

GEN 系列运动控制器编程手册

补充说明及补充指令

文档版本

版本号	修订日期
3.00	2021-01-21

指令返回值

返回值	说明	可能的原因	解决措施
0	指令执行正常	/	/
-2	读取数据长度错误	PCI 通信异常，通信区被破坏，数据丢失或包含了杂数据	PCI 通信异常的原因很多，需要根据具体问题具体分析。 可以先尝试更换 PCI 插槽，清除控制卡 PCI 上金手指的灰尘，实在不行只能寄回工厂对硬件进行检测。
-3	读取数据校验和错误	数据在 PCI 传输过程中受到干扰，电平发生改变，导致数据被更改，因此是无效数据	
-4	写入数据块错误	Windows API 函数向 PCI 写或读数据出错，例如 PCI 通信异常，或者没有创建 PCI 通信就向通信区读写数据	
-5	读取数据块错误		
-6	打开/关闭设备错误	打开卡时，没有控制卡、或者控制卡数量超过最大设定值（4 张卡）、创建 PCI 通信区失败 关闭卡时，已经没有卡、或者关闭 PCI 通信区失败	/
-7	DSP 忙	调用 GT 指令后，DSP 仍然在处理，不再接收新指令	/
-8	多线程资源忙	GT 指令在线程里执行超时才返回，有可能是 PCI 通信异常，导致 GT 指令无法及时返回	/
1	错误调用指令	相关的指令没有调用，该指令的执行条件不满足	/
7	参数错误	输入指令的参数出错	/
8	DSP 固件不支持该指令	DSP 固件不支持该指令对应的功能	/

补充指令

该部分功能多为客户定制指令，应用于特定场合。

指令列表

指令 1GTN_SetAxisPressCompensate	6
指令 2GTN_GetAxisPressCompensate.....	7
指令 3GTN_SetAxisPressCompensateTable.....	8
指令 4GTN_SelectAxisPressCompensateTable.....	8
指令 5GTN_RelateEcatIOSlaveToMCIOModule.....	8
指令 6GTN_SetMCIOModuleValue	9
指令 7GTN_GetMCIOModuleValue	9
指令 8GTN_RelateEcatSlaveToMcAuEncoder	10
指令 9GTN_RelateEcatSlaveToMcMpgEncoder.....	10
指令 10GTN_RelateEcatSlaveToMcMpgDi.....	10
指令 11GTN_RelateEcatSlaveToMcGpoBit.....	11
指令 12GTN_RelateEcatSlaveToMcGpiBit.....	11
指令 13GTN_RelateEcatSlaveToMcAuEncoderEx	12
指令 14GTN_RelateEcatSlaveToMcMpgEncoderEx	12
指令 15GTN_RelateEcatSlaveToMcMpgDiEx.....	13
指令 16GTN_RelateEcatSlaveToMcAdc.....	13
指令 17GTN_RelateEcatSlaveToMcAuAdc	14
指令 18 GTN_RelateEcatSlaveToMcDac	15
指令 19 GTN_RelateEcatSlaveToMcAuDac	15
指令 20GTN_SetCrdMPGMode.....	16
指令 21GTN_GetCrdMPGMode	16
指令 22 GTN_PrintLogInfo	17
指令 23GTN_GetAdc.....	18
指令 24GTN_GetAdcValue.....	18
指令 25GTN_GetAuAdc.....	18
指令 26GTN_GetAuAdcValue.....	19
指令 27GTN_SetAdcBias	20
指令 28GTN_GetAdcBias.....	20
指令 29GTN_SetAuAdcBias	20
指令 30GTN_GetAuAdcBias.....	20
指令 31GTN_SetLeadScrewCrossComp	21
指令 32GTN_EnableLeadScrewCrossComp	21
指令 33GTN_LmtsOnEx.....	22
指令 34GTN_LmtsOffEx	22
指令 35GTN_InitMcAxisGap	23
指令 36GTN_SetMcAxisGap	23

新增指令

指令 37	GTN_GetMcAxisGap.....	24
指令 38	GTN_PvtTableUserData.....	24
指令 39	GTN_SetPosCompareThresholdValue	25
指令 40	GTN_GetPosCompareThresholdValue.....	25
指令 41	GTN_GetMcVar	26
指令 42	GTN_GetEcatRawDataPro	26
指令 43	GTN_SetEcatRawDataPro.....	27
指令 44	GTN_InitEcatComm_MultiTask	27

表格索引

表 1 压力补偿指令列表	6
表 2 IO 模块映射指令列表	8
表 3 辅编和手轮映射指令列表	9
表 4 通用输入输出映射指令列表	11
表 5 辅编和手轮映射指令列表	12
表 6 模拟量输入映射指令列表	13
表 7 模拟量输出映射指令列表	14
表 8 手轮引导指令列表	15
表 9 访问模拟量指令列表	18
表 10 设置零漂指令列表	19
表 11 交叉补偿指令列表	21
表 12 限位开关指令列表	21
表 13 占位轴指令列表	23
表 14 占位轴指令列表	24

1、压力补偿

压力补偿功能实现的是压力位置补偿功能，传感器输入力的反馈，计算后补偿到位置上，实现压力的控制。

表 1 压力补偿指令列表

GTN_SetAxisPressCompensate	设置压力补偿参数
GTN_GetAxisPressCompensate	读取压力补偿参数
GTN_SetAxisPressCompensateTable	设置压力补偿表
GTN_SelectAxisPressCompensateTable	选择压力补偿表

指令 1GTN_SetAxisPressCompensate

指令原型	short GTN_SetAxisPressCompensate(short core,shortaxis,TAxisPressCompensate *pPressComp)															
指令说明	设置轴压力补偿（位置环，压力补偿到轴规划位置）。															
指令类型	立即指令，调用后立即生效。	章节页码														
指令参数	该指令共有 3 个参数，参数的详细信息如下。															
core	控制器核号															
axis	要叠加到的规划轴。															
pPressComp	<pre> Typedefstruct { double kp; double ki; double kd; double integralLimit; //积分极限 double derivativeLimit; //微分极限 double limit; //总极限 }TAxisPressPid; Typedefstruct { shortenable; //是否使能，0-关闭，1-使能 shorttype; //输入类型，电压或网络模块数据 </pre> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>type类型宏定义</th> <th>数值</th> </tr> </thead> <tbody> <tr> <td>MC_PRESS_SENSOR</td> <td>500</td> </tr> <tr> <td>MC_ADC</td> <td>24</td> </tr> <tr> <td>MC_AU_ADC</td> <td>19</td> </tr> </tbody> </table> <pre> short dimension; //补偿输入的维度，最大3维。二维补偿输出方向默认为index[0]方向。 short index[3]; //输入的索引,DAC从1开始，ECAT IO模块站号从0开始 shortsubIndex[3]; //输入的子索引，,DAC该参数无效，ECAT IO模块的IOMAP从0开始 short mode; //模式，0-线性还是1-查表 </pre> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>mode类型宏定义</th> <th>数值</th> <th>含义</th> </tr> </thead> <tbody> <tr> <td>PRESS_COMPENSATE_MODE_LINEAR</td> <td>0</td> <td>线性补偿</td> </tr> </tbody> </table>		type类型宏定义	数值	MC_PRESS_SENSOR	500	MC_ADC	24	MC_AU_ADC	19	mode类型宏定义	数值	含义	PRESS_COMPENSATE_MODE_LINEAR	0	线性补偿
type类型宏定义	数值															
MC_PRESS_SENSOR	500															
MC_ADC	24															
MC_AU_ADC	19															
mode类型宏定义	数值	含义														
PRESS_COMPENSATE_MODE_LINEAR	0	线性补偿														

