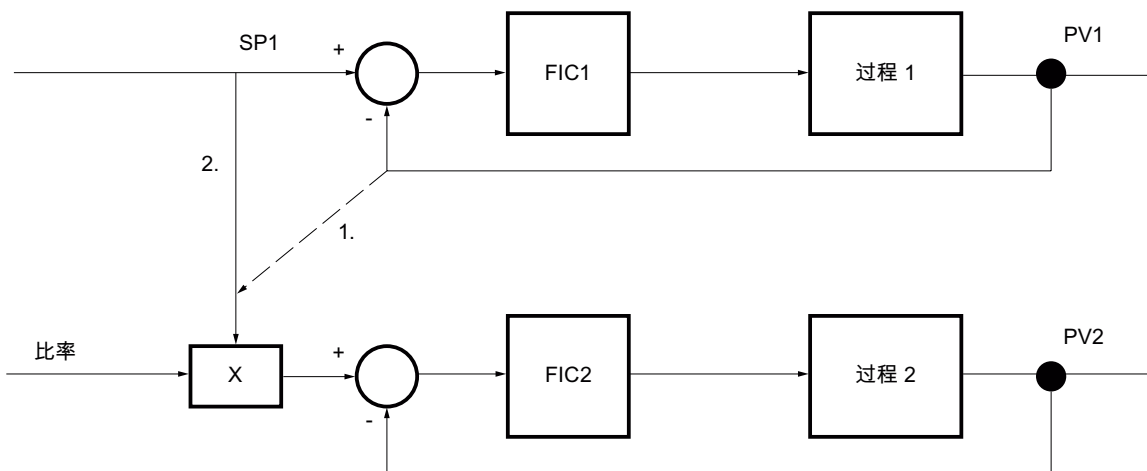


图 15-2 比率控制，面向过程值 (1.) 和面向设定值 (2.)

在 1 和 2 两种情况下，在闭环中不直接测量或控制比率的当前值（无 feedback-control）而是使用 feedforward-control。

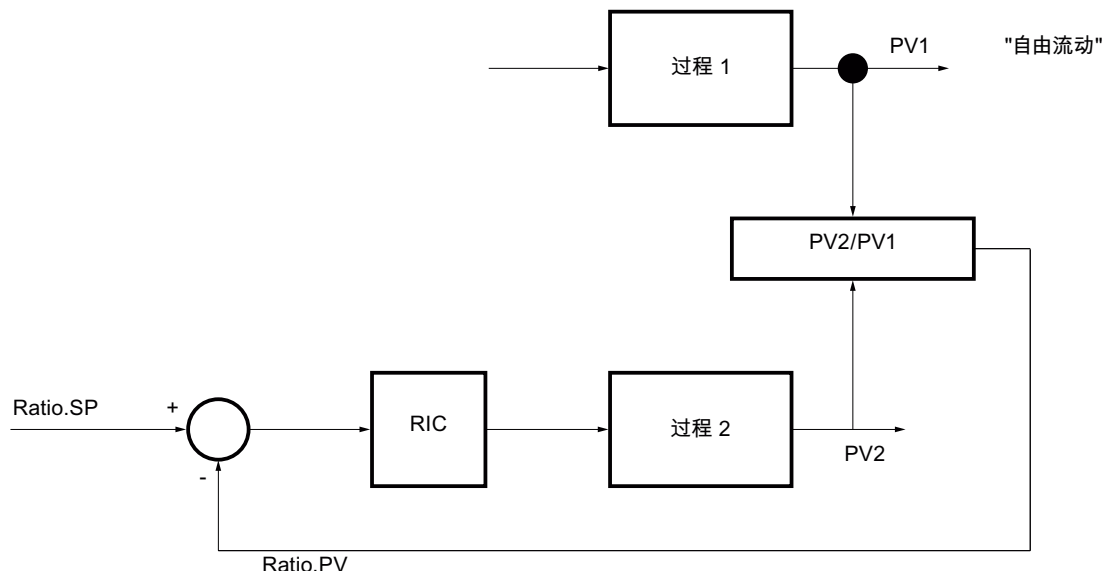


图 15-3 当前比率的反馈控制

但是，如果仅可以调节两个流量中的一个，则有时使用比率的真实反馈。动态流量 PV_IN 1 和调节流量 PV_IN 2 之间的当前比率通过除法来定义。调整第二个流速的控制器接收所需比率作为它的设定值。模板中不包含该第三种方案，但是，利用标准 PID 控制器可轻松实现该方案。但必须采取相应步骤以避免计算比率时被零除。

注：还可以使用方案 1 解决此类应用，因为它一开始就避免了除权限。

应用实例

- 进给到反应釜的两种反应物的流量比率控制
- 要混合的两种液体流量的比率控制，即混合物成分的控制
- 反应釜卸料和反应釜循环利用蒸汽之间的比率控制
- 燃烧器气体和用于优化燃烧性能的新鲜空气的比率控制
- 供应给吸收塔的液体和蒸汽的比率控制

15.1.6 级联控制

关于级联控制的常规信息

级联控制涉及两个或多个串连的 PID 控制器。将主控制器的调节变量连接到次级控制器的外部设定值，这样就将两个控制电路嵌套。级联控制的优点是，在次级回路中影响内部回路的干扰的补偿速度比在稍慢的主回路中快很多。在某些情况下，执行器的非线性影响可在次级回路中补偿，这样可为主回路创建线性过程响应。仅当过程中除了主控制变量外还存在其它可测量变量并且内部控制回路比外部回路快的多时，才可能进行级联控制。

级联控制需澄清的方面

对于任何级联控制，必须仔细考虑并澄清以下方面：

- 主控制器的作用范围必须与次级控制器的设定值范围相符，以确保主控制器的抗饱和和功能正常运行。
- 如果次级控制器不在“级联”模式下运行，而是在其它模式下运行（例如，手动模式或具有本地设定值的自动模式），进而使得该控制器不对主控制器的命令作出响应，这时必须将主控制器切换到“跟踪”模式以防止主控制器中 I 动作的积分。主控制器的调节值跟踪次级控制器的过程值，以便可以平稳地返回到级联模式。当次级控制器进入手动模式时，跟踪设定值与跟踪过程值之间的区别变得很明显。如果跟踪过程值，则响应类似于简单控制器的“在手动模式下跟踪设定值至过程值”。
- 如果次级控制器达到了调节变量的限制（上限/下限），则应阻止主控制器的积分器，以防止其在此方向上（向上/向下）进一步积分。无论如何次级控制器不能在此方向上进一步积分。当实际执行器已达到其物理限制而主控制器尚未达到其调节变量限制时，这会防止主控制器的积分饱和。

注意：

- 如果次级控制器有负增益，交换该互连的两个位。
- 如果这时次级控制器达到了上限或下限，则必须防止主控制器的积分器在此方向上进一步积分。

步骤

设置和调试控制器时，您从“内向外”工作，即首先对次级控制器进行设置并使其进入自动模式。然后设置主控制器参数并将次级控制器更改为级联模式。当为主控制器设置参数时，请记住从整体角度来看，内部闭合控制回路是“受控系统”。为主控制器设置的参数取决于为次级控制器进行的设置范围的大小。主控制回路和次级控制回路之间的动态响应的差异越大，此方面的重要性就变得越小。

优先级

通过次级控制器启动的主控制器的跟踪比主控制器上的手动模式具有更高的优先级。

15.2 实例项目

15.2.1 包括噪声发生器的过程仿真

模板“包括噪声发生器的过程仿真”的描述

用户仅需要几个标准模块即可创建一个动态过程模型，该模型反映具有适当精度的多个工艺过程的响应模式。此模型将在所有实例项目 (APC_ExaSP) 中使用。但是，此模型也可用于销售演示或者测试闭环控制功能，即用在真实工厂尚不可用的项目阶段（“虚拟过程”，“影子工厂”）。过程仿真作为开源 CFC 图表 "ProcSimC" 提供，用户可将其作为嵌套图表（图中图）安装在其它 CFC 图表中。该块提供三个一阶延迟元件、一个增益系数、一个等效值 PV（对于 MV=0），和一个用于白测量噪声的噪声发生器。为输入的（仿真）干扰提供附加输入。下面介绍的 Laplace 传递函数就是通过此模块实现的。

$$PV(s) = \frac{Gain}{(TimeLag1 \cdot s + 1)(TimeLag2 \cdot s + 1)(TimeLag3 \cdot s + 1)} (MV(s) + DisV(s)) + PV_0 + Noise$$

