

客服热线 400-820-9595

绵密网络 专业服务

中达电通已建立了 70 余个分支机构及服务网点，并塑建训练有素的专业团队，提供客户最满意的服务，公司技术人员能在 2 小时内回应您的问题，并在 48 小时内提供所需服务。

沈阳 电话:(024)2334-1160	济南 电话:(0531)8690-6277	杭州 电话:(0571)8882-0610	重庆 电话:(023)8806-0306	乌鲁木齐 电话:(0991)4678-141
哈尔滨 电话:(0451)5366-5568	太原 电话:(0351)4039-485	合肥 电话:(0551)6281-6777	昆明 电话:(0871)6313-7362	兰州 电话:(0931)6406-725
长春 电话:(0431)8892-5060	郑州 电话:(0371)6384-2772	武汉 电话:(027)8544-8475	广州 电话:(020)3879-2175	西安 电话:(029)8836-0780
呼和浩特 电话:(0471)6297-808	石家庄 电话:(0311)8666-7337	南昌 电话:(0791)8625-5010	厦门 电话:(0592)5313-601	贵阳 电话:(0851)8690-1374
北京 电话:(010)8225-3225	上海 电话:(021)6301-2827	成都 电话:(028)8434-2075	南宁 电话:(0771)2621-501	福州 电话:(0591)8755-1305
天津 电话:(022)2301-5082	南京 电话:(025)8334-6585	长沙 电话:(0731)8549-9156		

AX-3 系列操作手册



AX-3 系列操作手册



中达电通股份有限公司

地址：上海市浦东新区民夏路238号

邮编：201209

电话：(021) 5863-5678

传真：(021) 5863-0003

网址：<http://www.delta-china.com.cn>



扫一扫，关注官方微信

中达电通公司版权所有
如有改动，恕不另行通知

AX-3-0339410-05
2022/07/21

www.deltaww.com



AX-3 系列操作手册

版本修订一览表

版本	变更内容	发行日期
第一版	第一版发行	2020/09/21
第二版	<ol style="list-style-type: none">第 1 章及第 2 章：增加机种 AX-300NA0PA1、AX-324NA0PA1P、AX-308EA0MA1P 相关信息第 4 章：依新版软件更新软件图、第 4.4.1.4 节增加同步轴的运动控制指令、第 4.2.1.11 节 System Setting 依新软件图更新文字说明、第 4.2.2 节增加机种 AS02PU/AS04PU 数据交换所需时间、增加第 4.3.2.5 节变量清除为零时机说明第 7 章：第 7.4.2 节更新轴介绍说明、第 7.4.2.1 节轴参数介绍同步轴部份增加 Servo Gear Ratio Setting 而定位轴部份增加软件图及编号说明、第 7.4.3 节更新步骤说明、更正 Trapeziod 为 Trapezoid、第 7.5.2 节增加轴组变量第 8 章：第 8.2.1 节更新软件图、删除原先第 8.3.1 节 OPC UA Server 启动认证并增加第 8.3.4 节使用 Prosys OPC UA Client 建立加密联机第 9 章：第 9.1.3 节增加 Startup Checking 及 Timeouts 说明、第 9.3.1.2 节增加批注说明	2021/03/26
第三版	<ol style="list-style-type: none">第 1 章及第 2 章：增加机种 AX-304ELA0PA1T、AX-304ELA0PA1P、AX-316EA0MA1T、AS02PU-A、AS04PU-A、AS02HC-A 相关信息第 4 章：第 4.3.2.5 节更新表格、新增第 4.3.2.6 节变量生效时机、第 4.4.1.4 节更新表格内容及增加第 4.5 节配方管理器第 8.3.4 节更新步骤 9 内容第 9 章新增第 9.4 节 EtherNet/IP 通讯附录 A：第 A.4.2.2 节更新错误码 16#1807 排除方式及增加机种 AS02HC-A、AS02PU-A 及 AS04PU-A 相关信息	2021/09/03

版本	变更内容	发行日期
第四版	<ol style="list-style-type: none"> 1.第 1.1.2 节新增 AX-332 及 AS02ADH-A 模块介绍 2.第 2 章增加外壳防火等级规格及 AX-332 机种相关信息、新增第 2.2.5 节主机配线 3.第 3 章增加 AX-332 机种相关信息 4.第 4.2.1.11 节增加 project ID 及 AX-332 机种说明 5.第 7.4.2.1 节增加台达变频器相关设定、第 7.6.3.5 节增加版本说明、第 7.7.7.6 节更新软件画面及其说明、第 7.7.8.2 节增加轴组极限说明 6.第 9 章增加第 9.1.4 节 EtherCAT 从站参数备份还原、第 9.4.4.6 节更新当 instance=1 时的值内容、增加第 9.6 节 FTP 功能 	2022/7/21

AX-3 系列操作手冊

目录

第1章 简介

1.1 概述	1-2
1.1.1 相关手册	1-2
1.1.2 机种简易说明	1-2
1.2 DIADesigner-AX 软件概述	1-8
1.2.1 特色简介	1-8

第 2 章 规格与系统配置

2.1 一般规格	2-2
2.2 CPU 模块规格	2-4
2.2.1 主机性能规格	2-4
2.2.2 电源与重量及电气规格	2-10
2.2.3 CPU 模块外观尺寸及部位介绍	2-12
2.2.4 输入输出端子配置 (点位定义)	2-19
2.2.5 主机配线	2-21
2.2.5.1 AX-304ELAOPA1T 配线	2-21
2.2.5.2 AX-304ELAOPA1P 配线	2-22
2.2.5.3 AX-308EAOMA1T、AX-316EAMA1T、AX-364ELAOMA1T 配线	2-23
2.2.5.4 AX-308EAOM1P 配线	2-25
2.2.5.5 AX-324NAOPA1P 配线	2-26
2.2.5.6 AX-332EPOMB1T 配线	2-27
2.3 电源模块规格	2-29
2.3.1 一般规格	2-29
2.3.2 电源模块外观尺寸及部位介绍	2-29
2.3.3 端子配置	2-31
2.4 扩展模块	2-31

第 3 章 安装与启动

3.1 硬件安装	3-2
3.1.1 SD 卡插入与取出方法	3-2
3.1.1.1 AX-3 系列 (除 AX-332 外) SD 卡插入与取出方法	3-2
3.1.1.2 AX-332 SD 卡插入与取出方法	3-3
3.1.2 电池的安装与拆除方法	3-4
3.1.2.1 电池的安装	3-4
3.1.2.1.1 AX-3 系列 (除 AX-332 外) 电池的安装	3-4
3.1.2.1.2 AX-332 电池的安装	3-6
3.1.2.2 电池的拆除	3-7
3.1.2.2.1 AX-3 系列 (除 AX-332 外) 电池拆除	3-7
3.1.2.2.2 AX-332 电池拆除	3-9
3.1.3 安装至控制柜	3-11
3.1.3.1 安装至 DIN 导轨	3-11
3.1.3.2 控制柜内的安装方法	3-11
3.1.3.3 控制柜内的环境温度要求	3-12
3.1.3.4 提高抗干扰性的措施	3-12
3.1.3.5 控制柜内的尺寸要求	3-12
3.2 软件安装与卸除	3-13
3.2.1 软件安装	3-13
3.2.2 软件卸除	3-20
3.3 软件激活与通讯	3-21
3.3.1 启动	3-21
3.3.2 通讯	3-22

第 4 章 基本操作

4.1 DIADesigner-AX 软件操作	4-3
4.1.1 建立项目	4-3
4.2 Device 页面介绍	4-5
4.2.1 主机参数设定	4-5
4.2.1.1 Communication Settings	4-5
4.2.1.2 Applications	4-8
4.2.1.3 Backup and Restore	4-9
4.2.1.4 Files	4-10

4.2.1.5 Log	4-10
4.2.1.6 PLC Settings.....	4-12
4.2.1.7 PLC Shell	4-13
4.2.1.8 Users and Groups	4-14
4.2.1.9 Access Rights	4-17
4.2.1.10 Symbol Rights	4-23
4.2.1.11 System Setting	4-24
4.2.1.12 System Parameters	4-26
4.2.1.13 Task Deployment.....	4-27
4.2.1.14 Status	4-28
4.2.1.15 Information	4-28
4.2.2 扩展模块参数设定.....	4-29
4.3 数据类型与变量	4-30
4.3.1 数据类型.....	4-30
4.3.2 变数	4-31
4.3.2.1 变量宣告	4-31
4.3.2.2 寻址	4-32
4.3.2.3 变量类型	4-33
4.3.2.4 自定义数据类型.....	4-37
4.3.2.5 变量清除为零时机说明	4-40
4.3.2.6 变数初始值生效时机	4-40
4.4 控制器原理.....	4-41
4.4.1 工作 (Task)	4-41
4.4.1.1 工作 (Task) 类型	4-41
4.4.1.2 多工作 (Task) 执行.....	4-42
4.4.1.3 Watchdog	4-43
4.4.1.4 每种任务类型可以执行的运动控制指令	4-43
4.5 配方管理器	4-46
4.5.1 配方管理器	4-48
4.5.2 配方定义.....	4-50
4.5.3 RecipeManCommands.....	4-52

第5章 Hardware Configuration

5.1 Hardware Configuration.....	5-2
5.1.1 环境介绍.....	5-2
5.1.2 新增模块.....	5-5

5.1.3	删除模块	5-7
5.1.4	模块的复制/贴上	5-9
5.1.5	模块的剪下/贴上	5-11
第 6 章	Network Configuration	
6.1	Network Configuration	6-2
6.1.1	简介	6-2
6.1.2	基础介绍	6-3
6.1.3	网络架构的配置	6-5
6.1.3.1	配置节点	6-5
6.1.3.2	链接网络	6-7
第 7 章	运动控制基本设定与操作	
7.1	运动控制指令概述	7-5
7.1.1	运动控制指令	7-5
7.1.2	运动控制指令应用要点	7-5
7.1.3	运动控制指令的分类	7-5
7.2	运动项目建立流程	7-6
7.2.1	项目建立流程图	7-6
7.2.2	项目建立流程	7-6
7.3	监控与调测	7-13
7.3.1	使用方法	7-13
7.3.2	轴参数设定范例	7-13
7.3.3	测试轴运转	7-15
7.4	运动控制装置	7-18
7.4.1	轴设定	7-18
7.4.2	轴介绍	7-18
7.4.2.1	轴参数介绍	7-19
7.4.2.2	程序中使用轴	7-28
7.4.3	单轴配置流程与设定步骤	7-29
7.4.4	轴组设定	7-37
7.4.4.1	轴组参数介绍	7-37
7.4.4.2	程序中使用轴组	7-39

7.4.5	轴组配置流程与设定步骤.....	7-40
7.5	运动轴变量.....	7-45
7.5.1	单轴变数.....	7-45
7.5.2	轴组变数.....	7-47
7.6	运动控制程序编辑.....	7-49
7.6.1	运动控制程序.....	7-49
7.6.1.1	DIADesigner-AX 编成架构与种类.....	7-49
7.6.1.2	DIADesigner-AX 中的 POU.....	7-50
7.6.1.3	DIADesigner-AX 新增 POU.....	7-50
7.6.1.4	PDO 映射 (Mapping).....	7-53
7.6.2	轴状态转换.....	7-53
7.6.2.1	轴状态.....	7-53
7.6.2.2	轴组状态.....	7-55
7.6.3	运动控制指令的执行和状态.....	7-56
7.6.3.1	执行指令的基本规则.....	7-56
7.6.3.2	输入/输出引脚时序图.....	7-58
7.6.3.3	单一运动指令重复执行之行为.....	7-59
7.6.3.4	多个运动指令同时启动.....	7-59
7.6.3.5	运动指令同时执行之行为.....	7-60
7.6.4	位置.....	7-65
7.6.4.1	位置的种类.....	7-65
7.6.5	凸轮表和凸轮架构.....	7-65
7.6.5.1	电子凸轮架构.....	7-65
7.6.5.2	电子凸轮建立.....	7-66
7.7	运动控制功能.....	7-70
7.7.1	系统架构.....	7-70
7.7.2	单轴控制.....	7-70
7.7.2.1	周期同步位置模式 (Cyclic Synchronous Position Mode) ..	7-70
7.7.2.2	位置规划模式 (Profile Position Mode) ..	7-71
7.7.2.3	定位.....	7-71
7.7.2.4	停止方式.....	7-72
7.7.2.5	电子齿轮 (MC_GearIn) ..	7-75
7.7.2.6	位置指定电子齿轮 (MC_GearInPos) ..	7-77

7.7.2.7 电子凸轮 (MC_CamIn)	7-78
7.7.3 速度控制	7-87
7.7.3.1 CSP 模式.....	7-87
7.7.3.2 CSV 模式	7-88
7.7.3.3 速度规划模式 (Profile Velocity Mode)	7-88
7.7.4 扭矩控制	7-89
7.7.5 单轴控制的常用功能	7-90
7.7.5.1 位置命令	7-90
7.7.5.2 速度命令	7-91
7.7.5.3 加速度与减速度命令.....	7-92
7.7.5.4 加速度的变化率 (跃度) 命令.....	7-93
7.7.5.5 轴方向	7-94
7.7.6 轴组控制	7-96
7.7.6.1 轴组直线插补	7-96
7.7.6.2 轴组螺旋插补	7-97
7.7.6.3 轴组命令停止	7-98
7.7.7 本体高速 IO	7-99
7.7.7.1 I/O 功能配置	7-99
7.7.7.2 DIO 功能设定	7-100
7.7.7.3 SSI 编码器功能设定	7-103
7.7.7.4 Pulse Encoder 功能设定	7-109
7.7.7.5 Capture/Compare 功能设定	7-116
7.7.7.6 Pulse Output 功能设定	7-121
7.7.7.7 高速 IO 错误确认	7-126
7.7.8 其它功能	7-127
7.7.8.1 更改现在位置	7-127
7.7.8.2 软件极限	7-127
7.7.8.3 追随误差设定	7-128
7.7.8.4 挺杆点功能.....	7-129
7.7.8.5 捕捉位置	7-129
7.8 范例程序	7-131
7.8.1 设备架构	7-131
7.8.1.1 使用设备	7-131

7.8.1.2 设定	7-131
7.8.2 基本项目范例	7-131
7.8.2.1 伺服系统 Servo On	7-133
7.8.2.2 单轴错误重置与监控	7-134
7.8.2.3 监控指令错误	7-137
7.8.2.4 单轴操作立即停止	7-139
7.8.2.5 原点回归	7-141
7.8.2.6 绝对值定位	7-144
7.8.2.7 凸轮动作中切换凸轮表	7-145
7.8.2.8 凸轮动作中执行主轴的相位补偿	7-154
7.8.2.9 移动控制过程中变更当前位置	7-161
7.8.2.10 主从轴齿轮啮合过程中执行位移追加	7-166
第 8 章 OPC UA Server	
8.1 OPC UA Server 简介	8-2
8.1.1 建立项目中的 OPC UA 存取设定	8-2
8.2 建立一般联机	8-5
8.2.1 使用「UaExpert」Client 建立一般联机	8-5
8.3 建立加密联机	8-9
8.3.1 OPC UA Server 启动认证	8-9
8.3.2 设定用户账号密码	8-9
8.3.3 信任凭证设定	8-12
8.3.4 使用「Prosys OPC UA Client」建立加密联机	8-16
第 9 章 通讯功能	
9.1 EtherCAT 功能	9-4
9.1.1 EtherCAT 总线介绍	9-4
9.1.2 EtherCAT 主站设定	9-5
9.1.3 EtherCAT 从站设定	9-7
9.1.4 EtherCAT 从站参数备份还原	9-10
9.1.4.1 参数备份流程说明	9-10
9.1.4.2 备份还原介绍	9-11
9.1.4.2.1 参数备份操作	9-13

9.1.4.2.2 参数还原操作	9-15
9.2 串行 (Serial) 通讯	9-17
9.2.1 Modbus Serial Port 元件	9-17
9.2.1.1 Modbus Serial Port 元件建立	9-17
9.2.1.2 Modbus Serial Port 组件页面说明	9-19
9.2.2 Modbus Serial Master	9-20
9.2.2.1 Modbus Serial Master 组件建立	9-20
9.2.2.2 Modbus Serial Master 设定说明	9-23
9.2.2.3 Modbus Serial Slave 设定说明	9-26
9.2.3 Modbus Serial Slave	9-31
9.2.3.1 Modbus Serial Slave 组件建立	9-31
9.2.3.2 Modbus Serial Slave 设定说明	9-33
9.3 以太网 (Ethernet) 通讯	9-35
9.3.1 Ethernet 组件	9-35
9.3.1.1 Ethernet 组件建立	9-35
9.3.1.2 Ethernet 组件页面说明	9-36
9.3.2 Modbus TCP Master	9-40
9.3.2.1 Modbus TCP Master 组件建立	9-40
9.3.2.2 Modbus TCP Master 设定说明	9-42
9.3.2.3 Modbus TCP Slave 设定说明	9-43
9.3.3 Modbus TCP Slave	9-49
9.3.3.1 Modbus TCP Slave 组件建立	9-49
9.3.3.2 Modbus TCP Slave 设定说明	9-51
9.4 EtherNet/IP 通讯	9-53
9.4.1 EtherNet/IP 简介	9-53
9.4.1.1 EtherNet/IP 介绍	9-53
9.4.1.2 常见网络名词说明	9-53
9.4.1.3 Ethernet 特色	9-54
9.4.1.3.1 Delta EIP 系统架构图	9-54
9.4.1.3.2 EIP 特点	9-54
9.4.2 EtherNet/IP Scanner Function	9-55
9.4.2.1 建立变频器 MS300 数据读写	9-55
9.4.2.1.1 系统硬件配置	9-55
9.4.2.1.2 隐性报文数据读写设定	9-55

9.4.2.1.3 显性报文 CIP Object 数据读写设定	9-61
9.4.2.2 建立 AS00SCM-A (AS-FEN02 通讯卡) 数据读写	9-62
9.4.2.2.1 设定 AS00SCM-RTU 右侧 IO 模块配置	9-62
9.4.2.2.2 导入 AS00SCM-RTU EDS 文件	9-64
9.4.2.2.3 设定 AS00SCM-RTU 相关 EtherNet/IP 参数	9-71
9.4.2.2.4 AS00SCM-RTU 右侧 IO 模块操作	9-73
9.4.2.2.5 AS00SCM-RTU 模块参数信息	9-76
9.4.3 EtherNet/IP Adapter Function	9-76
9.4.3.1 Studio 5000 软件操作	9-76
9.4.3.1.1 架构	9-77
9.4.3.1.2 项目建立	9-77
9.4.3.2 Scanner 建立	9-77
9.4.3.2.1 模块建立	9-77
9.4.3.3 Adapter 联机	9-78
9.4.3.3.1 EDS 建立	9-78
9.4.3.3.2 EDS 载入	9-82
9.4.3.3.3 Adapter 建立 EDS 加载	9-82
9.4.3.3.4 下载	9-82
9.4.3.3.5 数据交换	9-83
9.4.4 CIP Object	9-84
9.4.4.1 Object list	9-84
9.4.4.2 数据类型定义 (Data Type)	9-84
9.4.4.3 Identity Object (Class ID : 01 Hex)	9-87
9.4.4.4 Assembly Object (Class ID : 04 Hex)	9-89
9.4.4.5 TCP/IP Interface Object (Class ID : F5 Hex)	9-89
9.4.4.6 Ethernet Link Object (Class ID : F6 Hex)	9-92
9.4.5 台达 EIP 产品一览表	9-94
9.4.5.1 台达 EIP 产品列表支持 Adapter 功能	9-94
9.4.5.2 台达 EIP 产品支持 Scanner 功能	9-94
9.5 网络安全	9-95
9.6 FTP 功能	9-95
9.6.1 FTP Server	9-95
9.6.1.1 FTP Server 规格	9-95
9.6.1.2 FTP Server 功能启动流程	9-96

9.6.1.3 FTP Client 联机 (AX308E 系列控制器为 FTP Server)	9-100
9.6.2 FTP Client	9-100

附录 A 故障排除

A.1 故障排除	A-3
A.1.1 基本检查	A-3
A.1.2 清除错误状态	A-3
A.1.3 故障排除流程	A-4
A.1.4 查询错误记录	A-4
A.2 CPU 模块故障排除	A-7
A.2.1 ERROR 灯号一般闪烁 (亮 0.5 秒、暗 0.5 秒)	A-7
A.2.2 ERROR 灯号快速闪烁 (亮 0.2 秒、暗 0.2 秒)	A-8
A.2.3 ERROR 灯号慢速闪烁 (亮 1 秒、暗 3 秒)	A-8
A.2.4 BAT. LOW 灯号常亮	A-8
A.2.5 BAT. LOW 灯号一般闪烁 (亮 0.5 秒、暗 0.5 秒)	A-8
A.2.6 其它	A-9
A.3 函式库故障排除	A-10
A.3.1 DL_BuiltInIO_AX3	A-10
A.3.2 运动指令相关	A-13
A.4 模块故障排除	A-14
A.4.1 模拟 (AD/DA/XA) 及温度 (RTD/TC) 模块故障排除	A-14
A.4.1.1 ERROR 灯常亮	A-14
A.4.1.2 ERROR 灯闪烁 (亮 0.2 秒、暗 0.2 秒)	A-14
A.4.2 AS02LC 称重模块故障排除	A-15
A.4.2.1 ERROR 灯常亮	A-15
A.4.2.2 ERROR 灯闪烁 (亮 0.2 秒、暗 0.2 秒)	A-15
A.4.3 AS02HC 高速计数模块故障排除	A-15
A.4.3.1 ERROR 灯常亮	A-15
A.4.3.2 ERROR 灯闪烁 (亮 0.5 秒、暗 0.5 秒)	A-16
A.4.4 AS02/04PU 定位模块故障排除	A-17
A.4.4.1 ERROR 灯闪烁 (亮 0.2 秒、暗 0.2 秒)	A-17
A.5 CPU 模块错误代码对应灯号及状态说明	A-18
A.5.1 CPU 模块错误代码对应灯号及状态	A-18

A.5.2	模拟与温度模块错误代码对应灯号.....	A-19
A.5.3	AS02LC 称重模块错误代码对应灯号	A-20
A.5.4	AS02HC 高速计数模块错误代码对应灯号	A-21
A.5.5	AS02/04PU 定位模块错误代码对应灯号.....	A-21

MEMO

第1章 简介

目录

1.1 概述.....	1-2
1.1.1 相关手册.....	1-2
1.1.2 机种简易说明	1-2
1.2 DIADesigner-AX 软件概述	1-8
1.2.1 特色简介.....	1-8

1.1 概述

本手册描述 AX-3 系列 PLC 主机程序设计相关的内容介绍，基本指令及应用指令操作方法说明。

1.1.1 相关手册

AX-3 系列 PLC 相关手册的组成如下表所示。请搭配使用。

- AX-3 系列快速入门手册：
主要介绍如何让用户在详细阅读完所有手册之前，能先快速的建立并使用本系统。
- AS 系列硬件及操作手册：
电气规格、主机与模块配线、外观及尺寸等。
- DIADesigner-AX 使用手册：
内容包含软件操作方式、程序编辑语言（梯形图、顺序功能图、ST (Structured Text)、和功能块)、程序组织单元 (POU) 以及任务 (Task) 的概念、以及运动控制程序的编辑方式。
- AX-3 系列操作手册：
CPU功能介绍、各项装置、I/O配置、硬件电气规格与外观尺寸、运动控制系统概念与基本设定及故障排除等。
- AS 系列模块手册：
特殊模块使用介绍，例如网络模块、模拟模块及温度模块等。
- AX 系列运动控制指令手册：
介绍运动控制程序编辑所需使用之单轴/多轴运动控制指令。
- AX 系列标准指令手册
说明编成 PLC 程序所需使用之应用指令。

1.1.2 机种简易说明

分类	机种名称	说明
电源模块	AS-PS02	输入：100-240VAC · 50/60Hz 交流电源输入 输出 (供 PLC 内部使用)：24VDC/2A · 48W
	AS-PS02A	输入：100-240VAC · 50/60Hz 交流电源输入 输出 (供 PLC 内部使用)：24VDC/1.5A · 36W 输出 (供外部使用)：24VDC/0.5A · 12W
AX-3 系列 逻辑控制 CPU 模块	AX-300NA0PA1	CPU 模块 · 内建 Ethernet 2 port switch · RS-485*1 · RS-232*1 · USB 通讯口 · Micro SD 卡接口，程序容量 32MB · 数据容量 32MB · 采用脱落式端子
	AX-324NA0PA1P	CPU 模块 · 晶体管 (PNP) 输出 · 内建 16 DI (200KHz) · 8 DO (200KHz) · Ethernet 2 port switch · RS-485*1 · RS-232*1 · USB 通讯口 · Micro SD 卡接口，程序容量 32MB · 数据容量 32MB · 采用脱落式端子

分类	机种名称	说明
AX-3 系列 运动控制 CPU 模块	AX-304ELA0PA1T	4 轴运动控制 CPU 模块，晶体管(NPN)输出，内建 16 DI(200KHz)、8 DO (200KHz)、Ethernet 2 port switch、EtherCAT*1、RS-485*1、RS-232*1、USB 通讯口、Micro SD 卡接口，程序容量 32MB，数据容量 32MB，采用脱落式端子
	AX-304ELA0PA1P	4 轴运动控制 CPU 模块，晶体型(PNP)输出，内建 16 DI(200KHz)、8 DO (200KHz)、Ethernet 2 port switch、EtherCAT*1、RS-485*1、RS-232*1、USB 通讯口、Micro SD 卡接口，程序容量 32MB，数据容量 32MB，采用脱落式端子
	AX-308EA0MA1T	8 轴运动控制 CPU 模块，晶体型 (NPN) 输出，内建 Relative Encoder*2、SSI*1、16 DI (200KHz)、8 DO (200KHz)、Ethernet 2 port switch、EtherCAT*1、RS-485*1、RS-232*1、USB 通讯口、Micro SD 卡接口，程序容量 32MB，数据容量 32MB，采用脱落式端子
	AX-308EA0MA1P	8 轴运动控制 CPU 模块，晶体管 (PNP) 输出，内建 Relative Encoder*2、SSI*1、16 DI (200KHz)、8 DO(200KHz NPN)、Ethernet 2 port switch、EtherCAT*1、RS-485*1、RS-232*1、USB 通讯口、Micro SD 卡接口，程序容量 32MB，数据容量 32MB，采用脱落式端子
	AX-316EA0MA1T	16 轴运动控制 CPU 模块，晶体型 (NPN) 输出，内建 Relative Encoder*2、SSI*1、16 DI (200KHz)、8 DO (200KHz)、Ethernet 2 port switch、EtherCAT*1、RS-485*1、RS-232*1、USB 通讯口、Micro SD 卡接口，程序容量 32MB，数据容量 32MB，采用脱落式端子
	AX-332EP0MB1T	32 轴运动控制 CPU 模块，晶体型 (NPN) 输出，内建 Relative Encoder*1、SSI*1、6 DI (200KHz)、6 DO (200KHz)、Ethernet 2 port、EtherCAT*1、RS-485/RS-422 *1、Micro SD 卡接口，程序容量和数据容量共 1GB，采用脱落式端子
	AX-364ELA0MA1T	64 轴运动控制 CPU 模块，晶体管 (NPN) 输出，内建 Relative Encoder*2、SSI*1、16 DI (200KHz)、8 DO (200KHz)、Ethernet 2 port switch、EtherCAT*1、RS-485*1、RS-232*1、USB 通讯口、Micro SD 卡接口，程序容量 32MB，数据容量 32MB，采用脱落式端子
数位 I/O 模块	AS08AM10N-A	24VDC，5mA，8 点输入，弹片式端子台
	AS08AN01P-A	5 ~ 30VDC，0.5A/点，4A/共点，8 输出点，源型输出，弹片式端子台
	AS08AN01R-A	240VAC/24VDC，2A/点，8A/共点，8 输出点，继电器，弹片式端子台
	AS08AN01T-A	5 ~ 30VDC，0.5A/点，4A/共点，8 输出点，漏型输出，弹片式端子台
	AS16AM10N-A	24VDC，5mA，16 点输入，弹片式端子台
	AS16AN01P-A	5 ~ 30VDC，0.5A/点，4A/共点，16 输出点，源型输出，弹片式端子台

分类	机种名称	说明
	AS16AN01R-A	240VAC/24VDC · 2A/点 · 8A/共点 · 16 输出点 · 继电器 · 弹片式端子台
	AS16AN01T-A	5 ~ 30VDC · 0.5A/点 · 4A/共点 · 16 输出点 · 漏型输出 · 弹片式端子台
	AS16AP11P-A	24VDC · 5mA · 8 输入点 · 5 ~ 30VDC · 0.5A/点 · 4A/共点 · 8 输出点 · 源型输出 · 弹片式端子台
	AS16AP11R-A	24VDC · 5mA · 8 输入点 · 240VAC/24VDC · 2A/点 · 8A/共点 · 8 输出点 · 继电器 · 弹片式端子台
	AS16AP11T-A	24VDC · 5mA · 8 输入点 · 5 ~ 30VDC · 0.5A/点 · 4A/共点 · 8 输出点 · 漏型输出 · 弹片式端子台
	AS32AM10N-A	24VDC · 3.2mA · 32 点输入 · 牛角座连接器 (MIL)
	AS32AN02T-A	5 ~ 30VDC · 0.1A/点 · 3.2A/共点 · 32 点输出 · 漏型输出 · 牛角座连接器 (MIL)
	AS64AM10N-A	24VDC · 3.2mA · 64 点输入 · 牛角座连接器 (MIL)
	AS64AN02T-A	5 ~ 30VDC · 0.1A/点 · 3.2A/共点 · 64 点输出 · 漏型输出 · 牛角座连接器 (MIL)
模拟 I/O 模块	AS04AD-A	4 通道模拟信号输入 16 位分辨率 0~10V · 0/1~5V · -5V~+5V · -10~+10V · 0/4~20mA · -20mA~+20mA 转换时间：2ms/通道
	AS08AD-B	8 通道模拟信号输入 16 位分辨率 0~10V · 0/1~5V · -5V~+5V · -10~+10V · 转换时间：2ms/通道
	AS08AD-C	8 通道模拟信号输入 16 位分辨率 0/4~20mA · -20mA~+20mA 转换时间：2ms/通道
	AS04DA-A	4 通道模拟信号输出 12 位分辨率-10v~10V · 0~20mA · 4~20mA 转换时间：2ms/通道
	AS06XA-A	4 通道模拟信号输入 16 位分辨率 · 0~10V · 0/1~5V · -5V~+5V · -10~+10V · 0/4~20mA · -20mA~+20mA 转换时间：2ms/通道 2 通道模拟信号输出 12 位分辨率-10V~10V · 0~20mA · 4~20mA 转换时间：2ms/通道

分类	机种名称	说明
	AS02ADH-A	2 通道模拟信号输入 16 位分辨率 · 0~10V · 0/1~5V · -5V~+5V · -10~+10V · 0/4~20mA · -20mA~+20mA 高速转换 · 最快转换时间：20us 全隔离（包含通道间隔离） 提供 2000 笔数值、峰值与触发记录功能 提供 FIR 低通与带通滤波功能
温度模块	AS04RTD-A	4 通道 2 线式或 3 线式 RTD 温度感测 传感器型式：Pt100 / Ni100 / Pt1000 / Ni1000 / JPt100 / LG-Ni1000 / Cu50 / Cu100 / 0~300Ω / 0~3000Ω 分辨率：0.1°C/0.1°F（16 位转换器） 转换时间：200ms/通道
	AS06RTD-A	6 通道 2 线式或 3 线式 RTD 温度感测 传感器型式：Pt100 / Ni100 / Pt1000 / Ni1000 / JPt100 / LG-Ni1000 / Cu50 / Cu100 / 0~300Ω / 0~3000Ω 分辨率：0.1°C/0.1°F（16 位转换器） 转换时间：200ms/通道
	AS04TC-A	4 通道热电耦温度感测 传感器型：J、K、R、S、T、E、N、B 或 -100~+100mV 分辨率：0.1°C/0.1°F（24 位转换器） 转换时间：200ms/通道
	AS08TC-A	8 通道热电耦温度感测 传感器型：J、K、R、S、T、E、N、B 或 -100~+100mV 分辨率：0.1°C/0.1°F（24 位转换器） 转换时间：200ms/通道
定位模块	AS02PU-A	2 轴运动控制 5~24VDC · 1 组（A/B/Z 相）差动输入 · 硬件输入带宽最高 200KHz 24VDC · 5mA · 5 点输入 · 硬件输入带宽最高 1KHz 5VDC · 2 轴（4 点）差动输出 · 最高输出频率 200KHz
	AS04PU-A	4 轴运动控制 24VDC · 5mA · 6 点输入 · 硬件输入带宽最高 1KHz 5~30VDC · 0.1A · 4 轴（8 点）NPN 输出 · 最高输出频率 100KHz
计数模块	AS02HC-A	2 通道高速计数模块 计数方式可选择脉冲型输入（最大 200kHz）或 SSI 通讯接口输入（最大 1.25MHz） 4 点开集极输出 · 5~30VDC · 0.1A · 可搭配高速比较输出

分类	机种名称	说明
称重模块	AS02LC-A	2 通道 4 线式或 6 线式荷重传感器 (Load Cell) 特征值 : 1、2、4、6、20、40、80 mV/V 最高准确度 : 满刻度的 0.04% ADC 分辨率 : 24 位 转换时间 : 可选择 2.5 ~ 400ms (共 9 项)
程序编辑 电缆	UC-PRG015-01A (1.5M)	PLC 连接 PC 之程序规划电缆 (mini USB)。
	UC-PRG030-01A (3M)	PLC 连接 PC 之程序规划电缆 (mini USB)。
	UC-PRG030-20A (3M)	PLC 连接 PC 之程序规划电缆 (RJ45)。
I/O 连接线	UC-ET010-24B (1M) UC-ET020-24B (2M) UC-ET030-24B (3M)	I/O 连接线 (牛角座)(40Pin ↔ 40Pin)(屏蔽) · 适用 AS32AM10N-A、AS32AN02T-A、AS64AM10N-A 及 AS64AN02T-A 机种。
	UC-ET010-24D (1M) UC-ET020-24D (2M) UC-ET030-24D (3M)	I/O 连接线 (牛角座)(40Pin ↔ 20Pin x2)(屏蔽) · 适用 AS32AM10N-A、AS32AN02T-A、AS64AM10N-A 及 AS64AN02T-A 机种。
配线模块	UB-10-ID16A	I/O 配线模块 (16 点输入/输出)(20-Pin 牛角座) · 适用 AS32AM10N-A、AS32AN02T-A、AS64AM10N-A 及 AS64AN02T-A 机种。
	UB-10-ID32A	I/O 配线模块 (32 点输入) (40-Pin 牛角座) · 适用 AS32AM10N-A 及 AS64AM10N-A。
	UB-10-IO32D	牛角座转换弹片型端子座 · 40-Pin 牛角座直接转换为 40-Pin 弹片型端子座 · 适合 AS32AM10N-A 及 AS32AN02T-A。
	UB-10-OR16A	I/O 配线模块 (16 点继电器输出) (20-Pin 牛角座) · NPN · 适用 AS32AN02T-A 及 AS64AN02T-A。
配线模块	UB-10-OR16B	I/O 配线模块 (16 点继电器输出) (20-Pin 牛角座) · PNP。
	UB-10-OT32A	I/O 配线模块 (32 点晶体管输出) (40-Pin 牛角座) · NPN · 适用 AS32AN02T-A 及 AS64AN02T-A。
ECAT 运 动控制电 缆	UC-EMC003-02A	EtherCAT 通讯连接电缆 · 0.3M
	UC-EMC005-02A	EtherCAT 通讯连接电缆 · 0.5M
	UC-EMC010-02A	EtherCAT 通讯连接电缆 · 1M
	UC-EMC020-02A	EtherCAT 通讯连接电缆 · 2M
	UC-EMC050-02A	EtherCAT 通讯连接电缆 · 5M
	UC-EMC100-02A	EtherCAT 通讯连接电缆 · 10M
	UC-EMC200-02A	EtherCAT 通讯连接电缆 · 20M
	UC-EMC003-02B	EtherCAT 通讯连接电缆 · 0.3M
	UC-EMC005-02B	EtherCAT 通讯连接电缆 · 0.5M
	UC-EMC010-02B	EtherCAT 通讯连接电缆 · 1M

分类	机种名称	说明
	UC-EMC020-02B	EtherCAT 通讯连接电缆 · 2M
	UC-EMC030-02B	EtherCAT 通讯连接电缆 · 3M
	UC-EMC050-02B	EtherCAT 通讯连接电缆 · 5M
	UC-EMC100-02B	EtherCAT 通讯连接电缆 · 10M

1.2 DIADesigner-AX 软件概述

DIADesigner-AX 是用于台达新一代控制器的 IEC 61131-3 编程工具，并采用了大量的应用指令，尤其是运动函数库。多语系的环境和友善的操作接口为用户提供了便利与有效率的开发环境。

1.2.1 特色简介

DIADesigner-AX 可应用于 AX-8 和 AX-3 系列。

- 支持 IEC 61131-3 的所有编辑器 (FBD、LD、ST、SFC) 以及各种不同的标准编辑器
- 功能强大且成熟的函数库概念可供应用程序重复使用
- 输入协助可用于数据的输入和配置
- 在所有 IEC 61131-3 编辑器中可使用鼠标和键盘进行人性化的编程
- 广泛的调试和在线功能，可快速优化应用程序代码并加快测试和调试速度
- 多种安全功能，可保护源程序代码并保护控制器的运行
- 可编程的设备来自不同的制造商
- 用户接口的可扩展性和适应性无须离开框架
- 开发工具和可用对象皆是透明的内部结构
- 许多无缝集成的工具可用于各种自动化的任务

有两种内建的配置

- HWCONFIG：用于配置系统的硬件和管理参数
- NWCONFIG：用于为 PLC 系统架构配置网络并管理数据交换

Motion 提供了各种运动控制解决方案，包含 PLCopen MC 标准功能块、G-Code 编辑器、E-CAM 编辑器和定位计划图表工具等。

- 支持 PLCopen POU 用于单轴和多轴运动
- 支持 PLCopen POU 的附加功能，例如诊断、停止和 CAM 控制器
- 用于不同任务的附加 POU，例如监控动态数据或跟随错误、操作 CAM 和 CAM 控制器
- 集成的图形 CAM 编辑器，具有丰富的配置选项
- 支持虚拟和逻辑轴
- 适用于众多 MODBUS 和 EtherCAT 协议的集成驱动器
- 将驱动器配置为标准的现场设备

第2章 规格与系统配置

目录

2.1	一般规格	2-2
2.2	CPU 模块规格	2-4
2.2.1	主机性能规格	2-4
2.2.2	电源与重量及电气规格	2-10
2.2.3	CPU 模块外观尺寸及部位介绍	2-12
2.2.4	输入输出端子配置 (点位定义)	2-19
2.2.5	主机配线	2-21
2.2.5.1	AX-304ELA0PA1T 配线	2-21
2.2.5.2	AX-304ELA0PA1P 配线	2-22
2.2.5.3	AX-308EA0MA1T、AX-316EAMA1T、AX-364ELA0MA1T 配线 ..	2-23
2.2.5.4	AX-308EA0M1P 配线	2-25
2.2.5.5	AX-324NA0PA1P 配线	2-26
2.2.5.6	AX-332EP0MB1T 配线	2-27
2.3	电源模块规格	2-29
2.3.1	一般规格	2-29
2.3.2	电源模块外观尺寸及部位介绍	2-29
2.3.3	端子配置	2-31
2.4	扩展模块	2-31

2.1 一般规格

项目	规格
操作环境温度	-20~55°C*1; -20~50°C (AX-332)
储存环境温度	-40~80°C
操作环境湿度	5~95% · 无结露
储存环境湿度	5~95% · 无结露
工作环境	无腐蚀性气体存在
安装位置	控制箱内
污染等级	2
防护等级	IP20
EMC 规范	请参考下表 EMI、EMS 与传导抗扰度测试。
耐振动	Tested with : 5 Hz ≤ f ≤ 8.4 Hz · constant amplitude 3.5 mm ; 8.4 Hz ≤ f ≤ 150 Hz · constant acceleration 1g Duration of oscillation : 10 sweep cycles per axis on each direction of the 3 mutually perpendicular axes 国际标准规范 IEC 61131-2 & IEC 60068-2-6 (TEST Fc)
冲击	Tested with : Half-sine wave : Strength of shock 15 g peak value · 11 ms duration ; Shock direction : The shocks in each in direction per axis · on 3 mutually perpendicular axes (total of 18 shocks) 国际标准规范 IEC 61131-2 & IEC 60068-2-27 (TEST Ea)
安全规范	IEC 61131-2、UL508
适用大气压	操作 : 1080 ~ 795hPa (相当于海拔 -1000 ~ 2000 公尺) 储存 : 1080 ~ 660hPa (相当于海拔 -1000 ~ 3500 公尺)
外壳防火等级	UL94V-0

*1 : 在使用前须于操作环境温度环境范围内放置一小时以上，并确保主机温度于可操作环境温度范围内方可使用。

● EMI

端口	频率范围	等级 (标准)	参考标准
外壳端口 (辐射) (在 10 公尺距离测量)	30-230 MHz	准峰值 40dB ($\mu\text{V}/\text{m}$)	IEC 61000-6-4
	230-1000 MHz	准峰值 47dB ($\mu\text{V}/\text{m}$)	
AC 电源端口 (传导)	0.15-0.5 MHz	准峰值 79dB (μV)	IEC 61000-6-4
		平均 66dB (μV)	
	0.5-30 MHz	准峰值 73dB (μV)	
		平均 60dB (μV)	

● EMS

环境现象	参考标准	测试		测试等级
静电放电	IEC 61000-4-2	接触		±4kV
		空气		±8kV
射频电磁场调幅	IEC 61000-4-3	80% AM · 1kHz 正弦波	2.0-2.7 GHz	1 V/m
			1.4-2.0 GHz	3 V/m
			80-1000 MHz	10 V/m
电源频率磁场	IEC 61000-4-8	60 Hz		30 A/m
		50 Hz		30 A/m

● 传导抗扰度测试

环境现象		快速瞬时脉冲	高能量浪涌	射频干扰
参考标准		IEC 61000-4-4	IEC 61000-4-5	IEC 61000-4-6
界面/ 通讯端口	特定界面/ 通讯端口	测试等级	测试等级	测试等级
数据传输	屏蔽电缆	1kV	1kV CM	10V
	非屏蔽电缆	1kV	1kV CM	10V
数位和 模拟 I/O	AC I/O (非屏蔽)	2kV	2kV CM 1kV DM	10V
	模拟或 DC I/O (非屏蔽)	1kV	1kV CM	10V
	所有屏蔽线 (接地)	1kV	1kV CM	10V
装置电源	AC 电源	2kV	2kV CM 1kV DM	10V
	DC 电源	2kV	0.5kV CM 0.5kV DM	10V
I/O 电源 和辅助电 源输出	AC I/O 和 AC 辅助电源	2kV	2kV CM 1kV DM	10V
	DC I/O 和 DC 辅助电源	2kV	0.5kV CM 0.5kV DM	10V

2.2 CPU 模块规格

2.2.1 主机性能规格

● 逻辑控制器

项目			AX-300NA ^{*1}	AX-324NA ^{*2}
处理时间	指令运行时间	LD 指令	5 ns	
		算术指令(LREAL 数据类型)	36 ns	
程序	程序容量	容量	8 MB	
	变量内存	有保持属性	Retain	768 KB (包含装置内存%M)
			Persist	128 KB
		无保持属性	16 MB	
装置内存 (%M)	尺寸	512 KB		
USB 埠	埠数		1	
	连接型式		Mini USB	
RS-232 埠	埠数		1	
	传输速率		9,600 、 19,200 、 38,400 、 57,600 、 76,800 、 115,200 bps	
	通讯格式		Stop bit : 1 、 2 ; Parity bit : None 、 Odd 、 Even ; Data bit : 7 、 8	
	通讯协议		MODBUS ASCII/RTU	
RS-485 埠	埠数		1	
	传输速率		9,600 、 19,200 、 38,400 、 57,600 、 76,800 、 115,200 bps	
	通讯格式		Stop bit : 1 、 2 ; Parity bit : None 、 Odd 、 Even ; Data bit : 7 、 8	
	通讯协议		MODBUS ASCII/RTU	
TCP	MODBUS TCP	联机数	32 (Server + Client)	
	SOCKET	TCP 联机数		
	MODBUS TCP	单笔联机最大数据长度	100 words	
	SOCKET	单笔指令最大数据长度	8 KB	

項目		AX-300NA* ¹	AX-324NA* ²
EtherNet/ IP	CIP 服务类型_ IO Connection	Scanner 联机数	12
		Adapter 联机数	1
		封包传送间隔时间 (RPI)	20 ~ 1,000ms (单位为 1ms)
		最大通讯能力	2,200 pps
		单笔联机最大数据长度	默认 100 bytes ° (O→T 最大 505 bytes · T→O 最大 509 bytes)
	CIP 服务类型_ Explicit Message	Class 3 / UCMM	Get_Attribute_Single (FB) Get_Attributes_All (FB) Set_Attribute_Single (FB) Set_Attributes_All (FB)
支持 CIP 对象		Identity 、 Message Router 、 Assembly 、 Connection Manager 、 Port 、 TCP/IP interface 、 Ethernet link 、 Vendor specific	
OPC UA 服务器	支持的配置文件、模型		PLCopen and OPC Foundation : OPC UA Information Model for IEC 61131-3
	预设端点、端口		TCP : 4840 (Reconfigurable via configuration file)
	Session (Client 端) 最大数量		5
	服务器整体的监控项目最大数量		1000
	监控项目的 Sampling Rate (ms)		100 、 300 、 500 、 1000 、 2500 、 5000
	服务器整体的订阅最大数量		100
	变量的最大可公开数量		10,000
	公共变量的 Value 属性数量		10,000
OPC UA 服务器	可公开的架构体定义数		100
	不可公共变量的限制		<ul style="list-style-type: none"> - More than three dimensional arrays - Array of Array - The OPC UA Stack will limit messages to about 300 kB. This is the maximum for values too. - Pointer variables, Interface variables - Structures containing pointers and interfaces

2

项目			AX-300NA ^{*1}	AX-324NA ^{*2}
	安全政策模式		None Sign - Basic256Sha256 SignAndEncrypt - Basic256Sha2566	
	应用认证	认证方法	X.509 标准	
		可储存的凭证数量	受信任的凭证：32 发行者凭证：32 拒绝清单：32	
	用户认证	认证方法	用户名称/密码 匿名 (Anonymous)	
IO 配置	扩充 IO 模块数量		32	
	I/O 数据容量		IN：8,192byte OUT：8,192byte	
	内建 IO	High speed counter	-	6 组 (200KHz)
记忆卡	SD 卡类型		Micro SD (SDHC · 32GB max)	
万年历	年、月、日、时、分、秒、星期		需自行安装市售 CR1620 电池 x1	

*1：AX-300NA 为机种 AX-300NA0PA1

*2：AX-324NA 为机种 AX-324NA0PA1P

● 运动控制器

项目			AX-304EL ^{*1}	AX-308EA ^{*2}	AX-316EA ^{*3}	AX-364EL ^{*4}	AX-332EP ^{*5}
处理时间	指令运行时间	LD 指令	5 ns				2.3 ns
		算数指令 (LREAL 数据类型)	36 ns				4.5 ns
程序	程序容量	容量	8 MB				128 MB
	变量内存	有保持属性	Retain	768 KB (包含装置内存%M)			3.5MB
			Persist	128 KB			
		无保持属性	16 MB				256 MB
装置内存 (%M)	尺寸	512 KB					
运动控制	控制轴数	控制轴最大值	4 轴	16 轴	32 轴	64 轴	32 轴
		EtherCAT 轴*	4 轴	8 轴	16 轴	64 轴	32 轴
		Pulse Out 轴	-	4 轴			1 轴

项目			AX-304EL*1	AX-308EA*2	AX-316EA*3	AX-364EL*4	AX-332EP*5	
	直线插补控制最大轴数	圆弧插补控制轴数	-	6 轴				
		轴组最大数	-	8 组			16 组	
	运动控制周期		与 EtherCAT 通讯的处理数据通讯周期相同					
	凸轮	凸轮数据点数	单一电子凸轮表的最大点数	-	256 点			32,767 点
			全部电子凸轮表的最大点数	-	20,480 点			655,340 点
		电子凸轮表最大数量	-	80			160	
Ethernet 端口	端口数		2					
	物理层		10BASE-T/100BASE-TX/1000BASE-T Switch				IEEE 802.3/802.3u/802.3ab 1G bps (Intel I210IT)	
	网络拓扑		星状、线型					
	传送速度		10/100/1000 Mbps					
	传输电缆		Category 5e 以上 · 100 公尺 (Max)					
	网络协议		ARP、IP、TCP、UDP、MODBUS TCP、EtherNet/IP、OPC UA					
USB 端口	端口数		1					
	连接型式		Mini USB				USB 2.0 (0.5A)	
RS-232 端口	端口数		1				无	
	传输速率		9,600、19,200、38,400、57,600、76,800、115,200 bps					
RS-232 端口	通讯格式		Stop bit : 1、2 ; Parity bit : None、Odd、Even ; Data bit : 7、8					
	通讯协议		MODBUS ASCII/RTU					
RS-485 端口	端口数		1					
	传输速率		9,600、19,200、38,400、57,600、76,800、115,200 bps					

项目		AX-304EL*1	AX-308EA*2	AX-316EA*3	AX-364EL*4	AX-332EP*5
	通讯格式	Stop bit : 1、2 ; Parity bit : None、Odd、Even ; Data bit : 7、8				
	通讯协议	MODBUS ASCII/RTU				
EtherCAT 通讯端口	EtherCAT 主站规格	Class B				
	物理层	100BASE-TX				IEEE 802.3/8 02.3u/8 02.3ab 1G bps (Intel I210IT)
	传送速度	100 Mbps				
	拓扑	Line、daisy chain and branching				
	传输电缆	Category 5e 以上、100 公尺 (Max)				
	最大子机数	16	64	64	96	256
	通讯周期	2,000µs ~ 32,000µs 注 : AX-332 能支持最小为 250us				
TCP	MODBUS TCP	联机数	32 (Server + Client)			
	SOCKET	TCP 联机数				
	MODBUS TCP	单笔联机最大数据长度	100 words			
	SOCKET	单笔指令最大数据长度	8 KB			
EtherNet/IP	CIP 服务类型_ IO Connection	Scanner 联机数	12			
		Adapter 联机数	1			
		报文传送间隔时间 (RPI)	20 ~ 1,000ms (单位为 1ms)			
		最大通讯能力	2,200 pps			
	CIP 服务类型_ Explicit Message	Class 3 / UCMM	Get_Attribute_Single (FB) Get_Attributes_All (FB) Set_Attribute_Single (FB) Set_Attributes_All (FB)			
		支持 CIP 对象	Identity、Message Router、Assembly、Connection Manager、Port、TCP/IP interface、Ethernet link、Vendor specific			

項目		AX-304EL*1	AX-308EA*2	AX-316EA*3	AX-364EL*4	AX-332EP*5	
OPC UA 服务器	支持的配置文件、模型	PLCopen and OPC Foundation : OPC UA Information Model for IEC 61131-3					
	默认端点、端口	TCP : 4840 (Reconfigurable via configuration file)					
	Session (Client 端) 最大数量	5					
	服务器整体的监控项目最大数量	1000					
	监控项目的 Sampling Rate (ms)	100 、 300 、 500 、 1000 、 2500 、 5000					
	服务器整体的订阅最大数量	100					
	变量的最大可公开数量	10,000					
	公共变量的 Value 属性数量	10,000					
	可公开的架构体定义数	100					
	不可公共变量的限制	<ul style="list-style-type: none"> - More than three dimensional arrays - Array of Array - The OPC UA Stack will limit messages to about 300 kB. This is the maximum for values too. - Pointer variables, Interface variables - Structures containing pointers and interfaces 					
OPC UA 服务器	安全政策模式		None Sign - Basic256Sha256 SignAndEncrypt - Basic256Sha2566				
	应用认证	认证方法	X.509 标准				
		可储存的凭证数量	受信任的凭证 : 32 发行者凭证 : 32 拒绝清单 : 32				
	用户认证	认证方法	用户名称/密码 匿名 (Anonymous)				
IO 配置	扩展 IO 模块数量		32				
	I/O 数据容量		IN : 8,192byte OUT : 8,192byte				
	内建 IO	Encoder	-	2		1	
		SSI	-	1			
		High speed counter	6 组 (200KHz)				
	Pulse out	-	4 组 (200KHz)		1 组 (200KHz)		
储存卡	SD 卡类型		Micro SD (SDHC · 32GB max)			Micro SD (SD3.0 /SDR50/SDXC)	

项目		AX-304EL*1	AX-308EA*2	AX-316EA*3	AX-364EL*4	AX-332EP*5
万年历	年、月、日、时、分、秒、星期	需自行安装电池*6				

*1：AX-304EL 包含机种 AX-304ELA0PA1T 及 AX-304ELA0PA1P

*2：AX-308EA 包含机种 AX-308EA0MA1T 及 AX-308EA0MA1P

*3：AX-316EA 为机种 AX-316EA0MA1T

*4：AX-364EL 为机种 AX-364ELA0MA1T

*5：AX-332EP 为机种 AX-332EP0MB1T

*6：AX-3 系列（除 AX-332 外）请购买市售 CR1620 3V 电池 x1；AX-332EP 电池配件请洽当地代理商

*注：EtherCAT 轴分为同步轴与定位轴，以下表格分别说明定位轴与同步轴最大建立轴数

机种	项目	定位轴最大轴数	同步轴最大轴数	定位轴+同步轴最大轴数
AX-304EL*1		4	-	4
AX-308EA*2		8	8	8
AX-316EA*3		16	16	16
AX-364EL*4		64	8	64
AX-332EP*5		32	32	32

*1：AX-304EL 包含机种 AX-304ELA0PA1T 及 AX-304ELA0PA1P

*2：AX-308EA 包含机种 AX-308EA0MA1T 及 AX-308EA0MA1P

*3：AX-316EA 为机种 AX-316EA0MA1T

*4：AX-364EL 为机种 AX-364ELA0MA1T

*5：AX-332EP 为机种 AX-332EP0MB1T

2.2.2 电源与重量及电气规格

机种	AX-300NA0PA1	AX-304ELA0PA1T/P AX-324NA0PA1P	AX-308EA0MA1T/P AX-316EA0MA1T AX-364ELA0MA1T	AX-332EP0MB1T
项目				
电源电压	24 VDC (20.4 VDC~28.8 VDC) (-15%~+20%)			
消耗功率 (W)	4	5	11	24
重量 (g)	240	300	380	390

● 机种内建数字输入点 (24VDC 直流) 电气规格

机种	AX-304ELA0PA1T/P、AX-308EA0MA1T/P、 AX-316EA0MA1T AX-324NA0PA1P、AX-364ELA0MA1T	AX-332EP0MB1T
项目		
输入点数	16	6
输入的连接方式	脱落式端子	
输入点类型	数字输入	
输入形式	直流 (漏型 Sink 或源型 Source)	

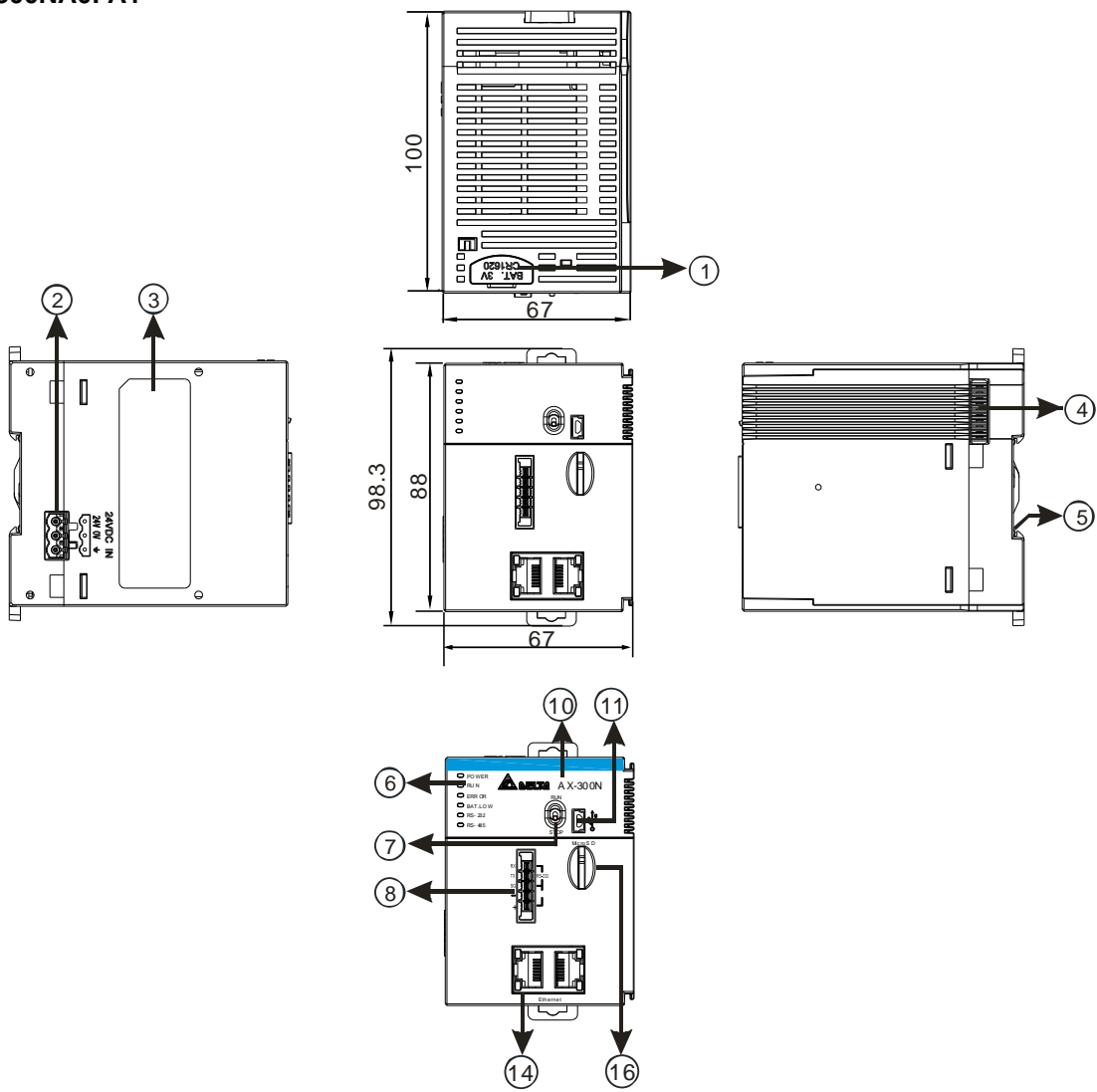
机种		AX-304ELA0PA1T/P、AX-308EA0MA1T/P、 AX-316EA0MA1T AX-324NA0PA1P、AX-364ELA0MA1T	AX-332EP0MB1T
项目			
输入电压/电流		24 VDC · 5mA	
动作准位	OFF→ON	>15VDC	
	ON→OFF	<5VDC	
反应时间	OFF→ON	2.5μs	
	ON→OFF	5 μs	
最大输入频率		200KHz	
输入阻抗		5.6kΩ	
输入信号形式		漏型输入 (Sink) : NPN 开集极输入形式 源型输入 (Source) : PNP 开集极输入形式	
输入电路隔离		光耦隔离	
输入动作显示		光耦驱动时 · 输入点指示灯亮	

● 内建数字输出点电气规格

机种		AX-304ELA0PA1T AX-308EA0MA1T AX-316EA0MA1T AX-364ELA0MA1T	AX-304ELA0PA1P AX-308EA0MA1P AX-324NA0PA1P	AX-332EP0MB1T
项目				
输出点数		8		6
输出的连接方式		脱落式端子		
输出点类型		晶体管-T (漏型)	晶体管-P (源型)	晶体管 (N-MOS)
电压规格		5~30 VDC		24 VDC (-15% ~ +20%)
最大负载	电阻性	0.1A/1 点		
	电感性	-		
	灯泡	-		
最大输出频率	电阻性	200KHz		
最大反应时间	OFF→ON	2.5μs		

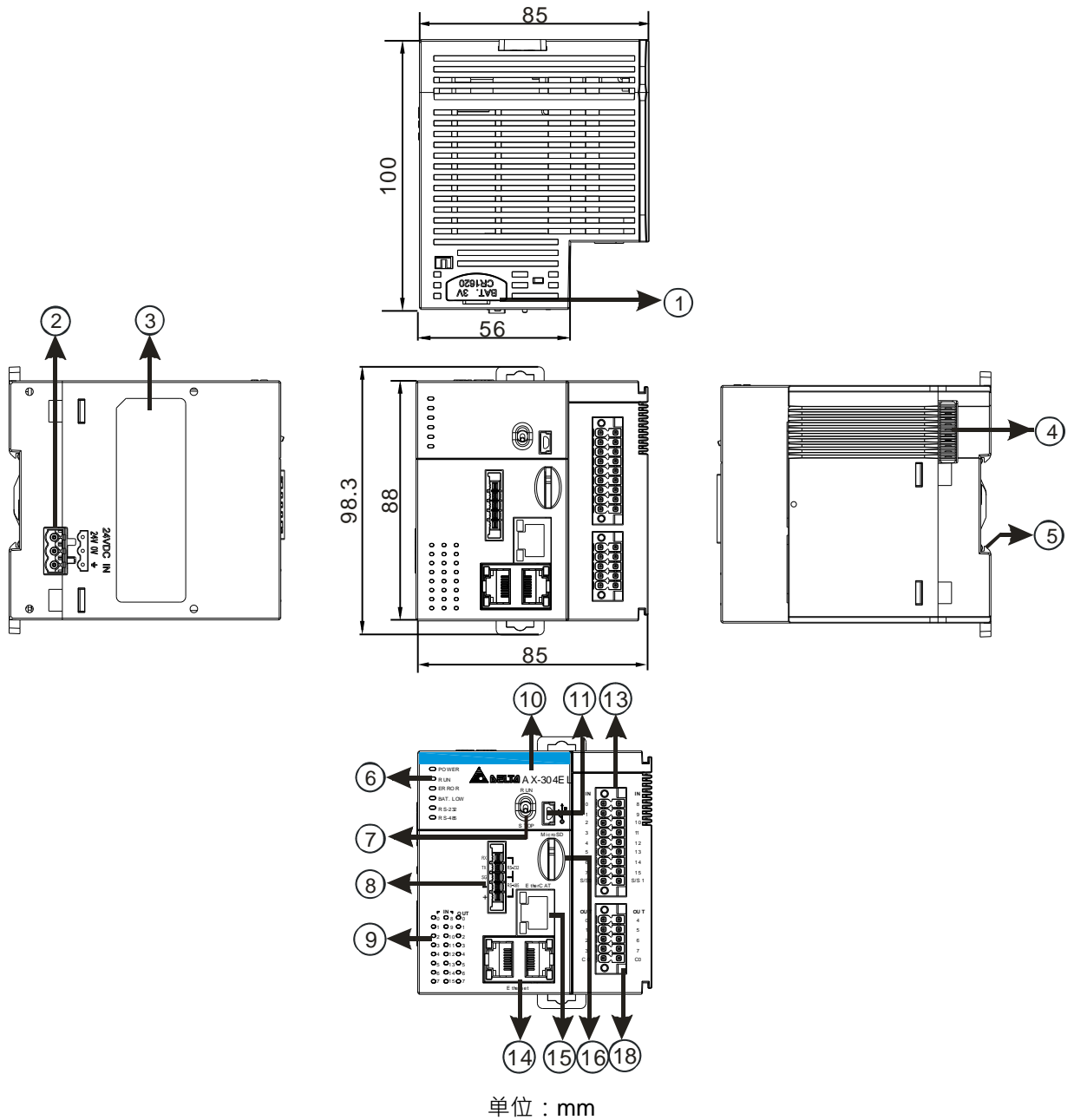
2.2.3 CPU 模块外观尺寸及部位介绍

● AX-300NA0PA1

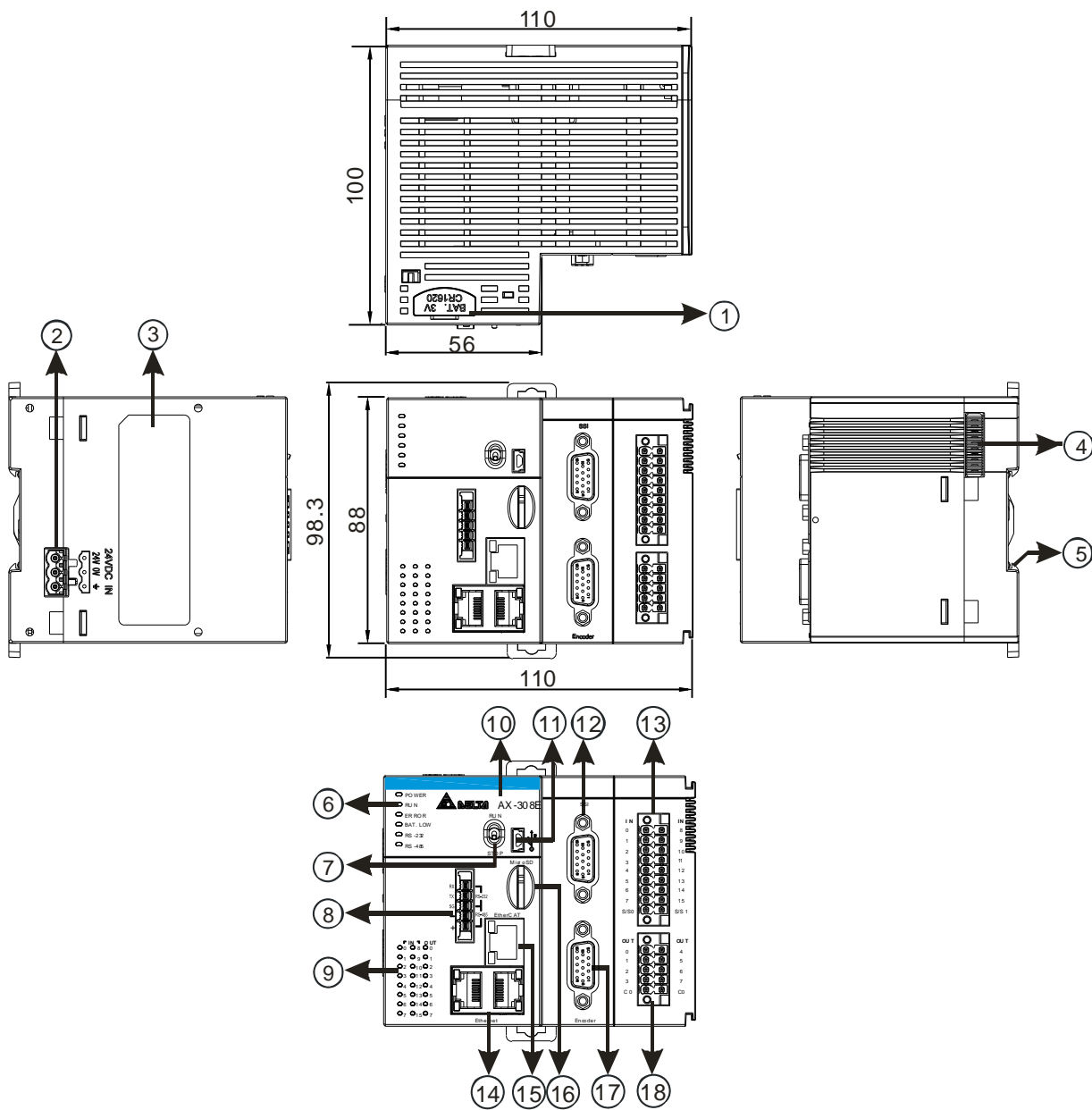


单位：mm

● AX-304ELA0PA1T / AX-304ELA0PA1P

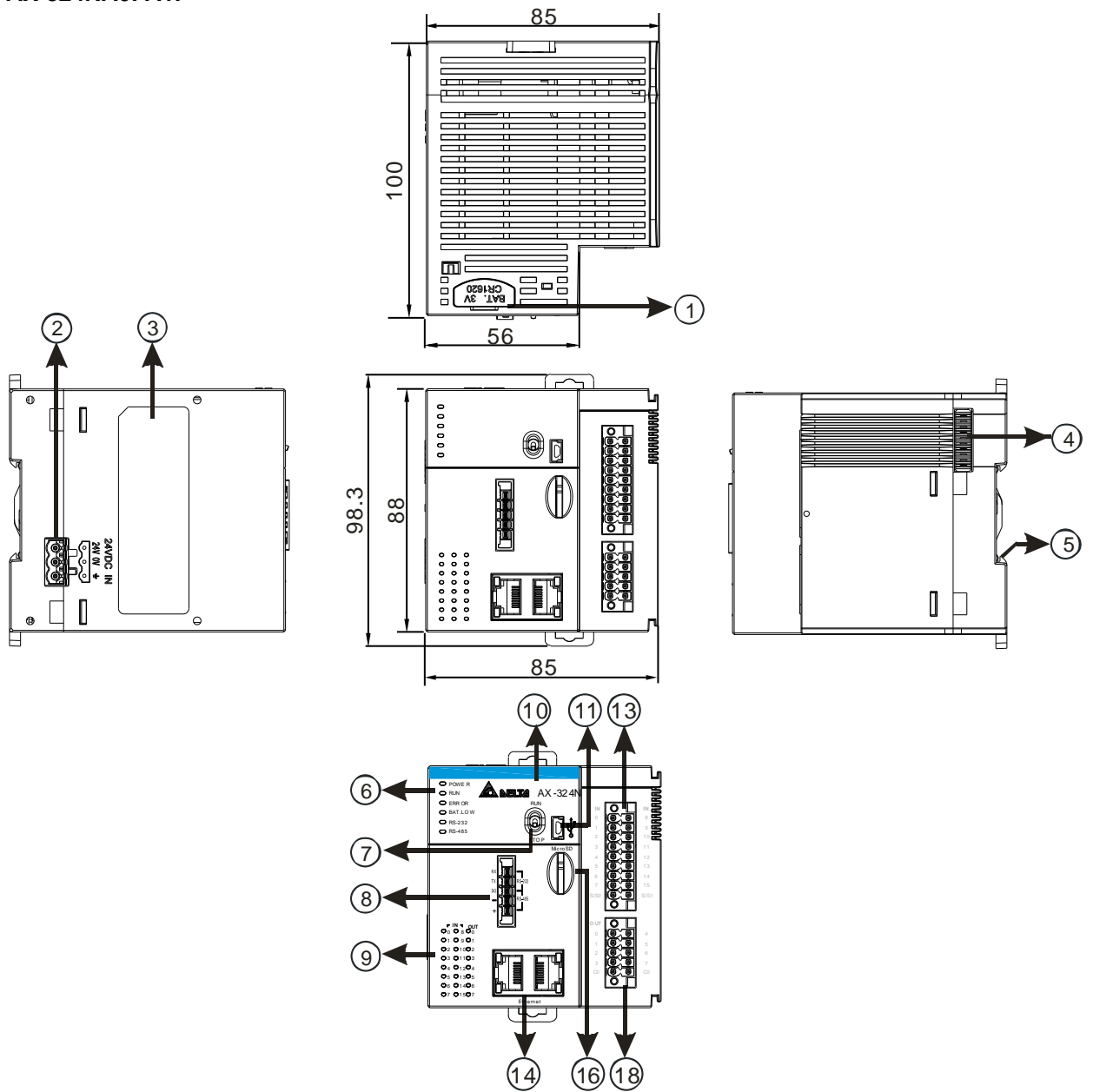


● AX-308EA0MA1T / AX-308EA0MA1P / AX-316EA0MA1T / AX-364ELA0MA1T



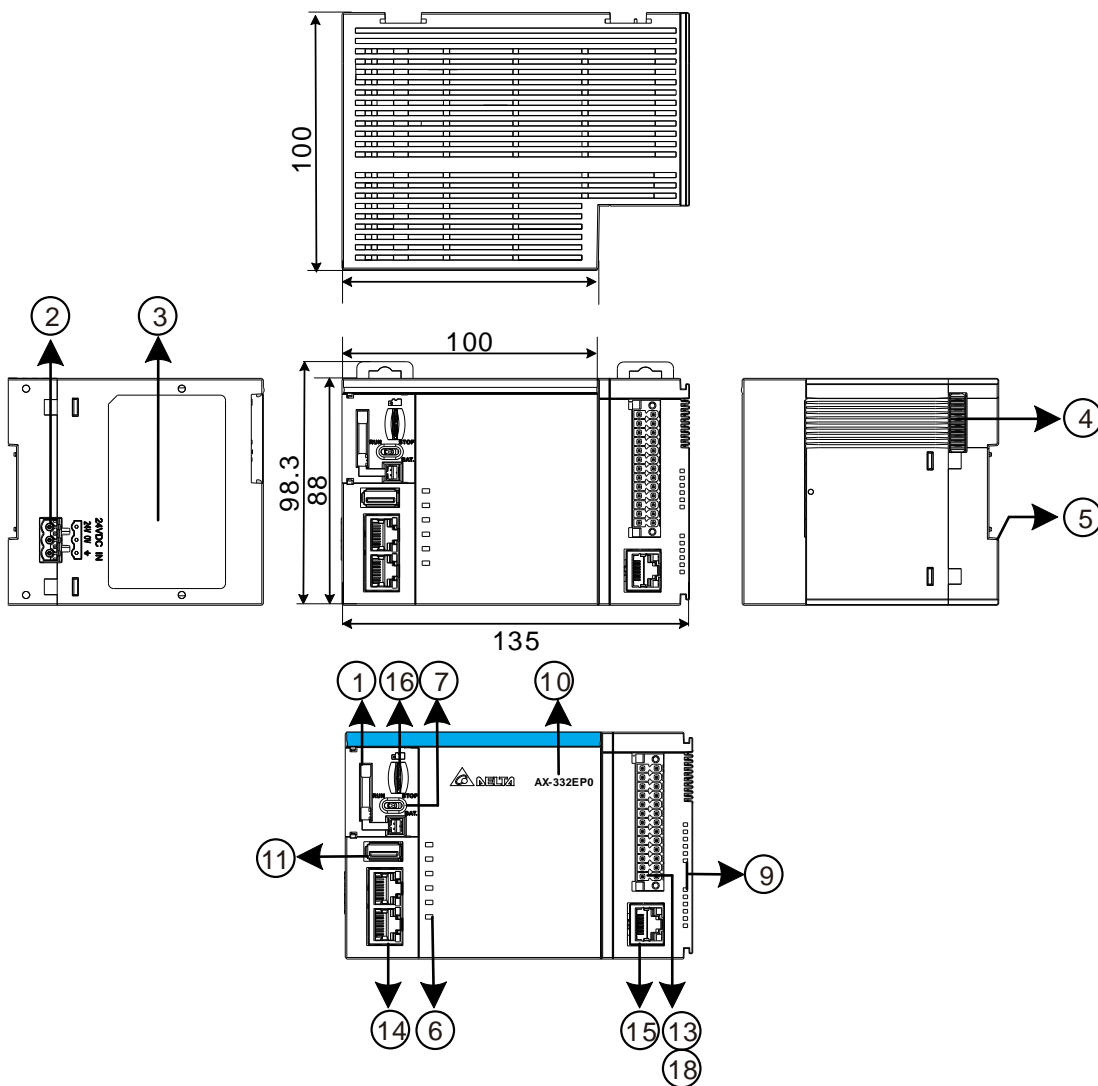
单位 : mm

● AX-324NA0PA1P



单位：mm

● AX-332EP0MB1T



单位：mm

序号	名称	说明
1	电池座	万年历功能所使用的电池（本产品不随机附赠电池）
2	电源输入口	供输入电源使用
3	标签	铭牌
4	扩展模块通讯端口	供连接模块使用
5	接地弹片	供以接地
6	POWER LED	指示 CPU 的电源状态

序号	名称	说明
	RUN LED	指示 CPU 的运行状态 常亮：用户程序执行中 灯灭：用户程序停止中 闪烁：用户程序处于错误检测模式中
	ERROR LED	指示 CPU 的错误状态 常亮：系统严重错误发生 灯灭：系统正常 闪烁：系统非严重错误发生
	BAT.LOW LED	指示 CPU 的电池状态
	COM1 LED COM2 LED	指示 COM 的通讯状态 灯灭：COM 无通讯 闪烁：COM 通讯中
	FB1 LED FB2 LED	指示 FB 的通讯状态 FB1 灯灭：ECAT 无通讯 FB1 常亮：ECAT 通讯无异常 FB1 闪烁：ECAT 通讯存在异常 FB2 灯灭：CAN 无通讯 FB2 常亮：CAN 通讯中
7	RUN/STOP 开关	RUN：执行用户程序运行 STOP：停止用户程序运行
8	COM PORT	提供 RS-485 与 RS-232 通讯接口
9	输入/输出指示灯	输入点导通时，输入指示灯亮起。 输出点导通时，输出指示灯亮起。
10	机种名称	主机机种名称
11	USB 通讯端口	AX-3 系列 (除 AX-332 外)：提供 mini USB 通讯接口 AX-332：提供 USB2.0 装置接口
12	SSI 通讯端口	提供 SSI ENCODER 通讯接口
13	输入点端子	提供输入点配线使用
14	Ethernet 通讯端口	提供 Ethernet Switch 通讯接口 指示灯 LINK (绿灯)： ■ 绿灯常亮：网络线已连接 ■ 灯灭：网络线未连接 指示灯 ACT (黄灯)： ■ 黄灯闪烁：数据传送/接收中 ■ 灯灭：无数据传送
15	EtherCAT 通讯端口	提供 EtherCAT 通讯接口 指示灯 LINK (绿灯)：

序号	名称	说明
		<ul style="list-style-type: none">■ 绿灯常亮：网络线已连接■ 灯灭：网络线未连接 指示灯 ACT (黄灯) : ACT <ul style="list-style-type: none">■ 黄灯闪烁：数据传送/接收中■ 灯灭：无数据传送
16	SD 卡插槽	提供 SD 卡储存接口
17	Encoder 通讯端口	提供增量型编码器通讯接口
18	输出点端子	提供输出点配线使用

2.2.4 輸入输出端子配置 (点位定义)

AX-304ELA0PA1T / AX-304ELA0PA1P			
		IN	
		0	8
		1	9
		2	10
		3	11
		4	12
		5	13
		6	14
		7	15
		S/S0	S/S1
		OUT	
		0	4
		1	5
		2	6
		3	7
		C0	C0

AX-308EA0MA1T / AX-308EA0MA1P / AX-316EA0MA1T / AX-364ELA0MA1T							
		SSI		ENCODER		IN	
		1	DATA+	1	A1+	0	8
		2	DATA-	2	A1-	1	9
		6	CLK+	10	B1+	2	10
		14	CLK-	11	B1-	3	11
		8	GND	4	Z1+	4	12
		15	5V	5	Z1-	5	13
				15	+5V1	6	14
				3	A2+	7	15
				9	A2-	S/S0	S/S1
				6	B2+	OUT	
				12	B2-	0	4
				13	Z2+	1	5
				14	Z2-	2	6
				7	+5V2	3	7
				8	0V	C0	C0

2

AX-324NA0PA1P

IN	
0	8
1	9
2	10
3	11
4	12
5	13
6	14
7	15
S/S0	S/S1
OUT	
0	4
1	5
2	6
3	7
C0	C0

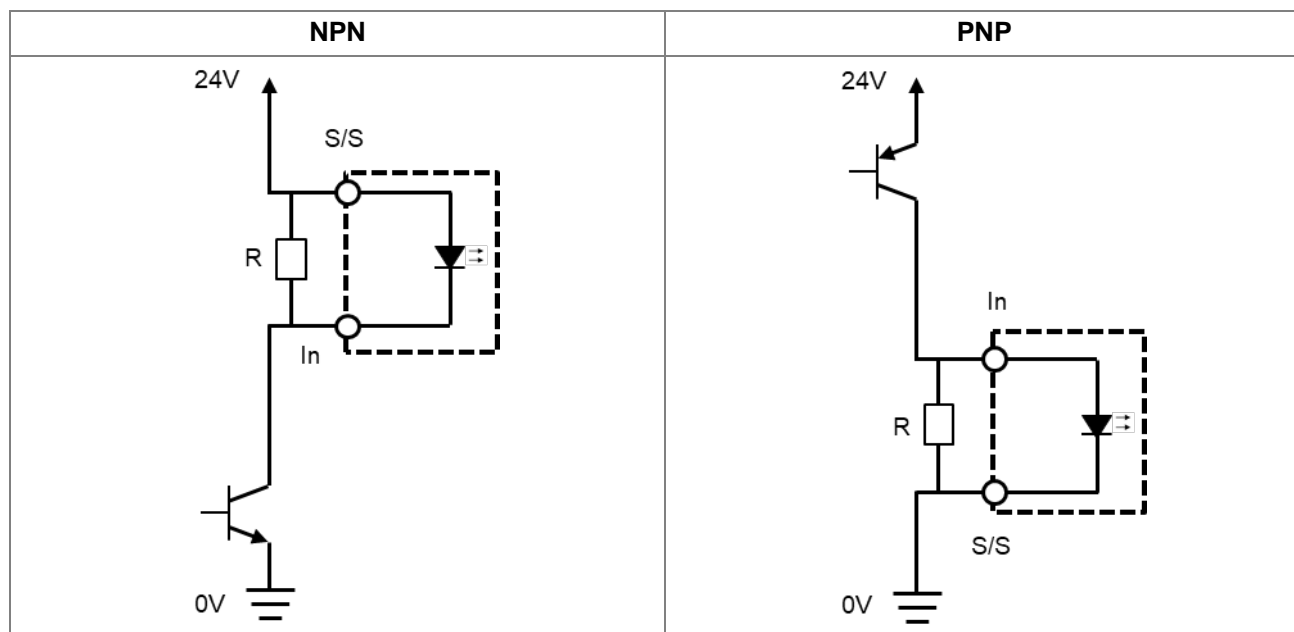
AX-332EP0MB1T

ENCODER			
1	TX+	2	TX-
3	SG	4	SG
5	RX+	6	RX-
7	A+	8	A-
9	B+	10	B-
11	Z+	12	Z-
13	IN 0	14	OUT 0
15	IN 1	16	OUT 1
17	IN 2	18	OUT 2
19	IN 3	20	OUT 3
21	IN 4	22	OUT 4
23	IN 5	24	OUT 5
25	S/S	26	C0

2.2.5 主机配线

主机配线注意事项：

1. 采用开集极 (NPN / PNP) 输出点连接 AX 主机高速输入点时，建议增加 (上拉/下拉) 并接电阻于指定 In 点和 S/S 点之间，电阻建议使用 3W / 470ohm 或 2W / 1Kohm 的电阻。
2. 若采用推挽式 (Push/Pull) 输出连接 AX 主机高速输入点，则不需要并接电阻。

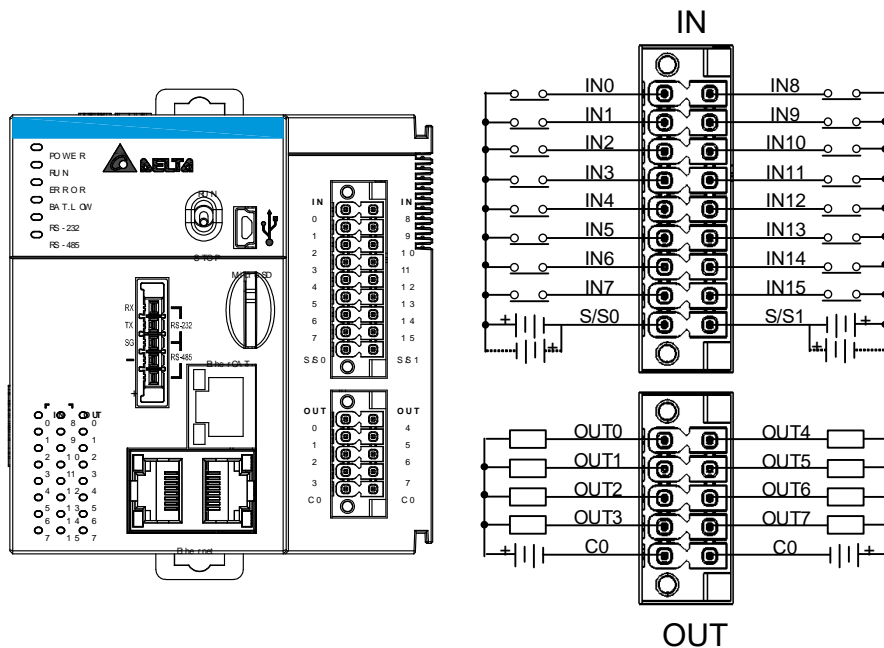


2.2.5.1 AX-304ELA0PA1T 配线

输入形式	直流 (漏型 Sink 或源型 Source)
输入电压/电流	24VDC · 5mA
输出点类型	晶体管-T (漏型)
输出电压/电流	5~30VDC · 0.1A

- 高速计数器配线，请参考 2.2.5.3 节。
- 一般输入/输出配线

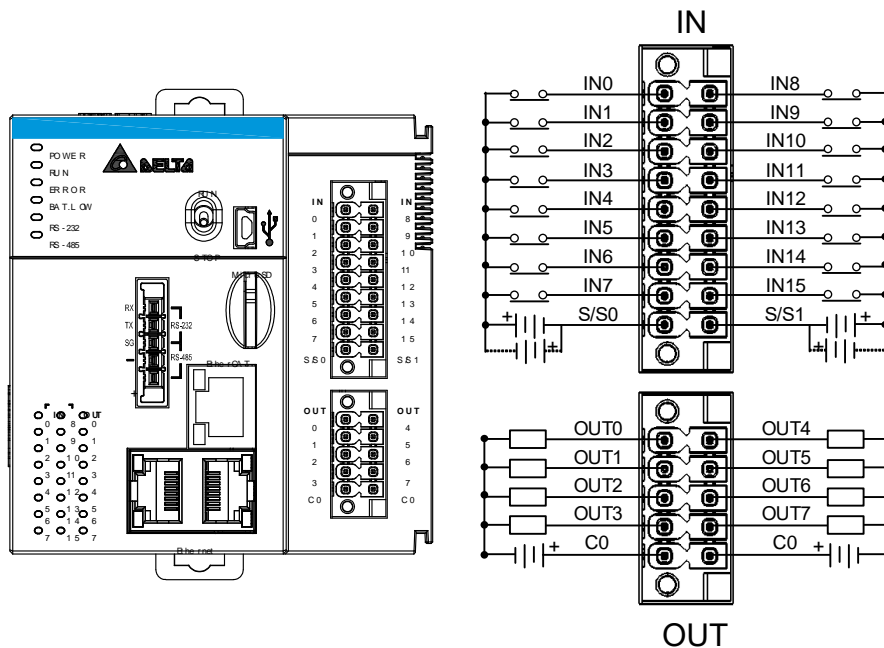
2



2.2.5.2 AX-304ELA0PA1P 配线

输入形式	直流 (漏型 Sink 或源型 Source)
输入电压/电流	24VDC · 5mA
输出点类型	晶体管-P (源型)
输出电压/电流	5~30VDC · 0.1A

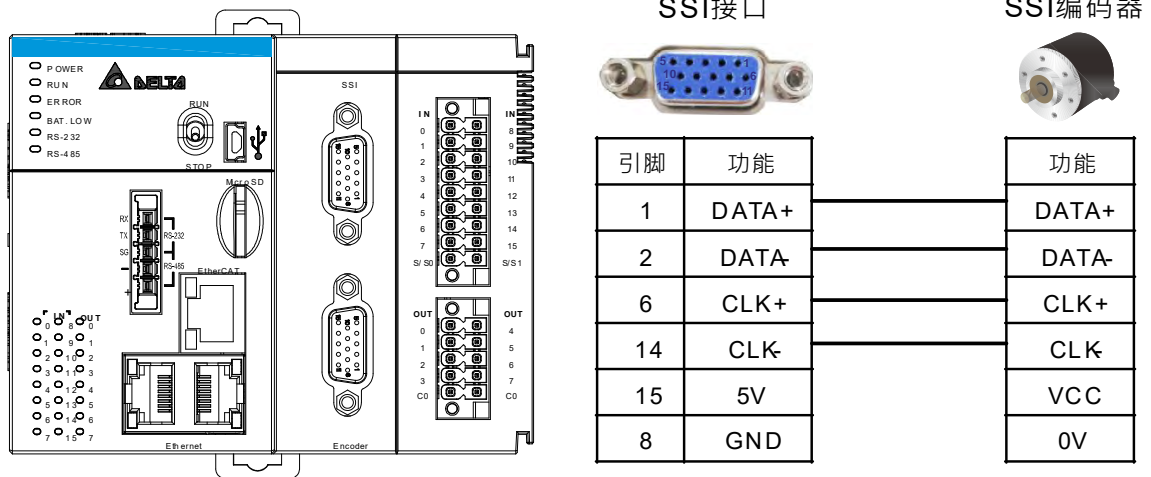
- 高速计数器配线，请参考 2.2.5.3 节。
- 一般输入/输出配线



2.2.5.3 AX-308EA0MA1T、AX-316EAMA1T、AX-364ELA0MA1T 配线

输入形式	IN0~IN15：直流（漏型 Sink 或源型 Source）
输入电压/电流	IN0~IN15：24VDC · 5mA
输出点类型	Out0~Out7：晶体管-T（漏型）
输出电压/电流	Out0~Out7：5~30VDC · 0.1A

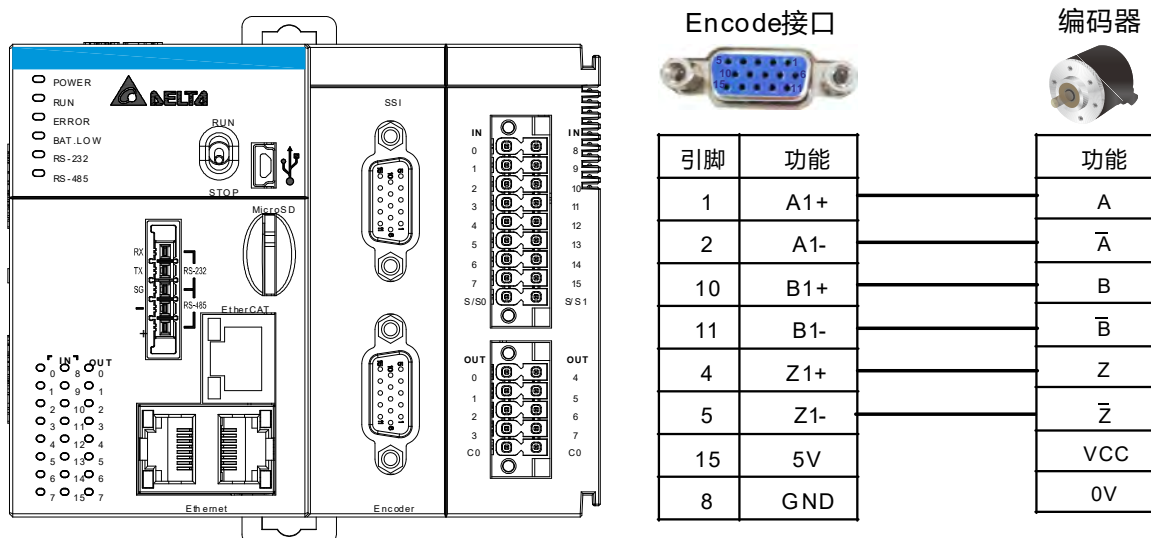
● SSI 配线



注：SSI 接口提供的是 5V 电源。

当 VCC = 5V 时，可以将 SSI 编码器电源电压 VCC 直接与 SSI 接口的引脚 15 连接，将 SSI 编码器的 0V 与 SSI 界面的引脚 8 连接；当 VCC ≠ 5V 时，请根据连接的 SSI 编码器实际电源电压对 SSI 编码器进行单独供电。

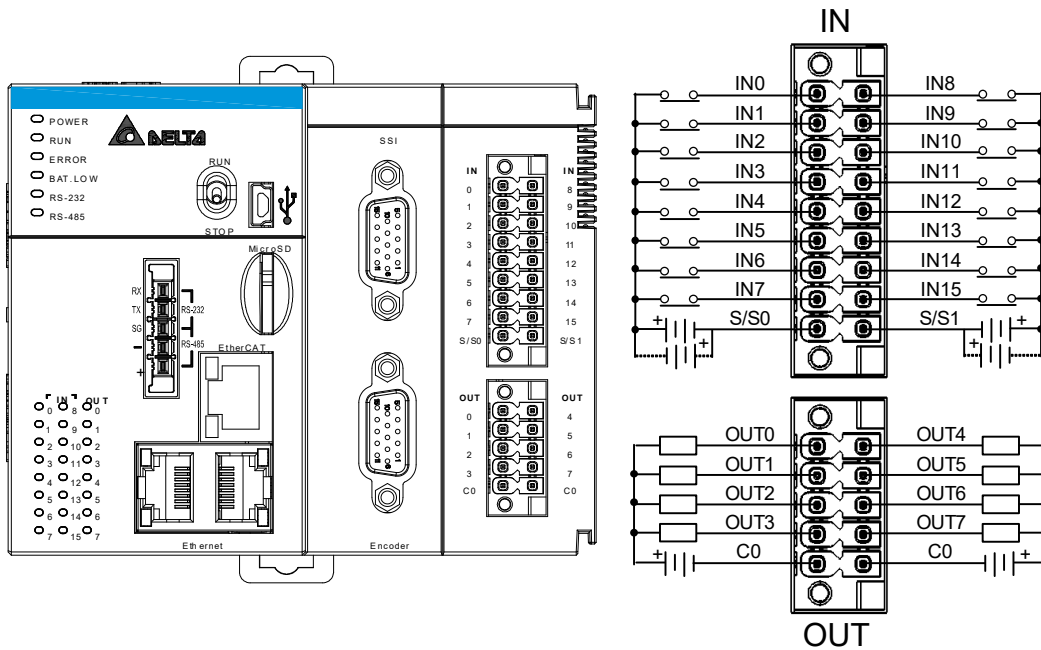
● Encoder 配线



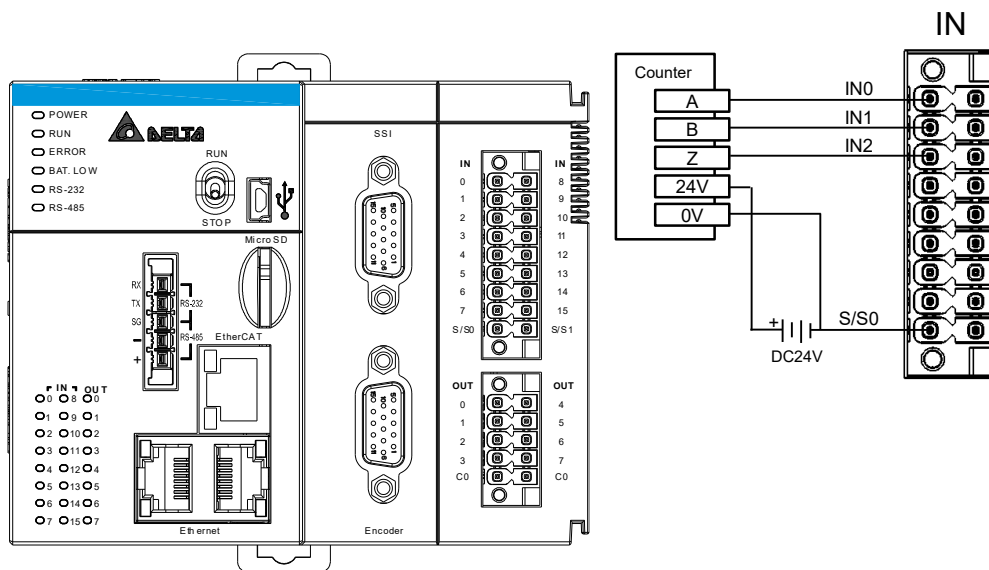
注：Encoder 接口提供的是 5V 电源。

当 VCC = 5V 时，可以将编码器电源电压 VCC 直接与 Encoder 接口的引脚 15 连接，将编码器的 0V 与 Encoder 界面的引脚 8 连接；当 VCC ≠ 5V 时，请根据连接的编码器实际电源电压对编码器进行单独供电。

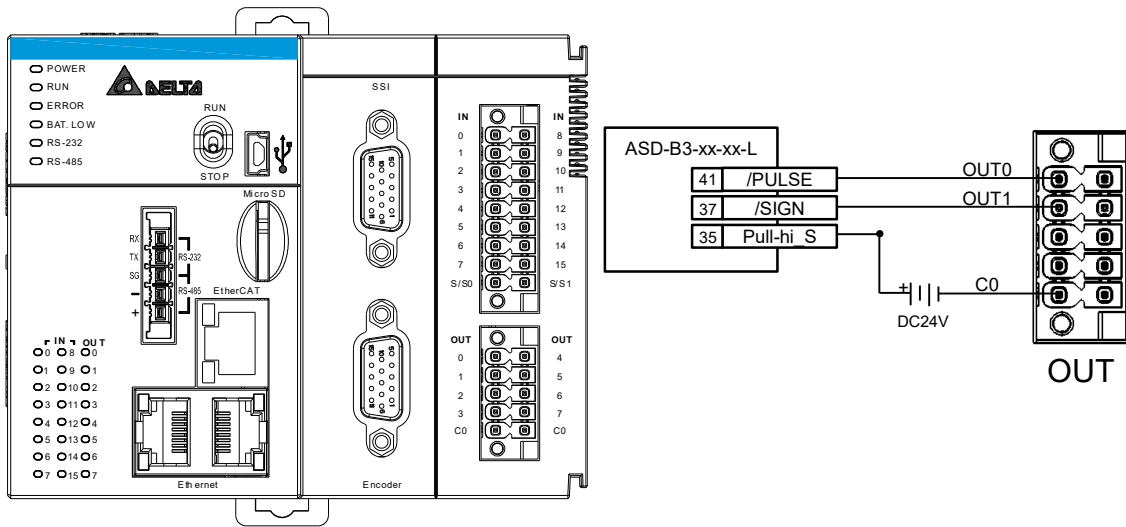
● 一般输入/输出配线



● 高速计数器配线 (NPN)



● 高速输出配线

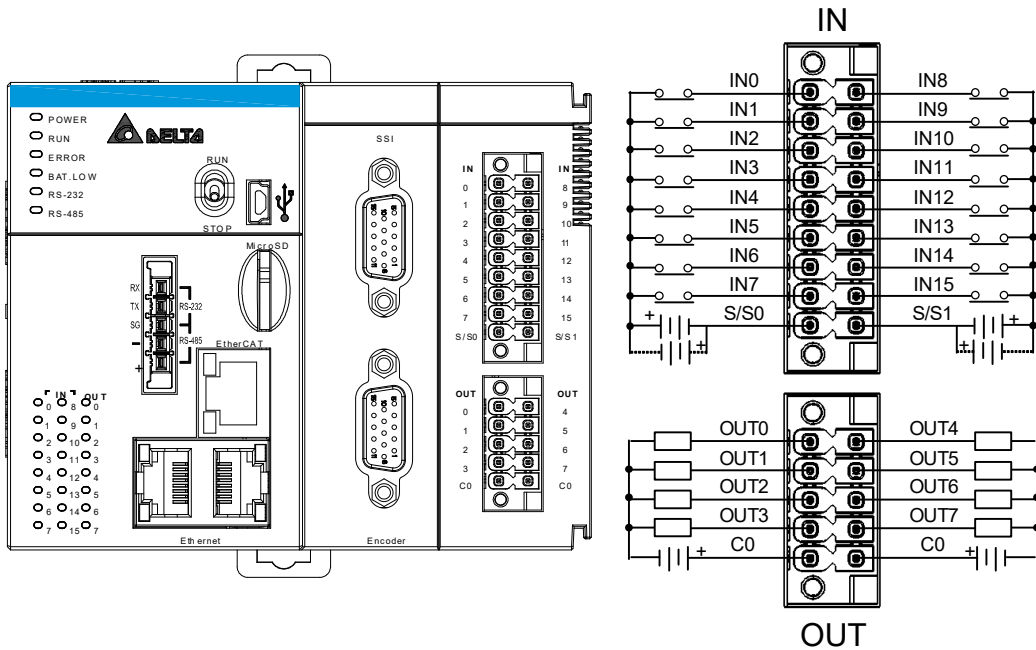


2

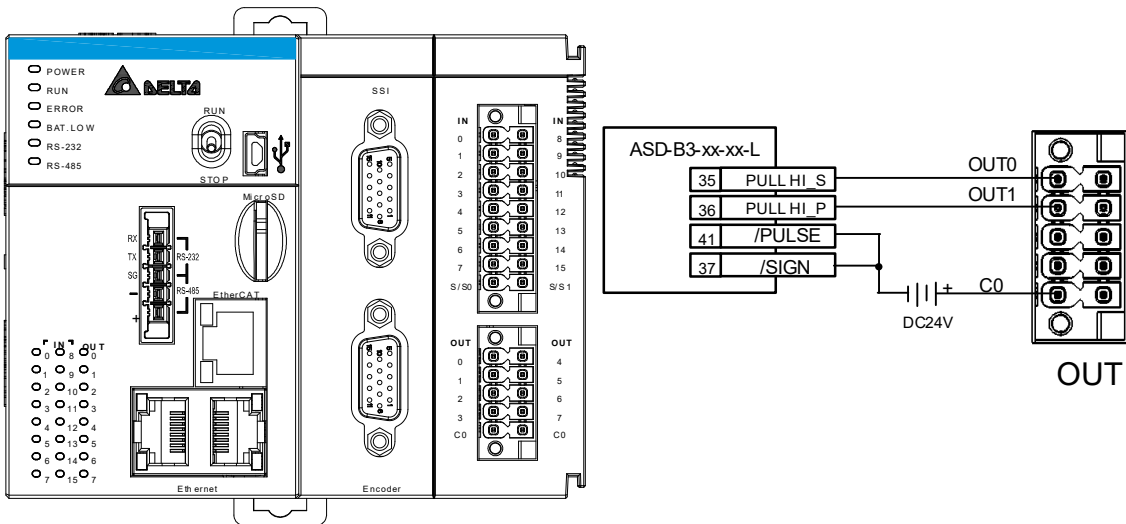
2.2.5.4 AX-308EA0M1P 配线

输入形式	IN0~IN15：直流（漏型 Sink 或源型 Source）
输入电压/电流	IN0~IN15：24VDC · 5mA
输出点类型	Out0~Out7：晶体管-P（源型）
输出电压/电流	Out0~Out7：5~30VDC · 0.1A

- SSI 配线、Encoder 配线、高速计数器配线，请参考第 2.2.5.3 节。
- 一般输入/输出配线



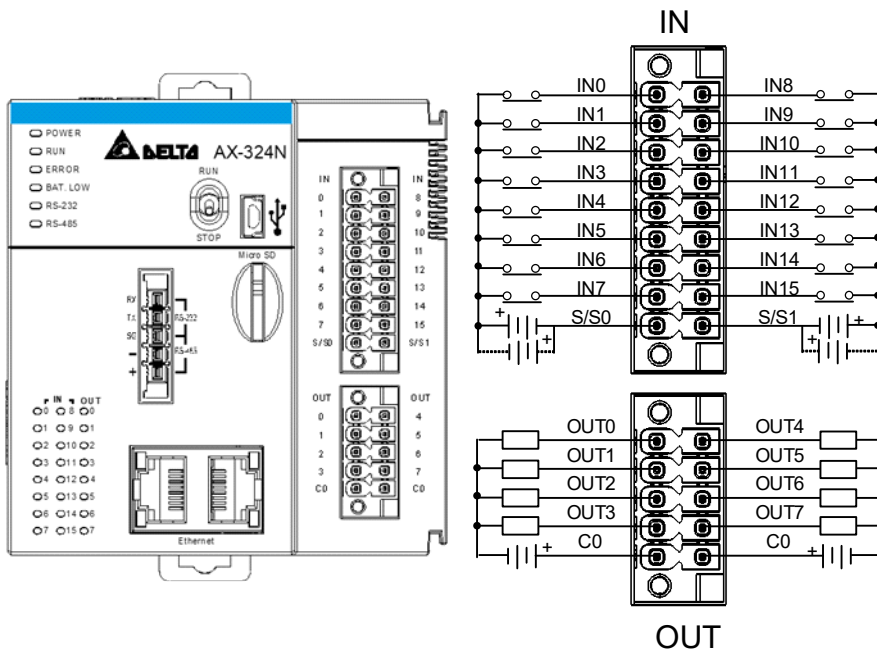
● 高速输出配线



2.2.5.5 AX-324NA0PA1P 配线

输入形式	直流 (漏型 Sink 或源型 Source)
输入电压/电流	24VDC · 5mA
输出点类型	晶体管-P (源型)
输出电压/电流	5~30VDC · 0.1A

● 一般输入/输出配线

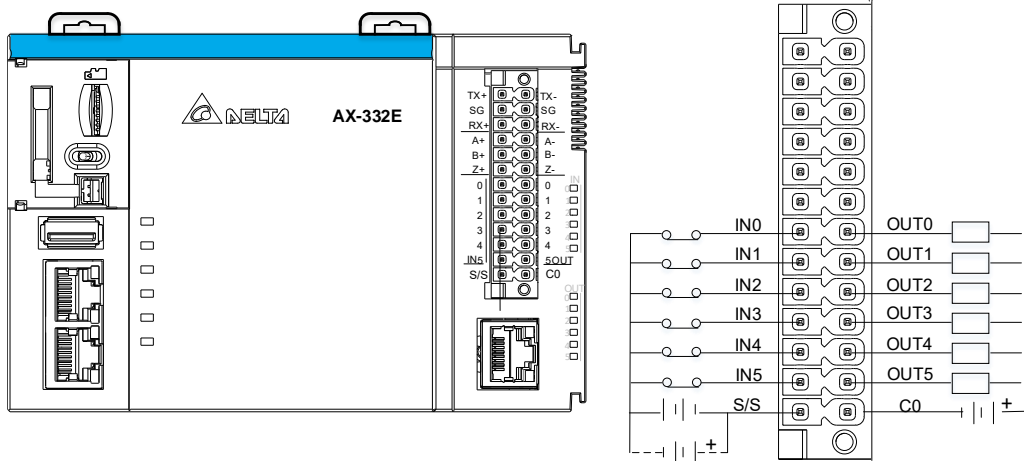


2.2.5.6 AX-332EP0MB1T 配线

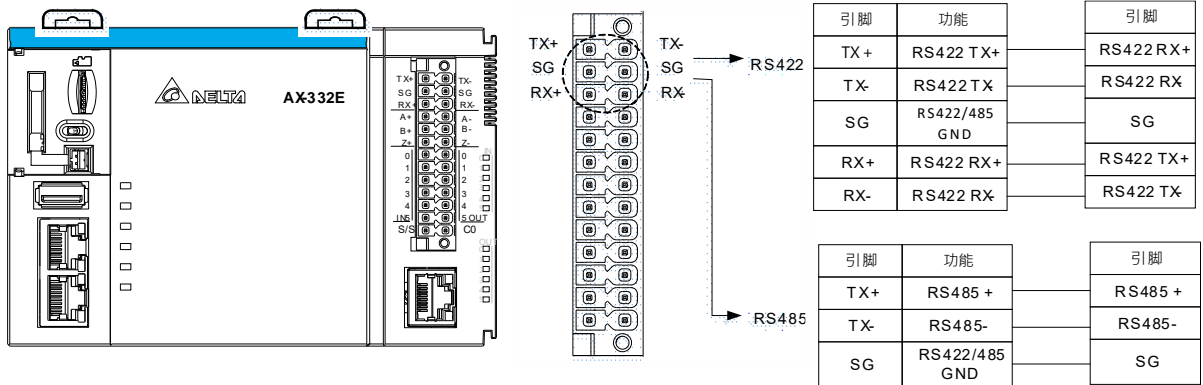
输入形式	直流 (漏型 Sink 或源型 Source)
输入电压/电流	24VDC · 5mA
输出点类型	晶体管-T (漏型)
输出电压/电流	5~30VDC · 0.1A

2

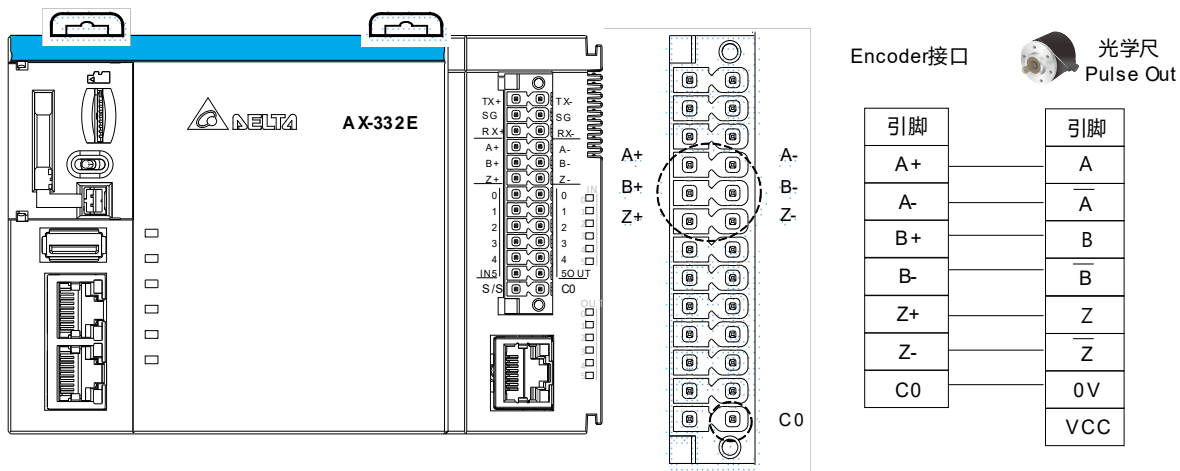
- 一般输入/输出配线



- RS-422 / RS-485 配线



● Encoder 配线



注：Encoder 界面仅支持 5V。

2.3 电源模块规格

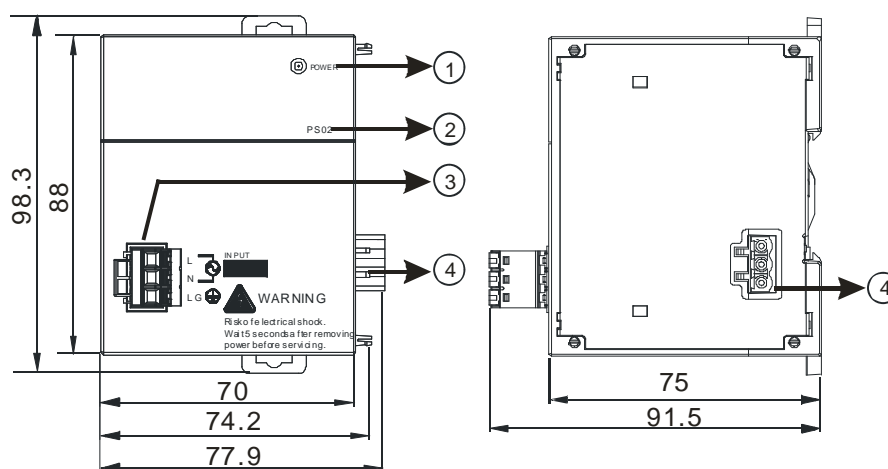
2.3.1 一般规格

- AS-PS02/AS-PS02A

项目	规格	
电源电压	100~240VAC (-15%~10%) · 50/60Hz□5%。	
动作规格	当输入电源大于 85VAC 时，电源模块可正常动作。	
允许瞬间断电时间	电源瞬间断电 10ms 以内继续运行。	
电源保险丝容量	2.5A/250VAC	
冲击电流	< 70A@115VAC	
24VDC 电流输出	AS-PS02 : 2A · 仅供应内部主机及模块使用 AS-PS02A : 1.5A · 仅供应内部主机及模块使用 ; 0.5A · 供外部使用	
电源保护	24VDC 输出具短路保护，与过电流保护。	
电气隔离	1,500VAC (Primary-secondary)、 1,500VAC (Primary-PE)、 500VAC (Secondary-PE)	
绝缘阻抗	5MΩ 以上 (所有输出/入点对地之间 500VDC)	
接地	接地配线之线径不得小于电源端 L · N 之线径	
重量	AS-PS02	270g
	AS-PS02A	310g

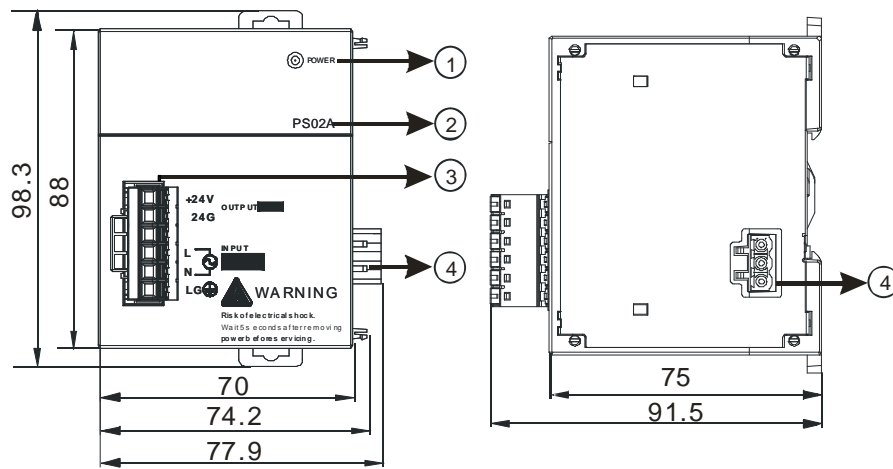
2.3.2 电源模块外观尺寸及部位介绍

- AS-PS02



单位：mm

● AS-PS02A

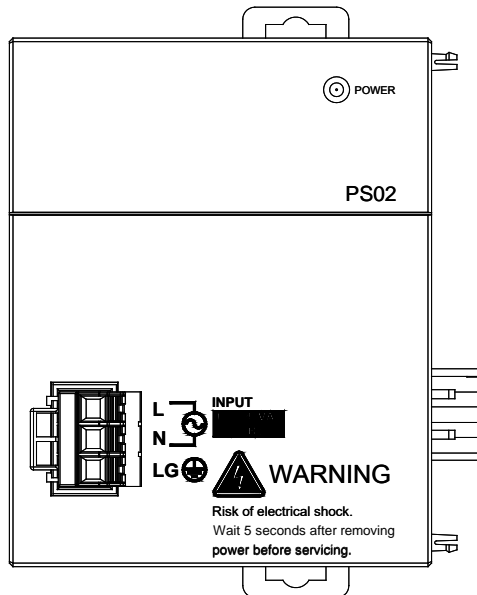


单位：mm

序号	名称	说明
1	电源指示灯（绿灯）	指示电源的状态
2	机种名称	模块机种名称
3	输出端子配置 (only for AS-PS02A)	+24V：直流电源输出：24VDC · 500mA 24G：直流电源输出参考地
	输入端子配置	L：AC 电源输入 Line N：AC 电源输入 Neutral LG：大地接地端
4	电源输出口 (与主机连接)	

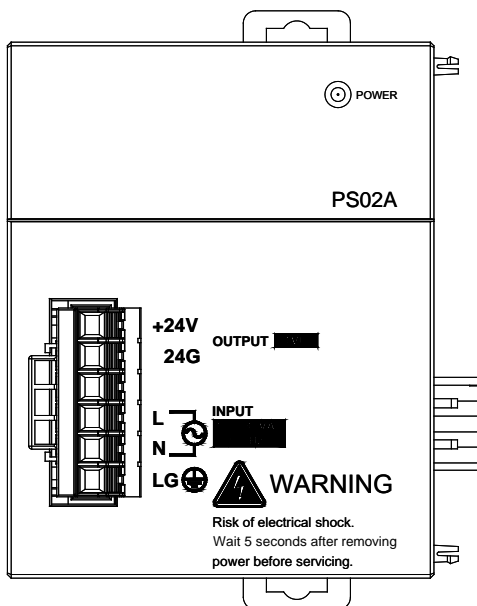
2.3.3 端子配置

- AS-PS02



- L/N : AC 电源输入
- LG : 大地接地端
-

- AS-PS02A



- +24V : 连接外部 24VDC 电源+
- 24G : 连接外部电源 24G
- L/N : AC 电源输入
- LG : 大地接地端

2.4 扩展模块

AX-3 系列的扩展模块可连接 AS 系列模块，相关的模块信息请参考 AS 系列模块手册。

MEMO

2

第3章 安装与启动

目录

3.1	硬件安装	3-2
3.1.1	SD 卡插入与取出方法	3-2
3.1.1.1	AX-3 系列 (除 AX-332 外) SD 卡插入与取出方法	3-2
3.1.1.2	AX-332 SD 卡插入与取出方法	3-3
3.1.2	电池的安装与拆除方法	3-4
3.1.2.1	电池的安装	3-4
3.1.2.1.1	AX-3 系列 (除 AX-332 外) 电池的安装	3-4
3.1.2.1.2	AX-332 电池的安装	3-6
3.1.2.2	电池的拆除	3-7
3.1.2.2.1	AX-3 系列 (除 AX-332 外) 电池拆除	3-7
3.1.2.2.2	AX-332 电池拆除	3-9
3.1.3	安装至控制柜	3-11
3.1.3.1	安装至 DIN 导轨	3-11
3.1.3.2	控制柜内的安装方法	3-11
3.1.3.3	控制柜内的环境温度要求	3-12
3.1.3.4	提高抗干扰性的措施	3-12
3.1.3.5	控制柜内的尺寸要求	3-12
3.2	软件安装与卸除	3-13
3.2.1	软件安装	3-13
3.2.2	软件卸除	3-20
3.3	软件激活与通讯	3-21
3.3.1	启动	3-21
3.3.2	通讯	3-22

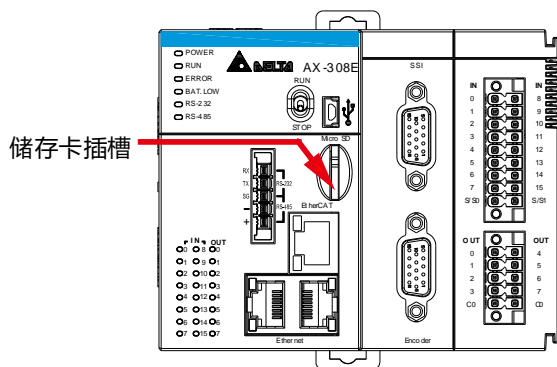
3.1 硬件安装

3.1.1 SD 卡插入与取出方法

3.1.1.1 AX-3 系列 (除 AX-332 外) SD 卡插入与取出方法

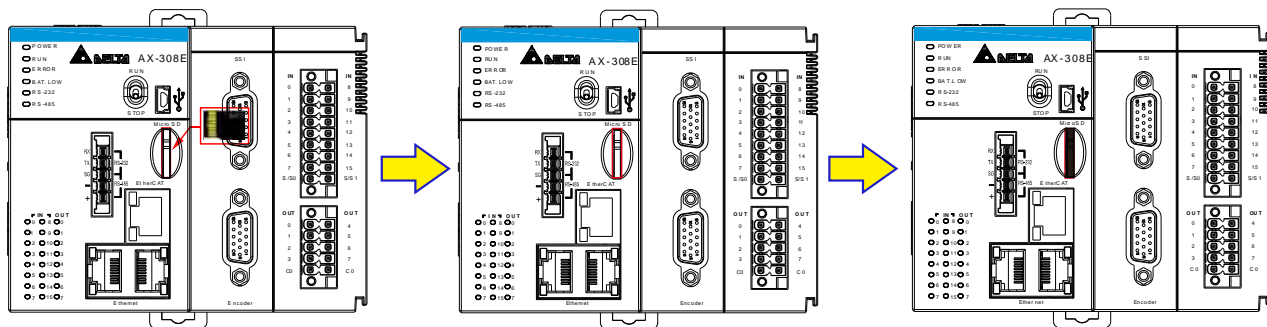
- 主机上的储存卡插槽

主机的储存卡插槽都被安排在机体正面，如下图所示所示。



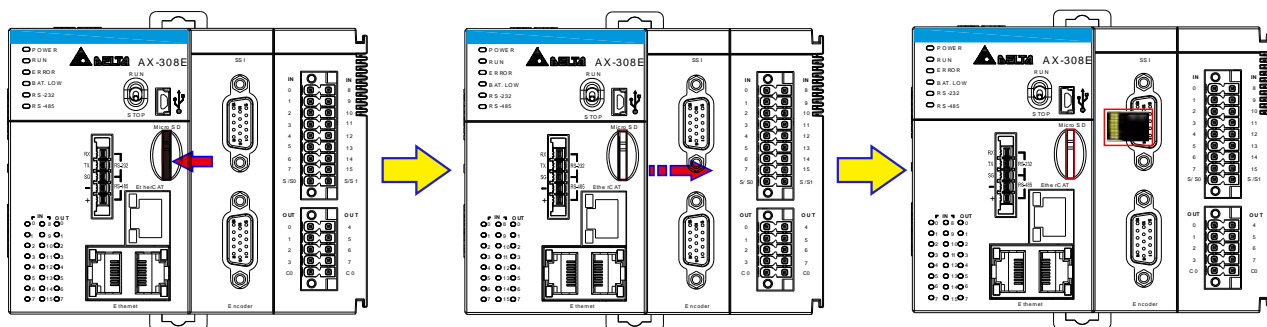
- 储存卡的插入

直接将储存卡垂直地插入主机的储存卡插槽并将其推至底部，直到听见卡榫固定的声音即可。顺利插入后，SD 卡应该会被牢牢的固定住，若仍是松脱的状态表示并未插入正确。另外，SD 卡本身有防呆设计，若 SD 卡插入的方向错误便会无法将其推至插槽底部，此时请勿强制推入以免造成机体的损坏。插入 SD 卡的正确方向请参考下方图示。



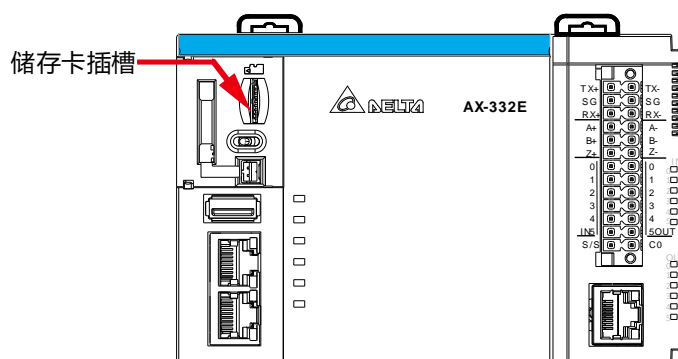
- 储存卡取出

直接将储存卡推至底部后，储存卡即会松脱弹出，此时便可将其取出。



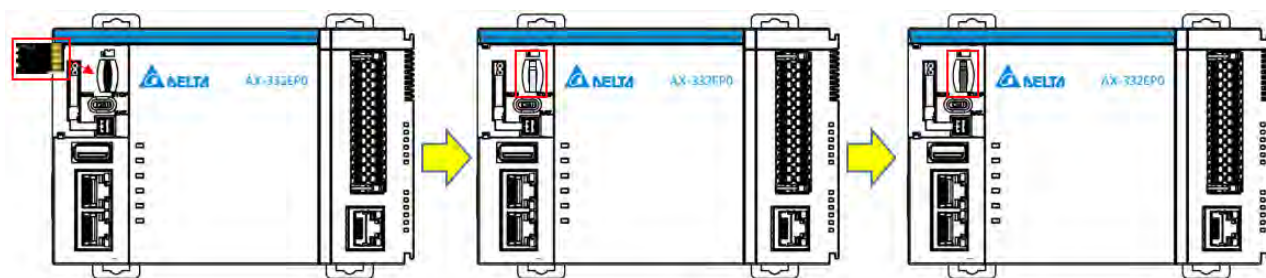
3.1.1.2 AX-332 SD 卡插入与取出方法

主机的储存卡插槽都被安排在机体正面，如下图箭头标示处所示。



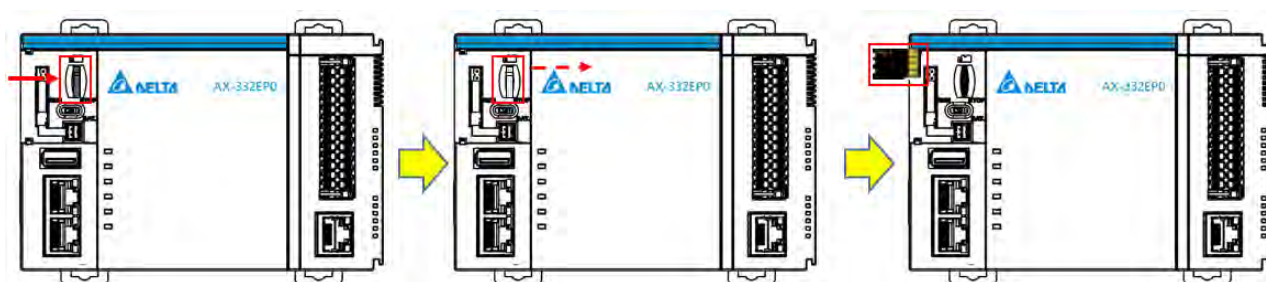
● 储存卡的插入

直接将储存卡垂直地插入主机的储存卡插槽并将其推至底部，直到听见卡榫固定的声音即可。顺利插入后，SD 卡应该会被牢牢的固定住，若仍是松脱的状态表示并未插入正确。另外，SD 卡本身有防呆设计，若 SD 卡插入的方向错误便会无法将其推至插槽底部，此时请勿强制推入以免造成机体的损坏。插入 SD 卡的正确方向请参考下方图示。



● 储存卡取出

直接将储存卡推至底部后，储存卡即会松脱弹出，此时便可将其取出。



3.1.2 电池的安装与拆除方法

3.1.2.1 电池的安装



电池正常供电时，控制器万年历才可以计时。控制器出厂时并不附带电池，用户需自行安装市售 CR1620 3V 电池。第一次安装电池时，控制器在通电或者断电时都可以。安装电池前，必须使用静电手套或者先通过接触接地金属等方式，将人体的静电释放后再进行安装或者更换电池。安装电池后，可通过软件设定万年历时间，安装电池操作步骤如下（请参考 AS500E 系列运动控制器操作手册，如下述步骤）。

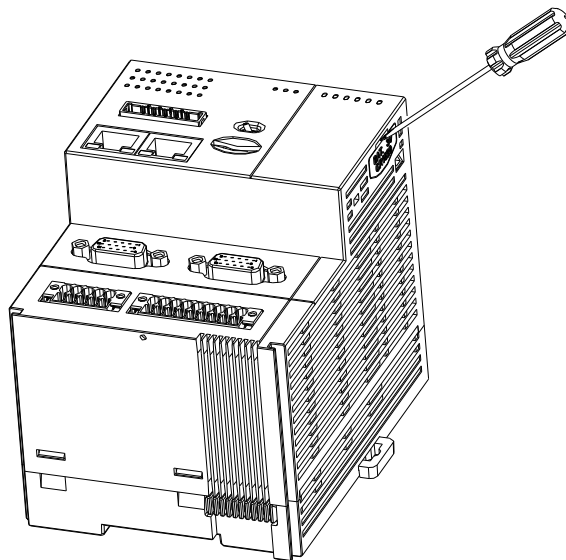
3

*AX-3系列（除AX-332外）：请购买市售CR1620 3V电池x1

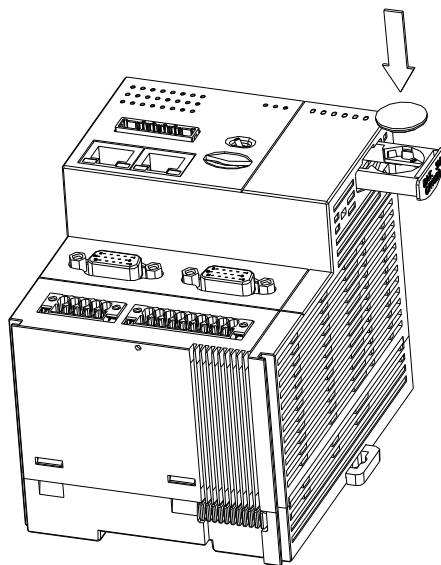
*AX-332电池配件请洽当地代理商

3.1.2.1.1 AX-3 系列（除 AX-332 外）电池的安装

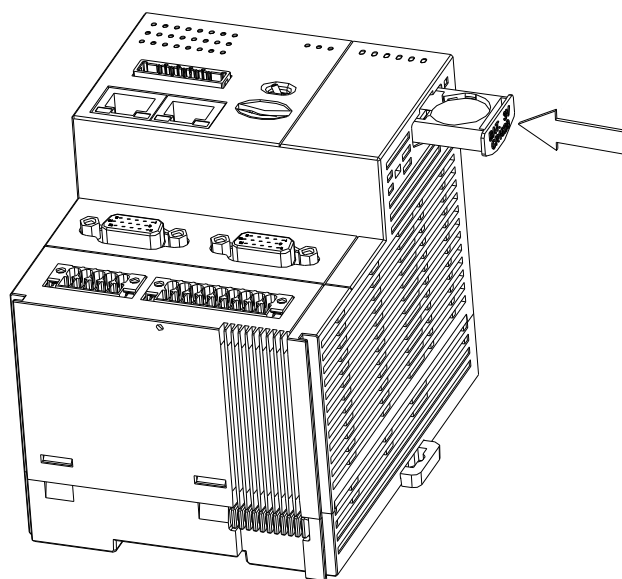
1. 将一字螺丝起子放在电池座缺口处，拨开运动控制器上抽屉式滑槽的电池座，如下图。



2. 将 CR1620 3V 电池由上而下放入到 AX-3 系列控制器的电池座内，如下图。

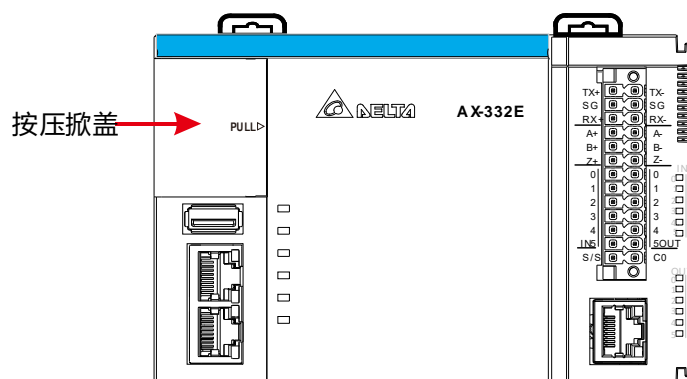


3. 电池放到电池座后，将抽屉式滑槽推进 AX-3 系列控制器中，如下图。

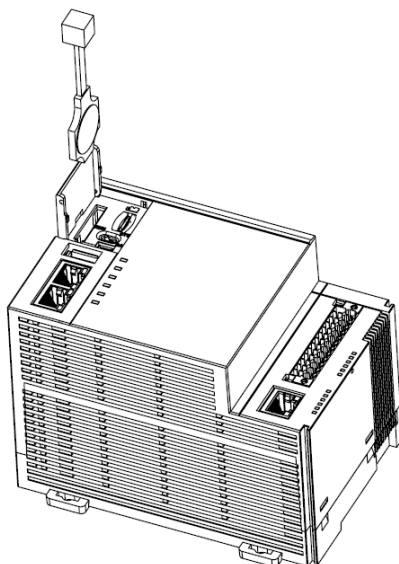


3.1.2.1.2 AX-332 电池的安装

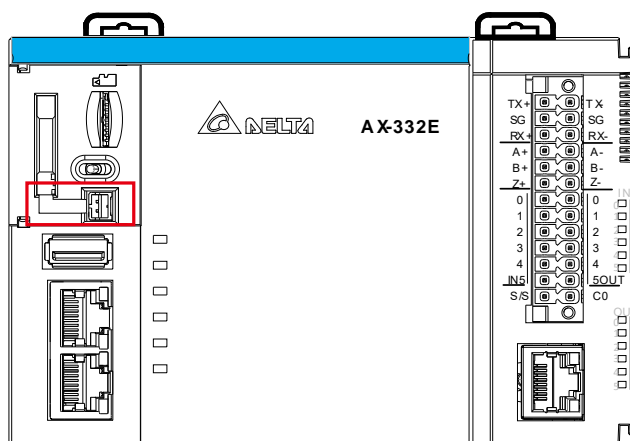
1. 手指轻压掀盖处，注意掀盖只能开启 90 度，如下图。



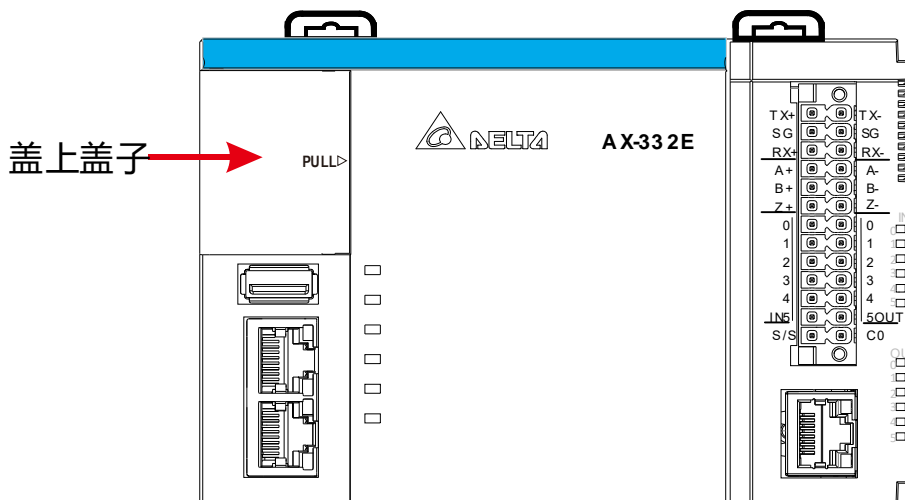
2. 将配件包电池由上而下放入到 AX-3 系列控制器的电池座内，如下图。



3. 电池放到电池座后，将电源线头接到电池插槽中，如下图。



4. 电源线整理好后，盖上掀盖，如下图。



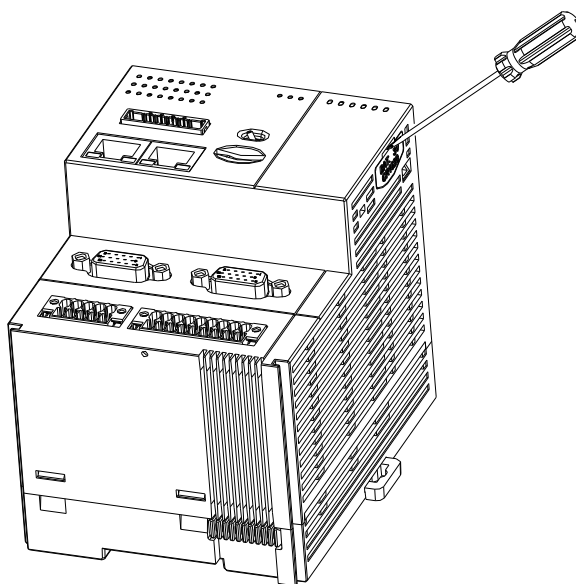
3.1.2.2 电池的拆除

警告

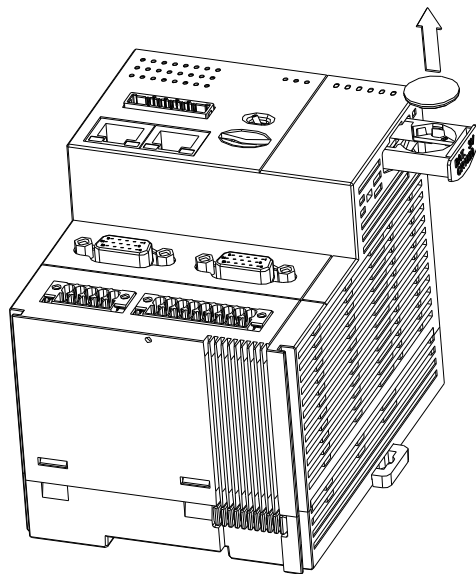
未安装电池或电池电压不足时，AX-3 系列控制器的 BAT LOW 指示灯会显示红灯常亮，需要安装电池或者更换电池。建议于控制器通电时更换电池，控制器断电更换电池将导致万年历时间恢复出厂值。更换电池前，必须使用静电手套或者先通过接触接地的金属等方式，将人体的静电释放后再进行安装或者更换电池。更换电池操作步骤如下（请参考 AS500E 系列运动控制器操作手册，如下述步骤）。

3.1.2.2.1 AX-3 系列（除 AX-332 外）电池拆除

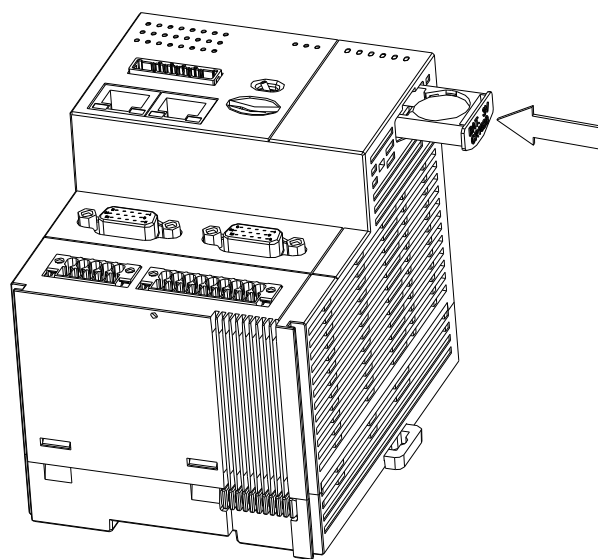
1. 将一字螺丝起子放在电池座缺口处，拔开控制器上抽屉式滑槽电池座，如下图。



2. 将 CR1620 3V 电池由下而上从控制器电池座内取出，然后再将新的电池放入，如下图所示。

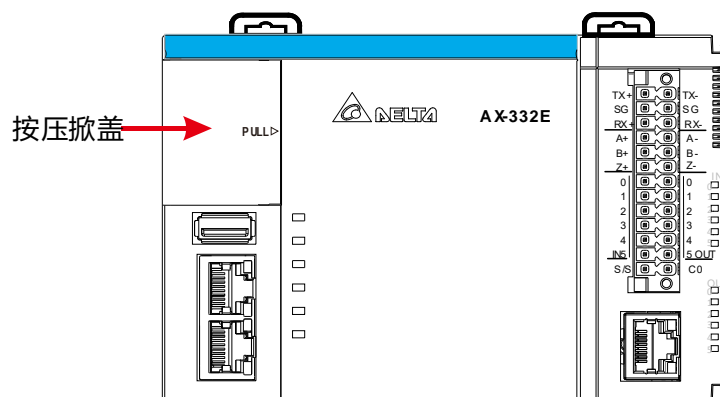


3. 放入新电池后，将抽屉式滑槽推进控制器中，如下图所示。

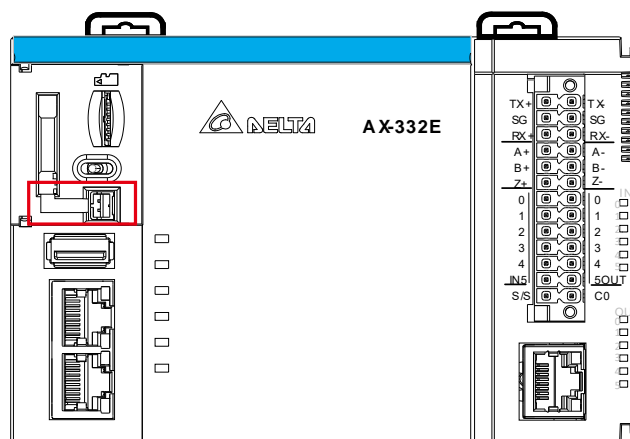


3.1.2.2.2 AX-332 电池拆除

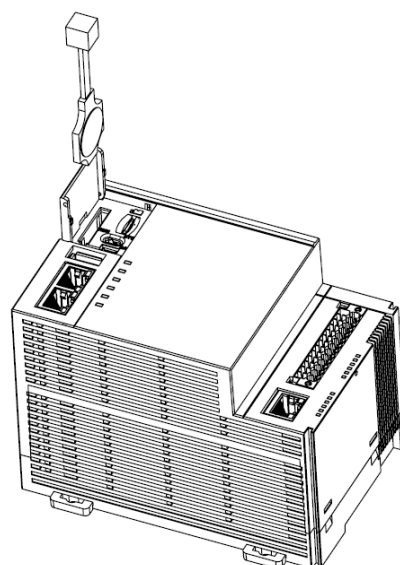
1. 手指轻压掀盖处，注意掀盖只能开启 90 度，如下图。



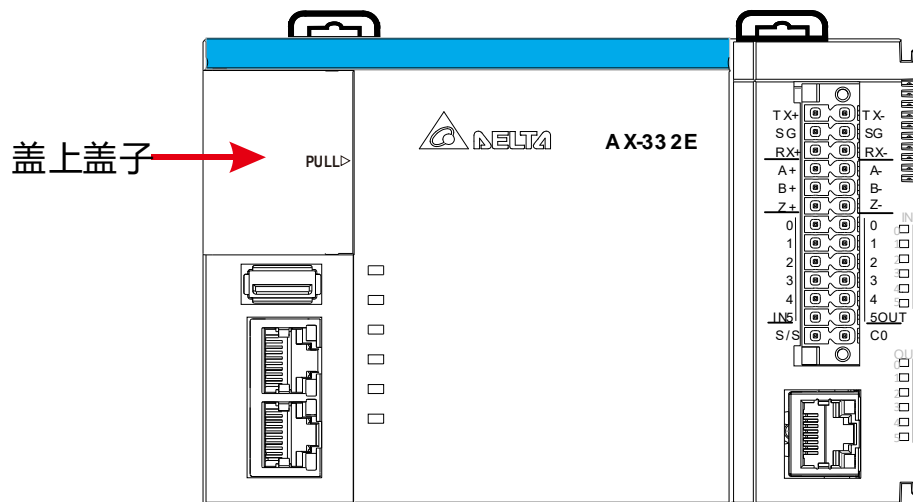
2. 电源线头从电池插槽中拔除，如下图。



3. 电池由下至上拔除，请注意小心勿用力拉扯以避免电源线断裂，如下图。



4. 电源线整理好后，盖上掀盖，如下图。

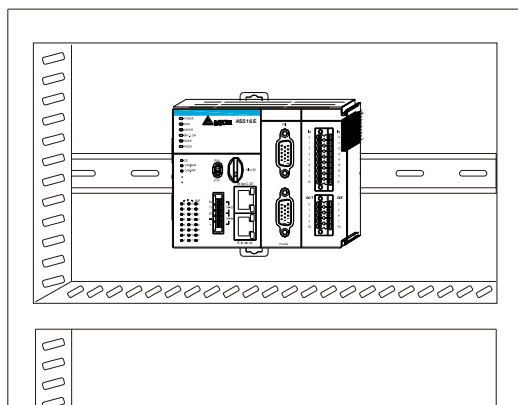


3

3.1.3 安装至控制柜

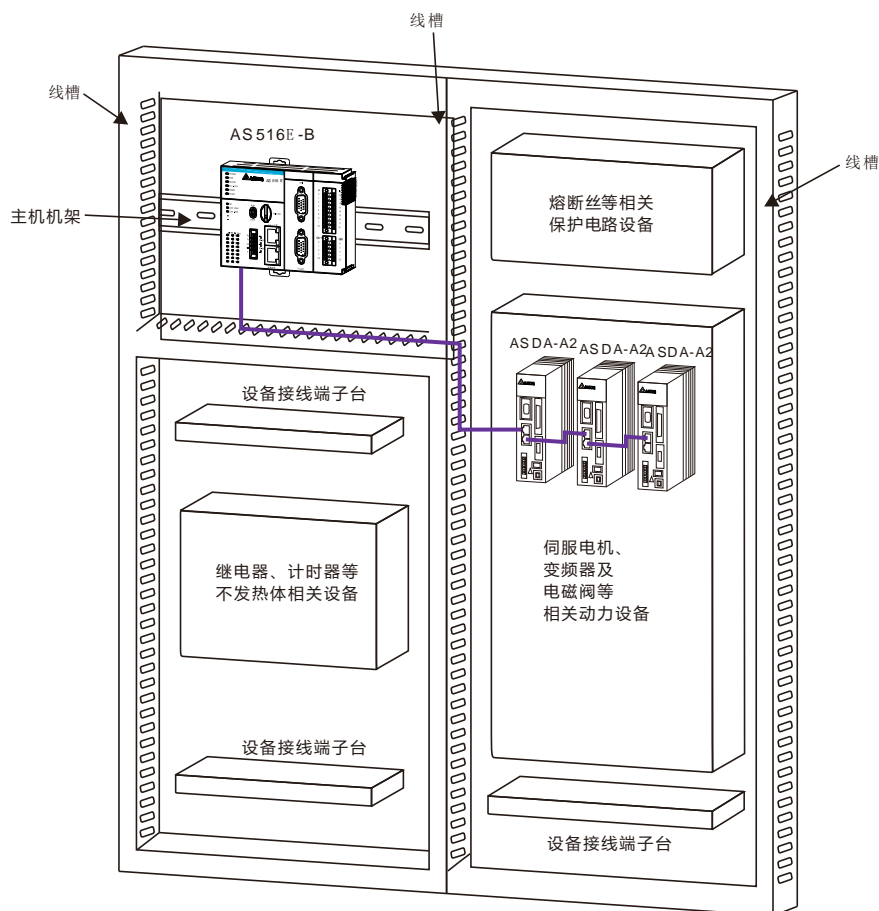
3.1.3.1 安装至 DIN 导轨

将控制器模块底部固定扣拉出，将模块底部横槽卡入控制柜中间导轨上，再推上固定扣，则控制器模块即固定于控制柜内。(请参考 AS500E 系列运动控制器操作手册，如下图。)



3.1.3.2 控制柜内的安装方法

(请参考 AS500E 系列运动控制器操作手册，如下图。)



3.1.3.3 控制柜内的环境温度要求

- 控制柜内安装 AX-3 系列控制器的要求：



- AX-3 系列控制器在控制柜里的环境温度为-20~55°C (温度) · 5 ~ 95% (湿度)。
- 请避免安装在高燃烧或温度易高的设备附近。
- 保留足够的通风空间。
- 超过环境温度 55°C · 必须安装风扇或空调。
- AX-3 系列控制器仅能于室内使用。
- 在控制柜安装时 · 1.0m~2.0m 的高度易于安装和操作。
- 安装时请远离高压设备和动力设备。
- 控制柜里电路状态要在断电情况下 · 才能安装 · 不可带电安装。

3.1.3.4 提高抗干扰性的措施

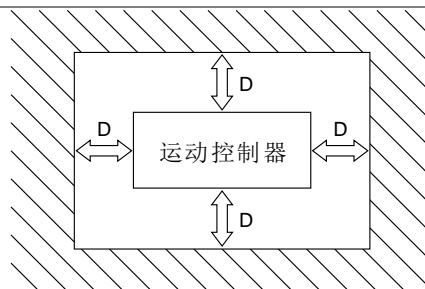


- 请避免安装在装有高压设备的柜里。
- 安装时请距动力线 200mm 以上。
- 控制柜须接地。
- 如果 AX-3 系列控制器未依照制造商指定的方式使用 · AX-3 系列控制器所提供的保护可能减弱。

3.1.3.5 控制柜内的尺寸要求

- 盘内安装

AX-3 系列控制器在安装时 · 请安装于封闭式控制箱内 · 其周围应保持一定的空间 · 以确保其散热功能正常 · (如右图所示)
 $D > 50\text{mm}$



在着手进行 AX-3 系统的开发工作前 · 必须先具备 DIADesigner-AX 的软件 · 其为整个系统的程序开发 · 硬件与网络组态的整合平台。

3.2 软件安装与卸除

● 系统需求

下表中标示了 DIADesigner-AX 的系统环境。请参考表格内信息进行软件安装。

项目	系统需求
运行系统	DIADesigner-AX V1.0.0 以上
操作系统	Windows 7 / 8.1 / 10 32/64 bits
CPU	Intel Celeron 540 1.8GHz 以上机种 · Intel Core i5 M520 2.4 GHz 以上机种
内存	2GB 以上 (建议使用 4GB 以上)
磁盘驱动器	容量至少须提供 10GB 以上空间
显示器	分辨率：1920 x 1080 建议显示设置
键盘/鼠标	一般的键盘鼠标或与 Windows 兼容的装置
通讯接口	Ethernet · USB · Serial port (依据主机本身所提供的通讯界面)
软件	须安装 Net Framework 4.6.2

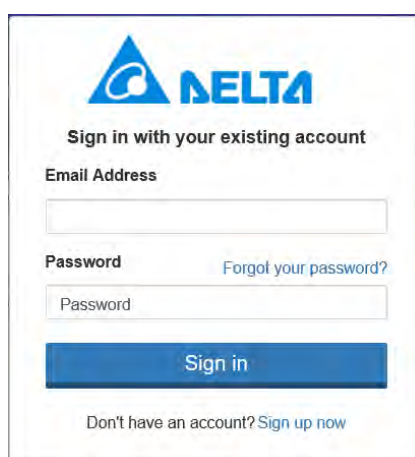
3.2.1 软件安装

在安装前先确定，要安装 DIADesigner-AX 的计算机是否符合系统要求中所提到的最低要求，且计算机内已运作 DIAInstaller。

DIAInstaller 是一个常驻程序，其用来管理所有台达 IA 软件。用户可以在 DIAInstaller 上简易检查下载/安装/更新状态。

如果尚未安装 DIAInstaller，请先至以下网址下载 DIAStudio：
<https://diastudio.deltaww.com/home/downloads>

进入台达官方网站下载前，若是第一次登入请先申请账号。



The image shows a login form for Delta DIAStudio. At the top is the Delta logo. Below it is the text "Sign in with your existing account". There are two input fields: "Email Address" and "Password". To the right of the password field is a link "Forgot your password?". Below the input fields is a blue "Sign in" button. At the bottom, there is a link "Don't have an account? Sign up now".

登入后即可下载 DIASudio 并进行安装

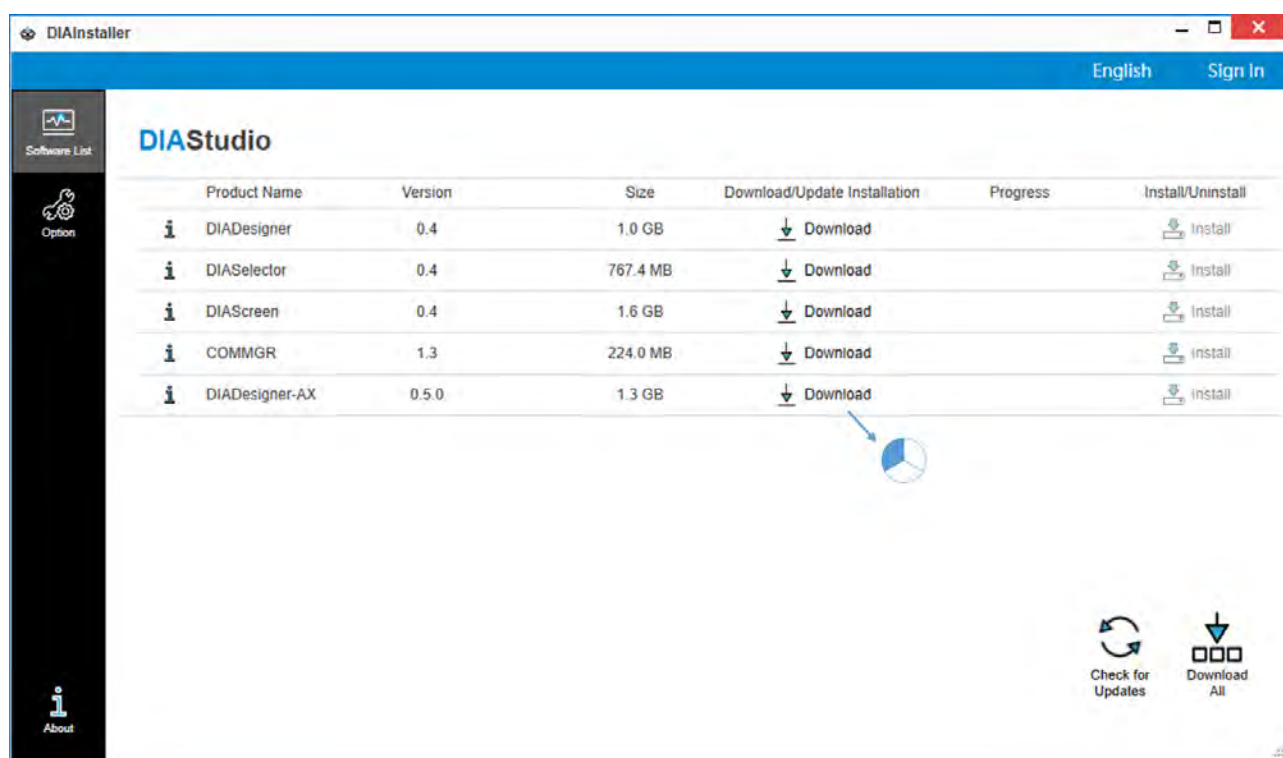
Software

Software Name	Description	OS	Issue Date	File
DIASelector App V0.4 (Early Access!)	DIASelector Mobile App	Android Lollipop (5.0) and above	2020/05/06	 
DIASudio V0.4 (Early Access!)	DIASudio Software download and Installation Tool	Windows 7 / 8.1 / 10 / Server 2012 R2 32/64 bit	2020/05/06	 

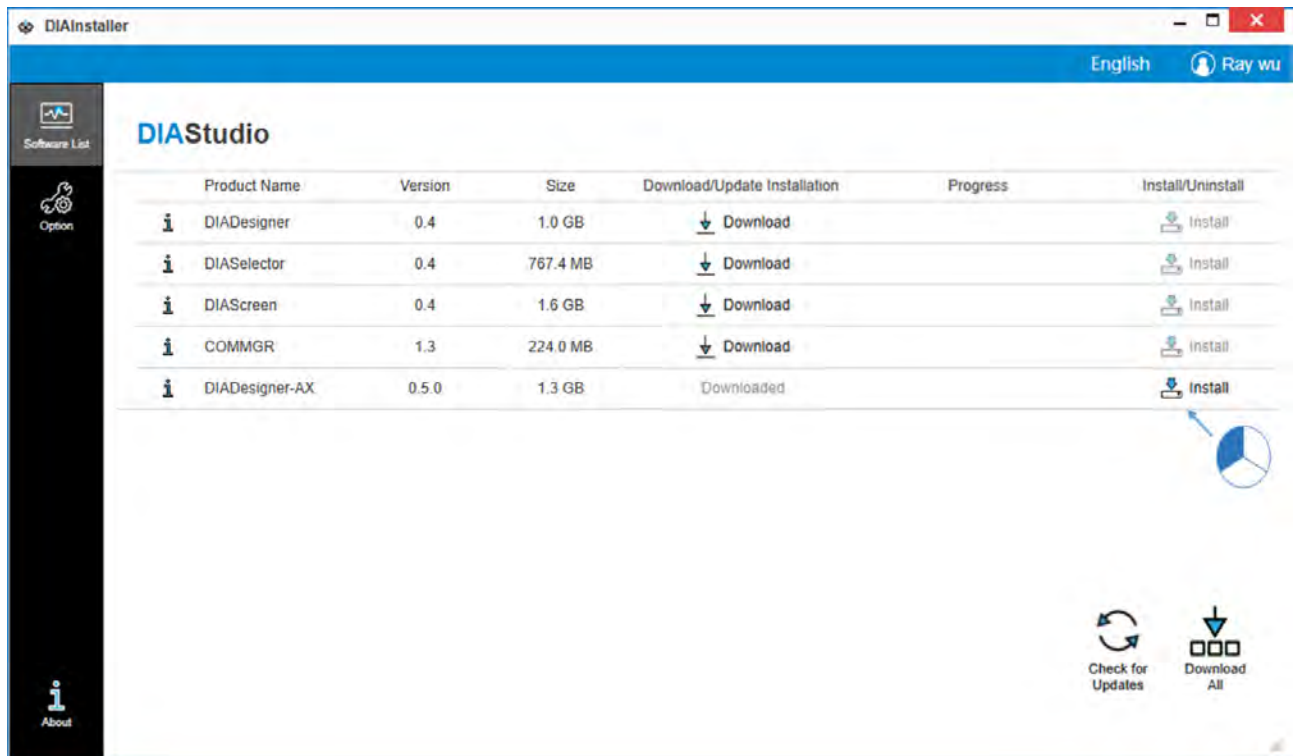
若是上述已完成，即可按照以下步骤开始进行安装 DIADesigner-AX：

1. 打开 DIASudio，检查 DIADesigner-AX 最新版本。
2. 鼠标单击「Download」以下载 DIADesigner-AX 安装档，如下图：

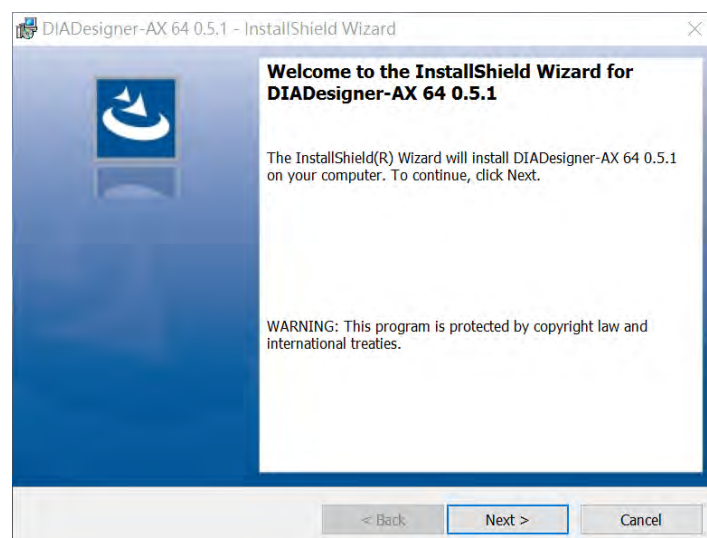
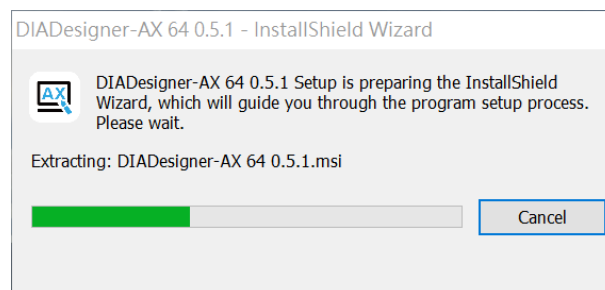
3



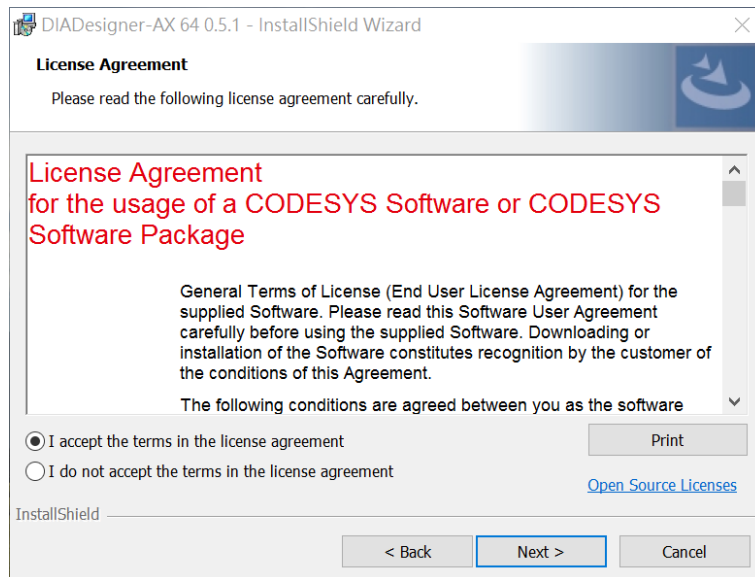
3. 下载完成后，单击「Install」即开始安装软件。



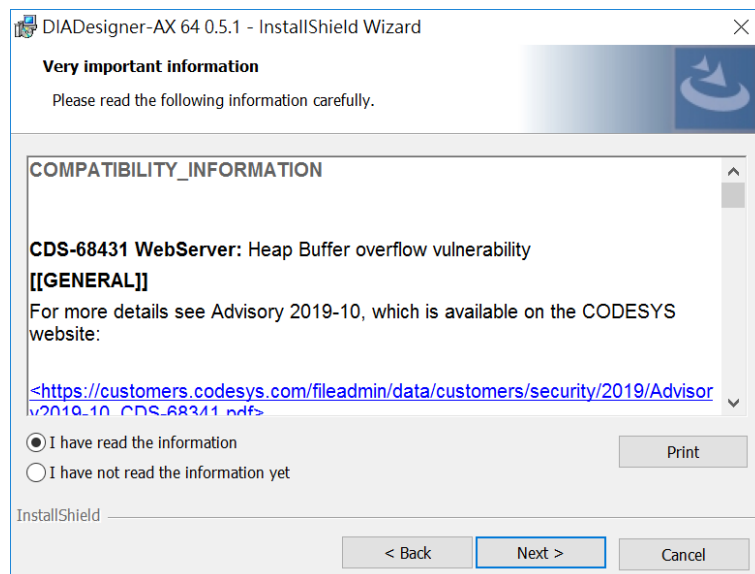
结果：系统将开始安装文件并显示 InstallShield Wizard 窗口，如下图所示。



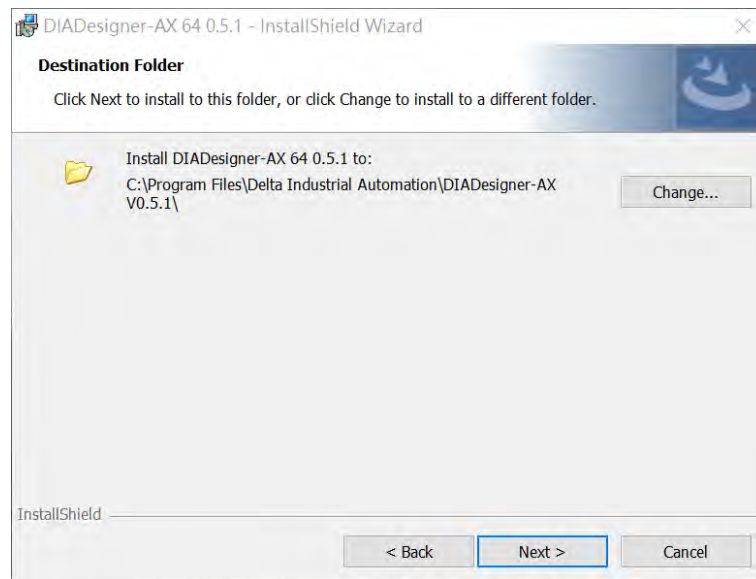
- 单击「Next」后显示 License Agreement 选项窗口，如下图所示。选择「I accept the terms in the license agreement」并单击「Next」。



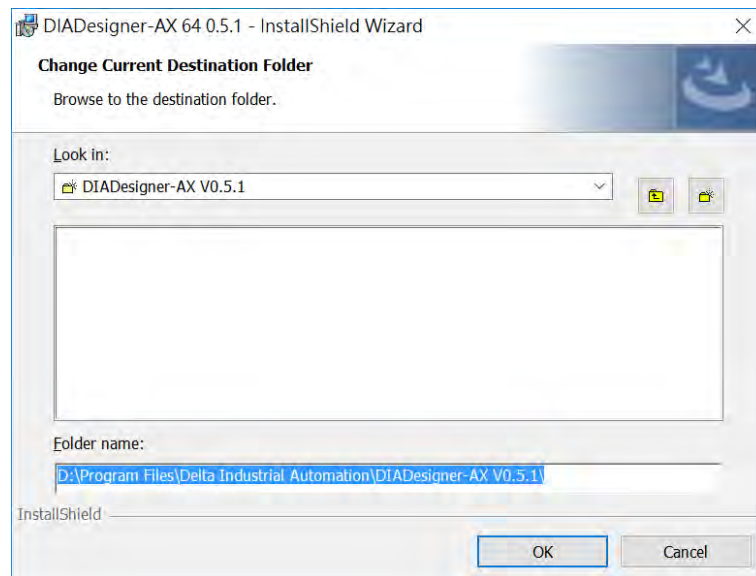
- 之后显示 Very important information 选项窗口，如下图所示。选择「I have read the information」并单击「Next」。



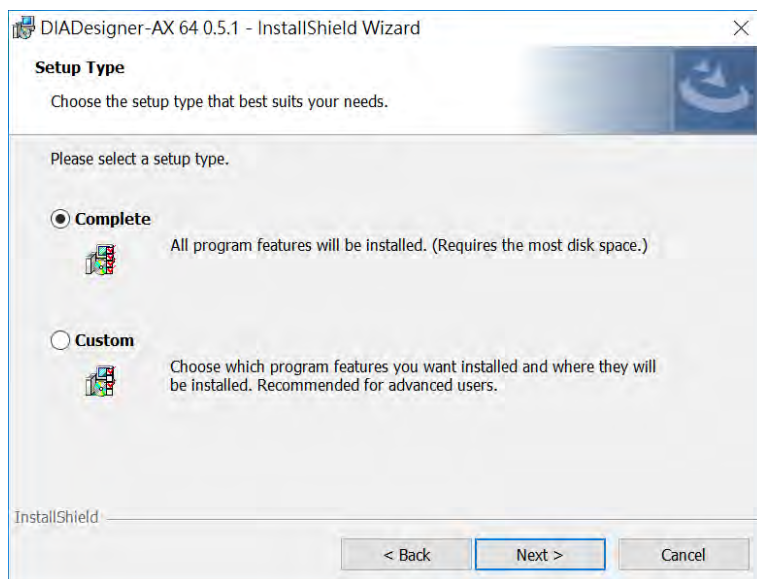
6. 之后显示 Destination Folder 选项窗口，如下图所示。如不变更默认路径，再单击「Next」。



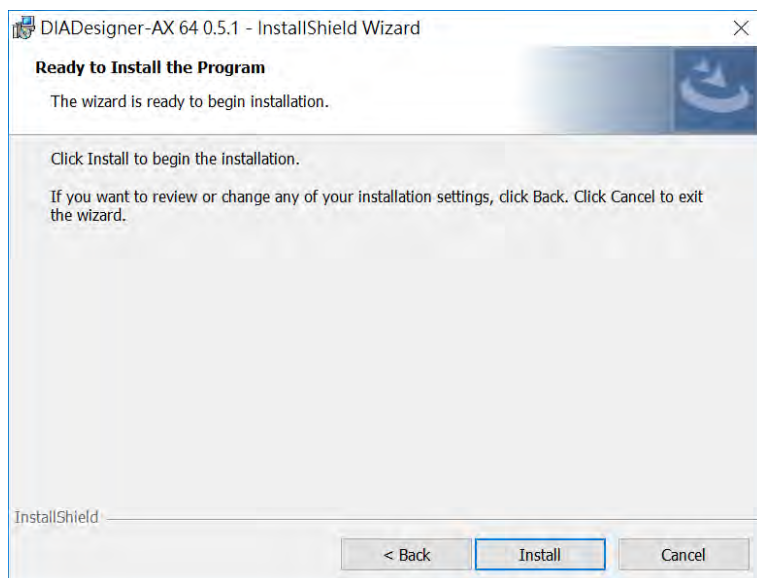
也可以单击「Change」后选择变更安装目标文件夹路径。



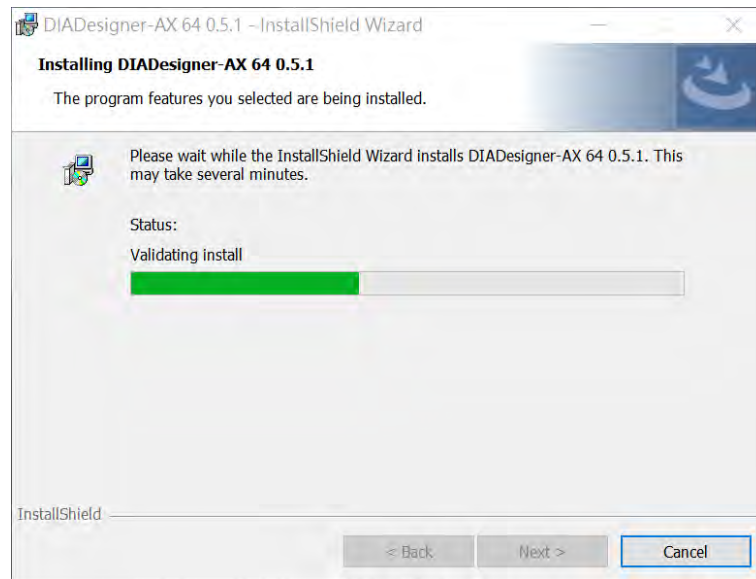
7. 之后显示 Setup Type 选项窗口，如下图所示。选择安装类型后，再单击「Next」。



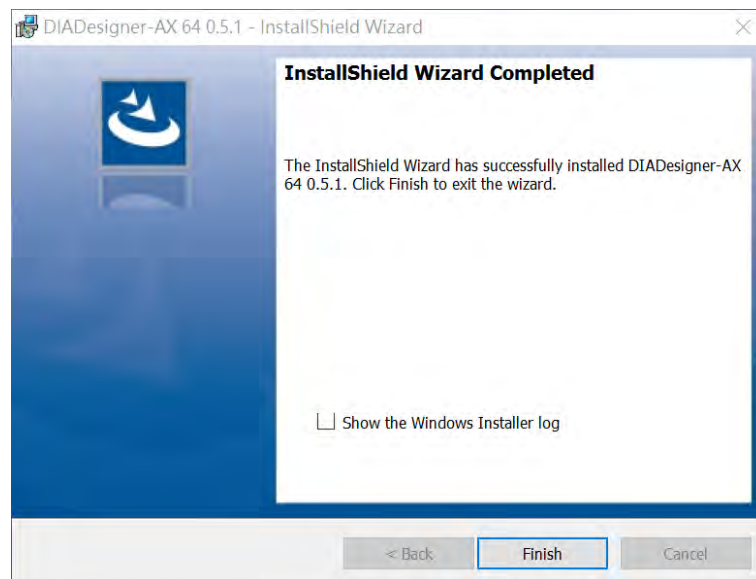
8. 之后显示 Ready to Install the Program 选项窗口，如下图所示。确定后单击「Install」后开始进行。



进行安装中，此处会须要等待一些时间完成安装。



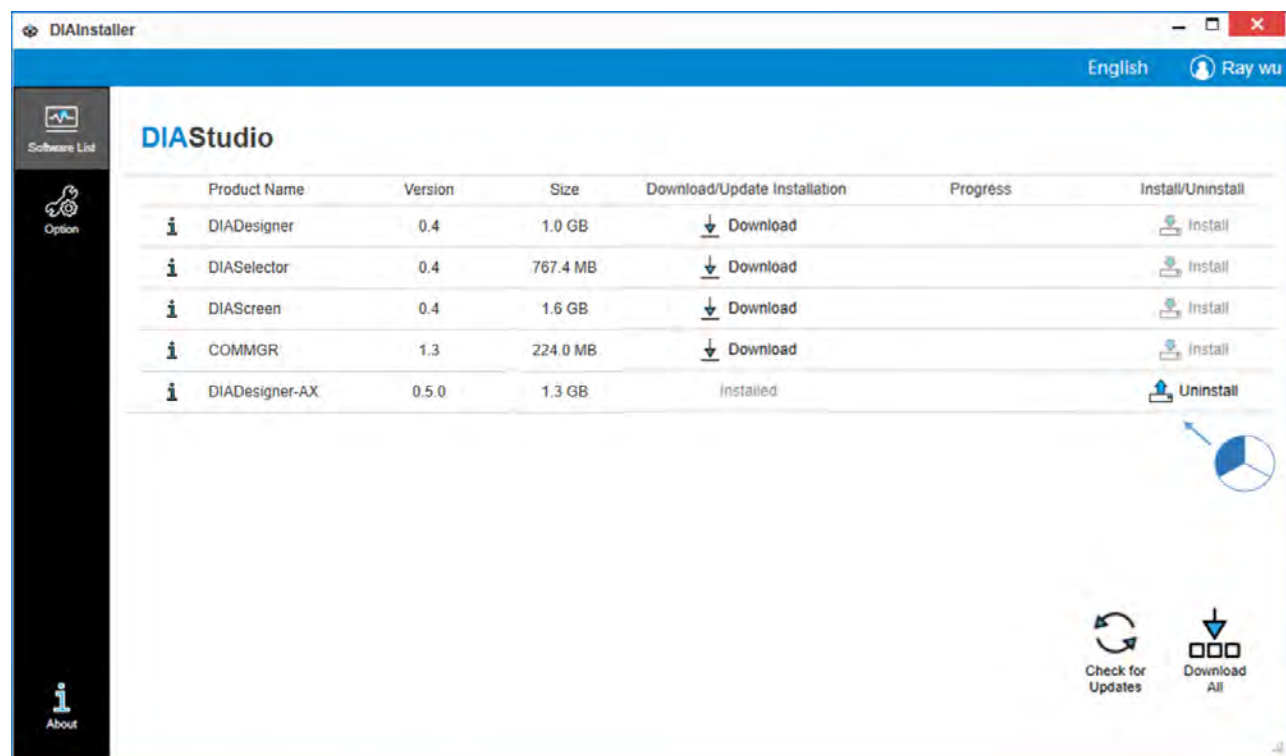
9. 安装完成后，将会显示 InstallShield Wizard Completed 窗口，如下图所示。单击「Finish」按钮即完成安装。



3.2.2 软件卸载

请按照以下步骤移除 DIADesigner-AX：

1. 打开 DIAInstaller，在目标软件上单击「Uninstall」卸载。

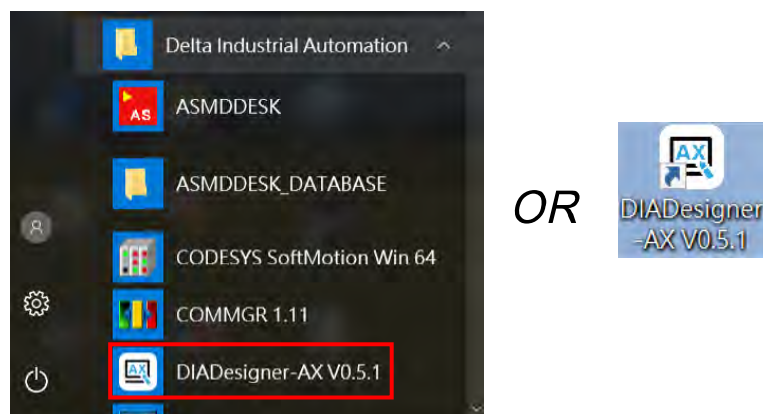


2. 卸载流程会在软件后台执行。
结果：软件已经被移除了。

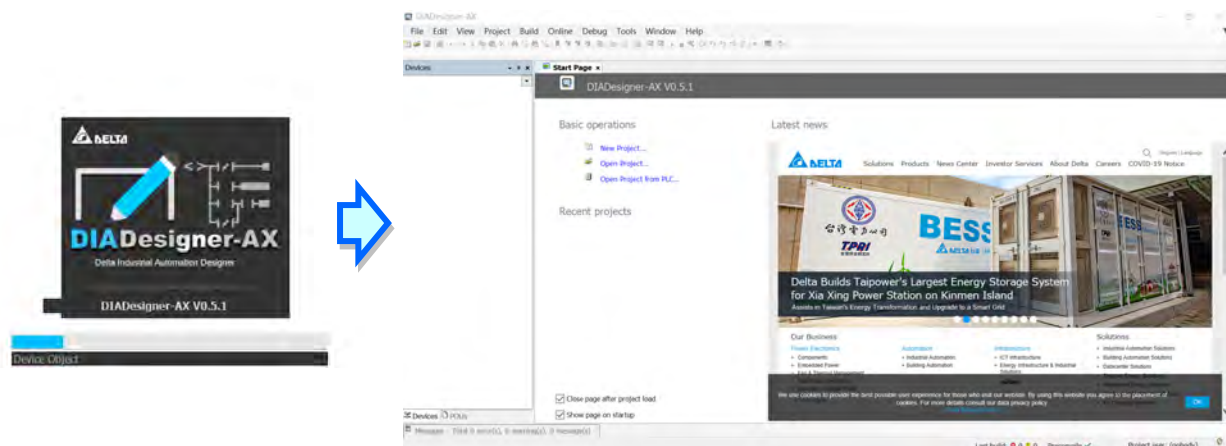
3.3 软件激活与通讯

3.3.1 启动

成功安装 DIADesigner-AX 之后，系统便会自动在桌面及程序集中建立快捷方式，单击该快捷方式即可启动 DIADesigner-AX。此外，DIADesigner-AX 亦支持同时开启多个软件窗口，只要依据相同步骤，另外再启动 DIADesigner-AX 即可。