PRESS_COMPENSATE_MODE_TABLE	1	查表补偿
PRESS_COMPENSATE_MODE_REGION_LEARN	2	学习补偿

```

shortrevolveAxis; //用来计算旋转角度的轴号
short regionAxisIndex; //补偿功能区间有效的参考轴
short relatedMasterIndex; //随动主轴的索引
double target; //目标力或电压
doublethreshold; //threshold, 什么时候开始补偿
doubledeadZone; //死区力或电压, 死区内不补偿
double factor; //力和位移的转化系数
doublerevolveOffset; //初始旋转的脉冲数, 默认合成方向和index[0]的方向重合
doublerevolveScale; //旋转轴的一圈脉冲数
TAxisPressPidpid;
long compPosMaxP; //输出补偿位置区间[N,P]的端点P
long compPosMaxN; //输出补偿位置区间[N,P]的端点N
double k; //补偿量滤波系数 0-1 数值越大滤波越强
long activeRegionP; //补偿功能有效的规划位置区间[N,P]的端点P
long activeRegionN; //补偿功能有效的规划位置区间[N,P]的端点N
longactiveRegionInterval; //补偿功能有效的规划位置区间内的自学习间隔, 当mode为自学习PRESS_COMPENSATE_MODE_REGION_LEARN时有效
long relatedMasterEven; //随动主轴的比例
long relatedSlaveEven; //随动从轴的比例
} TAxisPressCompensate;
    
```

指令返回值
相关指令
指令示例

请参照指令返回值列表。

指令 2GTN_GetAxisPressCompensate

指令原型	short GTN_GetAxisPressCompensate(short core,shortaxis,TAxisPressCompensate *pPressComp)
指令说明	获取压力补偿参数。
指令类型	立即指令, 调用后立即生效。
指令参数	该指令共有 3 个参数, 参数的详细信息如下。
core	控制器核号
axis	规划轴。
pPressComp	获取的参数值。
指令返回值	请参照指令返回值列表。
相关指令	
指令示例	

指令 3GTN_SetAxisPressCompensateTable

指令原型	short GTN_SetAxisPressCompensateTable(short core,shortaxis, short index,longcount,double *pPressData,double *pPosData)	
指令说明	设置压力补偿。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 5 个参数，参数的详细信息如下。	
core	控制器核号	
axis	规划轴。	
index	表格索引，从 1 开始，目前支持 2 个。	
count	传入的数据量，目前最大 2048。	
pPressData	压力数据，需要单调递增。	
pPosData	位置数据，单位：脉冲。	
指令返回值	请参照指令返回值列表。	
相关指令		
指令示例		

指令 4GTN_SelectAxisPressCompensateTable

指令原型	short GTN_SelectAxisPressCompensateTable(short core,shortaxis,short index)	
指令说明	选择补偿表。GTN_SetAxisPressCompensateTable 可以设置多个表格的数据，通过该指令选择哪张表格生效。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号	
axis	规划轴。	
index	表格索引，从 1 开始，目前支持 2 个。	
指令返回值	请参照指令返回值列表。	
相关指令		
指令示例		

2、EtherCAT IO Slave 到控制器 IO 模块映射

实现 EtherCAT 网络上的资源到运动控制器资源的映射。

表 2 IO 模块映射指令列表

GTN_RelateEcatIOSlaveToMCIOModule	映射 IO 模块到控制器 IO 模块
GTN_SetMCIOModuleValue	设置 IO 模块的值
GTN_GetMCIOModuleValue	读取 IO 模块的值

指令 5GTN_RelateEcatIOSlaveToMCIOModule

指令原型	GT_API GTN_RelateEcatIOSlaveToMCIOModule(shortcore,shortslaveindex,shortIOModuleindex);
指令说明	设置 EtherCAT 网络下连接的 IO Slave 从站到控制器 IO 模块的映射，可以选定总线上的其中一个 IO Slave 作为控制器的 IO 模块。

新增指令

指令类型	立即指令，调用后立即生效。	章节页码	
指令参数	该指令共有 3 个参数，参数的详细信息如下。		
core	控制器核号		
slaveindex	从站的序号，可以根据实际连接情况，从总线连接上任意选取一个 IO 从站。		
IOModuleindex	控制器映射的 IO 模块需要，取值范围 0-1		
指令返回值	请参照指令返回值列表。		
相关指令			
指令示例			

指令 6GTN_SetMCIOModuleValue

指令原型	GTN_SetMCIOModuleValue (shortcore,shortIOModuleindex,shortiomapindex,long *val,shortcnt);		
指令说明	设置控制器 IO 模块的输出值，对应所映射的 IO SLAVE RxPDO 值。		
指令类型	立即指令，调用后立即生效。	章节页码	
指令参数	该指令共有 4 个参数，参数的详细信息如下。		
core	控制器核号		
IOModuleindex	IO 模块索引。		
iomapindex	IO 模块的 MAP 索引。		
Val	设置数值的变量地址。		
cnt	设置数值的数量，取值范围 1-2		
指令示例			

指令 7GTN_GetMCIOModuleValue

指令原型	GT_API GTN_GetMCIOModuleValue (shortcore,shortIOModuleindex,shortiomapindex,long *val,shortcnt);		
指令说明	读取控制器 IO 模块的输出值，对应所映射的 IO SLAVE TxPDO 值。		
指令类型	立即指令，调用后立即生效。	章节页码	
指令参数	该指令共有 3 个参数，参数的详细信息如下。		
core	控制器核号		
IOModuleindex	IO 模块索引。		
iomapindex	IO 模块的 MAP 索引。		
Val	读取数值的变量地址。		
cnt	读取数值的数量，取值范围 1-2		
指令示例			