用例

用户可以调整此灵活的模型以满足各种使用情况的需要，例如：

- 温度控制系统仿真：** PV0 表示无需加热的温度（例如，环境温度）。TimeLag1 的值通常大大高于 TimeLag2 和 TimeLag3。后者的值还可以为零。传感器生成 0.1°C 的典型量化噪声。正值增益可解释为满负荷加热功率时可达到的理论最高温度。但在大多数情况下，这无法通过实验测量，因为许多执行器已标有尺寸，以便仅需要大约三分之一的加热功率即可在运行点处持续运行。余下的功率仅用于运行点更改和加热阶段。
- 压力控制系统仿真：** 如果定义阀门位置，使其在 0% 时关闭在 100% 时打开，容器压力控制系统的过程 Gain 通常为负值，因为当容器的出气阀打开时压力减小 (>0)。相反，过压进气阀的增益通常为正值。PV0 > 0 是阀门完全关闭时的压力。当然，涉及的压力低于外界压力时（例如，在真空系统中），情况正好相反。请注意，大部分阀门在其关闭位置区域不返回可再生特征（作用比率 1:20 或 1:50）。液体压力控制的时间常数通常很快，而对于气灌中的压力控制，特别是大型气灌，时间常数要慢一些。过程增益的量值很大程度上取决于压力的物理单位，例如 Bar 或 Pa。通常压力传感器产生的测量噪声比温度传感器高。

- **流量控制系统仿真：**如果定义阀门位置，使其在 0% 时关闭在 100% 时打开，则过程 Gain 通常为正值，因为阀门打开时流速增加。如果阀门关闭时完全停止流动，即阀门紧密关闭，则 PV0=0。时间常数明显快于温度控制中的时间常数，但通常为同一数量级。过程增益的量值很大程度上取决于流量的物理单位，例如，m3/s 或 l/min。影响流量传感器的测量噪声通常比温度传感器的测量噪声大。

要实现过程仿真的停滞时间，可以在 ProcSimC 输入之前插入 DeadTime 块并在不同的循环中断 OB (OB3x) 中调用它。

模型变体

提供两个不同的模型变体：

1. 连续过程仿真 ProcSimC，其中 MV 输入为模拟值，例如加热功率或阀门位置。
2. 步控制器的过程仿真 ProcSimS，通过两个二进制输入“调大”/“调小”或“打开”/“关闭”来控制执行器。执行器在内部模拟为积分器，其中 MotorHL = 100%, MotorLL = 0%, TI = MotorTime。积分器输入根据下列公式从二进制输入获得：

$$I_{\text{Integ.Input}} = \begin{cases} 100 & \text{if } Up = True \\ -100 & \text{if } Down = True \\ 0 & \text{otherwise} \end{cases}$$

噪声发生器功能块

噪声发生器作为已编译块 NoiseGen 提供并提供有少量的在线帮助。该块的任务是生成具有指定的标准偏差和偏移的伪随机噪声信号。算法来自于 <http://www.code4gold.net/tut/zufall.pdf>。

I/O (参数)	含义	数据类型	默认值	类型
Enable	Noise generation enabled	BOOL	1	I
Offset	Constant offset	REAL	0	I
Noise	Output noise sample	REAL	0	O
Restart	Restart (Initialization)	BOOL	1	I
StdDev	Standard deviation of the noise signal (Variance = StdDev^2)	REAL	1	I

所有 I/O 在 CFC 中均可见并可互连，并且不用于面板中的操作和监视功能。

15.2.2 使用热流对温度进行串级控制

“使用热流对温度进行串级控制”模板的描述

此模板包含流量和温度控制系统的仿真模型及干扰发生器块的参数（请参见 I/O 表）。可以使用该模型测试在 串级控制 (页 748)部分所描述的模式转移。也可以试验主控制器和次级控制器的不同参数集的属性。以下是此类应用的典型特征：

- 温度控制系统较流量控制系统慢。
- 有两个差异很大的时间常数。
- 存在对应环境温度的偏移。
- 干扰较流量控制系统少。

级联控制实例项目 (APC_ExaSP) 的过程参数

ProcSimC	Gain	TimeLag1	TimeLag2	PV0	NoiseVariance
流量控制回路	8	1	1	0	0.05
温度控制回路	0.3	8	1	20	0.01

控制响应速度快的 PID 控制器 --> PI 级联的参数

下表中列出的参数适用于控制偏差小而执行机构干预强的快速控制响应。

PID	Gain	TI	TD
TIC-501	10	8.44	2.5
FIC501	0.1	1.4	0

具有软控制器干预的 PI --> P 级联的控制器参数

对于下表中列出的参数，其优势在于控制器可以“轻松处理”最后的控制元件（如阀门）。

PID	Gain	TI	TD
TIC-501	10.5	6.8	0
FIC501	0.1	0	0

通常建议使次级控制器比主控制器“更简单”，即减少不同动态通道的数量，使其名副其实的成为主控制器的次级控制器。

次级回路中的稳态控制偏差通常与应用不相关。另一方面，次级回路的响应时间很重要，因为次级闭环控制回路的时间常数是主控制器控制系统的一部分。如果由于上述原因没有在次级控制器中加入 I 动作，则建议您不要在次级回路中将次级控制器的设定值范围精确地限制为过程值可达到的实际范围，因为这样可能会因存在稳态偏差而无法使用次级控制器的整个执行范围。相反，您应该为次级控制器设置更松的设定值限制，以及为主控制器设置更松调节变量限制。主控制器的抗 Windup 措施主要通过 INT_HNEG 和 INT_HPOS 的互连来解决。如果次级控制器不具有任何 I 动作，则无法进行平稳的手动/自动切换。因此，应该设置接近过程运行点的典型 LMN 值的 LMN_OFF。

用于加热和/或冷却介质流并使用了次级控制器的串级温度控制系统通常用于

- 热交换器
- 没有冷却套管的反应釜

更多应用实例

- 基于回流比（次级控制器在塔顶）和加热蒸汽流速（次级控制器在塔槽）的蒸馏塔（主控制器）的温度控制，
- 使用次级控制器控制燃料流速的火炉温度控制，
- 使用次级控制器控制入口和/或出口流量的容器填充液位控制。
- 使用次级控制器控制速度和扭矩的位置控制（在驱动工程中）。

次级控制器的使用

次级控制器通常用于流量控制，以防止流速变化对主控制器造成有害影响。因为次级闭环回路具有线性响应，因此通常在流量控制元件（如阀门）中出现的非线性特征在次级回路中会被“隐藏”，从而非线性特征不至于影响主控制器及其调节。

15.2.3 具有有色噪声仿真的控制回路监视

“具有有色噪声仿真的控制回路监视”模板的描述

CPM 块与 PID 控制器的互连可在过程标签类型 PIDCTRL_ConPerMon 中找到（请参见具有安全逻辑和控制回路监视的 PID 控制器 (页 742)）。实例项目 (APC_ExaSP) 支持并帮助您熟悉控制回路监视的概念和功能。为此，该模板包含使用干扰模型的过程仿真。有色噪声是借助整形滤波器通过白噪声信号生成的。这会生成连续的干扰信号，其中还包含在闭环控制回路带宽低频范围内的有功部分。因此，PID 控制器可以补偿部分干扰，而任何控制器都无法校正高频测量噪声。

应用

调试完控制器和 CPM 块之后，应该可以看到以下演示控制回路监视功能的操作的效果。

- 将控制器切换为手动模式：
控制变量的方差将增大但 CPI 会变成失效，因为除非控制回路闭合，否则不能说明控制是适合的。
- 更改过程仿真的参数，例如将 TimeLag2 从 2s 改为 8s：
这种过程动态特性的恶化（例如由于磨损或断裂）导致控制质量下降，这一点早在能够通过观察标准 PV 趋势发现之前就可通过 CPI 值判断出来。如果控制质量下降到定义的级别以下，就会生成一个 CPI 警告甚至是报警。
- 要求控制器进行设定值阶跃变化：
CPI 将暂时变为失效，因为控制质量的所有随机特征（如方差）均基于一个具有恒定平均值的稳态假设。从 CPM 面板的下拉列表框中选择“设定值”视图，以便观察确定性特征（如超调量和稳定比）。在新设定值处再次达到稳定状态且整个时间窗填充了此稳定状态的数据后，将自动重新激活对随机特征的监视。

有关 CPM 块的详细信息及解释其显示的说明，可参考该块的在线帮助。

15.2.4 补偿可测量变量的动态前馈

“补偿可测量干扰变量的动态前馈”模板的描述

当已知的强干扰影响过程并且其原因可测量时，可以使用 **Feedforward**。在上述情况下，有以下常规策略适用：“尽可能控制（如果干扰事先知道且已由模型描述），按最大需要控制（余下部分包括模型误差和不可测的干扰）”。

