

SIEMENS

SIMATIC HMI

WinCC V7.4

WinCC: 通信

系统手册




在线帮助的打印

过程通讯	1
Allen Bradley - Ethernet IP	2
Mitsubishi 以太网	3
Modbus TCP/IP	4
OPC 通道	5
OPC - 开放式互连	6
OPC UA WinCC Channel	7
PROFIBUS FMS	8
S5 Ethernet Layer 4	9
S5 PROFIBUS FDL	10
S5 Programmers Port AS511	11
S5 Serial 3964R	12
SIMATIC S7 Protocol Suite	13
SIMATIC S7-1200, S7-1500 Channel	14
SIMATIC TI Ethernet Layer 4	15
SIMATIC TI Serial	16
SIMOTION	17
System Info	18
通讯 - 诊断	19

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
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。
注意
表示如果不采取相应的小心措施，可能导致财产损失。


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

合格的专业人员

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

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。本印刷品中的其他符号可能是一些其他商标。若第三方出于自身目的使用这些商标，将侵害其所有者的权利。

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	过程通讯	15
1.1	通讯基本信息	15
1.2	组态连接的基本规则	16
1.3	WinCC 过程通讯	17
1.3.1	WinCC 过程通讯	17
1.3.2	WinCC 通讯的原理	17
1.3.3	外部变量	19
1.3.3.1	外部变量	19
1.3.3.2	如何创建新连接	22
1.3.3.3	外部变量的组态方法	22
1.3.3.4	按 WinCC 数据类型排序的格式改编	23
1.3.3.5	按 AS 数据类型排序的格式改编	31
1.3.3.6	二进制写入机制的原理	43
1.3.3.7	如何通过“二进制写入”组态变量	44
1.3.4	通过以太网连接的端口地址	46
2	Allen Bradley - Ethernet IP	49
2.1	WinCC 通道“Allen Bradley - Ethernet IP”	49
2.2	通道单元分配	50
2.3	支持的数据类型	51
2.4	组态通道	52
2.4.1	组态通道“Allen Bradley - Ethernet IP”	52
2.4.2	如何组态“Allen Bradley E/IP PLC5”通道单元连接	53
2.4.3	如何组态“Allen Bradley E/IP SLC50x”通道单元连接	54
2.4.4	如何组态“Allen Bradley E/IP ControlLogix”通道单元连接	55
2.4.5	实例：通讯路径	56
2.4.6	组态变量	56
2.4.6.1	组态变量	56
2.4.6.2	寻址	58
2.4.6.3	寻址语法	59
2.4.6.4	寻址类型	60
2.4.6.5	寻址实例	61
2.4.6.6	如何组态用于 Allen Bradley E/IP ControlLogix 通道单元的变量	62
2.4.6.7	如何组态用于 Allen Bradley E/IP PLC5 或 SLC50x 的按位访问变量	63
2.4.6.8	如何组态用于 Allen Bradley E/IP PLC5 或 SLC50x 的按字节访问变量	64
2.4.6.9	如何组态用于 Allen Bradley E/IP PLC5 或 SLC50x 的按字访问变量	65
2.4.6.10	如何组态用于 Allen Bradley E/IP PLC5 or SLC50x 的文本变量	66

3	Mitsubishi 以太网.....	67
3.1	WinCC 通道“Mitsubishi 以太网”	67
3.2	所支持的数据类型.....	68
3.3	组态通道.....	69
3.3.1	组态“Mitsubishi 以太网” 通道.....	69
3.3.2	如何组态“Mitsubishi FX3U 系列” 通道单元连接.....	70
3.3.3	如何组态“Mitsubishi Q 系列” 通道单元连接.....	71
3.3.4	组态变量.....	72
3.3.4.1	组态变量.....	72
3.3.4.2	如何组态变量.....	74
4	Modbus TCP/IP.....	77
4.1	“Modbus TCP/IP” 通道.....	77
4.2	支持的数据类型.....	78
4.3	组态通道.....	79
4.3.1	组态“Modbus TCP/IP” 通道.....	79
4.3.2	如何组态连接.....	80
4.3.3	组态变量.....	81
4.3.3.1	组态变量.....	81
4.3.3.2	如何组态按位访问的变量.....	83
4.3.3.3	如何组态按字访问的变量.....	84
4.3.3.4	如何组态文本变量.....	85
5	OPC 通道.....	87
5.1	WinCC OPC 通道.....	87
5.2	OPC 条目管理器.....	89
5.3	所支持的 WinCC 数据类型概述.....	93
5.4	WinCC OPC DA 客户机.....	94
5.4.1	WinCC OPC DA 客户机的功能.....	94
5.4.2	如何通过 OPC 条目管理器访问 WinCC 变量.....	95
5.4.2.1	如何通过 OPC 条目管理器访问 WinCC 变量.....	95
5.4.2.2	在 WinCC OPC DA 客户机上组态 OPC 通道.....	96
5.4.2.3	通过 OPC 条目管理器组态访问.....	97
5.4.3	在不使用 OPC 条目管理器的情况下访问 WinCC 变量.....	101
5.4.4	在 WinCC OPC DA 客户机上使用结构.....	103
5.4.4.1	如何在 WinCC OPC DA 客户机上使用结构.....	103
5.4.4.2	在 WinCC OPC DA 服务器上组态结构和结构变量.....	104
5.4.4.3	如何在 WinCC OPC DA 客户机上组态结构.....	105
5.4.5	OPC DA 通讯受到干扰时的错误处理.....	107
5.4.5.1	OPC 通讯受到干扰时的错误处理.....	107
5.4.5.2	WinCC 作为 OPC DA 服务器.....	108

5.4.5.3	WinCC 作为 OPC DA 客户机.....	110
5.5	WinCC OPC XML 客户机.....	112
5.5.1	WinCC OPC XML 客户机的功能.....	112
5.5.2	如何通过 OPC 条目管理器访问 WinCC 变量.....	113
5.5.2.1	如何通过 OPC 条目管理器访问 WinCC 变量.....	113
5.5.2.2	通过 OPC 条目管理器组态访问.....	114
5.5.3	在不使用 OPC 条目管理器的情况下访问 WinCC 变量.....	118
6	OPC - 开放式互连.....	121
6.1	OPC - 开放式互连.....	121
6.2	OPC 的功能.....	122
6.3	OPC 规范及兼容性.....	123
6.4	在 WinCC 中使用 OPC.....	124
6.5	如何组态 Windows 以使用 WinCC OPC.....	126
6.6	WinCC OPC XML DA 服务器.....	127
6.6.1	操作模式.....	127
6.6.2	安装.....	128
6.6.3	使用 IIS 进行安全设置.....	130
6.6.4	测试安装.....	130
6.7	WinCC OPC DA 服务器.....	132
6.7.1	WinCC OPC DA 服务端的功能.....	132
6.7.2	使用多个 OPC DA 服务器.....	133
6.7.3	查询 OPC DA 服务器名称.....	134
6.7.4	OPC DA 连接的实例.....	136
6.7.4.1	WinCC - WinCC 连接.....	136
6.7.4.2	WinCC - SIMATIC NET FMS OPC 服务器的连接.....	140
6.7.4.3	WinCC - SIMATIC NET S7-OPC 服务器的连接.....	142
6.7.4.4	WinCC - Microsoft Excel 的连接.....	148
6.8	WinCC OPC HDA 服务器.....	153
6.8.1	WinCC OPC HDA 服务器的功能.....	153
6.8.2	WinCC OPC HDA 服务器的数据结构.....	154
6.8.2.1	WinCC OPC HDA 服务器的数据结构.....	154
6.8.2.2	受支持属性的概述.....	155
6.8.2.3	受支持配件的概述.....	156
6.8.2.4	受支持函数的概述.....	158
6.8.2.5	WinCC OPC HDA 服务器的时间格式.....	158
6.8.3	质量代码.....	160
6.8.4	支持的写访问.....	161
6.8.5	OPC HDA 连接实例.....	163
6.8.5.1	OPC HDA 连接实例.....	163
6.8.5.2	HDA 服务器浏览器.....	165

6.8.5.3	如何使用 HDA 服务器浏览器来组态对 WinCC 归档变量的访问.....	166
6.8.5.4	读取 WinCC 归档变量的数值.....	167
6.8.6	WinCC 中 OPC HDA 服务器针对非周期性记录的特殊功能.....	169
6.9	WinCC OPC A&E 服务器.....	172
6.9.1	WinCC OPC A&E 服务器的功能.....	172
6.9.2	在 OPC A&E 上映射 WinCC 消息系统.....	173
6.9.2.1	在 OPC A&E 上映射 WinCC 消息系统.....	173
6.9.2.2	映射 WinCC 消息类别和消息类型.....	175
6.9.2.3	映射 WinCC 消息优先级.....	175
6.9.2.4	WinCC 消息系统的属性.....	176
6.9.2.5	确认方法.....	178
6.9.3	OPC A&E 质量代码.....	181
6.9.4	OPC A&E 连接实例.....	181
6.9.4.1	OPC A&E 连接实例.....	181
6.9.4.2	如何组态访问 WinCC 消息系统.....	182
6.9.5	带层级访问的 OPC A&E 服务器.....	184
6.9.5.1	OPC A&E 服务器的功能.....	184
6.9.5.2	OPC A&E 与带层级访问的 OPC A&E 之间的差异.....	186
6.9.5.3	在 OPC A&E 上映射 WinCC 消息系统.....	187
6.9.5.4	OPC A&E 质量代码.....	195
6.9.6	读取归档消息.....	196
6.9.6.1	访问归档消息.....	196
6.9.6.2	使用 OPC 访问归档消息的语法.....	197
6.9.6.3	归档消息的读方法.....	198
6.9.6.4	识别归档消息.....	200
6.10	WinCC OPC UA 服务器.....	201
6.10.1	WinCC OPC UA 服务器的工作原理.....	201
6.10.2	OPC UA 的安全概念.....	202
6.10.3	组态安全机制.....	206
6.10.4	支持的 OPC UA 服务和配置文件.....	209
6.10.5	WinCC OPC UA 服务器的名称区域.....	211
6.10.6	OPC UA 数据访问.....	214
6.10.7	OPC UA 日志访问.....	214
6.10.8	OPC UA 报警和条件.....	215
6.10.9	WinCC 消息系统的属性.....	220
6.10.10	组态 WinCC OPC UA 服务器.....	222
6.10.10.1	组态文件.....	222
6.10.10.2	如何组态 OPC UA 服务器.....	223
6.11	跟踪.....	226
7	OPC UA WinCC Channel.....	227
7.1	WinCC 通道“OPC UA WinCC 通道”.....	227
7.2	所支持的数据类型概述.....	229

7.3	WinCC OPC UA Configurator.....	230
7.4	WinCC OPC UA Configurator 中的符号.....	238
7.5	WinCC OPC UA Configurator 的设置.....	241
7.6	“控制台”输出窗口.....	242
7.7	组态 OPC UA 通道.....	243
7.7.1	常规序列.....	243
7.7.2	在 WinCC 项目中插入 OPC UA WinCC 通道.....	244
7.7.3	创建与 OPC UA 服务器的连接.....	244
7.7.4	设置通过证书进行认证。.....	246
7.7.5	OPC UA 变量.....	248
7.7.6	如何组态到 OPC UA 服务器的连接.....	251
7.7.7	导入 OPC UA 节点作为 WinCC 变量.....	254
7.8	通信受到干扰时的误差处理.....	256
8	PROFIBUS FMS.....	257
8.1	WinCC 通道“PROFIBUS FMS”.....	257
8.2	变量的数据类型.....	258
8.3	组态通道.....	259
8.3.1	组态连接.....	259
8.3.1.1	组态连接.....	259
8.3.1.2	在组态模式中组态连接.....	259
8.3.1.3	如何在运行系统中组态连接.....	262
8.3.2	组态变量.....	263
8.3.2.1	组态变量.....	263
8.3.2.2	在组态模式中组态变量地址.....	264
8.3.2.3	如何在运行期间组态变量地址.....	266
9	S5 Ethernet Layer 4.....	269
9.1	WinCC 通道“SIMATIC S5 Ethernet Layer 4”.....	269
9.2	变量的数据类型.....	271
9.3	组态通道.....	272
9.3.1	组态通道“SIMATIC S5 Ethernet Layer 4”.....	272
9.3.2	如何组态连接.....	272
9.3.3	组态变量.....	275
9.3.3.1	组态变量.....	275
9.3.3.2	变量的地址.....	275
9.3.3.3	如何组态按位访问的变量.....	277
9.3.3.4	如何组态按字节访问的变量.....	278
9.3.3.5	如何组态按字访问的变量.....	280
9.3.3.6	如何组态原始数据变量.....	281
9.3.4	系统参数.....	283

9.3.4.1	通道单元的系统参数.....	283
9.3.4.2	如何更改设备名称.....	284
9.3.4.3	如何更改传输参数.....	285
9.4	附录.....	287
9.4.1	附录.....	287
9.4.2	内部错误代码和常量.....	287
9.4.2.1	内部错误代码和常量.....	287
9.4.2.2	连接受干扰时的错误代码.....	287
9.4.2.3	iNA960 消息.....	291
9.4.2.4	SCI 消息.....	292
10	S5 PROFIBUS FDL.....	295
10.1	WinCC 通道“SIMATIC S5 Profibus FDL”.....	295
10.2	受支持的数据类型和数据范围.....	297
10.3	WinCC 通道“SIMATIC S5 Profibus FDL” 的功能.....	299
10.4	组态通道.....	301
10.4.1	如何组态通道“SIMATIC S5 Profibus FDL”.....	301
10.4.2	通道单元“FDL (CP5412/A2-1)”.....	301
10.4.3	如何组态连接.....	302
10.4.4	组态变量.....	304
10.4.4.1	组态变量.....	304
10.4.4.2	如何组态按位访问的变量.....	304
10.4.4.3	如何组态按字节访问的变量.....	305
10.4.4.4	如何组态按字访问的变量.....	306
10.4.5	系统参数.....	307
10.4.5.1	通道单元的系统参数.....	307
10.4.5.2	如何更改设备名称.....	308
10.4.5.3	如何更改过程值的写/读监视时间.....	309
10.5	特殊功能.....	310
10.5.1	“SIMATIC S5 Profibus FDL” 通道的特殊功能.....	310
10.5.2	“SIMATIC S5 Profibus FDL” 通道的原始数据变量.....	310
10.5.2.1	“SIMATIC S5 Profibus FDL” 通道的原始数据变量.....	310
10.5.2.2	如何组态原始数据变量.....	311
10.5.3	组态通讯类型.....	312
10.5.3.1	组态通讯类型.....	312
10.5.3.2	如何组态主动数据传送.....	312
10.5.3.3	如何组态被动数据传送.....	314
10.6	“SIMATIC S5 Profibus FDL” 通道的组态实例.....	316
10.6.1	“SIMATIC S5 Profibus FDL” 通道的组态实例.....	316
10.6.2	如何组态 AS 中的数据处理块.....	316
10.6.3	如何组态 I/O 域.....	318

11	S5 Programmers Port AS511	321
11.1	WinCC 通道“SIMATIC S5 Programmers Port AS511”.....	321
11.2	变量的数据类型.....	322
11.3	组态通道.....	323
11.3.1	组态“SIMATIC S5 Programmers Port AS511” 通道.....	323
11.3.2	如何组态连接.....	323
11.3.3	组态变量.....	325
11.3.3.1	组态变量.....	325
11.3.3.2	如何组态变量的地址.....	325
11.3.3.3	如何组态按位访问的变量.....	328
11.3.3.4	如何组态按字节访问的变量.....	329
12	S5 Serial 3964R	331
12.1	WinCC 通道“SIMATIC S5 Serial 3964R”.....	331
12.2	变量的数据类型.....	332
12.3	组态通道.....	333
12.3.1	组态“SIMATIC S5 Serial 3964R” 通道.....	333
12.3.2	如何组态连接.....	333
12.3.3	组态变量.....	335
12.3.3.1	组态变量.....	335
12.3.3.2	如何组态变量的地址.....	335
12.3.3.3	如何组态按位访问的变量.....	337
12.3.3.4	如何组态按字节访问的变量.....	338
13	SIMATIC S7 Protocol Suite	341
13.1	“SIMATIC S7 Protocol Suite” 通道.....	341
13.2	WinCC 通道“SIMATIC S7 Protocol Suite”.....	342
13.3	通道单元选择.....	344
13.4	所支持的数据类型概述.....	348
13.5	组态通道.....	349
13.5.1	“SIMATIC S7 Protocol Suite” 通道 - 组态.....	349
13.5.2	如何组态“SIMATIC S7 Protocol Suite” 通道.....	349
13.5.3	通道单元.....	350
13.5.3.1	“SIMATIC S7 Protocol Suite” 通道的通道单元.....	350
13.5.3.2	“工业以太网 (I+II)” 通道单元.....	351
13.5.3.3	“MPI” 通道单元.....	354
13.5.3.4	“命名连接” 通道单元.....	356
13.5.3.5	“PROFIBUS (I+II)” 通道单元.....	359
13.5.3.6	“Slot PLC” 通道单元.....	361
13.5.3.7	“Soft PLC” 通道单元.....	363
13.5.3.8	“TCP/IP” 通道单元.....	364

13.5.4	组态变量.....	367
13.5.4.1	组态变量.....	367
13.5.4.2	如何组态按位访问的变量.....	367
13.5.4.3	如何组态按字节访问的变量.....	368
13.5.4.4	如何组态按字访问的变量.....	370
13.5.4.5	如何组态文本变量.....	371
13.5.5	系统参数.....	372
13.5.5.1	通道单元的系统参数.....	372
13.5.5.2	PLC 中的周期性读取服务.....	373
13.5.5.3	如何组态系统参数.....	374
13.5.5.4	如何更改逻辑设备名称.....	376
13.6	特殊功能.....	378
13.6.1	“SIMATIC S7 Protocol Suite” 通道的特殊功能.....	378
13.6.2	使用 S7 功能块 AR_SEND 进行数据交换.....	378
13.6.2.1	使用 S7 功能块 AR_SEND 进行数据交换.....	378
13.6.2.2	数据块 - 结构和参数.....	380
13.6.2.3	AR_SEND 变量的属性概述.....	385
13.6.2.4	用于一个归档变量的 AR_SEND 变量.....	387
13.6.2.5	数据块结构实例 1: 一个归档变量; 每个过程值具有一个时间标志.....	389
13.6.2.6	数据块结构实例 2: 一个归档变量; 等间隔的时间标志.....	389
13.6.2.7	数据块结构实例 3: 一个归档变量; 每个过程值具有自己的时间标志.....	390
13.6.2.8	数据块结构实例 4: 一个归档变量; 每个过程值具有相对时间标志 (时差).....	391
13.6.2.9	用于多个归档变量的 AR_SEND 变量.....	392
13.6.2.10	数据块结构实例 5: 多个归档变量; 每个过程值具有自己的时间标志.....	396
13.6.2.11	数据块结构实例 6: 多个归档变量; 等间隔的时间标志.....	398
13.6.2.12	数据块结构实例 7: 多个归档变量; 每个过程值具有自己的时间标志.....	400
13.6.2.13	数据块结构实例 8: 多个归档变量; 过程值具有相对时间标志 (时差).....	402
13.6.2.14	用于多个归档变量的 AR_SEND 变量 (已优化).....	404
13.6.2.15	数据块结构实例 9: 多个归档变量; 已优化.....	405
13.6.2.16	如何组态用于归档变量的 AR_SEND 变量.....	406
13.6.2.17	如何组态用于多个归档变量的 AR_SEND 变量.....	408
13.6.3	通道“SIMATIC S7 Protocol Suite” 的原始数据变量.....	410
13.6.3.1	通道“SIMATIC S7 Protocol Suite” 的原始数据变量.....	410
13.6.3.2	原始数据变量作为字节数组.....	411
13.6.3.3	如何将原始数据变量组态为字节数组.....	412
13.6.3.4	用于 S7 通信 BSEND/BRCV 功能的原始数据变量.....	414
13.6.3.5	如何组态“BSEND/BRCV” 功能的原始数据变量.....	417
13.6.4	软件冗余.....	418
13.6.4.1	软件冗余.....	418
13.6.4.2	软件冗余 - 连接专用的内部变量.....	420
13.6.4.3	如何组态软件冗余.....	424
13.6.4.4	如何在 WinCC 中清除软件冗余.....	426
13.6.4.5	如何检查 WinCC 启动参数.....	427
13.6.4.6	如何将 WinCC 系统消息载入报警记录.....	427

13.6.4.7	当连接受干扰时的错误代码.....	427
14	SIMATIC S7-1200, S7-1500 Channel.....	429
14.1	“SIMATIC S7-1200, S7-1500 Channel” 通道.....	429
14.2	所支持的数据类型概述.....	430
14.3	组态通道.....	431
14.3.1	组态“SIMATIC S7-1200, S7-1500 Channel” 通道.....	431
14.3.2	如何组态连接.....	432
14.3.3	如何组态没有优先块访问的变量.....	433
14.3.4	如何组态具有优先块访问的变量.....	434
15	SIMATIC TI Ethernet Layer 4.....	439
15.1	WinCC 通道“SIMATIC TI Ethernet Layer 4”	439
15.2	变量的数据类型.....	440
15.3	组态通道.....	441
15.3.1	组态通道“SIMATIC TI Ethernet Layer 4”	441
15.3.2	如何组态连接.....	441
15.3.3	组态变量.....	443
15.3.3.1	组态变量.....	443
15.3.3.2	如何组态变量的地址.....	443
15.3.3.3	如何组态按位访问变量.....	445
15.3.3.4	如何组态按字节访问的变量.....	446
15.3.4	系统参数.....	447
15.3.4.1	通道单元的系统参数.....	447
15.3.4.2	如何更改设备名称.....	448
15.3.4.3	如何更改传输参数.....	449
16	SIMATIC TI Serial.....	451
16.1	WinCC 通道“SIMATIC TI Serial”	451
16.2	变量的数据类型.....	452
16.3	组态通道.....	453
16.3.1	组态“SIMATIC TI Serial” 通道.....	453
16.3.2	如何组态连接.....	453
16.3.3	组态变量.....	454
16.3.3.1	组态变量.....	454
16.3.3.2	如何组态变量的地址.....	455
16.3.3.3	如何组态按位访问变量.....	456
16.3.3.4	如何组态按字节访问的变量.....	458
17	SIMOTION.....	461
17.1	WinCC 通道“SIMOTION”	461
17.2	所支持的数据类型概述.....	462

17.3	组态通道.....	463
17.3.1	“SIMOTION”通道的组态.....	463
17.3.2	如何导出 SIMOTION SCOUT 项目	463
17.3.3	如何通过 Simotion Mapper 创建 WinCC 项目.....	464
17.3.4	如何通过 Simotion Mapper 更改 WinCC 项目.....	465
17.3.5	如何更改连接参数.....	467
17.3.6	如何更改变量地址.....	468
17.3.7	系统参数组态.....	468
17.3.7.1	通道单元的系统参数.....	468
17.3.7.2	如何组态系统参数.....	469
17.3.7.3	如何更改逻辑设备名称.....	471
17.4	诊断“SIMOTION”通道.....	473
17.4.1	“SIMOTION”通道的诊断可能性.....	473
17.4.2	日志文件条目的描述.....	473
18	System Info.....	477
18.1	“System Info”通道.....	477
18.2	WinCC System Info 通道.....	478
18.3	所支持的系统信息概述.....	480
18.4	与其它软件组件的不同之处.....	485
18.5	组态通道.....	486
18.5.1	如何组态 System Info 通道.....	486
18.6	评价和显示系统信息的实例.....	487
18.6.1	如何调用和评价系统信息.....	487
18.6.2	如何在“System Info”通道中组态变量.....	487
18.6.3	如何在 I/O 域中显示时间.....	489
18.6.4	如何用棒图显示空闲磁盘容量.....	490
18.6.5	如何在趋势窗口中显示 CPU 负载.....	491
18.6.6	如何组态有关空闲磁盘容量的消息.....	492
18.6.7	如何显示有关可用磁盘容量的消息.....	494
18.6.8	如何在状态显示中显示打印机状态.....	496
18.6.9	如何检查 WinCC 启动参数.....	497
18.6.10	如何插入棒图.....	498
18.6.11	如何插入 I/O 域.....	498
18.6.12	如何启动运行系统.....	498
18.7	特殊功能.....	499
18.7.1	在多用户和客户系统中使用.....	499
18.7.1.1	在多用户和客户系统中使用.....	499
18.7.2	监控多台服务器的系统信息的实例.....	499
18.7.2.1	在 WinCC 客户机上监控多台服务器的系统信息.....	499
18.7.2.2	如何组态第一台服务器.....	500

18.7.2.3	如何组态第二台服务器.....	501
18.7.2.4	如何将变量导入 WinCC 客户机.....	502
18.7.2.5	如何在 WinCC 客户机上组态过程画面.....	503
18.7.2.6	如何激活项目.....	505
19	通讯 - 诊断.....	507
19.1	通道和变量的诊断.....	507
19.2	有关错误检测的常规信息.....	508
19.3	通道诊断.....	509
19.3.1	通道诊断.....	509
19.3.2	“状态 - 逻辑连接”功能.....	509
19.3.3	如何使用“状态 - 逻辑连接”功能来检查通道.....	509
19.3.4	使用“通道诊断”进行通道诊断.....	511
19.3.4.1	通道诊断的原理.....	511
19.3.4.2	使用 ActiveX 控件进行通道诊断.....	512
19.3.4.3	如何使用作为 ActiveX 控件的通道诊断来检查通道.....	512
19.3.4.4	使用“通道诊断”来诊断通道.....	513
19.3.4.5	如何使用通道诊断检查通道.....	513
19.3.4.6	如何组态通道的跟踪功能.....	514
19.3.4.7	如何启动运行系统.....	515
19.4	“System Info”通道的诊断.....	517
19.4.1	“System Info”通道 - 诊断选项.....	517
19.4.2	日志文件条目的描述.....	517
19.4.3	确定错误变量值的原因.....	518
19.4.3.1	如何确定错误变量的原因.....	518
19.4.3.2	如何检查通道和连接.....	519
19.4.3.3	如何检查变量.....	521
19.5	诊断通道“SIMATIC S7-1200/S7-1500”.....	522
19.5.1	通过系统诊断控件进行系统诊断.....	522
19.5.2	如何组态系统诊断.....	526
19.6	“SIMATIC S7 Protocol Suite”通道的诊断.....	528
19.6.1	“SIMATIC S7 Protocol Suite”通道 - 诊断选项.....	528
19.6.2	日志文件条目的描述.....	529
19.6.3	确定错误变量值的原因.....	531
19.6.3.1	如何确定错误变量的原因.....	531
19.6.3.2	如何检查通信处理器的组态.....	532
19.6.3.3	检查 SIMATIC NET 中的通讯处理器.....	534
19.6.3.4	如何检查连接和变量的组态.....	535
19.6.3.5	如何检查通道和连接.....	536
19.6.3.6	如何检查变量.....	538
19.7	“SIMATIC S5 Profibus FDL”通道的诊断.....	540
19.7.1	“SIMATIC S5 PROFIBUS FDL”通道的诊断选项.....	540

19.7.2	日志文件条目的描述.....	541
19.7.3	确定错误变量值的原因.....	541
19.7.3.1	如何确定错误变量的原因.....	541
19.7.3.2	如何检查通讯处理器的组态.....	542
19.7.3.3	检查 SIMATIC NET 中的通讯处理器.....	544
19.7.3.4	如何检查连接和变量的组态.....	545
19.7.3.5	如何检查通道和连接.....	546
19.7.3.6	如何检查变量.....	549
19.8	“OPC” 通道的诊断.....	550
19.8.1	诊断“OPC” 通道的可能性.....	550
19.8.2	日志文件条目的描述.....	550
19.8.2.1	日志文件条目的描述.....	550
19.8.2.2	“INFO” 标记条目.....	552
19.8.2.3	“ERROR” 标记条目.....	553
19.8.3	确定错误变量值的原因.....	555
19.8.3.1	如何确定无效变量的原因.....	555
19.8.3.2	如何检查组态数据.....	555
19.8.3.3	如何检查通道和连接.....	556
19.8.3.4	如何检查变量.....	558
19.9	变量质量.....	560
19.9.1	变量质量.....	560
19.9.2	变量的质量代码.....	562
19.9.3	变量状态.....	568
19.9.4	使用变量状态监视连接状态.....	570
19.9.5	使用全局动作监视变量状态.....	572
19.9.6	如何检查内部变量.....	573
索引.....		575

过程通讯

1.1 通讯基本信息

简介

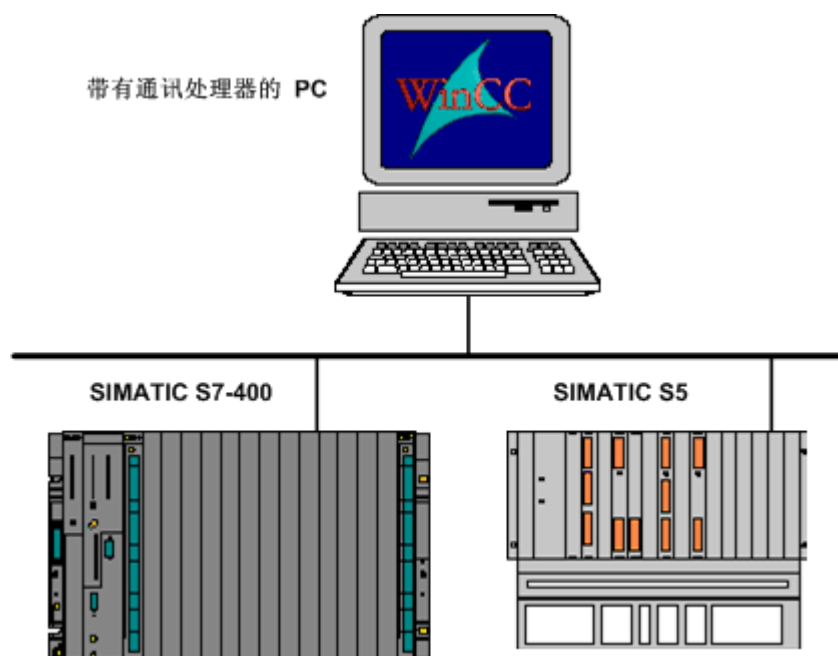
通讯是指在两个通讯伙伴之间进行数据交换。

通讯

通讯伙伴可以是用于与其它网络组件进行通讯并交换数据的任何网络组件。在 WinCC 中，通讯伙伴可以是自动化系统 (AS) 中的中央模块和通讯模块，也可以是 PC 中的通讯处理器。

通讯伙伴间传送的数据可以用于不同用途。在 WinCC 中，通讯有如下用途：

- 控制过程
- 调用过程数据
- 指示过程中的异常状态
- 归档过程数据



1.2 组态连接的基本规则

采集周期和更新时间

组态软件中定义的变量采集周期是决定可实现的更新时间的主要因素。

更新时间是采集周期、传输时间和处理时间之和。

要获得最佳的更新时间，在组态期间请记住以下几点：

- 优化数据区域的最大和最小尺寸。
- 将同属的数据区定义为一个整体。如果建立一个大数据区而非多个小数据区，则可改善更新时间。
- 采集周期过小会降低性能。应根据过程值的变化速率设置采集周期。以烤炉的温度变化为例，它的变化速度要比电气传动装置慢得多。
- 应将报警或画面的变量不留间隙地放在一个数据区中。
- 控制器中的变化至少要持续一个采集周期才能被可靠地检测到。
- 要实现无错误传输，应将传输率设置为尽可能高的值。

画面

画面的刷新频率取决于要显示的数据类型和数据量。

要缩短更新时间，务必为需要快速更新的对象组态更短的采集时间。

曲线

使用位触发的曲线时，如果在“曲线传送区域”中设置了组位，则在 WinCC 站中会更新在该区域中设置了组位的所有曲线。这些位会在下一个周期复位。

只有所有位在 WinCC 站中都复位完毕后，才能在 PLC 程序中再次设置组位。

1.3 WinCC 过程通讯

1.3.1 WinCC 过程通讯

简介

您可以从 WinCC 访问自动化系统中的过程变量（外部变量）。但是，在 WinCC 中组态过程链接之前，应使用检查单检查是否已满足以下先决条件：

- 自动化系统必须配备 WinCC 中的通讯驱动程序所支持的通讯接口。
- 必须在自动化系统中适当组态此接口，使控制器程序能够通过通讯调用访问该接口。通讯硬件的组态参数必须已知。
- WinCC 应访问的变量的地址必须已知。请注意，这些地址取决于自动化系统。
- WinCC 系统中必须已安装相应通讯硬件（通讯处理器、标准 I/O 端口 COMx、...）。为了安装通讯硬件，还必须先安装随附的操作系统驱动程序（硬件驱动程序）。通讯处理器的硬件和软件设置必须已知。
- 视 WinCC 系统中所使用的通讯处理器而定，可能还需要进行更多设置。例如，使用工业以太网或 PROFIBUS 时，必须建立本地数据库。此连接参数也必须已知。

为了在运行期间进行操作，WinCC 和 AS 之间还必须存在物理连接，以便访问外部变量。

1.3.2 WinCC 通讯的原理

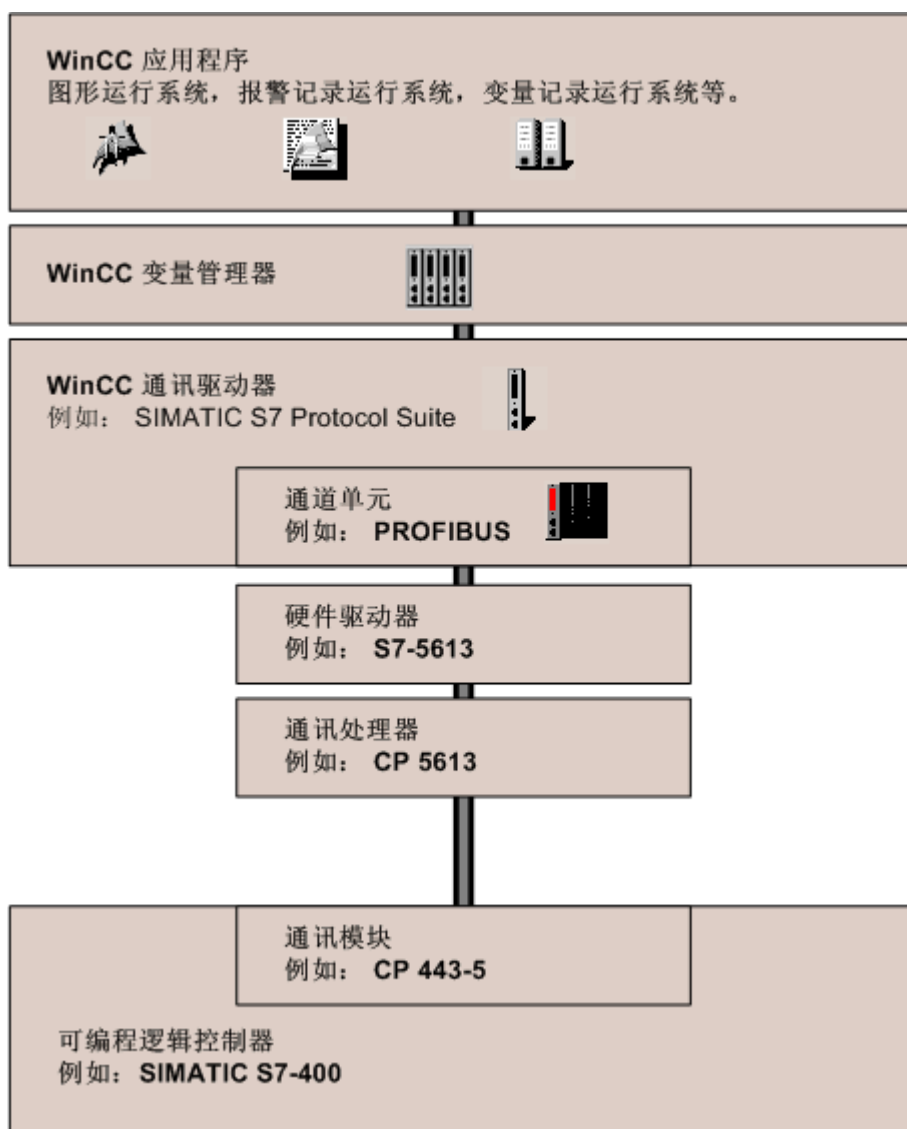
简介

WinCC 使用称为变量管理的功能集中管理其变量。WinCC 在运行期间会采集和管理在项目中创建的以及在项目数据库中存储的所有数据和变量。

图形运行系统、报警记录运行系统或变量记录运行系统等所有应用程序（全局脚本）必须请求来自变量管理的 WinCC 变量数据。

WinCC 和自动化系统 (AS) 之间的通讯

与 WinCC 进行工业通讯也就是通过变量和过程值交换信息。为了采集过程值，WinCC 通讯驱动程序向 AS 发送请求报文。而 AS 则在相应的响应报文中将所请求的过程值发送回 WinCC。



因此，WinCC 和 AS 之间必须存在物理连接。该连接的属性（如，传送介质和通讯网络）定义了通讯的状态，组态 WinCC 中的通讯时需要这些属性。

通讯驱动程序

通讯驱动程序是用于在 AS 和 WinCC 的变量管理之间建立连接的软件组件，这样可以提供 WinCC 变量和过程值。在 WinCC 中，提供了许多用于通过不同总线系统连接各个 AS 的通讯驱动程序。

每个通讯驱动程序一次只能绑定到一个 WinCC 项目。

WinCC 中的通讯驱动程序也称为“通道”，其文件扩展名为“*.chn”。计算机中安装的所有通讯驱动程序都位于 WinCC 安装目录的子目录“\bin”中。

一个通讯驱动程序针对不同通讯网络会有不同的通道单元。

通道单元

每个通道单元相当于与一个基础硬件驱动程序的接口，进而也相当于与 PC 中的一个通讯处理器的接口。因此，每个使用的通道单元必须分配到各自的通讯处理器。

对于某些通道单元，会在系统参数中进行额外的组态。对于使用 OSI 模型传输层（第 4 层）的通道单元，还将定义传输参数。

连接（逻辑）

对 WinCC 和 AS 进行了正确的物理连接后，WinCC 中需要通讯驱动程序和相应的通道单元来创建和组态与 AS 的（逻辑）连接。运行期间将通过此连接进行数据交换。

在 WinCC 中，已组态且已逻辑分配的两个通讯伙伴之间会有一个用于执行某种通讯服务连接。每个连接都有两个包含必要信息的端点，这些信息包括用来对通讯伙伴寻址的必要信息以及用来建立该连接的其它属性。

连接通过特定连接参数在通道单元下组态。一个通道单元下也可以创建多个连接，这取决于通讯驱动程序。

1.3.3 外部变量

1.3.3.1 外部变量

简介

获得 AS 中的某些数据需要 WinCC 变量。这些会影响与 AS 连接的变量称为外部变量。其它不包含过程连接的变量称为内部变量。

数据类型和类型转换

组态外部变量时，除变量名外还必须定义数据类型，其中某些数据类型还需要进行类型转换：

在 WinCC 中，数据类型决定了数据格式。从 AS 格式到 WinCC 格式的转换随类型转换一起定义。这种类型转换对两个传送方向都适用：

- 在 AS 中：例如，针对某些功能（如定时器值/BCD 显示）或要寻址的信息（如在数据块或 I/O 区中的字节地址和字地址）。
- 在 WinCC 中：例如，针对模拟值处理或计算。

实际上，通常情况下会定义 AS 数据格式。下面几种情况可以选择 WinCC 格式：

- WinCC 数据格式可以与 AS 格式相匹配。要实现这种模式，可选择这样的类型转换：它在两侧使用相同的类型并且无论 WinCC 数据类型为何都考虑前导符号（如，“字转换为有符号字”）。如果这通过所选数据类型无法实现，则必须在 WinCC 中进行更改。
- WinCC 格式将根据 WinCC 中处理的值进行设置。

有必要选择数据类型和类型转换时，请注意以下重要注意事项：

- 前导符号：是否应考虑调整前导符号？操作期间变量值是否会出现负值？（例如，控制差异的百分比）
- 值范围：操作期间出现的变量值是否在两种格式的值范围内，WinCC 或 AS 中是否可能出现值溢出？如果出现溢出，另一侧的值就不能显示，否则会导致后续处理出现故障。
- 不同的类型转换具有相同的值范围：一种数据类型的不同类型转换可能具有相同的值范围。如，“字节转换为无符号双字”和“字节转换为无符号字”的值范围均为 [0...127]。在这种情况下，请检查在 AS 中为数据定义的格式，检查此格式是否由于范围过大而不必要地浪费资源。（如，采用双字代替字）。

如果所选的类型转换未涵盖 AS 中需要的值范围，则必须在 WinCC 中更改数据类型。

说明

过程变量的组态不正确（例如，由于地址错误）时，与自动化系统的通讯可能会中断。

WinCC 数据类型和类型转换

下表显示了支持类型转换的 WinCC 数据类型。

数据类型	类型转换
二进制变量	不支持
无符号 8 位数	支持
有符号 8 位数	支持
无符号 16 位数	支持
有符号 16 位数	支持
无符号 32 位数	支持
有符号 32 位数	支持
浮点数 32 位 IEEE 754	支持
浮点数 64 位 IEEE 754	支持
文本变量 8 位字符集	不支持
文本变量 16 位字符集	不支持
原始数据类型	不支持

说明

AS 所发送的数据并不是通过类型转换解释的，而是在选定的格式改编内由 WinCC 来解释的。如果 WinCC 无法解释数据，则“..\Siemens\WinCC\Diagnose”目录下的“WinCC_sys_0x.log”文件中将出现一个故障条目。

数值变量类型的线性标定

可以对数值数据类型执行线性标定。过程中现有变量的值范围可以在 WinCC 变量的特定值范围内进行线性表示。

例如，过程可以要求以单位 [巴] 来定义一个设定值，但在 WinCC 中用户应以单位 [毫巴] 来定义该值。通过线性转换，过程中的数值范围 [0...1] 可以转换为 WinCC 变量的数值范围 [0...1000]。

文本变量的长度定义

“文本变量 8 位字符集”和“文本变量 16 位字符集”数据类型的变量需要长度定义。对于以后要容纳 10 个字符的文本变量，若是“8 位字符集”，必须是 10 个字符长；若是“16 位字符集”，则必须有 20 个字符长。

在自动化系统中寻址

在 AS 中，WinCC 变量被分配到某个数据范围。必须在 AS 中以某种方式对它们进行编址。寻址类型取决于通讯伙伴的类型。

1.3.3.2 如何创建新连接

简介

只能基于到 AS 的连接创建外部变量。如果不存在所需连接，则必须先进行创建。

要求

- 已安装了所需的通信处理器和相应的硬件驱动程序。
- 并已安装了要求的通信驱动程序，如，“SIMATIC S7 Protocol Suite”。

步骤

1. 在 Configuration Studio 的导航栏中选择“变量管理”(Tag Management)。
2. 在导航区域中选择所需的通道单元，例如“PROFIBUS”。
3. 在通道单元的弹出式菜单中选择“新建连接...”条目。
4. 在数据区域中为连接指定唯一名称。
5. 在“属性”(Properties) 窗口中为该连接定义所需参数。更多相关信息，可参见相关通道的帮助/文档。

1.3.3.3 外部变量的组态方法

简介

对于大多数数据类型，变量的创建步骤是相似的。

但是，某些数据类型则需要特殊设置（步骤 5 - 7）。

要求

- 已安装了所需的通信处理器和硬件驱动程序。
- 已安装了要求的通信驱动程序，如，“SIMATIC S7 Protocol Suite”。
- 已根据通道单元创建连接（例如“PROFIBUS”）。

步骤

1. 在导航区域的树形视图中，选择要为其创建变量的连接。
2. 在“名称”(Name) 列最上方的空单元格中输入在 WinCC 项目中唯一的变量名，例如：“WinCCTag_01”。
3. 在“数据类型”(Data type) 字段中定义变量的数据类型，如“浮点数 64 位 IEEE 754”。
4. 在 AS 的“AS 长度”(AS Length) 字段中指定变量的地址区域。
使用不支持对二进制或 8 位变量的位/字节访问的通道时，将首先显示“位/字节变量”(Bit/Byte Tag) 对话框，然后再显示“变量属性”(Tag properties) 对话框。
有关详细信息，请参阅“二进制写入机制的原理”(Principle of the BinWrite-Mechanism)。
单击“确定”(OK) 按钮关闭“位/字节变量”(Bit/Byte Tag) 对话框或“变量属性”(Tag properties) 对话框。
5. 使用数字变量时，WinCC 会在“格式改编”(Format adaptation) 域中提供一个建议的格式改编。
需要时，可选择另一种格式改编。显示依次为“X to Y”，其中 X = WinCC 格式，Y = AS 格式，如“DoubleToDouble”。
6. 激活“线性标定”(Linear scaling) 复选框对数字变量进行线性转换。输入“过程值范围”（在 AS 中）和“变量值范围”（在 WinCC 中）的上限和下限。
7. 将激活文本变量的“长度”域。在此输入文本变量的长度（以字节为单位）。
8. 使用“确定”按钮关闭所有对话框。

1.3.3.4 按 WinCC 数据类型排序的格式改编

简介

在组态外部变量时，必须针对所有数字数据类型进行另外的格式改编。

数据类型决定了 WinCC 一侧的数据格式。格式改编还定义从 WinCC 格式到 AS 格式的转换。该定义对两个传送方向均适用。

从以下选择框中选择所需要的 WinCC 数据类型。下表列出相应可能的格式改编和值范围。

WinCC 数据类型

表格 1-1 有符号 8 位数

格式改编“有符号 8 位值”	值范围
CharToUnsignedByte	0...127
CharToUnsignedWord	0...127
CharToUnsignedDword	0...127
CharToSignedByte	-128...+127 (无转换)
CharToSignedWord	-128...+127
CharToSignedDword	-128...+127
CharToMSBByte	-128...+127
CharToMSBWord	-128...+127
CharToMSBDword	-128...+127
CharToBCDByte	0...99
CharToBCDWord	0...127
CharToBCDDword	0...127
CharToSignedBCDByte	-9...+9
CharToSignedBCDWord	-128...+127
CharToSignedBCDDword	-128...+127
CharToExtSignedBCDByte	-79...+79
CharToExtSignedBCDWord	-128...+127
CharToExtSignedBCDDword	-128...+127
CharToAikenByte	0...99
CharToAikenWord	0...127
CharToAikenDword	0...127
CharToSignedAikenByte	-9...+9
CharToSignedAikenWord	-128...+127
CharToSignedAikenDword	-128...+127
CharToExcessByte	0...99
CharToExcessWord	0...127
CharToExcessDword	0...127
CharToSignedExcessByte	-9...+9

格式改编“有符号 8 位值”	值范围
CharToSignedExcessWord	-128...+127
CharToSignedExcessDword	-128...+127

表格 1-2 无符号 8 位数

格式改编“无符号 8 位值”	值范围
ByteToUnsignedByte	0...255 (无转换)
ByteToUnsignedWord	0...255
ByteToUnsignedDword	0...255
ByteToSignedByte	0...127
ByteToSignedWord	0...255
ByteToSignedDword	0...255
ByteToBCDByte	0...99
ByteToBCDWord	0...255
ByteToBCDDword	0...255
ByteToAikenByte	0...99
ByteToAikenWord	0...255
ByteToAikenDword	0...255
ByteToExcessByte	0...99
ByteToExcessWord	0...255
ByteToExcessDword	0...255

表格 1-3 有符号 16 位数

格式改编“有符号 16 位值”	值范围
ShortToUnsignedByte	0...255
ShortToUnsignedWord	0...32767
ShortToUnsignedDword	0...32767
ShortToSignedByte	-128...+127
ShortToSignedWord	-32768...+32767 (无转换)
ShortToSignedDword	-32768...+32767
ShortToMSBByte	-127...+127

格式改编“有符号 16 位值”	值范围
ShortToMSBWord	-32767...+32767
ShortToMSBWord	-32768...+32767
ShortToBCDByte	0...99
ShortToBCDWord	0...9999
ShortToBCDDword	0...32767
ShortToSignedBCDByte	-9...+9
ShortToSignedBCDWord	-999...+999
ShortToSignedBCDDword	-32768...+32767
ShortToExtSignedBCDByte	-79...+79
ShortToExtSignedBCDWord	-7999...+7999
ShortToExtSignedBCDDword	-32768...+32767
ShortToAikenByte	0...99
ShortToAikenWord	0...9999
ShortToAikenDword	0...32767
ShortToSignedAikenByte	-9...+9
ShortToSignedAikenWord	-999...+999
ShortToSignedAikenDword	-32768...+32767
ShortToExcessByte	0...99
ShortToExcessWord	0...9999
ShortToExcessDword	0...32767
ShortToSignedExcessByte	-9...+9
ShortToSignedExcessWord	-999...+999
ShortToSignedExcessDword	-32768...+32767

表格 1-4 无符号 16 位数

格式改编“无符号 16 位值”	值范围
WordToUnsignedWord	0...65535 (无转换)
WordToUnsignedByte	0...255
WordToUnsignedDword	0...65535
WordToSignedByte	0...127
WordToSignedWord	0...32767

格式改编“无符号 16 位值”	值范围
WordToSignedDword	0...65535
WordToBCDByte	0...99
WordToBCDWord	0...9999
WordToBCDDword	0...65535
WordToAikenByte	0...99
WordToAikenWord	0...9999
WordToAikenDword	0...65535
WordToExcessByte	0...99
WordToExcessWord	0...9999
WordToExcessDword	0...65535
WordToSimaticCounter	0...999
WordToSimaticBCDCounter	0...999

表格 1-5 有符号 32 位数

格式改编“有符号 32 位值”	值范围
LongToSignedDword	-2147483647...+2147483647 (无转换)
LongToUnsignedByte	0...255
LongToUnsignedWord	0...65535
LongToUnsignedDword	0...2147483647
LongToSignedByte	-128...+127
LongToSignedWord	-32768...+32767
LongToMSBByte	-127...+127
LongToMSBWord	-32767...+32767
LongToMSBDword	-2147483647...+2147483647
LongToBCDByte	0...99
LongToBCDWord	0...9999
LongToBCDDword	0...99999999
LongToSignedBCDByte	-9...+9
LongToSignedBCDWord	-999...+999
LongToSignedBCDDword	-9999999...+9999999
LongToExtSignedBCDByte	-79...+79

1.3 WinCC 过程通讯

格式改编“有符号 32 位值”	值范围
LongToExtSignedBCDWord	-7999...+7999
LongToExtSignedBCDDword	-79999999...+79999999
LongToAikenByte	0...99
LongToAikenWord	0...9999
LongToAikenDword	0...99999999
LongToSignedAikenByte	-9...+9
LongToSignedAikenWord	-999...+999
LongToSignedAikenDword	-9999999...+9999999
LongToExcessByte	0...99
LongToExcessWord	0...9999
LongToExcessDword	0...99999999
LongToSignedExcessByte	-9...+9
LongToSignedExcessWord	-999...+999
LongToSignedExcessDword	-9999999...+9999999
LongToSimaticTimer	10...9990000
LongToSimaticBCDTimer	10...9990000

表格 1-6 无符号 32 位数

格式改编“无符号 32 位值”	值范围
DwordToUnsignedDword	0...4294967295（无转换）
DwordToUnsignedByte	0...255
DwordToUnsignedWord	0...65535
DwordToSignedByte	0...127
DwordToSignedWord	0...32767
DwordToSignedDword	0...2147483647
DwordToBCDByte	0...99
DwordToBCDWord	0...9999
DwordToBCDDword	0...99999999
DwordToAikenByte	0...99
DwordToAikenWord	0...9999
DwordToAikenDword	0...99999999

格式改编“无符号 32 位值”	值范围
DwordToExcessByte	0...99
DwordToExcessWord	0...9999
DwordToExcessDword	0...99999999
DwordToSimaticTimer	10...9990000
DwordToSimaticBCDTimer	10...9990000

表格 1-7 浮点数 32 位 IEEE 754

类型改编“浮点数 32 位 IEEE 754”	值范围
FloatToFloat	+ -3.402823e+38 (无转换)
FloatToUnsignedByte	0...255
FloatToUnsignedWord	0...65535
FloatToUnsignedDword	0 至 4.294967e+09
FloatToSignedByte	-128...+127
FloatToSignedWord	-32768...+32767
FloatToSignedDword	-2.147483e+09 至 +2.147483e+09
FloatToDouble	+ -3.402823e+38
FloatToMSBByte	-127...+127
FloatToMSBWord	-32767...+32767
FloatToMSBDword	-2.147483e+09 至 +2.147483e+09
FloatToBCDByte	0...99
FloatToBCDWord	0...9999
FloatToBCDDword	0 至 9.999999e+07
FloatToSignedBCDByte	-9...+9
FloatToSignedBCDWord	-999...+999
FloatToSignedBCDDword	-9999999...+9999999
FloatToExtSignedBCDByte	-79...+79
FloatToExtSignedBCDWord	-7999...+7999
FloatToExtSignedBCDDword	-7.999999e+07 至 +7.999999e+07
FloatToAikenByte	0...99
FloatToAikenWord	0...9999
FloatToAikenDword	0 至 9.999999e+07

1.3 WinCC 过程通讯

类型改编“浮点数 32 位 IEEE 754”	值范围
FloatToSignedAikenByte	-9...+9
FloatToSignedAikenWord	-999...+999
FloatToSignedAikenDword	-9999999...+9999999
FloatToExcessByte	0...99
FloatToExcessWord	0...9999
FloatToExcessDword	0 至 9.999999e+07
FloatToSignedExcessByte	-9...+9
FloatToSignedExcessWord	-999...+999
FloatToSignedExcessDword	-9999999...+9999999
FloatToS5Timer	10...9990000
FloatToS5Float	+ -1.701411e+38

表格 1-8 浮点数 64 位 IEEE 754

类型改编“浮点数 64 位 IEEE 754”	值范围
DoubleToDouble	+ -1.79769313486231e+308 (无转换)
DoubleToUnsignedByte	0...255
DoubleToUnsignedWord	0...65535
DoubleToUnsignedDword	0...4294967295
DoubleToSignedByte	-128...+127
DoubleToSignedWord	-32768...+32767
DoubleToSignedDword	-2147483647...+2147483647
DoubleToFloat	+ -3.402823e+38
DoubleToMSBByte	-127...+127
DoubleToMSBWord	-32767...+32767
DoubleToMSBDword	-2147483647...+2147483647
DoubleToBCDByte	0...99
DoubleToBCDWord	0...9999
DoubleToBCDDword	0...99999999
DoubleToSignedBCDByte	-9...+9
DoubleToSignedBCDWord	-999...+999
DoubleToSignedBCDDword	-9999999...+9999999

类型改编 “浮点数 64 位 IEEE 754”	值范围
DoubleToExtSignedBCDByte	-79...+79
DoubleToExtSignedBCDWord	-7999...+7999
DoubleToExtSignedBCDDword	-79999999...+79999999
DoubleToAikenByte	0...99
DoubleToAikenWord	0...9999
DoubleToAikenDword	0...99999999
DoubleToSignedAikenByte	-9...+9
DoubleToSignedAikenWord	-999...+999
DoubleToSignedAikenDword	-99999999...+99999999
DoubleToExcessByte	0...99
DoubleToExcessWord	0...9999
DoubleToExcessDword	0...99999999
DoubleToSignedExcessByte	-9...+9
DoubleToSignedExcessWord	-999...+999
DoubleToSignedExcessDword	-99999999...+99999999
DoubleToS5Timer	10...9990000
DoubleToS5Float	+/-1.701411e+38

1.3.3.5 按 AS 数据类型排序的格式改编

简介

在组态外部变量时，必须将所有数字数据类型的数据改编为另一种格式。

在 WinCC 端，数据类型决定数据格式。格式改编还定义从 WinCC 格式到 AS 格式的转换。该定义对两个传送方向均适用。

从下面的选择框中选择需要的 AS 数据类型。下表将列出相应的可能格式改编及其对应值范围。

AS 数据类型

格式改编和值范围

表格 1-9 AikenByte

格式改编“AikenByte”	值范围
ByteToAikenByte	0...99
CharToAikenByte	0...99
DoubleToAikenByte	0...99
DwordToAikenByte	0...99
FloatToAikenByte	0...99
LongToAikenByte	0...99
ShortToAikenByte	0...99
WordToAikenByte	0...99

表格 1-10 AikenWord

格式改编“AikenWord”	值范围
ByteToAikenWord	0...255
CharToAikenWord	0...127
DoubleToAikenWord	0...9999
DwordToAikenWord	0...9999
FloatToAikenWord	0...9999
LongToAikenWord	0...9999
ShortToAikenWord	0...9999
WordToAikenWord	0...9999

表格 1-11 AikenDWord

格式改编“AikenDWord”	值范围
ByteToAikenDword	0...255
CharToAikenDword	0...127
DoubleToAikenDword	0...99999999

格式改编“ AikenDWord ”	值范围
DwordToAikenDword	0...999999999
FloatToAikenDword	0 至 9.999999e+07
LongToAikenDword	0...999999999
ShortToAikenDword	0...32767
WordToAikenDword	0...65535

表格 1-12 BCDByte

格式改编“ BCDByte ”	值范围
ByteToBCDByte	0...99
CharToBCDByte	0...99
DoubleToBCDByte	0...99
DwordToBCDByte	0...99
FloatToBCDByte	0...99
LongToBCDByte	0...99
ShortToBCDByte	0...99
WordToBCDByte	0...99

表格 1-13 BCDWord

格式改编“ BCDWord ”	值范围
ByteToBCDWord	0...255
CharToBCDWord	0...127
DoubleToBCDWord	0...9999
DwordToBCDWord	0...9999
FloatToBCDWord	0...9999
LongToBCDWord	0...9999
ShortToBCDWord	0...9999
WordToBCDWord	0...9999

1.3 WinCC 过程通讯

表格 1-14 BCDDWord

格式改编“BCDDWord”	值范围
ByteToBCDDword	0...255
CharToBCDDword	0...127
DoubleToBCDDword	0...99999999
DwordToBCDDword	0...99999999
FloatToBCDDword	0 至 9.999999e+07
LongToBCDDword	0...99999999
ShortToBCDDword	0...32767
WordToBCDDword	0...65535

表格 1-15 Double

格式改编“Double”	值范围
DoubleToDouble	+1.79769313486231e+308 (无转换)
FloatToDouble	+3.402823e+38

表格 1-16 ExcessByte

格式改编“ExcessByte”	值范围
ByteToExcessByte	0...99
CharToExcessByte	0...99
DoubleToExcessByte	0...99
DwordToExcessByte	0...99
FloatToExcessByte	0...99
LongToExcessByte	0...99
ShortToExcessByte	0...99
WordToExcessByte	0...99

表格 1-17 ExcessWord

格式改编“ExcessWord”	值范围
ByteToExcessWord	0...255
CharToExcessWord	0...127
DoubleToExcessWord	0...9999
DwordToExcessWord	0...9999
FloatToExcessWord	0...9999
LongToExcessWord	0...9999
ShortToExcessWord	0...9999
WordToExcessWord	0...9999

表格 1-18 ExcessDWord

格式改编“ExcessDWord”	值范围
ByteToExcessDword	0...255
CharToExcessDword	0...127
DoubleToExcessDword	0...99999999
DwordToExcessDword	0...99999999
FloatToExcessDword	0 至 9.999999e+07
LongToExcessDword	0...99999999
ShortToExcessDword	0...32767
WordToExcessDword	0...65535

表格 1-19 ExtSignedBCDByte

格式改编“ExtSignedBCDByte”	值范围
CharToExtSignedBCDByte	-79...+79
DoubleToExtSignedBCDByte	-79...+79
FloatToExtSignedBCDByte	-79...+79
LongToExtSignedBCDByte	-79..+79
ShortToExtSignedBCDByte	-79...+79

1.3 WinCC 过程通讯

表格 1-20 ExtSignedBCDWord

格式改编“ExtSignedBCDWord”	值范围
CharToExtSignedBCDWord	-128...+127
DoubleToExtSignedBCDWord	-7999...+7999
FloatToExtSignedBCDWord	-7999...+7999
LongToExtSignedBCDWord	-7999...+7999
ShortToExtSignedBCDWord	-7999...+7999

表格 1-21 ExtSignedBCDDWord

格式改编“ExtSignedBCDDWord”	值范围
CharToExtSignedBCDDword	-128...+127
DoubleToExtSignedBCDDword	-79999999...+79999999
FloatToExtSignedBCDDword	-7.9999999e+07 至 +7.9999999e+07
LongToExtSignedBCDDword	-79999999...+79999999
ShortToExtSignedBCDDword	-32768...+32767

表格 1-22 Float

格式改编“Float”	值范围
DoubleToFloat	+/-3.402823e+38
FloatToFloat	+/-3.402823e+38 (无转换)

表格 1-23 MSBByte

格式改编“MSBByte”	值范围
CharToMSBByte	-128...+127
DoubleToMSBByte	-127...+127
FloatToMSBByte	-127...+127
LongToMSBByte	-127...+127
ShortToMSBByte	-127...+127

表格 1-24 MSBWord

格式改编“MSBWord”	值范围
CharToMSBWord	-128...+127
DoubleToMSBWord	-32767...+32767
FloatToMSBWord	-32767...+32767
LongToMSBWord	-32767...+32767
ShortToMSBWord	-32767...+32767

表格 1-25 MSBDWord

格式改编“MSBDWord”	值范围
CharToMSBDword	-128...+127
DoubleToMSBDword	-2147483647...+2147483647
FloatToMSBDword	-2.147483e+09 至 +2.147483e+09
LongToMSBDword	-2147483647...+2147483647
ShortToMSBDword	-32768...+32767

表格 1-26 S5Float

格式改编“S5Float”	值范围
DoubleToS5Float	+1.701411e+38
FloatToS5Float	+1.701411e+38

表格 1-27 S5Timer

格式改编“S5Timer”	值范围
DoubleToS5Timer	10...9990000
FloatToS5Timer	10...9990000

表格 1-28 SignedByte

格式改编“SignedByte”	值范围
ByteToSignedByte	0...127
CharToSignedByte	-128...+127 (无转换)

格式改编“SignedByte”	值范围
DoubleToSignedByte	-128...+127
DwordToSignedByte	0...127
FloatToSignedByte	-128...+127
LongToSignedByte	-128...+127
ShortToSignedByte	-128...+127
WordToSignedByte	0...127

表格 1-29 SignedWord

格式改编“SignedWord”	值范围
ByteToSignedWord	0...255
CharToSignedWord	-128...+127
DoubleToSignedWord	-32768...+32767
DwordToSignedWord	0...32767
FloatToSignedWord	-32768...+32767
LongToSignedWord	-32768...+32767
ShortToSignedWord	-32768...+32767（无转换）
WordToSignedWord	0...32767

表格 1-30 SignedDWord

格式改编“SignedDWord”	值范围
ByteToSignedDword	0...255
CharToSignedDword	-128...+127
DoubleToSignedDword	-2147483647...+2147483647
DwordToSignedDword	0...2147483647
FloatToSignedDword	-2.147483e+09 至 +2.147483e+09
LongToSignedDword	-2147483647...+2147483647（无转换）
ShortToSignedDword	-32768...+32767
WordToSignedDword	0...65535

表格 1-31 SignedAikenByte

格式改编“SignedAikenByte”	值范围
CharToSignedAikenByte	-9...+9
DoubleToSignedAikenByte	-9...+9
FloatToSignedAikenByte	-9...+9
LongToSignedAikenByte	-9...+9
ShortToSignedAikenByte	-9...+9

表格 1-32 SignedAikenWord

格式改编“SignedAikenWord”	值范围
CharToSignedAikenWord	-128...+127
DoubleToSignedAikenWord	-999...+999
FloatToSignedAikenWord	-999...+999
LongToSignedAikenWord	-999...+999
ShortToSignedAikenWord	-999...+999

表格 1-33 SignedAikenDWord

格式改编“SignedAikenDWord”	值范围
CharToSignedAikenDword	-128...+127
DoubleToSignedAikenDword	-9999999...+9999999
FloatToSignedAikenDword	-9999999...+9999999
LongToSignedAikenDword	-9999999...+9999999
ShortToSignedAikenDword	-32768...+32767

表格 1-34 SignedBCDByte

格式改编“SignedBCDByte”	值范围
CharToSignedBCDByte	-9...+9
DoubleToSignedBCDByte	-9...+9
FloatToSignedBCDByte	-9...+9

1.3 WinCC 过程通讯

格式改编“SignedBCDByte”	值范围
LongToSignedBCDByte	-9...+9
ShortToSignedBCDByte	-9...+9

表格 1-35 SignedBCDWord

格式改编“SignedBCDWord”	值范围
CharToSignedBCDWord	-128...+127
DoubleToSignedBCDWord	-999...+999
FloatToSignedBCDWord	-999...+999
LongToSignedBCDWord	-999...+999
ShortToSignedBCDWord	-999...+999

表格 1-36 SignedBCDDWord

格式改编“SignedBCDDWord”	值范围
CharToSignedBCDDword	-128...+127
DoubleToSignedBCDDword	-9999999...+9999999
FloatToSignedBCDDword	-9999999...+9999999
LongToSignedBCDDword	-9999999...+9999999
ShortToSignedBCDDword	-32768...+32767

表格 1-37 SignedExcessByte

格式改编“SignedExcessByte”	值范围
CharToSignedExcessByte	-9...+9
DoubleToSignedExcessByte	-9...+9
FloatToSignedExcessByte	-9...+9
LongToSignedExcessByte	-9...+9
ShortToSignedExcessByte	-9...+9

表格 1-38 SignedExcessWord

格式改编“SignedExcessWord”	值范围
CharToSignedExcessWord	-128...+127
DoubleToSignedExcessWord	-999...+999
FloatToSignedExcessWord	-999...+999
LongToSignedExcessWord	-999...+999
ShortToSignedExcessWord	-999...+999

表格 1-39 SignedExcessDWord

格式改编“SignedExcessDWord”	值范围
CharToSignedExcessDword	-128...+127
DoubleToSignedExcessDword	-9999999...+9999999
FloatToSignedExcessDword	-9999999...+9999999
LongToSignedExcessDword	-9999999...+9999999
ShortToSignedExcessDword	-32768...+32767

表格 1-40 SimaticCounter

格式改编“SimaticCounter”	值范围
WordToSimaticCounter	0...999

表格 1-41 SimaticBCDCounter

格式改编“SimaticBCDCounter”	值范围
WordToSimaticBCDCounter	0...999

表格 1-42 SimaticTimer

格式改编“SimaticTimer”	值范围
DwordToSimaticTimer	10...9990000
LongToSimaticTimer	10...9990000

表格 1-43 SimaticBCDTimer

格式改编“SimaticBCDTimer”	值范围
DwordToSimaticBCDTimer	10...9990000
LongToSimaticBCDTimer	10...9990000

表格 1-44 UnsignedByte

格式改编“UnsignedByte”	值范围
ByteToUnsignedByte	0...255（无转换）
CharToUnsignedByte	0...127
DoubleToUnsignedByte	0...255
DwordToUnsignedByte	0...255
FloatToUnsignedByte	0...255
LongToUnsignedByte	0...255
ShortToUnsignedByte	0...255
WordToUnsignedByte	0...255

表格 1-45 UnsignedWord

格式改编“UnsignedWord”	值范围
ByteToUnsignedWord	0...255
CharToUnsignedWord	0...127
DoubleToUnsignedWord	0...65535
DwordToUnsignedWord	0...65535
FloatToUnsignedWord	0...65535
LongToUnsignedWord	0...65535
ShortToUnsignedWord	0...32767
WordToUnsignedWord	0...65535（无转换）

表格 1-46 UnsignedDWord

格式改编“UnsignedDWord”	值范围
ByteToUnsignedDword	0...255
CharToUnsignedDword	0...127
DoubleToUnsignedDword	0...4294967295
DwordToUnsignedDword	0...4294967295（无转换）
FloatToUnsignedDword	0 至 4.294967e+09
LongToUnsignedDword	0...2147483647
ShortToUnsignedDword	0...32767
WordToUnsignedDword	0...65535

1.3.3.6 二进制写入机制的原理

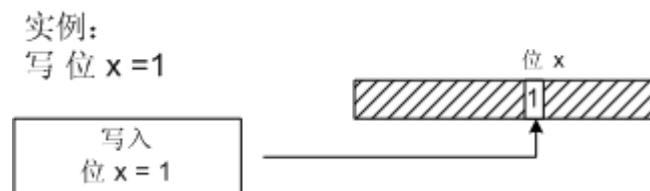
简介

在 WinCC 中，并不是所有的通讯驱动程序和其通道单元都支持直接按位或按字节访问（简称为：位/字节访问）所连接的自动化系统中的地址范围。而是需要借助二进制写入机制来完成。

位/字节访问

使用带位/字节访问的通讯驱动程序的通道单元，可直接读写期望的位或字节。

下图通过直接位/字节访问将值 1 分配给位 x。



二进制写入机制

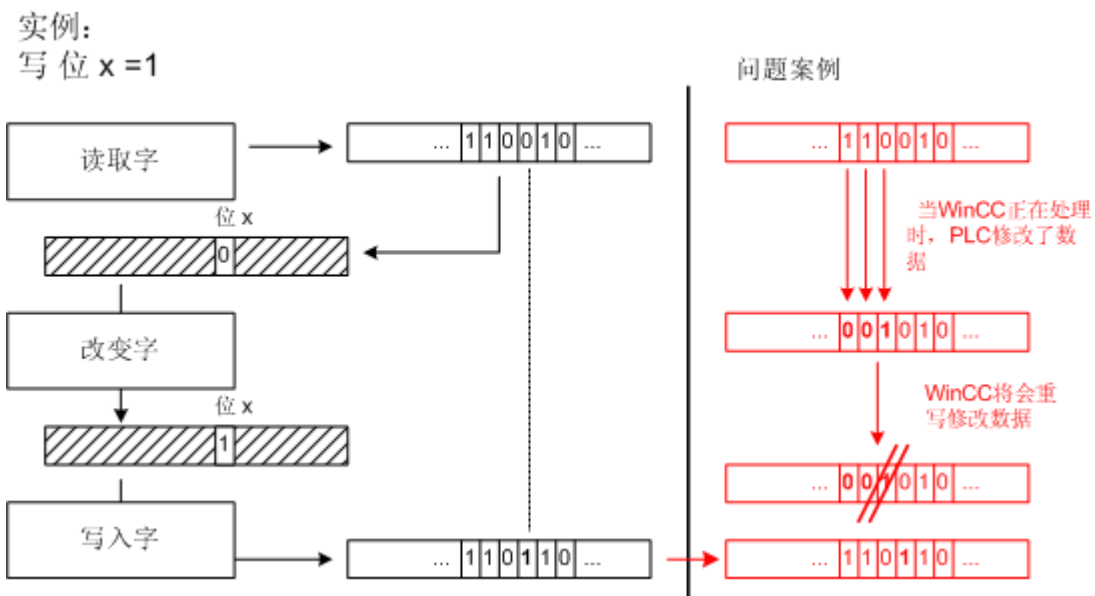
以下通讯驱动程序不支持位/字节访问，而需要对相应的通道单元使用二进制写入机制：

- Modbus Serial
- SIMATIC S5 Ethernet Layer 4

- SIMATIC S5 Programmers Port AS511
- SIMATIC S5 Serial 3964R
- SIMATIC TI Ethernet Layer 4
- SIMATIC TI Serial

为写入位或字节，通道单元首先使用二进制写入机制读取整个数据字。随后在读取的字中更改要寻址的数据。最后再写回整个 (!) 字，而不是仅写回改变的位或字节。

下图通过二进制访问机制将值 1 分配给位 x。



说明

如果在 WinCC 中通过二进制写入机制读取某个数据字的同时该数据字在 AS 中发生更改（请参见图“问题案例”），则当 WinCC 写回该数据字时，AS 中的更改即会丢失。

1.3.3.7 如何通过“二进制写入”组态变量


简介

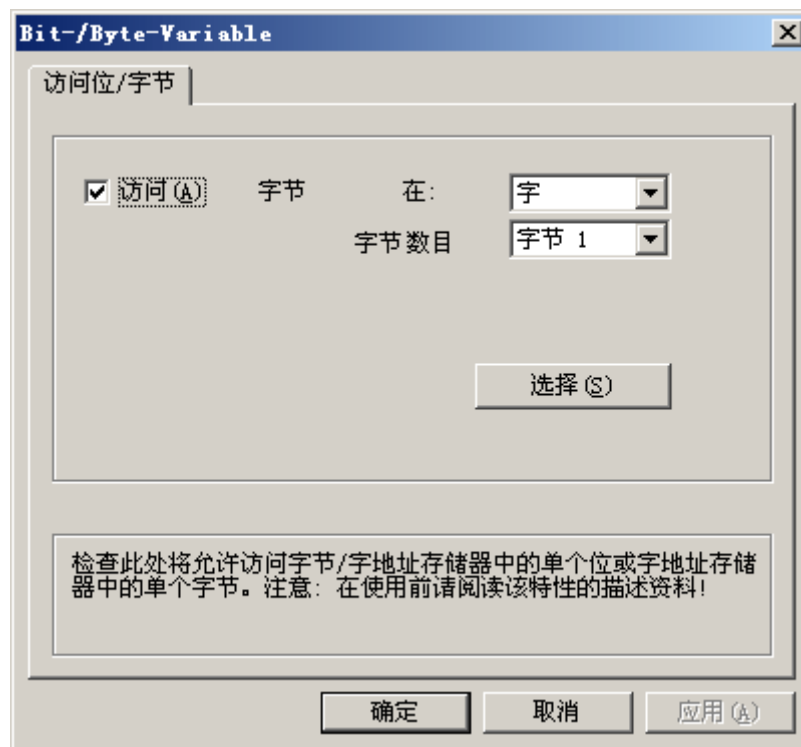
如果想要为不支持位/字节访问的通信驱动程序通道单元组态“二进制变量”，则必须使用对话框激活和组态二进制写入机制，否则该机制不可用。

要求

- 已安装了所需的通信处理器和硬件驱动程序。
- 已安装了所期望的通信驱动程序（但它不支持位/字节访问），例如，“SIMATIC S5 Ethernet Layer 4”。
- 已根据通道单元创建连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 选择“二进制变量”、“无符号 8 位数”或“有符号 8 位数”作为数据类型。
5. 打开“位/字节变量”(Bit/Byte Tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
“位/字节变量”(Bit/Byte Tag) 对话框随即打开。（使用此按钮设置位/字节访问通道在 AS 中的地址范围。）




6. 激活“访问一位”(Access to one bit) 复选框或“访问一个字节”(Access to one byte) 复选框并进行常规设置。该对话框中的显示取决于在第 2 步中选择的 datatype。
7. 单击“确定”(OK) 关闭对话框。

说明

未激活新“二进制变量”或 8 位变量的二进制写入机制时，对话框也会打开。如果要立即使用“确定”再次关闭“创建新变量”窗口，将出现错误消息“地址参数无效”。选择“是”确认对话框。这将打开对话框“位/字节变量”，您可采取这种方式重复设置。

1.3.4 通过以太网连接的端口地址

端口地址

 小心
<p>以太网通讯</p> <p>使用基于以太网的通讯时，最终用户应负责其自身数据网络的安全。例如，如果有针对性的攻击导致设备过载，则功能将无法得到保证。</p>

通过以太网连接时，用户会需要端口地址信息。组态防火墙或路由器需要利用该信息。下表显示了 WinCC 应用程序默认使用的端口地址。

	端口地址 TCP/IP	端口地址 UDP
S7 通讯	102	
HTTP (通讯; 传输)	80	
HTTPS (通讯; 传输)	443	
WebServices (SOAP)	80 HTTP 443 HTTPS	
OPC-XML (CE 作为 OPC 主机)	80 HTTP 443 HTTPS	
SendEmail	25	
传输 (通过以太网; CE-Stub; PC 装载程序; PC)	2308 或者 50523	
记录 (通过以太网) CSV 文件	139, 445	137, 138
Modbus Ethernet	502	

	端口地址 TCP/IP	端口地址 UDP
Allen-Bradley Ethernet CIP	44818	
Allen-Bradley Ethernet CSP2	2222	

Allen Bradley - Ethernet IP

2.1 WinCC 通道“Allen Bradley - Ethernet IP”

简介

通道“Allen Bradley - Ethernet IP”用于链接到 Allen-Bradley 自动化系统。其通讯使用 Ethernet IP 协议处理。

根据所用的通讯硬件，系统支持通过下列通道单元的连接：

- Allen Bradley E/IP PLC5
- Allen Bradley E/IP SLC50x
- Allen Bradley E/IP ControlLogix

2.2 通道单元分配

简介

必须为通道选择通道单元，以创建从 WinCC 到现有或已设计网络的连接。

通道单元分配

下表显示了通道“Allen Bradley - Ethernet IP”的通道单元如何分配到网络和自动化系统 (AS)。

通道的通道单元	通讯网络	AS
Allen Bradley E/IP PLC5	Ethernet IP	带以太网端口的 PLC-5
Allen Bradley E/IP SLC50x	Ethernet IP	带以太网端口的 SLC 500, 如 SLC 5/05
Allen Bradley E/IP ControlLogix	Ethernet IP	ControlLogix 5500

2.3 支持的数据类型

简介

为逻辑连接定义所需变量。“Allen Bradley - Ethernet IP”通道支持以下数据类型：

- 二进制变量
- 有符号 8 位数
- 无符号 8 位数
- 有符号 16 位数
- 无符号 16 位数
- 有符号 32 位数
- 无符号 32 位数
- 浮点数 32 位 IEEE 754
- 文本变量，8 位字符集
- 文本变量，16 位字符集

2.4 组态通道

2.4.1 组态通道“Allen Bradley - Ethernet IP”

简介

在 WinCC 中，WinCC 与自动化系统 (AS) 的通讯需要逻辑连接。本节介绍如何组态“Allen Bradley - Ethernet IP”通道。

使用 TCP/IP 协议时，必须定义 AS 的 IP 地址以用于逻辑连接。IP 地址由四个以圆点分隔的数值组成。数值范围必须在 0-255 之内。

说明

超时行为

使用 TCP/IP 协议时，不会立即检测到中断的连接。回查消息可能耗时达一分钟。

可连接的控制器

可对以下 Allen-Bradley PLC 进行连接：

- Allen-Bradley ControlLogix 5500
- Allen-Bradley CompactLogix 5300
- 带以太网端口的 PLC-5
- 带以太网端口的 SLC 500，如 SLC 5/05
- MicroLogix

已发布的通讯类型

以下通讯类型已经过系统测试并发布供“Allen Bradley - Ethernet IP”使用：

- 点对点连接：
- WinCC 站与可选数量控制器之间的多点连接。

在线组态

不支持“Allen Bradley - Ethernet IP”通道的在线组态。

2.4.2 如何组态“Allen Bradley E/IP PLC5” 通道单元连接

简介

组态“Allen Bradley E/IP PLC5” 通道单元的连接需要执行下列步骤：

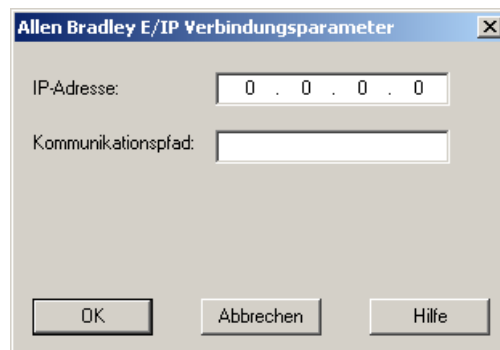
1. 组态连接
2. 组态变量

要求

- 通道“Allen Bradley - Ethernet IP” 的通信驱动程序已安装并集成到项目中。

步骤

1. 在通道单元“Allen Bradley E/IP PLC5” 的快捷菜单中选择条目 “新建连接”(New Connection)。 将创建一个新连接。
2. 选择新连接，然后在 “属性 - 连接”(Properties - Connection) 区域中的 “常规”(General) 下输入连接名。
3. 在快捷菜单中选择 “连接参数”(Connection parameters) 连接。“Allen Bradley E/IP 连接参数”(Allen Bradley E/IP connection parameters) 对话框随即打开。



4. 输入控制器 Ethernet/IP 模块的 IP 地址。默认情况下，始终为以太网 IP 设备设置端口 44818。
5. 在 “通信路径”(Communication path) 字段中定义从以太网模块到控制器的 CIP 路径。这一设置创建了以太网模块与 PLC 间的逻辑连接，即使它们位于不同 CIP 网络中。
6. 单击 “确定”(OK) 按钮，关闭对话框。

2.4 组态通道

2.4.3 如何组态“Allen Bradley E/IP SLC50x” 通道单元连接

简介

组态“Allen Bradley E/IP SLC50x” 通道单元的连接需要执行下列步骤：

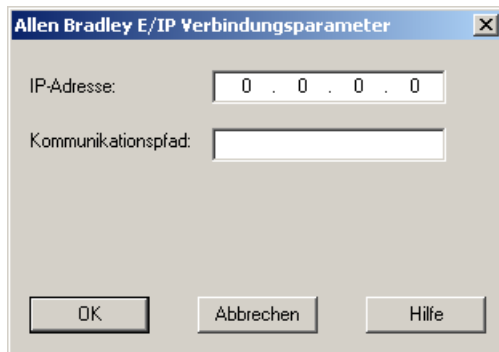
1. 组态连接
2. 组态变量

要求

- 通道“Allen Bradley - Ethernet IP” 的通信驱动程序已安装并集成到项目中。

步骤

1. 在通道单元“Allen Bradley E/IP PLC5” 的快捷菜单中选择条目 “新建连接”(New Connection)。 将创建一个新连接。
2. 选择新连接，然后在 “属性 - 连接”(Properties - Connection) 区域中的 “常规”(General) 下输入连接名。
3. 在快捷菜单中选择 “连接参数”(Connection parameters) 连接。“Allen Bradley E/IP 连接参数”(Allen Bradley E/IP connection parameters) 对话框随即打开。



4. 输入控制器 Ethernet/IP 模块的 IP 地址。 默认情况下，始终为以太网 IP 设备设置端口 44818。
5. 在 “通信路径”(Communication path) 字段中定义从以太网模块到控制器的 CIP 路径。 这一设置创建了以太网模块与 PLC 间的逻辑连接，即使它们位于不同 CIP 网络中。
6. 单击 “确定”(OK) 按钮，关闭对话框。

2.4.4 如何组态“Allen Bradley E/IP ControlLogix” 通道单元连接

简介

组态“Allen Bradley E/IP ControlLogix” 通道单元的连接需要执行下列步骤：

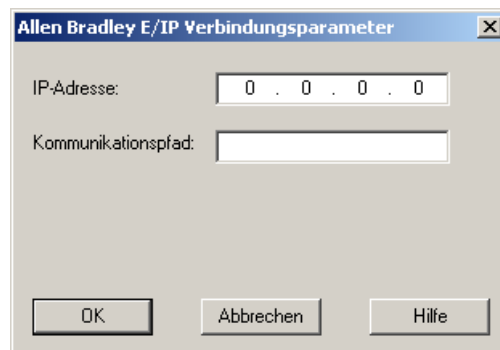
1. 组态连接
2. 组态变量

要求

- 通道“Allen Bradley - Ethernet IP” 的通信驱动程序已安装并集成到项目中。

步骤

1. 在通道单元“Allen Bradley E/IP PLC5” 的快捷菜单中选择条目 “新建连接”(New Connection)。 将创建一个新连接。
2. 选择新连接，然后在 “属性 - 连接”(Properties - Connection) 区域中的 “常规”(General) 下输入连接名。
3. 在快捷菜单中选择 “连接参数”(Connection parameters) 连接。“Allen Bradley E/IP 连接参数”(Allen Bradley E/IP connection parameters) 对话框随即打开。



4. 输入控制器 Ethernet/IP 模块的 IP 地址。默认情况下，始终为以太网 IP 设备设置端口 44818。
5. 在 “通信路径”(Communication path) 字段中定义从以太网模块到控制器的 CIP 路径。这一设置创建了以太网模块与 PLC 间的逻辑连接，即使它们位于不同 CIP 网络中。
6. 单击 “确定”(OK) 按钮，关闭对话框。

2.4 组态通道

2.4.5 实例： 通讯路径

实例 1:

与位于同一个 Allen-Bradley 机架中的 PLC 连接。

1,0

编号	含义
1	代表背板连接。
0	代表 CPU 的插槽编号

实例 2:

与位于远程 Allen-Bradley 机架中的 PLC 连接。 在以太网上连接有两个 Allen-Bradley 机架。

1,2,2,190.130.3.101,1,5

编号	含义
1	背板连接
2	代表第 2 个以太网模块的 CPU 插槽编号。
2	代表以太网连接。
190.130.3.101	网络中远程 AB 机架的 IP 地址 — 尤其是第 3 个以太网模块
1	背板连接
5	CPU 的插槽编号

2.4.6 组态变量

2.4.6.1 组态变量

简介

对于通过通道“Allen Bradley - Ethernet IP” 进行的 WinCC 和自动化系统 (AS) 之间的连接，可在 WinCC 中创建不同数据类型的变量。 本节将列出允许的数据类型。

变量更新

在画面中同时获取来自 PLC 的变量时，“Allen Bradley - Ethernet IP”通道会尝试优化更新。然而这只能在以下条件下完成：

- - 变量在相同的地址范围内。
- - 在地址范围内变量尽可能靠近彼此。

如果不遵守这些建议，则会导致在刷新包含大量变量的图片时出现显著的不同。在某些情况下，可能无法维持采集周期。

组态变量时遵守以下规则可使连接获得最佳性能：

- 最多同时更新 2000 个变量。
- 将变量组合在尽可能小的范围内，最好仅在一个地址范围内。

允许的数据类型

下面列出的数据类型可用来组态变量。

基本数据类型

数据类型	位地址空间
Bool	-
SInt	0-7
USInt	0-7
Int	0-15
UInt	0-15
DInt	0-31
UDInt	0-31
Real	-
String	-

数组

地址	允许的数据类型
数组	SInt、USInt、Int、UInt、DInt、UDInt 和 Real

2.4.6.2 寻址

寻址

WinCC 中通过控制器中的地址唯一地引用变量。地址必须与 PLC 中的变量名称相对应。变量的地址由最大长度为 128 个字符的字符串定义。

使用字符进行寻址

变量寻址的有效字符：

- 字母 (a 至 z, A 至 Z)
- 数字 (0 到 9)
- 下划线 (_)

变量地址由变量名和其它用于在 PLC 中指定变量的字符串组成。

变量名属性：

- 变量名能以下划线字符开头但不能以此结束。
- 出现连续的下划线和空格字符的字符串是无效的。
- 地址不能超过 128 个字符。

说明

保留用作变量寻址的字符不能用在程序/变量名或任何其它地址实例中。

下面列出了保留的字符：

保留的字符	功能
.	元素定界符
:	定义程序变量
,	用于寻址多维数组的定界符
/	保留用于位寻址。
[]	数组元素或数组的寻址

控制器和程序变量

“Allen-Bradley E/IP ControlLogix” 允许对 PLC 变量（全局项目变量）和/或程序变量（全局程序变量）进行寻址。程序变量通过控制器中的程序名称和实际变量名称进行声明。控制器变量按名称寻址。

说明

寻址错误

当变量名称和数据类型不一致时会产生寻址错误。

在 WinCC 的地址域中定义的变量名必须与控制器中的变量名对应。WinCC 与控制器中变量的数据类型必须对应。

说明

无法直接寻址模块特定的变量，例如，输入和输出模块上的数据。而应使用控制器中的变量别名代替。

实例：在 WinCC 中无法寻址 Local:3:O.data。

如果在控制器中为 Local:3:O 定义了别名“MyOut”，则使用 WinCC 时可通过 MyOut.Data 进行寻址。

2.4.6.3 寻址语法

地址的表示法

下表定义了各种寻址方式的写入方法。

表格 2-1 访问数组、基本数据类型和结构元素

数据类型	类型	地址
基本数据类型	PLC 变量	变量名称
	程序变量	程序名:变量名
数组	PLC 变量	数组变量
	程序变量	程序名: 数组变量
位	PLC 变量	变量名/位编号
	程序变量	程序名:变量名/位编号
结构元素	PLC 变量	结构变量.结构元素
	程序变量	程序名: 结构变量.结构元素

2.4 组态通道

说明

不允许对 **Bool**、**Real** 和 **String** 类型的数据进行位寻址，这样做将导致寻址故障。

语法描述

语法描述:

(程序名:) 变量名 ([x(, y) (, z)]) { . 变量名 ([x(, y) (, z)]) } (/位编号)

- “()” 定义表达式的可选的独立实例。
- “{ }” 定义可选的具有多个独立实例的表达式。

地址字符串的长度不能超过 128 个字符。

2.4.6.4 寻址类型

数组元素

通过在变量编辑器中设置索引和相应的表示法寻址 PLC 中的一维、二维和三维数组中的元素。对于可有效进行元素寻址的所有基本类型的数组，数组寻址从元素“0”开始。仅对寻址到的元素执行读/写操作，而不是针对整个数组执行。

位和位变量

允许对所有基本数据类型进行位访问，但不包括 **Bool**、**Real** 和 **String** 类型。还可以对数组/结构元素执行位寻址。在 WinCC 中定义数据类型 **Bool**，用于寻址基本数据类型中的位和位变量。

将使用“/x”或“/0x”（x = 位编号）寻址一位位编号。位编号最多由两位数字定义。

说明

通过数据类型 **SInt**、**Int** 和 **DInt** 中的“**Bool**”数据类型，在更改指定位后，整个变量将再次写入 PLC。同时，对于变量中的其它位是否已经改变，不进行任何检查。因此，PLC 只能对指定变量进行读访问。

结构

用户定义的数据类型由结构创建。这些结构将组合不同数据类型的变量。结构可以包含基本类型、数组或其它结构。在 WinCC 中，只有基本数据类型可作为结构元素进行寻址，而并非整个结构。

结构元素

结构元素通过结构和所需结构元素的名称进行寻址。结构及其元素的名称间用点分隔。除了基本数据类型外，结构元素还可表示数组或其它结构。只能将一维数组用作结构元素。

说明

结构的嵌套深度仅受到地址最大长度 128 个字符的限制。

2.4.6.5 寻址实例

寻址实例表

下表显示了控制变量的基本寻址实例。还可以通过组合形成其它寻址模式。

类型	类型	地址
常规数据类型	PLC 变量	变量名称
	程序变量	程序:变量名
数组	访问二维数组中的元素	Arraytag[Dim1,Dim2]
	结构数组（一维）中的元素	Arraytag[Dim1].structureelement
	基本类型数组（二维）元素中的位	Arraytag[Dim1,Dim2]/Bit
结构	结构中的数组	Structuretag.arraytag
	子结构数组元素中的位	Structuretag.structure2.arraytag[element]/bit

说明

通过在地址前使用来自 PLC 的程序名并用冒号与地址分隔来寻址程序变量。

实例： Programname:arraytag[Dim1,Dim2]

2.4 组态通道

访问数组元素

类型	地址
PLC 变量	Arraytag[Dim1]
	Arraytag[Dim1,Dim2]
	Arraytag[Dim1,Dim2,Dim3]
程序变量	Programname:arraytag[Dim1]
	Programname:arraytag[Dim1,Dim2]
	Programname:arraytag[Dim1,Dim2,Dim3]

2.4.6.6 如何组态用于 Allen Bradley E/IP ControlLogix 通道单元的变量

简介

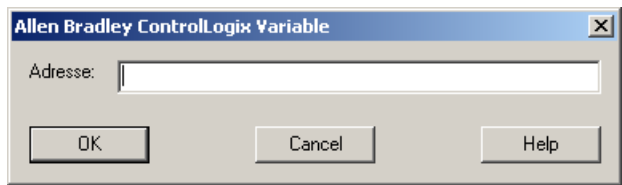
本节介绍如何在自动化系统 (AS) 地址范围内为通道单元“Allen Bradley E/IP ControlLogix”组态变量。

要求

- 通道“Allen Bradley - Ethernet IP” 必须已集成在项目中。
- 必须已在“Allen Bradley E/IP ControlLogix” 通道单元中创建连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。输入变量的名称。
4. 在“数据类型”(Data type) 字段中选择所需的数据类型。
5. 打开“Allen Bradley ControlLogix 变量”(Allen Bradley ControlLogix Tag) 对话框。为此，请单击“地址”(Address) 字段，然后单击  按钮。



- 在“地址”域中输入变量在 AS 中的地址。

说明

寻址错误

当变量名称和数据类型不一致时会产生寻址错误。

在 WinCC 的地址域中定义的变量名必须与控制器中的变量名对应。WinCC 中变量的数据类型必须与控制器中的数据类型对应。

- 单击“确定”(OK) 按钮，关闭对话框。

2.4.6.7 如何组态用于 Allen Bradley E/IP PLC5 或 SLC50x 的按位访问变量


简介

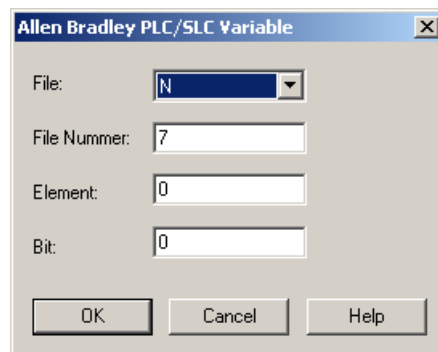
本节说明如何以按位访问方式组态自动化系统 (AS) 中地址区域的变量。

要求

- 通道“Allen Bradley - Ethernet IP” 必须已集成在项目中。
- 必须已在“Allen Bradley E/IP PLC5x” 或“Allen Bradley E/IP SLC50x” 通道单元中创建连接。

步骤

- 选择要为其组态变量的连接。
- 单击表格区下方的“变量”(Tags) 选项卡。
- 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
- 在“数据类型”(Data type) 字段中选择所需的数据类型。
- 打开“Allen Bradley PLC/SLC 变量”(Allen Bradley PLC/SLC Tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



2.4 组态通道

6. 在“文件”域中选择地址范围。条目 N、R、C、T、B、S、I、O、D、A 和 ST 可用。
7. 如与建议的文件号不同，输入“文件号”。
8. 输入“元素”。
9. 根据“文件”域中的设置，定义要寻址的“位”或选择“位（八进制）”值或“子”值。
10. 单击“确定”(OK) 按钮，关闭对话框。

2.4.6.8 如何组态用于 Allen Bradley E/IP PLC5 或 SLC50x 的按字节访问变量


简介

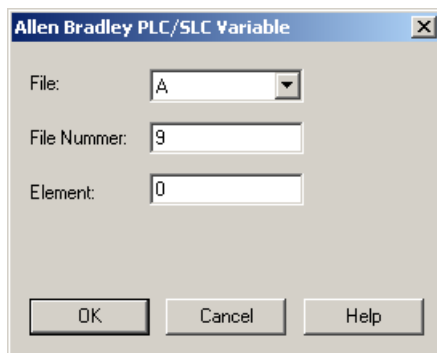
本节说明如何以按字节访问方式组态自动化系统中地址区域的变量。

要求

- 通道“Allen Bradley - Ethernet IP” 必须已集成在项目中。
- 必须已在“Allen Bradley E/IP PLC5x” 或“Allen Bradley E/IP SLC50x” 通道单元中创建连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data type) 字段中选择所需的数据类型。
5. 打开“Allen Bradley PLC/SLC 变量”(Allen Bradley PLC/SLC Tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



6. 在“文件”域中选择 A 或 ST 地址范围。
7. 如与建议的文件号不同，输入“文件号”。

8. 输入“元素”。
9. 单击“确定”(OK)按钮，关闭对话框。

2.4.6.9 如何组态用于 Allen Bradley E/IP PLC5 或 SLC50x 的按字访问变量


简介

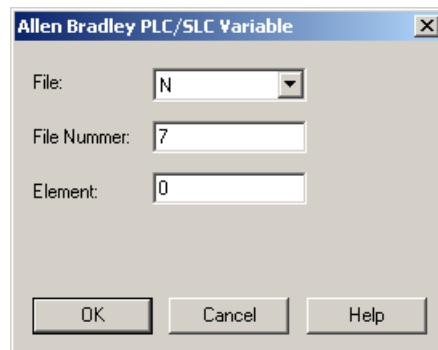
本节介绍如何以按字访问方式组态自动化系统 (AS) 中地址区域的变量。

要求

- 通道“Allen Bradley - Ethernet IP”必须已集成在项目中。
- 必须已在“Allen Bradley E/IP PLC5x”或“Allen Bradley E/IP SLC50x”通道单元中创建连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags)选项卡。
3. 单击“名称”(Name)列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data type)字段中选择所需的数据类型。
5. 打开“Allen Bradley PLC/SLC 变量”(Allen Bradley PLC/SLC Tag)对话框。
为此，请单击“地址”(Address)字段，然后单击  按钮。



6. 在“文件”域中选择地址范围。条目 N、R、C、T、B、S、I、O、D、A 和 ST 可用。
7. 如与建议的文件号不同，输入“文件号”。设置“S”的文件号无法更改。
8. 输入“元素”。
9. 如果显示“子”域，选择其中一个值。这取决于在“文件”域中进行的设置。
10. 单击“确定”(OK)按钮，关闭对话框。

2.4.6.10 如何组态用于 Allen Bradley E/IP PLC5 or SLC50x 的文本变量


简介

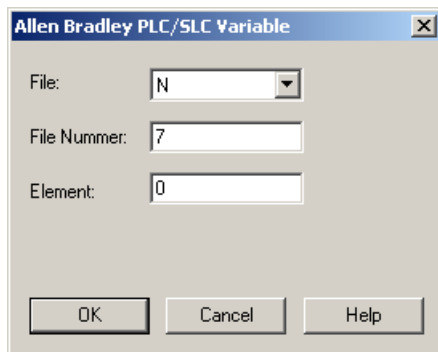
本节介绍如何以按字访问方式组态自动化系统 (AS) 中地址区域的变量。

要求

- 通道“Allen Bradley - Ethernet IP” 必须已集成在项目中。
- 必须已在“Allen Bradley E/IP PLC5x” 或“Allen Bradley E/IP SLC50x” 通道单元中创建连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data type) 字段中选择所需的数据类型。
5. 打开“Allen Bradley PLC/SLC 变量”(Allen Bradley PLC/SLC Tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



6. 在“文件”域中选择 A 或 ST 地址范围。
7. 如与建议的文件号不同，输入“文件号”。设置“S”的文件号无法更改。
8. 输入“元素”。
9. 如果显示“子”域，选择其中一个值。这取决于在“文件”域中进行的设置。
10. 单击“确定”(OK) 按钮，关闭对话框。

Mitsubishi 以太网

3.1 WinCC 通道“Mitsubishi 以太网”

简介

“Mitsubishi 以太网”通道用于 WinCC 站与 FX3U 和 Q 系列 Mitsubishi 控制器间的通信。通过 MELSEC 通信协议（MC 协议）进行通信。

通道单元

“Mitsubishi 以太网”通道配有“Mitsubishi FX3U 系列”和“Mitsubishi Q 系列”通道单元。

3.2 所支持的数据类型

简介

定义实现 WinCC 与相连控制器之间的逻辑连接所需的变量。

“Mitsubishi 以太网”通道支持以下数据类型：

- 二进制变量
- 有符号 16 位数
- 无符号 16 位数
- 有符号 32 位数
- 无符号 32 位数
- 浮点数 32 位 IEEE 754
- 浮点数 64 位 IEEE 754
- 文本变量 8 位字符集
- 文本变量 16 位字符集
- 原始数据变量

3.3 组态通道

3.3.1 组态“Mitsubishi 以太网”通道

简介

在 WinCC 中，WinCC 与自动化系统 (AS) 的通讯需要逻辑连接。本节说明如何组态“Mitsubishi 以太网”通道。

可以通过 TCP/IP 或 UDP/IP 协议，建立与 AS 的连接。因此，在组态“Mitsubishi 以太网”通道的逻辑连接以及选择 TCP/IP 或 UDP 协议时，必须输入 AS 的 IP 地址和 IP 端口号。IP 地址由四个以圆点分隔的数值组成。数值必须在 0 到 255 的范围内。

说明

超时行为

使用 TCP/IP 协议时，不会立即检测到中断的连接。反馈时间可能更长，具体时间则取决于操作系统。

支持 Mitsubishi 控制器

可以为以下 Mitsubishi 控制器组态逻辑连接：

- MELSEC FX3U 系列
- MELSEC System Q

对于这两个控制器系列，连接和变量的组态步骤完全相同。仅仅是特定控制器系列所用地址类型的组态不一样。此外，仅 MELSEC System Q 系列支持信息的路由。

在线组态

“Mitsubishi 以太网”通道支持在线组态变量和连接。

3.3 组态通道

3.3.2 如何组态“Mitsubishi FX3U 系列”通道单元连接

简介

组态“Mitsubishi 以太网”(Mitsubishi Ethernet) 通道时，需要执行以下步骤：

1. 组态连接
2. 组态变量

本节介绍如何组态“Mitsubishi FX3U 系列”通道单元的连接。

要求

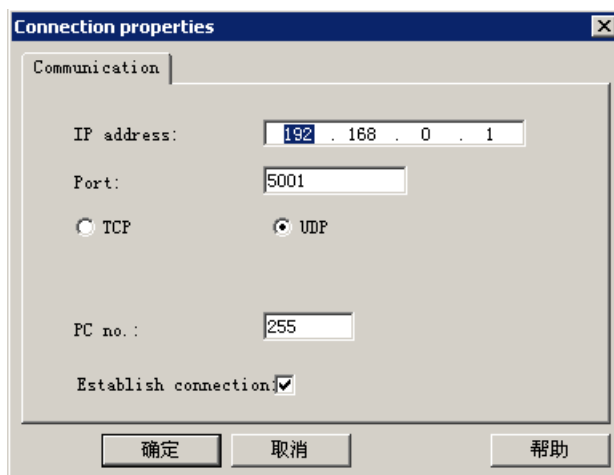
- 已安装“Mitsubishi 以太网”通道的通信驱动程序，并将其集成到项目中。

说明

使用 Mitsubishi 控制器时，还必须组态控制器中的连接。为此，请按照制造商提供的操作说明进行操作。

步骤

1. 在变量管理的导航区域中，从“Mitsubishi 以太网”(Mitsubishi Ethernet) 通信驱动程序树结构下选择通道单元“Mitsubishi FX3U 系列”(Mitsubishi FX3U Series)。
2. 在通道单元的快捷菜单中选择“新建连接”(New Connection) 条目。
3. 输入连接的名称。
4. 从连接的快捷菜单中选择“连接参数”(Connection parameters) 条目。
“连接属性”(Connection properties) 对话框随即打开。



5. 输入控制器的 IP 地址。
6. 输入想要用于 TCP/IP 连接的端口。
数值的有效范围是 0 到 65535。

7. 选择要使用的“TCP”或“UDP”协议。

说明

建议使用默认协议“UDP”。使用该协议时，超时后响应可能更佳。只有在无法使用 UDP 时，才使用 TCP。

8. 输入 PC 编号。
如果不想输入 PC 编号，则必须输入数值 255 或 0。
9. 要建立连接，请选择“建立连接”。
10. 在各种情况下，单击“确定”(OK) 均可关闭对话框。

3.3.3 如何组态“Mitsubishi Q 系列”通道单元连接

简介

组态“Mitsubishi 以太网”(Mitsubishi Ethernet) 通道时，需要执行以下步骤：

1. 组态连接。
2. 组态变量。

本节介绍如何组态“Mitsubishi Q 系列”通道单元的连接。

要求

- 已安装“Mitsubishi 以太网”通道的通信驱动程序，并将其集成到项目中。

说明

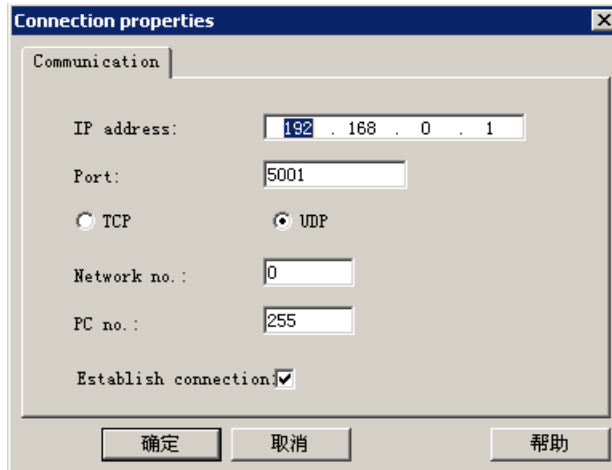
使用 Mitsubishi 控制器时，还必须组态控制器中的连接。为此，请按照制造商提供的操作说明进行操作。

步骤

1. 在变量管理的导航区域中，从“Mitsubishi 以太网”(Mitsubishi Ethernet) 通信驱动程序树结构下选择通道单元“Mitsubishi Q 系列”(Mitsubishi Q Series)。
2. 在通道单元的快捷菜单中选择“新建连接”(New Connection) 条目。
3. 输入连接的名称。

3.3 组态通道

4. 从连接的快捷菜单中选择“连接参数”(Connection parameters) 条目。
“连接属性”(Connection properties) 对话框随即打开。



5. 输入控制器的 IP 地址。
6. 输入想要用于 TCP/IP 连接的端口。
数值的有效范围是 0 到 65535。
7. 选择要使用的“TCP”或“UDP”协议。

说明

建议使用默认协议“UDP”。使用该协议时，超时后响应可能更佳。只有在无法使用 UDP 时，才使用 TCP。

8. 输入网络编号。
默认设置为 0。
9. 输入 PC 编号。
如果不想输入 PC 编号，则必须输入数值 255 或 0。
10. 要建立连接，请选择“建立连接”。
11. 单击“确定”，关闭各对话框。

3.3.4 组态变量

3.3.4.1 组态变量

简介

对于通过“Mitsubishi 以太网”通道进行的 WinCC 和自动化系统之间的连接，可在 WinCC 中创建不同数据类型的变量。

以下章节描述如何组态变量。

地址类型、地址范围和数据类型

下表列出了组态变量和结构变量时可使用的地址类型、地址范围和数据类型。此外，还指定了相应地址类型支持的自动化系统 (AS)。这里的“FX”代表 MELSEC FX3 U 系列，“Q”代表 MELSEC Q 系统。

说明

如果要使用特定的 WinCC 数据类型，必须考虑所选的地址类型。例如，地址类型“D”不支持 WinCC 数据类型“二进制”。

地址类型	代码	地址范围	显示	数据类型	所支持的 AS
输入继电器	X	000000...001FFF	十六进制 (FX: 八进制)	位	FX、Q
输出继电器	Y	000000...001FFF	十六进制 (FX: 八进制)	位	FX、Q
位存储器	M	0...8191	十进制	位	FX、Q
锁存继电器	L	0...8191	十进制	位	Q
位存储器错误	F	0...2047	十进制	位	Q
边沿继电器	V	0...2047	十进制	位	Q
链接继电器	B	000000...001FFF	十六进制	位	Q
数据寄存器	D	0...12287	十进制	字	FX、Q
链接寄存器	W	000000...001FFF	十六进制	字	Q
定时器 (触点)	TS	0...2047	十进制	位	FX、Q
定时器 (OUT 线圈)	TC	0...2047	十进制	位	Q
定时器 (当前值)	TN	0...2047	十进制	字	FX、Q
保持型定时器 (触点)	SS	0...2047	十进制	位	Q
保持型定时器 (线圈)	SC	0...2047	十进制	位	Q
保持型定时器 (值)	SN	0...2047	十进制	字	Q
计数器/触点	CS	0...1023	十进制	位	FX、Q
计数器/线圈	CC	0...1023	十进制	位	Q
计数器/当前值	CN	0...1023	十进制	字	FX、Q
扩展寄存器	R	0...65535	十进制	16 位	FX、Q

3.3 组态通道

地址类型	代码	地址范围	显示	数据类型	所支持的 AS
特殊链接继电器	SB	000000...0007FF	十六进制	位	Q
特殊链接寄存器	SW	000000...0007FF	十六进制	字	Q
步标志	S	0...8191	十进制	位	Q
输入继电器（直接）	DX	000000...001FFF	十六进制	位	Q
输出继电器（直接）	DY	000000...001FFF	十六进制	位	Q
索引寄存器	Z	0...15	十进制	字	Q
诊断位存储器	SM	0...2047	十进制	位	Q
诊断寄存器	SD	0...2047	十进制	字	Q

3.3.4.2 如何组态变量

简介


本节介绍如何组态适合自动化系统 (AS) 中的地址范围的变量。

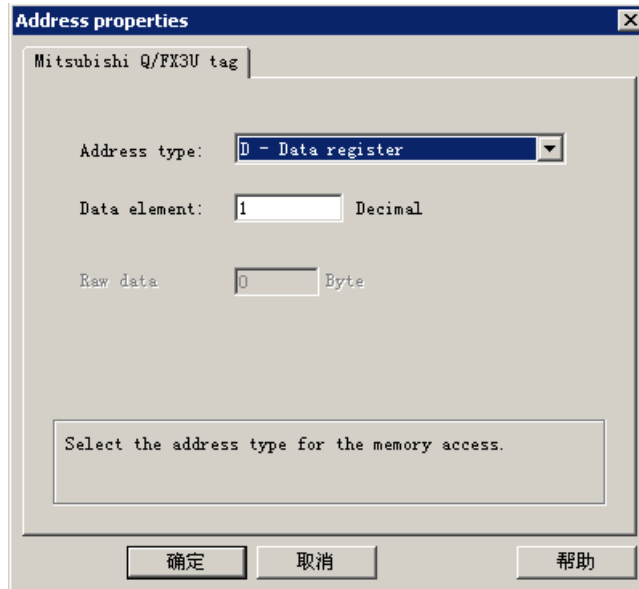
要求

- “Mitsubishi 以太网”通道已集成到项目中。
- 已在“Mitsubishi FX3U 系列”或“Mitsubishi Q 系列”通道单元中，建立连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data type) 字段中选择所需的数据类型。

5. 打开“地址属性”(Address properties)对话框。
为此,请单击“地址”(Address)字段,然后单击  按钮。



6. 设置地址类型。
7. 输入相应的数据元素。该值取决于控制器的组态。
8. 必要的话,输入原始数据长度。
9. 单击“确定”,关闭“地址属性”对话框。
变量地址将显示在“变量属性”对话框的“地址”字段中。地址将调整为 AS 的数据格式。

3.3 组态通道

Modbus TCPIP

4.1 “Modbus TCP/IP” 通道

简介

“Modbus TCPIP” 通道用于 WinCC 站和支持 Modbus 的 PLC 之间通过以太网的通讯。其通讯使用 Modbus TCP/IP 协议处理。

通道单元

“Modbus TCPIP” 通道附带“Modbus TCP/IP Unit #1” 通道单元。

4.2 支持的数据类型

简介

为与已连接控制器的逻辑连接定义所需变量。“Modbus TCP/IP”通道支持以下数据类型：

- 二进制变量
- 有符号 16 位数
- 无符号 16 位数
- 有符号 32 位数
- 无符号 32 位数
- 浮点数 32 位 IEEE 754
- 文本变量 8 位字符集
- 文本变量 16 位字符集

4.3 组态通道

4.3.1 组态“Modbus TCP/IP”通道

简介

在 WinCC 中，WinCC 与自动化系统 (AS) 的通讯需要逻辑连接。本节说明与“Modbus TCP/IP Unit #1”通道单元之间的通讯。所有连接特定的参数均在安装期间定义。