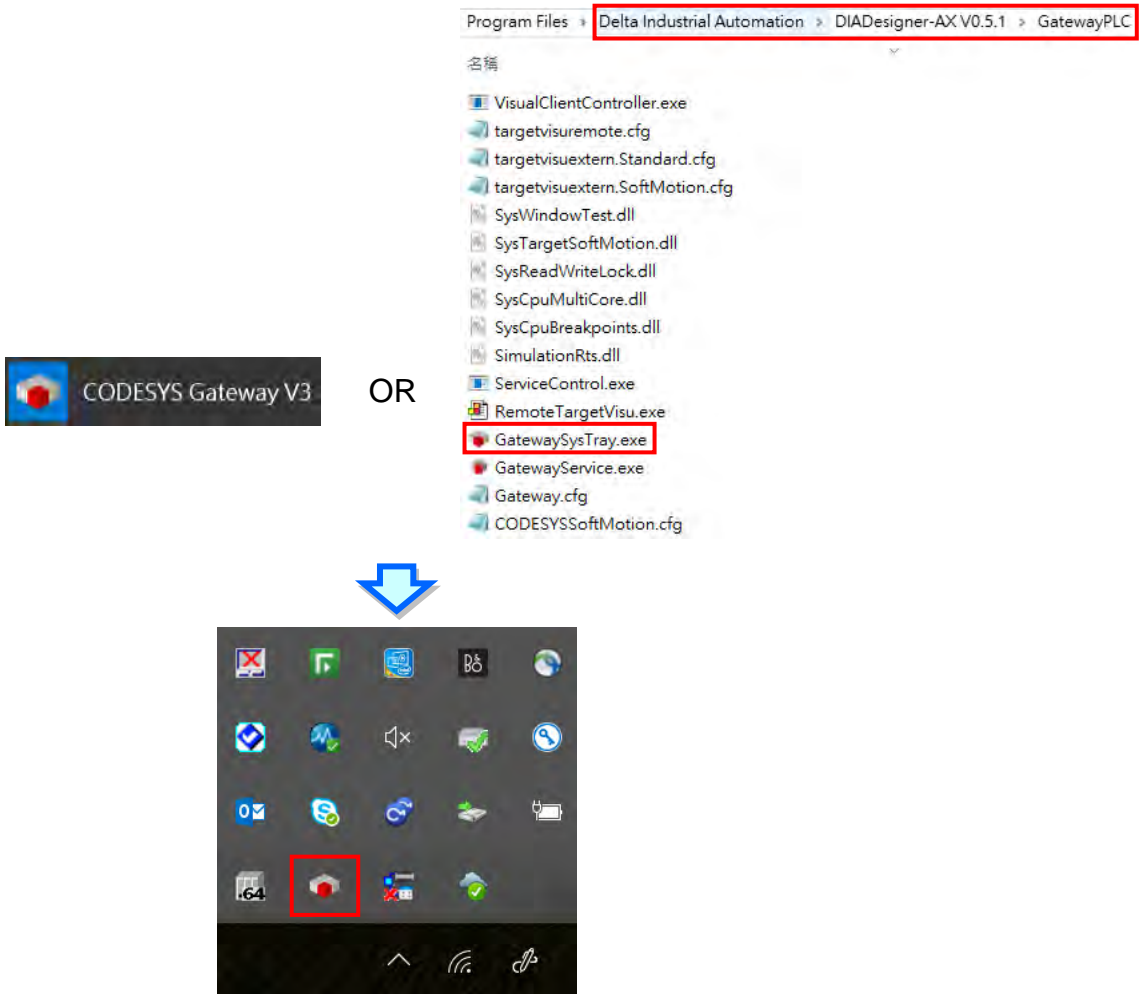


待欢迎画面结束后便可正式进入 DIADesigner-AX。

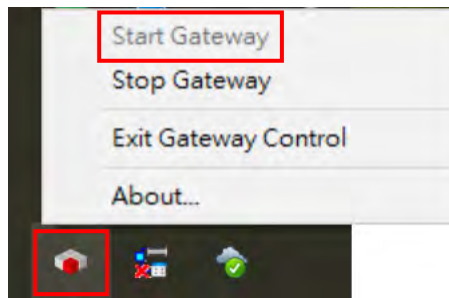


3.3.2 通讯

顺利完成 DIADesigner-AX 的安装后，系统便会自动于程序集当中建立一个执行快捷方式，直接单击该快捷方式即可启动 Gateway；之后每次计算机开机，系统便会自动启动 Gateway，并将其常驻在 Windows 的系统列当中；但若系统列中没有出现该图示时，请直接由程序集当中的快捷方式或 GatewaySysTray 的执行文件路径自行启动。



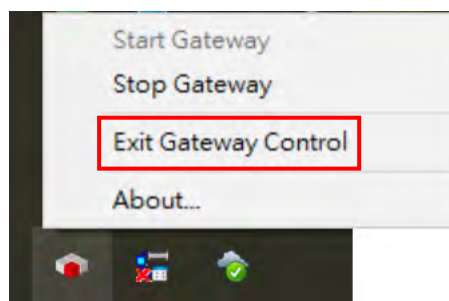
当顺利启动 Gateway 之后，于 Windows 的系统列当中便可发现 Gateway 的图标，于该图标上先使用鼠标单击后，可设定为「Start Gateway」或确认图示已为红色即为 Start 状态。



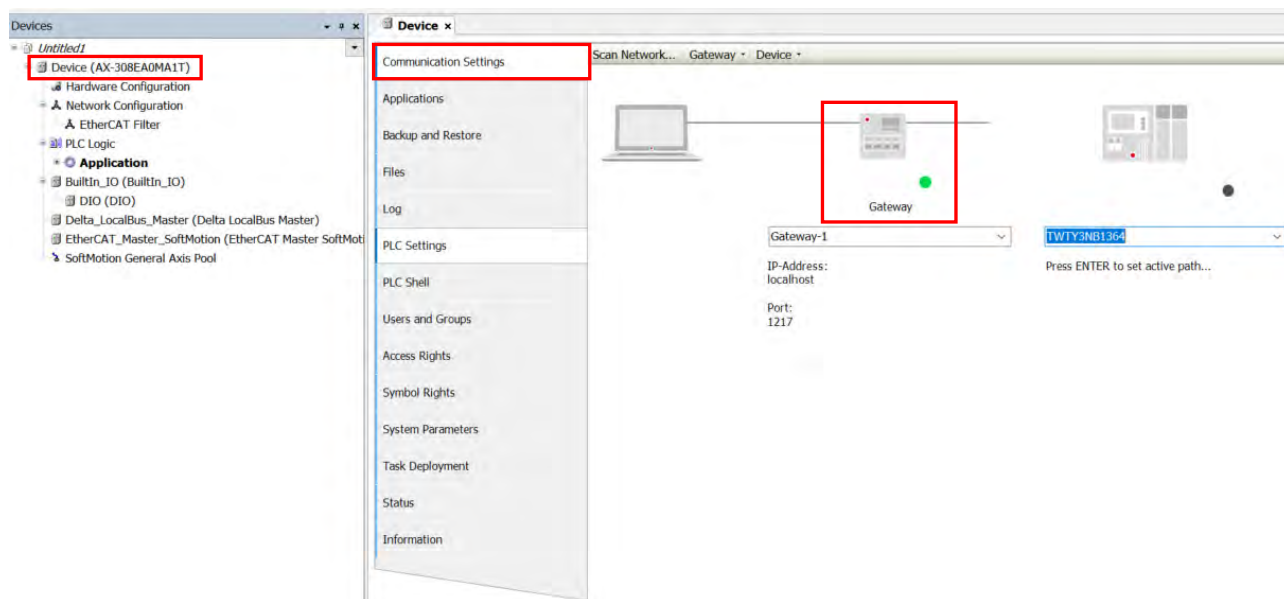
若要停止 Gateway 运行，一样于该图标上先使用鼠标单击过，再单击为「Stop Gateway」后确认图示变更为灰色即为 Stop 状态。



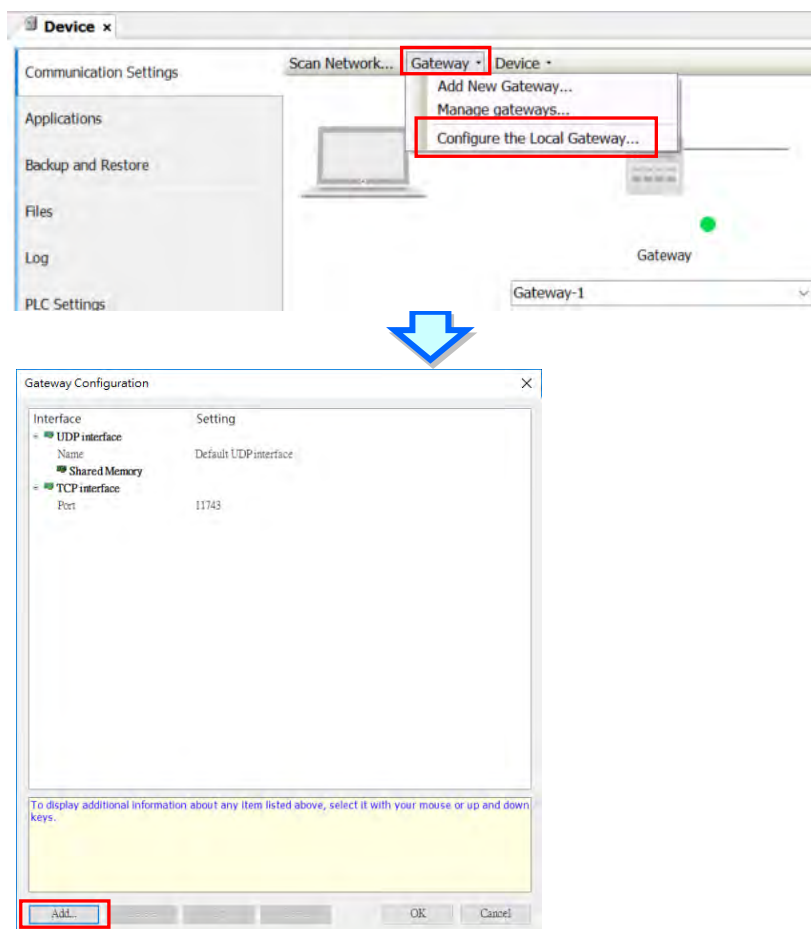
若要完全关闭 GatewaySysTray 时，请于该图示上单击过后选择「Exit Gateway Control」即可，该图标也立即于系统列中移除。



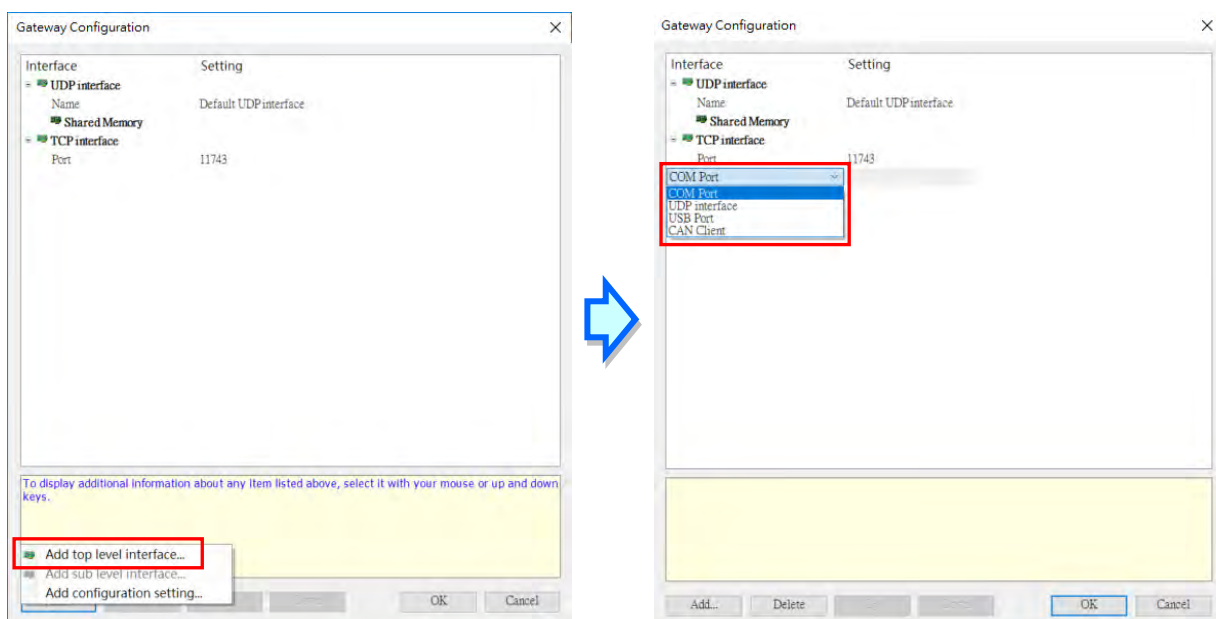
设定好启动 GatewaySysTray 后，开启 DIADesigner-AX 且成功打开项目后，于左方设备列中使用鼠标左键双击「Device」，Device 设定的分页即会出现在主画面中，且选单位置在「Communication Settings」；若 Gateway 已为 Start 状态，Gateway 旁的灯号会为绿灯，若为 Stop 状态，灯号会为红灯。



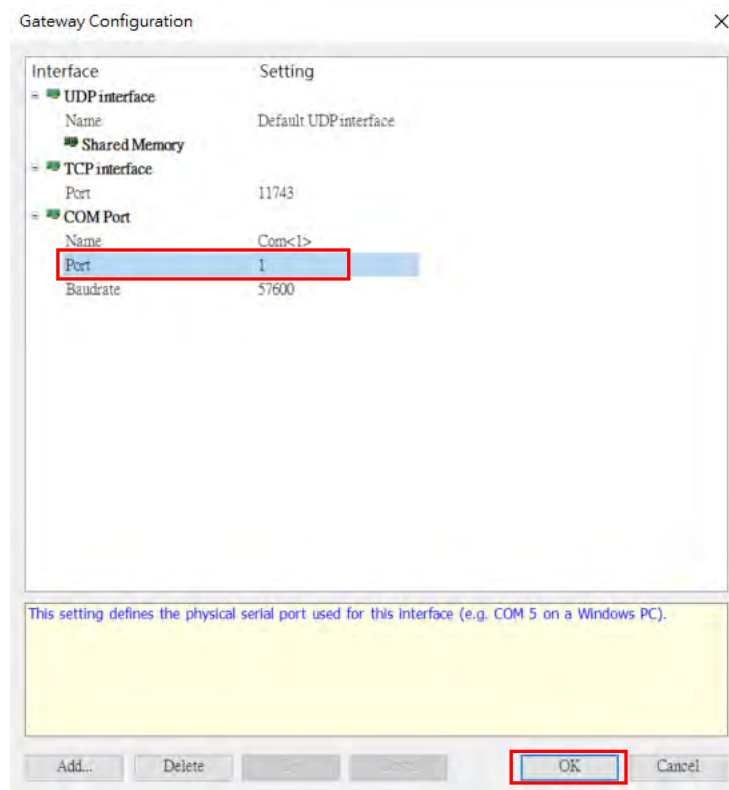
若要设定 Local Gateway，使用鼠标左键单击 Device 分页上方选项「Gateway」后，可从下拉式菜单中选择「Configure the Local Gateway...」，即会跳出设定窗口。




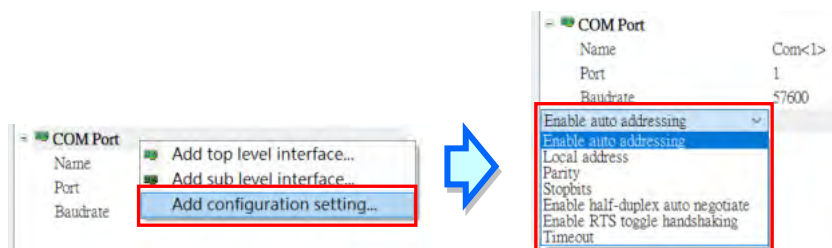
若联机须要使用 COM port，可于窗口左下角单击按钮「Add...」后从菜单中选择「Add top level interface...」，再从上方出现的下拉式菜单中选择「COM Port」。



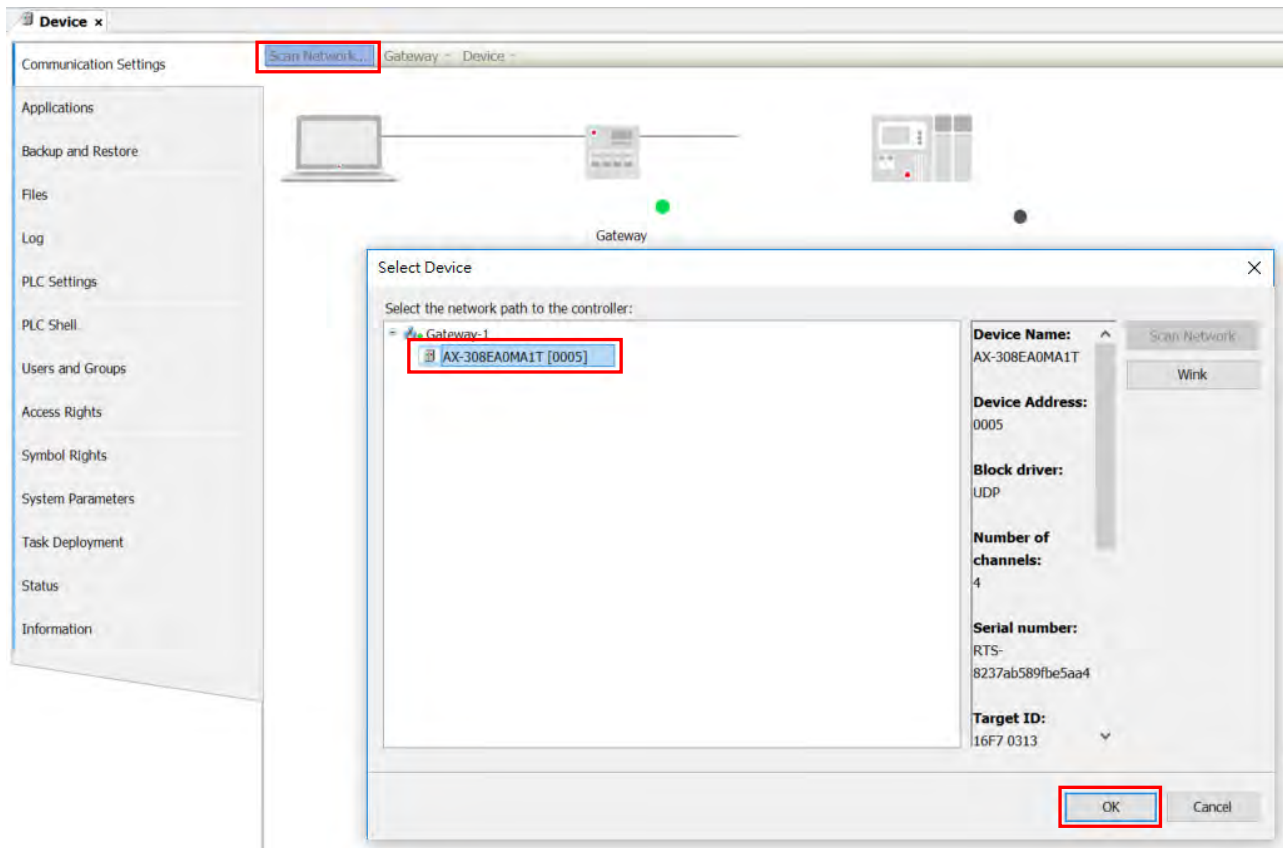
在 COM Port 下的参数中设定好对应的 Port 即可，确认后再按下右下角按钮「OK」。在新增后，GatewaySysTray 须要设定 Stop/Start 一次，后续 Scan Network 才会正常动作。



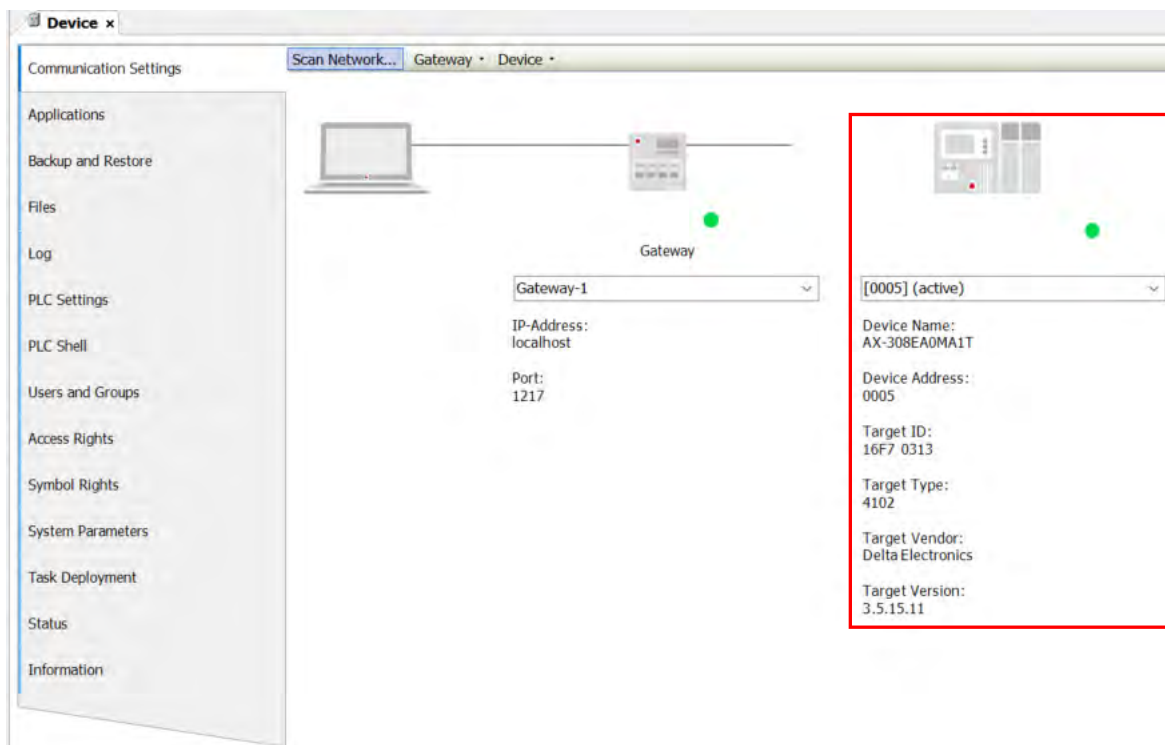
若要在 COM Port 下增加可以设定的参数，可在 Interface 下图示  COM Port 使用鼠标右键单击后，再从选单中选择「Add configuration setting...」，即可在下拉式选单中选择新增要增加的项目。



设定好 Local Gateway 后，使用鼠标左键单击 Device 分页上方选项「Scan Network...」即会跳出「Select Device」窗口，再单击要联机的设备「AX-308EA0MA1T」，确认后再按下右下角按钮「OK」。



完成好上述动作后，若成功联机，Device 页面内的设备右下角灯号会显示绿灯，且设备的相关内容会显示在设备的图标下方。



第4章 基本操作

目录

4.1	DIADesigner-AX 软件操作	4-3
4.1.1	建立项目	4-3
4.2	Device 页面介绍	4-5
4.2.1	主机参数设定	4-5
4.2.1.1	Communication Settings	4-5
4.2.1.2	Applications	4-8
4.2.1.3	Backup and Restore	4-9
4.2.1.4	Files	4-10
4.2.1.5	Log	4-10
4.2.1.6	PLC Settings	4-12
4.2.1.7	PLC Shell	4-13
4.2.1.8	Users and Groups	4-14
4.2.1.9	Access Rights	4-17
4.2.1.10	Symbol Rights	4-23
4.2.1.11	System Setting	4-24
4.2.1.12	System Parameters	4-26
4.2.1.13	Task Deployment	4-27
4.2.1.14	Status	4-28
4.2.1.15	Information	4-28
4.2.2	扩展模块参数设定	4-29
4.3	数据类型与变量	4-30
4.3.1	数据类型	4-30
4.3.2	变数	4-31
4.3.2.1	变量宣告	4-31
4.3.2.2	寻址	4-32
4.3.2.3	变量类型	4-33
4.3.2.4	自定义数据类型	4-37
4.3.2.5	变量清除为零时机说明	4-40
4.3.2.6	变数初始值生效时机	4-40
4.4	控制器原理	4-41




4.4.1	工作 (Task)	4-41
4.4.1.1	工作 (Task) 类型	4-41
4.4.1.2	多工作 (Task) 执行	4-42
4.4.1.3	Watchdog	4-43
4.4.1.4	每种任务类型可以执行的运动控制指令	4-43
4.5	配方管理器	4-46
4.5.1	配方管理器	4-48
4.5.2	配方定义	4-50
4.5.3	RecipeManCommands	4-52

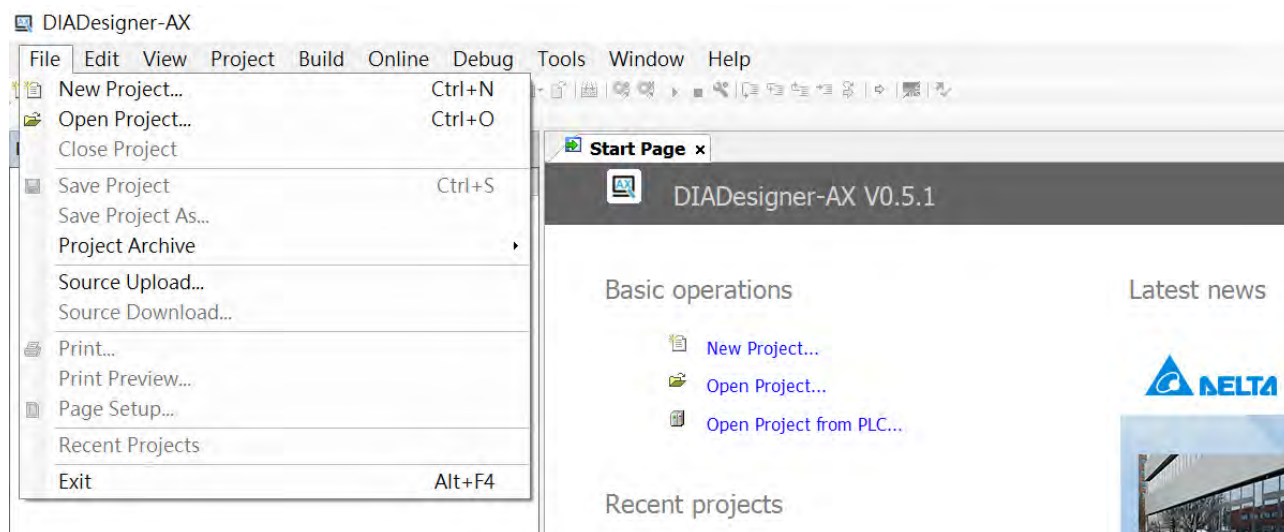
4.1 DIADesigner-AX 软件操作

DIADesigner-AX 是可编程逻辑控制 PLC 的开发环境且是整个自动化开发平台的核心，它为 IEC61131-3 所定义的编程语言提供了一个简单的方法，系统编辑器的功能是建立在高级编程语言的基础上，功能强大，易于开发，可靠性高，开放性好并且集成了 PLC、可视化、运动控制及安全的组件。

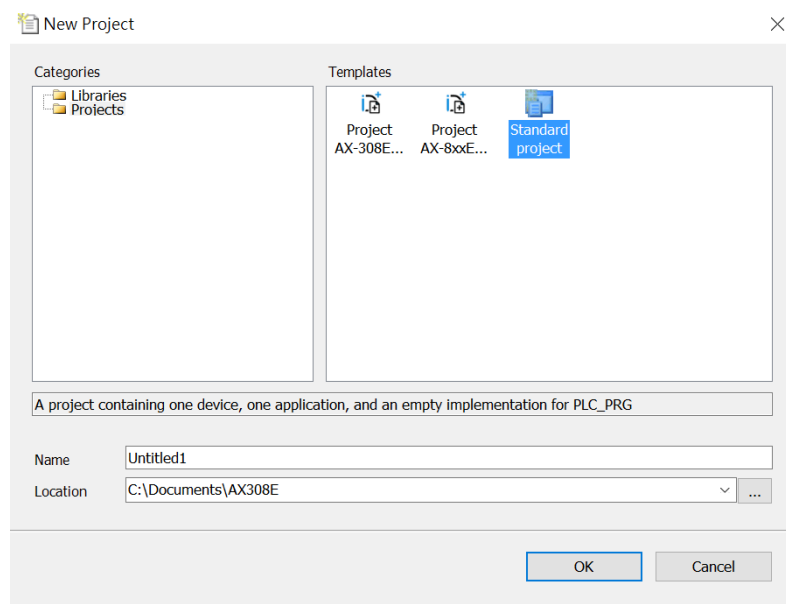
在 DIADesigner-AX 中，所有的窗口都不是固定的，用户可以根据自己的习惯将其通过鼠标拖曳的方式移动到目标位置，将窗口进行重新排列。

4.1.1 建立项目

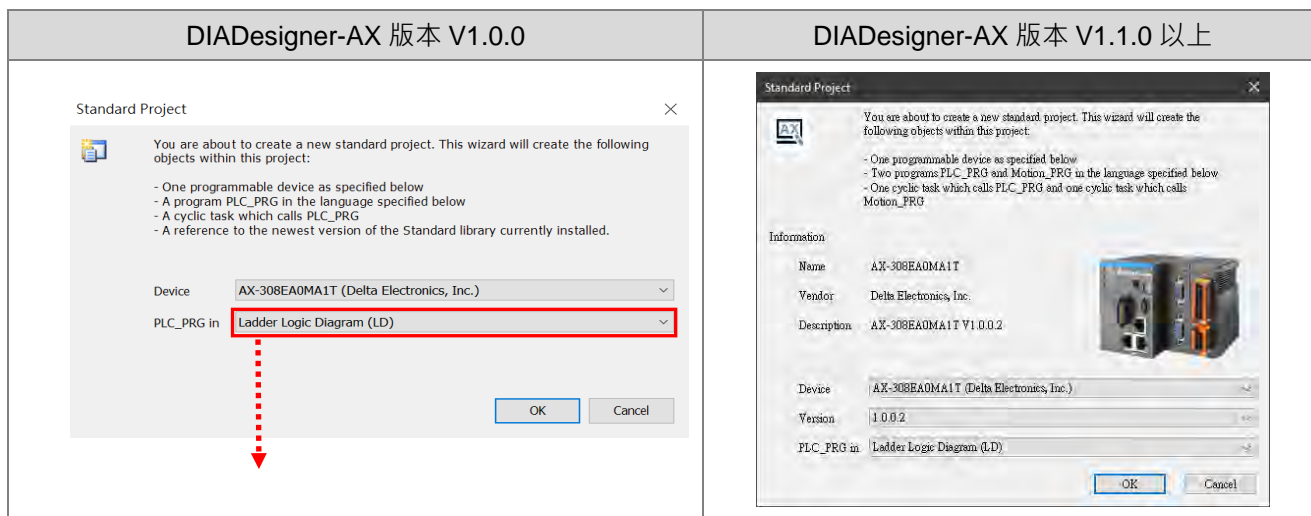
双击桌面编程软件图标 ，启动 DIADesigner-AX 后，在工具栏中单击「File」>「New Project」或直接单击  图标；或在「Start Page」分页内单击  图标旁「New Project...」。



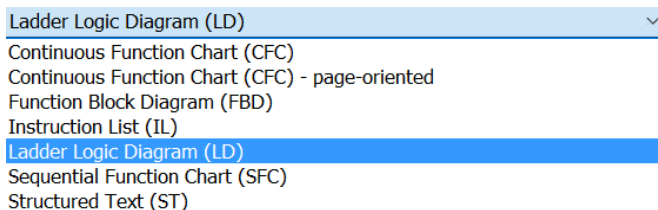
接着在新项目建立窗口中选择标准项目，并为项目依序输入「Name」及「Location」，完成后按下「OK」。



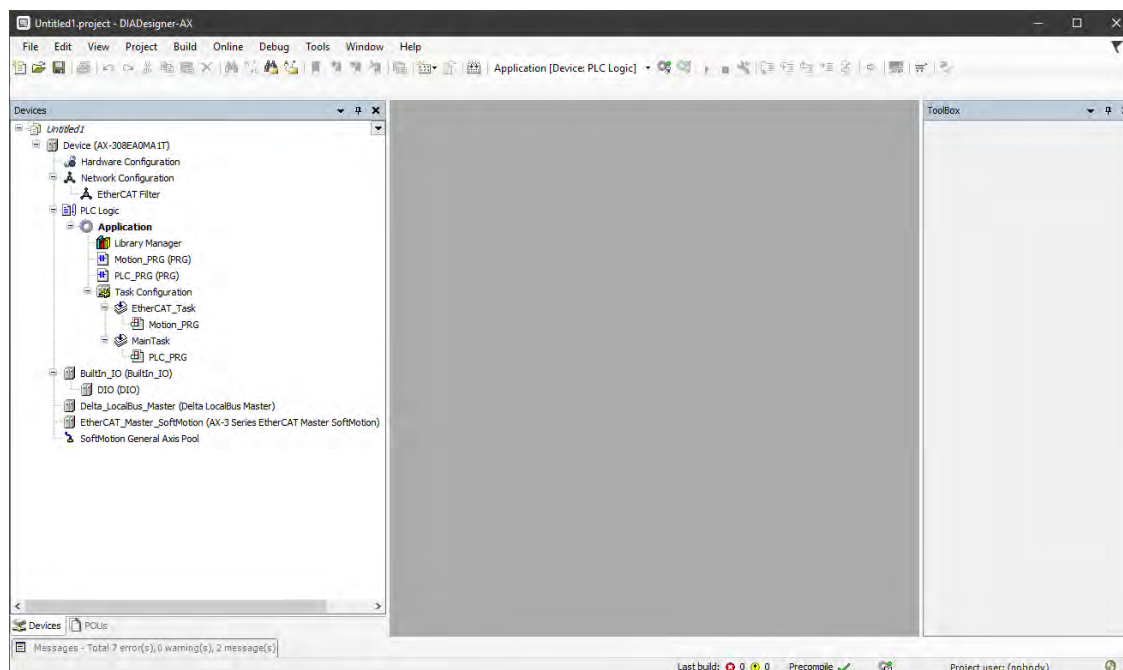
新建标准项目最后根据对话框提示，选择「Device」及「PLC_PRG in」，完成后按下「OK」，系统会自动生成一个默认为周期循环型的任务，任务中包含一个默认的程序 PLC_PRG。



不同编程语言可下拉后选择。

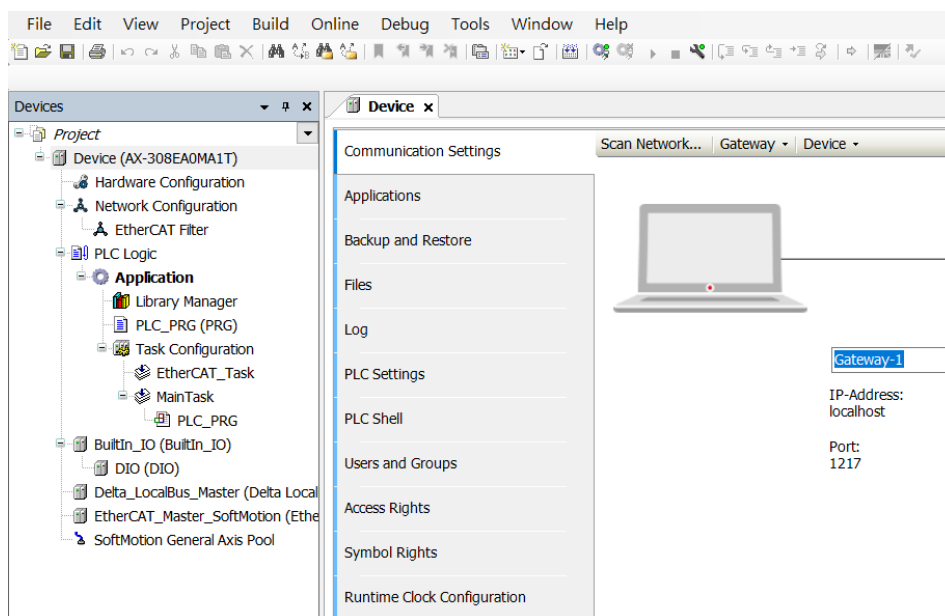


成功建立项目后，画面的左侧便会开启一个项目管理区，并以树形图的方式列出所有对象；若未出现该区域时，请在工具栏中单击「View」>「Devices (Alt+0)」即可。



4.2 Device 页面介绍

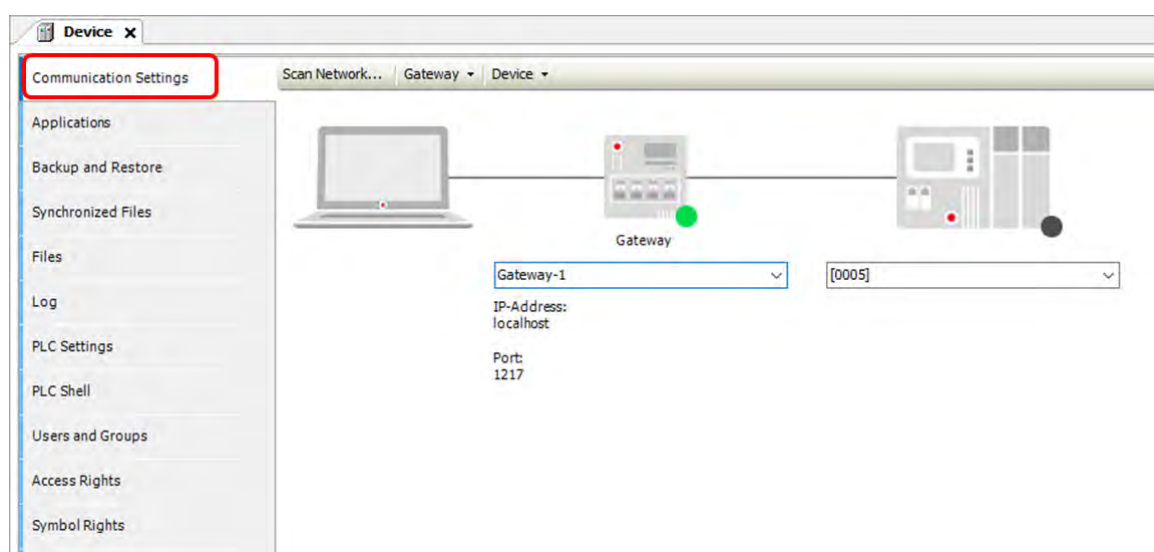
本章节将对 Device 内所有页面功能，个别进行介绍。



4.2.1 主机参数设定

4.2.1.1 Communication Settings

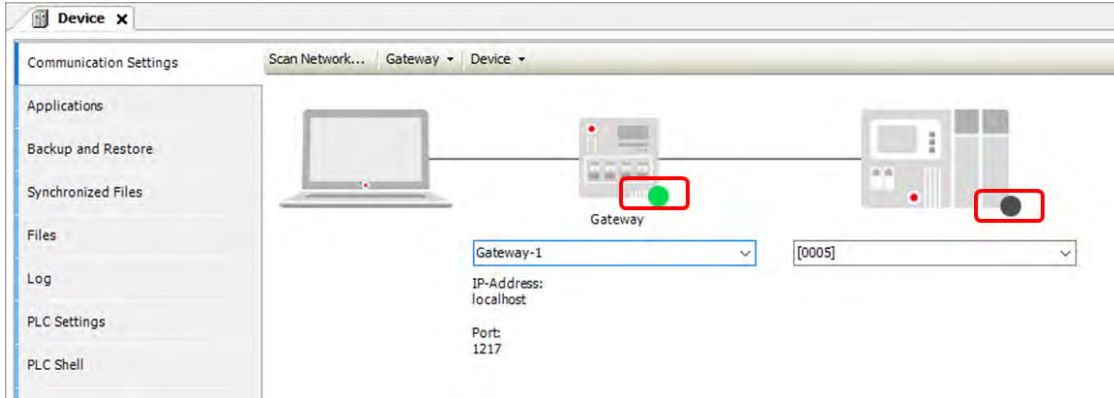
在「Communication Settings」页面，可以自行定义 DIADesigner-AX 与控制器之间的通讯方式，并可以在下拉式选单内，选择「gateway」与所欲连结的控制器，用户可利用「Manage gateways」与「Manage favorite devices」来对 gateway 与控制器进行管理，也可以直接输入控制器的 IP（如：192.168.1.5）或控制器名（如：AX-308EA0MA1T），当输入完毕后，DIADesigner-AX 会自动寻找 gateway 网络内的控制器。



● 联机状态

Gateway 与控制器图标右下角的图案代表当前联机状态。

- 红色：无法建立联机。
- 绿色：联机已建立。
- 黑色：未知的联机状态。



4

选单	说明
Scan network	用户可以通过此选单开启「Select Device」对话框，这里列出所有设定的 gateway 与其相关联的控制器，用户可以通过清单选择控制器。
Gateway	<p>选单包含了下列功能键：</p> <ul style="list-style-type: none"> ● Add New Gateway : 开启 Gateway 对话框，可在此定义新的 gateway 通道。 ● Manage Gateways : 开启「Manage Gateways」对话框，此对话框内列出所有 gateway，用户可在此进行 gateway 异动。 ● Configure the Local Gateway : 开启「Gateway Configuration」对话框，用户可以在此设定 gateway 的驱动。
Device	<p>此选单包含了下列功能键：</p> <p>1. Options :</p> <ul style="list-style-type: none"> ● Add Current Device to Favorites : 增加当前所选择的控制器至「favorite device」清单。 ● Manage Favorite Devices : 开启常用控制器清单，用户可以在此进行清单异动。(最上层的控制器为默认控制器) ● Filter Network Scans by Target ID : <ul style="list-style-type: none"> <input checked="" type="checkbox"/> : 仅显示与项目设定相同 target ID 的控制器。 ● Confirmed Online Mode : <ul style="list-style-type: none"> <input checked="" type="checkbox"/> : DIADesigner-AX 于联机模式时进行下列操作动作显示确认窗口。 <ul style="list-style-type: none"> ■ Force values ■ Write values ■ Multiple loading ■ Remove force list

选单	说明
	<ul style="list-style-type: none"> ■ Single cycle ■ Start ■ Stop <ul style="list-style-type: none"> ● Store Communication Settings in Project : <ul style="list-style-type: none"> <input checked="" type="checkbox"/> : DIADesigner-AX 将通信设置储存在项目，但如果使用不同计算机开启此项目时，则必须重新设定。 <input type="checkbox"/> : DIADesigner-AX 将通信设置储存在计算机所安装的地址，以便重复使用相同的通信设置。 2. Rename Active Device : 开启 Change Device Name 对话框。 3. Send Echo Service : 此功能键是用于测试 DIADesigner-AX 与控制器之间的网络联机，DIADesigner-AX 会发出类似网络 Ping 功能的网络报文，分别送出五个没有数据的报文与五个最大数据长度的报文，并显示每次发送的响应时间与平均响应时间于新的对话框。 4. Encrypted Communication : <ul style="list-style-type: none"> <input checked="" type="checkbox"/> : 启动加密通讯，当进行 login 时则需要认证，如控制器认证不存在，则会显示错误信息，提示是否安装或是显示认证，当「Security Screen」内「Enforce Encrypted Communication」被选择，则此功能键即隐藏。 5. Change Communication Policy : 开启「Change Communication Policy」对话框，用户可在此对话框对控制器的加密通讯的动作进行设定。

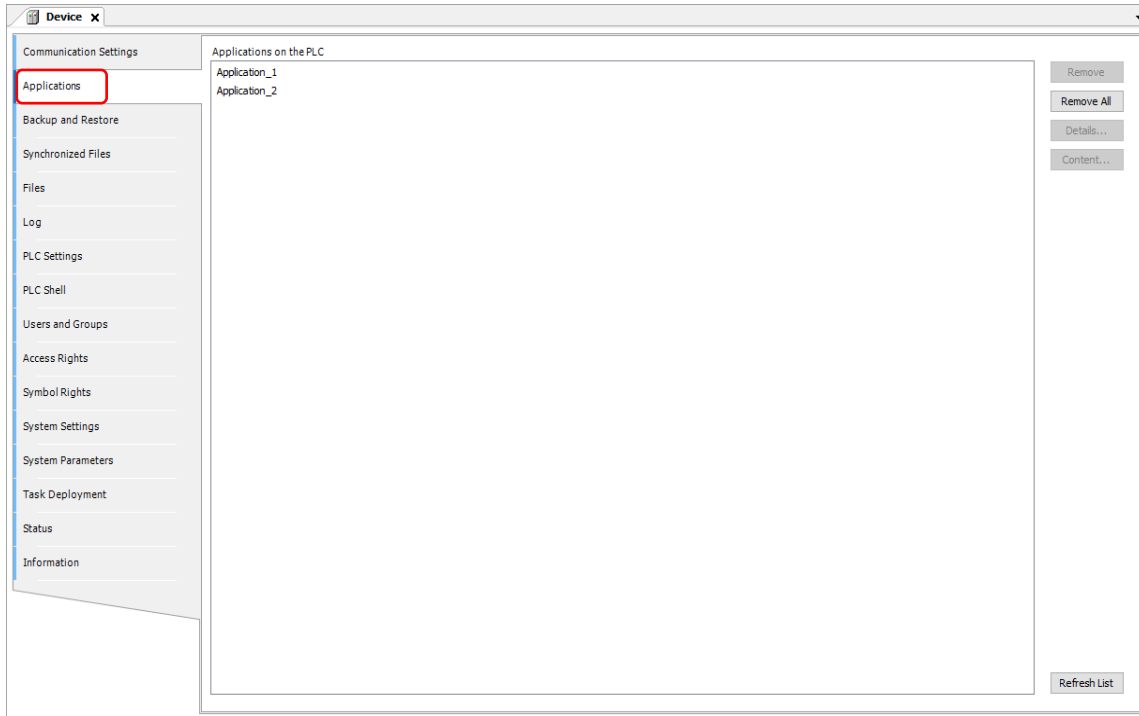
● Change Communication Policy 对话框

当 communication policy 设定变更后，则会连同控制器内的设定一起改变。

Communication	
Current policy	显示当前所选择通讯加密 policy。
New policy	<ul style="list-style-type: none"> ● No encryption : 不使用加密通讯。 ● Optional encryption : 可选择加密与非加密通讯。 ● Enforced encryption : 使用加密通讯。
Device User Management	
Current policy	显示当前用户所设定的 policy。
New policy	<ul style="list-style-type: none"> ● Optional user management : 用户可自行决定是否启动加密通讯。 ● Enforced user management : 用户必须使用加密通讯。

4.2.1.2 Applications

用户可以通过「applications」页面，确认当前控制器内的 applications，用户可执行删除 applications 或是确认 applications 相关信息等动作...

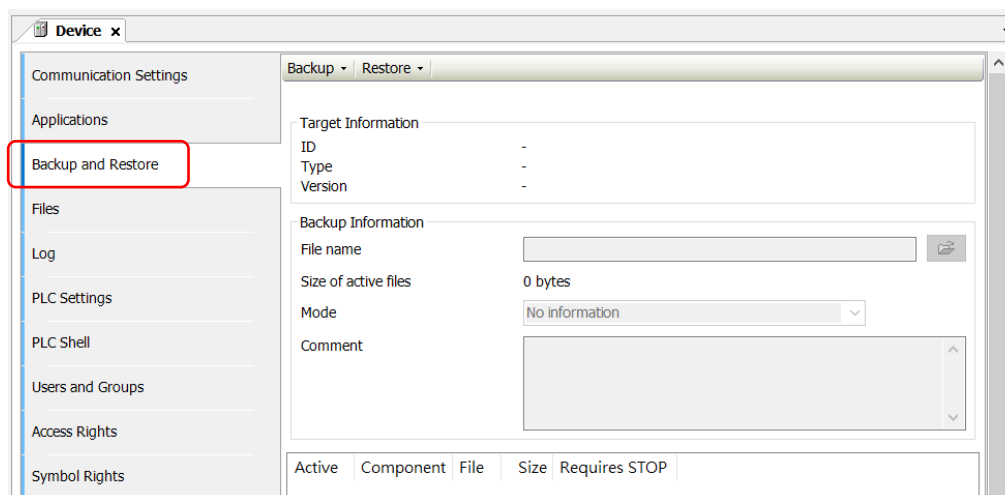


4

功能键	说明
Remove / Remove ALL	Delete：删除所当前选择的 application Delete ALL：删除当前控制器内所有的 applications
Details	开启 Details 对话框，此对话框显示了 application 其 Information 页面所设定的属性，如 Author、Version 与 Descriptions...
Contents	开启 Contents 对话框，此对话框显示了控制器与目前项目最后编译的内容。 注：需先将 application 物件的 Download the applications 勾选。 (其路径为 application > properties > Application generation options)
Refresh List	更新当前控制器内的 applications 列表

4.2.1.3 Backup and Restore

在 Backup and Restore 页面，用户可以通过储存与读取 zip archive 的方式对控制器的 application-specific 文件进行备份与还原。



选单	说明
Backup	<p>此页面包含了下列功能键：</p> <ul style="list-style-type: none"> ● Read Backup Information from Device：此功能键会列出控制器文件夹「PlcLogic」内的所 application-specific 文件。 ● Create Backup File and Save to Disk：此功能键会将所选择的文件建立一个压缩文件，供还原使用。（备份文件扩展名为.tbf） ● Save Backup File to Device：将备份文件储存至控制器。
Restore	<ul style="list-style-type: none"> ● Load Backup File from Disk：从计算机读取所储存的备份文件。 ● Load Backup File from Device：从控制器读取所储存的备份文件。 ● Restore Backup to Device：将备份文件还原至控制器。

● Target Information

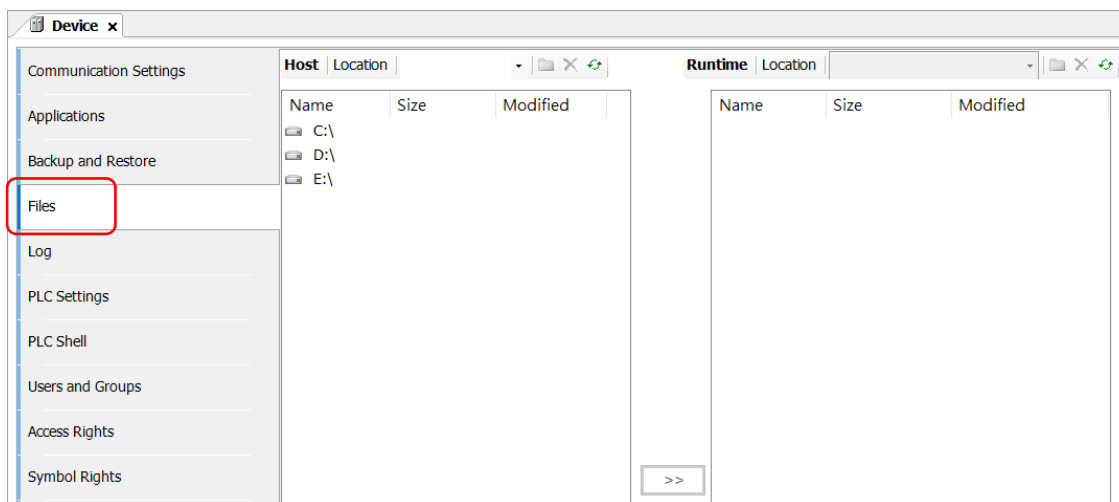
ID	公司代码与机种代码
Type	机种类型
Version	控制器韧体版本

● Backup Information






File name	备份文件的储存路径
Size of active files	备份文件大小
Mode	备份范围
Comment	用户可以在建立备份文件时，在此进行备份文件注释，并在读取备份文件时读取此信息，此信息保存在 meta.info 文件内。

4.2.1.4 Files

在 Files 页面用户可以通过 DIADesigner-AX 进行计算机与控制器之间的文件传输。



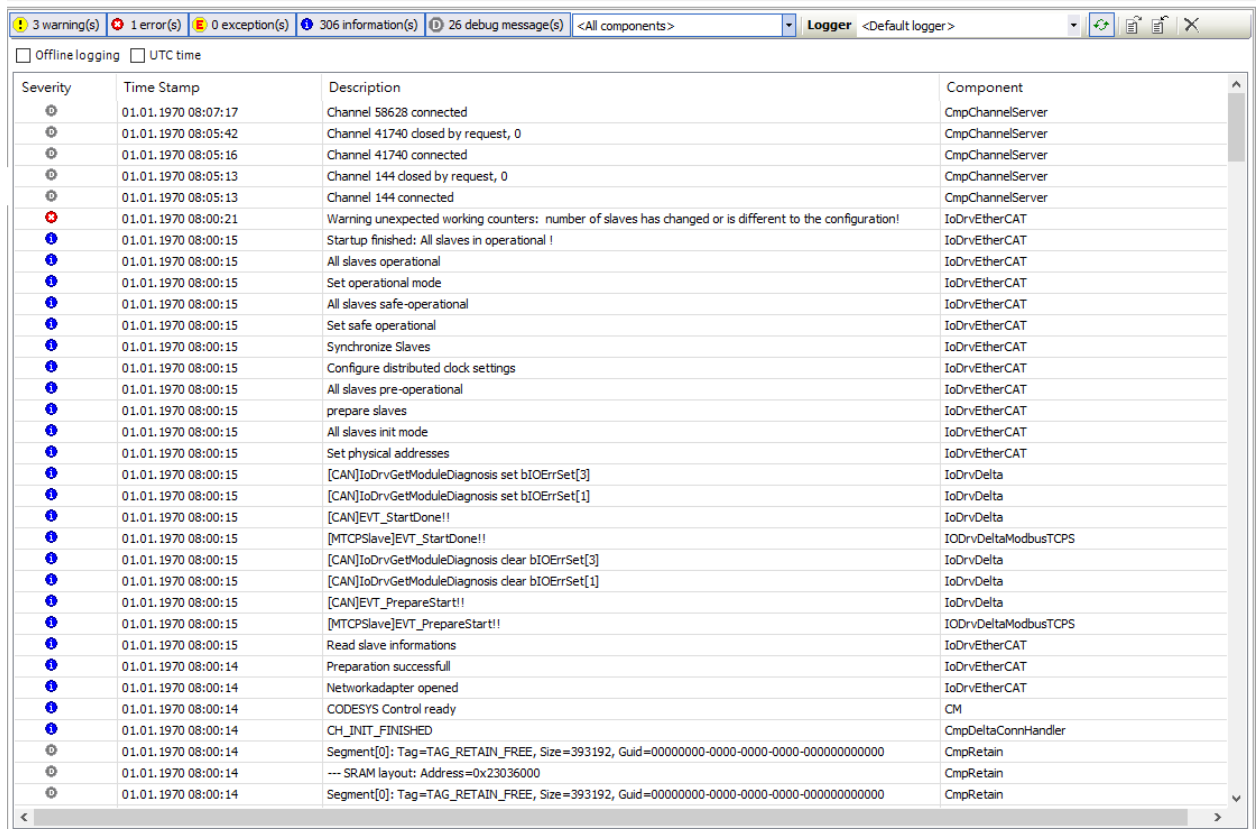
4

Location	系统当前路径，在下方的子目录会显示文件名，文件大小与修改日期。
	在当前文件路径建立新文件夹
	删除所选择的文件夹或是文件
	更新当前文件路径的文件与文件夹
	将文件写入控制器
	将文件从控制器读出

4.2.1.5 Log

在 Log 卷标中，列出了在系统上发生的事件记录。记录内容包含：

- 系统启动到关闭期间的事件（加载的组件，包含版本信息）
- 应用程序的下载及加载启动时的应用程序
- 自定义项目
- 从 I/O 驱动程序产生的日志
- 从数据来源产生的日志

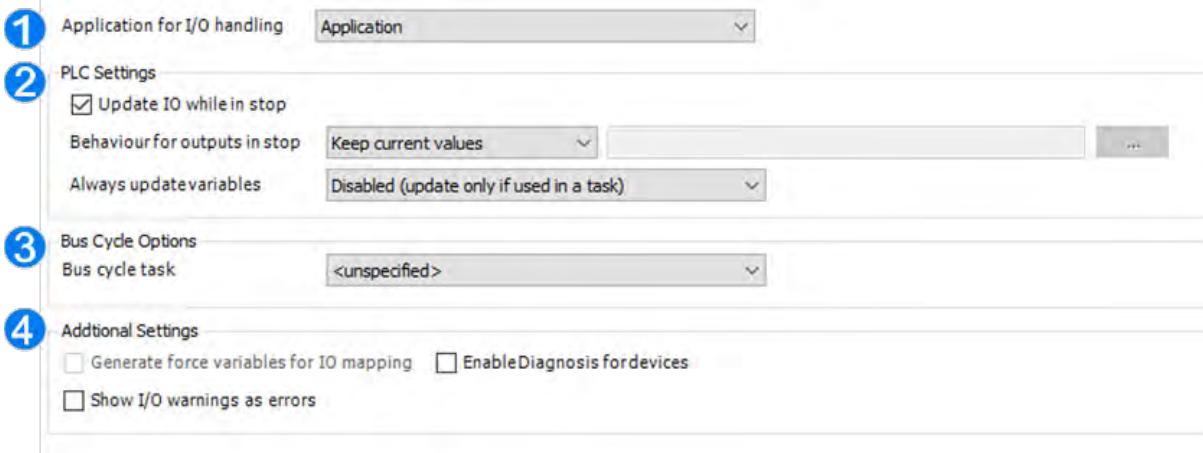


功能	说明
Offline logging	<input type="checkbox"/> : 默认值。 <input checked="" type="checkbox"/> : PLC 也记录与 AX-3 系列控制器连接无关的动作。但目前仅针对 CODESYS 的安全版本允许此项操作。
UTC time	<input type="checkbox"/> : 标准设置：时间会依照操作系统的时区转换为计算机上的本地时间。 <input checked="" type="checkbox"/> : 显示系统中实际运行的时间。
Severity	依据事件严重性分为四个类别： <ul style="list-style-type: none"> • : Message (信息) • : Warning (警告) • : Error (错误) • : Debugging (除错信息) 可以借助列表上方栏中对应的按钮来显示或隐藏每个类别，每个按钮有显示其类别的记录数。
Time stamp	日期-时间 (例如：12-01-2007 09 : 48)
Description	事件描述
Component	运行时，系统相关组件的名称。例如：CmpApp
Drop-down list with component names	日志列表仅显示与所选择之组件有关的事件
Logger	是一个可以下拉式的选单。标准设置为<Default Logger>，目前与 stdLogger 相同。

功能	说明
	重新整理日志列表
	将列表内容导出到 XML 文件
	将 XML 文件导入到日志列表
	清空显示的日志列表

4.2.1.6 PLC Settings

「PLC Settings」分页中，为通用设备的编辑，可以对 PLC 组态进行基本设置，例如：输入和输出的处理及循环任务的设定。



● ① Application for I/O handling

选单	说明
Application for I/O handling	负责 I/O 处理的应用程序

● ② PLC Settings

选单	说明
Update IO while in stop	<input type="checkbox"/> : 当 PLC 处于停止状态时，DIADesigner-AX 不会刷新输入和输出通道的值。 <input checked="" type="checkbox"/> : 当 PLC 处于停止状态时，DIADesigner-AX 也会刷新输入和输出通道的值。如果监视检测到故障，则会将输出设置为默认值。
Behaviour for outputs in stop	当 AX-3 系列控制器进入停止状态时，输出通道的处理： Keep current values：保持当前值。 Set all outputs to default：将所有 I/O 映像均设为默认值。 Execute program：可以通过此项设定，在 DIADesigner-AX 为 STOP 状态时，执行右侧程序。
Always update variables	全局设定，定义是否更新 bus cycle task 的 I/O 变量。只在从站和模块的更新设定中定义为「禁用」时，此设置才对从站和模块的 I/O 变量有效。

选单	说明
Always update variables	<p>Disabled (Update only if used in a task): 只在任务中使用 I/O 变量才能更新。</p> <p>Enabled1 (use bus cycle task if not used in any task): 如果没有在其它任何任务中使用，则会更新在 bus cycle task 中。</p> <p>Enabled2 (always in bus cycle task): 无论是否使用到 I/O 变量或映像到输入或输出通道，均会在 bus cycle task 的每个循环中更新所有的变量。</p>

- ③ Bus Cycle Options

选单	说明
Bus cycle task *1	控制 bus cycle 的任务。在默认情况下，由设备描述的定义进入任务周期。

注 1：将 bus cycle task 选择设定为<unspecified>，应注意，“<unspecified>”表示设备描述中给的默认值生效。因此应该检查此描述档。可以将使用最短循环时间的任务定义为默认值，但是同样的也能将最常循环时间的任务定义为默认值。

- ④ Additional Settings

选单	说明
Generate Force variables for I/O mapping	设备不支持此设定
Enable Diagnostics for devices	<input checked="" type="checkbox"/> : DIADesigner-AX 自动将 library CAA 设备诊断整合到项目中，并且建立一个隐藏式的功能块。如果设备中已经有功能块，则使用扩展功能块(例如：使用 EtherCAT) 或添加一个功能块实例。然后这包含设备诊断的一般实现。
Show I/O warnings as errors	有关 I/O 配置的警告显示为错误

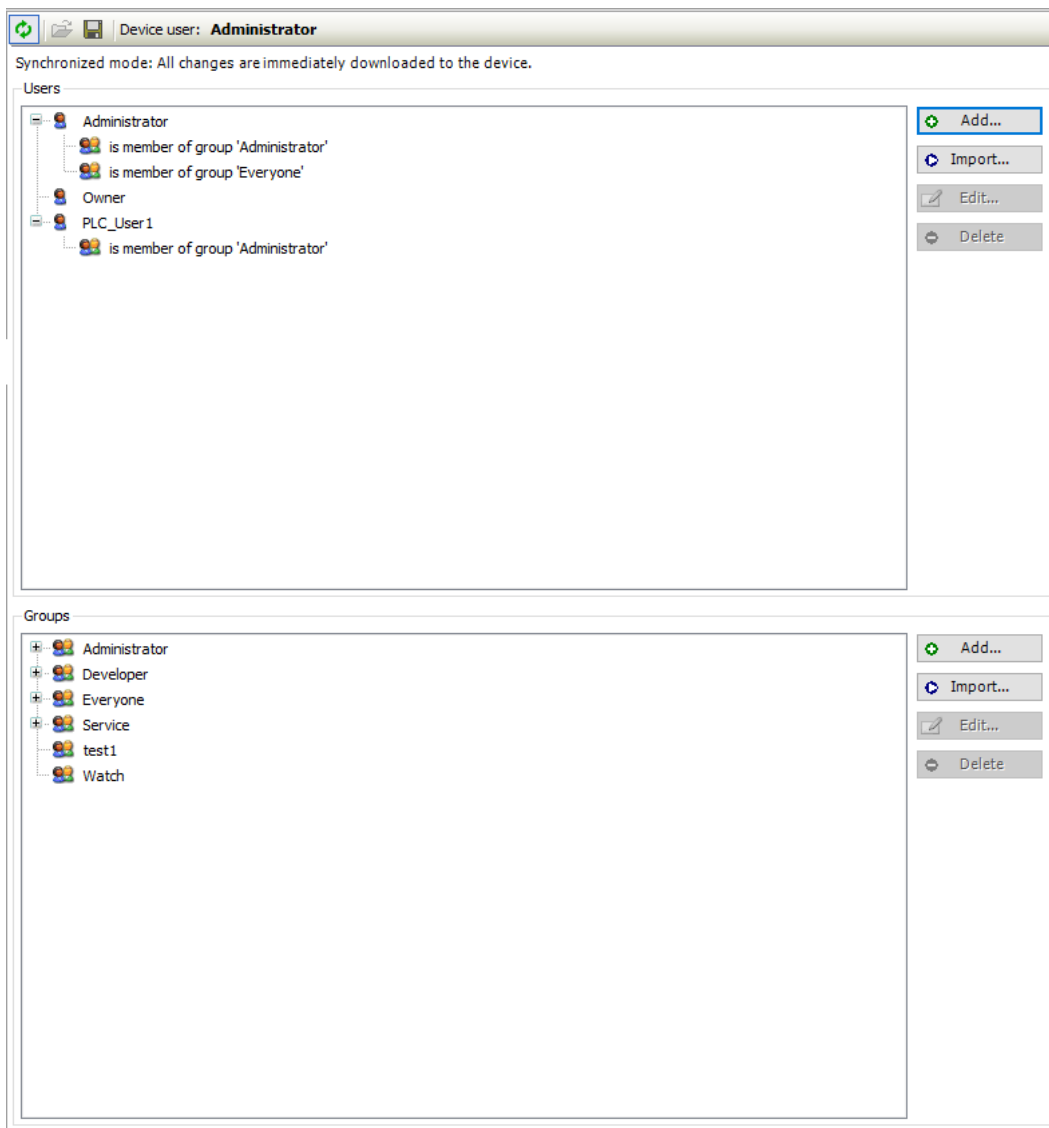
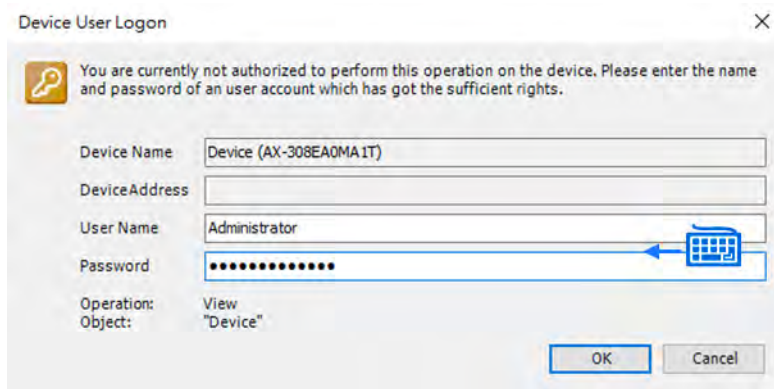
4.2.1.7 PLC Shell

用户可以在此页面，搭配特定的命令来查询控制器相关信息，并显示在此页面。






4.2.1.8 Users and Groups

编辑 AX-3 系列控制器的用户帐户及群组管理，在运行中可以控制的对象及浏览的权限设定。首次使用时，默认之用户账号 Administrator 密码 Administrator，登入后并进行密码修改设定。







4

- Toolbar of the tab

功能键	说明
 Synchronization	打开和关闭设备上编辑器和用户管理间的同步功能。 1. 未按下按钮：编辑器为空白。 2. 按下按钮：编辑器中会与目前连接设备的用户进行同步。 3. 编辑器与设备上之设定不同时，则会有提示信息，选项： Upload from the device and overwrite the editor content ：从设备中上载到编辑器，覆盖目前内容。 Download the editor content to the device and overwrite the user management there ：将编辑器内容下载至设备中，并应用它。
 Import from disk	从文件中导入用户管理配置
 Export to disk	将目前用户管理配置导出储存到文件
Device user	目前登入设备的用户名称

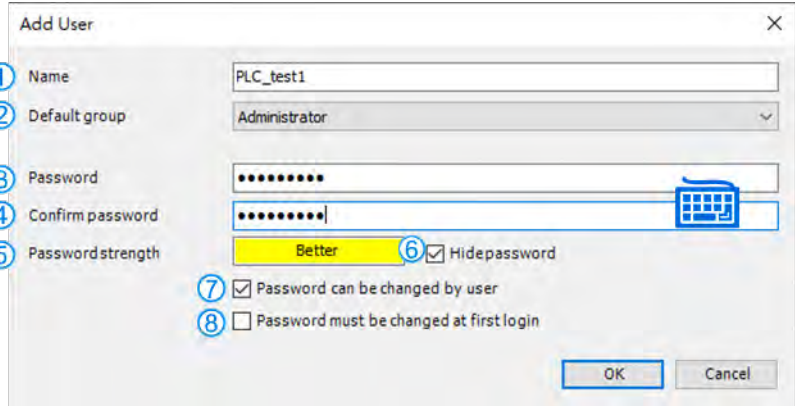
- Users

功能键	说明
 Add	新增用户*注一
 Import	打开 Import Users 对话框导入用户*注二
 Edit	编辑所选择之用户帐户
 Delete	删除所选择之用户帐户

- Groups

功能键	说明
 Add	新增群组*注三
 Import	打开 Import Groups 对话框导入用户*注四
 Edit	编辑所选择之群组
 Delete	删除所选择之群组

注一：新增用户对话框「Add user」

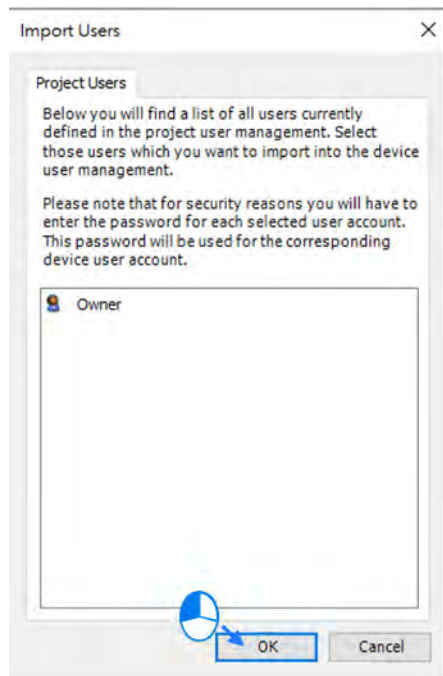


The screenshot shows the 'Add User' dialog box with the following fields and options:

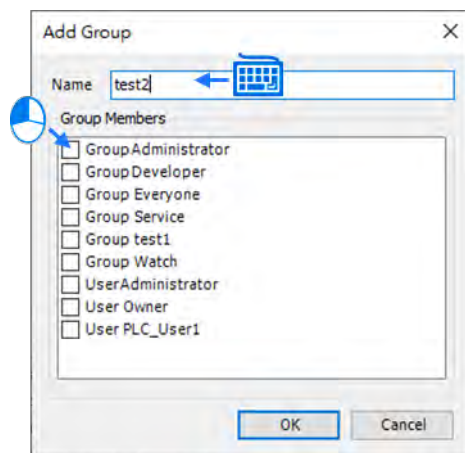
- ① Name: PLC_test1
- ② Default group: Administrator
- ③ Password: [masked]
- ④ Confirm password: [masked]
- ⑤ Password strength: Better
- ⑥ Hide password
- ⑦ Password can be changed by user
- ⑧ Password must be changed at first login

项目	项目	说明
①	Name	用户名称
②	Default group	选择用户默认群组（下拉式选单）
③	Password	密码
④	Confirm password	确认密码
⑤	Password strength	密码的安全性。从 Very weak 到 Very good 等级。
⑥	Hide password	<input checked="" type="checkbox"/> ：输入密码时，仅显示星号" * "。
⑦	Password can be changed by user	<input checked="" type="checkbox"/> ：密码允许被修改
⑧	Password must be changed at first login	<input checked="" type="checkbox"/> ：密码必须在第一次登入时被修改变更

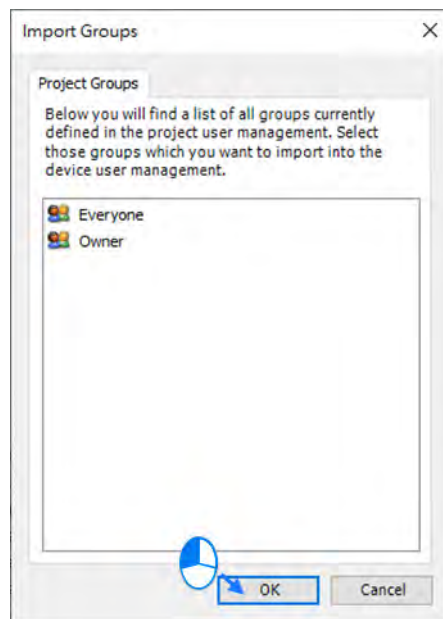
注二：「Import Users」对话框：选择后，按「OK」。



注三：「Add Group」对话框：输入组名，并且可以勾选群组成员同时加入群组中。



注四：「Import Groups」对话框：选择后，按「OK」。



4.2.1.9 Access Rights

针对用户对 AX-3 系列控制器上之对象控制及浏览的权限设定。用户管理项目和「users and groups」相同，用户必须是至少一个用户群组的成员，且只能针对用户群组设定操作权限。

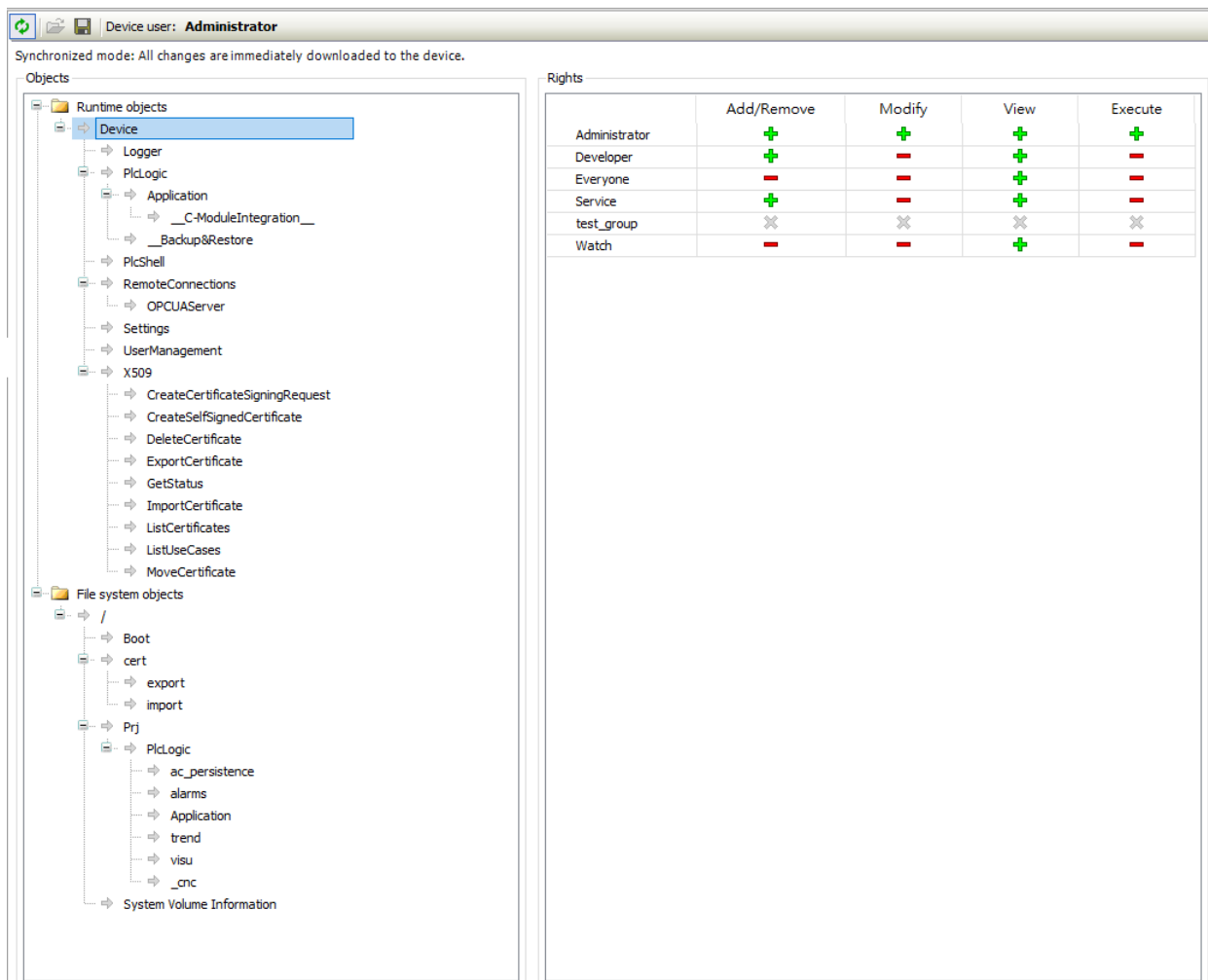
显示 Access Rights 的要求：

- 在 DIADesigner-AX 选项里，必须选择设备编辑器类别才会在显示权限设定的画面。

Note：DIADesigner-AX 选项可以被设备描述覆盖。

授予 Access Rights 的要求是针对用户群组设定：

- 用户管理组件必须在这 AX-3 系列控制器上是可用的，这是最主要的要求。
- 用户及用户群组设定在「Users and Groups」分页做设定。



4

● 工具栏

功能键	说明
Synchronization	打开和关闭设备上编辑器和用户管理间的同步功能 1. 未按下按钮：编辑器为空白 2. 按下按钮：编辑器中会与目前连接设备的用户进行同步 3. 编辑器与设备上之设定不同时，则会有提示信息，选项： <ul style="list-style-type: none"> ● Upload from the device and overwrite the editor content：从设备中上载到编辑器，覆盖目前内容。 ● Download the editor content to the device and overwrite the user management there：将编辑器内容下载至设备中，并应用它。
Import from disk	从文件中导入用户管理配置
Export to disk	将目前用户管理配置导出储存到文件
Device user	目前登入设备的用户名称

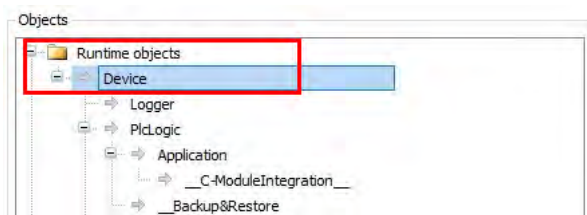
● Object

说明
在树状结构中，列出可以在运行时对其执行动作的对象。每个对象依其分类分组排列展开。在 Rights 画面中，可以对所选择的对象去设定群组所拥有的权限。
Object source (根节点) <ul style="list-style-type: none"> ● <i>Runtime objects > Device</i> : 这些对象中，可以将权限授予 AX-3 系列控制器目前执行目录的文件夹。 ● <i>File system objects > /</i> : 这些对象中，所有在 AX-3 系列控制器中具有在线存取的对象都受到管理，因此必须控制访问权限。 所有对象的描述都在表中， <i>Overview of the objects</i> 。
Object groups and objects (综合) 例如： <i>Device</i> 有子节点 <i>Logger</i> 、 <i>PlcLogic</i> 、 <i>Settings</i> 、 <i>UserManagement</i>

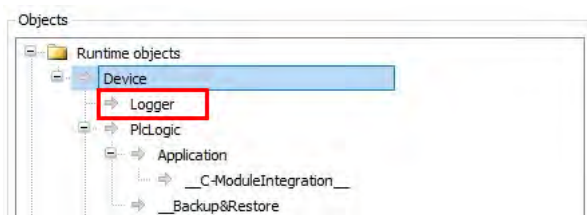
● Rights

说明																																			
通常，访问权限是从根对象 (<i>Device</i> 或是/) 继承到子对象的。如果群组的权限被禁止或授予在父对象中，则会影响所有子对象。 这个表适用于树状结构中选定的对象。对于每个用户群组，显示目前对于此对象能做的存取设定权限。																																			
<div style="display: flex; align-items: flex-start;"> <div style="flex: 1;"> <p style="font-size: small;">Synchronized mode: All changes are immediately downloaded to the device.</p>  </div> <div style="flex: 2;"> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="font-size: small;">Rights</th> <th style="font-size: small;">Add/Remove</th> <th style="font-size: small;">Modify</th> <th style="font-size: small;">View</th> <th style="font-size: small;">Execute</th> </tr> </thead> <tbody> <tr> <td style="font-size: x-small;">Administrator</td> <td style="color: green;">+</td> <td style="color: green;">+</td> <td style="color: green;">+</td> <td style="color: green;">+</td> </tr> <tr> <td style="font-size: x-small;">Developer</td> <td style="color: red;">-</td> <td style="color: red;">-</td> <td style="color: green;">+</td> <td style="color: red;">-</td> </tr> <tr> <td style="font-size: x-small;">Everyone</td> <td style="color: red;">-</td> <td style="color: red;">-</td> <td style="color: green;">+</td> <td style="color: red;">-</td> </tr> <tr> <td style="font-size: x-small;">Service</td> <td style="color: green;">+</td> <td style="color: red;">-</td> <td style="color: green;">+</td> <td style="color: red;">-</td> </tr> <tr> <td style="font-size: x-small;">test_group</td> <td style="color: red;">-</td> <td style="color: red;">-</td> <td style="color: red;">-</td> <td style="color: red;">-</td> </tr> <tr> <td style="font-size: x-small;">Watch</td> <td style="color: red;">-</td> <td style="color: red;">-</td> <td style="color: green;">+</td> <td style="color: red;">-</td> </tr> </tbody> </table> </div> </div>	Rights	Add/Remove	Modify	View	Execute	Administrator	+	+	+	+	Developer	-	-	+	-	Everyone	-	-	+	-	Service	+	-	+	-	test_group	-	-	-	-	Watch	-	-	+	-
Rights	Add/Remove	Modify	View	Execute																															
Administrator	+	+	+	+																															
Developer	-	-	+	-																															
Everyone	-	-	+	-																															
Service	+	-	+	-																															
test_group	-	-	-	-																															
Watch	-	-	+	-																															
对于对象能采取的行为： <ul style="list-style-type: none"> ● Add/Remove ● Modify ● View ● Execute 																																			
单击对象后，右侧 Rights 表会显示所选的对象，目前用户群组的访问权限。																																			
表格可以快速查看： <ul style="list-style-type: none"> ● 那些访问权限是由对象评估 ● 哪个群组用户对哪个对象有什么有效权限 表格中符号的涵义： <ul style="list-style-type: none"> ● +：明确授予访问权限 ● -：明确禁止访问权限 ● +：通过继承授予访问权限 ● -：通过继承禁止访问权限 ● X：没有明确授予或禁止访问权限，且也未继承父对象的访问权限。无法存取。 ● 没有符号：选择多个对象，且各对象具有不同的访问权限。 单击符号可以修改权限。																																			

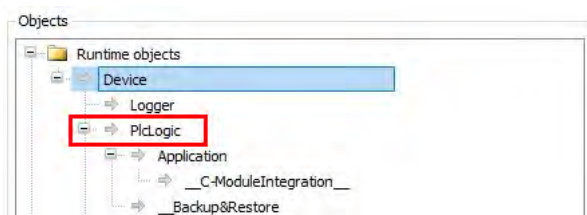
- 所有对象内容浏览：
- ◆ Runtime objects > Device



➤ **Logger**：在线使用 Logger 是只读。所以只能授予浏览或是禁止存取的权限。

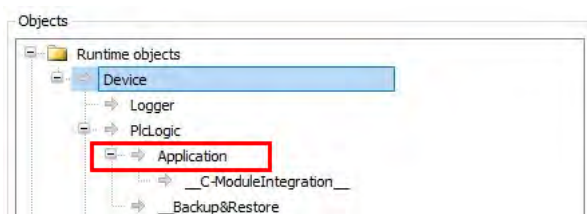


➤ **PlcLogic**：



在下载过程中，所有 IEC Application 都会作为子对象自动插入此处。当一个 Application 被删除时，它将自动被移除。对 Application 给在在线使用的特定控制。可以在 PlcLogic 中的所有 Application 集中分配其权限。管理员和开发人员用户群组中，对 IEC applications 有完全访问权限。服务和监视用户群组中，则只有读取权限（例如：只读监看数值）。

➤ **Application**：

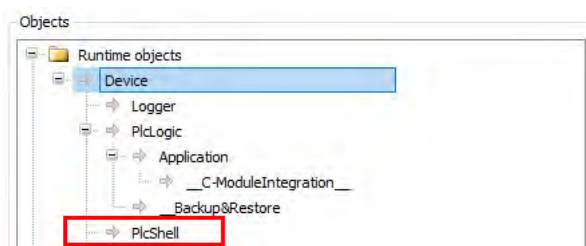


下表特别列出在 IEC application 中设定访问权限后，所受影响的动作。

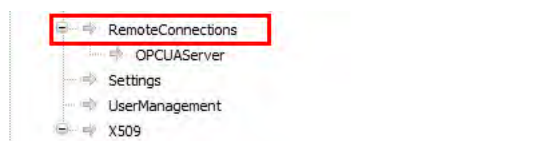
- x：禁止访问权限。
- ：允许访问权限。

	操作	访问权限			
		Add/Remove	Execute	Modify	View
Application	Login	●	●	●	x
	Create	x	●	●	●
	Create child object	x	●	●	●
	Delete	x	●	●	●
	Download / online change	x	●	●	●
	Create boot application	x	●	●	●
	Read variable	●	●	●	x
	Write variable	●	●	x	x
	Force variable	●	●	x	x
	Set and delete breakpoint	●	x	x	●
	Set next statement	●	x	x	●
	Read call stack	●	●	●	x
	Single cycle	●	x	●	●
	Switch on flow control	●	x	x	●
	Start / Stop	●	x	●	●
	Reset	●	x	●	●
	Restore retain variables	●	x	●	●
Save retain variables	●	●	●	x	

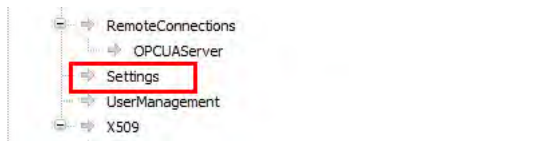
- PlcShell：此处只有「修改」权限。只有当修改权限授予用户群组后，才能使用 PLC shell 命令。



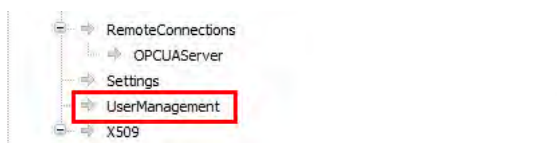
- RemoteConnections：可以在此节点下方配置到 AX-3 系列控制器的其它外部连结。一般，OPC UA server 的使用可以在此处设定。



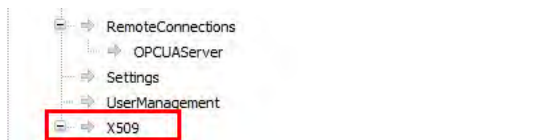
- **Settings** : 在线使用 · 对 AX-3 系列控制器的配置设定。默认为修改权限只授予管理员。



- **UserManagement** : 在线使用 · 对 AX-3 系列控制器的用户管理。默认为读取/写入权限只授予管理员。



- **X509** :



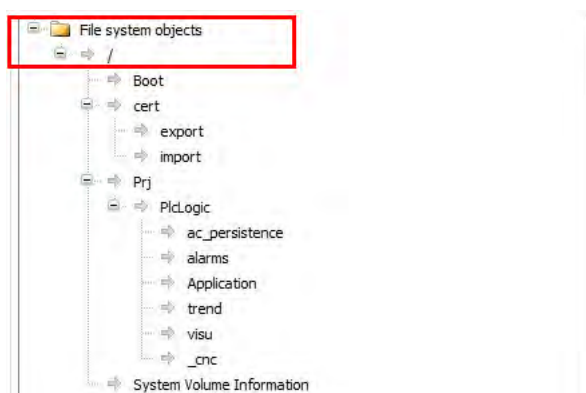
这控制 X.509 认证的在线使用。此处有两种访问权限类型：

- Read 读取 (View)
- Write 写入 (Modify)

每项操作都分配这两种权限之一。每项操作都作为子对象插入在 X.509 下方。因此，当前可以进一步微调每个操作的访问权限。

◆ **File system objects > /**

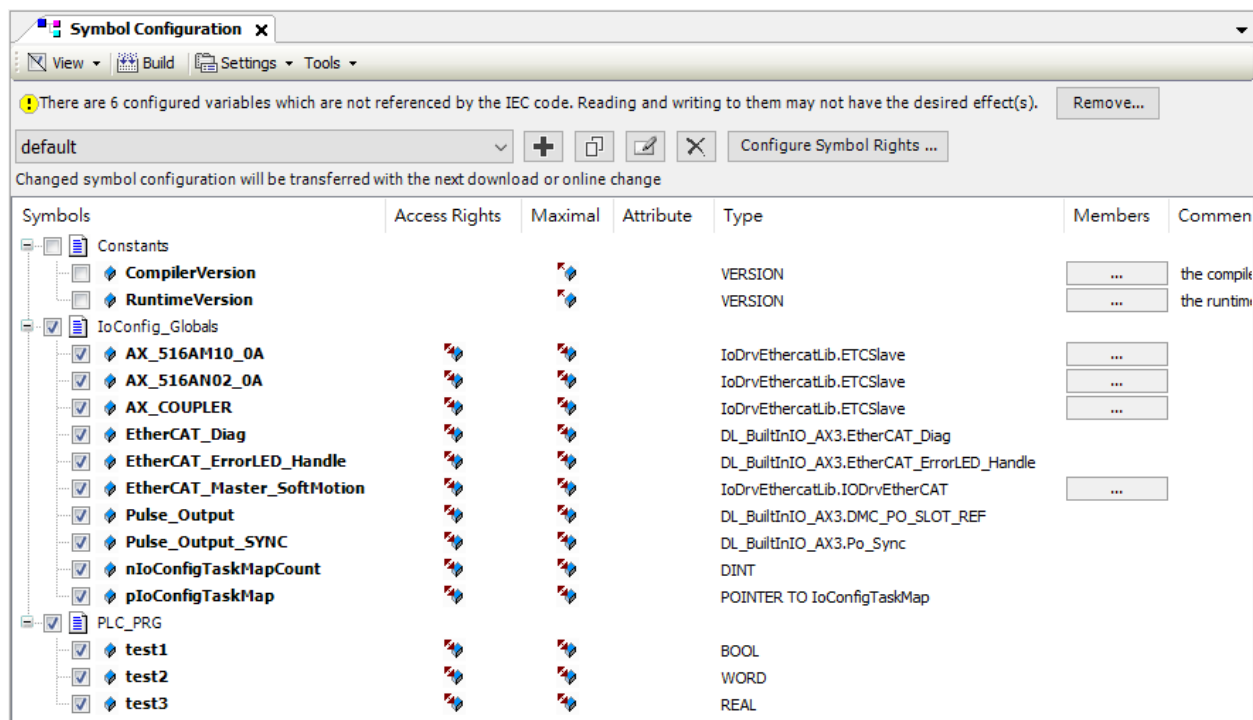
AX-3 系列控制器执行路径中的所有文件夹都插入在「/」文件系统对象下。使你可以授予文件系统的每个文件夹特定的权限。



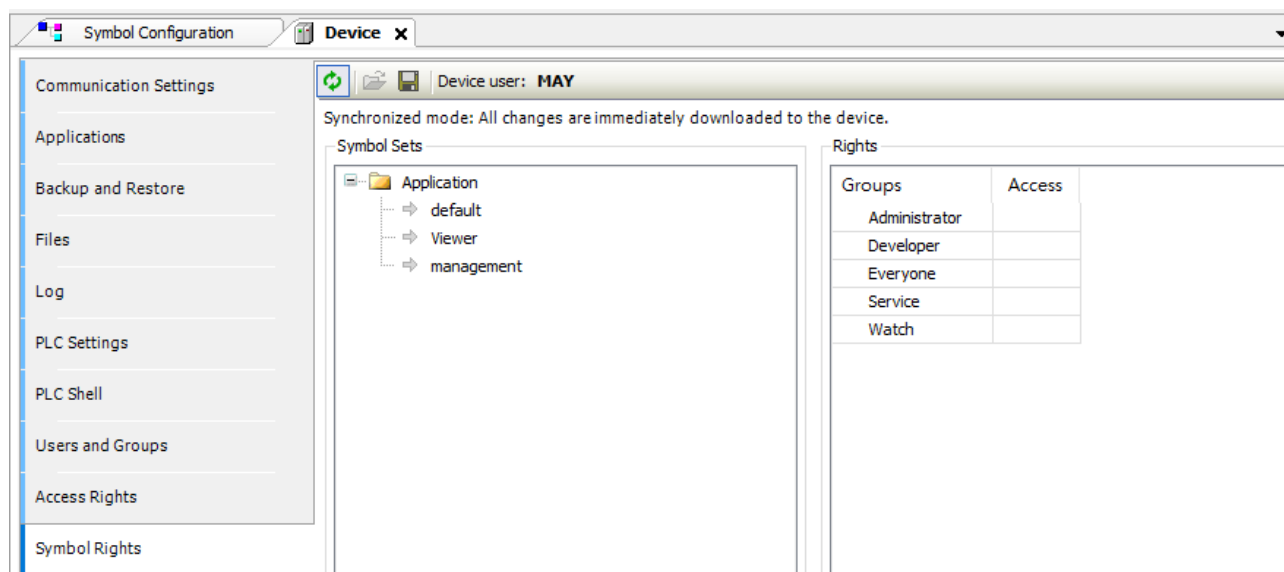
4.2.1.10 Symbol Rights

定义不同用户群组 (客户端) · 对 AX-3 系列控制器上可用的单个符号集的访问权限。

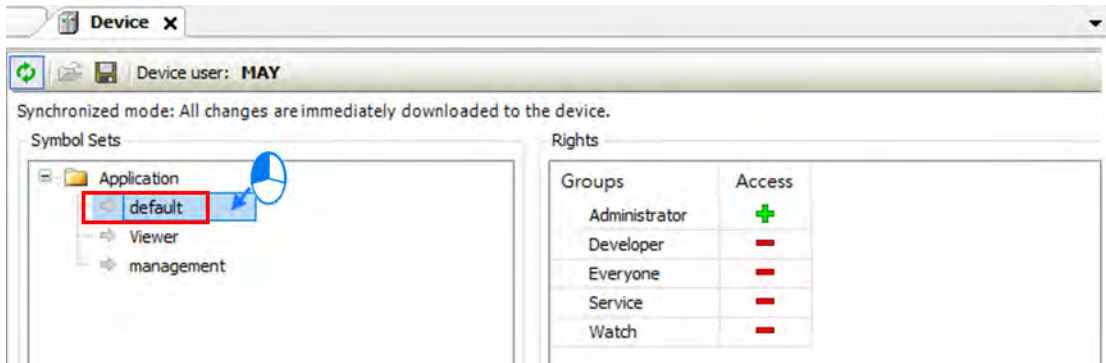
要求：必须在 PLC 上设置用户管理。已经将一个 Application 下载到 AX-3 系列控制器中，且 AX-3 系列控制器在 DIADesigner-AX Object 有定义了「Symbol sets」。用户群组具有在 AX-3 系列控制器中登入的数据访问权限。



在「Symbol Sets」窗口中，所有的符号集都列在 Application 节点下面，节点的定义与应用程序一起下载到 AX-3 系列控制器中。



在「Rights」窗口中，表里列出在 AX-3 系列控制器的用户管理中定义的用户群组。选择 Symbol Sets 后可以看到对应的权限。+：允许存取。-：禁止存取。你可以通过双击符号来变更权限。



单击 按钮可以将目前存取的设定保存到 XML 文件。其文件类型为 *Device symbol management files* (*.dsm)。单击 可从计算机里读取相同类型的文件。

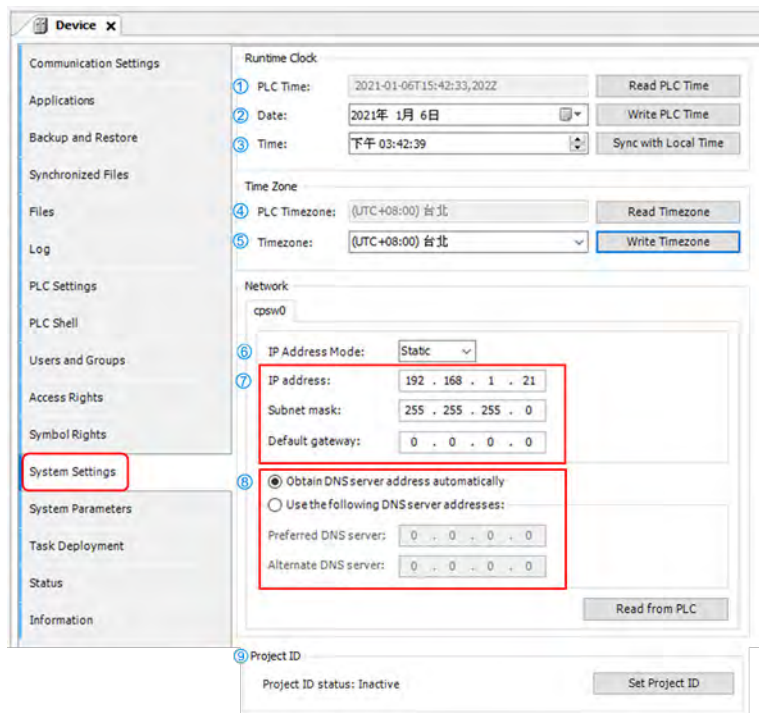
4

4.2.1.11 System Setting

在 AX-3 系列控制器中，系统相关设定可在「System Settings」页面进行设定，而设定前请先确认 DIADesigner-AX 与控制器已正常联机，联机详细设定请参考第 4.2.1.1 节当中的内容。

注：在 DIADesigner-AX V1.0.0 版中，此分页名称为「Runtime Clock Configuration」。在 DIADesigner-AX V1.1.0 版，由于新增 IP Address 设定功能，因此修改分页名称为「System Settings」。

在「Device」>「System Settings」



1. Runtime Clock

- ① **PLC Time** : PLC 目前的日期-时间信息。 **Read PLC Time** : 读取当前 PLC 中之日期-时间信息。
- ② **Date** : 软件接口上的日期信息。 **Write PLC Time** : 将设定之日期-时间, 写入 PLC 中。
- ③ **Time** : 软件接口上的时间信息。 **Sync with Local Time** : 与 Local PC 同步日期-时间信息。

2. Time Zone

- ④ **PLC Timezone** : PLC 目前的时区信息。 **Read Timezone** : 读取当前 PLC 中之日期-时间时区信息。
- ⑤ **Timezone** : PC 目前的时区信息。 **Write Timezone** : 将设定之时区, 写入 PLC 中。

3. Network

- ⑥ **IP Address Mode** : **Static**。静态 IP。
- ⑦ **IP 设定** : 由用户自行输入 IP Address/Subnet mask/Default gateway
- ⑧ **DNS 设定** : 用户可使用自动取得 DNS 服务器地址, 或使用自行指定的惯用 DNS 服务器地址或其它 DNS 服务器地址。

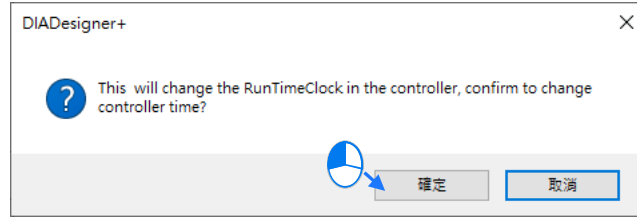
4. Project ID

- ⑨ **Set Project ID 设定** : 当启动 Project ID 功能后, 需在此设定 Project ID 后才可下载项目。

注 :

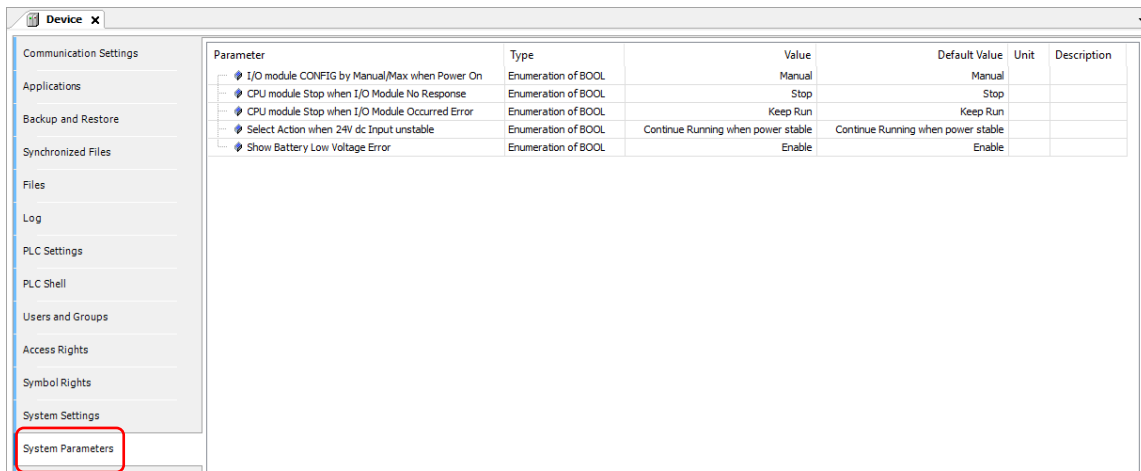
- 1. AX-3 系列主机描述文件 (DDF) 需于 V1.0.1.0 版本以上 (含), 才可在此页面进行 Network 参数设定。
- 2. AX-332 主机支持双网络接口, 网络设定接口参照下图。 **Enable Gateway and DNS Setting** 选项勾选时, 主机 Gateway 与 DNS 组态将以此页签 (如图为 GLAN1) 之值进行设定, 另一页签将无法设定此二组态。若未勾选 **Enable Gateway and Setting** 选项, 则 Gateway 与 DNS 字段将反灰无法输入。
- 3. Project ID 于韧体版本 1.0.4.0 版本以上 (含), 操作方式可参考 DIADesigner-AX 软件手册第 8.2 节。

The screenshot shows the 'Network' configuration window with the 'GLAN2' tab selected. At the top right, the checkbox 'Enable Gateway and DNS Setting' is checked. Below this, the 'IP Address Mode' is set to 'Static'. The 'IP address' field contains '192 . 168 . 0 . 10', the 'Subnet mask' contains '255 . 255 . 255 . 0', and the 'Default gateway' contains '0 . 0 . 0 . 0'. Underneath, there are two radio button options for DNS: 'Obtain DNS server address automatically' (unselected) and 'Use the following DNS server addresses:' (selected). The 'Preferred DNS server' and 'Alternate DNS server' fields both contain '0 . 0 . 0 . 0'. A 'Read from PLC' button is located at the bottom right of the configuration area.



4.2.1.12 System Parameters

「System Parameters」页面设定各项参数如下，可由下拉选单或在输入字段填入适当的数值，下载到 AX-3 系列控制器中，且此页面参数设定不支持在线监控时写入参数功能。



1. I/O module CONFIG by Manual/Max when Power On

设定 AX-3 系列控制器模块配置模式。

- **Manual** (默认值)：依据用户由 CONFIG 定义。设定配置与实际配置须完全符合，则可正常运行。
- **Max**：依据 CONFIG 配置模块数量若大于实际配置模块数，则可运行。若小于实际配置模块数量，则会显示警报。

2. CPU module Stop when I/O Module No Response

设定当连接的扩展模块其中有通讯断线无响应或无法控制的模块时，AX-3 系列控制器与其它正常的模块是否持续运行。

- **Stop** (默认值)：AX-3 系列控制器停止且回报错误。故障排除后，AX-3 系列控制器必须重新上电才能正常运作。
- **Keep RUN**：AX-3 系列控制器与其它正常模块继续运行。

3. CPU module Stop when I/O Module Occurred Error

设定当扩展模块出现轻微错误时的处理方式。

- **STOP**：AX-3 系列控制器停止且回报错误。错误排除后，AX-3 系列控制器能重新运行。
- **Keep RUN** (默认值)：AX-3 系列控制器继续运行，但会记录警告信息。

4. Select Action when 24Vdc Input unstable

可选择 24Vdc 电源发生跳电状况的处理机制。

- Continue Running when power stable(默认值):当 24Vdc 供应电源发生不稳定状况时,自动 STOP 等待电源恢复稳定之后,再持续运行 RUN。
- Into Error Status :当 24Vdc 供应电源发生不稳定状况时,强制 PLC 进入 STOP 状态,并且 ERROR 灯快闪,电源恢复稳定之后,仍然保持在 STOP 状态。

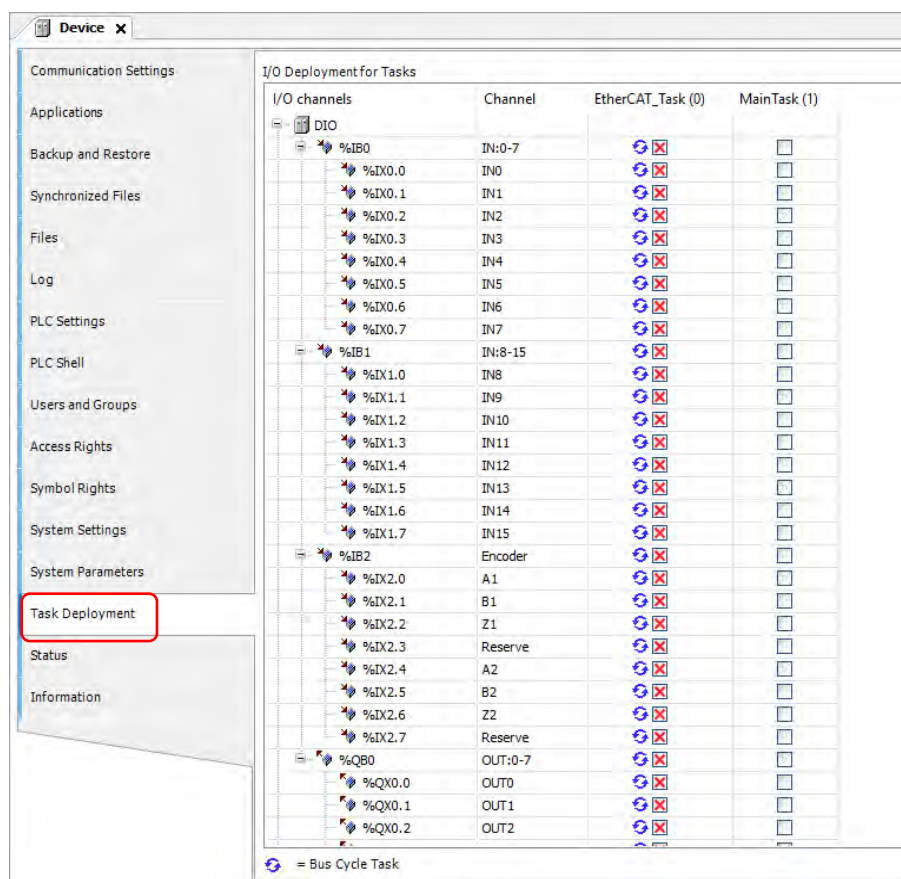
5. Show Battery Low Voltage Error

当记录时间的锂电池处于低电压状态或未安装时,可以设定是否显示警报。

- Disable :锂电池处于低电压状态或未安装时,不显示警报。
- Enable (默认值):锂电池处于低电压状态或未安装时,显示警报。

4.2.1.13 Task Deployment

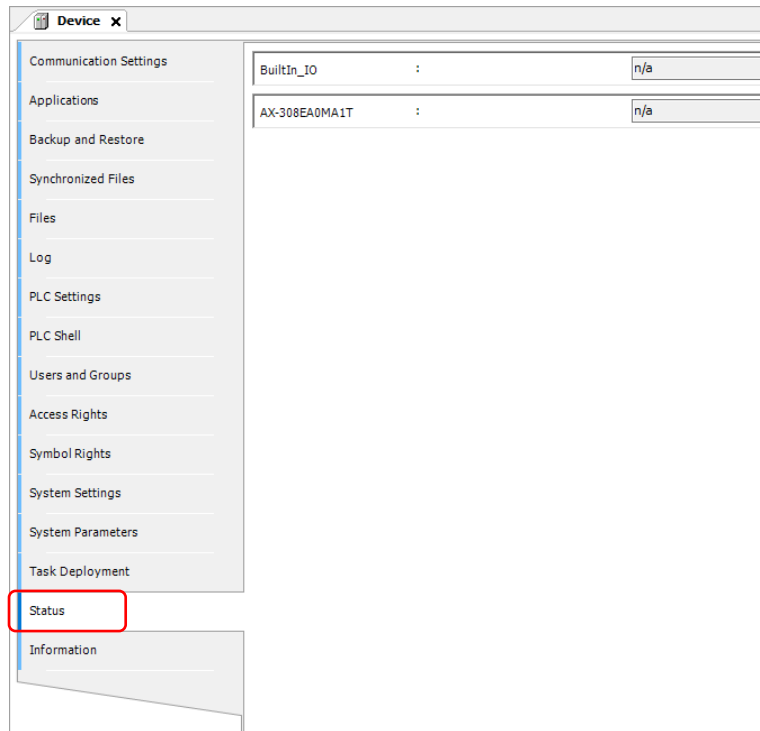
用户可以在此页面查询输入与输出点,所被指派的 task 与 Bus cycle task,当执行结果不如预期时,用户可以通过这些信息进行故障排除,此页面信息仅在项目编译过后,并下载至控制器后才会进行更新。



	此 task 为 Bus cycle task
	输入或输出点与此 task 被写入或读取

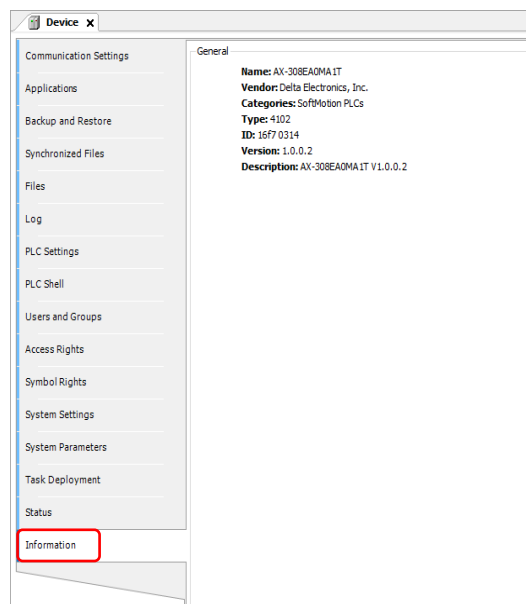
4.2.1.14 Status

用户可在此页面查询控制器内部运行状态，如 **Running**、**Stopped** 与其它特定说明，帮助用户了解控制器当前运作情形。



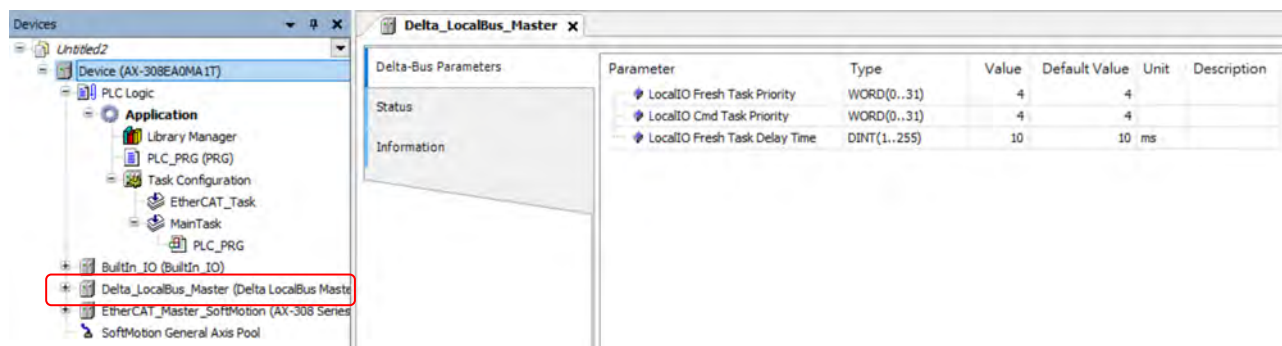
4.2.1.15 Information

此页面显示当前控制器相关信息，包含了机种、制造商、产品相关手册、韧体版本、产品代码、控制器 runtime 版本与控制器韧体版本等信息...



4.2.2 扩展模块参数设定

在扩展模块参数设定页面，用户可以依照使用的需求，自行调整扩展模块相关 Task 优先权与更新的时间。



- **LocalIO Fresh Task Priority :**

此项目为主机与扩展模块之间数据交换 Task 的优先权，其可设定范围为 0~31。(注 1)

- **LocalIO Cmd Task Priority :**

此项目为主机扩展模块命令运行 Task 的优先权，其包含模块运行模式 (Run / Stop)、扩展模块参数读写、From/To 指令等...，其可设定范围为 0~31。(注 1)

- **LocalIO Fresh Task Delay Time :**

此项目为调整扩展模块实际输入/输出的周期时间，用户可以根据所配置的扩展模块类型，推估所需运行时间并进行设定，下表为各模块数据交换所需时间。(其运作流程可参考第 4.4.1.2 节「多工作执行」)

Module	Time (ms)	Module	Time (ms)
AS02HC	(注 2)	AS08AM	0.6
AS02PU	1.5	AS08AN	0.6
AS04PU	2.1	AS06RTD	0.9
AS02LC	0.6	AS16AP11R	1.2
AS02ADH	0.6	AS16AN01T	0.6
AS04AD	0.9	AS16AM10N	0.6
AS04DA	0.9	AS06XA	1.5
AS08AD	1.2	AS32AM	0.6
AS04RTD	0.9	AS32AN	0.6
AS04TC	0.9	AS64AM	0.9
AS08TC	1.2	AS64AN	0.9

注 1：任意调整工作优先权，可能造成模块或是 EtherCAT 等其它周边功能异常。

注 2：此模块数据交换所需的时间，是依据项目内所使用到的指令而有所不同。

4.3 数据类型与变量

4.3.1 数据类型

数据类型	最小值	最大值	数据长度
BOOL	FALSE	TRUE	1-bit
BYTE	0	255	8-bit
WORD	0	65535	16-bit
DWORD	0	4294967295	32-bit
LWORD	0	$2^{64}-1$	64-bit
SINT	-128	127	8-bit
USINT	0	255	8-bit
INT	-32768	32767	16-bit
UINT	0	65535	16-bit
DINT	-2147483648	2147483647	32-bit
UDINT	0	4294967295	32-bit
LINT	-2^{63}	$2^{63}-1$	64-bit
ULINT	0	$2^{64}-1$	64-bit
REAL	-3.402823E+38	3.402823E+38	32-bit
LREAL	-1.7976931348623157E+308	1.7976931348623157E+308	64-bit
TIME	T#0ms	T#49d17h2m47s295ms	32-bit
LTIME	LTIME#0ns	LTIME#213503d23h34n33s 709ms551us615ns	64-bit
TIME_OF_DAY (TOD)	TOD#00 : 00 : 00.000	TOD#23 : 59 : 59.999	32-bit
DATE	D#1970-1-1 (01/01/70)	DATE#2106-2-7 (February 07 · 2106)	32-bit
DATE_AND_TIME	DT#1979-1-1-00 : 00 : 00 (01/01/1970 00 : 00 : 00)	DT#2106-2-7-6 : 28 : 15 (February 07 · 2106 6 : 28 : 15)	32-bit
STRING	ASCII 字符 (8-bit) · 最多 255 个字符。		
WSTRING	Unicode 字符 (16-bit) · 长度无限制。		

4.3.2 变数

变量命名规则：

- 不可包含空白或是关键词。
- 不区分大小写。(例如：Var0 与 VAR0 视为同一变数)
- 不可以连续使用下底线。(例如：b_Var0 允许使用，b__Var0 不允许使用)

变量重复命名规则

- 局部变量不可重复命名。
- 如全局变量与局部变量重复命名，则局部变量拥有较高的优先权
- 在不同的全局变量表，可以命名相同名称的变量。
(例如 globe_list1.bvar 与 globe_list2.bvar 为两个不相同的全局变量)

备注

- 单行备注「//」。(例如：// Variable Define)
- 多行备注是由「*」开始，至「*」结束。(例如：(*Variable Define*))

4.3.2.1 变量宣告

在 DIADesigner-AX 软件中变量的基本变量的宣告方式如下：

语法：<变量名> : <变量类型> := <初始值> ;

范例：

```
VAR
  bVar      :   BOOL ;
  byVar     :   BYTE := 1 ;
  wVar      :   WORD := 16#0001 ;
  todVar    :   TOD  := TOD#02:30:15.100;
END_VAR
```

数组

语法：<变量名> : ARRAY[0..N] OF <数据类型>

```
VAR
  byVar_Array :   ARRAY[0..10] OF BYTE ;
  wVar_Array  :   ARRAY[0..30] OF WORD ;
  rVar_Array  :   ARRAY[0..50] OF REAL ;
END_VAR
```


4.3.2.2 寻址

在 AX-3 系列控制器内存存储器可以分成三个区间，分别为 I (输入内存区间)、Q (输出内存区间) 与 M (标志内存区间) 三种，在编写程序时可以直接对内存寻址使用。

语法：%<内存区间><内存长度><内存地址>。

内存区间	说明	区间范围
I	输入内存	8 KB
Q	输出内存	8 KB
M	标志内存	512 KB

内存长度符号	数据格式	数据长度
X	--	1-bit
B	Byte	8-bit
W	Word	16-bit
D	DWord	32-bit
L	LWord	64-bit

注 1：AX-3 系列主机内标志内存 (M)，于在线模式时无法手动进行 Bit 操作，需于 V1.0.3.0 以上 (包含) 固件版本，才可支持 %MX 装置。

- 内存位置

在进行内存地址的寻址之前，必须要先了解映像的内存位置，来避免重复定义导致相互覆盖的情形发生。

内存区间							
X0.63~X0.56	X0.55~X0.48	X0.47~X0.40	X0.39~X0.32	X0.31~X0.17	X0.23~X0.16	X0.15~X0.8	X0.7~X0.0
X7.7~X7.0	X6.7~X6.0	X5.7~X5.0	X4.7~X4.0	X3.7~X3.0	X2.7~X2.0	X1.7~X1.0	X0.7~X0.0
B7	B6	B5	B4	B3	B2	B1	B0
W3		W2		W1		W0	
D1				D0			
L0							

- 使用范例

范例	说明
%QX7.5	输出内存区间 Bite X7.5
%IW215	输入内存区间 Word W215
%QB7	输出内存区间 Byte B7
%MD48	标志内存区间 Double Word DB48

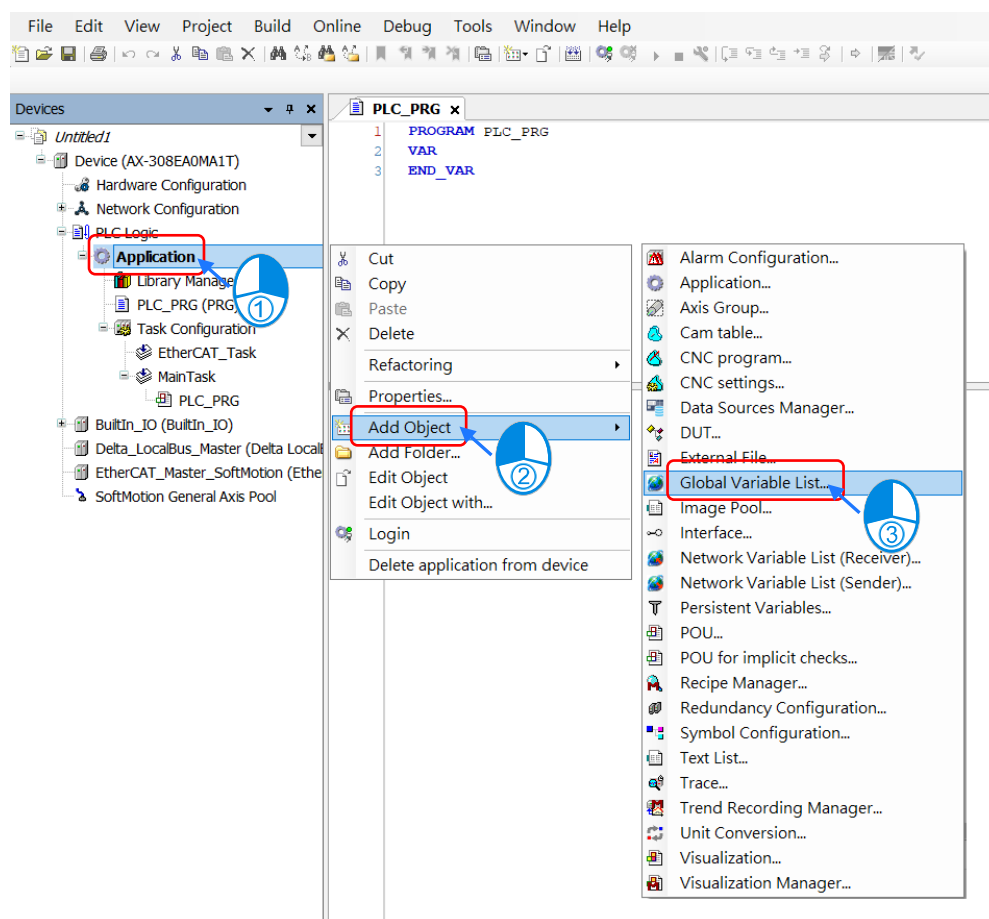
范例	说明
VAR wVar0 AT %IW0 : WORD ; END_VAR	将变量 wVar0 寻址至输入内存区间 Word W0
VAR bVar0 AT IX7.5 : BOOL ; END_VAR	将变量 bVar0 寻址至输入内存区间 Bit X7.5

4.3.2.3 变量类型

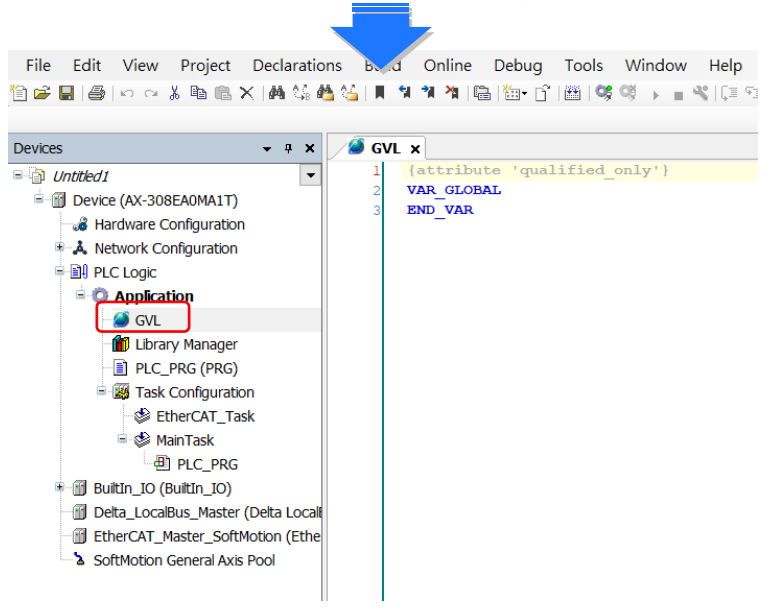
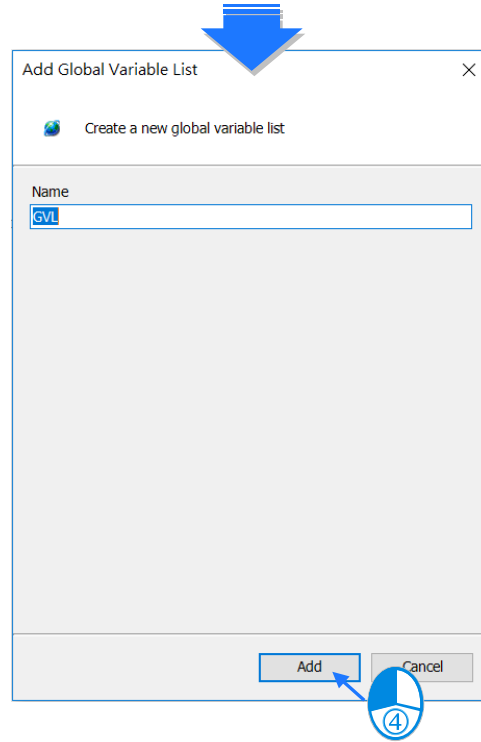
● 全局变量

在各个 POU 内所宣告的变量为局部变量，只能在当前 POU 内使用，而 Global Variable List 内所宣告的变量（全局变量），则可以被所有 POU 使用。

Global Variable List 新增：



4



● 常数型变数

如欲使用常数型变量，仅需将变量宣告成常数属性即可，而常数型变量只允许读取，不允许赋值。

常数型变量宣告

```

VAR CONSTANT
    pi : REAL := 3.14159 ;
END_VAR
    
```

● 停电保持型变量

如欲使用停电保持型变量，则需将变量宣告成 Retain 属性，或是直接使用 Persistent 变量，而一般变量、Retain 变量与 Persistent 变量三种变量的差异性可以参考下列表格。

	Initialize				
	Reboot PLC	Reset warm	Reset cold	Download	Reset Origin
Variable	O	O	O	O	O
Variable Retain	X	X	O	O	O
Variable Retain Persistent	X	X	X	X	O

Retain 变量宣告

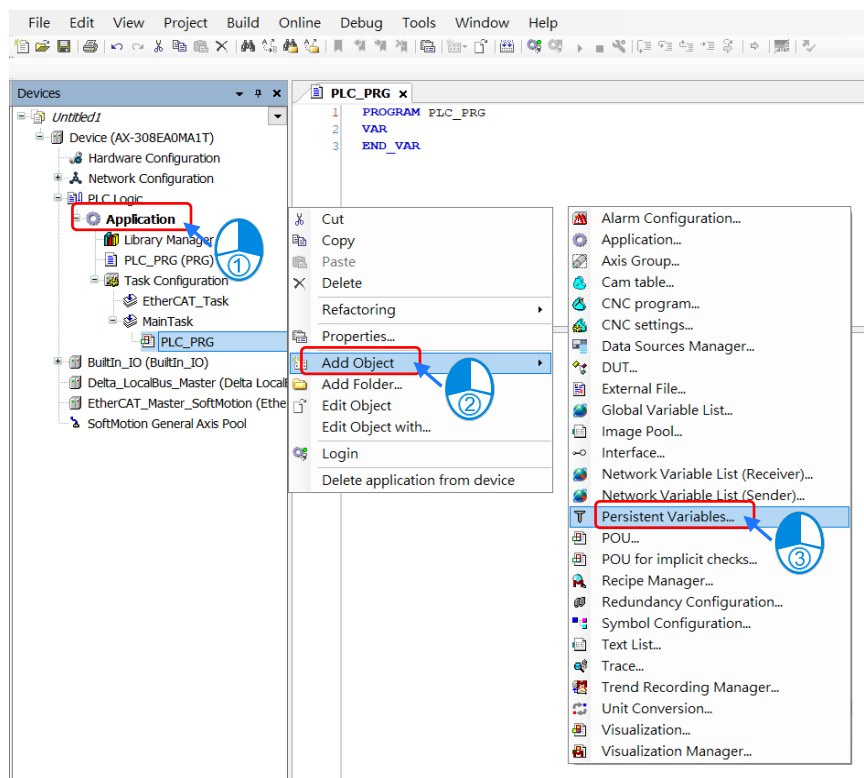
```

VAR RETAIN
  bVar : BOOL ;
  byVar : BYTE ;
  wVar : WORD ;
END_VAR

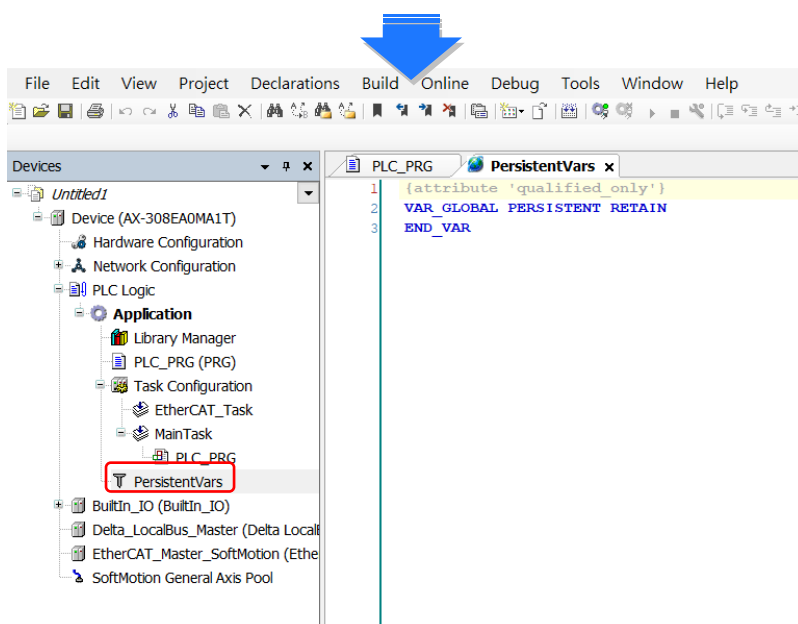
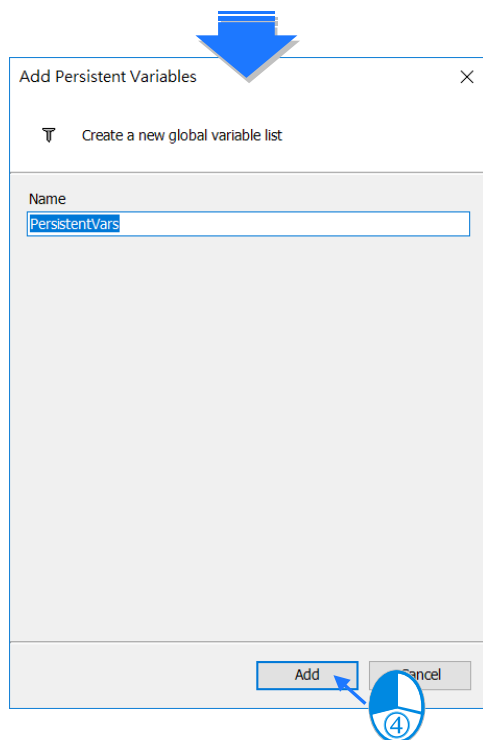
```

如要建立 Persistent 变量，则必须在 Persistent Variable 中进行宣告，且 RETAIN PERSISTENT、PERSISTENT RETAIN 与 PERSISTENT 这三种变量宣告作用皆相同。

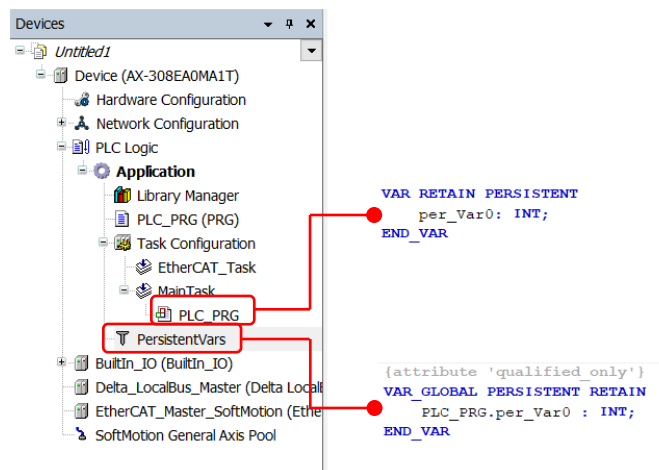
Persistent Variable List 新增：



4



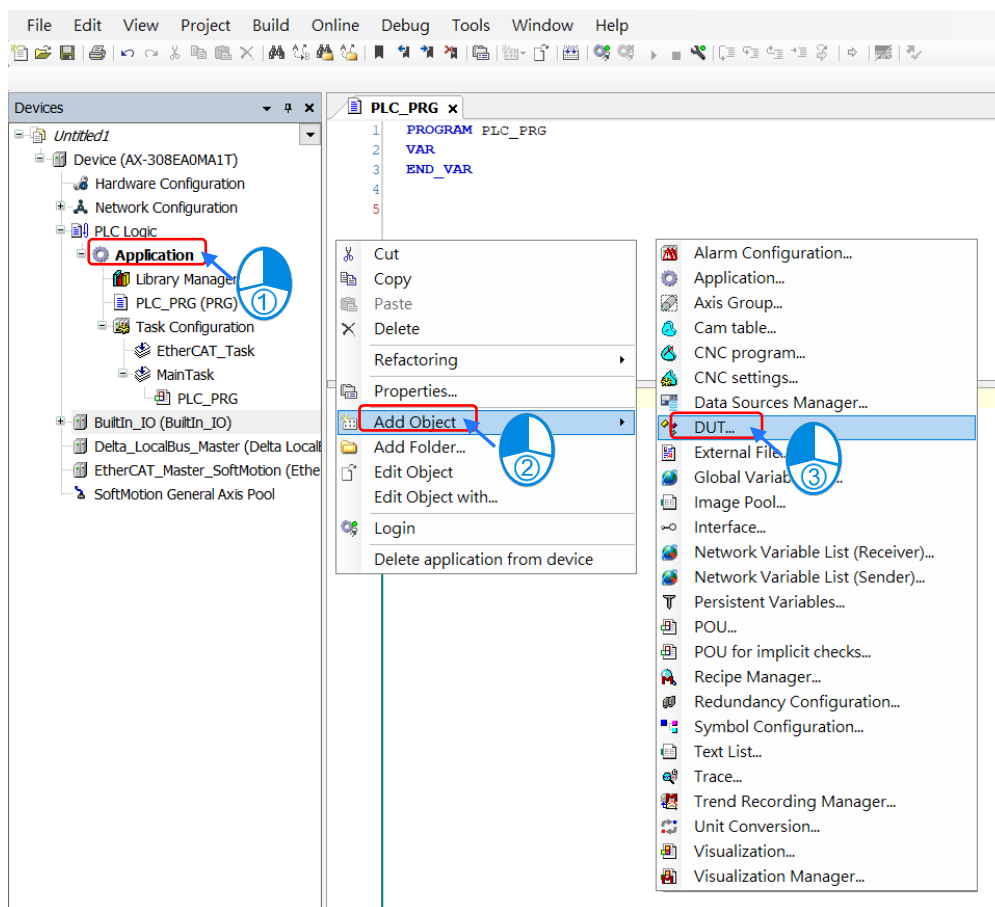
如需将局部变量宣告成 Persistent 变量属性，则必须要将变量的路径新增至 Persistent Variable 对象内。



4.3.2.4 自定义数据类型

自定义数据类型（称 DUT（Data Type Unit）或 UDT（User-defined Data Type）），供用户组合现有的数据类型成自定义的数据类型，其使用的方式为，在树形图新增 DUT 对象并选择类型类型，其类型可以分成 Structure、Enumeration、Alias 与 Union 四种。

DUT 新增



● **Structure**

结构 (Structure) 由多种数据类型变量所组成的数据结构，可利用这样的数据结构方便用户进行数据的管理与修改。

语法：

TYPE <结构名>：

STRUCT

<变量宣告 1>

...

<变量宣告 n>

END_STRUCT

END_TYPE

范例：

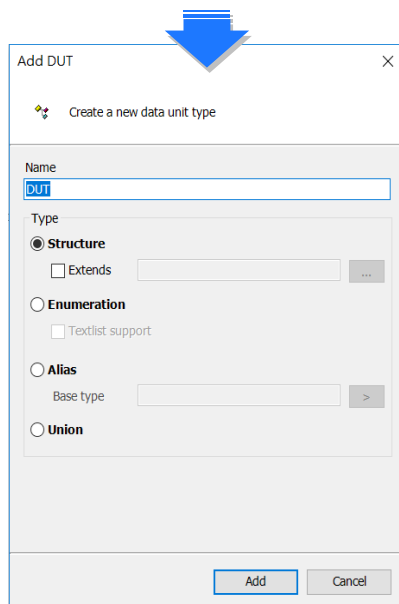
```

TYPE DUT :
STRUCT
    bVar    :   BOOL    ;
    wVar    :   WORD    ;
    iVar_Array :   ARRAY[0..2]OF INT  ;
END_STRUCT
END_TYPE
    
```

使用方式：

```

PLC_PRG x
1 PROGRAM PLC_PRG
2
3 VAR
4     byVar2 AT %QX7.5 :   BOOL  ;
5     DUT_Var :DUT      := (bVar:=TRUE,wVar:=12,iVar_Array:=[1,2,3]);
6 END_VAR
7
8
9
10
11 DUT_Var.bVar:=FALSE;
12 DUT_Var.iVar_Array[1]:=123;
    
```



● Enumeration

枚举 (Enumeration) 用于定义变量与相对应的数值，这样的方式连结了数值与其代表的意义，使程序更清晰易读。

语法：

```
TYPE <枚举名> :
(
    <第一个组件宣告> := 组件代表数值，
    ...，
    <最后一个组件宣告> := 组件代表数值
) <基本数据类型> := 枚举默认值；
END_TYPE
```

范例：

```
TYPE Enumeration_0 :
(
    GREEN := 0,
    YELLOW:=3,
    RED:=8
) INT:=YELLOW;
END_TYPE
```

● Alias

可以在基本的数据类型上，用户可以自行定义数据类型。

范例：

```
TYPE <Alias 名> : STRING ( 20 ) ; END_TYPE
```

● Union

Union 内所定义的变量将会共享相同的储存地址。

语法：

```
TYPE <Union 名> :
UNION
    <变量宣告 1>
    ...
    <变量宣告 n>
END_UNION
END_TYPE
```

范例：

```
TYPE DUT_Union :
UNION
    unVar0:WORD;
    unVar1:DWORD;
END_UNION
END_TYPE
```


4.3.2.5 变量清除为零时机说明

针对不同变量类型，在程序清除、下载与在线编辑其数值变化其下表说明

监控后执行动作	VAR	VAR Retain	VAR Retain Persistent
Online Change	●	●	●
Reboot PLC	○	●	●
Reset Warm	○	●	●
Reset Cold	○	○	●
Download	○	○	●
Reset Origin	○	○	○

● = 数值保持

○ = 清除为零

*注：当无数值保持功能，初始值数值为有效。

4.3.2.6 变数初始值生效时机

变数初始值生效

监控后执行动作	VAR	VAR Retain	VAR Retain Persistent
Online Change	●	●	●
Reboot PLC	○	●	●
Reset Warm	○	●	●
Reset Cold	○	○	●
Download	○	○	●

● = 无效

○ = 初始值生效

4.4 控制器原理

4.4.1 工作 (Task)

工作 (Task) 在控制器中是用来控制与执行程序组织单元 (POU)，用户可以自行设定 Task 名称，优先权和触发条件，Task 可以使用时间 (cyclic-interval, freewheeling) 或者利用内部与外部的 Event 作为执行触发条件。

每一个 Task 内可以包含一个或多个 POU，而 Task 执行的先后顺序，则是依照所设定的工作条件与其优先权所决定，也可以为每一个 Task 设定其 Watchdog 时间。

工作 (Task) 执行顺序条件：

- Task 所设定的触发条件成立
- 如多个 Task 的触发条件同时成立，则依照所设定的优先权的高低来，高优先权的 Task 优先执行。
- 如多个 Task 同时成立且优先权皆相同时，则触发条件最先成立的 Task 优先执行。
- 如项目内与使用的函式库 (Library) 内有相同命名的 POU，则执行项目内的 POU。

注：Task 优先权设定值越低，代表优先权越高，可分别设定为 0 ~ 31。

控制器工作类型可分成五种：

1. Cyclic Task
2. Event Task
3. External Task
4. Freewheeling Task
5. Status Task

4.4.1.1 工作 (Task) 类型

各类型任务说明如下

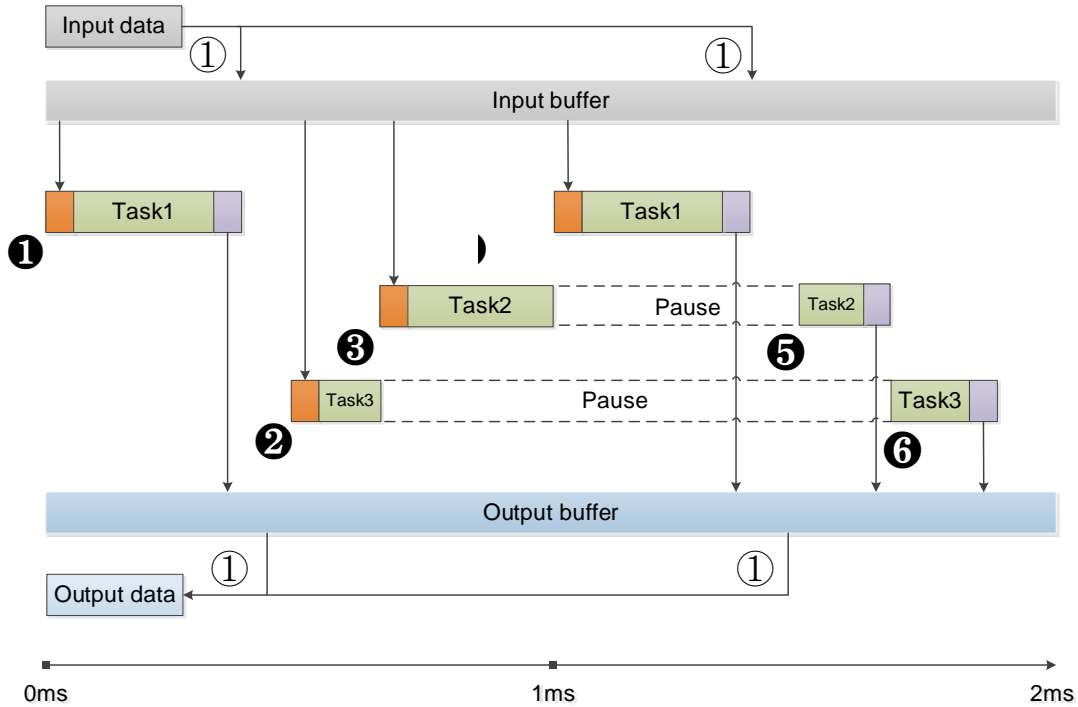
- **Cyclic Task**
周期性执行 Task，其执行周期是依照 Task 内所设定的 Interval 参数时间所决定。
- **Event Task^{*1}**
当 Event Task 内所定义的 Event 触发变量为上升沿时，则执行此 Event Task。
- **External Task**
当所设定的主机内建 Input 点为上升沿时，则执行此 Task 一次。
- **Freewheeling Task**
当控制器开始运行时，则 Freewheeling Task 即开始执行，而且控制器会连续的重复执行此 Task，直到控制器停止运行为止。
- **Status Task**
当 Status Task 内所定义的 Event 触发变量为「ON」时，则执行此 Status Task。

注：Event 触发变量必须为布尔变量。

4.4.1.2 多工作 (Task) 执行

各任务之间执行的优先级，是依照各任务所设定优先权的高低，优先权顺序可以设定 0 到 31 (0 表示最高优先级，31 代表最低优先级)。

多工作执行优先级



Task 1：优先权设定 = 1 · Bus cycle Task · Cyclic Task。

Task 2：优先权设定 = 3 · Event Task。

Task 3：优先权设定 = 5 · Freewheeling Task。

- ①：Task 1 执行条件成立，开始执行 Task 1。
- ②：Task 1 执行完毕，此时会将 Input/output buffer 数据更新至 IO 通道，Task 3 执行条件成立，开始执行 Task 3。
- ③：Task 2 执行条件成立，而且 Task 2 优先权较 Task 3 高，因此 Task 3 暂停执行，Task 2 开始执行。
- ④：Task 1 执行条件成立，而且 Task 1 优先权较 Task 2 高，因此 Task 2 暂停执行，Task 1 开始执行。
- ⑤：Task 1 执行完毕，此时会将 Input/output buffer 数据更新至 IO 通道，Task 2 重新开始执行。
- ⑥：Task 2 执行完毕，Task 3 重新开始执行。
- ① 扩展模块：主机所配置的扩展模块，则依照所设定的 Fresh time 周期时间，将 Input data 数据 (实际输入数据) 更新至 Input buffer，并将 Output buffer 数据更新至 Output data (实际输出)。

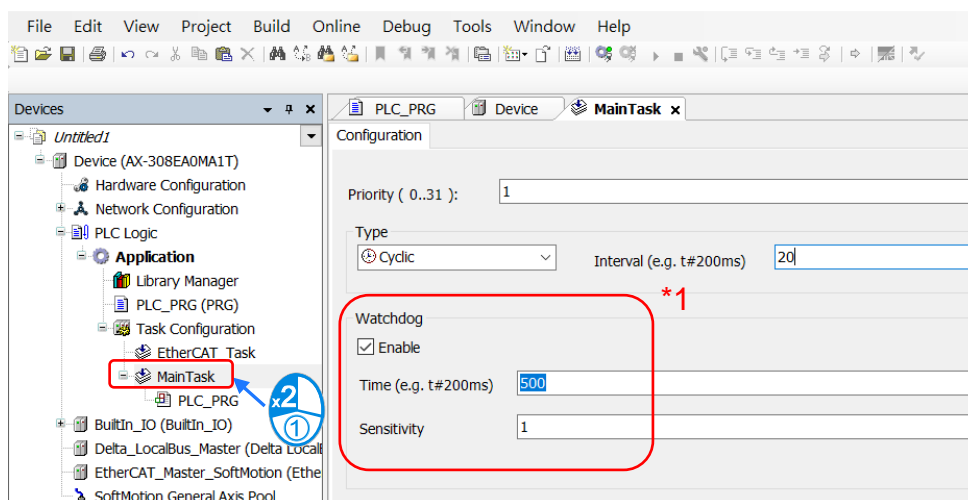
Component：各 Component (如 ethernet、EtherCAT...) 则依照其设定的 Bus cycle Task，待 Bus cycle Task 执行完毕后，即将 Input data 数据更新至 Input buffer，并将 Output buffer 数据更新至 Output data。

注：扩展模块 Fresh time 请参考第 4.2.2 节「扩展模块参数设定」

4.4.1.3 Watchdog

当 Task 设定内 Watchdog 生效，则当 Task 执行超过 Watchdog 所设定的时间时，则控制器会产生错误记录并停止执行。

Watchdog 启动方式：



注 1：Task 的 Watchdog 时间计算方式如下

Sensitivity 为 0 时，Watchdog 时间为 Time * 1。

Sensitivity 为 n 时，Watchdog 时间为 Time * n。

4.4.1.4 每种任务类型可以执行的运动控制指令

针对 AX 系列功能块，需使用在何种任务类型介绍如下：

- 同步轴

种类	指令名称	任务类型		
		Cyclic	Freewheeling	Bus Cycle EtherCAT
运 动 功 能 块	MC_Home			√
	MC_Stop			√
	MC_Halt			√
	MC_MoveAbsolute			√
	MC_MoveRelative			√
	MC_MoveAdditive			√
	MC_MoveSuperImposed			√
	MC_CamIn			√
	MC_CamOut			√
	MC_MoveVelocity			√
	MC_PositionProfile			√
	MC_VelocityProfile			√
	MC_AccelerationProfile			√
	MC_Jog			√
	MC_GearIn			√
MC_GearOut			√	

种类	指令名称	任务类型		
		Cyclic	Freewheeling	Bus Cycle EtherCAT
	MC_GearInPos			√
	MC_Phasing			√
	DMC_TorqueControl			√
	DMC_VelocityControl			√
	DMC_MoveLinearAbsolute			√
	DMC_MoveLinearRelative			√
	DMC_MoveCircularAbsolute			√
	DMC_MoveCircularRelative			√
	DMC_GroupStop			√
	DMC_GroupHalt			√
	DMC_Home_P			√
	DMC_GroupInterrupt			√
	DMC_GroupContinue			√
	DMC_ImmediateStop_P			√
管 理 指 令	MC_Power			√
	MC_SetPosition			√
	MC_ReadParameter			√
	MC_WriteParameter			√
	MC_ReadBoolParameter			√
	MC_WriteBoolParameter			√
	MC_ReadActualPosition			√
	MC_ReadActualVelocity			√
	MC_ReadActualTorque			√
	MC_Reset			√
	MC_ReadStatus			√
	MC_ReadAxisError			√
	MC_CamTableSelect			√
	MC_TouchProbe			√
	MC_AbortTrigger			√
	MC_DigitalCamSwitch			√
	DMC_GroupEnable			√
	管 理 指 令	DMC_GroupDisable		
DMC_GroupReadStatus				√
DMC_GroupReadError				√
DMC_GroupReset				√
DMC_CamReadTappetStatus				√
DMC_CamReadTappetValue				√
DMC_CamWriteTappetValue				√
DMC_CamAddTappet				√
DMC_CamDeleteTappet				√
DMC_CamReadPoint				√
DMC_CamWritePoint				√
DMC_ChangeMechanismGearRation				√
DMC_ReadMotionState				√
DMC_GroupReadParameter				√
DMC_GroupWriteParameter			√	

注：建议运动功能块放置在 Bus Cycle EtherCAT 中，以免导致运动不连续。

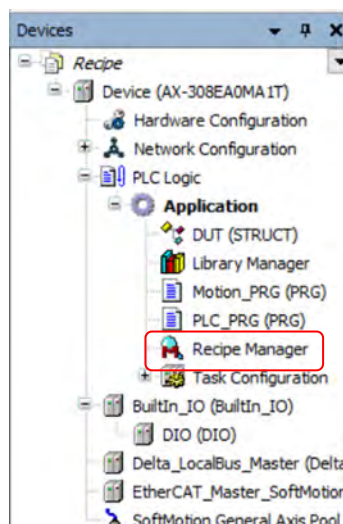
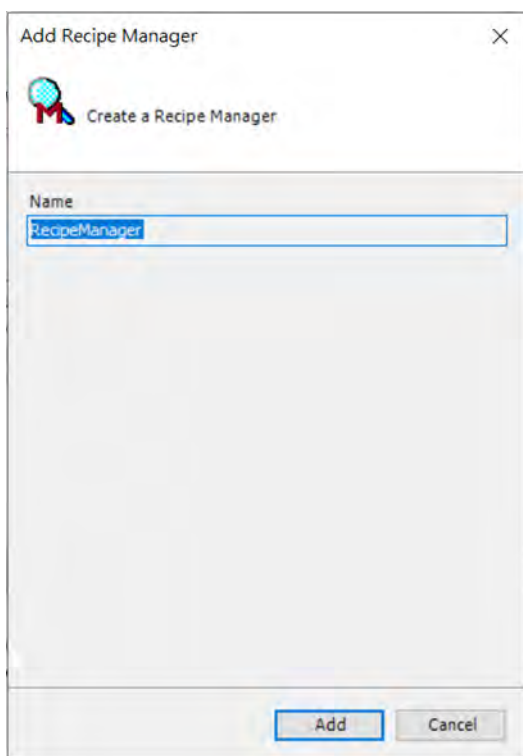
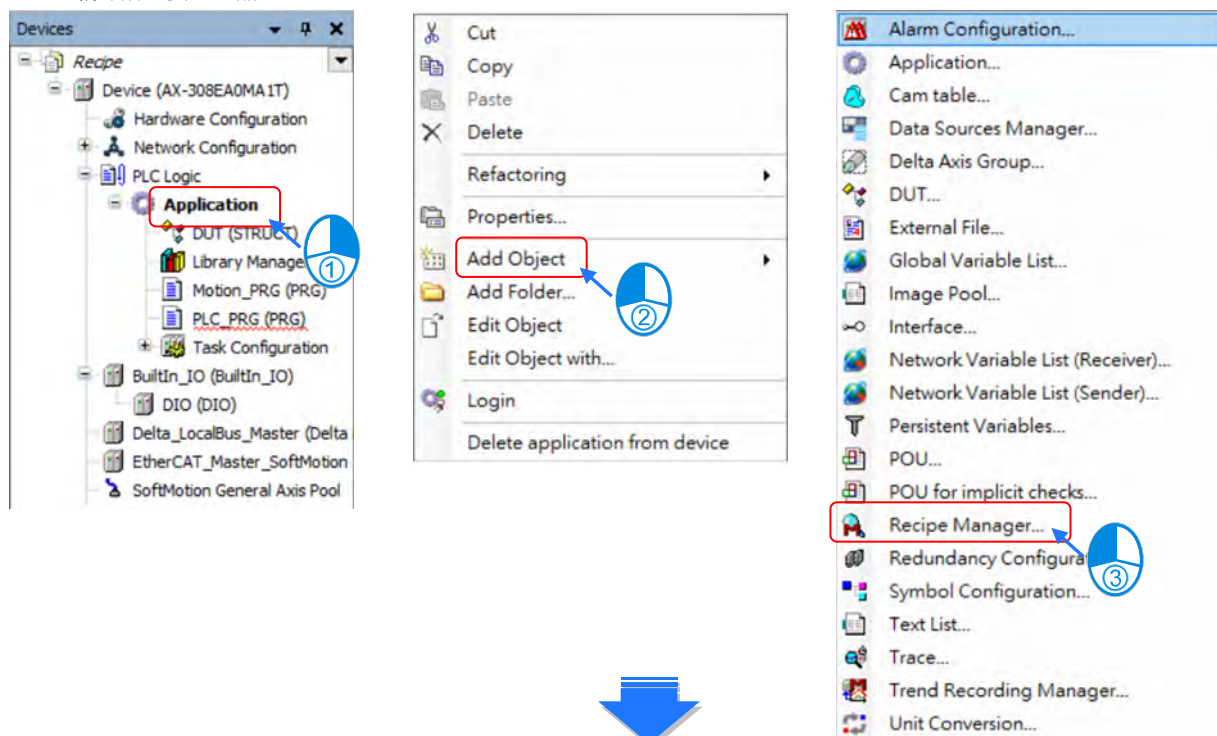
● 定位轴

种类	指令名称	任务类型		
		Cyclic	Freewheeling	Bus Cycle EtherCAT
运 动 功 能 块	MC_Halt_DML			V
	MC_Home_DML			V
	MC_MoveAbsolute_DML			V
	MC_MoveRelative_DML			V
	MC_MoveVelocity_DML			V
	MC_Stop_DML			V
管 理 指 令	MC_Power_DML			V
	MC_ReadBoolParameter_DML			V
	MC_ReadParameter_DML			V
	MC_ReadStatus_DML			V
	MC_Reset_DML			V
	MC_WriteBoolParameter_DML			V
	MC_WriteBoolParameter_DML			V
	MC_ChangeAxisConfig_DML			V
	MC_ReinitDrive_DML			V
	MC_SetOpmode_DML			V
MC_StartupDrive_DML			V	

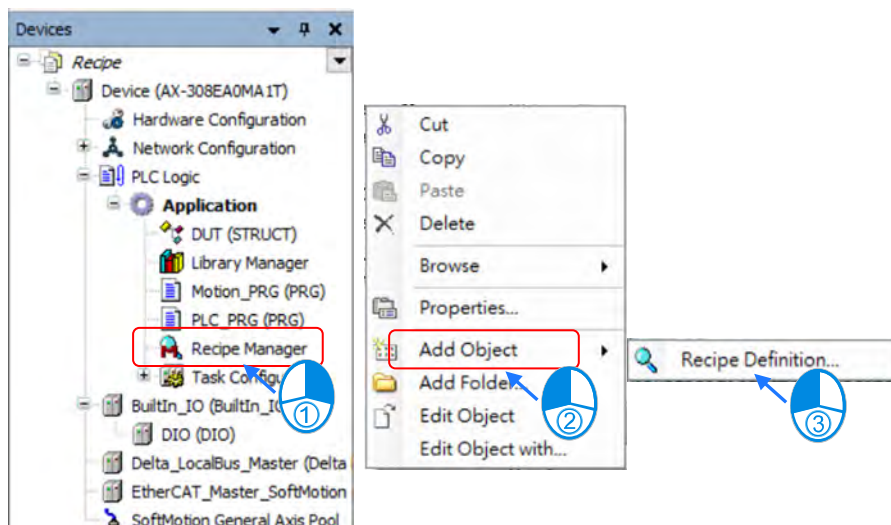
4.5 配方管理器

配方管理器可供用戶將配方檔案匯入至控制器的特定變數，或將控制器的特定變數匯出至配方檔案，用戶可使用「Recipe_Management.library」的功能塊內的「RecipeManCommands」的相對應方法進行操作。

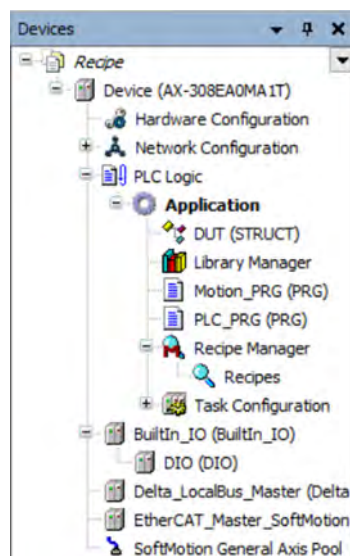
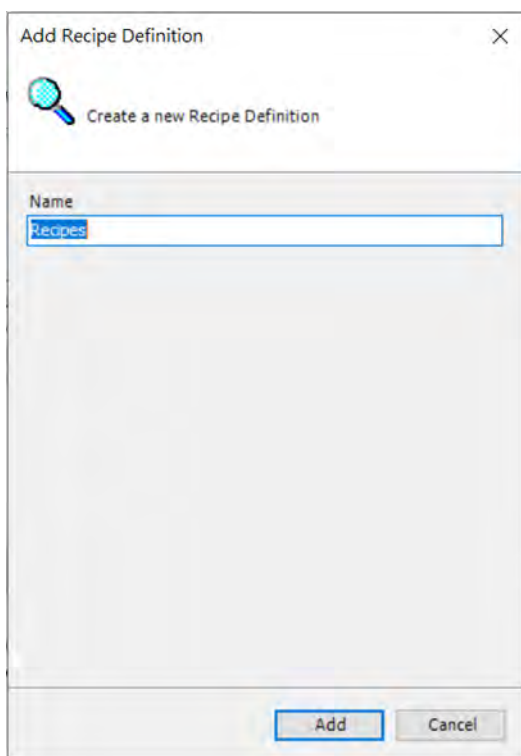
● 新增配方管理器



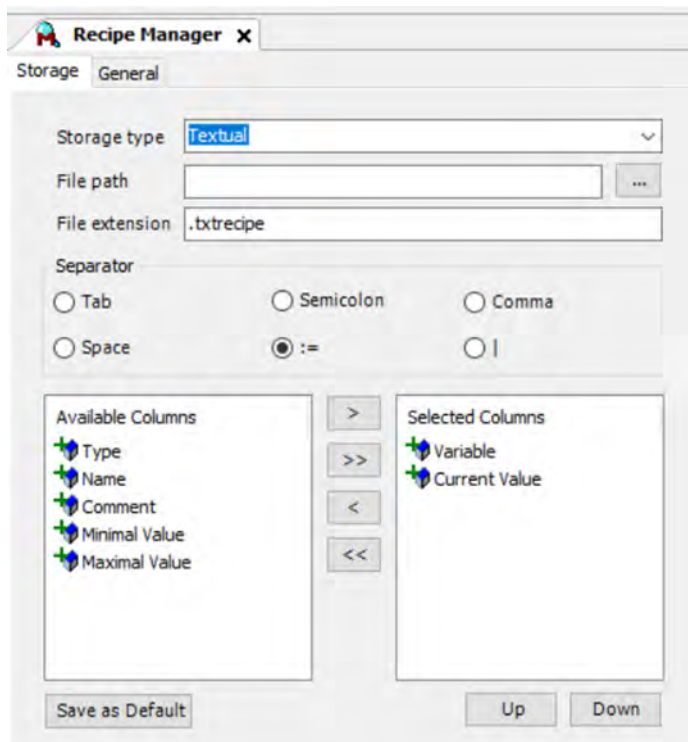
- 新增配方定義



4

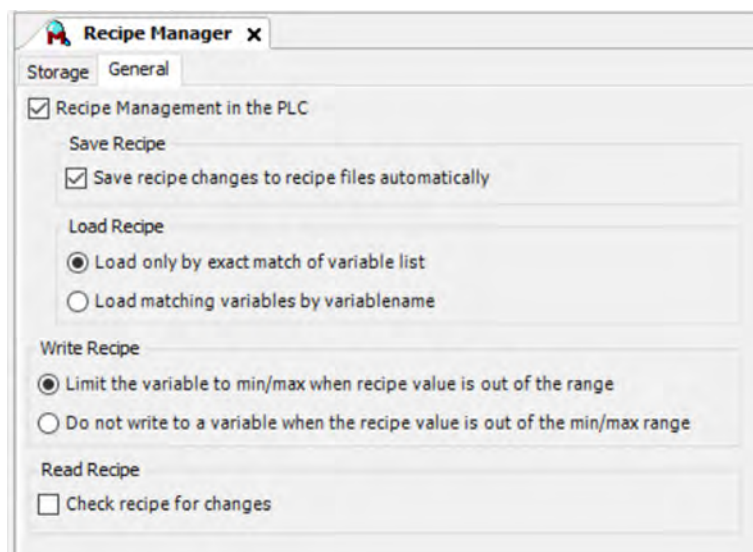


4.5.1 配方管理器



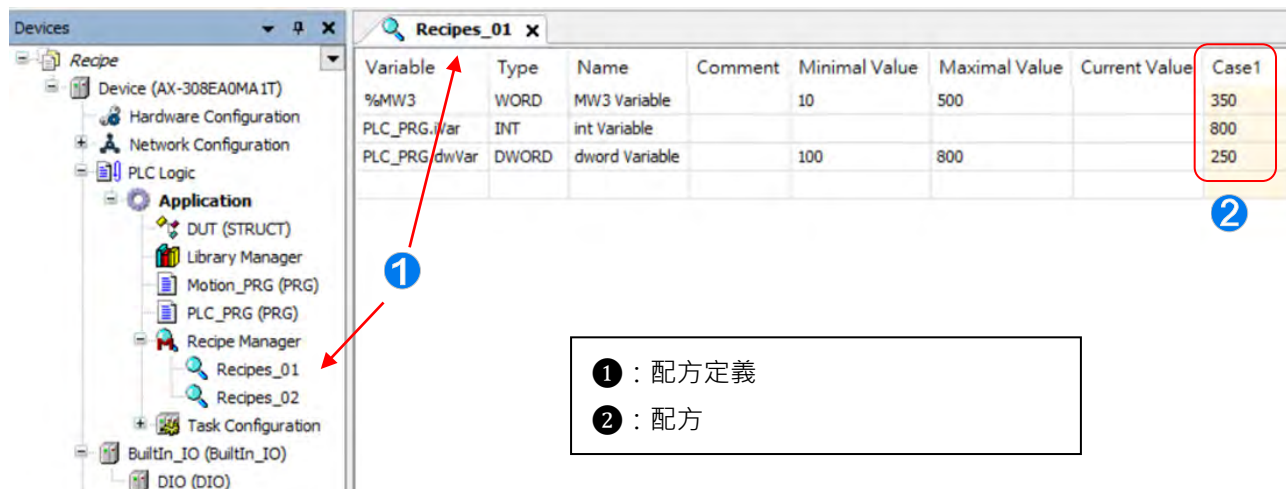
4

选单	说明
Storage type	配方文件单元格式，用户可以依照需求选择 Textual / Binary 格式。
File path	配方文件存储路径 Example：AllRecipes，则配方文件储存路径为 PlcLogic/AllRecipes。
File extension	配方文件扩展名.<file extension> 配方文件名命名格式 <recipe>.<recipe definition>.<file extension>.
Separator	于配方文件内，各数值之间间隔符号。
Available Columns Selected Columns	定义配方文件的内容与顺序。
Save as Default	将此设置应用于项目中其它所有的配方管理器。



选单	说明
Recipe management in the PLC	勾选此项目后即生效配方管理器
Save Recipe	
Save recipe changes to recipe files automatically	勾选此项目后当下载项目时，即自动更新配方文件，如配方有改变时，将自动储存至配方文件。
Load Recipe	
Load only by exact match variable list	勾选此项目后，当 Load 配方时，配方文件与所设定变量、变量顺序与 Name 字段内容值必须相同，否则无法从配方文件导入至控制器变量。（如配方文件于最末，多出其它变量信息，其它变量信息与控制器配方的变量顺序、Name 字段内容值与所设定的变量皆相同时，这些于配方文件最末的变量信息，则会自动忽略不进行导入）
Load matching variables by variablename	勾选此项目后，当 Load 配方时，控制器尝试将配方文件与控制器一致的变量导入，即便变量顺序或 Name 字段的内容值不相符。
Write Recipe	
Limit the variable to min/max when recipe value is out of the range	如配方值超出所设定的最大或是最小值范围时，则将最大或是最小值导入所相对应的变量。
Do not write to a variable when the recipe value is out of the min/max range	如配方值超出所设定的最大或是最小值范围时，则不将配方导入所相对应的变量。

4.5.2 配方定义

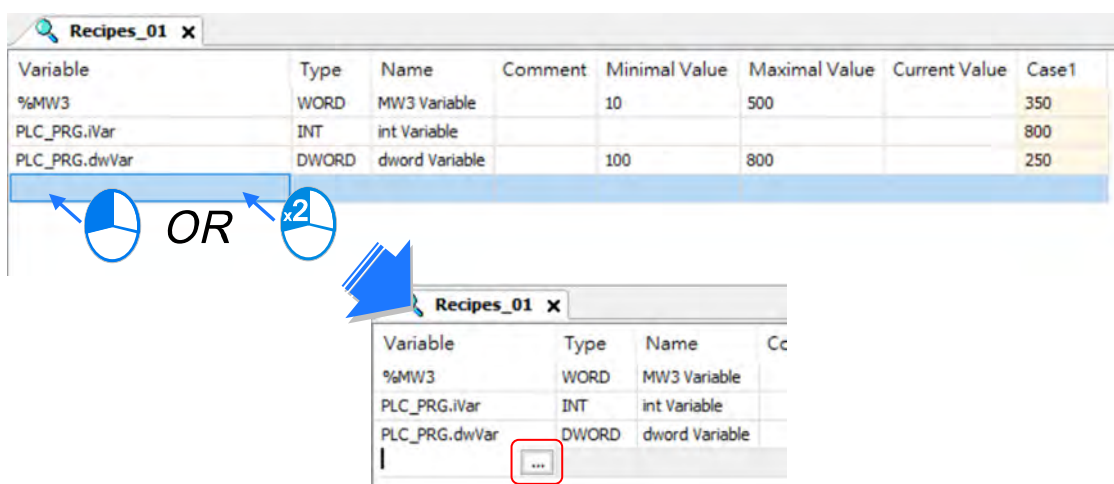


4

字段	说明
Variable	在表格内用户可以选择任意变量，包含了 POU 内所自定义的变量。
Type	此字段将会自动带入变量锁定的数据类型。
Name	用户可自行定义变量名称，供 Load Recipe 检查比对使用。
Comment	批注
Minimal Value Maximal Value	用户可以自行定义变量所允许导入的最大与最小值，当所导入的配方值超出范围时，控制器则会依照配方管理器，来决定是否要将配方值导入变量。
Current Value	于在线模式时，即于此显示此变量之当前值。

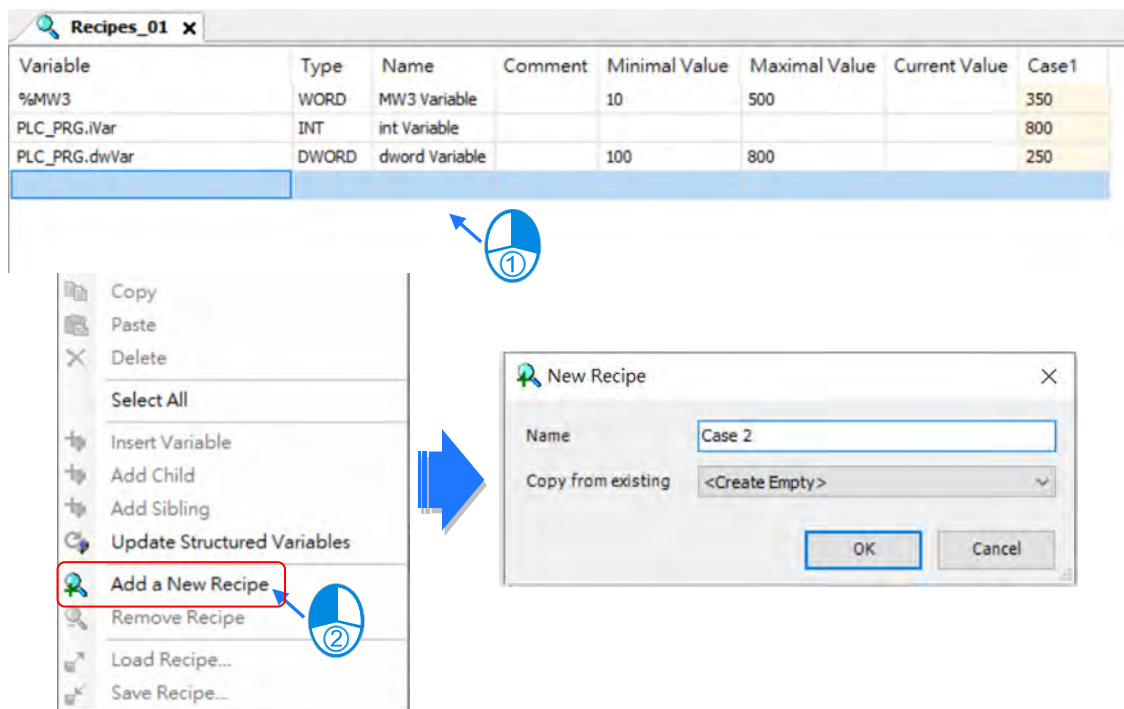
● 增加变量的方式

用户可以于空白的表格直接输入变量名，或是单击两下开启「Input Assistant」直接选择变量。

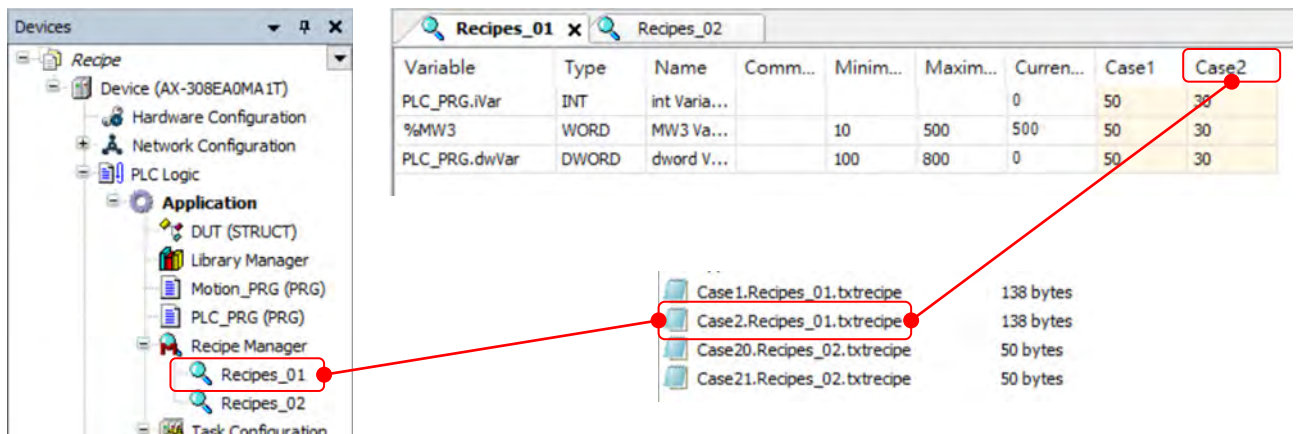


- 增加配方的方式

用户可以在页面中单击右键，单击「Add a New Recipe」，即可增加配方。



- 于控制器内所生成的配方文件



4.5.3 RecipeManCommands

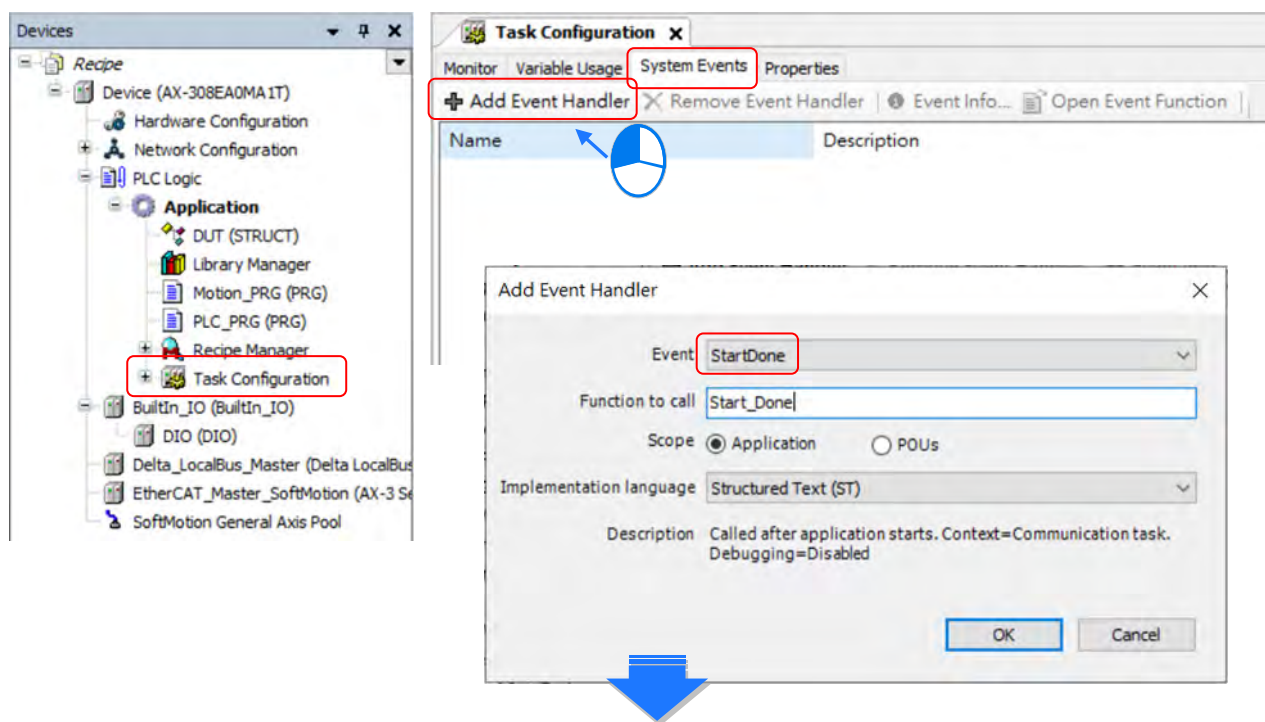
「Recipe_Management.library」的功能块「RecipeManCommands」提供了数种方法，供用户将配方文件导入至变量，或是将控制器变量导出配方文件。

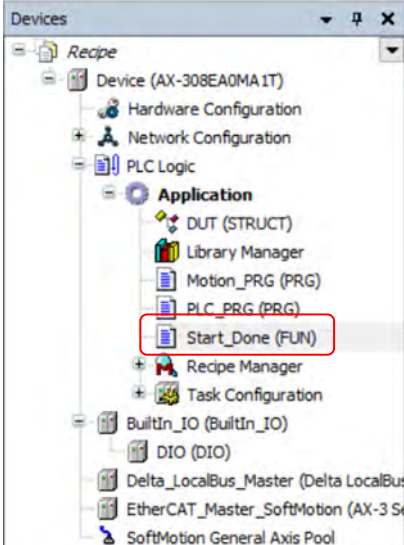
RecipeManCommands 方法	说明
LoadAndWriteRecipe	从默认的配方文件将配方导入控制器变量
LoadFromAndWriteRecipe	从指定的配方文件将配方导入控制器变量
ReadAndSaveAS	将控制器变量另存至指定文件
ReadAndSaveRecipe	将控制器变量导出至默认的配方文件
ReadAndSaveRecipeAS	将控制器变量导出至默认的配方文件，另存至指定文件。

● 使用范例1

本范例利用「Add Event Handler」，新增「StartDone」的事件并搭配「LoadAndWriteRecipe」方法，来达到于控制器运行状态从「STOP」改变成「RUN」后，自动将配方定义「Recipes_01」内的配方「Case1」导入控制器内相对应的变量。

4





The screenshot shows the 'Devices' tree view for a 'Recipe' device. Under the 'Application' folder, the 'Start_Done (FUN)' function is highlighted with a red box. The tree view includes folders for Hardware Configuration, Network Configuration, PLC Logic, Application, BuiltIn_IO, DIO, Delta_LocalBus_Master, EtherCAT_Master_SoftMotion, and SoftMotion General Axis Pool.

```

POU: Start_Done

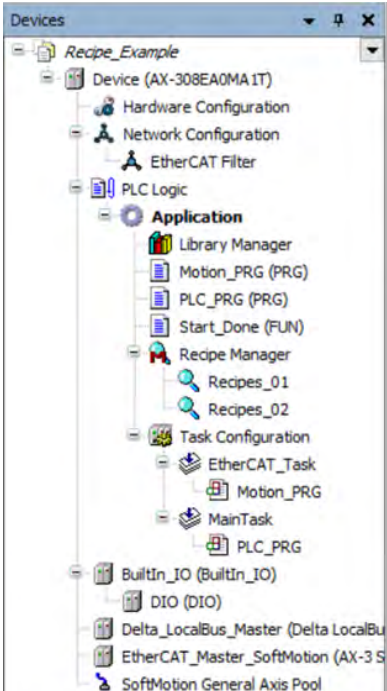
1  FUNCTION Start_Done : DWORD
2  VAR_IN_OUT
3      EventPrm : CmpApp . EVTPARAM_CmpApp ;
4  END_VAR
5  VAR
6      FB0 : RecipeManCommands ;
7  END_VAR

1  FB0 . LoadAndWriteRecipe ( RecipeDefinitionName := 'Recipes_01', RecipeName :=
2  'Case1' ) ;

```

- 使用范例2

本范例分别利用「ReadAndSaveRecipe」与「ReadAndSaveAS」方法，将控制器变量导出至原配方文件与指定的文件。



The screenshot shows the 'Devices' tree view for a 'Recipe_Example' device. Under the 'Application' folder, the 'PLC_PRG' program is highlighted. The tree view includes folders for Hardware Configuration, Network Configuration, EtherCAT Filter, PLC Logic, Application, Recipe Manager, Recipes_01, Recipes_02, Task Configuration, EtherCAT_Task, Motion_PRG, MainTask, BuiltIn_IO, DIO, Delta_LocalBus_Master, EtherCAT_Master_SoftMotion, and SoftMotion General Axis Pool.

```

POU: PLC_PRG

1  PROGRAM PLC_PRG
2  VAR
3      iVar : INT ;
4      dwVar , dw_Retur : DWORD ;
5      udi_Return : UDINT ;
6      bVar0 , bVar1 : BOOL ;
7      FB1 : RecipeManCommands ;
8  END_VAR

1  IF bVar0 THEN
2      %MW3 := 50 ;
3      iVar := 60 ;
4      dwVar := 70 ;
5      udi_Return := FB1 . ReadAndSaveRecipe ( RecipeDefinitionName :=
6      'Recipes_01', RecipeName := 'Case1' ) ;
7      bVar0 := FALSE ;
8  END_IF

9  IF bVar1 THEN
10     %MW3 := %MW3 + 10 ;
11     iVar := iVar + 20 ;
12     dwVar := dwVar + 30 ;
13     udi_Return := FB1 . ReadAndSaveAS ( RecipeDefinitionName := 'Recipes_01',
14     FileName := 'POU_Variable.txtrecipe' ) ;
15     bVar1 := FALSE ;
16  END_IF

```

- 将「bVar0」与「bVar1」设定为「ON」后，控制器其所生成的文件与内容。

Name	Size	Modified
..		
visu		
trend		
alarms		
ac_persistence		
_cnc		
Application		
Case1.Recipes_01.txtrecipe	89 bytes	2000/1/1 上午 08:34
Case2.Recipes_01.txtrecipe	92 bytes	2000/1/1 上午 08:33
Case20.Recipes_02.txtrecipe	50 bytes	2000/1/1 上午 08:33
Case21.Recipes_02.txtrecipe	50 bytes	2000/1/1 上午 08:33
POU_Variable.txtrecipe	90 bytes	2000/1/1 上午 08:34

```

Case1.Recipes_01.txtrecipe - Notepad
File Edit Format View Help
%MW3:=50:=WORD:==:==:==
PLC_PRG.iVar:=60:=INT:==:==:==
PLC_PRG.dwVar:=70:=DWORD:==:==:==
    
```

```

POU_Variable.txtrecipe - Notepad
File Edit Format View Help
%MW3:=60:=WORD:==:==:==
PLC_PRG.iVar:=80:=INT:==:==:==
PLC_PRG.dwVar:=100:=DWORD:==:==:==
    
```

