

## 目 录

第 1 章 报文介绍 .....	1
1.1 支持的报文 .....	1
1.2 I/O 数据信号 .....	2
1.3 控制字和状态字 .....	3
1.3.1 STW1 和 ZSW1 .....	3
1.3.2 STW2 和 ZSW2 .....	5
1.3.3 SATZANW、AKTSATZ、MDI_MOD .....	5
1.3.4 POS_STW1、POS_STW2、POS_ZSW1、POS_ZSW2 .....	6
1.3.5 G1_STW、G1_ZSW、MELDW .....	8
第二章 应用说明 .....	10
第三章 非周期通信 .....	11
3.1 驱动器 IP 和设备名设置 .....	11
3.2 组态配置 .....	12
3.3 SINA_PARA_S (FB287) 介绍 .....	14
3.4 FB287 读写参数示例 .....	15
3.5 使用 WRREC 与 RDREC 读写参数 .....	15
3.6 故障诊断功能 .....	18
第四章 应用类 1 .....	20
4.1 概述 .....	20
4.2 组态配置 .....	20
4.3 SINA_SPEED (FB285) 介绍 .....	23
第五章 应用类 3 .....	24
5.1 概述 .....	24
5.2 组态配置 .....	24
5.3 SINA_POS (FB284) 介绍 .....	27
5.4 FB284 功能说明 .....	29
5.4.1 运行条件和设置 .....	29
5.4.2 运行模式 1 (相对定位) .....	32
5.4.3 运行模式 2 (绝对定位) .....	33
5.4.4 运行模式 4 (主动回原点) .....	33
5.4.5 运行模式 5 (直接回原点) .....	34
5.4.6 运行模式 6 (位置表) (未支持) .....	34
5.4.7 运行模式 7 (按指定速度点动) .....	35

第六章 应用类 4	36
6.1 概述	36
6.2 组态配置	36
6.3 MC_Home (回原点) 介绍	42
6.3.1 回原点模式 0 (绝对式直接回原点)	45
6.3.2 回原点模式 1 (相对式直接回原点)	46
6.3.3 回原点模式 2 (被动回原点)	46
6.3.4 回原点模式 3 (主动回原点)	48
6.3.5 回原点模式 6 (绝对编码器相对调节)	49
6.3.6 回原点模式 7 (绝对编码器绝对调节)	49
6.4 MC_TorqueLimiting (转矩限制) 介绍	49
6.5 DSC (动态伺服控制) 介绍	50
第七章 S7-200 SMART 应用	52
7.1 应用说明	52
7.2 非周期通信	52
7.2.1 项目配置	52
7.2.2 SINA_PARA_S 介绍	56
7.2.3 SINA_PARA_S 读写参数示例	57
7.3 报文 1 应用	58
7.3.1 项目配置	58
7.3.2 SINA_SPEED 介绍	61
7.4 报文 111 应用	62
7.4.1 项目配置	62
7.4.2 SINA_POS 介绍	65
7.4.3 运行条件和设置	66
7.4.4 运行模式 1 (相对定位)	67
7.4.5 运行模式 2 (绝对定位)	68
7.4.6 运行模式 4 (主动回原点)	68
7.4.7 运行模式 5 (直接回原点)	69
7.4.8 运行模式 7 (按指定速度点动)	69

# 第 1 章 报文介绍

PROFINET 由 PROFIBUS 国际组织 (PROFIBUS International, PI) 推出, 是新一代基于工业以太网技术的自动化总线标准; 它可以满足工业自动化场景下的各种通讯要求, 是一种支持高性能、高可靠性、高实时性的工业网络协议。

PROFINET 提供了三种通道: 标准通道、实时通道 (RT)、等时实时通道 (IRT), 其中标准通道是基于 TCP/IP 协议的非实时协议, 而 RT 和 IRT 通道是基于 Ethernet 实现的。

## 1.1 支持的报文

Kinco PN 伺服(PN 固件 00000005、10000005)支持报文 1、3、5、9、102、105 和 111, 涵盖应用类 (AC, Application Class) 1、3 和 4, 如表 1-1 所示。暂不支持附加报文 750。

表 1-1 可用的报文

应用类	报文号
AC1	1
AC3	9、111
AC4	3、5、102、105



### 注意

- PN 固件 00000004 仅支持报文 1、111 和 3(不支持 IRT)。PN 固件版本可通过[308001]查看, 详见表 2-1。

各报文的结构如下所示:

表 1-2 报文 1 (应用类 1)

I/O 数据号	报文 1	
	设定值	实际值
1	STW1	ZSW1
2	NSOLL_A	NIST_A

表 1-3 报文 9、111 (应用类 3)

I/O 数据号	报文 9		报文 111	
	设定值	实际值	设定值	实际值
1	STW1	ZSW1	STW1	ZSW1
2	SATZANW	AKTSATZ	POS_STW1	POS_ZSW1
3	STW2	ZSW2	POS_STW2	POS_ZSW2

4	MDI_TARPOS	XIST_A	STW2	ZSW2
5			OVERRIDE	MELDW
6	MDI_VELOCITY		MDI_TARPOS	XIST_A
7				
8	MDI_ACC		MDI_VELOCITY	NIST_B
9	MDI_DEC			
10	MDI_MOD		MDI_ACC	FAULT_CODE
11			MDI_DEC	WARN_CODE
12			User	User

表 1-4 报文 3、5、102、105 (应用类 4)

I/O 数据号	报文 3		报文 102		报文 5		报文 105	
	设定值	实际值	设定值	实际值	设定值	实际值	设定值	实际值
1	STW1	ZSW1	STW1	ZSW1	STW1	ZSW1	STW1	ZSW1
2	NSOLL_B	NIST_B	NSOLL_B	NIST_B	NSOLL_B	NIST_B	NSOLL_B	NIST_B
3								
4	STW2	ZSW2	STW2	ZSW2	STW2	ZSW2	STW2	ZSW2
5	G1_STW	G1_ZSW	MOMRED	MELDW	G1_STW	G1_ZSW	MOMRED	MELDW
6		G1_XIST1	G1_STW	G1_ZSW	XERR	G1_XIST1	G1_STW	G1_ZSW
7								
8		G1_XIST2		G1_XIST1	KPC	G1_XIST2	XERR	G1_XIST1
9								
10				G1_XIST2			KPC	G1_XIST2

## 1.2 I/O 数据信号

I/O 数据信号 (设定值和实际值) 用于构成报文, 下表提供了 I/O 数据的所有信号。

表 1-5 信号列表

信号	缩写符	数据类型	说明
控制字 1	STW1	U16	见 1.3.1 节
状态字 1	ZSW1	U16	
控制字 2	STW2	U16	见 1.3.2 节
状态字 2	ZSW2	U16	
速度设定值 A	NSOLL_A	I16	4000h 对应参考速度 <sup>a</sup>
速度实际值 A	NIST_A	I16	
速度设定值 B	NSOLL_B	I32	40000000h 对应参考速度
速度实际值 B	NIST_B <sup>b</sup>	I32	
编码器 1 控制字	G1_STW	U16	见 1.3.5 节
编码器 1 状态字	G1_ZSW	U16	
消息状态字	MELDW	U16	

信号	缩写符	数据类型	说明
编码器 1 位置实际值 1	G1_XIST1	I32	等于实际位置[606300]与位置偏移[60FB07]之和, 单位 DEC
编码器 1 位置实际值 2	G1_XIST2	I32	—
位置偏差值	XERR	I32	—
位置控制系数	KPC	I32	—
位置实际值 A	XIST_A	I32	等于实际位置[606300], 单位 DEC
位置表控制字	SATZANW	U16	见 1.3.3 节
位置表状态字	AKTSATZ	U16	
MDI 模式控制字	MDI_MOD	U16	
位置控制字 1	POS_STW1	U16	见 1.3.4 节
位置状态字 1	POS_ZSW1	U16	
位置控制字 2	POS_STW2	U16	
位置状态字 2	POS_ZSW2	U16	
MDI 目标位置	MDI_TARPOS	I32	等于目标位置[607A00], 单位 DEC
MDI 速度	MDI_VELOCITY	U32	单位 DEC, $DEC = (RPM * \text{反馈精度}[641003] * 512) / 1875$
MDI 加速度百分比	MDI_ACC	U16	0~4000h 对应 0~100%, 2000h 表示将加/减速度缩小为伺服设定值的 1/2
MDI 减速度百分比	MDI_DEC	U16	
MDI 速度百分比	OVERRIDE	U16	0~4000h 对应 0~100%, 最大 199%, 2000h 表示将速度缩小为伺服设定值的 1/2
转矩限制值	MOMRED	I16	0~4000h 对应 100%~0, 4000h 表示将目标电流(转矩)限制到 0
故障代码	FAULT_CODE	U16	—
警告代码	WARN_CODE	U16	—
报文 111 自定义接收字	User	I16	见 5.4.1 节第 7 点
报文 111 自定义发送字	User	I16	
a 参考速度 = 最大速度[607F00]的 1/2。			
b 使用报文 111 时, 速度实际值 NIST_B 等于实际速度[606C00], 单位 DEC。			

## 1.3 控制字和状态字

### 1.3.1 STW1 和 ZSW1

- 控制字 1 (STW1)

表 1-6 STW1 中 Bit 值意义

Bit	意义	
	报文 1、3、5、102、105 (应用类 1、4)	报文 9、111 (应用类 3)
0	开启 / 关闭 (斜坡停止 <sup>a</sup> )	
1	非惯性停止 / 惯性停止	

2	非快速停止 / 快速停止	
3	操作使能 / 操作禁止	
4	启用 RFG <sup>b</sup> / 禁用 RFG (RFG 输出为 0)	接受定位任务 / 拒绝定位任务
5	解冻 RFG / 冻结 RFG (RFG 输入不更新)	不暂停定位任务 / 暂停定位任务
6	设定值有效 / 设定值无效 (RFG 输入为 0)	激活定位任务 (0→1)
7	故障确认 (0→1)	
8	保留 (应用类 1、4 不支持点动)	反向点动启动 / 反向点动停止
9	保留	正向点动启动 / 正向点动停止
10	PLC 控制 / 无 PLC 控制	
11	设定值取反 / 设定值不取反	回原点 / 停止回原点
12~15	保留	
a STW1 的比特 4 为 1 时, 斜坡停止对应暂停模式[605D00], 停止后松轴; STW1 的比特 4 为 0 时, 斜坡停止对应关机停止模式[605B00]。		
b RFG = Ramp Function Generator (梯形曲线速度发生器)。使用应用类 4 时, 比特 5 不起作用, 比特 4 或比特 6 为 0 均会导致 RFG 输出为 0。		
<b>说明: Bit 值为 1 时取 “/” 左侧的含义; Bit 值为 0 时取 “/” 右侧的含义。</b>		

- 状态字 1 (ZSW1)

表 1-7 ZSW1 中 Bit 值意义

Bit	意义	
	报文 1、3、5、102、105 (应用类 1、4)	报文 9、111 (应用类 3)
0	接通准备就绪 / 未准备接通	
1	操作准备就绪 / 未准备操作	
2	操作使能 / 操作禁止	
3	有故障 / 无故障	
4	惯性停止未激活 / 惯性停止激活	
5	快速停止未激活 / 快速停止激活	
6	接通禁止 / 接通未禁止	
7	有警告 / 无警告	
8	速度误差在容差内 / 速度误差超出容差	跟踪误差在容差内 / 跟踪误差超出容差
9	有控制请求 / 无控制请求	
10	速度达到或超过参考速度 / 速度未达到参考速度	达到目标位置 / 未达到目标位置
11	保留	原点已设定 / 原点未设定
12	保留	定位任务确认 (0→1)
13	保留	轴已静止 / 轴移动中
14	正向移动 / 反向移动	加速中 / 不在加速
15	保留	减速中 / 不在减速
<b>说明: 比特值为 1 时取 “/” 左侧的含义; 比特值为 0 时取 “/” 右侧的含义。</b>		

### 1.3.2 STW2 和 ZSW2

- 控制字 2 (STW2)

用于报文 3、5、102、105，仅在等时同步模式下生效。

表 1-8 STW2 中的 Bit 值意义

Bit	意义
0~11	保留
12~15	PLC 心跳信号



**注意**

- 若出现 PLC 心跳信号错误或其它通讯异常，驱动器会产生“CAN 总线故障 (报警代码 100.0)”的报警。用户可将通讯中断模式[600700]设置为 1：报错处理；设置为 0：不处理。

6004	00	电机绝对位置
6007	00	通讯中断模式
603F	00	错误代码
6040	00	控制字
6041	00	状态字
605A	00	快速停止模式

Value
1

帮助: 通讯中断模式  
CAN 通讯中断模式  
0: 不处理  
1: 报错

- 报警时间阈值由节点保护时间[100C00](单位 ms)和节点保护时间系数[100D00]的乘积确定，默认值为 1000ms \* 3 = 3000 ms。

N	Index	Type	Name	Value	Unit
0*	101801	uint32	设备厂商代码	000005EF	HEX
1	301107	uint16	ECAN 同步数据	0000	HEX
2	100500	uint32	同步 ID	?????	HEX
3	100C00	uint16	节点保护时间	1000	DEC
4	100D00	uint8	节点保护时间系数	3	DEC

- 状态字 2 (ZSW2)

用于报文 3、5、102、105，仅在等时同步模式下生效。

表 1-9 ZSW 2 中的 Bit 值意义

Bit	意义
0~11	保留
12~15	驱动器心跳信号

### 1.3.3 SATZANW、AKTSATZ、MDI\_MOD

- 位置表控制字 (SATZANW)

用于报文 9。

表 1-10 SATZANW 中的 Bit 值意义

Bit	意义
0~2	位置表中即将启动的定位任务的索引 (取值范围: 0~7) Bit 0~2 只在位置表模式中生效
3~14	保留
15	工作模式选择 = 1 选择 MDI 模式 = 0 选择位置表模式

- 位置表状态字 (AKTSATZ)

用于报文 9。

表 1-11 AKTSATZ 中的 Bit 值意义

Bit	意义
0~4	实际有效的定位任务的索引 (取值范围: 0~31) 比特 0~4 只在位置表模式中生效
5~14	保留
15	工作模式 = 1 MDI 模式被激活 = 0 位置表模式被激活

- MDI 模式控制字 (MDI\_MOD)

用于报文 9。

表 1-12 MDI\_MOD 中的 Bit 值意义

Bit	意义
0	选择绝对/相对定位模式 = 1 绝对定位模式 = 0 相对定位模式
1~15	保留

### 1.3.4 POS\_STW1、POS\_STW2、POS\_ZSW1、POS\_ZSW2

- 位置控制字 1 (POS\_STW1)

用于报文 111。

表 1-13 POS\_STW1 中的 Bit 值意义

Bit	意义
0~2	位置表中即将启动的定位任务的索引 (取值范围: 0~7) 比特 0~2 只在位置表模式中生效
3~7	保留

8	选择绝对/相对定位模式 = 1 绝对定位模式 = 0 相对定位模式
9~11	保留
12	绝对定位模式 = 1 根据目标位置变化立即执行绝对定位指令 = 0 不会根据目标位置变化立即执行绝对定位指令
13~14	保留
15	工作模式选择 = 1 选择 MDI 模式 = 0 选择位置表模式

- 位置控制字 2 (POS\_STW2)

用于报文 111。

表 1-14 POS\_STW2 中的 Bit 值意义

Bit	意义
0	保留
1	= 1 把当前位置设置成原点
2	= 1 激活原点开关信号
3~13	保留
14	= 1 激活软限位开关信号
15	= 1 激活硬限位开关信号

- 位置状态字 1 (POS\_ZSW1)

用于报文 111。

表 1-15 POS\_ZSW1 中的 Bit 值意义

Bit	意义
0~4	实际有效的定位任务的索引 (取值范围: 0~31) 比特 0~4 只在位置表模式中生效
5~7	保留
8	= 1 负向硬限位开关信号激活
9	= 1 正向硬限位开关信号激活
10	= 1 JOG 模式激活
11~12	保留
13	= 1 运行位置表激活
14	保留
15	= 1 MDI 激活 = 0 MDI 未激活

- 位置状态字 2 (POS\_ZSW2)

用于报文 111。

表 1-16 POS\_ZSW2 中的 Bit 值意义

Bit	意义
0~3	保留
4	= 1 正向移动 = 0 非正向移动
5	= 1 反向移动 = 0 非反向移动
6	= 1 达到负向软限位开关
7	= 1 达到正向软限位开关
8~15	保留

### 1.3.5 G1\_STW、G1\_ZSW、MELDW

- 编码器 1 控制字 (G1\_STW)

用于报文 3、5、102、105。

表 1-17 G1\_STW 中的 Bit 值意义

Bit	值	意义
0	1	功能 1 (Bit 7=0): 请求搜索编码器索引信号位置
1~3	—	保留
4~6	1~3	命令: 0: — 1: 激活功能 1 <sup>a</sup> 2: 读取编码器索引信号位置, 放入 G1_XIST2 中 3: 取消功能 1 4~7: —
7	0/1	模式: 比特 7=0: 编码器索引信号位置搜索 比特 7=1: 保留
8~12	—	保留
13	1	请求在 G1_XIST2 中传输绝对实际位置
14	1	请求关闭编码器的实际值测量
15	1	请求复位编码器错误

<sup>a</sup> 功能 1 (Bit0) 和命令 (Bit 4~Bit7) 的选择应同步设置。

- 编码器 1 状态字 (G1\_ZSW)

用于报文 3、5、102、105。

表 1-18 G1\_ZSW 中的 Bit 值意义

Bit	值	意义
0	1	状态: 功能 1 被激活
1~3	—	保留
4	1	状态: 编码器索引信号位置有效
5~10	—	保留
11	1	检测到复位编码器错误的请求
12	—	保留
13	1	在 G1_XIST2 中传输绝对实际位置的指示
14	1	确认关闭编码器的实际值测量
15	1	编码器错误

- 消息状态字 (MELDW)

用于报文 102、105。

表 1-19 MELDW 中的 Bit 值意义

Bit	意义
0	保留
1	= 1 转矩未达到限制值 = 0 转矩已达到限制值
2~15	保留

## 第二章 应用说明

第三章至第六章的 PROFINET 通讯应用基于以下条件：

1. TIA V17 PLC S7-1500(CPU 1511T-1 PN 6ES7 511-1TK01-0AB0 固件 V2.8)
2. Kinco PN 伺服

伺服型号	伺服固件	PN 固件
FDxx5-PA-004	版本年份为 2024 年及以后	00000005、10000005
FDxx5P-PA-000		
MDx0-0xx-DMxK-PA-000		

3. GSD 文件

PN 固件版本	适用 GSD 文件
00000005、10000005	GSDML-V2.43-Kinco-PA5-20240328、GSDML-V2.33-Kinco-PA5-20240328
00000004	GSDML-V2.33-Kinco-MD60-20210507

说明：

- a. 默认为 GSDML-V2.43-Kinco-PA5-20240328
- b. 如果 PLC 不支持 V2.43 版本 GSD，可使用 GSDML-V2.33-Kinco-PA5-20240328

第七章介绍了 S7-200 SMART 控制 Kinco PN 伺服的方法。

伺服的 PROFINET 通讯相关参数说明见表 2-1。

表 2-1 PROFINET 通讯相关参数

参数索引	参数名称	说明
308001	PN 软件版本	本手册适用的 PN 固件版本为 00000005、10000005
308003	PN 设备名称	PROFINET 设备名称 (只读)
308004~308007	PN 设备名称 1~4	用于修改 PROFINET 设备名称
308008	IP 地址	驱动器 IP 地址 (十六进制, 只读)
308009	子网掩码	驱动器子网掩码 (十六进制, 只读)
30800A	默认网关	驱动器默认网关 (十六进制, 只读)
30800B	PN 使能	= 1 PN 报文使能, 使用报文 (如报文 105) 控制 = 0 PN 报文断开, 使用 RS232 或非周期通信 (参数读写) 控制
30800C	软件限位使能	见 5.4.1 节第 5 点
30800D	PN 用户自定义接收字	见 5.4.1 节第 8 点
30800E	PN 用户自定义发送字	
30800F~308010	MAC 地址 1~2	驱动器 MAC 地址 (十六进制, 只读)
301101	ECAN 同步周期	可选 1ms、2ms、4ms、8ms 需要与博途软件组态的更新时间一致
301102	ECAN 时钟同步模式	= 1 使用等时同步模式 = 0 不使用等时同步模式
301103	ECAN 同步点偏移	—
600700	通讯中断模式	见 1.3.2 节
100C00	节点保护时间	
100D00	节点保护时间系数	

## 第三章 非周期通信

### 3.1 驱动器 IP 和设备名设置

驱动器的 PROFINET 设备名称可用 KincoServo 上位机软件、博途软件或 PRONETA 软件设置；驱动器的 IP 地址只能使用博途软件或 PRONETA 软件设置。请注意，如果要使用下述方法设置 IP 地址，则博途组态中应选择“在设备中直接设定 IP 地址”：



在 KincoServo 上位机软件设置设备名的步骤如下：

1. 断开驱动器与 PLC 的通讯，通过 RS232 将驱动器连接到上位机。
2. 在 PN 设备名称 1~4[308004~308007]输入设备名，每个 PN 设备名称长度限制为 4 个字符：

3080	04	PN设备名称1	Value	fd42
3080	05	PN设备名称2		
3080	06	PN设备名称3		
3080	07	PN设备名称4		
			帮助:PN设备名称1	
			PN设备名称成员1	
3080	04	PN设备名称1	Value	5-1
3080	05	PN设备名称2		
3080	06	PN设备名称3		
3080	07	PN设备名称4		
			帮助:PN设备名称2	
			PN设备名称成员2	

3. 设置完毕，可见 PN 设备名称[308003]已被修改：

3080	02	网口状态	Value	fd425-1
3080	03	PN设备名称		
3080	04	PN设备名称1		
3080	05	PN设备名称2		
			帮助:PN设备名称	
			PN设备名称	

在博途软件设置驱动器 IP 和设备名的步骤如下：

1. 用网线将驱动器连接到电脑。
2. 打开博途软件，展开“在线访问”下拉列表，选择相应网卡，点击“更新可访问的设备”。待软件扫描到驱动器之后，双击“在线和诊断”。
3. 在“功能”下拉列表中双击“分配 IP 地址”。填入 IP 地址和子网掩码后，点击“分配 IP 地址”按钮即可。如果在组态中选择“在项目中设置 IP 地址”，则该步可省略，直接到第 4 步设置 PROFINET 设备名称即可。



4. 在“功能”下拉列表中双击“分配 PROFINET 设备名称”。填入 PROFINET 设备名称后，点击“分配名称”按钮即可。PROFINET 设备名称必须与组态中的名称一致，否则无法与 PLC 通讯。



5. 设置完毕，再次点击“更新可访问的设备”，查看 IP 地址和设备名是否修改成功。

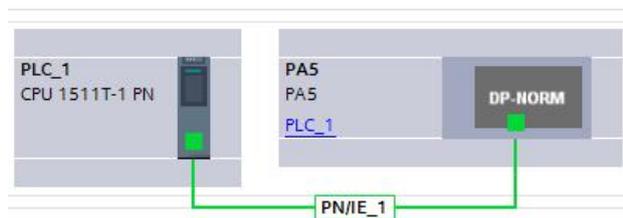
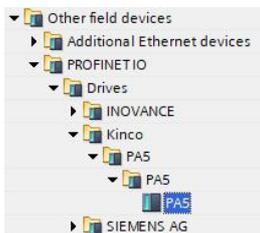
另外，也可以使用 PRONETA 软件设置驱动器 IP 地址和设备名称，PRONETA 的下载和使用方法见西门子官网：[PRONETA - Siemens China](http://www.siemens.com/PRONETA)。

