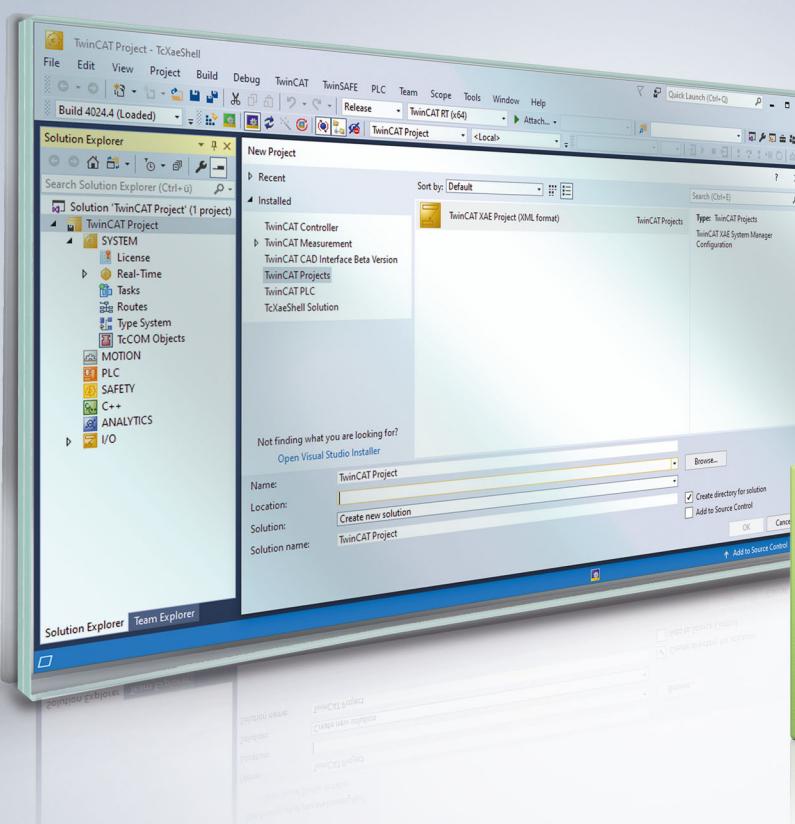


手册 | ZH TE1000

TwinCAT 3 | ADS Basics



目录

1 前言	5
1.1 文档说明	5
1.2 安全信息	6
1.3 信息安全说明	6
2 ADS 设备概念	8
3 ADS 设备标识	9
4 客户端-服务器关系	12
5 AMS/TCP 包	13
5.1 结构 AMS/TCP 包	13
5.2 AMS/TCP 报头	13
5.3 AMS 报头	13
5.4 ADS 命令	15
5.4.1 命令概览	15
5.4.2 ADS 读取设备信息	15
5.4.3 ADS 读取	16
5.4.4 ADS 写入	16
5.4.5 ADS 读取状态	17
5.4.6 ADS 写入控制	17
5.4.7 ADS 添加设备通知	18
5.4.8 ADS 删除设备通知	18
5.4.9 ADS 设备通知	19
5.4.10 ADS 读写	20
6 ADS 设备规范	22
6.1 一般信息	22
6.2 PLC 服务规范	22
6.3 ADS 系统服务规范	22
6.4 NC 的规范	26
6.4.1 Ring-0-Manager 规范	28
6.4.2 通道规范	29
6.4.3 规范组	34
6.4.4 规范轴	43
6.4.5 编码器规范	79
6.4.6 控制器规范	88
6.4.7 规范器驱动	92
6.4.8 表规范	97
6.4.9 附录	101
7 通过防火墙的 ADS 连接	114
8 ADS 返回代码	115

1 前言

1.1 文档说明

本说明仅适用于熟悉国家标准且经过培训的控制和自动化工程专家。

在安装和调试组件时，必须遵循文档和以下说明及解释。

操作人员应具备相关资质，并始终使用最新的生效文档。

相关负责人员必须确保所述产品的应用或使用符合所有安全要求，包括所有相关法律、法规、准则和标准。

免责声明

尽管本文档经过精心编制，然而，所述产品正在不断开发中。

我们保留随时修订和更改本文档的权利，恕不另行通知。

不得依据本文档中的数据、图表和说明对已供货产品的修改提出赔偿。

商标

Beckhoff®、TwinCAT®、TwinCAT/BSD®、TC/BSD®、EtherCAT®、EtherCAT G®、EtherCAT G10®、EtherCAT P®、Safety over EtherCAT®、TwinSAFE®、XFC®、XTS® 和 XPlanar® 是德国倍福自动化有限公司的注册商标并已获得授权。

本文档中所使用的其它名称可能是商标名称，任何第三方为其自身目的而引用，都可能触犯商标所有者的权利。

正在申请的专利

涵盖 EtherCAT 技术，包括但不限于以下专利申请和专利：

EP1590927、EP1789857、EP1456722、EP2137893、DE102015105702

并在多个国家进行了相应的专利申请或注册。



EtherCAT® 是注册商标和专利技术，由德国倍福自动化有限公司授权使用。

版权所有

© 德国倍福自动化有限公司。

未经明确授权，不得复制、分发、使用和传播本文档内容。

违者将被追究赔偿责任。德国倍福自动化有限公司保留所有发明、实用新型和外观设计专利权。

1.2 安全信息

安全规范

为了确保您的使用安全, 请务必仔细阅读并遵守本文档中每个产品的安全使用说明。

责任免除

所有组件在供货时都配有适合应用的特定硬件和软件配置。严禁未按文档所述修改硬件或软件配置, 否则, 德国倍福自动化有限公司对由此产生的后果不承担责任。

人员资格

本说明仅供熟悉适用国家标准的控制、自动化和驱动工程专家使用。

警示性词语

文档中使用的警示信号词分类如下。为避免人身伤害和财产损失, 请阅读并遵守安全和警告注意事项。

人身伤害警告

⚠ 危险

存在死亡或重伤的高度风险。

⚠ 警告

存在死亡或重伤的中度风险。

⚠ 谨慎

存在可能导致中度或轻度伤害的低度风险。

财产或环境损害警告

注意

可能会损坏环境、设备或数据。

操作产品信息



这些信息包括：
有关产品的操作、帮助或进一步信息的建议。

1.3 信息安全说明

Beckhoff Automation GmbH & Co.KG (简称 Beckhoff) 的产品, 只要可以在线访问, 都配备了安全功能, 支持工厂、系统、机器和网络的安全运行。尽管配备了安全功能, 但为了保护相应的工厂、系统、机器和网络免受网络威胁, 必须建立、实施和不断更新整个操作安全概念。Beckhoff 所销售的产品只是整个安全概念的一部分。客户有责任防止第三方未经授权访问其设备、系统、机器和网络。它们只有在采取了适当的保护措施的情况下, 方可与公司网络或互联网连接。

此外, 还应遵守 Beckhoff 关于采取适当保护措施的建议。关于信息安全和工业安全的更多信息, 请访问本公司网站 <https://www.beckhoff.com/secguide>。

Beckhoff 的产品和解决方案持续进行改进。这也适用于安全功能。鉴于持续进行改进, Beckhoff 明确建议始终保持产品的最新状态, 并在产品更新可用后马上进行安装。使用过时的或不支持的产品版本可能会增加网络威胁的风险。

如需了解 Beckhoff 产品信息安全的信息，请订阅 <https://www.beckhoff.com/secinfo> 上的 RSS 源。

2 ADS 设备概念

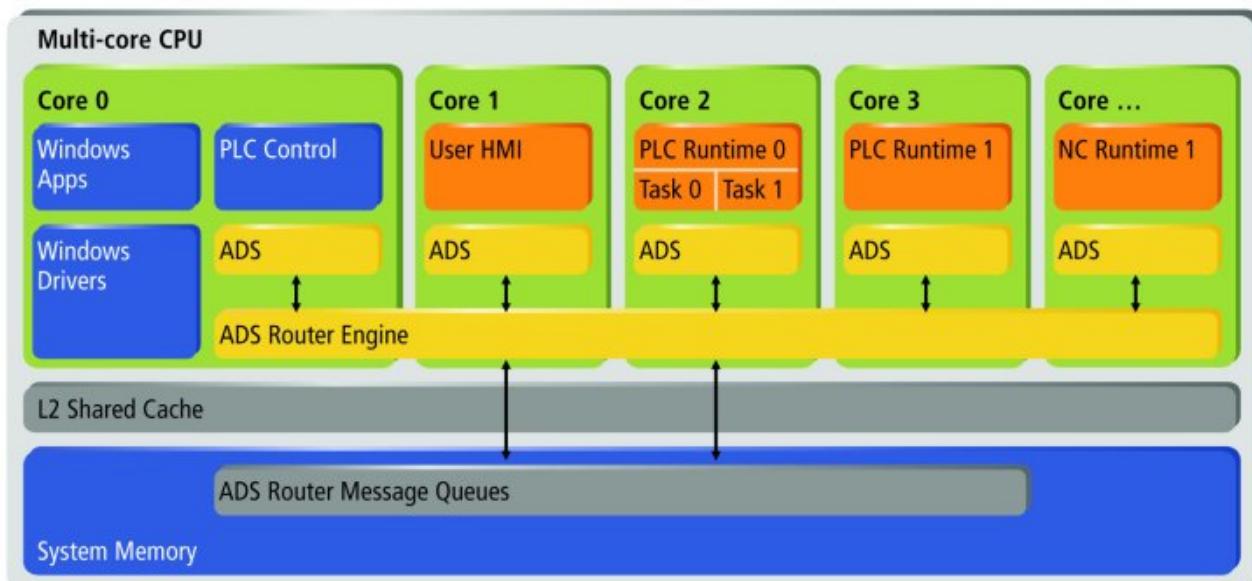
TwinCAT 系统架构允许将软件的各个模块（如 TwinCAT PLC、用户 HMI……）视为独立的设备：每个任务都有一个软件模块（“服务器”或“客户端”）。系统中的服务器是以软件形式执行工作的“设备”，其操作行为与硬件设备完全一样。因此，我们可以说在软件中实施了“虚拟”设备。“客户端”是请求“服务器”服务的程序，比如一个可视化工具，甚至是以程序形式存在的“编程设备”。因此，TwinCAT 具有可扩展性，因为总是有新的服务器和客户端用于凸轮轴控制器、示波器、PID 控制器等任务……

这些对象之间的消息由“消息路由器”通过一致的 ADS（自动化设备规范）接口进行交换。它管理和分配系统中以及通过 TCP/IP 连接进行的所有消息。

每个 TwinCAT 设备上都有 TwinCAT 消息路由器。

这样，所有 TwinCAT 服务器和客户端程序都可以交换命令和数据、发送消息、传输状态信息等……

下图显示了基于 ADS 的 TwinCAT 设备概念：



3 ADS 设备标识

ADS 设备的唯一标识借助两个标识符来实现：

- PortNr
- NetId

AMS 端口

TwinCAT 消息路由器中的 ADS 设备通过一个编号唯一标识，该编号称为 ADS 端口号。ADS 设备的端口号为指定和固定编号，而纯 ADS 客户端应用程序（如 HMI 系统）在首次访问消息路由器时会获分配可变端口号。

已分配以下 AMS 端口号：

AMS 端口	设备
1	ADS 路由器
2	AMS 调试器
10	TCom 服务器
11	TCom 服务器任务, 实时上下文 (RT context)
12	TCom 服务器, 被动级别
20	TwinCAT 调试器
21	TwinCAT 调试器任务
30	授权服务器
100	记录器
110	事件记录器
120	EtherCAT 设备的应用程序
130	事件记录器用户模式 (V2)
131	实时事件记录器 (V2)
132	事件记录器发布服务器 (V2)
200	Ring 0 实时
290	Ring 0 追踪
300	Ring 0 IO
400	Ring 0 PLC (传统)
500	Ring 0 NC
501	Ring 0 NC SEC
511	Ring 0 NC SPP
520	NC 实例
550	Ring ISG
600	Ring 0 CNC
700	Ring 0 线路
800	Ring 0 TC2 PLC
801	TC2 PLC 运行时系统 1
811	TC2 PLC 运行时系统 2
821	TC2 PLC 运行时系统 3
831	TC2 PLC 运行时系统 4
850	Ring 0 TC3 PLC
851	TC3 PLC 运行时系统 1
852	TC3 PLC 运行时系统 2

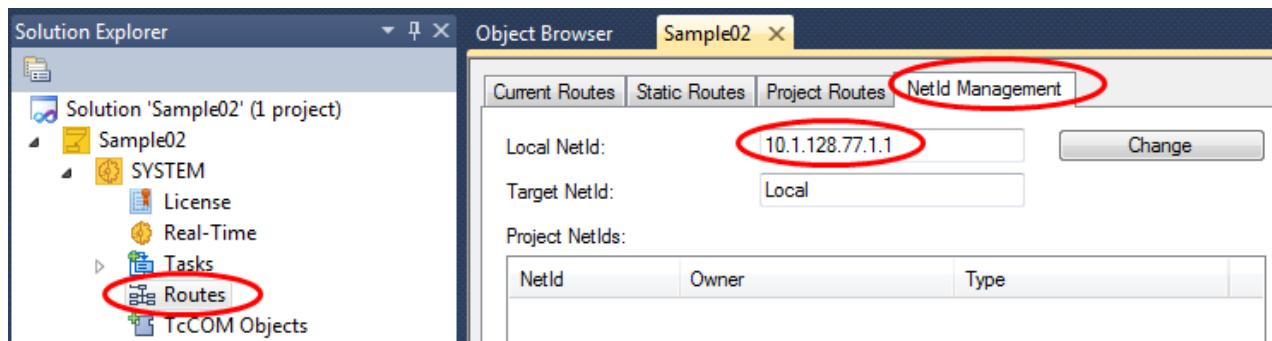
AMS 端口	设备
853	TC3 PLC 运行时系统 3
854 - ...	TC3 PLC 运行时系统 4 - ...
900	凸轮控制器
950	凸轮设计工具
1000-1199	Ring 0 IO 端口
2000	Ring 0 用户
2500	Crestron 服务器
10000	系统服务
10201	TCP/IP 服务器
10300	系统管理器
10400	SMS 服务器
10500	Modbus 服务器
10502	AMS 记录器
10600	XML 数据服务器
10700	自动配置
10800	PLC 控制
10900	FTP 客户端
11000	NC 控制系统
11500	NC 解释器
11600	GST 解释器
12000	追踪控制
13000	凸轮控制
14000	Scope Server
14100	状态监测
15000	正弦 CH1
16000	控制网
17000	OPC 服务器
17500	OPC 客户端
18000	邮件服务器
19000	虚拟 COM EL60xx
19100	管理服务器
19200	Miele@home 服务器
19300	CP-Link 3
19310	触摸锁
19500	视觉服务
19800	HMI 服务器
21372	数据库服务器
25000-25999	ADS 服务器保留端口范围
25013	FIAS 服务器
25014	Bang&Olufsen 服务器
26000-26999	客户私有端口范围
32768-65535	ADS 客户端保留端口范围

AMS NetId

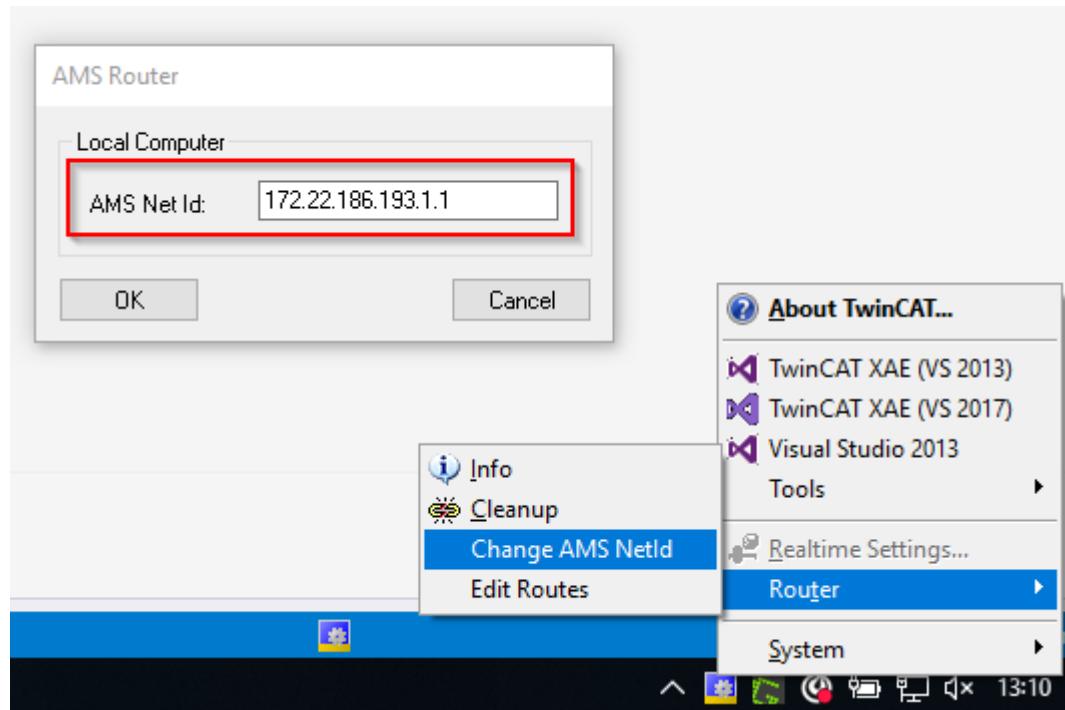
网络中的每台 TwinCAT 设备都由 AMS NetId 标识。AMS NetId 由六个八位字节组成。前四个八位字节可自行选定。最后两个八位字节（通常为 .1.1）用作现场总线或其他设备的子网掩码。AMS NetId 对所有通信伙伴来说都必须具有唯一性。

配置：

本地或远程 TwinCAT 设备的 AMS NetId 可在 TC3 项目的 SYSTEM\Routes\NetId 管理中设置。



另外，也可以通过 TwinCAT SysTray 菜单中的路由器类别在本地配置 AMS NetId。更改 AMS NetId 后必须重新启动设备。



4 客户端-服务器关系

ADS 服务最初可分为“已确认”和“未确认”服务。

ADS 客户端	ADS 服务器
请求 ->	-> 指示
确认 <-	<- 响应

ADS 通信的进程始于 ADS 请求，该请求到达 ADS 服务器时会被视为 ADS 指示。

ADS 服务器对 ADS 响应进行回复，ADS 客户端收到该响应作为 ADS 确认。

由 ADS 服务器自主发送的信息（例如，错误或其他状态信息）会被 ADS 客户端记录为通知指示。

一般 ADS 服务

一般 ADS 通信服务分为

- 异步
 - 客户端向服务器发出 ADS 请求
 - 客户端继续运行（没有 ADS 确认）
 - 服务器处理 ADS 请求，并通过回调向客户端提供结果（向客户端发送 ADS 确认）
- 通知
 - 客户端向服务器登记 ADS 请求，以获取特定服务
 - 服务器通过回调（向客户端发送 ADS 确认）自主向客户端提供服务，直到客户端取消其服务请求为止
 - 这种通信方式的优点是额外的 ADS 协议开销较低，因为客户端程序不会周期性地发出 ADS 请求

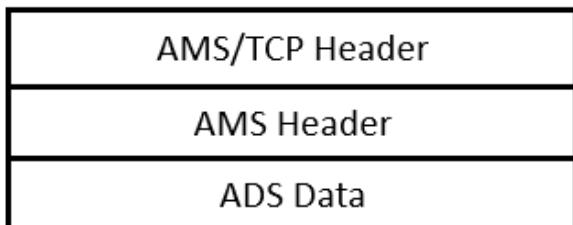
特定 ADS 服务

除了一般 ADS 服务外，还定义了封装 ADS 通信和允许使用 Visual Basic 或 Visual C++ 等进行操作的附加功能。这些“特定 ADS 服务”可在 ADS-OCX 或 ADS-DLL 中实现，并提供同步通信设施等功能，或考虑可能存在的限制（例如，在 Visual Basic 中，没有前导符号的变量类型）。

- 同步
 - 客户端向服务器发出 ADS 请求
 - 发出调用的客户端线程在 ADS 通信期间暂停运行
 - 当 ADS 请求调用返回时，来自 ADS 服务器的结果已经可用
 - 这种通信方式的优点是客户端程序所需的管理开销较低

5 AMS/TCP 包

5.1 结构 AMS/TCP 包



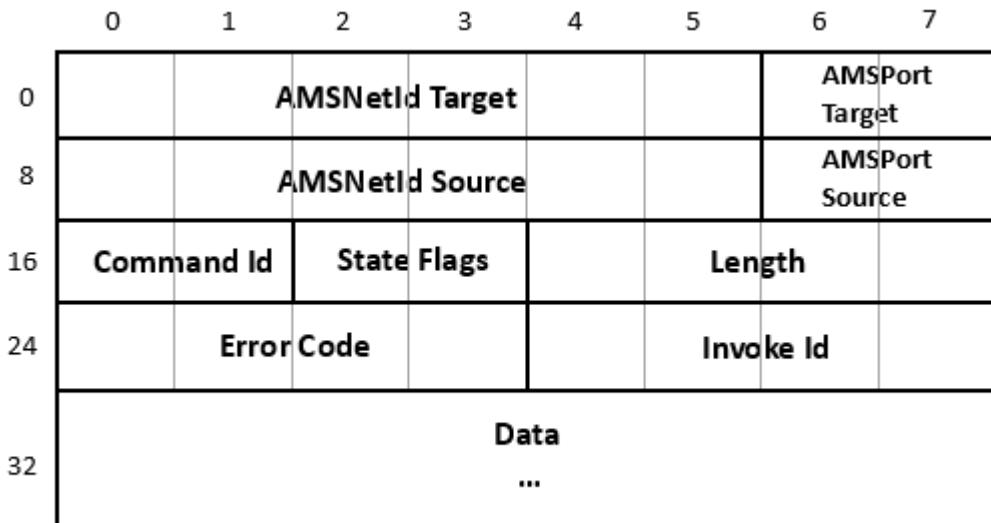
数据数组	大小	描述
AMS/TCP 报头	6 字节	包含数据包的长度。
AMS 报头	32 字节	AMS/TCP 报头包含发送器和接收器的地址。此外还有 AMS 错误代码、ADS 命令 ID 和其他一些信息。
ADS 数据	n 字节	ADS 数据范围包含单个 ADS 命令的参数。数据数组的结构取决于 ADS 命令。某些 ADS 命令不需要额外数据。

5.2 AMS/TCP 报头



数据数组	大小	描述
保留	2 字节	这些字节必须设置为 0。
长度	4 字节	该数组包含数据包的长度。它由 AMS 报头和所附 ADS 数据组成。单位是字节。

5.3 AMS 报头



数据数组	大小	描述
AMSNNetId 目标	6 字节	这是数据包接收站的 AMSNNetId。备注请参见下文 [▶ 14]。
AMSPort 目标	2 字节	这是数据包要接收站的 AMSPort。
AMSNNetId 来源	6 字节	这包含数据包发出站的 AMSNNetId。
AMSPort 来源	2 字节	这段信息包含了该站点发送该数据包的AMSPort。
命令 Id	2 字节	请参见下文 [▶ 14]。
状态旗标	2 字节	请参见下文 [▶ 14]。
数据长度	4 字节	数据范围的大小。单位为字节。
错误代码	4 字节	AMS 错误编号。请参见“ADS 返回代码”。
调用 Id	4 字节	可用的 32 位自由数组。通常，该数组用于发送一个 Id。有了这个 Id，就可以将收到的响应分配给之前发送的请求。
数据	n 字节	数据范围。数据范围包含考虑 ADS 命令的参数。

AMS Net Id

AMSNNetId 由 6 个字节组成，并用于寻址发送器或接收器。例如，一个可能的 AMSNNetId 是 172.16.17.10.1.1。本例中的存储方式如下所示：

	0	1	2	3	4	5
0	172	16	17	10	1	1

AMSNNetId 是纯逻辑的，通常与 IP 地址无关。AMSNNetId 在目标系统配置。为此，在 PC 上使用了 TwinCAT 系统控制器。如果使用其他硬件，请参见相关文档，了解有关 AMS NetId 设置的注意事项。

命令 Id

命令	描述
0x0000	无效
0x0001	ADS 读取设备信息 [▶ 15]
0x0002	ADS 读取 [▶ 16]
0x0003	ADS 写入 [▶ 16]
0x0004	ADS 读取状态 [▶ 17]
0x0005	ADS 写入控制 [▶ 17]
0x0006	ADS 添加设备通知 [▶ 18]
0x0007	ADS 删除设备通知 [▶ 18]
0x0008	ADS 设备通知 [▶ 19]
0x0009	ADS 读写 [▶ 20]

其他命令没有定义或不在内部使用。因此，命令 Id 只允许包含上述枚举值！

状态旗标

旗标	描述
0x0001	0：请求/1：响应
0x0004	ADS 命令

第一个位标志它是请求还是响应。第三个位必须设置为 1，以便与 ADS 命令交换数据。其他位没有定义或用于其他内部用途。

因此，其他位必须设置为 0！

旗标	描述
0x000x	TCP 协议
0x004x	UDP 协议

第 7 个位标志它是使用 TCP 还是 UDP 传输。

5.4 ADS 命令

5.4.1 命令概览

命令	描述
ADS 读取设备信息 [▶ 15]	读取 ADS 设备的名称和版本号。
ADS 读取 [▶ 16]	通过 ADS 读取，可从 ADS 设备读取数据
ADS 写入 [▶ 16]	通过 ADS 写入，可将数据写入 ADS 设备。
ADS 读取状态 [▶ 17]	读取 ADS 状态和 ADS 设备的设备状态。
ADS 写入控制 [▶ 17]	更改 ADS 状态和 ADS 设备的设备状态。
ADS 添加设备通知 [▶ 18]	在 ADS 设备中创建通知。
ADS 删除设备通知 [▶ 18]	在 ADS 设备中删除一个之前定义的通知。
ADS 设备通知 [▶ 19]	数据将从 ADS 设备独立传输到客户端
ADS 读写 [▶ 20]	通过 ADS 读写，数据将被写入 ADS 设备。此外，还可从 ADS 设备读取数据。

5.4.2 ADS 读取设备信息

读取 ADS 设备的名称和版本号。

请求

无需额外数据

响应

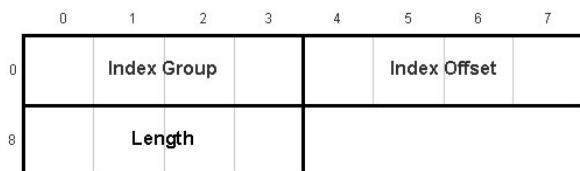
0	1	2	3	4	5	6	7
0	Result			Major Version	Minor Version	Version	Build
8				Device name			
16							

数据数组	大小	描述
结果	4 字节	ADS 错误编号。
主要版本	1 字节	主要版本号
次要版本	1 字节	次要版本号
版本构建	2 字节	构建号
设备名称	16 字节	ADS 设备的名称

5.4.3 ADS 读取

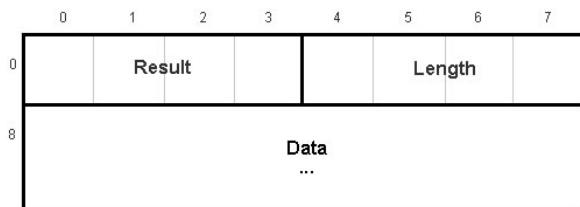
通过 ADS 读取，可从 ADS 设备读取数据。数据由索引组和索引/偏移寻址

请求



数据数组	大小	描述
索引组	4 字节	应读取数据的索引组。
索引偏移	4 字节	应读取数据的索引偏移。
长度	4 字节	应读取数据的长度（以字节为单位）。

响应

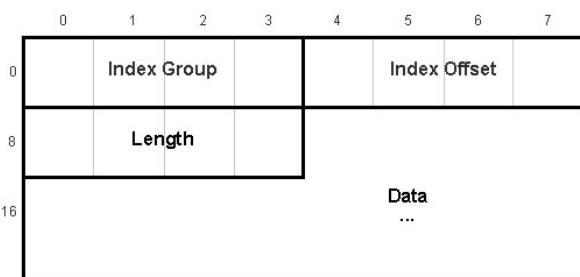


数据数组	大小	描述
结果	4 字节	ADS 错误编号
长度	4 字节	传回数据的长度。
数据	n 字节	传回的数据。

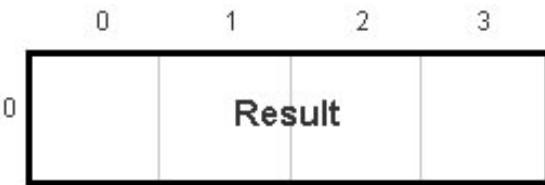
5.4.4 ADS 写入

通过 ADS 写入，可将数据写入 ADS 设备。数据由索引组和索引/偏移寻址

请求



数据数组	大小	描述
索引组	4 字节	应写入数据的索引组
索引偏移	4 字节	应写入数据的索引偏移
长度	4 字节	写入数据的长度（以字节为单位）
数据	n 字节	写入 ADS 设备的数据。

响应

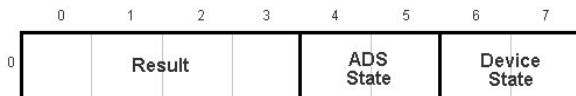
数据数组	大小	描述
结果	4 字节	ADS 错误编号

5.4.5 ADS 读取状态

读取 ADS 状态和 ADS 设备的设备状态。

请求

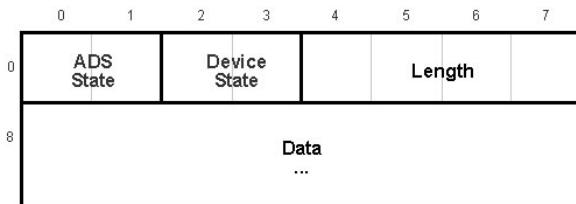
无需额外数据

响应

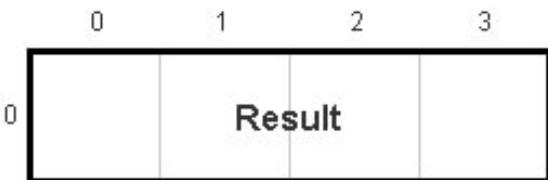
数据数组	大小	描述
结果	4 字节	ADS 错误编号。
ADS 状态	2 字节	ADS 状态（请参见 ADS-DLL 的数据类型 ADSSTATE）。
设备状态	2 字节	设备状态

5.4.6 ADS 写入控制

更改 ADS 状态和 ADS 设备的设备状态。此外，还可以向 ADS 设备发送数据，以传输更多信息。这些数据没有从当前的 ADS 设备（PLC、NC.....）进行分析

请求

数据数组	大小	描述
ADS 状态	2 字节	新 ADS 状态（请参见 ADS-DLL 的数据类型 ADSSTATE）。
设备状态	2 字节	新设备状态。
长度	4 字节	数据长度（以字节为单位）。
数据	n 字节	发送到 ADS 设备的附加数据

响应

数据数组	大小	描述
结果	4 字节	ADS 错误编号。

5.4.7 ADS 添加设备通知

在 ADS 设备中创建通知。

注意：我们建议每个设备公布的通知不超过 550 个。否则，通过使用结构或 sum 命令来增加有效负载。

请求

数据数组	大小	描述
索引组	4 字节	每个通知应发送数据的索引组。
索引偏移	4 字节	每个通知应发送数据的索引偏移。
长度	4 字节	每个通知应发送数据的长度（以字节为单位）。
传输模式	4 字节	请参见 ADS-DLL 的 ADSTRANSMODE 结构描述。
最大延迟	4 字节	最迟在此时间之后，将调用 ADS 设备通知。单位是 1 ms。
循环时间	4 字节	ADS 服务器会检查值在该时间片内是否发生变化。单位是 1 ms
保留	16 字节	必须设置为 0

响应

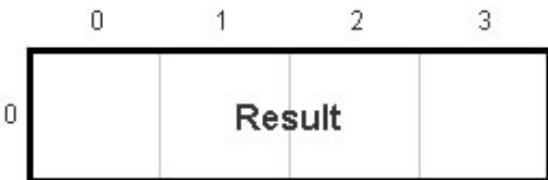
数据数组	大小	描述
结果	4 字节	ADS 错误编号
通知句柄	4 字节	通知句柄

5.4.8 ADS 删除设备通知

在 ADS 设备中删除一个之前定义的通知。

请求

数据数组	大小	描述
通知句柄	4 字节	通知句柄。该句柄由 ADS 命令添加设备通知创建

响应

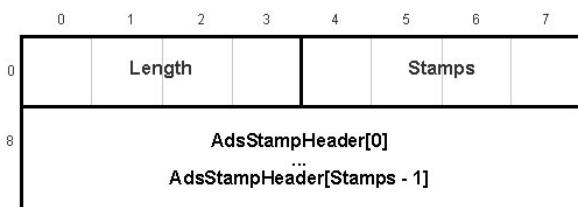
数据数组	大小	描述
结果	4 字节	ADS 错误编号

5.4.9 ADS 设备通知

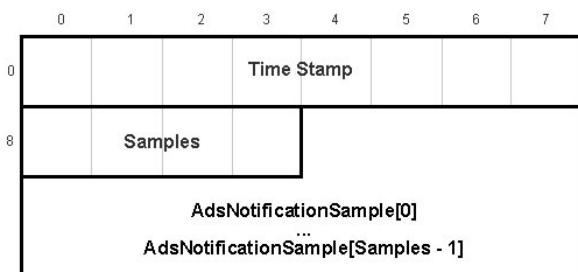
数据将从 ADS 设备独立传输到客户端。

请求

在设备通知处传输的数据是多重嵌套的。通知流包含一个数组，其中的元素类型为 *AdsStampHeader*。该数组同样包含 *AdsNotificationSample* 类型的元素。

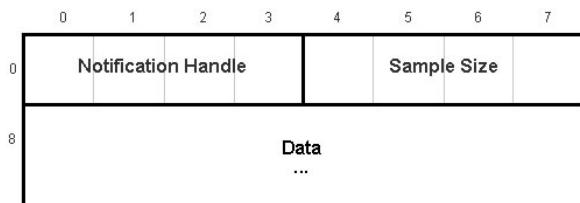
AdsNotificationStream

数据数组	大小	描述
长度	4 字节	数据大小（以字节为单位）。
戳记	4 字节	<i>AdsStampHeader</i> [▶ 19] 类型的元素数量
<i>AdsStampHeader</i>	n 字节	包含 <i>AdsStampHeader</i> [▶ 19] 类型元素的数组

AdsStampHeader

数据数组	大小	描述
TimeStamp	8 字节	时间戳采用 Windows FILETIME 格式进行编码。也就是说，该值包含自 1.1.1601 以来的 100 纳秒间隔数。此外，也不考虑当地时间的变化。因此，时间戳表现为通用协调时间 (UTC)。
示例	4 字节	AdsNotificationSample [▶ 20] 类型的元素数量
AdsNotificationSample	n 字节	包含 AdsNotificationSample [▶ 20] 类型元素的数组

AdsNotificationSample



数据数组	大小	描述
通知句柄	4 字节	通知句柄。
样例大小	4 字节	数据范围的大小（以字节为单位）。
数据	n 字节	数据



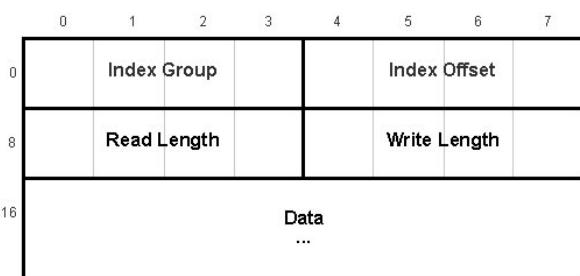
如果您的句柄失效，系统将会发送一个不带数据的提示通知作为建议。

5.4.10 ADS 读写

通过 *ADS ReadWrite*，数据将被写入 ADS 设备。此外，还可从 ADS 设备读取数据。

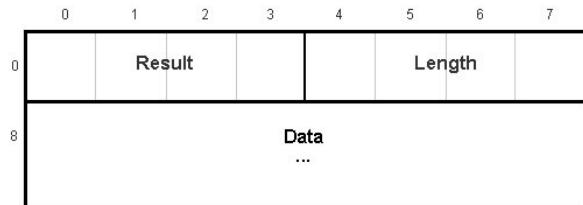
可读取数据由索引组和索引偏移寻址

请求



数据数组	大小	描述
索引组	4 字节	应写入数据的索引组。
索引偏移	4 字节	应写入数据的索引偏移
读取长度	4 字节	应读取数据的长度（以字节为单位）。
写入长度	4 字节	应写入数据的长度（以字节为单位）
数据	n 字节	写入 ADS 设备的数据。

响应



数据数组	大小	描述
结果	4 字节	ADS 错误编号
长度	4 字节	传回数据的长度。
数据	n 字节	传回的数据。

6 ADS 设备规范

6.1 一般信息

PLC 软件可以描述为虚拟现场单元（自动化设备），因为它是一个纯软件 PLC。因此，它为其他通讯伙伴（如其他虚拟现场单元或 Windows 程序）提供一个倍福 ADS（自动化设备规范）接口，通过该接口可以对其进行参数设置或查询。使用 ADS 标准可以统一访问 PLC，并将它整合到现有的虚拟现场单元中。