使用 TCP/IP 协议时，必须定义 AS 的 IP 地址以用于逻辑连接。IP 地址由四个以圆点分隔的数值组成。数值范围必须在 0-255 之内。

说明

超时行为

使用 TCP/IP 协议时，不会立即检测到中断的连接。回查消息可能耗时达一分钟。

可用的 Modbus TCP/IP 通讯方法

下列通讯类型已经过系统测试并得到认证：

- 点对点通讯
- WinCC 站与可选数量控制器之间的多点连接。

说明

因为 WinCC 站作为 Modbus 主机工作，所以不可能通过桥接器将 WinCC 站集成在 Modbus 网络中。

在线组态

不支持在线组态。

4.3 组态通道

4.3.2 如何组态连接

简介

组态通道“Modbus TCP/IP” 需要以下步骤：

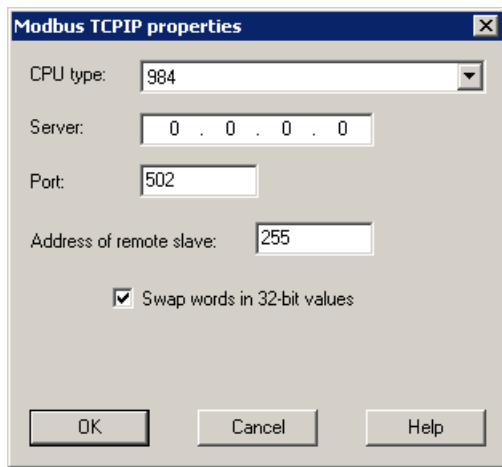
1. 组态连接
2. 组态变量

要求

- 通道“Modbus TCP/IP” 的通信驱动程序已安装并集成到项目中。

步骤

1. 在变量管理的导航区域中，从“Modbus TCP/IP” 通信驱动程序树结构下选择通道单元“Modbus TCP/IP Unit#1” 。
2. 在通道单元的快捷菜单中选择 “新建连接”(New Connection) 条目。
3. 输入连接的名称。
4. 从连接的快捷菜单中选择 “连接参数”(Connection parameters) 条目。“Modbus TCP/IP 属性”(Modbus TCP/IP properties) 对话框随即打开。



5. 在“CPU 类型” 下选择已连接的 Modicon 控制器。 以下 CPU 可供选择：
 - 984
将该 CPU 类型用于 CPU 984（CPU 984A、984B 和 984X 除外）
 - Modicon Compact、Modicon Quantum 和 Modicon Momentum
 - Modicon Premium 和 Modicon Micro
6. 在“服务器” 域中输入控制器的 IP 地址。
7. 在“端口” 域中输入用于 TCP/IP 连接的端口。 Modbus TCP/IP 连接的默认端口为 502。

8. 如果使用桥接器，在“远程从站的地址”(Address of remote slave) 字段中输入远程控制器的从站地址。
如果未使用桥接器，则必须输入默认值 255 或 0 作为地址。

说明

因为 WinCC 站作为 Modbus 主机工作，所以不可能通过桥接器将 WinCC 站集成在 Modbus 网络中。

9. 要交换 32 位过程值中 16 位寄存器的顺序，请选中“交换 32 位值中的字”(Swap words in 32-bit values)。
10. 单击“确定”(OK) 关闭对话框。

4.3.3 组态变量

4.3.3.1 组态变量

简介

对于通过通道“Modbus TCP/IP”进行的 WinCC 和自动化系统 (AS) 之间的连接，可在 WinCC 中创建不同数据类型的变量。

以下章节描述如何组态变量。AS 中的数据范围与 WinCC 变量的数据类型寻址方式不同。

使用 Modbus TCP/IP 协议进行变量更新

在画面中同时获取来自控制器的变量时，Modbus TCP/IP 通道会尝试优化更新。然而这只能在以下条件下完成：

- 变量在相同的地址范围内。
- 在地址范围内变量尽可能靠近彼此。

如果不遵守这些建议，则会导致在刷新包含大量变量的图片时出现显著的不同。在某些情况下，可能无法维持采集周期。

组态变量时遵守以下规则可使连接获得最佳性能：

- 最多同时更新 2000 个变量。
- 将变量组合在尽可能小的范围内，最好仅在一个地址范围内。

4.3 组态通道

控制器中的数据类型和地址范围

下表列出了组态变量和结构化变量时可使用的数据类型和地址范围。

名称	使用 CPU Premium/ Micro 时的区域	使用 CPU 984、 Compact、 Quantum 和 Momentum 时的区 域	数据类型
线圈 (离散量输出)	%M ¹⁾	0x	位
离散量输入	(%I) – Premium/ Micro 未实现	1x	位
输入寄存器	(%IW) – Premium/ Micro 未实现	3x	Bit、 +/- Int、 Int
暂停寄存器 (输出)	%MW	4x	Bit ²⁾ 、 +/-Int、 Int、 +/-Double、 Double、 Float、 ASCII
扩展内存 (仅适用于“Quantum/Momentum”CPU)	--	6x	Bit ²⁾ 、 +/-Int、 Int、 +/-Double、 Double、 Float、 ASCII

1) 由于外部控制器的系统特征，无法访问地址区域末尾的最后 x 位。

2) 对于写访问，请注意：

通过“4x”、“6x”和“%MW”区域中的“bit”数据类型，在更改指定位后，整个字将写回 PLC。但期间并不检查字中的其它位是否已改变。因此，PLC 只对指定字进行读访问。

984、Compact、Quantum 和 Momentum 系列控制器所使用的标准位计数方法 (16 LSB - 1 MSB) 只能用于那些在“变量”编辑器中显示数据类型“bit”的 CPU。位位置分配如下：

	左字节								右字节							
变量计数	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

当在 WinCC 中的其它位置输入位编号时，WinCC 适用的位分配如下 (0 LSB - 15 MSB)：

位置的计数方法	左字节								右字节							
在 WinCC 中，可组态：	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0
	5	4	3	2	1	0										

此位计数方法也适用于 Modicon Premium 和 Modicon Micro 控制器

“有符号数”的格式

占位符“+/-”适用于“有符号整数”和“有符号长整数”两种数据类型。

参见

如何组态按位访问的变量 (页 83)

如何组态按字访问的变量 (页 84)

如何组态文本变量 (页 85)

4.3.3.2 如何组态按位访问的变量

简介

本节说明如何以按位访问方式组态自动化系统 (AS) 中地址区域的变量。


要求

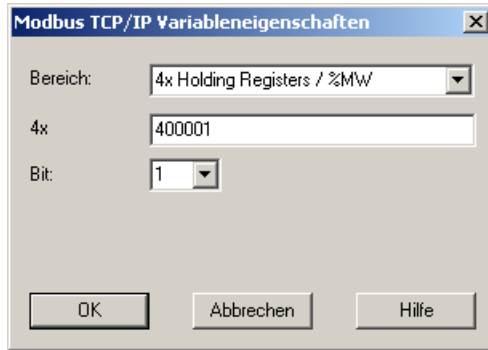
- 通道“Modbus TCP/IP”必须集成在项目中。
- 必须已在“Modbus TCP/IP Unit #1”通道单元中创建连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data Type) 字段中，设置“二进制变量”(Binary tag) 数据类型。

4.3 组态通道

5. 打开“Modbus TCP/IP 变量属性”(Modbus TCP/IP Tag Properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



6. 在相应的地址字段（例如“4x”）中输入变量的地址。该值取决于控制器的组态。
7. 如必要，在“位”域中输入位的地址。条目是否可用取决于您在“来源区域”域中的选择。
8. 如果已在“区域”域中设置了值“6x 扩展的内存”，则为“文件”选择一个值。
9. 单击“确定”(OK) 关闭对话框。

说明

关闭“Modbus TCP/IP 变量属性”对话框后，将在“变量属性”对话框的“地址”域中显示控制器中变量的内部地址。此地址已按照 AS 数据格式进行调整，因此可能会与输入的地址不同。

参见

如何组态文本变量 (页 85)

4.3.3.3 如何组态按字访问的变量


简介

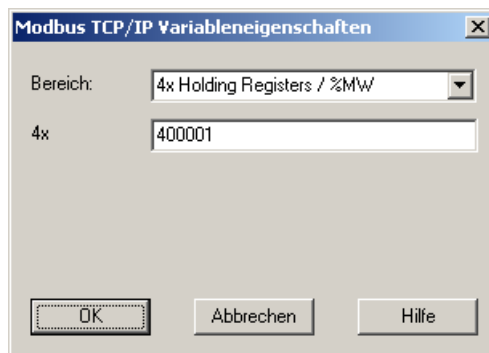
本节介绍如何以按字访问方式组态自动化系统 (AS) 中地址区域的变量。

要求

- 通道“Modbus TCP/IP”必须集成在项目中。
- 必须已在“Modbus TCP/IP Unit #1”通道单元中创建连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”域中，将数据类型设置为“无符号 16 位数”。
5. 打开“Modbus TCP/IP 变量属性”(Modbus TCP/IP tag properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



6. 在相应的地址字段（例如“4x”）中输入变量的地址。该值取决于控制器的组态。
7. 单击“确定”(OK) 按钮，关闭上述两个对话框。

说明

关闭“Modbus TCP/IP 变量属性”对话框后，将在“变量属性”对话框的“地址”域中显示控制器中变量的内部地址。此地址已按照 AS 数据格式进行调整，因此可能会与输入的地址不同。

4.3.3.4 如何组态文本变量

简介


本节介绍如何以按字访问方式组态自动化系统 (AS) 中地址区域的变量。

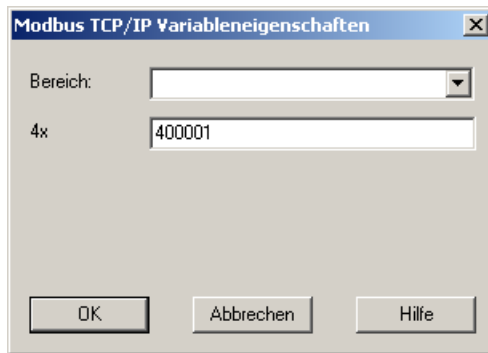
要求

- 通道“Modbus TCP/IP” 必须集成在项目中。
- 必须已在“Modbus TCP/IP Unit #1” 通道单元中创建连接。

4.3 组态通道

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”域中，将“文本变量 8 位字符集”设置为数据类型。
5. 打开“Modbus TCP/IP 变量属性”(Modbus TCP/IP tag properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



6. 在地址域“4x”中，输入变量的地址。该值取决于控制器的组态。
7. 单击“确定”(OK) 按钮，关闭上述两个对话框。

OPC 通道

5.1 WinCC OPC 通道

简介

WinCC 可以用作 OPC 服务器和 OPC 客户端。OPC 通道是 WinCC 的 OPC 客户端应用程序。

OPC 通信驱动程序可用作 OPC DA 客户端、OPC XML 客户端和 OPC UA 客户端。在“OPC UA 通道”下可找到有关 OPC UA 客户端的文档。

下列 OPC 组件会自动安装：

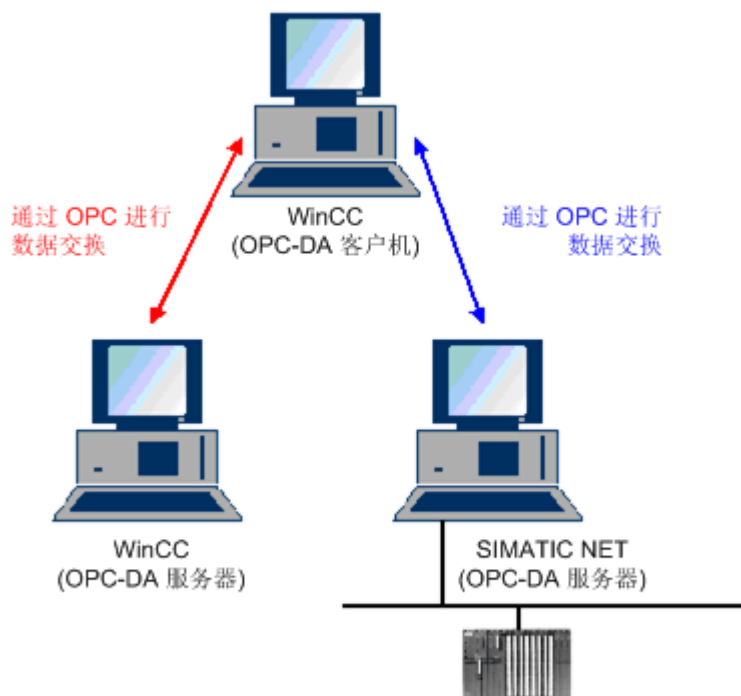
- OPC 通讯驱动程序
- OPC 条目管理器

可能的应用

WinCC 作为 OPC DA 客户端

将 WinCC 用作 OPC DA 客户端时，OPC 通道必须添加到 WinCC 项目中。在 WinCC OPC DA 客户端的 WinCC 项目中创建用于数据交换的连接；将用来处理对 OPC DA 服务器的 WinCC 变量的访问。

为了简化过程，系统使用 OPC 条目管理器。一个 WinCC OPC DA 客户端可以访问多个 OPC DA 服务器。这需要为每个 OPC 服务器创建一个连接。通过这种方式，WinCC OPC DA 客户端可用作中央操作和监控站。



说明

WinCC OPC 通道只能与状态为“OPC_STATUS_RUNNING”的 OPC 服务器建立连接。

说明

“OPC” 通道

连接名称不支持 Unicode。请确保使用同一种语言对项目中的所有连接进行命名。打开计算机的控制面板为不支持 Unicode 的程序中所使用的语言设置代码页。

5.2 OPC 条目管理器

简介

在 WinCC OPC 客户端的 WinCC 项目中组态连接和 WinCC 变量以访问 OPC 服务器的变量。OPC 条目管理器可简化此过程。OPC 条目管理器随 WinCC 自动安装。

说明

“OPC” 通道

连接名称不支持 **Unicode**。请确保使用同一种语言对项目中的所有连接进行命名。打开计算机的控制面板为不支持 **Unicode** 的程序中所使用的语言设置代码页。

要求

为使用 OPC 条目管理器实现组态，必须满足下列要求：

- OPC 服务器是 OPC DA 服务器或 OPC XML 服务器。
- 已在 OPC 服务器上组态了一个变量。
- 如果要将 WinCC 用作 OPC 服务器，必须启用 WinCC OPC 服务器的 WinCC 项目。如果不是这样，OPC 条目管理器不能访问 WinCC OPC 服务器。
- 必须可以通过 IP 地址或 HTTP 访问 OPC 服务器的计算机。
- OPC 服务器必须支持浏览器功能。如果不是这样，必须手动组态对 OPC 服务器变量的访问。

说明

如果在 OPC 条目管理器打开时更改 WinCC 项目管理器中的语言，则在单击“浏览服务器”(Browse Server) 按钮时，不会显示任何变量。请在更改语言之前退出 OPC 条目管理器。

OPC 条目管理器的任务

OPC 条目管理器完成下列任务：

- 选择 OPC 服务器
- 创建连接
- 变量选择
- 添加变量

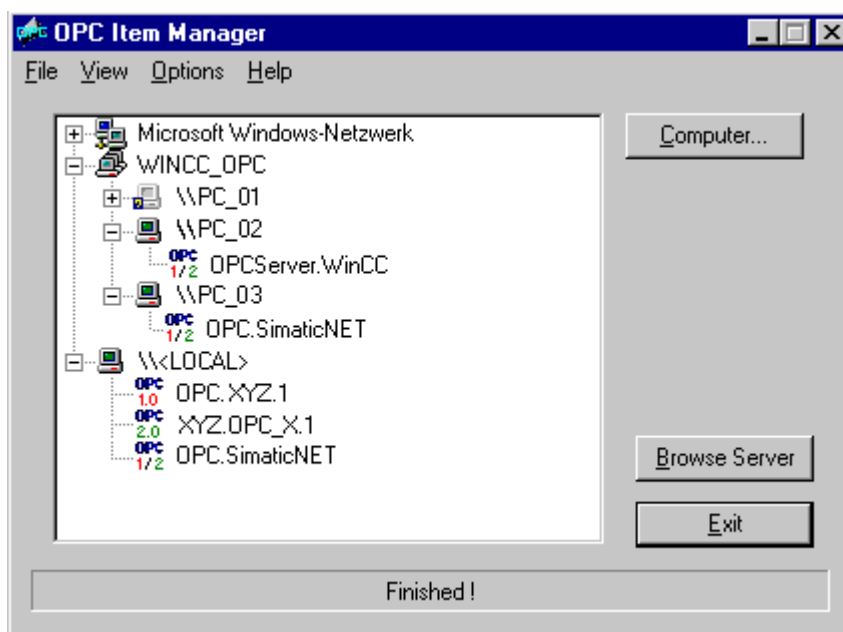
选择 OPC 服务器

OPC DA 服务器

OPC 条目管理器可用于确定网络中 OPC DA 服务器的名称。这些 OPC DA 服务器能够在同一计算机上运行或在已联网的不同计算机上运行。有关详细信息，请参见“WinCC OPC DA 客户端”。

OPC XML 服务器

OPC 条目管理器不能用来查询 OPC XML 服务器的名称。可以使用“OPC Web 服务器”按钮添加 OPC XML 服务器。有关详细信息，请参见“WinCC OPC XML 客户端”。



OPC 条目管理器的图标	描述
	联网计算机尚未检查其是否装有 OPC DA 服务器。
	未在网络中找到计算机或不能访问计算机。
	联网计算机经检查装有 OPC DA 服务器。
	联网计算机包含以 OPC 符号标识的 OPC DA 服务器。数字指示使用了 WinCC OPC DA 客户端的哪种 OPC DA 规范。
\\<LOCAL>	指运行 OPC 条目管理器的计算机。
	OPC XML 服务器的名称。
http://	可以使用“OPC Web 服务器”按钮添加 OPC XML 服务器。

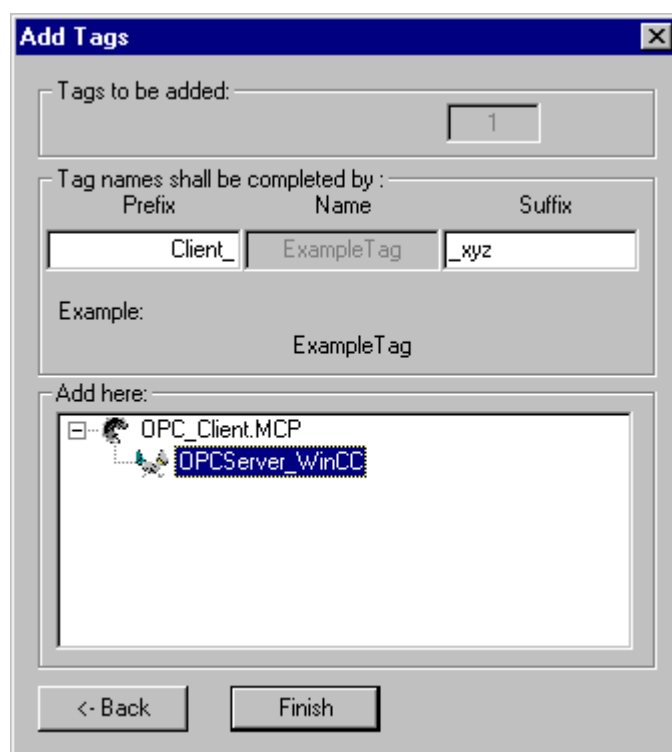
创建连接

创建连接时，OPC 条目管理器会组态所有需要的设置。如果到 OPC 服务器的连接已经创建，则该功能不可用。

变量选择

在 WinCC OPC 客户端要访问的 OPC 服务器上，可以使用变量选择对话框选择一个或多个变量。可以使用过滤标准限制变量选择对话框中的选择。

添加变量



可以在“添加变量”(Add Tags)对话框中设置访问 OPC 服务器变量的 WinCC 变量的名称。

WinCC 变量名由“前缀”、“名称”和“后缀”组成。“名称”域用“ExampleTag”文本预组态。“ExampleTag”代表 WinCC OPC 服务器的 WinCC 变量名称。

为了区分 WinCC OPC 客户端上的 WinCC 变量名和 WinCC OPC 服务器上的 WinCC 变量名，可以分配一个前缀或一个后缀。当组态项目监控时，必须指定一个前缀或一个后缀。

变量名在给定的 WinCC 项目中只能指定一次。

实例

WinCC OPC DA 服务器上的 WinCC 变量名称为“OPC_Server_Tag”。在前缀域中输入值“Client_”，在后缀域中输入“_xyz”。在 WinCC OPC DA 客户端的 WinCC 项目中，将创建 WinCC 变量“Client_OPC_Server_Tag_xyz”。

如果 OPC 服务器上的变量名中含有特殊字符，它们将用下划线“_”代替，因为 OPC 条目管理器并不支持在变量名中出现的所有特殊字符。

单击“完成”(Finish) 将 WinCC 变量添加至 WinCC OPC DA 客户端的 WinCC 项目中。OPC 条目管理器自动设置 WinCC 变量的数据类型、名称和地址参数。

参见

[如何通过 OPC 条目管理器访问 WinCC 变量 \(页 113\)](#)

[如何通过 OPC 条目管理器访问 WinCC 变量 \(页 95\)](#)

5.3 所支持的 WinCC 数据类型概述

下面的列表显示了 WinCC OPC DA 客户端和 WinCC OPC DA 服务器所支持的数据类型：

- 二进制变量
- 有符号 8 位数
- 无符号 8 位数
- 有符号 16 位数
- 无符号 16 位数
- 有符号 32 位数
- 无符号 32 位数
- 浮点数 32 位 IEEE 754
- 64 位 IEEE 754 浮点数
- 文本变量，8 位字符集
- 文本变量，16 位字符集
- 原始数据类型
- 结构类型
- 文本参考
- 日期/时间

说明

结构类型

对于结构类型，只支持结构元素，而不支持结构本身。但可在以后对结构进行组态。有关更多的信息，请参阅主题“在 WinCC OPC DA 客户端上使用结构”。

文本参考

如果使用 OPC 条目管理器创建一个文本变量，它会被分配 160 个字符的长度。可以任意更改该长度。

参见

如何在 WinCC OPC DA 客户机上使用结构 (页 103)

5.4 WinCC OPC DA 客户机

5.4.1 WinCC OPC DA 客户机的功能

简介

OPC 通道不需要独立的通讯模块。OPC 通道是使用 OPC 软件接口通过 OPC DA 服务器访问过程数据的应用程序。

当使用 WinCC 作为 OPC DA 客户端时，OPC 通道必须添加到 WinCC 项目上。

如果建立了与 WinCC OPC DA 服务器的通讯，将交换 WinCC 变量的值。为此，在 WinCC OPC DA 客户端的 WinCC 项目中建立一个连接，用于处理对 WinCC OPC DA 服务器的访问。

为使 WinCC OPC DA 客户端能访问多个 OPC DA 服务器，必须在 WinCC 项目中为每台 OPC DA 服务器创建连接。有关检测通道和变量故障的更多信息，请参阅“故障检测”。

说明

WinCC OPC 通道只能与状态为“OPC_STATUS_RUNNING”的 OPC 服务器建立连接。

说明

“OPC”通道

连接名称不支持 Unicode。请确保使用同一种语言对项目中的所有连接进行命名。打开计算机的控制面板为不支持 Unicode 的程序中所使用的语言设置代码页。

连接监控

WinCC OPC-DA 客户端中集成了三种用于连接监视的机制。因而在网络出错或 OPC DA 服务器出现故障的情况下，可以采取最佳的可行性措施。

1. 如果对 DCOM 激活的处理时间超出 5 秒的警告值，变量将被分配值“寻址错误”。如果处理时间段超出 10 秒的取消值，到 OPC DA 服务器的连接将被中断。这显示在 WinCC 项目管理器的“连接状态”对话框中。

OPC DA 规范 3.00 附带“保持有效”特性。如果 OPC DA 服务器支持 OPC DA 规范 3.00，将使用该特性。该特性促使 OPC DA 服务器自动触发周期更新（调用 OnDataChange），即使变量值没有改变。如果此定期更新被禁止，WinCC OPC DA 客户端将终止连接。

这一行为对于支持 OPC DA 规范 2.05a 的 OPC DA 服务器也同样适用。为了检查与 OPC DA 服务器的连接，WinCC OPC DA 客户端每 10 秒周期性地定期进行状态请求。如果此定期更新被禁止，WinCC OPC DA 客户端将终止连接。

通常，当连接失效时，WinCC OPC DA 客户端将终止与 OPC DA 服务器的连接。WinCC OPC DA 客户端每 10 秒自动试图再次重新建立连接。

参见

如何在 WinCC OPC DA 客户机上使用结构 (页 103)

在不使用 OPC 条目管理器的情况下访问 WinCC 变量 (页 101)

通过 OPC 条目管理器组态访问 (页 97)

所支持的 WinCC 数据类型概述 (页 93)

OPC 条目管理器 (页 89)

通道和变量的诊断 (页 507)

OPC 规范及兼容性 (页 123)

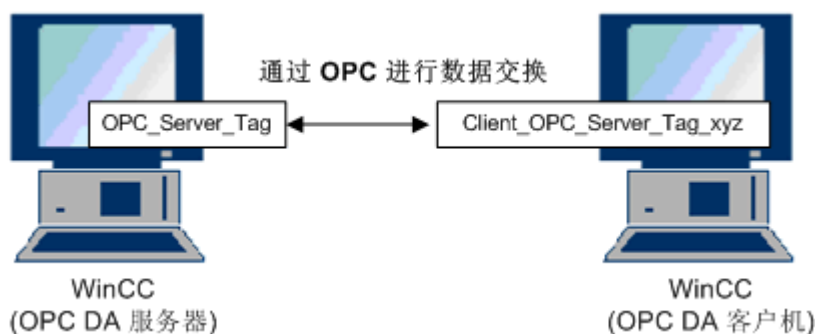
WinCC OPC DA 服务端的功能 (页 132)

5.4.2 如何通过 OPC 条目管理器访问 WinCC 变量

5.4.2.1 如何通过 OPC 条目管理器访问 WinCC 变量

简介

WinCC 之间建立 OPC 连接后，就会通过 WinCC 变量进行数据交换。WinCC OPC DA 客户机会通过 OPC 连接来读取 WinCC OPC DA 服务器上的 WinCC 变量“OPC_Server_Tag”。为了简化过程，系统使用 OPC 条目管理器。



要求

- 两台带 WinCC 项目的计算机。
- 这两台计算机必须能够通过其 IP 地址被访问。

组态步骤

在 WinCC OPC DA 客户机的 WinCC 项目中，需要下列组态：

- 创建一个连接。
- 在 WinCC OPCXML 客户端上组态“XMLClient OPC_Var1_xyz”WinCC 变量，此变量用于访问 WinCC OPC DA 服务器的 WinCC 变量。

参见

通过 OPC 条目管理器组态访问 (页 97)

在 WinCC OPC DA 客户机上组态 OPC 通道 (页 96)

通过 OPC 条目管理器组态访问 (页 114)

WinCC 至 WinCC 的连接实例 (页 136)

5.4.2.2 在 WinCC OPC DA 客户机上组态 OPC 通道

引言

为使用 OPC 进行数据交换，在 WinCC 项目中必须建立 OPC 通道。

步骤

1. 在 WinCC OPC DA 客户机上的 WinCC 项目浏览器的浏览窗口中，单击“变量管理器”图标。
2. 从“变量管理器”的快捷菜单中选择“添加新的驱动程序”。将打开“添加新的驱动程序”对话框。
3. 选择“OPC.chn”驱动程序并单击“打开”按钮。将创建通道并在变量管理器中显示通讯驱动程序。

参见

通过 OPC 条目管理器组态访问 (页 97)

5.4.2.3 通过 OPC 条目管理器组态访问

简介

本节说明如何用 OPC 条目管理器对 WinCC OPC DA 服务器的 WinCC 变量“OPC_Server_Tag” 的访问进行组态。

要求

- 在 WinCC OPC DA 服务器的 WinCC 项目中，组态一个数据类型为“有符号 16 位数”，名为“OPC_Server_Tag” 的内部变量。
- 启用 WinCC OPC DA 服务器的 WinCC 项目。
- 在 WinCC OPC DA 客户端的 WinCC 项目中添加“OPC” 通道。

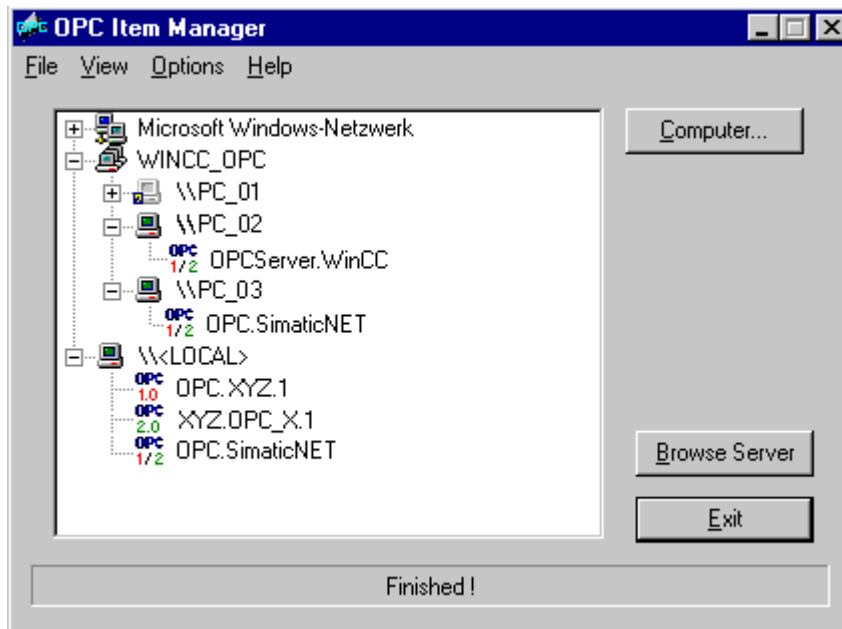
说明

“OPC” 通道

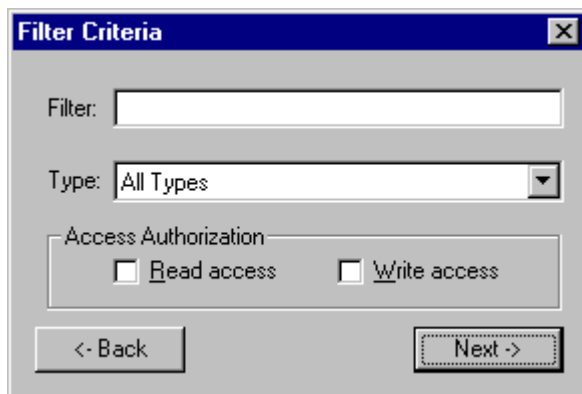
连接名称不支持 Unicode。请确保使用同一种语言对项目中的所有连接进行命名。打开计算机的控制面板为不支持 Unicode 的程序中所使用的语言设置代码页。

步骤

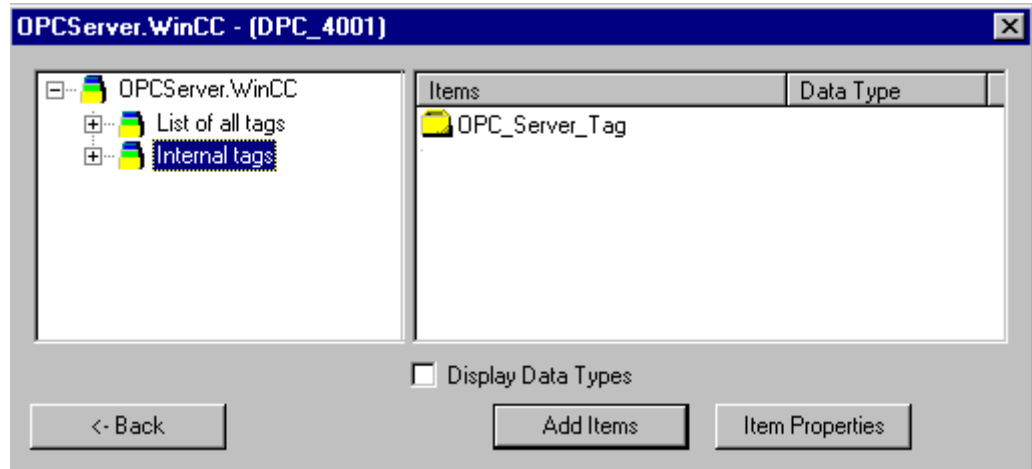
1. 在 WinCC OPC DA 客户端上，从“OPC 组 (OPCHN Unit#1)” 通道单元的快捷菜单中选择 “系统参数”。将打开“OPC 条目管理器”(OPC Item Manager)。



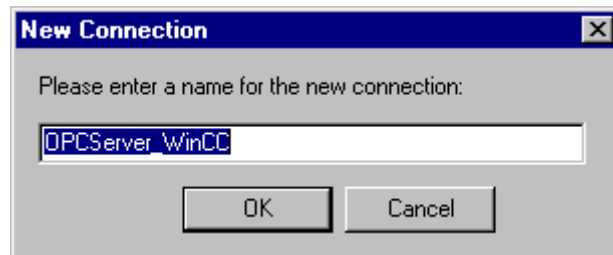
2. 从选择对话框中选择用作 WinCC OPC DA 服务器的计算机的名称。
从显示的列表中选择“OPCServer.WinCC”。
3. 单击 “浏览服务器” 按钮。
将打开 “过滤标准”(Filter criteria) 对话框。



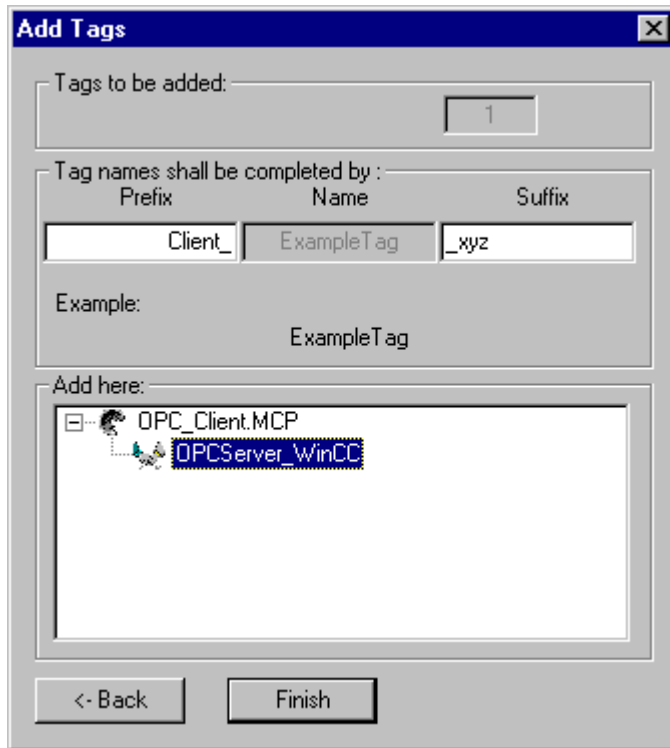
- 单击“下一步”(Next)。
将打开“OPCServer.WinCC ...”对话框。



- 选择 WinCC 变量“OPC_Server_Tag”。
单击“添加条目”按钮。
- 如果已创建到 WinCC OPC DA 服务器的连接，继续步骤 6。
如果没有创建连接，将显示一条消息。
单击“是”(Yes)按钮。将打开“新建连接”(New Connection)对话框。



7. 输入“OPCServer_WinCC”作为连接的名称。单击“确定”(OK)。将打开“添加变量”(Add Tags)对话框。



8. 在前缀字段中输入“Client_”，在后缀字段中输入“_xyz”。
9. 选择连接“OPCServer_WinCC”。单击“Finish”（完成）。
- 10.在“OPCServer.WinCC ...”对话框中，单击“返回”(Back)按钮。单击“退出”关闭 OPC 条目管理器。

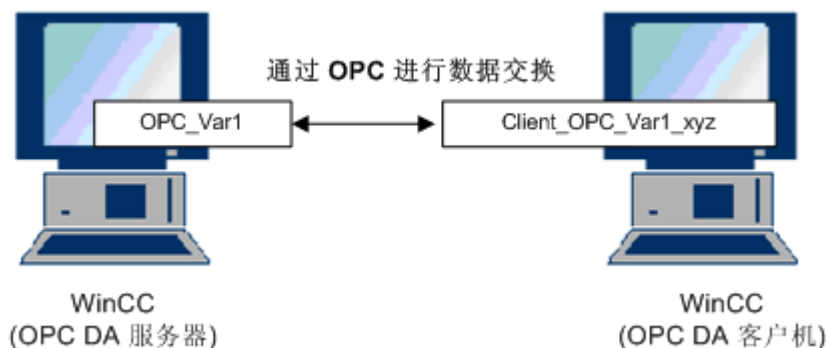
参见

在 WinCC OPC DA 客户机上组态 OPC 通道 (页 96)

5.4.3 在不使用 OPC 条目管理器的情况下访问 WinCC 变量

简介

不支持浏览器功能的 OPC 服务器需要手动组态访问。下面通过一个 WinCC-WinCC OPC 连接的实例来说明 WinCC OPC DA 客户机上 WinCC 变量的组态。



说明

要在不使用 OPC 条目管理器的情况下访问 WinCC 变量，必须手动设置 ItemID。对 WinCC 变量进行寻址时，也可以指定符号计算机名称（服务器前缀）。条目标识号具有下列语法：服务器前缀::WinCC 变量。对本地 WinCC 项目的 WinCC 变量编址时，将忽略服务器前缀。

在 WinCC OPC DA 客户机的 WinCC 项目中，需要下列组态：

1. 选择要访问的“OPC_Var1”WinCC 变量。
2. 创建一个连接。
3. 对访问 WinCC OPC DA 服务器 WinCC 变量的“Client_OPC_Var1_xyz”WinCC 变量进行组态。

要求

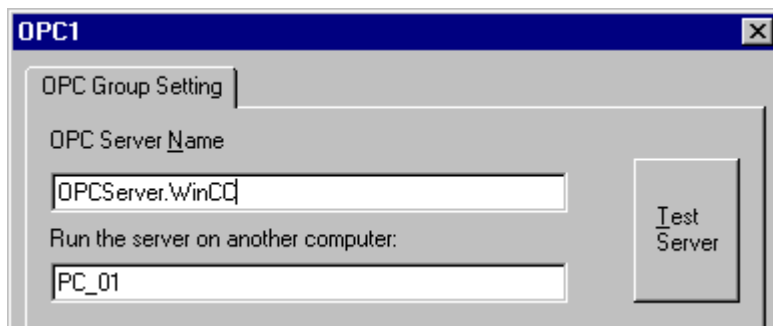
- 两台带 WinCC 项目的计算机。
- 这两台计算机必须能够通过其 IP 地址被访问。
- 在 WinCC OPC DA 服务器的 WinCC 项目中，组态名称为“OPC_Var1”且数据类型为“有符号 16 位数”的内部变量。
- 启用 WinCC OPC DA 服务器的 WinCC 项目。
- 将 OPC 通道添加到 WinCC OPC DA 客户机的 WinCC 项目。

说明**“OPC” 通道**

连接名称不支持 **Unicode**。请确保使用同一种语言对项目中的所有连接进行命名。打开计算机的控制面板为不支持 **Unicode** 的程序中所使用的语言设置代码页。

步骤

1. 在 WinCC OPC DA 客户机上，从“OPC 组 (OPCHN Unit#1)”通道单元的快捷菜单中选择“新建连接”。将打开“连接属性”(Connection Properties)对话框。在相应域中输入连接的名称。
2. 单击“属性”按钮。将显示以连接名称为标题的对话框。



对于与 WinCC V6 的连接，“OPC 服务器名称”域中的条目必须为“OPCServer.WinCC”。

3. 在“在本计算机上启动服务器”域中，输入要用作 OPC DA 服务器的计算机名称。单击“测试服务器”，检查与 WinCC OPC DA 服务器的连接。
4. 从连接的快捷菜单中选择“新建变量”。将打开“变量属性”对话框。
5. 在“变量”域中输入名称“Client_OPC_Var1_xyz”。将数据类型设置为“有符号 16 位数”。

- 在“变量属性”对话框中，单击“选择”按钮。将打开“地址属性”对话框。



在“条目名称”域中输入 WinCC OPC DA 服务器的 WinCC 变量的名称。保持“访问路径”域中的条目不变。将数据类型设置为“有符号 16 位数”。

- 单击“确定”关闭所有打开的对话框。

5.4.4 在 WinCC OPC DA 客户机上使用结构

5.4.4.1 如何在 WinCC OPC DA 客户机上使用结构

简介

结构用于组织构成逻辑单元的变量和变量类型。这允许使用单个的逻辑名称对其进行引用。

OPC DA 规范不支持结构。这样就不能使用 OPC 条目管理器建立结构，只能建立结构中的单个变量。如果仍然想要在 WinCC OPC DA 客户端上使用结构，必须随后在 WinCC OPC DA 客户端的 WinCC 项目中组态数据结构，以便为结构提供服务器变量的相关条目名称。

要求

- 两台带 WinCC 项目的计算机。
- 这两台计算机必须能够通过其 IP 地址被访问。

组态步骤

要在 WinCC OPC DA 客户端上使用结构，下列组态步骤是必需的：

- 在 WinCC OPC DA 服务器上组态结构和结构变量
- 使用 WinCC OPC DA 客户端上 WinCC 项目中的结构

参见

如何在 WinCC OPC DA 客户机上组态结构 (页 105)

在 WinCC OPC DA 服务器上组态结构和结构变量 (页 104)

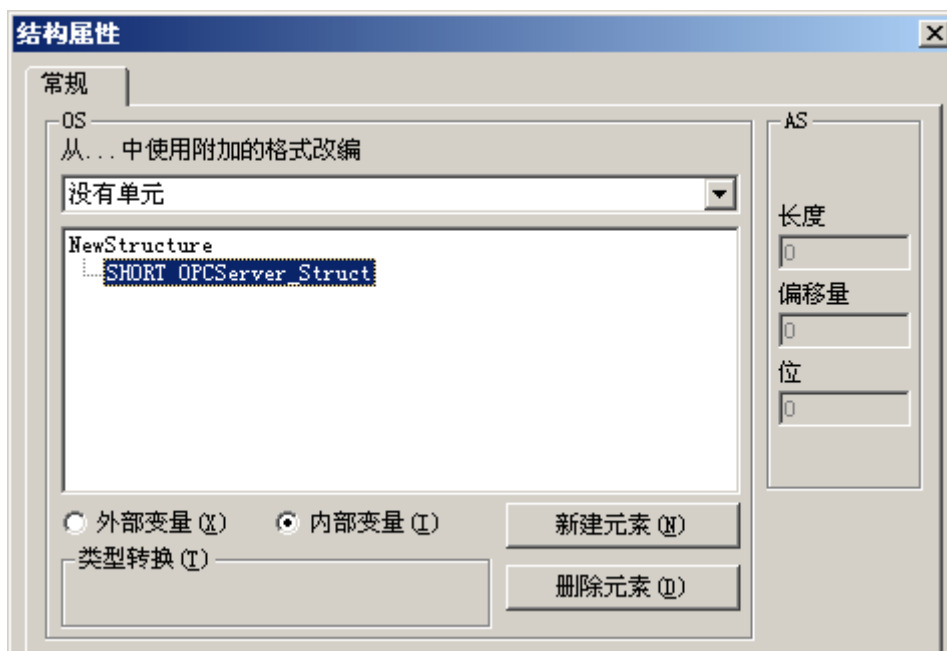
5.4.4.2 在 WinCC OPC DA 服务器上组态结构和结构变量

引言

在本节中，将在 OPC DA 服务器的 WinCC 项目中创建一个结构和一个结构变量。OPC DA 客户机需要该组态来访问结构变量。

步骤

1. 从 WinCC OPC DA 服务器上的结构类型快捷菜单中选择“新建结构类型”。将显示“结构属性”对话框。
2. 单击“新建元素”，创建数据类型为 SHORT 的内部变量“OPCServer_Struct”。



单击“确定”关闭对话框。

3. 在浏览窗口中，单击变量管理器图标前的加号。从内部变量快捷菜单中选择“新建变量”。创建名称为“Var”具有该结构类型的 WinCC 变量。
4. WinCC 项目管理器的数据框架显示单个变量“Var”和结构变量“Var.OPCServer_Struct”。
5. 激活 WinCC 项目。

参见

如何在 WinCC OPC DA 客户机上组态结构 (页 105)

5.4.4.3 如何在 WinCC OPC DA 客户机上组态结构

简介

OPC DA 规范不支持结构。这样就不能使用 OPC 条目管理器建立结构。在本节中，我们将 WinCC OPC DA 服务器 WinCC 项目中已有的结构针对 WinCC OPC DA 客户机的 WinCC 项目进行组态。将在 WinCC OPC DA 客户机上组态用于访问 WinCC OPC DA 服务器上现有结构变量的 WinCC 变量。

要求

- 在 WinCC OPC DA 服务器的 WinCC 项目中创建结构和名称为“Var.OPCServer_Struct”的结构变量。
- 启用 WinCC OPC DA 服务器的 WinCC 项目。
- 将 OPC 通道添加到 WinCC OPC DA 客户机的 WinCC 项目。

说明

“OPC”通道

连接名称不支持 Unicode。请确保使用同一种语言对项目中的所有连接进行命名。打开计算机的控制面板为不支持 Unicode 的程序中所使用的语言设置代码页。

步骤

1. 在 WinCC OPC DA 客户机上，从结构类型快捷菜单中选择“新建结构类型”。将显示“结构属性”对话框。
2. 单击“新建元素”按钮并设置一个外部变量。命名元素，使其名称与在 OPC-DA 服务器的 WinCC 项目中的名称完全相同。单击“确定”关闭“结构属性”对话框。
3. 如果已创建与 OPC DA 服务器的连接，继续步骤 6。
如果没有创建任何连接，从“OPC”通道单元的快捷菜单中选择“新建连接”(New Connection)。将打开“连接属性”(Connection Properties)对话框。在相应域中输入连接的名称。
4. 单击“属性”按钮。将显示以连接名称为标题的对话框。对于与 WinCC V6 的连接，域“OPC 服务器名称”中的条目必须为“OPCServer.WinCC”。
5. 在域“在本计算机上启动服务器”中，输入要用作 WinCC OPC DA 服务器的计算机名称。单击“测试服务器”，检查与 WinCC OPC DA 服务器的连接。单击“确定”关闭对话框。
6. 从连接的快捷菜单中选择“新建变量”。将打开“变量属性”对话框。选择新创建的结构类型作为数据类型。
7. 在“变量属性”对话框中，单击“选择”按钮。将打开“地址属性”(Address properties)对话框。
在“条目名称”(Item Name)字段中，输入“Var.OPCServer_Struct”作为 WinCC OPC DA 服务器的结构变量名称。保持“访问路径”域中的条目不变。
8. 单击“确定”关闭所有打开的对话框。

参见

在 WinCC OPC DA 客户机上组态 OPC 通道 (页 96)

在 WinCC OPC DA 服务器上组态结构和结构变量 (页 104)

5.4.5 OPC DA 通讯受到干扰时的错误处理

5.4.5.1 OPC 通讯受到干扰时的错误处理

引言

通讯测试的过程与 WinCC 的使用方式无关。

WinCC 用作 OPC DA 服务器

在 WinCC OPC DA 客户机上使用通道诊断来确定是否可以建立与 OPC DA 服务器的连接。有关通道故障分析的更多信息，请参阅“故障检测”。

WinCC 用作 OPC DA 客户机

在 WinCC OPC DA 客户机上使用通道诊断来确定是否可以建立与 OPC DA 服务器的连接。有关通道故障分析的更多信息，请参阅“故障检测”。

参见

WinCC 用作 OPC DA 客户机，未建立连接。(页 111)

WinCC 用作 OPC DA 客户机，建立连接。(页 110)

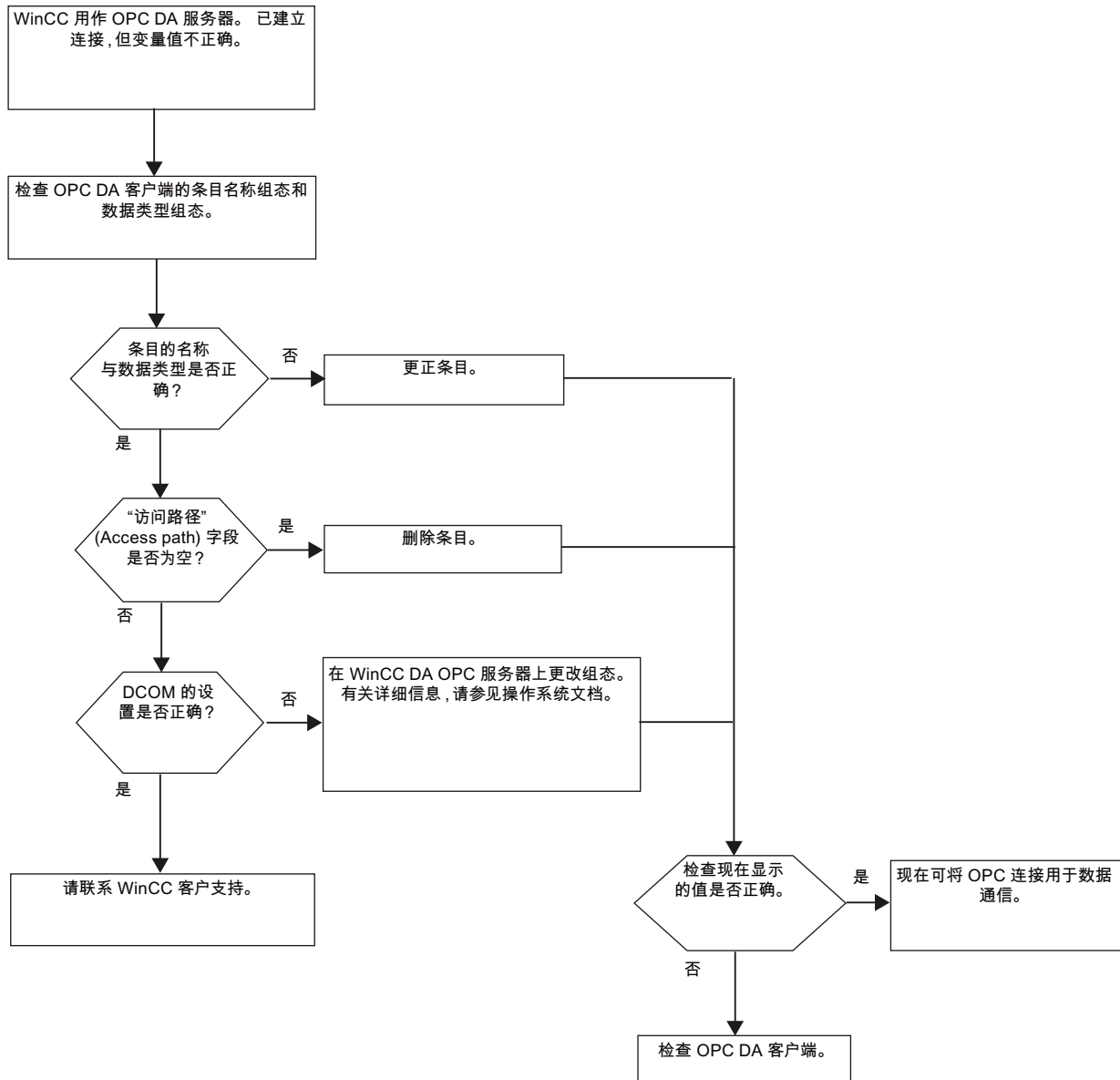
WinCC 用作 OPC DA 服务器，未建立连接。(页 109)

WinCC 用作 OPC DA 服务器，成功建立连接。(页 108)

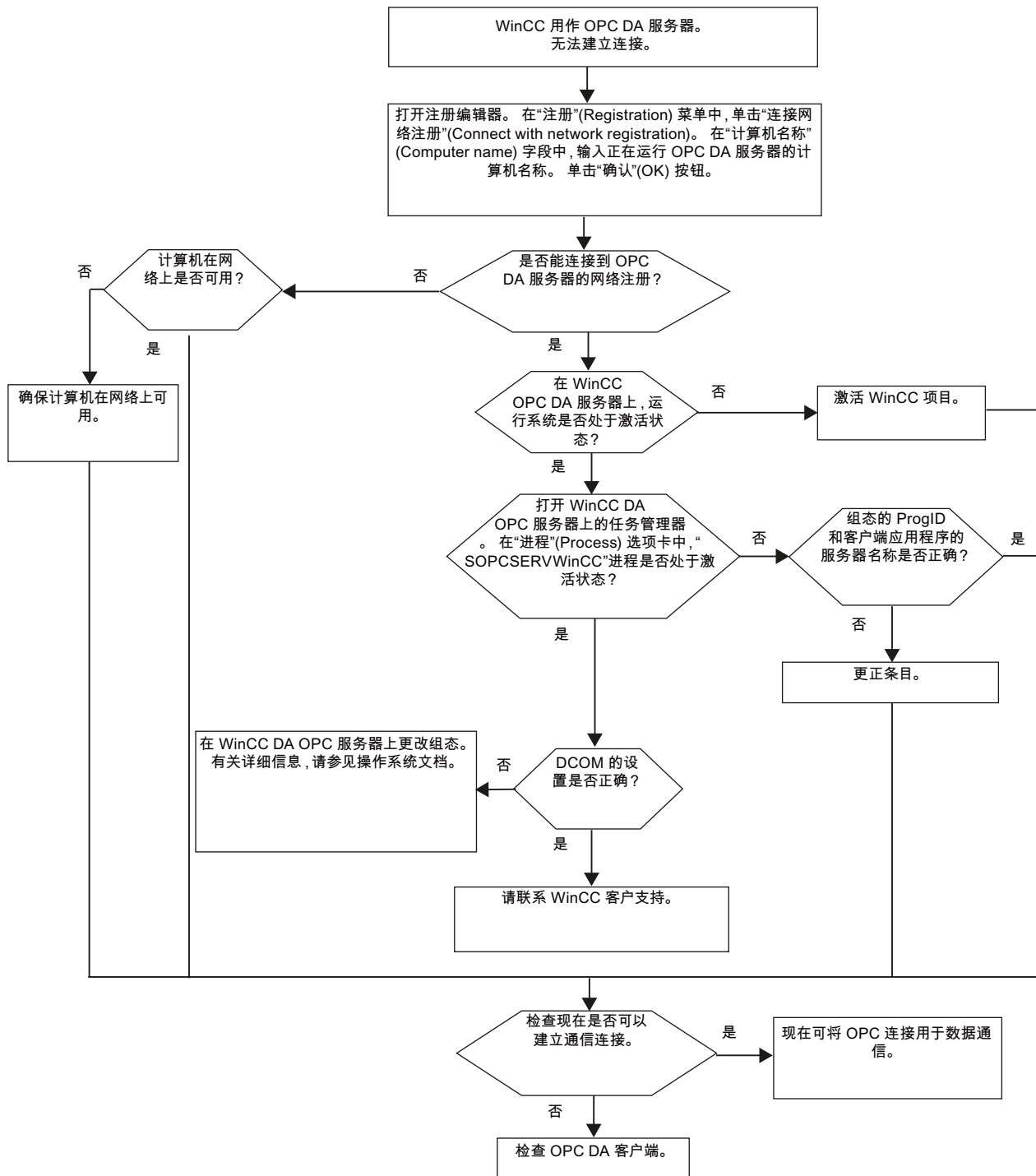
诊断“OPC”通道的可能性(页 550)

5.4.5.2 WinCC 作为 OPC DA 服务器

WinCC 用作 OPC DA 服务器，成功建立连接。

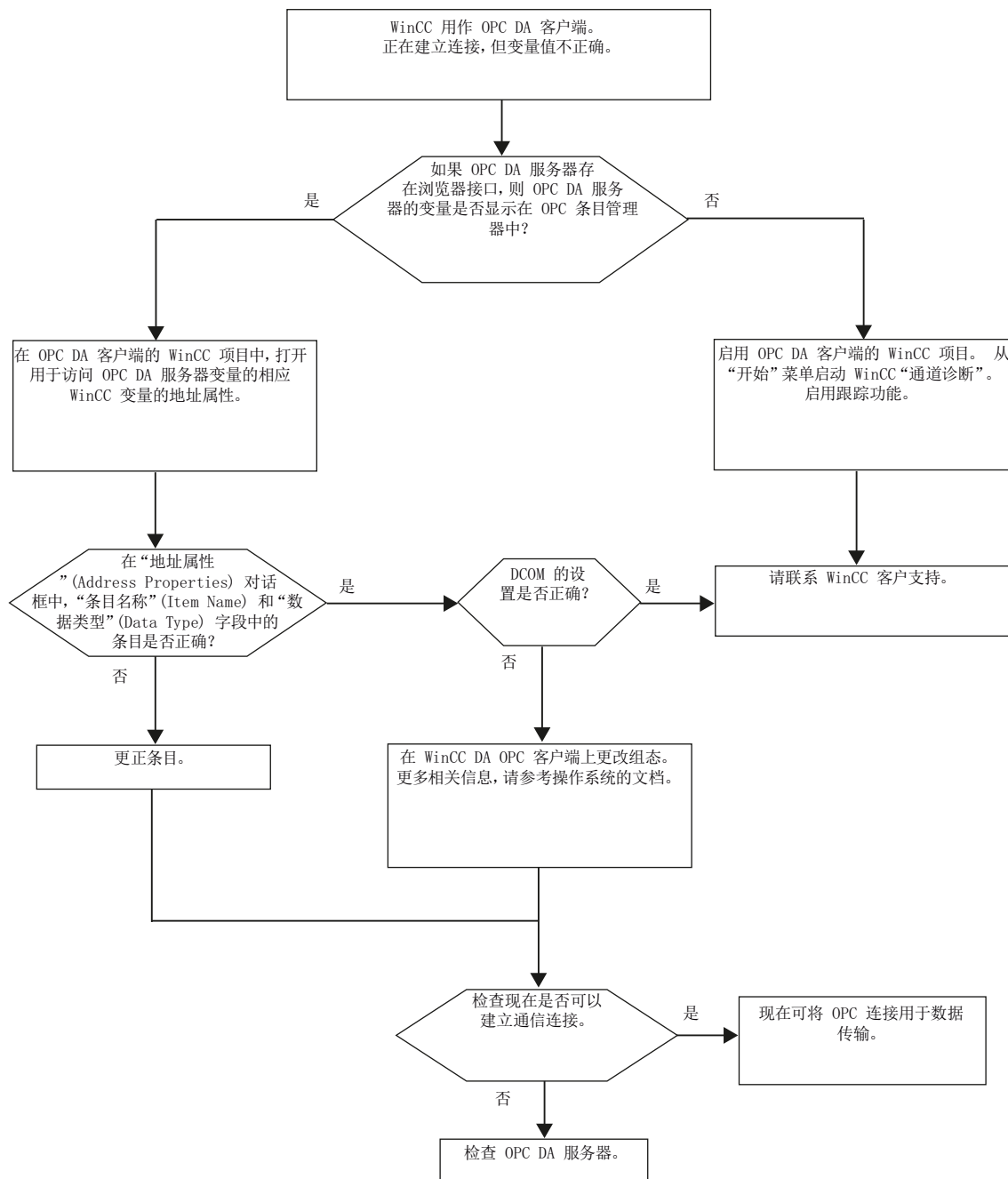


WinCC 用作 OPC DA 服务器，未建立连接。

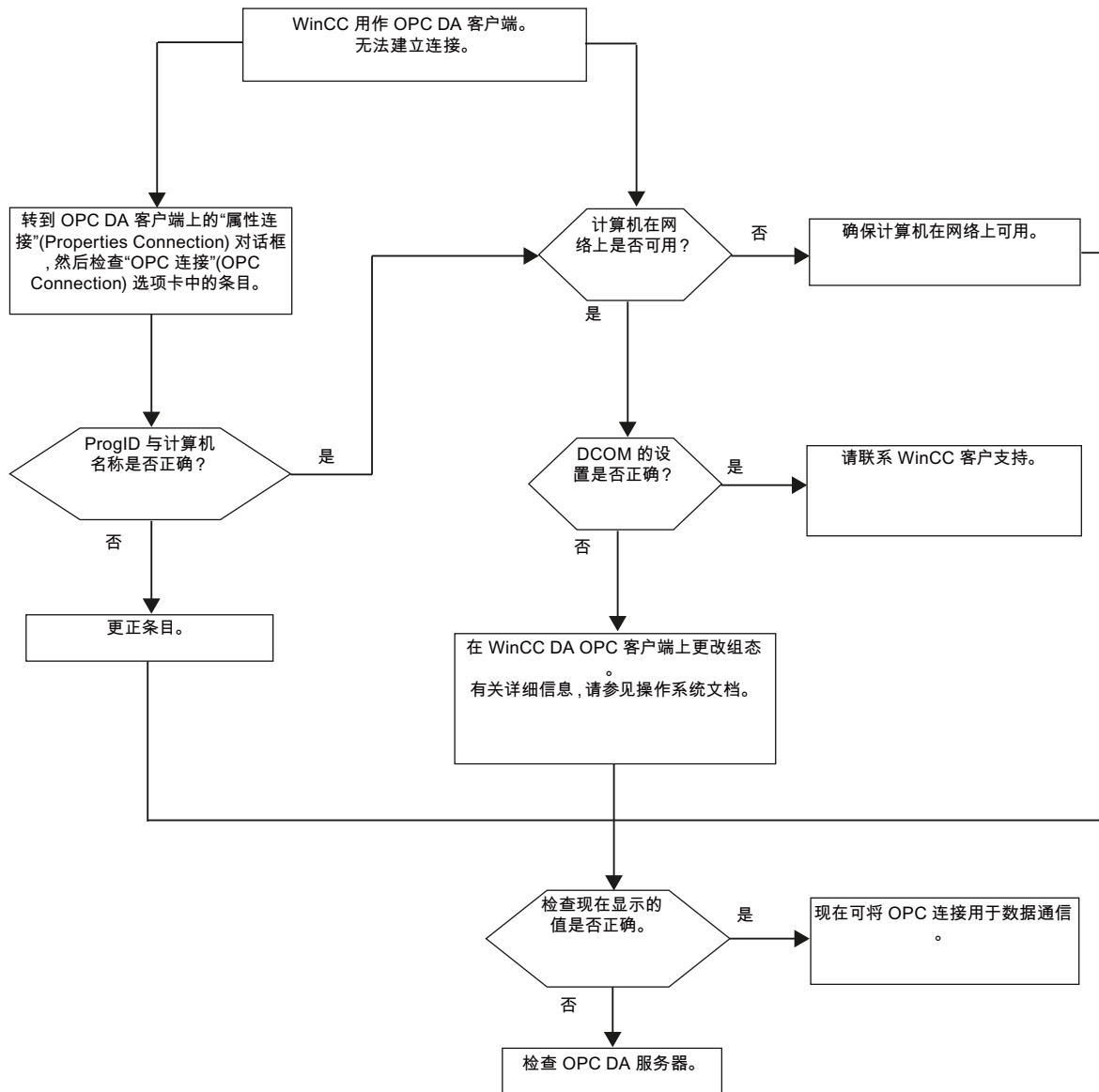


5.4.5.3 WinCC 作为 OPC DA 客户机

WinCC 用作 OPC DA 客户机，建立连接。



WinCC 用作 OPC DA 客户机，未建立连接。



5.5 WinCC OPC XML 客户机

5.5.1 WinCC OPC XML 客户机的功能

简介

OPC 通道不需要独立的通讯模块。OPC 通讯驱动程序可以作为 OPC XML 客户端使用。

为了将 WinCC 用作 WinCC OPC XML 客户端，必须将 OPC 通道添加到 WinCC 项目上。

WinCC OPC XML 客户端以 Web 页面形式为 OPC XML 服务器提供 OPC 过程数据。可以使用 HTTP 通过 Internet/Intranet 对 Web 页面进行访问。WinCC OPC XML 客户端请求数据时，由 Web 服务器自动启动 Web 服务。

为了使 WinCC OPC XML 客户端可以访问多个 OPC XML 服务器，必须与 WinCC 项目中的每个 OPC XML 服务器建立一个连接。

如果建立了与 WinCC OPC XML 服务器的通讯，将交换 WinCC 变量的值。在 WinCC OPC XML 客户端的 WinCC 项目中建立了一个连接，可以通过其处理对 WinCC OPC XML 服务器的访问。如果是 WinCC OPC XML 客户端，连接监视不被激活。

注意

使用多个网络适配器和激活的 TCP/IP 协议进行操作

要使用多个网络适配器和激活的 TCP/IP 协议进行操作，请阅读“WinCC 发行说明/与运行/网络技术及 UPS 有关的说明”和“组态/分布式系统/运行期间的系统行为/组合使用服务器与若干个网络适配器时的特殊通信功能”这两个部分中的信息。

说明

WinCC OPC 通道只能与状态为“OPC_STATUS_RUNNING”的 OPC 服务器建立连接。

说明

“OPC”通道

连接名称不支持 Unicode。请确保使用同一种语言对项目中的所有连接进行命名。打开计算机的控制面板为不支持 Unicode 的程序中所使用的语言设置代码页。

参见

所支持的 WinCC 数据类型概述 (页 93)

OPC 条目管理器 (页 89)

OPC 规范及兼容性 (页 123)

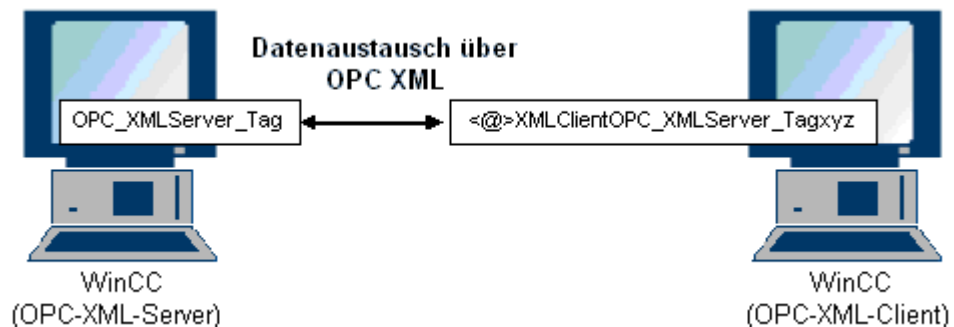
操作模式 (页 127)

5.5.2 如何通过 OPC 条目管理器访问 WinCC 变量

5.5.2.1 如何通过 OPC 条目管理器访问 WinCC 变量

引言

WinCC 之间建立 OPC 连接后，数据就会通过 WinCC 变量进行交换。WinCC OPC DA 客户机使用 OPC 连接来读取 WinCC OPC XML 服务器上的 WinCC 变量“OPC_Server_Tag”。为了简化过程，系统使用 OPC 条目管理器。



要求

- 两台带 WinCC 项目的计算机。
- 必须可以使用 HTTP 访问这两台计算机。

组态步骤

在 WinCC OPC XML 客户机的 WinCC 项目中，需要下列组态：

参见

在 WinCC OPC DA 客户机上组态 OPC 通道 (页 96)

通过 OPC 条目管理器组态访问 (页 114)

5.5.2.2 通过 OPC 条目管理器组态访问

简介

本节说明如何使用 OPC 条目管理器组态对 WinCC OPC XML 服务器上的“OPC_XMLServer_Tag”WinCC 变量的访问。

要求

- 在 WinCC OPC XML 服务器的 WinCC 项目中，组态一个名称为“OPC_XMLServer_Tag”、数据类型为“有符号 16 位数”的内部变量。
- 激活 WinCC OPC XML 服务器的 WinCC 项目。
- 必须将 OPC 通道添加到 WinCC OPC XML 客户机的 WinCC 项目。

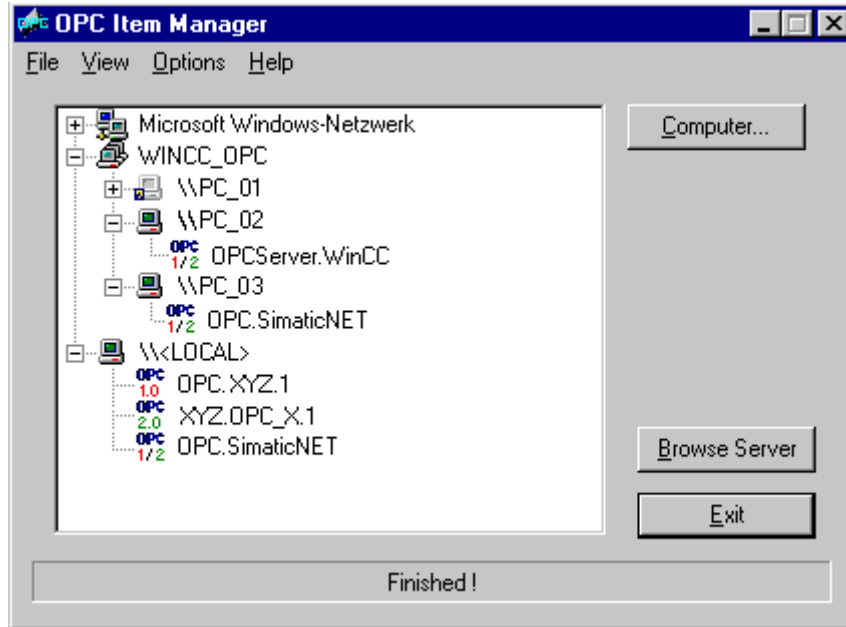
说明

“OPC”通道

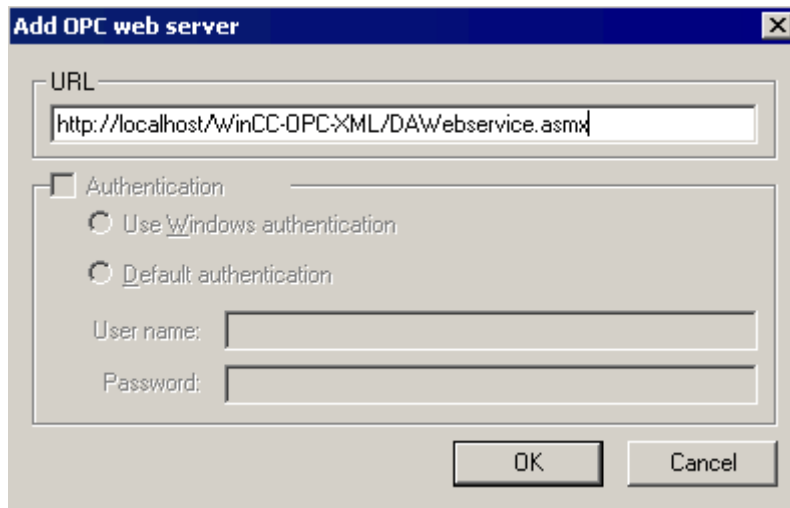
连接名称不支持 Unicode。请确保使用同一种语言对项目中的所有连接进行命名。打开计算机的控制面板为不支持 Unicode 的程序中所使用的语言设置代码页。

步骤

1. 在 WinCC OPC XML 客户机上，从“OPC 组 (OPCHN Unit#1)” 通道单元的快捷菜单中选择“系统参数”。将打开“OPC 条目管理器”(OPC Item Manager)。

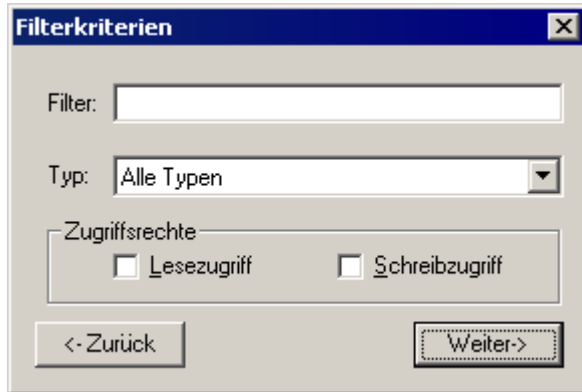


2. 单击“OPC Web 服务器”按钮。将打开“添加 OPC Web 服务器”对话框。以下列形式在“URL”域中输入 WinCC OPC XML 服务器的 URL: `<http://<xxx>/WinCC-OPC-XML/DAWebservice.asmx>`。使用运行 OPC XML Web 服务的计算机的 IP 地址或计算机名称替代 xxx。

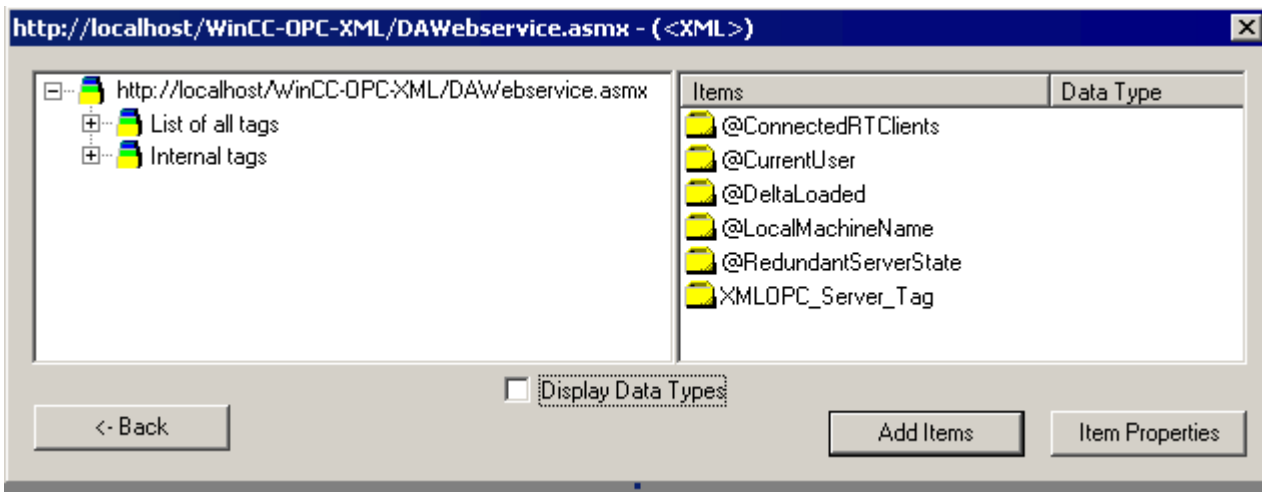


单击“确定”关闭对话框。

- 显示一个列表，从其中选择“/WinCC-OPC-XML/DAWebservice.asmx>”。单击“浏览服务器”按钮。将打开“过滤标准”(Filter criteria)对话框。



- 在“过滤标准”对话框中，单击“下一步->”按钮。将打开“http:// ...”对话框。

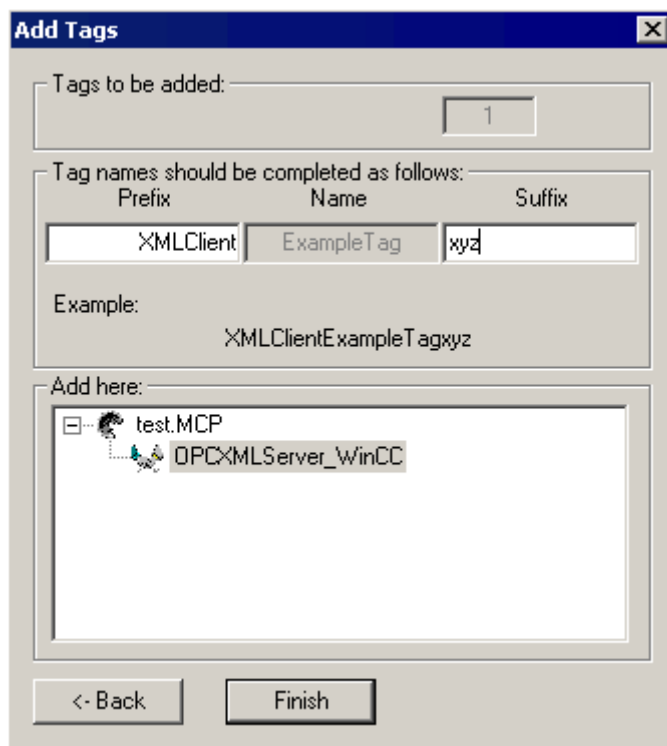


- 在“http:// ...”对话框中，选择 WinCC 变量“XMLOPC_Server_Tag”。单击“添加条目”按钮。
- 如果已创建与 WinCC OPC XML 服务器的连接，继续步骤 7。如果尚未组态连接，则会输出相应的消息。单击“是”(Yes)。将打开“新建连接”(New Connection)对话框。



为连接输入名称“OPCXMLServer_WinCC”。单击“确定”(OK)。

7. 将打开“添加变量”(Add Tags)对话框。
在前缀字段中输入“XMLClient_”，在后缀字段中输入“_xyz”。选择“OPCXMLServer_WinCC”连接。单击“Finish”（完成）。



8. 在“http:// ...”对话框中，单击“<-返回”。单击“退出”关闭 OPC 条目管理器。

参见

在 WinCC OPC DA 客户机上组态 OPC 通道 (页 96)

5.5.3 在不使用 OPC 条目管理器的情况下访问 WinCC 变量

引言

不支持浏览器功能的 OPC 服务器需要手动组态访问。下面通过一个 WinCC-WinCC OPC 连接的实例来说明 WinCC OPC XML 客户机上 WinCC 变量的组态。

说明

要在不使用 OPC 条目管理器的情况下访问 WinCC 变量，必须手动设置 ItemID。对 WinCC 变量进行寻址时，也可以指定符号计算机名称（服务器前缀）。条目标识号具有下列语法：`server_prefix::<@>WinCC 变量`。如果对本地 WinCC 项目的 WinCC 变量编址，ItemID 具有下列语法：`<@>WinCC 变量`。

组态步骤

在 WinCC OPC XML 客户机的 WinCC 项目中，需要下列组态：

1. 创建一个连接。
2. 在 WinCC OPC XML 客户机上组态“XMLClient_OPC_Var1_xyz”WinCC 变量，此变量用于访问 WinCC OPC DA 服务器 WinCC 变量。

要求

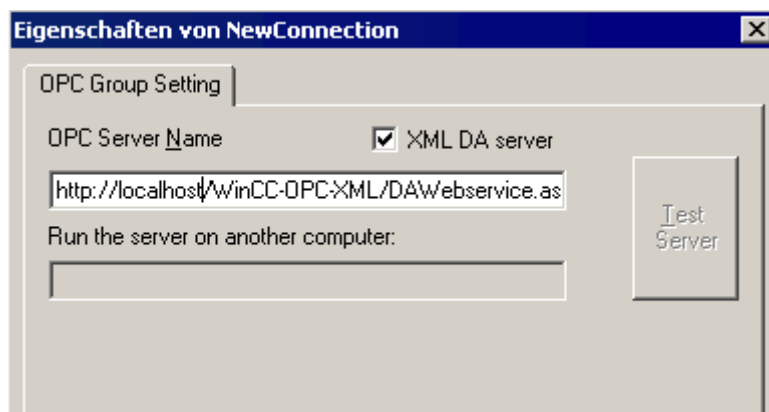
- 两台带 WinCC 项目的计算机。
- 必须可以使用 HTTP 访问这两台计算机。
- 在 WinCC OPC XML 服务器的 WinCC 项目中，组态一个名称为“XMLOPC_Server_Tag”、数据类型为“有符号 16 位数”的内部变量。
- 启用 WinCC OPC XML 服务器的 WinCC 项目。
- 必须将 OPC 通道添加到 WinCC OPC XML 客户机的 WinCC 项目。

说明

当在 OPC 通道中组态外部变量时，“变量属性”对话框的类型转换域中来自 WinCC 的预置值一定不能改变。正在处理的变量的数据类型在“地址属性”对话框的数据类型域中设置。

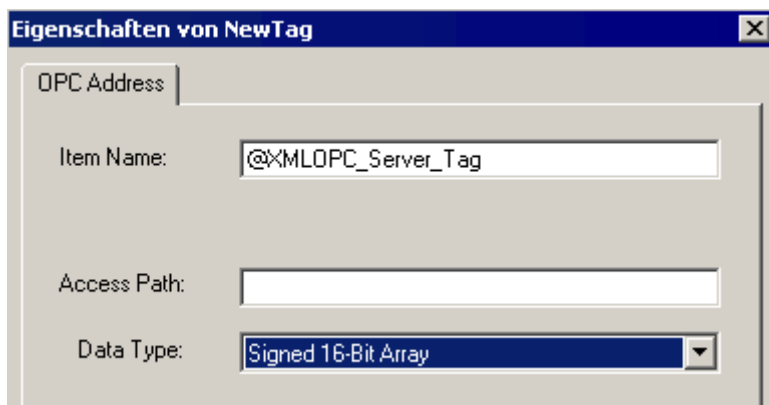
步骤

1. 在 WinCC OPC XML 客户机上，从“OPC Groups(OPCHN Unit#1)”通道单元的快捷菜单上选择“新建连接”。将打开“连接属性”对话框。在相应域中输入连接的名称。
2. 单击“属性”按钮。将显示以连接名称为标题的对话框。



选中“XML DA 服务器”复选框。如果是到 WinCC OPC XML 服务器的连接，“OPC 服务器名称”域必须包含 WinCC OPC XML 服务器的 URL。URL 具有下列语法：“http://<xxx>/WinCC-OPC-XML/DAWebservice.asmx”。使用运行 OPC-XML Web 服务的计算机的 IP 地址或计算机名称替代 xxx。

3. 从连接的快捷菜单中选择“新建变量”。将打开“变量属性”对话框。
4. 在“变量”域中输入名称“XMLClient_OPC_Var1_xyz”。将数据类型设置为“有符号 16 位数”。
5. 在“变量属性”对话框中，单击“选择”按钮。将显示以变量名为标题的对话框。



在“条目名称”域中输入符号“<@>”和 WinCC OPC XML 服务器的 WinCC 变量的名称。保持“访问路径”域中的条目不变。将数据类型设置为“有符号 16 位数”。

6. 单击“确定”关闭所有打开的对话框。

OPC - 开放式互连

6.1 OPC - 开放式互连

内容

使用 OPC 标准软件接口，各个生产商的设备和应用程序就能以统一的方式连接起来。

WinCC 可以用作 OPC 服务器或 OPC 客户机。“OPC”通道表示 WinCC 的 OPC 客户机应用程序。

本章描述了

- WinCC 的 OPC 服务器
- 如何在 WinCC 中使用 OPC
- 如何建立不同的 OPC DA 链接
- 如何对 WinCC 消息系统的访问进行组态
- WinCC 消息系统如何在 OPC A&E 上显示
- 如何建立对 WinCC 归档系统的访问

6.2 OPC 的功能

OPC 是一个独立于制造商的标准化软件接口，在自动化工程领域中用于数据交换。

通过 OPC 接口可在不同制造商的设备和应用程序之间实现标准连接。

OPC 基于 Windows 的 COM（组件对象模型）和 DCOM（分布式组件对象模型）技术。

OPC XML DA 额外提供了一个基于 XML、SOAP 和 HTTP Internet 标准的软件接口。

OPC UA（统一架构）是 OPC 的替代技术。OPC UA 与平台无关并且支持不同的协议作为通信媒介。

6.3 OPC 规范及兼容性

概述

访问 WinCC 中的以下对象时，OPC 指定了相应的接口：

- 过程值 (OPC Data Access 2.05a, 3.0; OPC XML Data Access 1.01; OPC UA 1.02)
- 归档的过程值 (OPC Historical Data Access 1.20; OPC UA Historical Access 1.02)
- 按时间排列的消息 (OPC Historical Alarms and Events v1.10)
- 消息 (OPC Alarms and Events 1.10; OPC UA Alarms and Conditions 1.02)

有关各个 OPC 规范的详细信息，请参见 OPC 基金会网站。

兼容性

这些规范的支持通常由 OPC Foundation 的“Compliance Test Tool”(CTT) 监视。加入“OPC Interoperability Workshops”可保证与其他制造商 OPC 产品的互用性。

提交的测试结果会在 OPC Foundation 网站上发布。要查看这些结果，请输入搜索项“OPC Self-Certified Products”。

参见

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.4 在 WinCC 中使用 OPC

简介

在 WinCC 中，服务器可供以下 OPC 接口使用：

- OPC Data Access / OPC XML Data Access: 访问 WinCC 数据主体
- OPC Historical Data Access: 访问 WinCC 归档系统
- OPC Alarms&Events: 访问 WinCC 消息系统
- OPC Unified Architecture: 访问 WinCC 数据主体和归档系统

默认情况下，WinCC 包含一个 OPC 通道。该 OPC 通道可作为客户端通过 OPC DA、OPC XML DA 或 OPC UA 访问相关 OPC 服务器。

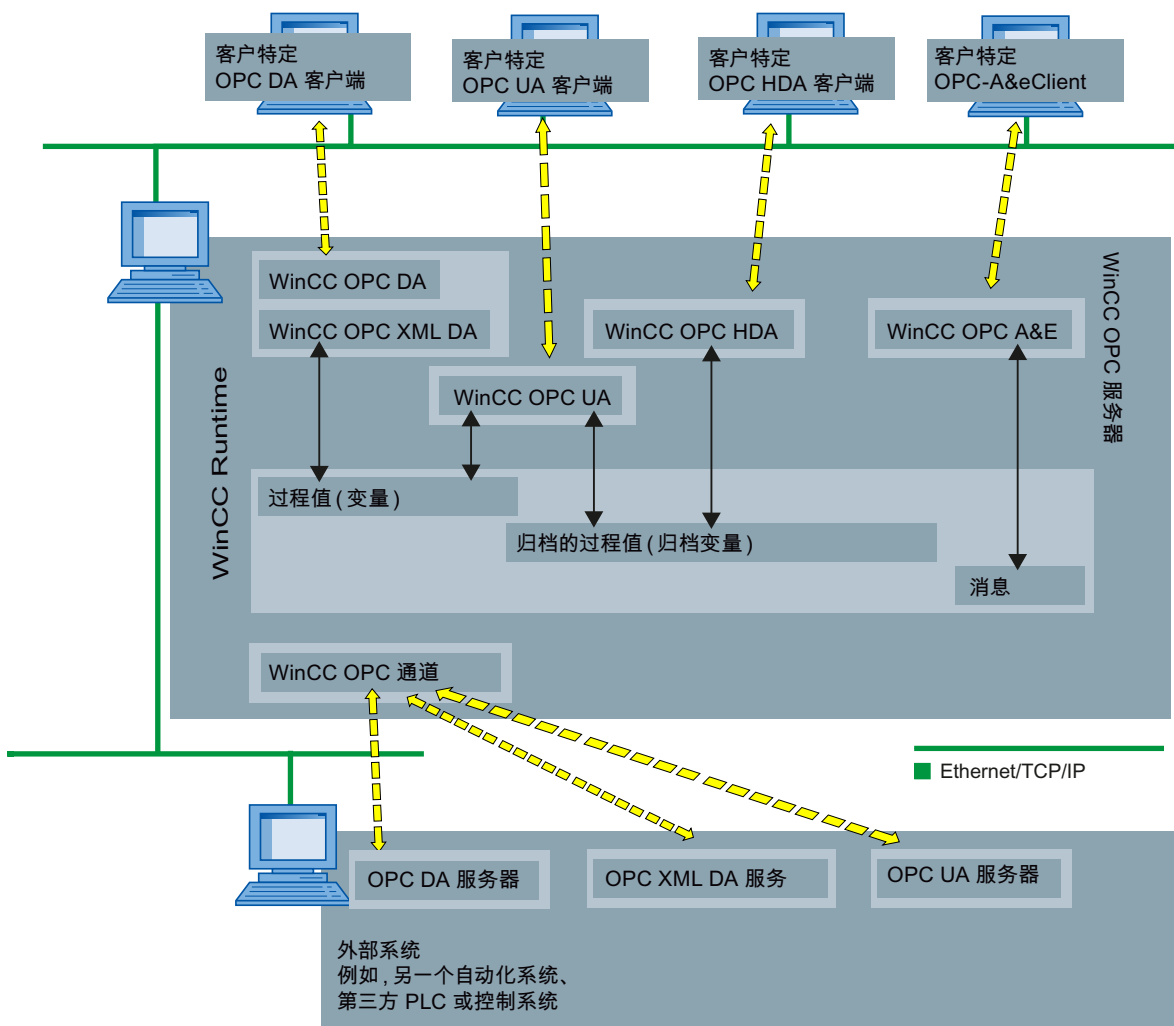
WinCC OPC 通信概念

通过 DCOM 完成 WinCC OPC 服务器和 OPC 客户端之间的数据交换。安装 WinCC 后，将正确组态 WinCC OPC 服务器的 DCOM 设置。

如果 WinCC OPC 服务器或客户端与外部 OPC 系统进行通信，则必须执行相应调整。必须在客户端上用户管理的“DCOM/工作区/COM 安全/访问权限/编辑默认值”中为用户输入“本地访问”和“远程访问”权限。

WinCC 的 OPC XML 服务器实现 Web 服务。这样便可以通过 Internet 访问 PC。因此需要定义适当的访问权限。

下图显示了 WinCC OPC 通信概念：



许可证

OPC 服务器	许可证
WinCC OPC DA 服务器	有效的 WinCC RT 许可证
WinCC OPC XML DA 服务器	有效的 WinCC RT 许可证
WinCC OPC UA 服务器	WinCC 选件连通性软件包
WinCC OPC HDA 服务器	
WinCC OPC A&E 服务器	

6.5 如何组态 Windows 以使用 WinCC OPC

简介

OPC 客户机和 OPC 服务器是 DCOM 应用程序。分布式 DCOM 应用程序只能使用同一用户帐户运行。因此，OPC 服务器必须识别 OPC 客户机的用户帐户，反之亦然。如果 WinCC OPC 服务器和 WinCC OPC 客户机一起使用，则在安装时已保证组态正确。

如果使用外部 OPC 服务器或客户机，请声明用户帐户。

有关授予用户权限的更多信息，请参阅 Windows 文档。

要求

作为管理员登录到 WinCC OPC 服务器和 OPC 客户机工作站来组态用户权限。

步骤（以 Windows 7 为例）

1. 转到“控制面板 > 系统和安全 > 管理工具 > 计算机管理 > 本地用户和组”(Control Panel > System and Security > Administrative Tools > Computer Management > Local Users and Groups)。
2. 在“用户”(Users) 快捷菜单中，选择“新用户”(New User)。在“新用户”(New User) 对话框中，输入通信伙伴的用户帐户详细信息。单击“创建”，关闭对话框。
3. 单击“用户”图标。双击相关的用户。将显示该用户的“属性”对话框。
4. 单击“成员”(Member Of) 选项卡。单击“添加”(Add)。将打开“选择组”对话框。
5. 添加组“Users”。如果是安装了 WinCC 的计算机，还要添加组“SIMATIC HMI”。单击“确定”关闭所有打开的对话框。

如何调整 Windows 防火墙设置

安装 WinCC 后，将正确组态 WinCC OPC 服务器的 Windows 防火墙设置。

如果 OPC 客户端访问不同子网中的 OPC 服务器，则必须调整 OPC 服务器所允许网络区域的组态。

6.6 WinCC OPC XML DA 服务器

6.6.1 操作模式

简介

WinCC 的 OPC XML DA 服务器是 Microsoft Internet 信息服务 (IIS) 的 Web 服务的一部分。

WinCC OPC XML DA 服务器以 Web 页面形式为 OPC XML 客户端提供 OPC 过程数据。可以使用 HTTP 通过 Internet 访问 Web 页面。WinCC OPC XML DA 服务器的地址是：
<http://<xxx>/WinCC-OPC-XML/DAWebservice.asmx>

WinCC OPC XML DA 服务器在 WinCC 中不可见。OPC XML 客户端请求数据时，Web 服务器便自动启动 Web 服务。

为了建立成功的 OPC 通讯，必须遵守以下内容：

- 必须激活 WinCC OPC XML DA 服务器的 WinCC 项目。
- 必须能够通过 HTTP 访问 WinCC OPC XML DA 服务器的计算机。

许可

为了操作 WinCC OPC XML DA 服务器，必须在每台作为 OPC XML 服务器的 WinCC 计算机上安装下列许可证：

- 有效的 WinCC RT 许可证
- WinCC 选件连通性软件包

“字符串”类型变量的特殊功能

如果使用逻辑上代表浮点值的“字符串”类型的变量，OPC 客户端要写入和读取时可能会出现

描述

OPC 客户端写入到字符串变量，并不会以字符串形式输入新值，而是作为浮点、双精度或十进制处理。

问题

表示小数值（德语）的逗号可能会丢失。这会导致错误的值。

如果以浮点、双精度或十进制格式请求读取值，还会影响对字符串变量的读访问。

纠正方法

只使用相应的浮点型变量用于浮点值。仅以字符串格式访问字符串变量。

参见

WinCC OPC XML 客户机的功能 (页 112)

所支持的 WinCC 数据类型概述 (页 93)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.6.2 安装

简介

OPC-XML 允许通过 Internet 访问过程变量。为了操作 OPC XML，必须安装各种附加的软件组件。

说明

必须遵守此处所述安装步骤的顺序。否则，在安装期间可能会出现問題。

要求

- 安装 Internet 信息服务器 (IIS)
- 已通过 WinCC 产品 DVD 安装 Microsoft .NET Framework V4.0
- 使用 WinCC 安装程序安装 WinCC OPC XML 服务器

安装 Internet 信息服务器 (IIS)

在 Windows Server 2008 R2/2012 R2 中，在服务器管理器中使用“Webserver (IIS)”角色组态相关角色服务中的设置。

选择以下设置：

- Web 管理工具：
 - IIS 管理服务
 - IIS 管理控制台
 - IIS 管理脚本和工具
- WWW 服务 > 常用 HTTP 功能或共享 HTTP 功能：
 - 默认文档
 - 静态内容
- WWW 服务 > 应用程序开发功能：
 - .NET 可扩展性
 - ASP.NET
 - ISAPI 扩展项
 - ISAPI 过滤器
- WWW 服务 > 安全：
 - 请求过滤
 - 基本验证
 - Windows 验证

说明

安装 Microsoft Internet 信息服务 (IIS) 时务必同时安装 ASP.NET

安装 Microsoft Internet 信息服务 (IIS) 时，务必安装 ASP.NET。

说明

WinCC OPC XML DA 服务器的 Web 服务通过端口 80 (HTTP) 进行通信。

确保针对所需的网络区域选择并激活“WWW 服务 (HTTP)” 防火墙规则。

安装 WinCC OPC XML 服务器

在安装 WinCC 过程中，可选择 WinCC OPC XML DA 服务器。有关更多信息，请参阅 WinCC 信息系统中的“安装注意事项 > 安装 WinCC”部分。

安装期间必须进行下列设置：

- 创建虚拟目录“WinCC-OPC-XML”
- 为目录定义访问权限

6.6.3 使用 IIS 进行安全设置

简介

Internet 信息服务使得可以通过 Internet 访问 PC。因此需要定义适当的访问权限。

说明

如果对下列设置有任何疑问或遇到任何问题，请联系您的 Intranet/Internet 管理员。

步骤（以 Windows 7 为例）

1. 转到“控制面板 > 系统和安全 > 管理工具 > 计算机管理 > 服务和应用程序 > Internet 信息服务 (IIS) 管理器”(Control Panel > System and Security > Administrative Tools > Computer Management > Services and Applications > Internet Information Services (IIS) Manager)。
2. 选择虚拟目录“WinCC-OPC-XML”。
3. 在功能视图中打开“验证”(Authentication) 功能。
随即打开一个包含已安装的验证方法的列表。
4. 要启用匿名访问，请激活“匿名验证”(Anonymous authentication)。
5. 要启用验证访问，请激活“Windows 验证”(Windows authentication)。
6. 关闭所有打开的对话框。

说明

限制和风险

请注意设置访问选项时 Microsoft 提示的限制和风险。

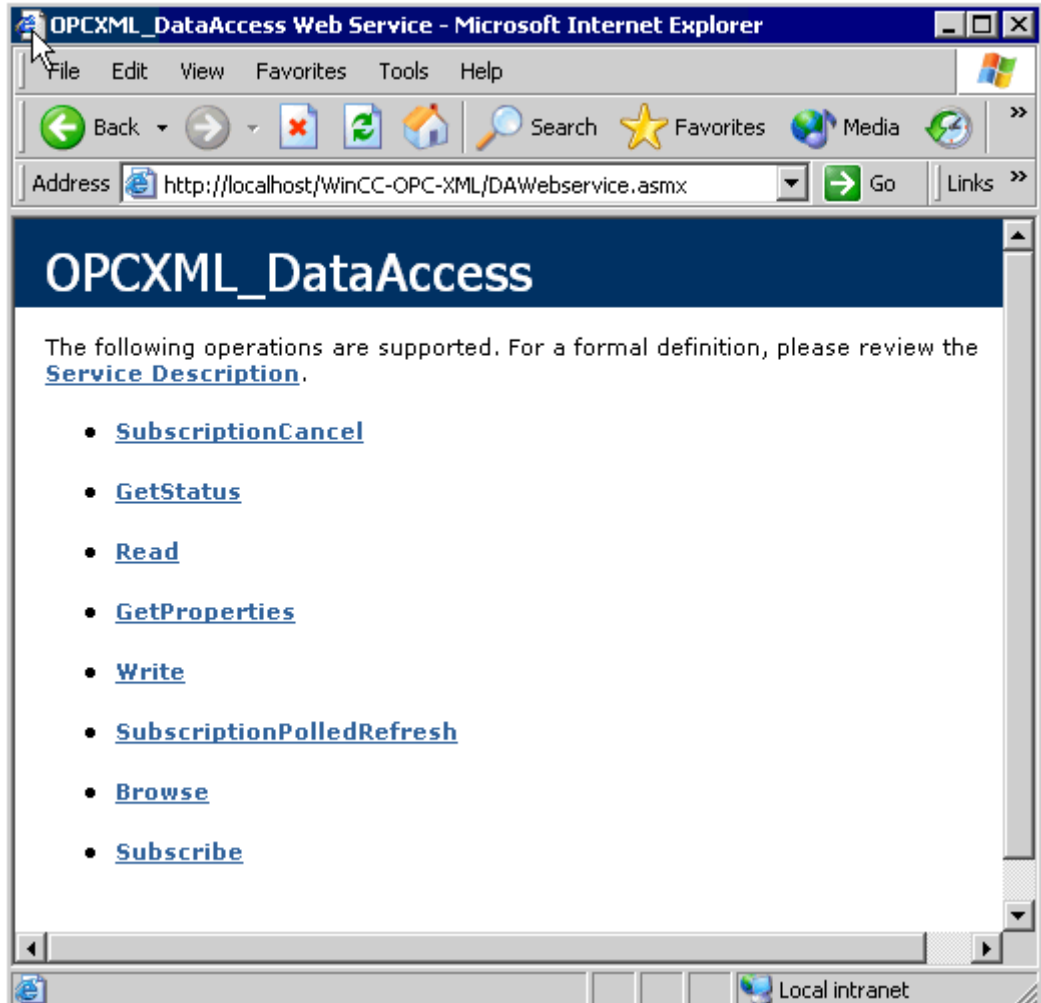
6.6.4 测试安装

简介

OPC XML-DA 使得 OPC 过程数据可显示为 Web 页面。可以使用 HTTP 通过 Internet 访问 Web 页面。下列部分说明如何测试安装。

步骤

1. 在作为 WinCC OPC XML 服务器的计算机上启动 Internet Explorer。
2. 在地址栏中输入 URL“<http://localhost/WinCC-OPC-XML/DAWebservice.asmx>”。使用 <ENTER> 键确认输入。
3. 显示 OPC XML DA 功能请求时，安装成功完成。



6.7 WinCC OPC DA 服务器

6.7.1 WinCC OPC DA 服务端的功能

简介

WinCC OPC DA 服务器支持 OPC 数据访问规范 2.05a 和 3.00。这已经由兼容性测试确认。

WinCC OPC DA 服务端是一个 DCOM 应用程序。WinCC OPC DA 服务端使用该接口向 WinCC 客户端提供关于 WinCC 变量的所需信息。

如果 WinCC OPC DA 客户端正在通过连接访问 WinCC OPC DA 服务端，则服务端处于活动状态。为了建立成功的 OPC 通信，必须遵守以下内容：

- WinCC OPC DA 服务端的 WinCC 项目必须激活。
- 运行 WinCC OPC DA 服务端的计算机必须能够通过其 IP 地址访问。

安装

在安装 WinCC 过程中，可选择 WinCC OPC DA 服务端。安装后，WinCC OPC DA 服务端无需进一步组态便立即可使用。

可以在 WinCC 服务器或 WinCC 客户端上安装 WinCC OPC DA 服务端。

关于组态的注意事项

- 可在 WinCC 项目中将变量集中到变量组中。变量名称不能与组名称相同。
- 在 WinCC 中发起的每个写入请求（例如，通过 VBScript 或对象“IO 字段”）都总是被视为同步“写入”调用。WinCC OPC DA 服务器为此会使用“IOPCSyncIO::Write”接口。WinCC OPC DA 通道中不支持异步写入机制。

说明

如果计算机中“连接 -> 局域网设置”(Connections -> LAN Settings) 下的 Internet 选项被设置为自动检测设置，则通过 Web 服务访问 OPC DA 将花费较长的时间。

参见

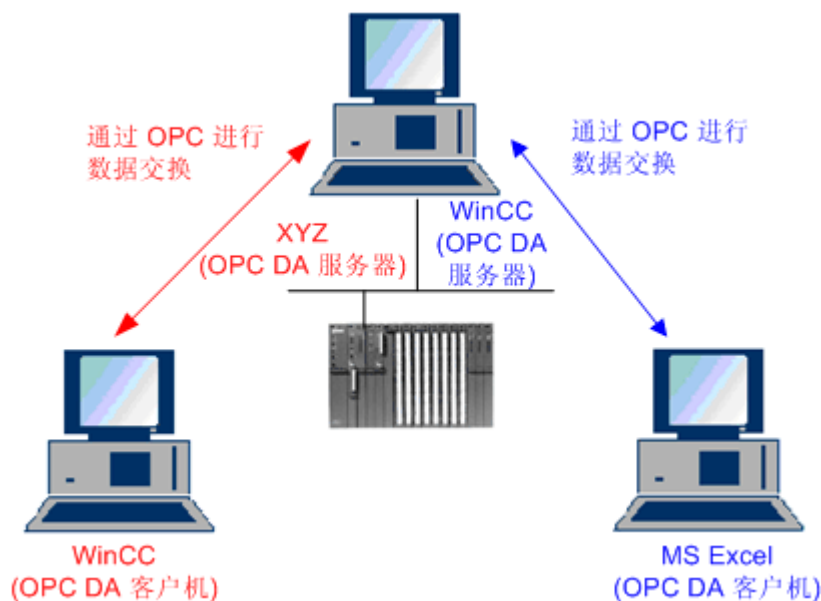
- 查询 OPC DA 服务器名称 (页 134)
- 使用多个 OPC DA 服务器 (页 133)
- WinCC 至 WinCC 的连接实例 (页 136)
- WinCC - SIMATIC NET FMS OPC 服务器连接的实例 (页 140)
- WinCC - SIMATIC NET S7 OPC 服务器连接的实例 (页 142)
- WinCC - Microsoft Excel 连接的实例 (页 148)
- 所支持的 WinCC 数据类型概述 (页 93)
- www.opcfoundation.org (<http://www.opcfoundation.org>)

6.7.2 使用多个 OPC DA 服务器

简介

在一台计算机上可以安装一个以上的 OPC DA 服务端，且任意数目的服务端都可并行地工作。通过这种方式，WinCC 的 OPC DA 服务端和另一（第三方）供应商的 OPC DA 服务端可在同一台计算机上彼此独立地运行。

WinCC OPC DA 客户端能通过第三方供应商的 OPC 服务端，访问自动化设备的过程数据。Microsoft Excel 的 OPC DA 客户端能使用 WinCC OPC DA 服务端访问 WinCC 数据。



6.7 WinCC OPC DA 服务器

有不同生产商提供的许多 OPC DA 服务端可用。其中每个 OPC DA 服务端都有唯一的名称 (ProgID) 以便识别。OPC DA 客户端必须使用该名称对 OPC 服务端进行寻址。

OPC 条目管理器用来查询 OPC DA 服务端的名称。WinCC V7 的 OPC DA 服务端的名称为：“OPCServer.WinCC”。

参见

查询 OPC DA 服务器名称 (页 134)

6.7.3 查询 OPC DA 服务器名称

简介

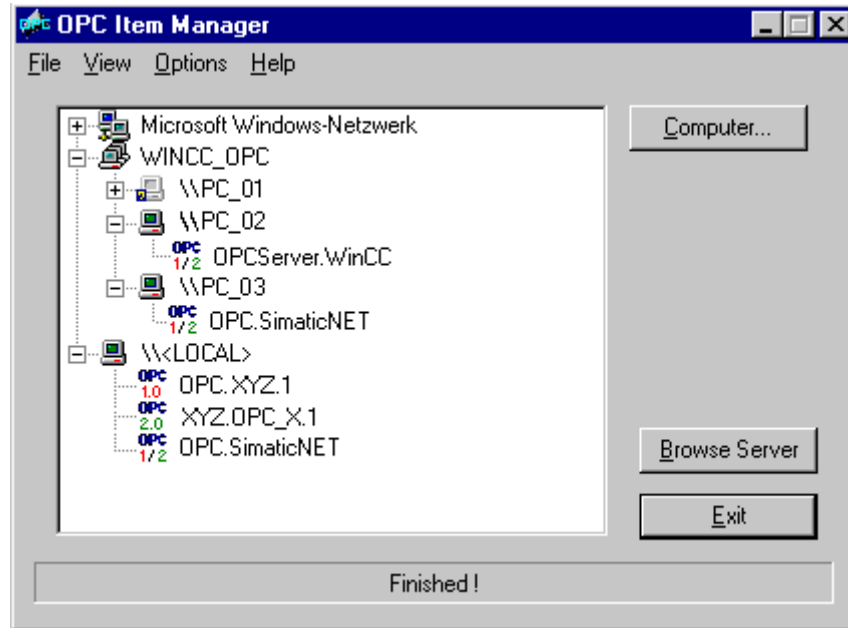
多个 OPC DA 服务器可安装在一台计算机上。OPC 条目管理器在选择窗口中显示可用于工作站的 OPC DA 服务器的名称。这些 OPC DA 服务器能够在同一计算机上运行或在已联网的计算机上运行。

要求

在 WinCC OPC DA 客户端的 WinCC 项目中添加“OPC”通道。

步骤

1. 在 WinCC OPC DA 客户端上，从“OPC 组 (OPCHN Unit#1)” 通道单元的快捷菜单中选择“系统参数”。将打开“OPC 条目管理器”。
2. 在 OPC 条目管理器的浏览窗口中，选择要访问的计算机名称。
3. OPC 条目管理器在选择窗口中显示可用于计算机的 OPC DA 服务器的名称。



参见

OPC 条目管理器 (页 89)

6.7.4 OPC DA 连接的实例

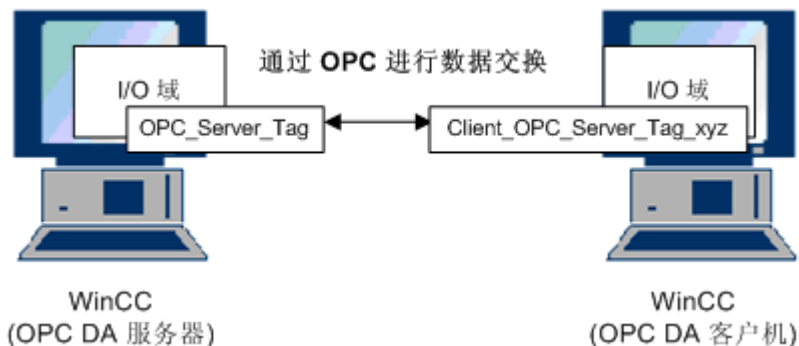
6.7.4.1 WinCC - WinCC 连接

WinCC 至 WinCC 的连接实例

引言

当建立 WinCC - WinCC 连接时，WinCC OPC DA 服务器和客户机之间的数据交换通过 WinCC 变量“OPC_Server_Tag”进行。客户机上的 WinCC 变量“Client_OPC_Server_Tag_xyz”读取服务器上的 WinCC 变量“OPC_Server_Tag”。如果 WinCC OPC 服务器上的“OPC_Server_Tag”变量的数值改变，WinCC OPC DA 客户机上的 WinCC 变量“Client_OPC_Server_Tag_xyz”的值也改变。客户机上的改变也会在服务器上反映出来。

变量值显示在两台计算机的 I/O 域中。



要求

- 两台带 WinCC 项目的计算机。
- 这两台计算机必须能够通过其 IP 地址被访问。

组态过程

建立 WinCC - WinCC 的连接需要以下组态：

1. 组态 WinCC OPC DA 服务器上的 WinCC 项目
2. 组态 WinCC OPC DA 客户机上的 WinCC 项目

参见

- 如何组态 WinCC OPC DA 服务器上的 WinCC 项目 (页 137)
- 组态 WinCC OPC DA 客户机上的 WinCC 项目 (页 138)

如何组态 WinCC OPC DA 服务器上的 WinCC 项目

引言

本节中介绍在 WinCC OPC DA 服务器上的 WinCC 项目中创建一个 WinCC 变量并显示在 I/O 域内。

步骤

1. 从 WinCC OPC DA 服务器上“内部变量”图标的快捷菜单中选择“新建变量”。创建一个类型为“有符号 16 位数”，名为“OPC_Server_Tag”的新变量。
2. 启动“图形编辑器”，打开一个新的画面。
3. 在画面中添加一个 I/O 域。从“智能对象”下的对象列表中选择“I/O 域”对象。将打开“I/O 域组态”对话框。



4. 在“变量”域中，输入名称“OPC_Server_Tag”。
5. 设置更新周期为“2 秒”，域类型为“I/O 域”。
6. 单击“确定”关闭对话框并保存画面。
7. 单击图形编辑器中的“激活”按钮以启用 WinCC 项目。

参见

- 组态 WinCC OPC DA 客户机上的 WinCC 项目 (页 138)

组态 WinCC OPC DA 客户机上的 WinCC 项目

引言

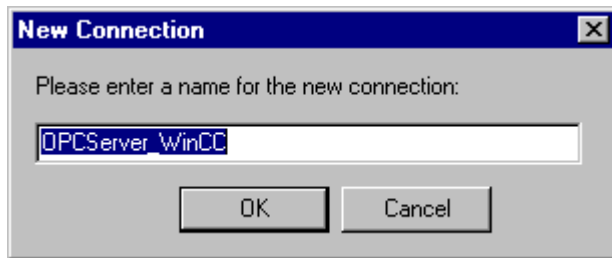
在本节中，将在 WinCC OPC DA 客户机上创建 WinCC 变量，以读取 WinCC OPC DA 服务器上的 WinCC 变量。变量值显示在 I/O 域中。

要求

- 在 WinCC OPC DA 客户机的 WinCC 项目中添加“OPC”通道。
- 在 WinCC OPC DA 服务器的 WinCC 项目中，组态一个数据类型为“有符号 16 位数”，名为“OPC_Server_Tag”的内部变量。
- 启用 WinCC OPC DA 服务器的 WinCC 项目。