## 3.2 组态配置

1. 点击上方菜单“选项 → 管理通用站描述文件”安装 GSD，请以 GSDML-V2.43-Kinco-PA5-20240328 为准。另外，PN 固件 00000004 适用的 GSD 为 GSDML-V2.33-Kinco-MD60-20210507。



- 在网络视图下，点击“硬件目录→其它现场设备→PROFINET IO→Drives→Kinco→PA5”，添加伺服并为其分配 PLC：

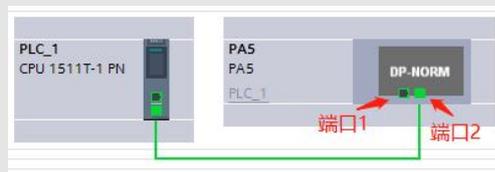
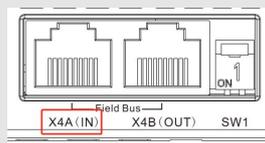


- 如果需要 PLC 自动配置驱动器设备名称（自动分配的前提是驱动器内部的名称为空），则必须连接拓扑图，否则可以不连接。在拓扑视图下，根据实际情况连接端口。

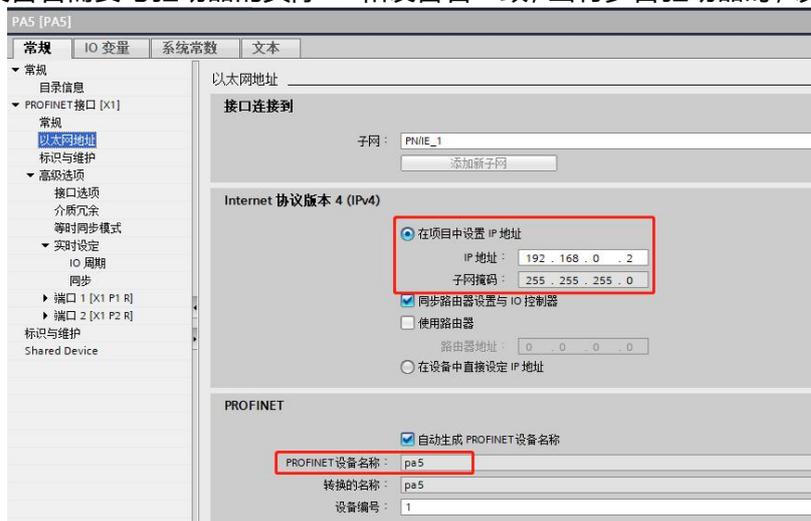


**注意**

- 不同系列的驱动器上的 IN 口对应的拓扑图上的端口有所区别，FD5P 系列驱动器和 MD 系列一体机的 IN 口对应拓扑图中的端口 1，FD5 系列驱动器的 IN 口对应拓扑图中的端口 2



- 双击 PA5 模块，在“常规”中点击“以太网地址”，设置组态的 IP 地址和 PROFINET 设备名称。组态的 IP 和设备名需要与驱动器的实际 IP 和设备名一致；当有多台驱动器时，设备名称不能相同：



- 点击“系统常数”栏，记下 Interface 的硬件标识符，后面读写参数模块会使用到：

名称	类型	硬件标识符	使用者
PA5-Interface-端口_1	Hw_Interface	260	PLC_1
PA5-Interface-端口_2	Hw_Interface	261	PLC_1
PA5-Interface	Hw_Interface	259	PLC_1
PA5-Proxy	Hw_SubModule	258	PLC_1
PA5-Head	Hw_SubModule	262	PLC_1

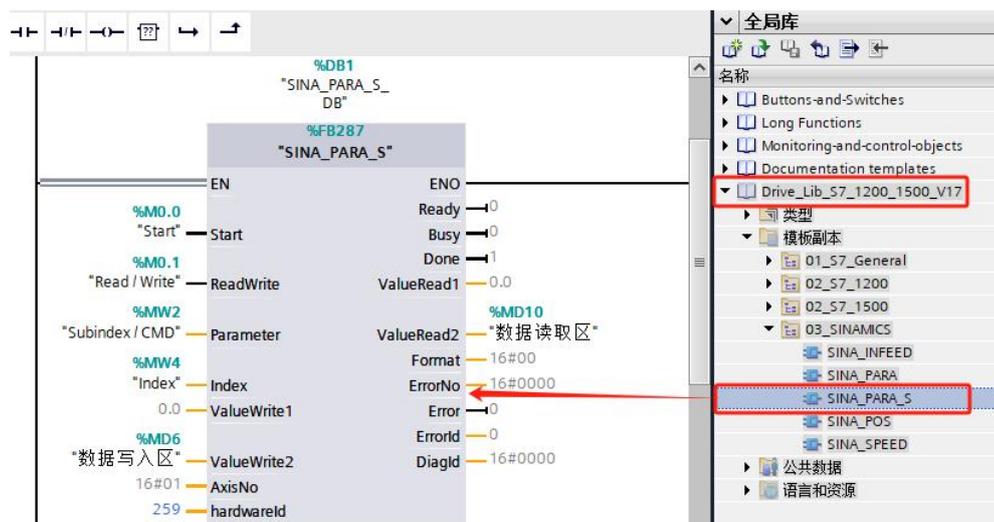
### 3.3 SINA\_PARA\_S (FB287) 介绍

FB287 可以实现 PLC 与驱动器的非周期数据交互，实现伺服参数读写功能。

如果找不到相应的功能块 (FB)，可到西门子官网下载库文件 Drive\_Lib\_S7\_1200\_1500\_V17，网址如下：

<https://support.industry.siemens.com/cs/document/109475044/sinamics-communication-blocks-drivelib-for-reading-and-writing-drive-data-within-tia-portal-context?dti=0&lc=en-WW>

下载完成后，点击博途软件上方的菜单“选项→全局库→打开库”，按照提示安装对应的库文件。将 FB287 拖拽到 Main[OB1]，并为其引脚分配好变量，如下图。



FB287 引脚定义如下：

表 3-1 FB287 引脚定义

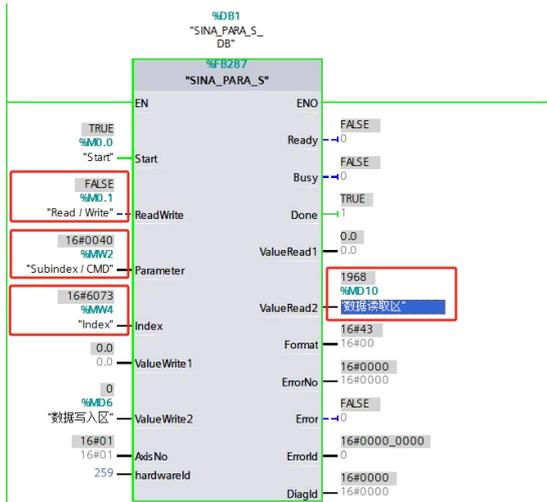
引脚名称	数据类型	说明
Start	Bool	上升沿触发数据发送
ReadWrite	Bool	= 0 读参数 = 1 写参数
Parameter	Int	比特 8~15: 参数子索引 比特 0~7: 发送命令字
Index	Int	参数索引
ValueWrite1	Real	保留
ValueWrite2	DInt	数据写入区，可写入 1~4 字节
AxisNo	Byte	固定为 1
hardwareId	HW_IO	见 2.3 节第 4 步，通过该参数来区分不同的轴
ValueRead1	Real	保留
ValueRead2	DInt	数据读取区
Error	Bool	指示读写参数错误，当 Kinco 通讯协议中“回复命令字”为 0x80 时，该位置 1

说明：参数索引及子索引、发送命令字等的含义，请参照 Kinco 伺服使用手册通讯相关章节，其中，发送命令字：读都是 40H；写 1 个字节 2FH，写 2 个字节 2BH，写 4 个字节 23H。

### 3.4 FB287 读写参数示例

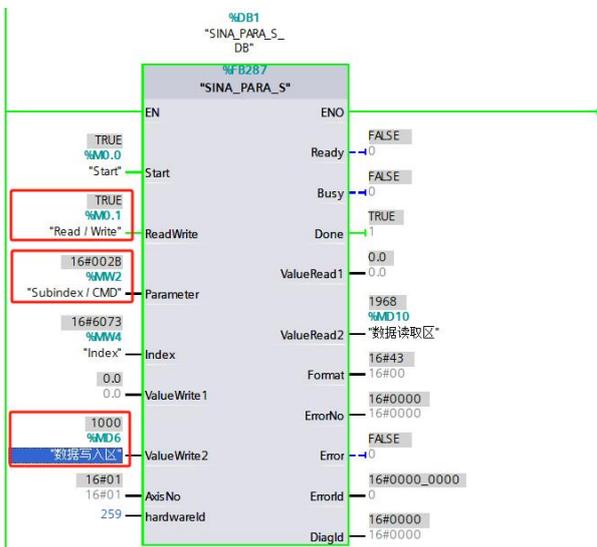
以读写目标电流限制[607300]为例：参数索引为 0x6073，子索引为 0x00，数据类型为 uint16，所以写参数的发送命令字为 0x2B，读参数的发送命令字统一为 0x40。

1. 读取目标电流限制：当“Start”引脚由0变为1后，“数据读取区”引脚自动变成1968，且“Error”引脚为0，数据读取成功，见下图。



N	Index	Type	Name	Value	Unit
0	606100	int8	有效工作模式	0	DEC
1	604100	uint16	状态字	4270	HEX
2	606300	int32	实际位置	-6	inc
3	606C00	int32	实际速度	1.36	rpm
4	607800	int16	实际电流	0.00	Ap
5	268000	uint16	警告状态字	0000	HEX
6	606000	int8	工作模式	-3	DEC
7	604000	uint16	控制字	0000	HEX
8	607A00	int32	目标位置	0	inc
9	608100	uint32	梯形速度	0.00	rpm
10	608300	uint32	梯形加速度	100.00	rps/s
11	608400	uint32	梯形减速度	100.00	rps/s
12	60FF00	int32	目标速度	0.00	rpm
13	607100	int16	目标扭矩%	0.00	%
14	607300	uint16	目标电流限制	1968	DEC
15	20200D	int8	工作模式选择0	-4	DEC
16	20200E	int8	工作模式选择1	-3	DEC
17	269000	uint8	通讯编码器数据复1	0	DEC

2. 写入目标电流限制：确认PN使能[30800B]为0，将“ReadWrite”引脚切换为1，“Parameter”改为0x002B，“数据写入区”写入1000，然后触发“Start”引脚，将数据写入。观察上位机软件，目标电流限制被改为1000 DEC。



N	Index	Type	Name	Value	Unit
0	606100	int8	有效工作模式	0	DEC
1	604100	uint16	状态字	4270	HEX
2	606300	int32	实际位置	1	inc
3	606C00	int32	实际速度	1.60	rpm
4	607800	int16	实际电流	0.00	Ap
5	268000	uint16	警告状态字	0000	HEX
6	606000	int8	工作模式	-3	DEC
7	604000	uint16	控制字	0000	HEX
8	607A00	int32	目标位置	0	inc
9	608100	uint32	梯形速度	0.00	rpm
10	608300	uint32	梯形加速度	100.00	rps/s
11	608400	uint32	梯形减速度	100.00	rps/s
12	60FF00	int32	目标速度	0.00	rpm
13	607100	int16	目标扭矩%	0.00	%
14	607300	uint16	目标电流限制	1000	DEC
15	20200D	int8	工作模式选择0	-4	DEC
16	20200E	int8	工作模式选择1	-3	DEC
17	269000	uint8	通讯编码器数据复1	0	DEC

### 3.5 使用 WRREC 与 RDREC 读写参数

下面介绍如何使用 WRREC / RDREC 块进行参数读写，这两个块可以在“指令→扩展指令→分布式 IO”中找到。

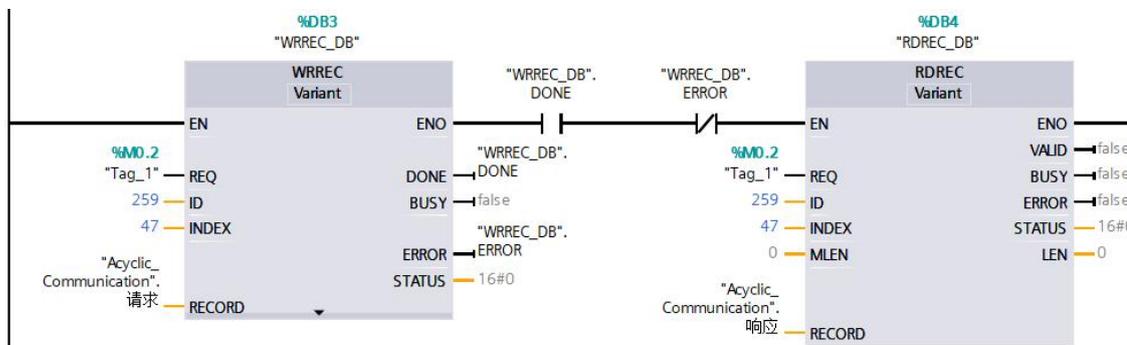
1. 首先新建一个名为Acyclic\_Communication的数据块 (DB)，块中的数据定义如下图：

Acyclic_Communication				
名称	数据类型	起始值	监视值	
Static				
请求	Struct			
ReqRef	Byte	16#01	16#01	
ReqID	Byte	16#01	16#01	
AxisID	Byte	16#01	16#01	
ParaNo	Byte	16#01	16#01	
Attribute	Byte	16#10	16#10	
Number of Element	Byte	16#01	16#01	
Subindex / CMD	Word	16#002B	16#002B	
Index	Word	16#6073	16#6073	
Write Format	Byte	16#43	16#43	
Write Values numb...	Byte	16#01	16#01	
Write Value	DInt	16#0	0	
响应	Struct			
RespRef	Byte	16#0	16#00	
RespID	Byte	16#0	16#00	
AxisID	Byte	16#0	16#00	
ParaNo	Byte	16#0	16#00	
Format	Byte	16#0	16#00	
CMD	Byte	16#0	16#00	
Value	DInt	0	0	

数据块 Acyclic\_Communication 说明如下：

请求	响应
ReqID: 0x01: 只读 0x02: 可读可写 其它: 保留	CMD: 回复命令字
Subindex / CMD: 比特 8~15: 参数子索引 比特 0~7: 发送命令字	Value: 数据读取区
Index: 参数索引	
Write Value: 数据写入区	
其它数据原则上与上图保持一致	

2. 按照下图建立 WRREC / RDREC 块的连接。“REQ” 引脚为 1 时开始传输，“ID” 为 Interface 的硬件标识符，可以通过该参数来区分不同的轴。“INDEX” 固定为 47，“MLEN” 为 0。



3. 下面利用WRREC / RDREC 块读写目标电流限制[607300]。目标电流限制参数索引为0x6073，子索引为 0x00，数据类型为 uint16，所以写参数的发送命令字为 0x2B，读参数的发送命令字统一为 0x40。

a. 读取目标电流限制：

用上位机软件查看此时的目标电流限制值为 1000 DEC。 Acyclic\_Communication 数据块的 “ReqID” 赋值 0x02，“Subindex / CMD” 赋值 0x0040，“Index” 赋值 0x6073。触发 WRREC / RDREC 块的 “REQ” 引脚完成数据读取，读取的数据在响应的 “Value” 中显示：

名称	数据类型	起始值	监视值
Static			
请求			
ReqRef	Byte	16#01	16#01
ReqID	Byte	16#01	16#02
AxisID	Byte	16#01	16#01
ParaNo	Byte	16#01	16#01
Attribute	Byte	16#10	16#10
Number of Element	Byte	16#01	16#01
Subindex / CMD	Word	16#002B	16#0040
Index	Word	16#6073	16#6073
Write Format	Byte	16#43	16#43
Write Values numb...	Byte	16#01	16#01
Write Value	Dint	16#0	0
响应			
RespRef	Byte	16#0	16#01
RespID	Byte	16#0	16#02
AxisID	Byte	16#0	16#01
ParaNo	Byte	16#0	16#01
Format	Byte	16#0	16#43
CMD	Byte	16#0	16#48
Value	Dint	0	1000

N	Index	Type	Name	Value	Unit
0	606100	int8	有效工作模式	0	DEC
1	604100	uint16	状态字	4270	HEX
2	606300	int32	实际位置	1	inc
3	606C00	int32	实际速度	1.60	rpm
4	607800	int16	实际电流	0.00	Ap
5	268000	uint16	警告状态字	0000	HEX
6	606000	int8	工作模式	-3	DEC
7	604000	uint16	控制字	0000	HEX
8	607A00	int32	目标位置	0	inc
9	608100	uint32	梯形速度	0.00	rpm
10	608300	uint32	梯形加速度	100.00	rps/s
11	608400	uint32	梯形减速度	100.00	rps/s
12	60FF00	int32	目标速度	0.00	rpm
13	607100	int16	目标扭矩%	0.00	%
14	607300	uint16	目标电流限制	1000	DEC
15	20200D	int8	工作模式选择0	-4	DEC
16	20200E	int8	工作模式选择1	-3	DEC
17	269000	uint8	通讯编码器数据复1	0	DEC

b. 写目标电流限制 1968 DEC：

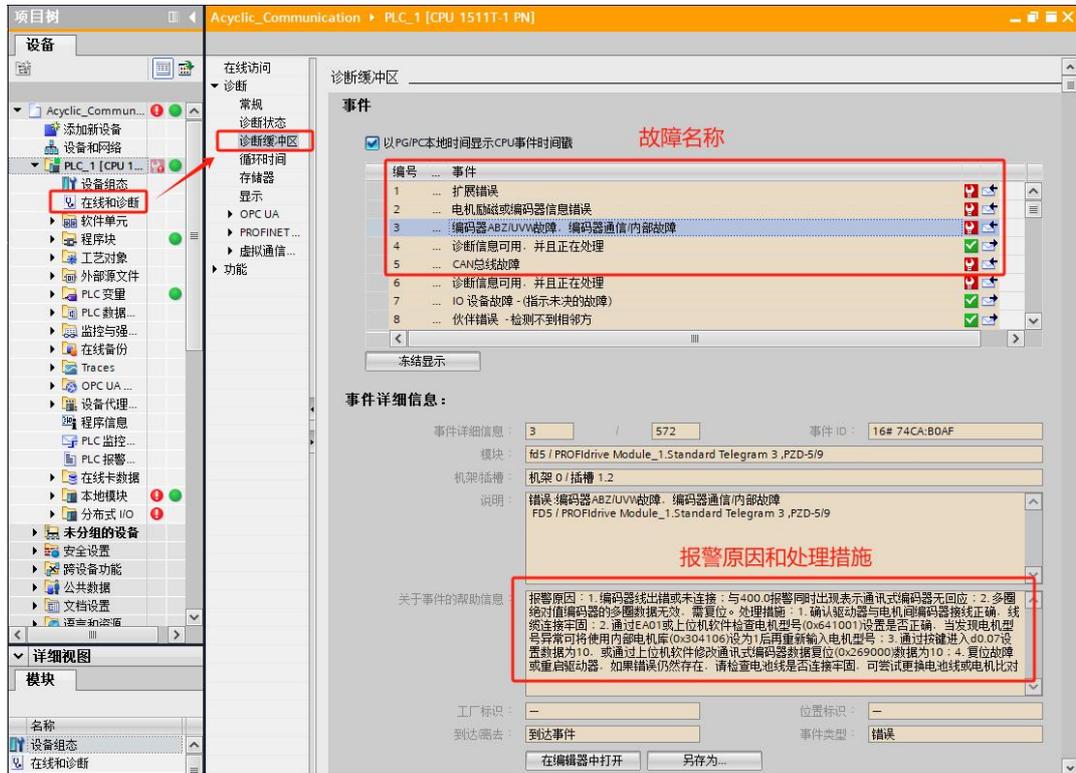
确认 PN 使能 [30800B]为 0。Acyclic\_Communication 数据块的 “ReqID” 赋值 0x02，“Subindex / CMD” 赋值 0x002B，“Index” 赋值 0x6073，“Write Value” 赋值 1968。触发 WRREC / RDREC 块的 “REQ” 引脚完成数据写入。见下图，响应的 “CMD” 为 0x60，说明写入成功，响应的 “Value” 为写入的数据 1968，用上位机软件查看目标电流限制变为 1968 DEC。

名称	数据类型	起始值	监视值
Static			
请求			
ReqRef	Byte	16#01	16#01
ReqID	Byte	16#01	16#02
AxisID	Byte	16#01	16#01
ParaNo	Byte	16#01	16#01
Attribute	Byte	16#10	16#10
Number of Element	Byte	16#01	16#01
Subindex / CMD	Word	16#002B	16#002B
Index	Word	16#6073	16#6073
Write Format	Byte	16#43	16#43
Write Values numb...	Byte	16#01	16#01
Write Value	Dint	16#0	1968
响应			
RespRef	Byte	16#0	16#01
RespID	Byte	16#0	16#02
AxisID	Byte	16#0	16#01
ParaNo	Byte	16#0	16#01
Format	Byte	16#0	16#43
CMD	Byte	16#0	16#60
Value	Dint	0	1968

N	Index	Type	Name	Value	Unit
0	606100	int8	有效工作模式	0	DEC
1	604100	uint16	状态字	4270	HEX
2	606300	int32	实际位置	-6	inc
3	606C00	int32	实际速度	1.36	rpm
4	607800	int16	实际电流	0.00	Ap
5	268000	uint16	警告状态字	0000	HEX
6	606000	int8	工作模式	-3	DEC
7	604000	uint16	控制字	0000	HEX
8	607A00	int32	目标位置	0	inc
9	608100	uint32	梯形速度	0.00	rpm
10	608300	uint32	梯形加速度	100.00	rps/s
11	608400	uint32	梯形减速度	100.00	rps/s
12	60FF00	int32	目标速度	0.00	rpm
13	607100	int16	目标扭矩%	0.00	%
14	607300	uint16	目标电流限制	1968	DEC
15	20200D	int8	工作模式选择0	-4	DEC
16	20200E	int8	工作模式选择1	-3	DEC
17	269000	uint8	通讯编码器数据复1	0	DEC

### 3.6 故障诊断功能

通过博途软件的在线诊断界面，可以查看驱动器的故障信息。双击 PLC\_1 下的“在线和诊断”，在“诊断”下拉列表中点击“诊断缓冲区”，可以看到伺服故障事件，与 KincoServo+ 上位机软件的故障显示一致；下方的“事件详细信息”栏提供了报警原因和处理措施：