读取和写入操作通过两个数字在 PLC 接口（由 ADS 定义）上进行：索引组（16 位）和索引偏移（32 位）。关于 PLC 的 ADS 界面，我们将在下文中有关组和偏移指数的页面中进行详细介绍。

PLC 的“索引组”规范

ADS 单元的四个全局范围如下所示，PLC 作为索引组中的四个部分：

索引组 (0x = hex)	索引组说明
0x00000000 0x00000FFF	保留
0x00001000	PLC ADS 参数范围
0x00002000	PLC ADS 状态范围
0x00003000	PLC ADS 单元功能范围
0x00004000	PLC ADS 服务（包括访问 PLC 内存范围（%M 字段）的服务）[▶ 22]
0x00006000 0x0000EFFF	为 PLC ADS 扩展保留
0x0000F000 0x0000FFFF	一般 TwinCAT ADS 系统服务（包括访问 PLC 物理输入和输出过程图的服务）[▶ 22]

6.2 PLC 服务规范

本节包括访问 PLC 内存范围（%M 字段）的服务。

索引组	索引偏移	访问	数据类型	描述	备注
0x00004020	0x00000000-0x0000FFFF	R/W	UINT8[n]	READ_M - WRITE_M PLC 内存范围（%M 字段）。偏移为字节偏移。	
0x00004021	0x00000000-0xFFFFFFFF	R/W	UINT8	READ_MX - WRITE_MX PLC 内存范围（%MX 字段）。索引偏移的低位字为字节偏移。索引偏移包含根据字号 *8 + 位号计算出的位地址	
0x00004025	0x00000000	R	ULONG	PLCADS_IGR_RMSIZE 内存范围过程图的字节长度	
0x00004030	0x00000000-0xFFFFFFFF	R/W	UINT8	PLCADS_IGR_RWRB 保留数据范围。索引偏移是字节偏移	
0x00004035	0x00000000	R	ULONG	PLCADS_IGR_RRSIZE 保留范围的字节长度	
0x00004040	0x00000000-0xFFFFFFFF	R/W	UINT8	PLCADS_IGR_RWDB 数据范围。索引偏移为字节偏移。	
0x00004045	0x00000000	R	ULONG	PLCADS_IGR_RDSIZE 数据范围的字节长度	

6.3 ADS 系统服务规范

本节将介绍对每个 TwinCAT ADS 单元具有相同意义和作用的 ADS 服务。本节还包括访问物理输入和输出 PLC 过程图的服务。

索引组	索引偏移	访问	数据类型	描述
0x0000F003	0x00000000	读写	W: UINT8[n] R: UINT32	GET_SYMHANDLE_BYNAME 给写入数据中所包含的名称分配一个句柄 (code word) , 并作为结果返回给调用者。
0x0000F004	0x00000000			保留。
0x0000F005	0x00000000-0xFFFFFFFF=symHandle	R/W	UINT8[n]	READ_/_WRITE_SYMVAL_BYHANDLE 读取由 ,symHdl' 标识的变量值或为变量赋值。,symHdl' 必须首先由 GET_SYMHANDLE_BYNAME 服务确定。
0x0000F006	0x00000000	W	UINT32	RELEASE_SYMHANDLE 对于一个被查询的、有特定名称的 PLC 变量, 其写入数据中包含的代码 (句柄) 被释放。
0x0000F020	0x001F400-0xFFFFFFFF	R/W	UINT8[n]	READ_I - WRITE_I 物理输入的 PLC 流程图 (%I 字段) 。偏移是字节偏移。
0x0000F021	0x000FA000-0xFFFFFFFF	R/W	UINT8	READ_IX - WRITE_IX 物理输入的 PLC 流程图 (%IX 字段) 。索引偏移包含位地址, 其计算公式为基偏移 (0xFA000) + 字节号 + 8 + 位号
0x0000F025	0x00000000	R	ULONG	ADSIGRP_IOIMAGE_RISIZEB yte物理输入的 PLC 流程图的字节长度。
0x0000F030	0x0003E800-0xFFFFFFFF	R/W	UINT8[n]	READ_Q - WRITE_Q 物理输出的 PLC 流程图 (%Q 字段) 。偏移是字节偏移。
0x0000F031	0x001F4000-0xFFFFFFFF	R/W	UINT8	READ_QX - WRITE_QX 物理输出的 PLC 流程图 (%QX 字段) 。索引偏移包含位地址, 其计算公式为基偏移 (0x1F4000) + 字节号 *8 + 位号。
0x0000F035	0x00000000	R	ULONG	ADSIGRP_IOIMAGE_ROSIZE 物理输出 PLC 流程图的字节长度。
0x0000F080	0x00000000-0xFFFFFFFF=n (内部子命令数) n (最大值) = 500	读写	读取: n * ULONG[3] := IG1, IO1, Len1, IG2, IO2, Len2, ..., IG(n), IO(n), Len(n) 写入: n * ULONG + UINT8[Len1] + UINT8[Len2] + ..., + UINT8[Len(n)] :=	ADSIGRP_SUMUP_READ 写入数据包含多个单独的 AdsReadReq(IG, IO, Len, Data) 子命令的列表。读取数据包含返回代码列表, 后接请求数据。

索引组	索引偏移	访问	数据类型	描述
			Result1, Result2, ..., Result(n), Data1, Data2, ..., Data(n)	
0x0000F081	0x00000000- 0xFFFFFFFF= n (内部子命令数) n (最大值) = 500	读写	<p>读取： (n * ULONG[3]) + UINT8[Len1] + UINT8[Len2] + ..., + UINT8[Len(n)] := IG1, IO1, Len1, IG2, IO2, Len2, ..., IG(n), IO(n), Len(n), Data1, Data2, ..., Data(n)</p> <p>写入： n * ULONG := Result1, Result2, ..., Result(n)</p>	ADSGRP_SUMUP_WRITE 写入数据包含多个单独的 AdsWriteReq(IG, IO, Len, Data) 子命令的列表。 读取数据包含返回代码列表。
0x0000F082	0x00000000- 0xFFFFFFFF= n (内部子命令数) n (最大值) = 500	读写	<p>读取： (n * ULONG[4]) + UINT8[WriteLen1] + UINT8[WriteLen2] + ..., + UINT8[WriteLen(n)] := IG1, IO1, ReadLen1, WriteLen1, IG2, IO2, ReadLen2, WriteLen2, ..., IG(n), IO(n), ReadLen(n), ..., WriteLen(n), WriteData1, WriteData2, ..., WriteData(n)</p> <p>写入： (n * ULONG[2]) + UINT8[ReturnLen1] + UINT8[ReturnLen2] + ..., + UINT8[ReturnLen(n)] := Result1, ReturnLen1, Result2, ReturnLen2, ..., Result(n), ReturnLen(n), ReadData1, ReadData2, ..., ReadData(n)</p>	ADSGRP_SUMUP_READWRITE 写入数据包含多个单独的 AdsReadWriteReq(IG, IO, readLen, writeLen, Data) 子命令的列表。 读取数据包含返回代码和返回数据长度列表，后接请求数据。
0x0000F083	0x00000000- 0xFFFFFFFF= n (内部子命令数) n (最大值) = 500	读写	<p>读取： n * ULONG[3] := IG1, IO1, Len1, IG2, IO2, Len2, ..., IG(n), IO(n), Len(n)</p>	ADSGRP_SUMUP_READEX 写入数据包含多个单独的 AdsReadReq(IG, IO, Len, Data) 子命令的列表。读取数据包含

索引组	索引偏移	访问	数据类型	描述
			<p>写入:</p> <pre>n * ULONG + UINT8[Len1] + UINT8[Len2] + ... + UINT8[Len(n)] := Result1, Result2, ..., Result(n), Data1, Data2, ..., Data(n)</pre>	返回代码列表，后接请求数据。
0x0000F084	0x00000000-0xFFFFFFFF= n (内部子命令数) n (最大值) = 500	读写	<p>读取:</p> <pre>n * ULONG[3] := IG1, IO1, Len1, IG2, IO2, Len2, ... IG(n), IO(n), Len(n)</pre> <p>写入:</p> <pre>n * ULONG + UINT8[Len1] + UINT8[Len2] + ... + UINT8[Len(n)] := Result1, Result2, ..., Result(n), Data1, Data2, ..., Data(n)</pre>	ADSIGRP_SUMUP_READEX2 写入数据包含多个单独的 AdsReadReq(IG, IO, Len, Data) 子命令的列表。读取数据包含返回代码列表，后接请求数据。
0x0000F085	0x00000000-0xFFFFFFFF= n (内部子命令数) n (最大值) = 500	读写	<p>读取:</p> <pre>(n * ULONG[3]) := IG1, IO1, Len1, IG2, IO2, Len2, ... IG(n), IO(n), Len(n)</pre> <p>写入:</p> <pre>(n * ULONG) + UINT8[Len1] + UINT8[Len2] + ... + UINT8[Len(n)] := Result1, Result2, ..., Result(n), Handle1, Handle2, ..., Handle(n)</pre>	ADSIGRP_SUMUP_ADDDEVN OTE 写入数据包含多个单独的 AdsAddDeviceNotifications(IG, IO, Len, Data) 子命令的列表。读取数据包含返回代码列表，后接请求通知句柄。
0x0000F086	0x00000000-0xFFFFFFFF= n (内部子命令数) n (最大值) = 500	读写	<p>读取:</p> <pre>Handle1, Handle2, ..., Handle(n)</pre> <p>写入:</p> <pre>(n * ULONG) + UINT8[Len1] + UINT8[Len2] + ...,</pre>	ADSIGRP_SUMUP_DELDEVN OTE 写入数据包含多个句柄的列表。读取数据包含返回代码列表。

索引组	索引偏移	访问	数据类型	描述
			+ UINT8[Len(n)] := Result1, Result2, ..., Result(n)	

6.4 NC 的规范

本文档包含所有 TC3 特定修改和新功能。

索引组 (十六进制)	描述	备注
0x1000	Ring-0-Manager: 参数 [▶ 28]	可选!
0x1100	Ring-0-Manager: 状态 [▶ 28]	可选!
0x1200	Ring-0-Manager: 功能 [▶ 29]	可选!
0x1300	Ring-0-Manager: 循环性过程数据	未执行!
0x2000 + ID	具有相应 ID 的通道: 参数 [▶ 29]	
0x2100 + ID	具有相应 ID 的通道: 状态 [▶ 31]	
0x2200 + ID	具有相应 ID 的通道: 功能 [▶ 32]	
0x2300 + ID	具有相应 ID 的通道: 循环性过程数据 [▶ 33]	
0x3000 + ID	具有相应 ID 的组: 参数 [▶ 34]	可选!
0x3100 + ID	具有相应 ID 的组: 状态 [▶ 36]	可选!
0x3200 + ID	具有相应 ID 的组: 功能 [▶ 40]	可选!
0x3300 + ID	具有相应 ID 的组: 循环性过程数据	未执行!
0x4000 + ID	具有相应 ID 的轴: 参数 [▶ 43]	
0x4100 + ID	具有相应 ID 的轴: 状态 [▶ 54]	
0x4200 + ID	具有相应 ID 的轴: 功能 [▶ 62]	
0x4300 + ID	具有相应 ID 的轴: 循环性过程数据 [▶ 76]	
0x5000 + ID	具有相应 ID 的编码器: 参数 [▶ 79]	可选!
0x5100 + ID	具有相应 ID 的编码器: 状态 [▶ 82]	可选!
0x5200 + ID	具有相应 ID 的编码器: 功能 [▶ 85]	可选!
0x5300 + ID	具有相应 ID 的编码器: 循环性过程数据 [▶ 86]	可选!
0x6000 + ID	具有相应 ID 的控制器: 参数 [▶ 88]	可选!
0x6100 + ID	具有相应 ID 的控制器: 状态 [▶ 90]	可选!
0x6200 + ID	具有相应 ID 的控制器: 功能 [▶ 92]	可选!
0x6300 + ID	具有相应 ID 的控制器: 循环性过程数据	未执行!
0x7000 + ID	具有相应 ID 的驱动器: 参数 [▶ 92]	可选!
0x7100 + ID	具有相应 ID 的驱动器: 状态 [▶ 95]	可选!
0x7200 + ID	具有相应 ID 的驱动器: 功能 [▶ 96]	可选!
0x7300 + ID	具有相应 ID 的驱动器: 循环性过程数据 [▶ 96]	可选!

索引组 (十六进制)	描述	备注	
0x0A000 + ID	表具有相应 ID 的表 (n x m): 参数 [▶ 97] 0x0A000+ID 表示表 ID [1..255] 0x1A000+ID 表示表 ID [256..4095] ... 0xFA000+ID 表示表 ID [3840..4095]	最大表数扩展至 4095 (TC3.1 B4021 及以上版本)	
0x0A100 + ID	具有相应 ID 的表 (n x m): 状态 [▶ 100] 0x0000A100+IdLowByte 表示表 ID [1..255] 0x0001A100+IdLowByte 表示表 ID [256..4095] ... 0x000FA100+IdLowByte 表示表 ID [3840..4095] 0x000nA100+IdLowByte 表示表 ID [1..4095] (TabID = n * 256 + IdLowByte)		
0x0A200 + ID	具有相应 ID 的表 (n x m): 功能 [▶ 100] 0x0000A100+IdLowByte 表示表 ID [1..255] 0x0001A100+IdLowByte 表示表 ID [256..4095] ... 0x000FA100+IdLowByte 表示表 ID [3840..4095] 0x000nA100+IdLowByte 表示表 ID [1..4095] (TabID = n * 256 + IdLowByte)		
0x0A300 + ID	具有相应 ID 的表 (n x m): 循环性过程数据 0x0000A100+IdLowByte 表示表 ID [1..255] 0x0001A100+IdLowByte 表示表 ID [256..4095] ... 0x000FA100+IdLowByte 表示表 ID [3840..4095] 0x000nA100+IdLowByte 表示表 ID [1..4095] (TabID = n * 256 + IdLowByte)	未执行!	
0xF000 ... 0xFFFF	保留区域 (TwinCAT 系统区域)		
IndexGroup:	IndexOffset:		
0xF081	0x00000000 ... 0xFFFFFFFF (n 个元素)	ADSIGRP_SUMUP_WRITE 读写命令在多个单独的 ADS 写入命令 (如组请求) 的写入数据中包含一个列表。 写入数据结构: [IdxGrp(1), IdxOff(1), WriteLen(1), ..., IdxGrp(n), IdxOff(n), WriteLen(n), WriteData(1), ..., WriteData(n)] 读取数据结构: [Error(1), ..., Error(n)]	
0xF082	0x00000000 ... 0xFFFFFFFF (n 个元素)	ADSIGRP_SUMUP_READWRITE 读写命令在多个单独的 ADS 读写命令 (如组请求) 的写入数据中包含一个列表。 写入数据结构: [IdxGrp(1), IdxOff(1), ReadLen(1), WriteLen(1), ..., IdxGrp(n), IdxGrp(n), ReadLen(n), WriteLen(n), WriteData(1), ..., WriteData(n)] 读取数据结构: [Error(1), ReadLen(1), ..., Error(n), ReadLen(n), ReadData(1), ..., ReadData(n)]	
0xF084	0x00000000 ... 0xFFFFFFFF (n 个元素)	ADSIGRP_SUMUP_READ (READEX2) 读写命令在多个单独的 ADS 写入命令 (如组请求) 的写入数据中包含一个列表。 写入命令结构: [IdxGrp(1), IdxOff(1), ReadLen(1), ..., IdxGrp(n), IdxGrp(n), ReadLen(n)] 读取命令结构:	

索引组 (十六进制)	描述	备注
	[Error(1), ReadLen(1), ..., Error(n), ReadLen(n), ReadData(1), ..., ReadData(n)]	



6.4.1 Ring-0-Manager 规范

6.4.1.1 Ring-0 参数的"索引偏移"规范 (索引组 0x1000)

索引偏移 (十 六进制)	访问	Ring-0- Manager	数据类型	物理单位	定义范围	描述	备注
0x00000010	读取	每个	UINT32	100 ns		循环时间 SAF 任务	
0x00000012	读取	每个	UINT32	100 ns		循环时间 SVB 任务	
0x00000014	读取	每个	INT32	ns		全局时间补偿漂移 (针对 SAF 任务)	
0x00000020	读/写	每个	UINT16	1	0/1	循环数据一致性检查和 NC 设定点值校正	

6.4.1.2 Ring-0 状态的"索引偏移"规范 (索引组 0x1100)

索引偏移 (十 六进制)	访问	Ring-0- Manager	数据类型	物理单位	定义范围	描述	备注
0x00000001	读取	每个	UINT32	1	0, 1...255	通道数量	
0x00000002	读取	每个	UINT32	1	0, 1...255	组的数量	
0x00000003	读取	每个	UINT32	1	0, 1...255	轴的数量	
0x00000004	读取	每个	UINT32	1	0, 1...255	编码器的数量	
0x00000005	读取	每个	UINT32	1	0, 1...255	控制器的数量	
0x00000006	读取	每个	UINT32	1	0, 1...255	驱动器的数量	
0x0000000A	读取	每个	UINT32	1	0, 1...255	表的数量 (n x m)	
0x00000010	读取	每个	UINT32	1		循环时间错误计数器 SAF 任务 (不可范围化)	保留!
0x00000014	读取	每个	UINT32	1		IO 循环时间错误计数器 SAF 任务 (不可范围化)	保留!
0x00000020	读取	每个	UINT32	s		计算时间 SAF 任务 (不可 范围化)	保留!
0x00000031	读取	每个	UINT32[n]	1	0, 1...255	为系统中所有通道提供通 道 ID	
0x00000032	读取	每个	UINT32[n]	1	0, 1...255	为系统中所有组提供组 ID	
0x00000033	读取	每个	UINT32[n]	1	0, 1...255	为系统中所有轴提供轴 ID	
0x00000034	读取	每个	UINT32[n]	1	0, 1...255	为系统中所有编码器提供 编码器 ID	
0x00000035	读取	每个	UINT32[n]	1	0, 1...255	为系统中所有控制器提供 控制器 ID	
0x00000036	读取	每个	UINT32[n]	1	0, 1...255	为系统中所有驱动器提供 驱动器 ID	
0x0000003A	读取	每个	UINT32[n]	1	0, 1...255	为系统中所有表格提供表 格 ID	

索引偏移 (十六进制)	访问	Ring-0-Manager	数据类型	物理单位	定义范围	描述	备注
0x0000001nn	读取	每个	UINT32	1	0, 1...255	为编码器 ID 提供相应的轴 IDnn = 编码器 ID	保留!
0x0000002nn	读取	每个	UINT32	1	0, 1...255	为控制器 ID 提供相应的轴 IDnn = 控制器 ID	保留!
0x0000003nn	读取	每个	UINT32	1	0, 1...255	为驱动器 ID 提供相应的轴 IDnn = 驱动器 ID	保留!

6.4.1.3 适用于 Ring-0 功能的"索引偏移"规范 (索引组 0x1200)

索引偏移 (十六进制)	访问	Ring-0-Manager	数据类型	物理单位	定义范围	描述	备注
0x000000020	写入	每个	VOID	1		清除循环时间错误计数器 SAF 和 SVB	保留!

6.4.2 通道规范

6.4.2.1 通道参数的"索引偏移"规范 (索引组 0x2000 + ID)

索引偏移 (十六进制)	访问	通道类型	数据类型	物理单位	定义范围	描述	备注
0x000000001	读取	每个	UINT32	1		通道 ID	
0x000000002	读取	每个	UINT8[30+1]	1		通道名称	
0x000000003	读取	每个	UINT32	1	枚举	通道类型 [▶ 101]	
0x000000004	读取	每个	UINT32	1	枚举	解释器类型 [▶ 101]	
0x000000005	读取	每个	UINT32	1		程序加载缓冲区大小 (以字节为单位)	
0x000000006	读取	每个	UINT32	1		根据工作清单划分的程序编号	
0x000000007	读/写	每个	UINT32	1	枚举	设置加载日志模式 [▶ 101]	
0x000000008	读/写	每个	UINT32	1	枚举	设置追踪模式 [▶ 101]	
0x000000009	读/写	每个	UINT32	1		保留	
0x00000000A	读/写	每个	UINT32	1	0/1	在名为 "TcNci.log" 的日志文件中记录所有进给器操作记录	
0x00000000B	读/写	每个	UINT32	1	0/1	NC 记录器消息的通道特定级别 0: 仅错误 1: 所有 NC 消息	
0x000000010	读写	每个	写入				
			{				
			UINT32	1	0..159	M 函数的起始索引	
			UINT32	1	1..160	要读取的 M 函数数	
			}				
			读取 [n]				
			{				
			UINT8	1	0..159	M 函数的规则位掩码	
			INT32[10]	1	-1..159	要清除的 M 函数数	
			}				
0x000000011	写入	插补				写入 M 函数说明	仅在内部使用!
0x000000012	读/写	插补	LREAL64	1		G70 系数	
0x000000013	读/写	插补	LREAL64	1		G71 系数	
0x000000014	写入	插补	{			轴的用户自定义符号	尚未发布
			char[32]			用户符号 (空字符串结尾)	

索引偏移 (十六进制)	访问	通道类型	数据类型	物理单位	定义范围	描述	备注
			char[10]			系统符号 (空字符结尾)	
			}				
0x00000015	读/写	插补	UINT16 resp. UINT32	1	0/1 默认: FALSE	激活默认 G 代码	从 TC3.1 B4014 起更新
0x00000021	读取	每个	UINT32	1		组 ID (仅对 3D 和 FIFO 通道显式)	
0x00000031	读/写	插补	UINT16	1		解释器的标准输出端口	保留功能, 无标准!
0x00000032	读/写	插补	UINT16	1	0/1	笛卡尔工具偏移输入	保留功能, 无标准!
0x00000040	读/写	插补	{ char[6] UINT16 UINT32 UINT32 }			解释器钩子的目标地址	保留功能, 无标准!
0x00000050	读/写	插补	UINT32	1	枚举	如果在半径补偿过程中识别到瓶颈，会做出反应 0: 错误和中止 1: 注意和故障排除 2: 仅注意，不进行轮廓调制	
0x00000051	读/写	插补	UINT32	1	1..24	预期瓶颈检测	
0x00000052	读/写	插补	UINT32	1	0/1	倒角开启/关闭	保留功能, 无标准!
0x00000053	读/写	插补	UINT32	1		激活以读取当前有效插补规则、零点漂移和旋转 0: 关闭 1: 打开	
0x00000054	读/写	插补	UINT32	1	0/1	回描开启/关闭	保留功能, 无标准!
0x00000055	读/写	插补	UINT32[4]	1		为 UINT32 配置循环通道接口；最多可配置 4 个索引偏移。	
0x00000056	读/写	插补	UINT32[4]	1		为 LREAL 配置循环通道接口；最多可配置 4 个索引偏移。	
0x00010K0L	读/写	每个	REAL64	例如 mm [1..3]	±MAX REAL64 [1..3]	零点漂移值 (NPV) 轴索引 K=1 → X K=2 → Y K=3 → Z	
					[1..0xA]	L=1 → G54F L=2 → G54G L=3 → G55F ...	
0x0002ww00	读/写	每个	UINT16			工具编号：工具补偿值	
0x0003ww00	读/写	每个	UINT16		[1...50]	工具类型 ww = 工具 1...50	
0x0004wwnn	读/写	每个	REAL64		[1...14]	参数： nn = 索引 1...14	
0x000500gg	读/写	每个	REAL64	例如 mm [1...9]	≥ 0 (值) (g)	公差球面半径 gg = 通道组 (默认: 1)	

6.4.2.2 通道状态的"索引偏移"规范 (索引组 0x2100 + ID)

索引偏移 (十六进制)	访问	通道类型	数据类型	物理单位	定义范围	描述	备注
0x00000001	读取	每个	INT32	1	枚举	错误代码通道	
0x00000002	读取	每个	UINT32	1		通道中的组数	
0x00000003	读取	每个	UINT32	1	枚举	解释器状态 [▶ 101]	无法用示波器追踪!
0x00000004	读取	每个	UINT32	1	枚举	解释器/通道运行模式 [▶ 101]	
0x00000005	读取	每个	UINT32	1		当前加载的程序	
0x00000007	读取	每个	UINT8[...]	1		当前加载程序的程序名 (100个字符, 空字符结尾)	最多 100 个字符, 空字符结尾
0x00000008	读取	解释器	UINT32	1	[0,1]	解释器模拟模式 0: 关闭 (默认) 1: 打开	无法用示波器追踪!
0x00000010	读取	解释器	UINT32	1		文本索引 如果解释器处于中止状态, 则可在此读出当前文本索引	无法用示波器追踪!
0x00000011	读写	解释器	写入				无法用示波器追踪!
			UINT32	1		文本索引	
			读取				
			UINT8[..]	1		来自文本索引的 NC 零件程序行	
0x00000012	读取	解释器	{				
			UINT32	1		当前显示为 1: SAF 2: 解释器 3: 误差偏移	
			UINT32	1		文件偏移	
			UINT8[260]	1		路径 + 程序名称	
			}				
0x00000013	读取	解释器	UINT32[18]			显示当前有效的 G 代码	
0x00000014	读取	解释器	{			确定当前有效的零点漂移	
			UINT32	1		块计数器	
			UINT32			虚拟	
			LREAL[3]	1		零点漂移 G54...G57	
			LREAL[3]	1		零点漂移 G58	
			LREAL[3]	1		零点漂移 G59	
			}				
0x00000015	读取	解释器	{			确定当前有效的旋转	
			UINT32	1		块计数器	
			UINT32	1		虚拟	
			LREAL[3]	1		X、Y 和 Z 的旋转 (以度数为单位)	
			}				
0x00000016	读取	解释器	UINT32	1	[0,1]	进给器信息	仅在内部使用! 不标准
0x00000100	读取	每个	UINT32 [n]	1	[0, 1...255]	在通道编号中返回相应的轴 ID: [1...255] 轴 ID: [0, 1...255]	无法用示波器追踪!

6.4.2.3 通道功能的"索引偏移"规范 (索引组 0x2200 + ID)

索引偏移 (十 六进制)	访问	通道类型	数据类型	物理单位	定义范围	描述	备注
0x00000001	写入	每个	UINT32	1		用程序编号加载 NC 程序	
0x00000002	写入	每个	VOID			启动解释器	
0x00000003	写入	每个	VOID			保留	
0x00000004	写入	每个	UINT8[...]			按名称加载 NC 程序。尽 管可以，但标准的NC路径 不是必要提供的。也允 许其他路径。	
0x00000005	写入	每个	UINT16	枚举	参见解释器运行 模式附录 [▶ 101]	设置解释器/通道运行模式	
0x00000006	写入	解释器	UINT8[...]			设置子程序路径	
0x00000008	写入	解释器	UINT32	1		解释器模拟模式： 0: 关闭 (默认) 1: 打开	尚未发布
0x0000000F	写入	每个	VOID			保留	
0x00000010	写入	每个	VOID			"重置"通道	
0x00000011	写入	每个	VOID			"停止" 通道	
0x00000012	写入	每个	VOID			"重试"通道 (重新启动通 道)	
0x00000013	写入	每个	VOID			"跳过"通道 (跳过任务/ 块)	
0x00000014/0 x00000015	写入	每个	{ UINT32 1 >0 UINT32 1 ≥ 0 REAL64[3] mm ±∞ REAL64[5] mm ±∞ }			"启用回描"/"禁用回描" 进给器方向： 1: 向前 2: 向后 条目索引 主轴 X、Y、Z 的位置 辅助轴 Q1、...、Q5 的位 置	保留功能， 无标准！
0x00000020	写入	每个	VOID			"保存"零点偏移量 (NPV)	
0x00000021	写入	每个	VOID			"加载"零点偏移量(NPV)	
0x00000022	写入	每个	VOID			保存" 工具补偿	
0x00000023	写入	每个	VOID			"加载" 工具补偿	
0x00000024	写入	插补	{ char[32] UINT32 1 0..1 }			在给定文件中保存解释器 快照 TwinCAT\CNC 文件夹中 的文件名 掩码： 0x1: R-Parameters 0x2: Zeroshifts 0x4: Tool Desc	
0x00000025	写入	插补	{ char[32] UINT32 1 0..1 }			将给定文件快照读入解释 器 TwinCAT\CNC 文件夹中 的文件名 掩码： 0x1: R-Parameters 0x2: Zeroshifts 0x4: Tool Desc	

索引偏移 (十 六进制)	访问	通道类型	数据类型	物理单位	定义范围	描述	备注
0x000000026	写入	插补	VOID			将所有工具参数 (包括类 型和编号) 设置为空	
0x000000027	写入	插补	VOID			将所有零点偏移 量设置为 空	
0x000000030	写入	每个	VOID			编程 解释器停止后重新启 动 (继续执行程序) 解释 器	
0x000000040	写入	每个	VOID			删除 NCI 中任何剩余行程 的触发器事件	
0x000000041	写入	每个				为 特定事件保留	
0x000000050	写入	插补	VOID	1		在解释器中设置 <i>ExecIdleInfo</i>	保留功能, 无标准!
0x000000051	写入	插补	UINT32	1		在解释器参数中设置块跳 过掩码: <i>SkippingMask</i>	保留功能, 无标准!
0x000000052	写入	内插法	UINT32	1		在解释器参数中设置 <i>ItpOperationMode: OperationMode mask</i>	保留功能, 无标准!
0x000000053	写入	插补	VOID			在 NC 设备中设置 <i>ScanningFlag</i>	保留功能, 无标准!
0x000000054	写入	插补				扫描位置	保留功能, 无标准!
			double[8]			位置	
0x000000055	写入	插补				预留	
0x000000056	写入	插补	VOID			在中止状态中设置解释器	保留功能, 无标准!
0x000000060	写入	插值	UINT16	1	0..159	手动重置快速 M 函数	

6.4.2.4 循环通道处理数据的"索引偏移"规范 (索引组 0x2300 + ID)

索引偏移 (十 六进制)	访问	通道类型	数据类型	物理单位	定义范围	描述	备注
0x000000000	读取	每个 (PLC→NC)	{128 字节}		STRUCT 参见 通道接口	通道结构 (PLC→NC) 备注: 大小和对齐方式已 更改。	当前 PLC 结 构: <i>NciChannelFro mPlc plctonc_ncich annel_ref</i>
0x000000001	读取	每个	UINT8[...] 最少 30 字节	1		解释器程序显示	无法用示波器追 踪!
0x000000002	读/写	每个 (PLC→NC)	UINT32	%	[0...1000000]	速度超驰通道 (通道中的 轴)	1000000 = 100%
0x000000003	读/写	每个 (PLC→NC)	UINT32	%	[0...1000000]	速度超驰主轴	1000000 = 100%
0x000000080	读取	每次 (NC→PLC)	{160 字节}		STRUCT 参见 通道接口	通道结构 (NC→PLC) 备注: 大小和对齐方式已 更改。	当前 PLC 结 构: <i>NciChannelTo Plc nctoplc_ncich annel_ref</i>
0x100000000 +RegIndex	读/写	每个	REAL64	1	[0...999]	解释器的 R 参数	无法用示波器追 踪!
0x200000001	读取	每个	UINT8[...] 最少 30 字节	1	[1...9]	组警示处理 (SAF) 的程序 显示	无法用示波器追 踪!

6.4.3 规范组

6.4.3.1 组参数的"索引偏移"规范 (索引组 0x3000 + ID)

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
0x00000001	读取	每个	UINT32	1		组 ID	
0x00000002	读取	每个	UINT8[30+1]	1		组名称	
0x00000003	读取	每个	UINT32	1	枚举	组类型 [▶ 102]	
0x00000004	读取	每个	UINT32	μs		SAF 循环时间组	
0x00000005	读取	每个	UINT32	μs		SVB 循环时间组	
0x00000006	读/写	每个	UINT16	1	0/1	单块运行模式?	
0x0000000B	读取	每个	UINT32	1		SVB 表的大小 (SVB 条目的最大数量)	
0x0000000C	读取	每个	UINT32	1		SAF 表的大小 (SAF 条目的最大数量)	
0x00000010	读/写	每个	UINT32	1	[1,2...32] 默认: 1	内部 SAF 循环时间除数 (将内部 SAF 循环时间除以该系数)	例如, DXD 组
0x00000021	读取	通道: 每个	UINT32	1		通道 ID	
0x00000022	读取	通道: 每个	UINT8[30+1]	1		通道名称	
0x00000023	读取	通道: 每个	UINT32	1	枚举	通道类型 [▶ 101]	
0x00000024	读取	通道: 每个	UINT32	1	>0	通道中的数字	
0x00000500	读/写	DXD 组	INT32	枚举	[0, 1]	转向速度减小法 [▶ 102] 0: 库仑散射 1: 余弦定律 2: 速度跳变	
0x00000501	读/写	DXD 组	REAL64	1	[0.0...1.0]	速度减小系数 C0 转换 (连续, 但既不是连续可微一次, 也不是连续可微两次)	
0x00000502	读/写	DXD 组	REAL64	1	[0.0...1.0]	速度减小系数 C1 转换 (连续和连续可微一次)	
0x00000503	读/写	DXD 组	REAL64	程度	[0.0...180.0]	分段转换临界角 "低" (必须严格小于或等于减速角 C0)	
0x00000504	读/写	DXD 组	REAL64	程度	[0.0...180.0]	分段转换临界角 "高" (必须严格小于或等于减速角 C0)	
0x00000505	读/写	DXD 组	REAL64	mm/s	≥ 0	最小速度, 在分段转换时, 即使速度可能降低, 也不能低于最小速度。	注意: 参数不会保存在解决方案中, 也不会作为 NC 启动参数传输!
0x00000506	读/写	DXD 组	REAL64	例如 mm	[0.0...1000.0]	用于混合的公差球面半径	未执行!
0x00000507	读/写	DXD 组	REAL64	1		速度减小系数 C2 转换	
0x00000508	读/写	DXD 组	UINT16	1	0/1	可计算总剩余路径长度	从 TC3.1 B4020.40 起更新
0x00000509	读/写	DXD 组	UINT16	1	0/1 默认: 1	主轴 (X、Y、Z) 软件限制位置监测的一般激活 (请参见编码器参数)	

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
0x00000050A	读/写	DXD 组	UINT32	1	0/1	NCI 速率比类型 0: 与内部降速有关 (无迭代) 1: 与原始外部 (编程) 速度有关 2: 与内部降速有关 (0 ... >100%)	
0x00000050C	读取	DXD 组	UINT32	1	[128 ... 1024] 默认: 128	用户定义 NCI SAF 表条目最大数	从 TC3.1 B4014 起的启动参数有更新
0x000000510	读/写	DXD 组	REAL64	1	≥ 0	对于减小方法 VeloJump C0 转换的减小系数: X 轴	未执行!
0x000000511	读/写	DXD 组	REAL64	1	≥ 0	对于减小方法 VeloJump C0 转换的减小系数: Y 轴	未执行!
0x000000512	读/写	DXD 组	REAL64	1	≥ 0	对于减小方法 VeloJump C0 转换的减小系数: Z 轴	未执行!
0x000000513	读/写	DXD 组	LREAL64	1]0.0...1.0[辅助轴混合: 如果有效路径速度小于编程速度与此系数相乘的结果, 则插入一个准停并删除公差球	尚未发布
0x000000514	读/写	DXD 组	UINT32	1	[1 … 20] 默认: 1	每个 NC 循环 (从 SVB 到 SAF) 传送作业的最大数	从 TC3.1 B4020.40 起更新
0x000000604	读/写	编码器组	REAL64	例如 mm/s	[0.0...1000.0]	速度窗口或静止窗口	基本单位 / s
0x000000605	读/写	编码器组	REAL64	s	[0.0...60.0]	静止窗口的滤波时间 (以秒为单位)	
0x000000606	读/写	编码器组	REAL64	s	[0.0...60.0]	死区时间补偿主轴/从轴耦合 ("角度预控制")	
0x000000701	读取	FIFO 组	UINT32	1	[1…16]	FIFO 尺寸 (m = 轴数) 注: FIFO 尺寸增至 16。	(n x m) FIFO 启动数据
0x000000702	读取	FIFO 组	UINT32	1	[1…10000]	FIFO 大小 (长度) (n = FIFO 条目数)	(n x m) FIFO 启动数据
0x000000703	读取	FIFO 组	UINT32	1	[0, 1, 4]	FIFO 设定点生成器的插值类型 0: INTERPOLATIONTYPE_LINEAR (默认) 1: INTERPOLATIONTYPE_4POINT 4: INTERPOLATIONTYPE_CUBICSPLINE (带 6 个点)	从 TC3.1 B4020 起更新
0x000000704	读/写	FIFO 组	UINT32	1	[1, 2]	FIFO 设定点生成器的速率比类型	

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
						类型 1: OVERRIDETYPE_INSTANTANEOUS (默认) 类型 2: OVERRIDETYPE_PT2	
0x000000705	读/写	FIFO 组	REAL64	s	> 0.0	超驰变化的 P-T2 时间 (T1=T2=T0)	
0x000000706	读/写	FIFO 组	REAL64	s	≥ 0.0	两个有序 FIFO 条目的时间差值 (FIFO 条目时基)	
0x000000801	读写	运动学组	写入			计算位置的运动学正向转换 (ACS -> MCS)	
			{				
			REAL64[8]	例如度数	±∞	ACS (轴坐标系) 轴位置, 最大尺寸: 8	
			UINT32	1	≥ 0	保留	
			UINT32	1	≥ 0	保留	
			}				
			读取				
			{				
			REAL64[8]	例如 mm	±∞	MCS (机器坐标系) 轴位置, 最大尺寸: 8	
			UINT32	1	≥ 0	保留	
			UINT32	1	≥ 0	保留	
			}				
0x000000802	读写	运动学组	写入			计算位置的运动学逆向转换 (MCS -> ACS)	
			{				
			REAL64[8]	例如 mm	±∞	MCS (机器坐标系) 轴位置, 最大尺寸: 8	
			UINT32	1	≥ 0	保留	
			UINT32	1	≥ 0	保留	
			}				
			读取				
			{				
			REAL64[8]	例如度数	±∞	ACS (轴坐标系) 轴位置, 最大尺寸: 8	
			UINT32	1	≥ 0	保留	
			UINT32	1	≥ 0	保留	
			}				

6.4.3.2 组状态的"索引偏移"规范 (索引组 0x3100 + ID)

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
0x000000001	读取	每个	INT32	1	枚举	错误代码组	
0x000000002	读取	每个	UINT32	1		主轴数	
0x000000003	读取	每个	UINT32	1		从轴数	
0x000000004	读取	每个	UINT32	1	参见 枚举	SVB 组状态 (状态)	
0x000000005	读取	每个	UINT32	1	参见 枚举	SAF 组状态 (主状态)	
0x000000006	读取	每个	UINT32	1	参见 枚举	运动状态 (状态)	
0x000000007	读取	每个	UINT32	1	参见 枚举	SAF 子组状态 (子状态)	
0x000000008	读取	每个	UINT32	1	参见 枚举	引用状态 (状态)	
0x000000009	读取	每个	UINT32	1	参见 枚举	耦合状态 (状态)	无法用示波器追踪!

