

EC3A-PS0200 产品使用手册

-- V1.0



目录

一、产品概述	3
1.1、产品简介	3
1.2、特点功能	3
1.3、应用场景	3
二、产品规格	4
2.1、命名规则与铭牌说明	4
2.2、部件说明	4
2.3、技术规格	4
2.4、端子接线	6
2.5、端子功能说明	7
2.6、指示灯说明	8
三、安装与拆卸	9
3.1、安装/拆卸注意事项	9
3.2、安装方向	9
3.3、最小间距	10
四、参数配置说明	11
4.1、配置前准备	11
4.2、对象字典	11
4.3、使用说明	17
五、TwinCAT 使用入门指导	23
5.1、添加模块	23
5.2、功能演示	26
5.3、配合 PLC 使用	32
关于我们	41

一、产品概述

1.1、产品简介

EC3A-PS0200 是一款双通道差分脉冲定位模块，是一款经济稳定、安装简易，适用性强的产品。该模块拥有两个脉冲定位通道、16 路输入通道以及 4 路数字输出通道。

1.2、特点功能

- 两个运动控制通道
- 可配置加减速的点动、相对定位功能
- 可灵活配置的回原点功能
- 控制信号形式为常用的 脉冲+方向 形式，5V 差分信号
- 每通道配备左限位、右限位、原点和刹车输入
- 可最高 500KHZ 的脉冲输出频率
- 电源电路采用防反接设计
- 广泛用于工业现场设备的运动控制

1.3、应用场景

EC3A-PS0200 模块可应用范围很广。运动控制广泛用于包装，印刷，纺织，半导体生产和装配业。运动控制包含与物体运动相关的每种技术。它涵盖了从微型系统（如硅型微感应执行器）到大型系统（如空间平台）的每个运动系统。但是，如今，运动控制的重点是具有电动执行器（如直流/交流伺服电机）的运动系统的特殊控制技术。机器人操纵器的控制也包括在运动控制领域中，因为大多数机器人操纵器由电伺服马达驱动，关键目标是控制运动。



二、产品规格

2.1、命名规则与铭牌说明



①产品信息 EC:Ether CAT	③信号类型 脉冲型	⑤脉冲输出类型 0: 差分输出
②系列号 3A:书本式模块	④通道数量 02: 2 路脉冲输出	⑥功能类型 0: 相对定位和速度控制

2.2、部件说明

名称	功能定义		
信号指示灯	IN	L/A (绿灯)	数据交互或异常
		ECAT (黄灯)	Ether CAT 状态指示灯
	OUT	L/A (绿灯)	数据交互或异常
		SYS (黄灯)	模块运行状态

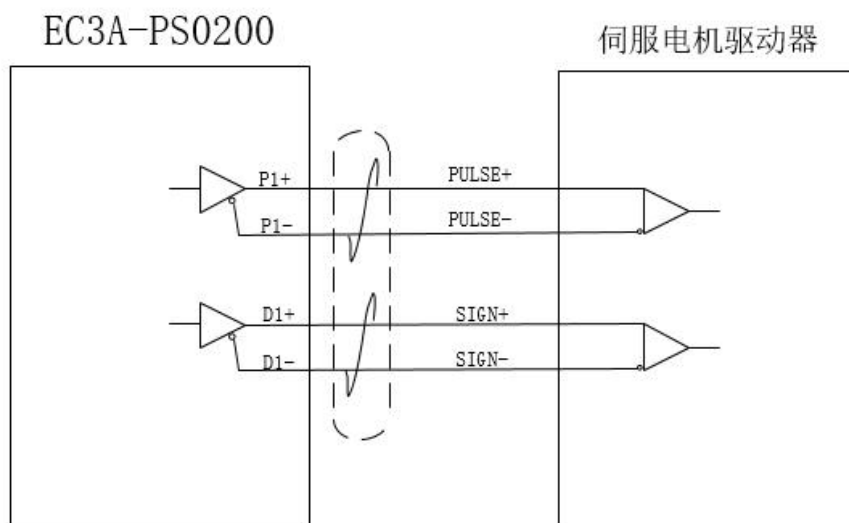
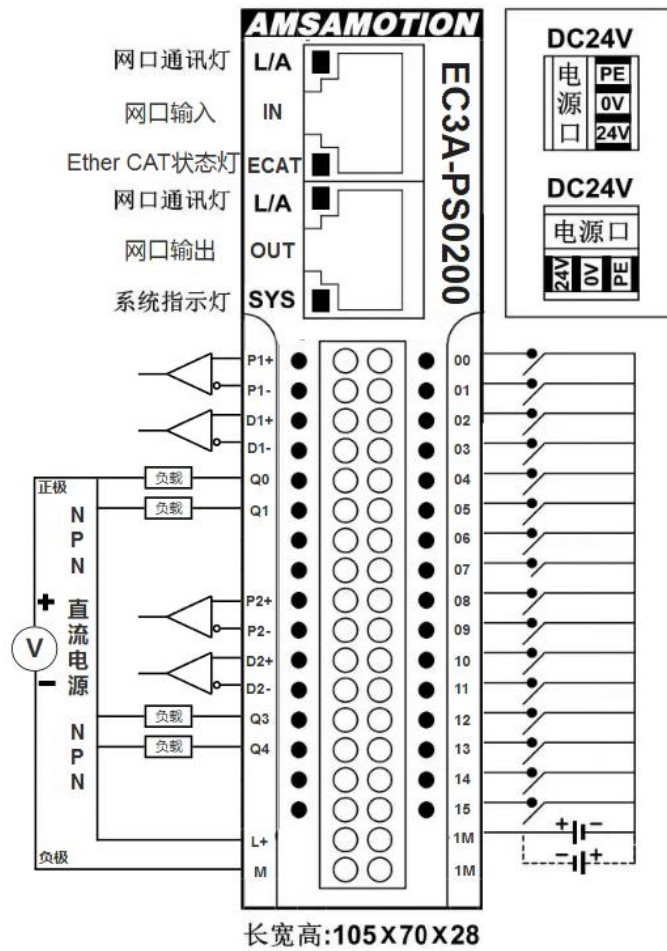
2.3、技术规格

总线参数	规格
总线协议	Ether CAT
I/O 站数量	根据主站



数据传输介质	Ethernet/Ether CAT ≥CAT6 电缆
传输距离	≤100m (站站距离)
传输速率	100Mbps
总线接口	2xRJ45
模块参数	规格
组态方式	通过主站
电源额定值 (范围)	24V DC (18~30V)
电气隔离	500V DC
尺寸	28x105x70mm (不含端子) /28x105x86(含端子)
工作温度	-20°C~+70°C
存储温度	-40°C~+85°C
相对湿度	95%，无冷凝
防护等级	IP20
脉冲+方向接口参数	规格
接口电平	5V 差分信号
脉冲频率范围	1Hz~500kHz
定位脉冲数量	0~4294967295
脉冲信号占空比	50%
数字量输入	规格
输出类型	NPN 型晶体管输出
输出能力	0.5A/点; 2A/4 点
绝缘回路	光耦隔离
数字量输出	规格
输出类型	NPN 型晶体管输出
输出能力	0.5A/点; 2A/4 点
绝缘回路	光耦隔离

2.4、端子接线



2.5、端子功能说明

端子标号	功能说明	端子标号	功能说明
P1+	轴 1 脉冲+ (5V 差分)	00	轴 1 左限位
P1-	轴 1 脉冲- (5V 差分)	01	轴 1 原点
D1+	轴 1 方向+ (5V 差分)	02	轴 1 右限位
D1-	轴 1 方向- (5V 差分)	03	轴 1 伺服准备好
Q0	轴 1 伺服使能 (数字量 NPN 输出)	04	轴 1 伺服报警
Q1	轴 1 报警清除 (数字量 NPN 输出)	05	轴 1 定位完成
NC	---	06	轴 1 机械制动器释放
NC	---	07	轴 1 急停
P2+	轴 2 脉冲+ (5V 差分)	08	轴 2 左限位
P2-	轴 2 脉冲- (5V 差分)	09	轴 2 原点
D2+	轴 2 方向+ (5V 差分)	10	轴 2 右限位
D2-	轴 2 方向- (5V 差分)	11	轴 2 伺服准备好
Q3	轴 2 伺服使能 (数字量 NPN 输出)	12	轴 2 伺服报警
Q4	轴 2 报警清除 (数字量 NPN 输出)	13	轴 2 定位完成
NC	---	14	轴 2 机械制动器释放
NC	---	15	轴 2 急停
L+	24V 输入正	1M	公共端
M	24V 输入负	1M	公共端

注意：左限位/原点/右限位/急停 这四个输入已经固定关联了功能无法当作普通 IO 使用，例如急停点位触发将停止脉冲输出。伺服准备好/伺服报警/定位完成/机械制动器释放 这四个输入没有关联功能只定义了名字，可当作普通 IO 使用不会影响脉冲的控制。

以上说明适用于轴 1 和轴 2

2.6、指示灯说明

功能	LED 状态
上电后，LED 灯初始状态	STA 灯 1000ms 闪烁
芯片校验错误	STA 灯双闪
INT	STA 灯 1000ms 闪烁
PRE-OP	ECAT 灯慢闪 STA 灯 1000ms 闪烁
SAFE-OP	ECAT 单闪 STA 灯 1000ms 闪烁
OP	ECAT 灯常亮 STA 灯 1000ms 闪烁
ERR	STA 灯常亮
升级模式功能	升级 LED 状态
升级模式初始化状态	ECAT 灯快闪 STA 灯 250ms 闪烁
文件传输完成，升级成功	ECAT 灯快闪 STA 灯 250ms 闪烁
传输文件头出现错误（文件后缀错误、大小错误）	ECAT 灯快闪 STA 灯 100ms 闪烁
文件传输过程中	ECAT 灯快闪 STA 灯：200ms 灭 200ms 亮 200ms 灭 200ms 亮 200ms 灭 2000ms 闪烁。
文件传输失败（包丢失，或者校验错误）	ECTA 灯快闪 STA 灯 1000ms 闪烁
升级模式跳转运行模式失败	ECTA 灯快闪 STA 灯常亮
硬件错误	STA 灯常亮

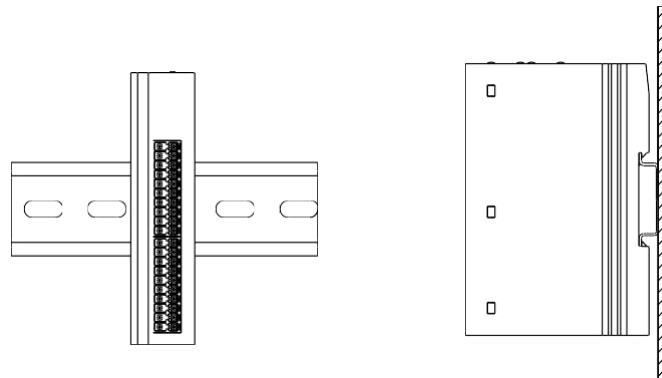
三、安装与拆卸

3.1、安装/拆卸注意事项

- 确保机柜有良好的通风措施（如机柜加装排风扇）。
- 请勿将本设备安装在可能引起过热的设备旁边或者上方。
- 务必将模块垂直安装，模块与周围设备之间确保有足够间距。
- 安装\拆卸务必在切断电源的状态下进行。

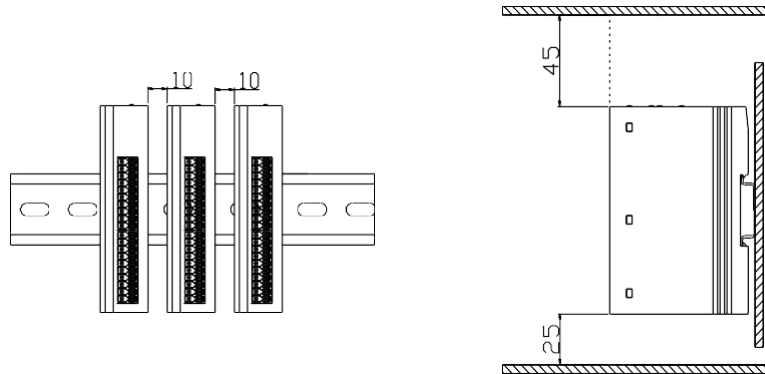
3.2、安装方向

为保持模块正常散热，务必将模块垂直安装，确保模块内部气流畅通。



3.3、最小间距

模块防护等级为 IP20，需箱内或柜内安装。安装时，模块与其他模块或者发热设备、模块上下与其他设备或接线槽，请按照下图所示的最小间距(单位：mm)。



四、参数配置说明

本章节针对 EC3A-PS0200 模块的对象字典进行详细介绍，本章只介绍和模块功能相关对象字，其他不作介绍。

4.1、配置前准备

将 DC 24V 外部电源接入模块并通电，通电前请检查电源正负极是否连接正确。

4.2、对象字典

索引	子索引	数据类型	名称	默认值	最小值	最大值	描述
//0x6nnx							
0x6000			Location Flags Bit				点位标志位
	1	BOOL	Axis1 Location Finish	0			轴 1 定位完成标志位，此位每次运动开始会自动清零，或通过运动完成清除位清零。0：未完成；1：定位完成；
	2	BOOL	Axis1 Home Finish	0			轴 1 回零完成标志，此位每次回零开始会自动清零，或通过运动完成清除位清零。0：未完成；1：定位完成；
	3	BOOL	Axis2 Location Finish	0			轴 2 定位完成标志位，此位每次运动开始会自动清零，或通过运动完成清除位清零。0：未完成；1：定位完成；
	4	BOOL	Axis2 Home Finish	0			轴 2 回零完成标志，此位每次回零开始会自动清零，或通过运动完成清除位清零。。0：未完成；1：定位完成；
0x6001			Position Feedback				轴位置反馈
	1	INT32	Axis1 Current Position Pulse	0			轴 1 当前位置。单位：脉冲；
	2	REAL	Axis1 Current Position	0			轴 1 当前位置。实际距离（脉冲数×实际距离单位 8000:17）
	4	INT32	Axis2 Current Position Pulse	0			轴 2 当前位置。单位：脉冲；
	5	REAL	Axis2 Current Position	0			轴 1 当前位置。实际距离（脉冲数×实际距