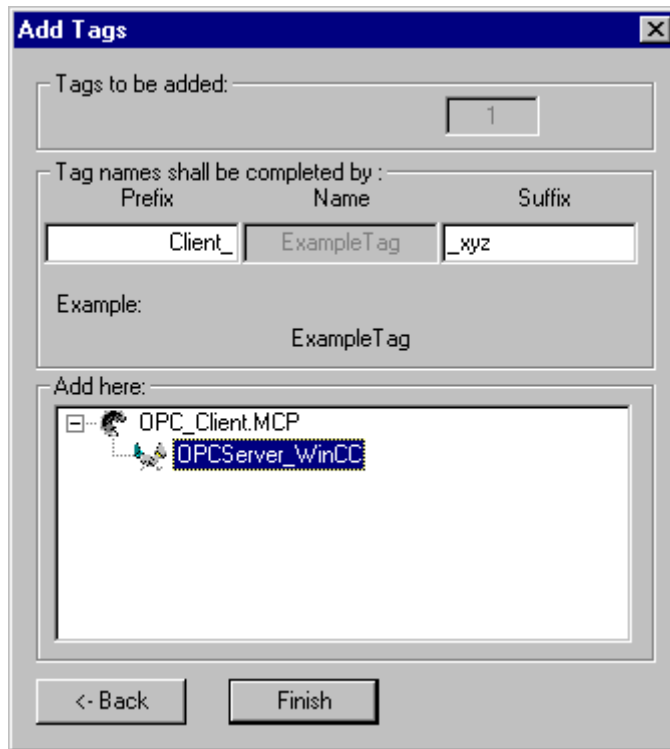
步骤

1. 在 WinCC OPC DA 客户机上，从“OPC 组 (OPCHN Unit#1)”通道单元的快捷菜单中选择“系统参数”。将打开 OPC 条目管理器。
2. 从选择对话框中选择要用作 OPC DA 服务器的计算机名称。从列表中选择“OPCServer.WinCC”。单击“浏览服务器”按钮。将打开“过滤标准”对话框。
3. 在“过滤标准”对话框中，单击“下一步->”按钮。在“OPCServer.WinCC ...”对话框中选择“OPC_Server_Tag”变量。单击“添加条目”按钮。
4. 如果到 OPC DA 服务器的连接已经存在，继续步骤 5。如果尚未组态连接，则会显示相应的消息。单击“是”。显示“New Connection”（新建连接）对话框。



输入“OPCServer.WinCC”作为连接的名称。单击“OK”（确定）。

- 显示“Add Tags”（添加变量）对话框。
在前缀域中输入“Client_”，在后缀域中输入“_xyz”。选择连接“OPCServer_WinCC”。单击“Finish”（完成）。



- 在“OPCServer.WinCC ...”对话框中，单击“<-返回”按钮。在“OPC 条目管理器”中，单击“退出”关闭 OPC 条目管理器。
- 启动“图形编辑器”，打开一个新的画面。在画面中添加一个 I/O 域。从“智能对象”下的对象列表中选择“I/O 域”对象。将打开“I/O 域组态”对话框。
- 在“变量”域中输入名称“Client OPC_Server_Tag_xyz”。将更新周期设置为“2 秒”。将域类型设置为“I/O 域”。关闭对话框并保存画面。单击图形编辑器中的“激活”按钮以启用 WinCC 项目。
- 在 WinCC OPC DA 服务器和客户机上的 I/O 域中，将显示组态变量的数值。在 WinCC OPC DA 服务器的 I/O 域中输入新的数值。新数值将显示在 WinCC OPC DA 客户机上的 I/O 域内。

参见

如何组态 WinCC OPC DA 服务器上的 WinCC 项目 (页 137)

在 WinCC OPC DA 客户机上组态 OPC 通道 (页 96)

6.7.4.2 WinCC - SIMATIC NET FMS OPC 服务器的连接

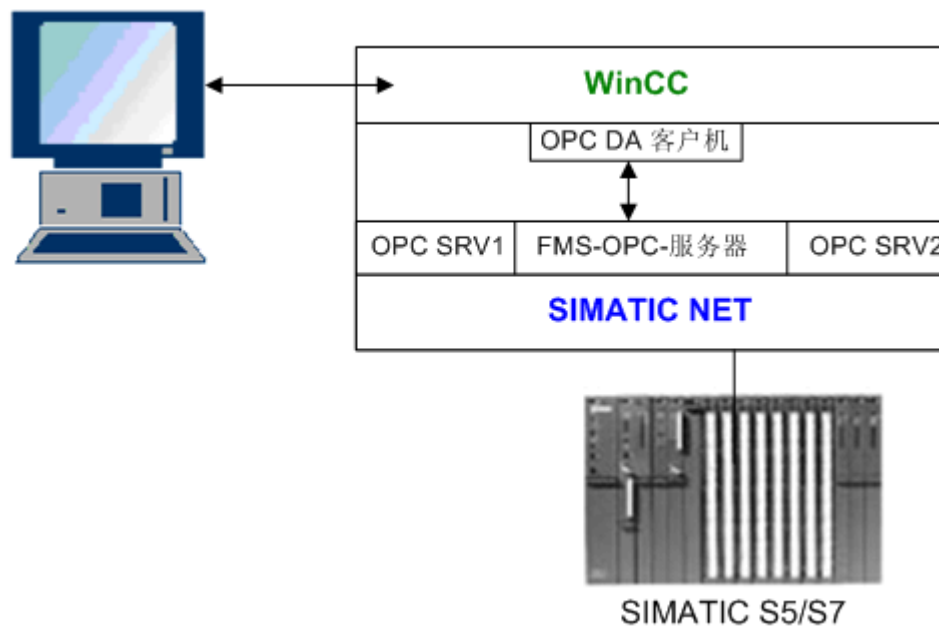
WinCC - SIMATIC NET FMS OPC 服务器连接的实例

简介

在 SIMATIC NET 的安装期间，可以选择要安装的 OPC 服务端。在以下实例中，将组态 WinCC 和 SIMATIC NET FMS OPC 服务端之间的连接。通过 SIMATIC NET FMS OPC 服务端，WinCC 可使用来自自动化设备的数据。

在本实例中，WinCC 用作 WinCC OPC DA 客户端。OPC 条目管理器显示为自动化设备组态的对象列表索引。

变量的当前值显示在一个 I/O 域中。SIMATIC NET FMS OPC 服务端上的变量值改变时，新的数值将会在 WinCC OPC DA 客户端的过程画面中反映出来。反之，在 I/O 域中输入的数值会发送到自动化设备上。



要求

- 安装有 WinCC、SIMATIC NET 软件的计算机。
- 已组态 SIMATIC NET FMS OPC 服务端。有关 SIMATIC NET S7 OPC 服务端设置的更多信息，请参阅 SIMATIC NET 文档。

组态步骤

在 WinCC OPC DA 客户端的 WinCC 项目中，需要下列组态：

1. 组态 WinCC - SIMATIC NET FMS OPC 服务端的连接

通信手册

通信手册包含通道组态的更多信息和扩展实例。本手册可在 Internet 上下载，地址如下：

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

搜索项目号：

- A5E00391327

如何组态 WinCC - SIMATIC NET FMS OPC 服务器的连接

引言

本节中，在 WinCC OPC DA 客户机的 WinCC 项目中组态一个访问 FMS 索引的 WinCC 变量。变量值显示在 I/O 域中。

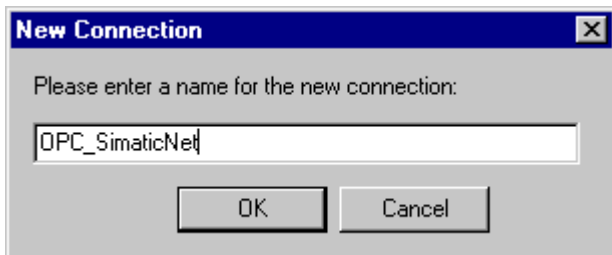
要求

- 在 WinCC OPC DA 客户机的 WinCC 项目中添加“OPC”通道。

步骤

1. 在 WinCC OPC DA 客户机上，从“OPC 组 (OPCHN Unit#1)”通道单元的快捷菜单中选择“系统参数”。将打开 OPC 条目管理器。
2. 从选择对话框中选择要用作 OPC DA 服务器的计算机名称。从列表中选择“OPC.SIMATICNet”。单击“浏览服务器”按钮。将打开“过滤标准”对话框。
3. 在“过滤标准”对话框中，单击“下一步->”按钮。“OPC.SIMATICNet..”对话框打开。所有组态的 FMS 索引将显示在选择列表中。选择索引。单击“添加条目”按钮。

4. 如果到 SIMATIC NET FMS OPC 服务器的连接已经存在，继续步骤 5。
如果尚未组态连接，则会显示相应的消息。
单击“是”。显示“New Connection”（新建连接）对话框。



输入“OPC_SlimaticNET”作为连接的名称。单击“OK”（确定）。

5. 将打开“添加变量”对话框。
在前缀域中输入“Client_”，在后缀域中输入“_xyz”。选择连接“OPC_SimaticNET”。单击“完成”。
6. 在“OPC.SIMATICNet..”对话框中，单击“<- Back”（<- 返回）按钮。在“OPC 条目管理器”中，单击“退出”关闭 OPC 条目管理器。
7. 启动“图形编辑器”，打开一个新的画面。在画面中添加一个 I/O 域。从“智能对象”下的对象列表中选择“I/O 域”对象。将打开“I/O 域组态”对话框。
8. 在“变量”域中，输入变量的名称。将更新周期设置为“2 秒”。将域类型设置为“I/O 域”。
9. 单击“确定”关闭对话框并保存画面。单击图形编辑器中的“激活”按钮以启用 WinCC 项目。
10. 在 I/O 域中，将显示 FMS 索引的当前值。数值每 2 秒更新一次。在 I/O 域中输入数值。更改后的数值将传递到自动化设备。

参见

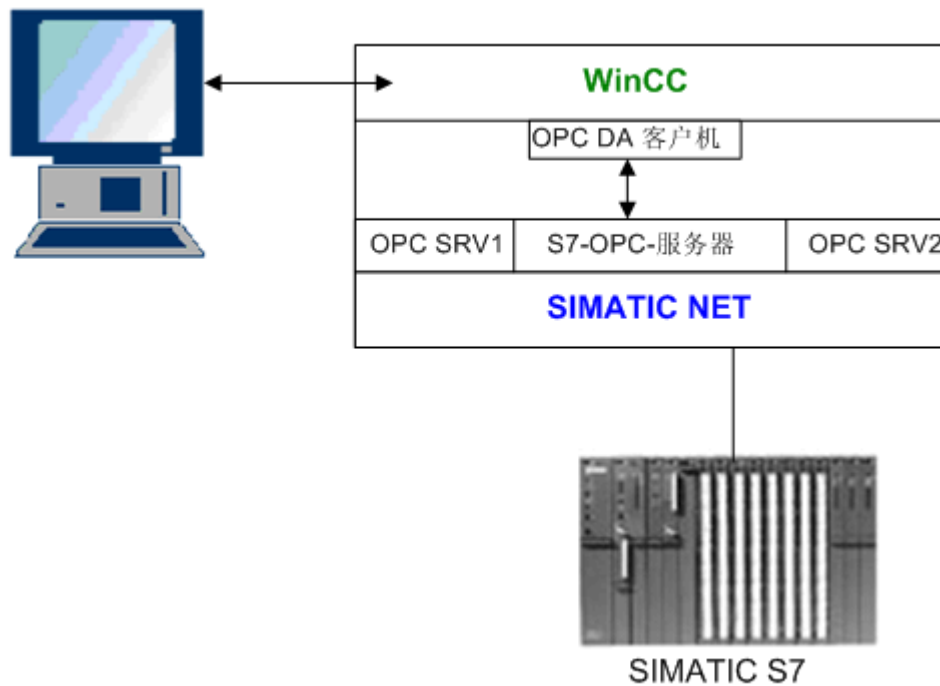
在 WinCC OPC DA 客户机上组态 OPC 通道 (页 96)

6.7.4.3 WinCC - SIMATIC NET S7-OPC 服务器的连接

WinCC - SIMATIC NET S7 OPC 服务器连接的实例

在 SIMATIC NET 的安装期间，可以选择要安装的 OPC 服务端。在下面的实例中，将组态 WinCC 到 SIMATIC NET S7 OPC 服务端的连接。通过 SIMATIC NET S7 OPC 服务端，WinCC 客户机可使用来自自动化设备的数据。

变量的当前值显示在 WinCC OPC 客户端的一个 I/O 域中。一旦 SIMATIC NET S7 OPC 服务端上的变量值改变，更改后的数值将显示在过程画面上。反之，在 I/O 域中输入的数值会发送到自动化设备上。



要求

- 安装有 WinCC、SIMATIC NET 软件的计算机。
- 已组态 SIMATIC NET S7 OPC 服务端。有关 SIMATIC NET S7 OPC 服务端设置的更多信息，请参阅 SIMATIC NET 文档。

组态步骤

为了建立 WinCC 到 SIMATIC NET S7 OPC 服务端的连接，需要下列组态：

1. 添加变量到 SIMATIC NET S7 OPC 服务端
2. 组态对 SIMATIC NET S7 OPC 服务端变量的访问

通信手册

通信手册包含通道组态的更多信息和扩展实例。本手册可在 Internet 上下载，地址如下：

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

搜索项目号：

- A5E00391327

添加变量到 SIMATIC NET S7 OPC 服务器

引言

为了用 OPC 条目管理器显示变量，必须添加变量至 SIMATIC NET S7 OPC 服务器的地址空间。使用“OPC Scout”程序进行组态。使用 SIMATIC NET 安装程序安装 OPC Scout。对于本实例，将为自动化设备上的标记字“0”编址。

所用参数表

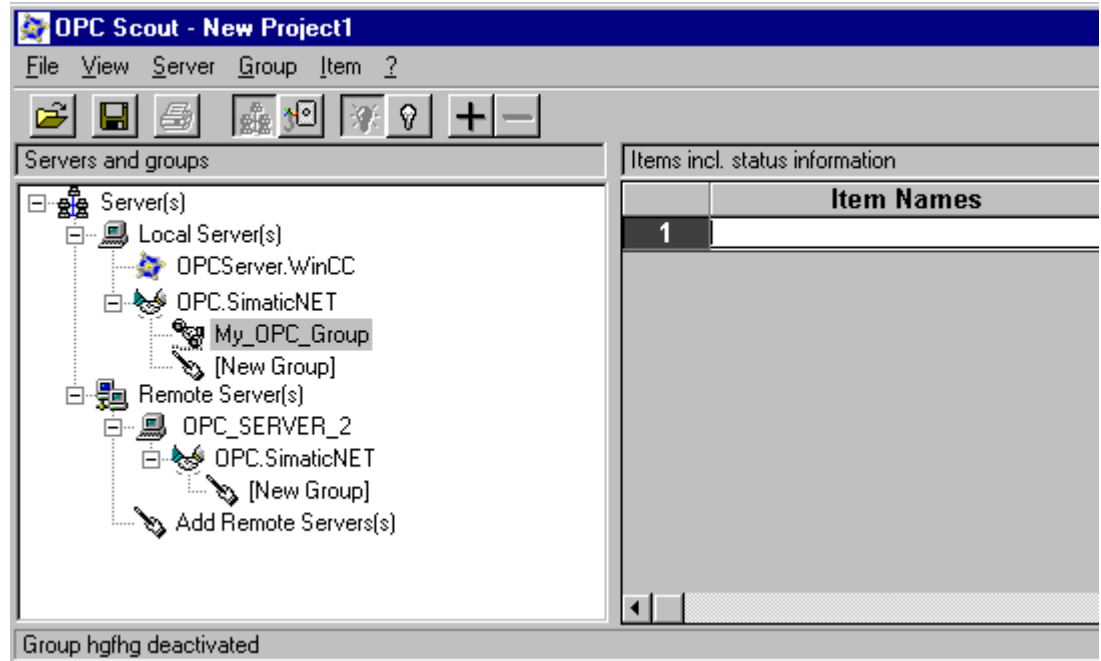
参数	数值
Data type	W
Range Byte	0
No. Values	1
Item alias	MW0

要求

- 在 SIMATIC NET 软件中组态 S7 连接。有关更多的信息，请参考 SIMATIC NET 文档。

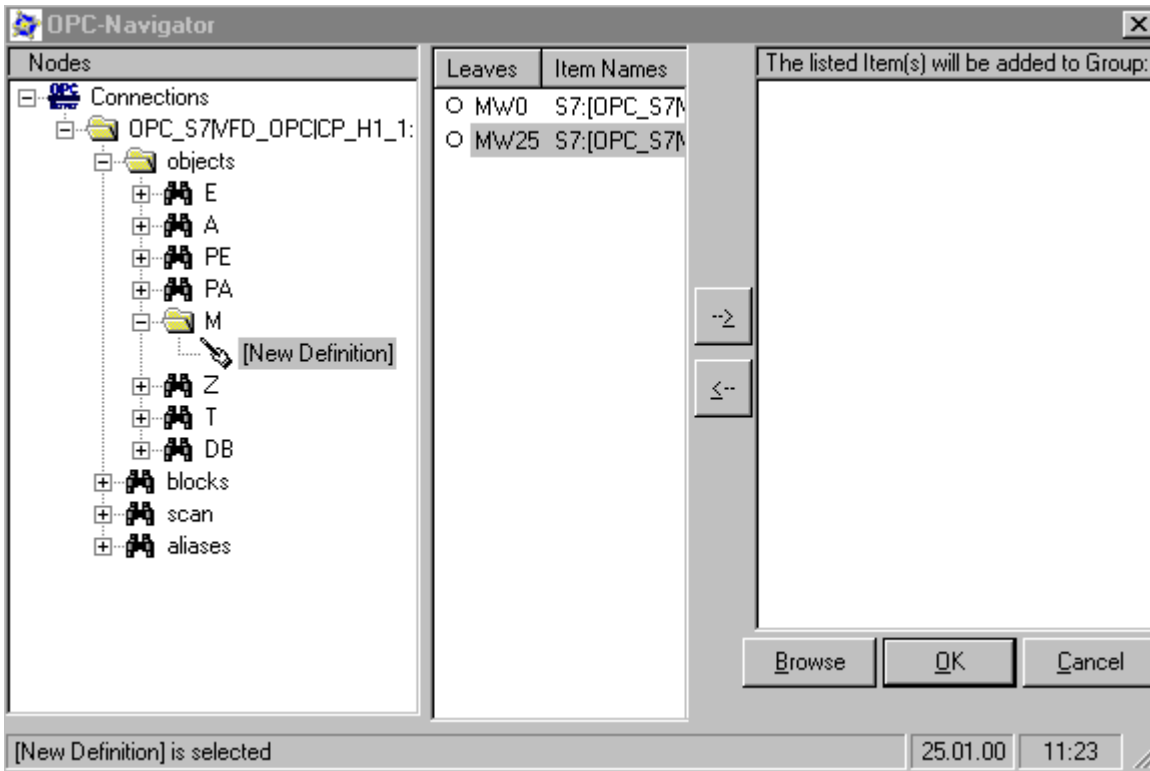
步骤

1. 通过“开始” → “程序” → “SimaticNet” → “OPCServer” → “OPCScout” 打开“OPC Scout”。

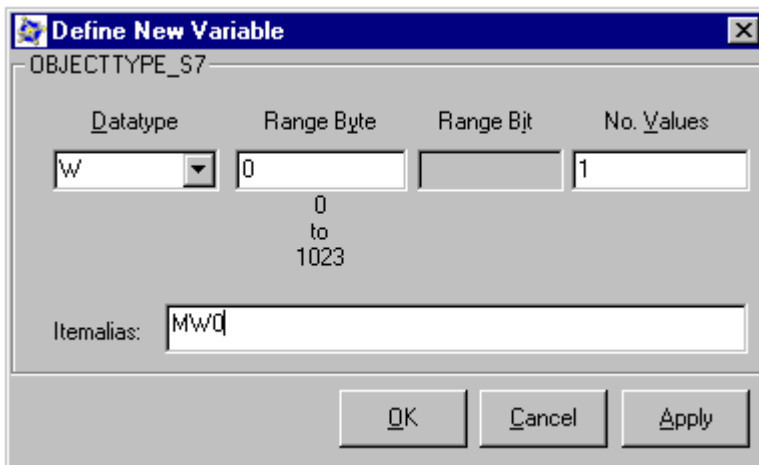


2. 选择“Local Server(s)”（本地服务器）下的“OPC.SimaticNet”。如果 SIMATIC S7 OPC 服务器不是运行在同一台计算机上，则选择“Server(s)”（服务器）快捷菜单中的“Add Remote Server(s)”（添加远程服务器）。在“Add Remote Server(s)”（添加远程服务器）对话框中输入用作 OPC 服务器的计算机名称，然后单击“OK”（确定）关闭对话框。
3. 选择“OPC.SimaticNet”快捷菜单中的“Connect”（连接）。显示“Add Group”（添加组）对话框。为该组输入一个名称。单击“OK”（确定）关闭对话框。

4. 从所添加组的快捷菜单中选择“Add Item”（添加条目）。“OPC Navigator”（OPC 浏览器）打开。



5. 在“OPC Navigator”（OPC 浏览器）中选择“Objects”（对象）下的“M”(标记)。双击“(New Definition)”（新的定义）打开“Define New Tag”（定义新的变量）对话框。
6. 在“Define New Tag”（定义新的变量）对话框中输入表格中的参数。



单击“OK”（确定）关闭“Define New Tag”（定义新的变量）对话框。

7. 在 OPC 浏览器的“Leaves”（叶）区域中标记变量“MW0”。单击“-->”按钮。单击 OPC 浏览器中的“OK”（确定）。

参见

组态对 SIMATIC NET S7 OPC 服务器变量的访问 (页 147)

组态对 SIMATIC NET S7 OPC 服务器变量的访问

引言

本节中，将在 WinCC OPC DA 客户机的 WinCC 项目中组态 WinCC 变量。该变量将访问 SIMATIC NET S7 OPC 服务器地址空间中的变量“MW0”。变量值显示在 I/O 域中。

要求

- 使用 OPC Scout 创建“MW0”变量。
- 在 WinCC OPC DA 客户机的 WinCC 项目中添加“OPC”通道。

步骤

1. 选择“OPC 组 (OPCHN Unit#1)”快捷菜单中的“系统参数”。将打开 OPC 条目管理器。
2. 从选择对话框中选择用作 OPC 服务器的计算机名称。从列表中选择“OPC.SIMATICNet”。单击“浏览服务器”按钮。将打开“过滤标准”对话框。
3. 在“过滤标准”对话框中，单击“下一步->”按钮。“OPC.SIMATICNet..”对话框打开。选择“MW0”变量。单击“添加条目”按钮。
4. 如果到 SIMATIC NET FMS OPC 服务器的连接已经存在，继续步骤 5。如果尚未组态连接，则会显示相应的消息。单击“是”。显示“New Connection”（新建连接）对话框。



输入“OPC_SlimaticNET”作为连接的名称。单击“OK”（确定）。

5. 将打开“添加变量”对话框。
在前缀域中输入“Client_”，在后缀域中输入“_xyz”。选择连接“OPC_SimaticNET”。单击“完成”。
6. 在“OPC.SIMATICNet..”对话框中，单击“<- Back”（<- 返回）按钮。在“OPC 条目管理器”中，单击“退出”关闭 OPC 条目管理器。
7. 启动图形编辑器并打开画面。在画面中添加一个 I/O 域。从“智能对象”对象列表中选择“I/O 域”对象。将打开“I/O 域组态”对话框。

6.7 WinCC OPC DA 服务器

- 8. 在“变量”域中，输入名称“Client_MW0_xyz”。将更新周期设置为“2 秒”。将域类型设置为“I/O 域”。
- 9. 关闭对话框并保存画面。单击图形编辑器中的“激活”按钮以启用 WinCC 项目。
- 10. S7 变量的当前值显示在 WinCC OPC DA 客户机的 I/O 域中。数值每 2 秒更新一次。在 I/O 域中输入数值。更改后的数值将传递到自动化设备。

参见

添加变量到 SIMATIC NET S7 OPC 服务器 (页 144)

在 WinCC OPC DA 客户机上组态 OPC 通道 (页 96)

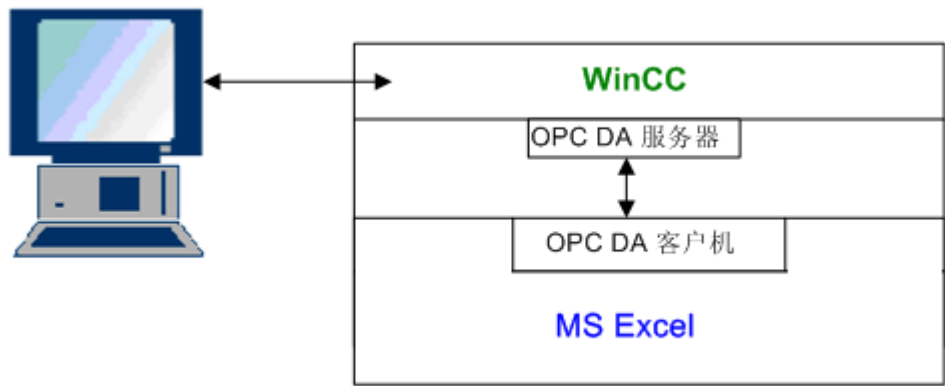
6.7.4.4 WinCC - Microsoft Excel 的连接

WinCC - Microsoft Excel 连接的实例

简介

在本实例中，将在 Microsoft Excel 中使用 Visual Basic 编辑器创建一个 OPC DA 客户机。该 OPC DA 客户机读取 WinCC OPC DA 服务器上 WinCC 项目中的 WinCC 变量，并将值写入单元格中。如果在该单元格中输入一个新值，该值将会传递到 WinCC OPC DA 服务器上。

需使用同时安装了 WinCC 和 Microsoft Excel 的计算机来进行连接。



组态步骤

必须在 Microsoft Excel 中进行下列组态：

1. 在 Microsoft Excel 的 Visual Basic 编辑器中创建一个 OPC DA 客户机
2. 在 Microsoft Excel 中组态对 WinCC 变量的访问

参见

如何组态在 Microsoft Excel 中访问 WinCC 变量 (页 152)

在 Microsoft Excel 中创建 OPC DA 客户机 (页 149)

在 Microsoft Excel 中创建 OPC DA 客户机


简介

要将 Microsoft Excel 用作 OPC DA 客户端，必须在 Microsoft Excel 的 Visual Basic 编辑器中创建特殊的脚本。

要求

具有基本的 Microsoft Excel Visual Basic 编辑器知识。

步骤

1. 打开 Microsoft Excel，将出现新的工作簿。
2. 在“Visual Basic 编辑器”的“工具”菜单中， 单击“宏”。将打开 Microsoft Excel 的 Visual Basic 编辑器。
3. 在“Visual Basic 编辑器”的“工具”菜单中选择“引用...”。将显示“引用 - VBAProject”对话框。在可用引用的列表中找到条目“Siemens OPC DAAutomation 2.0”。选中相应的复选框。单击“确定”(OK)。
4. 复制以下所显示的脚本。该脚本仅在在线帮助中可用。
5. 双击 Visual Basic 编辑器项目窗口中的“Sheet1”，打开新的代码窗口。
6. 将脚本粘贴到代码窗口中。
7. 从“文件”菜单中选择“保存”。从“文件”菜单中选择“关闭并返回到 Microsoft Excel”。

示例脚本

```

Option Explicit
Option Base 1

Const ServerName = "OPCServer.WinCC"

Dim WithEvents MyOPCServer As OpcServer
Dim WithEvents MyOPCGroup As OPCGroup
Dim MyOPCGroupColl As OPCGroups
Dim MyOPCItemColl As OPCItems
Dim MyOPCItems As OPCItems
Dim MyOPCItem As OPCItem

Dim ClientHandles(1) As Long
Dim ServerHandles() As Long
Dim Values(1) As Variant
Dim Errors() As Long
Dim ItemIDs(1) As String
Dim GroupName As String
Dim NodeName As String

'-----
' Sub StartClient()
' Purpose: Connect to OPC_server, create group and add item
'-----
Sub StartClient()
    ' On Error GoTo ErrorHandler
    '----- We freely can choose a ClientHandle and GroupName
    ClientHandles(1) = 1
    GroupName = "MyGroup"
    '----- Get the ItemID from cell "A1"
    NodeName = Range("A1").Value
    ItemIDs(1) = Range("A2").Value
    '----- Get an instance of the OPC-Server
    Set MyOPCServer = New OpcServer
    MyOPCServer.Connect ServerName, NodeName

    Set MyOPCGroupColl = MyOPCServer.OPCGroups
    '----- Set the default active state for adding groups
    MyOPCGroupColl.DefaultGroupIsActive = True
    '----- Add our group to the Collection
    Set MyOPCGroup = MyOPCGroupColl.Add(GroupName)

    Set MyOPCItemColl = MyOPCGroup.OPCItems
    '----- Add one item, ServerHandles are returned
    MyOPCItemColl.AddItem 1, ItemIDs, ClientHandles, ServerHandles, Errors
    '----- A group that is subscribed receives asynchronous notifications
    MyOPCGroup.IsSubscribed = True
Exit Sub

```

```
ErrorHandler:
  MsgBox "Error: " & Err.Description, vbCritical, "ERROR"
End Sub

'-----
' Sub StopClient()
' Purpose: Release the objects and disconnect from the server
'-----
Sub StopClient()
  '----- Release the Group and Server objects
  MyOPCGroupColl.RemoveAll
  '----- Disconnect from the server and clean up
  MyOPCServer.Disconnect
  Set MyOPCItemColl = Nothing
  Set MyOPCGroup = Nothing
  Set MyOPCGroupColl = Nothing
  Set MyOPCServer = Nothing
End Sub

'-----
' Sub MyOPCGroup_DataChange()
' Purpose: This event is fired when a value, quality or timestamp in our Group has changed
'-----
'----- 如果安装了 OPC-DA Automation 2.1, 使用:
Private Sub MyOPCGroup_DataChange(ByVal TransactionID As Long, ByVal NumItems As Long,
ClientHandles() As Long, ItemValues() As Variant, Qualities() As Long, TimeStamps() As
Date)
  '----- Set the spreadsheet cell values to the values read
  Range("B2").Value = CStr(ItemValues(1))
  Range("C2").Value = Hex(Qualities(1))
  Range("D2").Value = CStr(TimeStamps(1))
End Sub

'-----
' Sub worksheet_change()
' Purpose: This event is fired when our worksheet changes, so we can write a new value
'-----
Private Sub worksheet_change(ByVal Selection As Range)
  '----- Only if cell "B3" changes, write this value
  If Selection <> Range("B3") Then Exit Sub
  Values(1) = Selection.Cells.Value
  '----- Write the new value in synchronous mode
  MyOPCGroup.SyncWrite 1, ServerHandles, Values, Errors
End Sub
```

参见

如何组态 WinCC OPC DA 服务器上的 WinCC 项目 (页 137)

如何组态在 Microsoft Excel 中访问 WinCC 变量

引言

Excel OPC DA 客户机读取 WinCC OPC DA 服务器的 WinCC 变量并将变量值写入单元格中。在 WinCC OPC DA 服务器的 WinCC 项目中，变量值显示在一个 I/O 域内。如果单元格中的变量值改变，则在 WinCC OPC DA 服务器 I/O 域中的数值也随之改变。

要求

- 在 WinCC OPC DA 服务器的 WinCC 项目中，组态名称为“OPC_Excel”且数据类型为“有符号 16 位数”的内部变量。
- 将“OPC_Excel”变量的数值写入 WinCC OPC DA 服务器的 WinCC 项目的 I/O 域内。
- 启用 WinCC OPC DA 服务器的 WinCC 项目。

步骤

1. 在 Microsoft Excel 的单元格 A1 中，输入用作 OPC 服务器的计算机名称。在单元格 A2 中，输入变量名称“OPC_Excel”。

	A	B
1	dpc_4001	
2	OPC_Excel	
3		

2. 在 Excel 的“工具”菜单中，选择“宏” → “宏”。“宏”对话框将打开。从宏列表中选择条目“Sheet1.StartClient”。单击“运行”以启动 OPC 客户机。
3. 变量的值会写入单元格 B2 中，质量代码写入 C2 中，时间标志写入 D2 中。
4. 在单元格 B3 中输入新的数值。更改后的数值显示在 WinCC OPC 服务器上的 I/O 域内。
5. 在 Excel 的“工具”菜单中，选择“宏” → “宏...”。“宏”对话框将打开。从宏列表中选择条目“Sheet1.StopClient”。单击“运行”停止 OPC 客户机。

6.8 WinCC OPC HDA 服务器

6.8.1 WinCC OPC HDA 服务器的功能

简介

WinCC OPC HDA 服务器是 DCOM 应用程序，为 OPC HDA 客户端提供来自归档系统的必需数据。使用条目句柄访问数据。可以进行读访问或写访问。还可以分析数据。

WinCC OPC HDA 服务器支持 OPC 历史数据访问 1.20 规范。这已经由适应性测试确认。下列章节说明数据结构的设计，以及 WinCC OPC HDA 服务器支持的属性、集合和功能。这不是详细的说明，而是最重要信息的摘要。有关更多信息，可参见“OPC 历史数据访问 1.20”规范。

安装

在安装 WinCC 期间，可以选择 WinCC OPC HDA 服务器。可以选择对 WinCC 归档系统的访问是否具有写功能。安装后，WinCC OPC DA 服务器无需进行组态，立即可用。

如果安装时未选择写访问，则只能读取和分析 WinCC 归档系统中的数据。如果选择了写访问，可以分析、添加、删除和更新 WinCC 归档系统中的数据。

可以在 WinCC 服务器或 WinCC 客户端上安装 WinCC OPC HDA 服务器。

许可

为了操作 WinCC OPC HDA 服务器，必须在每台作为 OPC HDA 服务器的 WinCC 计算机上安装下列许可证：

- 有效的 WinCC RT 许可证
- WinCC 选件连通性软件包

OPC HDA 客户端

所有符合 OPC 历史数据访问 1.20 规范的 OPC HDA 客户端都可以访问 WinCC OPC HDA 服务器。也可自己创建 OPC HDA 客户端。通过创建专有 OPC HDA 客户端，可以满足大多数用户特定需求。

如何使用 OPC HDA 客户端的实例包括：

- 分析和判断归档数据
- 从不同的 OPC HDA 服务器对归档进行统计过程控制

要使用 OPC HDA 客户端请求历史数值，需要在组态过程中注意以下几点：

- 选择的查询周期应能使客户端在发送下一次查询之前收到所请求的数据。周期太短可能造成接收数据时时间延迟过长。
- WinCC 服务器的 CPU 负载取决于每次查询的变量数。

使用组态的交换写访问循环归档

在运行系统中，数据在 WinCC 服务器的循环归档中进行修改。

只有在创建数据后立即进行更改时才会将更改传送给换出归档。

如果循环归档的相关归档段已经换出，则随后不会在换出归档中完成更改。在 WinCC 服务器上删除了归档段，修改的数据也将删除。

参见

质量代码 (页 160)

WinCC OPC HDA 服务器的数据结构 (页 154)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.8.2 WinCC OPC HDA 服务器的数据结构

6.8.2.1 WinCC OPC HDA 服务器的数据结构

引言

WinCC OPC HDA 服务器上的数据是结构化的。以下列出了可用的数据结构。这不是详细的说明，而是最重要信息的摘要。有关更多信息，可参见“OPC 历史数据访问 1.20”规范。

数据结构

	描述
属性	为原始数据提供附加的质量特性。属性包括关于归档的数据类型、规范等。有关更多信息，请参见所支持属性的概述。
配件	指定时间间隔中原始数据的摘要。集合包括平均值、最小值和最大值等。有关更多信息，请参见所支持集合的概述。
开始时间/结束时间	设置时间间隔的开始点和结束点。
范围值	在开始和结束时记录的值。如果没有可用的范围值，靠近这些时间的数值将用作范围值。
原始数据	WinCC 归档系统中特定时间间隔的数据。这些数据包含时间标志和质量等级。
条目句柄	唯一指定给 WinCC 归档变量。
条目标识号	WinCC 归档变量的唯一标识符。条目标识号可用于获取条目句柄。

参见

受支持函数的概述 (页 158)

WinCC OPC HDA 服务器的时间格式 (页 158)

受支持属性的概述 (页 155)

受支持配件的概述 (页 156)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.8.2.2 受支持属性的概述

引言

下表包括了 WinCC OPC HDA 服务器支持的属性。有关更多信息，可参见“OPC 历史数据访问 1.20”规范。

属性

属性	属性标识号	描述
条目标识号	OPCHDA_ITEMID	指示要访问的 WinCC 归档变量。
条目数据类型	OPCHDA_DATA_TYPE	指示 WinCC 归档变量的数据类型。
描述	OPCHDA_DESCRIPTION	返回 WinCC 归档变量的说明。该说明在 WinCC 变量记录中定义。
工程单位	OPCHDA_ENG_UNITS	设置度量单位的显示。设定标签在 WinCC 变量记录中定义。

参见

WinCC OPC HDA 服务器的数据结构 (页 154)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.8.2.3 受支持配件的概述

引言

下表列出了 WinCC OPC HDA 服务器支持的集合。有关更多信息，可参见“OPC 历史数据访问 1.20”规范。

配件

配件	描述
OPCHDA_COUNT	返回指定时间间隔中原始数据的计数。
OPCHDA_START	返回时间间隔开始时原始数据的初始值。
OPCHDA_END	返回时间间隔结束时原始数据的最终值。
OPCHDA_AVERAGE	返回指定时间间隔中原始数据的平均值。
OPCHDA_TIMEAVERAGE	返回指定时间间隔中原始数据按时间的平均值。
OPCHDA_TOTAL	返回指定时间间隔中的总和值。
OPCHDA_STDEV	返回指定时间间隔中原始数据的标准偏差。

配件	描述
OPCHDA_MINIMUMACTUALTIME	返回指定时间间隔中原始数据的最小值及其时间标志。
OPCHDA_MINIMUM	返回指定间隔中原始数据的最小值。
OPCHDA_MAXIMUMACTUALTIME	返回指定时间间隔中原始数据的最大值及其时间标志。
OPCHDA_MAXIMUM	返回指定间隔中原始数据的最大值。
OPCHDA_DELTA	返回指定时间间隔中原始数据中第一个值和最后一个值的差。
OPCHDA_REGSLOPE	返回指定时间间隔中原始数据回归线的斜率。
OPCHDA_REGCONST	返回起始点处原始数据的回归值。
OPCHDA_REGDEV	返回指定时间间隔内原始数据回归的标准偏离。
OPCHDA_VARIANCE	返回指定时间间隔内原始数据的变化。
OPCHDA_RANGE	返回指定时间间隔内原始数据的 OPCHDA_MAXIMUM 和 OPCHDA_MINIMUM 的差。
OPCHDA_DURATIONGOOD	返回原始数据质量好的时间段。时间段以秒为单位。
OPCHDA_DURATIONBAD	返回原始数据质量差的时间段。时间段以秒为单位。
OPCHDA_PERCENTGOOD	返回质量好的原始数据的百分比。
OPCHDA_PERCENTBAD	返回质量差的原始数据的百分比。
OPCHDA_WORSTQUALITY	返回指定时间间隔内原始数据的最差质量。

参见

WinCC OPC HDA 服务器的数据结构 (页 154)

WinCC OPC HDA 服务器的功能 (页 153)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.8.2.4 受支持函数的概述

引言

下表列出了 WinCC OPC HDA 服务器支持的函数。 OPC HDA 客户机可使用这些函数进行数据交换。 有关更多信息，可参见“OPC 历史数据访问 1.20”规范。

读

函数	描述
ReadRaw	返回指定时间间隔内的原始数据及其质量和时间标志。
ReadProcessed	返回计算出的数值、数值质量和指定时间间隔的时间标志。 计算出的数值取决于所选的集合。
ReadAtTime	返回特定时间间隔内的原始数据及其质量和时间标志。 如果没有数值可用，将插入该点的数值。
ReadAttribute	返回指定时间间隔的条目属性和时间标志。

参见

WinCC OPC HDA 服务器的功能 (页 153)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.8.2.5 WinCC OPC HDA 服务器的时间格式

引言

在 WinCC OPC HDA 服务器上通过设置开始时间和结束时间指定时间间隔。 指定的时间间隔确定历史数据的观察期。 在指定时间时，必须保持特定的格式。

下列选项可用于指定时间格式：

- 基于 UTC 的绝对时间
- 相对于服务器当地时间的相对时间

根据 UTC 的绝对值

缺省状态下，WinCC OPC HDA 服务器使用相同的世界时间（UTC）作为其时间基准。该时间对应于格林威治标准时间（即中欧时间减去一小时）。

时间格式

YYYY/MM/DD hh:mm:ss.msmsms

参数

YYYY = 年份

MM = 月份

DD = 日

hh = 小时

mm = 分钟

ss = 秒

ms = 毫秒

输入实例

2002/06/10 09:27:30.000

指定相对于当地时间的的时间

对于该选项，输入的时间相对于服务器的当地时间。当地时区在计算机的“日期/时间”控制面板中设置。

时间格式

关键字 +/- 误差 1 +/- 误差(n)

误差是与服务器当地时间的差。

关键字

NOW = 服务器上当前的当地时间

SECOND = 当前秒钟

MINUTE = 当前分钟

HOUR = 当前小时

DAY = 当前日

WEEK = 当前星期

MONTH = 当前月份

YEAR = 当前年份

误差

+/-S = 秒钟差

+/-M = 分钟差

+/-H = 小时差

+/-D = 日差

+/-W = 星期差

+/-MO = 月份差

+/-Y = 年份差

实例:

DAY - 1D = 前一日

DAY-1D + 7H30 = 前一日的 7:30

MO-1D+5H = 上个月最后一天的 5:00

NOW-1H15M = 1 小时 15 分钟前

YEAR+3MO= 本年度的四月

参见

WinCC OPC HDA 服务器的功能 (页 153)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.8.3 质量代码

简介

质量代码用来判断原始数据的状态和质量。 OPC 的质量代码在“数据访问自定义界面标准 3.00 版”规范的“6.8 OPC 质量标记”中有说明。

WinCC OPC HDA 服务器的质量代码

代码	OPC	描述	质量
0x00040000	OPCHDA_RAW	指示原始数据传输的质量。	好 劣 不确定
0x00080000	OPCHDA_CALCULATED	指示计算出的数据传输的质量。	好 劣 不确定
0x00100000	OPCHDA_NOBOUND	在开始点或结束点处未发现范围值。	劣
0x00200000	OPCHDA_NODATA	指定时间间隔内没有发现任何原始数据。	劣
0x00400000	OPCHDA_DATALOST	所选间隔内的原始数据未完全归档。	劣

参见

www.opcfoundation.org (<http://www.opcfoundation.org>)



6.8.4 支持的写访问

引言

下表列出了 WinCC OPC HDA 服务器支持的写访问。

表格元素:

	描述
周期性归档	要归档的过程值存储在循环归档中。循环归档由数目可组态的数据缓冲区组成。定义数据缓冲区的大小和时间段（例如以天为单位）。当所有数据缓冲区都满时，第一个数据缓冲区中的过程数据被覆盖。
交换之后循环归档	为了使数据缓冲区中的过程数据不被覆盖，可以将其先交换出来（导出）。


	描述
	受 WinCC 支持。
	不受 WinCC 支持。

写访问

以后添加过程值

周期性归档	交换之后循环归档	受 WinCC 支持	描述
是	否		如果时间段包含在循环归档中，以后可以添加过程值。
是	是		相应时间段的数据缓冲区被交换至归档备份。以后，不能将过程值添加到归档备份。
否	否		循环归档不可用。不能存储过程值。
不可以	可以		循环归档不可用。不能存储过程值。

在运行系统中添加过程值

周期性归档	交换之后循环归档	受 WinCC 支持	描述
是	否		过程值被添加到循环归档当前有效的数据缓冲区中。

插入将来的过程值

周期性归档	交换之后循环归档	受 WinCC 支持	描述
是	否		在写访问期间，在未来不能添加值。
否	否		写访问时，在未来不能添加值。

删除过程值

周期性归档	交换之后循环归档	受 WinCC 支持	描述
是	否	☺	时间段包含在循环归档中时，可以删除过程值。
是	是	☹	相应时间段的数据缓冲区被交换至归档备份。可以从归档备份删除过程值。
否	否	☹	循环归档不可用。不能存储过程值。
不可以	可以	☹	循环归档不可用。不能存储过程值。

编辑过程值

周期性归档	交换之后循环归档	受 WinCC 支持	描述
是	否	☺	时间段包含在循环归档中时，可以编辑过程值。
是	是	☹	相应时间段的数据缓冲区被交换至归档备份。可以在归档备份中编辑过程值。
否	否	☹	循环归档不可用。不能存储过程值。
不可以	可以	☹	循环归档不可用。不能存储过程值。

6.8.5 OPC HDA 连接实例

6.8.5.1 OPC HDA 连接实例

引言

在下面的实例中，将组态 WinCC 和 OPC HDA 客户机之间的连接。来自 WinCC 归档系统的数据通过 WinCC OPC HDA 服务器变得可用。OPC HDA 客户机通过条目句柄访问数据。要简化组态过程，将使用 OPC HDA 浏览器。

将使用来自 OPC 基金会的 OPC HDA 客户机。所有符合 OPC 历史数据访问 1.20 规范的 OPC HDA 客户机都可以访问 WinCC OPC HDA 服务器。

要求

- 在 WinCC OPC HDA 服务器的 WinCC 项目中，创建名称为“OPC_HDA”、数据类型为“无符号 16 位数”的内部变量。
- 创建在 WinCC 归档系统中称为“HDA_ProcessValueArchive”的过程值归档。
- 在过程值归档“HDA_ProcessValueArchive”中创建称为“OPC_HDA_Tag”的 WinCC 归档变量。将 WinCC 归档变量与内部变量“OPC_HDA”链接。
- 在运行系统列表中，启动变量记录运行系统，禁用图形运行系统。
- 启动 WinCC OPC HDA 服务器的 WinCC 项目。

组态步骤

将 WinCC 连接到 OPC HDA 客户机需要以下组态：

1. 使用 HDA 服务器浏览器组态对 WinCC 归档变量的访问
2. 读取 WinCC 归档变量的数值

参见

如何使用 HDA 服务器浏览器来组态对 WinCC 归档变量的访问 (页 166)

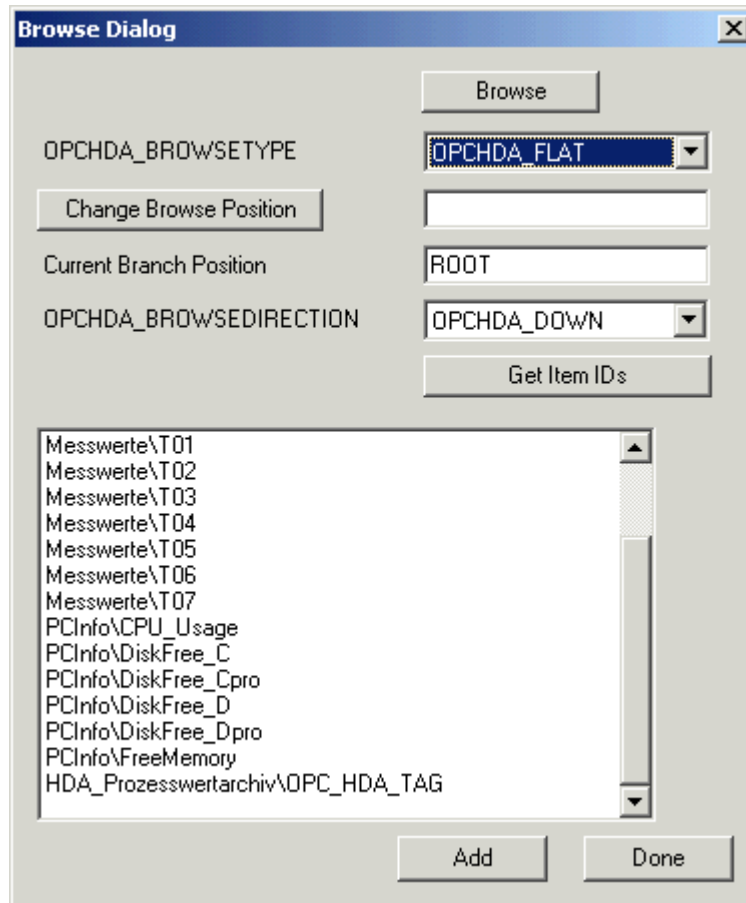
HDA 服务器浏览器 (页 165)

读取 WinCC 归档变量的数值 (页 167)

6.8.5.2 HDA 服务器浏览器

引言

OPC HDA 客户机通过条目句柄访问变量值。为了易于组态，WinCC OPC HDA 服务器支持浏览器功能。OPC HDA 客户机可以使用 HDA 服务器浏览器来搜索 WinCC OPC HDA 服务器的地址空间。数据以过程值归档分级列出。



说明

如果不通过 HDA 服务器浏览器来访问 WinCC 归档变量，就需要手动组态条目标识号。寻址 WinCC 归档变量时，计算机名称（服务器前缀）包含在路径中。条目标识号具有下列语法：Server-prefix::process_value_archive\WinCC_archive_tag。

参见

如何使用 HDA 服务器浏览器来组态对 WinCC 归档变量的访问 (页 166)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.8.5.3 如何使用 HDA 服务器浏览器来组态对 WinCC 归档变量的访问

简介

在本部分中，OPC HDA 客户机将用于访问 WinCC 归档变量。将使用来自 OPC 基金会的 OPC HDA 客户机。HDA 服务器浏览器将用于组态访问。

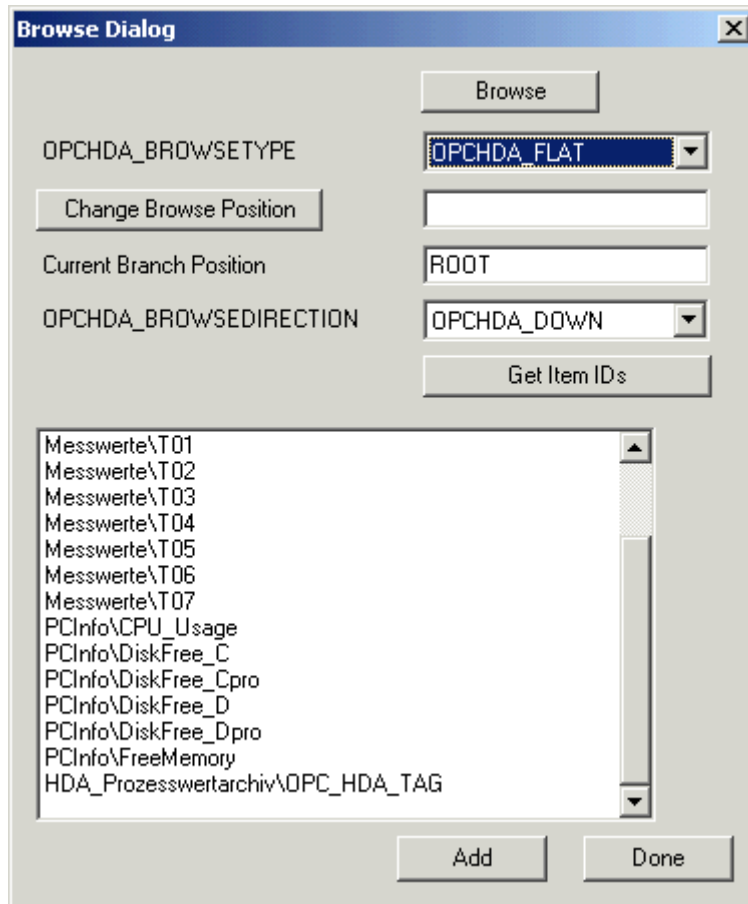
说明

此处的 OPC HDA 客户机是来自 OPC 基金会的演示客户机。源代码可以参见 Internet 网址 <http://www.opcfoundation.org>。

步骤

1. 将“SampleClientHDA.exe”文件从文件夹“Siemens\WinCC\documents\english”复制到您选择的文件夹中。
2. 双击“SampleClientHDA.exe”文件。HDA 客户机程序将启动。
3. 在“服务器名称”区域中，选择条目“OPCServerHDA.WinCC.1”。单击“连接”。确认随后的对话框。

- 在 HDA 客户机中单击“浏览”。将打开“浏览对话框”对话框。在“OPCHDA_BROWSETYPE”域中选择“OPCHDA_FLAT”。



- 在选择窗口中，选择“HDA_ProcessValueArchive_HDA_TAG”条目。单击“添加”，然后单击“完成”关闭对话框。

有关详细信息，请访问 <http://www.opcfoundation.org>。

参见

读取 WinCC 归档变量的数值 (页 167)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.8.5.4 读取 WinCC 归档变量的数值

引言

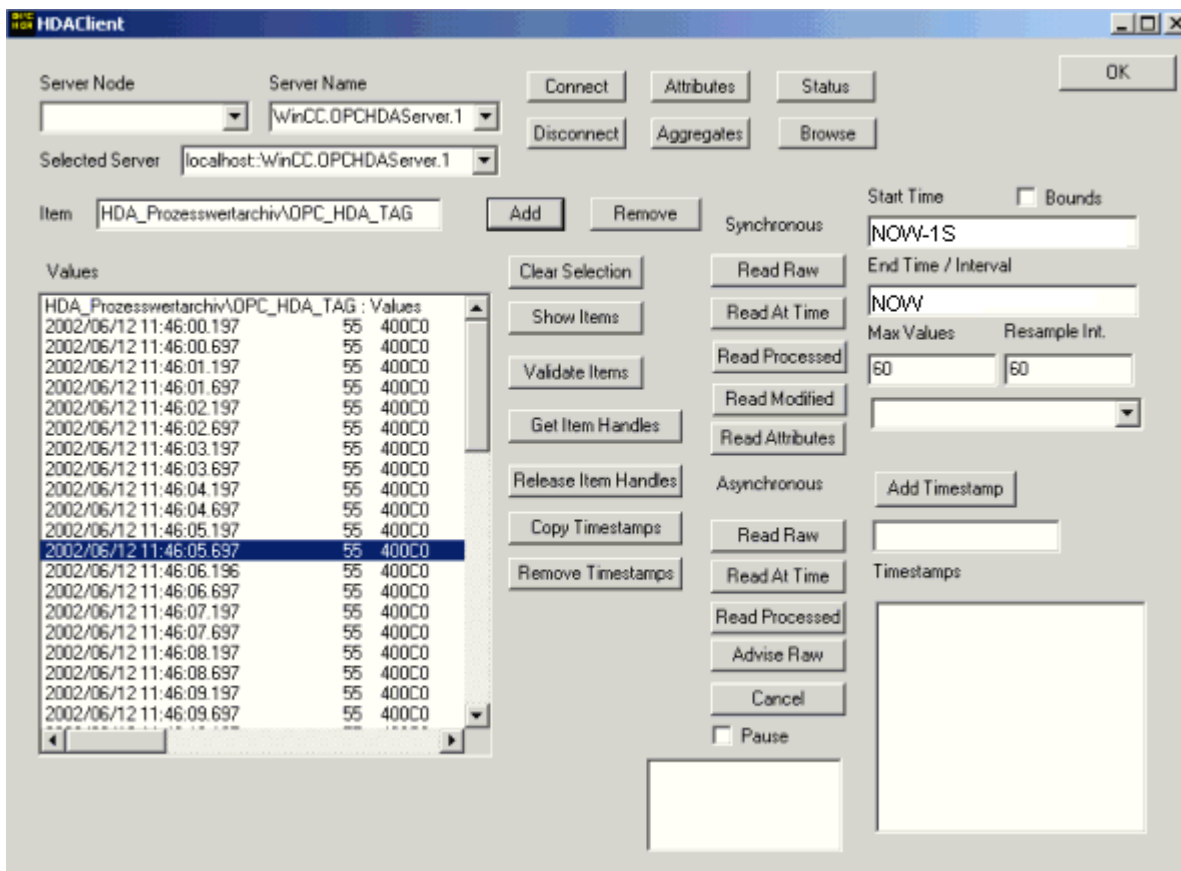
本节说明如何访问和读取 WinCC 归档变量。

要求

- OPC HDA 客户机必须在运行。

步骤

1. 在 HDA 客户机中单击“Show Items”（显示条目）。
2. 在 HDA 客户机中单击“Get Item Handles”（获取条目句柄）。
3. 双击“Value”（数值）选择域中的“HDA_ProcessValueArchive_HDA_Tag”。
4. 在“Start Time”（开始时间）域中输入“NOW-10S”。在“End Time”（结束时间）域中输入“NOW”。



5. 单击“Read Raw”（读取原始数据）。数值及其质量代码和时间标志显示在“Values”（数值）选择域中。

6.8.6 WinCC 中 OPC HDA 服务器针对非周期性记录的特殊功能

简介

在 WinCC 中，会周期性或非周期性地执行变量记录。WinCC OPC HDA 服务器会视变量记录方法而定以不同的方式工作：

- 对于所有周期性记录的值，OPC HDA 服务器都会遵照 OPC Foundation 的 HDA 规范运行。OPC 总计将进行线性内插。
- OPC Foundation 的 HDA 规范中不包括非周期性记录的变量。OPC 总计将进行增量式内插。尤其是如果变量长时间未经历变化，某个时间段内将没有可用的数据。应对以下事项加以考虑，从而在出现上述情况时仍可获得有效的数据。

说明

对于非周期性记录的变量，OPC HDA 服务器不遵从 OPC 的规定。OPC Foundation 的 HDA 规范不认可非周期性记录的变量，因此任何归档服务器都无法处理非周期性记录的变量。所支持的总计将依照 OPC HDA 规范进行计算。不支持任何非显式调用的函数。

说明

如果启用对过程值归档的写入访问，则不能再添加任何值。

组态非周期性记录的变量

为组态非周期性记录的变量，需要为变量启用“进行段更改后归档”(Archive after segment change) 设置。段发生变化时，此操作会在新记录中输入最新的有效值。

对于非周期性记录的变量 WinCC OPC HDA 服务器所支持的总计

OPC HDA 服务器支持以下总计：

- OPCHDA_MINIMUM
- OPCHDA_MAXIMUM
- OPCHDA_AVERAGE
- OPCHDA_END
- OPCHDA_INTERPOLATIVE
- OPCHDA_TIMEAVERAGE
- OPCHDA_TOTAL

- OPCHDA_DURATIONGOOD
- OPCHDA_PERCENTGOOD

对于非周期性记录的变量 WinCC OPC HDA 服务器所支持的函数

- 仅限有“约束”的 ReadRaw。针对变量的 ReadRaw 在执行时必须始终施加“约束”，以找到无记录值变化区域的最后一个实际存储值。
- ReadProcessed
- DeleteRaw
- DeleteAtTime
- Insert
- InsertReplace
- Replace

计算非周期性记录变量的总计

总计的计算基于扩展的“RawData”数据记录，该记录除包含实际存储值外，还包含用于计算的虚拟数据点。WinCC OPC HDA 服务器会根据“ReadProcessed”的要求准备包含的“RawData”。计算所需的虚拟数据点由边界实际数据点组成。虚拟数据点包括以下重要点：

- “StartTime” 的值
- “EndTime” 的值
- 间隔限制的值

实例

“00:59:00”、“01:02:00”和“01:03:00”的值是为非周期变量记录变量存储的。OPC HDA 客户机通过“ReadProcessed”要求具有以下参数的总计：

- StartTime = 01:00:00
- EndTime = 01:04:00
- Interval = 00:02:00

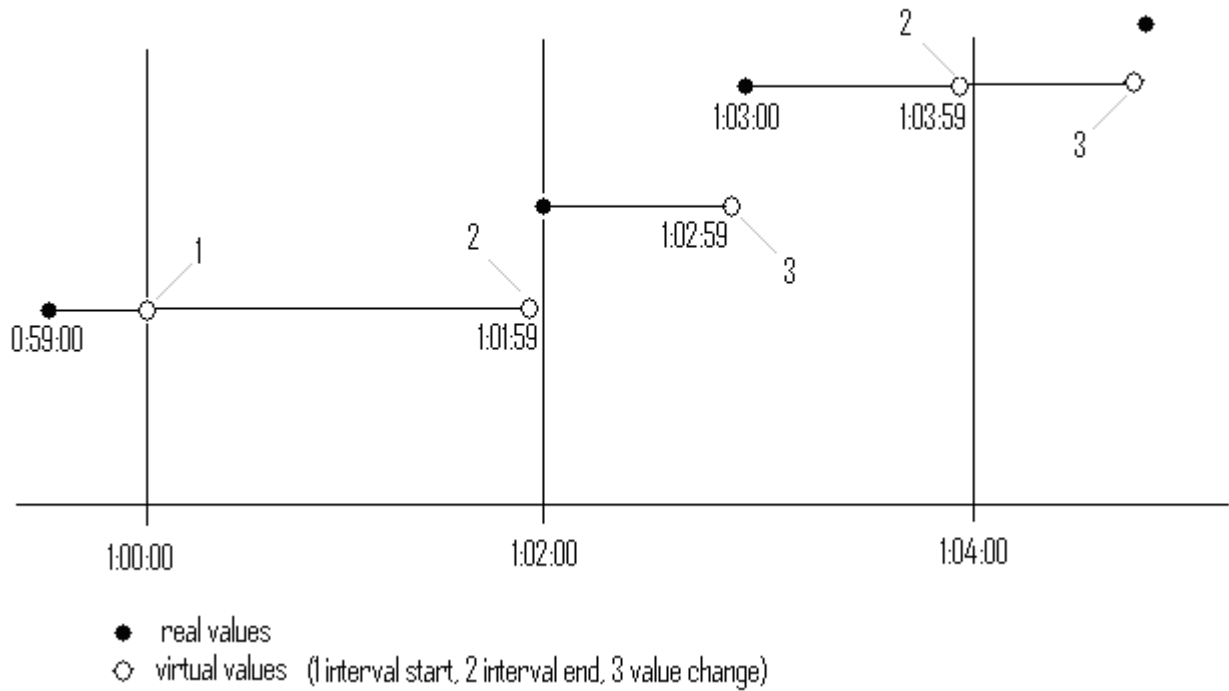
说明

在限制（“EndTime”/“Interval”）处生成虚拟值时，时间段长度始终比限制处计算的时间戳短 1 μ s。

下表中使用了 1 秒的增量，以提高总览效果。下图图解了该实例。

OPC 服务器使用下面的“RawData” 来计算总计：

编号	时间戳	实际存储值	生成的虚拟值
1	00:59:00	1.00	
2	01:00:00		1.00
3	01:01:59		1.00
4	01:02:00	2.00	
5	01:02:59		2.00
6	01:03:00	3.00	
7	01:03:59		3.00



6.9 WinCC OPC A&E 服务器

6.9.1 WinCC OPC A&E 服务器的功能

简介

WinCC OPC A&E 服务器是 DCOM 应用程序。OPC A&E 客户端通过订阅随时了解 WINCC 消息的状态变化情况。OPC A&E 客户端可以对订阅使用过滤器。该过滤器确定显示哪些消息和属性。

WinCC OPC A&E 服务器支持规范 OPC Alarm&Event 1.10。这已经由适应性测试确认。

下列章节说明 WinCC 消息系统在 OPC A&E 上的显示，以及 WinCC OPC A&E 服务器支持的属性。这不是详细的说明，而是最重要信息的摘要。有关更多信息，可参见“OPC 报警 & 事件 1.10”规范。

安装

可以在安装 WinCC 期间选择 WinCC OPC A&E 服务器。安装后，WinCC OPC A&E 服务器无需进行组态，立即可用。

可以在 WinCC 服务器和 WinCC 客户端上安装 WinCC OPC A&E 服务器。

许可

为了运行 WinCC OPC A&E 服务器，必须在每台作为 OPC A&E 服务器的 WinCC 服务器上安装下列许可证：

- 有效的 WinCC RT 许可证
- WinCC 选件连通性软件包

服务器类型

WinCC OPC A&E 服务器支持条件事件和简单事件。此外，还支持跟踪事件。

与条件相关的事件服务器

对于条件相关的事件服务器，事件与某个条件相关。例如，条件可以是变量超出限值。一旦超出边界值，便会在 WinCC 中生成消息。此消息在 OPC A&E 中显示为报警。

简单事件服务器

简单事件是提示 OPC A&E 客户端关于事件的消息。简单事件包括启动或退出程序等。

说明

在使用冗余系统时，请注意如下事项：

更新变量时，与内部变量互连的简单事件会发送两次。

第一个消息由主站触发，第二个消息由备用站触发。

跟踪事件服务器

如果在过程中发生了变化，OPC A&E 客户端会接收到消息。例如，这一变化可能是调节器调整。

OPC A&E 客户端

所有符合 OPC Alarms & Events 1.10 规范的 OPC A&E 客户端都可以访问 WinCC OPC A&E 服务器。也可自己创建 OPC A&E 客户端。通过创建专有 OPC 客户端，可以满足大多数用户特定需求。例如，OPC A&E 客户端可用于对来自多重 OPC A&E 服务器的报警进行分析和一般归档。

参见

OPC A&E 质量代码 (页 181)

在 OPC A&E 上映射 WinCC 消息系统 (页 173)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.9.2 在 OPC A&E 上映射 WinCC 消息系统

6.9.2.1 在 OPC A&E 上映射 WinCC 消息系统

导言

组态 WinCC 消息系统期间，将进行设置以确定哪些过程事件产生消息。此消息在 OPC A&E 中显示为报警。下面的表格列出了报警最重要的参数。还描述了如何通过 WinCC 消息系统使信息可用。更多信息，请参阅“报警结构”。

概述

OPC	WinCC 消息系统
Source	指示消息源。消息源的格式为“<服务器前缀>::@LOCALMACHINE::”。
Time	发布已接收、已发送和已确认消息的时间标志。发布 UTC（协调世界时）格式的时间标志。
Type	指示是简单事件、跟踪事件还是条件相关事件。WinCC - POC A&E 服务器支持简单事件、条件相关事件和跟踪事件。
Severity	指示 WinCC 消息的优先级。
EventCategory	返回消息类别。有关该主题的更多信息，请参阅“显示消息等级和类型”。
Message	指示相应消息号的消息文本。
ConditionName	指示消息号。
ChangeMask	指示消息改变后的状态。更多信息，请参阅“确认方法”。
NewState	返回消息状态。更多信息，请参阅“确认方法”。
ConditionQuality	返回消息质量。更多信息，请参阅“质量代码”。
AckRequired	指示消息是否需要确认(接收)。
ActiveTime	返回已接收消息的时间标志。
EventAttribute	列出相关消息所需的属性。更多信息，请参阅“WinCC 消息系统的属性”。
Quality	返回消息的质量代码。
Cookie	从 OPC A&E 服务器返回 cookie。cookie 对应于 WinCC 报警系统中的消息号

参见

确认方法 (页 178)

WinCC 消息系统的属性 (页 176)

映射 WinCC 消息类别和消息类型 (页 175)

6.9.2.2 映射 WinCC 消息类别和消息类型

简介

WinCC 消息系统向用户提示过程中的使用干扰和操作情况。WinCC 消息始终属于与事件类别有关的特定消息等级和消息类型。

通过“CcAeProvider.ini”文件组态 OPC 上 WinCC 消息系统的映射。

事件类别

在 WinCC OPC A&E 服务器上，为每种消息等级和类型的组合创建一个事件类别。

事件类别由类别 ID 和描述性的“类别说明”确定。类别 ID 由消息等级和消息类型的 WinCC 内部 ID 组成；类别说明由消息等级和消息类型组成。

说明

如果 OPC A&E 服务器在连通站的 WinCC 客户端上运行，则与其链接的 OS 服务器必须具有相同的消息等级和消息类型组态。否则，使用的 OPC 客户端必须直接访问 OS 服务器。

可以通过报警属性“CLASSNAME”和“TYPENAME”精准确定消息等级和消息类型的名称。

6.9.2.3 映射 WinCC 消息优先级

引言

通过 OPC 服务器的属性“Severity”来显示 WinCC 消息的优先级。

在 WinCC 消息系统中组态报警时，可以组态一个在 0 到 16 之间的优先级。OPC A&E 规范定义了 severity 的数值范围为 1 到 1000，其中 1 代表最低 severity，1000 代表最高 severity。

因此，WinCC 优先级的值相应显示为 OPC severity。在标准映射中，WinCC 优先级 0 将映射为 OPC severity 1。所有其它的优先级值都将以线性方式插入一直到 severity 1000。

可在 CcAeProvider.ini 文件中组态其它优先级映射规则。

6.9.2.4 WinCC 消息系统的属性

简介

下表列出了 WinCC 消息系统的 OPC 属性。这些属性在 WinCC 消息系统中组态。某些属性仅供 WinCC 内部使用，与 OPC A&E 客户机无关。因此未列出这些属性。

属性

OPC 属性	WinCC 消息系统	数据类型
CLASSNAME	返回消息类别名称。	VT_BSTR
TYPENAME	返回消息类型名称。	VT_BSTR
FORECOLOR	返回用于显示已接收、已发送和已确认消息的文本颜色。	VT_I4
BACKCOLOR	返回用于显示已接收、已发送和已确认消息的背景色。	VT_I4
FLASHCOLOR	返回闪烁颜色。	VT_I4
FLAGS	指示消息是否需要确认（回执）。	VT_I4
TEXT01	返回用户文本块 01 的内容。	VT_BSTR
TEXT02	返回用户文本块 02 的内容。	VT_BSTR
TEXT03	返回用户文本块 03 的内容。	VT_BSTR
TEXT04	返回用户文本块 04 的内容。	VT_BSTR
TEXT05	返回用户文本块 05 的内容。	VT_BSTR
TEXT06	返回用户文本块 06 的内容。	VT_BSTR
TEXT07	返回用户文本块 07 的内容。	VT_BSTR
TEXT08	返回用户文本块 08 的内容。	VT_BSTR
TEXT09	返回用户文本块 09 的内容。	VT_BSTR
TEXT10	返回用户文本块 10 的内容。	VT_BSTR
PROCESSVAL UE01	返回过程值块 01 的内容。	VT_VARIANT
PROCESSVAL UE02	返回过程值块 02 的内容。	VT_VARIANT
PROCESSVAL UE03	返回过程值块 03 的内容。	VT_VARIANT

OPC 属性	WinCC 消息系统	数据类型
PROCESSVAL UE04	返回过程值块 04 的内容。	VT_VARIANT
PROCESSVAL UE05	返回过程值块 05 的内容。	VT_VARIANT
PROCESSVAL UE06	返回过程值块 06 的内容。	VT_VARIANT
PROCESSVAL UE07	返回过程值块 07 的内容。	VT_VARIANT
PROCESSVAL UE08	返回过程值块 08 的内容。	VT_VARIANT
PROCESSVAL UE09	返回过程值块 09 的内容。	VT_VARIANT
PROCESSVAL UE10	返回过程值块 10 的内容。	VT_VARIANT
STATETEXT	返回状态消息。	VT_BSTR
INFOTEXT	返回消息的信息文本。	VT_BSTR
LOOPINALAR M	指示是否组态报警回路。	VT_I4
CLASSID	返回消息类别 ID。	VT_I4
TYPEID	返回消息类型 ID。	VT_I4
MODIFYSTAT E	输出消息的状态变量值。	VT_I4
AGNR	返回产生消息的自动化设备的编号。	VT_I2
CPUNR	返回产生消息的 CPU 的编号。	VT_I2
DURATION	指示消息接收、发送和确认的时间间隔。	VT_I4
COUNTER	输出自运行系统启动以来的消息数。	VT_I4
QUITSTATET EXT	指示消息是否被确认。	VT_BSTR
QUITCOUNT	输出未确认的活动消息数。	VT_I4
PARAMETER	输出消息参数。（消息组态画面）	VT_BSTR
BLOCKINFO	返回消息块的当前内容。	VT_BSTR
ALARMCOUN T	输出未决消息数。	VT_I4

OPC 属性	WinCC 消息系统	数据类型
LOCKCOUNT	输出锁定消息数。	VT_I4
PRIORITY	指示已组态的消息优先级。	VT_I4
APPLICATION	输出触发了消息的应用程序。	VT_BSTR
COMPUTER	输出处理了消息的计算机的名称。	VT_BSTR
USER	输出处理了消息的用户的名称。	VT_BSTR
COMMENT	输出消息注释。	VT_BSTR

6.9.2.5 确认方法

简介

在 WinCC 中，确认原则是消息从“进入”到“离开”的显示及处理方法。在 WinCC OPC A&E 服务器上，该消息状态用参数“ChangeMask”和“NewState”管理。

条件、简单和跟踪事件

通常将来自 WinCC 系统的消息作为条件事件发送到客户机。为了使消息作为简单事件处理，在组态消息等级时必须满足下列条件：

- “确认进入”未激活。
- “无离开状态的消息”激活。

根据映射组态，消息等级“没有确认的系统”和消息类型“操作消息”的消息将作为 OPC 跟踪事件传送。

ChangeMask

“ChangeMask”参数将记录消息状态改变的地址。

参数值：

- OPC_CHANGE_ACTIVE_STATE
- OPC_CHANGE_ENABLE_STATE
- OPC_CHANGE_ACK_STATE

NewState

“NewState” 参数指示改变后的消息状态。

参数值:

- OPC_CONDITION_ACTIVE
- OPC_CONDITION_ENABLED
- OPC_CONDITION_ACKED

概述

WinCC	NewState	ChangeState
已接收的消息	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
带回执的已发送消息	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
不带回执的已发送消息	OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
已确认的消息（消息未决）	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
已确认的消息(消息不再未决)	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
锁定的消息	----- --	OPC_CHANGE_ENABLED_STATE
解锁的消息	OPC_CONDITION_ENABLED	OPC_CHANGE_ENABLED_STATE

WinCC	NewState	ChangeState
已接收的、确认的消息	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
已接收的、带回执的已发送消息	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
已接收的、不带回执的已发送消息	OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
由系统确认的消息（消息未决）	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
由系统确认的消息（消息不再未决）	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
紧急确认的消息（消息未决）	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
紧急确认的消息（消息不再未决）	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE

参见

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.9.3 OPC A&E 质量代码

简介

质量代码用于判断消息的状态和质量。OPC 的质量代码在“数据访问自定义界面标准 3.00 版”规范的“6.8 OPC 质量标记”中有说明。

质量代码

代码	质量	状态
0xC0	OPC_GOOD	确定
0x40	OPC_UNCERTAIN	在不确定的情况下返回，例如如果发生延迟确认（回执）。
0x00	OPC_BAD	在与源的连接中断时返回。

6.9.4 OPC A&E 连接实例

6.9.4.1 OPC A&E 连接实例

简介

在下面的实例中，将组态 WinCC 和 OPC A&E 客户端之间的连接。WinCC OPC A&E 服务器使来自 WinCC 消息系统的数据变得可用。

OPC A&E 客户端通过订阅来获知 WinCC 消息的状态改变。

所有符合 OPC Alarms&Events 1.10 规范的 OPC A&E 客户端都可以访问 WinCC OPC A&E 服务器。

组态步骤

WinCC 和 OPC A&E 客户端之间的连接需要下列组态：

1. 组态对 WinCC 消息系统的访问

参见

如何组态访问 WinCC 消息系统 (页 182)

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.9.4.2 如何组态访问 WinCC 消息系统

引言

本节中，OPC 基金会的 OPC A&E 客户机将访问 WinCC 消息系统。

说明

此处描述的 OPC A&E 客户机是来自 OPC 基金会的演示客户机。源代码可以参见 Internet 网址 <http://www.opcfoundation.org>。

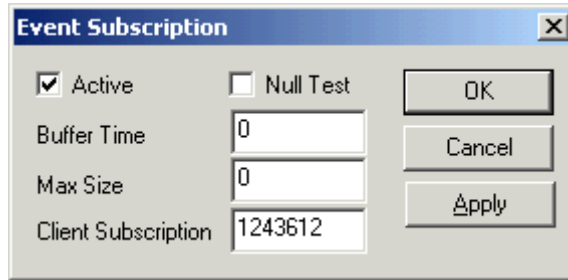
要求

- 在 WinCC OPC A&E 服务器的 WinCC 项目中创建多个“二进制”数据类型的内部变量。
- 在 WinCC OPC A&E 服务器的 WinCC 项目中组态 WinCC 消息系统。将消息链接至内部变量。
- 用图形编辑器组态画面。将 WinCC 报警控件和 I/O 域添加到画面中。将消息变量链接到图形对象。
- 在启动列表中启用“报警记录运行系统”。
- 启用 WinCC OPC A&E 服务器的 WinCC 项目。

步骤

1. 将“SampleClientAE.exe”文件从文件夹“Siemens\WinCC\documents\english”复制到您选择的文件夹中。该应用程序仅在在线帮助中可用。
2. 在菜单栏中选择“OPC” > “连接...”。在“OPC 报警服务器”对话框中选择“OPC.WinCC-AlarmsEvent”。单击“确定”关闭对话框。
3. 从菜单栏中选择“OPC” > “事件订阅...”。将打开“事件订阅”对话框。

- 选中对话框中标签为“激活”的复选框。在“缓冲时间”和“最大尺寸”域中输入“1000”。单击“确定”关闭“事件订阅”对话框。



- 来自 WinCC 消息系统的消息将显示在 OPC 事件实例客户机中。



- 从菜单栏中选择“OPC” > “过滤器”。将打开“过滤器”对话框。从“事件类别”域中选择类别。单击“确定”关闭“过滤器”对话框。
- 符合过滤标准的消息将显示在 OPC 事件实例客户机中。

“缓冲时间”和“最大大小”参数

根据 OPC 规范，在 WinCC 中按如下组态“缓冲时间”和“最大大小”参数：

OPC 客户机要求返回值	WinCC 使用
缓冲时间 < 100 OPC_S_INVALIDBUFFERTIME	修改后的缓冲时间 = 100
100 <= 缓冲时间 <= 600000 S_OK	修改后的缓冲时间 = 缓冲时间
缓冲时间 > 600000 OPC_S_INVALIDBUFFERTIME	修改后的缓冲时间 = 600000
最大大小 = 0 OPC_S_INVALIDMAXSIZE	修改后的最大大小 = 1000
0 < 最大大小 < 10 OPC_S_INVALIDMAXSIZE	修改后的最大大小 = 10

OPC 客户机要求返回值	WinCC 使用
10 <= 最大大小 <= 1000 S_OK	修改后的最大大小 = 最大大小
最大大小 = 1000 OPC_S_INVALIDMAXSIZE	修改后的最大大小 = 1000

创建用户时，可以设置参数。然而，在创建之后，不能使用 **SetState()** 修改已存在的用户。

更多的信息，请参阅 <http://www.opcfoundation.org>。

参见

www.opcfoundation.org (<http://www.opcfoundation.org>)

6.9.5 带层级访问的 OPC A&E 服务器

6.9.5.1 OPC A&E 服务器的功能

简介

OPC-A&E 服务器使用 DCOM 服务在具有 OPC 能力的应用程序之间传送消息。OPC A&E 服务器支持规范 OPC Alarm&Event 1.10。

下一章说明 WinCC 消息系统在带层级访问的 OPC A&E 上的映射，以及 OPC A&E 服务器支持的属性。本文档包括对特定信息的概述。有关更多信息，可参见“OPC 报警 & 事件 1.10”规范。

工作原理

OPC-A&E 客户端通过订阅接收 WinCC 消息。可以使用订阅过滤器减少随订阅传送的事件数。OPC-A&E 客户端可针对每个显示消息属性的事件类别进行设置。

安装

可以在安装 WinCC 期间选择 WinCC OPC A&E 服务器。安装后，WinCC OPC A&E 服务器无需进行组态，立即可用。

可以在 WinCC 服务器和 WinCC 客户端上安装 WinCC OPC A&E 服务器。

许可

为了运行 WinCC OPC A&E 服务器，必须在每台作为 OPC A&E 服务器的 WinCC 服务器上安装下列许可证：

- 有效的 WinCC RT 许可证
- WinCC 选件连通性软件包

事件类型

带层级访问的 OPC A&E 服务器支持条件事件、简单事件和跟踪事件。

与条件有关的事件

对于与条件有关的事件，事件会与某个条件相关联。例如，条件可以是变量超出限值。这种违反限制的情况会产生一条消息，该消息在 OPC A&E 中显示为报警。

简单事件

简单事件是提示 OPC A&E 客户端关于事件的消息。简单事件包括启动或退出程序等。

说明

在使用冗余系统时，请注意如下事项：

更新变量时，与内部变量互连的简单事件会发送两次。

第一个消息由主站触发，第二个消息由备用站触发。

跟踪事件

跟踪事件连同操作员输入消息一并发送到 OPC A&E 客户端。通过在过程中进行人工干预触发操作员输入消息。

OPC A&E 客户端

所有符合 OPC Alarms & Events 1.10 规范的 OPC A&E 客户端都可以访问 OPC A&E 服务器。也可自己创建 OPC A&E 客户端。通过创建专有 OPC 客户端，可以满足大多数用户特定需求。例如，可使用 OPC A&E 客户端对来自不同 OPC A&E 服务器的报警进行分析和联合归档。无法对已归档消息进行确认；只能对当前报警和事件进行确认。

如果使用带层级访问功能的 OPC A&E，并想要使用所有功能，则可能需要修改当前使用的 OPC A&E 客户端。

说明

OPC 上的文档

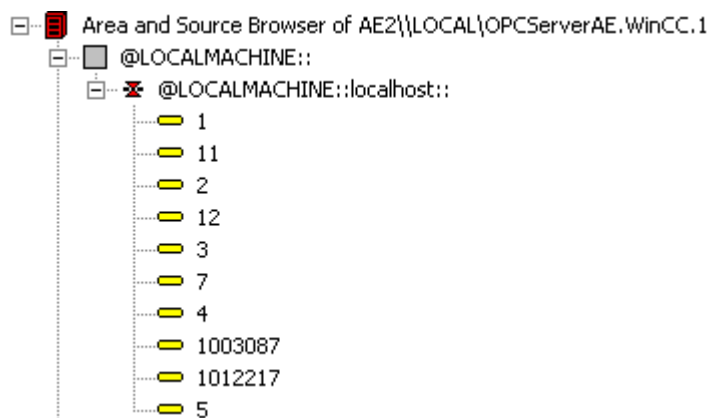
可以在“接口 > OPC - 用于过程控制的 OLE”一章中找到有关 OPC 的附加信息。

6.9.5.2 OPC A&E 与带层级访问的 OPC A&E 之间的差异

使用 OPC A&E 显示消息

OPC A&E 服务器支持用于访问消息系统的“条件事件”和“简单事件”。对于“条件事件”，将针对每个来源显示消息号。由于一台 WinCC 服务器可容纳许多消息号，因此难以维护消息概览。

下图显示的是 OPC 浏览器中的显示实例：



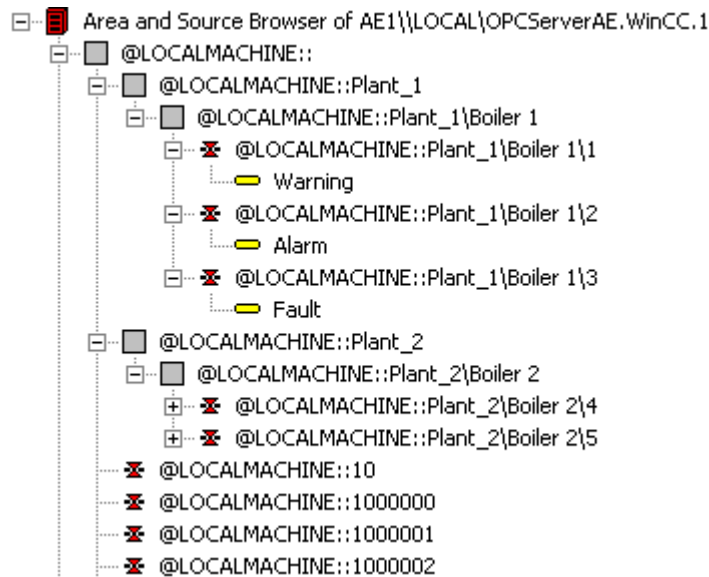
使用 OPC A&E 和层级访问显示消息

带层级访问功能的 OPC A&E 服务器支持的事件类型有条件事件、简单事件和跟踪事件。

用户文本块 2 决定“条件事件”消息的来源。使用默认设置时，用户文本块 2 对应于故障位置。要以层级形式呈现消息，必须将这些消息合并到报警记录消息的用户定义组消息中。组消息的结构由 OPC A&E 中的区域决定。

在系统中触发操作员输入消息时，会发生跟踪事件。

下图显示的是 OPC 浏览器中条件事件的显示实例。除了“区域”和“来源”外，还显示了“条件”：



切换到带层级访问功能的 OPC A&E

请在创建新项目时使用带层级访问功能的 OPC A&E 服务器。

在现有项目中，OPC A&E 服务器可像以前那样使用，也可进行转换从而实现层级访问功能。可再次撤消转换而不会丢失任何数据。

1. 将“CcAeProvider.ini”文件复制到项目文件夹中。该文件位于 WinCC 安装路径的“OPC \AlarmEvent\Hierarchical-Access”文件夹中。
2. 更新客户端或对 OS 服务器执行完整下载。

6.9.5.3 在 OPC A&E 上映射 WinCC 消息系统

映射 WinCC 消息系统

简介

因进行组态而产生的 WinCC 消息系统定义过程中的哪个事件将会生成消息。此消息在 OPC A&E 中显示为事件通知。

在带层级访问功能的 OPC A&E 上映射 WinCC 消息系统

WinCC 用户文本块 2 的 OPC 源和 WinCC 用户文本块 1 的 OPC 消息在 WinCC 中用作映射 WinCC 消息系统的默认设置。

概述

下表显示最重要的事件通知属性及来自 WinCC 消息系统的相应信息。

使用已组态属性的事件显示在表的第三列中：

- “S” 表示简单事件
- “C” 表示条件事件
- “T” 表示跟踪事件

OPC	WinCC 消息系统	事件类型
Area	组消息的结构决定 OPC A&E 中的区域。如果没有为消息组态组消息，则只有与服务器前缀对应的 OPC 区域可用。	S、C、T
Source	指示消息源。消息源的格式为“<服务器前缀>::Area\用户文本块 2”。本地计算机的服务器前缀为“@LOCALMACHINE”。服务器前缀始终显示服务器层级中的顶级 Areas。	S、C、T
Time	发布已接收、已发送和已确认消息的时间戳。以 UTC（协调世界时）格式发布时间戳。	S、C、T
Type	指示该事件是简单事件、跟踪事件还是条件事件。	S、C、T
Severity	返回消息的优先级。	S、C、T
EventCategory	指示消息类别。“事件类别”由“类别 ID”和“类别说明”组成。“类别 ID”对应于消息类别的内部 ID。“类别说明”对应于消息类别的名称。	S、C、T
Message	指示相应消息号的消息文本。	S、C、T
Condition	指示消息类型。	C
Sub-condition	对应于“Condition”参数。	C
ChangeMask	指定条件变化。有关详细信息，请参阅“确认理论”。	C
NewState	指示条件的当前状态。有关详细信息，请参阅“确认理论”。	C
ConditionQuality	返回消息质量。有关详细信息，请参阅“质量代码”。	C
AckRequired	指示消息是否需要确认。	C
EventAttribute	列出相关消息所需的属性。有关详细信息，请参阅“WinCC 消息系统的属性”。	C
Quality	返回消息的质量代码。	C

OPC	WinCC 消息系统	事件类型
Cookie	不包括客户机的任何可用信息	C
ActorID	指示确认消息的用户。	T

说明

如果将不带通配符的文本指定为该区域的过滤器，则只返回该区域的消息。如果想要包括位于该指定区域以外区域中的源，则需要使用通配符。

说明

如果如下所示运行 OPC A&E 服务器，则**必须**在所连接的 OS 服务器上对消息类别和消息类型进行完全相同的组态：

- 在 WinCC 客户机上
- 在连通站上

如果 OS 服务器未进行完全相同的组态，所使用的 OPC 客户机必须直接访问相应的 OS 服务器。

映射消息优先级

导言

消息优先级由 OPC A&E 服务器映射到属性“Severity”。

在消息传递系统中组态报警时，可以组态“0”和“16”之间的优先级。OPC A&E 规范为严重性定义的取值范围是“1”到“1000”。在这种情况下，“1”代表严重性最低，“1000”代表严重性最高。

因此，优先级的值相应显示为 OPC 严重性。在标准映射中，优先级“0”指定给 OPC 严重性“1”，优先级“16”指定给 OPC 严重性“1000”。所有其它优先级值在“0”与“1000”之间进行线性插值。

WinCC 消息系统的属性

简介

下表列出了 WinCC 消息系统的 OPC 属性。这些属性在 WinCC 消息系统中组态。某些属性仅供 WinCC 内部使用，与 OPC A&E 客户机无关。因此，这些属性未包含在该表中。

属性

OPC 属性	WinCC 消息系统	数据类型
CLASSNAME	输出消息类别名称。	VT_BSTR
TYPENAME	输出消息类型名称。	VT_BSTR
FORECOLOR	输出已激活、已禁用和已确认消息的文本颜色。	VT_I4
BACKCOLOR	输出已激活、已禁用和已确认消息的背景色。	VT_I4
FLASHCOLOR	输出闪烁颜色。	VT_I4
FLAGS	指示必须确认该消息	VT_I4
TEXT01	输出用户文本块 01 的内容。	VT_BSTR
TEXT02	输出用户文本块 02 的内容。	VT_BSTR
TEXT03	输出用户文本块 03 的内容。	VT_BSTR
TEXT04	输出用户文本块 04 的内容。	VT_BSTR
TEXT05	输出用户文本块 05 的内容。	VT_BSTR
TEXT06	输出用户文本块 06 的内容。	VT_BSTR
TEXT07	输出用户文本块 07 的内容。	VT_BSTR
TEXT08	输出用户文本块 08 的内容。	VT_BSTR
TEXT09	输出用户文本块 09 的内容。	VT_BSTR
TEXT10	输出用户文本块 10 的内容。	VT_BSTR
PROCESSVALUE 01	输出过程值块 01 的内容。	VT_VARIANT
PROCESSVALUE 02	输出过程值块 02 的内容。	VT_VARIANT
PROCESSVALUE 03	输出过程值块 03 的内容。	VT_VARIANT
PROCESSVALUE 04	输出过程值块 04 的内容。	VT_VARIANT
PROCESSVALUE 05	输出过程值块 05 的内容。	VT_VARIANT
PROCESSVALUE 06	输出过程值块 06 的内容。	VT_VARIANT
PROCESSVALUE 07	输出过程值块 07 的内容。	VT_VARIANT

OPC 属性	WinCC 消息系统	数据类型
PROCESSVALUE 08	输出过程值块 08 的内容。	VT_VARIANT
PROCESSVALUE 09	输出过程值块 09 的内容。	VT_VARIANT
PROCESSVALUE 10	输出过程值块 10 的内容。	VT_VARIANT
STATETEXT	输出状态消息。	VT_BSTR
INFOTEXT	输出消息信息文本。	VT_BSTR
LOOPINALARM	指示是否组态报警回路。	VT_I4
CLASSID	输出消息类别 ID。	VT_I4
TYPEID	输出消息类型 ID。	VT_I4
MODIFYSTATE	输出消息的状态变量值。	VT_I4
AGNR	输出产生消息的 AS 的编号。	VT_I2
CPUNR	输出产生消息的 CPU 的编号。	VT_I2
DURATION	输出激活、禁用和确认消息之间的时间间隔。	VT_I4
COUNTER	输出自运行系统启动以来的消息数。	VT_I4
QUITSTATETEXT	指示消息是否被确认。	VT_BSTR
QUITCOUNT	输出未确认的活动消息数。	VT_I4
PARAMETER	输出消息参数。(消息组态画面)	VT_BSTR
BLOCKINFO	输出消息块的当前内容。	VT_BSTR
ALARMCOUNT	输出未决消息数。	VT_I4
LOCKCOUNT	输出锁定消息数。	VT_I4
PRIORITY	指示已组态的消息优先级。	VT_I4
APPLICATION	输出触发了消息的应用程序。	VT_BSTR
COMPUTER	输出处理了消息的计算机的名称。	VT_BSTR
USER	输出处理了消息的用户的名称。	VT_BSTR
COMMENT	输出消息注释。	VT_BSTR
HIDDEN-COUNT	输出隐藏消息数。	VT_I4
BIG COUNTER	输出自运行系统启动以来的消息数。	VT_CY
OS-HIDDEN	输出消息的隐藏状态。	VT_BOOL
OS-EVENTID	输出消息的已组态消息 ID。	VT_I4

确认方法

导言

WinCC 中的确认策略是如何显示和处理消息从“进入”到“离开”的过程。在 OPC A&E 服务器上，该消息状态用“ChangeMask”和“NewState”参数来显示。

条件事件、简单事件和跟踪事件

来自系统的消息将作为有确认的条件事件发送到客户机。

为了使消息作为简单事件接受处理，消息的消息等级必须符合下列条件：

- “确认进入”未激活。
- “无离开状态的消息”激活。

在 WinCC 中，消息等级“系统，不需要确认”和消息类型“操作员输入消息”的消息作为跟踪事件传送。

说明

具有“系统，不需要确认”消息等级和“过程控制系统”消息类型的消息作为“系统消息”事件类别的简单事件进行传送。

ChangeMask

“ChangeMask”参数将记录消息状态改变的地址。

参数值：

- OPC_CHANGE_ACTIVE_STATE
- OPC_CHANGE_ENABLE_STATE
- OPC_CHANGE_ACK_STATE

NewState

“NewState”参数指示改变后的消息状态。

参数值:

- OPC_CONDITION_ACTIVE
- OPC_CONDITION_ENABLED
- OPC_CONDITION_ACKED

概述

WinCC	NewState	ChangeState
已接收的消息	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
有确认的离开消息	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
无确认的离开消息	OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
已确认的消息（消息未决）	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
已确认的消息(消息不再未决)	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
锁定的消息	----- --	OPC_CHANGE_ENABLED_STATE
解锁的消息	OPC_CONDITION_ENABLED	OPC_CHANGE_ENABLED_STATE

WinCC	NewState	ChangeState
已确认的进入消息	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACTIVE_STATE
有确认的进入、离开消息	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
无确认的进入、离开消息	OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
由系统确认的消息（消息未决）	OPC_CONDITION_ACTIVE OPC_CONDITION_ACKED OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE
由系统确认的消息（消息不再未决）	OPC_CONDITION_ACTIVE OPC_CONDITION_ENABLED	OPC_CHANGE_ACK_STATE

WinCC	NewState	ChangeState
紧急确认的消息（消息未决）	OPC_CONDITION_ACTIV E OPC_CONDITION_ACKED OPC_CONDITION_ENABL ED	OPC_CHANGE_ACK_STAT E
紧急确认的消息（消息不再未决）	OPC_CONDITION_ACTIV E OPC_CONDITION_ENABL ED	OPC_CHANGE_ACK_STAT E

说明

历史报警和事件不会被确认。 OPC A&E 历史事件接口仅有读取访问权限。

6.9.5.4 OPC A&E 质量代码**简介**

质量代码用于判断消息的状态和质量。 OPC 的质量代码在“数据访问自定义界面标准 3.00 版”规范的“6.8 OPC 质量标记”中有说明。

质量代码

代码	质量	状态
0xC0	OPC_GOOD	确定
0x40	OPC_UNCERTAIN	在不确定的情况下返回，例如：发生延迟确认（回执）时。
0x00	OPC_BAD	在与源的连接中断时返回。

6.9.6 读取归档消息

6.9.6.1 访问归档消息

简介

可以使用 OPC 客户机，通过 OPC A&E 服务器访问归档消息。支持使用以下两种方法来访问归档消息：

- 输出过去某个时间段的归档消息
- 输出过去某个时间段的归档消息，但不指定截止时间。输出归档消息后，其它所有新生成的消息会自动发送到 OPC 客户机。

说明

读取归档消息后，无法使用消息的已返回“ActiveTime”来确认该消息或跟踪消息转变。为确保这一点，OPC A&E 客户机必须检查带有额外标记“OPC_HAE_HISTORICAL_EVENTFLAG”的消息的“EventType”。归档消息的“ActiveTime”是不正确的。可以在“标识归档消息”下找到有关额外标记的信息。

查询“历史报警和事件”功能

除了标准过滤器外，WinCC 的扩展 OPC A&E 服务器还随附了以下过滤器：

滤波器	过滤器的值	描述
OPC_HAE_FILTER_BY_TIMEFRAME	0x80000000	与用于 OPC 历史数据访问的“ReadRaw”函数对应
OPC_HAE_FILTER_BY_STARTTIME	0x40000000	与用于 OPC 历史数据访问的“AdviseRaw”函数对应

源过滤器和历史报警请求

要请求归档消息，OPC 客户机必须支持“SetFilter”的订阅功能。如果将关键字“OPCHAEServer”也插入到订阅的“源过滤器”序列中，则 OPC 服务器还会发送归档消息。除此关键字外，还可使用其它参数定义要读取的消息：

- 方法
- 时间段
- 有限制或没有限制

在过滤器中指定的源列表除了“OPCHAEServer”源以外，还可以包括其它源名称。在此情况下，订阅将仅传送给定来源的历史事件。源名的顺序不连贯。

组态源过滤器后，所选时间段可从客户机上通过“Refresh”调用进行调用。

6.9.6.2 使用 OPC 访问归档消息的语法

语法

```
OPCHAEServer hMode=(read|advise) htStartTime=szTime  
[hEndTime=szTime] [bBounds=(TRUE|FALSE)]
```

参数

hMode = [read|advise]

该参数是必需的。定义如何读取归档消息和事件。

Read: 输出过往定期的归档消息和事件（如果是 OPC 历史数据访问，则可与 **ReadRaw** 相比）。

下面是设置用于在最后 30 分钟进行读取的过滤器的实例：

```
OPCHAEServer hMode=read htStartTime=NOW-30M bBounds=TRUE
```

Advise: 输出某个定期的归档消息和事件。接收所有归档消息后，将以与激活订阅情况下相同的方式发送新消息（相当于 OPC 历史数据访问情况下的 **AdviseRaw**）。

在下列实例中，将读取最后 30 分钟的消息（订阅必须激活）：

```
OPCHAEServer hMode=advise htStartTime=NOW-30M
```

说明

参数“htStartTime”和“htEndTime”支持以下符号：

- 相对符号，例如 NOW
- 符号值，例如 NOW、YEAR、MONTH
- 根据 XML 符号指定绝对 UTC 数据/时间值：2006-09-01T10:00:00.000Z

使用符号表示法对应于 OPC 历史数据访问语法。

htStartTime =

该参数是必需的。定义从归档中读取消息和事件的开始时间。

htEndTime =

该参数是可选的。定义从归档中读取消息和事件的结束时间。如果“hMode = read”，使用缺省设置“NOW”。

bBounds = [TRUE|FALSE]

该参数是可选的。定义如何处理接近开始时间和结束时间的消息。此功能与 OPC 历史数据访问完全相同。

bBounds=FALSE:

- 传送的第一个消息的时间标志 \geq htStartTime
- 传送的最后一个消息的时间标志 \geq htEndTime

bBounds=TRUE:

- 传送的第一个消息的时间标志 \leq htStartTime
- 传送的最后一个消息的时间标志 \geq hEndTime

缺省设置为 FALSE。

6.9.6.3 归档消息的读方法

引言

可以使用两种读模式之一来读取归档消息：

- 读
- 建议

“读”模式

“读”模式用于读取过去某个已定义时间段的归档消息。读取消息的顺序始终根据每个 OS 服务器按时间顺序从报警进行读取。通过设置开始和结束时间，可以指定是最先还是最后读取最后一条消息。如果开始时间早于结束时间，将最后输出最后一条消息。

要使用“读”模式，需要对订阅运行下列函数：

1. SetFilter
2. Refresh

“Refresh”期间运行的“SetFilter”会被拒绝。在“Refresh”期间激活订阅不会对刷新造成任何影响。

将继续以 **Refresh** 标记传送历史事件。

根据激活订阅的标准反应来传送新近生成的事件：

- 考虑除“历史”源“OPCHAEServer”外的 **set filter** 值
- 没有 **Refresh** 标记

客户机因而可以根据 **Refresh** 标记区分已接收的事件。事件数据包从未同时包含历史事件和新事件。

- 具有 **Refresh** 标记的事件数据包仅包含历史事件。这些事件还可以按队列排列。
- 没有 **Refresh** 标记的事件数据包仅包含新近生成的事件。

“建议”模式

“建议”模式用于读取过去某个已定义时间段之后的归档消息。读取所有归档消息后，将按照操作已激活订阅的方式发送新消息。将根据每台 OS 服务器按时间顺序传送归档消息：将传送开始时间以后的归档消息。然后，将传送新近的归档消息。

请注意，不应为“建议”定义结束时间。

激活的订阅将用于“建议”模式。如果对激活的订阅运行“**SetFilter**”函数，则会立即传送历史报警。

如果对未激活的订阅运行“**SetFilter**”函数，则只会在激活订阅后发送归档消息。如果想要对未激活的订阅使用“建议”读模式，请按如下步骤操作：

1. **SetFilter**
2. 使用 **SetState** 将订阅设置为激活

如果将订阅取消激活，将中断传送。

如果将订阅设置为“未激活”，则将结束传送。订阅激活时，“**SetFilter**”会被拒绝。

在“建议”模式下对激活的“历史”订阅运行“**Refresh**”时，其作用与其对标准订阅的作用相同：

所有排队等候的与条件相关的事件都将传送到具有 **Refresh** 标记的数据包。最后一个数据包还包含“**Last Refresh**”附加标记。

“**Refresh**”调用对在“建议”模式下读取历史报警没有影响。

6.9.6.4 识别归档消息

一般步骤

使用 EventType 中的附加标记来区分归档消息。该标记通过 OR 链接来链接到有效的 EventType。

名称	EventType	EventType (归档消息)
OPC_SIMPLE_EVENT	0x01	0x81
OPC_CONDITION_EVENT	0x04	0x84
OPC_TRACKING_EVENT	0x02	0x82
OPC_HAE_HISTORICAL_EVENTFL AG		0x80

实例

实例 1

下列源过滤器用于以“读取”模式输出前 30 分钟的归档消息和事件。每台 OS 服务器日期最早的消息将第一个输出。也会发送下限值。

```
OPCHAESEServer hMode=read htStartTime=NOW-30M bBounds=TRUE
```

实例 2

下列源过滤器用于以“读”模式输出 2006 年 9 月 1 日 10:00 至 12:00 的归档事件。每台 OS 服务器的最新消息将第一个输出。也会发送此时间段的极限。

```
OPCHAESEServer hMode=read htStartTime=2006-09-01T12:00:00.000Z  
htEndTime=2006-09-01T10:00:00.000Z bBounds=TRUE
```

实例 3

下列源过滤器用于以“建议”模式输出前 30 分钟的归档消息和事件。读取归档消息后，将按照操作已激活订阅的方式发送新近生成的消息。

```
OPCHAESEServer hmode=advise htStartTime=NOW-30M
```


6.10 WinCC OPC UA 服务器

6.10.1 WinCC OPC UA 服务器的工作原理

工作原理

WinCC OPC UA 服务器提供以下值：

- 过程值
- 变量归档的值
- WinCC 消息

WinCC OPC UA 服务器作为 Windows 服务安装，并自动启动。WinCC OPC UA 服务器仅支持“UA-TCP UA-SC UA Binary”通信配置文件。所使用的端口号可调整。

支持的规范

OPC Unified Architecture 是用于传送过程值、归档数据和消息的规范。WinCC OPC UA 服务器支持 OPC UA 规范 1.02。有关支持的 UA 功能的详细信息，请参见“支持的 OPC UA 服务和配置文件 (页 209)”。

安装

安装 WinCC 后，无需进一步组态，即可立即使用 WinCC OPC UA 服务器。
WinCC OPC UA 服务器可在 WinCC 服务器或 WinCC 客户端上使用。

WinCC OPC UA 服务器的 URL

通过以下 URL 访问 WinCC OPC UA 服务器：

- "opc.tcp://[HostName]:[Port]"

参数	说明
HostName	计算机名称的占位符。自动使用。
Port	端口号。默认设置为“4862”。

Discovery 服务器

“Discovery 服务器”由 OPC 基金会提供。默认情况下，“Discovery 服务器”作为 Windows 服务安装在 HMI 设备上。

在“Discovery 服务器”上，可通过 OPC UA 服务器获得在“Discovery 服务器”上注册的 UA 客户端信息。

根据组态，当运行系统启动时，WinCC OPC UA 服务器在一个或多个已组态的可用“Discovery 服务器”上注册或不在任何服务器上注册。之后，会周期性重复注册。如果结束运行系统，WinCC OPC UA 服务器会自动从“Discovery 服务器”注销。

WinCC 地址区域中支持的语言

WinCC OPC A&E 服务器支持以下几种语言的 WinCC 地址区域：

- 德语
- 英语
- 法语
- 意大利语
- 西班牙语

6.10.2 OPC UA 的安全概念

简介

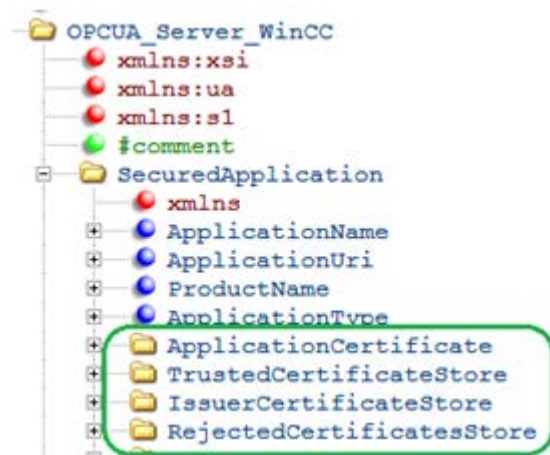
OPC UA 安全概念主要基于：

- 相关应用程序和用户的验证和授权
- 确保应用程序之间交换的消息的完整性和保密性

证书

证书是用于验证 OPC UA 应用程序的方法。每个应用程序都有自己的实例证书，通过该证书在公共密钥基础设施中标识自身。

WINCC OPC UA 服务器使用的证书存储在“OpcUaServerWinCC.xml”组态文件的设置中：



WinCC OPC UA 服务器的实例证书

为确保安全运行，每个 WinCC OPC UA 服务器都需要自己的带有私钥的实例证书。证书仅在相应的计算机上有效，且只能由该计算机上安装的 WINCC OPC UA 服务器使用。

安装服务器时，将为服务器生成自签名实例证书，并将其保存在服务器证书文件夹和操作系统的证书存储器中。

证书的私钥仅存储在证书文件夹中。必须将带私钥的文件夹的访问权限限制给：

- 服务器自身
- 系统管理员

注意

访问带私钥的文件夹

出于安全原因，除了服务器和系统管理员以外，其它用户或应用程序均不能访问 WINCC OPC UA 服务器的私钥。

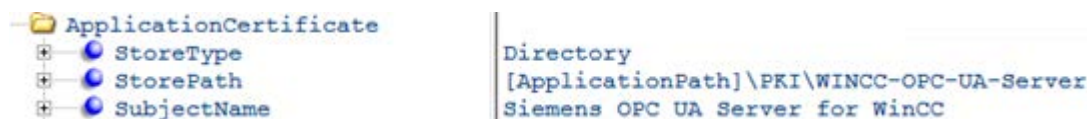
安装时生成的实例证书和相应的私钥可由系统管理员替换。根据适用的系统安全概念，新的实例证书可以自签名，也可由认证机构颁发。

6.10 WinCC OPC UA 服务器

WinCC OPC UA 服务器实例证书的存储位置在服务器组态文件的设置中进行定义。 如果需要，系统管理员可更改存储位置：

参数	值	含义
StoreType	Directory	证书存储类型。 存储位置必须为“目录”。
StorePath	[ApplicationPath]\PKI\WINCC-OPC-UA-Server	证书和私钥存储在此文件夹下。

实例证书组态示例



在此例中，服务器的实例证书存储在“...PKI\WINCC-OPC-UA-Server\Certs”目录，私钥存储在“...PKI\WINCC-OPC-UA-Server\Private”目录。

受信任的客户端证书

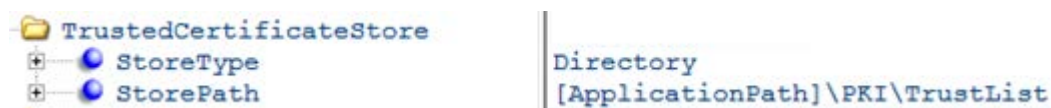
WinCC OPC UA 服务器仅支持与受信任的客户端进行安全通信。 客户端在以下情况下受到信任：

- 客户端具有有效的自签名证书，且该证书存储在 WinCC OPC UA 服务器的受信任证书存储器中时
- 或有效的客户端证书由认证机构颁发时。 来自认证机构的有效证书必须位于 WinCC OPC UA 服务器的受信任证书存储器中。 在此情况下，仅需要来自认证机构的证书。 客户端实例证书不需要位于受信任证书存储器中。

通过 WINCC OPC UA 服务器组态文件指定受信任证书的存储设置：

参数	含义
StoreType	证书存储类型。 存储位置可以是“目录”或“Windows”。
StorePath	受信任客户端的证书存储在此文件夹下。

使用“目录”存储的组态示例



在此例中，WINCC OPC UA 服务器信任所有在“...PKI\TrustList\Certs”文件夹中存储实例证书的客户端。

使用“Windows”存储的组态示例



对于此存储选项，客户端的实例证书必须位于“<本地计算机>\UA applications”下的操作系统证书存储器中。

验证客户端证书链所需的认证机构证书存储在认证机构的证书存储器中。在这里也使用 WINCC OPC UA 服务器组态文件指定存储设置：

参数	含义
StoreType	证书存储类型。存储位置可以是“目录”或“Windows”。
StorePath	受信任的认证机构的证书存储在此文件夹下。

说明

来自认证机构存储器的证书并非自动受到信任。

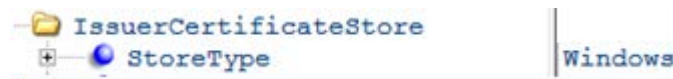
对于受信任的认证机构，其证书必须位于受信任证书存储器中。

使用“目录”存储的组态示例



在此例中，受信任的认证机构的证书位于“...PKI\CA\Certs”文件夹。

使用“Windows”存储的组态示例



“StorePath”参数不相关。来自认证机构的证书必须存储在 Windows 证书存储器中，以符合操作系统要求。

如果证书位于以下两个位置之一，则受到信任：

- <本地计算机>\受信任的根认证机构
- <本地计算机>\第三方根认证机构

说明

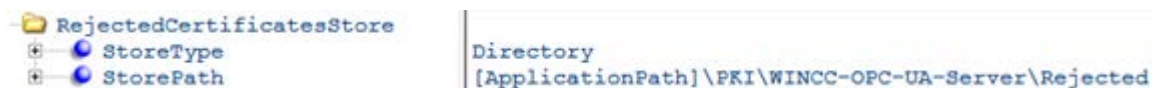
存储的重要信息

- 服务器证书的存储位置必须为“目录”。
 - 受信任客户端证书的存储位置和来自认证机构的证书的存储位置必须具有相同的 StoreType，即，必须都为“目录”或“Windows”。
-

客户端证书不被接受

如果 UA 客户端访问 WINCC OPC UA 服务器时没有受信任的证书，WINCC OPC UA 服务器不会允许安全通信，并将该客户端证书复制到存放被拒绝证书的文件夹。

通过 WINCC OPC UA 服务器组态文件指定被拒绝证书的存储设置，例如



说明

这里也仅支持 StoreType“目录”。

要启用与此客户端的安全通信，必须将被拒绝的证书移至受信任证书的证书存储器中。

6.10.3 组态安全机制

简介

在通信级别应确保：

- UA 应用程序真实性
- 所交换消息的保密性
- 所交换消息的完整性

所使用的安全机制（例如加密和签名算法）在标准化安全策略中进行设置。

WinCC OPC UA 服务器支持的安全策略通过“ServerConfiguration”和“SecuredApplication”中的服务器组态文件进行设置。

ServerConfiguration

“ServerConfiguration” 下的 XML 元素“SecurityPolicies” 包含用于服务器的所有可用的“Security Profile” 和“Message Security Mode” 组合的列表。

Security Profile	Message Security Mode	说明
http://opcfoundation.org/UA/SecurityPolicy#None	None	不安全通信
http://opcfoundation.org/UA/SecurityPolicy#Basic128Rsa15	Sign 或 SignAndEncrypt	安全通信, 已签名或已加密并签名的消息
http://opcfoundation.org/UA/SecurityPolicy#Basic256	Sign 或 SignAndEncrypt	安全通信, 已签名或已加密并签名的消息

说明

确保安全通信

安全通信需要服务器和客户端实例证书, 以及正确组态的证书存储器。

具有最大功能范围的组态文件示例

The screenshot shows a tree view of the OPCUA_Server_WinCC configuration. The 'ServerConfiguration' node is expanded to show 'SecurityPolicies'. Under 'SecurityPolicies', there are five 'SecurityPolicy' entries. Each entry has a 'ProfileUri' and 'MessageSecurityModes' property. The corresponding XML values are listed in the adjacent pane:

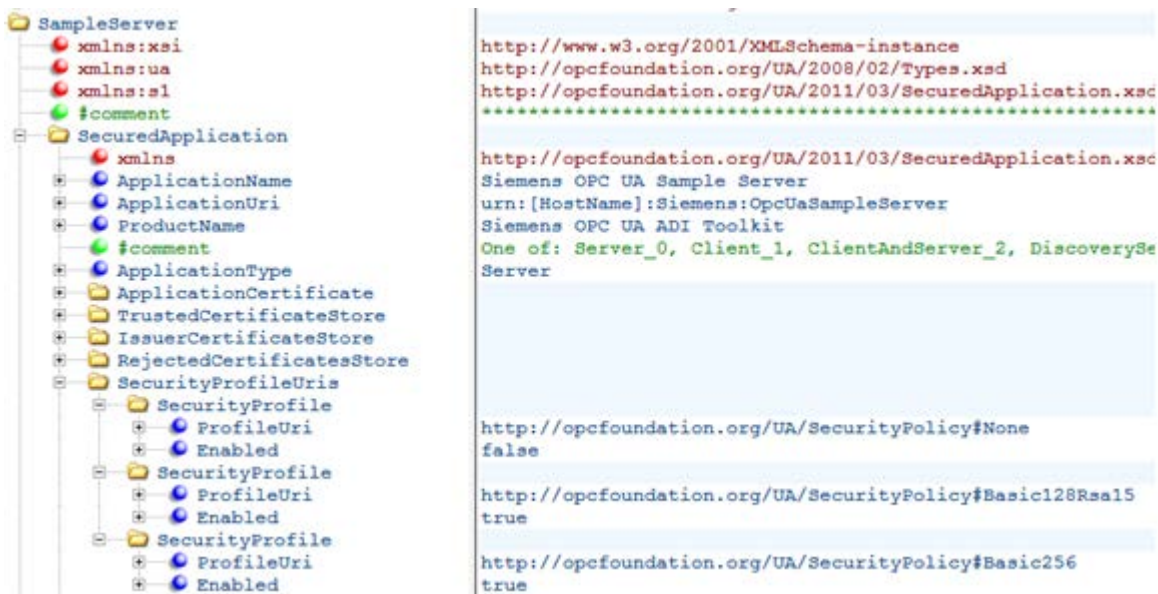
- ProfileUri: http://www.w3.org/2001/XMLSchema-instance
- ProfileUri: http://opcfoundation.org/UA/2008/02/Types.xsd
- ProfileUri: http://opcfoundation.org/UA/2011/03/SecuredApplication.xsd
- ProfileUri: http://opcfoundation.org/UA/SecurityPolicy#None
- ProfileUri: http://opcfoundation.org/UA/SecurityPolicy#Basic128Rsa15
- ProfileUri: http://opcfoundation.org/UA/SecurityPolicy#Basic128Rsa15
- ProfileUri: http://opcfoundation.org/UA/SecurityPolicy#Basic128Rsa15
- ProfileUri: http://opcfoundation.org/UA/SecurityPolicy#Basic256
- ProfileUri: http://opcfoundation.org/UA/SecurityPolicy#Basic256
- ProfileUri: http://opcfoundation.org/UA/SecurityPolicy#Basic256

The Message Security Modes for the policies are: None, Sign, SignAndEncrypt, Sign, and SignAndEncrypt.