3、EtherCAT Motion Slave 到控制器辅助编码器/手轮的映射

实现 EtherCAT 网络上的编码器和 IO 资源到控制器的辅助编码器和手轮的映射。

表 3 辅编和手轮映射指令列表

GTN_RelateEcatSlaveToMcAuEncoder	映射辅助编码器
----------------------------------	---------

新增指令

GTN_RelateEcatSlaveToMcMpgEncoder	映射手轮编码器
GTN_RelateEcatSlaveToMcMpgDi	映射手轮 DI

指令 8GTN_RelateEcatSlaveToMcAuEncoder

指令原型	GT_API GTN_RelateEcatSlaveToMcAuEncoder(shortcore,shortauenc,shortecatAxisIndex);	
指令说明	设置 EtherCAT 网络下连接的 Motion Slave 轴的辅助编码器资源（GNM-401-EM 模块为例，对象字 0x61F2）到控制器辅助编码器的映射。 可以选定总线上的其中一个拥有辅助编码器资源的 Motion Slave 轴作为控制器的辅助编码器。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
auenc	辅编的序号，取值范围 1-8。	
ecatAxisindex	可以根据实际连接情况，从总线连接上选取一个 Motion Slave 轴号	
指令返回值	请参照指令返回值列表。	
备注	上电及 GT_RESET 后控制器 AuEncoder 没有映射 Motion Slave。	

指令 9GTN_RelateEcatSlaveToMcMpgEncoder

指令原型	GT_API GTN_RelateEcatSlaveToMcMpgEncoder(short core,shortmpg,shortecatAxisIndex);	
指令说明	设置 EtherCAT 网络下连接的 Motion Slave 轴的辅助编码器资源（GNM-401-EM 模块为例，对象字 0x61F2）到控制器手轮编码器的映射。 可以选定总线上的其中一个拥有辅助编码器资源的 Motion Slave 轴作为控制器的手轮编码器。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 4 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
mpg	控制器的手轮编号。	
ecatAxisIndex	可以根据实际连接情况，从总线连接上选取一个 Motion slave 轴号。	
备注	上电及 GT_RESET 后控制器 AuEncoder 没有映射 Motion Slave。	

指令 10GTN_RelateEcatSlaveToMcMpgDi

指令原型	GTN_RelateEcatSlaveToMcMpgDi(short core,shortmpg,shortecatAxisIndex,shortbitOffset)	
指令说明	设置 EtherCAT 网络下连接的 Motion Slave 轴的 DI 资源（CIA402 对象字 0x60FD）到控制器手轮 DI 的映射。 可以选定总线上的其中一个拥有 DI 资源的 Motion Slave 轴作为控制器的手轮的 DI。	

指令类型	立即指令，调用后立即生效。	章节页码	
指令参数	该指令共有 3 个参数，参数的详细信息如下。		
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2		
mpg	控制器的手轮编号。		
ecatAxisIndex	可以根据实际连接情况，从总线连接上选取一个 Motion Slave 轴号。		
bitOffset	DI 的 bit 在 Motion Slave 的对象字 0x60FD 上的偏移。		
备注	上电及 GT_RESET 后控制器 MpgDi 没有映射 Motion Slave。		

4、EtherCAT Slave 到控制器通用输入输出的映射

实现 EtherCAT 的输入输出资源到控制器通用输入输出的映射。

表 4 通用输入输出映射指令列表

GTN_RelateEcatSlaveToMcGpoBit	映射通用输出
GTN_RelateEcatSlaveToMcGpiBit	映射通用输入

指令 11 GTN_RelateEcatSlaveToMcGpoBit

指令原型	GT_API GTN_RelateEcatSlaveToMcGpoBit(short core,short gpo,short ecatIndex,short ecatType,short bitoffset,short pdoOffset);	
指令说明	设置 EtherCAT 网络下连接的 Motion Slave 从站到控制器辅助编码器的映射，可以选定总线上的其中一个 Motion Slave 作为控制器的通用输出。	
指令类型	立即指令，调用后立即生效。	
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号	
gpo	DO 的序号 1-32。	
ecatType	1 Motion 站即控制器轴索引	0 IO 站即 EtherCAT 总线连接顺序的站号
ecatIndex	轴索引，使用 DigitalOUT	站号索引使用 pdoOffset
bitoffset	DO Bit 所在的索引位置	
pdoOffset	根据 Type 无效参数填	IO 点所在的 PDO Bytes 偏移
指令返回值	请参照指令返回值列表。	
相关指令		
指令示例		

指令 12 GTN_RelateEcatSlaveToMcGpiBit

指令原型	GT_API GTN_RelateEcatSlaveToMcGpiBit(short core,short gpi,short ecatIndex,short ecatType,short bitoffset,short pdoOffset);	
指令说明	设置 EtherCAT 网络下连接的 Motion Slave 从站到控制器辅助编码器的映射，可以选定总线上的其中一个 Motion Slave 作为控制器的 GPI 输入。	
指令类型	立即指令，调用后立即生效。	

指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号	
gpi	Di 的序号 1-32。	
ecatType	1 Motion 站即控制器轴索引	0 IO 站即 EtherCAT 总线连接顺序的站号
ecatIndex	轴索引，使用 60FE DigitalIN	站号索引使用 pdoOffset
bitoffset	Di Bit 所在的索引位置	
pdoOffset	根据 Type 无效参数	IO 点所在的 PDO Bytes 偏移
指令返回值	请参照指令返回值列表。	
相关指令		
指令示例		

5、EtherCAT Slave 到控制器辅助编码器/手轮的映射

实现 EtherCAT 网络上的编码器和 IO 资源到控制器的辅助编码器和手轮的映射。

表 5 辅编和手轮映射指令列表

GTN_RelateEcatSlaveToMcAuEncoderEx	映射辅助编码器
GTN_RelateEcatSlaveToMcMpgEncoderEx	映射手轮编码器
GTN_RelateEcatSlaveToMcMpgDiEx	映射手轮 DI

指令 13GTN_RelateEcatSlaveToMcAuEncoderEx

指令原型	GT_API GTN_RelateEcatSlaveToMcAuEncoderEx(short core,short auenc,short ecatIndex,short ecatType,short pdoOffset,short pdoByteLength);	
指令说明	设置 EtherCAT 网络下连接的 Motion Slave 轴的辅助编码器资源(GNM-401-EM 模块为例，对象字 0x61F2) 到控制器辅助编码器的映射。 可以选定总线上的其中一个拥有辅助编码器资源的 Motion Slave 轴作为控制器的辅助编码器。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
auenc	辅编的序号，取值范围 1-8。	
ecatType	1 Motion 站即控制器轴索引	0 IO 站即 EtherCAT 总线连接顺序的站号
ecatIndex	轴索引，使用轴的辅助编码器	站号索引，使用 pdoOffset
pdoOffset	根据 Type 无效参数	编码器所在的 PDO Bytes 偏移
pdoByteLength	根据 Type 无效参数	编码器数据的 PDO Bytes 长度
指令返回值	请参照指令返回值列表。	
备注	上电及 GT_RESET 后控制器 AuEncoder 没有映。	