						离单位 8000:17)
0x6002			Feedback	0		轴状态和错误反馈
	1	BYTE	Axis1 Status	0		轴 1 状态机状态 (详见状态机说明)。
	2	BYTE	Axis1 Error Code	0		轴 1 错误码 (详见错误码说明)。
	3	BYTE	Axis2 Status	0		轴 1 状态机状态 (详见状态机说明)。
	4	BYTE	Axis2 Error Code	0		轴 1 错误码 (详见错误码说明)。
0x6003			Digital Input			数字量输入
	1	BOOL	Axis1 Left	0		轴 1 左限位。
	2	BOOL	Axis1 Home	0		轴 1 原点有效位。
	3	BOOL	Axis1 Right	0		轴 1 右限位。
	4	BOOL	Axis1 Servo Ready	0		轴 1 伺服准备好 (不绑定软件功能,可当作普通 IO 使用)。
	5	BOOL	Axis1 Servo Alarm	0		轴 1 伺服报警 (不绑定软件功能,可当作普通 IO 使用)。
	6	BOOL	Axis1 Servo Location Finish	0		轴 1 伺服定位完成 (不绑定软件功能,可当作普通 IO 使用)。
	7	BOOL	Axis1 Mechanical Brake Release	0		轴 1 机械制动器释放 (不绑定软件功能,可当作普通 IO 使用)。
	8	BOOL	Axis1 Hard Stop	0		轴 1 硬件急停。
	9	BOOL	Axis2 Left	0		轴 2 左限位。
	10	BOOL	Axis2 Home	0		轴 2 原点有效位。
	11	BOOL	Axis2 Right	0		轴 2 右限位。
	12	BOOL	Axis2 Servo Ready	0		轴 2 伺服准备好 (不绑定软件功能,可当作普通 IO 使用)。
	13	BOOL	Axis2 Servo Alarm	0		轴 2 伺服报警 (不绑定软件功能,可当作普通 IO 使用)。
	14	BOOL	Axis2 Servo Location Finish	0		轴 2 伺服定位完成 (不绑定软件功能,可当作普通 IO 使用)。
	15	BOOL	Axis2 Mechanical Brake Release	0		轴 2 机械制动器释放 (不绑定软件功能,可当作普通 IO 使用)。
	16	BOOL	Axis2 Hard Stop	0		轴 2 硬件急停。

//0x7nnx						
0x7000			Location Control Bit	0		定位控制位
	1	BOOL	Axis1 Power	0		轴 1 使能位。0: 轴失能; 1: 轴使能;
	2	BOOL	Axis1 Power Enable	0		轴 1 使能控制位。此位 0->1 时轴的使能位会写入轴。
	3	BOOL	Axis1 Direction	0		轴 1 方向位。0: 负方向; 1: 正方向;
	4	UINT16	Axis1 Motion Mode	0		轴 1 运动模式选择。0: 相对定位模式; 1: 速度模式; 2: 点动模式;
	5	BOOL	Axis1 Motion Start	0		轴 1 运动使能位。0->1 触发运动 (点动模式为高电平触发, 低电平停止);
	6	BOOL	Axis1 Motion Halt	0		轴 1 终止运动位 (带减速终止)。0->1 触发终止运动;
	7	BOOL	Axis1 Home Enable	0		轴 1 回零使能位。0->1 触发回零;
	8	BOOL	Axis1 Emergency Stop Enable	0		轴 1 软件急停位 (急停后再次运动需要先复位轴状态)。0: 不触发急停; 1: 触发急停;
	9	BOOL	Axis1 Finish Clear	0		轴 1 定位完成清除位。0->1 清除定位完成标志;
	10	BOOL	Axis1 Home Finish Clear	0		轴 1 回零完成清除位。0->1 清除回零完成标志;
	11	BOOL	Axis1 Position Clear	0		轴 1 位置清除位, 当回零完成后会自动执行一次位置清零。0->1 清除当前位置;
	12	BOOL	Axis1 Motion Reset	0		轴 1 轴状态复位位。0->1 复位轴状态;
	13	BOOL	Axis2 Power	0		轴 2 使能位。0: 轴失能; 1: 轴使能;
	14	BOOL	Axis2 Power Enable	0		轴 2 使能控制位。此位 0->1 时轴的使能位会写入轴。
	15	BOOL	Axis2 Direction	0		轴 2 方向位。0: 负方向; 1: 正方向;
	16	UINT16	Axis2 Motion Mode	0		轴 2 运动模式选择。0: 相对定位模式; 1: 速度模式; 2: 点动模式;
	17	BOOL	Axis2 Motion Start	0		轴 2 运动使能位。0->1 触发运动 (点动模式为高电平触发, 低电平停止);
	18	BOOL	Axis2 Motion Halt	0		轴 2 终止运动位 (带减速终止)。0->1 触发终止运动;
	19	BOOL	Axis2 Home Enable	0		轴 2 回零使能位。0->1 触发回零;
	20	BOOL	Axis2 Emergency Stop	0		轴 2 软件急停位 (急停后再次运动需要先复位

			Enable				轴状态)。0: 不触发急停; 1: 触发急停;
	21	BOOL	Axis2 Finish Clear	0			轴 2 定位完成清除位。0->1 清除定位完成标志;
	22	BOOL	Axis2 Home Finish Clear	0			轴 2 回零完成清除位。0->1 清除回零完成标志;
	23	BOOL	Axis2 Position Clear	0			轴 2 位置清除位, 当回零完成后会自动执行一次位置清零。0->1 清除当前位置;
	24	BOOL	Axis2 Motion Reset	0			轴 2 轴状态复位位。0->1 复位轴状态;
0x7001			Location Parameter Set				定位参数设置
	1	UINT32	Axis1 Start Frequency	0	1	500000	轴 1 起始频率。单位: HZ;
	2	UINT32	Axis1 Max Frequency	0	1	500000	轴 1 最大频率 (即匀速时的频率)。单位: HZ;
	3	UINT32	Axis1 Number Of Pulse	0			轴 1 定位脉冲数。单位: 个;
	4	UINT16	Axis1 Ramp Time	0			轴 1 加减速时间。单位: ms;
	5	UINT32	Axis2 Start Frequency	0	1	500000	轴 2 起始频率。单位: HZ;
	6	UINT32	Axis2 Max Frequency	0	1	500000	轴 2 最大频率 (即匀速时的频率)。单位: HZ;
	7	UINT32	Axis2 Number Of Pulse	0			轴 2 定位脉冲数。单位: 个;
	8	UINT16	Axis2 Ramp Time	0			轴 2 加减速时间。单位: ms;
0x7002			Digital Output				数字量输出
	1	BOOL	Axis1 Servo Enable	0			轴 1 伺服使能 (不绑定软件功能,可当作普通 IO 使用)。
	2	BOOL	Axis1 Servo Alarm Clear	0			轴 1 伺服报警清除 (不绑定软件功能,可当作普通 IO 使用)。
	3	BOOL	Axis2 Servo Enable	0			轴 2 伺服使能 (不绑定软件功能,可当作普通 IO 使用)。
	4	BOOL	Axis2 Servo Alarm Clear	0			轴 2 伺服报警清除 (不绑定软件功能,可当作普通 IO 使用)。
//0x8nnx							
0x8000			Configuration Parameter				配置参数
	1	BOOL	Axis1 Forward Level	1			轴 1 方向位电平。0: 低电平代表正方向; 1:

							高电平代表正方向；
	2	BOOL	Axis1 Left Level	1			轴 1 左限位有效电平。0: 低电平有效；1: 高电平有效；
	3	BOOL	Axis1 Home Level	1			轴 1 原点有效电平。0: 低电平有效；1: 高电平有效；
	4	BOOL	Axis1 Right Level	1			轴 1 右限位有效电平。0: 低电平有效；1: 高电平有效；
	5	BOOL	Axis1 Hard Stop Level	1			轴 1 硬件急停有效电平。0: 低电平有效；1: 高电平有效；
	6	BOOL	Axis1 Home Direction	0			轴 1 回零方向。0: 从负方向查找限位开关；1: 从正方向查找限位开关
	7	BOOL	Axis1 Home Reference	0			轴 1 原点参考方向。0: 选择原点开关左边为原点；1: 选择原点开关右边为原点；
	8	BOOL	Axis1 Curve	0			轴 1 加减速曲线选择。0: 加减速曲线为梯形；1: 加减速曲线为双曲线；
	9	BOOL	Axis2 Forward Level	1			轴 2 方向位电平。0: 低电平代表正方向；1: 高电平代表正方向；
	10	BOOL	Axis2 Left Level	1			轴 2 左限位有效电平。0: 低电平有效；1: 高电平有效；
	11	BOOL	Axis2 Home Level	1			轴 2 原点有效电平。0: 低电平有效；1: 高电平有效；
	12	BOOL	Axis2 Right Level	1			轴 2 右限位有效电平。0: 低电平有效；1: 高电平有效；
	13	BOOL	Axis2 Hard Stop Level	1			轴 2 硬件急停有效电平。0: 低电平有效；1: 高电平有效；
	14	BOOL	Axis2 Home Direction	0			轴 2 回零方向。0: 从负方向查找限位开关；1: 从正方向查找限位开关
	15	BOOL	Axis2 Home Reference	0			轴 2 原点参考方向。0: 选择原点开关左边为原点；1: 选择原点开关右边为原点；
	16	BOOL	Axis2 Curve	0			轴 2 加减速曲线选择。0: 加减速曲线为梯形；1: 加减速曲线为双曲线；
	17	REAL	Axis1 Pulse Unit	0.0			轴 1 每脉冲代表的实际距离。
	18	UINT32	Axis1 Home Frequency	1000	1	500000	轴 1 回零速度（以此速度查找限位开关）。
	19	UINT32	Axis1 Home Approach	100	0	500000	轴 1 回零逼近速度（以此速度逐次逼近原点，

			Frequency				此速度必须小于等于回零速度)。
	20	INT32	Axis1 Home Offset	0			轴 1 原点偏移量。单位：脉冲个数；
	21	REAL	Axis2 Pulse Unit	0.0			轴 2 每脉冲代表的实际距离。
	22	UINT32	Axis2 Home Frequency	1000	1	500000	轴 2 回零速度 (以此速度查找限位开关)。
	23	UINT32	Axis2 Home Approach Frequency	100	1	500000	轴 2 回零逼近速度 (以此速度逐次逼近原点, 此速度必须小于等于回零速度)。
	24	INT32	Axis2 Home Offset	0			轴 2 原点偏移量。单位：脉冲个数；
	25	BYTE	Filtering Time	1	1	10	数字量输入滤波时间。

 **Note:** Axis(x) Servo Ready、Axis(x) Servo Alarm、Axis(x) Servo Location Finish、Axis(x) Mechanical Brake Release、Axis(x) Servo Enable、Axis(x) Servo Alarm Clear 等均未绑定定位功能, 只做普通数字输入或输出功能。所有 8000 段的数据都只能在 preop 下修改!!!

4.3、轴状态码和轴错误码

轴状态	
状态码	含义
0	轴未准备好
1	轴准备好
2	轴处于点动模式
3	轴处于速度模式
4	轴处于相对定位模式
5	轴处于回零模式
6	轴发生错误停止

轴错误	
错误码	含义
0	无错误
1	Power 失败, 运动过程中不能执行 Power 指令
2	频率超出范围
3	回零频率超出范围
4	频率设置有误 (最大频率无效或者最大频率小于最小频率)
5	回零最大频率小于回零逼近频率
6	脉冲个数无效
7	多个运动同时触发
8	定位中碰到左右限位
9	急停生效
10	限位信号错误, 多个限位信号同时生效
11	回零中遇到的限位顺序错误