第5章 Hardware Configuration


目录

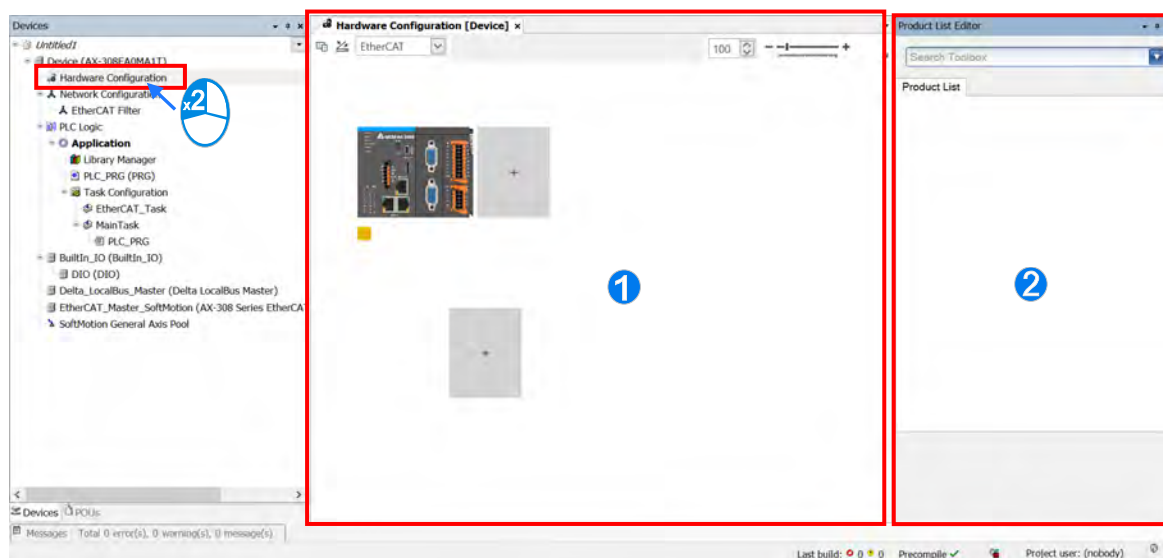
5.1	Hardware Configuration	5-2
5.1.1	环境介绍.....	5-2
5.1.2	新增模块.....	5-5
5.1.3	删除模块.....	5-7
5.1.4	模块的复制/贴上	5-9
5.1.5	模块的剪下/贴上	5-11

5.1 Hardware Configuration


Hardware Configuration 为 DIADesigner-AX 所提供的硬件规划辅助工具，其功能包括 CPU 主机与模块的参数设定，本章节会先介绍相关功能的设定方式。

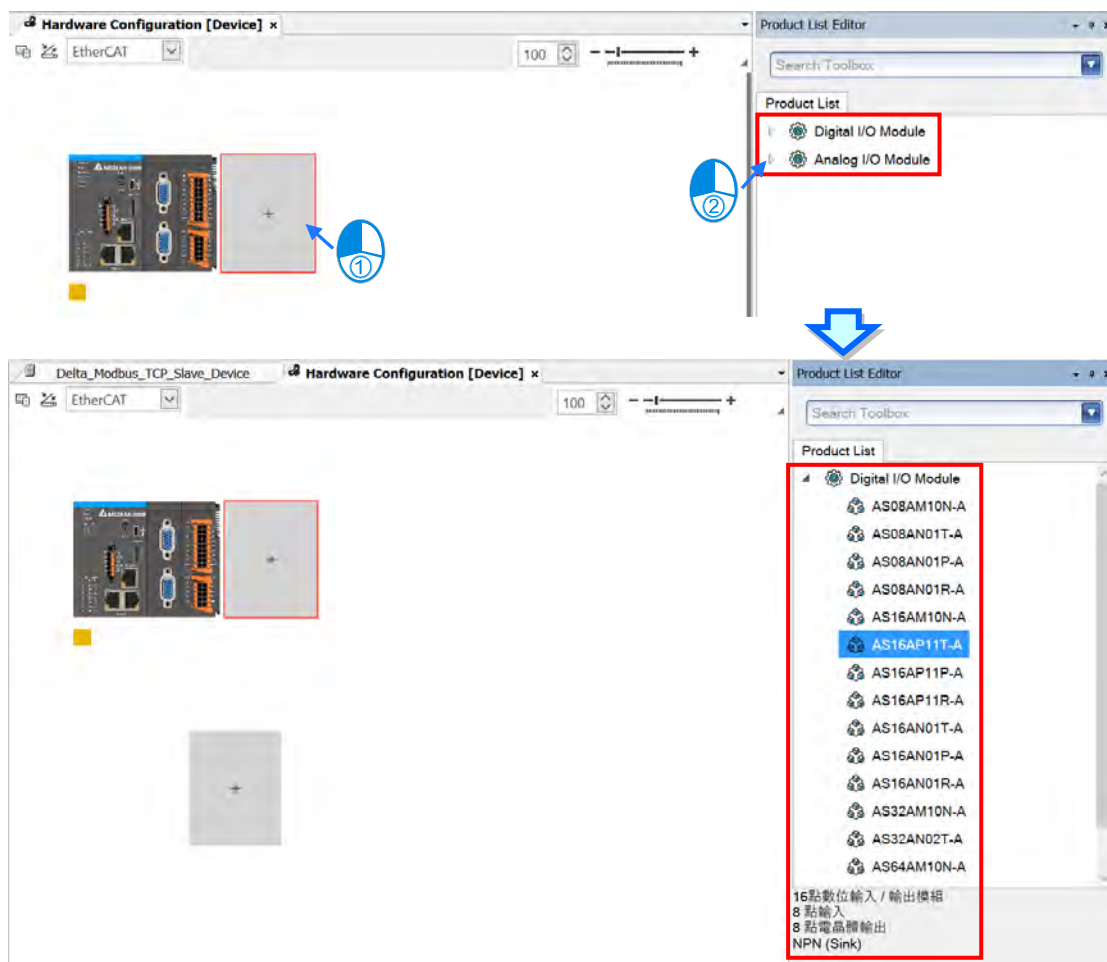
5.1.1 环境介绍


在左侧 Devices 窗口中，使用鼠标左键双击  Hardware Configuration，Hardware Configuration [Device] 选项分页窗口会在设备列右侧打开，如下图所示。

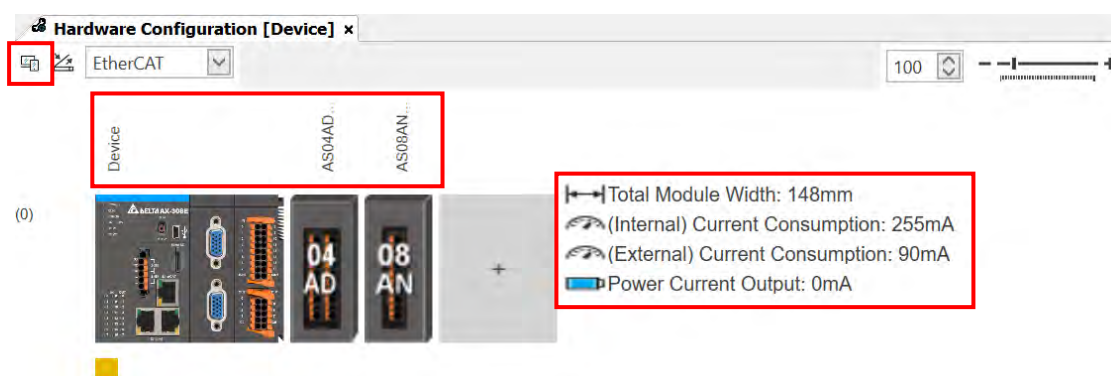


1. 系统配置区：为主要的工作区，用户可在此处进行整个系统的配置与设定。
2. 产品列表：此区会以目录管理的方式呈现所有可用的硬件资源。

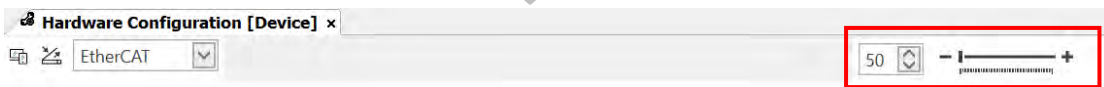
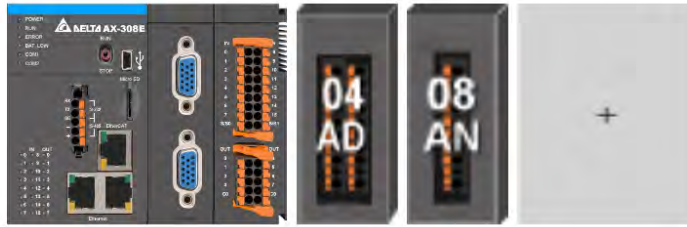
单击主机右侧“+”号图标，在右侧产品列表会出现可用的模块类型，再单击  图标后会往下展开可使用模块的机种名称，若再在该机种名称上单击单击，下方会有简易的模块说明。



鼠标左键单击分页左上角图示  后，在系统配置区内会显示目前配置的相关信息，如各新增模块名称、系统机壳总宽度和系统消耗电流量等。

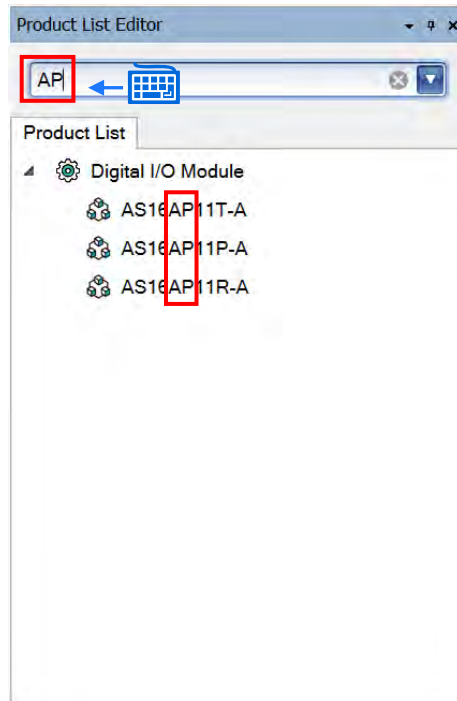
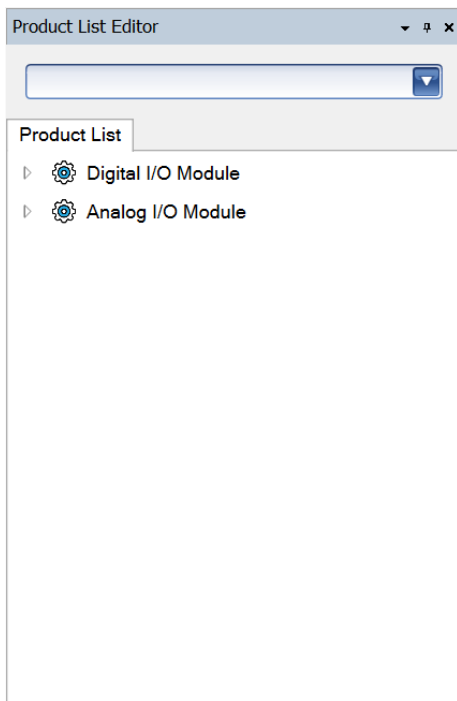


在分页右上角数值调整与+/-左右横拉对象如下图红框处，通过数值调大(往+方向)或数值调小(往-方向)，可以改变系统配置区内的图标变大或变小，方便用户观看目前系统整体配置。



5

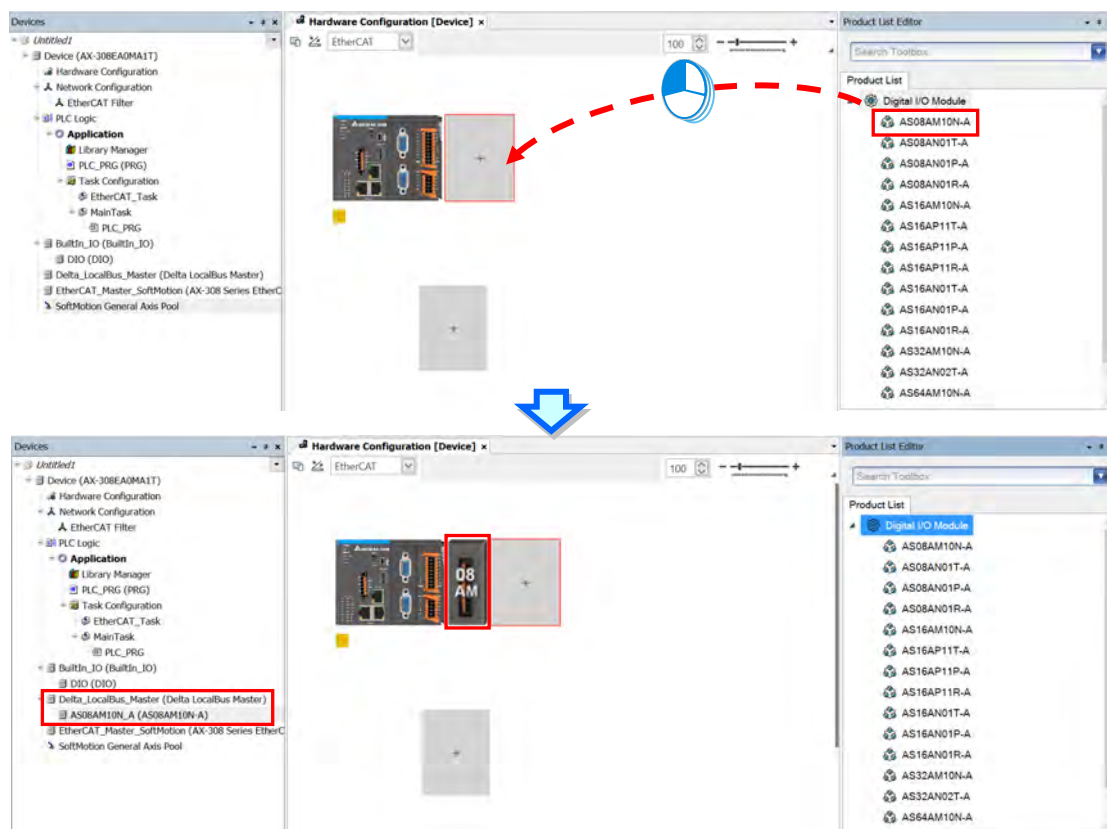
在右侧产品列表编辑窗口上方的「Search Toolbox」输入欲搜索的关键词后按下 Enter，下方列表即会显示名称内有完全相同连续字符串的模块项目。




5.1.2 新增模块

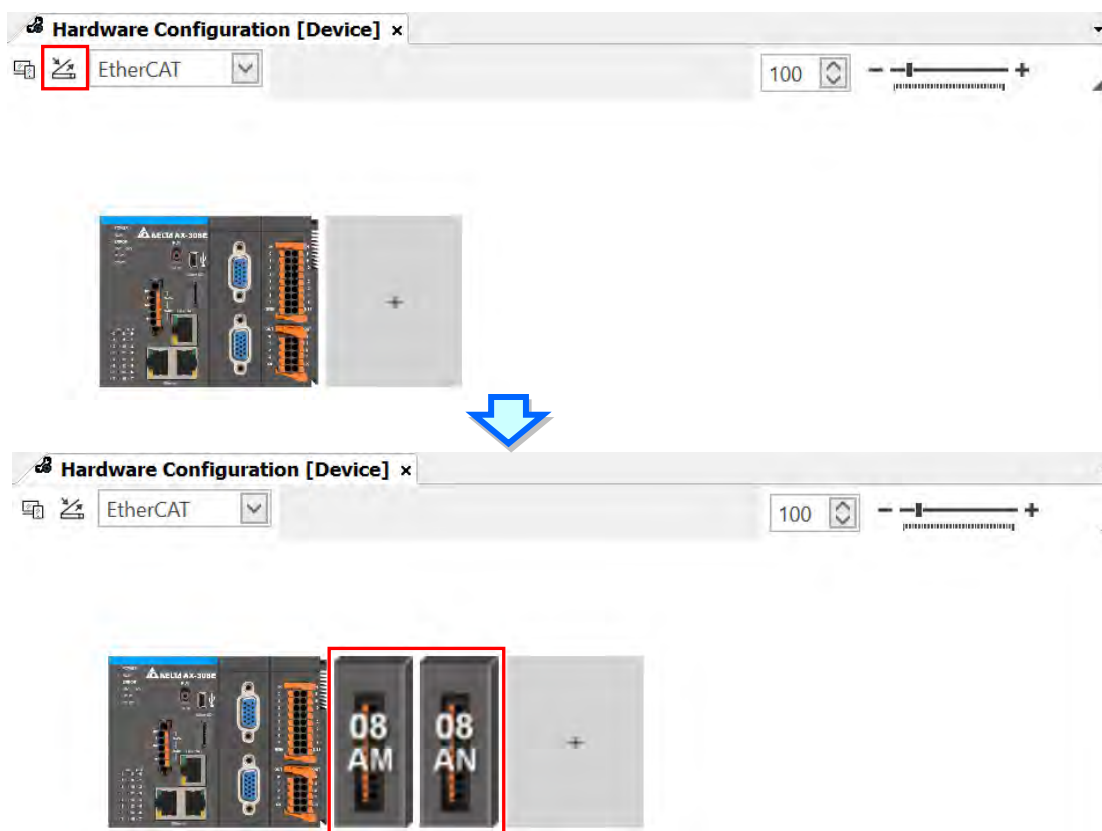
- 方法一

AX-3 系列机种为无背板设计，扩展模块只能安装至右侧的安装位置，手动新增模块后安装位置将会继续出现在最右侧供用户依序插入右侧模块，且在同时间新增的模块也会自动新增在左侧设备列「LocalBus」下方。



● 方法二

若主机与已插入的扩展模块是上电的情况下，且 Gateway 设定已经正常联机下，可通过按下分页左上角的按钮图标 ，来自动扫描后新增模块配置，同样新增的模块也会自动新增在左侧设备列「Delta_LocalBus_Master」下方。



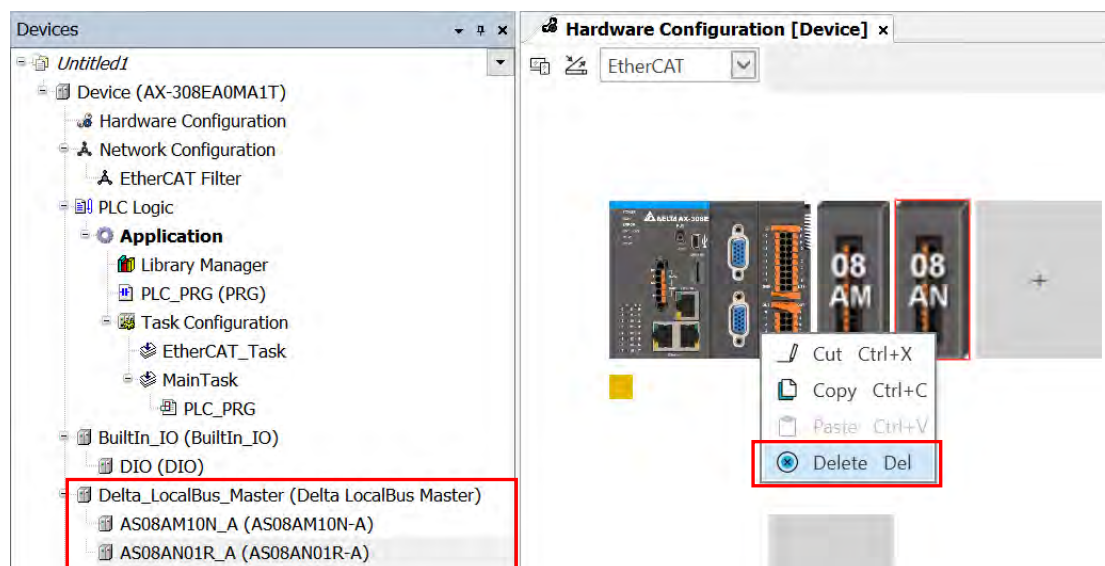
5

5.1.3 删除模块

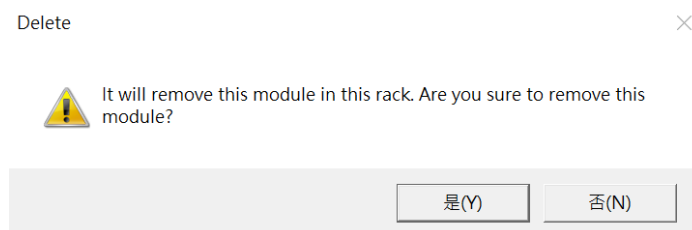
删除已配置模块的方法有下列两种 (CPU 主机不可删除):

- 方法一

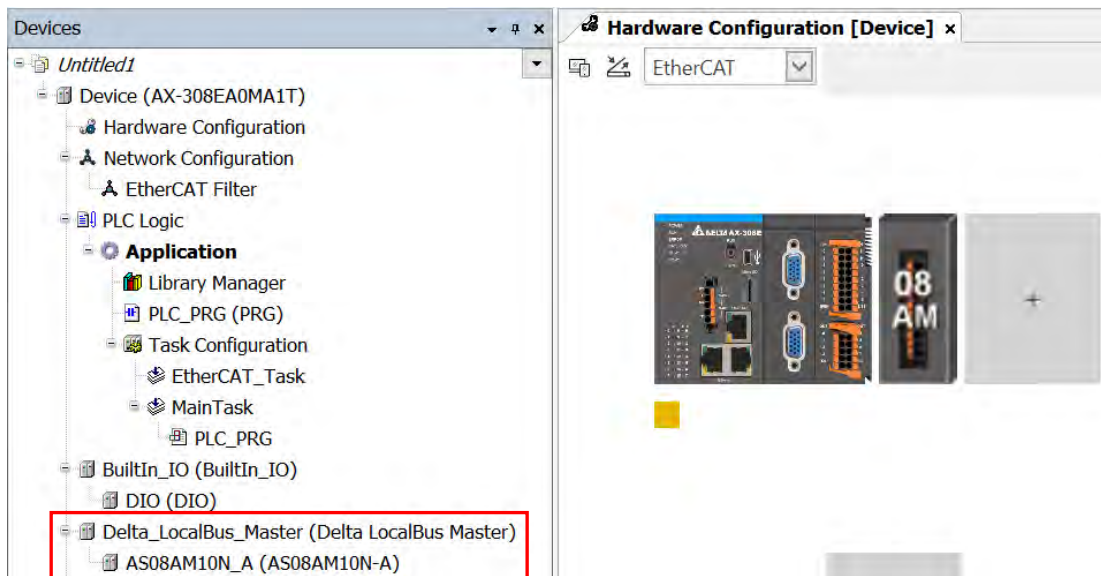
先在系统配置区单击欲删除的模块，按下鼠标右键后在快捷选单中单击「Delete」，或是直接按下键盘的【DEL】键。



按下后会跳出信息框确认是否移除模块，此时再按下「是 (Y)」即可移除。



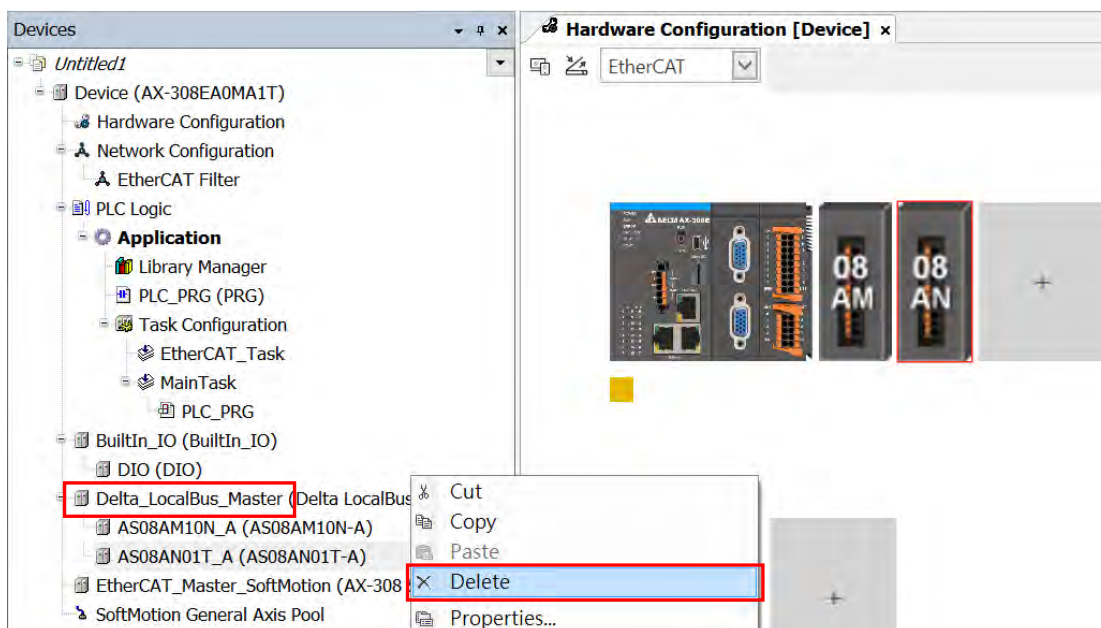
在同时间删除的模块也会自动被移除在左侧设备列「LocalBus」下方。



● 方法二

在左侧设备列「Delta_LocalBus_Master」下方单击欲删除的模块，按下鼠标右键后在快捷选中单击「Delete」即可，在同时间删除的模块也会自动被移除在系统配置区内图标。

5

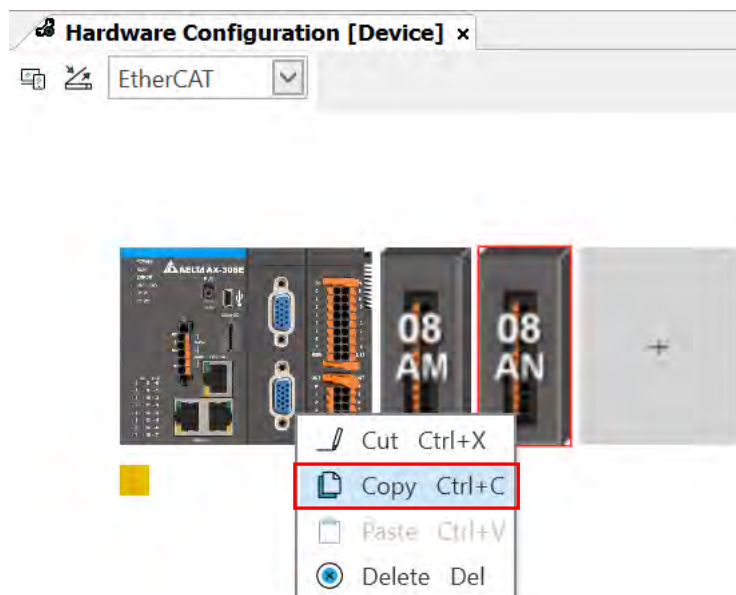


5.1.4 模块的复制/贴上

复制模块的方法有下列两种（CPU 主机不可进行复制/贴上操作）：

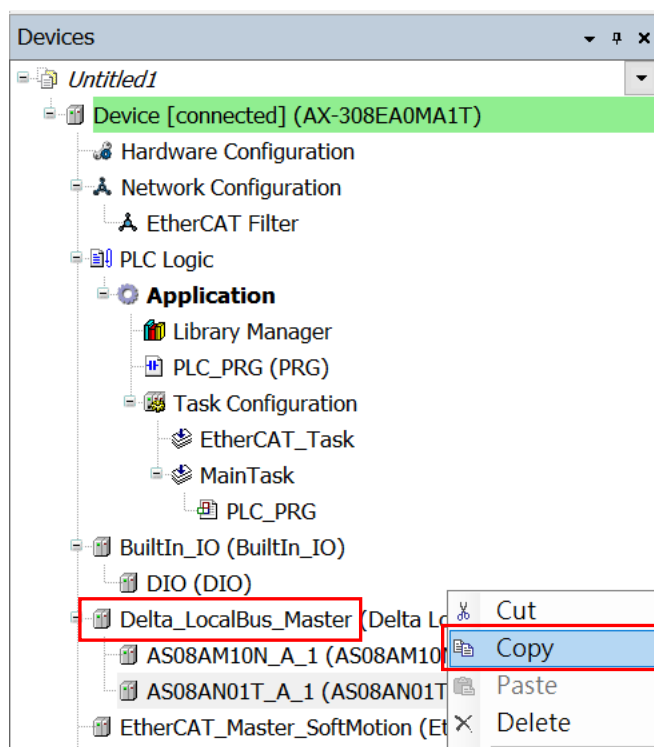
- 方法一

在系统配置区单击欲复制的模块后按下鼠标右键，再在快捷选单中单击「Copy」。



- 方法二

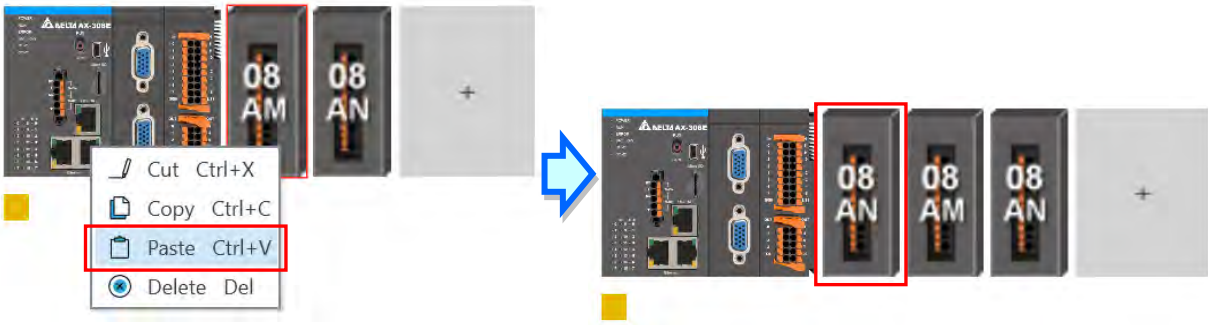
在左侧设备列「Delta_LocalBus_Master」下方单击欲复制的模块后按下鼠标右键，再在快捷选单中单击「Copy」。



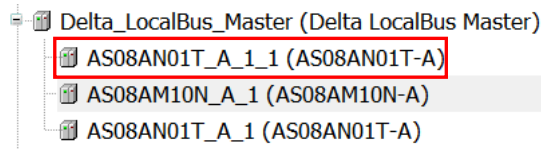
贴上模块的方式有下列两种：

● 方法一

在系统配置区单击欲贴上的位置后按下鼠标右键，之后在快捷选中单击「Paste」，若欲贴上的位置已有模块，原有的模块会往右顺移一个位置；若贴上的位置为图标「+」号的位置，模块则会贴在系统最右方的位置。

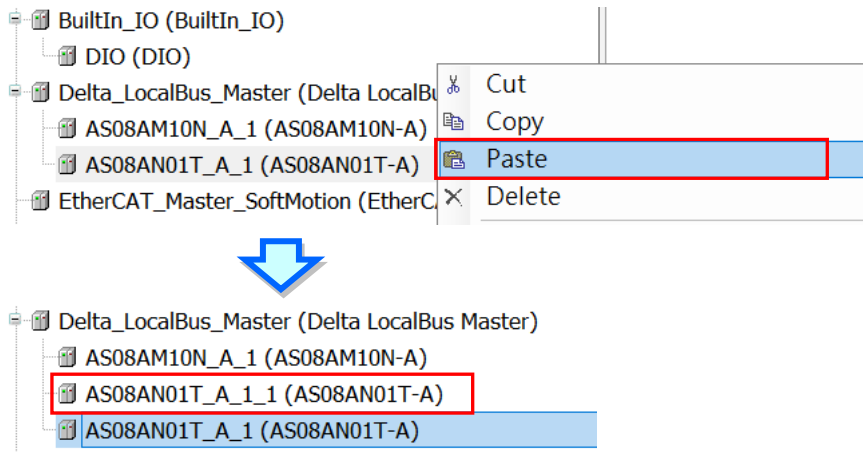


同样新贴上的模块也会依顺序自动新增在左侧设备列「Delta_LocalBus_Master」下方。



● 方法二

在左侧设备列「Delta_LocalBus_Master」上或下方模块单击欲贴上的位置后按下鼠标右键，之后在快捷选中单击「Paste」，若欲贴上的位置是在现有的模块上，原有的模块会往下顺移一个位置；若单击贴上的位置为「Delta_LocalBus_Master」，模块则会贴在列表最下方的位置。



同样新贴上的模块也会依顺序自动新增在系统配置区。

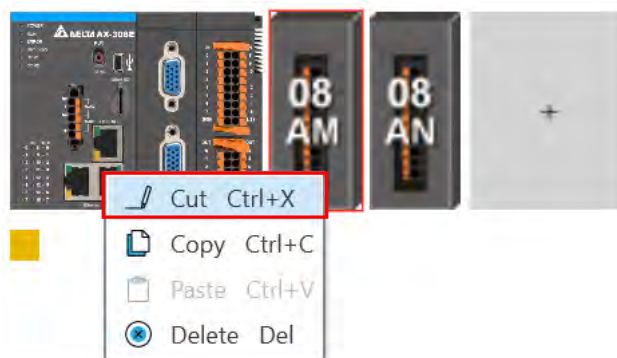


5.1.5 模块的剪下/贴上

剪下模块的方法有下列两种 (CPU 主机不可进行剪下/贴上操作):

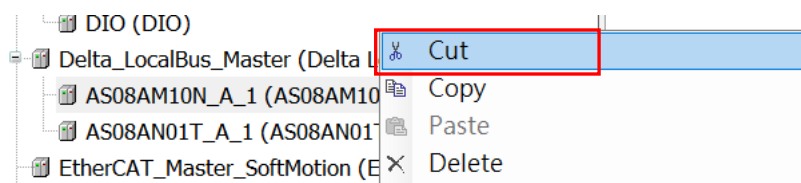
- 方法一

在系统配置区单击欲剪下的模块后按下鼠标右键，再在快捷选单中单击「Cut」。



- 方法二

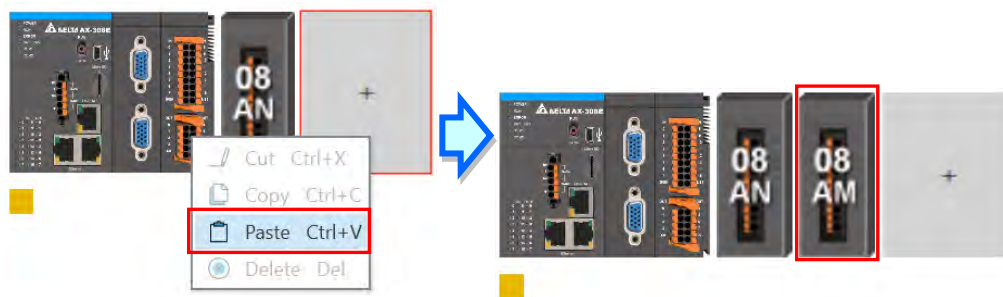
在左侧设备列「Delta_LocalBus_Master」下方单击欲剪下的模块后按下鼠标右键，再在快捷选单中单击「Cut」。



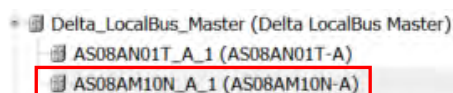
贴上模块的方式有下列两种：

- 方法一

在系统配置区单击欲贴的位置后按下鼠标右键，之后在快捷选单中单击「Paste」，若欲贴上的位置已有模块，原有的模块会往右顺移一个位置；若贴上的位置为图标「+」号的位置，模块则会贴在系统最右方的位置。

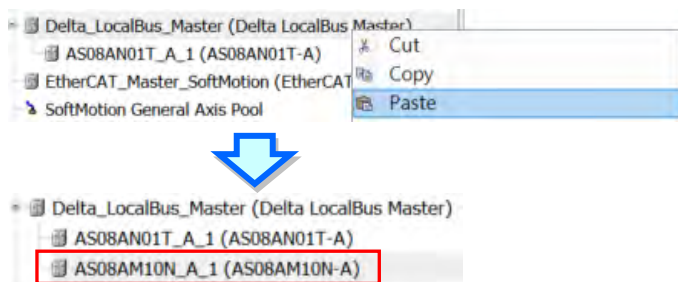


同样贴上的模块也会依新的顺序自动新增在左侧设备列「Delta_LocalBus_Master」下方。



● 方法二

在左侧设备列「Delta_LocalBus_Master」上或下方模块单击欲贴上的位置后按下鼠标右键，之后在快捷选单中单击「Paste」，若欲贴上的位置是在现有的模块上，原有的模块会往下顺移一个位置；若单击贴上的位置为「Delta_LocalBus_Master」，模块则会贴在列表最下方的位置。



同样贴上的模块也会依新的顺序自动新增在系统配置区。



第6章 Network Configuration

目录

6.1	Network Configuration	6-2
6.1.1	简介	6-2
6.1.2	基础介绍	6-3
6.1.3	网络架构的配置	6-5
6.1.3.1	配置节点	6-5
6.1.3.2	链接网络	6-7


6.1 Network Configuration

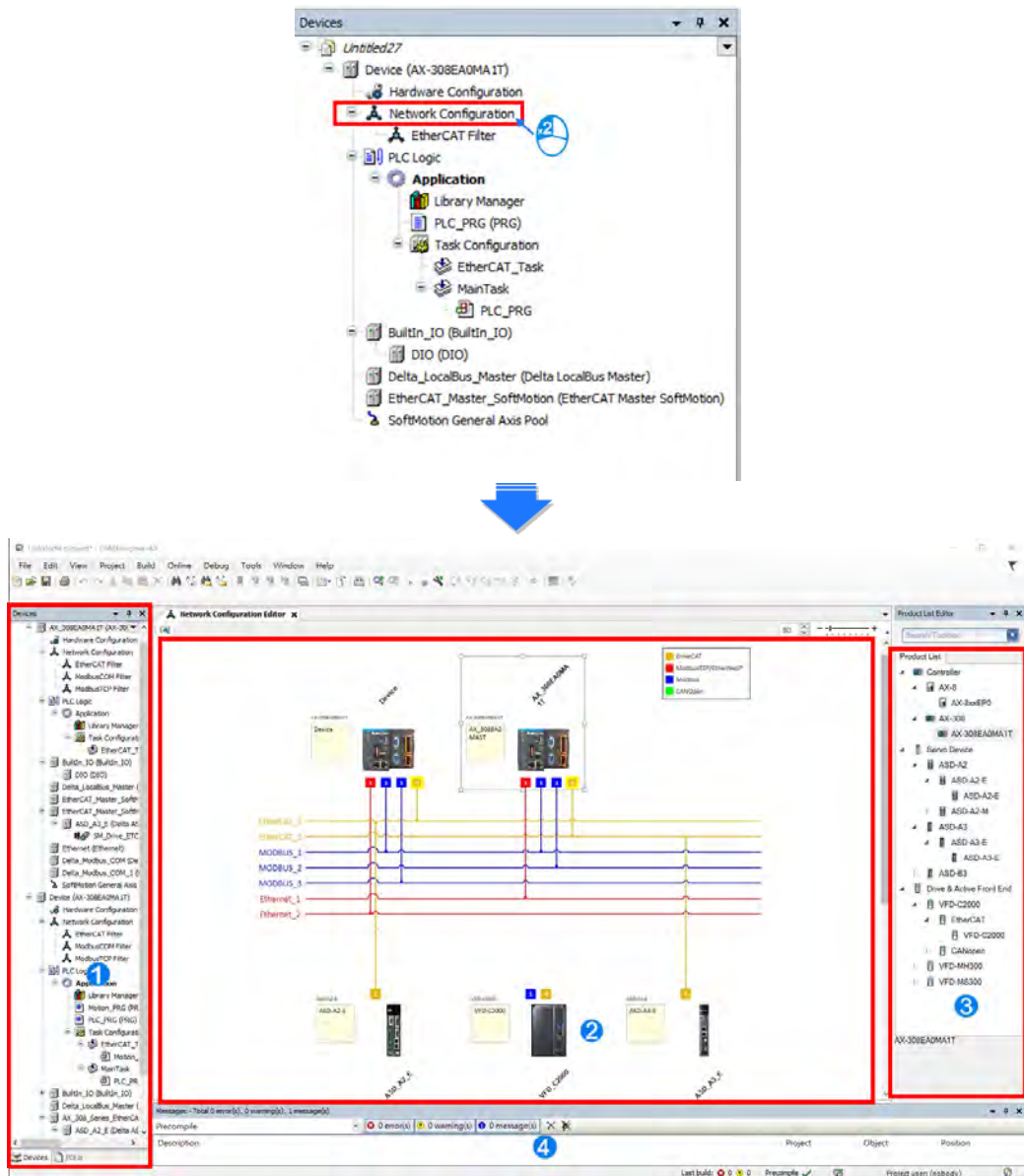
Network Configuration 为 DIADesigner-AX 所提供的网络规划辅助工具，功能在于规划整个项目的网络架构，详细说明将在后续的内容逐一介绍。

6.1.1 简介

Network Configuration 其主要负责的工作如下，

- (a) 规划整个项目的网络 (EtherCAT/MODBUS/Ethernet/CANOpen) 配置，并建立数据的传送路径。
- (b) 规划 EtherCAT 主站的设定。
- (c) 规划 MODBUS 的 COM port 设定。
- (d) 规划 Ethernet 的 IP 设定。

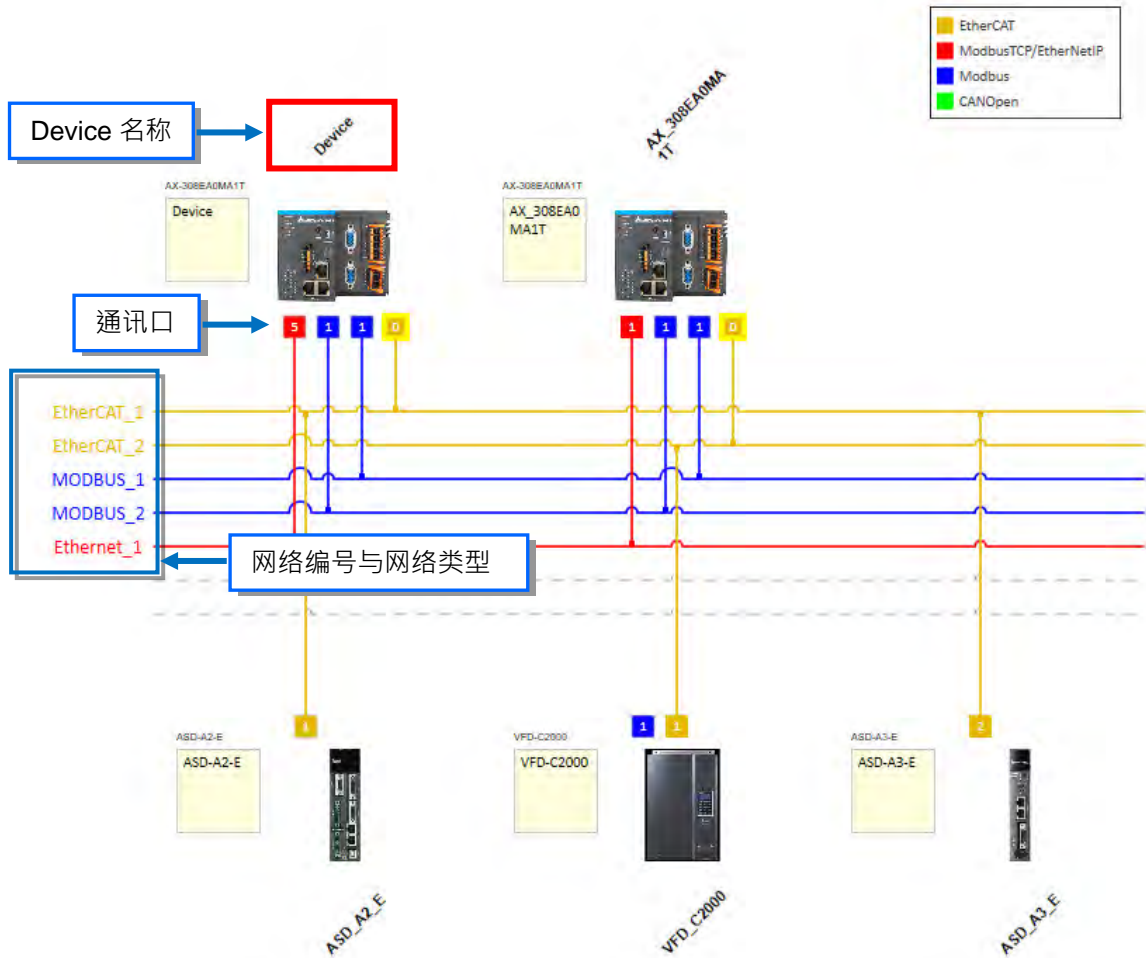
Network Configuration 的定位在于规划整个项目的网络架构，因此其所在的位置会在 Device 下层，而欲开启 Network Configuration 时，请直接在  Network Configuration 项目上双击鼠标左键即可。



- ❶ **Device**：此区以目录管理方式呈现所有已被配置使用的装置。
- ❷ **图标工作区**：为主要的工作区，用户可在此规划整个网络架构。
- ❸ **装置列表**：此区以目录管理的方式呈现所有可使用的装置资源。
- ❹ **信息窗口**：此处会显示操作过程中的相关信息。

6.1.2 基础介绍

在开始进行网络规划之前，针对操作过程所需具备的一些基础认知，我们将在本节当中先加以介绍。



● Device 与网络类型

Device 为 Network Configuration 中最基本的组件，其所指的可以是控制器、服务器、驱动器等等，或是用户自行定义的设备；而此处的**网络类型**则是指连接各个装置的联机，且每个通讯口都必须指定一个唯一的网络类型，而其类型则有 MODBUS、Ethernet、EtherCAT 及 CANOpen。另外，因各装置用以连接至网络的端点即是该装置的通讯口，因此当一个装置具有两个以上的通讯口时，该装置便可同时连接至不同网络。

● Device 名称

Device 名称主要的功能是在做为主机的识别，与 Device 树状结构中之名称相对应，而备注则对操作上无任何的意义。

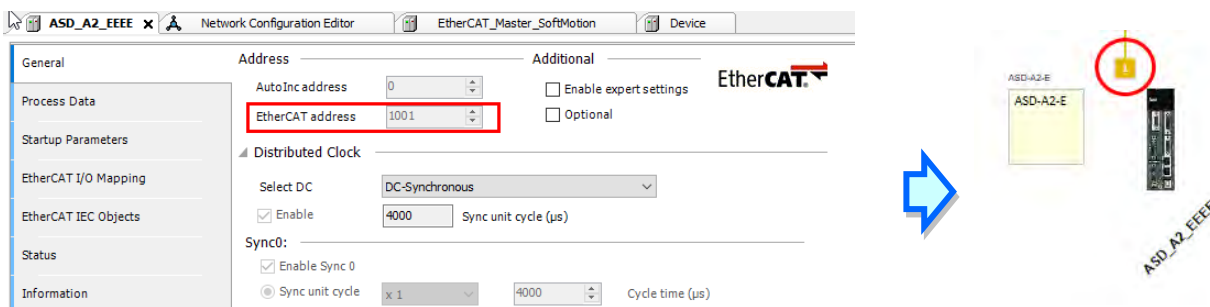
● 网络类型与通讯口

■ EtherCAT

土黄色线为 EtherCAT 通讯，在节点上双击鼠标左键可编辑 EtherCAT 设定。主站号码显示为 0 不可变动。

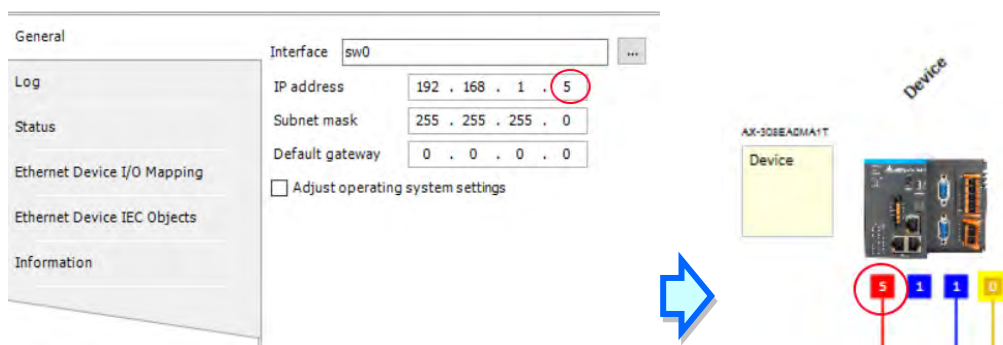
单击从站节点开启从站设定画面，EtherCAT address 后缀显示编号。

EtherCAT address 1001 节点显示 1。



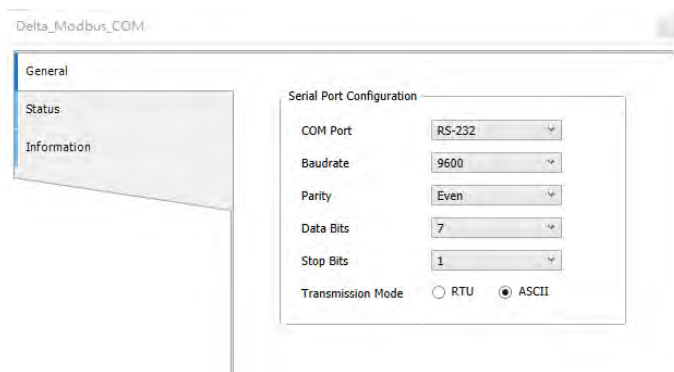
■ MODBUS TCP/EtherNet/IP

红色线为 Ethernet port 可以为 MODBUS TCP 通讯或是 EtherNet/IP 通讯，在节点上，双击鼠标左键，可编辑 IP Address。节点显示 IP Address 第四码。



■ MODBUS

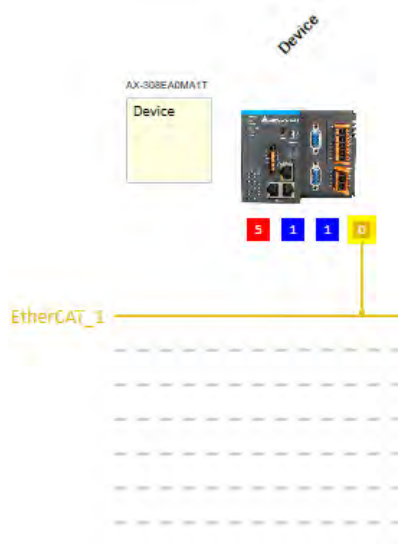
蓝色线为 MODBUS 通讯 (RS-232/RS-485)，在节点上，双击鼠标左键，可进入 COM port 设定画面。



6.1.3 网络架构的配置

6.1.3.1 配置节点

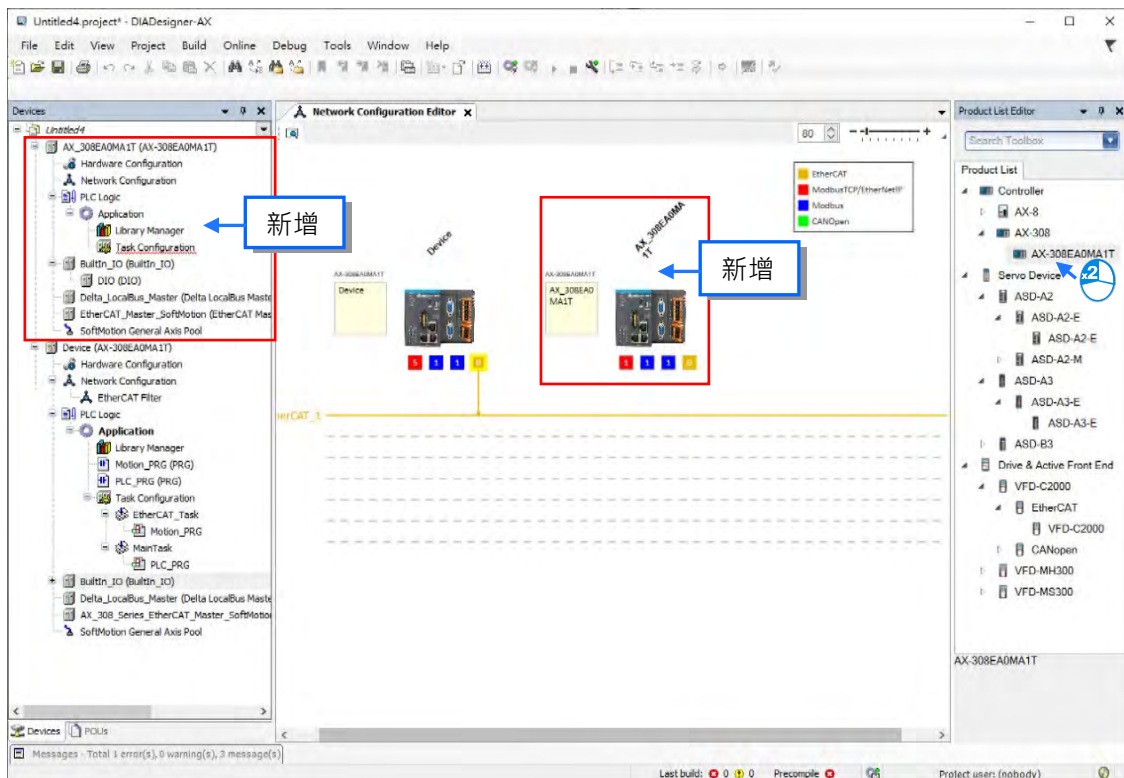
首次开启项目中的 Network Configuration 时，系统会依据项目中所建立的机种，自动配置在工作区中。



用户可另外加入其它控制器、服务器、驱动器或其它装置，而加入新装置的方法共有下列三种。

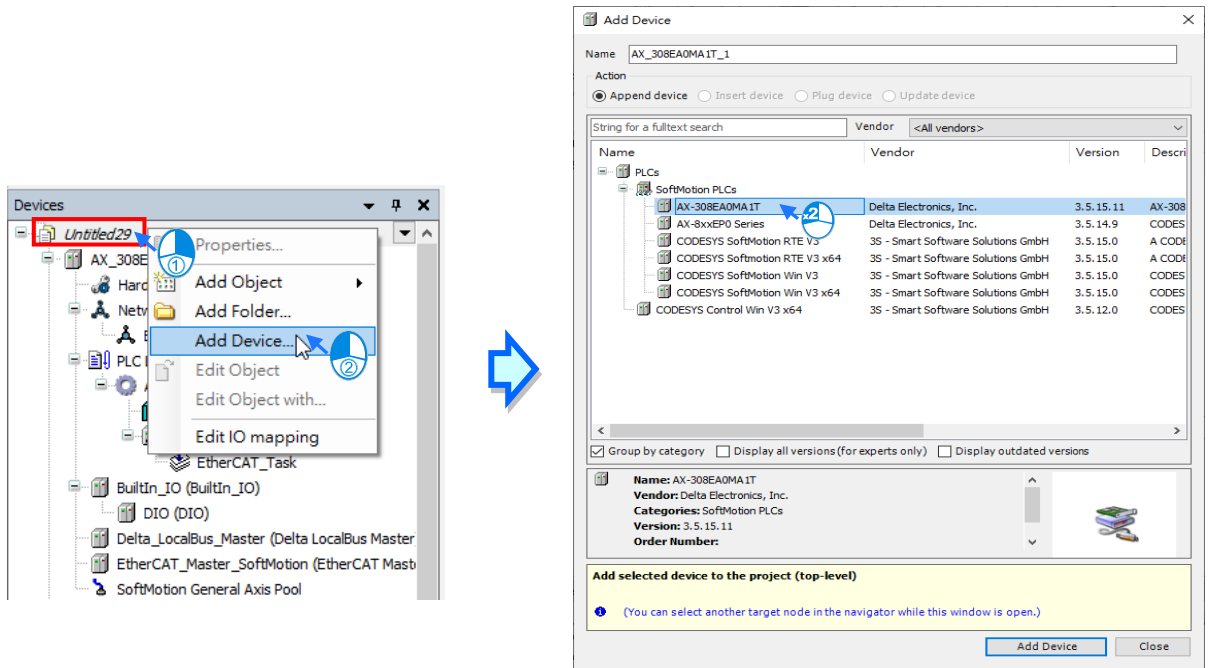
- 方法一

置于右侧 Product List Editor 中，选择欲新增的主机或装置型号后，在该项目上双击鼠标左键，其新增装置就会被新增到工作区及 Devices 树状结构中。



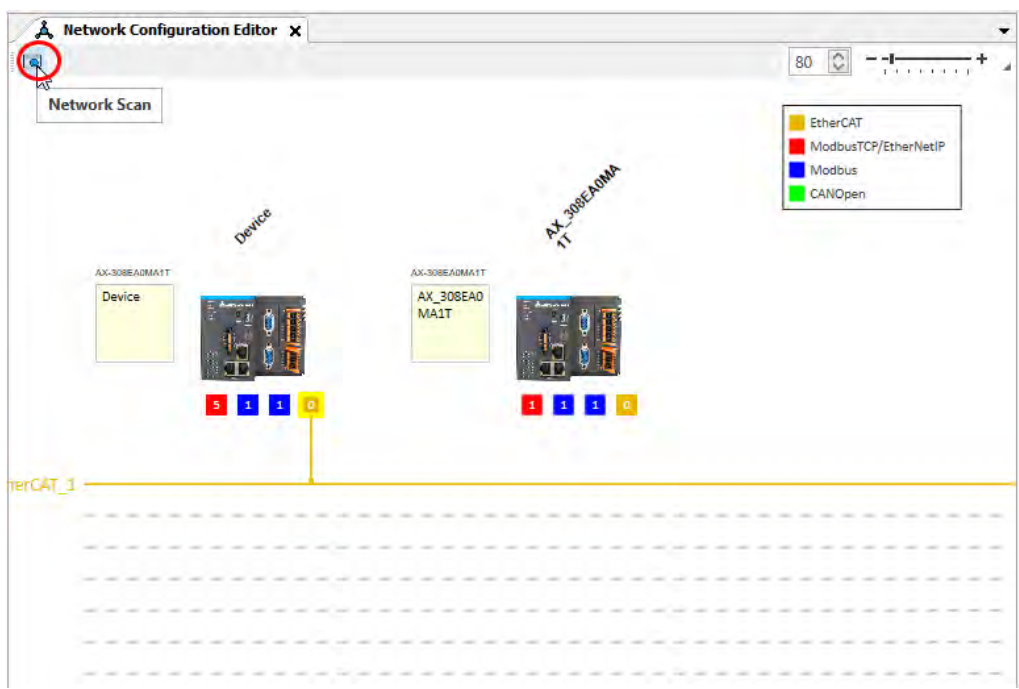
● 方法二

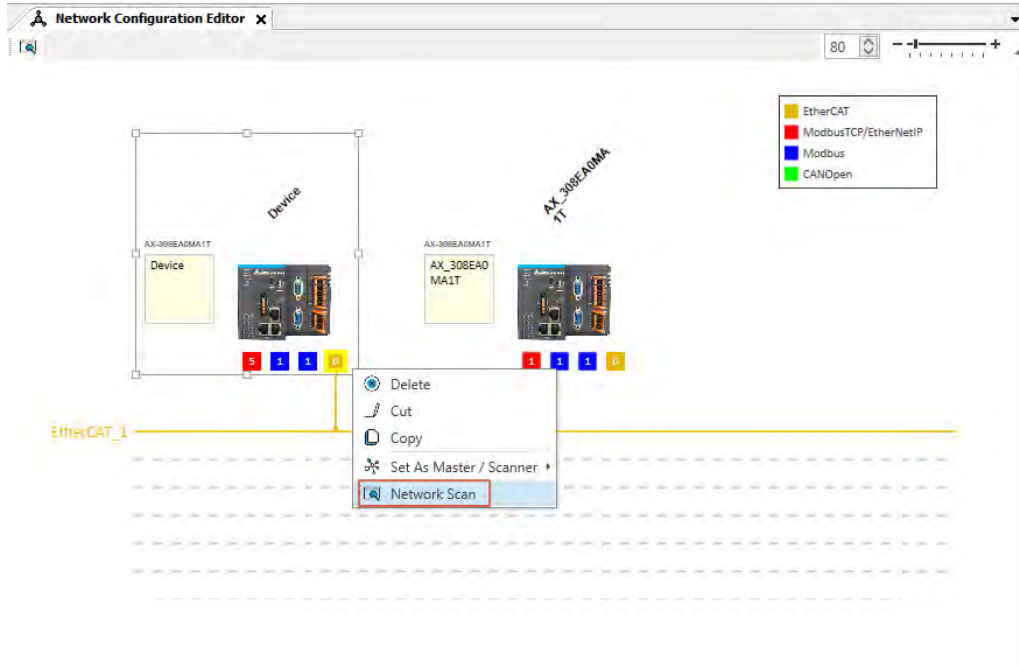
在左侧 Devices 树状结构中，单击项目名称，鼠标单击右键，选择「Add Device」。选择欲新增之装置即可。



● 方法三

单击图标工具栏的 图标或是单击控制器，单击鼠标右键，选取「Network Scan」，可自动扫描到已经配置之装置及网络联机。





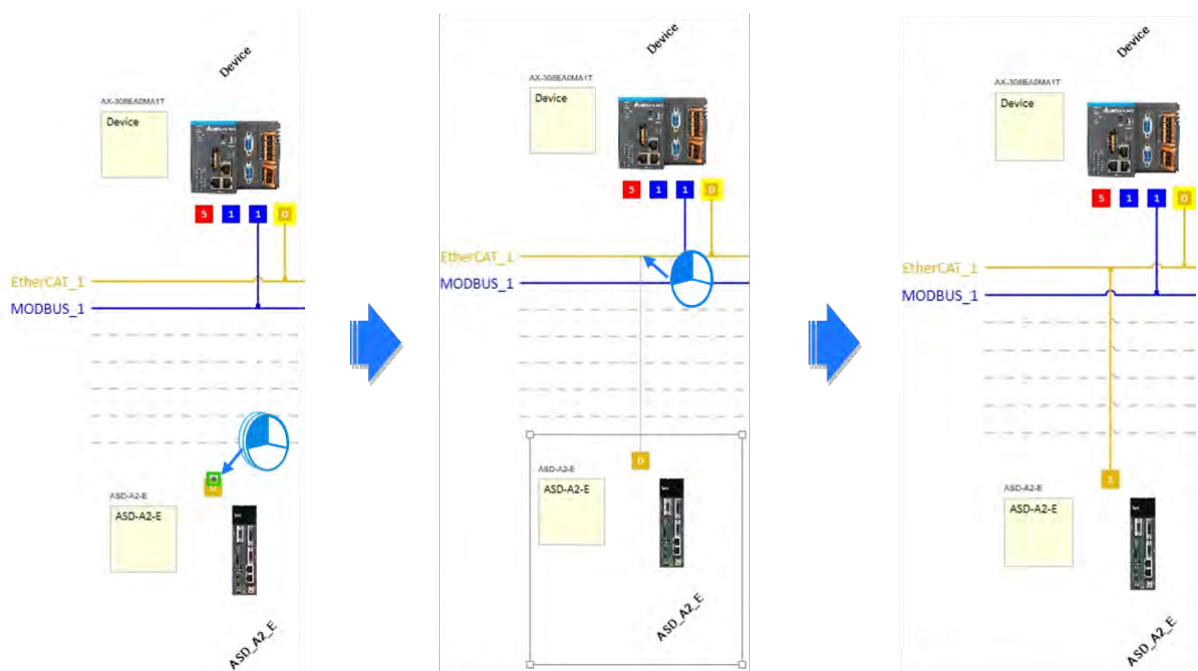
6.1.3.2 链接网络

完成节点配置之后，接下来便是将所有节点依据事先的规划连上所属的网络。所支持的网络方式已在第 6.1.2 节说明。

- 建立网络联机

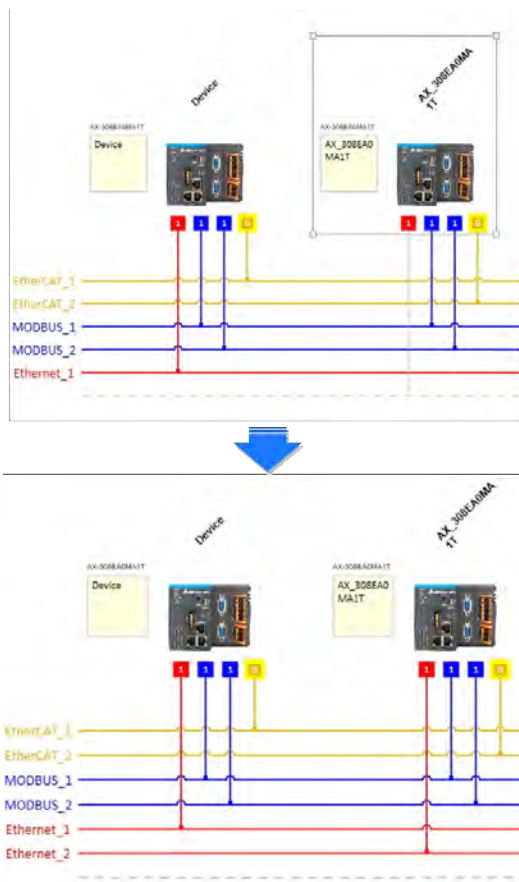
- 情境一：

将鼠标拖曳至已建立的网络上后，且该网络的类型与通讯口一致，则放开鼠标后便可建立联机。



■ 情境二：

将鼠标拖曳至最下方的虚线放开，则会新增一条相同类型之相同网络并将通讯口连接上，并在下方再新增一条虚线。



第7章 运动控制基本设定与操作

目录

7.1	运动控制指令概述	7-5
7.1.1	运动控制指令	7-5
7.1.2	运动控制指令应用要点	7-5
7.1.3	运动控制指令的分类	7-5
7.2	运动项目建立流程	7-6
7.2.1	项目建立流程图	7-6
7.2.2	项目建立流程	7-6
7.3	监控与调测	7-13
7.3.1	使用方法	7-13
7.3.2	轴参数设定范例	7-13
7.3.3	测试轴运转	7-15
7.4	运动控制装置	7-18
7.4.1	轴设定	7-18
7.4.2	轴介绍	7-18
7.4.2.1	轴参数介绍	7-19
7.4.2.2	程序中使用轴	7-28
7.4.3	单轴配置流程与设定步骤	7-29
7.4.4	轴组设定	7-37
7.4.4.1	轴组参数介绍	7-37
7.4.4.2	程序中使用轴组	7-39
7.4.5	轴组配置流程与设定步骤	7-40
7.5	运动轴变量	7-45
7.5.1	单轴变数	7-45
7.5.2	轴组变数	7-47
7.6	运动控制程序编辑	7-49
7.6.1	运动控制程序	7-49
7.6.1.1	DIADesigner-AX 编成架构与种类	7-49

7.6.1.2	DIADesigner-AX 中的 POU.....	7-50
7.6.1.3	DIADesigner-AX 新增 POU.....	7-50
7.6.1.4	PDO 映射 (Mapping)	7-53
7.6.2	轴状态转换	7-53
7.6.2.1	轴状态	7-53
7.6.2.2	轴组状态	7-55
7.6.3	运动控制指令的执行和状态	7-56
7.6.3.1	执行指令的基本规则.....	7-56
7.6.3.2	输入/输出引脚时序图.....	7-58
7.6.3.3	单一运动指令重复执行之行为.....	7-59
7.6.3.4	多个运动指令同时启动	7-59
7.6.3.5	运动指令同时执行之行为	7-60
7.6.4	位置.....	7-65
7.6.4.1	位置的种类.....	7-65
7.6.5	凸轮表和凸轮架构	7-65
7.6.5.1	电子凸轮架构	7-65
7.6.5.2	电子凸轮建立	7-66
7.7	运动控制功能	7-70
7.7.1	系统架构	7-70
7.7.2	单轴控制	7-70
7.7.2.1	周期同步位置模式 (Cyclic Synchronous Position Mode)	7-70
7.7.2.2	位置规划模式 (Profile Position Mode)	7-71
7.7.2.3	定位	7-71
7.7.2.4	停止方式	7-72
7.7.2.5	电子齿轮 (MC_GearIn)	7-75
7.7.2.6	位置指定电子齿轮 (MC_GearInPos)	7-77
7.7.2.7	电子凸轮 (MC_CamIn)	7-78
7.7.3	速度控制	7-87
7.7.3.1	CSP 模式.....	7-87
7.7.3.2	CSV 模式.....	7-88
7.7.3.3	速度规划模式 (Profile Velocity Mode)	7-88
7.7.4	扭矩控制	7-89
7.7.5	单轴控制的常用功能	7-90

7.7.5.1	位置命令	7-90
7.7.5.2	速度命令	7-91
7.7.5.3	加速度与减速度命令	7-92
7.7.5.4	加速度的变化率 (跃度) 命令	7-93
7.7.5.5	轴方向	7-94
7.7.6	轴组控制	7-96
7.7.6.1	轴组直线插补	7-96
7.7.6.2	轴组螺旋插补	7-97
7.7.6.3	轴组命令停止	7-98
7.7.7	本体高速 IO	7-99
7.7.7.1	I/O 功能配置	7-99
7.7.7.2	DIO 功能设定	7-100
7.7.7.3	SSI 编码器功能设定	7-103
7.7.7.4	Pulse Encoder 功能设定	7-109
7.7.7.5	Capture/Compare 功能设定	7-116
7.7.7.6	Pulse Output 功能设定	7-121
7.7.7.7	高速 IO 错误确认	7-126
7.7.8	其它功能	7-127
7.7.8.1	更改现在位置	7-127
7.7.8.2	软件极限	7-127
7.7.8.3	追随误差设定	7-128
7.7.8.4	挺杆点功能	7-129
7.7.8.5	捕捉位置	7-129
7.8	范例程序	7-131
7.8.1	设备架构	7-131
7.8.1.1	使用设备	7-131
7.8.1.2	设定	7-131
7.8.2	基本项目范例	7-131
7.8.2.1	伺服系统 Servo On	7-133
7.8.2.2	单轴错误重置与监控	7-134
7.8.2.3	监控指令错误	7-137
7.8.2.4	单轴操作立即停止	7-139
7.8.2.5	原点回归	7-141

7.8.2.6	绝对值定位.....	7-144
7.8.2.7	凸轮动作中切换凸轮表.....	7-145
7.8.2.8	凸轮动作中执行主轴的相位补偿.....	7-154
7.8.2.9	移动控制过程中变更当前位置.....	7-161
7.8.2.10	主从轴齿轮啮合过程中执行位移追加.....	7-166

7.1 运动控制指令概述

7.1.1 运动控制指令

本节介绍了运动控制程序的元素，包括装置、符号及运动控制指令。

运动控制指令被定义为功能块 (FB)，并在程序中被用于执行多种运动控制目的。本手册所介绍的「MC」运动控制指令即是基于PLCopen*的运动控制功能块的规范所开发。

本节提供PLCopen功能块和台达自定义功能块这两种运动控制指令的概述。PLCopen定义了程序和功能块的接口，从而实现了在IEC61131-3所定义的标准化运动控制编程环境。使用基于PLCopen的指令搭配台达定义的指令，可降低培训和技术支持等成本。

在使用运动控制指令之前，请确保您充分了解装置、符号和指令的功能。可以参考AXn运动功能块手册附录，以快速查询运动控制指令列表和相关错误代码。

*注：PLCopen是一国际标准化组织，其推动以IEC61131-3为基础的工业控制标准，并在PLC编程中被广泛采用。关于PLCopen的更多信息，请查看官方网站 <http://www.plcopen.org/>。

7.1.2 运动控制指令应用要点

此段落说明运动指令应用时的重要操作规格和限制，各指令的详细信息会在第7.6.3节运动控制指令说明。

- 运动控制指令支持的编程语言

DIADesigner-AX支持的所有编程语言来建立、编辑或是维护程序。支持的编辑语言包含梯形图 (LD)、顺序功能图 (SFC)、连续功能图 (CFC) 以及结构化语言 (ST)、功能块图 (FBD)。

关于编程语言的详细信息，请参考DIADesigner-AX软件手册。

7.1.3 运动控制指令的分类

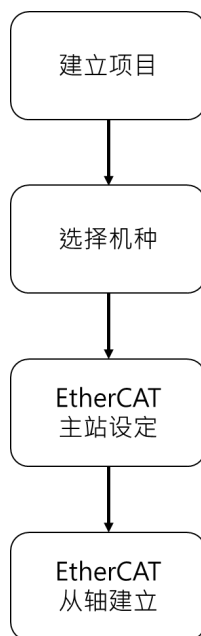
此段落说明运动指令分类，相对应的运动指令分别可于SM3_Basic、DL_MotionControl、DL_MotionControlLight函式库查找，详细内容可参考AX系列运动控制指令手册。

类别	型式	功能群组	说明
单轴运动控制指令	运动控制型	定位控制	“MC_”：基于PLCopen的运动控制指令 “DMC_”：台达自定义运动控制指令 “SMC_”：运动指令 “MC_XXX_DML”：台达自定义运动控制指令，使用于定位轴
		速度控制	
		力矩控制	
		同步控制	
	管理型	管理功能	
多轴运动控制指令	运动控制型	轴组运动	多轴运动
	管理型	多轴管理功能	多轴设定、监控、重置

7.2 运动项目建立流程

7.2.1 项目建立流程图

项目建立流程图，整个运动项目建立及轴配置。



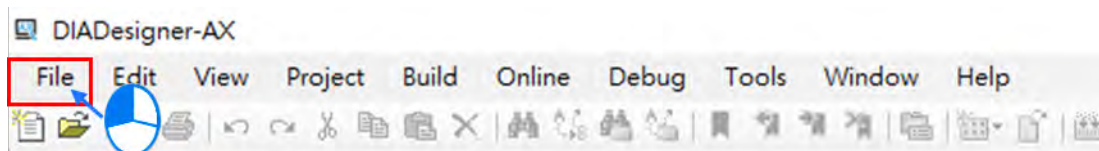
7.2.2 项目建立流程

- 建立运动项目流程介绍
- 在桌面快捷方式处，双击 DIADesigner-AX ICON 开启软件。

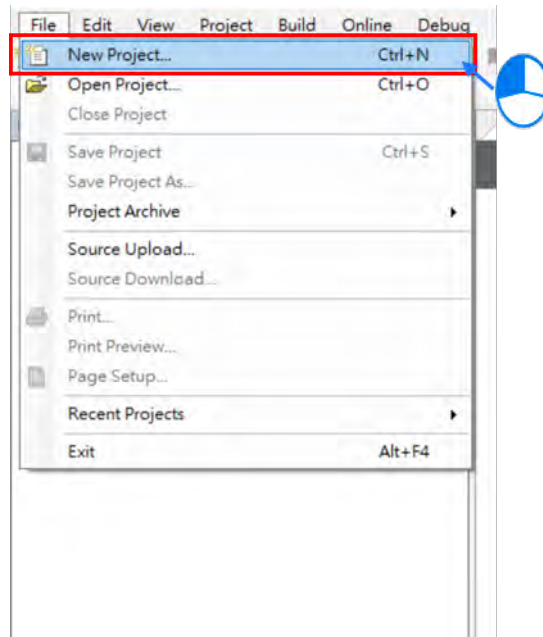
7



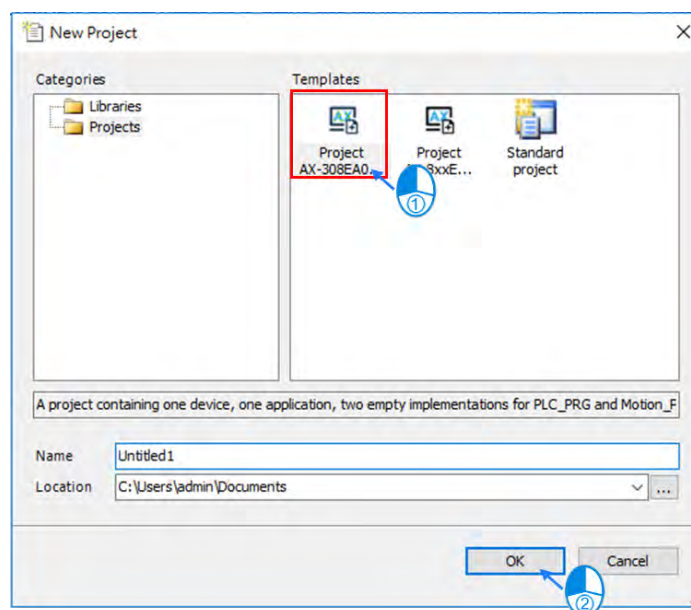
- 单击 File



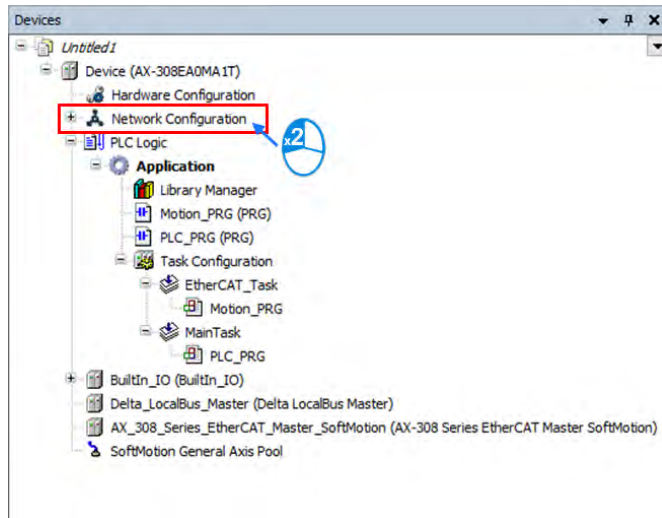
- 选择 New Project



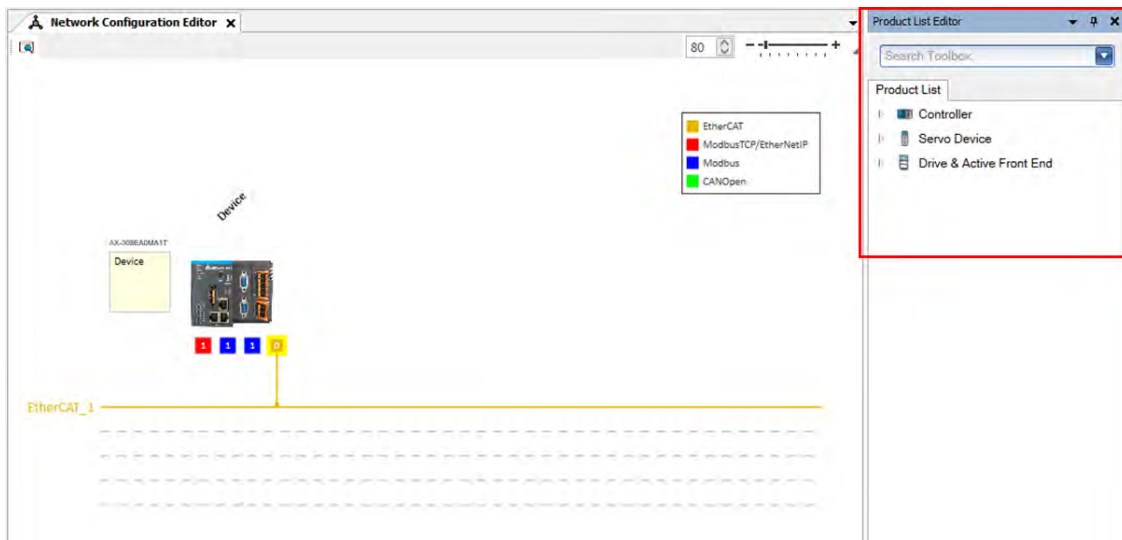
- 接着在 New Project 窗口中依序输入「Name」及「Location」，并选择欲建立项目机种，此流程以 AX-308E 机种作说明，因此请选择「Project AX-308EA0MA1T」。



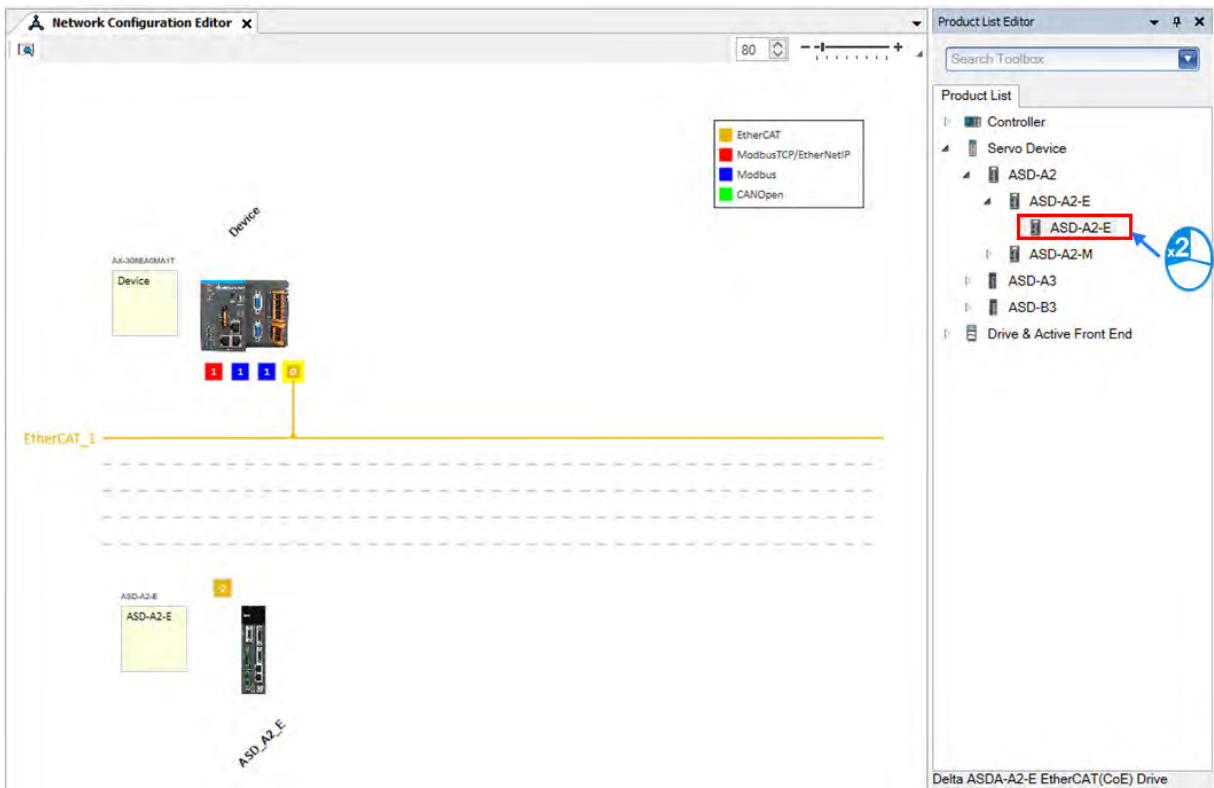
- 鼠标左键双击「Network Configuration」进行 EtherCAT 网络配置



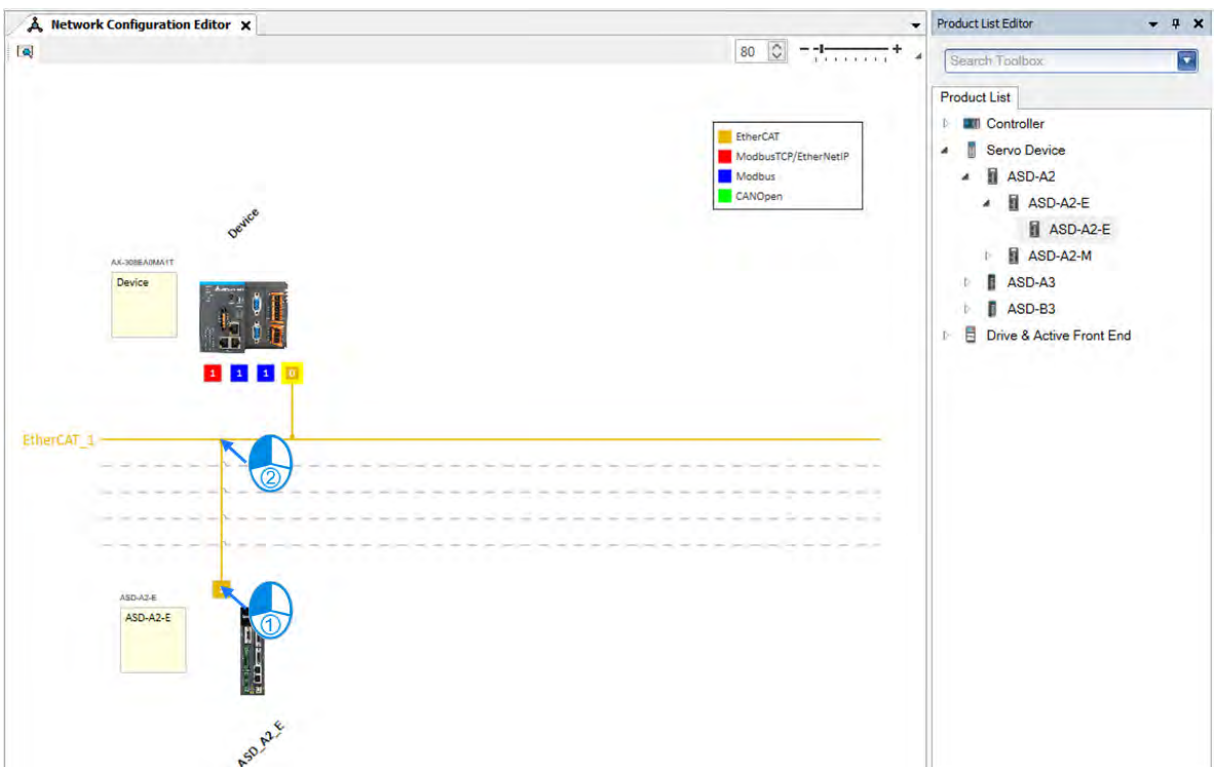
- 鼠标左键双击后会出现「Network Configuration Editor」，从站装置可由右侧「Product List Editor」寻找并新增。



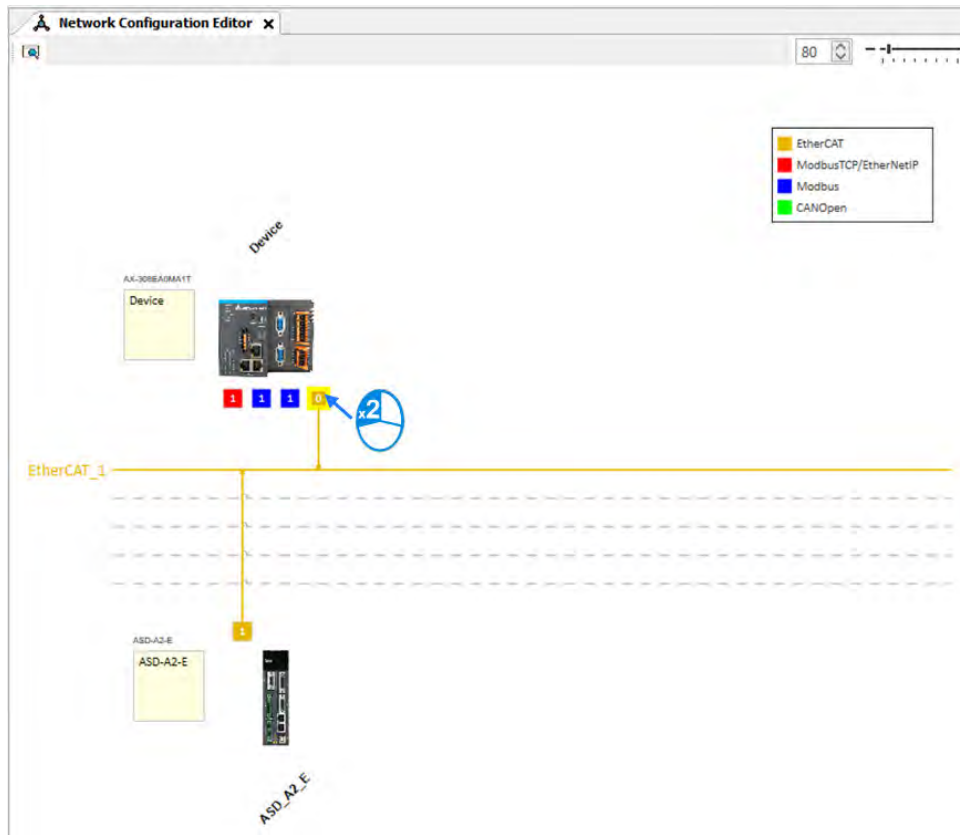
- 请选择「Servo Device」→「ASD-A2」→「ASD-A2-E」，鼠标左键双击后会自动新增至「Network Configuration Editor」中。



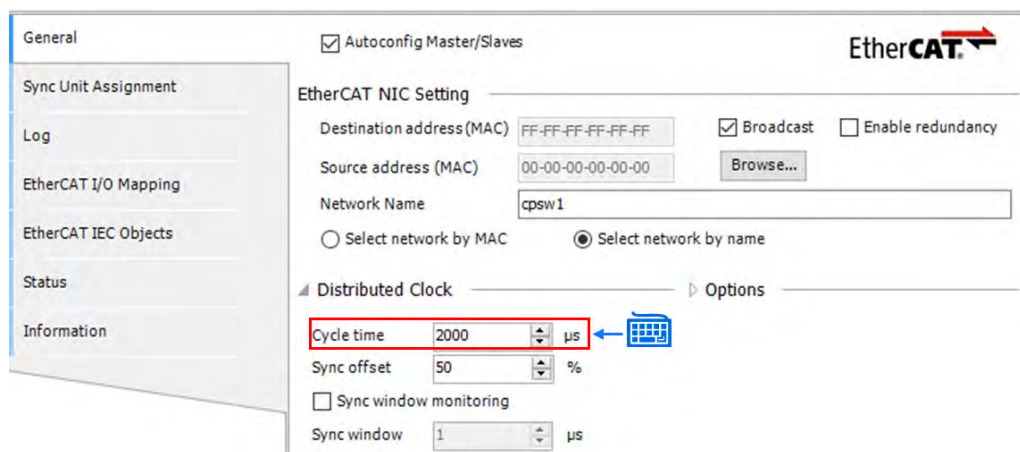
- 请单击从站上面黄色方框，并拖拽至 EtherCAT 主线上完成主站与从站联机配置。



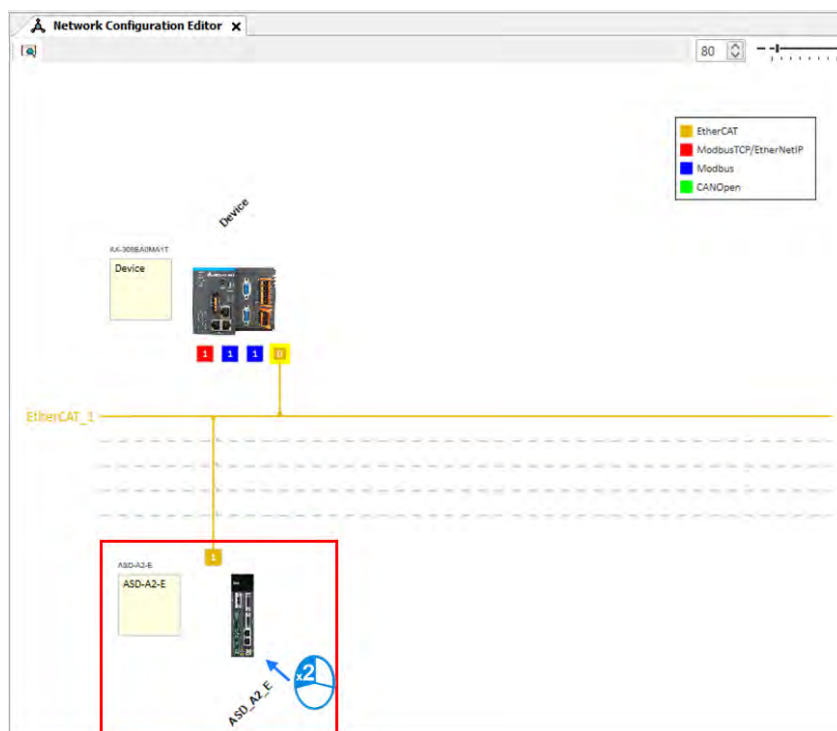
- 鼠标左键双击主站黄色小方框进行 EtherCAT 主站参数设定。



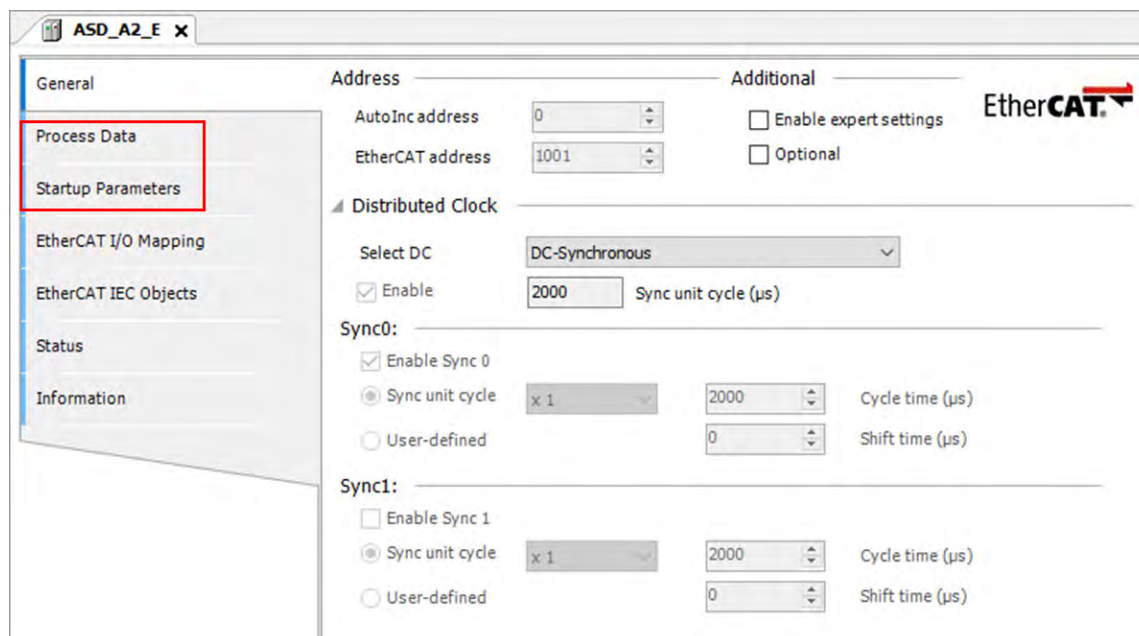
- 主站设定中，可设定 EtherCAT 同步时间。



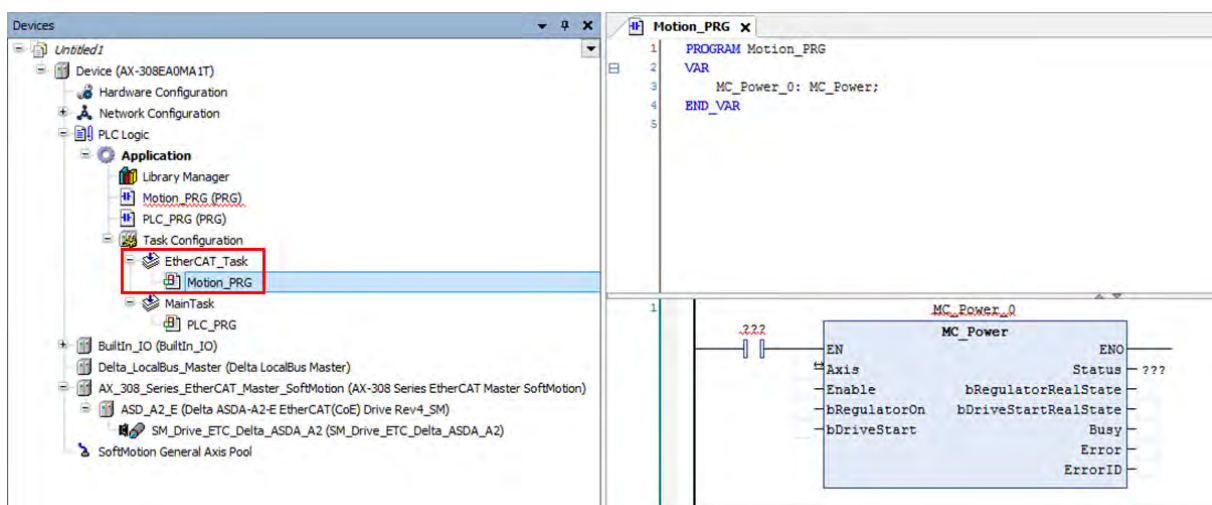
- 请在从站装置上执行鼠标左键双击进入从站 EtherCAT 设定



- 双击后会弹出从站相关设定页面，如站号设置、「Process Data」设定与「Startup Parameters」设定等...



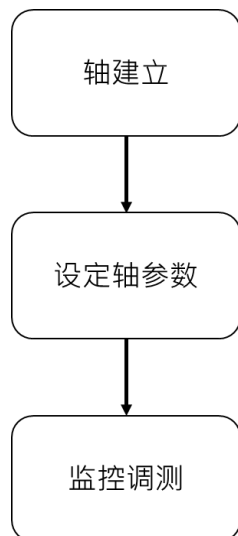
- 之后可以开始进行程序编写。需注意，有关于运动指令部分请在「EtherCAT_Task」底下之 POU 中使用运动功能块编写程序，避免运动功能块无法正常运行。



7.3 监控与调测

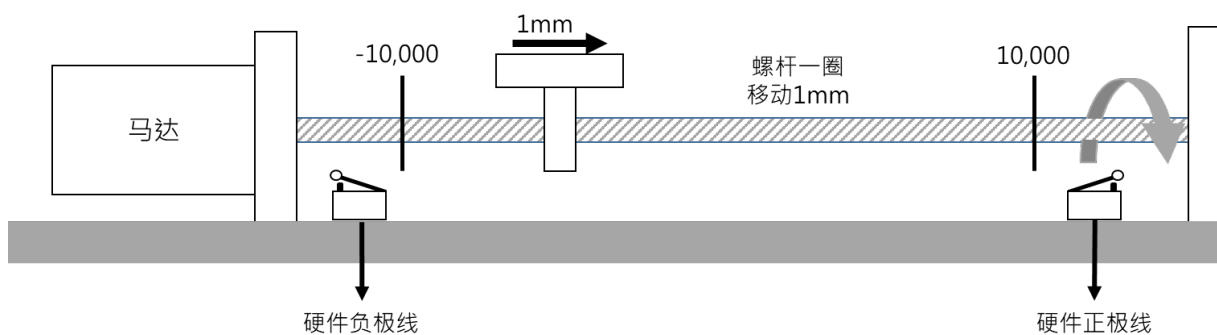
7.3.1 使用方法

建立监控调测方法步骤如下：

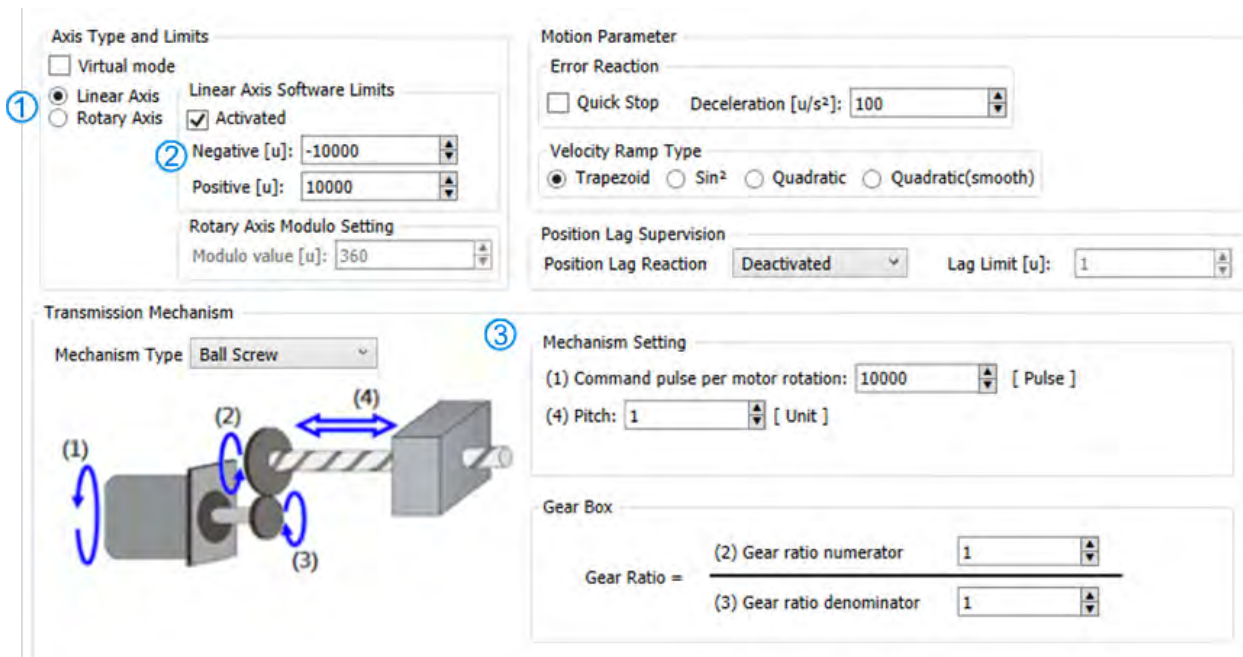


7.3.2 轴参数设定范例

使用软件测试时需先设定轴参数，设定方法如下：



● 轴设定画面



● 参数设定

参数名称	设定
轴种类①	直线轴
旋转一圈脉冲数③	10,000
旋转一圈移动距离③	1 ^{*1}
电子齿轮_分母	128 ^{*2}
电子齿轮_分子	1 ^{*2}
软件正极限②	10,000
软件反极限②	-10,000

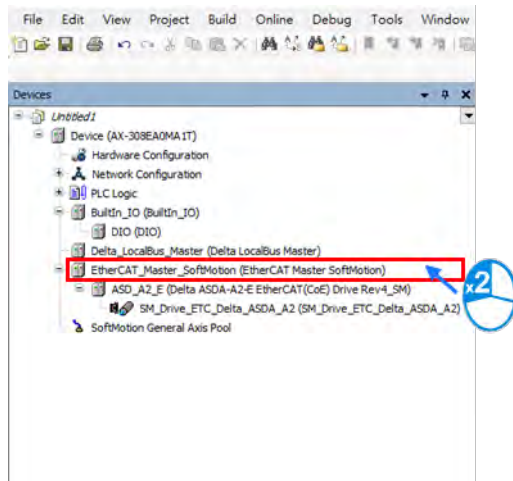
注：

*1 单位为 1mm，需移动 1um 功能块需输入 0.001。

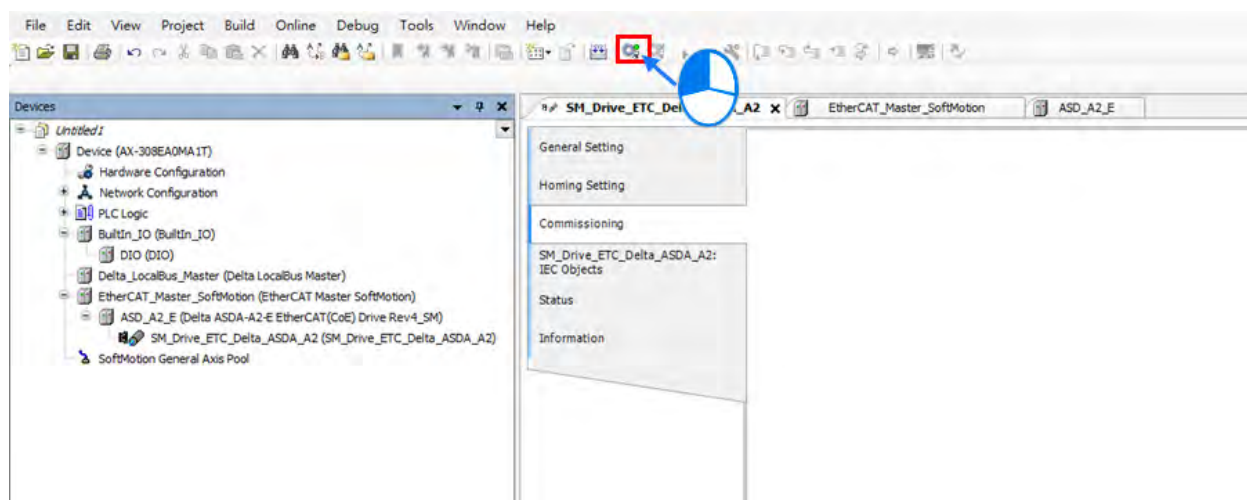
*2 此设定须设定伺服端的 P1-44 与 P1-45。

7.3.3 测试轴运转

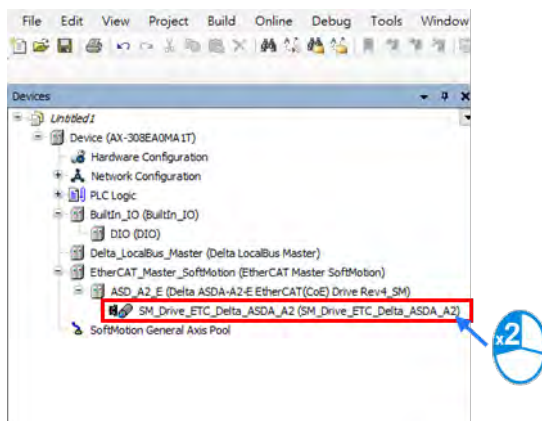
- 选取「EtherCAT_Master_SoftMotion」后双击左键



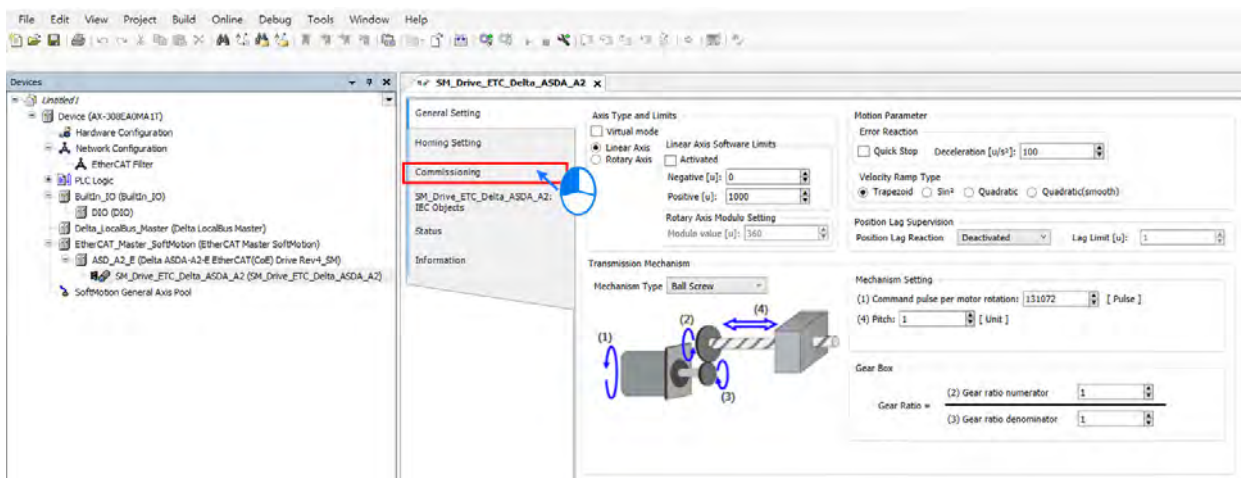
- 选取「Online Config Mode」后单击左键



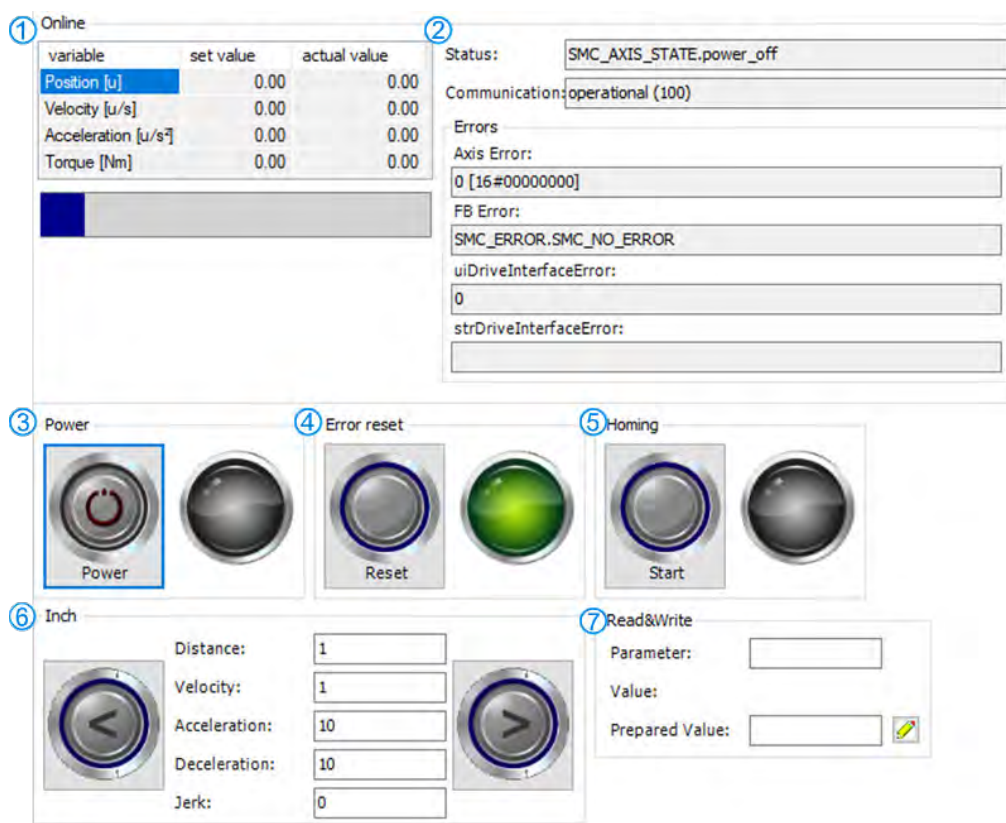
- 当进入在线调测后，选取「SM_Drive_ETC_Delta_ASDA_A2」后双击左键。



- 进入轴参数画面后，单击「Commissioning」分页。



- 针对监控调测画面介绍



■ ① 轴命令信息

名称	功能
Position[u]	命令位置与实际位置
Velocity[u/s]	命令速度与实际速度
Acceleration[u/s ²]	命令加速度与实际加速度

名称	功能
Torque[Nm]	命令力矩与实际力矩

■ ② 轴通讯状态及轴状态

名称	功能
Status	轴状态
Communication	通讯状态

■ ③ 轴启动：设定轴启动/关闭

■ ④ 清除错误：清除伺服轴错误信息

■ ⑤ 回原点：设定轴回原点

■ ⑥ 轴时动

名称	功能
Distance	移动距离
Velocity	移动速度
Acceleration	加速度
Deceleration	减速度
Jerk	加速度的变化率（跃度）

■ ⑦ 读写轴参数：设定与读取主控轴参数，如需读取或修改伺服轴对象字典（Object Dictionary）需以下列输入

- 以读取及写入伺服对象字典的 0x6098 此参数

16#1609800

1 = 为固定码

6098 = 为读写的参数

00 = 为此参数的 sub

1. 将 0x1609800 转为 10 进位为 23,107,584
2. 以 23,107,584 转为-23,107,584
3. 再将-23,107,584 输入至 Parameter 中就可读取伺服的参数 0x6098

7.4 运动控制装置

7.4.1 轴设定

运动控制装置主要用于设置轴参数。在多数的应用中，您可以在DIADesigner-AX软件内设定轴参数，DIADesigner-AX软件对于用户来说是方便的编辑环境，在DIADesigner-AX编辑环境中，运动控制的轴参数设置被定义成其专属数据结构，在其结构下是由多个相关参数集合在一起。

7.4.2 轴介绍

在运动控制系统中，将运动控制的对象称为轴。轴中含有实际伺服驱动器、编码器和虚拟伺服驱动器。下表中列出目前所有轴类型：

类别	功能
定位轴*1	通过 EtherCAT 通讯执行基本定位控制如绝对寻址、相对定位...等功能。
速度轴*1	通过 EtherCAT 通讯执行速度控制与力矩控制（此模式为 CIA 402 Velocity Mode）。
同步轴*2	通过 EtherCAT 通讯执行伺服控制，除基本定位控制外，还包含同步运动如电子凸轮...等功能。
伺服轴	通过 EtherCAT 通讯进行实体伺服控制
脉冲型伺服轴	通过脉冲方式进行实体伺服控制
虚拟轴	无连接实体伺服，仅在程序中被使用，可执行运动控制命令。
编码器轴	使用实体编码器（SSI 编码器、增量型编码器）作为信号来源
虚拟编码器轴	无连接编码器，仅在程序中被使用。

注 1：

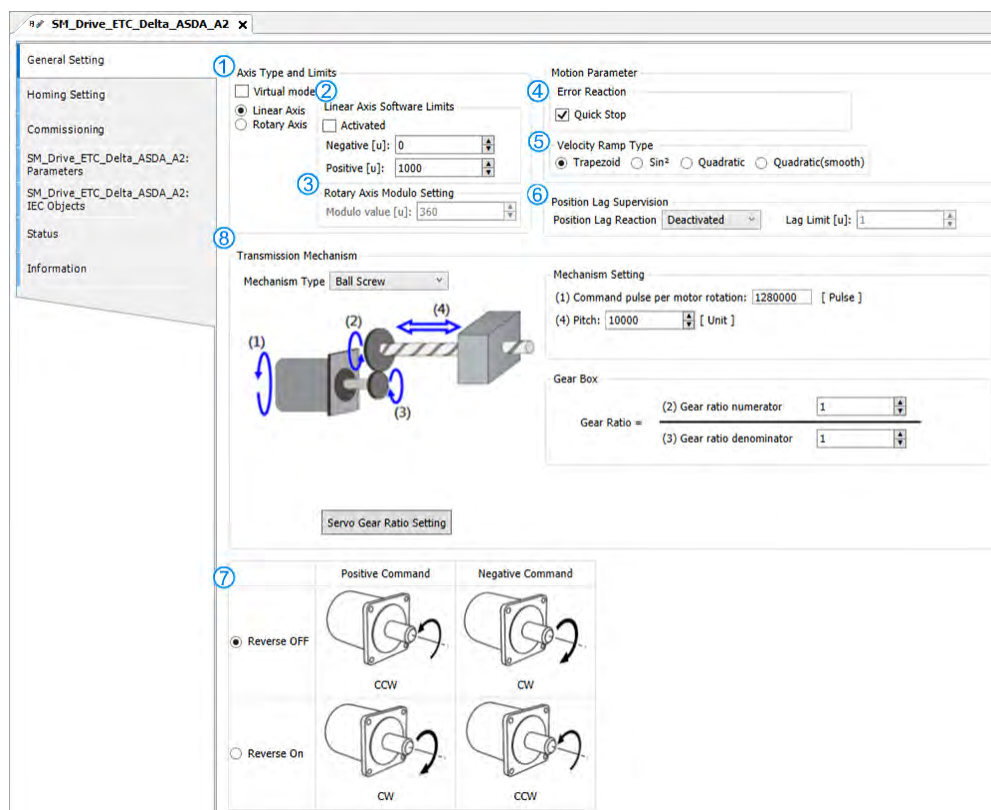
- 定位轴与速度轴需搭配 DL_MotionControlLight 函式库。
- 当 AX-364EL 使用 EtherCAT 轴数，超过 64 轴且 Soft Motion 版本在 V4.7.0.0 以下，需修改 Library 的 IODrvEtherCat → ETC_Parameter 的 MAX_MAILBOX_CHANNELS 与 MAX_SDO_Channels 参数，改为 128。

注 2：同步轴需搭配 DL_MotionControl、SM3_Basic 函式库。

7.4.2.1 轴参数介绍

当建立伺服轴后，该轴会产生对应的轴参数，下表列出其相对说明：

- 同步轴



- ① Axis Type and Limits

名称	功能
Virtual	启动虚轴
Linear Axis / Rotary Axis	设定直线轴或旋转轴

- ② Linear Axis Software Limits

名称	功能
Activated	启动软件极限（仅支持直线轴）
Negative[u]	反转软件极限
Positive[u]	正转软件极限

■ ③ Rotary Axis Modulo Setting

名称	功能
Modulo Value[u]	设定一圈范围 (仅支持旋转轴)

■ ④ Error Reaction

名称	功能
Quick Stop	轴紧急停止
Deceleration[u/s ²]	轴减速停止 (未启动 Quick Stop 才有效)

■ ⑤ Velocity Ramp Type

名称	功能
Trapezoid/Sin ² /Quadratic/ Quadratic (Smooth)	设定轴运动曲线

■ ⑥ Position Lag Supervision

名称	功能
Positon Lag Reaction	设定溢位后轴的行为
Lag Limit [u]	设定溢位数值

■ ⑦ Positive / Negative Command

名称	功能
Reverse OFF / ON	设定正反向命令

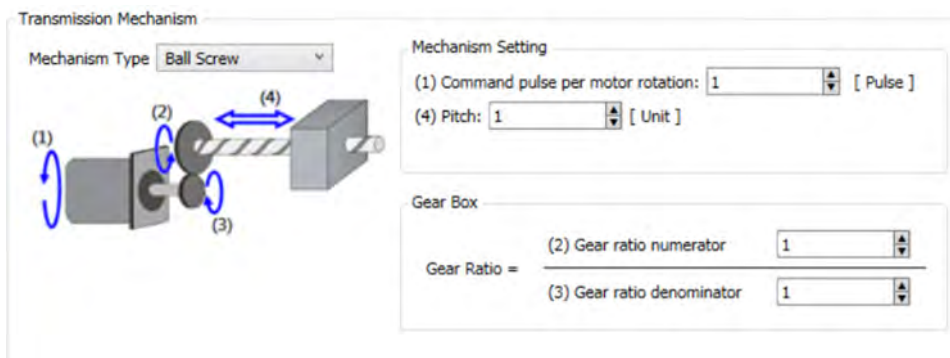
■ ⑧ Transmission Mechanism

◆ Servo Gear Ratio Setting

名称	功能
Unit Numerator	伺服分子
Unit Denominator	伺服分母

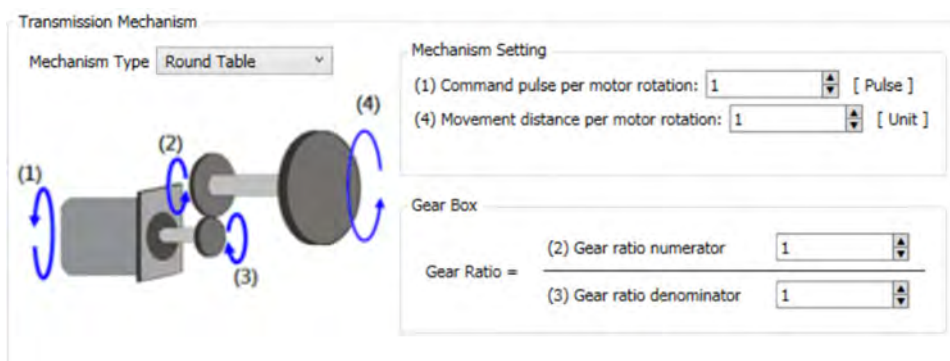
针对不同架构以下列介绍：

◆ Ball Screw



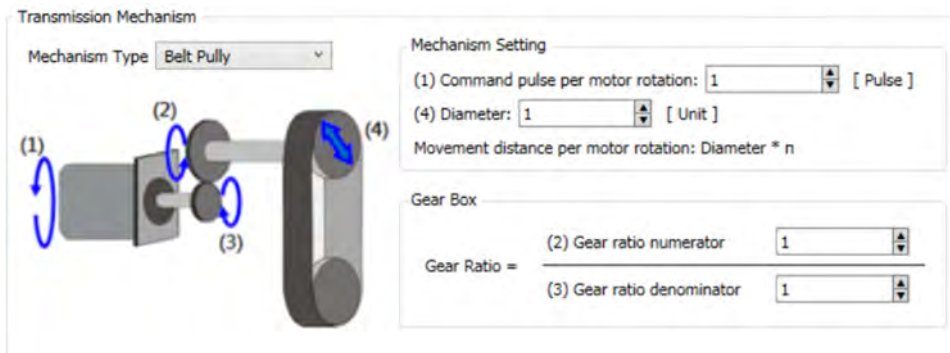
名称	功能
(1) Command Pulse per motor rotation	电机每转的指令脉冲
(4) Pitch	螺杆间距
(2) Gear ratio numerator	电子齿轮比分子
(3) Gear ratio denominator	电子齿轮比分母

◆ Round Table



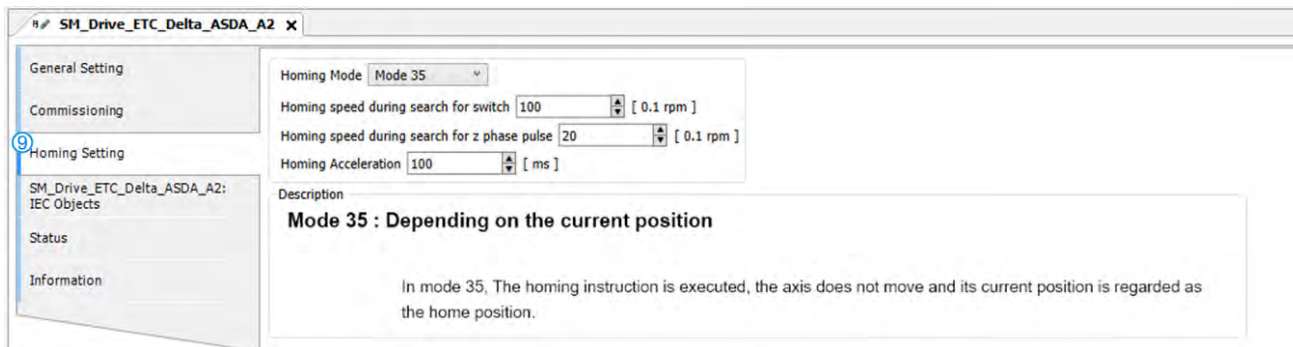
名称	功能
(1) Command Pulse per motor rotation	电机每转的指令脉冲
(4) Movement distance per motor rotation	电机每转的移动距离
(2) Gear ratio numerator	电子齿轮比分子
(3) Gear ratio denominator	电子齿轮比分母

◆ Belt Pully



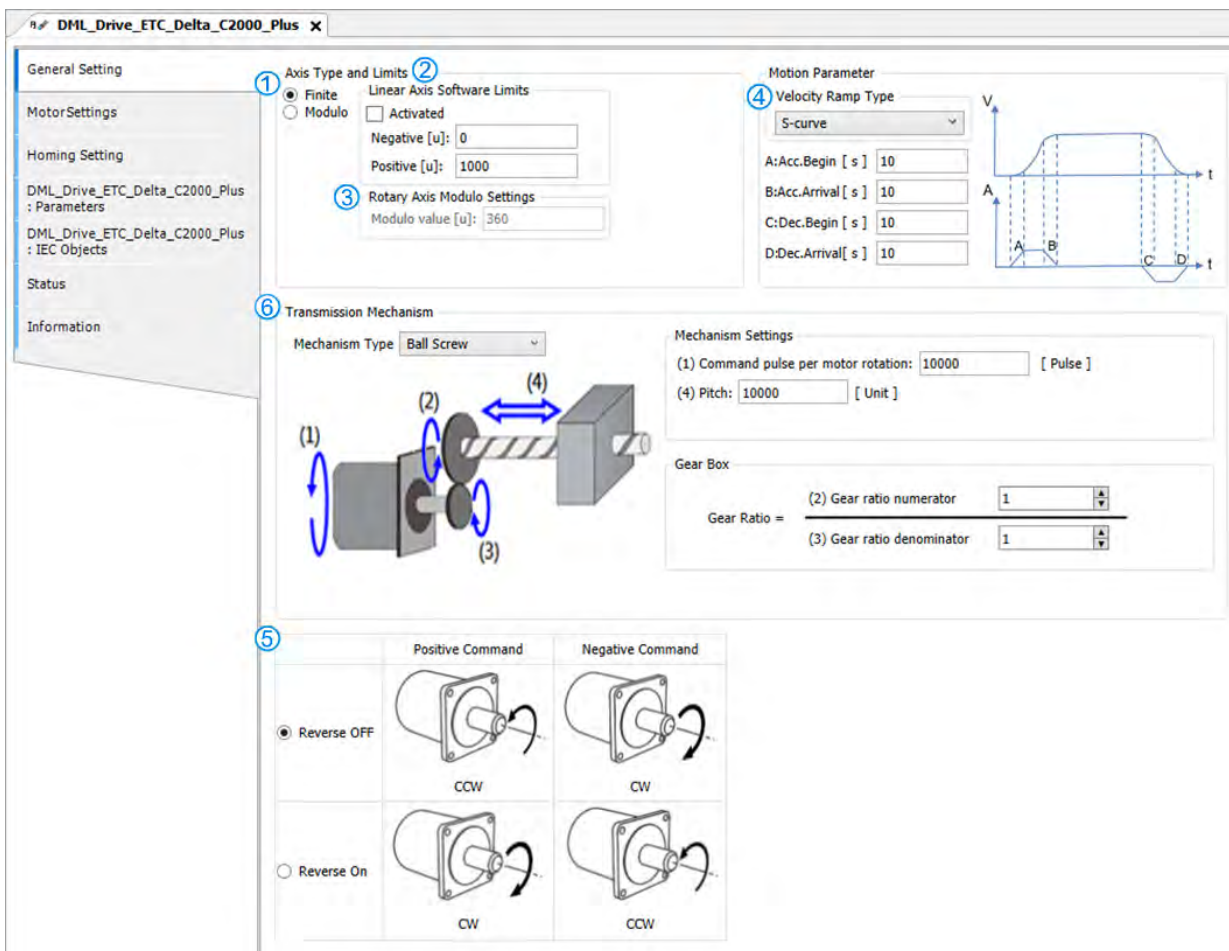
名称	功能
(1) Command Pulse per motor rotation	电机每转的指令脉冲
(4) Diameter (Movement distance motor rotation : Diameter *n)	直径 (电机每转移动距离 : 直径*n)
(2) Gear ratio numerator	电子齿轮比分子
(3) Gear ratio denominator	电子齿轮比分母

■ ⑨ Homing Setting



名称	功能
Homing Mode	设定原点模式
Homing Speed during search for switch	设定寻原点开关速度
Homing Speed during search for z phase pulse	设定寻 Z 相速度
Homing Acceleration	设定原点加速度

- 定位轴
 - ❖ 定位轴 - 台达伺服设定



■ ①Axis Type and Limits

名称	功能
Linear Axis / Rotary Axis	设定直线轴或旋转轴

■ ②Linear Axis Software Limits

名称	功能
Activated	启动软件极限 (仅支持直线轴)
Negative[u]	反转软件极限
Positive[u]	正转软件极限

■ ③Rotary Axis Modulo Setting

名称	功能
Modulo Value[u]	设定一圈范围 (仅支持旋转轴)

■ ④Velocity Ramp Type

❖ 台达伺服设定

名称	功能
Trapezoid/Sin ²	设定轴运动曲线

❖ 台达变频器设定

名称	功能
Trapezoid/S-Curve	设定轴运动曲线
A : Acc.Begin	S 加速起始时间设定 1 (s)
B : Acc.Arrival	S 加速到达时间设定 2 (s)
C : Dec.Begin	S 减速起始时间设定 1 (s)
D : Dec.Arrival	S 减速到达时间设定 2 (s)

■ ⑤Positive / Negative Command

名称	功能
Reverse OFF / ON	设定正反向命令

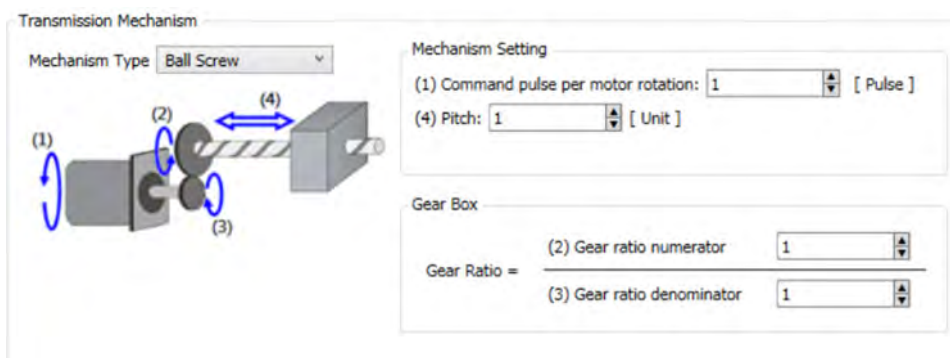
■ ⑥Transmission Mechanism

◆ Servo Gear Ratio Setting

名称	功能
Unit Numerator	伺服分子
Unit Denominator	伺服分母

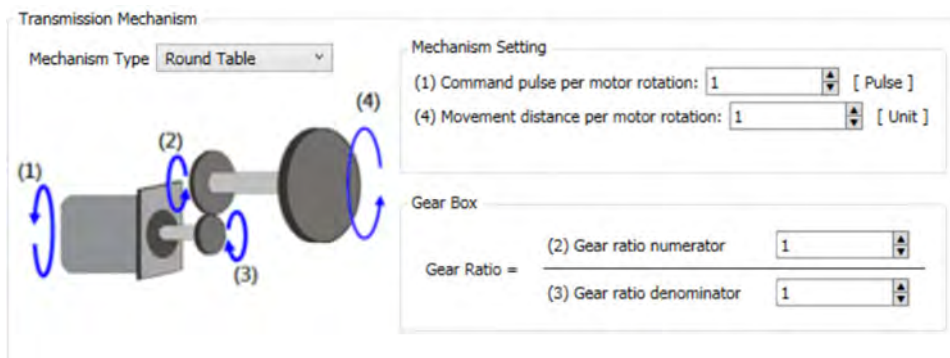
针对不同架构以下列介绍：

◆ Ball Screw



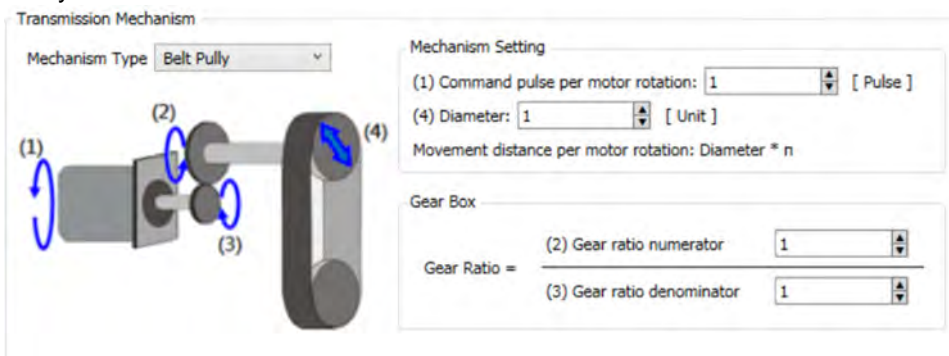
名称	功能
(1) Command Pulse per motor rotation	电机每转的指令脉冲
(4) Pitch	螺杆间距
(2) Gear ratio numerator	电子齿轮比分子
(3) Gear ratio denominator	电子齿轮比分母

◆ Round Table



名称	功能
(4) Command Pulse per motor rotation	电机每转的指令脉冲
(4) Movement distance per motor rotation	电机每转的移动距离
(2) Gear ratio numerator	电子齿轮比分子
(3) Gear ratio denominator	电子齿轮比分母

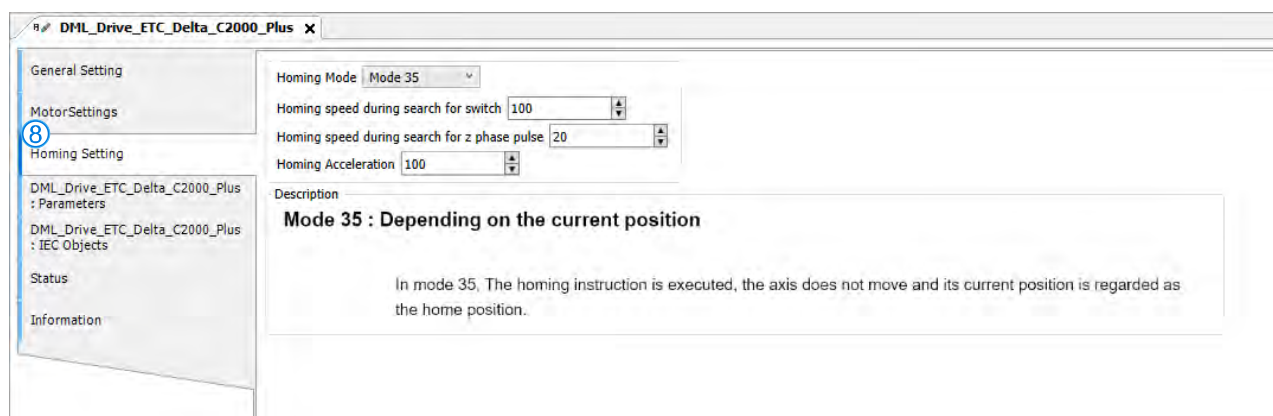
◆ Belt Pully



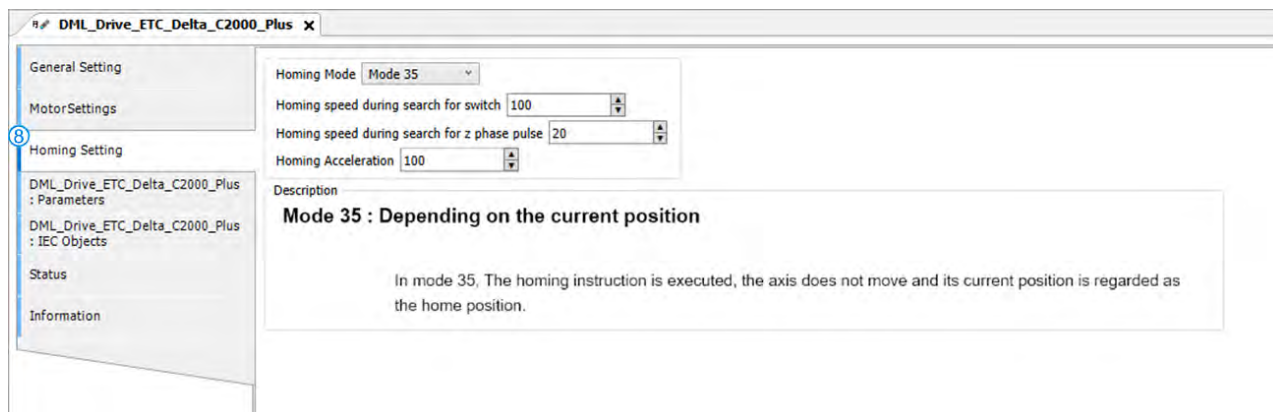
名称	功能
(1) Command Pulse per motor rotation	电机每转的指令脉冲
(4) Diameter (Movement distance motor rotation : Diameter * n)	直径 (电机每转移动距离 : 直径*n)
(2) Gear ratio numerator	电子齿轮比分子
(3) Gear ratio denominator	电子齿轮比分母

■ ⑧Homing Setting

❖ 台达伺服设定



❖ 台达变频器设定 (支持机种 C2000Plus、CH2000、MH300、MS300 ; DDF V1.0.1.0 (含) 以上支持此接口)



名称	功能
Homing Mode	设定原点模式
Homing Speed during search for switch	设定寻原点开关速度
Homing Speed during search for z phase pulse	设定寻 Z 相速度
Homing Acceleration	设定原点加速度

■ ⑦ Motor Settings - 台达变频器设定

The screenshot shows the 'Motor Settings' configuration window for a Delta C2000 Plus drive. The left sidebar contains a tree view with the following items: General Setting, MotorSettings (highlighted with a blue circle and the number 7), Homing Setting, DML_Drive_ETC_Delta_C2000_Plus : Parameters, DML_Drive_ETC_Delta_C2000_Plus : IEC Objects, Status, and Information. The main configuration area is divided into two sections: Motor Settings and Encoder Settings.

Motor Settings:

- Motor Selection: IM
- Pole Number: 4
- Maximum Operation Frequency: 100.00 [Hz]
- Motor Rated Frequency: 60.00 [Hz]
- Rated Current: 25.00 [A]
- Rated Power: 5.50 [kW]
- Rated Voltage: 220.0 [V]
- Rated Speed: 1200 [RPM]

Encoder Settings:

- Encoder Connected: Yes No
- Encoder Type Selection: ABZ Pulse
- Encoder Input Type Setting: FWD A Leads B
- Encoder Pulses Per Revolution: 2500 [ppr]

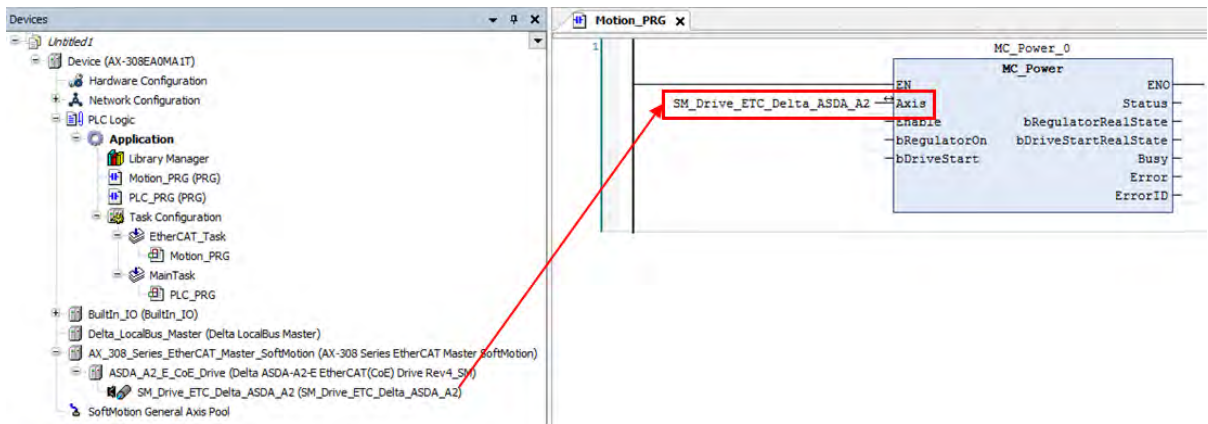
名称	功能
Motor Selection	设定电机类型
Pole Number	设定电机极数
Maximum Operation Frequency	设定电机最高操作频率
Motor Rated Frequency	设定电机基底频率
Rated Current	设定电机满载电流
Rated Power	设定电机额定功率

名称	功能
Rated Voltage	设定电机基底电压
Rated Speed	设定电机额定转速
Encoder Connected	设定是否启动 Encoder
Encoder Type Selection	设定编码器种类选择
Encoder Input Type Setting	设定编码器输入型式
Encoder Pulses Per Revolution	设定编码器每转脉冲数

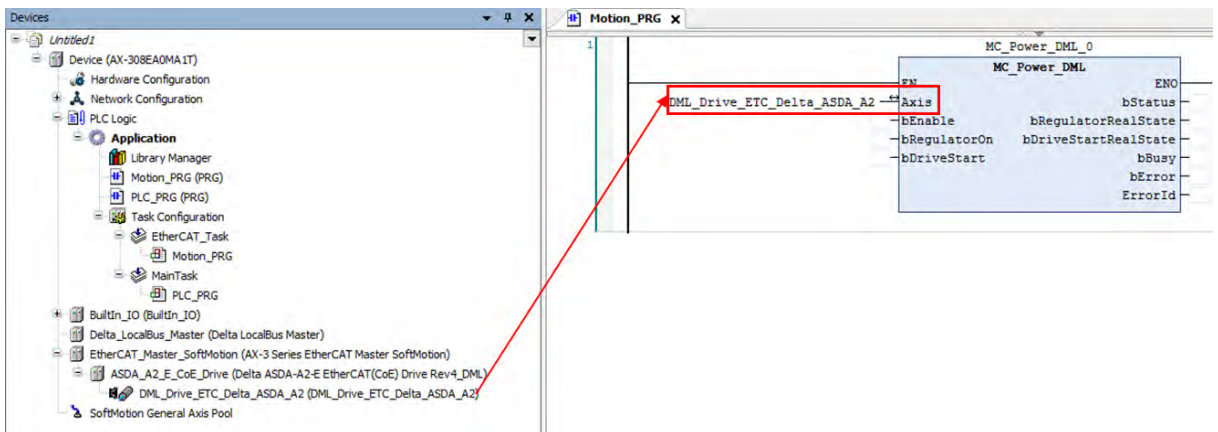
7.4.2.2 程序中使用轴

当项目中新增伺服轴后，系统会自动产生其伺服轴名称（用户也可自行修改其名称），并将名称输入至功能块。

- 同步轴

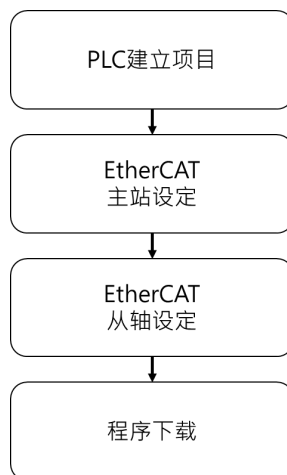


- 定位轴

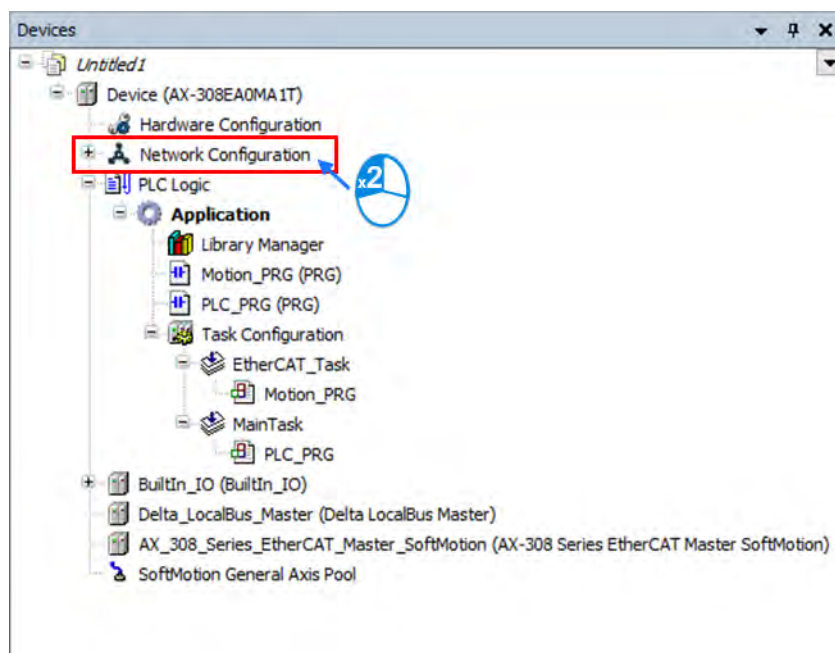



7.4.3 单轴配置流程与设定步骤

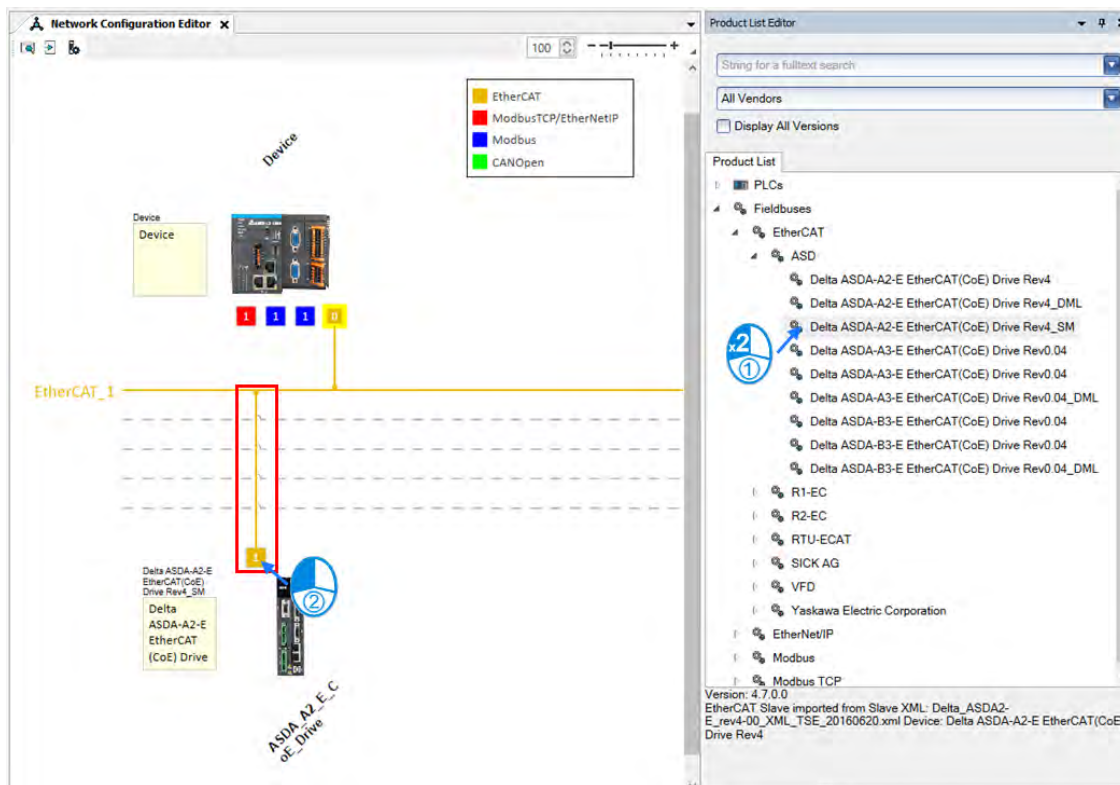
- 轴设定流程如下图所示，PLC 项目建立细部流程请参考第 7.2 节运动项目建立流程。



- 开启项目后设定 EtherCAT，首先单击「Network Configuration」。



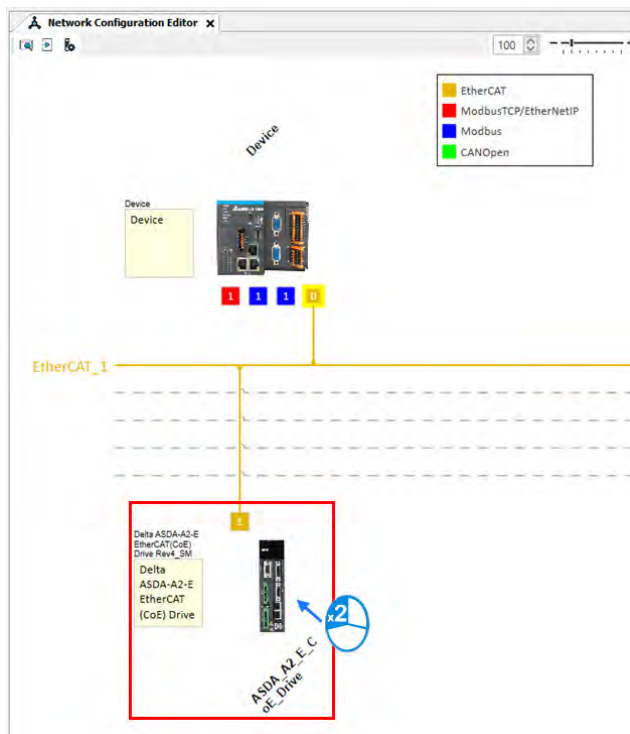
- 进入 Network Configuration 画面后单击「Delta ASDA-A2-E EtherCAT(CoE) Drive Rev4_SM」¹→ 再将  联机至上方。



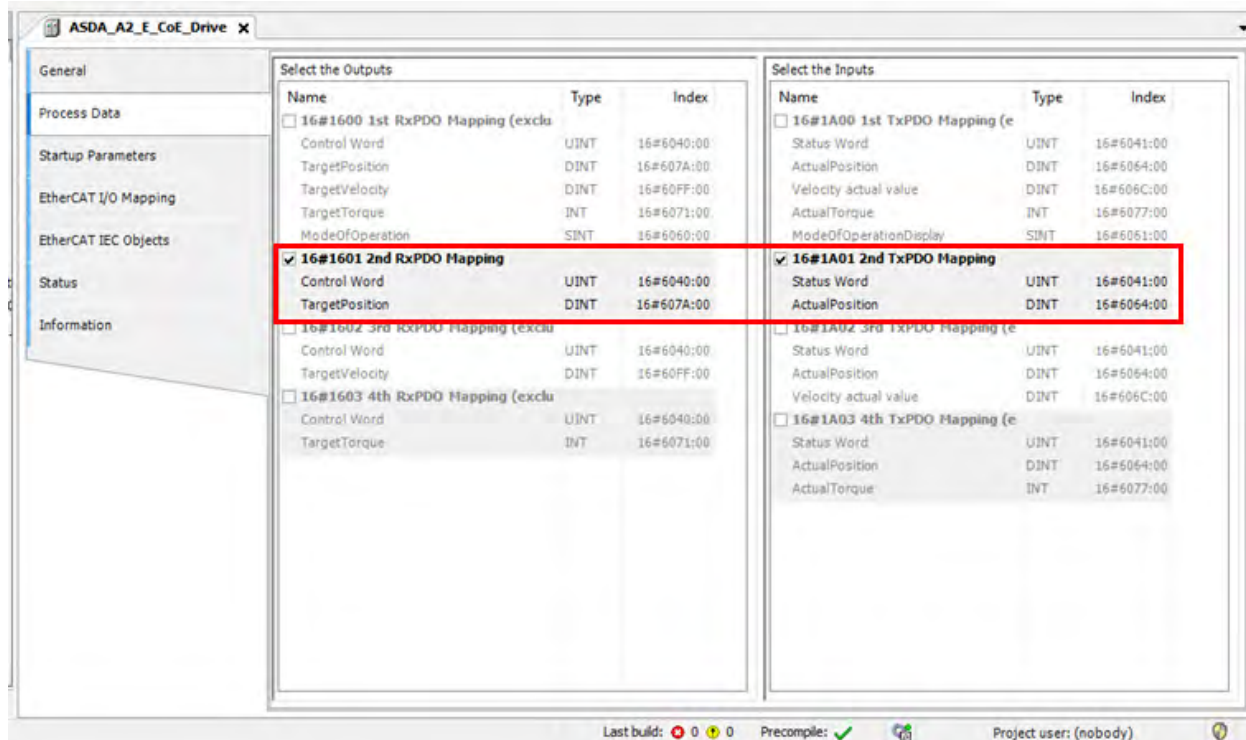
注：

*1 Delta ASDA-A2-E EtherCAT(CoE) Drive Rev4_SM 为同步轴，如需新增定位轴请选择 Delta ASDA-A2-E EtherCAT(CoE) Drive Rev4_DML 后续动作皆为相同。

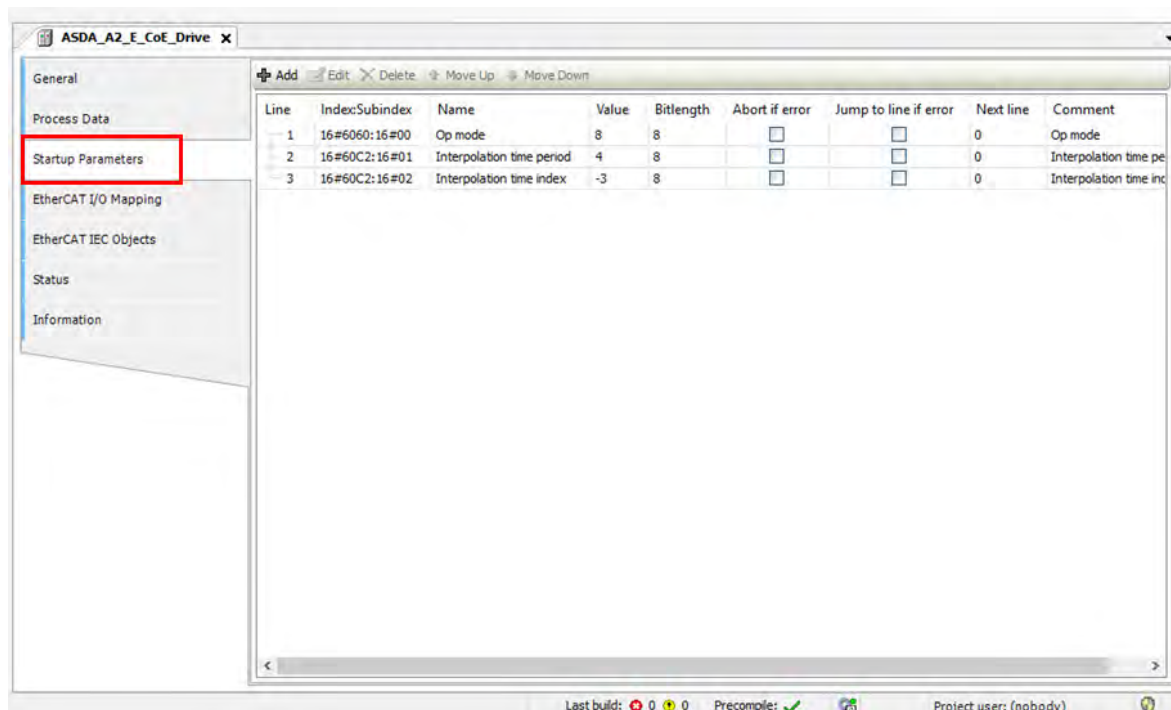
■ 完成轴建立后，请使用鼠标左键双击欲设定之伺服从站。



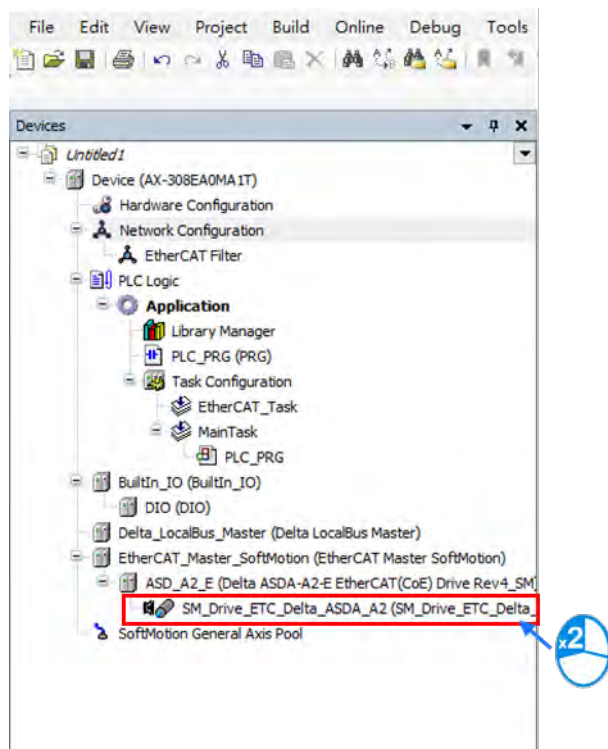
- 请切换页面至「Process Data」设定其 PDO 映射组别。台达 ASDA-A2 默认组别为第 2 组，大部份运动功能块使第 2 组即可正常运行，若功能块需额外选定组别/新增 PDO 参数，请参考 **AX 系列运动控制指令手册**各功能块内容说明。



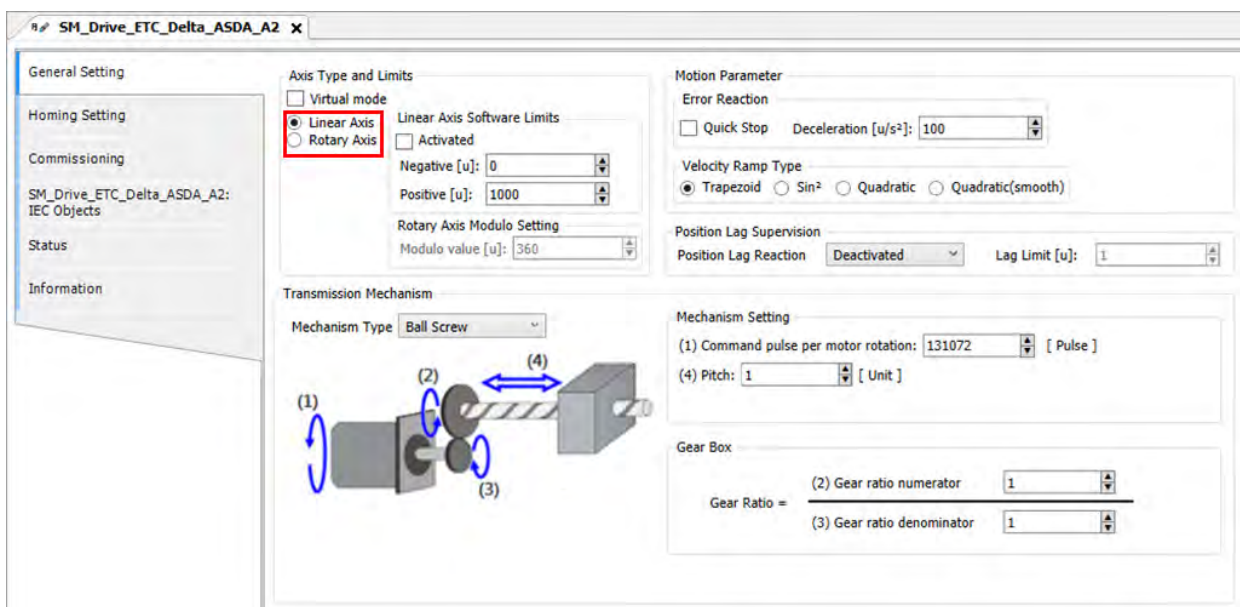
- EtherCAT 通讯初始化设定
若伺服从站在每次初始化完成后，需针对所需的对象辞典 (Object Dictionary) 写入固定值，可在「Startup Parameters」页面中进行设定。



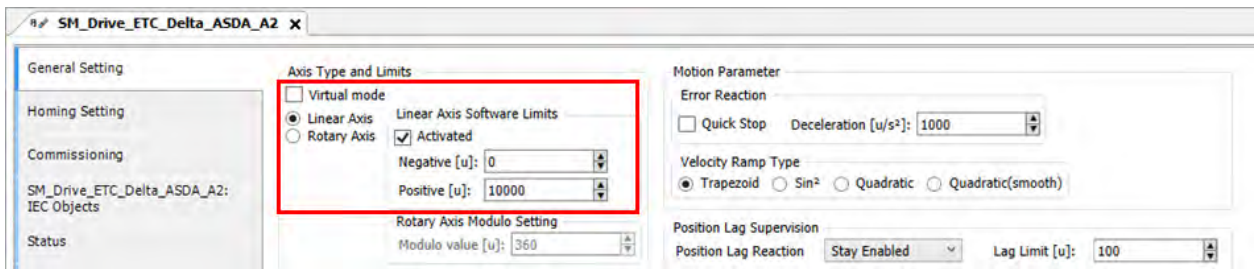
- 将轴通讯相关设定完毕后再次单击「SM_Drive_ETC_Delta_ASDA_A2」左键两下。



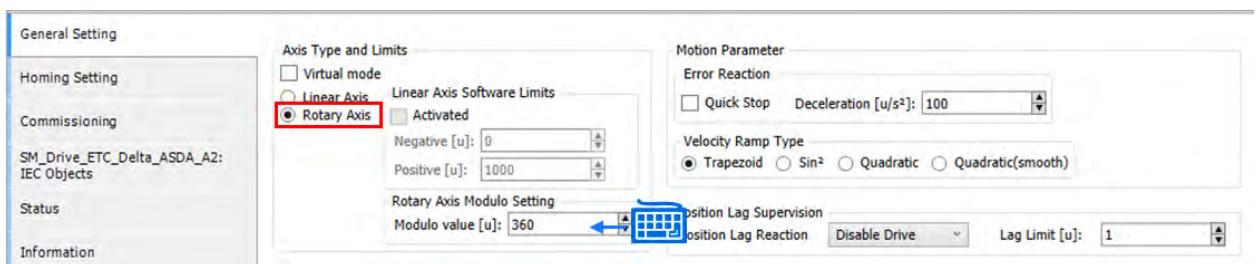
- 轴设定画面
轴模式设定 · 旋转轴「Rotary Axis」· 直线轴「Liner Axis」。



- 设定直线轴软件极限「Software limits」，当勾选 Activated 后会启动软件极限，「Negative」为反极限、「Positive」为正极限。



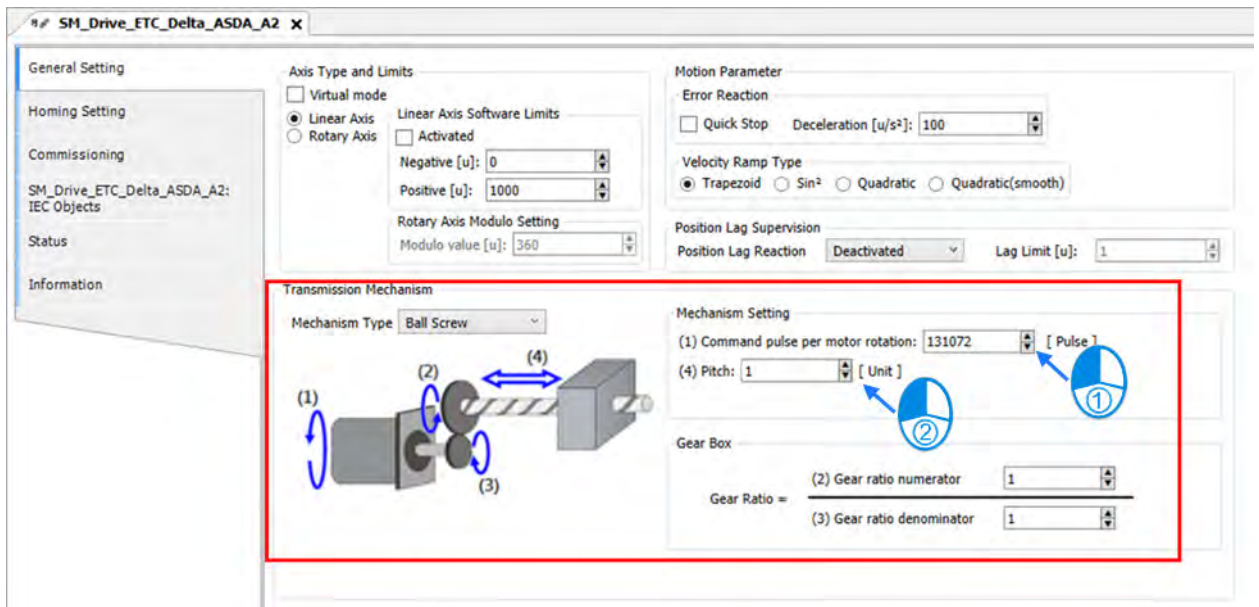
- 设定旋转轴后须定义旋转范围，请在「Modulo settings」中设定旋转轴范围「Modulo value」。



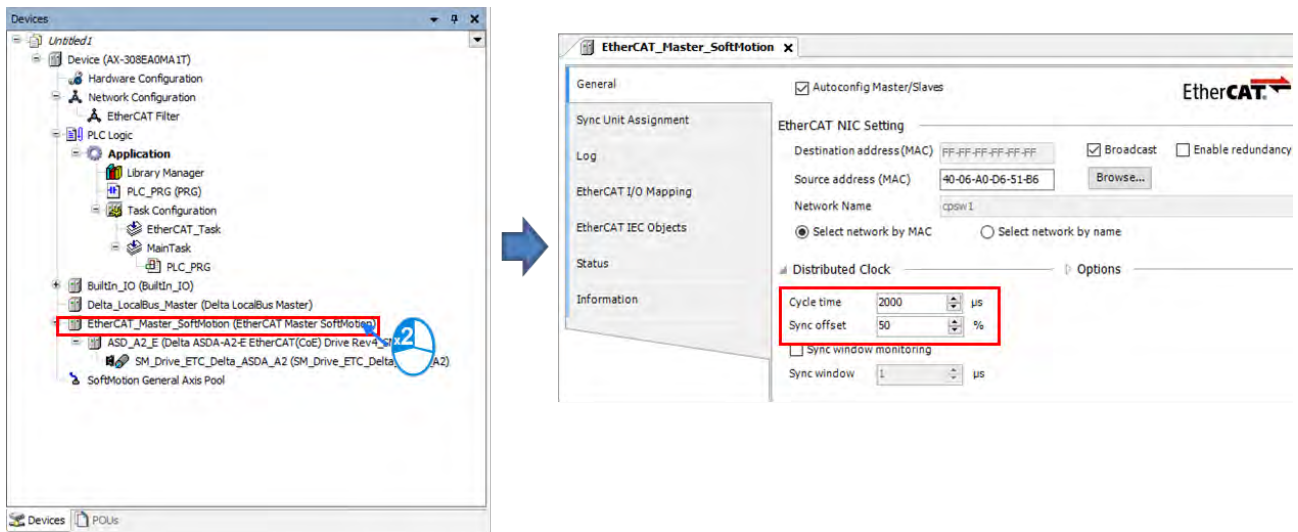
- Scaling/Mapping 画面

设定电机一圈脉冲量「Command pulse per motor rotation」

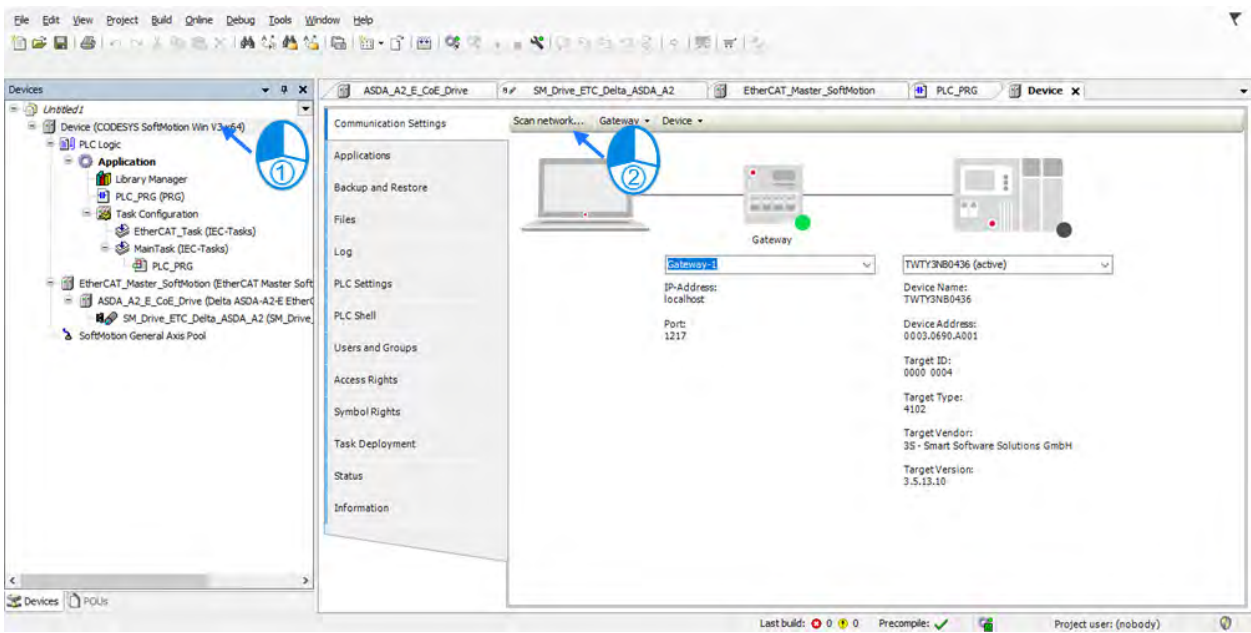
设定电机一圈移动距离「Pitch」



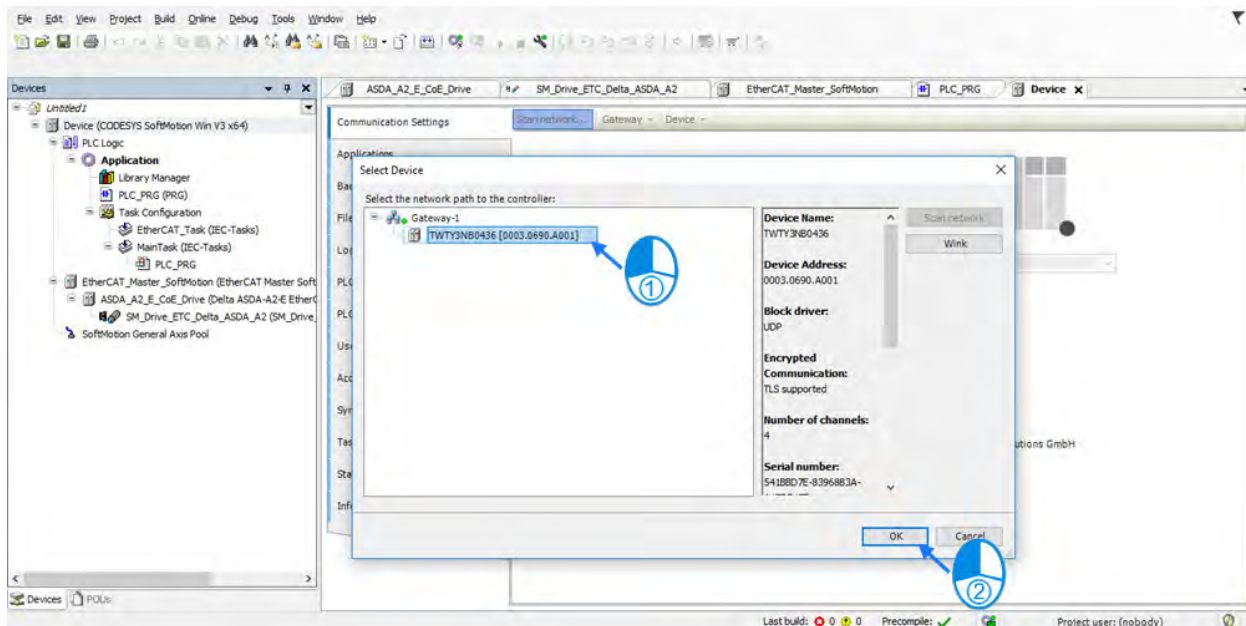
- 设定 EtherCAT 通讯周期，单击「EtherCAT_Master_SoftMotion」后再 Cycle time = 2000，Sync offset = 50。(AX-308 最大支持 8 轴，建议设定时间为 Cycle time = 2000，Sync offset = 50)



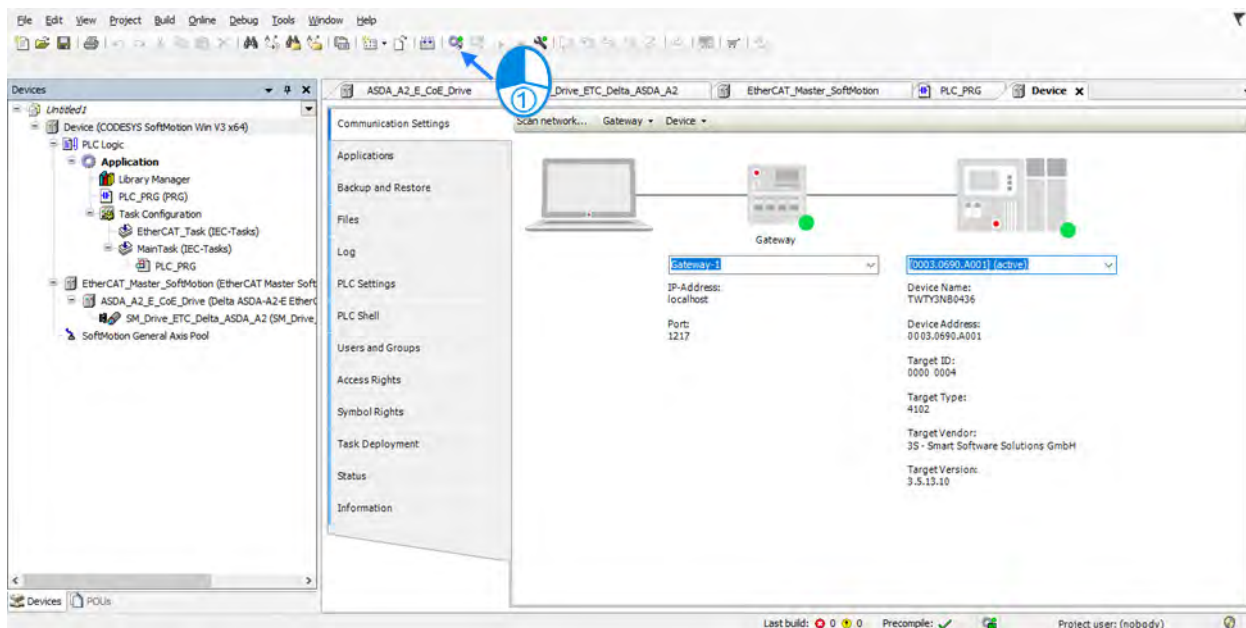
- 扫描 PLC 控制器



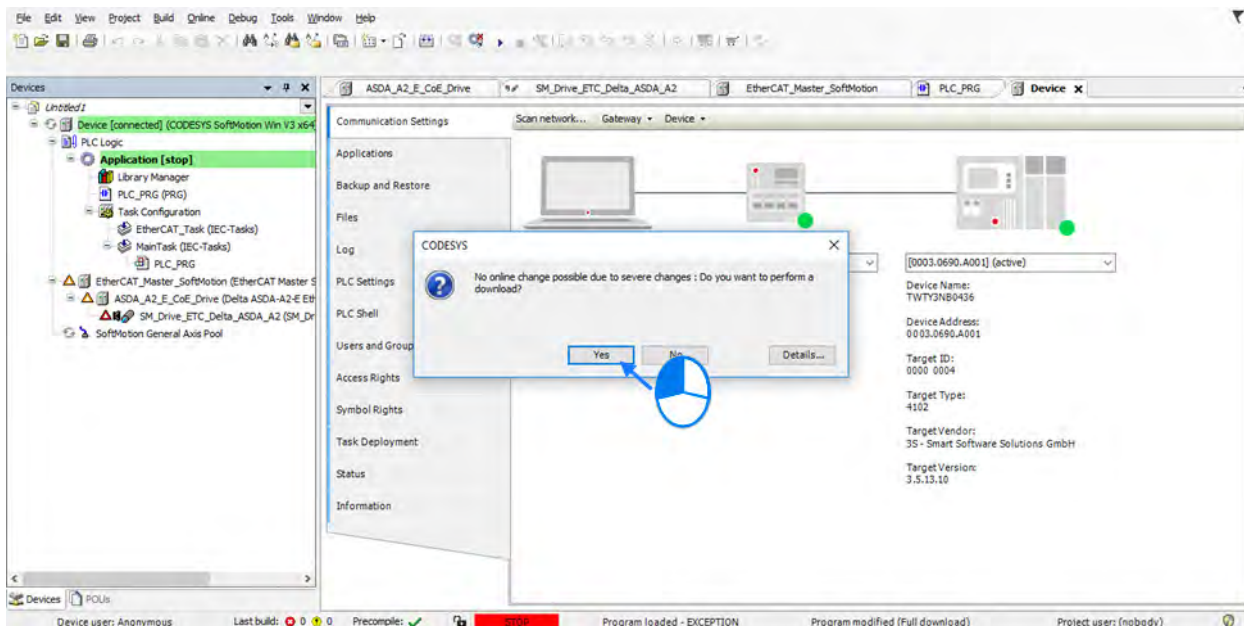
- 新增扫到的 PLC 控制器后单击 OK



- 新增完 PLC 后会出现绿灯符号确认已联机上，再单击 Login。



- 单击完 Login 后会出现是否下载程序，再单击 YES。



7.4.4 轴组设定

同时执行多轴并且直线插补与螺旋插补功能时就需要使用轴组运动，需通过DIADesigner-AX将轴设定为轴组。

控制轴数	直线插补控制最大轴数	6 轴
	圆弧插补控制轴数	6 轴 (3 轴跟随轴)

7.4.4.1 轴组参数介绍

轴组运动所使用到的参数如下：

① Kinematic Configuration

Axis X: ...

Axis Y: ...

Axis Z: ...

Following Axis

Axis A: ...

Axis B: ...

Axis C: ...

Note

$Following\ Ratio = \frac{Target\ Position\ of\ Following\ Axis}{Target\ Position\ of\ Axis\ Group}$

② Motion Parameter

RampType: S Curve

Max Velocity Limit: (user unit)/s

Max Acceleration Limit: (user unit)/s²

Max Deceleration Limit: (user unit)/s²

Max Jerk Limit (Reserved): (user unit)/s³

③ Tasks

Bus Task: ...