指令 14GTN_RelateEcatSlaveToMcMpgEncoderEx

指令原型	GT_API GTN_RelateEcatSlaveToMcMpgEncoderEx(short core,shortmpg,short ecatIndex,short ecatType,short pdoOffset,short pdoByteLength);
------	---

指令说明	设置 EtherCAT 网络下连接的 Motion Slave 轴的辅助编码器资源（GNM-401-EM 模块为例，对象字 0x61F2）到控制器手轮编码器的映射。 可以选定总线上的其中一个拥有辅助编码器资源的 Motion Slave 轴作为控制器的手轮编码器。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 4 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
mpg	控制器的手轮编号。	
ecatType	1 Motion 站即控制器轴索引	0 IO 站即 EtherCAT 总线连接顺序的站号
ecatIndex	轴索引，使用轴的辅助编码器	站号索引，使用 pdoOffset
pdoOffset	根据 Type 无效参数	编码器所在的 PDO Bytes 偏移
pdoByteLength	根据 Type 无效参数	编码器数据的 PDO Bytes 长度
备注	上电及 GT_RESET 后控制器 AuEncoder 没有映射 Motion Slave。	

指令 15GTN_RelateEcatSlaveToMcMpgDiEx

指令原型	GTN_RelateEcatSlaveToMcMpgDiEx(short core,shortmpg, short ecatIndex,short ecatType,short bitoffset,short pdoOffset);	
指令说明	设置 EtherCAT 网络下连接的 Motion Slave 轴的 DI 资源（CIA402 对象字 0x60FD）到控制器手轮 DI 的映射。 可以选定总线上的其中一个拥有 DI 资源的 Motion Slave 轴作为控制器的手轮的 DI。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
mpg	控制器的手轮编号。	
ecatType	1 Motion 站即控制器轴索引	0 IO 站即 EtherCAT 总线连接顺序的站号
ecatIndex	轴索引，使用 60FE DigitalIN	站号索引使用 pdoOffset
bitoffset	Di Bit 所在的索引位置	
pdoOffset	根据 Type 无效参数	IO 点所在的 PDO Bytes 偏移
备注	上电及 GT_RESET 后控制器 MpgDi 没有映射 Motion Slave。	

6、EtherCAT Slave 到控制器模拟量和辅助模拟量输入的映射

实现 EtherCAT 的输入资源到控制器模拟量和辅助模拟量输入的映射。

表 6 模拟量输入映射指令列表

GTN_RelateEcatSlaveToMcAdc	映射模拟量输入
GTN_RelateEcatSlaveToMcAuAdc	映射辅助模拟量输入

指令 16GTN_RelateEcatSlaveToMcAdc

指令原型	GT_API GTN_RelateEcatSlaveToMcAdc (short core,short adc,short ecatIndex,short ecatType,short pdoOffset,short pdoByteLength);	
指令说明	设置 EtherCAT 网络下连接的 Slave 从站到控制器模拟量输入的映射，可以选定总线上的其中一个 Slave 作为控制器的模拟量输入。	
指令类型	立即指令，调用后立即生效。	
指令参数	该指令共有 6 个参数，参数的详细信息如下。	
core	控制器核号	
adc	adc 的序号 1-32。	
ecatType	1 Motion 站即控制器轴索引	0 IO 站即 EtherCAT 总线连接顺序的站号
ecatIndex	轴索引，使用 analogIn1	站号索引,使用 pdoOffset
pdoOffset	根据 Type 无效参数	IO 点所在的 PDO Bytes 偏移
pdoLength	根据 Type 无效参数	IO 点所在的 PDO Bytes 长度
指令返回值	请参照指令返回值列表。	
相关指令		
指令示例		

指令 17GTN_RelateEcatSlaveToMcAuAdc

指令原型	GT_API GTN_RelateEcatSlaveToMcAuAdc(short core,short auadc,short ecatIndex,short ecatType,short pdoOffset,short pdoByteLength);	
指令说明	设置 EtherCAT 网络下连接的 Slave 从站到控制器辅助模拟量输入的映射，可以选定总线上的其中一个 Slave 作为控制器的辅助模拟量输入。	
指令类型	立即指令，调用后立即生效。	
指令参数	该指令共有 6 个参数，参数的详细信息如下。	
core	控制器核号	
audac	辅助模拟量输入的序号 1-32。	
ecatType	1 Motion 站即控制器轴索引	0 IO 站即 EtherCAT 总线连接顺序的站号
ecatIndex	轴索引，使用 analogIn2	站号索引，使用 pdoOffset
pdoOffset	根据 Type 无效参数	IO 点所在的 PDO Bytes 偏移
pdoLength	根据 Type 无效参数	IO 点所在的 PDO Bytes 长度
指令返回值	请参照指令返回值列表。	
相关指令		
指令示例		

7、EtherCAT Slave 到控制器模拟量和辅助模拟量输出的映射

实现 EtherCAT 的模拟量输出资源到控制器模拟量和辅助模拟量输出的映射。

表 7 模拟量输出映射指令列表

GTN_RelateEcatSlaveToMcDac	映射模拟量输出
GTN_RelateEcatSlaveToMcAuDac	映射辅助模拟量输出

指令18 GTN_RelateEcatSlaveToMcDac

指令原型	GT_API GTN_RelateEcatSlaveToMcDac(short core,short dac,short ecatIndex,short ecatType,short pdoOffset,short pdoByteLength);	
指令说明	设置 EtherCAT 网络下连接的 Slave 从站到控制器模拟量输出的映射，可以选定总线上的其中一个 Slave 作为控制器的模拟量输出。	
指令类型	立即指令，调用后立即生效。	
指令参数	该指令共有 6 个参数，参数的详细信息如下。	
core	控制器核号	
dac	dac 的序号 1-32。	
ecatType	1 Motion 站即控制器轴索引	0 IO 站即 EtherCAT 总线连接顺序的站号
ecatIndex	轴索引，使用 analogIout	站号索引,使用 pdoOffset
pdoOffset	根据 Type 无效参数	IO 点所在的 PDO Bytes 偏移
pdoLength	根据 Type 无效参数	IO 点所在的 PDO Bytes 长度
指令返回值	请参照指令返回值列表。	
相关指令		
指令示例		

