

2022年8月1日更新

安装手册

AZ系列 多轴驱动器 DC电源 EtherCAT对应

EtherCAT主站：倍福（BECKHOFF）

EPC(Embedded PC)CX2030、EK1110、CX2100

TwinCAT3 控制篇

以CiA402为基准进行Motion控制运行。

本书介绍如何形成位置管理、位置指令的运动控制的设定方法，以及简单运行的操作流程。各个项目的Index，如果没有明确标识即表示Axis1。

1. BECKHOFF TwinCAT 3 的通信设定

2. Cyclic同步位置模式（CSP）的运行方法

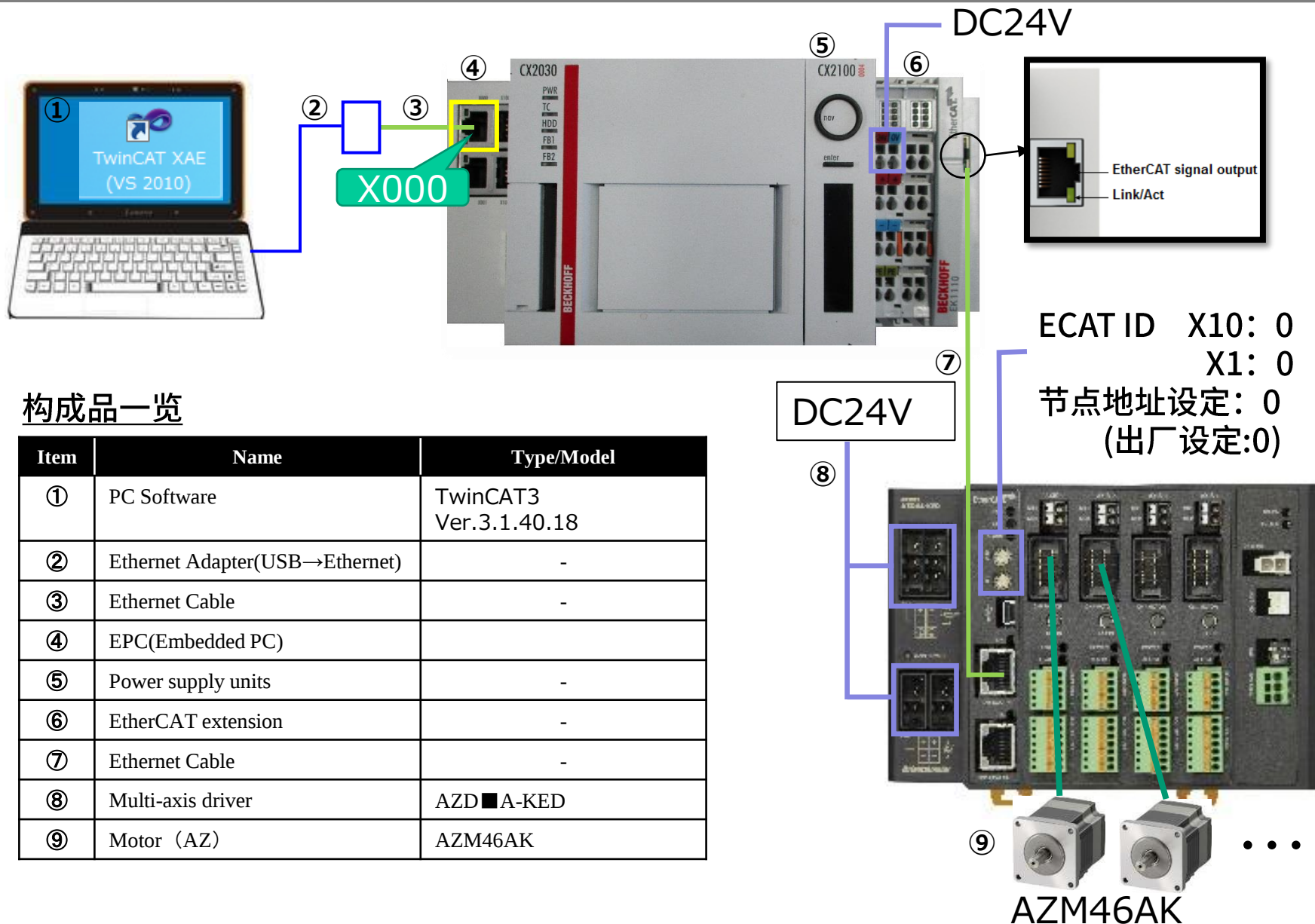
初学者请从 1 开始,对EtherCAT产品有使用经验者请从2开始阅览。

2 以样本程序为基础进行说明。根据您购买的产品，在一些设定上需要做相应的更改。请参考1的内容进行PLC的设定。

1.BECKHOFF TwinCAT 3 的通信设定	10
1.1 PC与EPC的通信设定	
Target PC与EPC之间通信连接测试	12
Target 在TwinCAT 3 上确认PC与EPC之间的通信	16
1.2 网络构成的设定	
Target 自动检测连接到EtherCAT上的设备，并确认通信状态	22
2. CSP模式的运行方法	26
2.1 运行前的各种设定	
Target 进行CSP测试运行	28
2.2 制作程序	
Target 制作Motion运行的程序	36
2.3 通过CSP模式进行定位运行	
Target 通过CSP模式运转电动机	53
2.4 通过SDO通信进行参数读写	
Target 通过SDO通信进行参数读写	57

- (1) 在建立实际系统时,请确认好构成系统的各个机器・装置的规格后,采取在额定・性能上留有余地的用法,采取即使万一发生故障也可将危险降至最低的安全电路等安全对策。
- (2) 为安全使用系统,请获取构成系统的各个机器・装置的手册或使用说明书等,确认好「安全上的注意」「安全上的要点」等有关安全的注意事项、内容后使用。
- (3) 请客户自行确认系统应符合的规格・法规或限制。
- (4) 本资料的一部分或全部内容未经东方马达株式会社的许可,禁止复写、复制、再分发。
- (5) 本资料所记载的内容会因改良而有所变更,恕不另行通知。

系统构成图



参考：Profile位置模式/原点返回模式

操作模式的profile位置模式（PP: Profile Position）与原点返回模式（HM: Homing）通过驱动器的内部profile运行。

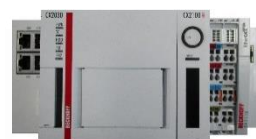
因此,与EPC连接时,不设定运动轴。

※Profile速度（PV:Profile Velocity）也同样通过内部profile运行。

※由于未设定运动轴,MC_POWER等的功能块(FB)不能使用。

【PP时】

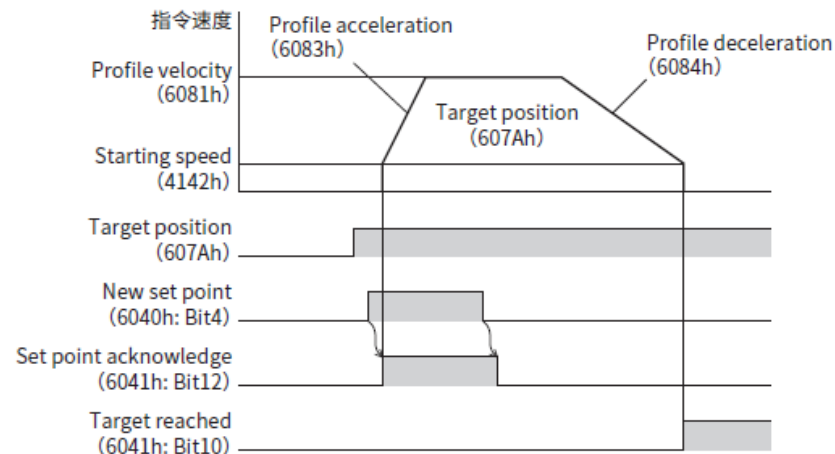
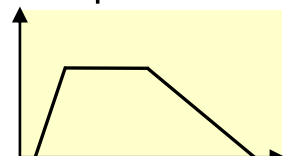
主站



起动指令、目标位置、Profile速度、...

(EtherCAT通信)

生成profile

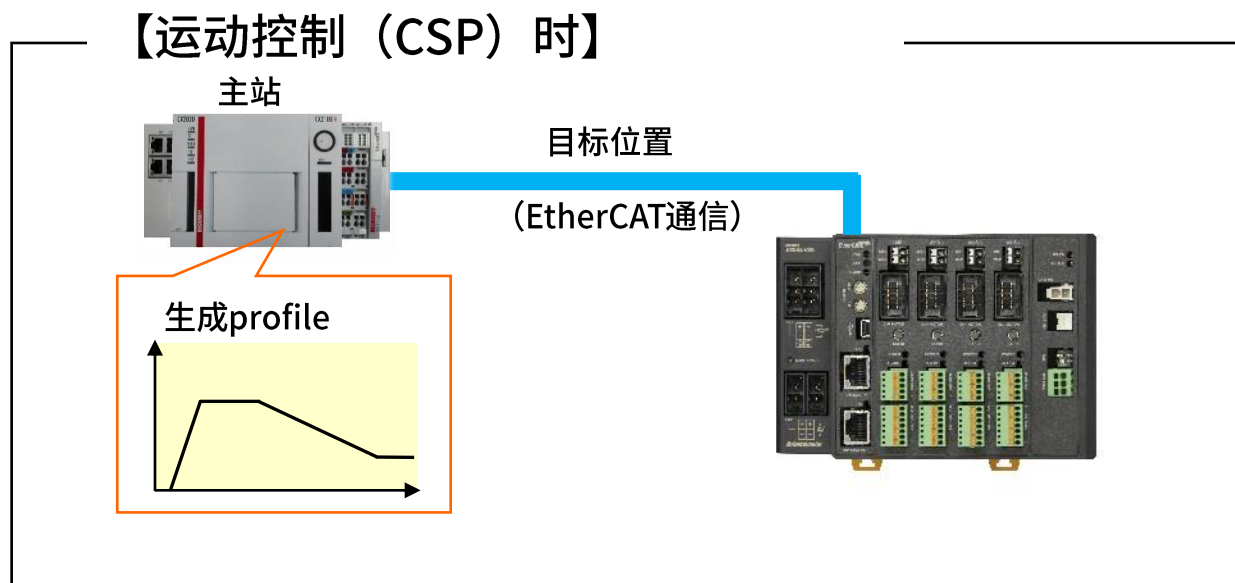


PP时, 预先设定目标位置（607Ah）、profile速度（6081h）,使运行起动指令（控制字Bit4: New set point）由0→1, 开始运行。

当通过EtherCAT通信发出运行起动指令时, 将在驱动器内部进行位置指令及速度的加减速计算。

若在TwinCAT 3 上进行轴设定，则在主站侧形成位置指令和速度加减速等的profile。

- AZ驱动器的操作模式(operation mode)包括Cyclic同步位置模式（CSP）和Cyclic同步速度模式（CSV）。
- 在EPC上设定运动轴时，可使用MC_POWER等功能块（FB）。



CSP通过EtherCAT的Cyclic 通信（PDO通信）在每个通信周期都将目标位置（绝对值）发送到驱动器。

参考：AZ多轴驱动器分辨率

• DC电源 电动机单体



	AZ单轴	AZ多轴
		
分辨率	0.36°/脉冲(=1,000P/R)	0.036°/脉冲(=10,000P/R)

POINT AZ多轴是以通信进行Motion运行为前提的商品。因此，分辨率的初始设定为10,000P/R。减速机输出轴分辨率的计算方法是用以上分辨率除以减速比。

• 电动滑台 EAS、EZS系列 电动缸 EAC系列



	AZ单轴	AZ多轴
最小移动量	0.01mm (数值与导程无关)	0.3μm、0.6μm、1.2μm (分别为导程3mm、6mm、12mm的数值)

POINT AZ单轴，根据实际组装的机构产品，事先设定好了最小移动量为0.01mm。
AZ多轴产品，由于电动机分辨率是统一的10,000P/R，因此需要从导程进行计算。

参考：AZ多轴驱动器分辨率

• 小型电动缸 DRS2



	AZ单轴  	AZ多轴 
最小分辨率	0.001mm (数值与导程无关)	0.2 μ m、0.4 μ m、0.8 μ m (分别为导程2mm、4mm、8mm的数值)

POINT AZ单轴，根据实际组装的机构产品，事先设定好了最小移动量为0.01mm。
AZ多轴产品，由于电动机分辨率是统一的10,000P/R，因此最小移动量不同。

• 中空旋转式传动装置 DG II



	AZ单轴	AZ多轴
输出工作台 最小移动量	0.01°/STEP	0.002°/STEP

POINT 本产品的机构内部为减速比18的减速器构造。
AZ单轴，事先将最小移动量设定为0.01°/STEP。
AZ多轴产品，由于电动机分辨率是统一的10,000P/R，因此最小移动量不同。

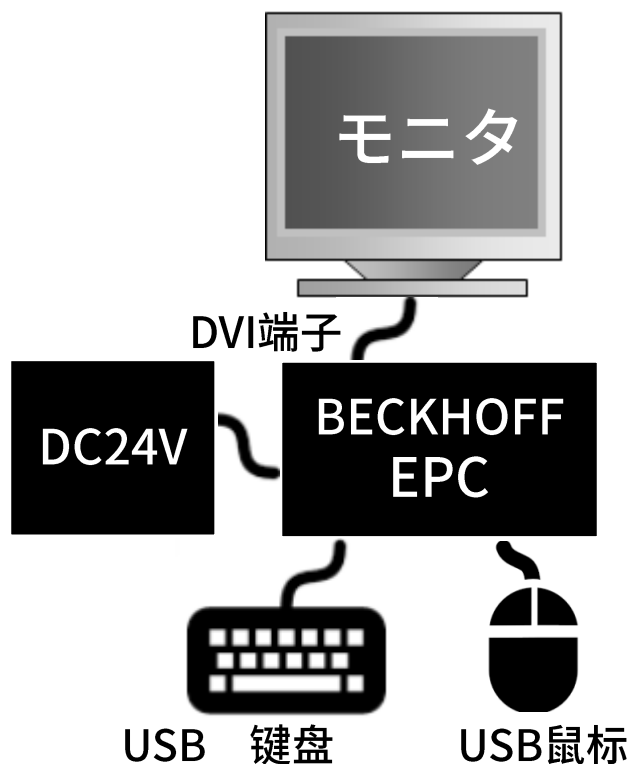
1. BECKHOFF TwinCAT 3 的通信设定
2. Cyclic同步位置模式（CSP）的运行方法

PC与EPC的通信设定

EPC与周边设备的配线示例

- 进行PC和EPC的通信，需要设定各自的IP。
设定EPC的IP,首先**不要连接PC和EPC**,而要先将显示器(DVI)、键盘(USB)、鼠标(USB)等周边设备连接到EPC上。
- EPC与周边设备的连接示意图如下。

- 示意图



- 实际的连接环境（EPC接通电源时）



- 如下所示,将设备连接到EPC上
- 打开EPC的主电源,启动Windows,然后登录EPC



至鼠标键盘

至显示器(屏幕)
*EPC的显示器输出是数字端子(DVI端子),因此请注意。



Target

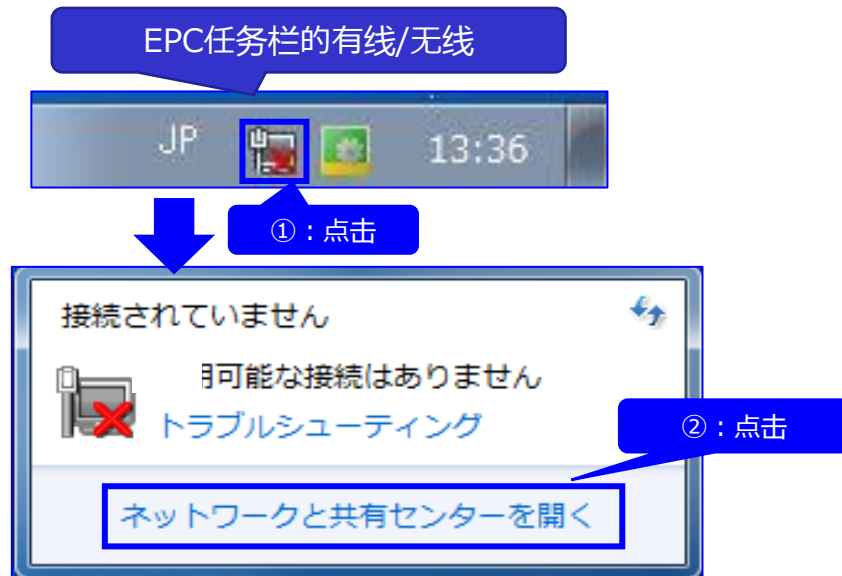
设定PC、EPC之间的通信,实施连接测试以确认能否正确通信。

EPC的Ethernet连接器:设定X000的IP地址和子网掩码。

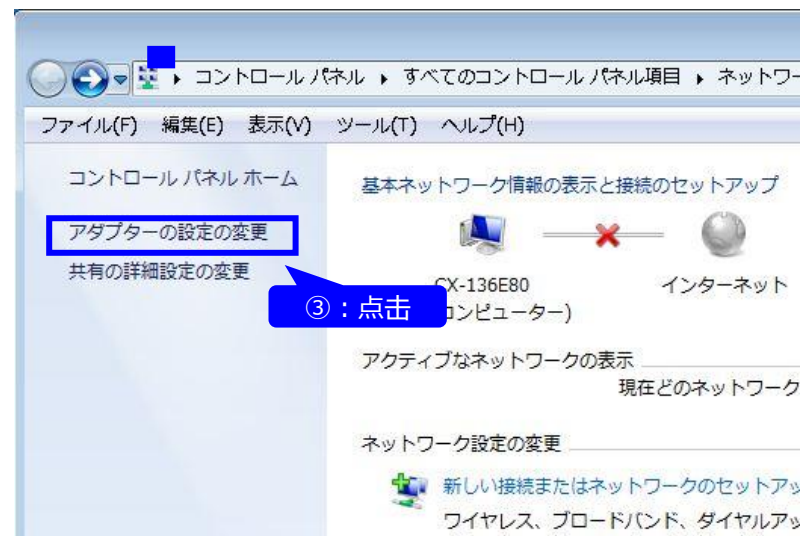
【注意】 设定之前,请与网络管理员确认IP地址。请不要在同一网络上设定重复的IP地址。

■ 设定步骤

1. 网络与共享中心



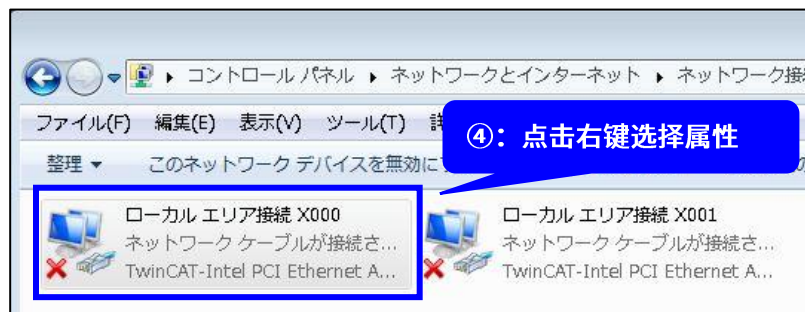
2. 变更适配器的设定



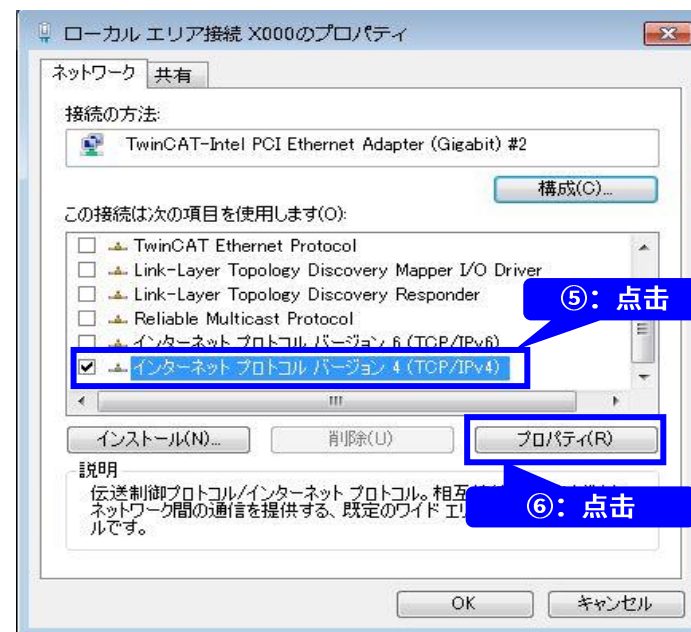
PC与EPC的通信设定

EPC的IP地址设定

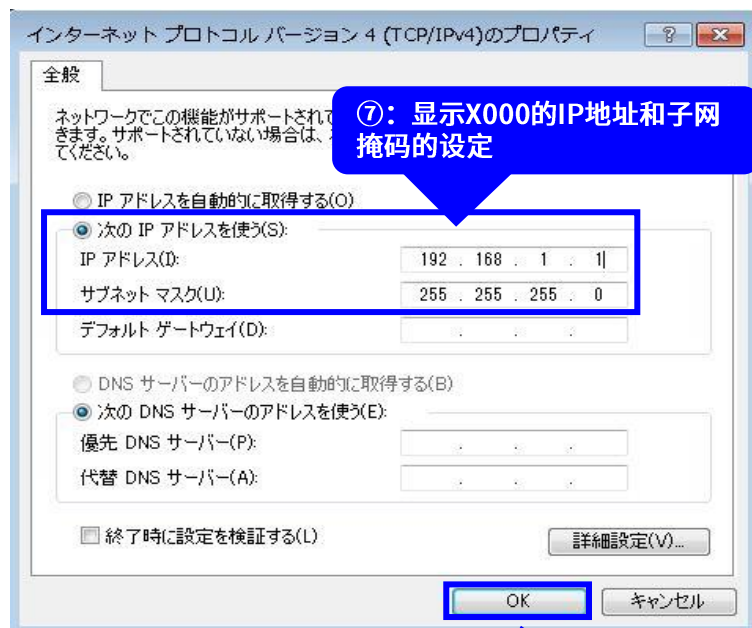
3. 选择本地连接X000,设定IP



4. 选择Internet协议版本4



5. 设定EPC的IP地址和子网掩码



⑨: 不仅仅是PC,还要考虑连接到HMI上,
下面显示的是X001的IP设定示例。

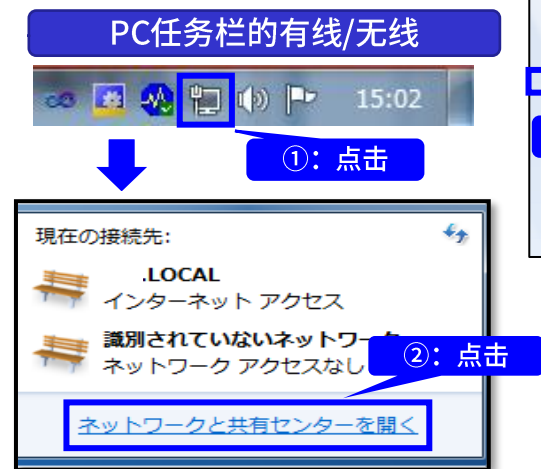
- IP地址是192.168.1.2
- 子网掩码: 255.255.255.0