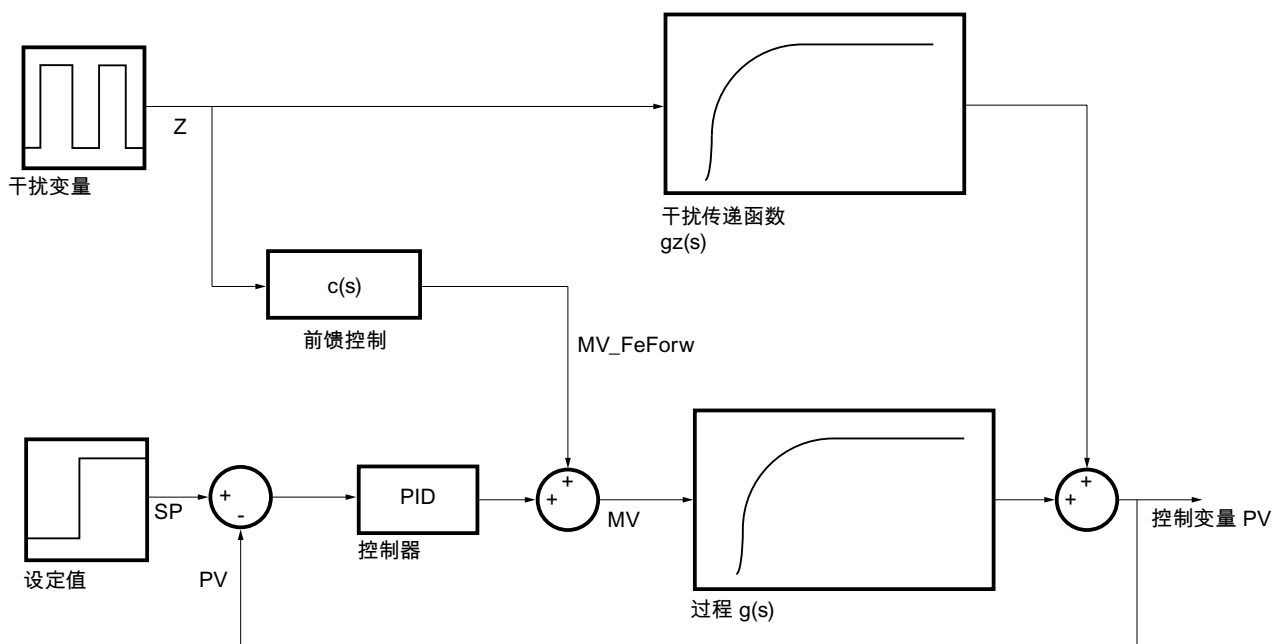


图 15-4 前馈控制

当控制器在手动模式下运行时，可以以传递函数 $g_z(s) = y(s) / z(s)$ 的形式估计可测量干扰的影响，这样调节变量便不会导致控制变量 $y = PV$ 发生任何变化，因而所有变化均可认为是由干扰 $z(s)$ 所致。

理想前馈控制的传递函数 $c(s)$ 可由以下要求得出，即对于任何干扰信号 $z(s)$ ， z 对 y 的影响应为零：

$$g_z(s)z + c(s)g(s)z = (g_z(s) + c(s)g(s))z = 0$$

为满足此方程，补偿块必须尽可能逼近该方程

$$c(s) = -\frac{g_x(s)}{g(s)}$$

这意味着干扰传递函数 $g_x(s)=y(s)/z(s)$ 必须已知，且主受控系统的传递函数 $g(s)=y(s)/u(s)$ 、 $u=MV$ 必须置反。如果这两个传递函数均可建模为带有时滞的一阶函数

$$g(s) = \frac{k_s}{1 + t_1 s} \cdot e^{-s\theta}$$

并且 $\theta < \theta_x$ 适用，则生成的补偿元素必须表示传递函数

$$c(s) = -\frac{k_{sz}}{k_s} \frac{1 + t_1 s}{1 + t_{1z} s} e^{-s(\theta_z - \theta)}$$

总之，附加前馈控制需要以下动态传递函数：

$$FFwd(s) = -k_c \frac{t_{cd}s + 1}{t_{cl} + 1} \cdot e^{-\theta_c s} \cdot z(s)$$

在上述实例中，此函数包含以下参数：

$$k_c = \frac{k_{sz}}{k_s}, \quad t_{cd} = t_1, \quad t_{cl} = t_{1z}, \quad \theta_c = \theta_z - \theta'$$

通过使用基本 CFC 块的组合作为 DT1 (DIF_P) 和 PT1 (PT1_P) 块的串联，可在控制器外部创建此传递函数。

输入参数 k_c ， t_{cd} ， t_{cl} 必须由用户设置。对于静态前馈控制，两个时间常数均设置为零。

在模板中使用以下参数集：

主控制系统：

$$g(s) = \frac{2}{2s+1} e^{-1.2s}$$

干扰传递函数：

$$g_x(s) = \frac{1}{3s+1} e^{-1.6s}$$

PID: Gain= 0.197、TI= 1.9、TD= 0

前馈控制：

$$c(s) = -\frac{g_x(s)}{g(s)} = -\frac{1}{2} \cdot \frac{2s+1}{3s+1} e^{-0.4s}$$

同样的过程仿真设置两次，其中一个实例具有干扰前馈，另一个则没有（所有其它过程参数和控制器参数相同）。通过直接比较（“基准仿真”或“二者对抗”）可测试前馈控制的优势。

应用实例

- 通过蒸汽压或加热/冷却介质流来控制热交换器的出口温度。介质的流量和入口温度是可测量干扰变量。
- 使用入口量在汽包式蒸汽锅炉中进行填充液位控制。流出量是可测量干扰变量，由工厂中的可变蒸汽消耗量确定。
- 使用回流比率或加热蒸汽量在蒸馏塔中进行温度控制。可测量干扰变量是混合输入量。
- 使用冷却介质流和排放量在搅拌槽反应釜中进行温度和浓度控制。流入量的温度（或许还包括浓度）是可测量干扰变量。

15.2.5 针对非线性过程的面向运行点的参数调整（增益调整）

“针对非线性过程的面向运行点的参数调整”模板的描述

由于存在非线性物理、化学或热力学效应，许多工艺过程具有非线性响应。如果需要将这样的过程保留在非常接近某个固定运行点的范围内，则可围绕此运行点线性化传递响应。可以针对该线性化传递函数设计一个线性 PID 控制器。不过，如果此过程具有很强的非线性响应和/或在不同的运行点处运行，则在整个运行范围内不会获得持续的良好控制响应。由于存在这种非线性，将在不同的运行点处使用不同的增益系数或过程时间常数。为与之保持一致，不同的控制器参数将被视为是最优的。

Gain-Scheduling

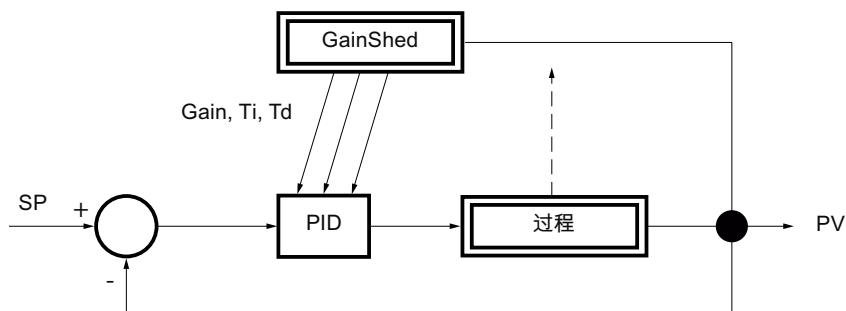


图 15-5 非线性过程的增益调整

此问题的一个可能（最简单）解决方案称为增益调整或参数调度。使用诸如 PCS 7 PID 调节器这样的工具，在不同的运行点处执行各种试验，每次试验的信号幅值都很低。这样对于每个运行点可得到不同的 PID 参数集。在 GainSched 功能块中最多可存储三个这样的参数集。根据描述过程状态的连续可测量变量（通常是控制变量 PV 本身）来选择合适的参数集。在具有精确参数值的运行点之间，通过邻近插值点的线性插值来计算这些值，以便在运行点之间实现柔和的平稳过渡。术语“参数调度”明确表明是事先指定调整参数的“时间表”。相反，自适应控制器会根据运行期间不同的过程响应自动自我调整。

功能块 GainSched 是通过将 CFC 图表“fbGainSched”编译为块类型来生成的。此 CFC 图表随库一起提供，因此用户可以根据需要来扩展现有基本功能，例如扩展到多于三个运行点。

注： 如果从数学角度来考虑，组合几个通过增益调整局部优化过的控制器而构成的非线性控制器并不一定就是适合非线性过程的最优非线性控制器。当在不同的运行点之间进行设定值阶跃变化时，即使是具有良好的非线性特征（连续且可以求微分），也可清楚地了解这一点。对于不连续或无法求微分的非线性特征，或非单调的非线性特征，则需要高度注意。

使用参数

在实例项目 (APC_ExaSP) 中，根据运行点的不同，将基于折线更改两个最重要过程参数的设置。下表给出了实例的过程参数和控制器参数。

运行点	X=PV	ProcSim.Gain	ProcSim.TmLag1	ProcSim.TmLag2	Gain	Ti	Td
1	20	4	5	10	0.6	14.7	3.7
2	100	3	3	10	1	8.8	2.2
3	200	2	1	10	10	4.1	1.1

同样的过程仿真设置两次，其中一个实例具有增益调整，另一个则没有（所有其它过程参数和控制器参数相同）。通过直接比较（“基准仿真”或“二者对抗”）可测试增益调整的优势。

15.2.6 超驰控制

模板“超驰控制”的描述

在超驰控制中，两个或更多控制器共用一个公用执行器。根据当前的过程状态，决定哪个控制器实际具有访问执行器的权限，换言之，不同控制器可以相互超越。

典型使用案例是通过单阀进行压力和流量控制的气体管道。主要控制目标是实现一个特定的流速，但出于安全考虑，压力必须保持在某一限制之内。因此压力控制器称为“限制控制器”或“次级控制器”。