指令 19 GTN_RelateEcatSlaveToMcAuDac

指令原型	GT_API GTN_RelateEcatSlaveToMcAuDac(short core,short auDac,short ecatIndex,short ecatType,short pdoOffset,short pdoByteLength);	
指令说明	设置 EtherCAT 网络下连接的 Slave 从站到控制器辅助模拟量输出的映射，可以选定总线上的其中一个 Slave 作为控制器的辅助模拟量输出。	
指令类型	立即指令，调用后立即生效。	
指令参数	该指令共有 6 个参数，参数的详细信息如下。	
core	控制器核号	
audac	辅助模拟量输出的序号 1-32。	
ecatType	1 Motion 站即控制器轴索引	0 IO 站即 EtherCAT 总线连接顺序的站号
ecatIndex	轴索引，使用 analogout	站号索引，使用 pdoOffset
pdoOffset	根据 Type 无效参数	IO 点所在的 PDO Bytes 偏移
pdoLength	根据 Type 无效参数	IO 点所在的 PDO Bytes 长度
指令返回值	请参照指令返回值列表。	
相关指令		
指令示例		

8、手轮引导指令

手轮引导功能的相关指令。

表 8 手轮引导指令列表

GTN_SetCrdMPGMode	设置手轮引导参数
-------------------	----------

GTN_GetCrdMPGMode	读取手轮引导参数
-------------------	----------

指令 20GTN_SetCrdMPGMode

指令原型	GT_API GTN_SetCrdMPGMode(short core,shortcrd,shortenable,shortmaster,longmasterEven,longslaveEven,shortfilterTime,short mode)	
指令说明	设置 GEN 坐标系运动为手轮模式。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 8 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
crd	坐标系号。	
enable	可以根据实际连接情况，从总线连接上选取一个 Motion 轴号。	
master	使用的主轴号可以是控制器 Axis 轴号，也可以是 AuEncoder 号	
masterEven	主轴的运动量单位 Pulse	
slaveEven	坐标系的运动量，单位 Pulse	
filterTime	速度滤波时间，单位 ms	
mode	100 对应 master 为规划器号 0 对应 master 为辅助编码器号	

指令 21GTN_GetCrdMPGMode

指令原型	GT_API GTN_GetCrdMPGMode(short core,shortcrd,short *pEnable,short *pMaster,long *pMasterEven,long *pSlaveEven,short *pFilterTime,short *pMode,short *pFifoEnd)	
指令说明	映射 ECAT Axis 上的辅助编码器到手轮的编码器。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 8 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
crd	坐标系号。	
pEnable	可以根据实际连接情况，从总线连接上选取一个 Motion 轴号。	
pMaster	使用的主轴号可以是控制器 Axis 轴号，也可以是 AuEncoder 号	
pMasterEven	主轴的运动量单位 Pulse	
pSlaveEven	坐标系的运动量，单位 Pulse	
pFilterTime	速度滤波时间，单位 ms	
pMode	手轮模式	
pFifoEnd	0 未启动手轮引导 1 正向段末-1 反向段末	

9、打印控制器日志

指令 22 GTN_PrintLogInfo

指令原型	GT_API GTN_PrintLogInfo(short core,const char *pFileName,longshart,long count)		
指令说明	打印控制器的日志。		
指令类型	立即指令，调用后立即生效。	章节页码	
指令参数	该指令共有 4 个参数，参数的详细信息如下。		
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2		
pFileName	文件名称(txt 格式).		
shart	打印起始点 0: 从 FIFO 中最旧的数据位置打印 <0 从最新数据倒退 start 索引位置打印 >0 跳过 start 个最旧的数据		
count	0: 打印从 start 位置到最新数据所有的记录 count >0 需要打印的数据个数		
备注	实际打印数据可能受控制器产生的日志信息数量影响，和 count 输入不符。		

10、GEN 访问模拟量输入

GEN 控制卡内部的 MC_ADC 和 MC_AUADC 没有对应的硬件资源。连接网络后，默认一一对应所连接的 EtherCAT Motion Slave 轴拥有的 AnalogIN1（映射到控制器的 ADC 输入）和 AnalogIN2 资源（映射到控制器的 AuADC 输入）。

AnalogIN1 和 AnalogIN2 在 CIA402 协议没有明确的对象字典约定。

例如 A：当 1 轴为 GTHD 驱动器，驱动器提供两路模拟量输入对对象字典的 0x20F2 和 0x20F9。

使用指令 GTN_GetAdcValue (1,1, &adcvalue,1,NULL);将获得 0x20F2 对应的模拟量输入值，具体与电压的对应关系，请参考驱动器说明手册。

使用指令 GTN_GetAuAdcValue (1,1, &auadcvalue,1,NULL);将获得 0x20F9 对应的模拟量输入值，具体与电压的对应关系，请参考驱动器说明手册。

例如 B：当连接的从站为固高系列 GNM-401-EM，每个轴提供两路模拟量输入对对象字典的 0x61F0 和 0x61F3。多个轴的对象字典偏移 0x800

使用指令 GTN_GetAdcValue (1,1, &adcvalue,1,NULL);将获得 0x61F0 对应的模拟量输入值，具体与电压的对应关系，请参考驱动器说明手册。

使用指令 GTN_GetAuAdcValue (1,1, &auadcvalue,1,NULL);将获得 0x61F3 对应的模拟量输入值，具体与电压的对应关系，请参考驱动器说明手册。

模拟量 GTN_GetAdc 可以直接读取转换后的电压值，对应关系-32768 – 32767 对应-10V- 10V。

表 9 访问模拟量指令列表

GTN_GetAdc	读取模拟量电压值
GTN_GetAdcValue	读取模拟量原始值
GTN_GetAuAdc	读取辅助模拟量电压值
GTN_GetAuAdcValue	读取辅助模拟量原始值

指令 23GTN_GetAdc

指令原型	short GTN_GetAdc(short core, short adc, double *pValue, short count=1, unsigned long *pClock=NULL)		
指令说明	读取模拟量输入的电压值。		
指令类型	立即指令，调用后立即生效。	章节页码	18
指令参数	该指令共有 5 个参数，参数的详细信息如下。		
core	控制器核号。GEN 8,16,32版本取值范围1， GEN 64版本取值范围1-2		
adc	adc 起始通道号(对应 GEN 版本的轴号范围)		
pValue	读取的输入电压值。单位：伏特。		
count	读取的通道数，默认为 1。 1 次最多可以读取 8 路 adc 输入电压值。		
pClock	读取控制器时钟，默认值为： NULL，即不用读取控制器时钟。		
指令返回值	请参照指令返回值列表。		
相关指令			
备注	控制器上电及 GT_RESET 后， adc 和 axis 索引一一对应。		