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
0x00000000A	读取	每个	UINT32	1	≥ 0	耦合表索引	无法用示波器追踪!
0x00000000B	读取	每个	UINT32	1	≥ 0	当前 SVB 条目/任务数	符号访问： 'SvbEntries' (DXD)
0x00000000C	读取	每个	UINT32	1	≥ 0	当前 SAF 条目/任务数	符号访问： 'SafEntries' (DXD)
0x00000000D	读取	每个	UINT32	1		当前块编号 (仅对插值组有效)	符号访问： 'BlockNumber' (DXD)
0x00000000E	读取	每个	UINT32	1	≥ 0	当前自由 SVB 条目/任务数	无法用示波器追踪!
0x00000000F	读取	每个	UINT32	1	≥ 0	当前自由 SAF 条目/任务数	无法用示波器追踪!
0x000000011	读取	每个	UINT16	1	0/1	紧急停止 (E-Stop) 激活?	无法用示波器追踪!
0x00000110	读取	PTP 组	{			内部 NC 信息 (分辨率)	保留!
			REAL64	例如 mm	$\pm \infty$	外部结束位置	
			REAL64	例如 mm/s	>0	外部目标速度	
			REAL64	例如 mm/s^2	>0	外部加速度	
			REAL64	例如 mm/s^2	>0	外部减速度	
			REAL64	例如 mm/s^3	>0	外部加加速度	
			UINT32	1	>0	外部速率比类型	
			REAL64	例如 mm	$\pm \infty$	内部结束位置	
			REAL64	例如 mm/s	>0	内部目标速度 (指的是 100%)	
			REAL64	%	[0 ... 100]	内部实际速率比	
			REAL64	例如 mm/s^2	>0	内部加速度	
			REAL64	例如 mm/s^2	>0	内部减速度	
			REAL64	例如 mm/s^3	>0	内部加加速度	
			REAL64	例如 mm	>0	位置分辨率	
			REAL64	例如 mm/s	≥ 0	速度分辨率	
			REAL64	例如 mm/s^2	≥ 0	加速度分辨率	
			REAL64	例如 mm/s	≥ 0	加速度为零时的速度分辨率	
			}				
0x00000500	读取	DXD 组	REAL64	例如 mm	≥ 0	当前路径分段上的剩余路径 (剩余弧长)	符号访问： 'SetPathRemLength'
0x00000501	读取	DXD 组	REAL64	例如 mm	≥ 0	当前路径分段上的累积弧长	符号访问： 'SetPathLength'
0x00000502	读取	DXD 组	REAL64	例如 mm/s	≥ 0	当前路径设置速度	符号访问： 'SetPathVelo'
0x00000503	读取	DXD 组	REAL64	例如 mm/s^2	$\pm \infty$	当前路径设置加速度	符号访问： 'SetPathAcc'

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
0x00000504	读取	DXD 组	REAL64	例如 mm/s^2	≥ 0	当前矢量集加速度的量	符号访问: 'SetPathAbsAcc'
0x00000505	读取	DXD 组	REAL64	例如 mm/s	≥ 0	最大分段结束路径设置速度	符号访问: 'SetPathVeloEnd'
0x00000506	读取	DXD 组	REAL64	例如 mm/s	≥ 0	分段最大路径设置速度	符号访问: 'SetPathVeloMax'
0x00000507	读取	DXD 组	REAL64	例如 mm	≥ 0	基于当前弧长的当前相对制动距离	符号访问: 'SetPathStopDist'
0x00000508	读取	DXD 组	REAL64	例如 mm	± ∞	安全距离 = 分段弧长 - 当前弧长 - 相对制动距离	符号访问: 'SetPathSecurityDist'
0x00000509	读取	DXD 组	REAL64	1	0/1	分段转换	符号访问: 'SetPathSegmentChange'
0x0000050A	读取	DXD 组	REAL64	%	[0 ... 100]	路径速度速率比	符号访问: 'SetPathOverrule'
0x00000511	读取	DXD 组	REAL64	例如 mm/s	≥ 0	实际路径速度的分量	符号访问: 'ActPathAbsVel0'
0x00000512	读取	DXD 组	REAL64	例如 mm/s^2	± ∞	当前分段上的实际路径加速度	符号访问: 'ActPathAcc'
0x00000513	读取	DXD 组	REAL64	例如 mm/s^2	≥ 0	当前分段上实际路径加速度的分量	符号访问: 'ActPathAbsAcc'
0x00000514	读取	DXD 组	REAL64	例如 mm	± ∞	路径上切线方向的位置误差 (用符号表示领先和滞后)	符号访问: 'PathDiffTangential'
0x00000515	读取	DXD 组	REAL64	例如 mm	≥ 0	路径上正交方向的位置误差	符号访问: 'PathDiffOrthogonal'
0x00000520	读取	DXD 组	REAL64	1	≥ 0	当前分段的覆盖弧长, 规范化为 1.0。	
0x00000521	读取	DXD 组	REAL64	1	0/1	改变部分分段 (公差球半径)	
0x00000522	读取	DXD 组	REAL64	1	≥ 0	到上一个几何条目或下一个准停的剩余路径总长度。参见组参数 0x508。	
0x00000523	读取	DXD 组	REAL64	1	≥ 0	当前分段的编程速度	
0x00000524	读取	DXD 组	REAL64	例如 mm	≥ 0	自程序启动后行进的路径距离 (弧长)	TwinCAT 3.1 B4022.31 及以上 TwinCAT 3.1 B4024.0 及以上
0x00000530	读取	DXD 组	{ REAL64 REAL64 REAL64 }	例如 mm ± ∞ ± ∞ ± ∞		主轴 X、Y 和 Z 的当前或上一个 MCS 目标位置 目标位置 X 轴 目标位置 Y 轴 目标位置 Z 轴	
0x00000531	读取	DXD 组	{ REAL64[5] }	例如 mm ± ∞		辅助轴 Q1 至 Q5 的当前或上一个 MCS 目标位置 轴 Q1 至 Q5 的目标位置	

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
0x000000532	读取	DXD 组	{			读取与当前 DC 时间相关的下 11 个分段的路径长度、H 参数和条目 ID	未广泛发布
			UINT32			DC 时间	
			UINT32			保留	
			PreViewTab[11]			11*24 字节	
			}				
			PreViewTab				
			{				
			REAL64	例如 mm		分段长度	
			UINT32	1		块号	
			UINT32	1		H 参数	
			UINT32	1		条目 ID	
			UINT32	1		保留	
			}				
0x00000054n	读取	DXD 组	REAL64	1	0/1	在辅助轴公差球内 n = 1..5 辅助轴数 (非轴 ID)	
0x000000546	读取	DXD 组	REAL64[8]	例如 mm	±∞	设置 3D 组 (3+5) 轴的位置阵列	TC3.1 B4022.17 及以上
0x000000547	读取	DXD 组	REAL64[8]	例如 mm	±∞	3D 组 (3+5) 轴的实际位置阵列	TC3.1 B4022.17 及以上
0x000000548	读取	DXD 组	REAL64[8]	例如 mm	±∞	作为 3D 组 (3+5) 轴阵列的位置差 (设置值/实际值) 或滞后误差	TC3.1 B4022.17 及以上
0x000000550	读取	DXD 组	{			读取 3D 组内的轴 ID:	
			UINT32	1	[0, 1...255]	X 轴 ID	
			UINT32	1	[0, 1...255]	Y 轴 ID	
			UINT32	1	[0, 1...255]	Z 轴 ID	
			}				
0x000000552	读取	DXD 组 FIFO 组 运动学组	{ UINT32[m] }	1	[0, 1...255]	组的轴分配: 第 1 轴 ID - 第 m 轴 ID m: 带有主轴和辅助轴 (X、Y、Z、Q1、Q2、Q3、Q4、Q5) 或 FIFO 组或运动学组 ACS 轴的 3D 组的尺寸	
0x000000553	读取	运动学组	{			读取运动学组内的轴分配 (ID):	
			UINT32[8]	1	[0, 1...255]	MCS 轴 ID (机器坐标系)	
			UINT32[8]	1	[0, 1...255]	ACS 轴 ID (轴坐标系)	
			UINT32	1	≥ 0	保留	
			UINT32	1	≥ 0	保留 (新)	
			}				
0x00000056n	读取	DXD 组	REAL64	1	±∞	公差球内辅助轴的当前位置误差 (仅设置值侧) 仅适用于辅助轴 n = 1..5 辅助轴数 (非轴 ID)	

6.4.3.3 组功能的"索引偏移"规范 (索引组 0x3200 + ID)

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
0x00000001	写入	每个	VOID			重置组	
0x00000002	写入	每个	VOID			停止组	
0x00000003	写入	每个	VOID			清除组 (缓冲/任务)	
0x00000004	写入	PTP 组、3D 组	{ REAL64 REAL64 }			紧急停止 (E-stop) (通过控制斜坡紧急停止) 减速度 (必须大于或等于原始减速度) 加加速度 (必须大于或等于原始加加速度)	
0x00000005	写入	PTP 组	{ REAL64 REAL64 }			可参数化停止 (使用控制斜坡) 减速度 加加速度	保留功能，无标准！
0x00000006	写入	PTP 组、3D 组	VOID			紧急停止 (E-Stop) 后的"步进"	
0x00000050	写入	PTP 组、3D 组	{ UINT32 UINT32 UINT32 }			组的轴分配: X 轴 ID Y 轴 ID Z 轴 ID	
0x00000051	写入	PTP 组、3D 组 FIFO 组	{ UINT32 UINT32 }			组的轴分配: 轴 ID 轴在组 m 中的位置索引: 组尺寸 (PTP: 1; DXD: 3, FIFO: 16)	
0x00000052	写入	3D 组 FIFO 组	{ UINT32[m] }	1	[0, 1...255]	组的轴分配: 第一轴 ID、...、M 轴 ID m: 3D 组 (X、Y、Z、Q1、Q2、Q3、Q4、Q5) 或 FIFO 组的尺寸	
0x00000053	写入	3D 组 FIFO 组 运动学组	VOID			删除 3D 轴分配、FIFO 轴分配或运动学轴分配，并将轴返回到各自的 PTP 组	
0x00000054	写入	运动学组	{ UINT32[8] UINT32[8] UINT32 UINT32 }	1	[0, 1...255]	运动学组的轴分配: MCS 轴 ID (机器坐标系) ACS 轴 ID (轴坐标系) 保留 保留 (新)	
0x00000060	读写	3D 组		1		内部"进给组"命令 ("进给器")	执行命令！
0x00000061	读写	3D 组		1		内部"进给组"命令 ("进给器")	执行命令！

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
0x000000110	写入	1D 组	VOID			参见 1D 组 (“校准”)	
0x000000111	写入	1D 组	{			新结束位置 1D 组	使用时请注意! 始终连接至 SAF 501 端口!
			UINT32	枚举	参见附录	结束位置类型 [▶ 103] (参见附录)	
			REAL64	例如 mm	±∞	新结束位置 (目标位置)	
			}				
0x00000011A	写入	1D 组	{			设置实际位置 1D 组	使用时请注意! 始终连接至 SAF 501 端口!
			UINT32	枚举	参见附录	实际位置类型 [▶ 103] (参见附录)	
			REAL64	例如 mm	±∞	轴的实际位置	
			}				
0x00000011B	写入	1D 组	UINT32	1	0/1	设置参考旗标 (“校准旗标”)	使用时请注意!
0x000000120	写入	1D 组	{			启动 1D 组 (标准启动) :	使用时请注意! 始终连接至 SAF 501 端口!
			UINT32	枚举	参见附录	启动类型 [▶ 102] (参见附录)	
			REAL64	例如 mm	±∞	结束位置 (目标位置)	
			REAL64	mm/s	≥ 0.0	所需速度	
			}				
0x000000121	写入	1D 组 (SERVO)	{			启动 1D 组 (扩展启动) :	使用时请注意! 始终连接至 SAF 501 端口!
			UINT32	枚举	参见附录	启动类型 [▶ 102] (参见附录)	
			REAL64	例如 mm	±∞	结束位置 (目标位置)	
			REAL64	mm/s	≥ 0.0	所需速度	
			UINT32	1	0/1	标准加速度?	
			REAL64	mm/s^2	≥ 0.0	加速度	
			UINT32	1	0/1	标准减速度?	
			REAL64	mm/s^2	≥ 0.0	减速度	
			UINT32	1	0/1	标准加加速度?	
			REAL64	mm/s^3	≥ 0.0	加加速度	
			}				
0x000000122	写入	1D 组 (MW 伺服)	{			启动 1D 组 (特殊启动) :	保留启动功能，无标准!
			UINT32	枚举	参见附录	启动类型 [▶ 102] (参见附录)	
			REAL64	例如 mm	±∞	结束位置 (目标位置)	
			REAL64	mm/s	≥ 0.0	所需启动速度	
			REAL64	例如 mm	±∞	新速度等级的位置	
			REAL64	mm/s	≥ 0.0	新结束速度级	
			UINT32	1	0/1	标准加速度?	
			REAL64	mm/s^2	≥ 0.0	加速度	
			UINT32	1	0/1	标准减速度?	
			REAL64	mm/s^2	≥ 0.0	减速度	
			UINT32	1	0/1	标准加加速度?	
			REAL64	mm/s^3	≥ 0.0	加加速度	
			}				
0x000000126	写入	1D 组	{			启动驱动输出:	使用时请注意! 始终连接至 SAF 501 端口!
			UINT32	枚举	参见附录	输出类型 [▶ 110] (参见附录)	
			REAL64	例如 %	±∞	所需输出值 (例如 %)	
			}				
0x000000127	写入	1D 组	VOID			停止驱动输出	

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
0x000000128	写入	1D 组	{ UINT32 REAL64 }	枚举	参见附录	更改驱动输出： 输出类型 [▶ 110] (参见附录)	
				例如 %	±∞	所需输出值 (例如 %)	
0x000000130	写入	1D 组 (SERVO)	{ UINT32 REAL64 REAL64 REAL64 REAL64 REAL64 REAL64 }	枚举	参见附录	1D 区间补偿 (伺服)： 补偿类型 [▶ 104] (附录)	
				mm/s/s	≥ 0.0	最大加速度增加	
				mm/s/s	≥ 0.0	最大减速度增加	
				mm/s	≥ 0.0	最大增速	
				mm/s	≥ 0.0	进程的基本速度	
				例如 mm	±∞	需要补偿的路径差	
				例如 mm	≥ 0.0	补偿的路径距离	
0x000000131	写入	1D 组 SERVO	VOID			停止区间补偿 (伺服)	
0x000000140 (0x00n00140)	写入	主轴/从轴耦合: 1D 组 (伺服)	{ UINT32 UINT32 UINT32 UINT32 REAL64 REAL64 REAL64 REAL64 }	枚举	参见附录	主轴/从轴耦合 (伺服)： 从轴类型/耦合类型 [▶ 104] (参见附录)	扩展功能：“飞锯”功能！锯切角度大于0.0度且小于或等于90.0度 (平行锯切: 90.0度)
				1	[1...255]	主轴/组的轴 ID	
				1	[0...8]	主轴的子索引 n (默认值: 0)	
				1	[0...8]	从轴的子索引 n (默认: 0)	
						参数 1: 线性: 齿轮系数 FlySawVelo: 保留 FlySaw: 绝对同步位置主轴 [mm]	
						参数 2: 线性: 保留 FlySawVelo: 保留 FlySawPos: 绝对同步位置从轴 [mm]	
						参数 3: 线性: 保留 FlySawVelo: 倾角, 以 [DEGREE] 为单位 FlySawPos: 倾角, 以 [DEGREE] 为单位	
						参数 4: 线性: 保留 FlySawVelo: 齿轮系数 FlySawPos: 齿轮系数	
						}	
0x000000141	写入	主轴/从轴耦合: 1D 组 (伺服)	VOID			主轴/从轴解耦 (伺服)	
0x000000142	写入	主轴/从轴参数 1D 组 (伺服)	{ REAL64 REAL64 REAL64 }	1	[±1000000.0]	更改耦合参数 (耦合)： 参数 1: 线性: 齿轮系数	
				1	[±1000000.0]	参数 2: 线性: 保留	
				1	[±1000000.0]	参数 3: 线性: 保留	
				1	[±1000000.0]	参数 4: 线性: 保留	

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
			}				
0x00000144	写入	从轴停止 1D 组 (伺服)	VOID			停止"飞锯" (伺服)	仅适用于"飞锯"
0x00000149	写入	从表 1D 组 (伺服)	REAL64	1	±∞	设置单表耦合的从表缩放比例 (伺服)	仅适用于单表从轴
0x00000150	写入	1D 组	VOID			停用完整的 1D 组/轴 (禁用)	
0x00000151	写入	1D 组	VOID			激活完整的 1D 组/轴 (启用)	
0x00000160	写入	1D 组	VOID			停用 1D 组的驱动输出 (禁用)	
0x00000161	写入	1D 组	VOID			激活 1D 组的驱动输出 (启用)	
0x00000362	写入	高速/低速组	UINT16	1	0/1	释放驻车制动? 0: 自动激活 (默认) 1: 必须始终释放!	
0x00000701	写入	FIFO 组	VOID			启动 FIFO 组 (FIFO 表 (n*m)-FIFO 必须事先填好)	
0x00000710	写入	FIFO 组	{ REAL64[x*m]}	例如 mm	±∞	写入 x 个 FIFO 条目 (行) : (x*m) 值 (一行或多行) n: FIFO 长度 (行数) m: FIFO 尺寸 (列数) 值范围 x: [1 ... n]	只能逐行进行! (整数倍)
0x00000711	写入	FIFO 组	{ REAL64[x*m]}	例如 mm	±∞	重写最后 x 个 FIFO 条目 (行) : (x*m) 值 (一行或多行) n: FIFO 长度 (行数) m: FIFO 尺寸 (列数) 值范围 x: [1 ... n]	只能逐行进行! (整数倍)
0x00000801	写入	运动学组	VOID			启动运动学组	保留功能, 无标准!

6.4.4 规范轴

6.4.4.1 轴参数的"索引偏移"规范 (索引组 0x4000 + ID)

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n00000	读取	每个 (所有轴参数的结构)	{			一般轴参数结构 (NC/CNC), 还包含编码器、控制器和驱动器等子元素 (参见 TcMc2.lib 中的 MC_ReadParameters 等)	从 TC3 起修改

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
						注: 尺寸和对齐方式有所改变。	
			UINT32	1		轴 ID	
			STRING[30+1]	1		轴名称	
			UINT32	1	枚举	轴类型 [▶ 102]	
			
			}			1024 字节 (而不是 512 字节)	
0x00000001	读取	每个	UINT32	1		轴 ID	
0x00000002	读取	每个	STRING[30+1] UINT8[...]	1		轴名称	任意字符数, 从 TC3.1 版本 4022.32 或 4024.6 起
0x00000003	读取	每个	UINT32	1	枚举	轴类型 [▶ 102]	
0x00000004	读取	每个	UINT32	μs		循环时间轴 (SEC)	
0x00000005	读取	每个	STRING[10+1]	1		物理单位	
0x00000006	读/写	每个	REAL64	例如 mm/s		凸轮方向的参考速度	
0x00000007	读/写	每个	REAL64	例如 mm/s		同步方向的参考速度	
0x00000008	读/写	每个	REAL64	例如 mm/s		速度手动慢速	
0x00000009	读/写	每个	REAL64	例如 mm/s		速度手动快速	
0x0000000A	读/写	每个	REAL64	例如 mm/s	[0.0...1.0E20]	速度快速移位	
0x0000000F	读/写	每个	UINT16	1	0/1	位置范围监测?	
0x00000010	读/写	每个	REAL64	例如 mm	[0.0...1.0E6]	位置范围窗口	
0x00000011	读/写	每个	UINT16	1	0/1	运动监测?	
0x00000012	读/写	每个	REAL64	s	[0.0...600]	运动监测时间	
0x00000013	读/写	每个	UINT16	1	0/1	循环?	
0x00000014	读/写	每个	REAL64	例如 mm		循环距离 (±)	
0x00000015	读/写	每个	UINT16	1	0/1	目标位置监测?	
0x00000016	读/写	每个	REAL64	例如 mm	[0.0...1.0E6]	目标位置窗口	
0x00000017	读/写	每个	REAL64	s	[0.0...600]	目标位置监测时间	
0x00000018	读/写	每个	REAL64	例如 mm		正方向脉冲路径	
0x00000019	读/写	每个	REAL64	例如 mm		负方向脉冲路径	
0x0000001A	读/写	每个	UINT32	1	枚举 (≥ 0)	错误反应模式: 0: 瞬时 (默认) 1: 延迟 (例如 主轴/从轴耦合)	
0x0000001B	读/写	每个	REAL64	s	[0...1000]	错误延迟时间 (如果选择延迟错误反应)	
0x0000001C	读/写	每个	UINT16	1	0/1	如果设备尚未准备好运行, 是否通过实际值耦合从轴?	
0x0000001D	读/写	每个	REAL64	例如 mm/s^2	[0, 0.01...1.0E10]	从设置值切换到实际值时的衰减剖面加速度: 默认: 0 (在这种情况下使用轴加速度的最小值, 即 MIN(Acc, Dec))	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00000001E	读/写	每个	UINT32	1	枚举 (≥ 0)	快速轴停止信号类型： 选择触发快速轴停止的 信号类型 (请参见 Drive->nStatus4 中的 第 7 位) "0 (SignalType_OFF) " ' " "1 (SignalType_RisingEdge) "、"2 (SignalType_FallingEdge) "、"3 (SignalType_BothEdges) "、"4 (SignalType_HighActive) "、"5 (SignalType_LowActive) "	
0x000000020	读/写	每个	UINT16	1	0/1	是否允许对从轴发出运动命令? 默认: FALSE	
0x000000021	读/写	每个	UINT16	1	0/1	是否允许对具有激活外部设定点发生器的轴发出运动命令? 默认: FALSE	
0x000000026	读/写	每个	UINT32	1		单位 (位置、速度、时间) 的解释 第 0 位: 速度单位为 x/ min, 而非 x/s 第 1 位: 以千分之一为 基本单位的位置 第 2 位: 模除位置显示	参见编码器! 位 数组
0x000000027	读/写	每个	REAL64	例如 mm/s	[>0...1.0E20]	最大允许速度	
0x000000028	读/写	每个	REAL64	例如 mm	[0.0...1.0E6]	运动监测窗口	
0x000000029	读/写	每个	UINT16	1	0/1	PEH 时间监测?	位置结束和准停
0x00000002A	读/写	每个	REAL64	s	[0.0...600]	PEH 监测时间	
0x00000002C	读/写	每个	REAL64	例如 mm	[-1000.0 ...1000.0]	间隙	
0x000000030	读取	每个	UINT16	1	[0,1]	持久性数据 (PERSISTENT DATA), 例如用于编 码器的实际位置和参考 状态?	启动参数, 无 法 在线更改。
0x000000031	读取	每个	{ uint8[6] uint16 uint16 }10 字节	AmsAdd r: AmsNetI d, AmsPort No. Channel No	1	读取硬件 AMS 地址 (AMS 网络 ID 和 AMS 端口号) 和 EtherCAT 通道号 (通信通道 0、 1、2、3...)	
0x000000031	读取	每个	{ uint8[6] uint16 uint16 // uint16 uint32	AmsAdd r: AmsNetI d, AmsPort No. Channel	1	读取硬件 AMS 地址 (AMS 网络 ID 和设备 AMS 端口号) 和 EtherCAT 通道号 (通 信通道 0、1、2、 3...)	从 TC3 起更新 DriveObjectId 和 EncObjectId, 从 NC build 4437 起

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			uint32 uint32 uint32 uint32 uint32 uint32 uint32 uint32 uint32 uint32[3] }64 字节	No Reserve rt NcDrive lD NcDrive lndex NcDrive Type NcEncID NcEncIn dex NcEncT ype NcAxisl D NcAxisT ype TcDrive ObjectId TcEncO bjectId 保留		由其他 NC 信息补充，如 NcDriveID、NcDriveType（参见附录）等……	
0x00000033	读取	每个	{ UINT16 ApplRequestBit UINT16 ApplRequestType UINT32 ApplCmdNo UINT32 ApplCmdVersion 。。 } 1024 字节	1 1 未执行 1	0/1 ≥0 >0 ≥0	一般应用请求结构 (NC/NCI)，例如用于 ApplicationHoming 请求 (请参见 <i>TcMc2.lib</i> 中的 <i>MC_ReadApplication Request</i>) 应用请求类型： 0：无 (空闲) 1：归位	TC3 中的更改
0x00000051	读取	通道：每个	UINT32			通道 ID	
0x00000052	读取	通道：每个	STRING[30+1]			通道名称	
0x00000053	读取	通道：每个	UINT32	1	枚举	通道类型 [▶ 101]	
0x00000054	读取	组：每个	UINT32			组 ID	
0x00000055	读取	组：每个	STRING[30+1]			组名称	
0x00000056	读取	组：每个	UINT32	1	枚举	组类型 [▶ 102]	
0x00000057	读取	每个	UINT32			编码器数	
0x00000058	读取	每个	UINT32			控制器数	
0x00000059	读取	每个	UINT32			驱动器数	
0x0000005A	读取	每个	{ UINT32[9] UINT32[9] UINT32[9] } 108 字节	1 1 1	[0, 1...255] [0, 1...255] [0, 1...255]	读取轴的所有子元素： 轴编码器 ID 轴控制器 ID 轴驱动器 ID	
0x000000F1	读/写	每个	REAL64	例如 mm/s^2	默认：1.0E5	最大允许加速度	从 TC 3.2 起更新
0x000000F2	读/写	每个	REAL64	例如 mm/s^2	默认：1.0E6	最大允许减速度	从 TC 3.2 起更新
0x00000101	读/写	伺服	REAL64	例如 mm/s^2	[0.01...1.0E20]	加速度 (默认数据集)	
0x00000102	读/写	伺服	REAL64	例如 mm/s^2	[0.01...1.0E20]	减速度 (默认数据集)	
0x00000103	读/写	伺服	REAL64	例如 mm/s^3	[0.1...1.0E30]	加速度 (默认数据集)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00000104	读/写	伺服	REAL64	s	[0.0 ... 1.0] 默认: 0.0 s	设定点发生器的速度值和位置值之间的减速度时间 (以秒为单位)	
0x00000105	读/写	伺服	UINT32	1	枚举 默认: 类型 1	速度的速率比类型 [► 102]: 1: 与内部降速有关 (无迭代) 2: 与原始外部起始速度有关 (无迭代) 3: 与内部降速有关 (通过迭代方式进行优化) 4: 与原始外部起始速度有关 (通过迭代方式进行优化)	
0x00000106	读/写	伺服	REAL64	1	[0.0 ... 1.0E6] 默认: 0.0	动态减速速度的最大允许步长变化 $DV = \text{系数}^* \text{最小值 } (A+, A-)^* DT$	
0x00000107	读/写	伺服	UINT16	1	[0.1] 默认: 1	激活辅助轴的加速度和加加速度限制 (Q1 至 Q5)	
	读/写	伺服	REAL64	例如 mm	[0.0..1000.0]	辅助轴公差球面半径	
	读/写	伺服	REAL64	例如 mm	[0.0..10000.0]	减小公差球面时允许的最大位置偏差 仅适用于辅助轴	
0x0000010A	读/写	伺服	REAL64	例如 mm/s^2	[0.01 ... 1.0E20]	快速轴停止: 加速度 (另见 快速轴停止信号类型)	
0x0000010B	读/写	伺服	REAL64	例如 mm/s^2	[0.01 ... 1.0E20]	快速轴停止: 减速度 (另见 快速轴停止信号类型)	
0x0000010C	读/写	伺服	REAL64	例如 mm/s^3	[0.1 ... 1.0E30]	快速轴停止: 加加速度 (另见 快速轴停止信号类型)	
0x0000010D	读/写	伺服	UINT32	1		在循环接口中作为 "UserData" 传递的轴状态的索引偏移。 0x00000000: 停用 0x00010012: 带位置偏置电压的编码器位置 (无位置校正且无死区时间补偿) 0x00010014: DriveActVelo 0x00010017: MC_SetPosition 偏移	
0x00000201	读/写	步进电机	UINT32	1	枚举	运行模式步进电机	
0x00000202	读/写	步进电机	REAL64	例如 mm/ STEP	[1.0E-6 ... 1000.0]	电机步进的距离缩放	
0x00000203	读/写	步进电机	REAL64	例如 mm/s	[0.0 ... 1000.0]	速度剖面的最小速度	
0x00000204	读/写	步进电机	UINT32	1	[0 ... 100]	每个频率/速度阶跃的阶跃数	
0x00000205	读/写	步进电机	UINT32	1		作为同步脉冲的电机掩码	未执行!
0x00000301	读/写	高/低	REAL64	例如 mm	[0.0 ... 100000.0]	正方向爬行距离	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00000302	读/写	高/低	REAL64	例如 mm	[0.0 ... 100000.0]	负方向漂移距离	
0x00000303	读/写	高/低	REAL64	例如 mm	[0.0 ... 100000.0]	正方向制动距离	
0x00000304	读/写	高/低	REAL64	例如 mm	[0.0 ... 100000.0]	负方向制动距离	
0x00000305	读/写	高/低	REAL64	s	[0.0 ... 60.0]	正方向制动减速度	
0x00000306	读/写	高/低	REAL64	s	[0.0 ... 60.0]	负方向制动减速度	
0x00000307	读/写	高/低	REAL64	s	[0.0 ... 60.0]	从高速到低速的切换时间	
0x00000308	读/写	高/低	REAL64	例如 mm	[0.0 ... 100000.0]	漂移距离停止	
0x00000309	读/写	高/低	REAL64	s	[0.0 ... 60.0]	松开制动器的延迟时间	
0x0000030A	读/写	高/低	REAL64	s	[0.0 ... 60.0]	正方向脉冲时间	
0x0000030B	读/写	高/低	REAL64	s	[0.0 ... 60.0]	负方向脉冲时间	
编码器							
0x00n10001	读取	编码器：每个	UINT32	1	[1 ... 255]	编码器 ID n = 0：轴的标准编码器 > 0：轴的第 n 个编码器（可选）	
0x00n10002	读取	编码器：每个	STRING[30+1]	1	30 个字符	编码器名称	
0x00n10003	读取	编码器：每个	UINT32	1	枚举 (>0)	编码器类型 [▶ 105]	
0x00n10004	读/写	编码器：每个	UINT32	1	Byteoffset	输入地址偏移 (I/O-输入-图像)	更改 I/O 地址
0x00n10005	读/写	编码器：每个	UINT32	1	Byteoffset	输出地址偏移 (I/O-输出-图像)	更改 I/O 地址
0x00n10006	读/写	编码器：每个	REAL64	例如 mm/INC	[1.0E-12 ... 1.0E+30]	结果缩放因子（分子/分母） 注：从 TC3 起，缩放因子（scaling factor）由分子和分母两部分组成（默认：1.0）。	如果已发出启用控制器的信号，则不允许写入。
0x00n10007	读/写	编码器：每个	REAL64	例如 mm	[±1.0E+9]	位置偏移	如果已发出启用控制器的信号，则不允许写入。
0x00n10008	读/写	编码器：每个	UINT16	1	[0,1]	编码器计数方向	如果已发出启用控制器的信号，则不允许写入。
0x00n10009	读/写	编码器：每个	REAL64	例如 mm	[0.001 ... 1.0E+9]	模除因子	
0x00n1000A	读/写	编码器：每个	UINT32	1	参见 枚举 (>0)	编码器模式 [▶ 106]	
0x00n1000B	读/写	编码器：每个	UINT16	1	0/1	软限位最小监测？	
0x00n1000C	读/写	编码器：每个	UINT16	1	0/1	软限位最大监测？	
0x00n1000D	读/写	编码器：每个	REAL64	mm		软限位最小结束位置	
0x00n1000E	读/写	编码器：每个	REAL64	mm		软限位最大结束位置	
0x00n1000F	读/写	编码器：每个	UINT32	1	参见 枚举 (>0) 在附录中	编码器评估方向 [▶ 106]（启用日志记录计数方向）	
0x00n10010	读/写	编码器：每个	REAL64	s	[0.0...60.0]	实际位置值的滤波时间（以秒为单位）(P-T1)	
0x00n10011	读/写	编码器：每个	REAL64	s	[0.0...60.0]	实际速度值的滤波时间（以秒为单位）(P-T1)	
0x00n10012	读/写	编码器：每个	REAL64	s	[0.0...60.0]	实际加速度值的滤波时间（以秒为单位）(P-T1)	
0x00n10013	读/写	编码器：每个	STRING[10+1]	1		物理单位	未执行！
0x00n10014	读/写	编码器：每个	UINT32	1		单位（位置、速度、时间）的解释	未执行！ 位数组

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
						第 0 位：速度单位为 x/min，而非 x/s 第 1 位：以千分之一为基本单位的位置	
0x00n10015	读取	编码器：每个	UINT32	INC	[0x0...0xFFFFFFFF]	编码器掩码（以增量表示的编码器实际值的最大值 注：编码器掩码可以是任何数值（如 3600000）。与过去不同，它不再需要对应于连续的一系列二进制 1(² n-1)。	只读参数 另请参见“编码器子掩码”参数
0x00n10016	读/写	编码器：每个	UINT16	1	0/1	实际位置校正（测量系统误差校正）？	
0x00n10017	读/写	编码器：每个	REAL64	s	[0.0...60.0]	实际位置校正的滤波时间（以秒为单位）(P-T1)	
0x00n10019	读/写	编码器：每个	UINT32	1	枚举 (>0)	编码器绝对尺寸系统 [▶ 107]	如果已发出启用控制器的信号，则不允许写入。
0x00n1001A	读取	编码器：每个	UINT32	1	枚举 (>0)	编码器位置初始化	未执行！
0x00n1001B	读/写	编码器：每个	REAL64	例如 mm	[>0, 模除因子/2]	模除启动的容差窗口	
0x00n1001C	读取	编码器：每个	UINT32	1	枚举 (>0)	编码器符号解释 [▶ 107] (数据类型)	
0x00n1001D	读取	编码器：每个	UINT16	1	0/1	增量或绝对编码器？ 0：增量编码器类型 1：绝对编码器类型	
0x00n10023	读/写	编码器：每个	REAL64	例如 mm/INC	[1.0E-12 ... 1.0E+30]	缩放因子(scaling factor)的组成部分：分子 (=> 缩放因子(scaling factor)分子/缩放(scaling factor)因子分子)	从 TC3 起更新 如果已发出启用控制器的信号，则不允许写入。
0x00n10024	读/写	编码器：每个	REAL64	1	[1.0E-12 ... 1.0E+30]	缩放因子(scaling factor)的组成部分：分母 (=> 缩放因子(scaling factor)分子/缩放(scaling factor)因子分母) 默认：1.0	从 TC3 起更新 如果已发出启用控制器的信号，则不允许写入。
0x00n10025	读/写	编码器：每个	{ real64 real64 }	例如 mm/INC 1	[1.0E-12 ... 1.0E+30] [1.0E-12 ... 1.0E+30]	缩放因子(scaling factor)的组成部分：分子 缩放因子(scaling factor)的组成部分：分母 (=> 缩放因子(scaling factor)分子/缩放因子(scaling factor)分母)	从 TC3 起更新
0x00n10030	读/写	编码器：每个	UINT32	1		内部编码器控制双字，用于指定运行模式和属性	从 TC3 起更新
0x00n10101	读/写	E: INC	UINT16	1	[0,1]	参考凸轮的反向搜索方向？	
0x00n10102	读/写	E: INC	UINT16	1	[0,1]	同步脉冲的反向搜索方向？	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n10103	读/写	E: INC	REAL64	例如 mm	[±1000000.0]	参考位置	
0x00n10104	读/写	E: INC	UINT16	1	[0,1]	是否激活了参考凸轮和同步脉冲之间的距离监测?	未执行!
0x00n10105	读/写	E: INC	UINT32	INC	[0 ... 65536]	参考凸轮与同步脉冲之间的最小间隙 (以增量为单位)	未执行!
0x00n10106	读/写	E: INC	UINT16	1	[0,1]	外部同步脉冲?	
0x00n10107	读/写	E: INC	UINT32	1	参见 枚举 (>0)	参考模式 [▶ 107]	
0x00n10108	读/写	E: INC	UINT32	1	[0x0000000F...0xFFFFFFFF]二进制掩码: (^n - 1)	编码器子掩码 (编码器实际值绝对范围的最大值, 以增量为单位) 例如, 用作参考模式"软件同步"和 NC 保留数据"绝对 (MODULO模除) "、"增量 (SINGLETURN ABSOLUTE单转绝对) "的参考标记。 注 1: 编码器子掩码必须小于或等于编码器掩码。 注 2: 编码器掩码必须是编码器子掩码的整数倍。 注 3: 编码器子掩码必须是连续的二进制 1 序列 (^n-1), 例如 0x000FFFFF。	另请参见"编码器掩码"参数
0x00n10110	读/写	E: INC (编码器模拟)	REAL64	1	[0.0 ... 1000000.0]	模拟编码器噪声部分的缩放/权重	
控制器							
0x00n20001	读取	控制器: 每个	UINT32	1	[1 ... 255]	控制器 ID n = 0: 轴的标准控制器 > 0: 轴的第 n 个控制器 (可选)	
0x00n20002	读取	控制器: 每个	STRING[30+1]	1	30 个字符	控制器名称	
0x00n20003	读取	控制器: 每个	UINT32	1	枚举 (>0)	控制器类型 [▶ 104]	
0x00n2000A	读/写	控制器: 每个		1	枚举 (>0)	控制器模式	
0x00n2000B	读/写	控制器: 每个	REAL64	%	[0.0 ... 1.0]	速度预控制的权重 (默认值: 1.0 = 100)	
0x00n20010	读/写	控制器: 每个	UINT16	1	0/1	位置滞后监测位置?	
0x00n20011	读/写	控制器: 每个	UINT16	1	0/1	位置滞后监测速度?	
0x00n20012	读/写	控制器: 每个	REAL64	例如 mm		最大滞后误差位置	
0x00n20013	读/写	控制器: 每个	REAL64	s		最大滞后误差滤波时间位置	
0x00n20014	读/写	控制器: 每个	REAL64	例如 mm/s		最大滞后误差速度	
0x00n20015	读/写	控制器: 每个	REAL64	s		最大滞后误差滤波时间速度	
0x00n20100	读/写	P/PID (位置、速度)	REAL64	1	[0.0...1.0]	控制器总输出的最大输出限制 (±)	(默认: 0.5 == 50%)