SecuredApplication

根据 OPC UA 规范, 使用“SecuredApplication” 下的“SecurityProfileUris” 元素显式启用和禁用安全机制。

下图显示了一个禁用不安全通信的 SecuredApplication:

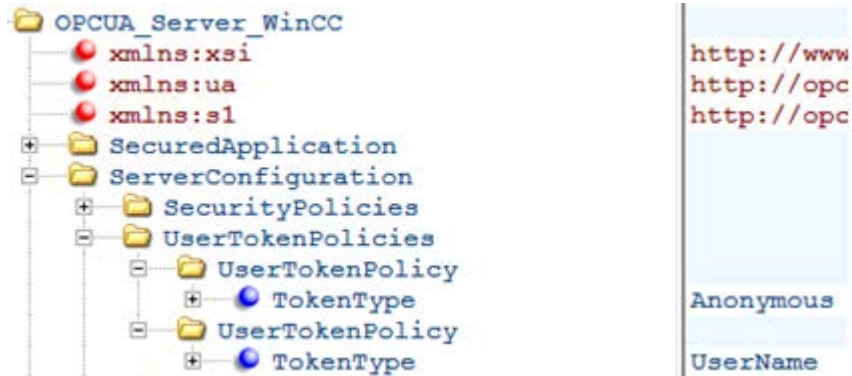


因此，WinCC OPC UA 服务器支持运行系统中的两个安全策略“Basic128Rsa15”和“Basic256”。虽然具有“Message Security Modes Sign”和“SignAndEncrypt”，但不是不安全通信。建立通信时，UA 客户端从此列表中选择所需的 Policy。

用户标识

同通信级别的安全机制一样，WINCC OPC UA 服务器也支持使用 UserTokenPolicy“UserName”进行客户端应用程序的用户验证。建立通信时，客户端应用程序必须提供有效的用户名和密码组合。WinCC OPC UA 服务器在操作系统的用户管理中验证该组合。

UserTokenPolicy 在 WINCC OPC UA 服务器的组态文件中进行设置。



通过该组态，WINCC OPC UA 服务器支持匿名用户和 Policy“UserName”。

6.10.4 支持的 OPC UA 服务和配置文件

OPC UA 服务

WinCC OPC A&E 服务器支持以下描述的功能。

下表列出了 OPC UA 服务器 1.0.2 支持的功能：

OPC UA Service Sets	Services	注释
Discovery Service Set	FindServers GetEndpoints	-
Secure Channel Service Session Service Set	全部	-
View Service Set	Browse BrowseNext RegisterNodes UnregisterNodes	检测映射的 WinCC 数据： 过程值和归档数据
Attribute Service Set	Read Write HistoryRead HistoryUpdate ^{*)}	仅 WinCC 变量 仅 WinCC 变量 仅归档的变量 仅归档的变量
Subscription Service Set	CreateSubscription SetPublishingMode Publish RePublish DeleteSubscription	
MonitoredItem Service Set	CreateMonitoredItems SetMonitoringMode DeleteMonitoredItems	仅 WinCC 变量的“Value” 属性 访问 WinCC 消息时的 .EventNotifier
Method Service Set	Call	Acknowledge ConditionRefresh
*)：有关限制的信息，请参见“支持的写访问 (页 161)”		

OPC UA 配置文件和 Conformance Units

WinCC OPC UA 服务器支持以下 OPC UA 配置文件 1.02，且没有限制：

- 6.5.3 基础服务器行为方面
- 6.5.12 标准事件订阅服务器方面
- 6.5.14 A & C 基础条件服务器方面
- 6.5.24 方法服务器方面
- 6.5.30 历史原始数据服务器方面
- 6.5.36 历史数据更新服务器方面
- 6.5.37 历史数据插入服务器方面
- 6.5.38 历史数据删除服务器方面
- 6.5.107 UA TCP UA SC UA 二进制
- 6.5.125 SecurityPolicy - Basic256
- 6.5.124 SecurityPolicy - Basic128Rsa15
- 6.5.123 SecurityPolicy - 无

WinCC OPC A&E 服务器支持下表中列出的 OPC UA 配置文件，但是有限制：

Profile	"Group"	不支持 "Conformance Unit"
6.5.8 Standard DataChange Subscription Server Facet	Monitored Item Services	ModifyMonitoredItems DeadBand Filter Monitor MinQueueSize_02
6.5.9 Enhanced DataChange Subscription Server Facet	Monitored Item Services	Monitor MinQueueSize_05
6.5.25 Core Server Facet	Attribute Services	Attribute Write Index
6.5.26 Data Access Server Facet	Data Access	Data Access Analog Data Access Multistate Data Access PercentDeadBand Data Access Semantic Changes Data Access Two State

Profile	"Group"	不支持 "Conformance Unit"
6.5.35 Standard UA Server	Attribute Services	Attribute Write StatusCode & TimeStamp
6.5.47 Standard UA Server Profile	Attribute Services	Attribute Write StatusCode & Timestamp

6.10.5 WinCC OPC UA 服务器的名称区域

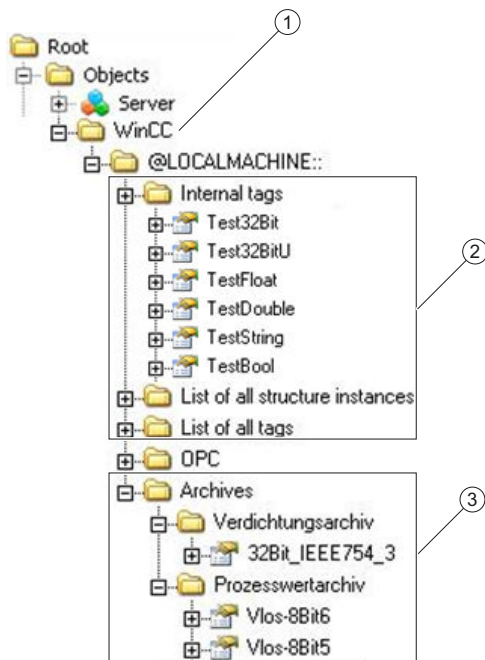
简介

WinCC OPC UA 服务器为 OPC UA 客户端提供了一个层级名称区域以及以下运行系统数据的访问权限：

- 过程值（WinCC 变量和 WinCC 变量组）
- 数据日志包含的记录变量
- WinCC 消息

WinCC OPC UA 服务器的名称区域附加在“Objects”默认文件夹中。

以下画面显示了本地 PC（“@LOCALMACHINE::”）上激活的 WinCC 项目的 WinCC OPC UA 服务器的名称区域：



- ① WinCC 特定名称区域的起始节点。
- ② 显示 WinCC 变量；结构对应于 WinCC 中变量的结构。
- ③ 显示数据日志

WinCC 变量的显示

变量组、通信驱动程序和连接由“FolderType”类型的 OPC UA 对象显示。这些文件夹中的每一个都具有“Organizes”类型的对从属对象和变量的引用。

内部和外部 WinCC 变量由“DataItem Type”类型的 OPC UA 变量显示。如果额外记录了 WinCC 变量，则显示的 OPC UA 变量额外具有“HasHistoricalConfiguration”类型的引用来进行日志组态。“Historizing”和“AccessLevel”属性分别设置。

下表显示了表示 WinCC 变量的 OPC UA 变量的最重要属性。在“OPC UA Part 3 - Address Space Model 1.02 Specification”文档的“5.6”下可以找到完整的属性列表：

属性	说明	注释
NodeId	WinCC 变量的唯一标识	-
BrowseName	WinCC 变量名称	-
DisplayName	WinCC 变量名称	-
Value	变量值和状态	-

属性	说明	注释
DataType	对应于 WinCC 变量类型的 OPC UA 数据类型，例如： <ul style="list-style-type: none"> • Int32；有符号的 32 位值 • UInt32；无符号的 32 位值 	-
AccessLevel	"CurrentRead" / "CurrentWrite" "HistoryRead" / "HistoryWrite"	对应于 WinCC 变量组态
ValueRank	始终为“Scalar”	-

记录变量的显示

过程值和压缩日志由“FolderType”类型的 OPC UA 对象显示。这些文件夹中的每一个都具有“Organizes”类型的对相关记录变量的引用。

过程值或压缩日志的记录变量由“BaseDateVariableType”类型的 OPC UA 变量显示。记录变量始终具有对日志组态的“HasHistoricalConfiguration”类型的引用。

下表显示了表示 WinCC 记录变量的 OPC UA 变量的最重要属性。在“5.6”下的“OPC UA Part 3 - Address Space Model 1.01 Specification”文档中可以找到完整的属性列表：

属性	说明	注释
NodId	记录变量的唯一标识	-
BrowseName	归档变量的名称	-
DisplayName	归档变量的名称	-
Description	节点说明	-
Value	不可用	对于记录变量，无法读取和更改此属性。
DataType	对应于 WinCC 变量类型的 OPC UA 数据类型，例如： <ul style="list-style-type: none"> • Double；64 位浮点数 • UInt32；无符号的 32 位值 	-
AccessLevel	"HistoryRead" / "HistoryWrite"	-
ValueRank	始终为“Scalar”	-

访问 WinCC 消息

WinCC 名称区域的起始节点是一个 Event Notifier，它允许 OPC UA 客户端通过 Subscriptions 接收运行系统中 WinCC 消息的状态更改。

6.10.6 OPC UA 数据访问

内部和外部 WinCC 变量由“DataItemType”类型的 OPC UA 变量显示。其它 DataAccess 变量类型（如“AnalogItem”或“DiscreteType”）不受支持。

WinCC OPC A&E 服务器支持对 OPC UA 变量属性（如“DataType”或“AccessLevel”）进行读取访问。“Value”属性仅支持写访问和订阅。

6.10.7 OPC UA 日志访问

简介

“OPC Historical Access”允许访问归档，并包括“Historical Data”和“Historical Events”服务。WinCC OPC UA 服务器仅支持“Historical Data”服务。

WinCC OPC UA 服务器通过“Services”为 OPC 客户端提供对变量归档原始数据的访问。

- HistoryRead (READRAW)
- HistoryUpdate (INSERTDATA, REPLACEDATA, UPDATEDATA, DELETE_RAW)

使用 OPC UA 客户端可以读取变量归档中的归档变量值，也可以有限制地将归档变量值写入变量归档。根据变量归档的组态，归档变量可以包含原始数据或已处理的过程值。

归档变量的特性

WinCC 中的一个过程变量可以位于多个变量归档中。在这种情况下，过程变量链接到一个相应的归档变量。

归档组态的属性/Properties

下表显示了“HistoricalConfigurationType”类型的 OPC UA 变量组态的 Properties：在“Description”属性中，将显示 WinCC 中组态的归档变量注释。在“OPC UA Part 11 - Historical Access 1.02 Specification”文档的“5.2.2”下可以找到完整的属性列表：

Property	说明/值	注释
Definition	WinCC 过程变量名称	对于过程值归档
Stepped	True	-

不支持以下可选 Properties:

- MaxTimeInterval
- MinTimeInterval
- ExceptionDeviation
- ExceptionDeviationFormat

Service“HistoryUpdate”的限制

只能对过程值归档使用 Service“HistoryUpdate”。

下表列出了 WinCC OPC UA 服务器支持的函数：支持哪些函数取决于 WinCC OPC UA 服务器的组态以及过程值归档组态。有关更多信息，请参见“§5.5”下的“OPC UA Part 11 - Historical Access 1.00 Specification”文档。

Service	函数	说明
HistoryUpdate	INSERTDATA	插入新归档值
	REPLACEDATA	替换现有归档值
	UPDATEDATA	替换插入归档值
	DELETE_RAW	删除归档值

6.10.8 OPC UA 报警和条件

简介

OPC UA 服务器提供了对 WinCC 消息系统（WinCC 7.3 及更高版本）消息的访问。

OPC UA 服务器使用 WinCC-Event-Notifications 通过 Subscriptions 和 Monitored Event Items 将 WinCC 消息状态更改转发给 OPC UA 客户端，但不会在名称区域保持一个 Condition 实例。要使用的 Event Notifier 节点是 WinCC 名称区域的起始节点。UA 客户端可以过滤消息，并定义返回的消息属性列表。

OPC UA 服务器支持“OPC UA Alarms & Conditions 1.02”规范。

以下部分概述了 WinCC 消息系统到 OPC UA 的映射。有关详细信息，请参见“Part 9: Alarms and Conditions 1.02 Specification”中的规范。

WinCC 消息系统到 UA 事件类型的映射

WinCC 消息映射到以下 OPC UA 事件类型：

WinCCEventType

该类型基于“BaseEventType”，并使用以下确认原则映射“简单的”WinCC 消息：

- “无离开状态的消息”激活
- “确认进入”未激活

此类型的消息的示例有启动和停止电机。

WinCCAlarmConditionType

此类型基于“AlarmConditionType”，映射所有无法映射到 WinCCEventType 的消息，例如可确认消息以及状态为“已到达”和“已离开”的消息。

“WinCCAlarmConditionType”类型的消息事件链接到条件。例如，只要超出变量限值，WinCC 就会生成一个消息。OPC UA 中的此消息等同于一个 Alarm Condition。

WinCC 消息属性

两个 Event 类型为基本类型增加了 WinCC 特定的消息属性。该属性作为 UA Event Properties 进行 1:1 映射，更多详细信息请参见“WinCC 消息系统的属性”。

消息类别和消息类型

WinCC 消息系统将过程中的干扰和操作情况通知给用户。WinCC 消息始终属于相应的 UA Events 的“CLASSID”、“TYPEID”、“CLASSNAME”和“TYPENAME”属性中指定的特定消息类别和消息类型。

优先级

在 WinCC 消息系统中组态消息时，可以组态“0”和“16”之间的优先级。OPC UA 规范为 Severity 定义的取值范围是“1”到“1000”。“1”代表最低严重性，“1000”代表最高 Severity。

因此，优先级的值必须相应映射到 OPC 严重性。在标准映射中，优先级“0”指定给 OPC-Severity “1”，而优先级“16”指定给 OPC-Severity “1000”。所有其它值在“0”与“1000”之间进行线性插值。

OPC UA 映射规则

组态 WinCC 消息系统期间，将进行设置以确定哪些过程事件产生消息。此消息通常在 OPC UA 中显示为一个 Event。

下表显示了 Events 的最重要 Properties，以及 WinCC 消息系统提供信息的方式。

OPC UA 属性	映射 WinCC 消息系统
对于所有事件类型:	
EventID	唯一的消息标识
EventType	事件类型: WinCCAlarmConditionType 节点或 WinCCEventType 节点的节点 ID
SourceNode	不相关
SourceName	指示消息源。下面详细说明了映射。
Message	相应消息编号的消息文本。
Time	事件的时间。时间戳以 UTC 格式给出
Severity	WinCC 消息的优先级
仅限于 WinCCAlarmConditionType:	
ConditionName	设置输出文本以及消息。文本输出取决于映射规则集: <ul style="list-style-type: none"> “模式 1”和“模式 2”: 消息编号 “模式 3”: 消息类别, 例如“过程控制消息”
Quality	返回消息质量
ConditionClassId	“ProcessConditionClassType”节点的节点 ID
ConditionClassName	"ProcessConditionClassType"
Retain	“TRUE”, 带未决消息
NodeId	ConditionId: UA-Condition 的唯一标识, 例如报警。需要确认, 即使不支持 Condition 实例
EnabledState	"TRUE" 表示消息已启用
ActiveState/Id	"TRUE" 表示消息已到达
AckedState/Id	"TRUE" 表示消息已确认
ClientUserId	指示用户已登录

说明

OPC UA 服务器不支持以下 OPC UA Condition 和 Alarm Properties :

- BranchId
- LastSeverity
- InputNode
- ConfirmedState
- SuppressedState
- ShelvingState
- SuppressedOrShelved
- MaxTimeShelved

消息状态/确认状态

下表显示了 WinCC 消息状态到相应 WinCCAlarmConditionType - Properties 的映射:

消息状态	EnabledState/Id	ActiveState/Id	AckedState/Id
锁定的消息	FALSE	-	-
启用的消息	TRUE		
已接收的消息	TRUE	TRUE	FALSE
带确认的已发送消息	TRUE	FALSE	TRUE
不带确认的已发送消息	TRUE	FALSE	FALSE
已确认的消息 (消息未决)	TRUE	TRUE	TRUE
已确认的消息(消息不再未决)	TRUE	FALSE	TRUE
已接收的、确认的消息	TRUE	TRUE	TRUE
已接收的、带确认的已发送消息	TRUE	FALSE	TRUE
已接收的、不带确认的已发送消息	TRUE	FALSE	FALSE
由系统确认的消息 (消息未决)	TRUE	TRUE	TRUE
由系统确认的消息 (消息不再未决)	TRUE	FALSE	TRUE

消息状态	EnabledState/Id	ActiveState/Id	AckedState/Id
紧急确认的消息（消息未决）	TRUE	TRUE	TRUE
紧急确认的消息（消息不再未决）	TRUE	FALSE	TRUE

WinCC 消息系统映射设置

就消息的 Properties“SourceName”和“Message”的映射而言，OPC UA 服务器的组态也适用于 OPC UA 服务器。

- 对于带层级访问的 OPC A&E 服务器：

SourceName	指示消息源。Source 的格式为“<服务器前缀>::Area\UserTextBlock 2”。本地计算机的服务器前缀为“@LOCALMACHINE”。
Message	返回相应消息编号的消息文本

- 对于不带层级访问的 OPC A&E 服务器：

SourceName	指示消息源。Source 的格式为“<服务器前缀>::localhost:”。本地计算机的服务器前缀为“@LOCALMACHINE”。
Message	返回相应消息编号的消息文本

报警组

在 WinCC 7.3 中，WinCC 报警组不显示在名称区域中。

支持的事件方法

确认

使用标准 OPC UA 信息模型中“AcknowledgeableConditionType”节点的“Acknowledge”方法确认 WinCC 消息。

只能确认“WinCCAlarmConditionType”类型的消息。

ConditionRefresh

使用标准 OPC UA 信息模型中“ConditionType”节点的“ConditionRefresh”方法建立仍未决的消息。

过滤器

OPC UA 客户端可以为 Monitored Event Items 定义过滤器。

但是，OPC UA 服务器不支持以下操作员：

- FilterOperator_Cast
- FilterOperator_BitwiseAnd
- FilterOperator_BitwiseOr
- FilterOperator_RelatedTo
- FilterOperator_InView

参见

WinCC 消息系统的属性 (页 220)

6.10.9 WinCC 消息系统的属性

概述

下表列出了 WinCC 消息系统的可组态属性。将属性 1:1 映射为 UA Event Properties 。

WinCC 消息属性	含义	数据类型
CLASSNAME	消息类别的名称	String
TYPENAME	消息类型的名称	String
FORECOLOR	进入消息、离开消息和已确认消息的前景色。	Int32
BACKCOLOR	进入消息、离开消息和已确认消息的背景色。	Int32
FLASHCOLOR	闪烁颜色	Int32
FLAGS	指示必须确认该消息	Int32
TEXT01...TEXT10	用户文本块 #1...#10 的内容	String
PROCESSVALUE01... PROCESSVALUE10	过程值块 #1...#10 的内容	
STATETEXT	状态消息	String
INFOTEXT	消息的信息文本	String
LOOPINALARM	指示是否已组态报警回路	Int32

WinCC 消息属性	含义	数据类型
CLASSID	消息类别 ID	Int32
TYPEID	消息类型 ID	Int32
MODIFYSTATE	消息状态变量的值	Int32
AGNR	输出生成消息的自动化系统的编号	Int16
CPUNR	输出生成消息的 CPU 的编号	Int16
DURATION	输出进入状态和离开状态与消息确认之间的时间段	Int32
COUNTER	自运行系统启动以来的消息数	Int32
QUITSTATETEXT	指示消息是否已确认	String
QUITCOUNT	处于打开状态的未确认消息数	Int32
PARAMETER	消息的组态参数	Int32
BLOCKINFO	消息块的当前内容	String
ALARMCOUNT	未决消息数	Int32
LOCKCOUNT	已锁定的消息数	Int32
PRIORITY	消息的优先级	Int32
APPLICATION	输出触发了消息的应用程序	String
COMPUTER	输出处理了消息的计算机的名称	String
USER	输出处理了消息的用户的名称	String
COMMENT	消息注释	String
HIDDEN-COUNT	隐藏的消息数	Int32
OS-HIDDEN	表示消息已隐藏	布尔型
OS_EVENTID	WinCC 消息数	Int32
BIG_COUNTER	消息计数器	Int64

参见

OPC UA 报警和条件 (页 215)

6.10.10 组态 WinCC OPC UA 服务器

6.10.10.1 组态文件

简介

WinCC OPC UA 服务器使用组态文件“OPCUAServerWinCC.xml”进行组态。本章介绍组态文件的布局。“如何组态 OPC UA 服务器 (页 223)”一章介绍了如何组态 WinCC OPC UA 服务器。

文件位置

项目特定的组态文件“OPCUAServerWinCC.xml”存储在 WinCC 项目文件夹下，路径为：
“<WinCC 项目文件夹>\OPC\UAServer”

布局

组态文件分为多个部分。

<Secured Application> 部分

在该部分中，根据 OPC UA 技术规范/第 6 部分/§“安全设置管理”设置 OPC UA 应用程序安全性。

<Secured Application>	
<pre><BaseAddresses> <...></...> </BaseAddresses></pre>	<p>WinCC OPC UA 服务器 URL 的组态。</p> <p>有关 URL 的更多信息，请参见“OPC UA 的安全概念 (页 202)”。</p>
<pre><SecurityProfileUris> <SecurityProfile> <...></...> </SecurityProfile> ... </SecurityProfileUris></pre>	<p>支持的安全策略的组态</p> <p>“none”设置仅用于测试和诊断目的</p> <p>有关安全策略的详细信息，请参见“OPC UA 的安全概念 (页 202)”。</p>
<pre><ApplicationCertificate> <TrustedCertificateStore> <TrustedCertificates> <...></pre>	<p>符合 OPC UA 技术规范/第 6 部分的默认证书组态的版本。</p> <p>(可选)</p>
</Secured Application>	

<Server Configuration> 部分

在该部分中可以设置更多服务器特定的参数。

<Server Configuration>	
<pre><SecurityPolicies> <SecurityPolicy> <...></...> </SecurityPolicy> ... </SecurityPolicies></pre>	<p>消息安全模式的组态。</p> <p>“none” 设置仅用于测试和诊断目的</p> <p>有关消息安全模式的详细信息，请参见“OPC UA 的安全概念 (页 202)”。</p>
<pre><UserTokenPolicies> <UserTokenPolicy> <...></...> </UserTokenPolicy> ... </UserTokenPolicies></pre>	<p>用户标识组态</p> <p>“Anonymous” 设置仅用于测试和诊断目的</p> <p>有关消息安全模式的详细信息，请参见“OPC UA 的安全概念 (页 202)”。</p>
<pre><FastInsert> <Users> <...></...> </Users> <Clients> <...></...> <Clients> </FastInsert></pre>	<p>优化的 WinCC 归档写访问的组态</p>
<Server Configuration>	

6.10.10.2 如何组态 OPC UA 服务器**要求**

已创建 WinCC 项目。

打开组态文件

1. 打开 Windows 资源管理器。导航到目录“<WinCC 项目文件夹>OPC\UAServer”。
2. 打开“OPCUAServerWinCC.xml” 组态文件。有关详细信息，请参见“组态文件 (页 222)”。

更改 WinCC OPC UA 服务器的端口号

1. 如果需要，更改 <BaseAddresses> 下的端口号 4862。
不要使用已经分配给其他应用程序的端口号。
参数 [HostName] 为计算机名称的占位符，在运行期间确定。

示例：

```
<BaseAddresses>
<ua:String>opc.tcp://[HostName]:5210</ua:String>
</BaseAddresses>
```

指定安全设置

1. 指定通信的安全设置。有关详细信息，请参见“OPC UA 的安全概念 (页 202)”
2. 在 <SecurityProfileUris> 下，组态支持的“安全策略”。
 - 使用“true”启用设置。
 - 使用“false”禁用设置。

示例：

```
<SecurityProfile>
  <ProfileUri>http://opcfoundation.org/UA/
  SecurityPolicy#None</ProfileUri>
  <Enabled>>false</Enabled>
</SecurityProfile>
```

3. 在 <SecurityPolicies> 下，组态关联的“消息安全模式”。
要禁用某一设置，请删除整个条目 <SecurityPolicy>... </Security Policy>。

示例：

```
<SecurityPolicy>
  <ProfileUri>http://opcfoundation.org/UA/SecurityPolicy#None</
  ProfileUri>
  <MessageSecurityModes>None</MessageSecurityModes>
</SecurityPolicy>
```

说明

客户端与服务器之间的不安全通信

“none”设置仅用于测试和诊断目的。

要在生产模式下实现安全的客户端/服务器通信，至少需要使用以下设置：

- SecurityPolicy: Basic128Rsa15
Message Security Mode: Sign
-

指定用户标识

1. 指定用于在 <UserTokenPolicies> 下设置连接的用户标识。有关详细信息，请参见“OPC UA 的安全概念 (页 202)”。

要禁用某一设置，请删除整个条目。

示例

```
<UserTokenPolicy>  
<TokenType>Anonymous</TokenType>  
</UserTokenPolicy>
```

组态优化的 WinCC 归档写访问

1. 如果需要，在 <FastInsert> 下组态优化的 WinCC 归档写访问。
 - 设为 “true” 可以激活所有 OPC UA 客户端对 WinCC 归档的优化写访问。
 - 设为 “false” 可以设置特定 Windows 用户或 OPC UA 客户端对 WinCC 归档的优化写访问。

在 <Users> 下指定 Windows 用户。

在 <Clients> 下指定 OPC UA 客户端。将客户端证书中输入的“Common Name”用作 ClientName。

示例：

```
<EnabledByDefault>>false</EnabledByDefault>  
<Users>  
  <User>domain\user1</User>  
</Users>  
<Clients>  
  <Client>ClientName1</Client>  
</Clients>
```

6.11 跟踪

引言

“跟踪”功能可用于记录变量值和函数调用，以进行测试和错误分析。

条目存储在跟踪文件中。跟踪显示逐步建立连接的过程，从而更易于识别连接故障的来源。

设置

跟踪输出必须在操作系统的注册表中设置。有关更多信息，请参阅“SIMATIC 客户支持”。

OPC UA WinCC Channel

7.1 WinCC 通道“OPC UA WinCC 通道”

简介

WinCC 可以用作 OPC UA 服务器和 OPC UA 客户端。通道“OPC UA WinCC Channel”是 WinCC 的 OPC UA 客户端应用程序。

借助 OPC UA（统一架构），WinCC 支持独立于平台的 OPC 跟进技术。有关详细信息，请参见 OPC UA 规范及 WinCC Information System 下的“接口 > OPC - 开放式互连 > WinCC OPC UA 服务器”(Interfaces > OPC - Open Connectivity > WinCC OPC UA Server)。需要了解 OPC UA 的基本知识才能进行组态。

可在 WinCC Information System 下的“通信 > OPC 通道”(Communication > OPC channel) 中找到有关 OPC-DA 客户端及 OPC-XML 客户端的文档。

WinCC OPC UA 客户端的功能

WinCC OPC UA 客户端支持对任何符合“OPC 统一架构”规范的 OPC UA 服务器的数据访问。

OPC Unified Architecture (OPC UA) 提供了验证和加密等其它机制以保证相关伙伴之间的通信安全。

下列 OPC UA 组件会自动安装：

- OPC UA 通信驱动程序
- WinCC OPC UA Configurator

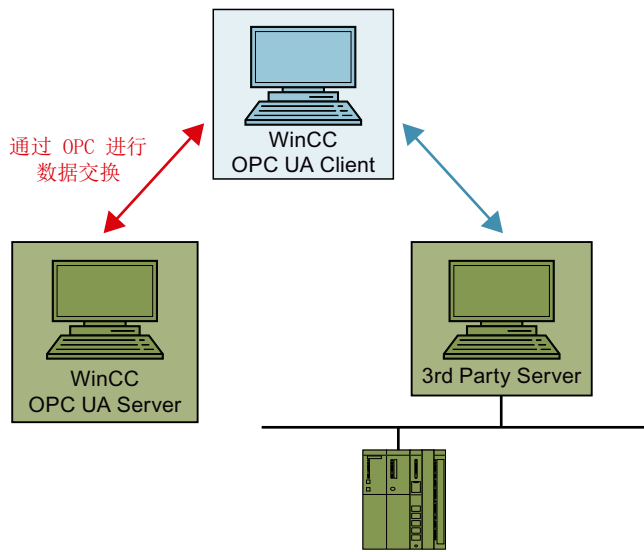
通过 OPC UA 通信

要将 WinCC 用作 WinCC OPC UA 客户端，请将 OPC UA 通信驱动程序“OPC UA WinCC Channel”插入到 WinCC 项目中。无需单独的通信模块。

一个 WinCC OPC UA 客户端可以访问多个 OPC UA 服务器。可以为此组态与每个 OPC UA 服务器的连接。可将 WinCC OPC UA 客户端用作中央操作员监控站。

WinCC 提供了 WinCC OPC UA 组态器以供组态使用。

7.1 WinCC 通道“OPC UA WinCC 通道”



7.2 所支持的数据类型概述

简介

为了进行数据通信，在 WinCC OPC UA 客户端的 WinCC 项目中组态访问 OPC UA 服务器的变量。要实现这一目的，在变量管理中导入 OPC UA 节点作为 WinCC 变量。

WinCC OPC UA 客户端和 WinCC OPC UA 服务器支持以下 WinCC 数据类型。

所支持的数据类型

- 二进制变量
- 有符号 8 位数
- 无符号 8 位数
- 有符号 16 位数
- 无符号 16 位数
- 有符号 32 位数
- 无符号 32 位数
- 浮点数 32 位 IEEE 754
- 64 位 IEEE 754 浮点数
- 文本变量，8 位字符集
- 文本变量，16 位字符集
- 原始数据类型
- 结构类型 ¹⁾
- 日期/时间

1) 然而，对于结构类型，只支持结构元素，而不支持结构本身。

7.3 WinCC OPC UA Configurator

简介

要通过 WinCC 访问 OPC UA 服务器的变量，需要在 WinCC 项目中组态与 OPC UA 服务器的连接及相应的 WinCC 变量。

WinCC OPC UA Configurator 可对此提供支持。WinCC OPC UA Configurator 是 WinCC 安装的一部分。

WinCC OPC UA Configurator 任务

WinCC OPC UA Configurator 执行以下任务以组态 WinCC 的 OPC UA 元素：

- 选择 OPC UA 服务器
- 创建连接
- 创建变量组
- 变量选择
- 在 WinCC 中添加变量

打开 WinCC OPC UA Configurator

在 WinCC Configuration Studio 中，可从变量管理中打开 WinCC OPC UA Configurator。

要求

WinCC OPC UA Configurator 需要 Java 运行系统环境。

要使用 WinCC OPC UA Configurator，需重下载和安装最新的 JRE 安装包。您可在 Oracle 主页下找到该安装包：

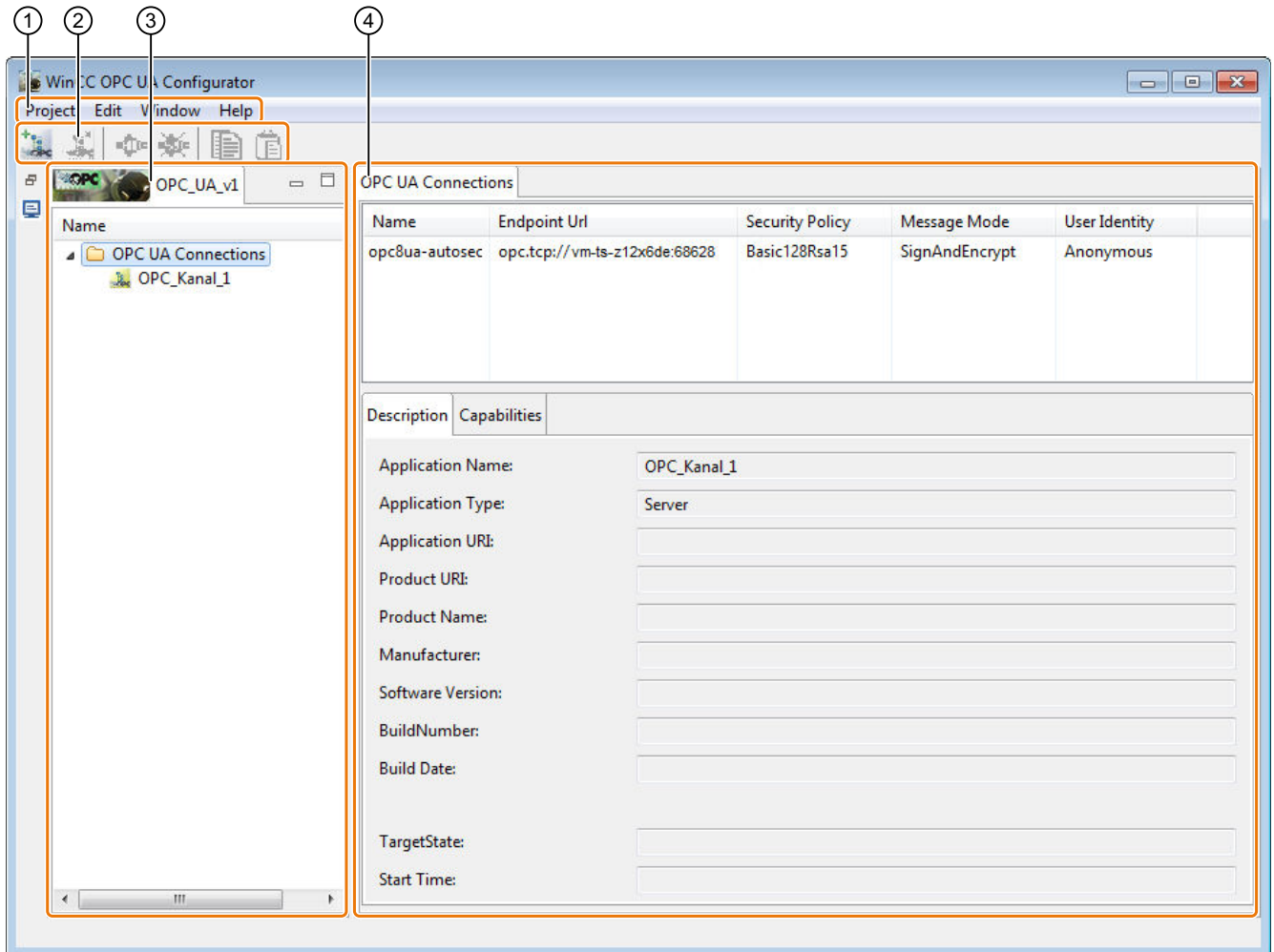
- 链接到 Oracle 下载主页

在 32 位版本中安装适用于操作系统类型的最新版本，最低版本为 JRE 8。

步骤

1. 在“变量管理”(Tag management) 导航区域中选择“OPC UA Connections”通道单元。
2. 从快捷菜单中选择“系统参数”(System Parameters) 命令。

WinCC OPC UA Configurator



- ① 菜单栏
- ② 工具栏
- ③ 项目窗口
- ④ “OPC-UA-Verbindungen” 数据区

① 菜单栏

该菜单包含 WinCC OPC UA Configurator 的大多数功能。
可以通过窗口符号或使用对象的快捷菜单来操作附加功能。

② 工具栏

可以在“WinCC OPC UA Configurator 中的符号 (页 238)”下找到这些符号的概述：

③ 项目窗口

将显示组态的 OPC UA 连接。

窗口的标题域中包含打开的 WinCC 项目名称。

显示连接属性

在树中单击文件夹“OPC UA 连接”时，含有连接及其属性的列表会显示在数据区“OPC UA 连接”中。

为每个 OPC UA 服务器都创建一个独立的连接。

创建连接时，使用 WinCC OPC UA Configurator 进行所需的设置。

④“OPC UA 连接”数据区

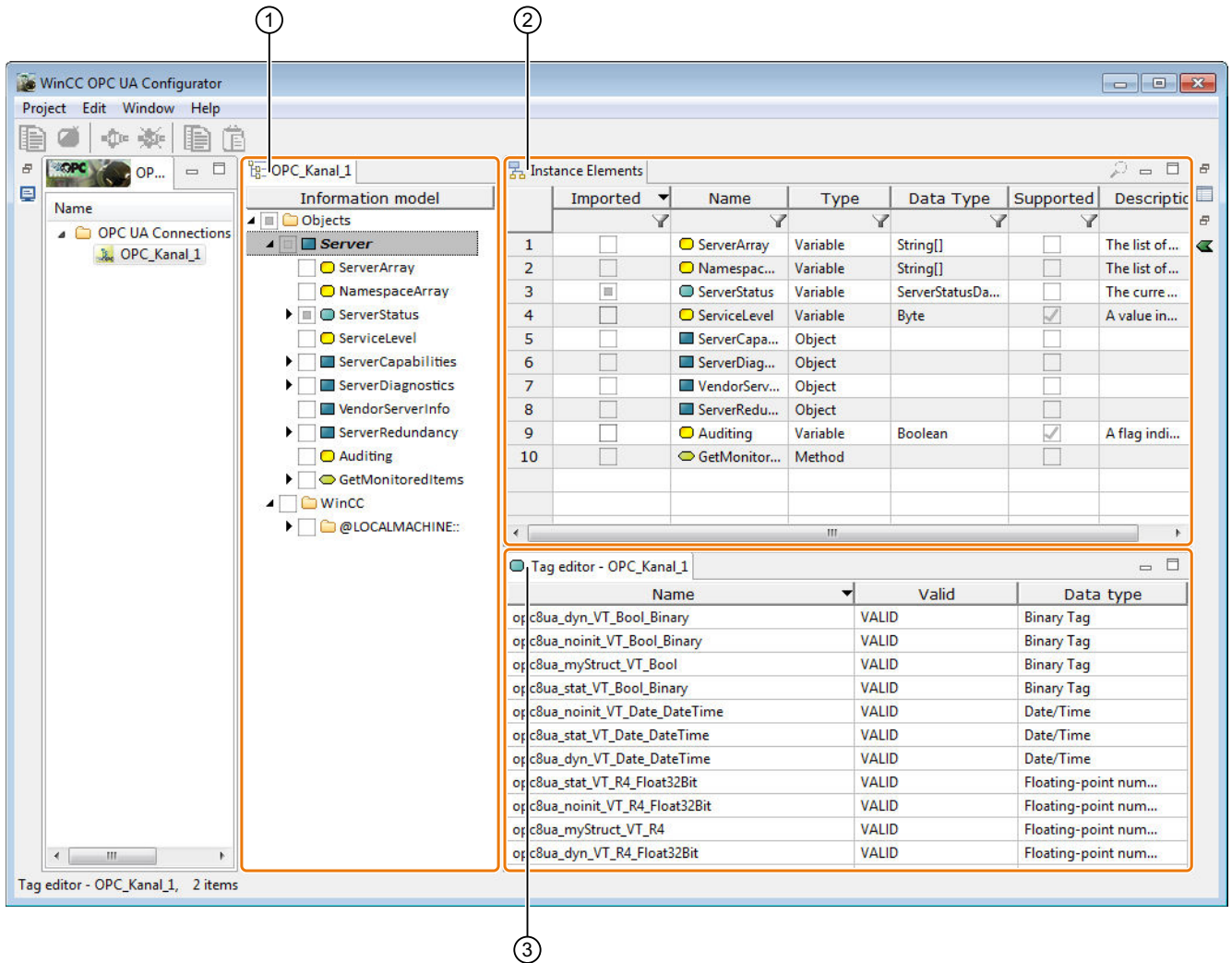
将显示组态的与 OPC UA 的连接。

单击 OPC UA 服务器时，服务器的属性会显示在“属性”(Properties) 和/或“功能”(Capabilities) 下。

连接状态由符号在导航窗格中进行指示，字体颜色则显示在数据区中：

- 绿色/黑色字体：已建立与 OPC UA 服务器的连接。
- 黄色/蓝色字体：未建立与 OPC UA 服务器的连接。
可能的原因包括：
 - 设置中含有错误的连接参数。
 - 客户端证书被 OPC UA 服务器拒绝。
 - 缺少组态步骤。
- 红色/红色字体：与 OPC UA 服务器的连接中断。
- 可能的原因包括：
 - 存在硬件故障。
 - 存在连接故障。

WinCC OPC UA Configurator 信息模型



- ① 导航区域“信息模型”(Information model)
- ② “实例元素”(Instance elements) 数据区
- ③ “Tag Editor” 数据区

① “信息模型”(Information model) 导航区域

名称空间和包含的 OPC UA 节点显示在层级结构中。

要显示 OPC UA 服务器的信息模型，在项目窗口选择一个连接的服务器（状态：绿色）。

OPC UA 节点前的符号 显示，此级别下是否包含已作为 WinCC 变量被导入到变量管理中的节点。

- 所有变量都已导入时显示复选标记。
- 如果只导入部分变量，则符号将填充到一半。
- 如果未导入变量，则符号为空。

更新信息模型

OPC UA 服务器的信息模型的内容与 OPC UA 服务器自动同步。

如果信息模型不同步，则在 OPC UA 节点快捷菜单中的树视图内选择“更新”(Update) 条目。例如，在下列情况下可能需要进行该操作：缺失服务器支持时或在连接失败后。


② “实例元素”(Instance elements) 数据区


显示未包含在“信息模型”(Information model) 导航区域中选定级别内的 OPC UA 节点。子元素（即子文件夹节点）不会显示在更高级别中。

导入 OPC UA 节点

使用 OPC UA 节点的快捷菜单，以 WinCC 变量的形式导入节点。导入的 WinCC 变量显示在“Tag Editor”数据区中。

搜索 OPC UA 节点

要在选定级别下显示所有节点，单击窗口工具栏中的  符号。要想只显示所选层的节点，再次单击符号。

要过滤节点，单击表格列标题中的  符号。可以使用此功能删除不支持的节点或搜索字符串。

同步显示

要导航到信息模型中的 OPC UA 节点，可在快捷菜单中选择条目“转到服务器节点”(Go to server node)。

③ “Tag Editor” 数据区

同时列出 OPC UA 连接的 WinCC 变量及其属性。

WinCC Configuration Studio 的变量管理显示相同的信息。但是，仅更改 WinCC OPC UA Configurator 中的变量属性。

列“有效”(Valid) 指示 WinCC 变量是否有效，并与 OPC UA 节点相连。

更改变量名称

要更改 WinCC 变量的名称，单击表格中的“功能”(Name) 域。

可以在“属性 > 变量名称”(Properties > Tag Names) 对话框中为创建的 WinCC 变量名称定义设置。

删除变量

删除数据区中的变量时，也会删除 WinCC 项目中的变量。

注意
删除变量 无法撤消变量删除操作。

在信息模型中选择 OPC UA 节点

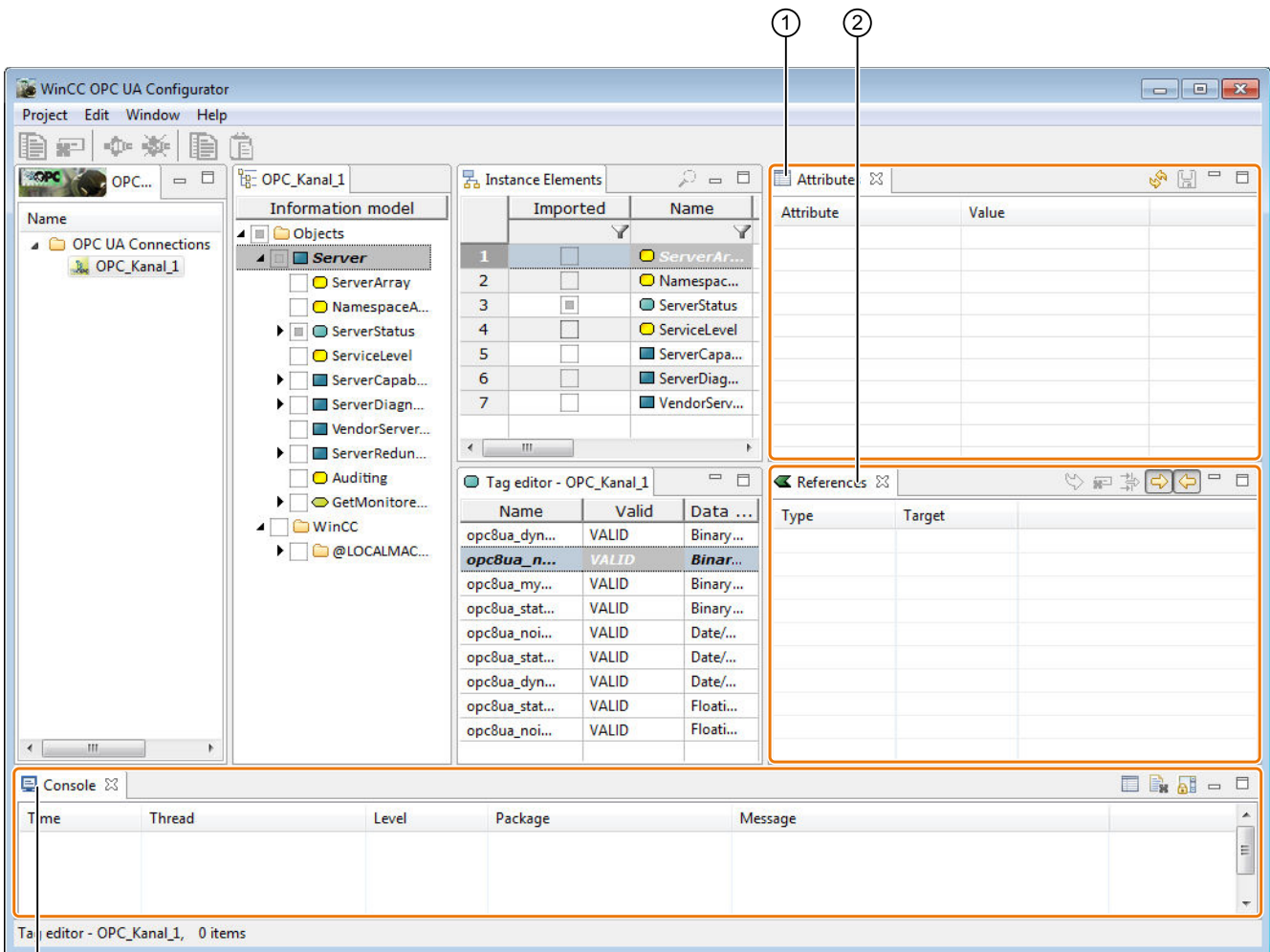
要在信息模型中导航到连接的 OPC UA 节点，可在变量快捷菜单中选择条目“转到服务器节点”(Go to server node)。

也可以双击 WinCC 变量。

WinCC OPC UA Configurator - 其他窗口

可以使用对话框“窗口 > 视图”(Windows > View) 显示其他窗口。

或者，也可以通过 WinCC OPC UA Configurator 中的相应符号激活窗口。




- ① “属性”(Attributes) 窗口
- ② “参考”(References) 窗口
- ③ “控制台”(Console) 输出窗口

① “属性”(Attributes) 窗口

有关所选 OPC UA 节点的详细信息显示在信息模型中。

② “参考”(References) 窗口

显示所选节点与其他 OPC UA 节点的所有关系。因此，该窗口会对“信息模型”(Information model) 窗口的层级显示加以补充。
使用相关的箭头符号过滤此显示。

使用  符号或双击，跳转到信息模型中的参考节点。

③ “控制台”(Console) 输出窗口

显示 WinCC OPC UA Configurator 的状态消息和错误消息，例如有关连接状态的信息。














更多相关信息，请参见““控制台”输出窗口 (页 242)”。

7.4 WinCC OPC UA Configurator 中的符号


可以使用菜单栏或符号操作 WinCC OPC UA Configurator。

下表概述了使用的图标及其含义：



WinCC OPC UA 组态器的符号

符号	含义
	创建与 OPC UA 服务器的新连接。 只有在项目窗口中选择了“OPC UA 连接”(OPC UA Connections) 时才激活。
	删除所选连接。
	建立与 OPC UA 服务器的连接。
	从 OPC UA 服务器关闭连接。
	创建新的变量组。 只有在项目窗口中的“OPC UA 连接”(OPC UA Connections) 下选择了连接时才激活。
	删除变量组。
	插入复制的元素。 只有在“实例元素”(Instance elements) 数据区中选择了支持的 OPC UA 节点时才激活。
	将复制的 OPC UA 节点作为 WinCC 变量导入到 Tag Editor 中。 只有在 OPC UA 节点已复制到剪贴板中且激活了 Tag Editor 时才激活。
	删除所选变量。 只有在选择了变量编辑器时激活。
	关闭窗口。 要重新显示关闭的窗口，在菜单栏中选择“窗口 > 视图”(Window > View)。
	最小化窗口。 要查看已最小化窗口的内容，单击相应的窗口符号。
	最大化窗口。
	还原窗口。



数据区符号

符号	含义
	OPC UA 服务器已连接。 在“OPC UA 连接”(OPC UA Connections) 数据区中，以黑色字母显示连接。
	OPC UA 服务器未连接。 在“OPC UA 连接”(OPC UA Connections) 数据区中，以蓝色字母显示连接。
	尝试连接 OPC UA 服务器时出错。 在“OPC UA 连接”(OPC UA Connections) 数据区中，以红色字母显示连接。
	名称空间中的 OPC UA 节点。 示例：文件夹，对象，变量
<input checked="" type="checkbox"/>	在信息模型中的级别下，存在已为其导入 WinCC 变量的 OPC UA 节点。
<input type="checkbox"/>	在级别下，没有带相关 WinCC 变量的 OPC UA 节点。
<input checked="" type="checkbox"/>	已为 OPC UA 节点创建 WinCC 变量。
	显示所选级别下的所有节点。 要只显示所选层的节点，再次单击符号。
	根据列特定标准过滤显示。 单击时，显示选择列表或激活空白文本域。
	指示已设置过滤器。 单击时删除过滤器。




“属性”(Attributes) 符号的符号

符号	含义
	更新属性。
	- (无功能)




“引用”(References) 窗口的符号

符号	含义
	切换到信息模型中的引用节点。 也可以双击该条目。
	- (无功能)

7.4 WinCC OPC UA Configurator 中的符号

符号	含义
	在打开的对话框中，选择要过滤的引用类型。 要显示所有引用，请选择顶端的条目。
	显示更多引用。
	显示之前的引用。


窗口“控制台”符号







符号	含义
	组态控制台设置。
	删除控制台上显示的消息。 此功能不会影响日志文件。
	暂停连续报警视图或重新启动。

窗口符号

如果窗口已隐藏，则单击 WinCC OPC UA Configurator 的右侧或左侧边框上的相应符号。

单击符号时，将显示窗口内容。

单击  符号还原窗口。

符号	含义
	项目窗口
	“控制台”输出窗口
	“属性”窗口
	“引用”窗口
	“实例元素”数据区
	“Tag Editor”数据区

7.5 WinCC OPC UA Configurator 的设置

“Settings” 对话框

在“设置”(Settings)对话框中组态“OPC UA WinCC Channel”的常规设置。

通过“窗口 > 设置”(Window > Settings) 菜单打开对话框。

可进行以下设置：

- WinCC 变量名称规范
可在“OPC UA 变量 (页 248)”下找到更多相关信息。
- 信息模型中的名称空间表示方式
- OPC UA 通信设置参数
- 日志文件的记录范围及存储路径
可在““控制台”输出窗口 (页 242)”下找到更多相关信息。

信息模型中的名称空间表示方式

在“服务器”(Server)对话框中，指定名称空间在导航区域“Informationsmodell”中的显示方式。

可以为每个名称空间进行以下设置：

- 字体
- 字号
- 字体颜色
- 背景色

通信设置

处理 OPC UA 服务的超时时间在“通信”(Communication)对话框中指定。

如果需要，可以针对连接 OPC UA 服务器的响应调整该值。

控制台设置

在“记录器”(Logger)对话框中进行控制台设置。


7.6 “控制台” 输出窗口

“控制台” 输出窗口包含 WinCC OPC UA Configurator 和/或 OPC UA 服务器的状态消息及错误消息。

常见消息包括（例如）：

- Bad_CertificateUntrusted (0x801A0000) "The Certificate is not trusted."
- The Certificate is not trusted.
- Server '服务器名称' not active
- All security tokens have expired
- The secure channel has been closed.
- 连接状态

确定设置

通过菜单“窗口 > 设置 > 记录器”(Window > Settings > Logger) 或通过输出窗口中的控制台图标，指定控制台设置：

日志文件

消息储存在日志文件中。无论输出窗口是否激活，日志文件都会包含所有消息。

可以在对话框“设置 > 记录器 > 日志文件”(Settings > Logger > Log file) 下找到该存储路径。

组态显示的消息

1. 在对话框“设置 > 记录器”(Settings > Logger) 中的表格快捷菜单中选择条目“添加记录器”(Add logger)。
2. 在“添加记录器”(Add logger) 对话框中，选择所需条目并单击“OK” 确认。
3. 在“级别”(Level) 列中选择记录级别。
4. 如果希望删除条目，则在快捷菜单中选择“删除记录器”(Remove logger)。

7.7 组态 OPC UA 通道

7.7.1 常规序列

简介

借助 WinCC 与 OPC UA 服务器间的 OPC UA 链接，WinCC OPC UA 客户端可以通过安全连接访问 OPC UA 服务器的变量。为此将 OPC UA 服务器的数据映射为 WinCC 变量。

使用 WinCC OPC UA Configurator 进行组态。

要求

- OPC UA 服务器已激活。
- 计算机必须通过 TCP 连接。
- 通信不得被防火墙拦截。
OPC UA 服务器的端口号必须激活。

组态步骤

- 在 WinCC 项目中插入 OPC UA WinCC 通道 (页 244)。
- 创建与 OPC UA 服务器的连接 (页 244)。
- 设置通过证书进行认证 (页 246)。
- 可选：创建 WinCC 变量和组并导入 WinCC 变量以访问 OPC UA 节点。

参见

在 WinCC 项目中插入 OPC UA WinCC 通道 (页 244)

创建与 OPC UA 服务器的连接 (页 244)

设置通过证书进行认证。(页 246)

7.7.2 在 WinCC 项目中插入 OPC UA WinCC 通道

要求

- 已创建 WinCC 项目。
- 已打开变量管理。

步骤

1. 在导航区域中打开变量管理的快捷菜单。
2. 选择“添加新的驱动程序 > OPC UA WinCC Channel”。

结果

通道 OPC UA WinCC Channel 添加到变量管理中。

7.7.3 创建与 OPC UA 服务器的连接

要连接 OPC UA 服务器，需要服务器和安全设置的相关信息。

组态连接设置

使用对话框“服务器选择”(Server selection) 为与 OPC UA 服务器的通信组态所需设置。

可以在 WinCC 项目中为每个 OPC UA 服务器创建一个连接。

可以通过以下方式打开对话框：

创建新连接

1. 在项目窗口中选择“OPC UA 连接”文件夹。
2. 单击“创建新连接”(Create a new connection) 图标或从快捷菜单中选择“创建新连接”(Create a new connection)。

显示创建的连接

要求：

- 连接创建完成。

步骤:

1. 在项目窗口中选择“OPC UA 连接”(OPC UA Connections) 条目。
2. 在“OPC UA 连接”(OPC UA Connections) 数据区中选择连接。

更改创建的连接

要求:

- 连接已关闭。

步骤:

1. 在项目窗口中选择“OPC UA 连接”(OPC UA Connections) 条目。
2. 在“OPC UA 连接”(OPC UA Connections) 数据区中选择连接。
如果连接仍处于激活状态, 则选择 “断开服务器”(Disconnect Server)。
3. 要更改名称或用户标识, 在数据区中单击相应域。
4. 要更改其他设置, 在快捷菜单中选择 “属性”(Properties)。

连接数据

Serverauswahl

Discovery Server: opc.tcp://localhost:4840

OPC-UA-Server

Name: UA_Sample_Server

Endpunkt Url: opc.tcp://address_host:51210

Sicherheitsprofil: Basic128Rsa15

Sicherheitsmodus der Meldung: SignAndEncrypt

Benutzeridentität: UserName

Benutzername

Benutzername: ES_user01

Passwort:

Speichern

Auswahl Abbrechen

7.7 组态 OPC UA 通道

域/设置	内容
Discovery Server	<p>Discovery 服务器提供了可用 OPC UA 服务器的列表，该列表会显示在“Name”域中。</p> <p>输入地址或从提供的列表选择一个地址。当前使用的有效 Discovery 服务器地址存储为建议地址列表。</p> <p>如果 OPC UA 服务器未在 Discovery 服务器上注册，则使用以下格式输入所需 OPC UA 服务器的 Discovery 地址：</p> <ul style="list-style-type: none"> • <opc.tcp://Discovery 服务器地址：端口号> <p>可以使用“Refresh”按钮更新服务器显示。</p>
Name	<p>在组合框中选择 OPC UA 服务器。</p> <p>可以接受或为显示的名称输入一个新的名称。此名称用作 WinCC 项目中的连接名称。OPC UA 服务器的名称不受此域影响。</p>
终点 URL	选择 OPC UA 服务器的地址。
安全配置文件	选择 OPC UA 服务器提供的其中一个安全配置文件。
消息安全模式	选择所需的安全机制。
用户标识	<p>选择连接是否需要用户 ID 或是否允许匿名访问。</p> <p>如果已设置用户标识，则在用户名”(User Name) 和“Passwort”中添加 WinCC OPC UA 客户端的访问数据。</p>
用户名/密码	<p>输入 WinCC OPC UA 客户端可以访问 OPC UA 服务器的用户名和密码。</p> <p>由 OPC UA 服务器检查授权。对于 WinCC OPC UA 服务器，可以通过正在运行 OPC UA 服务器的 PC 的 Windows 用户管理组态授权。</p>

7.7.4 设置通过证书进行认证。

服务器证书和客户端证书

组态时区分客户端和服务器证书。仅当客户端和服务器能识别对方的证书时，才能进行安全通信。

将证书链接到各自的计算机。将 WinCC 项目移动、拷贝或复制到其它计算机后，请重复该过程，从而使各计算机都能检查其它计算机的证书。

对于通过 WinCC 通道“OPC UA WinCC 通道”进行的通信，OPC UA 服务器必须将以下客户端证书识别为可信任：

- WinCC OPC UA 客户端
在没有有效客户端证书的情况下，WinCC OPC UA Configurator 会中止尝试建立连接。
- WinCC OPC UA Runtime
在没有有效运行系统证书的情况下，不会在运行系统中显示当前值。

使用 WinCC OPC UA Configurator 控制台和 WinCC 通道诊断进行分析。

更多相关信息，请参见：

- “接口 > OPC - 开放式互连 > WinCC OPC UA 服务器 > OPC UA 的安全概念 (页 202)”
- “通信 > 通信诊断 > 诊断通道“OPC” (页 550)”

设置有效证书

创建 WinCC OPC UA 客户端的自签名证书以供安装使用。只有 OPC UA 服务器将 OPC UA 客户端证书识别为可信任，此客户端才能与该服务器相连。

创建新连接时，OPC UA 服务器会通过 WinCC OPC UA Configurator 检查客户端证书。

如果服务器未将客户端证书识别为可信任，则该连接被拒绝并以黄色标记。消息在 WinCC OPC UA Configurator 控制台上生成，并被输入到日志文件中，例如：

- Bad_CertificateUntrusted (0x801A0000) "The Certificate is not trusted."

对于 WinCC OPC UA，证书存储在 WinCC 安装路径的以下文件夹中：

WinCC OPC UA 服务器	opc\UAServer\PKI\CA
WinCC OPC UA 客户端	opc\UAClient\PKI\OPCUA

被拒绝的证书存储在相应路径的“rejected\certs”文件夹中。

要指定证书受信任，将该证书移至“certs”文件夹中。

未识别的证书不会建立连接

如果 OPC UA 通道没有将证书识别为在运行系统中受信任，则系统将拒绝建立连接并返回消息“BadCertificateUntrusted”。

要再次请求该证书，可在 WinCC OPC UA Configurator 中更改连接属性（例如：安全配置文件）。

步骤

1. 在“OPC UA 连接”(OPC UA Connections) 数据区中选择连接。
2. 在连接的快捷菜单中选择“断开服务器”(Disconnect Server)。
3. 在连接的快捷菜单中选择“属性”(Properties)。
4. 更改连接属性，例如将安全配置文件从 Basic256 更改为 Basic128。记录这项更改。
5. 关闭“属性”(Properties) 对话框。
6. 重新打开“属性”(Properties) 对话框。
7. 将设置改回初始状态。
8. 关闭“属性”(Properties) 对话框。
9. 恢复连接。

结果

组态器会使远程服务器重新发送证书并将其移至文件夹“Trusted”中。

连接已建立。

7.7.5 OPC UA 变量

支持的 OPC UA 节点

OPC UA 服务器的对象和 OPC UA 节点显示在导航区域“信息模型”(Information model) 中。

可作为 WinCC 变量导入的 OPC UA 节点，在“实例元素”(Instance elements) 数据区内的“支持”(Supported) 列中以图标 标识。OPC UA 节点只能导入一次。

支持以下数据类型的变量：

- Binary Tag
- Byte
数据类型为“有符号 8 位数”或“无符号 8 位数”
- Int16
数据类型为“有符号 16 位数”
- UInt16
数据类型为“无符号 16 位数”
- Int32
数据类型为“有符号 32 位数”

- UInt32
数据类型为“无符号 32 位数”
- Float
数据类型为“32 位浮点数 IEEE 754”或“64 位浮点数 IEEE 754”
- String
- ByteString
数据类型为原始数据变量
- DateTime
- Enumerations
数据类型为“有符号 32 位数”

WinCC 变量名称

导入 OPC UA 节点时，自动分配 WinCC 变量名称。

如果 OPC UA 服务器上的变量名称含有特殊字符，则将替换为下划线“_”。

设置

可以在“窗口 > 设置 > 变量名称”(Window > Settings > Tag names) 对话框中，按照创建的变量名称指定原则。

“前缀”(prefix) 或 “后缀”(suffix) 选项可向变量名称添加指定字符串。当组态项目监控时，必须指定一个前缀或一个后缀。

可进行下列设置：

- 将 OPC UA 节点路径输入为名称。
- 应用 OPC UA 节点名称，如有必要则通过前缀或后缀补充该名称。
- 应用 OPC UA 节点路径，必要时为其添加前缀和/或后缀。

示例

在 WinCC OPC UA 服务器上，“Spectrometer/Channel_0/ChannelStateMachine”路径中存在“CurrentState”变量。

在“前缀”(Prefix) 域中输入“Prefix_”并在“后缀”(Suffix) 域中输入“_Suffix”。

7.7 组态 OPC UA 通道

在 WinCC OPC UA 客户端的 WinCC 项目中创建以下 WinCC 变量：

设置	WinCC 变量名称
不带前缀和后缀的路径名称：	Spectrometer/Channel_0/ChannelStateMachine/CurrentState
OPC UA 节点名称	Prefix_CurrentState_Suffix
路径名称	Prefix_Spectrometer/Channel_0/ChannelStateMachine/CurrentState_Suffix

说明

仅在 WinCC OPC UA Configurator 中组态变量

只能使用 WinCC OPC UA Configurator 组态 WinCC OPC UA 变量。

在 WinCC Configuration Studio 中，只能使用变量管理来显示变量。

删除 WinCC 变量

要删除 WinCC 变量，必须与 OPC UA 服务器没有活动连接。

可以在 WinCC 变量管理中通过以下几种方式来删除导入的 WinCC 变量：

- “Tag Editor” 数据窗口：
在 WinCC 变量的快捷菜单中，选择条目“删除所选变量”(Delete selected tag) 或使用“删除”(Del) 按钮。
- “实例元素”(Instance elements) 数据窗口：
选择相关的 OPC UA 节点并取消选中“导入”(Imported) 列中的复选框。

注意
<p>删除时不能使用“撤消”功能。</p> <p>如果在数据区中删除了变量，则 WinCC 项目中的变量也将被删除。 删除不能撤消。</p>

创建变量组

要在连接下创建变量组，在快捷菜单中选择“创建新变量组”(Create a new tag group) 连接。

要更改名称，单击变量组名称。

导入 WinCC 变量时发生的动作包括：

- 在项目导航窗口中选择了连接：
 - 直接在连接下创建 WinCC 变量。
 - “Tag Editor” 数据区只显示未分配给任何变量组的变量。
- 在项目导航窗口中选择了变量组：
 - 在变量组中创建 WinCC 变量。
 - “Tag Editor” 数据区只显示已在变量组中创建的变量。

说明

无法移动 WinCC 变量

导入后，无法为变量组分配 WinCC 变量。

要删除变量组中的 WinCC 变量，需要删除该变量并重新导入。

使用通道“OPC UA WinCC 通道”移植 WinCC 项目

低于 V7.4 版本的 WinCC，已在 OPC 通道中创建了 WinCC OPC UA 连接。

移植 WinCC 项目期间，WinCC OPC UA 客户端的连接和变量也将移植到更改的结构中。

如果已导出 WinCC OPC UA 变量，请注意以下顺序：

1. 导入已导出的 WinCC OPC UA 变量。
2. 移植 WinCC 项目。

7.7.6 如何组态到 OPC UA 服务器的连接

简介

本节说明如何连接 OPC UA 服务器。

要求

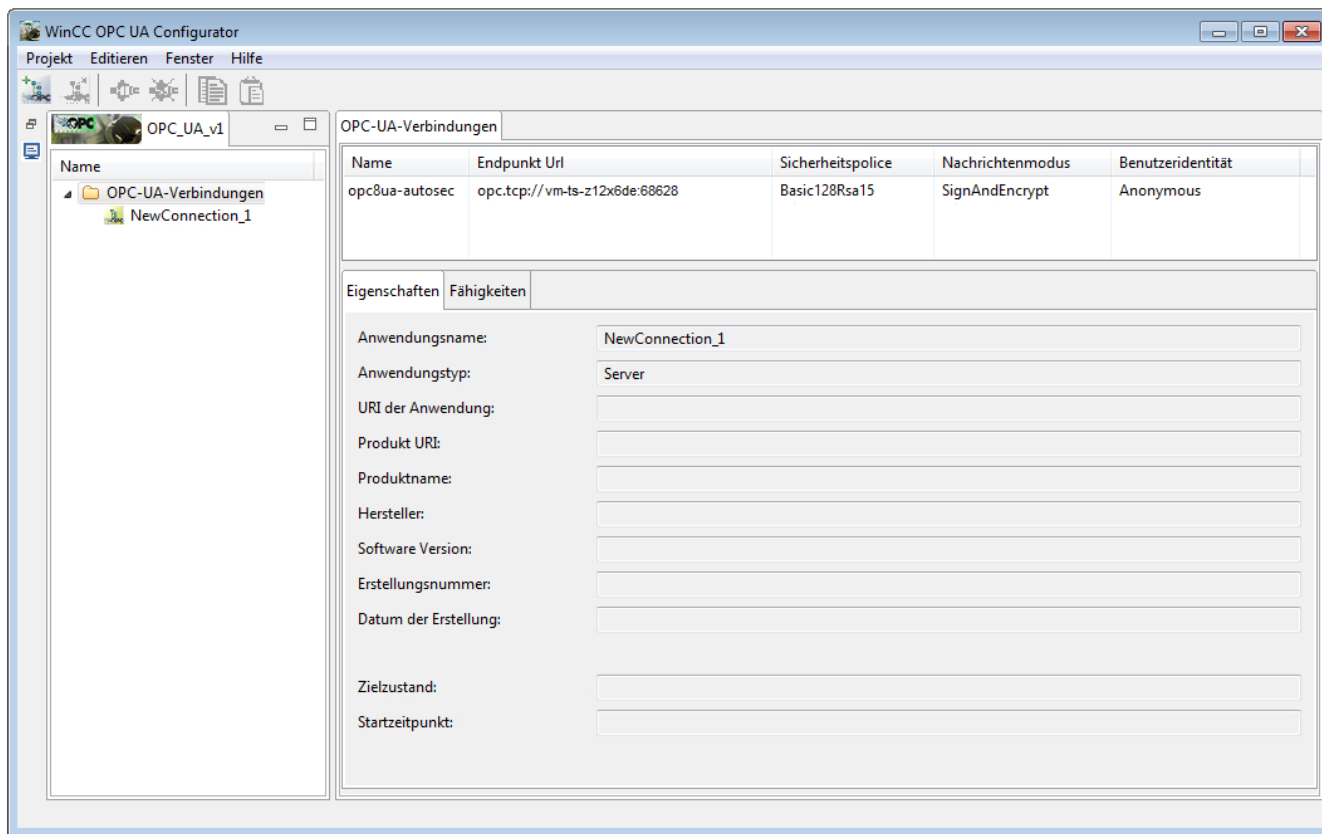
- OPC UA 服务器已激活。
- 通信不得被防火墙拦截。OPC UA 服务器的端口号必须激活。
- 必须可以从 WinCC PC 通过 IP 地址访问 OPC UA 服务器计算机。


7.7 组态 OPC UA 通道

- OPC UA 服务器信任客户端证书。
- 已在 WinCC OPC UA 客户端的 WinCC 项目中添加了通道“OPC UA WinCC Channel”。

步骤

1. 在 WinCC OPC UA 客户端上，打开 WinCC Configuration Studio 中的 WinCC 变量管理。
2. 在“OPC UA Connections”通道单元的快捷菜单中选择条目“系统参数”(System Parameters)。WinCC OPC UA Configurator 随即打开。



3. 通过“OPC UA 连接”(OPC UA Connections) 的快捷菜单打开 “服务器选择”(Server selection) 对话框。
4. 选择 Discovery 服务器或 OPC UA 服务器，或采用下列格式输入 URL：
 - `opc.tcp://<OPC-UA-ServerAddress:PortNumber>`
5. 使用  按钮更新显示
6. 通过 “名称”(Name) 组合框选择 OPC UA 服务器。

7. 选择所需的设置。

Serverauswahl

Discovery Server: opc.tcp://localhost:4840

OPC-UA-Server

Name: UA_Sample_Server

Endpoint Url: opc.tcp://address_host:51210

Sicherheitsprofil: Basic128Rsa15

Sicherheitsmodus der Meldung: SignAndEncrypt

Benutzeridentität: UserName

Benutzername

Benutzername: ES_user01

Passwort: ●●●●●●●●●●●●●●●●

Speichern

Auswahl Abbrechen

8. 单击“选择”(Selection) 创建新连接。

WinCC OPC UA Configurator 连接 WinCC OPC UA 客户端与 OPC UA 服务器。
如果 OPC UA 服务器未识别客户端证书，，则不建立连接。

- 连接以黄色标记。在“OPC UA 连接”(OPC UA Connections) 数据窗口中，以蓝色字母显示连接。
- 要查看 OPC UA 服务器消息，打开 WinCC OPC UA Configurator 的“控制台”输出窗口。消息还会储存在日志文件中。
确保 OPC UA 服务器接受客户端证书。

结果

已建立与 OPC UA 服务器的连接。

显示 OPC UA 服务器的信息模型。

7.7.7 导入 OPC UA 节点作为 WinCC 变量

简介

本节说明如何将 OPC UA 节点作为 WinCC 变量导入到 WinCC 变量管理中。

已在 WinCC Configuration Studio 中的通道单元“OPC UA 连接”下创建了 OPC UA WinCC 通道变量。

说明

仅在 WinCC OPC UA Configurator 中组态变量

只能使用 WinCC OPC UA Configurator 组态 WinCC OPC UA 变量。

在 WinCC Configuration Studio 中，只能使用变量管理来显示变量。

要求

- 已建立与 OPC UA 服务器的连接（绿色状态）。
- 已设置 WinCC 变量名称。

步骤

1. 打开 WinCC OPC UA Configurator。
2. 在项目窗口中的“OPC UA 连接”(OPC UA Connections) 下选择连接或变量组。
OPC UA 节点显示在“信息模型”(Information model) 导航区域中。
3. 在信息模型树中导航到包含所需节点的父节点。
4. 在“实例元素”(Instance elements) 数据区中，选择要映射的变量。
可作为 WinCC 变量导入的 OPC UA 节点在“支持”(Supported) 列中由符号 标识。
5. 在 OPC UA 节点的快捷菜单中选择“导入所选条目”(Import selected items)。
要在 WinCC 中导入选定对象的所有支持变量，选择“导入所有支持条目”(Import all supported items)。

结果

已创建相应的 WinCC 变量并显示在 Tag Editor 中。在“导入”(Imported) 列中选中复选框。

将在 WinCC 变量管理中看到新组态的 WinCC 变量。但是，只能在 OPC 服务器上或在 WinCC OPC UA Configurator 中更改变量属性。

可选步骤

要将 OPC UA 节点作为 WinCC 变量导入，也可以使用以下方式：

“实例元素”数据区

1. 在项目窗口中的“OPC UA 连接”(OPC UA Connections) 下选择连接或变量组。
2. 在信息模型中导航到包含所需节点的父节点。
3. 在“实例元素”(Instance elements) 数据区中选择支持的节点。
4. 在“导入”(Imported) 列中单击符号 。
在 Tag Editor 中创建 WinCC 变量。如果在项目窗口中选择了变量组，则将以组的形式创建 WinCC 变量。
在“导入”(Imported) 列中将节点标记为导入：

工具栏

1. 在项目窗口中的“OPC UA 连接”(OPC UA Connections) 下选择连接或变量组。
2. 在信息模型中导航到包含所需节点的父节点。
3. 在“实例元素”(Instance elements) 数据区中选择一个或多个支持的节点。
“复制所选元素”(Copy selected elements) 图标已激活。
4. 单击“复制所选元素”(Copy selected elements) 图标。
5. 单击 Tag Editor 窗口。
“导入所选条目”(Import selected items) 图标已激活。
6. 要创建 WinCC 变量，单击“导入所选条目”(Import selected items) 图标。

7.8 通信受到干扰时的误差处理

简介

通讯测试的过程与 WinCC 的使用方式无关。

有关通道诊断的更多信息，请参见“通信 > 通信诊断”(Communication > Communication Diagnostics)。

WinCC 用作 OPC UA 服务器

在 WinCC OPC UA 客户端上使用通道诊断来检查是否可以与 WinCC OPC UA 服务器建立连接。

WinCC 用作 OPC UA 客户端

在 WinCC OPC UA 客户端上使用通道诊断来检查是否可以与 WinCC OPC UA 服务器建立连接。

PROFIBUS FMS

8.1 WinCC 通道“PROFIBUS FMS”

简介

“PROFIBUS FMS” 通讯驱动程序用于建立 WinCC 站与自动化系统（如 S5 或 S7）之间的连接。

本章描述了

- 如何使用“PROFIBUS FMS” 通道组态数据传送
- 如何组态连接和变量。

通道单元

通讯驱动程序具有通道单元。这提供下列功能：

- 通道单元 PROFIBUS FMS，用于 SIMATIC NET PROFIBUS（通讯模块 DP 5613）。

说明

可在组态模式及运行系统中组态逻辑连接。将对这两种方法进行介绍。

只有 WinCC 可以使用“PROFIBUS FMS” 通道从连接的 AS 请求数据。不能从 AS 发送数据。

8.2 变量的数据类型

简介

为逻辑连接定义所需变量。从 WinCC 的角度而言，可以访问以下数据类型：

- 二进制变量
- 无符号 8 位数
- 有符号 8 位数
- 无符号 16 位数
- 有符号 16 位数
- 无符号 32 位数
- 有符号 32 位数
- 浮点数 32 位 IEEE 754
- 文本变量，8 位字符集。
- 原始数据类型

8.3 组态通道

8.3.1 组态连接

8.3.1.1 组态连接

简介

自动化系统必须通过适当的通讯处理器连接到 PROFIBUS。PROFIBUS FMS 协议必须受硬件/软件支持。

例如，在自动化系统 S5-115U、S5-135U 和 S5-155U 中，使用通讯处理器 CP 5431 FMS。

在 WinCC 系统中，必须安装通讯模块 CP 5613 和软件 SIMATIC NET。

组态连接时，将以不同的方式来处理下列情况：

- 在组态模式中组态
- 在运行系统中组态

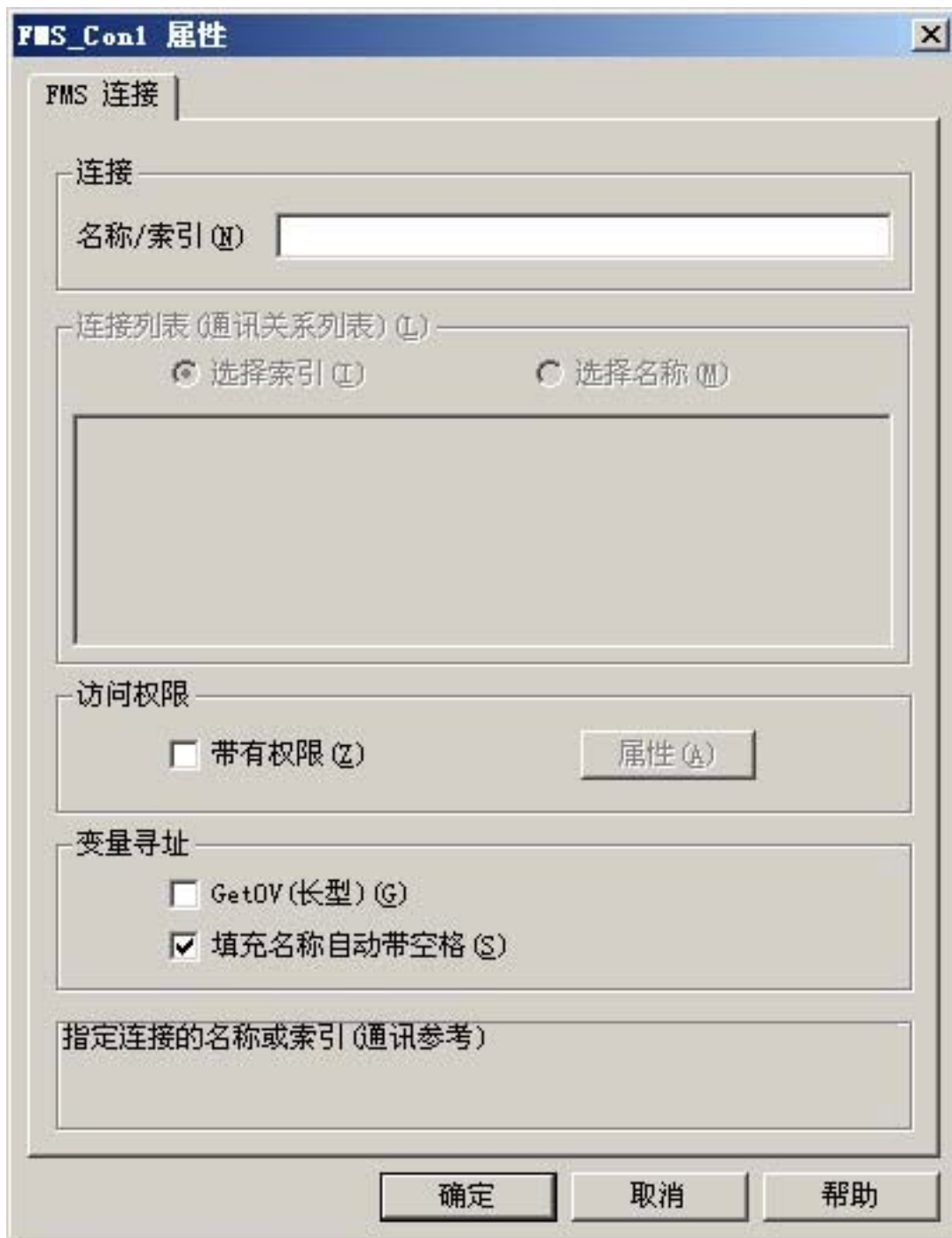
8.3.1.2 在组态模式中组态连接

步骤

1. 在变量管理的导航区域中，从“PROFIBUS FMS”通信驱动程序树结构下选择通道单元“PROFIBUS FMS”。
2. 在通道单元的快捷菜单中选择“新建连接”(New Connection) 条目。
3. 输入连接的名称。

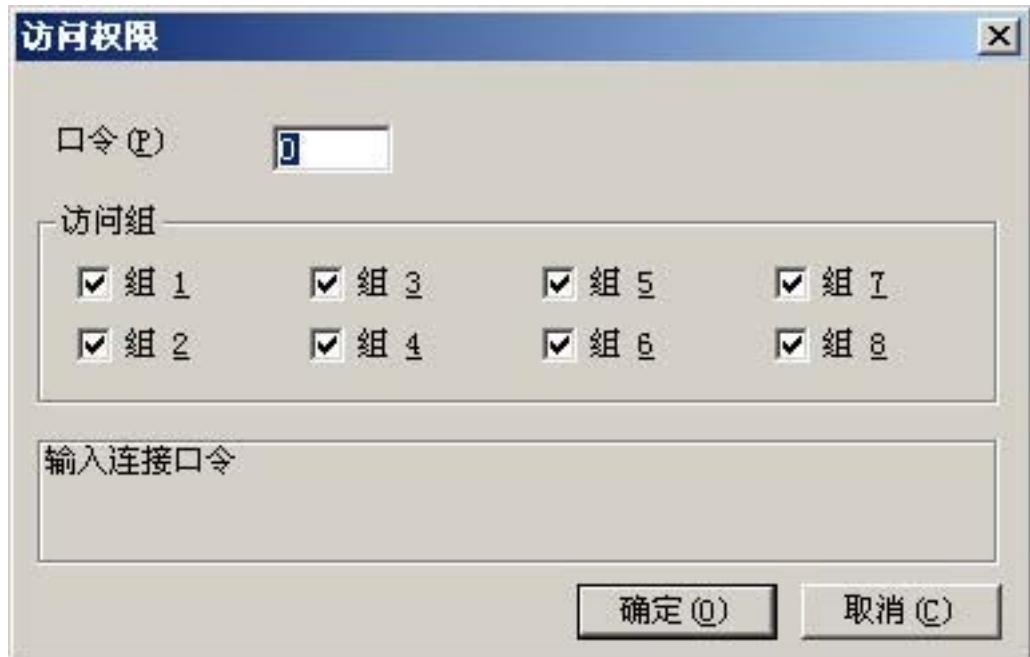
8.3 组态通道

4. 在连接的快捷菜单中选择“连接参数”(Connection parameters) 条目。
随即显示以连接名称作为标题的对话框



5. 在“名称/索引”(Name/Index) 字段中输入逻辑连接的名称或索引。
逻辑连接的“名称/索引”(Name/Index) 必须在本地通信关系列表 (CRL) 中组态。
6. 如果已为逻辑连接分配了访问权限，请选中“带有限权”域。

7. 通过“属性”按钮，可打开“访问权限”对话框：



8. 现在，可以输入逻辑连接的密码。单击“确定”(OK)按钮，关闭对话框。
9. 在“变量寻址”(Tag addressing)区域中，定义所需的参数。单击“确定”(OK)关闭对话框。

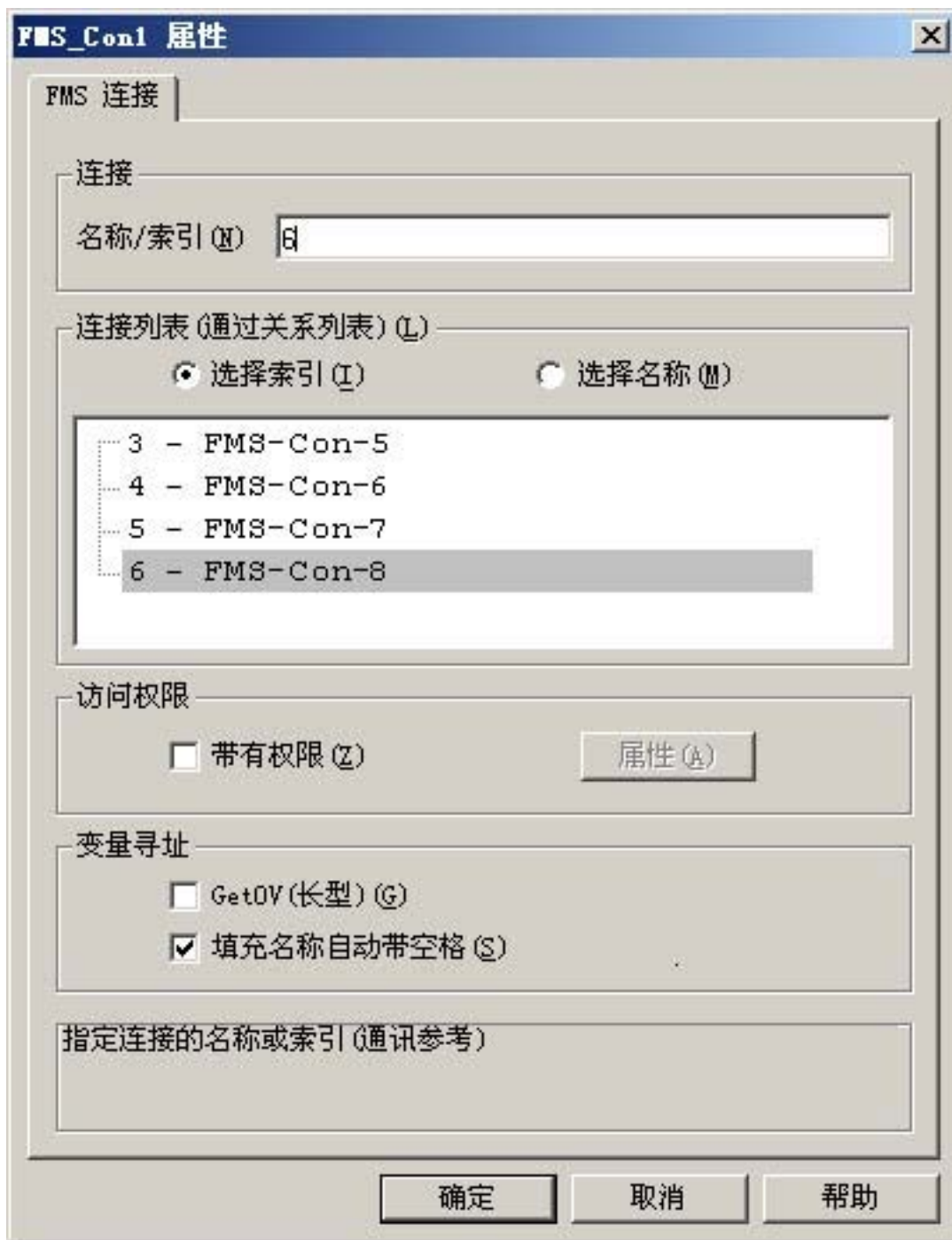
说明

要组态连接的“名称”或“索引”，必须访问通信关系列表，此列表可使用 SIMATIC NCM PC 或 STEP 7 创建。

8.3.1.3 如何在运行系统中组态连接

步骤

1. 选择连接，然后在快捷菜单中打开“连接属性”(Connection properties) 对话框。
2. 单击“属性”按钮。
随即显示以连接名称为标题的对话框。



3. 通信关系列表 (CRL) 中的已组态连接显示在“连接列表 (通信关系列表)”区域中。选择该连接应与“连接”区域内“名称/索引”域中的相应索引，还是名称一起使用。
4. 如果已为逻辑连接分配了访问权限，请选中“带有权限”域。
5. 通过“属性”按钮，可打开“访问权限”对话框。



6. 现在，可以输入逻辑连接的密码。单击“确定”(OK) 按钮，关闭对话框。
7. 在“变量寻址”(Tag addressing) 区域中，定义所需的参数。单击“确定”(OK) 关闭对话框。

8.3.2 组态变量

8.3.2.1 组态变量

简介


变量地址将根据 PROFIBUS FMS 协议的地址结构来输入。

组态地址时，应对下列情况进行区分：

- 在组态模式中
- 在运行期间

8.3.2.2 在组态模式中组态变量地址

步骤

1. 在“数据类型”(Data type) 字段中，选择变量所需的数据类型（例如：有符号 16 位数）。
2. 打开“属性”(Properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。




3. 在“名称/索引”域中，输入来自远程自动化系统的变量的名称或索引。
4. 如果变量为结构化变量（结构或数组），则还必须在“子索引”域中输入应读取或写入的变量部分（结构组件或数组元素）。

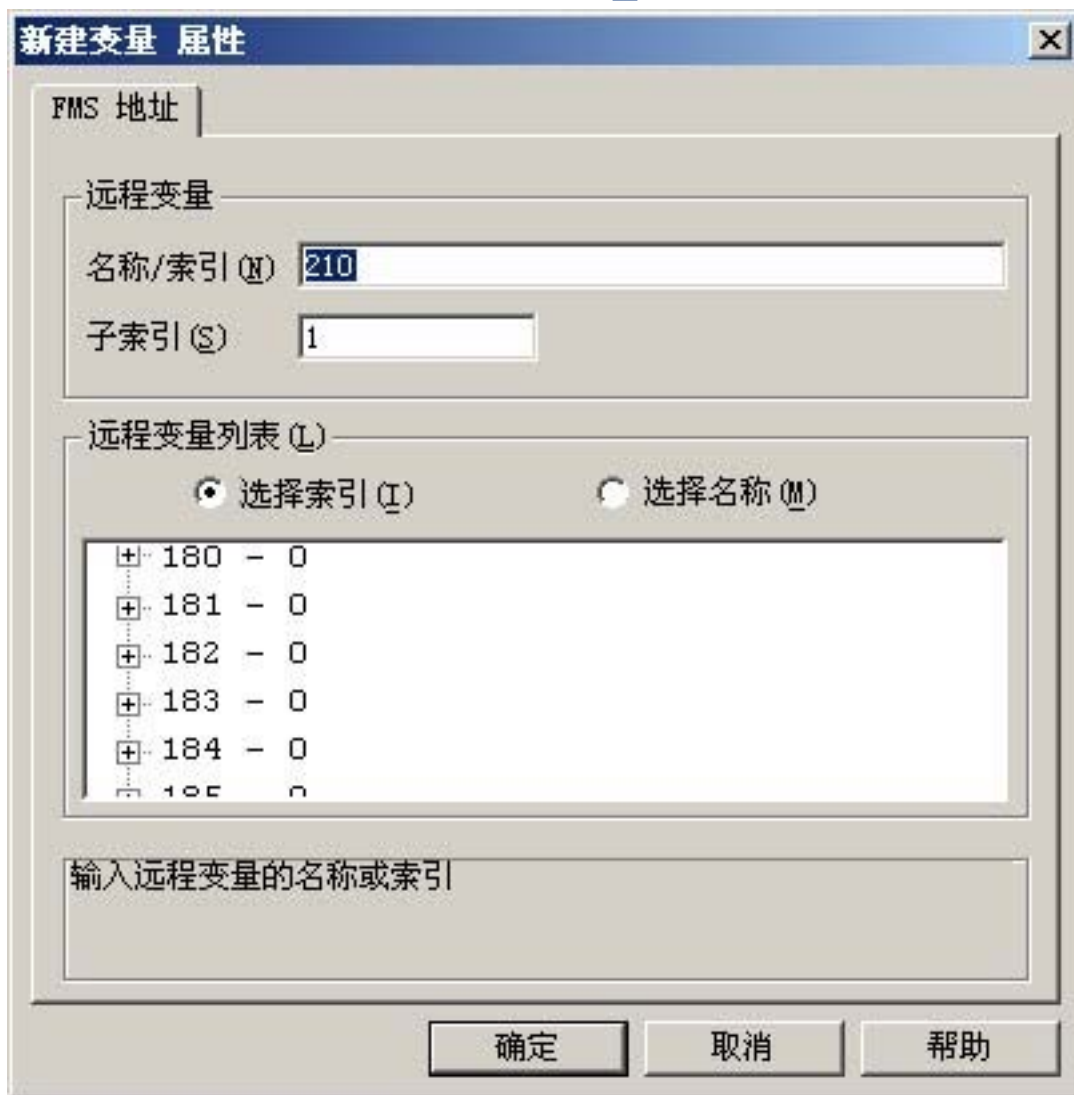
说明

名称或索引以及“子索引”必须与“对象列表”中外部自动化系统的逻辑连接的名称或索引相匹配。否则，运行期间将无法访问该变量。

8.3.2.3 如何在运行期间组态变量地址

步骤

1. 在“数据类型”(Data type) 字段中，选择变量所需的数据类型（例如：有符号 16 位数）。
2. 打开“属性”(Properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



3. 可通过所选连接进行访问的变量将列在“远程变量列表”区域中。选择是使用“远程变量”区域中的索引还是名称来选择变量。
4. 在“远程变量列表”中，双击所需变量以将其选定。
5. 所选变量将显示在“远程变量”区域中的“名称/索引”域中。
6. 如果变量为结构化变量（结构或数组），则还必须在“子索引”域中输入应读取或写入的变量部分（结构组件或数组元素）。

S5 Ethernet Layer 4

9.1 WinCC 通道“SIMATIC S5 Ethernet Layer 4”

简介

例如，该通信驱动程序可用于通过 ISO 传输协议或 TCP/IP 协议，来连接自动化系统 SIMATIC S5-115U/H、SIMATIC S5-135U 和 SIMATIC S5-155U/H。

根据所用的通信协议，将使用以下通信伙伴。

通信协议	WinCC 侧	SIMATIC S5 端
ISO 传输协议	CP1612（与 3Com 兼容） CP1613 CP1623	CP1430 TF
TCP/IP（符合 RFC1006）	CP1612（与 3Com 兼容） CP1613 CP1623	CP1430 TCP

使用此通道时，不需要本地数据库。

通道单元

通信驱动程序具有两个通道单元“CP1413-x”，最多可操作两个 CP1612、CP1613 或 CP1623。通道单元的功能完全相同。只是两个 CP 的逻辑设备名称不同。CP1623 与 CP1613 相同，但通过 PCI Express 进行操作。

可使用第三个通道单元“TCP/IP”通过 TCP/IP 协议与 CP1612、CP1613 或 CP1623 建立通信。

逻辑设备名称可在通道单元的系统参数中进行更改。在此，也可以为所用协议设置参数。

存在下列应用能力：

- 通道单元“S5-Transport (CP 1413-1)”，适用于 SIMATIC 工业以太网的通信模块 (CP 1612/1613/1623)。
- 通道单元“S5-Transport (CP 1413-2)”，适用于 SIMATIC 工业以太网的通信模块 (CP 1612/1613/1623)。
- 通道单元“S5-Transport (TCP/IP)”，适用于 SIMATIC 工业以太网的通信模块 (CP 1612/1613/1623)。

9.2 变量的数据类型

简介

为逻辑连接定义所需变量。从 WinCC 的角度而言，可以访问以下数据类型：

- 二进制变量
- 无符号 8 位数
- 有符号 8 位数
- 无符号 16 位数
- 有符号 16 位数
- 无符号 32 位数
- 有符号 32 位数
- 浮点数 32 位 IEEE 754
- 文本变量，8 位字符集
- 原始数据类型

9.3 组态通道

9.3.1 组态通道“SIMATIC S5 Ethernet Layer 4”

简介

组态通道“SIMATIC S5 Ethernet Layer 4”需要以下步骤。

1. 组态连接
2. 组态变量
3. 系统参数组态

9.3.2 如何组态连接

简介

连接参数对于使用的所有协议几乎完全相同。下例描述了使用 ISO 传输协议和通道单元“CP1413-x”的通信。

实施 TCP/IP 协议时，输入 AS 的 IP 地址而非以太网地址。IP 地址由四个以圆点分隔的数值组成。数值必须位于 0-255 的范围内。

对于逻辑连接，WinCC 在传输层中建立一个用于读取（“读功能”区域）的连接和一个用于写入（“写功能”区域）的连接。这两个功能的地址参数在对话框中定义。仅当这两个连接都已建立时，才会将逻辑连接指示为“已建立”。

读功能的分配

WinCC 端	SIMATIC S5 端
FETCH-Active (请求“READ-Active”)	READ-Passive (请求“READ-Passive”)
FETCH-Passive (请求“WRITE-Passive”)	WRITE-Active (请求“WRITE-Active”)

说明

如果 AS 的数据为发送主动，即在连接参数中读功能被设置为“读取被动”，则无法将二进制或字节变量写入 AS 的数据区域。

如果已将至少一个报文从 AS 发送到 WinCC，则只能为“读取被动”连接分配“正常”状态。

写功能的分配

WinCC 端	SIMATIC S5 端
请求“WRITE Active”	请求“WRITE Passive”

步骤

1. 从连接的快捷菜单中选择“连接参数”(Connection parameters) 条目。
“连接属性”(Connection properties) 对话框随即打开。



2. 在“以太网地址 AG”域中，输入工业以太网总线上 SIMATIC S5 的站地址。实施 TCP/IP 协议时，在“IP 地址 AG”域中输入 IP 地址。
3. 在 WinCC 系统中为读功能定义参数。这些参数独立于 SIMATIC S5 中所使用的请求。
4. 然后，在“自身 TSAP”分配域中输入组态 CP 1430 TF 时在“远程参数”中被组态为“TSAP”的值。
5. 现在，在“远程 TSAP”分配域中输入组态 CP 1430 TF 时在“本地参数”中被组态为“TSAP”的值。
6. 为写功能相应地定义“自身 TSAP”和“远程 TSAP”参数。

说明

在“TSAP”条目中，不得使用任何空格。

9.3.3 组态变量

9.3.3.1 组态变量

简介

对于 WinCC 与 AS 之间通过通道“SIMATIC S5 Ethernet Layer 4”进行的连接，可在 WinCC 中创建不同数据类型的变量。下面将说明如何组态这些数据类型的变量。

- 变量的地址
- 组态按位访问的变量
- 组态按字节访问的变量
- 组态按字访问的变量
- 组态原始数据变量

9.3.3.2 变量的地址

简介

变量地址根据 SIMATIC S5 的地址结构进行输入。

视变量类型而定，对 AS 中存储器区域的访问有按位、按字节或按字这几种类型。为此，每次写请求时均从 AS 读取已赋址的存储器区域，并修改相应的位和/或字节。此后，数据写回到 AS 的存储器。

说明

AS 在读取数据区中执行的更改会在写回到该数据区时被覆盖。

如果 AS 的数据为发送主动，即在连接参数中读功能被设置为“读取被动”，则无法将二进制或字节变量写入 AS 的数据区。


9.3 组态通道

组态变量的地址而不依赖于变量类型：

- 对于“二进制”或“8 位数”类型的变量，首先打开“位/字节变量”对话框，在此对话框中定义按位或按字节访问 AS 的存储器区域。
然后，在“地址属性”对话框中定义变量在 AS 存储器中的地址。
- 对于按字访问的变量，在“地址属性”对话框中定义变量在 AS 存储器中的地址。
由于对 AS 存储器的访问是按字进行的，所以不会打开“位/字节变量”对话框。

9.3.3.3 如何组态按位访问的变量

步骤

1. 选择连接，然后在快捷菜单中打开“位/字节变量”(Bit/Byte tag) 对话框。为此，请单击“地址”(Address) 字段，然后单击  按钮。
2. 单击“选择”(Select) 按钮。“位/字节变量”(Bit/Byte tag) 对话框随即打开。



3. 使用复选框定义是否启用对存储区中某些位的读写访问。
4. 在选择域中选择 AS 存储器的寻址方法，例如“字”或“字节”。
5. 在选择域中选择要更改的位号。
6. 使用“选择”按钮打开“地址属性”对话框，以定义 AS 中的变量地址。

说明


在 S5 中，可以按字节寻址标志位、输入和输出，按字寻址数据块（DB、DX）。

激活“访问位”(Access a bit) 复选框会影响“地址属性”(Address properties) 对话框中字段的显示。

对于按字访问的变量，不会打开所述的“位/字节变量”(Bit/Byte tag) 对话框，因为变量地址是按字存储，从而导致对 PLC 存储器的访问也是按字进行。

9.3.3.4 如何组态按字节访问的变量

步骤

1. 选择变量，然后在“数据类型”(Data type) 字段中选择数据类型“无符号 8 位数”(Unsigned 8-bit value) 或“有符号 8 位数”(Signed 8-bit value)。
2. 选择连接，然后在快捷菜单中打开“位/字节变量”(Bit/Byte tag) 对话窗口。为此，请单击“地址”(Address) 字段，然后单击  按钮。

3. 单击“选择”(Select)按钮。“位/字节变量”(Bit/Byte tag)对话框随即打开。



4. 使用复选框定义是否启用对存储区中某些字节的读写访问。
5. 在选择域中，只有“字”显示为 AS 存储器的寻址类型。
6. 在选择域中，选择要更改的字节号。
7. 使用“选择”按钮打开“地址属性”对话框，以定义 AS 中的变量地址。

说明

在 S5 中，可以按字节寻址标志位、输入和输出，按字寻址数据块（DB、DX）。

选择“访问字节”(Access a byte)复选框会影响“地址属性”(Address properties)对话框中字段的显示。

对于按字访问的变量，不会打开所述的“位/字节变量”(Bit/Byte tag)对话框，因为变量地址是按字存储，从而导致对 PLC 存储器的访问也是按字进行。


9.3.3.5 如何组态按字访问的变量

简介

使用此处介绍的对话框定义 AS 中的变量地址。

- 对于“二进制”或“8 位数”类型的变量，首先打开“位/字节变量”对话框，在此对话框中定义按位或按字节访问 AS 的存储器区域。
- 对于按字访问的变量，不会打开“位/字节变量”对话框，因为变量地址是按字存储，从而导致对 AS 存储器的访问也是按字进行。

步骤

1. 选择变量，然后在“数据类型”(Data type) 字段中选择变量所需的数据类型（例如：有符号 16 位数）。
2. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



3. 在“地址”选项卡的“数据区”域中，选择变量是位于数据块、标记区域、输入区域还是输出区域。
4. 如果变量位于数据块中，则还会显示“DB 号”域。在此，输入数据块的编号。
5. 在“寻址”域中输入寻址类型。通常，可以使用默认定义。
6. 在相应域（例如“DW”）中，输入地址。

说明

对于“二进制”或“8 位数”类型的变量，此对话框中显示的域取决于在“位/字节变量”对话框中对“访问位/字节”做出的选择。

如果要写入按字访问数据区的变量，则起始地址必须在左字节中，而变量长度必须为偶数。

9.3.3.6 如何组态原始数据变量


简介

以下描述了如何定义原始数据变量的地址。

说明

如果要写入按字访问数据区的变量，则起始地址必须在左字节中，而变量长度必须为偶数。

步骤

1. 选择变量，然后在“数据类型”(Data type) 字段中选择条目“原始数据类型”(Raw data type)。
2. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



3. 在“数据区”域中，选择变量是位于数据块、标记、输入区域还是输出区域。
4. 如果变量位于数据块中，则还会显示“DB 号”域。在此，输入数据块的编号。
5. 在“寻址”域中输入寻址类型。通常，可以使用默认定义。
6. 在下面的域中输入数据地址。域标签取决于“寻址”域中的条目，例如，对于“字”寻址则为“DBW”。
7. 在“长度”域中输入原始数据块的长度（以字节为单位）。
8. 在“原始数据类型”区域中，定义与原始数据变量相关的类型。

9.3.4 系统参数

9.3.4.1 通道单元的系统参数

简介

如果需要与 WinCC 标准设置不同的组态，则可使用通道单元的“系统参数”对话框进行所需的全部更改。

系统参数对于使用的所有协议几乎完全相同。实施 TCP/IP 协议时，只有安装期间所给定的设备名称不同。

以下各项均可更改：

- 设备名称
- 传输参数

说明

系统参数适用于 AS 中的所有 CP。

设备名称

WinCC 和自动化系统之间的通信通过逻辑设备名称进行。这些名称是在安装通信模块时分配的，并且是单元特定的。设备名称代表逻辑设备名称。实施 ISO 传输协议时，将名称“/CP_H1_1:/SCP”指定为逻辑设备名称的默认定义；实施 TCP/IP 协议时，将名称“/TCP_IP:/SCP”指定为逻辑设备名称的默认定义。

说明

使用 TCP/IP 协议时，必须检查 WinCC 中的设备名称是否与“设置 PG/PC 接口”对话框中的“应用程序访问点”相匹配。还必须在“设置 PG/PC 接口”中更改设备名称。

传输参数

在传输参数中进行通道单元的特定设置，例如 PDU 大小、尝试建立次数等。

9.3.4.2 如何更改设备名称

简介

使用系统参数（如逻辑设备名称或传输参数）设置通道单元的参数。

系统参数对于使用的所有协议几乎完全相同。

下例描述了使用 ISO 传输协议和通道单元“CP1413-x”的通讯。

实施 TCP/IP 协议时，只有安装期间所给定的设备名称不同。

步骤

1. 选择通道单元，然后通过上下文菜单打开“系统参数”对话框。
2. 选择“设备名称”选项卡。



3. 现在，可使用鼠标来选择以粗体显示的设备名称，并在该设备名称的名称域中单击鼠标对其进行更改。

说明

设备名称在安装硬件驱动程序时定义。只有在安装硬件驱动程序时定义了其它名称（不推荐），才必须也更改此处的设备名称。

9.3.4.3 如何更改传输参数

步骤

1. 选择通道单元，然后通过上下文菜单打开“系统参数”对话框。
2. 选择“传输参数”选项卡。



3. 将“PDU 大小”的值设置为在通讯模块 CP 1430 上组态的值。
4. 在“建立尝试”域中，定义尝试建立连接的频率。

9.3 组态通道

5. 在“重复发送持续时间”区域中选择“无限”。
6. 在“确认时间”域中输入值（如 30），以便在通讯伙伴未在此时间内做出响应时，至多在 30 秒后通知您变量状态（例如，AS 处于“停止”状态）。