指令 24GTN_GetAdcValue

指令原型	short GTN_GetAdcValue (short core, short adc, short *pValue, short count=1, unsigned long *pClock=NULL)		
指令说明	读取模拟量输入的数字转换值。		
指令类型	立即指令，调用后立即生效。	章节页码	18
指令参数	该指令共有 5 个参数，参数的详细信息如下。		
core	控制器核号。GEN 8,16,32版本取值范围1， GEN 64版本取值范围1-2		
adc	adc 起始通道号(对应 GEN 版本的轴号范围)		
pValue	读取的输入电压数值。单位： bit，取值范围： [-32768, 32767]，对应的电压值为[-10, 10]伏特。		
count	读取的通道数。默认为 1。 1 次最多可以读取 8 路 adc 输入电压值。		
pClock	读取控制器时钟，默认值为： NULL，即不用读取控制器时钟。		
指令返回值	请参照指令返回值列表。		
相关指令	指令 23GTN_GetAdc		
备注	控制器上电及 GT_RESET 后， adc 和 axis 索引一一对应。		

指令 25GTN_GetAuAdc

指令原型	short GTN_GetAuAdc(short core, short adc, double *pValue, short count, unsigned long *pClock)		
指令说明	读取本地模拟量输入的电压值。		
指令类型	立即指令，调用后立即生效。	章节页码	18
指令参数	该指令共有 5 个参数，参数的详细信息如下。		
core	控制器核号。GEN 8,16,32版本取值范围1， GEN 64版本取值范围1-2		
adc	adc 起始通道号取值范围[1,12]		
pValue	读取的输入电压值。单位：伏特。		
count	读取的通道数，取值范围： [1,12]。		
pClock	读取控制器时钟，默认值为： NULL，即不用读取控制器时钟。		
指令返回值	请参照指令返回值列表。		
相关指令	指令 26GTN_GetAuAdcValue		
备注	辅助模拟量输入最大 12 路，控制器上电及 GT_RESET 后， adc 和 axis 索引[1,12]一一对应。		

指令 26GTN_GetAuAdcValue

指令原型	short GTN_GetAuAdcValue(short core, short adc, short *pValue, short count, unsigned long *pClock)		
指令说明	读取本地模拟量输入的数字转换值。		
指令类型	立即指令，调用后立即生效。	章节页码	18
指令参数	该指令共有 5 个参数，参数的详细信息如下。		
core	控制器核号。GEN 8,16,32版本取值范围1， GEN 64版本取值范围1-2		
adc	adc 起始通道号取值范围[1,12]		
pValue	读取的输入电压数值。单位： bit，取值范围： [-32768, 32767]，对应的电压值为[-10, 10]伏特。		
count	读取的通道数，取值范围： [1,12]。		
pClock	读取控制器时钟，默认值为： NULL，即不用读取控制器时钟。		
指令返回值	请参照指令返回值列表。		
相关指令			
备注	辅助模拟量输入最大 12 路，控制器上电及 GT_RESET 后， adc 和 axis 索引[1,12]一一对应。		

11、GEN 设置模拟量输入的零漂

控制器复位后， adc 和 auadc 的 bias 值都是 0。用户将在 bias=0 时指令 GTN_GetAdcValue 和 GTN_GetAuAdcValue 读取到的值对应填入到 GTN_SetAdcBias 和 GTN_SetAuAdcBias 的 bias,这是将读取到值为 0。以此实现模拟量输入的软件归零。

表 10 设置零漂指令列表

GTN_SetAdcBias	设置模拟量输入的零漂
GTN_GetAdcBias	读取模拟量输入的零漂
GTN_SetAuAdcBias	设置辅助模拟量输入的零漂
GTN_GetAuAdcBias	读取辅助模拟量输入的零漂

指令 27GTN_SetAdcBias

指令原型	GT_API GTN_SetAdcBias(short core,shortadc,short bias)	
指令说明	设置 ADC 的零漂。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
adc	ADC 的索引(对应 GEN 版本的轴号范围)。	
bias	零漂值取值范围：[-32768 32767]	
备注	控制器上电及 GT_RESET 后，adc 和 axis 索引一一对应。	

指令 28GTN_GetAdcBias

指令原型	GT_API GTN_GetAdcBias(short core,shortadc,short *pBias)	
指令说明	获取 ADC 的零漂。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
adc	ADC 的索引(对应 GEN 版本的轴号范围)。	
pBias	获取的零漂值。	
备注	控制器上电及 GT_RESET 后，adc 和 axis 索引一一对应。	

指令 29GTN_SetAuAdcBias

指令原型	GT_API GTN_SetAuAdcBias(short core,shortauAdc,short bias)	
指令说明	设置 AUADC 的零漂。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
auAdc	通道号取值范围[1,12]	
bias	零漂值取值范围：-32768 -32767	
备注	辅助模拟量输入最大 12 路，控制器上电及 GT_RESET 后，adc 和 axis 索引[1,12]一一对应。	

指令 30GTN_GetAuAdcBias

指令原型	GT_API GTN_GetAuAdcBias(short core,shortauAdc,short *pBias)	
指令说明	获取 AUADC 的零漂。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	控制器核号。GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
auAdc	通道号取值范围[1,12]	
pBias	获取的零漂值	
备注	辅助模拟量输入最大 12 路，控制器上电及 GT_RESET 后，adc 和 axis 索引[1,12]一一对应。	

12、交叉补偿

表 11 交叉补偿指令列表

GTN_SetLeadScrewCrossComp	设置交叉补偿参数
GTN_EnableLeadScrewCrossComp	读取交叉补偿参数

指令 31 GTN_SetLeadScrewCrossComp

指令原型	GTN_SetLeadScrewCrossComp(shortcore,shortaxis,shortn,longstartPos,longlenPos,long *pPositive,long *pNegative,shortlink);		
指令说明	加载补偿表		
指令类型	立即指令，调用后立即生效。	章节页码	
指令参数	该指令共有 7 个参数，参数的详细信息如下。		
core	内核，正整数 GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2		
axis	补偿轴轴号,正整数		
n	补偿段数		
startPos	补偿开始位置		
lenPos	补偿总距离（每个补偿区间长度为： $lenPos/(n-1)$ ）		
pCompPos	正向补偿表数组地址		
pCompNeg	负向补偿表数组地址		
link	参考的主轴编号。		
指令返回值	请参照指令返回值列表。		
相关指令	无。		
指令示例			