PC与EPC的通信设定

PC的IP地址设定

设定PC端Ethernet Adapter的IP地址和子网掩码

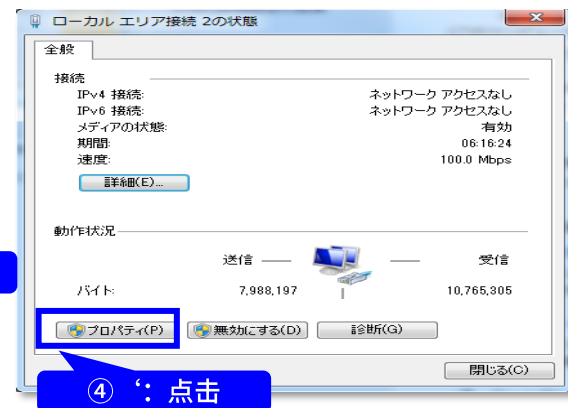
1. 打开网络 and 共享中心



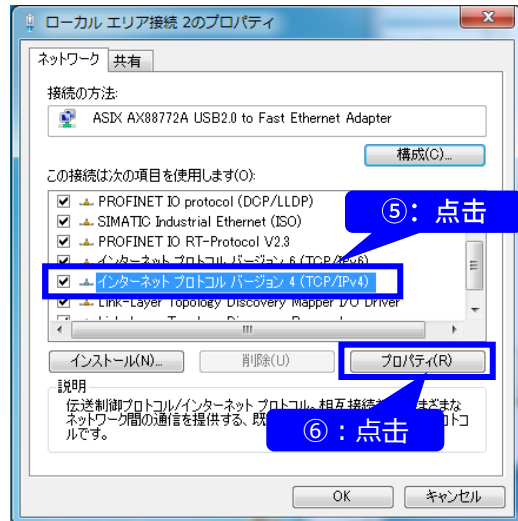
2. 选择Ethernet Adapter的网络



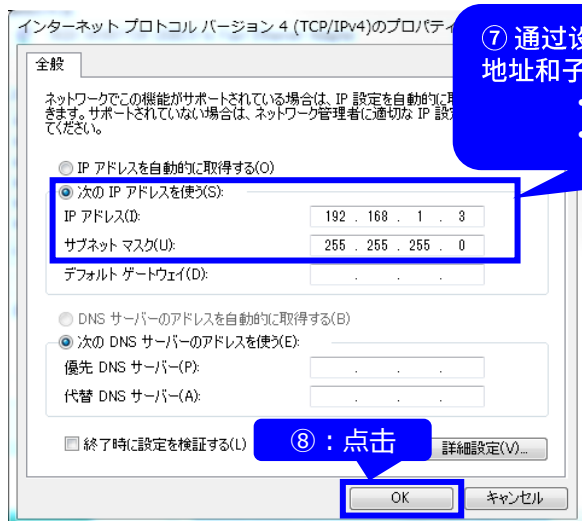
3. 选择Ethernet Adapter的连接状态的属性



4. Ethernet Adapter的属性



5. 设定要使用的Ethernet Adapter的IP地址和子网掩码



⑦ 通过设定EPC的IP地址来设定Ethernet Adapter的IP地址和子网掩码

- IP地址是: 192.168.1.3
- 子网掩码: 255.255.255.0

EPC的X000 IP地址 (参考)



EPC的X001 IP地址 (参考)



PC与EPC的通信设定

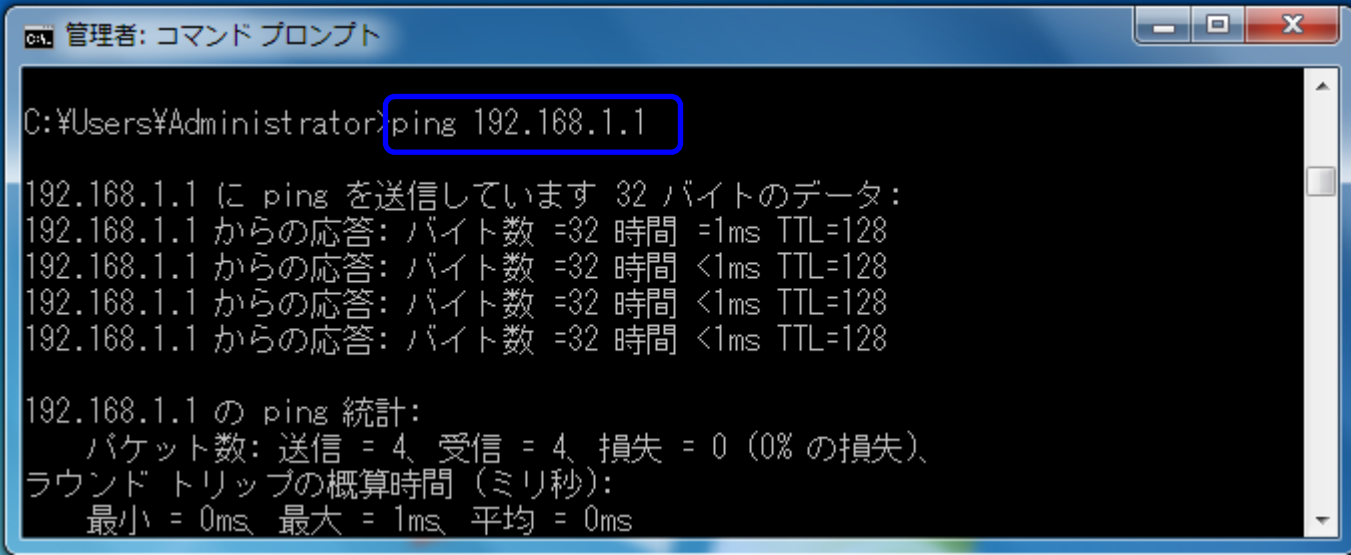
通过PING指令进行通讯测试

从PC向EPC传送数据，测试是否有应答。

IP地址的状态 EPC: 192.168.1.1 (X000ポート)

PC : 192.168.1.3

从内置Windows的PC上打开命令提示符，
使用ping指令，确认从EPC上有以下的应答。



```
管理员: コマンド プロンプト
C:\Users\Administrator>ping 192.168.1.1

192.168.1.1 に ping を送信しています 32 バイトのデータ:
192.168.1.1 からの応答: バイト数 =32 時間 =1ms TTL=128
192.168.1.1 からの応答: バイト数 =32 時間 <1ms TTL=128
192.168.1.1 からの応答: バイト数 =32 時間 <1ms TTL=128
192.168.1.1 からの応答: バイト数 =32 時間 <1ms TTL=128

192.168.1.1 の ping 統計:
    パケット数: 送信 = 4、受信 = 4、損失 = 0 (0% の損失)、
ラウンド トリップの概算時間 (ミリ秒):
    最小 = 0ms、最大 = 1ms、平均 = 0ms
```

Point 没有应答时明，请确认通信电缆线和通信时间的设定是否与问题。

PC与EPC的通信设定

下载ESI文件

Target

在TWINCAT 3 上确认PC与EPC之间的通信。

在PLC上安装ESI文件。

请从以下页面下载多轴驱动器的ESI文件。

https://www.orientalmotor.co.jp/products/stepping/az_driver/features/

※请确认驱动器上的生产日期。2017年2月以后生产的多轴驱动器的版本号更新为□□□□0200(hex)。(□的数字根据轴数不同儿不同)
如果需要2017年2月以前生产的多轴驱动器的ESI文件，
请向最近的办事处，营业所或客服中心询问。

● 回生抵抗(オプション)

2DCAD	回生抵抗	B1209.dxf
3DCAD	回生抵抗(STEP形式)	B1209.zip

その他

点击

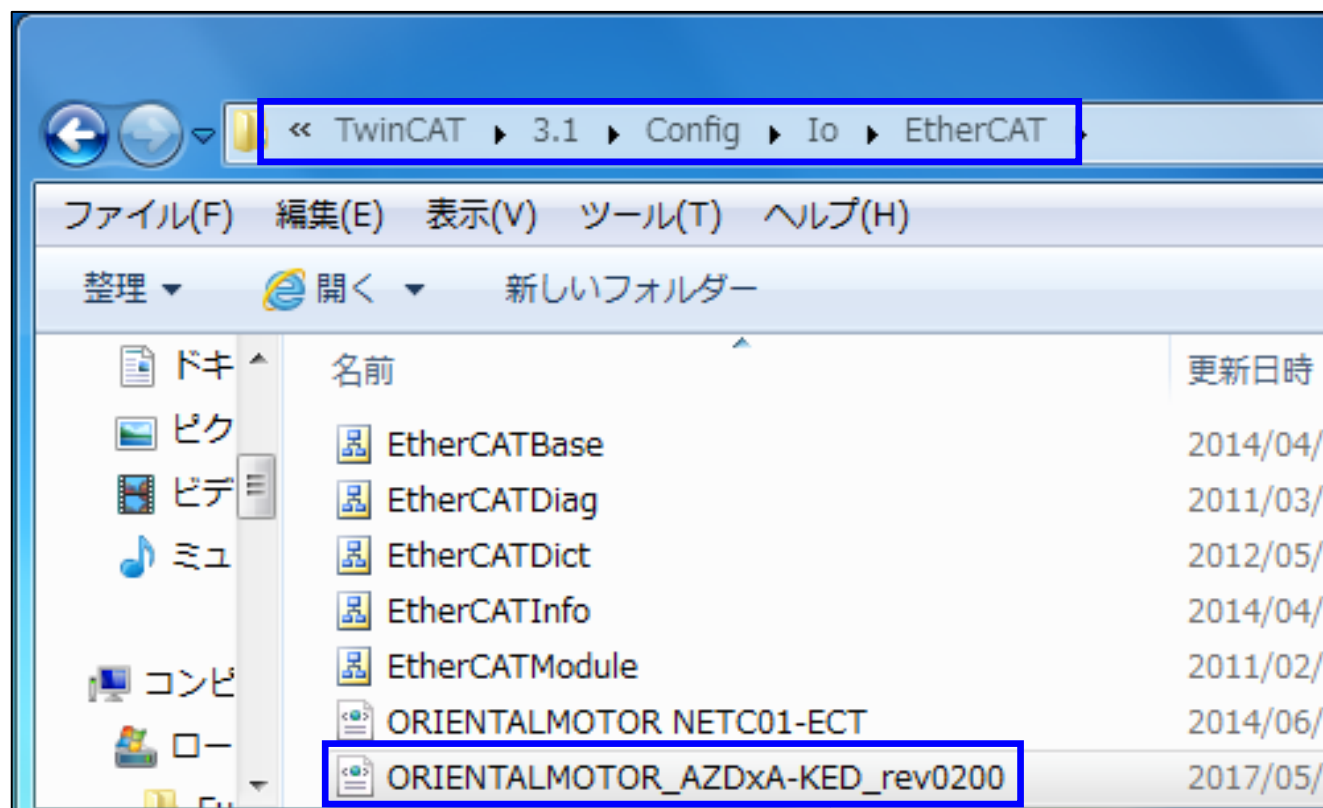
● [EtherCAT ESIファイル](#)

EtherCATスレーブ製品を当社ドライバと接続してお使いいただく場合に必要なESIファイルです。

PC与EPC的通信设定

安装ESI文件

请将下载的ESI文件复制到安装了TwinCAT 3 的文件夹中。
(\\TwinCAT\\3.1\\Config\\Io\\EtherCAT)



粘贴

PC与EPC的通信设定

TwinCAT3新建项目

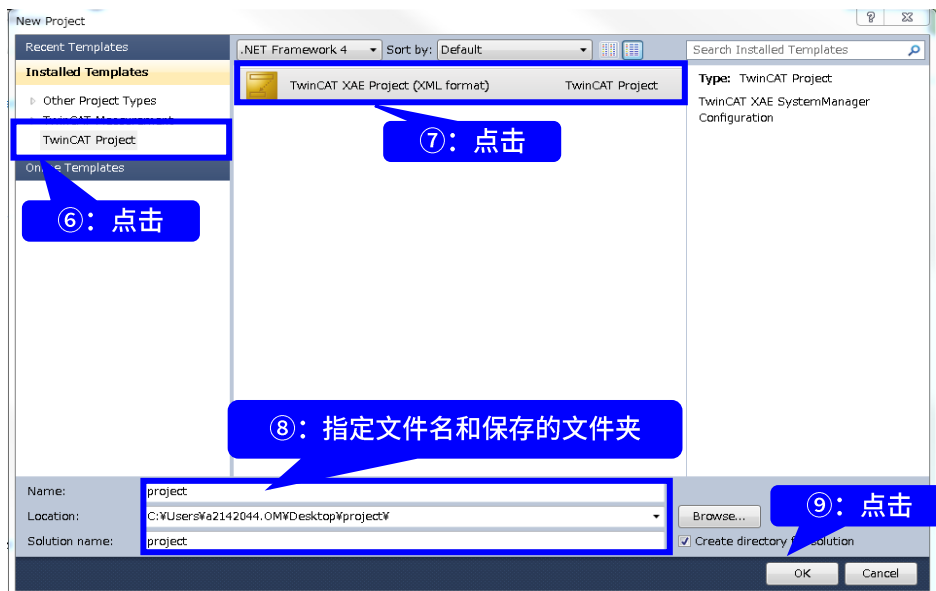
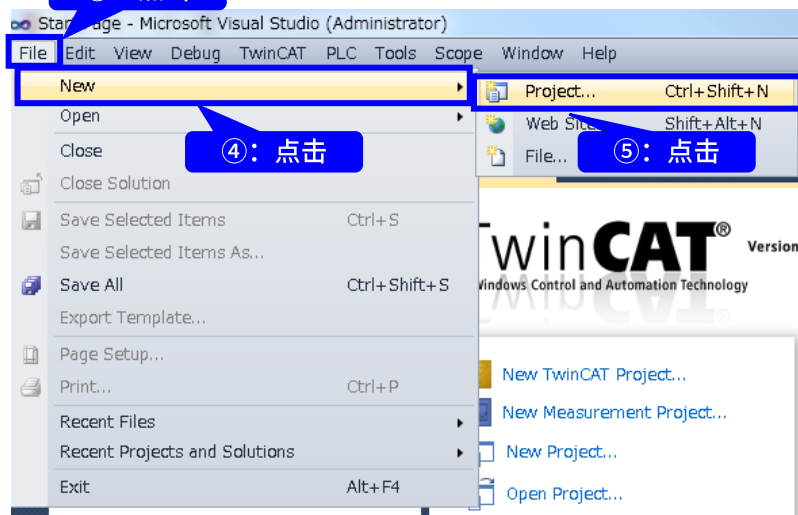
1. 启动TwinCAT XAE(VS2010)

①: 点击

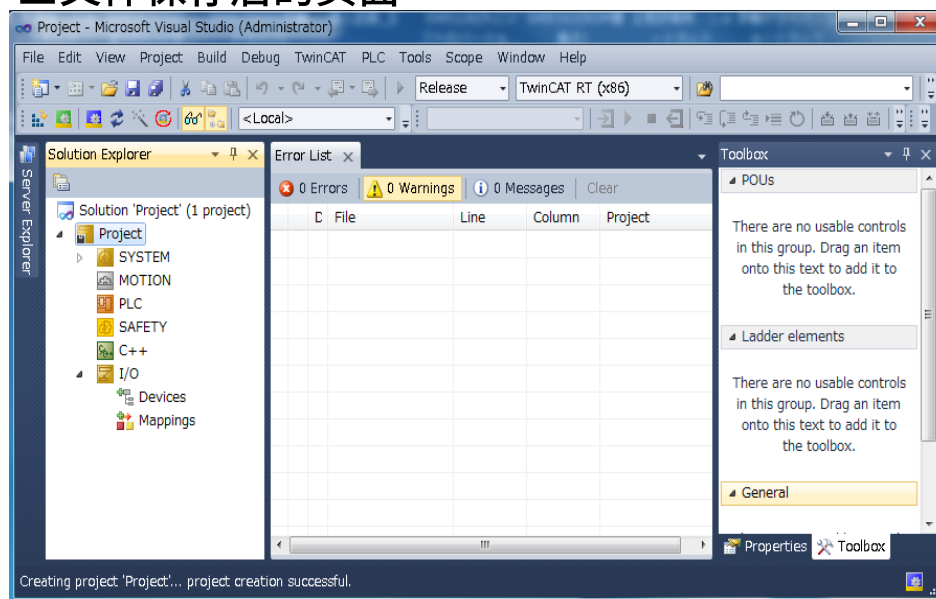


2. 新建项目

③: 点击

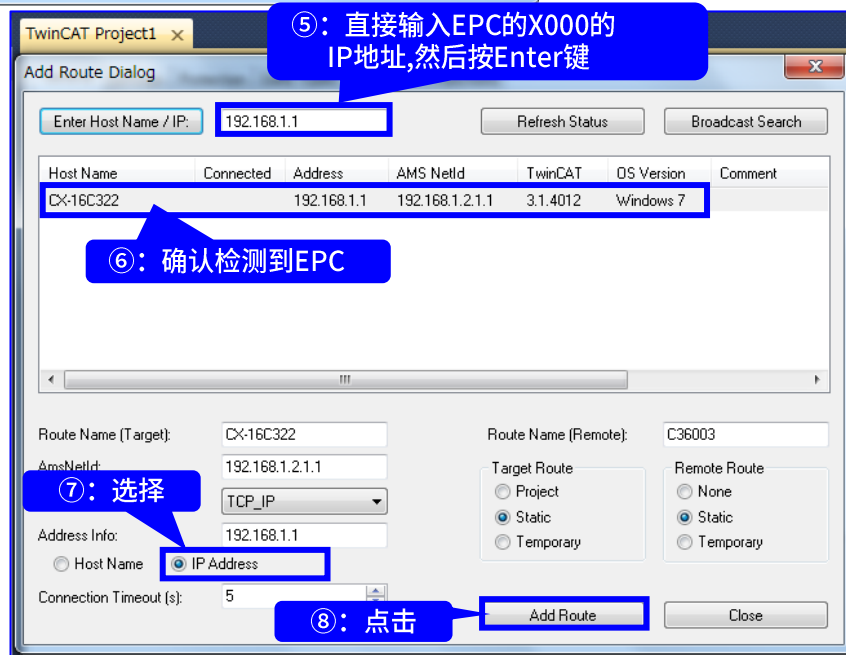
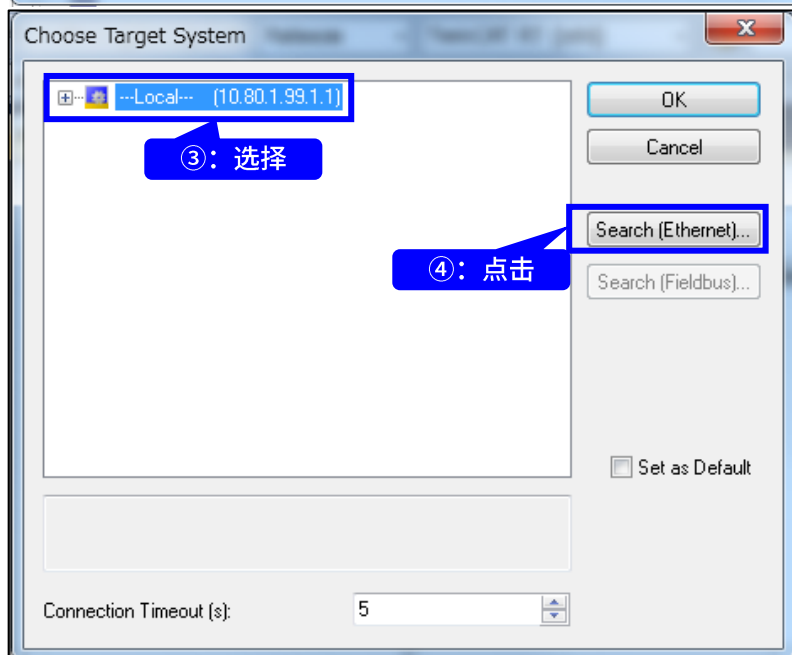
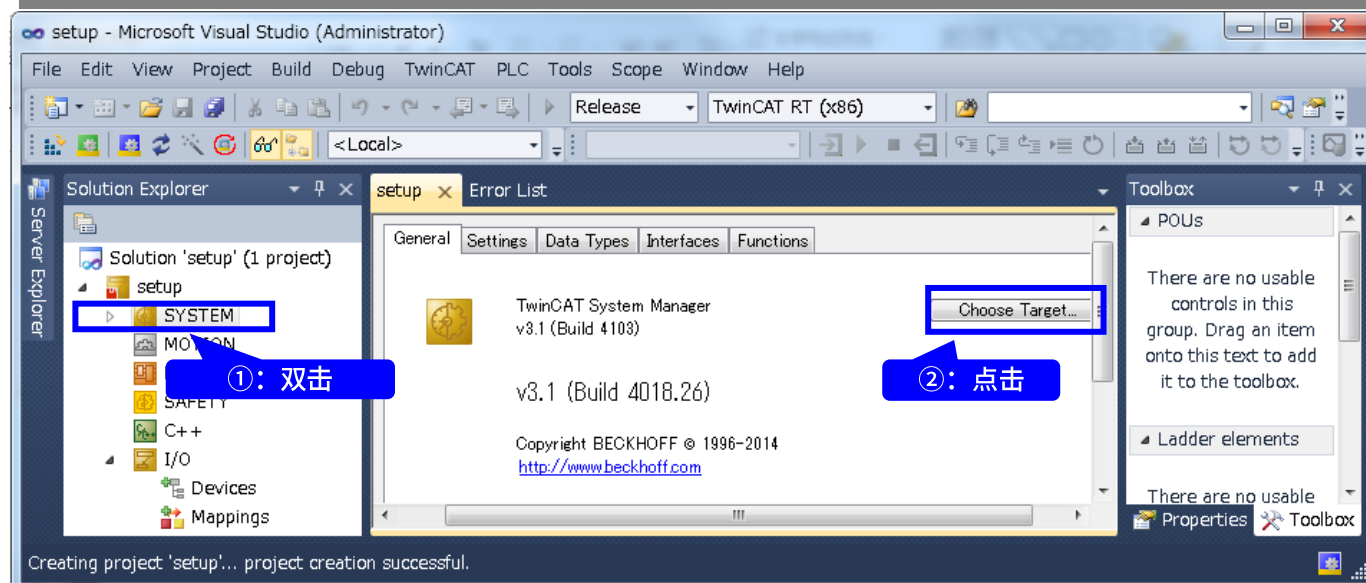