错误状态字1		错误状态字2	
屏蔽	状态	屏蔽	状态
<input checked="" type="checkbox"/>	0001 ● 扩展错误	<input checked="" type="checkbox"/>	0001 ● 电流传感器故障
<input checked="" type="checkbox"/>	0002 ● 编码器ABZ故障或通信故障	<input checked="" type="checkbox"/>	0002 ● 看门狗报错
<input checked="" type="checkbox"/>	0004 ● 编码器VW故障或内部故障	<input checked="" type="checkbox"/>	0004 ● 异常中断
<input checked="" type="checkbox"/>	0008 ● 编码器计数错误或CRC错误	<input checked="" type="checkbox"/>	0008 ● MCU故障
<input checked="" type="checkbox"/>	0010 ● 驱动器温度过高	<input checked="" type="checkbox"/>	0010 ● 电机配置错误
<input checked="" type="checkbox"/>	0020 ● 驱动器总线电压过高	<input checked="" type="checkbox"/>	0020 ● 逻辑输出故障
<input checked="" type="checkbox"/>	0040 ● 驱动器总线电压过低	<input checked="" type="checkbox"/>	0040 ● ST01
<input checked="" type="checkbox"/>	0080 ● 驱动器输出短路	<input checked="" type="checkbox"/>	0080 ● ST02
<input checked="" type="checkbox"/>	0100 ● 驱动器制动电阻异常	<input checked="" type="checkbox"/>	0100 ● 预使能报警
<input checked="" type="checkbox"/>	0200 ● 实际跟踪误差超过允许	<input checked="" type="checkbox"/>	0200 ● 正限位报错
<input checked="" type="checkbox"/>	0400 ● 逻辑电压	<input checked="" type="checkbox"/>	0400 ● 负限位报错
<input checked="" type="checkbox"/>	0800 ● 电机或驱动器II+故障	<input checked="" type="checkbox"/>	0800 ● SPI故障
<input checked="" type="checkbox"/>	1000 ● 输入脉冲频率过高	<input checked="" type="checkbox"/>	1000 ● CAN总线故障
<input checked="" type="checkbox"/>	2000 ● 电机温度过高	<input checked="" type="checkbox"/>	2000 ● 全闭环故障
<input checked="" type="checkbox"/>	4000 ● 电机励磁或编码器信息错误	<input checked="" type="checkbox"/>	4000 ● 主编码器ABZ故障
<input checked="" type="checkbox"/>	8000 ● EEPROM数据错误	<input checked="" type="checkbox"/>	8000 ● 主编码器计数错误

当驱动器故障被消除和确认后，“诊断缓冲区”会显示伺服故障事件已清除：

诊断缓冲区

事件

以PG/PC本地时间显示CPU事件时间戳

编号	事件	清除	确认
1	扩展错误	✓	→
2	电机励磁或编码器信息错误	✓	→
3	编码器ABZ/UUV故障：编码器通信内部故障	✓	→
4	诊断信息可用，并且正在处理	✓	→
5	CAN总线故障	✓	→
6	诊断信息可用，并且正在处理	✓	→
7	I/O 设备故障（指示未决的故障）	✓	→
8	伙伴错误 - 检测到相邻方	✓	→

冻结显示

事件详细信息：

事件详细信息： 5 / 572 事件 ID： 16# 74CA-B316

模块： 板5 / PROFIdrive Module\_1 Standard Telegram 3 ,P2D-5/9

机架插槽： 机架 0 / 插槽 1.2

说明： 错误-CAN总线故障  
FDS / PROFIdrive Module\_1 Standard Telegram 3 ,P2D-5/9

故障出现

关于事件的帮助信息： 报警原因：出现通讯异常，通讯中断模式(0x600700)设为1时才会开启。报警时间前值由节点保护时间(0x100C00)单位ms和节点保护时间系数(0x100D00)的乘积确定

工厂标识： -- 位置标识： --

到达而去： 到达事件 事件类型： 错误



诊断缓冲区

事件

以PG/PC本地时间显示CPU事件时间戳

编号	事件	清除	确认
1	扩展错误	✓	→
2	CAN总线故障	✓	→
3	电机励磁或编码器信息错误	✓	→
4	编码器ABZ/UUV故障：编码器通信内部故障	✓	→
5	扩展错误	✓	→
6	电机励磁或编码器信息错误	✓	→
7	编码器ABZ/UUV故障：编码器通信内部故障	✓	→
8	诊断信息可用，并且正在处理	✓	→

冻结显示

事件详细信息：

事件详细信息： 2 / 576 事件 ID： 16# 74CA-B316

模块： 板5 / PROFIdrive Module\_1 Standard Telegram 3 ,P2D-5/9

机架插槽： 机架 0 / 插槽 1.2

说明： 错误-CAN总线故障  
FDS / PROFIdrive Module\_1 Standard Telegram 3 ,P2D-5/9

故障消除

关于事件的帮助信息： 离去事件：无需用户操作

工厂标识： -- 位置标识： --

到达而去： 离去事件 事件类型： 确定

## 第四章 应用类 1

### 4.1 概述

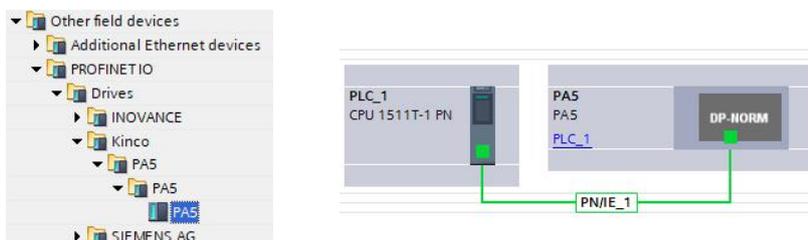
Kinco PN 伺服在应用类 1 可用报文 1。PLC 可通过 FB285 控制驱动器，实现速度控制。不支持等时同步 (IRT) 操作。

### 4.2 组态配置

1. 驱动器 IP 地址和设备名称的修改见 3.1 节。
2. 点击上方菜单“选项 → 管理通用站描述文件”安装 GSD，请以 GSDML-V2.43-Kinco-PA5-20240328 为准。另外，PN 固件 00000004 适用的 GSD 为 GSDML-V2.33-Kinco-MD60-20210507。



3. 在网络视图下，点击“硬件目录 → 其它现场设备 → PROFINET IO → Drives → Kinco → PA5”，添加伺服并为其分配 PLC：

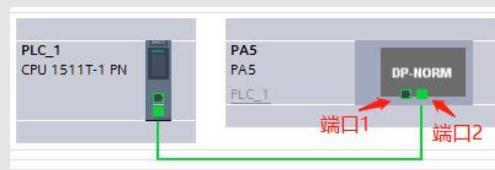
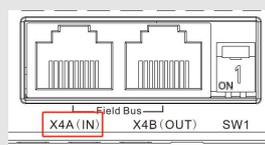


4. 如果需要 PLC 自动配置驱动器设备名称（自动分配的前提是驱动器内部的名称为空），则必须连接拓扑图，否则可以不连接。在拓扑视图下，根据实际情况连接端口。

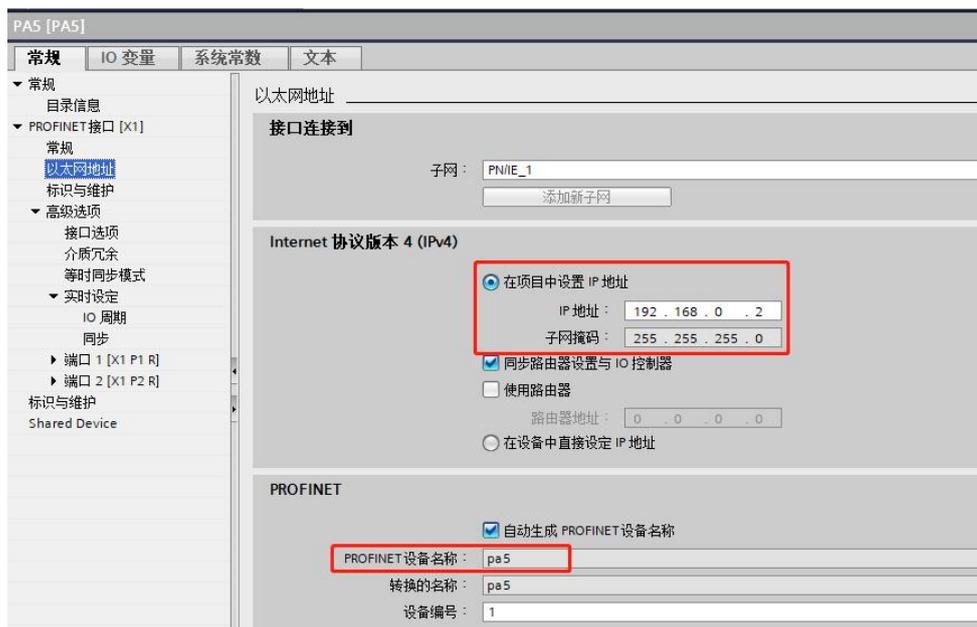


#### 注意

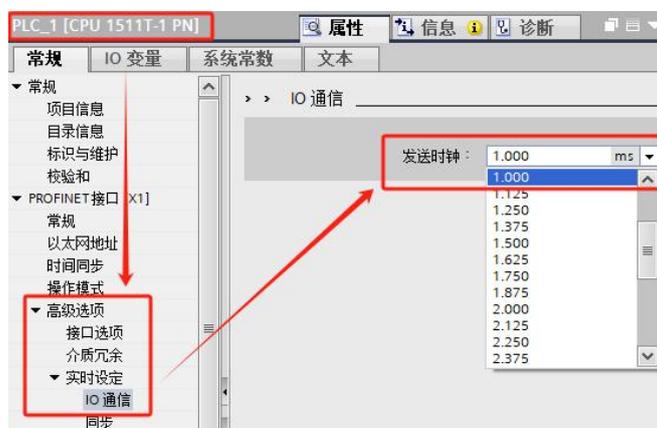
- 不同系列的驱动器上的 IN 口对应的拓扑图上的端口有所区别，FD5P 系列驱动器和 MD 系列一体机的 IN 口对应拓扑图中的端口 1，FD5 系列驱动器的 IN 口对应拓扑图中的端口 2



5. 双击 PA5 模块，在“常规”中点击“以太网地址”，设置组态的 IP 地址和 PROFINET 设备名称。组态的 IP 和设备名需要与驱动器的实际 IP 和设备名一致；当有多台驱动器时，设备名称不能相同：



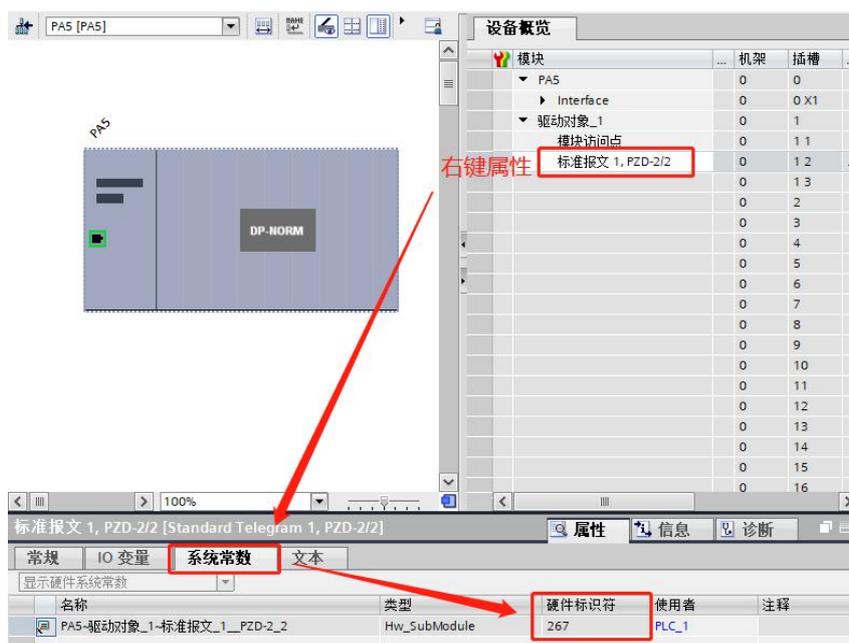
6. 在设备视图下，选择 PLC\_1，点击“常规→PROFINET 接口→高级选项→实时设定→IO 通信”，设置 PLC 的发送时钟：



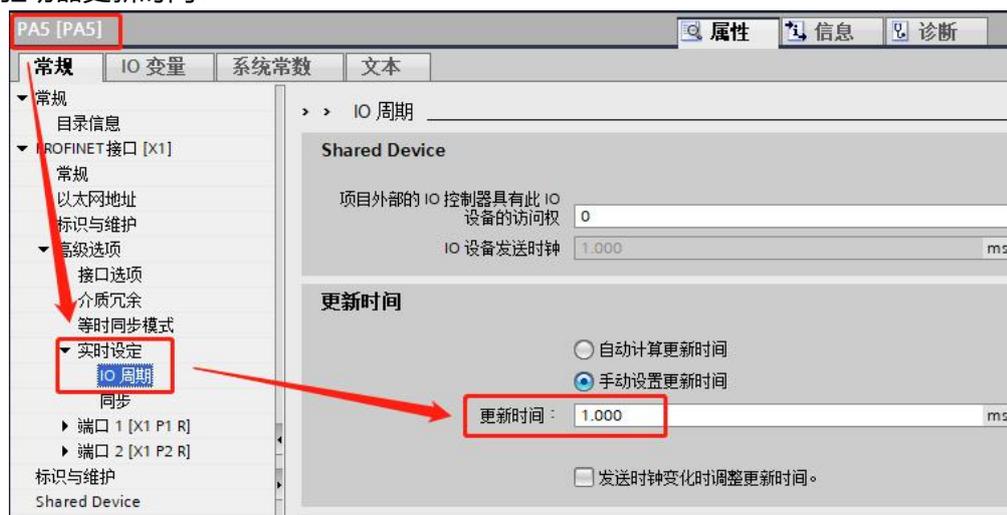
7. 在设备视图下，选择 PA5，点击“硬件目录→模块→PROFIdrive Module→子模块”，先删除“设备概览”下的报文 3，再把报文 1 添加到对应位置：



8. 记下硬件标识符，选择报文 1→右键属性→系统常数→硬件标识符：



9. 在设备视图下，选择 PA5，点击“常规→PROFINET 接口→高级选项→实时设定→IO 周期”，设置组态的驱动器更新时间：

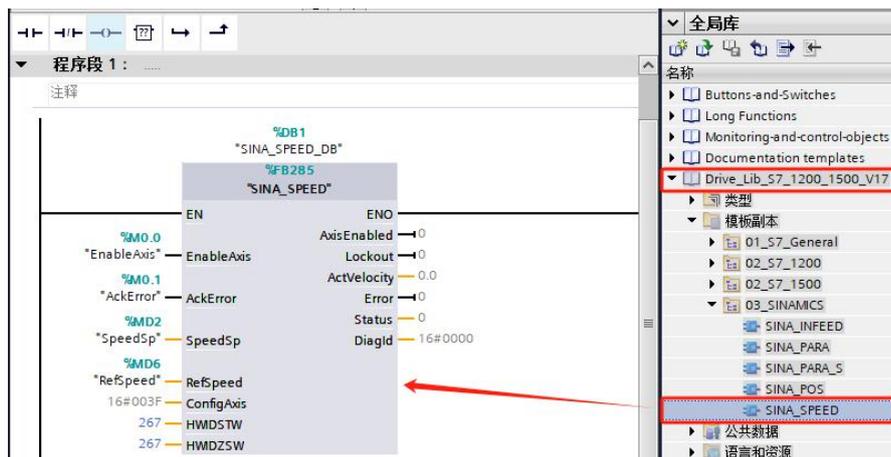


10. 将 FB285 拖拽到 Main[OB1]，并为其引脚分配好变量。

如果找不到相应的功能块 (FB)，可到西门子官网下载库文件 Drive\_Lib\_S7\_1200\_1500\_V17，网址：

<https://support.industry.siemens.com/cs/document/109475044/sinamics-communication-blocks-drivelib-for-reading-and-writing-drive-data-within-tia-portal-context?dti=0&lc=en-WW>

下载完成后，点击软件上方的菜单“选项→全局库→打开库”，按照提示安装对应的库文件。



### 4.3 SINA\_SPEED (FB285) 介绍

FB285 引脚定义见表 4-1。

表 4-1 FB285 引脚定义

引脚名称	数据类型	说明																		
EnableAxis	Bool	使能轴： =0 松轴，伺服控制字为 0xE =1 使能，伺服控制字为 0xF																		
AckError	Bool	复位故障，上升沿有效																		
SpeedSp	Real	速度设定值，单位 RPM																		
RefSpeed	Real	参考速度，必须为最大速度[607F00]的 1/2，单位 RPM																		
ConfigAxis	Word	默认 16#3F，与 STW1 的对应关系如下： <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>ConfigAxis 中的比特</th> <th>STW1 中的比特</th> </tr> </thead> <tbody> <tr><td>比特 0</td><td>比特 1</td></tr> <tr><td>比特 1</td><td>比特 2</td></tr> <tr><td>比特 2</td><td>比特 3</td></tr> <tr><td>比特 3</td><td>比特 4</td></tr> <tr><td>比特 4</td><td>比特 5</td></tr> <tr><td>比特 5</td><td>比特 6</td></tr> <tr><td>比特 6</td><td>比特 11</td></tr> <tr><td>比特 7~15: 保留</td><td></td></tr> </tbody> </table> STW1 中各比特赋值意义见表 1-6	ConfigAxis 中的比特	STW1 中的比特	比特 0	比特 1	比特 1	比特 2	比特 2	比特 3	比特 3	比特 4	比特 4	比特 5	比特 5	比特 6	比特 6	比特 11	比特 7~15: 保留	
ConfigAxis 中的比特	STW1 中的比特																			
比特 0	比特 1																			
比特 1	比特 2																			
比特 2	比特 3																			
比特 3	比特 4																			
比特 4	比特 5																			
比特 5	比特 6																			
比特 6	比特 11																			
比特 7~15: 保留																				
HWIDSTW	HW_IO	见 4.2 节第 7 步，通过该参数来区分不同的轴																		
HWIDZSW	HW_IO																			
AxisEnabled	Bool	=1 轴已使能																		
Lockout	Bool	=1 禁止接通																		
ActVelocity	Real	速度实际值，单位 RPM																		
Error	Bool	=1 存在故障																		
Status	Int	状态指示： 16#7002: 没有错误 16#8401: 伺服故障 16#8402: 禁止接通 16#8600: DPRD DAT 错误 16#8601: DPWR DAT 错误																		
DiagId	Word	扩展通讯错误																		

## 第五章 应用类 3

### 5.1 概述

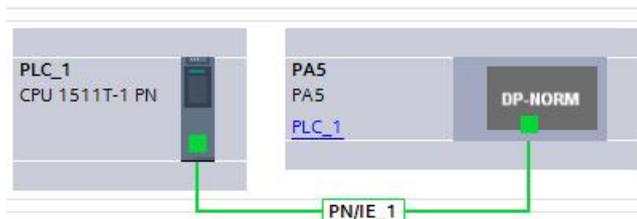
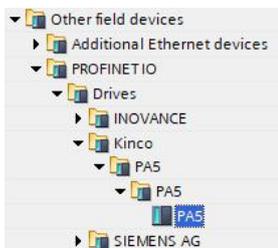
Kinco PN 伺服在应用类 3 可选择报文 9、111。使用报文 111 时，PLC 可通过 FB284 控制驱动器，实现基本定位(EPOS)功能；运行模式有 Jog、Homing、MDI、位置表等。不支持等时同步 (IRT) 操作。

### 5.2 组态配置

1. 驱动器 IP 地址和设备名称的修改见 3.1 节。
2. 点击上方菜单“选项 → 管理通用站描述文件”安装 GSD，请以 GSDML-V2.43-Kinco-PA5-20240328 为准。另外，PN 固件 00000004 适用的 GSD 为 GSDML-V2.33-Kinco-MD60-20210507。



3. 在网络视图下，点击“硬件目录 → 其它现场设备 → PROFINET IO → Drives → Kinco → PA5”，添加伺服并为其分配 PLC：

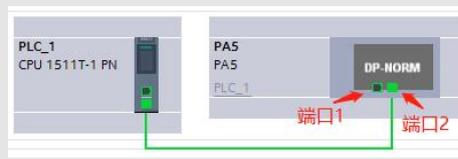
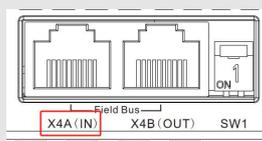


4. 如果需要 PLC 自动配置驱动器设备名称（自动分配的前提是驱动器内部的名称为空），则必须连接拓扑图，否则可以不连接。在拓扑视图下，根据实际情况连接端口。

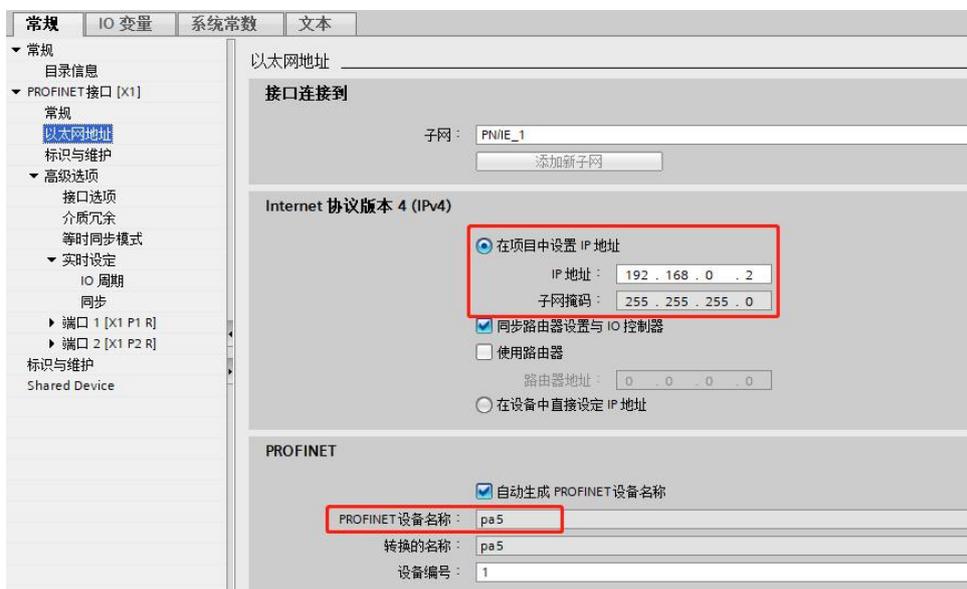


#### 注意

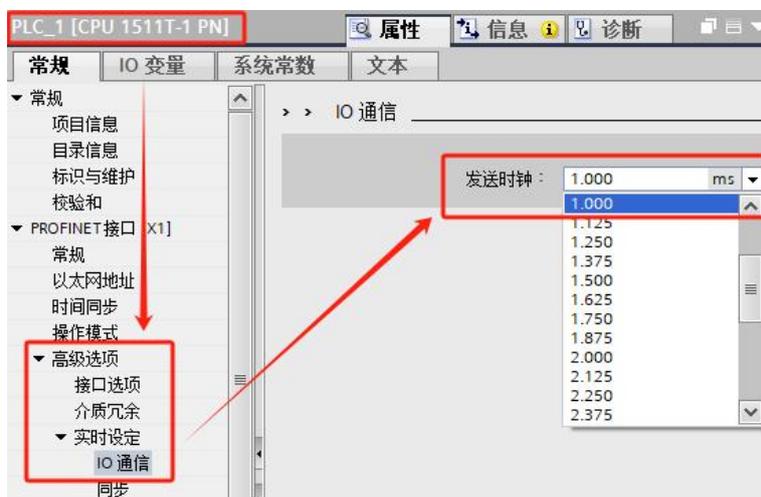
- 不同系列的驱动器上的 IN 口对应的拓扑图上的端口有所区别，FD5P 系列驱动器和 MD 系列一体机的 IN 口对应拓扑图中的端口 1，FD5 系列驱动器的 IN 口对应拓扑图中的端口 2