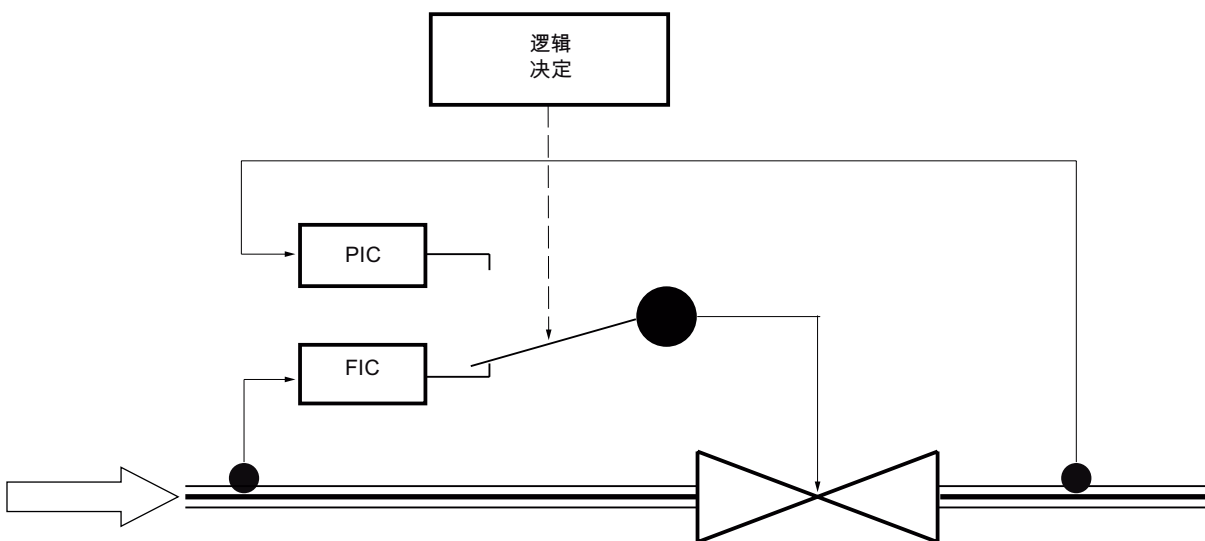


图 15-6 具有主控制器 FIC 和限制控制器 PIC 的超驰控制

这类超驰控制的标准

根据会导致两种不同超驰控制的两个不同的标准，可对应该激活哪个控制器做出合理决定：

1. 该决定基于可测量过程的输出变量，例如两个控制变量中的一个。在上述实例中，可以使用压力控制器的警告限值来决定是否激活压力控制器。被动控制器处于跟踪模式下，以避免饱和问题和确保无扰动切换。次级控制器的设定值必须略低于切换阈值，以便可以再次反向传递。此类超驰控制易于理解和实现。其优点在于可以监视次级控制变量（如压力）的上限和下限；而其缺点是一旦限制控制器需要干预，就会产生极限环振荡。次级控制器将始终尝试使其控制变量返回到安全范围以及将命令返回给主控制器（如流速），因而主动和被动控制器会连续交换。因此，建议只有在很少需要次级控制器并且其主要用作安全或备用系统时才使用此方案。
2. 可基于对两个控制器的调节变量的比较结果来做决定，例如，要求更高（或更低）控制变量的控制器控制执行器。在上述实例中，要进一步打开阀的控制器会接管控制。次级控制器的设定值定义切换阈值。两个控制器一直在自动模式下运行。为避免出现饱和问题，必须跨结构跟踪调节变量的限值：如果更高（更低）的调节变量优先，则必须对所有控制器或当前最高（最低）调节变量的下（上）限向上或向下进行轻微校正，如调节变量范围的 2% 大小。这意味着还可以在具有多于两个控制变量的应用中使用此方案。上限处不存在饱和问题，因为无论如何最大调节变量都会接管控制。此方法避免了方案 1 中的极限环振荡问题，但原则上是不对称的，即只能监视次级控制变量的上限或下限，而不能二者全监视。

在大多数控制类教科书（尤其在美国）中，都对此类超驰控制进行了介绍。不过，它只能结合允许在线调节调节变量限值的 PID 算法一起使用（在从 V6.0 开始的 PCS 7 中）。

使用参数

在该仿真实例中使用以下参数：

主过程（流量控制）：

$$g(s) = \frac{3}{(2s+1)^2}$$

，即阀打开时流量增加，阀关闭时流量为零。

PI 流量控制器： Gain= 0.33、TI= 2.7

次级过程（压力控制）：

$$g_p(s) = \frac{-0.8}{(7s+1)(1s+1)}$$

，即阀打开时压力增加，阀完全打开时压力为 80 巴。

PI 压力控制器： Gain= -2.8、TI= 4

切换限值 15 巴 < 压力 < 70 巴。

更多应用实例

- **蒸汽发生器：** 主控制变量是蒸汽压力，但必须监视蒸汽槽中的水位，以便加热线圈始终完全浸在水中且槽中的水不会溢出。唯一的调节变量是出口阀。
- **压缩机：** 主控制变量是生产量，但必须监视压力以确保其不会超过安全限制。唯一的调节变量是电机转速。
- **蒸汽配汽系统：** 每个涉及工业过程的工厂都具有管道网络，以在整个工厂中分配各种压力的蒸汽。通过一个阀门将高压蒸汽降为较低压力级别。主控制变量是较低压力阶段的压力，但还必须监视高压管路中的压力以确保其不会超过安全限制。

15.2.7 使用史密斯预测器应对时滞的 PID 控制器

“使用史密斯预测器应对时滞的 PID 控制器”模板的描述

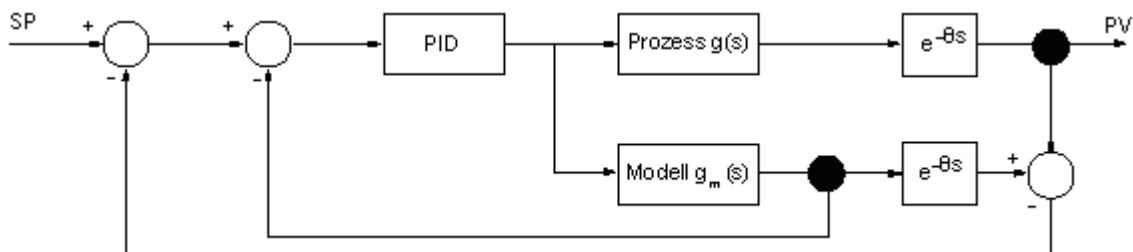


图 15-7 以 IMC 表示的史密斯预测器

在时滞大（相对于主要时延常量）的过程中，必须将标准 PI 控制器设置得很慢，因此控制质量会有所下降。使用史密斯预测器可以显著改进控制质量，该预测器源自基于模型控制的 IMC（Internal Model Control，内部模型控制）原理。为此，控制系统的传递函数 $g_s(s) = g(s) e^{-s\theta}$ 被分为两部分，一个没有时滞的部分 $g(s)$ 和一个时滞为 θ 的纯时滞部分 $e^{-s\theta}$ 。在实际过程中，只能测量受时滞影响的控制变量 y 。但是，可以通过过程模型实际估计一下不受时滞影响的控制变量（将成为控制器的一部分），并将估计值提供给控制器。这表明控制器本身可以针对没有时滞的过程专门进行设计，从而能够更紧密地设置控制器。为补偿未知干扰，在模型中评估受时滞影响的控制变量，并将其与实测的控制变量相比较。此差异也反馈给控制器。

就实际应用而言，必须指出的是，史密斯预测器的性能很大程度上取决于模型拟合度，即首先时滞必须为常量，其次该常量必须已知。

注：要控制具有大时滞的过程，模型预测控制器（请参见基于模型的预测控制（页 765））也适合单变量的情况。由于采用集成的设计步骤，因此它在系统建模方面能够提供更大的灵活性从而更为方便，但它需要更多的 CPU 资源。

在此模板中，同样的过程仿真设置两次，其中一个实例具有史密斯预测器，另一个则没有（所有其它过程参数相同）。通过直接比较（“基准仿真”或“二者对抗”）可测试史密斯预测器的优势。

15.2.8 过滤控制回路中的噪声测量值

“过滤控制回路中的噪声测量”模板的描述

该实例说明了在闭环控制回路中使用新 SIG_SMTM 块方法。无需专业知识即可将该块连接到任何信号源，因此不需要特殊的过程标签类型。此仿真模板在通过仿真测试低通滤波器对闭环控制回路的影响方面很有用。增大滤波时间常数会改进平滑效果，但同时会引起控制回路中的相位滞后，而相位滞后会控制质量甚至稳定性产生不良影响。