■ 文件保存后的页面



PC与EPC的通信设定

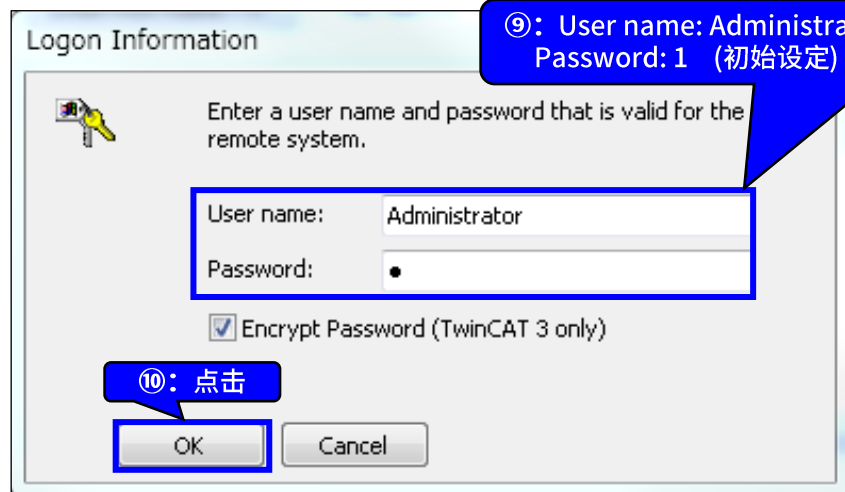
从TwinCAT3检测EPC



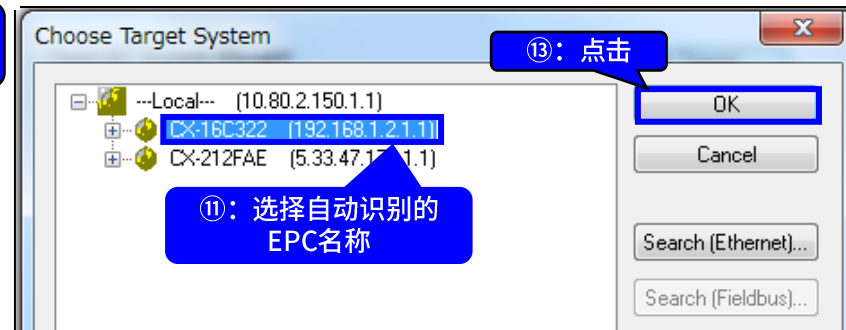
PC与EPC的通信设定

将检测出的EPC登录到Route

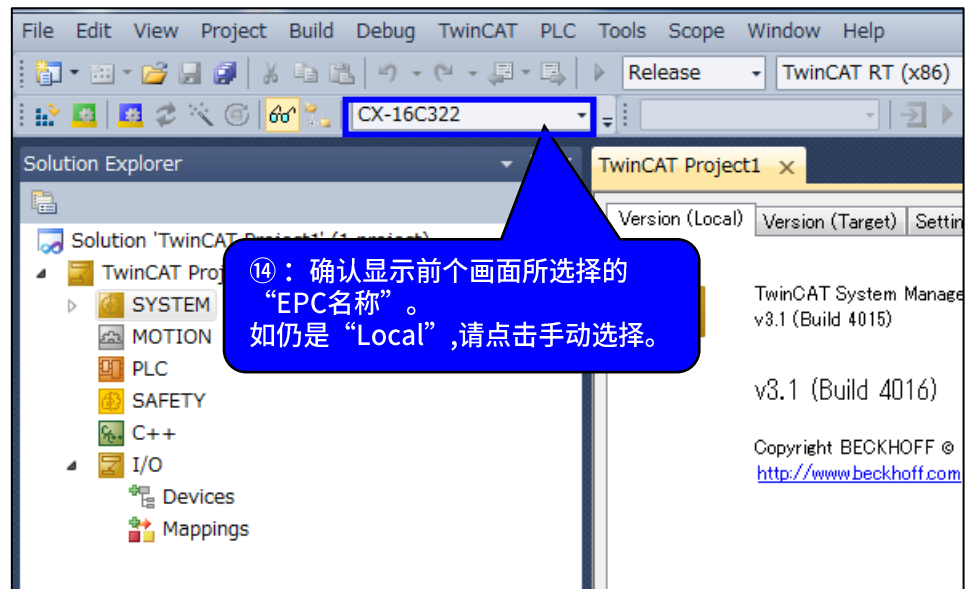
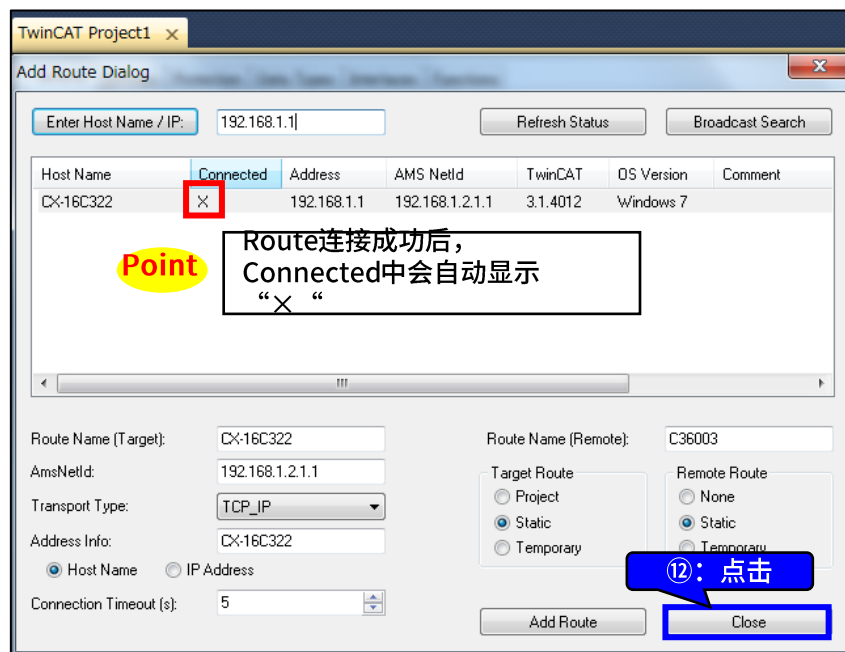
■按Add Route后的画面



■选择Target System



■Target System的确認



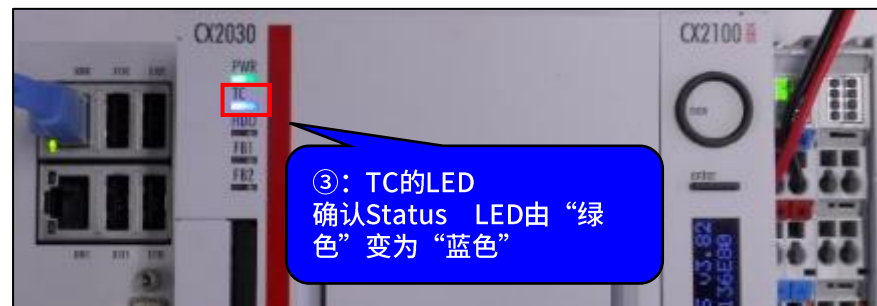
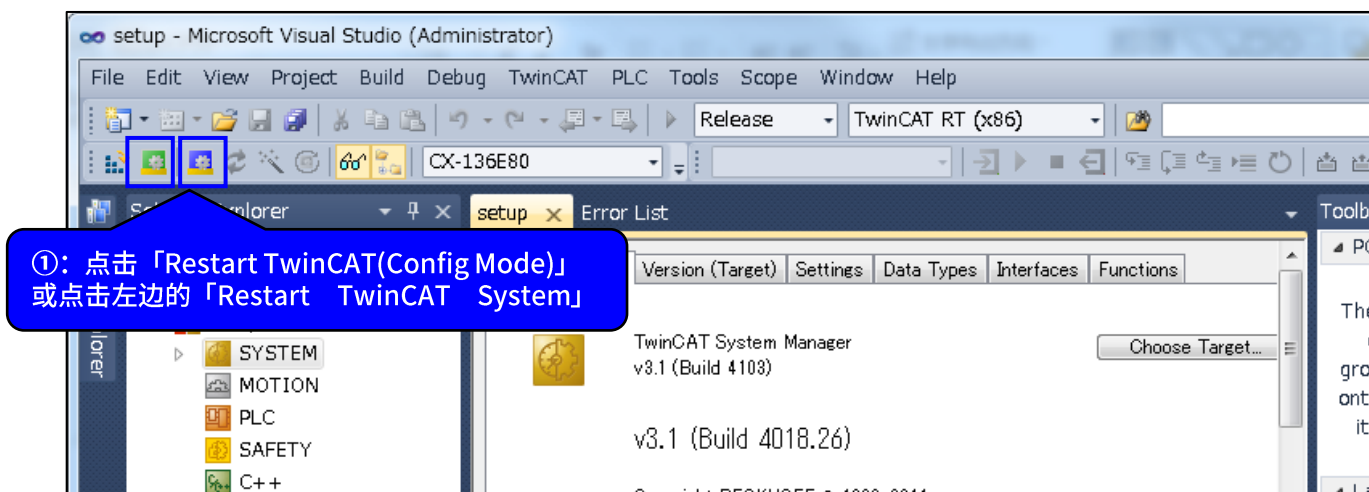
PC与EPC的通信设定

通过EPC的LED确认通信状态

PC与EPC之间的通信可通过EPC的TC的LED来确认

- 确认TC的LED的状态(绿：Run模式/蓝：Config模式)
- 点击「Restart TwinCAT (Config Mode)」 / 「Restart TwinCAT System」，切换Config模式 / Run模式
- 确认切换后的TC的LED的状态

【例：Run模式→Config模式】



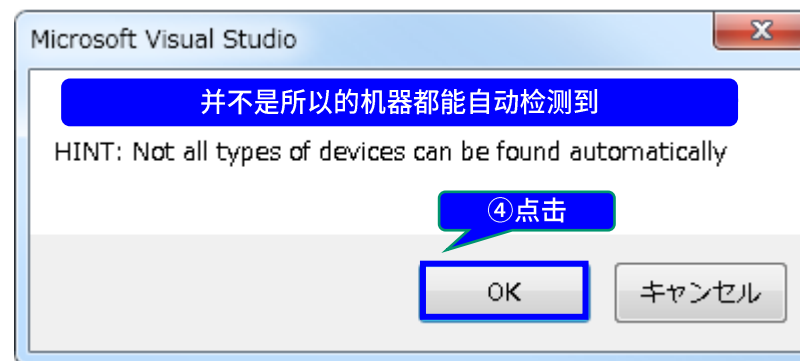
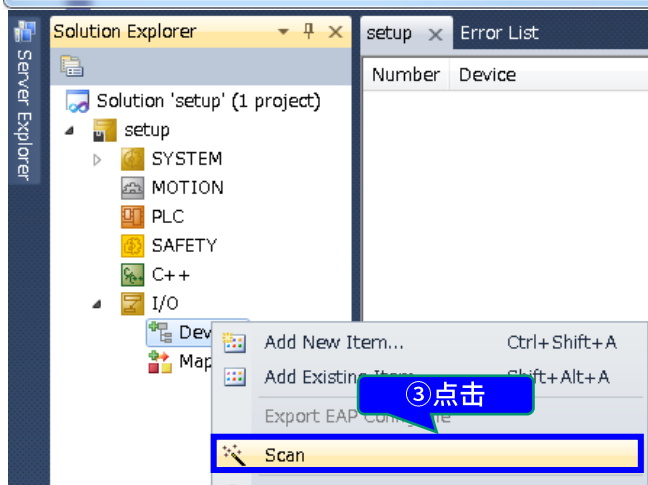
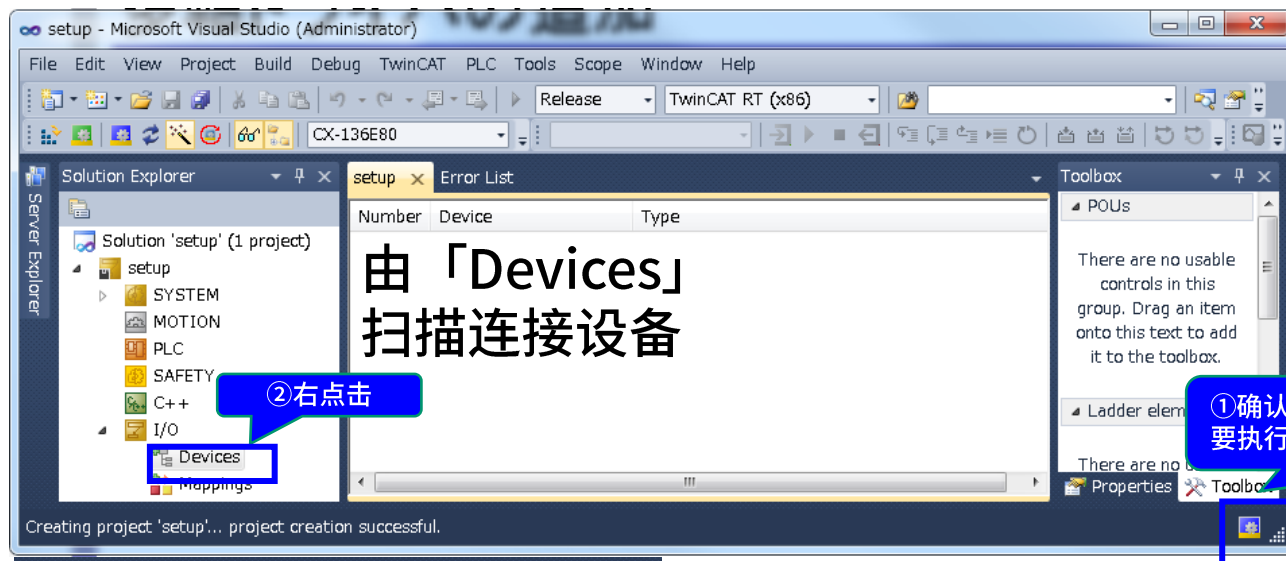
网络构成设定

EtherCAT检测连接设备

Target

自动检测连接到EtherCAT的设备并确认通信状态。

设定EtherCAT的网络构成。在此介绍如何在线检测设备、构成网络的步骤。



网络构成设定

EtherCAT检测连接设备

登录EtherCAT主站，并自动扫描从站设备。

⑥显示通过Scan检测出来的从站设备。
(按自动显示的结果，直接进行下一步⑦)

⑦：点击

⑤执行Scan后，
登录EtherCAT主站。

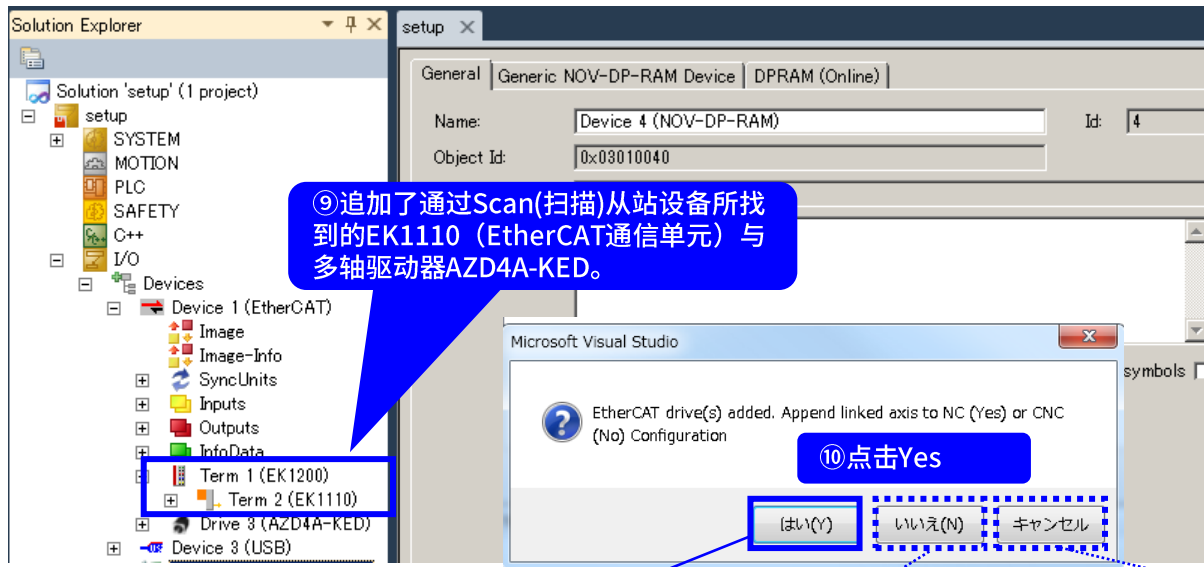
⑧：点击

接下来执行EtherCAT从站设备的Scan(扫描)。

网络构成设定

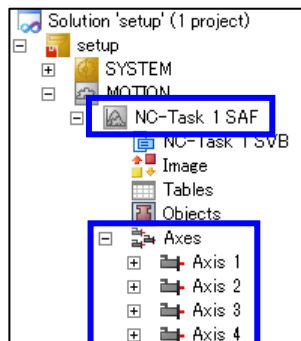
确认从站设备的轴登录

进行EtherCAT从站设备的登录。

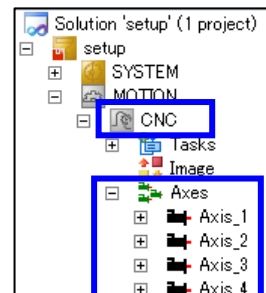


检测出多轴驱动器后，会询问是否要将轴追加到MOTION的NC或CNC。

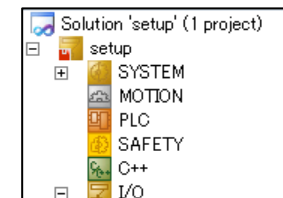
选择「Yes」，将轴登录到MOTION的NC。



选择「No」，将轴登录到MOTION的CNC。



选择「Cancle」，不进行MOTION的轴登录。



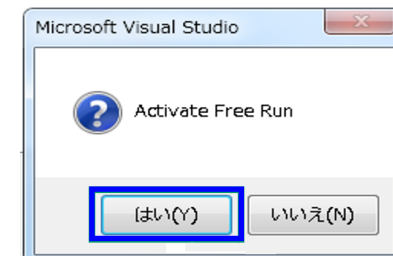
在PP/HM模式中，由于不进行Motion控制，因此不进行轴登录。

Motion控制，不进行轴登录。

网络构成设定

EPC和多轴驱动器的通信确认

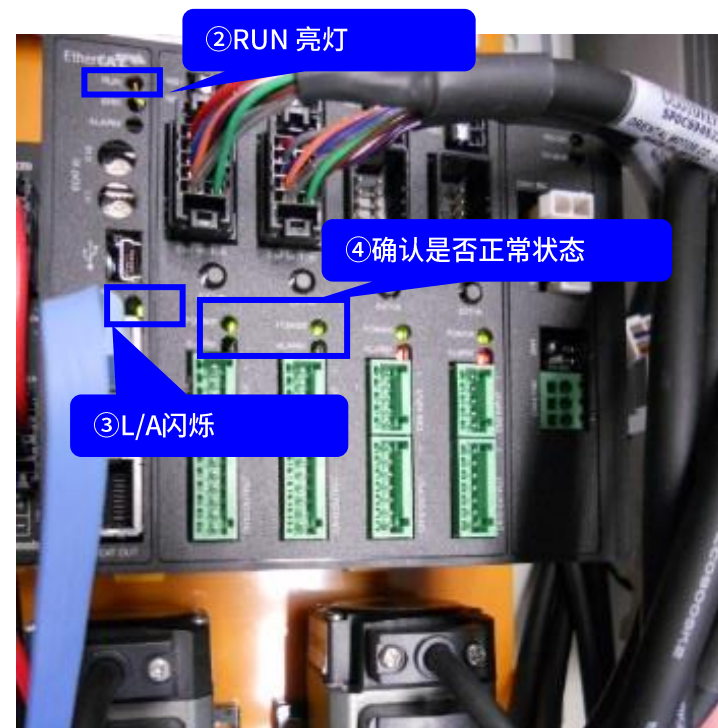
追加设备后,开始Active Free Run。
EPC与AZ驱动器的通信状态可通过EK1110和
驱动器的LED确认。



- EPC



- 多轴驱动器



④POWER LED: 亮灯(正常)
ALARM LED: 消灯(正常)

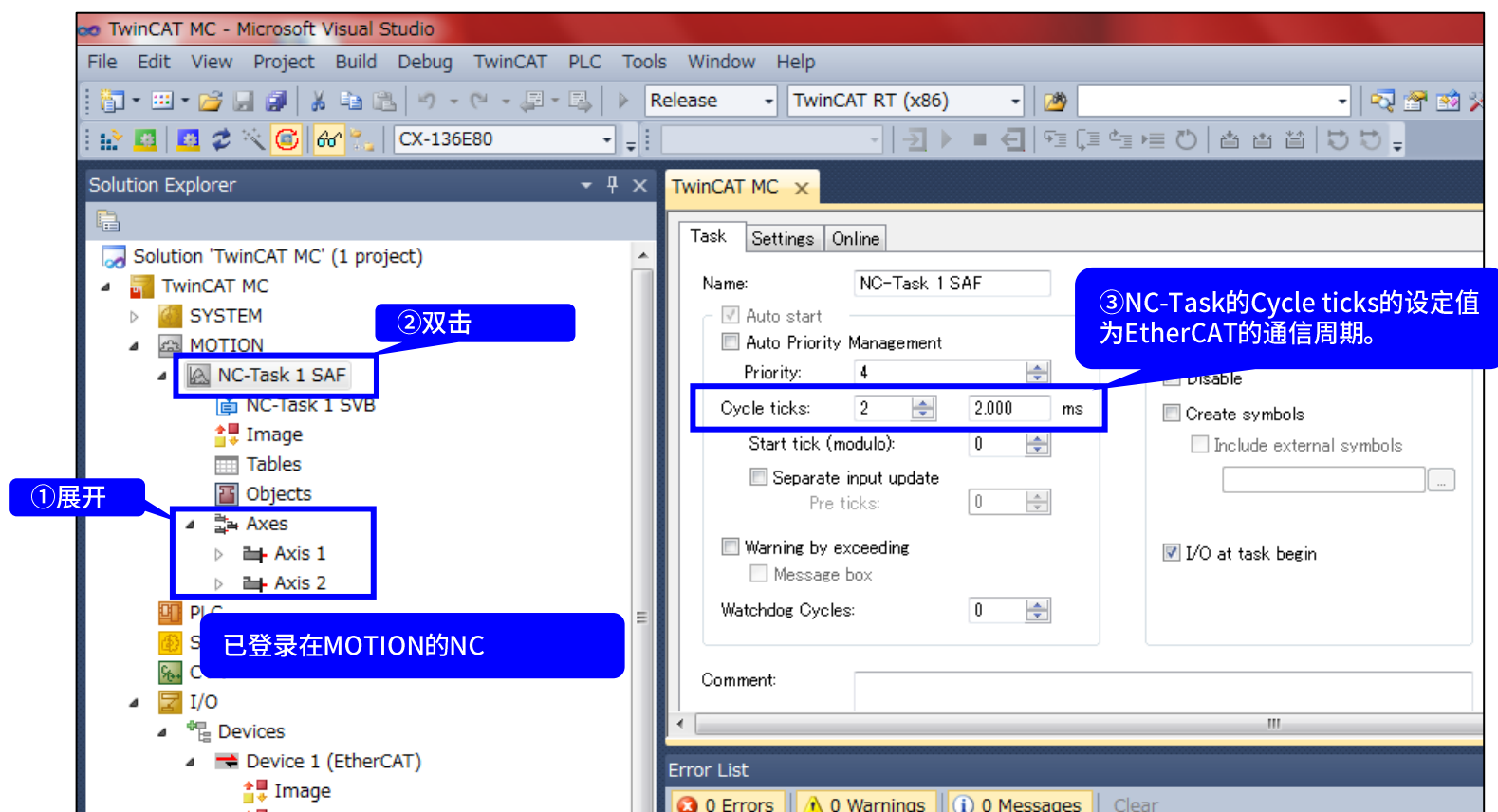
1. BECKHOFF TwinCAT 3 的通信设定

2. CSP模式的运行方法

运行前的各种设定

确认MOTION轴的登录

扫描从站设备后,确认轴已登录到MOTION的NC。另外EtherCAT通信周期(DC模式)为0.5ms,1~8ms。为此,MOTION要求Cycle ticks的设定小于8.0ms。在此保留默认设定2.0ms。