其中 1-3 为警告, 产生警告不影响定位的执行。4-7 为非致命错误, 产生错误定位不会执行。

8-11 为致命错误, 产生致命错误会在定位执行过程中停止。发生致命错误时必须通过轴复位才能执行下一次定位。

4.4、使用说明

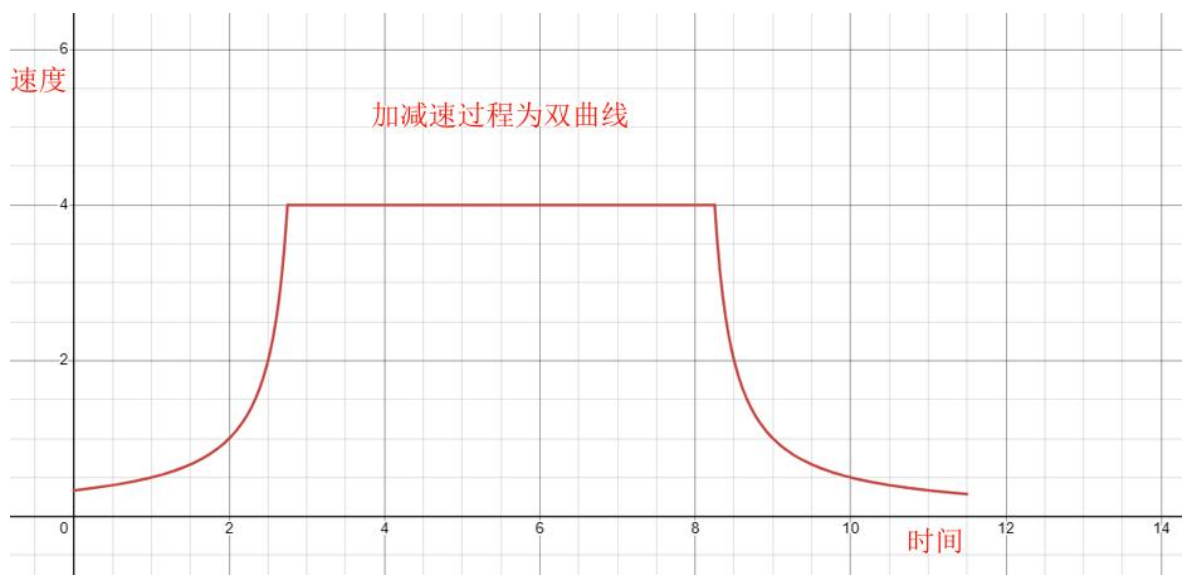
方向选择

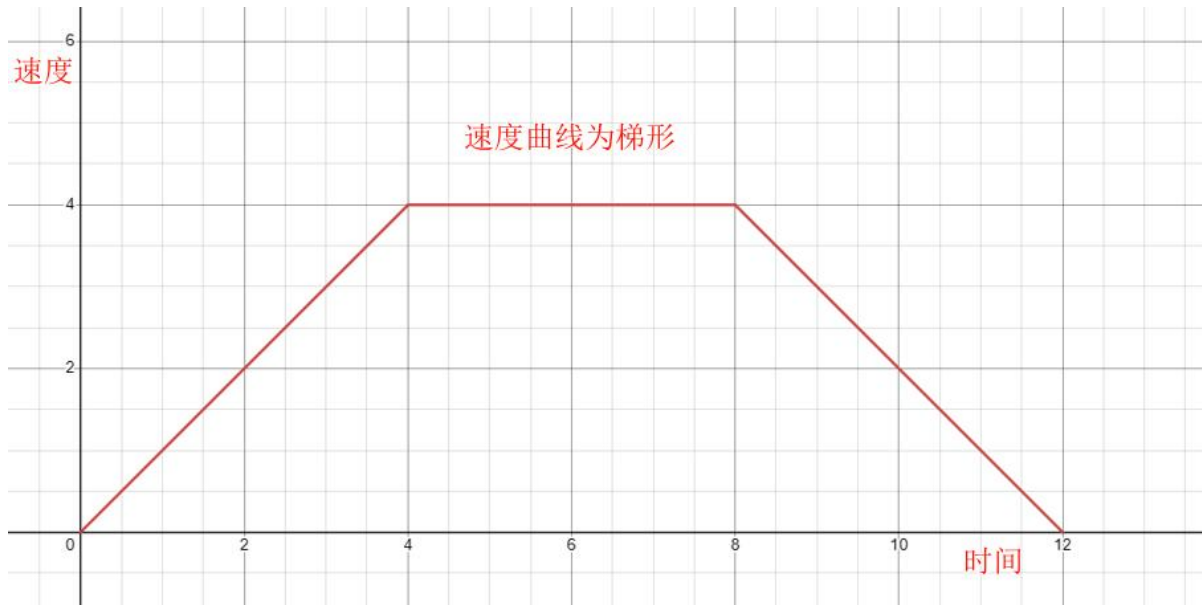
Axis(x) Direction 代表了轴 x 逻辑上的运动方向，1 表示正向运动，0 表示负向运动；正向运动时 Axis(x) Current Position Pulse 和 Axis(x) Current Position 会增加，反之则会减少；

Axis(x) Forward Level 表示正向运动所对应的电平，默认为 1；即，此位置 1 则正向运动时方向输出为高电平，此位置 0 则正向运动时方向输出为低电平；

速度曲线选择

Axis(x) Curve 为配置参数，只有在 safeop 状态下才可修改；其值为 1 表示加减速过程为双曲线，为 1 表示速度曲线为梯形；





定位功能使用举例

相对定位步骤如下：

- 1, 将 Axis(x) Motion Start 设置为 0, 为 Axis(x) Motion Start 的上升沿做准备；另外将 Axis(x) Finish Clear 设置为 1, 清除 Axis(x) Location Finish 标志位；
- 2, 等待 Axis(x) Location Finish 置 0；
- 3, Axis(x) Location Finish 置 0 后, 将 Axis(x) Direction 设置为 1 表示正向运动；
- 4, 将 Axis(x) Motion Mode 设置为 1 表示模式选择为相对定位模式；
- 5, 将 Axis(x) Start Frequency 设置为 1 表示起始频率为 1Hz, 频率代表速度, 具体代表多少速度与电机一圈的脉冲数, 一个脉冲代表的距离有关；
- 6, 将 Axis(x) Max Frequency 设置为 100000 表示匀速时频率为 100000Hz；
- 7, 将 Axis(x) Number Of Pulse 设置为 10000000 表示发送 10000000 个脉冲, 乘以单个脉冲所代表的距离即可得出所走的位移；
- 8, 将 Axis(x) Ramp Time 设置为 1000 表示加减速时间为 1000ms 即 1s 钟；
- 9, 将 Axis(x) Motion Start 设置为 1, 产生上升沿触发运动；另外将 Axis(x) Finish Clear

设置为 0，为下一次清除标志位做上升沿准备；

10，查看电机经历 1s 钟加速、匀速、1s 钟减速总计 10000000 个脉冲后停止；

11，查看 Axis(x) Location Finish 标志位，当 Axis(x) Location Finish 值变为 1 时，说明 10000000 个脉冲发送完毕；另外也可以查看 Axis(x) Current Position Pulse 的值看到实时位置的变化；

注意：Axis(x) Motion Start 和 Axis(x) Finish Clear 必须要上升沿有效，所以在用之前必须要先置为 0；另外如果不以模块自身的 Axis(x) Location Finish 作为定位完成的标志则有关 Axis(x) Location Finish 的操作可以省略，否则在运动之前必须要先将 Axis(x) Location Finish 清零，以确保读到的 Axis(x) Location Finish 标志是代表本次运动的标志；

点动的用法与相对定位类似，唯一的区别是点动模式下模块将不关心 Axis(x) Number Of Pulse 的值，在加速到最大速度后一直运动下去；

不管是在相对定位还是点动模式下，只要 Axis(x) Motion Start 产生了下降沿，运动都会按照设定好的加减速时间减速停止；

Axis(x) Emergency Stop Enable 对于相对定位和点动来说都是立马停止，没有任何减速过程，且 Axis(x) Emergency Stop Enable 是高电平有效，在 Axis(x) Emergency Stop Enable 为 1 的时候是不能够进行任何运动的；

回原点功能

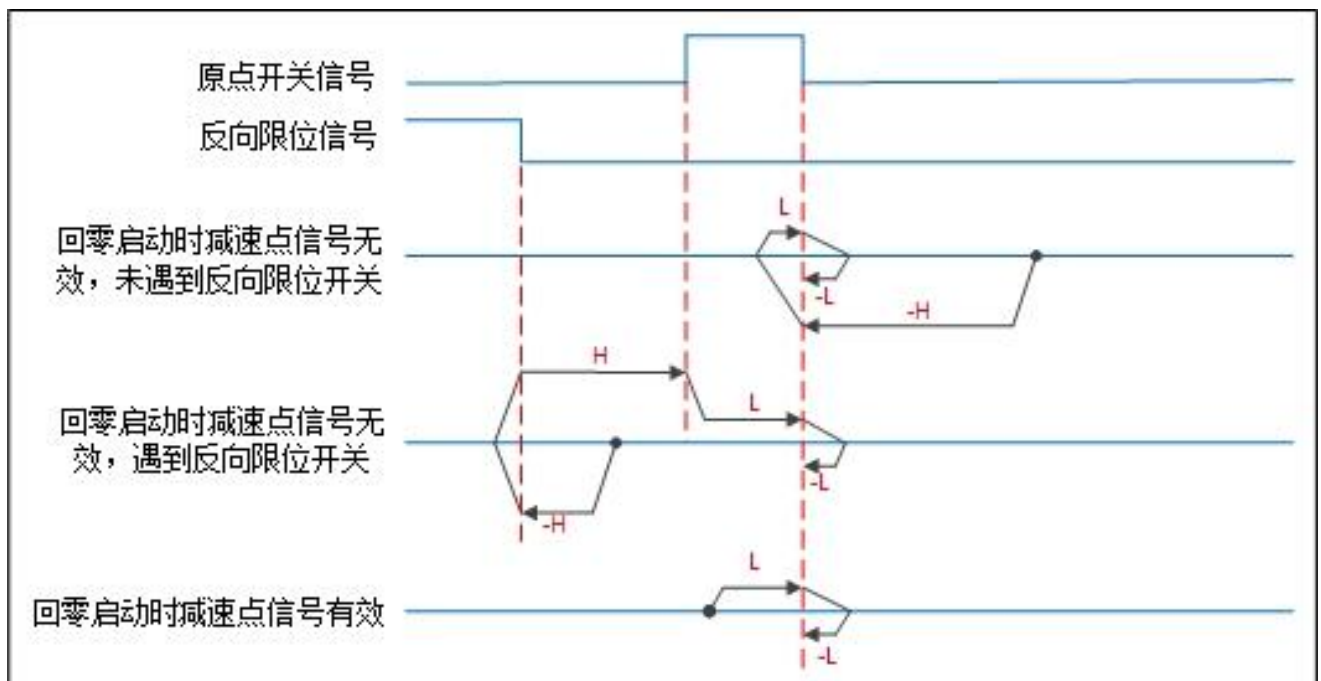
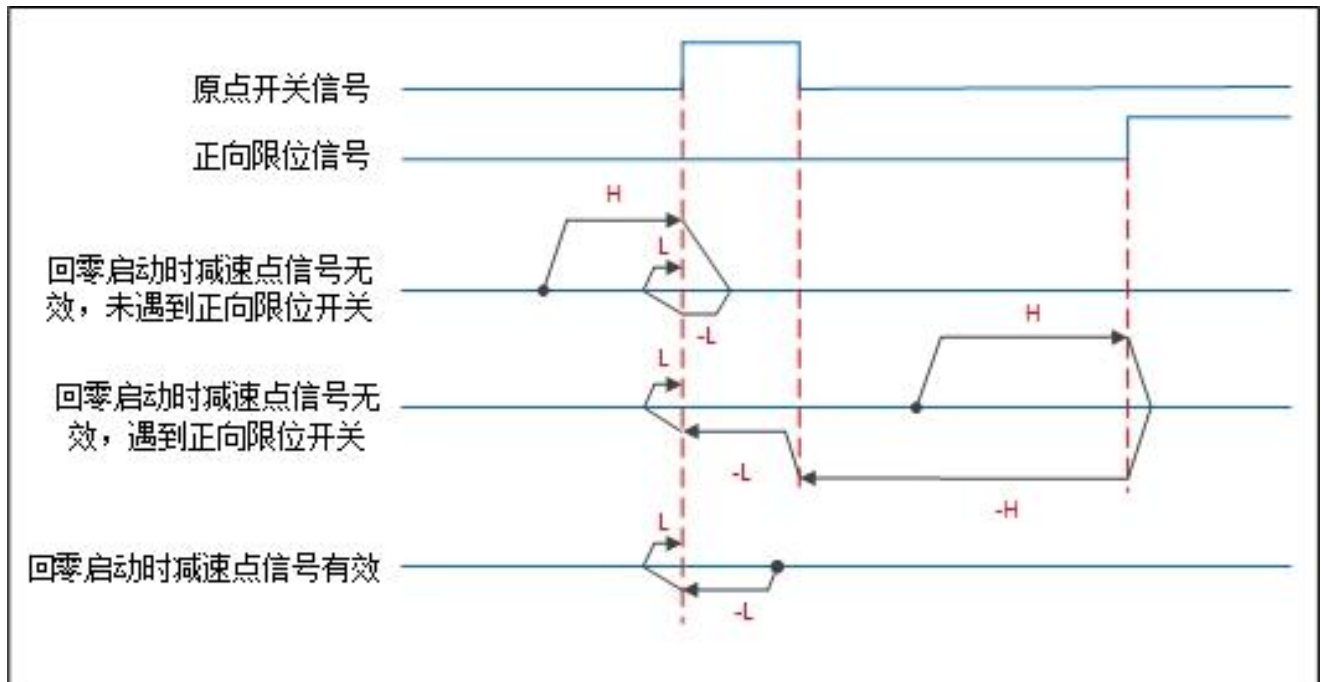
回原点的参数都在配置参数中配置，配置参数全部只能在 safeop 下写入，所以建议将所需要的回零参数加入启动项来方便回零；