■ ① Kinematic

名称	功能
Axis X ^{*1}	轴组中 X 轴
Axis Y ^{*1}	轴组中 Y 轴
Axis Z ^{*1}	轴组中 Z 轴
Axis A ^{*1}	轴组中 A 轴
Axis B ^{*1}	轴组中 B 轴
Axis C ^{*1}	轴组中 C 轴

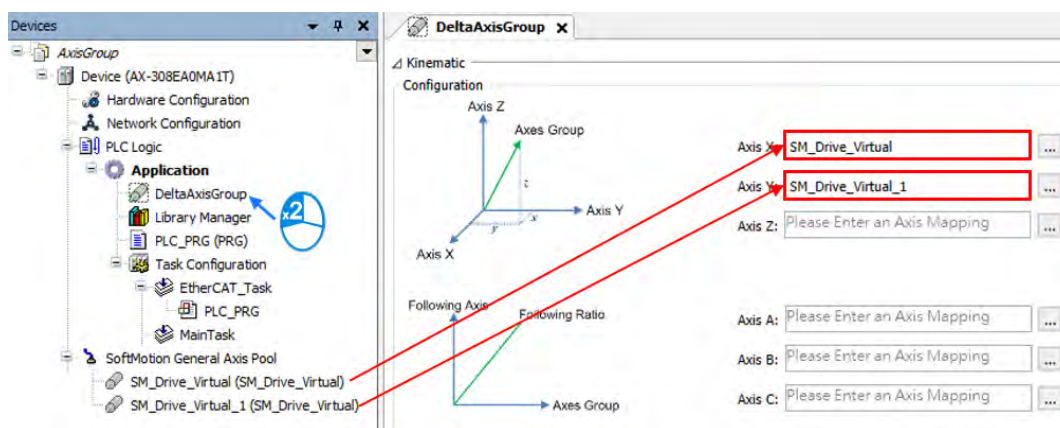
■ ② Motion Parameter

名称	功能
Ramp Type ^{*2}	速度曲线类型
Max Velocity Limit ^{*3}	轴组最大速度
Max Acceleration Limit ^{*3}	轴组最大加速度
Max Deceleration Limit ^{*3}	轴组最大减速度
Max Jerk Limit (Reserved) ^{*3}	轴组最大加速度的变化率 (跃度) (此功能保留)

■ ③ Tasks

名称	功能
Bus Task	设定轴组更新 Task

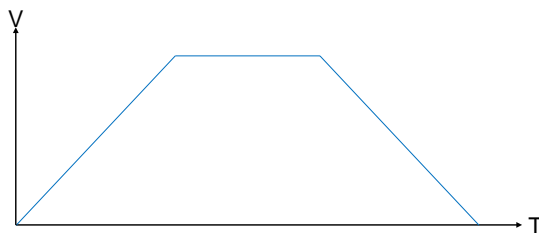
注 1 : Axis X ~ Axis C : 输入单轴名称



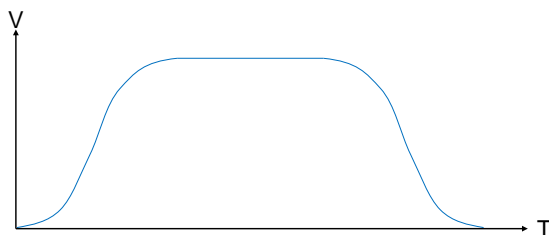
7

注 2：Ramp Type：模式有 Trapezoid 与 S Curve 两种曲线，曲线显示为以下两种：

- Trapezoid



- S Curve



注 3：

- Max Velocity Limit：当轴组运动速度超过设定参数就会报错。
- Max Acceleration Limit：当轴组运动加速度超过设定参数就会报错。
- Max Deceleration Limit：当轴组运动减速度超过设定参数就会报错。

7.4.4.2 程序中使用轴组

程序中使用轴组功能块前，需在项目树状中新增轴组节点并设定该轴组需使用到的单轴名称，设定完成后请将该轴组的节点名称指向至轴组各功能块的AxisGroup引脚。

```

1  PROGRAM POU
2  VAR
3      DMC_GroupEnable_0: DMC_GroupEnable;
4  END_VAR
5

```

DMC_GroupEnable_0

DMC_GroupEnable

EN

AxisGroup

bExecute

ENO

bDone

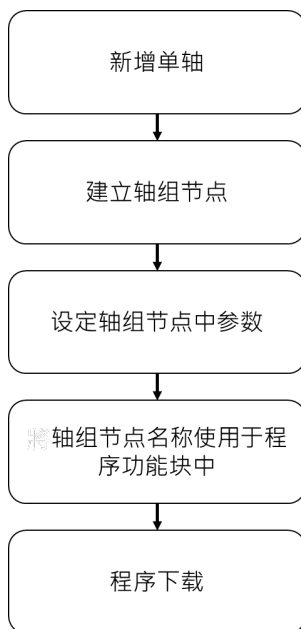
bBusy

bError

ErrorID

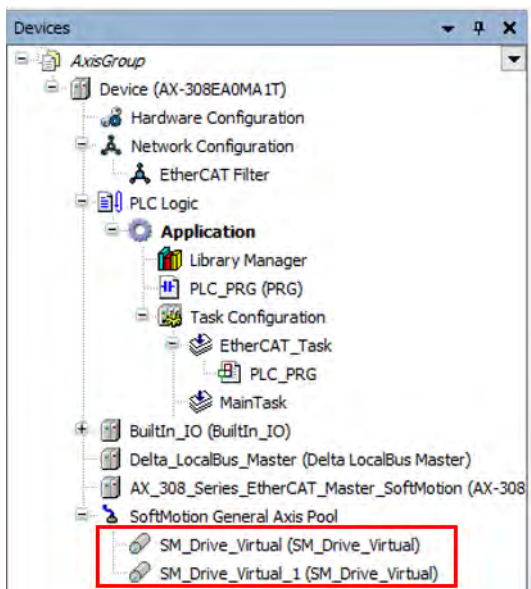
7.4.5 轴组配置流程与设定步骤

- 使用轴组运动需要通过 DIADesigner-AX 设定轴组名称与对应个别轴，轴组建立流程图如下：

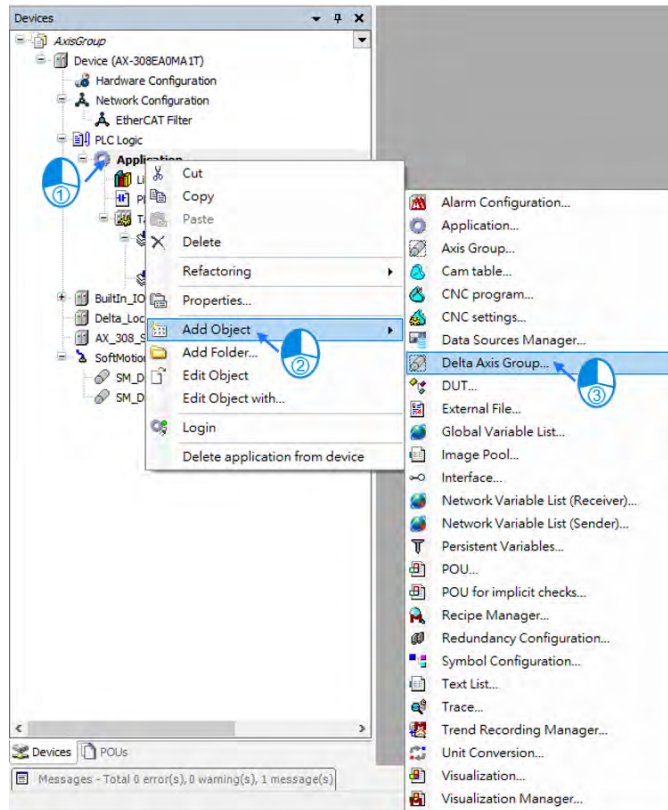


- 程序建立轴组步骤

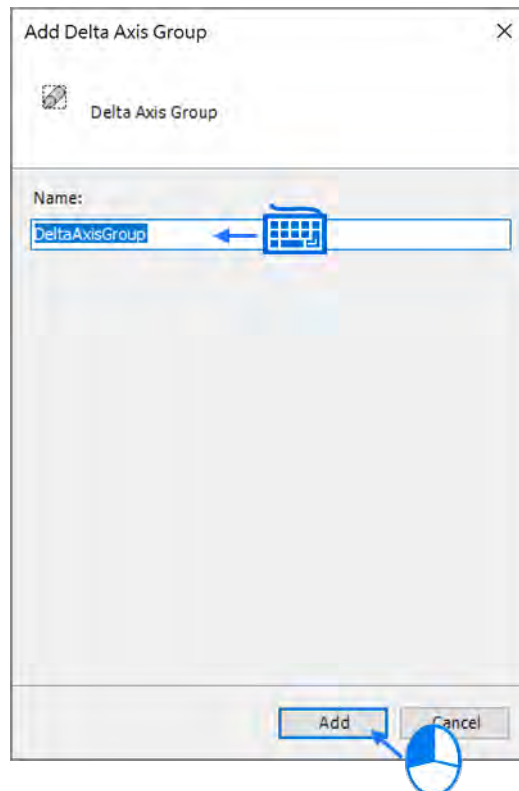
- 新增单轴，此流程以建立 2 虚轴为例。



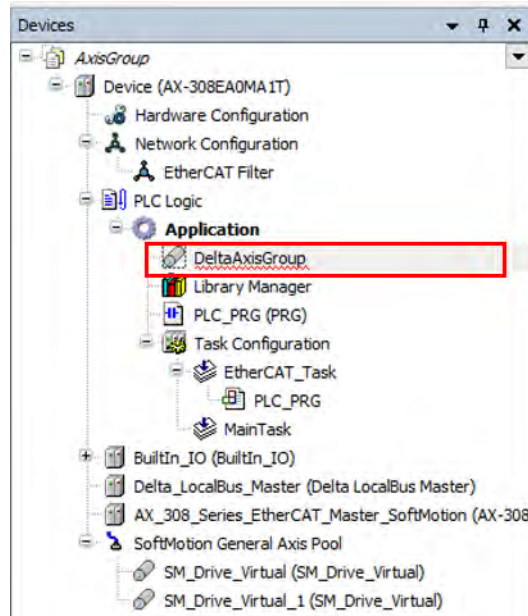
- 建立完成后请于「Application」单击鼠标右键后单击「Add Object」→「Delta Axis Group」。



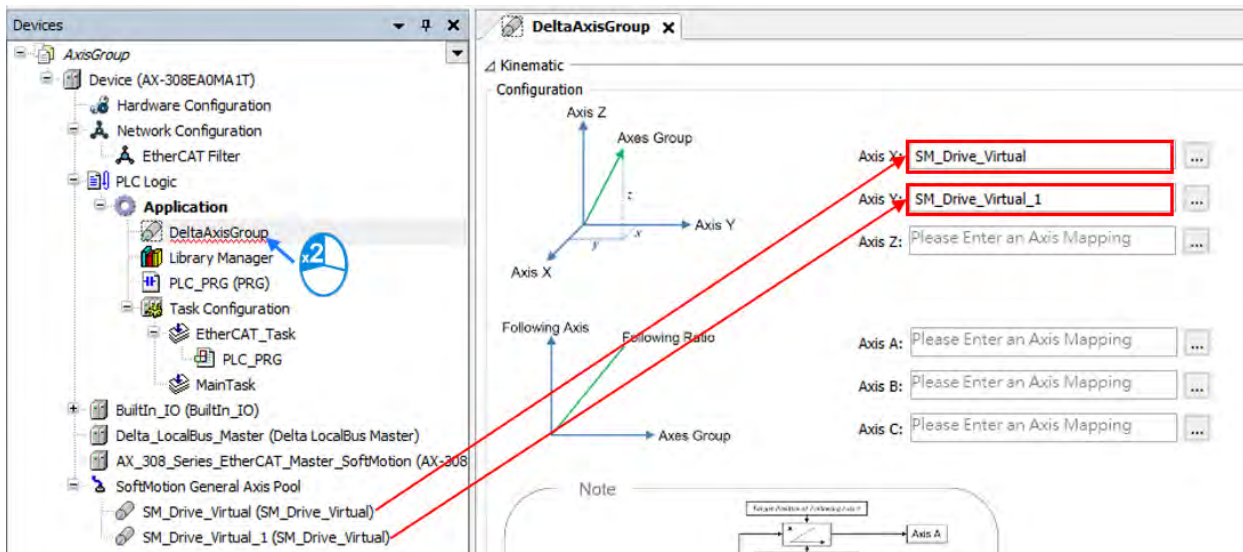
- 在「Add Delta Axis Group」画面设定名称再单击「Add」。



- 此时左边项目树会出现「DeltaAxisGroup」。

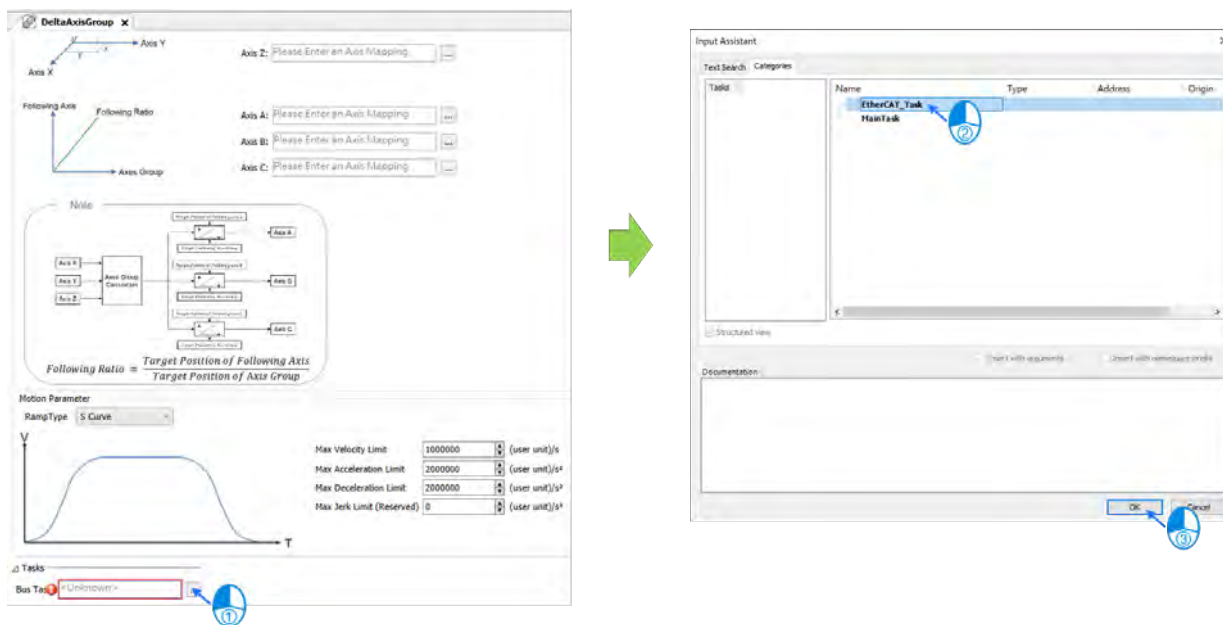


- 单击「DeltaAxisGroup」后将两轴虚轴名称输入至「Axis X」与「Axis Y」。

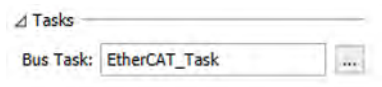


7

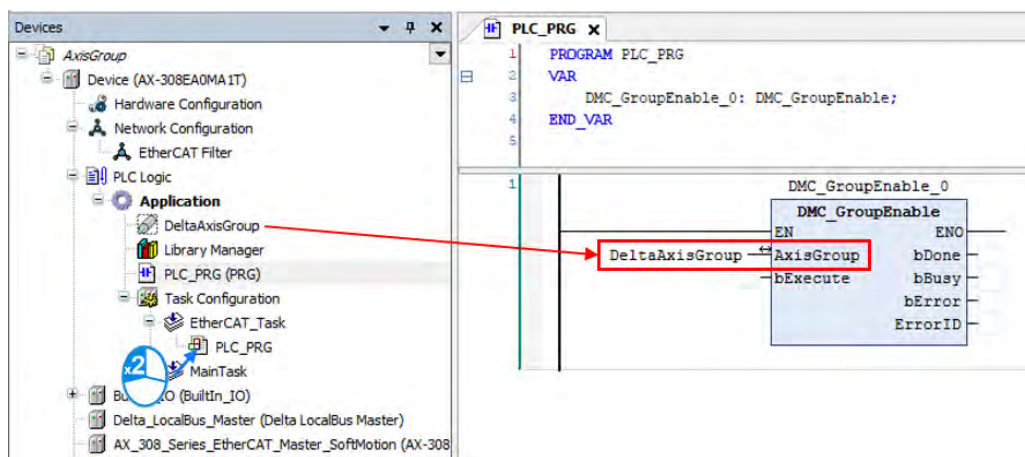
- 将 Axis 名称填写完毕后，再单击「Bus Task」。进入「Input Assistant」画面选择「EtherCAT_Task」后单击「OK」。



- 设定 Task 完毕后，会出现「EtherCAT_Task」。

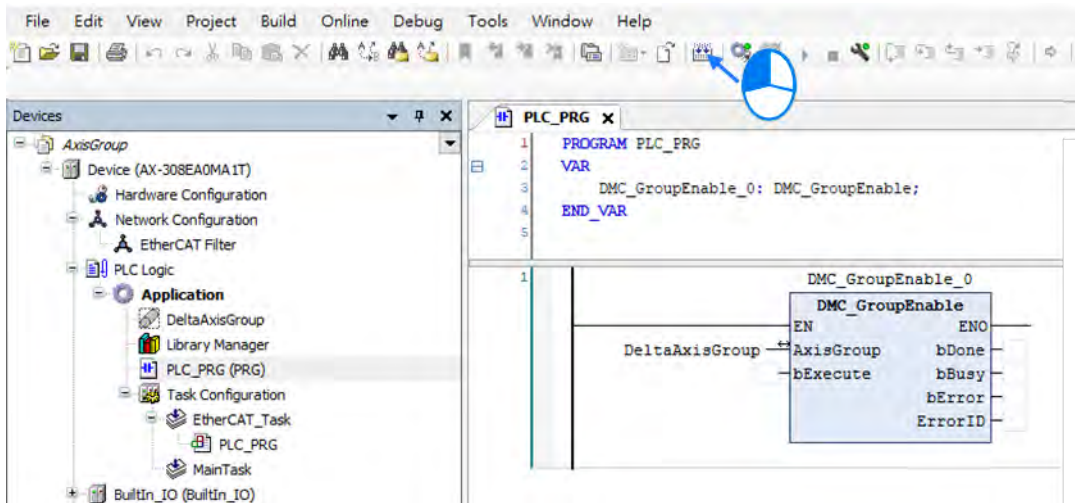


- 在「PLC_PRG」内新增「DMC_GroupEnable」功能块并在「AxisGroup」引脚填写轴组名称。

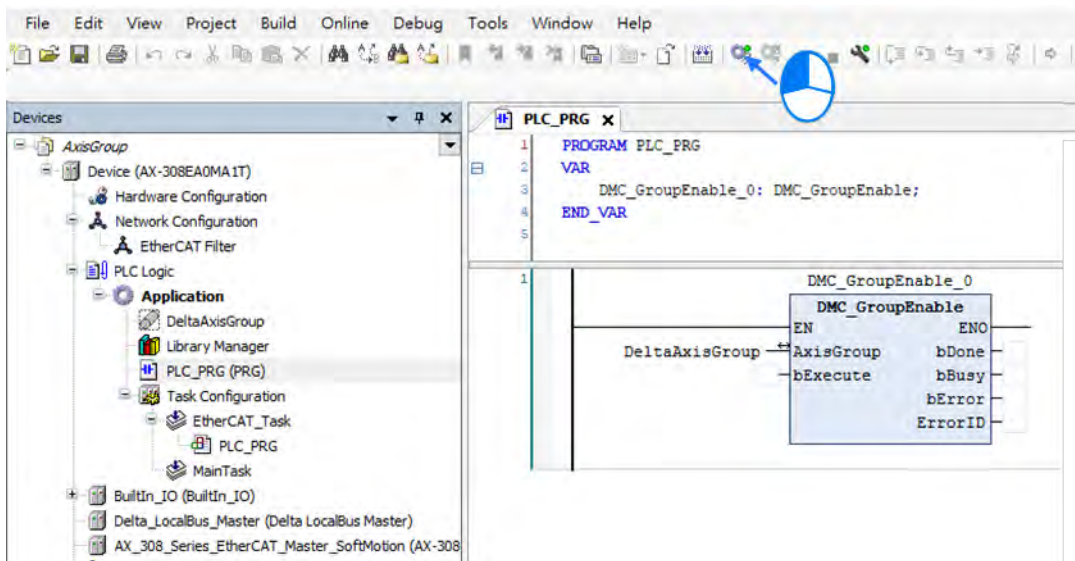


7

- 当程序编写完毕后，单击编译并确认是否有错误。



- 确认编译后若无错误，再单击在线监控就可以下载程序。



7.5 运动轴变量

7.5.1 单轴变数

在软件 DIADesigner-AX 项目树中建立轴后，会自动产生对应的轴参数，轴参数仅供读取使用，轴分成同步轴 (Axis_REF_SM3) 与定位轴 (Axis_REF_DML)，下表为轴参数内容：

● 同步轴 (Axis_REF_SM3)

编号	名称	数据类型	默认值	说明
1000	nAxisState	SMC_AXIS_STATE (INT)	Standstill (3)	显示当前轴状态机
1012	bCommunication	BOOL	FALSE	当 EtherCAT 轴通讯正常时 True，如果断线了为 False
1014	uiDriveInterfaceError	UINT	0	如果 Driver Interface 检测到错误会显示其对应之错误码
1021	wDriveld	WORD	Driver	Fieldbus 号码
1025	fTaskCycle	LREAL	Driver	EtherCAT 任务周期时间
1035	fbeFBError	ARRAY [0..g_SMC_NUMBER_FB_ERRORS] OF SMC_FB_ERROR	0	轴相关错误表
1040	bVirtual	BOOL	FALSE	虚轴的时候此状态为 True 实轴为 False
1051	iRatioTechUnitsNum	DINT	1	轴设定转换齿轮比 (分母)
1052	dwRatioTechUnitsDenom	DWORD	1	轴设定转换齿轮比 (分子)
1060	iMovementType	INT	1	0 = 旋转轴 1 = 直线轴
1061	fPositionPeriod	LREAL	1000	旋转轴的最大移动距离
1062	eRampType	SMC_RAMPTYPE	Trapez	设定轴 Ramp 型式： <ul style="list-style-type: none"> ■ Trapezoid ■ sin^2 ■ Quadrtatic ■ Quadrtatic (smooth)
1100/1	fSetPosition	LREAL	0	命令位置 (单位：用户单位)
1101	fActPosition	LREAL	0	反馈位置 (用户单位)
1110,11	fSetVelocity	LREAL	0	命令速度 (用户单位/s)
1111,10	fActVelocity	LREAL	0	反馈速度 (用户单位/s)
1115	bConstantVelocity	BOOL	FALSE	当速度为等速时显示为 True
1120	fSetAcceleration	LREAL	0	加速度命令 (单位：用户单位/s^2)

编号	名称	数据类型	默认值	说明
1125	bAccelerating	BOOL	FALSE	轴在加速状态时为 True
1135	bDecelerating	BOOL	FALSE	轴在减速状态时为 True
1140	fSetJerk	LREAL	0	命令加速度的变化率 (跃度)
1160	fSetTorque	LREAL	0	命令扭矩 (Nm)
1161	fActTorque	LREAL	0	实际扭矩 (Nm)
1200,2	fSWLimitPositive	LREAL	0	设定软件正极限范围
120,3	fSWLimitNegative	LREAL	0	设定软件反极限范围
1204	bSWEndSwitchActive	BOOL	FALSE	当 SW 到达设定位置后此状态会为 True, 状态机变成 ErrorStop
1205	bSWLimitEnable	BOOL	FALSE	软件极限开关 : True 启动/False 关闭
-	strDriveInterfaceError	STRING	"	轴错误

● 定位轴 (Axis_REF_DML)

编号	名称	数据类型	默认值	说明
1000	nAxisState	SML_AXIS_STATE	SML_AS_PowerOff (0)	显示当前轴状态机
1012	bCommunication	BOOL	FALSE	当 EtherCAT 轴通讯正常时 True · 如果断线了为 False
1014	uiDriveInterface Error	UINT	0	如果 Driver Interface 检测到错误会显示其对应之错误码
1051	iRatioTechUnits Num	DINT	1	轴设定转换齿轮比 (分母)
1052	dwRatioTechUnits Denom	DWORD	1	轴设定转换齿轮比 (分子)
1060	iMovementType	SML_Movement Type	SML_MT_MODULO	轴类型 SML_MT_MODULO = 旋转轴 SML_MT_FINITE = 直线轴
1062	eRampType*1	SMC_RAMP TYPE	Trapez	设定轴 Ramp 型式 : ■ Trapezoid ■ sin^2
1101	fActPosition	LREAL	0	回授位置 (用户单位)
-	strDriveInterface Error	STRING	"	轴错误

注 1: 仅支持 Trapezoid、sin^2。

7.5.2 轴组变数

在软件 DIADesigner-AX 项目树中建立轴组后，系统会自动建立对应的轴变量。轴组变量如下表所示：

名称	数据类型	设定值 (默认值)	功能
GroupState	DMC_GROUP_STATE	GroupDisabled / GroupStandby / GroupMoving / GroupHoming / GroupStopping / GroupErrorstop (GroupDisabled)	轴组的状态命令
bError	BOOL	TRUE / FALSE (FALSE)	当轴组有错误时为 TRUE
dwErrorId	DMC_ERROR	DMC_ERROR (DMC_GM_NO_ERROR)	详细的错误描述
lrVelocity	LREAL	0 ~ 1.798E+308 (0)	轴组当前速度
lrAcceleration	LREAL	正数、负数或零 (0)	轴组当前加速度
lrJerk	LREAL	正数、负数或零 (0)	轴组当前加速度的变化率
bAccelerating	BOOL	TRUE / FALSE (FALSE)	当加速度中为 TRUE
bDecelerating	BOOL	TRUE / FALSE (FALSE)	当减速度中为 TRUE
bConstantVelocity	BOOL	TRUE / FALSE (FALSE)	当等速时为 TRUE (包含速度= 0)
bInPosition	BOOL	TRUE / FALSE (FALSE)	当定位完毕为 TRUE
bContinueDataWritten	BOOL	TRUE / FALSE (FALSE)	当轴组运动被暂停后相关数据可供 DMC_GroupContinue 使用时为 TRUE
ContinuePos	ARRAY [0..5] OF LREAL	[0,0,0,0,0,0]	当 DMC_GroupInterrupt 执行成功，记录轴组当前位置
AxisX_Name*	String		显示在当前轴组于 Axis_X 所使用的轴名称
AxisY_Name*	String		显示在当前轴组于 Axis_Y 所使用的轴名称
AxisZ_Name*	String		显示在当前轴组于 Axis_Z 所使用的轴名称
AxisA_Name*	String		显示在当前轴组于 Axis_A 所使用的轴名称
AxisB_Name*	String		显示在当前轴组于 Axis_B 所使用的轴名称
AxisC_Name*	String		显示在当前轴组于 Axis_C 所使用的轴名称

名称	数据类型	设定值 (默认值)	功能
RampType	DMC_GROUP_RAMP_TYPE	Trapezoid / S Curve (S Curve)	当前曲线类型
IrMaxVelocityLimit	LREAL	正数或零 (1000000)	最大轴组速度极限
IrMaxAccelerationLimit	LREAL	正数或零 (2000000)	最大轴组加速度极限
IrMaxDecelerationLimit	LREAL	正数或零 (2000000)	最大轴组减速度极限
IrMaxJerkLimit (Reserved)	LREAL	正数或零 (0)	最大轴组加速度的变化率 (跃度) (此功能保留)
bVelocityWarning	BOOL	TRUE / FALSE (FALSE)	当轴组速度超过比例时为 TRUE
bAccelerationWarning	BOOL	TRUE / FALSE (FALSE)	当轴组加速度超过比例时为 TRUE
bDecelerationWarning	BOOL	TRUE / FALSE (FALSE)	当轴组减速度超过比例时为 TRUE
bJerkWarning (Reserved)	BOOL	TRUE / FALSE (FALSE)	当轴组加加速超过比例时为 TRUE
StopMethod	Enum of BYTE	Immediate Stop / MaxGroupDecStop / MaxAxisDecStop (Immediate Stop)	设定轴组错误或运动中中断时停止方法
IrVelocityWarningPercentage	LREAL	0 ~ 1 (0)	设定轴组速度限制最大比例时，显示速度警告，设置 0 为禁用速度警告。
IrAccelerationWarningPercentage	LREAL	0 ~ 1 (0)	设定轴组加速度限制最大比例时，显示加速度警告，设置 0 为禁用加速度警告。
IrDecelerationWarningPercentage	LREAL	0 ~ 1 (0)	设定轴组减速度限制最大比例时，显示减速度警告，设置 0 为禁用减速度警告。
IrJerkWarningPercentage (Reserved)	LREAL	0 ~ 1 (0)	设定轴组加加速度限制最大比例时，显示加加速度警告，设置 0 为禁用加加速度警告。
Radius Correction	LREAL	0 ~ 100 (0.1)	设置使用中心模式的圆弧插补时，设置半径的允许误差%。误差% = ([中心]与[起始和终止位置的垂直二等分线之间的距离])半径。

注：当轴类别为旋转轴时，所设定的移动距离不能超过 modulo 的设定值，超过会报「Axis limit violated」错误。

7.6 运动控制程序编辑

7.6.1 运动控制程序

在 DIADesigner-AX 中编写运动控制程序前，请先参考以下编成架构说明。

7.6.1.1 DIADesigner-AX 编成架构与种类

在传统架构中，所有的程序，包括子程序都写在同一段程序代码 PLC 当中。当程序愈来愈庞大时，对于程序的维护与除错都会是相当大的负担。在 IEC 61131-3 架构下，程序依功能或特性，被分成若干个单元，使程序的开发与维护变得容易。因为程序组织单元 (POU) 的模块化，因此不同的 POU 可由不同的设计者来开发，能对项目的人力的分配与进行带来显著的效益。在 DIADesigner-AX 中 POU 可分为三种：程序 (PROG)、功能块 (FB) 及函数 (FC)。

1. 程序 (Program · PROG)

程序 POU 可依据其指定的工作 (TASK) 类型而决定其扮演的角色。若被指定至周期性的工作时，该程序 POU 扮演的便是主程序的角色；而若被指定至中断型的工作时，该程序 POU 便扮演中断子程序的角色。此外，程序 POU 亦可以对功能块 (FB) 进行调用。

2. 功能块 (Function Block · FB)

功能块 POU 内部可以宣告静态型 (有储存性的) 的变量符号。因此静态变量符号的值在运算之后可以被保持，由于执行时会将功能块内部储存的值与输入值一起运算，即使输入的数值相同，所造成的输出也可能不同。此外，功能块的内部可以再调用其它功能块。

3. 函数 (Function · FC)

Function 在 POU 中权限最小也受限制，与 FB 相比它只能有一个函数回传值，且没有使用内存空间的权限。Function 用于运算再将结果回传。在使用上 Function 只能调用 Function，而不能调用 FB，原因是 Function 并没有内存空间。

在建立完 POU 后，每个程序 POU 均须指定工作 (TASK)。工作 (TASK) 用来决定程序 POU 执行或启动的顺序。IEC61131-3 程序架构特色在于将程序切割成若干个独立开发的 POU。当 POU 编译时，会将所有 POU 重新排列组合为一个可逐步扫描的执行码。POU 重新排列组合顺序是依照 POU 被指派到的工作 (TASK) 而定。工作 (TASK) 可分为下列几种类型：

- Cyclic
- Event
- External
- Freewheeling
- Status

详细工作执行流程，可参考本手册第 4.4.1 节。

7.6.1.2 DIADesigner-AX 中的 POU

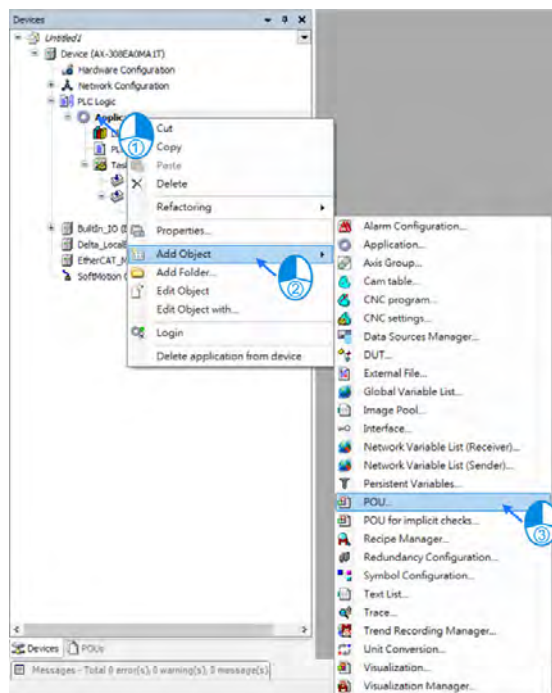
所有用户所建立的 POU 皆会在项目管理区中被列出。程序 POU 与功能块 POU 会被分开管理。此外，根据程序 POU 及功能块 POU 所编成的语言不同，所显示的 POU 图标也会有所不同，在每个 POU 名称的后方亦会带出 POU 相关的信息。在项目管理区的 POU 上双击鼠标左键即可开启该 POU 的编辑窗口。POU 编辑窗口由两个部份组成。窗口上半部为此 POU 的区域符号表，而窗口下半部则为程序的主体。此外，随着编程语言的不同，窗口下半部的编辑环境也会随之不同。



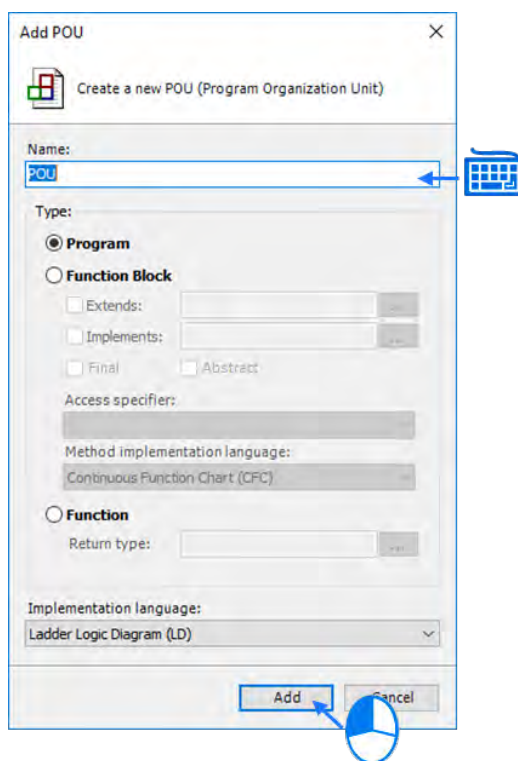
在区域符号表中用户能够自行新增所需使用到的变量。另外在系统中，当建立轴/轴组后会自动产生相对应的变量，而其变量内容可参考第 7.5 节。

7.6.1.3 DIADesigner-AX 新增 POU

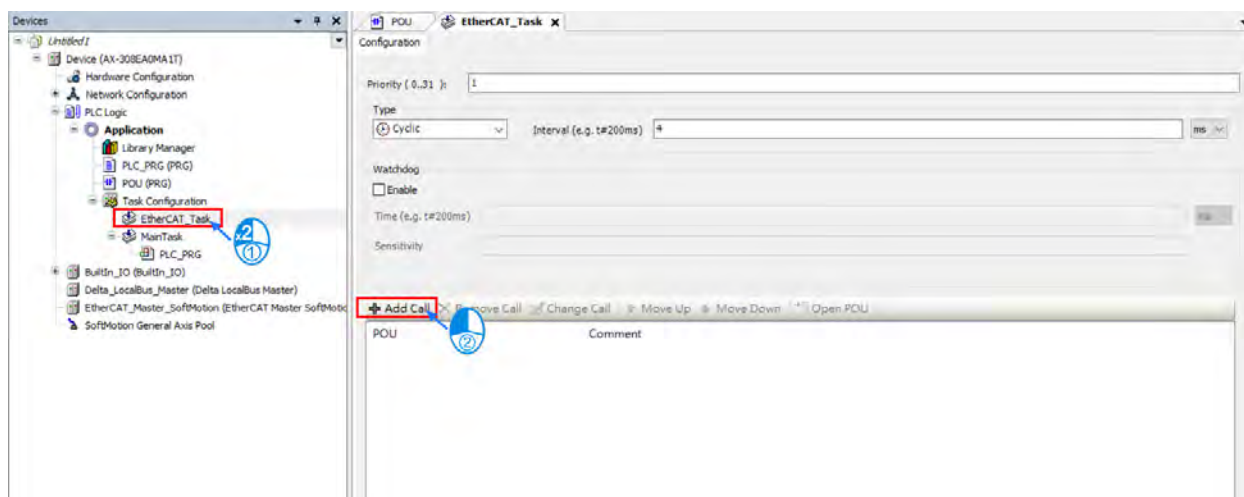
- 开启 DIADesigner-AX 已有的项目，并在程序中添加程序，在「Application」鼠标右击选取「Add Object」再选取「POU」。



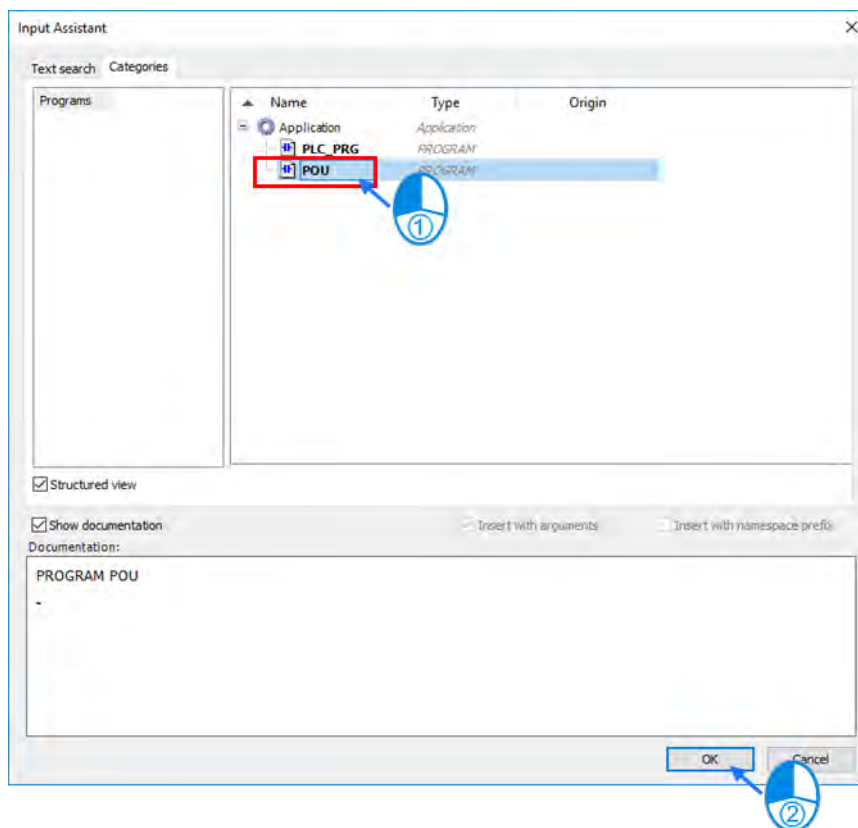
- 在「Name」栏中输入程序名称，并在「Implementation language」字段中选择程序语言，选择完成后，请按下「Add」。



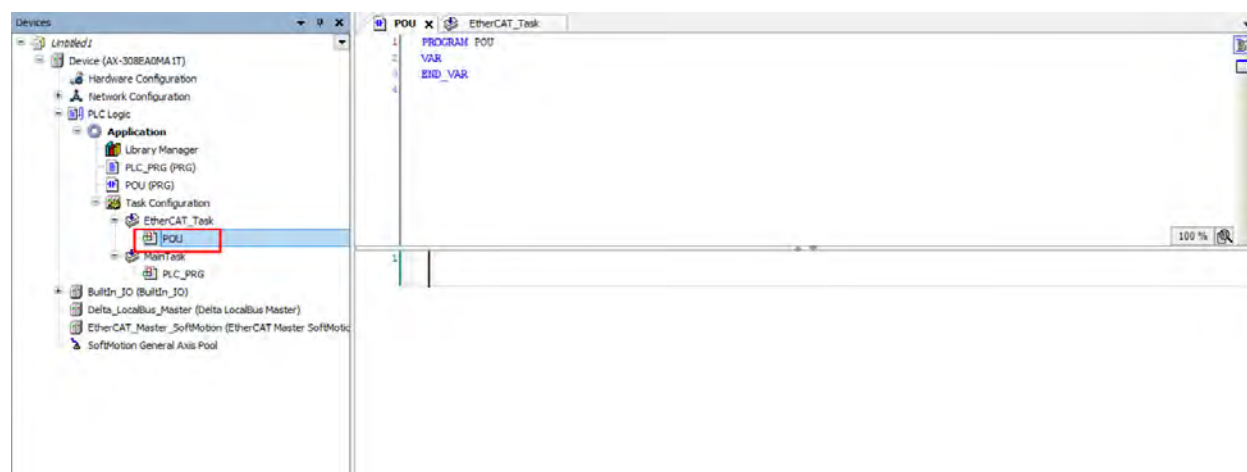
- 左方会出现 POU，并在「EtherCAT_Task」点取左键两下，选取「Add Call」。



- 选取所建立的 POU 并选择「OK」



- 选择「EtherCAT_Task」内的「POU」即可开始编写程序



7.6.1.4 PDO 映射 (Mapping)

在使用运动控制指令之前，必须完成 DIADesigner-AX 软件与 AX 运动控制 CPU 之间 PDO(Process Data Objects) 映射 (Mapping) 的通讯设置。

PDO 映设的设定值：

RxPDO (1600 hex)	Control Word (6040 hex) · TargetPosition (607A hex)
TxPDO (1A00 hex)	Status Word (6041 hex) · ActualPosition (6064 hex)

上表为 ASDA-A2-E 预先设定会使到用的 PDO 映射参数。

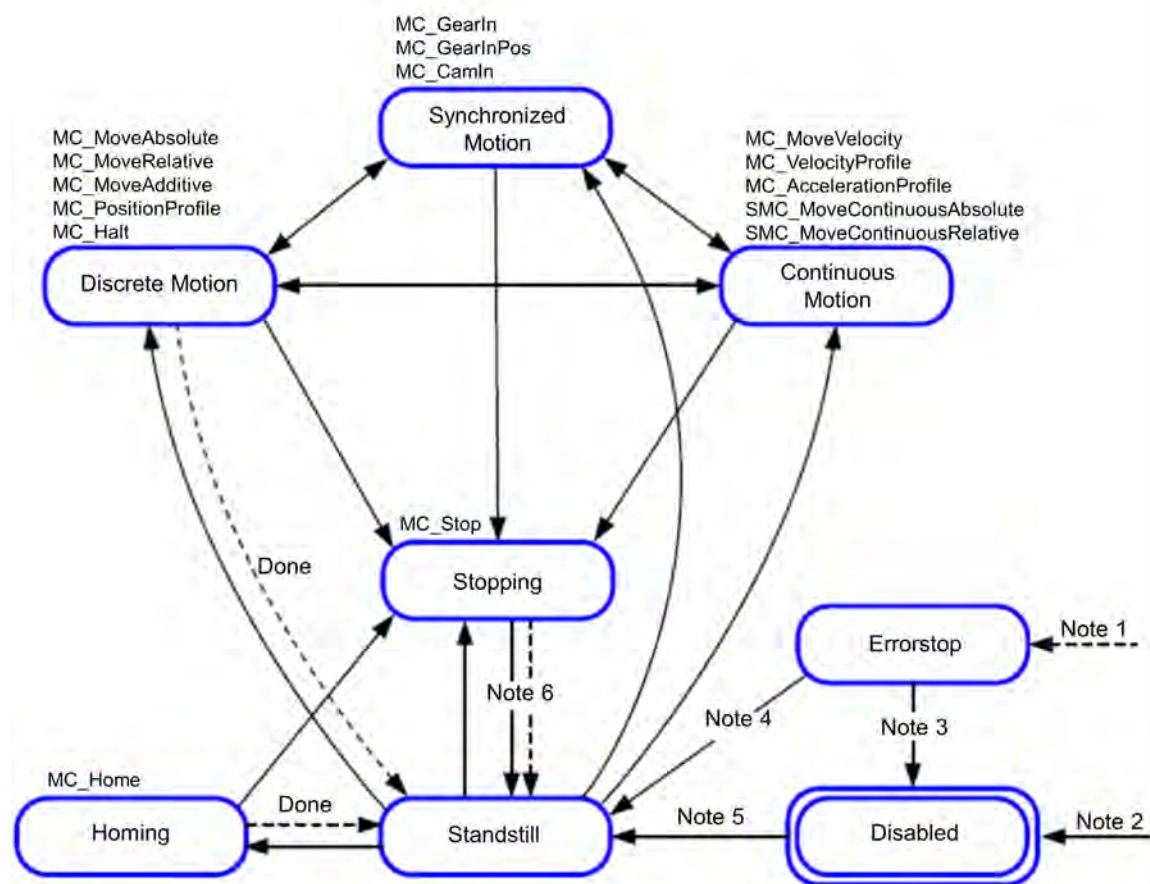
相对应运动功能块所需使用到的 PDO 参数，请参考 **AX 系列运动控制指令手册**。

7.6.2 轴状态转换

在说明如何使用运动功能块编写程序前，需先了解运动控制在各个运动功能块之间的状态机转换方式。此节介绍当使用到不同的运动功能块时，单轴状态机转换和轴组运动的多轴状态转换。转换规则符合 PLCopen 运动控制标准。

7.6.2.1 轴状态

- 同步轴



Note 1 : 轴只要发生错误就会进入“ErrorStop”

Note 2 : 当 MC_Power 的 Enable 变成 False 且没有发生轴错误时

Note 3 : 执行 MC_Reset 且 MC_Power 的 Status 为 False

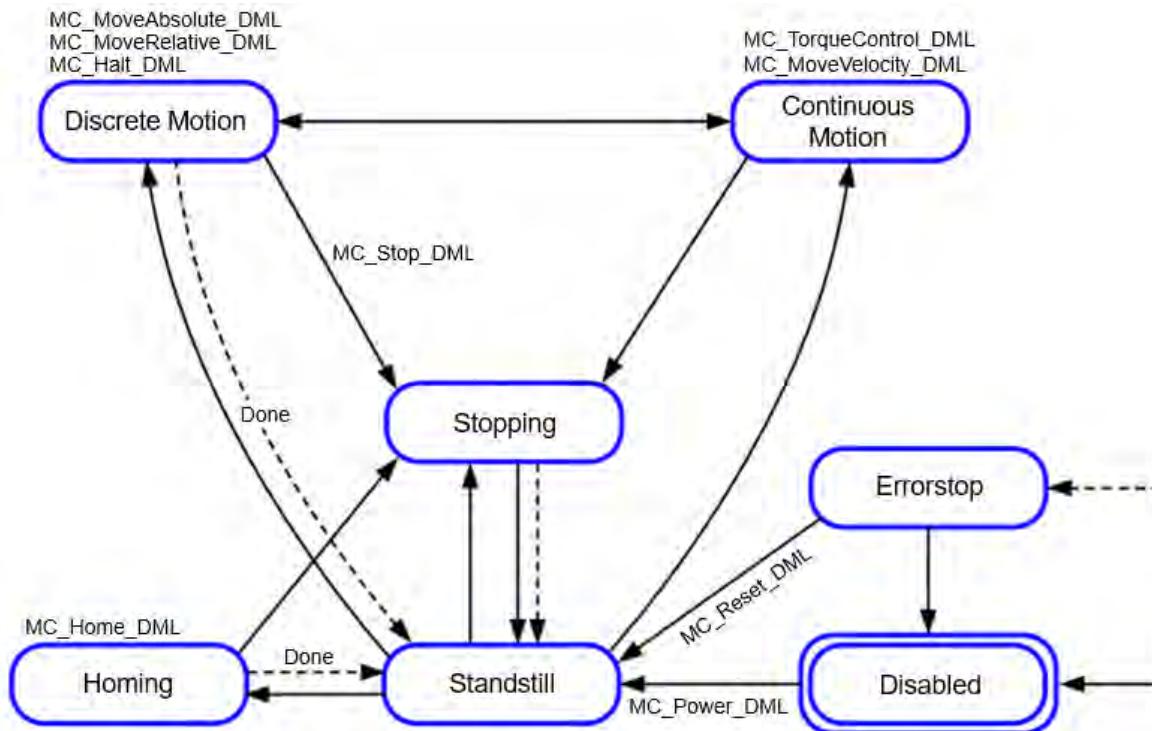
Note 4 : 执行 MC_Reset 且 MC_Power 的 Status 为 True 且 MC_Power 的 Enable 为 True

Note 5 : MC_Power 的 Enable 为 True · 且 MC_Power 的 Status 为 True

Note 6 : 当 MC_Stop 的 Done 变成 True · 且 MC_Stop 的 Execute 为 False 时

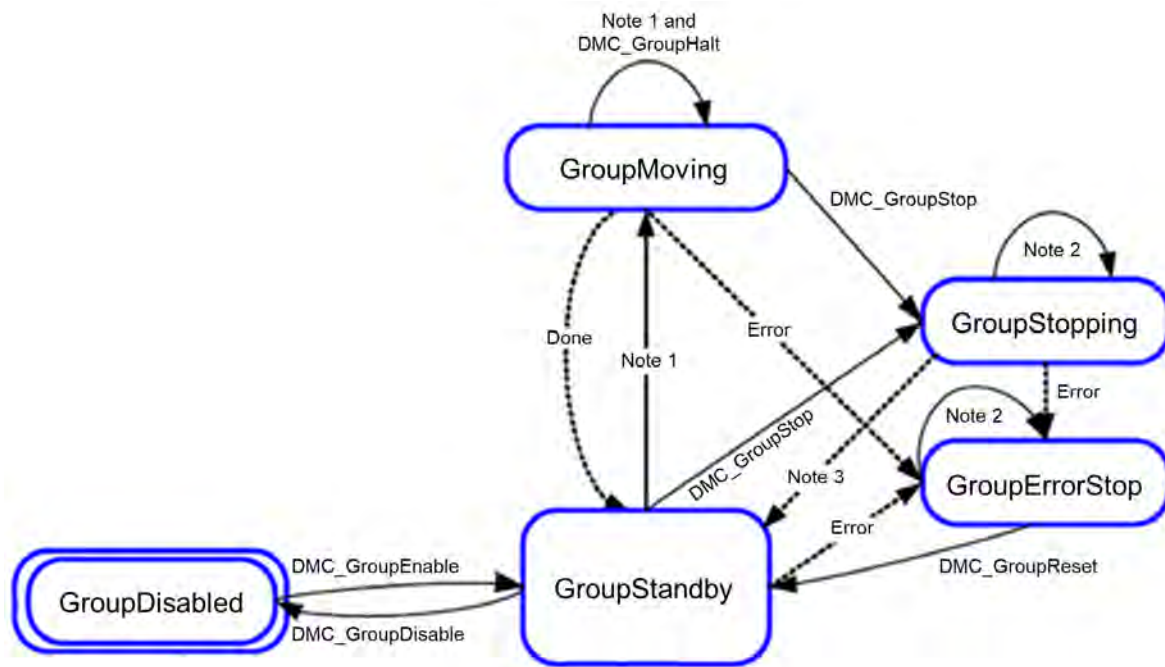
状态名称	定义
Disabled	轴处于伺服 Servo OFF 状态
Standstill	轴处于伺服 Servo On 的状态
Discrete Motion	执行单轴运动指令轴状态会为 Discrete Motion
Continuous Motion	执行单轴连续运动指令 · 轴状态为 Continuous Motion 。
Synchronized Motion	执行同步运动指令 · 轴状态为 Synchronized Motion 。
Stopping	执行 MC_Stop 指令停止后 · Active 为 TRUE 的状态 。
ErrorStop	当单轴只要错误 · 轴状态为 ErrorStop 。
Homing	执行单轴回原点指令 · 轴状态为 Homing 。

● 定位轴



状态名称	定义
Disabled	轴处于伺服 Servo OFF 状态
Standstill	轴处于伺服 Servo ON 的状态
Discrete Motion	执行单轴运动指令轴状态会为 Discrete Motion
Continuous Motion	执行单轴连续运动指令，轴状态为 Continuous Motion。
Stopping	执行 MC_Stop_DML 指令停止后，Active 为 TRUE 的状态。 在此状态下无法执行单轴任何运动指令。
ErrorStop	当单轴只要错误，轴状态为 ErrorStop。 在此状态下无法执行单轴任何运动指令。
Homing	执行单轴回原点指令，轴状态为 Homing。

7.6.2.2 轴组状态



Note 1：适用所有轴组移动功能块，非管理型。

Note 2：轴组状态为 GroupErrorStop 或 GroupStopping 时，可以执行所有轴组运动功能块，但是轴组不会有任何动作，除了 DMC_GroupResets 和 GroupErrorStop 以及发生任何错误，轴组状态会转为 GroupStandby 或 GroupErrorStop。

Note 3：当 DMC_GroupStop 为 Done 或 MC_GroupStop 非 Execute。

Note 4：GroupDisabled 只能在 GroupStandby 状态转换。如果轴组状态非 GroupStandby 下执行 DMC_GroupDisable 将会报错。

状态名称	定义
GroupDisabled	执行 MC_GroupDisable 指令，切换轴状态至 GroupDisabled。
GroupStandby	未执行任何轴轴组运动指令，轴组状态为 GroupStandby。

状态名称	定义
GroupMoving	正在执行轴组定位指令，轴组状态会在 GroupMoving。
GroupStopping	当 MC_GroupStop 的 Active 为 True 此时轴组状态会在 GroupStopping。 轴组在此状态，无法执行任何轴组运动指令。
GroupErrorStop	轴组只要发生错误时，就会进入 GroupErrorStop。

■ 单轴状态切换轴组状态

- 如果组中的一个轴在 ErrorStop 中，而组不在 GroupDisabled 中，则整个组在 GroupErrorStop 中。
- 当 GroupMoving/GroupStopping/GroupHoming 断开一个轴的电源时，Group 处于 GroupErrorStop 状态。
- 如果所有轴都处于 StandStill 状态，则组可以处于 GroupStandby、GroupDisabled 或 GroupErrorStop 状态。
- 如果单轴运动中中断了组运动，则该组中的其它单轴将立即停止并进入停止状态，组状态进入 GroupStandby。

■ 轴组状态切换单轴状态

- 如果组处于 GroupStandby 状态，则单轴状态不需全部处于 StandStill 状态。
- 组移动命令（包括 MC_GroupStop），则该组中的所有单轴均处于 SynchronizedMotion 状态。
- 当轴组运动期间发生错误时，轴组中所有轴将立即停止，直到组进入 GroupErrorStop。每个非错误单轴将进入 Standstill 状态。
- 轴组状态为 GroupErrorStop，则不影响单轴的状态。

7.6.3 运动控制指令的执行和状态

AX 系列控制器针对运动功能块分成以下以 2 大类别：

种类	描述
MC_	符合 PLCopen 标准的运动控制功能块指令
DMC_	台达自定义功能块指令*

而在每个运动控制功能块指令的一般引脚包括输入（Input）、输出（Output）及输入/输出（In-Out）。本节说明这些一般引脚的定义和一般行为

*注：台达自定义功能块指令（DMC）包括台达定义的运动控制型功能块和其它适用于 AX 运动控制 CPU 的管理/非管理型功能块。详细运动功能块内容，请参考 **AX 系列运动控制指令手册**。

7.6.3.1 执行指令的基本规则

● 输入和输出引脚的定义

运动控制功能块的一般输入和输出引脚如下所列。通常功能块包含至少一个或一部分下列的一般引脚。例如：

依照运动控制功能块的不同特性，一个功能块基本上会有 **Execute**（执行）或 **Enable**（使能）其中一个输入脚。

输入			
名称	功能	数据类型	设定值（默认值）
En	接收来自指令前端的逻辑状态	BOOL	True/False（False）
Enable	使能运动控制功能块	BOOL	True/False（False）
Execute	执行运动控制功能块	BOOL	True/False（False）

输出			
名称	功能	数据类型	设定值（默认值）
Eno	传送 En 输入的逻辑状态到下个串接指令	BOOL	True/False（False）
Done	表示功能块的执行已完成	BOOL	True/False（False）
Valid	表示输出引脚显示的值为有效值	BOOL	True/False（False）
Busy	表示运动功能块已排入执行序	BOOL	True/False（False）
Active	表示轴正被功能块所控制	BOOL	True/False（False）
CommandAborted	表示运动功能块的执行被中断	BOOL	True/False（False）
Error	表示功能块发生错误的状态	BOOL	True/False（False）

运动控制功能块通常会有 **Execute** 或 **Enable** 输入引脚。**Execute/Enable** 用于使能或启动运动控制功能块。运动控制功能块一般有 **Busy** 输出和 **Done** 输出。**Busy** 输出和 **Done** 输出表示运动控制功能块的状态。如果运动控制功能块的执行可被另一个运动控制功能块所中断，则此功能块输出脚会包含 **CommandAborted**。此外，当运动控制功能块执行时发生错误，输出脚 **Error** 将会转为 **True**。

运动控制功能块不仅具有 **Execute/Enable** 输入，也可能包含输入值/状态的输入引脚，其特性如下所述。

- 使用输入数值

- 具有 **Execute** 输入的功能块，各输入引脚的数值会在 **Execute** 输入信号从 **False** 到 **True** 的转态时被取用。但 **Execute** 在执行中重新被触发，并不会让输入数值被更新。
- 具有 **Enable** 输入的功能块，各输入引脚的数值会在 **Enable** 输入信号从 **False** 到 **True** 的转态时被取用。相较于 **Execute** 输入，**Enable** 输入引脚的功能块通常会有较多输入数值需要被持续更新（详见各功能块说明）

- 输入数值超出范围

- 运动控制功能块触发启动后，系统会限制用户输入超出允许范围的输入数值，或者在运动控制功能块执行时系统会显示功能块错误发生，进而让运动轴出现错误。用户应避免在应用程序中输入不正确的输入数值。

- 输出引脚的互斥特性

- 具有 **Execute** 输入的功能块，**Busy** 输出、**Done** 输出 **CommandAborted** 输出或 **Error** 输出，同时间只会有其中一个状态会被设置为 **True**。当 **Execute** 输入设置为 **True**、**Busy**、**Done**、**CommandAborted** 或 **Error** 输出其中一个必须设置为 **True**。
- 具有 **Enable** 输入的功能块，**Valid** 输出和 **Error** 输出是互斥的，意即只有其中一个可以设置为 **True**。

- 输出的数据/状态值有效时间点

- 具有 *Execute* 输入的功能块，当 *Execute* 输入信号由 True 到 False 转态，目前 True 的 *Done* 输出，*Error* 输出，*CommandAborted* 输出和输出脚位的数据会被复位或清零。但当功能块执行中 (*Busy*) 时，即使 *Execute* 输入信号由 True 到 False 转态，功能块的执行也不会停止。预计会出现的 *Done* 输出，*Error* 输出，*CommandAborted* 输出等输出状态仍然会产生 True 并保留一个周期。
- 具有 *Enable* 输入的功能块，当 *Enable* 输入信号由 True 向 False 转态，*Valid* 输出、*Busy* 输出和 *Error* 输出会复位。

- *Done* 输出特性

运动控制功能块成功执行完毕后，运动控制功能块的 *Done* 输出将被设置为 True。

- *Busy* 输出特性

- 具有 *Execute* 输入的功能块使用 *Busy* 输出以表示该功能块的执行尚未完成，且新输出状态 (值) 将在稍后产生。当 *Execute* 输入信号从 False 到 True 转态，*Busy* 输出设置为 True。当 *Done* 输出，*CommandAborted* 输出或 *Error* 输出被设置为 True，则 *Busy* 输出复位。
- 具有 *Enable* 输入的功能块使用 *Busy* 输出以显示该功能块的执行尚未完成，且新输出状态 (值) 将在稍后产生。当 *Enable* 输入信号从 False 到 True 转态，*Busy* 输出设置为 True，只要 *busy* 为 True 输入状态 (数值) 仍可预期变化。

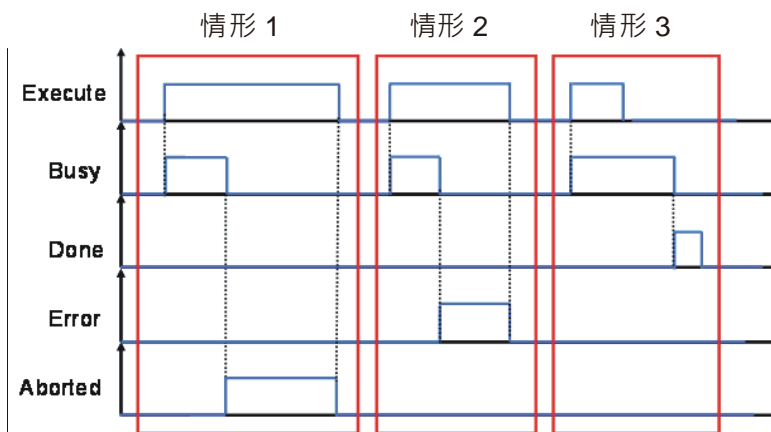
- *CommandAborted* 输出特性

当运动控制功能块的执行被中断时，*CommandAborted* 输出被设置为 True。

- *Enable* 输入与 *Valid* 输出的关系

具有 *Enable* 输入的功能块使用 *Valid* 输出表示目前输出数据/状态是否有效。只有当 *Enable* 输入设置为 True 且输出数据/状态是有效时，*Valid* 输出才会设置为 True，如果功能块出现错误，输出数据/状态将是无效的，且 *Valid* 输出将被设置为 False。运动控制功能块出现的错误被消除后，且输出数据/状态变为有效时，*Valid* 输出才会被设置为 True。

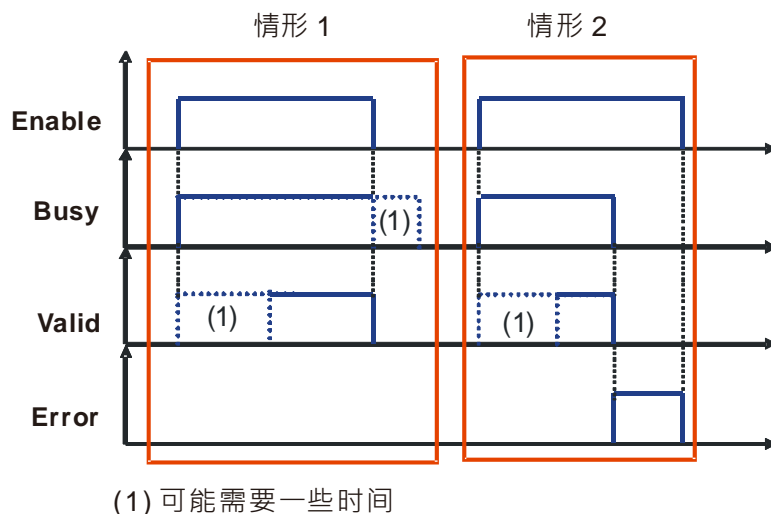
7.6.3.2 输入/输出引脚时序图



情形 1：运动控制功能块的执行中断。

情形 2：运动控制功能块发生错误。

情形 3：运动控制功能块正常执行完毕。



(1) 可能需要一些时间

情形 1：运动控制功能块正常执行。

情形 2：在运动控制功能块时发生错误。

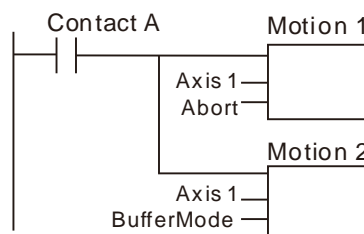
7.6.3.3 单一运动指令重复执行之行为

单轴运动功能块在功能块执行期间 (Busy 状态) 可修改输入引脚变量后再次对功能块执行引脚做上升沿触发。此时功能块输出引脚状态不变 (维持 Busy 状态)，但系统执行的动作等于以缓冲模式的 Aborting 打断前一次的上升沿触发命令。相关行为模式请参考本手册第 7.6.3.5 节说明。

7.6.3.4 多个运动指令同时启动

多个运动控制指令同时启动指的是同一个轴或轴组在同一程序扫描周期内有多个运动控制指令被触发启动。

- 下面程序中，当接点 A 为 ON 时，对应轴 1 的运动 1 和运动 2 会在同一个扫描周期启动。
- 在梯形图逻辑中，指令被执行是依据由上至下的顺序，所以运动 1 会先被启动而运动 2 会在运动 1 完成之后启动。
- 这个情况即被视为多个运动控制指令的同时启动。这种多重启动的模式是由输入变量 *BufferMode* 来定义运动间的结合方式，在此程序中会由运动 2 的 *BufferMode* 设定来决定运动 1 和运动 2 之间结合的模式。



7.6.3.5 运动指令同时执行之行为

单轴于 SoftMotion V4.10.0.0*(含)以上版本，且 SM3_Basic V4.10.0.0(含)以上版本支持 Buffered Mode 功能；DL_MotionControl V1.2.0.0 (含) 以上版本支持 SoftMotion V4.10.0.0 。

*注：SoftMotion V4.10.0.0 版本禁止修改 Axis_REF 内的参数，如修改会出现”SMC_MOVING_WITHOUT_ACTIVE_MOVEMENT”错误信息。

● **缓冲模式**

对于同一轴，当有运动指令控制轴在运动过程中，可以启动其它运动指令，前后两个运动指令进行交接时，有 6 种模式可供选择的交接模式，交接模式可以根据后一个运动指令的 BufferMode 脚位参数设定来选择。

BufferMode (交接模式) 相关术语的含义如下：

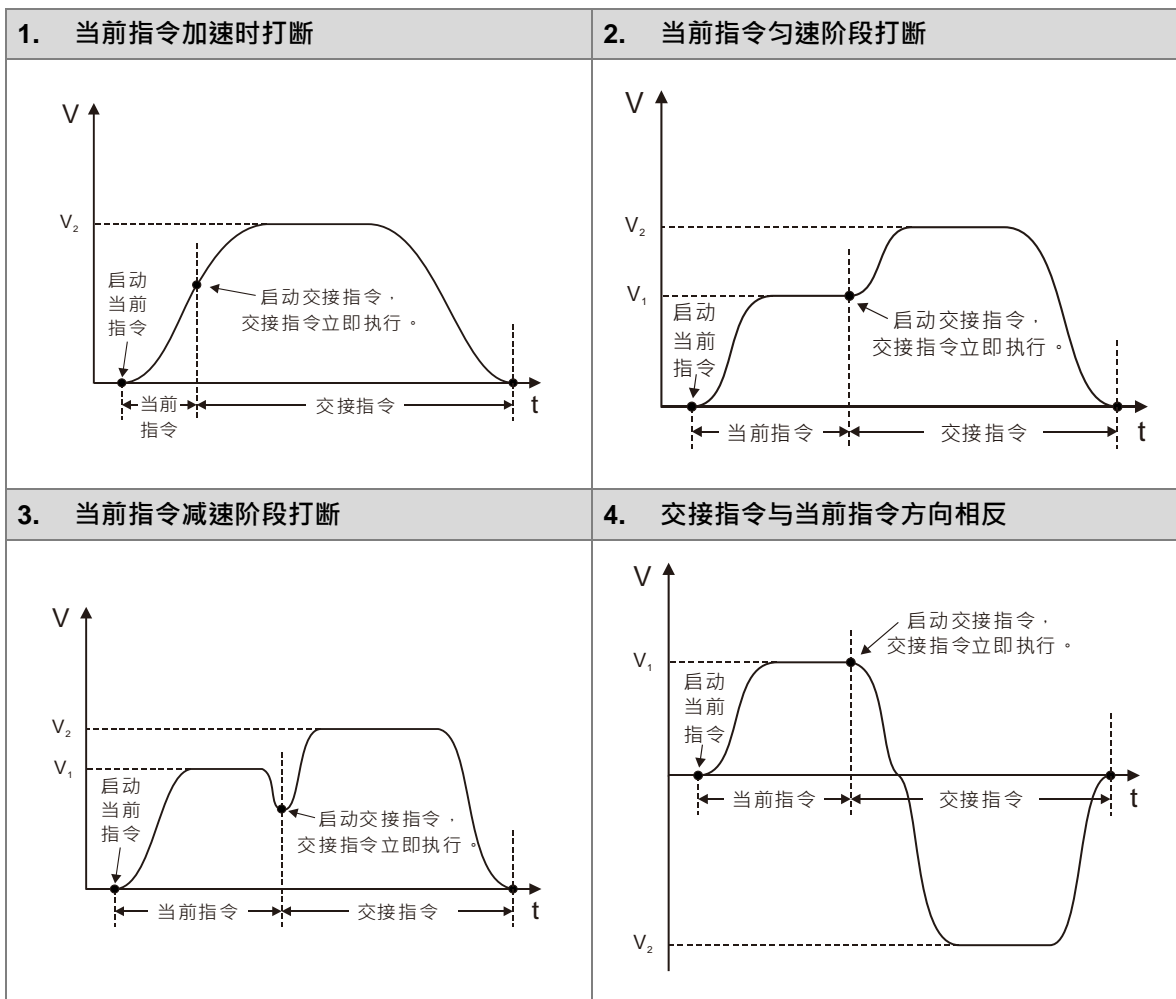
1. 当前指令：当前控制轴的运动指令
2. 交接指令：等待执行的指令
3. 交接速度：当前指令切换到交接指令时的速度
4. 目标速度：指令中的Velocity脚位参数
5. 目标位置：位移相关指令中Position或Distance脚位参数

交接模式	动作说明
0：mcAborting (打断)	立即打断当前指令的动作并执行交接指令的动作
1：mcBuffered (等待)	等待当前指令的动作正常执行结束后，并立即执行交接指令的动作
2：mcBlendingLow (以低速交接)	等待当前指令的目标位置到达后，并立即执行交接指令的动作；交接速度为当前指令与交接指令中较低的目标速度。
3：mcBlendingPrevious (以前一指令的速度交接)	等待当前指令的目标位置到达后，并立即执行交接指令的动作；交接速度为当前指令的目标速度。
4：mcBlendingNext (以后一指令的速度交接)	等待当前指令的目标位置到达后，并立即执行交接指令的动作；交接速度为交接指令的目标速度。
5：mcBlendingHigh (以高速交接)	等待当前指令的目标位置到达后，并立即执行交接指令的动作；交接速度为当前指令与交接指令中较高的目标速度。

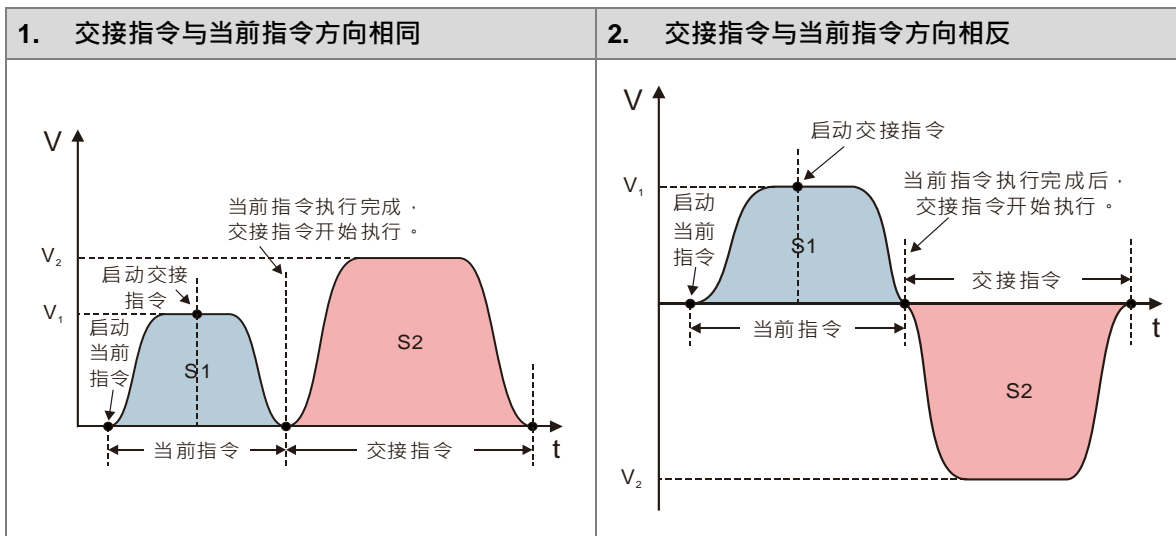
● **范例：用两个相对位移指令来简单说明**

第一个相对位移指令速度最大是 V_1 ，位移量为 S_1 ，第二个相对位移指令速度最大是 V_2 ，位移量是 S_2 。改变第二个位移指令的 BufferMode 使得这两个指令有不同的交接过程，如下说明：

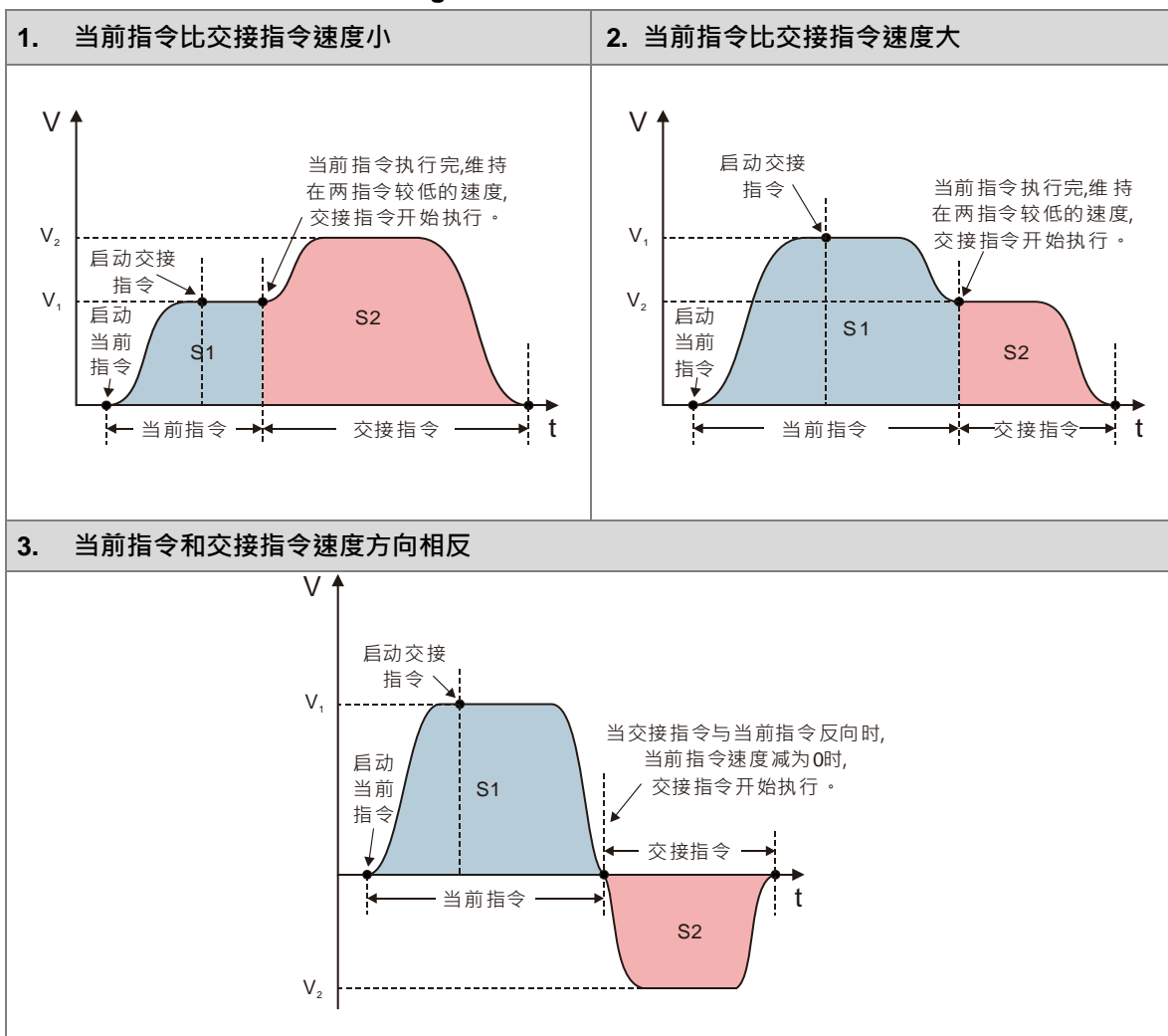
■ 打断 Buffermode=mcAborting，举例说明四种情况，分别如下：



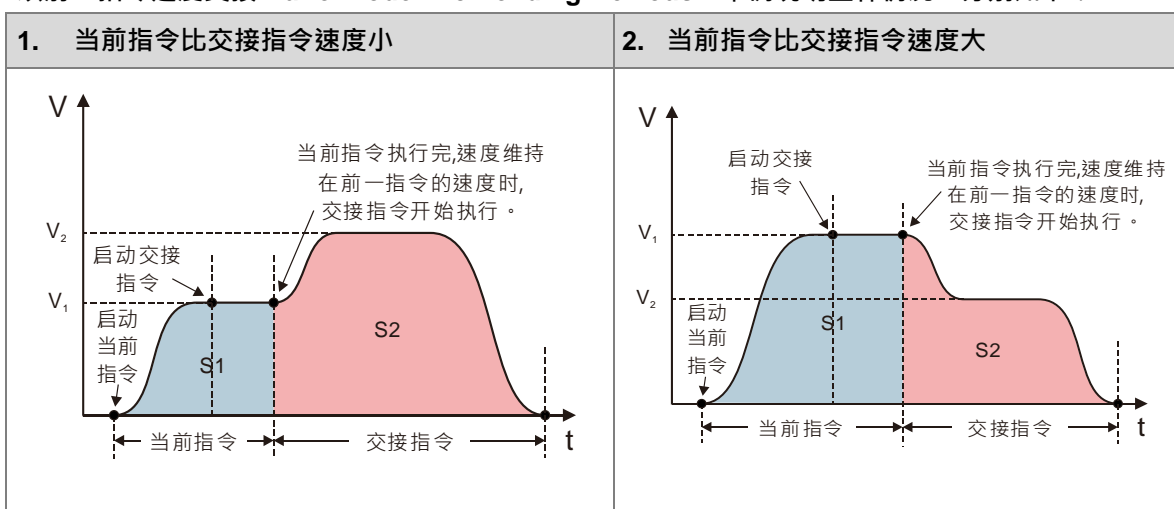
■ 等待 Buffermode=mcBuffered，举例说明二种情况，分别如下：

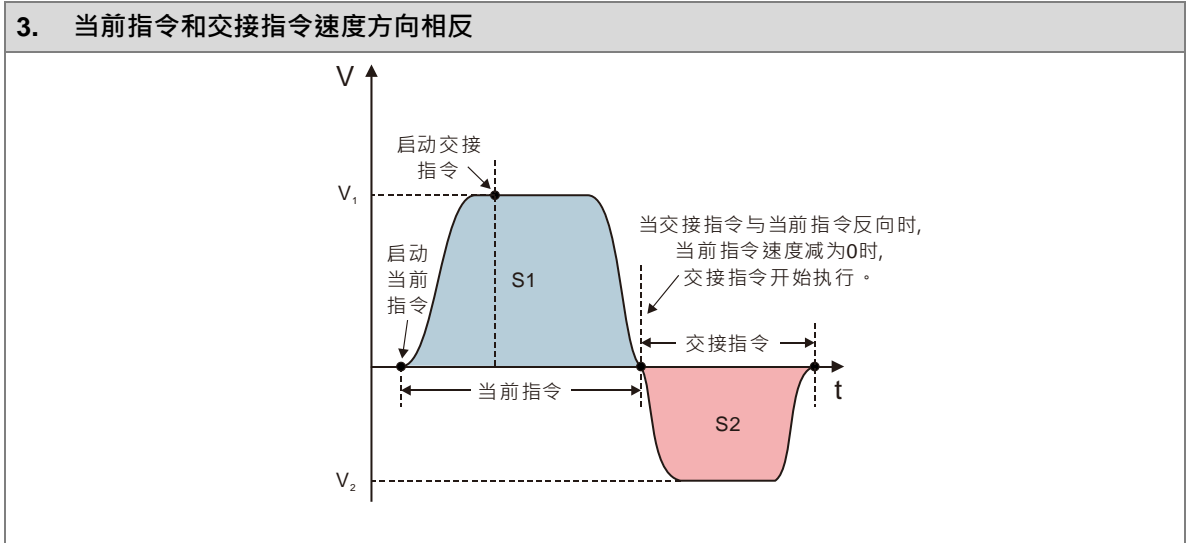


■ 低速交接 Buffermode=mcBlendingLow，举例说明三种情况，分别如下：

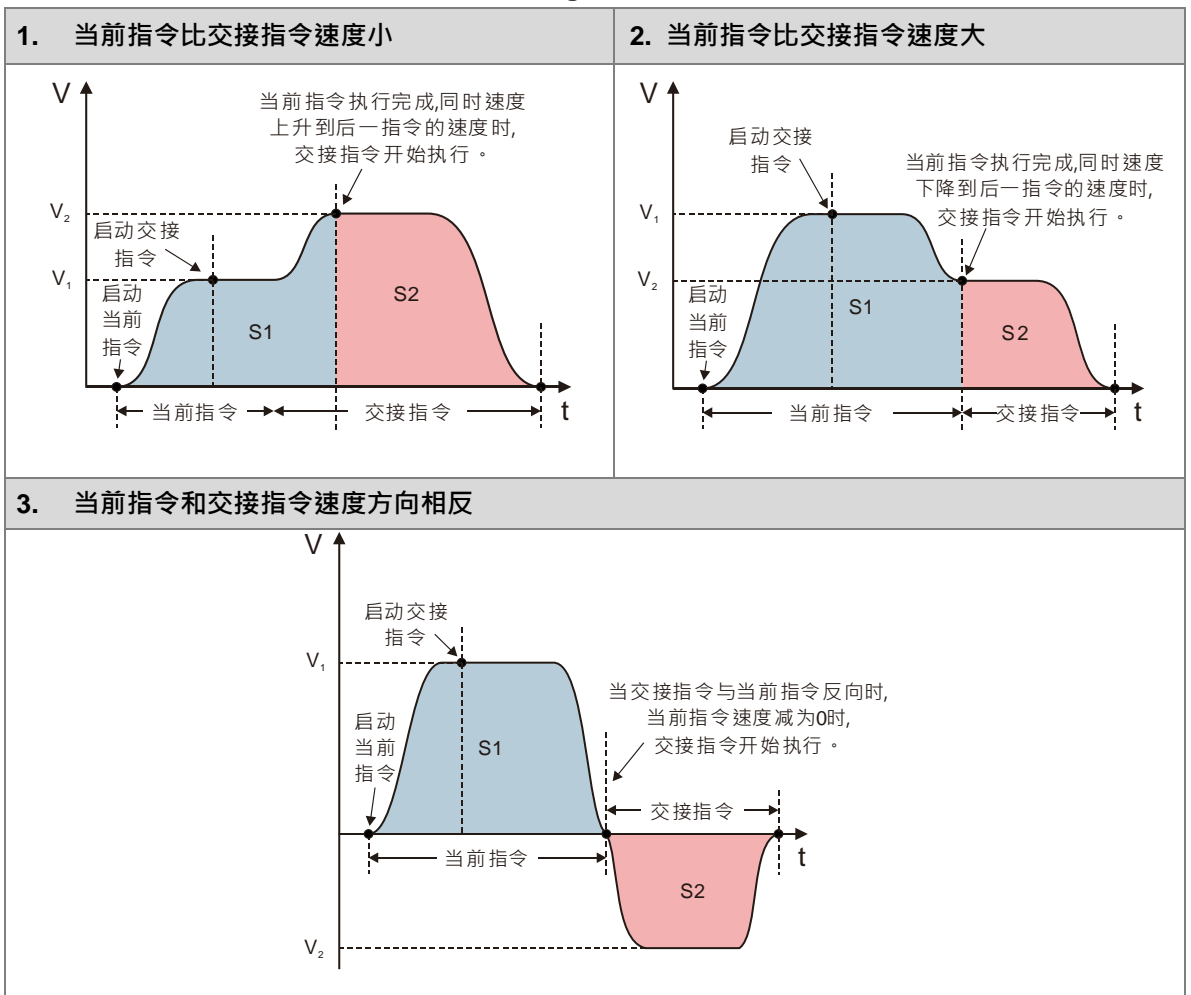


■ 以前一指令速度交接 Buffermode=mcBlendingPrevious，举例说明三种情况，分别如下：

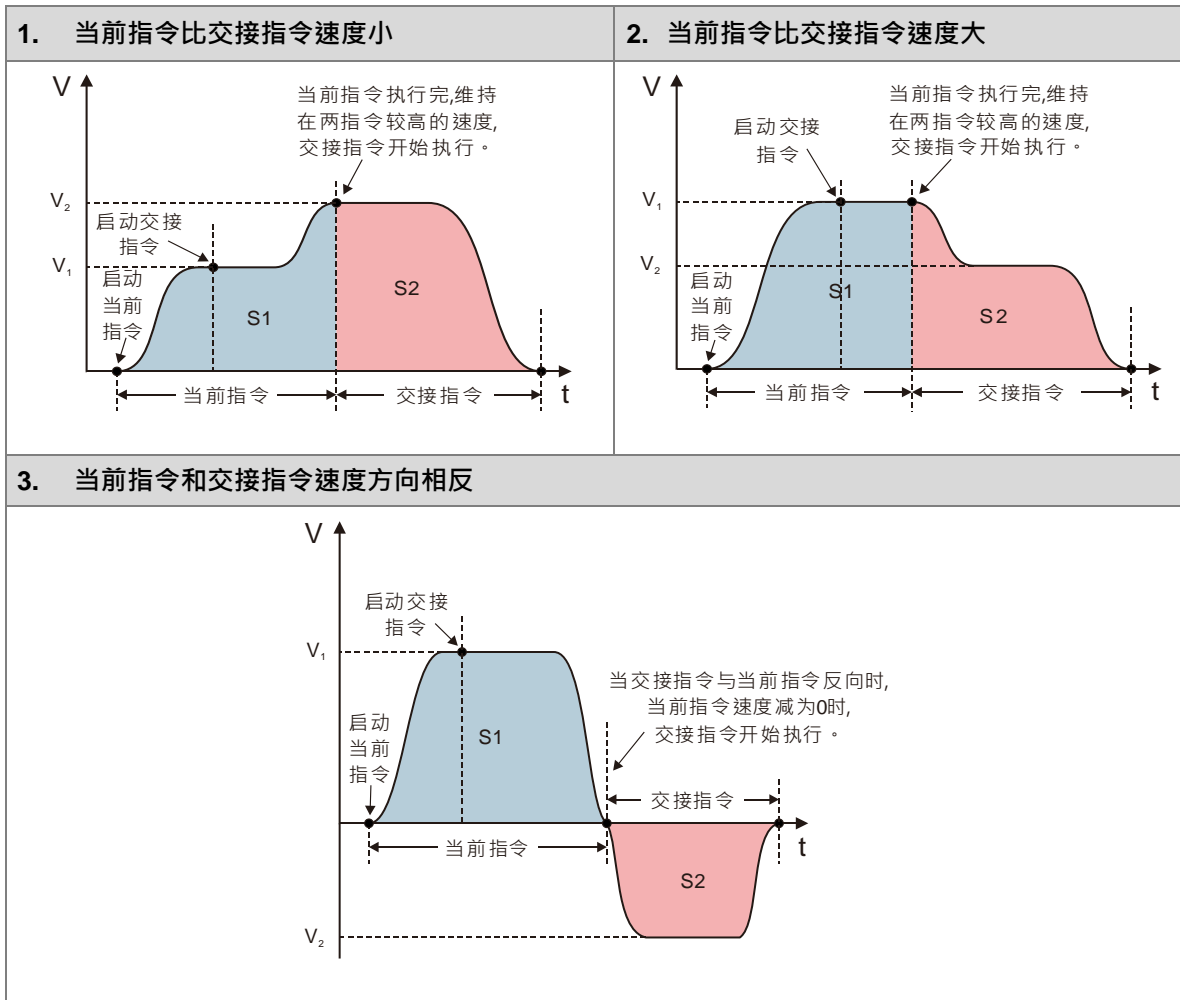




■ 以后一指令速度交接 $Buffermode=mcBlendingNext$ ，举例说明三种情况，分别如下：

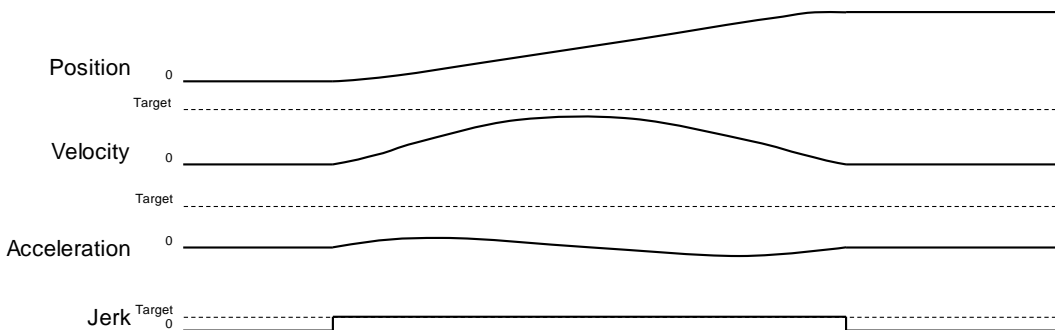


■ 以高速交接 Buffermode=mcBlendingHigh，举例说明三种情况，分别如下：



注：Buffered Mode 曲线规划主要以功能块输入参数 Jerk → Acceleration、Deceleration → Position → Velocity 依序满足条件后进行规划。

例：当 Position < Velocity 且 Velocity = Acceleration、Deceleration 而 Jerk < Acceleration、Deceleration 时曲线规划如下：



当 Jerk 达到 Target 后，由于路径不足，因此规划的 Acceleration、Deceleration、Velocity 不会达到 Target。

7.6.4 位置

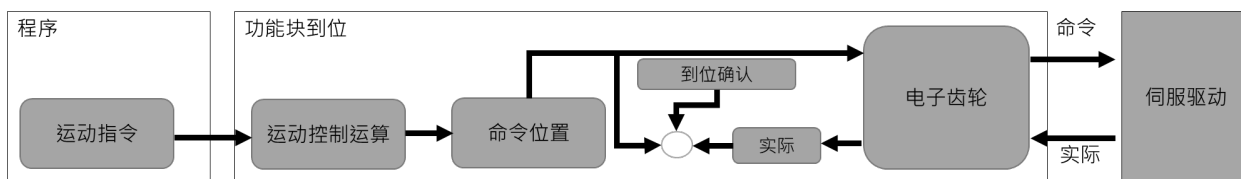
本节说明运动控制程序之位置定位流程。

7.6.4.1 位置的种类

MC 功能块由两种位置组成。

- 命令位置：运动功能块给定命令位置。
- 实际位置：从伺服驱动器实际的的实际位置*。

命令位置与实际位置的关系如下图所示：



指令位置与实际位置的以下项目相同。

位置种类	定义说明
命令位置	命令位置是运动控制器输出至驱动器的位置值
实际位置	反馈位置是为伺服驱动器或编码器实际的位置值

*注：虚轴时 命令位置等于实际位置

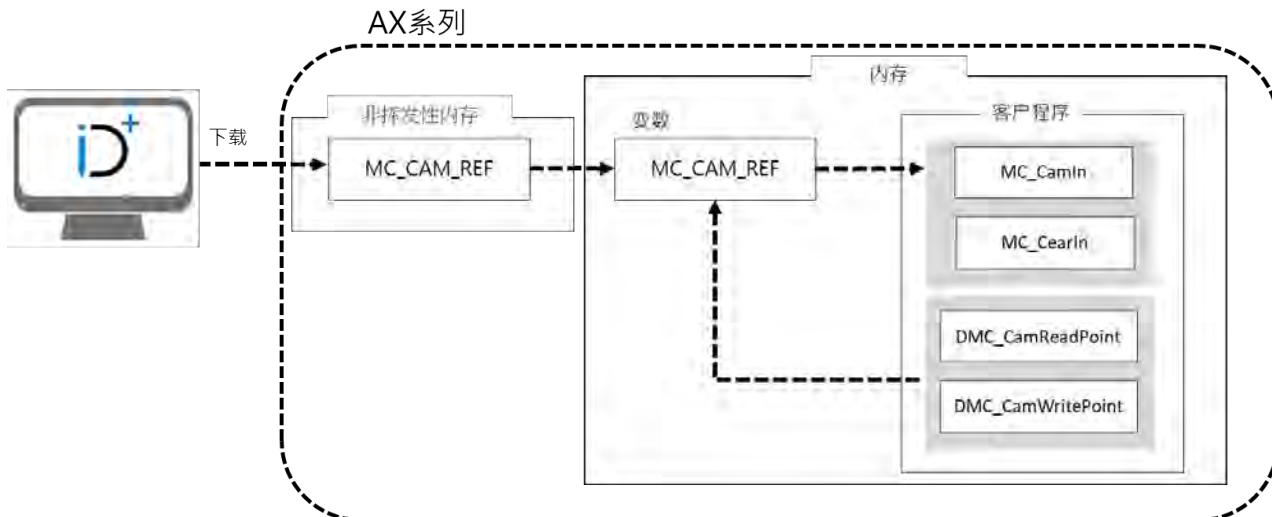
7.6.5 凸轮表和凸轮架构

此节介绍电子凸轮操作及使用 DIADesigner-AX 软件产生 CAM 表数据操作设定，以及电子凸轮相关功能块指令，详细指令说明请参考 AX 系列运动控制指令手册。

7.6.5.1 电子凸轮架构

使用 DIADesigner-AX 软件的凸轮编辑功能规划凸轮曲线。通过通讯下载至 PLC 并使用 MC 功能块控制凸轮。

- 凸轮表创建与架构



7.6.5.2 电子凸轮建立

定义主轴/从轴（凸轮轴）之间位置关系的数据，称之为电子凸轮数据。

若要使用 DIADesigner-AX 的 CAM 编辑器绘制，需要先知道主从轴位置的关系，主从轴位置的关系可通过以下两种方式进行建立：

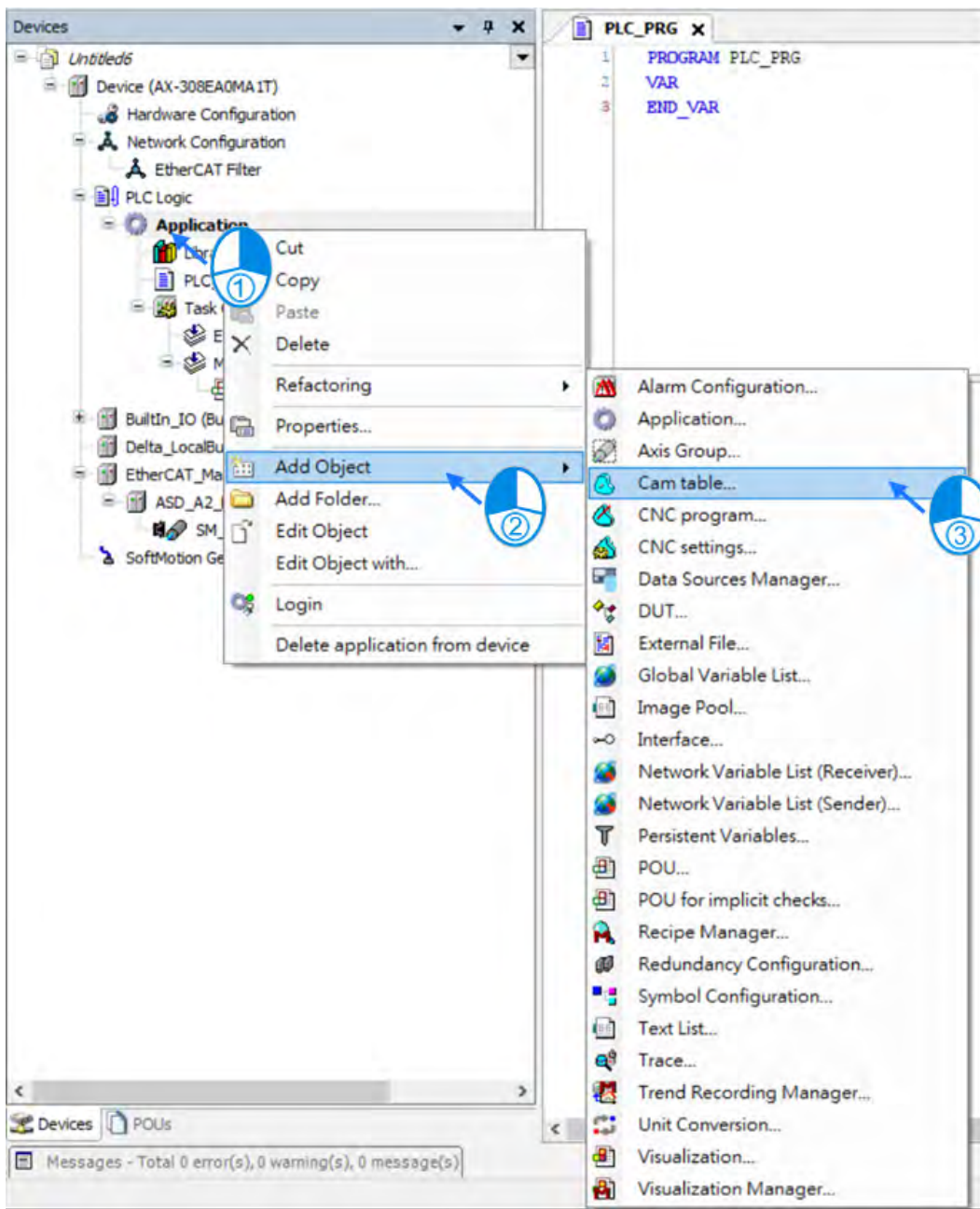
方法 1：根据设定的电子凸轮数据得到主从轴位置的关系。

方法 2：从实际工作中测量主从轴之间的对应关系。

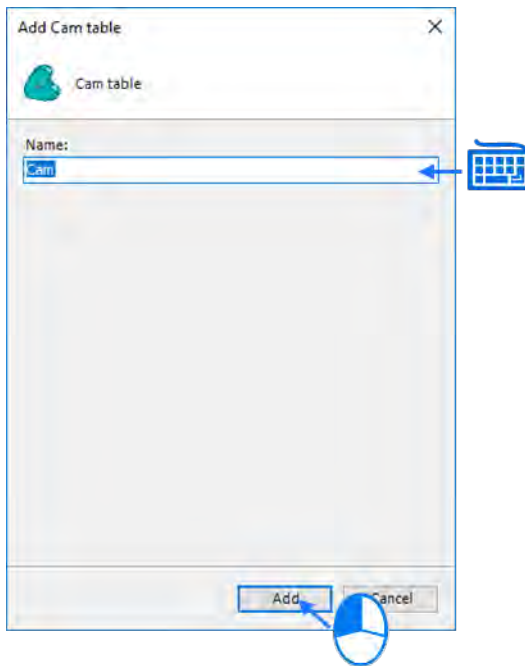
CAM 的主从轴关系确定后，根据主轴的位置就能得到从轴的位置。

- DIADesigner-AX CAM 表建立

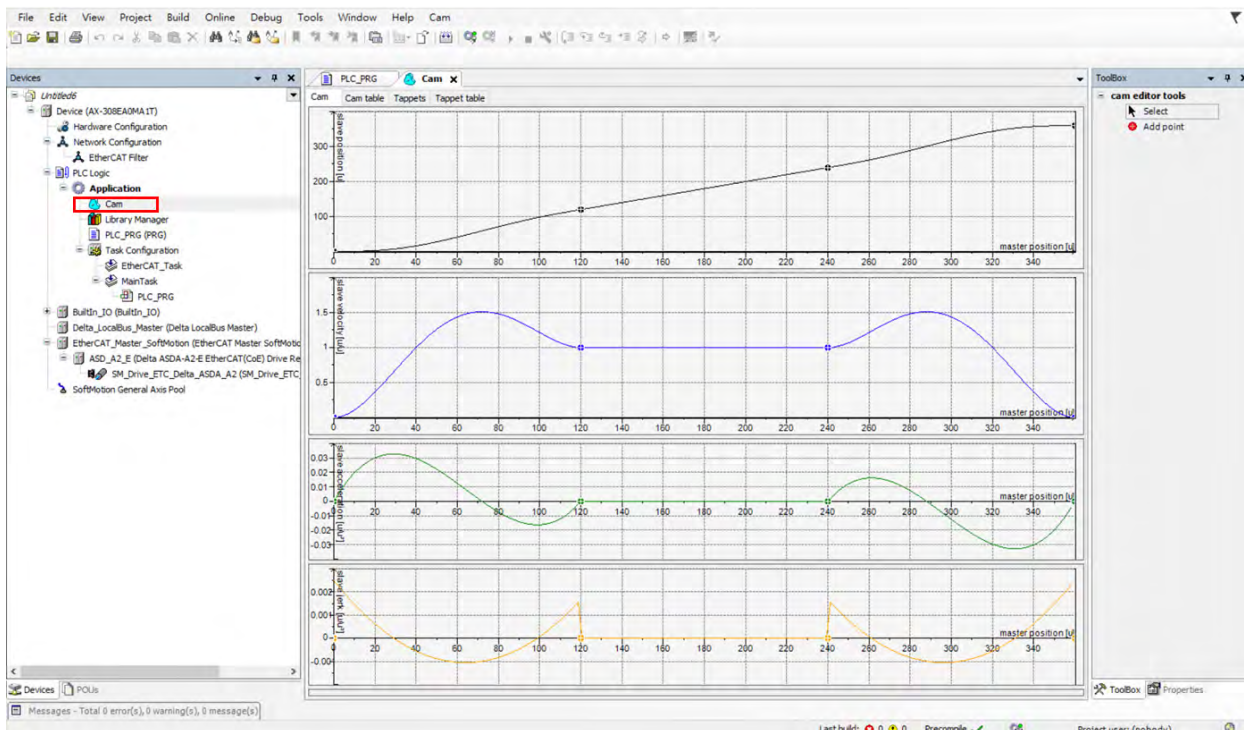
- 左方单击「Application」点右键，选择「Add Object」后，再选择「Cam Table」。



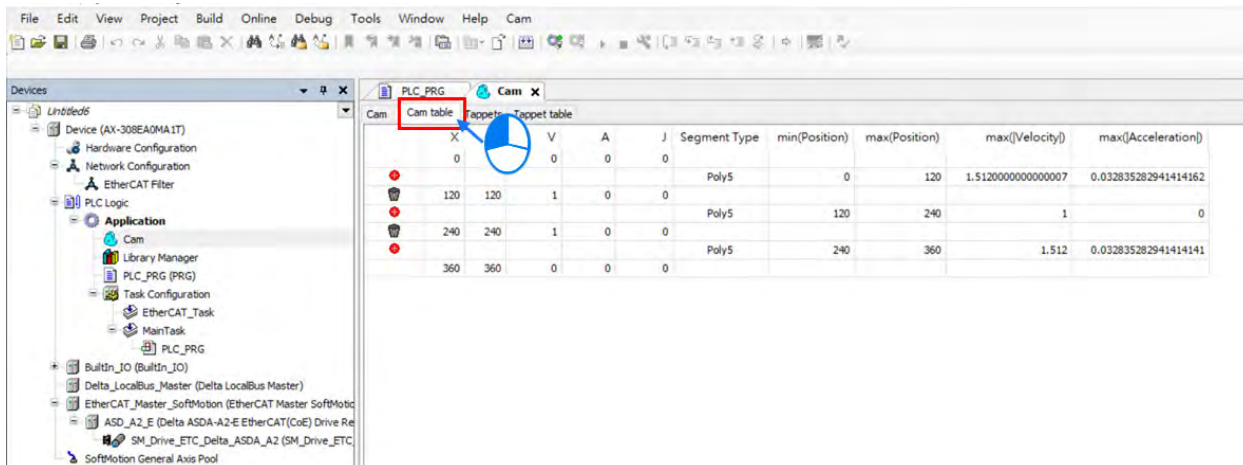
- 在「CAM Table」中「Name」为 CAM 表名称，可自行修改输入。



- 单击完「Add」即可在左方看到 Cam 建立

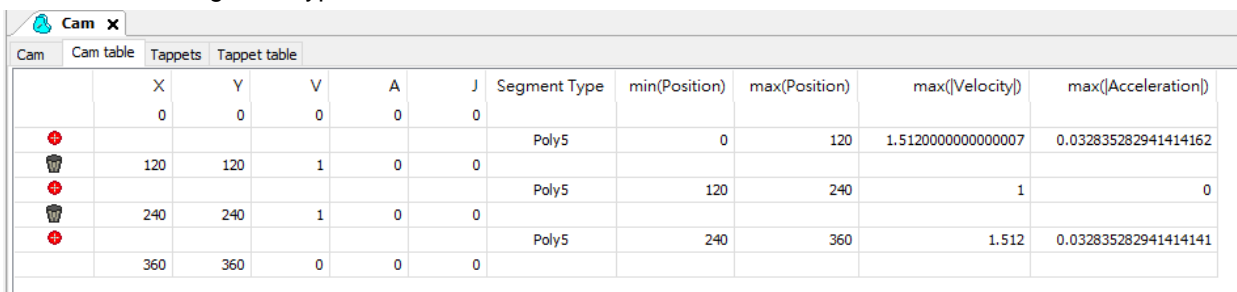


■ 在「CAM」的画面中单击「Cam Table」

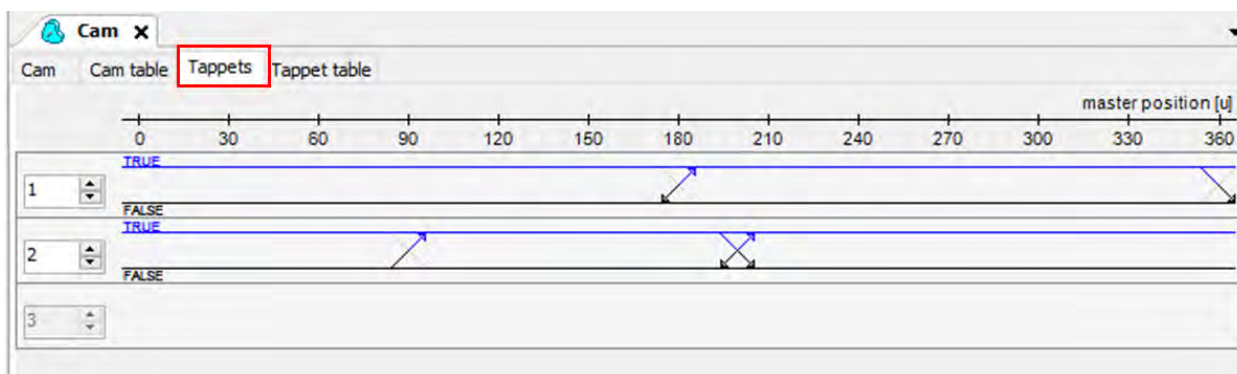




■ 「Cam Table」画面可修改 Cam 的新增或删除点位。

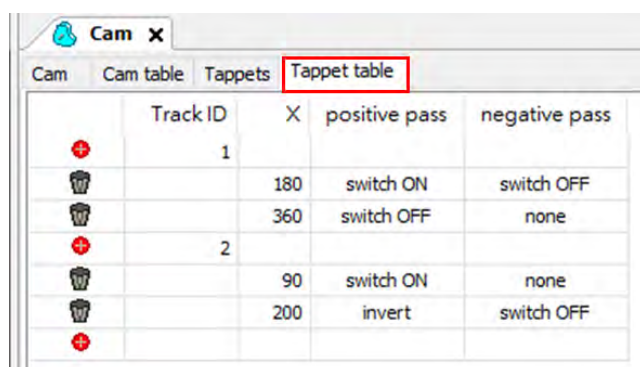
- 单击 + 可新增点位
- 单击 垃圾桶 可删除点位
- X：主轴点位
- Y：从轴点位
- V：从轴速度
- A：从轴加速度
- J：从轴加速度的变化率（跃度）
- Segment Type：曲线类型










■ 「Tappets」画面可设定多个挺杆，每个挺杆可设置数个挺杆点，设定完「Tappet table」可在 tappets 中观看挺杆点与主轴的关系图，拉动「Tappets」表上的挺杆点，可直接改动「Tappet table」上的设定参数。



- 「Tappet table」画面可设置挺杆与挺杆点，并搭配 SMC_GetTappetValue 来读取挺杆的状态。当凸轮主轴通过挺杆点时，判断主轴为正向或反向通过，并对应「Tappet table」中的设定来改变挺杆的状态。
 - 单击  可新增 Track ID
 - 单击  可删除 Track ID
 - Track ID：挺杆点 ID
 - X：挺杆点位置
 - Positive pass：正转遇到挺杆点动作，动作设定如下：
 - ◆ None：无动作
 - ◆ Switch to ON：TRUE
 - ◆ Switch to OFF：FALSE
 - ◆ Invert：反向
 - Negative pass：反转遇到挺杆点动作，动作设定如下：
 - ◆ None：无动作
 - ◆ Switch to ON：TRUE
 - ◆ Switch to OFF：FALSE
 - ◆ Invert：反向



	Track ID	X	positive pass	negative pass
	1			
		180	switch ON	switch OFF
		360	switch OFF	none
	2			
		90	switch ON	none
		200	invert	switch OFF
				

7.7 运动控制功能

7.7.1 系统架构

运动功能块的单轴运动指令，可在速度控制、位置控制、扭矩控制等三种控制模式之下，依用户设定参数，规划轴的运动轨迹。

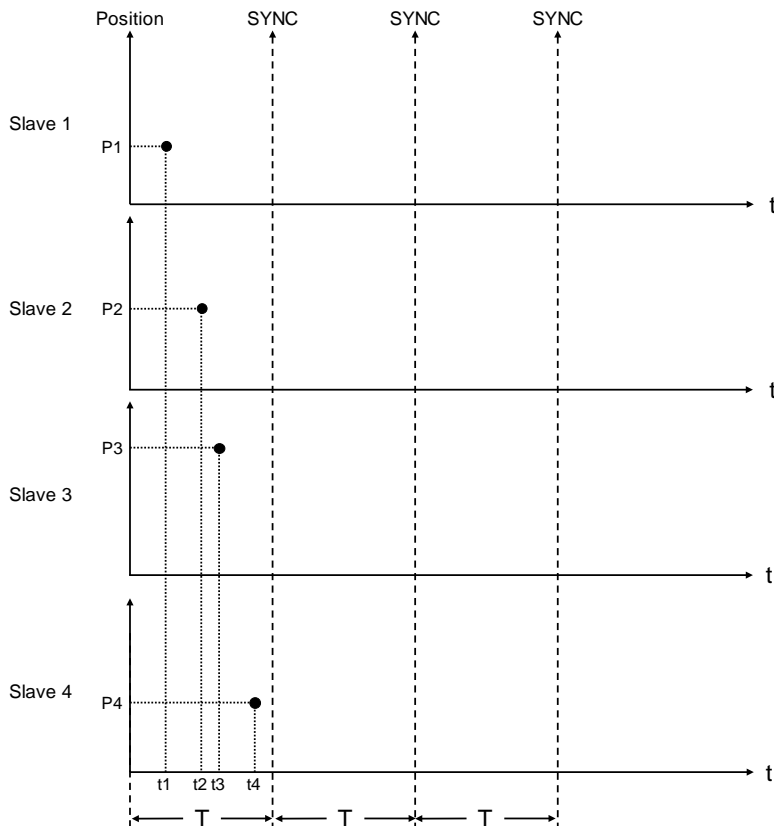
而在 AX 系列控制器其支持 CoE (CANopen over EtherCAT) 应用协议，依照 CiA402 中规范所支持的运动模式可分为：周期同步位置 (Cyclic Synchronous Position Mode)、周期同步速度模式 (Cyclic Synchronous Velocity Mode)、周期同步扭矩模式 (Cyclic Synchronous Torque Mode) 种类的模式，会在下列说明。

7.7.2 单轴控制

下节将说明在单轴控制中整体系统架构。

7.7.2.1 周期同步位置模式 (Cyclic Synchronous Position Mode)

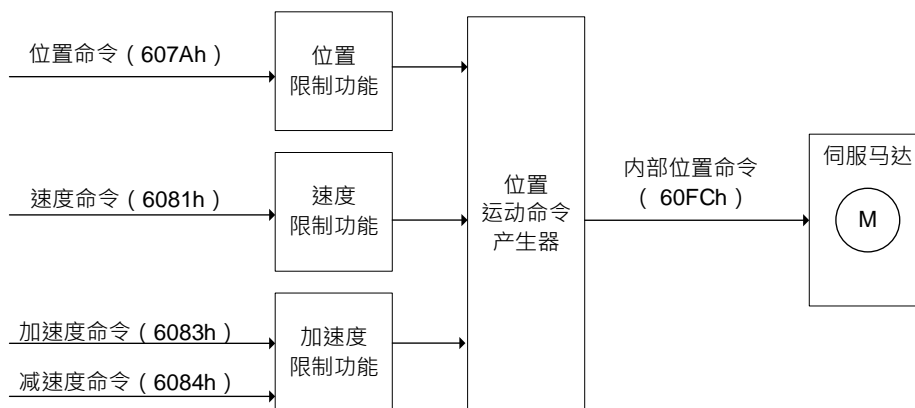
AX 系列控制器与多台伺服驱动器连接时，其数据传递是通过同步信号实现同步，同步信号由控制器以广播的方式发出。伺服驱动器接收到控制器发出的控制数据，这些控制数据并不会马上生效，需等待各驱动器内部计数时间达到后 (Distributed Clock · DC) *，以此实现多台伺服的同步。如下图所示接有四台伺服驱动器，T 为同步周期。在同步周期内，四台伺服驱动器分别在不同的时刻 (t1、t2、t3、t4) 接受到控制数据，但是控制数据并没有生效。当所有伺服驱动器计数时间到达同步信号 (SYNC) 后，控制数据同时生效。



注：周期同步位置模式使用于同步轴。

7.7.2.2 位置规划模式 (Profile Position Mode)

伺服驱动器接收到由上位控制器 (以下简称「上位机」) 传送的位置命令后，驱动器控制伺服电机到达目标位置。在位置控制模式下，上位机仅在一开始时告知驱动器目标位置、速度命令与加减速等相关设定。从命令触发到到达目标位置这中间的运动规划，都是由驱动器内部的运动命令产生器去执行。

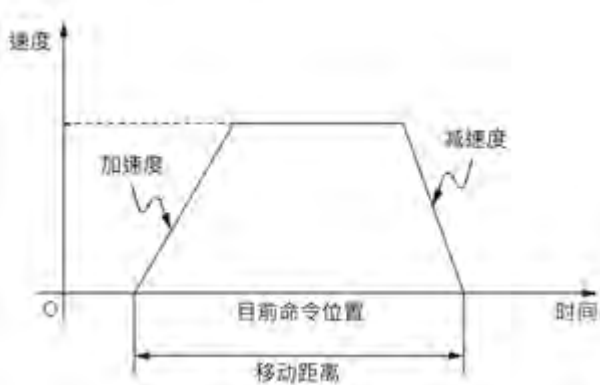


注：位置控制模式使用于定位轴。

7.7.2.3 定位

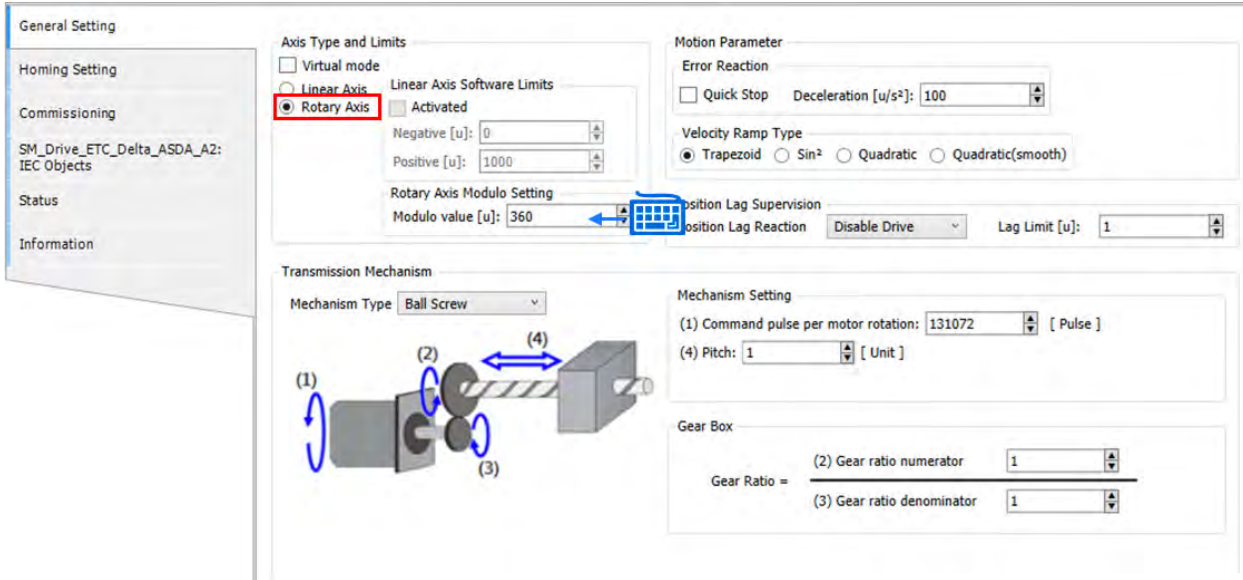
● 绝对寻址

功能块会依据用户输入设定值与当前位置规划轴移动曲线，使轴移动至以原点为基准点的目标绝对位置。此外旋转轴的绝对位置范围，会受限于旋转轴的旋转周期范围。详情可参照 MC_MoveAbsolute 功能块。绝对寻址的运动轨迹如下图所示。



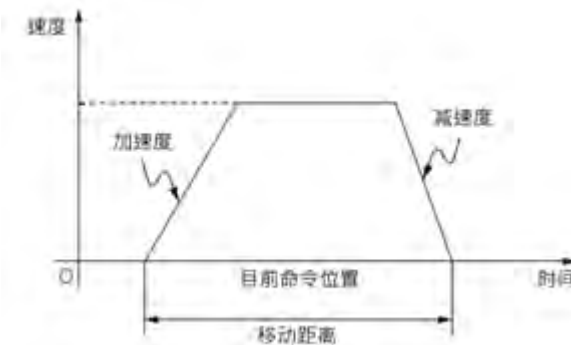
■ 旋转轴设定

- 轴设定画面选择 Rotary Axis 后，在 Modulo value 设定旋转轴范围。



● 相对定位

功能块会依据用户输入设定值与当前位置规划轴移动曲线，使轴移动至以当前位置为基准点的目标相对位置。详情可参照 MC_MoveRelative 功能块。相对定位的运动轨迹如下图所示。



7

7.7.2.4 停止方式

轴停止情况包含用户通过运动指令使轴停止，或是碰极限，轴发生异常错误导致停止。轴发生异常错误导致停止的错误清除。

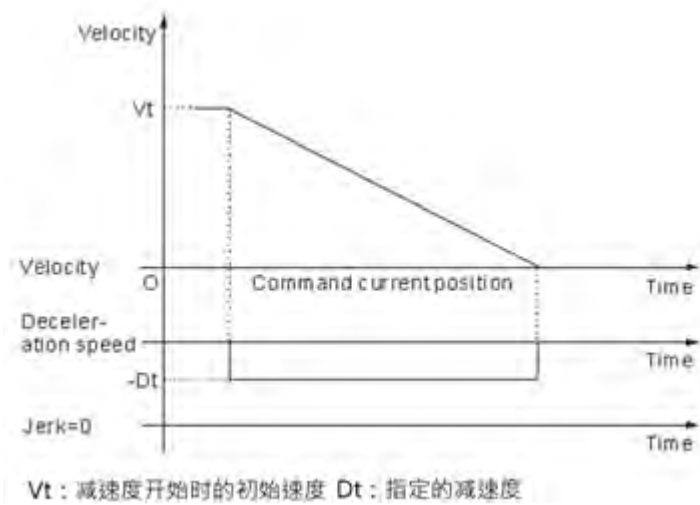
- 通过运动控制指令停止：需停止单轴动作时，应使用MC_Stop指令或MC_Halt指令。

MC_Stop

- MC_Stop依指定的方式停止正在运动中的轴并将轴状态转为“Stopping”。
- 此指令会中断任何正在执行中的指令。当此轴的状态在“Stopping”时，任何指令都不能执行。
- 状态“Stopping”会持续到达速度0或者是Execute变成False。当达到速度0，Done马上变为True。
- 当Done变为True且Execute变为False，轴会进入“Standstill”状态。

MC_Stop运动轨迹如下图所示

速度依据指定的减速度Dt而决定



MC_Halt

- MC_Halt暂时停止正在运动中的轴并且将轴状态转换至“DiscreteMotion”一直到轴速度为0。当轴停止时，停止动作完成后轴状态会进入“Standstill”。
- 在轴减速的期间可以执行其它运动指令来立即中断MC_Halt的运作

- 碰极限后停止

软件极限：

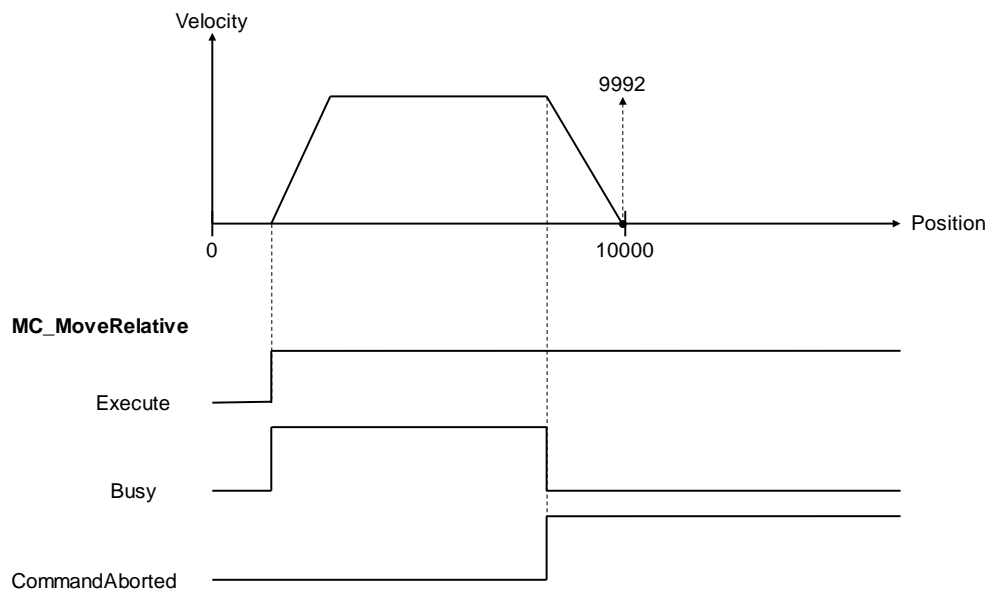
软件极限可在轴参数设定页面进行开启/关闭，并设定其范围。

当轴在运动中接近软件极限时，会以轴参数的减速度规划减速停止，并停止在软件极限内。

软件极限范例如下：

- 软件极限设置正极限为 10,000 反极限为 0 并开启软件极限 (Activated)，并设置减速度为 1,000。

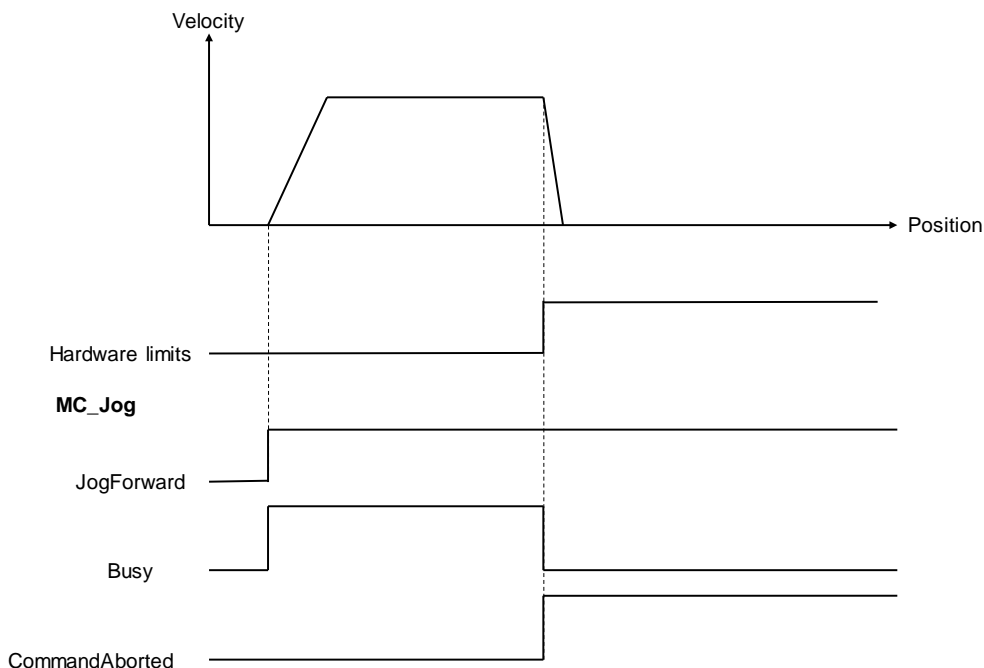
- 使用 MC_MoveRelative 功能块，并定位至 11,000 后启动该功能块，当轴移动至 8,000 左右后，功能块的 Busy 会由 TRUE 转为 FALSE 且 CommandAborted 由 FALSE 转为 TRUE，轴开始减速至软件极限内的位置。



硬件极限：

硬件信号会直接配线在 EtherCAT 伺服轴上，因此各家针对硬件极限停止方式略有不同。以下以台达 ASDA-A2-E 伺服为例：

- 使用 MC_Jog 执行轴正转，但轴运转中遇到极限开关，ASDA-A2-E 会立即停止运转，并通过通讯回报错误。



当碰触软件/硬件极限后，可使用 MC_Reset 清除错误后，系统会自动将命令位置将依据实际位置值进行同步，并执行脱离极限方向即可再正常运行。

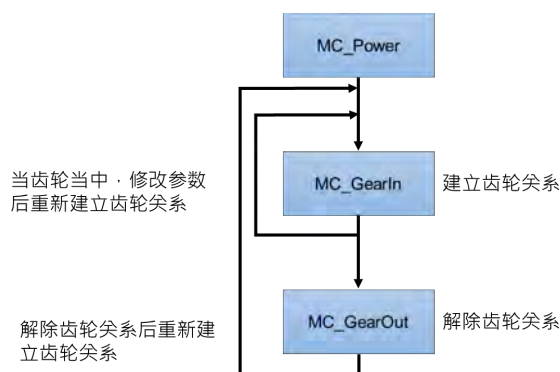
7.7.2.5 电子齿轮 (MC_GearIn)

当有主从轴需依据比例进行同步运动需求，可使用 MC_GearIn 指令执行齿轮动作。而 MC_GearOut 指令执行后，从轴会以执行当下从轴命令脱离同步运行并等速运行。详细功能块使用请参考 **AX 系列运动控制指令手册**。



MC_GearIn 中指定主轴与从轴、齿轮比分子、齿轮比分母、加速度、减速度、加速度的变化率（跃度）。

- 电子齿轮相关的指令执行顺序如下：



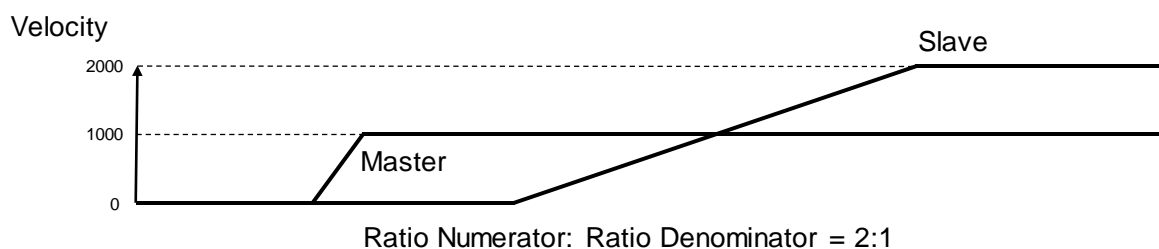
当执行 MC_GearIn 后从轴状态机进入 Synchronized motion，在执行 MC_GearOut 使从轴脱离同步状态后，从轴会维持脱离瞬间的速度继续运动，从轴状态机进入 Continuous Motion。

在同步运动时，对从轴执行 MC_Stop，此时 MC_GearIn 会被打断，主轴状态维持 Continuous Motion 继续运转，从轴则进入 Stopping，直到 MC_Stop 的 Done 拉下状态才会回到 standstill。

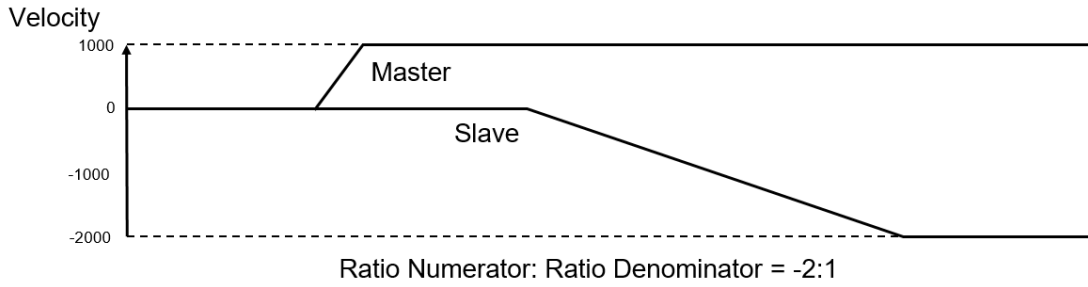
从轴状态机在 Synchronized motion 时，从轴的速度会跟随主轴的速度并经由齿轮比来调整。

当主轴从轴进入同步状态后，使用 MC_SetPosition 指令以免造成电机急速运转产生事故。

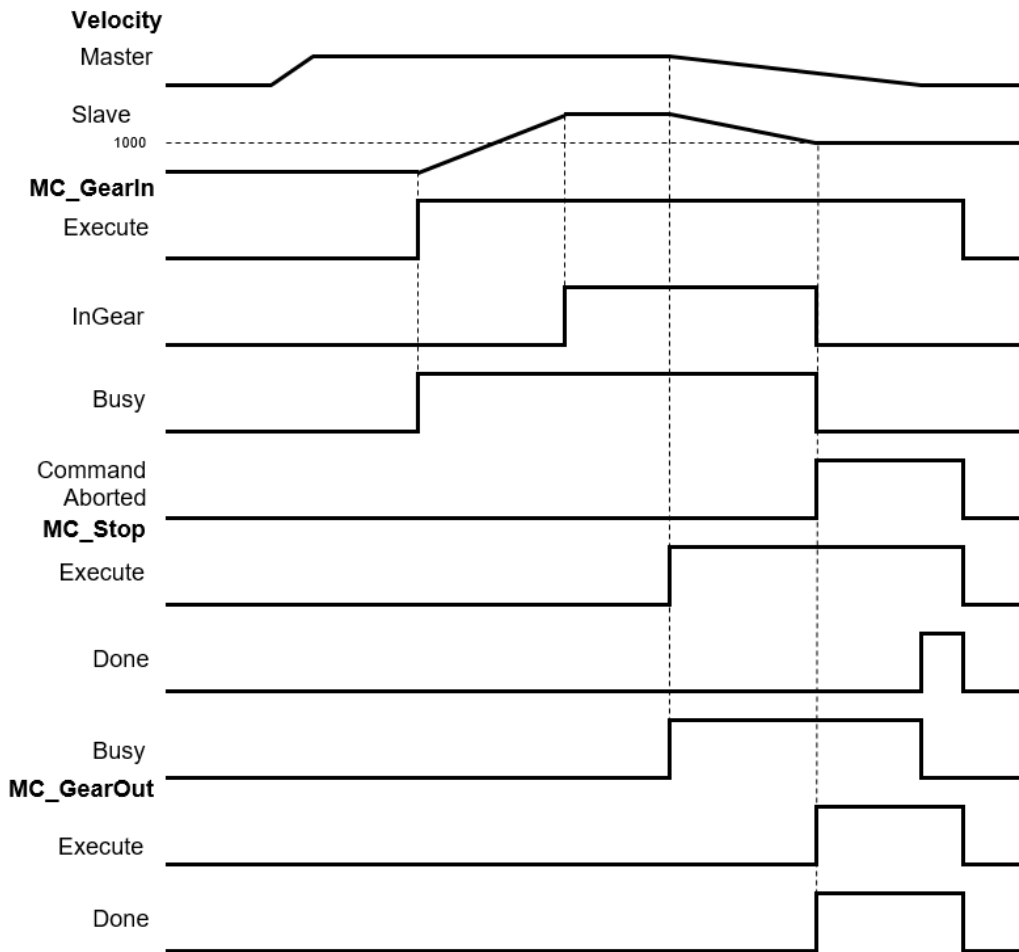
- 通过 MC_GearIn 的 RatioNumerator、RatioDenominator 来设定主轴、从轴间的齿轮比。
当齿轮比为正值时，主轴和从轴以相同的方向运动。



当齿轮比为负值时，主轴和从轴以相反的方向运动。



- 当从轴速度达到指令设定的速度时，主轴与从轴完成同步。



当 MC_GearIn 启动时，从轴开始啮合主轴。从轴速度到达 2 倍的主轴速度 (RatioNumerator : RatioDenominator = 2 : 1)。

InGear = True，主从轴完成同步，从轴轴状态进入 Synchronized Motion。

MC_Stop 启动后，主轴开始减速，从轴也依齿轮比同步减速。

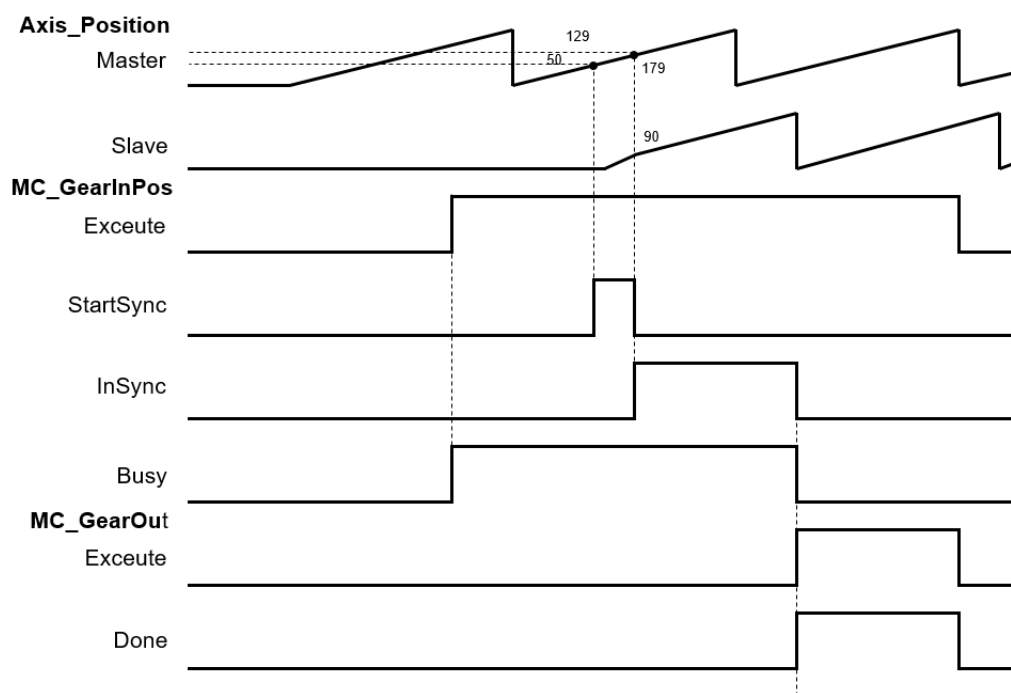
在 MC_Stop 运行期间，MC_GearOut 启动，主从轴脱离同步，从轴保持当下脱离时的速度继续运动，从轴状态机进入 Continuous Motion。

7.7.2.6 位置指定电子齿轮 (MC_GearInPos)

用户可以使用 MC_GearInPos 来指定主轴与从轴起始的同步位置。

- MC_GearInPos 时序

MC_GearInPos 中指定主轴与从轴、齿轮比分子、齿轮比分母、主轴与从轴同步位置、开始执行同步的主轴距离与是否允许反转。功能块会依据设定自行规划从轴的运动曲线，使主轴与从轴在指定位置完成啮合。



主轴开始进行同步的位置为 $\text{MasterSyncPosition} (180) - \text{MasterStartDistance} (50)$ ，当主轴到达该位置后， $\text{StartSync} \Rightarrow \text{True}$ 开始进行同步。

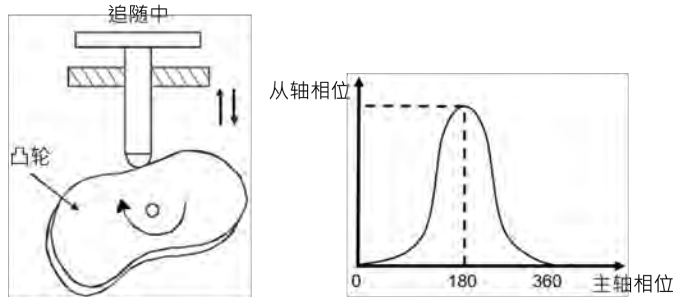
从轴则会依据其它参数自行规划一条运动曲线，并在主轴到达 $\text{MasterSyncPosition} (180)$ 时，同时从轴位也到达 $\text{SlaveSyncPosition} (90)$ ， $\text{StartSync} \Rightarrow \text{False}$ 、 $\text{InSync} \Rightarrow \text{True}$ 同步完成。

若 $\text{MasterStartDistance} \leq 0$ ，功能块一执行就同步完成，此时从轴位置会直接跳动到指定从轴同步位置。

若机构上不允许从轴反转，用户须将 AvoidReversal 设置为 True 。

7.7.2.7 电子凸轮 (MC_CamIn)

从轴根据凸轮表的信息跟随主轴作同步运动。通过 MC_CamTableSelect 预先指定电子凸轮运行时所遵循的凸轮表，并指定主从、从轴。通过 MC_CamIn 指令啮合凸轮，MC_CamOut 来解除齿轮啮合。



啮合后，主从轴顺利完成同步，此时从轴的状态机来到 Synchronized Motion。以下说明电子凸轮建立相关信息：

- 初始设定

● 建立电子凸轮数据

电子凸轮的曲线数据建立可通过以下两种方法绘制：

方法 1：根据标准函数决定主从轴位置的关系。

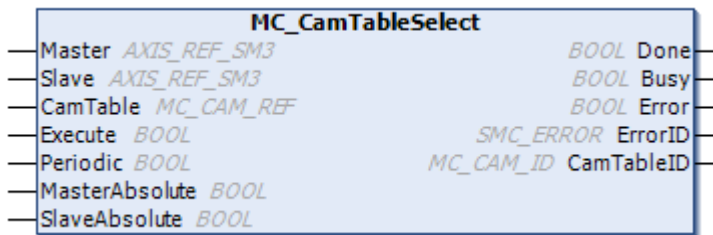
方法 2：根据实际工作中测量到主从轴之间的对应关系。

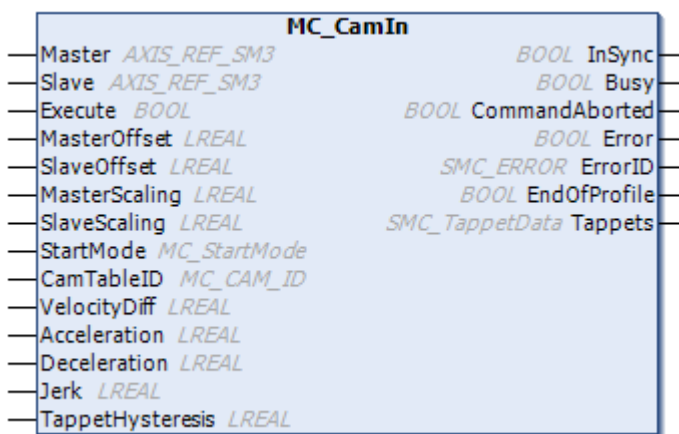
- 电子凸轮主从轴的设定及操作

电子凸轮的主从轴设定与电子凸轮的操作，需通过 MC_CamIn、MC_CamTableSelect 即可达成。

● 主轴与从轴来源设定

通过 MC_CamTableSelect、MC_CamIn 功能块的输入引脚 Master 决定主轴来源，Slave 决定从轴。



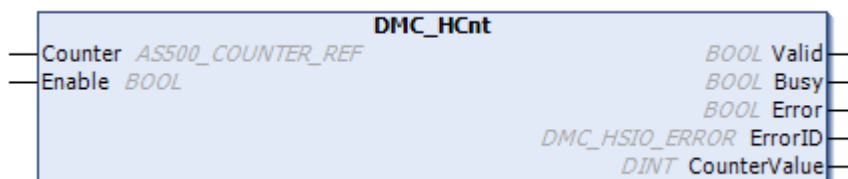
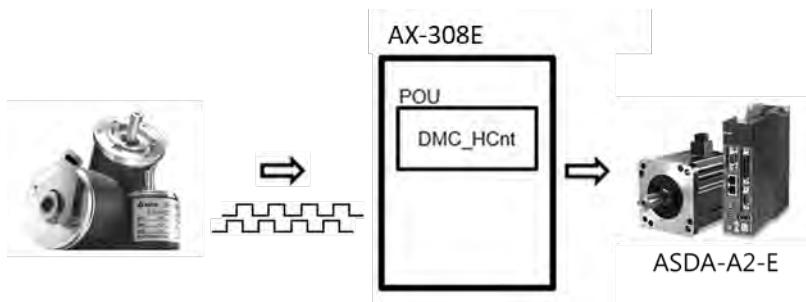


注：详细引脚功能定义，请参考 AX 系列运动控制指令手册

- 主轴为外部计数器

电子凸轮的主轴来源可分为实轴及虚拟轴与计数器。如需使用外部计数器时当主轴来源时，则需使用 DFB_HCnt 功能块启动外部计数功能。

系统架构与 DMC_HCnt 功能



- 主轴位置与从轴位置之间的关系

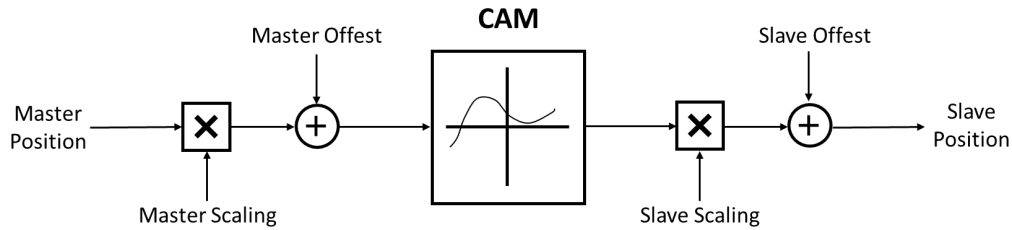
通过软件预先规划的凸轮关系是主从轴之间的位置关系，其凸轮表的位置为主从轴的凸轮相位，而非实际轴位置。若将预先规划的凸轮关系视为函数 CAM，则函数 CAM 的输入为主轴凸轮相位，输出为从轴凸轮相位，如下所示：

x：主轴凸轮相位 y：从轴凸轮相位

$$y = CAM(x)$$

凸轮相位来源于轴位置，它们之间存在换算关系。轴位置与凸轮相位之间的换算关系与参数 MasterAbsolute、SlaveAbsolute、MasterOffset、SlaveOffset、MasterScaling、SlaveScaling 有关。

从轴在 MC_CamIn 指令的作用下跟随主轴做凸轮同步运动。凸轮同步运动中，主从轴的轴位置对应关系建立在预先规划的凸轮关系（凸轮关系曲线或凸轮表）基础之上，由主轴轴位置计算从轴轴位置的过程如下图所示：



由上图得出计算方法如下：

$$\text{Position_Slave} = \text{SlaveScaling} \times \text{CAM} (\text{MasterScaling} \times \text{MasterPosition} + \text{MasterOffset}) + \text{SlaveOffset}$$

当主轴是绝对模式，主轴位置是主轴现在位置的旋转轴运算结果；当主轴是相对模式，主轴位置是相对应凸轮的起始点（通常是 0）。

● StartMode 与 CamTableSelect 的 MasterAbsolute、SlaveAbsolute 之关系

绝对模式（StartMode = 0）：在电子凸轮同步开始时，凸轮的计算与从轴当前的位置无关。如果从轴当前位置与计算后的起始位置不同，则造成跳动。

相对模式（StartMode = 1）：凸轮会根据当前从轴的位置进行改变；从轴位置会被当前从轴位置进行位置相加的计算。如果从轴啮合时位置与计算后的起始位置加当前位置不同，则造成跳动。

ramp 模式（StartMode = 2、3、4）：依据 VelocityDiff、Acceleration、Deceleration、Jerk，来增加一个补偿运动曲线，防止凸轮在开始啮合时的跳动。

MC_CamTableSelect.MasterAbsolute	主轴模式
absolute	绝对模式
relative	相对模式

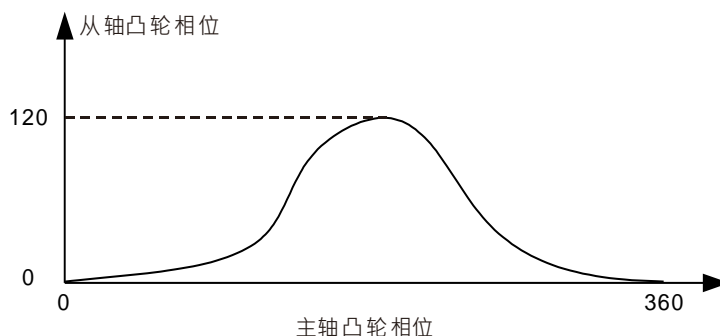
7

MC_CamIn.StartMode	MC_CamTableSelect.SlaveAbsolute	从轴模式
absolute	True	绝对模式
absolute	False	相对模式
relative	True	相对模式
relative	False	相对模式
ramp_in	True	Ramp in 绝对模式
ramp_in	False	Ramp in 相对模式
ramp_in_pos	True	正向 Ramp in 绝对模式
ramp_in_pos	False	正向 Ramp in 相对模式
ramp_in_neg	True	反向 Ramp in 绝对模式
ramp_in_neg	False	反向 Ramp in 相对模式

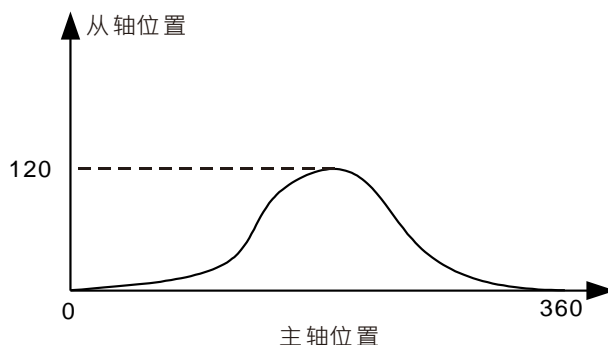
- 偏移量与缩放 (MasterOffset/MasterScaling/SlaveOffset/Slavescaling)

主从轴的凸轮关系为预先规划，但在执行凸轮时，可通过参数 **Offset** 和 **Scaling** 在预先规划的凸轮关系基础上进行位置偏移或位置缩放，例如：加工的同一产品有几种不同尺寸，则只需规划一种凸轮关系，然后通过改变参数 **Offset** 和 **Scaling** 以适应不同尺寸产品之间的加工切换。用户可指定缩放系数去缩放凸轮表的主轴相位与从轴位移量。主轴与从轴可分别设定个别的偏移量与缩放系数。

主从轴的位置偏移和比例缩放共同决定实际执行的凸轮关系，其作用效果通过以下范例进行描述。预先规划的凸轮关系如下图所示：

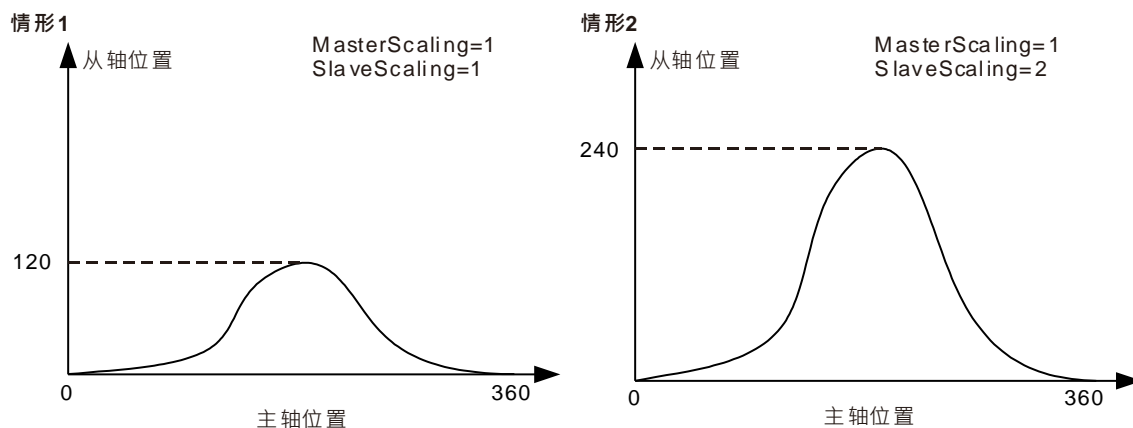


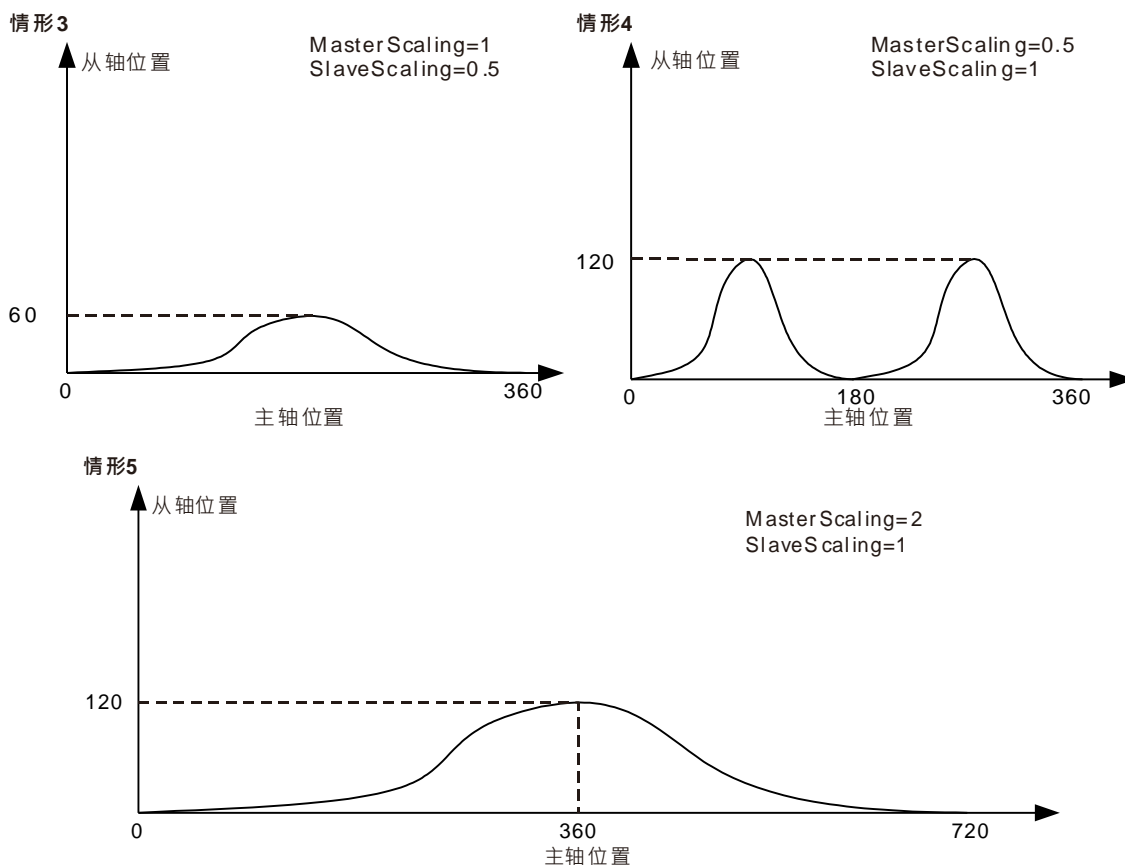
若主从轴均为绝对模式，且执行啮合动作时，主从轴的轴位置均为 0，在不使用偏移和缩放时（默认值），凸轮执行过程中的主从轴实际位置对应关系如下图所示：



当位置偏移量或缩放比例不为默认值时，其对凸轮执行过程中主从轴实际位置对应关系的影响如下：

主从轴偏移量为 0，主从轴缩放比例对实际执行的凸轮关系的影响。

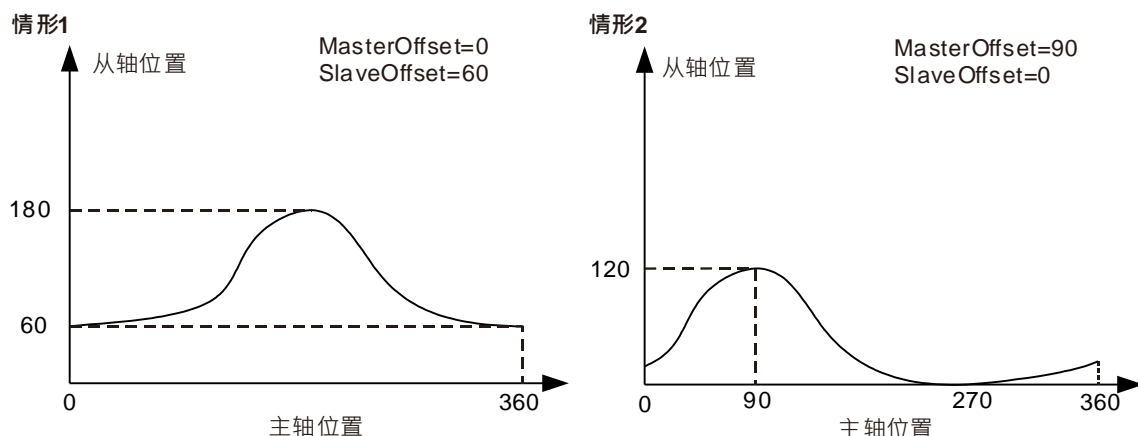




- 情形 1：当主从轴缩放比例为 1，偏移量为 0 时，实际凸轮关系与预先规划的一致。
- 情形 2：当主轴缩放比例为 1，从轴缩放比例为 2，主从偏移量为 0 时，与主轴位置对应的从轴位置变为预先规划的两倍。
- 情形 3：当主轴缩放比例为 1，从轴缩放比例为 0.5，主从轴偏移量为 0 时，与主轴位置对应的从轴位置变为预先规划的 1/2。
- 情形 4：当主轴缩放比例为 2，从轴缩放比例为 1，主从轴凸轮偏移量为 0 时，与从轴位置对应的主轴位置变为预先规划的两倍。如果从凸轮相位的角度看，则是主轴的凸轮相位为预先规划的两倍，即主轴凸轮周期从 360 变为 180，从轴凸轮相位不变。
- 情形 5：当主轴缩放比例为 0.5，从轴缩放比例为 1，主从轴偏移量为 0 时，与从轴位置对应的主轴位置变为预先规划的 1/2。如果从凸轮相位的角度看，则是主轴凸轮相位为预先规划的 1/2，即主轴凸轮周期从 360 变为 720，从轴凸轮相位不变。

主从轴缩放比例为 1，主从轴偏移实际执行的凸轮关系的影响。

主轴偏移相当于将执行凸轮时的实际轴位置关系曲线作横向移动；从轴偏移相当于将执行凸轮时的轴位置关系曲线作纵向移动。

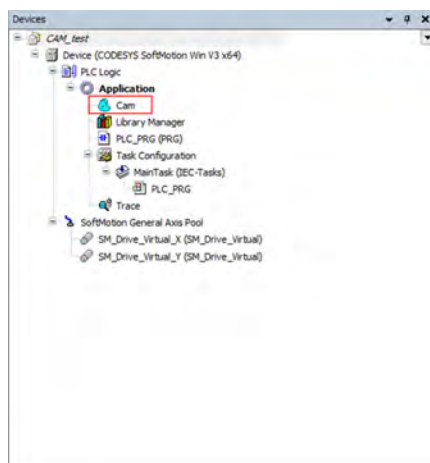


情形 1：当主从轴缩放比例为 1，主轴偏移量为 0，从轴偏移量为 60 时，与主轴位置对应的从轴位置均在预先规划的基础上加上 60。例如：规划的凸轮关系中，主轴轴位置 180 对应的从轴轴位置为 180，实际执行时，对应的从轴轴位置为 240 ($240=180+60$)。

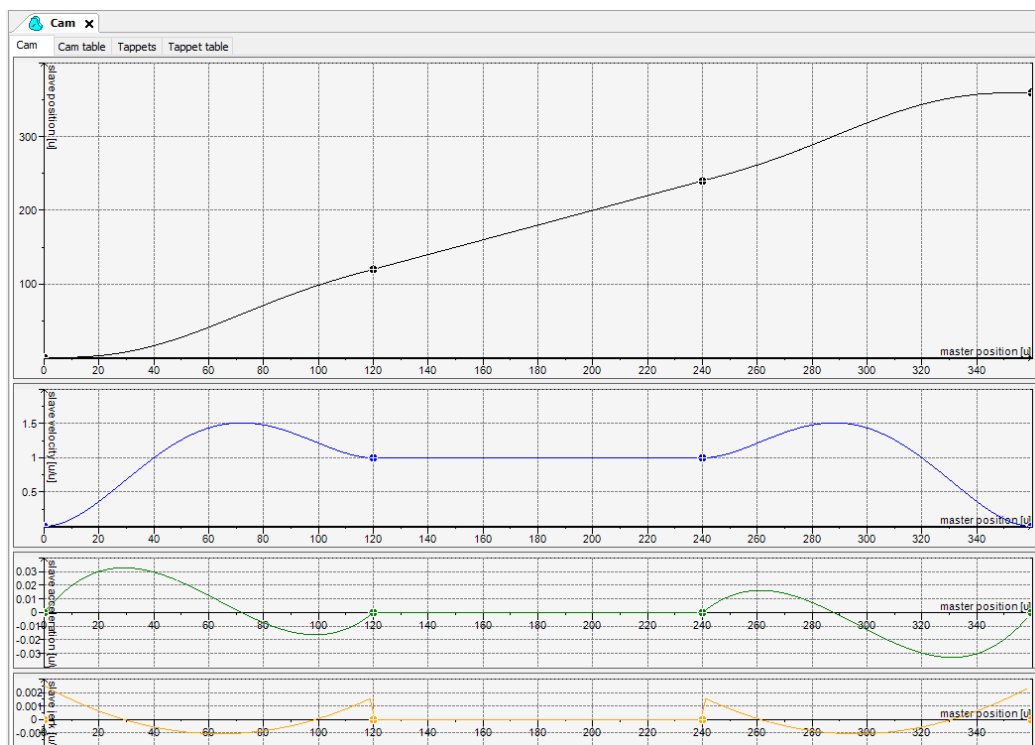
情形 2：当主从轴缩放比例为 1，主轴偏移量为 90，从轴偏移量为 0 时，与从轴轴位置对应的主轴轴位置在预先规划的基础上偏移 90 (加上偏移量)。例如：规划的凸轮关系中，主轴轴位置 180 对应的从轴轴位置为 180，实际执行时，主轴轴位置 90 对应的从轴轴位置为 180，即预先规划的凸轮关系中主轴轴位置 180 ($180=90+90$) 所对应的从轴轴位。

- 凸轮表

在 DIADesigner-AX 项目树中，可建立 Cam 对象，在此对象中可以自行编辑凸轮曲线。而曲线型式决定了从轴凸轮运行的结果。



- 凸轮曲线表特点：
 - ◆ 可从凸轮曲线可直接观察对应的从轴运动范围与速度、加速度、加速度的变化率（跃度）的任一时刻变化。在每一个关键点之间可以设定曲线类型，可设定为直线或者是五次曲线。
 - ◆ 主轴的起始坐标默认从 0 开始到 360 结束，用户可以根据实际的物理行程进行修改。
- 凸轮曲线编辑方式
 - ◆ DIADesigner-AX 图型编辑方式：



用户以图形方式的编辑 CAM 凸轮表，横坐标表示主轴位置，凸轮的运行行程取决于主轴的轴长度，画面中共有 4 条曲线，分别表示位置、速度、加速度、加速度的变化率（跃度）速度曲线。在设计凸轮时，可以藉由位置曲线、速度曲线来进行运动行程上的调整，调整加速度曲线能使运动更加平稳。

◆ DIADesigner-AX 表格编辑方式：

软件上除了提供图形方式编辑，也可以在 Cam Table 直接进行关键点的增减与位置上的修改。

Cam	Cam table	Tappets	Tappet table								
	X	Y	V	A	J	Segm...	min(P...	max(P...	max(V...	max(A...	
	0	0	0	0	0						
	120	120	1	0	0	Poly5	0	120	1.5120...	0.0328...	
	240	240	1	0	0	Poly5	120	240	1	0	
	360	360	0	0	0	Poly5	240	360	1.512	0.0328...	