注意：请确认您使用的驱动器的所有轴都登录成功。

运行前的各种设定

MOTIONとI/Oのリンク登録

Target

进行CSP测试运行

确认在TwinCAT上的Axis 1，是否登录了AZ多轴#A。

LINK

① 点击

② 点击

③ 确认显示为“Drive3(AZD□A-KED)#A”

④ 确认单位的设定

⑤ 与Axis1同样方法确认“Link To I/O”。
使用 4 轴驱动器时的显示情况如下。
Axis2:Drive3(AZD4A-KED)#B
Axis3:Drive3(AZD4A-KED)#C
Axis4:Drive3(AZD4A-KED)#D

Unit: mm

Display (Only)

Position: ☐ μm ☐ Modulo

Velocity: ☐ mm/min

Result

Position: mm Velocity: mm/s Acceleration: mm/s² Jerk: mm/s³

Axis Cycle Time / Access Divider

1 0

Cycle Time (ms): 2.000

0 Messages

Description

MOTION运行时需要设定操作模式(Mode of operation)。

注意：MOTION不会自动设定。

操作模式(Mode of operation)的设定方法有以下两种。

- Startup设定
- PLC程序设定

本手册介绍在Startup下设定的方法、在Cyclic同步位置模式(CSP)下使用梯形图和功能块的简易运行。

Operation mode 的设定值	Operation mode
0 (初始值)	运行功能无效
1	Profile 位置模式 (PP)
3	Profile 速度模式 (PV)
6	原点返回模式 (HM)
8	Cyclic 同步位置模式 (CSP)
9	Cyclic 同步速度模式 (CSV)

运行前的各种设定

从PDO映射中删除操作模式

默认设定下，“Mode of operation”已登录在PDO mapping中。
不事先将其从PDO mapping中删除，则操作模式(Mode of operation)将被覆盖为0。

① 点击

② 点击

③ 点击
[PDO index]
Axis1: 0x1600 Axis2: 0x1610
Axis3: 0x1620 Axis4: 0x1630

④ 选择右击

⑤ 选择

Process Data

Sync Manager:

SM	Size	Type	Flags
0	128	MbxO...	
1	128	MbxIn	
2	30	Output...	
3	30	Inputs	

PDO List:

Index	Size	Name
0x1B00	2.0	Controlword transmit PDO mapping 1
0x1600	7.0	Axis1 Receive PDO mapping 1
0x1601	11.0	Axis1 Receive PDO mapping 2
0x1602	7.0	Axis1 Receive PDO mapping 3
0x1603	0.0	Axis1 Receive PDO mapping 4
0x1610	7.0	Axis2 Receive PDO mapping 1
0x1611	11.0	Axis2 Receive PDO mapping 2

PDO Assignment (0x1C12):

Index	Size	Offs	Name
0x1600	7.0	0.0	Controlword
0x1601	11.0	2.0	Target position
0x1602	7.0	4.0	Target velocity
0x1603	0.0	6.0	Modes of operation
0x1610	7.0	8.0	Axis1 position
0x1611	11.0	10.0	Axis1 velocity

Download

☒ PDO Assignment

☒ PDO Configuration

Predefined PDO Assignment: (none)

Load PDO info from device

Sync Unit Assignment...

Delete...

运行前的各种设定

通过Startup设定操作模式

从“Process Data”删除“Modes of operation”后，
通过“Startup”直接设定“Modes of operation”的参数。

① 点击

② 点击

③ 选择

④ 输入「08」（08: CSP模式）

⑤ 点击

“Modes of operation”一览

Operation mode 的设定值	Operation mode
0 (初始值)	运行功能无效
1	Profile 位置模式 (PP)
3	Profile 速度模式 (PV)
6	原点返回模式 (HM)
8	Cyclic 同步位置模式 (CSP)
9	Cyclic 同步速度模式 (CSV)

⑥ 确认Axis1的“Mode of Operation”登录在STARTUP中

⑦ 使用同样方法设定其他轴。

Axis1:0x6060, Axis2:0x6860, Axis3: 0x7060 , Axis4:0x7860

运行前的各种设定

运行参数设定

设定动作运行、原点返回运行时的速度。
默认设定的运行速度太快,请变更为适当的速度。

① 点击

② 点击

③ 点击展开

④ 设定为右边的值

⑤ 关于设定值的单位, 请按以下方法确认。

⑥ 点击展开

⑦ 点击

⑧ 1step=0.0001mm。按照这个设定, 可以换算电动机速度。
例: 「Manual Velocity(fast)为15[mm/s]时, 实际的电动机速度为15/0.0001=150[kHz]。」

⑨ 同样的方法设定其他轴

Parameter	Offline Value	Online Value	Type	Unit
Maximum Dynamics:				
+ Manual Motion and Homing:				
Homing Velocity (towards plc cam)	30.0	15.0	F	mm/s
Homing Velocity (off plc cam)	30.0	15.0	F	mm/s
Manual Velocity (Fast)	600.0	15.0	F	mm/s
Manual Velocity (Slow)	100.0	10.0	F	mm/s
Jog Increment (Forward)	5.0		F	mm

Parameter	Value
Encoder Evaluation:	
Invert Encoder Counting	
Scaling Factor Numerator	0.0001
Scaling Factor Denominator (default: 1.0)	1.0
Position Bias	0.0
Modulo Factor (e.g. 360.0°)	360.0

运行前的各种设定

执行Configuration

为了保存PDO和Startup等的参数设定，执行“Activate Configuration”。

① 点击
执行Activate Configuration,
变更为Run Mode。

② 点击

③ 点击

④ 点击

⑤ 请确认“Config Mode” (青)
变为“Run Mode” (绿)。

Ready

运行前的各种设定

Online页面说明

Motion轴的Online，执行测试运行。

① 点击

② 点击

位置，速度信息

警报信息

设定电动机的当前ON/OFF、CW许可、CCW许可的区域

功能区域和各功能的说明

Function	Description
F1	Reverse travel with Manual Velocity (Fast)
F2	Reverse travel with Manual Velocity (Slow)
F3	Forward travel with Manual Velocity (Slow)
F4	Forward travel with Manual Velocity (Fast)
F5	Start a direct travel command •Enter the Target Position •Enter the Target Velocity •Start the travel command with F5
F6	Stop a direct travel command
F8	Reset the control (if hand control has stopped responding)
F9	Trigger homing (see TwinCAT documentation)

运行前的各种设定

Axis1测试运行

“Online” 进行测试运行。

setup - Microsoft Visual Studio (Administrator)

File Edit View Project Build Debug TwinCAT PLC Tools Scope Window Help

Release TwinCAT RT (x86)

CX-136E80

Solution Explorer

- Tables
- Objects
- Axes
 - Axis 1
 - Axis 2
 - Enc
 - Drive
 - Ctrl
 - Axis 3
 - Axis 4
- PLC
- SAFETY
- C++
- I/O
- Devices
 - Device 1 (EtherCAT)
 - Image
 - Image-Info
 - SyncUnits
 - Inputs
 - Outputs
 - InfoData

General Settings Parameter Dynamics Online Functions Coupling Compensation

830.8455

Actual Velocity: [mm/s] -9.9910

Setpoint Position: [mm] 830.7518

Setpoint Velocity: [mm/s] -10.0000

Lag Distance (min/max): [mm] -0.0937 (-0.094, 0.000)

Override: [%] 100.0000

Total / Control Output: [%] -0.46 / -0.00

Error:

Status (log)

- ☒ Ready
- ☐ NOT Moving
- ☐ Calibrated
- ☐ Moving Fw
- ☒ Has Job
- ☒ Moving Bw

Status (phys)

- ☐ Coupled Mode
- ☐ In Target Pos.
- ☐ In Pos. Range

Enabling

- ☒ Controller
- ☒ Feed Fw
- ☒ Feed Bw

Controller Kv-Factor: [mm/s/mm] 1

Reference Velocity: [mm/s] 2200

Target Position: [mm] 0

Target Velocity: [mm/s] 0

F1 F2 F3 F4 F5 F6 F8

⑤ 确认按设定速度(10mm/s)运行

① 点击

Set

② 点击

③ 再按以下“Set”，确认以下内容设定成功

④ 长按F2键 (Reverse Travel with Manual Velocity(Slow)), 进行测试运行

注意：程序执行中无法进行测试运行

运行前的各种设定

附：搭载DG2-AZ产品/减速机组使用时的循环坐标设定参考

说明：

由于BECKOFF PLC的循环坐标设定范围有限制，必须设定为2的n次方，AZ驱动器的循环设定坐标也必须设定为2的n次方。

不然BECKOFF PLC和AZ驱动器的坐标不一致，导致BECKOFF PLC出现位置偏差过大警报(0x4550)，或AZ驱动器出现脉冲指令异常警报(34h)；

不同减速比设定数值参考如下表(表一)：

减速比	电子减速机A 6091h-01h	电子减速机B 6091h-02h	马达分辨率 [step/rev]	减速机输出轴的分辨率	十六进制
				[step/rev]	
1	625	512	8192	8192	00002000h
3.6	1125	1024	9102.22	32768	00008000h
5	3125	2048	6553.60	32768	00008000h
7.2	1125	1024	9102.22	65536	00010000h
9	5625	4096	7281.78	65536	00010000h
10	3125	2048	6553.60	65536	00010000h
12	1875	1024	5461.33	65536	00010000h
15	9375	8192	8738.13	131072	00020000h
18	5625	4096	7281.78	131072	00020000h
20	3125	2048	6553.60	131072	00020000h
25	15625	8192	5242.88	131072	00020000h
30	9375	8192	8738.13	262144	00040000h
36	5625	4096	7281.78	262144	00040000h
50	15625	8192	5242.88	262144	00040000h
100	15625	8192	5242.88	524288	00080000h
120	9375	8192	8738.13	1048576	00100000h

运行前的各种设定

附：搭载DG2-AZ产品/减速机组合使用时的循环坐标设定参考

AZ多轴_Beckhoff_EtherCAT_MC_B

AZ驱动器侧设定：

- 设定条件：① 马达/驱动器参数恢复出厂设定数值后开始设定；
② 以DG2-AZ垂直安装面产品（搭载18比减速机）为例；
③ 以“脉冲”为单位设定。

设定内容1：参考上页表一，利用MEXE02软件修改电子减速机A、B数值：

(a5) 屏幕清单

当前打开的屏幕

(p1) Profile area的对象

参数

参数

(p1) Profile area的对象

厂家固有区域的对象

(p2) 基本设定

(p3) 电动机・机构(坐标/JOG/原点返回)设定

(p4) ETO・Alarm・Info设定

(p5) I/O动作・功能

(p6) Direct-IN 功能选择(DIN)

(p7) Direct-OUT功能选择(DOUT)

(p8) Remote-I/O功能选择(R-I/O)

(p9) EXT-IN・VIR-IN・USR-OUT功能选择(扩展)

(p10) I/F功能

(p1) Profile area的对象

1	Quick stop option code (605Ah)	按照Quick stop deceleration(6085h)减速停止。停止后，变为Switch on disabled。
2	Shutdown option code (6058h)	按照Profile deceleration(6084h)减速停止。停止后，电动机变为无励磁。
3	Disable operation option code (605Ch)	按照Profile deceleration(6084h)减速停止。停止后，电动机变为无励磁。
4	Halt option code(605Dh)	按照Profile deceleration(6084h)减速停止。停止后，保持Operation enabled。
5	Modes of operation (6060h)	运行功能无效
6	Following error window (6065h)	3.00 rev
7	Position window (6067h)	1.8 °
8	Home offset (607Ch)	0 step
9	Min position limit (607Dh-01h)	-2147483648 step
10	Max position limit (607Dh-02h)	2147483647 step
11	Profile velocity (6081h)	10000 Hz
12	Profile acceleration (6083h)	300000 step/s^2
13	Profile deceleration (6084h)	300000 step/s^2
14	Quick stop deceleration (6085h)	1000000 step/s^2
15	Electronic gear A (6091h-01h)	5625
16	Electronic gear B (6091h-02h)	4096
17	Homing method (6098h)	利用原点传感器 (HOMES)执行原点返回，向正方向起动
18	Speed during search for switch (6099h-01h)[Hz]	10000 Hz
19	Speed during search for zero (6099h-02h)[Hz]	5000 Hz
20	Homing acceleration (609Ah)	300000 step/s^2

运行前的各种设定

附：搭载DG2-AZ产品/减速机组合使用时的循环坐标设定参考

AZ多轴_Beckhoff_EtherCAT_MC_B

设定内容2: DG2-AZ垂直安装面产品搭载18比减速机, 修改电机循环范围为18rev , 循环偏置比例为0%:

▽ 当前打开的屏幕

(p3) 电动机・机构(坐标/JOG/原点返回)设定

(p1) Profile area的对象

▽ 参数

参数

(p1) Profile area的对象

厂家固有区域的对象

(p2) 基本设定

(p3) 电动机・机构(坐标/JOG/原点返回)设定

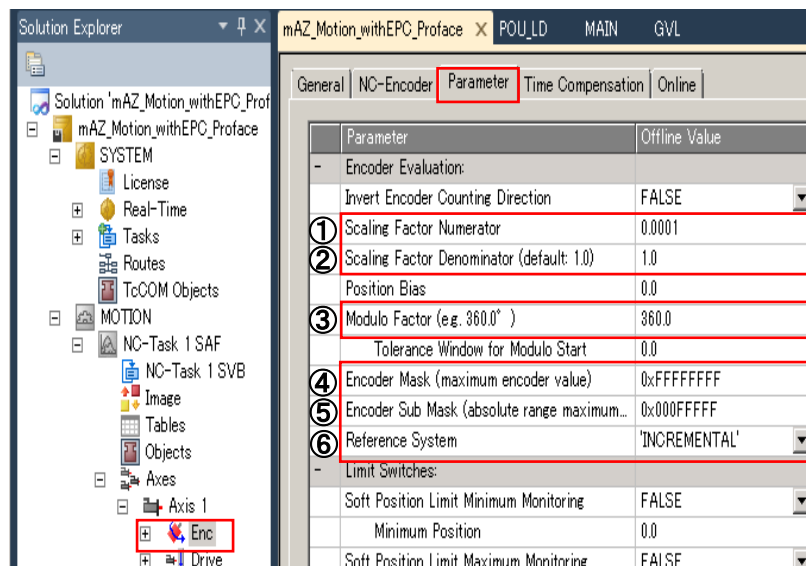
(p4) ETO・Alarm・Info设定

1	机构各条件设定	手动设定
2	电子减速机A	Electronic gear A (6091h-0...
3	电子减速机B	Electronic gear B (6091h-0...
4	电动机旋转方向	+侧=CW
5	减速比设定	0.00
6	初始坐标生成/循环坐标设定	手动设定
7	初始坐标生成/循环设定范围	18.0 rev
8	初始坐标生成/循环偏置比率设定	0.00 %
9	初始坐标生成/循环偏置值设定	0 step
10	循环设定	有效
11	RND-ZERO输出用RND分割数	1

运行前的各种设定

附：搭载DG2-AZ产品/减速机组使用时的循环坐标设定参考

BECKOFF PLC設定(必须)



请设定如下：

① Scaling Factor Numerator (分子) :0.0001

② Scaling Factor Denominator (分母) :1.0

③ Modulo Factor: 13.1072

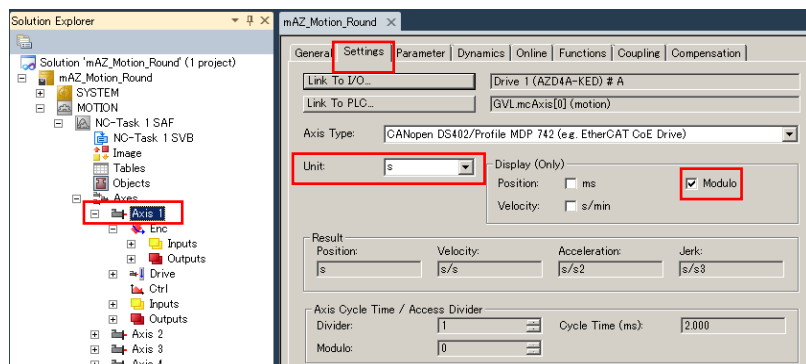
(参考表一“减速机输出轴的分辨率”数值, 由于上述①为0.0001, 因此设定“13.1072”实际上是“131072”。)

④ Encoder Mask (maximum encoder value) :0x0001FFFF

⑤ Encoder Sub Mask (absolute range maximum value) :
0x0001FFFF

⑥ Reference System: 'ABSOLUTE(modulo)'

※ 十六进制数0001FFFF=十进制数131071;
虽然 $2^{17}=131072$, 设定时需要按照 $(2^n)-1=131071$ 进行设定, 这是上位倍福的设定规则。



unit: s (设定例为STEP单位)

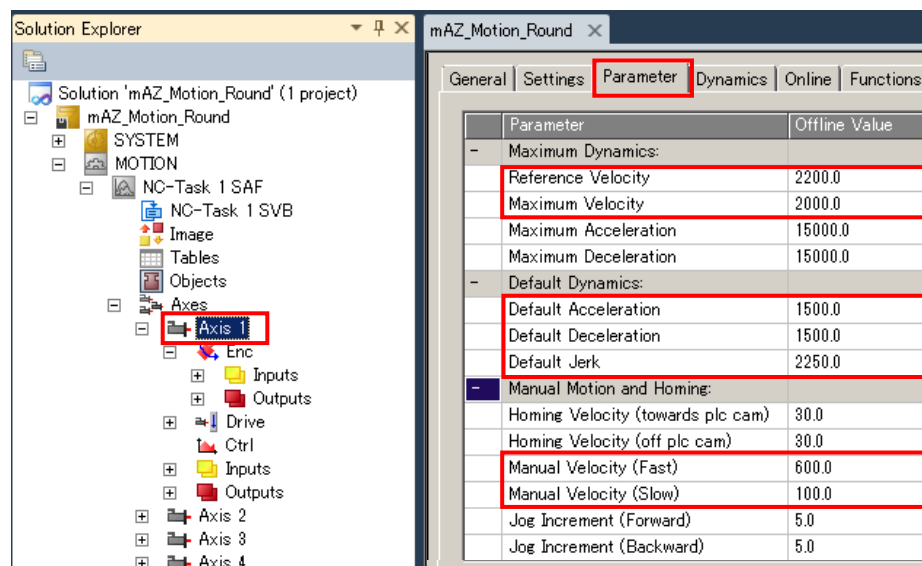
Modulo: 打勾

运行前的各种设定

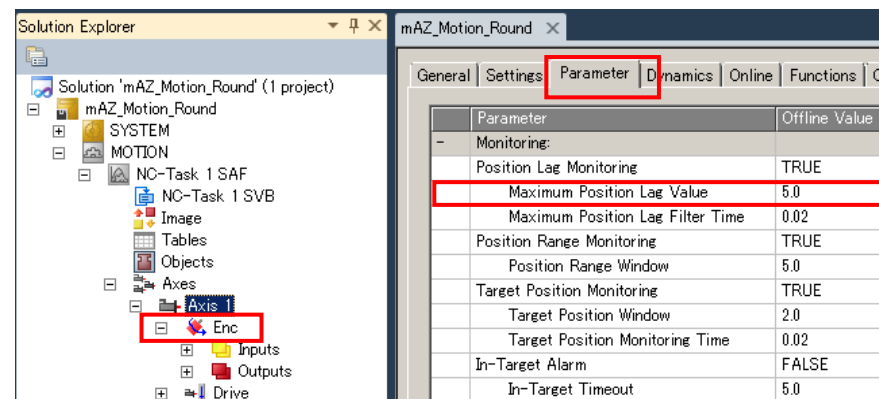
附：搭载DG2-AZ产品/减速机组使用时的循环坐标设定参考

BECKOFF PLC設定(注意)

修改上述参数后，会影响速度、加减速设定，以及警报的输出条件等；
请根据需要对其它参数进行修改(非必要)：



Parameter	Offline Value
Maximum Dynamics:	
Reference Velocity	2200.0
Maximum Velocity	2000.0
Maximum Acceleration	15000.0
Maximum Deceleration	15000.0
Default Dynamics:	
Default Acceleration	1500.0
Default Deceleration	1500.0
Default Jerk	2250.0
Manual Motion and Homing:	
Homing Velocity (towards plc cam)	30.0
Homing Velocity (off plc cam)	30.0
Manual Velocity (Fast)	600.0
Manual Velocity (Slow)	100.0
Jog Increment (Forward)	5.0
Jog Increment (Backward)	5.0