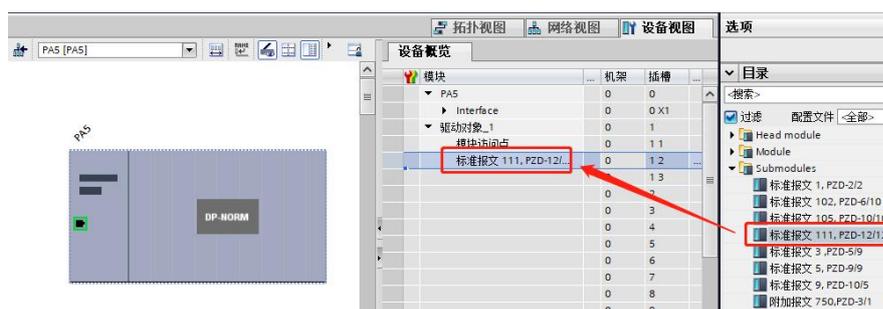
5. 双击 PA5 模块，在“常规”中点击“以太网地址”，设置组态的 IP 地址和 PROFINET 设备名称。组态的 IP 和设备名需要与驱动器的实际 IP 和设备名一致；当有多台驱动器时，设备名称不能相同：



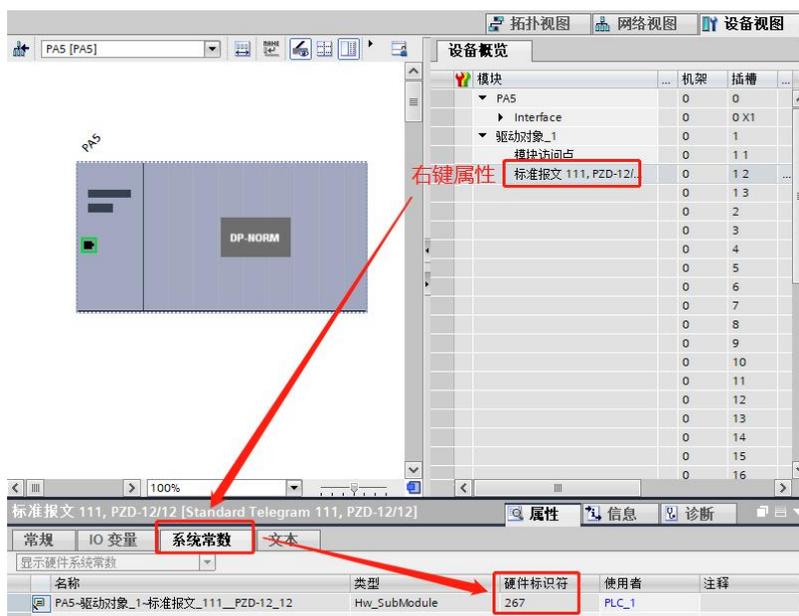
6. 在设备视图下，选择 PLC\_1，点击“常规→PROFINET 接口→高级选项→实时设定→IO 通信”，设置 PLC 的发送时钟：



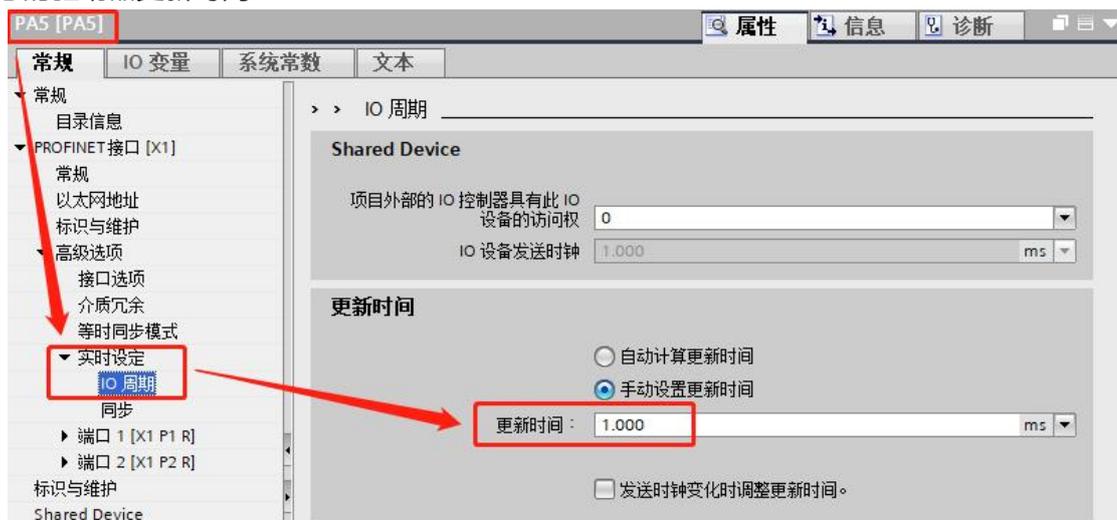
7. 在设备视图下，选择 PA5，点击“硬件目录→模块→PROFIdrive Module→子模块”，先删除“设备概览”下的报文 3，再把报文 111 添加到对应位置：



8. 记下硬件标识符，选择报文 111→右键属性→系统常数→硬件标识符：



9. 在设备视图下，选择 PA5，点击“常规→PROFINET 接口→高级选项→实时设定→IO 周期”，设置组态的驱动器更新时间：

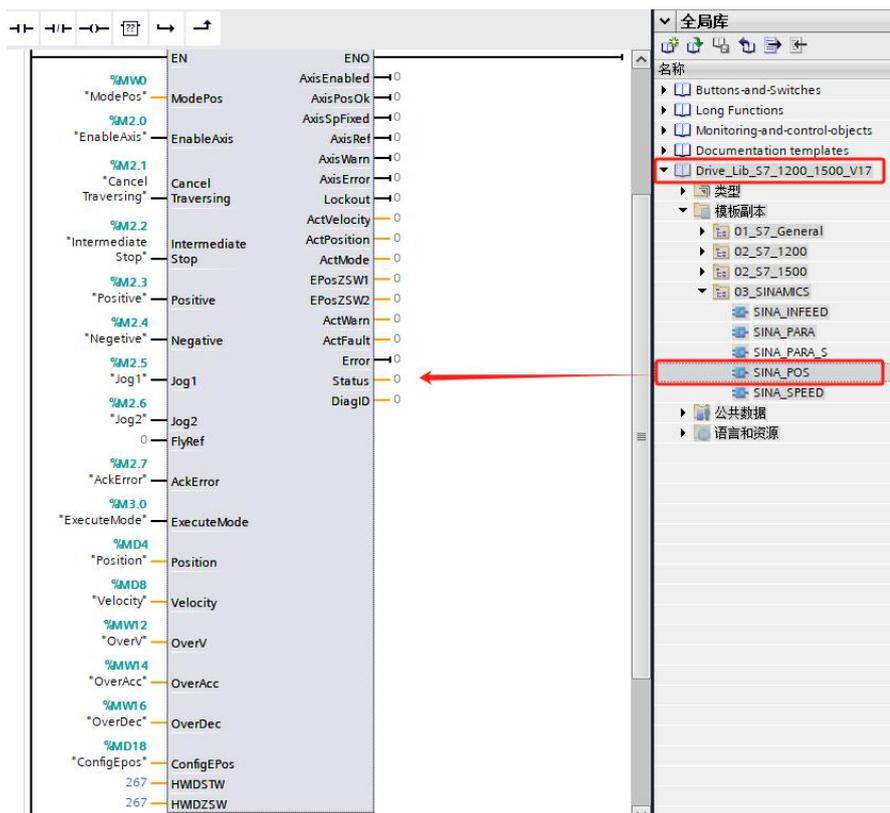


10. 将 FB284 拖拽到 Main[OB1]，并为其引脚分配好变量。

如果找不到相应的功能块 (FB)，可到西门子官网下载库文件 Drive\_Lib\_S7\_1200\_1500\_V17，网址：

<https://support.industry.siemens.com/cs/document/109475044/sinamics-communication-blocks-drivelib-for-re-ading-and-writing-drive-data-within-tia-portal-context?dti=0&lc=en-WW>

下载完成后，点击软件上方的菜单“选项→全局库→打开库”，按照提示安装对应的库文件。



### 5.3 SINA\_POS (FB284) 介绍

FB284 引脚定义见表 5-1。

表 5-1 FB284 引脚定义

引脚名称	数据类型	说明
ModePos	Int	运行模式： =0 — =1 相对定位模式 =2 绝对定位模式 =3 — =4 主动回原点模式 =5 直接设置原点模式 =6 位置表模式(未支持) =7 按指定速度点动模式 =8 —
EnableAxis	Bool	=0 松轴，伺服控制字为 0xE =1 使能，伺服控制字为 0xF
CancelTraversing	Bool	=1 接受定位任务 =0 拒绝定位任务
IntermediateStop	Bool	=1 不暂停定位任务 =0 暂停定位任务
Positive	Bool	正方向
Negative	Bool	负方向
Jog1	Bool	反向点动
Jog2	Bool	正向点动
FlyRef	Bool	无效，需设为 0

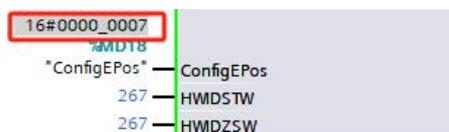
引脚名称	数据类型	说明																
AckError	Bool	复位故障, 上升沿有效																
ExecuteMode	Bool	激活运行模式, 上升沿有效																
Position	DInt	运行模式为 1、2 时的位置设定值 运行模式为 6 时的位置表起始/新任务索引, 范围 0~7																
Velocity	DInt	运行模式为 1、2、7 时的速度设定值																
ConfigEPos	DWord	默认 16#00000003, 与相关控制字的对应关系如下: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>ConfigEPos 中的比特</th> <th>相关控制字中的比特</th> </tr> </thead> <tbody> <tr> <td>比特 0</td> <td>STW1 比特 1</td> </tr> <tr> <td>比特 1</td> <td>STW1 比特 2</td> </tr> <tr> <td>比特 2</td> <td>POS_STW2 比特 14</td> </tr> <tr> <td>比特 3</td> <td>POS_STW2 比特 15</td> </tr> <tr> <td>比特 6</td> <td>POS_STW2 比特 2</td> </tr> <tr> <td>比特 8</td> <td>POS_STW1 比特 12</td> </tr> <tr> <td>其它比特: 保留</td> <td></td> </tr> </tbody> </table> <p>其中, STW1 中各比特赋值意义见表 1-6                      POS_STW1 中各比特赋值意义见表 1-13                      POS_STW2 中各比特赋值意义见表 1-14</p>	ConfigEPos 中的比特	相关控制字中的比特	比特 0	STW1 比特 1	比特 1	STW1 比特 2	比特 2	POS_STW2 比特 14	比特 3	POS_STW2 比特 15	比特 6	POS_STW2 比特 2	比特 8	POS_STW1 比特 12	其它比特: 保留	
ConfigEPos 中的比特	相关控制字中的比特																	
比特 0	STW1 比特 1																	
比特 1	STW1 比特 2																	
比特 2	POS_STW2 比特 14																	
比特 3	POS_STW2 比特 15																	
比特 6	POS_STW2 比特 2																	
比特 8	POS_STW1 比特 12																	
其它比特: 保留																		
OverV	Int	运行模式为 1、2、7 时的速度设定值百分比, 单位%, 范围 0~199%																
OverAcc	Int	运行模式为 1、2、7 时的加速度百分比, 单位%, 范围 0~100%																
OverDec	Int	运行模式为 1、2、7 时的减速度百分比, 单位%, 范围 0~100%																
HWIDSTW	HW_IO	见 5.2 节第 7 步, 通过该参数来区分不同的轴																
HWIDZSW	HW_IO																	
AxisEnabled	Bool	=1 轴已使能																
AxisPosOk	Bool	=1 达到目标位置																
AxisSpFixed	Bool	=1 达到设定位置																
AxisRef	Bool	=1 原点已设定																
AxisWarn	Bool	=1 存在报警																
AxisError	Bool	=1 存在故障																
Lockout	Bool	=1 禁止接通																
ActVelocity	DInt	速度实际值																
ActPosition	DInt	位置实际值																
ActMode	Int	当前的运行模式																
EPosZSW1	Word	POS_ZSW1 当前值, 见表 1-15																
EPosZSW2	Word	POS_ZSW2 当前值, 见表 1-16																
ActWarn	Word	伺服报警代码																
ActFault	Word	驱动器错误代码[603F00]																
Error	Bool	=1 存在故障																
Status	Word	状态指示: 16#7002: 没有错误 16#8401: 伺服故障 16#8402: 禁止接通 16#8403: 运行中禁止回原点 16#8600: DPRD DAT 错误 16#8601: DPWR DAT 错误 16#8202: 运行模式选择错误 16#8203: 设定值错误 16#8204: 位置表任务索引错误																
DiagID	Word	扩展通讯错误																

## 5.4 FB284 功能说明

### 5.4.1 运行条件和设置

1. 输入引脚 “ConfigEPos” 初始值应为 3，即 “ConfigEPos” 比特 0 和比特 1 为 1。
2. 输入引脚 “CancelTraversing” 和 “IntermediateStop” 对于除点动模式之外的所有模式均有效，在运行时应设为 1，说明如下：
  - a. “CancelTraversing” = 0 时，表示取消当前任务，停止减速度对应暂停模式[605D00]。轴停止后可以切换运行模式。
  - b. “IntermediateStop” = 0 时，表示暂停当前任务，停止减速度对应暂停模式[605D00]。重新设置 “IntermediateStop” = 1 后轴会继续运行。轴停止后可以切换运行模式。
3. 输入引脚 “ModePos” 用于选择运行模式；“ExecuteMode” 的上升沿触发定位运动。
4. 激活软限位开关

设定原点之后，如果需要使用软件限位开关，需要将输入引脚 “ConfigEPos” 的比特 2 置 1 (“ConfigEPos” = 16#00000007) 或者将限位使能[30800C]置 1:



激活软件限位功能，在驱动中设置软限位正设置[607D01]，软限位负设置[607D02]。

30800C	uint8	软件限位使能	1	DEC
607D01	int32	软限位正设置	10000000	DEC
607D02	int32	软限位负设置	-10000000	DEC

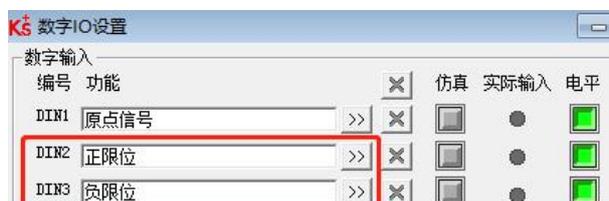
设置 “软限位正设置” > “软限位负设置”，并且找到原点之后，软件限位功能开始启用。

### 5. 激活硬限位开关

如果需要使用硬件限位开关，需要将输入引脚 “ConfigEPos” 的比特 3 置 1 (“ConfigEPos” = 16#0000000B):



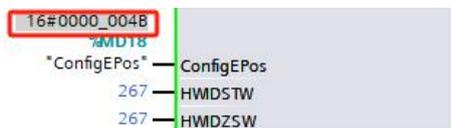
在上位机软件为驱动器的 DIN 定义正限位和负限位。只有在硬件限位开关信号为高电平时才能运行轴。



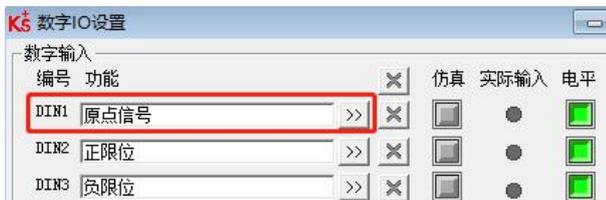
请注意，只是在上位机软件中配置正负限位，而“ConfigEPos”的比特 3 没有置 1，则硬件限位功能不生效。

### 6. 激活原点开关

通过将输入引脚“ConfigEPos”的比特 6 置 1 (“ConfigEPos” = 16#0000004B)，可以激活原点开关信号：

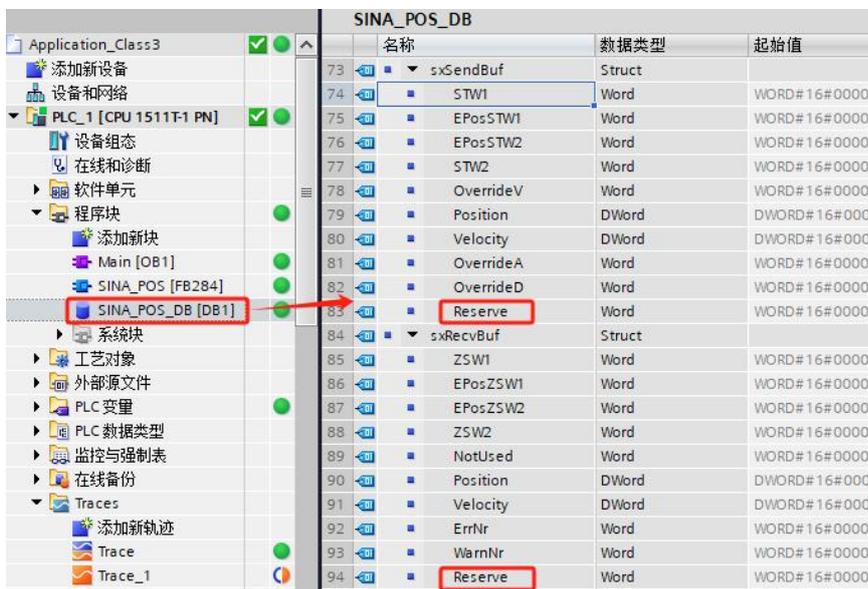


在上位机软件为驱动器的 DIN 配置原点信号，也能达到相同目的：



### 7. 使用报文 111 设置和读取伺服参数

PLC 为报文 111 分配了各 24 字节的接收区和发送区，可在 FB284 的 DB 块中查看，结构体“sxSendBuf”和“sxRecvBuf”下的“Reserve”变量可用于自定义设置和读取伺服参数：



相关的伺服参数为 PN 用户自定义接收字[30800D]和 PN 用户自定义发送字[30800E]：

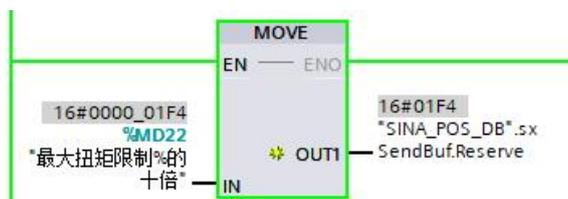
30800D	uint8	PN用户自定义接收字	0	DEC
30800E	uint8	PN用户自定义发送字	0	DEC

解释如下：

PN 用户自定义接收字	=0 无功能 =1 开启转矩限制，在“sxSendBuf”下的“Reserve”中传送转矩限制值
PN 用户自定义发送字	=0 无功能 =2 在“sxRecvBuf”下的“Reserve”中传送实际电流

a. 转矩限制功能

使用转矩限制时，应设置适当的转矩限制值，若限制值过小，可能会导致驱动器跟随误差过大。FB284 DB 块中“sxSendBuf”的“Reserve”变量对应于最大转矩限制%[607200]的十倍。首先将“Reserve”设置为 16#1F4 = 500，则最大转矩限制%应为 50%：



然后将 PN 用户自定义接收字[30800D]设定为 1：附加转矩，如下：

索引	子索引	名称	数据类型	属性
3080	00	PN用户自定义接收字	Unsigned8	RWL
Value		Unit		
1		DEC		

帮助:PN用户自定义接收字  
PN用户自定义接收字  
0: 无功能  
1: 附加转矩

可见最大转矩限制%已变成 50%：

索引	子索引	名称	数据类型	属性
6072	00	最大转矩限制%	Unsigned16	RWL
Value		Unit		
49.80		%		

帮助:最大转矩限制%  
最大转矩占额定力矩的百分比

请注意，一旦将 [30800D]设定为 1，转矩限制就会生效，再将[30800D]设为 0 后，转矩限制值并不会恢复到初始值，但可以在上位机软件直接设定转矩限制[607200]。

b. 读取实际电流

首先将 PN 用户自定义发送字[30800E]设定为 2：实际电流值，如下：

索引	子索引	名称	数据类型	属性
3080	0E	PN用户自定义发送字	Unsigned8	RWL
Value		Unit		
2		DEC		
帮助:PN用户自定义发送字				
PN用户自定义发送字				
0: 无功能				
1: 实际转矩				
2: 实际电流值				

FB284 DB 块中 “sxRecvBuf” 的 “Reserve” 变量读取到伺服实际电流，单位 DEC，需进行换算，如把读取到的值换算成 Ap:  $16\#4F = 79$  ,  $79 / 2048 * 18Ap$  (最大电流[651003]) = 0.7Ap :



索引	子索引	名称	数据类型	属性
6078	00	实际电流	Integer16	RLTM
Value		Unit		
0.70		Ap		
帮助:实际电流				
实际电流				

### 5.4.2 运行模式 1 (相对定位)

要求:

1. 运行模式 “ModePos” = 1
2. “ConfigEPos” = 16#00000003
3. 使能轴 “EnableAxis” = 1
4. “CancelTraversing” 和 “IntermediateStop” 必须为 1, “Jog1” 和 “Jog2” 必须为 0。

步骤:

1. 通过输入引脚 “Position” , “Velocity” 指定目标位置和目标速度，位置和速度单位均为 DEC，参考伺服手册换算，“Velocity” 的单位转化为:  $DEC = RPM * 512 * 反馈精度[641003] / 1875$ 。
2. 通过输入引脚 “OverV” 、 “OverAcc” 、 “OverDec” 对目标速度、梯形加减速度进行百分比缩放。
3. 运动方向由 “Position” 中设置值的正负来确定。
4. 通过 “ExecuteMode” 的上升沿触发运动，到达目标位置后 “AxisPosOk” 置 1。若定位时出现错误，则输出引脚 “Error” 置 1。
5. 当前运行的命令可通过 “ExecuteMode” 上升沿进行新命令替换。在任意时刻，可在运行模式 1、2 间切换。如果需要切换到其它运行模式，轴必须为静止状态。

### 5.4.3 运行模式 2 (绝对定位)

要求:

1. 运行模式 “ModePos” = 2
2. “ConfigEPos” = 16#00000003
3. 使能轴 “EnableAxis” = 1
4. “CancelTraversing” 和 “IntermediateStop” 必须为 1, “Jog1” 和 “Jog2” 必须为 0。

步骤:

1. 通过输入引脚 “Position”, “Velocity” 指定目标位置和目标速度, 位置和速度单位均为 DEC, 参考伺服手册换算, “Velocity” 的单位转化为:  $DEC = RPM * 512 * \text{反馈精度}[641003] / 1875$ 。
2. 通过输入引脚 “OverV”、“OverAcc”、“OverDec” 对目标速度、梯形加减速度进行百分比缩放。
3. 运行时伺服按照最短路径运行至目标位置, 输入引脚 “Positive” 及 “Negative” 必须为 0。
4. 通过 “ExecuteMode” 的上升沿触发运动, 当到达目标位置后 “AxisPosOk” 置 1。若定位过程中出现错误, 则输出引脚 “Error” 置 1。
5. 当前运行的命令可通过 “ExecuteMode” 上升沿进行新命令替换。在任意时刻, 可在运行模式 1、2 间切换。如果需要切换到其它运行模式, 轴必须为静止状态。
6. 通过置位 “ConfigEPos” 的比特 8 (“ConfigEPos” = 16#00000103), 在 PLC 侧更新 “Position” 后, 无需触发 “ExecuteMode”, 新的设定值会立即生效, 伺服会根据目标位置变化立即执行绝对定位指令。

### 5.4.4 运行模式 4 (主动回原点)

要求:

1. 运行模式 “ModePos” = 4
2. “ConfigEPos” = 16#00000003。若使用硬件限位, 则 “ConfigEPos” = 16#0000000B。
3. 使能轴 “EnableAxis” = 1
4. “CancelTraversing” 和 “IntermediateStop” 必须为 1, “Jog1” 和 “Jog2” 必须为 0。

步骤:

1. 回原点的方式以及回原点速度, 需用上位机软件进行配置 (PLC 只发送回原点命令, 回原点的规划由伺服参数设定), 具体回原点方式请参考伺服手册:

N	Index	Type	Name	Value	Unit
0	607C00	int32	原点偏移	0	inc
1	609800	int8	原点模式	1	DEC
2	609901	uint32	原点转折信号速度	300.00	rpm
3	609902	uint32	原点信号速度	100.00	rpm
4	609903	uint8	上电找原点	0	DEC
5	609A00	uint32	原点加速度	50.00	rps/s
6	609904	int16	寻找原点最大电流	5.94	Ap
7	609905	uint8	原点偏移模式	0	DEC
8	609906	uint8	原点索引信号盲区	0	DEC

2. 通过“ExecuteMode”的上升沿触发回原点运动，回原点完成后“AxisRef”置1，若运行过程中出现错误，则输出引脚“Error”置1。

#### 5.4.5 运行模式 5 (直接回原点)

要求:

1. 运行模式“ModePos” = 5
2. “ConfigEPos” = 16#00000003
3. 使能轴“EnableAxis” = 1
4. 轴必须为静止状态。

步骤:

通过“ExecuteMode”的上升沿设置轴的原点，回原点完成后“AxisRef”置1。

#### 5.4.6 运行模式 6 (位置表) (未支持)

在位置表模式下，FD 系列驱动器可以保存和运行最多 32 个不同的定位任务。MD 系列一体机不支持该模式。

要求:

1. 如果位置表中含有绝对定位任务，则需要设定原点。
2. 运行模式“ModePos” = 6
3. “ConfigEPos” = 16#00000003
4. 使能轴“EnableAxis” = 1
5. “CancelTraversing”和“IntermediateStop”必须为1，“Jog1”和“Jog2”必须为0。

步骤:

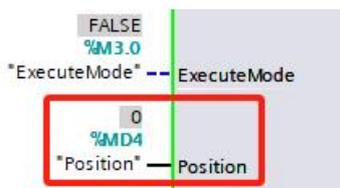
1. 参考伺服手册，在上位机软件配置位置表参数。
2. 通过输入引脚“Position”指定起始/新任务索引，取值范围0~7。
3. 运行方向由位置表中的设置决定，与“Positive”以及“Negative”无关。
4. 指定起始任务索引后，通过“ExecuteMode”的上升沿触发运行。
5. 在运行过程中，当前的定位任务可以被一个新任务通过“ExecuteMode”的上升沿触发进行替代。

示例:

1. 在上位机软件点击“驱动器→控制模式→位置表模式”，进入位置表设置界面，其中绝对定位任务“模式”选择“A”，相对定位任务“模式”选择“RA”，“触发”均选择“中断”，其它参数可参考伺服手册设置。下图设置了5个任务，任务0、1、3为绝对定位，任务2、4为相对定位；任务0~3完成后，会继续执行下一个任务，任务4完成后停止：

Ks 位置表模式													
控制寄存器:3													
位0-4:下一个				位5	位6	位7	位8:继续/停止	位9:条件0	位10:条件1	位11:与/或	位12-13:模式	位14-15:触发	
4				0	0	0	1	0	0	0	0	2	
Idx	模式	触发	位置inc	速度rpm	延时ms	Acc索引	Dec索引	控制寄存器	循环	剩余	加速度rps/s	减速度rps/s	
0	A	中断	1000000	100.00	0	0	0	8101	0	0	0	20.00	20.00
1	A	中断	2000000	200.00	0	1	1	8102	0	0	1	50.00	50.00
2	RA	中断	-1500000	300.00	0	1	1	A103	0	0	2	0.00	0.00
3	A	中断	1000000	200.00	0	1	1	8104	0	0	3	0.00	0.00
4	RA	中断	500000	100.00	0	0	0	A000	0	0	4	0.00	0.00

2. FB284 运行模式选择 6，输入引脚 “Position” 设为 0，触发 “ExecuteMode”，则伺服开始运行位置表，并从任务 0 开始：



3. 位置表中的任务运行完毕后，“AxisPosOk” 置 1。  
 请注意，在当前任务运行过程中，可以在 “Position” 指定新的任务索引，并触发 “ExecuteMode” 进行任务替换，驱动器可支持 32 个不同的任务，但是可指定的新的任务索引范围为 0~7。

### 5.4.7 运行模式 7 (按指定速度点动)

要求：

1. 运行模式 “ModePos” = 7
2. “ConfigEPos” = 16#00000003
3. 使能轴 “EnableAxis” = 1
4. 轴必须为静止状态。

步骤：

1. 通过输入引脚 “Velocity” 指定点动速度，必须为正值，单位 DEC。
2. 通过输入引脚 “OverV”、“OverAcc”、“OverDec” 对点动速度、梯形加减速度进行百分比缩放。
3. “CancelTraversing” 以及 “IntermediateStop” 与点动模式无关。运行方向与 “Positive” 以及 “Negative” 无关。
4. 无需触发 “ExecuteMode”，“Jog1” = 1 时反向点动，“Jog2” = 1 时正向点动。“Jog1” 和 “Jog2” 只能单独触发，当 “Jog1” 和 “Jog2” 同时为 0 或 1 时，轴停止。

## 第六章 应用类 4

### 6.1 概述

Kinco PN 伺服在应用类 4 可选择报文 3、5、102、105。在 PLC 中组态工艺对象，使用 MC\_Power、MC\_MoveAbsolute 等 PLCopen 功能块进行控制，其中伺服三环控制的位置环计算在 PLC 中完成，驱动器处于速度模式；在使用 DSC 功能时，伺服处于插补模式。不同报文的功能差别见表 6-1。

表 6-1 应用类 4 报文功能差别

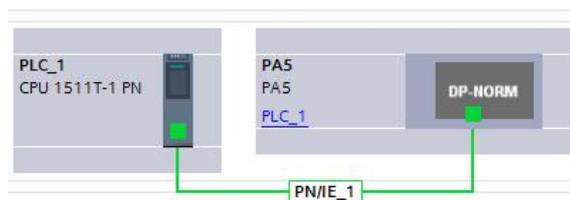
报文号	是否支持等时同步 (IRT)	是否支持转矩限制功能	是否支持 DSC 功能
105	是	是	是
102	是	是	否
5	是	否	是
3	是	否	否

### 6.2 组态配置

1. 驱动器 IP 地址和设备名称的修改见 3.1 节。
2. 点击上方菜单“选项 → 管理通用站描述文件”安装 GSD，请以 GSDML-V2.43-Kinco-PA5-20240328 为准。另外，PN 固件 00000004 适用的 GSD 为 GSDML-V2.33-Kinco-MD60-20210507。



3. 在网络视图下，点击“硬件目录 → 其它现场设备 → PROFINET IO → Drives → Kinco → PA5”，添加伺服并为其分配 PLC：

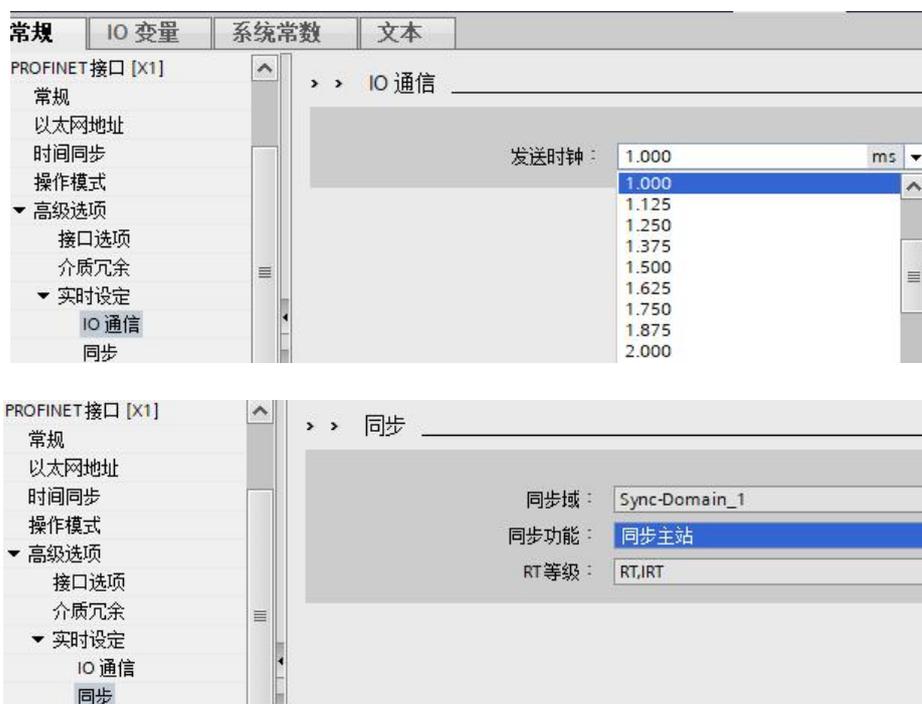


4. 使用等时同步模式必须连接拓扑图。在拓扑视图下，根据实际情况连接端口。

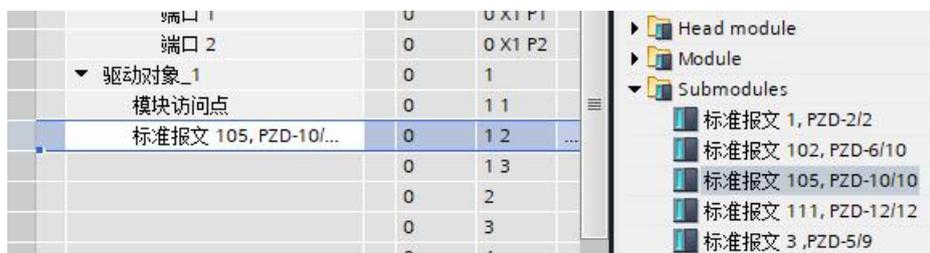
**注意**

- 不同系列的驱动器上的 IN 口对应的拓扑图上的端口有所区别，FD5P 系列驱动器和 MD 系列一体机的 IN 口对应拓扑图中的端口 1，FD5 系列驱动器的 IN 口对应拓扑图中的端口 2

5. 在设备视图下，选择 PLC\_1，点击“常规→PROFINET 接口→高级选项→实时设定”，设置发送时钟，并开启同步功能：



6. 在设备视图下，选择 PA5，点击“模块→PROFIdrive Module→子模块”，先删除“设备概览”下的报文 3，再把报文 105 添加到对应位置：



7. 点击“常规→PROFINET 接口→高级选项→等时同步模式”，勾选下方的报文 105，开启同步功能；并将 Ti 设置为 0.375ms：



8. 在 IRT 模式下，伺服的更新时间 = PLC 的发送时钟：



组态的更新时间需要与 ECAN 同步周期[301101]一致，设为 0 表示 1ms：

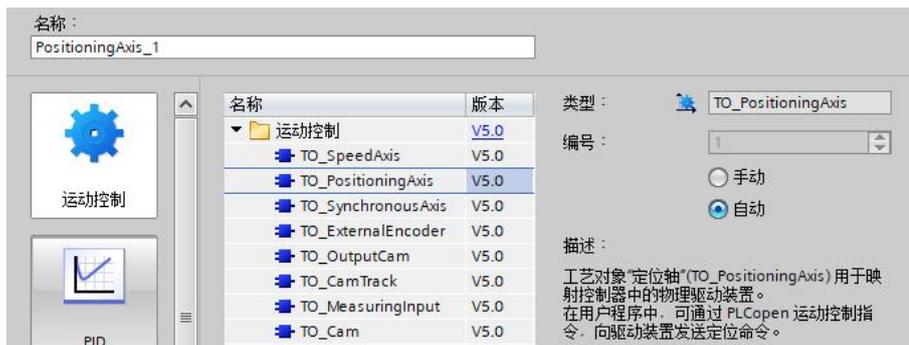
地址	位	名称	Value
3011	00	ECAN组	
3011	01	ECAN同步周期	0
3011	02	ECAN时钟同步模式	
3011	03	ECAN同步点偏移	
3011	04	ECAN同步丢失计数	
3011	05	Sync_P_Gain	
3011	06	ECAN标记位	
3011	07	ECAN同步数据	

帮助: ECAN同步周期  
 ECAN同步周期  
 0: 1ms  
 1: 2ms  
 2: 4ms  
 3: 8ms

9. 在驱动器端开启等时同步模式，将 ECAN 时钟同步模式[301102]设为 1，同时将 ECAN 同步点偏移[301103]设为 4：

地址	数据类型	名称	值	进制
9	301101	uint8	ECAN同步周期	0 DEC
10	301102	uint8	ECAN时钟同步模式	1 DEC
11	301103	uint8	ECAN同步点偏移	4 DEC

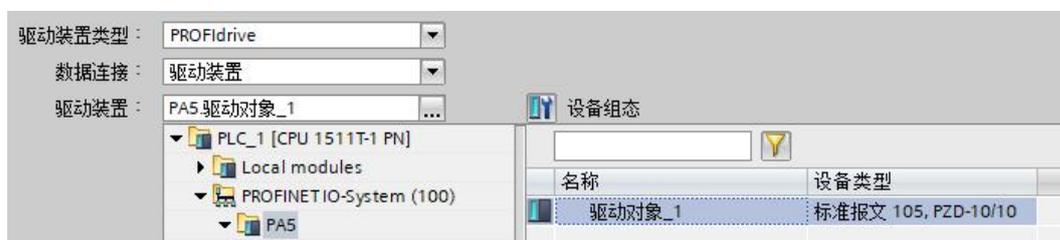
10. 设备组态完毕，点击左侧 PLC\_1 下拉列表中的“工艺对象”，新增工艺对象，选择定位轴：



11. 点击“工艺对象→PositioningAxis\_1→组态→基本参数”，选择轴类型：



12. 点击“硬件接口→驱动装置”，选择报文 105：



13. Kinco PN 伺服支持增量编码器和绝对值编码器。点击“硬件接口→编码器”，如果使用步科单圈编码器电机，就选择“增量”：



如果使用步科多圈绝对值编码器电机，就选择“循环绝对值编码器”：



14. 点击“与驱动装置进行数据交换”，设置参考转速，最大转速和基准扭矩：



其中，参考速度是最大速度[607F00]的 1/2：

索引	子索引	名称	数据类型	属性
607F	00	最大速度限制	Unsigned32	RWSLTM
		Value	Unit	
		5000.00	rpm	
帮助:最大速度限制 电机最大速度				

伺服的目标电流限制[607300]和最大扭矩限制%[607200]是关联的，基准扭矩取目标电流限制[607300]的初始默认值即可，单位 Ap，一般为驱动器峰值电流和电机峰值电流二者间的较小值：

索引	子索引	名称	数据类型	属性
6073	00	目标电流限制	Unsigned16	RWSLTM
		Value	Unit	
		17.30	Ap	
帮助:目标电流限制 电流指令最大值				

15. 点击“与编码器进行数据交换”，如果使用步科单圈编码器电机，那么每转增量为反馈精度[641003]，高精度选择 Gx\_XIST1 位 0：

索引	子索引	名称	数据类型	属性
6410	03	反馈精度	Unsigned32	RWSLEB
		Value	Unit	
		65536.00	inc/r	
帮助:反馈精度 电机编码器分辨率				

如果使用步科多圈绝对值编码器电机，那么每转增量为反馈精度[641003]，如 65536，转数也设置为 65536，高精度选择 G1\_XIST1 位 0 和 G1\_XIST2 位 0：

索引	子索引	名称	数据类型	属性
6410	03	反馈精度	Unsigned32	RWSLEB
		Value	Unit	
		65536.00	inc/r	
帮助:反馈精度 电机编码器分辨率				

16. 点击“扩展参数→机械”，设置位置参数：



17. 点击“扩展参数→位置监视→跟随误差”，可关闭跟随误差监控。在启用监控且随动误差设置较小时，使能运行后 PLC 容易报错：



18. 点击“扩展参数→位置限制→扭矩限值”，选择“电机侧”：



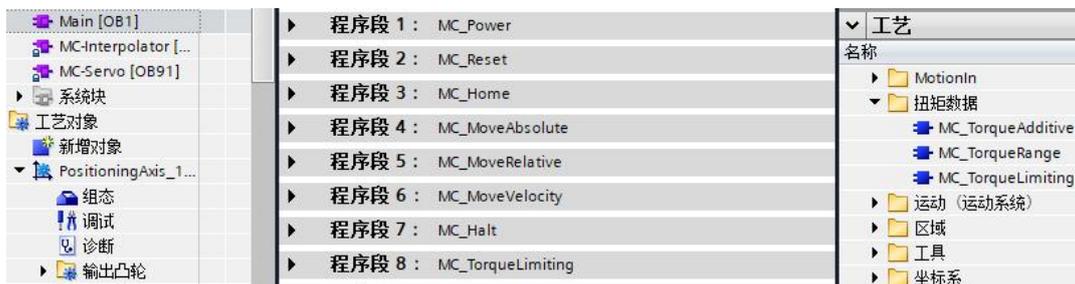
19. 点击“扩展参数→控制回路”，可启用 DSC 功能。DSC 功能的使用方法详见 6.5 节。



20. 工艺对象组态完毕，点击左侧 PLC\_1 下拉列表中的“程序块→MC-Servo[OB91]”，右键打开属性，可设置 PLC 的应用周期。请注意，使用 DSC 功能时，组态的 PLC 应用周期、伺服更新周期(见第 8 步)、ECAN 同步周期[301101]三者必须相同。

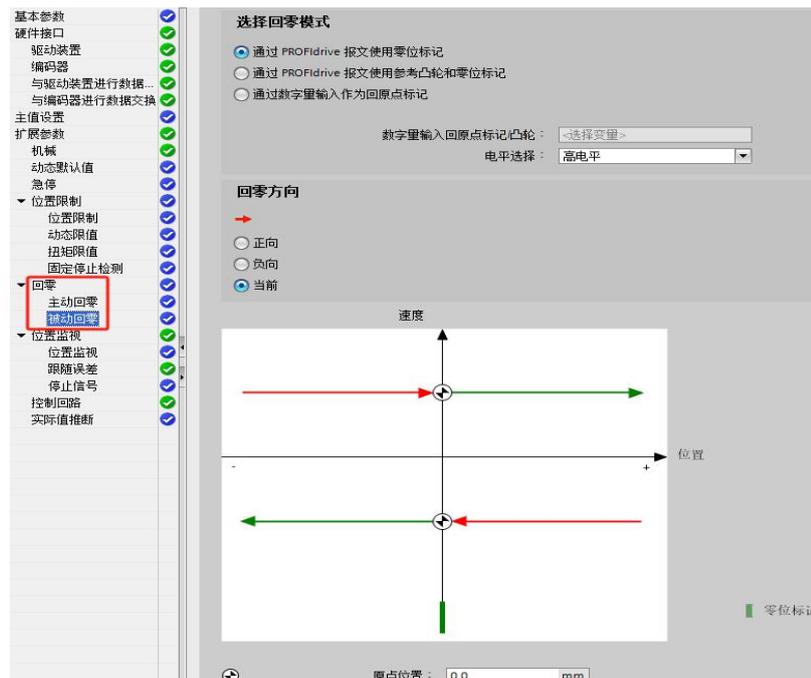
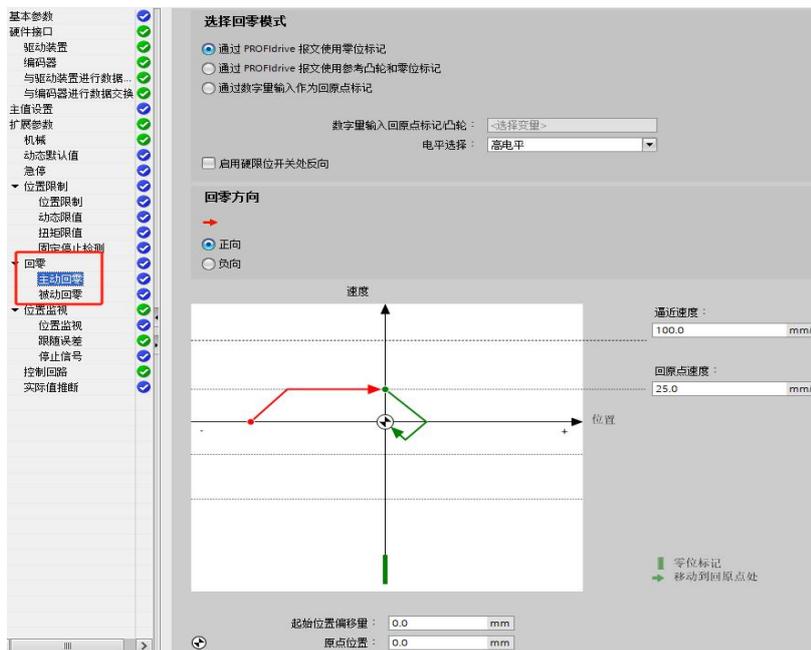


21. 组态完毕，编写程序，编译并下载到 PLC:



### 6.3 MC\_Home (回原点) 介绍

伺服在进行绝对定位之前必须使用 MC\_Home 指令回原点，伺服按照工艺对象组态的回原配置完成回原动作，回原轨迹由 PLC 生成，原点开关和限位开关通常接在 PLC 侧：



MC\_Home 引脚定义见表 6-2。

表 6-2 MC\_Home 引脚定义

引脚名称	数据类型	说 明
Axis	TO_PositioningAxis TO_SynchronousAxis TO_ExternalEncoder	工艺对象
Execute	Bool	上升沿触发回原点任务
Position	LReal	该指定值结合所选“Mode”使用
Mode	Int	回原点模式值 = 0 绝对式直接回原点，轴的位置值为“Position”的值 = 1 相对式直接回原点，轴的位置值为当前位置值+“Position”的值 = 2 被动回原点，轴的位置值为“Position”的值 = 3 主动回原点，轴的位置值为“Position”的值 = 6 绝对编码器相对调节，轴的位置值为当前位置值+“Position”的值 = 7 绝对编码器绝对调节，轴的位置值为“Position”的值
ReferenceMarkPosition	LReal	显示回原点的位置（“Done” = TRUE 时有效）
Done	Bool	= 1 原点已设定
Busy	Bool	= 1 任务进行中
CommandAborted	Bool	=1 回原点任务被另一个任务中止
Error	Bool	=1 出错，错误原因见“ErrorID”
ErrorID	Word	请查阅TIA portal的信息系统以获取帮助

不同编码器支持的回原点模式见表 6-3。

表 6-3 支持的回原点模式

回原点模式	增量式编码器 (步科单圈编码器)	绝对值编码器 (步科多圈绝对值编码器)
主动回原点 (“Mode” = 3)	√	—
被动回原点 (“Mode” = 2)	√	—
设置实际位置 (“Mode” = 0)	√	√
实际位置的相对偏移量 (“Mode” = 1)	√	√
绝对编码器调整 (“Mode” = 6、7)	—	√

### 6.3.1 回原点模式 0 (绝对式直接回原点)

在“Mode” = 0 模式下，触发 MC\_Home 指令后轴不会运动。执行指令的结果是：轴的实际位置直接变成“Position”引脚的值。下述示例中，“Position” = 0 mm，则回原后轴的实际位置变成了 0 mm。该坐标值属于“绝对”坐标值，即轴已建立了绝对坐标系，可进行绝对定位。

The screenshot shows the MC\_HOME function block configuration. The Position input is set to 0.0 and Mode is set to 0. The Done output is TRUE. The actual position output is 79979. Below the function block are two tables showing the actual position values.