9.4 附录

9.4.1 附录

简介

附录中提供有关通道“SIMATIC S5 Ethernet Layer 4”的附加信息。

9.4.2 内部错误代码和常量

9.4.2.1 内部错误代码和常量

简介

下表包含了最重要的错误代码和常量。这些信息仅供“内部人员”使用。因此，我们就不详细介绍这些代码和常量的含义了。

- 连接受干扰时的错误代码
- iNA960 消息
- SCI 消息

9.4.2.2 连接受干扰时的错误代码

简介

本部分列出了最重要的错误代码。如果出现该表中未列出的错误代码所对应的错误，请拨打 WinCC 热线。

Fehler_0002-INVALID_RQ

请求块出现故障。

Fehler_0004-NO_RESOURCES

CP 中无可用资源。

Fehler_0006-UNKNOWN_REFERENCE

所定义的 OPEN 参考不正确。

Fehler_0008-BUFFER_TOO_SHORT

用户缓冲区过短。

Fehler_0010-BUFFER_TOO_LONG

用户缓冲区过长。

Fehler_0012-ILLEGAL_REQ

所定义的“negot_options” 不正确。

Fehler_0014-REM_ABORT

连接被远程站中止。

Fehler_0016-LOC_TIMEOUT

超时。

Fehler_0018-UNKNOWN_CONN_CLASS

连接类别未知。

Fehler_0020-DUP_REQ

连接已建立。

Fehler_0022-CONN_REJECT

连接请求被远程拒绝。

Fehler_0024-NEGOT_FAILED

“negot-option” 出现故障，连接中止。

Fehler_0026-ILLEGAL_ADDRESS

传输地址有误。

Fehler_0028-NETWORK_ERROR

总线或 CP 受到干扰。

Fehler_0030-PROTOCOL_ERR

协议错误。

Fehler_0032-ILLEGAL_RB_LENGTH

请求块长度不正确。

Fehler_0784-E_NO_HW

未找到通讯硬件。

- 通讯模块存在缺陷。
- 通讯模块安装得不正确。
- 所定义的端口地址错误。

Fehler_0786-E_CNF

驱动程序组态不正确，或者注册表中的参数无效。

Fehler_0787-E_BAUDRATE

波特率不正确或所定义的中断向量不正确。

Fehler_0788-E_HSA

所定义的 HSA（最高站地址）不正确。

Fehler_0789-E_TS

所定义的本地参与者号 (TS_ADR) 已分配。

Fehler_0791-E_INT_NOT_PROV

所定义的中断向量 (IRQ) 在通讯模块上不可用。

Fehler_0792-E_INT_BUSY

所定义的中断向量 (IRQ) 在通讯模块上已被占用。

Fehler_0800-E_NO_FILE

无法装载所选的通讯驱动程序；文件未找到。

- 通讯驱动程序安装得不正确。

Fehler_0897-E_LOGDEV

注册表中未定义逻辑设备。

- 通讯驱动程序安装得不正确。
- 注册表项已损坏或删除。
- 使用“设置 PG/PC 接口”程序检查逻辑设备名称的设置。
- 检查“系统参数 - 设备”掩码中逻辑设备名称的设置。

Fehler_0898-E_L2DRIVER

注册表中缺少条目“L2DRIVER”。

- 模块错误或模块安装得不正确。

Fehler_0900-E_L4DRIVER

注册表中缺少条目“L4DRIVER”。

- 模块错误或模块安装得不正确。

Fehler_30000-EC_WATCHDOG

监视狗错误。

Fehler_30001-EC_PDUERROR

PDU 不是所需的。

Fehler_30005-EC_ONLERROR

装载 S7-Online-DLL 时出现故障。

9.4.2.3 iNA960 消息**常规 iNA960 消息**

OK_RESP	1	0x01	请求已正确执行
OK_EOM_RESP	3	0x03	数据块已正确接收
OK_DECIDE_REQ_RESP	5	0x05	请求已正确执行
OK_CLOSED_RESP	7	0x07	连接已由本地用户中止

iNA960 错误消息

INVALID_REQ	2	0x02	请求块出现故障
NO_RESOURCES	4	0x04	CP 中无可用资源
UNKNOWN_REFERENCE	6	0x06	所定义的 OPEN 参考不正确
BUFFER_TOO_SHORT	8	0x08	用户缓冲区过短
BUFFER_TOO_LONG	10	0x0A	用户缓冲区过长
ILLEGAL_REQ	12	0x0C	所定义的“negot_options”不正确
REM_ABORT	14	0x0E	连接被远程站中止
LOC_TIMEOUT	16	0x10	超时
UNKNOWN_CONNECTION	18	0x12	连接类别未知
DUP_REQ	20	0x14	连接已建立
CONN_REJECT	22	0x16	连接请求被远程拒绝
NEGOT_FAILED	24	0x18	negot-option 出现故障，连接中止
ILLEGAL_ADDRESS	26	0x1A	传输地址有误
NETWORK_ERROR	28	0x1C	总线或 CP 受到干扰

9.4 附录

PROTOCOL_ERR	30	0x1E	协议错误
ILLEGAL_RB_LENGTH	32	0x20	请求块长度不正确

9.4.2.4 SCI 消息

请参见《SINEC 通讯接口 SCI》手册中的说明 (A/5-15)。

SCI 消息

SCP_OK	0	0x00	无错
SCP_INCONS	201	0xC9	较小的设备编号不是 00
SCP_RESOURCE	202	0xCA	DPRAM 请求无效
SCP_CONFIG	203	0xCB	组态错误 (NUM_PROCS)
SCP_NOCONFIG	204	0xCC	SCP 驱动程序未组态
SCP_PARAM	206	0xCE	模式不正确
SCP_DEVOPEN	207	0xCF	已执行打开操作
SCP_BOARD	208	0xD0	未插入/不识别电路板
SCP_SOFTWARE	209	0xD1	IRQ 错误或软件未找到
SCP_MEM	210	0xD2	DPRAM 内存小
SCP_MODE	211	0xD3	下载过程尚未结束
SCP_LOADER	212	0xD4	装载程序无响应
SCP_SIGNAL	213	0xD5	过程被异步启动
SCP_NOMESS	215	0xD7	该过程中没有消息到达
SCP_USERMEM	216	0xD8	length_of_buffer 过小
SCP_WINDOW	217	0xD9	SEND 的调用次数过多
SCP_TIMEOUT	219	0xDB	SCP 超时
SCP_ATTACH	220	0xDC	未执行重置/通道仍处于活动状态
SCP_ILLEGAL_REQUEST	221	0xDD	非法请求
SCP_ERECOVERF	223	0xDF	未通过 scp_receive 获取缓冲区
SCP_ECLOSED	224	0xE0	已为连接分配了所有缓冲区
EUSERMAX	225	0xE1	

SCP_EINTR	226	0xE2	
SCP_BOARD_OPEN	231	0xE7	
SCP_NO_WIN_SERV	233	0xE9	
EPROTECT	234	0xEA	未找到许可证

SCI 消息

SCP_DB_FILE_DOES_NOT_EXIST	240	0xF0	
SCP_DB_FILE_CLOSE_NOT_OK	241	0xF1	
SCP_SEND_NOT_SUCCESSFUL	242	0xF2	
SCP_RECEIVE_NOT_SUCCESSFUL	243	0xF3	
SCP_NO_DEVICE_AVAILABLE	244	0xF4	
SCP_ILLEGAL_SUBSYSTEM	245	0xF5	
SCP_ILLEGAL_OPCODE	246	0xF6	
SCP_BUFFER_TOO_SHORT	247	0xF7	
SCP_BUFFER_1_TOO_SHORT	248	0xF8	
SCP_ILLEGAL_PROTOCOL_SEQUENCE	249	0xF9	
SCP_ILLEGAL_PDU_ARRIVED	250	0xFA	
SCP_REQUEST_ERROR	251	0xFB	
SCP_NO_LICENSE	252	0xFC	

SCP 接口的其它在线 DLL 消息

E_TIMER_INIT	768	0x0300	WIN Set-timer 请求不成功
E_INIT_COM	769	0x0301	
E_NO_HW	784	0x0310	未找到 MPI 模块
E_HW_DEFECT	785	0x0311	硬件出现问题
E_CNF	786	0x0312	组态参数不正确
E_BAUDRATE	787	0x0313	波特率/中断向量不正确
E_HSA	788	0x0314	组态的 HSA 不正确
E_TS	789	0x0315	所组态的地址已分配
E_OCC	790	0x0316	HW_Device 已分配

9.4 附录

E_INT_NOT_PROV	791	0x0317	中断不可用
E_INT_BUSY	792	0x0318	中断已占用
E_SAP	793	0x0319	SAP 禁用: SAP 未被占用
E_UNPLUGGED	794	0x031a	未找到远程站
E_SYNI	795	0x031b	发生 Syni 错误
E_AMPRO	796	0x031c	AMPRO 2 报告了系统错误
E_BUFFER_SIZE	797	0x031d	未创建此大小的缓冲区
E_NO_FILE	800	0x0320	未找到 DLL/VxD 文件或注册表项已遭到破坏
E_NO_ENTRY	801	0x0321	DLL 中不存在地址
E_VERSION	816	0x0330	SMC 驱动程序和 SMC 固件之间发生版本冲突
E_COMCNF	817	0x0331	COM 端口组态出现问题
E_NO_SMC	818	0x0332	SMC 不再响应
E_COMMBADID	819	0x0333	COM 端口未组态
E_COMMOPEN	820	0x0334	COM 端口不可用
E_SMCBUSY	821	0x0335	串行驱动程序目前正在与另一个组态一起使用
E_SMCMODEM	822	0x0336	PC/MPI 电缆不存在连接
E_SMCNOLEG	823	0x0337	PC/MPI 电缆拒绝请求, 缺少必需的授权
E_ONLINE	896	0x0380	IOCTL 接口出现内部错误
E_LOGDEV	897	0x0381	注册表中没有逻辑设备
E_L2DRIVER	898	0x0382	注册表中缺少 L2DRIVER 条目
E_L4DRIVER	900	0x0384	注册表中缺少 L4DRIVER 条目
E_SYSERROR	1023	0x03FF	系统错误

通道特定的错误代码

EC_WATCHDOG	30000	0x7530	监视狗错误
EC_PDUERROR	30001	0x7531	PDU 不是所需的
EC_ONLERROR	30005	0x7535	装载 S7-Online-DLL 时出现故障

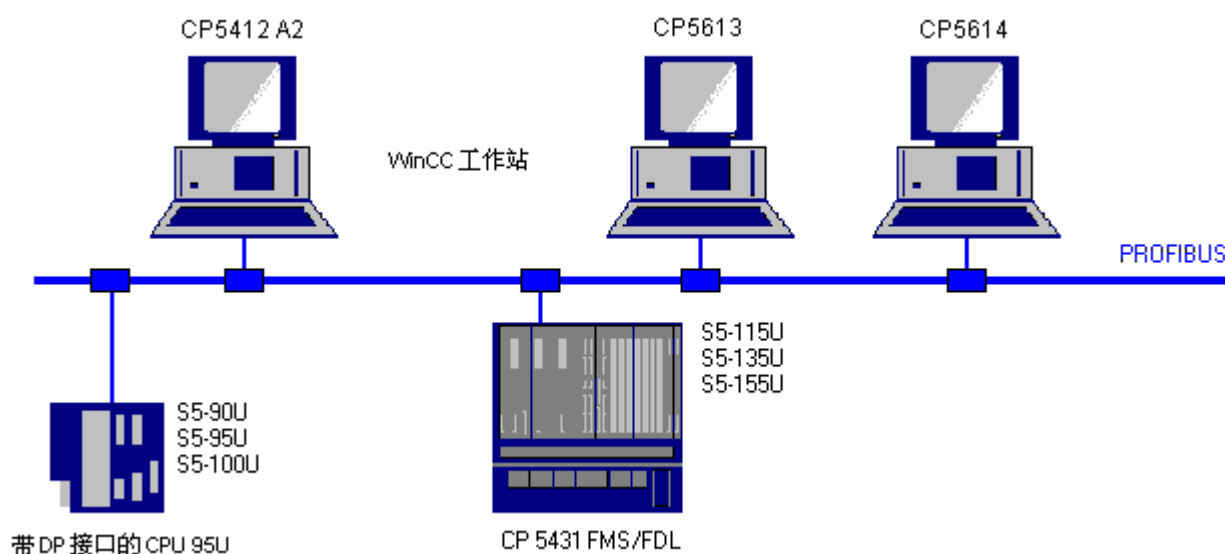
S5 PROFIBUS FDL

10.1 WinCC 通道“SIMATIC S5 Profibus FDL”

简介

通道“SIMATIC S5 Profibus FDL”用于在 WinCC 站与 SIMATIC S5 自动化系统之间进行通讯。在这种情况下，采用网络类型 PROFIBUS（过程现场总线）和协议 FDL（现场数据链接层）。

PROFIBUS 是适用于中小型数据量的网络。通过最多 127 个可能的参与者可以完成大量的自动化任务。



使用请求和响应报文来完成采用 FDL 协议、通过 PROFIBUS 进行的读/写变量操作。请求报文将被从 WinCC 发送至自动化设备。AS 以响应报文进行回答。

FDL 连接通过本地和远程连接端点（服务访问点）指定。

本章描述了

- 如何使用“SIMATIC S5 Profibus FDL”通道组态数据传送。
- 如何创建实例项目

通道单元 FDL (CP5412/A2-1)

通过通道单元“FDL (CP5412/A2-1)”可独立于所用的通讯处理器 CP 5613 或 CP 5614 与 SIMATIC S5 建立连接。此通道单元最多可支持 24 个连接。为了使通道起作用，必须设置通道单元和连接。

服务访问点

SAP 是 PROFIBUS 节点中的本地数据接口。SAP 必须在 WinCC 和 AS 中进行组态。使用“服务访问点”定义唯一标识。WinCC 和 AS 之间的通讯需要此唯一标识。

主动连接

主动连接也称为“读取”连接。主动伙伴通过此连接获取通讯伙伴的数据。从其获取数据的通讯伙伴被指定为被动伙伴。

被动连接

被动连接存在的条件是，主动 AS 以异步而没有请求报文的方式向被动 WinCC 伙伴发送数据。

10.2 受支持的数据类型和数据范围

简介

通过 PROFIBUS FDL 进行的 SIMATIC S5 通讯仅支持某些数据类型和数据范围。

所支持的数据类型

WinCC 数据类型	SIMATIC S5 数据类型
二进制变量	BIT
有符号 8 位数	SIMATIC S5 中不存在
无符号 8 位数	BYTE
有符号 16 位数	WORD
无符号 16 位数	WORD
有符号 32 位数	DWORD
无符号 32 位数	DWORD
浮点数 32 位 IEEE 754	DWORD
浮点数 64 位 IEEE 754	SIMATIC S5 中不存在
文本变量, 8 位字符集	ARRAY OF BYTE
文本变量, 16 位字符集	SIMATIC S5 中不存在
原始数据类型	ARRAY OF BYTE

访问 SIMATIC S5 变量

访问 SIMATIC S5 变量通过按字访问数据块 DB 或扩展数据块 DX 来完成。这允许读写访问。

访问 BIT 数据类型的 SIMATIC S5 变量

BIT 数据类型的 SIMATIC S5 变量只允许读访问。此限制适用于主动或被动连接。

访问 BYTE 数据类型的 SIMATIC S5 变量

BYTE 数据类型的 SIMATIC S5 变量只允许读访问。

要组态字节变量, 必须选择 16 位数据字的“左字节”或“右字节”以用于寻址。

10.2 受支持的数据类型和数据范围

访问 ARRAY OF BYTE 数据类型的 SIMATIC S5 变量

ARRAY OF BYTE 数据类型的 SIMATIC S5 变量只允许读访问。

10.3 WinCC 通道“SIMATIC S5 Profibus FDL”的功能

简介

以下部分列出了在 WinCC 中通过 PROFIBUS 的通讯驱动程序的通讯功能。同时还列出了所有受支持的数据类型以及相应的格式改编功能。

说明

写变量功能:

在 WinCC 中组态时，确保在将多个变量写入自动化系统的数据区时单独传送各个变量。

使用函数“SetTagMultiWait”将多个变量写入脚本时，此行为特别重要。由于此函数只有在将发送到该函数的所有变量传送结束时才执行，因而传送大量变量就会出现明显的等待时间。

查看在传送大量变量时是否需要使用“Wait”函数。在这种情况下，使用原始数据变量可能也是一个好方法，当 AS 数据区中的数据连续存储时尤其如此。

类型转换

如果需要某个值范围或者需要某种转换，例如，十进制转换为 BCD 码（“无符号 8 位数”转换为“ByteToBCDWord”），则需要进行格式改编。格式改编不作为标准步骤执行。

下表列出了受支持的 WinCC 数据类型以及相应的格式改编能力。

WinCC 数据类型	类型转换
二进制变量	不可以
有符号 8 位数	S5 中不存在
无符号 8 位数	可以
有符号 16 位数	可以
无符号 16 位数	可以
有符号 32 位数	可以
无符号 32 位数	可以
浮点数 32 位 IEEE 754	可以
浮点数 64 位 IEEE 754	S5 中不存在
文本变量，8 位字符集	不可以
文本变量，16 位字符集	S5 中不存在
原始数据类型	不可以

WinCC 端

通讯驱动程序 SIMATIC S5 Profibus FDL 支持使用下列通讯处理器进行通讯：

通讯处理器	总线类型
CP5613	PCI
CP5614	PCI

AS 端

通常，可以通过两种不同方式将自动化设备连接到 PROFIBUS 网络。可以通过中央模块上的集成接口进行连接，也可以使用特殊通讯模块进行连接。

系统	模块
S5-90U、S5-95U、S5-100U	CPU95U
S5-115U、S5-135U、S5-155U	CP5431 FMS/DP

说明

由于 WinCC 只能作为被动伙伴，因而使用 L2-SS 连接到 S5-95U 时，不能进行“读取”连接。

10.4 组态通道

10.4.1 如何组态通道“SIMATIC S5 Profibus FDL”

简介

本章及后文内容都介绍了组态通道“SIMATIC S5 Profibus FDL” 的步骤。

本部分介绍如何组态通道“SIMATIC S5 Profibus FDL” 。

步骤

1. 在变量管理的导航区域中，从节点“变量管理”(Tag Management) 的快捷菜单中选择条目“添加新的驱动程序”(Add new driver)。
2. 选择驱动程序“SIMATIC S5 Profibus FDL” 。 随即将创建该通道，并在变量管理器中显示该通信驱动程序。

10.4.2 通道单元“FDL (CP5412/A2-1)”

简介

通讯驱动程序“SIMATIC S5 Profibus FDL” 仅包含通道单元“FDL (CP5412/A2-1)” 。

WinCC 和 SIMATIC S5 自动化设备之间的通讯通过通道单元“FDL (CP5412/A2-1)” 进行。在一个通道单元中最多可创建 24 个连接。 需要为每个组态的连接定义特定的连接参数。必须使用变量参数定义各个组态的变量。

说明

通道单元“FDL (CP5412/A2-1)” 的名称受通讯驱动程序“SIMATIC S5 Profibus FDL.CHN” 的约束，而与所使用的通讯处理器无关。例如，对于通讯处理器，可使用 CP5613 和 CP5614。

10.4 组态通道

变量参数

必须为每个组态的变量定义下列变量参数：

- 数据区（如 DB）
- 数据块号
- 寻址（如“左字节”）
- 起始地址（如 DL 0，“左字节”已被选择用于寻址时）

连接参数

必须为每个组态的连接定义下列连接参数：

- AS 的站地址
- 优先级
- 必须为读写功能定义自身和外来的 SAP（服务访问点）

对于读功能，还需要组态是主动连接还是被动连接。对于主动读连接，则从 WinCC 站请求值。对于被动连接，则由 AS 启动向 WinCC 站进行的数值传送。

10.4.3 如何组态连接

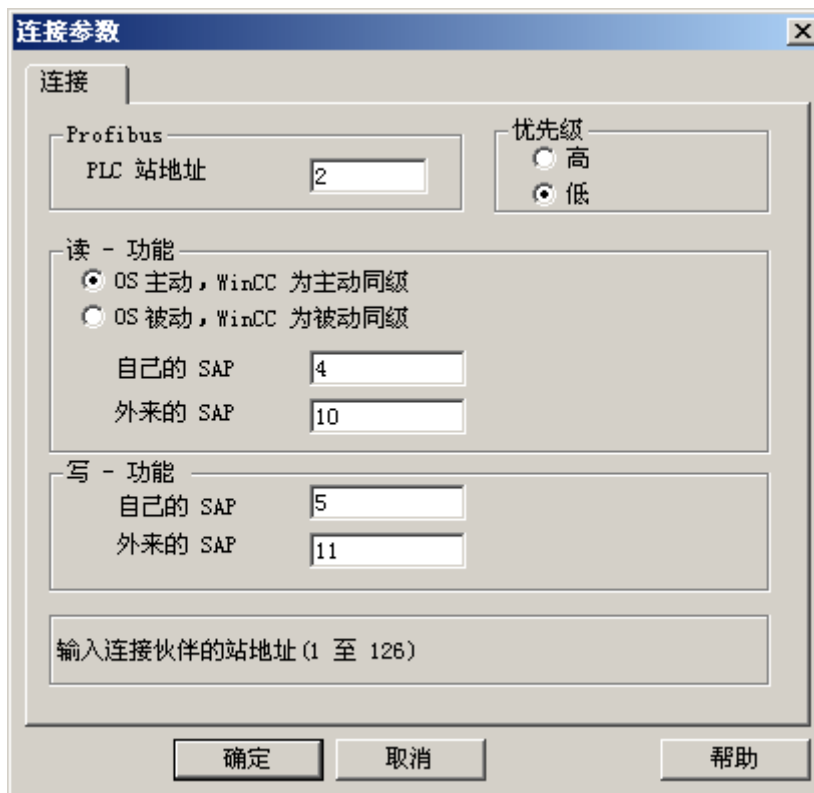
要求

- 通道“SIMATIC S5 Profibus FDL”必须集成在项目中。

步骤

1. 选择通道单元“FDL (CP5412/A2-1)”。
2. 在通道单元的快捷菜单中选择“新建连接”(New Connection) 条目。
3. 输入连接的名称。

- 从连接的快捷菜单中选择“连接参数”(Connection parameters) 条目。
“连接属性”(Connection properties) 对话框随即打开。



- 在“PLC 站地址”域中，输入 AS 的唯一地址。
- 对于 FDL 连接，“优先级”选项按钮必须始终设置为“低”。
- 使用“OS 主动，WinCC 为主动伙伴”或“OS 被动，WinCC 为被动伙伴”域定义 WinCC 站的功能。激活所需的选项按钮。
- 在“自身的 SAP”和“外来的 SAP”域中，输入为读写访问组态的 SAP 地址。SAP 值范围介于 2 和 54 之间。
- 单击“确定”(OK) 关闭所有打开的对话框。
- 从连接的快捷菜单中选择“新建变量”(New Tag) 选项。将打开“变量属性”(Tag Properties) 对话框。组态该变量。
- 单击“确定”(OK) 关闭所有打开的对话框。

10.4 组态通道

10.4.4 组态变量

10.4.4.1 组态变量

简介

对于 WinCC 与 AS 之间通过通道“SIMATIC S5 Profibus FDL”进行的连接，可在 WinCC 中定义二进制、字节和字这些数据类型。下面将说明如何组态这些数据类型的变量。

10.4.4.2 如何组态按位访问的变量

简介

本部分将介绍如何为 AS 中的地址区域组态一个按位访问的变量。

说明


按位访问变量为只读访问。

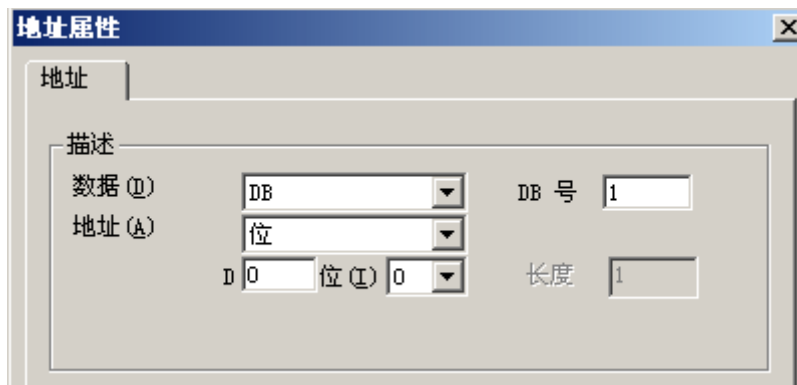
要求

1. 通道“SIMATIC S5 Profibus FDL”必须集成在项目中。
2. 必须在通道单元“FDL (CP5412/A2-1)”中定义连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data Type) 字段中，设置“二进制变量”(Binary tag) 数据类型。

5. 打开“地址属性”(Address properties)对话框。
为此,请单击“地址”(Address)字段,然后单击  按钮。



6. 在“D”域中输入字节地址,在“位”域中输入位地址。左侧域标签取决于“数据区”域中的条目,例如,如果数据区为“DB”且变量的数据类型为二进制变量,则左侧域标签为“D”。
7. 单击“确定”(OK)关闭所有打开的对话框。

说明

无法更改“寻址”域中的“位”条目,因为这是由 WinCC 变量的“二进制变量”数据类型定义的。

10.4.4.3 如何组态按字节访问的变量

简介

本节说明如何组态一个用于按字节访问 AS 中的地址区域的变量。


说明

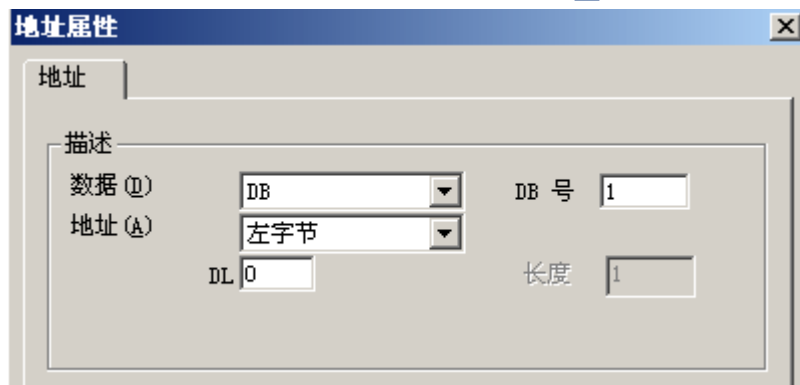
按字节访问变量只是读访问。

要求

1. 通道“SIMATIC S5 Profibus FDL”必须集成在项目中。
2. 必须在通道单元“FDL (CP5412/A2-1)”中定义连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data Type) 字段中，将数据类型设置为“无符号 8 位数”(Unsigned 8-bit value)。
5. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



6. 在“寻址”域中，可以在“左字节”和“右字节”之间进行选择。
7. 在“DL”域中，输入字节地址。域标签取决于“寻址”域中的条目，例如，对于“左字节”地址则为“DL”。
8. 单击“确定”(OK) 关闭所有打开的对话框。

10.4.4.4 如何组态按字访问的变量

简介

本节说明如何组态一个用于按字访问 AS 中的地址区域的变量。


说明

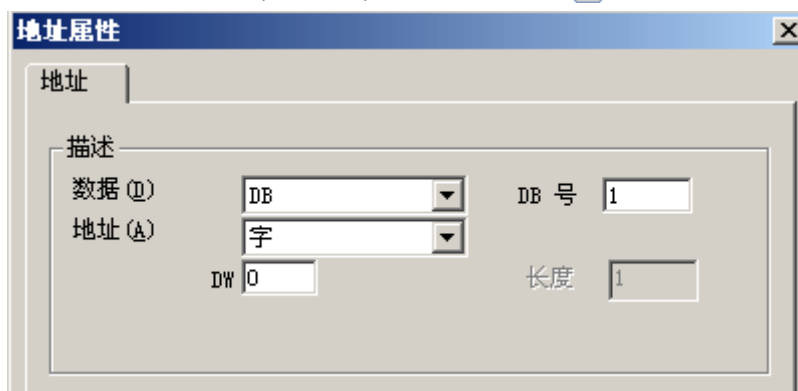
按字访问变量为读访问和/或写访问。

要求

1. 通道“SIMATIC S5 Profibus FDL” 必须集成在项目中。
2. 必须在通道单元“FDL (CP5412/A2-1)” 中定义连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data Type) 字段中，将数据类型设置为“无符号 16 位数”(Unsigned 16-bit value)。
5. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



6. 在下面的域中输入字地址。域标签取决于“寻址”域中的条目，例如，对于“字”地址则为“DW”。
7. 单击“确定”(OK) 关闭所有打开的对话框。

说明

“寻址”(Addressing) 字段中的条目“字”(Word) 无法更改，因为这已经由 WinCC 变量的数据类型“无符号 16 位数”(Unsigned 16-bit value) 定义。

10.4.5 系统参数

10.4.5.1 通道单元的系统参数

简介

如果需要与 WinCC 标准设置不一样的组态，则可使用通道单元的“系统参数”对话框进行所需的全部更改。

10.4 组态通道

可以更改以下各项：

- 设备名称
- 读/写监视时间

设备名称

WinCC 和自动化系统之间的通讯通过逻辑设备名称执行。这些名称是在安装通讯模块时分配的，并且是单元特定的条目。设备名称代表逻辑设备名称。默认情况下，此域定义为条目“/CP_L2_1:/SCP”。

读/写监视时间

读/写监视时间是 AS 写/读响应的最长等待时间（单位为秒）。如果 AS 在定义的时间内未做出任何响应，则表示连接断开了。默认情况下，为此域分配的等待时间值为 30 秒

说明

系统参数适用于 AS 中的所有 CP。

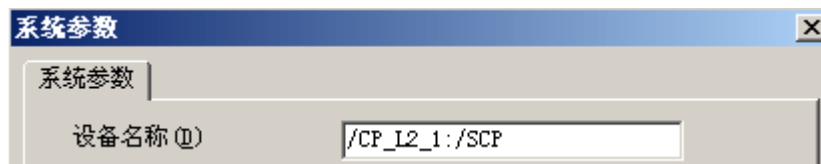
10.4.5.2 如何更改设备名称

要求

- 通道“SIMATIC S5 Profibus FDL”必须集成在项目中。

步骤

1. 在通道单元快捷菜单中，选择“系统参数”。将打开“System Parameters”（系统参数）对话框。



2. 在“设备名称”域中，输入访问点的名称。此名称必须与在 Windows 下通过“开始” → “设置” → “控制面板” → “设置 PG/PC 接口”所设置的一样。
3. 单击“确定”按钮，关闭对话框。

说明

这些更改只有在 WinCC 重启后才有效。

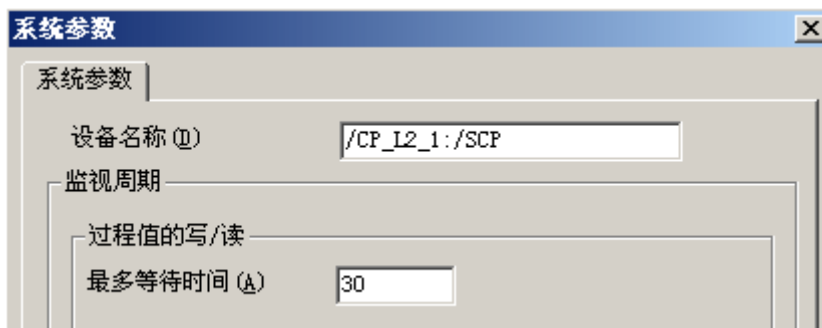
10.4.5.3 如何更改过程值的写/读监视时间

要求

- 通道“SIMATIC S5 Profibus FDL”必须集成在项目中。

步骤

1. 在通道单元快捷菜单中，选择“系统参数”。将打开“System Parameters”（系统参数）对话框。



2. 在“最多等待时间”域中，输入所需的值（单位为秒）。可以定义一个介于 1 和 3600 秒之间的值。为此域分配的默认值为 30 秒。
3. 单击“确定”按钮，关闭对话框。

说明

这些更改只有在 WinCC 重启后才有效。

10.5 特殊功能

10.5.1 “SIMATIC S5 Profibus FDL” 通道的特殊功能

简介

“SIMATIC S5 Profibus FDL” 通道具有一些特殊功能，本章将介绍这些特殊功能。

10.5.2 “SIMATIC S5 Profibus FDL” 通道的原始数据变量

10.5.2.1 “SIMATIC S5 Profibus FDL” 通道的原始数据变量

简介

“原始数据类型”的变量是一种数据报文。

原始数据变量是从 AS 传送用户数据块或者将用户数据块传送到 AS 所需的

SIMATIC S5 Profibus FDL 使用的原始数据变量最长可为 220 个字节。

原始数据变量作为字节数组

作为字节数组的原始数据变量的处理方式类似于通常的过程变量，都是通过数据块的地址和长度进行寻址（例如：DB 100，DW 20，长度为 40 个字节）。

使用脚本写入原始数据变量

如果使用 VB 脚本将长度长于 WinCC 中组态的变量长度的原始数据写入 S5，写入过程将中止。


而应通过 C 脚本使用“SetTagRaw”函数写入原始数据变量。对于此函数，需指定要写入的变量的长度。

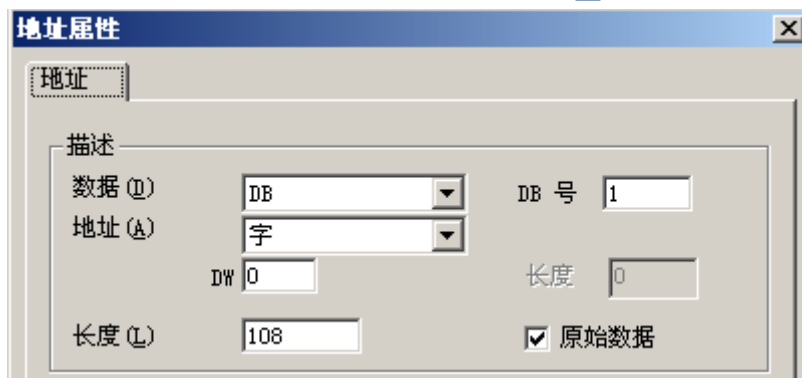
10.5.2.2 如何组态原始数据变量

要求

1. 通道“SIMATIC S5 Profibus FDL”必须集成在项目中。
2. 必须在通道单元“FDL (CP5412/A2-1)”中定义连接。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags)选项卡。
3. 单击“名称”(Name)列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data type)字段中,选择“原始数据类型”(Raw Data Type)。
5. 单击“选择”按钮以打开“地址属性”对话框。
打开“地址属性”(Address properties)对话框。
为此,请单击“地址”(Address)字段,然后单击  按钮。



6. 选中“原始数据”(Raw Data)复选框。
7. 在“长度”域中,输入原始数据块的长度(以字节为单位)。
8. 在“数据区”中,设置数据所在的 PLC 数据区。如果选择“DB”作为数据区,则在已启用的“DB 号”(DB No.)字段中输入数据块的编号。
9. 在“寻址”域中设置寻址类型。条目“左字节”、“右字节”、“字”和“双字”都可用于 WinCC 变量的数据类型“原始数据类型”。
10. 在下面的域中,输入起始地址的值。左侧域上的标签取决于“数据区”和“寻址”域中的条目,例如,数据区为“DB”、寻址类型为“字”时,显示“DW”条目。
11. 单击“确定”(OK)关闭所有打开的对话框。

10.5 特殊功能

10.5.3 组态通讯类型

10.5.3.1 组态通讯类型

简介

可以组态 FDL 连接，以便 WinCC 作为主动或被动伙伴运行。

如果将 WinCC 组态为主动伙伴，则由 WinCC 站请求数值。

如果将 WinCC 组态为被动伙伴，则由 AS 启动向 WinCC 站进行的数值传送。

10.5.3.2 如何组态主动数据传送

简介

本节说明如何组态一个向 AS 中的地址区域进行的主动数据传送。

说明

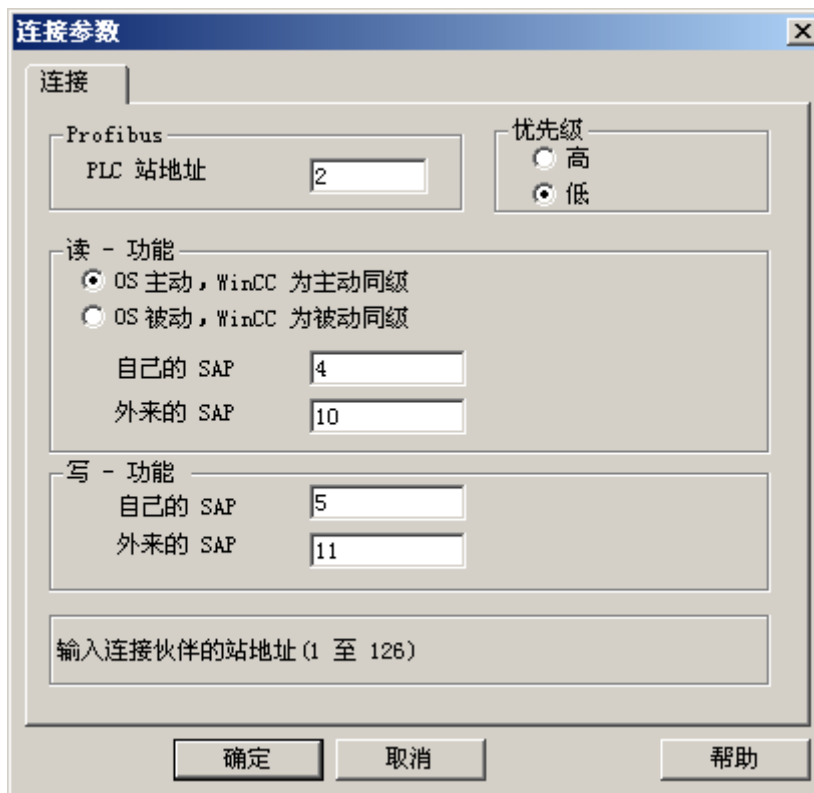
如果组态了多个连接，请注意：一个 SAP 只能被分配一次。

要求

1. 通道“SIMATIC S5 Profibus FDL”必须集成在项目中。
2. 必须在通道单元“FDL (CP5412/A2-1)”中定义连接。
3. 必须知道由 AS 定义的 SAP 地址。

步骤

1. 在连接快捷菜单中选择“连接参数”(Connection parameters) 条目。



2. 在“连接”选项卡上的“PLC 站地址”域中，输入 AS 的站地址。
3. 在“读 - 功能”区域中，必须进行下列设置：
4. 激活选项“OS 主动，WinCC 为主动伙伴”。
5. 在“自己的 SAP”域中，输入 WinCC 站的 SAP-ID。
6. 在“外来的 SAP”域中，输入 AS 的 SAP-ID。
7. 在“写 - 功能”区域中，必须进行下列设置：
8. 在“自己的 SAP”域中，输入 WinCC 站的 SAP-ID。
9. 在“外来的 SAP”域中，输入 AS 的 SAP-ID。
10. 单击“确定”(OK) 关闭所有打开的对话框。

10.5.3.3 如何组态被动数据传送

简介

本节说明如何组态一个向 AS 中的地址区域进行的被动数据传送。

说明

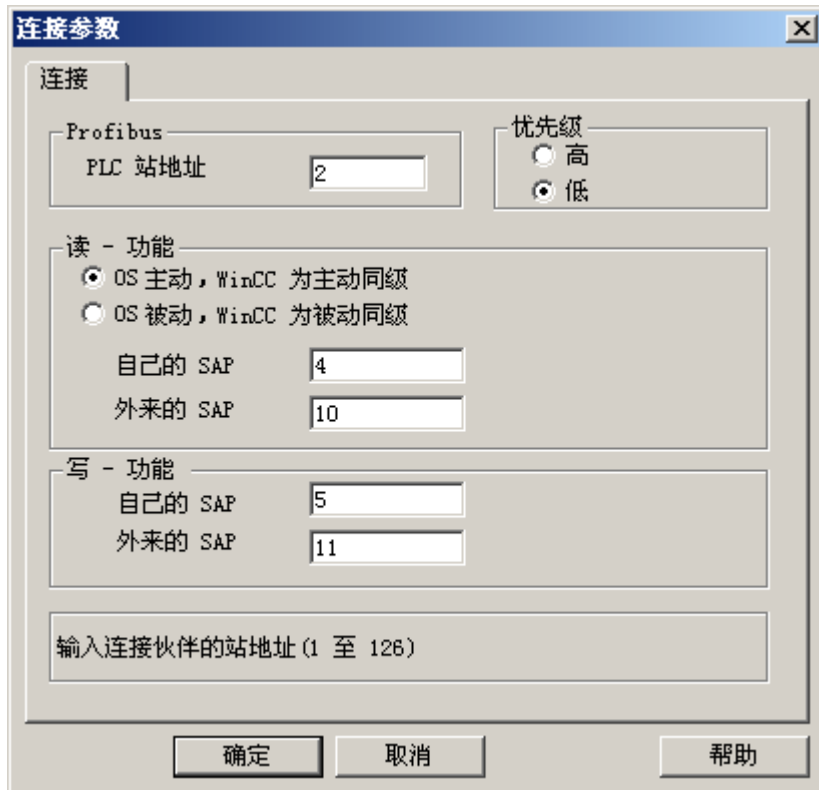
如果组态了多个连接，请注意：一个 SAP 只能被分配一次。

要求

1. 通道“SIMATIC S5 Profibus FDL”必须集成在项目中。
2. 必须在通道单元“FDL (CP5412/A2-1)”中定义连接。
3. 必须知道由 AS 定义的 SAP 地址。

步骤

1. 在连接快捷菜单中选择“连接参数”(Connection parameters) 条目。



2. 在“连接”选项卡上的“PLC 站地址”域中，输入 AS 的站地址。
3. 在“读 - 功能”区域中，必须进行下列设置：

4. 激活选项“OS 被动，WinCC 为被动伙伴”。
5. 在“自己的 SAP”域中，输入 WinCC 站的 SAP-ID。
6. 在“外来的 SAP”域中，输入 AS 的 SAP-ID。
7. 在“写 - 功能”区域中，必须进行下列设置：
8. 在“自己的 SAP”域中，输入 WinCC 站的 SAP-ID。
9. 在“外来的 SAP”域中，输入 AS 的 SAP-ID。
10. 单击“确定”(OK) 关闭所有打开的对话框。

10.6 “SIMATIC S5 Profibus FDL” 通道的组态实例

10.6.1 “SIMATIC S5 Profibus FDL” 通道的组态实例

简介

在本实例中，将在图形编辑器中组态 I/O 域，并将所需的值分配给 AS 中的数据处理块。

10.6.2 如何组态 AS 中的数据处理块

简介

在本部分，将组态 AS 中的标准功能块 OB 21 (L2ANLAUF) 和 OB 1 (L2SNDRCV)。

默认情况下，通过 PROFIBUS FDL 进行的 SIMATIC S5 连接的数据通信流量由以下块处理。

在实例中，采用以下 SAP 编号：

	WinCC	自动化设备
读功能的 SAP 编号	12	6
写功能的 SAP 编号	11	4

功能块

功能	功能块
用于启动的 OB 20、21、22	FB-L2ANLAUF (FB 9)
用于循环操作的 OB 1	FB-L2SNDRCV (FB 10)
作为两个 FB 的内部工作 DB	DB-L2DBVC3 (DB 10) DB-L2DBVC4 (DB 11) DB-L2DBVC5 (DB 12)

启动块

通讯参数在启动块中定义，注册工作 DB，并同步通讯处理器。例如，通过调用功能块 FB9 L2ANLAUF 来执行这些工步。

循环块

报文通信流量在循环 FB 中处理。接收到的报文被输入到目标数据块中。如果在此期间出现错误，就会拒绝该报文并发出错误消息。必须发送的报文将由用户指定，而指定的方式则与标准数据处理块相类似。传送完成后，会发送确认。

要求

- 数据处理块 SYNCHRON、CONTROL、SEND 和 RECEIVE 必须在自动化系统中可用。

步骤


1. 在 STEP5 中，通过选择菜单项“编辑器” → “STEP5 块” → “在程序文件中”，创建启动块（OB 20、21、22）。实例中的程序块称为“L2ANLAUF”。
2. 必须预先分配下列参数：
 - CP（例如 CP5431）的接口号 (SSNR)
 - 通讯处理器 CP 5613 的 PROFIBUS 地址 (RADR) 位于 WinCC 计算机上。该编号在网络中必须唯一。
 - 例如，所用请求类型的连接参数是参数 RVC4（用于写访问）和 RVC5（用于读访问），这些参数指定 WinCC 站的 SAP。当连接在 WinCC 中被建立时将会创建这些 SAP。
 - 在组态通讯处理器的 FDL 连接时设置的请求编号（ANR4 和 ANR5）为
 - 工作数据块的编号，即 DBX4（用于写访问）和 DBX5（用于读访问）

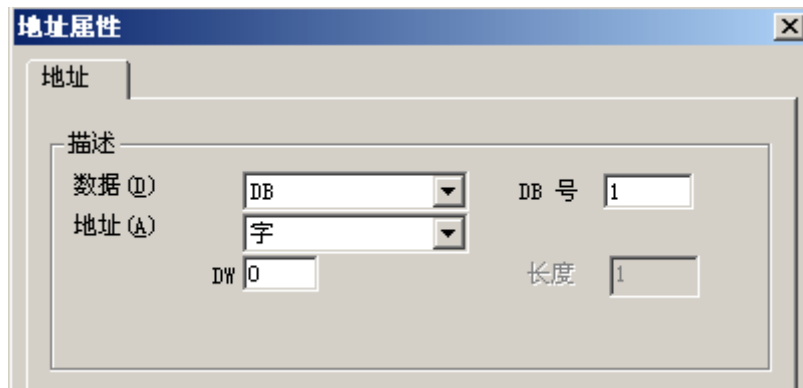
```

OB 21
NETZWERK 1 von 1          Synchronisieren CP 5431
:
:SPA FB 9
NAME :L2ANLAUF
SSNR :   KF +0           SSNR of CP 5431
TIM3 :   KT 000.0       nr
TIM7 :   KT 000.0       nr
RADR :   KF +8         PROFIBUS-Address WinCC Station
RUC3 :   KF +0         nr
RUC4 :   KF +4         SAP WRITE
RUC5 :   KF +6         SAP READ
RUC6 :   KF +0         nr
RUC7 :   KF +0         nr
ANR3 :   KF +0         nr
ANR4 :   KF +134       ANR WRITE
ANR5 :   KF +135       ANR READ
ANR6 :   KF +0         nr
ANR7 :   KF +0         nr
DBX3 :   KY 000,000    nr
DBX4 :   KY 000,011    Work-DB WRITE
DBX5 :   KY 000,012    Work-DB READ
DBX6 :   KY 000,000    nr
DBX7 :   KY 000,000    nr
S/R3 :   KF +0         nr
:
:BE

```


步骤

1. 从通道单元“FDL (CP5412/A2-1)”的快捷菜单中选择“新建连接”(New Connection) 选项，并建立名为“TestFDL”的连接。
2. 从连接的快捷菜单中选择“连接参数”(Connection parameters) 条目。
“连接属性”(Connection properties) 对话框随即打开。
在“PLC 站地址”(PLC Station Address) 字段中输入 AS 的站地址。
可以用此种方式组态 FDL 连接，使 WinCC 成为主动或被动伙伴。如果将 WinCC 组态为主动伙伴，则由 WinCC 站请求数值。如果将 WinCC 组态为被动伙伴，则由 AS 启动向 WinCC 站传送数值。
单击“确定”(OK)，关闭所有打开的对话框
3. 单击表格区下方的“变量”(Tags) 选项卡。
4. 单击“名称”(Name) 列最上方的空单元格。
输入“FDLWord1_Test”作为名称。
5. 在“数据类型”(Data Type) 字段中，将数据类型设置为“无符号 16 位数”(Unsigned 16-bit value)。
6. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



7. 在“数据区”中，设置数据所在的 PLC 数据区。如果选择“DB”作为数据区，则在已启用的“DB 号”(DB No.) 字段中输入数据块的编号。在“DW”域中，输入字节地址。
8. 单击“确定”(OK) 关闭所有打开的对话框。
9. 必须在图形编辑器中将智能对象“I/O 域”集成到画面中。
10. 将“I/O 域”与一个组态变量连接。
11. 通过单击工具栏上的“激活运行系统”按钮，或通过选择“文件”菜单中的“激活运行系统”来激活项目。组态变量的所有更改均显示在运行系统的“I/O 域”中。

S5 Programmers Port AS511

11.1 WinCC 通道“SIMATIC S5 Programmers Port AS511”

简介

通讯驱动程序“SIMATIC S5 Programmers Port AS511”用于通过 TTY 接口与 SIMATIC S5 自动化系统进行串行连接。

本章描述了

- 如何使用“SIMATIC S5 Programmers Port AS511”通道组态数据传送。
- 如何组态连接和变量。

通道单元

通讯驱动程序具有一个用于控制串行连接的 COM 端口的通道单元。

提供了以下功能：

- 通道单元 S5-AS511，用于通过“西门子特定的”协议进行串行通讯。

11.2 变量的数据类型

简介

为逻辑连接定义所需变量。从 WinCC 的角度而言，可以访问以下数据类型：

- 二进制变量
- 无符号 8 位数
- 有符号 8 位数
- 无符号 16 位数
- 有符号 16 位数
- 无符号 32 位数
- 有符号 32 位数
- 浮点数 32 位 IEEE 754
- 文本变量，8 位字符集
- 原始数据类型

11.3 组态通道

11.3.1 组态“SIMATIC S5 Programmers Port AS511”通道

简介

组态通道“SIMATIC S5 Programmers Port AS511”需要以下步骤。

- 组态连接
- 组态变量

11.3.2 如何组态连接

简介

SIMATIC S5 自动化系统可以使用串行连接来进行过程连接。自动化系统中使用 AS 511 通信处理器。

WinCC 中不需要附加通信模块。通信可通过 PG 760 上的 TTY 端口或作为系统标准设备一部分的 COM 端口加以建立。对于后者，还需要一个端口转换器 V.24/V.28 <---> TTY。这种串行连接所支持的传输率高达 19200 波特。

可以使用以下步骤将 PC 的一个串行端口分配给 AS511-NT 驱动程序。

说明

在 WinCC 与 S5 自动化系统通过“AS511”通道进行通信期间，不能在 AS 中传送、创建或删除数据块。S5 中的内存也不能进行压缩。这种终极限制是由于对 S5 中的内存进行绝对寻址而导致的。如果需要更改，则必须断开与 WinCC 的连接。

步骤

1. 在通道单元“S5-AS511” 下选择连接。
2. 在连接快捷菜单中选择 “连接参数”(Connection parameters) 条目。



3. 在“端口”域中，选择要用于串行连接的端口。

11.3.3 组态变量

11.3.3.1 组态变量

简介

对于 WinCC 与 AS 之间通过通道“SIMATIC S5 Programmers Port AS511”进行的连接，可在 WinCC 中创建不同数据类型的变量。以下部分将对此进行说明。

说明

在 WinCC 中，不检查变量地址的似真性。如果所使用的地址在 AS 中不可用，则会设置“寻址出错”状态。

在 DB 和 DX 数据块中，只能对地址 255 进行读写访问。

不能写入时间。

11.3.3.2 如何组态变量的地址


简介

变量地址根据 SIMATIC S5 的地址结构输入。

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡并选择变量。
3. 在“常规”(General) 选项卡中选择所需的数据类型（例如，有符号 8 位数）。

11.3 组态通道

- 4. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
选择“SYSVAR”选项卡。



- 5. 单击选择域，以选择该变量应传送“PLC 类型”、当前状态（“PLC 状态”）还是其它数据（“PLC 数据”）。

6. 只有当您选择了“PLC 数据”，您才会需要单击“地址”选项卡来定义变量的 S5 地址。



7. 在“数据区”域中，选择变量是位于数据块、扩展数据块、标记区、输入范围还是输出范围中。
8. 如果变量位于数据块中，则还会显示“DB 号”域。在此，输入数据块的编号。
9. 在“寻址”域中输入寻址类型。通常，可以使用默认定义。
10. 在相应域（例如“DW”）中，输入地址。

PLC 中的内存经常只能按字节或字存取。在使用二进制和“8 位”变量时，除了会打开对话框“地址属性”外，还将打开“位/字节变量”对话框，该对话框也可用于更改 PLC 内

11.3 组态通道

存中的各个位。为此，对于每个写入请求，均从 PLC 读取已赋址的存储区，从而修改相应的位和/或字节。然后，数据被写回到 PLC 的存储器。

说明

PLC 在读取数据区中执行的更改会在写回到该数据区时被覆盖。

视变量类型而定，可以按位或按字节访问自动化系统的内存。

在 WinCC 中，不检查变量地址的似真性。如果所使用的地址在 AS 中不可用，则会设置“寻址出错”状态。


在 DB 和 DX 数据块中，只能对地址 255 进行读写访问。

不能写入时间。

11.3.3.3 如何组态按位访问的变量

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 在“数据类型”(Data Type) 字段中，设置“二进制变量”(Binary tag) 数据类型。

5. 打开“位/字节变量”(Bit/Byte tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。




6. 单击“选择”(Select) 按钮。将打开“地址属性”(Address properties) 对话框。
7. 在选择域中，选择 PLC 存储器的寻址类型。
8. 在选择域中，选择要更改的位号。

11.3.3.4 如何组态按字节访问的变量

步骤

1. 选择要为其组态变量的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。

11.3 组态通道

- 在“数据类型”(Data Type) 字段中，将数据类型设置为“无符号 8 位数”(Unsigned 8-bit value) 或“有符号 8 位数”(Signed 8-bit value)。
- 打开“位/字节变量”(Bit/Byte tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



- 单击“选择”(Select) 按钮。将打开“地址属性”(Address properties) 对话框。
- 在选择域中，选择 PLC 存储器的寻址类型。
- 在选择域中，选择要更改的字节号。

S5 Serial 3964R

12.1 WinCC 通道“SIMATIC S5 Serial 3964R”

简介

通讯驱动程序“SIMATIC S5 Serial 3964R”用于使 WinCC 站与 SIMATIC S5 自动化系统之间实现串行连接。

本章描述了

- 如何使用“SIMATIC S5 Serial 3964R”通道组态数据传送。
- 如何组态连接和变量。

通道单元

通讯驱动程序具有一个用于控制串行连接的 COM 端口的通道单元。

提供了以下功能：

- 通道单元 S5-RK512 (3964R)，用于通过 3964R 或 3964 协议进行串行通讯。

12.2 变量的数据类型

简介

为逻辑连接定义所需变量。从 WinCC 的角度而言，可以访问以下数据类型：

- 二进制变量
- 无符号 8 位数
- 有符号 8 位数
- 无符号 16 位数
- 有符号 16 位数
- 无符号 32 位数
- 有符号 32 位数
- 浮点数 32 位 IEEE 754
- 文本变量，8 位字符集
- 原始数据类型

12.3 组态通道

12.3.1 组态“SIMATIC S5 Serial 3964R”通道

简介

组态通道“SIMATIC S5 Serial 3964R”需要以下步骤。

12.3.2 如何组态连接

简介

SIMATIC S5 自动化系统可以使用串行连接来进行过程连接。在自动化系统中，CPU 模块上使用通信处理器 CP 544 或另一个插入式串行端口（模块插座 SI2）。

WinCC 中不需要附加通信模块。通过系统中提供的默认 COM 端口进行通信。

这种串行连接所支持的传输率高达 19200 波特。

说明

当 SIMATIC S5 主动发送“Pseudowrite”作业类型时，消息长度不得超过 64 个字。

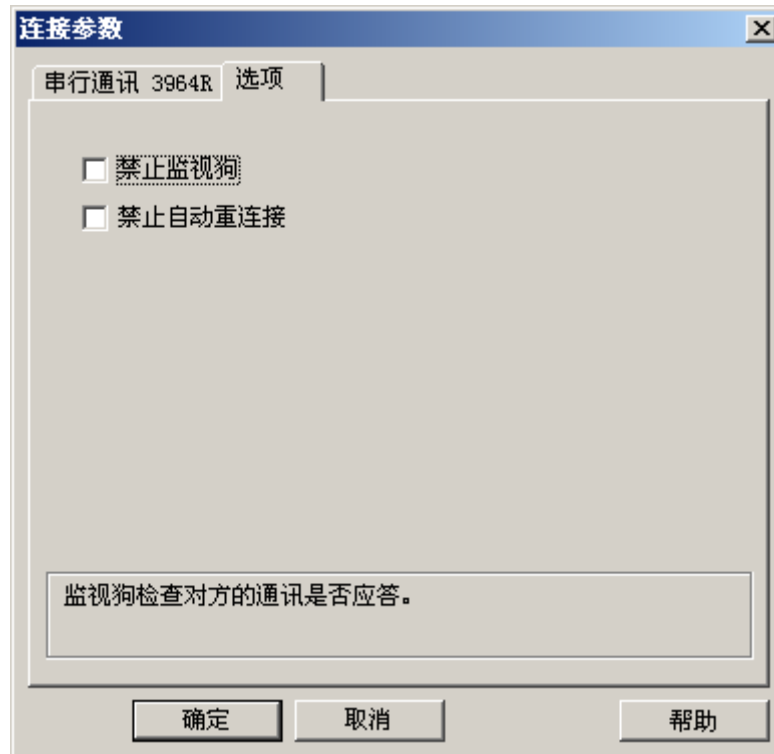
步骤

1. 选择一个连接，然后在快捷菜单中选择“连接参数”(Connection parameters)。
2. 选择“串行通信 3964R”选项卡。



3. 在“端口”域中，选择用于连接的通信端口（COM1 或 COM2）。
4. 在“程序参数”区域的“波特率”域中，将数据传送速度设置为所用的值。在“优先级”域中，设置出现启动冲突时的优先级（WinCC 和自动化系统同时发出的占线请求）。设置的优先级必须与 SIMATIC S5 中设置的优先级不同。
5. 在“程序数据”域中，选择线路协议“3964”或“3964R”。只有在出现意外情况时，才可更改程序数据（例如，确认时间、字符延迟时间等）的默认值。请确保它们与自动化系统中的参数匹配。

6. 现在，选择“选项”选项卡。



7. 在“选项”选项卡上，可以禁用设备状态循环监视，以及禁用自动重新连接。

12.3.3 组态变量

12.3.3.1 组态变量

简介


对于 WinCC 与 AS 之间通过通道“SIMATIC S5 3964R”进行的连接，可在 WinCC 中定义二进制和字节这些数据类型的变量。下面将说明如何组态这些数据类型的变量。

12.3.3.2 如何组态变量的地址

简介

变量地址根据 SIMATIC S5 的地址结构输入。

步骤

1. 选择变量，然后在“数据类型”(Data Type) 字段中设置变量所需的数据类型（例如：有符号 8 位数值）。
2. 单击“选择”(Select) 按钮。将打开“地址属性”(Address properties) 对话框。打开“地址属性”(Address properties) 对话框。为此，请单击“地址”(Address) 字段，然后单击  按钮。



3. 在“数据区”域中，选择变量是位于数据块、扩展数据块、标记区、输入范围还是输出范围中。
4. 如果变量位于数据块中，则还会显示“DB 号”域。在此，输入数据块的编号。
5. 在“寻址”域中输入寻址类型。通常，可以使用默认定义。
6. 在相应域（例如“DL”）中，输入地址。

说明

只能对输入、输出、定时器和计数器地址区域进行读访问。可以对数据块（DB、DX）进行读写访问。

不要使用值大于 255 的数据字地址。由于系统特性为 RK 512，因此只允许使用 0 到 255 之间的数据字地址。

也可以组态更大的地址，但这会导致此连接的所有组态变量数据损坏。

PLC 中的内存经常只能按字节或字存取。在使用二进制和“8 位”变量时，除了会打开对话框“地址属性”外，还将打开“位/字节变量”对话框，该对话框也可用于更改 PLC 内存中的各个位。为此，对于每个写入请求，均从 PLC 读取已赋址的存储区，从而修改相应的位和/或字节。然后，数据被写回到 PLC 的存储器。

说明

PLC 在读取数据区中执行的更改会在写回到该数据区时被覆盖。

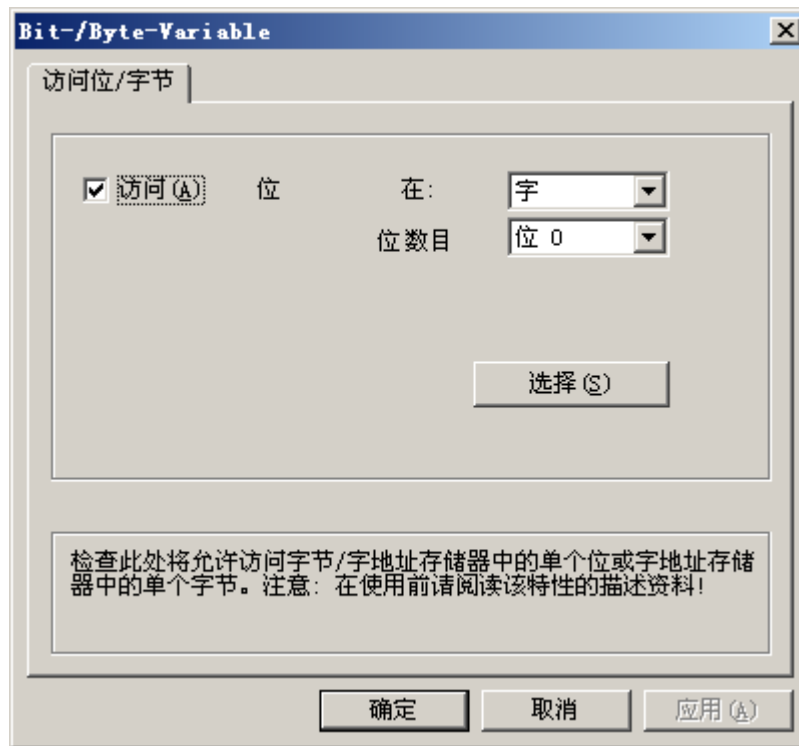
视变量类型而定，可以按位或按字节访问自动化系统的内存。

12.3.3.3 如何组态按位访问的变量

步骤

1. 选择变量，然后在“数据类型”(Data Type) 字段中设置数据类型“二进制变量”(Binary tag)。
2. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
3. 单击“选择”(Select) 按钮。“位/字节变量”(Bit/Byte tag) 对话框随即打开。

- 选择“访问位”(Access a bit) 复选框，并定义该位的地址。




- 单击“选择”(Select) 按钮。将打开“地址属性”(Address properties) 对话框。
- 在选择域中，选择 PLC 存储器的寻址类型。
- 在选择域中，选择要更改的位号。

说明

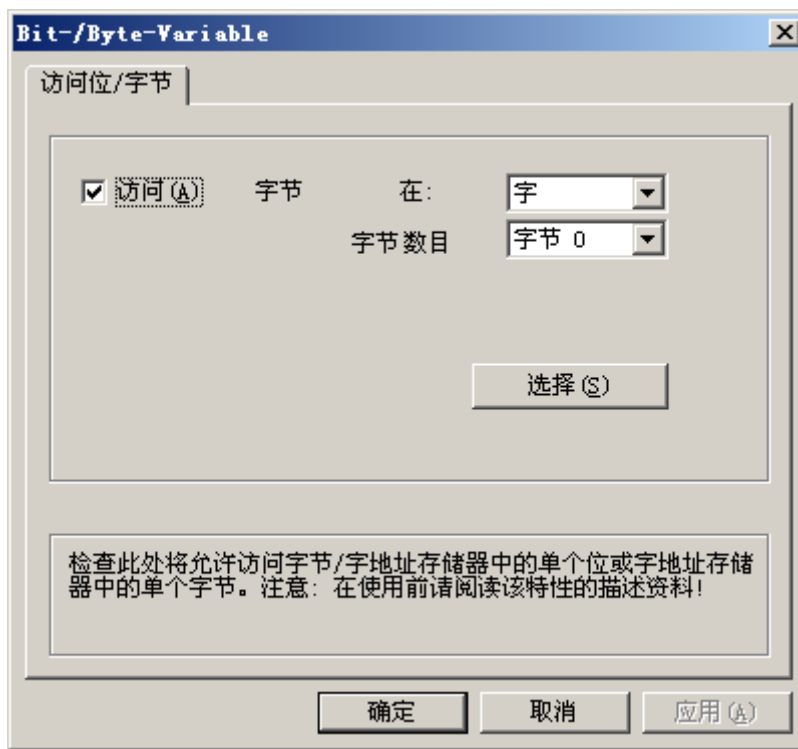
在 S5 中，可以按字节寻址标志位、输入和输出，按字寻址数据块 (DB、DX)。可以对输入、输出、定时器和计数器地址区域进行只读访问。可以对数据块 (DB、DX) 进行读写访问。

12.3.3.4 如何组态按字节访问的变量

步骤

- 选择变量，并在“数据类型”(Data Type) 字段中将数据类型设置为“无符号 8 位数”(Unsigned 8-bit value) 或“有符号 8 位数”(Signed 8-bit value)。
- 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。

- 单击“选择”(Select)按钮。“位/字节变量”(Bit/Byte tag)对话框随即打开。
- 选择“访问字节”(Access to a byte)复选框，并定义该字节的地址。



- 单击“选择”(Select)按钮。将打开“地址属性”(Address properties)对话框。
- 在选择域中，选择 PLC 存储器的寻址类型。
- 在选择域中，选择要更改的字节号。

说明

在 S5 中，可以按字节寻址标志位、输入和输出，按字寻址数据块 (DB、DX)。可以对输入、输出、定时器和计数器地址区域进行只读访问。可以对数据块 (DB、DX) 进行读写访问。

12.3 组态通道

SIMATIC S7 Protocol Suite

13.1 “SIMATIC S7 Protocol Suite” 通道

内容

“SIMATIC S7 Protocol Suite” 通道支持 WinCC 站和 SIMATIC S7 自动化系统之间的通讯。该协议集支持多种网络协议和类型。

本节说明

- 如何为通道组态各种连接和变量
- 如何创建实例项目
- 如何使用通道的特殊功能，例如 AR_SEND 功能、原始数据变量和软件冗余

13.2 WinCC 通道“SIMATIC S7 Protocol Suite”

功能原理

通道“SIMATIC S7 Protocol Suite”用于链接 SIMATIC S7-300 和 SIMATIC S7-400 自动化系统。

根据所用的通信硬件，系统支持通过下列通道单元的连接：

- 工业以太网和工业以太网 (II)：使用通信处理器（例如 CP 1612；CP1613）通过 SIMATIC NET 工业以太网进行通信。
- MPI：通过编程设备（例如 PG 760/PC RI45）的外部 MPI 端口、MPI 通信处理器或通信模块（例如 CP 5511、CP 5613）进行通信。
- 命名连接：通过符号连接与 STEP 7 进行通信。这些符号连接使用 STEP 7 组态，并且这些符号连接与 PLC S7-400 系列的 H/F 冗余系统一起，对于提供高可靠性通信是必需的。
- PROFIBUS 和 PROFIBUS (II)：使用通信处理器（例如 CP 5613）通过 SIMATIC NET PROFIBUS 进行通信。
- Slot-PLC：与 Slot PLC（例如 WinAC Pro）进行通信，这种 PLC 作为 PC 卡安装在 WinCC 计算机上。
- Soft-PLC：与 Software PLC（例如 WinAC Basis）进行通信，这种 PLC 作为应用程序安装在 WinCC 计算机上。
- TCP/IP：使用 TCP/IP 协议与网络进行通信。

有关通道和变量诊断的更多信息，请参阅“通信诊断”。

详细步骤

有关通道组态的更多信息和详细示例，请参见“WinCC V6 通信手册”：

- <http://support.automation.siemens.com/WW/view/en/21320307> (<http://support.automation.siemens.com/WW/view/zh/21320307>)

有关通道和变量诊断的更多信息，请参阅“WinCC V7.2: 通信”手册的“通信 - 诊断”：

- <http://support.automation.siemens.com/WW/view/en/73568736> (<http://support.automation.siemens.com/WW/view/zh/73568736>)

参见

软件冗余 - 连接专用的内部变量 (页 420)

“SIMATIC S7 Protocol Suite” 通道 - 组态 (页 349)

所支持的数据类型概述 (页 348)

通道单元选择 (页 344)

通道和变量的诊断 (页 507)

<http://support.automation.siemens.com/WW/view/en/21320307> (<http://support.automation.siemens.com/WW/view/zh/21320307>)

<http://support.automation.siemens.com/WW/view/en/73568736> (<http://support.automation.siemens.com/WW/view/zh/73568736>)

13.3 通道单元选择

引言

要与现有或计划网络建立通讯链接，必须进行选择：

- 某一种通道的通道单元
- 用于 WinCC 站的合适的通讯处理器
- 用于指定自动化系统的合适的通讯模块

本节提供各种选件的概述。

有两种不同类型的通讯处理器可用于 WinCC：

- 用于所谓的 **Hardnet** 的通讯处理器。它们具有自己的微处理器，可减轻系统 CPU 上的负荷。可以同时使用两种不同的协议（多协议操作）。
- 用于所谓的 **Softnet** 的通讯处理器。它们没有自己的微处理器。同一时间内只能使用一种协议（单协议操作）。

通道单元分配

下表显示了“SIMATIC S7 Protocol Suite”通道的通道单元与网络和自动化系统间的分配关系。

通道的通道单元	通讯网络	自动化系统
MPI	MPI	S7-300 和 S7-400
PROFIBUS 和 PROFIBUS (II):	PROFIBUS	S7-300 和 S7-400
工业以太网 + 工业以太网 (II)	工业以太网	S7-300 和 S7-400
TCP/IP	使用 TCP/IP 的工业以太网	S7-300 和 S7-400
命名连接	工业以太网或 PROFIBUS	S7-400 H/F 系统
Slot PLC	“Soft K-Bus”（内部）	PC（内部）
Soft PLC	“Soft K-Bus”（内部）	PC（内部）

MPI

对于通过 MPI 与 S7-300 和 S7-400 PLC 进行通讯，通道“SIMATIC S7 Protocol Suite”包括“MPI”通道单元。

从预置参数以及对用户数量和传送率方面的限制，MPI 网络很大程度上相当于 PROFIBUS 网络。MPI 通讯使用与 PROFIBUS 网络相同的通讯处理器和模块。而且也使用相同的通讯协议。

自动化系统通讯链接

S7-300 和 S7-400 可编程控制器可以使用内部 MPI 端口或合适的通讯模块通过 MPI 网络进行通讯。下表显示了建议的组件。

系统	CPU 或通讯模块（建议）
S7-300	CPU 31x CP 342-5 CP 343-5
S7-400	CPU 41x CP 443-5 Ext. CP 443-5 Basic

用于 WinCC 的通讯处理器

下表显示了建议用于将 WinCC 站连接到 MPI 网络的通讯处理器。每台 WinCC 计算机只能使用一个通讯处理器进行 MPI 通讯。每个卡还都必须有适合于各个通讯协议的驱动程序。

通讯处理器（WinCC）	配件/类型
CP 5613	PCI 卡/Hardnet
CP 5511	PCMCIA 卡/Softnet
CP 5611	PCI 卡/Softnet

PROFIBUS

对于通过 PROFIBUS 与 S7-300 和 S7-400 PLC 进行通讯，通道“SIMATIC S7 Protocol Suite”包括“PROFIBUS”和“PROFIBUS II”通道单元。

通道单元通过 Hardnet 和 Softnet 模块进行通讯。

自动化系统通讯链接

S7-300 和 S7-400 可编程控制器可以使用其内部端口或合适的通讯模块通过 PROFIBUS 网络进行通讯。下表显示了建议的组件。

系统	CPU 或通讯模块
S7-300	CPU 31x CP 342-5 CP 343-5
S7-400	CPU 41x CP 443-5 Ext. CP 443-5 Basic

用于 WinCC 的通讯处理器

下表显示了建议用于将 WinCC 站连接到 PROFIBUS 的通讯处理器。“PROFIBUS”通道单元支持通过 Hardnet 和 Softnet 卡进行通讯。WinCC 站至多支持使用两个这种的模块。每种通讯处理器都必须具有适用于相应通讯协议的驱动程序。

通讯处理器 (WinCC)	配件/类型
CP 5613	PCI 卡/Hardnet
CP 5511	PCMCIA 卡/Softnet
CP 5611	PCI 卡/Softnet

工业以太网和 TCP/IP

在 WinCC 中，“SIMATIC S7 Protocol Suite”通道支持多个通道单元通过工业以太网进行通讯：

- “工业以太网”和“工业以太网 (II)”通道单元，使用带 S7 功能的“ISO”协议
- “TCP/IP”通道单元，使用带 S7 功能的“ISO-on-TCP”协议

通道单元通过 Hardnet 和 Softnet 模块进行通讯。

自动化系统通讯模块

如果 S7-300 或 S7-400 PLC 使用“ISO”或“ISO-on-TCP”协议通过工业以太网进行通讯，则必须装有合适的通讯模块。下表显示了建议的组件。

系统	用于工业以太网的通讯模块	用于 TCP/IP 协议的通讯模块
S7-300	CP 343-1	CP 343-1 TCP
S7-400	CP 443-1	CP 443-1 TCP CP 443-1 IT

用于 WinCC 的通讯处理器

WinCC 站使用工业以太网通过“ISO”或“ISO-on-TCP”协议与下表中推荐的通讯处理器之一进行通讯。

每种通讯处理器都具有适用于每种相应通讯协议的驱动程序。

通讯处理器 (WinCC)	配件/类型
CP 1612	PCI 卡/Softnet
CP 1613	PCI 卡/Hardnet
CP 1512	PCMCIA 卡/Softnet

13.4 所支持的数据类型概述

引言

组态变量时，需要根据 AS 中的数据格式来定义数据类型和类型转换。

下表显示了通道支持的数据类型和类型转换的使用。

所支持的数据类型

数据类型	类型转换
二进制变量	否
有符号 8 位数	是
无符号 8 位数	是
有符号 16 位数	是
无符号 16 位数	是
有符号 32 位数	是
无符号 32 位数	是
浮点数 32 位 IEEE 754	是
文本变量，8 位字符集	否
原始数据类型	否

可以在“通讯”一节中找到有关类型转换的附加信息。

13.5 组态通道

13.5.1 “SIMATIC S7 Protocol Suite” 通道 - 组态

引言

本节说明如何组态“SIMATIC S7 Protocol Suite” 通道。

1. 安装通道
2. 通道单元选择
3. 组态连接
4. 变量组态

系统参数组态

有关通道、连接和变量诊断的详细信息，请参见“通讯诊断”。

参见

通道单元的系统参数 (页 372)

组态变量 (页 367)

“SIMATIC S7 Protocol Suite” 通道的通道单元 (页 350)

通道和变量的诊断 (页 507)

13.5.2 如何组态“SIMATIC S7 Protocol Suite” 通道

简介

本节说明如何安装“SIMATIC S7 Protocol Suite” 通道。


1. 安装通道
2. 通道单元选择
3. 创建连接
4. 插入一个变量
5. 在 WinCC 自定义安装中组态系统参数

13.5 组态通道

先决条件:

- 通信模块为内置模块。
- 已经安装硬件驱动程序。
- 已创建与 AS 的电缆连接。

步骤

1. 在变量管理的导航区域中，从节点“变量管理”(Tag Management) 的快捷菜单中选择条目“添加新的驱动程序”(Add new driver)。
2. 选择驱动程序“SIMATIC S7 Protocol Suite”。随后将建立该通道并在变量管理中显示通信驱动程序和通道单元。
3. 选择所需的通道单元，并在快捷菜单中选择条目“新建连接”(New Connection)。
4. 输入连接的名称。
5. 单击表格区下方的“变量”(Tags) 选项卡。
6. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
7. 在“数据类型”(Data type) 字段中选择所需的数据类型。
也可以在“属性”(Properties) 区域中为变量定义起始值和替换值。
如果要了解连接特定通道单元所需的变量组态的详细描述，请关闭对话框，然后继续参阅相应通道单元中的主题“组态变量”。
8. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
9. 单击“确定”(OK) 按钮，关闭上述两个对话框。
10. 如果 WinCC 系统和通信硬件为非标准安装，则还需要将系统参数设置为非标准值。关于此主题的其他信息请参见“系统参数”。

13.5.3 通道单元

13.5.3.1 “SIMATIC S7 Protocol Suite” 通道的通道单元

引言

以下章节描述了如何组态通道单元以及相应的连接。在同一通道单元中可能存在多个连接。

参见

“TCP/IP” 通道单元 (页 364)

“Soft PLC” 通道单元 (页 363)

“Slot PLC” 通道单元 (页 361)

通道单元“PROFIBUS (I + II)” (页 359)

“命名连接” 通道单元 (页 356)

“MPI” 通道单元 (页 354)

“工业以太网”+“工业以太网 (II)” 通道单元 (页 351)

13.5.3.2 “工业以太网 (I+II)” 通道单元

“工业以太网”+“工业以太网 (II)” 通道单元

工作原理

通道单元“工业以太网”用于通过工业以太网将 WinCC 连接到 S7 自动化系统。可以通过通讯模块 (CP) 进行通讯, 例如, 通过 CP 343-1 实现自动化系统 S7-300 通讯, 通过 CP 443-1 实现自动化系统 S7-400 通讯。

在 WinCC 中可以使用不同的通讯处理器, 例如 CP 1613。可以通过通道单元“工业以太网 (II)”对第二个通讯处理器进行寻址。因为通讯是通过“ISO”传输协议进行的, 因此不需在本地数据库中组态逻辑连接。

与这些通道单元有关的功能和组态是相同的。

典型单元术语

通讯处理器

通讯处理器 (CP) 是支持 WinCC 计算机和特定网络之间通讯的模块。

“ISO” 传输协议

ISO 传输层是 ISO-OSI 参考模型中的一层, 提供与数据传送相关的面向连接的服务。传输层处理数据流控制、阻塞和确认任务。

协议定义与物理线上内容相关的数据流的结构。它定义许多部分, 包括操作模式、建立连接的过程、数据备份或传输速度。

工业以太网

工业以太网是工业环境中最有效的一种子网。它适用于工厂和车间一级，有利于大量成员在大范围内的大数据量的交换。

工业以太网是一种开放式的通讯网络，且符合 IEEE 802.3 标准。其主要优点不仅在于其高可靠性、使用范围广，而且速度快、易扩展和开放性。组态过程十分简便。

参见

组态变量 (页 367)

如何组态“工业以太网”通道单元连接 (页 352)

如何组态“工业以太网”通道单元连接

简介

除了通道单元以外，WinCC 也需要逻辑连接来与 PLC 进行通信。所有特定的参数在建立逻辑连接时定义。

S7 自动化系统中使用通信模块进行通信，例如在 S7-300 中使用 CP 343-1，或者在 S7-400 中使用 CP 443-1。

在 WinCC 中使用通信处理器，例如 CP 1613。可以通过“工业以太网 II”通道单元寻址/增加第二个通信处理器。

有关通道、连接和变量诊断的详细信息，请参见“通信诊断”。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。

步骤

1. 在通道单元“工业以太网”(Industrial Ethernet) 的快捷菜单中选择“新建连接”(New Connection) 条目。
将创建一个新连接。
2. 输入一个连接名称，例如“Test_Ind_Eth”。

- 在快捷菜单中选择“连接参数”(Connection parameters) 连接。
将打开“连接属性 - 工业以太网”(Connection properties - Industrial Ethernet) 对话框。



- 在“以太网地址”域中输入总线上自动化系统的站地址。
- 在“机架号”域中输入要寻址 CPU 的机架号。
- 必须将指定机架中的 CPU 插槽号输入到相应的“插槽号”域中。
- 如果想要通过连接使用 BSEND/BRCV 传送数据块，激活复选框“发送/接收原始数据块”。如果激活了复选框，则可以编辑“连接资源”域。为连接源输入十六进制数值。在 PLC 中组态连接时，将由 STEP7 分配该连接资源。
- 单击“确定”(OK) 按钮，关闭上述两个对话框。

说明

当使用带外部通信模块的 S7-300 或 S7-400 时，必须输入 CPU 的机架/插槽号。
如果输错了机架或插槽号，将不会建立通信链路！

参见

组态变量 (页 367)

如何组态“SIMATIC S7 Protocol Suite” 通道 (页 349)

通道和变量的诊断 (页 507)

13.5.3.3 “MPI” 通道单元

“MPI” 通道单元

工作原理

“MPI” 通道单元用于通过 MPI 将 WinCC 连接到 SIMATIC S7-300 和 S7-400 自动化系统。

在 WinCC 中，这可以通过下列设备完成

- 编程设备（例如 PG 760/PC RI45）的内部 MPI 接口
- 通讯处理器，例如 CP 5613（PCI 卡）

所谓的 MPI 模块（ISA 卡）也适用 - 有这种模块，但难以获取。它已被通讯处理器取代。

在 PLC 中，通过 CPU 的 MPI 接口或相应的通信模块完成链接。

典型单元术语

MPI

MPI 即 Multi Point Interface（多点接口），是一种可供多个成员共用的通讯链接。与通讯网络的连接方式如下：

- 在 PLC 中，通过 CPU 的 MPI 接口或通讯模块，
- 在 WinCC 中，通过编程设备或通讯处理器（网卡）等的内置 MPI 接口。

通讯处理器

通讯处理器（CP）是支持 WinCC 计算机和特定网络之间通讯的模块。

参见

组态变量 (页 367)

如何组态“MPI” 通道单元连接 (页 354)

如何组态“MPI” 通道单元连接

简介

除了通道单元以外，WinCC 也需要逻辑连接来与 PLC 进行通信。所有特定的参数在建立逻辑连接时定义。

S7-300 和 S7-400 PLC 使用内部 MPI 接口或通信模块，例如 CP 342-5（SIMATIC S7-300）或 CP 443-5（SIMATIC S7-400）。

如果 PG 760/PC RI45 上安装了 WinCC，则可使用内部 MPI 接口；否则需要内置的 MPI 模块。或者，还可以使用通信模块，例如 CP 5511（PCMCIA 卡）。

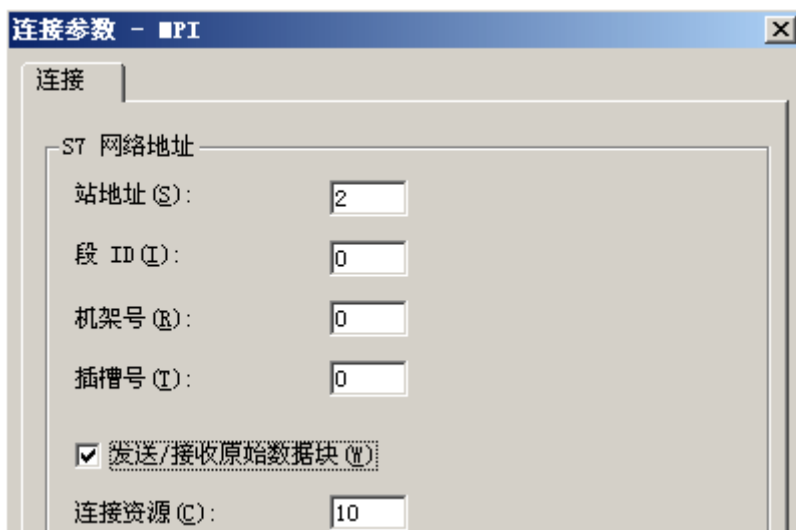
有关通道、连接和变量诊断的详细信息，请参见“通信诊断”。

要求

- 必须将“SIMATIC S7 Protocol Suite”（通信）驱动程序集成到项目中。

步骤

1. 在通道单元“MPI”的快捷菜单中选择“新建连接”(New Connection) 条目。将创建一个新连接。
2. 输入连接名称“Test_MPI”。
3. 从快捷菜单选择“连接参数”(Connection parameters) 命令。将打开“连接参数 - MPI”(Connection parameters - MPI) 对话框。



4. 在“站地址”域中输入总线上相应域中自动化系统的站地址。
5. 当前不支持“段 ID”域。数值必须保持为“0”。
6. 在“机架号”域中输入要寻址 CPU 的机架号。
7. 输入指定机架上 CPU 的“插槽号”。
8. 如果想要通过连接使用 BSEND/BRCV 传送数据块，激活复选框“发送/接收原始数据块”。如果选中该复选框，“连接资源”字段也将激活。为连接源输入十六进制数值。在 PLC 中组态连接时，将由 STEP7 分配该连接资源。
9. 单击“确定”(OK) 按钮，关闭上述两个对话框。

说明

当使用带外部通信处理器的 S7-300 或 S7-400 时，必须输入 CPU 的机架/插槽号。

如果输错了机架或插槽号，将不会建立通信链路！

使用 S7-300 时，对于通过 CPU 内部 MPI 接口的链接，必须给定机架/插槽号 = 0。

参见

组态变量 (页 367)

如何组态“SIMATIC S7 Protocol Suite”通道 (页 349)

13.5.3.4 “命名连接”通道单元**“命名连接”通道单元****工作原理**

该通道单元用于设置使用 STEP 7 组态的符号连接。从而 WinCC 可以通过符号连接名称访问冗余和非冗余连接。这些符号连接与 S7-400 PLC 系列 H/F 冗余系统一起，对于保证可靠性通讯是必需的。

符号连接名称在 STEP 7 中使用 NETPRO 应用程序组态。连接名称、连接参数和应用程序名称存储在数据库 (*.XDB) 中。该数据库由相应 WinCC 项目目录中的 PLC/OS 工程工具“Mapper”自动存储；但是，也可以在此目录外复制该数据库，例如在不使用“Mapper”时。

说明

WinCC 系统中的每位通讯成员只能有一个 XDB 文件。因此，不能在多台 WinCC 计算机上复制和使用 XDB 文件。

可使用下列选项在 WinCC 中激活此数据库：

- 如果 XDB 文件位于项目目录外（例如由于未使用 Mapper 工具），则在启动 WinCC 前，需要在“设置 PG/PC 接口”的“STEP 7 组态”标签（控制面板）中输入 XDB 文件的路径和名称。

启动 WinCC 时，如果项目目录中没有任何文件，将从外部目录中读取该 XDB 文件。当多个项目要使用同一个中央保存的数据库时，该过程十分有用。

- 如果使用了 Mapper 工具，将自动把 XDB 文件复制到 WinCC 项目目录中。启动 WinCC 并打开项目时，将从 S7 通道读取数据，并将数据输入 Windows 的注册数据库。

然后，可通过将一个符号连接名称指定给所选的应用程序名称，以在 WinCC 中组态连接。

说明

也可以手动输入应用程序和连接名称，例如，如果不存在符号连接名称的 XDB 文件或要将项目传送到其它计算机的情况。必须检查 STEP 7 中的名称拼写是否正确，因为在 CS 模式中不检查名称。

典型单元术语

通讯处理器

通讯处理器（CP）是支持 PLC 和指定网络之间通讯的模块。

参见

组态变量 (页 367)

如何组态“命名连接”通道单元连接 (页 357)

如何组态“命名连接”通道单元连接

简介

除通道单元外，WinCC 还需要逻辑连接，以便通过符号连接与 S7-400 PLC 进行通信。

设置逻辑连接时，“连接名称”域中列出的某个符号连接名称将分配给所选的应用程序名称。

符号连接名称和应用程序名称在 STEP 7 中组态。

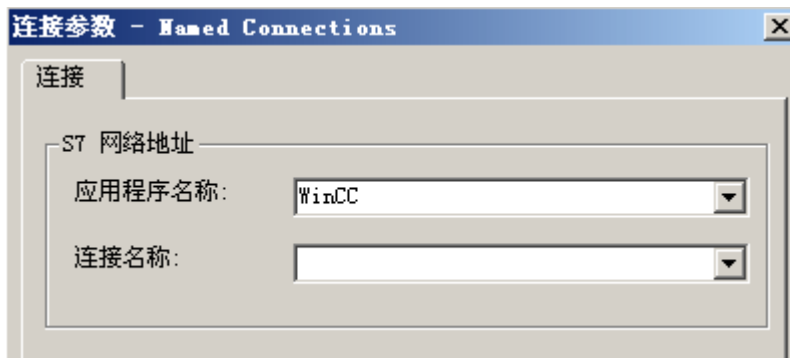
有关通道、连接和变量诊断的详细信息，请参见“通信诊断”。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。

步骤

1. 在通道单元“已命名连接”(Named Connections) 的快捷菜单中选择“新建连接”(New Connection) 条目。
将创建一个新连接。
2. 输入一个连接名称，例如“Test_NC”。
3. 从快捷菜单选择“连接参数”(Connection parameters) 命令。
将打开“连接参数 - 已命名连接”(Connection parameters - Named Connections) 对话框。



4. 在“应用程序名称”域中，输入 STEP 7 中组态的应用程序名称。缺省值为 WinCC。
5. 在“连接名称”域中，输入 STEP 7 中组态的符号连接名称。
6. 单击“确定”(OK) 按钮，关闭上述两个对话框。

说明

也可以手动输入应用程序和连接名称，例如，如果不存在符号连接名称的 XDB 文件或要将项目传送到其它计算机的情况。必须检查 STEP 7 中的名称拼写是否正确，因为在 CS 模式中不检查名称。

参见

组态变量 (页 367)

如何组态“SIMATIC S7 Protocol Suite”通道 (页 349)

13.5.3.5 “PROFIBUS (I+II)” 通道单元

通道单元“PROFIBUS (I + II)”

工作原理

通道单元用于通过 PROFIBUS 网络将 WinCC 连接到 SIMATIC S7-300 和 S7-400 自动化系统。

对于 S7 自动化系统，使用通讯模块，例如在 S7-300 中使用 CP 342-5，或者在 S7-400 中使用 CP 443-5。

在 WinCC 中使用通讯处理器，例如 CP 5613。

可以通过“PROFIBUS II”通道单元连接第二个通讯处理器。这样，可增大最大连接数。

典型单元术语

PROFIBUS

PROFIBUS 是基于车间和现场一级的开放式、非独占的通讯系统，最多可有 127 个成员站。PROFIBUS 基于欧洲标准 EN 50170，卷 2，PROFIBUS。PROFIBUS 使用令牌传递和主/从访问机制。

通讯处理器

通讯处理器（CP）是支持 WinCC 计算机和特定网络之间通讯的模块。

参见

组态变量 (页 367)

如何组态“PROFIBUS”通道单元连接 (页 359)

如何组态“PROFIBUS”通道单元连接

简介

除了通道单元，WinCC 还必须具有逻辑连接来与 PLC 进行通信。所有特定的参数在建立逻辑连接时定义。

对于 S7 自动化系统，使用通信模块，例如，在 S7-300 中使用 CP 342-5，或者在 S7-400 中使用 CP 443-5。

13.5 组态通道

在 WinCC 中使用通信处理器，例如 CP 5613。可以通过“PROFIBUS II”通道单元连接第二个通信处理器。

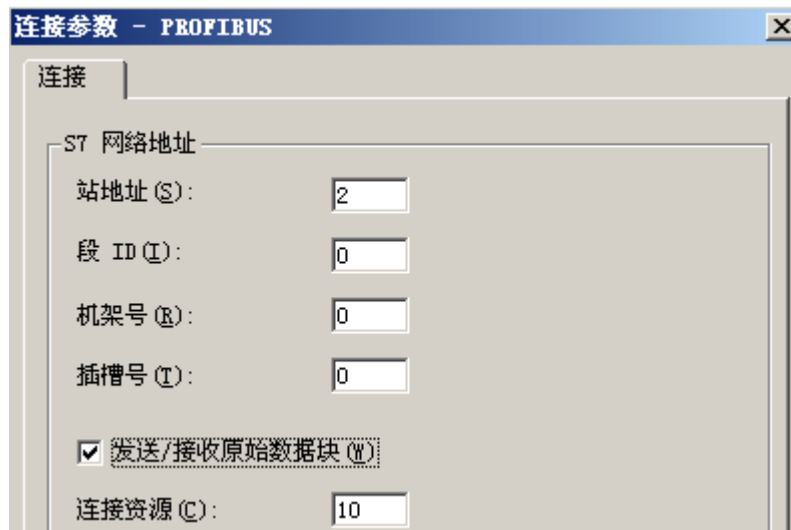
有关通道、连接和变量诊断的详细信息，请参见“通信诊断”。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。

步骤

1. 在通道单元“PROFIBUS”的快捷菜单中选择“新建连接”(New Connection) 条目。将创建一个新连接。
2. 输入连接名称“Test_PROFIBUS”。
3. 从快捷菜单选择“连接参数”(Connection parameters) 命令。将打开“连接参数 - PROFIBUS”(Connection parameters - PROFIBUS) 对话框。



4. 在相应“站地址”域输入总线上自动化系统的站地址。
5. 当前不支持“段 ID”域。数值必须保持为“0”。
6. 输入要寻址的 CPU 其中的“机架号”。
7. 输入指定机架上 CPU 的“插槽号”。
8. 如果想要通过连接使用 BSEND/BRCV 传送数据块，激活复选框“发送/接收原始数据块”。如果选中该复选框，“连接资源”字段也将激活。为连接源输入十六进制数值。在 PLC 中组态连接时，将由 STEP7 分配该连接资源。
9. 单击“确定”(OK) 按钮，关闭上述两个对话框。

说明

启动 PROFIBUS 通信时，如果在 WinCC 计算机打开时将通信处理器连接到 PROFIBUS，可能产生 PROFIBUS 错误。因此，建议计算机连接到 PROFIBUS 之前将其关闭。否则，（按照 PROFIBUS 标准）总线上可能会产生多个令牌，这些令牌将引起总线错误。

当使用带外部通信模块的 S7-300 或 S7-400 时，必须输入 CPU 的机架/插槽号。
如果输错了机架或插槽号，将不会建立通信链路！

参见

组态变量 (页 367)

如何组态“SIMATIC S7 Protocol Suite”通道 (页 349)

13.5.3.6 “Slot PLC”通道单元

“Slot PLC”通道单元

工作原理

“Slot PLC”通道单元用于实现 WinCC 和安装在 WinCC 计算机中的最多 4 个 Slot PLC (WinAC Pro) 之间的通讯。因为 Slot PLC 具有集成接口，因此 WinCC 和 Slot PLC 之间的连接不需要任何附加的通讯硬件。

参见

组态变量 (页 367)

如何组态“Slot PLC”通道单元连接 (页 361)

如何组态“Slot PLC”通道单元连接

简介

为了与所安装的 SPS 卡进行通信，除通道单元外，WinCC 还需要一个逻辑连接。所有特定的参数在建立逻辑连接时定义。

13.5 组态通道

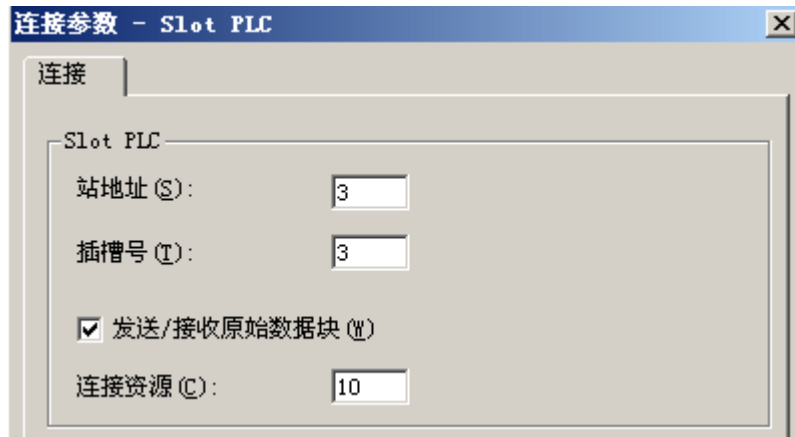
有关通道、连接和变量诊断的详细信息，请参见“通信诊断”。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 如果要组态多个 Slot PLC，则需要使用 Slot PLC V3.4 版本。

步骤

1. 在通道单元“插槽 PLC”(Slot PLC)的快捷菜单中选择“新建连接”(New connection) 条目。将创建一个新连接。
2. 输入一个连接名称，例如“Test_SPLC”。
3. 从快捷菜单选择“连接参数”(Connection parameters) 命令。
将打开“连接参数 - 插槽 PLC”(Connection parameters - Slot PLC) 对话框。



4. 在“站地址”域中，输入 Soft K 总线上 Slot PLC 的站地址。
5. 在“插槽号”域中，输入 Slot PLC 安装在其中的插槽的编号。
6. 如果想要通过连接使用 BSEND/BRCV 传送数据块，激活复选框“发送/接收原始数据块”。
7. 如果选中该复选框，“连接资源”字段也将激活。为连接源输入十六进制数值。在 PLC 内组态连接时，将由 STEP7 分配该连接资源。
8. 单击“确定”(OK) 按钮，关闭上述两个对话框。

说明

所安装的多个 Slot PLC 的“站地址”和“插槽号”连接参数必须完全相同，且“插槽号”从“3”开始编号。

参见

组态变量 (页 367)

如何组态“SIMATIC S7 Protocol Suite” 通道 (页 349)

13.5.3.7 “Soft PLC” 通道单元

“Soft PLC” 通道单元

工作原理

“Slot PLC” 通道单元用于实现 WinCC 和安装在 WinCC 计算机中的一个 Soft PLC (WinAC Pro) 之间的通讯。对于 WinCC 和 Soft PLC 之间的连接，不需要任何附加的通讯硬件。

参见

组态变量 (页 367)

如何组态“Soft PLC” 通道单元上的连接 (页 363)

如何组态“Soft PLC” 通道单元上的连接

简介

除了通道单元，WinCC 还必须具有逻辑连接来与 Soft PLC 进行通信。所有特定的参数在建立逻辑连接时定义。

有关通道、连接和变量诊断的详细信息，请参见“通信诊断”。

要求

- 通道“SIMATIC S7 Protocol Suite” 必须集成在项目中。

步骤

1. 在通道单元“Soft PLC” 的快捷菜单中选择“新建连接”(New connection) 条目。
将创建一个新连接。
2. 输入一个连接名称，例如“Test_SOFTPLC”。

13.5 组态通道

- 从快捷菜单选择“连接参数”(Connection parameters) 命令。
将打开“连接参数 - 插槽 PLC”(Connection parameters - Slot PLC) 对话框。



- 在“站地址”域中，输入 Soft K 总线上 Soft PLC 的站地址。
- 在域“插槽号”中，输入插槽的编号。插槽号在 Soft PLC 的硬件配置期间组态；并且在相同 WinCC 计算机中使用多个 Soft PLC 时需要插槽号。
- 如果想要通过连接使用 BSEND/BRCV 传送数据块，激活复选框“发送/接收原始数据块”。
- 如果选中该复选框，“连接资源”字段也将激活。为连接源输入十六进制数值。在 PLC 内组态连接时，将由 STEP7 分配该连接资源。
- 单击“确定”(OK) 按钮，关闭上述两个对话框。

参见

组态变量 (页 367)

如何组态“SIMATIC S7 Protocol Suite” 通道 (页 349)

13.5.3.8 “TCP/IP” 通道单元

“TCP/IP” 通道单元

工作原理

通道单元“TCP/IP”使用“ISO-on-TCP 传输”协议通过工业以太网将 WinCC 连接到自动化系统 SIMATIC S7-300 和 S7-400。

协议对应于标准 TCP/IP 扩展协议 RFC 1006。该扩展是必需的，因为 TCP/IP 使用其中没有任何数据阻塞的通讯。

如果是自动化系统 S7-300，通讯通过通讯模块 CP 343-1 TCP 等进行，如果是 S7-400，通过 CP 443-1 TCP 或 CP 443-1 IT 进行。

在 WinCC 中，使用 CP 1613 等通讯处理器。

因为通讯通过 ISO-on-TCP 传输协议进行，因此不必在本地数据库中组态逻辑连接。

典型单元术语

通讯处理器

通讯处理器（CP）是支持 WinCC 计算机和特定网络之间通讯的模块。

ISO 传输协议

ISO 传输层是 ISO-OSI 参考模型中的一层，提供与数据传送相关的面向连接的服务。传输层处理数据流控制、阻塞和确认任务。

协议定义与物理线上内容相关的数据流的结构。它定义许多部分，包括操作模式、建立连接的过程、数据备份或传输速度。

工业以太网

工业以太网是工业环境中最有效的一种子网。它适用于工厂和车间一级，有利于大量成员在大范围内的大数据量的交换。

工业以太网是一种开放式的通讯网络，且符合 IEEE 802.3 标准。其主要优点不仅在于其高可靠性、使用范围广，而且速度快、易扩展和开放性。组态过程十分简便。

参见

组态变量 (页 367)

如何组态“TCP/IP”通道单元连接 (页 365)

如何组态“TCP/IP”通道单元连接

简介

除了通道单元以外，WinCC 也需要逻辑连接来与 PLC 进行通信。所有特定的参数在建立逻辑连接时定义。

如果是自动化系统 S7-300，通信通过通信模块 CP 343-1 TCP 等进行，如果是 S7-400，通过 CP 443-1 TCP 或 CP 443-1 IT 进行。

在 WinCC 中，使用 CP 1613 等通信处理器。

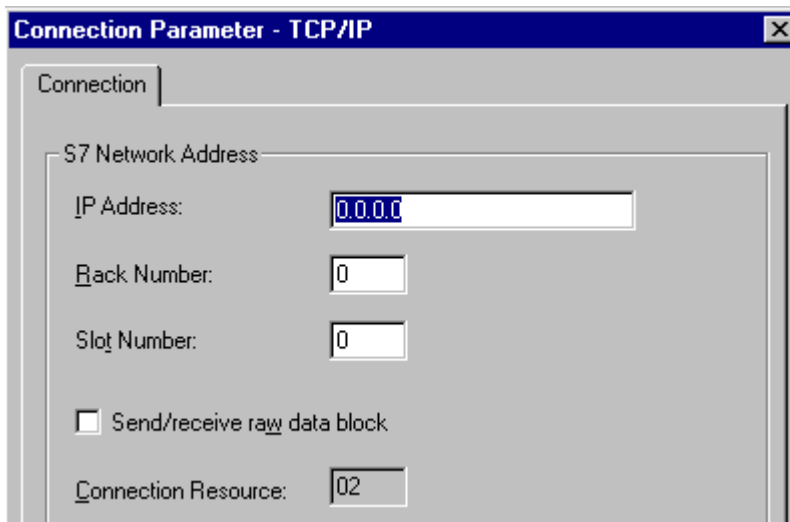
有关通道、连接和变量诊断的详细信息，请参见“通信诊断”。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。

步骤

1. 在通道单元“TCP/IP”的快捷菜单中选择条目“新建连接”(New Connection)。将创建一个新连接。
2. 输入连接名称“Test_TCP”。
3. 从快捷菜单选择“连接参数”(Connection parameters)命令。将打开“连接参数 - TCP/IP”(Connection parameters - TCP/IP)对话框。



4. 在“IP 地址”域中输入总线上自动化系统的 Internet 协议地址。
5. 在“机架号”域中输入要寻址 CPU 的机架号。
6. 必须将指定机架中的 CPU 插槽号输入到相应的“插槽号”域中。
7. 如果想要通过连接使用 BSEND/BRCV 传送数据块，激活复选框“发送/接收原始数据块”。如果选中该复选框，“连接资源”字段也将激活。为连接源输入十六进制数值。在 PLC 中组态连接时，将由 STEP7 分配该连接资源。
8. 单击“确定”(OK)按钮，关闭上述两个对话框。

说明

当使用带外部通信模块的 S7-300 或 S7-400 时，必须输入 CPU 的机架/插槽号。
如果输错了机架或插槽号，将不会建立通信链路！

参见

组态变量 (页 367)

如何组态“SIMATIC S7 Protocol Suite”通道 (页 349)

13.5.4 组态变量

13.5.4.1 组态变量

引言

以下章节描述如何组态变量。根据对 PLC 中数据区域的访问类型和 WinCC 变量的数据类型，组态有所不同。

有关通道、连接和变量诊断的详细信息，请参见“通讯诊断”。

参见

如何组态文本变量 (页 371)

如何组态按字访问的变量 (页 370)

如何组态按字节访问的变量 (页 368)

如何组态按位访问的变量 (页 367)

13.5.4.2 如何组态按位访问的变量


简介

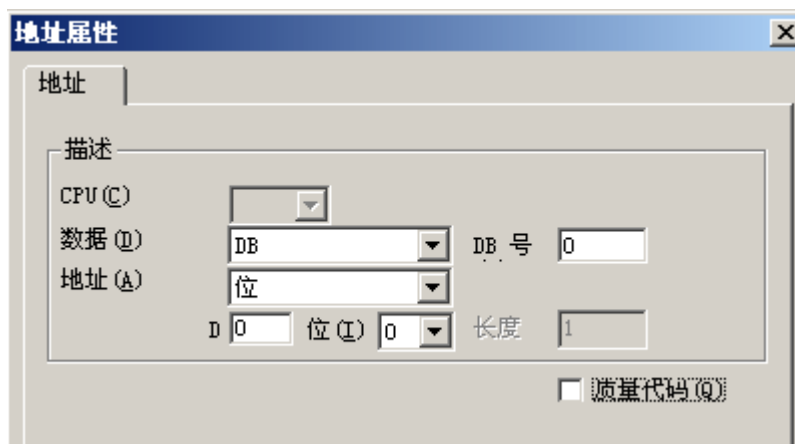
本节说明如何以按位访问方式组态 PLC 中地址区域的变量。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 必须在通道单元（例如“工业以太网”）中创建连接（例如“Test_Ind_Eth”）。

步骤

1. 选择连接“Test_Ind_Eth”。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
在“名称”(Name) 字段中输入“ETH_Var1_bit” 作为变量名称。
4. 在“数据类型”(Data Type) 字段中，设置“二进制变量”(Binary tag) 数据类型。
5. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
在“数据区”(Data area) 字段中，设置数据所在的自动化系统的数据区。如果选择“DB” 作为数据区，则在已启用的“DB 号”(DB No.) 字段中输入数据块的编号。



6. “寻址”域中的条目“位”无法改变，因为这已经由 WinCC 变量的数据类型“二进制变量”定义。
7. 在其下面的两个域中输入字节地址和位地址。左侧的域标签将取决于“数据区”域中的条目，例如：如果数据区为“DB”且类型域为“二进制变量”，则此域的标签显示为“D”。
8. 如果变量具有要在 WinCC 中使用的质量代码，则选中复选框“质量代码”(quality code)。因此，代码还必须存在于 PLC 中。只有在将数据区选作“DB”时，才能启用此复选框。
9. 单击“确定”(OK) 按钮，关闭上述两个对话框。

参见

如何组态“SIMATIC S7 Protocol Suite”通道 (页 349)

13.5.4.3 如何组态按字节访问的变量


简介

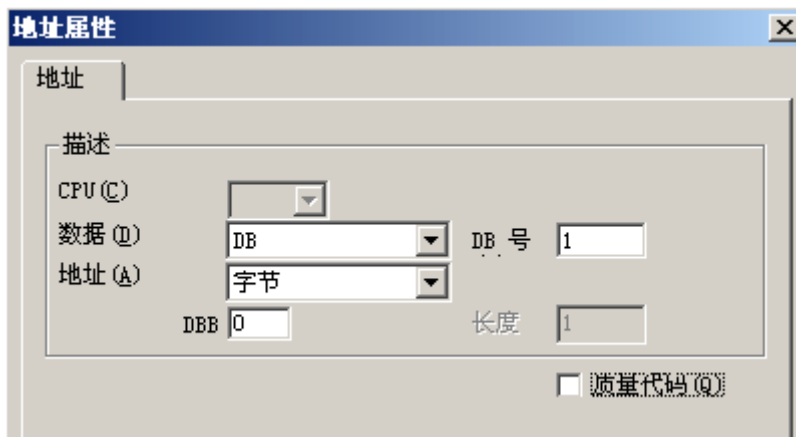
本节说明如何以按字节访问方式组态 PLC 中地址区域的变量。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 必须在通道单元（例如“工业以太网”）中创建连接（例如“Test_Ind_Eth”）。

步骤

1. 选择连接“Test_Ind_Eth”。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
在“名称”(Name) 字段中输入“ETH_Var1_byte”作为变量名称。
4. 在“数据类型”(Data Type) 字段中，将数据类型设置为“无符号 8 位数”(Unsigned 8-bit value)。
5. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
在“数据区”(Data area) 字段中，设置数据所在的自动化系统的数据区。如果选择“DB”作为数据区，则在已启用的“DB 号”(DB No.) 字段中输入数据块的编号。



6. “寻址”字段中的条目“位”无法改变，因为这已经由 WinCC 变量的数据类型“无符号 8 位数”定义。
7. 在下面的字段中输入字节地址。左侧的字段标签将取决于“数据区”(Data Area) 字段中的条目，例如：如果数据区为“DB”且类型为“无符号 8 位数”，则此字段的标签显示为“D”。
8. 如果变量具有要在 WinCC 中使用的质量代码，则选中复选框“质量代码”(quality code)。因此，代码还必须存在于 PLC 中。只有在将数据区选作“DB”时，才能启用此复选框。
9. 单击“确定”(OK) 按钮，关闭上述两个对话框。

参见

如何组态“SIMATIC S7 Protocol Suite”通道 (页 349)

13.5.4.4 如何组态按字访问的变量

简介


本节说明如何以按字访问方式组态 PLC 中地址区域的变量。

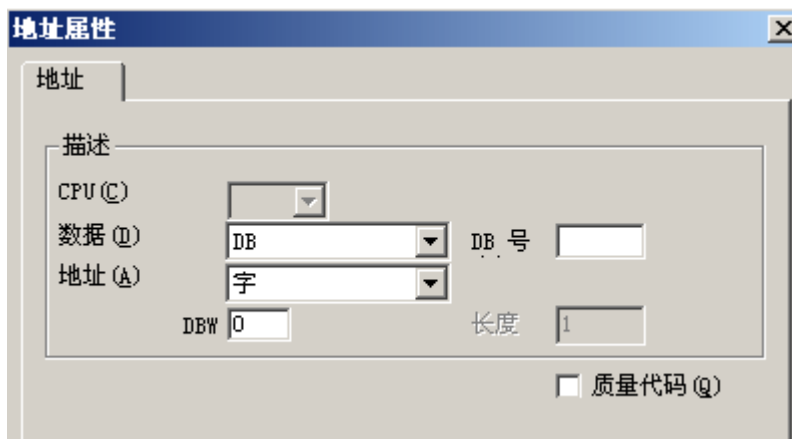
该过程也适用于长度为 4 个字节（“双字”）和更长的变量。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 必须在通道单元（例如“工业以太网”）中创建连接（例如“Test_Ind_Eth”）。

步骤

1. 选择连接“Test_Ind_Eth”。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
在“名称”(Name) 字段中输入“ETH_Var3_word”作为变量名称。
4. 在“数据类型”(Data Type) 字段中，将数据类型设置为“无符号 16 位数”(Unsigned 16-bit value)。
5. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
在“数据区”(Data area) 字段中，设置数据所在的自动化系统的数据区。如果选择“DB”作为数据区，则在已启用的“DB 号”(DB No.) 字段中输入数据块的编号。



6. “寻址”字段中的条目“字”无法改变，因为这已经由 WinCC 变量的数据类型“无符号 16 位数”定义。
7. 在“寻址”域中，输入地址的数字值。左侧的字段标签将取决于“数据区”(Data Area) 字段中的条目，例如：如果类型为“无符号 16 位数”，则此字段的标签显示为“DBW”。

8. 如果变量具有要在 WinCC 中使用的质量代码，则选中复选框“质量代码”(quality code)。因此，代码还必须存在于 PLC 中。只有在将数据区选作“DB”时，才能启用此复选框。
9. 单击“确定”(OK) 关闭所有打开的对话框。

参见

如何组态“SIMATIC S7 Protocol Suite”通道 (页 349)

13.5.4.5 如何组态文本变量

简介

本节说明如何组态文本变量。

WinCC 仅支持 S7 字符串类型，它由一个控制字和实际用户字符串的数据组成，用于 SIMATIC S7 Protocol Suite 通道中的文本变量：

- 要在 WinCC 中组态文本变量，需在用户数据之前输入 PLC 内存中的控制字的地址。控制字的第一个字节包含字符串自定义的最大长度，第二个字节包含实际长度。
- 当在 PLC 内存中插入数据结构时，必须注意 WinCC 中为文本变量组态的长度要增加控制字的 2 个字节。如果文本变量数据结构在内存中一个接一个地直接插入，则后面的数据将会被覆盖。
- PCS7 版本从 V4.01 更新为 V5.0 SP1 时，必须重新映射。因为在 V5.0 以前的版本中，组态文本变量时也涉及用户数据的地址，而从版本 V5.0 开始，必须输入控制字的地址。
- 在进行读操作时，控制字将和用户数据一起被读取，并将判断第二个字节中的当前长度。只有长度与第二控制字节中包含的当前长度一致的用户数据才传送到 WinCC 文本变量。
- 在进行写操作时，将确定字符串的实际长度（“0”字符），且具有当前长度的控制字节将和用户数据一起发送给 PLC。