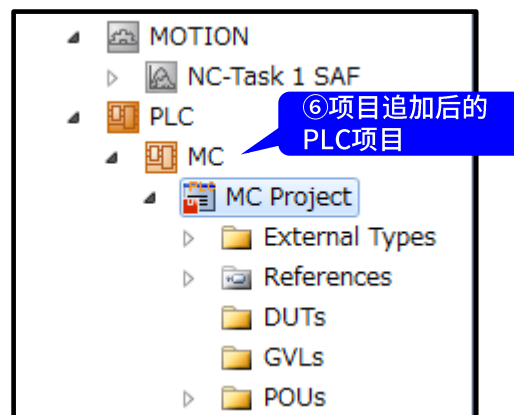
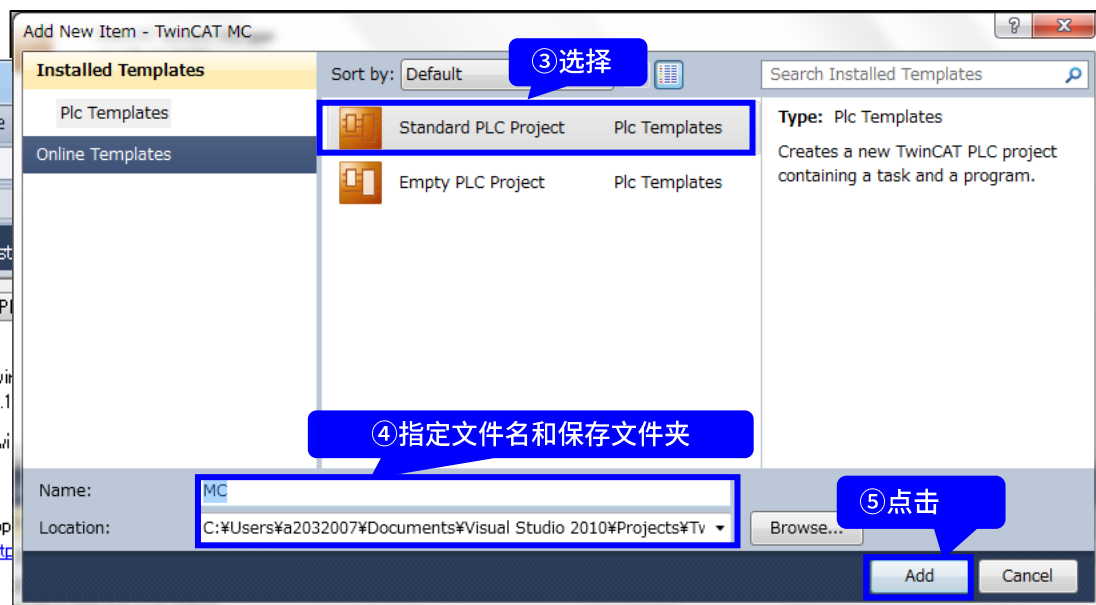
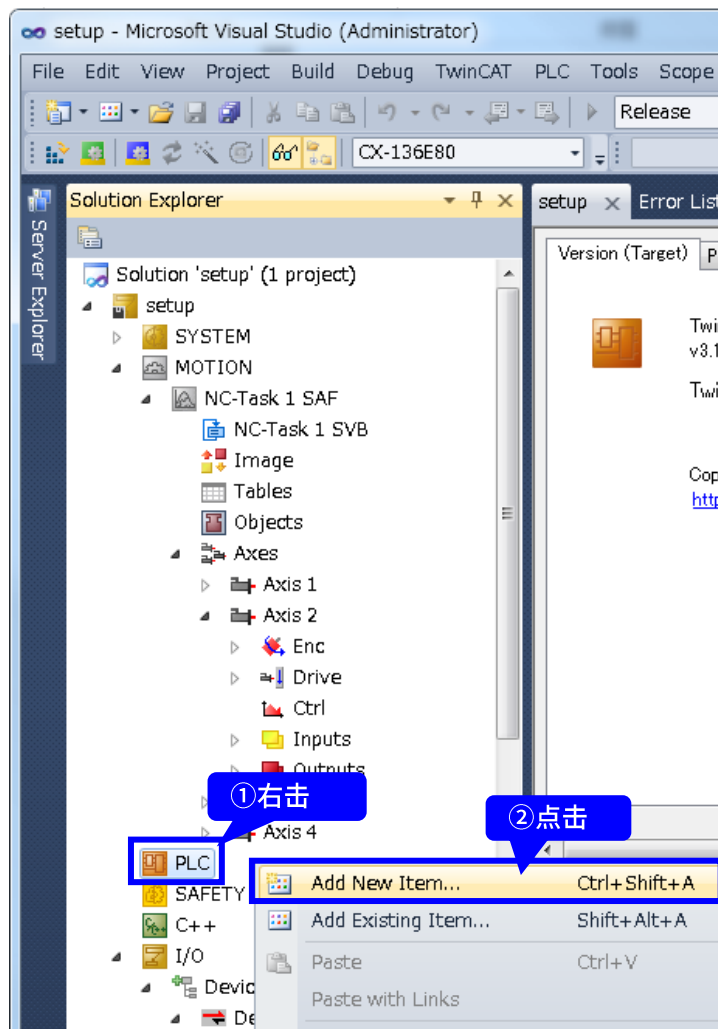


Parameter	Offline Value
Monitoring:	
Position Lag Monitoring	TRUE
Maximum Position Lag Value	5.0
Maximum Position Lag Filter Time	0.02
Position Range Monitoring	TRUE
Position Range Window	5.0
Target Position Monitoring	TRUE
Target Position Window	2.0
Target Position Monitoring Time	0.02
In-Target Alarm	FALSE
In-Target Timeout	5.0

Target

制作MOTION运行的程序

PLC追加项目



PLC程序要使用电动机控制用的功能块(FB),追加“TC2_MC2”的Library。

① 展开右击

② 点击

③ 展开

④ 展开

⑤ 点击

⑥ 点击

⑦ 确认Tc2_MC2成功登录到Library

TwinCAT MC - Microsoft Visual Studio

File Edit View Project Build Debug TwinCAT PLC Tools Window Help

Release TwinCAT RT (x)

CX-136E80 MC

Solution Explorer

Solution 'TwinCAT MC' (1 project)

- TwinCAT MC
 - SYSTEM
 - MOTION
 - NC-Task 1 SAF
 - PLC
 - MC
 - MC Project
 - External Types
 - References
 - Tc2_MC2
 - Tc2_Standard
 - Tc2_System
 - Tc3_Interfaces
 - Tc3_Module

Version (Target) Plc Settings

TwinCAT System Manager v3.1 (Build 4103)

TwinCAT PLC Server

Copyright BECKHOFF © 1996-2014

Add Library

Enter a string for a fulltext search in all libraries...

Library	Company
Communication	
Controller	
DataAccess	
Intern	
IO	
Math	
Measurement	
Motion	
Camming	
CNC	
Drive	
FifoAxes	
FlyingSaw	
NCI	
PTP	
Tc2_MC2	Beckhoff Automation GmbH
Tc2_NC	Beckhoff Automation GmbH

Advanced... OK Cancel

程序制作

新建POU(程序)

追加梯形图程序(LD)制作用POU。

The screenshot illustrates the process of creating a new POU in the TwinCAT MC environment. The main window shows the Solution Explorer on the left, the main editor in the center, and the Error List at the bottom. The Solution Explorer shows the project structure, including the 'POU' folder. The main editor shows the 'Add' context menu for the 'POU' folder, with the 'POU...' option highlighted. The 'Add POU' dialog box is open, showing the 'Name' field set to 'POU' and the 'Type' set to 'Program'. The 'Implementation language' is set to 'Ladder Logic Diagram (LD)'. The 'Open' button is highlighted.

①展开右击

②点击

③点击

④输入名称

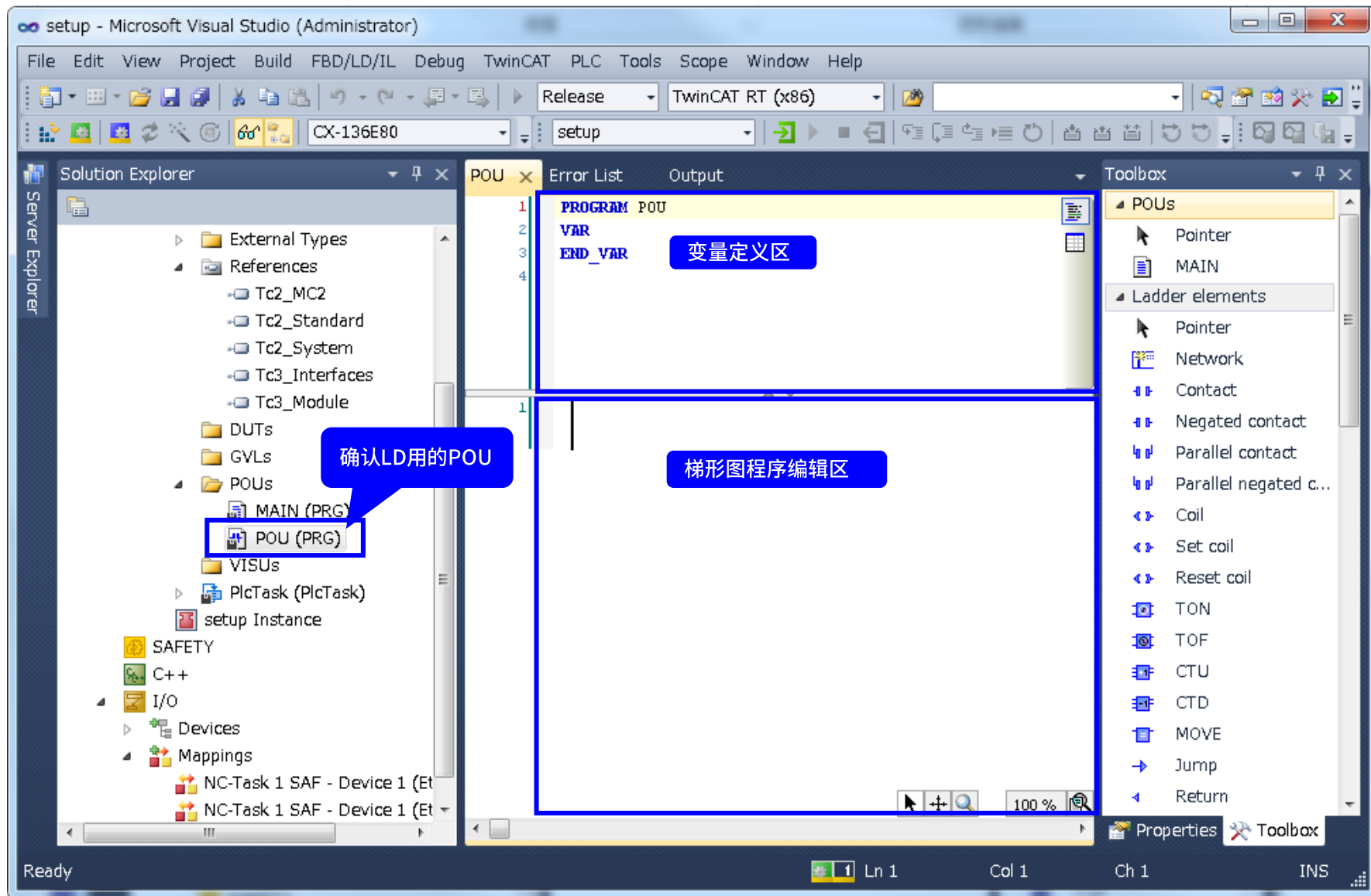
⑤选择“Ladder Logic Diagram(LD)”

⑥点击

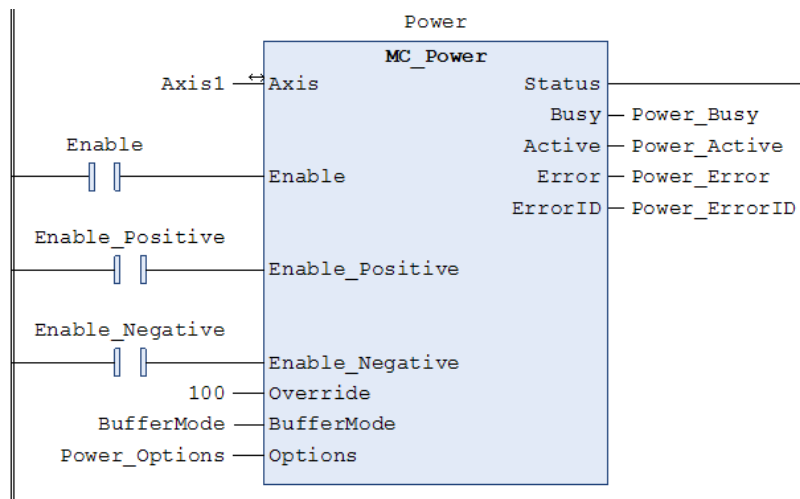
程序制作

POU页面说明

追加的LD用POU如下所示。

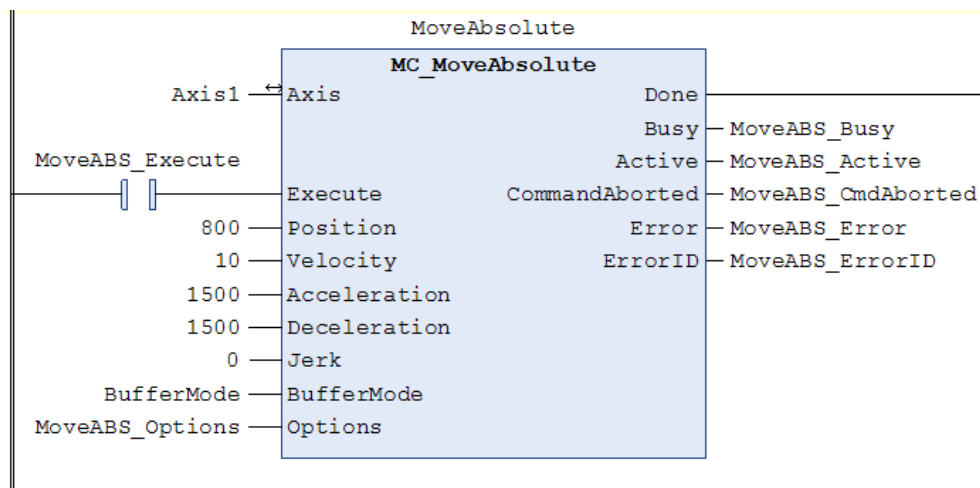


MC运行,利用MC_Power的FB进行电动机的当前ON/OFF。
同时,本书中,使用MC_MoveAbsolute的FB进行绝对定位运行。



MC_Power的功能块

【FB说明】：用Axis设定轴, Enable、Enable_Positive、Enable_Negative为TRUE(ON)时,电动机励磁。



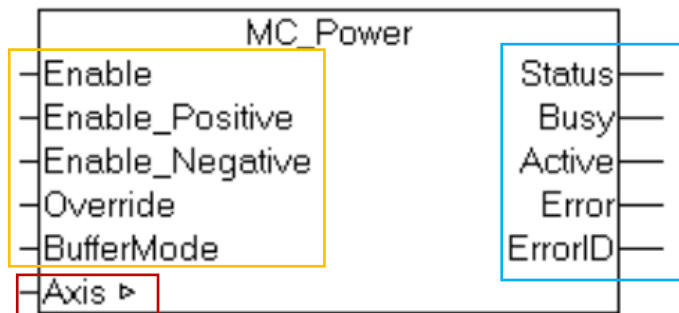
MC_MoveAbsolute的功能块

【FB说明】：用Axis设定轴, 设定运行参数(Position、Velocity · · ·), Execute为True(ON)时,开始运行。

Point

有关FB的详情,请参阅 Beckhoff Information System 。
https://infosys.beckhoff.com/index_en.htm

- MC_Power activates software enable for an axis. Enable can be activated for both directions of travel or only one direction. At Status output operational readiness of the axis is indicated.



Inputs

(本资料称之为“输入参数”。)

(本资料将Data structure of type成为“型”)

```
VAR_INPUT
    Enable          : BOOL; (* B *)
    Enable_Positive : BOOL; (* E *)
    Enable_Negative : BOOL; (* E *)
    Override        : LREAL (* V *) := 100.0; (* in percent - Beckhoff proprietary input *)
    BufferMode       : MC_BufferMode; (* V *)
END_VAR
```

Enable	General software enable for the axis.
Enable_Positive	Advance movement enable in positive direction. Only takes effect if Enable=TRUE.
Enable_Negative	Advance movement enable in negative direction. Only takes effect if Enable=TRUE.
Override	Velocity override in % for all movement commands. (0 ≤ Override ≤ 100.0)
BufferMode	The BufferMode is evaluated if Enable is reset. MC_Aborting mode leads to immediate deactivation of the axis enable. Otherwise, e.g. in MC_Buffered mode, the block waits until the axis no longer executes a command.

Outputs

(本资料称之为“输出参数”。)

```
VAR_OUTPUT
    Status : BOOL; (* B *)
    Busy   : BOOL; (* V *)
    Active : BOOL; (* V *)
    Error  : BOOL; (* B *)
    ErrorID : UDINT; (* E *)
END_VAR
```

Status	Status=TRUE indicates that the axis is ready for operation.
Busy	The Busy output is TRUE, as long as the function block is called up with Enable
Active	Active indicates that the command is executed
Error	Becomes TRUE if an error occurs.
ErrorID	If the error output is set, this parameter supplies the error number

Inputs/outputs

```
VAR_IN_OUT
    Axis : AXIS_REF;
END_VAR
```

The axis data structure of type AXIS_REF addresses an axis uniquely within the system. Among other parameters it contains the current axis status, including position, velocity or error status.

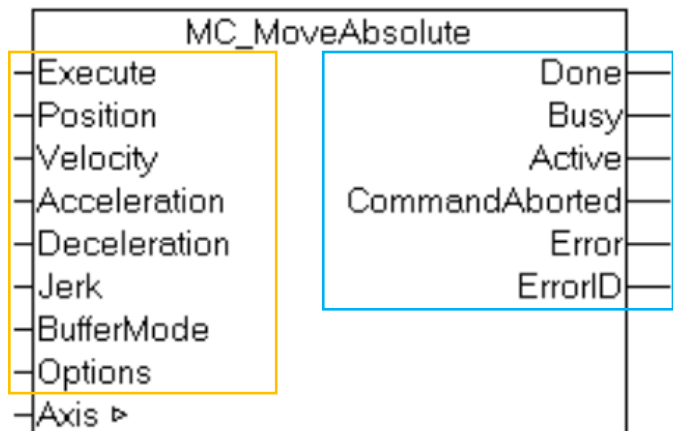
Notes:

In addition to software enable it may be necessary to activate a hardware enable signal in order to enable a drive. This signal is not influenced by MC_Power and must be activated separately by the PLC.

MC_MoveAbsolute详细说明

AZ多轴_Beckhoff_EtherCAT_MC_B

- MC_MoveAbsolute starts positioning to an absolute target position and monitors the axis movement over the whole travel path. The Done output is set once the target position has been reached. Otherwise the CommandAborted or, on error, the Error output is set.



Inputs

```
VAR_INPUT
    Execute           : BOOL;
    Position          : LREAL;
    Velocity          : LREAL;
    Acceleration      : LREAL;
    Deceleration      : LREAL;
    Jerk              : LREAL;
    BufferMode         : MC_BufferMode;
    Options            : ST_MoveOptions;
END_VAR
```

Execute	The command is executed with a rising edge at input <i>Execute</i> .
Position	Absolute target position to be used for positioning.
Velocity	Maximum travel velocity (>0).
Acceleration	Acceleration (≥0). If the value is 0, the standard acceleration from the axis configuration in the System Manager is used.
Deceleration	Deceleration (≥0). If the value is 0, the standard deceleration from the axis configuration in the System Manager is used.
Jerk	Jerk (≥0). If the value is 0, the standard jerk from the axis configuration in the System Manager is used.
BufferMode	The BufferMode is analysed, if the axis is already executing another command. The running command can be aborted, or the new command becomes active after the running command. The BufferMode also determines the transition condition from the current to the next command. If the command is passed to a slave axis just Buffermode Aborting can be used. A second instance of a Move function block is required to use the BufferMode. It is not possible to trigger the same function block with different parameters as long as it is busy.
Options	The data structure <i>Option</i> includes additional, rarely required parameters. The input can normally remain open.

Outputs

```
VAR_OUTPUT
    Done           : BOOL;
    Busy           : BOOL;
    Active          : BOOL;
    CommandAborted : BOOL;
    Error          : BOOL;
    ErrorID        : UDINT;
END_VAR
```

Done	The <i>Done</i> output becomes TRUE once the target position was reached.
Busy	The <i>Busy</i> output becomes TRUE when the command is started with <i>Execute</i> and remains TRUE as long as the motion command is processed. When <i>Busy</i> becomes FALSE again, the function block is ready for a new command. At the same time one of the outputs, <i>Done</i> , <i>CommandAborted</i> or <i>Error</i> , is set.
Active	Active indicates that the command is executed. If the command was queued, it becomes active once a running command is completed.
CommandAborted	Becomes TRUE, if the command could not be fully executed. The axis may have been stopped, or the running command may have been followed by a further Move command.
Error	Becomes TRUE if an error occurs.
ErrorID	If the error output is set, this parameter supplies the error number.

程序制作

MC_Power的设置

将空FB插入POU中。

The screenshot illustrates the steps to insert an empty function block (FB) into a Program Organization Unit (POU) in Microsoft Visual Studio:

- ①选择要插入FB的行(line)**: Select the line in the POU where the FB should be inserted.
- ②展开**: Expand the **FBD/LD/IL** menu in the top toolbar.
- ③点击**: Click on **Insert Empty Box** in the context menu.
- ④在所选行(line)上建立一个空FB**: An empty function block is created on the selected line in the POU.

The context menu options visible are:

- Paste below
- Paste above
- Paste right (after)
- View as function block diagram
- ☒ View as ladder logic
- View as instruction list
- Insert Network
- Insert Network (below)
- Insert Box
- Insert Empty Box**
- Insert Assignment
- Insert Coil
- Insert Set coil
- Insert Reset coil

The POU code in the editor shows:

```
1 PROGRAM POU
2 VAR
3 END_VAR
```

程序制作

MC_Power的设置

在空FB中设定MC_Power。

① 点击

② 点击

③ 选择

④ 在搜索栏中输入“MC_Power”

⑤ 点击检测到的“Tc2_MC2.MC_Power”

⑦ 制作MC_Power的FB功能块。

⑥ 点击

Input Assistant

Text search Categories

MC_Power

2 item(s) found.