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n20102	读/写	P/PID (位置)	REAL64	例如 mm/s/mm	[0.0...1000.0]	比例增益 kp 或 kv 单位: 基本单位 / s / 基本单位	位置控制
0x00n20103	读/写	PID (位置)	REAL64	s	[0.0 ... 60.0]	积分作用时间 Tn	位置控制
0x00n20104	读/写	PID (位置)	REAL64	s	[0.0 ... 60.0]	微分作用时间 Tv	位置控制
0x00n20105	读/写	PID (位置)	REAL64	s	[0.0 ... 60.0]	阻尼时间 Td	位置控制
0x00n20106	读/写	PP (位置)	REAL64	例如 mm/s/mm	[0.0...1000.0]	超过限制速度时适用的额外比例增益, 分别为 kp 或 kv, 以百分比为单位。 单位: 基本单位 / s / 基本单位	位置控制
0x00n20107	读/写	PP (位置)	REAL64	%	[0.0...1.0]	阈值速度 (以百分比为单位), 超过该速度时, 适用额外的比例增益 (分别为 kp 或 kv)	
0x00n20108	读/写	P/PID (加速度)	REAL64	s	[0.0 ... 100.0]	比例增益 ka	加速度预控制
0x00n2010D	读/写	P/PID	REAL64	mm	[0.0 ... 10000.0]	位置误差的"死区" (控制偏差) (适用于带速度或扭矩接口的 P/PID 控制器)	保留功能
0x00n2010F	读/写	P/PP/PID (位置) 从轴控制	REAL64	(mm/s) / mm	[0.0...1000.0]	从轴耦合差异控制: 比例增益 k _{cp}	从轴耦合差异控制
0x00n20110	读/写	P (位置)	UINT16	1	0/1	自动偏移校准: 主动/被动	
0x00n20111	读/写	P (位置)	UINT16	1	0/1	自动偏移校准: 保持模式	
0x00n20112	读/写	P (位置)	UINT16	1	0/1	自动偏移校准: 衰减模式	
0x00n20114	读/写	P (位置)	REAL64	%	[0.0 ... 1.0]	自动偏移校准: 预控制限制	
0x00n20115	读/写	P (位置)	REAL64	s	[0.1 ... 60.0]	自动偏移校准: 时间常数	
0x00n20116	读/写	PID (位置)	REAL64	%	[0.0...1.0]	部分的最大输出限制(±), 以百分比为单位 (默认设置: 0.1 = 10%)	
0x00n20117	读/写	PID (位置)	REAL64	%	[0.0...1.0]	D 部分的最大输出限制(±), 以百分比为单位 (默认设置: 0.1 = 10%)	
0x00n20118	读/写	PID (位置)	UINT16	1	0/1	在主动定位过程中停用 部分 (假设 部分激活)? (默认设置: 0 = FALSE)	
0x00n20120	读/写	P/PID (位置)	REAL64	s	≥0	PT-1 位置误差滤波值 (位置控制偏差)	保留功能, 无标准!
0x00n20202	读/写	P/PID (速度)	REAL64	1	[0.0...1000.0]	比例增益 kp 或 kv	速度控制
0x00n20203	读/写	PID (速度)	REAL64	s	[0.0 ... 60.0]	积分作用时间 Tn	速度控制
0x00n20204	读/写	PID (速度)	REAL64	s	[0.0 ... 60.0]	微分作用时间 Tv	速度控制
0x00n20205	读/写	PID (速度)	REAL64	s	[0.0 ... 60.0]	阻尼时间 Td	速度控制
0x00n20206	读/写	PID (速度)	REAL64	%	[0.0...1.0]	部分的最大输出限制(±), 以百分比为单位 (默认设置: 0.1 = 10%)	速度控制

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n20207	读/写	PID (速度)	REAL64	%	[0.0...1.0]	D 部分的最大输出限制 (±), 以百分比为单位 (默认设置: 0.1 = 10%)	速度控制
0x00n2020D	读/写	P/PID (速度)	REAL64	mm/s	[0.0 ... 10000.0]	速度误差的"死区" (控制偏差) (适用于带速度或扭矩接口的 P/PID 控制器)	保留功能
0x00n20220	读/写	P/PID (速度)	REAL64	s	≥0	PT-2 速度误差滤波值 (速度控制偏差)	速度控制, 非标准!
0x00n20221	读/写	P/PID (速度)	REAL64	s	≥0	PT-1 速度误差滤波值 (速度控制偏差)	保留功能, 无标准!
0x00n20250	读/写	P/PI (观测器)	UINT32	1	枚举 (>0)	扭矩接口中控制器的观测器模式 [▶ 105] 0: 关闭 (默认) 1: 卢恩伯格 (LUENBERGER)	
0x00n20251	读/写	P/PI (观测器)	REAL64	Nm / A	>0.0	电机: 转矩常数 K_T	
0x00n20252	读/写	P/PI (观测器)	REAL64	kg m ²	>0.0	电机: 转动惯量 J_M	
0x00n20253	读/写	P/PI (观测器)	REAL64	Hz	[100.0 ... 2000.0] 默认: 500	带宽 f_0	
0x00n20254	读/写	P/PI (观测器)	REAL64	1	[0.0 ... 2.0] 默认: 1.0	校正系数 k_c	
0x00n20255	读/写	P/PI (观测器)	REAL64	s	[0.0 ... 0.01] 默认: 0.001	速度滤波 (一阶) : 时间常量 T	
0x00n20A03	读/写	P/PID (MW)	REAL64	cm ²	[0.0 ... 1000000]	A 面气缸面积 $_{AA}$, 单位为 cm ²	保留参数!
0x00n20A04	读/写	P/PID (MW)	REAL64	cm ²	[0.0 ... 1000000]	B 面气缸面积 $_{AB}$, 单位为 cm ²	保留参数!
0x00n20A05	读/写	P/PID (MW)	REAL64	cm ³ /s	[0.0 ... 1000000]	额定体积流量 $_{Q_{nom}}$, 单位为 cm ³ /s	保留参数!
0x00n20A06	读/写	P/PID (MW)	REAL64	bar	[0.0 ... 1000000]	额定压力或阀压降, P_{nom} , 以bar为单位	保留参数!
0x00n20A07	读/写	P/PID (MW)	UINT32	1	[1 ... 255]	系统压力 P_o 的轴 ID	保留参数!
驱动:							
0x00n30001	读取	驱动器: 每个	UINT32	1	[1 ... 255]	驱动器 ID	
0x00n30002	读取	驱动器: 每个	STRING[30+1]	1	30 个字符	驱动器名称	
0x00n30003	读取	驱动器: 每个	UINT32	1	枚举 (>0)	驱动器类型 [▶ 109]	
0x00n30004	读/写	驱动器: 每个	UINT32	1	Byteoffset	输入地址偏移 (I/O-输入-图像)	更改 I/O 地址
0x00n30005	读/写	驱动器: 每个	UINT32	1	Byteoffset	输出地址偏移 (I/O-输出-图像)	更改 I/O 地址
0x00n30006	读/写	驱动器: 每个	UINT16	1	[0,1]	电机极性	如果已发出启用控制器的信号, 则不允许写入。
0x00n3000A	读/写	驱动器: 每个	UINT32	1	枚举 (≥0)	驱动模式	
0x00n3000B	读/写	驱动器: 每个	REAL64	%	[-1.0 ... 1.0]	最小输出限制 (输出限制) (默认设置: -1.0 = -100)	
0x00n3000C	读/写	驱动器: 每个	REAL64	%	[-1.0 ... 1.0]	最大输出限制 (输出限制) (默认设置: 1.0 = 100%)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n3000D	读取	驱动器: 每个	UINT32	INC		输出增量的最大数量 (输出掩码)	
0x00n30010	读/写	驱动器: 每个	UINT32	1		内部驱动控制双字, 用于确定驱动运行模式	保留!
0x00n30011	读/写	每个	UINT32	1	≥ 5	内部驱动复位计数器 (启用和复位的 NC 循环时间)	保留!
0x00n30101	读/写	D: 伺服	REAL64	例如 mm/s	>0.0	参考输出的参考速度 (速度预控制)	
0x00n30102	读/写	D: 伺服	REAL64	%	[0.0 ... 5.0]	以百分比为单位的参考输出 (默认设置: 1.0 = 100%)	
0x00n30103	读取	D: 伺服	REAL64	例如 mm/s	>0.0	100% 输出时产生的速度	
0x00n30104	读/写	D: 伺服	REAL64	例如 mm/s	±∞	用于轴漂移校准 (偏移校准) 的速度偏移 (DAC 偏移)	
0x00n30105	读/写	D: 伺服 (Sercos、Profi、Drive、AX200x、CANopen)	REAL64	1	[0.0 ... 100000000.0]	速度缩放 (响应驱动器中权重的缩放因子 (scaling factor))	适用于 Sercos、Profi Drive、AX200x、CANopen
0x00n30106	读/写	D: Profi Drive DSC	UINT32	0.001 * 1/s	≥ 0	Profibus/Profi Drive DSC: 位置控制增益 Kpc	仅适用于 Profi Drive DSC
0x00n30107	读/写	D: Profi Drive DSC	REAL64	1	≥ 0.0	Profibus/Profi Drive DSC: 用于计算 'XERR' 的缩放比例 (默认: 1.0)	仅适用于 Profi Drive DSC
0x00n30109	读/写	D: 伺服 (Sercos、CANopen)	REAL64	1	[0.0 ... 100000000.0]	位置缩放 (响应驱动器中权重的缩放因子 (scaling factor))	适用于 Sercos、CANopen
0x00n3010A	读/写	D: 伺服 (Sercos、Profi、Drive、AX200x、CANopen)	REAL64	1	[0.0 ... 100000000.0]	加速度缩放 (响应驱动器中权重的缩放因子 (scaling factor))	适用于 Sercos、Profi Drive、AX200x、CANopen
0x00n3010B	读/写	D: 伺服 (Sercos、Profi、Drive、AX200x、CANopen)	REAL64	1	[0.0 ... 100000000.0]	适用于 "TorqueOffset" 的扭矩缩放 (旋转电机) 或力缩放 (直线电机) (响应驱动器中权重的缩放因子 (scaling factor)) (作为预控制的加法力矩)	适用于 Sercos、Profi Drive、AX200x、CANopen
0x00n3010C	读/写	D: 伺服 (Sercos、Profi、Drive、AX200x、CANopen)	REAL64	1	[0.0 ... 100000000.0]	适用于 "SetTorque" 的扭矩缩放 (旋转电机) 或力缩放 (直线电机) (响应驱动器中重量的缩放因子 (scaling factor)) (例如 MC_TorqueControl, 使用驱动器运行模式 CST)	适用于 Sercos、Profi Drive、AX200x、CANopen TC3.1 B4024.2 及以上
0x00n30120	读/写	D: servo/hydraulics/	UINT32	1	≥ 0	表 ID (0: 无表)	仅适用于 KL4xxx、M2400、通用型
0x00n30121	读/写	D: 伺服/液压	UINT32	1	≥ 0	插值类型 0: 线性 2: 样条	仅适用于 KL4xxx、M2400、通用型

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n30122	读/写	伺服/液压	REAL64	%	[-1.0 ... 1.0]	输出偏移 (以百分比为单位) 注: 根据特征评估进行操作!	仅适用于 KL4xxx、 M2400、通用型
0x00n30151	读/写	D: 伺服/非线性	REAL64	1	[0.0 ... 100.0]	象限补偿系数 (I 和 III 象限之间的关系)	
0x00n30152	读/写	D: 伺服/非线性	REAL64	1	[0.01 ... 1.0]	以百分比为单位的速度参考点 (1.0 = 100 %)	
0x00n30153	读/写	D: 伺服/非线性	REAL64	1	[0.01 ... 1.0]	以百分比为单位的输出参考点 (1.0 = 100 %)	
0x00030301	读/写	D: 步进电机	UINT8	1		位掩码: 循环 1	
0x00030302	读/写	D: 步进电机	UINT8	1		位掩码: 循环 2	
0x00030303	读/写	D: 步进电机	UINT8	1		位掩码: 循环 3	
0x00030304	读/写	D: 步进电机	UINT8	1		位掩码: 循环 4	
0x00030305	读/写	D: 步进电机	UINT8	1		位掩码: 循环 5	
0x00030306	读/写	D: 步进电机	UINT8	1		位掩码: 循环 6	
0x00030307	读/写	D: 步进电机	UINT8	1		位掩码: 循环 7	
0x00030308	读/写	D: 步进电机	UINT8	1		位掩码: 循环 8	
0x00030310	读/写	D: 步进电机	UINT8	1		位掩码: 保持电流	

6.4.4.2 轴状态的"索引偏移"规范 (索引组 0x4100 + ID)

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n00000	读取	每个 (轴数据的在线结构)	{ INT32 INT32 REAL64 REAL64 REAL64 REAL64 REAL64 REAL64 UINT32 UINT32 REAL64 REAL64 REAL64 REAL64 REAL64 REAL64 UINT32 UINT32 UINT32 REAL64	1 例如 mm 例如度数 例如 mm 例如度数 例如 mm/s 例如 mm/s % 例如 mm 例如 mm 例如 mm % 1 1 1 0; 1,2,3... 例如 mm/s^2	轴在线结构 (NC/CNC) 错误状态 预留 实际位置 模轴实际位置 设置位置 模轴设置位置 可选: 实际速度 设置速度 速度超驰 (1000000 == 100%) 预留 滞后误差位置 最大负位置滞后的峰值保持值 (位置) 最大位置滞后的峰值保持值 (位置) 控制器输出 (以百分比为单位) 总输出 (以百分比为单位) 轴状态双字 轴控制双字 从轴耦合状态 (状态) 轴控制环路索引 实际加速度	从 TwinCAT 3 开始发生变化, 无法通过示波器 观察! (NCAXISSTATE ONLINESTRUCT)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			REAL64	例如 mm/s^2		设置加速度	
			REAL64	例如 mm/s^3		设置加加速度 (从 TwinCAT 3.1 B4013 起的新功能)	
			REAL64	例如, 100% = 1000		设置扭矩或设置力 ("SetTorque")	
			REAL64	例如, 100% = 1000		实际扭矩或实际力 (从 TwinCAT 3.1 B4013 起的新功能)	
			REAL64	例如 %/s		设置扭矩变化或设置力变化 (设置扭矩或设置力的时间导数) (从 TwinCAT 3.1 B4024.2 起)	
			REAL64	例如, 100% = 1000		附加设定扭矩或附加设定力 ("TorqueOffset") (从 TwinCAT 3.1 B4024.2 起)	
			...				
			}			256 字节	
0x00000001	读取	每个	UINT32	1		轴状态错误代码	符号访问: "ErrState"
0x00n00009	读取	每个	UINT32	1	≥ 0	设置循环计数器 (SAF 时间戳)	
0x00n0000A	读取	每个	REAL64	例如 mm		设置位置	符号访问: "SetPos"
0x00n0000B	读取	每个	REAL64	例如度数		模轴设置位置	符号访问: "SetPosModulo"
0x00n0000C	读取	每个	INT32	1		模轴设置旋转	
0x00n0000D	读取	每个	REAL64	1	[-1.0, 0.0, 1.0]	设置行进方向	
0x00n0000E	读取	每个	REAL64	例如 mm/s		设置速度	符号访问: "SetVelo"
0x00n0000F	读取	每个	REAL64	例如 mm/s^2		设置加速度	符号访问: "SetAcc"
0x00n00010	读取	每个	REAL64	例如 mm/s^3		设置加加速度 (设置加速度的时间导数)	符号访问: "SetJerk"
0x00n00011	读取	每个	REAL64	例如, 分别为 Nm 或 N, 例如, 100% = 1000		设置扭矩 (旋转电机) 或 设置力 (直线电机) ("SetTorque")	从 TwinCAT 3.1 B4022 起的新功能 符号访问: "SetTorque"
0x00n00012	读取	每个	REAL64	1		设置耦合系数 (设置齿轮比)	
0x00n00013	读取	每个	REAL64	例如 mm		预期目标位置	
0x00n00014	读取	伺服	{			剩余行进时间和距离 (伺服) :	始终连接至 SEC 501 端口!
			REAL64	s	≥ 0	剩余行进时间	
			REAL64	例如 mm	≥ 0	剩余距离	
			}				
0x00n00015	读取	每个	UINT32	1	≥ 0	设置命令编号	符号访问: "CmdNo"

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n00016	读取	伺服	REAL64	s	≥ 0	最后运动命令的定位时间 (起始 → 目标位置窗口)	
0x00n00017	读取	伺服	REAL64	%	[0.0…1.0] 1.0=100%	为速度设置速率比 注：最初仅针对 FIFO 组执行	从 TwinCAT 3.1 B4020 起的新功能
0x00000018	读写	伺服	写入			读取"停止信息" (停止距离、停止时间)	始终连接至 SEC 501 端口!
			REAL64	例如 mm/s^2	≥ 0	轴停止减速度	
			REAL64	例如 mm/s^3	≥ 0	轴停止加速度	
			读取				
			REAL64	例如 mm	≥ 0	停止距离	
			REAL64	s	≥ 0	停止时间	
0x00n0001A	读取	每个	REAL64	例如 mm		未校正设置位置	
0x00n0001D	读取	每个	REAL64	1	[-1.0, 0.0, 1.0]	未校正设置行进方向	
0x00n0001E	读取	每个	REAL64	例如 mm/s		未校正设置速度	
0x00n0001F	读取	每个	REAL64	例如 mm/s^2		未校正设置加速度	
0x00000020	读取	每个	UINT32	1	参见枚举	耦合状态 (状态)	
0x00000021	读取	每个	UINT32	1	≥ 0	耦合表索引	
0x00000022	读取	伺服主轴/从轴耦合 类型：线性，(&SPECIAL)	{ REAL64 REAL64 REAL64 REAL64 }			读取耦合参数 (伺服)： 参数 1：线性：齿轮比 参数 2：线性：保留 参数 3：线性：保留 参数 4：线性：保留	
0x00000023	读取	伺服主轴/从轴耦合 类型：线性，(&SPECIAL)	REAL64	1	[±1000000.0]	读取齿轮比 (伺服) 类型线性	
0x00000024	读取	伺服	UINT32	1	≥ 0	活动轴控制电路的编号/索引 (编码器、控制器和轴接口三合一)	
0x00000025	读取	伺服	UINT16	1	0/1	是否激活通过轴接口 PCLtoNC 指定外部设定点?	
0x00000026	读取	伺服主轴/从轴耦合 类型：同步	REAL64 [64]	1	$\pm\infty$	读取从轴同步配置文件的特征值 类型：同步	从 TwinCAT 3 起修改
0x00000027	读写	伺服主轴/从轴耦合 类型：表格、MF	写入 VOID 或 REAL64 或 dword、dword、real64 读取 REAL64 [32]	例如 mm	$\pm\infty$	读取"表耦合信息" - 没有"当前信息"数据 - 某个"主轴位置"可选 - 适用于某个表 ID 和可选"主轴位置" (TC 3.1 B4017) 读取表耦合信息 [▶ 111] 的结构	仅端口 500! 从 TwinCAT 3 起修改

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00000028	读写	伺服主轴/从轴耦合 类型: MULTICAM (CamAddition)	写入			读取"多表耦合信息" (CamAddition)	仅端口 500!
			UINT32	1	≥ 0	查询所涉及的表 ID	
			读取				
			96 字节			读取多表耦合信息 [▶ 111] 的结构	
0x00000029	读取	伺服	UINT32	1		延迟错误反应时的延迟错误代码 (错误预先警告) (参见位 <i>ErrorPropagationDelayed</i>)	
0x0000002A	读取	伺服	REAL64	例如 mm	$\pm\infty$	从设置位置衰减到实际位置时的位置差 (衰减部分)	
0x0000002B	读取	伺服	REAL64	例如 mm/s	$\pm\infty$	从设置位置衰减到实际位置时的相对速度 (衰减部分)	
0x0000002C	读取	伺服	REAL64	例如 mm/s^2	$\pm\infty$	从设置位置衰减到实际位置时的相对加速度 (衰减部分)	
0x0000002D	读取	伺服	UINT32	1	≥ 0	初始化命令计数器 (InitializeCommandCounter)	新
0x0000002E	读取	伺服	UINT32	1	≥ 0	复位命令计数器 (ResetCommandCounter)	新
0x00000030	读取	伺服	REAL64	例如 Nm/s 或 N/s	$\pm\infty$	设置扭矩变化 (旋转电机) 或 设置力变化 (直线电机) (设置扭矩或设置力的时间导数)	从 TwinCAT 3.1 B4024 起的新功能
0x00000031	读/写	伺服	REAL64	例如, 分别为 Nm 或 N, 例如, 100% = 1000		用于预控制的附加设定扭矩 (旋转电机) 或 加性附加设定力 (直线电机)。 ("TorqueOffset")	从 TwinCAT 3.1 B4024.2 起 符号访问: "TorqueOffset"
0x00000040	读取	伺服	UINT32	1	≥ 0	用于在数据不一致的情况下校正 NC 设定点的计数器 (通过 Idx-Group 0x1000 和 Idx-Offset 0x0020 激活)	从 TwinCAT 3.1 B4020 起的新功能
0x00000050	读取	每个	UINT32	1		设置行进阶段 (SWGenerator)	无法用示波器追踪!
0x00000051	读取	每个	UINT16	1		轴是否被禁用?	无法用示波器追踪!
0x00n00060	读/写	每个 (在线设定点结构) 40 字节	{			简单轴设定点结构 (NC/CNC)	无法用示波器追踪! TC 3.1 B4022.30 及以上
			REAL64	例如 mm		设置位置	
			REAL64	例如 mm/s		设置速度	
			REAL64	例如 mm/s^2		设置加速度/减速度	
			REAL64	1	[-1.0, 0.0, 1.0]	设置行进方向	
			REAL64	例如 mm/s^3		设置加加速度	
			}				

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n00060	读/写	每个 (在线设定点结构) 56 字节	{			扩展轴设定点结构 (NC/CNC)	无法用示波器追踪! TC 3.1 B4022.29 及以上
			REAL64	例如 mm		设置位置	
			REAL64	例如 mm/s		设置速度	
			REAL64	例如 mm/s^2		设置加速度/减速度	
			REAL64	1	[-1.0, 0.0, 1.0]	设置行进方向	
			REAL64	例如 mm/s^3		设置加加速度	
			REAL64	Nm 或 N 或 %		设置扭矩或设置力	
			REAL64	Nm/s 或 N/s 或 %/s		设置扭矩或设置力的时间导数 (斜坡)	
			}				
0x00n00061	读/写	每个 (在线动态设定点结构) 32 字节	{			轴动态设定点结构 (NC/CNC)	TC 3.1 B4022.30 及以上
			REAL64	例如 mm/s		设置速度	
			REAL64	例如 mm/s^2		设置加速度/减速度	
			REAL64	1	[-1.0, 0.0, 1.0]	设置行进方向	
			REAL64	例如 mm/s^3		设置加加速度	
			}				
0x00n00061	读/写	每个 (在线动态设定点结构) 48 字节	{			轴动态设定点结构 (NC/CNC)	TC 3.1 B4022.29 及以上
			REAL64	例如 mm/s		设置速度	
			REAL64	例如 mm/s^2		设置加速度/减速度	
			REAL64	1	[-1.0, 0.0, 1.0]	设置行进方向	
			REAL64	例如 mm/s^3		设置加加速度	
			REAL64	Nm 或 N 或 %		设置扭矩或设置力	
			REAL64	Nm/s 或 N/s 或 %/s		设置扭矩或设置力的时间导数 (斜坡)	
			}				
0x00n00062	读/写	每个 (在线扭矩设定点结构) 16 字节	{			扭矩设定点结构 (NC/CNC)	TC 3.1 B4022.30 及以上
			REAL64	Nm 或 N 或 %		设置扭矩或设置力	
			REAL64	Nm/s 或 N/s 或 %/s		设置扭矩或设置力的时间导数 (斜坡)	
			}				
0x00000063	读写	仅适用于 SERCOS/SoE 和 CANopen/CoE	写入			读取活动"驱动运行模式"	从 TC 3.1 B4022 (NC 4443) 起更新 始终连接至 SEC 501 端口!
			UINT32	1		保留	
			UINT32	1		保留	
			读取				
			INT32	枚举 [▶ 110] (参见附录)	[0; 1, 2, 3, …] 特殊情况: ≥ 100: SoE <0: CoE	当前活动"驱动运行模式" (通用模式)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			UINT32	1		保留	
0x00n10002	读取	每个 (编码器)	REAL64	例如 mm		实际位置 (带有实际位置补偿值) n = 0: 轴的标准编码器 > 0: 轴的第 n 个编码器 (可选)	符号访问: "ActPos"
0x00n10003	读取	每个 (编码器)	REAL64	例如度数		模轴实际位置	符号访问: "ActPosModulo"
0x00n10004	读取	每个 (编码器)	INT32	1		模轴实际旋转	
0x00n10005	读取	每个 (编码器)	REAL64	例如 mm/s		可选: 实际速度	符号访问: "ActVelo"
0x00n10006	读取	每个 (编码器)	REAL64	例如 mm/s^2		可选: 实际加速度	符号访问: "ActAcc"
0x00n10007	读取	每个 (编码器)	INT32	INC		编码器实际增量	
0x00n10008	读取	每个 (编码器)	INT64	INC		软件 - 实际增量计数器	
0x00n10009	读取	每个 (编码器)	UINT16	1	0/1	参考 标志 ("校准标志")	
0x00n1000A	读取	每个 (编码器)	REAL64	例如 mm		实际位置校正值 (测量系统错误校正)	
0x00n1000B	读取	每个 (编码器)	REAL64	例如 mm		无实际位置补偿值的实际位置	无法用示波器追踪!
0x00n10010	读取	每个 (编码器)	REAL64	例如 mm/s		无实际位置补偿值的实际速度	
0x00n10012	读取	每个 (编码器)	REAL64	例如 mm		未滤波实际位置 (带有实际位置补偿值)	
0x00n10014	读取	编码器: SoE、CoE、MDP 742	REAL64	例如 mm/s		可选: 实际驱动速度 (直接从 SoE、CoE 或 MDP 742 驱动器传输)	从 TwinCAT 3.1 B4020.30 起的新功能
0x00n10015	读取	每个 (编码器)	REAL64	例如 mm/s		可选: 未滤波实际速度	
0x00n10017	读取		REAL64	例如 mm		读出 MC_SetPosition 偏移	
0x00n10018	读取	PTP 轴 编码器轴	UINT32	0/1	0/1	在 NC 编码器重新初始化已开始后, 返回重新初始化的状态 (索引组 0x4200+ID; 索引偏移 0x00n0003B)。 n = 0: 轴的标准编码器 n > 0: 轴的第 n 个编码器 (可选)	端口 501
0x00n10101	读取	INC (编码器)	REAL64	例如 mm		内部硬件锁存器启动和生效时间之间的位置差的反馈	无法用示波器追踪!
0x00n20001	读取	R: 每个	INT32	1		控制器的错误状态 n = 0: 轴的标准控制器 > 0: 轴的第 n 个控制器 (可选)	
0x00n20002	读取	R: 每个	REAL64	例如 mm/s		控制器输出 (以绝对单位表示)	符号访问: "CtrlOutput"
0x00n20003	读取	R: 每个	REAL64	%		控制器输出 (以百分比为单位)	无法用示波器追踪!

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n20004	读取	R: 每个	REAL64	V		控制器输出 (以伏特为单位)	无法用示波器追踪!
0x00n2000D	读取	R: 每个	REAL64	例如 mm		滞后误差位置 (无死区时间补偿)	基本单位
0x00n2000F	读取	R: 每个	REAL64	例如 mm		滞后误差位置 (有死区时间补偿)	符号访问: "PosDiff"
0x00n20010	读取	R: 每个	REAL64	例如 mm		位置最大负滞后误差的峰值保持值	
0x00n20011	读取	R: 每个	REAL64	例如 mm		位置最小正滞后误差的峰值保持值	
0x00n20012	读取	R: 每个	REAL64	例如 mm/s		滞后误差速度	未执行!
0x00n20021	读取	R: 每个	REAL64	例如 mm		主轴与从轴位置滞后误差之间的差 (偏差) (主轴滞后误差减去从轴滞后误差)	符号访问: "PosDiffCouple"
0x00n20022	读取	R: 每个	REAL64	例如 mm		主轴和从轴位置滞后误差之间最大负差的峰值保持值	基本单位
0x00n20023	读取	R: 每个	REAL64	例如 mm		主轴和从轴位置滞后误差之间最大正差的峰值保持值	基本单位
0x00n20101	读取	R: P/PID (位置)	REAL64	例如 mm/s		控制器 P(比例) 部分 (以绝对单位表示)	
0x00n20102	读取	R: PID (位置)	REAL64	例如 mm/s		控制器 I(积分) 部分 (以绝对单位表示)	
0x00n20103	读取	R: PID (位置)	REAL64	例如 mm/s		控制器 D(微分) 部分 (以绝对单位表示)	
0x00n20104	读取	R: PID (位置)	UINT16	1	0/1	I(积分) 部分限制激活?	
0x00n20105	读取	R: PID (位置)	UINT16	1	0/1	D(微分) 部分限制激活?	
0x00n20106	读取	R: PID (位置)	UINT16	1	0/1	I 部分 ARW 测量激活? ARW: 防积分饱和	未执行!
0x00n20110	读取	R: PID (位置)	REAL64	例如 mm/s		控制器的加速度预控制 Yacc (以绝对单位表示) 注: 功能取决于控制器类型!	加速度预控制
0x00n20111	读取	R: PP (位置)	REAL64	mm/s/mm	≥0	内部插值比例增益 kp 或 kv	PP 控制器
0x00n20201	读取	R: P,PID (速度)	REAL64	例如 mm/s		控制器的速度部分	
0x00n20202	读取	R: P,PID (速度)	REAL64	%		控制器的速度部分 (以百分比为单位)	无法用示波器追踪!
0x00n20203	读取	R: P,PID (速度)	REAL64	V		控制器的速度部分 (以伏特为单位)	无法用示波器追踪!
0x00n20201	读取	R: P/PID (速度)	REAL64	例如 mm/s		控制器 P(比例) 部分 (以绝对单位表示)	
0x00n20202	读取	R: P/PID (速度)	REAL64	例如 mm/s		控制器 I(积分) 部分 (以绝对单位表示)	
0x00n20203	读取	R: P/PID (速度)	REAL64	例如 mm/s		控制器 D(微分) 部分 (以绝对单位表示)	
0x00n20204	读取	R: P/PID (速度)	UINT16	1	0/1	I(积分) 部分限制激活?	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n20205	读取	R: P/ PID (速度)	UINT16	1	0/1	D(微分) 部分限制激活?	
0x00n20206	读取	R: P/ PID (速度)	UINT16	1	0/1	I(积分) 部分 ARW 测量激活? (ARW: 积分抗饱和)	
0x00n2020A	读取	R: P/ PID (速度)	REAL64	例如 mm/s		速度控制器的总输入大小	
0x00n20A00	读取	R: PID (MW)	REAL64	%	[-1.0...1.0]	设置速度的位移 (预控制)	保留参数!
0x00n20A01	读取	R: PID (MW)	REAL64	例如 mm/s		控制器的 P(比例) 部分以绝对单位或百分比表示 (根据输出权重计算)	保留参数!
0x00n20A02	读取	R: PID (MW)	REAL64	例如 mm/s		控制器的 I(积分) 部分以绝对单位或百分比表示 (根据输出权重计算)	保留参数!
0x00n20A03	读取	R: PID (MW)	REAL64	例如 mm/s		控制器的 D(微分) 部分以绝对单位或百分比表示 (根据输出权重计算)	保留参数!
0x00n20A04	读取	R: PID (MW)	UINT16	1	0/1	I(积分) 部分限制激活?	保留参数!
0x00n20A05	读取	R: PID (MW)	UINT16	1	0/1	D(微分) 部分限制激活?	保留参数!
0x00n20A06	读取	R: PID (MW)	UINT16	1	0/1	I(积分) 部分 ARW 测量激活? ARW: 防积分饱和	保留参数!
0x00n20A10	读取	R: PID (MW)	REAL64	例如 mm/s		控制器的加速度预控制 Yacc (以绝对单位表示)	保留参数!
0x00n30001	读取	D: 每个	INT32	1		驱动器的错误状态	
0x00n30002	读取	D: 每个	REAL64	例如 mm/s		总输出 (以绝对单位表示)	符号访问: "DriveOutput" "
0x00n30003	读取	D: 每个	REAL64	%		总输出 (以百分比为单位)	
0x00n30004	读取	D: 每个	REAL64	V		总输出 (以伏特为单位)	无法用示波器追踪!
0x00n30005	读取	D: 每个	REAL64	例如 mm/s		最大负总输出的峰值保持值	
0x00n30006	读取	D: 每个	REAL64	例如 mm/s		最大正总输出的峰值保持值	
0x00n30007	读取	D: 每个	REAL64	例如 100% = 1000, 例如 Nm 或 N		分别为实际扭矩或实际力 (通常 100% = 1000)	从 TwinCAT 3.1 B4022 起 符号访问: "ActTorque"
0x00n30008	读取	D: 每个	REAL64	例如 Nm/s 或 N/s	±∞	分别为实际扭矩变化或实际力变化 (分别为实际扭矩或实际力的时间导数)	从 TwinCAT 3.1 B4024 起
0x00n30013	读取	D: 每个	REAL64	%		总输出以百分比为单位 (基于非线性特征曲线!)	
0x00n30014	读取	D: 每个	REAL64	V		总输出以伏特为单位 (基于非线性特征曲线!)	无法用示波器追踪!