要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 必须在通道单元（例如“工业以太网”）中创建连接（例如“Test_Ind_Eth”）。

步骤

1. 选择连接“Test_Ind_Eth”。
2. 单击表格区下方的“变量”(Tags) 选项卡。

13.5 组态通道

3. 单击“名称”(Name)列最上方的空单元格。
在“名称”(Name)字段中输入“ETH_Var3_Text”作为变量名称。
4. 在“数据类型”域中，指定数据类型“文本变量，8位字符集”。
5. 打开“地址属性”(Address properties)对话框。
为此，请单击“地址”(Address)字段，然后单击  按钮。
在“数据区”(Data area)中，设置PCL中数据所处的数据区。如果选择“DB”作为数据区，则在已启用的“DB号”(DB No.)字段中输入数据块的编号。



6. “寻址”域中的条目只能设定为“字节”或“字”，因为这些条目由 WinCC 变量的数据类型“文本变量，8位字符集”定义。
7. 在“寻址”域中，输入地址的数字值。涉及控制字的地址。左侧的域标签将取决于“数据区”域中的条目，例如：如果类型域为“字”，则此域的标签显示为“DBW”。
8. 如果变量具有要在 WinCC 中使用的质量代码，则选中复选框“质量代码”(quality code)。因此，代码还必须存在于 PLC 中。只有在将数据区选作“DB”时，才能启用此复选框。
9. 单击“确定”(OK)关闭所有打开的对话框。

参见

如何组态“SIMATIC S7 Protocol Suite”通道 (页 349)

13.5.5 系统参数

13.5.5.1 通道单元的系统参数

引言

如果需要与 WinCC 标准设置不一样的组态，则可以使用通道单元的“系统参数”对话框进行所需要的全部改变。

下列内容可修改：

- 逻辑设备名称
- AS 中周期性读取服务的使用

逻辑设备名称

WinCC 和自动化系统之间的通讯通过逻辑设备名称执行。设备名称是在安装通讯卡时分配的，并且是单元特定的条目。该域将由缺省的单元特定条目来填充，例如，如果是通道单元“MPI”，则为“MPI”。

使用 PLC 中的周期性读取服务

可以指定是否应使用 S7-PLC 的周期性读取服务（也称为周期性变量服务）。这些周期性读取服务对即将周期性读取到单个请求中的变量进行编组，并将其传送给 PLC。PLC 一旦收到请求，将立即传送所需数据，并且每当到达周期时间时都要传送数据。

当激活周期性读取服务时，也可使用变化传送。只有当数值改变后，才可传送数据。该功能必须由相应的自动化系统支持。

说明

SIMATIC S7 和单元标签上的系统参数都是单元特定的数据，因此可以为通道中的每个通道单元单独设置。

参见

如何更改逻辑设备名称 (页 376)

如何组态系统参数 (页 374)

PLC 中的周期性读取服务 (页 373)

13.5.5.2 PLC 中的周期性读取服务

引言

在通道“SIMATIC S7 Protocol Suite”的系统参数中，能够指定是否使用 S7-AS 的周期性读取服务（也称为周期性变量服务）。这些周期性读取服务对即将周期性读取到单个请求中的变量进行编组，并将其传送给 PLC。PLC 一旦收到请求，将立即传送所需数据，并且每当到达周期时间时都要传送数据。当不再需要请求的数据时，例如画面切换时，WinCC 将删除 PLC 中的周期性读取服务。

13.5 组态通道

正常情况下，应该使用 PLC 中的周期性读取服务。因此，已经激活通道单元系统参数中相应的复选框（缺省设置）。只有不想使用周期性服务时，才修改该设置。

只有当激活周期性读取服务时，才可使用变化传送。当数值改变时，数据才能从 AS 中传送且每个自动化系统周期只传送一次。该功能必须由相应的自动化系统支持。

使用周期性读取服务和变化传送会减轻 AS 和 AS-OS 二者的通讯负荷，因为读取请求不需要连续发送给 AS 并处理。

对于非周期性读取服务，要读取的变量被组合在单独的请求中，并被传送到 PLC。PLC 只能发送所需数据一次。请求周期的排列由 WinCC 执行。

CPU 中周期性读取服务的数目

周期性读取服务的数目取决于 S7-PLC 中可用的资源。对于 S7-300，最多有 4 个周期性服务可用，对于 S7-416 或 417，则最多为 32 个。该数目适用于与 PLC 进行通讯的所有成员，也就是说，如果有多个 WinCC 系统与 S7-PLC 进行通讯，则它们必须共享可用的资源。如果超过资源的最大数目，则更多的周期性读取服务访问将被拒绝。于是 WinCC 不得不使用非周期读取请求来请求该数据，并且还必须要执行周期排列。

请求脚本中的外部变量

如果选择的画面不包含任何使用函数“GetTagWord()”请求外部变量的脚本，则一旦画面已经打开，使用周期性读取服务对初始更新不会产生任何影响。如果在画面打开时使用“GetTagWord()”执行脚本，则该脚本的错误组态将导致画面切换后新的变量请求被连续发送到该通道。如果脚本中需要外部变量，则必须将“变量”作为触发事件。

13.5.5.3 如何组态系统参数

引言

本节说明如何组态通道“SIMATIC S7 Protocol Suite”的系统参数。

“系统参数”对话框由两个标签组成：

- SIMATIC S7 标签
- 单元标签

SIMATIC S7 和单元标签上的系统参数都是单元特定的数据，因此可以为通道中的每个通道单元单独设置。

对于 S7 通道的所有通道单元，这些标签是完全相同的。因此，通道单元“MPI”的对话框可用于所有实例中。

对参数值进行的任何改变都只有在 WinCC 重新启动之后才能生效。

说明

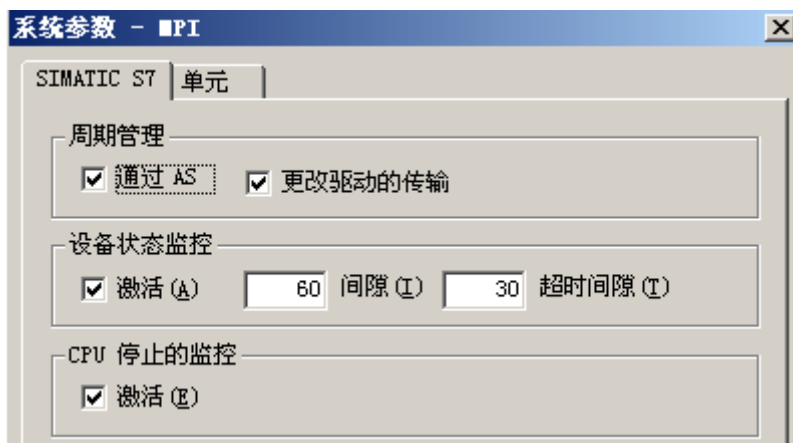
将项目复制到另一台计算机时，单元标签上的设置将保持不变，然而 SIMATIC S7 标签上的设置则可能改变。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。

步骤

1. 在变量管理器中选择通道“SIMATIC S7 Protocol Suite”。使用所期望通道单元的弹出式菜单打开“系统参数”对话框。
2. 选择 SIMATIC S7 标签。如果想要激活通道的变量周期性读取和变化传送功能，则勾选“通过 AS”和“with modification transfer”（使用变化传送）复选框。如果可用，此处将使用 PLC 中的周期性服务。详细信息请参见“PLC 周期性读取服务”。



3. 如果想要使用该功能，激活“设备状态监控”区中的复选框“激活”。在“间隙”域中，输入传送设备状态报文时间间隔的秒数。在“监视时间”域中，输入用于监控设备状态报文的响应的秒数值。
4. 如果要在 S7-CPU 处于停止状态时使 WinCC 指示通讯故障，则激活“CPU 停止的监控”区中的复选框“激活”。

13.5 组态通道

- 选择单元标签。根据所安装的通讯处理器，名称会显示在“逻辑设备名称”域中。只有在安装通讯处理器时选择了不同的名称时，才须改变该名称。详细信息请参见“更改逻辑设备名称”。



- 如果只安装了该通讯类型的一个通讯处理器，则激活复选框“自动设置”，运行系统启动时将自动设置设备名称。
- 如果写入请求的处理要优先于读取请求的处理，则激活复选框“写（带优先级）”。
- 使用“确定”按钮关闭对话框。

参见

如何更改逻辑设备名称 (页 376)

PLC 中的周期性读取服务 (页 373)

13.5.5.4 如何更改逻辑设备名称

引言

通过逻辑设备名称与 S7 通讯。设备名称是在安装通讯处理器时分配的，并且是单元特定的条目。

根据已安装的通讯处理器，现在已为设备名称建立了相应的预设值。这些在下面的表格“缺省设备名称”中列出。

S7 通道所有单元的标签均完全相同，因此，描述中使用了通道单元“MPI”的对话框。

缺省设备名称

通道单元	缺省设备名称
工业以太网	CP_H1_1:
工业以太网 (II)	CP_H1_2:
MPI	MPI
命名连接	VM/
PROFIBUS	CP_L2_1:
PROFIBUS (II)	CP_L2_2:
Slot PLC	SLOT_PLC
Soft PLC	SOFT_PLC
TCP/IP	CP-TCP/IP

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 必须创建与通道单元的连接，例如“MPI”。

步骤

1. 在变量管理器中选择所期望的通道单元。
2. 使用弹出式菜单打开“系统参数”对话框。
3. 选择单元标签。
4. 在域“逻辑设备名称”中指定设备名称。用户既可从选择列表中选择条目，也可手动输入新的名称。
所有可能的名称均由“组态 PG/PC 接口”（控制面板）工具确定。如果没有安装此工具，只能显示当前设定的设备名称。如果指定不同的逻辑设备名称，则将显示一条消息。
只有在目标站使用了组态站上尚未安装的通讯卡时，才能手动输入。
5. 单击“确定”按钮，关闭对话框。

说明

逻辑设备名称必须与设备设置中的名称的每个字母都完全一样。同时，“工业以太网”和“PROFIBUS”的缺省逻辑设备名称在名称末尾有一个冒号。

对参数值进行的任何改变都只有在 WinCC 重新启动之后才能生效。

13.6 特殊功能

13.6.1 “SIMATIC S7 Protocol Suite” 通道的特殊功能

引言

SIMATIC S7 Protocol Suite 包含一些特殊功能；本章将说明这些特殊功能。

参见

软件冗余 (页 418)

通道“SIMATIC S7 Protocol Suite” 的原始数据变量 (页 410)

使用 S7 功能块 AR_SEND 进行数据交换 (页 378)

13.6.2 使用 S7 功能块 AR_SEND 进行数据交换

13.6.2.1 使用 S7 功能块 AR_SEND 进行数据交换

简介

S7-400 AS 中的 S7 功能块 AR_SEND 用于将过程值传送到过程值归档。

操作原理

为了将 PLC 的过程值传送给 WinCC 中的过程值归档，“S7-400 PLC” 具有集成的功能组件 SFB37“AR_SEND”。

AR_SEND 组件的基本函数可向归档变量提供数据。如果使用了 AR_ID-Subnumber，则可以向多个变量提供数据。如果使用 AR_SEND 组件，则不是将过程值单个地发送到归档库，而是在 PLC 中累积多个过程值，然后以信息包方式传送。这样可减少所用网络上的负荷。

在 PLC 中，可使用的 AR_SEND 组件的数目取决于 CPI（例如 CPU 416 最多可使用 32 个 AR_SEND）。某 AR_ID 可依次分配给各 AR_SEND 组件。子编号用于增加可传送过程数据量，因为各 AR_ID 最多有 4095 个子编号。

实际上，每个 AR_SEND 组件要传送的归档变量的数目受到要传送的数据区的最大长度的

限制。有关“数据块结构的结构和参数”的详细信息，请参见“过程值的数目”参数的描述。

AR_ID 和 AR_ID 子编号在 PLC 的数据和归档变量间建立了分配，并在组态 PLC 中数据库内的数据结构时，使用其它参数对其进行定义。

当自动计算其它参数时，在 WinCC 中组态该数据分配。

必须首先完成 PLC 中的 SFB 37“AR_SEND”的组态以及数据块结构的设置，因为在 WinCC 中组态是基于 PLC 中的这些数值。关于组态“AR_SEND”功能组件的信息，请参见 S7-400 PLC 文档。

AR_SEND 变量概述

变量： AR_SEND 用于...	每个 AR_SEND 的过程 控制归档变量数目	预期用途
一个归档变量	a	用于为归档变量传送过程值，可在很短的时间间隔内读取过程值。
多个归档变量	对应于 AR_ID 子编号的 数目	用于为多个归档变量传送过程值，可在很短的时间间隔内读取过程值。
多个归档变量（已 优化）	对应于 AR_ID 子编号的 数目	用于周期性地为最大数目的归档变量提供数据，每次提供一个数值。

参见

用于多个归档变量的 AR_SEND 变量 (页 392)

如何组态用于多个归档变量的 AR_SEND 变量 (页 408)

如何组态用于归档变量的 AR_SEND 变量 (页 406)

用于多个归档变量的 AR_SEND 变量（已优化） (页 404)

用于一个归档变量的 AR_SEND 变量 (页 387)

13.6.2.2 数据块 - 结构和参数

简介

在使用“AR_SEND”功能块传送 PLC 中的数据前，首先要把数据组织成一个或多个数据块。数据块结构取决于各种参数，例如使用的 AR_SEND 变量是否有时间戳或过程值的数据类型。

数据块中所使用的参数如下所述。在 AS 的数据块和“AR_SEND”功能块参数中设置各参数值。

当在 WinCC 中评估数据块时，进行参数检查。如果 WinCC 检测到数据块的结构出错或归档变量组态与所接收的数据不匹配，具有以下结构的条目会记录在 WinCC 诊断记录册中：

“日期时间、1003080、4、用户名、计算机名称、NRMS7PMC、PdeReceive: 来自连接（连接名）...的未知参数 AR_SEND + 描述错误的附加信息”

如果使用 WinCC 系统消息组态消息系统，则该诊断条目也会触发编号为 1003080 的 OS 过程控制消息。关于记录册条目的文本，请参见该消息的注释。

数据块的结构

每个数据块由报头和用户数据区组成：

- 报头包含与过程值及其周期以及时间戳有关的信息。
- 用户数据区包含实际过程值。

要传送的数据区由一个或多个数据块组成。



说明

在数据块中，每一行代表两个字节。过程值的长度根据数据类型而定，可以是一个字节或多个字节。详细信息请参见“过程值的数目”参数的描述。

参数描述**报头类型**

报头类型定义了报头中所包含的信息的类型。

报头类型	时间戳	AR_ID 子编号
0	无时间戳的报头	无 AR_ID 子编号的报头
1	有时间戳的报头	无 AR_ID 子编号的报头
8	无时间戳的报头	有 AR_ID 子编号的报头
9	有时间戳的报头	有 AR_ID 子编号的报头

说明

如果报头类型为 0 和 8，报头中将省略时间戳的字节。因为这些字节也不会保留在数据块中，所以报头将减少 8 个字节。

AR_ID 子编号

建立 AS 用户数据和 WinCC 归档变量之间的分配，并在两处地方进行组态：

- 在 WinCC 中 中组态过程控制归档变量
- 在 PLC 中设置要传送的用户数据区

子编号只与报头类型 8 或 9 有关。子编号有效值的范围为 1 到 4095。参数按十六进制值 (1 - 0FFF) 输入 WinCC。

时间戳

时间戳包含 SIMATIC S7 BCD 格式的日期和时间。WinCC 不使用工作日条目。

说明

自动化系统 S7 无法识别夏令时/冬令时的切换。在 AS 中，应将本地冬令时设置为系统时间。夏令时或冬令时的时间戳在 WinCC 中由标准化的 DLL 更正。已更正的时间和夏令时/冬令时的时间标识号可用于 WinCC 应用程序。然后将更正的时间和 ID 添加至归档（例如变量记录）。

Cycle

在周期中读取过程值。该参数是在单位（范围）处指定的时间单位的因子。数据长度：双字。

例如：

“周期”= 10；“单位（区域）”= 4 意味着：过程值读取周期 = 10 秒

单元（类型）

指定时间信息的类型并修改参数“过程值的数目”。

编号	含义
1	按相等的时间间隔读取过程值。 启动时间在报头时间戳中给出，并且是强制性的。过程值之间的时间间隔由“单位（范围）”的时间单位和“周期”因素来定义。
2	每个过程值包含一个时间戳。 不会估算报头中可能给出的时间戳。其结构对应于具有 8 个字节长度的报头中的时间戳。
3	每个过程值均拥有一个相对时间差，该时间差具有 2 个字的数据长度的时间单位。 绝对时间是报头中的时间戳的总和（= 启动时间），而相对时间差则使用在“单位（范围）”中设置的时间单位。报头的时间戳条目是强制性的。
4	每个过程值包含 AR_ID 子编号。 报头中给定的时间戳应用于过程值。报头的时间戳条目是强制性的。

单位（范围）

指定用于单位（类型）的时间单位 = 1 或 3。

编号	含义
1	保留
2	保留
3	毫秒
4	秒
5	分
6	小时
7	日

过程数据 - 数据类型

过程值直接以 S7 格式存储。

编号	S7 数据类型	WinCC 数据类型
0	BYTE	BYTE
1	WORD	WORD
2	INT	SWORD
3	DWORD	DWORD
4	DINT	SDWORD
5	实型	FLOAT

过程值的数目

根据“单元（类型）”中的条目，所传送的数据区可包含特定数量的过程值。该数目受到所传送的数据区的最大长度 (16 KB) 的限制。

当使用 S7 功能“AR_SEND”和“BSEND/BRCV”来与 S7-400 进行通讯时，请注意存在资源限制，也就是说，使用 AR_SEND 和/或 BSEND/BRCV 同时从 AS 发送到 WinCC 的数据至多为 16 KB。

说明

对于 AR_SEND 变量“多个归档变量”，下列限制适用于该参数：

各种归档变量的数据块必须始终从字边界开始。因此，对于组合“数据类型过程值”= 0（字节）和“单元（类型）”= 1（具有相同时间间隔的过程值）的情况，必须输入一个偶数数目的过程值（= 字节），以用于参数“过程值的数目”。这种限制仅适用于该 AR_SEND 变量以及数据类型和“单元（类型）”的组合。

实例：

1x BSEND 最多为 16 KB

或 1 x 8 KB 的 AR_SEND + 1 x 8 KB 的 BSEND

或 1 x 10 KB 的 AR_SEND + 1 x 2 KB 的 AR_SEND + 1 x 4 KB 的 BSEND

单元 (类型)	过程值数目的含义
1	按相等的时间间隔读取过程值： 可以传送 8000 个 WORD/INT 数据类型的过程值，或 4000 个 DWORD/DINT/实型数据类型的值。
2	带时间戳的过程值： 用户数据区的每个元素由时间戳（8 个字节）和数值组成。可以传送 WORD/INT 数据类型的过程值 1600 个，或 DWORD/DINT/实型数据类型的值 1333 个。
3	带时差的过程值： 用户数据区的每个元素由时间戳（4 个字节）和数值组成。可以传送 WORD/INT 数据类型的过程值 2666 个，或 DWORD/DINT/实型数据类型的值 2000 个。
4	过程值包含 AR_ID 子编号（具有多个变量的 AR-SEND，已优化）。 对于类型 4，每个过程值均由一个带有 AR_ID 子编号（数值范围：1 - 0x0FFF）的字和数值组成。因此，用户数据区包括前面带“AR_ID”子编号的过程值数组。可以传送 WORD/INT 数据类型的过程值 3992 个，或 DWORD/DINT/实型数据类型的值 2660 个。

说明

数据块中给定的 AR_ID 子编号必须全部在 WinCC 中组态。如果发现存在未组态的子编号，WinCC 将停止对用户数据的说明。

各种归档变量的数据块必须始终从字边界开始。因此，对于数据类型字节和“单元（类型）”= 1（具有相同时间间隔的过程值），必须输入一个偶数数目的过程值（= 字节），以用于参数“过程值的数目”。这种限制仅适用于该 AR_SEND 变量以及数据类型和“单元（类型）”的组合。

参见

如何组态用于多个归档变量的 AR_SEND 变量 (页 408)

如何组态用于归档变量的 AR_SEND 变量 (页 406)

用于多个归档变量的 AR_SEND 变量（已优化）(页 404)

用于多个归档变量的 AR_SEND 变量 (页 392)

用于一个归档变量的 AR_SEND 变量 (页 387)

13.6.2.3 AR_SEND 变量的属性概述**引言**

借助实例，这些表格将说明各种 AR_SEND 变量的属性及可能的参数值。

该表格不能显示所有可能的组合。

“过程值的数据类型”的列“报头类型”以其在报头中的顺序显示。

说明

当组态功能块“AR_SEND”和 AS 中数据块的数据结构时，设置用于 AR_ID 和 AR_ID 子编号的数值以及一些其它参数。

13.6 特殊功能

用于归档变量的变量

实例/属性	实例编号	报头类型	日期/时间 (报头的时间标志)	周期因子	单元 (类型)	单位 (范围)	AR_ID 子编号	过程值 数据类型	最大 数目 过程值	实例中的 过程值结构
每个带有自己时间标志的过程值 (字节)	1	0	不可用	0	2	0	0	0 1; 2 3; 4; 5	3200 1600 1333	8 个字节时间标志 + 1 个字节过程值
具有等间隔时间标志的过程值	2	1	相关的	≥ 1	1	3 到 7	0	0 1; 2 3; 4; 5	16000 8000 4000	1 个字过程值
每个带有自己时间标志的过程值 (字)	3	1	不相关的	0	2	0	0	0 1; 2 3; 4; 5	3200 1600 1333	8 个字节时间标志 + 1 个字过程值
每个带有时差的过程值	4	1	相关的	≥ 1	3	3 到 7	0	0 1; 2 3; 4; 5	5332 2666 2000	8 个字节时间标志 + 1 个字节过程值

用于多个归档变量的变量

实例/属性	实例编号	报头类型	日期/时间 (报头的时间标志)	周期因子	单元 (类型)	单位 (范围)	AR_ID 子编号	过程值 数据类型	最大 数目 过程值	实例中的过 程值结构
每个带有自己时间标志的过程值 (字节)	5	8	不可用	0	2	0	1 到 4095	0 1; 2 3; 4; 5	3200 1600 1333	8 个字节时间标志 + 1 个字节过程值
具有等间隔时间标志的过程值	6	9	相关的	≥ 1	1	3 到 7	1 到 4095	0 1; 2 3; 4; 5	16000 8000 4000	1 个字过程值

实例/属性	实例编号	报头类型	日期/时间 (报头的时间标志)	周期因子	单元 (类型)	单位 (范围)	AR_ID 子编号	过程值 数据类型	最大 数目 过程值	实例中的过 程值结构
每个带有自己时间标志的过程值(字)	7	9	不相关的	0	2	0	1 到 4095	0 1; 2 3; 4; 5	3200 1600 1333	8 个字节时间标志 + 1 个字过程值
每个带有时差的过程值	8	9	相关的	0	3	3 到 7	1 到 4095	0 1; 2 3; 4; 5	5332 2666 2000	8 个字节时间标志 + 1 个字节过程值

用于多个归档变量的变量 - 已优化

实例/属性	实例编号	报头类型	日期/时间 (报头的时间标志)	周期因子	单元 (类型)	单位 (范围)	AR_ID 子编号	过程值 数据类型	最大 数目 过程值	实例中的过 程值结构
包含 AR_ID 子编号的各个过程值	9	1	相关的	0	4	0	0	1; 2 3; 4; 5	3992 2660	1 个字子编号 +1 个字过程值

参见

用于多个归档变量的 AR_SEND 变量 (已优化) (页 404)

用于多个归档变量的 AR_SEND 变量 (页 392)

用于一个归档变量的 AR_SEND 变量 (页 387)

13.6.2.4 用于一个归档变量的 AR_SEND 变量

引言

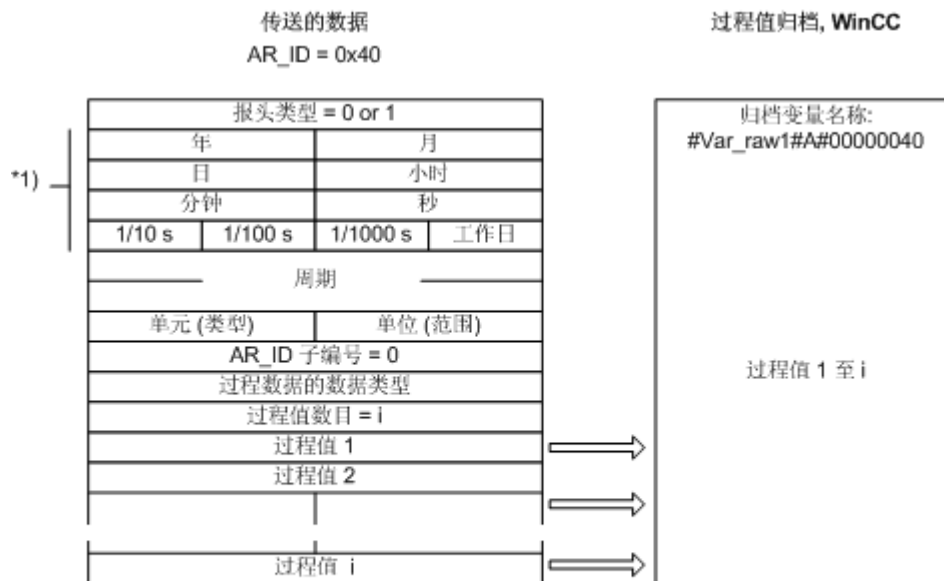
该变量可用于为归档变量提供过程值。也可用于较早的 WinCC 版本 (V5.0 之前的)。

变量的属性:

- 报头类型必须为 0 或 1，即没有 AR_ID 子编号和有/没有时间标志。
- 不计算报头中的 AR_ID 子编号。
- WinCC 中的归档变量名不包含任何 AR_ID 子编号，因为只传送归档变量的过程值。

数据区结构实例

要传送的数据区由数据块组成。



参见

数据块结构实例 4: 一个归档变量; 每个过程值具有相对时间标志 (时差) (页 391)

数据块结构实例 3: 一个归档变量; 每个过程值具有自己的时间标志 (页 390)

数据块结构实例 2: 一个归档变量; 等间隔的时间标志 (页 389)

数据块结构实例 1: 一个归档变量; 每个过程值具有一个时间标志 (页 389)

AR_SEND 变量的属性概述 (页 385)

13.6.2.5 数据块结构实例 1：一个归档变量；每个过程值具有一个时间标志

引言

在本实例中，将只为一个归档变量传送过程值。报头没有时间标志，并且不保留相应的字节数。因此，时间标志（8 个字节）位于每个过程值（1 个字节）之前。

过程值的数据类型为“字节”。

数据组件中的数据块结构

在 DB 中的地址	发送的数据块
0.0	报头类型 = 0
2.0	周期 = 0
4.0	单元 (类型) = 2 单位 (范围) = 0
6.0	AR_ID-子编号 = 0
8.0	过程数据 - 数据类型 = 0
10.0	过程值数目 = 3 (最大 3200)
12.0	年=2001 月=10
14.0	日=05 小时=13
16.0	分钟=40 秒=00
18.0	1/10 s 1/100 s 1/1000 s 工作日
20.0	过程值 1
22.0	-
24.0	年=2001 月=10
26.0	日=05 小时=14
28.0	分钟=40 秒=00
30.0	1/10 s 1/100 s 1/1000 s 工作日
32.0	过程值 2
34.0	-
36.0	年=2001 月=10
38.0	日=05 小时=15
40.0	分钟=40 秒=00
42.0	1/10 s 1/100 s 1/1000 s 工作日
	过程值 3
	-

参见

数据块 - 结构和参数 (页 380)

13.6.2.6 数据块结构实例 2：一个归档变量；等间隔的时间标志

引言

在本实例中，将只为一个归档变量传送过程值。

13.6 特殊功能

使用参数“周期”= 1 和“单位（范围）”= 4（= 秒）创建 1 秒的等间隔时间标志。
过程值的数据类型为“WORD”。

数据组件中的数据块结构

在 DB 中的地址	发送的数据块
0.0	报头类型 = 1
2.0	年=2001
4.0	日=05
6.0	分钟=40
8.0	1/10 s
10.0	1/100 s
12.0	1/1000 s
14.0	工作日
16.0	周期 = 1
18.0	单元 (类型) = 1
20.0	单位 (范围) = 4
22.0	AR_ID-子编号 = 0
24.0	过程数据 - 数据类型 = 1
26.0	过程值编号 = 8 (最大 8000)
28.0	过程值 1
30.0	过程值 2
32.0	过程值 3
34.0	过程值 4
36.0	过程值 5
	过程值 6
	过程值 7
	过程值 8

参见

数据块 - 结构和参数 (页 380)

13.6.2.7 数据块结构实例 3: 一个归档变量; 每个过程值具有自己的时间标志

引言

在本实例中, 将只为一个归档变量传送过程值。报头的时间标志不重要。因此, 时间标志 (8 个字节) 位于每个过程值 (1 个字) 之前。

过程值的数据类型为“SWORD”。

数据组件中的数据块结构

在 DB 中的地址	发送的数据块
0.0	报头类型 = 1
2.0	年=0
4.0	日=0
6.0	分钟=0
8.0	1/10 s 1/100 s 1/1000 s 工作日
10.0	周期 =0
12.0	
14.0	单元 (类型) = 2 单位 (范围) = 0
16.0	AR_ID-子编号 = 0
18.0	过程数据 - 数据类型 = 2
20.0	过程值数目 = 2 (最大为1600)
22.0	年=2001
24.0	日=05
26.0	分钟=40
28.0	1/10 s 1/100 s 1/1000 s 工作日
30.0	过程值 1
32.0	年=2001
34.0	日=05
36.0	分钟=40
38.0	1/10 s 1/100 s 1/1000 s 工作日
40.0	过程值 2

参见

数据块 - 结构和参数 (页 380)

13.6.2.8 数据块结构实例 4： 一个归档变量； 每个过程值具有相对时间标志（时差）

引言

在本实例中，将为具有时间标志的一个归档变量传送过程值。

参数“单元（类型）”= 3，每个过程值有一个相对于报头中时间标志的时差（4 个字节）。

参数“单位（范围）”= 4，定义时差的单位为秒。

过程值的数据类型为“DWORD”。

数据组件中的数据块结构

在 DB 中的地址	发送的数据块
0.0	报头类型 = 1
2.0	年=2001
4.0	日=05
6.0	分钟=40
8.0	1/10 s 1/100 s 1/1000 s 工作日
10.0	周期 = 0
12.0	
14.0	单元 (类型) = 3 单位 (范围) = 4
16.0	AR_ID-子编号 = 0
18.0	过程数据 - 数据类型 = 3
20.0	过程值数目 = 3 (最大为 2000)
22.0	
24.0	相对时间差, 以秒为单位
26.0	
28.0	过程值 1
30.0	
32.0	相对时间差, 以秒为单位
34.0	
36.0	过程值 2
38.0	
40.0	相对时间差, 以秒为单位
42.0	
44.0	过程值 3

参见

数据块 - 结构和参数 (页 380)

13.6.2.9 用于多个归档变量的 AR_SEND 变量

引言

该变量允许给一个或多个归档变量提供过程值。对每个归档变量，将会分配一个 AR_ID 子编号，并且在需传送的数据区中创建数据块。

可以为每个 AR_ID 子编号传送“x”个过程值。有关“数据块结构的结构和参数”的更多信息，请参见“过程值的数目”参数的描述。

按照给定的“单元（类型）”和“单位（范围）”，从要传送的数据区取出归档变量值的时间标志。然后将其发送到 WinCC 过程值归档。

变量的属性:

- 报头类型必须是 8 或 9（有/没有时间标志，有 AR_ID 子编号）。
- 对每个 AR_ID 子编号，必须在需传送的数据区中创建数据块。
- 每个数据块中的 AR_ID 子编号必须大于 0。
- 在 WinCC 中，归档变量名具有 AR_ID 子编号。

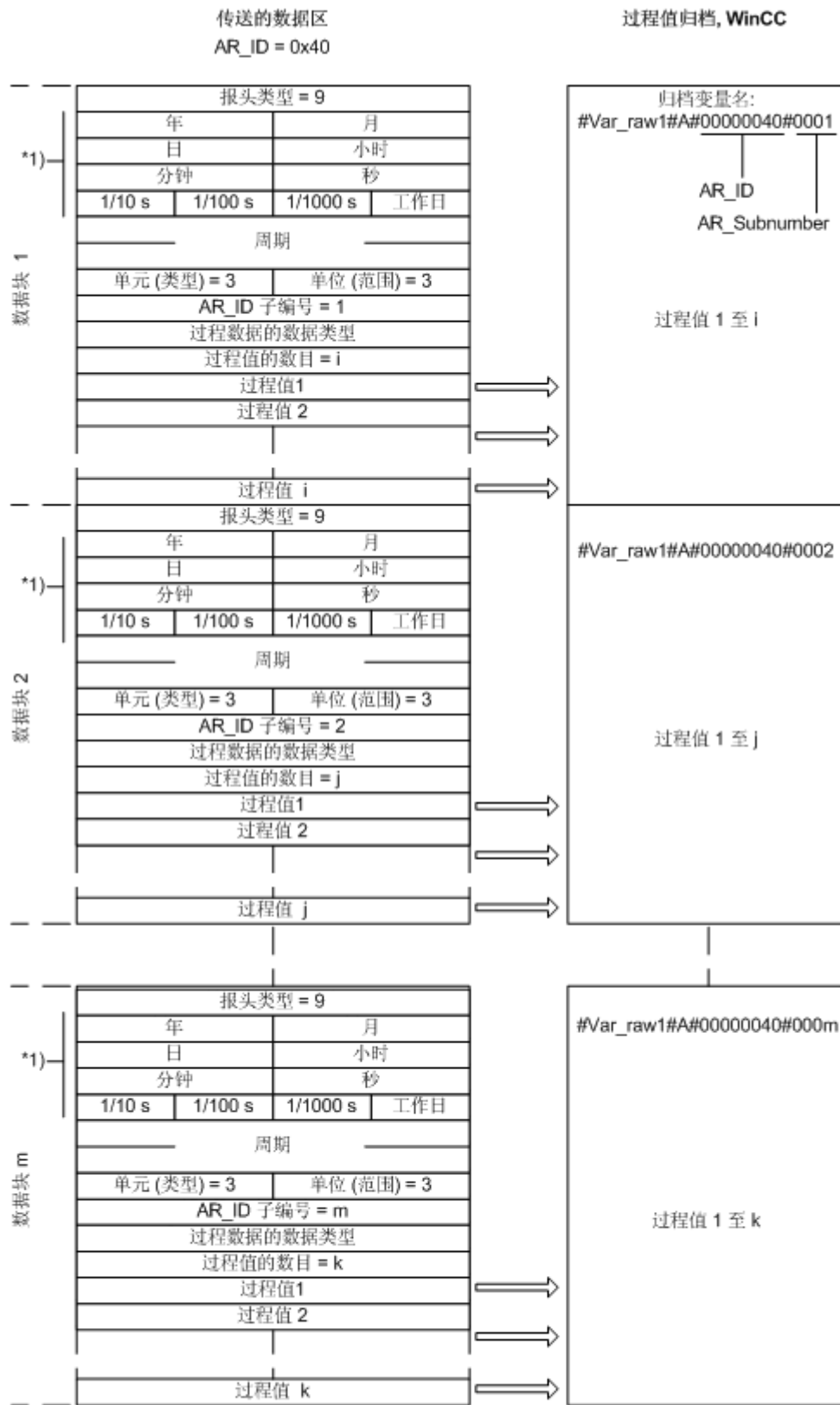
说明

数据块中给定的 AR_ID 子编号必须全部在 WinCC 中组态。如果发现存在未组态的子编号，WinCC 将停止对用户数据的说明。

各种归档变量的数据块必须始终从字边界开始。因此，对于组合“数据类型过程值”= 0（字节）和“单元（类型）”= 1（具有相同时间间隔的过程值）的情况，必须输入一个偶数数目的过程值（= 字节），以用于参数“过程值的数目”。这种限制仅适用于该 AR_SEND 变量以及数据类型和“单元（类型）”的组合。

数据区结构实例

所要传送的数据区由一个或多个数据块组成，这些数据块对应于需要数据的归档变量的数目。



13.6 特殊功能

参见

数据块结构实例 8: 多个归档变量; 过程值具有相对时间标志 (时差) (页 402)

数据块结构实例 7: 多个归档变量; 每个过程值具有自己的时间标志 (页 400)

数据块结构实例 6: 多个归档变量; 等间隔的时间标志 (页 398)

数据块结构实例 5: 多个归档变量; 每个过程值具有自己的时间标志 (页 396)

AR_SEND 变量的属性概述 (页 385)

13.6.2.10 数据块结构实例 5: 多个归档变量; 每个过程值具有自己的时间标志

引言

在本实例中, 将传送用于多个归档变量的过程值。

不同归档变量的数据块在数据组件中都按顺序放置。在各个数据块中, 输入不同的 AR_ID 子编号。

报头没有时间标志, 并且不保留相应的字节数。因此, 时间标志 (8 个字节) 位于每个过程值 (1 个字节) 之前。

过程值的数据类型为“字节”。

数据组件中的数据块结构

在 DB 中的地址	发送的数据块
0.0	报头类型 = 8
2.0	周期 = 0
4.0	
6.0	单元 (类型) = 2 单位 (范围) = 0
8.0	AR_ID-子编号 = 1
10.0	过程数据-数据类型 = 0
12.0	过程值数目 = 3
14.0	年=2001 月=10
16.0	日=05 小时=13
18.0	分钟=40 秒=00
20.0	1/10 s 1/100 s 1/1000 s 工作日
22.0	过程值 1 -
24.0	年=2001 月=10
26.0	日=05 小时=14
28.0	分钟=40 秒=00
30.0	1/10 s 1/100 s 1/1000 s 工作日
32.0	过程值 2 -
34.0	年=2001 月=10
36.0	日=05 小时=15
38.0	分钟=40 秒=00
40.0	1/10 s 1/100 s 1/1000 s 工作日
42.0	过程值 3 -
44.0	报头类型 = 8
46.0	周期 = 0
48.0	
50.0	单元 (类型) = 2 单位 (范围) = 0
52.0	AR_ID-子编号 = 2
54.0	过程数据 - 数据类型 = 0
56.0	过程值数目 = 2
58.0	年=2001 月=10
60.0	日=05 小时=12
62.0	分钟=40 秒=00
64.0	1/10 s 1/100 s 1/1000 s 工作日
66.0	过程值 1 -
68.0	年=2001 月=10
70.0	日=05 小时=13
72.0	分钟=40 秒=00
74.0	1/10 s 1/100 s 1/1000 s 工作日
76.0	过程值 2 -

参见

数据块 - 结构和参数 (页 380)

13.6.2.11 数据块结构实例 6：多个归档变量；等间隔的时间标志

引言

在本实例中，将传送用于多个归档变量的过程值。不同归档变量的数据块在数据组件中都按顺序放置。在各个数据块的报头中，输入不同的 AR_ID 子编号。

使用参数“周期”= 1 和“单位（范围）”= 4（= 秒）创建 1 秒的等间隔时间标志。

过程值的数据类型为“WORD”。

数据组件中的数据块结构

在 DB 中的地址	发送的数据块
0.0	报头类型 = 9
2.0	年=2001
4.0	日=05
6.0	分钟=40
8.0	1/10 s 1/100 s 1/1000 s 工作日
10.0	周期 =1
12.0	
14.0	单元 (类型) = 1 单位 (范围) = 4
16.0	AR_ID-子编号 = 1
18.0	过程数据 - 数据类型 = 1
20.0	过程值数目 = 8
22.0	过程值 1
24.0	过程值 2
26.0	过程值 3
28.0	过程值 4
30.0	过程值 5
32.0	过程值 6
34.0	过程值 7
36.0	过程值 8
38.0	报头类型 = 9
40.0	年=2001
42.0	日=05
44.0	分钟=40
46.0	1/10 s 1/100 s 1/1000 s 工作日
48.0	周期 =1
50.0	
52.0	单元 (类型) = 1 单位 (范围) = 4
54.0	AR_ID-子编号 = 2
56.0	过程数据 - 数据类型 = 1
58.0	过程值数目 = 5
60.0	过程值 1
62.0	过程值 2
64.0	过程值 3
66.0	过程值 4
68.0	过程值 5

参见

数据块 - 结构和参数 (页 380)

13.6.2.12 数据块结构实例 7：多个归档变量；每个过程值具有自己的时间标志

引言

在本实例中，将传送用于多个归档变量的过程值。不同归档变量的数据块在数据组件中都按顺序放置。在各个数据块中，输入不同的 AR_ID 子编号。

报头的时间标志不重要。因此，时间标志（8 个字节）位于每个过程值（1 个字）之前。

过程值的数据类型为“SWORD”。

数据组件中的数据块结构

在 DB 中的地址	发送的数据
0.0	报头类型= 9
2.0	年=0
4.0	日=0
6.0	分钟=0
8.0	1/10 s 1/100 s 1/1000 s 工作日
10.0	周期 =0
12.0	
14.0	单元 (类型) = 2 单位 (范围) = 0
16.0	AR_ID-子编号 = 1
18.0	过程数据 - 数据类型 = 2
20.0	过程值数目 = 3
22.0	年=2001
24.0	日=05
26.0	分钟=40
28.0	1/10 s 1/100 s 1/1000 s 工作日
30.0	过程值 1
32.0	年=2001
34.0	日=05
36.0	分钟=40
38.0	1/10 s 1/100 s 1/1000 s 工作日
40.0	过程值 2
42.0	年=2001
44.0	日=05
46.0	分钟=40
48.0	1/10 s 1/100 s 1/1000 s 工作日
50.0	过程值 3
52.0	报头类型= 9
54.0	年=0
56.0	日=0
58.0	分钟=0
60.0	1/10 s 1/100 s 1/1000 s 工作日
62.0	周期 =0
64.0	
66.0	单元 (类型) = 2 单位 (范围) = 0
68.0	AR_ID-子编号 = 2
70.0	过程数据 - 数据类型 = 2
72.0	过程值数目 = 2
74.0	年=2001
76.0	日=05
78.0	分钟=40
80.0	1/10 s 1/100 s 1/1000 s 工作日
82.0	过程值 1
84.0	年=2001
86.0	日=05
88.0	分钟=40
90.0	1/10 s 1/100 s 1/1000 s 工作日
92.0	过程值 2

参见

数据块 - 结构和参数 (页 380)

13.6.2.13 数据块结构实例 8: 多个归档变量; 过程值具有相对时间标志 (时差)

引言

在本实例中, 将为具有时间标志的一个归档变量传送过程值。

不同归档变量的数据块在数据组件中都按顺序放置。在各个数据块中, 输入不同的 AR_ID 子编号。

参数“单元 (类型)”= 3, 每个过程值有一个相对于报头中时间标志的时差 (4 个字节)。为每个归档变量使用参数“单位 (范围)”单独定义时差的时间单位, 因此, 也为各数据块进行了设置。

过程值的数据类型为“DWORD”。

数据组件中的数据块结构

在 DB 中的地址	发送的数据块
0.0	报头数据 = 9
2.0	年=2001
4.0	日=05
6.0	分钟=40
8.0	1/10 s 1/100 s 1/1000 s 工作日
10.0	周期 =0
12.0	
14.0	单元 (类型) = 3 单位 (范围) = 5
16.0	AR_ID-子编号 = 1
18.0	过程数据 - 数据类型 = 3
20.0	过程值数目 = 3
22.0	相对时间差, 以分钟为单位
24.0	
26.0	过程值 1
28.0	
30.0	相对时间差, 以分钟为单位
32.0	
34.0	过程值 2
36.0	
38.0	相对时间差, 以分钟为单位
40.0	
42.0	过程值 3
44.0	
46.0	报头数据 = 9
48.0	年=2001
50.0	日=05
52.0	分钟=40
54.0	1/10 s 1/100 s 1/1000 s 工作日
56.0	周期 =0
58.0	
60.0	单元 (类型) = 3 单位 (范围) = 6
62.0	AR_ID-子编号 = 2
64.0	过程数据 - 数据类型 = 3
66.0	过程值数目 = 2
68.0	相对时间差, 以小时为单位
70.0	
72.0	过程值 1
74.0	
76.0	相对时间差, 以小时为单位
78.0	
80.0	过程值 2
82.0	

参见

数据块 - 结构和参数 (页 380)

13.6.2.14 用于多个归档变量的 AR_SEND 变量（已优化）

引言

每当一次性为最大数目的归档变量提供过程值时，使用该变量。因此，将被传送的数据区只有一个数据块组成，并且每个过程值只有自己的 AR_ID 子编号及其相应数值。

在该数据块中，所有归档变量的过程值的数据类型是一样的。

变量的属性：

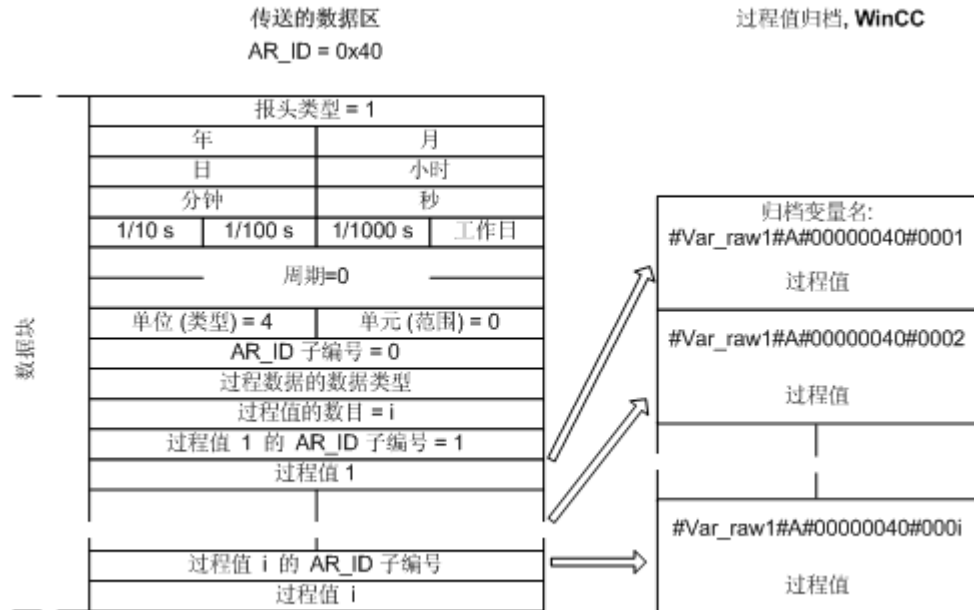
- 报头类型必须是 1（有时间标志，没有 AR_ID 子编号）。
- 在数据块中相关过程值的 AR_ID 子编号必须大于 0。不计算报头中的 AR_ID 子编号。
- “单元（类型）”参数必须是 4，也就是说，过程值包含 AR_ID 子编号。
- “单位（范围）”参数必须是 0，也就是说，报头中的时间标志适用于所有过程值，并且没有相对时间。
- 在 WinCC 中，归档变量名具有 AR_ID 子编号。

说明

如果对于过程值所包含的 AR_ID 子编号不存在任何 WinCC 归档变量与其对应，将出现一条 WinCC 诊断记录条目。继续处理其余的过程值。

数据区结构实例

要传送的数据区由一个数据块组成。



参见

AR_SEND 变量的属性概述 (页 385)

数据块结构实例 9: 多个归档变量; 已优化 (页 405)

13.6.2.15 数据块结构实例 9: 多个归档变量; 已优化

引言

在本实例中, 将为具有时间标志的一个归档变量传送过程值。时间标志适合于所有的归档变量。

相应的 AR_ID 子编号位于每个过程值的前面。

过程值的数据类型为“WORD”。

13.6 特殊功能

数据组件中的数据块结构

在 DB 中的地址	发送的数据块
0.0	报头类型 = 9
2.0	年=2001
4.0	日=05
6.0	分钟=40
8.0	1/10 s 1/100 s 1/1000 s 工作日
10.0	周期 = 0
12.0	
14.0	单元 (类型) = 4 单位 (范围) = 0
16.0	AR_ID-子编号 = 0
18.0	过程数据 - 数据类型 = 1
20.0	过程值数目 = 5 (最大为 3992)
22.0	AR_ID-子编号
24.0	过程值1
26.0	AR_ID-子编号
28.0	过程值 2
30.0	AR_ID-子编号
32.0	过程值 3
34.0	AR_ID-子编号
36.0	过程值 4
38.0	AR_ID-子编号
40.0	过程值 5

参见

数据块 - 结构和参数 (页 380)

13.6.2.16 如何组态用于归档变量的 AR_SEND 变量

简介

有大量变量使用 AR_SEND 功能来交换数据。对于变量“一个归档变量”，只需使用 AR_ID。而不使用 AR_ID 子编号。

AR_ID 在组态 AS 中的数据块与功能块 SFB 37“AR_SEND”期间，用于建立 AS 中的数据与归档变量之间的分配并和其它参数一起组态。

在 WinCC 中，这种分配在过程控制变量的属性中完成。该分配是 WinCC 中所必须的唯一组态步骤，本节将对其进行说明。

WinCC 中其它的参数不需要组态，因为它们自动计算。

说明

要将该变量与 AR_ID 结合使用，报头类型必须组态为 0 或 1。AR_ID 子编号必须设置为零。

在 WinCC V5.0 以前的版本中所组态的所有归档变量均可使用该变量，因为它们没有 AR_ID 子编号。


从 WinCC 版本 5.1 Hotfix 4 开始，可以为过程控制变量的归档变量名指定一个别名，或使用由系统生成的内部名称。由系统生成的名称包含已分配的原始数据变量的名称，而不是来自版本 V5.1 HF4 的原始数据标识号。在从 V5.1 HF4 移植而来的项目中，归档变量名称可以使用其原格式或进行转换。打开过程控制归档变量的属性对话框一次，随后关闭并保存来执行转换。不必指定别名。

如果项目中的所有外部变量都利用“AS-OS 传送”功能重新映射，则必须将归档变量名转换为新结构！然后保持新结构。

要求

- 必须在 AS 中首先组态“AR_SEND”功能块和数据块结构，并且该组态信息在下列过程中是可用的。
- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 必须在通道单元（例如“工业以太网”）中创建连接（例如“Test_Ind_Eth”）。
- 必须在“变量记录”编辑器中组态过程值归档。

步骤

1. 在通道“SIMATIC S7 Protocol Suite”中，选择要用于数据传送的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
为该变量输入名称“Var_raw_arsend”。名称不可超过 24 个字符。当组态过程控制变量时，原始数据变量名成了归档变量名的一部分，因此要考虑到此长度限制。
4. 在“数据类型”(Data type) 字段中，选择“原始数据类型”(Raw Data Type)。
5. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
6. 选中“原始数据”(Raw Data) 复选框。在“原始数据类型”(Raw Data Type) 区域中选择类型“归档数据链接”(Archive Data Link)。单击“确定”(OK) 关闭所有打开的对话框。

13.6 特殊功能

7. 打开“变量记录”(Tag Logging) 编辑器。在“变量记录”(Tag Logging) 编辑器的导航区域, 选择“过程值归档”(Process Value Archives) 文件夹。在表格区域选择“过程控制变量”(Process-controlled tags) 选项卡并添加原始数据变量“Var_raw_arsend”。
8. 在“属性”(Properties) 区域编辑属性。
9. 在“转换 DLL”(Conversion DLL) 字段中选择条目“nrms7pmc.nll”。
10. 在“块 ID”(Block ID) 中输入十六进制数值的 AR_ID。该数值通过 AS 中的组态进行定义。请不要在“子编号”(Subnumber) 复选框内输入任何数字, 因为该 AR_SEND 变量中没有使用子编号。
11. “变量名称”(Tag Name) 显示由系统生成的内部归档变量名称。它包含了分配的原始数据变量的名称和 AR_ID。如果需要, 在“归档变量名称”(Archive Tag Name) 中, 为该归档变量定义一个别名。如果不输入别名, 在 WinCC 中将使用内部归档变量名进行过程值归档期间的管理以及对归档变量进行寻址。
12. 关闭变量记录。

13.6.2.17 如何组态用于多个归档变量的 AR_SEND 变量

简介

有大量的使用 AR_SEND 功能的变量来为多个归档变量交换数据。

- 使用变量“多个归档变量”在不同时刻为多个归档变量提供多个数值。
- 使用变量“多个归档变量 - 已优化”, 同时为最大数目的归档变量每一个都提供一个数值。

这两种变量都使用 AR_ID 和 AR_ID 子编号。

AR_ID 和 AR_ID 子编号在 AS 的数据和归档变量间建立了分配。当组态数据块和功能模块 SFB 37“AR_SEND”时, 它们在 AS 中和其它参数一起定义。

在 WinCC 中，这种分配在过程控制变量的属性中完成。该分配是 WinCC 中所必须的唯一组态步骤，本节将对其进行说明。

WinCC 中其它的参数不需要组态，因为它们自动计算。

说明

为使用 AR_ID 子编号，报头类型必须组态为 8 或 9。

在 WinCC 版本 V5.0 中组态的归档变量没有 AR_ID 子编号，因此只能在“一个变量”中使用。


从 WinCC 版本 5.1 Hotfix 4 开始，可以为过程控制变量的归档变量名指定一个别名，或使用由系统生成的内部名称。由系统生成的名称包含已分配的原始数据变量的名称，而不是来自版本 V5.1 HF4 的原始数据标识号。在从 V5.1 HF4 移植而来的项目中，归档变量名称可以使用其原格式或进行转换。打开过程控制归档变量的属性对话框一次，随后关闭并保存来执行转换。不必指定别名。

如果项目中的所有外部变量都利用“AS-OS 传送”功能重新映射，则必须将归档变量名转换为新结构！然后保持新结构。

要求

- 必须在 AS 中首先组态“AR_SEND”功能块和数据块结构，并且该组态信息在下列过程中是可用的。
- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 必须在通道单元（例如“工业以太网”）中创建连接（例如“Test_Ind_Eth”）。
- 必须在“变量记录”编辑器中组态过程值归档。

步骤

1. 在通道“SIMATIC S7 Protocol Suite”中，选择要用于数据传送的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
为该变量输入名称“Var_raw_arsend”。名称不可超过 24 个字符。当组态过程控制变量时，原始数据变量名成了归档变量名的一部分，因此要考虑到此长度限制。
4. 在“数据类型”(Data type) 字段中，选择“原始数据类型”(Raw Data Type)。
5. 在通道“SIMATIC S7 Protocol Suite”中，选择要用于数据传送的连接。从连接的快捷菜单中选择“新建变量”(New Tag) 选项。将打开“变量属性”(Tag Properties) 对话框。
6. 在“名称”(Name) 字段中输入“Var_raw_arsend”作为变量的名称。名称不可超过 24 个字符。当组态过程控制变量时，原始数据变量名成了归档变量名的一部分，因此要考虑到此长度限制。
7. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。

13.6 特殊功能

8. 选中“原始数据”(Raw Data)复选框。在“原始数据类型”(Raw Data Type)区域中选择类型“归档数据链接”(Archive Data Link)。单击“确定”(OK)关闭所有打开的对话框。
9. 打开“变量记录”(Tag Logging)编辑器。在“变量记录”(Tag Logging)编辑器的导航区域,选择“过程值归档”(Process Value Archives)文件夹。在表格区域选择“过程控制变量”(Process-controlled tags)选项卡并添加原始数据变量“Var_raw_arsend”。
10. 在“属性”(Properties)区域编辑属性。
11. 在“转换 DLL”(Conversion DLL)字段中选择条目“nrms7pmc.nll”。在“块 ID”(Block ID)中输入十六进制数值的 AR_ID。该数值通过 AS 中的组态进行定义。对于“子编号”(Subnumber),将 AR_ID 子编号按十六进制数值输入。该数值也通过 AS 中的组态进行定义。
12. “变量名称”(Tag Name)显示由系统生成的内部归档变量名称。它包含了分配的原始数据变量的名称和 AR_ID。如果需要,在“归档变量名称”(Archive Tag Name)中,为该归档变量定义一个别名。如果不输入别名,在 WinCC 中将使用内部归档变量名进行过程值归档期间的管理以及对归档变量进行寻址。
13. 关闭变量记录。

13.6.3 通道“SIMATIC S7 Protocol Suite”的原始数据变量

13.6.3.1 通道“SIMATIC S7 Protocol Suite”的原始数据变量

引言

- 原始数据类型变量是一种传输层的数据报文。原始数据变量的内容并不固定,因此传送的数据只能由发送方和接收方来解释。该数据类型的 WinCC 中没有更改任何格式。最大长度是 65535 字节。
- WinCC 区分两种类型的原始数据变量:一种是对用户开放的原始数据变量,另一种是用于处理 S7 功能的原始数据变量。

对用户开放的原始数据变量

对用户开放的原始数据变量用于在 WinCC 和 PLC 之间传送用户数据块,并且只处理用户数据。这种原始数据变量区分:

原始数据变量作为字节数组

用于 BSEND/BRCV 功能的原始数据变量

用于处理 S7 功能的原始数据变量

这些原始数据变量没有任何通道专用的报头，通常由消息系统使用，并用于 WinCC 中的过程数据条目。

此处不需要更详细的描述，因为这些都是通道内部的变量和功能。

参见

用于 S7 通信 BSEND/BRCV 功能的原始数据变量 (页 414)

原始数据变量作为字节数组 (页 411)

13.6.3.2 原始数据变量作为字节数组

简介

作为字节数组的原始数据变量用于在 WinCC 和 PLC 之间传送用户数据块，并且只处理用户数据。

作为字节数组的原始数据变量在通道中将和通常的过程变量一样进行处理，通过数据区的地址和长度可对其进行寻址（例如，DB 100、DW 20、长度 40 字节）。

原始数据长度限于一个可传送的数据块，必须能够使用一个 PDU（协议数据单元）完整地传送。使用通信驱动程序能够传送的数据块最大长度取决于建立连接时协商确定的 PDU 长度减去报头以及其它附加信息。因此，SIMATIC S7 中通常使用的 PDU 长度产生下列最大长度：

- S7-300: PDU 长度 240 字节，数据块最大长度 208 字节
- S7-400: PDU 长度 480 字节，数据块最大长度 448 字节

如果要传送更大的数据块，则数据必须分块。在 PLC 中，块由 S7 软件组成；在 WinCC 中，块由脚本组成。

如何将原始数据变量组态为字节数组

用于传送数据块的原始数据变量组态为“发送/接收块”类型的原始数据，带有一个地址和一份长度详细资料。

下图显示了数据组件 100 中从数据字 20 开始的 40 字节长数据区的组态实例：



将原始数据变量读取为字节数组

原始数据变量以与“标准”过程变量相同的方式进行读取。在 AS 中请求相应的数据块，并在接收到数据后传送给用户。

数据传送始终在 WinCC 启动时发生。使用该原始数据变量，不能在 AS 启动时零星地或以事件控制方式接收数据。

将原始数据变量作为字节数组写入

原始数据变量以与“标准”过程变量相同的方式进行写入。发送数据块并从 AS 接收到肯定的确认后，数据块将传送到数据管理器映像。

参见

如何将原始数据变量组态为字节数组 (页 412)

13.6.3.3 如何将原始数据变量组态为字节数组

简介


本节将说明如何将“SIMATIC S7 Protocol Suite”通道的原始数据变量组态为字节数组。

对于通道的所有通道单元，这些组态是完全相同的。本实例中使用了“MPI”通道单元及其连接。

要求

- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 必须创建与通道单元的连接，例如“MPI”。

步骤

1. 在通道“SIMATIC S7 Protocol Suite”中，选择要用于数据传送的连接。
2. 单击表格区下方的“变量”(Tags)选项卡。
3. 单击“名称”(Name)列最上方的空单元格。
为该变量输入名称“Var1_raw_byte”。
4. 在“数据类型”(Data Type)字段中，选择“原始数据类型”(Raw data type)。
5. 打开“地址属性”(Address properties)对话框。
为此，请单击“地址”(Address)字段，然后单击  按钮。
6. 选中“原始数据”(Raw Data)复选框。在“原始数据类型”区域中选择类型“发送/接收块”。该设置将决定“地址描述”(Address description)区域中的字段显示以及“原始数据”(Raw data)复选框旁的字段显示。
7. 在“长度”域中输入原始数据块的长度（以字节为单位）。
8. 在“数据区”(Data area)中，设置PCL中数据所处的数据区。如果在数据区选择了“DB”，则在激活的“DB号”(DB No.)字段中输入数据块号。
“Cpu”字段被禁用以连接“MPI”通道单元。



9. 在“寻址”域中设置寻址类型。如果 WinCC 变量的数据类型是“原始数据类型”，可以使用条目“字节”、“字”或“双字”。
10. 在下面的域中，输入起始地址的值。左侧的域标签取决于“数据区”和“地址”域中的条目，例如数据域选择“DB”而地址域选择“字节”，则此域的标签显示为“DBB”。
11. 单击“确定”(OK) 关闭所有打开的对话框。

说明

原始数据长度限于一个可传送的数据块，必须能够使用一个 PDU（协议数据单元）完整地传送。使用通信驱动程序能够传送的数据块最大长度取决于建立连接时协商确定的 PDU 长度减去报头以及其它附加信息。因此，SIMATIC S7 中通常使用的 PDU 长度产生下列最大长度：

- S7-300：PDU 长度 240 字节，数据块最大长度 208 字节
 - S7-400：PDU 长度 480 字节，数据块最大长度 448 字节
- 长度错误将导致读/写作业被拒绝，同时会显示出错消息。
-

13.6.3.4 用于 S7 通信 BSEND/BRCV 功能的原始数据变量

简介

用于“BSEND/BRCV”功能的原始数据变量将用在 WinCC 和 PLC 之间传送用户数据块，并且只处理用户数据。

该原始数据类型可用于访问 S7 通信“BSEND/BRCV”功能。这些函数可用于以下自动化系统：

- S7-400
- S7-300
 - CPU319-3 PN/DP V2.5 和更高版本
 - CPU317-2 PN/DP V2.6 和更高版本
 - CPU315-2 PN/DP V3.1 和更高版本对于 S7-300 控制器，推荐使用固件版本 V3.x 或更高版本。
- WinAC RTX 2010

数据传送的启动总是由传送伙伴负责；因此，“BSEND/BRCV”功能也可以用于以事件控制方式或零星地传送数据块。

由于资源的原因，建议使用较少数量的 BSEND/BRCV 原始数据变量。

使用 S7 函数“AR_SEND”和“BSEND/BRCV”的资源限制

使用 AR_SEND 和/或 BSEND/BRCV 功能同时从 AS 传送到 WinCC 的最大数据量受到如下限制:

- 对于 S7-400 为 16 KB
- 对于 S7-300 为 8 KB

有关 S7-400 的示例:

- 1x BSEND 最多为 16 KB
- 1x AR_SEND (8kB) + 1x BSEND (8kB)
- 1x AR_SEND (10 KB) + 1x AR_SEND (2 KB) + 1x BSEND (4 KB)

说明

如果写入工作的数据块传送给 AS, 并且还没有从接收缓冲器删除或完全删除, 那么下一步的写入工作将被拒绝, 同时会显示出错消息。显示这样的错误时, R_ID > 0x8000 0000 的写入工作会写入连接专用的队列, 而且系统会尝试重复该写入工作, 时间持续 6 秒钟。用户负责传送的时间协调, 并且需要为写入工作指定较短的时间间隔。

组态 PBK 连接以使用“BSEND/BRCV”功能

“BSEND/BRCV”功能只能通过一个称为 PBK 连接(可编程组件通信)的“硬组态连接”来使用。要组态硬组态连接, 连接资源(十六进制: 10 ... DF)必须在连接参数中给出。在 PLC 内组态连接时, 将由 STEP7 分配该连接资源。该连接必须组态为自动化系统中的被动连接端点。

硬组态连接也可用于处理“普通”读和写作业。如果要通过该连接传送非常大的数据区, 则数据块要使用多个 PDU 进行传送。由于性能原因, 建议创建 BSEND/BRCV 功能专用连接。

组态用于 BSEND/BRCV 功能的原始数据变量

用于传送“BSEND/BRCV”数据块的原始数据变量将被组态为“BSEND/BRCV”类型且带有“R_ID”的原始数据。在发送或接收的数据量中隐含地给出了数据长度。



“R_ID” 参数

必须以十六进制数字给出 32 位长的 R_ID，以便利用“BSEND/BRCV”功能。R_ID 在 AS 中组态时指定，并且用于区分一个连接上的多个数据块传送。与下面的通信子系统（SIMATIC 设备驱动程序）中的 R_ID 有关的发送和接收调用始终会进行通知。所以总是给原始数据变量分配一个唯一的 R_ID。

发送“BSEND/BRCV” 原始数据变量

原始数据变量“BSEND/BRCV”发送的方法与“普通”过程变量写入的方式一样。发送数据块并从 AS 接收到肯定的确认后，数据块将传送到数据管理器映像。

接收“BSEND/BRCV” 原始数据变量

“BSEND/BRCV”原始数据将根据 AS 的触发进行发送。因此，直接读 S7 原始数据变量是不可能的。

BSEND/BRCV 功能不包含任何同步功能。如果在启动阶段没有用户登录以接收数据，由 AS 发送的数据块将在接收方跳动。因此，用户需注意同步，例如，通过在数据字中设置标记来释放 AS 上的发送方向。

参见

如何组态“BSEND/BRCV”功能的原始数据变量 (页 417)

13.6.3.5 如何组态“BSEND/BRCV”功能的原始数据变量

简介


本节将说明如何组态用于“BSEND/BRCV”功能的“SIMATIC S7 Protocol Suite”通道原始数据变量。

对于通道的所有通道单元，这些组态是完全相同的。本实例中使用了“MPI”通道单元及其连接。

要求

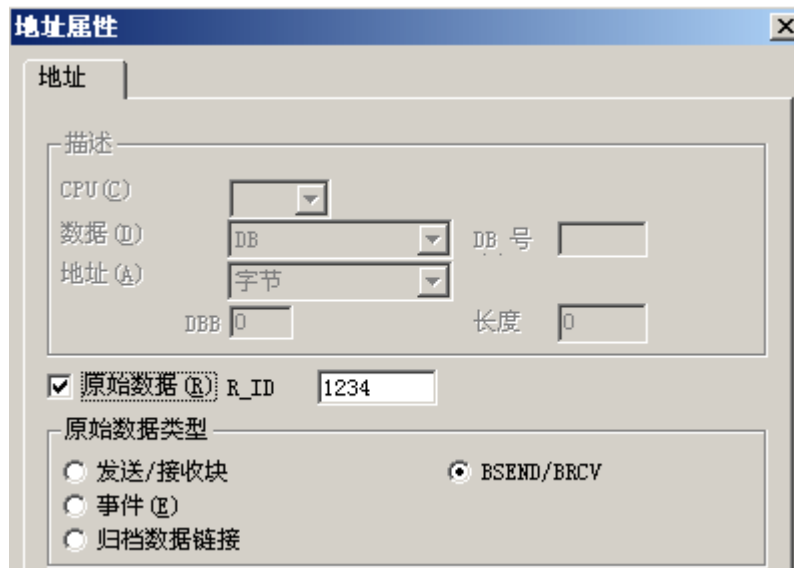
- 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
- 必须创建与通道单元的连接，例如“MPI”。

步骤

1. 在通道“SIMATIC S7 Protocol Suite”中，选择要用于数据传送的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
为该变量输入名称“Var2_raw_bsend”。
4. 在“数据类型”(Data Type) 字段中，选择“原始数据类型”(Raw data type)。
5. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
6. 选中“原始数据”(Raw Data) 复选框。在“原始数据类型”区域中选择类型“BSEND/BRCV”。“地址描述”区域中的域的显示现在被取消激活。

13.6 特殊功能

7. 将 ID 的十六进制数输入到“R_ID”域。组态时在 AS 中指定 R_ID



8. 单击“确定”按钮，关闭上述两个对话框。

13.6.4 软件冗余

13.6.4.1 软件冗余

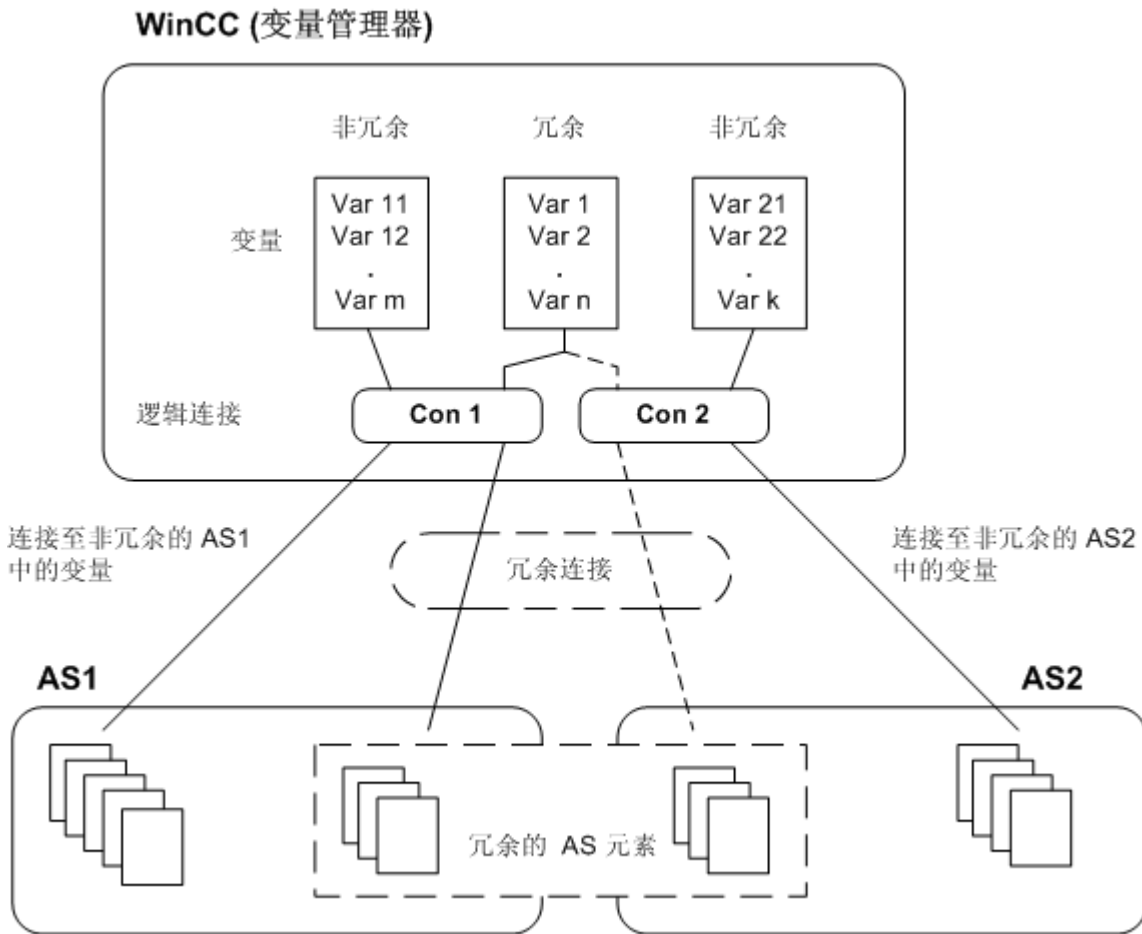
引言

软件冗余通过两个自动化系统 S7-300 或 S7-400 的冗余连接，提供一种经济的方式来监控系统中与安全性相关的、无时间紧迫性的部分。

说明

该通道的软件冗余不具有与 SIMATIC S7-400 H 的 H 层冗余相同的功能。

需要 AS 中的组态和 WinCC 中的组态来实现功能。



AS

如果两个自动化系统之间存在冗余连接，那么如果一个 AS 产生故障，另一个自动化系统可以接管监控功能。监控会覆盖整个过程或只是其中部分过程。

除了应用程序外，两个自动化系统上还安装了软件冗余程序包。该程序包不包括在 WinCC 交付范围内。为了匹配数据，需要通过 MPI、PROFIBUS-DP 或工业以太网，甚至可以使用已存在的通讯链接，在两个自动化系统间建立冗余连接。

WinCC

也可在相同通道单元的连接中组态冗余连接。组态时，只组态一个称为主连接的连接。只有通过动态向导“建立冗余连接”才能插入备用连接。

该向导同时还插入连接专用的内部变量和脚本。它控制运行期间连接之间的切换，以及相应的消息。

运行期间，可使用脚本在出错时在连接之间自动进行切换。但也可以不用脚本，而通过描述连接专用的内部变量“@<connectionname>@ForceConnectionAddress”来手动切换。

13.6 特殊功能

对于 AS2，如果要使用的数据来自 WinCC 中 AS2 的非冗余部分，则需要第二个固定组态连接。

使用软件冗余并不意味着只能将硬组态连接用于冗余。各个单个连接即使没有建立冗余，也都可以使用。

参见

如何在 WinCC 中清除软件冗余 (页 426)

如何组态软件冗余 (页 424)

软件冗余 - 连接专用的内部变量 (页 420)

13.6.4.2 软件冗余 - 连接专用的内部变量

引言

连接专用的内部变量用于控制冗余连接。这些变量使用动态向导“冗余连接组态”进行建立，并被集中到相关连接中的称之为“@<连接名称>”的变量组中。

这些变量可用于确定连接状态和控制连接的建立。因此，它们也可用于其它用途，例如，通过单个连接对多个 PLC 进行寻址。然而，如果在没有软件冗余程序包时要使用这些变量，则必须手工创建。

名称格式

连接专用内部变量的名称由关联连接的名称和标识符组成。

“@<连接名称>@<标识符>”

连接名称的前缀为“@”，用于将其识别为系统变量。连接名称前放置有一个标识符作为分隔符。

实例：“@CPU_3@ConnectionState”

<连接名称> = CPU_3

<标识符> = ConnectionState

说明

连接指定的内部变量作为外部变量计数（每个连接有 8 个外部变量）。

当相关的连接已准备好时，WinCC 的数据管理器只允许访问外部变量。然而，不管连接状态如何，都可以写入和读取指定连接的内部变量。

在运行系统中，可以使用“WinCC 通道诊断”工具调用一些连接专用内部变量的当前值。当选择了主连接时，变量将显示在“计数器”列中。此外，在 WinCC 项目管理器的变量管理器中，可以将变量的当前值显示为工具提示。

变量概述

下列标识符可用于软件冗余的内部变量：

ConnectionState

含义	连接状态 该变量可用于确定当前的连接状态。
类型	双字
访问	读
缺省值	0 = “错误”
数值	0 = 连接错误 1 = 连接准备就绪 2 = 连接冗余（只用于 H 系统出现冗余时）

ConnectionError

含义	错误原因 该变量包含对错误原因的描述。缺省 = 0，也就是说还没有建立连接或无错。建立连接时，再次给变量加载 0（无错）。从通道角度解释错误代码。S7 通道通过该变量传递 SIMATIC 设备驱动程序的错误代码。
类型	双字
访问	读
缺省值	0 = “无错”
数值	0 = 无错误 <> 0 = S7 错误代码

ConnectionErrorString

含义	错误原因字符串 该变量包含连接错误原因字符串。字符串以目前选定的语言输出。缺省 = “”，也就是说还没有建立连接或无错。在 S7 通道中，下列文本都以“英语”的方式输出，与所选的语言无关。
类型	TEXT8 [128]
访问	读
缺省值	“” = “无错”
数值	“No Error” “Error hhhh”= 产生错误 hhhh（这里 hhhh = 十六进制的 S7 错误代码）

ConnectionErrorCount

含义	连接错误计数器 每发生一次连接错误，该变量的值增加 1。溢出后，从 0 开始计数。
类型	双字
访问	读
缺省值	0

ConnectionEstablishMode

含义	连接建立模式 该变量可用于设置建立连接的自动模式。如果将其启用，每隔 4 秒左右，S7 通道试图重新建立失败了的连接。如果该变量的值等于 0，则在故障出现之后，不会在每 4 秒之后尝试重新建立连接，而是保持断开状态。
类型	双字
访问	写
缺省值	1
数值	写入变量会引起以下动作： 0 = 手动连接建立模式 动作：取消激活自动连接建立 <>0 = 自动连接建立模式 动作：激活自动连接建立模式

ForceConnectionState

含义	首选连接状态 该变量可用于通知通道首选连接的状态。通常该变量取数值 1，也就是说通道试图建立连接（如果适用，间隔大约为 4 秒）。如果写入变量的数值是 0，通道中断连接。
类型	双字
访问	写
缺省值	1
数值	写入到变量的数值具有下列含义： 0 = 首选连接状态： 中断建立 动作： 如果已建立连接，则启动断开连接 1 = 优先考虑的连接状态： 中断建立 动作： 如果已断开连接，则启动建立连接

ForceConnectionAddress

含义	选择连接地址 该变量定义将要用于建立连接的连接地址。
类型	双字
访问	写
缺省值	0
数值	如果 ConnectionEstablishMode 设置为“自动”，则将自动建立到相应地址的连接。 写入变量会引起以下动作： 0 = 通过已组态的连接参数进行连接 动作： 如果 @ForceConnectionAddress 已设置为 1，则启动断开连接。 1 = 通过其它连接参数进行连接 动作： 如果 @ForceConnectionAddress 先前为 0，则启动断开连接。

AlternateConnectionAddress

含义	其它连接地址 在该变量中，可以输入其它连接地址字符串。该字符串与将在 WinCC 项目管理器中显示的连接参数字符串相同。该字符串专用于通道。系统启动（运行时）时，此处输入组态的地址，作为 S7 通道的缺省设置。如果还没有组态地址，则为 S7 通道输入文本“非法地址”。 通过 MPI 且站地址为 3 的 S7-AS 的详细地址实例：“MPI,3 0,,0,0,02”
类型	TEXT8 [255]
访问	写
缺省值	“...” = 组态的地址
数值	写入变量会引起以下动作： - 如果由于写操作而更改了地址，则连接断开并返回设置“通过其它连接参数进行连接”。 - 如果连接模式设置为“自动”，则根据刚写入的地址自动建立连接。

13.6.4.3 如何组态软件冗余**简介**

本节说明如何在 WinCC 中为通道“SIMATIC S7 Protocol Suite”的连接组态软件冗余。PLC 也必须进行组态，以便使用该功能，但本文档将不对此进行描述。

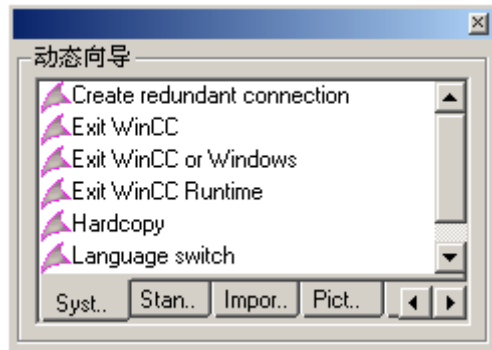
要求

1. 通道“SIMATIC S7 Protocol Suite”必须集成在项目中。
2. 必须在该通道的一个通道单元中创建连接，并为其组态冗余连接。

步骤

1. 在计算机的启动参数中，激活“全局脚本运行系统”、“报警记录运行系统”和“图形运行系统”模块。
有关详细信息，请参见“检查启动参数”。
2. 将 WinCC 系统消息载入报警记录。这些系统消息中包含有关软件冗余的消息。
可以在“读取报警记录中的 WinCC 系统消息”下找到有关该主题的附加信息。

- 在图形编辑器中打开画面。在“动态向导”窗口中，选择“系统功能”标签。双击以启动动态向导“设置冗余连接”。



- 向导的使用步骤在“欢迎”中有简短的描述。单击“下一步”，打开“设置选项”对话框。
- 选择将用作主连接的连接，然后单击“下一步”。
向导现在将创建连接专用的内部变量，并将其保存到主连接下的变量组“@”中。
- 输入 PLC 的地址，在“参数”域中将建立与该 PLC 之间的备用连接。
选中“自动切换”复选框以使向导为自动切换连接生成脚本。
单击“下一步”
冗余的图形（显示与两个 PLC 的 MPI 连接）将出现在画面中：



- 在“完成！”对话框中，将再次显示所进行的全部设置。如果要进行任何更正，只需单击“返回”即可。单击“完成”。
向导随即会生成一个脚本，并将其保存在全局脚本编辑器的目录“C-Editor \ Actions \ Actions : <计算机名称>”中的“@<连接名称>.pas”下。

说明

在下面的步骤中，将使用“建立冗余连接”向导。当完成步骤 6 时，该向导生成连接专用的内部变量。如果此时向导被取消或没有以单击“完成”来结束过程，则这些变量将保持不变。

参见

如何将 WinCC 系统消息载入报警记录 (页 427)

如何检查 WinCC 启动参数 (页 427)

13.6.4.4 如何在 WinCC 中清除软件冗余**引言**

本节说明如何在 WinCC 中为通道“SIMATIC S7 Protocol Suite”的连接删除软件冗余。PLC 也必须进行组态，以便使用该功能返回到非冗余连接，但本文档不对此进行描述。

要求

- 必须取消激活 WinCC 项目。

步骤

可按两个步骤删除软件冗余：

- 在“变量管理器”中删除变量组“@<计算机名称>”及其变量。
- 在“全局脚本”中删除脚本“@<连接名称>.pas”。

步骤

1. 在变量管理器中，为软件冗余选择将组态为主连接的连接。它包括变量组“@<连接名称>”及软件冗余的连接专用的内部变量。删除该变量组。
2. 删除用于动作“@<连接名称>.pas”的脚本。为此，打开“全局脚本”中的 C 编辑器。则会显示多个子目录。
3. 选择“Actions \ Action : " directory. <计算机名称>”。在数据窗口中，删除“动作”类型的脚本“@<连接名称>.pas”。
4. 关闭“全局脚本”编辑器。

13.6.4.5 如何检查 WinCC 启动参数

步骤

1. 在 WinCC 项目管理器中，选择“计算机”。
2. 打开快捷菜单并选择“属性”。
3. 将打开“计算机列表属性”对话框。单击“属性”按钮。
4. 将打开“计算机属性”对话框。
5. 单击“启动”标签并选中条目。必要时，可以激活或取消激活运行模块，或将其它应用程序添加到启动列表中。
6. 关闭打开的对话框。

13.6.4.6 如何将 WinCC 系统消息载入报警记录

简介

本节说明如何将 WinCC 系统消息载入项目。

步骤

1. 打开报警记录。
2. 在导航区域中选择“系统消息”节点。
3. 可以针对在表格区域或“属性”(Properties) 区域中使用的系统消息激活“已使用”(Used) 选项。
4. 在“系统消息”节点的快捷菜单中选择“已使用更新”(Update used) 命令。

13.6.4.7 当连接受干扰时的错误代码

13.6 特殊功能

SIMATIC S7-1200, S7-1500 Channel

14.1 “SIMATIC S7-1200, S7-1500 Channel” 通道

简介

"SIMATIC S7-1200, S7-1500 Channel" 用于 WinCC 站与自动化系统 S7-1200 和 S7-1500 之间的通信。通信采用 TCP/IP 协议。

通道单元

"SIMATIC S7-1200, S7-1500 Channel" 具有“OMS+”通道单元。

可以在“通信诊断”下找到有关通道诊断的详细信息。

建议的通信处理器

在 WinCC 站与自动化系统 S7-1200 或 S7-1500 进行通信时，建议您使用下列通信处理器：

- CP 1612 A2
- CP 1613 A2
- CP 1623
- CP 1628

在运行系统中建立或终止连接

要定义 WinCC 站和 S7-1200 / S7-1500 之间的状态，在连接中创建二进制变量“@<连接名称>@ForceConnectionState”：

- “1”：已建立连接。
当组态“1”作为起始值时，启动运行系统时建立和 S7-1200/S7-1500 的连接。

- “0”：连接被终止。

使用针对连接的变量“@<connection name>@ConnectionState”确定当前连接的状态：

- “1”：连接准备就绪。
- “0”：连接被中断或终止。

14.2 所支持的数据类型概述

简介

在组态变量时指定数据类型以及针对自动化系统 (AS) 中的数据格式的格式调整。

下表显示了通道支持的数据类型和类型转换的使用。

支持的数据类型/格式调整

数据类型	类型转换
二进制变量	否
有符号 8 位数	是
无符号 8 位数	是
有符号 16 位数	是
无符号 16 位数	是
有符号 32 位数	是
无符号 32 位数	是
浮点数 32 位 IEEE 754	是
64 位 IEEE 754 浮点数	是
文本变量, 8 位字符集	否
文本变量, 16 位字符集	否
原始数据变量	否
日期/时间	是

14.3 组态通道

14.3.1 组态“SIMATIC S7-1200, S7-1500 Channel”通道

简介

在 WinCC 中，WinCC 与自动化系统的通信需要逻辑连接。本节介绍如何组态 "SIMATIC S7-1200, S7-1500 Channel"。

TCP/IP 连接

使用 TCP/IP 协议时，必须定义自动化系统的 IP 地址以实现逻辑连接。IP 地址由四个以圆点分隔的数值组成。数值必须位于“0-255”的范围内。

说明

超时行为

使用 TCP/IP 协议时，不会立即检测到中断的连接。回查消息可能耗时达一分钟。

组态变量

对于通过 "SIMATIC S7-1200, S7-1500 Channel" 实现的 WinCC 和自动化系统之间的连接，可在 WinCC 中创建不同数据类型的变量。

变量的组态根据在自动化系统数据区中的寻址而定。

说明

只能在 TIA Portal 中对自动化系统的组态进行更改。

组态原始数据变量

"SIMATIC S7-1200, S7-1500 Channel"通道支持“原始数据变量”数据类型。

作为字节数组的原始数据变量用于在 WinCC 和 PLC 之间传送用户数据块，并且只处理用户数据。

原始数据变量仅支持控制器非循环读取服务，例如：通过 C 脚本请求变量。

“SIMATIC S7-1200, S7-1500 Channel”通道不支持原始数据变量的循环读取服务。

14.3 组态通道

原始数据变量寻址

作为字节数组的原始数据变量在通道中将和通常的过程变量一样进行处理，通过数据区的地址和长度可对其进行寻址（例如，DB 1、DBB 10、长度 100 字节）。

如果 WinCC 变量的数据类型是“原始数据变量”，仅可使用“字节”。除原始数据范围长度外，参数均为预设值且不可更改。

数据块的长度

请遵守可由通信驱动程序发送的最大数据块长度：

- S7-1200 / S7-1500：数据块长度最大为 8000 字节

14.3.2 如何组态连接

简介

组态 "SIMATIC S7-1200, S7-1500 Channel" 的必要步骤为：

1. 组态连接
2. 组态变量

要求

- "SIMATIC S7-1200, S7-1500 Channel" 的通信驱动程序已安装并集成到项目中。
- SIMATIC 项目已组态，并且在自动化系统中可用。

步骤

1. 在 WinCC 项目管理器的“变量管理”编辑器中打开 "SIMATIC S7-1200, S7-1500 Channel" 通信驱动程序的菜单结构。
2. 在通道单元“OMS+”的快捷菜单中选择“新建连接”(New connection) 条目。
3. 输入连接的名称。
4. 在连接快捷菜单中选择“连接参数”(Connection parameters) 条目。
“新建连接”(New connection) 对话框随即打开。
5. 输入自动化系统的 IP 地址。
6. 选择“访问点”(Access point)。
7. 从“产品系列”(product family) 中选择自动化系统。
8. 如果要使用密码保护，请单击“选项”(Options)。

9. 输入访问保护的密码。
10. 单击“确定”(OK)按钮，关闭上述两个对话框。

14.3.3 如何组态没有优先块访问的变量


简介

本节介绍如何在 WinCC 中组态没有对自动化系统中地址区域进行优先块访问的变量。

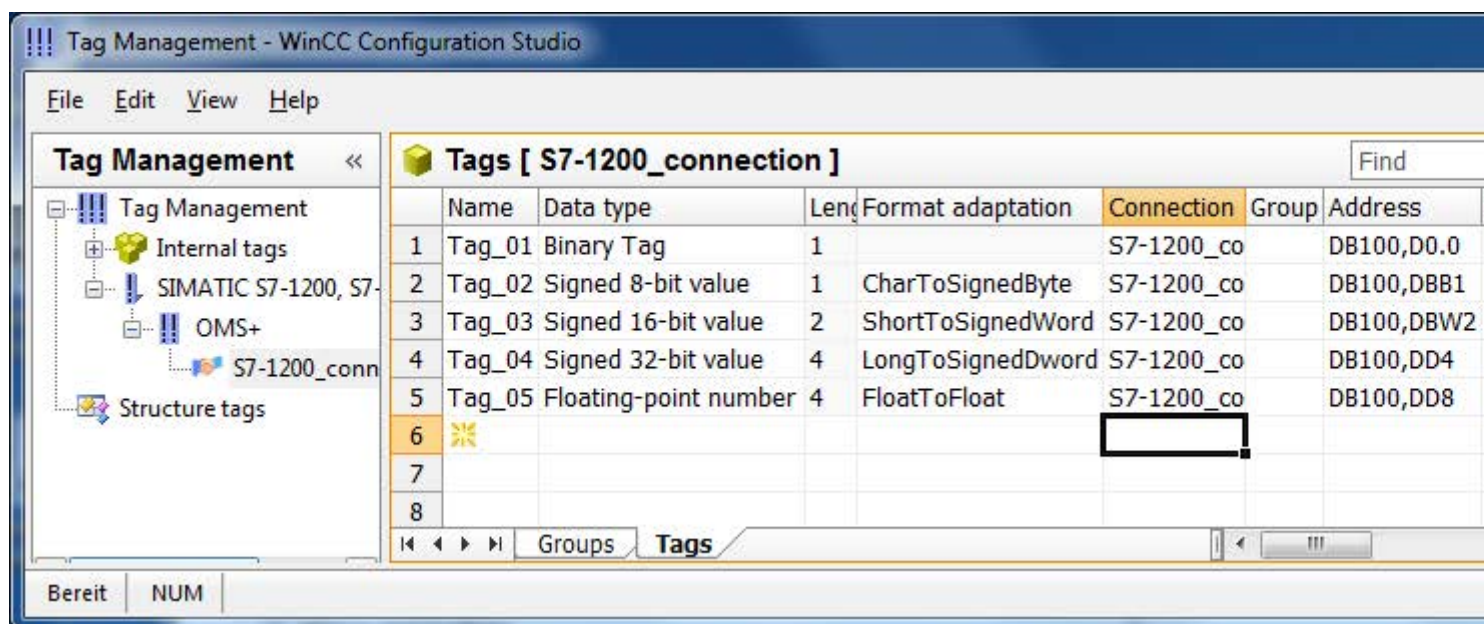
要求

- 在 TIA Portal 中将数据块的属性“优先块访问”(Optimized block access) 禁用。
- "SIMATIC S7-1200, S7-1500 Channel" 必须已集成到项目中。
- 必须在“OMS+”通道单元中创建了连接。

步骤

1. 选择所需连接。
2. 单击表格区下方的“变量”(Tags)选项卡。
3. 单击“名称”(Name)列最上方的空单元格。输入变量的名称。
在表格区或右侧的“属性 - 变量”(Properties - Tags)数据区组态以下设置。
4. 选择一种支持的数据类型。
5. 在“地址”(Address)列单击  按钮。
6. 输入变量地址。
7. 如果变量具有质量代码并且您要在 WinCC 中使用该代码，则选中“质量代码”(Quality code)复选框。该代码还必须存在于自动化系统中。只有选择了“DB”数据区才能激活此复选框。
8. 单击“确定”(OK)关闭对话框。

下图显示的是完成无优先块访问的变量组态后的变量管理。



组态 8 位文本变量的注意事项

对于 "SIMATIC S7-1200, S7-1500 Channel" 中的 8 位文本变量，WinCC 仅支持 S7 字符串类型，该类型由一个控制字和用户字符串数据组成：

- 要在 WinCC 中组态 8 位文本变量，请在用户数据之前输入自动化系统 (AS) 存储器中存在的控制字的地址。控制字的第一个字节包含字符串自定义的最大长度，第二个字节包含实际长度。
- 在自动化系统存储器中创建数据结构时，必须注意 WinCC 中为 8 位文本变量组态的长度要增加控制字的 2 个字节。如果在存储器中直接逐个创建 8 位文本变量的数据结构，则后面的数据将会被覆盖。
- 在进行读操作时，控制字将和用户数据一起被读取，并将判断第二个字节中的当前长度。只有长度与第二控制字节中包含的当前长度一致的用户数据才传送到 WinCC 的 8 位文本变量。
- 在进行写操作时，字符串的实际长度是确定的（“0” 字符），且具有当前长度的控制字节将和用户数据一起发送给自动化系统。

14.3.4 如何组态具有优先块访问的变量

简介

本节介绍如何在 WinCC 中组态具有对自动化系统中地址区域进行优先块访问的变量。

将变量从控制器导入 WinCC 项目中。

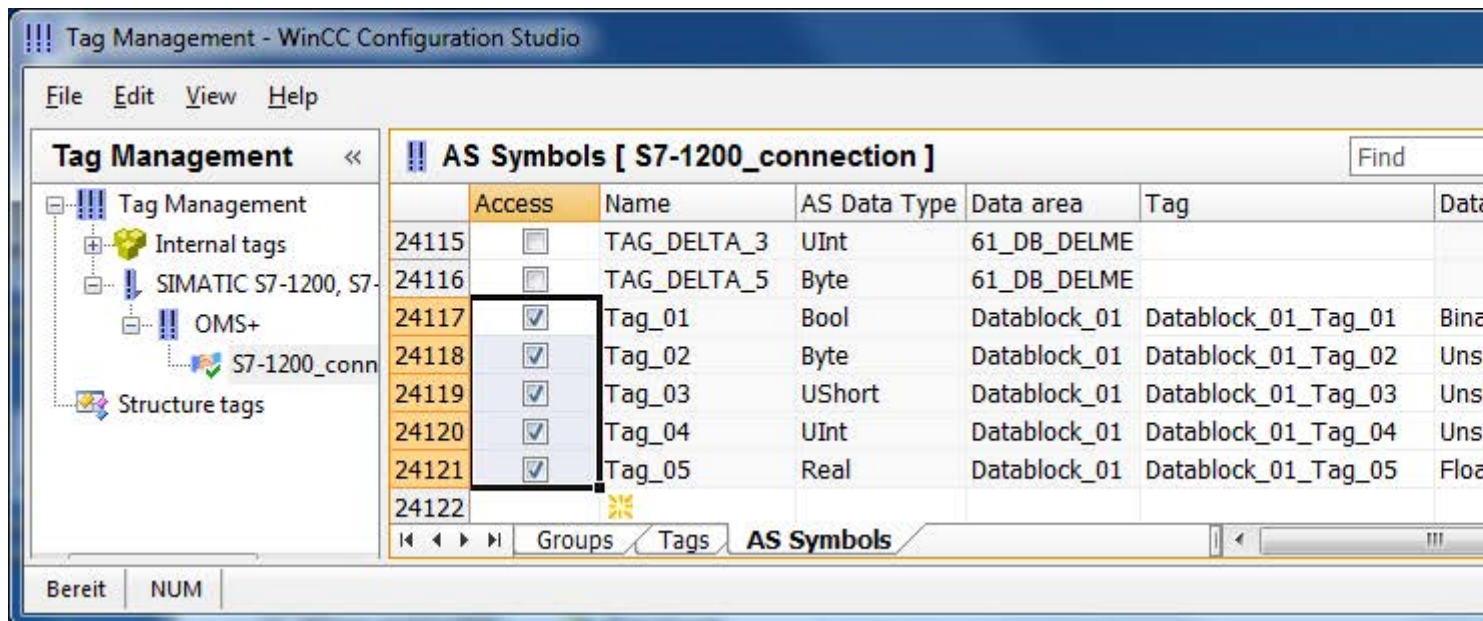
要求

- 在 TIA Portal 中将数据块的属性“优先块访问”(Optimized block access) 激活。
- "SIMATIC S7-1200, S7-1500 Channel" 必须已集成到项目中。
- 必须在“OMS+”通道单元中创建了连接。
- 必须在运行系统中建立了连接。

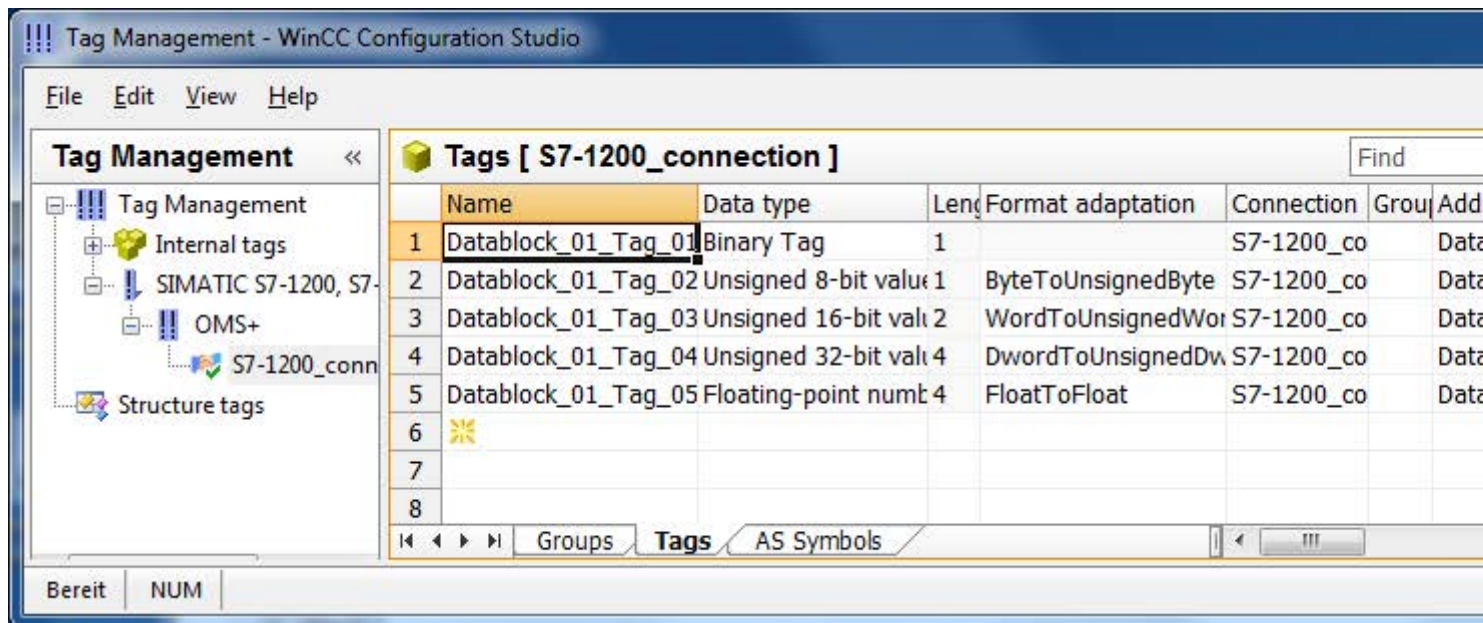
14.3 组态通道

步骤

1. 选择所需连接。
2. 在连接快捷菜单中选择“AS 符号 > 从 AS 读取”(AS Symbols > Read from AS)。所有控制器的可用数据都被加载到“AS Symbols”(AS 符号) 选项卡的表格区中。



3. AS 符号不会自动包含到变量管理中。要将所需 AS 符号传送到“变量”(Tags) 选项卡，需激活“访问”(Access) 列中相应的复选框。变量管理中不包含选定的变量。



如何在不连接控制器的情况下编辑 AS 符号

用户可在不连接控制器的情况下对其进行离线组态。为此，请将已加载的 AS 符号保存到文件中。

1. 选择所需连接。
2. 在连接快捷菜单中选择“AS 符号 > 保存到文件”(AS Symbols > Save to file)。

然后即可将 AS 符号加载到变量管理的离线项目中。

1. 选择所需连接。
2. 在连接快捷菜单中选择“AS 符号 > 从文件加载”(AS Symbols > Load from file)。

使用控制器同步 WinCC 变量

从控制器或文件进行加载后，变量管理检查 AS 符号的属性。

地址、数据类型和变量名和 WinCC 项目中的 AS 符号属性进行比较。

- 如果属性不匹配，“AS 符号”(AS Symbols) 选项卡中的“访问”(Access) 字段用红色突出显示。
字段的工具提示包含额外信息。

- 如果未在控制器中找到 WinCC 变量，连接的 AS 符号的整行用红色突出显示。

例如，在下列情况下会发生此问题：

- 使用 WinCC V7.3 创建 WinCC 项目。移植的项目还未包含所有同步的信息。
- AS 符号的地址已在控制器中改变，例如，由于 TIA Portal 的组态变更。
- AS 符号的数据类型或名称已被更改。
- 已在控制器中删除了 AS 符号。

再次导入变量

要同步属性，更新 WinCC 项目中使用的 AS 符号。

按以下步骤执行操作：

1. 在“AS 符号”(AS Symbols) 选项卡中选择带红色背景的 AS 符号。
要更新移植项目，选择所有行。
2. 在“访问”(Access) 域的快捷菜单中选择“更新”(Update)。

14.3 组态通道

AS 组态的参数被再次导入。

说明

移植 TIA Portal 项目前：更新 AS 符号

更新 TIA Portal 版本时，请遵循以下顺序进行操作：

1. 更新所有用作 WinCC 变量的 AS 符号。
2. 移植 TIA Portal 项目。
3. 将控制器加载到 TIA Portal 中。
4. 再次更新所有用作 WinCC 变量的 AS 符号。

该操作确保进行加载后，WinCC 变量到 AS 符号的分配保持在 WinCC 项目中。

SIMATIC TI Ethernet Layer 4

15.1 WinCC 通道“SIMATIC TI Ethernet Layer 4”

简介

通讯驱动程序“SIMATIC TI Ethernet Layer 4”处理 WinCC 站与 SIMATIC TI505 自动化系统之间通过工业以太网进行的连接。其通讯使用 ISO 传输协议处理。

本章描述了

- 如何使用“SIMATIC TI Ethernet Layer 4”通道组态数据传送。
- 如何组态连接和变量。

通道单元

IT 具有两个通道单元，最多可运行两个 CP 1613。通道单元的功能完全相同。两个 CP 1613 之间唯一的不同之处就是其逻辑设备名称不同。

逻辑设备名称可通过通道单元的系统参数进行更改。还可以在此设置 ISO 传输协议的参数。

存在下列应用能力：

- 通道单元 505 Ethernet (CP 1413-1)，适用于 SIMATIC 工业以太网的通讯模块（例如，CP 1613）。
- 通道单元 505 Ethernet (CP 1413-2)，适用于 SIMATIC 工业以太网的通讯模块（例如，CP 1613）。

15.2 变量的数据类型

简介

为逻辑连接定义所需变量。从 WinCC 的角度而言，可以访问以下数据类型：

- 二进制变量
- 无符号 8 位数（仅由 VMS 寻址支持）
- 有符号 8 位数（仅由 VMS 寻址支持）
- 无符号 16 位数
- 有符号 16 位数
- 无符号 32 位数
- 有符号 32 位数
- 浮点数 32 位 IEEE 754
- 原始数据类型

15.3 组态通道

15.3.1 组态通道“SIMATIC TI Ethernet Layer 4”

简介

组态通道“SIMATIC TI Ethernet Layer 4”需要以下步骤。

15.3.2 如何组态连接

简介

使用 SIMATIC TI505 自动化系统可实现通过工业以太网进行过程连接。自动化系统中使用通信模块 CP 1434 TF。其通信使用 ISO 传输协议处理。

WinCC 系统中使用通信模块 CP 1613。因为通信是通过 ISO 传输协议进行的，因此无需在本地数据库中组态逻辑连接。

对于逻辑连接，WinCC 在传送层中建立一个用于读取 (READ) 的连接和一个用于写入 (WRITE) 的连接。仅当这两个连接都已建立时，才表示逻辑连接已建立。

读功能的参数

组态连接时，在 WinCC 中为读功能定义参数。这些参数独立于在 SIMATIC TI 中使用的请求。

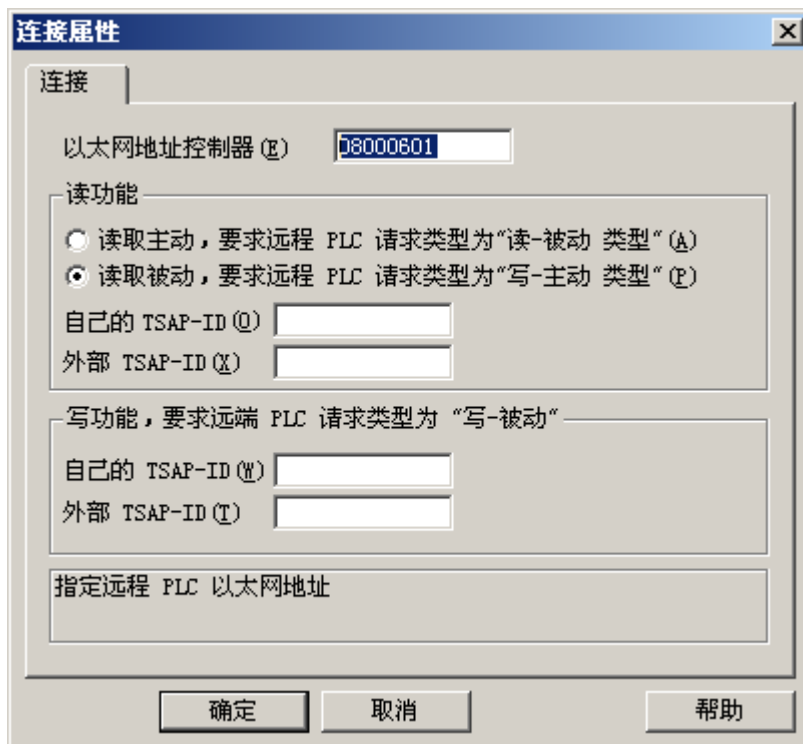
WinCC 端	SIMATIC TI 侧
FETCH-Active (请求“READ-Active”)	READ-Passive (请求“READ-Passive”)
FETCH-Passive (请求“WRITE-Passive”)	WRITE-Active (请求“WRITE-Active”)

写功能的“自身 TSAP-ID”和“外部 TSAP-ID”参数

WinCC 端	SIMATIC TI 侧
请求“WRITE Active”	请求“WRITE Passive”

步骤

1. 选择所需连接，并从快捷菜单中选择“连接参数”(Connection parameters)。“连接属性”(Connection properties) 对话框随即打开。



2. 在“以太网地址”域中输入工业以太网总线上 SIMATIC TI 的站地址。
3. 接下来定义 WinCC 系统中读功能的参数。这些参数独立于在 SIMATIC TI 中使用的请求。
4. 然后，在“自身 TSAP-ID”分配域中输入组态 CP 1434 TF 时在“远程参数”区中组态为“TSAP”的值。
5. 现在，在“外部 TSAP-ID”分配域中输入组态 CP 1434 TF 时在“本地参数”区中组态为“TSAP”的值。
6. 相应地定义写功能的“自身 TSAP-ID”和“外部 TSAP-ID”参数。

15.3.3 组态变量

15.3.3.1 组态变量

简介

对于 WinCC 与 AS 之间通过“SIMATIC TI Ethernet Layer 4”通道进行的连接，可在 WinCC 中创建不同数据类型的变量。下面将说明如何组态这些数据类型的变量。


15.3.3.2 如何组态变量的地址

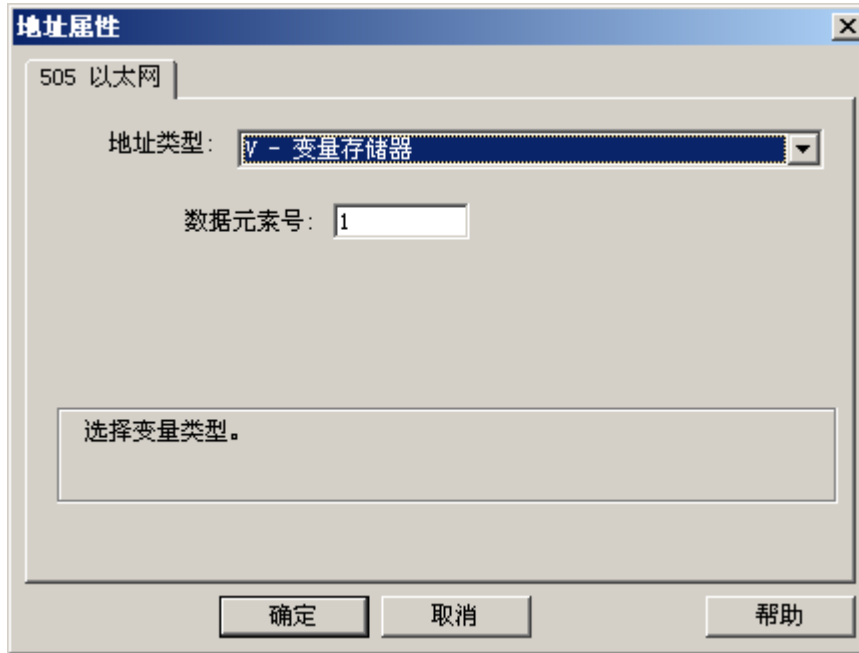
简介

变量地址根据 SIMATIC TI505 的地址结构输入。

步骤

1. 选择变量
2. 在“数据类型”(Data Type) 字段中选择所需的数据类型。
3. 如果是“二元”或“8 位”变量，则在“属性”(Properties) 区域会提供“位/字节变量”(Bit/Byte tag) 选项。如果要写入 AS 存储器，则选中相应的复选框“访问位/字节”(Access a Bit/Byte)。

4. 打开“地址属性”(Address properties)对话框。
为此，请单击“地址”(Address)字段，然后单击  按钮。



5. 在“地址类型”域中选择变量在自动化系统中的位置。视所选地址类型而定，必须进行更多定义（例如，地址类型为“V - 变量存储器”需定义“数据元素号”域）。
6. 在“只读变量”域中，可指定该变量不能由 WinCC 写入。

说明

不支持结构变量。

关于地址类型的描述，可参见 SIMATIC TI505 技术文档。

使用通道“TI Ethernet Layer 4”时只能按位或按字节执行对 AS 存储器区域的写访问。在使用二进制和“8 位”变量时，除了会打开“地址属性”对话框外，还将打开“位/字节变量”对话框，该对话框可用于定义对各个位或字节的写访问。为此，每次写请求时均从 AS 读取已赋址的存储器区域，从而修改相应的位和/或字节。此后，数据写回到 AS 的存储器。