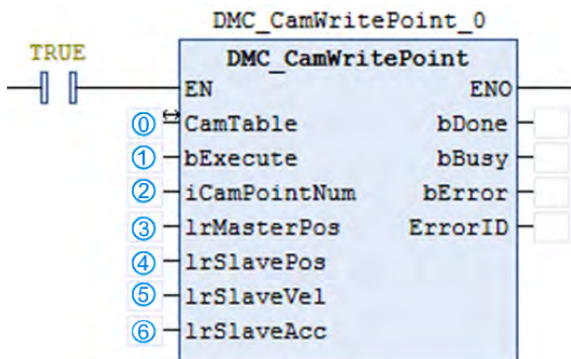
◆ 程序编辑：

用户也可以用程序来修改 CAM 凸轮表的关键点，若如下面程序修改，此时该凸轮表的起始位置（主轴·从轴）从 (0·0) → (0·30)，但是软件上的画面不会随之修改。

在程序中修改 CAM 表可通过 DMC_CamWritePoint 功能块修改，如下说明：

- ① 指定的凸轮表
- ② 启动功能块
- ③ 选择读取的凸轮点编号
- ④ 凸轮主轴的位置

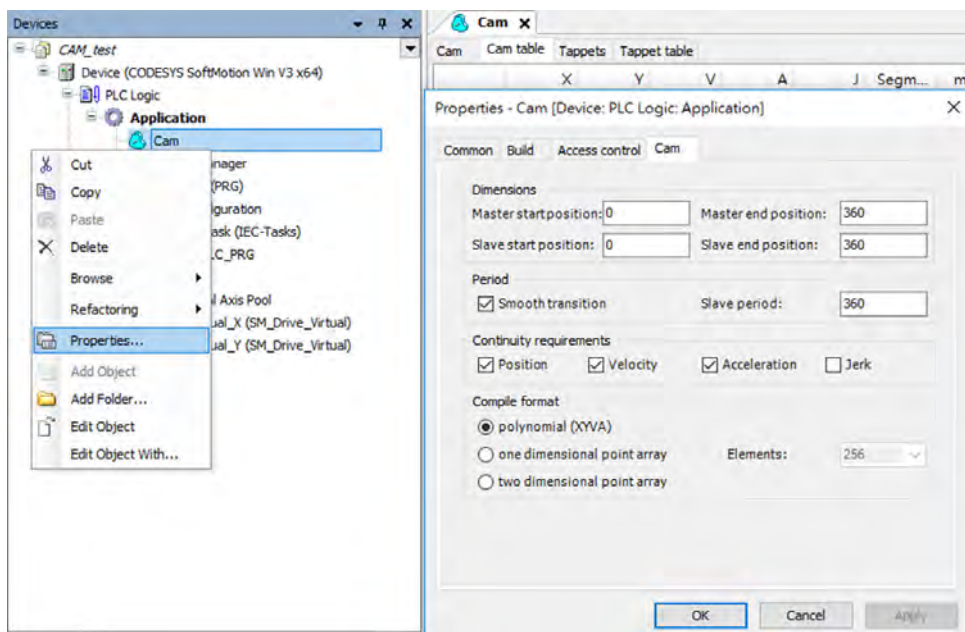
- ④ 凸轮从轴的位置
- ⑤ 凸轮从轴的速度
- ⑥ 凸轮从轴的加速度



注：详细功能块说明请参考 AX 系列运动控制指令手册。

● 凸轮表属性设置

用户可以在 Properties 窗口中，调整凸轮表的属性，如主从轴的起始与结束位置、周期性设定参数，曲线连续性的需求和编译型式。



● 电子凸轮使用步骤：

配置凸轮表：设定主轴范围、从轴范围，建立起始点、结束点与其它关键点，调整曲线类型。

使用 MC_CamTableSelect 指令将配置的凸轮表连接到实际凸轮表，并得到 Cam ID 让后续指令使用。

获得 Cam ID 后，使用 MC_CamIn 指令对指定的主轴与从轴进行啮合。

使用 MC_CamOut 指令可以使两轴脱离主从轴关系。此外在同步运动时，对从轴下达 MC_Stop 与 MC_Halt 皆可以使主轴与从轴脱离同步关系。

● 凸轮表切换

凸轮表运行时可以进行参考凸轮表的切换，藉由 MC_CAM_REF 来更换 MC_CamTableSelect 的凸轮表的参考。

◆ 变量宣告

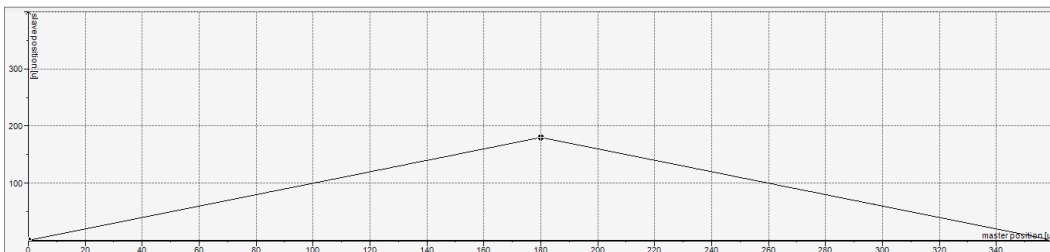
```
P : MC_CAM_REF; //CamTable reference
CamTableID : INT; //CamTable Switch
```

◆ 切换凸轮表

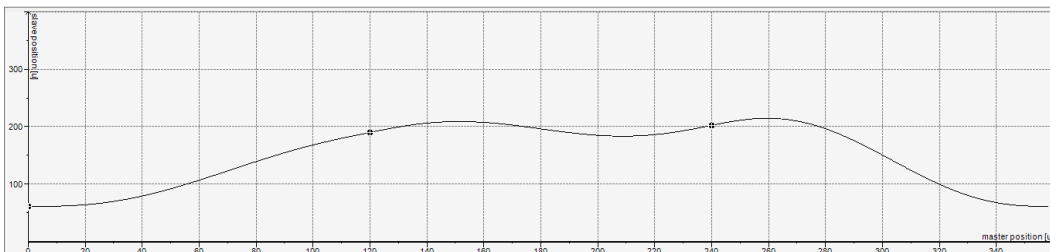
```
CASE CamTableID OF
  0: P:=Cam;
  1: P:=Cam_1;
END_CASE
```

上面的范例程序，利用 CamTableID 的切换来改变 MC_CAM_REF，便可实现多个凸轮表的切换操作了。两张凸轮表分别如下：

第一张凸轮表 (Cam)

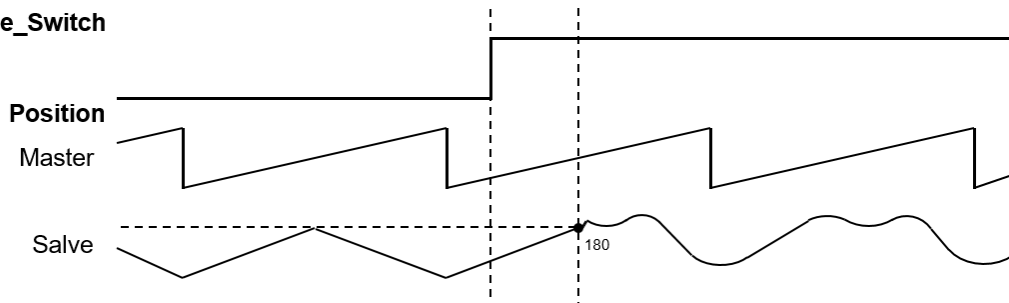


第二张凸轮表 (Cam_1)



凸轮表切换时序图

Cam Table_Switch



在切换凸轮表时，从轴依然遵循第一张凸轮表所规划的运动轨迹，直到主轴位置到达第一张凸轮表下一个关键点时，才会开始遵循第二张凸轮表的运动轨迹。

7.7.3 速度控制

速度控制可分类为三种伺服控制模式周期同步位置控制 (Cyclic Synchronous Position · CSP) 与周期同步速度控制 (Cyclic Synchronous Velocity · CSV) 及速度规划模式 (Profile Velocity Mode)。

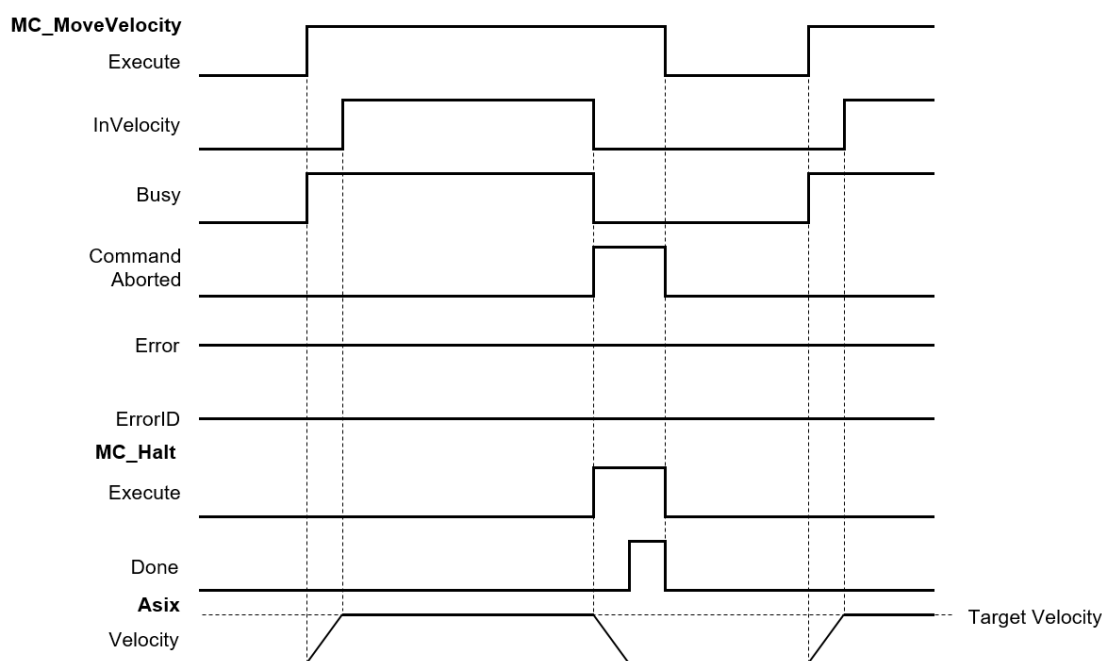
7.7.3.1 CSP 模式

如第 7.7.2.1 节的介绍，CSP 模式为周期同步定位模式。此模式下的速度控制，是控制器依照指定速度（以及加速度、减速度、加速度的变化率（跃度））算出每个周期应该走到的命令位置，然后在每周期把命令位置下给伺服执行定位动作。

此模式下进行速度控制时，如果因外力干扰导致当前伺服位置落后于控制器命令位置时，会为了弥补这些位置误差会出现速度抖动的现象。

使用 MC_MoveVelocity 运动功能块指令即为在 CSP 模式下执行速度控制，执行时轴的状态机会进入 continuous_motion 状态。可以指定速度调整时（到达指定速度前、缓冲时）的加速度、减速度、跃度。需要停止时，可以使用 MC_Stop、MC_Halt、或者是其它运动指令来中断。

下图为使用 MC_MoveVelocity 进行速度控制时，以 MC_Halt 来进行中断的时序图：

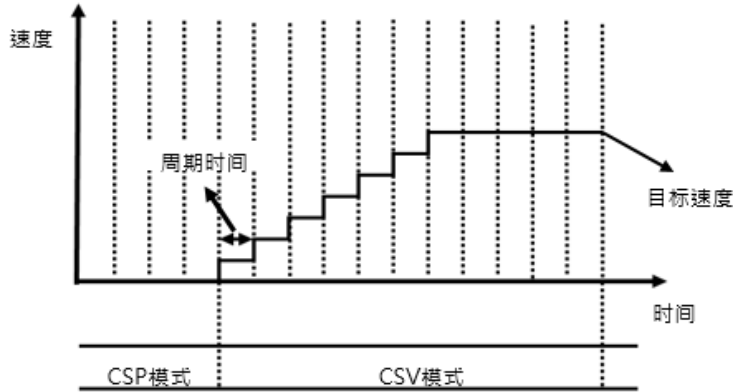


可以指定速度为 0，此时虽然静止不动，但系统状态机仍会在 continuous_motion 状态中。

AX 系列中可使用 MC_MoveVelocity 进行在 CSP 模式下的单轴速度控制，关于运动功能块细节请参考 **AX 系列运动控制指令手册**。

7.7.3.2 CSV 模式

CSV 模式为周期同步定速模式。此模式下的速度控制，是控制器依照指定速度（以及加速度、减速度、跃度）算出每个周期的行走速度，然后每个周期下指令给伺服执行定速动作。

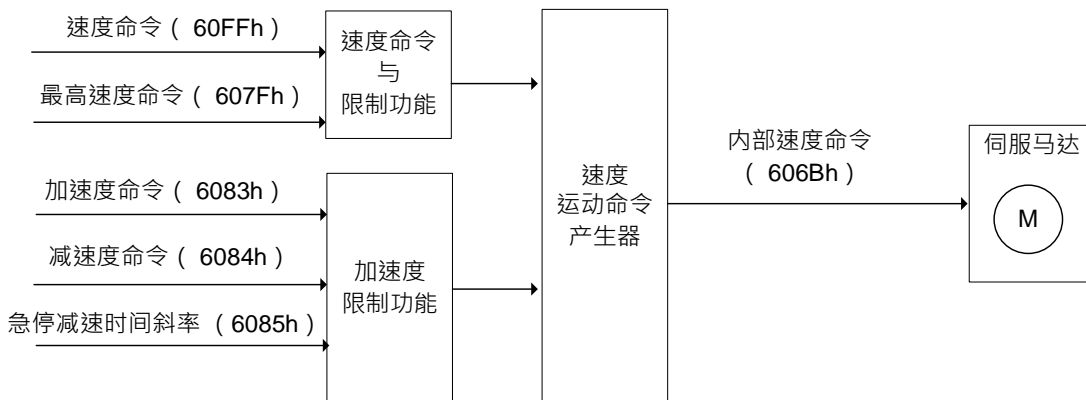


此模式下进行速度控制时，因为每个周期都是下命令速度给伺服，即使因外力干扰，也不会出现如 CSP 模式的为了弥补位置差距导致速度变动的现象。

AX 系列中可使用 MC_VelocityControl 进行在 CSV 模式下的单轴速度控制，关于功能块细节请参考 **AX 系列运动控制指令手册**。

7.7.3.3 速度规划模式 (Profile Velocity Mode)

在 PV (速度规划) 模式*下，上位机指定速度命令与加减速等条件，并由驱动器的运动命令产生器依据这些条件规划出运动轨迹。



注：速度规划模式使用于定位轴。

7.7.4 扭矩控制

扭矩控制可分类为两种伺服控制模式周期同步扭矩控制模式 (Cyclic Synchronous Torque · CST) 与扭矩控制模式 (Profile Torque · PT) 。

- 扭矩控制模式* (Profile Torque · PT)

用户可通过 `DMC_TorqueControl` 来指定单轴持续产生指定扭矩输出。

使用注意事项：

在使用 `DMC_TorqueControl` 时，会将控制模式切换为周期同步扭矩模式。

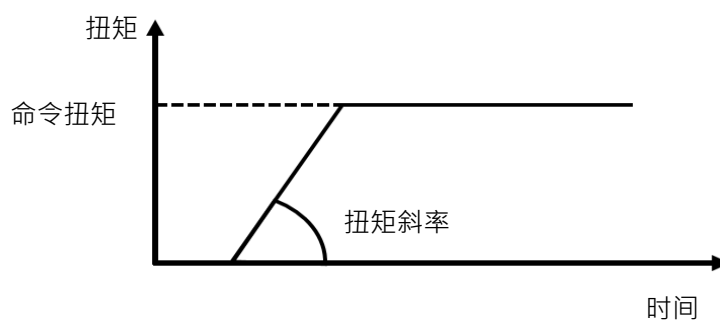
在使用 `DMC_TorqueControl` 时，控制模式会切换成扭矩模式，扭矩模式下不能使用位移或速度系列的功能块。不能用 `MC_Stop` 来停止电机，若要停止电机直接将 `DMC_TorqueControl Enable` 为 `False` 时，可立即停止。

`Torque` 不能设为 0，若设置 0，`DMC_TorqueControl` 会报错。

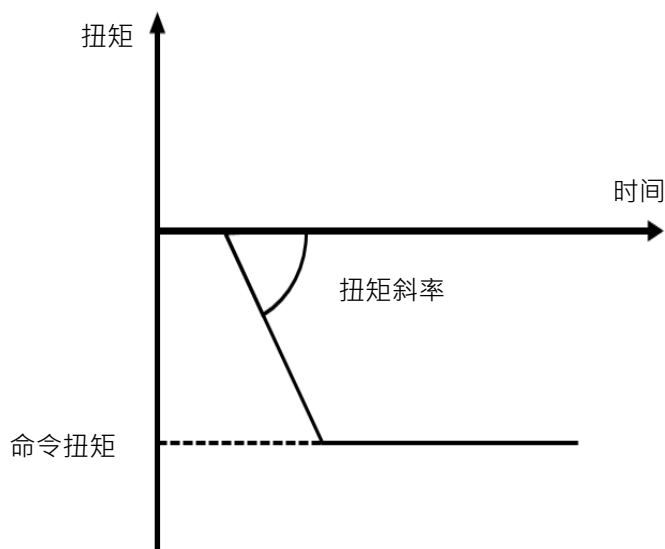
设定 `DMC_TorqueControl` 的 `Velocity` 可以向伺服设定电机的最大速度限制，通过此设定可避免在扭矩控制模式当电机负载变小而高速旋转。

通过扭矩斜率 (`TorqueRamp`) 来设定，到达目标扭矩值 (`Torque`) 的扭矩变化率。

`Torque > 0` 时，电机正方向运转



`Torque < 0` 时，电机反方向运转



*注：

1. 台达服务器ASDA-A3-E系列于V1.1165版本以上支持力矩正确数值
2. 台达服务器ASDA-B3-E系列于V1.0665版本以上支持力矩正确数值

7.7.5 单轴控制的常用功能

下面针对单轴运动控制时的常用项目进行说明。

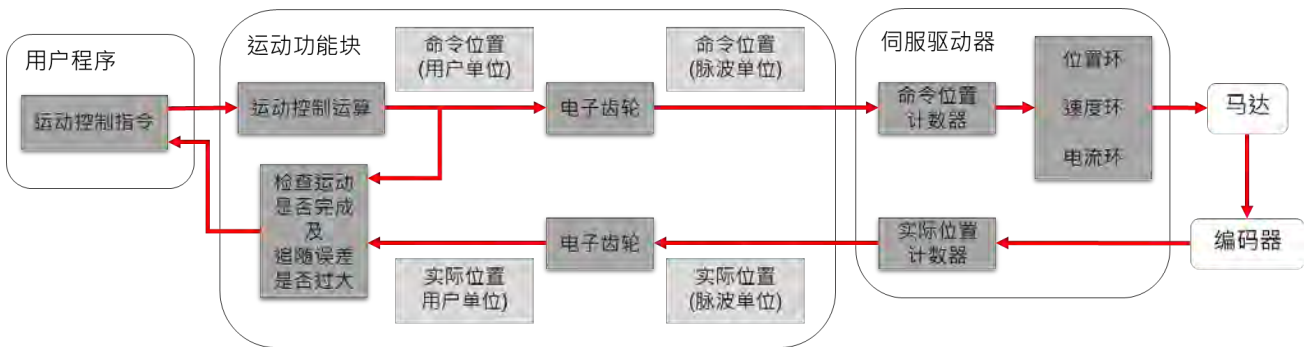
7.7.5.1 位置命令

轴运动中的位置可细分为以下 2 种类型。

位置类型	定义
命令位置	运动功能块输出用以控制轴的位置
实际位置	从伺服驱动器反馈的位置*

*若为虚拟轴，因无伺服驱动器反馈位置，实际位置直接由命令位置填入。

两者之间的关系如下图所示：



两者之间的异同比较：

设定	命令位置	实际位置
计数模式	直线轴 / 旋转轴	同命令位置设定
用户单位	长度单位 (m · mm · inch...) / 角度单位 (degree) / ...	同命令位置设定
软件极限	设定运动指令可到达的位置范围限制	同命令位置设定
定位	可定位至限定范围中任意位置	同命令位置定位之位置，但可能会有追随误差*

*注：

受到机构及伺服设定影响，命令位置与实际位置两者之间可能会有些微差距，称为追随误差(Position Lag)。追随误差随着运动速度增加可能会有些微增加。若要限制此差距，可调整轴的设定去监看追随误差，并设定追随误差过大时的动作。若为虚拟轴则实际位置等于命令位置，不会有追随误差。

相关参数说明如下：

- 位置单位：位置单位为用户单位。
- 追随误差设定

设定	设定值	意义
Position lag supervision	deactivated	不检查追随误差
	disable drive	追随误差超出设定极限时，轴会 servo off。
	do quick stop	追随误差超出设定极限时，轴会 quick stop。
	stay enabled	追随误差超出设定极限时，轴维持 servo on。
Lag limit (用户单位)	LREAL	可容许的误差极限

除了 deactivated 外其它设定下超出误差极限时，轴都会报错 SMC_ERROR.SMC_DI_POSITIONLAGERROR。

- 软件极限

设定	设定值	意义
Software limits Activated	Checked / Unchecked	是否启动软极限
Negative (用户单位)	LREAL	反方向的软极限
Positive (用户单位)	LREAL	正方向的软极限

- 运动指令的位置描述

运动功能块中跟位置相关的输入变量有两种不同的描述，使用时须注意：

项目	意义
Position	目标位置 (绝对位置)
Distance	移动距离 (相对位置)

- 位置的监控

若需观察位置变化，可针对轴变量 (数据类型为 AXIS_REF_SM3) 的下述两个变量做监控：

变量名称	位置类型	数据类型
.fSetPosition	命令位置	LREAL
.fActPosition	实际位置	LREAL

7.7.5.2 速度命令

轴运动中的速度可细分为以下二种类型。

位置类型	定义
命令速度	运动功能块输出用以控制轴的速度
实际速度	根据伺服驱动器在各时间点反馈的实际位置算出的速度*

*注：若为虚拟轴，因无伺服驱动器则实际位置直接由命令位置填入

相关参数说明如下：

- 速度单位：速度单位：用户单位/秒。
- 速度曲线类型

设定	设定值	意义
Velocity ramp type	Trapezoid	速度曲线为梯形 (每个线段的加速度都是常数)
	Sin ²	速度曲线为 sin ² 函数 (固定的加速度曲线)
	Quadratic	加速度曲线为梯形 (加速度被加速度的变化率 (跃度) 限制) 速度曲线
	Quadratic (smooth)	同Quadratic但加速度的变化率 (跃度) 曲线为连续的速度曲线

● 运动指令的速度描述

运动功能块中跟速度相关的输入变量如下：

项目	意义
Velocity	目标速度*

*注：若因为路径长度不足或加速度、加速度的变化率 (跃度) 过小等因素，会导致运动中无法达到目标速度。

● 速度的监控

如需观察速度变化，可针对轴变量 (数据类型为AXIS_REF_SM3) 的下列两个变量做监控：

变量名称	位置类型	数据类型
.fSetVelocity	命令速度	LREAL
.fActVelocity	实际速度	LREAL

7.7.5.3 加速度与减速度命令

轴运动中的加速度可细分为以下2种类型。

位置类型	定义
命令加速度	运动功能块输出用以控制轴的加速度
实际加速度	根据实际速度算出的加速度

相关参数说明如下：

- 加速度单位：加速度单位为用户单位/秒²。
- 加速度曲线类型：请参考本手册第7.7.5.2节。
- 运动指令的加速度描述

运动功能块中跟加速度/减速度相关的输入变量如下：

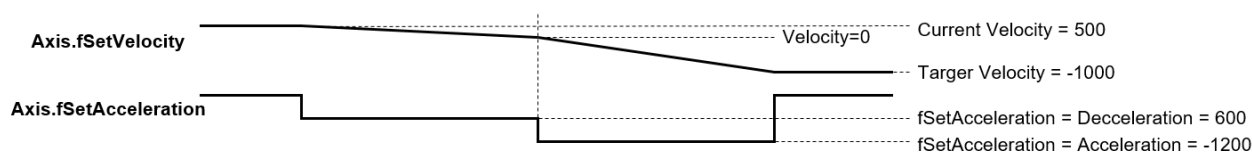
项目	意义
Acceleration	目标加速度*
Deceleration	目标减速度*

*注：

可能因为路径长度不足，加速度的变化率 (跃度) 过小等因素，导致运动中无法达到目标加速度、目标减速度。

加速度与减速度的套用标准为，当目前速度的绝对值需要变小时，会套用Deceleration；当目前速度的绝对值需要变大时，会套用Acceleration。

举例来说，若轴目前速度为 500，执行运动指令为反向运动（Velocity = 1000，Acceleration = 1200，Deceleration = 600），则速度曲线与加速度曲线会如下所示：



- 加速度的监控

如需观察加速度变化，可针对轴变量（数据类型为 AXIS_REF_SM3）的下列两个变量做监控：

变量名称	位置类型	数据类型
.fSetAcceleration	命令加速度	LREAL
.fActAcceleration	实际加速度	LREAL

7.7.5.4 加速度的变化率（跃度）命令

跃度指定加速度/减速度的变化比率。指定加速度的变化率（跃度）后，速度曲线为 S 型（加减速曲线为递增递减，无加速度的变化率（跃度）），可减小机械冲击。轴运动中的加速度可细分为以下 2 种类型：

位置类型	定义
命令加速度的变化率（跃度）	运动功能块输出用以控制轴的加速度的变化率（跃度）
实际加速度的变化率（跃度）	根据实际加速度算出的加速度的变化率（跃度）

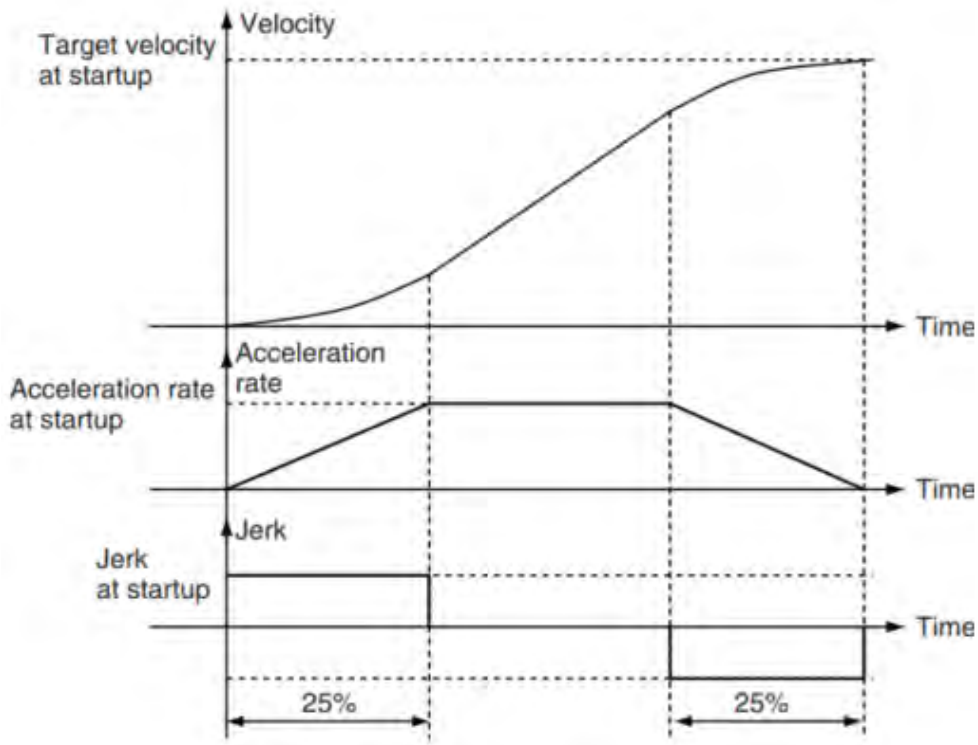
相关参数说明如下：

- 加速度的变化率（跃度）单位：跃度单位为用户单位/秒³。
- 加速度的变化率（跃度）曲线类型：请参考本手册第 7.7.5.2 节。
- 运动指令的加速度的变化率（跃度）描述

运动功能块中跟跃度相关的输入变量如下：

项目	意义
Jerk	目标加速度的变化率（跃度）*

*注：当 Velocity ramp type 为 Trapezoid 或 Sin² 时 Jerk 设定值不会对运动起作用；当 Velocity ramp type 为 Quadratic 或 Quadratic smooth 时 Jerk 数值设定才会影响速度曲线规划。



- 加速度的变化率（跃度）的监控
如需观察加速度的变化率（跃度）变化，可针对轴变量（数据类型为AXIS_REF_SM3）的下列两个变量做监控：

变量名称	位置类型	数据类型
.fSetJerk	命令加速度的变化率（跃度）	LREAL
.fActJerk	实际加速度的变化率（跃度）	LREAL

7.7.5.5 轴方向

在下述情形时，运动需指定操作方向：

- 轴作等速运动时输入之 Velocity 为绝对值，方向需由 Direction 来指定。
- 轴设定为旋转轴时，往正反两方向运动皆可移动至目标位置，因此需指定运动方向。

运动功能块中跟方向相关的输入变量如下：

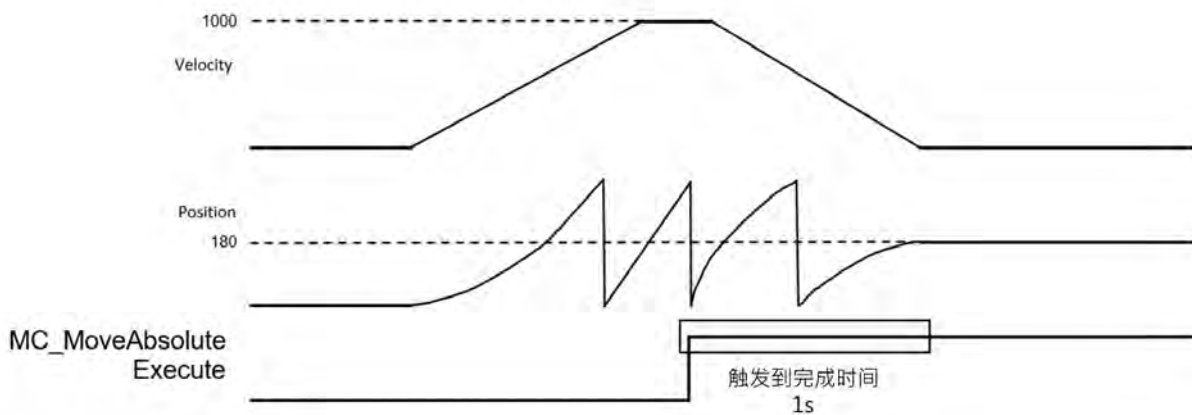
项目	设定值	意义
Direction	negative	往反方向运动
	shortest	往最短路径方向运动（只对旋转轴生效）
	positive	往正方向运动
	current	依照目前运动方向运动（只对旋转轴生效）
	fastest	往费时最短的路径方向运动（只对旋转轴生效）*

*注：

shortest（移动长度最短）与fastest（移动时间最短）概念相似但不完全相等，可参考下述例子：

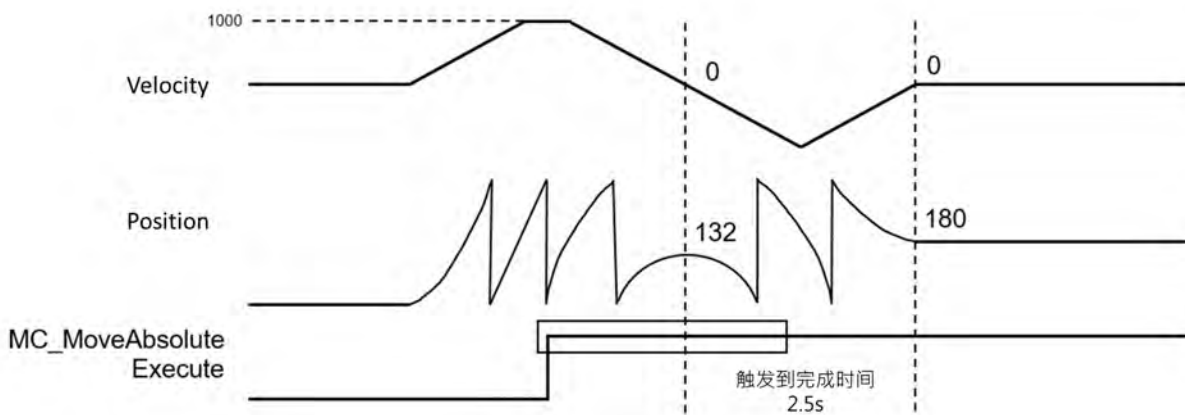
- 设定：
 - 设定轴为旋转轴，范围为360°。
 - 为方便观察，设定轴的Velocity ramp type为Trapezoid。
- 操作步骤：
 - 使用MC_MoveVelocity进行等速运动 (Velocity = 1000)
 - 当电机走到 350，并且速度到达 1000 时，同样情况下执行两次不同方向设定的 MC_MoveAbsolute，执行

MC_MoveAbsolute (Position = 180 · Velocity = Acceleration = Deceleration = 1000 · Direction = fastest)



MC_MoveAbsolute.Execute 触发时系统判断到达位置 180 的费时最短的方向是沿着正方向减速至 0 过程耗时约 1 秒。

执行 MC_MoveAbsolute (Position = 180 · Velocity = Acceleration = Deceleration = 1000 · Direction = shortest)。



MC_MoveAbsolute.Execute 触发时系统判断到达位置 180 的最短路径是沿着反方向走 (350 - 180 = 170)。但由于过程中须让速度反转，反而多转了几圈。过程耗时约 2.5 秒。

7.7.6 轴组控制

一轴以上即可组成轴组运动，并通过 DIADesigner-AX 设定轴组。直线轴最多支持六轴，旋转轴支持三轴且额外三轴可作为跟随轴。

7.7.6.1 轴组直线插补

TransitionMode：轴组运动进行过程中，改变插补轨迹的方向可能会机台产生震动声响状况。可利用轴组运动控制指令的输入引脚变量 TransitionMode（过度模式）指令间的连接动作方法以减小上述震动声响状况。

- TransitionMode模式介绍

模式	描述
None	无效果（默认）
Overlap	第一段减速度重迭第二段加速度进行接续

- TransitionMode模式支持接续介绍

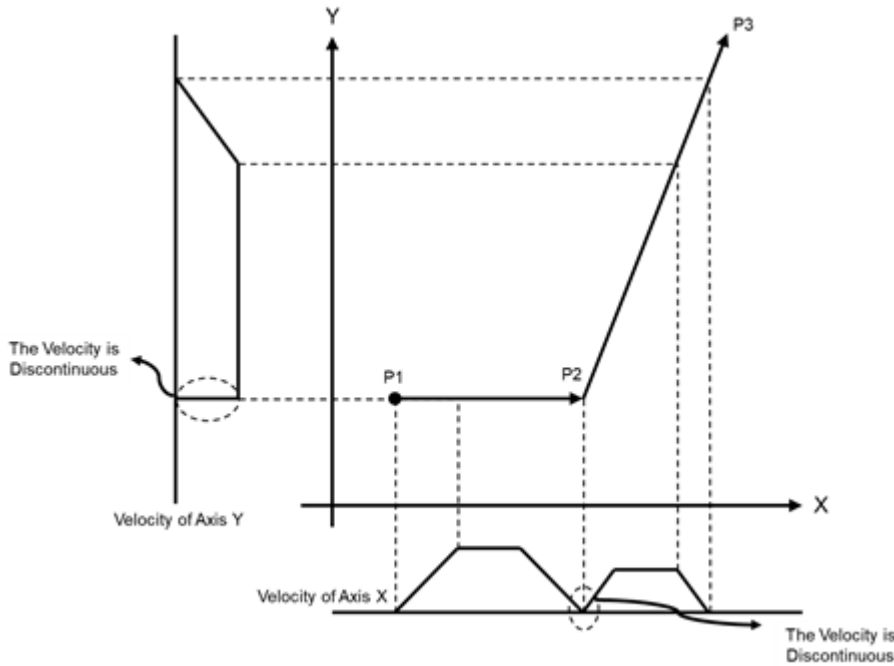
模式	Aborting	Buffered	Blending Low	Blending Previous	Blending Next	Blending High
None	A	A	N	N	N	N
Overlap	A	A	D	D	D	D

A = 支持

N = 不支持

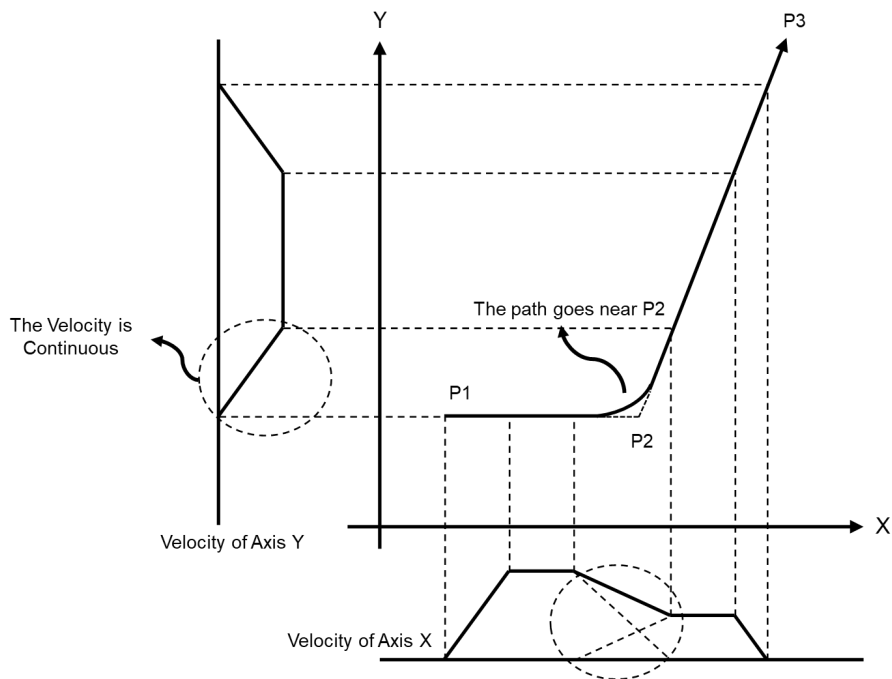
D = Blending 方式接续

- TransitionMode：设定None、Overlap接续选择Buffered为下列方式



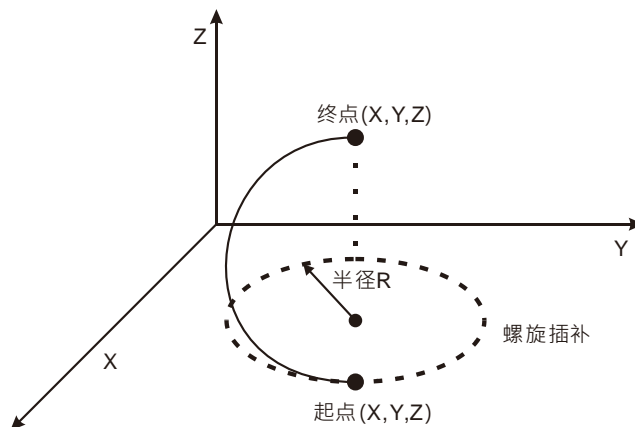
- TransitionMode：设定Overlap接续选择Blending为下列方式。

分别参考各轴组移动功能块的减速度与加速度做规划



7.7.6.2 轴组螺旋插补

螺旋插补支持 3 轴，在三维空间执行 X,Y,Z 轴与另外三轴跟随轴。

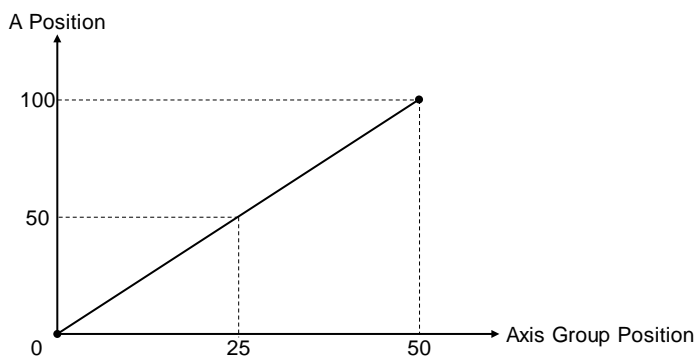


- 轴组追随轴概念介绍

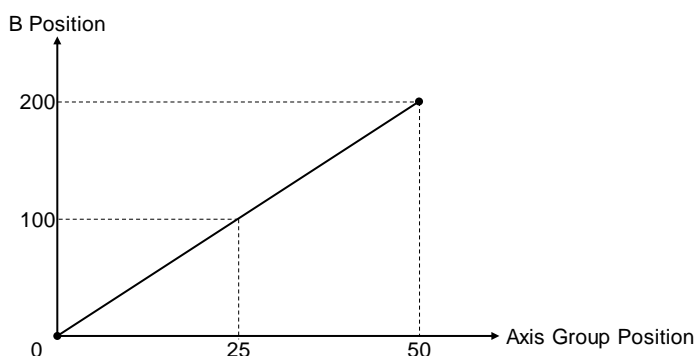
当组合轴 (X · Y · Z) 在运动过程中追随轴 (A · B · C) 会以等比例的方式同步运动。

轴组移动从位置 0 开始移动，分别为组合轴移动位置 (30 · 40 · 0)，组合轴的合成距离为 (50)，追随轴移动位置 (100 · 200 · 300)，在组合轴运动过程与追随轴的同步运动如下图所示：

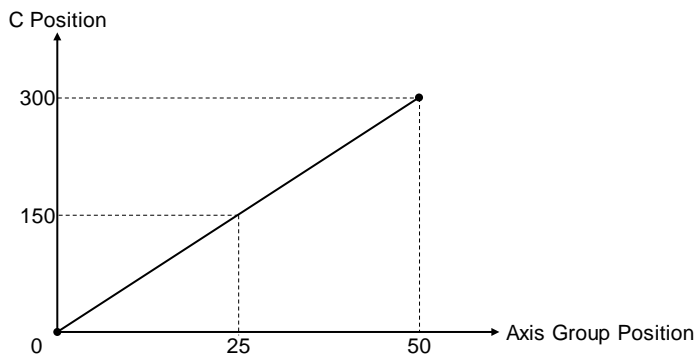
■ A 轴追随



■ B 轴追随



■ C 轴追随



注：当组合轴没有移动时，轴组功能块输入速度会使用于，追随轴较长距离的轴，其它追随轴依距离比例跟随同步运动。

7.7.6.3 轴组命令停止

轴组停止主要有分为程序停止与异常停止的方式，此两种方式停止方法不相同：

● 程序停止

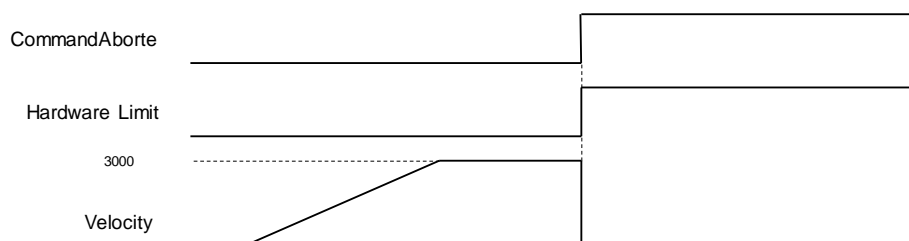
在程序中使用 `DMC_GroupStop` 将运动轴中轴组减速停止在当下位置，轴组状态会切换至 `GroupStopping`，在此状态无法执行任何轴组指令运动。

`IrDeceleration` 引脚需要输入数值，设定减速停止速度。

- 异常停止

当轴组运动时遇错误后，会以急停的方式停止运转。

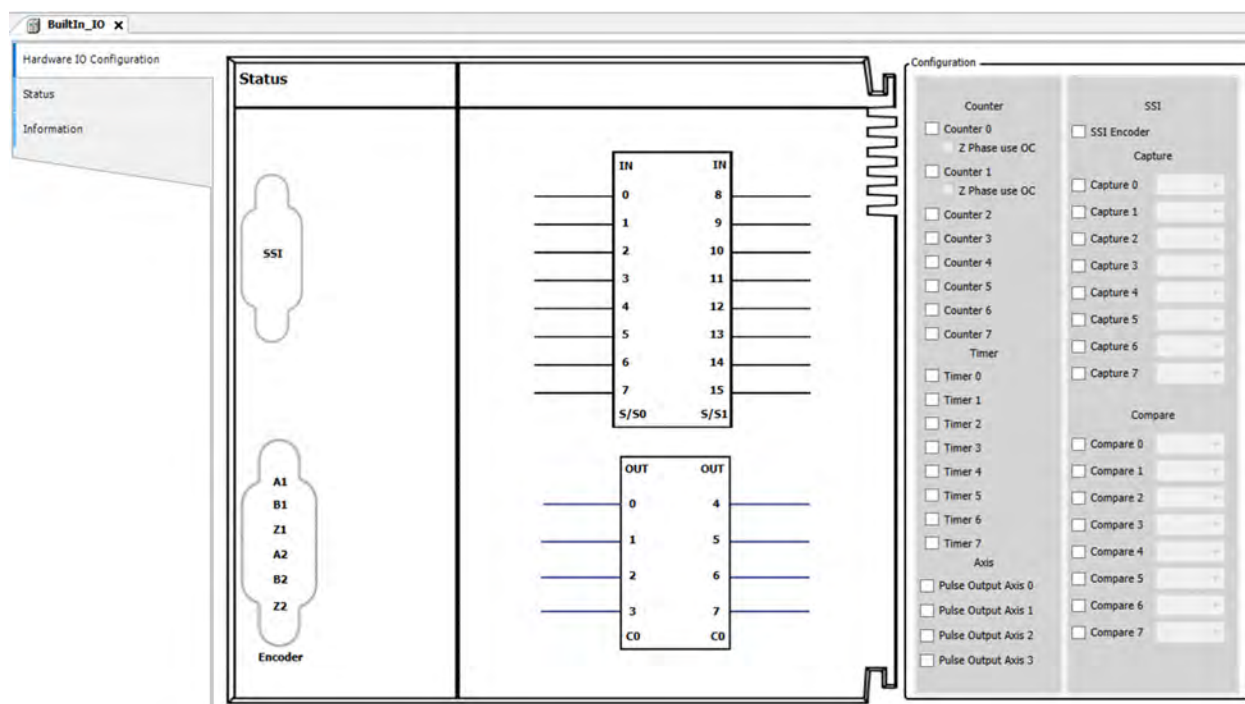
例如：当轴组运转时遇到硬件极限 (Hardware Limit)，功能块会报中断命令 (CommandAborted) 此时速度 (Velocity) 会以急停的方式降至为 0。



7.7.7 本体高速 IO

本章节描述 AX 系列本体的 IO 点位，可使用之功能配置与参数设定。

7.7.7.1 I/O 功能配置



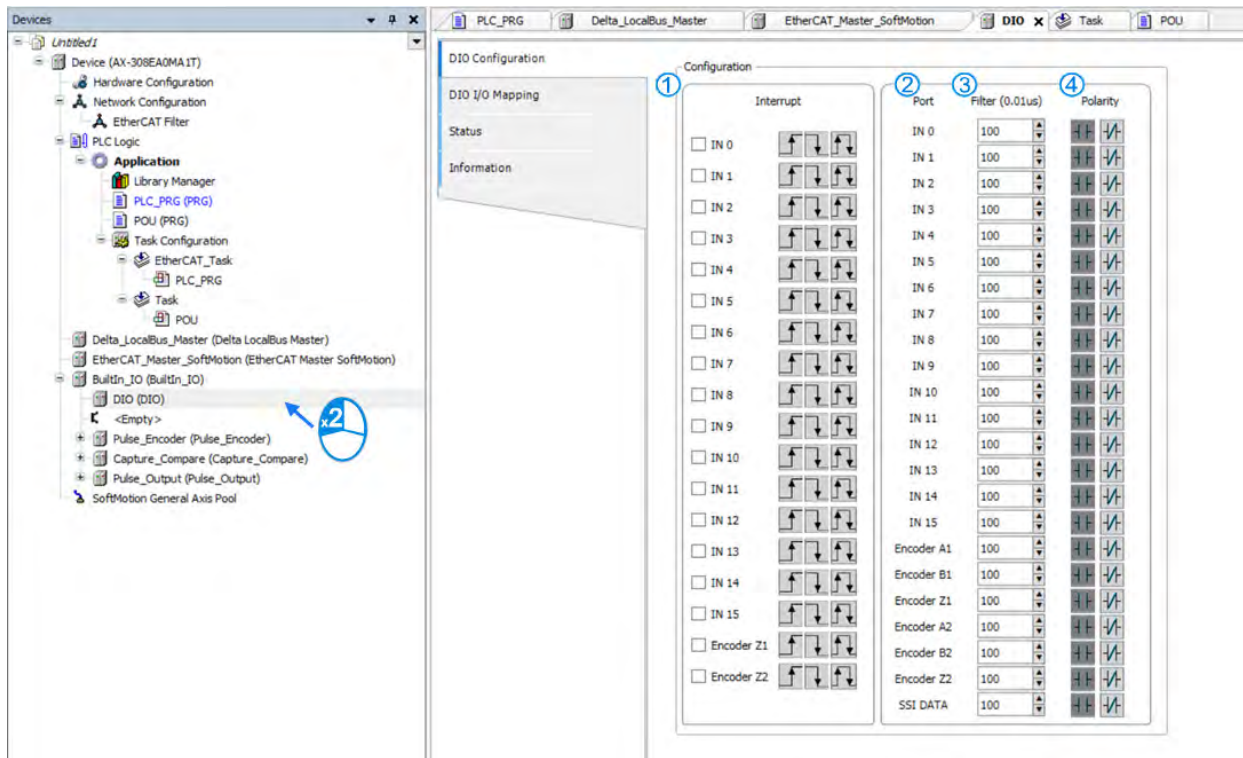
在主机上高速 I/O 提供以下功能设定：

- DIO：可设定中断、滤波、极性等功能，细部说明参考第 7.7.7.2 节。
- SSI Encode：可设定 SSI 的编码类型、时钟频率、SSI 数据长度等功能，细部说明参考第 7.7.7.3 节。
- Pulse Encoder：可设定高速计数器的变量与计数模式以及 Z 相信号是否启用等功能或是高速定时器变量宣告，细部说明参考第 7.7.7.4 节。
- Capture / Compare：可宣告高速捕捉器或高速比较器的变量，细部说明参考第 7.7.7.5 节。
- Pulse Output：可设定脉冲输出的模式、方向以及原点回归的模式与参数等功能，细部说明参考第 7.7.7.6 节。

7.7.7.2 DIO 功能设定

此章节描述 DIO 装置，设定主机 I/O 点位的中断、滤波、极性等功能。

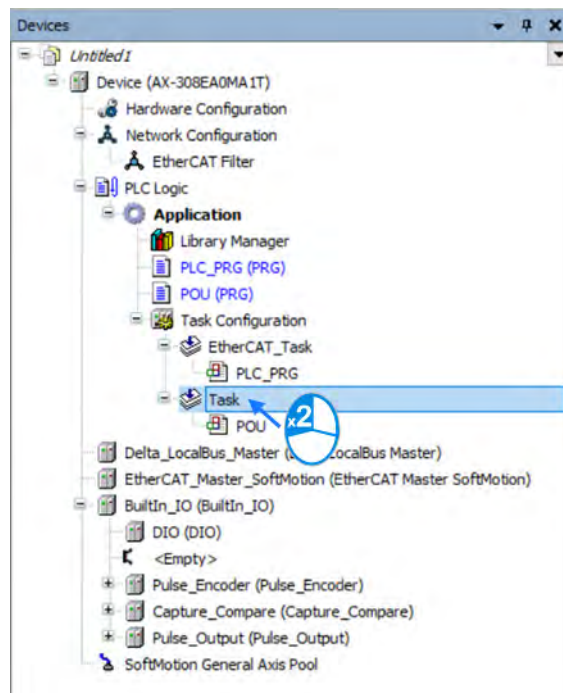
单击 DIO 两下可进入设定画面。



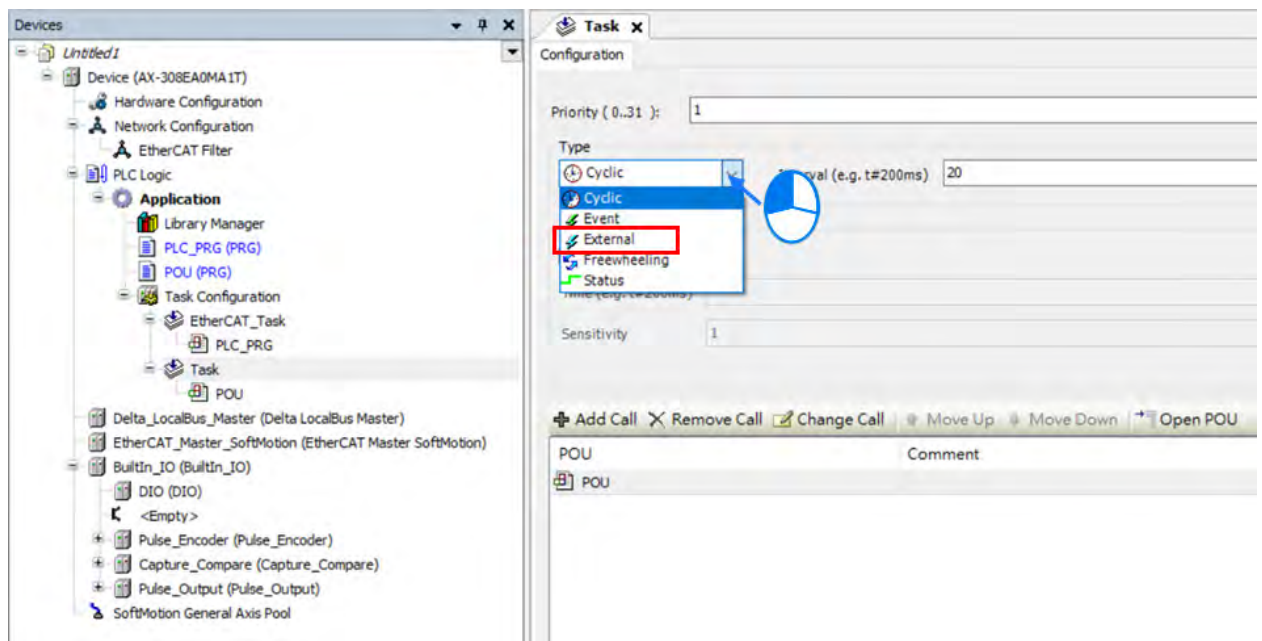
● Configuration

功能	说明
① 外部中断设定	<input type="checkbox"/> 默认值
	<input checked="" type="checkbox"/> 启动外部中断
	当启动外部中断，设定输入接点为上微分信号
	当启动外部中断，设定输入接点为下微分信号
	当启动外部中断，设定输入接点为上下微分信号
② Port	对应外部接点编号
③ Filter	<input type="text" value="100"/> 设定滤波时间 (us)，设定范围为 (0~10000000)，默认是 100us
④ Polarity	设定输入接点极性，默认值为 A 接点
	设定输入接点极性为 B 接点

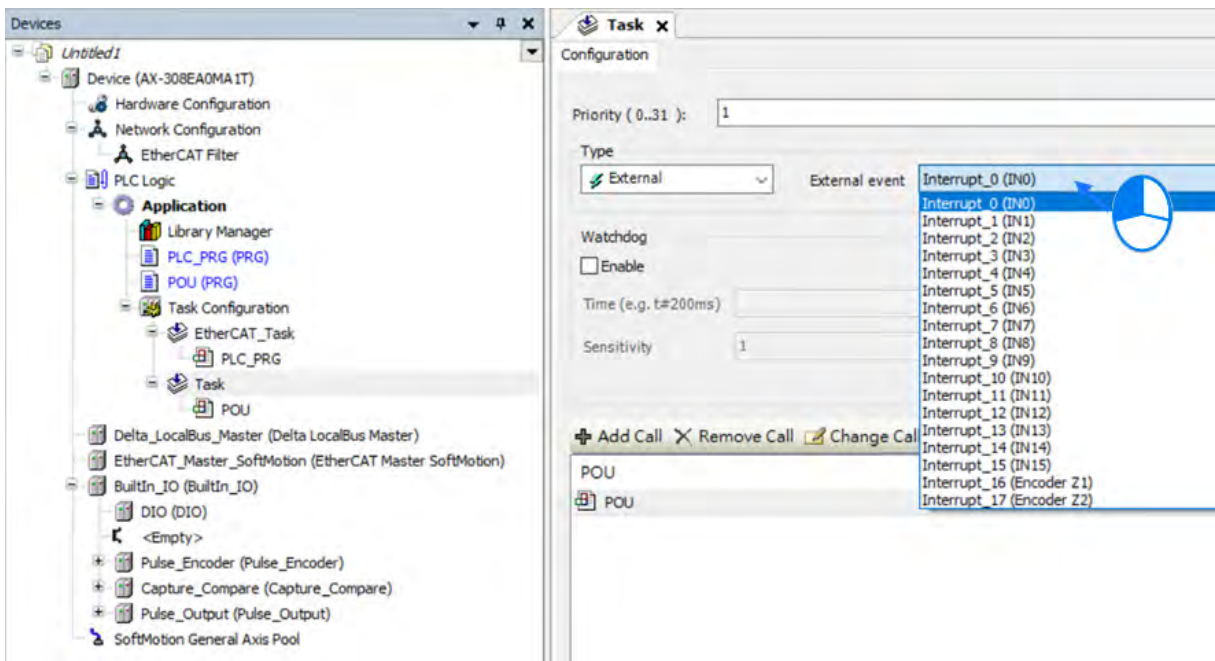
- I/O 中断使用方法：
 - 在 DIO 设定画面启动中断功能后，单击需中断功能的 Task。



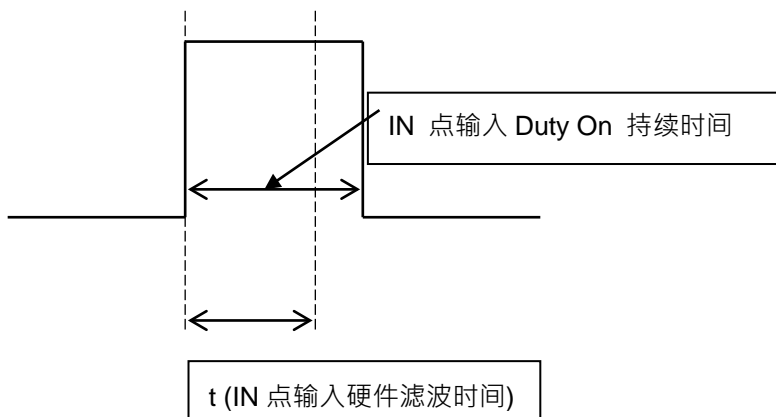
- 进入 Task 设定画面，单击 Type 后选择 External。



- 当选择 External 后，单击 External event 在选择对应中断接点。



- 硬件滤波时间设定值为小于 IN 点输入 Duty On 的时间，如下图所示：
 - 硬件滤波输入范围为 1~50000000，单位为 0.1us。



- 滤波频率与时间关系如下：

滤波频率² (Hz): 滤波频率 = 1 / (2 * t); t 为滤波时间设定值 (单位 0.01 μs)

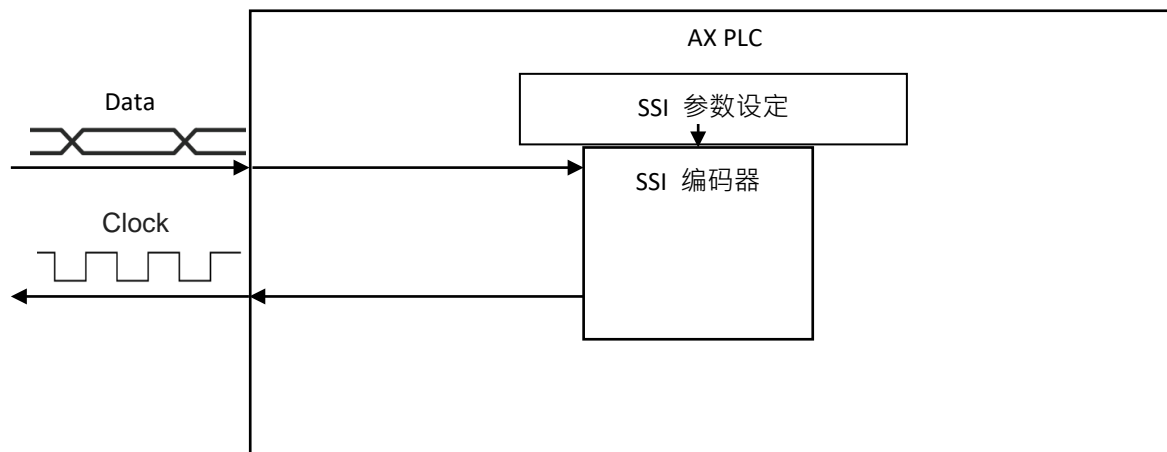
当输入频率高于此滤波频率范围之信号将会被滤除。

此功能针对 DFB_Capture、DFB_Hcnt、DFB_HTmr、DFB_Compare 以及本体 IO 中断等有到的 IN 输入点。

7.7.7.3 SSI 编码器功能设定

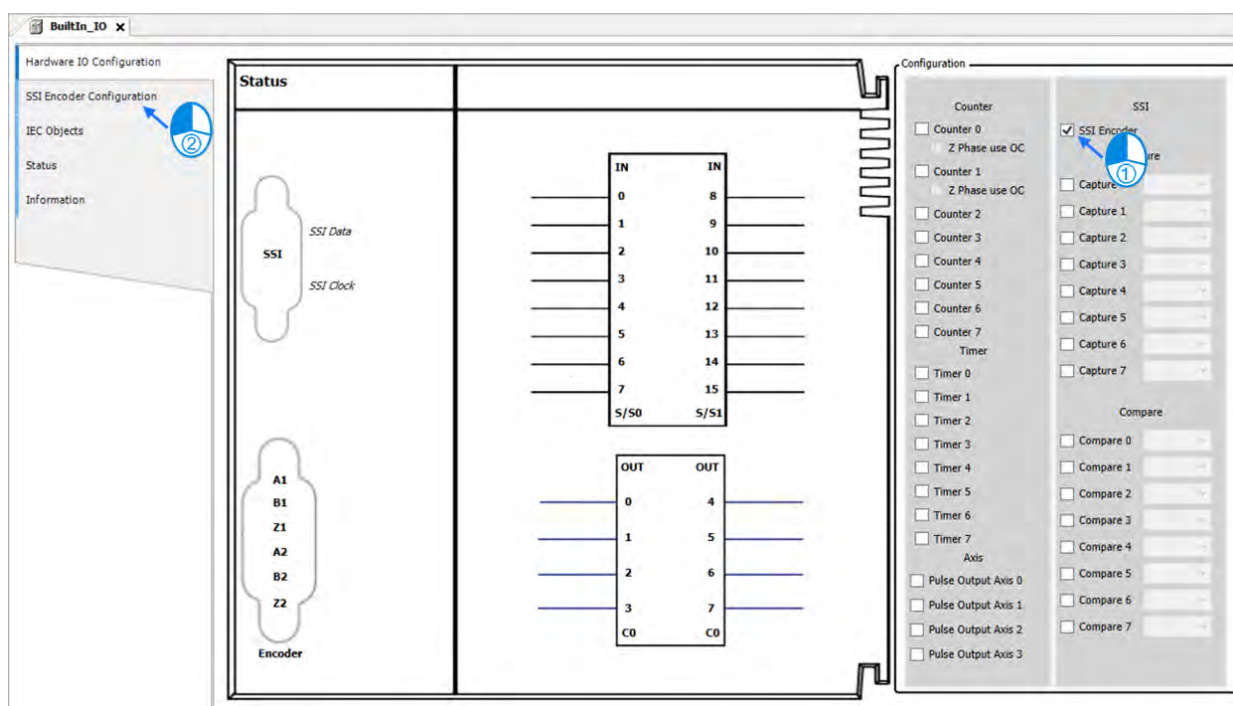
AX 控制器本体 IO 端上支持一组 SSI 编码器功能，通过 D-SUB 接口与 PLC 连接，此接口提供 5V 电源输出以供编码器电源，用户需勾选 SSI 编码器功能启用并设定好该编码器所需要的参数，通过硬件配置通道即可接收编码器的数据。

- SSI 编码器架构

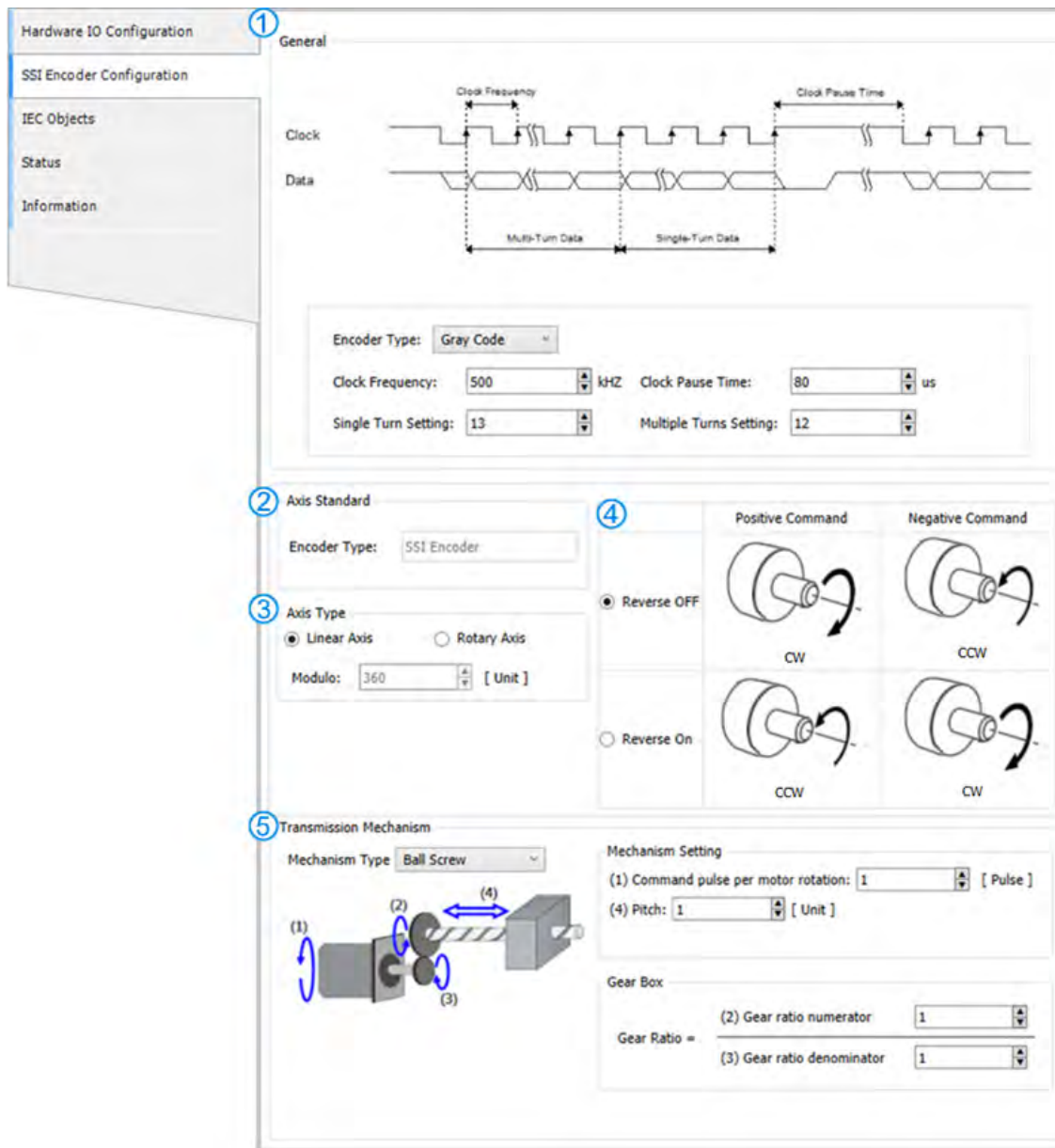


- SSI 编码器功能启动

- 在 BuiltIn_IO 画面中，单击 SSI Encoder 后再单击 SSI Encoder Configuration。



■ 在 SSI Encoder Configuration 画面中可设定 SSI 的相关设定，以下针对设定说明：



■ ① General

名称	功能	设定数值 (默认值)
Encoder Type	SSI 编码器类型	Gray code / Binary code (Gray code)
Clock Frequency	用于 SSI 编码器通讯的传输时钟信号的频率 (需要以 SSI 编码器数据表作为参考)	(500)
Single Turn Setting	单圈数据的数据长度 (需要以 SSI 编码器数据表作为参考)	(13)

名称	功能	设定数值 (默认值)
Multiple Turns Setting	多圈数据的数据长度 (需要以 SSI 编码器数据表作为参考)	(12)
Clock Pause Time	在时钟的最后一个下降缘之后, 数据线将保持低电平一段时间, 直到数据线拉高 (需要以 SSI 编码器数据表作为参考)	(80)

■ ② Axis Standard

名称	功能	设定数值 (默认值)
Encoder Type	编码器类型显示	-

■ ③ Axis Type

名称	功能	设定数值 (默认值)
Linear Axis / Rotary Axis	设定直线轴或旋转轴	Linear Axis Rotary Axis (Linear Axis)
Modulo	当选择旋转轴后可设定旋转轴对大范围值	(360)

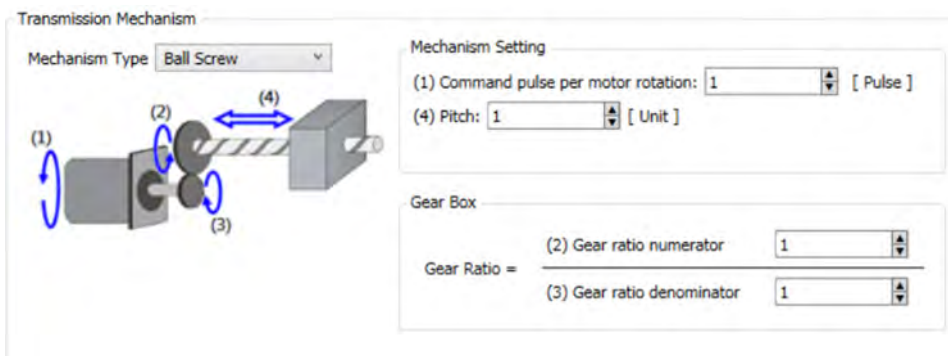
■ ④ Positive / Negative Command

名称	功能
Reverse OFF / ON	设定正反向命令

■ ⑤ Transmission Mechanism

针对不同架构以下列介绍 :

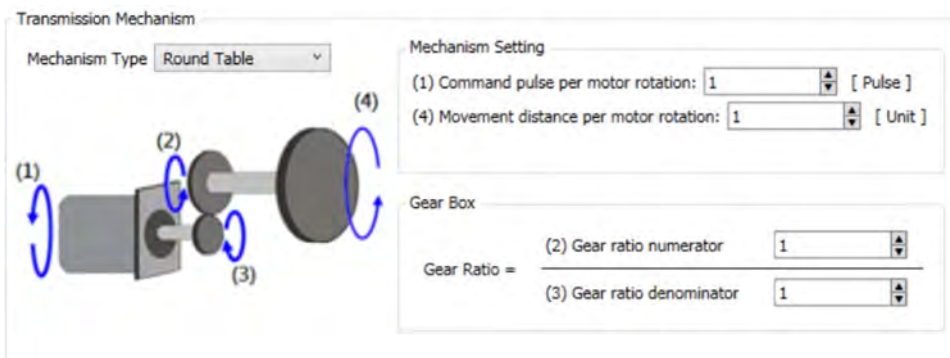
◆ Ball Screw



名称	功能
(1) Command Pulse per motor rotation	电机每转的指令脉冲

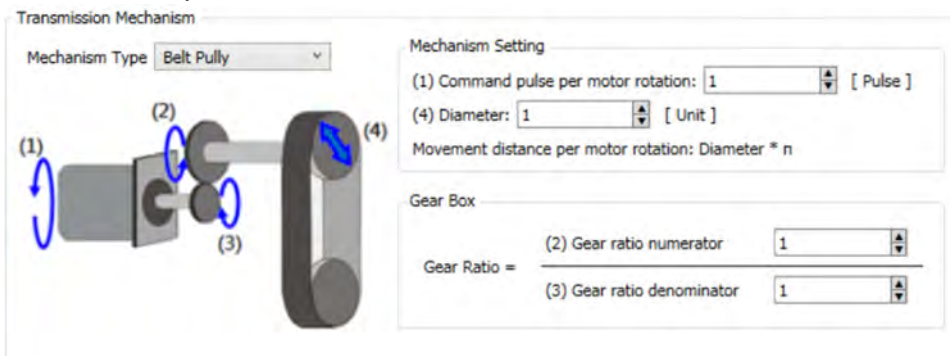
名称	功能
(4) Pitch	螺杆间距
(2) Gear ratio numerator	电子齿轮比分子
(3) Gear ratio denominator	电子齿轮比分母

◆ Round Table



名称	功能
(1) Command Pulse per motor rotation	电机每转的指令脉冲
(4) Movement distance per motor rotation	电机每转的移动距离
(2) Gear ratio numerator	电子齿轮比分子
(3) Gear ratio denominator	电子齿轮比分母

◆ Belt Pully



名称	功能
(1) Command Pulse per motor rotation	电机每转的指令脉冲
(4) Diameter (Movement distance motor rotation : Diameter * n)	直径 (电机每转移动距离 : 直径*n)

名称	功能
(2) Gear ratio numerator	电子齿轮比分子
(3) Gear ratio denominator	电子齿轮比分母

- 设定 SSI Encoder 映像变量
 - SSI Encoder 可通过以下参数读取实际位置与相对应的 ErrorID。

名称	内容
EncoderPosition	SSI 实际位置
ErrorID*注	SSI Encoder 通讯状态，状态如下： 0 : No Error 16#2501 : Error Communication 16#2502 : Wrong Parameter Setting

注：

ErrorID：

当没有连接 SSI Encoder 或是 SSI Encoder 与 CPU 断掉联机 ErrorID = 1。

当 MultiTurns + SingleTurns 大于 32，ErrorID = 2。

以上错误状况会使 BusCycle 停止更新 EncoderPosition，并且 EncoderPosition 会维持上一个值，目的为若 Encoder 有当同步运动的主轴时要避免其它从轴发生跳动。

ErrorID 清除：

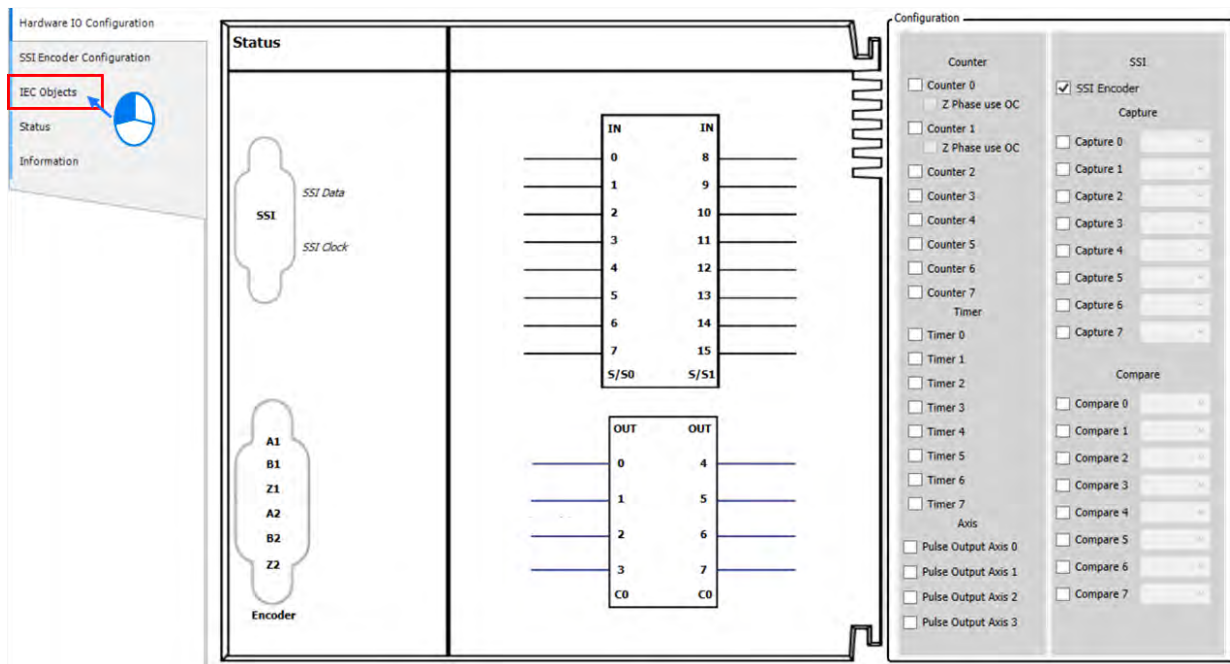
当没有连接 SSI Encoder 或是 SSI Encoder 与 CPU 断掉联机 Status Data = 1，BusCycle 会停止更新，并且 EncoderPosition 会维持上一个值，目的为若 Encoder 有当同步运动的主轴时要避免其它从轴发生跳动：

Ans：检查 SSI Encoder 与 CPU 的联机，修正后韧体会确定联机通道无误后会恢复 BusCycle 的 EncoderPosition 更新，错误原因可能很多种，如：SSI Encoder 未插好、SSI Encoder 坏掉、Drive 板异常等。

当 MultiTurns + SingleTurns 大于 32，Status Data = 2：

Ans：将 MultiTurns + SingleTurns 的参数设定值不超过 32 后重新下载即可

- SSI Encoder 使用在程序中：
 - SSI 编码器装置自带的编码器轴变量，此变量供用户在 POU 中的运动功能块使用 (ex.MC_CamIn)。
 - BuiltIn_IO 画面中单击 IEC Objects



■ 读取范例

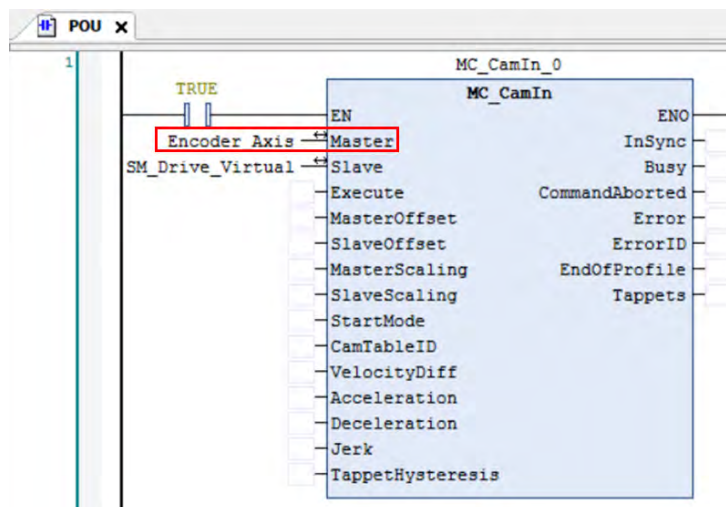
Hardware IO Configuration	Variable	Type	Configuration Function
SSl Encoder Configuration	SSl_Encoder	DFB_SSI_ENCODER_REF	SSl Encoder
	Encoder_Axis_1	DMC_ENCODER_AXIS_REF	SSl Encoder/FreeEncoder_Axis

可于程序中通过红框变量取得 SSI 编码器位置与 Error ID，例：「SSl_Encoder.EncoderPosition」、
「SSl_Encoder.ErrorID」。

■ IEC Objects 画面中①为配置功能介绍，针对 POU 中使用的轴名称需填 Encoder_Axis。

Hardware IO Configuration	Variable	Type	Configuration Function
SSl Encoder Configuration	SSl_Encoder	DFB_SSI_ENCODER_REF	SSl Encoder
	Encoder_Axis	DMC_ENCODER_AXIS_REF	SSl Encoder/FreeEncoder_Axis

- 在 POU 中针对 MC_CamIn 功能块，主轴来源可使用 SSI 当来源，输入的轴名称为 Encoder_Axis。

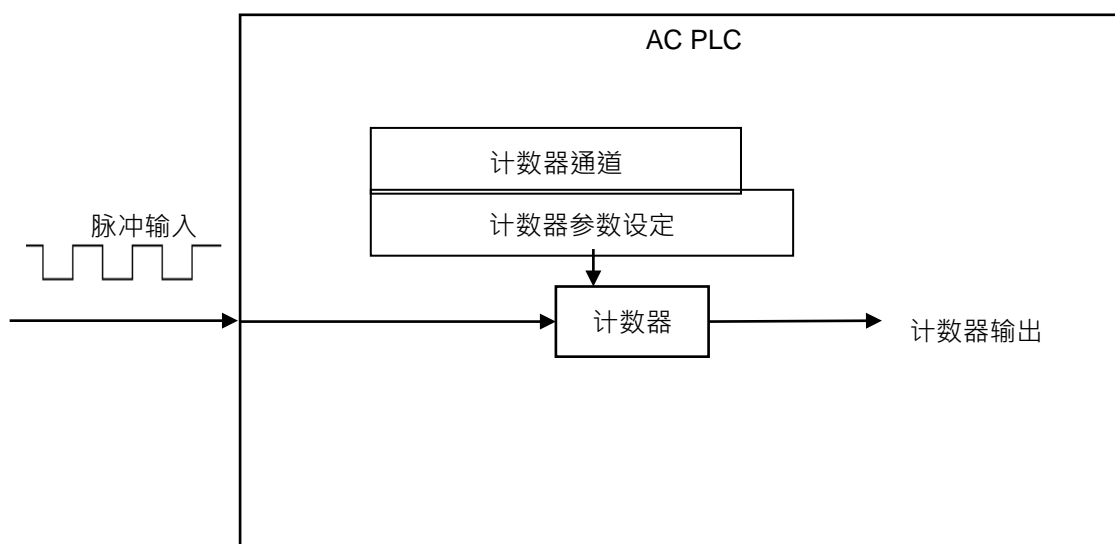


7.7.7.4 Pulse Encoder 功能设定

AX 系列与脉冲型编码器连接方式支持差动输入(2 组)以及开集极输入(6 组)接口。差动接口通过 D-SUB15 接口与 PLC 连接，此接口有 2 组高速计数(计时)器可以计数(计时)编码器的脉冲数值或频率；开集极输入的外部编码器需要通过 IO 板上的 Input 点位连接，此区域有 6 组高速计数(计时)器可以计数(计时)编码器的脉冲数值或频率，用户需勾选脉冲型编码器功能启用并设定好该编码器所需要的参数，通过硬件配置通道即可接收编码器的数据。

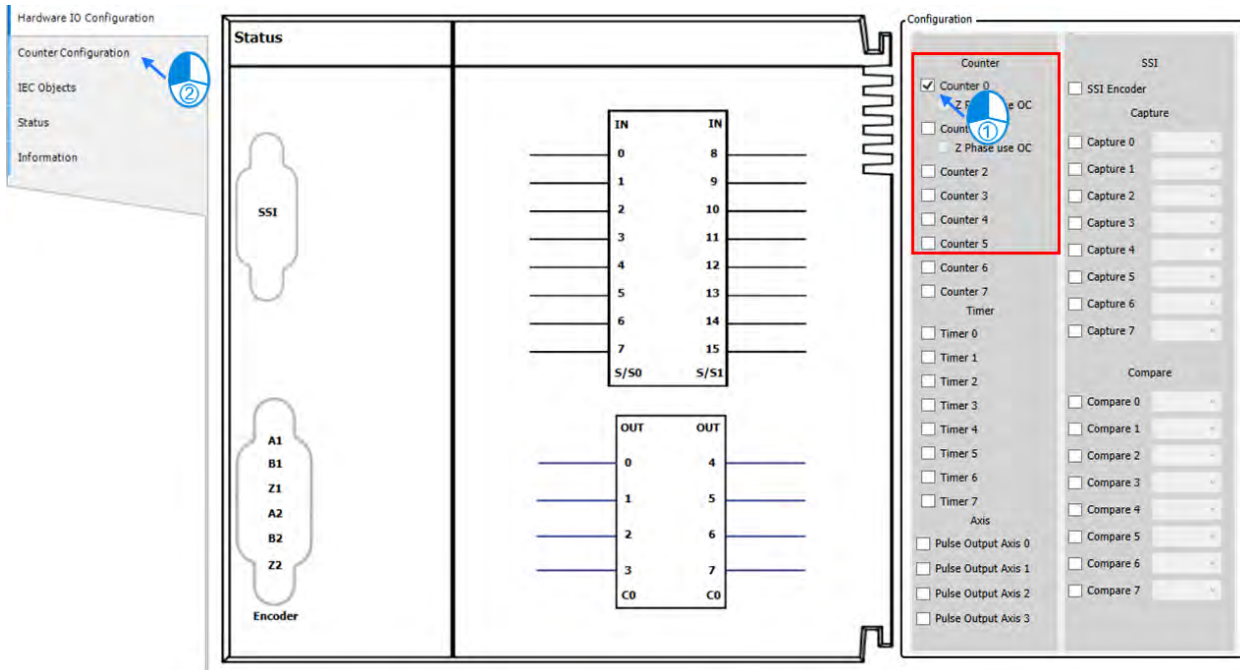
本小节描述本体 IO 的脉冲型编码器功能模块(如下图)，在 AX-308E 支持高速计数器与高速定时器总和最多 8 组。

- 高速计数器 (Cnt)

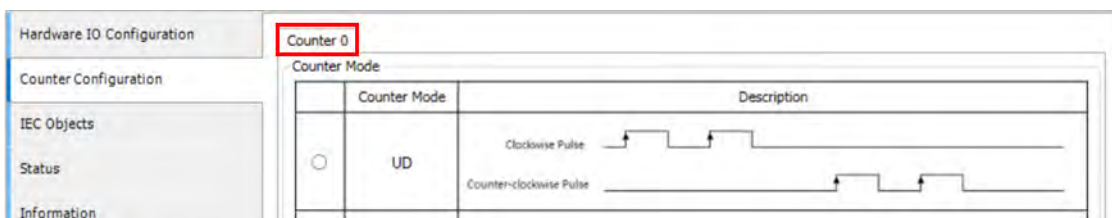


● 高速计数功能启动

- 在 BuiltIn_IO 画面中，共有 8 组 Counter，单击 Counter 0 后再单击 Counter Configuration。



- 在 Counter Configuration 画面中选择我们勾选的 Counter 0 的分页



- 在 Counter Configuration 画面中可设定 Counter 的相关设定，以下针对设定说明：

Hardware IO Configuration

Counter Configuration

IEC Objects

Status

Information

Counter 0

① Counter Mode

	Counter Mode	Description
<input type="radio"/>	UD	<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>Clockwise Pulse </p> <p>Counter-clockwise Pulse </p> </div> </div>
<input type="radio"/>	PD	<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>Pulse </p> <p>Direction Clockwise Counter-clockwise</p> </div> </div>
<input checked="" type="radio"/>	AB	<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>A-Phase Pulse </p> <p>B-Phase Pulse </p> </div> </div>
<input type="radio"/>	4AB	<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>A-Phase Pulse </p> <p>B-Phase Pulse </p> </div> </div>

External Trigger

② Axis Standard

Encoder Type:

③ Axis Type

Linear Axis Rotary Axis

Modulo: [Unit]

④

	Positive Command	Negative Command
<input checked="" type="radio"/> Reverse OFF		
<input type="radio"/> Reverse On		

⑤ Transmission Mechanism

Mechanism Type:

Mechanism Setting

(1) Command pulse per motor rotation: [Pulse]

(4) Pitch: [Unit]

Gear Box

Gear Ratio = $\frac{(2) \text{ Gear ratio numerator } \input{type="text" value="1}}$

(3) Gear ratio denominator

■ ① Counter Mode

名称	功能
UD	正转脉冲列及逆转脉冲列
PD	脉冲列+方向
AB	AB 相脉冲列

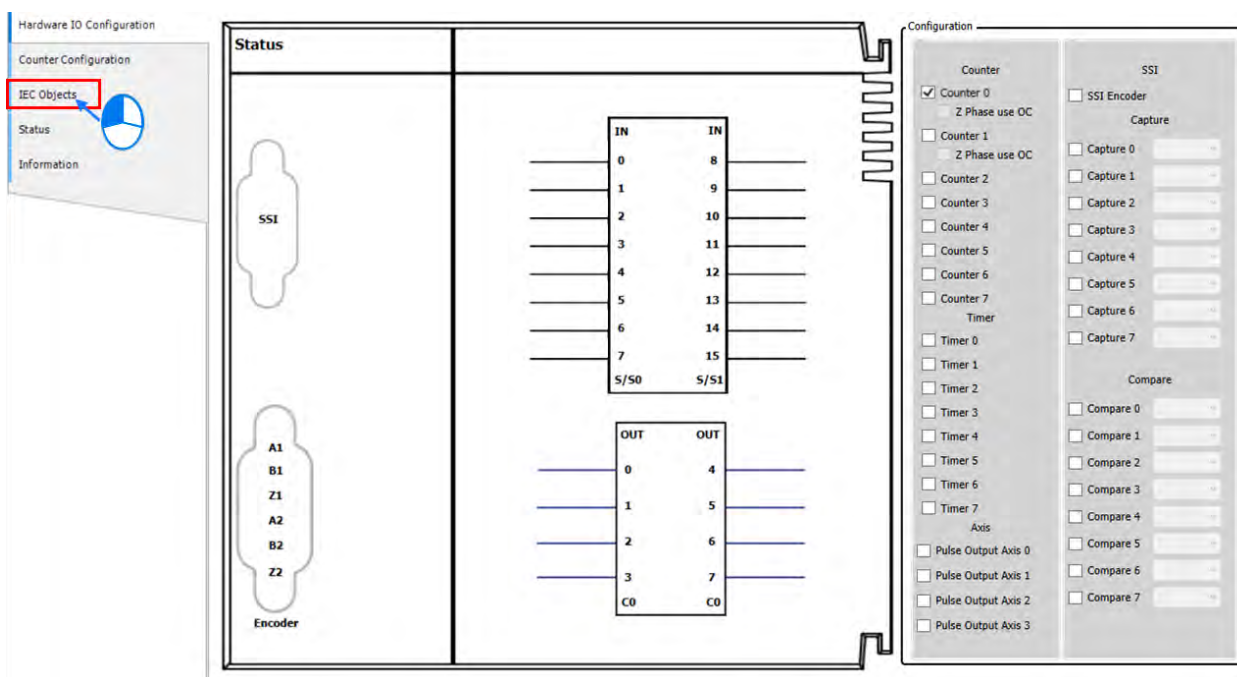
名称	功能
4AB	AB 相脉冲列 (4x)
External Trigger	Z 相信号启用

■ ② ③ ④ ⑤ 软件设定画面：可参考第 7.7.7.3 节 SSI 编码器功能设定。

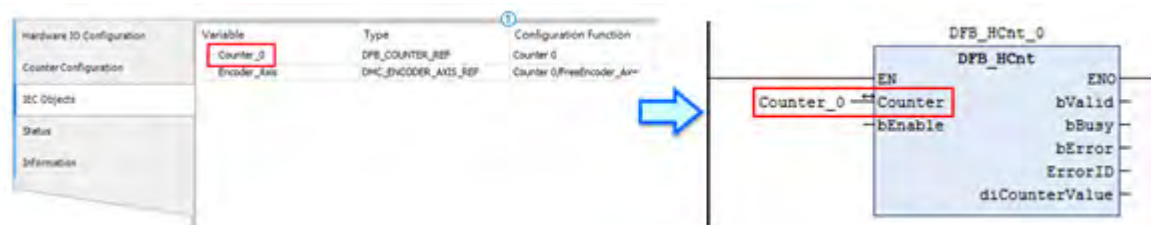
● Counter 使用在程序中：

高速计数器自带的编码器轴变量，此变量供用户在 POU 中的运动功能块使用。

■ BuiltIn_IO 画面中单击 IEC Objects



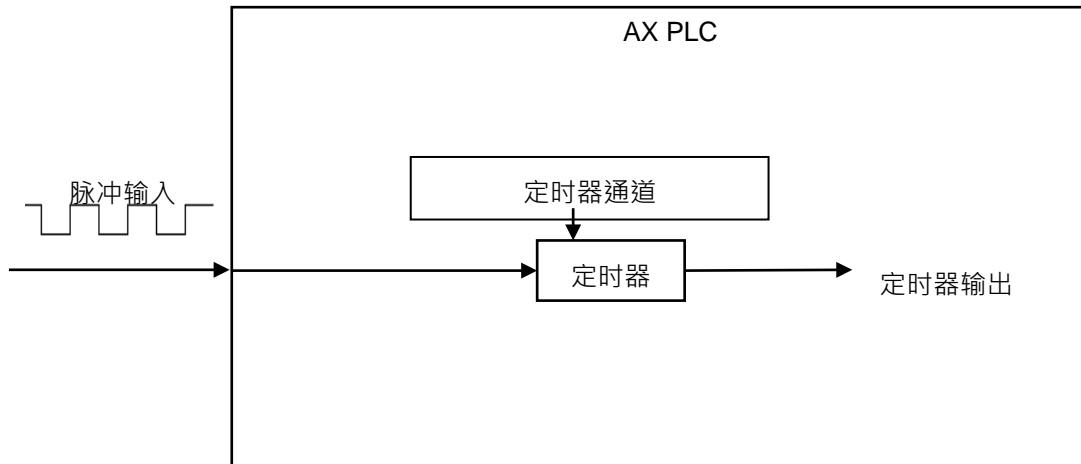
■ IEC Objects 画面中①为配置功能介绍，若要启动 Counter 计数功能，需将所建立的变量 Counter_0 输入至功能块 DFB_HCcnt 的 Counter 引脚。



- 在 POU 使用 MC_GerIn 功能块，主轴来源如使用 Counter_0，其对应的输入变量为 Encoder_Axis。

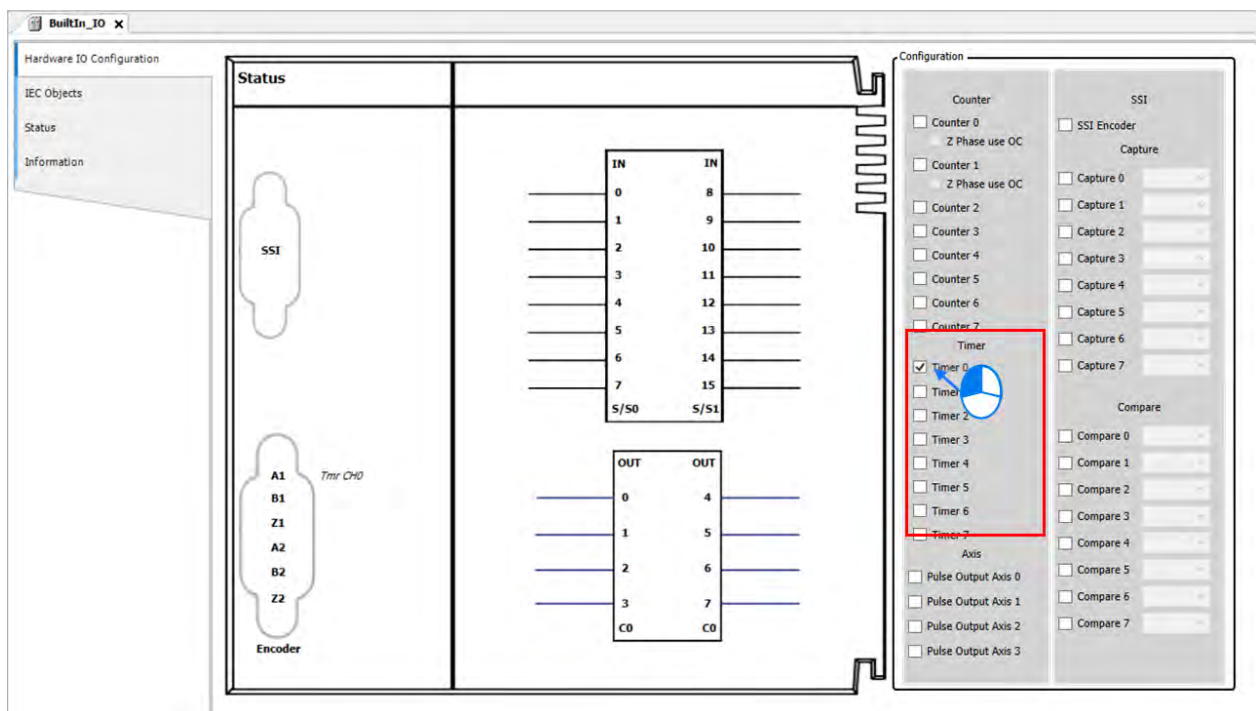


● 高速计时 (Tmr)

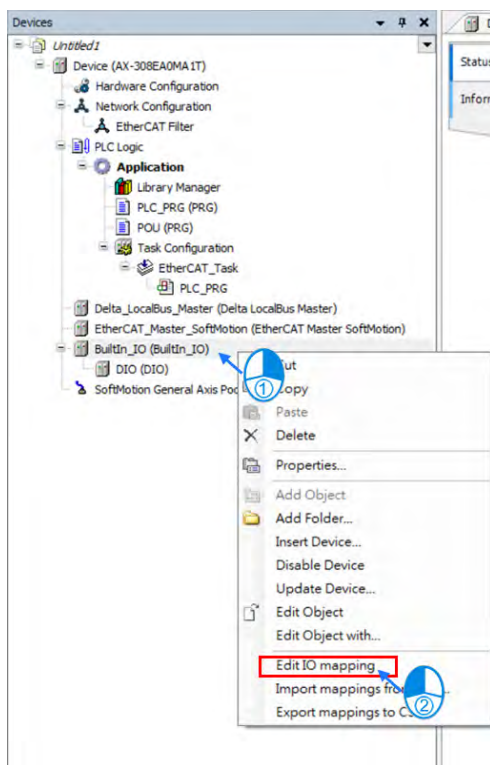


● 高速计时功能启动

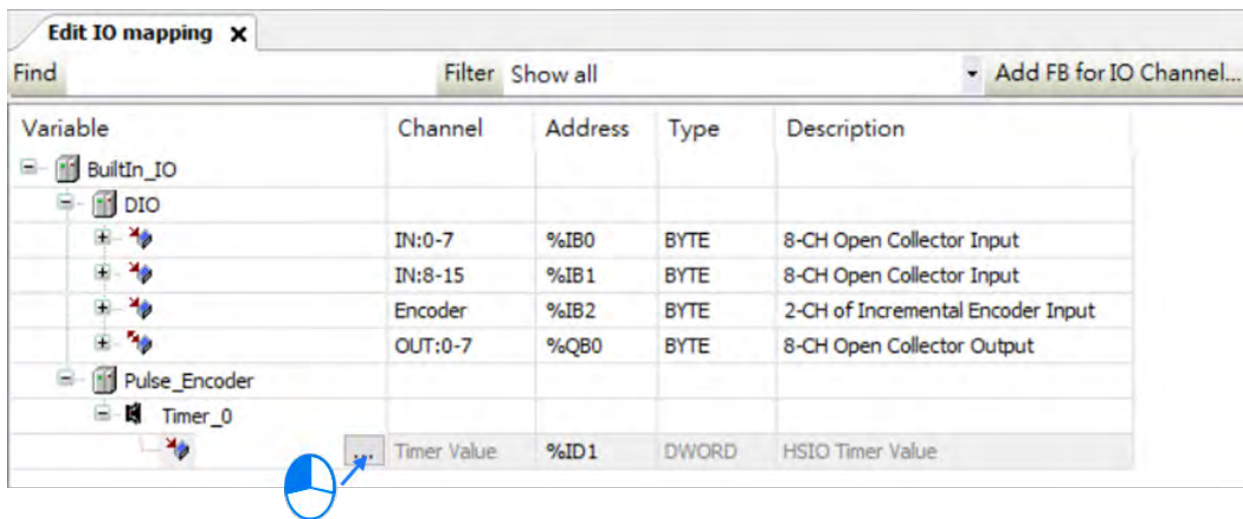
- 在 BuiltIn_IO 画面中，Timer 共有 8 组，单击 Timer 0。由于 Timer 不需设定画面所以勾选完毕就可以启动。高速定时器在 AX 系列内部计时时钟为 0.1μs。



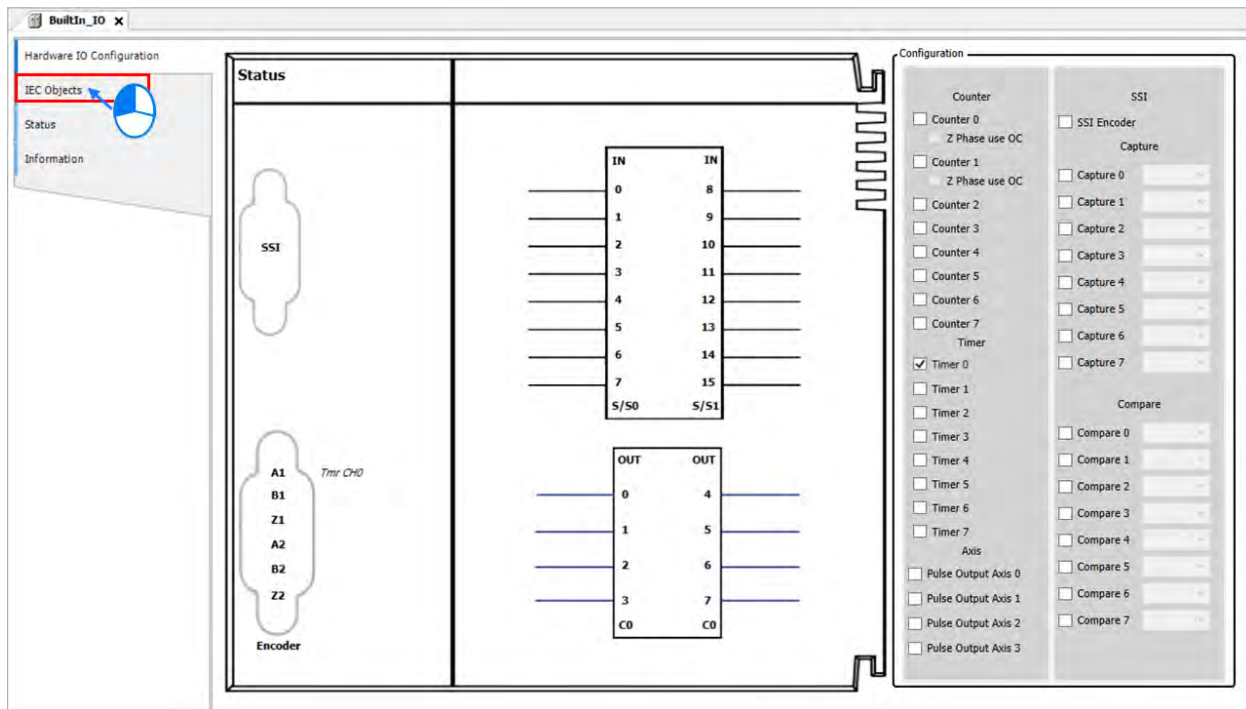
- 设定 Timer 映像变量
 - 单击 BuiltIn_IO 右键，选择 Edit IO Mapping。



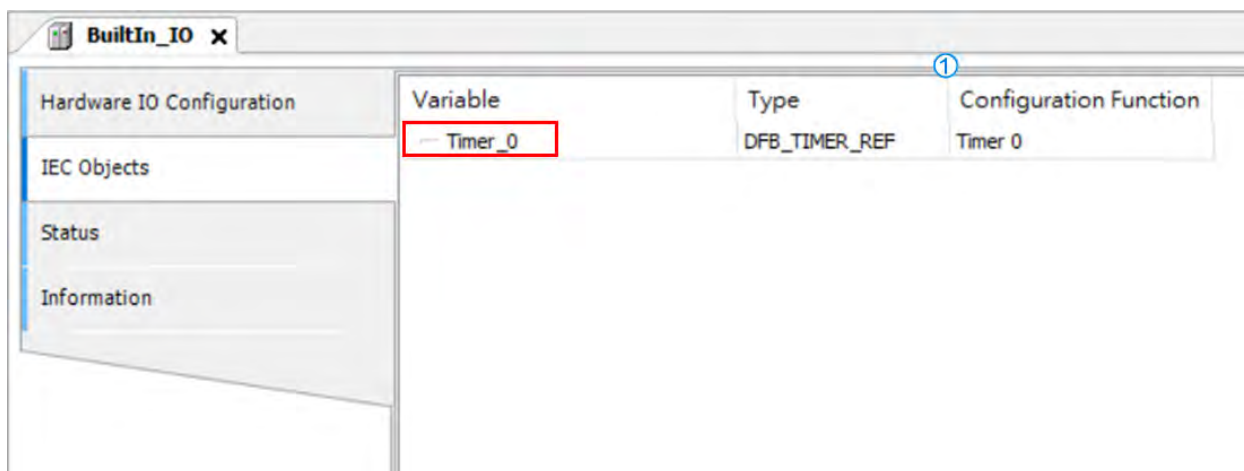
- Edit IO Mapping 画面中，单击 [...] 后新增变数。



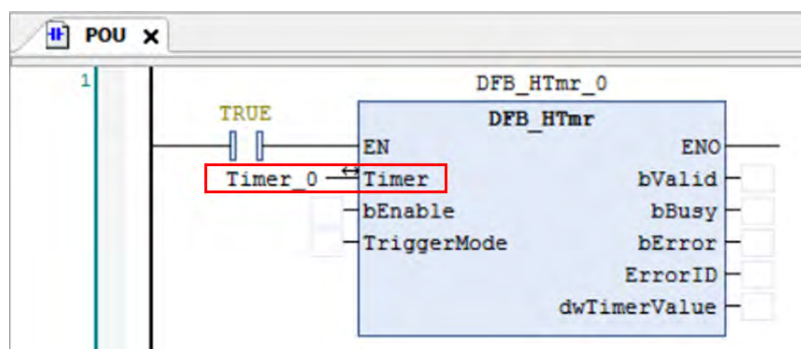
- Timer 使用在程序中：
 - 此变量供用户在 POU 中的运动功能块使用。
 - BuiltIn_IO 画面中单击 IEC Objects



- IEC Objects 画面中①为配置功能介绍，针对 POU 中使用的轴名称需填 Timer_0。



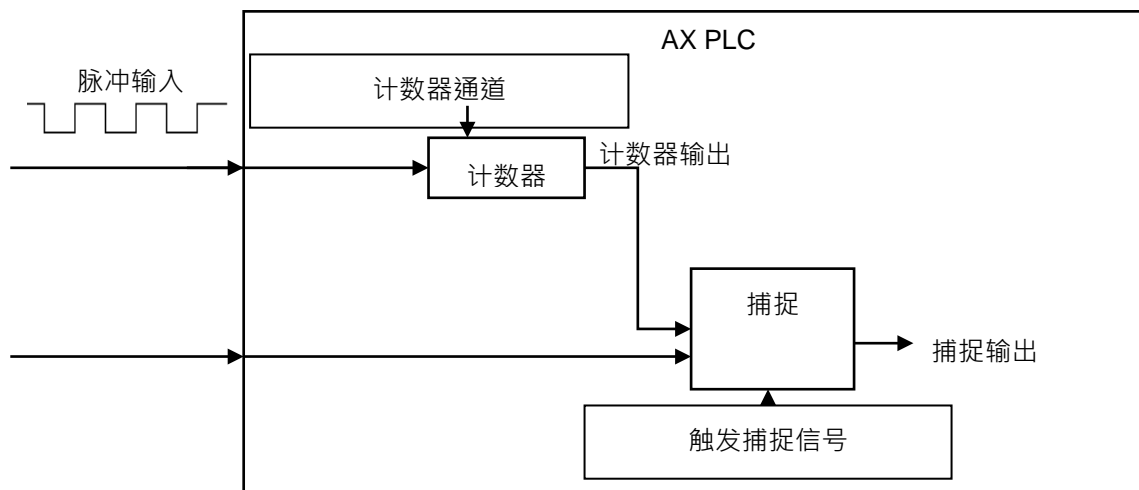
- 在 POU 中针对 DFB_HTmr 功能块，输入的轴名称为 Timer_0。



7.7.7.5 Capture/Compare 功能设定

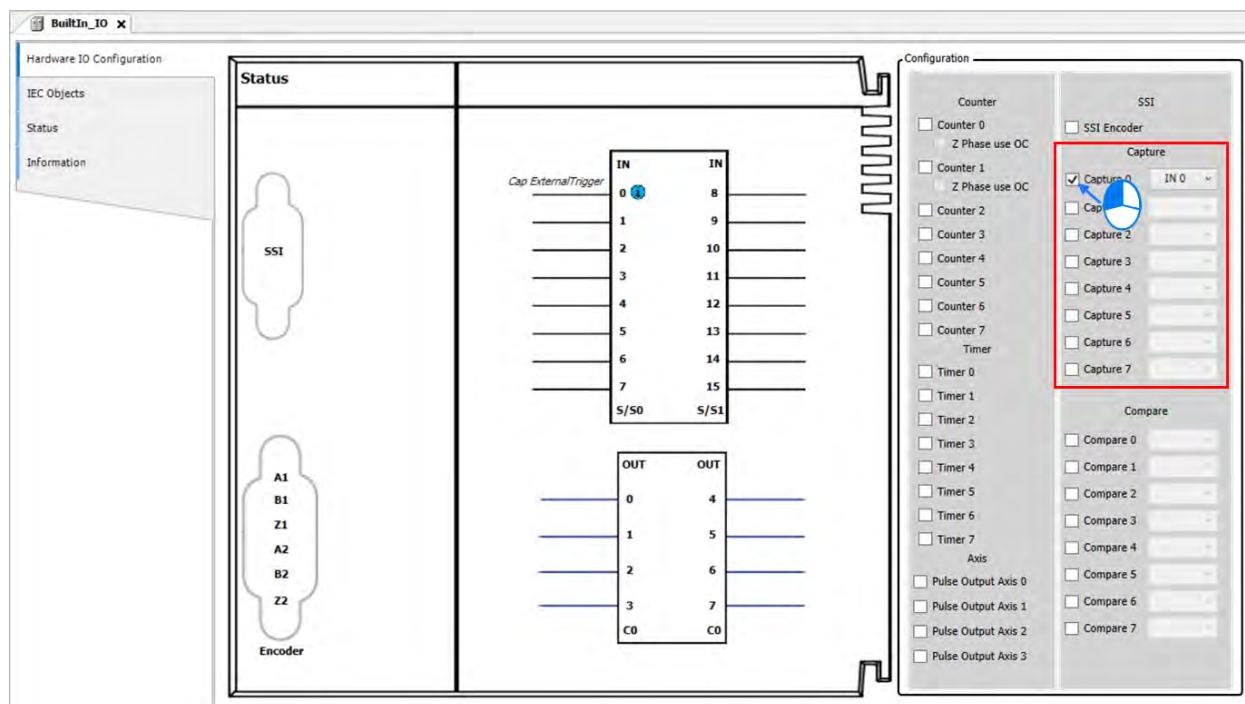
本小节描述本体 IO 的高速计数比较器与捕捉器功能模块，在 AX 系列支持高速捕捉器与高速计数比较器总和最多 8 组。

- 高速计数捕捉器 (Capture)



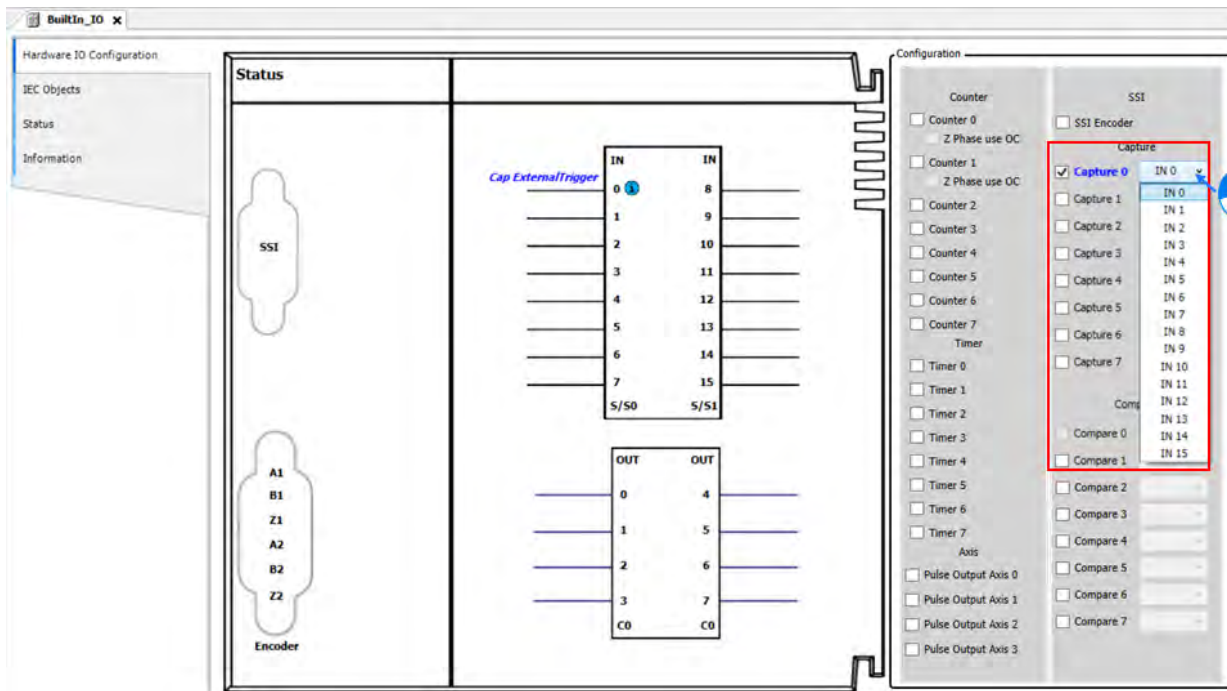
- Capture 功能启动

- 在 BuiltIn_IO 画面中，Capture 共有 8 个点位，单击其中一个启动。

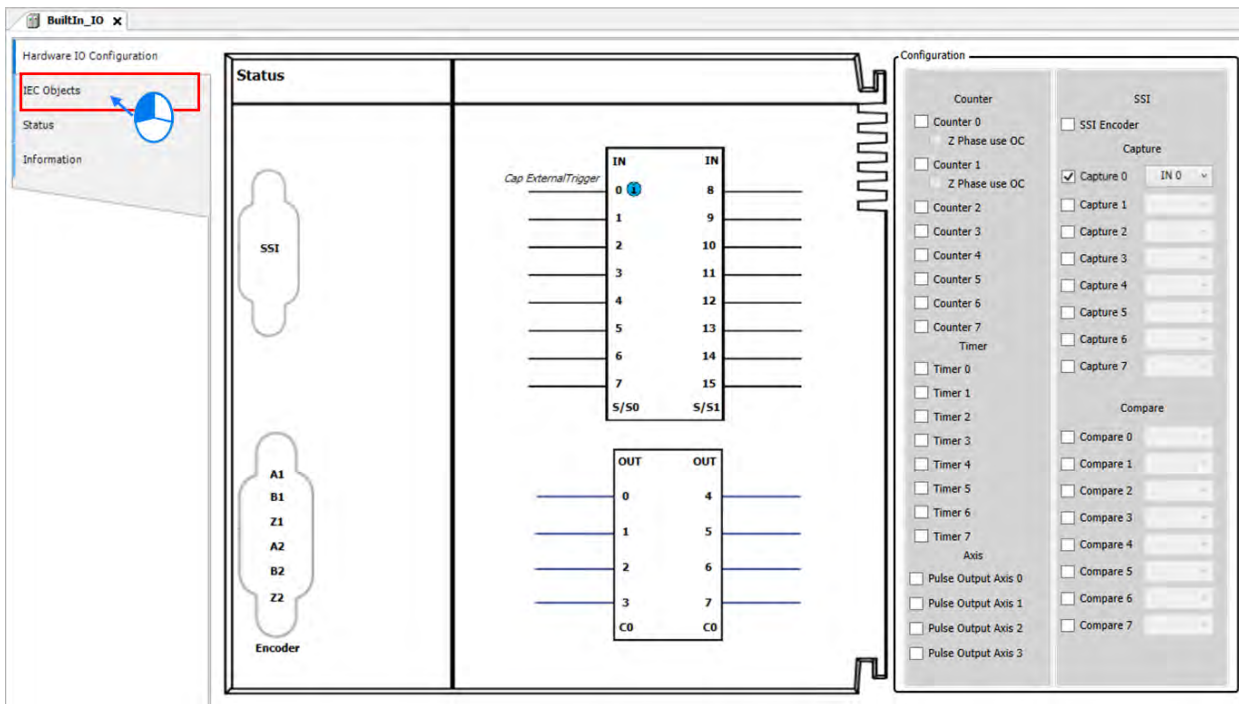


7

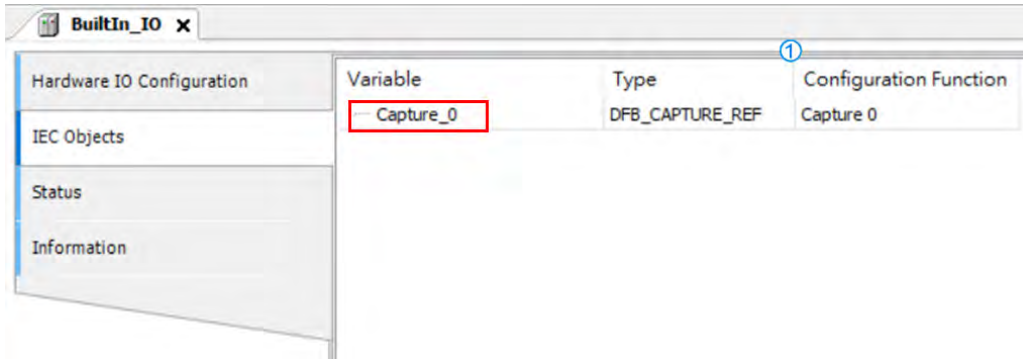
- 启动 Capture 后单击下拉式选单，选择外部触发来源。



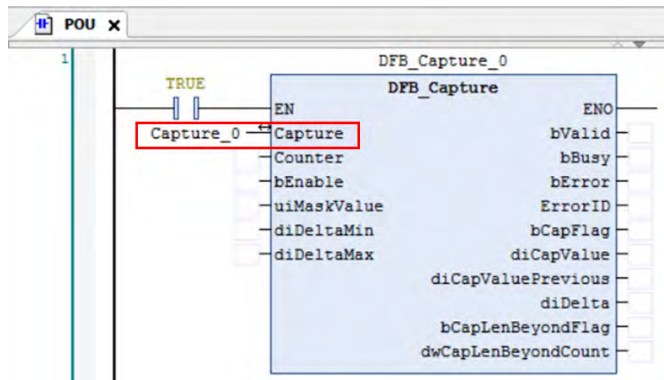
- Capture 使用在程序中：
此变量供用户在 POU 中的运动功能块使用。
- BuiltIn_IO 画面中单击 IEC Objects



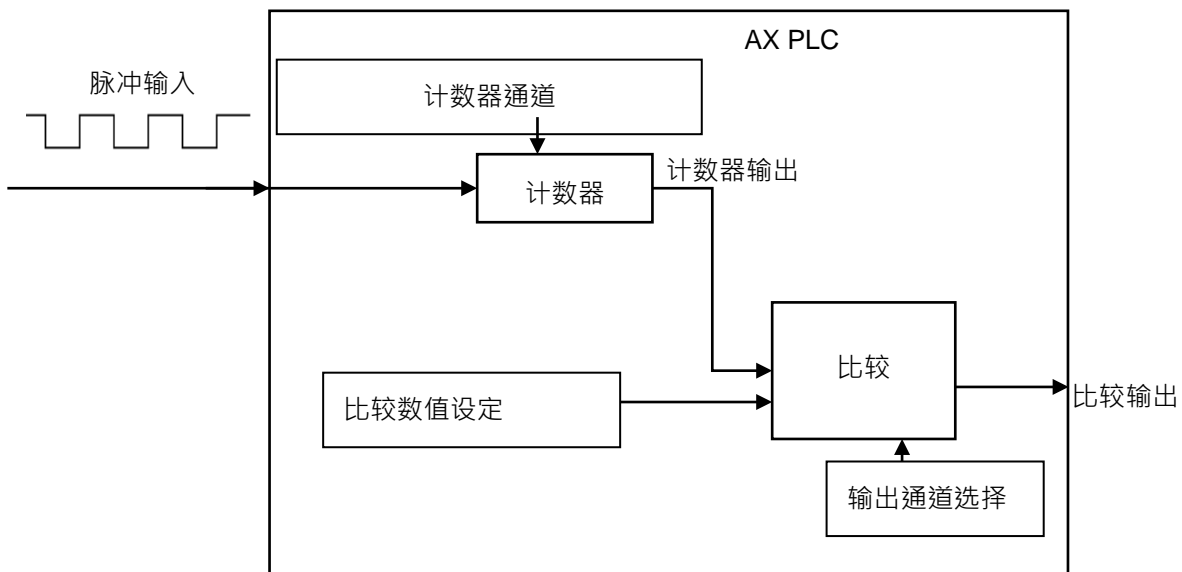
- IEC Objects 画面中①为配置功能介绍，针对 POU 中使用的轴名称需填 Capture_0。



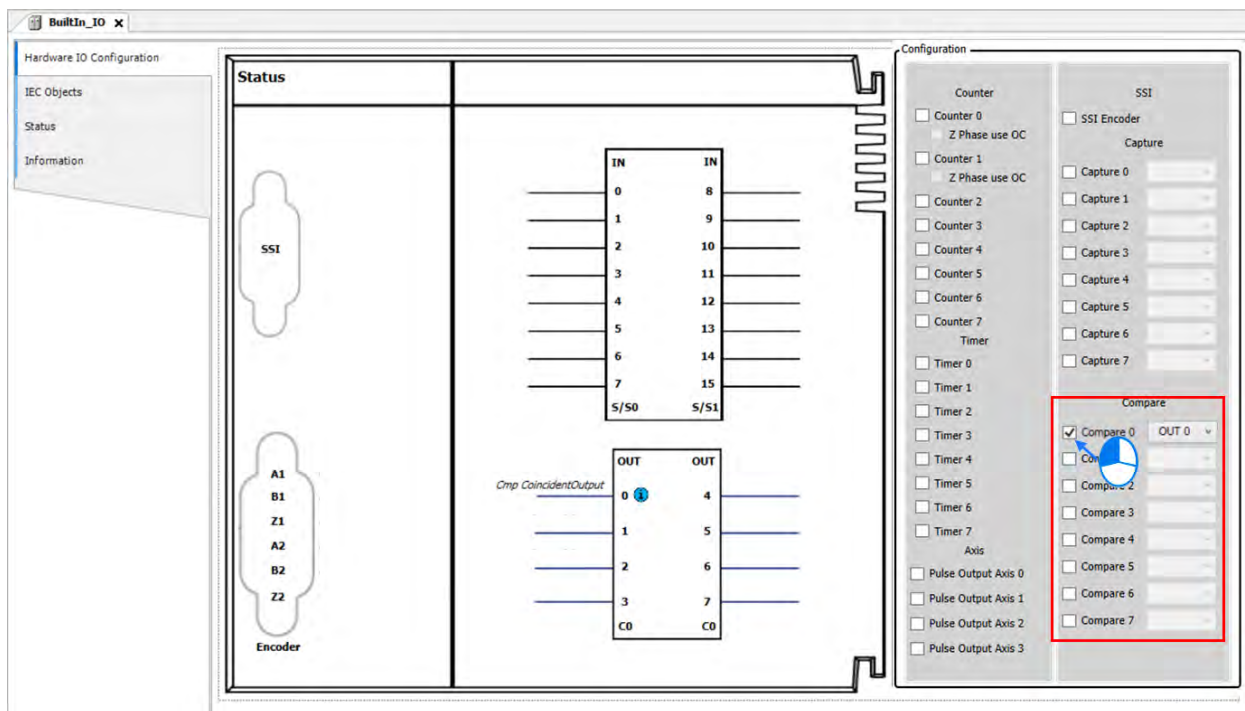
- 在 POU 中针对 DFB_Capture 功能块，输入的轴名称为 Capture_0。



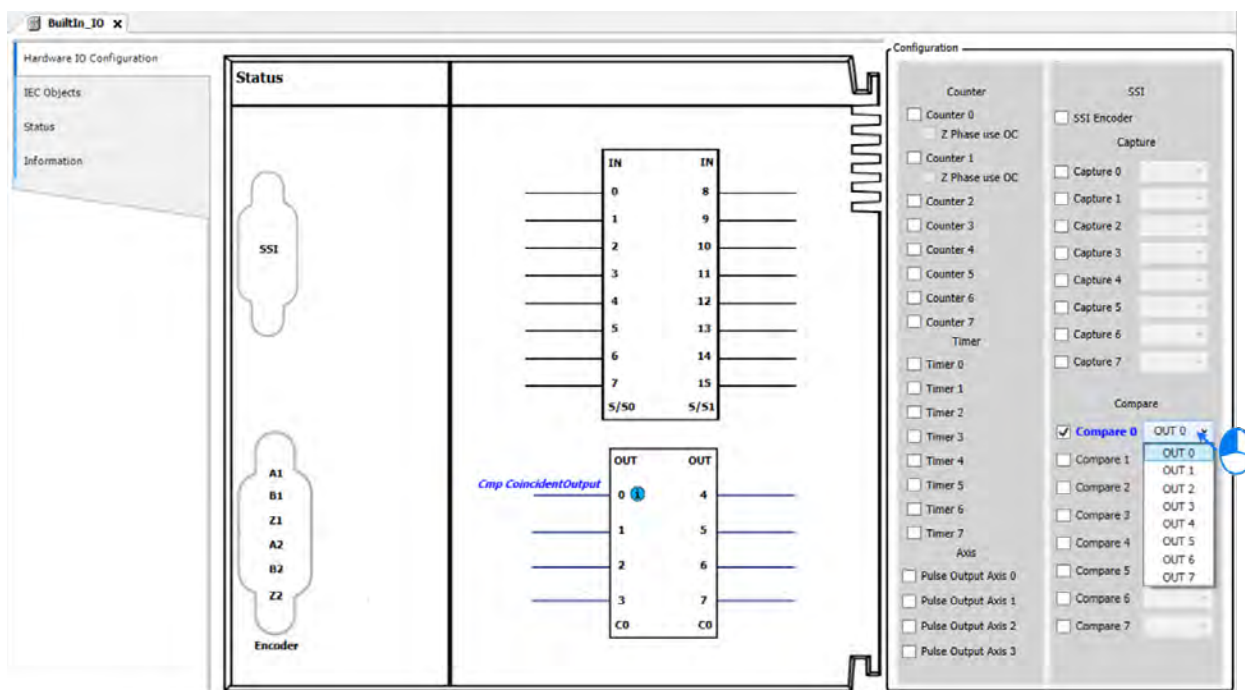
- 高速计数比较器 (Compare)



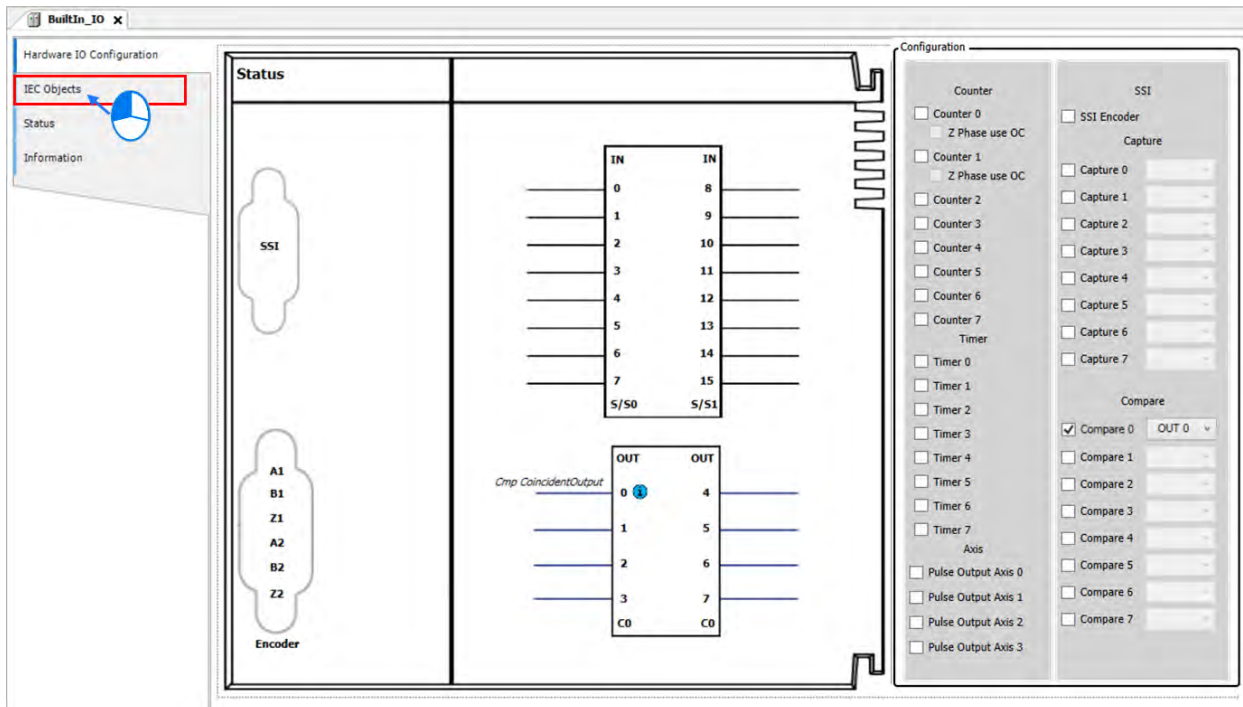
- Compare 功能启动
 - 在 BuiltIn_IO 画面中，Compare 共有 8 个点位，单击其中一个启动。



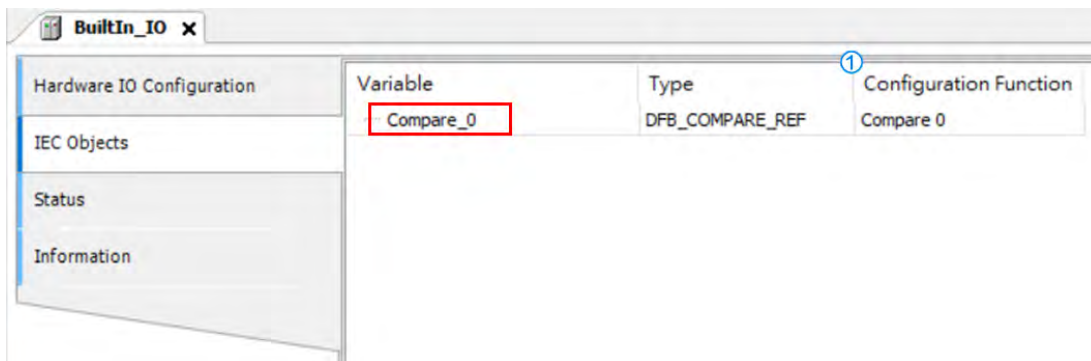
- 启动 Compare 后单击下拉式选单，选择 Compare 比较成功输出外部接口。



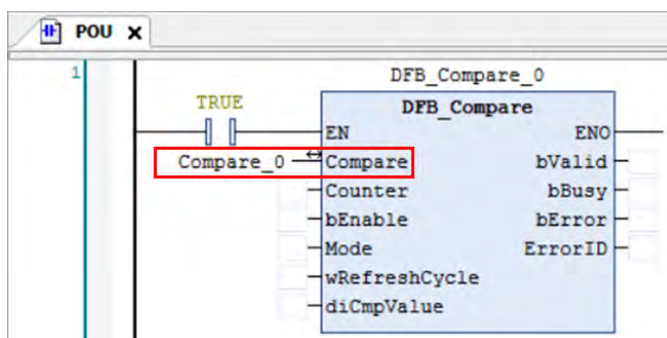
- Compare 使用在程序中：
 - 此变量供用户在 POU 中的运动功能块使用。
 - BuiltIn_IO 画面中单击 IEC Objects



- IEC Objects 画面中①为配置功能介绍，针对 POU 中使用的轴名称需填 Compare_0。



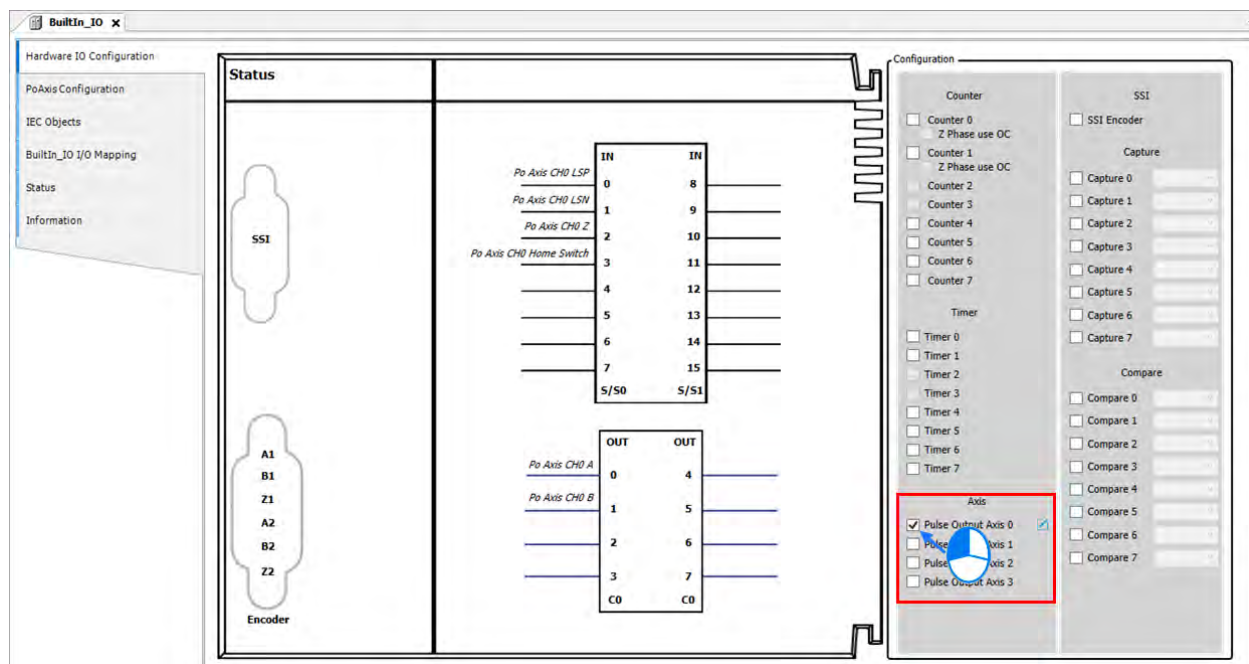
- 在 POU 中针对 DFB_Compare 功能块，输入的轴名称为 Compare_0。



7.7.7.6 Pulse Output 功能设定

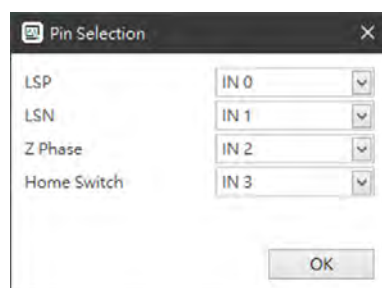
本小节描述本体 IO 脉冲输出功能模块 (如下图), 在 AX-308E 支持最多 4 组脉冲输出装置供选择使用。

- Axis 功能启动
 - 在 BuiltIn_IO 画面中, Axis 共有 4 轴脉冲输出, 单击其中一轴启动。

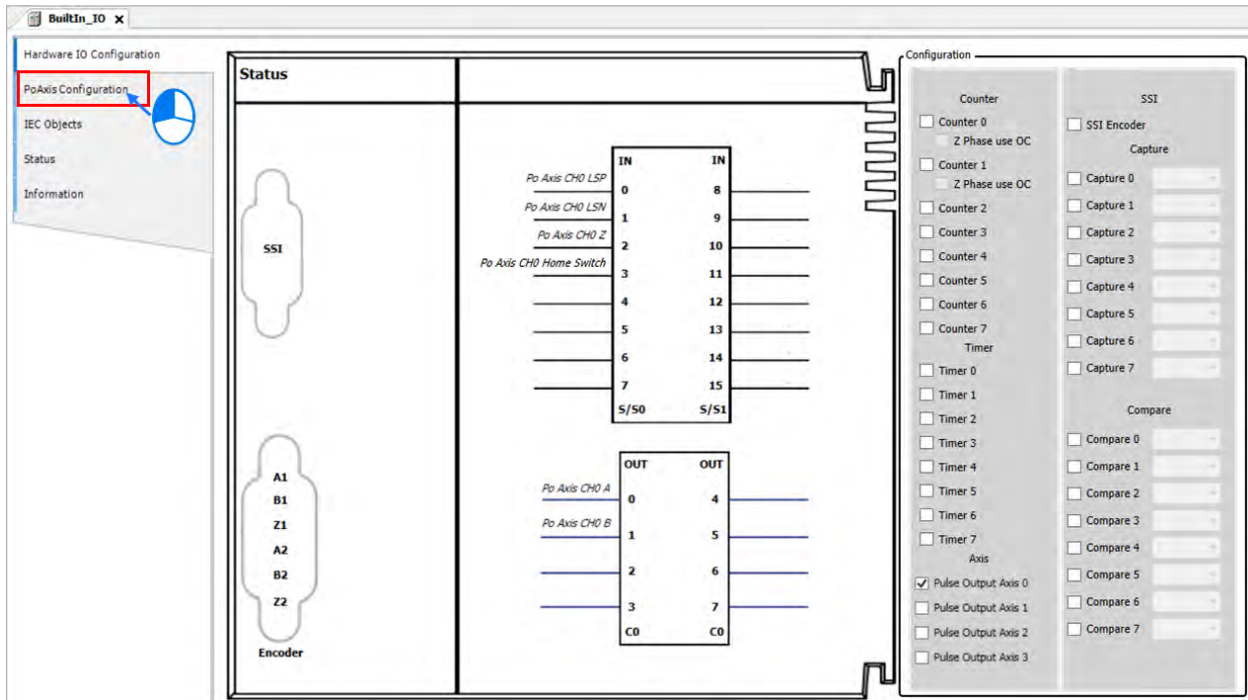


注: AX-3 韧体版本 V1.0.2.0 以上 (含), 支持脉冲轴输入信号设定。

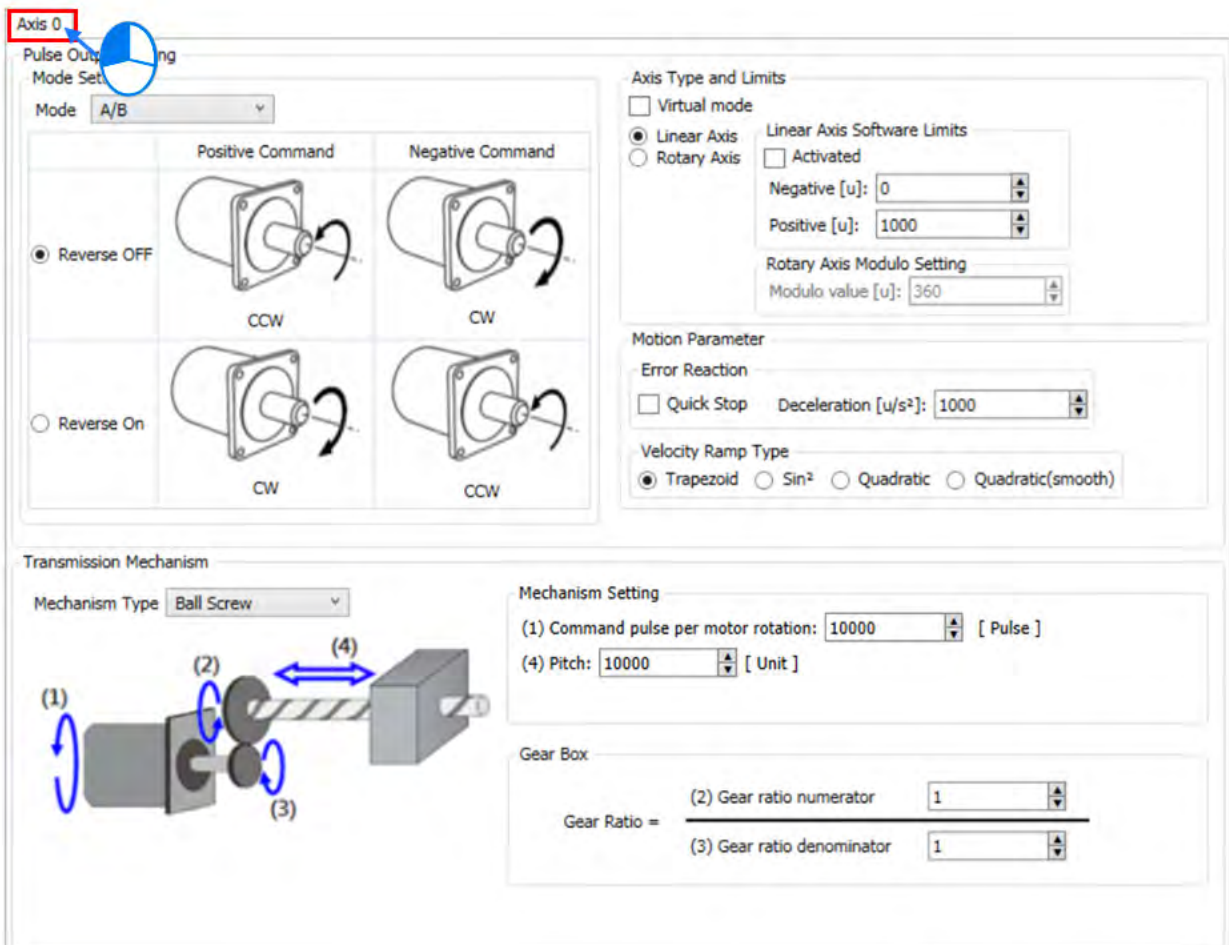
单击 , 可选择设定正极限、反极限、Z 相信号、原点开关在哪个输入点或者关闭信号。



- 启动 Axis 后再点 PoAxis Configuration



■ 在 PoAxis Configuration 画面后单击 Axis 0 分页



■ 在 Pulse Output Setting 画面中可设定 Axis 的相关设定，以下针对设定说明：

① Pulse Output Setting
 Mode Setting
 Mode: A/B

② Axis Type and Limits
 Virtual mode
 Linear Axis **③** Linear Axis Software Limits
 Activated
 Negative [u]: 0
 Positive [u]: 1000
 Rotary Axis **④** Rotary Axis Modulo Setting
 Modulo value [u]: 360

⑤ Motion Parameter
⑤ Error Reaction
 Quick Stop Deceleration [u/s²]: 1000
⑥ Velocity Ramp Type
 Trapezoid Sin² Quadratic Quadratic(smooth)

⑦ Transmission Mechanism
 Mechanism Type: Ball Screw

 Mechanism Setting
 (1) Command pulse per motor rotation: 10000 [Pulse]
 (4) Pitch: 10000 [Unit]
 Gear Box
 Gear Ratio = $\frac{(2) \text{ Gear ratio numerator } 1}{(3) \text{ Gear ratio denominator } 1}$

⑧ Homing Setting
 Homing Mode: Mode 35
 Homing speed during search for switch: 100 [Unit/s]
 Homing speed during search for z phase pulse: 50 [Unit/s]
 Homing Acceleration: 1000 [Unit/s²]
 Description
Mode 35 : Depending on the current position
 In mode 35, The homing instruction is executed, the axis does not move and its current position is regarded as the home position.

■ ① Mode Setting

名称	功能	设定数值 (默认值)
Mode	设定输出脉冲类型	CW/CCW Pulse and Direction (A/B)
Reverse On / Reverse OFF	设定脉冲轴正/反转	Reverse ON Reverse OFF (Reverse OFF)

■ ② Axis Type and Limits

名称	功能	设定数值 (默认值)
Virtual	启动虚轴	TRUE FALSE (FALSE)
Linear Axis / Rotary Axis	设定直线轴或旋转轴	Linear Axis Rotary Axis (Linear Axis)

■ ③ Linear Axis Software Limits

名称	功能	设定数值 (默认值)
Activated	启动软件极限 (仅支持直线轴)	TRUE/FALSE (FALSE)
Negative[u]	反转软件极限	(0)
Positive[u]	正转软件极限	(10000)

■ ④ Rotary Axis Modulo Setting

名称	功能	设定数值 (默认值)
Modulo Value[u]	设定一圈范围 (仅支持旋转轴)	(360)

■ ⑤ Error Reaction

名称	功能	设定数值 (默认值)
Quick Stop	轴紧急停止	(360)
Deceleration[u/s ²]	轴减速停止 (未启动 Quick Stop 才有效)	(10000)

■ ⑥ Velocity Ramp Type

名称	功能	设定数值 (默认值)
Trapezoid/Sin ² /Quadratic/ Quadratic (Smooth)	设定轴运动曲线	(Trapezoid)

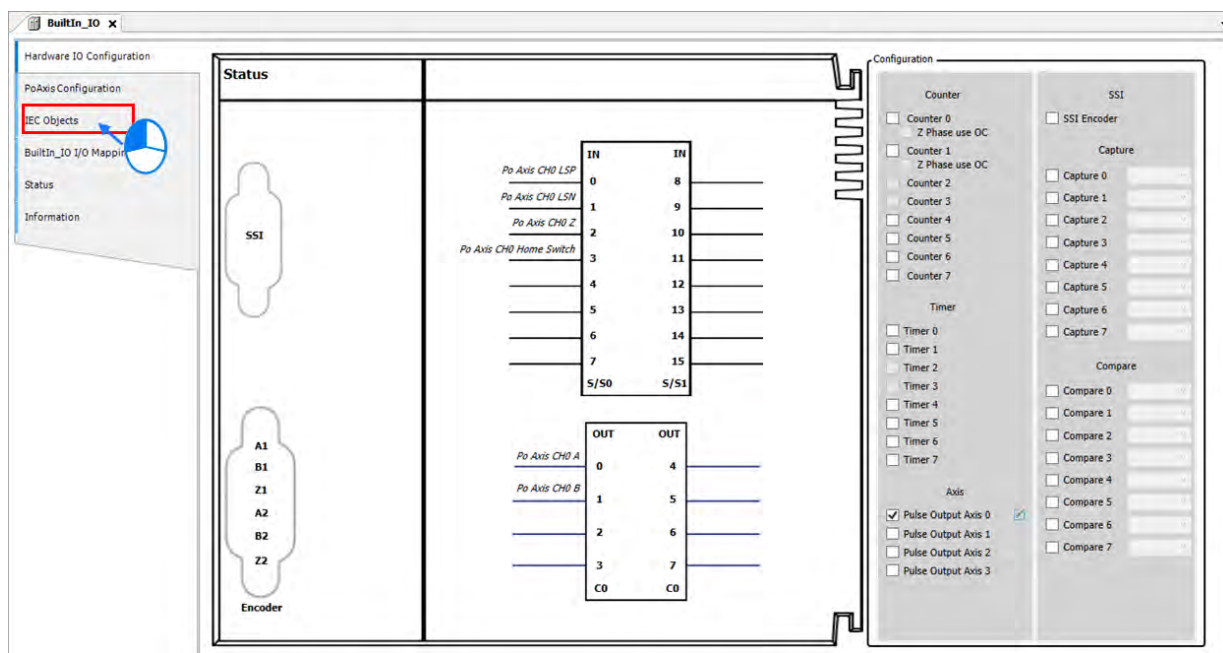
■ ⑦ 软件设定画面：可参考第 7.7.7.3 节 SSI 编码器功能设定

■ ⑧ Homing Setting

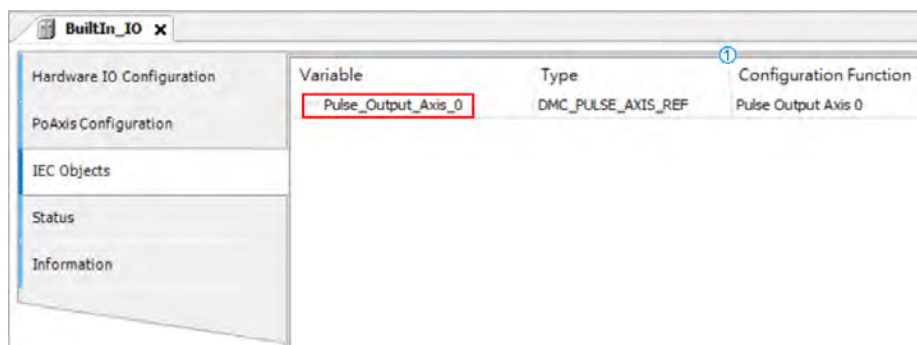
名称	功能	设定数值 (默认值)
Homing Mode	设定原点模式	(Mode 35)
Homing speed during search for switch	设定寻原点开关速度	(100)
Homing speed during search for z phase pulse	设定寻 Z 相速度	(50)

名称	功能	设定数值 (默认值)
Homing Acceleration	原点速度	(1000)

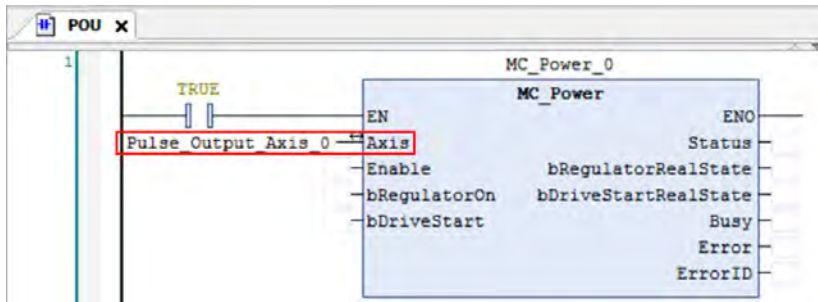
- Pulse Axis 使用在程序中：Pulse Axis 使用在 POU 中，需使用此变量在 POU 中的运动功能块使用。
 - BuiltIn_IO 画面中单击 IEC Objects



- IEC Objects 画面中① 为配置功能介绍，针对 POU 中使用的轴名称需填 Pulse_Output_Axis_0。

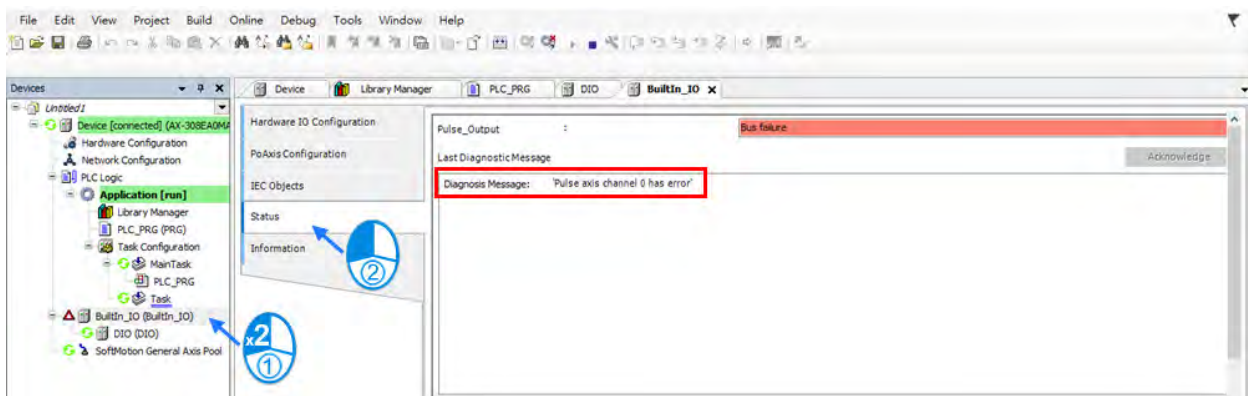


- 在 POU 中针对 MC_Power 功能块，输入的轴名称为 Pulse_Output_Axis_0。

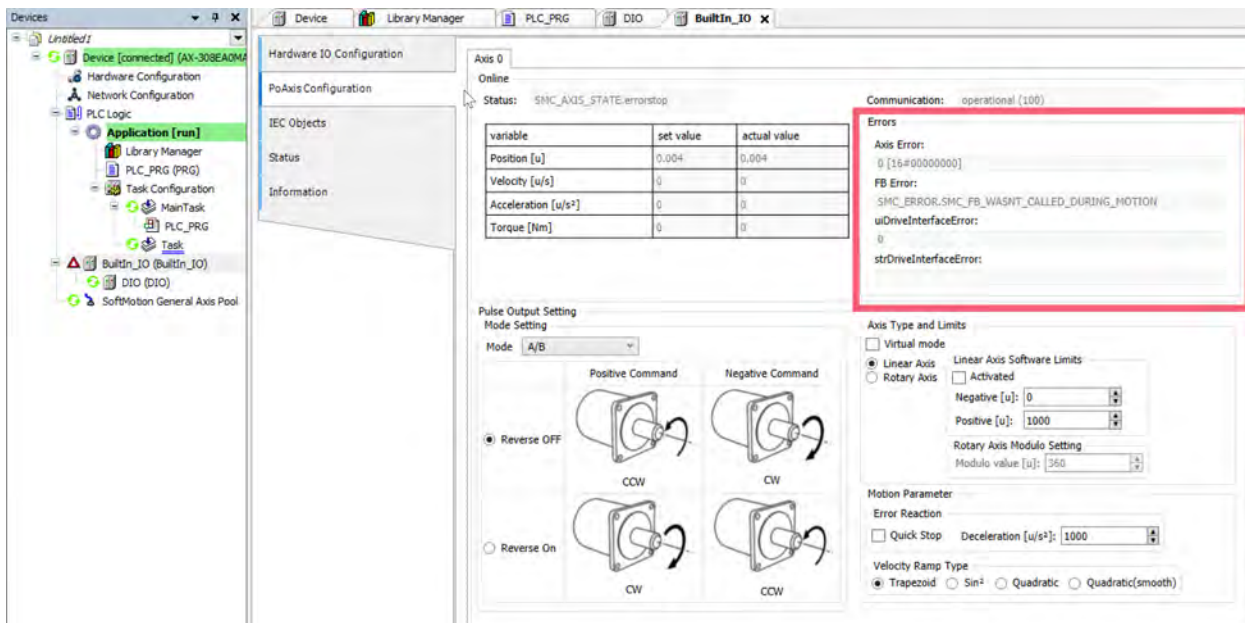


7.7.7.7 高速 IO 错误确认

当 Pulse Output Axis 发生错误时会显示在 BuiltIn_IO 的 Status，并告知哪一组脉冲轴发生错误。



接着用户可以在 PoAxis Configuration 中，以在线监控的方式查看错误信息。



7.7.8 其它功能

7.7.8.1 更改现在位置

MC_SetPosition

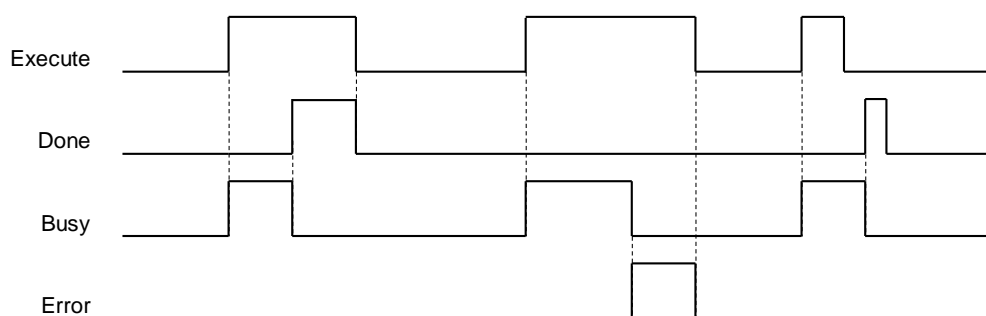
将伺服轴的指令当前位置变更为指定的目标位置。此外对编码器轴执行指令后，将反馈当前位置变更为指定的目标位置。

在指令当前位置变更的同时，反馈当前位置也会变更，两者间的追随误差保持不变。

此功能块作用是改变坐标系，不会使伺服及电机产生移动。

编码器轴可通过此功能块修改当前位置。

● 引脚时序图



7.7.8.2 软件极限

限制轴移动范围除了硬件极限以外还有软件极限。

当启动软件极限*需设定正极限与反极限范围值。用户操作错误可防止设备的损坏，默认不开启轴体极限功能。

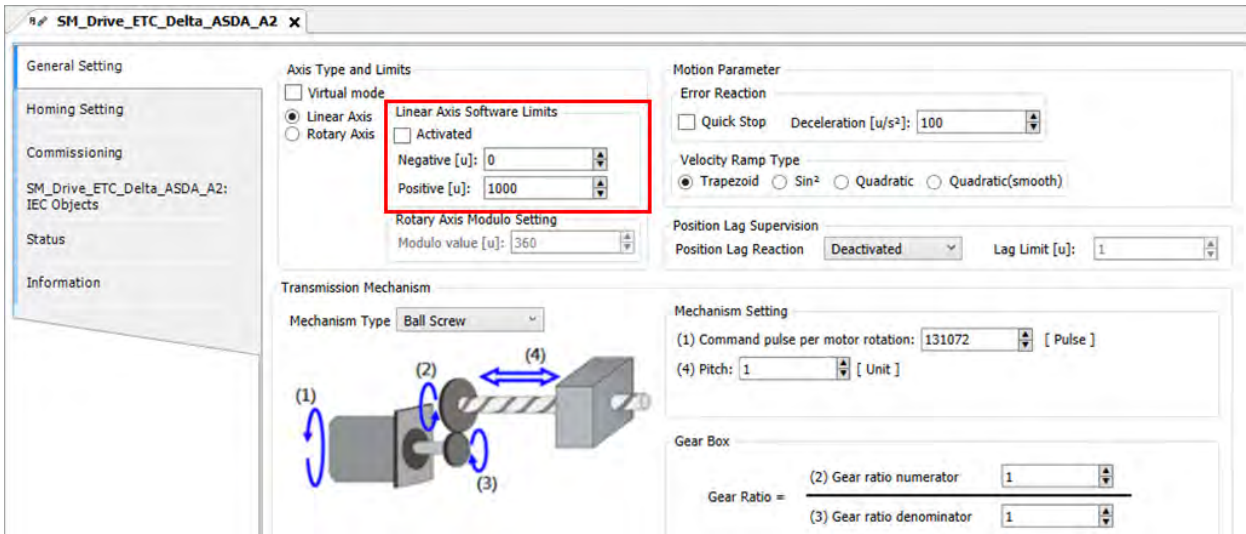


*注：

1. 停止范例可参考第 7.7.2.4 节。
2. 轴组遇极限行为：
 - ◆ 轴组运动命令前，各单轴都未超出极限：
 1. 单轴会在极限前停下，单轴不会发生错误。
 2. 轴组报错。
 - ◆ 轴组运动命令前，已有单轴在极限外：
 1. 单轴无动作。
 2. 轴组报错。

● 软件画面

可通过 DIADesigner-AX 进行设定：



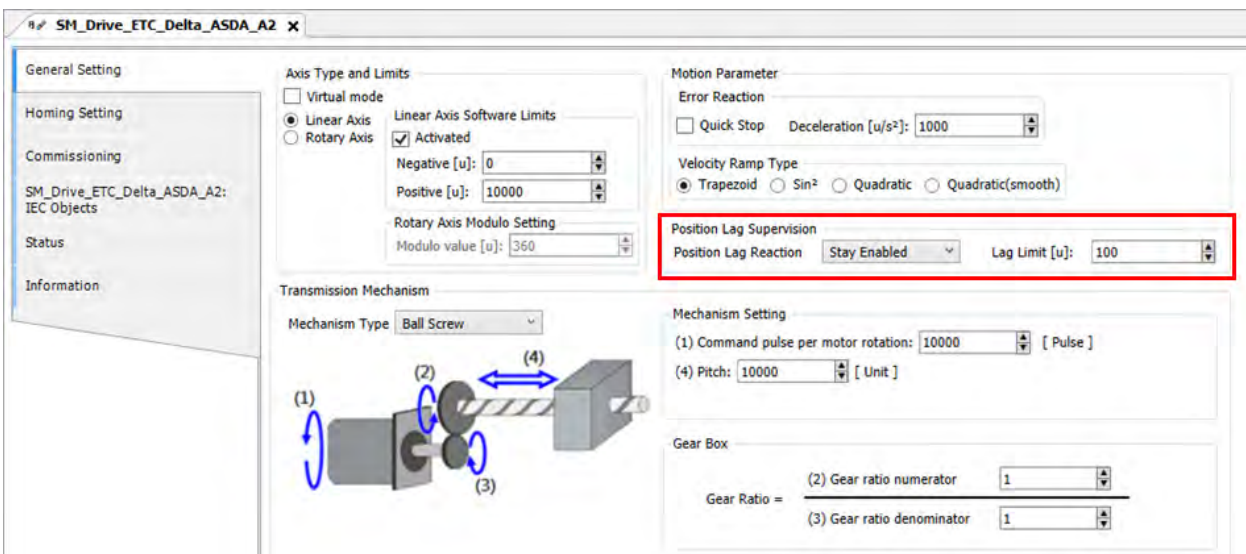
在设定页面中可调整正向/反向位置大小：

名称	数据类型	默认
Negative Position	LREAL	0.0
Positive Position	LREAL	10000.0

7.7.8.3 追随误差设定

执行轴移动时命令位置与实际位置会为 0，当移动过程中命令位置与实际位置差异太大时就会报错，默认未启动追随误差功能。

名称	功能
Deactivated	未启动
Disable drive	追随误差超出设定极限时，轴会 servo off
Do quickstop	追随误差超出设定极限时，轴会 quick stop
Stay enabled	追随误差超出设定极限时，轴维持 servo on



7.7.8.4 挺杆点功能

MC_DigitalCamSwitch

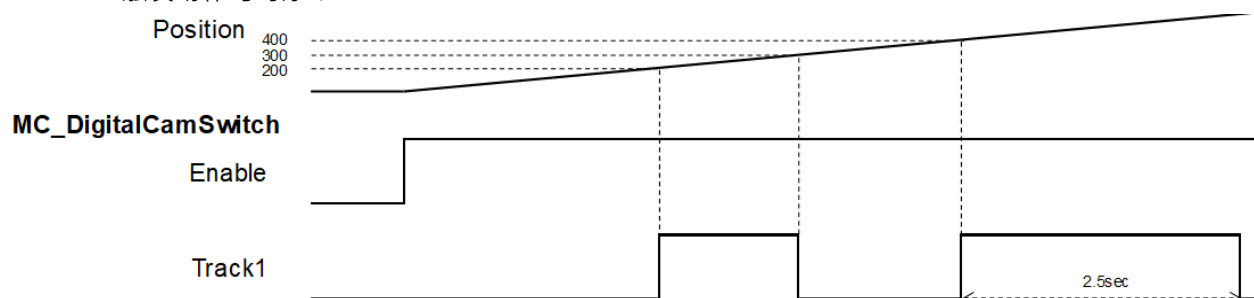
设定停杆点位置，当轴运行至指定位置后为 True，离开指定位置后 False。以下为设定范例：

- 设定范例：范例说明在 MC_DigitalCamSwitch 指令使用 2 个 Switch 在同一个 Track 之中。

- 参数设定：

参数名称	Type	Switch1	Switch2
TrackNumber	INT	1	1
FirstOnPosition [u]	REAL	200	400
LastOnPosition [u]	REAL	300	-
AxisDirection	INT	0=Both	0=Both
CamSwitchMode	INT	0=Position	1=TIME
Duration	TIME	-	2500ms

- 触发动作与时序：



- Track1 上的 Switch1 在位置达 200 时开启，当轴位置抵达 300 时 Switch1 关闭；
- 当位置达 400 时 Switch1 再次开启，经过 2500ms 后 Switch1 关闭。

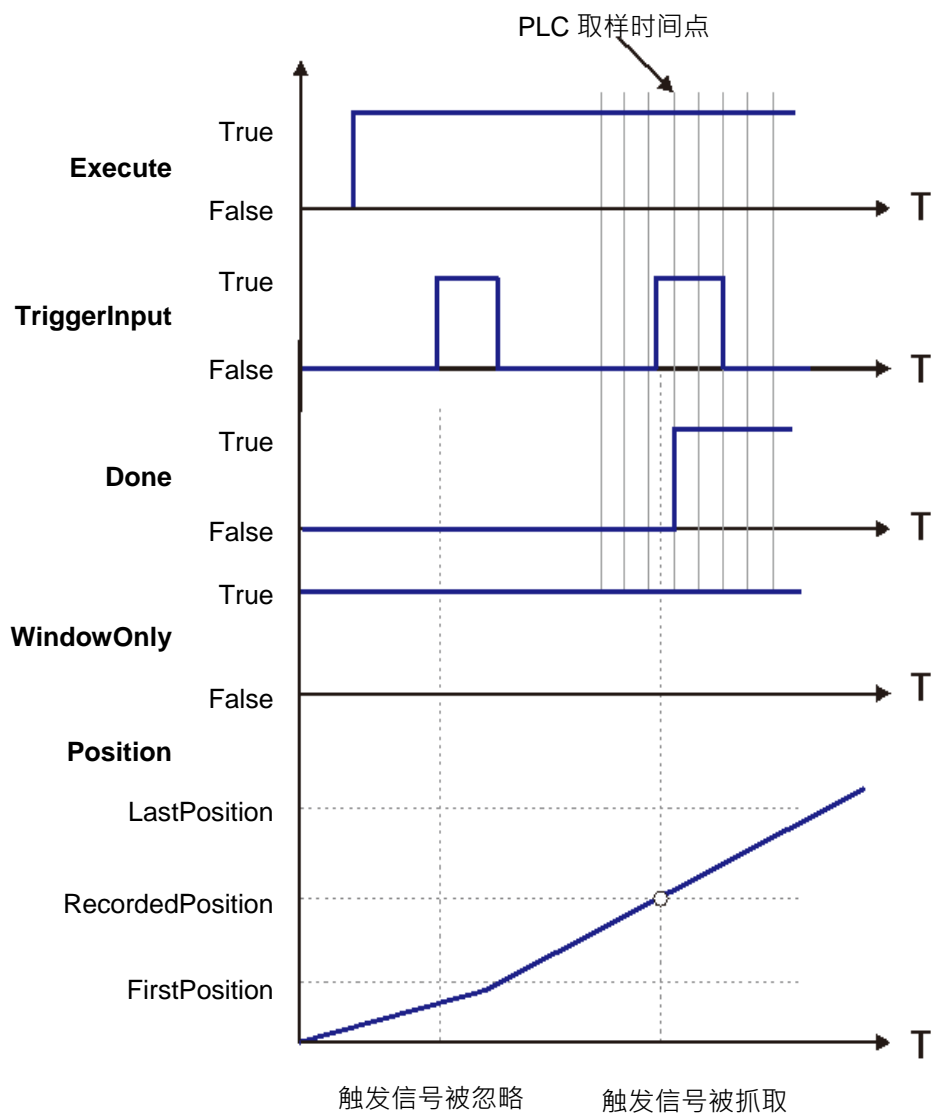
7.7.8.5 捕捉位置

MC_TorchProbe 使用传感器当作触发信号，在将信号输入至伺服驱动器中，当信号触发时记录伺服当下位置。每轴可以设定两个触发信号。MC_AbortTrigger 终止捕捉动作。

功能说明：

当 MC_TouchProbe 功能块的输入引脚 *Execute* 为 True 时，只会抓取第一次触发信号的有效位置值，之后的触发信号将被忽略。一个功能块实体 (instance) 只能对应一个 MC_TouchProbe 指令。如果在一个相同的轴有数个 MC_TouchProbe 功能块实例时，MC_TRIGGER_REF 需使用另外的 TouchProbeID，以便区分不同的 TouchProbe 动作。TouchProbeID 同时也对应到 MC_AbortTrigger。

MC_TouchProbe 的 window 功能作用如下说明：



在上图的第一次触发信号输入时，信号不被接受，因为轴位置没有到达 Window Mask 的区段。

在上图的第二次触发信号输入时，信号被接受，因为轴位置已经到达 Window Mask 的区段。并且在抓取信号完成后的下一个周期，输出信号 *Done* 为 True。

7

7.8 范例程序

下列针对本章节范例程序进行一些基本说明。

7.8.1 设备架构

下列节说明范例程序使用以下设备。

设备	构成范例
CPU	AX-308E
Power	DVP-PS02
伺服驱动器	台达 ASDA-A2-E
伺服电机	台达 ECMA-C

7.8.1.1 使用设备

关于使用设备的设置与接线，请参考下列手册。

设备	构成范例
CPU 和 Power	请参考此手册第二章
伺服驱动器	请参考台达伺服手册相关设置
EtherCAT 从站的接线	请参考台达伺服 A2-E 手册

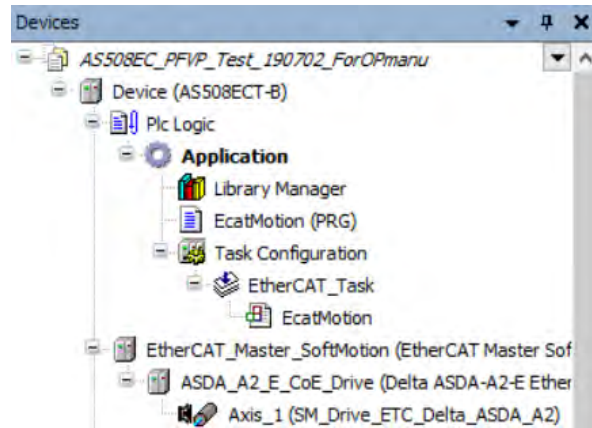
7.8.1.2 设定

下列节说明的范例程序使用以下设备。

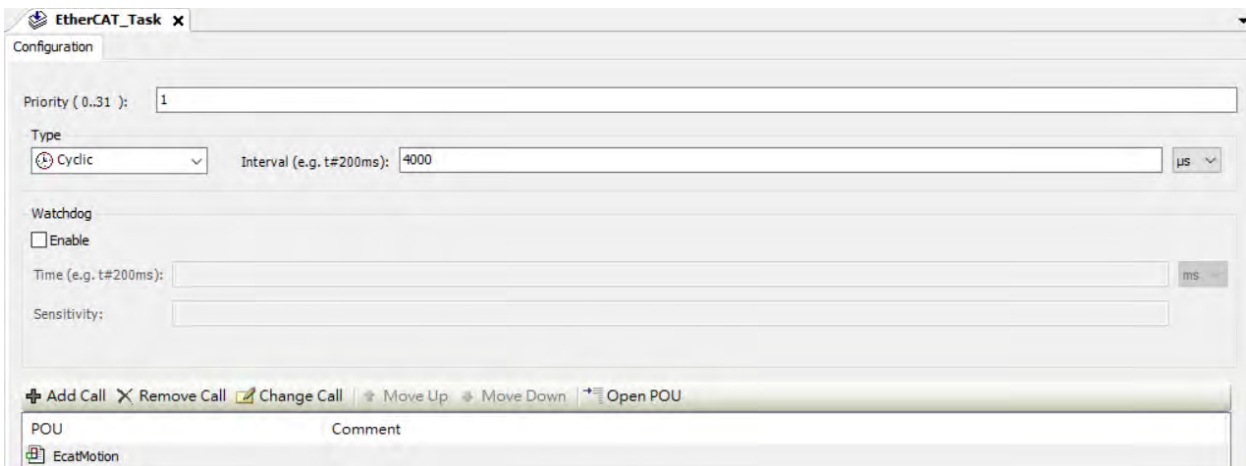
设备	伺服构成范例
控制器	本范例使用此手册第二章默认
运动控制设定	请参考此手册第七章
伺服参数	本范例使用 ASDA-A2-E 从站默认参数，齿比为 10000 : 10000

7.8.2 基本项目范例

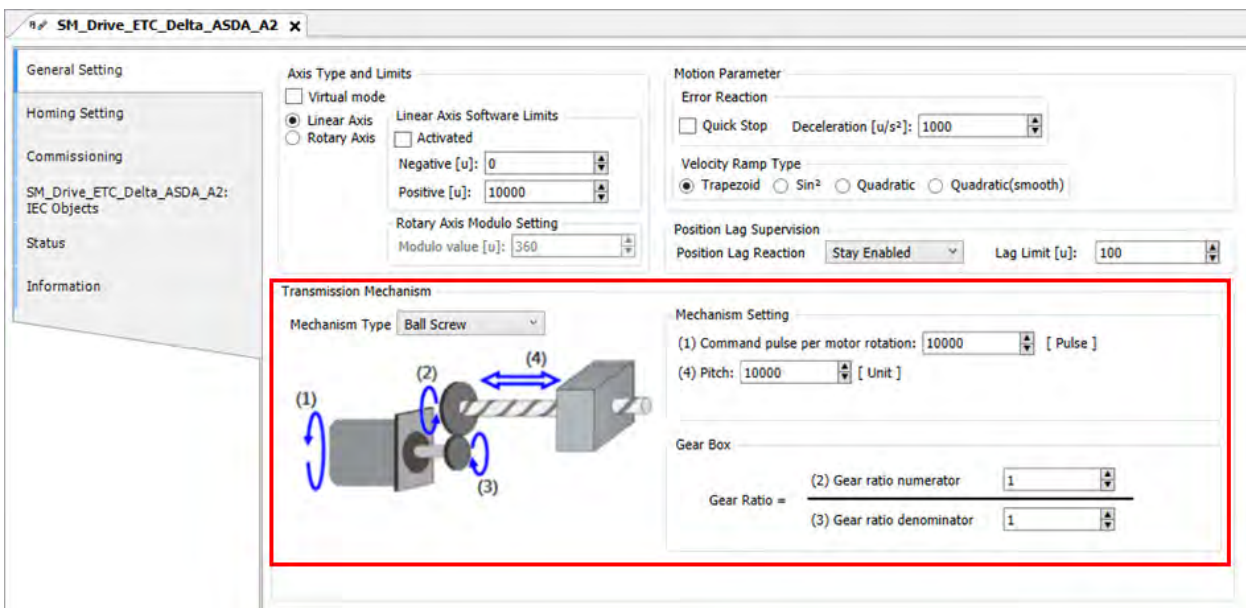
以下范例使用 EtherCAT task 内同一个 POU 做说明，所使用到的变量都会在此 POU task 内做宣告及使用（LD 及 ST 范例为了方便说明，POU 命名会使用不一致）



ECAT 同步时间设为 4ms



电子齿轮比设为 10000 : 10000



7.8.2.1 伺服系统 Servo On

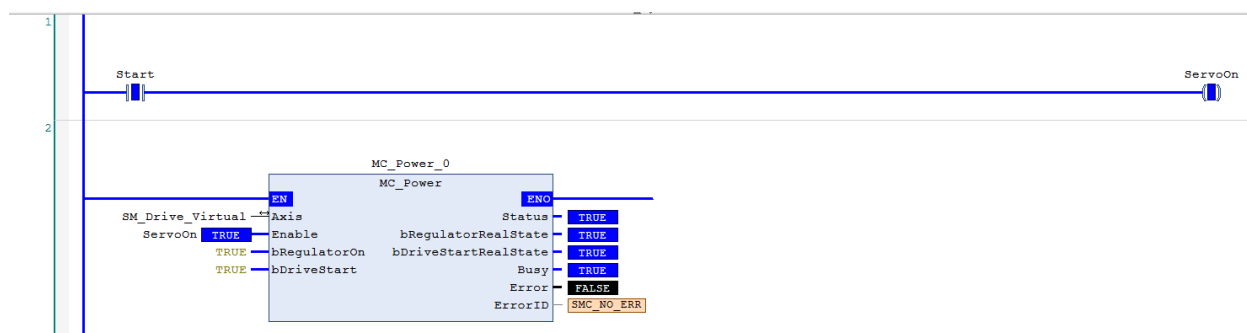
在此范例中，当建立 EtherCAT 数据通讯时，执行 MC_Power (Servo on) 指令用来使能伺服驱动器。此范例提供梯形图与 ST 语法两种。

程序中用到的主要变量

名称	数据类型	初始值	备注
SM_Drive_Virtual	AXIS_REF_SM3	-	虚拟轴变数
Start	BOOL	FALSE	当开启伺服时，切为 True 启动伺服 On

梯形图

当 Start 为 True 的时候，确认 EtherCAT 网络是否通讯正常激活，若通讯正常时则通过 ServoOn 输出引脚让 MC_Power 使能，当有看到输出引脚 Status 为 True 时，伺服 Servo On。



ST 语法

当 Start 为 True 的时候，确认 EtherCAT 网络是否通讯正常激活，若通讯正常时则通过 ServoOn 输出引脚让 MC_Power 使能，当有看到输出引脚 Status 为 True 时，伺服 Servo On。

同时也可以利用监控窗口观察变量输出状态，不需要特别对输出变量命名

```
IF Start THEN
    ServoOn :=TRUE;
ELSE
    ServoOn :=FALSE;
END_IF
```

```
//MC_Power
MC_Power_0(
    Axis:= SM_Drive_Virtual,
    Enable:= ServoOn,
    bRegulatorOn:= TRUE,
    bDriveStart:= TRUE,
    Status=> ,
    bRegulatorRealState=> ,
    bDriveStartRealState=> ,
    Busy=> ,
    Error=> ,
    ErrorID=> );
```

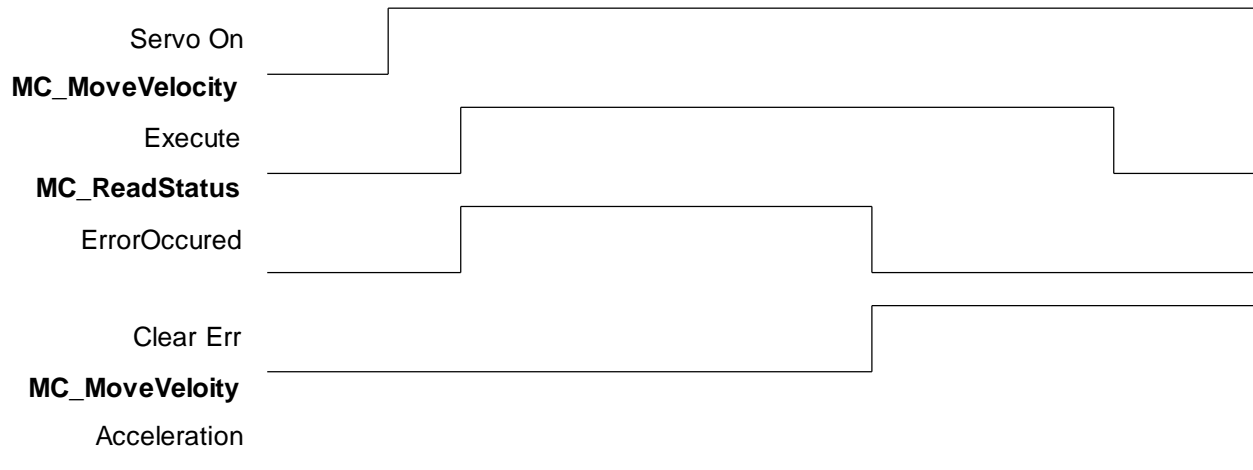

7.8.2.2 单轴错误重置与监控

在此范例中，可以通过监控表来监控变量状态的错误信息，以 MC_MoveVelocity 输入引脚加速度举例，当设置为 0 的时候，使能 Execute 为 True，发生功能块错误 Error 为 True 且 ErrorID 状态显示 Row Data 301 可在监控表利用变量查看完整错误信息为 SMC_MV_INVALID_ACCDEC_VALUES，查找手册故障排除后，直接通过 Execute False 到 True，可正常执行 MC_MoveVelocity 等速运动功能块。而 MC_Reset 是专门用来清除伺服错误。此范例提供梯形图与 ST 语法两种。

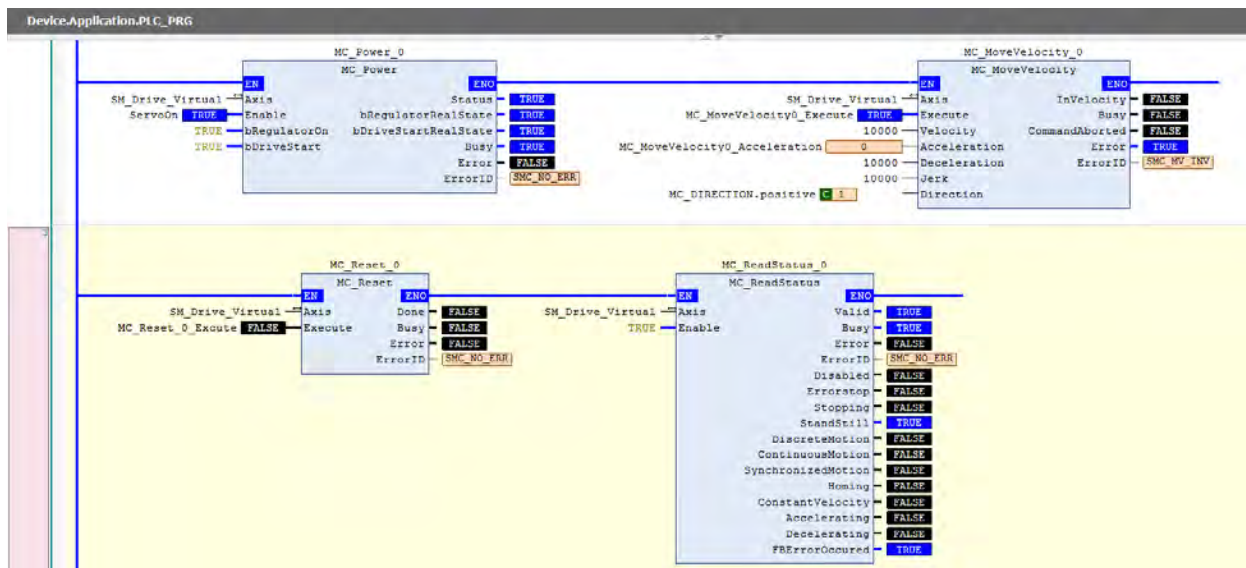
程序中用到的主要变量

名称	数据类型	初始值	备注
SM_Drive_Virtual	AXIS_REF_SM3	-	虚拟轴变数
ServoOn	BOOL	FALSE	用来 Enable MC_Power
MC_MoveVelocity0_Execute	BOOL	FALSE	速度指令 Execute 引脚
MC_MoveVelocity0_Acceleration	LREAL	0	速度指令 Acceleration 引脚，用来设定加速度
MC_DIRECTION.positive	MC_Direction	-	指定移动方向-正向
FBErrorOccured	MC_ReadStatus	FALSE	当功能块错误时引脚为 True
ClearErr	BOOL	FALSE	当 FBErrorOccured 为 True 时可通过此接点触发 SMC_ClearFBError 清掉 FB 错误

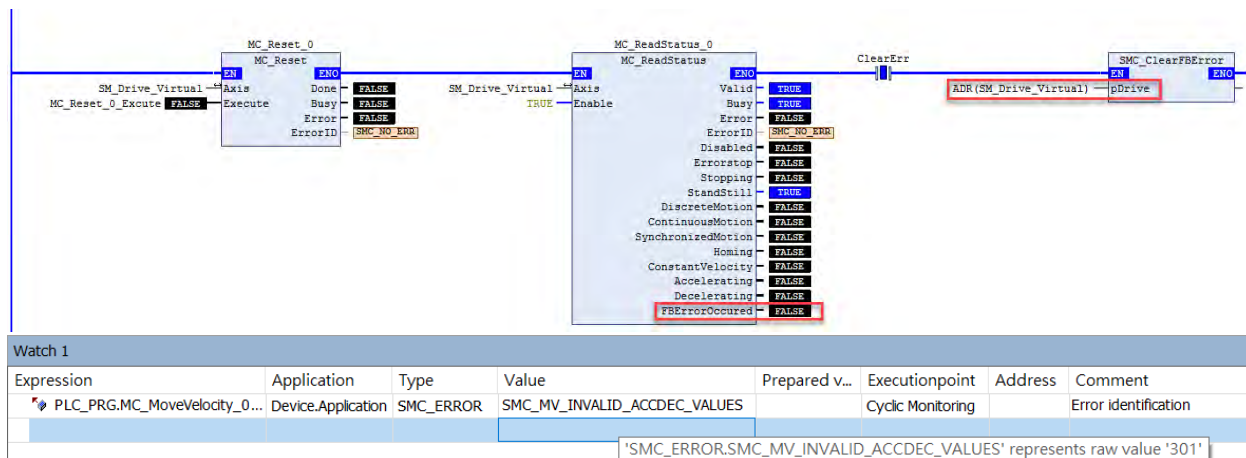
时序图



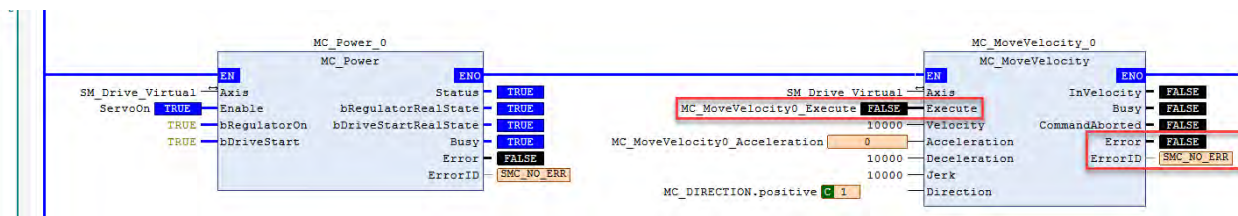
梯形图



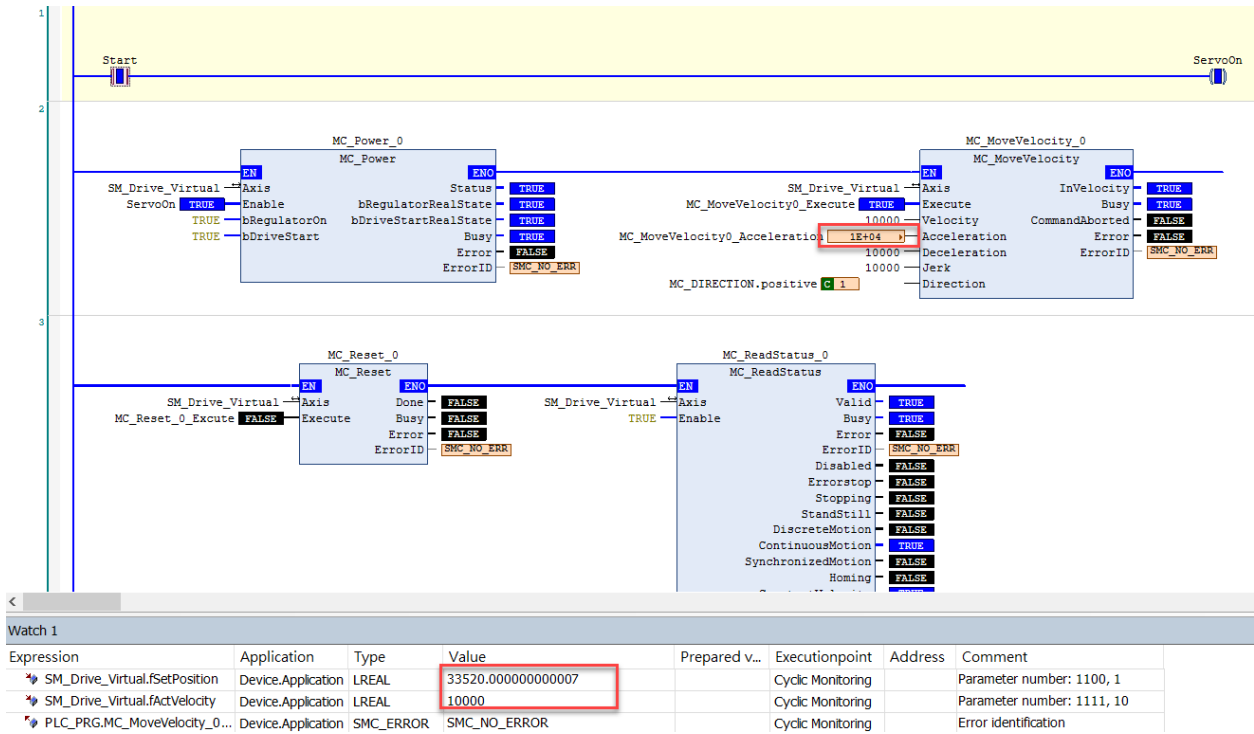
当运动功能块错误发生时，可以通过 SMC_ClearFBError 清除 MC_ReadStatus 的输出引脚 FBErrorOccured，因 SMC_ClearFBError 输入需要通过指针方式传递，所以需要加 ADR(输入变量)，利用 bool 清除 FB 错误 flag。



原 MC_MoveVelocity 只要把输入引脚 Execute 关闭，则输出错误引脚会被更新。



把 MC_MoveVelocity 加速度引脚设为 10000 后再重新启动 Execute 为 True，可以看到功能块输出为 Busy 且监控表的 fSetVelocity 与 fSetPosition 都有显示值代表正常动作。



ST 语法

```

MC_MoveVelocity_0(
  Axis:= SM_Drive_Virtual,
  Execute:= MC_MoveVelocity0_Execute,
  Velocity:= 10000,
  Acceleration:= MC_MoveVelocity0_Acceleration,
  Deceleration:= 10000,
  Jerk:= 10000,
  Direction:= MC_DIRECTION.positive,
  InVelocity=> ,
  Busy=> ,
  CommandAborted=> ,
  Error=> ,
  ErrorID=>);
    
```

```

MC_ReadStatus_0(
  Axis:= SM_Drive_Virtual,
  Enable:= TRUE);
    
```

把 MC_MoveVelocity 加速度引脚设为 10000 后再重新启动 Execute 为 True 可以看到功能块输出为 Busy 且监控表的 fSetVelocity 与 fSetPosition 都有显示值代表正常动作。

```

MC_MoveVelocity_0(
  Axis:= SM_Drive_Virtual,
  Execute:= MC_MoveVelocity0_Execute,
  Velocity:= 10000,
  Acceleration:= MC_MoveVelocity0_Acceleration := 10000,
  Deceleration:= 10000,
  Jerk:= 10000,
  Direction:= MC_DIRECTION.positive,
  InVelocity=> ,
    
```

```

Busy=> ,
CommandAborted=> ,
Error=> ,
ErrorID=> );

```

```

MC_ReadStatus_0(
  Axis:= SM_Drive_Virtual,
  Enable:= TRUE );

```

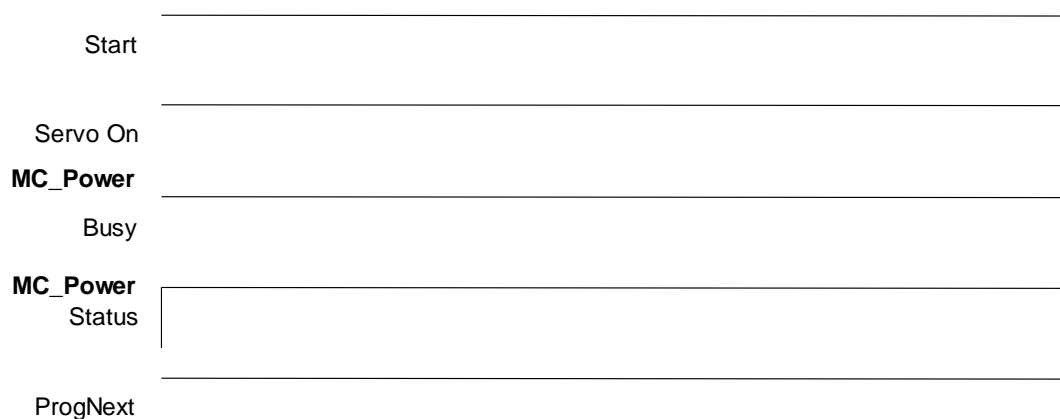
7.8.2.3 监控指令错误

在此范例中，如果执行 MC_Power (Servo On) 指令时出错，则不执行进一步处理。ProgNext 变量指示是否可以进一步执行。此范例提供梯形图与 ST 语法两种。

程序中用到的主要变量

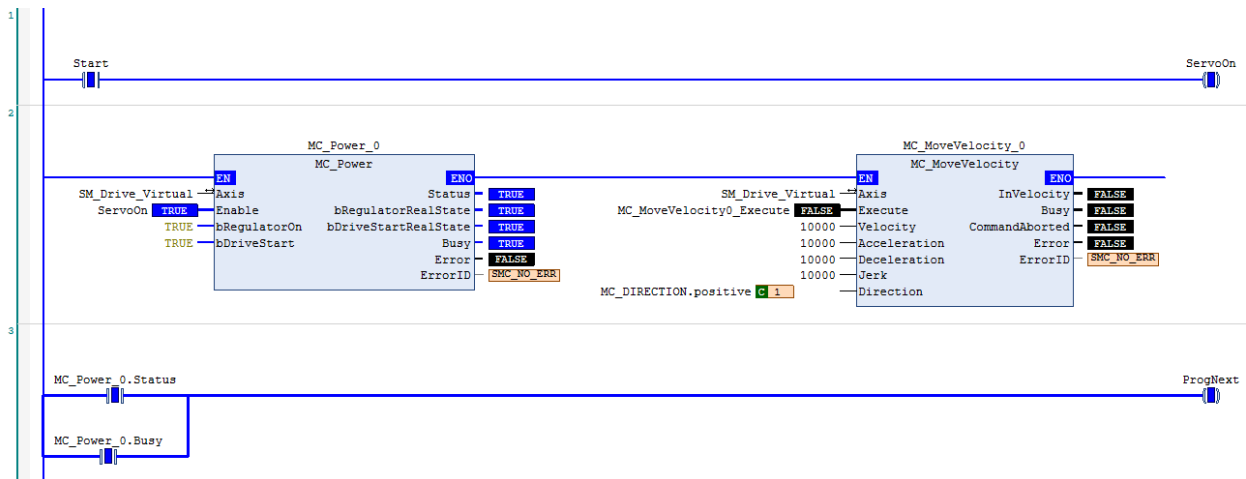
名称	数据类型	初始值	备注
SM_Drive_Virtual	AXIS_REF_SM3	-	虚拟轴变数
ServoOn	BOOL	FALSE	用来 Enable MC_Power
ProgNext	BOOL	FALSE	ProgNext 变量指示是否可以进一步执行
MC_Power_0.Status	BOOL	FALSE	状态为 True 时，轴代表已经准备好移动
MC_Power_0.Busy	BOOL	FALSE	状态为 True 时，FB 执行尚未完成

时序图



梯形图

执行进一步处理之前检查 MC_Power 时是否发生任何错误。



ST 语法

IF Start THEN

 ServoOn :=TRUE;

ELSE

 ServoOn :=FALSE;

END_IF

IF (MC_Power_0.Status=TRUE) OR (MC_Power_0.Busy=TRUE) THEN

 ProgNext :=TRUE;

ELSE

 ProgNext :=FALSE;

END_IF

//MC_Power

MC_Power_0(

 Axis:= SM_Drive_Virtual,

 Enable:= ServoOn,

 bRegulatorOn:= TRUE,

 bDriveStart:= TRUE,

 Status=> ,

 bRegulatorRealState=> ,

 bDriveStartRealState=> ,

 Busy=> ,

 Error=> ,

 ErrorID=>);

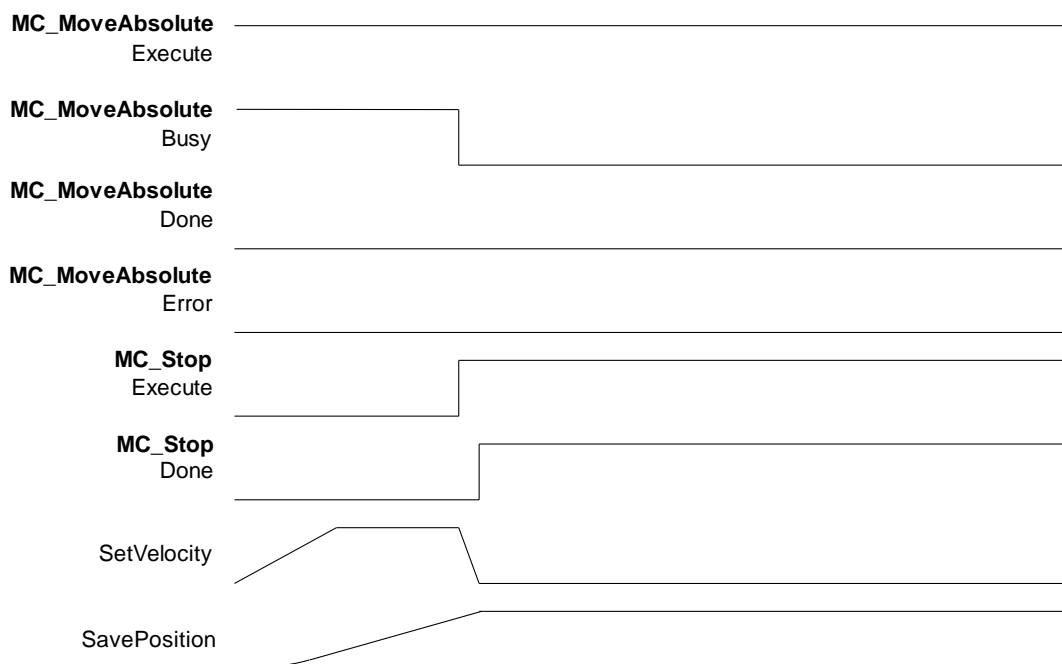
7.8.2.4 单轴操作立即停止

在此范例中，在执行 MC_MoveAbsolute(绝对寻址)指令期间，如果有轻微的错误发生，可以利用 MC_Stop 来停止单轴运动指令的操作。此范例提供梯形图与 ST 语法两种。