指令 32 GTN_EnableLeadScrewCrossComp

指令原型	short GTN_EnableLeadScrewCrossComp(short core,short axis,shortenable);		
指令说明	是否开启补偿		
指令类型	立即指令，调用后立即生效。	章节页码	
指令参数	该指令共有 3 个参数，参数的详细信息如下。		
core	内核，正整数，GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2		
axis	补偿轴轴号，正整数		
enable	0-关闭；1-开启		
指令返回值			
相关指令	无。		
指令示例			

13、限位开关

设置限位开关的状态，分别设置软限位和硬限位的状态。

表 12 限位开关指令列表

GTN_LmtsOnEx	设置限位开关有效
GTN_LmtsOffEx	设置限位开关无效

指令 33GTN_LmtsOnEx

指令原型	short GTN_LmtsOnEx (short core,short axis,short limitType=-1,short limitMode=-1);	
指令说明	开启控制器限位	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 4 个参数，参数的详细信息如下。	
core	内核，正整数，GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
axis	轴号，正整数	
limitType	MC_LIMIT_POSITIVE 正限位	
	MC_LIMIT_NEGATIVE 负限位	
limitMode	-1 正限位和负限位	
	LIMIT_MODE_EXTERMAL 硬限位	
limitMode	LIMIT_MODE_SOFT 软限位	
	-1 硬限位和软限位	
指令返回值		
相关指令	无。	
指令示例		

指令 34GTN_LmtsOffEx

指令原型	short GTN_LmtsOffEx (short core,short axis,short limitType=-1,short limitMode=-1);	
指令说明	关闭控制器限位	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 4 个参数，参数的详细信息如下。	
core	内核，正整数，GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
axis	轴号，正整数	
limitType	MC_LIMIT_POSITIVE 正限位	
	MC_LIMIT_NEGATIVE 负限位	
limitMode	-1 正限位和负限位	
	LIMIT_MODE_EXTERMAL 硬限位	
limitMode	LIMIT_MODE_SOFT 软限位	
	-1 硬限位和软限位	
指令返回值		
相关指令	无。	
指令示例		

14、占位轴指令

占位轴功能用于控制卡的轴号和实际连接的 EtherCAT 伺服轴顺序不一一对应时，将控制卡的某个或

新增指令

者某些轴设置为占位轴。

设置为占位轴的轴：规划器仍可以正常使用，但不实际操作网络上的伺服轴。

如下图所示，将轴 3 和轴 6 设置为占位轴时，实际操作控制卡 4 轴将操作网络上的 3 号驱动器。操作控制卡 8 轴将操作网络上的 6 号驱动器。

Axis 1	Axis 2	Axis 3 占位轴	Axis 4	Axis 5	Axis 6 占位轴	Axis 7	Axis 8
--------	--------	---------------	--------	--------	---------------	--------	--------

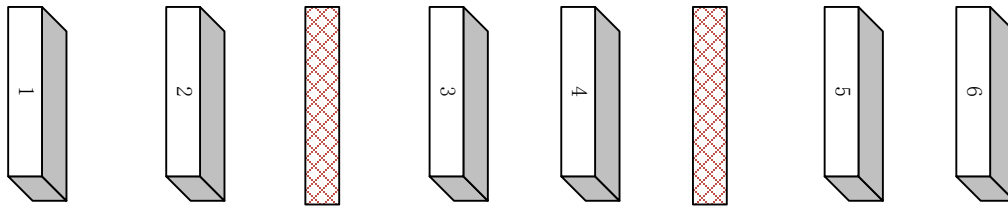


表 13 占位轴指令列表

指令 35GTN_InitMcAxisGap	初始化占位轴
指令 36GTN_SetMcAxisGap	设置占位轴
指令 37GTN_GetMcAxisGap	读取占位轴的信息

指令 35GTN_InitMcAxisGap

指令原型	ShortGTN_InitMcAxisGap(short core);	
指令说明	初始化占位轴信息,指令执行后控制卡的占位轴全部取消,控制卡轴号和连接的EtherCAT驱动器轴号一一对应。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 1 个参数，参数的详细信息如下。	
core	内核，正整数，GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
指令返回值		
相关指令	无。	
指令示例		

指令 36GTN_SetMcAxisGap

指令原型	short GTN_SetMcAxisGap(short core,short axis,short count);	
指令说明	设置控制卡的占位轴信息。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 3 个参数，参数的详细信息如下。	
core	内核，正整数，GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2	
axis	轴号，正整数。1 到控制卡的最大轴数。	

新增指令

count	设置连续的占位轴的个数。
指令返回值	
相关指令	无。
指令示例	

指令 37GTN_GetMcAxisGap

指令原型	short GTN_GetMcAxisGap(short core,short axis,short *pActualAxisNo,short count);
指令说明	读取控制卡的占位轴信息。
指令类型	立即指令，调用后立即生效。 章节页码
指令参数	该指令共有 4 个参数，参数的详细信息如下。
core	内核，正整数，GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2
axis	轴号，1 到控制卡的最大轴数。
actualAxisNo	应为一个长度大于等于 count 的数组，返回的值：-1 表示对应控制卡轴号是占位轴。非-1 时该值为 1 到控制卡最大轴数的数值，表示控制卡轴号所对应的 EtherCAT 驱动器轴号。
count	读取连续的占位轴的个数。
指令返回值	
相关指令	无。
指令示例	

15、PVT 缓冲区输出 DO 指令

PVT 缓冲区输出 DO 指令是指在 PVT 运动过程中的某一时间点，控制 MC_GPO 上的 DO 位输出电平。

由于 GEN 卡本身没有 GPO 资源，需要使用指令 GTN_RelateEcatSlaveToMcGpoBit 进行外部资源的映射。

表 14 占位轴指令列表

GTN_PvtTableUserData	PVT 缓冲区输出 DO
--------------------------------------	--------------

指令 38 GTN_PvtTableUserData

指令原型	Short GTN_PvtTableUserData(short core,short tableId,short userDataType,double time,void *pData);
指令说明	PVT缓冲区输出DO，需要先映射外部GPO资源。
指令类型	PVT 缓冲区指令，PVT 执行过程中生效。 章节页码
指令参数	该指令共有 5 个参数，参数的详细信息如下。
core	内核，正整数，GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2
tableId	PVT 数据表的表号，GEN 8 版本取值范围 1-8，GEN16 版本取值范围 1-16。，GEN 32,64 版本取值范围 1-32。
userDataTyp	输出数据类型，取值范围

e	#define CRD_OPERATION_TYPE_BUF_DO_BIT (94)
time	DO 输出在 PVT 数据表内的时间偏移量。
pData	输入的数据类型 typedef struct { short doType; //控制器 DO 类型, 支持 MC_GPO short index; //控制器 DO 索引 取值范围 1-16 short value; //输出的数值 }TPVTDobit;
指令返回值	参考指令返回表。
相关指令	无。
指令示例	