索引	子索引	名称	数据类型
6064	00	实际位置	Integer32
Value			Unit
30815			DEC

帮助: 实际位置  
电机实际位置

PLC上电时间伺服实际位置

索引	子索引	名称	数据类型
6064	00	实际位置	Integer32
Value			Unit
79979			DEC

帮助: 实际位置  
电机实际位置

当前伺服实际位置

上图是未回原点的状态，轴的实际位置为 45 mm，当前伺服实际位置[606300]为 79979 DEC。在 6.2 节第 16 步中，已设置电机每转负载位移为 60 mm。换算关系如下：

$$\text{PLC 组态轴的实际位置(mm)} = (\text{当前伺服实际位置}[606300] - \text{PLC 上电时间伺服实际位置}[606300]) / \text{反馈精度}[641003] * \text{电机每转的负载位移}$$

或者

$$\text{PLC 组态轴的实际位置变化量(mm)} = \text{伺服实际位置}[606300]\text{变化量} / \text{反馈精度}[641003] * \text{电机每转的负载位移}$$

下图是已回原点的状态，轴的实际位置 = “Position” 的值 = 0 mm，而伺服的实际位置[606300]不会因为组态轴已回原点而改变，仍然是 79979 DEC。

The screenshot shows the MC\_HOME function block configuration after homing. The Position input is 0.0 and Mode is 0. The Done output is TRUE. The actual position output is 79979. Below the function block are two tables showing the actual position values.

索引	子索引	名称	数据类型
6064	00	实际位置	Integer32
Value			Unit
79979			DEC

帮助: 实际位置  
电机实际位置

当前伺服实际位置

回原点完成后, 无论轴运动到任何位置, 只要 PLC 断电重启, 轴的实际位置数据都会丢失, 变为 0 mm, 而不是以伺服当前的编码器数值换算过去的。

### 6.3.2 回原点模式 1 (相对式直接回原点)

在 “Mode” = 1 模式下, 触发 MC\_Home 指令后轴不会运动。执行指令的结果是: 轴的实际位置直接变成轴当前位置值 + “Position” 的值。下述示例中, “Position” = 10 mm, 执行回原指令前轴的当前位置为 20 mm, 则回原后轴的实际位置变成了 30 mm。

索引	子索引	名称	数据类型
6064	00	实际位置	Integer32
Value		Unit	
101949		DEC	

帮助: 实际位置  
电机实际位置

上图是未回原点的状态, 轴的当前位置为 20 mm, 伺服实际位置[606300]为 101949 DEC。

下图是已回原点的状态, 轴的实际位置 = 轴当前位置 + “Position” 的值 = 30 mm, 而伺服的实际位置[606300]不会因为组态轴已回原点而改变, 仍然是 101949 DEC。

索引	子索引	名称	数据类型
6064	00	实际位置	Integer32
Value		Unit	
101949		DEC	

帮助: 实际位置  
电机实际位置

回原点完成后, 无论轴运动到任何位置, 只要 PLC 断电重启, 轴的实际位置数据都会丢失, 变为 0 mm, 而不是以伺服当前的编码器数值换算过去的。

### 6.3.3 回原点模式 2 (被动回原点)

在 “Mode” = 2 模式下, 只触发 MC\_Home 指令时轴不会运动, 需要通过其它运动控制指令 (如 MC\_MoveRelative) 使轴运行起来, 等到轴碰到原点开关后, 轴的当前位置会变成 “Position” 的值。即轴在运动过程中检测到原点信号后, 会自动把实际位置变成 “Position” 的值。

被动回原点需要 MC\_Home 指令与 MC\_MoveAbsolute 或 MC\_MoveRelative 或 MC\_MoveVelocity 指令结合使用，轴在执行其它运动指令的过程中完成回原点。PLC 断电重启后，轴的实际位置数据会丢失。

被动回原点需要在工艺对象组态中进行配置。如下图，被动回原点有 3 种模式：通过 PROFIdrive 报文使用零位标记（即检测电机 Z 相信号的回原方式）、通过 PROFIdrive 报文使用参考凸轮和零位标记（即使用外部数字量信号作为减速点，然后检测电机 Z 相信号的回原方式）、通过数字量输入作为回原点标记（使用外部数字量信号作为原点开关）。



#### 1. 通过 PROFIdrive 报文使用零位标记（即检测电机 Z 相信号）

运行顺序如下：

- a. 触发回原点指令 MC\_Home，启动被动回原点。
- b. 通过其它运动控制指令（如 MC\_MoveRelative）使轴运动，当轴按照指定的回原点方向运动时，将启用回原点标记检测（检测电机 Z 相信号）。
- c. 当检测到电机 Z 相信号时，将轴的实际位置变成“Position”的值。

或者

- a. 通过其它运动控制指令（如 MC\_MoveRelative）使轴按照指定的回原点方向运动。
- b. 触发回原点指令 MC\_Home，启动被动回原点。
- c. 当检测到电机 Z 相信号时，将轴的实际位置变成“Position”的值。

#### 2. 通过 PROFIdrive 报文使用参考凸轮和零位标记（即检测到外部的输入信号后才开始检测电机 Z 相信号）

运行顺序如下：

- a. 触发回原点指令 MC\_Home，启动被动回原点。
- b. 通过其它运动控制指令（如 MC\_MoveRelative）使轴运动，等待外部输入信号。
- c. 检测到外部输入信号且一脱离外部输入信号后，就开始检测电机 Z 相信号。
- d. 当检测到电机 Z 相信号时，将轴的实际位置变成“Position”的值。

或者

- a. 通过其它运动控制指令（如 MC\_MoveRelative）使轴运动。
- b. 触发回原点指令 MC\_Home，启动被动回原点，等待外部输入信号。
- c. 检测到外部输入信号且一脱离外部输入信号后，就开始检测电机 Z 相信号。
- d. 当检测到电机 Z 相信号时，将轴的实际位置变成“Position”的值。

### 3. 通过数字量输入作为回原点标记 (即检测外部输入信号)

运行顺序如下:

- a. 触发回原点指令 MC\_Home, 启动被动回原点。
  - b. 通过其它运动控制指令 (如 MC\_MoveRelative) 使轴运动, 等待外部输入信号。
  - c. 当检测到外部输入信号时, 将轴的实际位置变成 “Position” 的值。
- 或者
- a. 通过其它运动控制指令 (如 MC\_MoveRelative) 使轴运动。
  - b. 触发回原点指令 MC\_Home, 启动被动回原点, 等待外部输入信号。
  - c. 当检测到外部输入信号时, 将轴的实际位置变成 “Position” 的值。

### 6.3.4 回原点模式 3 (主动回原点)

在 “Mode” = 3 模式下, 触发 MC\_Home 指令后轴会自动运行完成回原动作。

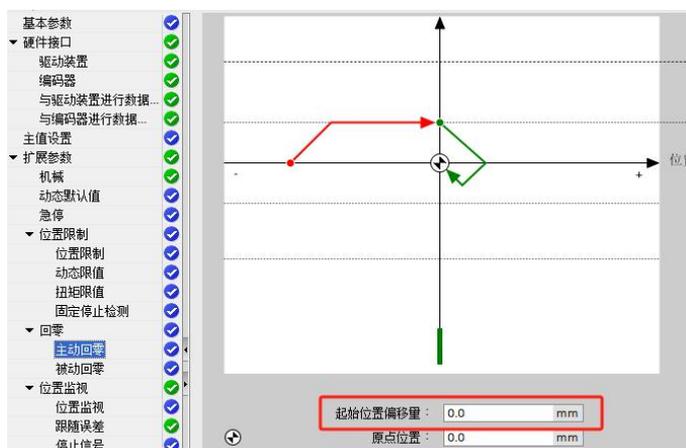
主动回原点需要在工艺对象组态中进行配置。如下图, 主动回原点有 3 种模式: 通过 PROFIdrive 报文使用零位标记 (即检测电机 Z 相信号的回原方式)、通过 PROFIdrive 报文使用参考凸轮和零位标记 (即使用外部数字量信号作为减速点, 然后检测电机 Z 相信号的回原方式)、通过数字量输入作为回原点标记 (使用外部数字量信号作为原点开关)。



#### 1. 通过 PROFIdrive 报文使用零位标记 (即检测电机 Z 相信号)

运行顺序如下:

- a. 触发回原点指令 MC\_Home, 启动主动回原点。
- b. 轴在运动过程中寻找电机 Z 相信号。
- c. 当检测到电机 Z 相信号时, 将实际位置变成 “Position” 的值- “起始位置偏移量” :



- d. 接着轴会运动一段“起始位置偏移量”的距离，完成后轴的实际位置等于“Position”的值。
2. 通过 PROFIdrive 报文使用参考凸轮和零位标记（即检测到外部的输入信号后才开始检测电机 Z 相信号）  
运行顺序如下：
    - a. 触发回原点指令 MC\_Home，启动主动回原点。
    - b. 轴在运动过程中检测外部输入信号。
    - c. 当检测到外部输入信号后减速，检测到外部输入信号的下降沿后，检测电机 Z 相信号。
    - d. 当检测到电机 Z 相信号时，将实际位置变成“Position”的值-“起始位置偏移量”。
    - e. 接着轴会运动一段“起始位置偏移量”的距离，完成后轴的实际位置等于“Position”的值。
  3. 通过数字量输入作为回原点标记（即检测外部输入信号）  
运行顺序如下：
    - a. 触发回原点指令 MC\_Home，启动主动回原点。
    - b. 轴在运动过程中检测外部输入信号。
    - c. 当检测到外部输入信号上升沿后减速，检测到外部输入信号的下降沿后，将实际位置变成“Position”的值-“起始位置偏移量”。
    - d. 接着轴会运动一段“起始位置偏移量”的距离，完成后轴的实际位置等于“Position”的值。

### 6.3.5 回原点模式 6 (绝对编码器相对调节)

此模式只针对连接的编码器类型为绝对值编码器，该模式下的 MC\_Home 指令触发后轴并不运行，也不会去寻找原点开关，会将当前位置值设为“当前位置值 + 参数 Position 的值”，绝对值保持性地保存在 CPU 内，CPU 断电再上电后轴的位置值不会丢失。

### 6.3.6 回原点模式 7 (绝对编码器绝对调节)

此模式只针对连接的编码器类型为绝对值编码器，该模式下的 MC\_Home 指令触发后轴并不运行，也不会去寻找原点开关，会将当前位置值设为“参数 Position 的值”，绝对值保持性地保存在 CPU 内，CPU 断电再上电后轴的位置值不会丢失。

## 6.4 MC\_TorqueLimiting (转矩限制) 介绍

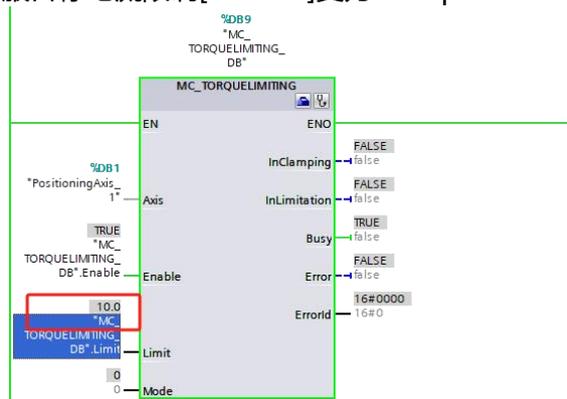
伺服的目标电流限制[607300]和最大扭矩限制 %[607200]是关联的。可使用 MC\_TorqueLimiting 指令对最大电流 (转矩) 进行限制。

1. MC\_TorqueLimiting 引脚定义见表 6-4。

表 6-4 MC\_TorqueLimiting 引脚定义

引脚名称	数据类型	说明
Axis	TO_SpeedAxis TO_PositioningAxis TO_SynchronousAxis	工艺对象
Enable	Bool	= 1 激活转矩限制功能
Limit	LReal	等于目标电流限制[607300]值，单位Ap 范围：0~驱动器/电机峰值电流
Mode	DInt	= 0 转矩限制模式 = 1 —
InClamping	Bool	—
InLimitation	Bool	= 1 实际电流[607800]达到目标电流限制[607300]
Busy	Bool	= 1 任务进行中
Error	Bool	= 1 出错，错误原因见“ErrorID”
ErrorID	Word	请查阅TIA portal的信息系统以获取使用帮助

2. 开启转矩限制功能，“Enable”引脚置 1，在“Limit”引脚输入目标电流限制值 10 (单位 Ap)，则伺服目标电流限制[607300]变为 10 Ap：



索引	子索引	名称	数据类型	属性
6073	00	目标电流限制	Unsigned16	RWSLTM
		Value	Unit	
		10.00	Ap	

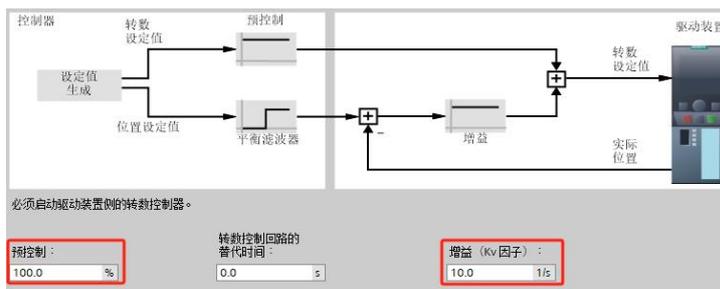
帮助: 目标电流限制  
电流指令最大值

3. 实际电流[607800]达到目标电流限制[607300]时，输出引脚“InLimitation”置 1，伺服返回的消息状态字 MELDW 比特 1 置 0 (见表 1-19)。

### 6.5 DSC (动态伺服控制) 介绍

使用报文 3 和 102 时，伺服工作在速度模式，位置环计算由 PLC 完成，位置环的计算更新周期与总线的通信周期相同。报文 5 和 105 支持 DSC (Dynamic Servo Control, 动态伺服控制)，开启 DSC 后，伺服工作在插补模式，位置环的计算更新周期为伺服本地的控制周期。DSC 功能将 PLC 的位置环计算放置于伺服中完成，PLC 只需进行位置的中心插补计算，再通过总线接口对伺服的位置环计算进行周期调整。

在工艺对象组态界面，点击“扩展参数→控制回路”，可对 DSC 功能相关参数进行设置，如下图所示：



上图中，“增益 (Kv 因子)” 不能为 0，其具体数值不会影响伺服的位置环比例增益，保持为默认值即可；“预控制” 为速度前馈百分比，与位置环速度前馈[60FB02]的作用相同：

N	Index	Type	Name	Value	Unit
0	60FB01	int16	位置环比例增益 [0]	10.00	Hz
1	60FB02	int16	位置环速度前馈	100.00	%
2	60FB03	int16	位置环加速度前馈	32767	DEC

请注意，在使用 DSC 功能时，如果连接的编码器类型为步科单圈编码器(增量式编码器)，则需要将实际位置保存使能[60FB06]设置为 1，然后对位置偏移[60FB07]写入 0，并存储控制参数：

N	Index	Type	Name	Value	Unit
0	60610C	int8	有效工作模式	0	DEC
1	60410C	uint16	状态字	4250	HEX
2	60630C	int32	实际位置	-17276	inc
3	606C0C	int32	实际速度	-0.24	rpm
4	60780C	int16	实际电流	0.00	Ap
5	26800C	uint16	警告状态字	0000	HEX
6	60600C	int8	工作模式	-3	DEC
7	60400C	uint16	控制字	0000	HEX
8	607A0C	int32	目标位置	0	inc
9	60810C	uint32	梯形速度	0.00	rpm
10	60830C	uint32	梯形加速度	100.00	rps/ε
11	60840C	uint32	梯形减速度	100.00	rps/ε
12	60FF0C	int32	目标速度	0.00	rpm
13	60710C	int16	目标扭矩%	0.00	%
14	60730C	uint16	目标电流限制	17.30	Ap
15	20200E	int8	工作模式选择0	-4	DEC
16	20200E	int8	工作模式选择1	-3	DEC
17	26900C	uint8	通讯编码器数据复	0	DEC
18	60FB06	uint8	实际位置保存使能	0	DEC
19	60FB07	int32	位置偏移	-17446	DEC



N	Index	Type	Name	Value	Unit
0	60610C	int8	有效工作模式	0	DEC
1	60410C	uint16	状态字	C250	HEX
2	60630C	int32	实际位置	-34721	inc
3	606C0C	int32	实际速度	-1.79	rpm
4	60780C	int16	实际电流	0.00	Ap
5	26800C	uint16	警告状态字	0000	HEX
6	60600C	int8	工作模式	-3	DEC
7	60400C	uint16	控制字	0000	HEX
8	607A0C	int32	目标位置	0	inc
9	60810C	uint32	梯形速度	0.00	rpm
10	60830C	uint32	梯形加速度	100.00	rps/s
11	60840C	uint32	梯形减速度	100.00	rps/s
12	60FF0C	int32	目标速度	0.00	rpm
13	60710C	int16	目标扭矩%	0.00	%
14	60730C	uint16	目标电流限制	17.30	Ap
15	20200E	int8	工作模式选择0	-4	DEC
16	20200E	int8	工作模式选择1	-3	DEC
17	26900C	uint8	通讯编码器数据复	0	DEC
18	60FB06	uint8	实际位置保存使能	1	DEC
19	60FB07	int32	位置偏移	0	DEC

## 第七章 S7-200 SMART 应用

### 7.1 应用说明

本章的 PROFINET 通讯应用基于以下条件：

- STEP 7-Micro/Win SMART V02.08.02.00\_00.01  
PLC S7-200 SMART(CPU ST20 DC/DC/DC V02.04.01\_00.00.03.00 固件 V2.6)
- Kinco PN 伺服

伺服型号	伺服固件	PN 固件
FDxx5-PA-004	版本年份为 2024 年及以后	00000005、10000005
FDxx5P-PA-000		
MDx0-0xx-DMxK-PA-000		

### 3. GSD 文件

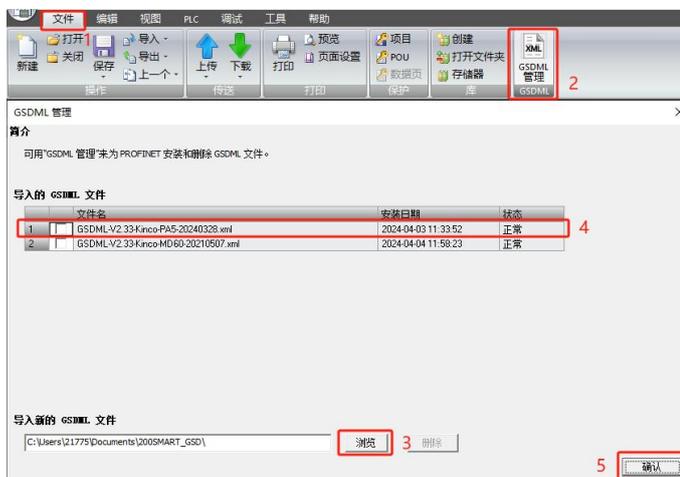
PN 固件版本	适用 GSD 文件
00000005、10000005	GSDML-V2.33-Kinco-PA5-20240328
00000004	GSDML-V2.33-Kinco-MD60-20210507

伺服的 PROFINET 通讯相关参数说明见表 2-1。

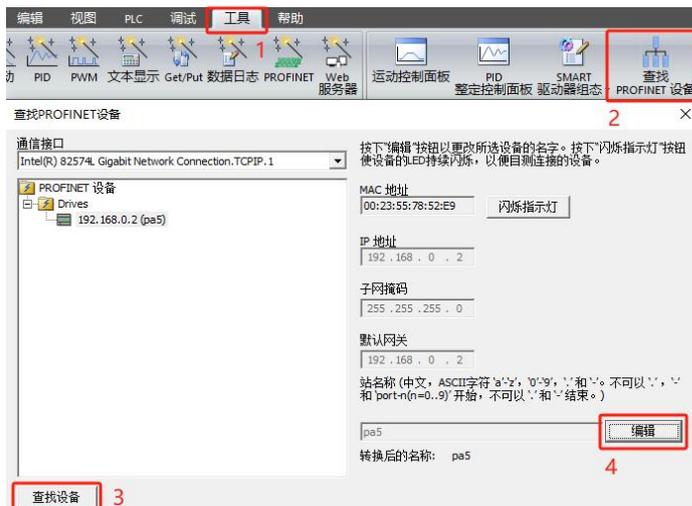
### 7.2 非周期通信

#### 7.2.1 项目配置

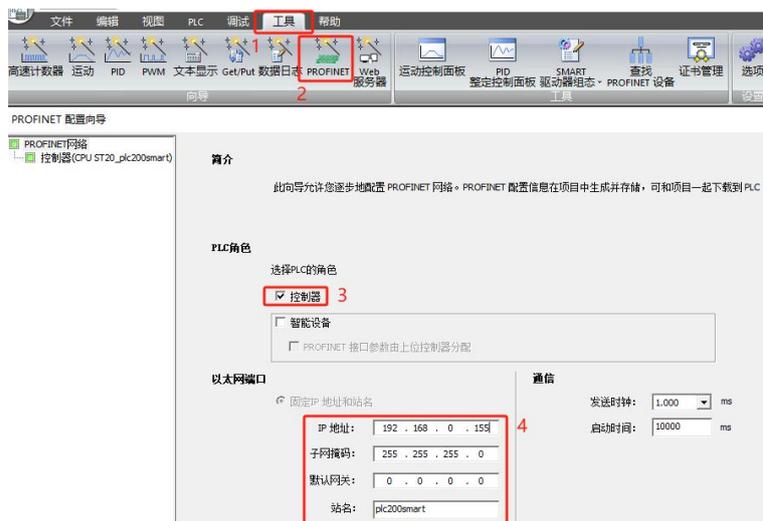
- 点击上方菜单“文件→GSDML 管理”安装 GSD 文件(GSDML-V2.33-Kinco-PA5-20240328)，安装成功后，新工程无须再次安装：