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n3011A	读取	D: 伺服 (Sercos、CANopen)	REAL64	例如 mm		可选输出滤波： 滤波设置位置	新 适用于 Sercos、CANopen
0x00n3011E	读取	D: 伺服 (Sercos、CANopen)	REAL64	例如 mm/s		可选输出滤波： 滤波设置速度	新 适用于 Sercos、CANopen
0x00n3011F	读取	D: 伺服 (Sercos、CANopen)	REAL64	例如 mm/s^2		可选输出滤波： 滤波设置加速度/设置减速度	新 适用于 Sercos、CANopen

6.4.4.3 轴功能的"索引偏移"规范 (索引组 0x4200 + ID)

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00000001	写入	每个	VOID			复位轴	也适用于 FIFO 轴!
0x00000002	写入	每个	VOID			停止轴	也适用于 FIFO 轴!
0x00000003	写入	每个	VOID			清除轴 (任务)	也适用于 FIFO 轴!
0x00000004	写入	每个	{ REAL64 REAL64 }	例如 mm/ s^2 例如 mm/ s^3	> 0.0	紧急停止 (使用控制斜坡) 减速度 (必须大于或等于原始减速度) 加加速度 (必须大于或等于原始加加速度)	仅适用于 PTP 轴!
0x00000005	写入	PTP 轴	{ REAL64 REAL64 }	例如 mm/ s^2 例如 mm/ s^3	> 0.0	可参数化停止 (使用控制斜坡) 减速度 加加速度	仅适用于 PTP 轴! 保留功能，无标准!
0x00000009	写入	PTP 轴	{ REAL64 REAL64 REAL64 }	例如度数 例如 mm/ s^2 例如 mm/ s^3	≥ 0.0	定向停止 (定向结束位置) 模除结束位置 (模除目标位置) 减速度 (目前未激活) 加加速度 (尚未执行)	仅适用于 PTP 轴!
0x00000010	写入	每个	VOID			参考轴 ("校准")	
0x00000011	写入	每个	{ UINT32 UINT32 REAL64 }	枚举 枚举 例如 mm	参见附录 参见附录 ±∞	轴的新结束位置 结束位置类型 [▶ 103] (参见附录) 预留 (TwinCAT 3) 新结束位置 (目标位置)	从 TwinCAT 3 起修改
0x00000012	写入	每个	{ UINT32 UINT32	枚举 枚举	参见附录 参见附录	轴的新结束位置和新速度 命令类型 [▶ 103] (参见附录) 结束位置类型 [▶ 103] (参见附录)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			REAL64	例如 mm	$\pm\infty$	新结束位置 (目标位置)	
			REAL64	例如 mm/s	≥ 0.0	新最终速度 (要求的行进速度)	
			REAL64	例如 mm	$\pm\infty$	可选: 激活新行进配置文件的切换位置	
			}				
0x000000015	写入	每个	{			用于主动定位的新动态参数	
			REAL64	例如 mm/ s^2	> 0.0	加速度	
			REAL64	例如 mm/ s^2	> 0.0	减速度	
			REAL64	例如 mm/ s^3	> 0.0	可选: 加加速度 (尚未执行)	
			}				
0x000000016	ReadWrite	每个伺服	写入 (80 字节)			通用轴启动 (UAS): 通过整合单轴启动指令与在线参数修改功能，并配合“缓冲模式”(请参见 TcMc2.lib) 实现动态控制流程的无缝衔接	始终连接至 SEC 501 端口! 从 TwinCAT 3 起修改
			{				
			UINT32	枚举	参见附录	启动类型 [▶ 102] (参见附录)	
			UINT32	1	≥ 0	用于检查和运行模式的位掩码 (默认值: 0)	
			REAL64	例如 mm	$\pm\infty$	结束位置 (目标位置)	
			REAL64	例如 mm/s	≥ 0.0	所需速度 Vrequ	
			REAL64	例如 mm/ s^2	≥ 0.0	可选: 加速度	
			REAL64	例如 mm/ s^2	≥ 0.0	可选: 减速度	
			REAL64	例如 mm/ s^3	≥ 0.0	可选: 加加速度	
			UINT32	枚举	参见附录	缓冲模式 [▶ 103] (命令缓冲)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm	$\pm\infty$	可选: 混合位置 (命令混合位置)	
			REAL64	例如 mm/s	≥ 0.0	可选: 初始分段速度 Vi ($0 \leq Vi \leq Vrequ$)	
			REAL64	例如 mm/s	≥ 0.0	可选: 分段结束速度 Vf ($0 \leq Vf \leq Vrequ$)	
			}				
			读取				
			{				
			UINT16	1	≥ 0	命令编号 (工作编号)	
			UINT16	1	≥ 0	命令状态	
			}				
0x000000017	ReadWrite	伺服	写入 (80 字节)			"主轴/从轴解耦"和"通用轴启动 (UAS)": 合并从轴的解耦命令 (IdxOffset: 0x00000041) 和后续通用轴启动 (UAS) (IdxOffset: 0x00000016)	尚未发布!
			{				

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			UINT32	枚举	参见附录	启动类型 [▶ 102] (参见附录)	
			UINT32	1	≥ 0	检查和运行模式的位掩码 (默认值: 0)	
			REAL64	例如 mm	±∞	结束位置 (目标位置)	
			REAL64	例如 mm/s	≥ 0.0	所需速度 <i>Vrequ</i>	
			REAL64	例如 mm/ s^2	≥ 0.0	加速度	
			REAL64	例如 mm/ s^2	≥ 0.0	减速度	
			REAL64	例如 mm/ s^3	≥ 0.0	加加速度	
			UINT32	枚举	参见附录	缓冲模式 [▶ 103] (命令缓冲)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm	±∞	可选: 混合位置 (命令混合位置)	
			REAL64	例如 mm/s	≥ 0.0	可选: 初始分段速度 <i>Vi</i> ($0 \leq Vi \leq Vrequ$)	
			REAL64	例如 mm/s	≥ 0.0	可选: 分段结束速度 <i>Vf</i> ($0 \leq Vf \leq Vrequ$)	
			}				
			读取				
			{				
			UINT16	1	≥ 0	命令编号 (工作编号)	
			UINT16	1	≥ 0	命令状态	
			}				
0x000000018	写入	每个	VOID			运动命令释放轴锁 (TcMc2)	
0x000000019	写入	每个	UINT32	1	> 0	设置外部轴错误 (运行时错误)	使用时请注意!
0x00n0001A	写入	每个	{			设置实际轴位置	使用时请注意! 也适用于 FIFO 轴!
			UINT32	枚举	参见附录	实际位置类型 [▶ 103] (参见附录)	始终连接至 SEC 501 端口! 从 TwinCAT 3 起修改
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm	±∞	轴的实际位置 n = 0: 轴的标准编码器 $n > 0$: 轴的第 n 个编码器 (可选)	
			}				
0x00n0001B	写入	每个	UINT32	1	0/1	设置参考标志 ("校准标志") $n = 0$: 轴的标准编码器 $n > 0$: 轴的第 n 个编码器 (可选)	使用时请注意! 也适用于 FIFO 轴!
0x00n0001C	写入	伺服	{			仅设置实际轴位置, 不干预设定位置 (也适用于及运行中进程)	使用时请注意!
			UINT32	枚举	参见附录	实际位置类型 [▶ 103] (参见附录)	使用时请注意!
			REAL64	例如 mm	±∞	轴的实际位置 n = 0: 轴的标准编码器 > 0: 轴的第 n 个编码器 (可选)	
			}				
0x00n0001D	写入	每个	{			驱动侧轴的实际值设置 (假设位置接口和编码器偏移为零!)。	使用时请注意! 仅适用于 CANopen!

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
						n = 0: 轴的标准编码器 n > 0: 轴的第 n 个编码器 (可选)	
			UINT32	枚举	参见附录	实际位置类型 [▶ 103] (参见附录)	
			REAL64	例如 mm	±∞	轴的实际位置	
			}				
0x00n0001E	写入	每个	{			实时设置新编码器缩放因子 (scaling factor) (在轴的运动过程中)	使用时请注意! 始终连接至 SEC 501 端口! 从 TwinCAT 3 起修改
			UINT16	枚举	1	编码器缩放因子 (scaling factor) 类型 1: 绝对 2: 相对	
			UINT16			ControlWord	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/ INC	[1.0E-8 ... 100.0]	新编码器缩放因子 (scaling factor) n = 0: 轴的标准编码器 n > 0: 轴的第 n 个编码器 (可选)	
			}				
0x00n0001F	写入	每个	{			实时设置实际轴位置 (在轴的运动过程中)	使用时请注意! 始终连接至 SEC 501 端口!
			UINT32	枚举		用于实时设置实际值的位置类型 1: 绝对 2: 相对	
			UINT32	1		控制双字, 例如用于"清除滞后误差"	
			REAL64			保留	
			REAL64	例如 mm	±∞	新实际轴位置	
			UINT32			保留	
			UINT32			保留	
			}				
0x000000020	写入	每个 1D 启动	{			标准轴启动:	从 TwinCAT 3 起修改
			UINT32	枚举	参见附录	启动类型 [▶ 102] (参见附录)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm	±∞	结束位置 (目标位置)	
			REAL64	例如 mm/s	≥0.0	所需速度	
			}				
0x000000021	写入	每个 1D 启动	{			扩展轴启动 (伺服):	从 TwinCAT 3 起修改
			UINT32	枚举	参见附录	启动类型 [▶ 102] (参见附录)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm	±∞	结束位置 (目标位置)	
			REAL64	例如 mm/s	≥0.0	所需速度	
			UINT32	0/1	0/1	标准加速度?	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/ s^2	≥0.0	加速度	
			UINT32	0/1	0/1	标准减速度?	
			UINT32			预留 (TwinCAT 3)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			REAL64	例如 mm/ s^2	≥ 0.0	减速度	
			UINT32	0/1	0/1	标准加速度?	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/ s^3	≥ 0.0	加速度	
			}				
0x000000022	写入	伺服 (MW)	{			特殊轴启动 (伺服) :	保留启动功能, 无标准!
			UINT32	枚举	参见附录	启动类型 [▶ 102] (参 见附录)	从 TwinCAT 3 起修改
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm	±∞	结束位置 (目标位置)	
			REAL64	mm/s	≥ 0.0	所需启动速度	
			REAL64	例如 mm	±∞	新速度等级的位置	
			REAL64	例如 mm/s	≥ 0.0	新结束速度等级	
			UINT32	0/1	0/1	标准加速度?	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/ s^2	≥ 0.0	加速度	
			UINT32	0/1	0/1	标准减速度?	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/ s^2	≥ 0.0	减速度	
			UINT32	0/1	0/1	标准加速度?	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/ s^3	≥ 0.0	加速度	
			}				
0x000000023	写入	伺服	{			启动外部设定点规范 (通过循环轴接口 PLCtoNC 设置)	从 TwinCAT 3 起修改
			UINT32	枚举	1: 绝对 2: 相对	启动类型 [▶ 102]	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm	±∞	新结束位置 (目标位 置) (可选) !	
			REAL64			预留 (TwinCAT 3)	
			}				
0x000000024	写入	伺服	VOID			停止/禁用外部设定点 规范 (循环轴接口 PLCtoNC)	
0x000000025	写入	伺服	{			开始反向定位操作 (伺 服) :	从 TwinCAT 3 起修改
			UINT32	枚举	1	启动类型 [▶ 102] (默 认: 1)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm	±∞	结束位置 1 (目标位 置)	
			REAL64	例如 mm	±∞	结束位置 2 (目标位 置)	
			REAL64	0/1	0/1	所需速度	
			REAL64	s	≥ 0.0	空闲时间	
			}				
0x000000026	写入	每个	{			启动驱动输出	从 TwinCAT 3 起修改
			UINT32	枚举	参见附录	输出类型 [▶ 110] (参 见附录)	
			UINT32			预留 (TwinCAT 3)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			REAL64 } }	例如 %	±∞	所需输出值 (例如 %)	
0x000000027	写入	每个	VOID			停止驱动输出	
0x000000028	写入	每个	{ UINT32 REAL64 }	枚举 例如 %	参见附录 ±∞	更改驱动输出: 输出类型 [▶ 110] (参见附录) 所需输出值 (例如 %)	
0x000000029	写入	每个	VOID			立刻采用当前速率比并保持不变, 直至下一次速率比更改!	保留功能, 无标准!
0x00000002A	写入	每个	{ 32 字节 }			计算和设置编码器偏移	保留功能, 无标准!
0x00000002B	读写	每个	数据写入: s. 'UAS' 数据读取: s. 'UAS'			停止外部设定点发生器并保持连续不断的运动 ('UAS': 通用轴启动)	保留功能, 无标准!
0x00000002C	写入	每个	UINT32		≥ 0	设置"归零状态" (供内部使用)	从 TwinCAT 3 起的新功能
0x00000002D	读写	伺服	写入			将 NC 控制轴切换到"循环同步扭矩模式" (CST), 并为其设置扭矩设定点。	谨防使用过程中的危险! (* 参见表结尾)
			{				
			UINT32			扭矩轴启动类型: 0x3001: 绝对 0x3002: 相对	
			UINT32	1 (位阵列)		内部控制掩码 (位阵列) : 00000000_00000001 (位 0) : 使用手动扭矩进行初始化。 10000000_00000000 (位 31) : 在 "ContinuousUpdate" 刷新当前命令参数 (fTorqueRamp、fVelocityLimitHigh、fVelocityLimitLow), 不增加命令编号。	
			UINT32	0/1	0/1	模式: 0: 默认 (离散) 1: 持续数据更新模式	
			UINT32	枚举	参见附录	缓冲模式 [▶ 103] 只能进行中止操作	
			REAL64	Nm 或 %	[0.0 ... 1.0E10]	扭矩目标值 (符号值)	
			REAL64	Nm/s 或 %/s	[0.0 ... 1.0E10]	扭矩变化速度	
			REAL64	例如 mm/s	[0.0 ... 1.0E10] 'VelocityLimitHigh' 必须大于或等于 'VelocityLimitLow' (两个值都可以是负数)。	高速度限制	
			REAL64	例如 mm/s	[0.0 ... 1.0E10]	低速度限制	
			REAL64	例如 mm/ s^2	[0.0 ... 1.0E10]	加速度	
			REAL64	例如 mm/ s^2	[0.0 ... 1.0E10]	减速度	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			REAL64	Nm 或 %	[0.0 ... 1.0E10]	可选：手动输入扭矩启动值 (同步值)	
			}				
			读取				
			{				
			UINT16	1	>=0	命令编号 (工作编号)	
			UINT16	1	>=0	命令状态	
			}				
0x0000002E						预留	
0x0000002F						预留	
0x00000030	写入	伺服	{			启动区间补偿 (伺服)	仅影响旧 TwinCAT 2 系统
			UINT32	枚举	参见附录	补偿类型 [▶ 104] (参见附录)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/ s^2	≥ 0.0	最大加速度增加	
			REAL64	例如 mm/ s^2	≥ 0.0	最大减速度增加	
			REAL64	例如 mm/s	> 0.0	最大增速	
			REAL64	例如 mm/s	> 0.0	过程的基本速度	
			REAL64	例如 mm	±∞	需要补偿的路径差	
			REAL64	例如 mm	> 0.0	补偿的路径距离	
			}				
0x00000030	读写	伺服将实际执行参数作为返回值返回	{ 读取+写入:			启动区间补偿 (伺服) 注: 仅包含在 'TcMc2.lib' 或 'Tc2_MC2.library' 中	从以下版本起更改 TwinCAT 2 211R3 TwinCAT 3
			UINT32	枚举	参见附录	补偿类型 [▶ 104] (参见附录)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/ s^2	≥ 0.0	=> 最大加速度增量 <= 返回实际执行的加速度增量 (在 'TcMc2.lib' 或 'Tc2_MC2.library' 中新增加)	
			REAL64	例如 mm/ s^2	≥ 0.0	=> 最大减速度增量 <= 返回实际执行的减速度增量 (在 'TcMc2.lib' 或 'Tc2_MC2.library' 中新增加)	
			REAL64	例如 mm/s	> 0.0	=> 要求的最大增速 <= 返回实际执行的增速	
			REAL64	例如 mm/s	> 0.0	过程的基本速度	
			REAL64	例如 mm	±∞	=> 需要补偿的路径差 <= 返回实际执行的路径差	
			REAL64	例如 mm	> 0.0	=> 需要补偿的最大距离 <= 返回实际执行的距离	
			UINT32	1	≥ 0	<= 返回警告 ID (例如 0x4243)	
			UINT32			预留 (TwinCAT 3)	
			}				
0x00000031	写入	伺服	VOID			停止区间补偿 (伺服)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00000032	写入	伺服	{			通过速度跳变 (伺服) 启动反转操作: (可用于确定速度阶跃响应)	从 TwinCAT 3 起修改
			UINT32	枚举	1	启动类型 [▶ 102] (默认: 1)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/s	±∞	所需速度 1 (允许为负值)	
			REAL64	例如 mm/s	±∞	所需速度 2 (允许为负值)	
			REAL64	s	> 0.0	速度 1 和 2 的行进时间	
			REAL64	s	≥ 0.0	空闲时间	
			UINT32	1	0, 1, 2, 3...	可选: 重复次数, 默认"0": 无时间限制	
			UINT32			预留 (TwinCAT 3)	
			}				
0x00000033	写入	伺服	{			正弦振荡序列 - 用作单一正弦波振荡 (正弦波发生器) - 用作正弦波振荡序列 (例如, 用于波特图)	从 TwinCAT 3 起修改
			UINT32	枚举	1	启动类型 [▶ 102] (固定为启动类型 1)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/s	> 0.0	基本振幅 (例如 2.5 mm/s)	
			REAL64	Hz	[0.0 10.0]	基频 (例如 1.953125 Hz)	
			REAL64	例如 mm/s	≥ 0.0	开始时的启动振幅 (例如 0.0 mm/s)	
			REAL64	例如 mm/ REV	> 0.0	进给恒定电机 (每电机 转一圈) (例如 10.0 mm/REV)	
			REAL64	Hz	≥ 1.0	频率范围: 启动频率 (例如 20.0 Hz)	
			REAL64	Hz	≤ 1/(2*dT)	频率范围: 停止频率 (例如 500.0 Hz)	
			REAL64	s	> 0.0	步进时长 (例如 2,048s)	
			UINT32	1	[1 ... 200]	测量次数 (步进循环) (例如 20)	
			UINT32	1		尚未使用的并行测量数 (如 1) !	
			}				
0x00000034	写入	伺服	{			定相 - 开始定相 - 停止定相	
			UINT32	枚举	1	定相类型: 1: 绝对 2: 相对 4096: 停止	
			UINT32	1	≥ 0	控制掩码 位 0: 持续更新	
			UINT32	1	≥ 0	主轴 ID (用于多主轴)	
			UINT32			保留	
			REAL64	例如 mm	±∞	相移	
			REAL64	例如 mm/s	> 0.0	速度	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			REAL64	例如 mm/ s^2	≥ 0.0	加速度	
			REAL64	例如 mm/ s^2	≥ 0.0	减速度	
			REAL64	例如 mm/ s^3	≥ 0.0	加加速度	
			REAL64[4]			保留	
			UINT32			保留	
			UINT32	1	枚举	缓冲模式 (未执行)	
			REAL64	例如 mm	±∞	混合位置 (未执行)	
			}				
0x00n0003B	写入	PTP 轴 编码器轴	VOID			触发 NC 编码器重新初始化，确保 IO 值有效 n = 0: 轴的标准编码器 n > 0: 轴的第 n 个编码器 (可选)	谨防使用过程中 的危险! 一定不能启用控制器 (位置跳转) 轴状态索引偏移 0x00n10018 可以用于读出 NC 编码器重新初始化是否已完成。 端口 501
0x000000040 (0x00n00040)	写入	主轴/从轴耦合 (伺服)	{			主轴/从轴耦合 (伺服) :	“飞锯”扩展! 角度 > 0.0 和 £90.0 度 (平行 锯: 90.0 度)
			UINT32	枚举	参见附录	从轴类型 [▶ 104]/耦合 类型 (参见附录)	
			UINT32	1	[1...255]	主轴/组的轴 ID	
			UINT32	1	[0...8]	主轴的子索引 n (默 认: 值: 0)	
			UINT32	1	[0...8]	从轴的子索引 n (默 认: 0)	
			REAL64	1	[±1000000.0]	参数 1: 线性: 齿轮比 FlySawVelo: 保留 FlySaw: 主轴绝对同步位置 [mm]	
			REAL64	1	[±1000000.0]	参数 2: 线性: 保留 FlySawVelo: 保留 FlySawPos: 从轴绝对同步位置 [mm]	
			REAL64	1	[±1000000.0]	参数 3: 线性: 保留 FlySawVelo: 倾斜角 (以 [度] 为单位) FlySawPos: 倾斜角 (以 [度] 为单位)	
			REAL64	1	[±1000000.0]	参数 4: 线性: 保留 FlySawVelo: 齿轮比 FlySawPos: 齿轮比	
			}				
0x000000040 (0x00n00040)	写入	主轴/从轴耦合 (伺服)	{			主轴/从轴耦合 (伺服) :	多主轴耦合 (MC_GearInMultiMaster) 版本 V1 和 V2 从 TwinCAT 3 起修改
			UINT32	枚举	参见附录	从轴类型 [▶ 104]/耦合 类型 (参见附录)	
			UINT32	1	[1...255]	主轴/组的轴 ID	
			UINT32	1	[1...8]	主轴的子索引 n (默 认: 值: 0)	
			UINT32	1	[1...8]	从轴的子索引 n (默 认: 0)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			UINT32	1	[0...255]	轴 ID 主轴 2	
			UINT32	1	[0...255]	轴 ID 主轴 3	
			UINT32	1	[0...255]	轴 ID 主轴 4	
			UINT32	1	[0...255]	保留 (轴 ID 主轴 5)	
			UINT32	1	[0...255]	保留 (轴 ID 主轴 6)	
			UINT32	1	[0...255]	保留 (轴 ID 主轴 7)	
			UINT32	1	[0...255]	保留 (轴 ID 主轴 8)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm/ s^2		从轴的最大加速度/减 速度	
			UINT32	1	≥ 0	控制掩码, 之前未使用 (配置文件的检查和运 行模式)	
			UINT32			预留 (TwinCAT 3)	
			扩展 V2 (可选) :				
			REAL64	例如 mm/ s^2	≥ 0.0	从轴的最大减速度	
			REAL64	例如 mm/ s^3	≥ 0.0	从轴的最大加速度	
			REAL64	例如 mm/s	≥ 0.0	从轴的最大速度	
			REAL64			保留	
			REAL64			保留	
			} 64 或 104 字节				
0x000000041	写入	主轴/从轴解耦 (伺服)	VOID			主轴/从轴解耦 (伺 服)	
0x000000041	写入	主轴/从轴解 耦, 具有可配置 跟随功能 (伺 服)	{			主轴/从轴解耦, 具有 可配置跟随功能 (例 如, 新结束位置、新速 度、停止、E-stop) (伺服)	尚未发布! 从 TC3 起修改
			UINT32	枚举	参见附录	解耦类型 [► 104] (参 见附录)	
			UINT32			预留 (TwinCAT 3)	
			REAL64	例如 mm	$\pm\infty$	可选: 新结束位置	
			REAL64	例如 mm/s	>0.0	可选: 新要求速度	
			REAL64	例如 mm/ s^2	≥ 0.0 (0: 默认)	可选: 新结束位置、新 速度和紧急停止 (E- stop) 的加速度	
			REAL64	例如 mm/ s^2	≥ 0.0 (0: 默认)	可选: 新结束位置、新 速度和紧急停止 (E- stop) 的减速度	
			REAL64	例如 mm/ s^3	≥ 0.0 (0: 默认)	可选: 新结束位置、新 速度和紧急停止 (E- stop) 的加加速度	
			}				
0x000000042	写入	主轴/从轴耦合 类型: 线性 (&SPECIAL)	{			更改耦合参数 (耦 合) :	
			REAL64	1	[± 1000000.0]	参数 1: 线性: 齿轮比	
			REAL64	1	[± 1000000.0]	参数 2: 线性: 保留	
			REAL64	1	[± 1000000.0]	参数 3: 线性: 保留	
			REAL64	1	[± 1000000.0]	参数 4: 线性: 保留	
			}				
0x000000043	写入	主轴/从轴表耦 合 类型: 表格	{			更改表耦合参数 (伺 服) :	
			REAL64	mm	$\pm\infty$	从轴位置偏移	
			REAL64	mm	$\pm\infty$	主轴位置偏移	
			}				

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
			读取: REAL32	例如 mm	±∞	根据对应的表格ID退出校正表, 来确定偏移量	
0x000000049	写入	主轴/从轴表耦合 类型: 表格	REAL64	1	±∞	更改从轴表列的从轴缩放因子 (scaling factor) (默认: 1.0)	
0x00000004A(0x00n0004A)	写入	主轴/从轴通用表耦合 (伺服)	{			主轴/从轴独表耦合 (伺服) :	从 TwinCAT 3 起修改
			UINT32	枚举	参见附录	从轴类型/耦合类型 [▶ 104] (参见附录)	
			UINT32	1	[1...255]	主轴的轴 ID	
			UINT32	1	[0...8]	主轴的子索引 n (默认: 值: 0)	
			UINT32	1	[0...8]	从轴的子索引 n (默认: 值: 0)	
			UINT32	1	1...255]	耦合表的表 ID (类型: TABULAR)	
			UINT32	1		表插值类型	
			REAL64	mm	±∞	从轴位置偏移 (类型: TABULAR)	
			REAL64	mm	±∞	主轴位置偏移 (类型: TABULAR)	
			REAL64	mm	±∞	从轴位置缩放 (类型: TABULAR)	
			REAL64	mm	±∞	主轴位置缩放 (类型: TABULAR)	
			UINT32	1	[0,1]	从轴绝对位置? (类型: TABULAR)	
			UINT32	1	[0,1]	主轴绝对位置? (类型: TABULAR)	
			UINT32	枚举	参见附录	更改的激活类型: 0: '瞬时的' (instantaneous) '(默认) 1: '在主凸轮位置' 2: '在主轴位置' 3: '下一个周期'	
			UINT32			预留 (TwinCAT 3)	
			REAL64	mm	±∞	激活位置	
			UINT32	枚举	参见附录	主轴缩放类型: 0: 用户定义 (默认) 1: 使用自动偏移缩放 2: 关闭	
			UINT32	枚举	参见附录	从轴缩放类型: 0: 用户定义 (默认) 1: 使用自动偏移缩放 2: 关闭	
			}				
0x00000004A(0x00n0004A)	写入	主轴/从轴通用表耦合 (伺服)	{			主轴/从轴独表耦合 (伺服) :	从 TwinCAT 3 起修改
			UINT32	枚举	参见附录	从轴类型/耦合类型 [▶ 104] (参见附录)	
			UINT32	1	[1...255]	主轴的轴 ID	
			UINT32	1	[0...8]	主轴的子索引 n (默认: 值: 0)	
			UINT32	1	[0...8]	从轴的子索引 n (默认: 值: 0)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明	
			UINT32	1	[1...255]	耦合表的表 ID (类型: TABULAR)		
			UINT32	1		表插值类型		
			REAL64	mm	$\pm\infty$	从轴位置偏移 (类型: TABULAR)		
			REAL64	mm	$\pm\infty$	主轴位置偏移 (类型: TABULAR)		
			REAL64	mm	$\pm\infty$	从轴位置缩放 (类型: TABULAR)		
			REAL64	mm	$\pm\infty$	主轴位置缩放 (类型: TABULAR)		
			UINT32	1	[0,1]	从轴绝对位置? (类型: TABULAR)		
			UINT32	1	[0,1]	主轴绝对位置? (类型: TABULAR)		
			UINT32	枚举	参见附录		更改的激活类型: 0: '瞬时的 (instantaneous)' (默认) 1: '在主凸轮位置' 2: '在主轴位置' 3: '下一个周期'	
			UINT32			预留 (TwinCAT 3)		
			REAL64	mm	$\pm\infty$	激活位置		
			UINT32	枚举	参见附录		主轴缩放类型: 0: 用户定义 (默认) 1: 使用自动偏移缩放 2: 关闭	
			UINT32	枚举	参见附录		从轴缩放类型: 0: 用户定义 (默认) 1: 使用自动偏移缩放 2: 关闭	
			MultiCam 扩展:					
			UINT32	枚举	参见附录	凸轮运行模式		
			UINT32	1	[1…255]	参考表 ID		
			BYTE[104]			预留 (TwinCAT 3)		
			}					
0x00000004B(0x00n0004B)	写入	主轴/从轴通用 飞锯 (伺服)	{			主轴/从轴同步耦合 (伺服) :	从 TwinCAT 3 起修改	
			UINT32	枚举	参见附录			
			UINT32	1	[1…255]	主轴的轴 ID		
			UINT32	1	[0…8]	主轴的子索引 n (默认: 值: 0)		
			UINT32	1	[0…8]	从轴的子索引 n (默认: 0)		
			REAL64	1	$\pm\infty (> 0.0)$	齿轮比		
			REAL64	mm	$\pm\infty$	主轴同步位置		
			REAL64	mm	$\pm\infty$	从轴同步位置		
			REAL64	mm/s	≥ 0.0	从轴速度 (可选)		
			REAL64	mm/s^2	≥ 0.0	从轴加速度 (可选)		
			REAL64	mm/s^2	≥ 0.0	从轴减速度 (可选)		
			REAL64	mm/s^3	≥ 0.0	从轴加加速度 (可选)		
			UINT32	1	≥ 0	位掩码 (默认值: 0)		
			UINT32			预留 (TwinCAT 3)		
			}					

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x0000004D(0x00n004D)	写入	主轴/从轴耦合 类型: TABULAR 和 MF	{ UINT32 UINT32 REAL64 UINT32 UINT32 REAL64 REAL64 REAL64 REAL64 MultiCam 可选扩展: UINT32 UINT32 }	枚举 枚举 例如 mm ±∞ 枚举 枚举 例如 mm 例如 mm 1 1 1 1 1 ≥ 0 ≥ 0 ≥ 0 0/1 0/1	参见附录 参见附录 ±∞ 参见附录 参见附录 主轴位置偏移 从轴位置偏移 主轴位置缩放 从轴位置缩放 凸轮表 ID 预留 (TwinCAT 3) 	修改表格里的缩放比例 (伺服) : 更改的激活类型 0: '瞬时的 (instantaneous)' (默认) 1: '在主凸轮位置' 2: '在主轴位置' 3: '下一个周期' 预留 (TwinCAT 3) 激活位置 主轴缩放类型 0: 用户定义 (默认) 1: 使用自动偏移缩放 2: 关闭 从轴缩放类型 0: 用户定义 (默认) 1: 使用自动偏移缩放 2: 关闭 主轴位置偏移 从轴位置偏移 主轴位置缩放 从轴位置缩放 	从 TwinCAT 3 起修改
0x00000050	写入	每个	VOID			停用整个轴 (禁用)	
0x00000051	写入	每个	VOID			激活整个轴 (启用)	
0x00000052	写入	伺服	{ UINT32 UINT32 REAL64 UINT32 UINT32 }	1 1 枚举 1 0/1 0/1	≥ 0 ≥ 0 参见附录 (>0) ±∞ 0/1 0/1	在有/无外部设定点规范的情况下, 更改活动轴控制回路 (编码器、控制器和轴接口三合一) : 轴控制回路的编号/索引 (默认值: 0) 同步行为的切换类型 [▶ 113] 1: '标准' 切换同步值 (可选) 通过轴接口的方式指定外部设定点? 注: 目前尚未使用! 预留 (TwinCAT 3) 	从 TwinCAT 3 起修改
0x00000060	写入	每个	VOID			关闭驱动输出 (禁用)	
0x00000061	写入	每个	VOID			激活驱动输出 (启用)	
0x00000062	写入	高/低	UINT16	1	0/1	释放保持制动器? 0: 自动激活 (默认) 1: 必须始终释放 注: 复位轴时重置为'0'!	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00000063	写入	仅适用于 SERCOS/SoE 和 CANopen/CoE	{			激活"驱动运行模式" (如位置速度、扭矩等)	从 TC 3.1 B4022 (NC 4443) 起更新 始终连接至 SEC 501 端口!
			INT32	枚举 ► 110 (参见附录)	[0; 1, 2, 3, ...] 特殊情况: ≥ 100: SoE < 0: CoE	新"驱动运行模式" (通用模式)	
			UINT32	1	0	保留	
			UINT32	1	0	保留	
			UINT32	1	0	保留	
			}				
0x00000070	写入	每个	VOID			将轴从例如 3D 组等返回其自身的 PTP 组	

* 以下警告与索引偏移 0x0000002D 有关:

⚠ 危险

有由于轴意外运动而造成生命危险或严重伤害或财产损失的风险

使用功能块时，轴将切换到 CST 模式。使用功能块后（尤其是出错情况下），轴可能仍处于 CST 模式。当释放轴时，这可能会导致突然和意外运动（尤其是升降轴）。

- 确保不存在风险评估中所界定的危险。
- 通过功能块 检查当前运行模式。
- 如果轴未处于与位置相关的运行模式 (CSV/CSP)，则应在启用之前转换：
 - 直接使用转换为所需的位置相关运行模式 (CSV/CSP)，或者
 - 直接使用 / 转换为所需的位置相关运行模式 (CSV/CSP) (TwinCAT 3.1.4024.40 及以上)
 将轴间接切换到位置相关运行模式的其他功能块只能在有限范围内做到这一点，因此不能用于刻意改变运行模式。
- ⇒ 随后，有必要再次检查轴是否真的处于位置相关运行模式 (CSV/CSP)，如果不是，则需要中止并进行错误处理。

6.4.4.4 循环轴过程数据的"索引偏移"规范 (索引组 0x4300 + ID)

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n00000	读/写	每个 (PLC→NC)	[128 字节]		STRUCT 参见轴接口	轴结构 (PLC→NC) n = 0: 标准轴接口 n > 0: 第 n 个轴接口 (可选)	写入命令仅为可选项! 考虑安全方面! <i>PLCTONC_AXIS_REF</i>
0x00n00001	读/写	每个 (PLC→NC)	UINT32	1	>0	控制双字	写入命令仅为可选项! 可进行符号访问! "ControlDWor d"
0x00n00002	读/写	每个 (PLC→NC)	UINT16	1	0/1	控制器启用	无法用示波器追踪!
0x00n00003	读/写	每个 (PLC→NC)	UINT16	1	0/1	进给启用加	无法用示波器追踪!
0x00n00004	读/写	每个 (PLC→NC)	UINT16	1	0/1	进给启用减	无法用示波器追踪!
0x00n00007	读/写	每个 (PLC→NC)	UINT16	1	0/1	参考凸轮	无法用示波器追踪!

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n00021	读/写	每个 (PLC→NC)	UINT32	%	0...1000000	速度超驰 (1000000 == 100%)	写入命令仅为可选项! 可进行符号访问! "OverrideV"
0x00n00022	读/写	每个 (PLC→NC)	UINT32	1	枚举	运行模式轴	写入命令仅为可选项!
0x00n00025	读/写	每个 (PLC→NC)	REAL64	例如 mm		实际位置校正值 (测量系统错误校正)	写入命令仅为可选项!
0x00n00026	读/写	每个 (PLC→NC)	REAL64	例如 mm/s		外部控制器组件 (位置控制器组件)	写入命令仅为可选项!
0x00n00027	读/写	每个 (PLC→NC)	{			外部设定点生成	写入命令仅为可选项! 从 TC3 起修改
			REAL64	例如 mm	±∞	外部设置位置	
			REAL64	例如 mm/s	±∞	外部设置速度	
			REAL64	例如 mm/s^2	±∞	外部设置加速度	
			INT32	1	+1, 0, -1	外部设置行进方向	
			UINT32			保留 (TC3)	
			REAL64			保留 (TC3)	
			}				
0x00n00080	读取	每个 (PLC→NC)	{256 字节}		STRUCT 参见轴接口	轴结构 (NC→PLC) 注: 尺寸和对齐方式有所改变 n = 0: 标准轴接口 n > 0: 第 n 个轴接口 (可选)	从 TC3.NCTOPLC_AXIS_REF 起更改
0x00n00071	读取	每个 (PLC→NC)	UINT8	1	>0	状态双字: 字节 1	
0x00n00072	读取	每个 (PLC→NC)	UINT8	1	>0	状态双字: 字节 2	
0x00n00073	读取	每个 (PLC→NC)	UINT8	1	>0	状态双字: 字节 3	
0x00n00074	读取	每个 (PLC→NC)	UINT8	1	>0	状态双字: 字节 4	
0x00n00081	读取	每个 (PLC→NC)	UINT32	1	>0	状态双字 (完整)	可进行符号访问! "StateDWord"
0x00n00082	读取	每个 (PLC→NC)	UINT16	1	0/1	轴已准备好运行	无法用示波器追踪!
0x00n00083	读取	每个 (PLC→NC)	UINT16	1	0/1	轴已被引用	无法用示波器追踪!
0x00n00084	读取	每个 (PLC→NC)	UINT16	1	0/1	轴处于保护运行模式 (例如从轴)	无法用示波器追踪!
0x00n00085	读取	每个 (PLC→NC)	UINT16	1	0/1	轴处于快速模式	无法用示波器追踪!
0x00n00088	读取	每个 (PLC→NC)	UINT16	1	0/1	轴 I/O 数据无效	无法用示波器追踪!
0x00n00089	读取	每个 (PLC→NC)	UINT16	1	0/1	轴处于错误状态	无法用示波器追踪!
0x00n0008A	读取	每个 (PLC→NC)	UINT16	1	0/1	轴向较大值移动	无法用示波器追踪!
0x00n0008B	读取	每个 (PLC→NC)	UINT16	1	0/1	轴向较小值移动	无法用示波器追踪!
0x00n0008C	读取	每个 (PLC→NC)	UINT16	1	0/1	轴处于逻辑静止状态 (仅考虑设定点) (位置控制器?)	无法用示波器追踪!