程序中用到的主要变量

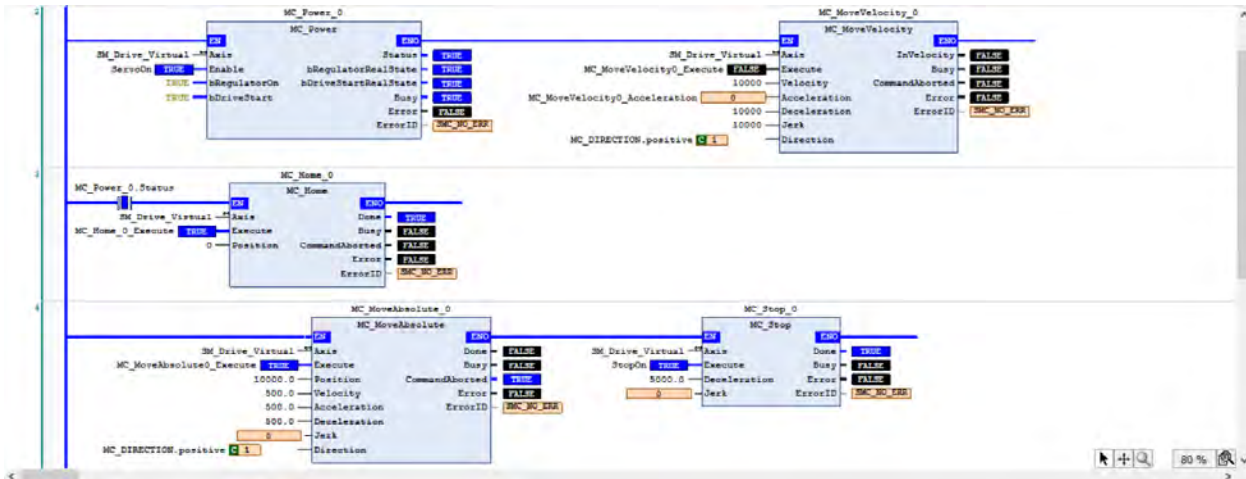
名称	数据类型	初始值	备注
SM_Drive_Virtual	AXIS_REF_SM3	-	虚拟轴变数
ServoOn	BOOL	FALSE	用来 Enable MC_Power
MC_MoveAbsolute0_Execute	BOOL	FALSE	绝对寻址指令 Execute 引脚
MC_DIRECTION.positive	MC_Direction	-	指定移动方向-正向(旋转轴有效)
StopOn	BOOL	FALSE	状态为 True 时，启动 MC_Stop
MC_Stop_0.Done	BOOL	FALSE	状态为 True 时，MC_Stop 停止完成

时序图



梯形图

确认 MC_Power 输出 Status 正常，执行原点回归，当原点回归完成后，执行绝对位置定位 (MC_MoveAbsolute) 此时若要紧急停止，则执行 MC_Stop，此时绝对寻址工作被打断，输出引脚 CommandAborted 为 True，轴会依照所设置的减速度去作停止，完成紧急停止后 MC_Stop 输出引脚 Done 为 True。



ST 语法

动作流程同 LD 表示，MC_Home 完后状态机在 StandStill。

```
//MC_Power
MC_Power_0(
    Axis:= SM_Drive_Virtual,
    Enable:= ServoOn,
    bRegulatorOn:= TRUE,
    bDriveStart:= TRUE,
    Status=> ,
    bRegulatorRealState=> ,
    bDriveStartRealState=> ,
    Busy=> ,
    Error=> ,
    ErrorID=> );

//MC_Home
IF MC_Power_0.Status THEN
    MC_Home_0(
        Axis:= SM_Drive_Virtual,
        Execute:= MC_Home_0_Execute,
        Position:= 0,
        Done=> ,
        Busy=> ,
        CommandAborted=> ,
        Error=> ,
        ErrorID=>);
END_IF
```

执行绝对位置定位 (MC_MoveAbsolute) 此时若要紧急停止，则执行 MC_Stop，此时绝对寻址工作被打断，状态机会在 Stopping。

```
//MC_MoveAbsolute & MC_Stop
MC_MoveAbsolute_0(
    Axis:= SM_Drive_Virtual,
    Execute:= MC_MoveAbsolute0_Execute,
    Position:= 10000.0,
    Velocity:= 500.0,
    Acceleration:= 500.0,
    Deceleration:= 500.0,
    Jerk:= ,
    Direction:= MC_DIRECTION.positive,
    Done=> ,
    Busy=> ,
    CommandAborted=> ,
    Error=> ,
    ErrorID=>);

MC_Stop_0(
    Axis:= SM_Drive_Virtual,
    Execute:= StopOn,
    Deceleration:= 5000.0,
    Jerk:= ,
    Done=> ,
    Busy=> ,
    Error=> ,
    ErrorID=>);
```

7.8.2.5 原点回归

在此范例中，通过原点回归指令让用户了解如何做原点回归，回 Home 模式台达目前支持 0-35，用到的 OD 为 6098(Homing method)/6099sub1(Speed during search for switch)/6099sub2(Speed during search for zero)。详细内容可以参考台达高阶交流伺服驱动器 ASDA-A2-E 系列应用技术手册。

以此范例为例

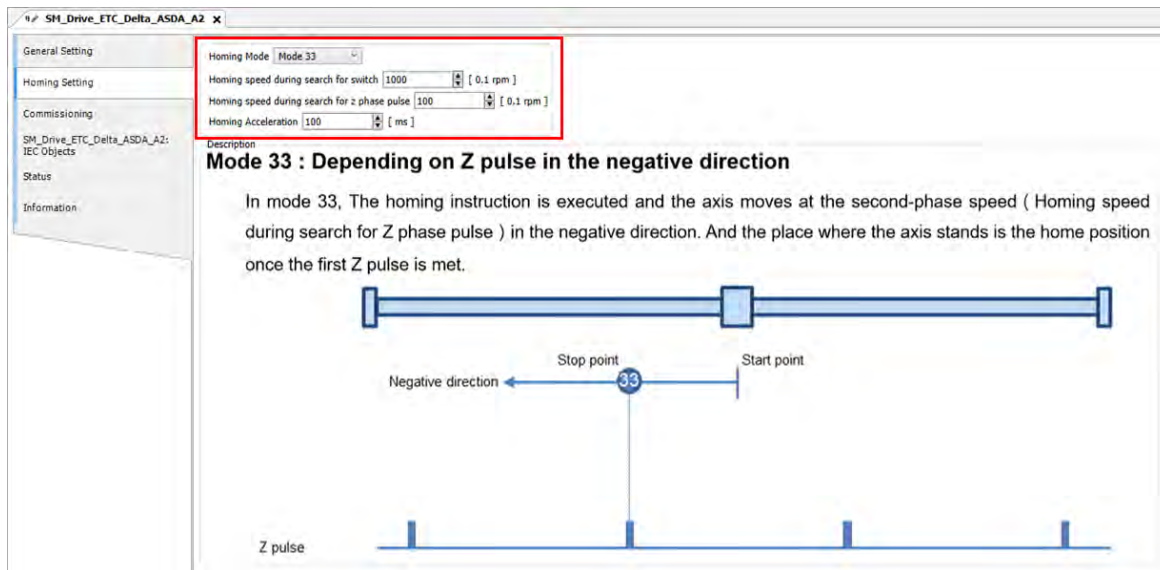
在 EtherCAT slave 增加 A2-E 伺服后单击内容的起始参数加入上述所说的 OD，并给予初始值

Homing method =33 (遇 Z 脉冲进行复位)

Speed during search for switch =1000[单位 1.0rpm] (以 100rpm 速度寻找极限开关)

Speed during search for zero =100[单位 0.1rpm] (以 10rpm 寻找零点)

设定完成后再 LD/ST 选取 MC_Home 在运作的时候就会对应以上 OD 对相对应 Homing method

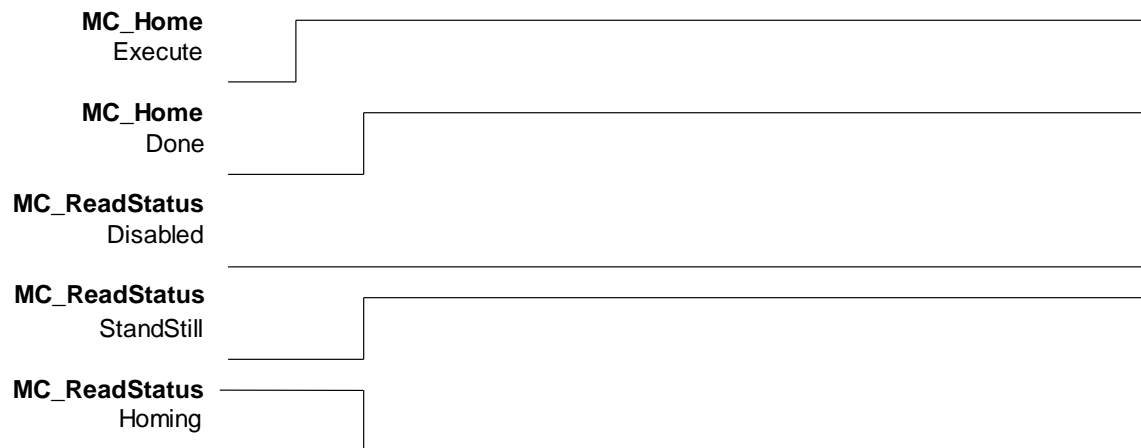


此范例提供梯形图与 ST 语法两种。

程序中用到的主要变量

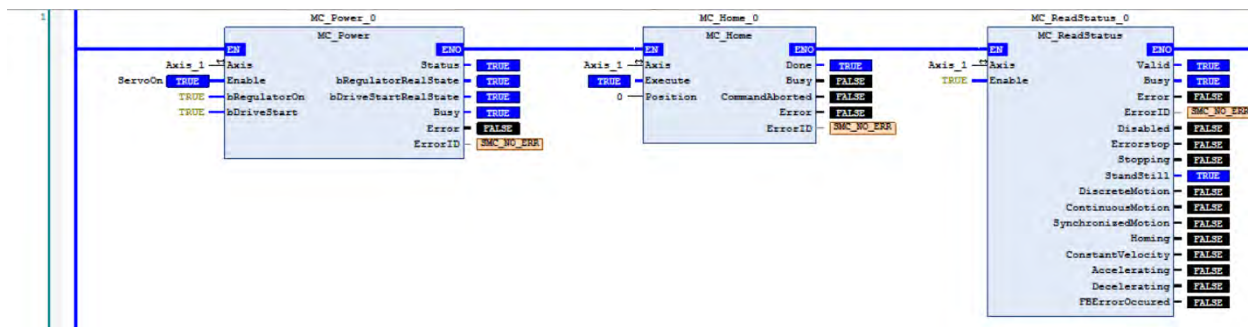
名称	数据类型	初始值	备注
Axis_1	AXIS_REF_SM3	-	实体轴变量
ServoOn	BOOL	FALSE	用来 Enable MC_Power

时序图



梯形图

确认 MC_Power 输出 Status 正常 · 状态机为 Standstill · 执行 MC_Home 原点回归 · 此时状态机为 homing 当原点回归完成后 · 状态机切为 Standstill。



ST 语法

动作流程同 LD 表示，MC_Home 完后状态机在 StandStill，输出状态可利用赋与变量或是通过监控表去查看。

MC_Home_0(

```
Axis:= Axis_1,
Execute:= ,
Position:= 0,
Done=> ,
Busy=> ,
CommandAborted=> ,
Error=> ,
ErrorID=> );
```

MC_ReadStatus_0(

```
Axis:= Axis_1,
Enable:= TRUE,
Valid=> ,
Busy=> ,
Error=> ,
ErrorID=> ,
Disabled=> ,
Errorstop=> ,
Stopping=> ,
StandStill=> ,
DiscreteMotion=> ,
ContinuousMotion=> ,
SynchronisedMotion=> ,
Homing=> ,
ConstantVelocity=> ,
Accelerating=> ,
Decelerating=> ,
FBErrorOccured=>);
```

7.8.2.6 绝对值定位

在此范例中，通过绝对寻址指令让用户了解如何做单段速位移。

此范例提供梯形图与 ST 语法两种

程序中用到的主要变量

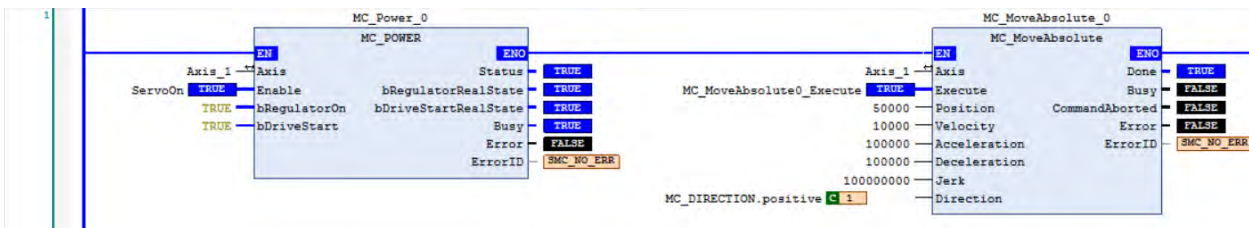
名称	数据类型	初始值	备注
Axis_1	AXIS_REF_SM3	-	实体轴变量
ServoOn	BOOL	FALSE	用来 Enable MC_Power
MC_MoveAbsolute0_Execute	BOOL	FALSE	绝对寻址指令 Execute 引脚
MC_DIRECTION.positive	MC_Direction	-	指定移动方向-正向(旋转轴有效)

时序图



梯形图

确认 MC_Power 输出 Status 正常，执行绝对位置定位 (MC_MoveAbsolute) 从起始位置 0→设定位置 50000。



ST 语法

```
MC_Home_0(
  Axis:= Axis_1,
  Execute:= ,
  Position:= 0,
  Done=> ,
  Busy=> ,
  CommandAborted=> ,
  Error=> ,
  ErrorID=> );
```

```
MC_MoveAbsolute_0(
  Axis:= Axis_1,
  Execute:= MC_MoveAbsolute0_Execute,
  Position:= 50000,
  Velocity:= 10000,
  Acceleration:= 100000,
  Deceleration:= 100000,
  Jerk:= 100000,
  Direction:= SM3_Basic.MC_DIRECTION.positive,
  Done=> ,
  Busy=> ,
  CommandAborted=> ,
  Error=> ,
  ErrorID=> );
```

7.8.2.7 凸轮动作中切换凸轮表

在此范例中，可在执行凸轮动作的过程中切换凸轮表。

增加主从轴并且建立两个 MC_CamIn 指令来实现切换两个凸轮表。两个凸轮输出的设定中配置不同的输出参数，主轴的指令位置为 3000 以下时，使用 CamTable1，当超过 3000 时，切换为 CamTable2。

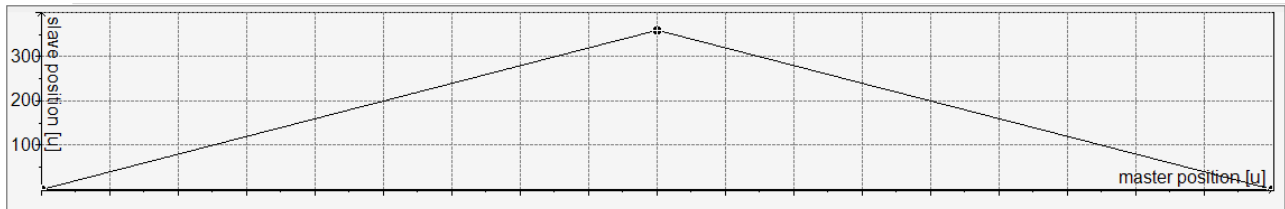
此范例提供梯形图与 ST 语法两种。

程序中使用的主要变量:

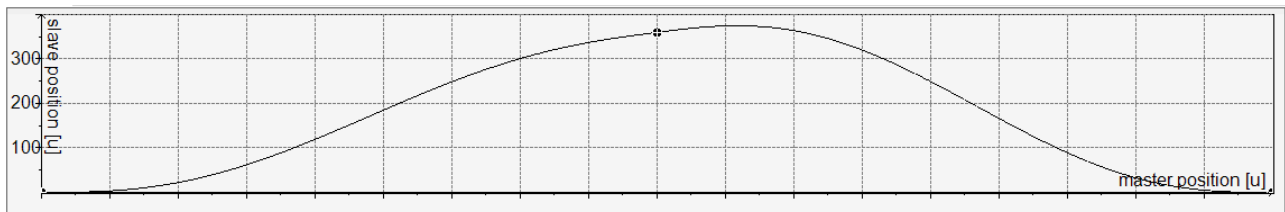
名称	变量类型	初始值	注释
Axis_Master	AXIS_REF_VIRTUAL_SM3	-	主轴的相关轴变量
Axis_Slave	AXIS_REF_VIRTUAL_SM3	-	从轴的相关轴变量
CamTable1	MC_CAM_REF	-	凸轮表 1 相关变量
CamTable2	MC_CAM_REF	-	凸轮表 2 相关变量
StartFlag	BOOL	FALSE	若此变量为 TRUE 且与轴通讯正常，则启动 Servo On，并开始执行后续动作
MC_Power0_Status	BOOL	FALSE	主轴 MC_Power 输出 Status 的变量 Servo On 时为 TRUE
MC_Power1_Status	BOOL	FALSE	从轴 MC_Power 输出 Status 的变量 Servo On 时为 TRUE

名称	变量类型	初始值	注释
MC_Home0_Done	BOOL	FALSE	主轴 MC_Home 输出 Done 的变量 当执行原点回归完成后为 TRUE
MC_Home1_Done	BOOL	FALSE	从轴 MC_Home 输出 Done 的变量 当执行原点回归完成后为 TRUE
MC_MoveAbs_Busy	BOOL	FALSE	主轴 MC_MoveAbsolute 输出 Busy 的变量 当功能块执行后轴在动作中为 TRUE
CamTableSelect	MC_CAM_REF	-	当下选择对应的凸轮表变量
CamTable1_En	BOOL	FALSE	将 CamTable1 作为凸轮表使用时为 TRUE
CamTable2_En	BOOL	FALSE	将 CamTable2 作为凸轮表使用时为 TRUE
CamTableID	MC_CAM_ID	-	此为描述凸轮表的内部数据结构，由 MC_CamTableSelect 生成，并作为 MC_CamIn 的输入
MC_CamIn1_InSync	BOOL	FALSE	凸轮表 1 的 MC_CamIn 输出 InSync 的变量 当主从轴凸轮同步时为 TRUE
MC_CamIn2_InSync	BOOL	FALSE	凸轮表 2 的 MC_CamIn 输出 InSync 的变量 当主从轴凸轮同步时为 TRUE

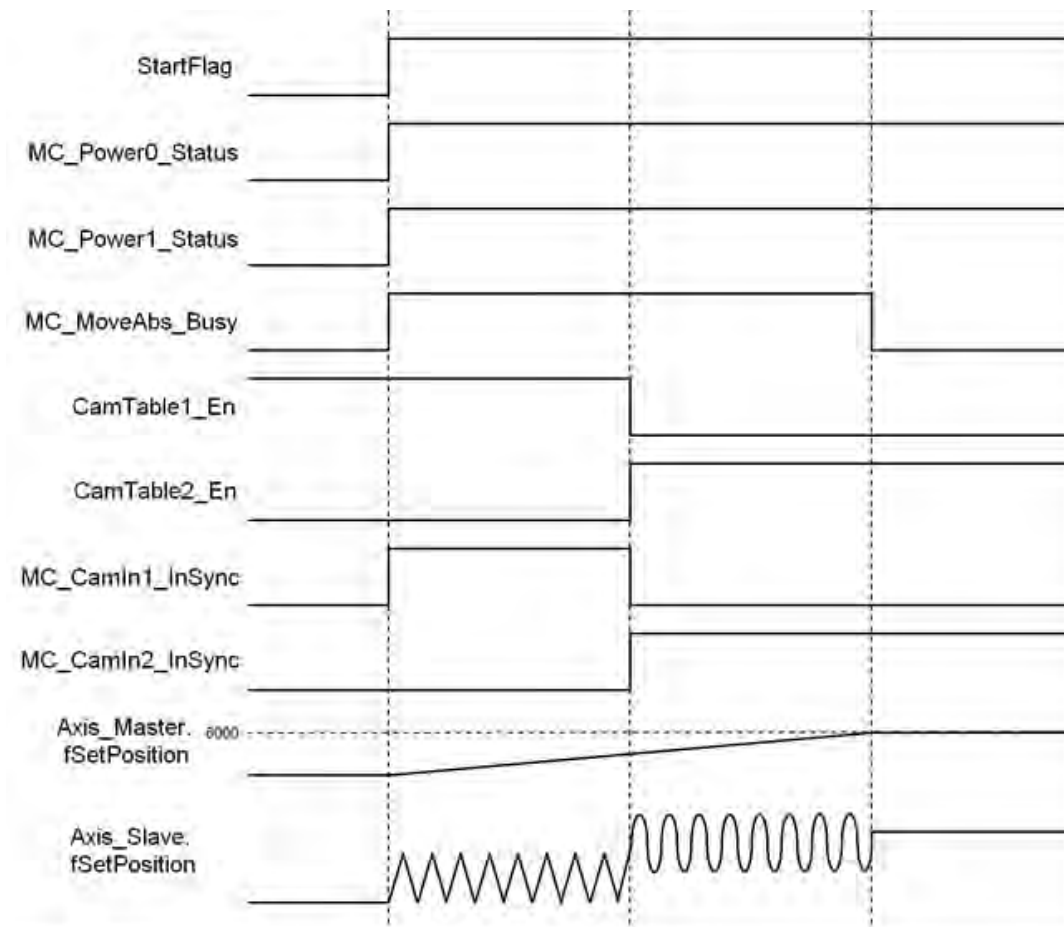
CamTable1 :



CamTable2 :



时序图：

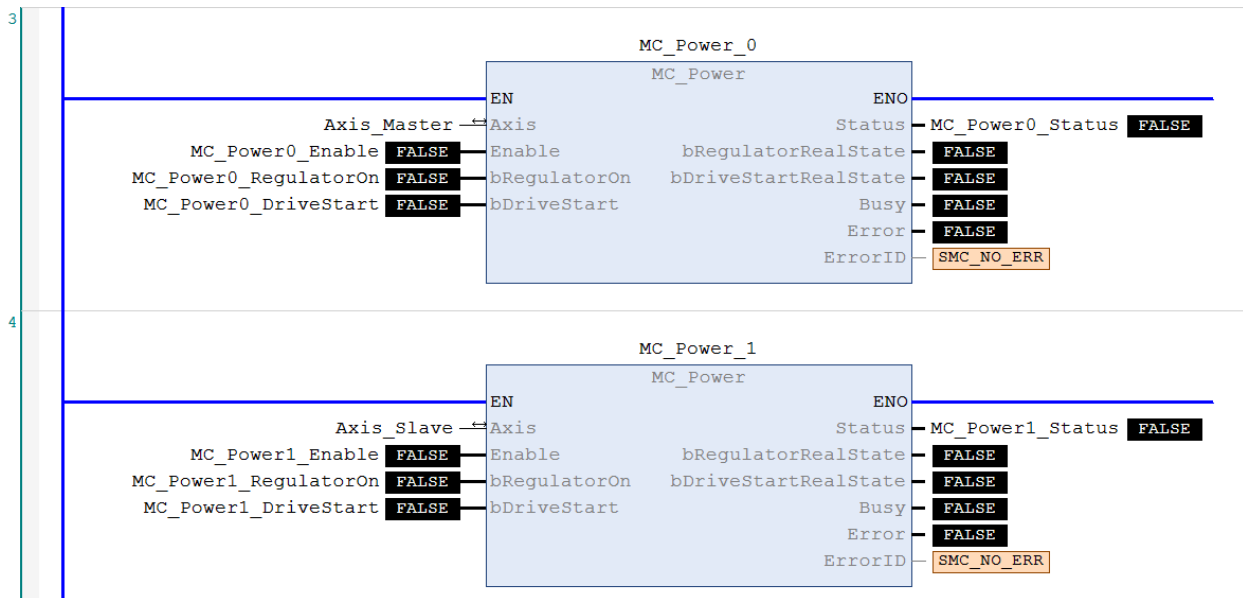


梯形图程序：

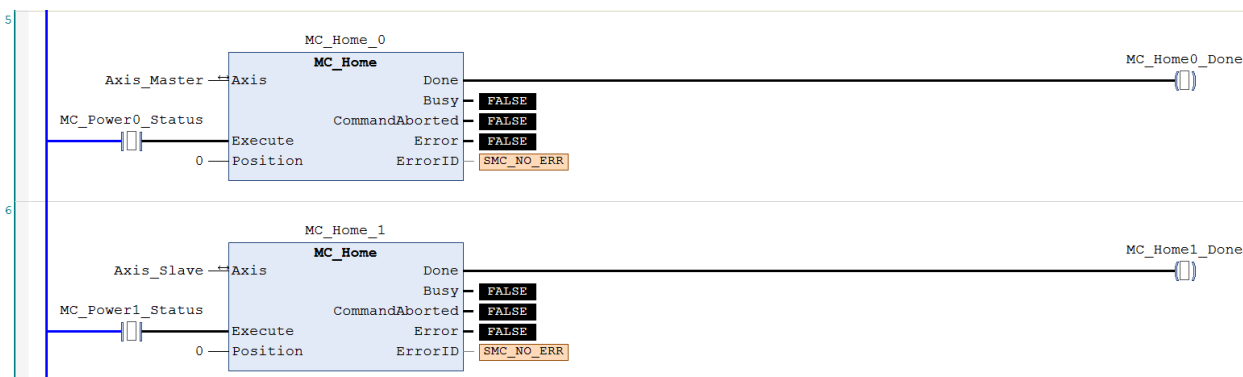
设定接点 StartFlag 为 True 后会分别确认主从轴的通讯是否有正常的执行。



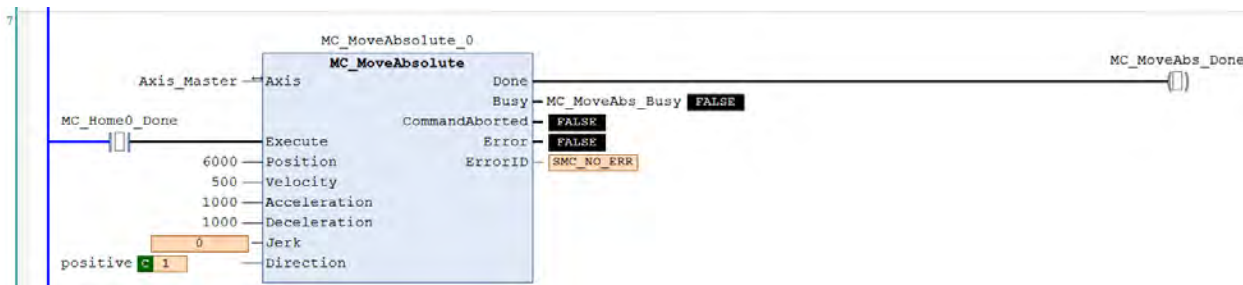
若在正常的情况下，会再分别对主从轴设定为 Servo On 状态。



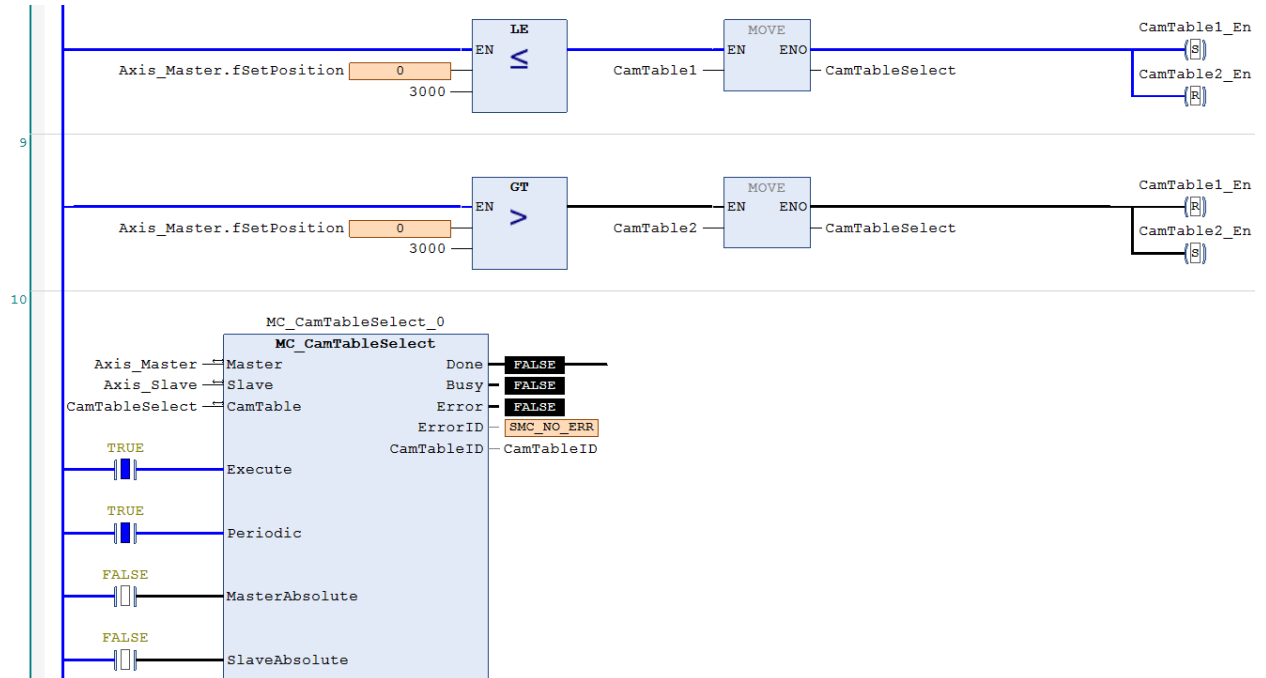
主从轴为 Servo On 状态且处于原点未确定状态，先执行一次原点回归动作。



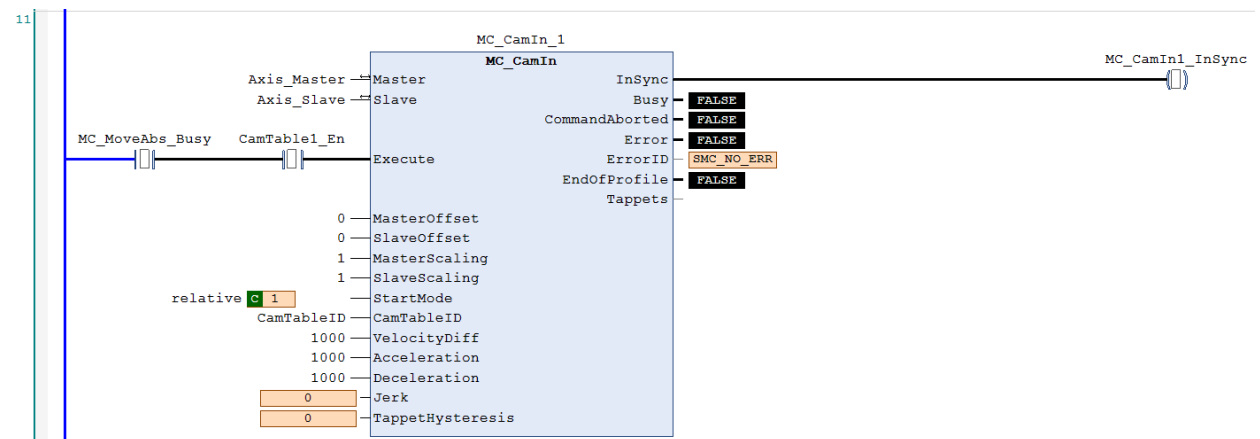
主轴的原点回归动作完成后，执行绝对寻址指令 (MC_MoveAbsolute)。



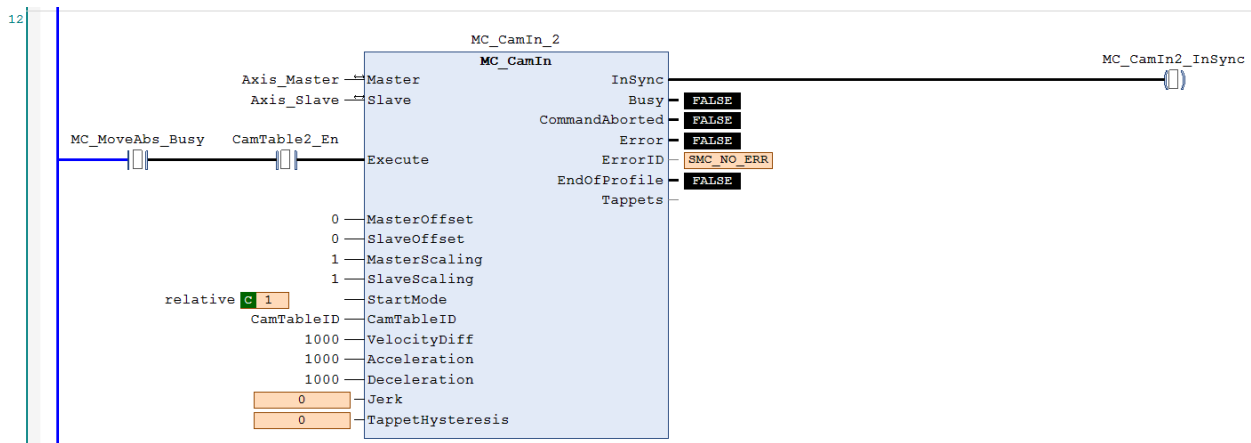
主轴的指令位置为 3000 以下时，使用 CamTable1，CamTable1_En=True、CamTable2_En=False，当大于 3000 时，使用 CamTable2，CamTable1_En=False、CamTable2_En=True，搭配指令 MC_CamTableSelect 设定对应凸轮表。



当主轴进行绝对值定位且 CamTable1_En=True 时，执行凸轮表 CamTable1。



当主轴进行绝对值定位且 CamTable2_En=True 时，执行凸轮表 CamTable2。



ST 程序：

//设定接点 StartFlag 为 True 后会分别确认主从轴的通讯是否有正常的执行。

```

IF StartFlag = TRUE THEN
  IF Axis_Master.bCommunication = TRUE THEN
    MC_Power0_Enable := TRUE;
    MC_Power0_RegulatorOn := TRUE;
    MC_Power0_DriveStart := TRUE;
  END_IF

  IF Axis_Slave.bCommunication = TRUE THEN
    MC_Power1_Enable := TRUE;
    MC_Power1_RegulatorOn := TRUE;
    MC_Power1_DriveStart := TRUE;
  END_IF
END_IF
    
```

//若在正常的情况下，会再分别对主从轴设定为 Servo On 状态。

```

MC_Power_0(
  Axis:= Axis_Master,
  Enable:= MC_Power0_Enable,
  bRegulatorOn:= MC_Power0_RegulatorOn,
  bDriveStart:= MC_Power0_DriveStart,
  Status=> MC_Power0_Status,
  bRegulatorRealState=> ,
  bDriveStartRealState=> ,
  Busy=> ,
  Error=> ,
  ErrorID=> );
    
```

```

MC_Power_1(
  Axis:= Axis_Slave,
  Enable:= MC_Power1_Enable,
  bRegulatorOn:= MC_Power1_RegulatorOn,
  bDriveStart:= MC_Power1_DriveStart,
  Status=> MC_Power1_Status,
  bRegulatorRealState=> ,
    
```

```
bDriveStartRealState=> ,  
Busy=> ,  
Error=> ,  
ErrorID=> );
```

//主从轴为 Servo On 状态且处于原点未确定状态，先执行一次原点回归动作。

```
IF MC_Power0_Status = TRUE THEN  
    MC_Home0_Execute := TRUE;  
END_IF
```

```
IF MC_Power1_Status = TRUE THEN  
    MC_Home1_Execute := TRUE;  
END_IF
```

```
MC_Home_0(  
    Axis:= Axis_Master,  
    Execute:= MC_Home0_Execute,  
    Position:= 0,  
    Done=> MC_Home0_Done,  
    Busy=> ,  
    CommandAborted=> ,  
    Error=> ,  
    ErrorID=>);
```

```
MC_Home_1(  
    Axis:= Axis_Slave,  
    Execute:= MC_Home1_Execute,  
    Position:= 0,  
    Done=> MC_Home1_Done,  
    Busy=> ,  
    CommandAborted=> ,  
    Error=> ,  
    ErrorID=> );
```

//主轴的原点回归动作完成后，执行绝对寻址指令 (MC_MoveAbsolute)。

```
MC_MoveAbsolute(  
    Axis:= Axis_Master,  
    Execute:= MC_Home1_Done,  
    Position:= 6000,  
    Velocity:= 500,  
    Acceleration:= 1000,  
    Deceleration:= 1000,  
    Jerk:= ,  
    Direction:= positive,  
    Done=> MC_MoveAbs_Done,  
    Busy=> MC_MoveAbs_Busy,  
    CommandAborted=> ,  
    Error=> ,  
    ErrorID=> );
```

//主轴的指令位置为 3000 以下时，使用 CamTable1，CamTable1_En=True、CamTable2_En=False，
//当大于 3000 时，使用 CamTable2，CamTable1_En=False、CamTable2_En=True，搭配指令

//MC_CamTableSelect 设定对应凸轮表。

```
IF Axis_Master.fSetPosition > 3000 THEN
```

```
    CamTableSelect := CamTable2;
```

```
    CamTable1_En := FALSE;
```

```
    CamTable2_En := TRUE;
```

```
ELSE
```

```
    CamTableSelect := CamTable1;
```

```
    CamTable1_En := TRUE;
```

```
    CamTable2_En := FALSE;
```

```
END_IF
```

```
IF (CamTable1_En = TRUE) OR (CamTable2_En = TRUE) THEN
```

```
    CamTable_En := TRUE;
```

```
END_IF
```

```
MC_CamTableSelect(
```

```
    Master:= Axis_Master,
```

```
    Slave:= Axis_Slave,
```

```
    CamTable:= CamTableSelect,
```

```
    Execute:= CamTable_En,
```

```
    Periodic:= TRUE,
```

```
    MasterAbsolute:= FALSE,
```

```
    SlaveAbsolute:= FALSE,
```

```
    Done=> MC_CamTableSelect_Done,
```

```
    Busy=> ,
```

```
    Error=> ,
```

```
    ErrorID=> ,
```

```
    CamTableID=> CamTableID);
```

//当主轴进行绝对值定位且 CamTable1_En=True 时，执行凸轮表 CamTable1

```
IF (MC_MoveAbs_Busy = TRUE) AND (CamTable1_En = TRUE) THEN
```

```
    MC_CamIn_1(
```

```
        Master:= Axis_Master,
```

```
        Slave:= Axis_Slave,
```

```
        Execute:= TRUE,
```

```
        MasterOffset:= 0,
```

```
        SlaveOffset:= 0,
```

```
        MasterScaling:= 1,
```

```
        SlaveScaling:= 1,
```

```
        StartMode:= relative,
```

```
        CamTableID:= CamTableID,
```

```
        VelocityDiff:= 1000,
```

```
        Acceleration:= 1000,
```

```
        Deceleration:= 1000,
```

```
        Jerk:= ,
```

```
        TappetHysteresis:= ,
```

```
        InSync=> MC_CamIn1_Insync,
```

```
        Busy=> ,
```

```
        CommandAborted=> ,
```

```
        Error=> ,
```

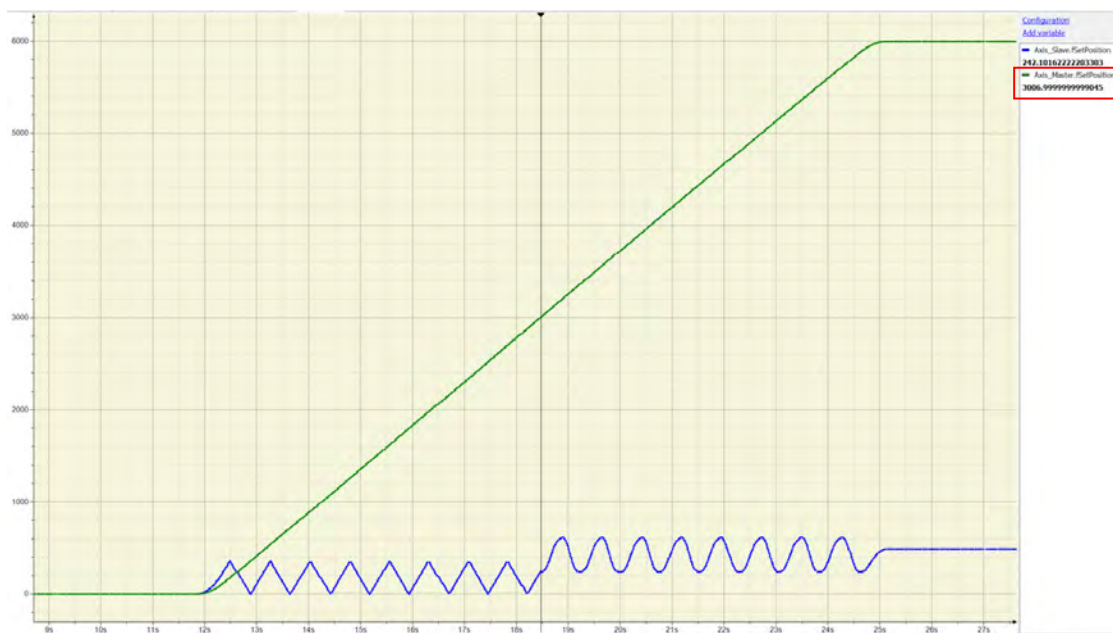
```

ErrorID=> ,
EndOfProfile=> ,
Tappets=> );
END_IF

//当主轴进行绝对值定位且 CamTable2_En=True 时，执行凸轮表 CamTable2
IF (MC_MoveAbs_Busy = TRUE) AND (CamTable2_En = TRUE) THEN
  MC_CamIn_2(
    Master:= Axis_Master,
    Slave:= Axis_Slave,
    Execute:= TRUE,
    MasterOffset:= 0,
    SlaveOffset:= 0,
    MasterScaling:= 1,
    SlaveScaling:= 1,
    StartMode:= relative,
    CamTableID:= CamTableID,
    VelocityDiff:= 1000,
    Acceleration:= 1000,
    Deceleration:= 1000,
    Jerk:= ,
    TappetHysteresis:= ,
    InSync=> MC_CamIn2_Insync,
    Busy=> ,
    CommandAborted=> ,
    Error=> ,
    ErrorID=> ,
    EndOfProfile=> ,
    Tappets=> );
END_IF

```

依照以上设定实现两个不同凸轮表切换，当主轴大于 3000 时进行切换。



7.8.2.8 凸轮动作中执行主轴的相位补偿

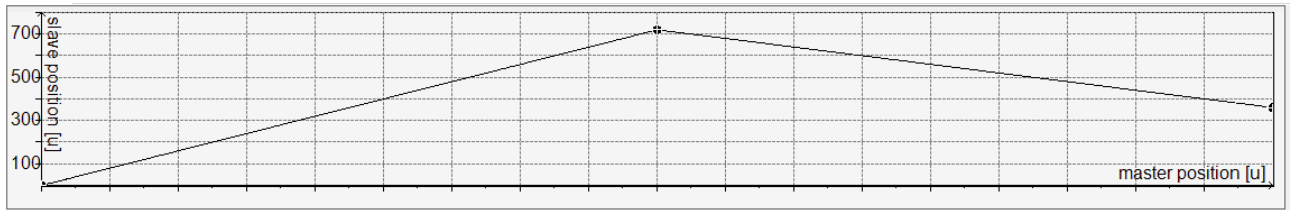
在此范例中，原先凸轮动作中的从轴在执行 MC_Phasing 被打断后与控制中的主轴同步进行动作，当 PhasingActive 为 True 时，执行 MC_Phasing 指令补偿主轴的相位，从轴会与补偿后的相位同步。

此范例提供梯形图与 ST 语法两种。

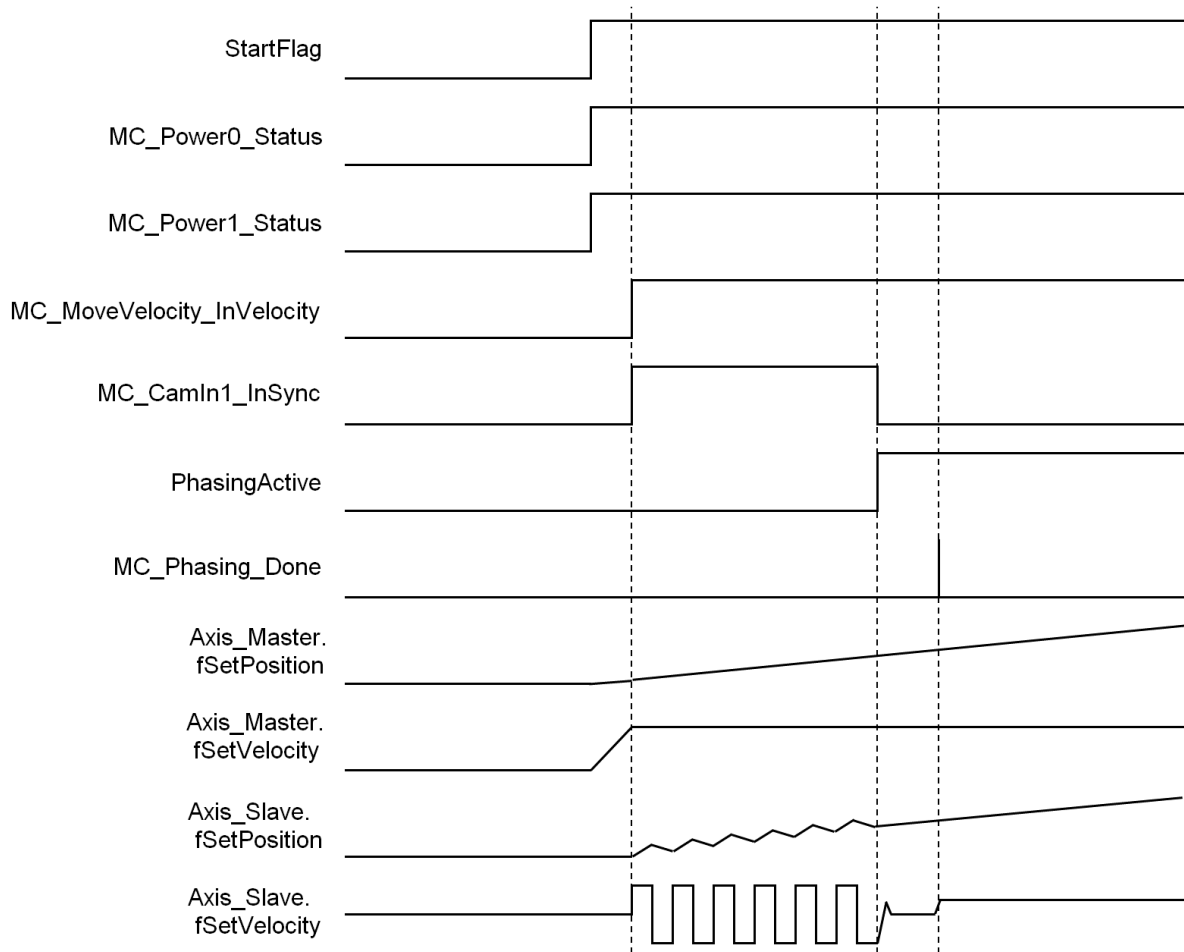
程序中使用的主要变量：

名称	变量类型	初始值	注释
Axis_Master	AXIS_REF_VIRTUAL_SM3	-	主轴的相关轴变量
Axis_Slave	AXIS_REF_VIRTUAL_SM3	-	从轴的相关轴变量
CamTable	MC_CAM_REF	-	凸轮表的相关变量
StartFlag	BOOL	FALSE	若此变量为 TRUE 且与轴通讯正常，则启动 Servo On，并开始执行后续动作。
MC_Power0_Status	BOOL	FALSE	主轴 MC_Power 输出 Status 的变量 Servo On 时为 TRUE
MC_Power1_Status	BOOL	FALSE	从轴 MC_Power 输出 Status 的变量 Servo On 时为 TRUE
MC_Home0_Done	BOOL	FALSE	主轴 MC_Home 输出 Done 的变量 当执行原点回归完成后为 TRUE
MC_Home1_Done	BOOL	FALSE	从轴 MC_Home 输出 Done 的变量 当执行原点回归完成后为 TRUE
MC_MoveVelocity_Velocity	LREAL	500	控制主轴做均速运动的目标速度
MC_MoveVelocity_InVelocity	BOOL	FALSE	主轴 MC_MoveVelocity 输出 InVelocity 的变量 当指定速度达到目标速度时为 TRUE
CamTableID	MC_CAM_ID	-	此为描述凸轮表的内部数据结构，由 MC_CamTableSelect 生成，并作为 MC_CamIn 的输入
MC_CamIn1_InSync	BOOL	FALSE	凸轮表的 MC_CamIn 输出 InSync 的变量 当主从轴凸轮同步时为 TRUE
PhasingActive	BOOL	FALSE	若此变量为 TRUE 且 Cam 已 InSync，则开始执行 MC_Phasing
MC_Phasing_PhaseShift	LREAL	500	指定主轴与从轴的相对相位补偿量
MC_Phasing_Velocity	LREAL	300	指定主轴与从轴进行相位补偿时的相对速度
MC_Phasing_Done	BOOL	FALSE	功能块 MC_Phasing 输出 Done 的变量 当相位补偿完成时为 TRUE

CamTable :



时序图 :

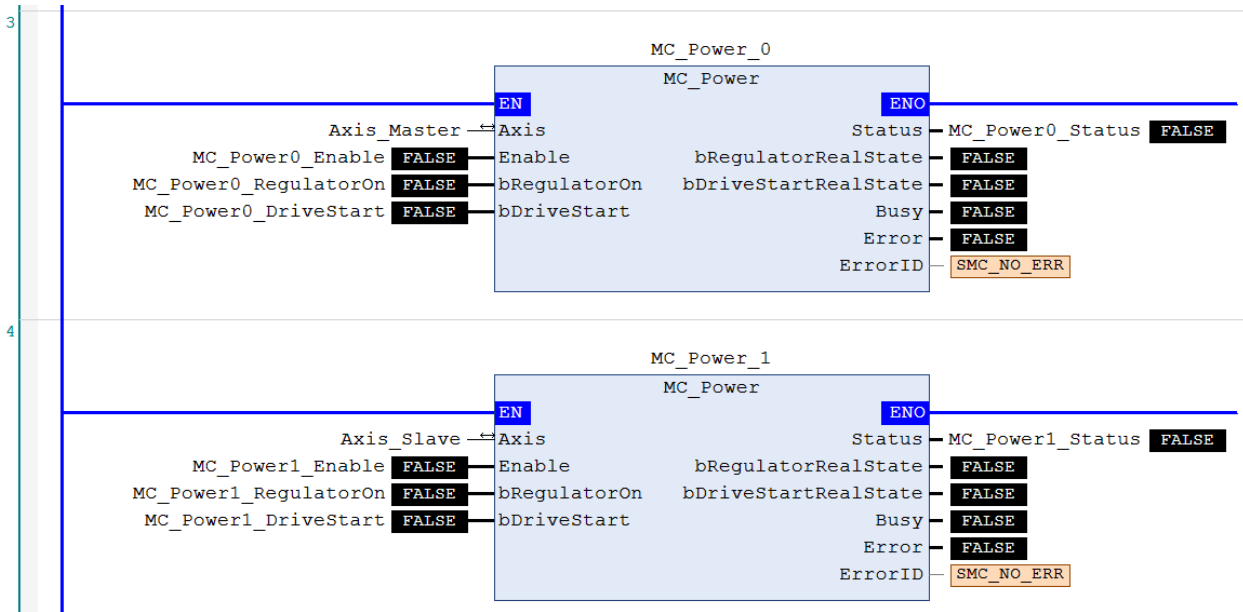


梯形图程序 :

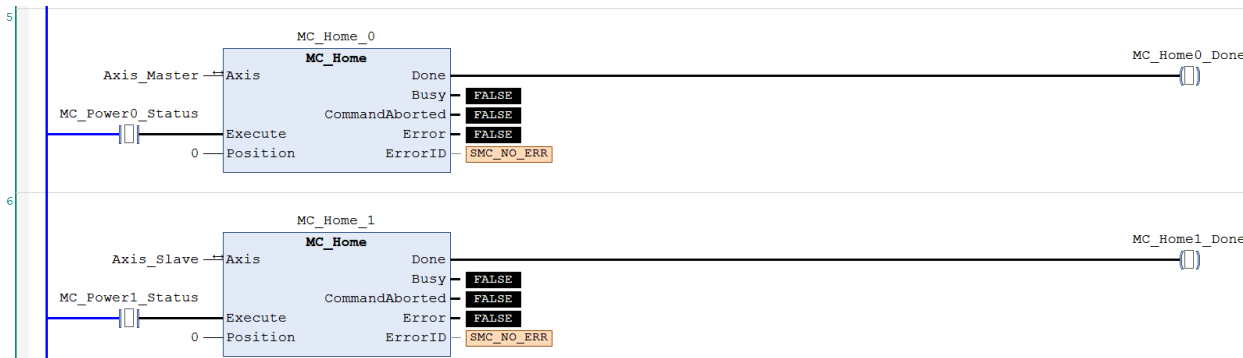
设定接点 StartFlag 为 True 后会分别确认主从轴的通讯是否有正常的执行。



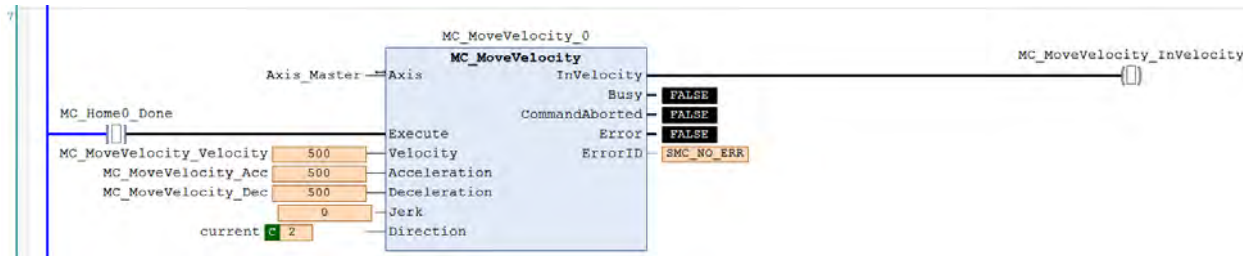
若在正常的情况下，会再分别对主从轴设定为 Servo On 状态。



主从轴为 Servo On 状态且处于原点未确定状态，先执行一次原点回归动作。

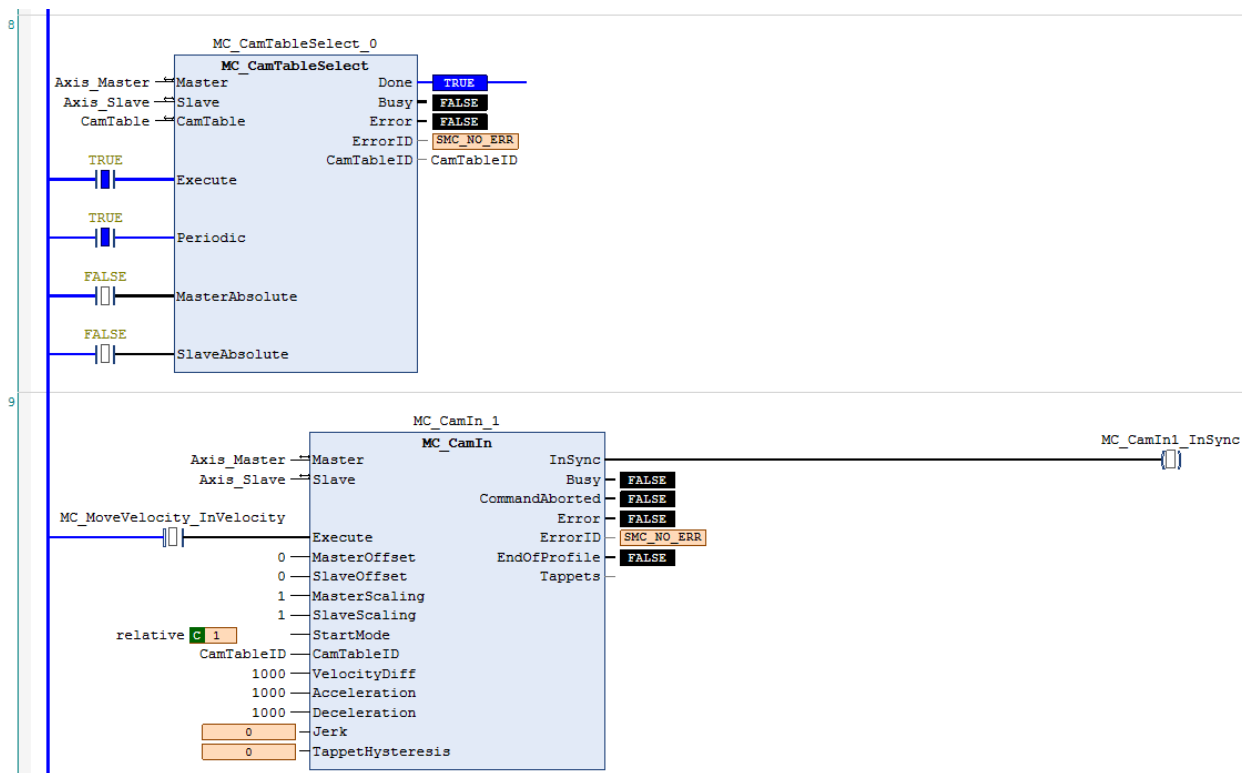


主轴的原点回归动作完成后，执行速度控制指令 (MC_MoveVelocity)。

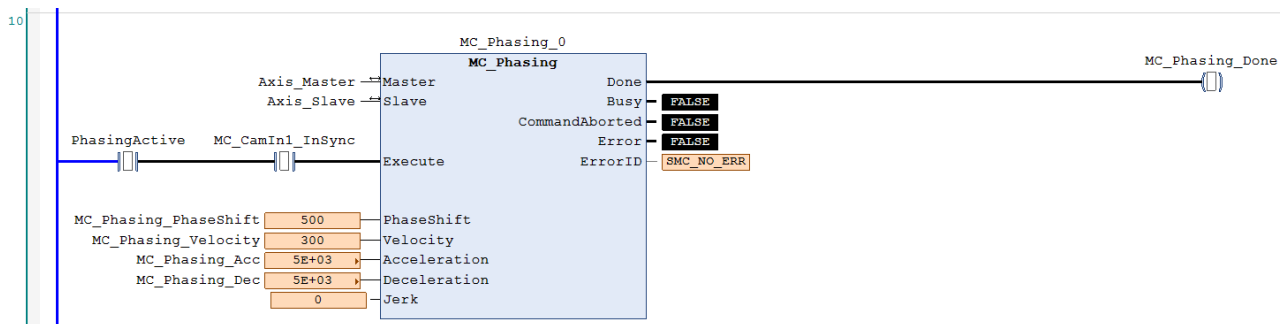


7

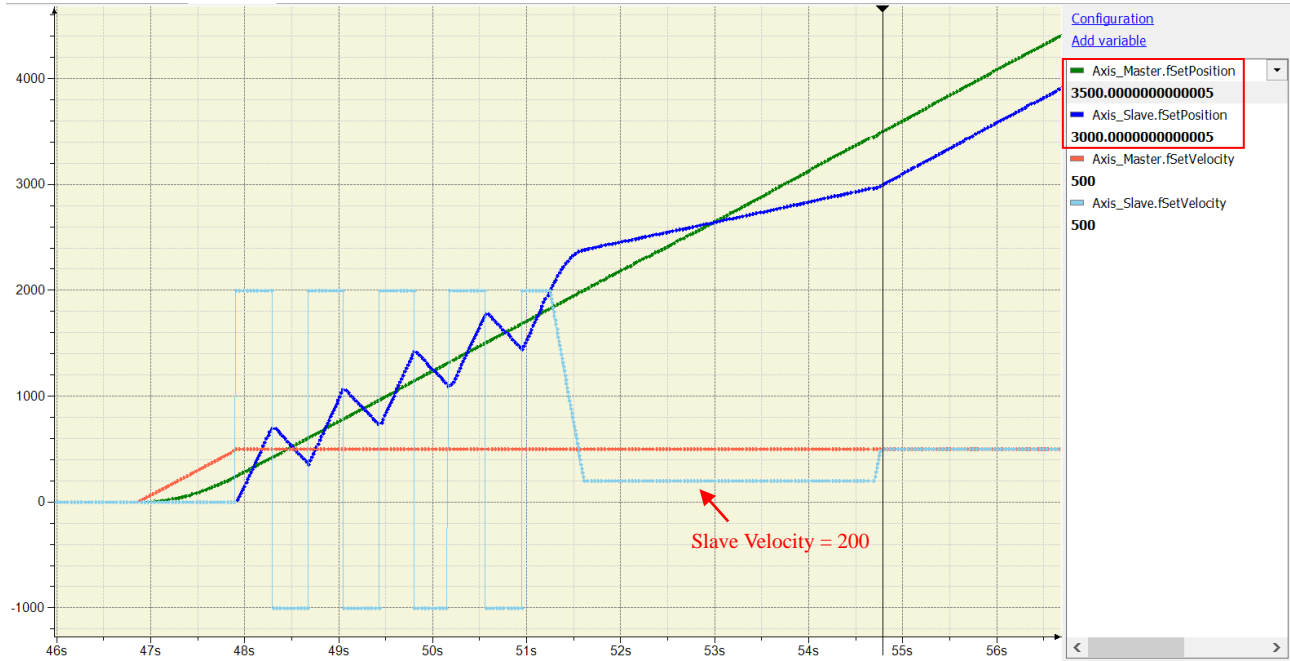
主轴到达目标速度后，搭配指令 MC_CamTableSelect 设定对应凸轮表，执行凸轮动作。



若 PhasingActive 为 True 且凸轮动作同步中，则开始进行主轴与从轴的相位补偿，此时原先凸轮主从轴关系会被打断，从轴依照 MC_Phasing 的设定与主轴进行相位同步。



依照以上设定实现凸轮动作时进行主轴的相位补偿，从轴会与补偿后的相位同步，主轴与从轴的相位差固定。同步后主轴位置与从轴位置的相位差 PhaseShift 为 500 (以下图 Cursor 时间点为例 3500-3000)，进行补偿时从轴的速度为 200 (主轴速度 500- Velocity 300)。



ST 程序：

//设定接点 StartFlag 为 True 后会分别确认主从轴的通讯是否有正常的执行。

```

IF StartFlag = TRUE THEN
    IF Axis_Master.bCommunication = TRUE THEN
        MC_Power0_Enable := TRUE;
        MC_Power0_RegulatorOn := TRUE;
        MC_Power0_DriveStart := TRUE;
    END_IF

    IF Axis_Slave.bCommunication = TRUE THEN
        MC_Power1_Enable := TRUE;
        MC_Power1_RegulatorOn := TRUE;
        MC_Power1_DriveStart := TRUE;
    END_IF
END_IF
    
```

//若在正常的情况下，会再分别对主从轴设定为 Servo On 状态。

```

MC_Power_0 (
    Axis:= Axis_Master,
    Enable:= MC_Power0_Enable,
    bRegulatorOn:= MC_Power0_RegulatorOn,
    bDriveStart:= MC_Power0_DriveStart,
    Status=> MC_Power0_Status,
    bRegulatorRealState=> ,
    bDriveStartRealState=> ,
    Busy=> ,
    )
    
```

```
Error=> ,
ErrorID=> );
```

```
MC_Power_1(
  Axis:= Axis_Slave,
  Enable:= MC_Power1_Enable,
  bRegulatorOn:= MC_Power1_RegulatorOn,
  bDriveStart:= MC_Power1_DriveStart,
  Status=> MC_Power1_Status,
  bRegulatorRealState=> ,
  bDriveStartRealState=> ,
  Busy=> ,
  Error=> ,
  ErrorID=> );
```

主从轴为 Servo On 状态且处于原点未确定状态，先执行一次原点回归动作。

```
IF MC_Power0_Status = TRUE THEN
  MC_Home0_Execute := TRUE;
END_IF
```

```
IF MC_Power1_Status = TRUE THEN
  MC_Home1_Execute := TRUE;
END_IF
```

```
MC_Home_0(
  Axis:= Axis_Master,
  Execute:= MC_Home0_Execute,
  Position:= 0,
  Done=> MC_Home0_Done,
  Busy=> ,
  CommandAborted=> ,
  Error=> ,
  ErrorID=> );
```

```
MC_Home_1(
  Axis:= Axis_Slave,
  Execute:= MC_Home1_Execute,
  Position:= 0,
  Done=> MC_Home1_Done,
  Busy=> ,
  CommandAborted=> ,
  Error=> ,
  ErrorID=> );
```

主轴的原点回归动作完成后，执行速度控制指令 (MC_MoveVelocity)。

```
MC_MoveVelocity(
  Axis:= Axis_Master,
  Execute:= MC_Home0_Done,
  Velocity:= MC_MoveVelocity_Velocity,
  Acceleration:= MC_MoveVelocity_Acc,
  Deceleration:= MC_MoveVelocity_Dec,
  Jerk:= ,
  Direction:= current,
```

```

InVelocity=> MC_MoveVelocity_InVelocity,
Busy=> ,
CommandAborted=> ,
Error=> ,
ErrorID=> );

```

主轴到达目标速度后，搭配指令 MC_CamTableSelect 设定对应凸轮表，执行凸轮动作。

```

MC_CamTableSelect(
  Master:= Axis_Master,
  Slave:= Axis_Slave,
  CamTable:= CamTable,
  Execute:= TRUE,
  Periodic:= TRUE,
  MasterAbsolute:= FALSE,
  SlaveAbsolute:= FALSE,
  Done=> MC_CamTableSelect_Done,
  Busy=> ,
  Error=> ,
  ErrorID=> ,
  CamTableID=> CamTableID);

```

```

IF MC_MoveVelocity_InVelocity = TRUE THEN

```

```

  MC_CamIn_1(
    Master:= Axis_Master,
    Slave:= Axis_Slave,
    Execute:= TRUE,
    MasterOffset:= 0,
    SlaveOffset:= 0,
    MasterScaling:= 1,
    SlaveScaling:= 1,
    StartMode:= relative,
    CamTableID:= CamTableID,
    VelocityDiff:= 1000,
    Acceleration:= 1000,
    Deceleration:= 1000,
    Jerk:= ,
    TappetHysteresis:= ,
    InSync=> MC_CamIn1_Insync,
    Busy=> ,
    CommandAborted=> ,
    Error=> ,
    ErrorID=> ,
    EndOfProfile=> ,
    Tappets=> );

```

```

END_IF

```

若 PhasingActive 为 True 且凸轮动作同步中，则开始进行主轴与从轴的相位补偿，此时原先凸轮主从轴关系会被打断，从轴依照 MC_Phasing 的设定与主轴进行相位同步。

```

IF (PhasingActive = TRUE) AND (MC_CamIn1_Insync = TRUE) THEN
  MC_Phasing_Execute := TRUE;
END_IF

```

```

MC_Phasing(
  Master:= Axis_Master,
  Slave:= Axis_Slave,
  Execute:= MC_Phasing_Execute,
  PhaseShift:= MC_Phasing_PhaseShift,
  Velocity:= MC_Phasing_Velocity,
  Acceleration:= MC_Phasing_Acc,
  Deceleration:= MC_Phasing_Dec,
  Jerk:= ,
  Done=> MC_Phasing_Done,
  Busy=> ,
  CommandAborted=> ,
  Error=> ,
  ErrorID=> );

```

7.8.2.9 移动控制过程中变更当前位置

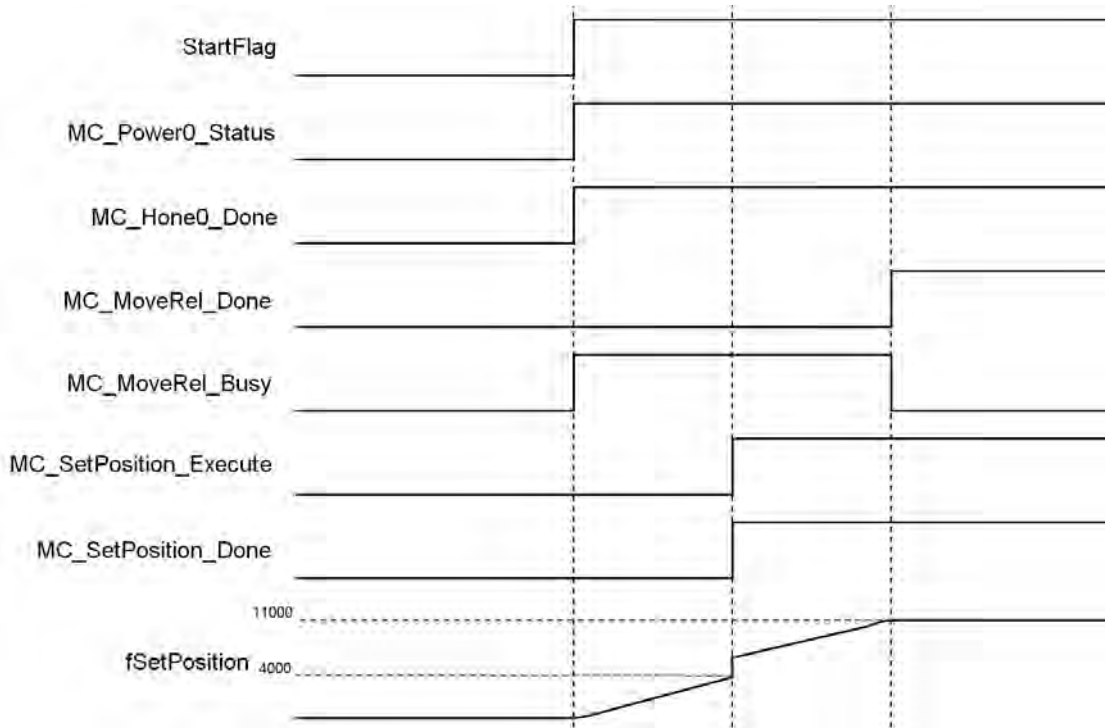
在移动控制中要改变坐标系统中轴的当前位置为指定的目标位置·将反馈当前位置变更为指定的目标位置。在此范例中·说明 MC_MoveRelative 和 MC_SetPosition 的功能块指令的交互影响。

此范例提供梯形图与 ST 语法两种。

程序中使用的主要变量：

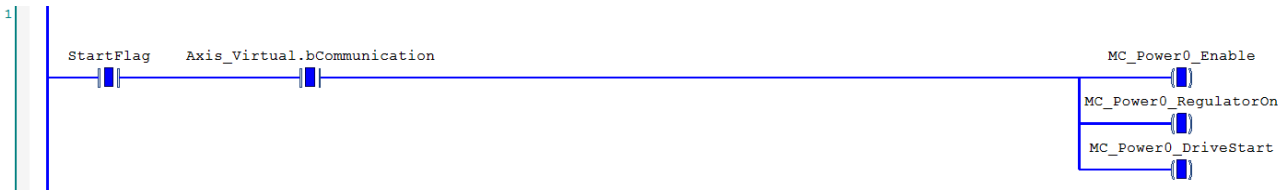
名称	变量类型	初始值	注释
Axis_Virtual	AXIS_REF_VIRTUAL_SM3	-	轴的相关变量
StartFlag	BOOL	FALSE	若此变量为 TRUE 且与轴通讯正常·则启动 Servo On·并开始执行后续动作
MC_Power0_Status	BOOL	FALSE	轴 MC_Power 输出 Status 的变量 Servo On 时为 TRUE
MC_Home0_Done	BOOL	FALSE	轴 MC_Home 输出 Done 的变量 当执行原点回归完成后为 TRUE
MC_MoveRel_Distance	LREAL	8000	MC_MoveRelative 功能块的目标相对位置
MC_MoveRel_Done	BOOL	FALSE	轴 MC_MoveRelative 输出 Done 的变量 当相对定位完成时为 TRUE
MC_MoveRel_Busy	BOOL	FALSE	轴 MC_MoveRelative 输出 Busy 的变量 当指令被触发执行时为 TRUE
MC_SetPosition_Execute	BOOL	FALSE	若此变量为 TRUE·则开始执行 MC_SetPosition
MC_SetPosition_Position	LREAL	3000	MC_SetPosition 功能块设定改变的绝对位置或相对距离
MC_SetPosition_Mode	BOOL	TRUE	MC_SetPosition 功能块是指定为绝对位置或相对位置
MC_SetPosition_Done	BOOL	FALSE	功能块 MC_SetPosition 输出 Done 的变量 当修改位置完成时为 TRUE

时序图：

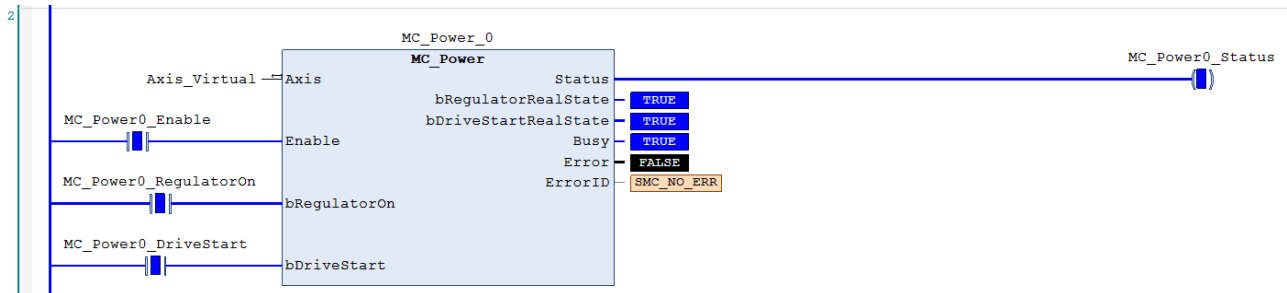


梯形图程序：

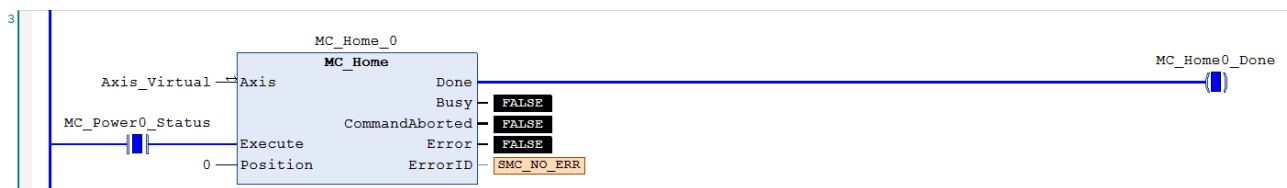
设定接点 StartFlag 为 True 后会确认轴的通讯是否有正常的执行。



若在正常的情况下，再把此轴设定为 Servo On 状态。

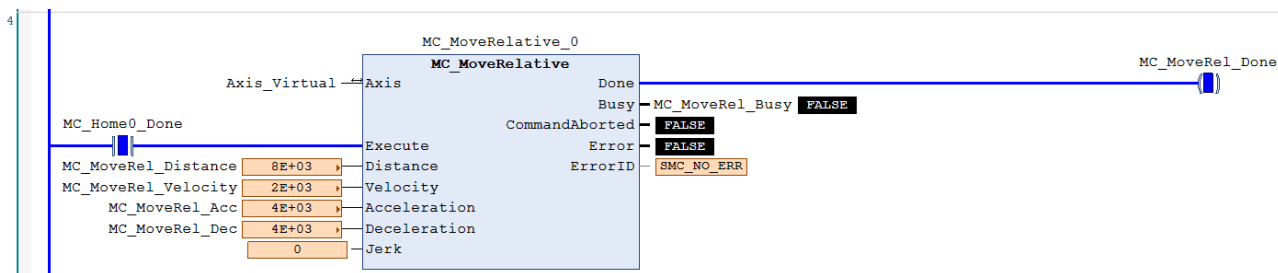


轴为 Servo On 状态且处于原点未确定状态，先执行一次原点回归动作。

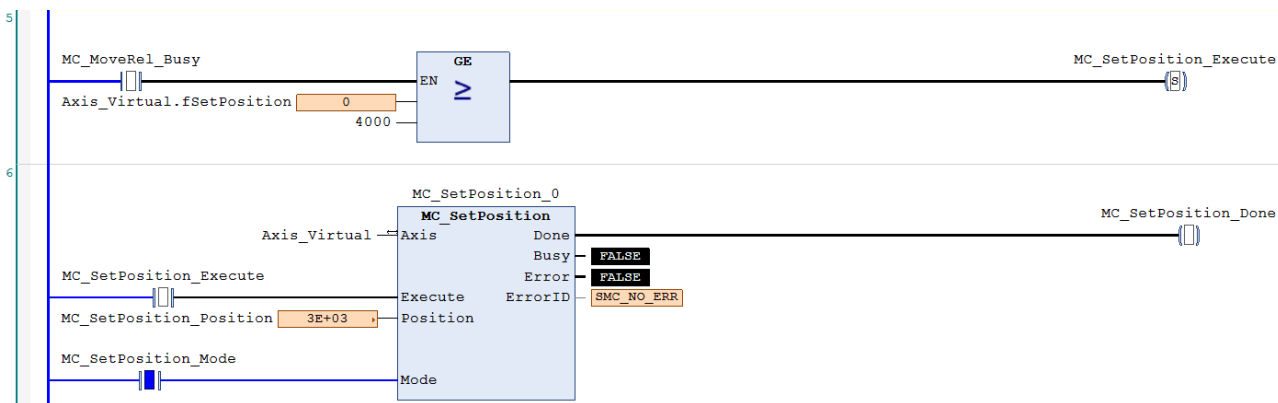


当轴的原点回归动作完成后，执行相对移动控制指令 MC_MoveRelative。

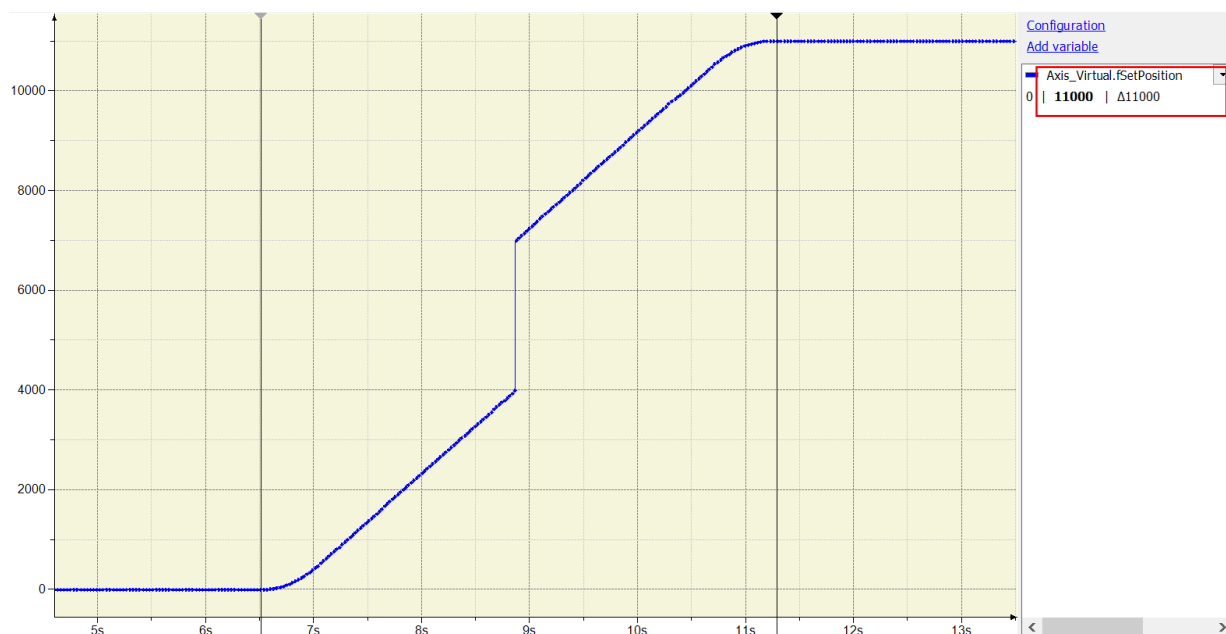
相对位移目标位置 = 8000。



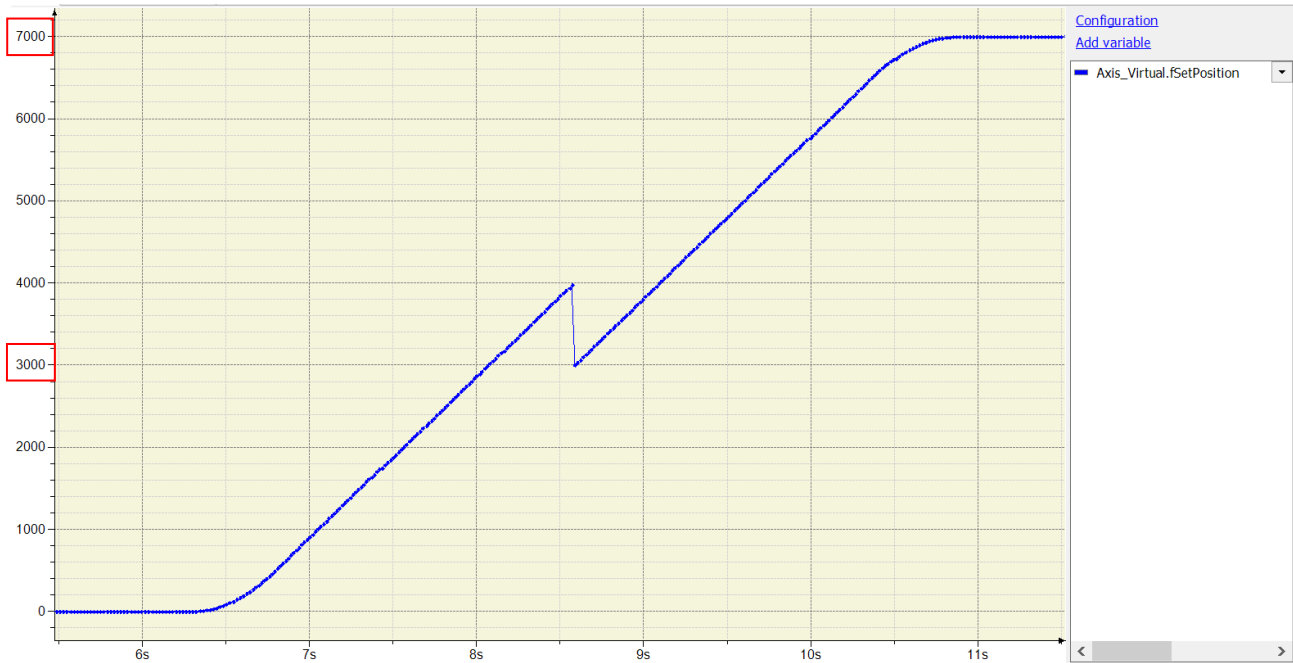
当轴的当前位置经过 4000 时，执行 MC_SetPosition (Mode = Relative · Distance = 3000) 依指定的目标位置变更为当前位置。



依照以上设定实现移动控制中改变坐标系统中轴的当前位置至相对目标位置，最后位置到达变更后的 11000 ($11000 = 4000 + 3000 + (8000 - 4000)$)，执行变更命令位置后不会影响到 MC_MoveRelative 操作中的实体运动机构位移量，移动距离与原先设定的 8000 ($8000 = (4000 - 0) + (11000 - 7000)$) 相同。



下图与上图的差异是 MC_SetPosition 的 Mode 更改为 Absolute (Position = 3000)，改变坐标系统中轴的当前位置至绝对目标位置，最后位置到达变更后的 7000 ($7000 = 3000 + (8000 - 4000)$)，执行变更命令位置后不会影响到 MC_MoveRelative 操作中的实体运动机构位移量，移动距离与原先设定的 8000 ($8000 = (4000 - 0) + (7000 - 3000)$) 相同。



ST 程序：

设定接点 StartFlag 为 True 后会确认轴的通讯是否有正常的执行。

```

IF StartFlag = TRUE THEN
    IF Axis_Virtual.bCommunication = TRUE THEN
        MC_Power0_Enable := TRUE;
        MC_Power0_RegulatorOn := TRUE;
        MC_Power0_DriveStart := TRUE;
    END_IF
END_IF
    
```

//若在正常的情况下，再把此轴设定为 Servo On 状态。

```

MC_Power_0 (
    Axis:= Axis_Virtual,
    Enable:= MC_Power0_Enable,
    bRegulatorOn:= MC_Power0_RegulatorOn,
    bDriveStart:= MC_Power0_DriveStart,
    Status=> MC_Power0_Status,
    bRegulatorRealState=> ,
    bDriveStartRealState=> ,
    Busy=> ,
    Error=> ,
    ErrorID=> );
    
```

轴为 Servo On 状态且处于原点未确定状态，先执行一次原点回归动作。

```
IF MC_Power0_Status = TRUE THEN
    MC_Home0_Execute := TRUE;
END_IF
```

```
MC_Home_0(
    Axis:= Axis_Virtual,
    Execute:= MC_Home0_Execute,
    Position:= 0,
    Done=> MC_Home0_Done,
    Busy=> ,
    CommandAborted=> ,
    Error=> ,
    ErrorID=> );
```

当轴的原点回归动作完成后，执行相对移动控制指令 MC_MoveRelative。

相对位移目标位置 = 8000。

```
MC_MoveRelative(
    Axis:= Axis_Virtual,
    Execute:= MC_Home0_Done,
    Distance:= MC_MoveRel_Distance,
    Velocity:= MC_MoveRel_Velocity,
    Acceleration:= MC_MoveRel_Acc,
    Deceleration:= MC_MoveRel_Dec,
    Jerk:= ,
    Done=> MC_MoveRel_Done,
    Busy=> MC_MoveRel_Busy,
    CommandAborted=> ,
    Error=> ,
    ErrorID=> );
```

当轴的当前位置经过 4000 时，执行 MC_SetPosition (Mode = Relative · Distance = 3000) 依指定的目标位置变更为当前位置。

```
IF (MC_MoveRel_Busy = TRUE) AND (Axis_Virtual.fSetPosition >= 4000) THEN
    MC_SetPosition_Execute := TRUE;
END_IF
```

```
MC_SetPosition(
    Axis:= Axis_Virtual,
    Execute:= MC_SetPosition_Execute,
    Position:= MC_SetPosition_Position,
    Mode:= MC_SetPosition_Mode,
    Done=> MC_SetPosition_Done,
    Busy=> ,
    Error=> ,
    ErrorID=> );
```


7.8.2.10 主从轴齿轮啮合过程中执行位移追加

在此范例中，主从轴在齿轮已啮合的过程中，对指定的从轴执行 MC_MoveSuperImposed 位移追加，从轴最后的目标位置会为齿轮比相对主轴的位移量加上在运动中追加指定的距离。

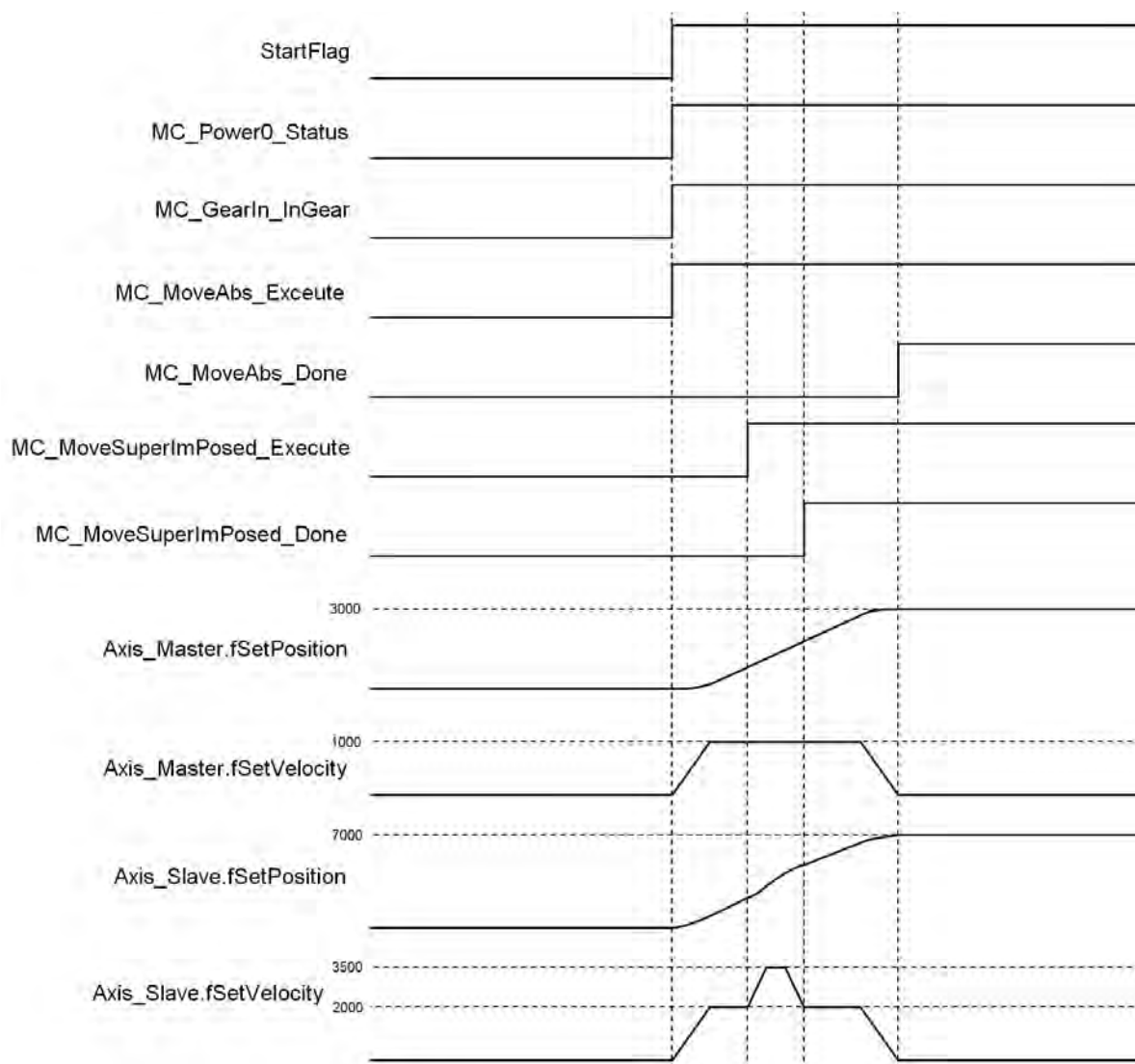
此范例提供梯形图与 ST 语法两种。

程序中使用的主要变量：

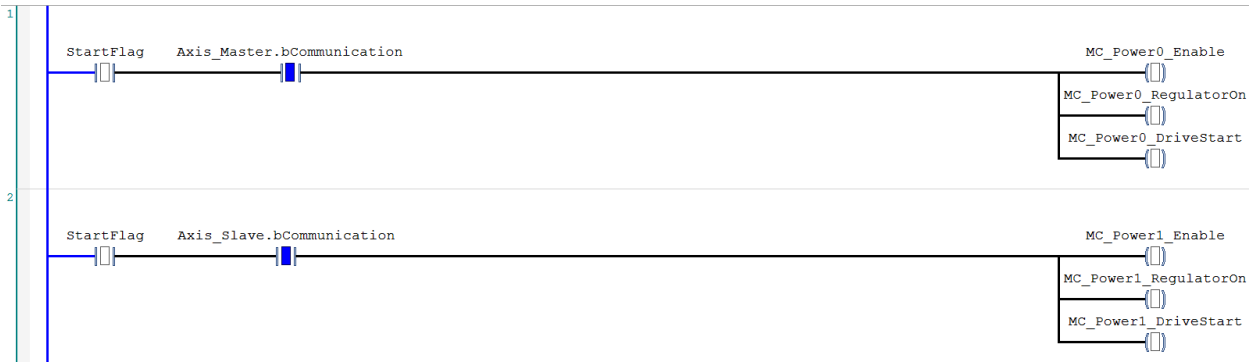
名称	变量类型	初始值	注释
Axis_Master	AXIS_REF_VIRTUAL_SM3	-	主轴的相关轴变量
Axis_Slave	AXIS_REF_VIRTUAL_SM3	-	从轴的相关轴变量
StartFlag	BOOL	FALSE	若此变量为 TRUE 且与轴通讯正常，则启动 Servo On，并开始执行后续动作
MC_Power0_Status	BOOL	FALSE	主轴 MC_Power 输出 Status 的变量 Servo On 时为 TRUE
MC_Power1_Status	BOOL	FALSE	从轴 MC_Power 输出 Status 的变量 Servo On 时为 TRUE
MC_Home0_Done	BOOL	FALSE	主轴 MC_Home 输出 Done 的变量 当执行原点回归完成后为 TRUE
MC_Home1_Done	BOOL	FALSE	从轴 MC_Home 输出 Done 的变量 当执行原点回归完成后为 TRUE
MC_GearIn_InGear	BOOL	FALSE	齿轮的 MC_GearIn 输出 InGear 的变量 当啮合动作完成时为 TRUE
MC_GearIn_RatioNumerator	DINT	2	主从轴间齿轮比之分子
MC_GearIn_RatioDenominator	UDINT	1	主从轴间齿轮比之分母
MC_MoveAbs_Execute	BOOL	FALSE	若此变量为 TRUE，则开始执行 MC_MoveAbsolute
MC_MoveAbs_Position	LREAL	3000	指定主轴的绝对目标位置
MC_MoveAbs_Velocity	LREAL	1000	指定主轴的目标速度
MC_MoveAbs_Done	BOOL	FALSE	主轴 MC_MoveAbsolute 输出 Done 的变量，当绝对寻址完成时为 TRUE
MC_MoveAbs_Busy	BOOL	FALSE	主轴 MC_MoveAbsolute 输出 Busy 的变量，当执行后轴在动作中为 TRUE
MC_MoveSuperImposed_Execute	BOOL	FALSE	若此变量为 TRUE，则开始执行 MC_MoveSuperImposed

名称	变量类型	初始值	注释
MC_MoveSuperImposed_Done	BOOL	FALSE	从轴 MC_MoveSuperImposed 输出 Done 的变量，当追加完成时为 TRUE
MC_MoveSuperImposed_Distance	LREAL	1000	指定从轴的追加位移量
MC_MoveSuperImposed_VelocityDiff	LREAL	1500	指定从轴进行追加时与主轴的相对速度

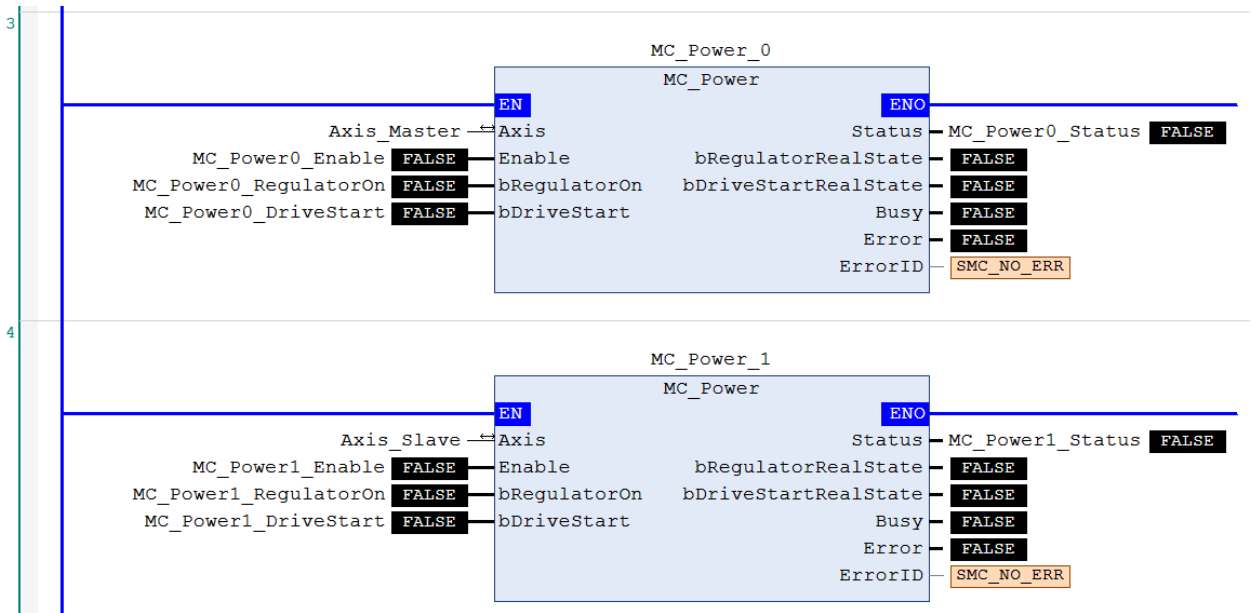
时序图：



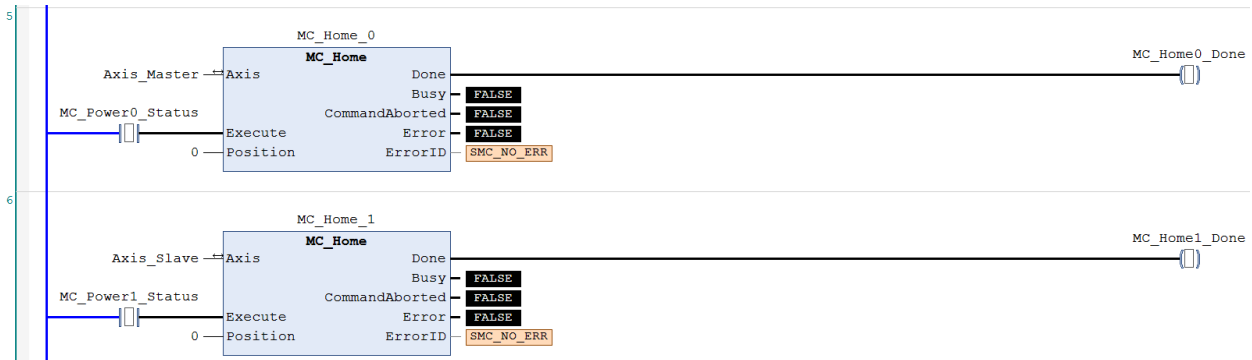
梯形图程序：设定接点 StartFlag 为 True 后会分别确认主从轴的通讯是否有正常的执行。



若在正常的情况下，会再分别对主从轴设定为 Servo On 状态。

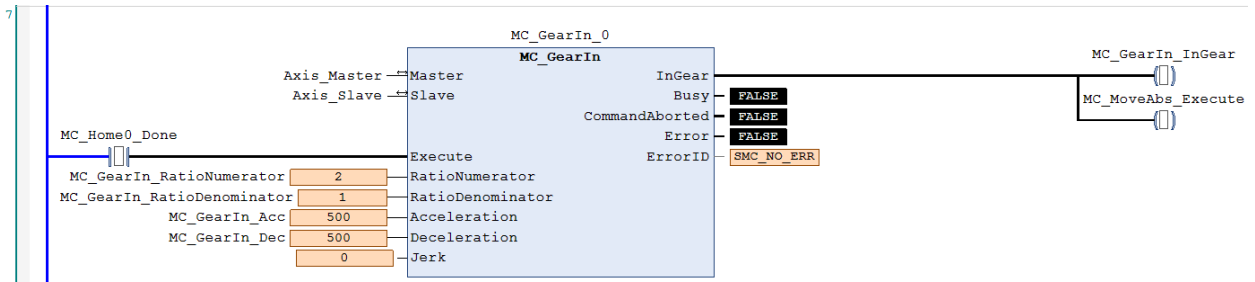


主从轴为 Servo On 状态且处于原点未确定状态，先执行一次原点回归动作。

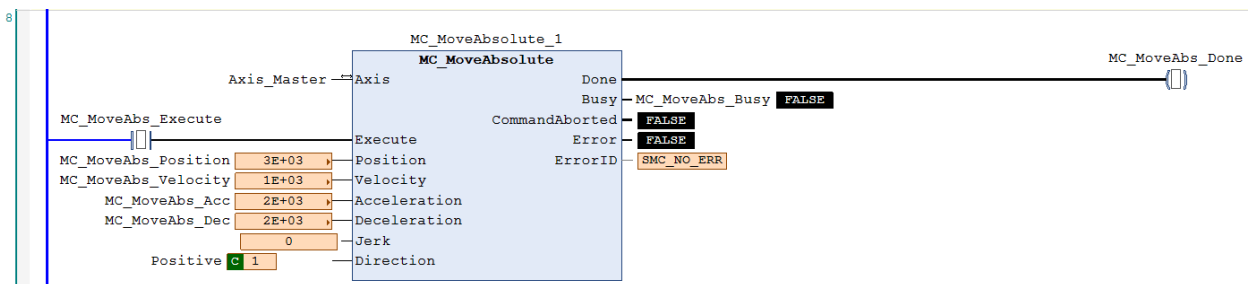


7

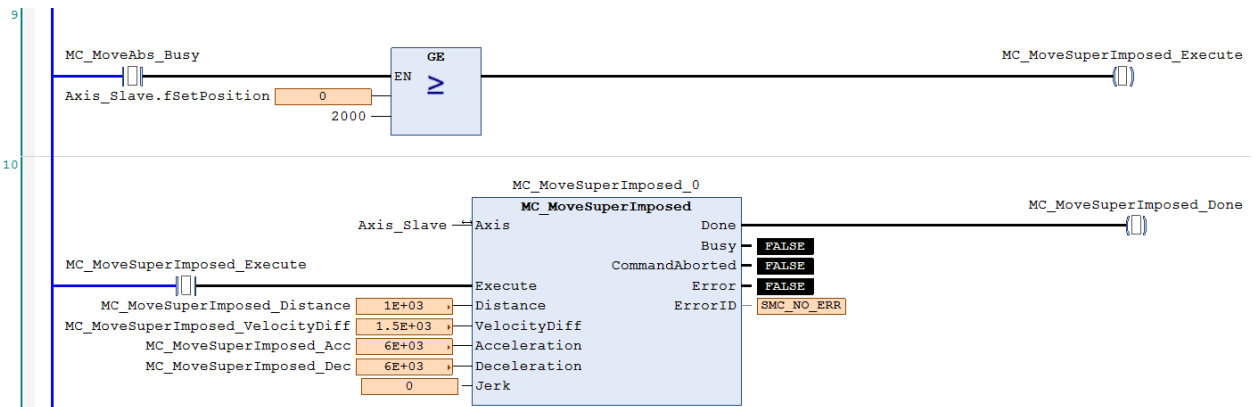
主轴的原点回归动作完成后，执行 MC_GearIn 建立主从轴齿轮同步关系。



当主从轴啮合作 InGear 完成后，对主轴执行绝对目标位置 (MC_MoveAbsolute) 移动。

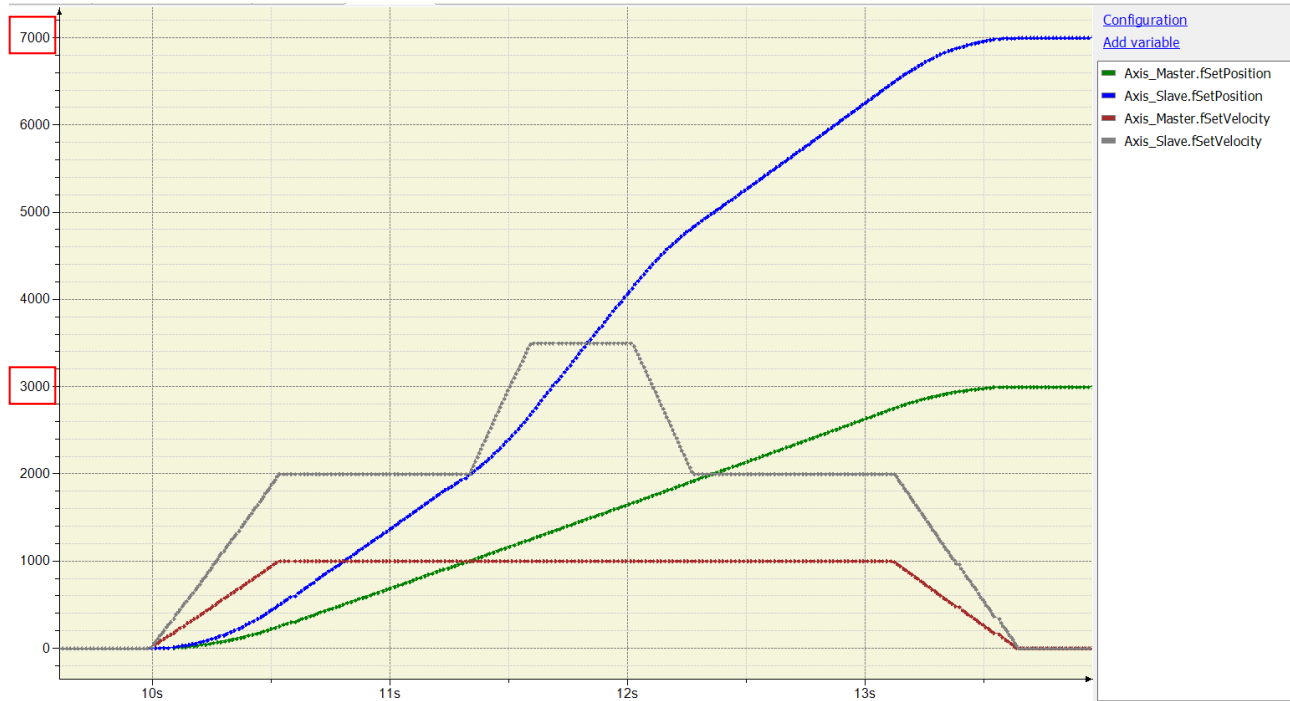


此时从轴会同时以啮合关系移动，当从轴位置移动到设定触发位置 2000 时，执行追加指令 MC_MoveSuperImposed，从轴会以原先齿轮比关系的设定目标位置再追加指定的位移量动作。



依照以上设定实现啮合作时进行对从轴的追加距离，从轴最后的目标位置会为齿轮比相对主轴的位移量加上在运动中追加指定的距离，并以追加时设定的速度动作。

主轴的移动距离为 3000，从轴原先的目标位置为齿轮比 1 : 2 关系计算为 6000，之后对从轴追加的距离为 1000，所以从轴最后的目标位置变更为 7000 (6000 + 1000)，进行啮合时主轴的速度为 1000 从轴的速度为 2000，追加时从轴的速度为 3500 (从轴原先的速度 2000 + 追加速度差 VelocityDiff 1500)。



ST 程序：

设定接点 StartFlag 为 True 后会分别确认主从轴的通讯是否有正常的执行。

```

IF StartFlag = TRUE THEN
  IF Axis_Master.bCommunication = TRUE THEN
    MC_Power0_Enable := TRUE;
    MC_Power0_RegulatorOn := TRUE;
    MC_Power0_DriveStart := TRUE;
  END_IF

  IF Axis_Slave.bCommunication = TRUE THEN
    MC_Power1_Enable := TRUE;
    MC_Power1_RegulatorOn := TRUE;
    MC_Power1_DriveStart := TRUE;
  END_IF
END_IF
    
```

若在正常的情况下，会再分别对主从轴设定为 Servo On 状态。

```

MC_Power_0(
  Axis:= Axis_Master,
  Enable:= MC_Power0_Enable,
  bRegulatorOn:= MC_Power0_RegulatorOn,
  bDriveStart:= MC_Power0_DriveStart,
  Status=> MC_Power0_Status,
  bRegulatorRealState=> ,
  bDriveStartRealState=> ,
  Busy=> ,
  Error=> ,
  ErrorID=> );
    
```

```

MC_Power_1(
  Axis:= Axis_Slave,
  Enable:= MC_Power1_Enable,
  bRegulatorOn:= MC_Power1_RegulatorOn,
  bDriveStart:= MC_Power1_DriveStart,
  Status=> MC_Power1_Status,
  bRegulatorRealState=> ,
  bDriveStartRealState=> ,
  Busy=> ,
  Error=> ,
  ErrorID=> );

```

主从轴为 Servo On 状态且处于原点未确定状态，先执行一次原点回归动作。

```

IF MC_Power0_Status = TRUE THEN
  MC_Home0_Execute := TRUE;
END_IF

```

```

IF MC_Power1_Status = TRUE THEN
  MC_Home1_Execute := TRUE;
END_IF

```

```

MC_Home_0(
  Axis:= Axis_Master,
  Execute:= MC_Home0_Execute,
  Position:= 0,
  Done=> MC_Home0_Done,
  Busy=> ,
  CommandAborted=> ,
  Error=> ,
  ErrorID=> );

```

```

MC_Home_1(
  Axis:= Axis_Slave,
  Execute:= MC_Home1_Execute,
  Position:= 0,
  Done=> MC_Home1_Done,
  Busy=> ,
  CommandAborted=> ,
  Error=> ,
  ErrorID=> );

```

主轴的原点回归动作完成后，执行 MC_GearIn 建立主从轴齿轮同步关系。

```

MC_GearIn(
  Master:= Axis_Master,
  Slave:= Axis_Slave,
  Execute:= MC_Home0_Done,
  RatioNumerator:= MC_GearIn_RatioNumerator,
  RatioDenominator:= MC_GearIn_RatioDenominator,
  Acceleration:= MC_GearIn_Acc,
  Deceleration:= MC_GearIn_Dec,
  Jerk:= ,
  InGear=> MC_GearIn_InGear,
  Busy=> ,

```

```

CommandAborted=> ,
Error=> ,
ErrorID=> );

```

```

IF MC_GearIn_InGear = TRUE THEN
  MC_MoveAbs_Execute := TRUE;
END_IF

```

当主从轴啮合动作 InGear 完成后，对主轴执行绝对目标位置 (MC_MoveAbsolute) 移动。

```

MC_MoveAbsolute(
  Axis:= Axis_Master,
  Execute:= MC_MoveAbs_Execute,
  Position:= MC_MoveAbs_Position,
  Velocity:= MC_MoveAbs_Velocity,
  Acceleration:= MC_MoveAbs_Acc,
  Deceleration:= MC_MoveAbs_Dec,
  Jerk:= ,
  Direction:= Positive,
  Done=> MC_MoveAbs_Done,
  Busy=> MC_MoveAbs_Busy,
  CommandAborted=> ,
  Error=> ,
  ErrorID=> );

```

此时从轴会同时以啮合关系移动，当从轴位置移动到设定触发位置 2000 时，执行追加指令 MC_MoveSuperImposed，从轴会以原先齿轮比关系的设定目标位置再追加指定的位移量动作。

```

IF MC_MoveAbs_Busy = TRUE THEN
  IF Axis_Slave.fSetPosition >= 2000 THEN
    MC_MoveSuperImposed_Execute := TRUE;
  END_IF
END_IF

```

```

7 MC_MoveSuperImposed(
  Axis:= Axis_Slave,
  Execute:= MC_MoveSuperImposed_Execute,
  Distance:= MC_MoveSuperImposed_Distance,
  VelocityDiff:= MC_MoveSuperImposed_VelocityDiff,
  Acceleration:= MC_MoveSuperImposed_Acc,
  Deceleration:= MC_MoveSuperImposed_Dec,
  Jerk:= ,
  Done=> MC_MoveSuperImposed_Done,
  Busy=> ,
  CommandAborted=> ,
  Error=> ,
  ErrorID=> );

```

第8章 OPC UA Server

目录

8.1	OPC UA Server 简介	8-2
8.1.1	建立项目中的 OPC UA 存取设定	8-2
8.2	建立一般联机.....	8-5
8.2.1	使用「UaExpert」Client 建立一般联机	8-5
8.3	建立加密联机.....	8-9
8.3.1	设定用户账号密码.....	8-9
8.3.2	信任凭证设定	8-9
8.3.3	使用「Prosyst OPC UA Client」建立加密联机	8-12
8.3.4	使用「UaExpert」建立加密联机.....	8-16

8.1 OPC UA Server 简介

AX-3 系列控制器内，包含 OPC UA 服务器，通过客户端访问控制器的控制接口去使用这项服务。OPC UA 服务器通过 TCP 联机与 OPC UA 客户端进行通讯。因此，必须就安全性再次单独检查这些连接，以使用加密通讯保护客户端和 OPC UA 用户管理，从而保护 OPC UA 服务器。

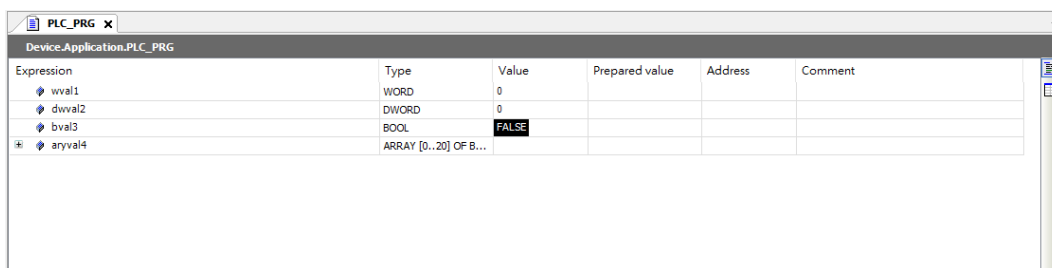
OPC UA Server 支持以下功能：

- 浏览数据类型和变量
- 标准读 / 写
- 值的变更通知（针对已订阅和监控中的项目）
- 依据 OPC UA standard，支持 Basic256SHA256 的加密通信

8.1.1 建立项目中的 OPC UA 存取设定

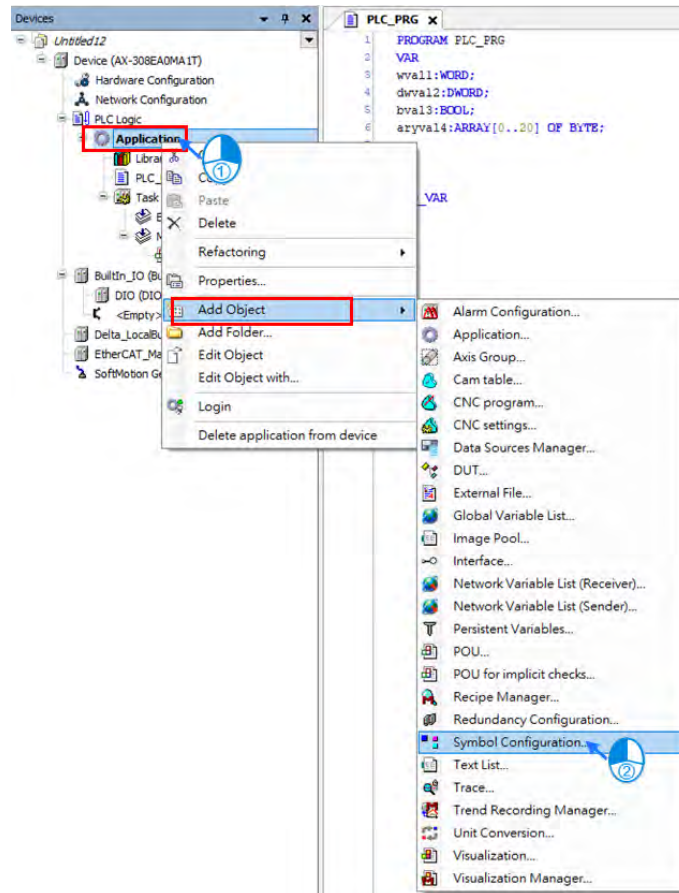
要使用 OPC UA Server 功能，首先须先建立项目中的 OPC UA 存取功能，操作步骤如下：

- (1) 在 DIADesigner-AX 中建立一个新项目。
- (2) 在项目「PLC_PRG」程序中，建立一些不同类型的变量。

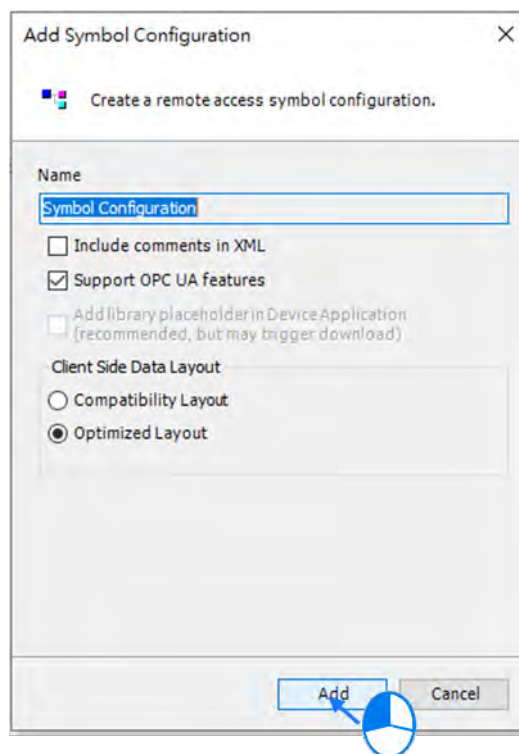


Expression	Type	Value	Prepared value	Address	Comment
wval1	WORD	0			
dwval2	DWORD	0			
bval3	BOOL	FALSE			
aryval4	ARRAY [0..20] OF B...				

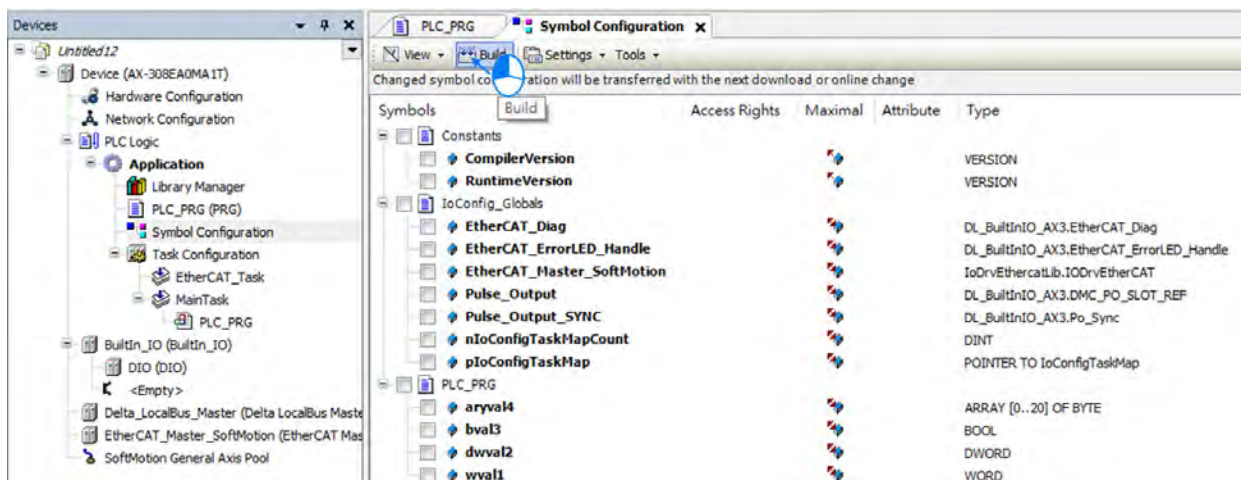
- (3) 从 Application 中，增加一个 Symbol Configuration object：「Application」>「Add Object」>「Symbol Configuration」



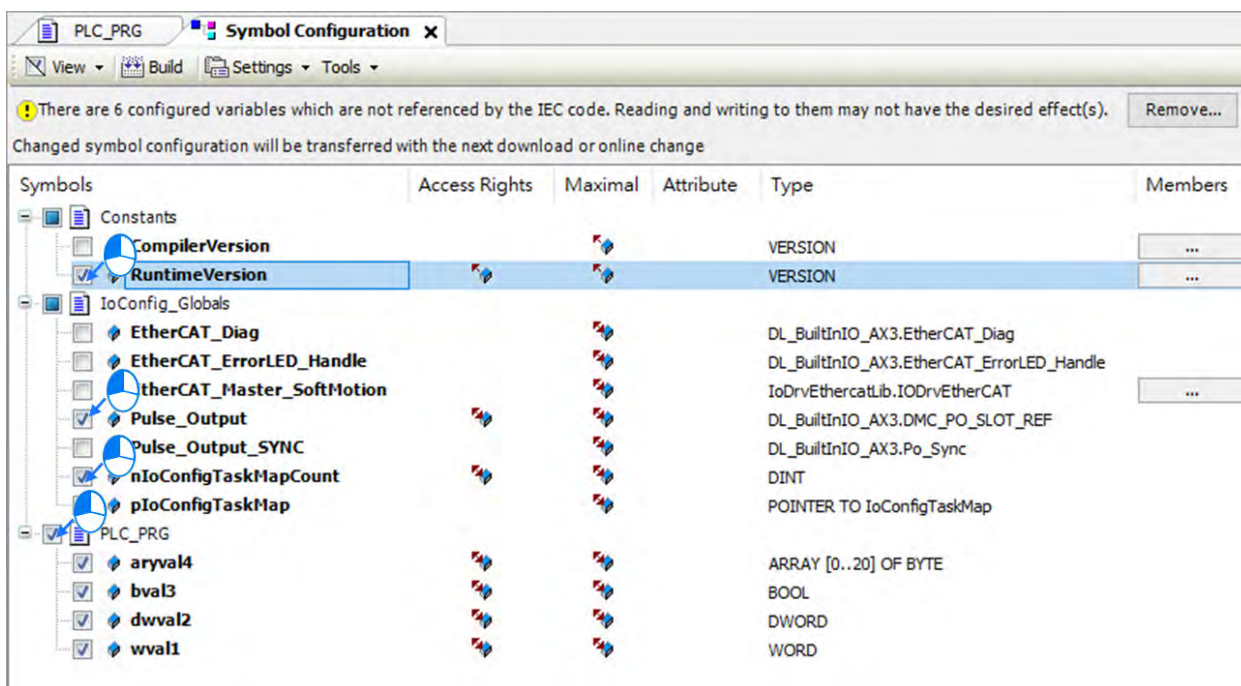
(4) 在「Add Symbol Configuration」的设定中，勾选「Support OPC UA Features」后按「Add」。



(5) 开启「symbol configuration」设定接口并单击「Build」：已建立的项目及变量显示在下面树状结构表中。



(6) 勾选，允许 OPC UA 客户端设定监控的变量。每个变量均可在 Access Rights 字段设定其读取/写入权限，更改设定后都需重新 Build 后，再下载至 PLC 主机中。



(7) 最后，下载项目至主机中。

8.2 建立一般联机

8.2.1 使用「UaExpert」Client 建立一般联机

使用 OPC UA Client 与 AX-3 系列控制器 OPC UA server 连接。一般无加密联机。

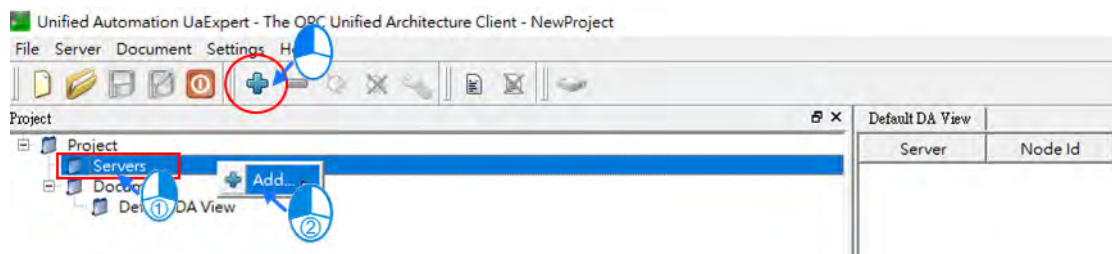
以下为「UaExpert」client 设定方式说明。其它的 OPC UA Client 也是类似的设定方式。

用户可先至 UaExpert 官方网站注册免费下载。

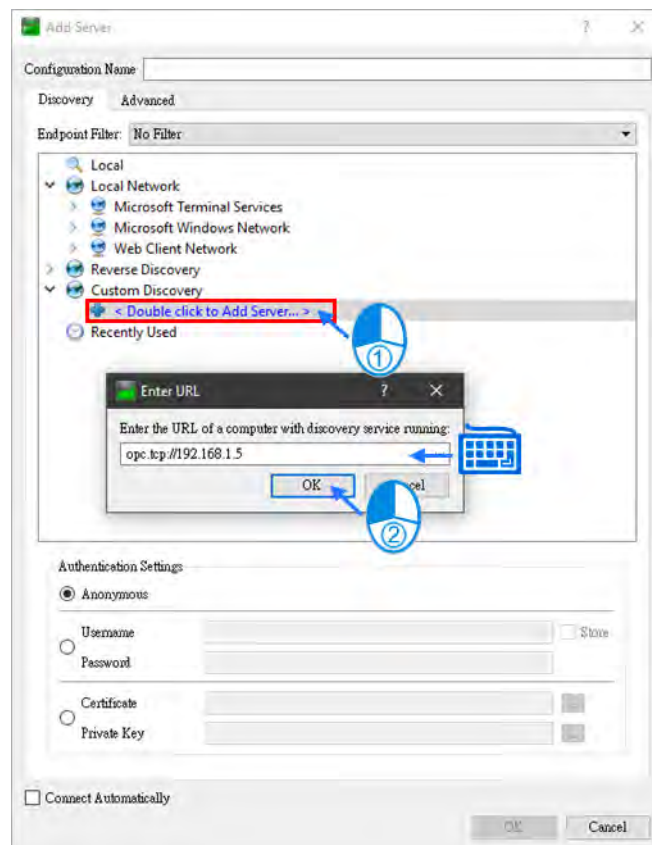
下载网址：<https://www.unified-automation.com/downloads/opc-ua-clients.html>

(1) 执行「UaExpert」

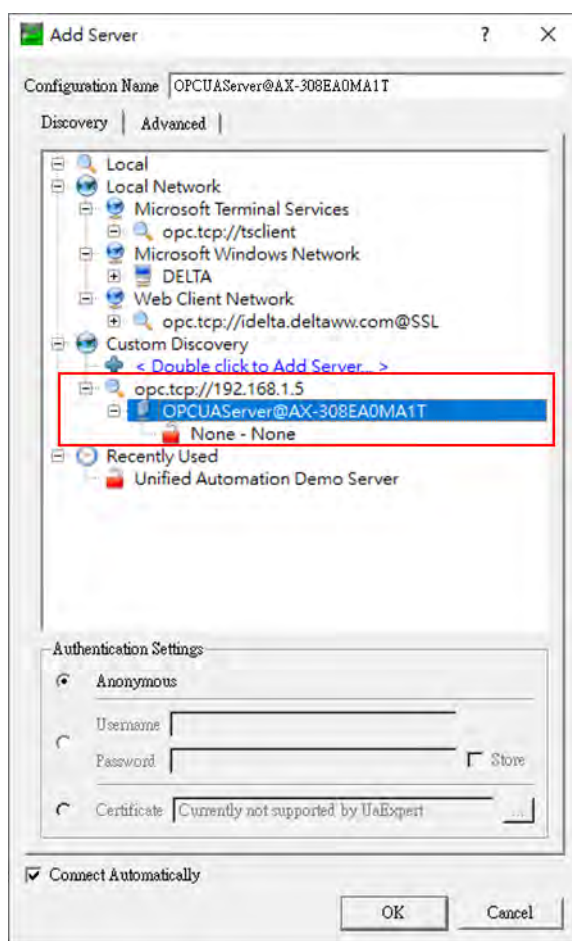
(2) 单击「Servers」>「Add」：开启「Add Server」对话框。



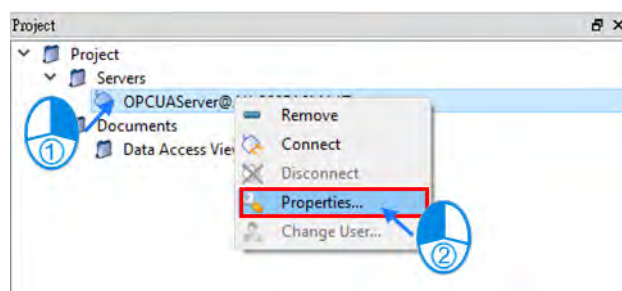
(3) 单击「Custom Discovery」>「<Double click to Add Server...>」>「Enter Url」对话框：输入 opc.tcp://192.168.1.5



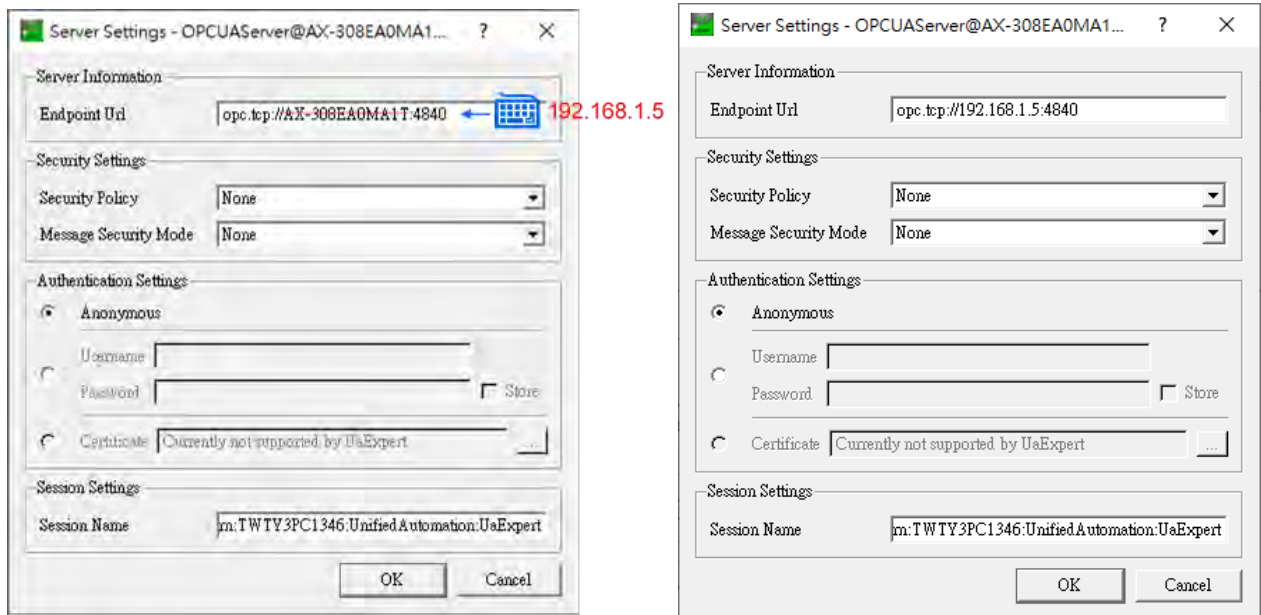
- (4) 完成上一步骤后会出现以下画面并在 **Project** 底下新增「AX308E」
 未使用加密功能，联机方式只会扫出 [无加密] None-None。(不需要输入账号密码及选择加密格式)



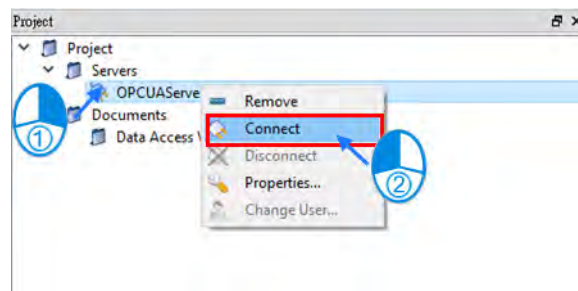
- (5) 修改联机内容：在「Project」>「AX308E」单击鼠标右键 > 「Properties」



(6) 在 Server Settings 对话框里设定 `opc.tcp://192.168.1.5:4840`



(7) Connect



(8) 通过 OPC UA client 变更变量数据状态

选择变量并将其拖拉至 Default DA View 中，可以看到变量信息及状态；在变量 Value 字段单击两下，可修改变量值。

The screenshot displays a software interface with a project tree on the left and a data table in the center. The table contains the following data:

Server	Node Id	Display Name	Value	Datatype	Source Timestamp	Server Timestamp	Statuscode
AX308E	NS4StringIvarDelta-ARM-VWWorks...	aryval4	0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0	Byte	08:25:51.428	08:25:51.428	Good
AX308E	NS4StringIvarDelta-ARM-VWWorks...	bval3	false	Boolean	08:25:52.826	08:25:52.826	Good
AX308E	NS4StringIvarDelta-ARM-VWWorks...	dval2	0	UInt32	08:25:54.211	08:25:54.211	Good
AX308E	NS4StringIvarDelta-ARM-VWWorks...	wval1	0	UInt16	08:25:55.622	08:25:55.622	Good

8.3 建立加密联机

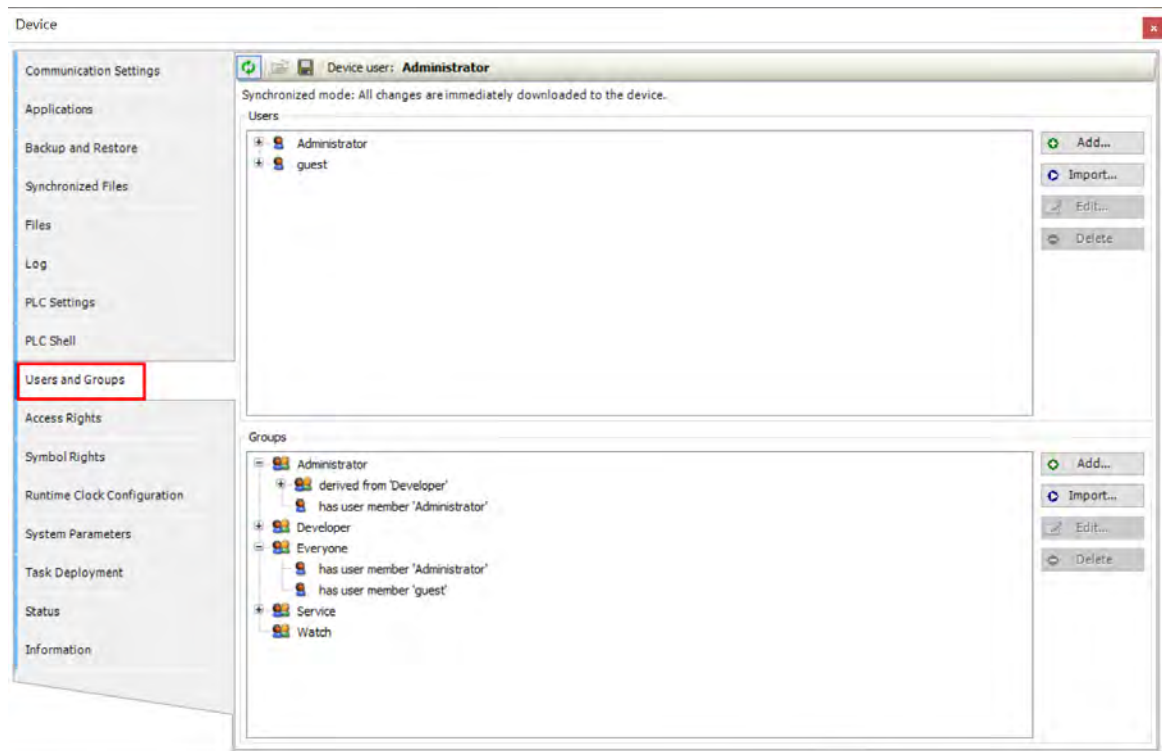
为了加密数据并且安全的与客户端进行数据交换，当第一次建立联机时，服务器需要一个凭证，客户端必须将此凭证分类到可信任的凭证。信任凭证后，也须使用正确的用户账号才能建立联机。下面章节为认证所需设定及操作说明，请用户依序设定操作。

8.3.1 设定用户账号密码


OPC UA Server 使用之账号密码与主机加密之账号密码设定相同。设定方式在 Users and Groups 章节中有详细介绍其权限及设定方式。（请参考第 4.2.1.8 节）

设定两组账号：

1. 默认帐号：Administrator
2. 新建立账号：guest

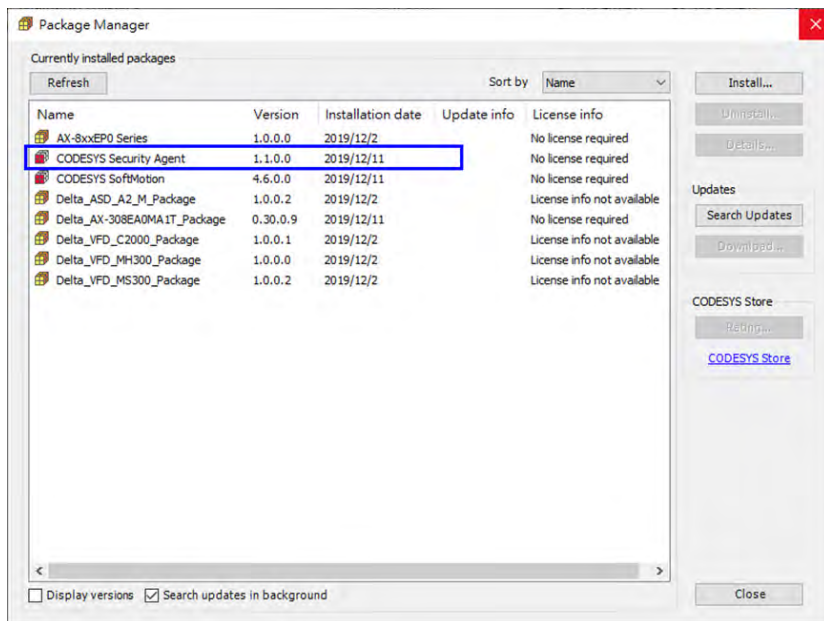


8.3.2 信任凭证设定

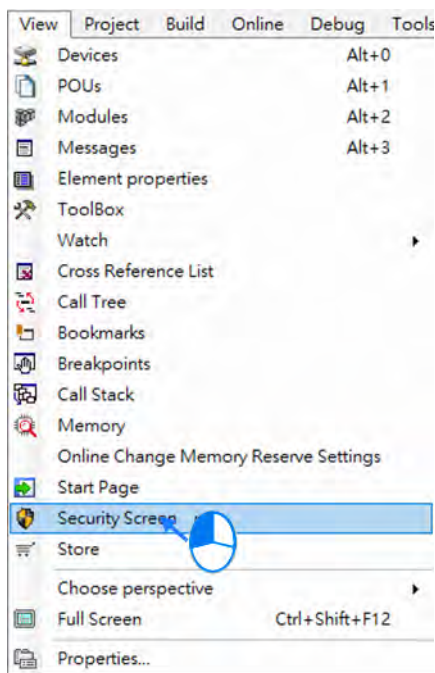
- (1) 安装  CODESYS Security Agent 附加组件。（第一次安装完成后，需重新启动 DIADesigner-AX）

用户可先至 CoDeSys Store 免费下载 CODESYS Security Agent。

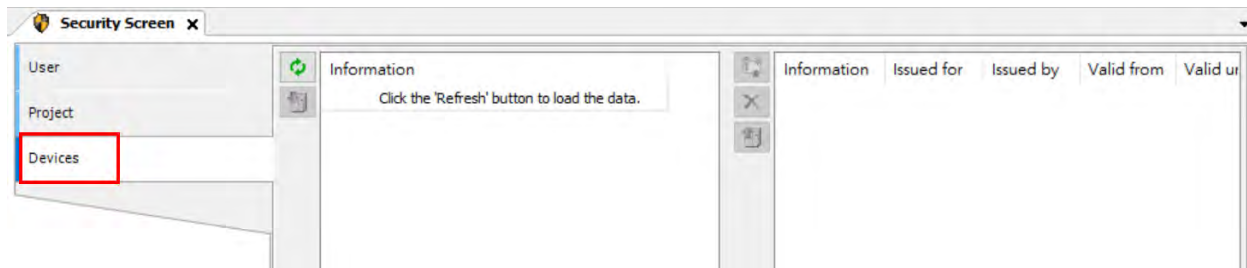
下载网址：https://store.codesys.com/codesys-security-agent.html?_SID=U




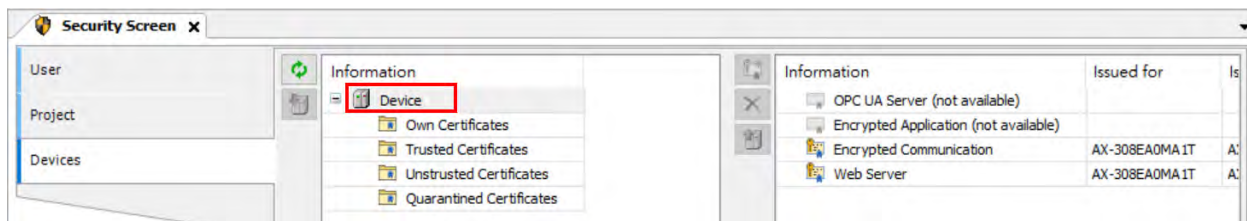
(2) 单击「View」 > 「Security Screen」




(3) 选择「Devices」页签

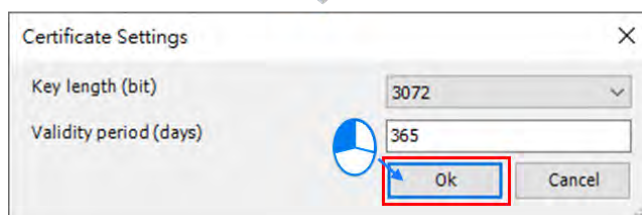
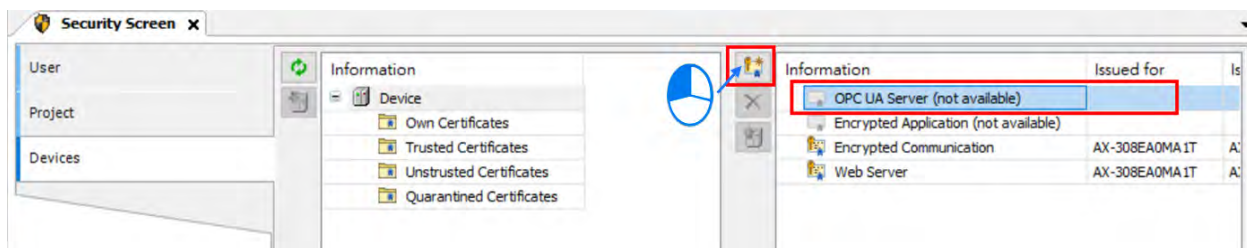


(4) 单击  后，选择左侧窗口中控制器「Device」：控制器中需要凭证的所有服务都显示在右侧窗口中

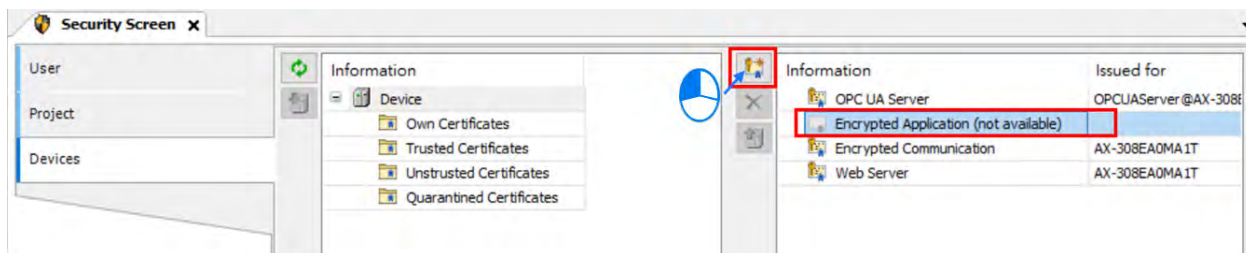


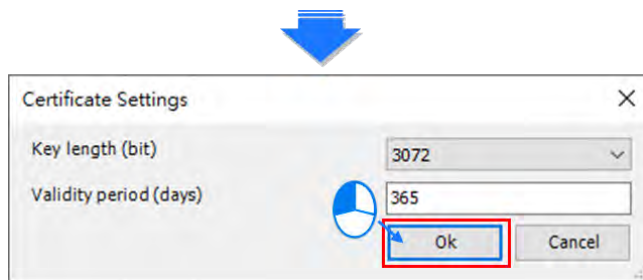
(5) 单击 ，建立控制器的新凭证：在以下「Certificate Settings」对话框做设定。

选择「OPC UA Server」

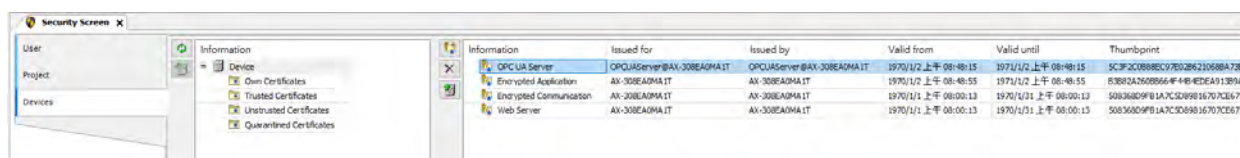


选择「Encrypted Application」





(6) 启动完成 OPC UA Server 及 Encrypted Application 凭证*



*注：凭证建立完成，需将 PLC 主机断电重启才会生效。

8.3.3 使用「Prosys OPC UA Client」建立加密联机

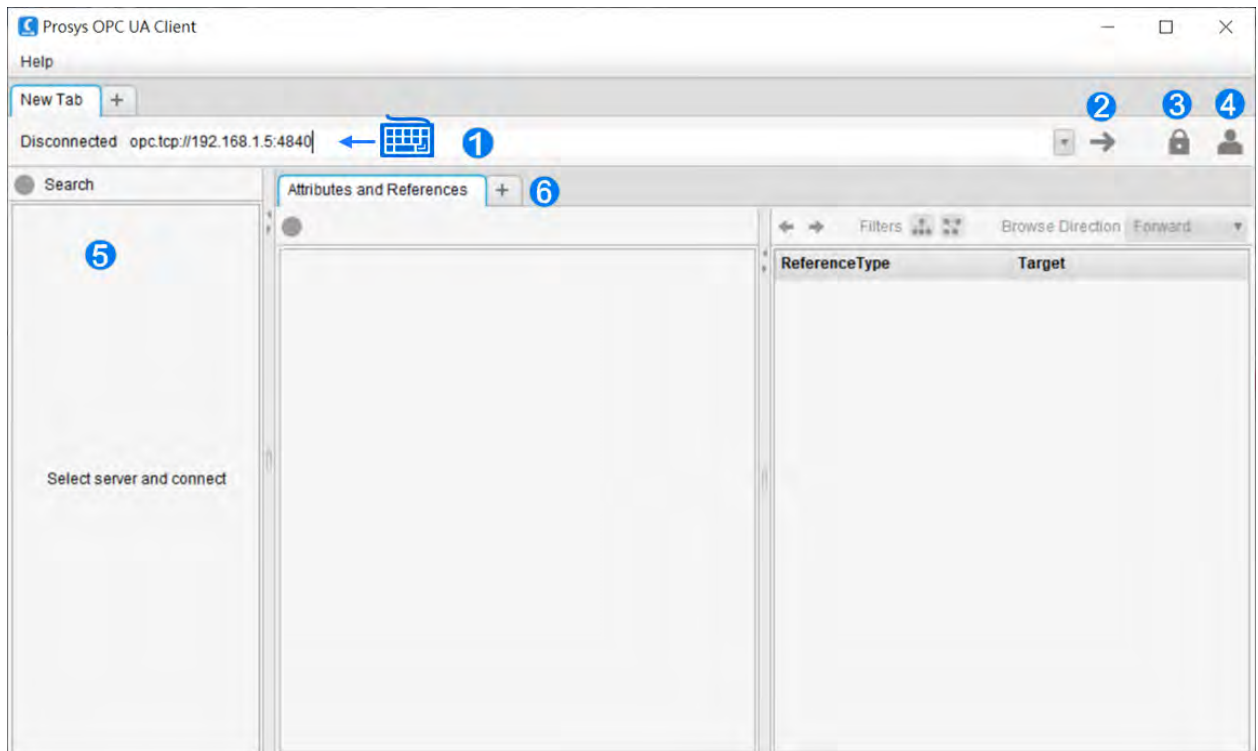
此节以 Prosys OPC UA Client V3.2.0 版本 操作联机。


用户可先至 Prosys OPC UA Client 官方网站注册免费下载。

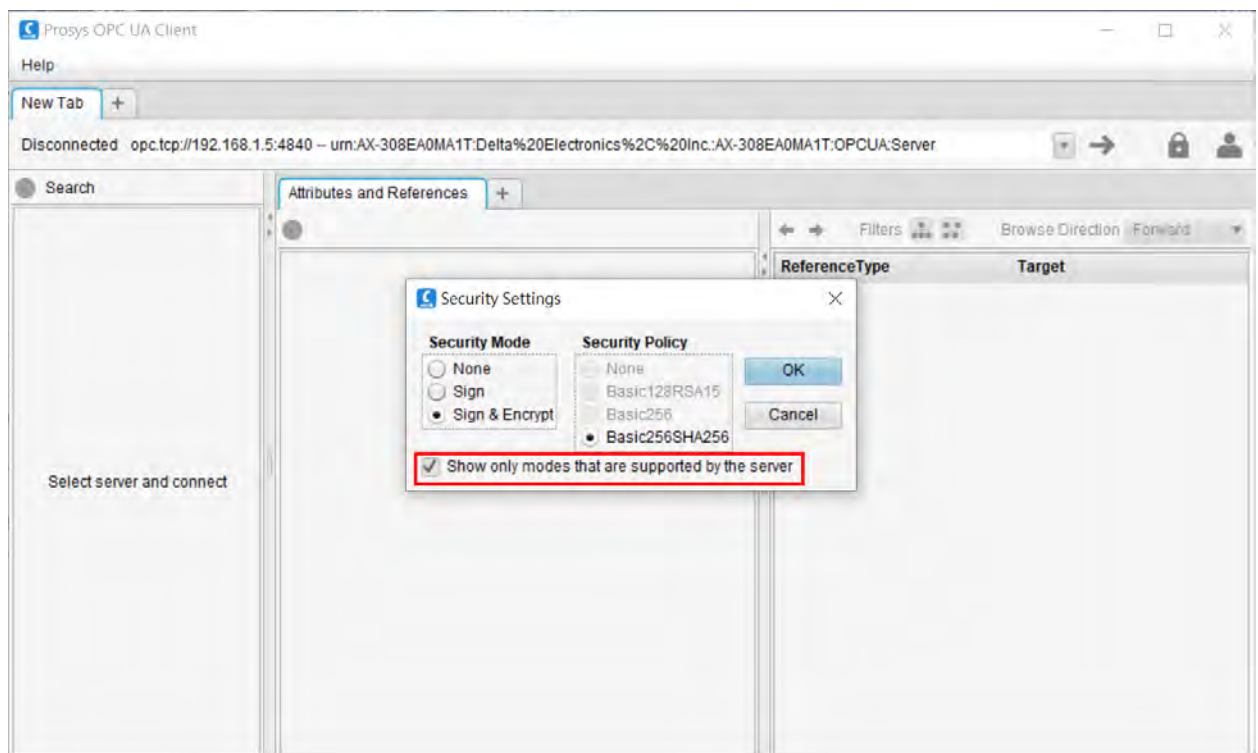
下载网址：<https://downloads.prosysopc.com/opc-ua-client-downloads.php>

(1) 执行「Prosys OPC UA Client」

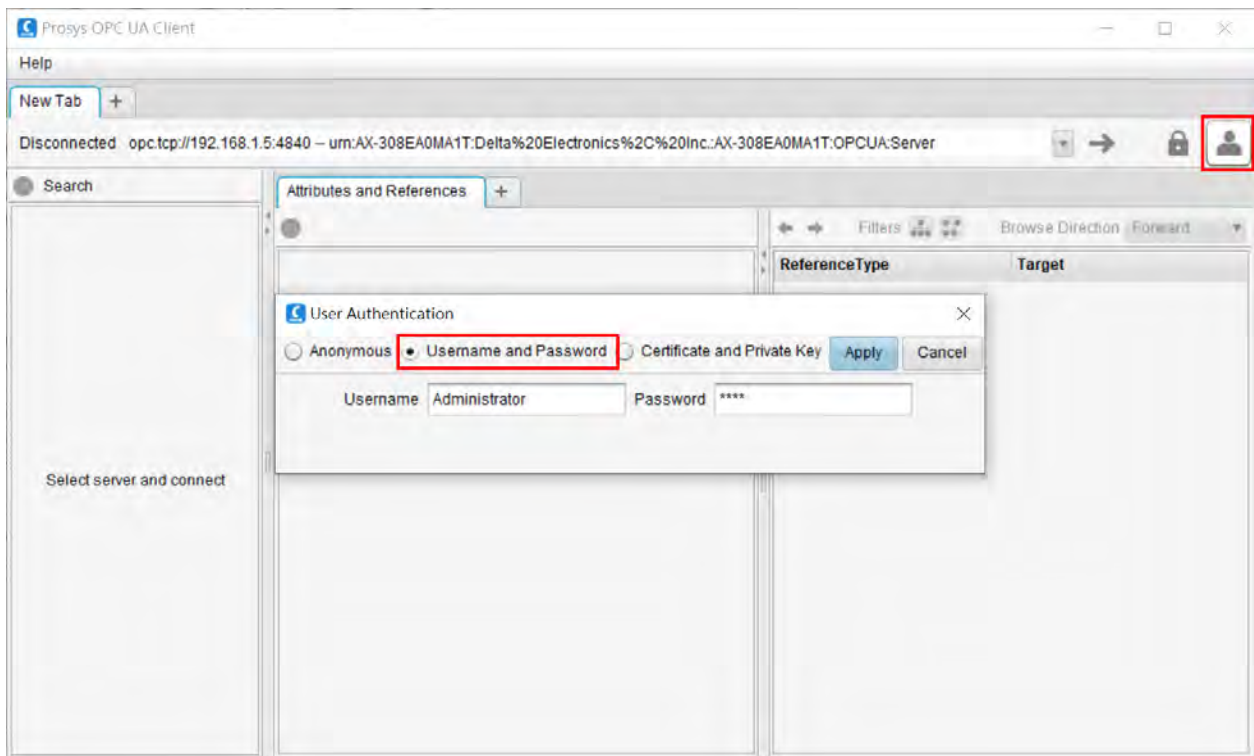
(2) 在①「Disconnected」中，输入 OPC UA Server IP 地址，opc.tcp://192.168.1.5:4840



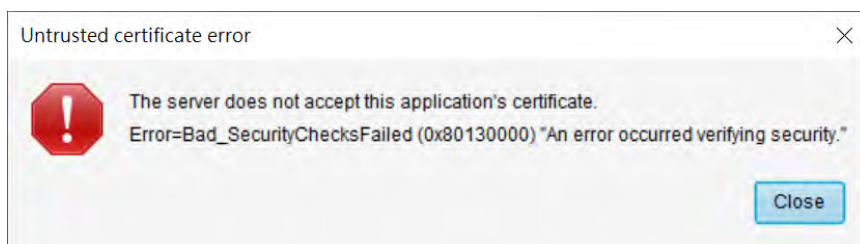
- (3) 选择联机加密方式③ ·  「Security Settings」窗口(仅支持 Basic256SHA256)「勾选 Show only modes that are supported by the server」



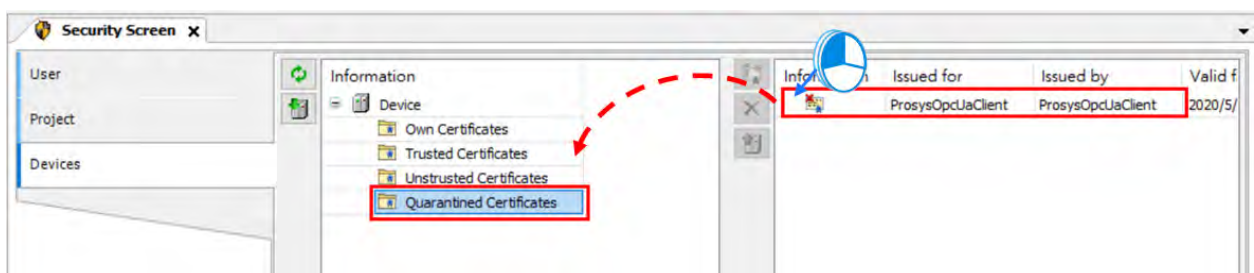
- (4) 单击④ ·  设定用户账号密码 > 「Apply」。设定完成后，单击②  联机。

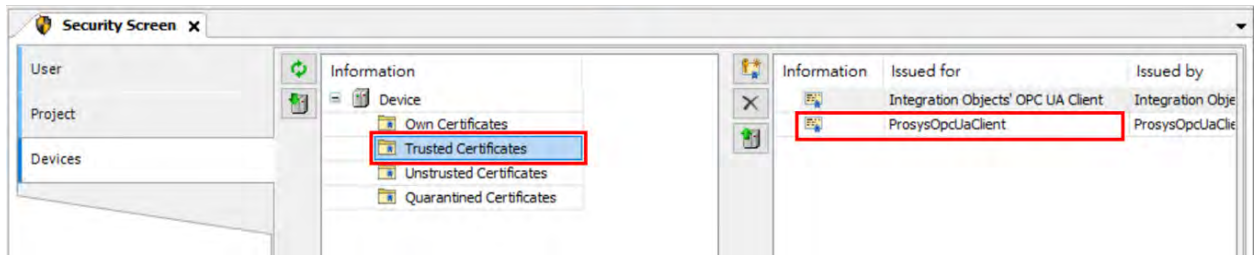


(5) 跳出提示窗口→ 不被 Server 允许的应用程序凭证。



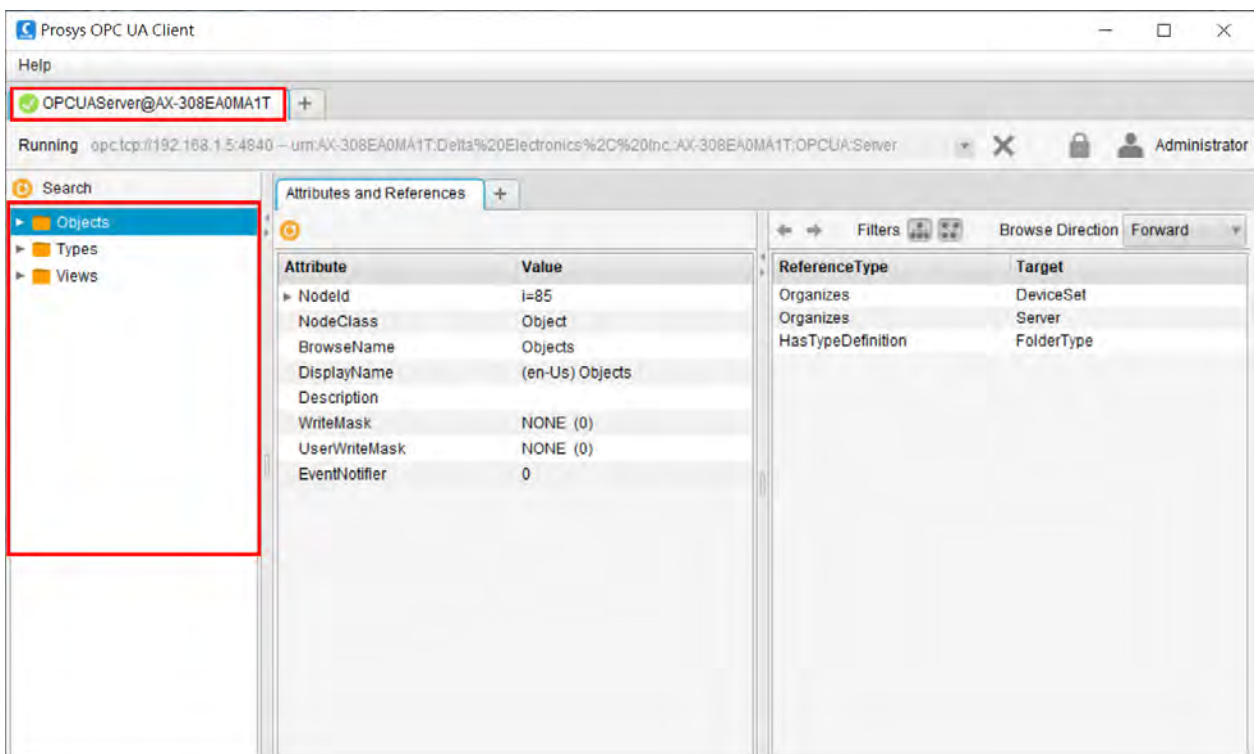
(6) 回到 Security Screen，「Quarantined Certificates」里，有未信任的凭证「ProsysOpcUaClient」，将此凭证拖拉至「Trusted Certificates」中。



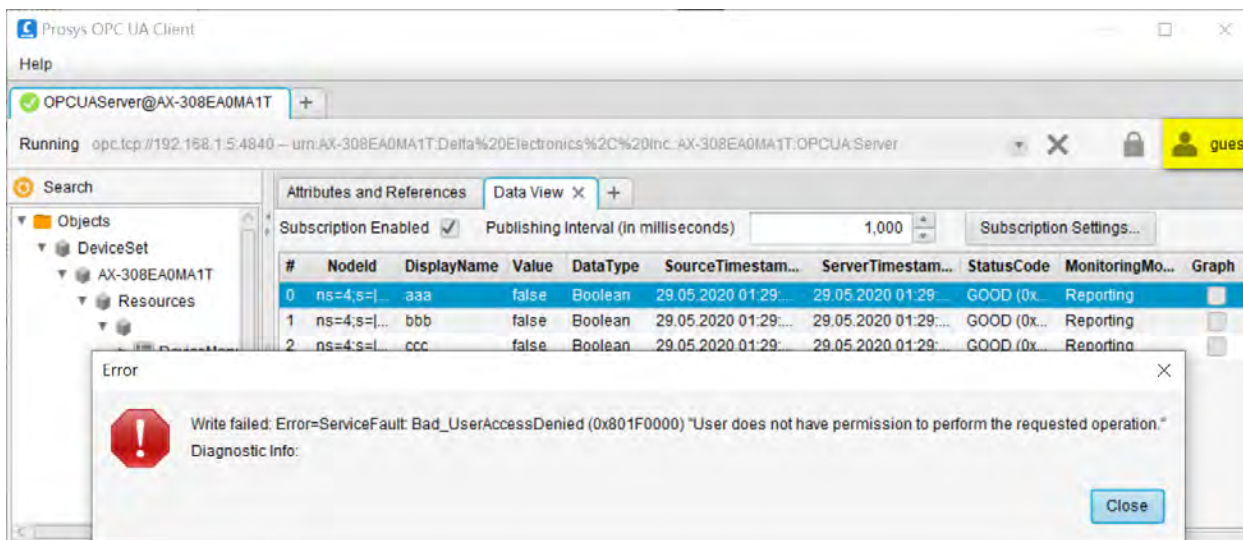


- (7) Client 再重新联机就可联机成功。在⑤区域中，显示联机至 OPC UA Server，可供读写的信息树状结构组织。在⑥区域中则显示单击项目之详细说明。亦可添加分页 Data View，拉取用户所需读写之装置至分页中监控。

User : Administrator



User : guest (无写入权限)



8.3.4 使用「UaExpert」建立加密联机

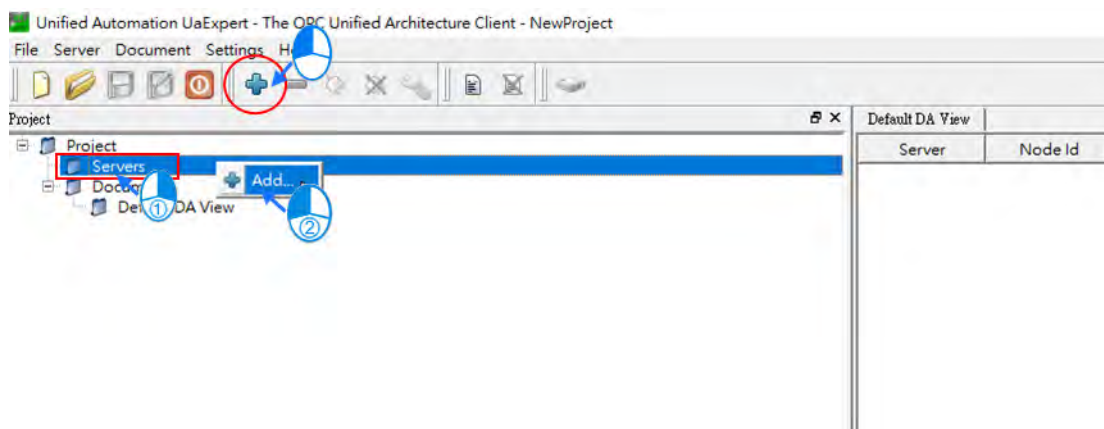
此节以 UaExpert V1.5 版本操作联机。

用户可先至 UaExpert 官方网站注册免费下载。

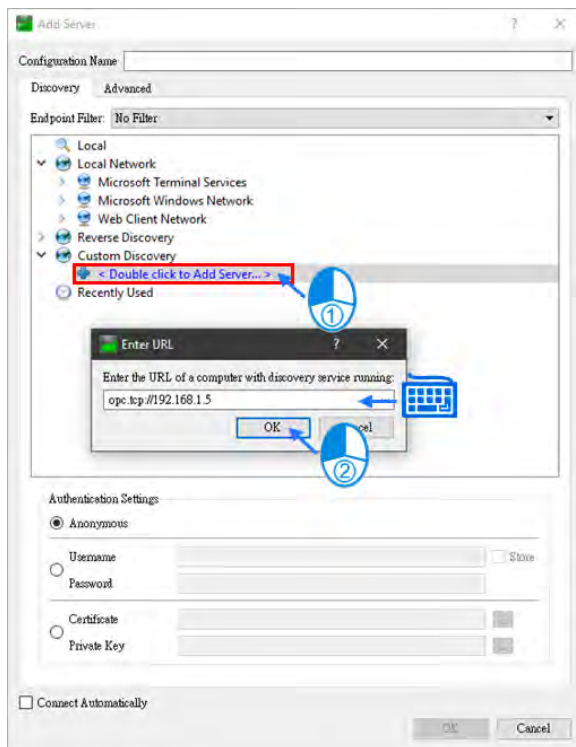
下载网址：<https://www.unified-automation.com/downloads/opc-ua-clients.html>

(1) 执行「UaExpert」

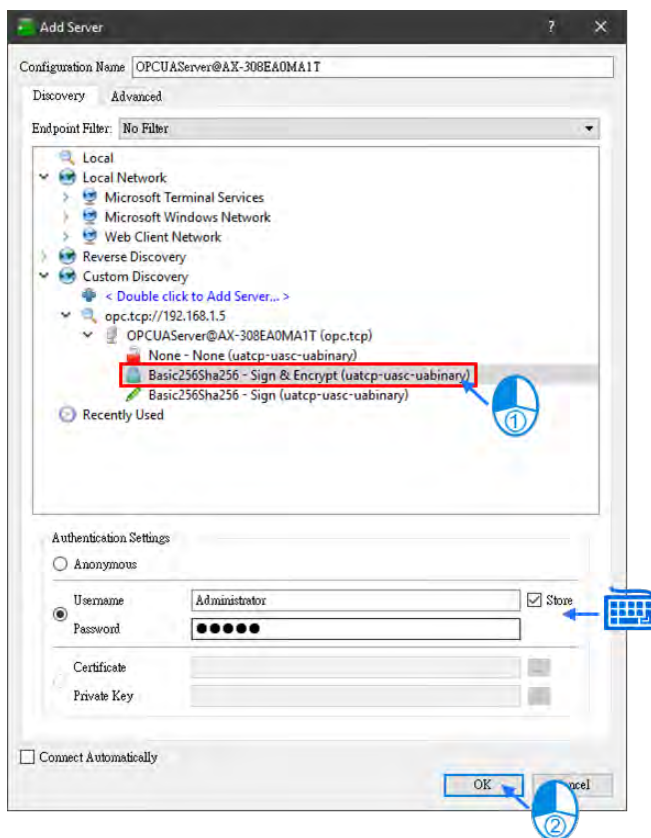
(2) 单击「Servers」 > 「Add」：开启「Add Server」对话框。



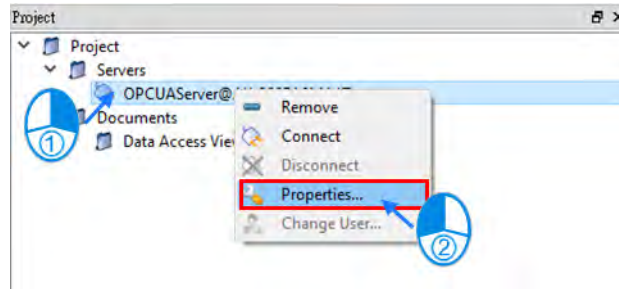
(3) 单击「Custom Discovery」 > 「<Double click to Add Server...>」 > 「Enter URL」对话框：输入 `opc.tcp://192.168.1.5`



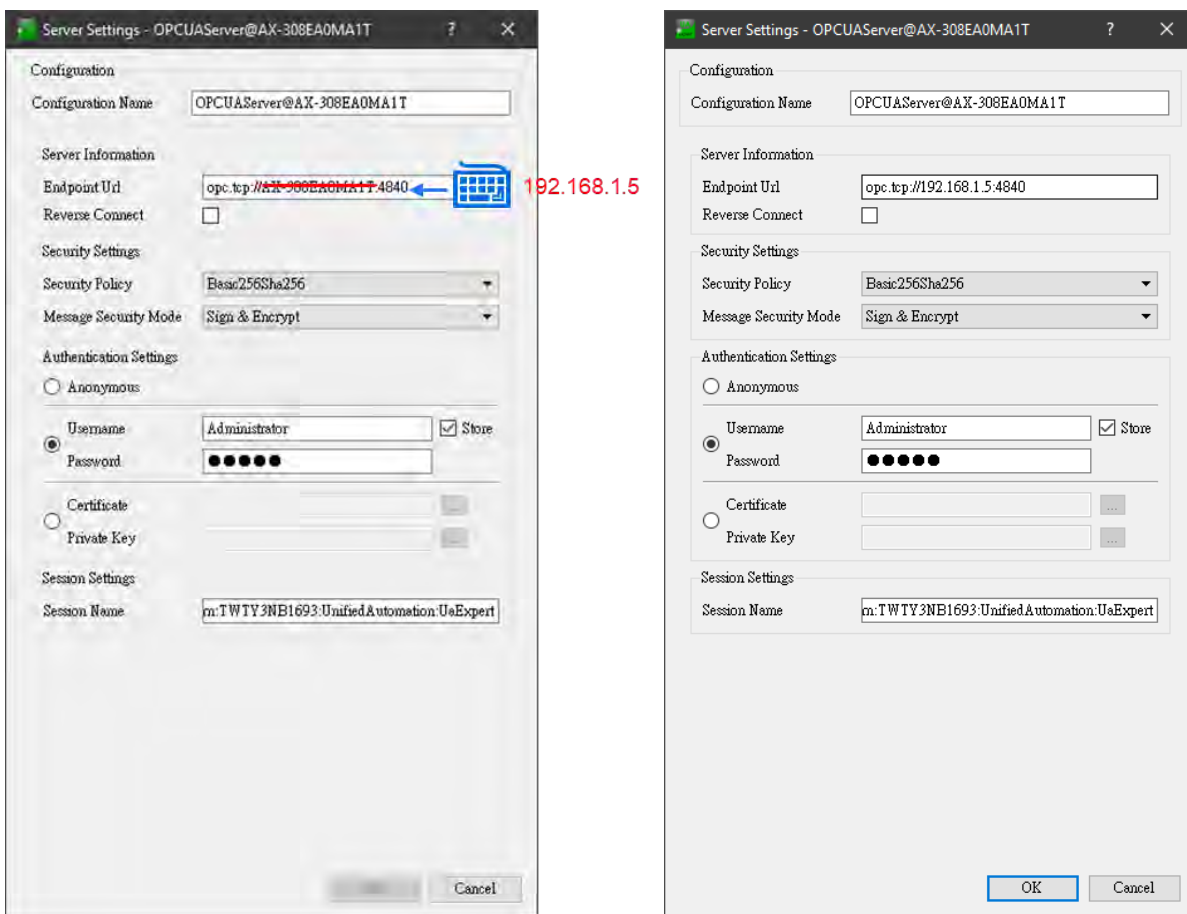
- (4) 完成上一步骤后会出现以下画面并在 Project 底下新增「AX-308E」，联机方式选择「Basic256SHA256」，使用加密功能需设定用户账号密码。



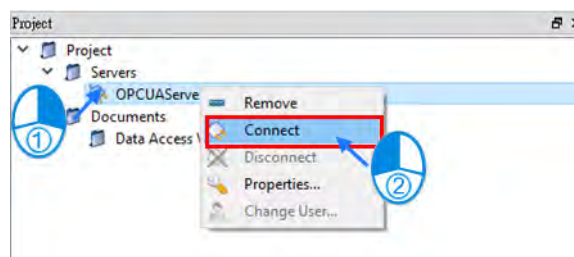
(5) 修改联机内容：在「Project」 > 「AX-308E」右击鼠标键 > 「Properties」



(6) 在 Server Settings 对话框里设定 opc.tcp://192.168.1.5:4840



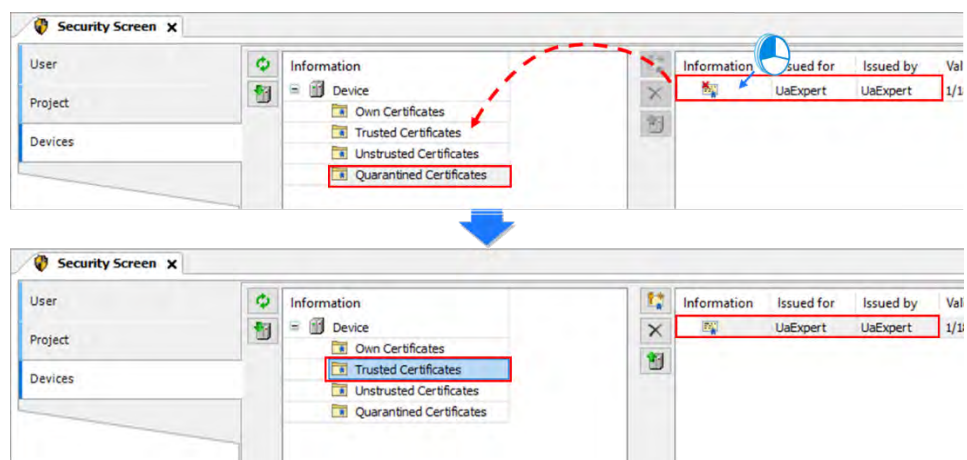
(7) Connect



(8) 跳出提示窗口→ 不被 Server 允许的应用程序凭证。

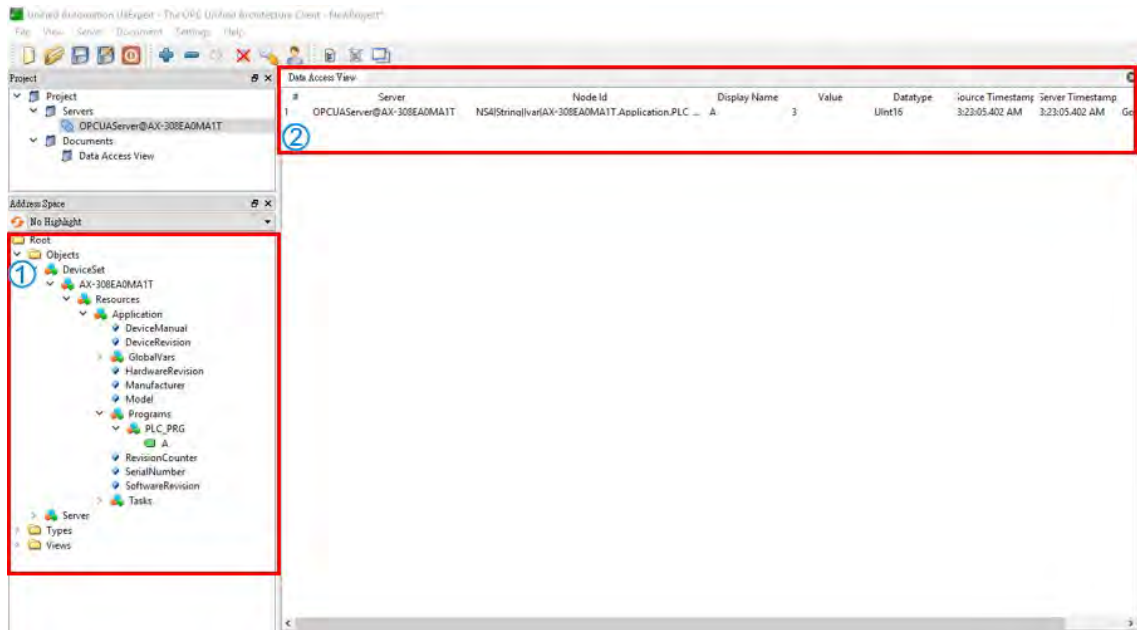
Timestamp	Source	Server	Message
1/18/2021 6:25:10.622 PM	Server Node	OPCUAServer@AX-308EA0MA1T	Endpoint: 'opc.tcp://AX-308EA0MA1T:4840'
1/18/2021 6:25:10.622 PM	Server Node	OPCUAServer@AX-308EA0MA1T	Security policy: 'http://opcfoundation.org/UA/SecurityPolicy#Basic256Sha256'
1/18/2021 6:25:10.622 PM	Server Node	OPCUAServer@AX-308EA0MA1T	ApplicationUri: 'urn:AX-308EA0MA1T:Delta%20Electronics%2C%20Inc.:AX-308EA0MA1T:OPCUA:Server'
1/18/2021 6:25:10.622 PM	Server Node	OPCUAServer@AX-308EA0MA1T	Used UserTokenType: UserName
1/18/2021 6:25:10.737 PM	Server Node	OPCUAServer@AX-308EA0MA1T	Error 'BadSecurityChecksFailed' was returned during OpenSecureChannel
1/18/2021 6:25:10.737 PM	Server Node	OPCUAServer@AX-308EA0MA1T	Connection status of server 'OPCUAServer@AX-308EA0MA1T' changed to 'Disconnected'.