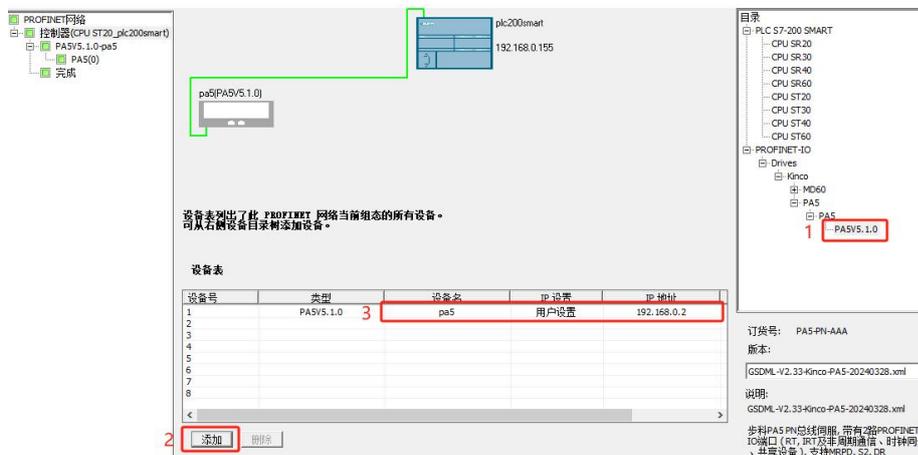
2. 点击“工具→查找 PROFINET 设备”，确保能连接到伺服，驱动器 IP 地址和设备名称的修改见 3.1 节，也可以在此处点击“编辑”修改设备名称：



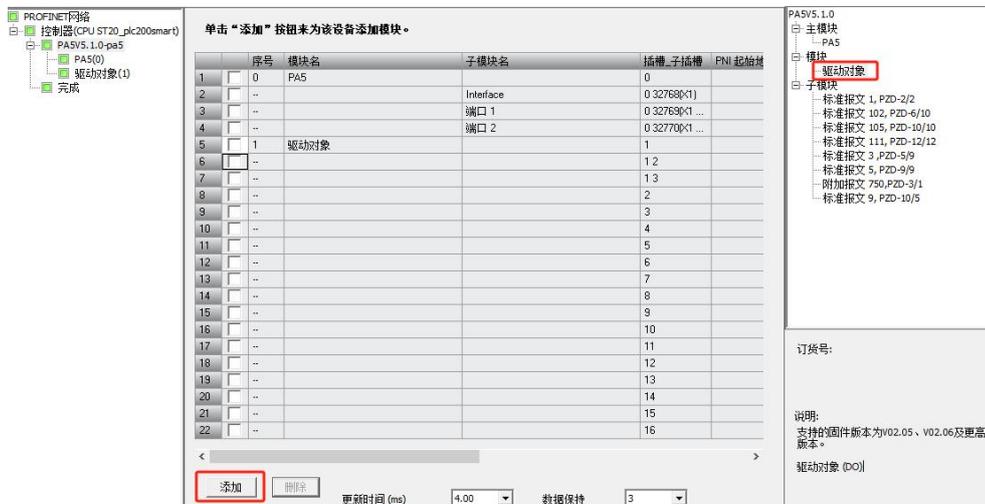
3. 点击“工具→PROFINET”，勾选“控制器”，并设置 PLC 的 IP 地址和名称，然后点击“下一步”：



4. 添加伺服到设备表中，并设置 IP 地址和设备名称(要与实际连接的设备一致)，然后点击“下一步”：



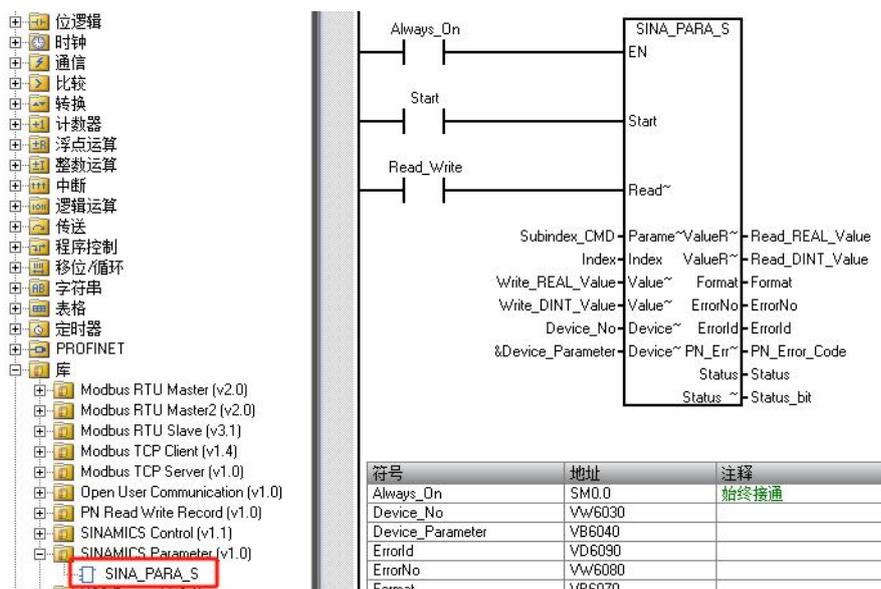
5. 给伺服添加通讯模块(驱动对象), 默认会添加标准报文 3 的子模块, 如果只进行非周期通信, 可以把报文字模块删除:



6. 随后一直点击“下一步”，直至最后生成:



7. 在主程序中, 调用 SINA\_PARA\_S, 编写以下程序:



8. 程序中使用的符号表地址定义如下图所示：

符号表	符号	地址	注释
1	Start	V6000.0	上升沿触发读写
2	Read_Write	V6000.1	0: 读 1: 写
3	Device_Parameter	VB6040	
4	Format	VB6070	
5	Status	VB6100	
6	Status_bit	VB6102	
7	Write_REAL_Value	VD6020	
8	Write_DINT_Value	VD6024	写入伺服的数据
9	APINumber	VD6042	
10	Read_REAL_Value	VD6060	
11	Read_DINT_Value	VD6064	从伺服读出的数据
12	ErrorId	VD6090	
13	PN_Error_Code	VD6094	
14	Subindex_CMD	Vw6010	参数子索引+发送命令字
15	Index	Vw6012	参数索引
16	Device_No	Vw6030	
17	SlotNumber	Vw6046	
18	SubSlotNumber	Vw6048	
19	ErrorNo	Vw6080	

9. 下载程序前，分配程序库使用的 V 地址区：



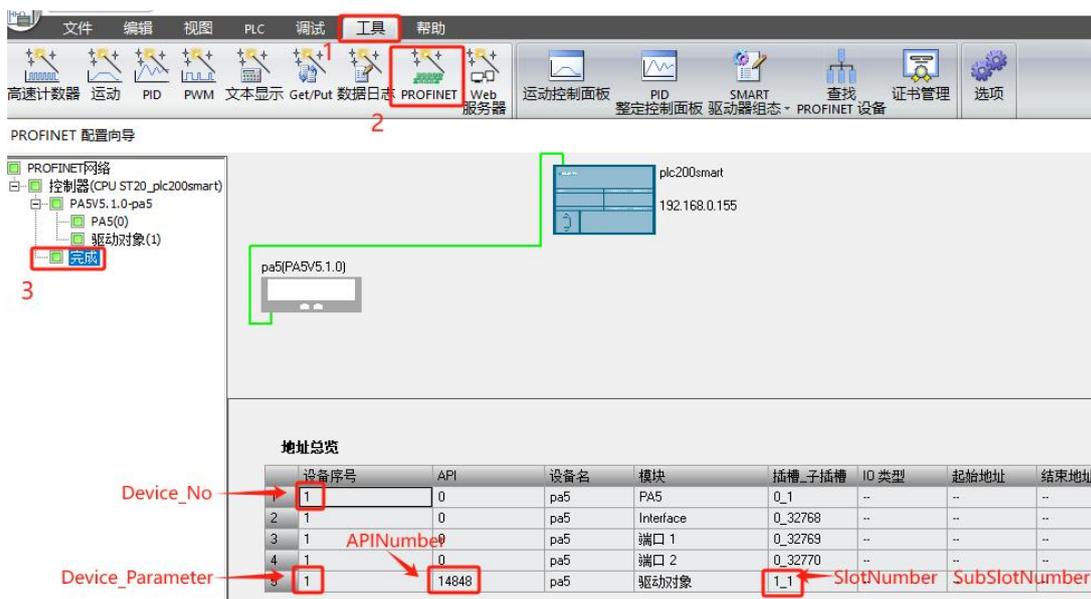
## 7.2.2 SINA\_PARA\_S 介绍

SINA\_PARA\_S 引脚定义见表7-1。

表7-1 SINA\_PARA\_S 引脚定义

引脚名称	数据类型	说 明
Start	Bool	上升沿触发数据发送
ReadWrite	Bool	0=读取; 1=写入
Parameter	Int	比特8~15: 参数子索引; 比特0~7: 发送命令字
Index	Int	参数索引
ValueWrite1	Real	保留
ValueWrite2	DInt	数据写入区, 可写入1~4字节
DeviceNo	Word	设备编号
Device_Parameter	DWord	PROFINET从站参数, 字节偏移如下: 0: 轴编号 1: 保留 2~5: API编号(固定值14848) 6~7: 插槽编号 8~9: 子插槽编号
ValueRead1	Real	保留
ValueRead2	DInt	数据读取区
Format	Byte	读取的参数的格式: 02H: 整型8 03H: 整型16 04H: 整型32 05H: 无符号8 06H: 无符号16 07H: 无符号32 08H: 浮点 10H: 八进制字符串(16位) 13H: 时间差(32位) 41H: 字节 42H: 字 43H: 双字 44H: 错误
ErrorNo	Word	根据PROFIdrive规范的错误编号
ErrorID	DWord	第一个字: 二进制编码, 指示出现故障的参数访问 第二个字: 故障类型
PN_Error_Code	DInt	根据PROFINET协议的错误代码
Status	Byte	比特0~4: 指令RDREC和WRREC的系统定义错误代码; 比特5: 错误; 比特6: 正在进行请求
Status_bit	Byte	指示读写参数的状态, 当读写成功时, 该值为4
说明: 参数索引及子索引、发送命令字等的含义, 请参照 Kinco 伺服使用手册通讯相关章节, 其中, 发送命令字: 读都是40H; 写1个字节2FH, 写2个字节2BH, 写4个字节23H。		

表7-1中的Device\_Parameter(设备参数)可通过下图确认：



### 7.2.3 SINA\_PARA\_S 读写参数示例

1. 读取反馈精度：参数索引为0x6410，参数子索引为0x03，读参数命令字统一为0x40，当“Start”引脚由0变为1后，数据读取区“ValueRead2”引脚自动变成10000，且“Status\_bit”引脚为4，数据读取成功。

N	Index	Type	Name	Value	Unit
0	60F612	uint16	电机实际Iit	0.00	%
1	641016	uint16	当前电机型号	64	ASCII
2	304106	uint8	使用内部电机库	1	DEC
3	641001	uint16	电机型号	64	ASCII
4	641002	uint8	反馈类型	04	HEX
5	641003	uint32	反馈精度	10000.00	inc/r
6	641004	uint32	反馈周期	327702	DEC

**程序段注释**

```

Always_On=ON
Start=ON
Read_Write=OFF
SINA_PARA_S
EN
Start
Read~
+832 Subind~ Read~ -0.0
+25616 Index Read~ +10000
0.0 Write~ Format -67
+0 Write~ ErrorNo -0
1 Device~ ErrorId -0
16#08001798 &Devic~ PN_Err~ -+0
Status -0
Status ~ -4
                    
```

符号	地址	注释
Always_On	SM0.0	始终接通
Device_No	VW6030	
Device_Parameter	VB6040	

**状态图表**

地址	格式	当前值	新值
1	Start	位	2#1
2	Read_Write	位	2#0
3	Index	十六进制	16#6410
4	Subindex_CMD	十六进制	16#0340
5	Write_DINT_Value	有符号	+0
6	Read_DINT_Value	有符号	+10000
7	Format	无符号	67
8	Device_No	有符号	+1
9	Device_Parameter	无符号	1
10	APINumber	有符号	+14848
11	SlotNumber	有符号	+1
12	SubSlotNumber	有符号	+1
13	ErrorNo	有符号	+0
14	ErrorId	有符号	+0
15	Status	无符号	0
16	Status_bit	无符号	4

2. 写入目标位置：参数索引为0x607A，参数子索引为0x00，数据类型为Int32，所以写参数命令字为0x23，将“ReadWrite”引脚切换为1，数据写入区写入-10000，当“Start”引脚由0变为1后，将数据写入，“Status\_bit”引脚为4，数据写入成功。观察上位机软件，目标位置被改为-10000。

地址	格式	当前值	新值
1	Start	位	2#1
2	Read_Write	位	2#1
3	Index	十六进制	16#607A
4	Subindex_CMD	十六进制	16#0023
5	Write_DINT_Value	有符号	-10000
6	Read_DINT_Value	有符号	+0
7	Format	无符号	67
8	Device_No	有符号	+1
9	Device_Parameter	无符号	1
10	APINumber	有符号	+14848
11	SlotNumber	有符号	+1
12	SubSlotNumber	有符号	+1
13	ErrorNo	有符号	+0
14	ErrorId	有符号	+0
15	Status	无符号	0
16	Status_bit	无符号	4

6	606000	int8	工作模式	1	DEC
7	604000	uint16	控制字	0006	HEX
8	607A00	int32	目标位置	-10000	inc
9	608100	uint32	梯形速度	0.00	rpm
10	608300	uint32	梯形加速度	100.00	rps/s
11	608400	uint32	梯形减速度	100.00	rps/s

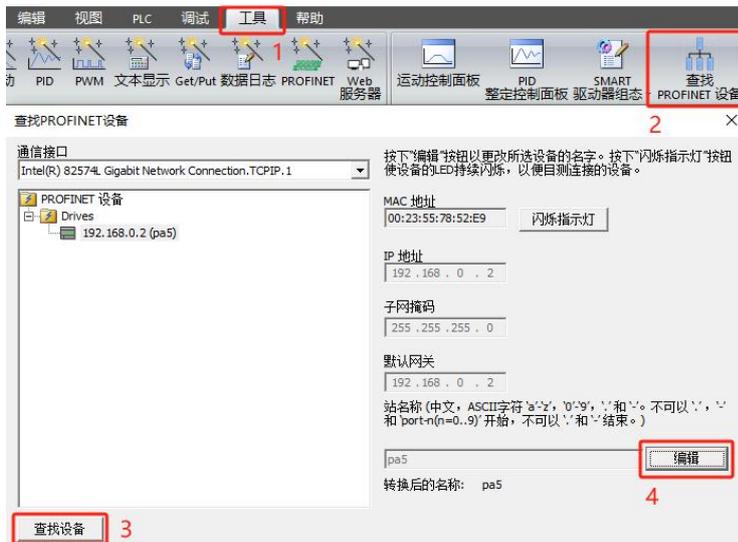
## 7.3 报文 1 应用

### 7.3.1 项目配置

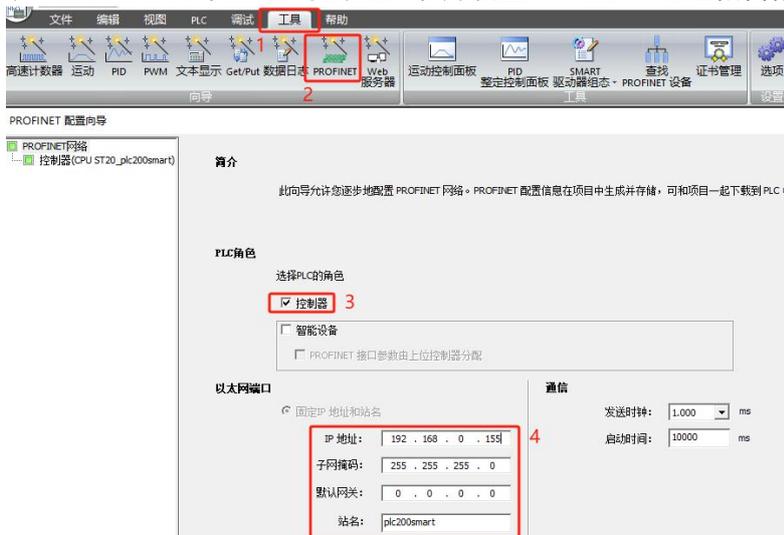
1. 点击上方菜单“文件→GSDML 管理”安装 GSD 文件(GSDML-V2.33-Kinco-PA5-20240328)，安装成功后，新工程无须再次安装：

文件名	安装日期	状态	
1	GSDML-V2.33-Kinco-PA5-20240328.xml	2024-04-03 11:33:52	正常
2	GSDML-V2.33-Kinco-MD60-20210507.xml	2024-04-04 11:58:23	正常

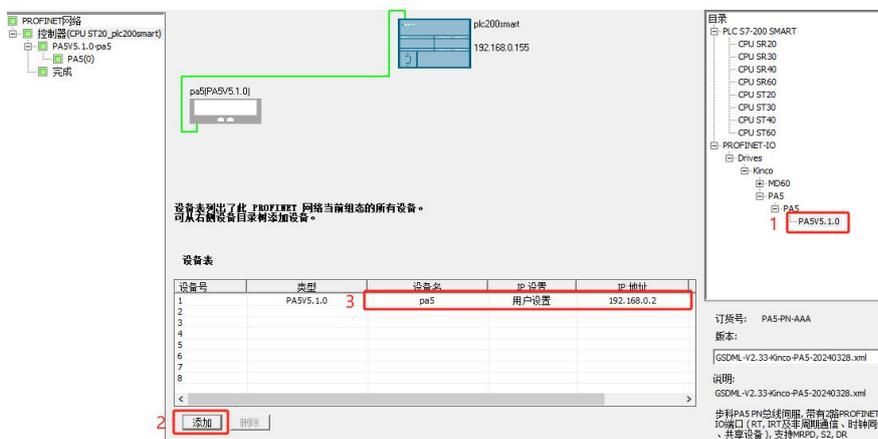
2. 点击“工具→查找 PROFINET 设备”，确保能连接到伺服，驱动器 IP 地址和设备名称的修改见 3.1 节，也可以在此处点击“编辑”修改设备名称：



3. 点击“工具→PROFINET”，勾选“控制器”，并设置 PLC 的 IP 地址和名称，然后点击“下一步”：



4. 添加伺服到设备表中，并设置 IP 地址和设备名称(要与实际连接的设备一致)，然后点击“下一步”：



5. 给伺服添加通讯模块(驱动对象), 然后选中并删除默认添加的标准报文 3 的子模块, 再添加报文 1 的子模块, 并记下 I/O 数据的起始地址, 在调用 SINA\_SPEED 时需要使用:

序号	模块名	子模块名	插槽_子插...	PN1 起始地址	输入长...	PNQ 起始地址	输出长度
1	0	PA5	0				
2	--	Interface	0_32768				
3	--	端口 1	0_32769				
4	--	端口 2	0_32770				
5	1	驱动对象	1				
6	✓	--	标准报文 1, PZD-2/2	146	4	138	4
7	--		1_3				
8	--		2				
9	--		3				
10	--		4				
11	--		5				
12	--		6				
13	--		7				
14	--		8				
15	--		9				
16	--		10				
17	--		11				
18	--		12				
19	--		13				
20	--		14				
21	--		15				
22	--		16				

模块

- 驱动对象
  - 标准报文 1, PZD-2/2
  - 标准报文 102, PZD-6/10
  - 标准报文 105, PZD-10/10
  - 标准报文 111, PZD-12/12
  - 标准报文 3, PZD-5/9
  - 标准报文 5, PZD-9/9
  - 附加报文 750, PZD-3/1
  - 标准报文 9, PZD-10/5

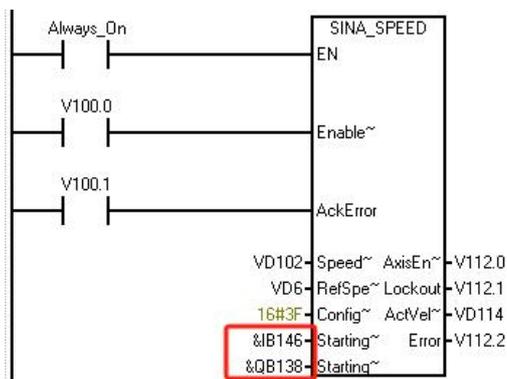
添加 删除 更新时间 (ms) 4.00 数据保持 3

6. 随后一直点击 “下一步”, 直至最后生成:

The screenshot shows the 'STEP 7-Micro/WIN SMART' interface. On the left, a tree view shows the network configuration: PROFINET网络, 控制器(CPU ST20\_plc200smart), PA5V5.1.0-pa5, PA5(0), 驱动对象(1), and 完成. On the right, a 'plc200smart' block is shown with IP address 192.168.0.155. A dialog box is open with the message: '如果已配置PROFINET网络, 系统会自动调整“系统块”对话框“通信”节点中的“背景时间”到20%.' The '确定' button is highlighted with a red box. Below the dialog, a table shows the module configuration:

插槽	子插槽	地址	名称	地址	名称	插槽_子插槽
1	1	0	pa5	PA5		0_1
2	1	0	pa5	Interface		0_32768
3	1	0	pa5	端口 1		0_32769

7. 在主程序中, 调用 SINA\_SPEED, 编写以下程序, 其中输入 “Starting\_I\_add” 和 “Starting\_Q\_add” 必须和报文 1 的 I/O 数据起始地址对应(见第 5 步):



8. 下载程序前，分配程序库使用的 V 地址区：



7.3.2 SINA\_SPEED 介绍

SINA\_SPEED 引脚定义见表7-2。

表7-2 SINA\_SPEED 引脚定义

引脚名称	数据类型	说 明																
EnableAxis	Bool	=1 使能																
AckError	Bool	复位故障，上升沿有效																
SpeedSp	Real	速度设定值，单位RPM																
RefSpeed	Real	参考速度，必须为最大速度[607F00]的1/2，单位RPM																
ConfigAxis	Word	默认16#3F，与STW1的对应关系如下： <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>ConfigAxis 中的比特</th> <th>STW1 中的比特</th> </tr> </thead> <tbody> <tr><td>比特0</td><td>比特1</td></tr> <tr><td>比特1</td><td>比特2</td></tr> <tr><td>比特2</td><td>比特3</td></tr> <tr><td>比特3</td><td>比特4</td></tr> <tr><td>比特4</td><td>比特5</td></tr> <tr><td>比特5</td><td>比特6</td></tr> <tr><td>比特6</td><td>比特11</td></tr> </tbody> </table> STW1中各比特赋值意义见表1-6	ConfigAxis 中的比特	STW1 中的比特	比特0	比特1	比特1	比特2	比特2	比特3	比特3	比特4	比特4	比特5	比特5	比特6	比特6	比特11
ConfigAxis 中的比特	STW1 中的比特																	
比特0	比特1																	
比特1	比特2																	
比特2	比特3																	
比特3	比特4																	
比特4	比特5																	
比特5	比特6																	
比特6	比特11																	
Starting_I_add	DWord	报文1的PN I存储区起始地址的指针																
Starting_Q_add	DWord	报文1的PN Q存储区起始地址的指针																
AxisEnabled	Bool	=1 轴已使能																
Lockout	Bool	=1 禁止接通																
ActVelocity	Real	速度实际值，单位RPM																
Error	Bool	=1 存在故障																