Name	Type
Tc2_MC2.MC_Power	FUNCTION_BLOCK
Tc2_MC2.MC_PowerStepper	FUNCTION_BLOCK

Filter: None

☒ Insert with arguments ☐ Insert with namespace prefix

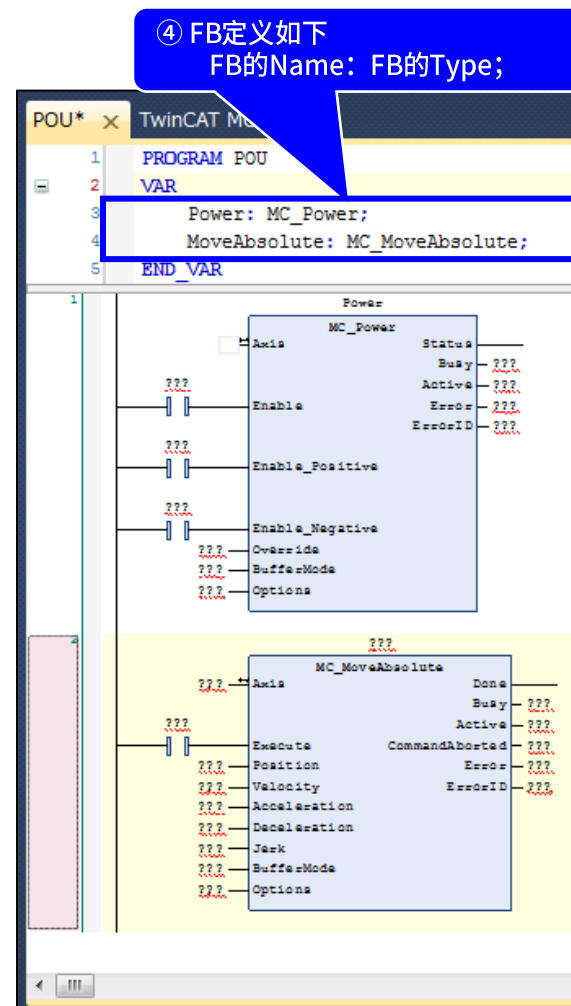
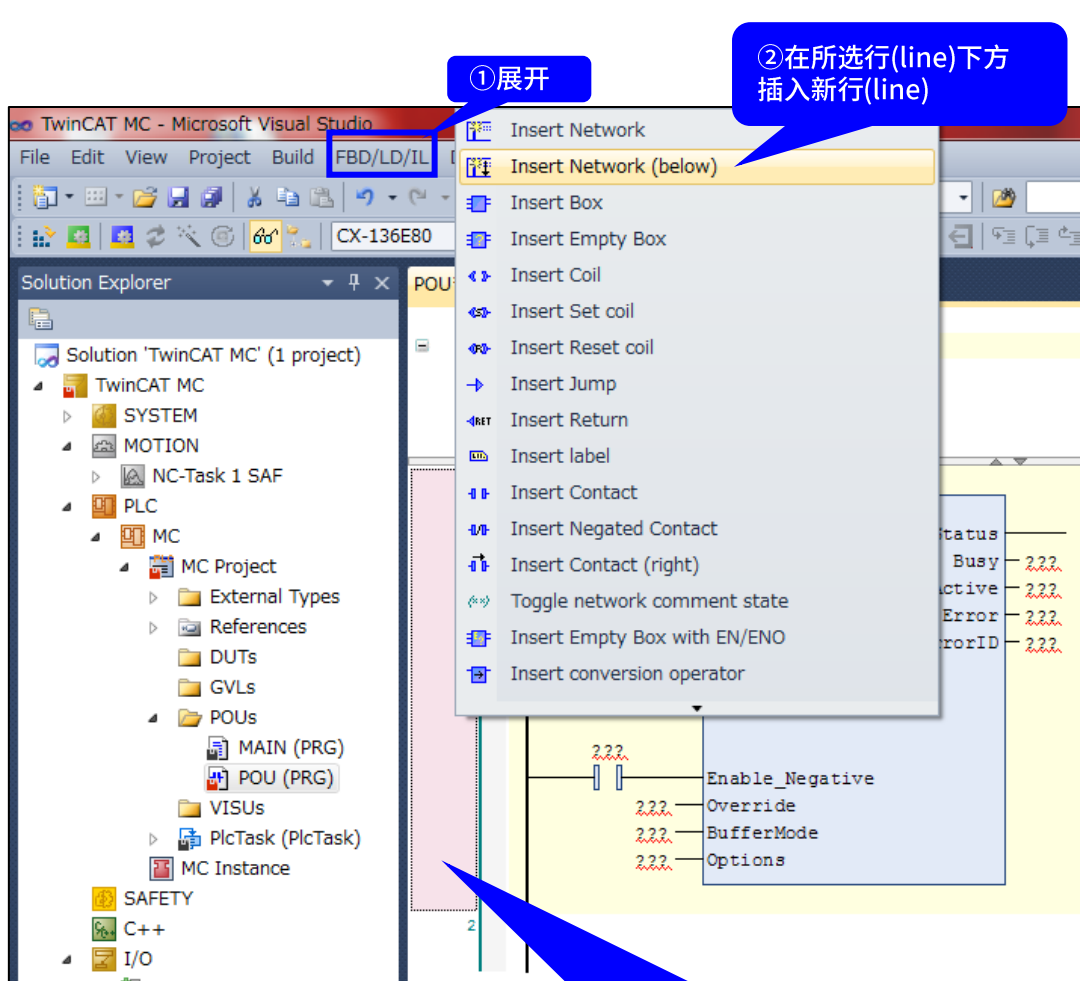
Documentation:

FUNCTION_BLOCK MC_Power

Axis	AXIS_REF	VAR_IN_OUT	B
Enable	BOOL	VAR_INPUT	B
Enable_Positive	BOOL	VAR_INPUT	E
Enable_Negative	BOOL	VAR_INPUT	E

OK Cancel

将新FB插入下一行。追加了两个FB后,定义各自的输入/输出参数。



程序制作

添加Global变量

追加FB输入/输出用的变量。

The screenshot shows the Microsoft Visual Studio (Administrator) interface with the 'setup' project open. The Solution Explorer on the left shows the project structure, including 'setup', 'SYSTEM', 'MOTION', and 'PLC'. The 'PLC' folder is expanded, showing 'setup' and 'POUs'. The 'GVLs' folder is highlighted, and a right-click context menu is open. The 'Add' option is selected, and the 'Global Variable List...' option is chosen from the submenu. The 'Add Parameter list' dialog box is open, showing the 'Name' field with 'GVLs' entered. The 'Open' button is highlighted.

① 点击

② 点击

③ 选择

④ 输入名称

⑤ 点击

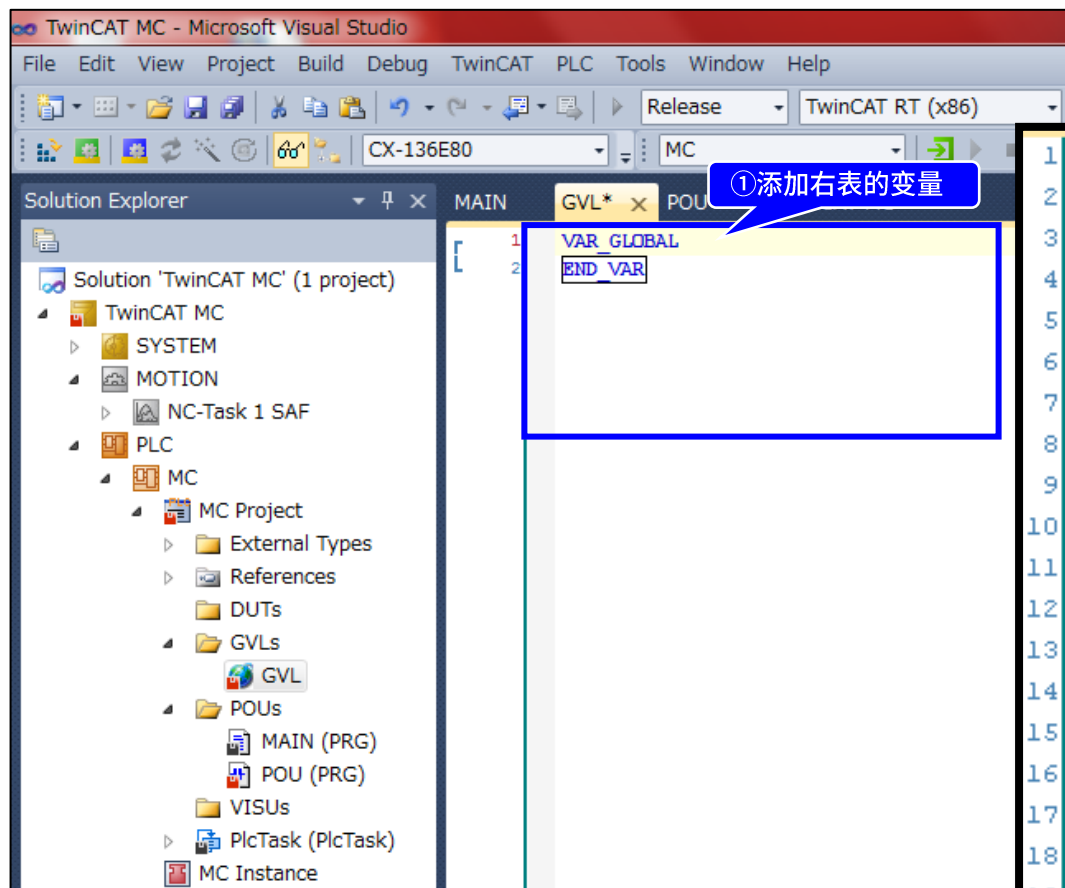
Open

Cancel

程序制作

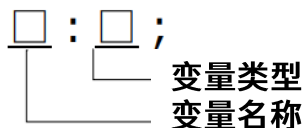
添加Global变量

追加全局变量。有关类型，请参阅每个FB的详细说明页面。



```
1  VAR_GLOBAL
2      Axis1:AXIS_REF;
3      Enale:BOOL;
4      Enable_Positive:BOOL;
5      Enable_Negative:BOOL;
6      BufferMode:MC_BufferMode;
7      Power_Options:ST_PowerOptions;
8      Power_Status:BOOL;
9      Power_Busy:BOOL;
10     Power_Active:BOOL;
11     Power_Error:BOOL;
12     Power_ErrorID:UDINT;
13
14     MoveABS_Execute:BOOL;
15     MoveABS_Options:ST_MoveOptions;
16     MoveABS_Done:BOOL;
17     MoveABS_Busy:BOOL;
18     MoveABS_Active:BOOL;
19     MoveABS_CmdAborted:BOOL;
20     MoveABS_Error:BOOL;
21     MoveABS_ErrorID:UDINT;
22  END_VAR
```

变量与类型的构成



程序制作

FB输入输出参数设定

设定FB输入/输出参数刚刚追加的全局变量。

① 点击

② 点击

③ 点击

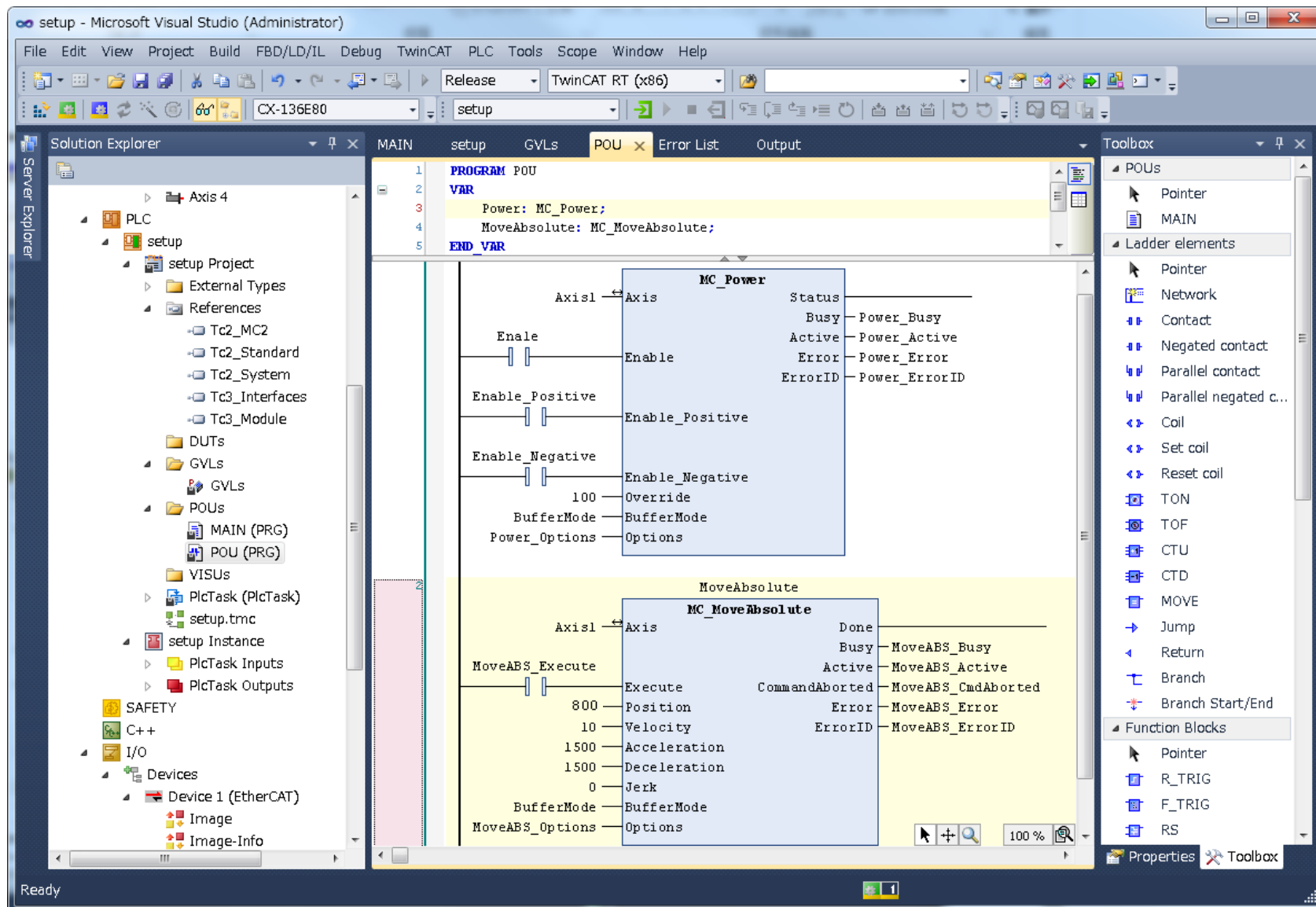
④ 点击

⑤ 双击

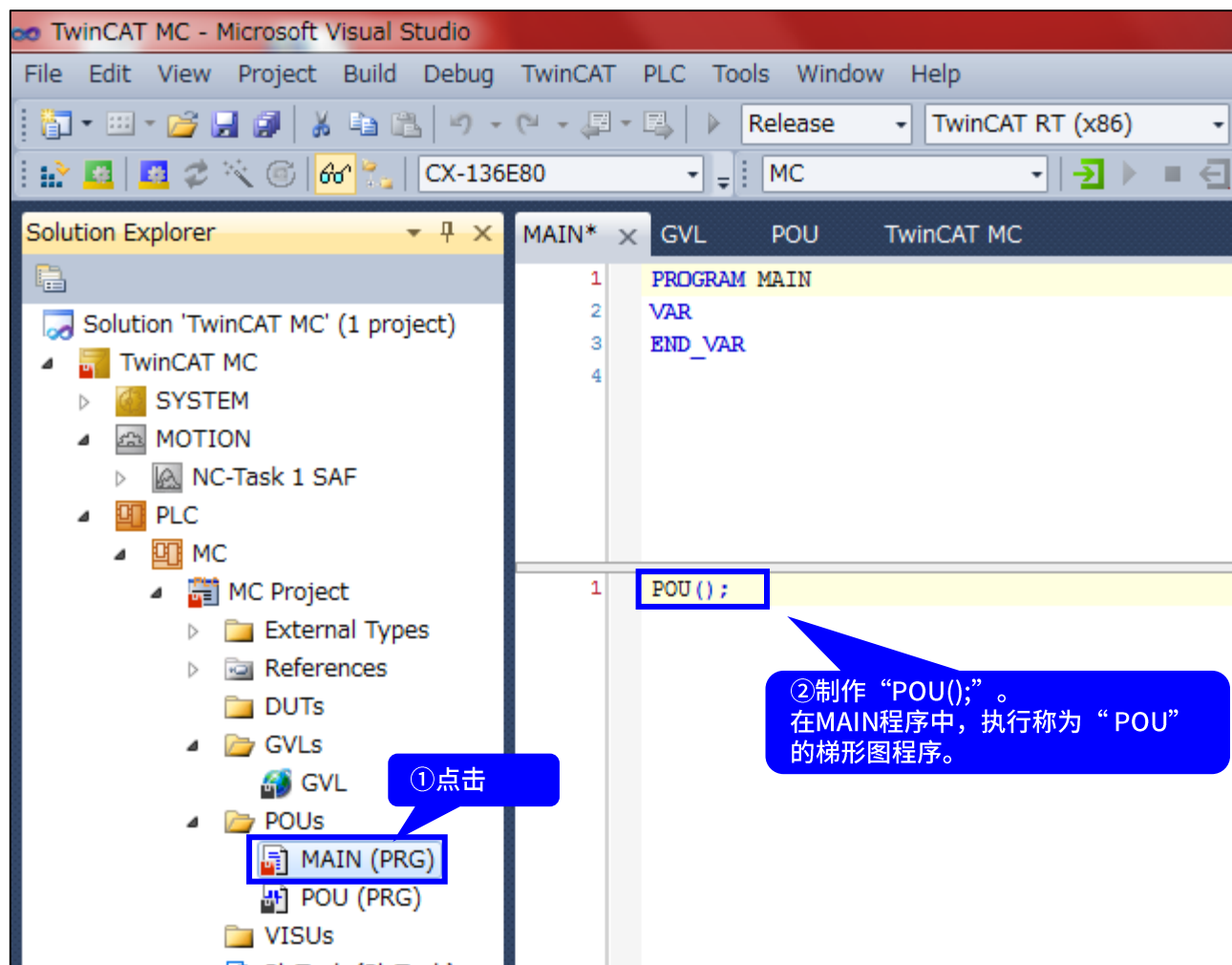
Input Assistant

Name	Type	Address
setup	Application	
Axis1	CONSTANT AXIS_REF	
BufferMode	CONSTANT MC_Buffe...	
Enable_Negative	CONSTANT BOOL	
Enable_Positive	CONSTANT BOOL	
Enale	CONSTANT BOOL	
MoveABS_Active	CONSTANT BOOL	
MoveABS_Busy	CONSTANT BOOL	
MoveABS_CmdAborted	CONSTANT BOOL	
MoveABS_Done	CONSTANT BOOL	
MoveABS_Error	CONSTANT BOOL	

追加所有输入/输出参数后,将显示以下画面。

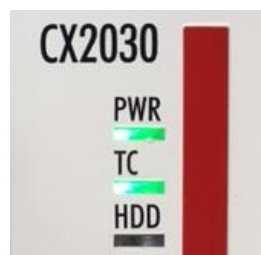


MAIN程序是ST语言，POU程序是LD语言。因此，需要在MAIN内读取POU。



链接在MOTION中进行了轴登录的Axis1与PLC中定义了变量的Axis1。
由此可以将I/O与PLC通过MOTION连接起来。

注意：在RUN模式中，
EPC的TC LED亮绿灯。



LINK

⑤ 将EPC变为“Run模式”

① 点击

② 点击

③ 点击

④ 选择在PLC中设定的轴数变量 (Axis1)

General Settings Parameter Dynamics Online Functions C

Link To I/O... Drive 3 (AZD2A-KED) # A

Link To PLC... GVL.Axis1 (MC)

Axis Type: CANopen MDP 742 (e.g. EtherCAT CoE Drive)

Unit: mm

Display (Only)

Position: ☐ μ m ☐ Modulo

Velocity: ☐ mm/min

Result

Position: mm Velocity: mm/s Acceleration: mm/s² Jerk: mm/s³

Axis Cycle Time / Access Divider

Divider: 1 Cycle Time (ms): 2.000

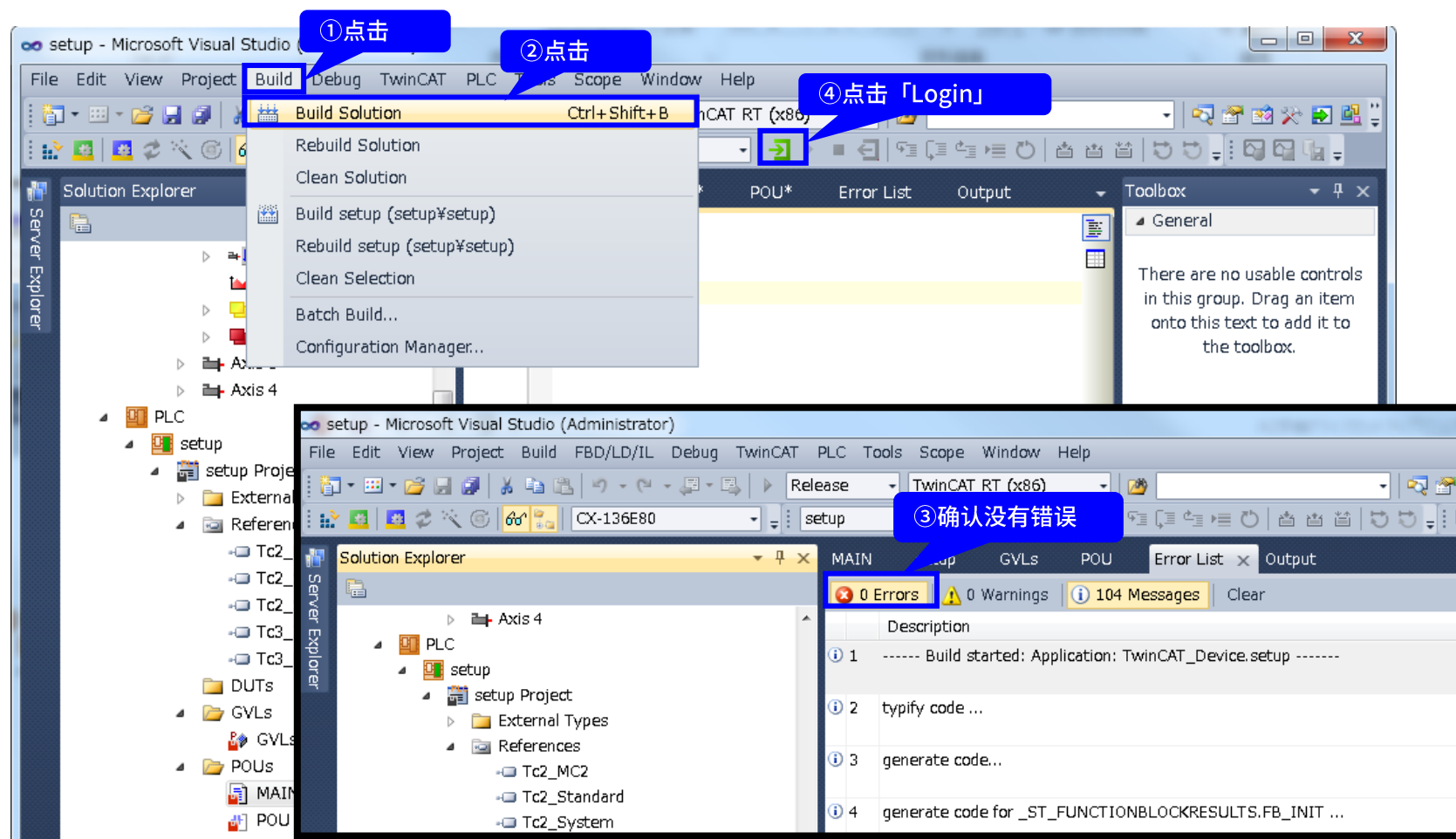
Modulo: 0

Error List

0 Errors 0 Warnings 5 Messages Clear

Description	File	Line	Column	Project
-------------	------	------	--------	---------