使用的参数

在该仿真实例中使用以下参数：

过程传递函数：

$$g(s) = \frac{3}{(15s+1)(2s+1)}$$

，在输出信号中有白噪声。

PI 控制器： Gain = 0.5、TI = 7s、采样时间 = 0.1s

Butterworth 滤波器： TimeConstant = 3s。在 0.3s 时，几乎很难发现平滑效果；在 15s 时，已经可看到控制质量明显大大降低。

一些典型应用领域（如压力传感器）和敏感的执行器（如阀门），都有一些信号受噪声强烈影响的过程。

有关 SIG_SMTM 块的详细信息，可参考该块的在线帮助。

15.2.9 基于模型的预测控制

“模型预测控制”模板的描述

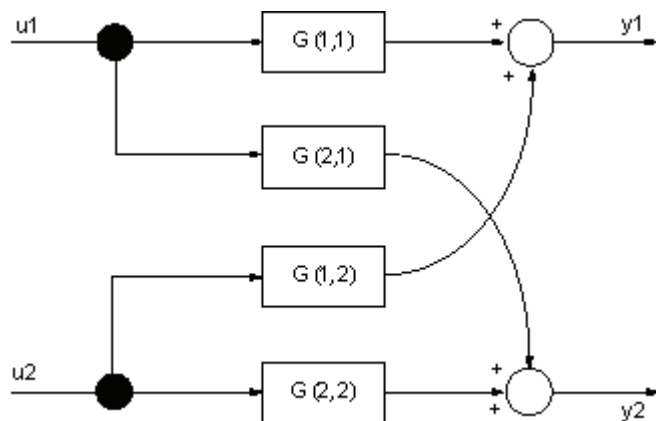


图 15-8 具有 p 规范结构的 MIMO 2x2 过程

此模板说明了应用 MPC 块模拟一个包含以下四个传递函数的 2x2 多变量过程的情况：

$$\underline{G}(s) = \begin{bmatrix} G(1,1) & G(1,2) & \cdots & G(1,n_u) \\ G(2,1) & G(2,2) & \cdots & G(2,n_u) \\ \vdots & \vdots & \ddots & \vdots \\ G(n_y,1) & G(n_y,2) & \cdots & G(n_y,n_u) \end{bmatrix} = \begin{bmatrix} \frac{3}{(30s+1)(4s+1)} & \frac{1.2}{(34s+1)(14s+1)(6s+1)} \\ \frac{1.3}{(28s+1)(12s+1)(6s+1)} & \frac{4}{(26s+1)(6s+1)} \end{bmatrix}$$

其中 $n_y = 2 =$ 控制变量的数目， $n_u = 2 =$ 操作变量的数目， $G(i_y, i_u)$ 是从输入 i_u 到输出 i_y 的传递函数。这个最简单的多变量控制系统有助于新手熟悉基于模型的多变量控制器的概念和应用。此模板还展示了用户如何能够通过添加额外的功能来扩展 MPC 块：使用 MEAS_MON 块报告外部报警，使用 CPM 块监视控制质量。测量值失败的安全逻辑未包含在实例项目 (APC_ExaSP) 中，而是包含在过程标签类型中。

有关 MPC 块的详细信息，可参考该块的在线帮助。

有关在多变量系统中使用 CPM 块的说明

CPM 块的数学原理是针对单变量应用而设计的。如果在多变量控制的通道中检测到任何方差增大，则 CPM 算法无法确定此问题是由自身的内部控制通道引起还是由邻近通道的交互而引起。但是，可以在多变量系统的每个控制通道中都包含一个 CPM 块，以监视操作期间的控制质量是否保持在调试期间所确定的范围之内。为此，在每个 CPM 块的 ManSuprCPI 输入位之前必须执行几个逻辑运算：

- 如果由于本地原因（如设定值阶跃变化）导致多变量系统的一个或几个其它通道未处于输出位 CPI_SuRoot=true 所指示的稳定状态 ("root caused in this channel")，则其自身通道中的方差增加无法避免，因此不应在其自身通道中触发 CPI 警告。
- 如果多变量系统的一个或几个其它通道与输出位 CPI_WrnAct = true 所指示的内容明显不同，则其自身通道中的方差增加无法避免，因此不应在其自身通道中触发 CPI 警告。这或许对找出问题的真正原因有所帮助。最先在其中检测到过度差异的通道输出第一个报警；其它可能仅受到第一个错误引发的错误所影响的通道并不生成其自己的报警。

说明

PCS 7 库中的模板（高级过程控制模板）包括一个如何将 MPC 块与 CPM 块结合使用的实例。

应用实例

- 蒸馏塔质量控制，如基于回流比率和加热蒸汽量对塔顶和塔槽温度进行控制
- 具有多个燃烧器的加热炉的多个相邻区域的温度控制，如隧道炉、玻璃熔化设备、玻璃注料道等
- 通过调整反应条件（如压力或温度等）进行的化学反应釜质量控制
- 蒸馏器，如汽包式蒸汽锅炉
- 磨机，如水泥磨机

参见

具有安全逻辑和控制回路监视的 PID 控制器 (页 742)

15.2.10 示例项目 APC_ExaSP

15.2.10.1 PCS 7 高级过程控制模板简介

简介

与模板（可插入的过程标签类型）相比，仿真模板（示例项目“APC_ExaSP”）主要用于培训目的：主要目的是通过允许用户进行实验而不必干预实际过程来使用户熟悉新的高级控制结构。模板提供实际过程仿真。使用这些模板可以帮助您理解概念和特定结构要求，并在实际系统中实现这些功能之前对其使用进行评估。因此，这些实例项目 (APC_ExaSP) 中包含一个具有增益、等效值和测量噪声的三阶仿真模型，但不包含模拟驱动程序块。过程模型作为“ProcSimC”CFC 图表提供并使用图中图技术并入模板中。

PCS 7 示例项目“APC_ExaSP”中包括下列模板：

- 基于加热介质流量的温度串级控制 (CaskadeSim)。
- 具有粉红噪声干扰的过程的控制回路监视 (ConPerMonSim)。
- 用来补偿可测量干扰变量的动态前馈控制 (DisturCompSim)。
- 针对非线性过程的面向运行点的参数调整（增益调整）(GainSchedSim)。
- 主控制器和限制控制器访问同一个执行器的超驰控制 (OverrideSim)。
- 用于控制长停滞时间过程的带有史密斯预测器的 PID (SmithPredictorSim)。
- 控制回路中噪音测量值的过滤 (SigSmoothSim)。
- 模型预测控制 (ModPreConSim)。

使用“FuzzyControl++”PCS 7 可选软件包的 Fuzzy Control 的模板可从 *Siemens I&S Web* 页下载，因此演示项目中未包含该模板。

在 PCS 7 示例项目“APC_ExaSP”中为每个仿真模板提供了一个单独的层级文件夹（“Unit”）。每个文件夹包含一个带有互连实例的 CFC 图表、简要的说明性文本和已分配 OS 画面，该画面带有基于预组态趋势记录器的过程实例的自说明显示。OS 画面中的简短文本描述调试和表示。

15.2.10.2 标签类型

“模板”(Templates) 文件夹中的 PCS 7 库包含以下过程标签类型:

有关 PCS 7 库中的模板（可插入的过程标签类型）的控制工程信息，可参考此帮助中的专门内容:

- 具有安全逻辑和控制回路监视的 PID 控制器 (PIDCTRL_ConPerMon)
- 直接访问执行器的无位置反馈步控制器 (Step_CTRL_Direct)
- 分程控制器 (SPLITRING)
- 比率控制器 (RATIO)
- 串级控制 (Cascade CTRL)
- 连续 PID 控制器 (PIDCTRL)
- 具有前馈控制的 PID 控制 (PIDCTRL_DistComp)
- 具有面向工作点参数调整（增益调整）的 PID 控制 (PIDCTRL_GainSched)
- 超驰控制 (PIDCTRL_Override)
- 具有史密斯预测器的 PID 控制 (PIDCTRL_SmithPredictor)
- 模型预测控制 (MPC_CTRL)

“模板”(Templates) 文件夹包含仿真实例中显示的多个控制回路结构的原型过程标签类型。这些原型是此类过程标签类型可能如何出现的简单示例。在大部分过程工厂中，假定必须将一些高级控制回路结构组态为与低级控制回路结合使用。这意味着，作为过程标签类型实例的高级控制回路结构的大量产生表示异常。