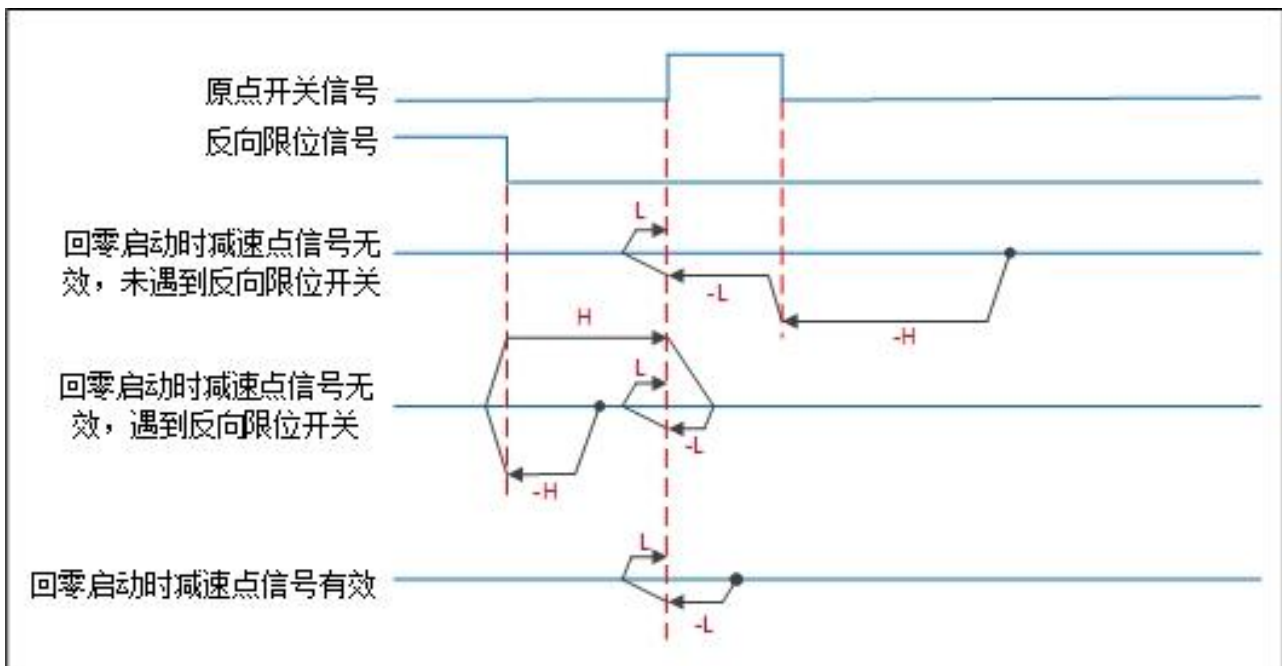
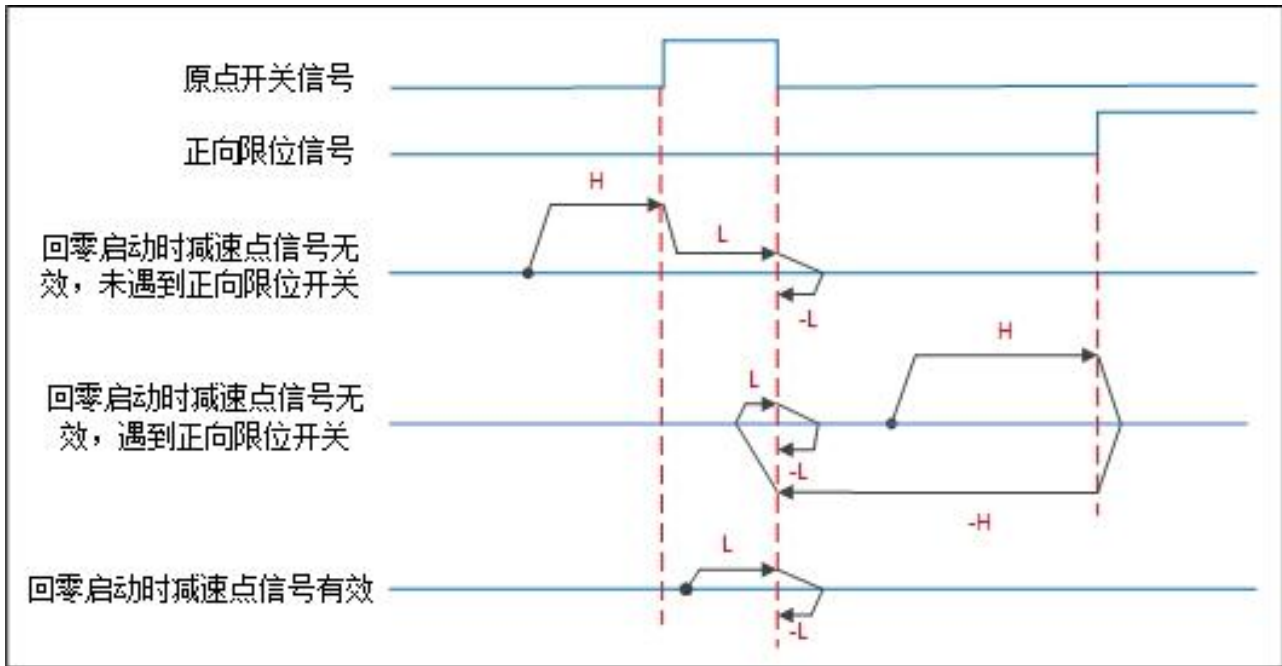
回原点相关参数详解：Axis(x) Left Level 表示左限位的有效电平，1 代表高电平有效，0 代表低电平有效，以方便用户对应不同的情况，Axis(x) Home Level 和 Axis(x) Right Level 的含义也是如此，它们的默认值都是 1；Axis(x) Home Direction 表示回原点方向，回原点启动时向此方向运动寻找限位开关，1 表示正向寻找限位开关，0 表示负向寻找限位开关，默认值为 0，如果寻找到左右限位则会停止、反向运动寻找原点开关，如果启动是原点开关已经有效则会直接以低速逼近原点位置；Axis(x) Home Reference 表示原点参考侧，1 表示以原点开关的右侧（正方向侧）为原点，0 表示以原点开关的左侧为原点（负方向侧）；Axis(x) Home Frequency 表示回原点速度，具体含义是寻找原点开关或限位开关时的速度；Axis(x) Home Approach Frequency 表示回原点逼近速度，在寻找到原点开关之后会以一个比较低的速度逼近原点参考侧，所以这个值必须要小于 Axis(x) Home Frequency 的值，这个值越小回原点会越精确但速度会越慢；Axis(x) Home Offset 表示原点位置偏移，电机在最终停到原点开关的左侧或右侧后会根据此值以 Axis(x) Home Approach Frequency 速度进行一段位移停在用户想要的位置；

先确保 Axis(x) Home Enable 置 0，根据以上参数的配置，再将 Axis(x) Home Enable 置 1 产生上升沿触发回原点；然后等待 Axis(x) Home Finish 置 1，表示回原点完成；如果要多次回原点则要想定位一样在回原点之前用 Axis(x) Home Finish Clear 将 Axis(x) Home Finish 标志清零，以确保最终判断的回原点完成标志是本次会原点产生的；

回原点成功后模块会自动将当前位置值清零；

回原时序图

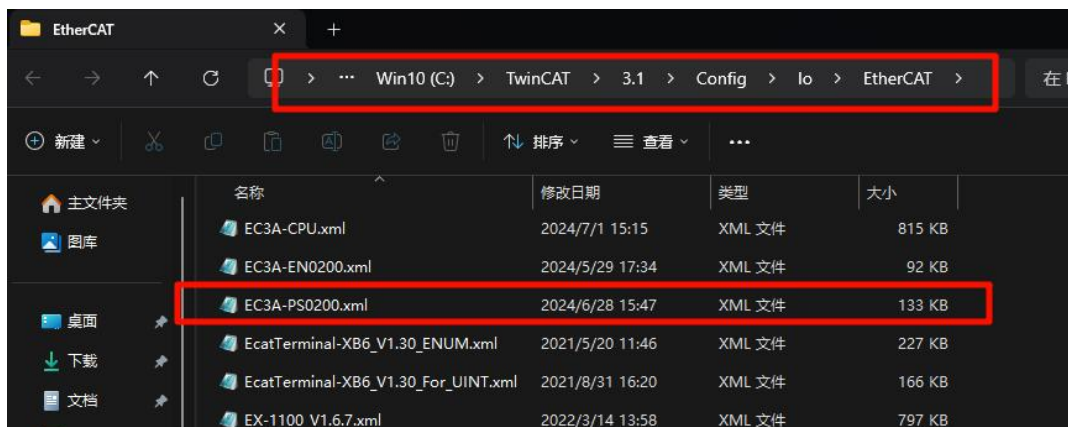




五、TwinCAT 使用入门指导

5.1、添加模块

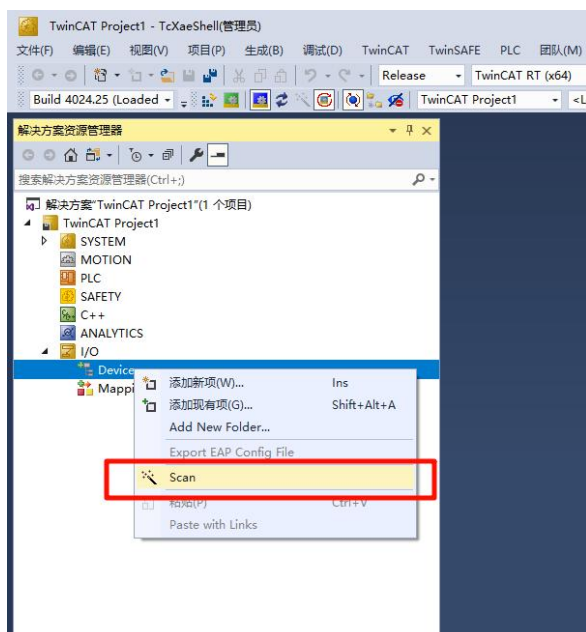
1, 将 EtherCAT 所需 XML 拷贝至主站安装文件夹下的模块 xml 文件夹下, 以 TwinCAT 3 为例, 如下图:



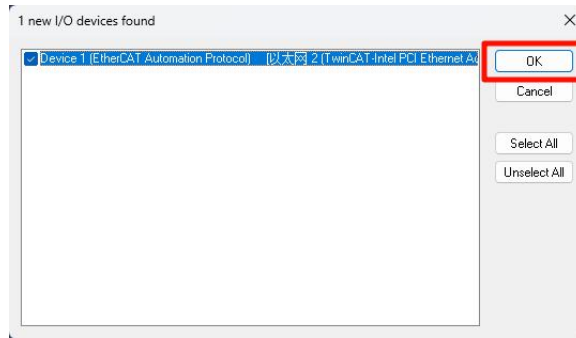
2, 右击桌面右下角 TwinCAT 图标, 选择 “TwinCAT XAE(xx..)”, 打开 TwinCAT 软件, 如下图所示:



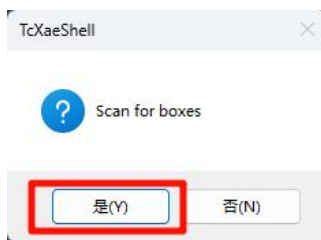
3, 打开 TwinCAT 软件后, 新建工程或打开已有工程, 在 “I/O -> Devices” 处右击选择, 然后点击 “Scan”



勾选当前扫描到的“本地连接”网卡，点击“OK”



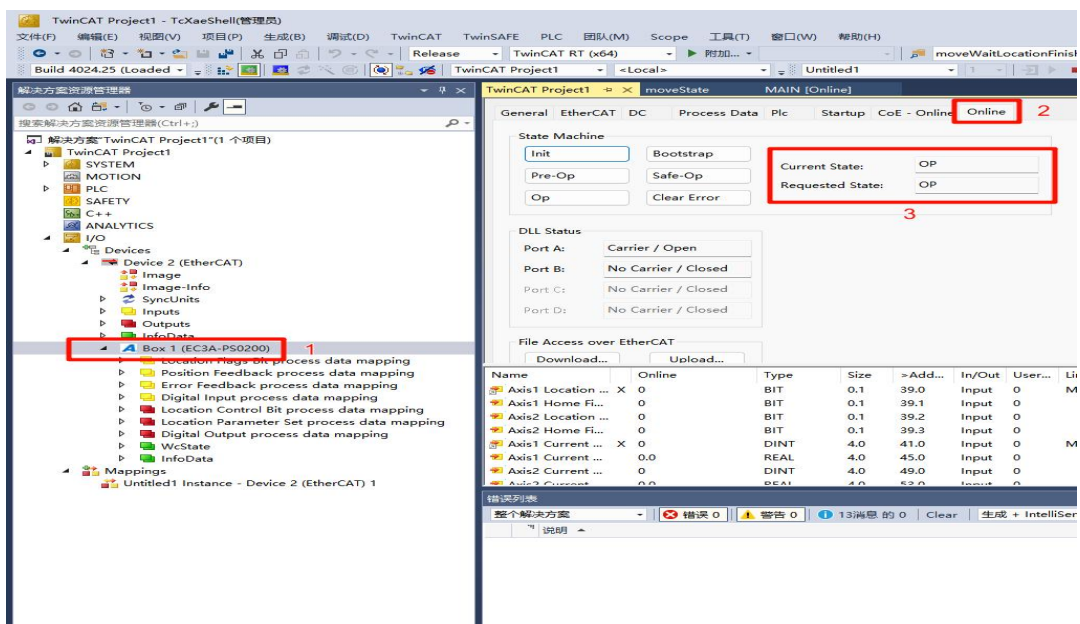
弹出窗口“Scan for boxes”选择“是”



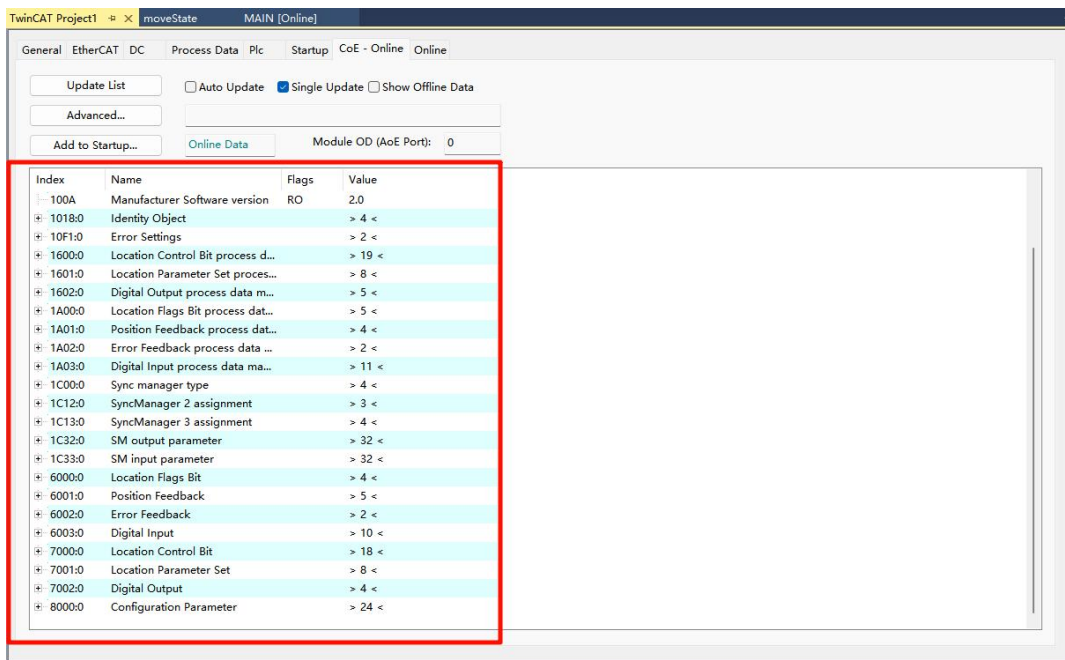
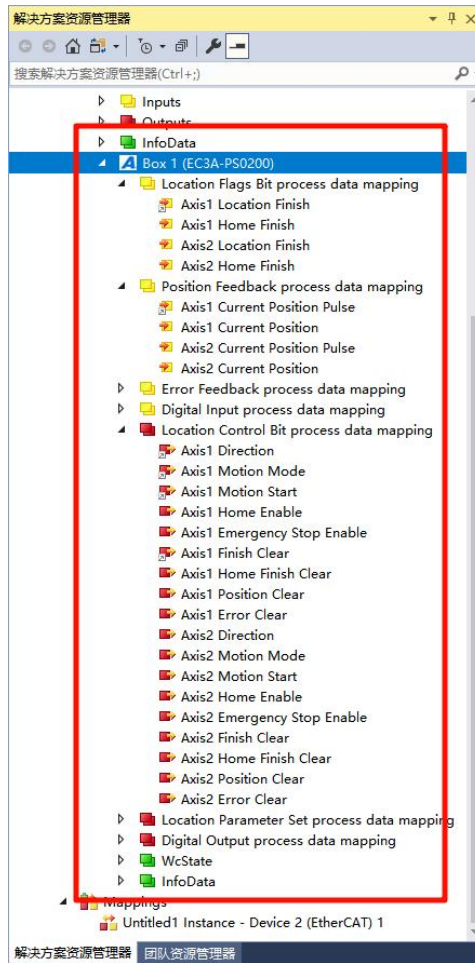
弹出窗口“Activate Free Run”选择“是”