说明

AS 在读取数据区中执行的更改会在写回到该数据区时被覆盖。

视变量类型而定，可以按位或按字节访问 AS 中的存储器。

15.3.3.3 如何组态按位访问变量


简介

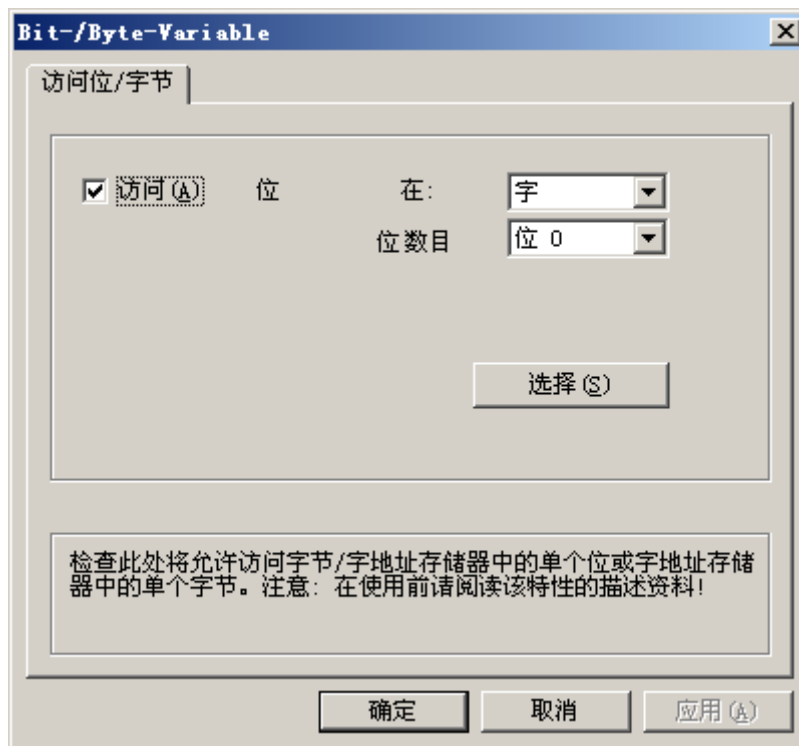
使用通道“TI Ethernet Layer 4”时只能按位或按字节执行对 AS 存储器区域的写访问。在使用二进制和“8 位”变量时，除了会打开“地址属性”对话框外，还将打开“位/字节变量”对话框，该对话框可用于定义对各个位或字节的写访问。为此，每次写请求时均从 AS 读取已赋址的存储器区域，从而修改相应的位和/或字节。此后，数据写回到 AS 的存储器。

说明

AS 在读取数据区中执行的更改会在写回到该数据区时被覆盖。

步骤

1. 选择变量。
2. 在“数据类型”(Data Type) 字段中，设置“二进制变量”(Binary tag) 数据类型。
3. 打开“位/字节变量”(Bit/Byte tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
4. 选择“访问位”(Access a bit) 复选框，并定义该位的地址。



5. 单击“选择”(Select) 按钮。将打开“地址属性”(Address properties) 对话框。

15.3 组态通道

6. 在“地址类型”(Address type) 选择域中，选择 PLC 存储器的寻址类型。
7. 从下方的表中选择待更改元素的编号。

说明

关于地址类型的描述，可参见 SIMATIC TI505 技术文档。

15.3.3.4 如何组态按字节访问的变量


简介

使用通道“TI Ethernet Layer 4”时只能按位或按字节执行对 AS 存储器区域的写访问。在使用二进制和“8 位”变量时，除了会打开“地址属性”对话框外，还将打开“位/字节变量”对话框，该对话框可用于定义对各个位或字节的写访问。为此，每次写请求时均从 AS 读取已赋址的存储器区域，从而修改相应的位和/或字节。此后，数据写回到 AS 的存储器。

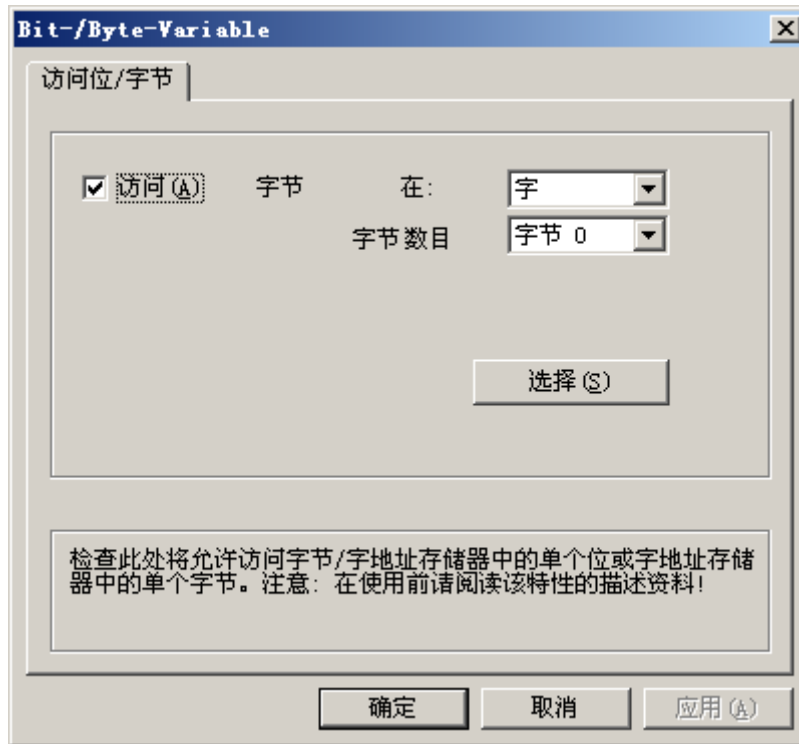
说明

AS 在读取数据区中执行的更改会在写回到该数据区时被覆盖。

步骤

1. 选择变量。
2. 在“数据类型”(Data Type) 字段中，将数据类型设置为“无符号 8 位数”(Unsigned 8-bit value) 或“有符号 8 位数”(Signed 8-bit value)。
3. 打开“位/字节变量”(Bit/Byte tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。

- 选择“访问字节”(Access to a byte) 复选框，并定义该字节的地址。



- 单击“选择”(Select) 按钮。将打开“地址属性”(Address properties) 对话框。
- 在“地址类型”(Address type) 选择域中，选择 PLC 存储器的寻址类型。
- 从下方的表中选择待更改元素的编号。

说明

关于地址类型的描述，可参见 SIMATIC TI505 技术文档。

15.3.4 系统参数

15.3.4.1 通道单元的系统参数

简介

如果需要与 WinCC 标准设置不一样的组态，则可使用通道单元的“系统参数”对话框进行所需的全部更改。

15.3 组态通道

以下各项均可更改：

- 设备名称
- 传输参数

设备名称

WinCC 和自动化系统之间的通讯通过逻辑设备名称执行。这些名称是在安装通讯模块时分配的，并且是单元特定的条目。设备名称代表逻辑设备名称。默认情况下，逻辑设备名称初始定义为“/CP_H1_1:/SCP”。

传输参数

可在传输参数中进行通道单元的特定设置，例如 PDU 大小、尝试建立等。

说明

系统参数适用于 AS 中的所有 CP。

15.3.4.2 如何更改设备名称

简介

使用 SIMATIC TI505 自动化系统可实现通过工业以太网进行过程连接。自动化系统中使用通讯模块 CP 1434 TF。其通讯使用 ISO 传输协议处理。

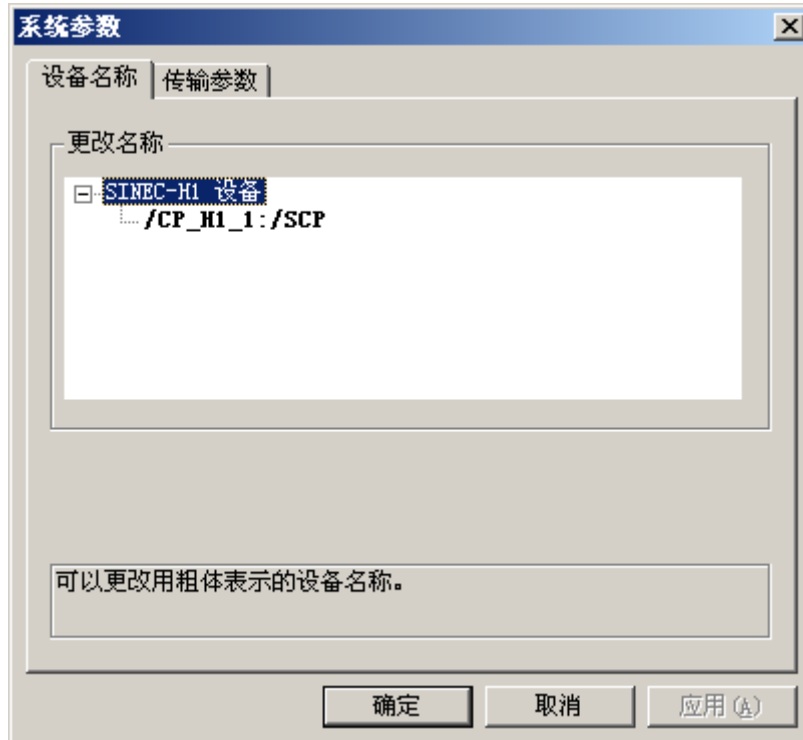
WinCC 系统中使用通讯模块 CP 1613。因为通讯是通过 ISO 传输协议进行的，因此无需在本地数据库中组态逻辑连接。

要求

- 通道“SIMATIC TI Ethernet Layer 4”必须已集成在项目中。

步骤

1. 选择通道单元，然后通过上下文菜单打开“系统参数”对话框。
2. 选择“设备名称”选项卡。



3. 现在，可使用鼠标来选择用粗体表示的逻辑设备名称，并通过在名称域中单击鼠标对其进行更改。
默认情况下，逻辑设备名称在安装硬件驱动程序时定义为“/CP_H1_1:/SCP”。只有在安装硬件驱动程序时定义了其它名称（不推荐），才必须也更改此处的设备名称。

15.3.4.3 如何更改传输参数

简介

使用 SIMATIC TI505 自动化系统可实现通过工业以太网进行过程连接。自动化系统中使用通讯模块 CP 1434 TF。其通讯使用 ISO 传输协议处理。

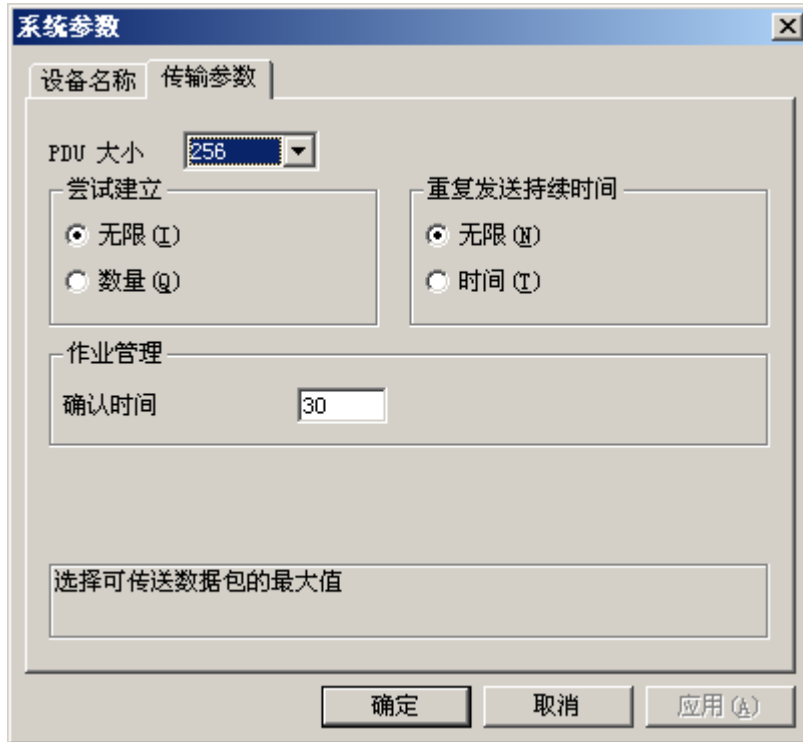
WinCC 系统中使用通讯模块 CP 1613。因为通讯是通过 ISO 传输协议进行的，因此无需在本地数据库中组态逻辑连接。

要求

- 通道“SIMATIC TI Ethernet Layer 4”必须已集成在项目中。

步骤

1. 选择通道单元，然后通过上下文菜单打开“系统参数”对话框。
2. 选择“传输参数”选项卡。



3. 将“PDU 大小”的值设置为在 CP 1434 TF 上所组态的值。
4. 在“建立尝试”域中，定义尝试建立连接的频率。
5. 在“重复发送持续时间”区域中选择“无限”。
6. 在“确认时间”域中输入值 30，这样，如果通讯伙伴在此时间内未作出响应，则会在 30 秒内通知您变量状态（例如，AS 处于“停止”状态）。

SIMATIC TI Serial

16.1 WinCC 通道“SIMATIC TI Serial”

简介

通讯驱动程序“SIMATIC TI Serial”用于建立 WinCC 站与 SIMATIC TI505 自动化设备之间的串行连接。

本章描述了

- 如何组态使用“SIMATIC TI Serial”通道的数据传输。
- 如何组态连接和变量。

通道单元

通讯驱动程序具有一个用于控制串行连接的 COM 端口的通道单元。

提供了以下功能：

- 通道单元“505 Serial Unit #1”，用于通过 TBP 协议或 NITP 协议进行串行通讯。

说明

允许通过一个单元通道运行多个逻辑连接（使用不同的 COM 端口）。

16.2 变量的数据类型

简介

为逻辑连接定义所需变量。从 WinCC 的角度而言，可以访问以下数据类型：

- 二进制变量
- 无符号 8 位数（仅 VMS 寻址支持）
- 有符号 8 位数（仅 VMS 寻址支持）
- 无符号 16 位数
- 有符号 16 位数
- 无符号 32 位数
- 有符号 32 位数
- 浮点数 32 位 IEEE 754
- 原始数据类型

16.3 组态通道

16.3.1 组态“SIMATIC TI Serial”通道

简介

组态通道“SIMATIC TI Serial”需要以下步骤。

16.3.2 如何组态连接

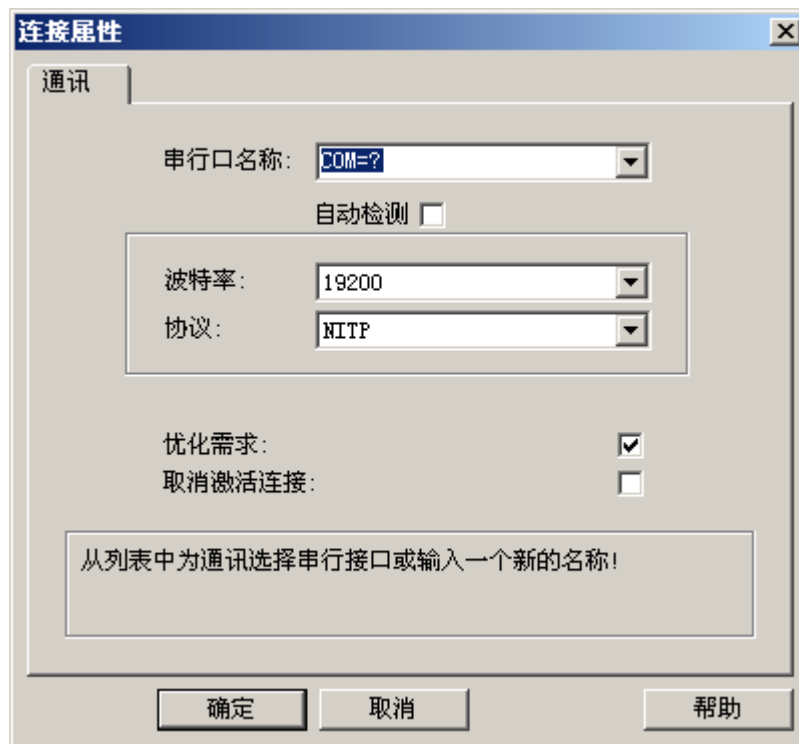
简介

使用 SIMATIC TI505 自动化系统可实现使用串行连接进行过程连接。自动化系统中使用 CPU 模块上的串行接口。

WinCC 中不需要附加通信模块。可通过系统中提供的默认 COM 端口进行通信。

步骤

1. 选择所需连接，并从快捷菜单中选择“连接参数”(Connection parameters)。“连接属性”(Connection properties) 对话框随即打开。



2. 在“串行口名称”域中，选择用于串行连接的通信接口（例如，COM1、COM2 或组态的端口）。
3. 如果需要通道单元自动检测 PLC 使用的数据传输速度和协议，请选择“自动检测”域。
4. 在“波特率”(Baud rate) 和“协议”(Protocol) 字段中设置要使用的数据传输速率和协议。
5. 通过选中“优化需求”复选框，可优化数据传送以通过一个请求传输多个变量。
6. 如果选中“禁用连接”域，则会禁用逻辑连接。这通常是在调试期间临时禁用连接的一个好方法。

16.3.3 组态变量

16.3.3.1 组态变量

简介

对于 WinCC 与 PLC 之间通过“SIMATIC TI Serial”通道进行的连接，可在 WinCC 中创建不同数据类型的变量。以下部分将对此进行说明。

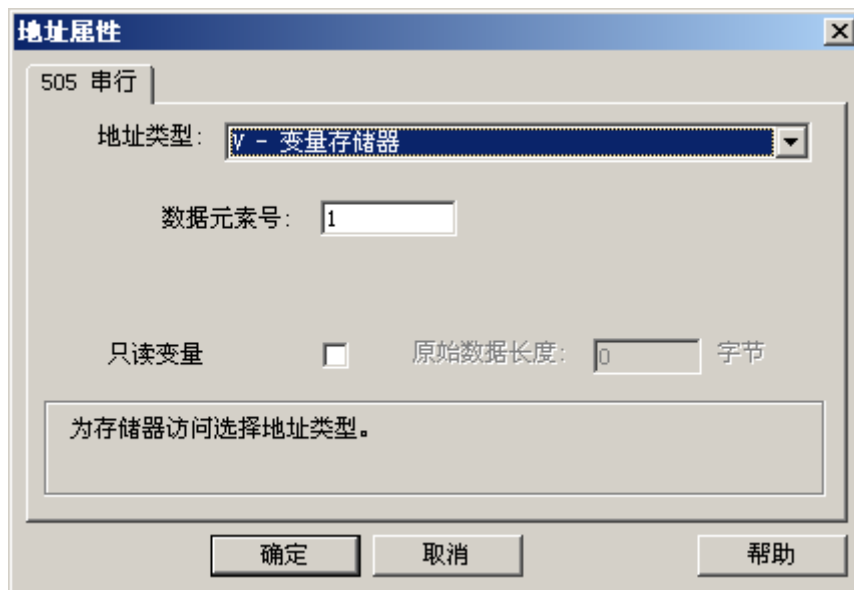
16.3.3.2 如何组态变量的地址

简介

变量地址根据 SIMATIC TI505 的地址结构输入。

步骤

1. 选择变量。
2. 在“数据类型”(Data type) 字段中设置所需数据类型（如有符号 16 位数）。
3. 如果是“二元”或“8 位”变量，则在“属性”(Properties) 区域会提供“位/字节变量”(Bit/Byte tag) 选项。如果要写入 AS 存储器，则选中相应的复选框“访问位/字节”(Access a Bit/Byte)。
4. 打开“地址属性”(Address properties) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。



5. 在“地址类型”域中选择变量在自动化系统中的位置。视所选地址类型而定，必须进行更多定义（例如，地址类型为“V - 变量存储器”需定义“数据元素号”域）。
6. 在“只读变量”域中，可指定该变量不能由 WinCC 写入。

说明

地址区域 V、K、X、Y 和 C 中支持结构变量。

关于地址类型的描述，可参见 SIMATIC TI505 技术文档。

16.3 组态通道

使用通道“TI Serial”时只能按位或按字节执行对 AS 存储器区域的写访问。在使用二进制和“8 位”变量时，除了会打开“地址属性”对话框外，还将打开“位/字节变量”对话框，该对话框可用于定义对各个位或字节的写访问。为此，每次写请求时均从 AS 读取已赋址的存储器区域，从而修改相应的位和/或字节。此后，数据写回到 AS 的存储器。

视变量类型而定，可以按位或按字节访问 AS 中的存储器。

16.3.3.3 如何组态按位访问变量


简介

使用通道“SIMATIC TI Serial”时只能按位或按字节执行对 AS 存储器区域的写访问。在使用二进制和“8 位”变量时，除了会打开“地址属性”对话框外，还将打开“位/字节变量”对话框，该对话框可用于定义对各个位或字节的写访问。为此，每次写请求时均从 AS 读取已赋址的存储器区域，从而修改相应的位和/或字节。此后，数据写回到 AS 的存储器。

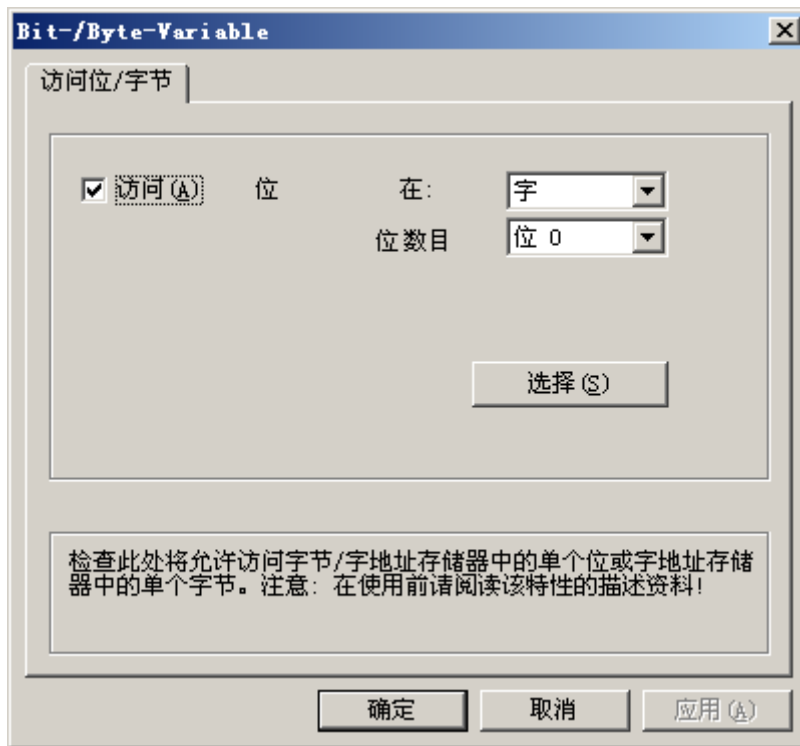
说明

AS 在读取数据区中执行的更改会在写回到该数据区时被覆盖。

步骤

1. 选择变量。
2. 在“数据类型”(Data Type) 字段中，设置“二进制变量”(Binary tag) 数据类型。
3. 打开“位/字节变量”(Bit/Byte tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。

4. 选择“访问位”(Access a bit) 复选框，并定义该位的地址。



5. 单击“选择”(Select) 按钮。将打开“地址属性”(Address properties) 对话框。
6. 在选择域中，选择 PLC 存储器的寻址类型。
7. 在选择域中，选择要更改的位号。

说明

在 S5 中，可以按字节寻址标志位、输入和输出，按字寻址数据块（DB、DX）。

16.3.3.4 如何组态按字节访问的变量


简介

使用通道“SIMATIC TI Serial”时只能按位或按字节执行对 AS 存储器区域的写访问。在使用二进制和“8 位”变量时，除了会打开“地址属性”对话框外，还将打开“位/字节变量”对话框，该对话框可用于定义对各个位或字节的写访问。为此，每次写请求时均从 AS 读取已赋址的存储器区域，从而修改相应的位和/或字节。此后，数据写回到 AS 的存储器。

说明

AS 在读取数据区中执行的更改会在写回到该数据区时被覆盖。

步骤

1. 选择变量。
2. 在“数据类型”(Data Type) 字段中，将数据类型设置为“无符号 8 位数”(Unsigned 8-bit value) 或“有符号 8 位数”(Signed 8-bit value)。
3. 打开“位/字节变量”(Bit/Byte tag) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
4. 选择“访问字节”(Access to a byte) 复选框，并定义该字节的地址。



5. 单击“选择”(Select) 按钮。将打开“地址属性”(Address properties) 对话框。

6. 在选择域中，选择 PLC 存储器的寻址类型。
7. 在选择域中，选择要更改的字节号。

17.1 WinCC 通道“SIMOTION”

简介

“SIMOTION”通道可连接 WinCC 站与 SIMOTION 自动化系统。使用 TCP/IP 协议通过工业以太网建立连接。

SIMOTION 是着重于运动控制应用及技术任务的自动化和驱动解决方案的系统平台。

SIMOTION 模块化系统包含 SIMOTION SCOUT 工程系统以及针对各种硬件平台的公用运行系统。

从 SIMOTION SCOUT 导出项目以访问 SIMOTION SCOUT 项目的数据库。然后，通过 Simotion Mapper 从导出的文件创建 WinCC 项目。

以后可在 WinCC 中对 SIMOTION 自动化系统的组态进行某些更改，例如更改 IP 地址。进一步更改必须在 SIMOTION SCOUT 中组态，再次导出然后通过 Simotion Mapper 传送。

说明

您必须熟悉 SIMOTION SCOUT 工程系统和 WinCC 的组态以在 WinCC 中组态“SIMOTION”通道。

17.2 所支持的数据类型概述

简介

在变量的组态中确定数据类型以及针对自动化系统中的数据格式的格式调整。

下表显示了通道支持的数据类型和格式调整应用。

所支持的数据类型

数据类型	类型转换
二进制变量	否
有符号 8 位数	是
无符号 8 位数	是
有符号 16 位数	是
无符号 16 位数	是
有符号 32 位数	是
无符号 32 位数	是
浮点数 32 位 IEEE 754	是
文本变量, 8 位字符集	否
原始数据类型	否

17.3 组态通道

17.3.1 “SIMOTION” 通道的组态

简介

本章将介绍如何组态“SIMOTION”通道。

说明

您必须熟悉 SIMOTION SCOUT 工程系统和 WinCC 的组态以组态“SIMOTION”通道。

请按以下步骤操作以组态“SIMOTION”通道：

1. 从 SIMOTION SCOUT 导出 SIMOTION SCOUT 项目。
2. 通过 Simotion Mapper 创建 WinCC 项目。
3. 打开 WinCC 项目。
4. 组态系统参数。

有关通道、连接和变量诊断的详细信息，请参见“诊断‘SIMOTION’通道”一章。

17.3.2 如何导出 SIMOTION SCOUT 项目

简介

此部分介绍如何从 SIMOTION SCOUT 导出变量和消息定义。

要求：

- 您熟悉 SIMOTION SCOUT 工程系统。
- 您对要导出的 SIMOTION SCOUT 项目具有访问权限。

步骤

1. 在 SIMOTION SCOUT 中打开要导出的 SIMOTION SCOUT 项目。
2. 选择“工具”(Tools)下的“导出 OPC 数据”(Export OPC Data)。

3. 为导出选择版本“SIMATIC NET V6.4”、所需范围以及至少“OPC-报警/事件”(OPC-Alarm/Event) 选项。

说明

Simotion Mapper 不处理“SIMATIC NET V6.4”外的其他导出版本。

4. 选择目标目录。
5. 选择通信接口。
该项目被导出。
6. 如果您正在使用路由，则输入路由信息。

该 SIMOTION SCOUT 项目被导出。“OPC_Data.sti”和“OPC_AE.xml”文件保存在目标目录中。

17.3.3 如何通过 Simotion Mapper 创建 WinCC 项目

简介

此部分介绍如何通过 Simotion Mapper 从导出的 SIMOTION SCOUT 项目创建 WinCC 项目。

说明

如果已为旧版本的 SIMOTION SCOUT 项目创建了 WinCC 项目，则在传送中仅更改 SIMOTION 参数。WinCC 项目中的所有其他组态设置（例如归档）保持不变。

要求:

- 您对 SIMOTION SCOUT 项目的导出文件“OPC_Data.sti”和“OPC_AE.xml”具有访问权限。
- 您对 WinCC 安装目录具有访问权限。

说明

在 SIMOTION SCOUT 编程环境中，可使用监视表过滤要导出的变量。使用监视表保持 WinCC 项目中较低的变量数。有关监视表的详细信息，请参见 SIMOTION SCOUT 在线帮助。

步骤

1. 在 WinCC 安装目录中启动“SimotionMapper.exe”程序。
2. 单击“打开”(Open)。导航到包含文件“OPC_Data.sti”和“OPC_AE.xml”的目录。
在 Simotion Mapper 中读取并显示数据。
3. 在 Simotion Mapper 项目管理器中，选择 WinCC 项目中所需的组和变量。
4. 选择“创建新 WinCC 项目”(Create new WinCC project)。
5. 如果要更改“WinCC 连接名称”，单击显示的名称，然后输入新名称。
6. 指定技术报警的“第一个 TA 消息编号”。必须以此方法选择值：其不会导致与其他通信通道的消息产生冲突。默认值为 100。
Simotion Mapper 为起始于指定编号的每个 Simotion 连接创建一个消息以及合计六个模板消息。
7. 单击“开始映射”(Start mapping)。选择用于存放 WinCC 项目的目标文件夹。
创建 WinCC 项目。进度条指示过程的进度。
8. 关闭 Simotion Mapper。

WinCC 项目已创建，现在可在 WinCC 中打开并编辑该项目。

说明

您可能必须在 WinCC 中设置“SIMOTION”通道的系统参数，以使用创建的 WinCC 项目。

17.3.4 如何通过 Simotion Mapper 更改 WinCC 项目

简介

此部分介绍如何通过 Simotion Mapper 将导出的 SIMOTION SCOUT 项目添加到现有的 WinCC 项目。这样，您可在一个 WinCC 项目中多次使用相同的 Simotion 项目。

说明

如果已为旧版本的 SIMOTION SCOUT 项目创建了 WinCC 项目，则在传送中仅更改 SIMOTION 参数。WinCC 项目中的所有其他组态设置（例如归档）保持不变。

要求:

- 您对 SIMOTION SCOUT 项目的导出文件“OPC_Data.sti”和“OPC_AE.xml”具有访问权限。
- 您对 WinCC 安装目录具有访问权限。

说明

在 SIMOTION SCOUT 编程环境中，可使用监视表过滤要导出的变量。使用监视表限制 WinCC 项目中的变量数。有关监视表的详细信息，请参见 SIMOTION SCOUT 在线帮助。

步骤

1. 打开要编辑的 WinCC 项目。
2. 在 WinCC 安装目录中启动“SimotionMapper.exe”程序。
3. 单击“打开”(Open)。导航到包含文件“OPC_Data.sti”和“OPC_AE.xml”的目录。在 Simotion Mapper 中读取并显示数据。
4. 在 Simotion Mapper 项目管理器中，选择 WinCC 项目中所需的组和变量。
5. 选择“添加到打开的项目”(Add to the open project)。
6. 如果要再次添加已创建的组或变量，必须通过单击显示的名称更改“WinCC 连接名称”。
7. 如果不想为连接传送任何消息、组或变量，则取消选择“WinCC 连接名称”(WinCC connection name)。
8. 指定是否应覆盖变量。
9. 指定技术报警的“第一个 TA 消息编号”。必须以此方法选择值：其不会导致与其他通信通道的消息产生冲突。默认值为 100。
Simotion Mapper 为起始于指定编号的每个 Simotion 连接创建一个消息以及合计六个模板消息。

说明

请不要更改任何已映射的“第一个 TA 消息编号”。否则，可能会出现不可预见的消息行为。

10. 单击“开始映射”(Start mapping)。选择用于存放 WinCC 项目的目标文件夹。
SIMOTION SCOUT 项目添加到打开的 WinCC 项目。进度条指示过程的进度。
11. 关闭 Simotion Mapper。

用 SIMOTION SCOUT 项目扩展了 WinCC 项目，并且该 WinCC 项目随设置一起保存。

17.3.5 如何更改连接参数

简介

在本部分中，您将学习如何更改 SIMOTION 网络地址的连接参数。

说明

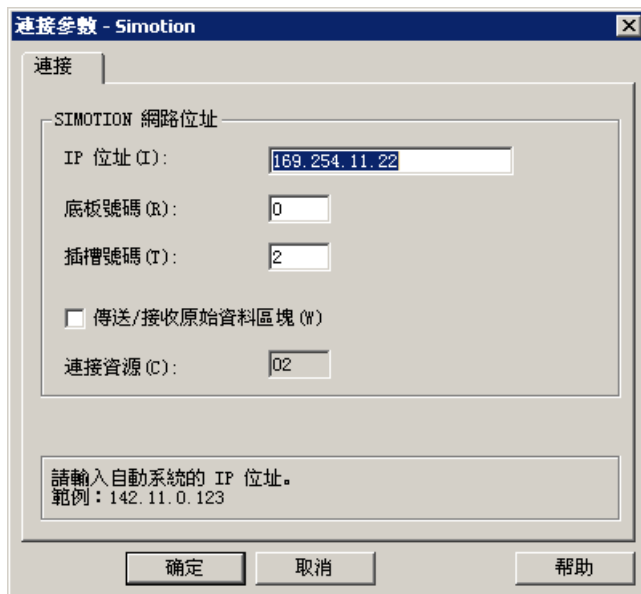
仅更改此处列出的连接参数。不创建“SIMOTION”通道的新连接。错误设置的连接可能导致 PLC 中出现控制错误。根据““SIMOTION”通道的组态 (页 463)”部分的描述组态新的连接。

要求

- SIMOTION 通信驱动程序集成在 WinCC 项目中。
- 必须在“SIMOTION”通道单元中创建了连接。

步骤

1. 在“变量管理”编辑器中打开“SIMOTION”通信驱动程序的目录结构。
2. 从通道单元“Simotion”的一个连接的快捷菜单中，选择条目“连接参数”(Connection parameters)。将打开“连接参数 - SIMOTION”(Connection parameters - SIMOTION) 对话框。



3. 在相应字段中更改 SIMOTION 网络地址的连接参数。
4. 单击“确定”(OK)，关闭每个打开的对话框。

17.3.6 如何更改变量地址

简介

本部分介绍如何更改“SIMOTION”通道中的变量地址。


说明

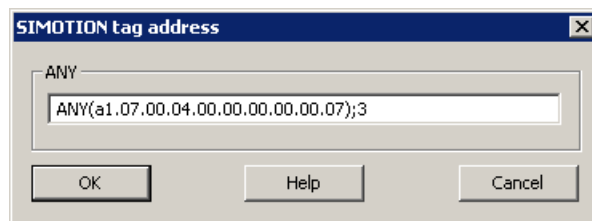
您必须十分了解如何使用 ANY 指针来更改 SIMOTION 通道中的变量地址。如果变量地址输入不正确，则可能无法建立通信连接。

要求

- “SIMOTION”通道集成到 WinCC 项目中。
- 已在“SIMOTION”通道单元中创建与变量的连接。

步骤

1. 打开“SIMOTION 变量地址”(SIMOTION tag address)对话框。
为此，请单击“地址”(Address)字段，然后单击  按钮。



2. 更改变量地址。

17.3.7 系统参数组态

17.3.7.1 通道单元的系统参数

简介

如果需要不同的组态而不是 WinCC 默认设置，则在“系统参数”(System Parameters)对话框中进行这些设置。

可以更改以下系统参数：

- 逻辑设备名称
- 通道使用 AS 中的周期性读取服务

逻辑设备名称

在 PLC 中安装通讯处理器后，WinCC 和 PLC 通过已分配的逻辑设备名称进行通信。

通道使用 AS 中的周期性读取服务

PLC 周期性读取服务对即将周期性读取到单个请求中的变量进行编组，并将其传送给 PLC。收到请求后，PLC 首次发送已请求的数据，然后每次经过周期时间后，再次发送。

启用周期性读取服务后，可使用变化驱动的传送功能。如果 PLC 支持变化驱动的传送，则仅当更改值时才传送数据。

17.3.7.2 如何组态系统参数

简介

此部分介绍如何组态“SIMOTION”通道的系统参数。

“系统参数”对话框由两个标签组成：

- “SIMOTION”选项卡
- “单元”(Unit)选项卡

说明

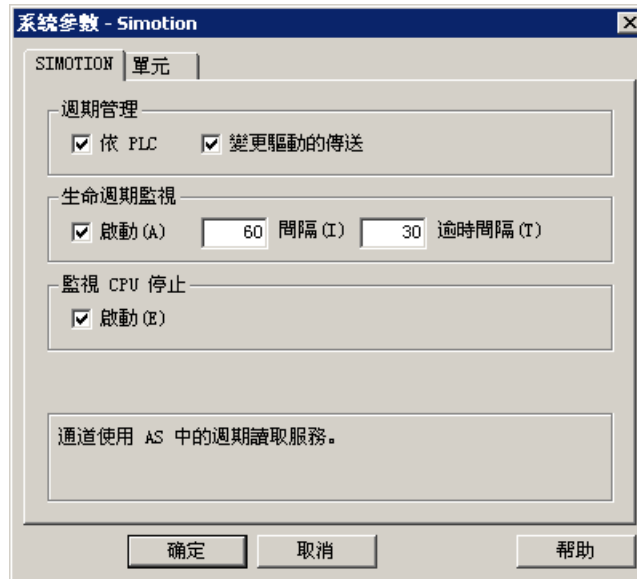
将项目复制到另一台计算机时，“单元”(Unit)选项卡中的设置会得到保留。相反，将删除“SIMOTION”选项卡上的设置。

要求

- “SIMOTION”通道集成到 WinCC 项目中。

步骤

1. 在变量管理器中选择“SIMOTION”通道。在“Simotion”通道单元的快捷菜单中，打开“系统参数”(System Parameters)对话框。
2. 选择“SIMOTION”选项卡。



3. 要启用周期性读取变量和变化驱动的传送，则选择“通过 PLC”(by PLC)和“变化驱动的传送”(Change-driven transfer)。

说明

集成的 SINAMICS 伺服控制不支持“周期控制”、“设备状态监视”和“停止监视”功能。因此，“SIMOTION”通道忽略与 SINAMICS 伺服控制器的连接的相应设置。建立连接时，通道确定是否 AS 支持相应功能。

4. 如果需要，选择“设备状态监视”(Lifebeat monitoring) 功能。
确定发送设备状态监视消息的时间间隔（以秒为单位）。
确定用于监视设备状态监视消息响应的监视时间（以秒为单位）。
5. 当 SIMOTION CPU 处于停止状态时，如果要 WinCC 在通信中发出故障信号，则启用“CPU 停止监视”。

6. 选择“单元”(Unit)选项卡。
“STONLINE”显示为“逻辑设备名称”的默认值。如果在安装使用的通信处理器时选择了不同名称，则必须更改设备名称。



7. 要在运行系统启动时自动设置设备名称，则选择“自动设置”(Set automatically)。
8. 要在处理期间使写作业的优先级高于读作业，则选择“写优先”(Write with priority)。
9. 单击“确定”(OK)，关闭对话框。

说明

设置更改只有在 WinCC 重启后才有效。

17.3.7.3 如何更改逻辑设备名称

简介

WinCC 和 SIMOTION 通过逻辑设备名称进行通信。安装通讯处理器时分配这些逻辑设备名称。

要求

- “SIMOTION”通道集成到项目中。
- 已在“SIMOTION”通道单元中创建连接。

步骤

1. 在变量管理器中选择 SIMOTION 通道。
2. 在快捷菜单中打开“系统参数”(System Parameters) 对话框。
3. 选择“单元”(Unit) 选项卡。
4. 在“逻辑设备名称”(Logical device name) 字段中输入设备名称。可从列表中选择条目或输入新的名称。
设备名称由“设置 PG/PC 接口”工具确定。在系统控制中调用该工具。如果未安装该工具，则仅显示当前设置的设备名称。
如果指定不同的逻辑设备名称，则将显示一条消息。
仅当目标站上使用的通讯处理器未安装在组态系统上时，才输入名称。
5. 单击“确定”(OK)，关闭对话框。

说明

设置更改只有在 WinCC 重启后才有效。

17.4 诊断“SIMOTION”通道

17.4.1 “SIMOTION”通道的诊断可能性

“SIMOTION”通道及其变量的诊断和错误检测存在以下可能性。

检查通讯处理器组态

检查访问点后，可以使用应用程序“设置 PG/PC 接口”来测试通讯处理器。也可以在 SIMATIC NET 中以相同的方式检查通讯处理器。

检查连接和变量的组态

系统和连接参数的组态可能出现错误。不正确的变量寻址也可能导致变量值错误。

使用“通道诊断”诊断通道

可以通过“通道诊断”查询运行系统中通道和连接的状态。通过“错误代码”显示错误。

诊断通道变量

在运行系统的变量管理器中，可以查询当前值、当前质量代码以及变量的上次变更时间。

17.4.2 日志文件条目的描述

简介

通道在日志文件中输入重要的状态变化和错误。这些条目支持通信故障的分析。

文件中的每个条目都包含日期和时间标志，后面带有标记名称和描述。

记录册条目实例：

```
2009-10-28 12:10:11,467 INFO Log starting ...  
2009-10-28 12:10:11,483 INFO | LogFileName : D:\SIEMENS\WINCC\Diagnosis  
\Simotion_01.LOG  
2009-10-28 12:10:11,483 INFO | LogFileCount : 3
```

17.4 诊断“SIMOTION”通道

```

2009-10-28 12:10:11,483 INFO | LogFileSize : 1400000
2009-10-28 12:10:11,483 INFO | TraceFlags : fa000000
2009-10-28 12:10:11,498 INFO SIMOTION channel DLL started!
2009-10-28 12:10:11,498 INFO SIMOTION channel with own cycle creation!
2009-10-28 12:10:11,967 INFO Connection "D445": StartRegisterEvVariable for
dwVariableCount = 89
2009-10-28 12:10:11,967 INFO Connection "D445": RegisterEvVariable for Variable
"@D445@Checksum"!
...
2009-10-28 12:10:11,983 INFO Connection "D445": EndRegisterEvVariable
2009-10-28 12:10:12,436 INFO S7DOS release: @(#)TIS-Block Library DLL Version
R8.0.0.0-REL-BASIS
2009-10-28 12:10:12,436 INFO S7DOS version: V8.0 / 0
2009-10-28 12:10:12,436 INFO SIMOTION version: V6.0 / Sep 15 2009 / 08:06:43
2009-10-28 12:10:12,436 INFO SIMOTION channel unit "Simotion" activated!
2009-10-28 12:10:12,451 ERROR Cannot connect to "SINAMICS_Integrated":Errorcode
0xFFDF 42C2!
2009-10-28 12:10:12,451 ERROR Cannot connect to "D445":Errorcode 0xFFDF 42C2!

```

“INFO” 标记中最重要条目的描述

消息文本	含义
LogFileName : C:\ Siemens\ WinCC\ Diagnose\ "channel_name".LOG	带路径日志文件的名称
LogFileCount : "n"	通道的日志文件的数目
LogFileSize : "x"	单个日志文件的大小（以字节为单位）
TraceFlags : c4000000	用十六进制数字显示跟踪功能使用的标记
SIMOTION channel DLL started!	起始消息
SIMOTION channel DLL terminated!	结束消息

“ERROR”标记中最重要条目的描述

消息文本	含义
Cannot connect to <connectionname>: Errorcode 0x0000 7<xxx>!	通信错误 激活 WinCC 后，不能立即建立与 SIMOTION 的通信。 <connectionname> = 连接的名称 <xxx> 1...fff 通道已接收所有其他错误代码，作为 S7DOS 函数调用、更低层或来自 AS 的结果。
Connectionerror <nnn> <connectionname>: Errorcode 0x0000 7xxx!	通信错误 激活 WinCC 后，不能建立与 SIMOTION 的通信。 连接已中断。 <nnn> = 此连接的连接中止数 <connectionname> = 连接的名称 <xxx> 1...fff 通道已接收所有其他错误代码，作为 S7DOS 函数调用、更低层或来自 AS 的结果。
Channel API error: errorstring	通道 API 出错 通道将错误字符串“errorstring”传到 WinCC 项目管理器。根据错误关联，错误字符串显示在信息框中。有关错误字符串描述，请参见 API 错误文本。
Max. count of API errors reached - API logbook deactivated	通道 API 出错 根据错误和功能，可能在 API 上周期性发生错误。要避免这些错误信息占满记录册文件，一个 API 错误最多可以输出 32 条信息。
Cannot write storage data! Cannot read storage data / use default data Storage data illegal or destroyed / use default data! No storage data / use default data!	常规通道错误消息
单元“单元名”中的设备名从“旧 设备名”改为“新设备名”	初始化消息

说明

“SIMOTION”通道的错误代码对应于“SIMATIC S7 Protocol Suite”通道的错误代码。有关错误代码的描述，请参见“SIMATIC S7 Protocol Suite”通道帮助的“连接故障的错误代码”一章。

此外，如果一致性检查失败，SIMOTION 将报告错误代码 0x000 7301。产生错误消息的原因是：从 SIMOTION SCOUT 导出的 AS 输入校验和的数据与已连接设备中的校验和不匹配。

System Info

18.1 “System Info” 通道

内容

“System Info” 通道用于解释系统信息，如时间、日期、磁盘容量，并提供定时器和计数器等功能。

该章将介绍

- 组态通道、连接和变量
- 在过程画面中显示系统信息
- 使用系统信息来触发和显示消息
- 用图表显示系统信息
- 显示多用户系统中多个服务器的系统信息

18.2 WinCC System Info 通道

原理

“System Info” 通道用于计算系统信息，如时间、日期、磁盘容量，并提供定时器和计数器等功能。

可能的应用如下：

- 在过程画面中显示时间、日期和星期几
- 通过在脚本中判断系统信息来触发事件
- 在趋势图中显示 CPU 负载
- 显示和监视客户端系统中不同服务器上可用的驱动器空间
- 监视可用磁盘容量并触发消息

该通道无硬件要求，因为通道直接访问安装该通道的计算机的系统信息。为了使通道起作用，必须建立连接。可以进行附加连接，但不需要进行特别的操作。

有关通道和变量诊断的更多信息，请参阅“通信诊断”。

说明

授权

“System Info” 通道所需的过程变量不需要许可证。因此，这些变量不算在许可证计数中。

用户权限

如果您没有管理员权限，则必须是高级用户及“Performance Monitor 用户”组的成员，才能使用“System Info”通道。

通信手册

通信手册包含通道组态的更多信息和扩展实例。本手册可在 Internet 上下载，地址如下：

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

搜索项目号：

- A5E00391327

参见

在多用户和客户系统中使用 (页 499)

如何调用和评价系统信息 (页 487)

如何组态 System Info 通道 (页 486)

与其它软件组件的不同之处 (页 485)

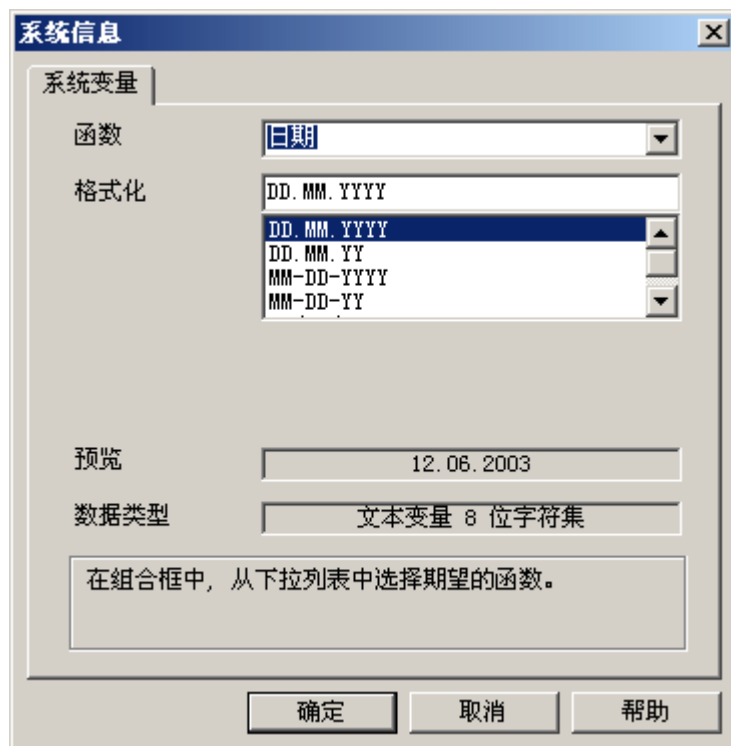
所支持的系统信息概述 (页 480)

通道和变量的诊断 (页 507)

18.3 所支持的系统信息概述

引言

在“System Info”对话框的“功能”域中，指定要分配给 WinCC 变量的系统信息。在“格式”域中设置显示格式。



System Info 通道支持的系统信息 - 概述

函数	数据类型	格式	预览
日期	文本变量 8 位字符集	DD.MM.YYYY DD.MM.YY MM-DD-YYYY MM-DD-YY MM/DD/YY	21.10.1999 21.10.99 10-21-1999 10-21-99 10/21/99
日	无符号 16 位数	DD	1...31

函数	数据类型	格式	预览
月	无符号 16 位数	MM	1...12
年	无符号 16 位数	YYYY	2000
工作日	无符号 16 位数 文本变量 8 位字符集	文本： 1 指 星期一... 7 指星期日 文本： 周一、周二、周三、 周四、周五、周六、周日	1...7 周一 ... 周日
时间	文本变量 8 位字符集； 长度 = 10 字节 长度 = 12 字节	HH:MM:SS HH:MM HH:MM AM,PM HH:MM:SS AM,PM	23:45:37 23:45 23:45 PM 23:45:37 PM
小时	无符号 16 位数	HH	0...23
分	无符号 16 位数	MM	0...59
秒	无符号 16 位数	SS	0...59
毫秒	无符号 16 位数	MSC	0...999
计数器	有符号 32 位数	ZZZZ	0...9999
CPU 负载	浮点数 32 位 IEEE 754	总负载 (%) 空闲负载 (%) 处理负载 (%)	0...100%
定时器	有符号 32 位数	TTTT	0...9999
空闲主内存	浮点数 32 位 IEEE 754	空闲容量 (KB) 空闲 (%) 空闲 (字节)	0...n kB 0...100% 0...n B
空闲 磁盘容量 (本地磁盘)	浮点数 32 位 IEEE 754	空闲容量 (MB) 空闲容量 (%)	0...n MB 0...100%

函数	数据类型	格式	预览
打印机监控	无符号 32 位数	假脱机磁盘 满容量 打印机状态 作业状态 空闲的假脱机磁盘 区域 (KB) 空闲 PRT_OUT 磁盘区域 (KB) 假脱机目录 大小 (KB) PRT_OUT 目录大小 (KB)	0...n % 0...n 0...n 0...n KB 0...n KB 0...n KB 0...n KB
交换文件 的状态	浮点数 32 位 IEEE 754	已使用 (KB) 已使用 (%) 可用 (KB)	0...n kB 0...100% 0...n kB

计数器

该函数可在脚本中用于测试。

定时器

选择该函数后，“限制从”和“到”域将扩展到“System Info”对话框。

每过一秒后，定时器数值增加或减小。要进行改变的方向由域“限制从”和“到”的开始和结束值来确定。如果起始值小于结束值，则定时器数值将增加。如果起始值大于结束值，则定时器数值将减小。

如果在运行系统中，在链接到定时器的 I/O 域中输入一个数值，则定时器的起始值和当前值都将被设置为该值。实例：组态为 0 至 60 的定时器。如果在运行系统写入“0”，定时器将复位。

取消激活后，将再次启用原来的起始值。

CPU 负载

对于与多处理器 PC 有关的格式“总负载 (%)”和“空闲负载 (%)”，输入从“0”开始的 CPU 编号。

对于格式“处理负载 (%)”，如果一个进程有多个事例，则输入事例编号。

空闲磁盘容量

系统仅可确定本地硬盘或软盘的可用空间。

打印机监控

带有“打印机状态”和“作业状态”格式的服务器名称必须输入到“打印机”域中。使用中的打印机必须支持该状态信息以便使用该系统信息。

要分析打印机状态，请遵守以下内容：

- 端口监视器负责将打印机状态发送到假脱机程序。根据所选的打印机端口，安装不同的端口监视器 DLL。在 Windows 提供的端口监视器中，只有“TCPMON.DLL”能够使用 TCP/IP 端口发送打印机状态。使用 LPT 端口的“LOCALMON.DLL”不发送打印机状态。
- 只有在提交打印作业之后，才会评估打印机状态，而不是在轮询端口状态时进行评估。

对于“空闲 PRT_OUT 驱动空间”和“PRT_OUT 目录大小”格式，通道会自动确定“目录”域的路径。

“打印机状态”格式的错误代码

状态	错误代码
PRINTER_STATUS_PAUSED	0x00000001
PRINTER_STATUS_ERROR	0x00000002
PRINTER_STATUS_PENDING_DELETION	0x00000004
PRINTER_STATUS_PAPER_JAM	0x00000008
PRINTER_STATUS_PAPER_OUT	0x00000010
PRINTER_STATUS_MANUAL_FEED	0x00000020
PRINTER_STATUS_PAPER_PROBLEM	0x00000040
PRINTER_STATUS_OFFLINE	0x00000080
PRINTER_STATUS_IO_ACTIVE	0x00000100
PRINTER_STATUS_BUSY	0x00000200
PRINTER_STATUS_PRINTING	0x00000400
PRINTER_STATUS_OUTPUT_BIN_FULL	0x00000800
PRINTER_STATUS_NOT_AVAILABLE	0x00001000
PRINTER_STATUS_WAITING	0x00002000
PRINTER_STATUS_PROCESSING	0x00004000

18.3 所支持的系统信息概述

状态	错误代码
PRINTER_STATUS_INITIALIZING	0x00008000
PRINTER_STATUS_WARMING_UP	0x00010000
PRINTER_STATUS_TONER_LOW	0x00020000
PRINTER_STATUS_NO_TONER	0x00040000
PRINTER_STATUS_PAGE_PUNT	0x00080000
PRINTER_STATUS_USER_INTERVENTION	0x00100000
PRINTER_STATUS_OUT_OF_MEMORY	0x00200000
PRINTER_STATUS_DOOR_OPEN	0x00400000
PRINTER_STATUS_SERVER_UNKNOWN	0x00800000
PRINTER_STATUS_POWER_SAVE	0x01000000

“作业状态”格式的错误代码

状态	错误代码
JOB_STATUS_PAUSED	0x00000001
JOB_STATUS_ERROR	0x00000002
JOB_STATUS_DELETING	0x00000004
JOB_STATUS_SPOOLING	0x00000008
JOB_STATUS_PRINTING	0x00000010
JOB_STATUS_OFFLINE	0x00000020
JOB_STATUS_PAPEROUT	0x00000040
JOB_STATUS_PRINTED	0x00000080
JOB_STATUS_DELETED	0x00000100
JOB_STATUS_BLOCKED_DEVQ	0x00000200
JOB_STATUS_USER_INTERVENTION	0x00000400
JOB_STATUS_RESTART	0x00000800

说明

“打印机状态”和“作业状态”格式的错误代码对应于 Visual C 参考文件“Winspool.h”中的数值。

18.4 与其它软件组件的不同之处

引言

“System Info”通道的一些系统信息也可通过使用 WinCC ActiveX 控件进行解释和显示。

与 ActiveX 控件的根本差异是在“System Info”通道中，系统信息分配给 WinCC 变量。可以重复进行连续解释（例如消息、限制值），然后分别进行组态。ActiveX 控件用于指定的应用程序，也可用于多用户或客户机系统。

对于下列系统信息，ActiveX 控件和通道之间存在差异：

时间

在 WinCC 中使用 ActiveX 控件“WinCC 数字/模拟时钟控件”显示时间。该控件也支持时间的模拟显示。即使该控件在过程控制系统选件中使用，也不需要“System Info”通道。使用控件时，在其过程画面中可显示 WinCC 客户机的时间。对于“System Info”通道这是不可能的，因为它总是显示服务器的系统时间。

空闲磁盘容量

ActiveX 控件“IX Diskspace”用于显示 WinCC 中磁盘可用的空间。该 ActiveX 控件也可显示网络驱动器上可用的空间并支持其它组态选项，例如直接设置控件中的多个限制值。

18.5 组态通道

18.5.1 如何组态 System Info 通道

简介

本节说明如何组态“System Info”通道。

步骤

1. 在变量管理的导航区域中，从节点“变量管理”(Tag Management) 的快捷菜单中选择条目“添加新的驱动程序”(Add new driver)。
2. 选择“系统信息”(System Info) 驱动程序。随即将创建该通道，并在变量管理器中显示该通信驱动程序。
3. 选择关联的“System Info”通道单元并调用快捷菜单。在该快捷菜单中，选择“新建连接”。
4. 输入连接的名称。
5. 单击表格区下方的“变量”(Tags) 选项卡。
6. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
如果要使用该通道的示例，请按照主题“如何组态变量”的说明继续操作。
7. 在“数据类型”(Data Type) 字段中选择所需的数据类型。

18.6 评价和显示系统信息的实例

18.6.1 如何调用和评价系统信息

本节以实例说明如何以多种方式显示和评价系统信息。

参见

如何在状态显示中显示打印机状态 (页 496)

如何组态有关空闲磁盘容量的消息 (页 492)

如何在趋势窗口中显示 CPU 负载 (页 491)

如何用棒图显示空闲磁盘容量 (页 490)

如何在 I/O 域中显示时间 (页 489)

如何在“System Info”通道中组态变量 (页 487)

18.6.2 如何在“System Info”通道中组态变量

简介

本节说明如何在“System Info”通道中组态变量。将在实例中使用这些变量。

要求


安装“SystemInfo.chn”通道。

所使用数据类型的表格

下表显示了“System Info”通道中使用的变量类型和格式。

示例	系统信息	变量名称	数据类型	格式
I/O 字段	时间	Sysinfo_Time	文本变量 8 位字符集	小时:分: 秒 (HH:MM:SS)
棒图, 消息	空闲驱动器 容量 (驱动器: C)	Sysinfo_Drive_ C	浮点数 32 位 IEEE 754	0-100% (空闲容量 %)
趋势	CPU 负载	Sysinfo_CPU	浮点数 32 位 IEEE 754	0-100% (总负载 %)
打印机状 态	打印机监控	Sysinfo_Printe rstate	无符号 32 位数	0-n (十六进制) (打印机状态)

步骤

1. 从关联的“System Info”通道单元的快捷菜单中选择条目“新建连接”，并创建名为“Testinfo”的连接。
2. 单击表格区下方的“变量”(Tags) 选项卡。
3. 单击“名称”(Name) 列最上方的空单元格。
输入变量名。
4. 打开“系统信息”(System Info) 对话框。
为此，请单击“地址”(Address) 字段，然后单击  按钮。
5. 在“数据类型”域中，按照表格设置变量的数据类型。
6. 按照表格中的信息来设置系统信息和显示格式。
7. 关闭对话框。

参见

如何组态有关空闲磁盘容量的消息 (页 492)

如何组态 System Info 通道 (页 486)

如何在状态显示中显示打印机状态 (页 496)

如何在趋势窗口中显示 CPU 负载 (页 491)

如何用棒图显示空闲磁盘容量 (页 490)

如何在 I/O 域中显示时间 (页 489)

18.6.3 如何在 I/O 域中显示时间

要求

用数据类型“文本变量 8 位字符集”组态“Sysinfo_Time”变量。该变量必须以“HH:MM:SS”显示格式来表达“时间”系统信息。

步骤

1. 启动图形编辑器并打开画面。
2. 在画面中添加一个 I/O 域。从“智能对象”对象列表中选择“I/O 域”对象。将打开“I/O 域组态”对话框。



3. 在“变量”域中，输入名称“Sysinfo_Time”。
4. 将更新周期设置为“1 秒”。
5. 将域类型设置为“输出”。关闭对话框。
6. 在 I/O 域的快捷菜单中单击“属性”以打开“对象属性”对话框。
7. 在“属性”标签上，选择“输出/输入”。将“数据格式”属性设置为“字符串”。
8. 关闭对话框并保存画面。
9. 单击图形编辑器工具栏中的相应按钮来激活运行系统。

说明

应仔细考虑如何选择更新周期，因为它会影响计算机负载。因此，每 250 ms 更新一次时间显示将有损于系统性能。

参见

如何启动运行系统 (页 498)

如何插入 I/O 域 (页 498)

如何在“System Info”通道中组态变量 (页 487)

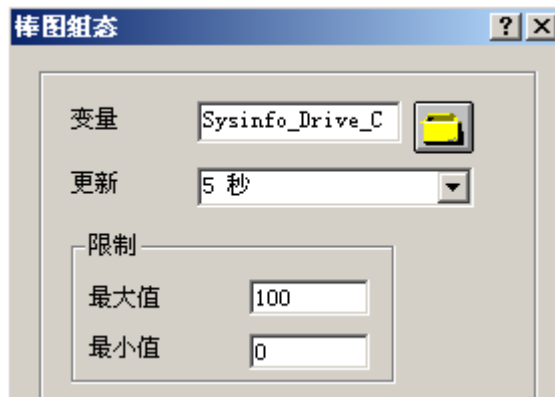
18.6.4 如何用棒图显示空闲磁盘容量

要求

组态数据类型为“浮点数 32 位 IEEE754”的变量“Sysinfo_Drive_C”。该变量必须以显示格式“空闲容量 (%)”表达驱动器“C”的“空闲磁盘空间”系统信息。

步骤

1. 启动图形编辑器并打开画面。
2. 在画面中插入一个棒图。为此，从对象选项板的“智能对象”中选择对象“棒图”。将打开“棒图组态”对话框。



3. 在“变量”域中，输入名称“Sysinfo_Drive_C”。
4. 将更新周期设置为“5 秒”。
5. 将最大数值设置为“100”，最小数值设置为“0”。关闭对话框。
6. 在棒图的快捷菜单中单击“属性”以打开“对象属性”对话框。
7. 在“属性”标签上，选择“坐标轴”。将属性“小数位”设置为“0”。
8. 关闭对话框并保存画面。
9. 单击图形编辑器工具栏中的相应按钮来激活运行系统。

说明

应仔细考虑如何选择更新周期，因为它会影响计算机负载。因此，每秒更新一次可用的驱动器空间显示将有损于系统性能。

参见

如何启动运行系统 (页 498)

如何插入棒图 (页 498)

如何在“System Info”通道中组态变量 (页 487)

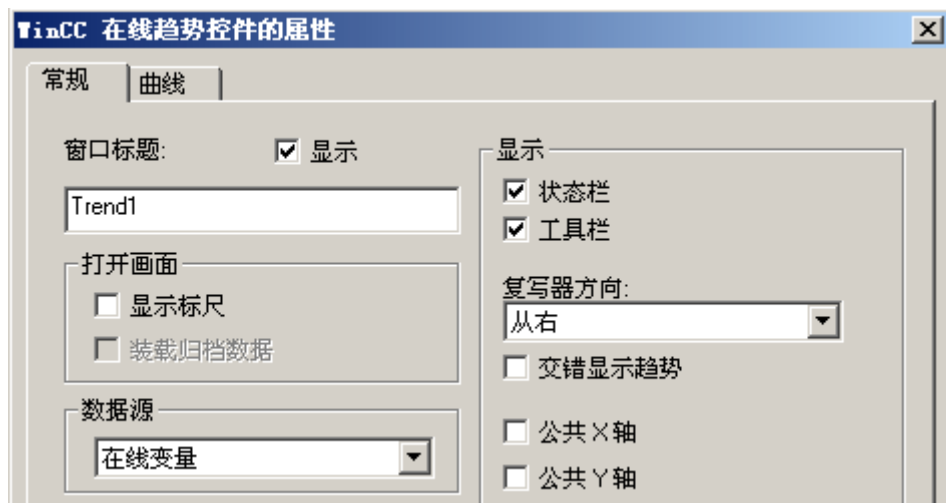
18.6.5 如何在趋势窗口中显示 CPU 负载

要求

组态数据类型为“浮点数 32 位 IEEE754”的变量“Sysinfo_CPU”。该变量必须以显示格式“总负载 (%)”表达“CPU 负载”系统信息。

步骤

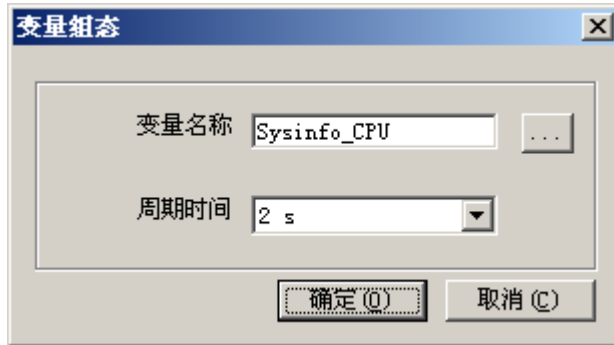
1. 启动图形编辑器并打开画面。
2. 将趋势显示插入画面。为此，可从“控件”对象选项板中选择“WinCC 在线趋势控件”对象。将打开“WinCC 在线趋势控件的属性”对话框。



3. 在“常规”标签的“窗口标题”域中输入名称“Trend1”。

18.6 评价和显示系统信息的实例

4. 在“数据源”域中，选择“在线变量”。
5. 先单击“趋势”标签，然后单击“选择”按钮打开“变量组态”对话框。
6. 输入“Sysinfo_CPU”作为变量名，然后选择周期时间“2 秒”。关闭对话框。



7. 关闭“WinCC 在线趋势控件的属性”对话框并保存画面。
8. 单击图形编辑器工具栏中的相应按钮来激活运行系统。

说明

应仔细考虑如何选择更新周期，因为它会影响计算机负载。因此，每 500 毫秒更新一次 CPU 负载显示将有损于系统性能。

参见

如何启动运行系统 (页 498)

如何在“System Info”通道中组态变量 (页 487)

18.6.6 如何组态有关空闲磁盘容量的消息

要求

组态数据类型为“浮点数 32 位 IEEE754”的变量“Sysinfo_Drive_C”。该变量必须以显示格式“空闲容量 (%)”表达驱动器“C”的“空闲磁盘空间”系统信息。

步骤

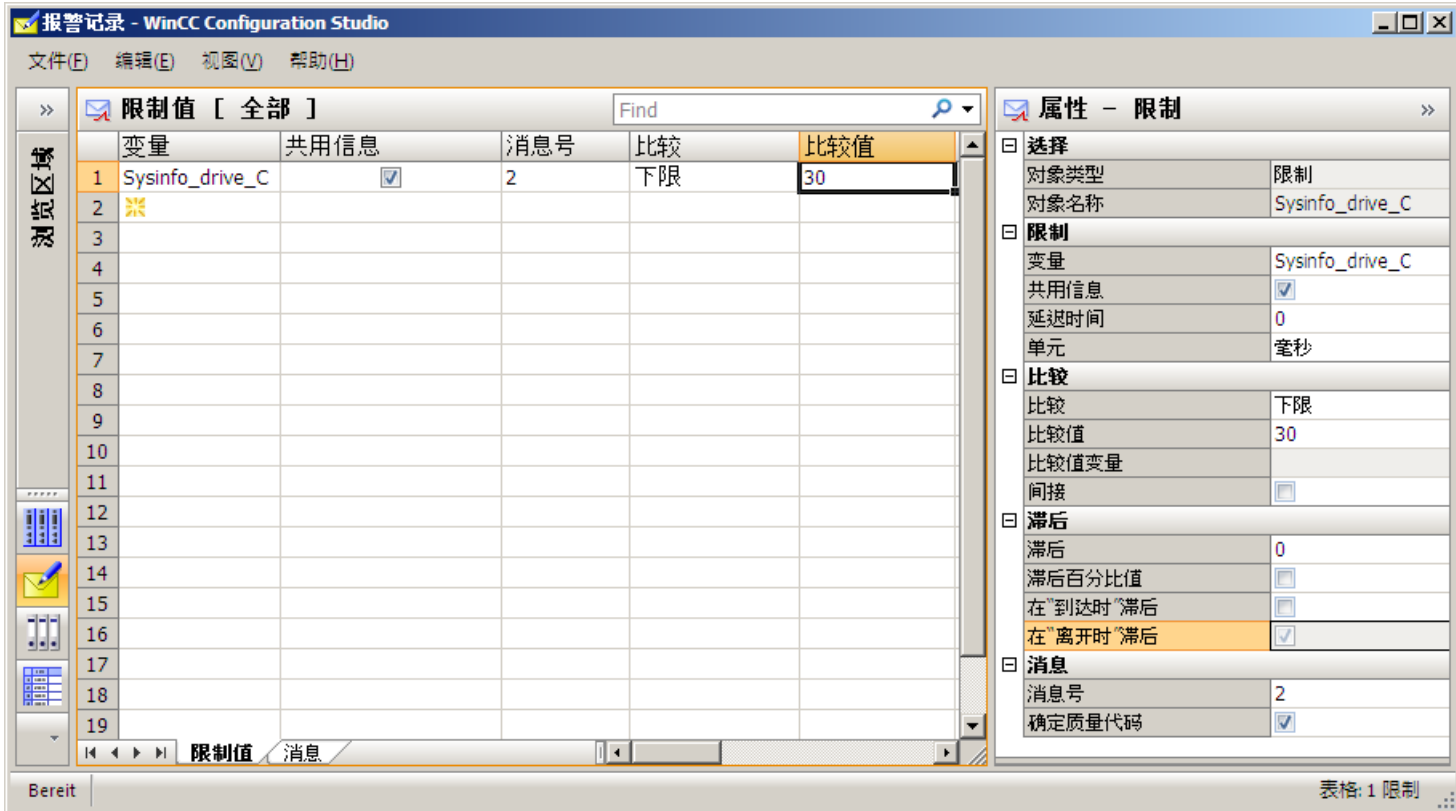
1. 打开“报警记录”编辑器并创建一条新消息。
2. 在“属性”(Properties) 区域中，定义：
 - 文本“存储空间”为“消息文本”
 - 文本“硬盘”为“错误点”



3. 在“报警记录”(Alarm Logging) 导航区域中，选择“模拟量报警”节点。
4. 为变量“Sysinfo_Drive_C”创建一个新的模拟量报警。
5. 激活选项“共享消息”(Shared message)。在“消息号”域中输入新建消息的数目。

18.6 评价和显示系统信息的实例

6. 在浏览窗口中选择“模拟量报警”。在数据窗口中选择条目“Sysinfo_Drive_C”。单击快捷菜单中的“新建”以打开“属性”对话框。单击选项钮“下限”并在“限制值或变量”域中输入数值“30”。
7. 关闭对话框。保存并退出报警记录。



参见

如何在“System Info”通道中组态变量 (页 487)

如何显示有关可用磁盘容量的消息 (页 494)

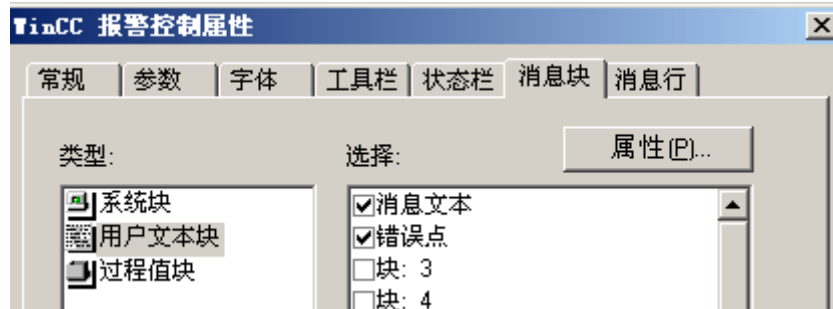
18.6.7 如何显示有关可用磁盘容量的消息

要求

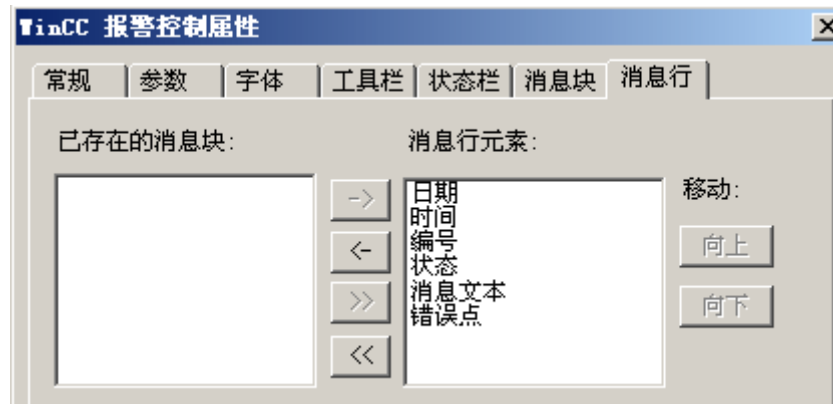
- 数据类型为“浮点数 32 位 IEEE754”的变量“Sysinfo_Drive_C”。必须以显示格式“空闲存储器 (%)”将驱动器“C”的“空闲磁盘空间”系统信息分配给该变量。
- 用于监视此变量限制值的消息文本和下限值。
- “报警记录运行系统”必须设置在计算机启动参数中。

步骤

1. 启动图形编辑器并打开画面。
2. 在画面中插入一个消息窗口。从“控件”对象选项板中选择“WinCC 报警控件”对象，并将其放置到画面中。
关闭快速组态对话框，不进行任何其它更改。
3. 双击插入的控件以打开“WinCC 报警控件属性”对话框。
4. 在“消息块”选项卡的“类型”域中选择“用户文本块”条目。在“选择”下，选中复选框“消息文本”和“错误点”。



5. 单击“消息行”选项卡。在“已存在的消息块”域中，选中条目“消息文本”和“错误点”，然后单击“右箭头”按钮将其传送到“消息行元素”域中。关闭对话框。



6. 保存该画面。
7. 单击图形编辑器工具栏中的相应按钮来激活运行系统。

参见

如何检查 WinCC 启动参数 (页 497)

如何启动运行系统 (页 498)

如何在“System Info”通道中组态变量 (页 487)

18.6.8 如何在状态显示中显示打印机状态

简介

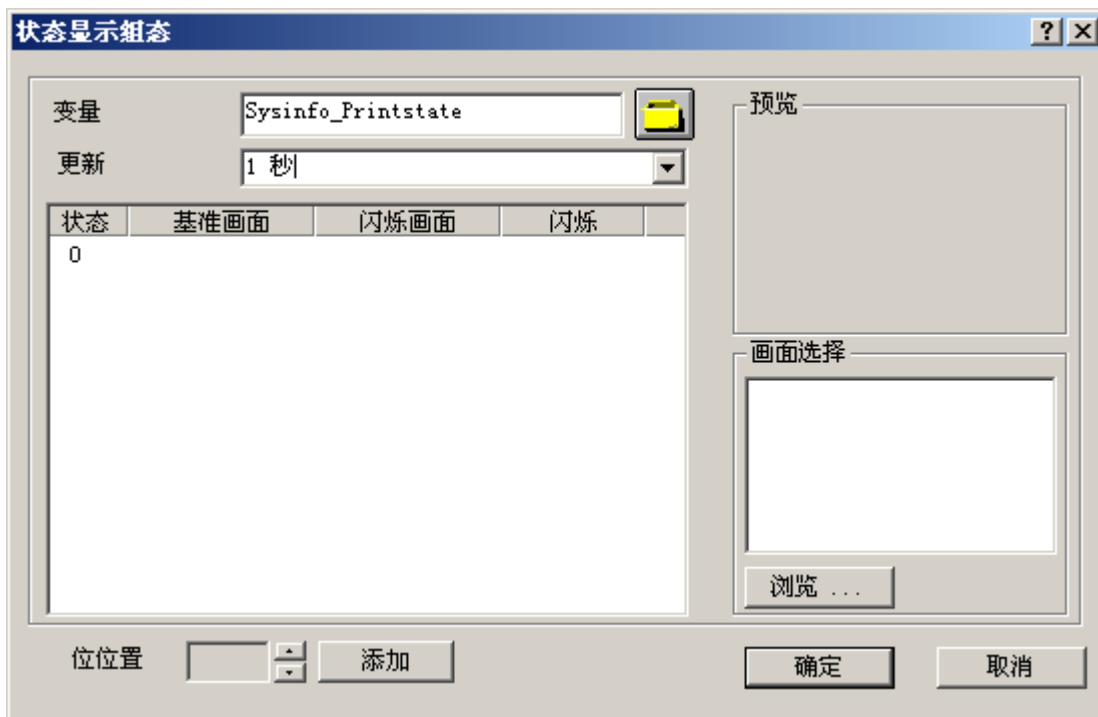
该实例基于“打印机状态”显示格式，显示可能的“打印机监控”系统信息评估。打印机或作业状态也可由组态消息评估，其由打印机或作业状态中的单一位触发。

要求

- 组态数据类型为“无符号 32 位数”的“Sysinfo_Printerstate”变量。该变量必须以“打印机状态”显示格式分配到“打印机监控”系统信息。
- 使用中的打印机必须支持这些状态显示以能够使用该系统信息。

步骤

1. 启动图形编辑器并打开画面。
2. 将状态显示插入到画面中。从对象选项板的“智能对象”中选择“状态显示”对象。将打开“状态显示组态”对话框。



3. 在“变量”域中，输入名称“Sysinfo_Printerstate”。
4. 将更新周期设置为“1 秒”。

5. 在“状态”列中选择数值“0”。从“画面选择”区域为该状态分配一个图标，如代表打印机的图标。选择期望的图标，用鼠标将其拖到“0”行并放于“基准画面”列。如果在“画面选择”区域中没有显示画面或期望的画面，可以单击“浏览...”按钮打开一个选择对话框。
6. 可根据需要使用“添加”按钮添加附加的位置，并将其它画面分配到这些状态。
7. 关闭对话框并保存画面。
8. 单击图形编辑器工具栏中的相应按钮来激活运行系统。

在运行系统中，准备就绪的打印机显示为在步骤 5 中分配的状态为“0”的图标。如果打印机有错误或您未运行步骤 6，则不显示任何图片。如果在步骤 6 中为其它位置分配了图标，则相应显示这些图标。

说明

应仔细考虑如何选择更新周期，因为它会影响计算机负载。因此，每 250 ms 更新一次时间显示将有损于系统性能。

参见

如何在“System Info”通道中组态变量 (页 487)

如何启动运行系统 (页 498)

18.6.9 如何检查 WinCC 启动参数

步骤

1. 在 WinCC 项目管理器中，选择“计算机”。
2. 打开快捷菜单并选择“属性”。
3. 将打开“计算机列表属性”对话框。单击“属性”按钮。
4. 将打开“计算机属性”对话框。
5. 单击“启动”标签并选中条目。必要时，可以激活或取消激活运行模块，或将其它应用程序添加到启动列表中。
6. 关闭打开的对话框。

18.6.10 如何插入棒图

步骤

1. 启动图形编辑器并打开画面。
2. 在“智能对象”下的“标准”对象选项板中，选择对象“棒图”。
3. 在画面中插入一个棒图。为此，将鼠标指向工作区中要放置棒图的位置。同时按住鼠标，拖动对象到期望的大小。
4. 释放鼠标按钮后，将打开“棒图组态”对话框。在该对话框中，输入 WinCC 变量的名称并设置更新的规范和限制。此外，还可以使用“棒图方向”来设置显示棒图的方向。
5. 关闭对话框。

18.6.11 如何插入 I/O 域

步骤

1. 启动图形编辑器并打开画面。
2. 在“智能对象”下的“标准”对象选项板中，选择对象“I/O 域”。
3. 在画面中插入 I/O 域。为此，将鼠标置于工作区中要放置 I/O 域的位置。同时按住鼠标，拖动对象到期望的大小。
4. 释放鼠标按钮后，将打开“I/O 域组态”对话框。在该对话框中，输入 WinCC 变量的名称并设置更新和域类型设置。此外，也可以选择用于显示该值的“字体”。
5. 关闭对话框。

18.6.12 如何启动运行系统

要求

激活运行系统前，必须定义启动画面。

步骤

1. 保存并关闭所有可能在编辑器中打开的文件。
2. 选择 WinCC 项目管理器。
3. 通过单击工具栏上的“激活”按钮或通过选择“文件”菜单中的“激活”来激活项目。

18.7 特殊功能

18.7.1 在多用户和客户系统中使用

18.7.1.1 在多用户和客户系统中使用

引言

在多用户和客户系统中，**System Info** 通道可用于在客户系统中处理服务器的系统信息。这样就可以在客户系统中，用单台 WinCC 客户机来监控多台服务器。

参见

在 WinCC 客户机上监控多台服务器的系统信息 (页 499)

18.7.2 监控多台服务器的系统信息的实例

18.7.2.1 在 WinCC 客户机上监控多台服务器的系统信息

引言

在该实例中，单台 WinCC 客户机监控两台服务器。监控的系统信息（例如可用的磁盘空间和 CPU 负载）显示在 WinCC 客户机的过程画面中。

这需要进行下列组态：

组态第一台服务器

组态第二台服务器

在 WinCC 客户机上导入变量

在 WinCC 客户机上组态过程画面

激活项目

要求

服务器和 WinCC 客户机必须通过 Windows 网络连接。

参见

- 如何激活项目 (页 505)
- 如何在 WinCC 客户机上组态过程画面 (页 503)
- 如何将变量导入 WinCC 客户机 (页 502)
- 如何组态第二台服务器 (页 501)
- 如何组态第一台服务器 (页 500)

18.7.2.2 如何组态第一台服务器

简介

本节介绍了第一台服务器的组态，这对该实例来说是必要的。

1. 组态“System Info”通道的变量以显示可用的硬盘空间和 CPU 负载。
2. 生成数据包。



所使用数据类型的表格

在该“System Info”通道实例中使用的变量名和格式如下表所示。

变量	函数	数据类型	格式
Sysvar_1_Drive_C	空闲硬盘容量	浮点数 32 位 IEEE 754	0-100% (空闲 %)
Sysvar_1_CPU	CPU 负载	浮点数 32 位 IEEE 754	0-100% (满载 %)

步骤

1. 在第一台服务器上创建名为“Testinfo_1”的多用户项目。在该项目中安装驱动程序“System Info”。
2. 从关联的“System Info”通道单元的快捷菜单中选择条目“新建连接”，并创建名为“Connection1”的连接。
3. 单击表格区下方的“变量”(Tags) 选项卡。
4. 单击“名称”(Name) 列最上方的空单元格。为该变量输入名称“Sysvar_1_Drive_C”。
5. 将“数据类型”(Data Type) 设置为“32 位浮点数 IEEE 754”(Floating-point number 32-bit IEEE 754)。

6. 打开“系统信息”(System Info)对话框。
为此,请单击“地址”(Address)字段,然后单击  按钮。
7. 将“功能”(Function)字段设置为“可用磁盘空间”(Available Drive Space),“驱动器”(Drive)设置为“C”,“格式”(Format)设置为“空闲容量 (%)”(Free capacity in %)。关闭对话框。
8. 单击“名称”(Name)列最上方的空单元格。
为该变量输入名称“Sysvar_1_CPU”。
9. 将“数据类型”(Data Type)设置为“32 位浮点数 IEEE 754”(Floating-point number 32-bit IEEE 754)。
10. 打开“系统信息”(System Info)对话框。
为此,请单击“地址”(Address)字段,然后单击  按钮。
11. 将“功能”(Function)字段中的值设置为“CPU 负载”(CPU Load),“格式”(Format)中的值设置为“总负载 (%)”(Total load in %)。关闭对话框。
12. 创建数据包。通过在浏览窗口中选择“服务器数据”(Server data)并打开快捷菜单来进行操作。选择菜单项“创建”(Create)。对数据包已创建的消息进行确认。

参见

如何组态第二台服务器 (页 501)

18.7.2.3 如何组态第二台服务器

简介

本节介绍了第二台服务器的组态,这对该实例来说是必要的。



1. 组态“System Info”通道的变量以显示可用的硬盘空间和 CPU 负载。
2. 生成数据包。

所使用数据类型的表格

在该“System Info”通道实例中使用的变量名和格式如下表所示。

变量	函数	数据类型	格式
Sysvar_2_Drive_C	空闲硬盘容量	浮点数 32 位 IEEE 754	0-100% (空闲 %)
Sysvar_2_CPU	CPU 负载	浮点数 32 位 IEEE 754	0-100% (满载 %)

步骤

1. 在第二台服务器上创建名为“Testinfo_2”的多用户项目。在该项目中安装驱动程序“System Info”。
2. 从关联的“System Info”通道单元的快捷菜单中选择条目“新建连接”，并创建名为“Connection2”的连接。
3. 单击表格区下方的“变量”(Tags)选项卡。
4. 单击“名称”(Name)列最上方的空单元格。
为该变量输入名称“Sysvar_2_Drive_C”。
5. 将“数据类型”(Data Type)设置为“32位浮点数 IEEE 754”(Floating-point number 32-bit IEEE 754)。
6. 打开“系统信息”(System Info)对话框。
为此，请单击“地址”(Address)字段，然后单击  按钮。
7. 将“功能”(Function)字段设置为“可用磁盘空间”(Available Drive Space)，“驱动器”(Drive)设置为“C”，“格式”(Format)设置为“空闲容量 (%)”(Free capacity in %)。
8. 单击“名称”(Name)列最上方的空单元格。
输入“Sysvar_2_CPU”作为变量名称。
9. 在连接快捷菜单中，选择“新建变量”。
10. 将“数据类型”(Data Type)设置为“32位浮点数 IEEE 754”(Floating-point number 32-bit IEEE 754)。
11. 打开“系统信息”(System Info)对话框。
为此，请单击“地址”(Address)字段，然后单击  按钮。
12. 将“功能”(Function)字段中的值设置为“CPU 负载”(CPU Load)，“格式”(Format)中的值设置为“总负载 (%)”(Total load in %)。
13. 创建数据包。通过在浏览窗口中选择“服务器数据”(Server data)并打开快捷菜单来进行操作。选择菜单项“创建”(Create)。对数据包已创建的消息进行确认。

参见

如何将变量导入 WinCC 客户机 (页 502)

18.7.2.4 如何将变量导入 WinCC 客户机

引言

本节介绍了 WinCC 客户机的组态，这对该实例来说是必要的。

1. 装载第一台服务器项目的数据包。
2. 装载第二台服务器项目的数据包。

要求

该实例需要使用两个服务器项目数据包。

服务器	项目	数据包
1	Testinfo_1	Testinfo_1_<计算机名称>
2	Testinfo_2	Testinfo_2_<计算机名称>

步骤

1. 在 WinCC 客户机上创建名为“mc_info”的客户项目。
2. 在服务器数据的快捷菜单中选择“装载”。将打开“打开”对话框。
3. 选择有第一台服务器项目“Testinfo_1”的计算机。
4. 在“<项目名称>\<计算机名称>\Packages”目录中选择数据包“estinfo_1_<计算机名称>.pck”。
5. 单击“打开”按钮，并在数据包打开后确认消息。
6. 在第二台服务器上装载数据包“Testinfo_2_<计算机名称>.pck”。为此，对“要求”表格中的第二个项目采用合适的设置和名称重复步骤 2 至 5。

参见

如何在 WinCC 客户机上组态过程画面 (页 503)

18.7.2.5 如何在 WinCC 客户机上组态过程画面

引言

本节阐明了 WinCC 客户机的组态，在该实例中要求在 WinCC 客户机上的过程画面中显示服务器的系统信息。

1. 组态第一台服务器的系统信息显示
2. 组态第二台服务器的系统信息显示

要求

该实例需要在客户机项目中装载服务器项目数据包。

数据包	项目	变量
Testinfo_1_<计算机名称>	Testinfo_1	Sysvar_1_Drive_C
Testinfo_1_<计算机名称>	Testinfo_1	Sysvar_1_CPU
Testinfo_2_<计算机名称>	Testinfo_2	Sysvar_2_Drive_C
Testinfo_2_<计算机名称>	Testinfo_2	Sysvar_2_CPU

步骤

1. 在 WinCC 客户机上，启动图形编辑器并创建名为“p_serverinfo”的画面。
2. 在画面中添加一个 I/O 域。从“智能对象”对象列表中选择“I/O 域”对象。将打开“I/O 域组态”对话框。
3. 单击按钮以进行变量选择。将打开“变量”对话框。
4. 选择第一个服务器项目“Testinfo_1”的变量“Sysvar_1_Drive_C”。为此，在数据包“Testinfo_1_<计算机名称>”下打开目录结构。关闭对话框。
5. 在“I/O 域组态”对话框中，将更新周期设置为“5 秒”。
6. 将域类型设置为“输出”。关闭对话框。
7. 将第二个 I/O 域插入画面中，并组态同一个项目的变量“Sysvar_1_CPU”。为此，对“要求”表格中的相应设置重复步骤 2 至 6。
8. 重复步骤 2 至 7 来组态第二个服务器项目“Testinfo_2”的变量。
9. 关闭对话框并保存画面。

说明

应仔细考虑如何选择更新周期，因为它会影响计算机负载。因此，每秒更新数据显示将有损于系统性能。

参见

如何激活项目 (页 505)

18.7.2.6 如何激活项目

引言

本节显示如何激活服务器和 WinCC 客户机上的项目。

1. 在服务器项目“Testinfo_1”中创建启动画面并激活项目。
2. 在服务器项目“Testinfo_2”中创建启动画面并激活项目。
3. 在 WinCC 客户机上定义项目“mc_info”的启动画面并激活项目。

步骤

1. 在服务器 1 上的项目“Testinfo_1”的浏览窗口中，选择图形编辑器并使用快捷菜单来创建新画面。
2. 将该画面设置为启动画面。为此，从快捷菜单中选择“设置为启动画面”。
3. 单击工具栏上的“激活”按钮来激活项目。
4. 在服务器 2 上的项目“Testinfo_2”的浏览窗口中，选择图形编辑器并使用快捷菜单来创建新画面。
5. 将该画面设置为启动画面。为此，从快捷菜单中选择“设置为启动画面”。
6. 单击工具栏上的“激活”按钮来激活项目。
7. 在 WinCC 客户机上的项目“mc_info”的浏览窗口中，选择图形编辑器。过程画面“p_serverinfo”将显示在数据窗口中。
8. 将该画面设置为启动画面。为此，从快捷菜单中选择“设置为启动画面”。
9. 单击工具栏上的“激活”按钮来激活项目。

通讯 - 诊断

19.1 通道和变量的诊断

本节说明通道及其变量以及内部变量的诊断。该诊断可用于如通讯故障或出现意料之外的变量值等情况。

参见

“SIMATIC S5 PROFIBUS FDL” 通道的诊断选项 (页 540)

变量的质量代码 (页 562)

使用全局动作监视变量状态 (页 572)

使用变量状态监视连接状态 (页 570)

如何检查内部变量 (页 573)

通道诊断 (页 509)

有关错误检测的常规信息 (页 508)

诊断“OPC” 通道的可能性 (页 550)

“SIMATIC S7 Protocol Suite” 通道 - 诊断选项 (页 528)

“System Info” 通道 - 诊断选项 (页 517)

19.2 有关错误检测的常规信息

在运行系统中通常会首先识别出在建立通讯链接时发生的故障或错误。

使用 WinCC 变量的动态对象（不能提供当前过程值给它们）在过程画面内将显示为未激活状态。例如，这些对象可以是 I/O 域、滚动条对象或棒图。

如果故障不影响连接的部分 WinCC 变量，就表示某个 WinCC 变量是故障源。此时，应检查变量的编址或者图形编辑器中变量的拼写是否正确。

如果故障影响连接的所有 WinCC 变量，就表示错误发生在连接本身。

以下章节描述定位错误源可采取的措施及可使用的工具。

19.3 通道诊断

19.3.1 通道诊断

以下功能可用于支持通道及其连接的诊断：

- “状态 - 逻辑连接”功能
- WinCC“通道诊断”

参见

如何使用“状态 - 逻辑连接”功能来检查通道 (页 509)

“状态 - 逻辑连接”功能 (页 509)

通道诊断的原理 (页 511)

19.3.2 “状态 - 逻辑连接”功能

通过“状态 - 逻辑连接”功能，WinCC 项目管理器允许用户方便地显示所有已组态连接的当前状态。但状态只能在运行系统中显示。

该功能在 WinCC 项目管理器中通过“工具”菜单启动。

说明

“WinCC 通道诊断”可提供关于连接状态的详细数据。

19.3.3 如何使用“状态 - 逻辑连接”功能来检查通道

简介

本节说明如何使用“状态 - 逻辑连接”功能来显示所有组态连接的当前状态。

要求

- 在本通道中组态一个通道，并在该通道中创建一个变量。

步骤

1. 单击 WinCC 项目管理器工具栏中的相应按钮来激活运行系统。
2. 在 WinCC 项目管理器的“工具”中，选择“驱动程序连接的状态”选项。将打开“状态 - 逻辑连接”对话框。



3. 在“更新”区域中，可以指定周期性地执行更新。否则，可以单击“更新”按钮手动启动显示的更新。
4. 组态的连接将显示在“名称”列中。“状态”列显示相应连接的状态。
5. 检查“状态”列中的条目。“Disconnected”状态表示组态或硬件错误。更多信息请参见相关通道的“通道诊断”。

19.3.4 使用“通道诊断”进行通道诊断

19.3.4.1 通道诊断的原理

引言

在运行系统中，“WinCC 通道诊断”为 WinCC 用户提供了快速浏览激活连接的状态的方法。一方面，“通道诊断”提供了有关通道单元的状态和诊断信息，另一方面，它可用作组态诊断输出的用户界面：

- 有关诸如过程画面中通讯状态的统计或信息的输出
- 文本输出到记录册文件中以用于错误分析和由服务进行纠正。
- 文本输出到跟踪文件中以帮助热线查明通讯故障的原因。

诊断模块可作为 **ActiveX** 控件插入过程画面，或作为独立的应用程序从 Windows“开始”菜单启动。该模块只显示支持诊断的通道的状态信息。

在针对通道诊断的描述中可以找到通道变量的诊断。

记录册文件

“通道诊断”将为每个组态的 WinCC 通道创建一个名为 <通道名称.log> 的记录册文件。系统在其中记录重要的信息和错误。确切的文本内容取决于通道。

文件和导出文本的创建不能组态。

记录册文件包含许多信息，例如：启动和结束消息、版本信息以及有关通讯错误的信息。

文件中的每个条目由日期和时间标志、标记名称和描述组成。文件总是在输入完成后立即保存，以确保（如发生断电时）所有的信息都可用。

跟踪文件

可以创建名为 <通道名称.trc> 的跟踪文件，为每个组态的 WinCC 通道保留错误消息和任何附加信息。可以在运行系统中选择是否使用跟踪文件。当激活跟踪功能时，将显示消息以警告链接的传播时间会受影响。

跟踪文件中的每个条目都有时间标志，后面带有标记名称和描述。

当启用跟踪功能时，所有记录在记录册中的信息也将写入跟踪文件。

19.3 通道诊断

记录在跟踪文件内的信息有助于热线定位通讯故障的原因。

说明

跟踪和记录册文件条目只以英文记录。

两个文件都保存在 WinCC 目录结构的“Diagnostics”目录下。

当前计数器数值不记录在这些文件中。

参见

使用 ActiveX 控件进行通道诊断 (页 512)

如何使用作为 ActiveX 控件的通道诊断来检查通道 (页 512)

使用“通道诊断”来诊断通道 (页 513)

如何使用通道诊断检查通道 (页 513)

如何组态通道的跟踪功能 (页 514)

如何启动运行系统 (页 515)

19.3.4.2 使用 ActiveX 控件进行通道诊断

引言

使用 ActiveX 控件“WinCC 通道诊断控件”也可以在过程画面中显示有关通道的状态信息。

ActiveX 控件可从图形编辑器的“Control”对象选项板得到并简单地插入到画面中。这样用户即可创建诊断过程画面，在其中用户可以查看运行系统中通讯的状态以及其它消息，而无需每次重新组态安排。

19.3.4.3 如何使用作为 ActiveX 控件的通道诊断来检查通道

引言

本节说明如何使用“WinCC Channel Diagnosis Control”ActiveX 控件组态通道的诊断。

要求

- 在本通道中组态一个通道，并在该通道中创建一个变量。

步骤

1. 启动图形编辑器并打开画面。
2. 将“WinCC Channel Diagnosis Control”ActiveX 控件插入画面。通过从“控件”对象选项板中选择 ActiveX 控件，将其插入画面中并拖动到期望的大小，也可以实现。
3. 保存该画面。
4. 单击图形编辑器工具栏中的相应按钮来激活运行系统。
5. 选择插入了 ActiveX 控件的画面。通道的状态信息将出现在“通道诊断”应用程序窗口的“通道/连接”标签中。
6. 单击“组态”标签。选择其中一个显示的通道并组态哪些出错消息将记录到相关的日志文件中。
7. 在“组态通道的跟踪功能”处可以找到关于激活跟踪功能的更多信息。

说明

“通道诊断”只显示支持通道诊断的通道的状态信息。

参见

如何组态通道的跟踪功能 (页 514)

19.3.4.4 使用“通道诊断”来诊断通道

引言

“通道诊断”也可作为应用程序独立于 WinCC 从 Windows“开始”菜单启动。

因此通道诊断总是可以使用且不受过程画面选择的约束，就如同“WinCC 通道诊断控件”的情况一样。当 WinCC 处于运行系统中时，状态信息只由“通道诊断”显示。

19.3.4.5 如何使用通道诊断检查通道

引言

本节说明如何将“通道诊断”作为应用程序从 Windows“开始”菜单处启动。

要求

- 在本通道中组态一个通道，并在该通道中创建一个变量。

步骤

1. 在 Windows“开始”菜单中，选择“Simatic\WinCC\工具”组中的条目“通道诊断”。
2. 将打开“通道诊断”应用程序窗口。如果运行系统中当前没有 WinCC 项目，则显示消息“没有连接到 WinCC”。
3. 单击 WinCC 项目管理器工具栏中的相应按钮来激活运行系统。
4. 选择插入了 ActiveX 控件的画面。通道的状态信息将出现在“通道诊断”应用程序窗口的“通道/连接”标签中。
5. 单击“组态”标签。选择其中一个显示的通道并组态哪些出错消息将记录到相关的日志文件中。
6. 在“组态通道的跟踪功能”处可以找到关于激活跟踪功能的更多信息。

说明

“通道诊断”只显示支持通道诊断的通道的状态信息。

参见

如何组态通道的跟踪功能 (页 514)

如何启动运行系统 (页 515)

19.3.4.6 如何组态通道的跟踪功能

引言

本节说明如何在运行系统中组态和激活通道的跟踪功能。有关错误和通讯状态的附加信息将记录在跟踪文件内。

说明

记录在跟踪文件内的信息有助于热线定位通讯故障的原因。因此，此处不详细描述文件中数据的分析。

要求

- 组态通道加连接以及变量。
- 激活 WinCC 项目。

标准标记概述

标记	描述
致命错误	严重错误（要求用户干预）
错误	错误（例如帧出错）
警告	警告（例如校验错误）
信息	信息（例如功能调用）
成功	成功执行（例如成功完成功能调用）
选中用户标记	允许“用户标记”复选框

步骤

1. 从“开始”菜单启动 WinCC 通道诊断。
2. 在“组态”标签上，选择期望的通道。
3. 在“标记”下，激活要记录在跟踪文件内的状态和错误消息。在“缺省标记概述”表中可以找到对标准标记的描述。
4. 如果要将“用户标记”记录到跟踪文件，请选中“检查用户标记”复选框。“用户标记”的数量和含义取决于通道。
5. 选中期望的“用户标记”的复选框。通过单击“置位”或“复位”按钮，可以置位或复位所有“用户标记”。
6. 在“跟踪文件”部分，选中“允许”复选框。这样可以激活该区域内的其它域。
7. 在“最多文件”域中，输入跟踪文件的最大数目。
8. 在“最大大小”域中，设置单个跟踪文件的最大大小。
9. 如果要在文件数目和文件大小达到最大后覆盖通道现有的跟踪文件（从最早的开始），请激活“覆盖”域。
10. 单击“保存”以保存设置并激活更改。

19.3.4.7 如何启动运行系统

要求

激活运行系统前，必须定义启动画面。

步骤

1. 保存并关闭所有可能在编辑器中打开的文件。
2. 选择 WinCC 项目管理器。
3. 通过单击工具栏上的“激活”按钮或通过选择“文件”菜单中的“激活”来激活项目。

参见

如何使用通道诊断检查通道 (页 513)

19.4 “System Info” 通道的诊断

19.4.1 “System Info” 通道 - 诊断选项

可以使用“System Info” 通道诊断的下列选项或其变量之一：

使用“通道诊断” 诊断通道

“通道诊断” 可以查询运行系统中通道和连接的状态。产生的任何错误都用“错误代码” 显示。

诊断通道变量

在运行系统的变量管理器中，可以查询当前值、当前质量代码和变量改变的最后时刻。

参见

如何检查变量 (页 521)

如何检查通道和连接 (页 519)

19.4.2 日志文件条目的描述

引言

在日志文件中记录通道的错误和状态的重要变化。这些条目可以用于分析通讯问题。

文件中的每个条目都有日期和时间标志，后面带有标记名称和描述。

记录册条目实例：

```
2000-03-10 12:00:21,050 INFO Log starting ...
2000-03-10 12:00:21,050 INFO | LogFileName :C:\Siemens\WinCC\Diagnose
\SYSTEM_INFO_01.LOG
2000-03-10 12:00:21,050 INFO | LogFileCount :3
2000-03-10 12:00:21,050 INFO | LogFileSize :1400000
2000-03-10 12:00:21,050 INFO | TraceFlags :fa000001
```

19.4 “System Info” 通道的诊断

2000-03-10 12:00:21,050 INFO start timer

2000-03-10 12:00:21,360 ERROR Illegal tag type! tag: "Format_0" correct type:"Text Tag 8-Bit Character Set"!

“INFO” 标记条目

消息文本	描述
Log starting ...	起始消息
LogFileName : C:\ Siemens\ WinCC\ Diagnose\ "channel_name".LOG	带路径日志文件的名称
LogFileCount : "n"	通道的日志文件的数目
LogFileSize : "x"	单个日志文件的大小（以字节为单位）
TraceFlags :fa000001	通道使用的十六进制格式标记
start timer	起始消息

“ERROR” 标记条目

消息文本	描述
Illegal tag type! tag:"tag" correct type:"data type"!	变量的数据类型不正确 变量 = 数据类型错误的变量的名称 数据类型 = 正确的数据类型

19.4.3 确定错误变量值的原因

19.4.3.1 如何确定错误变量的原因

如果在运行系统中产生意外的变量值，进行如下处理以确定原因：

1. 检查通道和连接
2. 检查通道变量

参见

如何检查变量 (页 521)

如何检查通道和连接 (页 519)

19.4.3.2 如何检查通道和连接

简介

本节描述如何在运行系统中检查“System Info”通道及其连接。

要求

- 为“System Info”通道组态连接和变量。
- 激活 WinCC 项目。

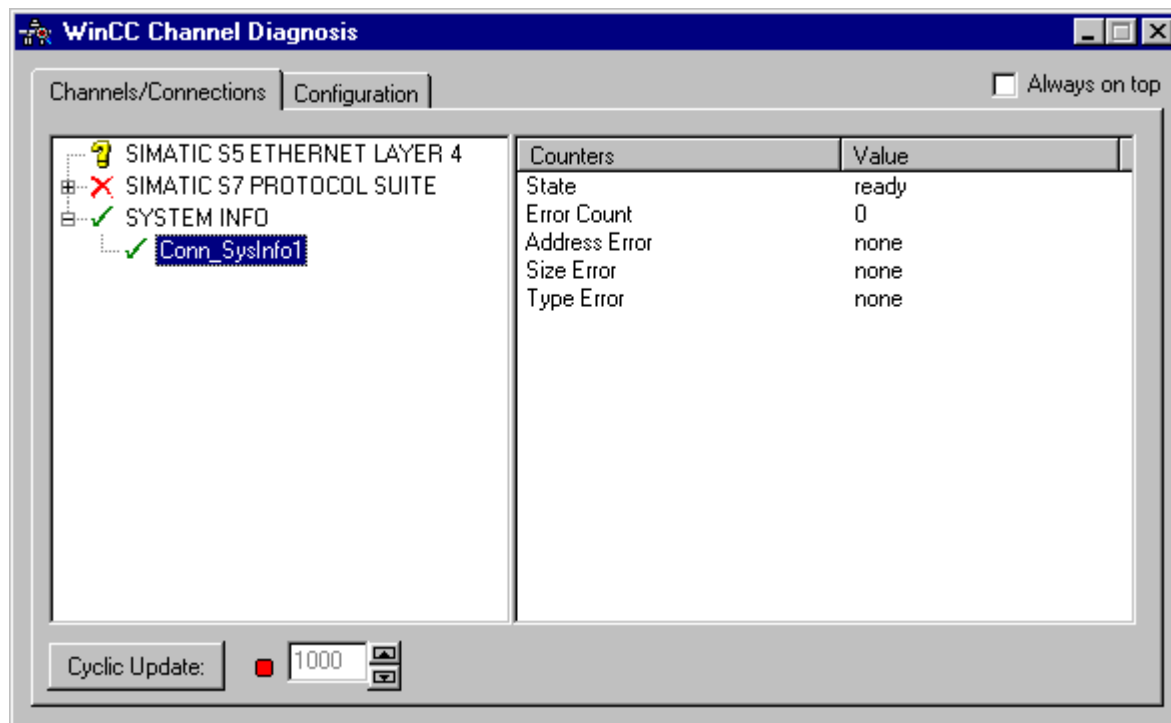
状态消息概述

图标	描述
	通道/连接无限制准备就绪
	通道/连接准备就绪，带有一些限制
	通道/没有关于连接状态的陈述
	通道/连接失败

19.4 “System Info” 通道的诊断

步骤

1. 从“开始”(Start) 菜单启动 WinCC 通道诊断。
2. 将打开“通道诊断”(Channel Diagnosis) 应用程序窗口。所有安装的通道及其连接的状态信息显示在“通道/连接”(Channels/Connections) 选项卡的左侧。



3. 检查通道“System Info”前的符号及其连接。如果通道和连接状况正常，将有一个绿色的复选标记显示在相应条目的前面。有关各个图标含义的信息，请参见表格“状态消息概述”。
4. 如果通道名称和连接前面没有绿色复选标记，则选择左侧窗口中的连接。在窗口右边，检查“地址错误”、“大小错误”和“类型错误”的计数器数值。这些值表示检测到的错误。
5. 检查通道特定的日志文件。要执行此操作，请使用文本编辑器打开目录“Siemens\WinCC \Diagnose”中的文件。检查带“ERROR”标记的最新条目。有关该主题的更多信息，请参阅“日志文件条目的描述”。
6. 如果检查日志文件之后仍然不能查明错误，请激活跟踪功能并联系客户支持部门。有关该主题的更多信息，请参见“组态通道的跟踪功能”。

参见

如何组态通道的跟踪功能 (页 514)

日志文件条目的描述 (页 517)

如何检查变量 (页 521)

19.4.3.3 如何检查变量

简介

如果运行系统中外部变量不具有预期的数值，请使用下列步骤来检查变量。

要求

- 为“System Info”通道组态连接和变量。
- 激活 WinCC 项目。

步骤

1. 在 WinCC 项目管理器的变量管理器中，选择“System Info”通道。
2. 在数据窗口中，选择希望检查的外部变量。为此，打开目录结构直到此变量显示在表格区域中。
3. 移动鼠标指针，指向要检查的变量。将打开工具提示窗口，显示当前变量值、质量代码和最近一次数值改变的时间。
4. 检查质量代码。如果显示数值“80”，则该变量值正确。其它数值的描述可以参见“变量质量代码”。
5. 如果质量代码不等于“80”，从变量管理器中选择变量，然后单击快捷菜单中的“属性”(Properties) 来打开“变量属性”(Tag Properties) 对话框。
6. 检查是否已在“限制/报告”(Limits/Reportin) 选项卡上组态了上限或下限、启动或替换值。这些数值会影响显示。
7. 如果变量值被其中一个已组态的数值改变，则取消激活该项目并改变该限制值或替换值。

说明

变量值、质量代码等只在运行期间显示。

参见

变量的质量代码 (页 562)

19.5 诊断通道“SIMATIC S7-1200/S7-1500”

19.5.1 通过系统诊断控件进行系统诊断

概述

系统诊断指示“S7-1200/1500”控制器的故障和错误。

通过 SysDiagControl，WinCC 提供了快速错误定位概述。

在系统诊断控件中，可以指定在报警视图中立即向操作员显示特定控制器状态的诊断概述。其中会显示控制器错误的详细信息。

要求

- 已将 "SIMATIC S7-1200, S7-1500 Channel" 通道集成到项目中。
- 已在“OMS+”通道单元中创建连接。
- 如果希望在诊断缓冲区视图中显示 S7-1500 通道的消息和文本，则必须将控制器中的消息和文本列表条目加载到运行系统中的 WinCC Configuration Studio/报警记录的“AS 消息”(AS messages) 区域中。

系统诊断视图

系统诊断显示中以下视图可用：

- 诊断概述
- 详细视图
- 诊断缓冲区视图

系统诊断显示也提供显示的拆分视图。您可以同时查看控制器和相关的详细信息。区域上方显示系统诊断和诊断缓冲区视图。区域下方显示详细视图。

诊断概述

诊断概述显示所有可用的 S7-1200/1500 通道。双击控制器打开详细视图。第一列中的符号给出了关于控制器当前状态的信息。

The screenshot shows a software interface with a toolbar at the top containing icons for a document, home, back, forward, and mail. Below the toolbar is a section titled "Diagnostic overview" containing a table with the following data:

Status	Name	Operating mode	Address	Plant designation	Sub-system
✓	Plant				
✓	S7-1200-St		32*		0
✓	S7-1200-St		32*		0
✓	S71500/ET		32*		0

详细视图

详细视图给出了关于所选控制器的详细信息。检查详细视图中的数据是否正确。在详细视图中无法对错误文本进行排序。下图显示了诊断概述和详细视图的拆分视图。

19.5 诊断通道“SIMATIC S7-1200/S7-1500”

The screenshot shows the 'Diagnostic overview' window in WinCC. It contains a table with columns: Status, Name, Operating mode, Address, Plant designation, and Sub-system. The third row is selected, showing 'S71500/ET' with status 'Good' and address '32*'. Below the table is a 'Property' list for the selected station, including Status (Good), Name (S71500/ET200MP-Station_1), Address (32*), and various other technical details.