(9) 回到 Security Screen，「Quarantined Certificates」里，有未信任的凭证「UaExpert」，将此凭证拖拉至「Trusted Certificates」中。



(10) Client 再重新联机就可联机成功。在①区域中，显示联机至 OPC UA Server，可供读写的信息树状结构组织。在②区域中则显示单击项目之详细说明。亦可添加分页 Data View，拉取用户所需读写之装置至分页中监控。

User : Administrator



第9章 通讯功能

目录

9.1	EtherCAT 功能.....	9-4
9.1.1	EtherCAT 总线介绍	9-4
9.1.2	EtherCAT 主站设定	9-5
9.1.3	EtherCAT 从站设定	9-7
9.1.4	EtherCAT 从站参数备份还原	9-10
9.1.4.1	参数备份流程说明	9-10
9.1.4.2	备份还原介绍.....	9-11
9.1.4.2.1	参数备份操作	9-13
9.1.4.2.2	参数还原操作	9-15
9.2	串行 (Serial) 通讯	9-17
9.2.1	Modbus Serial Port 元件	9-17
9.2.1.1	Modbus Serial Port 元件建立	9-17
9.2.1.2	Modbus Serial Port 组件页面说明	9-19
9.2.2	Modbus Serial Master	9-20
9.2.2.1	Modbus Serial Master 组件建立	9-20
9.2.2.2	Modbus Serial Master 设定说明	9-23
9.2.2.3	Modbus Serial Slave 设定说明.....	9-26
9.2.3	Modbus Serial Slave	9-31
9.2.3.1	Modbus Serial Slave 组件建立.....	9-31
9.2.3.2	Modbus Serial Slave 设定说明.....	9-33
9.3	以太网 (Ethernet) 通讯.....	9-35
9.3.1	Ethernet 组件	9-35
9.3.1.1	Ethernet 组件建立.....	9-35
9.3.1.2	Ethernet 组件页面说明	9-36
9.3.2	Modbus TCP Master	9-40

9.3.2.1	Modbus TCP Master 组件建立	9-40
9.3.2.2	Modbus TCP Master 设定说明	9-42
9.3.2.3	Modbus TCP Slave 设定说明	9-43
9.3.3	Modbus TCP Slave	9-49
9.3.3.1	Modbus TCP Slave 组件建立	9-49
9.3.3.2	Modbus TCP Slave 设定说明	9-51
9.4	EtherNet/IP 通讯	9-53
9.4.1	EtherNet/IP 简介	9-53
9.4.1.1	EtherNet/IP 介绍	9-53
9.4.1.2	常见网络名词说明	9-53
9.4.1.3	Ethernet 特色	9-54
9.4.1.3.1	Delta EIP 系统架构图	9-54
9.4.1.3.2	EIP 特点	9-54
9.4.2	EtherNet/IP Scanner Function	9-55
9.4.2.1	建立变频器 MS300 数据读写	9-55
9.4.2.1.1	系统硬件配置	9-55
9.4.2.1.2	隐性报文数据读写设定	9-55
9.4.2.1.3	显性报文 CIP Object 数据读写设定	9-61
9.4.2.2	建立 AS00SCM-A (AS-FEN02 通讯卡) 数据读写	9-62
9.4.2.2.1	设定 AS00SCM-RTU 右侧 IO 模块配置	9-62
9.4.2.2.2	导入 AS00SCM-RTU EDS 文件	9-64
9.4.2.2.3	设定 AS00SCM-RTU 相关 EtherNet/IP 参数	9-71
9.4.2.2.4	AS00SCM-RTU 右侧 IO 模块操作	9-73
9.4.2.2.5	AS00SCM-RTU 模块参数信息	9-76
9.4.3	EtherNet/IP Adapter Function	9-76
9.4.3.1	Studio 5000 软件操作	9-76
9.4.3.1.1	架构	9-77
9.4.3.1.2	项目建立	9-77
9.4.3.2	Scanner 建立	9-77
9.4.3.2.1	模块建立	9-77
9.4.3.3	Adapter 联机	9-78
9.4.3.3.1	EDS 建立	9-78
9.4.3.3.2	EDS 载入	9-82

9.4.3.3.3 Adapter 建立 EDS 加载	9-82
9.4.3.3.4 下载	9-82
9.4.3.3.5 数据交换	9-83
9.4.4 CIP Object	9-84
9.4.4.1 Object list	9-84
9.4.4.2 数据类型定义 (Data Type)	9-84
9.4.4.3 Identity Object (Class ID : 01 Hex)	9-87
9.4.4.4 Assembly Object (Class ID : 04 Hex)	9-89
9.4.4.5 TCP/IP Interface Object (Class ID : F5 Hex)	9-89
9.4.4.6 Ethernet Link Object (Class ID : F6 Hex)	9-92
9.4.5 台达 EIP 产品一览表	9-94
9.4.5.1 台达 EIP 产品列表支持 Adapter 功能	9-94
9.4.5.2 台达 EIP 产品支持 Scanner 功能	9-94
9.5 网络安全	9-95
9.6 FTP 功能	9-95
9.6.1 FTP Server	9-95
9.6.1.1 FTP Server 规格	9-95
9.6.1.2 FTP Server 功能启动流程	9-96
9.6.1.3 FTP Client 联机 (AX308E 系列控制器为 FTP Server)	9-100
9.6.2 FTP Client	9-100

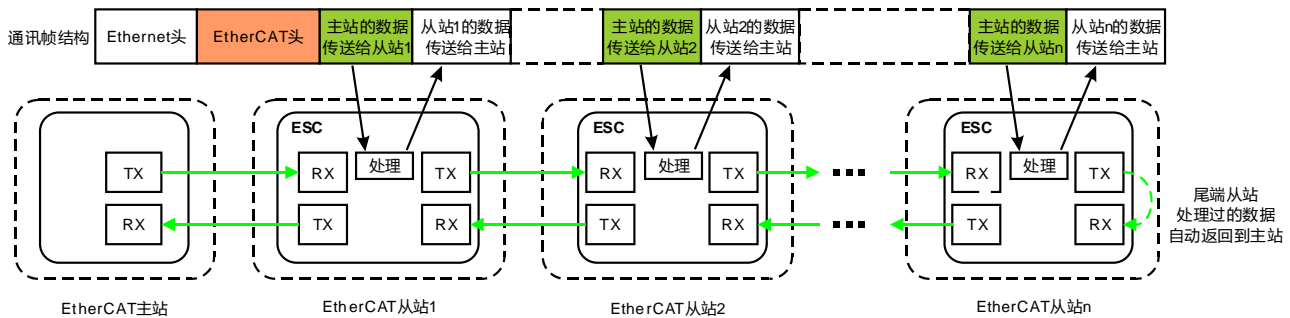
9.1 EtherCAT 功能

9.1.1 EtherCAT 总线介绍

- EtherCAT 总线优势

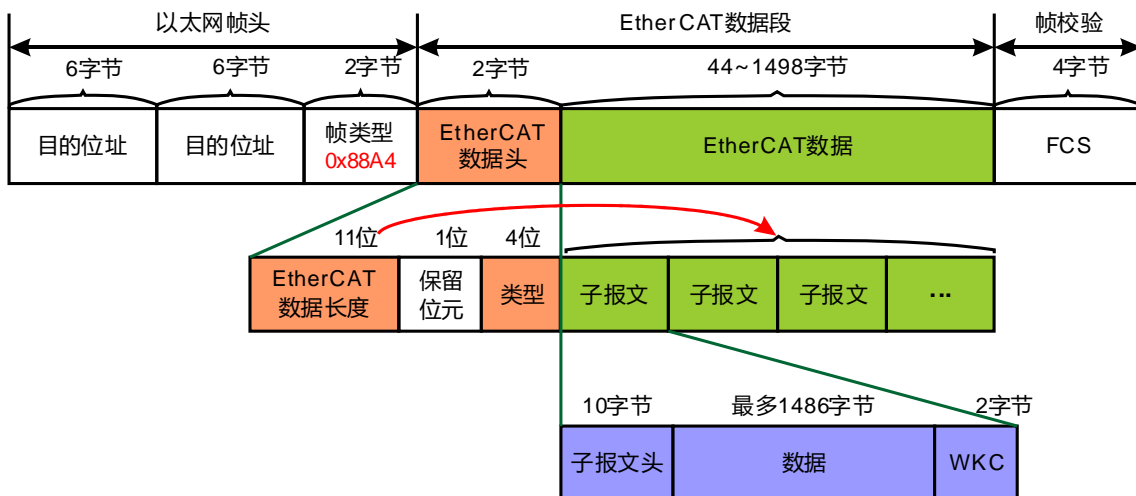
EtherCAT 总线是基于以太网的现场总线，EtherCAT 网络通讯速率为 100Mbps，相邻两个节点之间的距离不超过 50 米。EtherCAT 网络和普通以太网有明显不同，同一个 EtherCAT 网络内，只有一个 EtherCAT 主站，另外 EtherCAT 从站有专门处理 EtherCAT 通讯数据的芯片 ESC (EtherCAT Slave Controller)。ESC 芯片可以在 EtherCAT 数据帧通过时，取出主站发送给该从站的数据，并将该从站需要传送给主站的数据插入到 EtherCAT 数据帧中，网络中的最后一个 EtherCAT 从站将处理过的报文依次返回给主站，数据传输示意图如下图所示。

主站和从站数据交换的过程中，从站对 EtherCAT 总线数据的处理，是通过 ESC 芯片硬件实现，如此可以大大提高总线数据处理的效率。由于从站具有 ESC 芯片，可以使主站在一笔 EtherCAT 数据帧中和所有从站通讯一次，如此可以提高通讯效率。



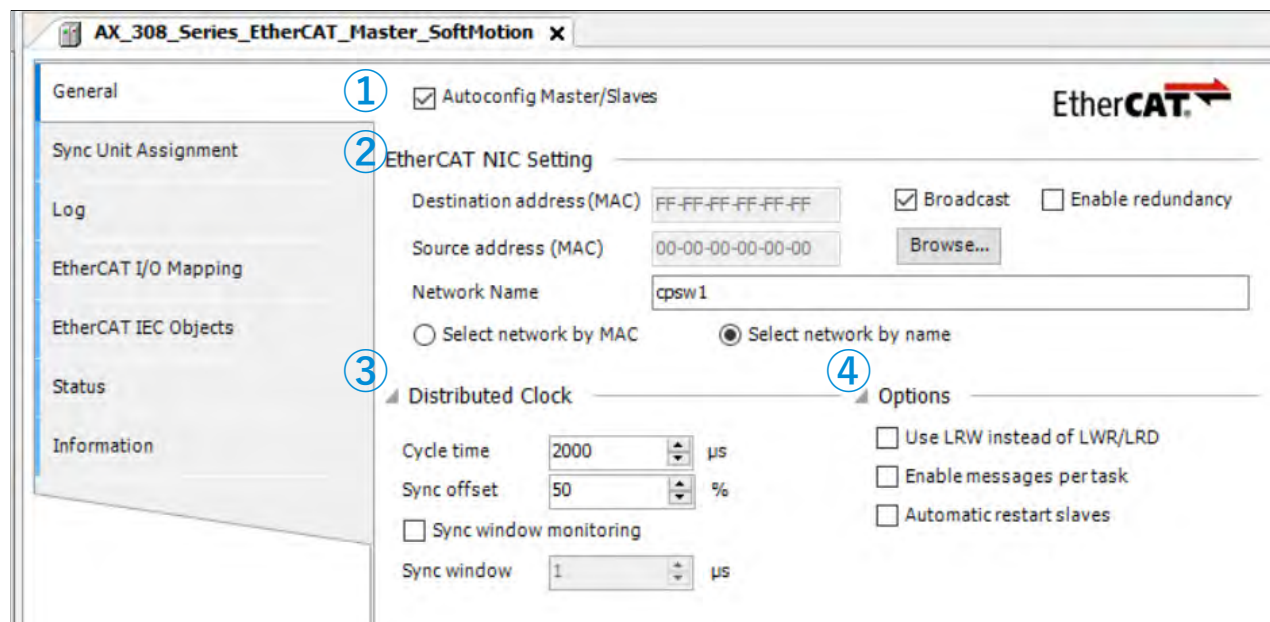
- 控制器和从站之间的 EtherCAT 通讯说明

EtherCAT 总线是基于以太网的现场总线，所以 EtherCAT 数据帧仍使用 UDP/IP 以太网数据帧结构。EtherCAT 数据帧结构如下图所示，“EtherCAT 数据段”包括 2 字节的 EtherCAT 数据头和 44~1498 字节的 EtherCAT 数据，数据区有一个或多个 EtherCAT 子报文组成。EtherCAT 数据可以以某种协议定义和解析，只要主站和从站都符合定义的协议即可，目前使用比较多的协议有两种：COE(CANopen Over EtherCAT)和 SOE (Sercos Over EtherCAT)。



9.1.2 EtherCAT 主站设定

此章节针对 EtherCAT_Master_SoftMotion 画面介绍



● General

① Autoconfig Master/Slaves : 启动主站的此选项，可进行快速与简易的基本配置。一般建议勾选此设定。

② EtherCAT NIC Setting

- ◆ Destination address (MAC) : EtherCAT 网络中要接收电报的设备的 MAC 地址。
- ◆ Source address(MAC) : PLC 网络接口的 MAC 地址 (使用 Browse 扫描从站，请选择 CPSW1 模块)
- ◆ Network Name : 网络的名称或 MAC。
- ◆ Select Network by MAC : 网络由 MAC-ID 指定。(默认名称为 CPSW1*1)
- ◆ Select network by Name : 网络由网络名称标识，并且项目与设备无关。

*注 1 : EtherCAT_Master_SoftMotion 默认为 Network by MAC，名称为 CPSW1。

③ Distributed Clock

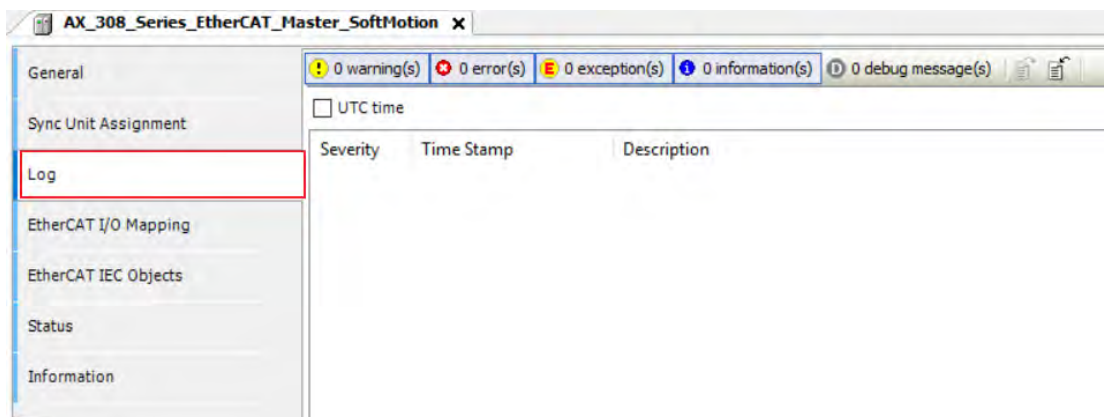
- ◆ Cycle time : 主站发送将会根据该循环时间向从站发送相应数据报文。
- ◆ Sync offset : 设定 PLC cycletime 对应 EtherCAT 从站 SYNC 中断的延迟时间。例如，若设定 sync offset 为 20%，则 PLC cycle 较 EtherCAT 从站的 SYNC 中断延后 20% 的周期时间，代表 PLC cycle 仍有 80% 的周期时间裕度，若 PLC cycle 在此时间裕度中延迟，不会造成同步数据遗失。
- ◆ Sync window monitoring : 当启动后可以监视从站的同步。
- ◆ Sync window : 设定监视从站同步时间范围。

④ Options

- ◆ Use LRW instead of LWR/LRD：默认以单独的读（LRD）和写命令（LWR）。当启动选项后，会以组合的读/写命令（LRW）下达 PDO。
- ◆ Automatic restart slaves：当启动后主站 EtherCAT 断线会立即重启从站。

● Log

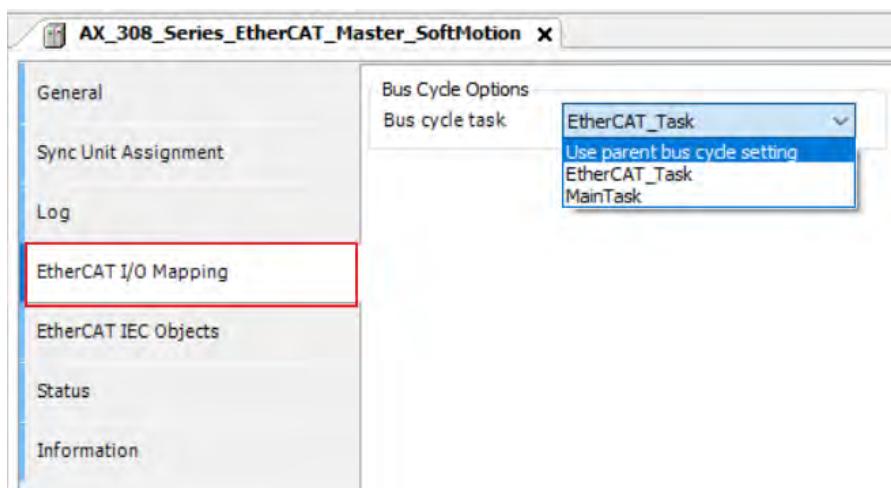
此画面为错误异常后做为记录与显示，可参考第 4.2.1.5 节 Log。



● EtherCAT I/O Mapping

这章节介绍 EtherCAT 需选择对应的 Task，当被选定后的 Task 的扫描时间会跟随 EtherCAT_Master 的周期同步时间。

- Bus cycle task：选择一个 Task 为 EtherCAT 同步时间。如果选择 Use parent bus cycle setting 则系统会自动选择 Task 中最短周期时间改为 EtherCAT 周期时间。



9.1.3 EtherCAT 从站设定

此章节针对 EtherCAT_Slave 画面介绍，添加从站的方式有两种，第一种是选轴从站设备添加，第二种为扫描的方式添加。

- General
 - Address
 - ① EtherCAT address：由主站在启动期间分配给从站地址。
 - Distributed Clocks
 - ② Select DC：设置从站分布式时钟。
- ③ Startup Checking

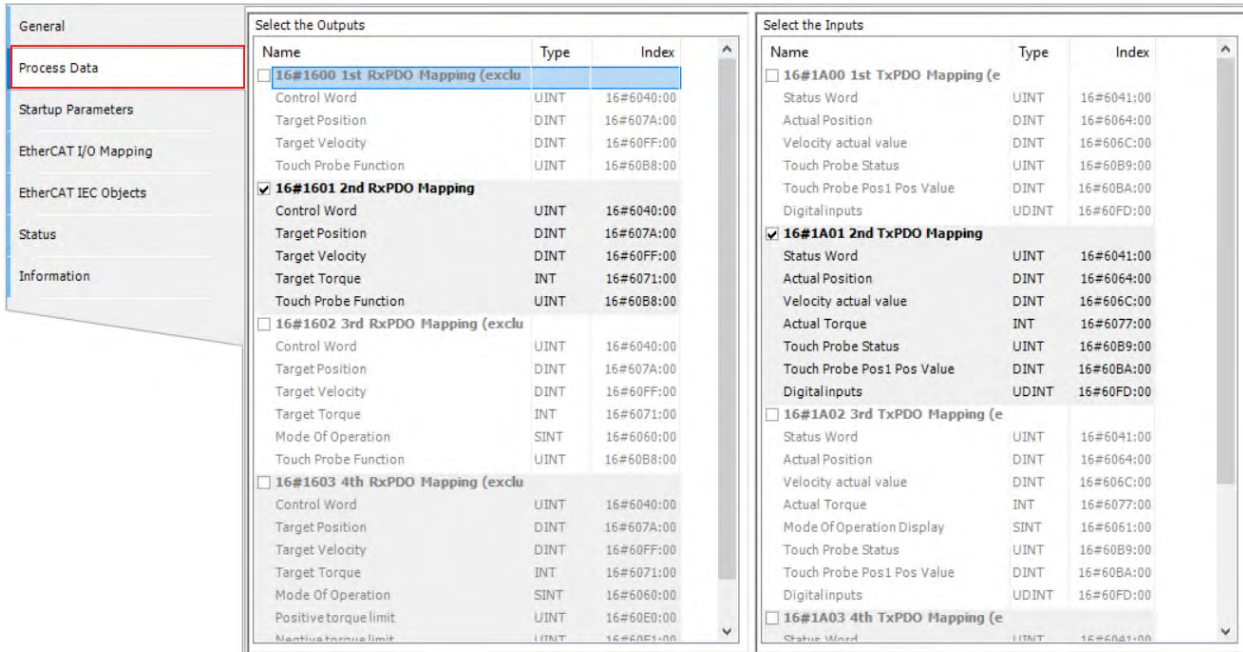
功能	说明
Check vendor ID	默认情况下，系统启动时会根据当前配置设置检查设备的供货商 ID 和产品 ID。如果未通过，则总线停止运行并不执行后续动作。
Check product ID	
Check revision number	系统启动过程中根据在下拉列表中的选择检查修订号。

- ④ Timeouts

功能	说明
SDO access	在系统启动时传输 SDO 列表。单位为毫秒
I -> P	从 Init 模式切换到 Pre operational 模式。单位为毫秒。
P -> S / S -> O	从 Pre operational 模式切换到 Safe Operational 模式或从 Safe-Op 模式切换到 Operational 模式。单位为毫秒。

● Process Data

EtherCAT 网络的数据交换为 CoE 的 PDO 映像，是一种主站与从站架构的周期性数据交换；从站要送出到主站的数据将包入 TxPDO 中，从站要从主站读取的数据则包入 RxPDO 中，在「PDO 映像」页面的输入设定与输出设定字段便列出可供数据交换的 PDO，并且可对 PDO 内容进行编辑。在设备的 ESI 文件中，已经定义了默认勾选的 PDO 与 PDO 的内容，并依据 ESI 的定义，有些设备的 PDO 内容允许用户自行编辑。

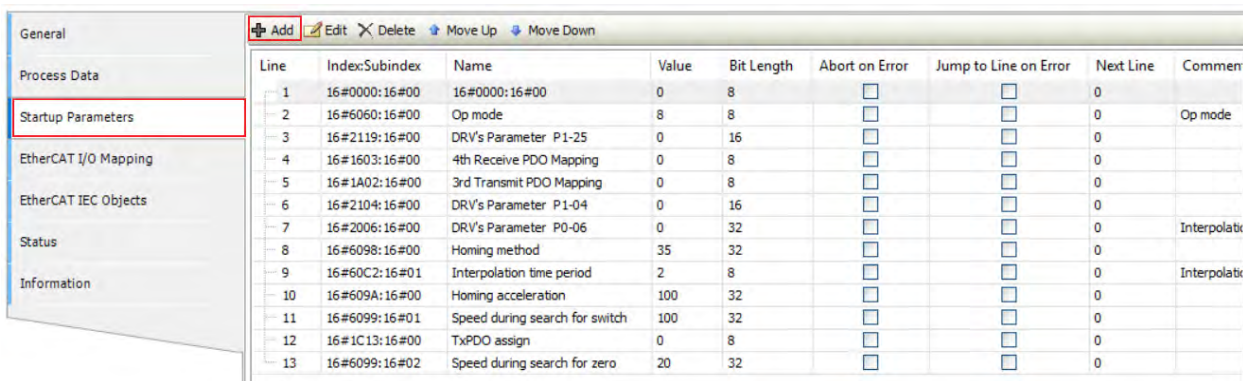


用户可依照需求选择不同组 PDO 做使用。选择 PDO 的数据量越多，占 PLC 效能会越大。

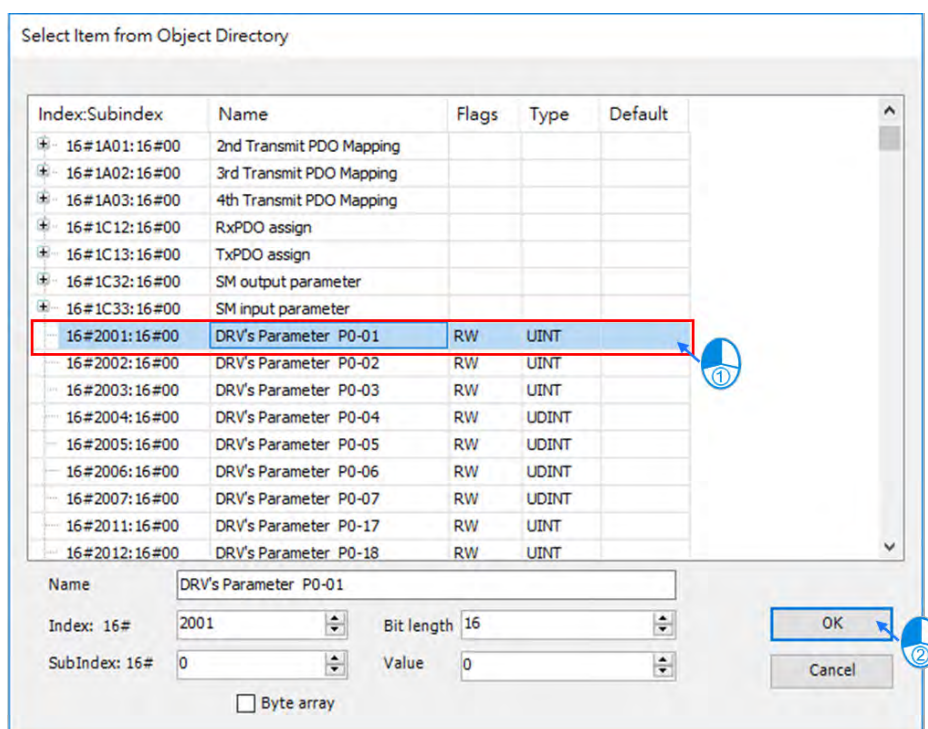
● Startup Parameters

EtherCAT 通讯建立完成后主站将对从站下载表格中的所有参数，列表中会带出 ESI 文件已经默认定义的指令，用户可视需求增减或修改。

功能	说明
Add	新增参数
Edit	编辑参数
Move Up	将选择参数往上移动一行
Move Down	将选择参数往下移动一行



单击「Add」按钮后，将出现「Select Item Object Directory」窗口，先从目录中单击要写入的参数。单击「OK」后便会加入到指令列表中。

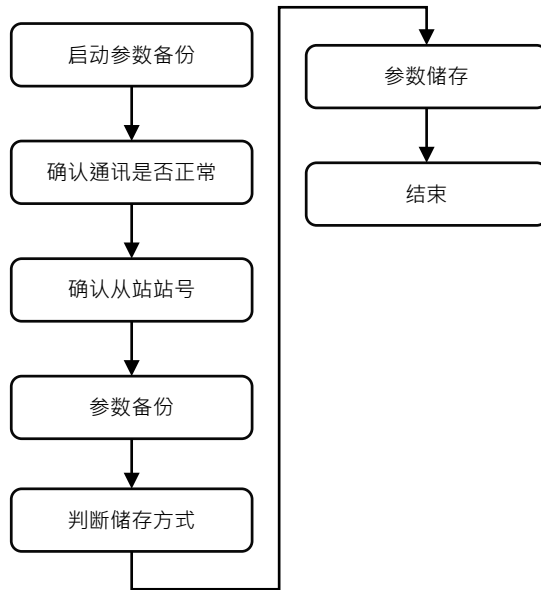


9.1.4 EtherCAT 从站参数备份还原

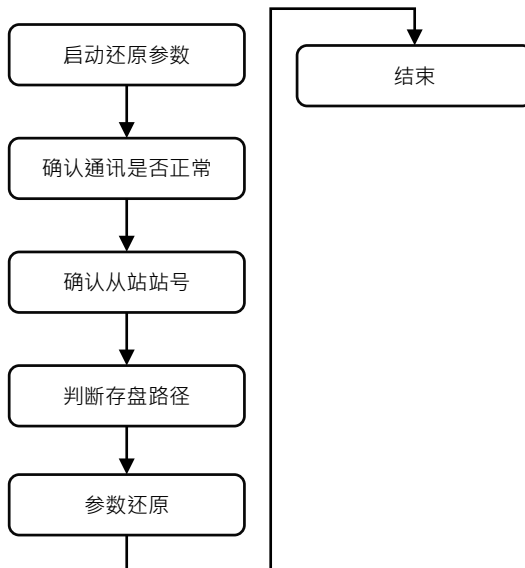
使用 EtherCAT 通讯时，台达有针对 ASDA 伺服驱动定制化参数保存功能，可方便用户针对下方从站所有参数备份还原功能。

9.1.4.1 参数备份流程说明

- 参数备份流程



- 参数还原流程

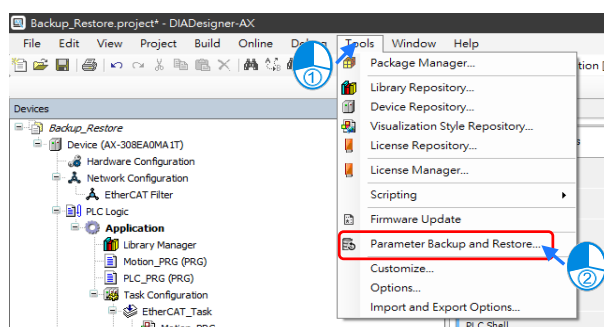


注：执行参数备份/还原时如其中任一轴有错误时，该从站则会跳过该轴继续备份/还原流程，待所有从站备份/还原完成后，软件会一并显示所有失败轴相关信息。

9.1.4.2 备份还原介绍

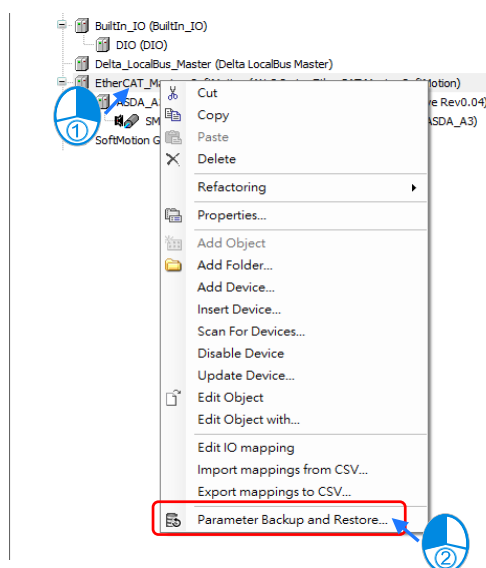
- 参数备份支持版本
 - AX-3 系列韧体：V1.0.2.0 以上
 - DIADesigner-AX 版本：V1.2 以上
 - 参数备份还原仅支持台达伺服 ASDA-A3-E、ASDA-B3-E
 - ◆ ASDA-A3-E 韧体版本：V11165 sub 92 以上
 - ◆ ASDA-B3-E 韧体版本：V10665 sub 75 以上
- 从站伺服备份内容

备份内容为伺服参数 P0~P4 (P0.001 与 P4.000 不备份还原)、及 P5.0003 + P5.0008 ~P5.0009 + P5.0020 ~P5.0030 + P6.0000 ~P6.0001 参数内容
- 参数备份还原画面开启
 - 方法一

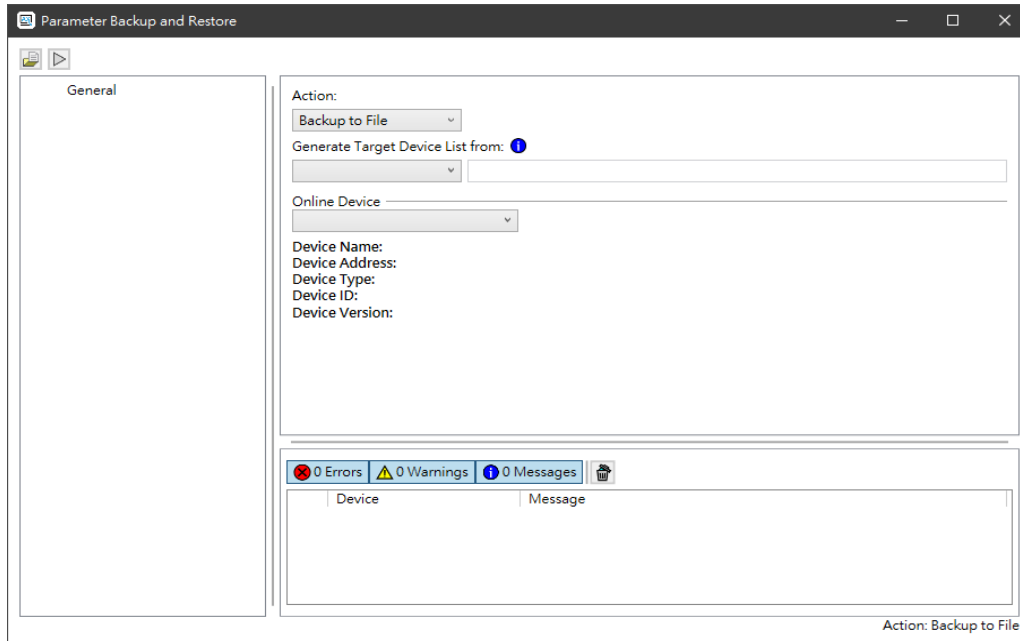




注：使用此方法在 Generate Target Device List from 的内容无法选择 Current Project 方式。

- 方法二



● 参数备份还原画面



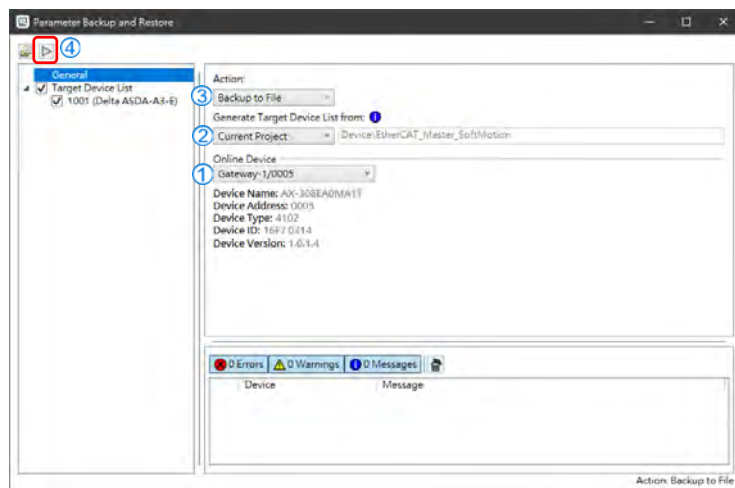
名称	功能
Online Device	选择欲连接 Device
Generate Target Device List from	选择 EtherCAT 树形图 - Archive File → 文件 EtherCAT 拓扑图 - Current Project → 目前项目中的 EtherCAT 拓扑图 - Online Topology → 在线的 EtherCAT 拓扑图
Action	选择功能 - Backup to File → 参数备份至文件 - Backup to SD Card → 参数备份文件至外部 SD 卡中 - Restore from File → 从文件回复参数 - Restore from SD Card → 从 SD 卡文件回复参数
	储存目前 EtherCAT 拓扑图 (Archive File)
	执行备份/还原功能

● 备份至外部 SD 卡路徑

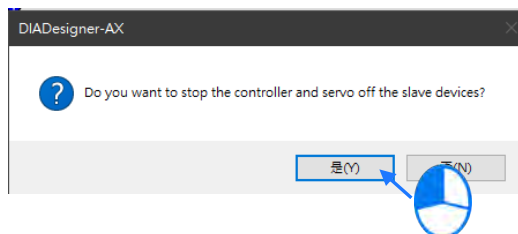
外部 SD 卡路徑：/PLC CARD/AX_/SysDup/ECAT/BackupRestore/ (其中的_为机种系列，例如读取到机种名称 AX-3 则为 AX3)

9.1.4.2.1 参数备份操作

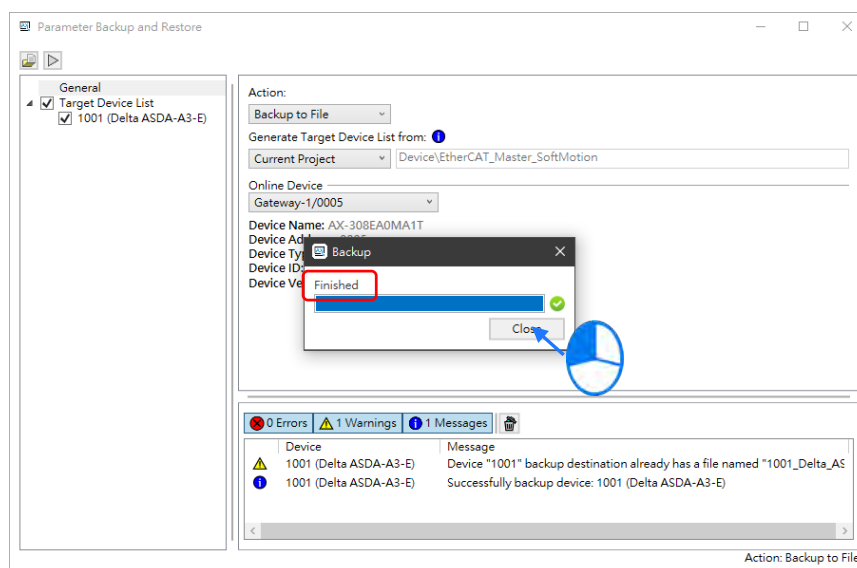
- 参数备份
 - ① 选择 Device
 - ② 设定目前项目的 EtherCAT 拓扑图
 - ③ 选择进行动作为备份
 - ④ 执行备份动作



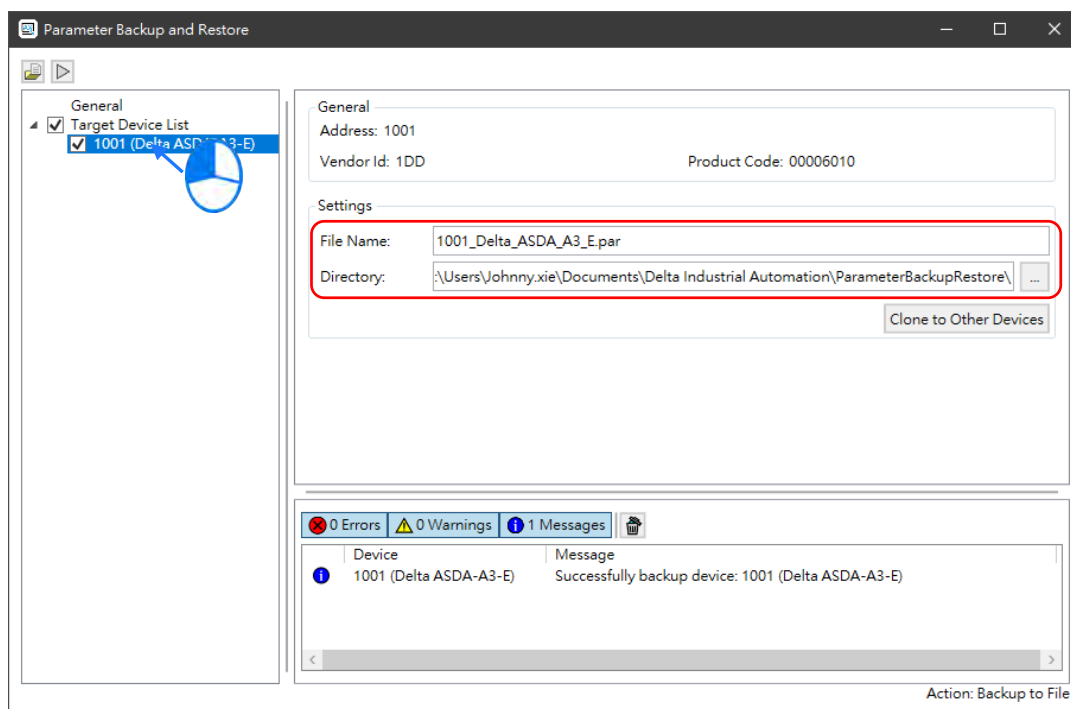
- 将 PLC 状态切换到 Stop



- 参数备份完成后，单击 Close



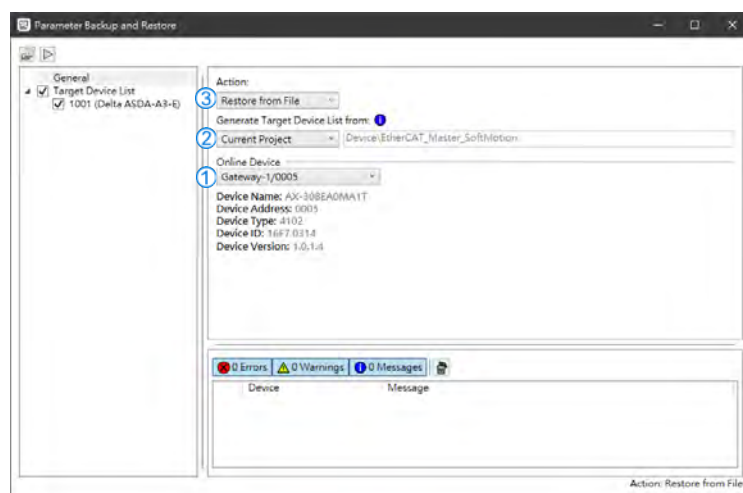
● 参数备份路径



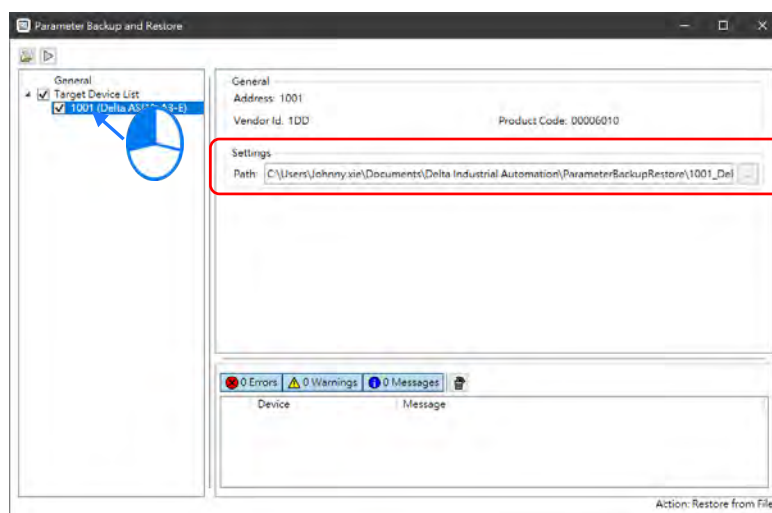
名称	功能
File Name	设定参数备份名称
Directory	设定参数备份存盘路径
Clone to Other Devices	修改所有从站参数备份存盘路径

9.1.4.2.2 参数还原操作

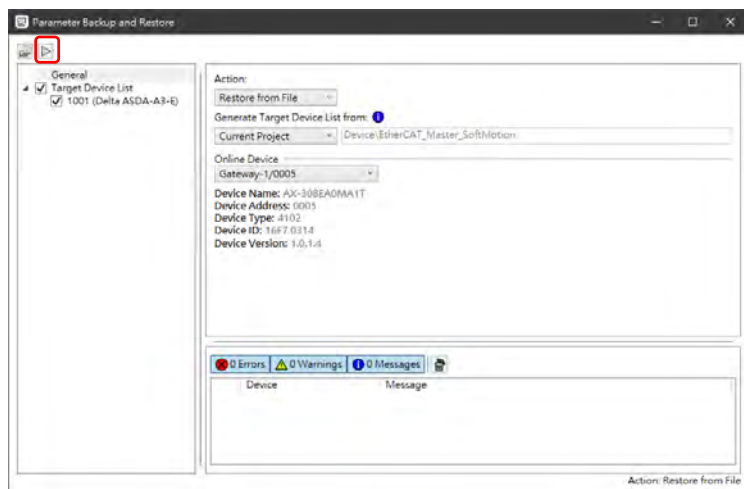
- 参数还原
 - ① 选择 Device
 - ② 设定目前项目的 EtherCAT 拓扑图
 - ③ 选择进行动作作为还原



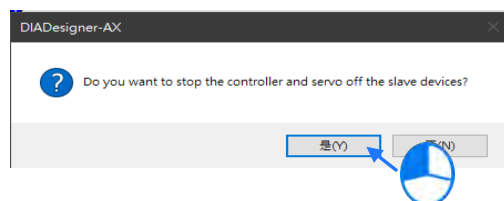
- 单击从站，可设定欲还原参数文件



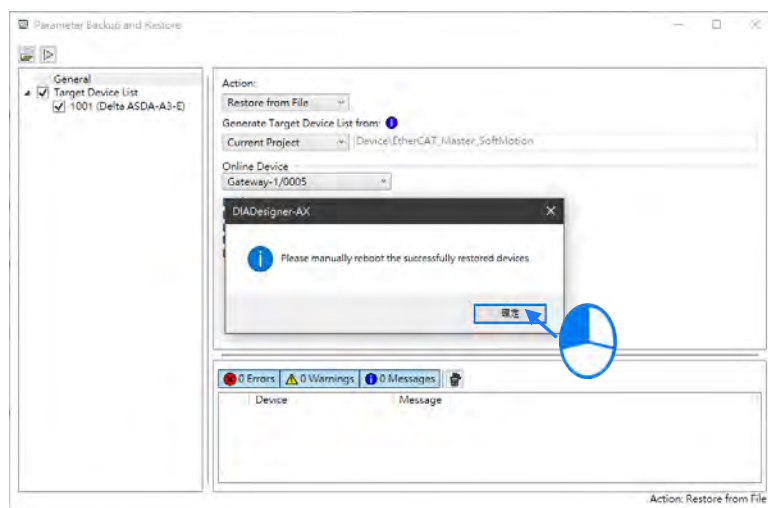
- 启动参数还原功能



- 将 PLC 状态切换到 Stop



- 参数还原完成后单击确定并将从站断电重启



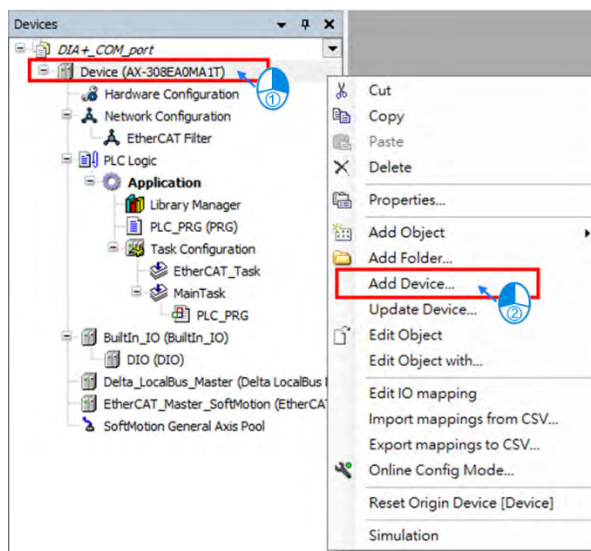
9.2 串行 (Serial) 通讯

9.2.1 Modbus Serial Port 元件

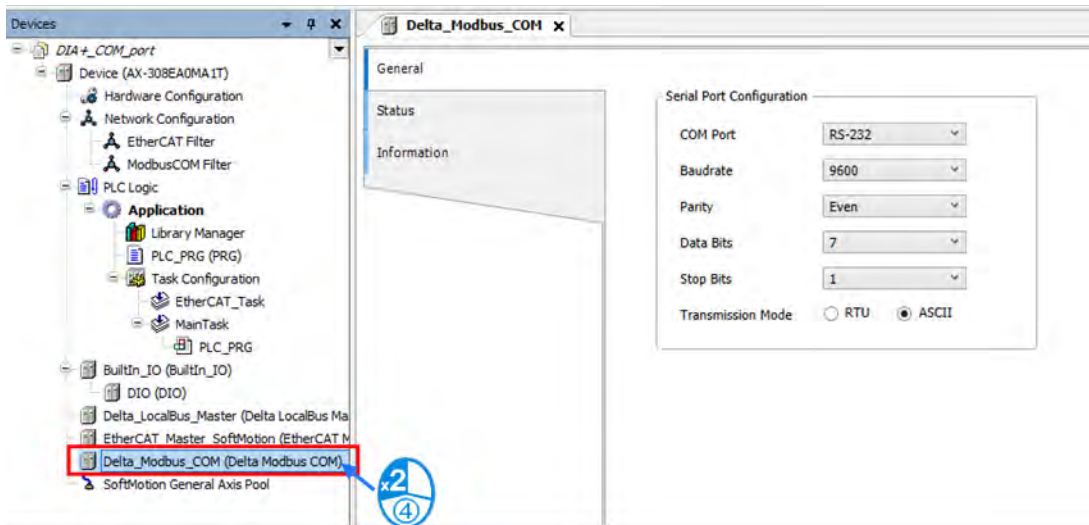
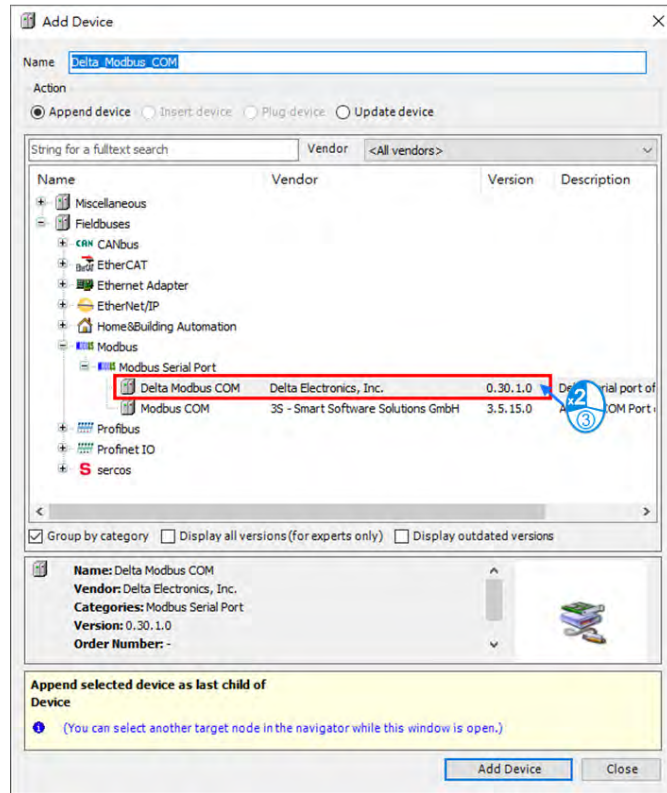
用户欲改变控制串行通讯相关参数时，则必须新增 Modbus Serial Port 方可调整。AX-3 系列控制器支持 1 个 RS-232 及 1 个 RS-485 通讯界面，需分别建立两个独立的 Modbus Serial Port。下面为 Modbus Serial Port 建立步骤。

9.2.1.1 Modbus Serial Port 元件建立

1. 单击 Device (AX-308EA0MA1T) 右键→选择 Add Device



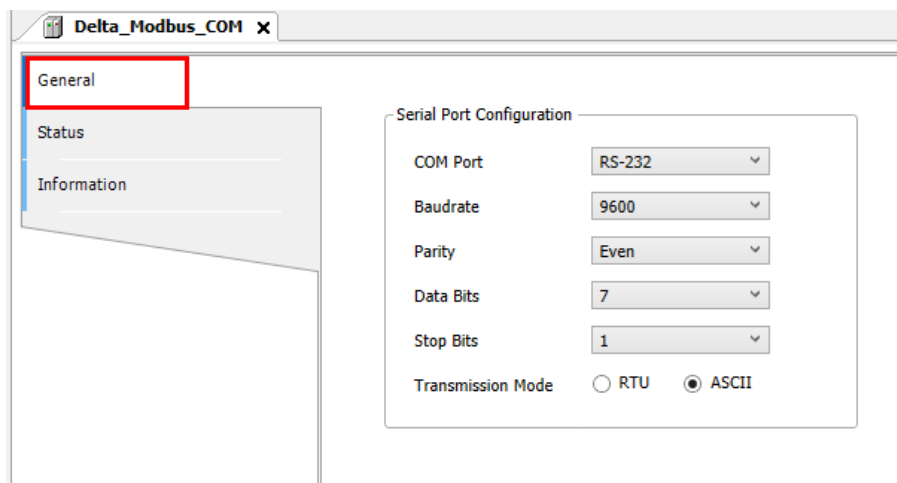
2. 从 Add Device 中选择，Modbus→Modbus Serial Port→Delta Modbus COM



9.2.1.2 Modbus Serial Port 组件页面说明

- **General**

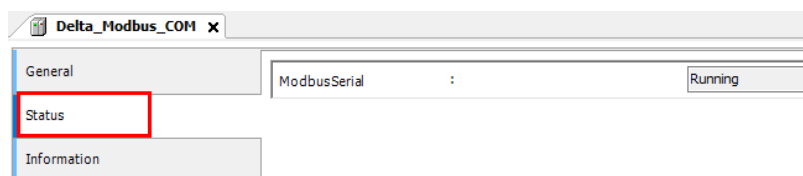
用户可以在此页面进行控制器 Serial Port 参数设定，如选择 COM Port (RS-232/RS-485) 及通讯格式设定。



标题	内容	
COM Port	通讯接口	RS-232/RS-485
Baudrate	波特率	9600/19200/38400/57600/115200
Parity	同位	None/Odd/Event
Data Bits	数据长度	7/8 (选择 RTU 通讯格式时，长度须为 8)
Stop Bits	停止位	1 bit/2bits
Transmission Mode	传输模式	RTU/ASCII

- **Status**

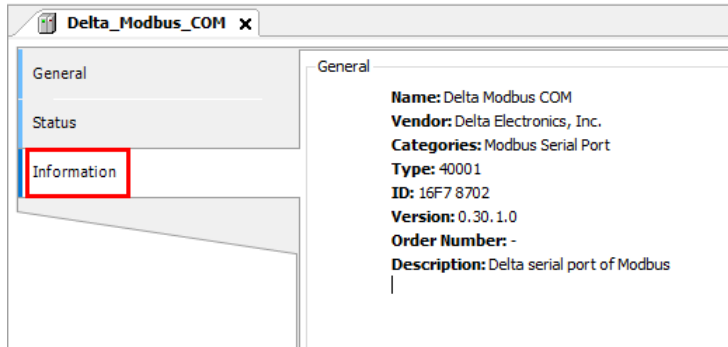
用户可在此页面查询 Delta Modbus COM 运行状态，如 Running、Stopped 与其它特定说明，帮助用户了解当前运作情形。



标题	说明
ModbusSerial	控制器串行通讯状态

● Information

此页面显示当前 Delta Modbus COM 相关信息，包含了制造商、产品相关手册、韧体版本等信息...

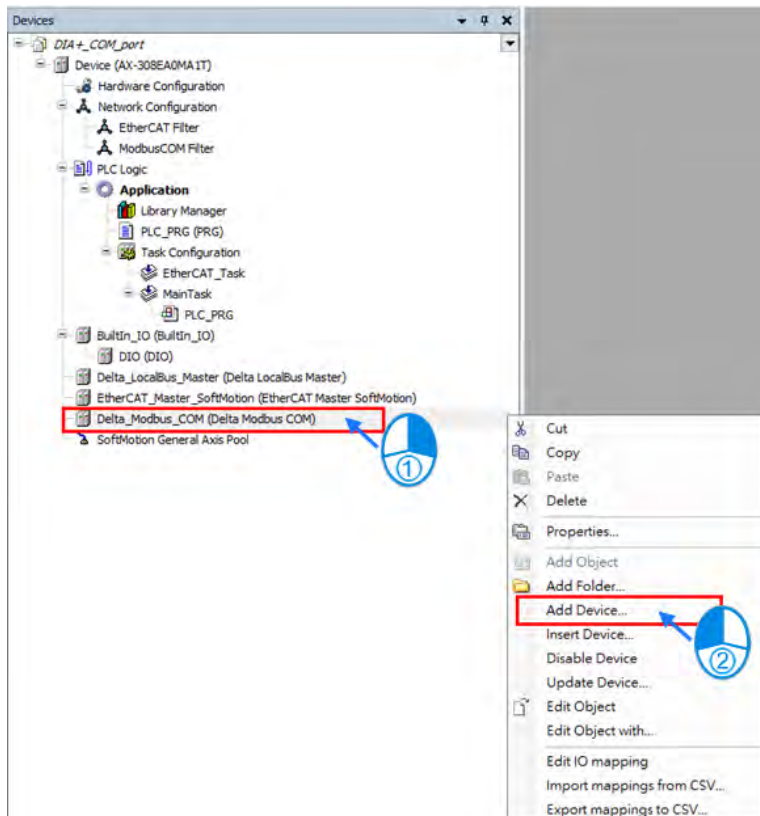


9.2.2 Modbus Serial Master

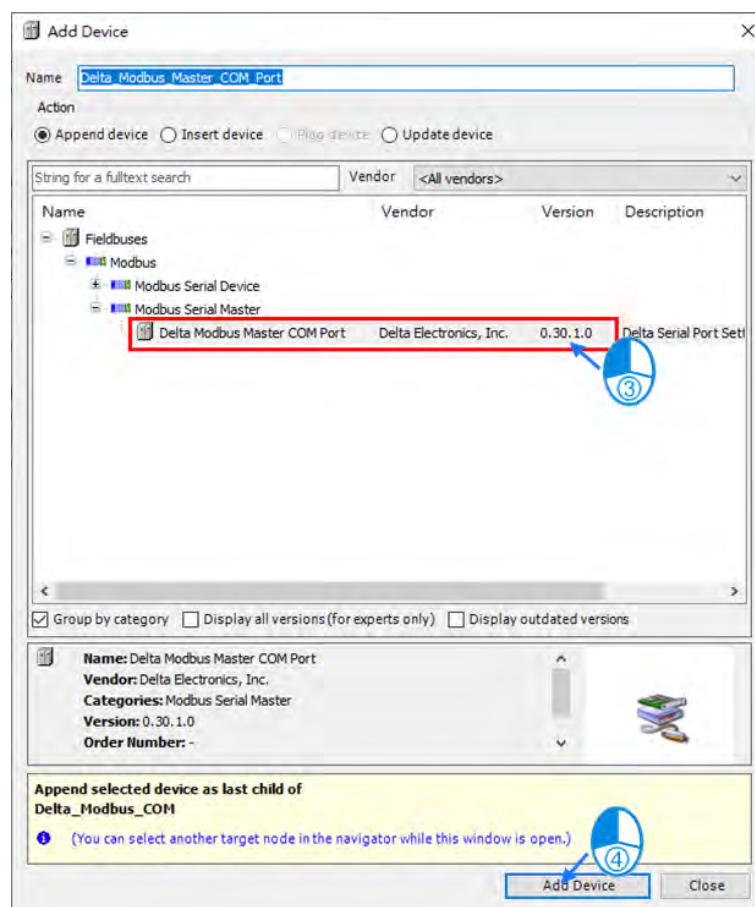
AX-3 系列控制器除了提供标准 Modbus 通讯协议之外，更进一步的针对台达 PLC 进行控制器内部装置转换 (X、M、D 装置...)，可免去用户查表动作，在使用上如 AX-3 系列控制器要担任 Modbus Serial Master 时，则须先新增 Modbus Master COM port 后，再增加 Modbus Slave COM port 并进行设定，请参考下列 Modbus Serial Master 建立步骤。

9.2.2.1 Modbus Serial Master 组件建立

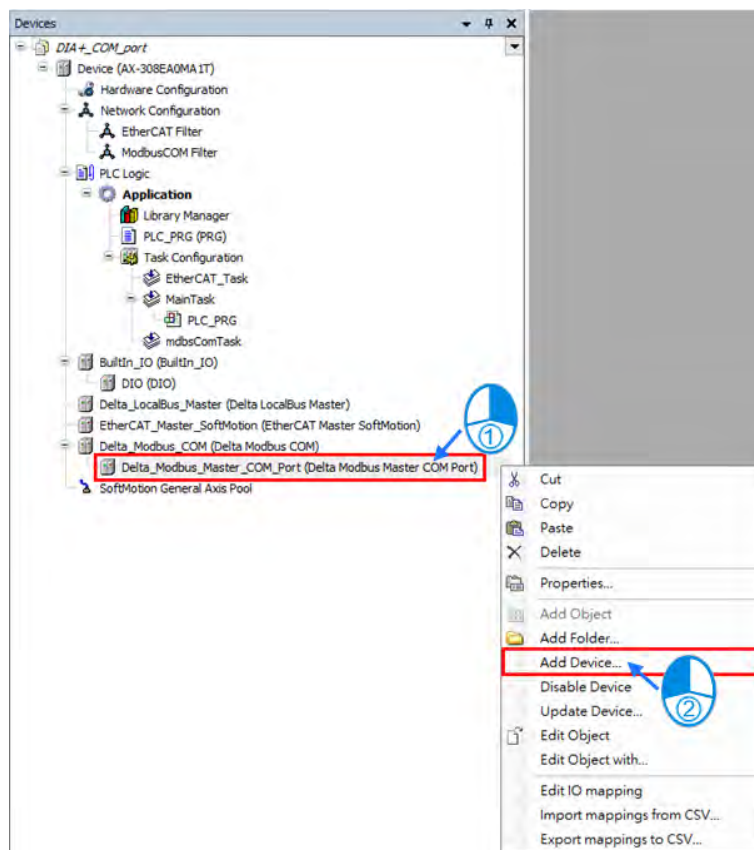
1. 单击树状结构中，Delta_Modbus_COM 单击鼠标右键→选择 Add Device 开启选择窗口。



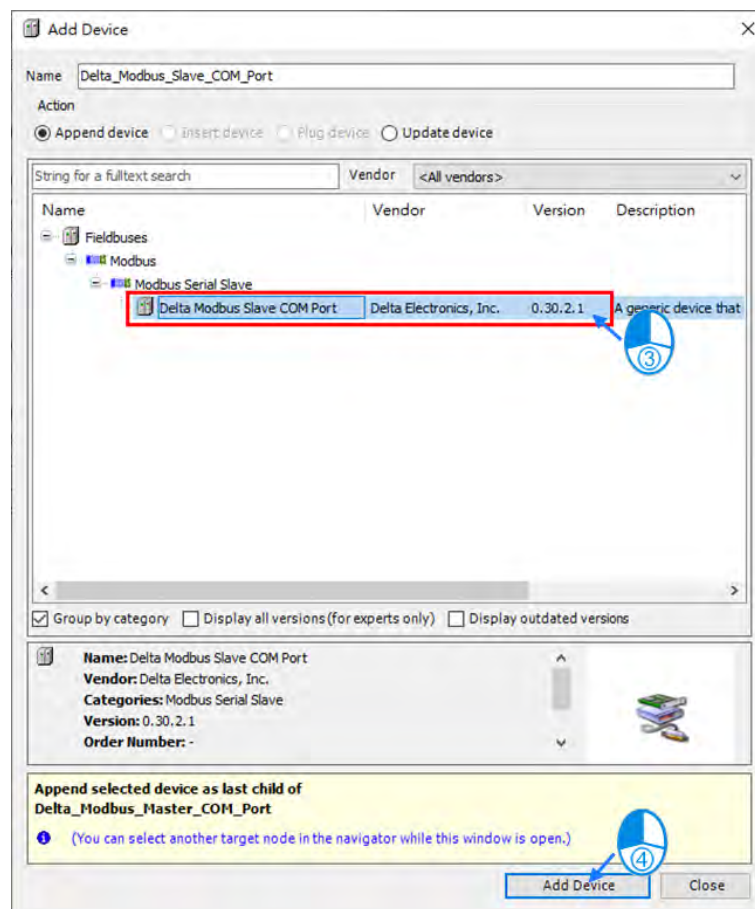
2. 选择 Delta Modbus Master COM port 鼠标双击加入。(Fieldbuses→Modbus→Modbus Serial Master→Delta Modbus Master COM Port)



3. 选择 Delta Modbus Master COM port 单击鼠标右键→选择 Add Device 开启选择窗口。

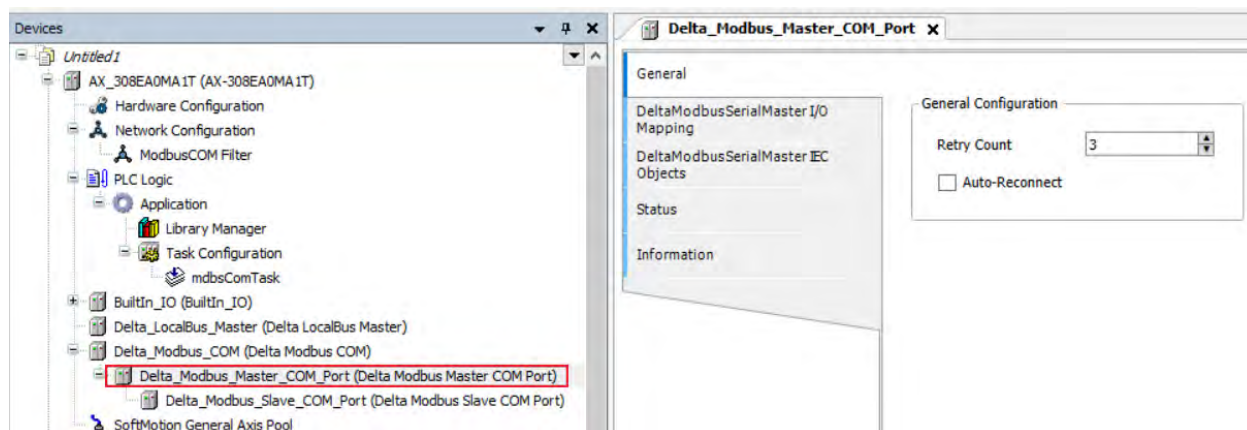


4. 选择 Delta Modbus Slave COM port 鼠标双击加入。(Fieldbuses→Modbus→Modbus Serial Slave→Delta Modbus Slave COM Port)



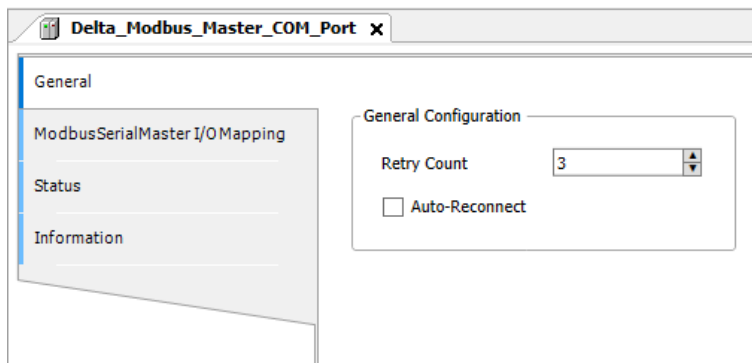
9.2.2.2 Modbus Serial Master 设定说明

单击树状结构中已建立之 Modbus Master COM Port，鼠标单击两下右侧窗口可进入设定页面。



● **General**

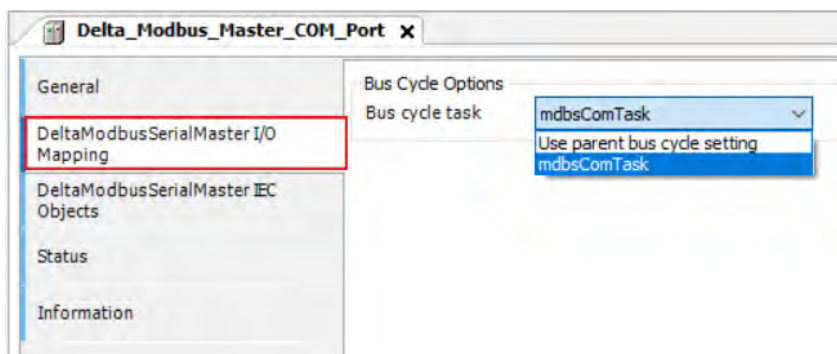
为 Delta Modbus Master COM Port 基本设定



标题	说明
Retry Count	设定 COM Port 重传次数。(默认次数 3 次)
Auto-Reconnect	当发生逾时或是错误时，则会自动进行重新联机。(默认不勾选)

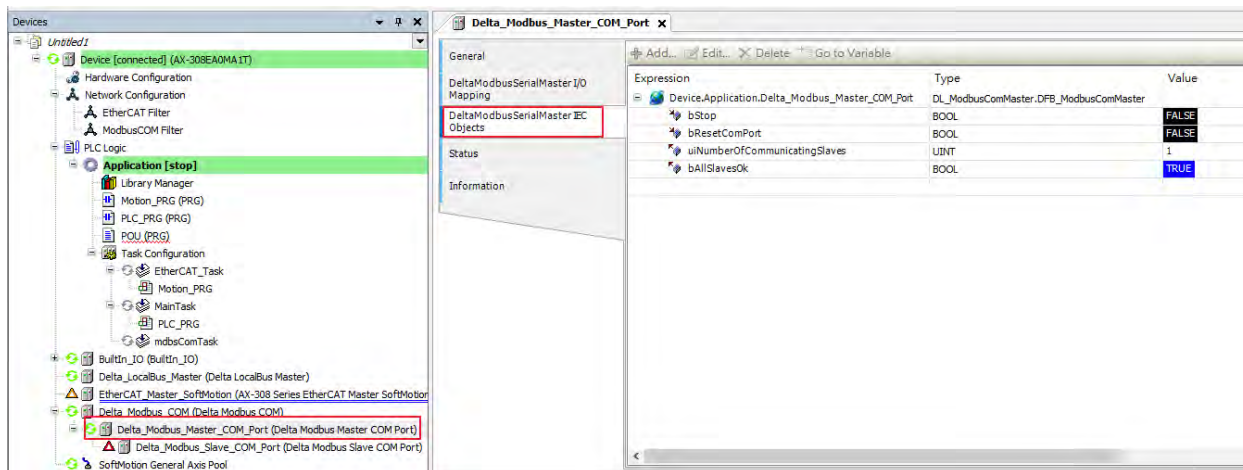
● **Delta Modbus Serial Master I/O Mapping**

用户可以于此页面设定 Modbus Serial Master 的 Bus Cycle Task，其详细说明请参考第 4.2.1.6 节 PLC Settings。



● **Delta Modbus Serial Master IEC Objects**

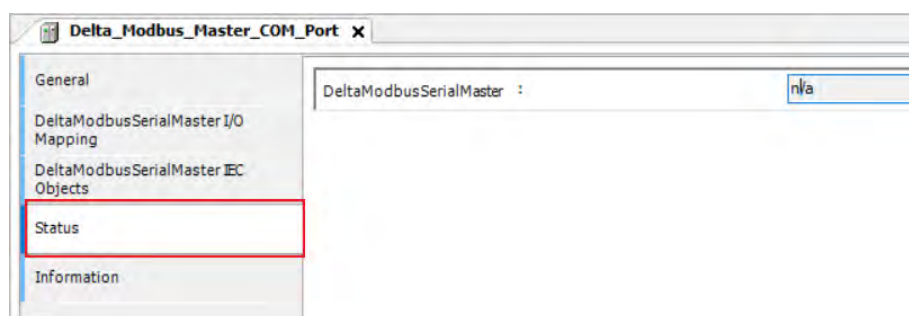
用户可以通过此页面对 Modbus Serial Master 状态进行判断。



标题	说明
bStop	停止向从站发送任何新请求
bResetComPort	重置 COM 通讯端口
uiNumberOfCommunicatingSlaves	显示正在通讯的从站的数量。
bAllSlavesOk	显示从站通讯状态

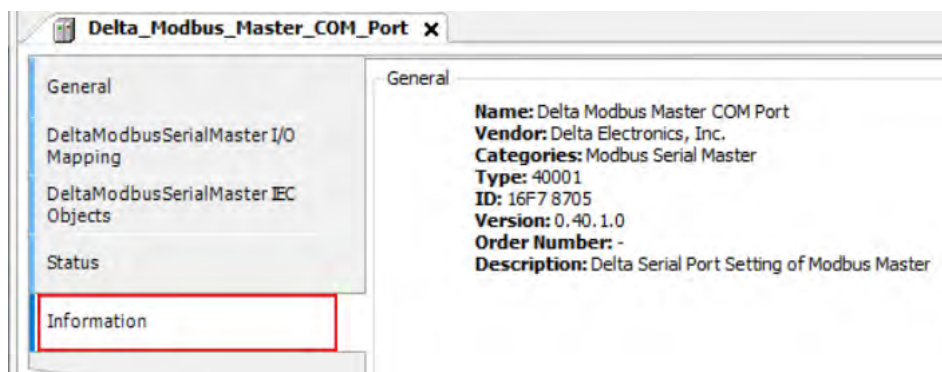
● Status

用户可由此页面确认目前 Modbus Serial Master 运行状态。例如：“Running”、“Stopped”、“specific diagnostic”信息。



● Information

Modbus Serial Master 相关信息，包含了制造商、产品相关手册、韧体版本等信息...

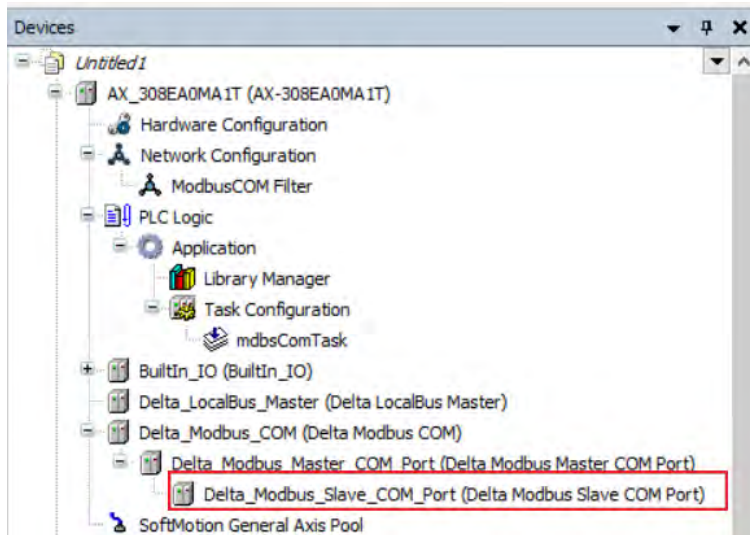


补充说明

- (a) 每一个 Modbus Serial Port 只能添加 1 个 Modbus Serial Master，而每个 Modbus Serial Master 能添加 32 个 Modbus Slave COM Port。
- (b) RS-232 虽可以加 32 个 Modbus Slave COM Port，但是只有第一个 Modbus Slave COM Port 会运作，其余 31 个 Modbus Slave COM Port 不会运作，因为 RS-232 只支持 1 对 1 通讯。而 RS-485 则无此限制。

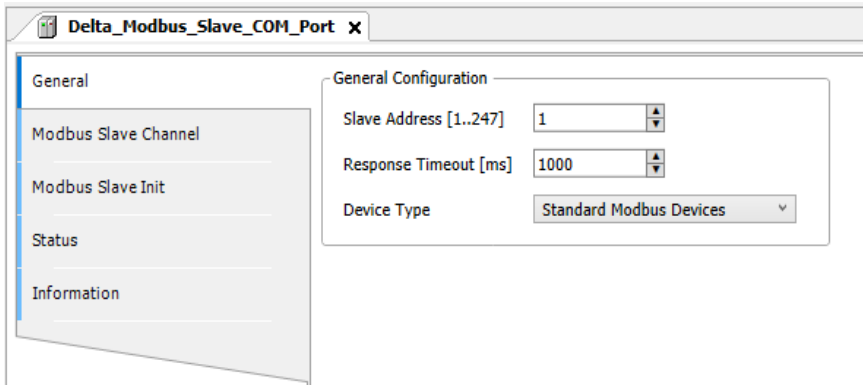
9.2.2.3 Modbus Serial Slave 设定说明

单击树状结构中已建立之 Delta Modbus，鼠标单击两下右侧窗口可进入设定页面。



- **General**

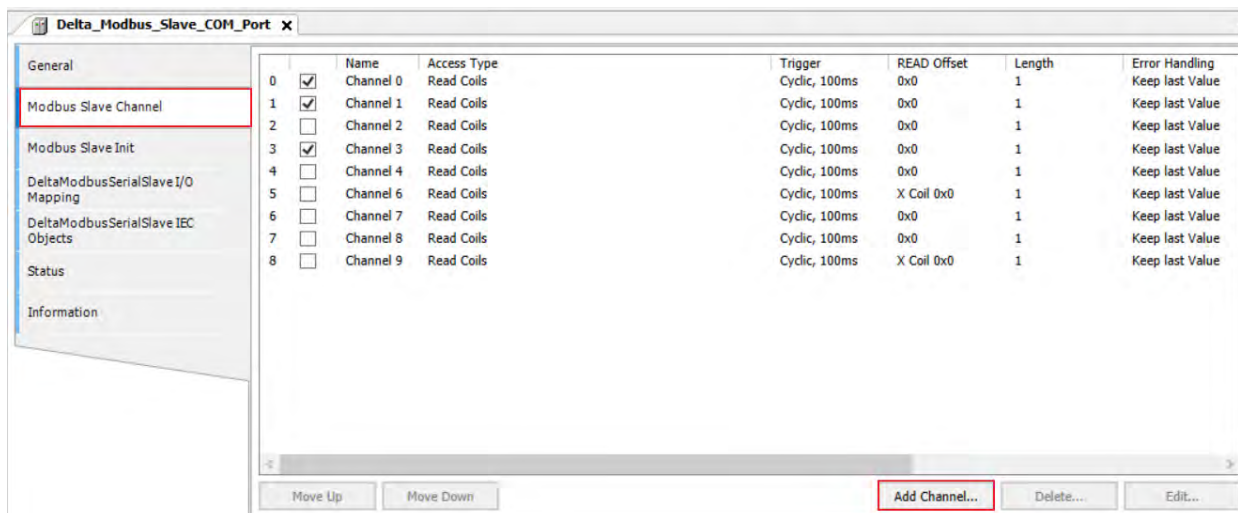
用户可以于此页面设定所欲连结的 Modbus Slave COM Port 相关信息如站号、响应逾时时间、通讯装置格式...



标题	说明
Slave Address	从站站号
Response Timeout	从站响应逾时时间
Device Type	从站类型，可选择标准 Modbus 通讯或是台达特定控制器。(如选择台达系列控制器，则会自动转换装置与 Modbus 地址)

- **Modbus Slave Channel**

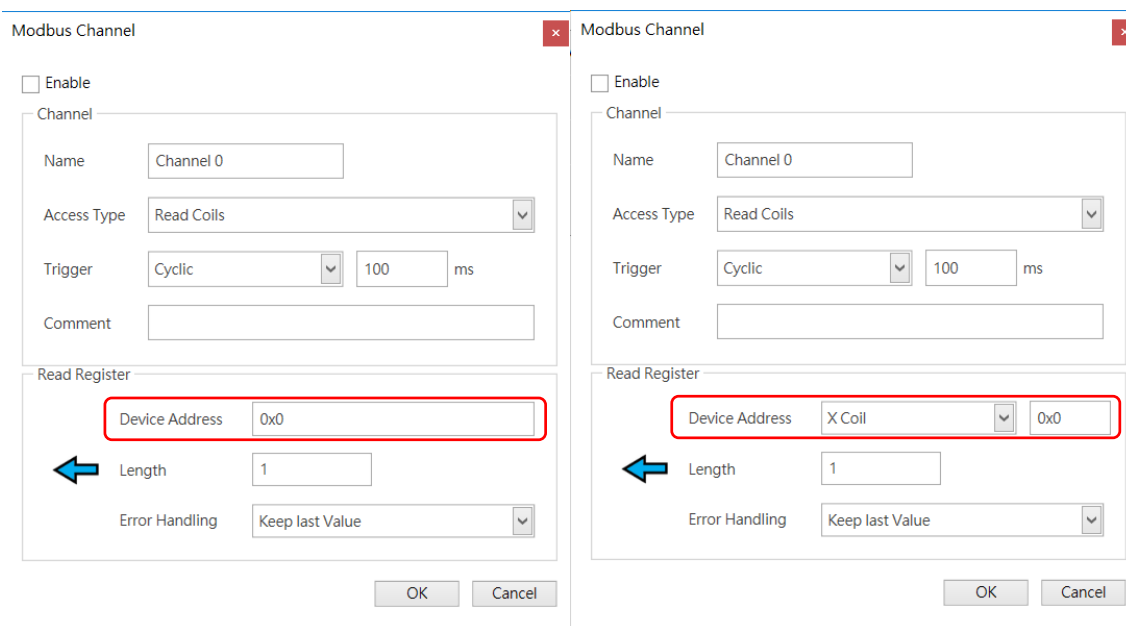
用户可以利用此表格定义从站通道，每通道代表一个 Modbus request 报文，目前每从站最多建立 10 个通道，AX-3 系列控制器会表格顺序依序发出 Modbus request 报文，且所有通道共同使用一条 Modbus 联机。



Modbus Channel 会随着 General 页面所设定的 Device Type 类型而有所不同，如果选择台达特定控制器，可直接选择对应之装置名称（D Register、M Coil、X Coil...等），可免去查表的动作。（如下图）

Device Type : Standard Modbus

Device Type : AH Series

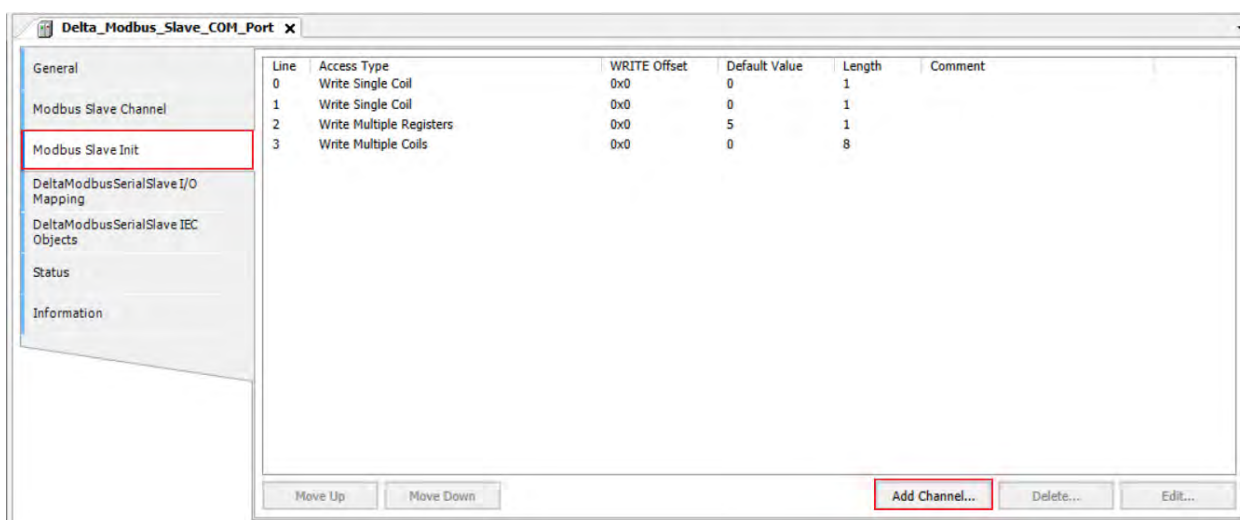


Item	Description	
Device Type	Standard Modbus Device	Delta Series Device
Enable	此通道是否生效	
Name	定义此通道名称	
Access Type	Modbus 通讯功能码： ● Read coils (0x01) ● Read discrete inputs (0x02) ● Read holding registers (0x03) ● Read input registers (0x04)	通道读写设定： ● Read coils ● Read registers ● Write coils ● Write registers

Item	Description	
Device Type	Standard Modbus Device	Delta Series Device
	<ul style="list-style-type: none"> ● Read single coil (0x05) ● Write single register (0x06) ● Write multiple coils (0x0F) ● Write multiple registers (0x10) ● Read/Write multiple registers (0x17) 	注：控制器会依照所读/写的装置类型选择相对应的 Modbus 功能码。
Trigger	<ul style="list-style-type: none"> ● Cyclic：依照设定的周期时间触发 ● Rising edge：通过布尔变量触发，此布尔变量定义于 I/O Mapping 页面。 ● Application：通过功能块「<i>DFB_ModbusComChannel</i>」触发。 	<ul style="list-style-type: none"> ● Cyclic：依照设定的周期时间触发 ● Rising edge：通过布尔变量触发，此布尔变量定义于 I/O Mapping 页面。 ● Application：通过功能块「<i>DFB_ModbusComChannel</i>」触发。
Comment	通道备注	
Device Address	Modbus 通讯地址	Delta register address (will be converted into Modbus protocol in the background)
Length	读取/写入的数据长度。(最大长度为 100 个 coils 与 100 个 registers)	Number of the register to be read/written to. (up to 256 coils and 100 registers)
Error Handling	当通讯发生异常时，Register 数据的设定动作。 <ul style="list-style-type: none"> ● Set To ZERO ● Keep last value 	

● **Modbus Slave Init**

用户可以利用此表格决定，是否要在 AX-3 系列控制器与从站成功建立 Modbus 联机之后，对从站的 Coils/Registers 写入初始值。



单击 **Add Channel** 可加入装置地址及其初始值设定，设定完成后单击 **OK**。下载设定后，此页面之设定值会被执行一次。即设定初始值完成。

Initialization Value

Access Type: Write Multiple Registers

Device Address: 0x0

Length: 1

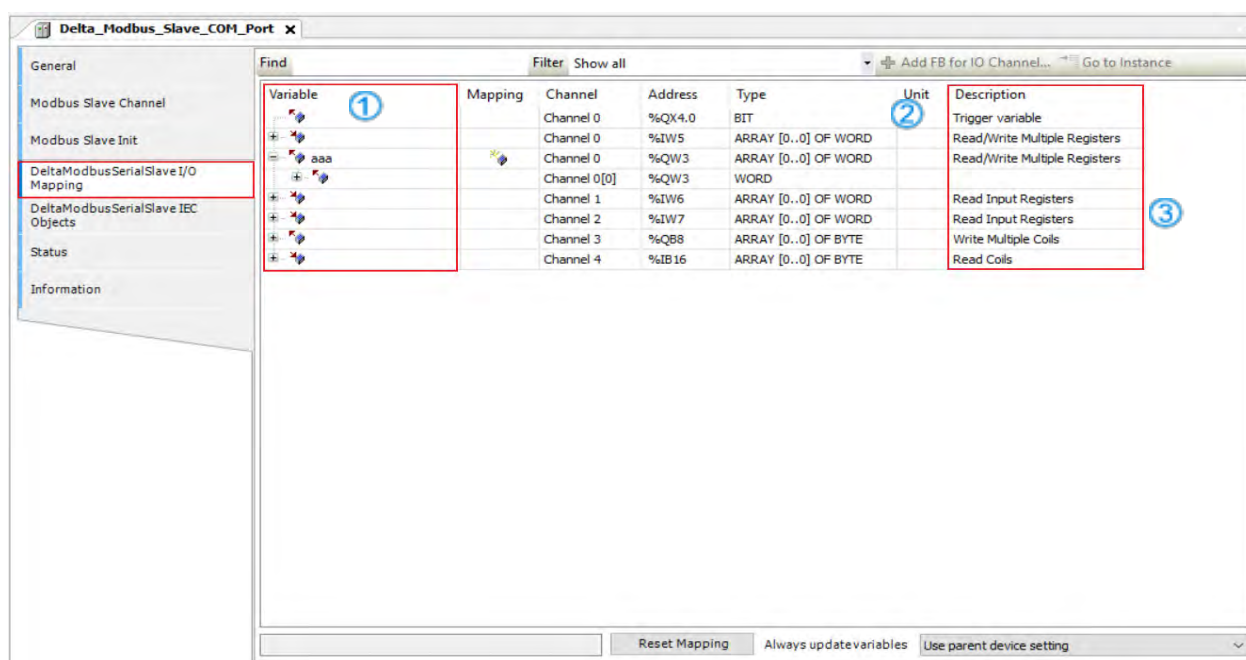
Initialization Value: 5

Comment:

OK Cancel

● Delta Modbus Generic Serial Slave I/O Mapping

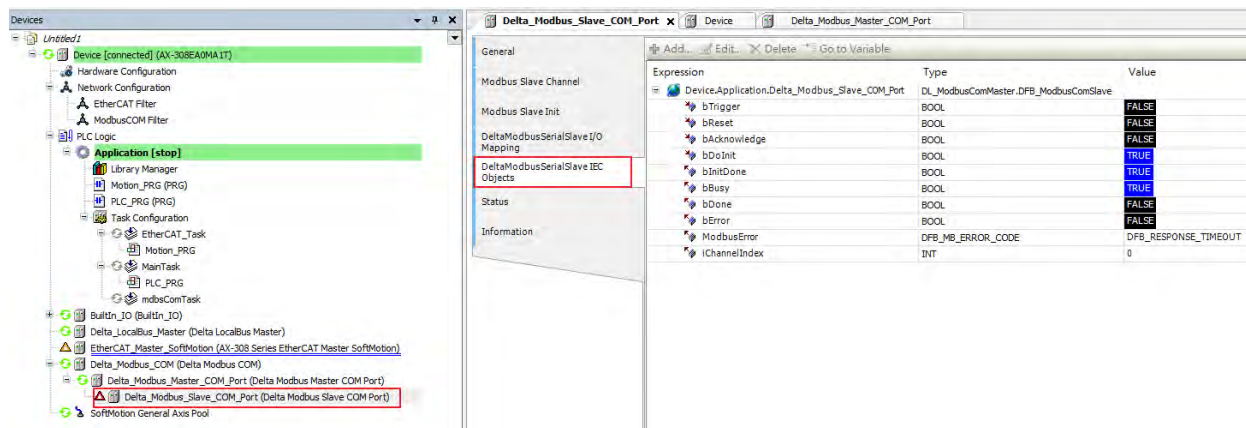
当用户在 Modbus Slave Channel 页面设定通道完毕后，在 **Modbus Generic Serial Slave I/O Mapping** 页面中则会自动产生所对应的 I/O 对应表，用户可在这页面里针对使用的装置进行变量配置与通道控制。



- ① : 用户可以于此字段将所有相关参数映像到用户自行变量
- ② : 此通道的触发布尔变量。(设定模式为 Rising edge 时，会自动产生此 Trigger 变量)
- ③ : 此通道读取/写入的控制器类型 Coils/Registers

● Delta Modbus Serial Slave IEC Objects

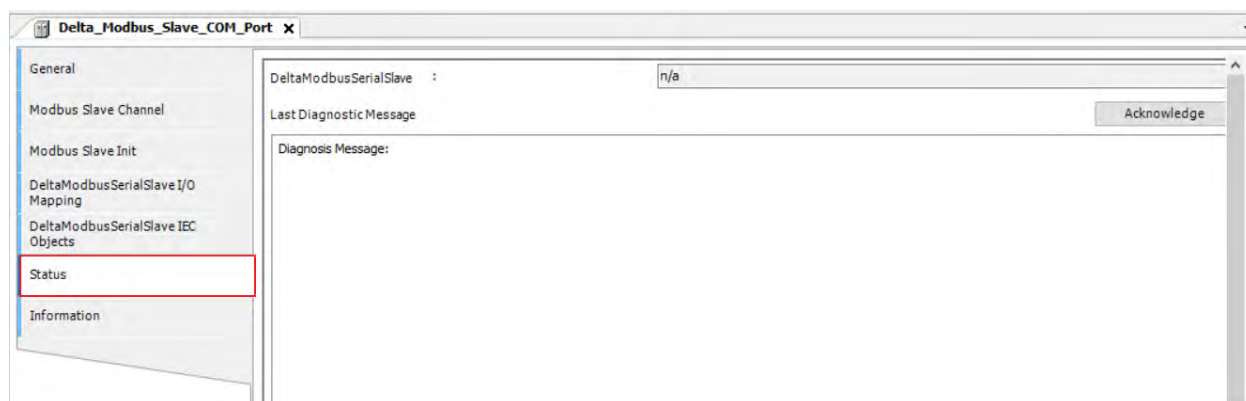
用户可以通过此页面对 Modbus Serial Slave 状态进行判断。



标题	说明
bTrigger	单次触发所有 Modbus channels。
bReset	重新联机并且重置 bError 和 modbusError。 *此功能只用于错误状态且没有勾选「Auto-Reconnect」
bAcknowledge	重新联机并且从上一次发生错误之 Modbus channels 接续开始执行数据交换。*此功能只用于错误状态且没有勾选「Auto-Reconnect」
bDoInit	执行从站初始值设定
bInitDone	从站初始值设定完毕
bBusy	此 Channel 写入/读取报文中
bDone	此 Channel 写入/读取报文完成
bError	此 Channel 写入/读取错误发生
ModbusError	记录发生的错误描述
iChannelIndex	正在执行的 channel 编号

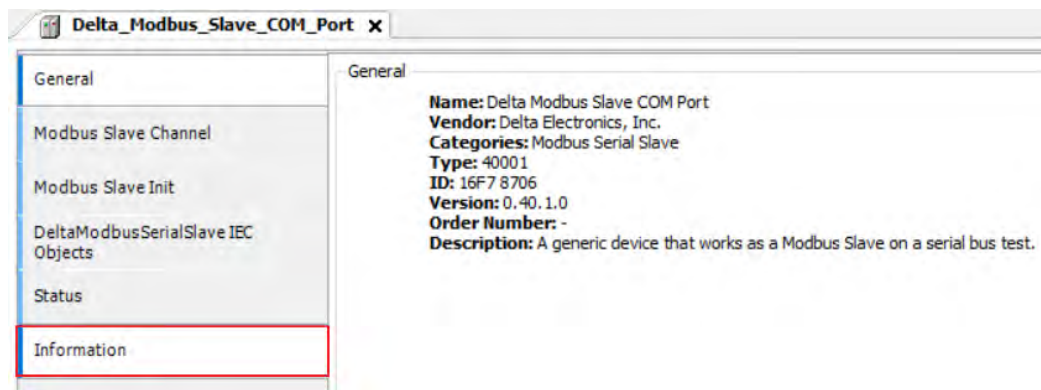
● Status

用户可于此页面查询 Modbus Slave COM Port 运行状态，如 Running、Stopped 与其它特定说明，帮助用户了解当前运作情形。



- **Information**

当前 Modbus Slave COM Port 相关信息，包含了制造商、产品相关手册、韧体版本等信息...

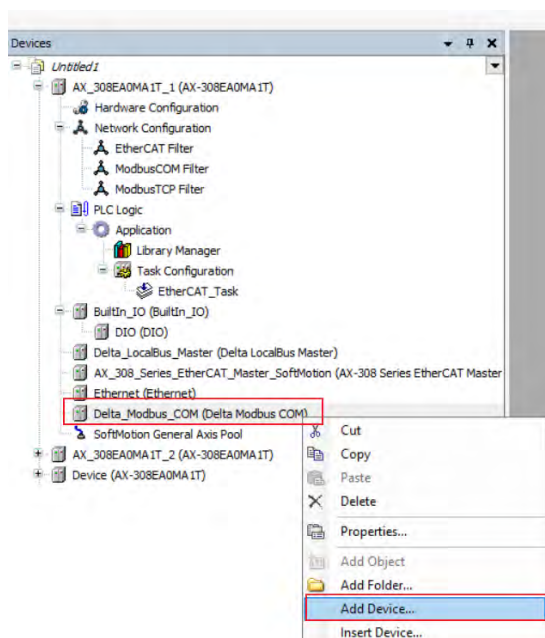


9.2.3 Modbus Serial Slave

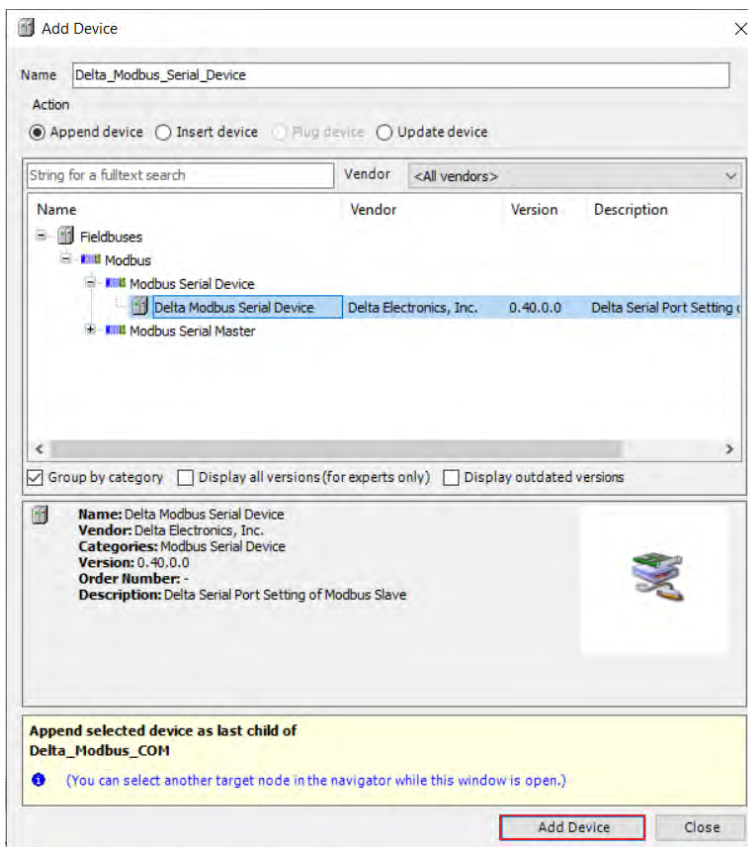
在使用上如 AX-3 系列控制器要担任 Modbus Serial Slave 时，则须新增 Modbus Serial Device，并设定其允许的访问的 Coils/Registers 区间，但如果 Modbus Serial Master 使用台达特定通讯协议，则不受区间限制，即可对所有 Coils/Registers 进行读/写，但仍需新增 Modbus Serial Device。

9.2.3.1 Modbus Serial Slave 组件建立

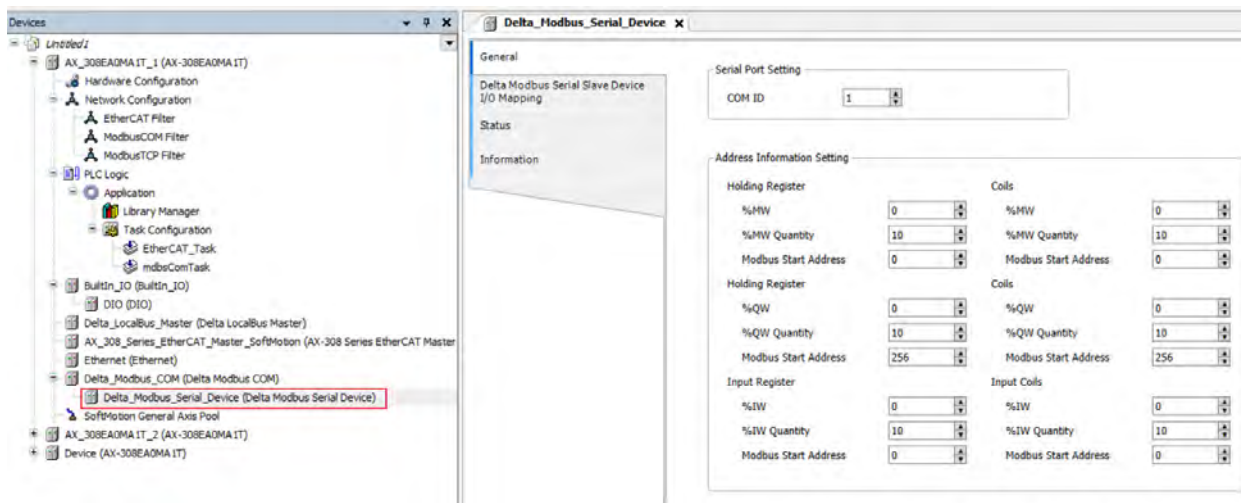
1. 单击 Delta Modbus COM 右键→Add Device→在 Add Device 窗口中选择 Delta Modbus Slave Device 加入。



2. 在 Add Device 窗口中选择 Delta Modbus Slave Device (Fieldbus→Modbus→Modbus Serial Device→Delta Modbus Serial Device) → 单击 Add Device 加入 Device。



3. 左侧树状结构中则会加入 Delta Modbus Slave Device，鼠标右键双击此 Device，则右方窗口可对此 Device 进行设定。



9.2.3.2 Modbus Serial Slave 设定说明

- **General**

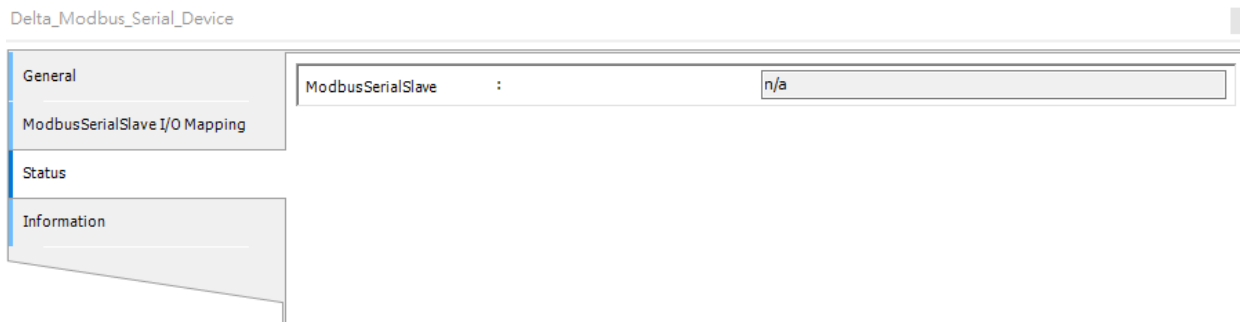
用户可以于此页面设定 AX-3 系列控制器的站号与允许访问的 Coils/Registers 区间，如 Modbus Serial Master 使用台达特定通讯协议，则不受限制可以访问所有区间。

- **Delta Modbus Serial Slave I/O Mapping**

用户可以于此页面设定 Modbus TCP Slave Device 的 Bus Cycle Task，其详细说明请参考第 4.2.1.6 节 PLC Settings。

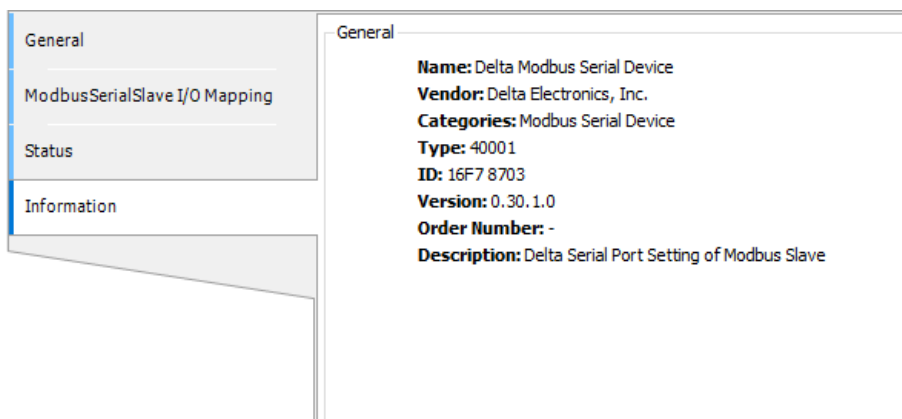
- **Status**

用户可于此页面查询 Modbus Serial Device 运行状态，如 Running、Stopped 与其它特定说明，帮助用户了解当前运作情形。



● **Information**

当前 Modbus TCP Slave Device 相关信息，包含了制造商、产品相关手册、韧体版本等信息...

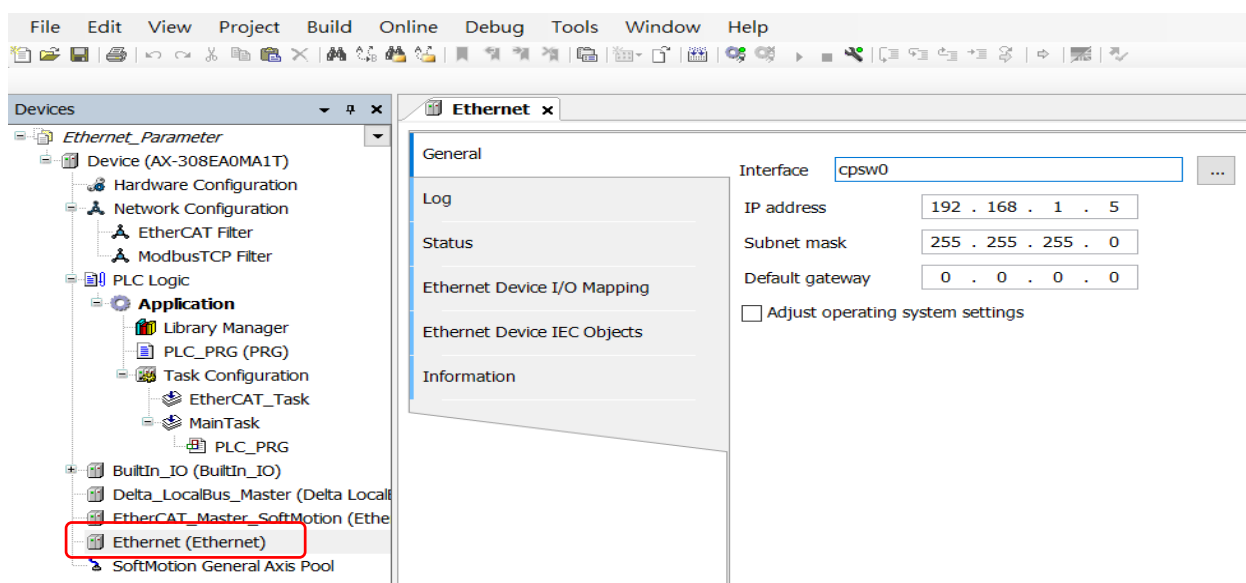


9.3 以太网 (Ethernet) 通讯

用户欲改变控制器网络相关参数时，则必须新增 Ethernet 组件方可调整，而且所有网络相关功能皆需于此组件下进行设定（如 Modbus TCP 与 EtherNet/IP...）。

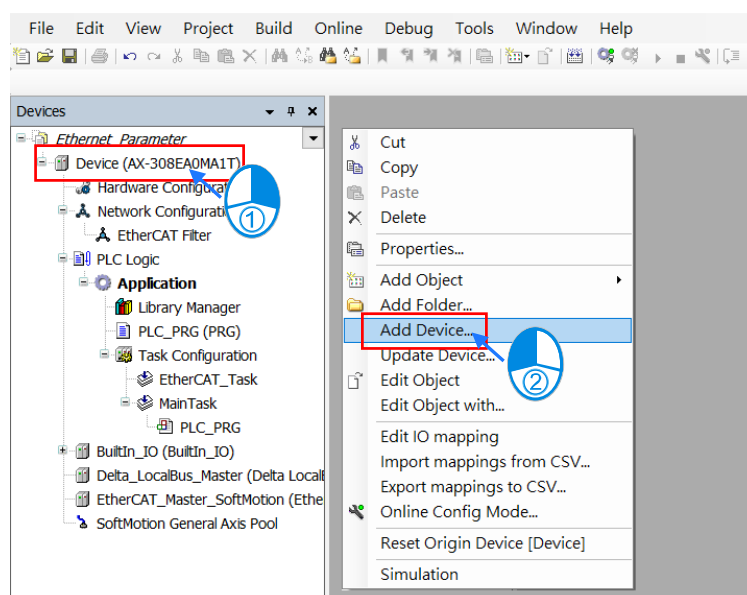
9.3.1 Ethernet 组件

- Ethernet 树形图

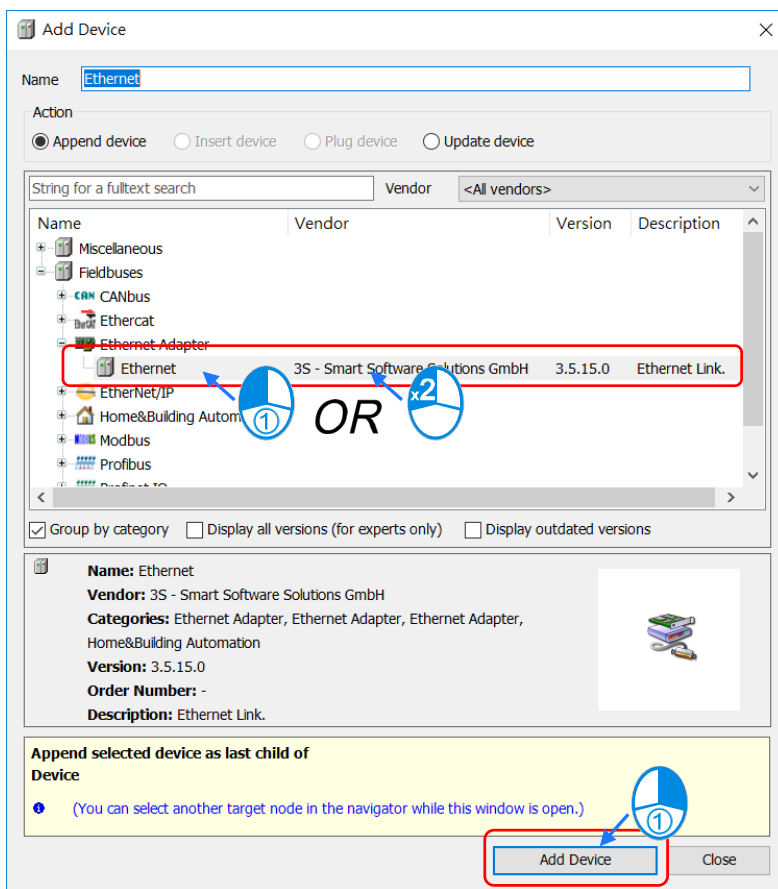


9.3.1.1 Ethernet 组件建立

1. 在树形图中 AX 系列控制器图标单击右键开页面，单击「Add Device」开启「Add Device」窗口

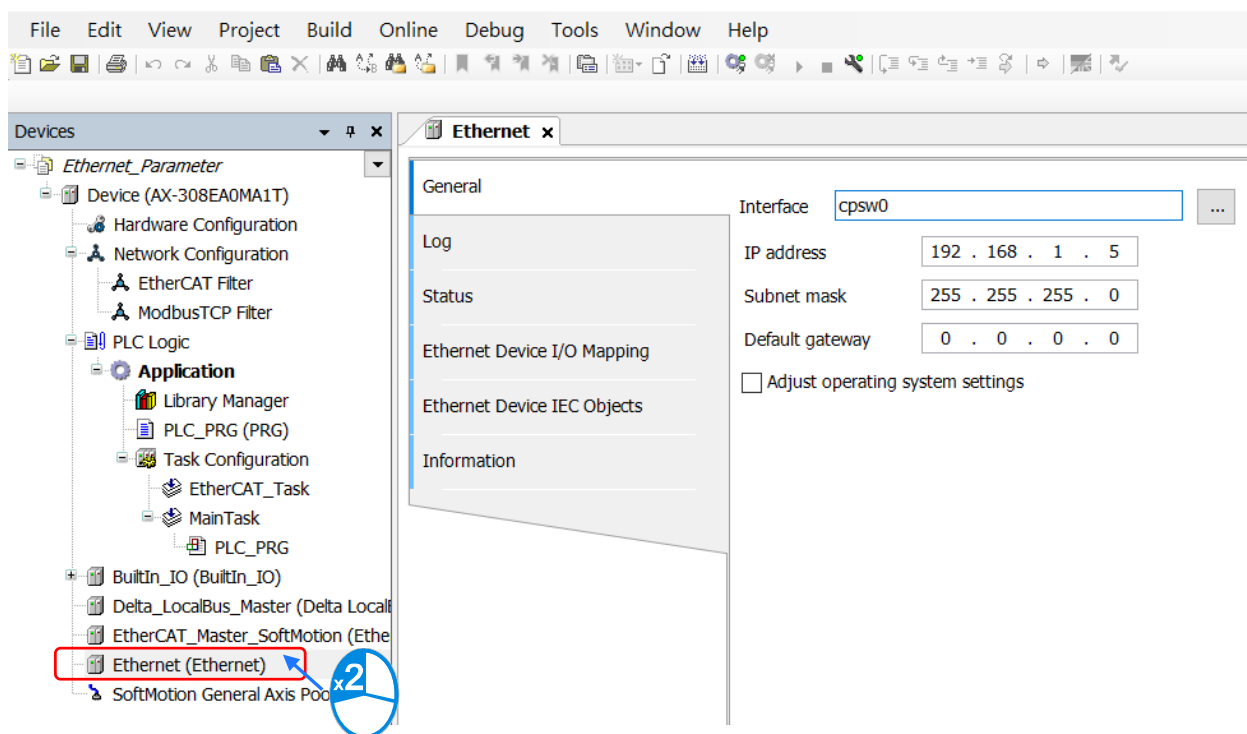


- 在 Ethernet 组件单击左键两下，或选择「Ethernet」组件再单击「Add Device」，即将 Ethernet 组件新增于树形图。

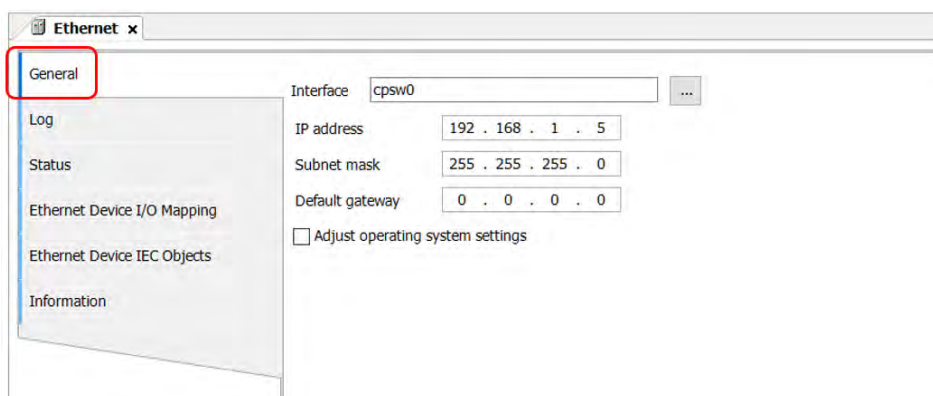


9.3.1.2 Ethernet 组件页面说明

在树形图中单击 Ethernet 组件两下，即开启设定页面，用户可以于此页面进行控制器网络参数设定，如选择控制器 Interface、IP Address...。



- **General**

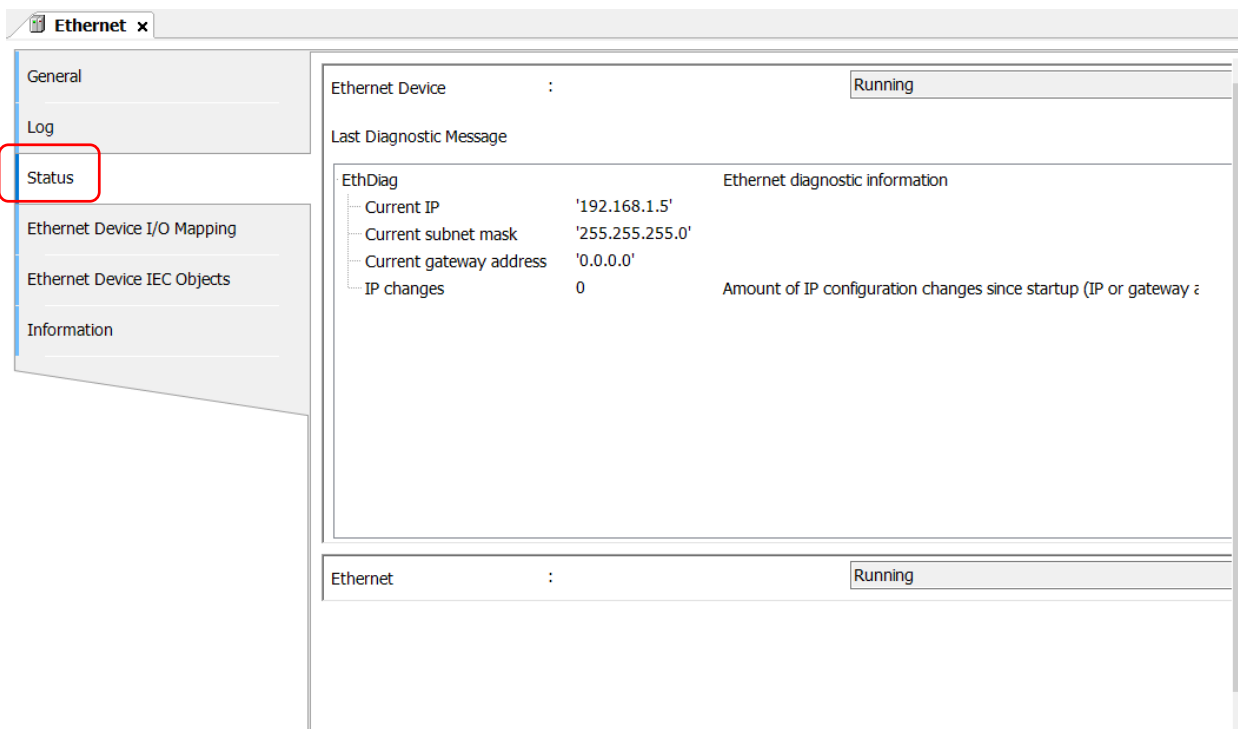


标题	说明
Interface	当前链接之控制器的通讯接口
IP address · Subnet mask · Standard gateway*	网络设定参数
Adjust operating system settings*	将当前设定参数覆写至控制器

*注：AX-3 系列主机描述文件（DDF）于 V1.0.1.0 版本以上（含），Device > System Setting 页面中，请参考第 4.2.1.11 节。

- **Status**

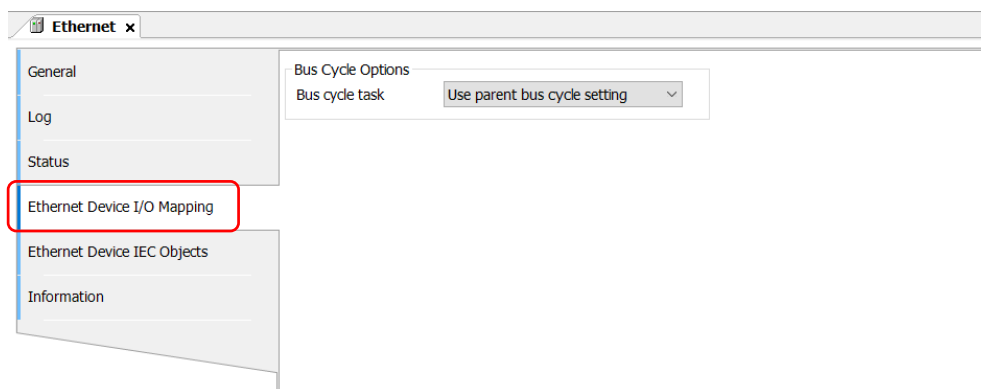
用户可在此页面查询 Ethernet Adapter Device 运行状态，如 Running、Stopped 与其它特定说明，帮助用户了解当前运作情形。



标题	说明
Ethernet Device	控制器网络状态
Last Diagnostic Message	网络诊断信息

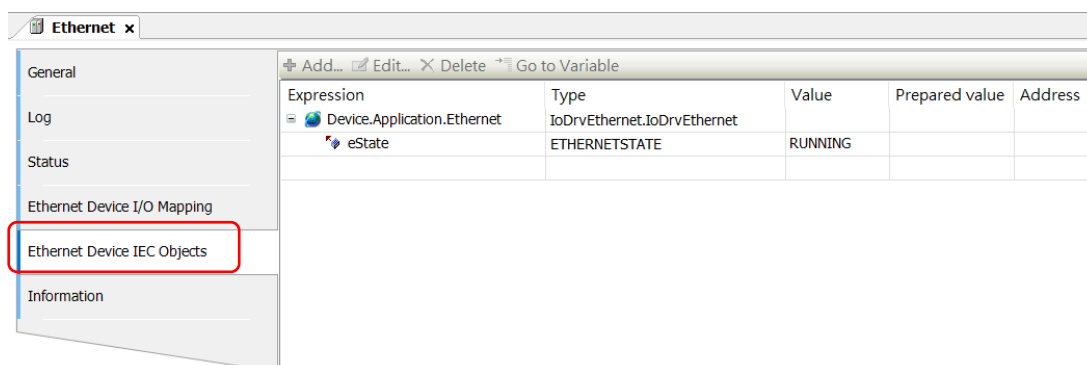
● **Ethernet Device I/O Mapping**

用户可以在此页面设定 Ethernet Adapter Device 的 Bus Cycle Task，其详细说明请参考第 4.2.1.6 节 PLC Settings。



● Ethernet Device IEC Objects

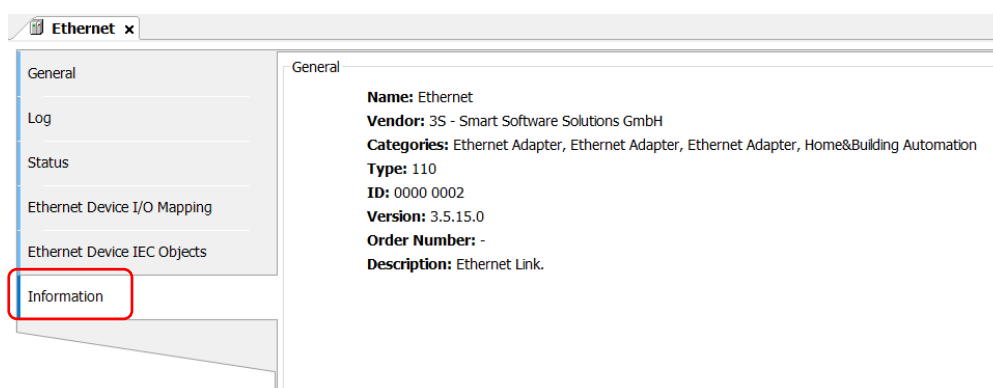
此页面显示 Ethernet Adapter Device 所定义的状态变量，用户可以通过状态变量来确认 Ethernet Adapter Device 当前状态。



Expression	Type	Value	Prepared value	Address
Device.Application.Ethernet eState	IoDrvEthernet.IoDrvEthernet ETHERNETSTATE	RUNNING		

● Information

此页面显示当前 Ethernet Adapter Device 相关信息，包含了制造商、产品相关手册、韧体版本等信息...

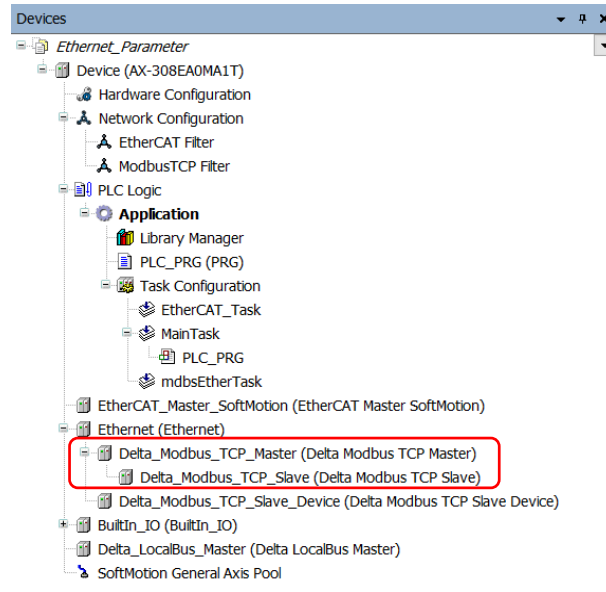


General
Name: Ethernet
Vendor: 3S - Smart Software Solutions GmbH
Categories: Ethernet Adapter, Ethernet Adapter, Ethernet Adapter, Home&Building Automation
Type: 110
ID: 0000 0002
Version: 3.5.15.0
Order Number: -
Description: Ethernet Link.

9.3.2 Modbus TCP Master

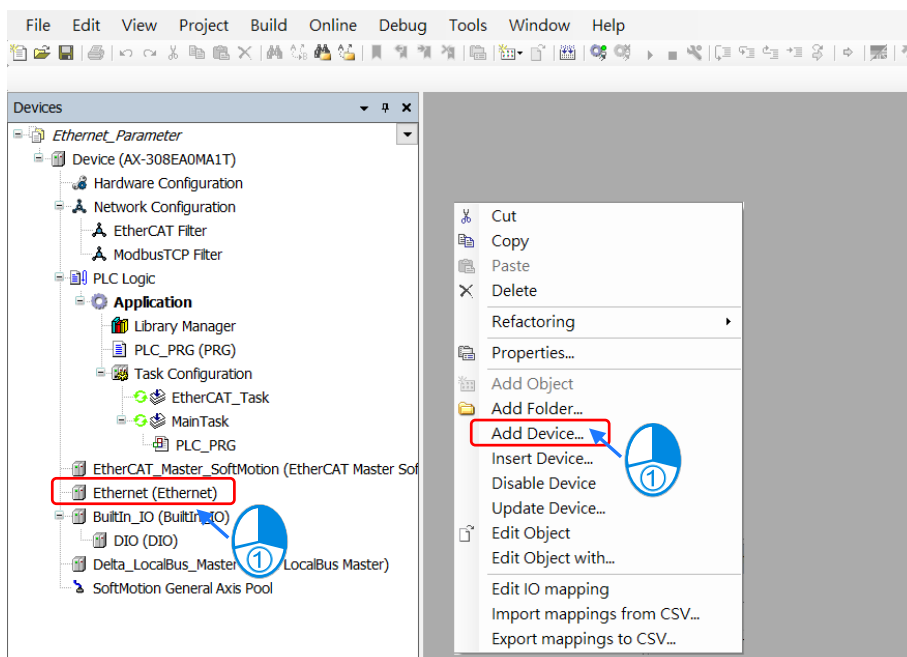
AX308E 系列控制器除了提供标准 Modbus 通讯协议之外，更进一步的针对台达 PLC 进行控制器内部装置转换(X、M、D 装置...)，可免去用户查表动作。在使用上如 AX308E 系列控制器要担任 Modbus TCP Master 时，则须先新增 Modbus TCP Master 后，再增加 Modbus TCP Slave 并进行设定，请参考下列 Modbus TCP Master 建立步骤。

● **Modbus TCP Master 树形图**

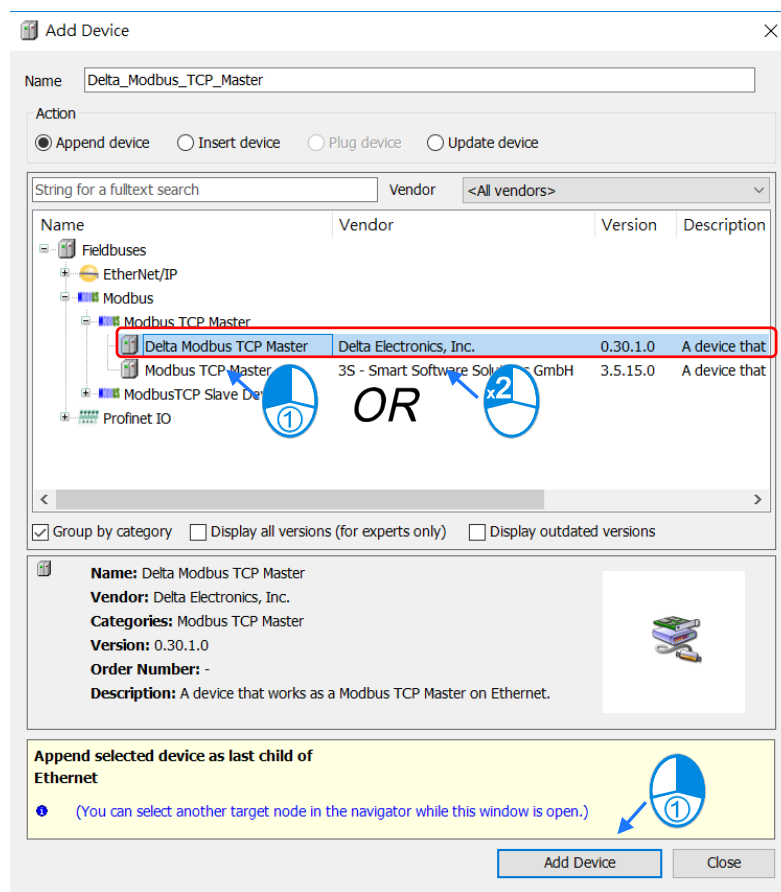


9.3.2.1 Modbus TCP Master 组件建立

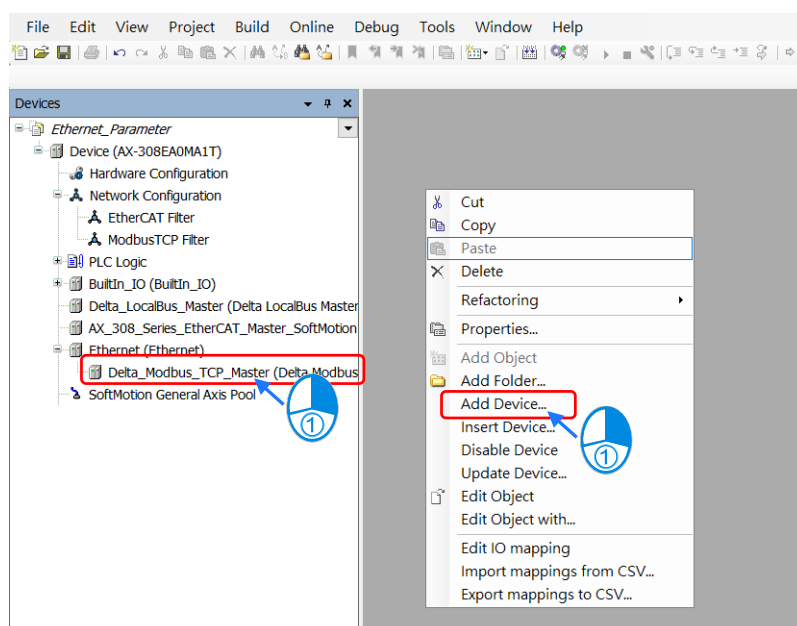
1. 在树形图中 Ethernet 图示单击右键开页面，单击「Add Device」开启「Add Device」窗口。



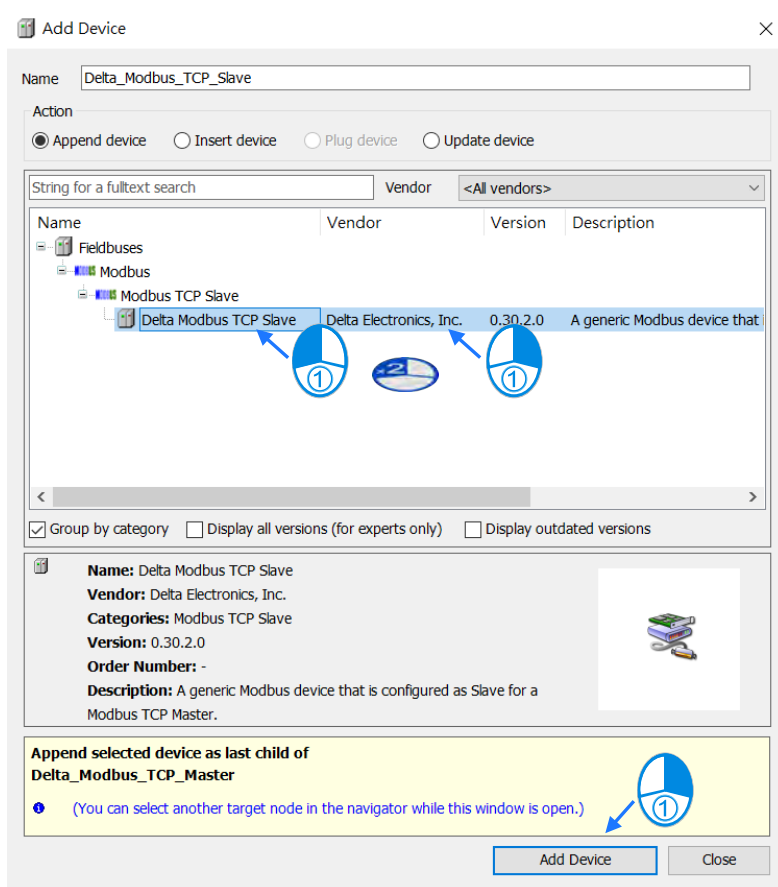
2. 在 Delta Modbus TCP Master 组件单击左键两下，或选择「Delta Modbus TCP Master」组件再单击「Add Device」，即将 Delta Modbus TCP Master 组件新增于树形图。



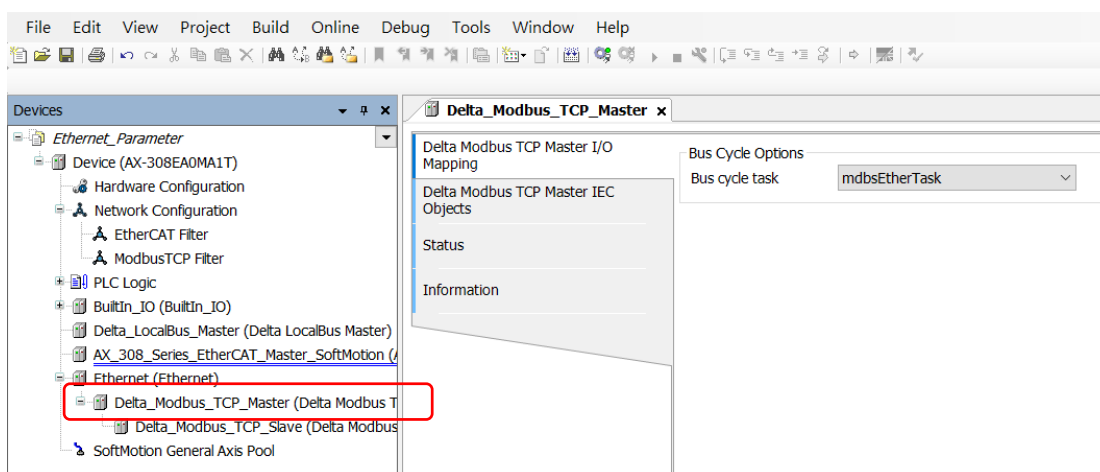
3. 在树形图中 Delta Modbus TCP Master 图示单击右键开页面，单击「Add Device」开启「Add Device」窗口。(此动作为新增所欲通联的 Modbus TCP Slave 从站)



4. 在 Delta Modbus TCP Slave 组件单击左键两下，或选择「Delta Modbus TCP Slave」组件再单击「Add Device」，即将 Ethernet 组件新增于树形图。(此动作为新增所欲通联的 Modbus TCP Slave 从站)



9.3.2.2 Modbus TCP Master 设定说明



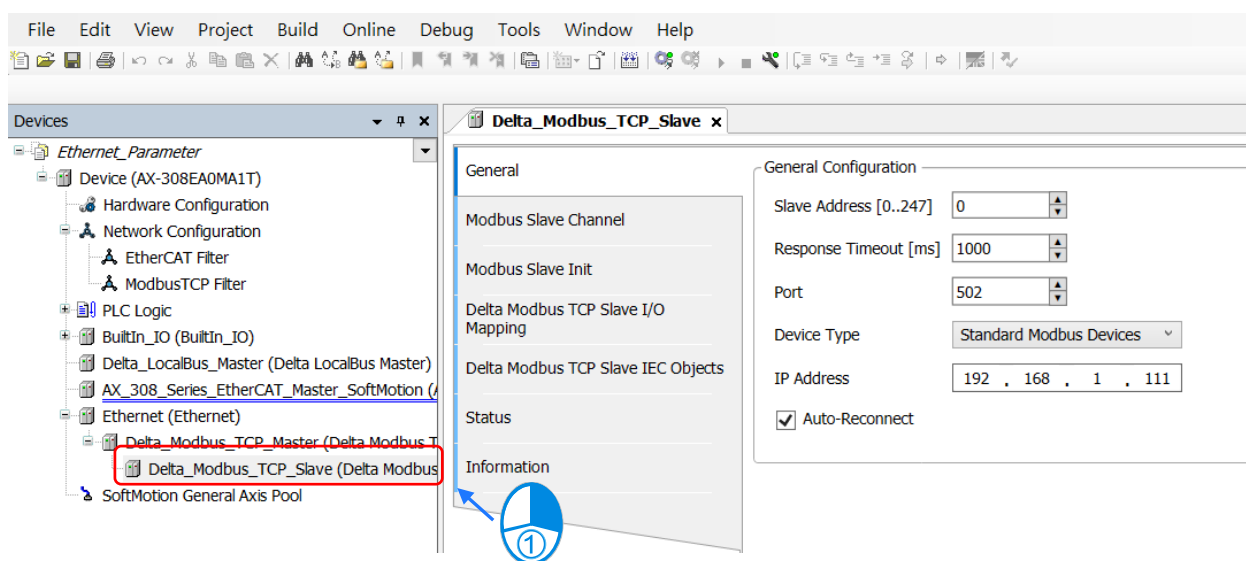
9

页面	说明
Delta Modbus TCP Master I/O Mapping	用户可以于此页面设定 Modbus TCP Master 的 Bus Cycle Task，其详细说明请参考第 4.2.1.6 节 PLC Settings。

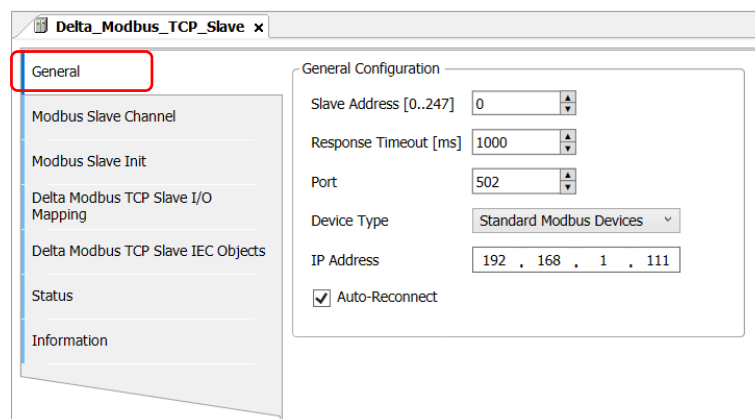
页面	说明
Delta Modbus TCP Master IEC Objects	用户可以通过此页面对 Modbus TCP Master 状态进行判断。 <ul style="list-style-type: none"> ● bStop : TRUE : 停止 Modbus TCP 报文的发送。 ● bSlaveError : TRUE : 与从站联机/通讯异常 ● uiConnectedSlaves : 当前建立联机的从站数量 例 (ST 语言) : Delta Modbus TCP Master.bStop:= TRUE;
Status	Modbus TCP Master 当前运行状态
Information	当前 Modbus TCP Master 相关信息，包含了制造商、产品相关手册、韧体版本等信息...

9.3.2.3 Modbus TCP Slave 设定说明

用户可于树形图单击左键两下开启「Delta_Modbus_TCP_Slave」设定页面，用户可于此对 Modbus TCP 从站进行相关设定。

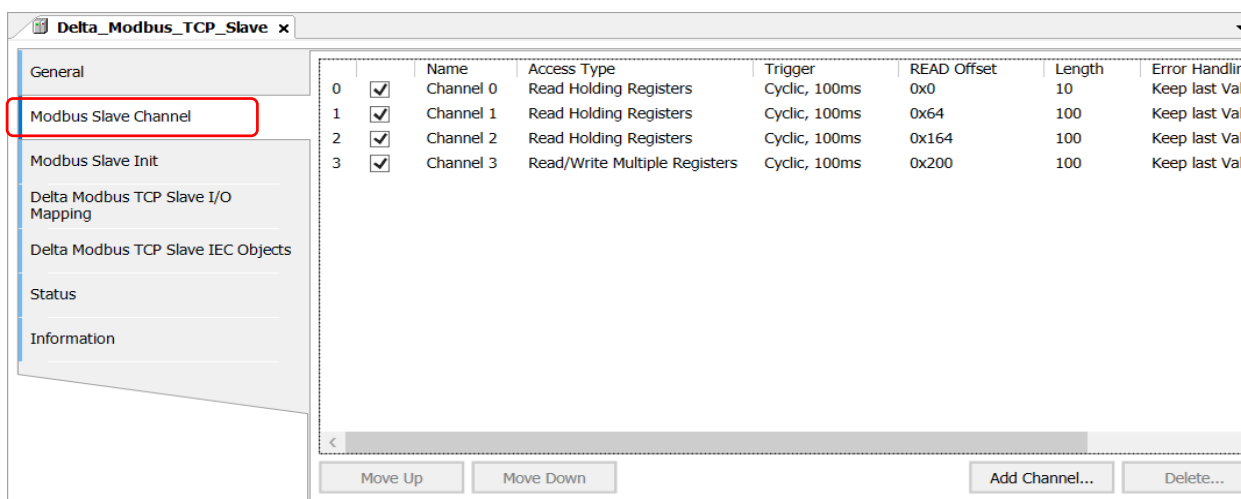


- **General** : 用户可于此页面设定所欲连结的 Modbus TCP 从站的相关信息如站号、IP、通讯端口、通讯格式...

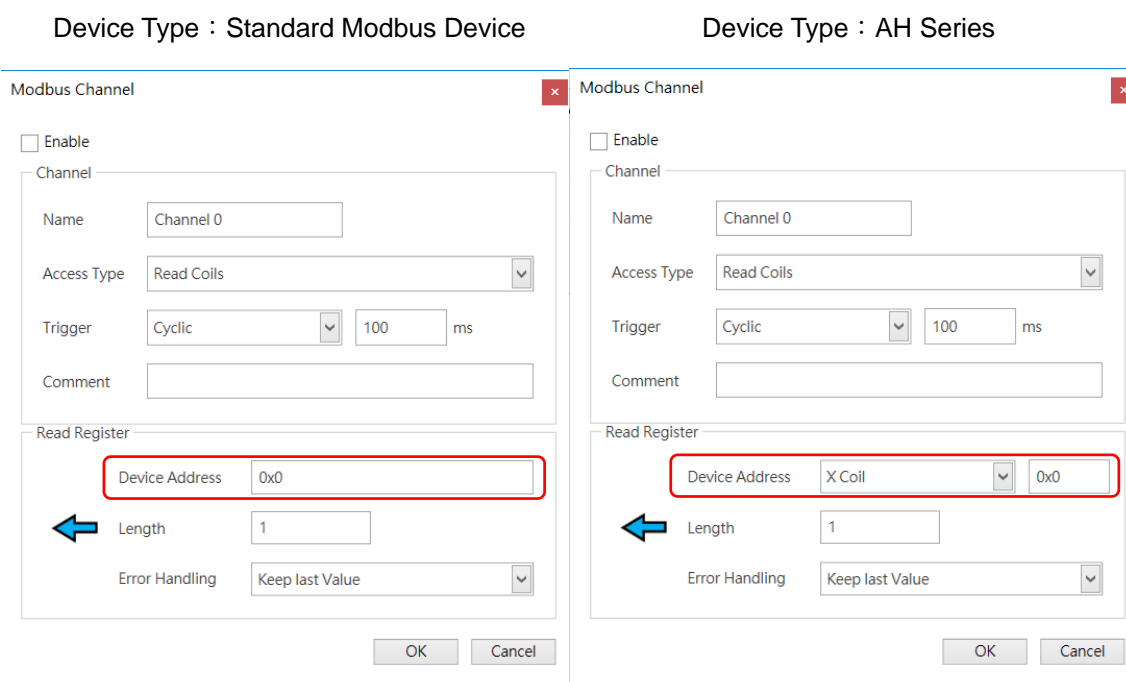


标题	说明
Slave Address	从站站号
Response Timeout	从站响应超时时间
Port	从站通讯端口
Device Type	从站类型，可选择标准 Modbus 通讯或是台达特定控制器。(如选择台达系列控制器，则会自动转换装置与 Modbus 地址)
IP Address	从站 IP 地址
Auto-Reconnect	当发生超时或是错误时，则会自动进行重连。

- **Modbus Slave Channel**：用户可以利用此表格定义从站通道，每通道代表一个 Modbus request 报文，目前每从站最多建立一百条通道，AX308E 系列控制器会表格顺序依序发出 Modbus request 报文，且所有通道共同使用一条 TCP 联机。



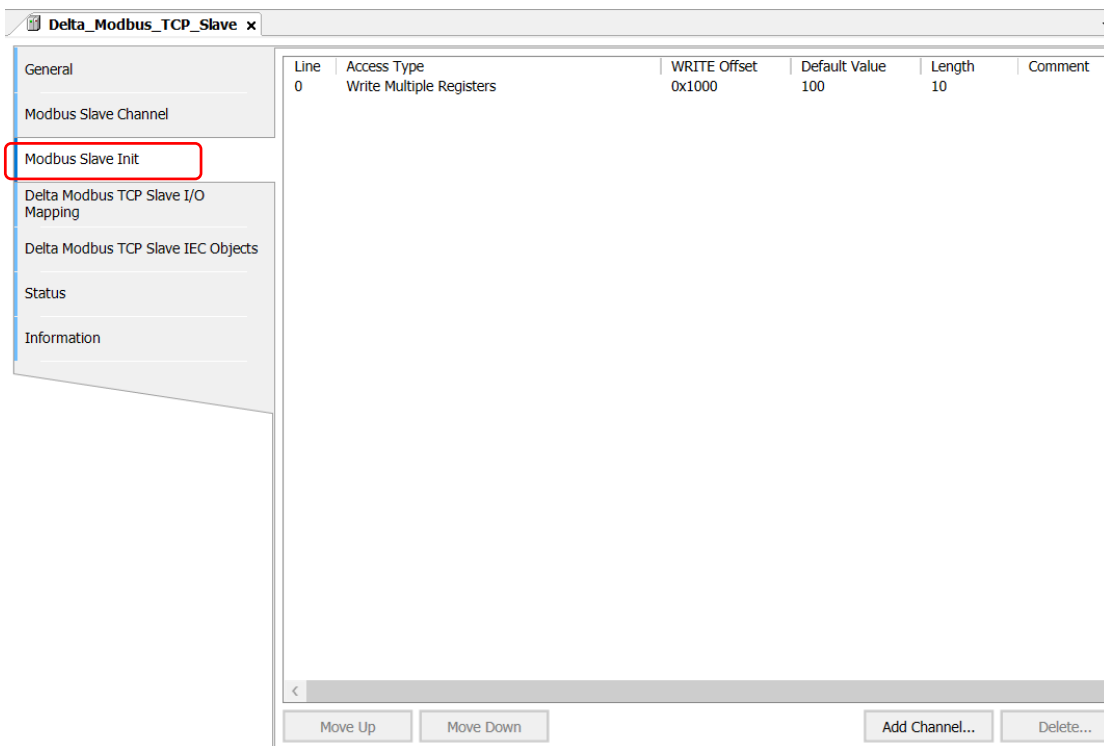
Modbus Channel 会随着 General 页面所设定的 Device Type 类型而有所不同，如果选择台达特定控制器，则会自动换算装置的 Modbus 地址，免去查表的动作。(如下图)



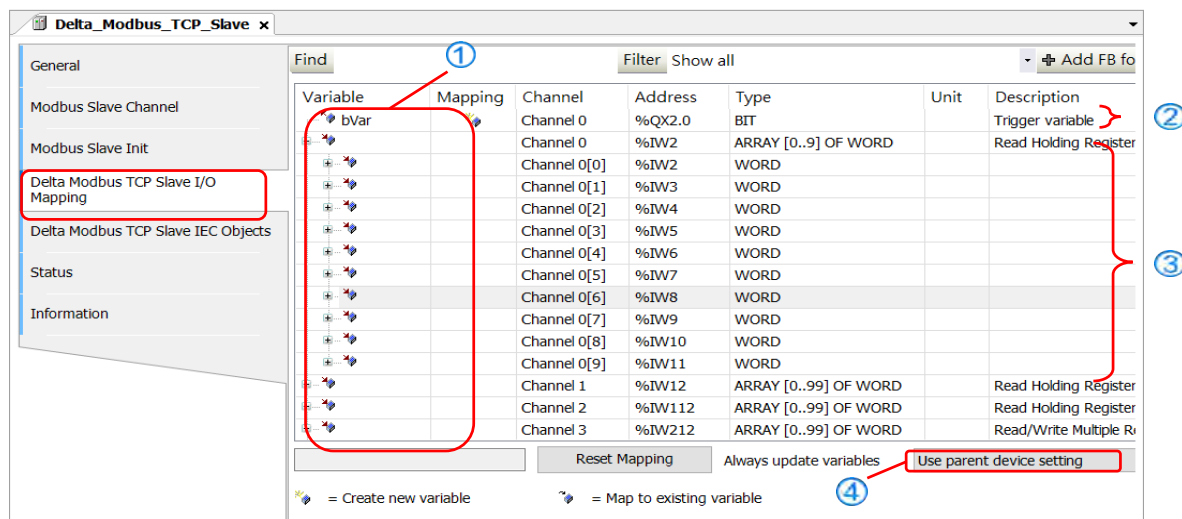
Device Type	Standard Modbus Device	Delta Series Device
Enable	决定此通道是否生效	
Name	定义此通道名称	
Access Type	Modbus 通讯功能码 : <ul style="list-style-type: none"> ● Read coils (0x01) ● Read discrete inputs (0x02) ● Read holding registers (0x03) ● Read input registers (0x04) ● Read single coil (0x05) ● Write single register (0x06) ● Write multiple coils (0x0F) ● Write multiple registers (0x10) ● Read/Write multiple registers (0x17) 	Read/Write Registers <ul style="list-style-type: none"> ● Read coils ● Read registers ● Write coils ● Write registers 注：控制器会依照所读/写的装置类型选择相对应的 Modbus 功能码。
Trigger	<ul style="list-style-type: none"> ● Cyclic : Modbus request 依照设定的周期时间触发 ● Rising edge : Modbus request 通过布尔变量触发，此布尔变量定义于 I/O Mapping 页面。 ● Application : Modbus request 可以通过功能块「DFB_ModbusTCPchannel」触发。 	<ul style="list-style-type: none"> ● Cyclic : Modbus request 依照设定的周期时间触发 ● Rising edge : Modbus request 通过布尔变量触发，此布尔变量定义于 I/O Mapping 页面。 ● Application : Modbus request 可以通过功能块「DFB_ModbusTCPchannel」触发。
Comment	通道备注	

Device Type	Standard Modbus Device	Delta Series Device
Device Address	Register 的 Modbus 地址	台达系列控制器的装置地址
Length	读取/写入的数据长度。	读取/写入的数据长度。(最大长度为 256 个 coils 与 100 个 registers)
Error Handling	当通讯发生异常时，Register 数据的设定动作。 <ul style="list-style-type: none"> ● Set To ZERO ● Keep last value 	

- Modbus Slave Init：用户可以利用此表格决定，是否要在 AX308E 系列控制器与从站成功建立 TCP 联机之后，对从站的 Registers 写入初始值。

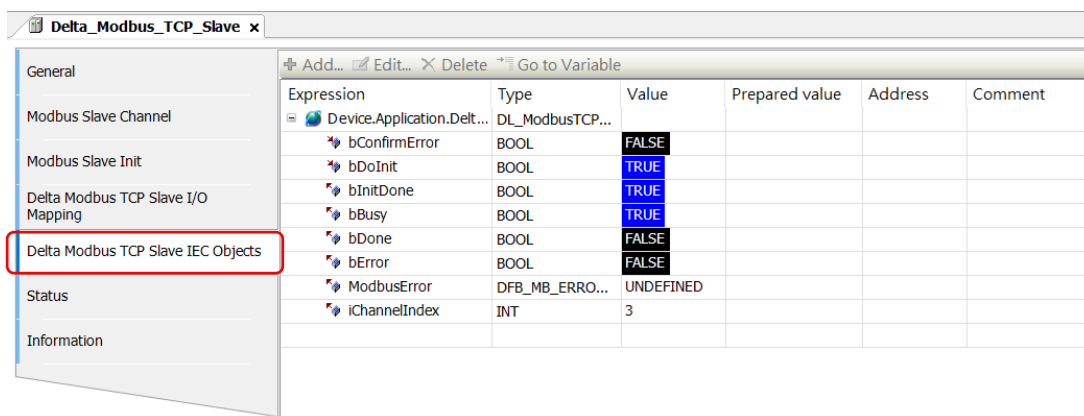


- Delta Modbus TCP Slave I/O Mapping：当 Modbus Slave Channel 页面设定通道完毕后，即于此页面自动定义所需的参数。



- ① : 用户可以于此字段将所有相关参数映像到用户自行变量。
- ② : 此通道的触发布尔变量。
- ③ : 此通道读取/写入的控制器 registers。
- ④ : 数据更新时机。(详细说明可参考第 4.2.1.6 节「PLC Settings」)

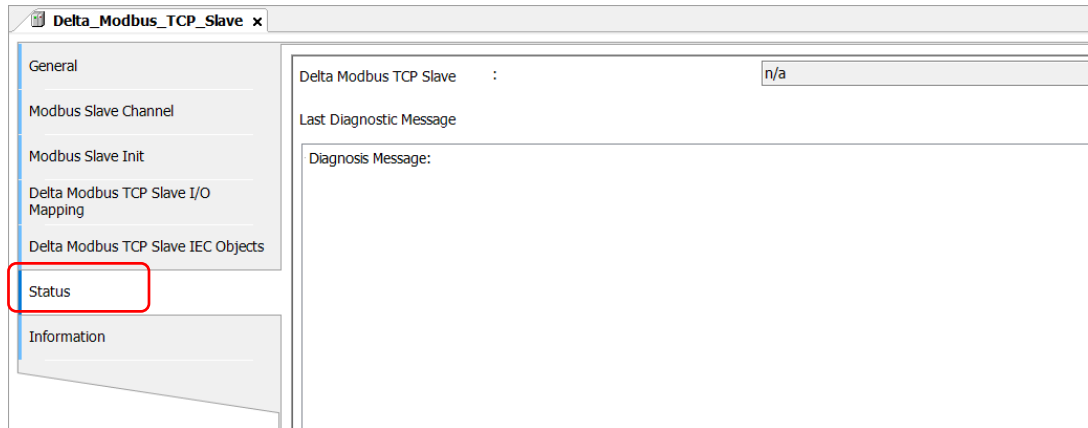
- Delta Modbus TCP Slave IEC Object : 用户可以通过此页面对 Modbus TCP Slave 状态进行判断。



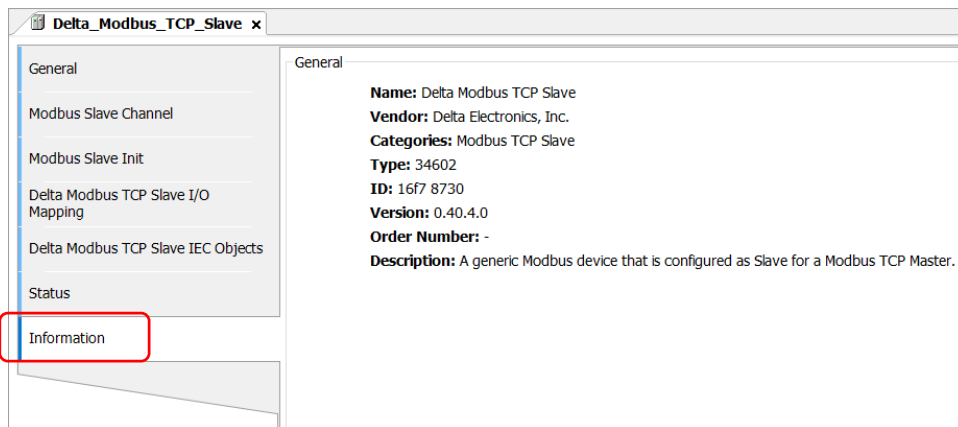
标题	说明
bConfirmError	如从站设定没有勾选「Auto-Reconnect」, 则当任一 Channel 发生错误时, 则此从站将停止后续的 Channel 报文的发送, 如将「bConfirmError」设定为 TRUE, 则后续 Channel 将开始报文发送
bDoInit	执行从站初始值设定
bInitDone	从站初始值设定完毕
bBusy	写入/读取报文中
bDone	此 Channel 写入/读取报文完成
bError	此 Channel 写入/读取错误发生
ModbusError	错误码

标题	说明
iChannelIndex	当前写入/读取的 Channel

- **Status** : 用户可于此页面查询 Modbus TCP Slave 运行状态, 如 Running、Stopped 与其它特定说明, 帮助用户了解当前运作情形。



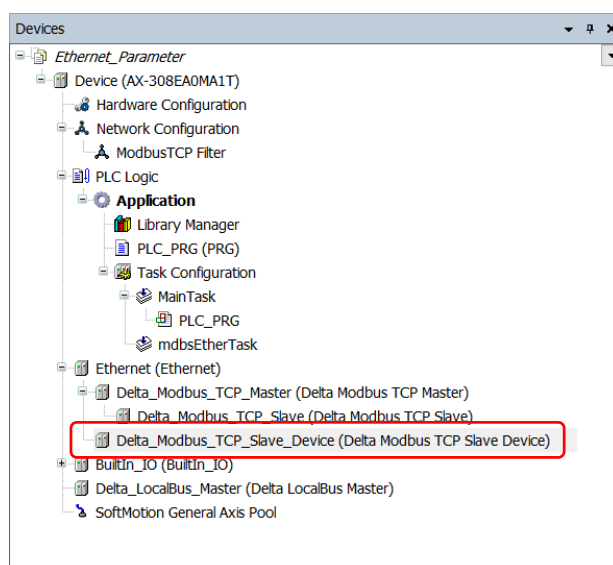
- **Information** : 当前 Modbus TCP Slave 相关信息, 包含了制造商、产品相关手册、韧体版本等信息...



9.3.3 Modbus TCP Slave

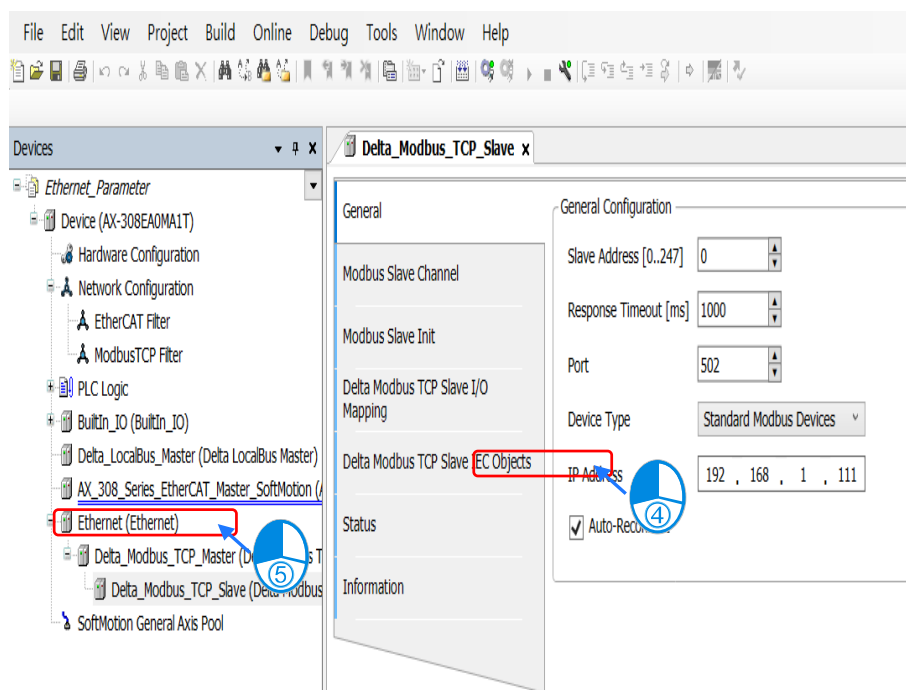
在使用上如 AX308E 系列控制器要担任 Modbus TCP Slave 时，则须新增 Modbus TCP Slave，并设定其允许的访问的 registers 区间，但如果 Modbus TCP Master 使用台达特定通讯协议，则不受区间限制，即可对所有 register 进行读/写。

- **Modbus TCP Slave 树形图**

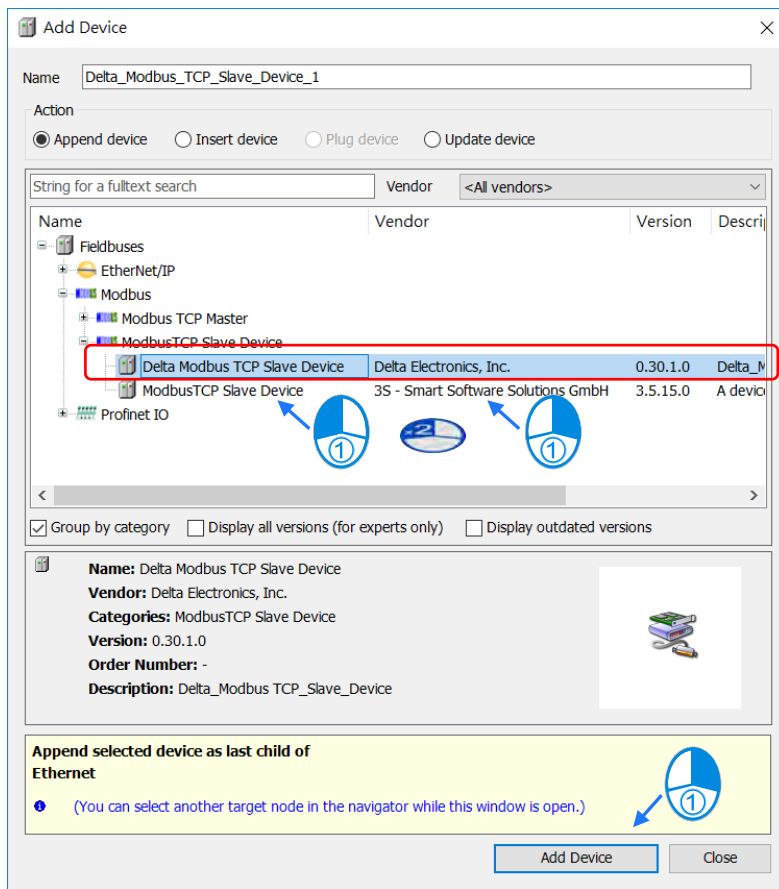


9.3.3.1 Modbus TCP Slave 组件建立

1. 在树形图中 Ethernet 图示单击右键开页面，单击「Add Device」开启「Add Device」窗口。



- 在 Delta Modbus TCP Slave Device 组件单击左键两下，或选择「Delta Modbus TCP Slave Device」组件再单击「Add Device」，即将 Delta Modbus TCP Slave 组件新增于树形图。



9.3.3.2 Modbus TCP Slave 设定说明

- General

用户可于此页面设定控制器所预计开放的通讯端口、站号与允许访问的 registers 区间，如 Modbus TCP Master 使用台达特定通讯协议，则不受限制可以访问所有区间。

The screenshot shows the configuration page for a Delta Modbus TCP Slave Device. The left sidebar has a red box around the 'General' tab. The main content area is divided into two sections:

- General Configuration:**
 - TCP Port: 502
 - Station ID: 3
- Address Information Setting:**

Register Type	Device Start Address	Device Quantity	Modbus Start Address
Holding Register (%MW)	0	10	0
Coils (%MW)	0	10	0
Holding Register (%QW)	0	10	256
Coils (%QW)	0	10	256
Input Register (%IW)	0	10	0
Input Coils (%IW)	0	10	0

- Delta Modbus TCP Slave Device I/O Mapping

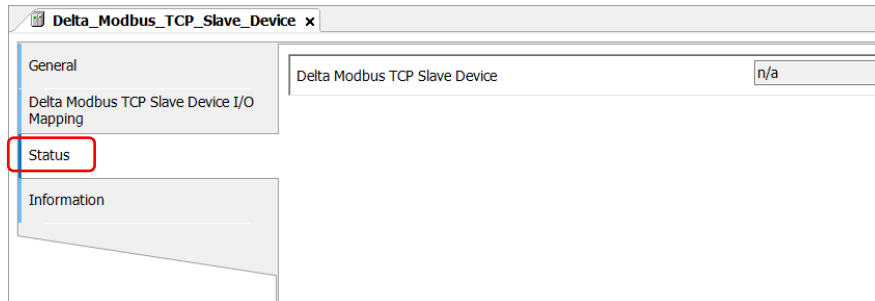
用户可于此页面设定 Modbus TCP Slave Device 的 Bus Cycle Task，其详细说明请参考第 4.2.1.6 节「PLC Settings」。

The screenshot shows the configuration page for a Delta Modbus TCP Slave Device, specifically the 'Delta Modbus TCP Slave Device I/O Mapping' tab. The left sidebar has a red box around this tab. The main content area shows:

- Bus Cycle Options:**
 - Bus cycle task: mdbEtherTask

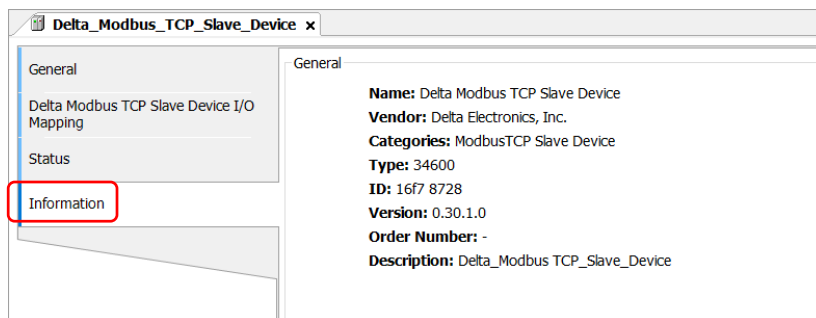
- Status

用户可于此页面查询 Modbus TCP Slave Device 运行状态，如 Running、Stopped 与其它特定说明，帮助用户了解当前运作情形。



- Information

当前 Modbus TCP Slave Device 相关信息，包含了制造商、产品相关手册、韧体版本等信息...