4, 查看已扫到 EC3A-PS0200 从站, 单击 Box1, 进入 Online 选项, 查看当前状态应当为“OP”



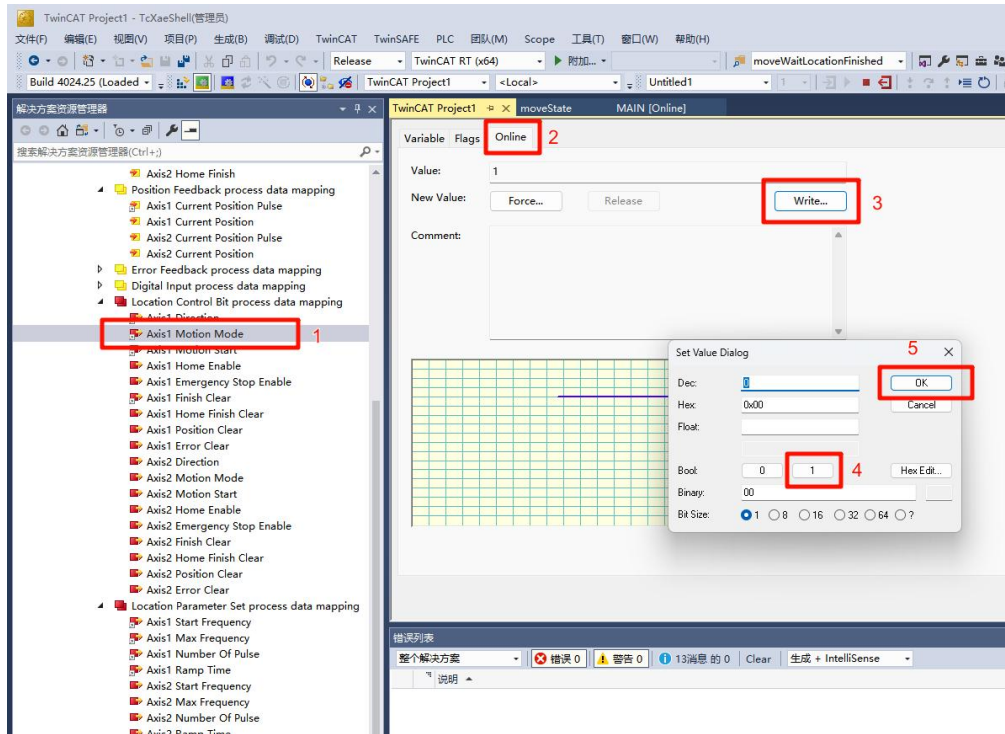
展开 Box, 查看过程对象; 或者单击 CoE-Online, 下拉可以见到本模块所有对象字典:



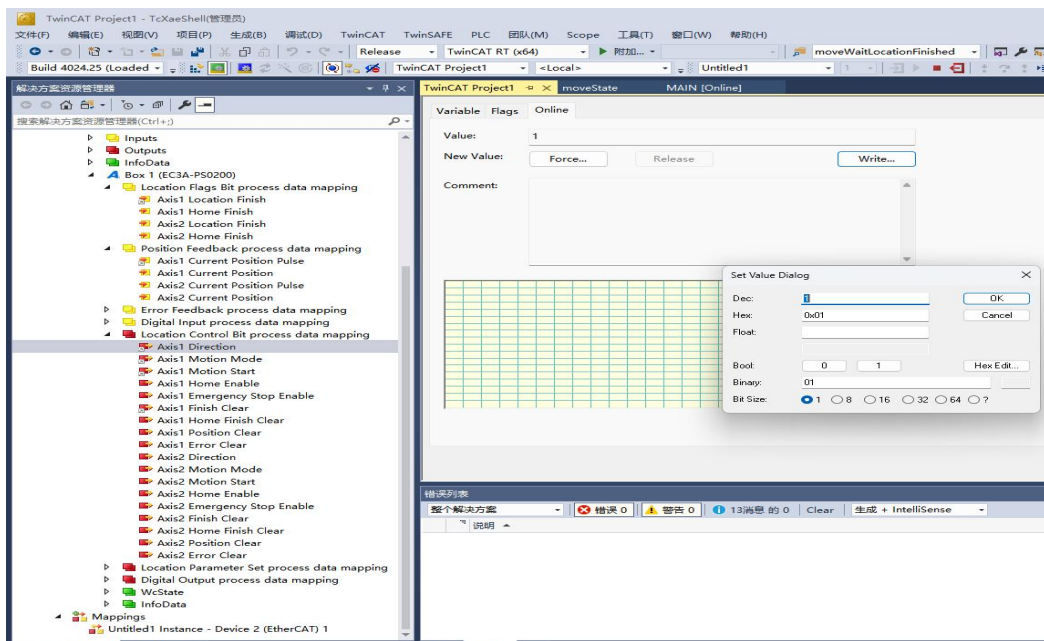
5.2、功能演示

点动功能：

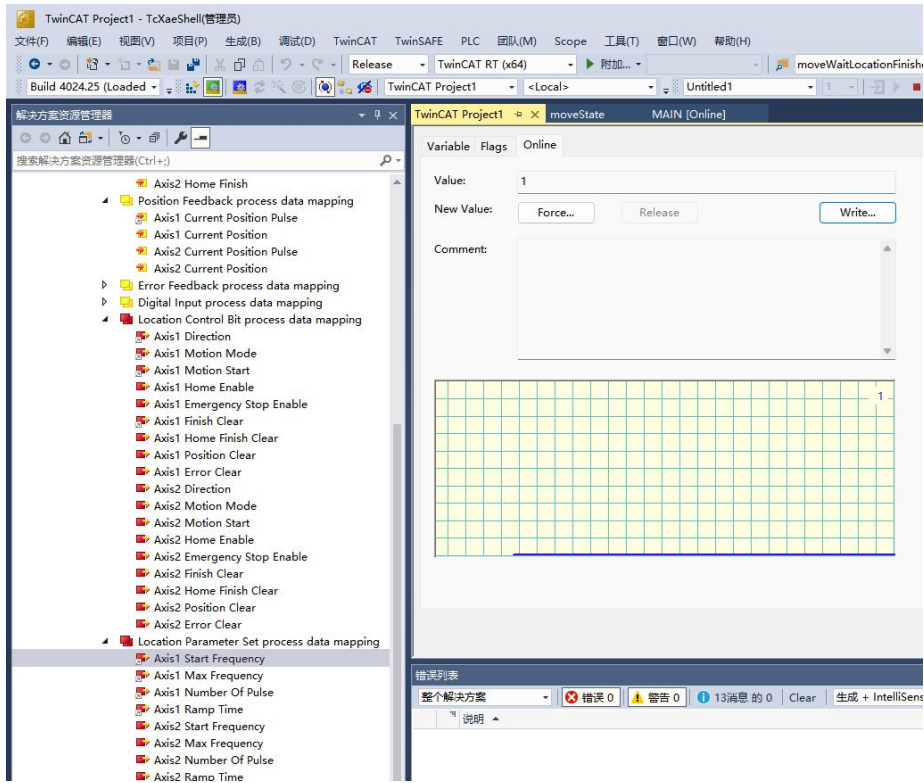
确保模块在 op 状态。如图在展开的对象字典中双击 Axis1 Motion Mode，在右侧界面点击 Online，点击 Write，然后点击 0 写入值 0，或者输入值再点击 OK 写入值。



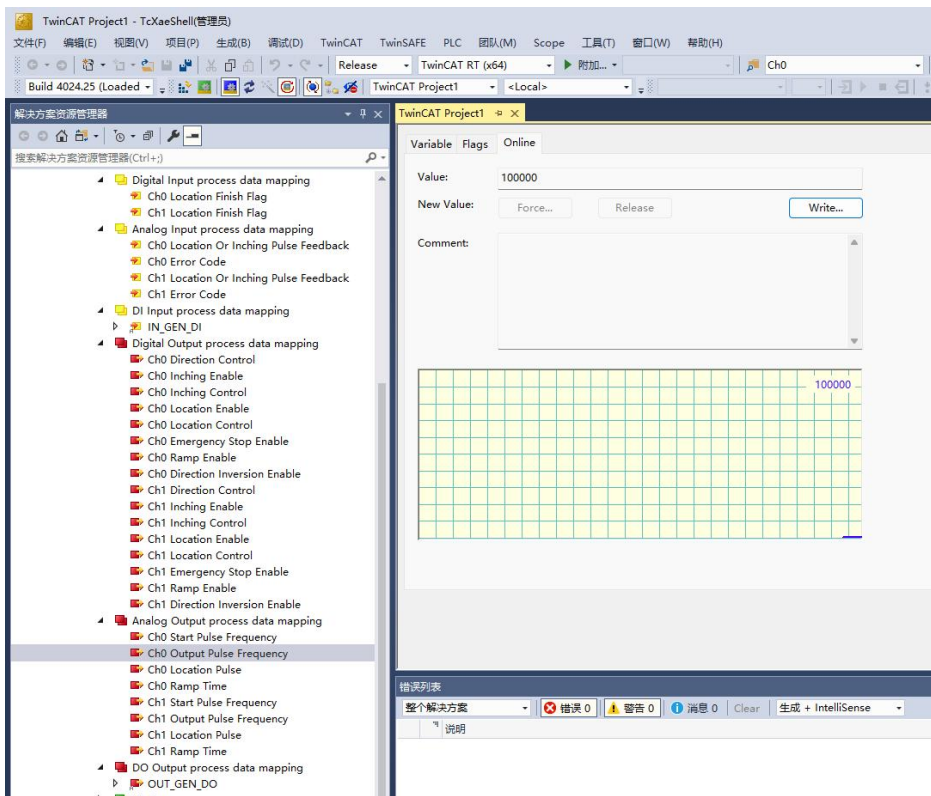
同样方法，选择 Axis1 Direction，写入 1 表示电机正转



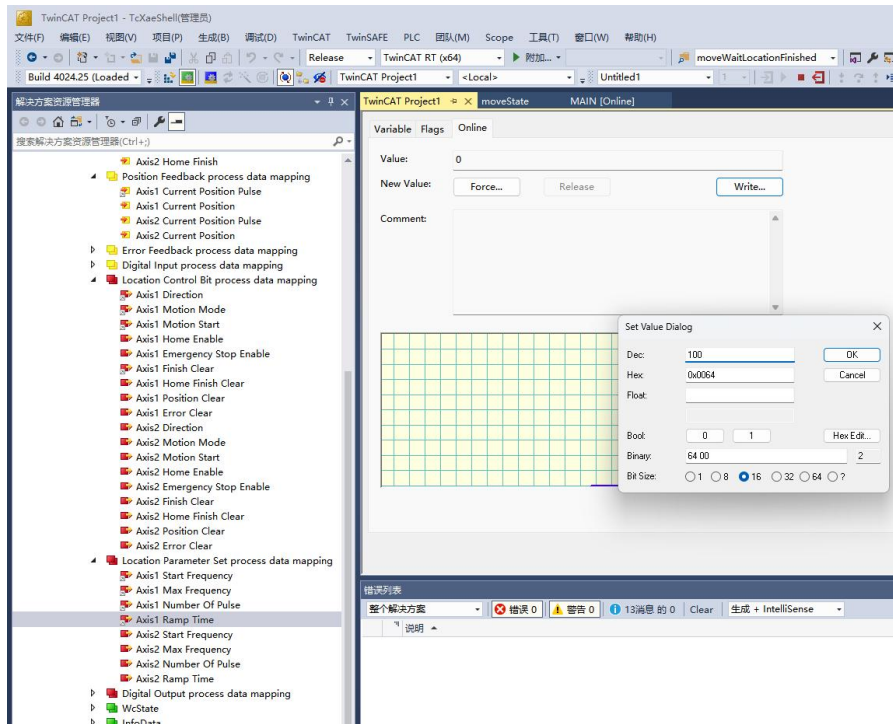
选择 Axis1 Start Frequency，写入 1，初始速度设置为 1



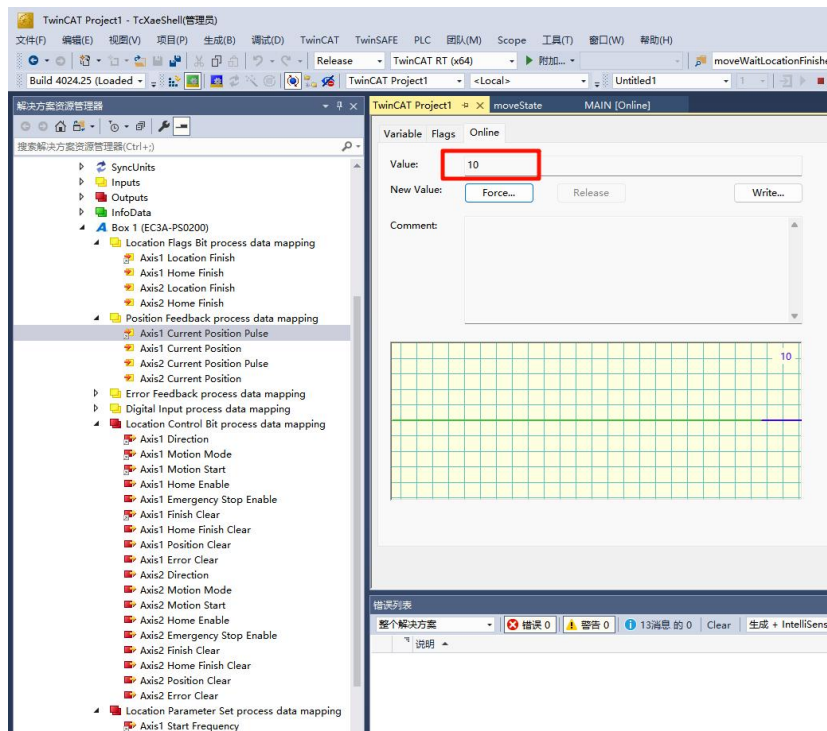
选择 Ch0 Output Pulse Frequency，写入 100000，最终速度设置为 100000