索引

A

ADD4_P, 528

I/O, 528

描述, 527

ADD8_P, 529

I/O, 529

描述, 529

AL_DELAY

工作原理, 521

ASSETMON, 505, 511, 514, 651

I/O, 511

描述, 505

操作和监视, 514

面板, 651

AVER_P, 532

I/O, 532

描述, 530

B

BATCH 功能

MPC, 273

C

CH_AI, 338

I/O, 338

描述, 331

CH_AO, 340, 344

I/O, 344

描述, 340

CH_CNT, 345, 350

I/O, 350

描述, 345

CH_CNT1, 353, 360

I/O, 360

描述, 353

CH_CNT2C

I/O, 366

应用领域, 362

CH_CNT2M

I/O, 372

应用领域, 368

CH_DI, 374, 377

I/O, 377

描述, 374

CH_DO, 378, 381

I/O, 381

描述, 378

CH_MS, 382, 387

I/O, 387

描述, 382

CH_U_AI, 389, 396

I/O, 396

描述, 389

CH_U_AO, 398, 403

I/O, 403

描述, 398

CH_U_DI, 405, 409

I/O, 409

描述, 405

CH_U_DO, 411, 415

I/O, 415

描述, 411

- ConPerMon
 - I/O, 61
 - 报告, 60
 - 消息响应, 60
 - 消息文本, 60
 - 消息类别, 60
- COUNT_P
 - I/O, 535
 - 描述, 533
- CPM
 - 功能, 48
 - 启动特征, 45
 - 图形评估和长期统计, 55
 - 安装在 OB 中, 43
 - 对象名, 42
 - 寻址, 43
 - 工作原理, 43
 - 应用领域, 42
 - 报警延迟, 57
 - 故障排除, 59
 - 时间响应, 45
 - 模式, 48
 - 特殊情况串级控制和多变量控制, 56
 - 状态字分配, 46
 - 用于确定参考方差的供选方案, 56
 - 监视控制质量的确定特征, 51
 - 监视控制质量的随机特征, 49
 - 组态, 43
 - 计算质量代码, 57
- CTRL_PID
 - I/O, 79
 - VSTATUS, 86
 - 启动特征、动态响应和消息响应, 77
 - 手动、自动和跟踪模式, 72
 - 描述, 65
 - 操作和监视, 86
 - 方框图, 78
 - 模式更改, 74
 - 消息文本和关联值, 84
 - 设定值的生成, 限制值和错误信号, 68
 - 调节变量的生成, 70
 - 错误处理, 76
- CTRL_PID 所有视图, 583
- CTRL_PID 的动态响应, 77
- CTRL_PID 的启动特征, 77
- CTRL_PID 的消息响应, 77
- CTRL_PID 的设定值和实际值分支中的信号处理, 68
- CTRL_PID 级联, 72
- CTRL_PID 调节变量的生成, 70
- CTRL_S, 91, 93, 104
 - I/O, 107
 - VSTATUS, 116
 - 信号处理, 91
 - 启动特征, 104
 - 手动、自动和跟踪模式, 97
 - 控制信号的生成, 93
 - 描述, 87
 - 操作和监视, 116
 - 方框图, 105
 - 时间响应, 104
 - 模式更改, 100
 - 消息响应, 104
 - 消息文本和关联值, 114
 - 错误处理, 103
- CTRL_S 所有视图, 592
- CTRL_S 控制信号的生成, 93
- CTRL_S 的启动行为、动态响应和消息功能, 104
- CTRL_S 的方框图, 105
- CTRL_S 的设定值和实际值分支中的信号处理, 91
- CTRL_S 级联, 97

D

- DEADT_P, 117, 119
 - I/O, 119
 - 描述, 117
- DIF_P
 - I/O, 122
 - 描述, 120
- DIG_MON
 - I/O, 126
 - VSTATUS, 130
 - 描述, 123
 - 操作和监视, 130
 - 消息文本和关联值, 128
- DIG_MON 所有视图, 599
- DOSE, 136, 140, 142
 - I/O, 136
 - VSTATUS, 142
 - 描述, 131
 - 操作和监视, 142
 - 消息文本和关联值, 140
- DOSE 所有视图, 600

E

- ELAP_CNT, 147, 148
 - I/O, 145
 - VSTATUS, 148
 - 描述, 143
 - 操作和监视, 148
 - 消息文本和关联值, 147
- ELAP_CNT 所有视图, 608

F

- FF 设备, 727
 - 在资产中集成, 727
- FF_A_AI
 - I/O, 420
 - 描述, 416
- FF_A_AO
 - I/O, 428
 - 描述, 423
- FF_A_DI
 - I/O, 436
 - 描述, 432
- FF_A_DO
 - I/O, 443
 - 描述, 438
- FM 355, 193
 - 备份模式, 164, 193
- FM 355 模块, 291
- FM 355 的备份模式, 164, 193
- FMCS_PID, 165
 - I/O, 165
 - VSTATUS, 174
 - 使用过程映像采集和写入过程值, 154
 - 出错处理, 162
 - 功能, 152
 - 启动特征、动态响应和消息响应, 163
 - 手动、自动和跟踪模式, 158
 - 描述, 149
 - 操作和监视, 174
 - 模式更改, 160
 - 消息文本和关联值, 172
- FMCS_PID 所有视图, 610
- FMCS_PID 的功能, 152
- FMCS_PID 的启动特征, 163
- FMCS_PID 的时间响应, 163

FMCS_PID 的消息响应, 163

FMT_PID, 180, 191, 200, 202

I/O, 193

VSTATUS, 202

功能, 178

启动特征、动态响应和消息响应, 191

手动、自动和跟踪模式, 184

描述, 175

操作和监视, 202

模式更改, 186

消息文本和关联值, 200

通过过程映像采集和写入过程值, 180

错误处理, 190

FMT_PID 所有视图, 620

FMT_PID 的功能, 178

FMT_PID 的启动特征, 191

FMT_PID 的时间响应, 191

FMT_PID 的消息响应, 191

G

GAIN_SHD

I/O, 207

功能, 206

启动特征, 205

安装在 OB 中, 204

对象名, 203

寻址, 204

工作原理, 203

应用领域, 203

控制器参数的手动设置, 206

时间响应, 205

消息响应, 205

组态, 204

I

I/O

ConPerMon, 61

GAIN_SHD, 207

ModPreCon, 275

SIG_SMTH, 298

I/O 属于, 26, 29, 34, 38, 119, 136, 212, 215, 220, 223, 289, 293, 302, 305, 338, 344, 350, 360, 377, 381, 387, 396, 403, 409, 415, 456, 463, 471, 477, 485, 493, 501, 511, 520, 528, 529, 532, 540, 542, 544, 574, 578

ADD4_P, 528

ADD8_P, 529

ASSETMON, 511

AVER_P, 532

CH_AI, 338

CH_AO, 344

CH_CNT, 350

CH_CNT1, 360

CH_DI, 377

CH_DO, 381

CH_MS, 387

CH_U_AI, 396

CH_U_AO, 403

CH_U_DI, 409

CH_U_DO, 415

COUNT_P, 535

CTRL_PID, 79

CTRL_S, 107

DEADT_P, 119

DIF_P, 122

DIG_MON, 126

DOSE, 136

ELAP_CNT, 145

FF_A_AI, 420

FF_A_AO, 428

- FF_A_DI, 436
 FF_A_DO, 443
 FMCS_PID, 165
 FMT_PID, 193
 INT_P, 212
 INTERLOK, 215
 LIMITS_P, 220
 MEANTM_P, 538
 MEAS_MON, 223
 MESSAGE, 581
 MOT_REV, 232
 MOT_SPED, 242
 MOTOR, 253
 MS_MUX, 516
 MSG_NACK, 578
 MUL4_P, 540
 MUL8_P, 542
 OB1_TIME, 525
 OP_A, 553
 OP_A_LIM, 557
 OP_A_RJC, 561
 OP_D, 565
 OP_D3, 570
 OP_TRIG, 574
 PA_AI, 456
 PA_AO, 463
 PA_DI, 471
 PA_DO, 477
 PA_TOT, 485
 POLYG_P, 280
 PT1_P, 283
 R_TO_DW, 544
 RAMP_P, 286
 RATIO_P, 289
 RCV_341, 493
 READ355P, 293
 REC_BO, 26
 REC_R, 29
 SEND_BO, 34
 SEND_R, 38
 SND_341, 501
 SPLITR_P, 302
 ST_MUX, 518
 STATEREP, 520
 SWIT_CNT, 305
 VAL_MOT, 314
 VALVE, 323
 INT_P, 209, 212
 I/O, 212
 描述, 209
 INTERLOK, 213, 215, 217
 I/O, 215
 VSTATUS, 217
 描述, 213
 操作和监视, 217
- L**
- LIMITS_P, 220
 I/O, 220
 描述, 218
- M**
- MEANTM_P
 I/O, 538
 描述, 536
 MEAS_MON, 221, 223, 225, 226
 I/O, 223
 VSTATUS, 226
 描述, 221
 操作和监视, 226

- 消息文本和关联值, 225
- MEAS_MON 所有视图, 627
- MESSAGE, 582
 - I/O, 581
 - 描述, 579
 - 消息文本和关联值, 582
- MODB_341, 504
 - 描述, 504
- MODE, 709
- ModPreCon
 - I/O, 275
- MOT_REV, 235, 236
 - I/O, 232
 - VSTATUS, 236
 - 描述, 227
 - 操作和监视, 236
 - 消息文本和关联值, 235
- MOT_REV 所有视图, 630
- MOT_SPED, 246, 247
 - I/O, 242
 - VSTATUS, 247
 - 描述, 237
 - 操作和监视, 247
 - 消息文本和关联值, 246
- MOT_SPED 所有视图, 633
- MOTOR, 256, 257
 - I/O, 253
 - VSTATUS, 257
 - 描述, 248
 - 操作和监视, 257
 - 消息文本和关联值, 256
- MOTOR 所有视图, 636
- MPC
 - BATCH 功能, 273
 - 内部设置设定值, 266
 - 功能, 265
 - 启动特征, 260
 - 基于模型的干扰变量补偿, 270
 - 安装在 OB 中, 260
 - 对象名, 258
 - 寻址, 260
 - 工作原理, 259
 - 应用领域, 258
 - 手动模式下的设定值跟踪, 267
 - 抗饱和, 269
 - 控制误差生成和死区, 268
 - 故障诊断, 274
 - 方形或非方系统控制, 271
 - 时间响应, 260
 - 模式, 263
 - 消息响应, 260
 - 状态字分配, 261
 - 生成和限制调节变量, 266
 - 线性和非线性系统控制, 271
 - 组态, 260
 - 设定值滤波器, 267
 - 调节值跟踪, 266
 - 质量代码的显示和输出, 273
 - 错误号概述, 274
 - 预测控制器算法, 269
- MS, 702
- MS 的维护状态, 732
- MS_MUX
 - I/O, 516
 - 描述, 515
- MSG_NACK, 576, 578
 - I/O, 578
 - 描述, 576
- MSG_TS
 - 描述, 447
 - 消息文本, 451
- MUL4_P, 540