Status	Name	Operating mode	Address	Plant designation	Sub-system
✓	Plant				
✓	S7-1200-St		32*		0
✓	S7-1200-St		32*		0
✓	S71500/ET		32*		0

Property	Value
> Status	✓ Good
> Name	S71500/ET200MP-Station_1
> Operating mode	N/A
> Rack	
> Slot	
> Type	S71500/ET200MP-Station
> Item number	
> Address	32*
> Plant designation	
> Location identifier	
> Sub-system	0
> Station	
> Subslot	
> Sub-address	
> Software version	
> Installation	
> Additional information	
> Error Text	
> Manufacturer ID	
> Hardware version	
> Profile ID	
> Specific profile details	
> I&M data version	
> Serial number	
> Revision Counter	

诊断缓冲区视图

诊断缓冲区视图显示了控制器诊断缓冲区中的当前数据。仅可以在诊断概述中调用诊断缓冲区视图。

要更新诊断缓冲区视图，请选择“更新”(Update)按钮。



No.	Date	Time	Event	Status
1	4/3/2015	7:57:01 PM		✓
2	4/3/2015	7:54:51 PM		✓
3	3/20/2015	7:07:24 PM		✓
4	3/20/2015	6:57:44 PM		✓
5	3/9/2015	9:50:13 AM		✓
6	3/7/2015	4:50:37 PM		✓
7	3/7/2015	4:45:22 PM		✓
8	3/7/2015	4:45:22 PM		✓
9	3/7/2015	4:45:22 PM		✓
10	3/7/2015	4:45:17 PM		✓
11	3/7/2015	4:44:40 PM		✓
12	3/6/2015	3:59:29 PM		✓
13	3/2/2015	10:37:09 AM		✓
14	3/2/2015	10:39:36 AM		✓
15	3/1/2015	3:06:48 AM		✓
16	3/1/2015	2:59:38 AM		✓
17	2/26/2015	12:18:15 PM		✓

No.: 1
Date: 4/3/2015
Time: 7:57:01 PM

系统诊断视图中的按钮

按钮	功能
	打开组态对话框，可在其中更改系统诊断控件的属性。
	打开子设备，如果没有子设备，则打开详细视图。
	打开父设备，如果没有父设备，则打开诊断概述。
	打开诊断概述。
	打开诊断缓冲区视图。 仅在诊断概述中可见。
	更新诊断缓冲区视图。
	打开一个对话框，用于设置所显示诊断概述列的用户定义排序标准。

19.5 诊断通道“SIMATIC S7-1200/S7-1500”

按钮	功能
	启动显示值的打印输出。用于打印的打印作业在组态对话框的“常规”(General) 选项卡上定义。
	此按钮用于将全部或选定的运行系统数据导出到“CSV”文件。如果激活“显示对话框”(Display dialog) 选项，则会打开一个对话框，从中可查看导出设置并启动导出。如果具有相应权限，还可以选择要导出的文件和目录。如果显示了一个对话框，则立即开始将数据导出到预定义文件。

19.5.2 如何组态系统诊断

简介

控制器中的故障和错误显示在运行系统的各种系统诊断视图中。可以在图形编辑器中组态相应的 WinCC 系统诊断控件。

要求

- 已在图形编辑器中打开画面。

组态步骤

1. 将系统诊断控件插入图形编辑器的画面中。
2. 在“常规”(General) 选项卡中，组态系统诊断控件的基本属性。
 - 诊断窗口的属性
 - 控件的常规属性
 - 控件的时间基准
3. 在“列”(Columns) 选项卡中，指定控制器数据以列或行的形式显示在系统诊断的视图中。
4. 使用排序对话框确定数据排序采用的列。更多详细信息，请参见页面 AUTOHOTSPOT 上用户归档控件的示例。
5. 在“参数”(Parameter)、“显示”(Display) 和“标记”(Marker) 选项卡中，组态表格的显示和属性。更多详细信息，请参见 AUTOHOTSPOT。
6. 在相应选项卡中，组态表格窗口的工具栏和状态栏。更多详细信息，请参见 AUTOHOTSPOT。

7. 如果希望从报警控件中所选的 AS 报警直接跳转到系统诊断控件，可以通过脚本在画面中组态按钮：
 - 在画面中插入按钮。例如，以事件形式创建脚本以执行鼠标单击动作。
 - 当报警控件“AlarmControl_1”和系统诊断控件“SysDiagControl_1”在同一画面时，可以使用下列脚本示例：
使用 C：

```
SetPropChar(lpszPictureName, "SysDiagControl_1", "NavigateTo",  
GetPropChar(lpszPictureName, "AlarmControl_1", "DiagnosticsContext"));
```


使用 VBS：

```
ScreenItems("SysDiagControl_1").NavigateTo =  
ScreenItems("AlarmControl_1").DiagnosticsContext
```
8. 保存组态数据。

19.6 “SIMATIC S7 Protocol Suite” 通道的诊断

19.6.1 “SIMATIC S7 Protocol Suite” 通道 - 诊断选项

有以下选项可用于错误检测和“SIMATIC S7 Protocol Suite” 通道或其变量之一的诊断：

检查通讯处理器组态

除了检查相应访问点，可以使用应用程序“设置 PG/PC 接口”来测试通讯处理器。也可以在 SIMATIC NET 中以相同的方式检查通讯处理器。

检查连接和变量的组态

系统和连接参数的组态可能出现错误。无效变量值还可能由于错误引用 AS 中的变量编址而引起。

使用“通道诊断”诊断通道

“通道诊断”可以查询运行系统中通道和连接的状态。产生的任何错误都用“错误代码”显示。

诊断通道变量

在运行系统的变量管理器中，可以查询当前值、当前质量代码和变量改变的最后时刻。

参见

如何检查变量 (页 538)

如何检查通道和连接 (页 536)

如何检查连接和变量的组态 (页 535)

检查 SIMATIC NET 中的通讯处理器 (页 534)

如何检查通信处理器的组态 (页 532)

19.6.2 日志文件条目的描述

引言

在日志文件中记录通道的错误和状态的重要变化。这些条目可以用于分析通讯问题。

文件中的每个条目都有日期和时间标志，后面带有标记名称和描述。

记录册条目实例：

```
01.04.99 12:00:24.524 INFO Log starting ...
1999-04-01 12:00:24,524 INFO LogFileName :C:\Siemens\WinCC\Diagnose
\SIMATIC_S7_Protocol_Suite_01.LOG
1999-04-01 12:00:24,524 INFO LogFileCount :3
1999-04-01 12:00:24,524 INFO LogFileSize :1400000
1999-04-01 12:00:24,524 INFO TraceFlags :c4000000
1999-04-01 12:00:24,524 INFO S7 channel DLL started!
1999-04-01 12:00:26,096 ERROR Illegal tag address "nCPU3_1"!
1999-04-01 12:00:27,428 INFO S7DOS release: @(#)TIS-Block Library DLL Version
C5.0.17.3-REL5,0,17,47,3-BASIS
1999-04-01 12:00:27,428 INFO S7DOS version:\V5.0 / 0
1999-04-01 12:00:27,428 INFO S7CHN version:\V5.0 / Mar 1 1999 / 22:36:40
1999-04-01 12:00:27,428 INFO S7 channel unit "Industrial Ethernet" activated!
1999-04-01 12:00:27,468 ERROR Cannot connect to "CPU_4":Errorcode 0xFFDF 42C2!
1999-04-01 12:00:27,538 INFO S7 channel unit "MPI" activated!
```

“INFO” 标记中最重要条目的描述

消息文本	描述
LogFileName : C:\ Siemens\ WinCC\ Diagnose\ "channel_name".LOG	带路径日志文件的名称
LogFileCount : "n"	通道的日志文件的数目
LogFileSize : "x"	单个日志文件的大小（以字节为单位）

19.6 “SIMATIC S7 Protocol Suite” 通道的诊断

消息文本	描述
TraceFlags :c4000000	用十六进制数字显示跟踪功能使用的标记
S7 channel DLL started!	起始消息
S7 channel DLL terminated!	结束消息
S7 channel unit "unitname" activated!	通道单元激活
S7 channel unit "unitname" deactivated!	通道单元取消激活
S7DOS version:versionsstring	版本信息
S7CHN version:versionsstring	版本信息

“ERROR” 标记中最重要条目的描述

消息文本	描述
Cannot connect to "connectionname": Errorcode 0xhhhh ffff!	<p>通讯错误</p> <p>激活 WinCC 后，无法立即建立与自动化系统的连接。如果可以至少一次没有错误地建立连接，以后出错时，则输出下列信息。</p> <p>nnn = 本连接的断开次数 连接名称 = 连接的名称 hhh = 第一个错误代码，十六进制 S7DOS/SAPI-S7 ffff = 第二个错误代码，十六进制 S7DOS/SAPI-S7</p>
Cannot connect to "connectionname": Errorcode 0xhhhh ffff!	<p>通讯错误</p> <p>激活 WinCC 后，无法立即建立与自动化系统的连接。至少一次正确地建立了连接。</p>
Channel API error:errorstring	<p>通道 API 出错</p> <p>通道将错误字符串“errorstring”传到 WinCC 项目管理器。根据错误的重要程度，错误字符串可能显示也可能不显示在注意框中。至于错误字符串的说明，请参见 API 错误文本。</p>
Max. count of API errors reached - API logbook deactivated	<p>通道 API 出错</p> <p>根据错误和功能，可能在 API 上周期性发生错误。要避免这些错误信息占满记录册文件，一个 API 错误最多可以输出 32 条信息。</p>

消息文本	描述
Cannot write storage data! Cannot read storage data / use default data Storage data illegal or destroyed / use default data! No storage data / use default data!	常规通道错误消息
单元“单元名”中的设备名从“旧设备名”改为“新设备名”	初始化消息
达到记录册极限大小 - 记录册被取消激活	日志文件超出其最大长度时发送的消息。监控记录册输出的长度。如果达到了指定的长度，记录册将取消激活。只有在消息输出导致超出最大文件长度时才会输出消息。如果使用编辑器改变文件长度或在 INI 文件中减少最大文件长度，不会输出任何消息！

19.6.3 确定错误变量值的原因

19.6.3.1 如何确定错误变量的原因

如果在运行系统中产生意外的变量值，进行如下处理以确定原因：

1. 检查通讯处理器的组态
2. 检查 SIMATIC NET 中的通讯处理器
3. 检查连接和变量的组态
4. 检查通道和连接
5. 检查通道变量

参见

如何检查变量 (页 538)

如何检查通道和连接 (页 536)

如何检查连接和变量的组态 (页 535)

检查 SIMATIC NET 中的通讯处理器 (页 534)

如何检查通信处理器的组态 (页 532)

19.6 “SIMATIC S7 Protocol Suite” 通道的诊断

19.6.3.2 如何检查通信处理器的组态

简介

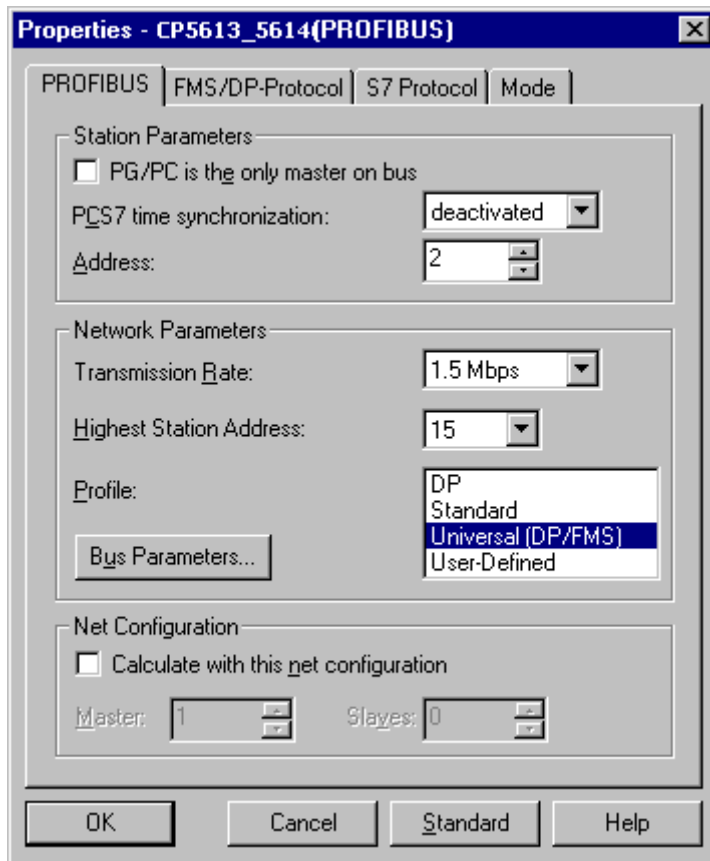
本部分说明如何使用“PG/PC 端口”程序检查通信处理器。在本例中，“CP 5613”类型的处理器用于 PROFIBUS 通信。

要求

- 安装 CP 5613。
- 安装相关的通信驱动程序。
- 组态 CP 5613。

步骤

1. 在控制面板中，单击“设置 PG/PC 端口”(Set PG/PC Port) 图标。将打开“设置 PG/PC 端口”(Set PG/PC Port) 对话框。
2. 检查访问点条目。安装 CP 5613 后，将自动为 Profibus 连接添加访问点“CP_L2_1:”。选择该访问点的条目。单击“属性”(Properties) 打开“属性 - CP5613_5614 (PROFIBUS)” (Properties - CP5613_5614 (PROFIBUS)) 对话框。



3. 检查“PROFIBUS”选项卡上的条目。
4. 单击“操作状态”(Operational State) 选项卡。单击“测试”(Test) 按钮在 CP 5613 上执行功能测试。测试结果将显示在下面的输出字段中。根据测试结果，可单击“重启”(Restart) 按钮执行 CP 5613 的复位和完全重启。
5. 单击“确定”(OK) 关闭所有打开的对话框。

参见

如何检查连接和变量的组态 (页 535)

检查 SIMATIC NET 中的通讯处理器 (页 534)

19.6.3.3 检查 SIMATIC NET 中的通讯处理器

简介

本节说明如何使用 SIMATIC NET 软件中的“设置 PC 站”程序检查通信处理器。

在本实例中，PROFIBUS 通信的“CP 5613”用于“SIMATIC S7 Protocol Suite”通道。

要求

- 安装 CP 5613。
- 安装 SIMATIC NET 软件。
- 在 SIMATIC NET 中组态 CP 5613。

操作步骤

1. 激活 Windows 开始菜单，打开“SIMATIC / SIMATIC NET / 设置”(SIMATIC / SIMATIC NET / Settings) 并选择菜单项“设置 PC 站”(Set PC Station)。将打开“组态控制台 PC 站”(Configurations Console PC Station) 对话框。
2. 检查访问点条目。在浏览窗口中选择“访问点”(Access Point) 目录。已存在的访问点显示在数据窗口中。安装 CP 5613 时，自动为 Profibus 连接插入访问点“CP_L2_1:”。在数据窗口中选择该访问点。使用快捷菜单中的“属性”(Properties) 菜单项打开“CP_L2_1 的属性:”(Properties of CP_L2_1:) 对话框。
3. 检查“分配的接口参数”(Assigned Interface Parameters) 字段内的条目。如果是 PROFIBUS 网络中的 CP5613，选择条目“CP5613_5614 (PROFIBUS)”。
4. 打开浏览窗口，选择“组件”(Components) 目录，然后选择“CP5613/CP5614”(CP5613/CP5614) 子目录。
5. 选择“网络诊断”(Network Diagnosis) 目录。单击“测试”(Test) 按钮在 CP 5613 上执行功能测试。测试结果显示在输出窗口中。根据测试结果，单击“常规”(General) 目录中的“重新启动”(Restart) 按钮以执行 CP 5613 的复位，然后完全重新启动。
6. 检查连接到“总线成员”(Bus Participants) 目录列表中 PROFIBUS 的成员列表。根据显示，可确定您自己工作站的功能和状态以及其它连接成员的功能和状态。
7. 关闭对话框。
8. 如果在通信处理器组态中检测到故障，只能使用 SIMATIC NET 工具修改组态。详细信息请参见 SIMATIC NET。

参见

如何检查连接和变量的组态 (页 535)

19.6.3.4 如何检查连接和变量的组态

简介

本节说明如何检查系统参数以及连接和变量的组态。在本实例中，“CP 5613”通信处理器用于 PROFIBUS 通信。

要求

- 安装 CP 5613。
- 安装相关的通信驱动程序。
- 组态 CP 5613。
- 为“SIMATIC S7 Protocol Suite”通道组态连接和变量。
- 激活 WinCC 项目。

步骤

1. 在 WinCC 项目管理的变量管理器中，选择“SIMATIC S7 Protocol Suite”通道。在数据窗口中，选择“PROFIBUS”通道单元。在通道单元快捷菜单上单击“系统参数”。将打开“系统参数 - PROFIBUS”对话框。
2. 在“单元”标签上，检查“逻辑设备名称”域中的条目。默认状态下，此条目设置为访问点“CP_L2_1:”。在 CP 5613 系统中安装通信处理器期间分配访问点。关闭对话框。



3. 在变量管理器浏览窗口中，选择“PROFIBUS”通道单元。在数据窗口中，选择要检查的连接。在快捷菜单中，单击“属性”打开“连接属性”对话框。
4. 单击“属性”按钮以打开“连接参数 - PROFIBUS”对话框。
5. 检查“连接”标签上的设置。关闭打开的对话框。

19.6 “SIMATIC S7 Protocol Suite” 通道的诊断

6. 在浏览窗口中，选择经检查的连接。在数据窗口中，选择要检查的变量。在快捷菜单中，单击“属性”打开“变量属性”对话框。检查“类型转换”和“数据类型”域中的数值。
7. 单击“选择”按钮以打开“地址属性”对话框。检查自动化系统中变量的地址设置。
8. 单击“确定”(OK) 关闭所有打开的对话框。

参见

如何检查通道和连接 (页 536)

19.6.3.5 如何检查通道和连接





简介

本节说明如何在运行系统中检查“SIMATIC S7 Protocol Suite”通道及其连接。

要求

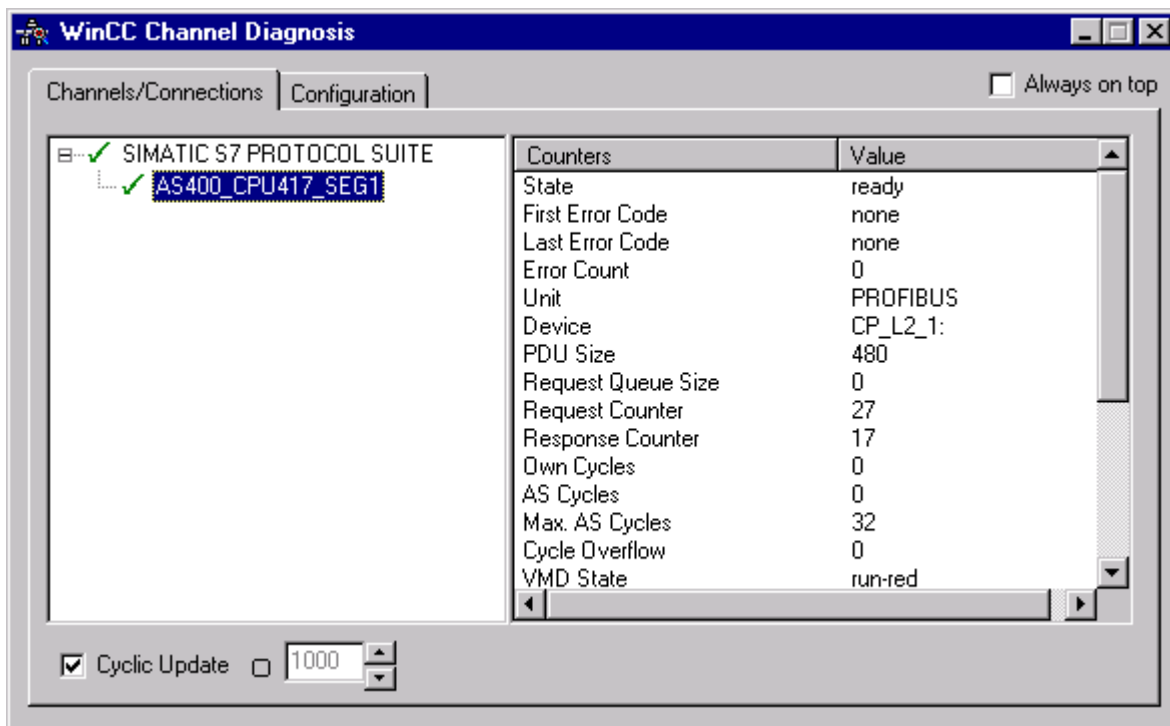
- 在 WinCC 计算机中安装通信处理器，例如用于 PROFIBUS 通信的 CP 5613。
- 安装相关的通信驱动程序。
- 组态 CP 5613。
- 创建 STEP7 项目。
- 为“SIMATIC S7 Protocol Suite”通道组态连接和变量。
- 激活 WinCC 项目。

状态消息概述

图标	描述
	通道/连接无限制准备就绪
	通道/连接准备就绪，带有一些限制
	通道/没有关于连接状态的陈述
	通道/连接失败

步骤

1. 从“开始”(Start) 菜单启动 WinCC 通道诊断。
2. 将打开“通道诊断”(Channel Diagnosis) 应用程序窗口。所有安装的通道及其连接的状态信息显示在“通道/连接”(Channels/Connections) 选项卡的左侧。



3. 选中名称为“SIMATIC S7 Protocol Suite”的通道和连接前面的图标。如果通道和连接状况良好，将有一个绿色的复选标记显示在相应条目的前面。有关各个图标含义的信息，请参见表格“状态消息概述”。
4. 如果通道名称和连接前面没有绿色复选标记，则选择左侧窗口中的连接。在右边的窗口中，检查计数器“第一个错误代码”和“最后一个错误代码”的条目。这些值表示检测到的错误。单击数值显示的快捷菜单可以访问直接帮助。
5. 在“组态”标签上，选择将输出到日志文件的状态和错误消息。这通过选择“SIMATIC S7 Protocol Suite”并组态出错显示得以实现。有关详细信息，请参见“组态通道的日志文件”。
6. 检查通道特定的日志文件。要执行此操作，请使用文本编辑器打开目录“Siemens\WinCC \Diagnose”中的文件。检查带“ERROR”标记的最新条目。有关该主题的更多信息，请参见“日志文件条目的描述”。
7. 如果检查日志文件之后仍然不能查明错误，请激活跟踪功能并联系客户支持部门。有关该主题的更多信息，请参见“组态通道的跟踪功能”。

19.6 “SIMATIC S7 Protocol Suite” 通道的诊断

参见

- 如何组态通道的跟踪功能 (页 514)
- 日志文件条目的描述 (页 529)
- 如何检查变量 (页 538)

19.6.3.6 如何检查变量

简介

如果运行系统中外部变量不具有预期的数值，请使用下列步骤来检查变量。

在“SIMATIC S7 Protocol Suite”通道中，也可以使用连接专用的内部变量。该步骤也可以用于检查这些变量。

要求

- 在 WinCC 计算机中安装通信模块，例如用于 MPI 通信的 CP 5613。
- 安装相关的通信驱动程序。
- 组态 CP 5613。
- 创建 STEP7 项目。
- 为“SIMATIC S7 Protocol Suite”通道组态连接和变量。
- 激活 WinCC 项目。

步骤

1. 在 WinCC 项目管理器的变量管理器中，选择“SIMATIC S7 Protocol Suite”通道。
2. 在数据窗口中，选择希望检查的外部变量。为此，打开目录结构直到此变量显示在表格区域中。
3. 移动鼠标指针，指向要检查的变量。将打开工具提示窗口，显示当前变量值、质量代码和最近一次数值改变的时间。
4. 检查质量代码。如果显示数值“80”，则该变量值正确。其它数值的描述可以参见“变量质量代码”。
5. 如果质量代码不等于“80”，从变量管理器中选择变量，然后单击快捷菜单中的“属性”(Properties) 来打开“变量属性”(Tag Properties) 对话框。
6. 检查是否已在“限制/报告”(Limits/Reportin) 选项卡上组态了上限或下限、启动或替换值。这些数值会影响显示。
7. 如果变量值被其中一个已组态的数值改变，则取消激活该项目并改变该限制值或替换值。

说明

在运行系统中，可以使用“WinCC 通道诊断”工具详细查看连接专用的内部变量的当前值。选择主连接后，变量将显示在“计数器”列中。

变量值、质量代码等只在运行期间显示。

参见

变量的质量代码 (页 562)

19.7 “SIMATIC S5 Profibus FDL” 通道的诊断

19.7.1 “SIMATIC S5 PROFIBUS FDL” 通道的诊断选项

以下选项可用于错误检测和“SIMATIC S5 PROFIBUS FDL” 通道诊断或其变量诊断：

检查通讯处理器组态

除了检查相应访问点，可以使用应用程序“设置 PG/PC 接口”来测试通讯处理器。也可以在 SIMATIC NET 中以相同的方式检查通讯处理器。

检查连接和变量的组态

系统和连接参数的组态可能出现错误。无效变量值还可能由于错误引用 AS 中的变量编址而引起。

使用“通道诊断”诊断通道

“通道诊断”可以查询运行系统中通道和连接的状态。产生的任何错误都用“错误代码”显示。

诊断通道变量

在运行系统的变量管理器中，可以查询当前值、当前质量代码和变量改变的最后时刻。

参见

如何检查变量 (页 549)

如何检查通道和连接 (页 546)

如何检查连接和变量的组态 (页 545)

检查 SIMATIC NET 中的通讯处理器 (页 544)

如何检查通讯处理器的组态 (页 542)

19.7.2 日志文件条目的描述

引言

在日志文件中记录通道的错误和状态的重要变化。这些条目可用于分析通讯问题。

文件中的每个条目都有日期和时间标志，后面带有标记名称和描述。

记录册条目实例：

```
2000-05-03 14:43:48,733 INFO Log starting ...
2000-05-03 14:43:48,733 INFO | LogFileName :d:\Siemens\WinCC\Diagnose
\SIMATIC_S5_PROFIBUS_FDL_01.LOG
2000-05-03 14:43:48,733 INFO | LogFileCount :3
2000-05-03 14:43:48,733 INFO | LogFileSize :1400000
2000-05-03 14:43:48,733 INFO | TraceFlags :fa017fff
```

“INFO” 标记条目的描述

消息文本	描述
Log starting ...	起始消息
LogFileName : C:\ Siemens\ WinCC\ Diagnose\ "channel_name".LOG	带路径日志文件的名称
LogFileCount : "n"	通道的日志文件的数目
LogFileSize : "x"	单个日志文件的大小（以字节为单位）
TraceFlags :fa017fff	用十六进制数字显示跟踪功能使用的标记

19.7.3 确定错误变量值的原因

19.7.3.1 如何确定错误变量的原因

如果在运行系统中产生意外的变量值，进行如下处理以确定原因：

1. 检查通讯处理器的组态
2. 检查 SIMATIC NET 中的通讯处理器

19.7 “SIMATIC S5 Profibus FDL” 通道的诊断

3. 检查连接和变量的组态
4. 检查通道和连接
5. 检查通道变量

参见

- 如何检查连接和变量的组态 (页 545)
- 如何检查变量 (页 549)
- 如何检查通道和连接 (页 546)
- 检查 SIMATIC NET 中的通讯处理器 (页 544)
- 如何检查通讯处理器的组态 (页 542)

19.7.3.2 如何检查通讯处理器的组态

简介

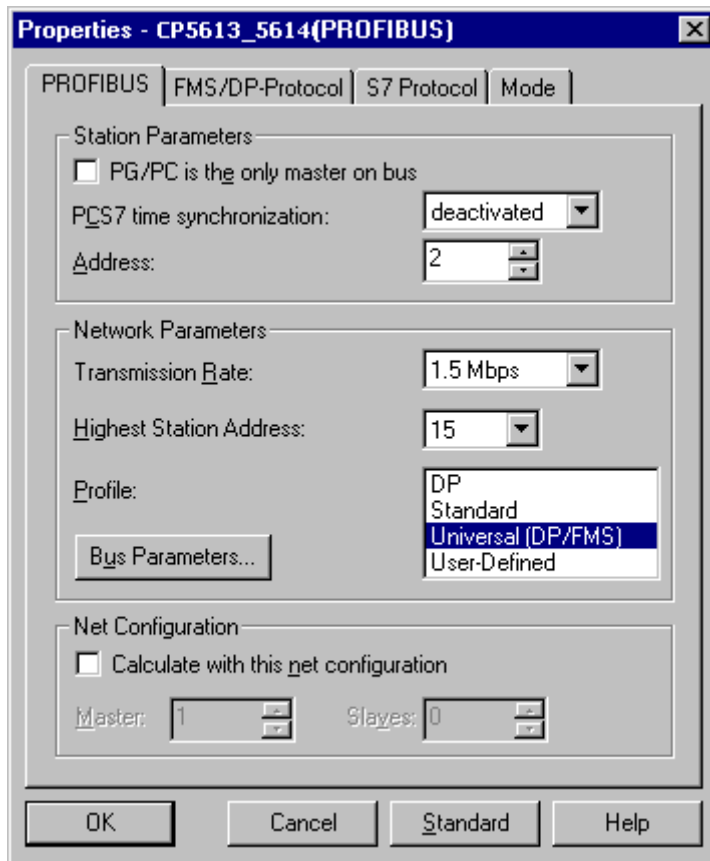
本部分说明如何使用“PG/PC 端口”程序检查通信处理器。在本例中，“CP 5613”类型的处理器用于 PROFIBUS 通信。

要求

- 安装 CP 5613。
- 安装相关的通信驱动程序。
- 组态 CP 5613。

步骤

1. 在控制面板中，单击“设置 PG/PC 端口”(Set PG/PC Port) 图标。将打开“设置 PG/PC 端口”(Set PG/PC Port) 对话框。
2. 检查访问点条目。当安装 CP 5613 时，为 PROFIBUS 连接自动添加访问点“CP_L2_1:/SCP”。选择该访问点的条目。单击“属性”(Properties) 打开“属性 - CP5613_5614 (PROFIBUS)”(Properties - CP5613_5614(PROFIBUS)) 对话框。



3. 检查“PROFIBUS”选项卡上的条目。
4. 单击“操作状态”(Operational State) 选项卡。单击“测试”(Test) 按钮在 CP 5613 上执行功能测试。测试结果将显示在下面的输出字段中。根据测试结果，可单击“重启”(Restart) 按钮执行 CP 5613 的复位和完全重启。
5. 单击“确定”(OK) 关闭所有打开的对话框。

参见

如何检查连接和变量的组态 (页 545)

检查 SIMATIC NET 中的通讯处理器 (页 544)

19.7.3.3 检查 SIMATIC NET 中的通讯处理器

简介

本节说明如何使用 SIMATIC NET 软件中的“设置 PC 站”程序检查通信处理器。

在本实例中，“CP 5613”类型用于与通道“SIMATIC S5 PROFIBUS FDL”的 PROFIBUS 通信。

要求

- 安装 CP 5613。
- 安装 SIMATIC NET 软件。
- 在 SIMATIC NET 中组态 CP 5613。

操作步骤

1. 激活 Windows 开始菜单，打开“SIMATIC / SIMATIC NET / 设置”(SIMATIC / SIMATIC NET / Settings) 并选择菜单项“设置 PC 站”(Set PC Station)。将打开“组态控制台 PC 站”(Configurations Console PC Station) 对话框。
2. 检查访问点条目。在浏览窗口中选择“访问点”(Access Point) 目录。已存在的访问点显示在数据窗口中。安装 CP 5613 时，自动为 Profibus 连接插入访问点“CP_L2_1:”。在数据窗口中选择该访问点。使用快捷菜单中的“属性”(Properties) 菜单项打开“CP_L2_1 的属性:”(Properties of CP_L2_1:) 对话框。
3. 检查“分配的接口参数”(Assigned Interface Parameters) 字段内的条目。如果是 PROFIBUS 网络中的 CP5613，选择条目“CP5613_5614 (PROFIBUS)”。
4. 打开浏览窗口，选择“组件”(Components) 目录，然后选择“CP5613/CP5614”(CP5613/CP5614) 子目录。
5. 选择“网络诊断”(Network Diagnosis) 目录。单击“测试”(Test) 按钮在 CP 5613 上执行功能测试。测试结果显示在输出窗口中。根据测试结果，单击“常规”(General) 目录中的“重新启动”(Restart) 按钮以执行 CP 5613 的复位，然后完全重新启动。
6. 检查连接到“总线成员”(Bus Participants) 目录列表中 PROFIBUS 的成员列表。根据显示，可确定您自己工作站的功能和状态以及其它连接成员的功能和状态。
7. 关闭对话框。
8. 如果在通信处理器组态中检测到故障，只能使用 SIMATIC NET 工具修改组态。详细信息请参见 SIMATIC NET。

参见

如何检查连接和变量的组态 (页 545)

19.7.3.4 如何检查连接和变量的组态

简介

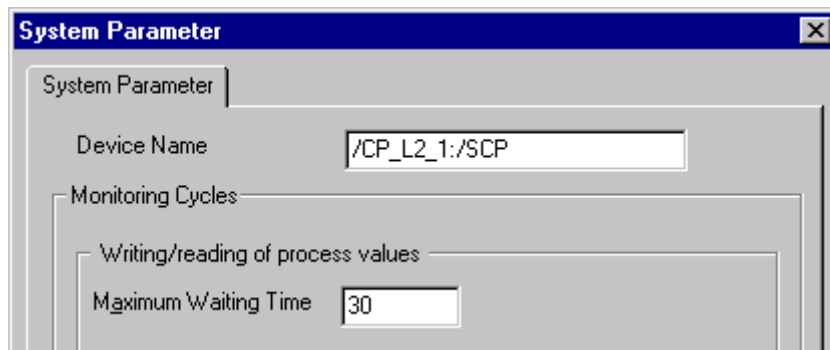
本节说明如何检查系统参数以及连接和变量的组态。

要求

- 安装 CP 5613。
- 安装通信驱动程序。
- 组态 CP 5613。
- 为“SIMATIC S5 PROFIBUS FDL” 通道组态连接和变量。
- 激活 WinCC 项目。

步骤

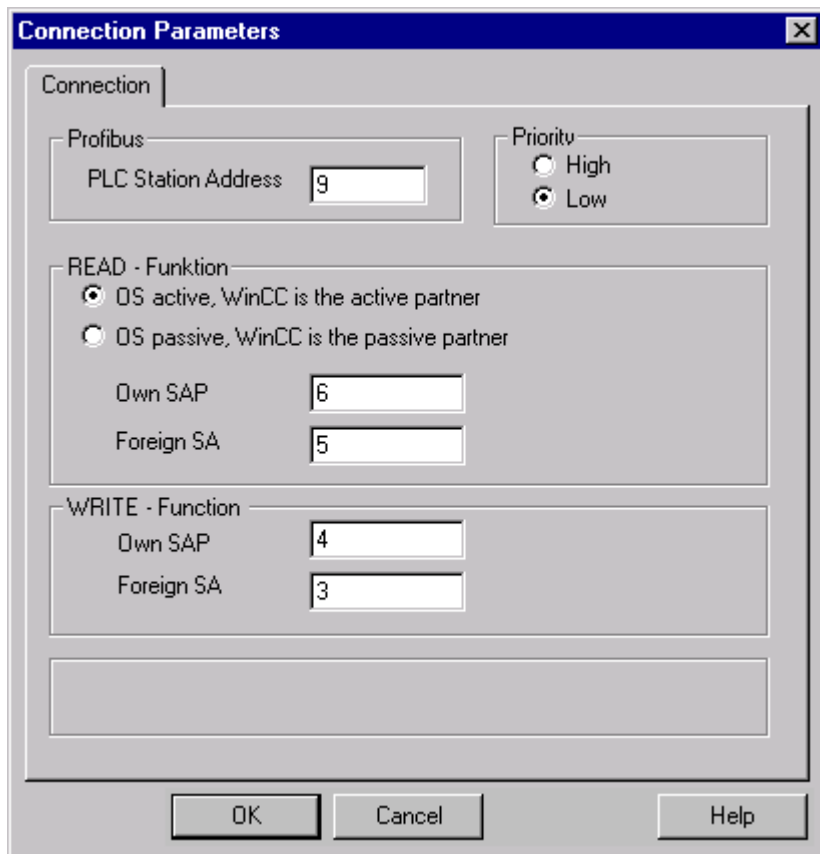
1. 在 WinCC 项目管理器浏览窗口中，单击“SIMATIC S5 PROFIBUS FDL” 图标前的加号。在“FDL (CP5412/A2-1)” 图标的关联菜单中，单击“系统参数”。将打开“System Parameters”（系统参数）对话框。



2. 检查“Device Name”（设备名称）域中的条目。缺省状态下，设置的访问点是“CP_L2_1:/SCP”。在 CP 5613 系统中安装通信处理器期间分配访问点。关闭对话框。
3. 单击“FDL (CP5412/A2-1)” 图标前的加号。在要测试变量的关联菜单中，选择条目“属性”。将打开“连接属性”对话框。

19.7 “SIMATIC S5 Profibus FDL” 通道的诊断

- 在“连接属性”对话框中单击“属性”按钮。将打开“连接参数”对话框。



- 检查“连接”标签上的设置。关闭打开的对话框。
- 单击连接图标前的加号。在要测试变量的关联菜单中，单击条目“属性”。将打开“变量属性”对话框。检查“类型转换”和“数据类型”域中的条目。
- 在“变量属性”对话框中，单击“选择”按钮。将打开“地址属性”对话框。检查这些设置。
- 单击“确定”(OK) 关闭所有打开的对话框。

参见

如何检查通道和连接 (页 546)

19.7.3.5 如何检查通道和连接





简介

本节说明如何在运行系统中检查“SIMATIC S5 PROFIBUS FDL”通道及其连接。

要求

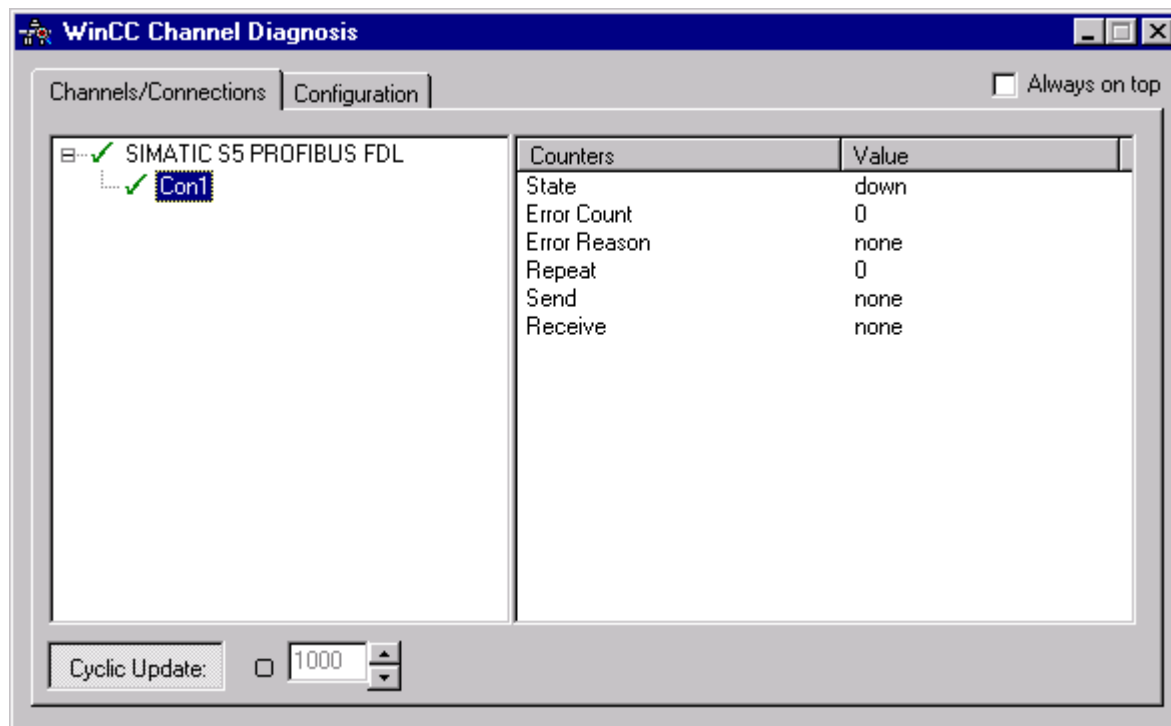
- 安装 CP 5613。
- 安装通信驱动程序。
- 组态 CP 5613。
- 创建 STEP5 项目。
- 为“SIMATIC S5 PROFIBUS FDL” 通道组态连接和变量。
- 激活 WinCC 项目。

状态消息概述

图标	描述
	通道/连接无限制准备就绪
	通道/连接准备就绪，带有一些限制
	通道/没有关于连接状态的陈述
	通道/连接失败

步骤

1. 从“开始”(Start) 菜单启动 WinCC 通道诊断。
2. 将打开“通道诊断”(Channel Diagnosis) 应用程序窗口。所有安装的通道及其连接的状态信息显示在“通道/连接”(Channels/Connections) 选项卡的左侧。



3. 检查名为“SIMATIC S5 PROFIBUS FDL” 通道前的图标和连接。如果通道和连接状况正常，将有一个绿色的复选标记显示在相应条目的前面。有关各个图标含义的信息，请参见表格“状态消息概述”。
4. 如果通道名称和连接前面没有绿色复选标记，则选择左侧窗口中的连接。在右侧窗口中，检查计数器“状态”、“错误计数”、“错误原因”、“发送”和“接收”的条目。这些值表示检测到的错误。
5. 检查通道特定的日志文件。要执行此操作，请使用文本编辑器打开目录“Siemens\WinCC \Diagnose” 中的文件。检查带“ERROR” 标记的最新条目。有关该主题的更多信息，请参见“日志文件条目的描述”。
6. 如果检查日志文件之后仍然不能查明错误，请激活跟踪功能并联系客户支持部门。有关该主题的更多信息，请参见“组态通道的跟踪功能”。

参见

如何组态通道的跟踪功能 (页 514)

日志文件条目的描述 (页 541)

如何检查变量 (页 549)

19.7.3.6 如何检查变量

简介

如果运行系统中外部变量不具有预期的数值，请使用下列步骤来检查变量。

要求

- 安装 CP 5613。
- 安装通信驱动程序。
- 组态 CP 5613。
- 创建 STEP5 项目。
- 为“SIMATIC S5 PROFIBUS FDL”通道组态连接和变量。
- 激活 WinCC 项目。

步骤

1. 在 WinCC 项目管理器的变量管理器中，选择“SIMATIC S5 PROFIBUS FDL”通道。
2. 在数据窗口中，选择希望检查的外部变量。为此，打开目录结构直到此变量显示在表格区域中。
3. 移动鼠标指针，指向要检查的变量。将打开工具提示窗口，显示当前变量值、质量代码和最近一次数值改变的时间。
4. 检查质量代码。如果显示数值“80”，则该变量值正确。其它数值的描述可以参见“变量质量代码”。
5. 如果质量代码不等于“80”，从变量管理器中选择变量，然后单击快捷菜单中的“属性”(Properties) 来打开“变量属性”(Tag Properties) 对话框。
6. 检查是否已在“限制/报告”(Limits/Reportin) 选项卡上组态了上限或下限、启动或替换值。这些数值会影响显示。
7. 如果变量值被其中一个已组态的数值改变，则取消激活该项目并改变该限制值或替换值。

说明

变量值、质量代码等只在运行期间显示。

参见

变量的质量代码 (页 562)

19.8 “OPC” 通道的诊断

19.8.1 诊断“OPC” 通道的可能性

下列可能性用于检测错误和诊断“OPC” 通道或其变量之一：

检查连接和变量的组态

系统和连接参数的组态可能出现错误。无效变量值还可能由于错误引用 AS 中的变量编址而引起。

使用“通道诊断” 诊断通道

“通道诊断” 可以查询运行系统中通道和连接的状态。产生的任何错误都用“错误代码” 显示。

诊断通道变量

在运行系统的变量管理器中，可以查询当前值、当前质量代码和变量改变的最后时刻。

参见

如何检查变量 (页 558)

如何检查通道和连接 (页 556)

如何检查组态数据 (页 555)

19.8.2 日志文件条目的描述

19.8.2.1 日志文件条目的描述

引言

在日志文件中记录通道的错误和状态的重要变化。下列章节只包含最重要的条目。这些条目可用于分析通讯问题。

下列两种类型的条目之间存在差别：

- INFO
- ERROR

条目的结构

日期/时间标记	标志名	描述
---------	-----	----

记录册中条目实例

```

2000-03-24 10:43:18,756 INFO Log starting ...
2000-03-24 10:43:18,756 INFO | LogFileName :C:\Siemens\WinCC\Diagnose\OPC.LOG
2000-03-24 10:43:18,756 INFO | LogFileCount :3
2000-03-24 10:43:18,756 INFO | LogFileSize :1400000
2000-03-24 10:43:18,756 INFO | TraceFlags :fa000007
000-03-24 10:43:18,756 INFO Process attached at 2000-03-24 09:43:18,746 UTC
2000-03-23 10:46:18,756 INFO Process detached at 2000-03-2410:46:18,746UTC
2000-03-27 13:22:43,390 ERROR ..FOPCData::InitOPC CoCreateInstanceEx- ERROR
800706ba
2000-03-27 13:22:43,390 ERROR - ChannelUnit::SysMessage("[OPC Groups (OPCHN
Unit #1)]![OPC_No_Machine]: CoCreateInstance for server "OPCServer.WinCC" on
machine OPC_No_Machine failed, Error=800706ba (HRESULT = 800706ba -
RPC_S_SERVER_UNAVAILABLE (Der RPC-Server ist nicht verfügbar.))")
    
```

参见

“ERROR” 标记条目 (页 553)

“INFO” 标记条目 (页 552)

19.8 “OPC” 通道的诊断

19.8.2.2 “INFO” 标记条目

引言

文件中的每个条目都有日期和时间标志，后面带有标记名称和描述。

日期/时间标志	标志名	描述
---------	-----	----

记录册中条目实例

```

2000-03-24 10:43:18,756 INFO Log starting ...
2000-03-24 10:43:18,756 INFO | LogFileName :C:\Siemens\WinCC\Diagnose\OPC.LOG
2000-03-24 10:43:18,756 INFO | LogFileCount :3
2000-03-24 10:43:18,756 INFO | LogFileSize :1400000
2000-03-24 10:43:18,756 INFO | TraceFlags :fa000007
000-03-24 10:43:18,756 INFO Process attached at 2000-03-24 09:43:18,746 UTC
2000-03-23 10:46:18,756 INFO Process detached at 2000-03-2410:46:18,746UTC
    
```

最重要记录册条目的描述

消息文本	描述
Log starting ...	起始消息
LogFileName : C:\ Siemens\ WinCC\ Diagnose\ "channel_name".LOG	带路径日志文件的名称
LogFileCount : "n"	通道的日志文件的数目
LogFileSize : "x"	单个日志文件的大小（以字节为单位）
TraceFlags :fa000007	用十六进制数字显示跟踪功能使用的标记
Process attached at 2000-03-24 09:43:18,746 UTC	通道由 WinCC 数据管理器装载。

消息文本	描述
Process detached at 2000-03-24 10:46:18,746 UTC	通道由 WinCC 数据管理器卸载。
IOPCChnShutdown::ShutdownRequest was called... Reason:system going down" IOPCChnShutdown::ShutdownRequest	取消激活 WinCC OPC 服务器的 WinCC 项目。 请求 WinCC OPC 客户机与 WinCC OPC 服务器断开。

19.8.2.3 “ERROR” 标记条目

简介

文件中的每个条目都有日期和时间标志，后面带有标记名称和描述。如果是“Error”标记，描述包含消息文本、错误代码和错误消息的文本。一些错误代码没有出错消息的文本。

日期/时间标记	标志名	描述 消息文本 + 错误代码 + 出错消息文本
---------	-----	----------------------------

记录册中条目实例

2000-03-27 13:22:43,390 ERROR ..FOPCData::InitOPC CoCreateInstanceEx- ERROR 800706ba

2000-03-27 13:22:43,390 ERROR - ChannelUnit::SysMessage("[OPC Groups (OPCHN Unit #1)][OPC_No_Machine]: CoCreateInstance for server \"OPCServer.WinCC\" on machine OPC_No_Machine failed, Error=800706ba (HRESULT = 800706ba - RPC_S_SERVER_UNAVAILABLE (RPC server not available.))")

最重要注册条目的描述

错误代码	错误消息文本	可能的原因
c0040004	服务器不支持“canonicalDatatype”和“requestedDatatype”之间的转换。	访问 OPC 服务器上的 WinCC 变量失败。可以转换但是失败了。 WinCC 变量不在服务器上，或者组态的数据类型不匹配。
c0040007	该名称在服务器的名称空间中不存在。	如果使用服务器的名称空间中不存在的变量名称访问 OPC 客户端，服务器将始终返回错误代码。 示例：浏览、读取变量、写入变量、将变量插入订阅。
00000001	添加条目	访问 OPC 服务器上的 WinCC 变量失败。 WinCC 变量不在服务器上，或者组态的数据类型不匹配。 数据类型 WinCC 变量 OPC 服务器 = 数据类型 WinCC 变量 OPC 客户机。
80004005	无法解析服务器名称	用作 WinCC OPC 服务器的计算机在网络上不可用。 通过“OPC”通道访问的 WinCC OPC 服务器不可用。
80040154	未注册等级	WinCC OPC 服务器未在系统中正确注册。 WinCC OPC 服务器的 WinCC 项目未激活。
80070057	参数错误	WinCC 变量不在 OPC 服务器上，或者组态的数据类型不匹配。
800706ba	RPC 服务器不可用。	在网络上找不到要在其上启动 OPC 服务器的计算机。

19.8.3 确定错误变量值的原因

19.8.3.1 如何确定无效变量的原因

如果在运行系统中产生意外的变量值，进行如下处理以确定原因：

1. 检查组态数据
2. 检查连接
3. 检查通道变量

参见

如何检查变量 (页 558)

如何检查通道和连接 (页 556)

如何检查组态数据 (页 555)

19.8.3.2 如何检查组态数据

要求

- 一台计算机用作运行 WinCC 项目的 WinCC OPC 客户机。
- “OPC” 通道必须集成在 OPC 客户机的 WinCC 项目中。
- 在 OPC 服务器的 WinCC 项目中组态 WinCC 变量。
- 在 OPC 客户机上组态一个连接和一个 WinCC 变量，与创建的服务器变量通讯。
- 激活 OPC 服务器和客户机上的 WinCC 项目。

步骤

1. 在 OPC 客户机的 WinCC 项目管理器浏览窗口中，单击“OPC”图标前的加号。单击图标“OPC 组 (OPCHN Unit#1)”前的“加号”。
2. 在要测试变量的快捷菜单中，选择条目“属性”。将打开“连接属性”对话框。单击“OPC 组”标签。
3. 在域“OPC 服务器名称”中，检查 OPC 服务器的 ProgID。
如果连接到装有 WinCC V 5.0 或更高版本的服务器，必须输入“OPCServer.WinCC”。
如果服务器运行 WinCC V4.x，必须输入“OE.Groups”。
4. 在域“在本计算机上启动服务器”中，输入要用作 OPC 服务器的计算机名称。单击“测试服务器”按钮来测试与 OPC 服务器的连接。关闭对话框。
5. 单击连接图标前的加号。在要测试变量的关联菜单中，单击条目“属性”。将打开“变量属性”对话框。

19.8 “OPC” 通道的诊断

6. 必须为该变量输入与 OPC 服务器上变量相同的“数据类型”。
7. 在“变量属性”对话框中，单击“选择”按钮。将打开“地址属性”对话框。
8. 检查“条目名称”和“数据类型”域中的条目。“条目名称”必须与 OPC 服务器上的变量名相匹配。“数据类型”必须与 OPC 服务器上变量的数据类型相匹配。
9. 检查通道专用的日志文件。为此，使用文本编辑器来打开目录“Siemens\WinCC\Diagnose”中的文件。检查带“ERROR”标记的最新条目。有关该主题的更多信息，请参阅“日志文件条目的描述”。
10. 如果检查日志文件之后仍然不能查明错误，请激活跟踪功能并联系客户支持部门。有关该主题的更多信息，请参阅“组态通道的跟踪功能”。

参见

- 如何组态通道的跟踪功能 (页 514)
- 日志文件条目的描述 (页 550)
- 如何检查通道和连接 (页 556)

19.8.3.3 如何检查通道和连接



简介

本节说明如何在运行系统中检查“OPC”通道及其连接。

要求

- 一台计算机用作运行 WinCC 项目的 WinCC OPC 客户机。
- “OPC”通道必须集成在 OPC 客户机的 WinCC 项目中。
- 在 OPC 服务器的 WinCC 项目中组态 WinCC 变量。
- 在 OPC 客户机上组态一个连接和一个 WinCC 变量，与创建的服务器变量通信。
- 激活 OPC 服务器和客户机上的 WinCC 项目。

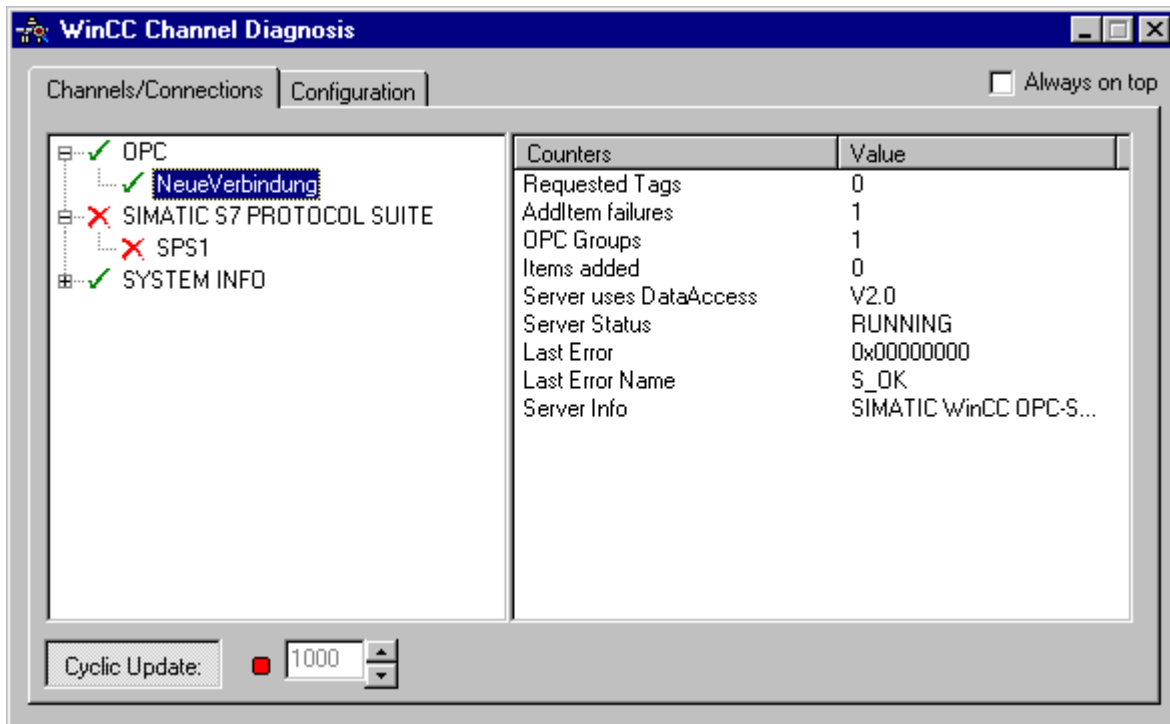
状态消息概述

图标	描述
	通道/连接无限制准备就绪
	通道/连接准备就绪，带有一些限制

图标	描述
	通道/没有关于连接状态的陈述
	通道/连接失败

步骤

1. 从“开始”(Start) 菜单启动 WinCC 通道诊断。
2. 将打开“通道诊断”(Channel Diagnosis) 应用程序窗口。所有安装的通道及其连接的状态信息显示在“通道/连接”(Channels/Connections) 选项卡的左侧。



3. 检查 OPC 连接前的图标。如果连接状态良好，将有一个绿色的复选标记显示在相应条目的前面。有关各个图标含义的信息，请参见表格“状态消息概述”。
4. 如果通道名称和连接前面没有绿色复选标记，则选择左侧窗口中的连接。在右侧窗口中，检查计数器“添加项目失败”、“服务器状态”、“最后错误”和“最后错误名称”的条目。这些值表示检测到的错误。
5. 检查通道特定的日志文件。要执行此操作，请使用文本编辑器打开目录“Siemens\WinCC \Diagnose”中的文件。检查带“ERROR”标记的最新条目。有关该主题的更多信息，请参见“日志文件条目的描述”。
6. 如果检查日志文件之后仍然不能查明错误，请激活跟踪功能并联系客户支持部门。有关该主题的更多信息，请参见“组态通道的跟踪功能”。

参见

如何组态通道的跟踪功能 (页 514)

日志文件条目的描述 (页 550)

如何检查变量 (页 558)

19.8.3.4 如何检查变量

简介

如果运行系统中外部变量不具有预期的数值，请使用下列步骤来检查变量。

要求

- 一台计算机用作运行 WinCC 项目的 WinCC OPC 客户机。
- “OPC” 通道必须集成在 OPC 客户机的 WinCC 项目中。
- 在 OPC 服务器的 WinCC 项目中组态 WinCC 变量。
- 在 OPC 客户机上组态一个连接和一个 WinCC 变量，与创建的服务器变量通信。
- 激活 OPC 服务器和客户机上的 WinCC 项目。

步骤

1. 在 WinCC 项目管理器的变量管理器中，选择“OPC” 通道。
2. 在数据窗口中，选择希望检查的外部变量。为此，打开目录结构直到此变量显示在表格区域中。
3. 移动鼠标指针，指向要检查的变量。将打开工具提示窗口，显示当前变量值、质量值和最近一次修改的时间。
4. 检查质量值。如果显示值“C0”，则变量值正确。其它数值的描述可以参见“变量质量代码”。
5. 如果质量代码不等于“C0”，从变量管理器中选择变量，然后单击快捷菜单中的“属性”来打开“变量属性”对话框。
6. 检查是否已在“限制/报告”(Limits/Reportin) 选项卡上组态了上限或下限、启动或替换值。这些数值会影响显示。
7. 如果变量值被其中一个已组态的数值改变，则取消激活该项目并改变该限制值或替换值。

说明

变量值、质量代码等只在运行期间显示。

参见

变量的质量代码 (页 562)

19.9 变量质量

19.9.1 变量质量

引言

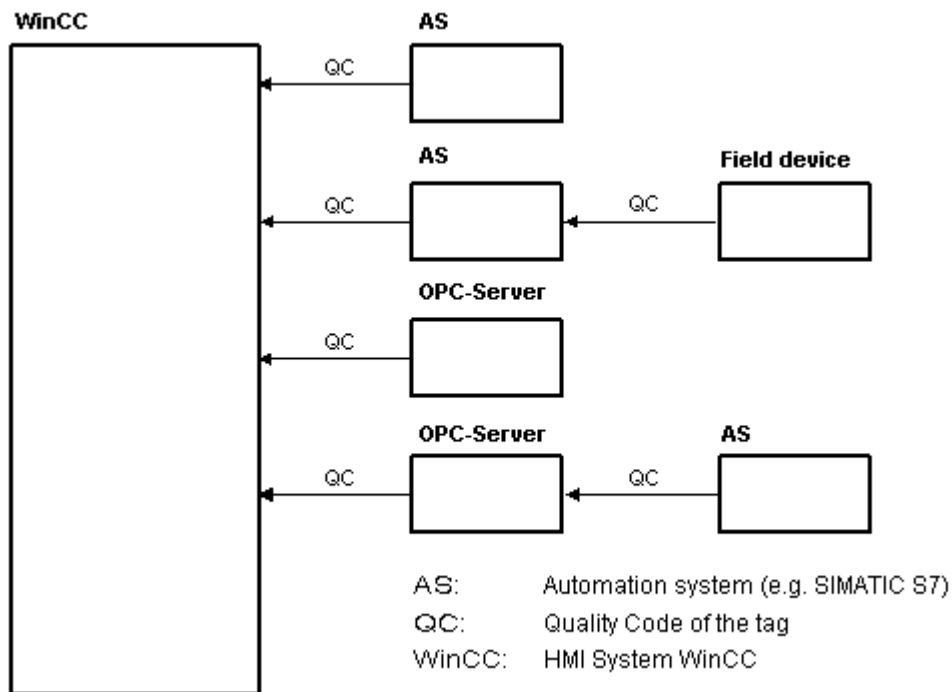
在 WinCC 中，有两个质量指标用来评估变量质量。这两个指标为变量状态和质量代码。

变量状态在 WinCC 中形成，用于指示 OS 中组态设置的质量。另外，变量状态还指示与 WinCC 通讯伙伴的连接状态。通讯伙伴可以是自动化系统或服务器计算机。

质量代码包含与变量状态相同的信息。除此以外，质量状态还包含有关连接伙伴的质量语句，用于评估或处理变量。可能的伙伴为：

- 自动化系统
- 带现场设备的自动化系统
- OPC 服务器
- 带下位自动化系统的 OPC 服务器

其中质量代码在处理链内转发。如果在处理链的某处存在一个变量的多个质量代码有待处理，那么将转发最差的代码。



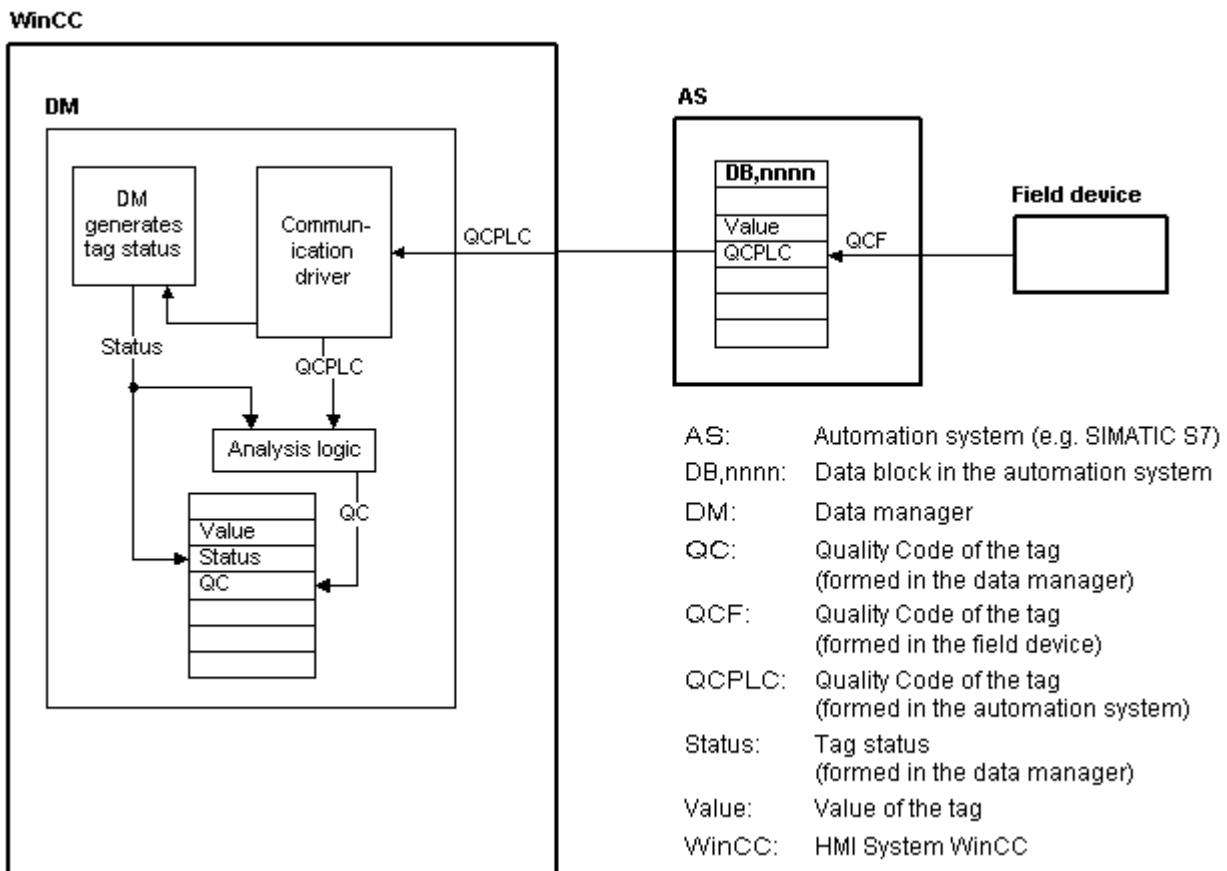
质量代码提示变量质量，与在何处形成该代码无关。

层叠的质量代码

通过使用连接了现场设备的自动化系统实例，可以概述质量代码层叠。

自动化系统读取由现场设备生成的质量代码。通过一个分析逻辑，根据优先级对等待同一个变量同时处理的质量代码进行评估。并将表示最差状态的质量代码分配给变量。必须在数据块中，将该质量代码紧跟在相关的变量值之后保存。

可通过 PCS7 库的通道模块来启动分析逻辑。如果不能使用 PCS7 库，那么必须在自动化系统中手动组态分析逻辑。



通过其中一个通讯驱动程序，WinCC 在运行系统中从自动化设备读取变量，包括相关的质量代码在内。对于每个变量，将在数据管理器中生成变量状态。可以包括如超出组态的测量范围限制以及 WinCC 和自动化设备之间的链接状态等。

通过数据管理器中的分析逻辑，从数据管理器的变量状态以及自动化设备的质量代码中生成质量代码。此处也同样传送表示最差状态的代码，并由 WinCC 将其保存为质量代码。对于在自动化系统中没有质量代码的变量，质量代码始终与变量状态相同。

19.9.2 变量的质量代码

简介

检查变量的状态和质量时需要质量代码。所显示的质量代码概述了整个数值传送和各个变量的数值处理的质量。于是例如使用质量代码可以查看当前值是起始值还是替换值。

质量代码具有优先次序。如果同时产生多个代码，那么将显示具有最差状态的代码。

质量代码的计算

可通过各种不同方法计算质量代码：

- 使用 VB 脚本计算
- 使用 C 脚本计算
- 通过动态对话框计算
- I/O 字段的“质量代码更改变量”结果的计算

说明

为了将整个数值传送和处理包含在过程变量的质量代码中，已连接的自动化系统必须支持质量代码。在 AS 中组态变量时，确保有足够的存储空间用于质量代码。例如，在 S7 系列的 AS 中，质量代码需要一个额外的字节附加到过程值上。为了防止错误，在组态变量时必须将该字节考虑进去，例如在数据块的末尾。

在变量管理中显示质量代码

可以在“变量管理”(Tag Management) 中查看变量的质量代码。

要求：

- WinCC 项目已激活。
- “变量管理”(Tag Management) 数据区域中显示了“质量代码”(Quality Code) 列。



在过程画面中显示质量代码

对于具有过程连接的图形对象中的变量值显示，质量代码可能会影响该显示。如果质量代码的值为 0x80（优）或 0x4C（初始值），变量值不会显示为灰色。所有其它值都会显示为灰色。此外，根据所设置的 WinCC 设计，对于以下对象将显示一个黄色警告三角标志：

- I/O 字段
- 棒图，3D 棒图
- 复选框，单选框
- 组显示，状态显示
- 滚动条对象

结构

质量代码具有如下的二进制结构：

QQSSSSL

Q: 质量

S: 质量的子状态

19.9 变量质量

L: 限制。该数值是可选的。

说明

显示在“质量”表格中的质量代码是质量级的基本数值。使用子状态和限制元素导致中间数值超过相关的质量级。

质量

前两位指定变量的质量。

	Q	Q	S	S	S	S	L	L
	2	2	2	2	2	2	2	2
	7	6	5	4	3	2	1	0
劣 - 数值没有用。	0	0	-	-	-	-	-	-
不确定 - 数值的质量低于普通情况，但是数值可能仍然有用。	0	1	-	-	-	-	-	-
优（非层叠） - 数值的质量是优的。可能的报警条件会通过子状态指示。	1	0	-	-	-	-	-	-
优（层叠） - 数值可能用在控件中。	1	1	-	-	-	-	-	-

子状态

仅仅使用质量是不够的。单个的质量分为子状态。质量代码是二进制编码的。为了分析质量代码，这些数值必须转换为十六进制表达式。

变量的质量代码

在下表中列出了可能的质量代码。列表顶部列出了最差的质量代码，而列表底部列出了最好的质量代码。为最好的质量代码分配最低优先级，而为最差的质量代码分配最高优先级。如果此过程中的一个变量发生多种状态，则将传送最差的质量代码。

代码 (十六进制)	质量		Q	Q	S	S	S	S	L	L
0x23	劣	设备钝化 - 禁止诊断报警	0	0	1	0	0	0	1	1
0x3F	劣	功能检查 - 本地优先	0	0	1	1	1	1	1	1

代码 (十六进制)	质量		Q	Q	S	S	S	S	S	L	L
0x1C	劣	不能用 - 该数值不可靠，因为该块还未判断，并且可能由组态设计者构造。如果该块模式是 O/S，则进行置位。	0	0	0	1	1	1	-	-	-
0x73	不确定	模拟值 - 开始	0	1	1	1	0	0	1	1	-
0x74	不确定	模拟值 - 结束	0	1	1	1	0	1	-	-	-
0x84	优 (非层叠)	激活更新事件 - 如果数值是优的并且块具有激活的更新事件，则进行置位。	1	0	0	0	0	1	-	-	-
0x24	劣	维护报警 - 提供更多诊断。	0	0	1	0	0	1	-	-	-
0x18	劣	无通信，没有可用的数值 - 如果自该数值上次不能使用以来从未进行过通信，则进行置位。	0	0	0	1	1	0	-	-	-
0x14	劣	无通信，具有最后有用的数值 - 如果该数值已经由现在发生故障的通信置位，则进行置位。	0	0	0	1	0	1	-	-	-
0x0C	劣	设备故障 - 如果数值源受设备故障的影响，则进行置位。	0	0	0	0	1	1	-	-	-
0x10	劣	传感器故障	0	0	0	1	0	0	-	-	-
0x08	劣	未连接 - 如果该输入需要连接而没有连接，则进行置位。	0	0	0	0	1	0	-	-	-
0x04	劣	组态错误 - 根据特定制造商所检测到的，如果该数值是因为存在有关参数化或组态的一些不一致而无用，则进行置位。	0	0	0	0	0	1	-	-	-
0x00	劣	未指定 - 没有指出数值劣的原因。用于传播。	0	0	0	0	0	0	-	-	-
0x28	劣	与过程有关 - 替换值	0	0	1	0	1	0	-	-	-
0x2B	劣	与过程有关 - 无维护	0	0	1	0	1	0	1	1	-
0x68	不确定	需要维护	0	1	1	0	1	0	-	-	-
0x60	不确定	模拟数值 - 当块处于手动模式时如果由操作员写入过程值，则进行置位。	0	1	1	0	0	0	-	-	-
0x64	不确定	传感器校准	0	1	1	0	0	1	-	-	-
0x5C	不确定	组态错误	0	1	0	1	1	1	-	-	-
0x58	不确定	低于正常	0	1	0	1	1	0	-	-	-
0x54	不确定	工程单位范围超出 - 如果数值位于该参数定义的数值设置范围之外，则进行置位。限制定义已经超出了哪一个方向。	0	1	0	1	0	1	-	-	-

19.9 变量质量

代码 (十六进制)	质量		Q Q S S S S L L							
			Q	Q	S	S	S	S	L	L
0x50	不确定	传感器转换不精确	0	1	0	1	0	0	-	-
0x4B	不确定	替换 (常数)	0	1	0	0	1	0	1	1
0x78	不确定	与过程有关 - 无维护	0	1	1	1	1	0	-	-
0x4C	不确定	初始值 - 复位设备或参数期间和之后的易失参数的数值。	0	1	0	0	1	1	-	-
0x48	不确定	替换值 - 预定义数值用于替换已计算的数值。这用于故障安全处理。	0	1	0	0	1	0	-	-
0x44	不确定	最后有用的数值 - 无论写入什么, 该数值都已停止操作。这用于故障安全处理。	0	1	0	0	0	1	-	-
0x40	不确定	未指定 - 没有指出数值不确定的原因。	0	1	0	0	0	0	-	-
0xE0	优 (层叠)	启动故障安全 (IFS) - 来自需要其后续程序输出块 (例如 AO) 进入故障安全的程序块的值。	1	1	1	0	0	0	-	-
0xD8	优 (层叠)	本地替换值 (LO) - 数值来自本地键切换锁定的块或来自带有互锁逻辑激活的综合 AO/DO。正常控制故障必须传播到主机系统报警和显示目的的功能运行。这同样意味着不引入。	1	1	0	1	1	0	-	-
0xD4	优 (层叠)	不选择 (DNS) - 数值来自不应选择的块, 由于条件位于块内或超过块。	1	1	0	1	0	1	-	-
0xCC	优 (层叠)	不引入 (NI) - 数值来自不具有使用该输入的目标模式的块。	1	1	0	0	1	1	-	-
0xC8	优 (层叠)	初始化请求 (IR) - 数值是源 (返回计算输入参数) 的初始化数值, 因为低级循环中断或者模式错误。	1	1	0	0	1	0	-	-
0xC4	优 (层叠)	初始化确认 (IA) - 数值是来自源 (层叠输入、参数的远程层叠和参数的远程输出) 的初始化数值。	1	1	0	0	0	1	-	-
0xC0	优 (层叠)	好 - 没有错误或特殊情况与该数值相关。	1	1	0	0	0	0	-	-
0xA0	优 (非层叠)	启动故障安全	1	0	1	0	0	0	-	-
0x98	优 (非层叠)	不确定的紧急报警 - 如果数值是优的并且块具有优先级大于或等于 8 的不确定报警, 则进行置位。	1	0	0	1	1	0	-	-
0x94	优 (非层叠)	不确定的报告报警 - 如果数值是优的并且块具有优先级小于 8 的不确定报警, 则进行置位。	1	0	0	1	0	1	-	-

代码 (十六进制)	质量		Q	Q	S	S	S	S	L	L
0x90	优 (非层叠)	不确定的更新事件 - 如果数值是优的并且块具有不确定的更新事件, 则进行置位。	1	0	0	1	0	0	-	-
0x8C	优 (非层叠)	激活紧急报警 - 如果数值是优的并且块具有优先级大于或等于 8 的激活报警, 则进行置位。	1	0	0	0	1	1	-	-
0x88	优 (非层叠)	激活报告报警 - 如果数值是优的并且块具有优先级小于 8 的激活报警, 则进行置位。	1	0	0	0	1	0	-	-
0xA8	优 (非层叠)	需要维护	1	0	1	0	1	0	-	-
0xA4	优 (非层叠)	要求维护	1	0	1	0	0	1	-	-
0xBC	优 (非层叠)	功能检查 - 本地优先	1	0	1	1	1	1	-	-
0x80	优 (非层叠)	好 - 没有错误或特殊情况与该数值相关。	1	0	0	0	0	0	-	-

限制

质量代码可以由限制进一步细分。限制是可选的。

	Q	Q	S	S	S	S	L	L
好 - 数值可以随意移动。	-	-	-	-	-	-	0	0
下限 - 数值已经接受其下限。	-	-	-	-	-	-	0	1
上限 - 数值已经接受其上限。	-	-	-	-	-	-	1	0
常数 (上限和下限) - 数值不能变动, 不管过程如何操作。	-	-	-	-	-	-	1	1

利用 OPC 通信的质量代码

在通过“OPC”通道进行的通信中, 将转换 OPC 不支持的质量代码。

WinCC 中的质量代码	OPC 中的质量代码
0x48	0x40
0x4C	0x40
0x5C	0x40

WinCC 中的质量代码	OPC 中的质量代码
0x60	0x40
0x80...0xD4	0xC0
0xD8	0xC0

19.9.3 变量状态

简介

在运行系统中，可以监视各个 WinCC 变量的状态。变量状态包含已组态的测量范围过限信息以及 WinCC 和自动化设备之间的链接状态。

质量代码提示变量质量，与在何处形成该代码无关。因此，将整个数值传送和数值处理的状态考虑在内。

例如，如果低于测量范围的下限，那么质量代码通信为“0x55”。在 WinCC 数据管理器或现场设备中可能发生该类超出测量范围的情况。变量状态使您能查找到是否在 WinCC 中出现了超出测量范围或在将数值传送到 WinCC 之前就已经出现了超出测量范围的情况。

例如，变量状态以代码 0x0010 报告超出限制，它表明在 WinCC 中组态了低于下限的值。如果变量状态没有报告过限信息，那么传送到 WinCC 中的质量代码已经包含过限信息。

质量代码的计算

可通过各种不同方法计算质量代码：

- 使用 C 脚本计算
- 通过动态对话框计算
- I/O 字段的“质量代码更改变量”结果的计算

WinCC 状态标记

下表给出了可能的变量状态。

标记名称	数值	描述
	0x00 00	无错
DM_VARSTATE_NOT_ESTABLISHED	0x00 01	未建立到伙伴的连接
DM_VARSTATE_HANDSHAKE_ERROR	0x00 02	信号交换错误
DM_VARSTATE_HARDWARE_ERROR	0x00 04	网络模板故障
DM_VARSTATE_MAX_LIMIT	0x00 08	超过所组态的上限
DM_VARSTATE_MIN_LIMIT	0x00 10	超过所组态的下限
DM_VARSTATE_MAX_RANGE	0x00 20	超过格式处理上限
DM_VARSTATE_MIN_RANGE	0x00 40	超出格式处理下限
DM_VARSTATE_CONVERSION_ERROR	0x00 80	显示转换出错（与超过格式限制 xxx 有关）
DM_VARSTATE_STARTUP_VALUE	0x01 00	变量初始化值
DM_VARSTATE_DEFAULT_VALUE	0x02 00	变量的替换值
DM_VARSTATE_ADDRESS_ERROR	0x04 00	通道寻址出错
DM_VARSTATE_INVALID_KEY	0x08 00	没有找到变量/不可用
DM_VARSTATE_ACCESS_FAILURE	0x10 00	不允许访问变量

19.9 变量质量

标记名称	数值	描述
DM_VARSTATE_TIMEOUT	0x2000	超时/没有来自通道的回查消息
DM_VARSTATE_SERVERDOWN	0x4000	服务器不可用。

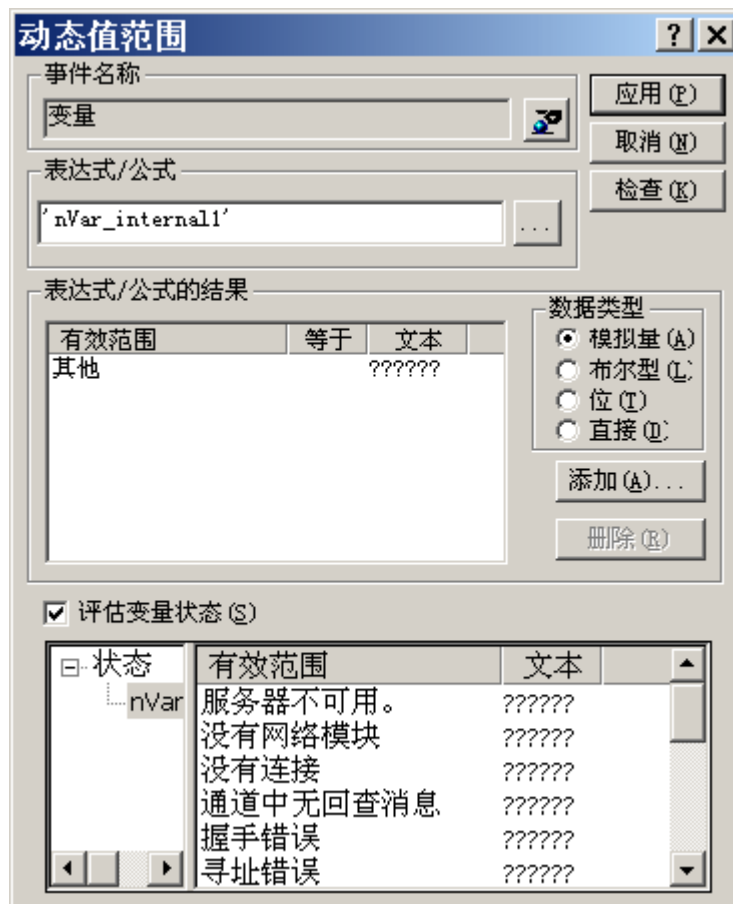
19.9.4 使用变量状态监视连接状态

引言

在运行系统中可以监视各个 WinCC 变量的状态，从而提供关于相应连接状态的信息。

该监视作为对象属性在图形编辑器中组态。监视的一种可能性是使用静态文本的“文本”属性。

在期望属性“动态”对话框的快捷菜单中，选择“动态对话框”以打开“动态值范围”对话框。



可以在此指定下列设置：

- 要监视的变量
- 对于变量值： 分配有效范围和状态显示
- 变量状态评估的激活
- 对于状态： 分配有效范围和相应的状态文本

在运行系统中，输入的状态文本之一（对应于变量的当前状态的）将显示在组态的对象中。

19.9.5 使用全局动作监视变量状态

引言

监视变量状态的途径之一是使用全局脚本编辑器中的内部函数“GetTagState”和“GetTagStateWait”。与函数“GetTag”和“GetTagWait”不同，除了返回变量的数值外，它们还返回变量的状态。可以核定该状态值，并且由它触发各种事件。它也可以用于判断相关连接的状态。

在全局动作中，要监视变量的状态值通过用于该数据类型的“GetTagState”函数来确定。对于每种变量类型都存在这样一个函数。状态值“0”指示连接完好，没有错误。这种状态现在可以按期望判断。

实例：

该实例说明监视“有符号 16 位数”类型的 WinCC 变量。为了确定该变量的状态，使用函数“GetTagSWordState”。第一个函数参数是要监视的 WinCC 变量的名称。第二个参数给出返回的状态值要写入的地方。

```
#include "apdefap.h"

int gscAction( void )
{
    DWORD dwState = 0;
    GetTagSWordState("Variable_01",&dwState);

    if ( dwState == 0 )
    {
        //Connection OK
        SetTagBit("BINi_E_CONNECTION",FALSE);
    }
    else
    {
        //Connection Error
        SetTagBit("BINi_E_CONNECTION",TRUE);
    }

    return 0;
}
```

在内部变量 BINi_E_CONNECTION 中输出变量状态。在发生错误的情况下，该变量值被设置为 TRUE。例如，在错误操作时，该变量可以用于触发报警或显示出错消息。

19.9.6 如何检查内部变量

简介

如果运行系统中内部变量不具有预期的数值，使用下列步骤来检查变量。

要求

- 已组态内部变量。
- WinCC 项目已激活。

步骤

1. 在 WinCC 项目管理器中打开“变量管理”(Tag Management)。
2. 在导航区域中选择条目“内部变量”以及要检查的变量。
3. 要在数据区域中显示“质量代码”(Quality Code) 和“值”(Value) 列，可能需要转到“显示”(Show)，然后在列标题的快捷菜单中选择这些列。
4. 检查质量代码。如果显示值“80”，则该变量值正确。其它数值的描述可以参见“变量质量代码”。
5. 如果质量代码不等于“80”，请检查右侧属性下的设置。
6. 检查是否已组态上限值和下限值或起始值。这些数值可以影响显示。
7. 如果变量值被其中一个已组态的数值改变，则取消激活该项目并改变该限制值或替换值。



说明

运行系统中只显示变量值和质量代码。

参见

变量的质量代码 (页 562)

索引

A&E 服务器

- 映射 WinCC 消息类别和消息类型,
- OPC A&E 服务器,
- 测试,
- 连接 WinCC - OPC A&E 客户端
 - OPC A&E 的示例,
 - 作为 ActiveX 控件,

“

- “MPI” 通道单元, 354
 - 组态连接, 354
- “SIMATIC S5 PROFIBUS FDL” 通道, 540
 - 诊断选项, 528
- “SIMATIC S7 Protocol Suite” 通道, 528
 - 诊断选项, 528
- “Slot PLC” 通道单元, 361
 - 组态连接, 361
- “Soft PLC” 通道单元, 363
 - 组态连接, 363
- “System Info” 通道, 517
 - 诊断选项, 517
- “TCP/IP” 通道单元, 364
 - 组态连接, 365
- “命名连接” 通道单元, 356
 - 组态连接, 357

5

- 505-Ethernet (CP 1413-x)
 - 通道单元, 439

A

- A&E 服务器, 172, 173, 184, 186, 187
 - 层级访问, 186
 - 跟踪事件, 172, 184
 - 简单事件, 184
 - 条件事件, 184
 - 映射 WinCC 消息系统, 173, 187
 - 与条件相关的事件, 172
- AR_SENDAR_SEND-FunctioninWinCC, 378
- AS 数据类型, 19
 - 类型转换, 31

AS511, 321

C

- Configurator (OPC UA), 230
- CP1434 TF, 448, 449
- CPU 负载, 480
 - 系统信息示例, 491

E

ERROR 标记, 517, 529

I

- I/O 域组态, 137
 - OPC DA 的实例, 137
- I/O 字段
 - S5 Profibus FDL 的实例, 318
 - 系统信息示例, 489
- INFO 标记, 517, 529, 541, 550

M

- MELSEC FX3U 系列, (请参见 Mitsubishi 以太网)
- MELSEC Q 系列, (请参见 Mitsubishi 以太网)
- Mitsubishi 以太网, 67
 - 变量, 74
 - 地址类型, 72
 - 受支持的控制器, 69
 - 数据类型, 68, 72
 - 通道单元, 67, 70, 71
 - 组态变量, 69, 72, 74
 - 组态连接, 69, 70, 71

O

OPC

- HDA 服务器浏览器, 165
- OPC A&E 服务器的功能, 184
- OPC UA, 227
- OPC 条目管理器, 89
- OPC 条目管理器功能概述, 89
- OPC 通讯受到干扰时的错误处理, 107, 256
- OPC1, 101, 118
- OPCServer.WinCC-(DPC_4001), 97
- ProgID, 134

- WinCC OPC A&E 服务器的功能, 172
- WinCC OPC DA 服务端的功能, 132
- WinCC OPC DA 客户端的功能, 94
- WinCC OPC HDA 服务器的功能, 153
- WinCC OPC UA 服务器的工作原理,
- WinCC OPC XML 服务器的功能, 127
- WinCC OPC XML 客户端的功能, 112
- WinCC 中的 OPC, 87, 124
- WinCC 作为 OPC DA 客户端, 87
- WinCC 作为 OPC DA 客户机, 107
- WinCC 作为 OPC UA 客户端, 256
- WinCC 作为 OPC XML 客户端, 87
- 变量选择, 89
- 查询 Progid, 89
- 创建连接, 89
- 地址属性, 101, 118
- 访问变量, 95
- 跟踪, 226
- 规范, 123
- 过滤标准, 97
- 兼容性, 123
- 结构属性, 104
- 实例, 141, 144, 147, 148, 149, 152
- 实例: , 136, 137, 138, 163
- 使用 HDA 服务器浏览器组态对 WinCC 归档变量的访问, 166
- 使用多个 OPC DA 服务端, 133
- 示例, 140, 142
- 所支持的 WinCC 数据类型, 93
- 所支持的 WinCC 数据类型概述, 93
- 添加变量, 89, 97
- 通道诊断, 550
- 通信概念, 124
- 新建连接, 97
- 新用户, 126
- 用 OPC 条目管理器组态对 WinCC 变量的访问, 97
- 在 OPC 计算机上设置用户帐户, 126
- 在 WinCC OPC DA 客户端上使用结构, 103
- 在 WinCC OPC DA 客户机上使用结构, 105
- 在 WinCC OPC DA 客户机上组态 OPC 通道, 96
- 在不使用 OPC 条目管理器的情况下组态对 WinCC 变量的访问, 101, 118
- 组态对 WinCC 消息系统的访问, 182
- 组态结构, 104
- 组态结构变量, 104
- OPC A 和原始数据, 181
- OPC A&E 服务器, 127, 172, 175, 181, 182, 184, 186, 187
 - 质量代码, 181, 195
 - 组态对 WinCC 消息系统的访问, 182
- OPC A&E 服务器的功能
 - 跟踪事件, 184
 - 简单事件, 184
 - 条件事件, 184
- OPC A&E 服务器上的消息类别, 187
- OPC A&E 服务器上的消息类型, 187
- OPC A&E 客户端
 - OPC A&E 的示例,
- OPC DA 服务端, 132
 - WinCC OPC DA 服务端的功能, 132
 - 使用多个 OPC DA 服务端, 133
- OPC DA 客户端, 94
 - WinCC OPC DA 客户端的功能, 94
- OPC HDA 服务器, 153
 - WinCC OPC HDA 服务器的时间格式, 158
 - 范围值, 154
 - 工作原理, 153
 - 配件, 156
 - 属性, 155
 - 数据结构, 154, 155
 - 条目标识号, 154
 - 条目句柄, 154
 - 写访问, 161
 - 原始数据, 153
 - 支持的函数, 158
 - 质量代码, 160, 181
- OPC UA
 - WinCC 变量的显示,
 - Configurator, 230
 - Conformance units, 210
 - WinCC OPC UA Configurator, 230
 - WinCC 作为 OPC UA 客户端, 227
 - 访问变量, 243
 - 记录变量的显示, 213
 - 配置文件, 210
 - 数据类型, 229
- OPC UA Configurator, 230
- OPC UA 服务器
 - URL, 201
 - 安全概念, 202
 - 工作原理, 201
 - 实例证书, 203
 - 受信任的客户端证书, 204
 - 通信配置文件, 201
 - 支持的规范, 201
 - 组态文件, 222
- OPC UA 客户端, 227
- OPC UA 历史访问, 214
- OPC UA 数据访问, 214
- OPC XML, 130
 - IIS 中的安全性设置, 130

测试安装, 130
 访问变量, 113
 OPC XML 服务器, 127
 安装, 128
 OPC XML 客户端, 112
 WinCC OPC XML 客户端的功能, 112
 OPC 条目管理器, 89, 97, 134
 OPC 条目管理器功能概述, 89
 OPC 通道, 550
 在 WinCC OPC DA 客户机上组态 OPC 通道, 96
 诊断选项, 550
 OPC 通信, 256
 处理 OPC UA 通信中的错误, 256
 OPC 通讯, 107
 OPC DA 通讯受到干扰时的错误处理, 107
 OPC1, 101, 118
 OPCScout new project
 OPC DA 的实例, 144
 OPCServer.WinCC-(DPC_4001), 97

P

PROFIBUS FMS, 257
 PROFIBUS FMS 通道单元, 259, 262
 变量的地址, 264, 266
 连接参数, 259
 数据类型, 258
 组态变量, 263, 264, 266
 PROFIBUS FMS 通讯驱动程序
 PROFIBUSFMS, 257
 PROFIBUS 通道单元, 359
 组态连接, 359
 PROFIBUS 系统参数, 535
 ProglD, 89
 查询, 134
 查询 ProglD, 89

S

S5 AS511, 323
 变量地址, 325
 定义按字节访问的变量, 329
 数据类型, 322
 组态, 323
 组态按位访问的变量, 328
 组态变量, 325
 S5 Ethernet Layer 4, 269
 iNA960 消息, 291
 SCI 消息, 292
 变量地址, 275
 传输参数, 283, 285

连接参数, 272
 连接受干扰时的错误代码, 287
 内部错误代码和常量, 287
 设备名称, 283, 284
 数据类型, 271
 通道单元的系统参数, 283
 原始数据变量, 281
 组态, 272
 组态按位访问的变量, 277
 组态按字访问的变量, 280
 组态按字节访问的变量, 278
 组态变量, 275
 S5 Profibus FDL, 295
 被动连接, 295
 变量参数, 301
 变量属性, 304, 305, 306, 311
 地址属性, 304, 305, 306, 311
 读/写监视时间, 307
 访问 ARRAY OF BYTE 数据类型的 S5 变量, 297
 访问 BIT 数据类型的 S5 变量, 297
 访问 BYTE 数据类型的 S5 变量, 297
 访问 S5 变量, 297
 服务访问点, 295
 更改读/写监视时间, 309
 更改设备名称, 308
 功能块, 316
 类型转换, 299
 连接参数, 301, 302, 312, 314
 连接属性, 302, 312, 314
 启动块, 316
 设备名称, 307
 实例, 316
 示例, 318
 数据类型, 297
 数据区, 297
 特殊功能, 310
 添加新驱动程序, 301
 通道单元 FDL (CP5412/A2-1), 295
 通道单元的系统参数, 307
 通道的功能, 299
 通道诊断, 540
 系统参数, 308, 309
 循环块, 316
 原始数据变量作为字节数组, 310
 支持的数据类型, 297
 主动连接, 295
 组态 SIMATIC S5 Profibus FDL, 301
 组态按位访问的变量, 304
 组态按字访问的变量, 306
 组态按字节访问的变量, 305
 组态被动数据传送, 314
 组态变量, 304

- 组态连接, 302
 - 组态数据处理块, 316
 - 组态通道 SIMATIC S5 Profibus FDL, 301
 - 组态通讯类型, 312
 - 组态原始数据变量, 311
 - 组态主动数据传送, 312
 - S5 Serial 3964R, 331
 - 变量的数据类型, 332
 - 组态, 333
 - 组态按位访问的变量, 337
 - 组态按字节访问的变量, 338
 - 组态变量, 335
 - S5 变量, 297
 - 访问 ARRAY OF BYTE 数据类型的 S5 变量, 297
 - 访问 BIT 数据类型的 S5 变量, 297
 - 访问 S5 变量, 297
 - 访问字节数据类型的变量, 297
 - S7 Protocol Suite, 342
 - AR_SEND 变量的属性概述, 385
 - AR_SEND 功能数据块的结构和参数, 380
 - 更改逻辑设备名称, 376
 - 将原始数据变量组态为字节数组, 412
 - 连接受干扰时的错误代码, 427
 - 软件冗余, 418
 - 软件冗余 - 连接专用的内部变量, 420
 - 使用 FB S7ProtocolSuite:AR_SEND 变量进行数据交换, 378
 - 数据块结构 AR_SEND 的实例, 389, 390, 391, 396, 398, 400, 402, 405
 - 数据类型, 348
 - 所支持的数据类型, 348
 - 通道单元, 344, 350
 - 通道诊断, 528
 - 系统参数, 372
 - 用于 BSEND/BRCV 功能的原始数据变量, 414
 - 用于多个归档变量的 AR_SEND 变量, 392
 - 用于多个归档变量的 AR_SEND 变量 (已优化), 404
 - 用于一个归档变量的 AR_SEND 变量, 387
 - 原始数据变量, 410
 - 原始数据变量作为字节数组, 411
 - 在 WinCC 中删除软件冗余, 426
 - 周期性读取服务, 373
 - 组态, 349
 - 组态按位访问的变量, 367
 - 组态按字访问的变量, 370
 - 组态按字节访问的变量, 368
 - 组态连接, 352, 354, 357, 359, 361, 363, 365
 - 组态软件冗余, 424
 - 组态文本变量, 371
 - 组态系统参数, 374
 - 组态用于“BSEND/BRCV”功能的原始数据变量, 417
 - 组态用于多个归档变量的 AR_SEND 变量, 408
 - 组态用于归档变量的 AR_SEND 变量, 406
 - SIMATIC S5 Profibus FDL, 295
 - S5 Profibus FDL, 295
 - 特殊功能, 310
 - 原始数据变量, 310
 - SIMATIC S5 Programmers Port AS511, 321
 - SIMATIC S5 Serial 3964R, 331, 333
 - SIMATIC S5 Serial 3964R ,
 - SIMATIC S7 Protocol Suite, 342
 - SIMATIC S7-1200, S7-1500 Channel
 - 数据类型, 430
 - 组态, 431, 432
 - 组态连接, 432
 - SIMATIC TI Ethernet Layer 4, 439
 - 传输参数, 447
 - 通道, 439
 - SIMOTION 通道, 461
 - 数据类型, 462
 - 系统参数, 468, 469, 471
 - 诊断, 473
 - 组态, 463, 464, 465, 467, 468
 - System Info, 478
 - 空闲驱动器容量, 480
 - 实例, 503, 505
 - 实例:, 494
 - 实例: , 502
 - 使用的数据类型, 487, 500
 - 示例, 487, 500, 501
 - 所支持的系统信息概述, 480
 - 通道诊断, 517
 - 与其它软件组件的不同之处, 485
 - 组态, 486
 - System Info 系统信息
 - 在多用户系统中使用, 499
- ## T
- TI Ethernet Layer 4, 439
 - 按位访问, 445
 - 按字节访问, 446
 - 变量的地址, 443
 - 传输参数, 447
 - 连接参数, 439, 441
 - 设备名称, 447
 - 数据类型, 440
 - 系统参数, 439, 447, 448, 449
 - 组态, 441
 - 组态按位访问的变量, 445

- 组态按字节访问的变量, 446
- 组态变量, 443
- TI Serial, 451
 - 变量的地址, 455
 - 变量地址, 455
 - 连接参数, 453
 - 数据类型, 452
 - 组态, 453
 - 组态按位访问的变量, 456
 - 组态按字节访问的变量, 458
 - 组态变量, 454
- TTY 接口, 321
- U**
- URL
 - OPC UA 服务器, 201
- W**
- WinAC Basis, 363
- WinAC Pro, 361
- WinCC
 - WinCC 中的 OPC, 87, 107, 124, 256
 - WinCC 中的 OPC UA, 227
 - 作为 OPC DA 客户端, 87
 - 作为 OPC DA 客户机, 107
 - 作为 OPC UA 客户端, 227, 256
 - 作为 OPC XML 客户端, 87
- WinCC - Microsoft Excel 的连接, 148
 - OPC DA 的实例, 148
- WinCC - OPC HDA 客户机连接, 163
 - OPC HDA 的实例, 163
- WinCC - SIMATIC NET FMS OPC 服务器的连接, 140
 - OPC DA 的实例, 140
- WinCC - SIMATIC NET S7 OPC 服务端的连接, 142
 - OPC DA 的实例, 142
- WinCC - WinCC 连接, 136
 - OPC DA 的实例, 136
- WinCC Explorer-OPC_Client.MPC, 134
- WinCC OPC A&E 服务器
 - 层级访问, 186
- WinCC OPC A&E 服务器的功能
 - 简单事件, 172
- WinCC OPC UA Configurator, 230
- WinCC OPC UA 服务器, 201
 - Discovery 服务器, 202
 - 组态, 223
- WinCC 过程通讯, 17
- WinCC 类型转换, 19
- WinCC 数据类型, 19, 23
 - WinCC 数据类型, 23
 - 类型转换, 23, 31
- WinCC 通道诊断, 519, 536, 546, 556
- WinCC 通讯, 17
 - 常规步骤, 17
- WinCC 消息系统
 - OPC A&E 服务器上的 WinCC 消息等级, 175
 - 属性, 176, 189
 - 映射 WinCC 消息等级和消息类型, 175
 - 在 OPC A&OPC-A&WinCC 消息系统上, 173
 - 组态对 WinCC 消息系统的访问, 182
- WinCC 中的 AR_SEND 功能
 - 变量的属性概述, 385
 - 数据块, 结构和参数, 380
 - 数据块结构的实例, 389, 390, 391, 396, 398, 400, 402, 405
 - 用于多个归档变量的变量, 392
 - 用于多个归档变量的变量 (已优化), 404
 - 用于归档变量的变量, 387
 - 组态用于多个归档变量的变量, 408
 - 组态用于归档变量的变量, 406
- 按**
- 按位访问, 43, 277, 367
 - S5 AS511, 328
 - S5 Ethernet Layer 4, 277
 - S5 Profibus FDL, 304
 - S5 Serial 3964R, 337
 - S7 Protocol Suite, 367
 - TI Ethernet Layer 4, 445
 - TI Serial, 456
- 按字访问, 367
 - S7 Protocol Suite, 370
- 按字节访问, 43, 278, 367
 - S5 AS511, 329
 - S5 Ethernet Layer 4, 278
 - S5 Profibus FDL, 305
 - S5 Serial 3964R, 338
 - S7 Protocol Suite, 368
 - TI Ethernet Layer 4, 446
 - TI Serial, 458
- 棒**
- 棒图
 - 系统信息示例, 490

变

- 变量, 19, 275, 367, 487, 558, 560
 - HDA 服务器浏览器, 165
 - OPC DA 的实例, 144, 147, 152
 - OPC HDA 的实例, 166, 167
 - OPC 条目管理器, 97
 - PowerTag, 19
 - System Info 实例, 487, 502
 - 变量选择, 89
 - 测试变量组态, 535, 545
 - 检查, 521, 538, 549, 558
 - 检查内部变量, 573
 - 添加变量, 89
 - 通过二进制写入组态变量, 44
 - 外部, 19
 - 文本变量的长度定义, 19
 - 在自动化系统中对外部变量寻址, 19
 - 指定线性转换, 22
 - 质量, 560
 - 质量代码, 560, 562
 - 状态, 560, 568
 - 组态, 263, 264, 266, 275, 304, 443, 454
 - 组态 HDA 服务器浏览器, 166
 - 组态 OPC 条目管理器, 97, 101
 - 组态 PC 条目管理器, 118
 - 组态 powertag, 22
 - 组态按位访问, 277, 304, 337, 367, 445, 456
 - 组态按字访问, 280, 306, 370
 - 组态按字节访问, 278, 305, 338, 368, 446, 458
 - 组态变量地址, 335
 - 组态外部变量, 22
 - 组态文本变量, 22
- 变量的地址, 443
 - TI Ethernet Layer 4, 443
- 变量的质量代码, 560, 562
- 变量地址, 275
 - PROFIBUS FMS, 257, 258, 264, 266
 - S5 Ethernet Layer 4, 269
 - TI Serial, 451
- 变量属性, 22, 304, 305, 306, 311
 - S5 Profibus FDL 的实例, 318
- 变量状态, 560, 568, 570, 572
 - 使用全局动作监视变量状态, 572
 - 在动态对话框中监视变量状态, 570

标

- 标准标记, 514

超

- 超时, 241

传

- 传输参数, 17, 283
 - SIMATIC S5 Ethernet Layer 4, 283
 - SIMATIC TI Ethernet Layer 4, 447
 - TI Ethernet Layer 4, 447
- 传送更改, 373

磁

- 磁盘容量, 480
 - 空闲驱动器容量, 480, 485
 - 系统信息示例, 490

错

- 错误点, 492

打

- 打印机监控, 480
 - 系统信息示例, 496

地

- 地址类型
 - Mitsubishi 以太网, 72
- 地址属性, 101, 118, 304, 305, 306, 311
 - S5 Profibus FDL 的实例, 318

电

- 电子服务器
 - WinCC OPC A&OPC 的功能, 127

定

- 定时器, 480
- 定义新变量, 144
 - OPC DA 的实例, 144

读

读/写监视时间, 307, 309
更改, 309

多

多用户系统, 499
通道的使用, 499

二

二进制写入机制, 43
一般步骤, 43

范

范围值, 154

服

服务访问点, 295
服务器证书, 246

跟

跟踪功能, 514
组态, 514

工

工业以太网通道单元, 351
组态连接, 352

功

功能块, 316
循环块, 316

过

过程画面
System Info 实例, 503
过滤标准, 97

缓

缓冲时间, 182

计

计数器, 480

记

记录册条目, 517, 529, 541, 550
记录册文件, 511

兼

兼容性, 123

结

结构
使用 WinCC OPC 客户机, 105
在 WinCC OPC DA 客户端上使用结构, 103
组态结构, 104
结构变量, 104
组态结构变量, 104
结构属性, 104

客

客户端证书, 246

类

类型转换, 19, 31, 299

连

连接, 17, 350, 556
被动连接, 295
测试, 556
测试连接组态, 535, 545
创建连接, 89
创建新连接, 22
检查, 519, 536, 546
主动连接, 295
组态, 302, 352, 354, 357, 359, 361, 363, 365
连接 WinCC - OPC A&E 客户端, 181

连接参数, 17, 272, 301, 302, 312, 314, 350, 545

PROFIBUS FMS, 257

S5 AS511, 321, 323

S5 Ethernet Layer 4, 269

S7 Protocol Suite, 342

SIMATIC S5 Serial 3964R, 331

TI Ethernet Layer 4, 439, 441

TI Serial, 451

通道单元, 441

连接监控, 94

连接属性, 22, 302, 312, 314

逻

逻辑设备名称, 372

修改, 376

启

启动块, 316

确

确认策略, 192

确认方法, 178

日

日期, 480

日志文件, 517, 529, 541, 550

软

软件冗余, 418

连接专用的内部变量, 420

在 WinCC 中删除, 426

在 WinCC 中组态, 424

设

设备名称, 283, 307, 447

TI Ethernet Layer 4, 447

更改, 308

时

时间, 480, 485

系统信息示例, 489

时间戳, 380

实

实例: , 572

实例证书, 246

属

属性 - CP5613, 532, 534, 542, 544

数

数据传输, 312

组态被动数据传送, 314

组态主动数据传送, 312

数据类型, 19, 93, 229, 271, 348, 430, 480

Mitsubishi 以太网, 68, 72

OPC UA, 229

PROFIBUS FMS, 257, 258

S5 AS511, 322

S5 Ethernet Layer 4, 271

S5 Profibus FDL, 297

S5 Serial 3964R, 332

S7 Protocol Suite, 348

SIMATIC S7-1200, S7-1500 Channel, 430

TI Ethernet Layer 4, 440

类型转换, 31

使用的数据类型, 480, 487

所支持的 WinCC 数据类型, 93

所支持的 WinCC 数据类型概述, 93

数据区, 297

数值变量类型, 19

线性标定, 19

特

特殊功能, 310

添

添加变量, 89, 97

OPC DA 的实例, 138

添加新驱动程序, 301

条

条目标识号, 154

条目句柄, 154

调

调试, 226

通

通道, 17, 257, 269, 331, 342, 556

Mitsubishi 以太网, (请参见 Mitsubishi 以太网)
PROFIBUS FMS, 257

S5 PROFIBUS FDL 的实例, 316

SIMATIC S5 Ethernet Layer 4, 269

SIMATIC S5 Programmers Port AS511, 321

SIMATIC S5 Serial 3964R, 331

SIMATIC S7 Protocol Suite, 342

SIMATIC TI Ethernet Layer 4, 439

TI Serial, 451

测试, 556

检查, 519, 536, 546

通道的诊断, 509

状态 - 逻辑连接功能, 509

组态 SIMATIC S5 PROFIBUS FDL, 301

通道 S5Seral3964R 的设备状态监控

设备状态监控,

通道单元, 17, 269, 301, 344

505 Serial Unit #1, 451

505-Ethernet (CP 1413-x), 439

FDL (CP5412/A2-1), 301

Mitsubishi 以太网, (请参见 Mitsubishi 以太网)

MPI, 354

PROFIBUS, 359

PROFIBUS-FMS, 259

S5 Transport (CP1413-x), 272

S5 Transport (TCP/IP), 272

S5-AS511, 321, 323

S5-RK512 (3964R), 331

SIMATIC S7 Protocol Suite S7 Protocol Suite 通道选

择: 选择通道单元, 344

Slot PLC, 361

Soft PLC, 363

TCP/IP, 364

工业以太网, 351

连接参数, 441

命名连接, 356

系统参数, 283, 307, 372, 447, 448

通道单元 FDL (CP5412/A2-1), 295, 301

通道诊断, 511

“SIMATIC S5 PROFIBUS FDL” 通道, 540

“SIMATIC S7 Protocol Suite” 通道, 528

“System Info” 通道, 517

OPC 通道, 550

PROFIBUS 系统参数, 535

WinCC 通道诊断, 519, 536, 546, 556

测试通道, 519, 536, 546, 556

跟踪功能的标准标记, 514

函数,

记录册条目的 ERROR 标记, 517, 529

记录册条目的 INFO 标记, 517, 529, 541, 550

检查变量, 521, 538, 549, 558

检查连接, 519, 536, 546, 556

检查内部变量, 573

检查通信处理器的组态, 532, 534, 542, 544

检查组态数据, 555

具有状态 - 逻辑连接功能的通道,

连接参数, 545

日志文件, 517, 529, 541, 550

实例, 572

使用全局动作监视变量状态, 572

属性 - CP5613, 532, 534, 542, 544

通道, 使用, 512, 513

通道诊断, 509, 511, 512, 513

通道诊断, 使用, 511, 512

系统参数, 545

在动态对话框中监视变量状态, 570

诊断选项, 517, 528, 540, 550

状态 - 逻辑连接, 509

状态消息概述, 519, 536, 546, 556

组态通道的跟踪功能, 514

作为 Windows 应用程序的通道诊断, 513

通道诊断 通道诊断

使用通道诊断进行通道诊断, 512

通道诊断 通道诊断, 513

通道诊断跟踪文件, 511

通道诊断通道

测试,

通信

AS 数据类型, 22

变量属性, 22

创建新连接, 22

连接属性, 22

通过二进制写入组态变量, 44

位/字节变量, 44

组态 powertag, 22

通信处理器, 532, 534, 542, 544

检查通信处理器的组态, 532, 534, 542, 544

通信驱动程序, (请参见通道)

通讯, 15

AS 数据类型, 31

PowerTag, 19

WinCC 过程通讯, 17

WinCC 和自动化系统之间的通讯, 17

WinCC 类型转换, 19

WinCC 数据类型, 19

WinCC 通讯的原理, 17
 按 AS 数据类型排序格式改编, 31
 按 WinCC 数据类型排序的格式改编, 23
 二进制写入机制, 43
 二进制写入机制的原理, 43
 基本信息, 15
 类型转换, 19, 31
 连接, 17
 数据类型, 19, 31
 数值变量类型的线性标定, 19
 通道单元, 17
 通讯驱动程序, 17
 位/字节访问, 43
 文本变量的长度定义, 19
 在自动化系统中寻址, 19
 值范围, 31
 通讯类型, 312
 组态, 312
 通讯驱动程序, 17

网

网络, 15

位

位/字节变量, 44
 位/字节访问, 43

文

文本变量, 19, 367
 组态, 22, 371

系

系统参数, 17, 283, 372, 468, 469, 471, 545
 S5 Ethernet Layer 4, 269
 S5 Profibus FDL, 295
 SIMATIC S7 Protocol Suite, 342
 TI Ethernet Layer 4, 439, 447, 448, 449
 通道单元, 447, 448
 组态, 374, 376
 系统信息
 I/O 域组态, 489
 棒图组态, 490
 错误点, 492
 示例: , 489, 490, 491, 492, 496
 消息文本, 492

要监视的变量, 492
 状态显示组态, 496
 系统诊断
 设备视图, 522
 详细视图, 522
 诊断缓冲区视图, 522
 组态, 526

夏

夏令时/标准时间, 380

项

项目
 OPC DA 的实例, 138

消

消息
 System Info 实例, 494
 系统信息示例, 492
 消息文本, 492

新

新建连接, 97
 新用户, 126

要

要监视的变量, 492

用

用户帐户, 126
 公布 OPC 计算机, 126

原

原始数据, 153
 原始数据变量, 310, 410
 用于 BSEND/BRCV 功能, 414
 组态, 311
 组态为字节数组, 412
 组态以使用 BSEND/BRCV 功能, 417
 作为字节数组, 310, 411

诊

- 诊断
 - 通道, 473
- 诊断选项, 517, 528, 540, 550
 - “SIMATIC S5 PROFIBUS FDL” 通道, 540
 - “SIMATIC S7 Protocol Suite” 通道, 528
 - “System Info” 通道, 517
 - OPC 通道, 550

证

- 证书, 246

支

- 支持的系统信息, 480
 - 概述, 480

值

- 值范围, 31

质

- 质量代码, 160, 181, 195

周

- 周期性读取服务, 373

状

- 状态 - 逻辑连接功能状态 - 逻辑连接, 509
- 状态显示, 519, 536, 546, 556
 - 系统信息示例, 496

自

- 自动化设备, 299

字

- 字节数组, 310

组

- 组态
 - TI Ethernet Layer 4, 441
 - WinCC OPC UA 服务器, 223
 - 安全设置, 224
 - 变量, 443
 - 端口号, 224
 - 用户标识, 225
 - 优化的 WinCC 归档写访问, 225
 - 组态按位访问
 - 变量, 445
 - 组态按位访问的变量
 - TI Ethernet Layer 4, 445
 - 组态按字节访问
 - 变量, 446
 - 组态按字节访问的变量
 - TI Ethernet Layer 4, 446
 - 组态变量
 - TI Ethernet Layer 4, 443
 - 组态数据, 555
 - 组态文件
 - WinCC OPC UA 服务器的 URL, 222
 - WinCC 归档写访问, 223
 - 安全设置, 222
 - 布局, 222
 - 端口号, 222
 - 文件位置, 222
 - 用户标识, 223

最

- 最大大小, 182