选择 Axis1 Ramp Time, 写入 100, 斜坡时间设置为 100ms



选择 Axis1 Motion Start, 写入 1, 使能点动, 点动开始。在 Axis1 Current Position Pulse 可以查看到当前运动到的位置。

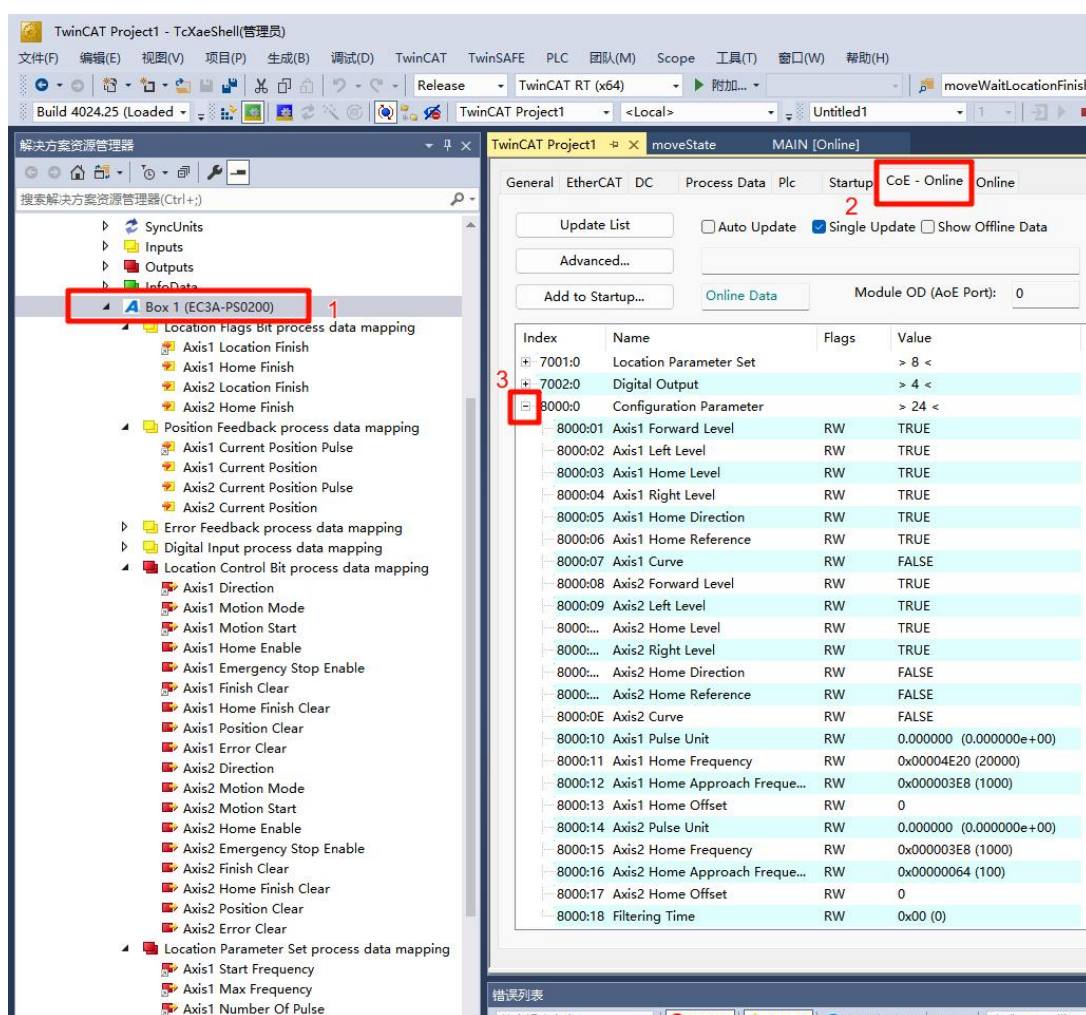


定位功能:

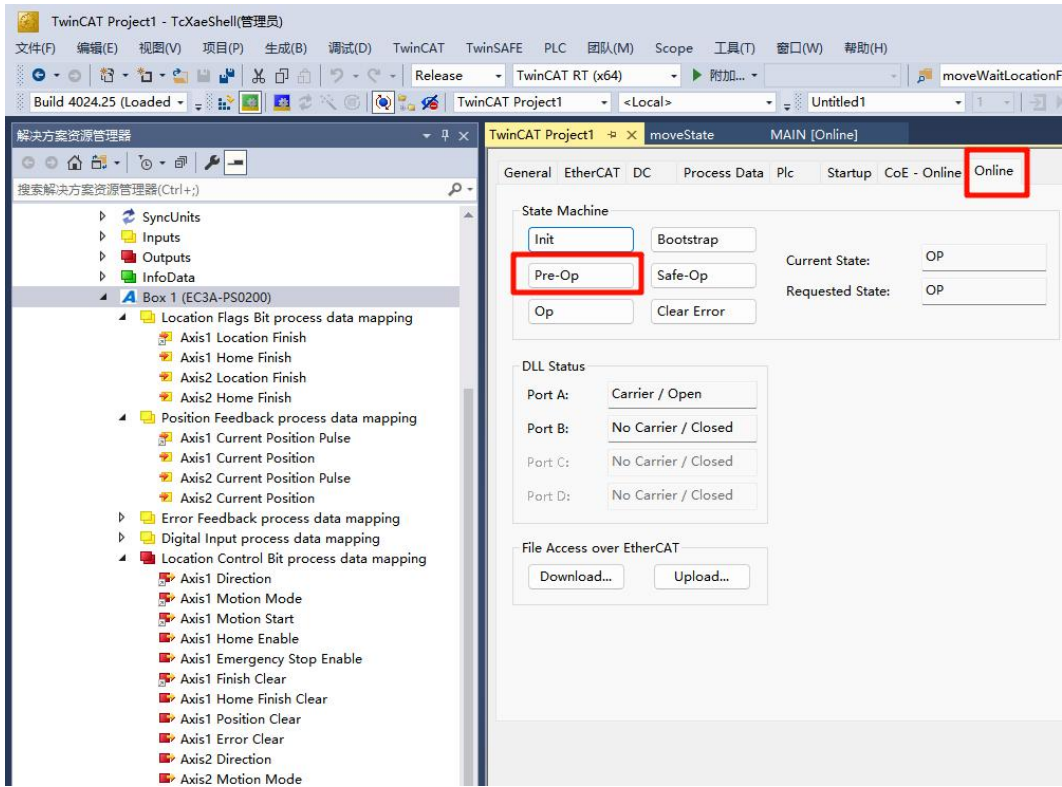
定位功能只需要将点动功能里的 Axis1 Motion Mode 设置为 1,再设置好 Axis1 Number Of Pulse 要发送的脉冲数, 其他参观点动功能。

配置参数设置:

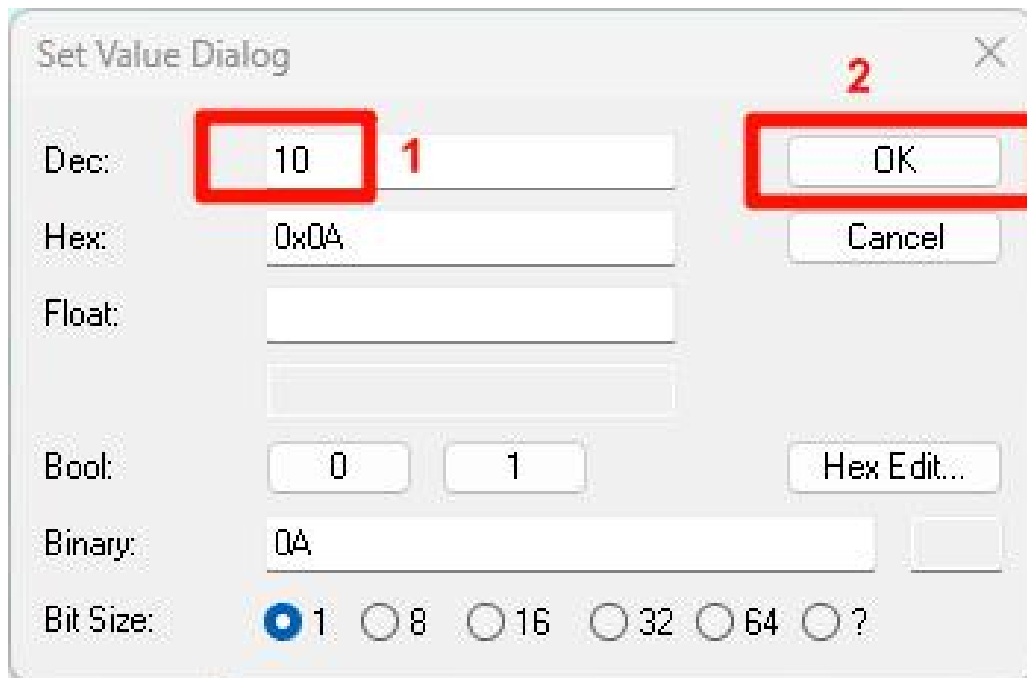
所有配置参数都在 0x8000 对象字里设置。双击 EC3A-PS0200 模块, 在右侧点击 CoE-Online 选项, 下拉找到 8000:0 展开, 即可看到所有配置数据, 如下图所示:



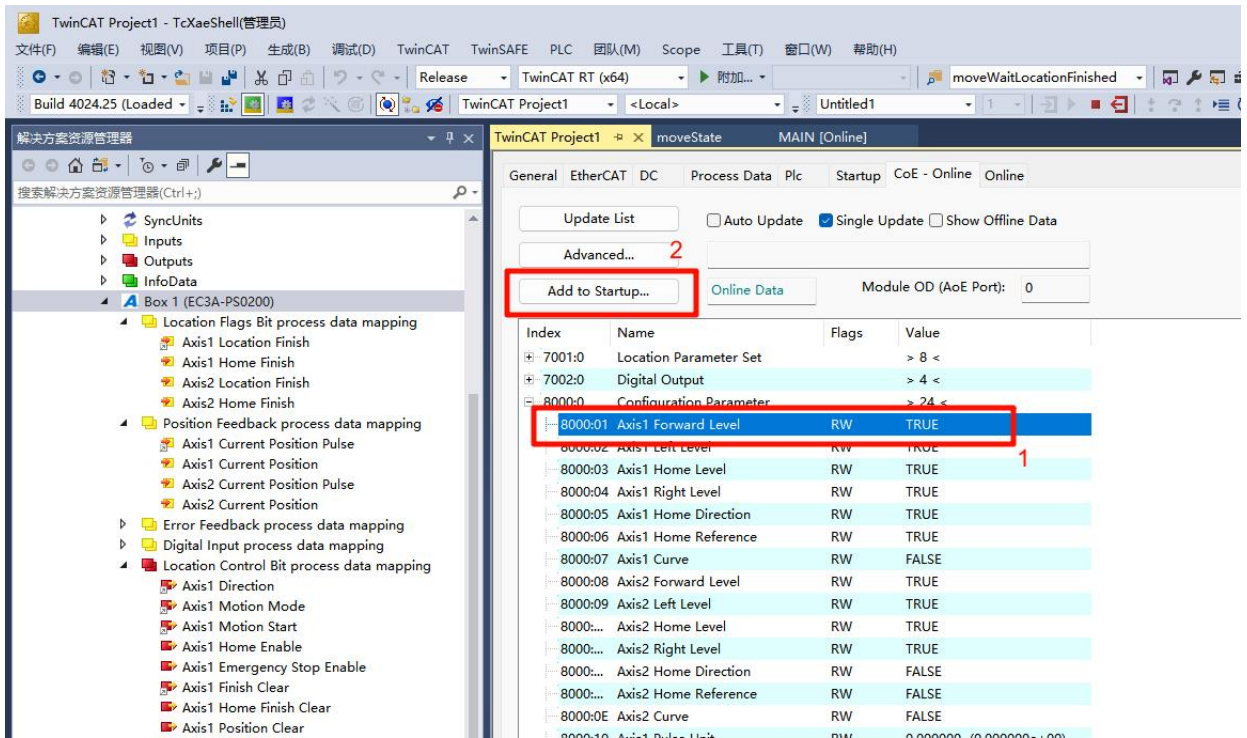
点击 Online, Pre-Op 将模块调到 PreOP 状态:



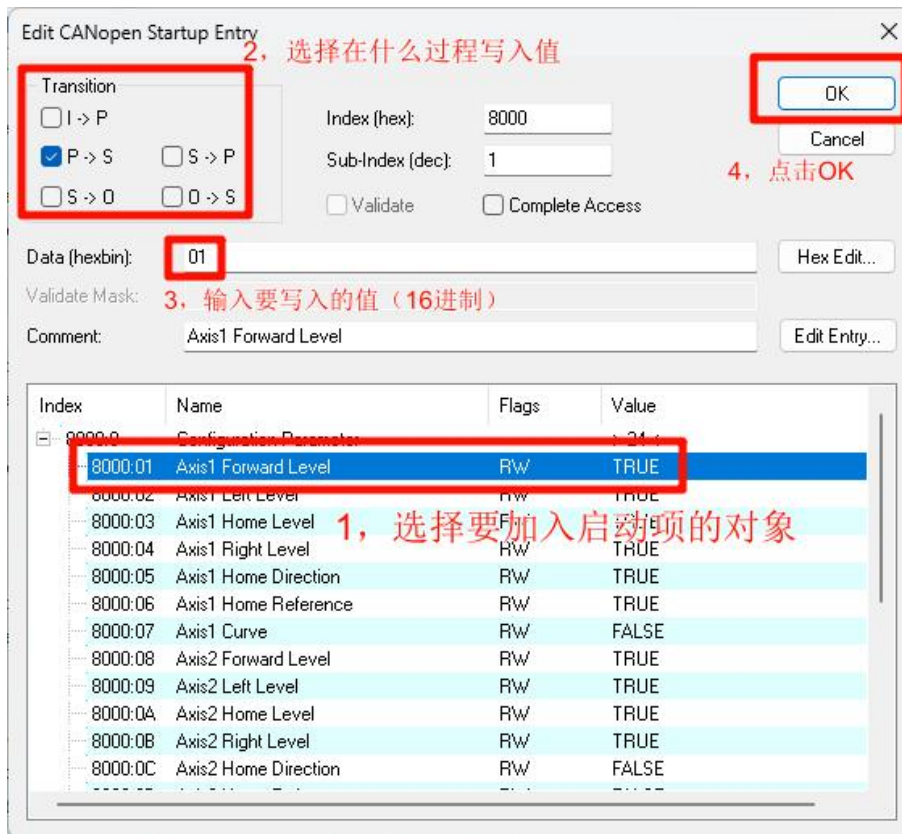
然后双击想要改变的配置参数写入期望的值:



添加启动项，单击需要加入启动项的对象字，再单击 Add to Startup...:



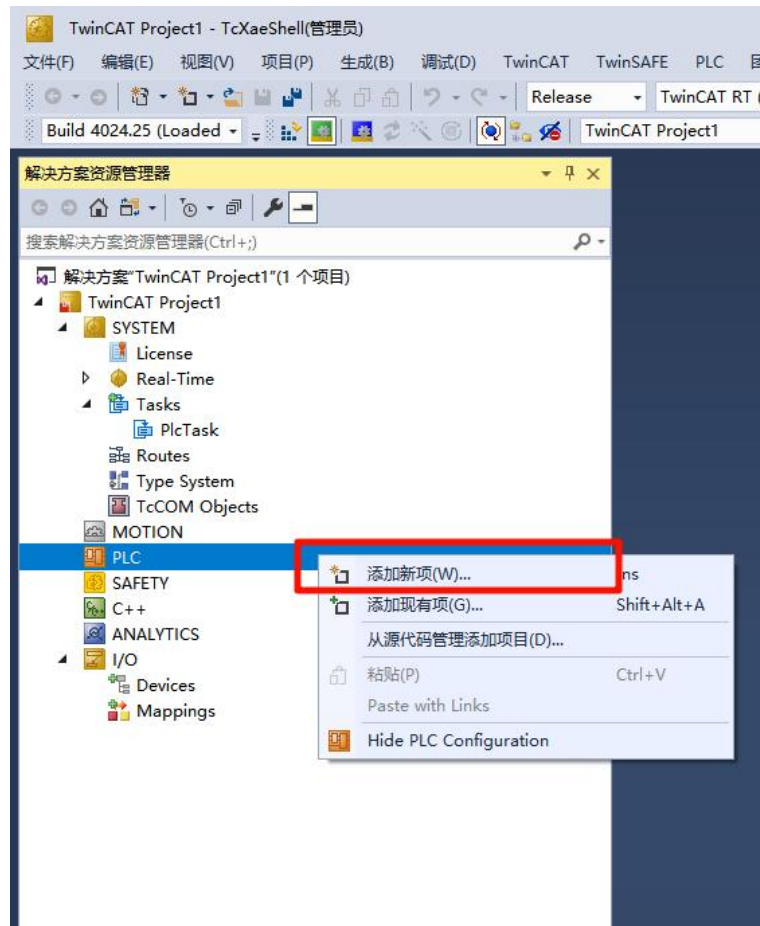
根据需要写入值，点击 OK，然后重载设备:



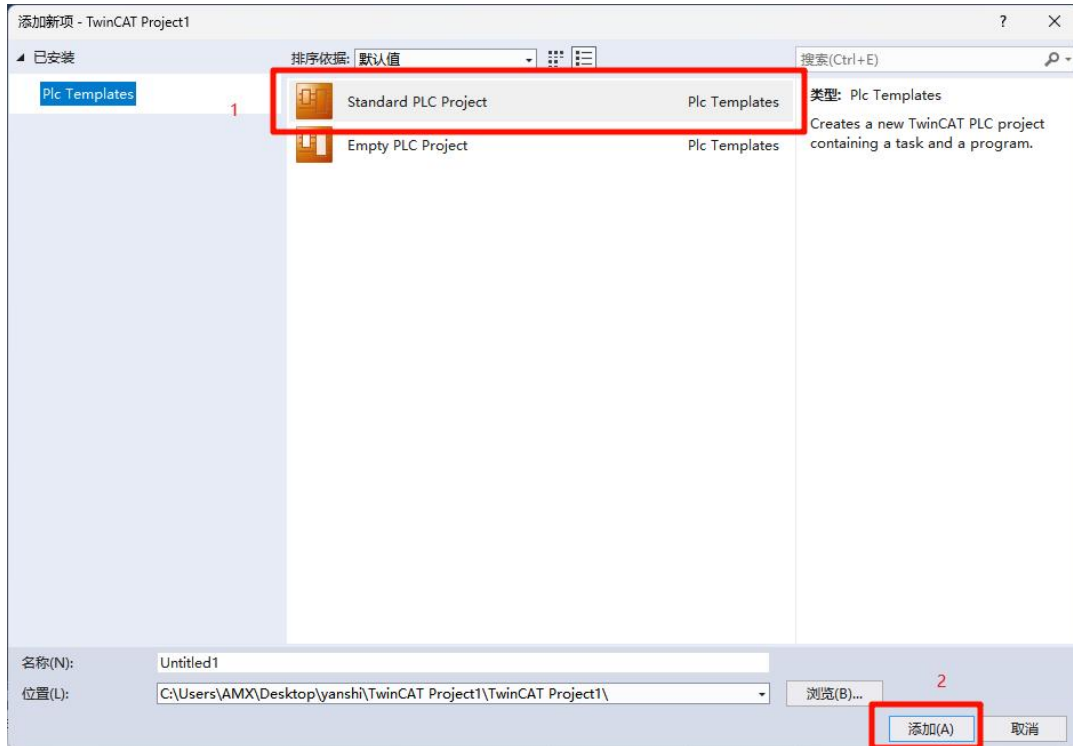
5.3、配合 PLC 使用

本章节利用该从站模块，用 ST 语言在 TwinCAT 实现电机循环正反转的例子。起始速度为 1，匀速速度为 100000，加减速时间为 1000ms。流程为：模块上电->回原点->电机正转 1000000 个脉冲->等待 1s->电机反转 1000000 个脉冲->等待 1s->电机正转 1000000 个脉冲.....如此循环。

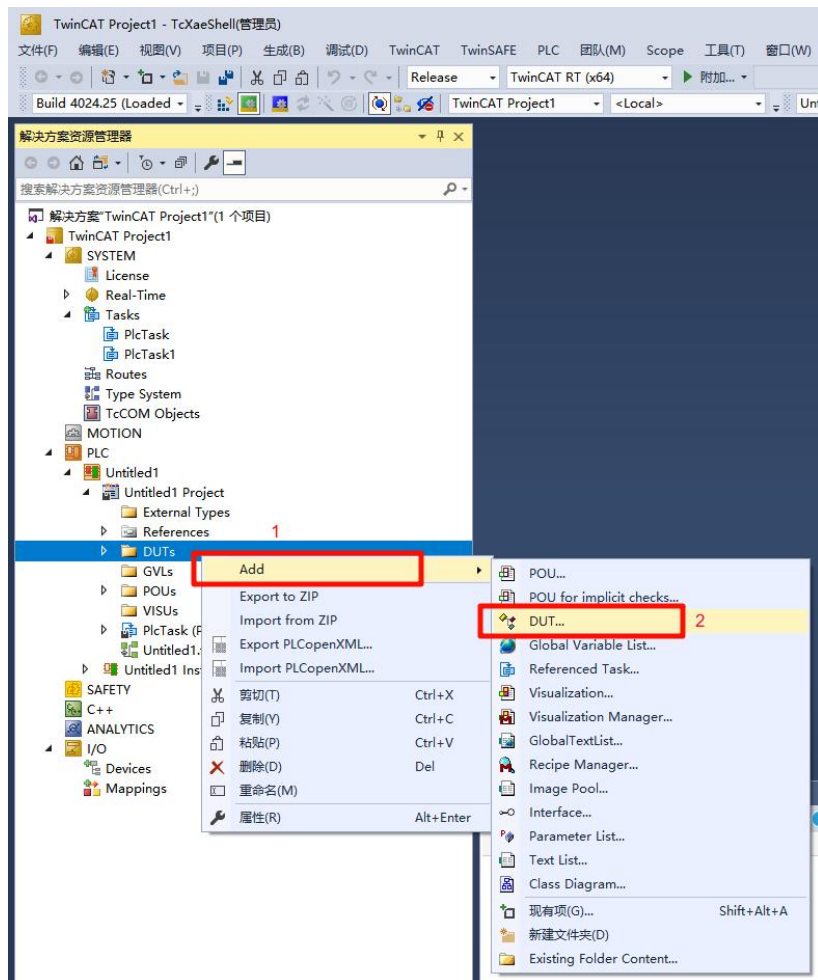
1) 如下图，找到 PLC 选项，先右键，在左键选择添加新项：



2) 选择 Standard PLC Project，点击添加（如有需要可以在名称和位置设置工程名称和工程存储的位置）：



3) 右键 DUTs, 在 Add 里选择 DUT, 如下图。



- 4) 输入数据类型名称“moveState”，选择枚举，点击打开：



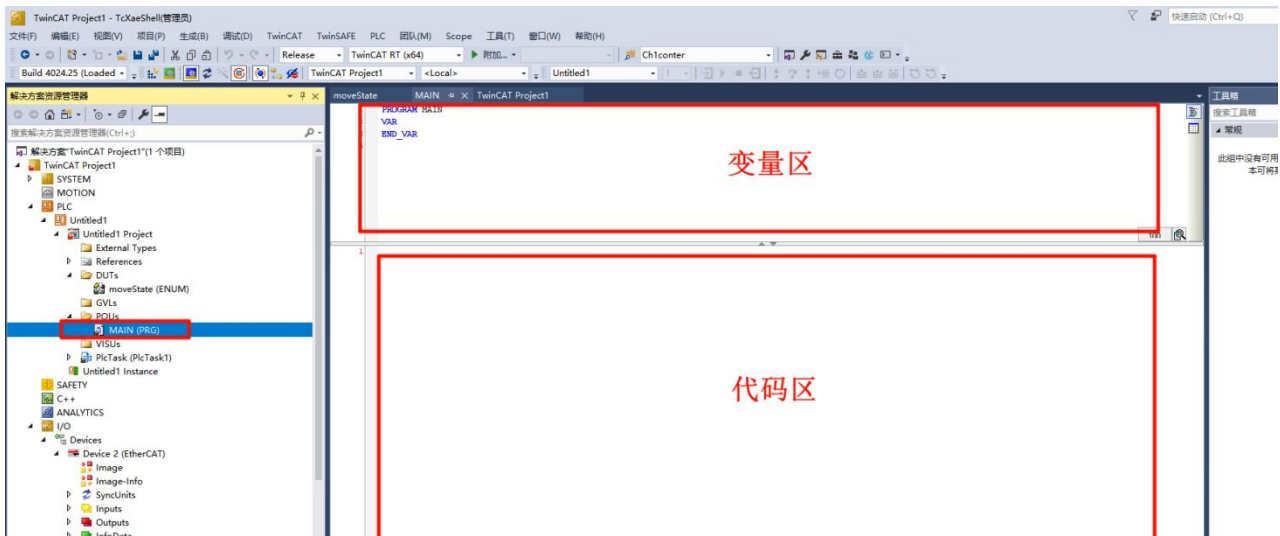
- 5) 输入以下代码，定义控制中的几种状态：

```

1 {attribute 'qualified_only'}
2 {attribute 'strict'}
3 TYPE moveState :
4 (
5     moveInit                := 0,
6     moveHome,
7     moveHomeFClear,
8     moveWaitHomeFinished,
9     moveLocation,
10    moveLocationFClear,
11    moveWaitLocaFClear,
12    moveWaitTimeOut,
13    moveWaitLocationFinished
14 );
15 END_TYPE
16