I/O, 540
 描述, 539
 MUL8_P, 542
 I/O, 542
 描述, 541

N

NOISE_GN
 对象名, 278

O

OB1_TIME
 I/O, 525
 描述, 523
 OP_A, 553
 I/O, 553
 描述, 551
 操作和监视, 553
 OP_A_LIM, 557
 I/O, 557
 描述, 554
 操作和监视, 557
 OP_A_RJC, 561
 I/O, 561
 描述, 558
 操作和监视, 561
 OP_D, 565
 I/O, 565
 描述, 562
 操作和监视, 565
 OP_D3, 571
 I/O, 570
 描述, 566
 操作和监视, 571

OP_TRIG, 574
 I/O, 574
 描述, 572
 操作和监视, 574

P

P_RCV_RK, 522
 描述, 522
 P_SND_RK, 522
 描述, 522
 PA_AI, 456
 I/O, 456
 描述, 452
 PA_AO, 459, 463
 I/O, 463
 描述, 459
 PA_DI, 467, 471
 I/O, 471
 描述, 467
 PA_DO, 473, 477
 I/O, 477
 描述, 473
 PA_TOT, 481, 485
 I/O, 485
 描述, 481
 PID 调谐器
 控制器调谐, 41
 POLYG_P
 I/O, 280
 描述, 279
 PT1_P
 I/O, 283
 描述, 281

- R**
- R_TO_DW, 544
 - I/O, 544
 - 描述, 544
 - RAMP_P
 - I/O, 286
 - 描述, 284
 - RATIO_P, 289, 290
 - I/O, 289
 - VSTATUS, 290
 - 描述, 287
 - 操作和监视, 290
 - RATIO_P 所有视图, 640
 - RCV_341, 488, 493
 - I/O, 493
 - 描述, 488
 - READ355P, 291, 293
 - I/O, 293
 - 描述, 291
 - REC_BO, 23, 26
 - I/O, 26
 - 描述, 23
 - REC_R, 27, 29
 - I/O, 29
 - 描述, 27
- S**
- SEND_BO, 31, 34
 - I/O, 34
 - 描述, 31
 - SEND_R, 35, 38
 - I/O, 38
 - 描述, 35
 - SIG_SMTH
 - I/O, 298
 - 功能, 296
 - 启动特征, 294
 - 安装在 OB 中, 294
 - 对象名, 294
 - 寻址, 294
 - 工作原理, 294
 - 应用领域, 294
 - 时间响应, 294
 - 消息响应, 295
 - 激活和禁止异常检测, 296
 - 运行模式, 295
 - 重启低通滤波器, 296
 - 错误号概述, 297
 - SM 模块的 MODE 设置, 709
 - SM 模块的 OMODE 设置, 719
 - SND_341, 496, 501
 - I/O, 501
 - 描述, 496
 - SPLITR_P, 302
 - I/O, 302
 - 描述, 300
 - ST_MUX
 - I/O, 518
 - 描述, 517
 - STATERP, 519, 520
 - I/O, 520
 - 描述, 519
 - SWIT_CNT, 305, 306, 307
 - I/O, 305
 - VSTATUS, 307
 - 描述, 303
 - 操作和监视, 307
 - 消息文本和关联值, 306
 - SWIT_CNT 所有视图, 643

V

VAL_MOT, 317, 318
 I/O, 314
 VSTATUS, 318
 描述, 308
 操作和监视, 318
 消息文本和关联值, 317

VAL_MOT 所有视图, 645

VALVE, 326, 327
 I/O, 323
 VSTATUS, 327
 描述, 319
 操作和监视, 327
 消息文本和关联值, 326

VALVE 所有视图, 648

VSTATUS
 CTRL_PID, 86
 CTRL_S, 116
 DIG_MON, 130
 DOSE, 142
 ELAP_CNT, 148
 FMCS_PID, 174
 FMT_PID, 202
 INTERLOK, 217
 MEAS_MON, 226
 MOT_REV, 236
 MOT_SPED, 247
 MOTOR, 257
 RATIO_P, 290
 SWIT_CNT, 307
 VAL_MOT, 318
 VALVE, 327

X

XE * MERGEFORMAT, 175, 452, 732

—

一阶滞后元件, 281

从

从模块中读取数据, 188

使

使用 BRCV (REC_BO) 接收 128 个 BOOL 值, 23
 使用 BRCV 接收 32 个 BOOLEAN 值和 32 个 REAL 值, 27
 使用 BSEND (SEND_BO) 发送 128 个 BOOL 值, 31
 使用 BSEND 可发送 32 个 BOOL 值和 32 个 REAL 值 (受变化驱使), 35
 使用 FMT_PID, 通过过程映像来采集和写入过程值, 180

停

停滞时间元件, 117

全

全局视图, 656, 658, 659
 批生产视图, 658
 消息视图, 656
 趋势视图, 659

关

关于使用驱动程序块的注意事项, 329
 关于块描述的常规信息, 17

具

具有一个控制信号的电机, 248

内

内部设置设定值

MPC, 266

冗

冗余组件 [资产] 的状态显示, 735

出

出错处理

FMCS_PID, 162

切

切换周期计数器, 303

功

功能

CPM, 48

GAIN_SHD, 206

MPC, 265

SIG_SMTH, 296

参

参数, 187

下载, 187

双

双向电机, 227

双速电机, 237

启

启动特征

CPM, 45

GAIN_SHD, 205

MPC, 260

SIG_SMTH, 294

图

图形评估和长期统计

CPM, 55

在

在资产中集成 FF 设备, 727

块

块图标, 681, 685, 688, 689, 691, 692, 693, 694, 696, 697, 698, 699, 700, 701, 702, 705, 706, 707

CTRL_PID, 687

CTRL_S, 685

DIG_MON, 688

DOSE, 689

ELAP_CNT, 691

FMCS_PID, 692

FMT_PID, 693

INTERLOK, 694

MEAS_MON, 694

MOT_REV, 696

MOT_SPED, 697

MOTOR, 698

OB_BEGIN, 661

OP_A, 705

OP_A_LIM, 705

OP_A_RJC, 705

OP_D, 706

OP_D3, 706
 OP_TRIG, 707
 RATIO_P, 699
 SWIT_CNT, 700
 VAL_MOT, 700
 VALVE, 701
 在 PH 中更新, 681
 常规属性, 681
 资产管理, 702
 块图标的常规属性, 681

基

基于模型的干扰变量补偿
 MPC, 270

安

安全模式, 187
 安全模式 FMCS_PID, 161
 安装在 OB 中
 CPM, 43
 GAIN_SHD, 204
 MPC, 260
 SIG_SMTH, 294

定

定量给料过程, 131

对

对 FM 355 控制器通道进行寻址, 152
 对控制器通道进行寻址, 178
 对象名
 CPM, 42
 GAIN_SHD, 203
 MPC, 258

NOISE_GN, 278

SIG_SMTH, 294

对象总览, 654

寻

寻址

CPM, 43

GAIN_SHD, 204

MPC, 260

SIG_SMTH, 294

将

将参数下载到模块, 187

工

工作原理

CPM, 43

GAIN_SHD, 203

MPC, 259

SIG_SMTH, 294

平

平均时间值, 530

应

应用领域

MPC, 258

归

归档, 728

形

形成设定值 FMCS_PID, 156
形成误差信号 FMCS_PID, 156
形成调节变量 FMCS_PID, 156
形成限制值 FMCS_PID, 156

微

微分, 120

手

手动、自动和跟踪模式
 FMCS_PID, 158
 FMT_PID, 184
手动模式
 CTRL_PID, 72
 CTRL_S, 97
 FMCS_PID, 158
 FMT_PID, 184
手动模式下的设定值跟踪
 MPC, 267

技

技术规范
 标准库的块, 720

抗

抗饱和
 MPC, 269

报

报告
 ConPerMon, 60
报警延迟
 CPM, 57

控

控制器参数的手动设置
 GAIN_SHD, 206
控制器调谐
 PID 调谐器, 41
控制误差生成和死区
 MPC, 268

描

描述, 23, 27, 31, 35, 117, 209, 213, 221, 291, 340, 345,
353, 374, 378, 382, 389, 398, 405, 411, 452, 459, 467,
473, 481, 488, 496, 519, 529, 544, 576
 ADD4_P, 527
 ADD8_P, 529
 ASSETMON, 505
 AVER_P, 530
 CH_AI, 331
 CH_AO, 340
 CH_CNT, 345
 CH_CNT1, 353
 CH_DI, 374
 CH_DO, 378
 CH_MS, 382
 CH_U_AI, 389
 CH_U_AO, 398
 CH_U_DI, 405
 CH_U_DO, 411
 COUNT_P, 533
 CTRL_PID, 65
 CTRL_S, 87
 DEADT_P, 117
 DIF_P, 120
 DIG_MON, 123
 DOSE, 131
 ELAP_CNT, 143