执行build，确认没有错误后请登录。



如果build成功,轴变量 (Axis1) 会追加到PLC的MC Instance。

通过CSP模式进行定位运行

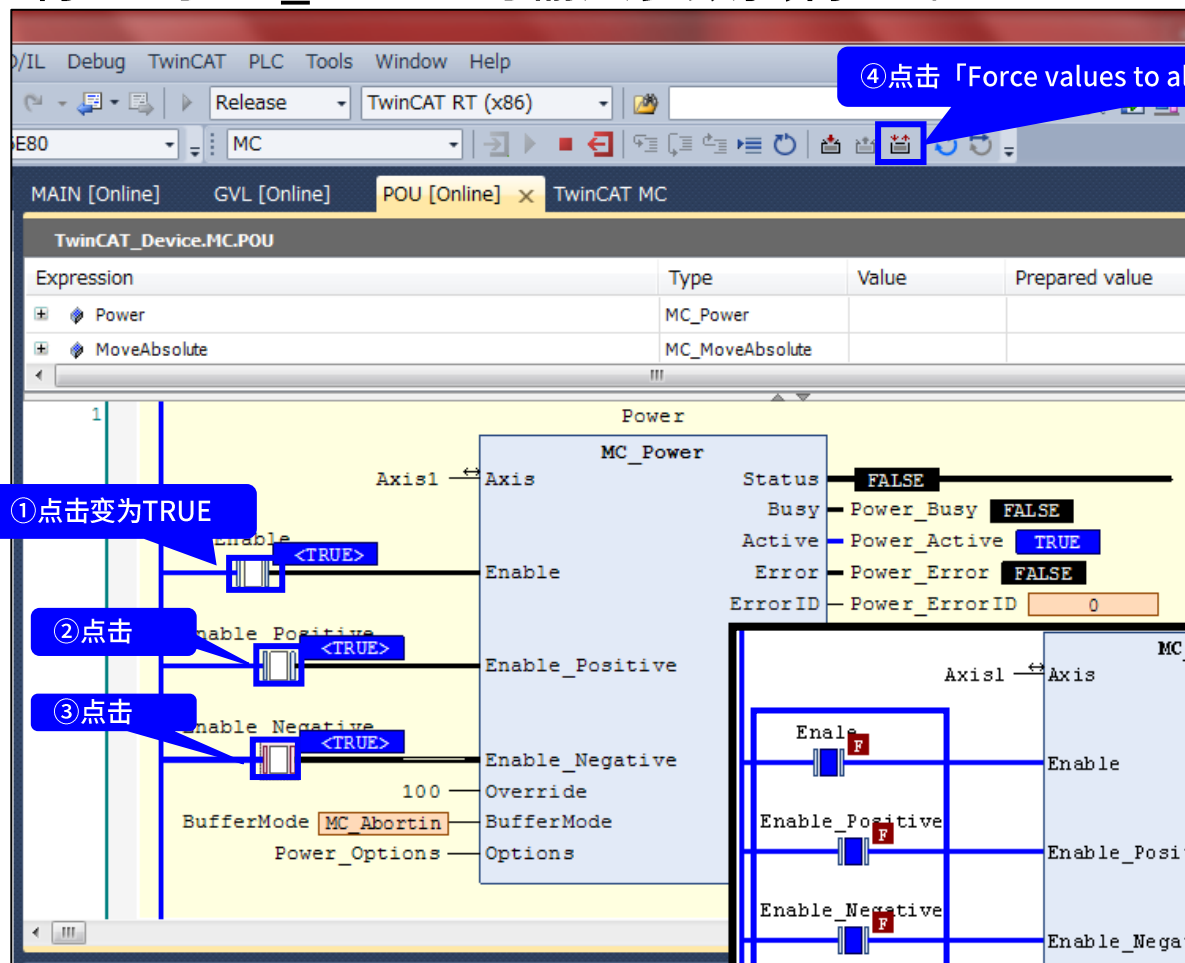
执行MC_Power

Target

通过CSP模式运转电动机

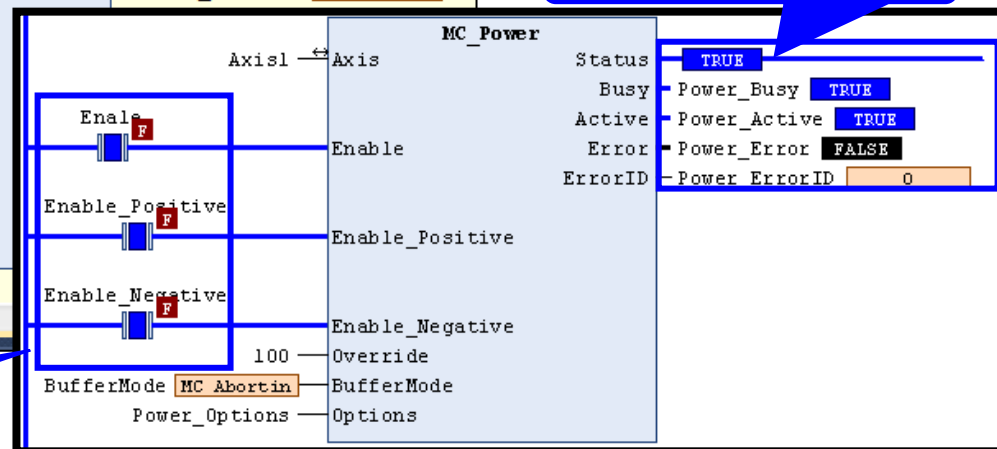
将FB的MC_Power的输入参数强制ON。

④点击「Force values to all on line application」



⑥确认输出参数没有发生错误，且Status变为TRUE

⑤确认输入参数变为TRUE



通过CSP模式进行定位运行

Online页面MC_Power动作确认

MC_Power的输入参数由于与Enabling绑定，因此从Enabling的内容可以确认MC_POWER的执行状态。

MC_Powerの入力パラメータ		OnlineのEnabling
Enable	↔	Controller
Enable Positive	↔	Feed Fw
Enable Negative	↔	Feed Bw

① 点击

② 点击

③ 确认Enabling状态是否全部被选。

General	Settings	Parameter	Dynamics	Online	Functions	Coupling	Compensation
52.8255 Lag Distance (min/max): [mm] -0.0001 (-0.000, 0.000) Override: [%] 100.0000 % Status (log) ☒ Ready ☐ Calibrated ☐ Has Job Status (phys) ☒ NOT Moving ☐ Moving Fw ☐ Moving Bw Controller Kv-Factor: [mm/s/mm] 1 Reference Velocity: [mm/s] 2200 Target Position: [mm] 0 Target Velocity: [mm/s] 0 Setpoint Position: [mm] 50.0000 Setpoint Velocity: [mm/s] 0.0000 Error: [mm] 0.0000 ☒ Enabling ☒ Controller ☒ Feed Fw ☒ Feed Bw Set							

F1

F2

F3

F4

F5

F6

F8

F9

通过CSP模式进行定位运行

MC_MoveAbsolute的执行

与强行将MC_POWER的输入参数变为ON一样，将MoveABS_ExecuteのExecute输入变为ON。

① 点击

② 强行ON

③ 电动机运转。
电动机如果不运转，请确认是否发生输出错误，以及Done是否为TRUE。

Expression	Type	Value	Prepared value	Address
Power	MC_Power			
MoveAbsolute	MC_MoveAbsolute			

MoveAbsolute

MC_MoveAbsolute

Axis1 → Axis

Done → TRUE

Busy → MoveABS_Busy FALSE

Active → MoveABS_Active FALSE

CommandAborted → MoveABS_CmdAborted FALSE

Error → MoveABS_Error FALSE

ErrorID → MoveABS_ErrorID 0

Execute

Position 800

Velocity 10

Acceleration 1500

Deceleration 1500

Jerk 0

BufferMode MC_Abortin

MoveABS_Options

通过CSP模式进行定位运行

确认目标位置，设定速度

从Online画面可以确认电动机是否按照设定的程序运转。

① 点击

② 点击

③ 确认速度正确。

④ 确认定位运行完成，并且目标位置正确。

Parameter	Value	Unit
Setpoint Position	402.9994	[mm]
Actual Velocity	9.9902	[mm/s]
Setpoint Velocity	10.0000	[mm/s]
Lag Distance (min/max)	0.0937 (0.000, 0.094)	[mm]
Override	0.46 / 0.00	%
Controller Kv-Factor	1	[mm/s/mm]
Target Position	0	[mm]

Parameter	Value	Unit
Setpoint Position	800.0000	[mm]
Actual Velocity	0.0000	[mm/s]
Setpoint Velocity	0.0000	[mm/s]
Lag Distance (min/max)	0.0000 (0.000, 0.094)	[mm]
Override	100.0000	%
Controller Kv-Factor	1	[mm/s/mm]
Reference Velocity	2200	[mm/s]
Target Position	0	[mm]
Target Velocity	0	[mm/s]

通过SDO通信读写参数

各种参数的设定方法和存储器保存场所

■ 驱动器的参数设定有以下3种方法。

- ① SDO通信的FB(功能块)
- ② BECKHOFF的TwinCAT
- ③ 支援软件MEXE02（可从本公司的网站免费下载）

■ 关于参数的保存

写入的参数保存在驱动器的RAM或NV存储器中。

写入RAM存储器的参数，若切断驱动器的控制电源, 数据将被删除。

写入NV存储器的参数，即使切断控制电源数据也会被保存。

①②保存在RAM存储器中。要将写入RAM的参数保存到NV存储器，请进行“NV存储器批量写入（40C9h）”。

NV存储器的可写入次数约为10万次。

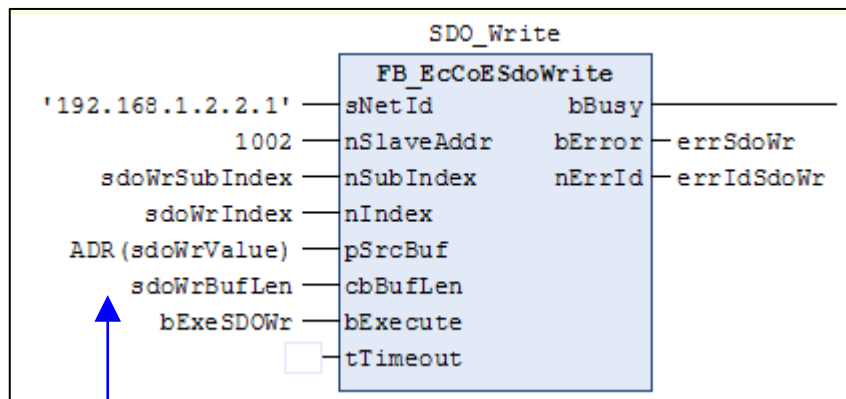
③保存在NV存储器中。

通过SDO通信读写参数

写入用功能块（FB）的说明

下面介绍如何使用SDO通信的FB写入参数。

• 编程例



将bExecute由FALSE(0)变为TRUE(1)时，
执行参数写入。

例：将轴1的运行电流设定为75.0%时
输入运行电流的Index、Sub-Index、
类型(数据长度)，然后执行写入。

Index	Sub	名称	型
4120h	00h	Operating current [1=0.1%]	INT16

• FB输入参数

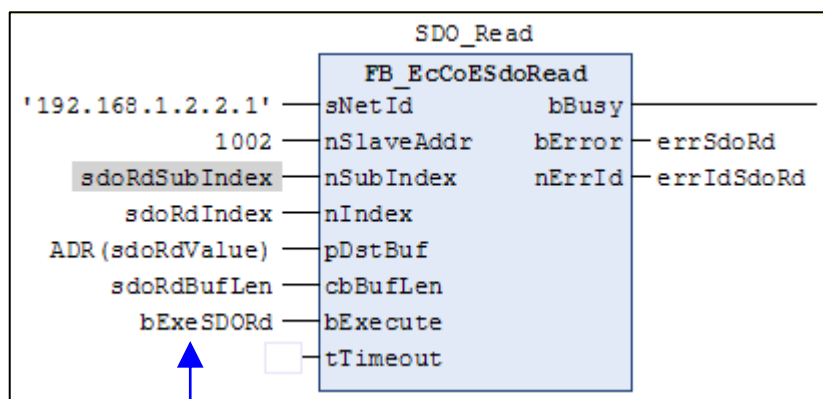
```
SdoWrIndex    := 16#4120;  
SdoWrSubIndex := 1; //Axis1  
SdoWrBufLen   := 2; //2byte  
SdoWrValue    := 750; //0.1%单位
```

通过SDO通信读写参数

读取用功能块 (FB) 的说明

下面介绍如何使用SDO通信的FB读取参数。

• 编程例



输入要读取对象的Index、Sub-Index、字节长度，
将bExecute由FALSE(0)变为TRUE(1)时,执行读取。
读取的结果存储在SdoRdValue中。

例：读取运行电流时，输入运行电流的
Index、Sub-Index、类型(数据长度)，
然后执行读取。

Index	Sub	名称	型
4120h	00h	Operating current [I=0.1%]	INT16

• FB的输入参数

```
SdoRdIndex    := 16#4120;  
SdoRdSubIndex:= 1; //Axis1  
SdoRdBufLen   := 2; //2byte
```

通过SDO通信读写参数 添加Library

Target

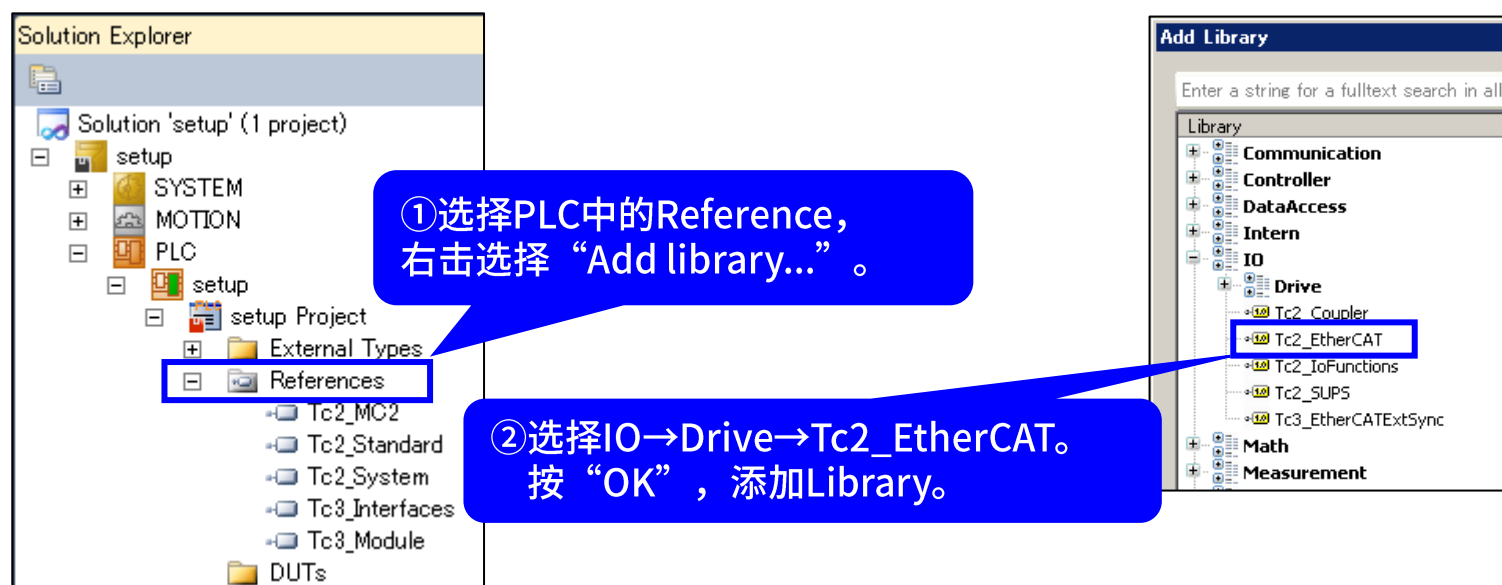
用SDO通信读写参数。

添加以下FB，可以通过SDO执行参数的读写。

写入参数：FB_EcCoESdoWrite

读取参数：FB_EcCoESdoRead

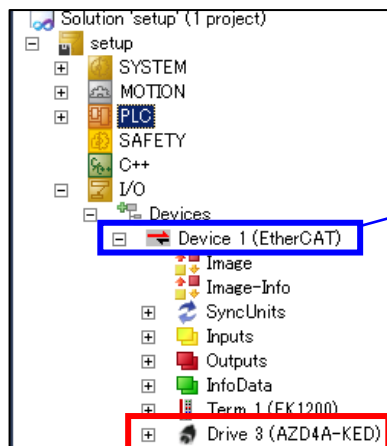
为了使用FB_EcCoESdoWrite, FB_EcCoESdoRead，需要按以下步骤添加Tc2_EtherCAT的Library。



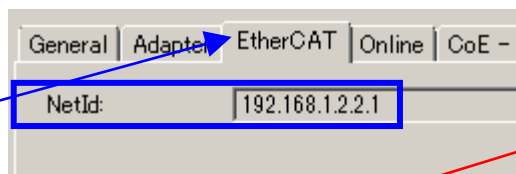
通过SDO通信读写参数

FB输入参数的设定内容

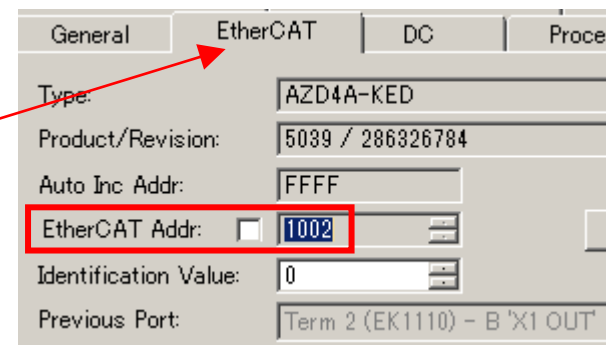
• 确认写入到FB的输入参数值



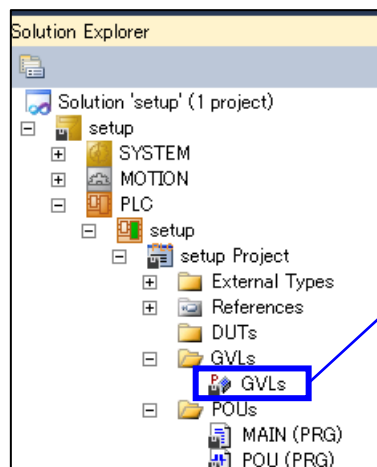
“sNetId” 的确认



“sSlaveAddr” 的确认



• 分配到各种输入参数的Global变量一览



```
VAR_GLOBAL
    bExeSDOWr      : BOOL;
    SdoWrValue      : DINT;
    SdoWrIndex       : WORD;
    SdoWrSubIndex    : BYTE;
    SdoWrSrcBuf      : PVOID;
    SdoWrBufLen      : UDINT;
    ErrSdoWr         : BOOL;
    ErrIdSdoWr       : UDINT;
END_VAR
```

例)写入用Global变量

```
VAR_GLOBAL
    bExeSDORd       : BOOL;
    SdoRdValue       : DINT;
    SdoRdIndex       : WORD;
    SdoRdSubIndex    : BYTE;
    SdoRdSrcBuf      : PVOID;
    SdoRdBufLen      : UDINT;
    ErrSdoRd         : BOOL;
    ErrIdSdoRd       : UDINT;
END_VAR
```

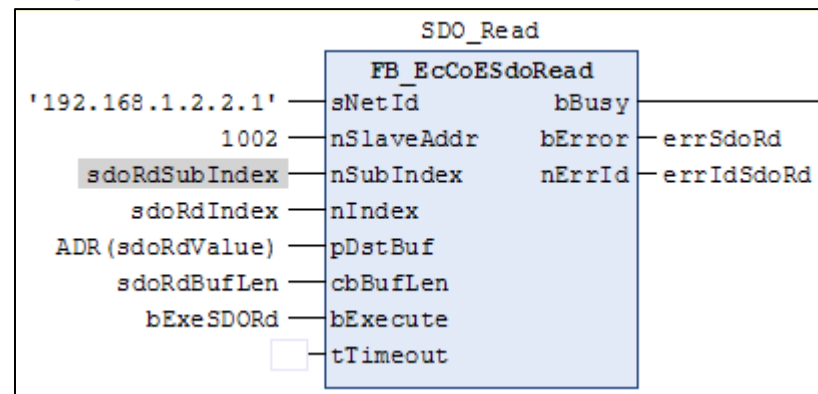
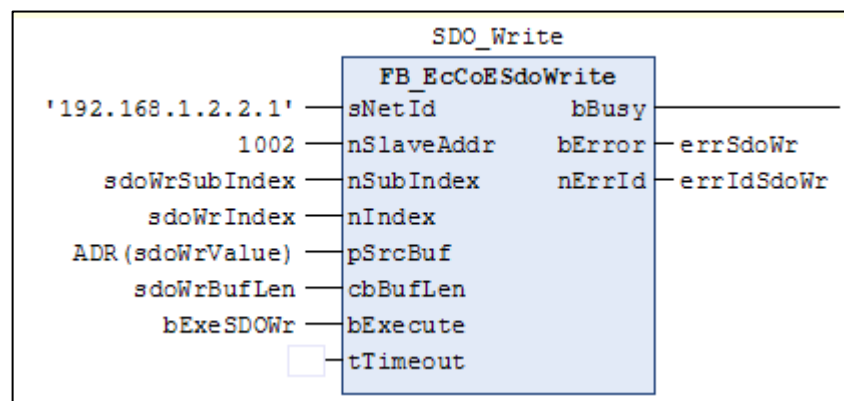
例)读取用Global变量

通过SDO通信读写参数

FB输入参数说明

FB_EcCoESdoWrite , FB_EcCoESdoRead的输入参数

- 程序例（使用View as function block diagram显示）



输入参数	输入的内容
sNetId	EtherCAT主站的AMS networkID
nSlaveAddr	EtherCAT从站地址
nSubIndex	读写对象的Sub-Index
nIndex	输入要读写对象的Index
pSrcBuf	指定读写用的数据缓冲器的地址。 制作数据缓冲器用的变量，用ADR()进行地址转换
cbBufLen	读写对象的最大字节长度