16、设置位置比较输出阈值功能

指令 39 GTN_SetPosCompareThresholdValue

指令原型	short GTN_SetPosCompareThresholdValue(short core,short index,short thresholdValue)
指令说明	设置位置比较输出阈值。
指令参数	该指令共有 3 个参数, 参数的详细信息如下。
core	内核, 正整数, GEN 8,16,32 版本取值范围 1, GEN 64 版本取值范围 1-2
index	位置比较输出通道, 正整数, 取值范围为[1,2]。
thresholdValue	位置比较输出阈值, 正整数, 取值范围为[0,31]。
指令返回值	参考指令返回表。
相关指令	无。
指令示例	

指令 40 GTN_GetPosCompareThresholdValue

指令原型	short GTN_SetPosCompareThresholdValue(short core,short index,short *pThresholdValue)
指令说明	读取位置比较输出阈值。
指令参数	该指令共有 3 个参数, 参数的详细信息如下。
core	内核, 正整数, GEN 8,16,32 版本取值范围 1, GEN 64 版本取值范围 1-2
index	位置比较输出通道, 正整数, 取值范围为[1,2]。
pThresholdValue	位置比较输出阈值, 正整数, 取值范围为[0,31]。
指令返回值	参考指令返回表。
相关指令	无。

17、读取插补运动目标合成速度倍率

指令 41 GTN_GetMcVar

指令原型	short GT_API GTN_GetMcVar(short core,const TWatchVar *pVar,double *pValue)
指令说明	读取控制卡资源数据。
指令参数	该指令共有 3 个参数，参数的详细信息如下。
core	内核，正整数，GEN 8,16,32 版本取值范围 1，GEN 64 版本取值范围 1-2
pVar	资源数据结构体 typedef struct { unsigned short type; unsigned short index; unsigned short id; } TWatchVar; type: 资源数据类型。 index: 资源数据的逻辑索引。 id: 资源数据的二级逻辑索引。 WATCH_VAR_CRD_OVERRIDE:
pValue	获取的资源数据对应的数值。
指令返回值	参考指令返回表。
相关指令	无。
指令示例	

18、多通道 PDO 数据读写

指令 42 GTN_GetEcatRawDataPro

指令原型	short GTN_GetEcatRawDataPro(short core, unsigned short offset, unsigned short nByteSize, short mode,unsigned char *pValue ,short chn)	
指令说明	获取控制器 PDO 数据。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 6 个参数，参数的详细信息如下。	
core	内核，正整数[1,2]。	
offset	读取 PDO 数据偏移值，取值范围：0 ~当前 PDO 长度，单位：字节。	
nByteSize	读取数据长度，取值范围：[1,56]，单位：字节。	
mode	PDO 模式，0：收发独立，1：收发复用。默认 0	
pValue	读取数据缓冲区地址	
chn	PDO 所在的通道号。	

指令返回值	若返回值为 1：请检查 EtherCAT 通讯是否正常。 其它返回值：请参照指令返回值列表。
相关指令	GTN_SetEcatRawData
指令示例	无

指令 43 GTN_SetEcatRawDataPro

指令原型	shortGTN_SetEcatRawDataPro(short core, unsigned short offset, unsigned short nByteSize, unsigned char *pValue ,short chn)	
指令说明	设置控制器 PDO 数据。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 5 个参数，参数的详细信息如下。	
core	内核，正整数[1,2]。	
offset	设置 PDO 数据偏移值，取值范围：0 ~当前 PDO 长度，单位：字节。	
nByteSize	设置数据长度，取值范围：[1,56]，单位：字节。	
pValue	设置数据缓冲区地址。	
chn	PDO 所在的通道号。	
指令返回值	若返回值为 1：请检查 EtherCAT 通讯是否正常。 其它返回值：请参照指令返回值列表。	
相关指令	GTN_GetEcatRawData	
指令示例	无	

19、多程序开卡

多程序开卡功能是指在多个应用程序（最多支持 4 个）同时访问控制卡的情形，每个应用程序均需要调用 GTN_Open 指令打开控制卡。

在不使用 EtherCAT 网络功能时，可以直接使用该功能操作控制卡。

在使用 EtherCAT 网络指令时，需要在初始化网络时使用 GTN_InitEcatComm_MultiTask 指令以避免网络被重复初始化而意外断开或运动异常。

指令 44 GTN_InitEcatComm_MultiTask

指令原型	short GTN_InitEcatComm_MultiTask(short core,TEcatInitPrm *pPrm)	
指令说明	多程序方式初始化 EtherCAT 网络。	
指令类型	立即指令，调用后立即生效。	章节页码
指令参数	该指令共有 2 个参数，参数的详细信息如下。	
core	内核，正整数[1,2]。	
pPrm	<pre>typedef struct { unsigned short skip_count; //跳站功能预留参数，默认填0 unsigned short netOpenSts; //初始化操作返回参数。1：表示指令调用时</pre>	

	<p>EtherCAT网络没有初始化，指令进行了初始化从站到OP状态操作；2：表示指令调用时EtherCAT网络已经初始化在OP状态正常工作，指令未对从站进行操作；</p> <pre> unsigned short *slave_position; //跳站功能预留参数，默认填0 char *eniFilePath; //配置文件路径参数，使用库文件目录时，配置文件Gecat.xml存放在库文件目录，该参数可以为NULL。 unsigned short reserve2[20]; //预留参数，默认填0 }TEcatInitPrm; </pre>
<p>指令返回值</p>	<p>若返回值为 1：请检查 EtherCAT 通讯是否正常。 其它返回值：请参照指令返回值列表。</p>
<p>相关例程</p>	<pre> char path[100] = "D:\\GENGitDevelop\\Gecat.xml"; TEcatInitPrm Prm; rtn = GTN_Open(5,1); memset((char *) &Prm,0,sizeof(TEcatInitPrm)); Prm.eniFilePath =path; rtn = GTN_InitEcatComm_MultiTask(1,&Prm); if(2 != Prm.netOpenSts) //在重复开卡时跳过复位控制卡和启动网络操作 { rtn =GTN_Reset(1); rtn = GTN_ZeroPos(1,1,6); do { rtn = GT_IsEcatReady(&ecatsts); Sleep(100); delaycnt++; }while((0 == ecatsts)&&(delaycnt <150)); if(delaycnt < 150) { rtn = GT_StartEcatComm(); } } </pre>
<p>指令示例</p>	<p>无</p>