FF_A_AI, 416
 FF_A_AO, 423
 FF_A_DI, 432
 FF_A_DO, 438
 FMCS_PID, 149
 FMT_PID, 175
 INT_P, 209
 INTERLOK, 213
 LIMITS_P, 218
 MEANTM_P, 536
 MEAS_MON, 221
 MESSAGE, 579
 MODB_341, 504
 MOT_REV, 227
 MOT_SPED, 237
 MOTOR, 248
 MS_MUX, 515
 MSG_NACK, 576
 MSG_TS, 447
 MUL4_P, 539
 MUL8_P, 541
 OB1_TIME, 523
 OP_A, 551
 OP_A_LIM, 554
 OP_A_RJC, 558
 OP_D, 562
 OP_D3, 566
 OP_TRIG, 572
 P_RCV_RK, 522
 P_SND_RK, 522
 PA_AI, 452
 PA_AO, 459
 PA_DI, 467
 PA_DO, 473
 PA_TOT, 481
 POLYG_P, 279
 PT1_P, 281
 R_TO_DW, 544
 RAMP_P, 284
 RATIO_P, 287
 READ355P, 291
 REC_BO, 23
 REC_R, 27
 SEND_BO, 31
 SEND_R, 35
 SND_341, 496
 SPLITR_P, 300
 ST_MUX, 517
 STATEREP, 519
 SWIT_CNT, 303
 VAL_MOT, 308
 VALVE, 319

操

操作员控制块

概述, 545

操作员控制块概述, 545

操作员监控, 202, 217, 226, 257, 557, 561, 565, 571

CTRL_PID, 86

CTRL_S, 116

ELAP_CNT, 148

FMCS_PID, 174

MEAS_MON, 226

MOTOR, 257

OP_A, 553

OP_D, 565

OP_D3, 571

OP_TRIG, 574

VALVE, 327

操作和监视, 142, 148, 236, 247, 290, 307, 318, 327, 514, 553, 574

ASSETMON, 514
DIG_MON, 130
DOSE, 142
FMT_PID, 202
INTERLOK, 217
MOT_REV, 236
MOT_SPED, 247
OP_A_LIM, 557
OP_A_RJC, 561
RATIO_P, 290
SWIT_CNT, 307
VAL_MOT, 318
操纵变量跟踪 FMT_PID, 186

故

故障排除
 CPM, 59
故障诊断
 MPC, 274

数

数字值的操作员输入（1 个按钮）, 572
数字值的操作员输入（2 个按钮）, 562
数字值的操作员输入（3 个按钮）, 566
数字值监视, 123
数据
 读, 188

文

文本库, 729
 ASSETMON, 729

斜

斜坡生成, 284

方

方形或非方系统控制
 MPC, 271
方框图
 CTRL_PID, 78

时

时间响应
 CPM, 45
 GAIN_SHD, 205
 MPC, 260
 SIG_SMTH, 294

显

显示质量代码, 730

最

最多 4 个值的加法, 527
最多 8 个顶点的多边形, 279
最多处理 4 个值的乘法运算, 539
最多处理 8 个值的乘法运算, 541
最多处理 8 个值的加法器, 529

标

标识视图 [资产], 665

模

模式
 CPM, 48
 MPC, 263
模式更改
 CTRL_PID, 74
 CTRL_S, 100

- FMCS_PID, 160
- FMT_PID, 186
- 模拟值的操作员控制, 551
- 模拟值的操作员控制 (拒绝), 558
- 模拟值的操作员控制 (限制), 554

- 比**
- 比率控制, 287

- 测**
- 测量值监视, 221

- 消**
- 消息响应
 - ConPerMon, 60
 - GAIN_SHD, 205
 - MPC, 260
 - SIG_SMTH, 295
- 消息块, 575
 - 概述, 575
- 消息块 (可组态消息), 576, 579
- 消息块概述, 575
- 消息文本
 - ConPerMon, 60
 - MSG_TS, 451
- 消息文本和关联值, 140, 147, 200, 225, 235, 246, 256, 306, 317, 326, 582
 - CTRL_PID, 84
 - CTRL_S, 114
 - DIG_MON, 128
 - DOSE, 140
 - ELAP_CNT, 147
 - FMCS_PID, 172
 - FMT_PID, 200
 - MEAS_MON, 225
 - MESSAGE, 582
 - MOT_REV, 235
 - MOT_SPED, 246
 - MOTOR, 256
 - SWIT_CNT, 306
 - VAL_MOT, 317
 - VALVE, 326
- 消息类别
 - ConPerMon, 60
- 消息视图 [资产], 662

- 激**
- 激活和禁止异常检测
 - SIG_SMTH, 296

- 特**
- 特殊情况串级控制和多变量控制
 - CPM, 56

- 状**
- 状态字分配
 - CPM, 46
 - MPC, 261
- 状态显示锁, 213

- 生**
- 生成和限制调节变量
 - MPC, 266

- 用**
- 用于确定参考方差的供选方案
 - CPM, 56

电

电机阀门控制, 308

监

监视控制质量的确定特征

CPM, 51

监视控制质量的随机特征

CPM, 49

短

短期归档, 728

积

积分, 209

突

突出显示报警循环和通过过程标签选择画面的块图
标, 684

线

线性和非线性系统控制

MPC, 271

组

组态

CPM, 43

GAIN_SHD, 204

MPC, 260

组态工具, 188

维

维护视图 [资产], 663

自

自动模式

CTRL_PID, 72

CTRL_S, 97

FMCS_PID, 158

FMT_PID, 184

计

计数器, 533

计算平均时间值, 536

计算质量代码

CPM, 57

设

设定值滤波器

MPC, 267

设定值的生成, 182

设定值跟踪 FMT_PID, 186

误

误差信号的生成, 182

调

调节值跟踪

MPC, 266

调节变量的生成, 182

质

质量代码显示, 730

质量代码的显示和输出

MPC, 273

资

资产面板的全局表示和视图, 671

跟

跟踪模式

- CTRL_PID, 72
- CTRL_S, 97
- FMCS_PID, 158
- FMT_PID, 184

转

转换块, 543

运

运行时间计数器, 143

运行模式

- SIG_SMTH, 295

通

通过带有 FMCS_PID 的过程映像采集和写入过程值, 154

错

错误号概述

- MPC, 274
- SIG_SMTH, 297

错误处理

- CTRL_PID, 76
- CTRL_S, 103
- FMT_PID, 190

阀

阀控制, 319

限

限制, 218

限制生成, 182

面

面板, 584, 588, 590, 593, 596, 599, 600, 603, 604, 606, 608, 610, 617, 618, 621, 624, 625, 626, 627, 629, 630, 632, 633, 635, 637, 639, 640, 642, 643, 645, 647, 648, 650, 651, 676, 677, 678, 679

ASSETMON, 651

CTRL_PID: 参数视图, 588

CTRL_PID: 标准视图, 584

CTRL_PID: 维护视图, 586

CTRL_PID: 限制视图, 590

CTRL_S: StandardS 视图, 596

CTRL_S: 参数视图, 596

CTRL_S: 维护视图, 596

CTRL_S: 限制视图, 596

CTRL_S: 标准视图, 593

DIG_MON: 标准视图, 599

DOSE: 限制视图, 606

DOSE: 参数视图, 604

DOSE: 标准视图, 600

DOSE: 维护视图, 603

ELAP_CNT: 标准视图, 608

FMCS_PID: 参数视图, 615

FMCS_PID: 维护视图, 613

FMCS_PID: StandardS 视图, 618

FMCS_PID: 标准视图, 610

FMCS_PID: 限制视图, 617

FMT_PID: StandardS 视图, 625

FMT_PID: 参数视图, 624

FMT_PID: 标准视图, 621

FMT_PID: 维护视图, 624

FMT_PID: 限制视图, 624

INTERLOK: 标准视图, 626
MEAS_MON: 标准视图, 627
MEAS_MON: 限制视图, 629
MOT_REV: 标准视图, 630
MOT_REV: 维护视图, 632
MOT_SPED: 标准视图, 633
MOT_SPED: 维护视图, 635
MOTOR: 标准视图, 637
MOTOR: 维护视图, 639
OP_A: 标准视图, 676
OP_A_LIM: 标准视图, 676
OP_A_RJC: 标准视图, 677
OP_D: 标准视图, 677
OP_D3: 标准视图, 678
OP_TRIG: 标准视图, 679

RATIO_P: 限制视图, 642
RATIO_P: 标准视图, 640
SWIT_CNT: 标准视图, 643
VAL_MOT: 标准视图, 645
VAL_MOT: 维护视图, 647
VALVE: 标准视图, 648
VALVE: 维护视图, 650
面板总览行, 654
面板的位置, 683

预

预测控制器算法
MPC, 269