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n0008D	读取	每个 (PLC→NC)	UINT16	1	0/1	轴被引用	无法用示波器追踪!
0x00n0008E	读取	每个 (PLC→NC)	UINT16	1	0/1	轴处于位置窗口中	无法用示波器追踪!
0x00n0008F	读取	每个 (PLC→NC)	UINT16	1	0/1	轴处于目标位置 (达到目标位置)	无法用示波器追踪!
0x00n00090	读取	每个 (PLC→NC)	UINT16	1	0/1	轴具有恒定速度或旋转速度	无法用示波器追踪!
0x00n0009A	读取	每个 (PLC→NC)	UINT16	1	0/1	运行模式未执行 (忙)	无法用示波器追踪!
0x00n0009B	读取	每个 (PLC→NC)	UINT16	1	0/1	轴有指令, 正在执行指令	无法用示波器追踪!
0x00n000B1	读取	每个 (PLC→NC)	UINT32	1	≥0	轴错误代码	
0x00n000B2	读取	每个 (PLC→NC)	UINT32	1	枚举	轴的运动状态 (主轴状态 [▶ 110]/从轴状态 [▶ 110])	可进行符号访问! "AxisState"
0x00n000B3	读取	每个 (PLC→NC)	UINT32	1	枚举	轴的运行模式 (修订版 NC)	
0x00n000B4	读取	每个 (PLC→NC)	UINT32	1	枚举	轴参考状态	可进行符号访问! "HomingState"
0x00n000B5	读取	每个 (PLC→NC)	UINT32	1	枚举	轴耦合状态	可进行符号访问! "CoupleState"
0x00n000B6	读取	每个 (PLC→NC)	UINT32	1	≥0	轴的 SVB 条目/任务 (PRE 表)	
0x00n000B7	读取	每个 (PLC→NC)	UINT32	1	≥0	轴的 SAF 条目/任务 (EXE 表)	
0x00n000B8	读取	每个 (PLC→NC)	UINT32	1	≥0	轴 ID	
0x00n000B9	读取	每个 (PLC→NC)	UINT32	1	≥0	运行模式状态双字: 位 0: 位置范围监测激活? 位 1: 目标位置窗口监测激活? 位 2: 循环距离激活? 位 3: 物理运动监测激活? 位 4: PEH 时间监测激活? 位 5: 间隙补偿激活? 位 6: 延迟错误反应模式激活? 位 7: 模除运行模式激活 (模除轴) ? 位 16: 跟随错误监测位置激活? 位 17: 跟随错误监测速度激活? 位 18: 最小结束位置监测激活? 位 19: 最大结束位置监测激活? 位 20: 实际位置校正激活?	
0x00n000BA	读取	每个 (PLC→NC)	REAL64	例如 mm		实际位置 (计算出的绝对值)	

索引偏移 (十六进制)	访问	轴类型	数据类型	物理单位	定义范围	描述	说明
0x00n000BB	读取	每个 (PLC→NC)	REAL64	例如 mm		模除实际位置	
0x00n000BC	读取	每个 (PLC→NC)	INT32	1		模除旋转	
0x00n000BD	读取	每个 (PLC→NC)	REAL64	例如 mm/s		实际速度 (可选)	
0x00n000BE	读取	每个 (PLC→NC)	REAL64	例如 mm		跟随误差位置	
0x00n000BF	读取	每个 (PLC→NC)	REAL64	例如 mm		设置位置	
0x00n000C0	读取	每个 (PLC→NC)	REAL64	例如 mm/s		设置速度	
0x00n000C1	读取	每个 (PLC→NC)	REAL64	例如 mm/s^2		设置加速度	
0x00n10000	读/写	编码器: 每个 (NC→IO)	{40 字节 }		STRUCT 参见编码器 IO 接口	编码器输出结构 (NC→IO, 40 字节) <i>NCENCODERSTRUCT_OUT2</i>	写入命令仅为可选项! 考虑安全方面!
0x00n10080	读取	编码器: 每个 (IO→NC)	{40 字节 }		STRUCT 参见编码器 IO 接口	编码器输入结构 (IO→NC, 40 字节) <i>NCENCODERSTRUCT_IN2</i>	
0x00n30000	读/写	驱动器: 每个 (NC→IO)	{40 字节 }		STRUCT 参见驱动器 IO 接口	驱动器-输出-结构 (NC→IO, 40 字节) <i>NCDRIVESTRUCT_OUT2</i>	写入命令仅为可选项! 考虑安全方面!
0x00n30080	读取	驱动器: 每个 (IO→NC)	{40 字节 }		STRUCT 参见驱动器 IO 接口	驱动器-输入-结构 (NC→IO, 40 字节) <i>NCDRIVESTRUCT_IN2</i>	

6.4.5 编码器规范

6.4.5.1 编码器参数的"索引偏移"规范 (索引组 0x5000 + ID)

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
0x00000001	读取	每个	UINT32	1	[1 ... 255]	编码器 ID	
0x00000002	读取	每个	UINT8[30+1]	1	30 个字符	编码器名称	
0x00000003	读取	每个	UINT32	1	参见 枚举 (>0)	编码器类型 ▶ 105]	
0x00000004	读/写	每个	UINT32	1	Byteoffset	输入地址偏移 (IO-输入-图像)	更改 I/O 地址
0x00000005	读/写	每个	UINT32	1	Byteoffset	输出地址偏移 (IO-输出-图像)	更改 I/O 地址
0x00000006	读/写	每个	REAL64	例如 mm/INC	[1.0E-12 ... 1.0E+30]	结果缩放因子 (scaling factor) (分子/分母) 注: 从 TC3 起, 缩放因子 (scaling factor) 由分子和分母两部分组成 (默认: 1.0)。	如果已发出启用控制器的信号, 则不允许写入。
0x00000007	读/写	每个	REAL64	例如 mm	[±1.0E+9]	位置偏移	如果已发出启用控制器的信号, 则不允许写入。
0x00000008	读/写	每个	UINT16	1	[0,1]	编码器计数方向	如果已发出启用控制器的信号, 则不允许写入。
0x00000009	读/写	每个	REAL64	例如 mm	[0.001 ... 1.0E+9]	模除因子	

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
0x00000000A	读/写	每个	UINT32	1	参见 枚举 (>0) 在附录中	编码器模式 [▶ 106]	
0x00000000B	读/写	每个	UINT16	1	0/1	软结束最小监测?	
0x00000000C	读/写	每个	UINT16	1	0/1	软结束最大监测?	
0x00000000D	读/写	每个	REAL64	mm		最小软结束位置	
0x00000000E	读/写	每个	REAL64	mm		最大软结束位置	
0x00000000F	读/写	每个	UINT32	1	参见 枚举 (>0) 在附录中	编码器评估方向 [▶ 106] (启用日志记录计数方向)	
0x000000010	读/写	每个	REAL64	s	[0.0...60.0]	实际位置值的滤波时间 (以秒为单位) (P-T1)	
0x000000011	读/写	每个	REAL64	s	[0.0...60.0]	实际速度值的滤波时间 (以秒为单位) (P-T1)	
0x000000012	读/写	每个	REAL64	s	[0.0...60.0]	实际加速度值的滤波时间 (以秒为单位) (P-T1)	
0x000000013	读/写	每个	UINT8[10+1]	1		物理单位	未执行!
0x000000014	读/写	每个	UINT32	1		单位 (位置、速度、时间) 的解释 第 0 位：速度单位为 x/min，而非 x/s 第 1 位：以千分之一为基本单位的位置	未执行! 位数组
0x000000015	读/写	每个	UINT32	INC	[0x0...0xFFFFFFFF]	编码器掩码 (以增量表示的编码器实际值的最大值) 注：编码器掩码可以是任何数值 (如 3600000)。与过去不同，它不再需要对应于连续的一系列二进制 1(² n-1)。	必须禁用轴才能进行写入访问。 另请参见“编码器掩码”参数
0x000000016	读/写	每个	UINT16	1	0/1	实际位置校正 (测量系统误差校正) ?	
0x000000017	读/写	每个	REAL64	s	[0.0...60.0]	实际位置校正的滤波时间 (以秒为单位) (P-T1)	
0x000000018	读/写	每个	UINT32	1	[0x0...0xFFFFFFFF]	原始增量值的滤波掩码 (0x0: 全通)	
0x000000019	读/写	每个	UINT32	1	参见 枚举 (>0) 在附录中	编码器绝对尺寸系统 [▶ 107]	如果已发出启用控制器的信号，则不允许写入。
0x00000001A	读/写	每个	UINT32	1	参见 枚举 (>0)	编码器位置初始化	未执行!
0x00000001B	读/写	每个	REAL64	例如 mm	[≥0, 模除因子/2]	模除启动的容差窗口	
0x00000001C	读取	每个	UINT32	1	参见 枚举 (>0)	编码器符号解释 [▶ 107] (数据类型)	
0x00000001D	读取	每个	UINT16	1	0/1	增量或绝对编码器? 0: 增量编码器类型 1: 绝对编码器类型	
0x000000020	读/写	每个	UINT32	1	参见 枚举 (>0)	编码器死区时间补偿模式 0: 关闭 (默认) 1: 开 (带速度) 2: 开 (带速度和加速度)	
0x000000021	读/写	每个	UINT32	1		用于编码器死区时间补偿的控制双字 (32 位) :	

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
						位 0 = 0: 相对 I/O 时间 (默认) 位 0 = 1: 绝对 I/O 时间	
0x000000022	读/写	每个	INT32	ns	[±1.0E+9]	用于编码器死区时间补偿的参数化时间移位之和 (通常为正数值)	
0x000000023	读/写	每个	REAL64	例如 mm/INC	[1.0E-12 ... 1.0E+30]	缩放因子(scaling factor)的组成部分: 分子 (=> 缩放因子(scaling factor)分子/缩放 (scaling factor)因子分母)	从 TC3 起更新 如果已发出启用控制器的信号, 则不允许写入。
0x000000024	读/写	每个	REAL64	1	[1.0E-12 ... 1.0E+30]	缩放因子(scaling factor)的组成部分: 分母 (=> 缩放因子(scaling factor)分子/缩放 (scaling factor)因子分母) 默认: 1.0	从 TC3 起更新 如果已发出启用控制器的信号, 则不允许写入。
0x000000025	读/写	每个	{ real64 real64 }16 字节	例如 mm/INC 1	[1.0E-12 ... 1.0E+30] [1.0E-12 ... 1.0E+30]	缩放因子(scaling factor)的组成部分: 分子 缩放因子(scaling factor)的组成部分: 分母 (=> 缩放因子(scaling factor)分子/缩放 (scaling factor)因子分母)	从 TC3 起更新
0x000000030	读/写	每个	UINT32	1		内部编码器控制双字, 用于指定运行模式和属性	从 TC3 起更新
0x000000101	读/写	INC	UINT16	1	[0,1]	参考凸轮的反向搜索方向?	
0x000000102	读/写	INC		1	[0,1]	同步脉冲的反向搜索方向?	
0x000000103	读/写	INC	REAL64	例如 mm	[±1.0E+9]	参考位置	
0x000000104	读/写	INC	UINT16	1	[0,1]	是否激活了参考凸轮和同步脉冲之间的距离监测?	未执行!
0x000000105	读/写	INC	UINT32	INC	[0 ...65536]	参考凸轮与同步脉冲之间的最小间隙 (以增量为单位)	未执行!
0x000000106	读/写	INC	UINT16	1	[0,1]	外部同步脉冲?	
0x000000107	读/写	INC	UINT32	1	参见 枚举 (>0)	参考模式 (同步条件) [▶ 107]	
0x000000108	读/写	INC	UINT32	1	[0x0000000F... 0xFFFFFFFF]二进制掩码: (² n - 1)	编码器子掩码 (编码器实际值绝对范围的最大值, 以增量为单位) 例如, 用作参考模式"软件同步"和 NC 保留数据"绝对 (MODULO模除) "、"增量 (SINGLETURN ABSOLUTE单转绝对) "的参考标记。	新 另请参见参数 "编码器掩码"

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
						注 1：编码器子掩码必须小于或等于编码器掩码。 注 2：编码器掩码必须是编码器子掩码的整数倍。 注 3：编码器子掩码必须是连续的二进制 1 序列 ($^2n-1$)，例如 0x000FFFFF。	
0x000000109	读/写	INC	UINT32	1	参见 枚举 (≥ 0)	回零传感器源 [▶ 108] 设置参考凸轮的数字输入源。	
0x000000110	读/写	INC (编码器模拟)	REAL64	1	[0.0 ... 1000000.0]	模拟编码器噪声部分的缩放/权重	

6.4.5.2 编码器状态的"索引偏移"规范 (索引组 0x5100 + ID)

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
0x000000001	读取	每个	INT32			误差状态编码器	
0x000000002	读取	每个	REAL64			实际位置 (带有实际位置补偿值)	可进行符号访问！ <i>ActPos</i>
0x000000003	读取	每个	REAL64			模除实际位置	可进行符号访问！ <i>ActPosModulo</i>
0x000000004	读取	每个	INT32			模除实际旋转	
0x000000005	读取	每个	REAL64			可选：实际速度	基本单位 / s 可进行符号访问！ <i>ActVelo</i>
0x000000006	读取	每个	REAL64			可选：实际加速度	基本单位 / s^2 可进行符号访问！ <i>ActAcc</i>
0x000000007	读取	每个	INT32			编码器实际增量	
0x000000008	读取	每个	INT64			软件 – 实际增量计数器	
0x000000009	读/写	每个	UINT16			参考旗标 (“校准旗标”)	
0x00000000A	读取	每个	REAL64			实际位置校正值 (测量系统误差校正)	
0x00000000B	读取	每个	REAL64			无实际位置补偿值的实际位置	
0x00000000C	读取	每个	REAL64	例如 mm		死区时间补偿导致的实际位置补偿值	
0x00000000D	读取	每个	REAL64	s		编码器死区时间补偿的时间位移总和 (参数化和可变死区时间) 注：系统中规定的死区时间为正值。	
0x00000000E	读取	每个	REAL64	例如 mm		内部位置偏移作为基准期 (模除范围) 减值的校正值	
0x000000010	读取	每个	REAL64	例如 mm/s		无实际位置补偿值的实际速度	
0x000000012	读取	每个	REAL64	例如 mm		未滤波实际位置 (带有实际位置补偿值)	
0x000000013	读取	每个	REAL64	例如 mm		滤波后的实际位置 (带实际位置校正值的偏移，无死区时间补偿)	

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
0x000000014	读取	类型SoE、CoE、MDP 742	REAL64	例如 mm/s		可选：实际驱动速度 (直接从 SoE、CoE 或 MDP 742 驱动器传输)	基本单位 / s 从 TC3.1 B4020.30 起更新
0x000000015	读取	每个	REAL64	例如 mm/s		可选：未滤波实际速度	基本单位 / s
0x000000016	读取	每个	读取 (16 字节 * N) { UINT32 ns ≥0 带有 32 位的 DcTimeStamp UINT32 保留 REAL64 例如 mm ±∞ 相关时间戳的实际位置 } [N]			读取实际位置缓冲	
0x000000017	读取		REAL64	例如 mm		读出 MC_SetPosition 偏移	
0x000000101	读取	INC	REAL64	例如 mm		读回硬件锁存器从激活到生效之间的位置差	无法用示波器追踪！
0x000000200	读取 写入	功能组"TouchProbeV2": - SERCOS/SoE - EtherCAT/CoE (CANopen DS402) - SoftDrive (TCom), - MDP 511 (EL5101, EL5151, EL5021, EL7041, EL7342)	写入 (24 字节) { UINT32 1 [1,2,3,4] 探头单元 (探头 1、2、3、4) UINT32[5] 保留 } 读取 (64 字节) { UINT32 1 [0/1] 接触式探头上升沿激活? UINT32 1 [0/1] 接触式探头上升沿有效? REAL64 例如 mm 接触式探头上升沿位置值 UINT32 1 ≥0 接触式探头上升沿计数器 (连续模式) UINT32 保留 UINT32 1 [0/1] 接触式探头下降沿激活? UINT32 1 [0/1] 接触式探头下降沿有效? REAL64 例如 mm 接触式探头下降沿位置值 UINT32 1 ≥0 接触式探头下降沿计数器 (连续模式) UINT32[5] 保留 }			读取"接触式探头"状态 (外部锁存器状态)	仅适用于 SAF 端口 501
0x000000201	读取	KL5101、SERCOS、AX2xxx、ProviDrive	UINT16	1	[0,1]	"外部锁存器功能"激活? 或 "接触式探头功能"激活? (边沿独立?)	无法用示波器追踪！
0x000000201	读取	CANopen	UINT32[4]	1	[0,1]	"外部锁存器功能 1 至 4"激活? 或	无法用示波器追踪！

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
						"接触式探头功能 1 至 4"激活?	
0x00000202	读取	KL5101、SERCOS、AX2xxx、ProviDrive	UINT16	1	[0,1]	外部锁存器值有效? 或 接触式探头锁定? (边沿独立?)	另请参见轴接口 NcToPlc (状态双字)
0x00000202	读取	CANopen	UINT32[4]	1	[0,1]	外部锁存器值 1 至 4 有效? 或 接触式探头 1 至 4 锁定?	另请参见轴接口 NcToPlc (状态双字)
0x00000203	读取	KL5101、SERCOS、AX2xxx、ProviDrive	UINT32	INC		外部/接触式探头硬件增量锁存器值	
0x00000204	读取	KL5101、SERCOS、AX2xxx、ProviDrive	UINT64	INC		外部/接触式探头软件增量锁存器值	
0x00000205	读取	KL5101、SERCOS、AX2xxx、ProviDrive	REAL64	例如 mm		外部/接触式探头位置锁存器值	基本单位
0x00000205	读取	CANopen	REAL64[4]	例如 mm		外部接触式探头值/位置锁存器值	基本单位
0x00000206	读取	KL5101、SERCOS、AX2xxx、ProviDrive	UINT32	INC		硬件增量锁存器值差 (NewLatch - LastLatch)	无法用示波器追踪!
0x00000207	读取	KL5101、SERCOS、AX2xxx、ProviDrive	UINT64	INC		软件增量锁存器值差 (NewLatch - LastLatch)	无法用示波器追踪!
0x00000208	读取	KL5101、SERCOS、AX2xxx、ProviDrive	REAL64	例如 mm		位置锁存器值差 (NewLatch - LastLatch)	无法用示波器追踪! 基本单位
0x00000210	读取	KL5101、AX2xxx、ProviDrive	UINT16	1	[0,1]	上升沿"外部锁存器功能"激活? 或 上升沿"接触式探头功能"激活?	无法用示波器追踪!
0x00000210	读取	CANopen	UINT16[4]	1	[0,1]	上升沿"外部锁存器功能"激活? 或 上升沿"接触式探头功能"激活?	无法用示波器追踪!
0x00000211	读取	KL5101、AX2xxx、ProviDrive	UINT16	1	[0,1]	下降沿"外部锁存器功能"激活? 或 下降沿"接触式探头功能"激活?	无法用示波器追踪!
0x00000211	读取	CANopen	UINT16[4]	1	[0,1]	下降沿"外部锁存器功能"激活? 或 下降沿"接触式探头功能"激活?	无法用示波器追踪!
0x00000212	读取	CANopen	UINT16	1	[0,1]	"接触式探头 1" 输入信号的状态	无法用示波器追踪!

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	说明
							TC3.1 B4024.11 及以上
0x000000213	读取	CANopen	UINT16	1	[0,1]	“接触式探头 2” 输入信号的状态	无法用示波器追踪! TC3.1 B4024.11 及以上

6.4.5.3 编码器功能的"索引偏移"规范 (索引组 0x5200 + ID)

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
0x00000001A	写入	每个	{			设置实际位置编码器/轴	基本单位
			UINT32	枚举	参见附录	实际位置类型 [▶ 103] (参见附录)	
			REAL64	mm	±∞	编码器/轴的实际位置 使用时请注意!	
			}				
0x00000001B	写入	每个	VOID			重新初始化实际编码器位置 注：对参考系统“绝对多圈量程（带单个溢出）”和“绝对单圈量程（带单个溢出）”有效。	从 TC3 起更新
0x000000200	写入	功能组"TouchProb eV2": - SERCOS/SoE, - EtherCAT/CoE (CANopen DS402) - SoftDrive (TCom), - MDP 511 (EL5101, EL5151, EL5021, EL7041, EL7342)	{			激活“接触式探头”（外部锁存器）	仅适用于 SAF 端口 501
			UINT32	1	[1,2,3,4]	探头单元（探头 1、2、3、4）	
			UINT32	1	[0,1]	信号边沿（0=上升沿，1=下降沿）	
			UINT32	1	[1,2]	探头模式（1=单，2=连续，...）	
			UINT32	1	[1,2,3,4; 128,129]	信号源（1=输入 1，2=输入 2，...）	
			UINT32			保留	
			UINT32			保留	
			}	24 字节			
0x00000201	写入	KL5101、SERCOS、AX2xxx、PROFIDrive	VOID			激活“外部锁存器”或激活“测量探头功能”（通常为上升沿）	
0x00000201	写入	CANopen	UINT32[4]			激活“外部锁存器”1至4或激活“测量探头功能”1至4（通常为上升沿）	
0x00000202	写入	KL5101、SERCOSAX2xxx、PROFIDrive	VOID			激活“外部锁存器”或激活“测量探头功能”（下降沿）	
0x00000202	写入	CANopen	UINT32[4]			激活“外部锁存器”1至4或激活“测量探头功能”1至4（下降沿）	
0x00000205	写入	功能组"TouchProb eV2": - SERCOS/SoE, - EtherCAT/CoE (CANopen DS402) - SoftDrive	{			停用“接触式探头”（外部锁存器）	仅适用于 SAF 端口 501
			UINT32	1	[1,2,3,4]	探头单元（探头 1、2、3、4）	
			UINT32	1	[0,1]	信号边沿（0=上升沿，1=下降沿）	
			UINT32			保留	

索引偏移(十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
		(TCom), - MDP 511 (EL5101, EL5151, EL5021, EL7041, EL7342)	UINT32			保留	
			UINT32			保留	
			UINT32			保留	
			} 24 字节				
0x00000205	写入	KL5101、 SERCOS、 AX2xxx、 PROFIDrive	VOID			停用"外部锁存器"或停用"测量探头功能"	
0x00000205	写入	CANopen	UINT32[4]			停用"外部锁存器"或停用"测量探头功能"	
0x00000210	写入	KL5101、 SERCOS、 AX2xxx、 PROFIDrive	REAL64	例如 mm	±∞	设置"外部锁存器事件"和"外部锁存器位置"	仅适用于 EtherCAT:

6.4.5.4 循环编码器过程数据的"索引偏移"规范 (索引组 0x5300 + ID)

索引偏移 (十 六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
0x00000000	读/写	每个 (NC→IO)	{ INT32 INT32 UINT8 UINT8 UINT8 UINT8 UINT8 INT32 INT32 INT32 INT32 UINT8 UINT8 UINT8 UINT8 INT32 INT32 } 40 字节	STRUCT 参见编 码器接口	编码器-输出-结构 (NC→IO, 40 字节) NCENCODERSTRUCT_O UT2	nDataOut1 nDataOut2 nCtrl1 nCtrl2 nCtrl3 nCtrl4 nDataOut3 nDataOut4 nDataOut5 nDataOut6 nCtrl5 nCtrl6 nCtrl7 nCtrl8 保留 保留	写入命令仅为可 选项! 考虑安全方面!
0x00000000	读/写	每个 (NC→IO), 可选 64 位编码器接 口 (例如 MDP513, 带 64 位)	{ UINT64 UINT64 UINT64 UINT64 UINT64 UINT64 UINT64 UINT64 UINT16 UINT16	STRUCT 参见编码器接口	可选 ENCODER- OUTPUT-STRUCTURE (NC→IO, 80 字节) NCENCODERSTRUCT_O UT3	nDataOut1 nDataOut2 nDataOut3 nDataOut4 nDataOut5 nDataOut6 nDataOut7 nDataOut8 nCtrl1 nCtrl2	写入命令仅为可 选项! 考虑安全方面! 从 TC3 起更新

索引偏移 (十六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
			UINT16	1	≥ 0	nCtrl3	
			UINT16	1	≥ 0	nCtrl4	
			UINT16	1	≥ 0	nCtrl5	
			UINT16	1	≥ 0	nComCtrl	
			INT32	1	≥ 0	保留	
			{ 80 字节				
0x00000001	写入	每个 (NC→IO)	{		STRUCT 参见编码器接口	按位访问 编码器-输出-结构 (NC→IO, 40 字节) NCENCODERSTRUCT_OUT2	写入命令仅为可选项! 考虑安全方面!
			UINT32	1	[0 … 39]	ByteOffset 输出结构中的相对地址偏移 [0..39]。 例如：要写入 "nControl1"， ByteOffset 必须为 8。	
			UINT32	1	[0x00000000… 0xFFFFFFFF]	BitSelectMask (BSM) 掩码在 DWORD 中定义允许写入的位。零位受保护，不受影响。	
			UINT32	1	[0x00000000… 0xFFFFFFFF]	数值 只有 BSM 等于 1 的值位才会被重写。	
			}				
0x00000080	读取	每次 (IO→NC)	{		STRUCT 参见编码器接口	编码器-输入-结构 (IO→NC, 40 字节) NCENCODERSTRUCT_IN2	
			INT32	INC	≥ 0	nDataIn1	
			INT32	INC	≥ 0	nDataIn2	
			UINT8	1	≥ 0	nState1	
			UINT8	1	≥ 0	nState2	
			UINT8	1	≥ 0	nState3	
			UINT8	1	≥ 0	nState4 (Bit0: WcState, Bit1: InputToggle)	
			INT32	INC	≥ 0	nDataIn3	
			INT32	INC	≥ 0	nDataIn4	
			INT32	INC	≥ 0	nDataIn5	
			INT32	INC	≥ 0	nDataIn6	
			UINT8	1	≥ 0	nState5	
			UINT8	1	≥ 0	nState6	
			UINT8	1	≥ 0	nState7	
			UINT8	1	≥ 0	nState8	
			INT32	[ns]	≥ 0	nDcInputTime (用于死区时间补偿的绝对/相对 DcInputShift)	
			INT32	1	≥ 0	保留	
			{ 40 字节				
0x00000080	读取	每个 (NC→IO) , 可选 64 位编码器接口 (例如, MDP513, 带 64 位)	{		STRUCT 参见编码器接口	可选编码器-输入-结构 (IO→NC, 80 字节) NCENCODERSTRUCT_IN3	从 TC3 起更新
			UINT64	INC	≥ 0	nDataIn1	
			UINT64	INC	≥ 0	nDataIn2	
			UINT64	INC	≥ 0	nDataIn3	

索引偏移 (十 六进制)	访问	组类型	数据类型	物理单位	定义范围	描述	备注
			UINT64	INC	≥ 0	nDataIn4	
			UINT64	INC	≥ 0	nDataIn5	
			UINT64	INC	≥ 0	nDataIn6	
			UINT64	INC	≥ 0	nDataIn7	
			UINT64	INC	≥ 0	nDataIn8	
			UINT16	1	≥ 0	nState1	
			UINT16	1	≥ 0	nState2	
			UINT16	1	≥ 0	nState3	
			UINT16	1	≥ 0	nState4	
			UINT16	1	≥ 0	nState5	
			UINT16	1	≥ 0	nComState (Bit0: <i>WcState</i> , Bit1: <i>InputToggle</i>)	
			INT32	[ns]	≥ 0	nDcInputTime (用于死 区时间补偿的绝对/相对 <i>DcInputShift</i>)	
			{ 80 字节				

6.4.6 控制器规范

6.4.6.1 控制器参数的"索引偏移"规范 (索引组 0x6000 + ID)

索引偏移 (十 六进制)	访问	控制器类型	数据类型	物理单位	定义范围	描述	备注
0x00000001	读取	每个	UINT32	1	[1 ... 255]	控制器 ID	
0x00000002	读取	每个	UINT8[30+1]	1	30 个符号	控制器名称	
0x00000003	读取	每个	UINT32	1	参见 枚举 (>0)	控制器类型 [▶ 104]	
0x0000000A	读/写	每个	UINT32	1	参见 枚举 (>0)	控制器模式	默认： 标准
0x0000000B	读/写	每个	REAL64	%	[0.0 ... 1.0]	速度预控制的权重 (标准 值: 1.0 = 100%)	
0x00000010	读/写	每个	UINT16	1	0/1	跟随错误监测位置?	
0x00000011	读/写	每个	UINT16	1	0/1	跟随错误监测速度?	
0x00000012	读/写	每个	REAL64	mm	[0.0...1.0E.6]	最大跟随误差位置	
0x00000013	读/写	每个	REAL64	s	[0.0...600]	最大跟随误差时间位置	
0x00000014	读/写	每个	REAL64	mm/s	[0.0...1.0E.6]	最大跟随误差速度	
0x00000015	读/写	每个	REAL64	s	[0.0...1.0E.6]	最大跟随误差时间速度	
0x00000021	读/写	每个	REAL64	1	[0.0...1000000. 0]	主轴和从轴之间位置差的 缩放因子 (scaling factor) (乘数) (在同 一坐标系中转换)	保留功能, 无标准!
0x00000100	读/写	P/PID (位置、 速度)	REAL64	1	[0.0...1.0]	控制器总输出的最大输出 限制 ()	(标准值: 0.5 == 50%)
0x00000102	读/写	P/PID (位置)	REAL64	mm/s/ mm	[0.0...1000.0]	比例放大系数 k_p 或 k_v	基础单位 / s / 基础单位位置控 制
0x00000103	读/写	PID (位置)	REAL64	s	[0.0 ... 60.0]	积分作用时间 T_n	位置控制
0x00000104	读/写	PID (位置)	REAL64	s	[0.0 ... 60.0]	微分作用时间 T_v	位置控制
0x00000105	读/写	PID (位置)	REAL64	s	[0.0 ... 60.0]	阻尼时间 T_d	位置控制
0x00000106	读/写	PP (位置)	REAL64	mm/s/ mm	[0.0...1000.0]	在超过速度限制时, 增加 比例放大系数 k_p 或 k_v , 这个系数是以百分比的形 式来表示的。	基础单位 / s / 基础单位位置控 制

索引偏移（十 六进制）	访问	控制器类型	数据类型	物理单位	定义范围	描述	备注
0x00000107	读/写	PP (位置)	REAL64	%	[0.0...1.0]	阈值水平速度（以百分比表示），超过该值时，将适用额外的比例放大系数 kp 或 kv。	(标准值：0.01 == 1%)
0x00000108	读/写	P/PID (加速 度)	REAL64	s	[0.0 ... 100.0]	比例放大系数 ka	加速度预控制
0x0000010A	读/写	每个	UINT32	1	枚举	额定速度最大斜率滤波 (加速度受限)： 0: Off, 1: Velo, 2: Pos+Velo	保留功能，无标准！
0x0000010B	读/写	每个	REAL64	mm/s^2		额定速度最大斜率的滤波值 (最大加速度)	保留功能，无标准！
0x0000010D	读/写	P/PID	REAL64	mm	[0.0 ... 10000.0]	位置误差（位置偏差）的“死区” (适用于带速度或扭矩接口的 P/PID 控制器)	保留功能
0x0000010F	读/写	P/PP/PID (位 置) 从轴控制器	REAL64	(mm/s) / mm	[0.0...1000.0]	从轴耦合控制器： 主轴和从轴之间位置偏差的比例增益 k _{cp}	从轴耦合控制器
0x00000110	读/写	P (位置)	UINT16	1	0/1	自动偏移校准： 主动/被动	
0x00000111	读/写	P (位置)	UINT16	1	0/1	自动偏移校准： 保持模式	
0x00000112	读/写	P (位置)	UINT16	1	0/1	自动偏移校准： 衰减模式	
0x00000114	读/写	P (位置)	REAL64	%	[0.0 ... 1.0]	自动偏移校准： 预控制限制	(标准值：0.05 == 5%)
0x00000115	读/写	P (位置)	REAL64	s	[0.1 ... 60.0]	自动偏移校准： 时间常量	
0x00000116	读/写	PID (位置)	REAL64	%	[0.0...1.0]	I 部分最大输出限制 ()， 以百分比为单位 (默认设置：0.1 == 10 %)	
0x00000117	读/写	PID (位置)	REAL64	%	[0.0...1.0]	D 部分最大输出限制 ()， 以百分比为单位 (默认设置：0.1 == 10 %)	
0x00000118	读/写	PID (位置)	UINT16	1	0/1	在主动定位过程中关闭 I 部分 (只要 I 部分激活) ? (默认设置：0 = FALSE)	
0x00000120	读/写	P/PID (位置)	REAL64	s	≥0	PT-1 位置误差 (位置差) 滤波时间	保留功能，无标准！
0x00000202	读/写	P/PID (速度)	REAL64	1	[0.0...1000.0]	比例放大系数 k _p 或 k _v	速度控制
0x00000203	读/写	PID (速度)	REAL64	s	[0.0 ... 60.0]	积分作用时间 T _n	速度控制
0x00000204	读/写	PID (速度)	REAL64	s	[0.0 ... 60.0]	微分作用时间 T _v	速度控制
0x00000205	读/写	PID (速度)	REAL64	s	[0.0 ... 60.0]	阻尼时间 T _d	速度控制
0x00000206	读/写	PID (速度)	REAL64	%	[0.0...1.0]	I 部分最大输出限制 ()，以百分比为单位 (默认设置：0.1 == 10 %)	速度控制
0x00000207	读/写	PID (速度)	REAL64	%	[0.0...1.0]	D 部分最大输出限制 ()，以百分比为单位 (默认设置：0.1 == 10 %)	速度控制
0x0000020D	读/写	P/PID (速度)	REAL64	mm/s	[0.0 ... 10000.0]	速度误差 (速度偏差) 的“死区” (适用于带速度或扭矩接口的 P/PID 控制器)	保留功能

索引偏移 (十六进制)	访问	控制器类型	数据类型	物理单位	定义范围	描述	备注
0x00000220	读/写	P/PID (速度)	REAL64	s	≥0	PT-2 速度误差 (速度差) 滤波时间	速度控制, 无标准!
0x00000221	读/写	P/PID (速度)	REAL64	s	≥0	PT-1 速度误差 (速度差) 滤波时间	保留功能, 无标准!
0x00000250	读/写	P/PI (观测器)	UINT32	1	参见 枚举 (≥0)	带扭矩接口控制器的观测器模式 [▶ 105] 0: 关闭 (默认) 1: 卢恩伯格 (LUENBERGER)	
0x00000251	读/写	P/PI (观测器)	REAL64	Nm / A	>0.0	电机: 扭矩常数 K_T	
0x00000252	读/写	P/PI (观测器)	REAL64	kg m ²	>0.0	电机: 转动惯量 J_M	
0x00000253	读/写	P/PI (观测器)	REAL64	Hz	[100.0 ... 2000.0] 默认: 500	带宽 f_0	
0x00000254	读/写	P/PI (观测器)	REAL64	1	[0.0 ... 2.0] 默认: 1.0	校正系数 k_c	
0x00000255	读/写	P/PI (观测器)	REAL64	s	[0.0 ... 0.01] 默认: 0.001	速度滤波 (1. 阶) : 滤波时间常数 T	
0x00000A03	读/写	PID (MW)	REAL64	cm ²	[0.0 ... 1000000]	A 面气缸面积 A_{AA} , 单位为 cm ²	
0x00000A04	读/写	PID (MW)	REAL64	cm ²	[0.0 ... 1000000]	B 面气缸面积 A_{AB} , 单位为 cm ²	
0x00000A05	读/写	PID (MW)	REAL64	cm ³ /s	[0.0 ... 1000000]	额定体积流量 Q_{nenn} , 单位为 cm ³ /s	
0x00000A06	读/写	PID (MW)	REAL64	bar	[0.0 ... 1000000]	额定压力或阀门减压 P_{nenn} , 单位为 bar	
0x00000A07	读/写	PID (MW)	UINT32	1	[1 ... 255]	系统压力 p_o 的轴 ID	

6.4.6.2 控制器状态的"索引偏移"规范 (索引组 0x6100 + ID)

索引偏移 (十六进制)	访问	控制器类型	数据类型	物理单位	定义范围	描述	说明
0x00000001	读取	每个	INT32			错误状态控制器	
0x00000002	读取	每个	REAL64	例如 mm/s		控制器输出 (以绝对单位表示)	基本单位 / s 可以进行符号访问! "CtrlOutput"
0x00000003	读取	每个	REAL64	%		控制器输出 (以百分比为单位)	无法用示波器追踪!
0x00000004	读取	每个	REAL64	V		控制器输出 (以伏特为单位)	无法用示波器追踪!
0x0000000D	读取	每个	REAL64	mm		跟随误差位置 (无死区时间补偿)	基本单位
0x0000000E	读取	每个	REAL64	mm		跟随误差位置 (无设置位置校正)	基本单位
0x0000000F	读取	每个	REAL64	mm		跟随误差位置 (带设置位置校正和死区时间补偿)	基本单位 可以进行符号访问! "PosDiff"
0x00000010	读取	每个	REAL64	mm		位置最大负跟随误差的峰值保持值	基本单位
0x00000011	读取	每个	REAL64	mm		位置最小正跟随误差的峰值保持值	基本单位