通过SDO通信读写参数

定义Global变量

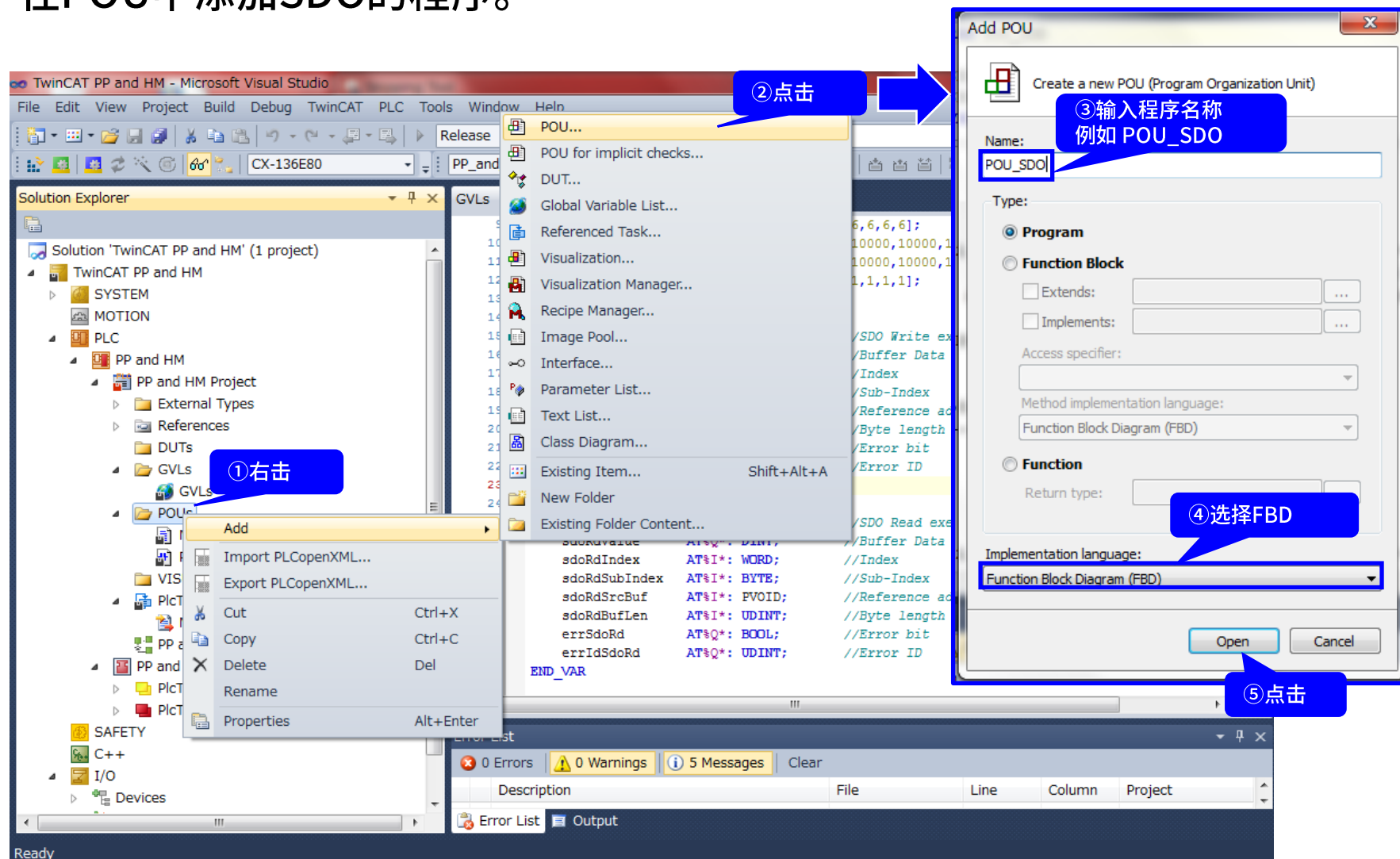
在GVLs变量中添加SDO用参数。将以下作为FB输入变量使用。

```
GVLs x
10  TPOS  AT%Q*:ARRAY[0..3] OF DINT:= [10000,10000,10000,10000]; //Target
11  PVEL  AT%Q*:ARRAY[0..3] OF DINT:= [10000,10000,10000,10000]; //Profile
12  OPMD  AT%Q*:ARRAY[0..3] OF SINT:= [1,1,1,1]; //Operati
13
14  //SDO Write
15  bExeSDOWr      AT%I*: BOOL;           //SDO Write execution
16  sdoWrValue      AT%I*: DINT;           //Buffer Data for writing
17  sdoWrIndex       AT%I*: WORD;          //Index
18  sdoWrSubIndex    AT%I*: BYTE;          //Sub-Index
19  sdoWrSrcBuf      AT%I*: PVOID;         //Reference address of buffer data
20  sdoWrBufLen      AT%I*: UDINT;         //Byte length
21  errSdoWr        AT%Q*: BOOL;          //Error bit
22  errIdSdoWr      AT%Q*: UDINT;         //Error ID
23
24  //SDO Read
25  bExeSDORd      AT%I*: BOOL;           //SDO Read execution
26  sdoRdValue      AT%Q*: DINT;          //Buffer Data for reading
27  sdoRdIndex       AT%I*: WORD;          //Index
28  sdoRdSubIndex    AT%I*: BYTE;          //Sub-Index
29  sdoRdSrcBuf      AT%I*: PVOID;         //Reference address of buffer data
30  sdoRdBufLen      AT%I*: UDINT;         //Byte length
31  errSdoRd        AT%Q*: BOOL;          //Error bit
32  errIdSdoRd      AT%Q*: UDINT;         //Error ID
33  END_VAR
```

通过SDO通信读写参数

编辑SDO通信用程序

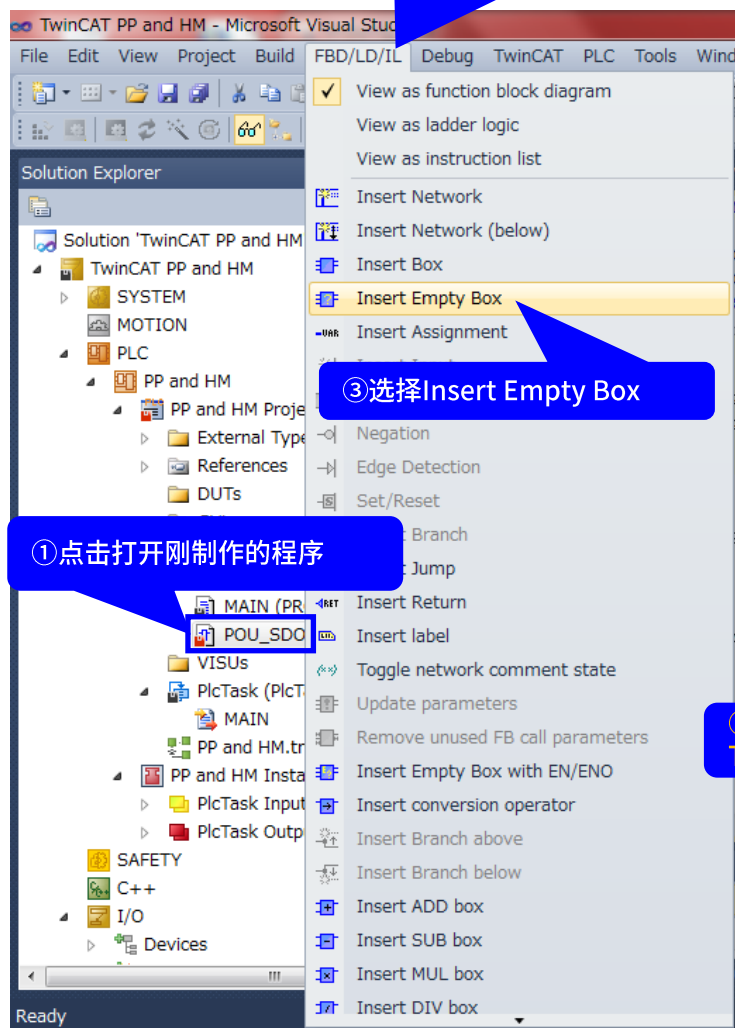
在POU中添加SDO的程序。



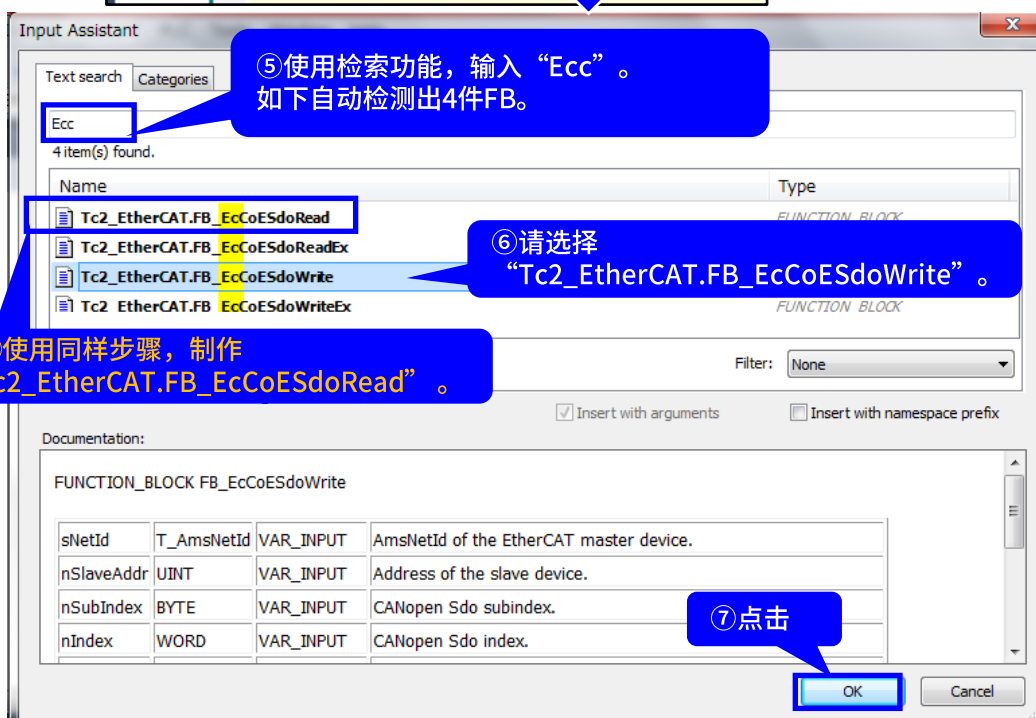
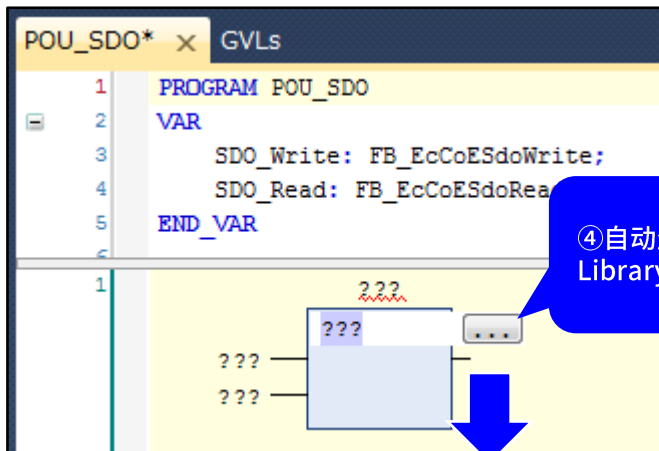
通过SDO通信读写参数 使用功能块

制作SDO读写用的程序。

②选择需要插入FB的行,点击菜单



①点击打开刚制作的程序



Documentation:

FUNCTION_BLOCK FB_EcCoESdoWrite

sNetId	T_AmsNetId	VAR_INPUT	AmsNetId of the EtherCAT master device.
nSlaveAddr	UINT	VAR_INPUT	Address of the slave device.
nSubIndex	BYTE	VAR_INPUT	CANopen Sdo subindex.
nIndex	WORD	VAR_INPUT	CANopen Sdo index.

通过SDO通信读写参数

样本程序的说明

设定FB的输出输入变量

POU_SDO* x GVLs

```
1 PROGRAM POU_SDO
2 VAR
3   SDO_Write: FB_EcCoESdoWrite;
4   SDO_Read: FB_EcCoESdoRead;
5 END_VAR
```

①请分别设定以下红色部分。

SDO_Write

Variable	Value
sNetId	'192.168.1.2.2.1'
nSlaveAddr	1002
nSubIndex	sdoWrSubIndex
nIndex	sdoWrIndex
pSrcBuf	ADR(sdoWrValue)
cbBufLen	sdoWrBufLen
bExecute	bExeSDOWr
tTimeout	

SDO_Read

Variable	Value
sNetId	'192.168.1.2.2.1'
nSlaveAddr	1002
nSubIndex	sdoRdSubIndex
nIndex	sdoRdIndex
pDstBuf	ADR(sdoRdValue)
cbBufLen	sdoRdBufLen
bExecute	bExeSDORD
tTimeout	

在MAIN程序中添加POU_SDO。

MAIN x POU_SDO* GVLs

```
1 PROGRAM MAIN
2 VAR
3 END_VAR
4
5 //SDO READ/WRITE
6 POU_SDO();
7
8 //Controlword
9 IF 1=bCON[0] THEN
10   CTWD[0] := 15;
```

②Main程序添加一行。
POU_SDO();

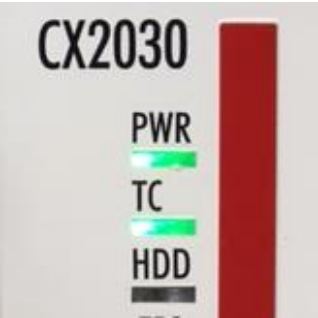
通过SDO通信读写参数

将执行文件下载到EPC

执行Build，确认没有错误后，下载到EPC。

③将EPC设定为RUN模式

注意：在RUN模式中，EPC的TC LED亮绿灯。



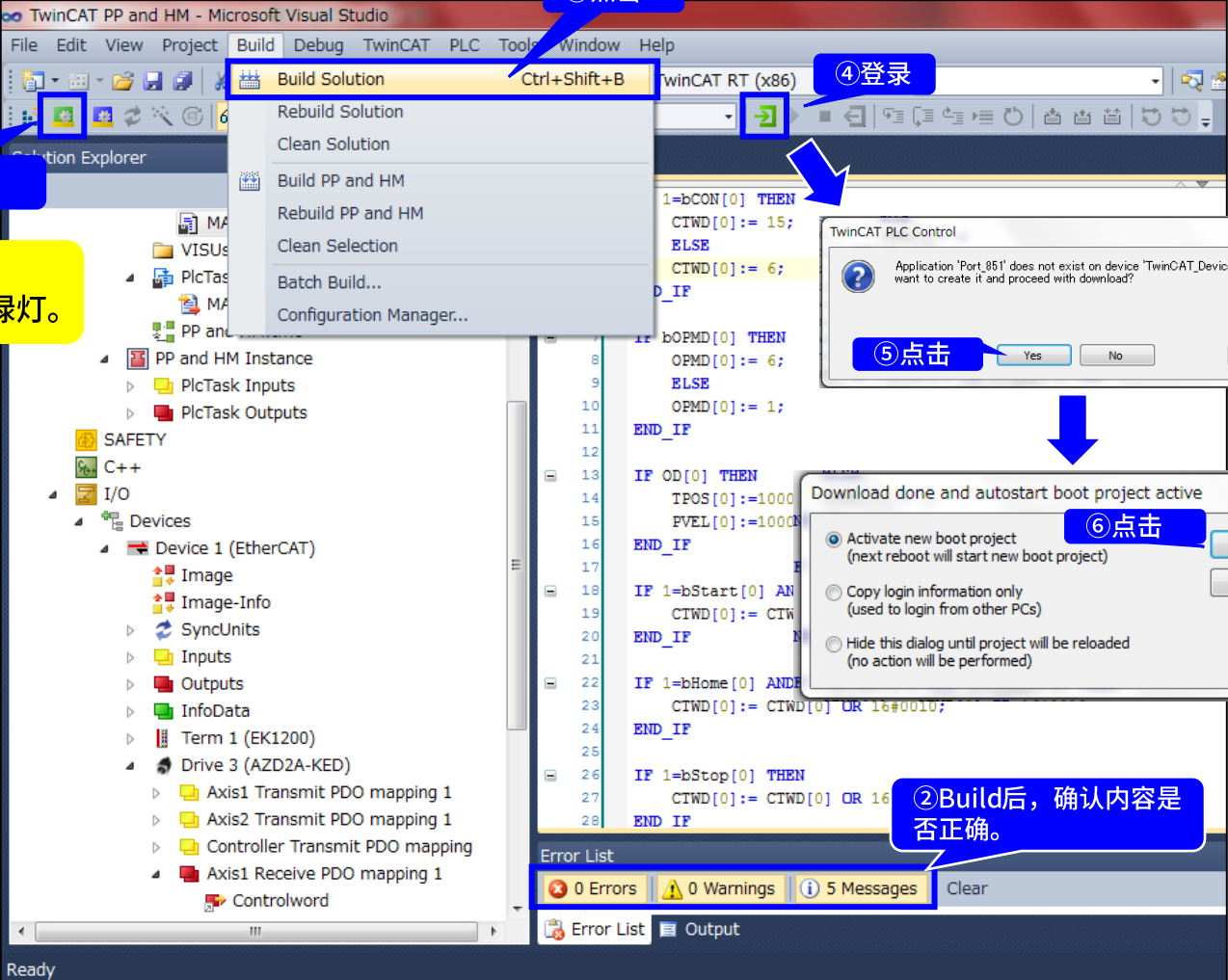
①点击

④登录

⑤点击

⑥点击

②Build后，确认内容是否正确。



通过SDO通信读写参数

强行更改输入参数

登录后，从GVLs[Online]可以写入SDO写入用的FB输入变量。

②写入Prepared value 输入的数据。

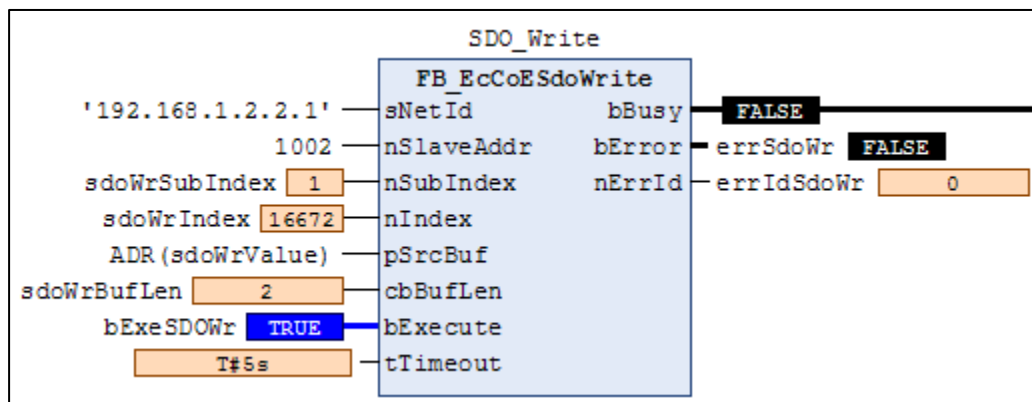
Expression	Type	Value	Prepared value	Address
OPMD	ARRAY [0..3...]			%Q*
bExeSDOWr	BOOL	FALSE	TRUE	%I*
sdoWrValue	DINT	0	750	%I*
sdoWrIndex	WORD	0	16672	%I*
sdoWrSubIndex	BYTE	0	1	%I*
sdoWrSrcBuf	UDINT	0		%I*
sdoWrBufLen	UDINT	0	2	%I*
errSdoWr	BOOL	FALSE		%Q*
errIdSdoWr	UDINT			%Q*
bExeSDORd	BOOL	FALSE		%I*
sdoRdValue	DINT	0		%Q*

①输入各变量的Prepared Value。

执行写入后

Expression	Type	Value	Prepared value
OPMD	ARRAY [0..3...]		
bExeSDOWr	BOOL	TRUE	
sdoWrValue	DINT	750	
sdoWrIndex	WORD	16672	
sdoWrSubIndex	BYTE	1	
sdoWrSrcBuf	UDINT	0	
sdoWrBufLen	UDINT	2	
errSdoWr	BOOL	FALSE	
errIdSdoWr	UDINT	0	
bExeSDORd	BOOL	FALSE	
sdoRdValue	DINT	0	

●通过FB的输出参数也可以确认是否发生错误。



通过SDO通信读写参数

通过MEXE02确认写入内容

确认SDO写入的“运行电流[%]”值。



使用支援软件“MEXE02”，确认参数。软件可从本公司官网免费下载。

<https://www.orientalmotor.co.jp/download/software/mexe02/>

MEXE02 简体中文版 - [新規1*] | AZ (多轴) EtherCAT对应: 电动机/传动装置 10000P/R

文件(F) 编辑(E) 移动(M) 查看(V) 通信(C) 工具(T) 窗口(W) 支援(S) 帮助(H)

显示/印刷轴数选择 ☐ 2轴型 ☐ 3轴型 ☒ 4轴型

AZ(多轴)EtherCAT对应 / 电动机/传动装置

参数

- 驱动器轴
 - Profile area的对象
 - 基本设定
 - 电动机 机构(坐标/位置)
 - Alarm-Info设定
 - I/O动作 功能
 - Direct-IN 功能选择

运行

遥控运行

监控

组合信息监视

状态监视

Profile area of the object 基本设定

		Axis 1	Axis 2
1	Motor user memo		
2	Driver user memo		
3			
4	Driver simulation mode	实际使用电动机	
5	Main power mode	自动识别	
6			
7	Base current [%]	100.0	100.0
8	Operating current [%]	75.0	100.0
9	Stop current [%]	50.0	50.0
10	Push current [%]	20.0	20.0
11	Command filter setting	LPF(速度平滑调整)	LPF(速度平滑调整)
12	Command filter time constant [ms]	1	1

① 点击

② 运行电流[%]的初始值为100.0，通过SDO通信成功将其修改为75.0。

Point

PC与AZ多轴驱动器连接时使用的USB电缆线(连接器A型-miniB型)需要另行购买。使用MEXE02设定软件，可以确认EtherCAT通信时AZ多轴驱动器是否按设计接受到数据。对于设备前期准备时的测评，以及发生警报时的原因调查都能起到很大的帮助。

通过SDO通信读写参数

确认读取的内容

从GVLs[Online]可以写入SDO写入用的FB输入变量。
读取已写入的运行电流值，确认是否相同。

PP_and_HM

MAIN [Online] POU_SDO [Online] **GVLs [Online]** x

TwinCAT_Device.PP_and_HM.GVLs

Expression	Type	Value	Prepared value
sdoWrSrcBuf	UDINT	0	
sdoWrBufLen	UDINT	2	
errSdoWr	BOOL	FALSE	FALSE
errIdSdoWr	UDINT	0	
bExeSDORd	BOOL	FALSE	TRUE
sdoRdValue	DINT	0	
sdoRdIndex	WORD	0	16672
sdoRdSubIndex	BYTE	0	1
sdoRdSrcBuf	UDINT	0	
sdoRdBufLen	UDINT	0	

①输入各变量的Prepared Value.

②点击。写入 Prepared value的值。

PP_and_HM

MAIN [Online] POU_SDO [Online] **GVLs [Online]** x

T_Device.PP_and_HM.GVLs

Expression	Type	Value	Prepared value
sdoWrSrcBuf	UDINT	0	
sdoWrBufLen	UDINT	2	
errSdoWr	BOOL	FALSE	
errIdSdoWr	UDINT	0	
bExeSDORd	BOOL	TRUE	
sdoRdValue	DINT	750	
sdoRdIndex	WORD	16672	
sdoRdSubIndex	BYTE	1	
sdoRdSrcBuf	UDINT	0	
sdoRdBufLen	UDINT	2	

③在SDO通信中，成功的读取了数值。

●可以确认FB的输出参数是否发生错误。