## 7.4 报文 111 应用

### 7.4.1 项目配置

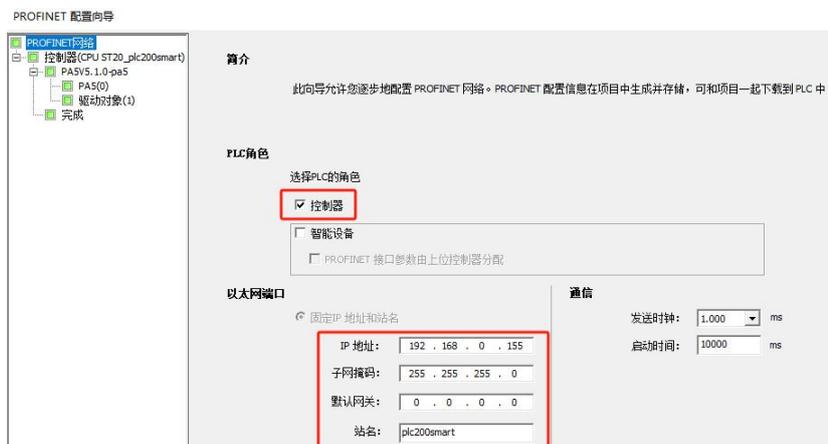
1. 点击上方菜单“文件→GSDML 管理”安装 GSD，安装成功后，新工程无须再次安装：



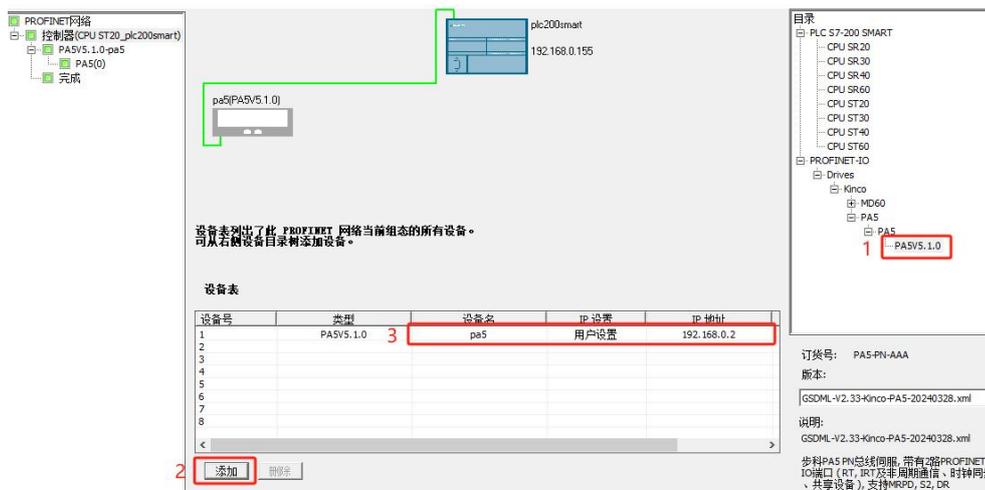
2. 点击“工具→查找 PROFINET 设备”，确保能连接到伺服，驱动器 IP 地址和设备名称的修改见 3.1 节，也可以在此处点击“编辑”修改设备名称：



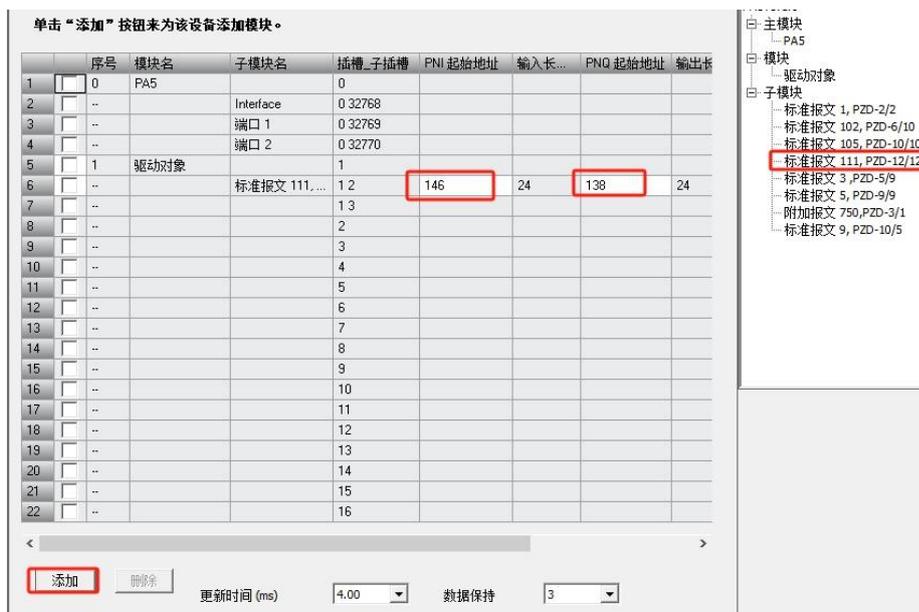
3. 点击“工具→PROFINET”，勾选“控制器”，并设置 PLC 的 IP 地址和名称，然后点击“下一步”：



4. 添加伺服到设备表中, 并设置 IP 地址和设备名称(要与实际连接的设备一致), 然后点击 “下一步” :



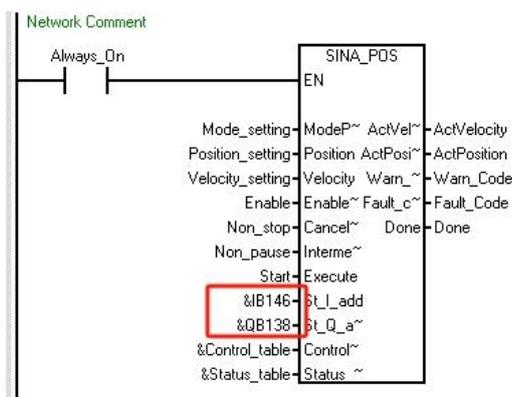
5. 给伺服添加通讯模块(驱动对象), 然后选中并删除默认添加的标准报文 3 的子模块, 再添加报文 111 的子模块, 并记下 I/O 数据的起始地址, 在调用 SINA\_POS 时需要使用:



6. 随后一直点击 “下一步” , 直至最后生成:



- 在主程序中，调用 SINA\_POS，编写以下程序，其中输入 “St\_I\_add” 和 “St\_Q\_add” 必须和报文 111 的 I/O 数据起始地址对应(见第 5 步)：



- 程序中使用的符号表地址定义如下所示：

Mode_setting	VW7000
Position_setting	VD7002
Velocity_setting	VD7006
Enable	V7010.0
Non_stop	V7010.1
Non_pause	V7010.2
Start	V7010.3
Control_table	VD8000
Status_table	VD7500
ActPosition	VD7020
ActVelocity	VD7024
Warn_Code	VW7028
Fault_Code	VW7030
Done	V7032.0
OverV	VW8002
OverAcc	VW8004
OverDec	VW8006
ConfigEpos	VD8008

- 下载程序前，分配程序库使用的 V 地址区：



10. 下载程序后，可通过状态图表进行相关功能测试：

The screenshot displays a PLC programming environment. On the left, a ladder logic network is shown with a normally open contact labeled 'Always\_On=ON' connected to a coil labeled 'SINA\_POS'. Below the network, a list of parameters for the SINA\_POS module is provided, including Mode\_~, ActVel~, Positio~, ActPosi~, Velocit~, Warn\_~, Enable, Fault\_C~, Non\_st~, Done, Non\_p~, Start, and their corresponding addresses. On the right, a '状态图表' (Status Table) window is open, showing a table with columns for address, format, current value, and new value. The table lists 12 parameters with their current values. Below the status table, a symbol table lists the symbols used in the program and their addresses.

符号	地址
ActPosition	VD7020
ActVelocity	VD7024
Always_On	SM0.0
Control_table	VD8000
Done	V7032.0
Enable	V7010.0

地址	格式	当前值	新值	
1	Mode_setting	有符号	+1	
2	Position_setting	有符号	+1000000	
3	Velocity_setting	有符号	+3579139	
4	Enable	位	2#1	
5	Non_stop	位	2#1	
6	Non_pause	位	2#1	
7	Start	位	2#1	
8	OverV	有符号	+100	
9	OverAcc	有符号	+100	
10	OverDec	有符号	+100	
11	ConfigEpos	有符号	+3	
12	Qw138	二进制	2#0000_0100_0111_1111	

### 7.4.2 SINA\_POS 介绍

SINA\_POS 引脚定义见表7-3。

表7-3 SINA\_POS 引脚定义

引脚名称	数据类型	说明
ModePos	Int	运行模式： =0 — =1 相对定位模式 =2 绝对定位模式 =3 — =4 主动回原点模式 =5 直接设置原点模式 =6 位置表模式(未支持) =7 按指定速度点动模式 =8 —
Position	DInt	运行模式为1、2时的位置设定值 运行模式为6时的位置表起始/新任务索引，范围0~7
Velocity	DInt	运行模式为1、2、7时的速度设定值
EnableAxis	Bool	=0 松轴，伺服控制字为0xE =1 使能，伺服控制字为0xF
CancelTraversing	Bool	=1 接受定位任务 =0 拒绝定位任务
IntermediateStop	Bool	=1 不暂停定位任务 =0 暂停定位任务
Execute	Bool	激活运行模式，上升沿有效
St_I_add	DWord	报文111的PN I存储区起始地址的指针
St_Q_add	DWord	报文111的PN Q存储区起始地址的指针

引脚名称	数据类型	说明																																																																																													
Control_table	DWord	Control table 起始地址的指针, 例如&VD8000																																																																																													
		<table border="1"> <thead> <tr> <th>字节</th> <th>位7</th> <th>位6</th> <th>位5</th> <th>位4</th> <th>位3</th> <th>位2</th> <th>位1</th> <th>位0</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>保留</td> <td>保留</td> <td>Ack-Error</td> <td>保留</td> <td>Jog2</td> <td>Jog1</td> <td>Negative</td> <td>Positive</td> </tr> <tr> <td>1</td> <td colspan="8">保留</td> </tr> <tr> <td>2</td> <td colspan="8" rowspan="2">OverV: 运行模式为1、2、7时的速度设定值百分比0~199%</td> </tr> <tr> <td>3</td> </tr> <tr> <td>4</td> <td colspan="8" rowspan="2">OverAcc: 运行模式为1、2、7时的加速度百分比0~100%</td> </tr> <tr> <td>5</td> </tr> <tr> <td>6</td> <td colspan="8" rowspan="2">OverDec: 运行模式为1、2、7时的减速度百分比0~100%</td> </tr> <tr> <td>7</td> </tr> <tr> <td>8</td> <td colspan="8">ConfigEpos ——</td> </tr> <tr> <td>9</td> <td colspan="4">比特0: 惯性停止</td> <td colspan="4">比特1: 快速停止</td> </tr> <tr> <td>10</td> <td colspan="4">比特2: 激活软件限位</td> <td colspan="4">比特3: 激活硬件限位</td> </tr> <tr> <td>11</td> <td colspan="8">比特6: 原点开关信号 比特8: 绝对定位设定值立即改变 请注意, 需确保ConfigEpos的初始值为3</td> </tr> </tbody> </table>	字节	位7	位6	位5	位4	位3	位2	位1	位0	0	保留	保留	Ack-Error	保留	Jog2	Jog1	Negative	Positive	1	保留								2	OverV: 运行模式为1、2、7时的速度设定值百分比0~199%								3	4	OverAcc: 运行模式为1、2、7时的加速度百分比0~100%								5	6	OverDec: 运行模式为1、2、7时的减速度百分比0~100%								7	8	ConfigEpos ——								9	比特0: 惯性停止				比特1: 快速停止				10	比特2: 激活软件限位				比特3: 激活硬件限位				11	比特6: 原点开关信号 比特8: 绝对定位设定值立即改变 请注意, 需确保ConfigEpos的初始值为3							
		字节	位7	位6	位5	位4	位3	位2	位1	位0																																																																																					
		0	保留	保留	Ack-Error	保留	Jog2	Jog1	Negative	Positive																																																																																					
		1	保留																																																																																												
		2	OverV: 运行模式为1、2、7时的速度设定值百分比0~199%																																																																																												
		3																																																																																													
		4	OverAcc: 运行模式为1、2、7时的加速度百分比0~100%																																																																																												
		5																																																																																													
		6	OverDec: 运行模式为1、2、7时的减速度百分比0~100%																																																																																												
		7																																																																																													
		8	ConfigEpos ——																																																																																												
9	比特0: 惯性停止				比特1: 快速停止																																																																																										
10	比特2: 激活软件限位				比特3: 激活硬件限位																																																																																										
11	比特6: 原点开关信号 比特8: 绝对定位设定值立即改变 请注意, 需确保ConfigEpos的初始值为3																																																																																														
Status_table	DWord	Status table 起始地址的指针, 例如&VD7500																																																																																													
		<table border="1"> <thead> <tr> <th>字节</th> <th>位7</th> <th>位6</th> <th>位5</th> <th>位4</th> <th>位3</th> <th>位2</th> <th>位1</th> <th>位0</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>保留</td> <td>Over-range</td> <td>Axis-Error</td> <td>Axis-Warn</td> <td>Lock-out</td> <td>Axis-Ref</td> <td>Axis-PosOk</td> <td>Axis-Enabled</td> </tr> <tr> <td>1</td> <td colspan="8">Error ID (错误代码) —— 0: 无错误 1: 检测到驱动器错误 2: 驱动器已禁用 3: 不支持所选模式 4: OverV、OverAcc 和 OverDec 的设置超出限制</td> </tr> <tr> <td>2</td> <td colspan="8" rowspan="2">ActMode: 当前激活的运行模式</td> </tr> <tr> <td>3</td> </tr> <tr> <td>4</td> <td colspan="8" rowspan="2">POS_ZSW1 当前值, 见表1-15</td> </tr> <tr> <td>5</td> </tr> <tr> <td>6</td> <td colspan="8" rowspan="2">POS_ZSW2 当前值, 见表1-16</td> </tr> <tr> <td>7</td> </tr> </tbody> </table>	字节	位7	位6	位5	位4	位3	位2	位1	位0	0	保留	Over-range	Axis-Error	Axis-Warn	Lock-out	Axis-Ref	Axis-PosOk	Axis-Enabled	1	Error ID (错误代码) —— 0: 无错误 1: 检测到驱动器错误 2: 驱动器已禁用 3: 不支持所选模式 4: OverV、OverAcc 和 OverDec 的设置超出限制								2	ActMode: 当前激活的运行模式								3	4	POS_ZSW1 当前值, 见表1-15								5	6	POS_ZSW2 当前值, 见表1-16								7																																				
		字节	位7	位6	位5	位4	位3	位2	位1	位0																																																																																					
		0	保留	Over-range	Axis-Error	Axis-Warn	Lock-out	Axis-Ref	Axis-PosOk	Axis-Enabled																																																																																					
		1	Error ID (错误代码) —— 0: 无错误 1: 检测到驱动器错误 2: 驱动器已禁用 3: 不支持所选模式 4: OverV、OverAcc 和 OverDec 的设置超出限制																																																																																												
		2	ActMode: 当前激活的运行模式																																																																																												
		3																																																																																													
		4	POS_ZSW1 当前值, 见表1-15																																																																																												
5																																																																																															
6	POS_ZSW2 当前值, 见表1-16																																																																																														
7																																																																																															
ActVelocity	DWord	速度实际值																																																																																													
ActPosition	DWord	位置实际值																																																																																													
Warn_code	Word	伺服报警代码																																																																																													
Fault_code	Word	驱动器错误代码[603F00]																																																																																													
Done	Bool	当操作模式为相对定位或绝对定位时达到目标位置																																																																																													

### 7.4.3 运行条件和设置

- 通过输入引脚“EnableAxis” = 1 使能, 如果轴已准备好且无故障 (“AxisError” = 0), 则 “Status\_table” 中的 “AxisEnabled” 为 1。
- 输入引脚 “CancelTraversing” 和 “IntermediateStop” 对于除点动模式之外的所有模式均有效, 在运行时应设为 1, 说明如下:
  - “CancelTraversing” = 0 时, 表示取消当前任务, 停止减速度对应暂停模式[605D00]。轴停止后可以切换运行模式。
  - “IntermediateStop” = 0 时, 表示暂停当前任务, 停止减速度对应暂停模式[605D00]。重新设置

“IntermediateStop” = 1 后轴会继续运行。轴停止后可以切换运行模式。

3. 输入引脚 “ModePos” 用于选择运行模式；“Execute” 的上升沿触发定位运动。

#### 4. 激活软限位开关

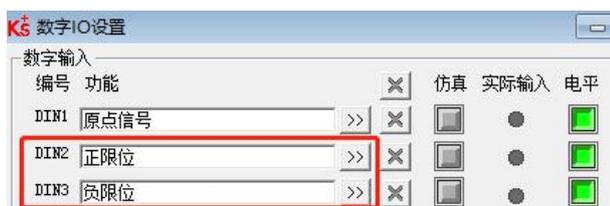
如果需要使用软件限位开关，需要将 “ConfigEPos” 的比特 2 置 1 ( “ConfigEPos” = 16#00000007) 或者将限位使能[30800C]置 1。在伺服中设置软限位正设置[607D01]，软限位负设置[607D02]：

30800C	uint8	软件限位使能	1	DEC
607D01	int32	软限位正设置	10000000	DEC
607D02	int32	软限位负设置	-10000000	DEC

设置 “软限位正设置” > “软限位负设置” ，并且找到原点之后，软件限位功能开始启用。

#### 5. 激活硬限位开关

如果需要使用硬件限位开关，需要将 “ConfigEPos” 的比特 3 置 1 ( “ConfigEPos” = 16#0000000B)：在上位机软件为驱动器的 DIN 定义正限位和负限位。只有在硬件限位开关信号为高电平时才能运行轴。



请注意，只是在上位机软件中配置正负限位，而 “ConfigEPos” 的比特 3 没有置 1，则硬件限位功能不生效。

### 7.4.4 运行模式 1 (相对定位)

要求：

1. 运行模式 “ModePos” = 1
2. “ConfigEPos” = 16#00000003
3. 使能轴 “EnableAxis” = 1
4. “CancelTraversing” 和 “IntermediateStop” 必须为 1，“Jog1” 和 “Jog2” 必须为 0。

步骤：

1. 通过输入引脚 “Position”，“Velocity” 指定目标位置和目标速度，位置和速度单位均为 DEC，参考伺服手册换算，“Velocity” 的单位转化为：DEC = RPM \* 512 \* 反馈精度[641003] / 1875。
2. 通过 “OverV”、“OverAcc”、“OverDec” 对目标速度、梯形加减速度进行百分比缩放。
3. 运动方向由 “Position” 中设置值的正负来确定。
4. 通过 “Execute” 的上升沿触发运动，到达目标位置后 “AxisPosOk” 置 1。若定位时出现错误，则 “AxisError” 置 1。
5. 当前运行的命令可通过 “Execute” 上升沿进行新命令替换。在任意时刻，可在运行模式 1、2

间切换。如果需要切换到其它运行模式，轴必须为静止状态。

6. 请注意，在通过“Execute”的上升沿触发相对定位运动后，需将“Execute”复位为0。

#### 7.4.5 运行模式 2 (绝对定位)

要求：

1. 运行模式“ModePos” = 2
2. “ConfigEPos” = 16#00000003
3. 使能轴“EnableAxis” = 1
4. “CancelTraversing”和“IntermediateStop”必须为1，“Jog1”和“Jog2”必须为0。

步骤：

1. 通过输入引脚“Position”，“Velocity”指定目标位置和目标速度，位置和速度单位均为DEC，参考伺服手册换算，“Velocity”的单位转化为： $DEC = RPM * 512 * \text{反馈精度}[641003] / 1875$ 。
2. 通过“OverV”、“OverAcc”、“OverDec”对目标速度、梯形加减速度进行百分比缩放。
3. 运行时间伺服按照最短路径运行至目标位置，“Positive”及“Negative”必须为0。
4. 通过“Execute”的上升沿触发运动，当到达目标位置后“AxisPosOk”置1。若定位过程中出现错误，则“AxisError”置1。
5. 当前运行的命令可通过“Execute”上升沿进行新命令替换。在任意时刻，可在运行模式1、2间切换。如果需要切换到其它运行模式，轴必须为静止状态。
6. 通过置位“ConfigEPos”的比特8（“ConfigEPos” = 16#00000103），在PLC侧更新“Position”后，无需触发“Execute”，新的设定值会立即生效，伺服会根据目标位置变化立即执行绝对定位指令。
7. 请注意，在通过“Execute”的上升沿触发绝对定位运动后，需将“Execute”复位为0。

#### 7.4.6 运行模式 4 (主动回原点)

要求：

1. 运行模式“ModePos” = 4
2. “ConfigEPos” = 16#00000003。若使用硬件限位，则“ConfigEPos” = 16#0000000B。
3. 使能轴“EnableAxis” = 1
4. “CancelTraversing”和“IntermediateStop”必须为1，“Jog1”和“Jog2”必须为0。
5. “Negative”和“Positive”任意一位必须置1。

步骤：

1. 回原点的方式以及回原点速度，需用上位机软件进行配置（PLC只发送回原点命令，回原点的规划由伺服参数设定），具体回原点方式请参考伺服手册：

N	Index	Type	Name	Value	Unit
0	607C00	int32	原点偏移	0	inc
1	609800	int8	原点模式	1	DEC
2	609901	uint32	原点转折信号速度	300.00	rpm
3	609902	uint32	原点信号速度	100.00	rpm
4	609903	uint8	上电找原点	0	DEC
5	609A00	uint32	原点加速度	50.00	rps/s
6	609904	int16	寻找原点最大电流	5.94	Ap
7	609905	uint8	原点偏移模式	0	DEC
8	609906	uint8	原点索引信号盲区	0	DEC

- 通过“Execute”的上升沿触发回原点运动，在回原过程中应保持为1。回原点完成后“AxisRef”置1，若运行过程中出现错误，则“AxisError”置1。

#### 7.4.7 运行模式 5 (直接回原点)

要求:

- 运行模式“ModePos” = 5
- “ConfigEPos” = 16#00000003
- 使能轴“EnableAxis” = 1
- 轴必须为静止状态。

步骤:

通过“Execute”的上升沿设置轴的原点，回原点完成后“AxisRef”置1。

#### 7.4.8 运行模式 7 (按指定速度点动)

要求:

- 运行模式“ModePos” = 7
- “ConfigEPos” = 16#00000003
- 使能轴“EnableAxis” = 1
- 轴必须为静止状态。

步骤:

- 通过输入引脚“Velocity”指定点动速度，必须为正值，单位DEC。
- 通过“OverV”、“OverAcc”、“OverDec”对点动速度、梯形加减速度进行百分比缩放。
- “CancelTraversing”以及“IntermediateStop”与点动模式无关。运行方向与“Positive”以及“Negative”无关。
- 无需触发“Execute”，“Jog1” = 1时反向点动，“Jog2” = 1时正向点动。“Jog1”和“Jog2”只能单独触发，当“Jog1”和“Jog2”同时为0或1时，轴停止。