索引偏移 (十六进制)	访问	控制器类型	数据类型	物理单位	定义范围	描述	说明
0x00000012	读取	每个	REAL64	mm/s		跟随误差速度	基本单位 / s
0x00000021	读取	每个	REAL64	mm		主轴和从轴的跟随误差之间的差 (偏差) (主轴误差减去从轴误差)	基本单位 可通过轴进行符号访问! "PosDiffCouple"
0x00000022	读取	每个	REAL64	mm		主轴和从轴位置跟随误差之间最大负差的峰值保持值	基本单位
0x00000023	读取	每个	REAL64	mm		主轴和从轴位置跟随误差之间最大正差的峰值保持值	基本单位
0x00000101	读取	P/PID (位置)	REAL64	例如 mm/s		控制器的 P 部分 (以绝对单位表示)	
0x00000102	读取	PID (位置)	REAL64	例如 mm/s		控制器的 I 部分 (以绝对单位表示)	
0x00000103	读取	PID (位置)	REAL64	例如 mm/s		控制器的 D 部分 (以绝对单位表示)	
0x00000104	读取	PID (位置)	UINT16	1	0/1	I 部分限制激活?	
0x00000105	读取	PID (位置)	UINT16	1	0/1	D 部分限制激活?	
0x00000106	读取	PID (位置)	UINT16	1	0/1	I 部分 ARW 测量激活?	ARW: 防积分饱和
0x0000010F	读取	P/PP/PID (速度)	REAL64	例如 mm/s		自动偏移补偿比例 (以绝对单位表示)	新
0x00000110	读取	PID (位置)	REAL64	例如 mm/s		控制器的加速度预控制 Y _{acc} (以绝对单位表示) 注: 功能取决于控制器类型!	加速度预控制
0x00000111	读取	PP (位置)	REAL64	mm/s/mm	≥0	内部插值比例增益 kp 或 kv	PP 控制器
0x0000011A 0x0000011B 0x0000011C 0x0000011D 0x0000011E 0x0000011F 0x00000120 0x00000121 0x00000122 0x00000123 0x00000124	读取	P (位置)	uint32 real64 real64 real64 real64 real64 real64 real64 real64 real64 real64 real64 real64 real64	1 mm mm/s mm/s mm/s^2 mm mm mm/s mm/s^2 mm/s mm/s^2		设置速度滤波: InternalPhase InternalPosSollError! TestVeloSoll InternalLimitedVeloSoll InternalAccSollRel InternalPosSollRel PosSollCorrected! VeloSollCorrected! AccSollCorrected! TestVeloSollCorrected TestAccSollCorrected	列表! 保留功能, 无标准!
0x00000201	读取	P,PID (速度)	REAL64	例如 mm/s		控制器的速度部分	基本单位 / s
0x00000202	读取	P,PID (速度)	REAL64	%		控制器的速度部分 (以百分比为单位)	无法用示波器追踪!
0x00000203	读取	P,PID (速度)	REAL64	V		控制器的速度部分 (以伏特为单位)	无法用示波器追踪!
0x00000201	读取	P/PID (速度)	REAL64	例如 mm/s		控制器的 P 部分 (以绝对单位表示)	
0x00000202	读取	P/PID (速度)	REAL64	例如 mm/s		控制器的 I 部分 (以绝对单位表示)	
0x00000203	读取	P/PID (速度)	REAL64	例如 mm/s		控制器的 D 部分 (以绝对单位表示)	

索引偏移 (十六进制)	访问	控制器类型	数据类型	物理单位	定义范围	描述	说明
0x00000204	读取	P/PID (速度)	UINT16	1	0/1	I 部分限制激活?	
0x00000205	读取	P/PID (速度)	UINT16	1	0/1	D 部分限制激活?	
0x00000206	读取	P/PID (速度)	UINT16	1	0/1	I 部分 ARW 测量激活?	ARW: 防积分饱和
0x0000020A	读取	P/PID (速度)	REAL64	例如 mm/s		速度控制器的总输入大小	
0x00000250	读取	P/PI (观测器)	REAL64	例如 mm		观测器: 位置差 (实际位置 - 观测器位置)	
0x00000251	读取	P/PI (观测器)	REAL64	例如 mm		观测器: 位置	
0x00000252	读取	P/PI (观测器)	REAL64	例如 mm/s		观测器: 速度 2 (用于 P 部分)	
0x00000253	读取	P/PI (观测器)	REAL64	例如 mm/s		观测器: 速度 1 (用于 I 部分)	
0x00000254	读取	P/PI (观测器)	REAL64	例如 mm/s^2		观测器: 加速度	
0x00000255	读取	P/PI (观测器)	REAL64	A		观测器: 电机实际电流	
0x00000256	读取	P/PI (观测器)	UINT16	1	0/1	观测器: I 部分限制激活?	
0x00000A00	读取	PID (MW)	REAL64	%	[-1.0...1.0]	计算设置速度 (预控制) (以百分比为单位)	
0x00000A01	读取	PID (MW)	REAL64	例如 mm/s		控制器的 P 部分 (以绝对单位或百分比表示) (根据输出权重计算)	
0x00000A02	读取	PID (MW)	REAL64	例如 mm/s		控制器的 I 部分 (以绝对单位或百分比表示) (根据输出权重计算)	
0x00000A03	读取	PID (MW)	REAL64	例如 mm/s		控制器的 D 部分 (以绝对单位或百分比表示) (根据输出权重计算)	
0x00000A04	读取	PID (MW)	UINT16	1	0/1	I 部分限制激活?	
0x00000A05	读取	PID (MW)	UINT16	1	0/1	D 部分限制激活?	
0x00000A10	读取	PID (位置)	REAL64	例如 mm/s		控制器的加速度预控制 Y_{acc} (以绝对单位表示)	加速度预控制

6.4.6.3 控制器功能的"索引偏移"规范 (索引组 0x6200 + ID)

索引偏移 (十进制)	访问	控制器类型	数据类型	物理单位	定义范围	描述	备注

6.4.7 规范器驱动

6.4.7.1 驱动参数的"索引偏移"规范 (索引组 0x7000 + ID)

索引偏移 (十六进制)	访问	驱动器类型	数据类型	物理单位	定义范围	描述	说明
0x00000001	读取	每个	UINT32	1	[1 ... 255]	驱动器 ID	
0x00000002	读取	每个	UINT8[30+1]	1	30 个字符	驱动器名称	
0x00000003	读取	每个	UINT32	1	参见 枚举 (>0)	驱动器类型 ▶ 109]	
0x00000004	读/写	每个	UINT32	1	Byteoffset	输入地址偏移 (IO-输入-图像)	更改 I/O 地址

索引偏移 (十六进制)	访问	驱动器类型	数据类型	物理单位	定义范围	描述	说明
0x00000005	读/写	每个	UINT32	1	Byteoffset	输出地址偏移 (IO-输出-图像)	更改 I/O 地址
0x00000006	读/写	每个	UINT16	1	[0,1]	电机极性	如果已发出启用控制器的信号，则不允许写入。
0x0000000A	读/写	每个	UINT32	1	参见 枚举 (>0)	驱动模式	默认：1 = STANDARD
0x0000000B	读/写	每个	REAL64	%	[-1.0 ... 1.0]	最小输出限制 (输出限制) (默认设置：-1.0 == -100)	
0x0000000C	读/写	每个	REAL64	%	[-1.0 ... 1.0]	最大输出限制 (输出限制) (默认设置：1.0 == 100%)	
0x0000000D	读取	每个	UINT32	INC		输出增量的最大数量 (输出掩码)	
0x00000010	读/写	每个	UINT32	1		内部驱动控制双字，用于确定驱动运行模式	保留！
0x00000011	读/写	每个	UINT32	1	≥ 5	内部驱动复位计数器 (启用和复位的 NC 循环时间)	保留！
0x00000020	读/写	每个	UINT32	1	参见 枚举 (>0) 参见附录	驱动死区时间补偿模式 0: 关闭 (默认) 1: 开 (有速度) 2: 开 (有速度和加速度)	
0x00000021	读/写	每个	UINT32	1		用于驱动死区时间补偿的控制双字 (32 位)： 位 0=0: 相对 IO 时间 (默认) 位 0=1: 绝对 IO 时间	
0x00000022	读/写	每个	INT32	ns	[±1.0E+9]	用于驱动死区时间补偿的参数化时间偏移之和 (通常为正值)	
0x00000031	读/写	每个	REAL64	例如 %/ INC	[-1.0E+30 ... 1.0E+30]	驱动器实际扭矩值的缩放因子 (scaling factor) (或分别为力或电流的实际值) 例如，AX5xxx: 0.1 => ±100%	从 TC3.1 起更新
0x00000032	读/写	每个	REAL64	s	[0.0 ... 60.0]	P-T1 实际扭矩值滤波时间 (或分别为力或电流的实际值)	从 TC3.1 起更新
0x00000033	读/写	每个	REAL64	s	[0.0 ... 60.0]	实际扭矩值时间导数的 P-T1 滤波时间 (或分别为力或电流的实际值)	从 TC3.1 起更新
0x00000101	读/写	伺服	REAL64	例如 mm/s	>0.0	参考输出的参考速度 (速度预控制)	基本单位 / s
0x00000102	读/写	伺服	REAL64	%	[0.0 ... 5.0]	参考输出 (以百分比为单位)	
0x00000103	读取	伺服	REAL64	例如 mm/s	>0.0	100% 输出时产生的速度	基本单位 / s
0x00000104	读/写	伺服	REAL64	例如 mm/s	±∞	轴漂移校准 (偏移校准) 的速度偏移 (DAC 偏移)	基本单位 / s

索引偏移 (十六进制)	访问	驱动器类型	数据类型	物理单位	定义范围	描述	说明
0x000000105	读/写	伺服 (Sercos、Profi Drive、AX200x、CANopen)	REAL64	1	[0.0 ... 100000000.0]	速度缩放 (响应驱动器中权重的缩放因子 (scaling factor))	适用于 Sercos、Profi Drive、AX200x、CANopen
0x000000106	读/写	Profi Drive DSC	UINT32	0.001 * 1/s	≥ 0	Profibus/Profi Drive DSC: 位置控制增益 Kpc	仅适用于 Profi Drive DSC
0x000000107	读/写	Profi Drive DSC	REAL64	1	≥ 0.0	Profibus/Profi Drive DSC: 用于计算 "XERR" 的缩放比例 (默认: 1.0)	仅适用于 Profi Drive DSC
0x000000109	读/写	伺服	REAL64	1	[0.0 ... 100000000.0]	位置缩放 (响应驱动器中权重的缩放因子 (scaling factor))	适用于 Sercos、CANopen
0x00000010A	读/写	伺服	REAL64	1	[0.0 ... 100000000.0]	加速度缩放 (响应驱动器中权重的缩放因子 (scaling factor))	适用于 Sercos、Profi Drive、AX200x、CANopen
0x00000010B	读/写	伺服	REAL64	1	[0.0 ... 100000000.0]	适用于 "TorqueOffset" 的扭矩缩放 (旋转电机) 或力缩放 (直线电机) (响应驱动器中权重的缩放因子 (scaling factor)) (作为预控制的加法力矩)	适用于 Sercos、Profi Drive、AX200x、CANopen
0x00000010C	读/写	伺服	REAL64	1	[0.0 ... 100000000.0]	适用于 "SetTorque" 的扭矩缩放 (旋转电机) 或力缩放 (直线电机) (响应驱动器中重量的缩放因子 (scaling factor)) (例如 MC_TorqueControl, 使用驱动器运行模式 CST)	适用于 Sercos、Profi Drive、AX200x、CANopen TC 3.1 B4024.2 及以上
0x00000010D	读/写	伺服 (Sercos、CANopen)	REAL64	s	[0.0 ... 1.0]	驱动速度输出的阻尼时间	适用于 Sercos、CANopen
0x00000010E	读/写	伺服 (Sercos、CANopen)	REAL64	s	[0.0 ... 1.0]	驱动加速度输出的阻尼时间	适用于 Sercos、CANopen
0x00000010F	读/写	伺服 (Sercos、CANopen)	REAL64	s	[0.0 ... 1.0]	驱动扭矩输出或力输出的阻尼时间	适用于 Sercos、CANopen
0x000000120	读/写	伺服/液压/	UINT32	1	≥ 0	表 ID (0: 无表)	仅适用于 KL4xxx、M2400、通用型
0x000000121	读/写	伺服/液压	UINT32	1	≥ 0	插值类型 0: 线性 2: 样条	仅适用于 KL4xxx、M2400、通用型
0x000000122	读/写	伺服/液压	REAL64	%	[-1.0 ... 1.0]	输出偏移 (以百分比为单位) 注: 根据特征评估进行操作!	仅适用于 KL4xxx、M2400、通用型
0x000000151	读/写	伺服/非线性	REAL64	1	[0.0 ... 100.0]	象限补偿系数 (I 和 III 象限之间的关系)	
0x000000152	读/写	伺服/非线性	REAL64	1	[0.01 ... 1.0]	速度参考点 (以百分比为单位) (1.0 == 100)	

索引偏移 (十六进制)	访问	驱动器类型	数据类型	物理单位	定义范围	描述	说明
0x000000153	读/写	伺服/非线性	REAL64	1	[0.01 ... 1.0]	输出参考点 (以百分比为单位) (1.0 == 100%)	
0x000000301	读/写	步进电机	UINT8			位掩码: 循环 1	
0x000000302	读/写	步进电机	UINT8			位掩码: 循环 2	
0x000000303	读/写	步进电机	UINT8			位掩码: 循环 3	
0x000000304	读/写	步进电机	UINT8			位掩码: 循环 4	
0x000000305	读/写	步进电机	UINT8			位掩码: 循环 5	
0x000000306	读/写	步进电机	UINT8			位掩码: 循环 6	
0x000000307	读/写	步进电机	UINT8			位掩码: 循环 7	
0x000000308	读/写	步进电机	UINT8			位掩码: 循环 8	
0x000000310	读/写	步进电机	UINT8			位掩码: 保持电流	

6.4.7.2 驱动器状态的"索引偏移"规范 (索引组 0x7100 + ID)

索引偏移 (十六进制)	访问	驱动器类型	数据类型	物理单位	定义范围	描述	说明
0x000000001	读取	每个	INT32			错误状态驱动器	
0x000000002	读取	每个	REAL64	例如 mm/s		总输出 (以绝对单位表示)	基本单位 / s 可进行符号访问! "DriveOutput"
0x000000003	读取	每个	REAL64	%		总输出 (以百分比为单位)	
0x000000004	读取	每个	REAL64	V		总输出 (以伏特为单位)	无法用示波器追踪!
0x000000005	读取	每个	REAL64	例如 mm/s		最大负总输出的峰值保持值	基本单位 / s
0x000000006	读取	每个	REAL64	例如 mm/s		最大正总输出的峰值保持值	基本单位 / s
0x000000007	读取	每个	REAL64	例如 100% = 1000, 例如 Nm 或 N		分别为实际扭矩或实际力 (通常 100% = 1000)	TC3.1 B4022 及以上 可进行符号访问! "ActTorque"
0x000000008	读取	每个	REAL64	例如 Nm/s 或 N/s	±∞	分别为实际扭矩变化或实际力变化 (分别为实际扭矩或实际力的时间导数)	TC3.1 B4024 及以上
0x00000000C	读取	每个	REAL64	例如 mm		设置由于死区时间补偿产生的驱动输出的位置校正值	
0x00000000D	读取	每个	REAL64	s		驱动死区时间补偿的时移总和 (参数化和可变死区时间) 注: 系统中规定的死区时间为正值。	
0x000000013	读取	每个	REAL64	%		总输出 (以百分比为单位) (基于非线性特征曲线!)	
0x000000014	读取	每个	REAL64	V		总输出 (以伏特为单位) (基于非线性特征曲线!)	无法用示波器追踪!

索引偏移 (十六进制)	访问	驱动器类型	数据类型	物理单位	定义范围	描述	说明
0x00000011A	读取	伺服 (Sercos、CANopen)	REAL64	例如 mm		可选输出滤波：滤波设置位置	新适用于 Sercos、CANopen
0x00000011E	读取	伺服 (Sercos、CANopen)	REAL64	例如 mm/s		可选输出滤波：滤波设置速度	新适用于 Sercos、CANopen
0x00000011F	读取	伺服 (Sercos、CANopen)	REAL64	例如 mm/s^2		可选输出滤波：滤波设置加速度/设置减速度	新适用于 Sercos、CANopen
0x000000200	读写		读取：			读取数字输入 1 至 8 的状态	TC3.1 B4024.12 及以上
			UINT32	1	0/1	所选输入的状态	仅适用于 SAF-Port 501!
			写入：				
			UINT32	1	[1…8]	选择输入 1 至 8	

6.4.7.3 驱动器功能的"索引偏移"规范 (索引组 0x7200 + ID)

索引偏移 (十 六进制)	访问	驱动器类型	数据类型	物理单位	定义范围	描述	备注
0x000000102	写入	伺服	{ ULONG 1 >0 }			移除和删除特征驱动表 表 ID S.A. 轴功能，带索引偏移 0x00000012	仅适用于 SAF 端口 501!

6.4.7.4 循环驱动过程数据的"索引偏移"规范 (索引组 0x7300 + ID)

索引偏移 (十 六进制)	访问	驱动器类型	数据类型	物理单位	定义范围	描述	备注
0x000000000	读/写	每个 (NC→IO)	{ INT32 INC ≥ 0 nOutData1 INT32 INC 2^31 nOutData2 UINT8 1 ≥ 0 nControl1 UINT8 1 ≥ 0 nControl2 UINT8 1 ≥ 0 nControl3 UINT8 1 ≥ 0 nControl4 INT32 INC ≥ 0 nOutData3 INT32 INC ≥ 0 nOutData4 INT32 INC ≥ 0 nOutData5 INT32 INC ≥ 0 nOutData6 UINT8 1 ≥ 0 nControl5 UINT8 1 ≥ 0 nControl6 UINT8 1 ≥ 0 nControl7 UINT8 1 ≥ 0 nControl8 INT32 1 ≥ 0 保留 INT32 1 ≥ 0 保留 }		STRUCT 参见驱动器接口 驱动器-输出-结构 (NC→IO, 40 字节) NCDRIVESTRUCT_OUT2	写入命令仅为可选项! 考虑安全方面!	
0x000000001	写入	每个 (NC→IO)	{ UINT32 1 [0 … 39] ByteOffset			逐位访问驱动器-输出-结构 (NC→IO, 40 字节) NCDRIVESTRUCT_OUT2	写入命令仅为可选项! 考虑安全问题

索引偏移 (十 六进制)	访问	驱动器类型	数据类型	物理单位	定义范围	描述	备注
						输出结构中的相对地址偏移 [0..39]。 例如：要写入 "nControl1"， ByteOffset 必须为 8。	
			UINT32	1	[0x00000000... 0xFFFFFFFF]	BitSelectMask (BSM) 掩码在 DWORD 中定义允许写入的位。零位受保护，不受影响。	
			UINT32	1	[0x00000000... 0xFFFFFFFF]	数值 只有 BSM 等于 1 的值位 才会被重写。	
			}				
0x00000080	读取	每次 (IO→NC)	{		STRUCT 参见驱 动器接口	驱动器-输入-结构 (IO→NC, 40 字节) <i>NCDRIVESTRUCT_IN2</i>	
			INT32	INC	≥ 0	nInData1	
			INT32	INC	≥ 0	nInData2	
			UINT8	1	≥ 0	nStatus1	
			UINT8	1	≥ 0	nStatus2	
			UINT8	1	≥ 0	nStatus3	
			UINT8	1	≥ 0	nStatus4	
			INT32	INC	≥ 0	nInData3	
			INT32	INC	≥ 0	nInData4	
			INT32	INC	≥ 0	nInData5	
			INT32	INC	≥ 0	nInData6	
			UINT8	1	≥ 0	nStatus5	
			UINT8	1	≥ 0	nStatus6	
			UINT8	1	≥ 0	nStatus7	
			UINT8	1	≥ 0	nStatus8	
			INT32	1	≥ 0	保留	
			INT32	1	≥ 0	保留	
			}				

6.4.8 表规范

6.4.8.1 表参数的"索引偏移"规范 (索引组 0xA000 + ID)

索引偏移 (十 六进制)	访问	表类型	数据类型	物理单位	定义范围	描述	备注
0x00000001	读取	每个	UINT32	1	[1 ... 255]	表 ID	
0x00000002	读取	每个	UINT8[30+1]	1	30 个符号	表格名称	
0x00000003	读取	每个	UINT32	1	参见 枚举 (>0)	表格子类型 [▶ 111]	
0x00000004	读取	每个	UINT32	1	参见 枚举 (>0)	表主类型 [▶ 111]	
0x00000010	读取	每个	UINT32	1	[0... 16777216]	行数 (n)	
0x00000011	读取	每个	UINT32	1	[0... 16777216]	列数 (m)	
0x00000012	读取	每个	UINT32	1	≥0	元素总数 (n*m)	
0x00000013	读取	等距表	REAL64	例如 mm	≥0.0	步宽 (位置增量) (等距 表)	基本单位
0x00000014	读取	循环表	REAL64	例如度数	≥0.0	主轴周期 (循环表)	基本单位
0x00000015	读取	循环表	REAL64	例如度数	≥0.0	每个主轴周期 (循环表) 的从轴差	基本单位

索引偏移（十 六进制）	访问	表类型	数据类型	物理单位	定义范围	描述	备注
0x00000001A	读/写	"运动功能"	{			从表格数据进行在线更改的启动模式（仅限 MF）	
			UINT32	枚举	参见附录	激活模式： 0: '瞬时的 (instantaneous)' (默认) 1: '主凸轮的位置.' 2: '主轴的位置.' 3: '下一个周期' 4: '下个周期只循环一次' 5: '尽快' 6: '停止' 7: '删除待处理的数据'	
			REAL64	例如 mm	±∞	激活位置	
			UINT32	枚举	参见附录	主轴缩放类型 0: 用户定义 (默认) 1: 使用自动偏移缩放 2: 关闭	
			UINT32	枚举	参见附录	从轴缩放类型 0: 用户定义 (默认) 1: 使用自动偏移缩放 2: 关闭	
			}				
0x000000020	读/写	每个	{			写入单个值 [n,m]:	
			UINT32	1	[0 ... 16777216]	第 n 行	
			UINT32	1	[0 ... 16777216]	第 m 列	
			REAL64	例如 mm	±∞	单个值	基本单位
			}				
0x000000021	读写	每个	*REAL64	例如 mm	±∞	将从轴位置读取到给定主轴位置（只与表格的“行值”有关）	
0x000000022	读写	"运动功能"	写入			将“运动函数”读取为固定值（“散点图”）	只能逐行！（多 个整数）
			{				
			UINT16	1	0/1	数据初始化（实际数据的副本）？	
			UINT16	1	位掩码 (≥ 0)	选择位掩码（列数为一个主轴位置列加位数）： 位 0: Pos (从轴) 位 1: 速度 (从轴) 位 2: 加速度 (从轴) 位 3: 加加速度 (从轴)	
			REAL64	例如 mm	±∞	Startposition (主轴)	
			REAL64	例如 mm	> 0.0	Inkrement	
			}				
			读取				
			{				
			REAL64[x*m]	例如 mm	±∞	从主轴起始位置开始生成 x 行的值：(x*m)-值（一行或多行）	
			}				
0x000000023	读写	每个	写入			将从轴值读取到给定主轴位置（仅与表的“行值”有关）	

索引偏移 (十六进制)	访问	表类型	数据类型	物理单位	定义范围	描述	备注
			REAL64	例如 mm	$\pm\infty$	主轴位置	
			读取				
			{				
			REAL64	例如 mm	$\pm\infty$	从轴位置	
			REAL64	mm/s	$\pm\infty$	从轴速度	
			REAL64	mm/s ²	$\pm\infty$	从轴加速度	
			}				
0x00000050	读/写	每个	REAL64 [64]	1	$\pm\infty$	特性表值 [▶ 112]	
0x00000050	读写	每个	写入			读取与额定主轴速度相关的特性表值	
			REAL64 [64]	...	$\pm\infty$	可选额定主轴速度 "fMasterVeloNom" (标准化为 => 1.0 mm/s), 不使用其他值	
			读取				
			REAL64 [64]	...	$\pm\infty$	读取特性表值 [▶ 112]	
0x00000115	写入	单调线性、单调循环	{			设置/更改表缩放比例:	
			REAL64	1	[± 1000000.0]	表的原始权重	
			REAL64	例如 mm	[± 1000000.0]	主轴列位置偏移	
			REAL64	1	[± 1000000.0]	主轴列缩放	
			REAL64	例如 mm	[± 1000000.0]	从轴列位置偏移	
			REAL64	1	[± 1000000.0]	从轴列缩放	
			REAL64	例如 mm	[± 1000000.0]	下部区域边界 (起始位置)	
			REAL64	例如 mm	[± 1000000.0]	上部区域边界 (结束位置)	
			}				
0x01000000 +n-te 起始行	读/写 [≤ 16777216]	每个	{ REAL64[x*m] }	例如 mm	$\pm\infty$	从第 n 行读取/写入 x 行: (x*m)-values (一行或多行) 值范围 n: [0 ... 16777216]	只能逐行! (多个整数)
0x02000000 +m-te Startspold	读/写 [≤ 16777216]	每个	{ REAL64[x*n] }	例如 mm	∞	从第 m 列读取/写入 x 列: (x*n)-values (一列或多列) 值范围 m: [0 ... 16777216]	只能逐列! (多个整数)
0x05000000 +n-te 起始行	读/写 [≤ 16777216]	"运动函数" (运动定律) 数据: STRUCT[x*m]	{			从第 n 行开始读/写 x 行: (x*m)-values (一行或多行) 值范围 n: [0 ... 16777216]	只能逐行! (多个整数)
			UINT32	1		绝对点索引 (未检查)	
			UINT16	枚举		函数类型 1: 多项式 1 15: 多项式 5	
			UINT16	枚举		点类型 0: 默认 1: 忽略	
			INT32	1		目标点的相对地址索引 (默认: 1)	
			REAL64	mm		主轴位置	
			REAL64	mm		从轴位置	
			REAL64	mm/s		从轴速度	

索引偏移 (十 六进制)	访问	表类型	数据类型	物理单位	定义范围	描述	备注
			REAL64	mm/s^2		从轴加速度	
			REAL64	mm/s^3		从轴加加速度	
			}				
0x06000000 +m-te Startspold	读/写 [<=16777216]	"运动函数" (运 动定律) 数据: STRUCT[x*n]	{			从第 m 列读取/写入 x 列: (x*n)-values (一列 或多列) 值范围 m: [0 ... 16777216]	只能逐列! (多 个整数)
			UINT32	1		绝对点索引 (未检查)	
			UINT16	枚举		函数类型 1: 多项式 1 15: 多项式 5	
			UINT16	枚举		点类型 0: 默认 1: 忽略	
			INT32	1		目标点的相对地址索引 (默认: 1)	
			REAL64	mm		主轴位置	
			REAL64	mm		从轴位置	
			REAL64	mm/s		从轴速度	
			REAL64	mm/s^2		从轴加速度	
			REAL64	mm/s^3		从轴加加速度	
			}				

6.4.8.2 表状态的"索引偏移"规范 (索引组 0xA100 + ID)

索引偏移 (十 六进制)	访问	表类型	数据类型	物理单位	定义范围	描述	备注
0x0000000A	读取	每个	INT32	1	≥ 0	"用户计数器" (表用户数量)	无法用示波器追 踪!

6.4.8.3 表功能的"索引偏移"规范 (索引组 0xA200 + ID)

索引偏移 (十 六进制)	访问	表类型	数据类型	物理单位	定义范围	描述	备注
0x00010000	写入	每个	{			生成尺寸为 (n*m) 的表 格:	表类型: 1、2、 3、4 尺寸: 至 少 2x1
			UINT32	1	参见 枚举 (>0)	表类型 [► 111] (参见附 录)	
			UINT32	1	[2...16777216]	行数量	
			UINT32	1	[1...16777216]	列数量	
			}				
0x00010001	写入	阅图	{			生成尺寸为 (n*m) 的阅图 表:	表类型: 1,3 尺 寸: 至少 2x1
			UINT32	1	参见 枚举 (>0)	表类型 [► 111] (参见附 录)	
			UINT32	1	[2...16777216]	行数量	
			UINT32	1	[1...16777216]	列数量	
			}				
0x00010010	写入	"运动函数" (运动定律)	{			生成尺寸为 (n*m) 的"运 动函数"表:	表类型: 3、4 尺寸: 至少 2x1
			UINT32	1	参见 枚举 (>0)	表格类型 (参见附录)	
			UINT32	1	[2...16777216]	行数量	

索引偏移 (十 六进制)	访问	表类型	数据类型	物理单位	定义范围	描述	备注
			UINT32	1	[1...16777216]	列数量	
			}				
0x00020000	写入	每个	VOID			删除尺寸为 (n*m) 的表	表类型: 1,2,3,4
0x00030000	写入	每个	VOID			初始化表格 初始化不再需要手动进行了, 因为现在在以下情况下会自动进行初始化: a) 与表格关联时 b) 耦合; b) 选择从位置时 (参见表格参数)	

6.4.9 附录

枚举通道类型

定义	通道类型
1	标准
2	解释器
3	FIFO
4	运动转换

枚举解释器类型

定义	解释器类型
0	未定义
1	NC 解释器 DIN 66025 (GST)
2	NC 解释器 DIN 66025 (经典方言)

枚举解释器运行模式

定义	解释器/通道运行模式
0x0	默认 (停用其他模式)
0x1	NC 核中的单块模式 (块执行任务/SAF)
0x1000	保留
0x2000	保留
0x4000	解释器中的单块模式

枚举插值加载日志模式

定义	加载日志模式
0	关闭加载程序日志
1	仅来源
2	来源与编译

枚举插值追踪模式

定义	追踪模式
0	关闭追踪
1	追踪行数
2	追踪源

枚举解释器状态

已移至: 解释器的系统管理器界面 — 解释器元素

枚举组类型

定义	组类型
0	未定义
1	PTP 组 + x 从轴
2	1D-Group + x 从轴
3	2D-Group + x 从轴
4	3D-Group + x 从轴
5	高速/低速 + x 从轴
6	低成本步进电机 (数字 IO) + x 从轴
7	表组 + x 从轴
9	编码器组 + x 从轴
11	FIFO 组 + x 从轴
12	运动转换组 + x 从轴

枚举曲线速度减小法

已移至：解释器的系统管理器界面 – 组元素

枚举轴类型

定义	轴类型
0	未定义
1	连续轴 (伺服)
2	离散轴 (高速/低速)
3	连续轴 (步进电机)
5	编码器轴
6	连续轴 (带运行模式开关，用于位置/压力控制)
7	时基发生器
100	

枚举步进电机运行模式

定义	步进电机运行模式
0	未定义
1	2 相励磁 (4 个循环)
2	1-2 相励磁 (6 个循环)
3	动力部分

PTP 轴枚举超驰类型 (速率比)

定义	超驰类型
1	减少 旧版本，已被"(3) 减少 (迭代) "替代
2	原始 旧版本，已被"(4) 原始 (迭代) "替代
3	减少 (迭代) 默认值：速率比值与特殊情况下会在内部降低的速度有关。这导致从 0% 到 100% 的整个速率比范围速度成正比 (=> 线性关系)。
4	原始 (迭代) 速率比值始终是指用户编程的速度。但是，如果无法驱动此速度，则会得出一个最大速率比值，超过此值无法达到更高速度 (=> 限制)。

枚举组/轴启动类型

定义	组/轴启动类型
0	未定义

定义	组/轴启动类型
1	绝对启动
2	相对启动
3	连续正启动
4	连续负启动
5	模除启动 (旧)
261	模除从最短距离启动
517	模除在正方向上启动 (带模除公差窗口)
773	模除在负方向上启动 (带模除公差窗口)
4096	停止并锁定 (轴锁定以执行运动命令)
8192	停止 (无运动锁定)

通用轴启动 (UAS) 的枚举命令缓冲类型 (缓冲模式)

定义	缓冲模式
0	终止 (默认) (瞬时, 终止当前运动并删除任何缓冲命令)
1	缓冲 (存储在命令缓冲中, 在主动运动后执行)
18	低混合 (缓冲, 无停止, 以两个命令的最低速度运行通过中间目标位置)
19	混合上一个 (缓冲, 无停止, 以主动命令的速度运行通过中间目标位置)
20	混合下一个 (缓冲, 无停止, 以缓冲命令的速度运行通过中间目标位置)
21	混合高 (缓冲, 无停止, 以两个命令的最高速度运行通过中间目标位置)

枚举结束位置类型 (新结束位置)

定义	结束位置类型
0	未定义
1	绝对位置
2	相对位置
3	连续正位置
4	连续负位置
5	模除位置

新速度下新结束位置的枚举命令类型 (新结束位置和/或新速度)

定义	新速度下新结束位置的命令类型
0	未定义
1	位置 (瞬时)
2	速度 (瞬时)
3	位置和速度 (瞬时)
9	位置 (切换位置)
10	速度 (切换位置)
11	位置和速度 (切换位置)

枚举实际位置类型 (设置实际位置)

定义	实际位置类型
0	未定义
1	绝对位置
2	相对位置
5	模除位置

枚举补偿类型（区间补偿或叠加）

定义	补偿类型
0	未定义
1	veloreduction_additivemotion 减小最大速度 VelocityDiff。补偿行程生效路径由长度和距离组成。
2	veloreduction_limitedmotion 减小最大速度 VelocityDiff。补偿行程生效路径由长度参数定义。
3	LENGTHREDUCTION_ADDITIVEMOTION 最大可用路径减小并由长度和距离组成。系统会尽量利用最大速度。VelocityDiff.
4	LENGTHREDUCTION_LIMITEDMOTION 最大可用路径减小并受到长度参数的限制。系统会尽量利用最大速度。VelocityDiff.