9.4 EtherNet/IP 通讯

用户欲改变控制器网络相关参数时，则必须新增 Ethernet 组件方可调整，而且所有网络相关功能皆需于此组件下进行设定（如 Modbus TCP 与 EtherNet/IP...）。

9.4.1 EtherNet/IP 简介

9.4.1.1 EtherNet/IP 介绍

EtherNet/IP 为由 ODVA 协会（Open DeviceNet Vendors Association）管理的工业以太网通讯协议，其名称中 IP 是工业协议（Industrial Protocol）的缩写。

EtherNet/IP 架构于 TCP/IP 通讯协议上，可兼容于一般 IT 网络，提供工厂自动化（FA）、楼宇自动化（BA）和程序自动化（PA）等应用高速稳定的应用。

台达 EtherNet/IP 产品涵盖控制类与驱动类产品，包括可程序逻辑控制器、变频器、人机、伺服和交换器，支持产品一览表请参考第 9.4.5 节。此外，可通过 EDS 文件与其它厂牌 EtherNet/IP 设备连接。

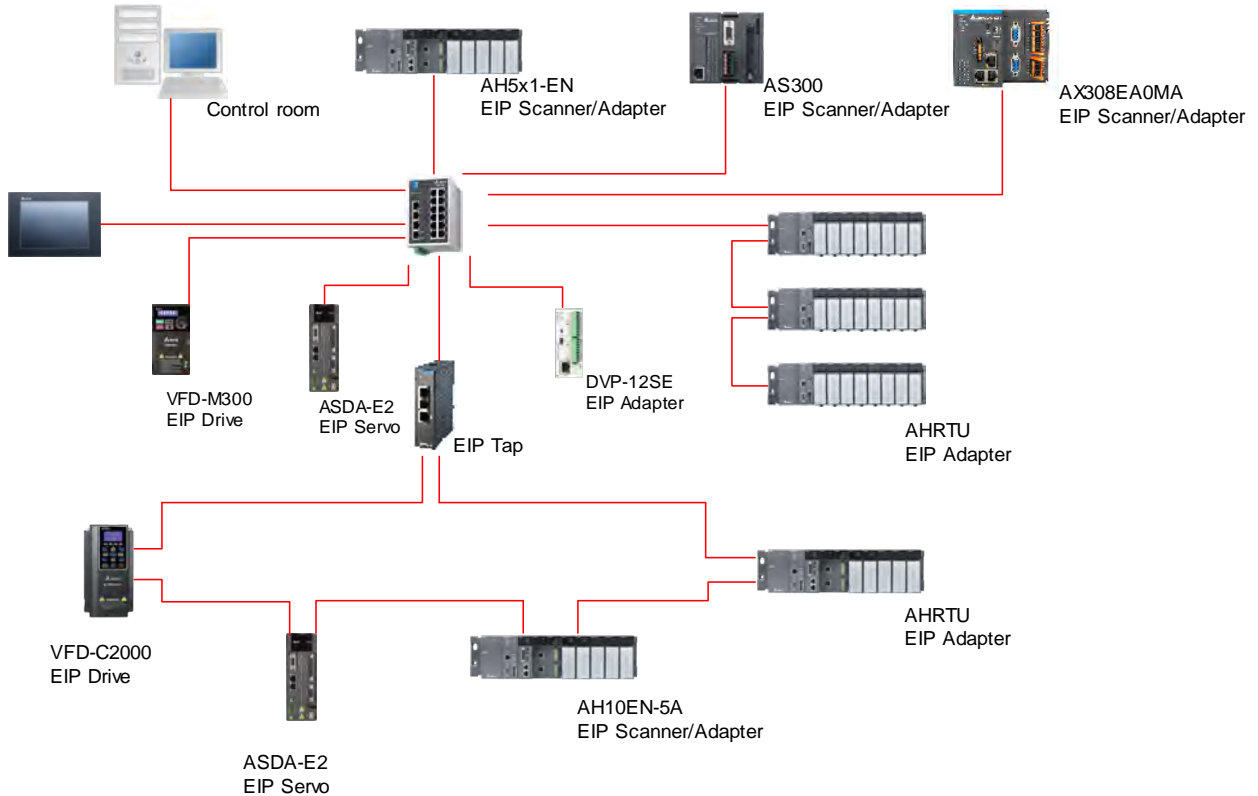
9.4.1.2 常见网络名词说明

缩写	说明
ODVA	Open DeviceNet Vendor Association，EtherNet/IP 管理组织。
EIP	EtherNet/IP 缩写，工业以太网网络协议。各供货商产品有互操作性，IP 为 Industrial Protocol 缩写。本文中缩写为 EIP 替代。
I/O Connection	EtherNet/IP 周期性数据交换
Explicit Message	EtherNet/IP 非周期性数据交换，数据通过指令单笔传送。
RPI	Requested Packet Interval，IO connection 周期性数据交换间隔时间。
ACD	Address Conflict Detection，IP 地址重复错误检测功能。
P/C TAG	Produced / Consumed TAG，生产者/消费者地址卷标。—Produced TAG 可对应至多台设备 Consumed TAG。TAG 为 Rockwell PLC 中数据表示方式，如同台达 PLC 之寄存器。
EDS	Electronic Data Sheets，电子参数文件。EtherNet/IP 中用以识别 EtherNet/IP 设备。
Data Mapping	数据交换，设备与设备间数据转换。
EIP Scanner	EIP 主站，EtherNet/IP 中称为 Scanner。
EIP Adapter	EIP 从站，EtherNet/IP 中称为 Adapter。
MODBUS TCP	常被使用于 Ethernet 通讯的一种泛用 MODBUS 通讯协议

9.4.1.3 Ethernet 特色

9.4.1.3.1 Delta EIP 系统架构图

台达 EtherNet/IP 产品架构图如下，包含 EIP Scanner 和 Adapter，各产品间通过 IO Connection 与 Explicit Message 进行数据交换。



9.4.1.3.2 EIP 特点

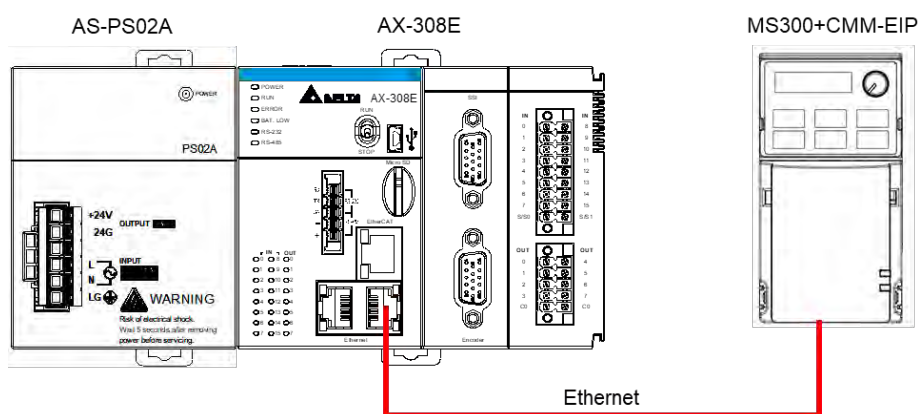
- 弹性
 - 弹性拓扑，EIP设备包含单以太网网络通讯端口 (Single port) 与双以太网网络通讯端口 (Dual port)，支持星状 (Star)、线性 (Linear) 和环状 (Ring) 网络拓扑，方便现场快速扩充与产线变更管理。
 - 兼容于IT网络，无须专业IT技术人员即可轻松组网，并支持Wi-Fi连接。
- 简易
 - 网络线：台达提供完整人机接口、可程序逻辑控制器、变频器和伺服电机等完整产品线，通过RJ-45网络线即可完成组网，简化线材备料与库存。
 - (单)一网络：取代传统三层工业网络架构，提供100MB/bits高速周期性与非周期性数据交换。完整工业网络诊断机制，缩短除错时间。

9.4.2 EtherNet/IP Scanner Function

9.4.2.1 建立变频器 MS300 数据读写

9.4.2.1.1 系统硬件配置

本范例使用 AX-308E 通过 Ethernet 端口口通讯连接 MS300 变频器搭配 CMM-EIP 通讯卡。



注：CMM-EIP 通讯卡固件需为 V2.04.01 以上版本

9.4.2.1.2 隐性报文数据读写设定

EtherNet/IP 隐性报文为周期性数据交换，通过 EtherNet/IP 上位机（Scanner）对配件卡的映射寄存器来设定读取/写入的数据地址，再通过映像寄存器来做一次性的读取/写入数据值。

- 变频器搭配 EIP 通讯卡使用的前置设定

- 变频器相关设定

欲使用网络通过配件卡操作变频器时，请先将变频器控制权设定为配件卡。设定方式为使用操作器并参考以下操作步骤：

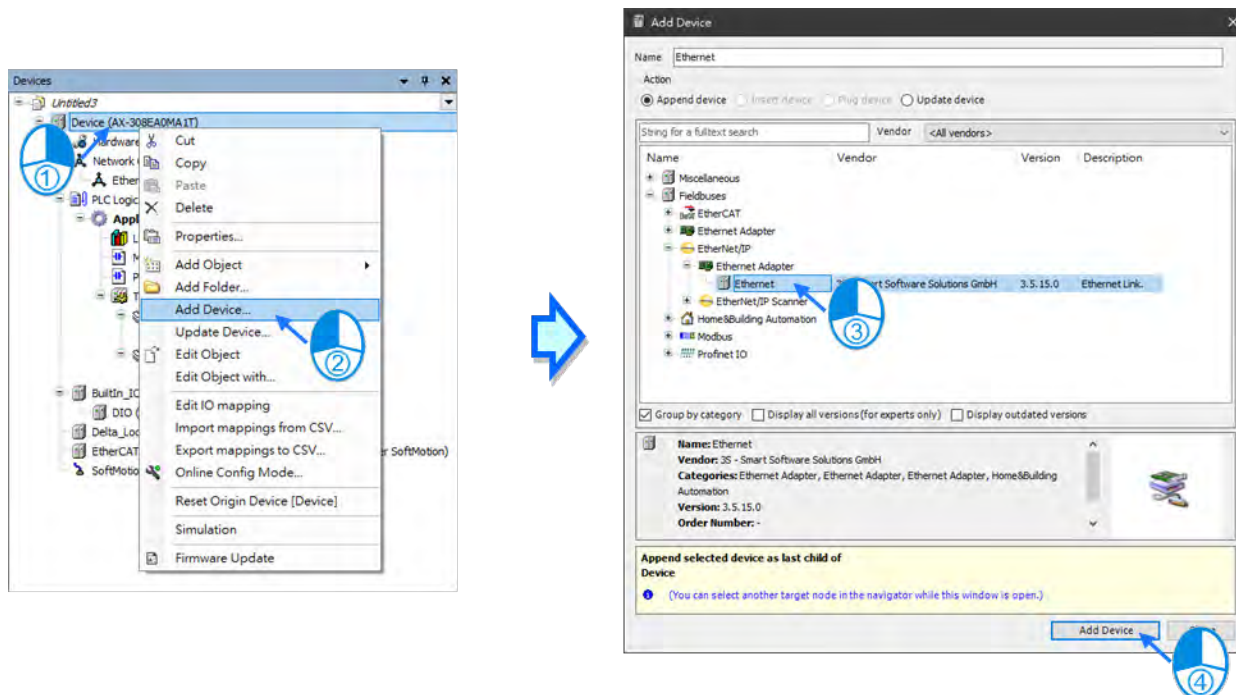
1. 当配件卡接通后，检查参数 09-60 是否有值，该参数值会显示为 5（EtherNet/IP）
2. 设定参数 09-75=0（静态 IP），IP 地址由用户自行设定
3. 设定参数 09-76~参数 09-79 配件卡 IP 地址（默认为 192.168.1.5），修改为 192.168.1.30，设定完成后将参数 09-91 设定为 2，使参数生效。
4. 设定参数 00-20=8（设定 AUTO 频率命令来源为通讯卡）
5. 设定参数 00-21=5（设定 AUTO 控制来源为通讯卡）
6. 设定参数 09-30=1（设定通讯译码方式为 60xx 或 20xx，译码方式说明请参阅变频器 EtherNet/IP 应用手册第 4.2 节 EtherNet/IP 控制方式规范）

● 建立 EIP 操作范例

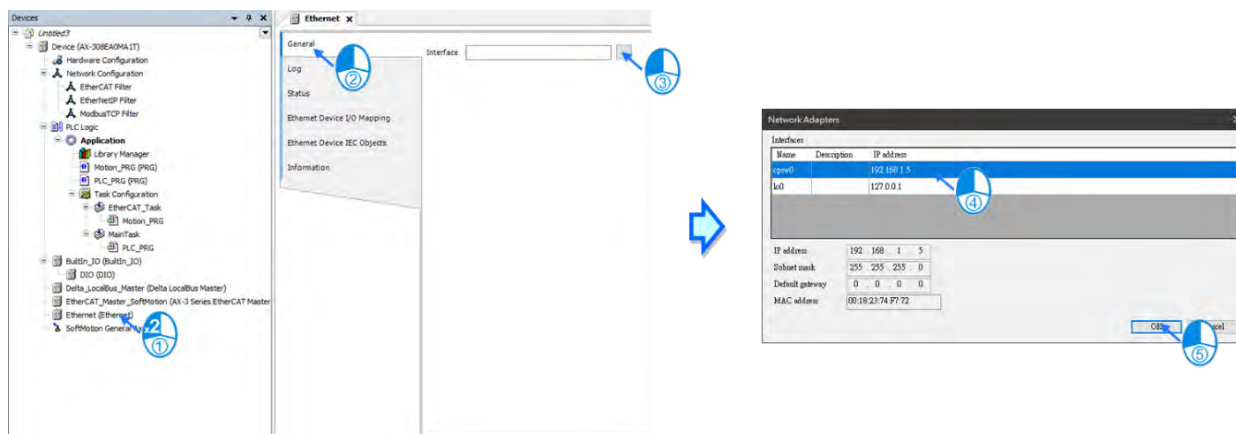
本范例使用的设备 IP 设置如下：

使用设备	AX-308E	192.168.1.5 (默认)
	MS300* CMM-EIP02	192.168.1.30

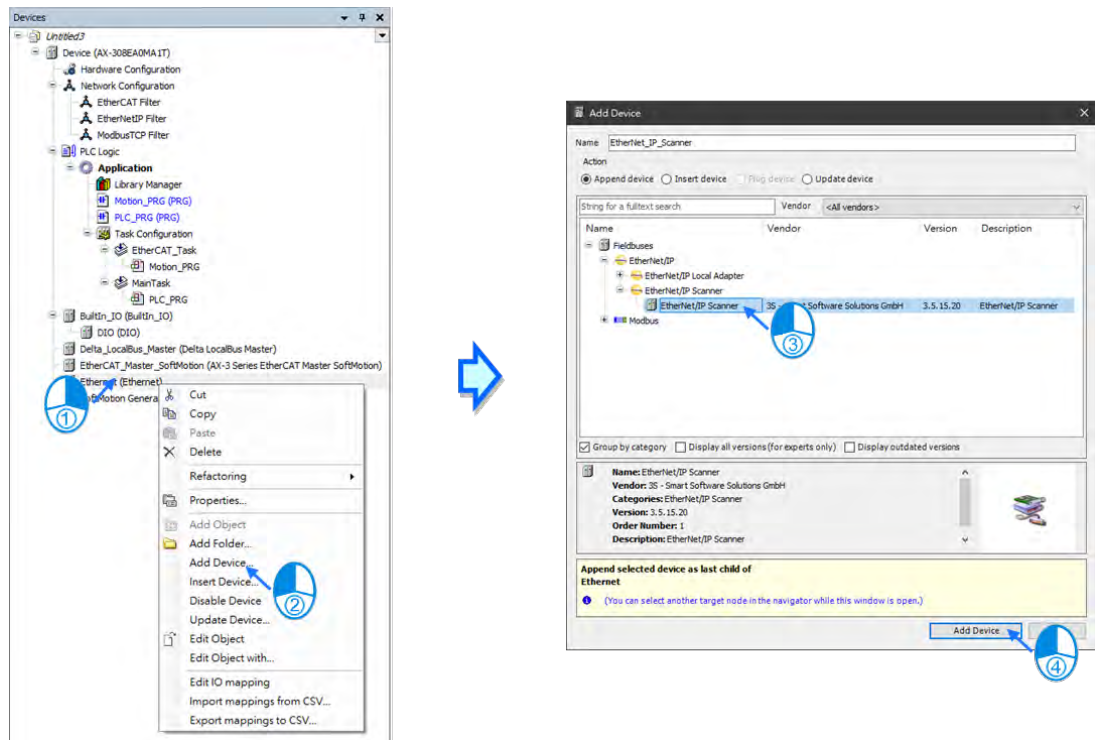
1. 建立 Ethernet Device



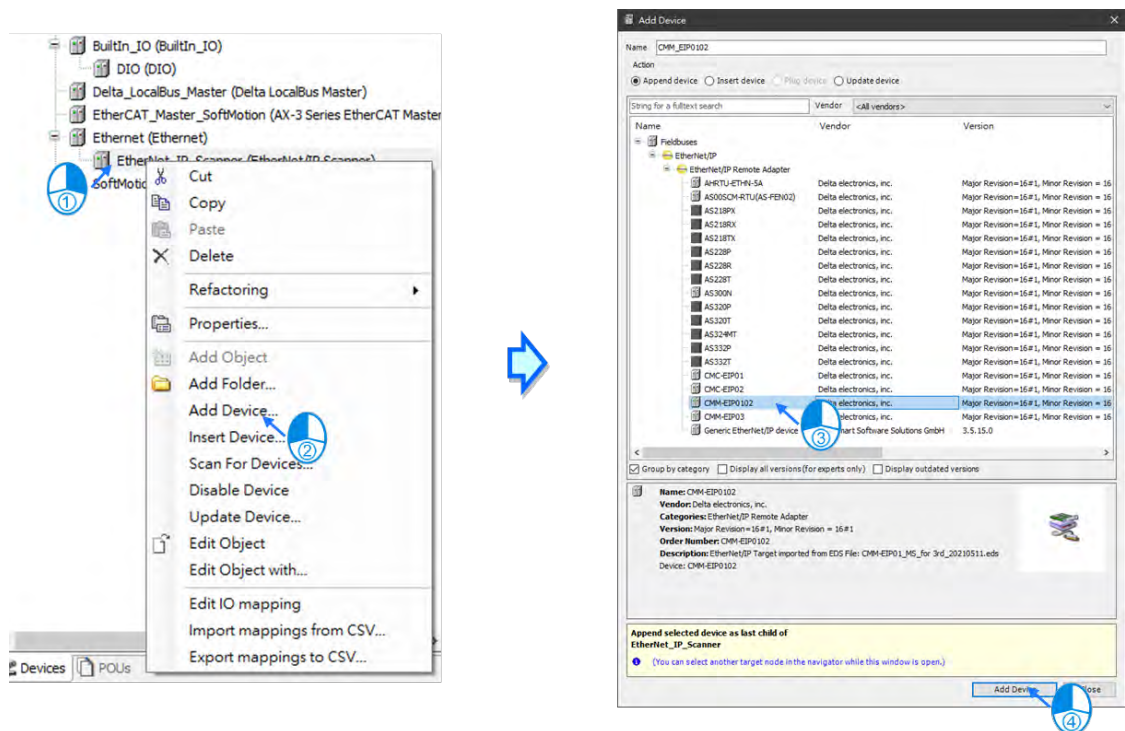
2. 建立 Interface · 单击 Ethernet -> General



3. 建立 EtherNet/IP Scanner

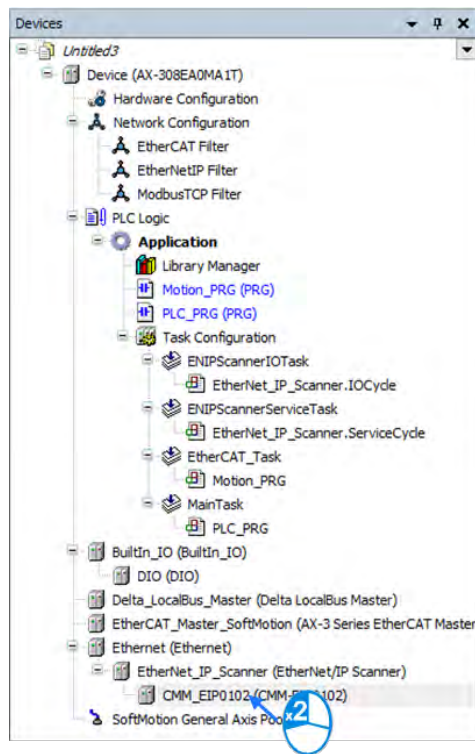


4. 建立 CMM-EIP0102，首先选择 Ethernet 右键 Add Device 选择对应的 Adapter*。

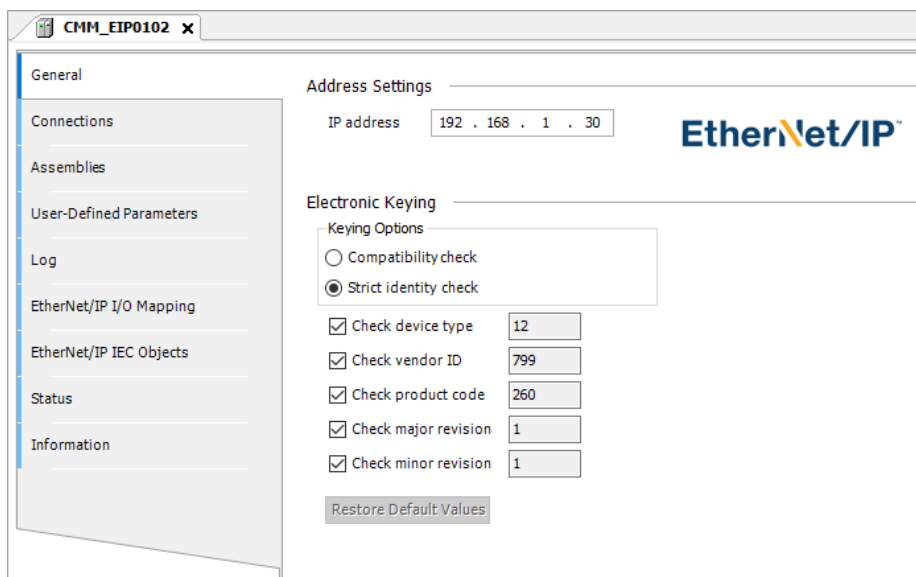


*注：可通过 Scan For Device 方式，建立 Adapter。

5. 单击 CMM_EIP0102



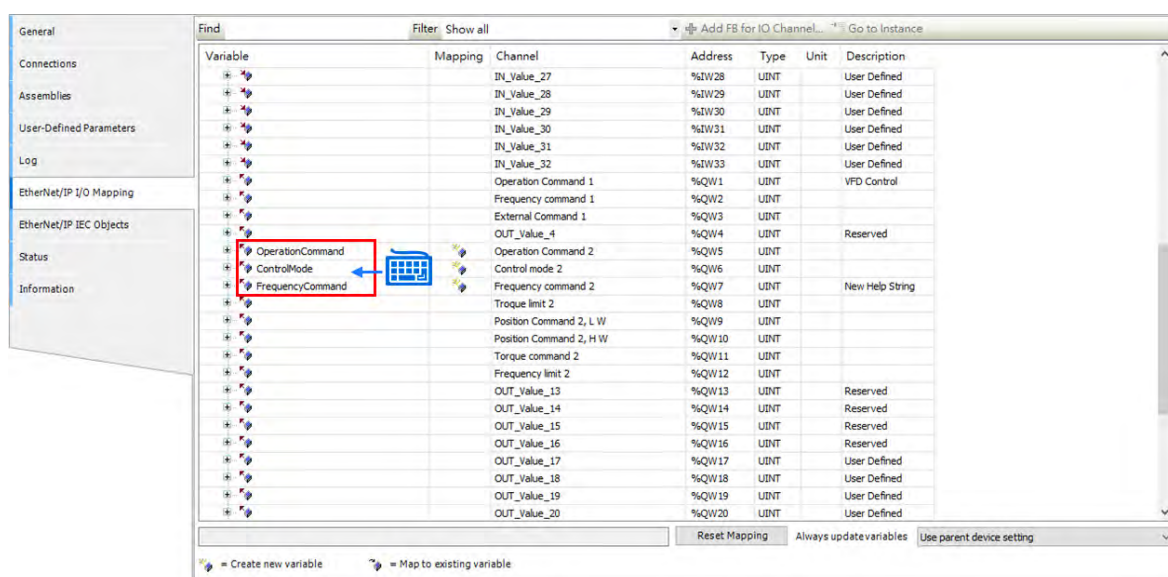
6. 单击 General，设定 IP Address 为 192.168.1.30。



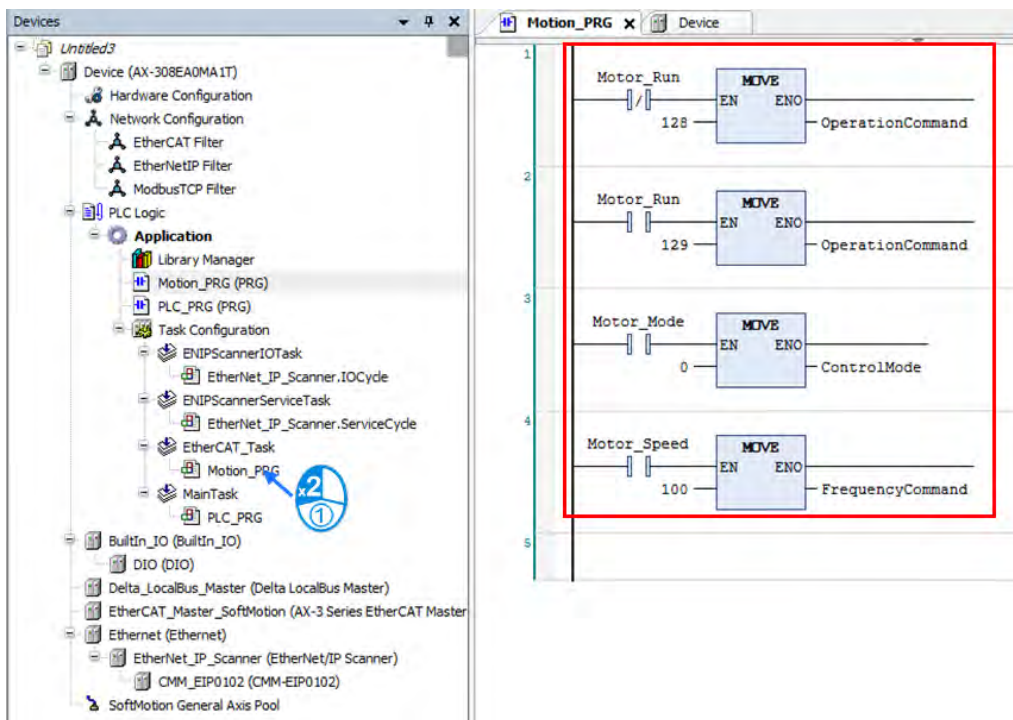
标题	说明
IP address	目标设备的地址
Compatibility check	目标与 EDS 文件中的相关信息进行兼容性检查
Strict identity check	目标与 EDS 文件中的相关信息进行严谨检查。用户可设定需检查信息。
Check Device type	检查设备类型
Check Vendor ID	供货商 ID
Check Product code*	产品代码
Check Major revision	主要修订
Check Minor revision	次要修订

*注：如需同时使用 Adapter、Scanner，请取消勾选 Check Product code。

- 单击 EtherNet/IP I/O Mapping，在 Channel 的 Operation Command 2、Control Mode 2、Frequency command 2 新增变量名称。

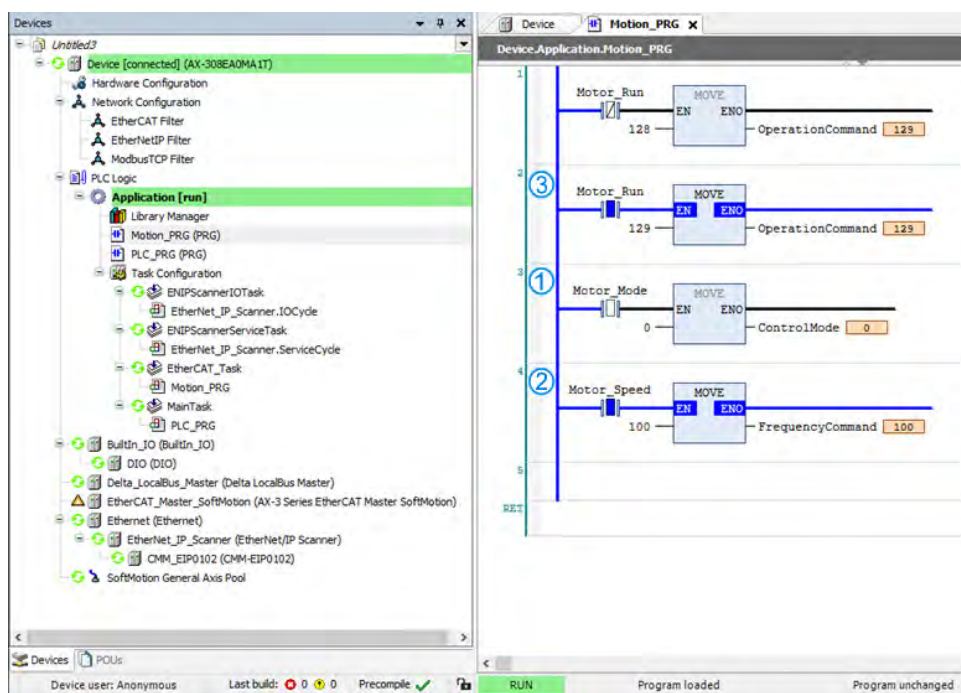


8. 单击 Motion_PRG · 新增下方程式



9. 在线监控执行下方流程：

- ① 执行 Motor_Mode 启动速度模式。
- ② 执行 Motor_Speed 写入速度 100。(单位为 Hz, 值为小数 2 位, 例如写入 100 表示 1.00 Hz)
- ③ 执行 Motor_Run 写入 129 为启动运转, 反之 128 为激磁状态。



注：关于 CMM-EIP 参数说明请参阅变频器 EtherNet/IP 应用手册。

9.4.2.1.3 显性报文 CIP Object 数据读写设定

使用此功能前，请先参阅变频器 **EtherNet/IP 应用手册附录 A<EtherNet/IP 服务与对象>**，确认配件卡支持的对象，并确认您了解显性报文的读写方式。上位机可通过对应的 **Object Class** 地址直接对变频器设定内容值，目前变频器所占的 **Object class code** 为 0x300，参数地址的显性报文规则性对应方式如下：

EIP 通讯数据型式

Object class		Instance		Attribute
0x300	+	Pr. Group	+	Pr. Number

- 读写范例

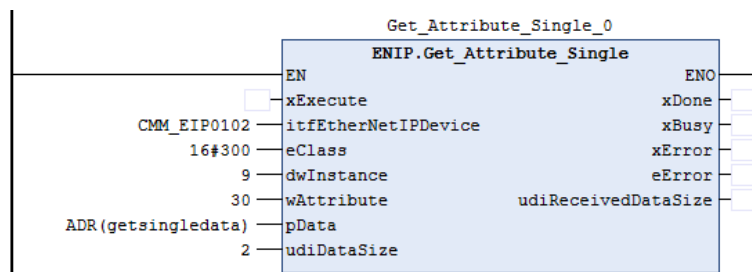
若要对参数 09-30 (通讯译码方式) 读写命令，方式如下：

- 宣告功能块与变量

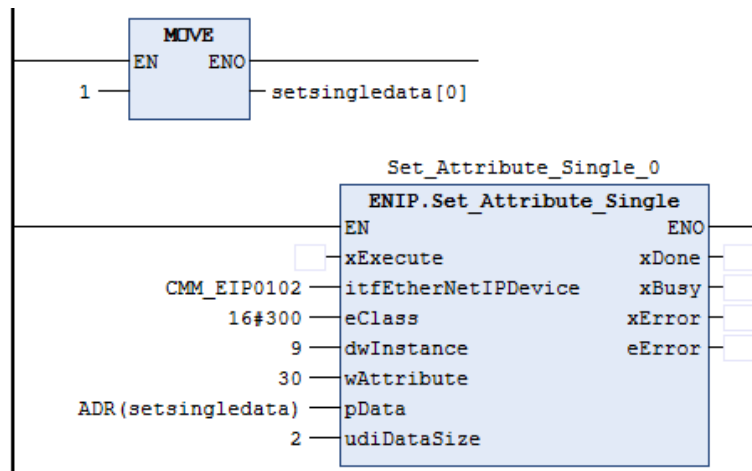
```
PROGRAM PLC_PRG
VAR

    Get_Attribute_Single_0: ENIP.Get_Attribute_Single;
    Set_Attribute_Single_0: ENIP.Set_Attribute_Single;
    getsingledata: ARRAY[0..999] OF BYTE;
    setsingledata: ARRAY[0..999] OF BYTE;
END_VAR
```

- 通过下方功能块读取 9-30 参数



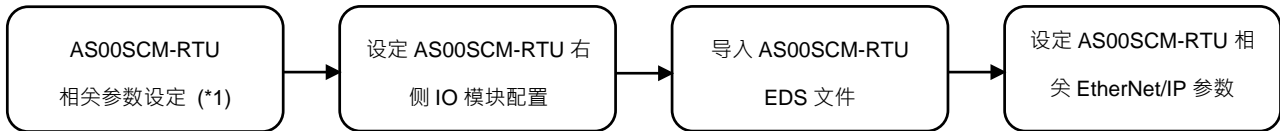
- 通过下方功能块写入 9-30 为 1



9.4.2.2 建立 AS00SCM-A (AS-FEN02 通讯卡) 数据读写

本节将介绍如何通过 EtherNet/IP 连接 AS00SCM-RTU (AS-FEN02)，在进行操作之前，请务必详阅「AS 系列模块手册」第九章「AS00SCM 串行通讯模块」，了解此模块相关设定与使用。

操作步骤：

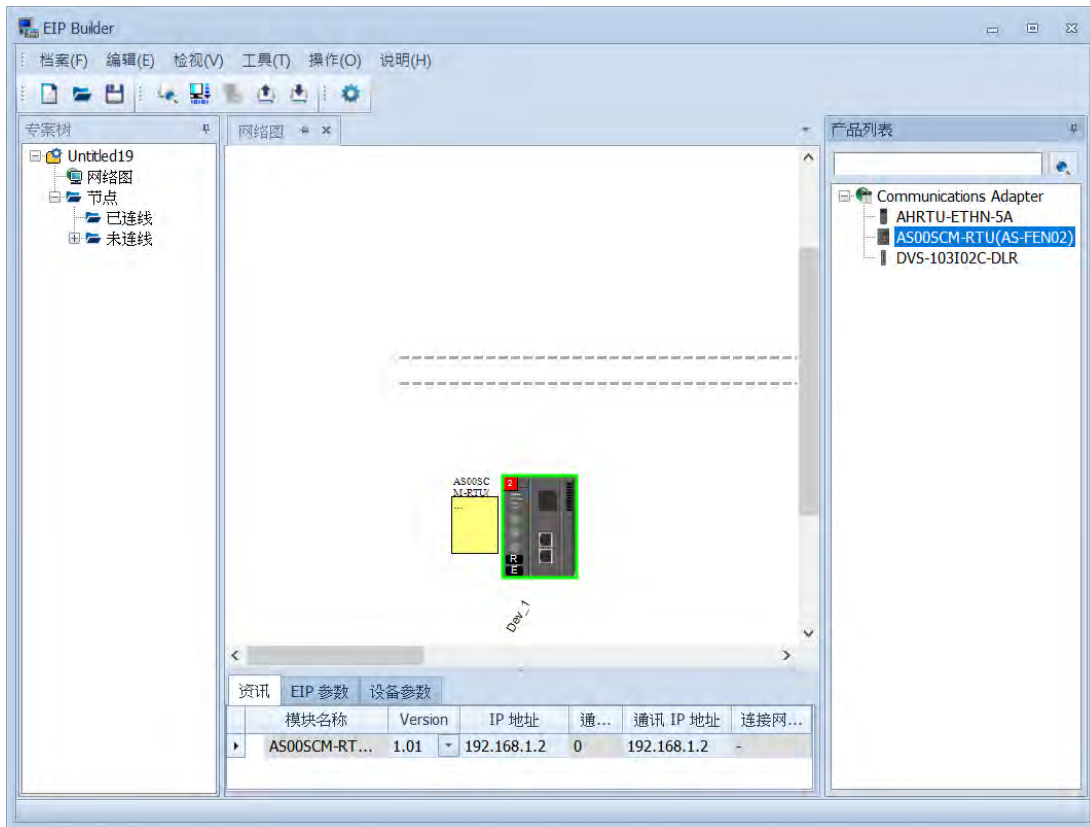


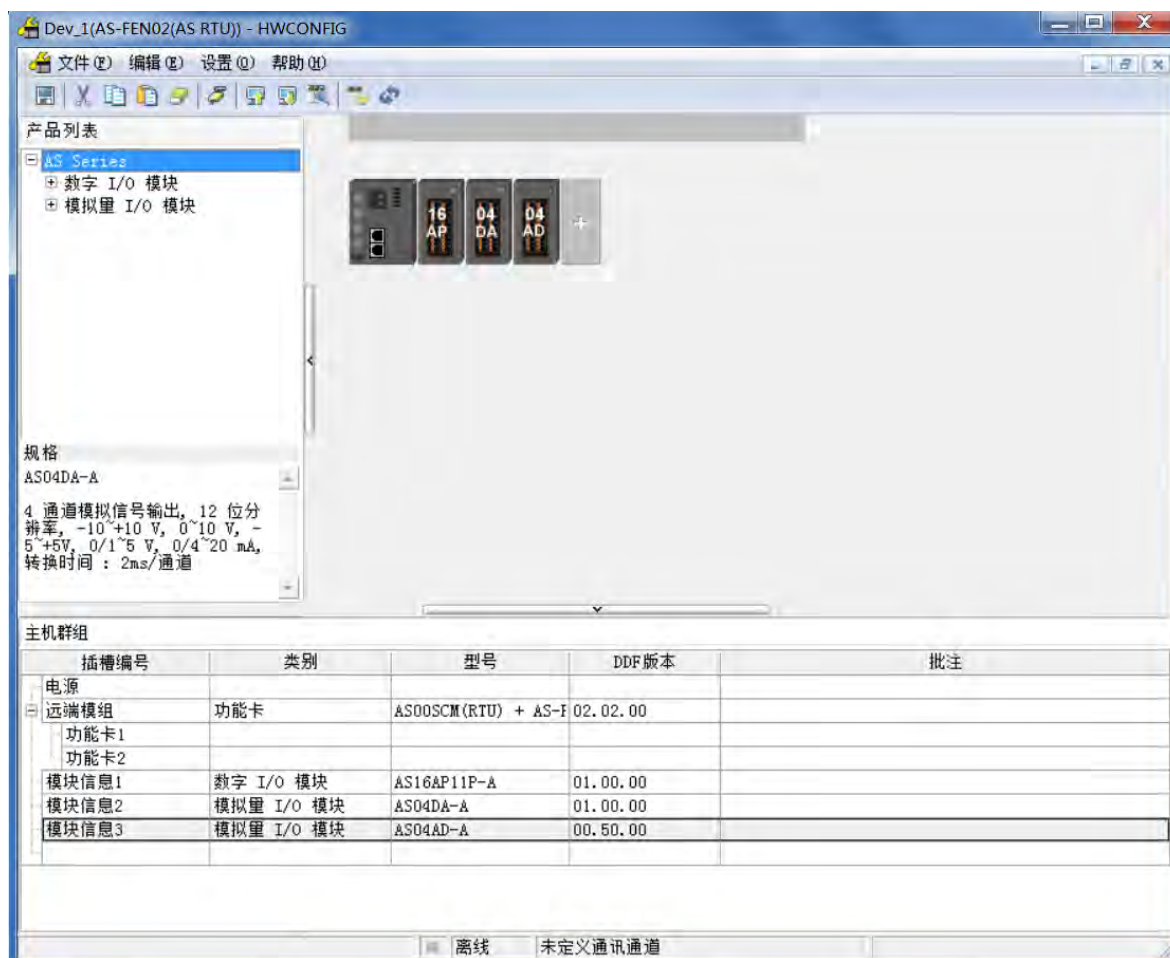
*1：请参考「AS 系列模块手册」第九章「AS00SCM 串行通讯模块」，设定 AS00SCM-A IP 地址与 RTU 模式。

9.4.2.2.1 设定 AS00SCM-RTU 右侧 IO 模块配置

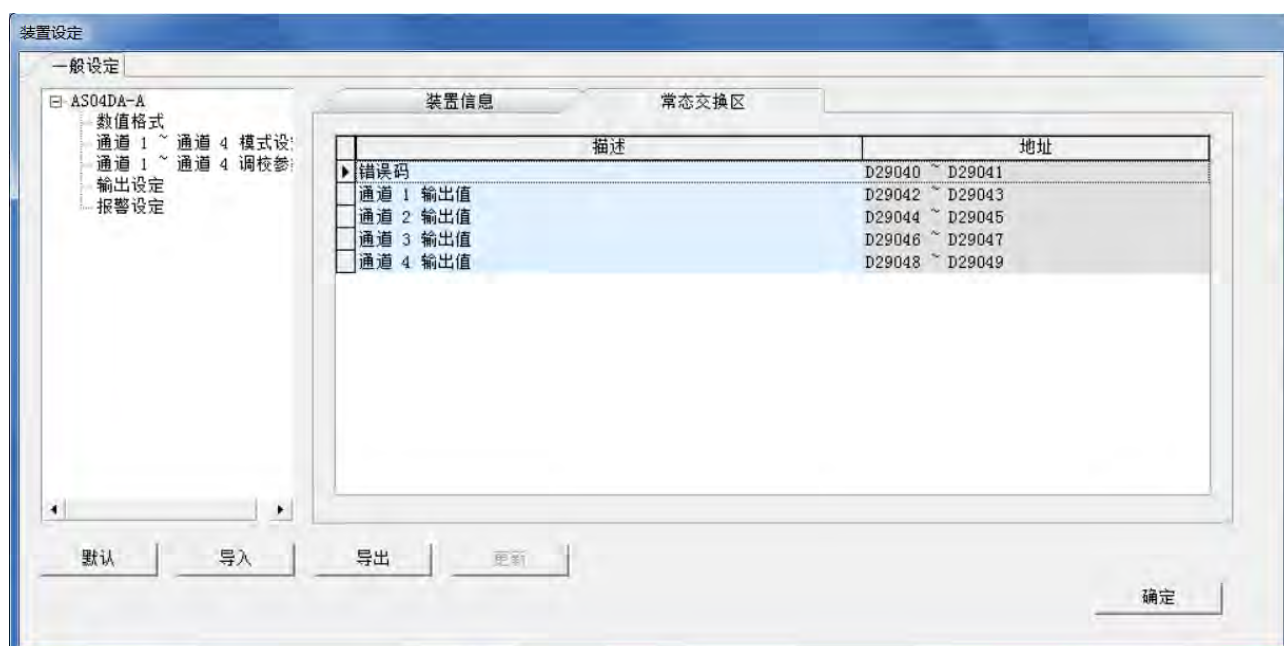
在连接 AS00SCM-RTU (AS-FEN02) 之前，需使用计算机并通过软件 EIP Builder 设定 AS00SCM-RTU (AS-FEN02) 右侧 IO 模块配置。

- 以下为 EIP Builder 操作步骤
 1. 手动配置或是自动扫描网络，将远程模块加入网络图，单击远程模块开启 HWCONFIG 即可扫描以及下载右侧模块配置。





- 记下常态交换区内容，后续在第三方设备上会以此顺序进行数据交换。以 AS04DA-A 为例：第一个输入值为错误码（所有模块的错误码皆为输入值，定义为远程模块输入至 Scanner），第一个到第四个输出值为通道 1 到通道 4 的输出值，单位皆为 REAL。

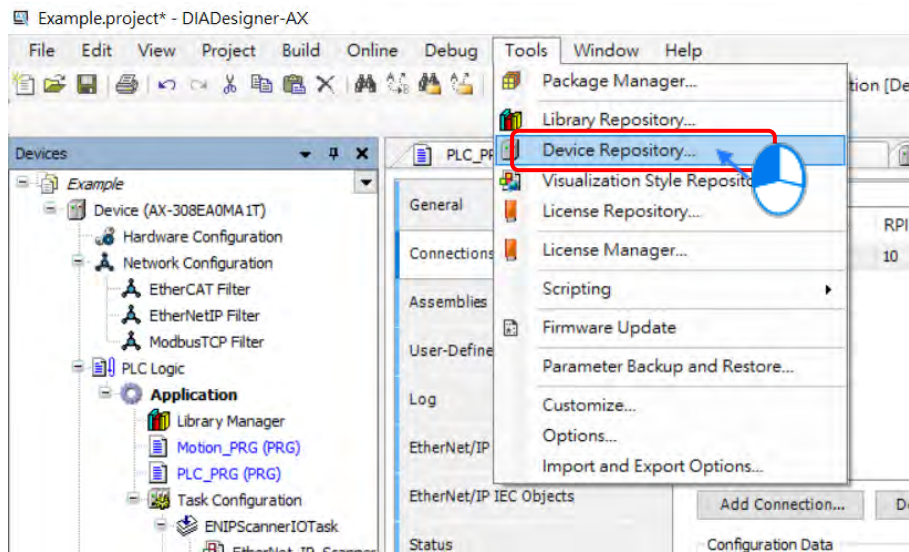


9.4.2.2.2 导入 AS00SCM-RTU EDS 文件

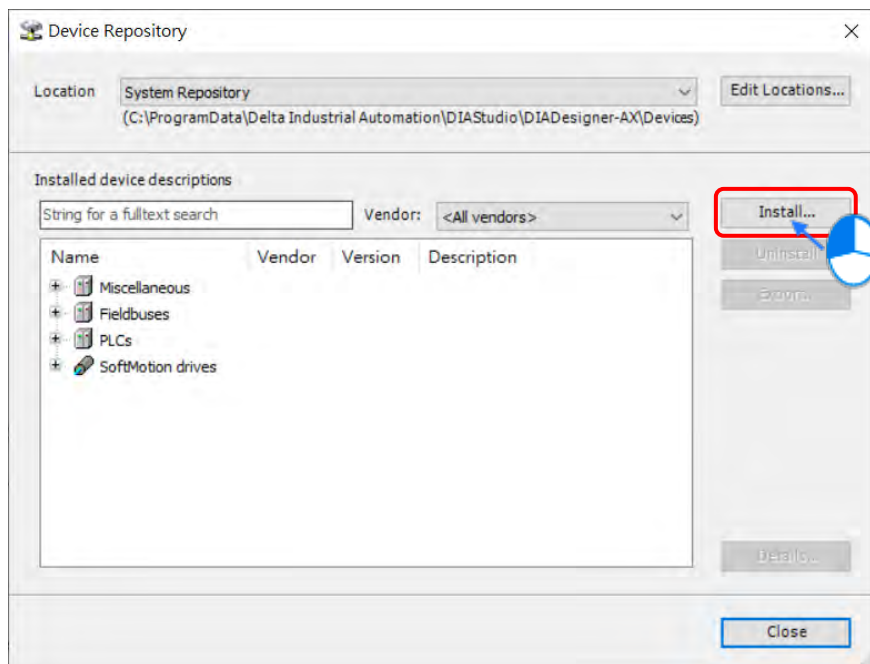
请先至台达官网下载 AS00SCM-RTU 模块装置参数文件 (EDS file)。

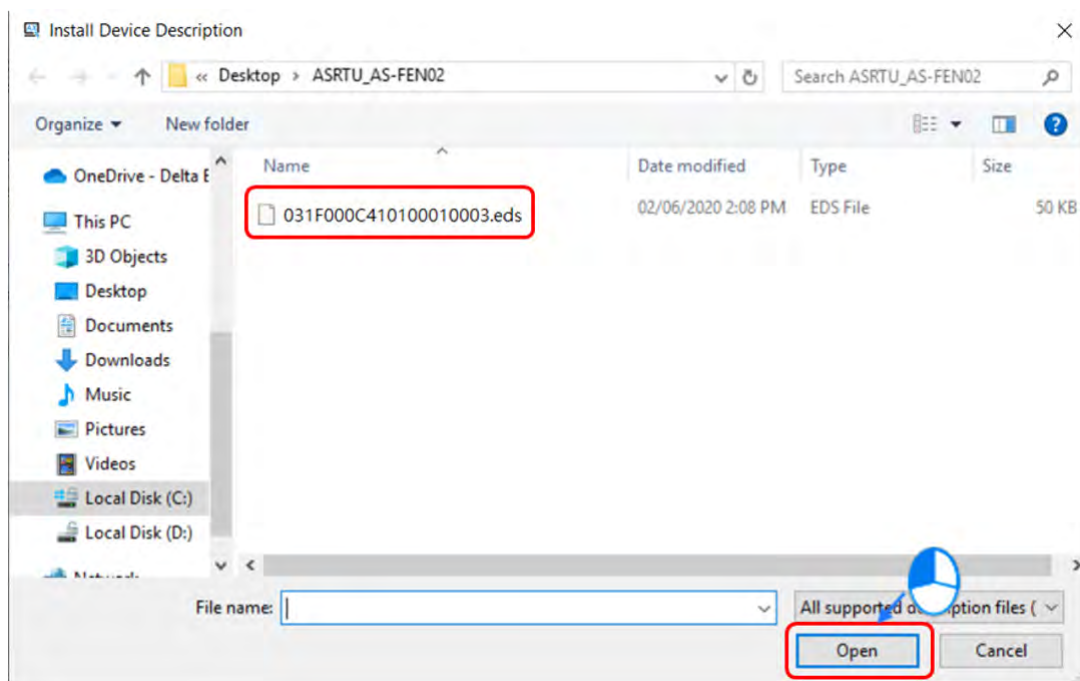
1. 安装装置参数文件

1.1 开启 Device Repository

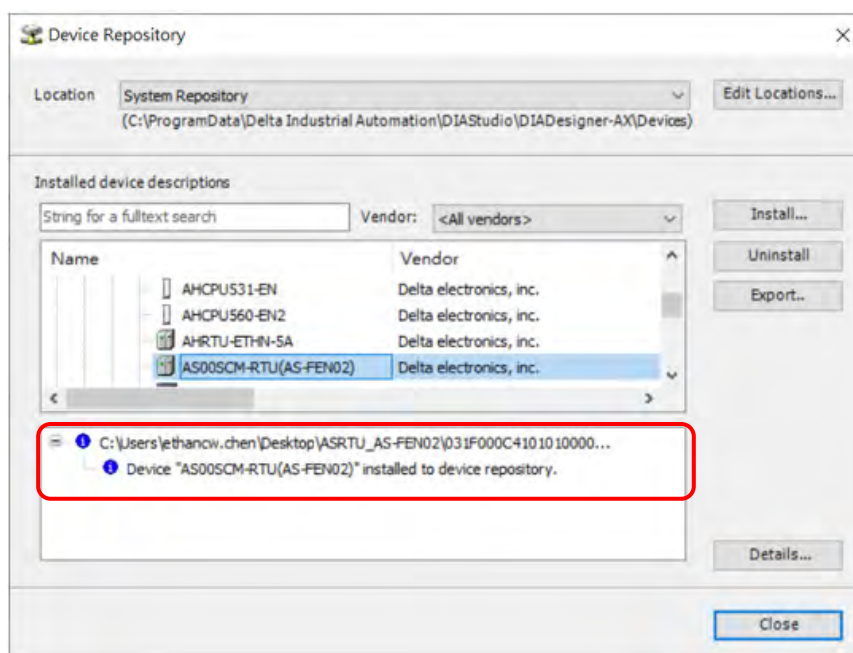


1.2 选择装置参数文件



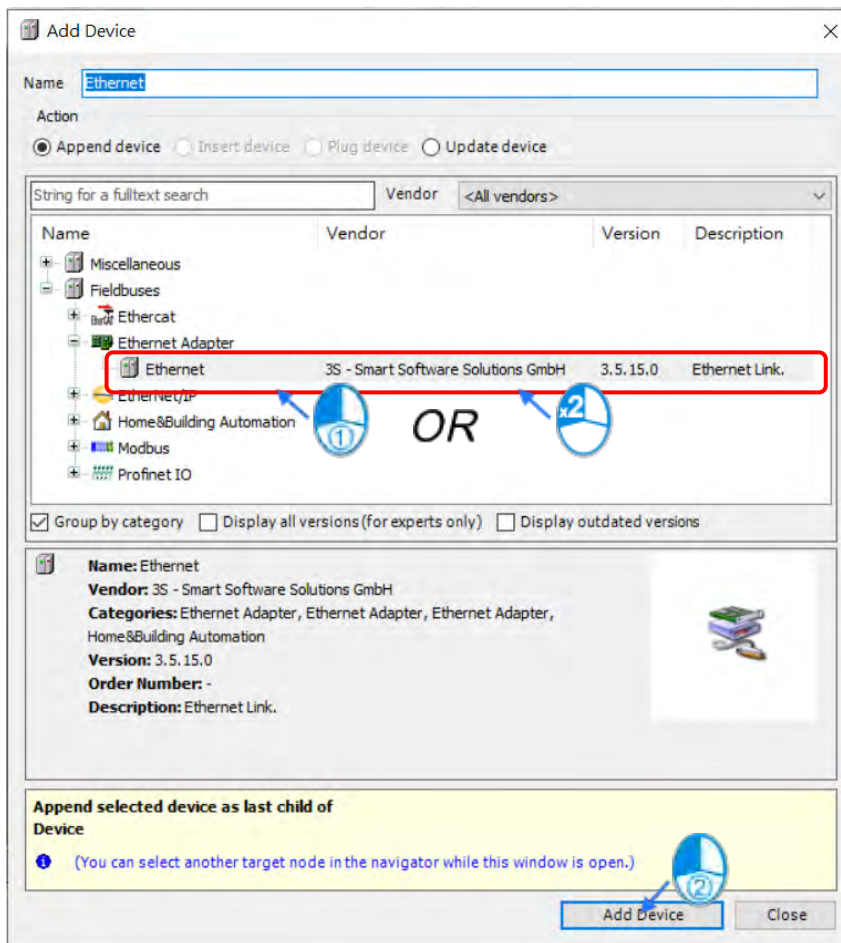
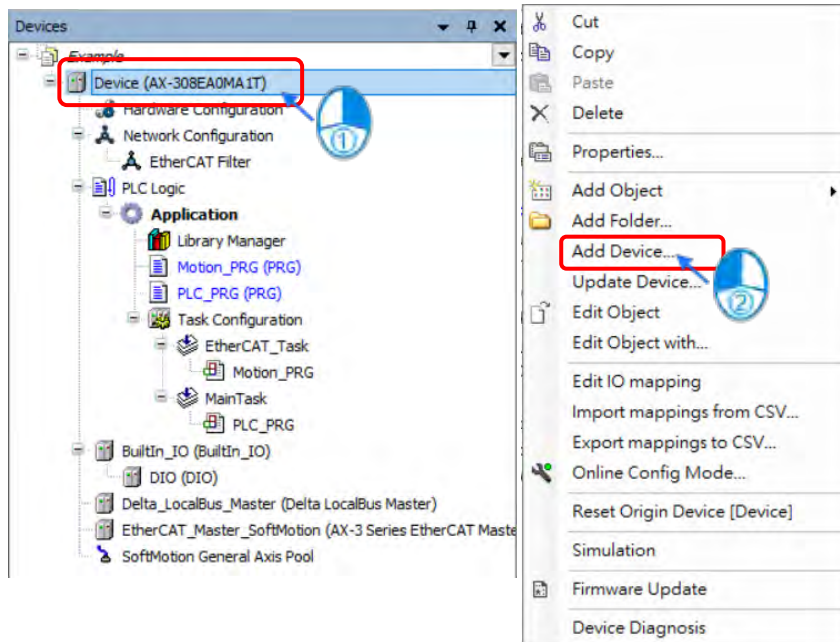


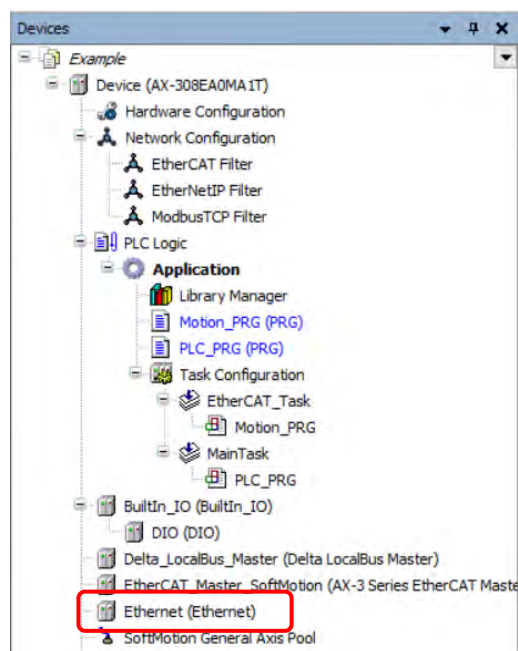
1.3 即完成装置参数文件导入



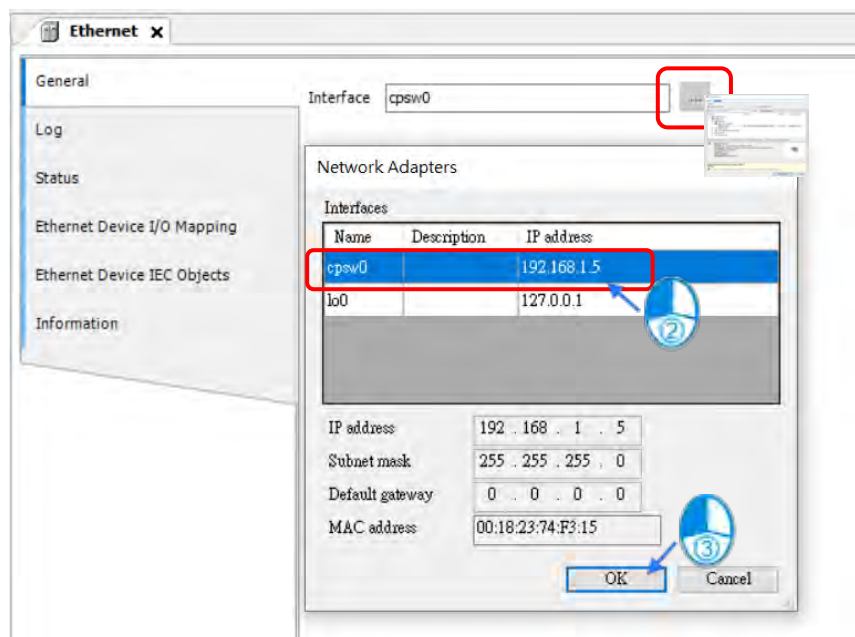
2. 裝置参数文件导入完毕后，即可增加 AS00SCM-RTU 装置

2.1 新增 Ethernet 装置

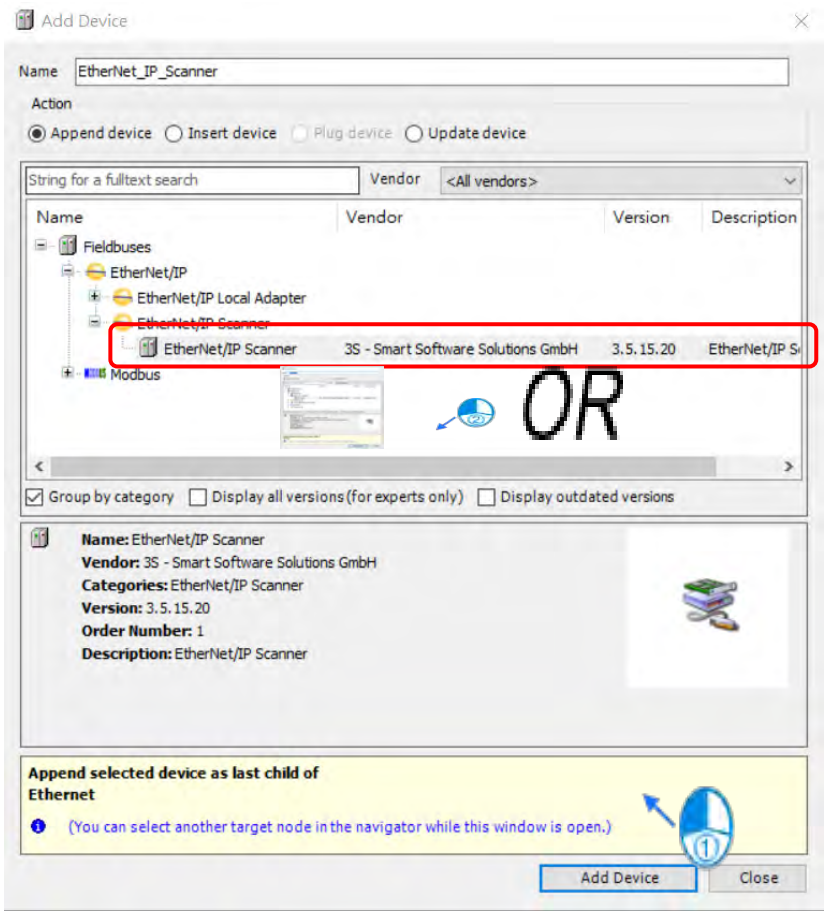
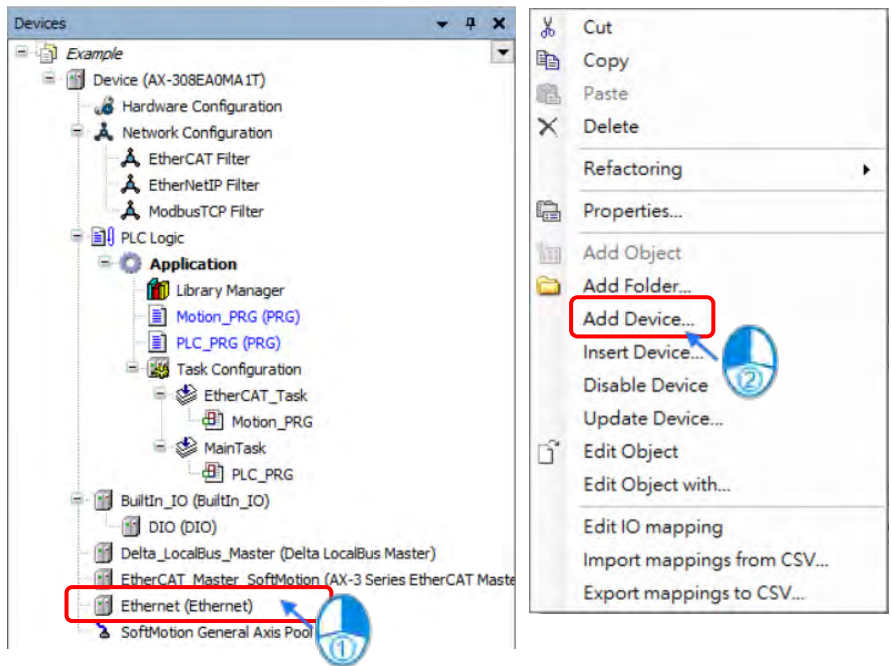


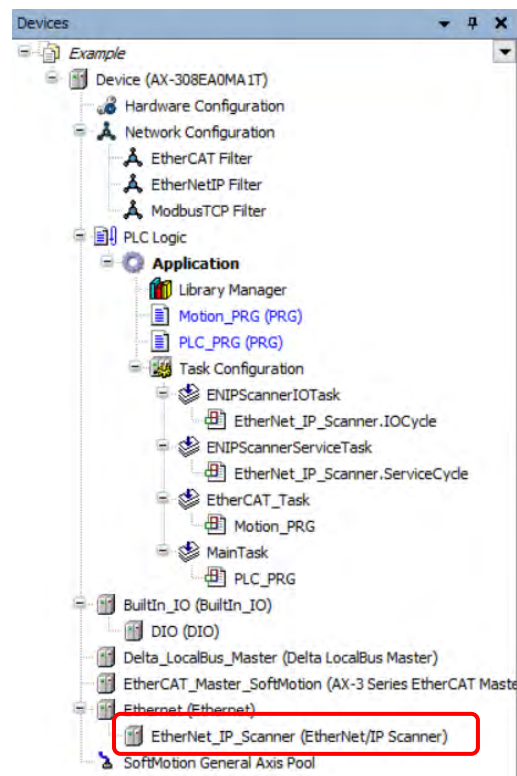


2.2 选择所欲使用的网络通讯接口

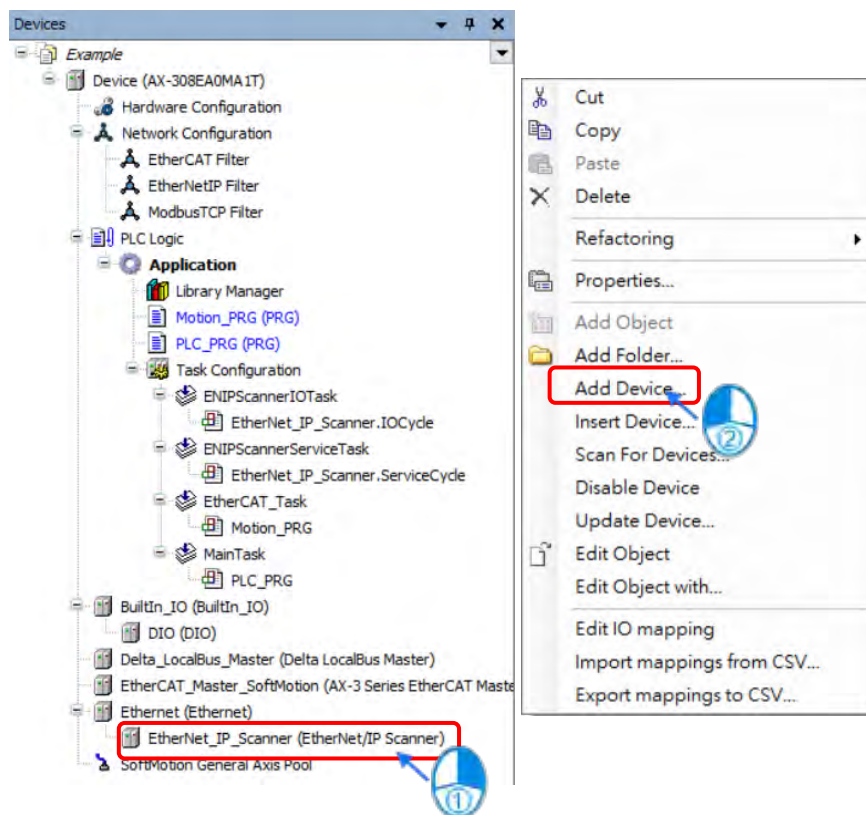


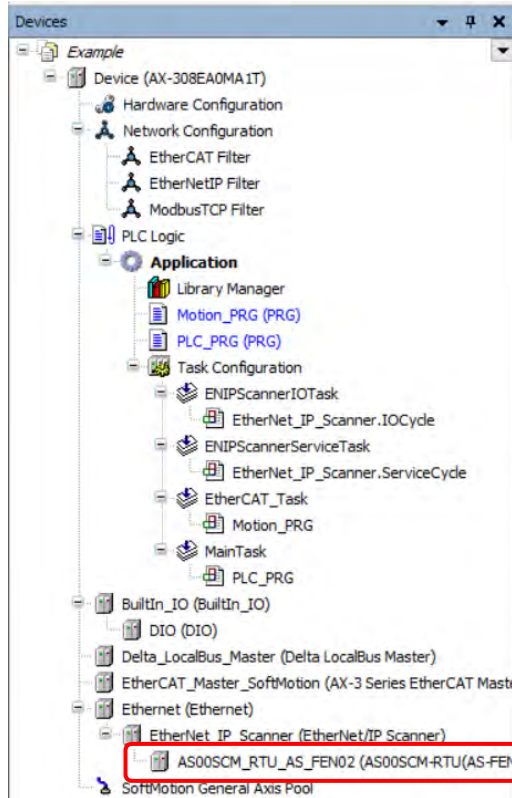
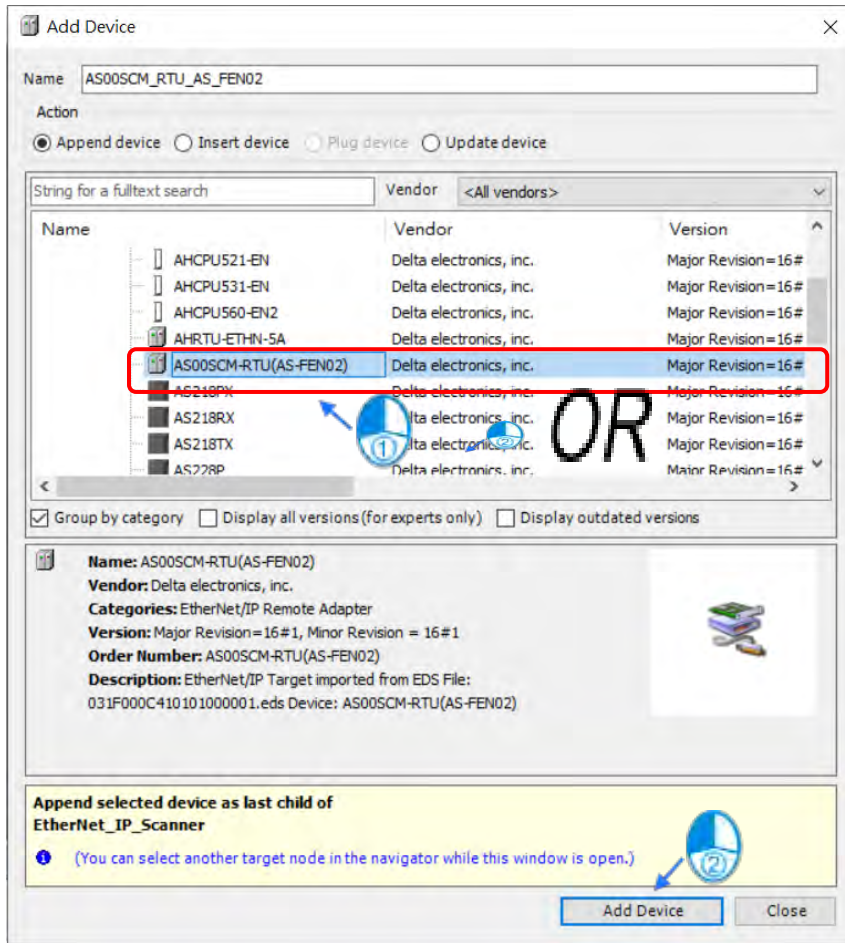
2.3 新增 EtherNet/IP Scanner 裝置





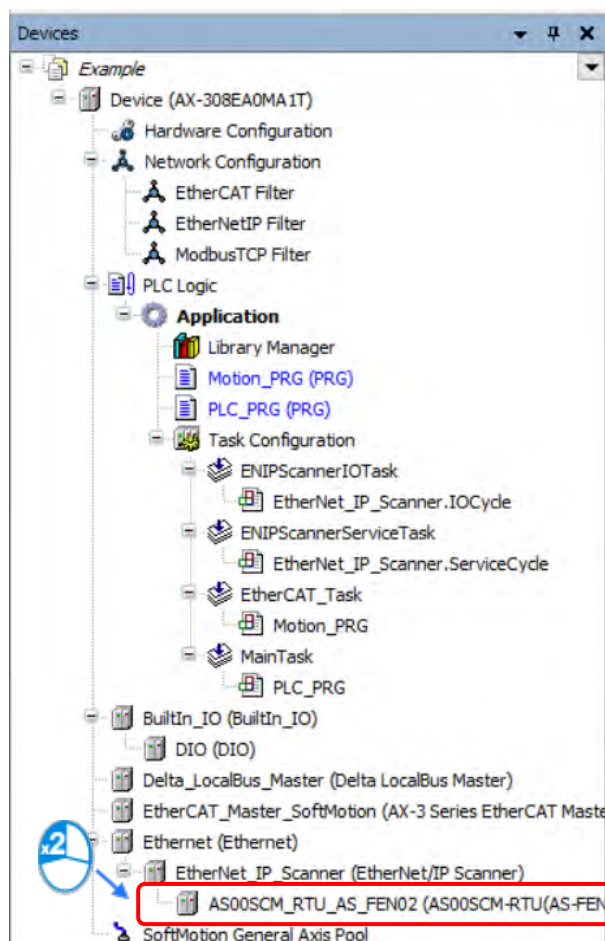
2.4 新增 EtherNet/IP Adapter (AS00SCM-RTU)



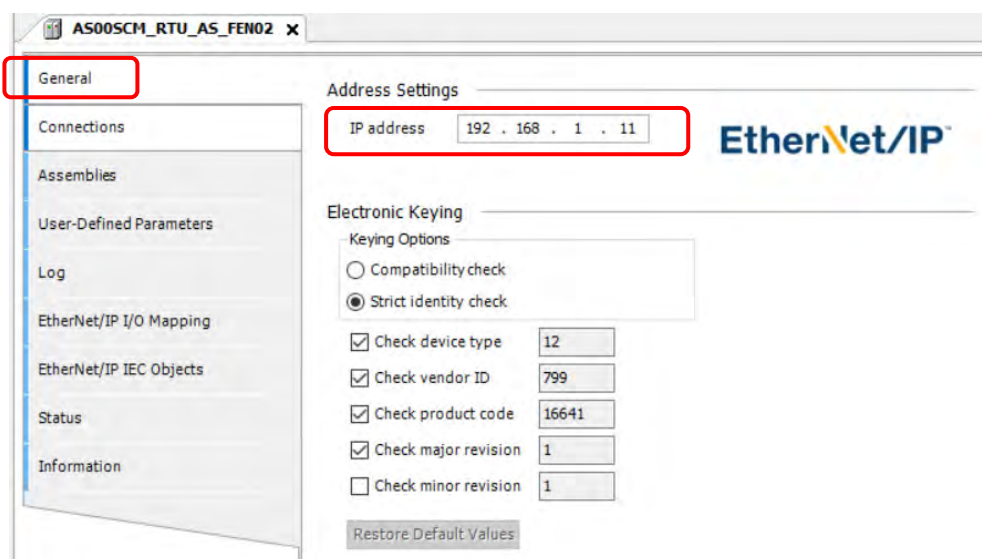


9.4.2.2.3 设定 AS00SCM-RTU 相关 EtherNet/IP 参数

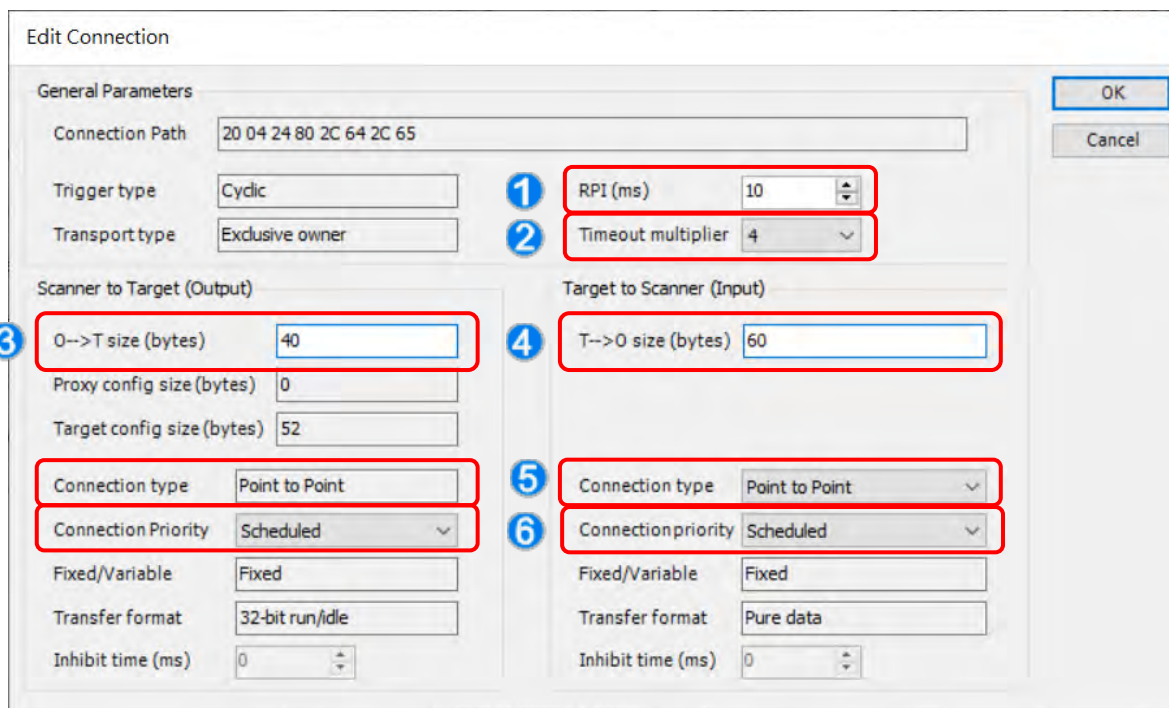
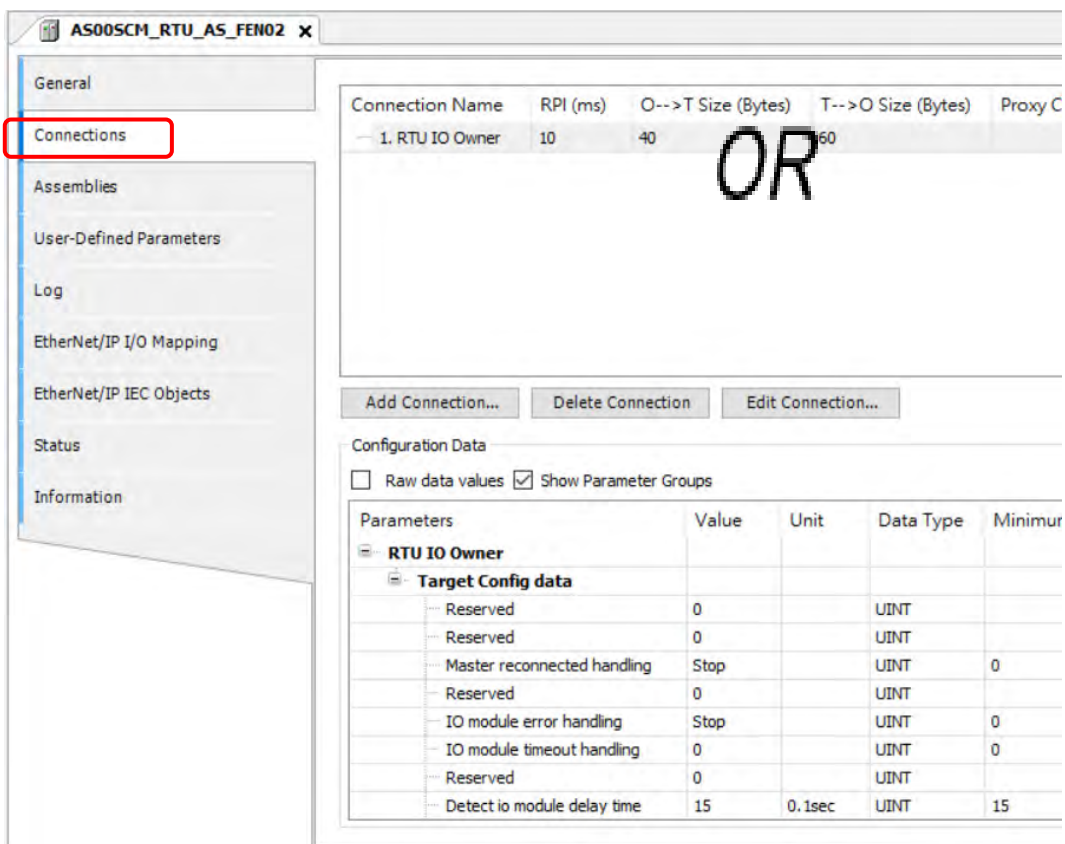
用户可于 AS00SCM-RTU 装置直接开启参数页面，并设定相关参数后进行下载，即可对 AS00SCM-RTU 右侧 IO 模块进行操作。



1. 设定 AS00SCM-RTU IP 地址



2. 设定 EtherNet/IP Connection 参数，这边要依照实际所配置的 IO 模块进行调整（参考第 9.4.2.2.1 节设定）



①：RPI：Requested Packet Interval，IO connection 周期性数据交换间隔时间。

②：Timeout multiplier：通讯超时时间设定，为更新时间（RPI）的倍数时间。

- ③ : O → T size (bytes) : Scanner 传送数据给 Adapter 的数据长度，对 Scanner 而言即为 Output 数据。
- ④ : T → O size (bytes) : Adapter 传送数据给 Scanner 的数据长度，对 Scanner 而言即为 Input 数据。
- ⑤ : Connection type : 可分成「Point to Point」与「Multicast」两种模式。
- ⑥ : Connection Priority : 联机优先权，AS00SCM-RTU 仅支持「Scheduled」模式。

注 1 : T → O size 与 O → T size 需要依照第 9.4.2.2.1 节所配置的右侧 IO 模块进行修改，各类型模块所需占用的数据长度可参考下列表格。

● DIO 模块 Input/Output 数据长度表

Digital I/O Module	T → O size bytes (Input)	O → T size bytes (Output)
AS08AM10N-A	2	0
AS08AN01T-A	0	2
AS08AN01P-A	0	2
AS08AN01R-A	0	2
AS16AM10N-A	2	0
AS16AP11T-A	2	2
AS16AP11P-A	2	2
AS16AP11R-A	2	2
AS16AN01T-A	0	2
AS16AN01P-A	0	2
AS16AN01R-A	0	2
AS32AM10N-A	4	0
AS32AN02T-A	0	4
AS64AM10N-A	8	0
AS64AN02T-A	0	8

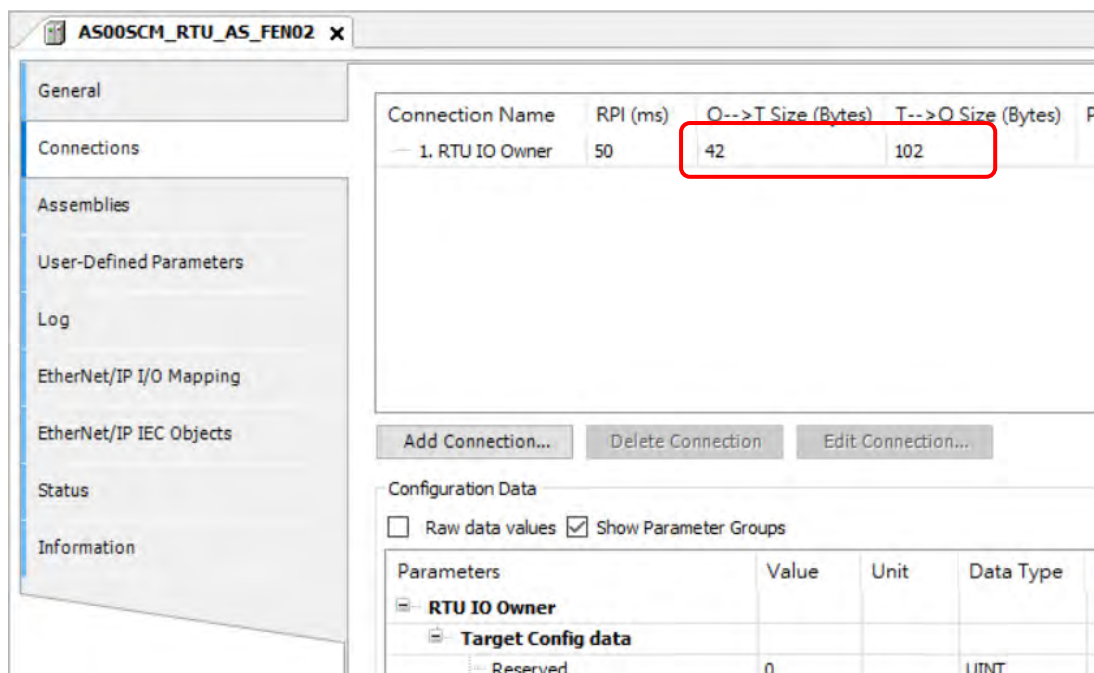
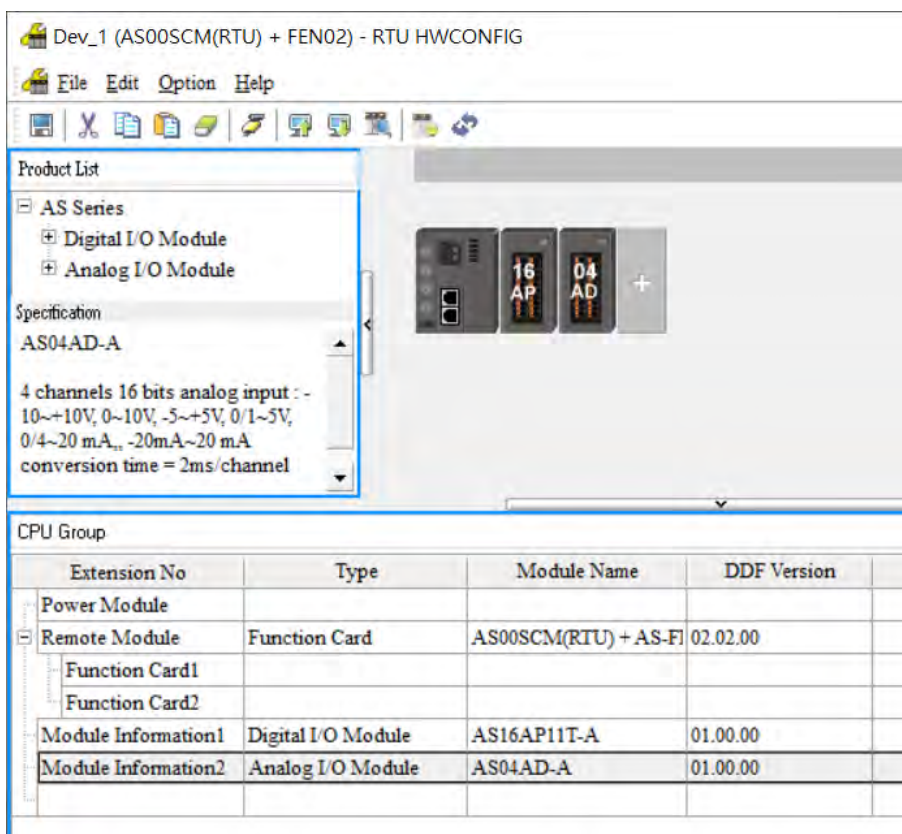
● AIO 模块 Input/Output 数据长度表

Analog I/O Module	T → O size bytes (Input)	O → T size bytes (Output)
AS02ADH-A	40	0
AS04AD-A	40	0
AS08AD-B	40	0
AS08AD-C	40	0
AS04DA-A	4	36
AS06XA-A	20	20
AS04RTD-A	40	0
AS06RTD-A	40	0
AS04TC-A	40	0
AS08TC-A	40	0

9.4.2.2.4 AS00SCM-RTU 右侧 IO 模块操作

将 EtherNet/IP 联机设定完毕后，可于「EtherNet/IP IO Mapping」页面找到 Input 与 Output data，用户即可于此操作 AS00SCM-RTU 右侧所配置的 IO 模块，如下图于 AS00SCM-RTU 右侧配置 AS16AP11T-A (T → O : 2 Bytes，O → T : 2 Bytes) 与 AS04AD-A 模块 (T → O : 40 Bytes，O → T : 0 Bytes)，则其 T →

O 总长度为 102 Bytes · O → T 总长度为 42 Bytes。



注 1：请注意 AIO 模块通道模式与其它相关参数，应于执行第 9.4.2.2.1 节设定完毕，于本章节仅对输入与输出通道进行读取或操作。

注 2：如果读取或是写入的数值为浮点数类型，则需要将其 High word 与 Low word 对换，方可显示正确数值。

AS005CM_RTU_AS_FEN02 x

General Connections Assemblies User-Defined Parameters Log **EtherNet/IP I/O Mapping** EtherNet/IP IEC Objects Status Information

Find Filter Show all

Variable	Mapping	Channel	Address	Type
+ [icon]		Reserved	%IW31	UINT
+ [icon]		Input_data30	%IW32	UINT
+ [icon]		Input_data31	%IW33	UINT
+ [icon]		Input_data32	%IW34	UINT
+ [icon]		Input_data33	%IW35	UINT
+ [icon]		Input_data34	%IW36	UINT
+ [icon]		Input_data35	%IW37	UINT
+ [icon]		Input_data36	%IW38	UINT
+ [icon]		Input_data37	%IW39	UINT
+ [icon]		Input_data38	%IW40	UINT
+ [icon]		Input_data39	%IW41	UINT
+ [icon]		Input_data40	%IW42	UINT
+ [icon]		Input_data41	%IW43	UINT
+ [icon]		Input_data42	%IW44	UINT
+ [icon]		Input_data43	%IW45	UINT
+ [icon]		Input_data44	%IW46	UINT
+ [icon]		Input_data45	%IW47	UINT
+ [icon]		Input_data46	%IW48	UINT
+ [icon]		Input_data47	%IW49	UINT
+ [icon]		Input_data48	%IW50	UINT
+ [icon]		Input_data49	%IW51	UINT
+ [icon]		Reserved	%QW1	UINT

Reset Mapping Always

[icon] = Create new variable [icon] = Map to existing variable

AS005CM_RTU_AS_FEN02 x

General Connections Assemblies User-Defined Parameters Log **EtherNet/IP I/O Mapping** EtherNet/IP IEC Objects Status Information

Find Filter Show all

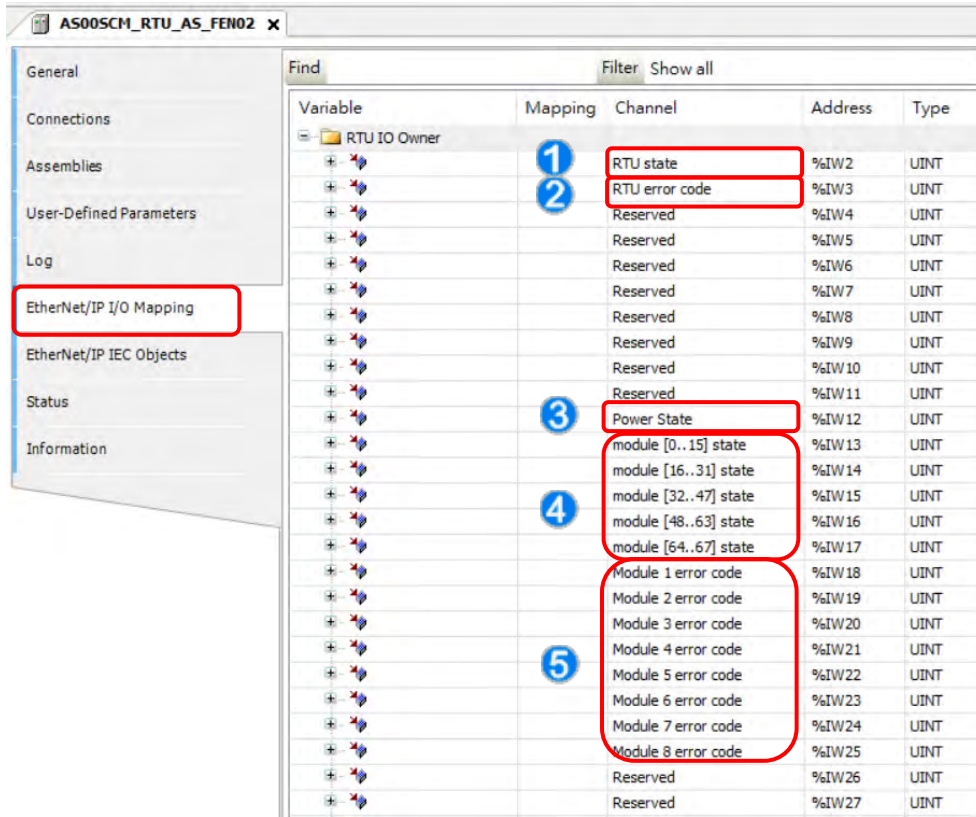
Variable	Mapping	Channel	Address	Type
+ [icon]		Reserved	%QW1	UINT
+ [icon]		Reserved	%QW2	UINT
+ [icon]		Reserved	%QW3	UINT
+ [icon]		Reserved	%QW4	UINT
+ [icon]		Reserved	%QW5	UINT
+ [icon]		Reserved	%QW6	UINT
+ [icon]		Reserved	%QW7	UINT
+ [icon]		Reserved	%QW8	UINT
+ [icon]		Reserved	%QW9	UINT
+ [icon]		Reserved	%QW10	UINT
+ [icon]		Reserved	%QW11	UINT
+ [icon]		Reserved	%QW12	UINT
+ [icon]		Reserved	%QW13	UINT
+ [icon]		Reserved	%QW14	UINT
+ [icon]		Reserved	%QW15	UINT
+ [icon]		Reserved	%QW16	UINT
+ [icon]		Reserved	%QW17	UINT
+ [icon]		Reserved	%QW18	UINT
+ [icon]		Reserved	%QW19	UINT
+ [icon]		Reserved	%QW20	UINT
+ [icon]		Output_data20	%QW21	UINT

Reset Mapping Always

[icon] = Create new variable [icon] = Map to existing variable

9.4.2.2.5 AS00SCM-RTU 模块参数信息

用户可以通过「EtherNet/IP IO Mapping」页面内的模块参数信息，对 AS00SCM-RTU 状态进行诊断。



- ①：RTU state：通讯模块状态（0 = 通讯模块正常，1 = 通讯模块发生异常）。
- ②：RTU error code：请参考 AS 系列模块手册第 9.7 节错误码。
- ③：Power State：通讯模块电源状态（0：电源正常，1：电源异常）。
- ④：Module state [0..67]：I/O 模块状态，以 bit 表示（0：正常运行，1：无法正常运行）。
- ⑤：Module error code：I/O 模块错误码，错误码相关信息，请参考各模块手册。

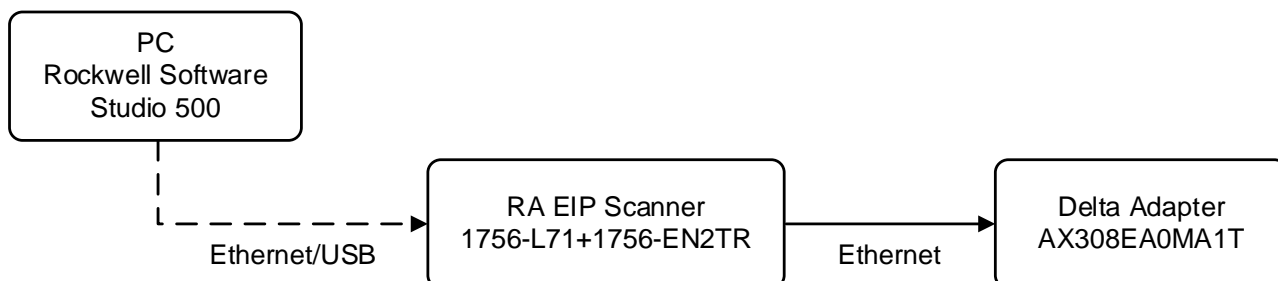
9.4.3 EtherNet/IP Adapter Function

9.4.3.1 Studio 5000 软件操作

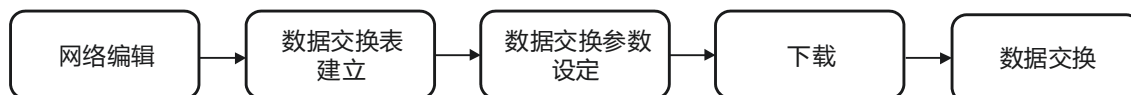
本节将介绍台达 EtherNet/IP Adapter 产品如何通过他牌软件使用 EtherNet/IP 连接，以下将以 Rockwell 软件为例。

9.4.3.1.1 架构

RA EIP Scanner 通过 Ethernet 连接 Delta Adapter，PC 端通过 Ethernet 或 USB 连接至 RA Scanner。



※ Rockwell Software Studio 5000、ControlLogix、RSLogix 是 Rockwell Automation 的商标或注册商标。
执行流程如下所示：



9.4.3.1.2 项目建立

- 开启 Studio 5000，于「Create」底下，开启「New Project」。
- 选择 PLC 型号，下面范例以 1756-L71 为例。
- 开启「Finish」，完成项目建立。
- 项目新增完毕，自动开启设定页面。

9.4.3.2 Scanner 建立

项目建立后，于 PLC 背板建立 EtherNet/IP 模块 (1756-EN2TR)，再通过 EtherNet/IP 模块建立欲连接的 EtherNet/IP 设备。

9.4.3.2.1 模块建立

- 开启「1756 Backplane 1756-A7」后右键单击，选择「New Module」。
- 于 Filter 字段输入「1756-EN2TR」，开启「Create」。
- 输入 Name 与 IP 等信息。输入完毕，按「OK」，即完成 EtherNet/IP 模块建立。
- 树形图中展出 1756-EN2TR 模块。

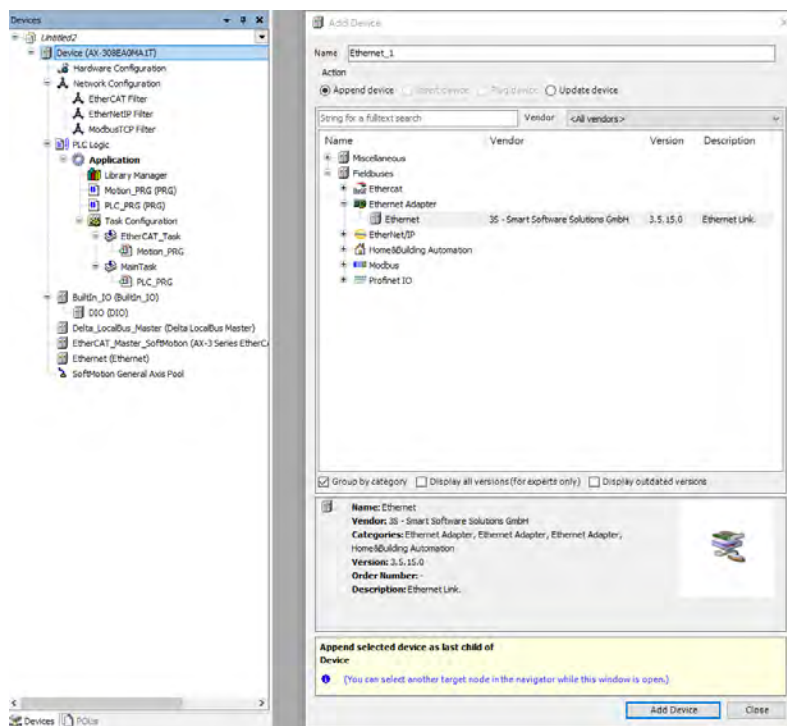
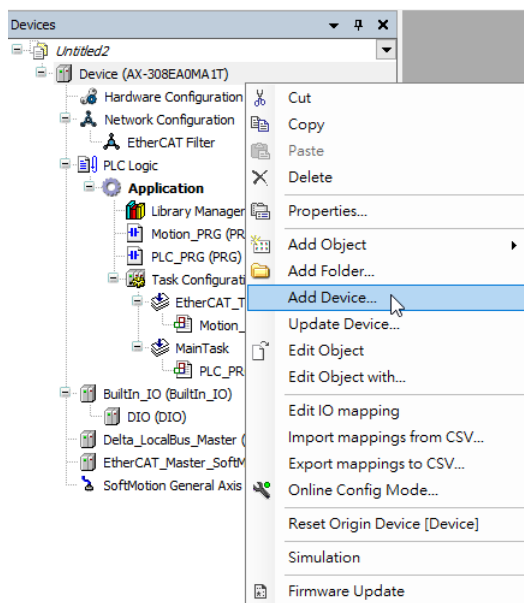
9.4.3.3 Adapter 联机

本节将介绍如何于 Studio 5000 中建立 AX-3 系列产品做为 EIP Adapter。

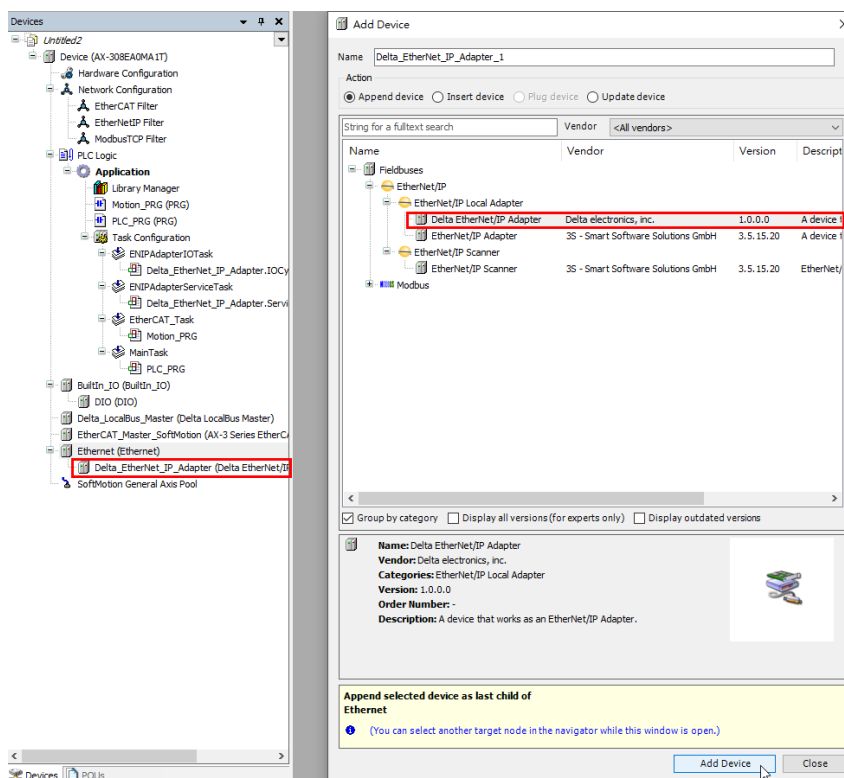
9.4.3.3.1 EDS 建立

本节将介绍 AX-3 系列主机如何建立 EDS 文件。

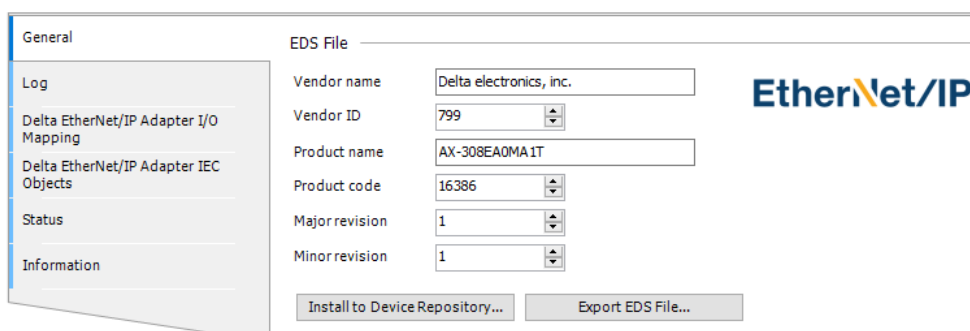
- 新增 Ethernet Device



● 新增 Delta_EtherNet_IP_Adapter Device



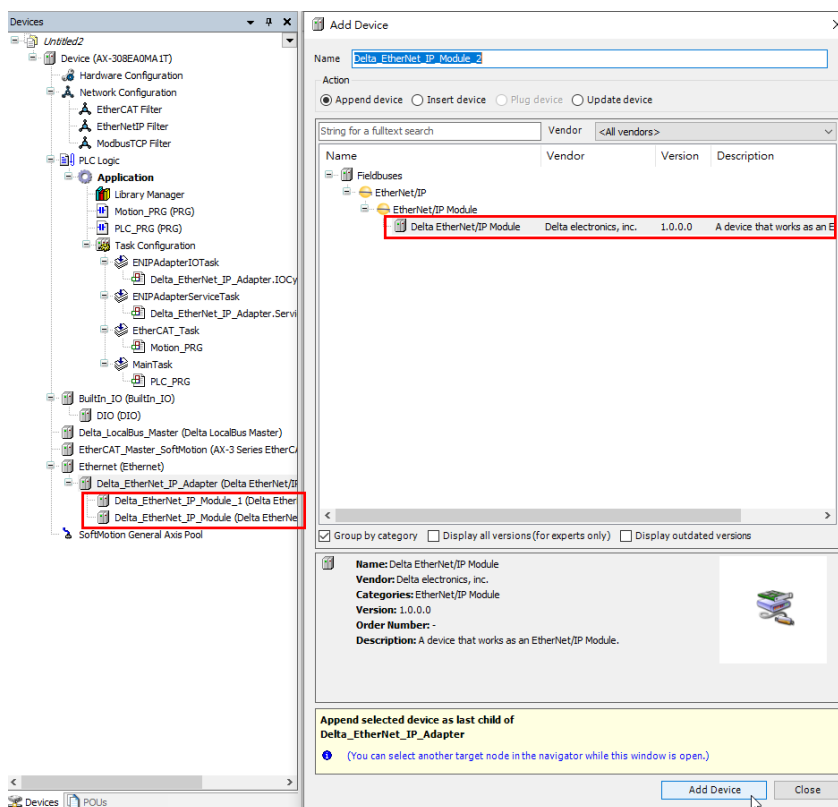
■ General – 设定 EDS File



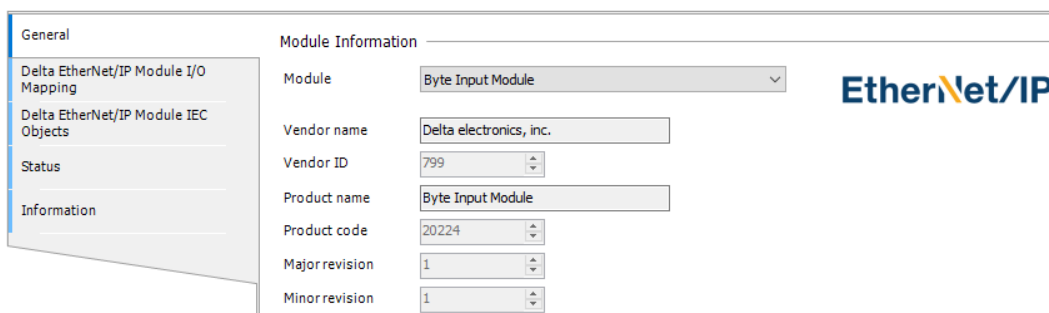
标题	说明	默认值
Vender name	供货商名称	Delta electronics, inc.
Vendor ID	供货商 ID	799
Product name	产品名称	AX-308EA0MA1T
Product code	产品代码	16386
Major revision	主要修订	1
Minor revision	次要修订	1

标题	说明	默认值
Install to Device Repository	如果已经安装了具有相同设备标识的设备，则会询问您是否应覆盖该设备。如果将设备作为远程 Adapter 插入 EtherNet/IP Scanner 下方，则将要求您自动更新设备。	
Export EDS File	将创建 EDS 文件并将其储存在本地端计算机上。这样，可以在外部配置文件中使用 EDS 文件。	

● 新增 Delta_EtherNet_IP_Module Device 建立数据交换长度



■ General-Module Information



■ Module

标题	说明
Byte Input Module	1 Byte 输入
Byte Output Module	1 Byte 输出
Word Input Module	1 Word 输入
Word Output Module	1 Word 输出
DWord Input Module	1 DWord 输入
DWord Output Module	1 DWord 输出
Real Input Module	1 Real 输入
Real Output Module	1 Real 输出
100 words Input Module	100 Words 输入
100 words Output Module	100 Words 输出

■ 建立 DWord Input Module 为例

读取 Scanner 中 1 DWORD 数据类型。需新增 Delta_EtherNet_IP_Module Device → Module → DWord Input Module

Module Information

Module: DWord Input Module

Vendor name: Delta electronics, inc.

Vendor ID: 799

Product name: DWord Input Module

Product code: 20228

Major revision: 1

Minor revision: 1

EtherNet/IP™

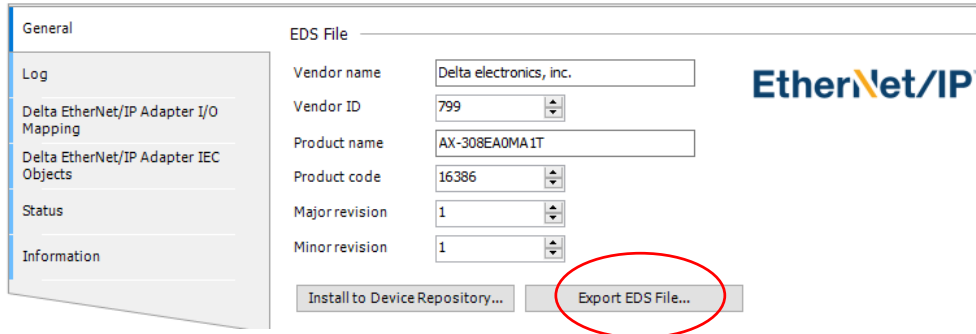
在 I/O Mapping 分页中，可读取 Scanner 之 1 DOWRD 数据内容。

Find Filter Show all Add FB

Variable	Mapping	Channel	Address	Type	Current Value
7409		Input Data	%ID2	DWORD	

● Export EDS File

设定完成后，导出 EDS File。将 EDS file – AX-308EA0MA1T.eds 存放至计算机中。



9.4.3.3.2 EDS 载入

- Tools < EDS Hardware Installation Tool 以开启载入工具
- 选择「Register an EDS file (s)」。
- 选择 Register a single file > Browse 选择欲载入的 EDS 文件：AX-308EA0MA1T.eds。
- 依指示开启「下一步」，直至 EDS 建立完成。

9.4.3.3.3 Adapter 建立 EDS 加载

- 在项目树中 EtherNet/IP Scanner 模块下，开启「Ethernet」，并单击右键以选择「New Module」。
- 输入已导入 EDS 文件之台达产品型号，出现后选取该型号（如 AX-308EA0MA1T），并开启「Create」。
- 输入产品名称与 IP，检查 Module Definition 信息是否与产品一致。
- 若要修改 Connections 信息，可单击 Module Definition 中「Change」开启修改画面。
- 编辑 Connections
 - (1) Name：开启旁边箭头会列出设备所有可使用的 Connection
 - (2) Size：数字表示 Input/Output 数据交换长度。
- ※ 若非进阶使用，默认 EDS 文件加载之设定可直接连接，无须修改。
- 选择 Connection 页签，可修改 RPI 设定与 Input Type。RPI 设定为与 Scanner 周期性数据交换周期时间，单位为 ms。Input Type 可由 Unicast 与 Multicast 择一（依产品是否提供此功能显示）。
- 设定完毕后，开启 OK，完成新增 Adapter，项目树中出现台达 Adapter 型号。

9.4.3.3.4 下载

台达 Adapter 设备新增完毕后，将项目设定下载到 PLC 并 Online。

- 开启 Communications > Who Active 后，选择 PC 端连接 Scanner 的型号，再开启 Communications > Download 以开始建立联机。
- 若实体联机皆正确，左上角状态会显示 I/O OK。

9.4.3.3.5 数据交换

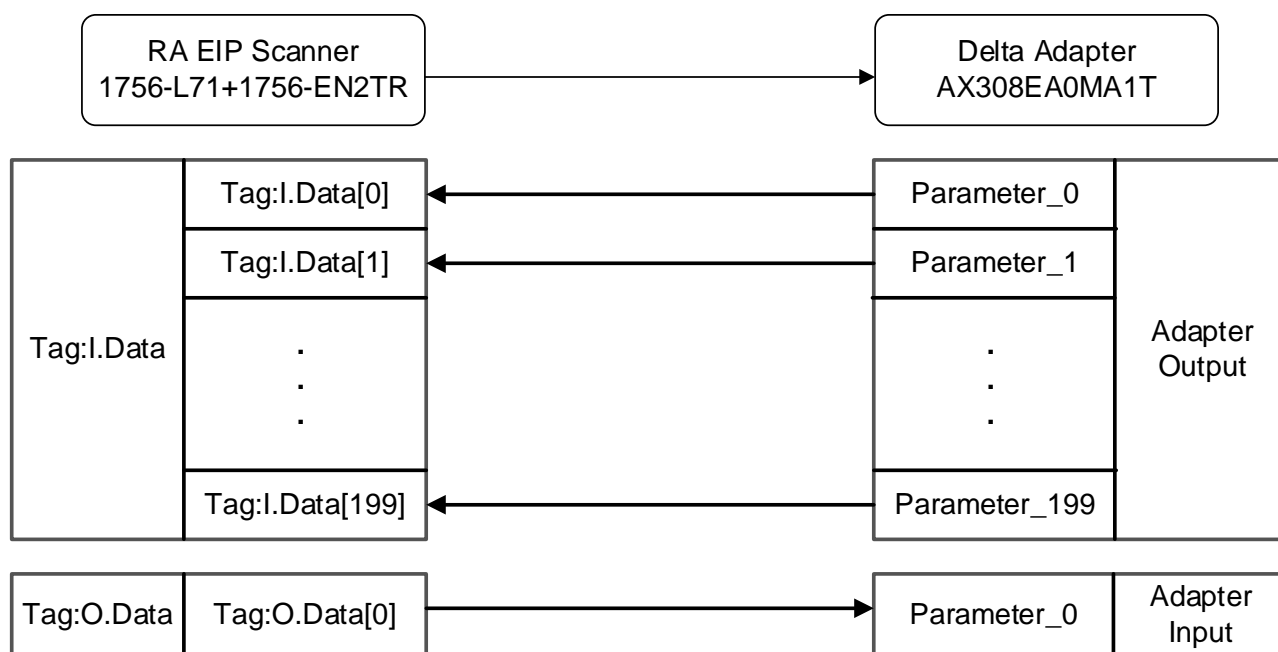
单击项目树中 Program TAGs，进行数据交换的定义与处理，分为 Configure、Input 及 Output 三种 TAG。于 IO Configuration 中建立设备的网络关系后，自动建立 TAG。

- 开启 Program TAGs
- 于右侧窗口依据产品名称出现对应 TAG : C，TAG : I1，TAG : O1

TAG : C 为依据 Adapter EDS file 产生的对应数据内容信息，包含 Input 和 Output。若产品支持对应参数可修改，可于此修改 Input 与 Output 数据对应内容。

TAG : I1 对应数据由 TAG : I1[0]开始，对应至 Adapter Output 第一个参数，长度为 Adapter 中提供的 Output 长度。

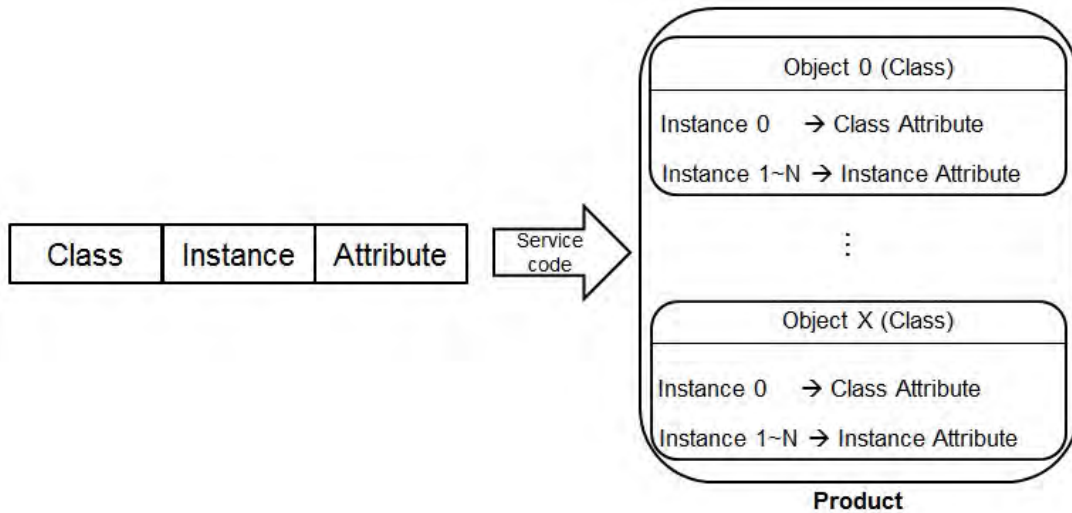
TAG : O1 对应数据由 TAG : O1[0]开始，对应至 Adapter Input 第一个参数，长度为 Adapter 中提供的 Input 长度。



9.4.4 CIP Object

9.4.4.1 Object list

EtherNet/IP 中以 Object 作为参数的集合，各 Object 依 Class、Instance 和 Attribute 的结构定义参数，其中 Instance 0 定义了各 Object 的基本信息，如版本与长度。Instance 1~N 则为各产品建立联机或状态所需的参数内容。用户可通过各 Object 所支持的 Service code 存取产品相关参数。如下图：



可使用 EtherNet/IP Services.library 或是显性报文工具读取与写入 Object，详细 Object 对应请参照后续章节。产品所支持的 EtherNet/IP Object 如下，参数数据类型定义于第 9.4.4.2 节，各 Object 内容说明于第 9.4.4.3 节~第 9.4.4.6 节。

Object Name	功能	Class ID
Identity Object	描述装置信息，包含制造商信息、装置类型与版本信息	1 (H'01)
Assembly Object	定义 I/O Connection 数据交换功能的参数	4 (H'04)
TCP/IP Interface Object	显示 IP 设定方式与 IP 设定接口	245 (H'F5)
Ethernet Link Object	显示设备上每个 Ethernet port 连接状态	246 (H'F6)

9.4.4.2 数据类型定义 (Data Type)

本节介绍各 Object 支持的数据类型。

数据类型	描述
BOOL	False (H'00) or True (H'01)

数据类型	描述										
SIGNED INTEGER	SINT (1 byte) · INT (2 bytes) · DINT (4 bytes) · LINT (8 bytes)										
	Number	1st	2nd	3rd	4th	5th	6th	7th	8th		
	SINT	0LSB	--	--	--	--	--	--	--		
	INT	0LSB	1LSB	--	--	--	--	--	--		
	DINT	0LSB	1LSB	2LSB	3LSB	--	--	--	--		
	LINT	0LSB	1LSB	2LSB	3LSB	4LSB	5LSB	6LSB	7LSB		
Ex : DINT value = H'12345678											
		Number	1st	2nd	3rd	4th					
		DINT	78	56	34	12					
UNSIGNED INTEGER	USINT (1 byte) · UINT (2 bytes) · UDINT (4 bytes) · ULINT (8 bytes)										
	Ex : UDINT value = H'AABBCCDD										
	Number	1st	2nd	3rd	4th						
UDINT	DD	CC	BB	AA							
STRING	ASCII 字符 · 1 or 2 bytes/字符										
	STRING : 2 bytes character count + 1 byte character										
		Contents (Charcount)			Contents (String contents)						
	STRING	04	00	4D	69	6C	6C				
	STRING2 : 2 bytes character count + 2 byte character										
		Contents (Charcount)			Contents (String contents)						
	STRING2	04	00	4D	00	69	00	6C	00	6C	00
	SHORT_STRING : 1 bytes character count + 1 byte character										
		Contents (Charcount)			Contents (String contents)						
	STRING	04		4D	69	6C	6C				
Fixed LENGTH BIT STRING	BYTE (1 byte) · WORD (2 bytes) · DWORD (4 bytes) · LWORD (8 bytes)										
		1st	2nd	3rd	4th	5th	6th	7th	8th		
	Byte	7...0	--	--	--	--	--	--	--		
	WORD	7...0	15...8	--	--	--	--	--	--		
	DWORD	7...0	15...8	23...16	31...24	--	--	--	--		
LWORD	7...0	15...8	23...16	31...24	39...32	47...40	55...48	63...56			

数据类型	描述							
STRINGI	A single string consists multiple language representation							
	Name		Data Type		Meaning			
	Number		USINT		The number of internationalized character strings			
	Strings		Array of : Struct of :		Array of individual internationalized character strings			
	LanguageChar1		USINT		The first ASCII character of the ISO 639-2/T language			
	LanguageChar2		USINT		The second ASCII character of the ISO 639-2/T language			
	LanguageChar3		USINT		The third ASCII character of the ISO 639-2/T language			
	CharStringStruct		USINT		The structure of the character string · limited to the Elementary Data type value 0xD0 (STRING) · 0xD5 (STRING2) · 0xD9 (STRINGN) and 0xDA (SHORT_STRING)			
	CharSet		UINT		The character set which the character string is based on which comes from IANA MIB Printer Code (RFC 1759) .			
	InternationalString		Defined in CharStringStruct		An array of 8-bit octet elements which is the actual international character string			
ISO 639-2/T language :								
Language		First Character		Second Character		Third Character		
English		e		n		G		
French		f		r		e		
Spanish		s		p		a		
Italian		i		t		a		
STRUCT	STRUCT of : Any Data Type composes the structure.							
	Ex. : STRUCT of { BOOL · UINT · DINT } = { TRUE · H'1234 · H'56789ABC }							
	Byte	01	34	12	BC	9A	78	56
ARRAY	Array of : Any Data Type composes the array.							
	Ex. : ARRAY of UINTs = { 1 · 2 · 3 }							
	Number	1st	2nd	3rd	4th	5th	6th	
Array	01	00	02	00	03	00		

数据类型	描述
EPATH	It's a path that consists of multiple segments and references the class, instance and attribute of another object. Ex. : Identity Object, Instance attribute 5 = " 20 01 24 01 30 05 "

9.4.4.3 Identity Object (Class ID : 01 Hex)

辨识产品身分对象，包含制造商信息、装置类型与版本等信息。

- Service Code

Service code	Service 名称	支持		描述
		Class Attribute	Instance Attribute	
H'01	Get_Attributes_All	X	V	读取对象多笔 Attribute 内容
H'05	Reset	X	V	执行 Reset
H'0E	Get_Attribute_Single	X	V	读取单一 Attribute 内容

- Class

- Class ID : H'01

- Instance

- H'01 : Instance Attribute

- Instance =1 时，Instance Attribute 如下所示：

Instance Attribute	名称	存取	数据类型	值	描述
H'01	Vendor ID	Get	UINT	H'31F	厂商代码：Delta Electronics · inc.
H'02	Device Type	Get	UINT	H'0C	数据类型：Communication Adatper
H'03	Product Code	Get	UINT	H'4002	产品代码
H'04	Revision	Get	STRUCT	--	设备版本，显示方式：Major.Minor
	Major Revision		USINT	H'01	主版本 Range：H'01~H'7F
	Minor Revision		USINT	H'01	次版本 Range：H'01~H'FF
H'05	Status	Get	WORD	H'64	状态描述，请参考※1
H'06	Serial Number	Get	UDINT	H'2374F75C	序号：MAC 地址末八码 23 : 74 : f7 : 5C
H'07	Product Name	Get	SHORT_STRING	产品名称，最大 32 字符 (数据长度+产品名称) (H'0D) AX-308EA0MA1T	

※1 状态说明 (H'05)

Bit (s)	名称	描述
0	Owned	显示设备是否已建立 Owner 联机 0 : 未建立 1 : 已建立
1	Reserved	0 : Always OFF
2	Configured	显示设备是否已设定 0 : 未设定 1 : 已设定
3	Reserved	0 : Always OFF
4-7	Extended Device Status	设备进阶状态 0 : Self-Testing 1 : Firmware Update 2 : At least one faulted I/O connection 3 : No I/O connections established 4 : Non-Volatile Configuration bad 5 : Major Fault 6 : At least one I/O connection in run mode 7 : At least one I/O connection established · all in idle mode 8-15 : Reserved
8	Minor Recoverable Fault	可恢复的 Minor 错误 0 : 未检测到 Minor 错误 1 : 检测到可恢复的 Minor 错误
9	Minor Unrecoverable Fault	不可恢复的 Minor 错误 0 : 未检测到 Minor 错误 1 : 检测到不可恢复的 Minor 错误
10	Major Recoverable Fault	可恢复的 Major 错误 0 : 未检测到 Major 错误 1 : 检测到可恢复的 Major 错误
11	Major Unrecoverable Fault	不可恢复的 Major 错误 0 : 未检测到 Major 错误 1 : 检测到不可恢复的 Major 错误

9.4.4.4 Assembly Object (Class ID : 04 Hex)

设备自定义对象，定义 IO connection 数据交换相关参数。

- Service Code

Service Code	Service 名称	支持		描述
		Class Attribute	Instance Attribute	
H'0E	Get_Attribute_Single	X	V	读取单一 Attribute 内容

- Class

- Class ID : H'04

- Instance

- H'64 : Output assembly
- H'65 : Input assembly
- H'66 : Dummy (needed for compatibility)
- Instance = 64~66 时，Instance Attribute 如下所示：

Instance Attribute	名称	存取	数据类型	值	描述
H'03	Data	Get	ARRAY of BYTE	H'2	数据内容

- Object 读写范例

(1) 读取 Output assembly 数据内容，数据填写如下：

Service code : H'0E

Class ID : H'04

Instance ID : H'64

Attribute ID : H'03

(2) 读取 Input assembly 数据内容，数据填写如下：

Service code : H'0E

Class ID : H'04

Instance ID : H'65

Attribute ID : H'03

9.4.4.5 TCP/IP Interface Object (Class ID : F5 Hex)

- Service Code

Service Code	Service 名称	支持		描述
		Class Attribute	Instance Attribute	
H'0E	Get_Attribute_Single	V	V	读取单一 Attribute 内容
H'10	Set_Attribute_Single	X	V	修改单一 Attribute 内容值

- Class
 - Class ID = H'F5
- Instance
 - H'00 : Class Attribute
 - H'01 : Instance Attribute
 - Instance = 0 时 · Class Attribute 如下所示 :

Class Attribute	名称	存取	数据类型	值	描述
H'01	Revision	Get	UINT	H'4	Object 版本

- Instance =1 时 · Instance Attribute 如下所示 :

Instance Attribute	名称	存取	数据类型	值	描述
H'01	Status	Get	DWORD	H'2	IP 设定状态 ※1
H'02	Configuration Capability	Get	DWORD	H'20	支持 IP 设定方式 ※2
H'03	Configuration Control	Get/Set	DWORD	H'0	设定 IP 模式 ※3
H'04	Physical Link Object :	Get	STRUCT of	--	Path to physical link object
	Path Size		UINT	H'0	Size of Path
	Path		EPATH	--	Logical segments identifying the physical link object
H'05	Interface Configuration :	Get/Set	STRUCT of	--	TCP/IP network interface configuration.
	IP Address		UDINT	192.168.1.5	设备 IP 地址
	Network Mask		UDINT	255.255.255.0	设备网络屏蔽
	Gateway Address		UDINT	0	设备默认网关地址
	Name Server		UDINT	0	Primary name server
	Name Server 2		UDINT	0	Secondary name server
	Domain Name		STRING	00 00	Default domain name
H'06	Host Name	Get	STRING	AX-308EA0MA1T	设备名称
H'13	Encapsulation Inactivity Timeout	Get/Set	UINT	120	EIP 设备联机保持时间 · 单位 : 秒 · 数值范围 : 0~3600 。

※ 当设备有主站正在通讯时 · Instance Attribute H'03 和 H'05 无法执行写入

- Object 读写范例
 - (1) 读取 Instance Attribute H'03 数据内容 · 数据填写如下 :
 - Service code : H'0E
 - Class ID : H'F5

Instance ID : H'01

Attribute ID : H'03

(2) 写入 Instance Attribute H'05 数据内容 · 数据填写如下 :

Service code : H'10

Class ID : H'F5

Instance ID : H'01

Attribute ID : H'05

Data Byte[0~3] : IP Address=192.168.1.5

Byte[4~7] : Network Mask=255.255.255.0

Byte[8~11] : Gateway Mask=0.0.0.0

Byte[12~15] : Name Server =0

Byte[16~19] : Name Server2 =0

※1 Interface 状态栏表

Status	描述
0	Interface Configuration attribute 尚未设定
1	The Interface Configuration attribute 由 BOOTP · DHCP or 非断电保持储存
2	Interface Configuration attribute 已由硬件配置

※2 Interface capability flags

Bit	描述
0	BOOTP Client
1	DNS Client
2	DHCP Client
3	DHCP-DNS Update
4	Configuration Settable
5	Hardware Configurable
6	Interface Configuration Change Requires Reset

※3 Interface Configuration Control

Status	描述
0	设备由硬件旋钮或非断电保持内存设定 IP 地址
1	设备由 BOOTP 设定
2	设备由 DHCP 设定

9.4.4.6 Ethernet Link Object (Class ID : F6 Hex)

● Service Code

Service Code	Service 名称	支持		描述
		Class Attribute	Instance Attribute	
H'0E	Get_Attribute_Single	V	V	读取单一 Attribute 内容

● Class

- Class ID : H'F6

● Instance

- H'00 : Class Attribute
- H'01 : Instance Attribute

• Instance = 0 时 · Class Attribute 如下所示 :

Class Attribute	名称	存取	数据类型	值	描述
H'01	Revision	Get	UINT	H'04	Object 版本

• Instance =1 时 · Instance Attribute 如下所示 :

Instance Attribute	名称	存取	数据类型	值	描述
H'01	Interface Speed	Get	DWORD	0	通讯速度 (indeterminate)
H'02	Interface Flags	Get	DWORD	H'0D	Ethernet 通讯口状态※1
H'03	Physical Address	Get	ARRAY of 6 USINTs	By Product	MAC 地址
H'0B	Interface Capability	Get	STRUCT of :	--	Ethernet 通讯口支持能力※2
	Capability Bits		DWORD	H'02000000	Ethernet 通讯口能力定义
	Speed/Duplex Options		STRUCT of :	--	Ethernet 通讯口速度和双工能力支持定义
	Speed/Duplex Array Count		USINT	H'00	支持速度和双工组合的数量
	Speed/Duplex Array		ARRAY of STRUCT of :	--	支持速度和双工组合的内容
	Interface Speed		UINT	H'00	Ethernet 通讯口速度 · 例如 : 10 bps 为 H'0A · 100 bps 为 H'64
	Interface Duplex Mode		USINT	H'00	Ethernet 通讯口双工能力 · 例如 : 半双工为 H'00 · 全双工为 H'01

※1 Interface Flag Table

Bit (s)	名称	描述
0	Link Status	0 indicates an inactive link 1 indicates an active link
1	Half/Full Duplex	0 indicates half duplex 1 indicates full duplex
2-4	Negotiation Status	0 : Auto-negotiation in progress 1 : Auto-negotiation and speed detection failed 2 : Auto negotiation failed but detected speed 3 : Successfully negotiated speed and duplex 4 : Auto-negotiation not attempted. Forced speed and duplex.
5	Manual Setting Requires Reset	shall be set zero
6	Local Hardware Fault	0 indicates the interface detects no local hardware fault 1 indicates a local hardware fault is detected
7-31	Reserved	0

※2 Interface Capability Bits

Bit (s)	名称	描述
0	Manual Setting Requires Reset	Indicates whether or not the device requires a reset when instance attribute #6 (Interface Control attribute) changes. 0 indicates the device does not require a reset 1 indicates the device requires a rest
1	Auto-negotiate	0 indicates the interface does not support auto-negotiaton 1 indicates the interface supports auto-negotiation
2	Auto-MDIX	0 indicates the interface does not support auto MDIX operation 1 indicates the interface supports auto MDIX operation
3	Manual Speed/Duplex	0 indicates the interface does not support to set speed/duplex. (Instance attribute #6, Interface Control attribute) 1 indicates the interface supports to set speed/duplex
4-31	Reserved	shall be set 0

9.4.5 台达 EIP 产品一览表

9.4.5.1 台达 EIP 产品列表支持 Adapter 功能

分类	产品	韧体版本
中型 PLC	AHCPU501-EN、AHCPU511-EN、AHCPU521-EN、 AHCPU531-EN	V2.00
	AHCPU560-EN2	V1.00
	AH10EN-5A	V2.00
	AHRTU-ETHN-5A	V1.00
	AH10EMC-5A	V1.00
	AS300 系列	V1.00
	AS200 系列	V1.00
	AS300 系列 (AS-FEN02 通讯卡)	V1.06 (V1.00)
	AS00SCM-A (AS-FEN02 通讯卡)	V2.02 (V1.00)
	AX-3 系列	V1.01
小型 PLC	DVPES2-E 系列	V3.60
	DVP26SE	V1.00
	DVP-ES3 系列	V1.00
变频器	CMM-EIP01/02 通讯卡 (搭配 VFD-M300 系列)	V1.00
	CMC-EIP01 通讯卡 (搭配 VFD-C2000 系列)	V1.06
	CMM-EIP03 通讯卡 (搭配 VFD-M300 系列)	V1.00
	CMC-EIP02 通讯卡 (搭配 VFD-C2000 系列)	V1.00

9.4.5.2 台达 EIP 产品支持 Scanner 功能

分类	产品	版本
中型 PLC	AHCPU501-EN、AHCPU511-EN、AHCPU521-EN、 AHCPU531-EN	V2.00
	AHCPU560-EN2	V1.00
	AH10EN-5A	V2.00
	AS300 系列、AS200 系列	V1.00
	AX-3 系列	V1.01
小型 PLC	DVP-ES3 系列	V1.00

9.5 网络安全

本产品之以太网网络通讯，建议用户应用于封闭式之网络通讯，或具有防火墙之局域网络，以防止有心人士进行不正当之网络攻击，进而影响产品正常运行。

9.6 FTP 功能

在许多的使用情境下，控制器与控制器/计算机之间需要进行文件传递，因此 AX-308E 系列控制器更进一步提供 FTP 功能来满足使用上的需求，用户可以自行设定是否要开启 FTP server 或是藉由 FTP client 相关功能块来访问 FTP server，进行文件的上传与下载。

9.6.1 FTP Server

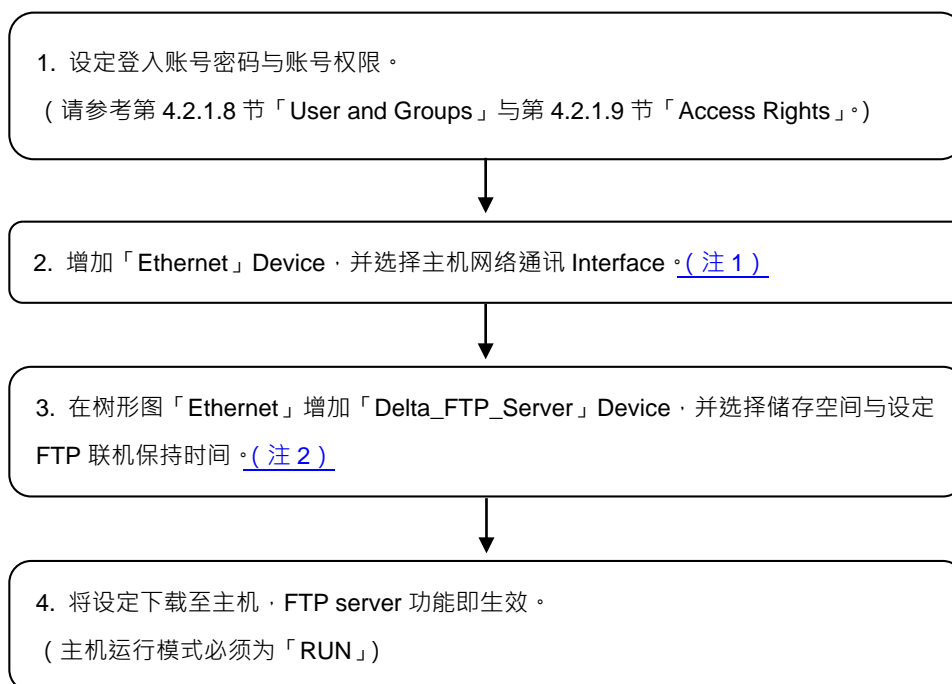
9.6.1.1 FTP Server 规格

规格	说明
支持的 FTP 命令	USER : FTP server 登入账号。 PASS : FTP server 账号密码。 PWD : 显示当前文件夹路径。 PORT : FTP 主动模式。 PASV : FTP 被动模式。 DELE : 删除文件。 CWD : 变更当前文件夹路径。改变工作目录。 RMD : 删除文件夹。 CDUP : 退至上一层文件夹。 MKD : 建立文件夹。 LIST : 显示当前文件夹文件内容。 STOR : 上载文件至 FTP server。 APPE : 将数据增加当前文件的最后。 RETR : 从 FTP server 下载文件。 NOOP : 无操作。 QUIT : 关闭 FTP server 联机。 RNFR : 欲更名的文件...。 RNT0 : 将文件更名为...。
账号密码权限管理	可以设定登入账号的动作权限。(请参考第 4.2.1.9 节 Access Rights) 账号与密码可设定长度最大为 63 个 bytes。

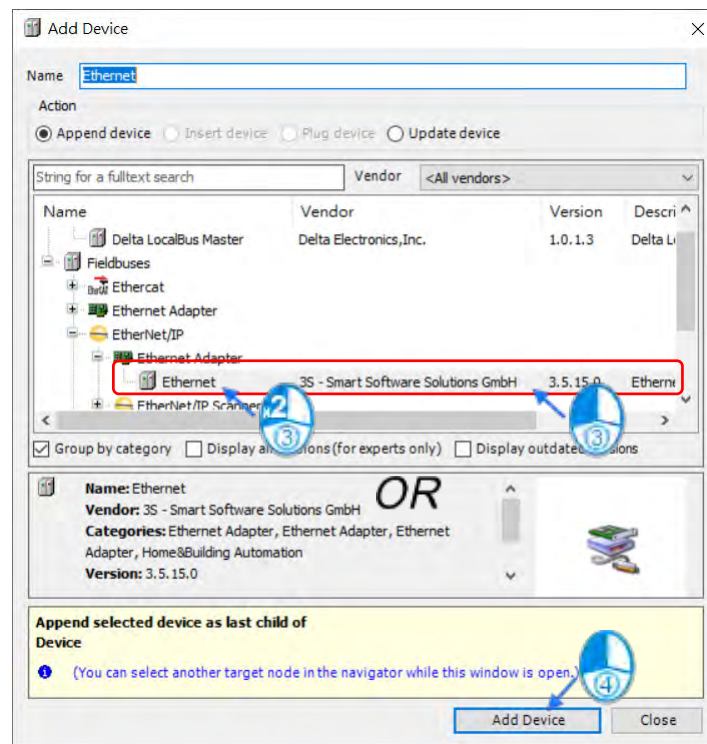
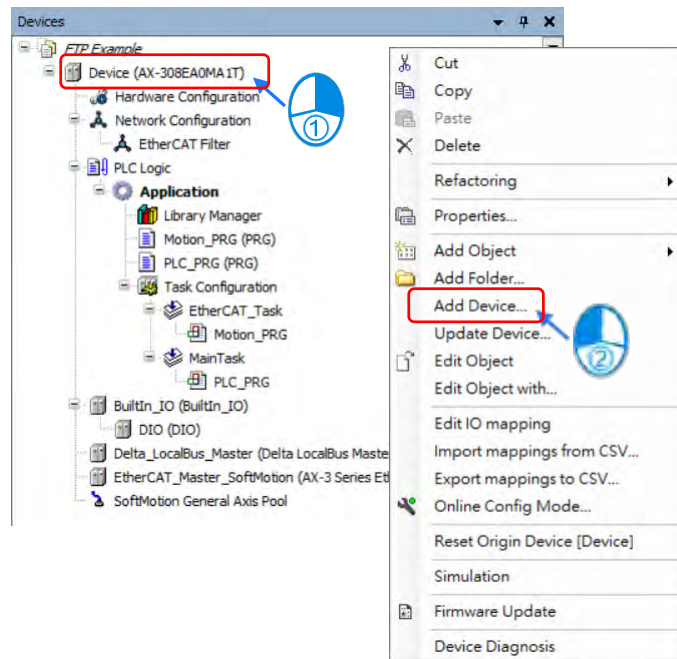
规格	说明
权限设定	Add/Remove : 上载文件与移除文件权限 Modify : 覆盖或编辑文件权限 View : 下载文件权限 Execute : 变更文件夹路径权限
可同时登入数量	1
储存空间	内部存储器 (Internal Memory) 外部 SD 卡 (External SD)
联机保持时间	1 ~ 65535 秒

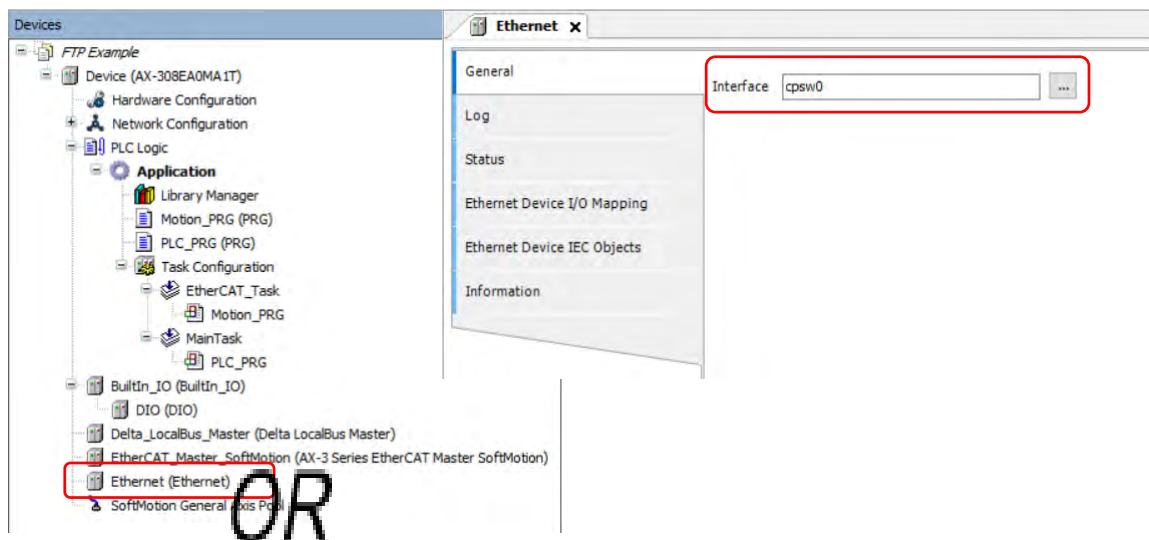
9.6.1.2 FTP Server 功能启动流程

AX-308E 系列控制器在建立账号密码与账号权限之后，仅需将「Delta_FTP_Server」Device 加入项目树形图并下载，待 PLC 运行状态为「RUN」模式时，其 FTP server 即开始运行。

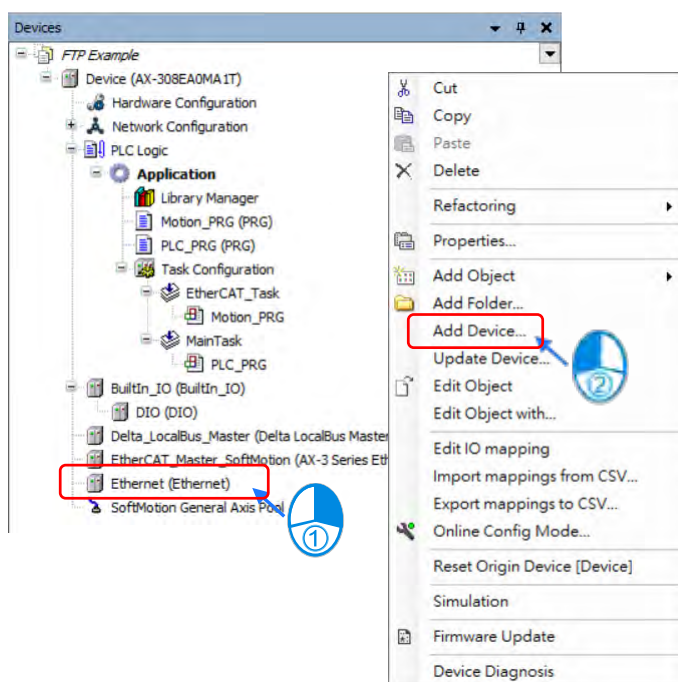


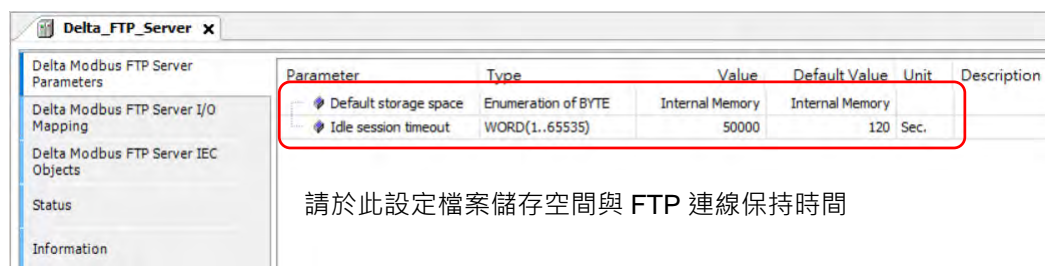
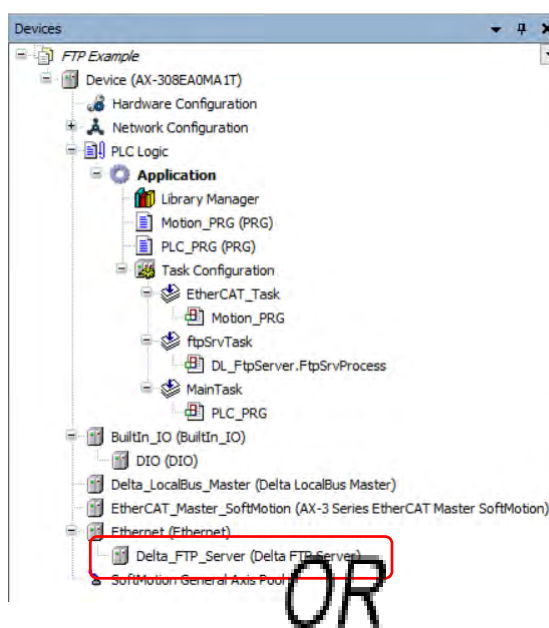
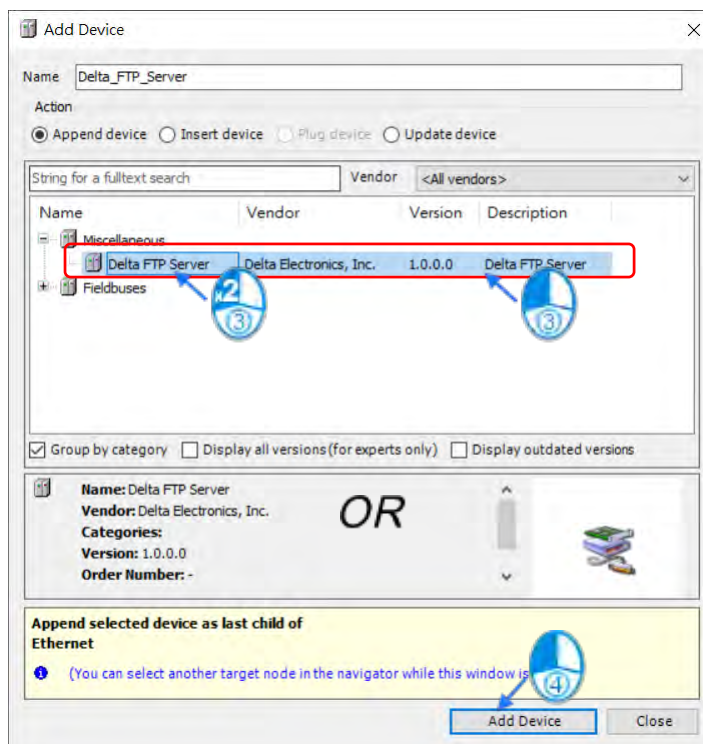
- 注 1：增加「Ethernet」Device，并选择主机网络通讯 Interface。





- 注 2 : 在树形图「Ethernet」增加「Delta_FTP_Server」Device，并选择储存空间与设定 FTP 联机保持时间。





請於此設定檔案儲存空間與 FTP 連線保持時間

9.6.1.3 FTP Client 联机 (AX308E 系列控制器为 FTP Server)

用户可以通过 windows 文件总管或是免费的 FTP client 软件 (如「FileZilla」) , 即可通过 FTP 功能来访问 AX308E 系列控制器, 仅需输入 AX308E 系列控制器的 IP 地址与登录的账号密码, 即可进行文件的上传与下载。

9.6.2 FTP Client

在 AX308E 系列控制器, 用户可以通过 Library「DL_FtpClient」内的「DFB_FTPClient」功能块来访问 FTP server, 进行文件的上传与下载, 于本章节仅简易说明, 其详细说明请参考「AX 系列标准指令手册」。

● DFB_FTPClient

功能块 DFB_FTPClient	说明
支持的命令类型	NOOP : 无操作。 PUT_FILE : 上传文件。 GET_FILE : 下载文件。 DELETE : 删除文件。 APPEND : 将资料新增至源文件结尾处。
储存位置	外部储存卡 (EXTERNAL_SD) 内部存储器 (INTERNAL_MEM)



第A章 故障排除

目录

A.1	故障排除	A-3
A.1.1	基本检查	A-3
A.1.2	清除错误状态	A-3
A.1.3	故障排除流程	A-4
A.1.4	查询错误记录	A-4
A.2	CPU 模块故障排除	A-7
A.2.1	ERROR 灯号一般闪烁 (亮 0.5 秒、暗 0.5 秒)	A-7
A.2.2	ERROR 灯号快速闪烁 (亮 0.2 秒、暗 0.2 秒)	A-8
A.2.3	ERROR 灯号慢速闪烁 (亮 1 秒、暗 3 秒)	A-8
A.2.4	BAT. LOW 灯号常亮	A-8
A.2.5	BAT. LOW 灯号一般闪烁 (亮 0.5 秒、暗 0.5 秒)	A-8
A.2.6	其它	A-9
A.3	函式库故障排除	A-10
A.3.1	DL_BuiltInIO_AX3	A-10
A.3.2	运动指令相关	A-13
A.4	模块故障排除	A-14
A.4.1	模拟 (AD/DA/XA) 及温度 (RTD/TC) 模块故障排除	A-14
A.4.1.1	ERROR 灯常亮	A-14
A.4.1.2	ERROR 灯闪烁 (亮 0.2 秒、暗 0.2 秒)	A-14
A.4.2	AS02LC 称重模块故障排除	A-15
A.4.2.1	ERROR 灯常亮	A-15
A.4.2.2	ERROR 灯闪烁 (亮 0.2 秒、暗 0.2 秒)	A-15
A.4.3	AS02HC 高速计数模块故障排除	A-15
A.4.3.1	ERROR 灯常亮	A-15
A.4.3.2	ERROR 灯闪烁 (亮 0.5 秒、暗 0.5 秒)	A-16
A.4.4	AS02/04PU 定位模块故障排除	A-17
A.4.4.1	ERROR 灯闪烁 (亮 0.2 秒、暗 0.2 秒)	A-17

A.5	CPU 模块错误代码对应灯号及状态说明	A-18
A.5.1	CPU 模块错误代码对应灯号及状态	A-18
A.5.2	模拟与温度模块错误代码对应灯号	A-19
A.5.3	AS02LC 称重模块错误代码对应灯号	A-20
A.5.4	AS02HC 高速计数模块错误代码对应灯号	A-21
A.5.5	AS02/04PU 定位模块错误代码对应灯号	A-21

A.1 故障排除

A.1.1 基本检查

本章说明系统运行时可能发生各种类型的故障，以及这些故障的原因和解决方法。排除故障须先确认故障的原因。在分析故障原因之前请先确认以下三点：

(1) 检查以下项目

- PLC 系统必须处于规范条件下运作（包括环境、电气、机械震动...等条件）
- 设备电源妥善接通，输入至 PLC 系统的电源有正常开启。
- 各模块、端子与电缆之间有妥善安装
- 各个显示灯号为正常状态
- 各种设定开关位于正确位置

(2) 按照下述操作 AX-3 系列控制器，以检查运行情况。

- 切换 RUN/STOP 开关
- 确认主机所设定的 RUN/STOP 条件
- 确认并排除外部装置可能造成的影响
- 以 DIADesigner-AX 软件确认系统的运作状况及错误记录

(3) 依照以上（1）（2）两点判断故障发生原因的范围

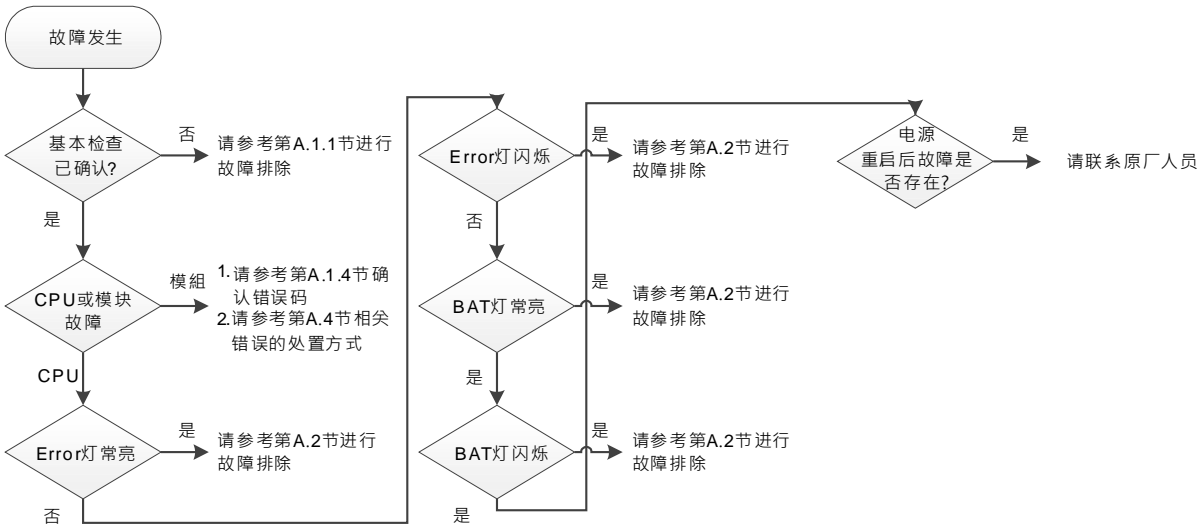
- AX-3 系列控制器或者外部装置
- 主机或扩展模块
- 参数设定或控制程序

A.1.2 清除错误状态

当系统处于错误状态时，可尝试以下述几种方式清除系统的错误状态，但在清除之后，若错误来源仍然存在，则系统还是会再次进入错误状态。

- (1) 将主机运行状态切换至 STOP 后再切换至 RUN。
- (2) 将主机断电后重开。
- (3) 以 DIADesigner-AX 执行 Reset Warn。
- (4) 通过 DIADesigner-AX 执行 Reset Origin 后，重新下载项目并执行。

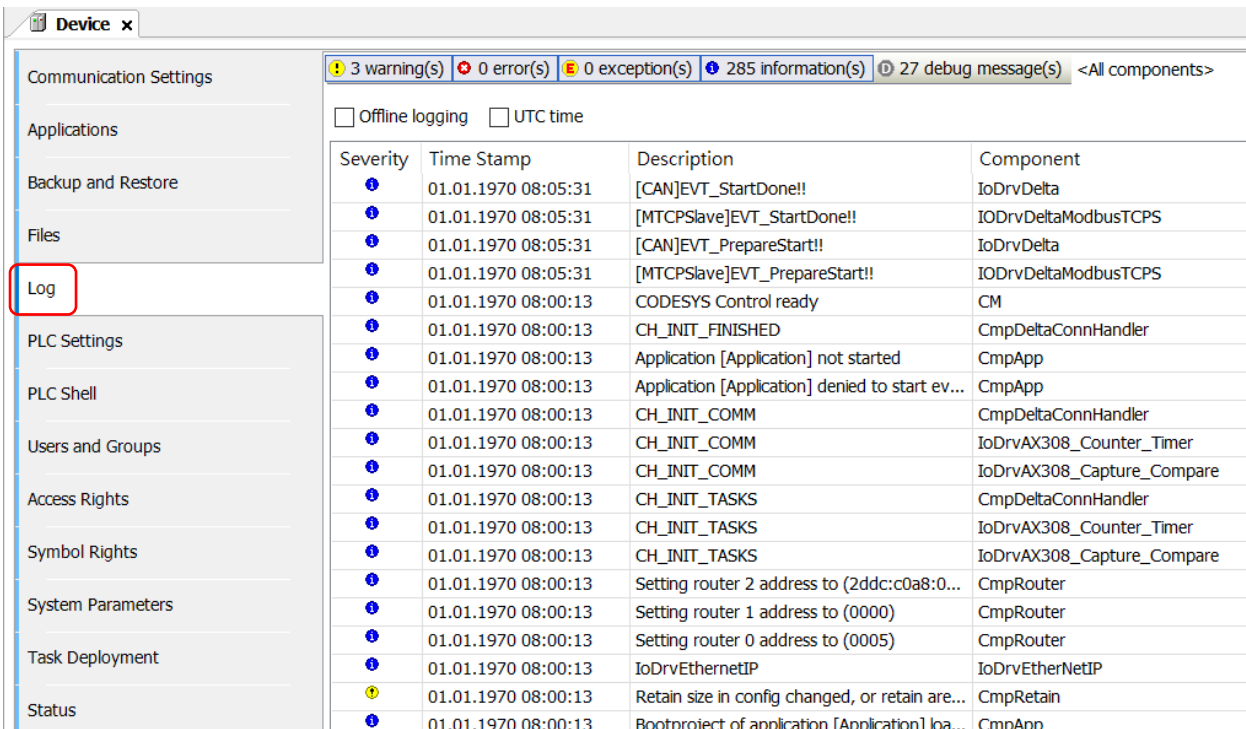
A.1.3 故障排除流程



A.1.4 查询错误记录

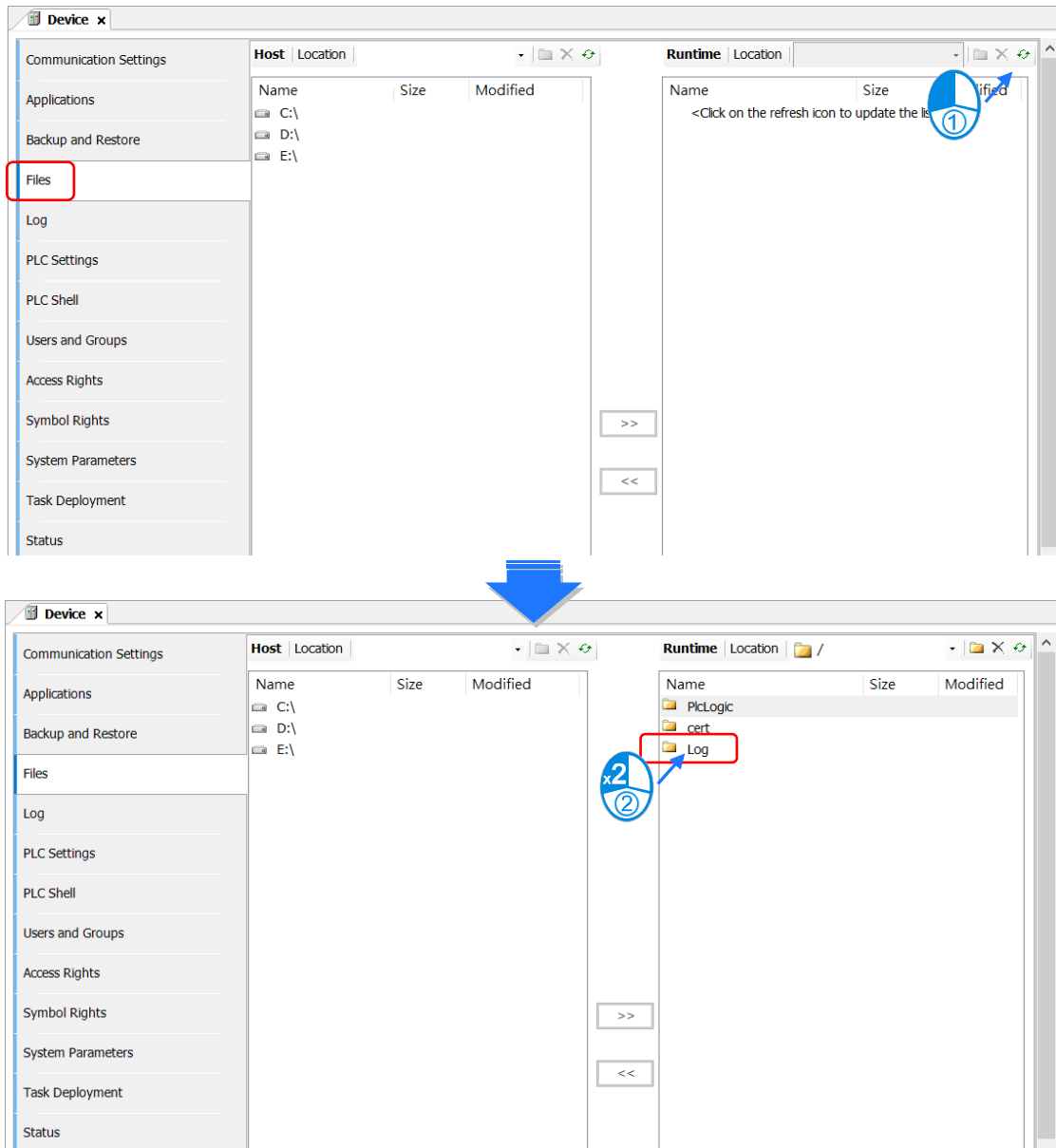
错误发生时，系统将产生相对应的错误代码并记录于控制器中，用户可以 Device 内的 Log 页面确认，在 Log 页面会列出所有控制器事件，其包含启动与停止过程中所产生的事件、application 与 boot application 加载事件、自定义事件、I/O 驱动事件记录与 data source 事件记录，详细说明请参考第 4.2.1.5 节 LOG。

- LOG 页面 (路径地址 : Device > LOG) :



2. 错误记录文件：

用户可以通过软件 DIADesigner-AX 读取控制器内 LOG 记录文件，来确认控制器所曾经发生过的所有异常，当控制器关闭电源或 LOG 记录超过 64KB 时，即于 LOG 文件夹产生 LOG 记录文件（.CSV）。





The screenshot displays a software window titled 'Device x' with a left-hand navigation pane and two main content panes. The navigation pane includes categories like Communication Settings, Applications, Backup and Restore, Files, Log, PLC Settings, PLC Shell, Users and Groups, Access Rights, Symbol Rights, System Parameters, Task Deployment, and Status. The 'Host' pane shows a file list with columns for Name, Size, and Modified, listing drive letters C:\, D:\, and E:\. The 'Runtime' pane shows a file list with columns for Name, Size, and Modified, listing multiple files named 'StdLogger52...' with a size of 64.50 K... and a modified date of 1980/1/... There are '>>' and '<<' buttons between the two panes.

A.2 CPU 模块故障排除

请依据 CPU 模块上的 LED 指示灯号状态及错误代码，从以下表格中获知故障排除方式。

A.2.1 ERROR 灯号一般闪烁（亮 0.5 秒、暗 0.5 秒）

CPU 相关错误

错误代码 (16#)	错误说明	排除方式
140E	远程模块右侧连接超出 8 台限制	确认远程模块右侧连接台数
1600	主机扩展模块 ID 配置超出范围	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。
1601	主机扩展模块无法配置 ID	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。
1602	主机扩展模块 ID 配置重复	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。
1603	主机扩展模块无法进入运行模式	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。
1604	主机扩展模块通讯超时	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。
2000	主机内部存储器检测错误	请联络原厂
2001	主机外部内存检测错误	请联络原厂
2100	MODBUS TCP 联机数超出范围	检查 MODBUS TCP (Server+Client) 联机数量是否超过 32 条
2200	实际 IO 模块不符合配置设定	确认 Hardware Configuration 中的模块配置设定与实际的模块配置是否相符
2201	通讯模块超出 4 台限制	确认通讯模块台数
2202	定位模块超出 8 台限制	确认定位模块台数
2203	所有扩展模块超出 32 台限制	确认扩展模块台数

EtherCAT 相关错误

错误代码 (16#)	错误说明	排除方式
1	EtherCAT 通讯遗失	确认线材或端子是否有松脱或接触不良，网络线插回后重置 EtherCAT Master
2	EtherCAT 数据交换遗失	确认线材或端子是否有松脱或接触不良，网络线插回后重置 EtherCAT Master
4	EtherCAT 网卡名称错误	确认 EtherCAT Master 一般页面中设定的网卡名称或网卡地址

错误代码 (16#)	错误说明	排除方式
5	EtherCAT 从站初始化失败	确认 Network Configuration 与实际配置是否相同
6	从站 Vendor ID 不匹配	<ul style="list-style-type: none"> ● 确认 Network Configuration 与实际配置是否相同 ● 确认从站 ESI 是否匹配 ● 取消 EtherCAT Master 一般页面中 Vendor ID 检查
7	从站 Product ID 不匹配	<ul style="list-style-type: none"> ● 确认 Network Configuration 与实际配置是否相同 ● 确认从站 ESI 是否匹配 ● 取消 EtherCAT Master 一般页面中 Product ID 检查

注：EtherCAT 相关错误是由 IODrvEtherCAT 函式库定义

A.2.2 ERROR 灯号快速闪烁 (亮 0.2 秒、暗 0.2 秒)

表示 CPU 模块的输入电源 24VDC 正在断电进行中，或者电源供应不足或不稳定，造成无法运行。

错误代码 (16#)	错误说明	排除方式
2004	曾经发生过 24V 低电压又恢复	确认电源系统是否稳定。

A.2.3 ERROR 灯号慢速闪烁 (亮 1 秒、暗 3 秒)

错误代码 (16#)	错误说明	排除方式
1800 ~ 180F	扩展模块发出警告信息	参考第 A.4 节模块警告信息

A.2.4 BAT. LOW 灯号常亮

此灯号表示未安装电池 (CR1620) 或电池低电压之警告，若用户不须 CPU 断电后持续让万年历计时，则可通过 System Parameters 页面 (Device → System Parameter) 将 "Show Battery Low Voltage Error" 设定成 Disable。(默认为 Enable)

错误代码 (16#)	错误说明	排除方式
2003	电池低电压	更换电池或由参数设定关闭灯号显示

A.2.5 BAT. LOW 灯号一般闪烁 (亮 0.5 秒、暗 0.5 秒)

此闪烁灯号表示万年历计时功能异常，须回厂进行维修。

错误代码 (16#)	错误说明	排除方式
2002	万年历无法计时	请联络原厂

A.2.6 其它

错误代码 (16#)	错误说明	排除方式
2500	装置描述版本 (DDF) 与控制器韧体不兼容	请确认装置描述文件 (DDF) 与控制器韧体是否兼容
2501	SSI 编码器未连接到 PLC	请检查 SSI 编码器与 PLC 的之间线路
2502	SSI 编码器设定单圈+多圈超出设定范围 (最大为 32 bits)。	单圈+多圈的参数设定值相加须小于等于 32
2503	脉冲输出轴发生错误	请于对应的脉冲输出轴，确认在线监控页面的错误信息。

A.3 函式库故障排除

A.3.1 DL_BuiltInIO_AX3

当使用函式库「DL_BuiltInIO_AX3」发生错误时，并不会影响 PLC 主机运行状态，也不会有任何的 Error 灯号显示。

错误代码 (16#)	名称	错误说明	排除方式
0	DFB_HSIO_NO_ERR	DFB 高速 IO 功能块没有错误	无
186A0	DMC_HP_INVALID_HOME_SPEED	脉冲轴设定的原点回归速度值设定无效	请在 Pulse Axis 装置设定页面的原点回归速度 "Search for switch" 或 "Search for Z phase pulse" 设定非零数值
186A1	DMC_HP_INVALID_HOME_ACC_DEC	脉冲轴设定的原点回归加速度或减速度值设定无效	请在 Pulse Axis 装置设定页面的原点回归加速度或减速度设定非零数值
186A2	DMC_HP_INVALID_HOME_POSITION	原点回归的位置值设定无效	将功能块引脚 "IrPosotion" 设定在脉冲轴的旋转范围内 [0 ~ PulseAxis.Modulo Value]
186A3	DMC_HP_AXIS_NOT_PULSEAXIS	设定功能块输入引脚类型的变量的类型不是 PulseAxis_REF	确认勾选 IO Configuration 上的 Pulse Axis 后，IEC Object 的变量输入至功能块 DMC_Home_P 输入引脚 "Axis"
186A4	DMC_HP_HOMING_METHOD_RESERVED	此版本不支持此原点回归模式	确认使用的原点回归模式设定是否在此版本有提供，请参考规格文件并修改模式
186A5	DMC_HP_HOMING_MOVEMENT_HW_LIMIT	正极限信号或负极限信号处于激活状态，并且轴在此回归模式下无法继续动作	确认使用的原点回归模式支持使用的硬件极限信号，请参考规格文件并修改模式或硬件极限信号配置
186A6	DMC_HP_HOMING_AXIS_STATE_NOT_STANDSTILL	脉冲轴状态机不是处于 Standstill	确认 DMC_Home_P 功能块是在轴状态机为 Standstill 时启动
186AC	DFB_CAP_INVALID_CAPTURE_REF	设定功能块输入引脚类型的变量的类型不是 Capture_REF	确认勾选 IO Configuration 上的 Capture 后，将 IEC Object 的变量输入至功能 DFB_Capture 输入引脚 "Capture"
186AD	DFB_CAP_INVALID_COUNTER_REF	设定功能块输入引脚类型的变量的类型不是 Counter_REF	确认勾选 IO Configuration 上的 Counter 后，将 IEC Object 的变量输入至功能块 DFB_Capture 输入引脚 "Counter"
186AE	DFB_CAP_INVALID_VALUE_SETTING	DFB_Capture 的屏蔽设定值 (uiMaskValue) 超过旋转轴范围	将功能块引脚 "uiMaskValue" 设定在编码器轴的旋转范围内 [0 ~ EncoderAxis.Modulo Value]

错误代码 (16#)	名称	错误说明	排除方式
186AF	DFB_CAP_INVALID_D ELTARANGE	输入引脚 "diDeltaMax" 或 "diDeltaMin" 超过旋 转轴范围当高速计数器的 编码器轴是旋转轴	将功能块引脚 "diDeltaMax" 或 "diDeltaMin" 设定在编码器轴的旋转范 围内 [0 ~ EncoderAxis.Modulo Value]
186B0	DFB_CAP_CAPTURE _ALREADY_ENABLE	高速捕捉装置已经被启用	确认此 Capture 装置是否被其它的 DFB_Capture 启用
186B6	DFB_CMP_INVALID_ COMPARE_REF	设定功能块输入引脚类型 的变量的类型不是 Compare_REF	确认勾选 IO Configuration 上的 Compare 后·将 IEC Object 的变量输入至功能块 DFB_Compare 输入引脚 "Compare"
186B7	DFB_CMP_INVALID_ COUNTER_REF	设定功能块输入引脚类型 的变量的类型不是 Counter_REF	确认勾选 IO Configuration 上的 Counter 后·将 IEC Object 的变量输入至功能块 DFB_Compare 输入引脚 "Counter"
186B8	DFB_CMP_INVALID_ CMPVALUE	输入引脚 "diCompareValue" 超过 旋转轴范围当高速计数器的 编码器轴是旋转轴	将功能块引脚 "diCompareValue" 设定 在编码器轴的旋转范围内 [0 ~ EncoderAxis.Modulo Value]
186B9	DFB_CMP_INVALID_ REFRESHCYCLE	输入引脚 "wRefreshCycle" 超过范 围 [0-30000]·单位 0.1us	将功能块引脚 "wRefreshCycle" 设定在 范围 [0 ~ 30000]内
186BA	DFB_CMP_ COMPARE_ALREADY _ENABLE	高速比较装置已经被启用	确认此 Compare 装置是否被其它的 DFB_Compare 启用
186C0	DFB_HC_INVALID_ COUNTER_REF	设定功能块输入引脚类型 的变量的类型不是 Counter_REF	确认勾 IO Configuration 上的 Counter 后·将 IEC Object 的变量输入至功能块 DFB_HCnt 输入引脚 "Counter"
186C1	DFB_HC_COUNTER_ ALREADY_ENABLE	高速计数装置已经被启用	确认此 Counter 装置是否被其它的 DFB_HCnt 启用
186C2	DFB_HC_COUNTER_ REF_CHANGED_ DURING_ OPERATION	输入引脚 "Counter" 在功 能块运行时被修改替换	确认是否在 DFB_HCnt 功能块启动后有 改变输入引脚 Counter 的变量
186C8	DFB_HT_INVALID_ TIMER_REF	设定功能块输入引脚类型 的变量的类型不是 Timer_REF	确认勾选 IO Configuration 上的 Timer 后·将 IEC Object 的变量输入至功能块 DFB_HTmr 输入引脚 "Timer"
186C9	DFB_HT_TIMER_ ALREADY_ENABLE	高速计时装置已经被启用	确认此 Timer 装置是否被其它的 DFB_HTmr 启用
186CA	DFB_HT_TIMER_REF _CHANGED_DURING _OPERATION	输入引脚 "Timer" 在功 能块运行时被修改替换	确认是否在 DFB_HTmr 功能块启动后有 改变输入引脚 Timer 的变量

错误代码 (16#)	名称	错误说明	排除方式
186D0	DFB_PV_INVALID_COUNTER_REF	设定功能块输入引脚类型的变量的类型不是 Counter_REF	确认勾选 IO Configuration 上的 Counter 后，将 IEC Object 的变量输入至功能块 DFB_PresetValue 输入引脚 "Counter"
186D1	DFB_PV_NOT_ENABLE_EXTERNAL_TRIGGER	计数器装置设定未勾选使用外部触发信号但 DFB_PresetValue 模式设定 "EXTERNAL_TRIGGER"	请在 Counter 装置设定页面勾选 External Trigger
186D2	DFB_PV_PREVIOUS_PRESET_NOT_DONE	计数器的预置计数功能已被其它 DMC_PresetValue 功能块启用且尚未完成	请等 Counter 装置正在使用的 DFB_PresetValue 功能块作用完成后再执行此功能块
186D3	DFB_PV_CANNOT_PRESET_WHEN_SAMPLING	计数器正在执行 DFB_Sample 功能	关闭此 Counter 装置的 Sample 功能，请将用此 Counter 的 DFB_Sample 关闭
186D4	DFB_PV_SETRING_NOT_DONE	计数器的正在执行 DFB_SetRing 功能且尚未完成	请等 Counter 装置正在使用的 DFB_SetRing 功能块作用完成后再执行此功能块
186D5	DFB_PV_INVALID_PRESET_VALUE	输入引脚 "diPresetValue" 超过旋转轴范围当高速计数器的编码器轴是旋转轴	将功能块引脚 "diPresetValue" 设定在旋转轴范围 [0 ~ EncoderAxis.Modulo Value]
186D6	DFB_PV_COUNTER_REF_CHANGED_DURING_OPERATION	输入引脚 "Counter" 在功能块运行时被修改替换	确认是否在 DFB_PresetValue 功能块启动后有改变输入引脚 Counter 的变量
186DC	DFB_SP_INVALID_COUNTER_REF	设定功能块输入引脚类型的变量的类型不是 Counter_REF	确认勾选 IO Configuration 上的 Counter 后，将 IEC Object 的变量输入至功能块 DFB_Sample 输入引脚 "Counter"
186DD	DFB_SP_COUNTER_NOT_ENABLE	DFB_Counter 尚未完成高速计数器允许	确认 Counter 装置已经被 DFB_HCnt 启动后再使用此 DFB_Sample 功能块
186DE	DFB_SP_ALREADY_SAMPLING	计数器正在执行 DFB_Sample 功能	确认此 Counter 装置是否被其它的 DFB_Sample 启用
186DF	DFB_SP_PRESET_NOT_DONE	计数器的正在执行 DFB_PresetValue 功能且尚未完成	请等 Counter 装置正在使用的 DFB_PresetValue 功能块作用完成后再执行此功能块
186E0	DFB_SP_INVALID_SAMPLE_TIME	DFB_Sample 的输入引脚 "wSampleTime" 超过范围 [10-65535]	将功能块引脚 "wSampleTime" 设定在范围 [10 ~ 65535] 内

错误代码 (16#)	名称	错误说明	排除方式
186E1	DFB_SP_COUNTER_REF_CHANGED_DURING_OPERATION	输入引脚 "Counter" 在功能块运行时被修改替换	确认是否在 DFB_Sample 功能块启动后有改变输入引脚 Counter 的变量
186E7	DFB_SR_INVALID_COUNTER_REF	设定功能块输入引脚类型的变量的类型不是 Counter_REF	确认勾选 IO Configuration 上的 Counter 后，将 IEC Object 的变量输入至功能块 DFB_SetRing 输入引脚 "Counter"
186E8	DFB_SR_COUNTER_HAS_NO_CHILD_ENCODER_AXIS	高速计数装置没有子节点与编码器轴连接	对 Counter 装置插入 EncoderAxis 且设定为旋转轴后重新启动功能块
186E9	DFB_SR_COUNTER_NOT_RING	高速计数装置下的编码器轴不是旋转轴类型	在 Counter 装置设定页面将编码器轴类型选择旋转轴
186EA	DFB_SR_PREVIOUS_SETRING_NOT_DONE	计数器的预置计数功能已被其它 DFB_SetRing 功能块启用且尚未完成	请等 Counter 装置正在使用的 DFB_SetRing 功能块作用完成后再执行此功能块
186EB	DFB_SR_PRESET_NOT_DONE	计数器的正在执行 DFB_PresetValue 功能且尚未完成	请等 Counter 装置正在使用的 DFB_PresetValue 功能块作用完成后再执行此功能块
186EC	DFB_SR_INVALID_RING_RANGE	当高速计数器的编码器轴是旋转轴时，输入引脚 "diPositionPeriod" 小于零或是现在计数值大于此设定值 bSetDown = FALSE	将功能块引脚 "diPositionPeriod" 设定在大于零且小于现在的计数值之间
186ED	DFB_SR_COUNTER_REF_CHANGED_DURING_OPERATION	输入引脚 "Counter" 在功能块运行时被修改替换	确认是否在 DFB_SetRing 功能块启动后有改变输入引脚 Counter 的变量

A.3.2 运动指令相关

当使用函式库「DL_MotionControl」或「DL_MotionControlLight」发生错误时，并不会影响 PLC 主机运行状态，也不会有任何的 Error 灯号显示。有关详细的故障排除内容，请参考 **AX 系列运动控制指令手册附录**。

A.4 模块故障排除

•模块简介与错误代码类型说明

AX-3 系列控制器可以安装 AS 系统的下列模块，详细规格请参阅各模块使用手册。模块错误代码主要分为两大类，一类为错误型之代码，当发生此类错误时，主机会跟随模块进行停止运行之动作；另一类为警告型之代码，当发生此类警告信息时，主机将持续运行。详细代码与故障排除方式表列如下：

A.4.1 模拟 (AD/DA/XA) 及温度 (RTD/TC) 模块故障排除

A.4.1.1 ERROR 灯常亮

下表错误代码须由模块的 Alarm Setting 设定为 “True”，则模块才会以错误信息回报，否则将以一般警告信息回报。

错误代码	错误说明	排除方式
16#1605	模块内部硬件错误	退回原厂检修
16#1607	模块外部电源错误	检查电源
16#1608	校正值或 CJC 补偿异常	请联络原厂

A.4.1.2 ERROR 灯闪烁 (亮 0.2 秒、暗 0.2 秒)

为有利于 AX-3 系列控制器运行 (RUN) 时不受模块的错误信息影响整个系统，因此模块默认下表这些错误代码为警告类信息，若用户期望模块发生下表前 4 项错误时，PLC 须立即停止 (STOP)，则请由模块的硬件组态内报警设定勾选为 “报警” 信息 (默认值为 “警告”)。

错误代码	错误说明	排除方式
16#1801	模块外部电源错误	检查电源
16#1802	模块硬件错误	退回原厂检修
16#1804	校正值异常	请联络原厂
16#1807	CJC 温度错误	退回原厂检修
16#1808	模拟 (温度) 输入通道 1 超出范围值	检查通道 1 输入信号
16#1809	模拟 (温度) 输入通道 2 超出范围值	检查通道 2 输入信号
16#180A	模拟 (温度) 输入通道 3 超出范围值	检查通道 3 输入信号
16#180B	模拟 (温度) 输入通道 4 超出范围值	检查通道 4 输入信号
16#180C	模拟 (温度) 输入通道 5 超出范围值	检查通道 5 输入信号
16#180D	模拟 (温度) 输入通道 6 超出范围值	检查通道 6 输入信号
16#180E	模拟 (温度) 输入通道 7 超出范围值	检查通道 7 输入信号
16#180F	模拟 (温度) 输入通道 8 超出范围值	检查通道 8 输入信号
-	上电时未收到主机检测命令	检查主机与模块连接是否确实或重新组合

A.4.2 AS02LC 称重模块故障排除

A.4.2.1 ERROR 灯常亮

下表错误代码须由模块的 Alarm Setting 设定为“True” 错误信息显示，则模块才会以错误信息回报，否则将以一般警告信息回报。

错误代码	错误说明	排除方式
16#1605	模块内部硬件（含驱动板）错误	退回原厂检修
16#1607	模块外部电源错误	检查电源

A.4.2.2 ERROR 灯闪烁（亮 0.2 秒、暗 0.2 秒）

为有利于 PLC 运行（RUN）时不受模块的错误信息影响整个系统，因此模块默认下表这些错误代码为警告类信息，若用户期望模块发生前 3 项错误时，PLC 立即停止（STOP），则请由模块的硬件组态内报警设定勾选为“报警”信息（默认为“警告”）。

错误代码	错误说明	排除方式
16#1801	模块外部电源错误	检查电源
16#1802	模块硬件错误	退回原厂检修
16#1807	模块驱动板错误	检查端子线是否受到干扰或短路（如 EXC+与 EXC-短路），如排除上述问题之后，还是持续出现此错误，请退回原厂检修。
16#1808	通道 1 输入超出量测范围或 SEN 电压错误	检查通道 1 输入信号与接线
16#1809	通道 1 超出重量上限	检查通道 1 输入重量值与最大重量值设定
16#180A	通道 1 调校错误	检查通道 1 调校重量值与调校步骤
16#180B	通道 2 输入超出量测范围或 SEN 电压错误	检查通道 2 输入信号与接线
16#180C	通道 2 超出重量上限	检查通道 2 输入重量值与最大重量值设定
16#180D	通道 2 调校错误	检查通道 2 调校重量值与调校步骤
-	上电时未收到主机检测命令	检查主机与模块连接是否确实或重新组合

A.4.3 AS02HC 高速计数模块故障排除

A.4.3.1 ERROR 灯常亮

错误代码	错误说明	排除方式
16#1605	计数值停电保持异常（严重错误）	计数值记录已遗失，请将模块断电重启（重启后，错误代码将自行清除）。 若一再出现此信息，请联络原厂。

错误代码	错误说明	排除方式
16#1606	模块设定值停电保持异常 (严重错误)	模块设定已遗失并恢复出厂设定, 模块断电重启或重新下载此模块 Parameters 可清除错误代码。 若一再出现此信息, 请联络原厂。
16#1607	模块设定值配置内容错误 (严重错误)	检查此模块 Parameters 配置内容并重新下载。 若一再出现此信息, 请联络原厂。

A.4.3.2 ERROR 灯闪烁 (亮 0.5 秒、暗 0.5 秒)

为有利于 PLC 运行 (RUN) 时不受模块的错误信息影响整个系统, 因此模块默认下表这些错误代码为警告类信息。

错误代码	错误说明	排除方式
16#1800	CH1 计数器溢位 / 借位	检查计数值, 若不须此报警请由此模块 Parameters 设定的报警设定页面中关闭。 此错误状态可由下述任一方式清除: 清除计数器 (Reset)、预置计数器 (Preset)、重启模块、重启 DHCCNT 指令。
16#1801	CH2 计数器溢位 / 借位	
16#1802	CH1 线性计数超出设定上下限	检查通道一 / 通道二计数值。 硬件内部计数值依然持续计数, 当内部硬件计数值回到最大计数值至最小计数值范围内, 错误代码将自行清除。
16#1803	CH2 线性计数超出设定上下限	
16#1804	CH1 SSI 编码器位置变化量超出限制	检查是否受干扰、检查最大位移量限制设定值是否与实际操作转速匹配; 当下一笔读取位置值回到正常变化量范围内, 错误代码将自行清除。
16#1805	CH2 SSI 编码器位置变化量超出限制	
16#1806	CH1 SSI 通讯异常	检查 DHCCNT 计数器状态, 若为同位检查错误, 检查是否有干扰存在, 以及数据格式是否正确设定; 若为 SSI 通讯异常, 检查配线方式是否有断线情形、编码器供电是否正常以及数据格式设定是否正确。
16#1807	CH2 SSI 通讯异常	
16#1808	CH1 SSI 绝对位置跨越零点	检查 SSI 绝对型编码器操作范围, 若不须此报警请由模块设定的报警设定页面中关闭。 此错误状态可由下述任一方式清除: 清除计数器 (Reset)、预置计数器 (Preset)、重启模块、重启 DHCCNT 指令。
16#1809	CH2 SSI 绝对位置跨越零点	

A.4.4 AS02/04PU 定位模块故障排除

A.4.4.1 ERROR 灯闪烁 (亮 0.2 秒、暗 0.2 秒)

错误代码	错误说明	排除方式
16#1802	模块硬件错误	退回原厂检修

A.5 CPU 模块错误代码对应灯号及状态说明

A. 字段简介

- a. 错误代码：该错误发生时系统所产生之错误代码。
- b. 说明：该错误之说明。
- c. CPU 状态：该错误发生时，CPU 主机的状态变化。
 - 停止：发生该错误时 CPU 停止运行。
 - 持续：发生该错误时 CPU 持续运行。
- d. 灯号状态：该错误发生时的主机灯号变化。
 - ERROR：系统错误灯号。

● 各灯号说明

模块名称	灯号	说明
CPU	Error LED	<p>指示 CPU 的错误或警告状态，其灯号显示共有底下五种状态，前三种直亮快速与一般显示方式为错误类别信息，当此类错误发生时，用户须排除后，才可再重新让 CPU 运行；最后一种慢速显示为警告类型信息，用户不需立即处理，但最好在设备机台未运行状态下，进行排除之动作。</p> <p>错误类：</p> <ul style="list-style-type: none"> 常亮：系统运行过程中发生严重错误。 快速闪烁：电源不稳定或硬件组件自我检查错误。(亮 0.2 秒、暗 0.2 秒) 一般闪烁：程序错误或系统无法执行之错误发生。(亮 0.5 秒、暗 0.5 秒) <p>警告类：</p> <ul style="list-style-type: none"> 慢速闪烁：警告信息发生，系统可持续运行。(亮 1 秒、暗 3 秒) 无：有警告信息发生但不影响运行，仅由编辑软件提示。

A.5.1 CPU 模块错误代码对应灯号及状态

CPU 相关错误

错误代码 (16#)	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
140E	远程模块的扩展模块超出 8 台限制	停止			√		
1500	远程模块断线警告信息	持续				√	
1600	主机扩展模块 ID 配置超出范围	停止			√		
1601	主机扩展模块无法配置 ID	停止			√		
1602	主机扩展模块 ID 配置重复	停止			√		
1603	主机扩展模块无法进入运行模式	停止			√		
1604	主机扩展模块通讯超时	停止			√		
2000	主机内部存储器检测错误	停止			√		
2001	主机外部内存检测错误	停止			√		
2002	万年历无法计时 (电池灯号闪烁)	持续					√

错误代码 (16#)	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
2003	电池低电压 (电池灯号常亮)	持续					√
2004	曾经发生过 24V 低电压又恢复	持续		√			
2100	MODBUS TCP 联机数超出范围	持续			√		
2200	实际 IO 模块不符合配置设定	停止			√		
2201	通讯模块超出 4 台限制	停止			√		
2202	定位模块超出 8 台限制	停止			√		
2203	所有扩展模块超出 32 台限制	停止			√		
2500	装置描述版本 (DDF) 与控制器本体不兼容	持续					√
2501	SSI 编码器未连接到 PLC	持续					√
2502	SSI 编码器设定单圈+多圈超出设定范围(最大为 32 bits)	持续					√
2503	脉冲输出轴发生错误	持续					√

EtherCAT 相关错误

错误代码 (16#)	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
1	EtherCAT 通讯遗失	持续			√		
2	EtherCAT 数据交换遗失	持续			√		
4	EtherCAT 网卡名称错误	持续			√		
5	EtherCAT 从站初始化失败	持续			√		
6	从站 Vendor ID 不匹配	持续			√		
7	从站 Product ID 不匹配	持续			√		

A.5.2 模拟与温度模块错误代码对应灯号

错误代码	说明	灯号状态	
		A → D / D → A / A ↔ D	ERROR
16#1605	模块内部硬件错误	OFF	常亮
16#1607	模块外部电源错误	OFF	常亮
16#1608	校正值或 CJC 补偿异常	OFF	常亮
16#1801*1	模块外部电源错误	OFF	闪烁
16#1802*1	模块硬件错误	OFF	闪烁
16#1804*1	内部错误 · 出厂校正异常	RUN : 闪烁 STOP : OFF	闪烁

错误代码	说明	灯号状态	
		A → D / D → A / A ↔ D	ERROR
16#1807*1	CJC 温度错误	OFF	闪烁
16#1808	模拟 (温度) 输入通道 1 超出范围值	RUN : 闪烁 STOP : OFF	闪烁
16#1809	模拟 (温度) 输入通道 2 超出范围值		
16#180A	模拟 (温度) 输入通道 3 超出范围值		
16#180B	模拟 (温度) 输入通道 4 超出范围值		
16#180C	模拟 (温度) 输入通道 5 超出范围值		
16#180D	模拟 (温度) 输入通道 6 超出范围值		
16#180E	模拟 (温度) 输入通道 7 超出范围值		
16#180F	模拟 (温度) 输入通道 8 超出范围值		

*1：此四个错误码默认为警告类信息，若需更改为错误类信息，请由模块硬件组态进行修改。

A.5.3 AS02LC 称重模块错误代码对应灯号

错误代码	说明	灯号状态	
		A → D	ERROR
16#1605	模块内部硬件 (含驱动板) 错误	OFF	常亮
16#1607	模块外部电源错误	OFF	常亮
16#1801*1	模块外部电源错误	OFF	闪烁
16#1802*1	模块内部硬件错误	OFF	闪烁
16#1807*1	模块驱动板错误	OFF	闪烁
16#1808	通道 1 输入超出量测范围或 SEN 电压错误	RUN : 闪烁 STOP : OFF	闪烁
16#1809	通道 1 超出重量上限		
16#180A	通道 1 调校错误		
16#180B	通道 2 输入超出量测范围或 SEN 电压错误		
16#180C	通道 2 超出重量上限		
16#180D	通道 2 调校错误		
-	上电时未收到主机检测命令	OFF	闪烁一次或两次，两秒后重复闪烁

*1：此三个错误码默认为警告类信息，若需更改为错误类信息，请由模块硬件组态进行修改。

A.5.4 AS02HC 高速计数模块错误代码对应灯号

错误代码	说明	ERROR 灯号状态	
		常亮	一般闪烁
16#1605	计数值停电保持异常	V	
16#1606	模块设定值停电保持异常	V	
16#1607	模块设定值配置内容错误	V	
16#1800	CH1 计数器溢位 / 借位		V
16#1801	CH2 计数器溢位 / 借位		V
16#1802	CH1 线性计数超出设定上下限		V
16#1803	CH2 线性计数超出设定上下限		V
16#1804	CH1 SSI 编码器位置变化量超出限制		V
16#1805	CH2 SSI 编码器位置变化量超出限制		V
16#1806	CH1 SSI 通讯异常		V
16#1807	CH2 SSI 通讯异常		V
16#1808	CH1 SSI 绝对位置跨越零点		V
16#1809	CH2 SSI 绝对位置跨越零点		V

A.5.5 AS02/04PU 定位模块错误代码对应灯号

错误代码	说明	灯号状态	
		A ↔ D	ERROR
16#1802	模块内部硬件错误	OFF	闪烁

MEMO