```

6) 展开 POU, 双击 MAIN(PRG), 打开编辑区, 如图分为变量编辑区和代码编辑区:



7) 在变量区输入以下代码:

```

1  PROGRAM MAIN
2  VAR_INPUT
3      ch0LocaFins      AT %I* : BOOL; //Axis1 Location Finish
4      ch0HomeFins     AT %I* : BOOL; //Axis1 Home Finish
5      ch0CurrPosition AT %I* : DINT; //Axis1 Current Position Pulse
6  END_VAR
7  VAR_OUTPUT
8      ch0Dir          AT %Q* : BOOL := 0; //Axis1 Direction
9      ch0LocaEn       AT %Q* : BOOL := 0; //Axis1 Motion Start
10     ch0LocaMode      AT %Q* : BOOL := 1; //Axis1 Motion Mode
11     ch0LocFClear     AT %Q* : BOOL := 0; //Axis1 Finish Clear
12     ch0StartFre      AT %Q* : DWORD := 1; //Axis1 Start Frequency
13     ch0MaxFre        AT %Q* : DWORD := 100000; //Axis1 Max Frequency
14     ch0LocaPulse     AT %Q* : DWORD := 1000000; //Axis1 Number Of Pulse
15     ch0RampTime      AT %Q* : WORD := 1000; //Axis1 Ramp Time
16     ch0HomEn         AT %Q* : BOOL := 0; //Axis1 Home Enable
17     ch0HomFClear    AT %Q* : BOOL := 0; //Axis1 Home Finish Clear
18  END_VAR
19  VAR
20     moveLength       : DWORD := 1000000; //往复运动距离
21     Ch0Timer         : ton; //1s定时器
22     Ch0TimerEn       : BOOL := 0; //定时器使能位
23     Ch0TimeOut       : BOOL := 1; //定时时间到标志
24     Ch0State         : moveState := moveState.moveInit; //状态机状态
25     runTimes         : DWORD := 0;
26  END_VAR
27
1  Ch0Timer(IN:= Ch0TimerEn, PT:= T#0.5S, Q=> Ch0TimeOut, ET=> ); //通道1暂停计时0.5s
2

```

8) 代码区输入以下代码:

```

Ch0Timer(IN:= Ch0TimerEn, PT:= T#1S, Q=> Ch0TimeOut, ET=> ); //通道1暂停

//通道1往复运动状态机
CASE Ch0State OF
  moveState.moveInit:
    ch0LocFClear := 0;
    ch0HomFClear := 0;
    ch0HomEn := 0;
    Ch0State := moveState.moveHome;

  moveState.moveHome:
    ch0HomFClear := 1;
    Ch0State := moveState.moveHomeFClear;

  moveState.moveHomeFClear:
    IF ch0HomeFins = 0 THEN
      ch0HomEn := 1;
      Ch0State := moveState.moveWaitHomeFinished;
    END_IF

  moveState.moveWaitHomeFinished:
    IF ch0HomeFins = 1 THEN
      ch0Dir := 0;
      Ch0State := moveState.moveLocationFClear;
    END_IF

  moveState.moveLocation:
    ch0LocFClear := 0;
    ch0Dir := NOT ch0Dir;
    ch0LocaEn := 1;
    Ch0State := moveState.moveWaitLocationFinished;

  moveState.moveLocationFClear:
    ch0LocFClear := 1;
    Ch0TimerEn := 0;
    Ch0State := moveState.moveWaitLocaFClear;

  moveState.moveWaitLocaFClear:
    IF ch0LocaFins = 0 THEN
      ch0LocaEn := 0;
      ch0LocaMode := 1;
      Ch0TimerEn := 1;
      Ch0TimeOut := 0;

      Ch0State := moveState.moveWaitTimeOut;
    END_IF

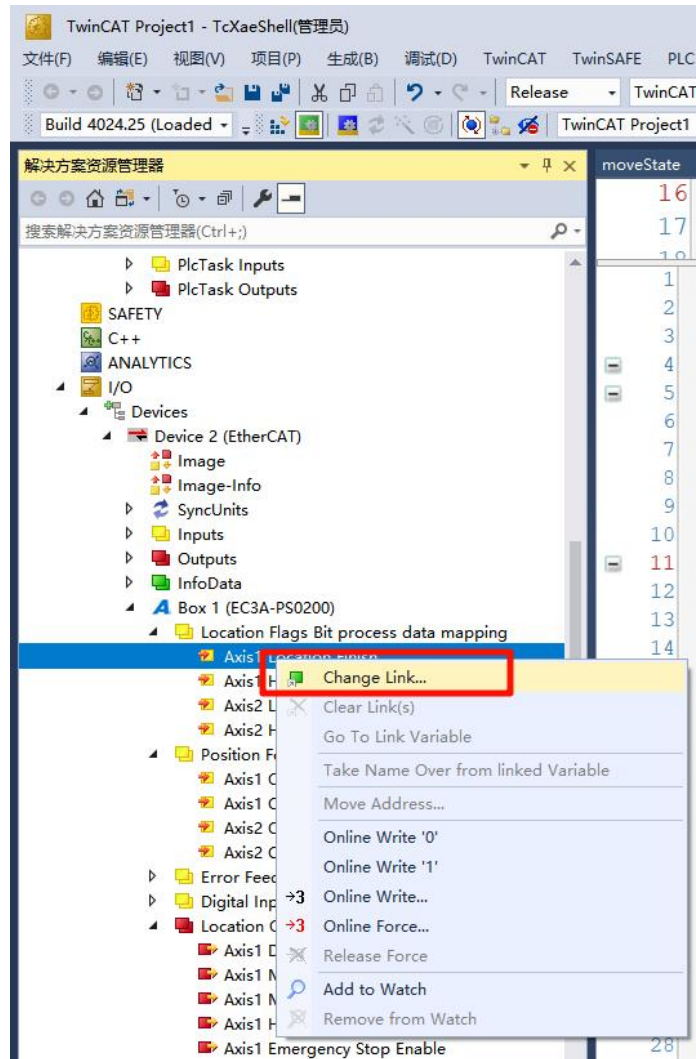
  moveState.moveWaitTimeOut:
    IF Ch0TimeOut = 1 THEN
      Ch0State := moveState.moveLocation;
    END_IF

  moveState.moveWaitLocationFinished:
    IF ch0LocaFins = 1 THEN
      runTimes := runTimes + 1;
      Ch0State := moveState.moveLocationFClear;
    END_IF

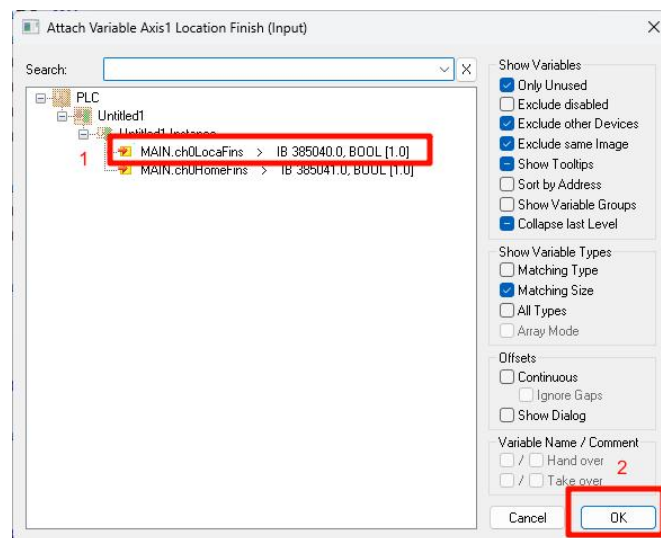
END_CASE

```

9) 在 Box 里右击需要链接的对象，然后单击“Change Link...”：



选择对应的变量，点击 OK，将变量与对象字链接：



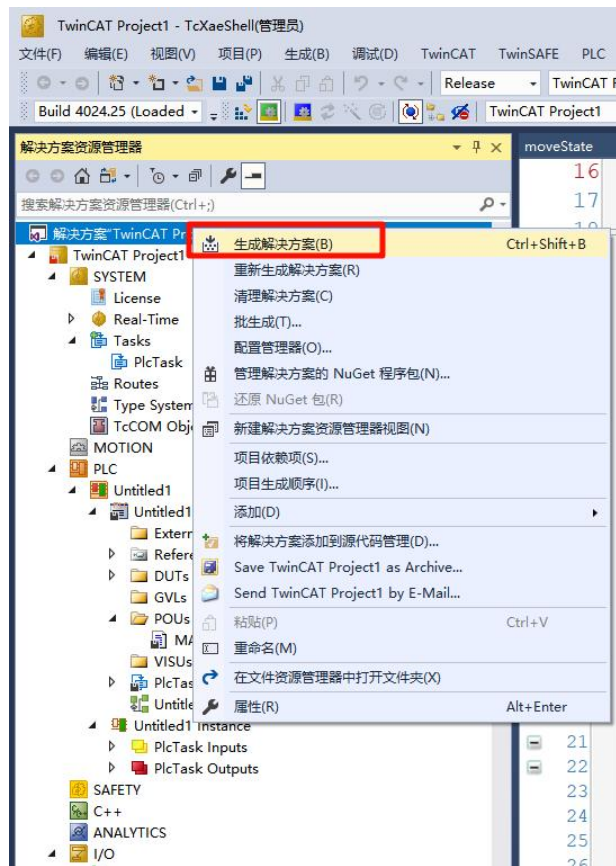
同样方法将下图变量区中以下变量全部都与注释里的对象进行链接：

```

1 PROGRAM MAIN
2 VAR_INPUT
3   ch0LocaFins      AT %I* : BOOL; //Axis1 Location Finish
4   ch0HomeFins     AT %I* : BOOL; //Axis1 Home Finish
5   ch0CurrPosition AT %I* : DINT; //Axis1 Current Position Pulse
6 END_VAR
7 VAR_OUTPUT
8   ch0Dir          AT %Q* : BOOL := 0; //Axis1 Direction
9   ch0LocaEn       AT %Q* : BOOL := 0; //Axis1 Motion Start
10  ch0LocaMode     AT %Q* : BOOL := 1; //Axis1 Motion Mode
11  ch0LocFClear    AT %Q* : BOOL := 0; //Axis1 Finish Clear
12  ch0StartFre     AT %Q* : DWORD := 1; //Axis1 Start Frequency
13  ch0MaxFre       AT %Q* : DWORD := 100000; //Axis1 Max Frequency
14  ch0LocaPulse    AT %Q* : DWORD := 1000000; //Axis1 Number Of Pulse
15  ch0RampTime     AT %Q* : WORD := 1000; //Axis1 Ramp Time
16  ch0HomEn        AT %Q* : BOOL := 0; //Axis1 Home Enable
17  ch0HomFClear    AT %Q* : BOOL := 0; //Axis1 Home Finish Clear
18 END_VAR
19 VAR
20   moveLength      : DWORD := 1000000; //往复运动距离
21   Ch0Timer        : ton; //1s定时器
22   Ch0TimerEn      : BOOL := 0; //定时器使能位
23   Ch0TimeOut      : BOOL := 1; //定时时间到标志
24   Ch0State        : moveState := moveState.moveInit; //状态机状态
25   runTimes        : DWORD := 0;
26 END_VAR
27
1 Ch0Timer(IN:= Ch0TimerEn, PT:= T#0.5S, Q=> Ch0TimeOut, ET=> ); //通道1暂停计时0.5s
2

```

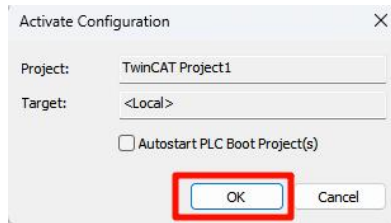
10) 右键“解决方案.....”，再点击“生成解决方案”进行编译，如果报错请根据错误提示检查错误：



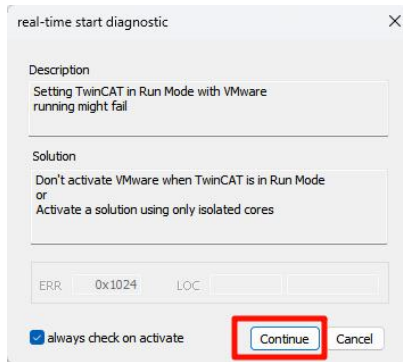
11) 检查没有错误之后，点击激活配置：



点击 OK:



点击 Continue:



点击“确定”:



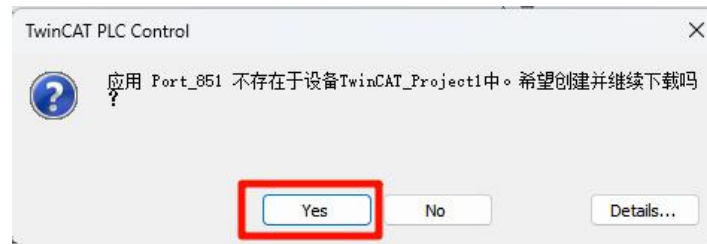
此时应当进入运行模式，右下角显示为绿色:



12) 登录查看运行情况，点击登录到:



点击 Yes:



13) 点击运行，启动 PLC 开始运行:



监视窗口查看当前位置在 0~1000000 之间变化:

表达式	类型	值	准备值	地址	注释
ch0LocaFins	BOOL	FALSE		%I*	
ch0HomeFins	BOOL	FALSE		%I*	
ch0CurrPosition	DINT	10		%I*	
ch0Dir	BOOL	TRUE		%Q*	Ch0 Direction Control
ch0LocaEn	BOOL	FALSE		%Q*	Ch0 Location Enable



修订历史

版本	修订日期	修订说明	维护人
V1.0.0	2024.11.21	初始版本	WH

关于我们

企业名称：东莞市艾莫迅自动化科技有限公司

官方网站：www.amsamotion.com

技术服务：4001-522-518 拨 1

企业邮箱：sale@amsamotion.com

公司地址：广东省东莞市南城区袁屋边艺展路 9 号兆炫智造园 B 栋 1 楼



官方公众号



官方抖音