枚举从轴类型

定义	从轴类型
0	未定义
1	线性
2	飞锯（速度、加加速度受限配置文件）
3	飞锯（位置和速度、加加速度受限配置文件）
5	同步发生器（速度、加加速度受限配置文件）
6	同步发生器（位置和速度、加加速度受限配置文件）
10	表格
11	多表格
13	“运动功能”（MF）
15	线性，带有循环性齿轮系数变化（加速度限制的斜坡滤波）
100	特定

枚举从轴去耦类型（用于后续轴命令）

定义	从轴解耦类型（用于后续轴命令）
0	停止、E-stop 或 P-stop（默认） (STOP)
1	定向停止(O-stop) (ORIENTEDSTOP)
2	将任何加速度减至 0（无动力），并继续移动到无限远的目标位置 (ENDLESS)
3	以新请求速度继续移动到无限远的目标位置 (endless_newvelo)
4	新结束位置 (NEWPOS)
5	新结束位置和新请求速度 (NEWPOSANDVELO)
6	逻辑解耦并立即停止轴，无速度斜坡 (INSTANTANEOUSSTOP)

枚举控制器类型

定义	控制器类型
0	未定义
1	P 控制器（标准） (位置)
2	PP 控制器（带 ka） (位置)

定义	控制器类型
3	PID 控制器 (带 ka) (位置)
5	P 控制器 (速度)
6	PI 控制器 (速度)
7	高速/低速控制器 (位置)
8	步进电机控制器 (位置)
9	SERCOS 控制器 (驱动器中的位置)
10	保留
11	保留
12	保留
13	保留
14	TCom 控制器 (软驱动) (驱动器中的位置)

枚举控制器观测器模式

定义	控制器观测器模式
0	未启用观测器 (默认)
1	"卢恩伯格 (Luenberger) " 观测器 (经典观测器设计)

枚举编码器类型

定义	编码器类型
0	未定义
1	模拟编码器 (增量)
2	M3000 编码器 (单圈/多圈) (绝对)
3	M31x0 / M2000 编码器 (增量)
4	MDP 511 编码器: EL7041、EL7342、EL5101、EL5151、 EL2521、EL5021、IP5101 (增量)
5	MDP 500/501 编码器: EL5001、IP5009、KL5001 (SSI) (绝对)
6	MDP 510 编码器: KL5051、KL2502-30K 编码器 (BiSSI) (增量)
7	KL30xx 编码器 (模拟) (绝对)
8	SERCOS 和 EtherCAT SoE (位置) (增量)
9	SERCOS 和 EtherCAT SoE (位置和速度) (增量)
10	二进制编码器 (0/1) (增量)
11	M2510 编码器 (绝对)
12	FOX50 编码器 (绝对)

定义	编码器类型
14	AX2000 (Lightbus) (增量)
15	Provi-Drive MC (Simodrive 611U) (增量)
16	通用编码器 (可变位掩码) (增量)
17	NC 后面板 (增量)
18	特殊 CANopen 类型 (如 Lenze Drive 9300) (增量)
19	MDP 513 (DS402): CANopen 和 EtherCAT CoE (AX2xx-B1x0/ B510、EL7201) (增量)
20	AX2xx-B900 (以太网) (增量)
21	KL5151 编码器 (增量)
24	IP5209 编码器 (增量)
25	KL2531/KL2541 编码器 (步进电机) (增量)
26	KL2532/KL2542 编码器 (DC 电机) , KL2535/KL2545 (PWM 电 流端子) (增量)
27	时基编码器 (时基发生器) (增量)
28	TCom 编码器 (软驱动) (增量)

编码器模式

定义	编码器模式
0	未定义
1	确定位置
2	确定位置和速度
3	确定位置、速度和加速度

枚举编码器评估方向 (日志记录计数方向)

定义	编码器评估方向 (日志记录计数方向)
0	正负计数方向评估 (默认配置, 即与先前状态兼容)
1	仅按正计数方向进行评估
2	仅按负计数方向进行评估
3	既不按正计数方向也不按负计数方向进行评估 (评估受阻)



不适用于所有类型的编码器; 仅适用于 KL5101、KL5151、KL2531、KL2541、IP5209、通用编码器等。

编码器评估方向 (日志记录计数方向)	编码器类型		
	KL5101, ...	通用编码器	其他类型
0: 正和负	✓	✓	—
1: 仅正	✓	✓	—
2: 仅负	✓	✓	—

编码器类型			
编码器评估方向 (日志记录计数方向)	KL5101, ...	通用编码器	其他类型
3: 受阻	√	√	—

枚举编码器符号解释 (数据类型)

定义	编码器实际增量的符号解释 (数据类型)
0	未定义 (默认配置, 即与先前状态兼容)
1	无符号: 编码器实际增量的无符号解释
2	有符号: 编码器实际增量的有符号解释



目前仅适用于 KL30xx/KL31xx

枚举编码器绝对尺寸系统

定义	编码器绝对尺寸系统
0	INC: 带下溢和上溢偏移的增量绝对尺寸系统 (默认设置, 即与先前状态兼容)
1	ABS: 无下溢和上溢偏移的绝对尺寸系统 (不允许编码器下溢或上溢)
2	ABS MODULO: 有条件的绝对尺寸系统, 因为它有下溢和上溢偏移 (绝对值通过模运算 (无限循环) 继续)



不适用于所有编码器类型; 仅适用于 Profi Drive MC、M3000、KL5001/EL5001、IP5009、SERCOS、UNIVERSAL 等。

增量式编码器枚举参考模式

定义	参数文本	增量式编码器参考模式
0	默认值	未定义 (默认分配, 即与先前状态兼容)
1	仅归位传感器 (PLC 凸轮或数字输入)	锁存器事件: 关闭 PLC 凸轮 (负边沿)
2	硬件同步 (反馈参考脉冲)	锁存器事件: 硬件同步脉冲 (零轨)
3	硬件锁存器 1 (位置边沿)	锁存器事件: 带正边沿的外部硬件锁存器 (带正边沿的测量探头或各自地即时测量)
4	硬件锁存器 1 (负边沿)	锁存器事件: 带负边沿的外部硬件锁存器 (带负边沿的测量探头或各自地即时测量)
5	软件同步	锁存器事件: 合成模拟软件同步脉冲 (软件零轨); 前提条件: 每电机转速绝对值, 例如旋转变压器!
6	硬件锁存器 1 (位置边沿), 驱动器已定义	锁存器事件: 驱动器中定义的具有正边沿的硬件锁存器事件 (例如 SoftDrive)
7	硬件锁存器 1 (负边沿), 驱动器已定义	锁存器事件: 驱动器中定义的具有负边沿的硬件锁存器事件 (例如 SoftDrive)
20	应用程序 (PLC 代码)	用户特定的引用实现 (PLC 代码): 通过 ApplicationRequest 位向 PLC 发送用户请求信号

: 锁存器事件						
编码器类型	0: 未定义	1: PLC 凸轮 (负边沿)	2: 硬件同步脉冲 (零/C 轨)	3: 带正边沿的外部硬件锁存器	4: 带负边沿的外部硬件锁存器	5: 软件同步脉冲 (软件零轨)
AX2xxx-B200 (Lightbus)	—	√	√	√	√	√ (仅旋转变压器)
AX2xxx-B510 (CANopen)	—	√	—	—	—	√ (仅旋转变压器) (请参见"参考掩码"参数)
AX2xxx-B1x0 (EtherCAT)	—	√	√	√	√	√ (仅旋转变压器) (固定 20 位)

		: 锁存器事件				
编码器类型	0: 未定义	1: PLC 凸轮 (负边沿)	2: 硬件同步脉冲 (零/C 轮)	3: 带正边沿的外部硬件锁存器	4: 带负边沿的外部硬件锁存器	5: 软件同步脉冲 (软件零轨)
AX2xxx-B900 (以太网)	—	√	√	√	√	√ (仅旋转变压器)
Sercos	—	√	√	√ (特定于 AX5xxx 执行)	√	√ (请参见"参考掩码"参数)
Profi Drive	—	√	√	√	√	√
KL5101 IP5109	—	√	√	√	√	√
KL5111	—	√	√	—	—	√
KL5151	—	√	√	√	√	√ (无意义)
IP5209	—	√	√	—	—	√ (无意义)
CANopen (如 Lenze)	—	√	—	√ (输入 E1)	√ (输入 E2)	√ (仅旋转变压器) (固定 16 位)
其他类型	—	—	—	—	—	—

枚举归位传感器源

该参数设置参考凸轮 (归位传感器) 的数字输入源。同时确定信号是高电平有效还是低电平有效。

定义	参数文本	归位传感器源
0	默认: PLC 凸轮 (MC_Home)	参考凸轮由 PLC 提供。输入 MC_Home 功能块的 bCalibrationCam。
1	数字输入 1 (高电平有效), 设备相关映射	驱动器->输入->nState8.bit0 或 MDP703/733 设备的 E1, 例如 7031,7041,7201,7411
2	数字输入 2 (高电平有效), 设备相关映射	驱动器->输入->nState8.bit1 或 MDP703/733 设备的 E2, 例如 L7031,7041,7201,7411
3	数字输入 3 (高电平有效)	驱动器->输入->nState8.bit2
4	数字输入 4 (高电平有效)	驱动器->输入->nState8.bit3
5	数字输入 5 (高电平有效)	驱动器->输入->nState8.bit4
6	数字输入 6 (高电平有效)	驱动器->输入->nState8.bit5
7	数字输入 7 (高电平有效)	驱动器->输入->nState8.bit6
8	数字输入 8 (高电平有效)	驱动器->输入->nState8.bit7
9	数字输入 1 (低电平有效), 设备相关映射	驱动器->输入->nState8.bit2
10	数字输入 2 (低电平有效), 设备相关映射	驱动器->输入->nState8.bit0 或 MDP703/733 设备的 E1, 例如 L7031、7041、7201、7411
11	数字输入 3 (低电平有效)	驱动器->输入->nState8.bit1 或 MDP703/733 设备的 E2, 例如 L7031、7041、7201、7411
12	数字输入 4 (低电平有效)	驱动器->输入->nState8.bit2
13	数字输入 5 (低电平有效)	驱动器->输入->nState8.bit3
14	数字输入 6 (低电平有效)	驱动器->输入->nState8.bit4
15	数字输入 7 (低电平有效)	驱动器->输入->nState8.bit5
16	数字输入 8 (低电平有效)	驱动器->输入->nState8.bit6

数字输入 [1-8]

使用与 NC 程序相连的数字输入。为此, 在过程映像 (Drive->Inputs->nState8) 中定义了一个带有 8 个数字输入的通用驱动器状态字节, 可作为归位传感器的信号源。因此, 必须手动将要使用的数字输入映射到该字节中的所需位置。



数字输入 1 和 2 可能因使用的硬件而异。对于 MDP703/733 硬件（例如 EL7031、EL7041、EL7201、EL7411），则使用端子模块的直接数字输入 E1 和 E2，它们位于端子模块 Drive.nState2 字节的第 3 位 (E1) 和第 4 位 (E2) 位置。在这种情况下，Drive.nState8 的两个低位不会分配。

枚举驱动器类型

定义	驱动器类型
0	未定义
1	模拟伺服驱动器：M2400 DAC 1 (模拟)
2	模拟伺服驱动器：M2400 DAC 2 (模拟)
3	模拟伺服驱动器：M2400 DAC 3 (模拟)
4	模拟伺服驱动器：M2400 DAC 4 (模拟)
5	MDP 252 驱动器：模拟伺服驱动器：KL4xxx、KL2502-30K (模拟)
6	MDP 252 驱动器：模拟伺服驱动器（非线性）：KL4xxx、KL2502-30K (模拟)
7	高速/低速驱动器 (数字)
8	步进电机驱动器 (数字)
9	SERCOS-Drive (数字)
10	MDP 510 驱动器：KL5051（BiSSI 接口） (数字)
11	AX2000 (Lightbus) (数字)
12	Provi-Drive MC (Simodrive 611U) (数字)
13	通用驱动器 (模拟)
14	NC 后面板 (模拟)
15	特殊 CANopen 类型（如 Lenze Drive 9300） (数字)
16	MDP 742 (DS402)：CANopen 和 EtherCAT CoE (AX2xx-B1x0/B510) (数字)
17	AX2xx-B900 驱动器（以太网） (数字)
20	KL2531/KL2541 编码器（步进电机） (数字)
21	KL2532/KL2542 编码器（DC 电机），KL2535/KL2545 编码器 (PWM 电流端子) (数字)
22	TCom 驱动器（软驱） (数字)
23	MDP 733 驱动器：配置文件 MDP 733 (EL7332、EL7342、EP7342) (数字)

定义	驱动器类型
24	MDP 703 驱动器：配置文件 MDP 703 (EL7031、EL7041、EP7041) (数字)

枚举驱动器-输出-启动类型

定义	枚举驱动器-输出-启动类型
0	未定义
1	输出值（以百分比为单位）
2	输出为速度，例如 m/min

枚举驱动器运行模式

定义	驱动器运行模式（独立于驱动器的通用运行模式）
0	默认模式 (如果模式已知，则重新激活 NC 默认运行模式)
1 (标准类型)	扭矩控制
2 (标准类型)	速度控制，带反馈 1
3 (标准类型)	速度控制，带反馈 2
4 (标准类型)	位置控制，带反馈 1 (滞后少)
5 (标准类型)	位置控制，带反馈 2 (滞后少)
6 (CANopen/CoE 特定)	扭矩控制，带换相角
17 (超采样类型)	使用动态容器进行扭矩控制
18 (超采样类型)	速度控制，带1个反馈，使用动态容器
19 (超采样类型)	速度控制，带2个反馈，使用动态容器
20 (超采样类型)	位置控制，带1个反馈 (滞后少)，使用动态容器
21 (超采样类型)	位置控制，带2个反馈 (滞后少)，使用动态容器
38 (CANopen/CoE 特定)	IO 驱动器控制的归位模式 (适用于第三方设备)
100 (Sercos/SoE 特定)	Sercos/SoE 主要运行模式 0 (参见 S-0-0032)
101 (Sercos/SoE 特定)	Sercos/SoE 次要运行模式 1 (参见 S-0-0033)
102 (Sercos/SoE 特定)	Sercos/SoE 次要运行模式 2 (参见 S-0-0034)
103 (Sercos/SoE 特定)	Sercos/SoE 次要运行模式 3 (参见 S-0-0035)
104 (Sercos/SoE 特定)	Sercos/SoE 次要运行模式 4 (参见 S-0-0284)
105 (Sercos/SoE 特定)	Sercos/SoE 次要运行模式 5 (参见 S-0-0285)
106 (Sercos/SoE 特定)	Sercos/SoE 次要运行模式 6 (参见 S-0-0286)
107 (Sercos/SoE 特定)	Sercos/SoE 次要运行模式 7 (参见 S-0-0287)

主轴的枚举运动阶段/运动状态

定义	运动阶段/运动状态（内部和外部设定点生成之间的区别）
内部设定点生成	
0	设定点发生器未激活 (INACTIVE)
1	设定点发生器激活 (RUNNING)
2	速率比为零 (OVERRIDE_ZERO)
3	匀速 (PHASE_VELOCONST)
4	加速度阶段 (PHASE_ACCPOS)
5	减速度阶段 (PHASE_ACCNEG)
外部设定点生成：	
41	外部设定点生成激活 (EXTSETGEN_MODE1)
42	内部和外部设定点生成激活 (EXTSETGEN_MODE2)

从轴的枚举运动阶段/运动状态

定义	运动阶段/运动状态
0	从轴发生器未激活 (INACTIVE)
11	从轴处于运动前阶段 (PRE-PHASE)

定义	运动阶段/运动状态
12	从轴正在同步 (SYNCHRONIZING)
13	从轴已同步并同步运动 (SYNCHRON)



目前只适用于同步发生器类型的从轴

枚举表主要类型

定义	表主要类型
1	(n*m) 平板形凸轮表 (凸轮系统)
10	(n*m) 特征曲线表 (特性) (例如液压阀特征曲线) 仅支持非循环表格子类型 (1、3) !
16	(n*m) "运动功能"表 (MF) 仅支持非等距表格子类型 (3、4) !

枚举表格子类型

定义	表格子类型
1	(n*m) 主轴位置等距且主轴配置文件无循环延拓 (等距线性) 的表格
2	(n*m) 主轴位置等距且主轴配置文件无循环延拓 (等距循环) 的表格
3	(n*m) 主轴位置非等距但严格单调递增且主轴配置文件无循环延拓 (单调线性) 的表格
4	(n*m) 主轴位置非等距但严格单调递增且主轴配置文件循环延拓 (单调循环) 的表格

枚举表插值类型

定义	参考点之间的表插值类型
0	线性插值 (NC_INTERPOLATIONTYPE_LINEAR) (标准)
1	4 点插值 (NC_INTERPOLATIONTYPE_4POINT) (仅适用于等距表类型)
2	所有参考点的三次样条插值 ("全局样条") (NC_INTERPOLATIONTYPE_SPLINE)
3	通过 n 个插值点进行滑动三次样条插值 ("局部样条") (NC_INTERPOLATIONTYPE_SLIDINGSPLINE)

枚举表运行模式

定义	用于添加、交换和删除表的表运行模式
0	(默认)
1	加法 – 添加另一个表格
2	交换 – 用新表格替换现有表格
3	删除 – 删除现有表格

表格 (凸轮) 耦合信息的结构

表		(凸轮) 耦合信息
nTableID;	1.	凸轮表 ID
nTableMainType;	2.	例如：CAMMING, CHARACTERISTIC, MOTIONFUNCTION
nTableSubType;	3.	例如：EQUIDIST_LINEAR, EQUIDIST_CYCLE, NONEQUIDIST_LINEAR, NONEQUIDIST_CYCLE
nInterpolationType;	4.	例如，LINEAR, 4POINT, SPLINE

表		(凸轮) 耦合信息
nNumberOfRows;	5.	行/元素数量
nNumberOfColumns;	6.	列数
fMasterCamStartPos	7.	主轴凸轮启动位置 (表中第一点)
fSlaveCamStartPos	8.	从轴凸轮启动位置 (表中第一点)
fRawMasterPeriod;	9.	主轴周期/循环 (原始值, 未缩放)
fRawSlaveStroke;	10.	每个主轴周期/循环的从轴差 (原始值, 未缩放)
fMasterAxisCouplingPos	11.	当从轴与主轴耦合后, 凸轮原点的总绝对主动偏移量
fSlaveAxisCouplingPos	12.	当从轴与主轴耦合后, 凸轮原点的总绝对从动偏移量
nMasterAbsolute	13.	主轴绝对位置 (0/1)
nSlaveAbsolute	14.	从轴绝对位置 (0/1)
fMasterOffset;	15.	主轴总偏移
fSlaveOffset;	16.	从轴总偏移
fMasterScaling;	17.	主轴总缩放
fSlaveScaling;	18.	从轴总缩放
fSumOfSlaveStrokes	19.	到 "fActualMasterAxisPos" 为止的从轴冲程总和
fSumOfSuperpositionDistance	20.	叠加距离总和 (位置补偿偏移)
fActualMasterAxisPos;	21.	实际主轴设定点 (绝对)
fActualSlaveAxisPos;	22.	实际从轴设定点 (绝对)
fActualMasterCamPos;	23.	主轴凸轮实际设定点
fActualSlaveCamPos;	24.	主轴凸轮实际设定点
nSlaveStateDWord	25.	从轴状态 DWORD (参见 AxisRef)
...

特征值的结构

特征值		
fMasterVeloNom;	1.	主轴额定速度 (标准化: => 1.0)
fMasterPosStart;	2.	主轴启动位置
fSlavePosStart;	3.	从轴启动位置
fSlaveVeloStart;	4.	从轴启动速度
fSlaveAccStart;	5.	从轴启动加速度
fSlaveJerkStart;	6.	从轴启动加加速度
fMasterPosEnd;	7.	主轴结束位置
fSlavePosEnd;	8.	从轴结束位置
fSlaveVeloEnd;	9.	从轴结束速度
fSlaveAccEnd;	10.	从轴结束加速度
fSlaveJerkEnd;	11.	从轴结束加加速度
fMPosAtSPosMin;	12.	从轴最小位置时的主轴位置
fSlavePosMin;	13.	从轴最小位置
fMPosAtSVeloMin;	14.	从轴最小速度时的主轴位置
fSlaveVeloMin;	15.	从轴最小速度
fMPosAtSAccMin;	16.	从轴最小加速度时的主轴位置
fSlaveAccMin;	17.	从轴最小加速度
fSVeloAtSAccMin;	18.	从轴最小加速度时的从轴速度
fSlaveJerkMin;	19.	从轴最小加加速度
fSlaveDynMomMin;	20.	从轴最小动态动量 (尚不支持!)
fMPosAtSPosMax;	21.	从轴最大位置时的主轴位置
fSlavePosMax;	22.	从轴最大位置
fMPosAtSVeloMax;	23.	从轴最大速度时的主轴位置
fSlaveVeloMax;	24.	从轴最大速度
fMPosAtSAccMax;	25.	从轴最大加速度时的主轴位置
fSlaveAccMax;	26.	从轴最大加速度

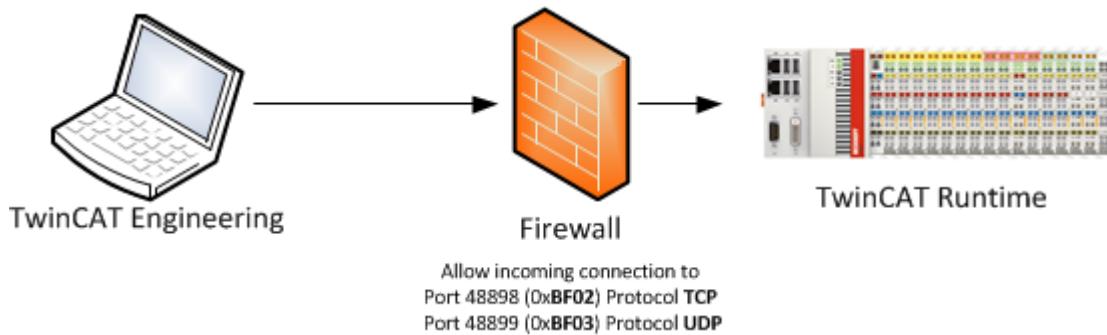
特征值		
fSVeloAtSAccMax;	27.	从轴最大加速度时的从轴速度
fSlaveJerkMax;	28.	从轴最大加加速度
fSlaveDynMomMax;	29.	从轴最小动态动量（尚不支持！）
fSlaveVeloMean;	30.	从轴平均绝对速度
fSlaveAccEff;	31.	从轴有效加速度
nCamTableID;	32.	凸轮表 ID
nNumberOfRows;	33.	行数/条目数，例如点数
nNumberOfColumns;	34.	列数（通常为 1 或 2）
nCamTableType;	35.	凸轮表类型（10=EQUIDIST、 11=NONEQUDIST、22=MOTIONFUNC、 23=CHARACTERISTIC）
nPeriodic;	36.	线性或循环/周期
nReserved	37.	保留

枚举轴控制环路开关类型

定义	轴控制环路开关类型
0	未定义
1	简单切换（类似于轴复位） (标准)
2	通过控制器的 I/D 部分切换/同步至内部初始值（无加加速度/平滑）
3	通过控制器的 I/D 部分切换/同步至参数化初始值

7 通过防火墙的 ADS 连接

本示例用于描述 ADS 设备（例如，TwinCAT 开发环境 PC、第三方 Scada 系统等）如何通过防火墙与 TwinCAT Runtime 通信。



防火墙规则/端口过滤器设置

您必须为与 TwinCAT Runtime 的传入连接配置以下规则，以建立 ADS 通信：

方向	本地端口	远程端口	协议	操作	用途
传入	48898	*	TCP	允许	通信
传入	48899	*	UDP	允许	广播搜索
传入	8016	*	TCP	允许	安全 ADS

8 ADS 返回代码

错误代码分组：

全局错误代码：[ADS 返回代码 \[▶ 115\]...](#) (0x9811_0000 ...)

路由器错误代码：[ADS 返回代码 \[▶ 115\]...](#) (0x9811_0500 ...)

一般 ADS 错误：[ADS 返回代码 \[▶ 116\]...](#) (0x9811_0700 ...)

RTime 错误代码：[ADS 返回代码 \[▶ 117\]...](#) (0x9811_1000 ...)

全局错误代码

Hex	Dec	HRESULT	名称	描述
0x0	0	0x98110000	ERR_NOERROR	无错误。
0x1	1	0x98110001	ERR_INTERNAL	内部错误。
0x2	2	0x98110002	ERR_NORTIME	不具有实时性。
0x3	3	0x98110003	ERR_ALLOCLOCKEDMEM	分配已锁定 – 内存错误。
0x4	4	0x98110004	ERR_INSERTMAILBOX	邮箱已满 – 无法发送 ADS 消息。减少每个周期的 ADS 消息数量将有所帮助。
0x5	5	0x98110005	ERR_WRONGRECEIVEHMSG	HMSG 错误。
0x6	6	0x98110006	ERR_TARGETPORTNOTFOUND	未找到目标端口 – ADS 服务器未启动或无法访问。
0x7	7	0x98110007	ERR_TARGETMACHINENOTFOUND	未找到目标计算机 – 未找到 AMS 路由。
0x8	8	0x98110008	ERR_UNKNOWNCMDID	未知命令 ID。
0x9	9	0x98110009	ERR_BADTASKID	任务 ID 无效。
0xA	10	0x9811000A	ERR_NOIO	No IO。
0xB	11	0x9811000B	ERR_UNKNOWNAMSCMD	未知 AMS 命令。
0xC	12	0x9811000C	ERR_WIN32ERROR	Win32 错误。
0xD	13	0x9811000D	ERR_PORTNOTCONNECTED	端口未连接。
0xE	14	0x9811000E	ERR_INVALIDAMSLENGTH	AMS 长度无效。
0xF	15	0x9811000F	ERR_INVALIDAMSNETID	AMS Net ID 无效。
0x10	16	0x98110010	ERR_LWINSTLEVEL	安装级别 – TwinCAT 2 授权错误。
0x11	17	0x98110011	ERR_NODEBUGINTAVAILABLE	无调试可用。
0x12	18	0x98110012	ERR_PORTDISABLED	端口已禁用 – TwinCAT 系统服务未启动。
0x13	19	0x98110013	ERR_PORTALREADYCONNECTED	端口已连接。
0x14	20	0x98110014	ERR_AMSSYNC_W32ERROR	AMS Sync Win32 错误。
0x15	21	0x98110015	ERR_AMSSYNC_TIMEOUT	AMS Sync 超时。
0x16	22	0x98110016	ERR_AMSSYNC_AMSError	AMS Sync 错误。
0x17	23	0x98110017	ERR_AMSSYNC_NOINDEXINMAP	不存在适用于 AMS Sync 的索引映射。
0x18	24	0x98110018	ERR_INVALIDAMSPORT	AMS 端口无效。
0x19	25	0x98110019	ERR_NOMEMORY	无内存。
0x1A	26	0x9811001A	ERR_TCPSEND	TCP 发送错误。
0x1B	27	0x9811001B	ERR_HOSTUNREACHABLE	主机无法访问。
0x1C	28	0x9811001C	ERR_INVALIDAMSFRACTMENT	AMS 片段无效。
0x1D	29	0x9811001D	ERR_TLSSEND	TLS 发送错误 – ADS 安全连接失败。
0x1E	30	0x9811001E	ERR_ACCESSDENIED	拒绝访问 – 拒绝 ADS 安全访问。

路由器错误代码

Hex	Dec	HRESULT	名称	描述
0x500	1280	0x98110500	ROUTERERR_NOLOCKEDMEMORY	无法分配锁定的内存。
0x501	1281	0x98110501	ROUTERERR_RESIZEMEMORY	路由器内存大小无法更改。
0x502	1282	0x98110502	ROUTERERR_MAILBOXFULL	邮箱已达到最大消息数。
0x503	1283	0x98110503	ROUTERERR_DEBUGBOXFULL	调试邮箱已达到最大消息数。
0x504	1284	0x98110504	ROUTERERR_UNKNOWNPORTTYPE	端口类型未知。
0x505	1285	0x98110505	ROUTERERR_NOTINITIALIZED	路由器未初始化。

Hex	Dec	HRESULT	名称	描述
0x506	1286	0x98110506	ROUTERERR_PORTALREADYINUSE	端口号已分配。
0x507	1287	0x98110507	ROUTERERR_NOTREGISTERED	端口未注册。
0x508	1288	0x98110508	ROUTERERR_NOMOREQUEUES	已达到最大端口数。
0x509	1289	0x98110509	ROUTERERR_INVALIDPORT	端口无效。
0x50A	1290	0x9811050A	ROUTERERR_NOTACTIVATED	路由未激活。
0x50B	1291	0x9811050B	ROUTERERR_FRAGMENTBOXFULL	邮箱已达到最大片段消息数。
0x50C	1292	0x9811050C	ROUTERERR_FRAGMENTTIMEOUT	发生片段超时。
0x50D	1293	0x9811050D	ROUTERERR_TOBEREMOVED	端口已移除。

ADS 通用错误代码

十六进制	十进制	结构化错误码	名称	描述
0x700	1792	0x98110700	ADSERR_DEVICE_ERROR	设备通用错误。
0x701	1793	0x98110701	ADSERR_DEVICE_SRVNOTSUPP	服务器不支持该服务。
0x702	1794	0x98110702	ADSERR_DEVICE_INVALIDGRP	索引组无效。
0x703	1795	0x98110703	ADSERR_DEVICE_INVALIDOFFSET	索引偏移量无效。
0x704	1796	0x98110704	ADSERR_DEVICE_INVALIDACCESS	不允许读取或写入。 可能有几种原因。例如，创建路由时输入了错误的密码。
0x705	1797	0x98110705	ADSERR_DEVICE_INVALIDSIZE	参数大小不正确。
0x706	1798	0x98110706	ADSERR_DEVICE_INVALIDDATA	无效数据值。
0x707	1799	0x98110707	ADSERR_DEVICE_NOTREADY	设备尚未准备好运行。
0x708	1800	0x98110708	ADSERR_DEVICE_BUSY	设备正忙。
0x709	1801	0x98110709	ADSERR_DEVICE_INVALIDCONTEXT	无效的操作系统运行环境。这可能是由于在不同的任务中使用 ADS 函数块造成的。可以通过在 PLC 中进行多任务同步解决这个问题。
0x70A	1802	0x9811070A	ADSERR_DEVICE_NOMEMORY	内存不足。
0x70B	1803	0x9811070B	ADSERR_DEVICE_INVALIDPARM	参数值无效。
0x70C	1804	0x9811070C	ADSERR_DEVICE_NOTFOUND	未找到（文件…）。
0x70D	1805	0x9811070D	ADSERR_DEVICE_SYNTAX	文件或命令中存在语法错误。
0x70E	1806	0x9811070E	ADSERR_DEVICE_INCOMPATIBLE	对象不匹配。
0x70F	1807	0x9811070F	ADSERR_DEVICE_EXISTS	对象已存在。
0x710	1808	0x98110710	ADSERR_DEVICE_SYMBOLNOTFOUND	符号未找到。
0x711	1809	0x98110711	ADSERR_DEVICE_SYMBOLVERSIONINVALID	符号版本无效。这可能是由于联机更改造成的。创建新句柄。
0x712	1810	0x98110712	ADSERR_DEVICE_INVALIDSTATE	设备（服务器）处于无效状态。
0x713	1811	0x98110713	ADSERR_DEVICE_TRANSMODENOTSUPP	不支持 AdsTransMode。
0x714	1812	0x98110714	ADSERR_DEVICE_NOTIFYHNDINVALID	通知句柄无效。
0x715	1813	0x98110715	ADSERR_DEVICE_CLIENTUNKNOWN	通知客户端未注册。
0x716	1814	0x98110716	ADSERR_DEVICE_NOMOREHDLS	没有更多的句柄可用。
0x717	1815	0x98110717	ADSERR_DEVICE_INVALIDWATCHSIZE	通知长度过大。
0x718	1816	0x98110718	ADSERR_DEVICE_NOTINIT	设备未初始化。
0x719	1817	0x98110719	ADSERR_DEVICE_TIMEOUT	设备超时。
0x71A	1818	0x9811071A	ADSERR_DEVICE_NOINTERFACE	接口查询失败。
0x71B	1819	0x9811071B	ADSERR_DEVICE_INVALIDINTERFACE	请求的接口错误。
0x71C	1820	0x9811071C	ADSERR_DEVICE_INVALIDCLSID	Class ID 无效。
0x71D	1821	0x9811071D	ADSERR_DEVICE_INVALIDOBJID	Object ID 无效。
0x71E	1822	0x9811071E	ADSERR_DEVICE_PENDING	请求待定。
0x71F	1823	0x9811071F	ADSERR_DEVICE_ABORTED	请求已中止。
0x720	1824	0x98110720	ADSERR_DEVICE_WARNING	警告信号。
0x721	1825	0x98110721	ADSERR_DEVICE_INVALIDARRAYIDX	数组索引无效。
0x722	1826	0x98110722	ADSERR_DEVICE_SYMBOLNOTACTIVE	符号未激活。
0x723	1827	0x98110723	ADSERR_DEVICE_ACCESSDENIED	拒绝访问。 有几种可能的原因。例如，单向 ADS 路由用于相反方向。
0x724	1828	0x98110724	ADSERR_DEVICE_LICENSENOTFOUND	缺少授权。

十六进制	十进制	结构化错误码	名称	描述
0x725	1829	0x98110725	ADSERR_DEVICE_LICENSEEXPIRED	授权已过期。
0x726	1830	0x98110726	ADSERR_DEVICE_LICENSEEXCEEDED	超出授权。
0x727	1831	0x98110727	ADSERR_DEVICE_LICENSEINVALID	授权无效。
0x728	1832	0x98110728	ADSERR_DEVICE_LICENSESYSTEMID	授权问题：系统 ID 无效。
0x729	1833	0x98110729	ADSERR_DEVICE_LICENSENOTIMELIMIT	授权不受时间限制。
0x72A	1834	0x9811072A	ADSERR_DEVICE_LICENSEFUTUREISSUE	授权问题：许可证设置在未来的时间。
0x72B	1835	0x9811072B	ADSERR_DEVICE_LICENSESETIMETOOLONG	授权期限太长。
0x72C	1836	0x9811072C	ADSERR_DEVICE_EXCEPTION	系统启动异常。
0x72D	1837	0x9811072D	ADSERR_DEVICE_LICENSEDUPLICATED	授权文件读取了两次。
0x72E	1838	0x9811072E	ADSERR_DEVICE_SIGNATUREINVALID	签名无效。
0x72F	1839	0x9811072F	ADSERR_DEVICE_CERTIFICATEINVALID	证书无效。
0x730	1840	0x98110730	ADSERR_DEVICE_LICENSEOEMNOTFOUND	系统无法识别来自OEM的公钥。
0x731	1841	0x98110731	ADSERR_DEVICE_LICENSERESTRICTED	此系统 ID 授权无效。
0x732	1842	0x98110732	ADSERR_DEVICE_LICENSEDEMODENIED	禁止使用演示授权。
0x733	1843	0x98110733	ADSERR_DEVICE_INVALIDFNID	无效 的函数 ID。
0x734	1844	0x98110734	ADSERR_DEVICE_OUTOFRANGE	超出有效范围。
0x735	1845	0x98110735	ADSERR_DEVICE_INVALIDALIGNMENT	无效对齐。
0x736	1846	0x98110736	ADSERR_DEVICE_LICENSEPLATFORM	无效平台级别。
0x737	1847	0x98110737	ADSERR_DEVICE_FORWARD_PL	上下文 - 转到被动级别。
0x738	1848	0x98110738	ADSERR_DEVICE_FORWARD_DL	上下文 - 转到调度级别。
0x739	1849	0x98110739	ADSERR_DEVICE_FORWARD_RT	上下文 - 转到实时。
0x740	1856	0x98110740	ADSERR_CLIENT_ERROR	客户端错误。
0x741	1857	0x98110741	ADSERR_CLIENT_INVALIDPARM	服务包含无效参数。
0x742	1858	0x98110742	ADSERR_CLIENT_LISTEMPTY	轮询列表为空。
0x743	1859	0x98110743	ADSERR_CLIENT_VARUSED	Var 连接已在使用中。
0x744	1860	0x98110744	ADSERR_CLIENT_DUPLINVOKEID	调用的 ID 已在使用中。
0x745	1861	0x98110745	ADSERR_CLIENT_SYNCTIMEOUT	已发生超时 - 远程终端在指定的 ADS 超时 时间内没有响应。远程终端的路由设置可能配置不正确。
0x746	1862	0x98110746	ADSERR_CLIENT_W32ERROR	Win32 子系统中发生错误。
0x747	1863	0x98110747	ADSERR_CLIENT_TIMEOUTINVALID	客户端超时值无效。
0x748	1864	0x98110748	ADSERR_CLIENT_PORTNOTOPEN	端口未打开。
0x749	1865	0x98110749	ADSERR_CLIENT_NOAMSADDR	无 AMS 地址。
0x750	1872	0x98110750	ADSERR_CLIENT_SYNCINTERNAL	Ads 同步过程中发生内部错误。
0x751	1873	0x98110751	ADSERR_CLIENT_ADDHASH	哈希表溢出。
0x752	1874	0x98110752	ADSERR_CLIENT_REMOVEHASH	未在表格中找到密钥。
0x753	1875	0x98110753	ADSERR_CLIENT_NOMORESYM	缓存中没有符号。
0x754	1876	0x98110754	ADSERR_CLIENT_SYNCRESINVALID	收到无效响应。
0x755	1877	0x98110755	ADSERR_CLIENT_SYNCPORTLOCKED	同步端口已锁定。
0x756	1878	0x98110756	ADSERR_CLIENT_REQUESTCANCELLED	请求被取消。

RTime 错误代码

Hex	Dec	HRESULT	名称	描述
0x1000	4096	0x98111000	RTERR_INTERNAL	实时系统中发生内部错误。
0x1001	4097	0x98111001	RTERR_BADTIMERPERIODS	计时器值无效。
0x1002	4098	0x98111002	RTERR_INVALIDTASKPTR	任务指针提供了无效值 0 (零)。
0x1003	4099	0x98111003	RTERR_INVALIDSTACKPTR	堆栈指针提供了无效值 0 (零)。
0x1004	4100	0x98111004	RTERR_PRIOEXISTS	请求任务优先级已分配。
0x1005	4101	0x98111005	RTERR_NOMORETCB	没有可用的空闲 TCB (任务控制块)。TCB 的最大数量为 64。
0x1006	4102	0x98111006	RTERR_NMORESEMAS	没有可用的空闲信号。信号的最大数量为 64。
0x1007	4103	0x98111007	RTERR_NMOREQUEUES	队列中没有可用的空闲空间。队列中的最大位置数为 64。
0x100D	4109	0x9811100D	RTERR_EXTIRQALREADYDEF	已应用外部同步中断。
0x100E	4110	0x9811100E	RTERR_EXTIRQNOTDEF	未应用外部同步中断。

Hex	Dec	HRESULT	名称	描述
0x100F	4111	0x9811100F	RTERR_EXTIRQINSTALLFAILED	外部同步中断应用失败。
0x1010	4112	0x98111010	RTERR_IRQLNOTLESSOREQUAL	在错误的上下文中调用服务函数
0x1017	4119	0x98111017	RTERR_VMXNOTSUPPORTED	不支持 Intel VT-x 扩展。
0x1018	4120	0x98111018	RTERR_VMXDISABLED	BIOS 中未启用 Intel VT-x 扩展。
0x1019	4121	0x98111019	RTERR_VMXCONTROLSMISSING	Intel VT-x 扩展中缺少函数。
0x101A	4122	0x9811101A	RTERR_VMXENABLEFAILS	Intel VT-x 激活失败。

具体的正向 HRESULT 返回代码：

HRESULT	名称	描述
0x0000_0000	S_OK	无错误。
0x0000_0001	S_FALSE	无错误。 示例：处理成功，但有一个负面或不完整的结果。
0x0000_0203	S_PENDING	无错误。 示例：处理成功，但还没有结果。
0x0000_0256	S_WATCHDOG_TIMEOUT	无错误。 示例：处理成功，但发生了超时。

TCP Winsock 错误代码

十六进制	十进制	名称	描述
0x274C	10060	WSAETIMEDOUT	发生连接超时 - 在创建连接时发生错误，因为远程终端在特定时间后未正确响应，或者所建立连接因所连接主机未响应而无法维持。
0x274D	10061	WSAECONNREFUSED	连接遭到拒绝 - 无法建立连接，因为目标计算机明确拒绝了该连接。此错误通常由尝试连接到外部主机上处于非活动状态的服务（即没有运行服务器应用程序的服务）导致。
0x2751	10065	WSAEHOSTUNREACH	不存在至主机的路由 - 套接字操作引用了不可用的主机。
更多 Winsock 错误代码：Win32 错误代码			

Trademark statements

Beckhoff[®], TwinCAT[®], TwinCAT/BSD[®], TC/BSD[®], EtherCAT[®], EtherCAT G[®], EtherCAT G10[®], EtherCAT P[®], Safety over EtherCAT[®], TwinSAFE[®], XFC[®], XTS[®] and XPlanar[®] are registered trademarks of and licensed by Beckhoff Automation GmbH.

Third-party trademark statements

BiSS is a trademark of IC Haus GmbH.

Intel, the Intel logo, Intel Core, Xeon, Intel Atom, Celeron and Pentium are trademarks of Intel Corporation or its subsidiaries.

Microsoft, Microsoft Azure, Microsoft Edge, PowerShell, Visual Studio, Windows and Xbox are trademarks of the Microsoft group of companies.

更多信息：
www.beckhoff.com/te1000

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
电话号码: +49 5246 9630
info@beckhoff.com
www.beckhoff.com

